

SpecNeRF: Gaussian Directional Encoding for Specular Reflections

Li Ma^{1,2} Vasu Agrawal² Haithem Turki^{2,3} Changil Kim²
Chen Gao² Pedro Sander¹ Michael Zollhöfer² Christian Richardt²

¹The Hong Kong University of Science and Technology

²Meta Reality Labs ³Carnegie Mellon University

Abstract

Neural radiance fields have achieved remarkable performance in modeling the appearance of 3D scenes. However, existing approaches still struggle with the view-dependent appearance of glossy surfaces, especially under complex lighting of indoor environments. Unlike existing methods, which typically assume distant lighting like an environment map, we propose a learnable Gaussian directional encoding to better model the view-dependent effects under near-field lighting conditions. Importantly, our new directional encoding captures the spatially-varying nature of near-field lighting and emulates the behavior of prefiltered environment maps. As a result, it enables the efficient evaluation of preconvolved specular color at any 3D location with varying roughness coefficients. We further introduce a data-driven geometry prior that helps alleviate the shape radiance ambiguity in reflection modeling. We show that our Gaussian directional encoding and geometry prior significantly improve the modeling of challenging specular reflections in neural radiance fields, which helps decompose appearance into more physically meaningful components.

1. Introduction

Neural radiance fields (NeRFs) have emerged as a popular scene representation for novel-view synthesis [34, 45, 51]. By training a neural network based on sparse observations of a 3D scene, NeRF-like representations are able to synthesize novel views with photorealistic visual quality. In particular, with a scalable model design, such as InstantNGP [36], NeRFs are able to model room-scale 3D scenes with extraordinary detail [53]. However, existing approaches typically only manage to model mild view-dependent effects like those seen on nearly diffuse surfaces. When encountering highly view-dependent glossy surfaces, NeRFs struggle to model the high-frequency changes when the viewpoint changes. Instead, they tend to “fake” specular reflections by placing them behind surfaces, which may result in poor view interpolation and “foggy” geometry [47]. Moreover, fake

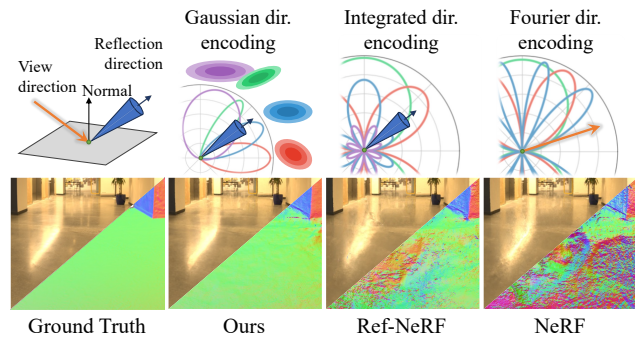


Figure 1. We propose a Gaussian directional encoding that leads to better modeling of specular reflections under near-field lighting conditions. In contrast, the integrated directional encoding utilized in Ref-NeRF [47] and Fourier directional encoding in NeRF [34] exhibit suboptimal performance under similar conditions.

reflections are not viable if one can look behind the surface, as NeRF can no longer hide the reflections there.

Accurately modeling and reconstructing specular reflections presents notable challenges, especially for room-scale scenes. Physically correct reflection modeling involves path-tracing many rays for every single pixel, which is impractical for NeRF-like volumetric scene representations, primarily due to the large computational requirements to shade a single pixel. Consequently, an efficient approximation of the reflection shading is needed for a feasible modeling of reflections. Existing works [14, 47] address this challenge by incorporating heuristic modules inspired by real-time image-based lighting (IBL) [35] techniques, such as explicit ray bounce computations to enhance NeRF’s capability to simulate reflections, and integrated directional encoding to simulate appearance change under varying surface roughness.

While these improvements have shown to be effective in modeling specular reflections for NeRFs, they are limited to object-level reconstruction under distant lighting, which assumes the object is lit by a 2D environment map. They work poorly for modeling near-field lighting, where the corresponding environment map varies spatially. The issue is that existing methods rely on directional encodings to embed ray directions for generating view-dependent reflections.

These encodings, such as Fourier encoding or spherical harmonics, are spatially invariant. Figure 1 demonstrates one example of NeRF [34] and Ref-NeRF [47] reconstructions of an indoor scene with spatially-varying lighting. NeRF produces extremely noisy geometry, resulting in artifacts in the rendering result. Ref-NeRF offers a slight improvement, but still struggles with noisy geometry and view interpolation. This illustrates that the spatial invariance in the directional encodings of existing methods presents challenges under spatially-varying lighting conditions.

In this work, we propose a novel Gaussian directional encoding that is tailored for spatially varying lighting conditions. Instead of only encoding a 2D ray direction, we use a set of learnable 3D Gaussians as the basis to embed a 5D ray space including both ray origin and ray direction. We show that, with appropriately optimized Gaussian parameters, this encoding introduces an important inductive bias towards near-field lighting, which enhances the model’s ability to capture the characteristics of specular surfaces, leading to photorealistic reconstructions of shiny reflections. We further demonstrate that by changing the scale of the 3D Gaussians, we can edit the apparent roughness of a surface.

While our proposed Gaussian directional encoding improves the reflection modeling of NeRF, high-quality reflection reconstruction also requires an accurate surface geometry and normal in order to compute accurate reflection rays. However, the geometry within NeRFs is often noisy in the early phases of training, which presents challenges in simultaneously optimizing for good geometry and reflections. To better address this challenge, we introduce a data-driven prior to direct the NeRF model towards the desired solution. We deploy a monocular normal estimation network to supervise the normal of the geometry at the beginning of the training stage, and show that this bootstrapping strategy improves the reconstruction of normals, and further leads to successful modeling of specular reflections. We conduct experiments on several public datasets and show that the proposed method outperforms existing methods, achieving higher-quality photorealistic rendering of reflective scenes while also providing more meaningful and accurate color component decomposition. Our contributions can be summarized as follows:

- We propose a novel Gaussian directional encoding that is more effective in modeling view-dependent effects under near-field lighting conditions.
- We propose to use monocular normal estimation to resolve shape-radiance ambiguity in the early training stages.
- Our full NeRF pipeline achieves state-of-the-art novel-view synthesis performance for specular reflections.

2. Related Work

Reflection-aware NeRFs. Successfully modeling view-dependent effects, such as specular reflections, can greatly

enhance the photorealism of the reconstructed NeRF. NeRF models view-dependency by conditioning the radiance on the positional encoding [43] of the input ray direction, which is only capable of mild view-dependent effects. Ref-NeRF [47] improves NeRF’s capability for modeling reflections by conditioning the view-dependent appearance on the reflection ray direction instead of incident ray direction, and by modulating the directional encoding based on surface roughness. This reparameterization of outgoing radiance makes the underlying scene function simpler, leading to a better geometry and view interpolation quality for glossy objects. Ref-NeuS [14] further extends these concepts to a surface-based representation. However, these are primarily designed for object-level reconstruction under environment map lighting conditions. Modeling large-scale scenes with near-field lighting remains a problem. Clean-NeRF [30] decomposes the radiance into diffuse and specular colors, and supervises the two components by least-square estimations of multiple input rays. This alleviates the ambiguity of highly specular regions; yet, it does not change the view-dependent structure of the NeRF model, thus limiting its ability to model reflections. NeRF-DS [54] models specularities in dynamic scenes and considers the variations in reflections caused by dynamic geometry through the use of a dynamic normal field, but requires additional object masks for accurate specular reconstruction.

NeRF-based Inverse Rendering. Inverse rendering goes beyond simple reflection modeling and aims to jointly recover one or more of scene geometry, material appearance and the lighting condition. In practice, the material appearance is typically modeled using physically-based rendering assets such as albedo, roughness and glossiness. Mesh-based inverse rendering methods [2, 38, 49, 68] try to recover materials using differentiable path tracing [25]. However, they typically assume a given geometry, since optimizing mesh geometry is challenging. On the contrary, NeRF-based inverse rendering approaches [5, 42, 65, 66] make it easier to optimize geometry jointly by modeling material properties and density continuously in a volumetric 3D space. The lighting is usually represented as point or directional lights [5, 23, 60], an environment texture map [31–33, 42, 65], or an implicit texture map modeled by spherical Gaussians [6, 11, 18, 63, 66, 66, 67] or MLPs [7, 29]. Most methods are limited to object-level reconstruction and assume the lighting is spatially invariant (i.e. distant). Several light estimation techniques [13, 26, 27, 41] explore using 3D light primitives or spatially-varying spherical Gaussians to model spatially varying lighting. However, these methods focus on data-driven approaches to estimate lighting for image editing. NeILF [56] and NeILF++ [62] model lighting as a 5D light field using another MLP, but still focus mainly on small-scale reconstruction. Several works apply inverse rendering for relighting outdoor scenes [24, 40, 48, 58]. However, they fo-

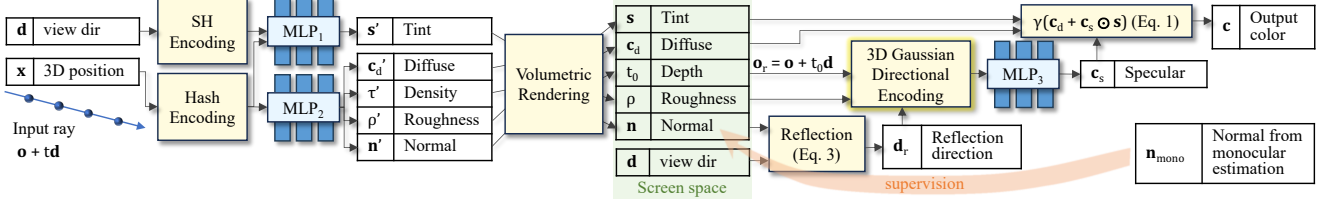


Figure 2. An overview of our model. The key enabler for specular reflections is our novel 3D Gaussian directional encoding module that converts the reflected ray into a spatially-varying embedding, which is further decoded into specular color.

cus more on diffuse materials with correct shadow modeling instead of reflections. In this work, we have a different goal compared to inverse rendering, focusing only on correctly modeling reflections for better novel-view synthesis, rather than trying to discern material properties for standalone use.

NeRF with mirror reflections. One special case of reflection is mirror reflection. One approach represents the reflected scene as a separate NeRF [15], and composites the two NeRF results in image space. This is also deployed in image-based rendering [52] and large-scale NeRF reconstruction [50]. Given a multi-mirror scene, the idea can be further extended to multi-space NeRFs [57]. An alternate approach is to explicitly model the mirror geometry, and to render the mirrored scene by path tracing [16, 61]. However, since estimating the mirror geometry is highly ill-posed, manual annotation is usually needed. Curved reflectors need even more careful handling [22, 46].

3. Preliminaries

We first review Ref-NeRF [47] for decomposing view-dependent appearance. Similar to NeRF, Ref-NeRF models the scene as a function that maps the position \mathbf{x} and view direction \mathbf{d} to the final color \mathbf{c} and density τ . The difference is that Ref-NeRF predicts the color as a combination of diffuse color \mathbf{c}_d and specular color \mathbf{c}_s :

$$\mathbf{c} = \gamma(\mathbf{c}_d + \mathbf{c}_s \odot \mathbf{s}), \quad (1)$$

where \mathbf{s} is the specular tint, ‘ \odot ’ the element-wise product, and $\gamma(\cdot)$ a tone-mapping function. To predict the specular color \mathbf{c}_s , Ref-NeRF first predicts the surface normal \mathbf{n} , roughness ρ , and features φ at location \mathbf{x} using an MLP. Then, the specular color \mathbf{c}_s is parameterized as a function of the reflection direction \mathbf{d}_r :

$$\mathbf{c}_s = F_\theta(\lambda_{\text{IDE}}(\mathbf{d}_r, \rho), \varphi), \quad (2)$$

where $\lambda_{\text{IDE}}(\cdot)$ is the integrated directional encoding introduced by Ref-NeRF, $F_\theta(\cdot)$ represents an MLP with parameters θ , and the reflection direction \mathbf{d}_r is the input direction \mathbf{d} reflected at the predicted surface normal \mathbf{n} :

$$\mathbf{d}_r = \mathbf{d} - 2(\mathbf{d} \cdot \mathbf{n})\mathbf{n}. \quad (3)$$

By conditioning the specular color on reflection direction and roughness, the function F_θ needs to fit is much simpler.

4. Method

Our goal is to enhance NeRF’s capabilities for modeling specular reflections under near-field lighting conditions. Figure 2 presents an overview of our pipeline. A key contribution is the 3D Gaussian directional encoding that maps a ray and surface roughness to a ray embedding.

To render a pixel, we sample points along an input ray $\mathbf{o} + t\mathbf{d}$, and predict volume density τ' , diffuse color \mathbf{c}'_d , tint \mathbf{s}' , roughness ρ' , and normal direction \mathbf{n}' at each sample point (we denote per-sample properties using a prime). Given that reflections occur only at the surface, we evaluate the specular component once per ray on the surface obtained from the NeRF depth. This also results in less computation than per-sample-point specular shading. Consequently, we calculate volumetric depth t_0 by rendering the ray marching distance at each sample point. We also volumetrically render all attributes to synthesize screen-space attributes $(\mathbf{c}_d, \mathbf{s}, \rho, \mathbf{n})$. Note that the rendered normal must be normalized to yield the final screen-space normal \mathbf{n} . We then evaluate the specular component by first computing the reflected ray using origin $\mathbf{o}_r = \mathbf{o} + t_0\mathbf{d}$, and the reflection direction \mathbf{d}_r derived using Equation 3. The reflected ray $\mathbf{o}_r + t\mathbf{d}_r$ and surface roughness ρ are then encoded using our novel 3D Gaussian directional encoding. After a tiny MLP, we compute the specular color \mathbf{c}_s , and the final rendering result using Equation 1.

From a physically based rendering perspective, Equation 1 is analogous to the Cook–Torrance approximation [10] of the rendering equation [19]. The term $\mathbf{c}_s \odot \mathbf{s}$ can be interpreted as the split-sum approximation of the specular part of the Cook–Torrance model, with the specular color \mathbf{c}_s corresponding to the preconvolved incident light, and the tint \mathbf{s} to the pre-integrated bidirectional reflectance distribution function (BRDF).

4.1. Gaussian Directional Encoding

Existing works parameterize view-dependent appearance by first encoding view or reflection direction into Fourier or spherical harmonics (SH) features, which results in a spatially invariant encoding of the view direction. Therefore, it becomes challenging for the NeRF to model spatially varying view-dependent effects, such as near-field lighting. We illustrate this via a toy example in Figure 3, where we place

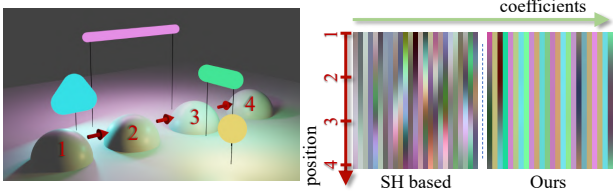


Figure 3. **Toy example of 3D Gaussian encoding.** **Left:** A hemisphere probe translates underneath 4 lights along positions numbered 1 to 4. Note that we dilate the lights for better visualization. **Right:** Representation of the probe’s specular components using spherical harmonics and our 3D Gaussian directional encoding. The SH encoding shows a more complex pattern under position change, while ours has spatially largely invariant coefficients. This suggests a simpler function for the specular prediction MLP to fit using Gaussian directional encoding.

a hemispherical specular probe in a simple scene with four lights of different shapes and colors. Then, we represent the specular component of the toy example by linearly combining the directional encoding features. We find the optimal coefficients for each encoding type that best fit the ground-truth specular component using stochastic gradient descent, and visualize them in Figure 3. We can see that even for this simple toy setup, the SH-based encoding requires complex, spatially varying coefficients, which complicates the underlying function for the NeRF to fit and interpolate.

We propose to spatially vary the encoding function by defining the basis functions via several learnable 3D Gaussians. Specifically, we parameterize 3D Gaussians using their position $\mu_i \in \mathbb{R}^3$, scale $\sigma_i \in \mathbb{R}^3$, and quaternion rotation $\mathbf{q}_i \in \mathbb{H}$:

$$\mathcal{G}_i(\mathbf{x}) = \exp\left(-\|\mathcal{Q}(\mathbf{x} - \mu_i; \mathbf{q}_i) \odot \sigma_i^{-1}\|_2^2\right), \quad (4)$$

where $\mathcal{Q}(\mathbf{v}; \mathbf{q}_i)$ represents applying quaternion rotation \mathbf{q}_i to the vector \mathbf{v} . To compute the i -th dimension of the encoding for a ray $\mathbf{o} + t\mathbf{d}$, we need to define a basis function $\mathcal{P}_i(\mathbf{o}, \mathbf{d}) \in \mathbb{R}$ that maps the ray to a scalar value given the Gaussian parameters. While there are many ways to define the mapping, we find one that is efficient and has a closed-form solution by defining the projection as the maximum value of the Gaussian along the ray:

$$\mathcal{P}_i(\mathbf{o}, \mathbf{d}) = \max_{t \geq 0} \mathcal{G}_i(\mathbf{o} + t\mathbf{d}). \quad (5)$$

In the supplement, we derive the closed-form solution:

$$\mathcal{P}_i(\mathbf{o}, \mathbf{d}) = \begin{cases} \exp\left(\frac{(\mathbf{o}_i^\top \mathbf{d}_i)^2}{\mathbf{d}_i^\top \mathbf{d}_i} - \mathbf{o}_i^\top \mathbf{o}_i\right) & \mathbf{o}_i^\top \mathbf{d}_i < 0 \\ \mathcal{G}_i(\mathbf{o}) & \text{otherwise,} \end{cases} \quad (6)$$

where \mathbf{o}_i and \mathbf{d}_i are the ray origin and direction transformed into Gaussian-local space:

$$\mathbf{o}_i = \mathcal{Q}(\mathbf{o} - \mu_i; \mathbf{q}_i) \odot \sigma_i^{-1}, \quad (7)$$

$$\mathbf{d}_i = \mathcal{Q}(\mathbf{d}; \mathbf{q}_i) \odot \sigma_i^{-1}. \quad (8)$$

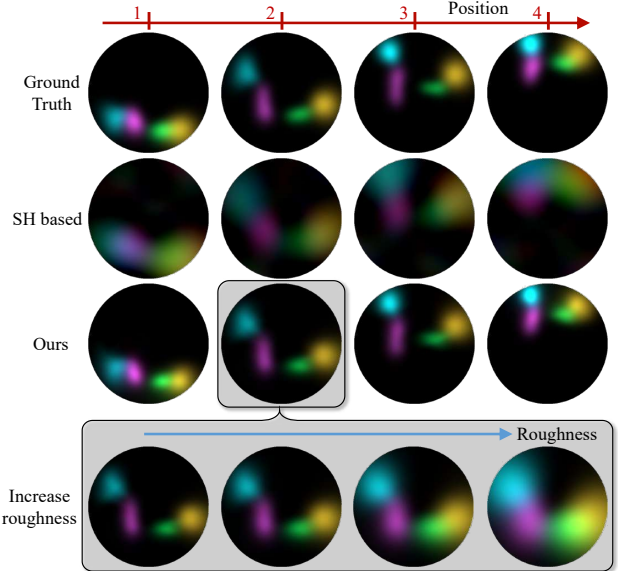


Figure 4. Stereographic projections of the specular fitting results for the toy example in Figure 3. Both encodings produce 25 coefficients for each color channel, which are then summed to produce the final color. Note that the GT shows soft boundaries because it is preconvolved. The 3D Gaussian-based encoding demonstrates superior performance in representing the specular change with positional changes, and is also capable of smoothly varying roughness.

By applying Equation 6 for every 3D Gaussian, we obtain a vector of projected values $\{\mathcal{P}_i\}$, which forms our final encoding features. Similar to existing NeRF-based representations [9, 34, 36], we rely on a small MLP to convert the encoding to a specular color \mathbf{c}_s .

As illustrated by the toy example in Figure 3, our Gaussian directional encoding exhibits more constant coefficients in response to the position changes, suggesting a smoother mapping from the embedding features to the specular color. This smoothness is due to the Gaussian basis function producing spatially varying features that mimic the behavior of how the specular component would change under near-field lighting conditions. As a result, the underlying functions that model the specular reflections are easier to learn.

We also visualize the fitted specular color of both approaches in Figure 4. Our 3D Gaussian directional encoding more accurately captures the spatial variations of the specular components.

Similar to Ref-NeRF, we use an additional “roughness” value ρ to control the maximum frequency of the specular color. We achieve this in our Gaussian embedding by multiplying each Gaussian’s scale σ_i with the roughness ρ . Intuitively, a larger Gaussian results in a smoother function with varying direction \mathbf{d} . Substituting the σ_i with $\rho\sigma_i$ in Equation 6 leads to the complete equation of our 3D Gaussian encoding. Figure 4 demonstrates the ability of our 3D Gaussian-based encoding to modify roughness on the fly.

4.2. Optimizing the Gaussian Directional Encoding

It is worth noting that our proposed Gaussian encoding correctly models spatially varying specular reflections only when the Gaussians are positioned properly in 3D space. We thus jointly optimize the Gaussian parameters together with the NeRF during training, to ensure the Gaussians are in the optimal state for modeling reflections. However, there is no direct supervision for the Gaussian parameters.

Our experiments show that without proper initial Gaussian parameters, the optimization may lead to suboptimal local minima, resulting in inconsistent quality of specular reconstruction. To address this, we propose an initialization stage for the Gaussian parameters and to bootstrap the specular color prediction. As mentioned earlier, the specular color is essentially the preconvolved incident light, which can be directly deduced from input images.

Motivated by this observation, we train the 3D Gaussians and the specular decoder (MLP₃ in Figure 2) in the initialization stage using the input images. We train an incident light field that accommodates a diversity of rays and roughness values. Therefore, we apply a range of Gaussian blurs to all input images using a series of standard deviations, generating Gaussian pyramids. These pyramids of input images provide a pseudo target for incident light under different degrees of surface roughness. In each iteration of the training, we sample pixels from the pyramids and trace rays to these pixels. The traced rays are also associated with a roughness value that is equivalent to the blur’s standard deviation. We encode each ray with roughness using our Gaussian directional encoding, and predict the specular color c_s using the decoder. By minimizing the errors between c_s and the pseudo ground truth, we refine the Gaussian parameters and the specular decoder, which then serve as initialization for the subsequent joint optimization stage.

4.3. Resolving the Shape–Radiance Ambiguity

Regardless of any view-dependent parameterization, there remains a fundamental ambiguity between shape and radiance in NeRFs. For example, consider a perfect mirror reflection. Without any prior knowledge, it is nearly impossible for the NeRF model to tell whether the mirror is a flat surface with perfect reflection, or a window to a (virtual) scene behind the surface. Therefore, prior information is needed to guide the model to learn the correct geometry. Inspired by recent progress in monocular geometry estimation [4, 12, 39, 59], we propose to supervise the predicted normal \mathbf{n} using monocular normal estimation \mathbf{n}_{mono} [12]:

$$\mathcal{L}_{\text{mono}} = \sum_j \|\mathbf{n}^j - \mathbf{R}^j \mathbf{n}_{\text{mono}}^j\|_2^2, \quad (9)$$

where the superscript j is a ray index, and \mathbf{R}^j is the corresponding camera rotation matrix that converts normals from view space to world space.

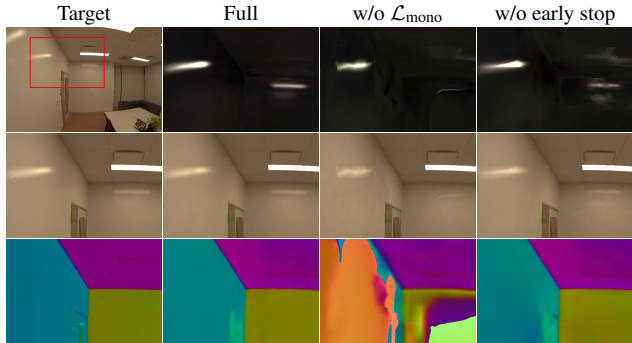


Figure 5. The specular component reconstruction (first row, except the first image), novel-view synthesis results (second row) and normal visualizations (third row) under varying monocular normal supervision. The target normal visualizes the monocular normal prediction. Without $\mathcal{L}_{\text{mono}}$, the predicted normal exhibits enormous error, leading to poor specular reconstruction. Without early stopping $\mathcal{L}_{\text{mono}}$, minor errors in the predicted normals lead to a slight degradation in the reflection quality compared to our full model.

However, monocular normals are prone to error. We therefore use them primarily as initialization and apply $\mathcal{L}_{\text{mono}}$ only at the beginning of the training, so that the errors in the normals do not overwhelm the geometry of the NeRF. Figure 5 and Table 2 show results with different configurations of $\mathcal{L}_{\text{mono}}$. We can see that without monocular normal as supervision (‘w/o $\mathcal{L}_{\text{mono}}$ ’), the predicted normals have catastrophic errors, such as those pointing inwards (orange) or lying parallel (violet) to the surface. Consequently, the learned specular component is less accurate due to the incorrect normals. Despite this, a somewhat plausible specular reflection can still be learned as the Gaussian encoding can “cheat” the reflections even with erroneous normals. On the other hand, without early stopping of the loss (‘w/o early stop’), minor inaccuracies from the monocular normals permeate into predicted normals, leading to a degradation of the reflection quality.

4.4. Losses

To jointly optimize all parameters within our proposed pipeline, we use a combination of loss terms:

$$\mathcal{L} = \mathcal{L}_c + \mathcal{L}_{\text{prop}} + \lambda_{\text{dist}} \mathcal{L}_{\text{dist}} + \lambda_{\text{mono}} \mathcal{L}_{\text{mono}} + \lambda_{\text{norm}} \mathcal{L}_{\text{norm}}. \quad (10)$$

In this equation, \mathcal{L}_c is the L1 reconstruction loss between the predicted and ground-truth colors. The terms $\mathcal{L}_{\text{prop}}$ and $\mathcal{L}_{\text{dist}}$ are adopted from mip-NeRF 360 [3], where $\mathcal{L}_{\text{prop}}$ supervises the density proposal networks, and $\mathcal{L}_{\text{dist}}$ is the distortion loss encouraging density sparsity. To tie predicted normals to the density field, we use Ref-NeRF’s normal prediction loss $\mathcal{L}_{\text{norm}}$ [47], which guides the predicted normal \mathbf{n} with the density gradient direction. Further elaboration on these loss components can be found in the supplementary material.

In our experiments, we set $\lambda_{\text{dist}} = 0.002$, aligning with the settings of the “nerfacto” model in Nerfstudio [44]. For

Table 1. Quantitative comparisons of novel-view synthesis on three datasets. We highlight the best numbers in **bold**.

Methods	Eyeful Tower dataset [53]			NISR [50] + Inria [38] dataset			Shiny dataset [47]		
	PSNR \uparrow	SSIM \uparrow	LPIPS \downarrow	PSNR \uparrow	SSIM \uparrow	LPIPS \downarrow	PSNR \uparrow	SSIM \uparrow	LPIPS \downarrow
Ours	32.583	0.9328	0.1445	30.771	0.8909	0.1655	26.564	0.7277	0.2776
NeRF [34]	31.854	0.9254	0.1626	30.748	0.8873	0.1728	26.469	0.7235	0.2852
Ref-NeRF [47]	31.652	0.9258	0.1570	30.654	0.8903	0.1669	26.502	0.7242	0.2827
MS-NeRF [57]	31.715	0.9311	0.1561	30.224	0.8840	0.1816	26.466	0.7070	0.3225

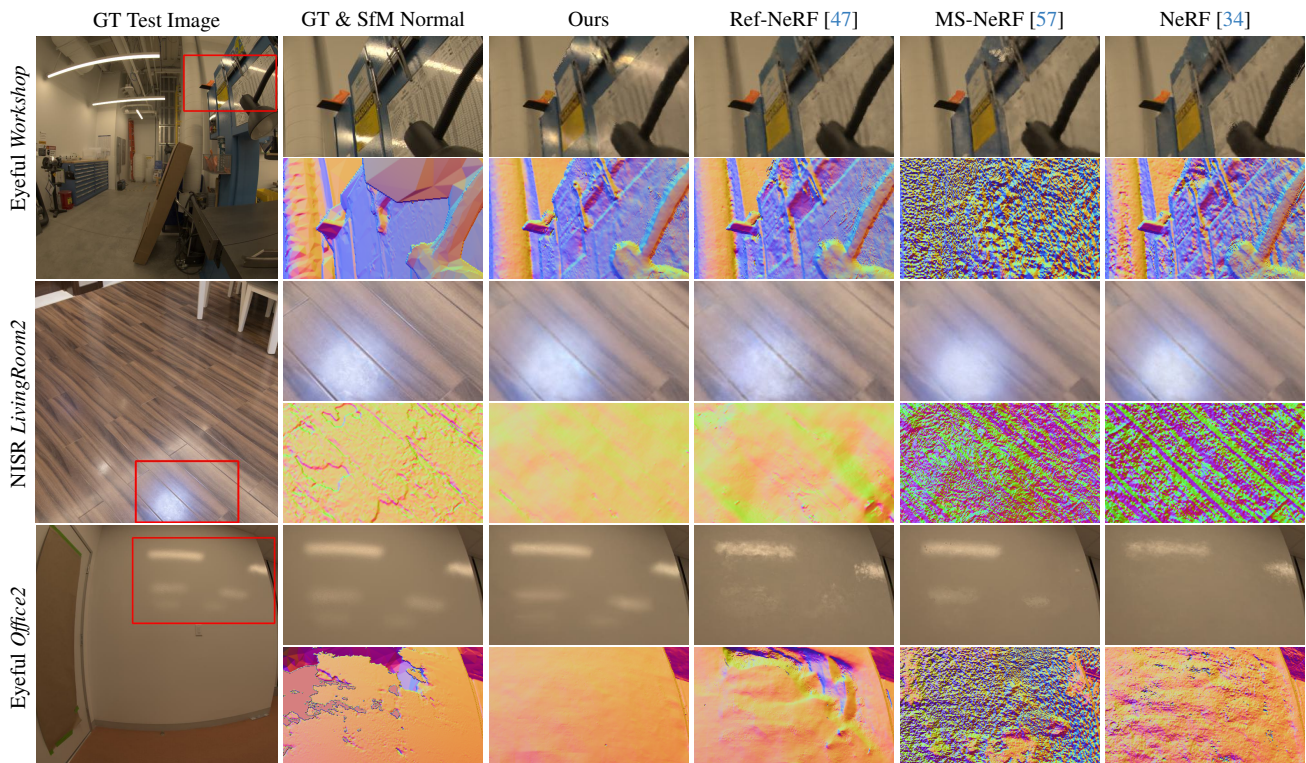


Figure 6. Comparisons of novel-view synthesis quality and normal map visualizations. Our method consistently reconstructs reflections while other methods either produce ‘faked’ reflections, resulting in incorrect normals, or fail to model reflections entirely.

λ_{mono} , we choose a value of 1 in the first 4K iterations, and reduce to 0 thereafter to cease its effect, as described earlier. We find that our method is robust to the value of λ_{mono} as it only serves as initialization. We also assign $\lambda_{\text{norm}} = 10^{-3}$, which is slightly higher than the weight in Ref-NeRF [47], as we find that this produces slightly smoother normals without substantially compromising the rendering quality.

5. Experiments

Implementation. To model room-scale scenes, we employ a network architecture similar to the “nerfacto” model presented in Nerfstudio [44]. We use two small density networks as proposal networks, supervised via $\mathcal{L}_{\text{prop}}$. We sample 256 and 96 points for each proposal network, and 48 points for the final NeRF model. These three networks all use hash-based positional encodings. When querying the hash features in the final NeRF model, we incorporate the LOD-aware scheme proposed in VR-NeRF [53]. We train our model

for 100,000 iterations and randomly sample 12,800 rays in each iteration. This process takes around 8 GB of GPU memory and approximately 3.5 hours to train a model using an NVIDIA A100 GPU. Further details regarding the model’s structure can be found in the supplementary materials.

Datasets. We evaluate our method on several datasets with a focus on indoor scenes characterized by near-field lighting conditions. First, we evaluate on the Eyeful Tower dataset [53], which provides high-quality HDR captures of 11 indoor scenes. Each scene is coupled with calibrated camera parameters and a mesh reconstructed via Agisoft Metashape [1]. We select 9 scenes that feature notable reflective properties. We downsample the images of each scene to a resolution of 854×1280 pixels. We curated around 50–70 views per scene that contain glossy surfaces for evaluation, leaving the remaining views for training. We also evaluate our approach on public indoor datasets NISR [50] and Inria [38] (NISR+Inria). Moreover, to assess the performance under

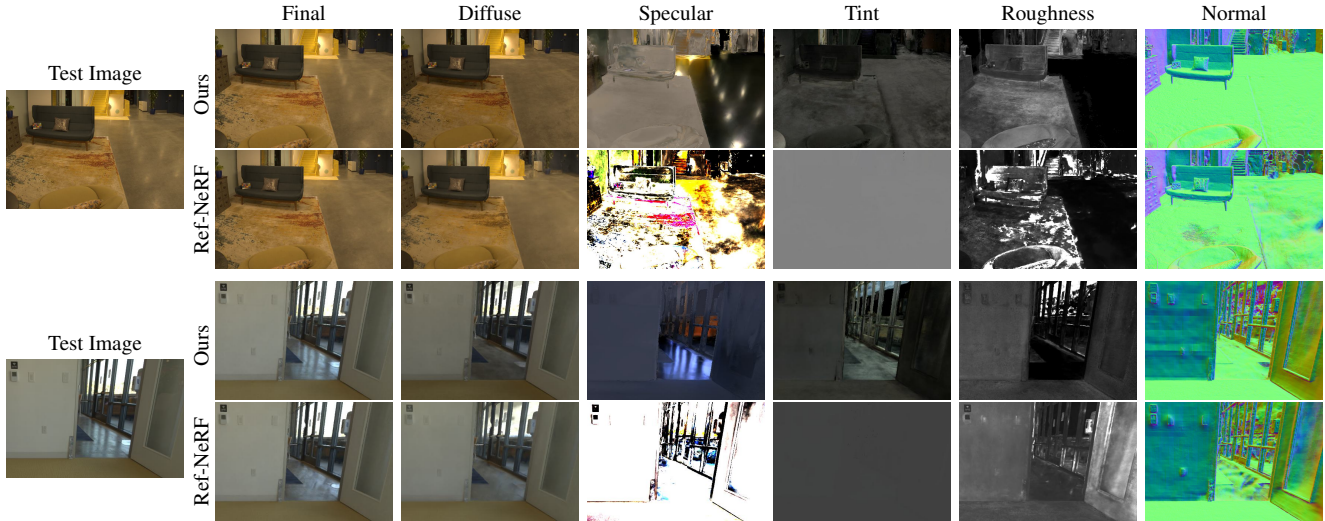


Figure 7. Intermediate components of our approach compared to Ref-NeRF [47]. Our approach produces a more meaningful decomposition under room-scale lighting settings.

far-field lighting, we evaluate the real shiny dataset in Ref-NeRF [47]. We report the average PSNR, SSIM, and LPIPS [64] metrics for evaluating rendering quality.

5.1. Comparisons

We compare our method with several baselines: NeRF [34], Ref-NeRF [47], and MS-NeRF [57], which specializes in mirror-like reflections by decomposing NeRF into multiple spaces. For a fair comparison, we re-implement these baselines, such that we share the same NeRF backbone and rendering configurations, with the only difference being the way different methods decompose and parameterize the output color. We report the numerical results across three datasets in Table 1. Our method demonstrates superior performance on the Eyeful Tower dataset, indicating the effectiveness of our method. On the NISR+Inria datasets, our method marginally outperforms the baselines, likely due to the dataset containing few reflection surfaces. Notably, while our method is tailored for near-field lighting conditions, it also shows promising results on the Shiny dataset, which comprises far-field lighting scenarios. This is because our Gaussian directional encoding can simulate a spatially invariant encoding by positioning Gaussians at a significant distance.

Qualitative results on the Eyeful Tower and NISR+Inria datasets are provided in Figure 6. We can see that while other baselines occasionally synthesize plausible reflections, they resort to approximations that fake the reflections by placing emitters underneath the surface. As a result, they either produce incorrect geometry, or fail to model the reflections. Our method, in contrast, successfully models specular highlights on the surface. We provide additional video results in the supplementary material.

We also visualize and compare the decomposition produced by our method and Ref-NeRF in Figure 7. We can see

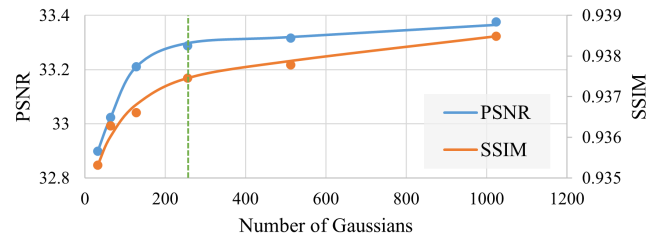


Figure 8. We evaluate the novel-view synthesis quality with respect to the number of Gaussians across five scenes. The green dashed line is the setting we use in our experiments.

that Ref-NeRF fails to obtain a meaningful decomposition under near-field lighting, and produces holes in the geometry, whereas our method consistently achieves a realistic separation of specular and diffuse components.

5.2. Ablation Studies

Number of Gaussians. One important hyperparameter in the Gaussian directional encoding is the number of Gaussians, as it directly influences the model’s capacity to represent specular colors. We conduct experiments to evaluate the impact of varying the number of Gaussians on five scenes from the Eyeful Tower dataset, and show the relationship between the number of Gaussians and the rendering quality in Figure 8. The rendering quality improves when using more Gaussians, but the improvement saturates as the number increases beyond 400. Note that using a larger number of Gaussians also entails greater computation costs and GPU memory requirements for every rendered pixel. Therefore, we use 256 Gaussians for all experiments, to strike a balance between rendering quality and computational efficiency.

Optimizing Gaussians. To optimize the Gaussian directional encoding effectively, we first initialize them by training an incident light field, and then jointly finetune the Gaus-

Table 2. Ablations of our method on the Eyeful Tower dataset [53]. The “e.s.” indicates early stopping the $\mathcal{L}_{\text{mono}}$ after 4K iterations.

Method	Gaussians		Mono.	E. S.	PSNR \uparrow	SSIM \uparrow	LPIPS \downarrow
	Init.	Opt.	Prior	$\mathcal{L}_{\text{mono}}$			
Full	✓	✓	✓	✓	32.58	0.9328	0.1445
w/o init	✗	✓	✓	✓	32.52	0.9304	0.1429
w/o opt.	✓	✗	✓	✓	32.06	0.9265	0.1581
w/o $\mathcal{L}_{\text{mono}}$	✓	✓	✗	—	32.31	0.9288	0.1503
w/o e.s.	✓	✓	✓	✗	32.46	0.9292	0.1502

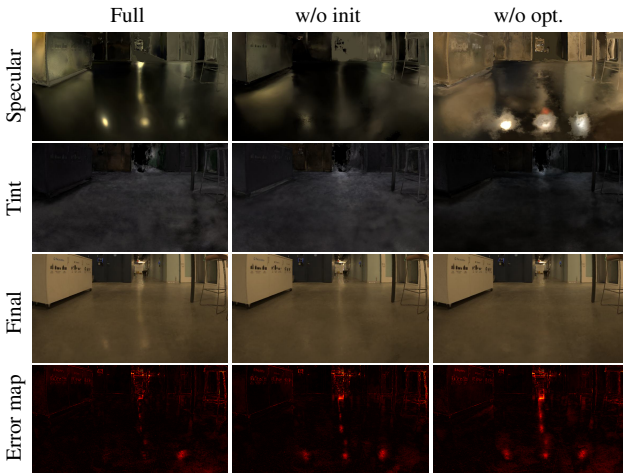


Figure 9. Example results under different Gaussian optimization settings. Without initializing the Gaussian parameters (‘w/o init’) or optimizing Gaussians jointly with the NeRF (‘w/o opt.’), the Gaussian embedding struggles to model specularities accurately.

sian encoding together with the NeRF model. We demonstrate the significance of initialization (‘w/o init’) and fine-tuning (‘w/o opt.’) by omitting each process individually. We show quantitative results in Table 2 and a qualitative example in Figure 9. Without initialization, the model can still reconstruct reflections to some extent, resulting in a slightly better average LPIPS score, yet it fails to model some specular details, such as the light blobs. Neglecting the joint optimization of Gaussians leads to complete failure in modeling specular reflections. As illustrated in Figure 9, with the inaccurate specular modeling, the tints suppress the specular reflections, which ultimately leads to the inability to represent reflections in the final rendered image.

6. Discussion and Conclusion

Applications. Our primary goal was to improve the quality of novel-view synthesis with specular reflective surfaces. We achieve this via our proposed Gaussian directional encoding that enables a meaningful decomposition of specular and diffuse components in a scene. Moreover, this also enables applications other than novel-view synthesis, such as reflection removal, and surface roughness editing. For instance, Figure 7 shows that we can easily remove reflections using the diffuse component. Furthermore, Figure 10 demonstrates

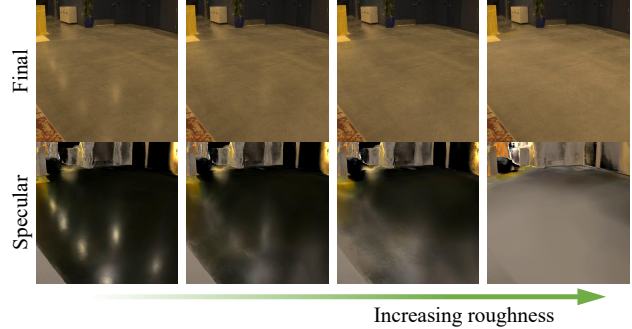


Figure 10. We can control the roughness of the scene by adding an offset to the input roughness.



Figure 11. Our method cannot reconstruct mirror-like perfect reflections due to the limited capacity of the 3D Gaussian encoding.

an example of editing roughness. By adding an offset to the predicted roughness during rendering, we can effectively manipulate the glossiness of the real surface.

Limitations. While our method improves on existing baselines, it has some limitations. As we parameterize the specular color via only several hundreds of Gaussians, the encoding is limited to relatively low frequency compared with perfect mirror-like reflections. We show such a failure case in Figure 11. We can see that our method is only able to learn a blurry version of the reflection. This could be alleviated by using many more Gaussians, as demonstrated in 3D Gaussian splatting [20]. However, the computational cost of traversing all Gaussians for every pixel quickly becomes prohibitive in our implementation. More efficient traversal, such as by rasterization, could be interesting future work.

Conclusion. In this paper, we proposed a pipeline to improve the existing approach in modeling and reconstructing view-dependent effects in a NeRF representation. Central to our approach is a new Gaussian directional encoding to enhance the capability of neural radiance fields to model specular reflections under near-field lighting. We also utilize monocular normal supervision to help resolve shape–radiance ambiguity. Experiments have demonstrated the effectiveness of each of our contributions. We believe this work proposes a practical and effective solution for reconstructing NeRFs in room-scale scenes, specifically addressing the challenges of accurately capturing specular reflections.

Acknowledgments. The authors from HKUST were partly supported by the Hong Kong Research Grants Council (RGC). We would like to thank Linning Xu and Zhao Dong for helpful discussions.

References

- [1] Agisoft LLC. Agisoft Metashape Professional. Computer software, 2022. 6
- [2] Dejan Azinović, Tzu-Mao Li, Anton Kaplanyan, and Matthias Nießner. Inverse path tracing for joint material and lighting estimation. In *CVPR*, 2019. 2
- [3] Jonathan T. Barron, Ben Mildenhall, Dor Verbin, Pratul P. Srinivasan, and Peter Hedman. Mip-NeRF 360: Unbounded anti-aliased neural radiance fields. In *CVPR*, 2022. 5, 4
- [4] Shariq Farooq Bhat, Reiner Birkl, Diana Wofk, Peter Wonka, and Matthias Müller. ZoeDepth: Zero-shot transfer by combining relative and metric depth. [arXiv:2302.12288](https://arxiv.org/abs/2302.12288), 2023. 5
- [5] 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:2008.03824](https://arxiv.org/abs/2008.03824), 2020. 2
- [6] Mark Boss, Raphael Braun, Varun Jampani, Jonathan T. Barron, Ce Liu, and Hendrik P.A. Lensch. NeRD: Neural reflectance decomposition from image collections. In *ICCV*, 2021. 2
- [7] Mark Boss, Varun Jampani, Raphael Braun, Ce Liu, Jonathan T. Barron, and Hendrik P.A. Lensch. Neural-PIL: Neural pre-integrated lighting for reflectance decomposition. In *NeurIPS*, 2021. 2
- [8] G. Bradski. The OpenCV Library. *Dr. Dobb's Journal of Software Tools*, 2000. 3
- [9] Anpei Chen, Zexiang Xu, Andreas Geiger, Jingyi Yu, and Hao Su. TensoRF: Tensorial radiance fields. In *ECCV*, 2022. 4
- [10] Robert L Cook and Kenneth E Torrance. A reflectance model for computer graphics. *ACM Trans. Graph.*, 1(1):7–24, 1982. 3
- [11] Youming Deng, Xueting Li, Sifei Liu, and Ming-Hsuan Yang. DIP: Differentiable interreflection-aware physics-based inverse rendering. In *3DV*, 2024. 2
- [12] Ainaz Eftekhari, Alexander Sax, Jitendra Malik, and Amir Zamir. Omnidata: A scalable pipeline for making multi-task mid-level vision datasets from 3D scans. In *ICCV*, 2021. 5
- [13] Marc-Andre Gardner, Yannick Hold-Geoffroy, Kalyan Sunkavalli, Christian Gagne, and Jean-Francois Lalonde. Deep parametric indoor lighting estimation. In *ICCV*, 2019. 2
- [14] Wenhao Ge, Tao Hu, Haoyu Zhao, Shu Liu, and Ying-Cong Chen. Ref-NeuS: Ambiguity-reduced neural implicit surface learning for multi-view reconstruction with reflection. In *ICCV*, 2023. 1, 2
- [15] Yuan-Chen Guo, Di Kang, Linchao Bao, Yu He, and Song-Hai Zhang. NeRFReN: Neural radiance fields with reflections. In *CVPR*, 2022. 3
- [16] Leif Van Holland, Ruben Bliersbach, Jan U. Müller, Patrick Stotko, and Reinhard Klein. TraM-NeRF: Tracing mirror and near-perfect specular reflections through neural radiance fields. [arXiv:2310.10650](https://arxiv.org/abs/2310.10650), 2023. 3
- [17] Yoonwoo Jeong, Seungjoo Shin, and Kibaek Park. NeRF-Factory: An awesome PyTorch NeRF collection, 2024. 5
- [18] Haiyan Jin, Isabella Liu, Peijia Xu, Xiaoshuai Zhang, Songfang Han, Sai Bi, Xiaowei Zhou, Zexiang Xu, and Hao Su. TensoIR: Tensorial inverse rendering. In *CVPR*, 2023. 2
- [19] James T. Kajiya. The rendering equation. *Computer Graphics (Proceedings of SIGGRAPH)*, 20(4):143–150, 1986. 3
- [20] Bernhard Kerbl, Georgios Kopanas, Thomas Leimkühler, and George Drettakis. 3D Gaussian splatting for real-time radiance field rendering. *ACM Trans. Graph.*, 42(4):139:1–14, 2023. 8
- [21] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *ICLR*, 2015. 1, 3
- [22] Georgios Kopanas, Thomas Leimkühler, Gilles Rainer, Clément Jambon, and George Drettakis. Neural point catcaustics for novel-view synthesis of reflections. *ACM Trans. Graph.*, 41(6):201:1–15, 2022. 3
- [23] Junxuan Li and Hongdong Li. Neural reflectance for shape recovery with shadow handling. In *CVPR*, 2022. 2
- [24] Quewei Li, Jie Guo, Yang Fei, Feichao Li, and Yanwen Guo. NeuLighting: Neural lighting for free viewpoint outdoor scene relighting with unconstrained photo collections. In *SIGGRAPH Asia*, pages 13:1–9, 2022. 2
- [25] Tzu-Mao Li, Miika Aittala, Frédo Durand, and Jaakko Lehtinen. Differentiable Monte Carlo ray tracing through edge sampling. *ACM Trans. Graph.*, 37(6):222:1–11, 2018. 2
- [26] Zhengqin Li, Mohammad Shafiei, Ravi Ramamoorthi, Kalyan Sunkavalli, and Manmohan Chandraker. Inverse rendering for complex indoor scenes: Shape, spatially-varying lighting and SVBRDF from a single image. In *CVPR*, 2020. 2
- [27] Zhengqin Li, Jia Shi, Sai Bi, Rui Zhu, Kalyan Sunkavalli, Miloš Hašan, Zexiang Xu, Ravi Ramamoorthi, and Manmohan Chandraker. Physically-based editing of indoor scene lighting from a single image. In *ECCV*, 2022. 2
- [28] Zhaoshuo Li, Thomas Müller, Alex Evans, Russell H Taylor, Mathias Unberath, Ming-Yu Liu, and Chen-Hsuan Lin. Neuralangelo: High-fidelity neural surface reconstruction. In *CVPR*, 2023. 4
- [29] Ruofan Liang, Jiahao Zhang, Haoda Li, Chen Yang, Yushi Guan, and Nandita Vijaykumar. SPIDR: SDF-based neural point fields for illumination and deformation. [arXiv:2210.08398](https://arxiv.org/abs/2210.08398), 2022. 2
- [30] Xinhang Liu, Yu-Wing Tai, and Chi-Keung Tang. Clean-NeRF: Reformulating NeRF to account for view-dependent observations. [arXiv:2303.14707](https://arxiv.org/abs/2303.14707), 2023. 2
- [31] Yuan Liu, Peng Wang, Cheng Lin, Xiaoxiao Long, Jiepeng Wang, Lingjie Liu, Taku Komura, and Wenping Wang. NeRO: Neural geometry and BRDF reconstruction of reflective objects from multiview images. *ACM Trans. Graph.*, pages 114:1–22, 2023. 2
- [32] Linjie Lyu, Ayush Tewari, Thomas Leimkuehler, Marc Habermann, and Christian Theobalt. Neural radiance transfer fields for relightable novel-view synthesis with global illumination. In *ECCV*, 2022.
- [33] Alexander Mai, Dor Verbin, Falko Kuester, and Sara Fridovich-Keil. Neural microfacet fields for inverse rendering. In *ICCV*, 2023. 2
- [34] 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. In *ECCV*, 2020. 1, 2, 4, 6, 7, 5
- [35] Gene S Miller and CR Hoffman. Illumination and reflection maps. In *ACM SIGGRAPH*, 1984. 1
- [36] Thomas Müller, Alex Evans, Christoph Schied, and Alexander Keller. Instant neural graphics primitives with a multiresolution hash encoding. *ACM Trans. Graph.*, 41(4):102:1–15, 2022. 1, 4
- [37] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Kopf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. PyTorch: An imperative style, high-performance deep learning library. In *NeurIPS*, 2019. 1, 4
- [38] Julien Philip, Sébastien Morgenthaler, Michaël Gharbi, and George Drettakis. Free-viewpoint indoor neural relighting from multi-view stereo. *ACM Trans. Graph.*, 40(5):194:1–18, 2021. 2, 6
- [39] René Ranftl, Katrin Lasinger, David Hafner, Konrad Schindler, and Vladlen Koltun. Towards robust monocular depth estimation: Mixing datasets for zero-shot cross-dataset transfer. *TPAMI*, 44(3):1623–1637, 2022. 5
- [40] Viktor Rudnev, Mohamed Elgharib, William Smith, Lingjie Liu, Vladislav Golyanik, and Christian Theobalt. NeRF for outdoor scene relighting. In *ECCV*, 2022. 2
- [41] Pratul P. Srinivasan, Ben Mildenhall, Matthew Tancik, Jonathan T. Barron, Richard Tucker, and Noah Snavely. Light-house: Predicting lighting volumes for spatially-coherent illumination. In *CVPR*, 2020. 2
- [42] 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 *CVPR*, 2021. 2
- [43] Matthew Tancik, Pratul P. Srinivasan, Ben Mildenhall, Sara Fridovich-Keil, Nithin Raghavan, Utkarsh Singhal, Ravi Ramamoorthi, Jonathan T. Barron, and Ren Ng. Fourier features let networks learn high frequency functions in low dimensional domains. In *NeurIPS*, 2020. 2
- [44] Matthew Tancik, Ethan Weber, Evonne Ng, Ruilong Li, Brent Yi, Terrance Wang, Alexander Kristoffersen, Jake Austin, Kamyar Salahi, Abhik Ahuja, David Mcallister, Justin Kerr, and Angjoo Kanazawa. Nerfstudio: A modular framework for neural radiance field development. In *SIGGRAPH*, pages 72:1–12, 2023. 5, 6, 1, 2
- [45] Ayush Tewari, Justus Thies, Ben Mildenhall, Pratul Srinivasan, Edgar Tretschk, Yifan Wang, Christoph Lassner, Vincent Sitzmann, Ricardo Martin-Brualla, Stephen Lombardi, Tomas Simon, Christian Theobalt, Matthias Niessner, Jonathan T. Barron, Gordon Wetzstein, Michael Zollhöfer, and Vladislav Golyanik. Advances in neural rendering. *Comput. Graph. Forum*, 41(2):703–735, 2022. 1
- [46] Kushagra Tiwary, Akshat Dave, Nikhil Behari, Tzofi Klinghoffer, Ashok Veeraraghavan, and Ramesh Raskar. ORCa: Glossy objects as radiance field cameras. In *CVPR*, 2023. 3
- [47] 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. In *CVPR*, 2022. 1, 2, 3, 5, 6, 7
- [48] Zian Wang, Tianchang Shen, Jun Gao, Shengyu Huang, Jacob Munkberg, Jon Hasselgren, Zan Gojic, Wenzheng Chen, and Sanja Fidler. Neural fields meet explicit geometric representations for inverse rendering of urban scenes. In *CVPR*, 2023. 2
- [49] Liwen Wu, Rui Zhu, Mustafa B. Yaldiz, Yin hao Zhu, Hong Cai, Janarбек Matai, Fatih Porikli, Tzu-Mao Li, Manmohan Chandraker, and Ravi Ramamoorthi. Factorized inverse path tracing for efficient and accurate material-lighting estimation. In *ICCV*, 2023. 2, 5
- [50] Xiuchao Wu, Jiamin Xu, Zihan Zhu, Hujun Bao, Qixing Huang, James Tompkin, and Weiwei Xu. Scalable neural indoor scene rendering. *ACM Trans. Graph.*, 41(4):98:1–16, 2022. 3, 6, 7
- [51] Yiheng Xie, Towaki Takikawa, Shunsuke Saito, Or Litany, Shiqin Yan, Numair Khan, Federico Tombari, James Tompkin, Vincent Sitzmann, and Srinath Sridhar. Neural fields in visual computing and beyond. *Comput. Graph. Forum*, 2022. 1
- [52] Jiamin Xu, Xiuchao Wu, Zihan Zhu, Qixing Huang, Yin Yang, Hujun Bao, and Weiwei Xu. Scalable image-based indoor scene rendering with reflections. *ACM Trans. Graph.*, 40(4):60:1–14, 2021. 3
- [53] Linning Xu, Vasu Agrawal, William Laney, Tony Garcia, Aayush Bansal, Changil Kim, Samuel Rota Bulò, Lorenzo Porzi, Peter Kotschieder, Aljaž Božič, Dahua Lin, Michael Zollhöfer, and Christian Richardt. VR-NeRF: High-fidelity virtualized walkable spaces. In *SIGGRAPH Asia*, 2023. 1, 6, 8, 4, 7
- [54] Zhiwen Yan, Chen Li, and Gim Hee Lee. NeRF-DS: Neural radiance fields for dynamic specular objects. In *CVPR*, 2023. 2
- [55] Jiawei Yang, Marco Pavone, and Yue Wang. FreeNeRF: Improving few-shot neural rendering with free frequency regularization. In *CVPR*, 2023. 4
- [56] Yao Yao, Jingyang Zhang, Jingbo Liu, Yihang Qu, Tian Fang, David McKinnon, Yanghai Tsin, and Long Quan. NeILF: Neural incident light field for physically-based material estimation. In *ECCV*, 2022. 2
- [57] Ze-Xin Yin, Jiaxiong Qiu, Ming-Ming Cheng, and Bo Ren. Multi-space neural radiance fields. In *CVPR*, 2023. 3, 6, 7, 5
- [58] Bohan Yu, Siqi Yang, Xuanning Cui, Siyan Dong, Baoquan Chen, and Boxin Shi. MILO: Multi-bounce inverse rendering for indoor scene with light-emitting objects. *IEEE TPAMI*, 45(8):10129–10142, 2023. 2
- [59] Zehao Yu, Songyou Peng, Michael Niemeyer, Torsten Sattler, and Andreas Geiger. MonoSDF: Exploring monocular geometric cues for neural implicit surface reconstruction. In *NeurIPS*, 2022. 5
- [60] Chong Zeng, Guojun Chen, Yue Dong, Pieter Peers, Hongzhi Wu, and Xin Tong. Relighting neural radiance fields with shadow and highlight hints. In *SIGGRAPH*, 2023. 2
- [61] Junyi Zeng, Chong Bao, Rui Chen, Zilong Dong, Guofeng Zhang, Hujun Bao, and Zhaopeng Cui. Mirror-NeRF: Learn-

- ing neural radiance fields for mirrors with Whitted-style ray tracing. In *ACM Multimedia*, 2023. 3
- [62] Jingyang Zhang, Yao Yao, Shiwei Li, Jingbo Liu, Tian Fang, David McKinnon, Yanghai Tsin, and Long Quan. NeILF++: Inter-reflectable light fields for geometry and material estimation. In *ICCV*, 2023. 2
- [63] Kai Zhang, Fujun Luan, Qianqian Wang, Kavita Bala, and Noah Snavely. PhysSG: Inverse rendering with spherical Gaussians for physics-based material editing and relighting. In *CVPR*, 2021. 2
- [64] Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *CVPR*, 2018. 7
- [65] Xiuming Zhang, Pratul P. Srinivasan, Boyang Deng, Paul Debevec, William T. Freeman, and Jonathan T. Barron. NeR-Factor: Neural factorization of shape and reflectance under an unknown illumination. *ACM Trans. Graph.*, 40(6):237:1–18, 2021. 2
- [66] Yuanqing Zhang, Jiaming Sun, Xingyi He, Huan Fu, Rongfei Jia, and Xiaowei Zhou. Modeling indirect illumination for inverse rendering. In *CVPR*, 2022. 2
- [67] Youjia Zhang, Teng Xu, Junqing Yu, Yuteng Ye, Junle Wang, Yanqing Jing, Jingyi Yu, and Wei Yang. NeMF: Inverse volume rendering with neural microflake field. In *ICCV*, 2023. 2
- [68] Yiyu Zhuang, Qi Zhang, Xuan Wang, Hao Zhu, Ying Feng, Xiaoyu Li, Ying Shan, and Xun Cao. NeAI: A pre-convoluted representation for plug-and-play neural ambient illumination. In *AAAI*, 2024. 2