DOI: 10.1111/cgf.15163

COMPUTER GRAPHICS *forum*
Volume 0 (2024), number 0, e15163

TraM-NeRF: Tracing Mirror and Near-Perfect Specular Reflections Through Neural Radiance Fields

Leif Van Holland, Ruben Bliersbach, Jan U. Müller, Patrick Stotko and Reinhard Klein

Institute of Computer Science II, University of Bonn, Bonn, Germany
holland@cs.uni-bonn.de, s6rublie@uni-bonn.de, {muellerj, stotko, rk}@cs.uni-bonn.de

Abstract

Implicit representations like neural radiance fields (NeRF) showed impressive results for photorealistic rendering of complex scenes with fine details. However, ideal or near-perfectly specular reflecting objects such as mirrors, which are often encountered in various indoor scenes, impose ambiguities and inconsistencies in the representation of the re-constructed scene leading to severe artifacts in the synthesized renderings. In this paper, we present a novel reflection tracing method tailored for the involved volume rendering within NeRF that takes these mirror-like objects into account while avoiding the cost of straightforward but expensive extensions through standard path tracing. By explicitly modelling the reflection behaviour using physically plausible materials and estimating the reflected radiance with Monte-Carlo methods within the volume rendering formulation, we derive efficient strategies for importance sampling and the transmittance computation along rays from only few samples. We show that our novel method enables the training of consistent representations of such challenging scenes and achieves superior results in comparison to previous state-of-the-art approaches.

Keywords: rendering, image-based rendering, ray tracing, reflectance and shading models

CCS Concepts: • Computing methodologies → Image-based rendering; Ray tracing; Reflectance modelling

1. Introduction

3D re-construction and modelling of real-world scenes has been a major research field for decades and plays a crucial role in a diverse range of applications such as video gaming, movies, advertisement, education as well as AR and VR scenarios. With the recent emergence of neural scene representations and, especially, neural radiance fields (NeRF) [MST*21], a compelling degree of photorealism and immersion of the rendered views has been achieved which inspired many further developments [ZRSK20, RPLG21, BMT*21, MESK22, WWG*22, CZL*22]. By combining graphics-based volume rendering with an efficient representation of scene density and radiance using neural networks in terms of multi-layer perceptrons (MLP), NeRF enables capturing various effects including view-dependent changes of object appearances or volumetric phenomena like clouds. However, objects with ideal and near-perfect specular reflection behaviour which are often encountered in various scenarios and, in particular, many indoor scenes impose a significant challenge to the representation capabilities of radiance fields as they induce a very specific pattern in the light transport. For the case of a planar mirror, a symmetric version of the visible scene parts can be observed which appears to be located behind the mirror and gives the illusion of viewing the respective content through a window. While, *a priori*, this ambiguity results into two plausible and consistent interpretations of the structure of the surrounding environment, additional views directly from behind the mirror object allows resolving the scenario as the representation of the virtual mirrored scene collides with the observations. As a consequence, severe artifacts will be introduced in the scene representation of NeRF as the underlying volume rendering approach for rendering traces the primary viewing rays and, in turn, implicitly always prefers the inconsistent interpretation. Several approaches addressed this issue by decomposing the scene into two or more individually consistent radiance fields [GKB*22, YQCR23] or employ standard path tracing [ZXY*23, MVKFK23] in combination with an extended volumetric field to infer normal directions and specular reflection probabilities [ZBC*23]. However, this significantly increases the computational burden both in terms of training performance and rendering speed and limits their application into other sophisticated and advanced NeRF approaches.

Figure 1: *Examples of novel views rendered with our proposed approach on scenes with mirror surfaces (left, centre) and near-perfect specular surfaces (right).*

In this paper, we direct our attention towards an efficient formulation of reflection tracing within the volume rendering procedure of NeRF that can be easily adopted in several NeRF variants to enhance their capabilities in handling mirror-like objects. To this end, we extend the single-ray absorption volume integration by considering the contributions of reflected radiance towards the observed radiance by the camera, effectively moving our model closer to full physically interpretable light transport in the process. Our proposed method, referred to as TraM-NeRF, explicitly integrates reflected radiance based on sparsely annotated near-specular surfaces. Combining NeRF volume rendering and ray-tracing with physically plausible materials at intersection points introduces an inductive bias into the training of TraM-NeRF that enables it to learn a single coherent scene representation, even when geometry has only been observed in a reflection. Our combined radiance estimator allows us to reduce its variance compared to a standard Monte-Carlo approach with only little computational overhead by increasing the number of observed reflection directions independently of the number of network queries. Some of our results are shown in Figure 1.

In summary, the key contributions of this work are given below:

- We present TraM-NeRF, an extension of NeRF that efficiently represents scenes with mirror-like surfaces, modelling high-frequency reflections in a physically plausible manner within a single coherent scene representation.
- We derive a transmittance-aware formulation of the rendering equation to explicitly model reflected radiance at mirror-like surfaces. Additionally, we introduce efficient strategies for importance sampling and transmittance computation, resulting in a reduction in the number of network evaluations compared to Monte-Carlo estimation.
- We demonstrate the benefits of our formulation in comparison to previous state-of-the-art methods on a variety of challenging synthetic and real-world scenes, some of which include multiple planar and non-planar mirror-like surfaces.

The source code of our implementation is available at https://github.com/Rubikalubi/TraM-NeRF.

2. Related Work

2.1. Neural scene representations

Synthesizing novel views of complex scenes has gained increasing interest due to the promising results achieved with neural scene representations [LSS*19, SZW19, NMOG20, BXS*20a, BXS*20b]. Among these, especially the work on NeRF [MST*21] excels in terms of the quality and degree of photorealism of the rendered images and has become very popular, also due to its simple but effective formulation. In particular, NeRF leverages volume rendering to accumulate the scattered lighting contributions along the traced viewing rays, which are represented using volumetric density and view-dependent radiance and parametrized using MLPs. Various extensions have been developed to further enhance the performance and quality of the original approach such as accelerating the training [MESK22, CXG*22, FKYT*22] as well as the rendering processes [RPLG21, GKJ*21], reducing aliasing artifacts by replacing ray-based marching with an integration of 3D frustums [BMT*21, BMV*22, BMV*23, IMWB23], rendering fine details at very high resolution [WWG*22, WLS*22, LLGG23] or lifting its capabilities to also handle unbounded scenes [ZRSK20, BMV*22] and to reconstruct from in-the-wild image collections [MBRS*21, CZL*22, FKMW*23] or low dynamic range images with low or varying exposure [HZF*22, MHMB*22].

Besides these advances, the underlying representation given by volumetrically baked radiance and density does not account for manipulation tasks like exchanging the environment illumination, so significant effort has been spent into more plausible, physics-inspired scene representations. Thus, various methods [ZSD*21, BBJ*21, SDZ*21, BEK*22, JLX*23] considered factorizing the radiance field into shape with normals, surface material parameters in terms of a bidirectional reflectance distribution function (BRDF) as well as environment illumination. Further approaches [ZLW*21, FSV*23, LCL*23, WHL*23, GHZ*23] replaced the density-based shape representation by implicit surfaces via signed distance functions (SDF) for a more accurate estimation of the object geometry and normals.

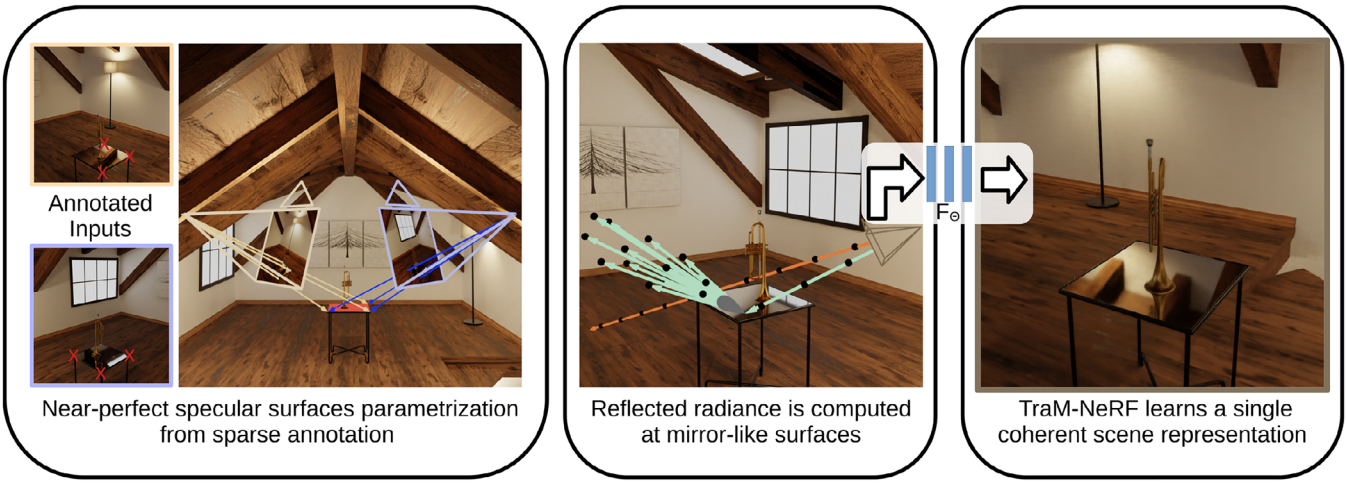


Figure 2: *Overview of our proposed method TraM-NeRF. Our approach to parameterize nearly specular surfaces using only sparse annotations. We introduce a radiance estimator, a crucial component of TraM-NeRF, which combines volume and reflected radiance integration for training and rendering the model. TraM-NeRF learns to represent the observed radiance in a single coherent network.*

2.2. Specular reflections in neural representations

Objects with highly reflective materials often exhibited in captured scenes impose are challenging to re-construct in decomposed representations of NeRF and have, in turn, attracted increasing attention. Ref-NeRF [VHM*22] re-parametrizes the observed radiance based on the local normal vector and its angle to the view direction to a simpler model that shares common structures across multiple views. PhySG [ZLW*21] employs Spherical Gaussians to represent specular reflections in the BRDF which has been later extended by splitting the illumination into a direct and indirect component, each modelling an individual specular reflection [ZSH*22]. Ref-NeuS [GHZ*23] detects anomalies in the rendered images caused by reflections and incorporate a respective reflection score into the photometric loss as a guidance. NeRO [LWL*23] uses a split-sum approximation to estimate the shape of an reflective object in a first stage and then optimize its BRDF in a second stage. Other works instead directly trace reflections either only in the ideal reflection direction assuming a low material roughness [LCL*23] or using path tracing evaluated via Monte-Carlo estimators [WHL*23]. Recently, volumetric microflake [ZXY*23] and microfacet [MVKFK23] fields presented a hybrid rendering approach by combining the ray marching of volume rendering with importance-sampled path tracing according to the distribution of the micro structures.

Most closely related to our work are techniques that explicitly model mirror reflections within the scene. NeRFReN [GKB*22] decomposes the scene into two independently traversed and rendered radiance fields, consisting of an ordinary NeRF for the transmitted radiance as well as an additional NeRF that only covers the reflected radiance. The final synthesized image is then obtained by blending the results for the transmitted and reflected fields. MS-NeRF [YQCR23] learns radiance and weights into multiple feature fields that are decoded by small MLPs for rendering and then blended together. Mirror-NeRF [ZBC*23] follows a different direction by representing the scene in a single radiance field and instead further tracing the rays in the ideal reflection direction after hitting a mirror. The respective normal directions and reflection probabilities used for reflecting the rays are additionally learned in the volumetric neural field using additional regularization terms that constrains the solution to follow the assumption of planar mirrors.

In contrast to the aforementioned approaches, the primary focus of this work lies in the efficient rendering of unified radiance fields of scenes with not necessarily planar but instead polygonial-shaped mirror and near-perfect specular reflecting objects. Here, we particularly consider using only a low number of network evaluations for each importance-sampled reflection ray while achieving a significantly lower variance than standard Monte-Carlo estimators.

3. Method

We start with a brief overview of NeRF and then introduce our radiance estimator, a key component of TraM-NeRF. This estimator combines volume and reflected radiance integration for rendering and model training. We then discuss our approach to parameterize nearly specular surfaces using sparse annotations. Finally, we provide implementation and training details for transparency and reproducibility. An overview of our approach is shown in Figure 2.

3.1. Neural radiance fields

We build upon the neural implicit scene representation proposed by Mildenhall *et al.* [MST*21] which uses a simple MLP F_Θ to infer a RGB colour value $c \in \mathbb{R}^3$ and a density $\sigma \in \mathbb{R}$ for a given spatial location $x \in \mathbb{R}^3$ and viewing direction $d \in \mathbb{R}^3$. In order to also capture high-frequency details, x is first lifted into a higher-dimensional space using a positional encoding

$$\gamma(x) = \left(\sin\left(2^l \pi x\right), \cos\left(2^l \pi x\right)\right)_{l=0}^{L-1}. \quad (1)$$

To render an image using F_Θ, we define a camera and consider rays starting at the camera centre $o \in \mathbb{R}^3$ with direction $d \in \mathbb{R}^3$, such that $r(t) = o + t\,d$ represents a point on the ray. In the original NeRF formulation, the observed radiance

$$C_e(r) = \int_{t_n}^{t_f} T(t_n \to t)\,\sigma(r(t))\,c(r(t), d)\,\mathrm{d}t \tag{2}$$

corresponding to ray r is given by the volume integration through an absorbing medium where $T(t_n \to t) = \exp(-\int_{t_n}^{t} \sigma(r(s))\,\mathrm{d}s)$ represents the accumulated transmittance up to distance t, and $t_n, t_f \in \mathbb{R}$ are hyperparameters that determine the nearest and farthest distance for points along any ray r. To be able to compute Equation 2 in practice, the integral is numerically approximated using a quadrature [Max95] with K samples at locations t_k along the ray, yielding the following discrete sum:

$$C_e(r) \approx \hat{C}_e(r) = \sum_{k=1}^{K} T_k \left(1 - e^{-\sigma_k \delta_k}\right) c_k. \tag{3}$$

Here, $\delta_k = t_k - t_{k-1}$ is the distance between successive locations and $T(t_n \to t_k) \approx T_k = e^{-\sum_{j=1}^{k-1} \sigma_j \delta_j}$ approximates the accumulated transmittance. To avoid relying solely on a discrete subset of locations, we employ stratified sampling to select t_k, as proposed by Mildenhall *et al.* [MST*21].

During training, the parameters Θ of the MLP are optimized via gradient descent using a photometric loss $\mathcal{L}$ defined as the mean squared error between the ground-truth colours $C^*(r)$ and the rendered images $\hat{C}_e(r)$ over a batch of rays R:

$$\mathcal{L} = \frac{1}{|R|} \sum_{r \in R} \left\| C^*(r) - \hat{C}_e(r) \right\|_2^2. \tag{4}$$

3.2. Radiance integration at near-perfect specular surfaces

Assume that the camera ray $r(t) = o + t\,d$ intersects in a point x with a known near-specular surface, whose parameterization will be discussed in Section 3.3. To allow a model to learn a consistent representation of observed radiance in a single radiance field, we drop the assumption of NeRF that the ray terminates (i.e. the transmittance vanishes) at an opaque surface. Instead, TraM-NeRF relies on the rendering equation [Kaj86] to compute its predicted radiance at intersection points, which states that the radiance $C(x, \omega_o)$ at a point x when observed from direction ω_o is the sum of the emitted radiance $C_e(x, \omega_o)$ and the reflected radiance $C_r(x, \omega_o)$:

$$C(x, \omega_o) = C_e(x, \omega_o) + C_r(x, \omega_o). \tag{5}$$

Here, the reflected radiance is obtained by evaluating the transport integral

$$C_r(x, \omega_o) = \int_\Omega f(x, \omega_i, \omega_o)\, C(x, \omega_i) \cos\theta_i\,\mathrm{d}\omega_i \tag{6}$$

over the visible hemisphere Ω where $f(x, \omega_i, \omega_o)$ denotes the BRDF and θ_i is the angle between the surface normal at x and the incident direction ω_i.

We assume light travels from a source toward the camera. Following established literature [Hei18, DB23], incident and outgoing light directions extend outward from x. Thus, the incident direction faces toward the light source, while the outgoing direction faces toward the camera. Accordingly, we set the outgoing direction as $\omega_o = -d$, where d represents the camera direction.

Combining surface rendering and volume integration. In order to combine the radiance integration and volume integration, TraM-NeRF assumes that a primary ray scatters into multiple reflected rays at the intersecting point. Since this ray has passed through an absorbing medium, the combined radiance of the out-branching rays should be attenuated by the transmittance along the intersecting ray. Whereas NeRF integrates the density along a ray from a starting point close to the camera position up to a point which is chosen *a priori* based on the extent of the scene, TraM-NeRF modifies the upper integration bound to stop at the intersection point. We formalize this concept by introducing a ray length function $\tau(x, \omega)$ which returns the length from the ray origin to the point where it intersects with the detected geometry. Therefore, we obtain a transmittance-aware version of the rendering equation

$$C(x, \omega_o) = C_e(x, \omega_o) + T_{\omega_o}(t_n \to \tau(x, \omega_o)) \cdot C_r(x, \omega_o). \tag{7}$$

which takes the attenuation from the absorbing medium into account by multiplying the reflected radiance with the transmittance $T_{\omega_o}(t_n \to \tau(x, \omega_o))$. The emitted radiance observed at point x from direction ω_o is computed following [MST*21] by raymarching through the emissive volume until the intersection point is reached:

$$C_e(x, \omega_o) = \int_{t_n}^{\tau(x,\omega_o)} T_{\omega_o}(t_n \to t)\,\sigma_{\omega_o}(t)\,c_{\omega_o}(t)\,\mathrm{d}t \tag{8}$$

where $c_{\omega_o}(t)$ represents the direction-dependent colour estimated by the model at a point located t units along the ray (x, ω_o). Note that our modified version retains the offset t_n to prevent double-counting the intersection point in emitted and reflected radiance calculations.

Monte-Carlo estimator of reflected radiance. Our efficient reflected radiance estimator builds upon an established approximation method for the transport integral, employing importance sampling to evaluate a Monte-Carlo estimator. This estimator is then modified to reduce additional variance introduced when importance sampling a BRDF function, all while keeping the number of network evaluations constant. These adjustments enhance the computational efficiency in determining the reflected radiance.

Considering the transport integral in Equation 6, the respective estimator, which samples N incident light directions ω_i from a candidate distribution $p(\cdot)$, is given by

$$C_r(x, \omega_o) \approx \frac{1}{N} \sum_{i=1}^{N} \frac{f(x, \omega_o, \omega_i)}{p(\omega_i)} C(x, \omega_i). \tag{9}$$

Note how the subscript i in ω_i not only denotes the incident direction but also serves as an index indicating the ith (incident) direction sample drawn from the hemisphere Ω with probability $p(\omega_i)$.

In order to obtain a suitable candidate distribution derived from the BRDF f, we utilize the well-established microfacet theory to model f at the intersection point, assuming that roughness arises from a height field of tiny facets distributed according to a distribution $D_\alpha(\cdot)$ with roughness parameter α [CT82]. In particular, we

use the widely used GGX reflection model [WMLT07] which defines the BRDF to be

$$f(x, \omega_o, \omega_i) = \frac{F(\omega_i, h)\, G(\omega_o, \omega_i, h)\, D_\alpha(h)}{4 \cos\theta_i \cos\theta_o} \tag{10}$$

where h is the half-vector between the incident and outgoing direction, θ_o the angle between ω_o and the surface normal, F is the Fresnel term, and $G(\cdot)$ a coefficient describing the average attenuation that results from shadowing and masking between microfacets. For formal definitions of D_α, F and G, please refer to Walter *et al.* [WMLT07]. Utilizing an optics-based analytical GGX reflectance model ensures physical consistency in the learned radiance function and comes with the additional benefit of a well-studied importance sampling technique [Hei18, DB23]. We use visible normal sampling (VNDF) which defines a candidate distribution

$$p(h) = \frac{\max\{0, \langle \omega_o | h \rangle\}\, G_1(\omega_o)\, D_\alpha(h)}{4 \cos\theta_i \cos\theta_o} \tag{11}$$

that takes the average attenuation due to microfacet masking G_1 into consideration and has a closed-form sampling routine [Hei18]. Note that VNDF is a distribution over the half-vectors h instead of incident light directions. However, the incident light direction can be computed via a reflection of the outgoing direction about the half-vector. Thus, the resulting estimator for the reflected radiance is

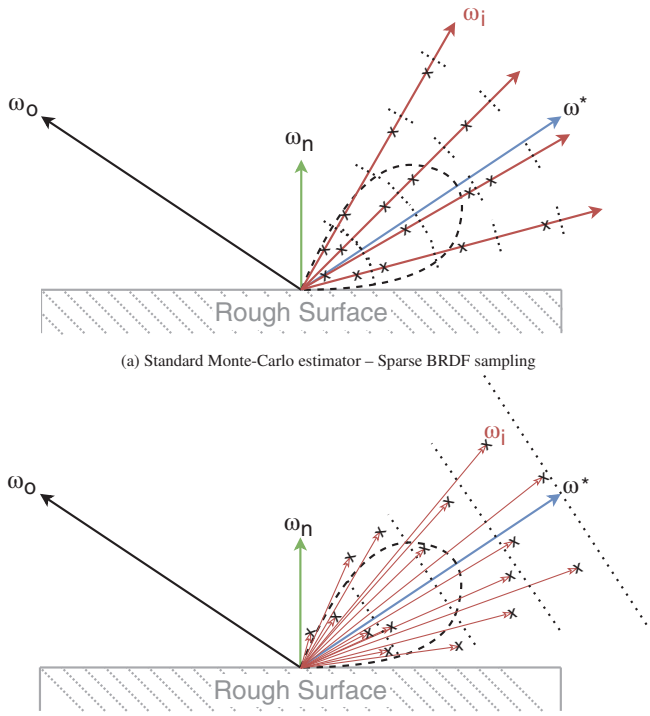
$$C_r(x, \omega_o) \approx \frac{1}{N} \sum_{i=1}^{N} \underbrace{\frac{F(\omega_i, h)\, G(\omega_o, \omega_i, h)}{G_1(\omega_o)}}_{=: f'(\omega_o, \omega_i)} C(x, \omega_i). \tag{12}$$

Efficient reflected radiance approximation. To contextualize our efficient radiance approximation, we initially examine the number of network evaluations needed when the estimator from Equation 12 is used in the combined surface rendering and volume integration. This analysis follows the assumption made in NeRF [MST*21], where the volume integral is discretized under the assumption of piece-wise constant radiance along the ray direction. For simplicity, the analysis focuses on a scenario where the camera ray incurs no absorption before intersecting with the surface and the reflected rays do not intersect with any detected surface. In this setting, the radiance observed for a camera ray would be

$$C(o, \omega_o) = \frac{1}{N} \sum_{i=1}^{N} f'(\omega_o, \omega_i) \sum_{k=1}^{K} T_{\omega_i}(t_n \to t_k)\big(1 - e^{-\sigma_{\omega_i,k}\delta_{\omega_i,k}}\big) c_{\omega_i,k}. \tag{13}$$

Figure 3a provides a visualization of the network evaluation pattern for each directional sample. A crucial observation is that computing this equation entails K network evaluations for each direction sampled using VNDF. To achieve a radiance estimate with minimal noise, a large number of directional samples are required, necessitating a significant number of costly network evaluations.

In light of the aforementioned challenges, we aim to improve the computational efficiency of this procedure in TraM-NeRF. Our strategy involves increasing the number of directional samples without the need for additional network evaluations. This optimization leverages the observation that scenes with diffuse or low frequency surfaces reflections can be adequately handled by the standard NeRF model. However, it encounters difficulties in representing scenes



(a) Standard Monte-Carlo estimator – Sparse BRDF sampling

(b) Our estimator – Dense BRDF sampling

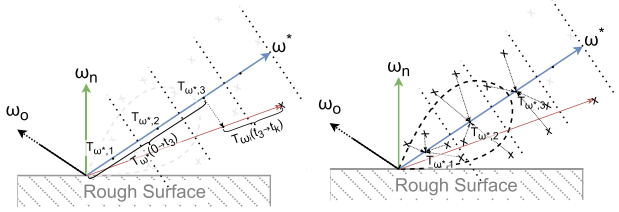
Figure 3: *Patterns for BRDF sampling and network evaluation, resulting in different radiance estimators. (a) In the standard Monte-Carlo approach, the network is evaluated at positions (indicated by cross markers) chosen using stratified sampling for each sampled direction ω_i. (b) Our estimator draws directional samples within segments (dashed lines) along the ideal reflection direction ω^*, resulting in a higher angular coverage of the specular lobe with the same number of network evaluations.*

with surfaces which display high-frequency reflections that vary significantly with the viewing direction. In TraM-NeRF, we combine this observation with the assumptions of scene boundedness and locally smooth network-predicted density. These assumptions allow our estimator to focus primarily on estimating reflected radiance for near-specular surfaces. Consequently, our estimator can assume a narrow spread of light directions in the samples. By combining these insights, we expect that the transmittance remains nearly constant with respect to the sampled directions. In particular, we approximate the transmittance along all sampled directions ω_i with the transmittance along the ideal reflection direction ω^*:

$$T_{\omega_i}(t_n \to t_k) \approx T_{\omega^*}(t_n \to t_k). \tag{14}$$

This way, we can interchange the order of the Monte-Carlo integration and the NeRF volume integration:

$$\begin{aligned} C_r(x, \omega_o) &= \frac{1}{N} \sum_{i=1}^{N} f'(\omega_o, \omega_i) \sum_{k=1}^{K} T_{\omega_i}(t_n \to t_k)\big(1 - e^{-\sigma_{\omega_i,k}\delta_{\omega_i,k}}\big) c_{\omega_i,k} \\ &\approx \sum_{k=1}^{K} T_{\omega^*}(t_n \to t_k) \frac{1}{N} \sum_{i=1}^{N} f'(\omega_o, \omega_i)\big(1 - e^{-\sigma_{\omega_i,k}\delta_{\omega_i,k}}\big) c_{\omega_i,k} \end{aligned} \tag{15}$$



(a) Transmittance approximation (b) BRDF-weighted density

Figure 4: *Transmittance approximation used by our estimator. (a) The transmittance $T_{\omega_i}(t_n \to t_k)$ in direction ω_i is approximated using the transmittance $T_{\omega^*}(t_n \to t_{k-1})$ along the ideal reflection direction ω^* for near-perfect specular surfaces. (b) The transmittance in the ideal reflection direction is computed using the BRDF-weighted density of samples within each segment.*

Intuitively, our estimator divides the ideal reflection ray into K segments and traces transmittance solely along the ideal reflection direction. Within each segment, we randomly select positions and evaluate n directional samples, each contributing to the calculation only once with their impact attenuated based on the transmittance along the ideal reflection direction, which is depicted in Figure 4a. These positions for evaluating the directional samples are uniformly chosen from the interval $[\frac{t_{k-1}+t_k}{2}, \frac{t_k+t_{k+1}}{2}]$. A visualization of this sampling strategy and its resulting evaluation points is shown in Figure 3b. For a more in-depth explanation and its adaptation to a hierarchical optimization procedure, please refer to Section 3.4.

Our estimator computes transmittance once per segment, enabling us to increase the number of directional samples without requiring additional network evaluations to accumulate transmittance. This trade-off balances transmittance precision against directional sample count, reducing noise in reflected radiance. To further reduce the number of network evaluations, we use an average of BRDF-weighted density predictions per segment to calculate the transmittance

$$T_{\omega^*}(t_n \to t_k) \approx e^{-\sum_{j=1}^{k-1}\left[\frac{1}{n}\sum_{i=1}^{n} f'(\omega_o,\omega_i)\,\sigma_{\omega_i,j}\right]\delta_j}. \tag{16}$$

which is visualized in Figure 4b.

3.3. Mirror parameterization and annotation

To compute the reflection of a ray at near-specular surfaces, TraM-NeRF requires an intersection test function that returns both the intersection location and the surface normal at that location.

In case of planar polygonal surfaces, we get sufficiently accurate annotations using only a small number of annotated input images per scene. We represent these surfaces as triplets of triangle vertices $T = (v_1, v_2, v_3)$, $v_i \in \mathbb{R}^3$ which allows for efficient intersection tests with rays in the rendering step [MT97]. Given the screen space annotations of three corners in at least two images and their camera poses, the annotations correspond to rays through the scene. In particular, the jth annotation of vertex v_i defines a ray $r_{ij}(t) = o_j + t\,d_{ij}$ with camera origin o_j and ray direction d_{ij} with $\|d_{ij}\|_2 = 1$. The estimated 3D location $\hat{v}_i$ of the vertex is then given as the point minimizing the lengths of the orthogonal projection onto each ray:

$$\hat{v}_i = \min_{v} \sum_{j} \left\| v - (o_j + d_{ij}\,(v - o_j)\,d_{ij}) \right\|_2^2. \tag{17}$$

We can additionally exploit the property that all triangles of a planar polygonal mirror lie on the same plane. To increase the robustness against inaccuracies in the annotations, we compute the normal of that plane using principal component analysis applied on the set of annotated vertices of the planar surface.

In Section 4, we show how TraM-NeRF can be used to represent more complex, non-planar surfaces using a cylinder as an example. We parameterize a cylindrical reflector using start and endpoints $p, p' \in \mathbb{R}^3$ and radius $R \in \mathbb{R}$. To infer these parameters, we first create sparse annotations of the cylinder region in multiple images and generate a binary segmentation mask for each of these images [DM21]. Next, we generate a visual hull from the masks [Lau94] and compute an oriented bounding box of its vertices. The parameters are initialized as the maximal cylinder that fits inside this bounding box, where the longitudinal axis is chosen as the longest side of the box, assuming that the cylinder's length is larger than its radius. Then, the parameters are optimized by rendering the silhouette of the cylinder from the views of the mask images and minimizing a silhouette loss between the renderings and the segmentation masks using a differentiable rasterization pipeline [LHK*20]. We were able to produce accurate estimates for the cylinder geometry by sparsely annotating only 11 input images for the real-world scene shown in the last row of Figure 5.

3.4. Implementation and training details

To assess and compare our estimator for reflected radiance, TraM-NeRF leverages the NeRF framework [MST*21] and is implemented using PyTorch [PGM*19]. We chose to use the standard NeRF implementation to ensure a clear comparison of the improvements resulting from our contributions and to avoid potential confusion in the assessment of enhancements attributable specifically to our estimator in comparison to those resulting from different unrelated improvements. Nevertheless, our estimator is adaptable to various implementations of the radiance field networks, making it compatible with methods that enhance parametrization [BMT*21, BMV*22] or architecture [MESK22, CXG*22].

For training TraM-NeRF, we use a modified version of the NeRF training protocol [MST*21], using the Adam optimizer with a learning rate of 10^{-3} without decay. The Adam hyperparameters remained at their default values: $\beta_1 = 0.9$, $\beta_2 = 0.999$ and $\epsilon = 10^{-7}$. The model underwent 1.2×10^5 iterations of training on the synthetic scenes, and 2.4×10^5 iterations on the real-world scenes, with a batch size of 2^{14} pixels per iteration.

We apply the sampling process outlined in Section 3.2 to align with NeRF's hierarchical volume sampling approach [MST*21] consisting of a coarse and a fine stage. In the coarse stage, we employ stratified sampling to select points along each ray for network evaluation, which involves dividing the ray into K_c equal segments and uniformly sampling a position from each segment. In the fine stage, additional samples along each ray are generated using inverse transform sampling based on density predictions from the coarse

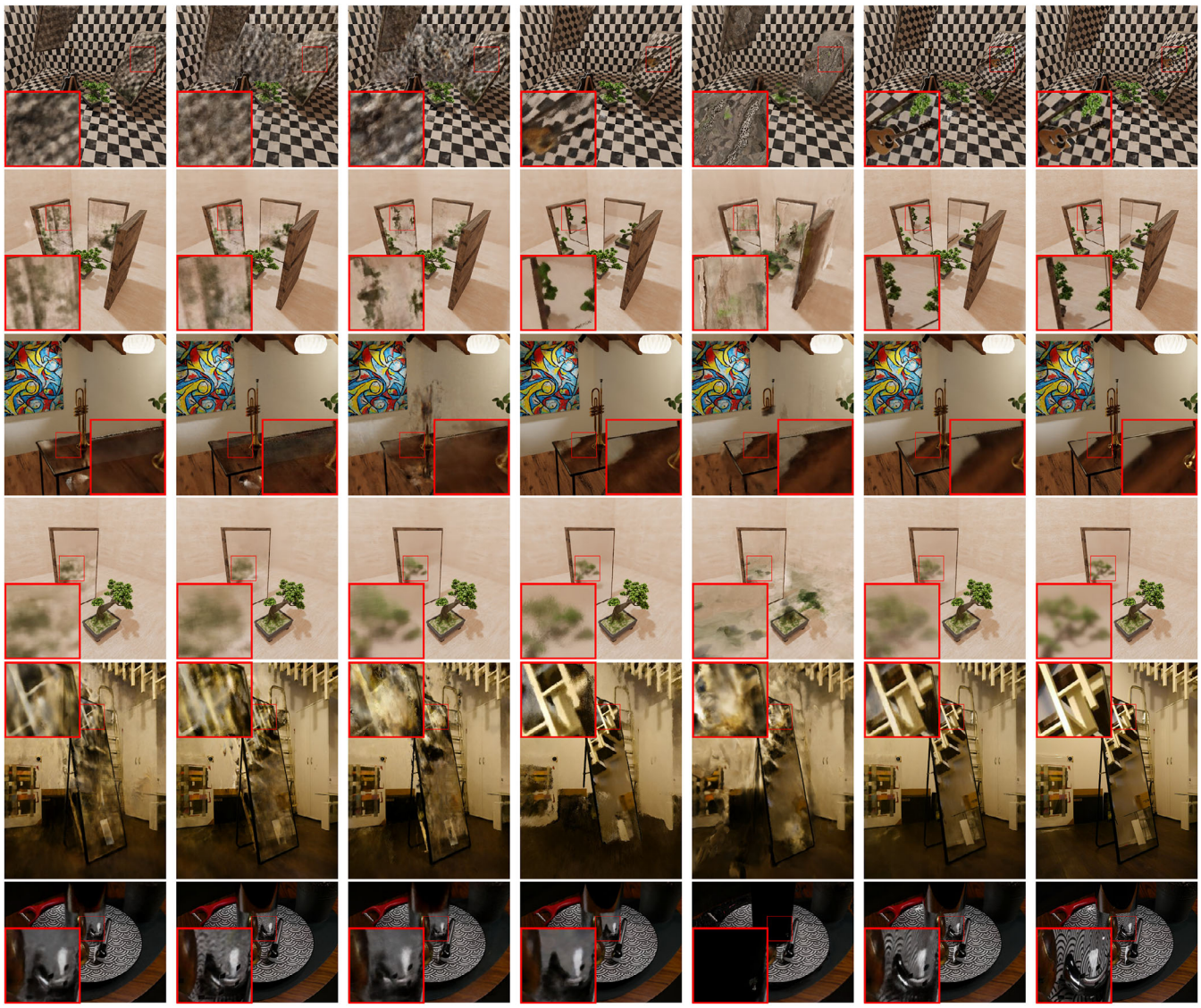


Figure 5: *Comparison of different methods (a–e) compared to ours (f) on our synthetic dataset with multiple mirrors (rows 1 and 2), near-perfect specular surfaces (rows 3 and 4) and real-world scenes (rows 5 and 6). The images shown are views from the test set of the respective scenes. (g) Shows the ground truth test image.*

stage, which results in an additional set of K_f samples. Given a set of samples, denoted as $t_1, t_2, \ldots, t_K$, where K corresponds to the number of coarse (K_c) or fine samples ($K_c + K_f$), TraM-NeRF establishes non-uniformly spaced intervals based on them. These intervals are defined as $\left[\frac{t_{k-1}+t_k}{2}, \frac{t_k+t_{k+1}}{2}\right]$, ensuring that each sample point t_k serves as the centre of a specific section. We now generate directional samples by choosing a uniformly distributed length

$$t_{k,i} \sim U_{\left[\frac{t_{k-1}+t_k}{2}, \frac{t_k+t_{k+1}}{2}\right]} \tag{18}$$

for each interval k and direction ω_i. Subsequently, the radiance field is queried for its density and colour predictions at specific points

$$x_{k,i} = x + \frac{t_{k,i}}{\langle \omega^* | \omega_i \rangle} \omega_i \tag{19}$$

where the dot product between the sampled direction and the ideal reflection direction ensures that $x_{k,i}$ falls within the interval, as illustrated in Figure 3a.

During training, we used two directional samples per camera ray to keep the number of network queries close to the original NeRF formulation. For the final results, we increased this number to 50 directional samples to enable rendering high-quality images.

4. Experiments

We ran multiple experiments to evaluate different facets of our approach on scenes with multiple planar and non-planar mirrors, and scenes containing near-specular surfaces, both generated synthetically and captured in the real world.

The synthetic scenes are created in the open-source 3D graphics tool Blender to be able to retrieve ground-truth parameters from the BRDF model. Matching the description in Section 3.2, the Blender material we selected for mirror-like surface uses the GGX distribution. All 3D models and textures are provided by BlenderKit and CGTrader as royalty-free assets. In total, we created 11 synthetic scenes with mirrors and five scenes with near-specular surfaces. For non-forward-facing scenes, we rendered 150 images per scene with cameras sampled from the upper hemisphere around the scene centre, looking towards the centre. As less coverage is required for the forward-facing scenes, we only rendered 100 images per scene for these cases.

To show the performance of our approach on real-world data, we captured three scenes using a DSLM camera and determined the camera parameters using structure-from-motion [SF16]. The real-world scenes have a varying amount of images, depending on the complexity of the mirror setup. We further assume that the surfaces in these scenes have zero roughness. For evaluation purposes, we withheld 20% of the images for each synthetic and real-world scene from the training process.

We compare our approach on our datasets, both qualitatively and quantitatively, against multiple baseline methods [MST*21, BMV*22] and recent methods that explicitly model reflections [VHM*22, GKB*22, YQCR23, ZBC*23]. To quantify the reconstruction quality, we use the commonly used metrics peak signal-to-noise ratio (PSNR), structural similarity index (SSIM) [WBSS04] and learned perceptual image patch similarity (LPIPS) [ZIE*18].

In addition to evaluating these over the whole image, we report a measurement of the quality restricted to the mirror regions. Here, the distance to ground truth is computed by first setting all non-mirror pixels to zero in both the result image and the corresponding ground truth using the mirror mask we attain from the annotations and then evaluating the metrics on the masked image pair. To reduce the influence of large non-mirror regions, we weight the contribution of each image to the overall score using the number of mirror pixels in that image, thus yielding a weighted mean of the individual per-image scores.

For the choice of hyperparameters, we followed the configuration files provided alongside the implementations of the respective methods. In the case of Mirror-NeRF [ZBC*23], which requires precise ground-truth mirror masks for each image, we generated these from the annotations that we use in our approach. Note that this generation does not account for partial occlusions of the mirrors, as determining these occlusions requires *a priori* knowledge of the entire scene geometry and would in turn be a strong prior. The usage of our annotations in NeRFReN is detailed in Section 4.3.

4.1. Multi-mirror scenes

The first two rows of Figure 5 show results of various related approaches on scenes with multiple mirrors. It can be seen that both baseline methods (a, b) and approaches that consider reflections more explicitly (c, d, e) struggle to re-construct higher-order reflections with regard to overall quality (a, b, c) and high-frequency details (d), while our method (f) can by design represent these regions with the same quality as the rest of the scene. This strength is also reflected in the quantitative evaluation in Table 1, as our method outperforms the other approaches consistently. Even though (e) focuses on modelling mirror effects using ray-tracing, we found that the method does not robustly handle multi-view inconsistencies imposed by our dataset.

4.2. Reflections of near-specular surfaces

The lower two rows of Figure 5 show a similar comparison on scenes with near-perfect specular surfaces. As before, (a) and (b) show a lack of re-construction quality in regions close to the near-specular surface due to multi-view inconsistencies. While (c) is able to resolve the inconsistencies, in the third row, it can be seen that it fails to learn a clear reflection. In both the mirror reflections and near-specular surface scenarios, we suppose that the lack of detail in the results of MS-NeRF are due to two effects: First, Yin *et al.* reduced the sizes of the individual radiance fields to roughly match the size of approaches using a single radiance field. This leads to less capacity per radiance field in the multi-space formulation. Second, the previous approaches are unable to aggregate information in the reflections from different views consistently, as they are either trying to resolve the multi-view consistencies directly, or move them into a separate radiance field.

Yet, MS-NeRF is able to achieve best PSNR scores when only considering mirror regions in scenes with near-specular surfaces. This could be caused as a side-effect of the change in radiance field sizes, as MS-NeRF is less prone to overfit on high-frequency details due to the reduced capacity, which is advantageous for non-perfect reflections. The low performance of Mirror-NeRF can be expected on this subset of scenes, as the method does not model non-zero surface roughness during training.

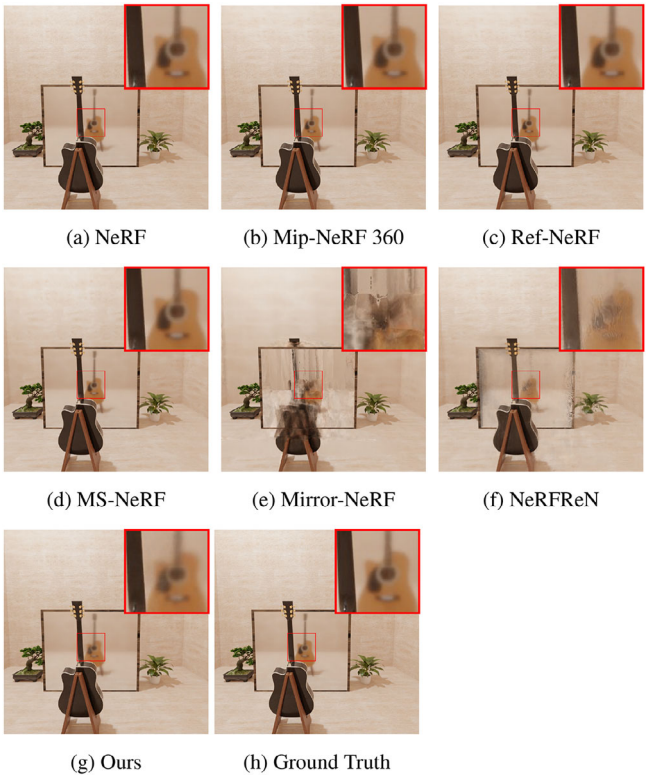
4.3. Forward-facing scenes

In order to compare our results with the approach of Guo *et al.* [GKB*22], we additionally created three scenes where all camera centres are located on a single plane. Two scenes contain mirrors, while the surface in the third scene is near-specular. The default parameters for scenes without manual annotations that are provided by the authors were used for comparison. We also experimented with providing one or multiple ground-truth masks to their approach, but we found that providing no masks consistently produced the best results.

A qualitative comparison between NeRFReN and our approach is shown in Figure 6. It can be seen that our approach is able to better re-construct details in the near-specular region.

Table 1: *Quantitative comparison of our approach against NeRF baselines and recent works on both synthetic multi-mirror scenes and scenes with near-specular surfaces. Metrics are averaged over all test images and across all scenes. Best and second-best results are highlighted.*

	Multi-mirror			Near-specular surface		
Full images	PSNR ↑	SSIM ↑	LPIPS ↓	PSNR ↑	SSIM ↑	LPIPS ↓
NeRF [MST*21]	26.01 ± 5.09	0.800 ± 0.099	0.374 ± 0.107	32.18 ± 3.01	0.883 ± 0.006	0.285 ± 0.045
Mip-NeRF 360 [BMV*22]	25.69 ± 6.28	0.774 ± 0.128	0.410 ± 0.115	32.33 ± 2.83	0.869 ± 0.004	0.342 ± 0.015
Ref-NeRF [VHM*22]	25.21 ± 6.00	0.778 ± 0.108	0.425 ± 0.099	31.75 ± 4.18	0.853 ± 0.010	0.379 ± 0.022
MS-NeRF [YQCR23]	28.27 ± 5.98	0.812 ± 0.121	0.379 ± 0.126	32.29 ± 2.79	0.857 ± 0.011	0.401 ± 0.028
Mirror-NeRF [ZBC*23]	21.83 ± 4.54	0.719 ± 0.082	0.515 ± 0.125	25.38 ± 1.30	0.799 ± 0.035	0.485 ± 0.052
Ours	31.38 ± 3.70	0.868 ± 0.051	0.295 ± 0.094	33.35 ± 0.70	0.892 ± 0.014	0.275 ± 0.014
Mirror regions						
NeRF [MST*21]	29.49 ± 3.28	0.949 ± 0.018	0.065 ± 0.024	36.83 ± 4.96	0.986 ± 0.011	0.029 ± 0.010
Mip-NeRF 360 [BMV*22]	29.85 ± 3.39	0.950 ± 0.017	0.064 ± 0.023	37.29 ± 5.29	0.985 ± 0.013	0.036 ± 0.011
Ref-NeRF [VHM*22]	29.97 ± 3.67	0.954 ± 0.017	0.064 ± 0.024	39.89 ± 5.87	0.989 ± 0.011	0.026 ± 0.010
MS-NeRF [YQCR23]	33.81 ± 4.26	0.967 ± 0.017	0.056 ± 0.027	41.75 ± 7.68	0.988 ± 0.013	0.026 ± 0.017
Mirror-NeRF [ZBC*23]	26.58 ± 3.79	0.937 ± 0.023	0.075 ± 0.030	32.25 ± 1.32	0.977 ± 0.004	0.044 ± 0.004
Ours	38.97 ± 3.28	0.984 ± 0.008	0.031 ± 0.019	39.67 ± 1.67	0.992 ± 0.002	0.024 ± 0.002



(a) NeRF (b) Mip-NeRF 360 (c) Ref-NeRF

(d) MS-NeRF (e) Mirror-NeRF (f) NeRFReN

(g) Ours (h) Ground Truth

Figure 6: *Results of different methods (a–f) compared to ours (g) on a test view of one of the forward facing scenes that contains a near-specular surface. (h) Shows the ground truth test image.*

The quantitative comparison in Table 2 additionally shows results of other approaches. While our approach is not reaching the highest scores when evaluating on the whole images, the evaluation on mirror regions shows that TraM-NeRF is on par with the NeRF baseline on perceptual image metrics. This excellent performance of the baseline methods on the forward-facing scenes can be explained by the fact that this scenario does not impose multi-view inconsistencies, which are difficult to resolve using the original NeRF formulation.

Table 2: *Quantitative results on the three forward facing scenes. Best and second-best results are highlighted.*

Full images	PSNR ↑	SSIM ↑	LPIPS ↓
NeRF	36.33 ± 1.54	0.920 ± 0.013	0.213 ± 0.021
Mip-NeRF 360	35.60 ± 1.79	0.888 ± 0.022	0.325 ± 0.063
Ref-NeRF	35.51 ± 1.68	0.881 ± 0.022	0.344 ± 0.058
NeRFReN	27.38 ± 8.43	0.817 ± 0.062	0.492 ± 0.127
MS-NeRF	34.86 ± 1.30	0.868 ± 0.013	0.387 ± 0.013
Mirror-NeRF	27.50 ± 0.93	0.813 ± 0.010	0.535 ± 0.013
Ours	35.82 ± 1.24	0.911 ± 0.010	0.241 ± 0.019
Mirror regions			
NeRF	44.28 ± 3.11	0.994 ± 0.004	0.018 ± 0.005
Mip-NeRF 360	44.01 ± 4.78	0.990 ± 0.006	0.031 ± 0.009
Ref-NeRF	44.67 ± 4.67	0.991 ± 0.006	0.031 ± 0.008
NeRFReN	33.25 ± 9.82	0.978 ± 0.018	0.054 ± 0.015
MS-NeRF	44.37 ± 4.93	0.992 ± 0.005	0.025 ± 0.012
Mirror-NeRF	30.83 ± 3.42	0.966 ± 0.006	0.066 ± 0.015
Ours	43.54 ± 2.06	0.994 ± 0.002	0.018 ± 0.003

4.4. Real-world scenes

In addition to the synthetic results, we also tested the performance on real-world scenes with both planar and cylindrical mirror surfaces. Quantitative results in Table 3 show that our method is able to significantly improve the quality of the re-constructions in mirror regions while still being competitive regarding the overall reconstruction quality. Rows 5 and 6 in Figure 5 also show a visible improvement of the quality of mirror regions in real-world scenes,

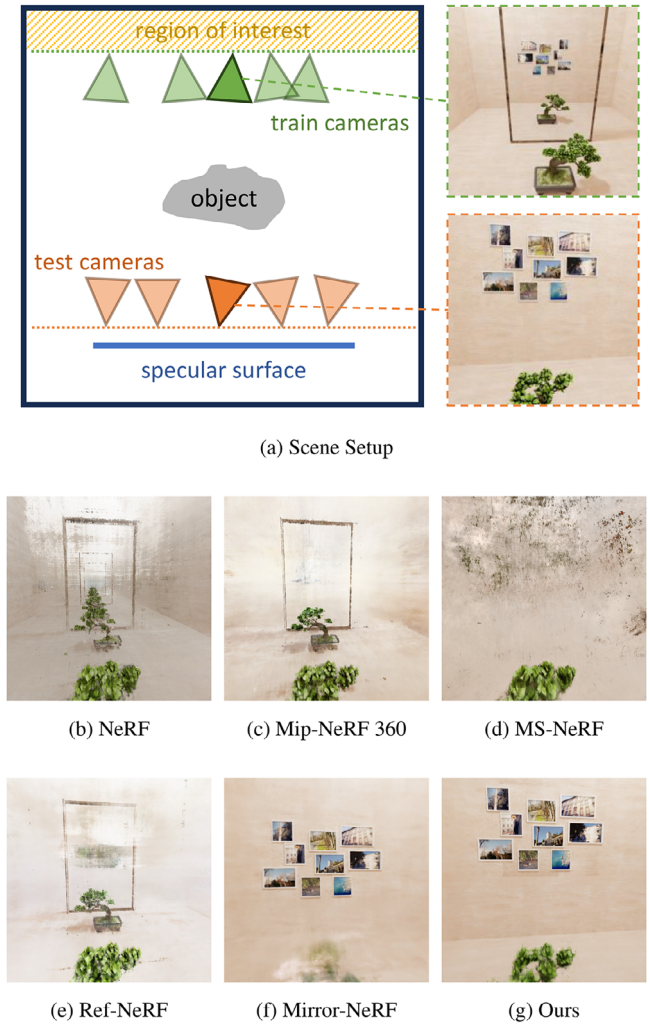
Table 3: *Comparison of our method on real-world data of both planar and non-planar mirror surfaces. Best and second-best results are highlighted.*

Full images	PSNR ↑	SSIM ↑	LPIPS ↓
NeRF	25.67 ± 2.39	0.809 ± 0.016	0.406 ± 0.065
Mip-NeRF 360	26.98 ± 2.24	0.864 ± 0.018	0.286 ± 0.048
Ref-NeRF	26.15 ± 1.87	0.815 ± 0.015	0.390 ± 0.040
MS-NeRF	26.30 ± 2.80	0.818 ± 0.064	0.390 ± 0.110
Mirror-NeRF	18.06 ± 5.12	0.607 ± 0.167	0.533 ± 0.133
Ours	27.53 ± 1.56	0.845 ± 0.012	0.362 ± 0.055
Mirror regions			
NeRF	29.54 ± 3.07	0.943 ± 0.029	0.077 ± 0.044
Mip-NeRF 360	29.85 ± 2.77	0.947 ± 0.028	0.070 ± 0.043
Ref-NeRF	29.57 ± 3.34	0.939 ± 0.034	0.081 ± 0.050
MS-NeRF	31.59 ± 2.02	0.952 ± 0.023	0.074 ± 0.045
Mirror-NeRF	24.50 ± 0.50	0.923 ± 0.023	0.086 ± 0.040
Ours	33.52 ± 2.28	0.962 ± 0.015	0.061 ± 0.035

both for planar (row 5) and cylindrical (row 6) mirrors. In the scene with the cylindrical mirror surface, it can also be seen that our method re-constructs more highlights of the glossy surface of the plate (next to the peeler and between the candle and the small metal rod), which could be due to the increased number of observations our method intrinsically provides for these regions, as they are often only visible in the reflection and not by primary camera rays. Even though we experimented with multiple hyperparameter configurations and disabled the plane consistency loss for scenes with non-planar mirror surfaces, Mirror-NeRF was not able to converge to a plausible solution for the mirror regions.

4.5. Re-construction of indirectly observed regions

One of the advantages of our approach compared to works that model reflections as separate radiance fields [GKB*22, YQCR23] is that information contained in the reflection improves the re-construction quality of the regions the reflected ray passes through. To visualize and quantify this, we created an experiment where certain regions of the scene not visible to primary camera rays in any of the training images. The cameras used to generate the test images are then chosen to cover the regions not seen in the training. An example of this setup and results are shown in Figure 7. Because the other approaches do not model a change in ray directions, they only extrapolate directly observed scene elements in the unseen regions. The periodicity in the positional encoding seems to lead to a copy of the observed scene in (b) and (d), while (c) produces noise in the respective regions. Our approach (f) on the other hand re-constructs high-frequency details that were observable in the reflection of the mirror. While Mirror-NeRF (e) is also modelling the reflection explicitly, it struggled to place the textured region on the wall at the correct location. One possibility for this erroneous result is that the re-construction of that region is not constrained enough due to the design of the experiment. Many choices for the mirror normal may lead to a plausible re-construction of the training data and Mirror-NeRF does not use additional cues to optimize towards the correct normal. In our approach, the normal is explicitly given by the annotation and thus automatically leads to a correct ray reflection.



(a) Scene Setup

(b) NeRF (c) Mip-NeRF 360 (d) MS-NeRF

(e) Ref-NeRF (f) Mirror-NeRF (g) Ours

Figure 7: *Experiment with indirectly observed regions. (a) Schematic top view of the scene. Training cameras (green) are placed on a single plane, oriented towards a mirror (blue) that reflects light rays from an unseen region of interest (yellow) towards the training cameras. The test cameras (orange) are placed on a second plane and can directly observe the region of interest. (b)–(g) show the resulting novel views from test cameras produced by previous approaches compared to ours. The captions also report the PSNR averaged over all test views.*

4.6. Ablation studies

We conducted additional experiments to validate some of the design choices of our approach.

4.6.1. *Annotation robustness*

To validate our claim that annotations in a small subset of images are sufficient to accurately define the position of a rectangular planar mirror in 3D space, we perturbed the ideal annotation locations extracted from Blender by different amounts in screen space. More specifically, we projected the ground truth 3D corners

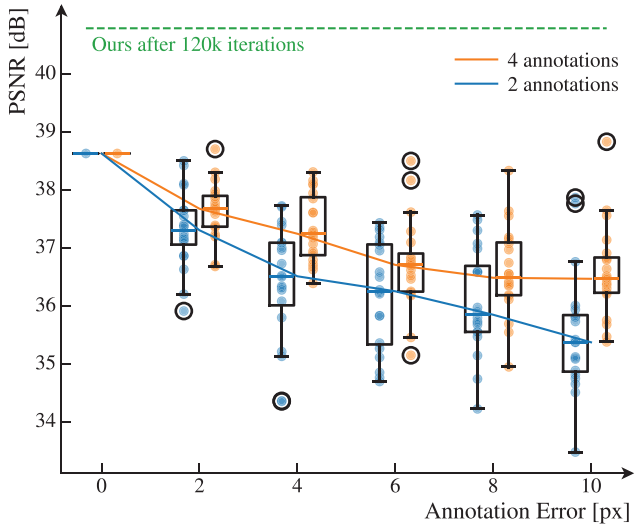


Figure 8: *PSNR in the mirror regions evaluated on one of the scenes from our dataset after 10,000 training iterations, using two (blue) and four (orange) perfectly annotated images to compute the 3D location of the mirror surface. Training is then performed 20 times with annotations perturbed in screen space by randomly sampled errors of fixed magnitude. The boxes encompass lower and upper quartiles with the straight line showing the median value, lower and upper fences, as well as outliers marked with circles.*

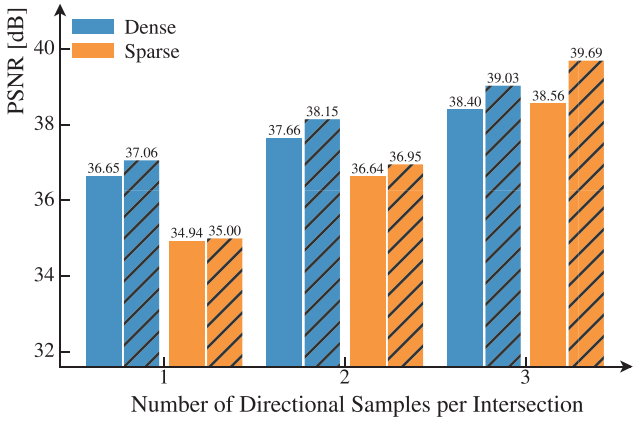


Figure 9: *Comparison of the effect of different sampling strategies used during training on the final result. Experiments were performed on one of the synthetic scenes with a near-perfect specular mirror. The PSNR is measured in the mirror regions of the test dataset after training using our proposed dense sampling strategy (blue) and the standard sparse sampling strategy (orange). Bars without hatches indicate that the same number of secondary rays was used as in the training, while hatched bars show the results when using 50 secondary rays for the final renderings.*

of the mirror surface into the images and introduced a fixed error $\epsilon \in \{2, 4, 6, 8, 10\}$ to the projected points by randomly sampling a new screen space point on a circle around the ideal point with radius ϵ. These perturbed annotations are then used to determine the 3D position of the mirror for the training, as described in Section 3.3. We ran this process for the cases of annotating a subset of two and four images, respectively, and repeated the experiment 20 times for each choice of ϵ, amounting to 100 models in total with non-zero error.

Figure 8 shows the resulting PSNR of the re-construction with pixel-exact annotations and different levels of noise on a single scene in our dataset. Our method remains robust even under severe noise, only dropping by around 1.5 dB in re-construction quality after 10,000 iterations when each annotation is perturbed by 10 px. The experiments also exhibit some outliers that even exceed the quality of the perfectly annotated case, which we assume to be artifacts from run-by-run variance in conjunction with stopping the optimization and evaluating the models mid-training.

4.6.2. *Modified ray sampling*

As motivated in Section 3.2, we draw the microfacet normals independently for each sampling location along the main ray (dense) instead of generating multiple rays at the surface intersection and choosing sample points along these rays (sparse). We ran an experiment to compare the re-construction quality of these variants. Figure 9 shows that our proposed dense estimator variant is advantageous when using the same or double the amount of rays compared to the number of primary rays. Only when increasing the number of secondary rays by significantly larger factors, the sparse sampling shows an increase in re-construction quality compared to the dense sampling. However, this also comes at the cost of an increased amount of network evaluations scaling linearly by this factor for each pixel in the training data belonging to a mirror.

To reduce the variance in the re-construction results, we also show the quality when the number of directional samples after training to 50 rays per surface intersection. This indicates that the lower quality of the sparse variant is not due to high variance in the final rendering but instead is caused by a lower quality scene model resulting from the training.

4.6.3. *Incorrect roughness values*

Figure 10 visualizes the effect of different choices of the roughness value α during training (a)–(e) and the corresponding ground truth view (f). The correct roughness value of the surface is $\alpha = 0.017$ and the resulting rendering of our method is shown in (c). While our proposed sampling strategy introduces a bias in the final renderings, our method produces plausible results for each of the choices of α and remains stable during training, even in the presence of inconsistent observations caused by deviations from the correct roughness.

4.7. Limitations

While TraM-NeRF achieves promising results for novel view synthesis, it also has some limitations that require further attention. In the context of generating novel views, our model inherits NeRF's limitations in extrapolating effectively in areas with insufficient input image coverage, leading to reduced performance. We observed these hallucinations in parts of the scene that are concealed behind mirrors in a majority of training images. Additionally, our estimator can overestimate density and transmittance when the

(a) $\alpha = 0.000$ (b) $\alpha = 0.008$ (c) $\alpha = 0.017$ (d) $\alpha = 0.025$ (e) $\alpha = 0.338$ (f) Ground Truth

Figure 10: *Results produced by our method when trained with different choices for the surface roughness α used during training (a–e) on one of the synthetic scenes. A rendering with correctly chosen roughness is shown in (c), whereas (f) shows the ground truth test image.*

assumption of a narrow spread of light directions does not hold. In this case, object reflections can appear larger than expected. Regarding this, it would be interesting to further investigate the effect of NeRF formulations that consider the integration over a region around the sampling point [BMT*21, IMWB23, BMV*23]. Here, supporting rough reflections would require a non-trivial extension of the idea proposed by Barron *et al.* [BMT*21], as the involved cones will be additionally affected in an anisotropic manner by the BRDF close to their boundaries. Moreover, the current estimator implementation is limited to a single ray bounce in case of reflections with non-zero roughness, assumes that reflective surfaces have no diffuse component and that the normal is close to the analytical normal of the geometry model.

5. Conclusions

We presented TraM-NeRF, an extension of NeRF that effectively models mirror-like surfaces, accurately capturing high-frequency reflections within a single scene representation. By introducing a transmittance-aware variant of the rendering equation for explicit reflection modelling as well as efficient sampling techniques, we are able to reduce the number of network evaluations during ray tracing without increasing the variance. In a qualitative and quantitative evaluation, we demonstrated that our techniques outperform previous methods in challenging scenes with single and multiple mirror-like surfaces on both synthetic and real-world data.

Acknowledgements

This work has been funded by the DFG project KL 1142/11-2 (DFG Research Unit FOR 2535 Anticipating Human Behaviour), and additionally by the Federal Ministry of Education and Research of Germany and the state of North-Rhine Westphalia as part of the Lamarr-Institute for Machine Learning and Artificial Intelligence and by the Federal Ministry of Education and Research under Grant No. 01IS22094E WEST-AI.

Open access funding enabled and organized by Projekt DEAL.

Conflicts of Interest

All authors declare that they have no conflict of interest.

References

[BBJ*21] Boss M., Braun R., Jampani V., Barron J. T., Liu C., Lensch H.: NeRD: Neural reflectance decomposition from image collections. In *IEEE International Conference on Computer Vision (ICCV)* (2021), pp. 12684–12694.

[BEK*22] Boss M., Engelhardt A., Kar A., Li Y., Sun D., Barron J., Lensch H., Jampani V.: SAMURAI: Shape and material from unconstrained real-world arbitrary image collections. In *Advances in Neural Information Processing Systems (NeurIPS)* (2022), vol. *35*, pp. 26389–26403.

[BMT*21] Barron J. T., Mildenhall B., Tancik M., Hedman P., Martin-Brualla R., Srinivasan P. P.: Mip-NeRF: A multiscale representation for anti-aliasing neural radiance fields. In *IEEE International Conference on Computer Vision (ICCV)* (2021), pp. 5855–5864.

[BMV*22] Barron J. T., Mildenhall B., Verbin D., Srinivasan P. P., Hedman P.: Mip-NeRF 360: Unbounded anti-aliased neural radiance fields. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* (2022), pp. 5470–5479.

[BMV*23] Barron J. T., Mildenhall B., Verbin D., Srinivasan P. P., Hedman P.: Zip-NeRF: Anti-aliased grid-based neural radiance fields. In *IEEE International Conference on Computer Vision (ICCV)* (2023), pp. 19697–19705.

[BXS*20a] Bi S., Xu Z., Srinivasan P., Mildenhall B., Sunkavalli K., Hašan M., Hold-Geoffroy Y., Kriegman D., Ramamoorthi R.: Neural reflectance fields for appearance acquisition. *arXiv preprint arXiv:2008.03824* (2020).

[BXS*20b] Bi S., Xu Z., Sunkavalli K., Hašan M., Hold-Geoffroy Y., Kriegman D., Ramamoorthi R.: Deep reflectance volumes: Relightable reconstructions from multi-view photometric images. In *European Conference on Computer Vision (ECCV)* (2020), Springer, pp. 294–311.

[CT82] Cook R. L., Torrance K. E.: A reflectance model for computer graphics. *ACM Transactions on Graphics (TOG) 1* 1 (1982), 7–24.

[CXG*22] Chen A., Xu Z., Geiger A., Yu J., Su H.: TensoRF: Tensorial radiance fields. In *European Conference on Computer Vision (ECCV)* (2022), pp. 333–350.

[CZL*22] Chen X., Zhang Q., Li X., Chen Y., Feng Y., Wang X., Wang J.: Hallucinated neural radiance fields in the wild. In

IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) (2022), pp. 12943–12952.

[DB23] Dupuy J., Benyoub A.: Sampling visible GGX normals with spherical caps. *Computer Graphics Forum (CGF) 42* (2023), e14867.

[DM21] Dröge H., Moeller M.: Learning or modelling? An analysis of single image segmentation based on scribble information. In *IEEE International Conference on Image Processing (ICIP)* (2021), pp. 2274–2278.

[FKMW*23] Fridovich-Keil S., Meanti G., Warburg F. R., Recht B., Kanazawa A.: K-Planes: Explicit radiance fields in space, time, and appearance. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* (2023), pp. 12479–12488.

[FKYT*22] Fridovich-Keil S., Yu A., Tancik M., Chen Q., Recht B., Kanazawa A.: Plenoxels: Radiance fields without neural networks. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* (2022), pp. 5501–5510.

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

[GHZ*23] Ge W., Hu T., Zhao H., Liu S., Chen Y.-C.: Ref-NeuS: Ambiguity-reduced neural implicit surface learning for multi-view reconstruction with reflection. In *IEEE International Conference on Computer Vision (ICCV)* (2023), pp. 4251–4260.

[GKB*22] Guo Y.-C., Kang D., Bao L., He Y., Zhang S.-H.: NeRFReN: Neural radiance fields with reflections. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* (2022), pp. 18409–18418.

[GKJ*21] Garbin S. J., Kowalski M., Johnson M., Shotton J., Valentin J.: FastNeRF: High-fidelity neural rendering at 200fps. In *IEEE International Conference on Computer Vision (ICCV)* (2021), pp. 14346–14355.

[Hei18] Heitz E.: Sampling the GGX distribution of visible normals. *Journal of Computer Graphics Techniques (JCGT) 7*, 4 (2018), 1–13.

[HZF*22] Huang X., Zhang Q., Feng Y., Li H., Wang X., Wang Q.: HDR-NeRF: High dynamic range neural radiance fields. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* (2022), pp. 18398–18408.

[IMWB23] Isaac-Medina B. K., Willcocks C. G., Breckon T. P.: Exact-NeRF: An exploration of a precise volumetric parameterization for neural radiance fields. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* (2023), pp. 66–75.

[JLX*23] Jin H., Liu I., Xu P., Zhang X., Han S., Bi S., Zhou X., Xu Z., Su H.: TensoIR: Tensorial inverse rendering. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* (2023), pp. 165–174.

[Kaj86] Kajiya J. T.: The rendering equation. In *Annual Conference on Computer Graphics and Interactive Techniques (SIGGRAPH)* (1986), pp. 143–150.

[Lau94] Laurentini A.: The visual hull concept for silhouette-based image understanding. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI) 16*, 2 (1994), 150–162.

[LCL*23] Liang R., Chen H., Li C., Chen F., Panneer S., Vijaykumar N.: ENVIDR: Implicit differentiable renderer with neural environment lighting. In *IEEE International Conference on Computer Vision (ICCV)* (2023), pp. 79–89.

[LHK*20] Laine S., Hellsten J., Karras T., Seol Y., Lehtinen J., Aila T.: Modular primitives for high-performance differentiable rendering. *ACM Transactions on Graphics (TOG) 39*, 6 (2020), 1–14.

[LLGG23] Li Q., Li F., Guo J., Guo Y.: UHDNeRF: Ultra-high-definition neural radiance fields. In *IEEE International Conference on Computer Vision (ICCV)* (2023), pp. 23097–23108.

[LSS*19] Lombardi S., Simon T., Saragih J., Schwartz G., Lehrmann A., Sheikh Y.: Neural volumes: Learning dynamic renderable volumes from images. *ACM Transactions on Graphics (TOG) 38*, 4 (2019), 1–14.

[LWL*23] Liu Y., Wang P., Lin C., Long X., Wang J., Liu L., Komura T., Wang W.: NeRO: Neural Geometry and BRDF Reconstruction of Reflective Objects from Multiview Images. *ACM Transactions on Graphics (TOG) 42*, 4 (2023), 1–22.

[Max95] Max N.: Optical models for direct volume rendering. *IEEE Transactions on Visualization and Computer Graphics (TVCG) 1*, 2 (1995), 99–108.

[MBRS*21] Martin-Brualla R., Radwan N., Sajjadi M. S., Barron J. T., Dosovitskiy A., Duckworth D.: NeRF in the wild: Neural radiance fields for unconstrained photo collections. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* (2021), pp. 7210–7219.

[MESK22] Müller T., Evans A., Schied C., Keller A.: Instant neural graphics primitives with a multiresolution hash encoding. *ACM Transactions on Graphics (TOG) 41*, 4 (2022), 102:1–102:15.

[MHMB*22] Mildenhall B., Hedman P., Martin-Brualla R., Srinivasan P. P., Barron J. T.: NeRF in the dark: High dynamic range view synthesis from noisy raw images. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* (2022), pp. 16190–16199.

[MST*21] Mildenhall B., Srinivasan P. P., Tancik M., Barron J. T., Ramamoorthi R., Ng R.: NeRF: Representing scenes as neural radiance fields for view synthesis. *Communications of the ACM 65*, 1 (2021), 99–106.

[MT97] Müller T., Trumbore B.: Fast, minimum storage ray-triangle intersection. *Journal of Graphics Tools 2*, 1 (1997), 21–28.

[MVKFK23] Mai A., Verbin D., Kuester F., Fridovich-Keil S.: Neural microfacet fields for inverse rendering. In *IEEE International Conference on Computer Vision (ICCV)* (2023), pp. 408–418.

[NMOG20] Niemeyer M., Mescheder L., Oechsle M., Geiger A.: Differentiable volumetric rendering: Learning implicit 3D representations without 3D supervision. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* (2020), pp. 3504–3515.

[PGM*19] Paszke A., Gross S., Massa F., Lerer A., Bradbury J., Chanan G., Killeen T., Lin Z., Gimelshein N., Antiga L., Desmaison A., Kopf A., Yang E., DeVito Z., Raison M., Tejani A., Chilamkurthy S., Steiner B., Fang L., Bai J., Chintala S.: PyTorch: An imperative style, high-performance deep learning library. In *Advances in Neural Information Processing Systems (NeurIPS)* (2019), pp. 8024–8035.

[RPLG21] Reiser C., Peng S., Liao Y., Geiger A.: KiloNeRF: Speeding up neural radiance fields with thousands of tiny MLPs. In *IEEE International Conference on Computer Vision (ICCV)* (2021), pp. 14335–14345.

[SDZ*21] Srinivasan P. P., Deng B., Zhang X., Tancik M., Mildenhall B., Barron J. T.: NeRV: Neural reflectance and visibility fields for relighting and view synthesis. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* (2021), pp. 7495–7504.

[SF16] Schönberger J. L., Frahm J.-M.: Structure-from-motion revisited. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* (2016), pp. 4104–4113.

[SZW19] Sitzmann V., Zollhöfer M., Wetzstein G.: Scene representation networks: Continuous 3D-structure-aware neural scene representations. In *Advances in Neural Information Processing Systems (NeurIPS)* (2019), vol. *32*.

[VHM*22] Verbin D., Hedman P., Mildenhall B., Zickler T., Barron J. T., Srinivasan P. P.: Ref-NeRF: Structured view-dependent appearance for neural radiance fields. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* (2022), IEEE, pp. 5481–5490.

[WBSS04] Wang Z., Bovik A. C., Sheikh H. R., Simoncelli E. P.: Image quality assessment: From error visibility to structural similarity. *IEEE Transactions on Image Processing (TIP) 13*, 4 (2004), 600–612.

[WHL*23] Wu H., Hu Z., Li L., Zhang Y., Fan C., Yu X.: NeFII: Inverse rendering for reflectance decomposition with near-field indirect illumination. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* (2023), pp. 4295–4304.

[WLS*22] Wang Z., Li L., Shen Z., Shen L., Bo L.: 4K-NeRF: High fidelity neural radiance fields at ultra high resolutions. *arXiv preprint arXiv:2212.04701* (2022).

[WMLT07] Walter B., Marschner S. R., Li H., Torrance K. E.: Microfacet models for refraction through rough surfaces. In *Eurographics Symposium on Rendering (EGSR)* (2007), pp. 195–206.

[WWG*22] Wang C., Wu X., Guo Y.-C., Zhang S.-H., Tai Y.-W., Hu S.-M.: NeRF-SR: High quality neural radiance fields using supersampling. In *ACM International Conference on Multimedia* (2022), pp. 6445–6454.

[YQCR23] Yin Z.-X., Qiu J., Cheng M.-M., Ren B.: Multi-space neural radiance fields. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* (2023), pp. 12407–12416.

[ZBC*23] Zeng J., Bao C., Chen R., Dong Z., Zhang G., Bao H., Cui Z.: Mirror-NeRF: Learning neural radiance fields for mirrors with whitted-style ray tracing. In *Proceedings of the 31st ACM International Conference on Multimedia* (2023), pp. 4606–4615.

[ZIE*18] Zhang R., Isola P., Efros A. A., Shechtman E., Wang O.: The unreasonable effectiveness of deep features as a perceptual metric. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* (2018), pp. 586–595.

[ZLW*21] Zhang K., Luan F., Wang Q., Bala K., Snavely N.: PhySG: Inverse rendering with spherical Gaussians for physics-based material editing and relighting. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* (2021), pp. 5453–5462.

[ZRSK20] Zhang K., Riegler G., Snavely N., Koltun V.: NeRF++: Analyzing and improving neural radiance fields. *arXiv preprint arXiv:2010.07492* (2020).

[ZSD*21] Zhang X., Srinivasan P. P., Deng B., Debevec P., Freeman W. T., Barron J. T.: NeRFactor: Neural factorization of shape and reflectance under an unknown illumination. *ACM Transactions on Graphics (TOG) 40*, 6 (2021), 1–18.

[ZSH*22] Zhang Y., Sun J., He X., Fu H., Jia R., Zhou X.: Modeling indirect illumination for inverse rendering. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* (2022), pp. 18643–18652.

[ZXY*23] Zhang Y., Xu T., Yu J., Ye Y., Jing Y., Wang J., Yu J., Yang W.: NeMF: Inverse volume rendering with neural microflake field. In *IEEE International Conference on Computer Vision (ICCV)* (2023), pp. 22919–22929.

Supporting Information

Additional supporting information may be found online in the Supporting Information section at the end of the article.

Video S1