Don't Mask Out the Background! Natural-Light Photometric Stereo via Illumination Reconstruction

Taiga Hashida[1], Hiroaki Santo[2], and Fumio Okura[1]

[1] Graduate School of Information Science and Technology, The University of Osaka, Japan
[2] D3 Center, The University of Osaka, Japan
{hashida.taiga,santo.hiroaki,okura}@ist.osaka-u.ac.jp

Abstract. This paper introduces an inverse-rendering framework for natural-light, uncalibrated photometric stereo (PS) that leverages background cues captured alongside the target object. Natural-light PS (NaPS) acquires shading variations by moving or rotating the camera and object under fixed, uncontrolled illumination, such as indoor lighting, while maintaining their relative geometry. However, uncalibrated NaPS, in which the lighting conditions are unknown, remains inherently ill-posed. To tackle this challenge, we propose explicitly reconstructing the lighting environment from the image background, which is typically masked out in prior work, thereby converting uncalibrated NaPS into a tractable inverse-rendering problem. Specifically, we move and rotate the camera-object pair while keeping their relative pose fixed, and reconstruct the surrounding illumination directly from the observed background using a 3D Gaussian Splatting (3DGS) representation. We then optimize the target object's shape and reflectance via inverse rendering under the reconstructed illumination. Experimental results demonstrate that our inverse-rendering-based approach yields more accurate estimates of both geometry and reflectance than existing learning-based PS methods, especially under realistic near-field indoor conditions.

Keywords: Photometric stereo · Natural illumination · Material estimation · Illumination reconstruction

1 Introduction

Photometric stereo (PS) aims to recover the detailed shape and reflectance of an object by observing shading variations across multiple images captured under varying lighting conditions. Classical PS methods [37, 39] assume calibrated, directional light sources in controlled laboratory environments, which limits their practicality in everyday capture scenarios. To expand PS beyond such constraints, natural-light photometric stereo (NaPS) [26] has recently emerged as an attractive alternative that operates under fixed, uncontrolled illumination, such as

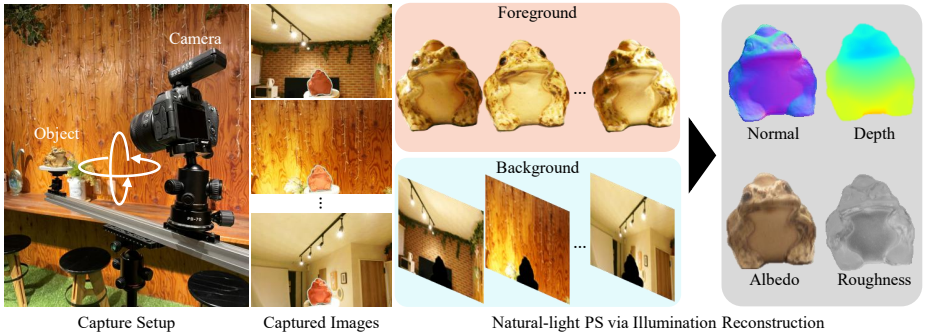


Fig. 1: The camera simultaneously captures the target object and the background. Unlike existing photometric stereo (PS) methods that mask out background regions, we explicitly use it to *reconstruct* the surrounding illumination for the natural-light PS (NaPS) setup, enabling inverse rendering for high-fidelity reconstruction of target shape and reflectance.

typical indoor lighting. NaPS acquires multiple images by moving or rotating the camera-object pair while maintaining their relative geometry, realizing shading variation without modifying the lighting setup.

Despite its practical potential, uncalibrated NaPS, in which the lighting conditions are unknown, is inherently ill-posed. Unlike traditional PS, where lighting directions and intensities are explicitly measured or estimated, NaPS relies entirely on environmental illumination, which can be spatially complex. Furthermore, near-field illumination, which is common in indoor scenes due to the proximity of light sources such as lamps or monitors, violates the far-field assumption and introduces spatially varying lighting effects. These effects cause severe ambiguities among illumination, reflectance, and geometry, leading to inaccurate shape and reflectance recovery.

In this work, we propose a method that leverages the NaPS capture setup, *i.e.*, the static background environment. Rather than treating the background as a nuisance to be masked out, as is common in existing PS pipelines, we exploit it as a direct observation of the surrounding lighting environment. Our key idea is that when capturing an object under natural illumination, the background provides cues about the incoming light distribution that also influences the object. By explicitly *reconstructing* the illumination from background observations, we can transform the ill-posed NaPS problem into a tractable inverse-rendering formulation.

Our method reconstructs the illumination from the observed background using a 3D Gaussian Splatting (3DGS) representation [21,30], which provides a compact, continuous, and spatially aware representation of the surrounding illumination. With the reconstructed illumination, we then perform inverse rendering to jointly optimize the object's shape and reflectance. This framework enables accurate uncalibrated NaPS under near-field illumination without requiring calibrated

light sources, using only captured images that include both the object and its background, as shown in Fig. 1.

We validate the proposed approach through experiments using both synthetic and real indoor scenes. Our results demonstrate that our method, based on inverse rendering with reconstructed background illumination, significantly improves both shape and reflectance estimation compared to the state-of-the-art PS methods [15, 23]. The findings suggest that leveraging the background as a source of illumination cues opens a new direction for practical, physics-based PS in uncontrolled lighting environments.

Contributions. Our contributions are twofold. First, we introduce the first PS method that leverages background information, which is particularly useful for the static lighting environment in the NaPS setup. This reformulates the challenging NaPS problem as an inverse-rendering optimization problem. Second, we propose an illumination reconstruction and inverse rendering method using 3DGS, which provides a lightweight and spatially aware representation of natural illumination.

2 Related Work

While early PS methods typically employ analytical approaches [37,39], recent PS methods often rely on learning-based models or inverse-rendering optimization to estimate shape and reflectance under various illumination conditions.

NaPS provides a practical acquisition setting in which images are captured under fixed but uncontrolled illumination. It has therefore been treated primarily as an imaging condition rather than as a technical formulation of PS. Namely, when background observations are masked out, NaPS can be viewed as one of the target scenarios to which both learning-based and inverse-rendering-based methods can be applied. We therefore review prior work from these two technical perspectives, and then discuss light representations used in inverse-rendering methods, which are closely related to one of our technical contributions.

Learning-based PS. Learning-based approaches estimate object parameters using a data prior obtained from numerous training datasets, typically through deep neural networks. Early learning-based PS methods mainly deal with calibrated scenarios, where the lighting directions and intensities are known [5, 6, 13, 18, 33, 34]. Recent methods tackle uncalibrated or self-calibrated PS via light source estimation, as in SDPS-Net [4] and its extension [7]. While these methods assume a directional light source, Universal Photometric Stereo (UniPS) [10, 14, 15, 23] estimates surface normals under arbitrary and unknown lighting environments without any specific assumptions. UniPS uses a generic lighting representation called global lighting contexts, whereas our approach explicitly reconstructs the lighting environment from background observations.

Inverse-rendering-based PS. In contrast to learning-based approaches, inverse-rendering-based methods do not rely on large-scale training datasets; instead,

they jointly optimize the lighting, surface normals, and reflectance through differentiable rendering. To alleviate the inherent ambiguity of inverse rendering, Kaya *et al.* [20] introduced a pre-trained light estimation network; however, their method does not update the lighting parameters during optimization. To address this limitation, SCPS-NIR [24] first estimates the lighting via a dedicated network and then jointly refines the light source, surface normals, and reflectance by fine-tuning the network during training. To handle general material properties and leverage shadow cues, DANI-Net [27] further incorporates an anisotropic reflectance model and differentiable shadow reasoning. These inverse-rendering-based methods, however, mainly assume a single directional light per image and therefore do not explicitly address natural illumination.

As a method specialized for the NaPS capture setup, Spin-UP [26] estimates the surface normals and reflectance under natural illumination. It assumes a fixed relative pose between the camera and object and captures images by rotating them around a specific axis using a turntable. To mitigate light source ambiguity, this approach extracts and uses a prior from the object boundary during initialization. In contrast, our method allows more flexible rigid motion of the camera-object pair and mitigates light source ambiguity by directly reconstructing the illumination from the background in captured images.

Light representation in inverse rendering. Most inverse-rendering methods represent the lighting environment using spherical Gaussians [26, 40, 43], spherical harmonics [22, 42], and environment maps in the latitude-longitude format [8, 44] because these representations are simple and computationally convenient. However, these representations are incapable of representing near-field lighting as they assume distant light. Additionally, they have difficulty representing high-frequency light sources. Recent work [28] uses NeRF [29] as a near-field environment emitter. In this method, they first reconstruct the scene from multi-view images using NeRF. Subsequently, they partition the scene into the light source and the object using a bounding box, and then optimize the object geometry, reflectance, and the NeRF-represented light source. More recently, EnvGS [41] represents the light source using Gaussian primitives [12]. They employ two types of 2D Gaussians to represent the scene: a base Gaussian that represents the object's albedo and an environment Gaussian that represents the reflection color. By blending the colors derived from these two Gaussians, they can model accurate near-field reflections. Inspired by prior work, our approach uses 3D Gaussian primitives to represent the lighting environment with high fidelity.

3 Method

In this section, we present our approach for NaPS. Figure 2 provides an overview. We begin by describing the capture setup and data acquisition protocol. Next, we detail how the illumination is reconstructed from background observations. Finally, we introduce an inverse-rendering framework that jointly optimizes geometry and reflectance under the reconstructed illumination.

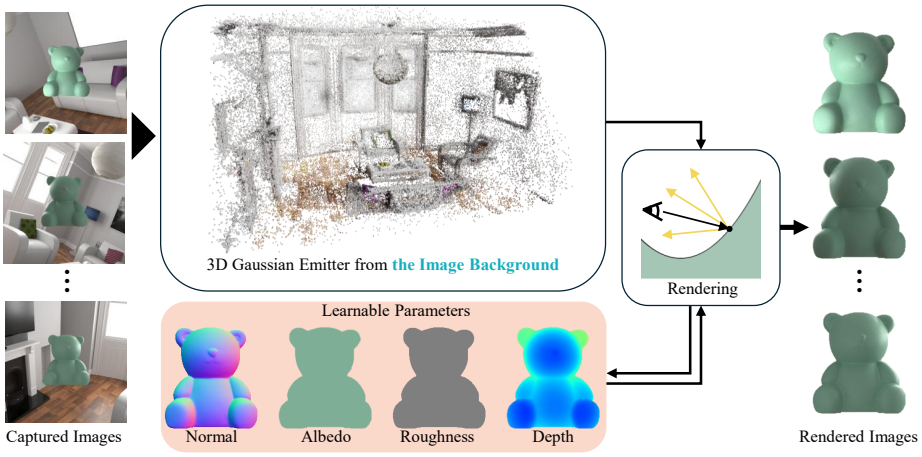


Fig. 2: Overview of the proposed method. We estimate 3D Gaussian primitives representing the lighting environment from the image background, captured with a fixed relative pose between the object and the camera. Then, using the reconstructed illumination, we optimize surface normals, albedo, roughness, and depth via inverse rendering.

3.1 Capture Setup

As shown in Fig. 3, we acquire N RGB images $\{\boldsymbol{I}_i\}_{i=1}^{N}$, where each image $\boldsymbol{I}_i \in \mathbb{R}^{H\times W\times 3}$ has spatial resolution $H \times W$, by rotating the camera and object together as a rigid pair while keeping their relative pose fixed. Because the camera pose with respect to the static scene changes at each capture, we estimate the extrinsics $\boldsymbol{P}_i = [\,\boldsymbol{R}_i \mid \boldsymbol{t}_i\,] \in \mathbb{R}^{3\times 4}$, where $\boldsymbol{R}_i \in \mathrm{SO}(3)$ and $\boldsymbol{t}_i \in \mathbb{R}^3$. Since the relative pose between the camera and object is fixed, the surface points $\boldsymbol{x}_c \in \mathbb{R}^3$ and normal $\boldsymbol{n}_c \in \mathbb{R}^3$ in the camera coordinate system are identical across all views. Their world-frame representations for the i-th view are

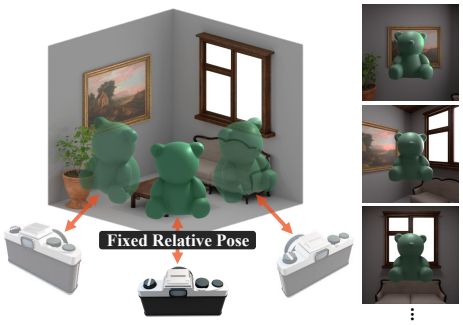


Fig. 3: Capture setup. Left: an example configuration. The relative pose between the camera and the object is fixed during image acquisition. Right: example observations from this setup. The object is seen from a fixed viewpoint under varying illumination, and the background records a static lighting environment from multiple viewpoints.

$$\boldsymbol{x}_w^i = \boldsymbol{R}_i^\top(\boldsymbol{x}_c - \boldsymbol{t}_i), \boldsymbol{n}_w^i = \boldsymbol{R}_i^\top \boldsymbol{n}_c, \quad (1)$$

where $\boldsymbol{x}_w^i, \boldsymbol{n}_w^i \in \mathbb{R}^3$. These relations fix the object geometry in the camera frame while accounting for view-dependent lighting in the world frame, which sets up our subsequent inverse-rendering formulation.

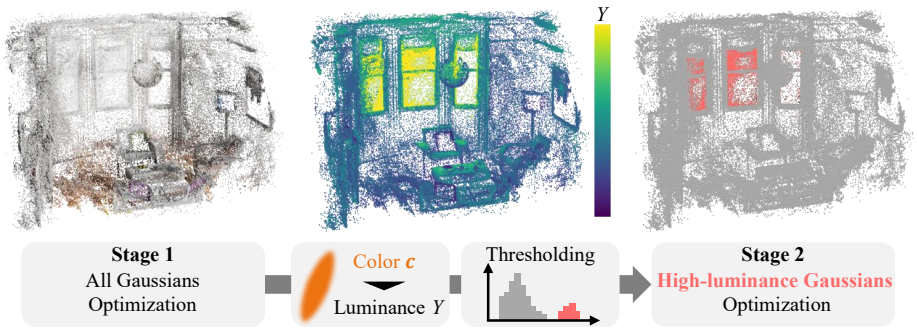


Fig. 4: Two-stage pipeline for reconstructing 3D Gaussian emitters. We robustly recover the lighting environment by training 3D Gaussians in two stages. In Stage 1, we optimize all Gaussian parameters using the original 3DGRT framework [30]. We then identify high-luminance Gaussians corresponding to the brightest regions via luminance-based weighting and thresholding. In Stage 2, we refine the reconstruction by optimizing only the color parameters of these high-luminance Gaussians while keeping all other parameters fixed.

For stable reconstruction of the illumination, we capture on the order of 100 images while varying the camera and object poses. The number of input images is comparable to that typically used for scene reconstruction with standard 3DGS [21] and for PS [36].

3.2 Illumination Reconstruction from Background

We explicitly reconstruct the surrounding illumination from the image background. To capture complex, potentially near-field lighting with high fidelity, we adopt 3DGS [21] and use the ray-tracing formulation of 3DGRT [30] to model light transport. Figure 4 shows the pipeline for reconstructing illumination with 3D Gaussians.

Preliminaries. 3DGS represents a scene with a set of M anisotropic 3D Gaussian primitives. The k-th Gaussian is parameterized by its mean $\boldsymbol{\mu}_k \in \mathbb{R}^3$ and covariance $\boldsymbol{\Sigma}_k \in \mathbb{R}^{3\times3}$, and its density at a point $\boldsymbol{x} \in \mathbb{R}^3$ is

$$G_k(\boldsymbol{x}) \;=\; \exp\!\big(-(\boldsymbol{x}-\boldsymbol{\mu}_k)^\top \boldsymbol{\Sigma}_k^{-1}(\boldsymbol{x}-\boldsymbol{\mu}_k)\big). \tag{2}$$

To ensure $\boldsymbol{\Sigma}_k$ is positive semidefinite, we use the factorization $\boldsymbol{\Sigma}_k = \boldsymbol{R}_k \boldsymbol{S}_k \boldsymbol{S}_k^\top \boldsymbol{R}_k^\top$ with $\boldsymbol{R}_k \in \mathrm{SO}(3)$ and a diagonal scaling $\boldsymbol{S}_k = \mathrm{diag}(s_{k,x}, s_{k,y}, s_{k,z})$. Each primitive carries an opacity parameter $\sigma_k \in \mathbb{R}_{\geq 0}$ and a view-dependent color $\boldsymbol{c}_k(\boldsymbol{v}) \in \mathbb{R}^3$ modeled with spherical harmonics (SH), *i.e.*, $\boldsymbol{c}_k(\boldsymbol{v}) = \sum_{\ell,m} \boldsymbol{\beta}_{k,\ell m}\, Y_{\ell m}(\boldsymbol{v})$.

Given a camera ray $\boldsymbol{r}(\tau) = \boldsymbol{o} + \tau \boldsymbol{v}$ with origin $\boldsymbol{o} \in \mathbb{R}^3$ and direction $\boldsymbol{v} \in \mathbb{R}^3$, the rendered color $\boldsymbol{C}(\boldsymbol{o}, \boldsymbol{v}) \in \mathbb{R}^3$ is obtained by front-to-back alpha compositing

of depth-sorted Gaussians:

$$\boldsymbol{C}(\boldsymbol{o},\boldsymbol{v}) = \sum_{k=1}^{M} \boldsymbol{c}_k(\boldsymbol{v})\,\alpha_k \prod_{j=1}^{k-1}\big(1-\alpha_j\big), \tag{3}$$

$$\alpha_k = 1 - \exp\big(-\sigma_k\, G_k(\boldsymbol{x}_k)\big), \tag{4}$$

where $\boldsymbol{x}_k = \boldsymbol{o} + \tau_k \boldsymbol{v}$ denotes the ray position associated with the k-th primitive after depth sorting. In practice, we fit Gaussians only to the background regions of the captured views, using SfM poses and masks, and then employ 3DGRT to query incident radiance and visibility while accounting for near-field effects. This background-driven reconstruction yields a compact, spatially aware illumination field that serves as the emitter in the subsequent inverse-rendering optimization for 3D reconstruction of the target object.

Training. We train the background 3DGS using 3DGRT [30] with a background mask so that only background pixels supervise the reconstruction. To faithfully capture illumination, the model must predict high-dynamic-range (HDR) radiance. However, because 3DGRT relies on low-dynamic-range (LDR) inputs, its direct application to HDR images reduces accuracy. To address this, we employ a two-stage training strategy to improve robustness under HDR illumination. In the first stage, we reconstruct the entire scene by optimizing all 3D Gaussian parameters. During this stage, we adopt an HDR-robust objective via a log-intensity L_1 loss:

$$\mathcal{L}_{\text{stage1}} \;=\; \frac{1}{N_b}\sum_{j=1}^{N_b}\big\|\log(1+\hat{\boldsymbol{m}}_j) - \log(1+\boldsymbol{m}_j)\big\|_1, \tag{5}$$

where N_b is the number of pixels in the background regions of the images, $\hat{\boldsymbol{m}}_j \in \mathbb{R}^3$ and $\boldsymbol{m}_j \in \mathbb{R}^3$ denote the observed and rendered color at the j-th background pixel, respectively.

While the first stage reconstructs the overall scene structure, it may fail to capture the correct radiance, particularly in high-luminance regions. Consequently, the second stage focuses on accurately reconstructing these bright regions. Specifically, we first isolate the subset of 3D Gaussians that correspond to high-luminance regions, denoted as high-luminance Gaussians. To identify high-luminance Gaussians, we employ Otsu's method [31] based on a weight Y assigned to each primitive, where Y is the luminance value derived from its RGB color. We then optimize only the color $\boldsymbol{c}$ of the high-luminance Gaussians while keeping other parameters fixed. For color refinement, we use a weighted L_1 loss:

$$\mathcal{L}_{\text{stage2}} \;=\; \frac{1}{N_b}\sum_{j=1}^{N_b}\left\|\frac{\hat{\boldsymbol{m}}_j - \boldsymbol{m}_j}{\text{sg}(\boldsymbol{m}_j)+\epsilon}\right\|_1, \tag{6}$$

where $\text{sg}(\cdot)$ is the stop-gradient operator and $\epsilon \in \mathbb{R}_+$ is a small constant that prevents division by zero.

3.3 Optimization of Surface Normals and Materials

Our inverse-rendering stage optimizes the surface normal field $\boldsymbol{n}$, the per-pixel BRDF parameters of a simplified Disney model $f(\cdot)$ [2], and the depth map $\boldsymbol{z}$ while keeping the 3DGS-based emitter from Sec. 3.2 fixed.

Rendering equation. The outgoing radiance L_o at a surface point $\boldsymbol{x}$ in direction $\boldsymbol{\omega}_o$ follows the rendering equation [19]:

$$L_o(\boldsymbol{x}, \boldsymbol{\omega}_o) \;=\; \int_{\Omega(\boldsymbol{n})} L_i(\boldsymbol{x}, \boldsymbol{\omega}_i)\, f(\boldsymbol{x}, \boldsymbol{\omega}_o, \boldsymbol{\omega}_i)\, (\boldsymbol{\omega}_i \cdot \boldsymbol{n})\, d\boldsymbol{\omega}_i, \tag{7}$$

where $L_i(\boldsymbol{x}, \boldsymbol{\omega}_i)$ is incident radiance from direction $\boldsymbol{\omega}_i$ and $\Omega(\boldsymbol{n})$ is the hemisphere oriented by the surface normal $\boldsymbol{n}$. Direct evaluation is intractable because it requires integrating over directions, so we use path tracing with Monte Carlo (MC) estimation. Let N_s be the number of MC samples per pixel, with samples $\{\boldsymbol{\omega}_i^{(k)}\}_{k=1}^{N_s}$ drawn from a probability density $p(\boldsymbol{\omega})$. We estimate Eq. (7) by

$$\widehat{L}_o(\boldsymbol{x}, \boldsymbol{\omega}_o) = \frac{1}{N_s} \sum_{k=1}^{N_s} \frac{L_i(\boldsymbol{x}, \boldsymbol{\omega}_i^{(k)})\, f(\boldsymbol{x}, \boldsymbol{\omega}_o, \boldsymbol{\omega}_i^{(k)})\, \left(\boldsymbol{\omega}_i^{(k)} \cdot \boldsymbol{n}\right)}{p\left(\boldsymbol{\omega}_i^{(k)}\right)}. \tag{8}$$

This estimator forms the basis of our differentiable renderer in which L_i is queried from the emitter and the surface parameters appear in f and $\boldsymbol{n}$.

BRDF model. We use the two-parameter Disney BRDF [2] with per-pixel albedo $\boldsymbol{a}(\boldsymbol{x}) \in [0,1]^3$ and roughness $r(\boldsymbol{x}) \in [0,1]$. The BRDF is decomposed into diffuse and specular terms:

$$f(\boldsymbol{x}, \boldsymbol{\omega}_o, \boldsymbol{\omega}_i) \;=\; f_d(\boldsymbol{x}) + f_s(\boldsymbol{x}, \boldsymbol{\omega}_o, \boldsymbol{\omega}_i), \tag{9}$$

$$f_d(\boldsymbol{x}) \;=\; \frac{\boldsymbol{a}(\boldsymbol{x})}{\pi}, \tag{10}$$

$$f_s(\boldsymbol{x}, \boldsymbol{\omega}_o, \boldsymbol{\omega}_i) \;=\; \frac{D(\boldsymbol{\omega}_h, r(\boldsymbol{x}))\, F(\boldsymbol{\omega}_o, \boldsymbol{\omega}_h)\, G(\boldsymbol{\omega}_i, \boldsymbol{\omega}_o, \boldsymbol{n}, r(\boldsymbol{x}))}{4\, (\boldsymbol{n} \cdot \boldsymbol{\omega}_i)\, (\boldsymbol{n} \cdot \boldsymbol{\omega}_o)}, \tag{11}$$

where $\boldsymbol{\omega}_h = \frac{\boldsymbol{\omega}_i + \boldsymbol{\omega}_o}{\|\boldsymbol{\omega}_i + \boldsymbol{\omega}_o\|_2}$ is the half-vector, D is the GGX normal distribution, F is Schlick's approximation to the Fresnel term [35], and G is the geometric attenuation.

Emitter query and visibility. Given a surface point $\boldsymbol{x}$ and direction $\boldsymbol{\omega}_i$, incident radiance $L_i(\boldsymbol{x}, \boldsymbol{\omega}_i)$ from the 3DGS emitter is evaluated using Eq. (3). This accounts for direct illumination but not occlusion, so we compute binary visibility $V(\boldsymbol{x}, \boldsymbol{\omega}_i)$ using a mesh reconstructed from the depth map $\boldsymbol{z}$. We define $V(\boldsymbol{x}, \boldsymbol{\omega}_i) = 0$ if the ray from $\boldsymbol{x}$ along $\boldsymbol{\omega}_i$ intersects the object geometry and $V(\boldsymbol{x}, \boldsymbol{\omega}_i) = 1$ otherwise. The effective incident radiance is then $V(\boldsymbol{x}, \boldsymbol{\omega}_i)\, L_i(\boldsymbol{x}, \boldsymbol{\omega}_i)$, which enables direct lighting with cast shadows for the subsequent optimization.

Importance sampling. High-quality MC estimates require many samples, which is computationally expensive. We therefore combine cosine-weighted hemisphere sampling with emitter-guided light sampling tailored to 3DGS. Inspired by NeRF emitter sampling [28], we construct a point set directly from the centers of the Gaussians and fit a Gaussian mixture model (GMM) with $K = 64$ components. The i-th component has mean $\boldsymbol{\mu}_i \in \mathbb{R}^3$, scalar concentration proxy $\kappa_i \in \mathbb{R}_{>0}$, and weight $\lambda_i \in \mathbb{R}_{>0}$. For a shading point at $\boldsymbol{x}$, we project each component onto the unit sphere to obtain a von Mises–Fisher (vMF) distribution with parameters:

$$\hat{\boldsymbol{\mu}}_i \;=\; \frac{\boldsymbol{\mu}_i - \boldsymbol{x}}{\|\boldsymbol{\mu}_i - \boldsymbol{x}\|}, \qquad \hat{\kappa}_i \;=\; s\,\frac{\kappa_i}{\|\boldsymbol{\mu}_i - \boldsymbol{x}\|}, \qquad \hat{\lambda}_i \;=\; \lambda_i, \tag{12}$$

where s controls the sharpness of directional sampling and is set to 10 in our experiments. We draw directions from this vMF mixture using the method of [16] and employ multiple importance sampling (MIS) [32,38] to combine light sampling with cosine sampling. This strategy concentrates samples toward bright, nearby emitters while maintaining low variance in diffuse lobes.

Losses. We optimize a weighted sum of color, normal, and smoothness terms:

$$\mathcal{L} \;=\; \mathcal{L}_{\text{color}} + \mathcal{L}_{\text{normal}} + \lambda_{\text{smooth}}\,\mathcal{L}_{\text{smooth}}, \tag{13}$$

where λ_{smooth} controls the regularization strength. To handle HDR image observations, we use a log-intensity L_1 color loss between the observed color $\hat{\boldsymbol{m}}_j$ and the rendered color $\boldsymbol{m}_j$:

$$\mathcal{L}_{\text{color}} \;=\; \frac{1}{N_f}\sum_{j=1}^{N_f}\big\|\log(\mathbf{1}+\hat{\boldsymbol{m}}_j) - \log(\mathbf{1}+\boldsymbol{m}_j)\big\|_1, \tag{14}$$

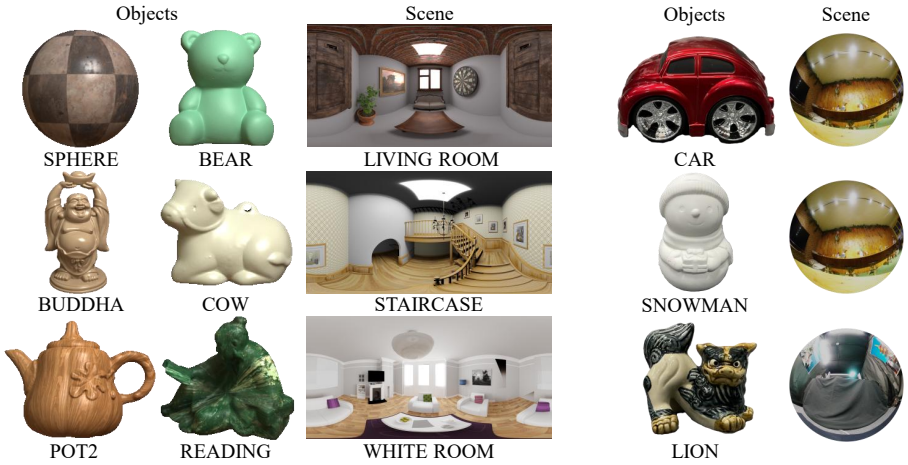
where N_f is the number of pixels in the mask of the target object. For normal supervision, we encourage agreement between the optimized normal $\boldsymbol{n}$ and a pseudo normal $\hat{\boldsymbol{n}}$ computed by differentiating the depth map $\boldsymbol{z}$. We adopt a cosine similarity loss:

$$\mathcal{L}_{\text{normal}} \;=\; \frac{1}{N_f}\sum_{j=1}^{N_f}\left(1 - \frac{\hat{\boldsymbol{n}}_j^{\top}\boldsymbol{n}_j}{\|\hat{\boldsymbol{n}}_j\|\,\|\boldsymbol{n}_j\|}\right). \tag{15}$$

Finally, we impose total-variation (TV) regularization on albedo $\boldsymbol{a}$, roughness r, and depth $\boldsymbol{z}$ to suppress spurious high-frequency noise:

$$\mathcal{L}_{\text{smooth}} \;=\; \text{TV}(\boldsymbol{a}) + \text{TV}(r) + \text{TV}(\boldsymbol{z}), \tag{16}$$

where $\text{TV}(I) = \frac{1}{N_f}\sum_{j=1}^{N_f}\sqrt{|\partial_u I|^2 + |\partial_v I|^2}$ denotes isotropic total variation. Together, these losses guide stable optimization of geometry and reflectance under the reconstructed, spatially aware illumination field.



(a) Synthetic dataset. We use six objects and three scenes. **(b)** Real-world dataset.

Fig. 5: Dataset used in the experiments.

4 Experiments

We evaluate our method on both synthetic and real-world data. Since no existing dataset matches our capture configuration, we constructed a new dataset for this study, as shown in Fig. 5. We first describe the experimental setup and then report quantitative and qualitative results.

Implementation details. To reconstruct the surrounding illumination via a two-stage approach with 3D Gaussians, we use the default learning rates of 3DGRT [30] during the first stage. In the second stage, we set the learning rate for the color $\boldsymbol{c}$ of the 3D Gaussians to 1×10^{-1} to fit HDR radiance. We train each stage for $30K$ iterations.

During optimization of surface normals and materials, we use 256 samples per pixel. We optimize for 100 epochs with a batch size of 4. On an NVIDIA RTX 6000 Ada GPU, this optimization typically takes about two hours. We use the Adam optimizer with default parameters. The learning rates are set to 5×10^{-4}, 1×10^{-2}, 1×10^{-2}, and 1×10^{-4} for the normal $\boldsymbol{n}$, albedo $\boldsymbol{a}$, roughness r, and depth $\boldsymbol{z}$, respectively, and each learning rate is halved every 25 epochs. The weight of the smoothness loss λ_{smooth} is 0.01 for the first 50 epochs and 0 thereafter.

To facilitate convergence, we initialize the normal and roughness with the estimates from [23]. Furthermore, we obtain the initial depth by normal integration [3]. Because depth has a scale ambiguity, we rescale the estimated depth in synthetic experiments so that its mean matches the ground-truth mean. For real data, we set the camera-to-object distance to 1 and normalize the estimated depth so that the mean is 1. The visibility mesh is updated at each epoch. Finally, we apply a sigmoid activation to the albedo and roughness heads to constrain their ranges to $[0, 1]$, which stabilizes optimization.

Table 1: Quantitative comparison results of normal, albedo, and roughness on the synthetic dataset. The reported numbers represent the means of three lighting environments in terms of MAngE (Normal), PSNR (Albedo), and MAbsE (Roughness). **Bold numbers** indicate the best results.

	SDM-UniPS [15]			LINO-UniPS [23]			Ours		
	Normal ↓	Albedo ↑	Roughness ↓	Normal ↓	Albedo ↑	Roughness ↓	Normal ↓	Albedo ↑	Roughness ↓
SPHERE	2.94	27.9	0.107	3.47	28.9	0.0728	**1.38**	**34.1**	**0.0341**
BEAR	4.05	**34.4**	0.0305	5.81	33.7	**0.0189**	**2.97**	30.2	0.0536
BUDDHA	**6.52**	26.2	0.173	7.52	24.8	**0.109**	7.31	**29.3**	0.144
COW	4.48	29.3	0.0700	5.64	**31.3**	0.0896	**1.53**	30.4	**0.0276**
POT2	4.91	23.5	0.0603	6.36	22.6	0.0556	**2.57**	**34.1**	**0.0422**
READING	6.60	27.8	0.0573	7.90	29.2	**0.0502**	**4.32**	**36.8**	0.0643
Average	4.92	28.2	0.0830	6.12	28.4	0.0659	**3.35**	**32.5**	**0.0610**

Baseline methods. We compare the proposed method with SDM-UniPS [15] and LINO-UniPS [23], which are learning-based methods that estimate surface normals, albedo, and roughness. These methods cannot process a large number of images simultaneously due to memory constraints. Therefore, we evaluate the comparison methods by averaging the results over 10 independent runs. In each run, the estimation is performed using 100 images randomly sampled from the full set of images.

4.1 Synthetic Experiments

Evaluation metrics. We report the mean angular error (MAngE) for normals, the peak signal-to-noise ratio (PSNR) for albedo, and the mean absolute error (MAbsE) for roughness. Because the albedo estimated by the comparison methods has scale ambiguity, we rescale it so that its mean matches the ground-truth mean albedo, ensuring a fair comparison.

Dataset. Figure 5a shows our synthetic dataset, which is rendered with Mitsuba 3 [17]. We use five meshes obtained from DiLiGenT-MV [25] (BEAR, BUDDHA, COW, POT2, and READING) as well as a generated mesh (SPHERE), with materials sourced from Poly Haven[3], a 3D asset library. The lighting environments (LIVING ROOM, STAIRCASE, and WHITE ROOM) are sourced from Poly Haven[3] and [1]. We render 150 HDR images for each object at a resolution of 512×512 from randomly sampled poses.

Results. Table 1 and Fig. 6 show the quantitative and visual evaluations, respectively. Comprehensive visualizations are provided in the supplementary materials. The learning-based methods, SDM-UniPS and LINO-UniPS, estimate surface normals, albedo, and roughness independently, which can lead to inconsistencies among these factors in this ill-posed setting. In contrast, by leveraging background observations of the surrounding illumination, our method performs inverse rendering and achieves accurate estimates while enforcing consistency among lighting, shape, and reflectance.

[3]Poly Haven, `https://polyhaven.com/`, last accessed on June 30, 2026.

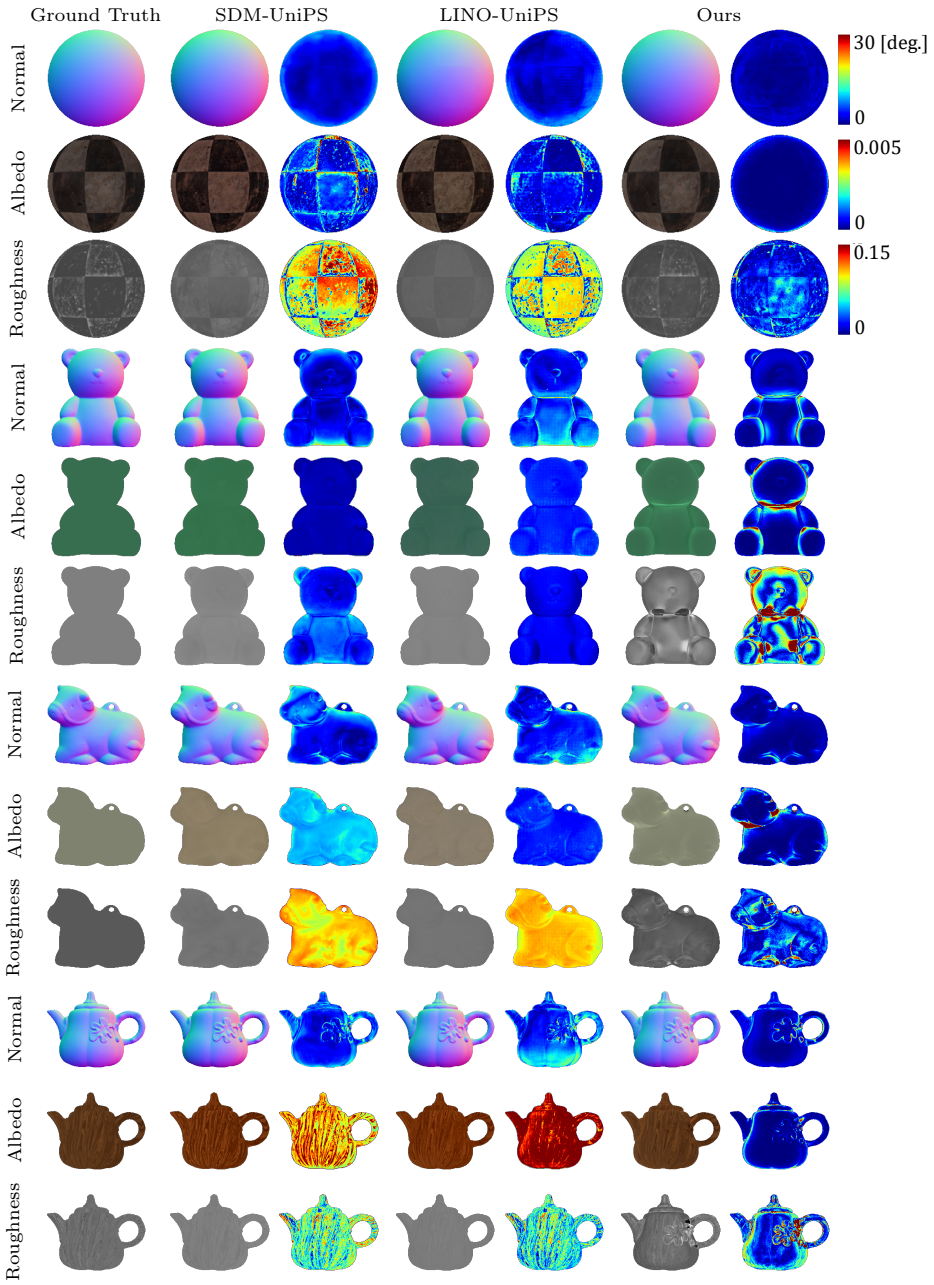


Fig. 6: Qualitative comparisons on LIVING ROOM of the synthetic scenes. For each object and method, we present the estimation results and their corresponding error maps side by side.

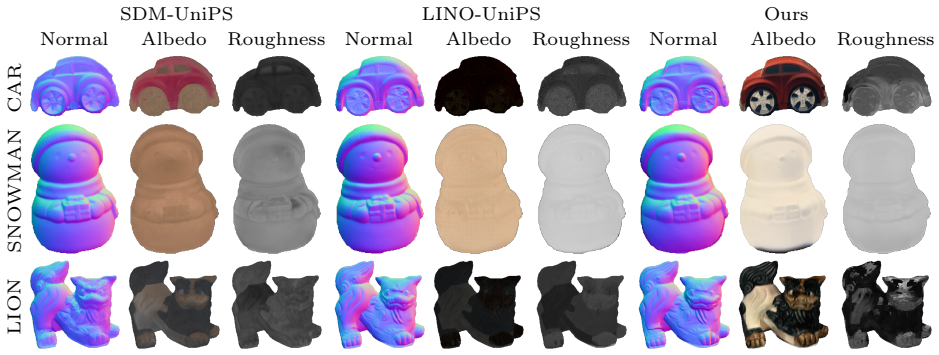


Fig. 7: Visual comparisons on the real-world scene.

4.2 Real-world Experiments

Dataset. We capture approximately 130 HDR images of three objects: CAR, SNOWMAN, and LION in two real-world environments, a dark room and a natural indoor environment, as shown in Fig. 5b, using a Canon EOS R8 camera with bracketed exposures. Each HDR image is reconstructed from three images captured at different exposures. We use Agisoft Metashape[4] to estimate camera parameters and SegNeXt [9] to obtain foreground masks.

Results. Figure 7 presents qualitative results of our method on the real-world captures, including the estimated normals and materials. Supplementary materials provide additional visual examples. Although ground truth is unavailable, our approach yields more plausible reconstructions than the learning-based baselines, which is consistent with the trend observed in the synthetic experiments. The estimated surface normals exhibit coherent geometry with detailed shapes, while the albedo maps preserve both consistent and fine-grained material details without contamination from shading. In particular, existing learning-based methods often produce less accurate albedo estimates, which are crucial for recovering the object's appearance. Our method, in contrast, produces reasonable albedo estimates thanks to the inverse rendering pipeline. The recovered roughness also closely corresponds to visually perceived glossiness, demonstrating that our method effectively disentangles intrinsic reflectance from illumination effects.

4.3 Effect of the Number of Viewpoints

To investigate how the number of viewpoints affects estimation accuracy, we conduct experiments using fewer input views on the same synthetic dataset as in the main experiment. As shown in Tab. 2, the accuracy of the state-of-the-art UniPS methods changes only marginally as the number of views increases, whereas our method improves substantially with more viewpoints. As the number of views

[4]Agisoft Metashape, `https://www.agisoft.com/`, last accessed on June 30, 2026.

Table 2: Quantitative results with varying numbers of views. The reported numbers represent the average over the synthetic dataset in terms of MAngE for normals, PSNR for albedo, and MAbsE for roughness. For SDM-UniPS and LINO-UniPS, when more than 100 views are available, we randomly sample 100 images due to memory constraints and report the average over 10 independent runs. **Bold numbers** indicate the best results.

#views	SDM-UniPS [15]			LINO-UniPS [23]			Ours		
	Normal ↓	Albedo ↑	Roughness ↓	Normal ↓	Albedo ↑	Roughness ↓	Normal ↓	Albedo ↑	Roughness ↓
40	**4.84**	27.7	0.0778	5.96	**27.9**	**0.0601**	5.10	23.6	0.125
80	4.79	28.2	0.0822	6.02	28.2	**0.0659**	**3.59**	**32.1**	0.0676
150	4.92	28.2	0.0830	6.12	28.4	0.0659	**3.35**	**32.5**	**0.0610**

increases, the background is observed more densely, enabling more accurate reconstruction of the lighting environment. This improved illumination reconstruction directly benefits subsequent inverse rendering optimization, leading to better object reconstruction. These results indicate that our method is particularly effective when sufficiently dense background observations are available. Although accuracy decreases as the number of views decreases, our method still achieves reasonable performance even with 40 views. These results suggest that highly accurate background geometry is not always required, since the background is primarily used to capture illumination rather than detailed geometry.

4.4 Ablation Study

To validate the contributions of our proposed components, we perform ablation studies on the synthetic dataset used in the main experiments. Table 3 shows the quantitative results for each configuration. For all ablations, the number of ray samples per pixel is set to 256.

Light sampling. To assess the effect of light sampling with the 3DGS-based emitter, we compare it against a multiple importance sampling (MIS) baseline combining cosine sampling with GGX importance sampling [11]. As shown in Tab. 3, under natural illumination, radiance is dominated by a few high-intensity regions, and our emitter-guided strategy samples these regions more effectively than the MIS baseline, yielding lower error under the same sample budget.

Two-stage optimization of 3D Gaussian emitter. We validate our two-stage training strategy for the 3D Gaussian emitter by comparing it with a single-stage baseline. Figure 8 shows the final renderings of the lighting environments and the mean squared error (MSE) maps for both stages. While single-stage optimization captures the overall scene structure, it struggles to fit the radiance in bright regions. This mismatch degrades estimation accuracy, particularly for albedo, as reported in Tab. 3. Because bright regions often correspond to the dominant illumination sources, the two-stage optimization that explicitly fits their radiance is essential to our method.

Table 3: Ablation studies of emitter-guided light sampling and two-stage optimization of 3D Gaussian emitter.

Method	Normal ↓	Albedo ↑	Roughness ↓
Ours	**3.35**	**32.5**	**0.0610**
w/o light sampling	3.78	32.2	0.0731
w/o two-stage	3.59	25.9	0.0684

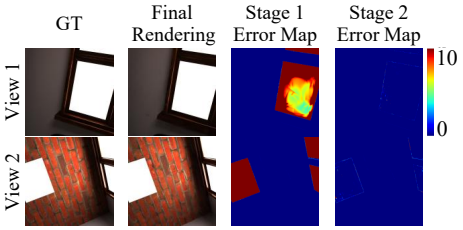


Fig. 8: Results of two-stage 3D Gaussian emitter optimization.

5 Conclusion

We presented an inverse-rendering-based method for NaPS that explicitly leverages the static background captured under flexible rigid motion of the camera-object pair while keeping the relative pose between the camera and object fixed. Rather than masking out the background as in conventional PS pipelines, we reconstruct the surrounding illumination from background observations using a 3DGS representation. With this 3DGS-based emitter, we then jointly optimize surface normals, albedo, roughness, and depth via differentiable rendering. This formulation turns the severely ill-posed, uncalibrated NaPS problem under near-field illumination into a tractable optimization that enforces consistency among illumination, geometry, and reflectance.

Experiments on both synthetic and real indoor scenes demonstrate that our approach substantially improves the accuracy of shape and reflectance estimation compared with state-of-the-art learning-based methods in natural-light settings. These results indicate that treating the background as a useful source of illumination cues provides a promising direction for practical photometric stereo in everyday environments.

Limitations. The proposed method does not explicitly model global illumination effects on the target object, *e.g.*, inter-reflections, which is one of its main limitations and a promising direction for future work. Another limitation is that the optimization relies on computationally expensive differentiable rendering using a 3DGS-based emitter, which makes it slower than purely learning-based approaches. Extending the framework to account for indirect illumination while reducing computational overhead, for example, by integrating learned priors or more efficient neural rendering techniques, is an interesting avenue for future research.

Acknowledgments

This work was supported by JSPS KAKENHI Grant Numbers JP23H05491, JP25K03140, and JP25K00142, and by JST FOREST JPMJFR206F.

References

1. Bitterli, B.: Rendering resources (2016), `https://benedikt-bitterli.me/resources/`, last accessed on June 30, 2026.
2. Burley, B., Studios, W.D.A.: Physically-based shading at Disney. In: Proceedings of ACM Annual Conference on Computer Graphics and Interactive Techniques (SIGGRAPH) (2012)
3. Cao, X., Santo, H., Shi, B., Okura, F., Matsushita, Y.: Bilateral normal integration. In: Proceedings of European Conference on Computer Vision (ECCV) (2022)
4. Chen, G., Han, K., Shi, B., Matsushita, Y., Wong, K.Y.K.: Self-calibrating deep photometric stereo networks. In: Proceedings of IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) (2019)
5. Chen, G., Han, K., Shi, B., Matsushita, Y., Wong, K.Y.K.: Deep photometric stereo for non-lambertian surfaces. IEEE Transactions on Pattern Analysis and Machine Intelligence **44**(1), 129–142 (2020)
6. Chen, G., Han, K., Wong, K.Y.K.: PS-FCN: A flexible learning framework for photometric stereo. In: Proceedings of European Conference on Computer Vision (ECCV) (2018)
7. Chen, G., Waechter, M., Shi, B., Wong, K.Y.K., Matsushita, Y.: What is learned in deep uncalibrated photometric stereo? In: Proceedings of European Conference on Computer Vision (ECCV) (2020)
8. Gao, J., Gu, C., Lin, Y., Li, Z., Zhu, H., Cao, X., Zhang, L., Yao, Y.: Relightable 3D gaussians: Realistic point cloud relighting with brdf decomposition and ray tracing. In: Proceedings of European Conference on Computer Vision (ECCV) (2024)
9. Guo, M.H., Lu, C.Z., Hou, Q., Liu, Z., Cheng, M.M., Hu, S.M.: SegNeXt: Rethinking convolutional attention design for semantic segmentation. Proceedings of Annual Conference on Neural Information Processing Systems (NeurIPS) **35**, 1140–1156 (2022)
10. Hardy, C., Quéau, Y., Tschumperlé, D.: Uni MS-PS: A multi-scale encoder-decoder transformer for universal photometric stereo. Computer Vision and Image Understanding **248**, 104093 (2024)
11. Heitz, E.: Sampling the GGX distribution of visible normals. Journal of Computer Graphics Techniques (JCGT) (2018)
12. Huang, B., Yu, Z., Chen, A., Geiger, A., Gao, S.: 2D gaussian splatting for geometrically accurate radiance fields. In: Proceedings of ACM Annual Conference on Computer Graphics and Interactive Techniques (SIGGRAPH) (2024)
13. Ikehata, S.: CNN-PS: CNN-based photometric stereo for general non-convex surfaces. In: Proceedings of European Conference on Computer Vision (ECCV) (2018)
14. Ikehata, S.: Universal photometric stereo network using global lighting contexts. In: Proceedings of IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) (2022)
15. Ikehata, S.: Scalable, detailed and mask-free universal photometric stereo. In: Proceedings of IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) (2023)
16. Jakob, W.: Numerically stable sampling of the von Mises-Fisher distribution on $\mathcal{S}^2$ (and other tricks). Interactive Geometry Lab, ETH Zürich, Technical Report p. 6 (2012)
17. Jakob, W., Speierer, S., Roussel, N., Nimier-David, M., Vicini, D., Zeltner, T., Nicolet, B., Crespo, M., Leroy, V., Zhang, Z.: Mitsuba 3 renderer (2022), `https://mitsuba-renderer.org`, last accessed on June 30, 2026.

18. Ju, Y., Lam, K.M., Xie, W., Zhou, H., Dong, J., Shi, B.: Deep learning methods for calibrated photometric stereo and beyond. IEEE Transactions on Pattern Analysis and Machine Intelligence **46**(11), 7154–7172 (2024)
19. Kajiya, J.T.: The rendering equation. In: Proceedings of ACM Annual Conference on Computer Graphics and Interactive Techniques (SIGGRAPH). pp. 143–150 (1986)
20. Kaya, B., Kumar, S., Oliveira, C., Ferrari, V., Van Gool, L.: Uncalibrated neural inverse rendering for photometric stereo of general surfaces. In: Proceedings of IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) (2021)
21. Kerbl, B., Kopanas, G., Leimkühler, T., Drettakis, G.: 3D Gaussian Splatting for real-time radiance field rendering. ACM Transactions on Graphics (TOG) **42**(4) (2023)
22. Kuang, Z., Olszewski, K., Chai, M., Huang, Z., Achlioptas, P., Tulyakov, S.: NeROIC: Neural rendering of objects from online image collections. ACM Transactions on Graphics (TOG) **41**(4), 1–12 (2022)
23. Li, H., Chen, H., Ye, C., Chen, Z., Li, B., Xu, S., Guo, X., Liu, X., Wang, Y., Zhang, B., Ikehata, S., Shi, B., Rao, A., Zhao, H.: Light of normals: Unified feature representation for universal photometric stereo. arXiv preprint arXiv:2506.18882 (2025)
24. Li, J., Li, H.: Self-calibrating photometric stereo by neural inverse rendering. In: Proceedings of European Conference on Computer Vision (ECCV) (2022)
25. Li, M., Zhou, Z., Wu, Z., Shi, B., Diao, C., Tan, P.: Multi-view photometric stereo: A robust solution and benchmark dataset for spatially varying isotropic materials. IEEE Transactions on Image Processing (2020)
26. Li, Z., Lu, Z., Yan, H., Shi, B., Pan, G., Zheng, Q., Jiang, X.: Spin-UP: Spin light for natural light uncalibrated photometric stereo. In: Proceedings of IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) (2024)
27. Li, Z., Zheng, Q., Shi, B., Pan, G., Jiang, X.: DANI-Net: Uncalibrated photometric stereo by differentiable shadow handling, anisotropic reflectance modeling, and neural inverse rendering. In: Proceedings of IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) (2023)
28. Ling, J., Yu, R., Xu, F., Du, C., Zhao, S.: NeRF as a non-distant environment emitter in physics-based inverse rendering. In: Proceedings of ACM Annual Conference on Computer Graphics and Interactive Techniques (SIGGRAPH) (2024)
29. Mildenhall, B., Srinivasan, P.P., Tancik, M., Barron, J.T., Ramamoorthi, R., Ng, R.: NeRF: Representing scenes as neural radiance fields for view synthesis. In: Proceedings of European Conference on Computer Vision (ECCV) (2020)
30. Moenne-Loccoz, N., Mirzaei, A., Perel, O., de Lutio, R., Esturo, J.M., State, G., Fidler, S., Sharp, N., Gojcic, Z.: 3D Gaussian Ray Tracing: Fast tracing of particle scenes. ACM Transactions on Graphics (TOG) (2024)
31. Otsu, N.: A threshold selection method from gray-level histograms. IEEE Transactions on Systems, Man, and Cybernetics **9**(1), 62–66 (1979)
32. Pharr, M., Jakob, W., Humphreys, G.: Physically Based Rendering: From Theory to Implementation. Morgan Kaufmann Publishers Inc. (2016)
33. Santo, H., Samejima, M., Sugano, Y., Shi, B., Matsushita, Y.: Deep photometric stereo network. In: Proceedings of IEEE/CVF International Conference on Computer Vision Workshops (ICCVW). pp. 501–509 (2017)
34. Santo, H., Samejima, M., Sugano, Y., Shi, B., Matsushita, Y.: Deep photometric stereo networks for determining surface normal and reflectances. IEEE Transactions on Pattern Analysis and Machine Intelligence **44**(1), 114–128 (2020)

35. Schlick, C.M.: An inexpensive BRDF model for physically-based rendering. Computer Graphics Forum **13** (1994)
36. Shi, B., Wu, Z., Mo, Z., Duan, D., Yeung, S.K., Tan, P.: A benchmark dataset and evaluation for non-lambertian and uncalibrated photometric stereo. In: Proceedings of IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 3707–3716 (2016)
37. Silver, W.M.: Determining shape and reflectance using multiple images. Master's thesis, Massachusetts Institute of Technology (1980)
38. Veach, E., Guibas, L.J.: Optimally combining sampling techniques for Monte Carlo rendering. In: Proceedings of ACM Annual Conference on Computer Graphics and Interactive Techniques (SIGGRAPH). pp. 419–428 (1995)
39. Woodham, R.J.: Photometric method for determining surface orientation from multiple images. Optical Engineering **19**(1), 139–144 (1980)
40. Wu, H., Hu, Z., Li, L., Zhang, Y., Fan, C., Yu, X.: NeFII: Inverse rendering for reflectance decomposition with near-field indirect illumination. In: Proceedings of IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) (2023)
41. Xie, T., Chen, X., Xu, Z., Xie, Y., Jin, Y., Shen, Y., Peng, S., Bao, H., Zhou, X.: EnvGS: Modeling view-dependent appearance with environment Gaussian. In: Proceedings of IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) (2025)
42. Yu, Y., Smith, W.A.: Outdoor inverse rendering from a single image using multiview self-supervision. IEEE Transactions on Pattern Analysis and Machine Intelligence **44**(7), 3659–3675 (2021)
43. Zhang, K., Luan, F., Wang, Q., Bala, K., Snavely, N.: PhySG: Inverse rendering with spherical gaussians for physics-based material editing and relighting. In: Proceedings of IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) (2021)
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)