TensoIR: Tensorial Inverse Rendering

Haian Jin*[1] Isabella Liu*[2] Peijia Xu[3] Xiaoshuai Zhang[2] Songfang Han[2]
Sai Bi[4] Xiaowei Zhou[1] Zexiang Xu†[4] Hao Su†[2]

[1] Zhejiang University [2] UC San Diego [3] Kingstar Technology Inc. [4] Adobe Research

Abstract

We propose TensoIR, a novel inverse rendering approach based on tensor factorization and neural fields. Unlike previous works that use purely MLP-based neural fields, thus suffering from low capacity and high computation costs, we extend TensoRF, a state-of-the-art approach for radiance field modeling, to estimate scene geometry, surface reflectance, and environment illumination from multi-view images captured under unknown lighting conditions. Our approach jointly achieves radiance field reconstruction and physically-based model estimation, leading to photo-realistic novel view synthesis and relighting results. Benefiting from the efficiency and extensibility of the TensoRF-based representation, our method can accurately model secondary shading effects (like shadows and indirect lighting) and generally support input images captured under single or multiple unknown lighting conditions. The low-rank tensor representation allows us to not only achieve fast and compact reconstruction but also better exploit shared information under an arbitrary number of capturing lighting conditions. We demonstrate the superiority of our method to baseline methods qualitatively and quantitatively on various challenging synthetic and real-world scenes.

1. Introduction

Inverse rendering is a long-standing problem in computer vision and graphics, aiming to reconstruct physical attributes (like shape and materials) of a 3D scene from captured images and thereby supporting many downstream applications such as novel view synthesis, relighting and material editing. This problem is inherently challenging and ill-posed, especially when the input images are captured in the wild under unknown illumination. Recent works [6, 7, 28, 41] address this problem by learning neural field representations in the form of multi-layer perceptrons (MLP) similar to NeRF [22]. However, pure MLP-based methods usually suffer from low capacity and high computational costs, greatly limiting the accuracy and efficiency of inverse rendering.

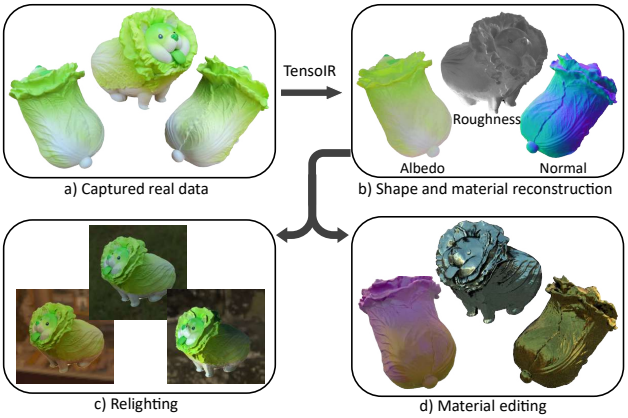

Figure 1. Given multi-view captured images of a real scene (a), our approach – TensoIR – is able to achieve high-quality shape and material reconstruction with high-frequency details (b). This allows us to render the scene under novel lighting and viewpoints (c), and also change its material properties (d).

In this work, we propose a novel inverse rendering framework that is efficient and accurate. Instead of purely using MLPs, we build upon the recent TensoRF [11] scene representation, which achieves fast, compact, and state-of-the-art quality on radiance field reconstruction for novel view synthesis. Our tensor factorization-based inverse rendering framework can simultaneously estimate scene geometry, materials, and illumination from multi-view images captured under unknown lighting conditions. Benefiting from the efficiency and extensibility of the TensoRF representation, our method can accurately model secondary shading effects (like shadows and indirect lighting) and generally support input images captured under a single or multiple unknown lighting conditions.

Similar to TensoRF, our approach models a scene as a neural voxel feature grid, factorized as multiple low-rank

* Equal contribution. † Equal advisory.

tensor components. We apply multiple small MLPs on the same feature grid and regress volume density, view-dependent color, normal, and material properties, to model the scene geometry and appearance. This allows us to simultaneously achieve both radiance field rendering – using density and view-dependent color, as done in NeRF [22] – and physically-based rendering – using density, normal and material properties, as done in inverse rendering methods [3, 20]. We supervise both renderings with the captured images to jointly reconstruct all scene components. In essence, we reconstruct a scene using both a radiance field and a physically-based model to reproduce the scene's appearance. While inverse rendering is our focus and primarily enabled by the physically-based model, modeling the radiance field is crucial for the success of the reconstruction (see Fig. 3), in significantly facilitating the volume density reconstruction and effectively regularizing the same tensor features shared by the physically-based model. Despite that previous works [41] similarly reconstruct NeRFs in inverse rendering, their radiance field is pre-computed and fixed in the subsequent inverse rendering stage; in contrast, our radiance field is jointly reconstructed and also benefits the physically-based rendering model estimation during optimization, leading to much higher quality. Besides, our radiance field rendering can also be directly used to provide accurate indirect illumination for the physically-based rendering, further benefiting the inverse rendering process.

Accounting for indirect illumination and shadowing is a critical challenge in inverse rendering. This is especially challenging for volume rendering, since it requires sampling a lot of secondary rays and computing the integrals along the rays by performing ray marching. Limited by the high-cost MLP evaluation, previous NeRF-based methods and SDF-based methods either simply ignore secondary effects [6, 7, 39], or avoid online computation by approximating these effects in extra distilled MLPs [41, 42], requiring expensive pre-computation and leading to degradation in accuracy. In contrast, owing to our efficient tensor-factorized representation, we are able to explicitly compute the ray integrals online for accurate visibility and indirect lighting with the radiance field rendering using low-cost second-bounce ray marching. Consequently, our approach enables higher accuracy in modeling these secondary effects, which is crucial in achieving high-quality scene reconstruction (see Tab. 2).

In addition, the flexibility and efficiency of our tensor-factorized representation allows us to perform inverse rendering from multiple unknown lighting conditions with limited GPU memory. Multi-light capture is known to be beneficial for inverse rendering tasks by providing useful photometric cues and reducing ambiguities in material estimation, thus being commonly used [13,15,18]. However, since each lighting condition corresponds to a separate radiance field, this can lead to extremely high computational costs if reconstructing multiple purely MLP-based NeRFs like previous works [28,41,42]. Instead, we propose to reconstruct radiance fields under multi-light in a joint manner as a factorized tensor. Extending from the original TensoRF representation that is a 4D tensor, we add an additional dimension corresponding to different lighting conditions, yielding a 5D tensor. Specifically, we add an additional vector factor (whose length equals the number of lights) per tensor component to explain the appearance variations under different lighting conditions, and we store this 5D tensor by a small number of bases whose outer-product reconstructs the 5D tensor. When multi-light capture is available, our framework can effectively utilize the additional photometric cues in the data, leading to better reconstruction quality than a single-light setting (see Tab. 1).

As shown in Fig. 1, our approach can reconstruct high-fidelity geometry and reflectance of a complex real scene captured under unknown natural illumination, enabling photo-realistic rendering under novel lighting conditions and additional applications like material editing. We evaluate our framework extensively on both synthetic and real data. Our approach outperforms previous inverse rendering methods [41,42] by a large margin qualitatively and quantitatively on challenging synthetic scenes, achieving state-of-the-art quality in scene reconstruction – for both geometry and material properties – and rendering – for both novel view synthesis and relighting. Owing to our efficient tensorial representation and joint reconstruction scheme, our approach also leads to a much faster reconstruction speed than previous neural field-based reconstruction methods while achieving superior quality. In summary,

- We propose a novel tensor factorization-based inverse rendering approach that jointly achieves physically-based rendering model estimation and radiance field reconstruction, leading to state-of-the-art scene reconstruction results;
- Our framework includes an efficient online visibility and indirect lighting computation technique, providing accurate second-bounce shading effects;
- We enable efficient multi-light reconstruction by modeling an additional lighting dimension in the factorized tensorial representation.

2. Related Works

Neural scene representations. Neural representations [1, 4, 21, 22, 30, 34, 35], as a promising alternative to traditional representations (like meshes, volumes, and point clouds), have been revolutionizing 3D content generation and modeling. Compared to traditional representations, such neural representations are more flexible and can more

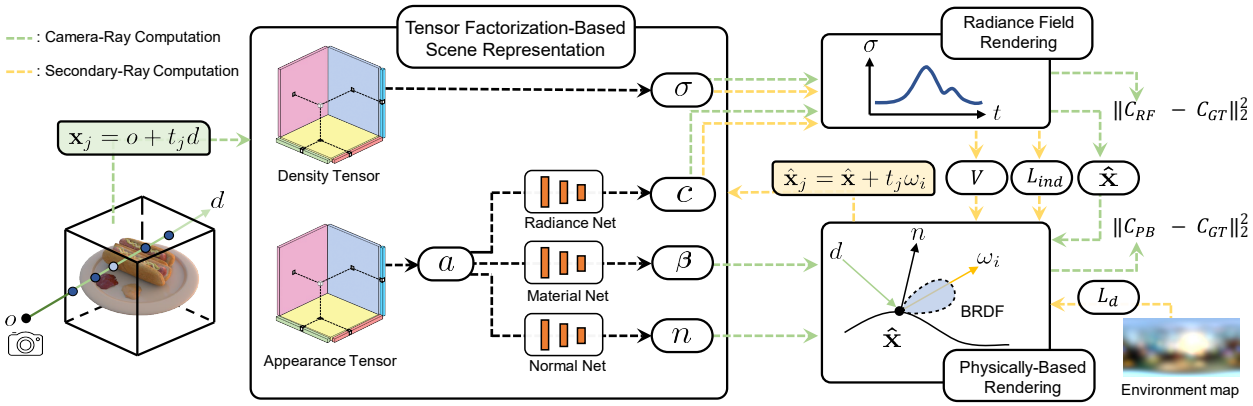

Figure 2. Overview. We propose a novel inverse rendering approach to reconstruct scene geometry, materials, and unknown natural illumination (as an environment map) from captured images. We reconstruct a scene as a novel representation (Sec. 3.2) that uses factorized tensors and multiple MLPs to regress volume density σ, view-dependent color c, normals $\mathbf{n}$, and material properties (i.e. BRDF parameters) $\boldsymbol{\beta}$, enabling both radiance field rendering and physically-based rendering (Sec. 3.1). In particular, we march a camera ray from camera origin $\mathbf{o}$ in viewing direction $\mathbf{d}$, sample points $\mathbf{x}_j$ along the ray, and apply radiance field rendering using the density and view-dependent colors regressed from our representation (Eqn. 1). We also use the volume rendering weights to determine the surface point $\hat{\mathbf{x}}$ on the ray (Eqn. 2), at which we perform physically based rendering using the normals and material properties (Eqn. 3). We compute accurate visibility V and indirect lighting L_{ind} using radiance field rendering by marching secondary rays from the surface point $\hat{\mathbf{x}}$ along sampled incoming light direction $\boldsymbol{\omega}_i$ (Sec. 3.3), enabling accurate physically-based rendering. We supervise both the radiance field rendering C_{RF} and physically-based rendering C_{PB} with the captured images in a per-scene optimization for joint scene reconstruction (Sec. 3.5).

accurately reproduce the geometry and appearance of real-world scenes. In particular, NeRF and many following neural field representations [11, 22, 23, 33, 36] have been proposed and applied to enable high-fidelity rendering results on novel view synthesis and many other applications [9, 10, 19, 25, 37]. Originally neural fields [22] are modelled in the form of MLPs, which have limited capacity and high computational costs. Recently, many works [11,23,29] introduce more efficient neural scene representations that combine neural feature maps/volumes with light-weight MLPs to reduce the computational cost and accelerate the training and rendering. In this work, we adapt the efficient tensor-factorized neural representation, TensoRF [11], to achieve accurate and efficient inverse rendering.

Inverse rendering. Abundant works [5,15–17,24,32] have been proposed to infer the geometry and material properties of real-world objects from image collections. They typically represent the scene geometry using triangle meshes that are known or can be pre-reconstructed with depth sensors or multi-view stereo techniques [27]. To reduce the ambiguities in the inverse rendering, they often require controlled lighting conditions [5, 24], or make use of learnt domain-specific priors [2,5,14,20]. In this work, we jointly estimate the geometry, materials and lighting from images captured under unknown lighting conditions with neural field representation that is more efficient and robust. While neural representations have recently been used for inverse rendering tasks, they [3,6,7,28,38,39,41,42] are limited by the usage of computation-intensive MLPs. This inefficiency adds extra burden on inverse rendering when computing secondary shading effects (like shadows), which requires to extensively sample secondary rays. Therefore, previous methods often ignore secondary effects [6, 7, 39], consider collocated flash lighting [3, 4, 38], or take extra costs to distill these effects into additional MLP networks [28, 41, 42]. In contrast, we base our model on the advanced TensoRF representation, utilizing factorized tensors, to achieve fast reconstruction. Our TensoRF-based approach supports fast density and radiance evaluation, enabling efficient online computation for secondary effects; this leads to more accurate shadow and indirect lighting modeling during reconstruction, further benefiting our reconstruction. Moreover, in contrast to previous methods [28,39,41,42] that can only handle captures under a single lighting condition, our model is easily extended to support multi-light capture by modeling an additional lighting dimension in the tensor factors.

3. Method

In this section, we present our tensor factorization-based inverse rendering framework, shown in Fig. 2, which reconstructs scene geometry, materials, and illumination from multi-view input images under unknown lighting. We leverage rendering with both neural radiance fields and physically-based light transport to model and reproduce scene appearance (Sec. 3.1). We introduce a novel TensoRF-based scene representation that allows for both rendering methods in scene reconstruction (Sec. 3.2). Our representation not only enables accurate and efficient computation for visibility and indirect lighting (Sec. 3.3) but also supports capturing under multiple unknown lighting

conditions (Sec. 3.4). We simultaneously estimate all scene components in a joint optimization framework with rendering losses and regularization terms (Sec. 3.5).

3.1. Rendering

We apply ray marching to achieve differentiable radiance field rendering as done in NeRF [22]. We further determine the expected surface intersection point for each ray using the volume density, and perform physically-based rendering at the surface point with the predicted scene properties.

Radiance field rendering. Given a camera ray $\mathbf{r}(t) = \mathbf{o} + t\mathbf{d}$ from ray origin $\mathbf{o}$ in direction $\mathbf{d}$, radiance field rendering samples N points on the ray and compute the pixel color as

$$C_{\text{RF}}(\mathbf{o}, \mathbf{d}) = \sum_{j=1}^{N} T_j(1 - \exp(-\sigma_j \delta_j))c_j$$
$$T_j = \exp(-\sum_{q=1}^{j-1} \sigma_q \delta_q) \tag{1}$$

where σ_j, δ_j, c_j and T_j are volume density, step size, view-dependent radiance color, and volume transmittance at each sampled point $\mathbf{r}(t_j)$.

Physically-based rendering. We apply a physically-based parametric BRDF model [8] f_r and perform physically-based rendering using predicted geometry and material properties at surface points on the camera rays. Similar to previous methods [22, 41], these surface points $\hat{\mathbf{x}}$ are naturally determined using the volume rendering weights and sampled points from Eqn. 1:

$$\hat{\mathbf{x}} = \sum_{j=1}^{N} w_j \mathbf{r}(t_j), \quad w_j = T_j(1 - \exp(-\sigma_j \delta_j)) \tag{2}$$

We leverage the classic surface rendering equation to compute a physically-based shading color with an integral over the upper hemisphere Ω at each surface point:

$$C_{\text{PB}}(\hat{\mathbf{x}}, \mathbf{d}) = \int_{\Omega} L_{\text{i}}(\hat{\mathbf{x}}, \boldsymbol{\omega}_i) f_r(\hat{\mathbf{x}}, \boldsymbol{\omega}_i, \mathbf{d}, \boldsymbol{\beta})(\boldsymbol{\omega}_i \cdot \mathbf{n}) \mathrm{d}\boldsymbol{\omega}_i \tag{3}$$

where $L_{\text{i}}(\hat{\mathbf{x}}, \boldsymbol{\omega}_i)$ is the incident illumination coming from direction $\boldsymbol{\omega}_i$. $\boldsymbol{\beta}$ and $\mathbf{n}$ represent the spatially-varying material parameters (albedo and roughness) of the BRDF and the surface normal at $\hat{\mathbf{x}}$.

In theory, accurately evaluating this integral in Eqn. 3 requires extensively sampling the lighting direction and computing the incident lighting $L_{\text{i}}(\hat{\mathbf{x}}, \boldsymbol{\omega}_i)$ to account for shadowing and indirect illumination. It requires additional ray marching along each lighting direction, which is known to be extremely computation-expensive. Previous NeRF-based methods often simplify it by only considering direct illumination [6, 7] or using extra auxiliary MLPs to make approximations and avoid full ray marching [28, 41]. Instead, we evaluate the integral more accurately by marching secondary rays online and computing the incident lighting $L_{\text{i}}(\hat{\mathbf{x}}, \boldsymbol{\omega}_i)$ with accurate shadowing and indirect illumination (see Sec. 3.3). This is made possible by our novel efficient TensoRF-based scene representation.

3.2. TensoRF-Based Representation

We now introduce our tensor factorization-based scene representation that simultaneously models volume density σ, view-dependent color c, shading normal $\mathbf{n}$, and material properties $\boldsymbol{\beta}$ (including diffuse albedo and specular roughness of the Disney BRDF [8]). Our method can model the scene under both single or multiple lighting conditions. We first introduce the single-light setting in this section and will discuss the multi-light extension in Sec. 3.4.

At a high-level, radiance, geometry, and material properties can all be represented as a 3D field. We can use a feature volume and voxel feature decoding functions to extract information at any point of the 3D field. To compress the space of the feature volume and also regularize the learning process, TensoRF [11] proposed to use a low-rank factorized tensor as the feature volume. In this work, we adopt the Vector-Matrix factorization proposed by TensoRF. In particular, we use two separate VM-factorized tensors, i.e. feature grids $\mathcal{G}_\sigma$ and $\mathcal{G}_a$, to model volume density and appearance, respectively. The appearance tensor $\mathcal{G}_a$ is followed by multiple light-weight MLP decoders to regress various appearance properties.

Density tensor. The density tensor $\mathcal{G}_\sigma$ is 3D and directly expresses volume density without any network. Specifically, the VM-factorized density tensor is expressed by the sum of vector-matrix outer products.

$$\mathcal{G}_\sigma = \sum_k \mathbf{v}^X_{\sigma,k} \circ \mathbf{M}^{YZ}_{\sigma,k} + \mathbf{v}^Y_{\sigma,k} \circ \mathbf{M}^{XZ}_{\sigma,k} + \mathbf{v}^Z_{\sigma,k} \circ \mathbf{M}^{XY}_{\sigma,k}$$
$$= \sum_k \sum_{m \in XYZ} \mathbf{v}^m_{\sigma,k} \circ \mathbf{M}^{\tilde{m}}_{\sigma,k} \tag{4}$$

Here, $\mathbf{v}^m_{\sigma,k}$ and $\mathbf{M}^{\tilde{m}}_{\sigma,k}$ represent the k^{th} vector and matrix factors of their corresponding spatial axes m; for simplicity, $\tilde{m}$ denotes the two axes orthogonal to m (e.g. $\tilde{X}$=YZ).

Appearance tensor. The appearance tensor $\mathcal{G}_a$ is 4D, modeled by similar vector-matrix spatial factors and additional feature basis vectors $\mathbf{b}$, expressing a multi-channel voxel feature grid:

$$\mathcal{G}_a = \sum_k \sum_{m \in XYZ} \mathbf{v}^m_{a,k} \circ \mathbf{M}^{\tilde{m}}_{a,k} \circ \mathbf{b}^m_k \tag{5}$$

Representing scene properties. We obtain density σ directly by linearly interpolating $\mathcal{G}_\sigma$, and apply multiple MLP

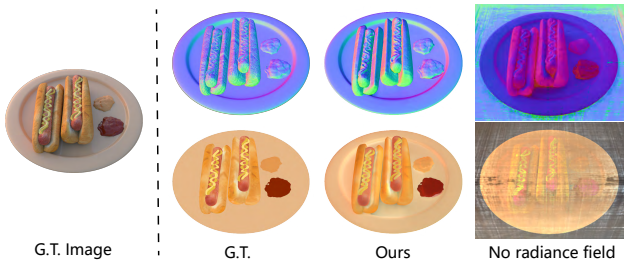

Figure 3. We compare normal and albedo reconstruction results between our joint reconstruction model and an ablated model without radiance field rendering during reconstruction. Radiance field reconstruction is crucial for us to achieve good reconstruction with a clean background and reasonable scene geometry.

networks to decode appearance properties from interpolated appearance features $\mathcal{G}_a$. This includes a radiance network $\mathcal{D}_c$, a shading normal network $\mathcal{D}_\mathbf{n}$, and a material network $\mathcal{D}_{\boldsymbol{\beta}}$. Overall, our scene representation is expressed by

$$\begin{aligned} \sigma_\mathbf{x} = \mathcal{G}_\sigma(\mathbf{x}), \quad a_\mathbf{x} = \mathcal{G}_a(\mathbf{x}) \\ c_\mathbf{x}, \mathbf{n}_\mathbf{x}, \boldsymbol{\beta}_\mathbf{x} = \mathcal{D}_c(a_\mathbf{x}), \mathcal{D}_\mathbf{n}(a_\mathbf{x}), \mathcal{D}_{\boldsymbol{\beta}}(a_\mathbf{x}) \end{aligned} \tag{6}$$

Here, $\mathbf{x}$ denotes an arbitrary 3D location, $\sigma_\mathbf{x} = \mathcal{G}_\sigma(\mathbf{x})$ and $a_\mathbf{x} = \mathcal{G}_a(\mathbf{x})$ are the linearly interpolated density and multi-channel appearance features, computed by linearly interpolating the spatial vector and matrix factors (please refer to the TensoRF paper [11] for the details of feature computation and interpolation).

In particular, volume density σ and shading normals $\mathbf{n}$ both express scene geometry, which describes global shapes and high-frequency geometric details, respectively. View-dependent color c and physical shading properties (normal $\mathbf{n}$ and material parameters $\boldsymbol{\beta}$) duplicatively model the scene appearance, determining the colors in the scene. We bind shading normal with volume density using a regularization term (see details in Sec. 3.5), correlating the scene geometry estimation and appearance reasoning.

In essence, our TensoRF-based scene representation provides scene geometry and appearance properties that are required for both radiance field rendering (Eqn. 1) and physically-based rendering (Eqn. 3) described in Sec. 3.1. *This represents a scene as a radiance field and a physically-based rendering model jointly.* We let the two sub-representations share the same neural features in our tensors, allowing their learning processes to benefit each other. While the physically-based model is our main focus and achieves inverse rendering, modeling the radiance field can facilitate the volume density reconstruction and also regularize the appearance features to be meaningful since it has a shorter gradient path. As a result, modeling the radiance field is crucial and necessary for us to achieve high-quality physically-based scene reconstruction as shown in Fig. 3. In addition, the radiance field can naturally provide indirect illumination for physically-based rendering, enabling more accurate physical model reconstruction.

Figure 4. We show our computed indirect illumination (b) of the full rendered image (a) and our lighting visibility (c) under two different directional lights.

3.3. Illumination and Visibility

Our TensoRF-based scene representation is highly efficient for optimization and evaluation. Especially, our volume density can be computed by simple tensor interpolation without using any MLPs, leading to highly efficient computation of volume transmittance and rendering weights (Eqn. 1,2). This allows us to compute accurate incident illumination L_i (as defined in Eqn. 3) that accounts for secondary shading effects with ray marching as shown in Fig. 4. In particular, the incident illumination is computed by

$$L_\text{i}(\hat{\mathbf{x}}, \boldsymbol{\omega}_i) = V(\hat{\mathbf{x}}, \boldsymbol{\omega}_i) L_\text{d}(\boldsymbol{\omega}_i) + L_\text{ind}(\hat{\mathbf{x}}, \boldsymbol{\omega}_i) \tag{7}$$

where V is the light visibility function, L_d is the direct illumination, and L_ind is the indirect illumination.

Direct illumination. We assume the unknown natural environment to be distant from the captured object and represent the global illumination as an environment map, parameterized by a mixture of spherical Gaussians (SG), which represents the direct illumination. Note that in contrast to previous methods [39,42] that use SG for computing the integral of the rendering equation with a closed-form approximation, we use SG only for its compact parameterized representation and compute the integral numerically by sampling secondary rays and performing ray marching, leading to more accurate lighting visibility and indirect lighting.

Visibility and indirect illumination. We make use of the jointly-trained radiance fields to model secondary shading effects. More specifically, the indirect illumination arriving at $\hat{\mathbf{x}}$ from $\boldsymbol{\omega}_i$ is inherently explained by the radiance color C_RF along the ray $\mathbf{r}_i(t) = \hat{\mathbf{x}} + t\boldsymbol{\omega}_i$. The visibility function is exactly modeled by the transmittance function in volume rendering. Therefore, the light visibility and indirect illumination term in Eqn. 7 can be calculated as:

$$V(\hat{\mathbf{x}}, \boldsymbol{\omega}_i) = T(\hat{\mathbf{x}}, \boldsymbol{\omega}_i), \quad L_\text{ind}(\hat{\mathbf{x}}, \boldsymbol{\omega}_i) = C_\text{RF}(\hat{\mathbf{x}}, \boldsymbol{\omega}_i) \tag{8}$$

Here $T(\hat{\mathbf{x}}, \boldsymbol{\omega}_i)$ represents the volume transmittance of the final point sampled on the ray $\mathbf{r}_i(t) = \hat{\mathbf{x}} + t\boldsymbol{\omega}_i$.

Second-bounce ray marching. To make our physically-based rendering C_PB accurate, we compute the rendering

integral (Eqn. 3) via Monte Carlo integration by marching multiple rays from each surface point $\hat{\mathbf{x}}$ with stratified sampling when training. For each ray, we obtain direct illumination from the SGs and compute visibility $T(\hat{\mathbf{x}}, \boldsymbol{\omega}_i)$ and indirect illumination $C_{\text{RF}}(\hat{\mathbf{x}}, \boldsymbol{\omega}_i)$ using Eqn. 1 directly. This second-bounce ray marching is known to be expensive for previous NeRF-based methods, which either requires extremely high computational resources (128 TPUs) [28] or utilizes long (several days of) offline pre-computation [41]; both require training extra MLPs, which takes excessive computational costs and is unable to achieve high accuracy. We instead perform second-bounce ray marching online, achieving higher accuracy (see Fig. 6), which is affordable, thanks to our highly efficient tensor factorization-based representation.

3.4. Multi-Light Representation

We have discussed our scene representation under a single unknown lighting condition in Sec. 3.2. Our method can be easily extended to support capturing under multiple unknown lighting conditions, owing to the generality and extensibility of tensor factorization. We achieve this by adding an additional lighting dimension with vector factors $\mathbf{e}$ – where the length of each vector $\mathbf{e}$ equals the number of lighting conditions – into the factorized appearance tensor $\mathcal{G}_a$, leading to a 5D tensor that expresses scene appearance under different lighting conditions. We denote the 5D multi-light appearance tensor as $\mathcal{G}_a^{\text{5D}}$, represented by the factorization:

$$\mathcal{G}_a^{\text{5D}} = \sum_k \sum_{m \in XYZ} \mathbf{v}_{a,k}^m \circ \mathbf{M}_{a,k}^{\tilde{m}} \circ \mathbf{e}_k^m \circ \mathbf{b}_k^m \tag{9}$$

Note that, it is only the view-dependent colors that require being modeled separately under different lighting, while the physical scene properties – including volume density, normals, and material properties – are inherently shared across multiple lighting conditions. Therefore, we decode view-dependent color using the neural features per light and decode other shading properties using the mean features along the lighting dimension:

$$\begin{aligned} \bar{a}_{\mathbf{x}} &= \frac{\sum_l a_{\mathbf{x},l}}{P}, \quad a_{\mathbf{x},l} = \mathcal{G}_a^{\text{5D}}(\mathbf{x}, l) \\ c_{\mathbf{x}}, \mathbf{n}_{\mathbf{x}}, \boldsymbol{\beta}_{\mathbf{x}} &= \mathcal{D}_c(a_{\mathbf{x},l}), \mathcal{D}_{\mathbf{n}}(\bar{a}_{\mathbf{x}}), \mathcal{D}_{\boldsymbol{\beta}}(\bar{a}_{\mathbf{x}}) \end{aligned} \tag{10}$$

where l is the light index, P is the number of lighting conditions, and $\bar{a}_{\mathbf{x}}$ is the mean feature.

By simply adding additional vectors in the factorized appearance tensor, our method allows us to efficiently reconstruct and query the radiance under different illumination conditions, thereby still allowing for indirect illumination computation in the multi-light setting with the method discussed in Sec. 3.3. The multi-light input can provide useful photometric cues and reduce ambiguity in material prediction, and therefore leads to more accurate reconstruction of geometry and material estimation (see Tab. 1).

3.5. Joint Reconstruction and Training Losses

For both single- and multi- light settings, we jointly reconstruct the scene geometry and appearance properties that are modeled by our tensor factorization-based representation, through an end-to-end per-scene optimization.

Rendering losses. We supervise the rendered colors from both the radiance field rendering C_{RF} and the physically-based rendering C_{PB} with the ground-truth colors C_{gt} from the captured images and include two rendering loss terms:

$$\ell_{\text{RF}} = \|C_{\text{RF}} - C_{\text{gt}}\|_2^2, \quad \ell_{\text{PB}} = \|C_{\text{PB}} - C_{\text{gt}}\|_2^2 \tag{11}$$

Normal regularization. Our shading normals $\mathbf{n}$ are regressed from an MLP to express high-frequency surface variations and used in physically-based rendering. However, since the entire system is highly ill-conditioned, only supervising the network output with rendering loss is prone to overfitting issues and produces incorrect normal predictions. On the other hand, many previous NeRF-based method [6, 26, 28, 41] use the negative direction of volume density gradients $\mathbf{n}_\sigma = -\frac{\nabla_{\mathbf{x}}\sigma}{\|\nabla_{\mathbf{x}}\sigma\|}$ as normals for shading computation. Such derived normals are better aligned with the surface but can be noisy and lack fine details. Inspired by the rent work Ref-NeRF [31], we correlate our shading normal with the density-derived normal using a loss

$$\ell_{\mathbf{n}} = \sum_j w_j \|\mathbf{n}_j - \mathbf{n}_{\sigma,j}\|_2^2 \tag{12}$$

This regularization term back-propagates gradients to both the appearance tensor (through the normal network) and the density tensor, further correlating our entire scene geometry and appearance reasoning, leading to better normal reconstruction. Note that while Ref-NeRF uses the same normal regularization, their normals are used in MLP-based shading computation; in contrast, our normals are used in a physically-based rendering process, allowing for more meaningful geometry reasoning and leading to more accurate normal reconstruction.

Final loss. Following TensoRF, we apply additional ℓ_1-regularization on all tensor factors, denoted as $\ell_{\mathcal{G}}$. We also apply a smoothness regularization term on BRDF parameters $\boldsymbol{\beta}$ to enhance their spatial consistency. A loss term penalizing back-facing normals is used in addition. Please refer to the supplementary materials for the details of these regularization terms. We apply all these losses to supervise and regularize our whole system to jointly reconstruct the scene, optimizing our scene representation (with all tensor factors and MLPs), as well as the SG parameters of the en-

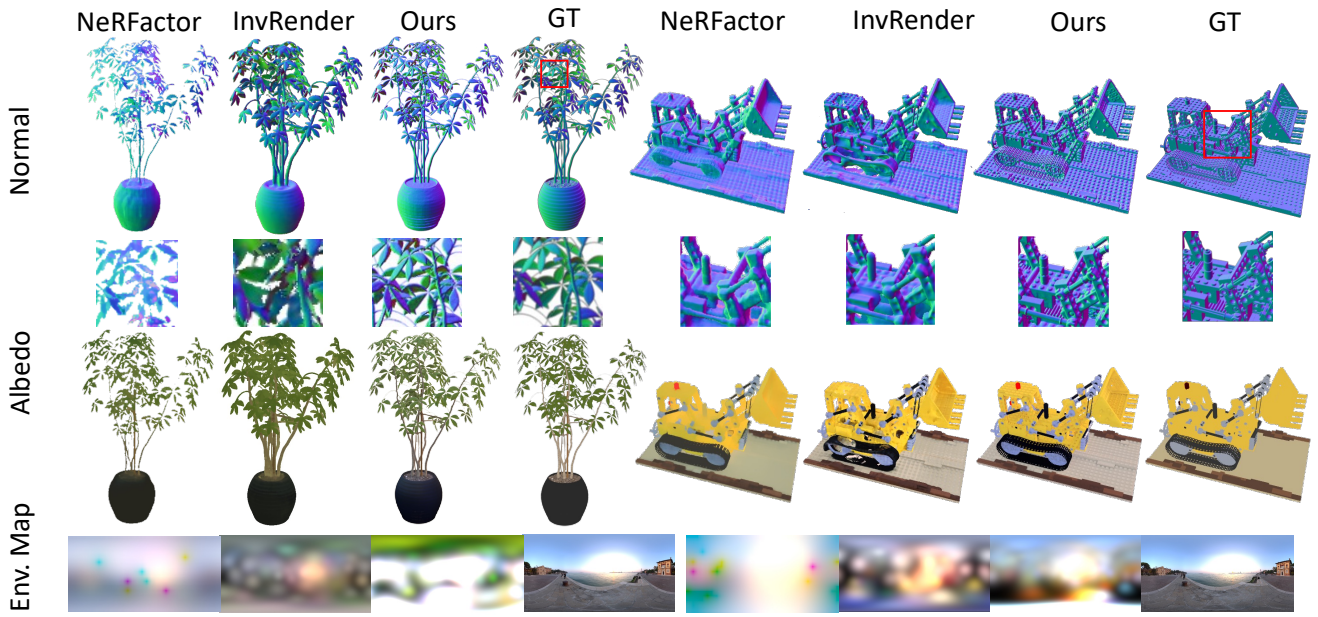

Figure 5. Visual comparison against baseline methods. Our method produces inverse rendering results of higher quality with more detailed normals and more accurate albedo, thus leading to more photo-realistic relighting results.

Method	Normal	Albedo			Novel View Synthesis			Relighting			Average Runtime
	MAE ↓	PSNR ↑	SSIM ↑	LPIPS ↓	PSNR ↑	SSIM ↑	LPIPS ↓	PSNR ↑	SSIM ↑	LPIPS ↓	
NeRFactor	6.314	25.125	0.940	0.109	24.679	0.922	0.120	23.383	0.908	0.131	$>$ 100 hrs
InvRender	5.074	27.341	0.933	0.100	27.367	0.934	0.089	23.973	0.901	0.101	15 hrs
Ours in 25 minutes	4.876	28.210	0.947	0.091	32.350	0.964	0.061	27.431	0.935	0.094	25 mins
Ours	4.100	29.275	0.950	0.085	35.088	0.976	0.040	28.580	0.944	0.081	5 hrs
Ours w/ rotated multi-light	3.602	29.672	0.956	0.079	34.395	0.974	0.043	28.955	0.949	0.077	5 hrs
Ours w/ general multi-light	3.551	29.326	0.951	0.084	34.223	0.973	0.045	29.008	0.947	0.078	5 hrs

Table 1. Quantitative comparisons on the synthetic dataset. Our (single-light) results have significantly outperformed the baseline methods by producing more accurate normal and albedo, thus achieving more realistic novel view synthesis and relighting results. Our method can further take images captured under different lighting conditions, and boost the performance in inverse rendering. (We scale each RGB channel of all albedo results by a global scalar, as done in NeRFactor [41]. For a fair comparison, all novel view synthesis results are generated with physically-based rendering, though our radiance field rendering has better quality.)

vironment map, with a final loss

$$\ell = \alpha_{\text{RF}}\ell_{\text{RF}} + \alpha_{\text{PB}}\ell_{\text{PB}} + \alpha_{\boldsymbol{\beta}}\ell_{\boldsymbol{\beta}} + \alpha_{\mathbf{n}}\ell_{\mathbf{n}} + \alpha_{\mathbf{d}}\ell_{\mathbf{d}} + \alpha_{\mathcal{G}}\ell_{\mathcal{G}} \quad (13)$$

4. Experiments

We now evaluate our model on various challenging scenes. We make comparisons against previous state-of-the-art methods and also present an ablation study to verify the effectiveness of our design choices.

Datasets. We perform experiments on four complex synthetic scenes, including three blender scenes from [22] and one from the Stanford 3D scanning repository [12]. We re-render these scenes to obtain their ground-truth images, as well as BRDF parameters and normal maps. We also render two types of multi-light data: rotated multi-light and general multi-light. We also perform experiments on the original NeRF-synthetic dataset and 4 captured real data. Please refer to the supplementary for experimental results of those extra data, more details about our datasets, and more explanation about our multi-light setting.

Comparisons with previous methods. We compare our model with previous state-of-the-art neural field-based inverse rendering methods, NeRFactor [41] and InvRender [42], on these scenes using images captured under a single unknown lighting condition. We also compare with our model trained under multi-light settings. Table 1 shows the accuracy of the estimated albedo, normal, and relighting results using metrics including PSNR, SSIM and LPIPS [40] averaged over the four scenes. We can see that our single-light method (using the same input as baselines) outperforms both previous methods significantly on all metrics, demonstrating the superiority of our method. We also include a visual comparison in Fig. 5, showing that our method predicts more accurate albedo and normal that are closer to the ground truth compared to baseline methods, thus generating more realistic relighting results. In particular, our results produce normals that are of much higher quality and more faithfully reflect the geometry variations, while the baseline methods produce over-smooth results that lack shape details. Although NeRFactor's albedo result on the lego scene looks closer to the ground truth than our results. We claim that this is because that NeRFactor uses a high-weight BRDF smoothness loss, which dam-

Figure 6. We compare our relighting results with previous methods. Note that our approach recovers more accurate shadows thanks to our second-bounce ray marching.

ages its reconstruction quality of other components. In the supplementary, we showcase that we can achieve a similar albedo result by increasing the BRDF smoothness weight. In particular, NeRFactor approximates the visibility function in a distilled MLP from a pre-trained NeRF and completely ignores the indirect illumination. While InvRender considers both visibility and indirect illumination in their pre-computation, its spherical Gaussian-based shading computation can only achieve limited accuracy. Both methods do inverse rendering in a second stage after pre-training NeRFs. In contrast, our tensor-factorized representation achieves a single-stage joint reconstruction and efficiently performs explicit second-bounce ray marching for more accurate shadowing (see Fig. 6) and indirect lighting, thus leading to significantly higher reconstruction quality.

Meanwhile, owing to our more efficient representation, our high reconstruction quality is achieved with the fastest reconstruction speed, as reflected by the reconstruction time. NeRF-based method NeRFactor takes days to compute because of visibility pre-computation. InvRender is faster than NeRFactor, due to its SDF-based rendering [36] and SG-based closed-form rendering integral computation; however, its SDF-based reconstruction fails on challenging scenes like the Lego shown in Fig. 5 and the SG-based integration leads to inaccurate secondary shading effects as mentioned. On the other hand, our approach leverages ray marching-based radiance rendering in the reconstruction, robustly producing high-quality results on all testing scenes with accurate secondary effects, while still being faster than InvRender. In fact, while our method takes 5 hours to finish its full training for its best performance, it can achieve good quality in a much shorter training period (25 minutes). As shown in Tab. 1 and Fig. 7, our approach can achieve high-quality geometry reconstruction in only 25 minutes and outperform previous methods that trained for tens of or even hundreds of hours.

Multi-light results. In addition, our framework can also effectively leverage the additional input in a multi-light capture and further boost the accuracy of the inverse rendering performance without adding additional computation costs, as shown in Tab. 1, while the MLP-based baseline methods cannot be trivially extended to support such setups in an efficient manner. We also found that multi-light settings can greatly improve geometry reconstruction and help solving

Method	Normal MAE ↓	Albedo PSNR ↑	NVS PSNR ↑
w/o visibility	4.716	27.265	34.703
w/o indirect illum.	4.186	29.003	34.910
w/ indirect illum. + visibility	**4.100**	**29.275**	**35.088**

Table 2. Our full method models more physically-accurate light transport by accounting for lighting visibility and indirect illuminations, thus achieving much better accuracy in inverse rendering.

Figure 7. We compare geometry reconstruction results of our model taking only 25 minutes of optimization, with previous methods, taking 15 and $>$ 100 hours. Our approach even recovers more high-frequency details with substantially less reconstruction time.

the color ambiguity between lighting and materials. Please refer the supplementary for more results and analysis.

Indirect illumination and visibility. Our tensor-factorized representation allows us to explicit sample secondary rays in an efficient way to account for computing lighting visibility and indirect illumination. As shown in Tab. 2, without including these terms, the model cannot accurately represent the secondary shading effects and tend to bake them in the albedo or normal, resulting in lower quality. Nonetheless, note that these ablated models of our method in fact already achieves superior quality compared to previous methods (that pre-computes NeRFs and do inverse rendering disjointly) shown in Tab. 1, demonstrating the effectiveness of our joint reconstruction framework. Our full method further achieves more accurate reconstruction because of more accurate light transport modeling.

5. Conclusion

We present a novel inverse rendering approach that achieves efficient and high-quality scene reconstruction from multi-view images under unknown illumination. Our approach models a scene as both a radiance field and a physically-based model with density, normals, lighting, and material properties. By jointly reconstructing both models, we achieve high-quality geometry and reflectance reconstruction, enabling photo-realistic rendering and relighting. Owing to the efficiency and generality of the tensor factorized representation, our framework allows for accurate computation for shadowing and indirect lighting effects, and also flexibly supports capturing under an arbitrary number of lighting conditions. We demonstrate that our approach is able to achieve state-of-the-art inverse rendering results, outperforming previous neural methods in terms of both reconstruction quality and efficiency.

References

[1] Kara-Ali Aliev, Artem Sevastopolsky, Maria Kolos, Dmitry Ulyanov, and Victor Lempitsky. Neural point-based graphics. In *European Conference on Computer Vision*, pages 696–712. Springer, 2020. 2

[2] Jonathan T Barron and Jitendra Malik. Shape, illumination, and reflectance from shading. *IEEE transactions on pattern analysis and machine intelligence*, 37(8):1670–1687, 2015. 3

[3] Sai Bi, Zexiang Xu, Pratul Srinivasan, Ben Mildenhall, Kalyan Sunkavalli, Miloš Hašan, Yannick Hold-Geoffroy, David Kriegman, and Ravi Ramamoorthi. Neural reflectance fields for appearance acquisition. *arXiv preprint arXiv:2008.03824*, 2020. 2, 3

[4] Sai Bi, Zexiang Xu, Kalyan Sunkavalli, Miloš Hašan, Yannick Hold-Geoffroy, David Kriegman, and Ravi Ramamoorthi. Deep reflectance volumes: Relightable reconstructions from multi-view photometric images. In *European Conference on Computer Vision*, pages 294–311. Springer, 2020. 2, 3

[5] Sai Bi, Zexiang Xu, Kalyan Sunkavalli, David Kriegman, and Ravi Ramamoorthi. Deep 3d capture: Geometry and reflectance from sparse multi-view images. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5960–5969, 2020. 3

[6] Mark Boss, Raphael Braun, Varun Jampani, Jonathan T Barron, Ce Liu, and Hendrik Lensch. Nerd: Neural reflectance decomposition from image collections. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 12684–12694, 2021. 1, 2, 3, 4, 6

[7] Mark Boss, Varun Jampani, Raphael Braun, Ce Liu, Jonathan Barron, and Hendrik Lensch. Neural-pil: Neural pre-integrated lighting for reflectance decomposition. *Advances in Neural Information Processing Systems*, 34:10691–10704, 2021. 1, 2, 3, 4

[8] Brent Burley and Walt Disney Animation Studios. Physically-based shading at disney. In *ACM SIGGRAPH*, volume 2012, pages 1–7. vol. 2012, 2012. 4

[9] Eric R. Chan, Connor Z. Lin, Matthew A. Chan, Koki Nagano, Boxiao Pan, Shalini De Mello, Orazio Gallo, Leonidas Guibas, Jonathan Tremblay, Sameh Khamis, Tero Karras, and Gordon Wetzstein. Efficient geometry-aware 3D generative adversarial networks. In *arXiv*, 2021. 3

[10] Eric R Chan, Marco Monteiro, Petr Kellnhofer, Jiajun Wu, and Gordon Wetzstein. pi-gan: Periodic implicit generative adversarial networks for 3d-aware image synthesis. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 5799–5809, 2021. 3

[11] Anpei Chen, Zexiang Xu, Andreas Geiger, Jingyi Yu, and Hao Su. Tensorf: Tensorial radiance fields. *arXiv preprint arXiv:2203.09517*, 2022. 1, 3, 4, 5

[12] Brian Curless and Marc Levoy. A volumetric method for building complex models from range images. In *Proceedings of the 23rd annual conference on Computer graphics and interactive techniques*, pages 303–312, 1996. 7

[13] Paul Debevec, Tim Hawkins, Chris Tchou, Haarm-Pieter Duiker, Westley Sarokin, and Mark Sagar. Acquiring the reflectance field of a human face. In *Proceedings of the 27th annual conference on Computer graphics and interactive techniques*, pages 145–156, 2000. 2

[14] Yue Dong, Guojun Chen, Pieter Peers, Jiawan Zhang, and Xin Tong. Appearance-from-motion: Recovering spatially varying surface reflectance under unknown lighting. *ACM Transactions on Graphics*, 33(6):193, 2014. 3

[15] Dan B Goldman, Brian Curless, Aaron Hertzmann, and Steven M Seitz. Shape and spatially-varying brdfs from photometric stereo. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 32(6):1060–1071, 2009. 2, 3

[16] Carlos Hernandez, George Vogiatzis, and Roberto Cipolla. Multiview photometric stereo. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 30(3):548–554, 2008. 3

[17] Jason Lawrence, Szymon Rusinkiewicz, and Ravi Ramamoorthi. Efficient brdf importance sampling using a factored representation. *ACM Transactions on Graphics (ToG)*, 23(3):496–505, 2004. 3

[18] Hendrik PA Lensch, Jochen Lang, Asla M Sá, and Hans-Peter Seidel. Planned sampling of spatially varying brdfs. In *Computer graphics forum*, volume 22, pages 473–482. Wiley Online Library, 2003. 2

[19] Zhengqi Li, Simon Niklaus, Noah Snavely, and Oliver Wang. Neural scene flow fields for space-time view synthesis of dynamic scenes. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6498–6508, 2021. 3

[20] Zhengqin Li, Zexiang Xu, Ravi Ramamoorthi, Kalyan Sunkavalli, and Manmohan Chandraker. Learning to reconstruct shape and spatially-varying reflectance from a single image. In *SIGGRAPH Asia 2018*, page 269. ACM, 2018. 2, 3

[21] Stephen Lombardi, Tomas Simon, Jason Saragih, Gabriel Schwartz, Andreas Lehrmann, and Yaser Sheikh. Neural volumes: Learning dynamic renderable volumes from images. *arXiv preprint arXiv:1906.07751*, 2019. 2

[22] Ben Mildenhall, Pratul P Srinivasan, Matthew Tancik, Jonathan T Barron, Ravi Ramamoorthi, and Ren Ng. Nerf: Representing scenes as neural radiance fields for view synthesis. *Communications of the ACM*, 65(1):99–106, 2021. 1, 2, 3, 4, 7

[23] Thomas Müller, Alex Evans, Christoph Schied, and Alexander Keller. Instant neural graphics primitives with a multiresolution hash encoding. *arXiv preprint arXiv:2201.05989*, 2022. 3

[24] Giljoo Nam, Joo Ho Lee, Diego Gutierrez, and Min H Kim. Practical SVBRDF acquisition of 3D objects with unstructured flash photography. In *SIGGRAPH Asia 2018*, page 267. ACM, 2018. 3

[25] Keunhong Park, Utkarsh Sinha, Jonathan T Barron, Sofien Bouaziz, Dan B Goldman, Steven M Seitz, and Ricardo Martin-Brualla. Nerfies: Deformable neural radiance fields. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 5865–5874, 2021. 3

[26] Viktor Rudnev, Mohamed Elgharib, William Smith, Lingjie Liu, Vladislav Golyanik, and Christian Theobalt. Nerf for outdoor scene relighting. In *European Conference on Computer Vision (ECCV)*, 2022. 6

[27] Johannes Lutz Schönberger, Enliang Zheng, Marc Pollefeys, and Jan-Michael Frahm. Pixelwise view selection for unstructured multi-view stereo. In *European Conference on Computer Vision (ECCV)*, 2016. 3

[28] Pratul P Srinivasan, Boyang Deng, Xiuming Zhang, Matthew Tancik, Ben Mildenhall, and Jonathan T Barron. Nerv: Neural reflectance and visibility fields for relighting and view synthesis. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7495–7504, 2021. 1, 2, 3, 4, 6

[29] Cheng Sun, Min Sun, and Hwann-Tzong Chen. Direct voxel grid optimization: Super-fast convergence for radiance fields reconstruction. In *CVPR*, 2022. 3

[30] Justus Thies, Michael Zollhöfer, and Matthias Nießner. Deferred neural rendering: Image synthesis using neural textures. *ACM Transactions on Graphics (TOG)*, 38(4):1–12, 2019. 2

[31] Dor Verbin, Peter Hedman, Ben Mildenhall, Todd Zickler, Jonathan T. Barron, and Pratul P. Srinivasan. Ref-NeRF: Structured view-dependent appearance for neural radiance fields. *CVPR*, 2022. 6

[32] Rui Xia, Yue Dong, Pieter Peers, and Xin Tong. Recovering shape and spatially-varying surface reflectance under unknown illumination. *ACM Transactions on Graphics*, 35(6):187, 2016. 3

[33] Qiangeng Xu, Zexiang Xu, Julien Philip, Sai Bi, Zhixin Shu, Kalyan Sunkavalli, and Ulrich Neumann. Point-nerf: Point-based neural radiance fields. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5438–5448, 2022. 3

[34] Zexiang Xu, Sai Bi, Kalyan Sunkavalli, Sunil Hadap, Hao Su, and Ravi Ramamoorthi. Deep view synthesis from sparse photometric images. *ACM Transactions on Graphics*, 38(4):76, 2019. 2

[35] Zexiang Xu, Kalyan Sunkavalli, Sunil Hadap, and Ravi Ramamoorthi. Deep image-based relighting from optimal sparse samples. *ACM Transactions on Graphics*, 37(4):126, 2018. 2

[36] Lior Yariv, Yoni Kasten, Dror Moran, Meirav Galun, Matan Atzmon, Basri Ronen, and Yaron Lipman. Multiview neural surface reconstruction by disentangling geometry and appearance. *Advances in Neural Information Processing Systems*, 33:2492–2502, 2020. 3, 8

[37] Kai Zhang, Nick Kolkin, Sai Bi, Fujun Luan, Zexiang Xu, Eli Shechtman, and Noah Snavely. Arf: Artistic radiance fields. In *European Conference on Computer Vision*, pages 717–733. Springer, 2022. 3

[38] Kai Zhang, Fujun Luan, Zhengqi Li, and Noah Snavely. Iron: Inverse rendering by optimizing neural sdfs and materials from photometric images. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5565–5574, 2022. 3

[39] Kai Zhang, Fujun Luan, Qianqian Wang, Kavita Bala, and Noah Snavely. 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*, pages 5453–5462, 2021. 2, 3, 5

[40] Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 586–595, 2018. 7

[41] Xiuming Zhang, Pratul P Srinivasan, Boyang Deng, Paul Debevec, William T Freeman, and Jonathan T Barron. Nerfactor: Neural factorization of shape and reflectance under an unknown illumination. *ACM Transactions on Graphics (TOG)*, 40(6):1–18, 2021. 1, 2, 3, 4, 6, 7

[42] Yuanqing Zhang, Jiaming Sun, Xingyi He, Huan Fu, Rongfei Jia, and Xiaowei Zhou. Modeling indirect illumination for inverse rendering. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 18643–18652, 2022. 2, 3, 5, 7