UniVoxel: Fast Inverse Rendering by Unified Voxelization of Scene Representation

Shuang Wu*[1], Songlin Tang*[1], Guangming Lu[1], Jianzhuang Liu[2], and Wenjie Pei†[1]

[1] Harbin Institute of Technology, Shenzhen
[2] Shenzhen Institute of Advanced Technology, Chinese Academy of Sciences
{wushuang9811,wenjiecoder}@outlook.com
tangsonglin@stu.hit.edu.cn
luguangm@hit.edu.cn
jz.liu@siat.ac.cn

Abstract. Typical inverse rendering methods focus on learning implicit neural scene representations by modeling the geometry, materials and illumination separately, which entails significant computations for optimization. In this work we design a Unified Voxelization framework for explicit learning of scene representations, dubbed *UniVoxel*, which allows for efficient modeling of the geometry, materials and illumination jointly, thereby accelerating the inverse rendering significantly. To be specific, we propose to encode a scene into a latent volumetric representation, based on which the geometry, materials and illumination can be readily learned via lightweight neural networks in a unified manner. Particularly, an essential design of *UniVoxel* is that we leverage local Spherical Gaussians to represent the incident light radiance, which enables the seamless integration of modeling illumination into the unified voxelization framework. Such novel design enables our *UniVoxel* to model the joint effects of direct lighting, indirect lighting and light visibility efficiently without expensive multi-bounce ray tracing. Extensive experiments on multiple benchmarks covering diverse scenes demonstrate that *UniVoxel* boosts the optimization efficiency significantly compared to other methods, reducing the per-scene training time from hours to 18 minutes, while achieving favorable reconstruction quality. Code is available at https://github.com/freemantom/UniVoxel.

Keywords: Inverse Rendering · Neural Rendering · Relighting

1 Introduction

Inverse rendering is a fundamental problem in computer vision and graphics, which aims to estimate the scene properties including geometry, materials and illumination of a 3D scene from a set of multi-view 2D images. With the great

* Equal contribution.
† Corresponding author.

success of Neural Radiance Fields (NeRF) [23] in novel view synthesis, it has been adapted to inverse rendering by learning implicit neural representations for scene properties. A prominent example is NeRD [3], which models materials as the spatially-varying bi-directional reflectance distribution function (SV-BRDF) using MLP networks. Another typical way of learning implicit representations for inverse rendering [8, 44, 46] is to first pre-train a NeRF or a surface-based model like IDR [40] or NeuS [33] to extract the scene geometry, and then they estimate the materials as well as the illumination by learning implicit neural representations for the obtained surface points. A crucial limitation of such implicit learning methods is that they seek to model each individual scene property by learning a complicated mapping function from spatial locations to the property values, which entails significant computations since modeling of each property demands learning a deep MLP network with sufficient modeling capacity. Meanwhile, expensive multi-bounce ray tracing is typically required for modeling illumination. As a result, these methods suffer from low optimization efficiency, typically requiring several hours or even days of training time for each scene, which limits their practical applications.

It has been shown that modeling scenes with explicit representations [5,10,30] rather than implicit ones is an effective way of accelerating the optimization of NeRF. TensoIR [14] makes the first attempt at explicit learning for inverse rendering, which extends TensoRF [5] and performs VM decomposition to factorize 3D spatially-varying scene features into tensor components. While TensoIR accelerates the optimization substantially compared to the implicit learning methods, it follows the typical way [8, 25, 35, 44] to model the illumination by learning environment maps which results in two important limitations. First, the methods based on environment maps have to simulate the lighting visibility and indirect lighting for each incident direction of a surface point, which still incurs heavy computational burden. Second, it is challenging for these methods to deal with complex illumination in real-world scenarios due to the limited modeling capability of the environment maps.

In this work, we propose to boost the optimization efficiency of inverse rendering by unified learning of all scene properties via constructing explicit voxelization of scene representation. As shown in Fig. 1, we devise a Unified Voxelization framework for scene representation, dubbed *UniVoxel*, which encodes a scene into latent volumetric representations consisting of two essential components: 1) Signed Distance Function (SDF) field for capturing the scene geometry and 2) semantic field for characterizing the materials and illumination of the scene. As a result, our *UniVoxel* is able to estimate the materials and illumination of a scene based on the voxelization of the semantic field by learning lightweight MLP networks while the surface normal and opacity for an arbitrary 3D point can be easily derived from the voxelization of the SDF field. Thus, our *UniVoxel* is able to perform inverse rendering more efficiently than other methods, reducing the optimizing time from several hours to 18 minutes.

A crucial challenge of performing inverse rendering with explicit representation lies in the modeling of illumination. Previous methods typically represent

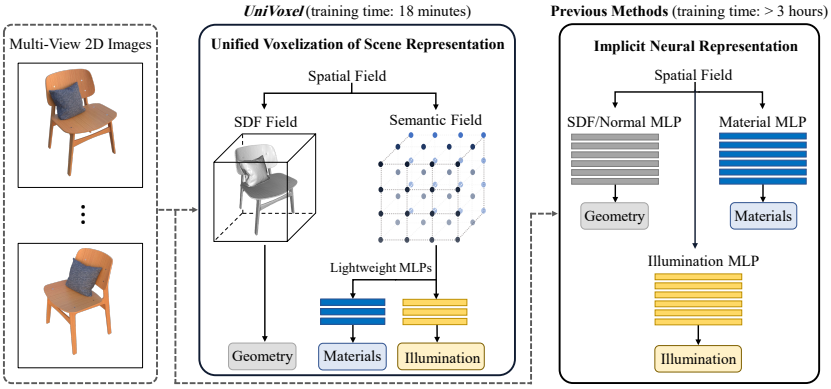

Fig. 1: Overview of the proposed *UniVoxel*. Typical methods [8, 44, 46] for inverse rendering learn implicit neural scene representations from spatial field by modeling the geometry, materials and illumination individually employing deep MLP networks. In contrast, our *UniVoxel* learns explicit scene representations by performing voxelization towards two essential scene elements: SDF field and semantic field, based on which the geometry, materials and illumination can be learned with lightweight networks in a unified manner, boosting the optimization efficiency of inverse rendering substantially.

the illumination as environment maps, which incurs significant computational cost due to multi-bounce ray tracing. In this work, we propose a unified illumination modeling mechanism, which leverages Spherical Gaussians (SG) [18, 31] to represent the local incident light radiance. In particular, we model the SG parameters by a unified learning manner with the modeling of geometry and materials, i.e., learning them from the voxelization of the semantic field by a lightweight MLP, which enables seamless integration of illumination modeling into the unified voxelization framework of our *UniVoxel*. Then we can efficiently query the incident light radiance from any direction at any position in the scene. A prominent advantage of the proposed illumination representation is that it can model direct lighting, indirect illumination and light visibility jointly without multi-bounce ray tracing, significantly improving the training efficiency. To conclude, we make the following contributions:

- We design a unified voxelization framework of scene representation, dubbed *UniVoxel*, which allows for efficient learning of all essential scene properties for inverse rendering in a unified manner, including the geometry, materials and illumination.
- We propose to model incident light field with Spherical Gaussians, which eliminates multi-bounce ray tracing and enables unified illumination modeling with other scene properties based on the learned voxelization of scene representation by our *UniVoxel*, substantially accelerating the training efficiency.
- Extensive experiments on various benchmarks show that our method achieves favorable reconstruction quality compared to other state-of-the-art approaches for inverse rendering while boosting the optimization efficiency significantly: 40× faster than MII [46] and over 12× faster than Nvdiffrec-mc [13].

2 Related Work

2.1 Inverse Rendering

Inverse rendering aims to reconstruct geometry, materials and illumination of the scene from observed images. Early works [2, 6, 7, 21, 26, 37] perform inverse rendering with a given triangular mesh as the fixed or initialized scene geometry representation. In contrast, Nvdiffrec [25] represents scene geometry as triangular mesh and jointly optimizes geometry, materials and illumination by a well-designed differentiable rendering paradigm. Nvdiffrec-mc [13] further incorporates ray tracing and Monte Carlo integration to improve reconstruction quality. Inspired by the success of NeRF [23], some methods [1, 28] utilize Neural Reflectance Fields to model the scene properties. PhySG [42] employs Spherical Gaussians to model environment maps. NMF [22] devises an optimizable microfacet material model. NeRFactor [44], L-Tracing [8] and MII [46] adopt the multi-stage framework to decompose the scene under complex unknown illumination. Some works apply inverse rendering to more challenging scenarios, such as photometric stereo [38], scattering object [45] and urban scenes [34]. Although achieving promising results, most of these works require several hours or even days to train for each scene, which limits their practical applications.

2.2 Explicit Representation

Learning implicit neural representations for scenes with MLP networks typically introduces substantial computation, leading to slow training and rendering. To address this limitation, explicit representation [10] and hybrid representation [5, 9, 20, 30] have been explored to model the radiance field for a scene. DVGO [30] employs dense voxel grids and a shallow MLP to model the radiance field. TensoRF [5] proposes VM decomposition to factorize 3D spatially-varying scene features to compact low-rank tensor components. Voxurf [36] combines DVGO [30] and NeuS [33] to achieve efficient surface reconstruction. The methods mentioned above are all used for the explicit representation of radiance field in static or dynamic scenes, but cannot be directly applied to inverse rendering task which requires explicit representation of geometry, materials and illumination simultaneously.

There is limited research on explicit representation for inverse rendering. Neural-PBIR [29] pre-computes lighting visibility and distills physics-based materials from the radiance field. GS-IR [19] and Relightable 3D Gaussian [11] introduce Gaussian Splatting (GS) [16] to scene relighting, but the quality of the scene geometry predicted by these point-based methods is limited. TensoIR [14] extends TensoRF [5] to inverse rendering. However, it does not model the illumination based on the learned explicit representations, but follows the traditional way [8] to represent the illumination as environment maps, which incurs heavy computational cost for simulating lighting visibility and indirect lighting. In this paper, we devise a unified voxelization framework for efficient modeling of the geometry, materials and illumination in a unified manner, reducing the per-scene optimization time to 18 minutes.

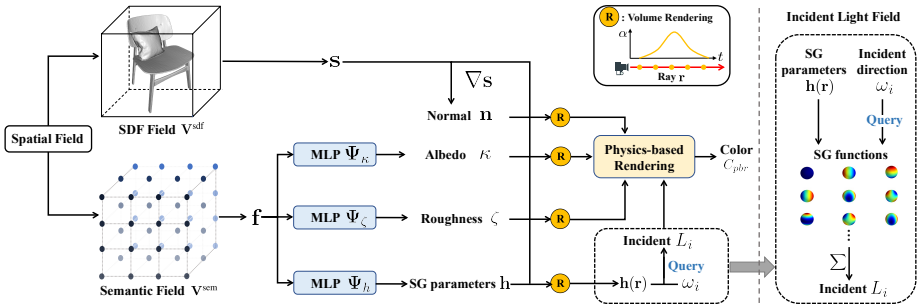

Fig. 2: Overall framework of the proposed *UniVoxel*. It performs voxelization towards the SDF field and semantic field to obtain explicit scene representations. The learned volumetric SDF field focuses on capturing the scene geometry while the semantic field characterizes the materials and illumination for the scene. As a result, our *UniVoxel* is able to learn the materials (including the albedo and roughness) and illumination using lightweight MLP networks based on the voxelization of the semantic field. Meanwhile, the surface normal and opacity for an arbitrary 3D point can be easily derived from the voxelization of the SDF field. Hence, our model is able to learn all these scene properties efficiently in a unified manner. In particular, we leverage Spherical Gaussians (SG) to model the incident light field, which allows for unified learning of the illumination with other scene properties based on the voxelization of the scene representation.

3 Method

3.1 Overview

We devise a Unified Voxelization framework (*UniVoxel*) for explicit scene representation learning, which allows for efficient learning of essential scene properties including geometry, materials and illumination in a unified manner, thereby improving the optimization efficiency of inverse rendering significantly. Fig. 2 illustrates the overall framework of our *UniVoxel*. It encodes a scene by performing voxelization toward two essential scene elements: 1) Signed Distance Function (SDF) field for capturing the geometry and 2) semantic field for characterizing the materials and illumination. We model both of them as learnable embeddings for each voxel. As a result, we can obtain the SDF value and semantic feature for an arbitrary position in 3D space by trilinear interpolation efficiently.

For a sampled point along a camera ray, our *UniVoxel* estimates the albedo, roughness and illumination based on the voxelization of the semantic field by learning lightweight MLP networks. Meanwhile, the surface normal and opacity of the sampled point can be easily derived from the voxelization of the SDF field. Leveraging these obtained scene properties, our *UniVoxel* performs volumetric physics-based rendering to reconstruct the 2D appearance of the scene.

3.2 Physics-Based Rendering

Our model renders a 3D scene into 2D images by applying the classical physics-based rendering formulation [15]. Formally, for a surface point $\mathbf{x} \in \mathbb{R}^3$, we calcu-

late the outgoing radiance, namely the rendered color $C(\mathbf{x}, \omega_o)$ in 2D, in direction ω_o as follows:

$$C(\mathbf{x}, \omega_o) = \int_{\Omega} L_i(\mathbf{x}, \omega_i)\, f_r(\mathbf{x}, \omega_i, \omega_o)\, (\omega_i \cdot \mathbf{n}(\mathbf{x}))d\omega_i, \tag{1}$$

where $\mathbf{n}(\mathbf{x})$ is the surface normal at $\mathbf{x}$ and $L_i(\mathbf{x}, \omega_i)$ denotes the incident light radiance in direction ω_i. Ω denotes the hemisphere satisfying $\{\omega_i : \omega_i \cdot \mathbf{n}(\mathbf{x}) > 0\}$, while f_r is the BRDF describing the materials at the surface point $\mathbf{x}$. In this work, we adopt the Simplified Disney BRDF model [4] which derives BRDF from the spatially-varying diffuse albedo $\kappa(\mathbf{x})$ and roughness $\zeta(\mathbf{x})$.

Unlike the typical methods for inverse rendering that estimate the scene properties in Eq. (1) including the geometry, materials and illumination based on implicit neural representation learning, our *UniVoxel* obtains these properties by volumetric rendering along camera rays based on the voxelization of scene representation. Specifically, given a camera ray $\mathbf{r}$ with origin $\mathbf{o}$, direction $\mathbf{d}$ and P sampled points $\{\mathbf{x}_i = \mathbf{o} + t_i\mathbf{d} | i = 1, ..., P\}$, we follow NeuS [33] to represent the geometry as a zero-level set based on the learned voxelization of the SDF field, and calculate the opacity value α_i at point $\mathbf{x}_i$ by:

$$\alpha_i = \max(\frac{\sigma(\mathbf{s}(\mathbf{x}_i)) - \sigma(\mathbf{s}(\mathbf{x}_{i+1}))}{\sigma(\mathbf{s}(\mathbf{x}_i))}, 0), \quad \sigma(\mathbf{s}(\mathbf{x}_i)) = (1 + e^{-d\mathbf{s}(\mathbf{x}_i)})^{-1}, \tag{2}$$

where $\mathbf{s}(\mathbf{x}_i)$ is the signed distance at $\mathbf{x}_i$ and $\frac{1}{d}$ is the standard deviation of $\sigma(\mathbf{s}(\mathbf{x}_i))$. Then we compute the albedo $\kappa(\mathbf{r})$ along the camera ray $\mathbf{r}$ by volume rendering [23] as:

$$\kappa(\mathbf{r}) = \sum_{i=1}^{P} T_i \alpha_i \kappa_i, \tag{3}$$

where $T_i = \prod_{j=1}^{i-1}(1-\alpha_j)$ denotes the accumulated transmittance. We can obtain the roughness $\zeta(\mathbf{r})$ and surface normal $\mathbf{n}(\mathbf{r})$ in the same way. Thus, the essential of such modeling boils down to learning the geometry and materials for sampled points from the voxelization of the SDF and semantic fields, which is elaborated in Sec. 3.3. Besides, we will also explicate how to derive the incident light radiance $L_i(\mathbf{x}, \omega_i)$ in Eq. (1) from the voxelization of the semantic field in Sec. 3.4.

3.3 Unified Voxelization of Scene Representation

Our *UniVoxel* constructs a unified voxelization framework for explicit learning of scene representations, which allows for efficient estimation of scene properties and fast inverse rendering. To be specific, our *UniVoxel* performs voxelization toward the SDF and semantic fields separately to capture different scene properties. The SDF field focuses on capturing scene geometry while the semantic field characterizes scene materials and illumination. Formally, we learn volumetric embeddings for both of them: $\mathbf{V}^{\text{sdf}} \in \mathbb{R}^{1 \times N_x \times N_y \times N_z}$ for the SDF field and $\mathbf{V}^{\text{sem}} \in \mathbb{R}^{C \times N_x \times N_y \times N_z}$ for the semantic field, where N_x, N_y and N_z denote the

resolution of voxelization and C is the feature dimension of semantics. The SDF value $\mathbf{s}(\mathbf{x})$ and semantic features $\mathbf{f}(\mathbf{x})$ for a position $\mathbf{x} \in \mathbb{R}^3$ in the space can be queried by trilinear interpolation $\mathcal{F}_{\text{interp}}$ on its eight neighboring voxels:

$$\mathbf{s}(\mathbf{x}) = \mathcal{F}_{\text{interp}}(\mathbf{x}, \mathbf{V}^{\text{sdf}}), \quad \mathbf{f}(\mathbf{x}) = \mathcal{F}_{\text{interp}}(\mathbf{x}, \mathbf{V}^{\text{sem}}). \tag{4}$$

The surface normal at position $\mathbf{x}$ can be easily derived based on the learned SDF field of the neighboring samples. For example, we approximate the x-component of the surface normal of $\mathbf{x}$ as:

$$\mathbf{n}_x(\mathbf{x}) = (\mathbf{s}(\mathbf{x} + [v, 0, 0]) - \mathbf{s}(\mathbf{x} - [v, 0, 0]))/(2v), \tag{5}$$

where v denotes the size of one voxel. $\mathbf{n}_y(\mathbf{x})$ and $\mathbf{n}_z(\mathbf{x})$ can be calculated in the similar way along the dimension y and z, respectively.

Based on the learned volumetric semantic field, our *UniVoxel* models the albedo and roughness using two lightweight MLP networks:

$$\kappa(\mathbf{x}) = \mathbf{\Psi}_\kappa(\mathbf{f}(\mathbf{x}), \mathbf{x}), \quad \zeta(\mathbf{x}) = \mathbf{\Psi}_\zeta(\mathbf{f}(\mathbf{x}), \mathbf{x}), \tag{6}$$

where $\kappa(\mathbf{x})$ and $\zeta(\mathbf{x})$ are the learned albedo and roughness at the position $\mathbf{x}$, respectively.

Memory optimization by multi-resolution hash encoding. Our *UniVoxel* can be readily optimized w.r.t. the memory usage by directly applying multi-resolution hash encoding [24]. The sparsity of hash voxel grids enables high spatial resolution of voxelization with low memory cost. For a point $\mathbf{x}$ in space, its semantic features can be represented as the concatenation of hash encoding from L resolution levels: $\mathbf{f}(\mathbf{x}) = \{\mathbf{f}^i(\mathbf{x})\}_{i=1}^L$. For each resolution, the learnable feature $\mathbf{f}^i$ of a voxel vertex can be quickly queried from the hash table and the feature embedding $\mathbf{f}^i(\mathbf{x})$ of position $\mathbf{x}$ is obtained by trilinear interpolation. The concatenated multi-resolution semantic features $\mathbf{f}(\mathbf{x})$ are fed to three tiny MLP networks to decode into SDF, albedo and roughness respectively.

3.4 Illumination Modeling in the Unified Voxelization

We present two feasible ways to model illumination based on the learned voxelization of scene representations. We first follow classical methods [8, 14, 42, 44, 46] that learn an environment map to model lighting. Then we propose a unified illumination modeling method by leveraging Spherical Guassians to represent the incident light radiance, which enables seamless integration of illumination modeling into the unified voxelization framework of our *UniVoxel*, leading to more efficient optimization.

Learning the environment map. A typical way of modeling illumination is to represent lighting as an environment map [8, 14, 42, 44, 46], assuming that all lights come from an infinitely faraway environment. Different from other methods [44, 46] using an MLP network to predict light visibility, we compute

it by volumetric integration. To be specific, considering the surface point $\mathbf{x}$ and an incident direction ω_i, the light visibility $\mathbf{v}(\mathbf{x}, \omega_i)$ is calculated as:

$$\mathbf{v}(\mathbf{x}, \omega_i) = 1 - \sum_{i=1}^{N_l} \alpha_i \prod_{j<i} (1 - \alpha_j), \tag{7}$$

where N_l is the number of sampled points along the ray $\mathbf{r}_i = \mathbf{x} + \omega_i$. Benefiting from the efficiency of the voxel-based representation, $\mathbf{v}(\mathbf{x}, \omega_i)$ can be computed in an online manner. However, sampling a larger number of incident lights or considering multi-bounce ray tracing still results in significant computational cost. To alleviate this issue, we propose to utilize the light field with volumetric representation to model incident radiance.

Unified illumination modeling based on Spherical Gaussians. Illumination can be also modeled by learning the light field by implicit neural representation [39, 41], which employs an MLP network to learn a mapping function taking a 3D position $\mathbf{x}$ and incident direction ω_i as input, and producing the light field comprising direct lighting, indirect lighting and light visibility. Such implicit modeling way also suffers from the low optimization efficiency since it demands a deep MLP with sufficient modeling capacity to model the complicated mapping function.

In contrast to above implicit neural representation learning of illumination, we propose to leverage Spherical Gaussians (SG) to represent the incident light field based on the learned unified voxelization framework of our *UniVoxel*. SG have been explored to model illumination [42, 46] by representing the entire scene's environment map, which requires expensive multi-bounce ray tracing. In contrast, our *UniVoxel* predicts a set of SG parameters for each position $\mathbf{x}$ in 3D space to model the incident radiance at that local position. Formally, the parameters of a SG lobe are denoted as $\mathbf{h} = \{a \in \mathbb{R}^3, \lambda \in \mathbb{R}, \mu \in \mathbb{S}^2\}$. Given an incident direction w_i at the position $\mathbf{x}$, the incident light radiance can be obtained by querying the SG functions as the sum of SG lobes:

$$L_i(\mathbf{x}, \omega_i) = \sum_{i=0}^{k} a e^{\lambda(\mu \cdot w_i - 1)}, \tag{8}$$

where k denotes the number of SG lobes. Herein, we model the essential component of the SG parameters $\mathbf{h}$ in a unified learning manner with the modeling of the geometry and materials as shown in Sec. 3.3 based on the voxelization of the scene representation:

$$\mathbf{h}(\mathbf{x}) = \mathbf{\Psi}_h(\mathbf{f}(\mathbf{x}), \mathbf{x}), \tag{9}$$

where $\mathbf{\Psi}_h$ denotes a lightweight MLP network. Then we obtain the $\mathbf{h}(\mathbf{r})$ along the camera ray $\mathbf{r}$ by the volume rendering shown in Eq. (3) with κ_i replaced by $\mathbf{h}(\mathbf{x}_i)$. Thus, we can efficiently query incident light radiance from an arbitrary direction at a surface point. As a result, our *UniVoxel* is able to integrate illumination modeling into the constructed unified voxelization framework, which boosts the optimization efficiency substantially.

Note that some prior works [12, 27] use Spherical Harmonics (SH) instead of SG to model illumination with the crucial limitation that they fail to recover the high-frequency lighting. We will compare these two ways in Sec. 4.4.

Extension to varying illumination conditions. Thanks to the flexibility of the proposed illumination model, our *UniVoxel* can be easily extended to varying illumination conditions, where each view of the scene can be captured under different illuminations. Specifically, given N_v multi-view images of a scene, we maintain a learnable view embedding $\mathbf{e} \in \mathbf{R}^{N_v \times C_v}$, where C_v is the dimension of the view embedding. Then we employ the view embedding of current view as the additional input of $\mathbf{\Psi}_h$ to predict the SG parameters at each position, so the Eq. (9) is modified as:

$$\mathbf{h}(\mathbf{x}) = \mathbf{\Psi}_h(\mathbf{f}(\mathbf{x}), \mathbf{x}, \mathbf{e}_\mathbf{x}). \tag{10}$$

Thus, our *UniVoxel* is able to model the view-varying illumination conditions.

3.5 Optimization

Our *UniVoxel* is optimized with three types of losses in an end-to-end manner.

Reconstruction loss. Similar to other inverse rendering methods [8, 44, 46], we compute the reconstruction loss between the physics-based rendering colors C_{pbr} and the ground truth colors C_{gt}. To ensure a stable geometry during training, we use an extra radiance field taking features $\mathbf{f}$, position $\mathbf{x}$, and normal $\mathbf{n}$ as inputs to predict colors C_{rad}. Thus, the reconstruction loss is formulated as:

$$\mathcal{L}_{\text{rec}} = \lambda_{\text{pbr}} \|C_{\text{pbr}} - C_{\text{gt}}\|_2^2 + \lambda_{\text{rad}} \|C_{\text{rad}} - C_{\text{gt}}\|_2^2, \tag{11}$$

where λ_{pbr} and λ_{rad} are the loss weights.

Smoothness constraints. We apply a smoothness loss to regularize the albedo near the surfaces:

$$\mathcal{L}_{s-\kappa} = \sum_{\mathbf{x}_{surf}} \|\mathbf{\Psi}_\kappa(\mathbf{x}_{surf}) - \mathbf{\Psi}_\kappa(\mathbf{x}_{surf} + \epsilon)\|_2^2. \tag{12}$$

where ϵ is a random variable sampled from a normal distribution. Similar regularizations are conducted for normal $\mathcal{L}_{s-n}$ and roughness $\mathcal{L}_{s-\zeta}$. Thus, the smoothness loss is formulated as:

$$\mathcal{L}_{\text{smo}} = \lambda_\kappa \mathcal{L}_{s-\kappa} + \lambda_\zeta \mathcal{L}_{s-\zeta} + \lambda_n \mathcal{L}_{s-n}, \tag{13}$$

where λ_κ, λ_ζ and λ_n are balancing weights for the different terms.

Illumination regularization. Neural incident light field could lead to material-lighting ambiguity [39] due to the lack of constraints. We propose two regularization constraints to alleviate this ambiguity. First, we encourage a smooth variation of lighting conditions between adjacent surface points by applying a smoothing regularization on the Spherical Gaussian parameters:

$$\mathcal{L}_{\text{sg}} = \sum_{\mathbf{x}_{surf}} \|\mathbf{h}(\mathbf{x}_{surf}) - \mathbf{h}(\mathbf{x}_{surf} + \epsilon)\|_2^2. \tag{14}$$

Since the incident light is primarily composed of direct lighting, which is mostly white lighting [25], the second regularization of illumination is performed on the incident light by penalizing color shifts:

$$\mathcal{L}_{\text{white}} = |L_i(\mathbf{x}, \omega_i) - \overline{L_i(\mathbf{x}, \omega_i)}|, \tag{15}$$

where $\overline{L_i(\mathbf{x}, \omega_i)}$ denotes the average of the incident intensities along the RGB channel. Thus, the illumination regularization is formulated as:

$$\mathcal{L}_{\text{reg}} = \lambda_{\text{sg}}\mathcal{L}_{\text{sg}} + \lambda_{\text{white}}\mathcal{L}_{\text{white}}, \tag{16}$$

where λ_{sg} and λ_{white} are the weights of regularization loss.

Combing all the losses together, our *UniVoxel* is optimized by minimizing:

$$\mathcal{L} = \mathcal{L}_{\text{rec}} + \mathcal{L}_{\text{smo}} + \mathcal{L}_{\text{reg}}. \tag{17}$$

4 Experiments

4.1 Dataset

We conduct experiments on both synthetic and real-world datasets for evaluation. First, we select 4 challenging scenes from the MII synthetic dataset [46] for experiments. Each scene consists of 100-200 training images and 200 validation images from novel viewpoints. We show both the quantitative and qualitative results for the reconstructed albedo, roughness, novel view synthesis (NVS) and relighting. Furthermore, we evaluate our approach on 5 scenes of the NeRD real-world dataset [3]. To compare the quality of reconstructed geometry, we also conducted experiments on the Shiny Blender dataset [32], and the results are presented in Sec. D of the appendix.

4.2 Implementation Details

To calculate the outgoing radiance $C(\mathbf{x}, \omega_o)$ by Eq. (1) using a finite number of incident lights, we utilize Fibonacci sampling over the half sphere to sample incident lights for each surface point, and the sampling number is set to 128. As for relighting, the incident light field obtained from previous training is not applicable to the new illumination. Therefore, we adopt a similar procedure as the previous methods [14, 44], where we compute light visibility using Eq. (7) and consider only direct lighting.

We employ the coarse-to-fine training paradigm used in [30]. During the coarse stage, we only optimize the radiance field branch to accelerate training. The resolution of voxelization is set to 96^3 in the coarse stage and 160^3 in the fine stage. Each lightweight MLP network in our *UniVoxel* comprises 3 hidden layers with 192 channels. The number of the feature channels of the semantic field $\mathbf{V}^{\text{sem}}$ is 6. The sampling step size along a ray is set to half of the voxel size. The number of Spherical Gaussian lobes is $k = 16$. The weights of the losses

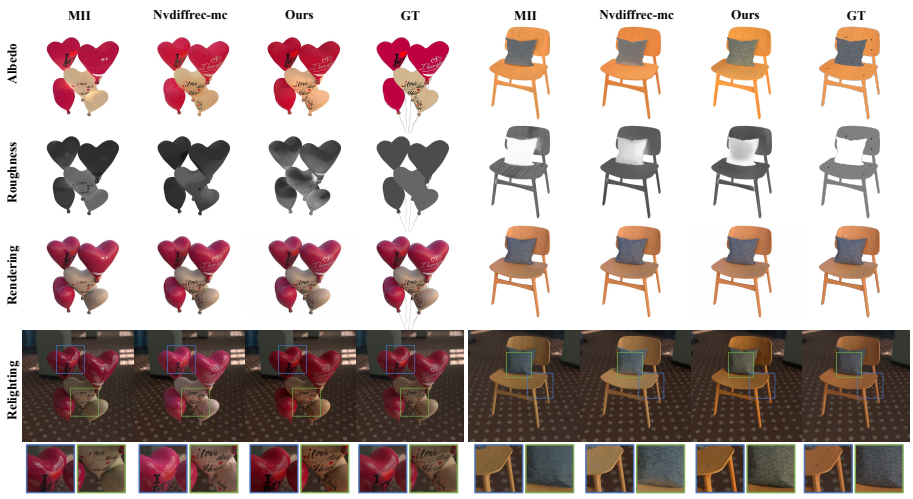

Fig. 3: Qualitative comparisons on 2 scenes from the MII synthetic dataset. More qualitative results are shown in the appendix.

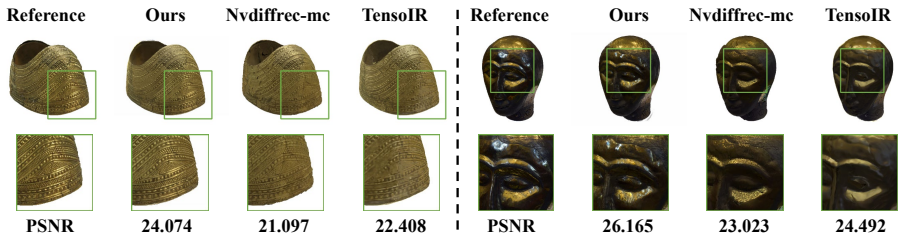

Fig. 4: Novel view synthesis results on 2 real-world scenes in a fixed environment from the NeRD dataset: *Ethiopian Head* and *Gold Cape*.

are tuned to be $\lambda_{pbr} = 1.0$, $\lambda_{rad} = 1.0$, $\lambda_n = 0.002$, $\lambda_\kappa = 0.0005$, $\lambda_\zeta = 0.0005$, $\lambda_{sg} = 0.0005$ and $\lambda_{white} = 0.0001$. We use the Adam optimizer [17] with a batch size of 8192 rays to optimize the scene representation for 10k iterations in both the coarse and fine stages. The base learning rate is 0.001 for MLP networks and 0.1 for $\mathbf{V}^{\text{sdf}}$ and $\mathbf{V}^{\text{sem}}$. And the learning rate for $\mathbf{V}^{\text{sdf}}$ is reduced to 0.005 in the fine stage. For the experiments on NeRD real-world dataset, we adopt the extended version of our illumination model, and set the dimension of the view embedding to $C_v = 6$. We apply a 1.5 power correction to the roughness during the relighting stage considering the optimization bias towards higher roughness values. The hyperparameter settings for multi-resolution hash encoding are presented in Sec. B of the appendix.

We run all experiments on a single RTX 3090 GPU, and the training time of other baselines is measured on the same machine.

Table 1: Quantitative evaluation on the MII synthetic dataset. We report the mean metrics over 200 novel validation views of all 4 scenes. The best result is denoted in bold, while the second best is underlined. '*UniVoxel (Hash)*' indicates the storage-optimized version by multi-resolution hash encoding [24] as explained in Sec. 3.3. Following previous works [8,25,44], we align the albedo with the ground truth before calculating the metrics to eliminate the scale ambiguity. The NVS results of our *UniVoxel* are generated by physics-based rendering for a fair comparison, although the ones generated by the radiance field are better.

Method	NVS			Albedo			Relighting			Roughness	Time↓
	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓	MSE↓	
NerFactor [44]	22.795	0.917	0.151	19.486	0.864	0.206	21.537	0.875	0.171	-	>2 days
MII [46]	30.727	0.952	0.085	28.279	0.935	0.072	28.674	0.950	0.091	0.008	14 hours
Nvdiffrec-mc [13]	34.291	0.967	0.067	29.614	0.945	0.075	24.218	0.943	0.078	0.009	4 hours
TensoIR [14]	35.804	0.979	0.049	**30.582**	0.946	0.065	29.686	0.951	0.079	0.015	3 hours
UniVoxel	**36.232**	**0.980**	0.049	29.933	**0.957**	**0.057**	29.445	**0.960**	**0.070**	**0.007**	**18 minutes**
UniVoxel(Hash)	35.873	**0.980**	**0.043**	29.994	0.950	0.073	**29.752**	0.958	0.080	0.011	26 minutes

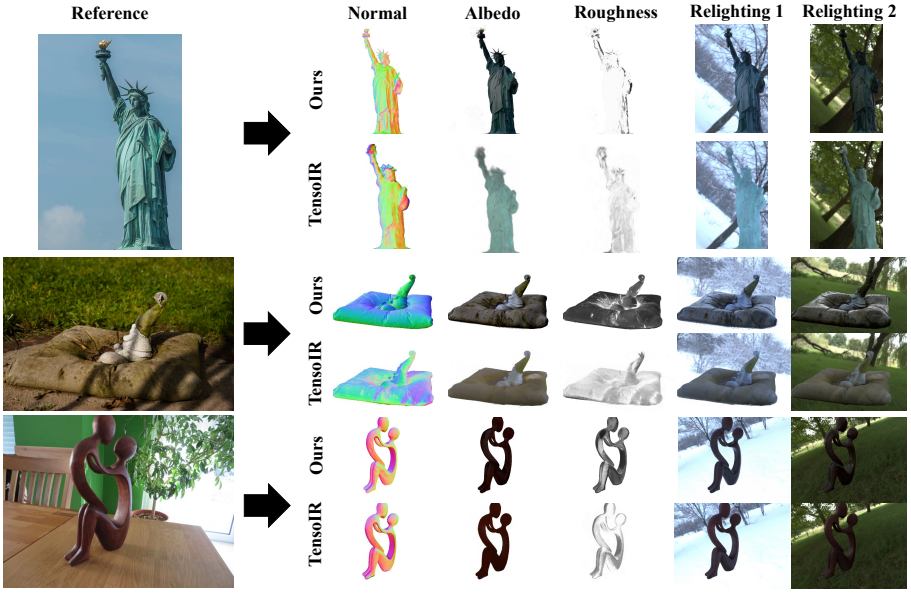

Fig. 5: Qualitative comparisons on 3 real-world scenes from the NeRD dataset. All the scenes are captured under varying illumination, which are more challenging. More qualitative results are shown in the appendix.

4.3 Comparisons with State-of-the-art Methods

Results on synthetic datasets. We compare our *UniVoxel* with NeRFactor [44], MII [46], Nvdiffrec-mc [13] and TensoIR [14] on the MII synthetic dataset, adopting Peak Signal-to-Noise Ratio (PSNR), Structural Similarity Index Measure (SSIM), and Learned Perceptual Image Patch Similarity (LPIPS) [43] as the quantitative metrics. As shown in Tab. 1, our *UniVoxel* outperforms other methods in most metrics while taking much less training time. We show the qual-

Table 2: Ablation studies of different illumination modeling methods. All methods are built on our unified voxelization framework to have a fair comparison.

Method	NVS PSNR↑	Albedo PSNR↑	Roughness MSE↓	Relighting PSNR↑	Time↓
Envmap(Mixture of 128 SG)	34.185	27.368	0.012	27.446	58 minutes
Envmap(256×512×3 learnable map)	35.042	29.369	0.010	**29.592**	2 hours
MLP (NeILF)	**36.355**	28.974	**0.007**	28.694	30 minutes
SH (order-3 with 48 paramaters)	35.328	29.185	0.020	28.981	22 minutes
SH (order-4 with 75 paramaters)	35.344	29.200	0.016	28.813	24 minutes
SG (12 lobes with 72 parameters)	36.177	29.746	0.008	29.148	**16 minutes**
SG (16 lobes with 96 parameters)	36.232	**29.933**	**0.007**	29.445	18 minutes

itative results in Fig. 3. It can be observed that MII fails to restore the high-frequency details on the albedo maps, such as the text on the air balloons, the nails on the chair and the textures on the pillow. Nvdiffrec-mc performs badly in specular areas. In contrast, our *UniVoxel* can produce accurate reconstructions and relighting. In '*UniVoxel(Hash)*', we also build our proposed unified voxelization based on multi-resolution hash encoding [24], which achieves higher relighting quality at the expense of a slight decrease in training speed.

Results on real-world datasets. To demonstrate the generalization ability of our method, we conduct experiments on 5 scenes from the NeRD real-world dataset. First, we evaluate on 2 scenes which are captured in a fixed environment. The qualitative and quantitative results of novel view synthesis are shown in Fig. 4. Both Nvdiffrec-mc and TensoIR suffer from various artifacts such as holes on *Gold Cape* and specular areas on *Ethiopian Head*, while our *UniVoxel* achieves better rendering results. Furthermore, we evaluate on the other 3 scenes captured under varying illumination. The qualitative results are shown in Fig. 5. Due to the difficulty of estimating the complex illumination in the wild via environment maps, TensoIR fails to recover the geometry and materials of the objects, thus causing poor relighting results. In contrast, our *UniVoxel* produces plausible normal, albedo and roughness, and achieves realistic relighting.

4.4 Ablation Studies for Illumination Modeling

To showcase the effectiveness of our proposed voxelization representation of the incident light field, we perform the ablation studies for illumination modeling. The quantitative results are reported in Tab. 2.

In 'Envmap(Mixture of 128 SG)', we represent the illumination of the scene as an environment map, parameterized by a mixture of 128 Spherical Gaussians. It is not surprising that the training time is much longer since it requires computing the lighting visibility via Eq. (7) for each incident light. And the rendering quality is also worse compared to our proposed illumination model. In 'Envmap (256×512×3 learnable map)', we use a learnable embedding with a resolution of 256×512 as the environment map. Compared to using a mixture of 128 SG, it has a stronger modeling capability for complex lighting and can achieve results close to our *UniVoxel*. However, the training speed is much slower, taking even up to two hours per scene.

Table 3: Ablation studies of each loss.

Method	NVS PSNR↑	Albedo PSNR↑	Roughness MSE↓
UniVoxel	36.232	29.933	0.007
w/o radiance field	36.304	29.604	0.008
w/o smoothness constraints L_{smo}	36.216	29.654	0.008
w/o regularization for white lights L_{white}	36.230	29.781	0.007
w/o regularization for SG L_{sg}	36.260	29.085	0.006

In 'MLP(NeILF)', we utilize the neural incident light field [39] to directly predict the incident light using an 8-layer MLP with a feature dimension of 128. Its training time is about twice as long as ours. Besides, without constraints for the incident light field, the lighting would be baked into the estimated albedo, resulting in a decrease in the quality of albedo maps. In contrast, thanks to the voxelization of the incident light, our *UniVoxel* can easily constrain the lighting conditions in adjacent regions of the scene, thereby alleviating the ambiguity between materials and illumination.

In 'SH', we represent the incident light field via Spherical Harmonics (SH) instead of Spherical Gaussians, and the SH coefficients are predicted by the MLP mentioned in Eq. (9). We employ 3-order and 4-order SH respectively, however, due to the difficulty of modeling high-frequency lighting with SH, the quality of the generated materials is comparatively poor, even using more parameters than SG. The qualitative comparisons of different illumination models are shown in Sec. C.1 of the appendix.

4.5 Effectiveness of Each Loss

We conduct experiments to explore the effectiveness of each loss used in our *UniVoxel*. As shown in Tab. 3, the performance of our method does not rely on the introduction of the extra radiance field, although it can provide more stability during training and slightly improve the quality of predicted materials. Smoothness constraints and regularization for white light also contribute to enhancing the reconstruction to a certain extent. While the regularization for Spherical Gaussians (SG) leads to a slight decrease in the results of novel view synthesis and roughness, it improves the quality of albedo. We further discuss the efficacy of L_{sg} in Sec. C.2 of the appendix.

5 Conclusion

We propose a unified voxelization framework for inverse rendering (*UniVoxel*). It learns explicit voxelization of scene representations, which allows for efficient modeling of all essential scene properties in a unified manner, boosting the inverse rendering significantly. Particularly, we leverage Spherical Gaussians to learn the incident light field, which enables the seamless integration of illumination modeling into the unified voxelization framework. Extensive experiments show that *UniVoxel* outperforms state-of-the-art methods in terms of both quality and efficiency.

Acknowledgements

This work was supported in part by the National Natural Science Foundation of China (U2013210, 62372133), in part by Shenzhen Fundamental Research Program (Grant NO. JCYJ20220818102415032), in part by Guangdong Basic and Applied Basic Research Foundation (2024A1515011706), in part by the Shenzhen Key Technical Project (NO. JSGG20220831092805009, JSGG20201103153802006, KJZD20230923115117033).

References

1. Bi, S., Xu, Z., Srinivasan, P., Mildenhall, B., Sunkavalli, K., Hašan, M., Hold-Geoffroy, Y., Kriegman, D., Ramamoorthi, R.: Neural reflectance fields for appearance acquisition. arXiv preprint arXiv:2008.03824 (2020)
2. Bi, S., Xu, Z., Sunkavalli, K., Kriegman, D., Ramamoorthi, R.: Deep 3d capture: Geometry and reflectance from sparse multi-view images. In: CVPR (2020)
3. Boss, M., Braun, R., Jampani, V., Barron, J.T., Liu, C., Lensch, H.: Nerd: Neural reflectance decomposition from image collections. In: ICCV (2021)
4. Burley, B., Studios, W.D.A.: Physically-based shading at disney. In: SIGGRAPH (2012)
5. Chen, A., Xu, Z., Geiger, A., Yu, J., Su, H.: Tensorf: Tensorial radiance fields. In: ECCV (2022)
6. Chen, W., Ling, H., Gao, J., Smith, E., Lehtinen, J., Jacobson, A., Fidler, S.: Learning to predict 3d objects with an interpolation-based differentiable renderer. In: NeurIPS (2019)
7. Chen, W., Litalien, J., Gao, J., Wang, Z., Fuji Tsang, C., Khamis, S., Litany, O., Fidler, S.: Dib-r++: learning to predict lighting and material with a hybrid differentiable renderer. In: NeurIPS (2021)
8. Chen, Z., Ding, C., Guo, J., Wang, D., Li, Y., Xiao, X., Wu, W., Song, L.: L-tracing: Fast light visibility estimation on neural surfaces by sphere tracing. In: ECCV (2022)
9. Fang, J., Yi, T., Wang, X., Xie, L., Zhang, X., Liu, W., Nießner, M., Tian, Q.: Fast dynamic radiance fields with time-aware neural voxels. In: SIGGRAPH Asia (2022)
10. Fridovich-Keil, S., Yu, A., Tancik, M., Chen, Q., Recht, B., Kanazawa, A.: Plenoxels: Radiance fields without neural networks. In: CVPR (2022)
11. Gao, J., Gu, C., Lin, Y., Zhu, H., Cao, X., Zhang, L., Yao, Y.: Relightable 3d gaussian: Real-time point cloud relighting with brdf decomposition and ray tracing. arXiv preprint arXiv:2311.16043 (2023)
12. Garon, M., Sunkavalli, K., Hadap, S., Carr, N., Lalonde, J.F.: Fast spatially-varying indoor lighting estimation. In: CVPR (2019)
13. Hasselgren, J., Hofmann, N., Munkberg, J.: Shape, light, and material decomposition from images using monte carlo rendering and denoising. In: Advances in Neural Information Processing Systems (2022)
14. Jin, H., Liu, I., Xu, P., Zhang, X., Han, S., Bi, S., Zhou, X., Xu, Z., Su, H.: Tensoir: Tensorial inverse rendering. arXiv preprint arXiv:2304.12461 (2023)
15. Kajiya, J.T.: The rendering equation. In: SIGGRAPH (1986)
16. Kerbl, B., Kopanas, G., Leimkühler, T., Drettakis, G.: 3d gaussian splatting for real-time radiance field rendering. ACM Transactions on Graphics (TOG) (2023)

17. Kingma, D.P., Ba, J.: Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980 (2014)
18. Li, Z., Shafiei, M., Ramamoorthi, R., Sunkavalli, K., Chandraker, M.: Inverse rendering for complex indoor scenes: Shape, spatially-varying lighting and svbrdf from a single image. In: CVPR (2020)
19. Liang, Z., Zhang, Q., Feng, Y., Shan, Y., Jia, K.: Gs-ir: 3d gaussian splatting for inverse rendering. arXiv preprint arXiv:2311.16473 (2023)
20. Liu, J.W., Cao, Y.P., Mao, W., Zhang, W., Zhang, D.J., Keppo, J., Shan, Y., Qie, X., Shou, M.Z.: Devrf: Fast deformable voxel radiance fields for dynamic scenes. arXiv preprint arXiv:2205.15723 (2022)
21. Liu, S., Li, T., Chen, W., Li, H.: Soft rasterizer: A differentiable renderer for image-based 3d reasoning. In: ICCV (2019)
22. Mai, A., Verbin, D., Kuester, F., Fridovich-Keil, S.: Neural microfacet fields for inverse rendering. In: ICCV (2023)
23. Mildenhall, B., Srinivasan, P., Tancik, M., Barron, J., Ramamoorthi, R., Ng, R.: Nerf: Representing scenes as neural radiance fields for view synthesis. In: ECCV (2020)
24. Müller, T., Evans, A., Schied, C., Keller, A.: Instant neural graphics primitives with a multiresolution hash encoding. ACM Transactions on Graphics (TOG) **41**(4), 1–15 (2022)
25. 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: CVPR (2022)
26. Nam, G., Lee, J.H., Gutierrez, D., Kim, M.H.: Practical svbrdf acquisition of 3d objects with unstructured flash photography. ACM Transactions on Graphics (TOG) **37**(6), 1–12 (2018)
27. Rudnev, V., Elgharib, M., Smith, W., Liu, L., Golyanik, V., Theobalt, C.: Nerf for outdoor scene relighting. In: ECCV (2022)
28. 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: CVPR (2021)
29. 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: ICCV (2023)
30. Sun, C., Sun, M., Chen, H.T.: Direct voxel grid optimization: Super-fast convergence for radiance fields reconstruction. In: CVPR (2022)
31. Tsai, Y.T., Shih, Z.C.: All-frequency precomputed radiance transfer using spherical radial basis functions and clustered tensor approximation. ACM Transactions on Graphics (TOG) (2006)
32. 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: CVPR (2022)
33. Wang, P., Liu, L., Liu, Y., Theobalt, C., Komura, T., Wang, W.: Neus: Learning neural implicit surfaces by volume rendering for multi-view reconstruction. In: NeurIPS (2021)
34. Wang, Z., Shen, T., Gao, J., Huang, S., Munkberg, J., Hasselgren, J., Gojcic, Z., Chen, W., Fidler, S.: Neural fields meet explicit geometric representations for inverse rendering of urban scenes. In: CVPR (2023)
35. Wu, H., Hu, Z., Li, L., Zhang, Y., Fan, C., Yu, X.: Nefii: Inverse rendering for reflectance decomposition with near-field indirect illumination. In: CVPR (2023)

36. Wu, T., Wang, J., Pan, X., Xu, X., Theobalt, C., Liu, Z., Lin, D.: Voxurf: Voxel-based efficient and accurate neural surface reconstruction. arXiv preprint arXiv:2208.12697 (2022)
37. Xia, R., Dong, Y., Peers, P., Tong, X.: Recovering shape and spatially-varying surface reflectance under unknown illumination. ACM Transactions on Graphics (TOG) **35**(6), 1–12 (2016)
38. Yang, W., Chen, G., Chen, C., Chen, Z., Wong, K.Y.K.: Ps-nerf: Neural inverse rendering for multi-view photometric stereo. In: ECCV (2022)
39. 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: ECCV (2022)
40. Yariv, L., Kasten, Y., Moran, D., Galun, M., Atzmon, M., Ronen, B., Lipman, Y.: Multiview neural surface reconstruction by disentangling geometry and appearance. In: NeurIPS (2020)
41. Zhang, J., Yao, Y., Li, S., Liu, J., Fang, T., McKinnon, D., Tsin, Y., Quan, L.: Neilf++: Inter-reflectable light fields for geometry and material estimation. arXiv preprint arXiv:2303.17147 (2023)
42. Zhang, K., Luan, F., Wang, Q., Bala, K., Snavely, N.: Physg: Inverse rendering with spherical gaussians for physics-based material editing and relighting. In: CVPR (2021)
43. Zhang, R., Isola, P., Efros, A.A., Shechtman, E., Wang, O.: The unreasonable effectiveness of deep features as a perceptual metric. In: CVPR (2018)
44. 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) **40**(6), 1–18 (2021)
45. Zhang, Y., Xu, T., Yu, J., Ye, Y., Jing, Y., Wang, J., Yu, J., Yang, W.: Nemf: Inverse volume rendering with neural microflake field. In: ICCV (2023)
46. Zhang, Y., Sun, J., He, X., Fu, H., Jia, R., Zhou, X.: Modeling indirect illumination for inverse rendering. In: CVPR (2022)