SHINOBI: *Sh*ape and *I*llumination using *N*eural *O*bject Decomposition via *B*RDF Optimization *I*n-the-wild

Andreas Engelhardt[†]
University of Tübingen

Amit Raj
Google Research

Mark Boss[*]
Unity

Yunzhi Zhang[†]
Stanford University

Abhishek Kar
Google Research

Yuanzhen Li
Google Research

Deqing Sun
Google Research

Ricardo Martin Brualla
Google Research

Jonathan T. Barron
Google Research

Hendrik P. A. Lensch
University of Tübingen

Varun Jampani[*]
Google Research

Figure 1. **Object reconstruction using SHINOBI.** SHINOBI decomposes challenging in-the-wild image collections into shape, material and illumination using a neural field representation while also optimizing camera parameters. Visit the project page at Project page: https://shinobi.aengelhardt.com

Abstract

We present SHINOBI, an end-to-end framework for the reconstruction of shape, material, and illumination from object images captured with varying lighting, pose, and background. Inverse rendering of an object based on unconstrained image collections is a long-standing challenge in computer vision and graphics and requires a joint optimization over shape, radiance, and pose. We show that an implicit shape representation based on a multi-resolution hash encoding enables faster and robust shape reconstruction with joint camera alignment optimization that outperforms prior work. Further, to enable the editing of illumination and object reflectance (i.e. material) we jointly optimize BRDF and illumination together with the object's shape. Our method is class-agnostic and works on in-the-wild image collections of objects to produce relightable 3D assets for several use cases such as AR/VR, movies, games, etc.

1. Introduction

We present a category-agnostic technique to jointly reconstruct 3D shape and material properties of objects from unconstrained in-the-wild image collections. This data regime poses multiple challenges as images are captured in different environments using a variety of devices resulting in varying backgrounds, illumination, camera poses, and intrinsics. In addition, camera baselines tend to be large. Fig. 1 (left) shows examples from an input image set. Many graphics

[*]Current affiliation is Stability AI.
[†]Work done during a Student Researcher position at Google.

applications in AR/VR, games, and movies depend on high-quality 3D assets of real-world objects. Physically based materials are essential to integrate objects into new environments. The conventional acquisition involves laborious tasks like 3D modeling, texture painting, and light calibration or use controlled setups [6, 47] that are hard to scale. It is easier to obtain casually captured images from smartphones or image collections from the internet for a large number of objects.

Conventional structure-from-motion techniques like COLMAP [54, 55] fail to reconstruct image collections under these challenging circumstances [14, 28]. Despite constraining the correspondences to lie within object bounds, specifically in the context of the NAVI [28] in-the-wild scenes, less than half of the views are registered on average with half the scenes failing completely. Consequently, we observe that camera pose optimization has the largest impact on the reconstruction quality in this setting. Many existing works on shape and material estimation [6, 13, 58, 67, 69, 74, 77] assume constant camera intrinsics and initialization of camera poses close to the true poses. We support 360° multiview data with a rough quadrant-based pose initialization with poses potentially far from the ground truth, as in SAMURAI [14] and NeRS [71]. For challenging data this can be annotated in only a few minutes per image collection. Even though in SAMURAI [14], camera poses can be initialized from very coarse directions slight offsets often lead to overly smooth textures and shapes in the final reconstructions. Further, existing methods for material decomposition with camera pose optimization are slow, often running more than 12 hours on a single object [14, 32]. In contrast, we propose a pipeline based on multiresolution hash grids [45] which allows us to process more rays in a shorter time during optimization. Using this advantage we are able to improve reconstruction quality compared to SAMURAI while still keeping a competitive run-time (Tab. 1).

Naive integration of multi-resolution hash grids is not well suited to camera pose estimation due to discontinuities in the gradients with respect to the input positions. We propose several components that work together to stabilize the camera pose optimization and encourage sharp features. The key distinguishing features of SHINOBI include:

- *Hybrid Multiresolution Hash Encoding with level annealing.* We combine the multiresolution hash-based encoding [45] with regular Fourier feature transformation of the input coordinates to regularize the low-frequency gradient propagation. This makes the optimization significantly more robust while only adding a small overhead. A similar approach has been recently proposed by Zhu *et al*. [79] for a different task. We show that it is also beneficial for camera pose optimization.
- *Camera multiplex constraint.* We modify the camera parameterization of SAMURAI to avoid over-parameterization of the camera rotations. Furthermore, we constrain the camera optimization with a projection-based loss to enforce consistency over the camera proposals inside a multiplex which further helps to smooth the optimization in the initial phase.
- *Per-view importance weighting.* We propose a per-view importance weighting to leverage the important observation that some views are more useful for optimization than others. Specifically, we use well-working cameras to anchor the reconstruction during the optimization.
- *Patch-based alignment losses.* SHINOBI proposes a novel patch level loss to aid in camera alignment and additionally introduces a silhouette loss inspired by Lensch *et al*. [33] for better image to 3D alignment.

Experiments on NAVI [28] in-the-wild datasets demonstrate better view synthesis and relighting results with SHINOBI compared to existing works with a reduced run-time. Compared to SAMURAI the results look sharper and the average runtime is cut in half. Fig. 1 (right) shows some sample application results with 3D assets generated by SHINOBI . Our representation enables editing of appearance parameters, illumination and based on the mesh extraction also shape, facilitating various tasks in a downstream graphics pipeline.

2. Related works

Neural fields have emerged as a popular technique of late to encode spatial information in the network weights of *e.g*. an MLP, which can be retrieved by simply querying the coordinates [16, 43, 50, 59]. Works like NeRF [44] leverage this neural volume rendering to achieve photo-realistic view synthesis results with view-dependent appearance variations. Rapid research in neural fields followed, which alternated the surface representations [49, 61–63, 65, 68], allowed reconstruction from sparse data [8, 27, 40, 48, 52, 60, 66], enabled extraction of 3D geometry and materials [12, 32, 46, 71], or enabled relighting of scenes [5, 12–14, 37, 41, 69]. However, most prior works rely on pose information extracted from COLMAP [54, 55], which can be inaccurate or completely fail in complex settings or sparse data regimes. SHINOBI is independent of any *pose reconstruction that relies on feature matching* and robust to very *coarse initialization*.

Instant Neural Graphics Primitives (I-NGP) [45] is a popular geometric representation that enables fast optimization with improved memory utilization by using an encoding scheme based on multi-resolution hash tables. Despite the improvement in speed, I-NGP suffers from discontinuous and oscillating gradient flow through the hash-based encoding, which complicates camera pose optimization [26, 79, 79]. To enable reconstruction with camera pose fine-tuning using hash grids, Heo *et al*. [26] propose a modification to the interpolation weighting, BAA-NGP [39] dynamically replicates low-resolution features and CAMP [51] pairs a robust sampling scheme [4] with camera preconditioning. These methods however are sensitive to camera initialization and

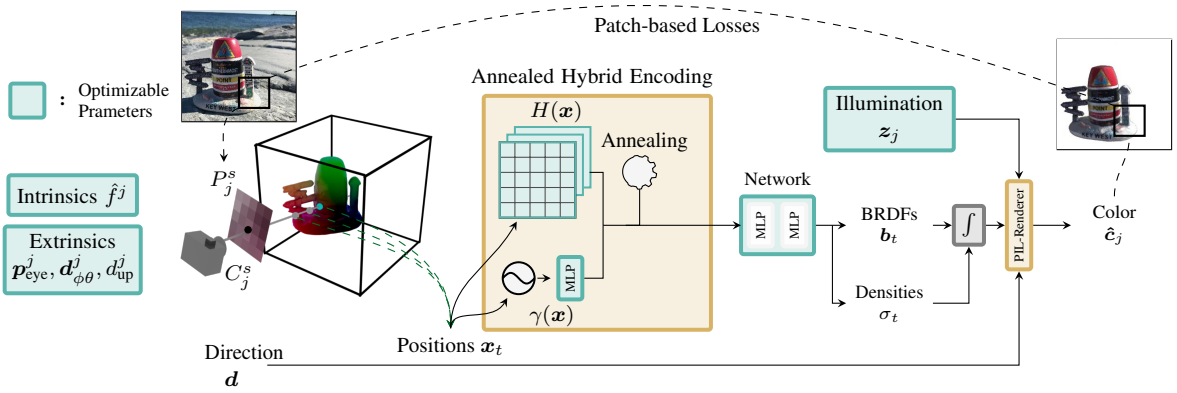


Figure 2. **The SHINOBI pipeline.** Two resolution annealed encoding branches, the multiresolution hash grid $H(\boldsymbol{x})$ and the Fourier embedding $\gamma(\boldsymbol{x})$ are used to learn a neural volume conditioned on the input coordinates. This enables robust optimization of camera parameters jointly with the shape, material and illumination.

lighting conditions. In contrast to these works, SHINOBI is able to reconstruct consistent objects from images captured under *varying illuminations and backgrounds* besides supporting *coarser poses.*

Joint camera and shape estimation is a highly ambiguous task, traditionally relying on accurate poses for precise shape reconstruction and vice versa. Often techniques rely on correspondences across images to estimate camera poses [54, 55]. Recent approaches integrate camera calibration with neural volume training; SCNeRF [29] and NopeNeRF [8] use correspondences and monocular depth images, respectively. Other recent methods rely on rough initialization of the camera, global alignment, or a template shape for joint optimization [15, 38, 64, 71]. Other methods use transformer-based models[21] to predict the initial pose from image collection [57, 72]. In comparison, SHINOBI works on *unconstrained image collections*, including various camera parameters and object environments, where existing methods struggle to generalize or require additional input data like depth.

BRDF and illumination estimation is a challenging and ambiguous problem. Casual BRDF estimation enables on-site material acquisition with simple cameras and a co-located camera flash. These techniques often constrain the problem to planar surfaces with either a single shot [2, 9, 18, 25, 34, 53], few-shot [2] or multi-shot [3, 10, 19, 20, 22] captures. Casual capture can also be extended to a joint BRDF and shape reconstruction [5–7, 11, 30, 47, 53, 70], even on entire scenes [35, 56]. Most of these methods, however, require a known active illumination. Recovering a BRDF under unknown passive illumination is significantly more challenging as it requires disentangling the BRDF from the illumination. Recently, neural field-based decomposition achieved decomposition of scenes under varying illumination [12, 13] or fixed illumination [37, 73, 74, 76, 77]. IntrinsicNeRF [69] extends decomposition to larger scenes at the cost of a simplified reflectance model. However, all these approaches require known, near-perfect camera poses, whereas SHINOBI can work with *unposed image collection* to recover per-image illumination.

3. Method

The aim of SHINOBI is to convert 2D image collections into a 3D representation with minimal manual work. The representation includes shape, material parameters and per-view illumination, allowing for view synthesis with relighting.

Problem setup. We define in-the-wild data as a collection of q images $C_j \in \mathbb{R}^{s_j \times 3}; j \in \{1, \dots, q\}$ that show the same object captured with different backgrounds, illuminations and cameras with potentially varying resolutions s_j. In addition, we assume a rough camera initialization. For our experiments we annotate camera pose quadrants as in SAMURAI [14]. Foreground masks can be added if available or automatically generated and might be imperfect at this point. At each point $\boldsymbol{x} \in \mathbb{R}^3$ in the neural volume $\mathcal{V}$, we estimate the BRDF parameters for the Cook-Torrance model [17] $\boldsymbol{b} \in \mathbb{R}^5$ (basecolor $\boldsymbol{b}_c \in \mathbb{R}^3$, metallic $b_m \in \mathbb{R}$, roughness $b_r \in \mathbb{R}$), unit-length surface normal $\boldsymbol{n} \in \mathbb{R}^3$ and volume density $\sigma \in \mathbb{R}$ (Fig. 1). To enable the decomposition we also estimate the latent per-image illumination vectors $\boldsymbol{z}^l_j \in \mathbb{R}^{128}; j \in \{1, \dots, q\}$ [13]. Furthermore, we estimate per-image camera poses and intrinsics. Next, we provide a brief overview of prerequisites: NeRF [44], InstantNGP [45] and SAMURAI [14].

Coordinate-based MLPs and NeRF [44] uses a dense neural network to model a continuous function that takes 3D location $\boldsymbol{x} \in \mathbb{R}^3$ and view direction $\boldsymbol{d} \in \mathbb{R}^3$ and outputs a view-dependent output color $\boldsymbol{c} \in \mathbb{R}^3$ and volume density $\sigma \in \mathbb{R}$. Mildenhall *et al*. [44] overcome the spectral bias of the MLPs by transforming the input coordinates by a second

function; A frequency encoding γ that maps from $\mathbb{R}$ to $\mathbb{R}^{2L}$ [44, 59]:

$$\gamma(\boldsymbol{x}) = (\sin(2^0\pi\boldsymbol{x}), \cos(2^0\pi\boldsymbol{x}), \\ \ldots, \sin(2^{L-1}\pi\boldsymbol{x}), \cos(2^{L-1}\pi\boldsymbol{x})) \tag{1}$$

InstantNGP [45] speed up the NeRF optimization drastically by replacing the MLP-based volume representation by a multiresolution voxel hash grid that is tailored to current GPU hardware. For a hash-size T, grid vertices are indexed by a spatial hash function $h(x) = \left(\bigoplus_{i=1}^{d} x_i\pi_i\right) \mod T$ using large unique prime numbers π_i [45]. At each voxel vertex a d-dimensional embedding is optimized. Instead of the Fourier embedding, the 3D coordinates $\boldsymbol{x}$ are directly used to tri-linearly interpolate between neighboring vertices at each level. The results are concatenated and fed to a MLP to decode the representation. We denote the full encoding function including interpolation and concatenation as $H(\boldsymbol{x})$.

Brief overview of SAMURAI. SAMURAI is a method for joint optimization of 3D shape, BRDF, per-image camera parameters, and illuminations for a given in-the-wild image collection. SAMURAI [14] follows the NeRF idea outlined above but uses the Neural-PIL [13] method for physically-based differentiable rendering. It takes 3D locations as input and outputs volume density and BRDF parameters. An additional GLO (generative latent optimization) embedding models the changes in appearances (due to different illuminations) across images. Neural-PIL [13] introduced the use of per-image latent illumination embedding $\boldsymbol{z}_j^l$ and a specialized illumination pre-integration (PIL) network for fast rendering, which we refer to as 'PIL rendering'. Neural-PIL optimizes a per-image embedding to model image-specific illumination. The rendered output color $\hat{\boldsymbol{c}}$ is equivalent to NeRF's output $\boldsymbol{c}$, but due to the explicit BRDF decomposition and illumination modeling, it enables relighting and material editing. To address the unavailability of accurate camera parameters for in-the-wild images, SAMURAI jointly optimizes camera extrinsics and per-view intrinsics from a very coarse initialization. In addition to a coarse-to-fine annealing [38], this is achieved with a multiplexed optimization scheme where multiple camera proposals per view are kept and weighted according to their performance on the loss over time.

3.1. SHINOBI Optimization with Hash Encoding

We identify misaligned and inconsistent camera poses as the main limiting factor for in-the-wild reconstructions. Joint shape and camera optimization is a severely underdetermined problem. Reconstruction is typically slow and often lacks high-frequency detail in textures and shape. Multiresolution hash grids from Instant-NGP [45] have the potential to speed up the reconstruction while simultaneously allowing for larger ray counts to be processed and thereby improving visual quality and alignment (see Tab. 1). However, the naive replacement of the point encoding with Hash grids reduces the reconstruction quality and robustness of the joint camera and shape optimization.

Hash grids adapt to individual views faster resulting in a noisy shape in the presence of misaligned cameras. As reported previously [26, 36, 39, 79] multi-resolution hash grids with the default linear interpolation backpropagate noisy and discontinuous gradients with respect to the input position. Additionally, the coarse-to-fine scheme from BARF [38] often used for camera fine-tuning cannot be directly transferred to hash grids. Therefore, we propose an approach that makes use of a camera multiplex, adds additional geometrical constraints, and a new encoding scheme to be able to improve both reconstruction speed and quality. Next, we explain each of the components in detail.

Architecture overview. A high-level overview of the SHINOBI architecture is shown in Fig. 2, which follows the skeleton of SAMURAI [14] with the 'PIL renderer' [13]. However, we map the input coordinates $\boldsymbol{x}$ using a new hybrid encoding. The combined embedding is processed by a small MLP like in I-NGP [45] to predict the density σ, and the view and appearance conditioned radiance for a given image patch. We also predict a regular direction-dependent radiance $\tilde{\boldsymbol{c}}$ to stabilize the early training stages as in [12, 14]. The BRDF decoder operates as in SAMURAI [14], expanding the feature representation to the BRDF (base color, metallic, roughness). Per sample, we estimate normal direction from the first order derivative of the density w.r.t. the input position $\frac{\partial\sigma}{\partial\boldsymbol{x}}$. From there the volumetric rendering from NeRF [44] is performed and the shading for the given pixel coordinate is determined using BRDF, normals and the pre-integrated illumination estimated by the NeuralPIL network. See supplementary material for further details on the architecture.

Camera pose initialization and parameterization. Camera pose optimization is a highly non-convex problem and tends to quickly get stuck in local minima. Our initial camera poses are much noisier and feature larger distances between initial and true poses compared to many related works [32, 64]. To combat this, we assume a rough initialization in the form of camera pose quadrants in line with SAMURAI [14] and NeRS [71]. We use a 'lookat + direction' representation for the camera parameters, storing initial values and offsets for an eye position $\boldsymbol{p}_{eye} \in \mathbb{R}^3$, lookat direction $\Delta\boldsymbol{d}_{\phi\theta} \in \mathbb{R}^2$. and up rotation angle $d_{up} \in \mathbb{R}$ as well as the focal length $f \in \mathbb{R}$ per camera. We notice that this removes the overparameterization regarding the rotation component encoded in eye and center position of the regular 'lookat' parameterization. This formulation performs best in our setting also compared to other recently proposed representations [51, 78].

Hybrid positional encoding. We use a hash grid hybrid as coordinate encoding to improve the gradient flow w.r.t. the input coordinates $\boldsymbol{x}$. A Fourier-based coordinate mapping $\gamma(\boldsymbol{x})$

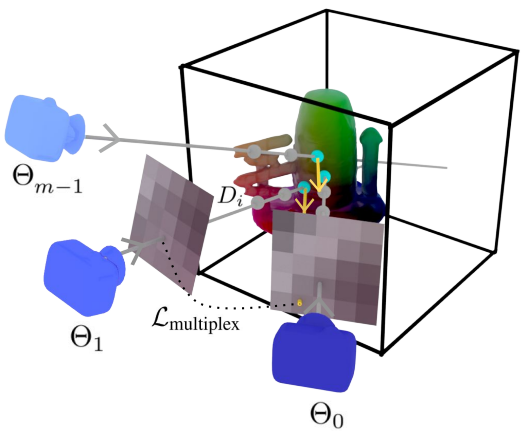


Figure 3. **Constrained camera multiplex.** We optimize multiple camera proposals per image and weight the contribution to the reconstruction according to a camera's performance on the loss. Between cameras of a multiplex we add a projection based regularization: Points from all members are projected into the currently best camera and then compared against a new render to enforce a consistent geometry.

followed by a small MLP generates a base embedding that is concatenated with the output of the multiresolution hash grid $H(\boldsymbol{x})$ resulting in the following formulation of the neural volume $F_{\oplus}\left((H(\boldsymbol{x}), \gamma(\boldsymbol{x}))\right)$. On γ, we apply BARF's [38] Fourier annealing. Similarly, we progressively add resolution levels to the hash grid encoding. Starting with only the features from a low resolution dense grid we increase the weights of the higher resolution levels gradually over time (cf. [36, 39].

Camera multiplexes. An effective way to reduce the chance of camera pose optimization to be stuck in local minima is the camera multiplex [14, 23]. For each image, m cameras are jittered around the initial camera and simultaneously optimized. Over time the worst performing camera is repeatedly faded out until $m = 1$. This process is visualized in Fig. 3. Since we render multiple proposals for a given image anyway, we see an opportunity to further constrain the optimization using projective geometry. Specifically, we project the 2D point sets X_i rendered by the $m-1$ members into the currently highest ranking camera Θ_0 of the multiplex using the estimated depth D_i from the volumetric rendering. Then we render the projected coordinates using Θ_0 and compare the rendered color c_i and alpha values α_i of all cameras in the multiplex to the ones originally rendered at $\Theta_{1\ldots m-1}$.

$$\mathcal{L}_{\text{multiplex}} = \sum_{i=1}^{m-1} \mathcal{L}_{\text{image}}(\boldsymbol{c}_i, F\hat{c}_{\mathcal{V}}\left(P_{i,0}(X_i, D_i, \Theta_i, \Theta_0)\right)) + \mathcal{L}_{\text{mask}}(\boldsymbol{\alpha}_i, F\alpha_{\mathcal{V}}\left(P_{i,0}(X_i, D_i, \Theta_i, \Theta_0)\right)) \quad (2)$$

where $P_{i,0}$ is the perspective warp from image coordinates in camera i to the reference camera. $F_{\mathcal{V}}$ is the rendering function connected to the neural field outputting color $\hat{c}$ and mask value α, respectively. This regularization comes roughly at the cost of adding a camera to the multiplex. Subsampling of X_i can decrease the memory footprint if needed. $\mathcal{L}_{\text{image}}$ and $\mathcal{L}_{\text{mask}}$ are the optimization losses active at the time as outlined in Sec. 3.2. Naturally, this component is only active while there are multiple cameras rendered during the first part of the overall schedule. Used as an additional loss it turns out to be surprisingly effective in constraining the camera optimization and therefore increasing the robustness of the overall optimization. Essentially, we are enforcing a consistent surface to be generated and smooth the optimization landscape around an initial camera pose.

View importance scaling of input images. Not every input might contribute to the reconstruction in the same way and individual views that are not aligned with the current 3D shape might have a negative impact on the overall optimization progress. To improve high-frequency detail in the reconstruction we reduce the impact of potentially misaligned cameras while anchoring the optimization using cameras that work well given the loss. We keep a circular buffer of around 1000 elements with the recent per-image losses. Like in SAMURAI, this is used to re-weigh images in the given collection according to: $\mathcal{L}^{(j)}_{\text{network}} = s_{\text{p}_j}\, \mathcal{L}^{(j)}_{\text{network}}$, where

$$s_{\text{p}_j} = \max\left(\tanh\left(\frac{\mu_l - (\mathcal{L}^{(j)}_{\text{mask}} + \mathcal{L}^{(j)}_{\text{image}})}{\sigma_l}\right) + 1, 1\right), \quad (3)$$

with the mean μ_l and standard deviation σ_l of the loss buffer. This limits the influence of badly aligned camera poses on the shape reconstruction. In addition, we also apply an importance weighting on $\mathcal{L}_{\text{camera}}$ that reduces the gradient magnitude for views that are performing well given the loss history. Specifically, at step t we compute: $\mathcal{L}^{(j)}_{\text{camera}} = s_{\text{q}_{j,t}}\, \mathcal{L}^{(j)}_{\text{camera}}$, with

$$s_{\text{q}_{j,t}} = s_{\text{q}_{j,t-1}} \lambda_p \max\left(\tanh\left(\frac{\mu_l - (\mathcal{L}^{(j)}_{\text{mask}} + \mathcal{L}^{(j)}_{\text{image}})}{\sigma_l}\right) + 1, 1\right) + (1 - \lambda_p) s_{\text{q}_{j,t-1}} \quad (4)$$

In practice, we set the hyperparameter λ_p to 0.05.

3.2. Losses and Optimization

Multiscale patch loss. After a short initial phase of random ray sampling, we render randomly sampled patches of size 16x16 to 32x32. The goal is to constrain the updates and especially the alignment to be consistent on local neighborhoods. Therefore, we add a multi-scale patch loss on the

(a) Reference silhouette (b) Rendered silhouette (c) Loss map

Figure 4. **Our silhouette based alignment loss** penalizes the unaligned pixels given a reference and the rendered gray scale masks.

rendered color $\hat{c}$ which computes a Charbonnier loss at four different resolution levels, by simple bilinear resampling. We weigh each level to compensate for the different pixel counts and enforce the low-resolution version to align first.

Mask losses. We add a silhouette loss $\mathcal{L}_{\text{Silhouette}}$ whenever patch-based sampling is active. Here, we penalize the area between the two silhouettes which can be interpreted as the result of an xor operation on the rendered and input mask [33]. Both masks are filtered using a Gaussian blur where the radius is heuristically chosen based on the patch size. Fig. 4 visualizes how the loss helps with the alignment task. We combine this loss with a regular binary-cross-entropy loss on the mask value as well as a loss enforcing a transparent background.

Regularization losses. To regularize the hash grid encoding we apply a normalized weight decay as proposed in [4] to put a higher penalty on coarser grid levels compared to naive weight decay. Additionally, we apply regularization to the camera poses and normal output. Refer to the supplements for details and the hyperparameters used.

Optimization. In total, we use three optimizers: One ADAM [31] optimizer for the networks, hash grid embeddings and cameras, respectively. The learning rate is decayed exponentially on all optimizers. In addition to the camera representation and constraints mentioned above we use ADAM with the $\beta 1$ value reduced to 0.2 to smooth out the noise in the camera updates. The learning rate is tuned between 1e-3 to 2e-3 depending on scene size. Render resolution is continuously increased over the first half of the optimization while the number of active multiplex cameras is reduced. The direct color optimization is faded to the BRDF optimization and the encoding annealing is performed over the first third of the optimization. Focal length updates and the view importance weighting are delayed until an initial shape has been formed. See the supplementary material for a detailed description and visualization of the optimization scheduling.

Implementation. We implement the multi-resolution hash grid encoding as a custom CUDA extension for Tensorflow [1]. The implementation roughly follows the official CUDA implementation [45]. We enable first- and second-order gradients for the encoding to allow for computing analytical surface normals. The remaining components are implemented in Tensorflow.

Method	PSNR↑		SSIM↑		LPIPS↓		Runtime
	S_C	$\sim S_C$	S_C	$\sim S_C$	S_C	$\sim S_C$	
NeROIC [32]	22.75	21.31	0.91	0.90	0.0984	0.0845	18 hours (4 GPUs)
NeRS [71]	17.92	18.02	0.92	0.93	0.114	0.1098	3 hours (1 GPU)
SAMURAI [14]	25.34	24.61	0.92	0.91	0.0958	0.1054	12 hours (1 GPU)
SHINOBI	**27.69**	**27.79**	**0.94**	**0.94**	**0.0607**	**0.0578**	4 hours (1 GPU)

Table 1. **Metrics for view synthesis on NAVI.** View synthesis metrics are computed over two subsets from all wild-sets depending on the success of COLMAP (S_C / $\sim S_C$). Rendering quality is evaluated on a holdout set of test views. We initialize with the GT poses provided by NAVI [28].

Method	PSNR↑	SSIM↑	Transl.↓	Rot. °↓
w/o Multiplex Consistency Loss	25.80	**0.93**	**0.29**	23.12
w/o Per View Importance	22.43	0.90	0.36	35.10
w/o Coarse-to-fine (annealing)	21.47	0.90	0.37	30.44
w/o Hybrid Encoding	25.31	**0.93**	0.30	23.33
w/o Patch-based Training	20.60	0.89	0.45	41.30
Full	**25.87**	**0.93**	0.30	**22.90**

Table 3. **Ablation study.** Ablating components of our framework results in worse view synthesis and relighting results (averaged over "Keywest" and "School Bus" scenes from NAVI) demonstrating their importance.

4. Experiments

Dataset For evaluations, we use the in-the-wild collections from the NAVI dataset [28] which feature objects captured in diverse environments using multiple mobile devices. High-quality annotated camera poses allow us to ablate and perform quantitative evaluation of our pose estimation.

Baselines. The closest prior work that can tackle our task outline in Sec. 3 is SAMURAI [14] on which our method is based. We compare against SAMURAI as a baseline and also conduct experiments using NeROIC [32], GNeRF [42], and a modified version of NeRS [71] (details in the supplement). For experiments on joint shape and pose estimation, we use the same quadrant-based pose initialization for NeRS, SAMURAI and SHINOBI (ours); and we use the the methods' default pose initializations for NeROIC (COLMAP) and GNeRF (Random).

Evaluation. We use two strategies for evaluation. First, the standard novel view synthesis metrics using the learned volumes that measure PSNR, SSIM, and LPIPS [75] scores on held-out test images. Second, to evaluate camera poses w.r.t GT poses, we use Procrustes analysis [24] to align the cameras and then compute the mean absolute rotation and translation differences in camera pose estimations for all available views. For evaluation purposes, we optimize the cameras and illuminations on the test images but do not allow the test images to affect the other network parts or hash grid embedding. For a fair comparison, we use the ground truth masks as input to all methods although our method also

Method	Pose Init	PSNR↑		SSIM↑		LPIPS↓		Translation↓		Rotation ° ↓	
		S_C	$\sim S_C$	S_C	$\sim S_C$	S_C	$\sim S_C$	S_C	$\sim S_C$	S_C	$\sim S_C$
GNeRF [42]	Random	8.30	6.25	0.64	0.63	0.52	0.57	1.02 ± 0.16	1.04 ± 0.09	93.15 ± 26.54	80.22 ± 27.64
NeROIC [32]	COLMAP	19.77	-	0.88	-	0.150	-	0.09 ± 0.12	-	42.11 ± 17.19	-
NeRS [71]	Directions	18.67	18.66	**0.92**	**0.93**	0.108	0.107	0.49 ± 0.21	0.52 ± 0.19	122.41 ± 10.61	123.63 ± 8.80
SAMURAI [14]	Directions	**25.34**	24.61	**0.92**	0.91	0.096	0.105	0.24 ± 0.17	0.35 ± 0.24	26.16 ± 22.72	36.59 ± 29.98
SHINOBI	Directions	25.15	**24.77**	**0.92**	0.92	**0.090**	**0.095**	0.250 ± 0.085	0.28 ± 0.09	22.84 ± 16.19	33.00 ± 19.97

Table 2. **Metrics for 3D shape and pose on NAVI.** View synthesis and pose metrics over two subsets from all wild-sets depending on the success of COLMAP (S_C / $\sim S_C$). Rendering quality is evaluated on a holdout set of test views that are aligned as part of the optimization without contributing to the shape recovery. We include GNeRF as a separate baseline although this method is not designed for multi-illumination data. We report metrics with the methods' default camera initialization and evaluate against the annotation provided in NAVI [28].

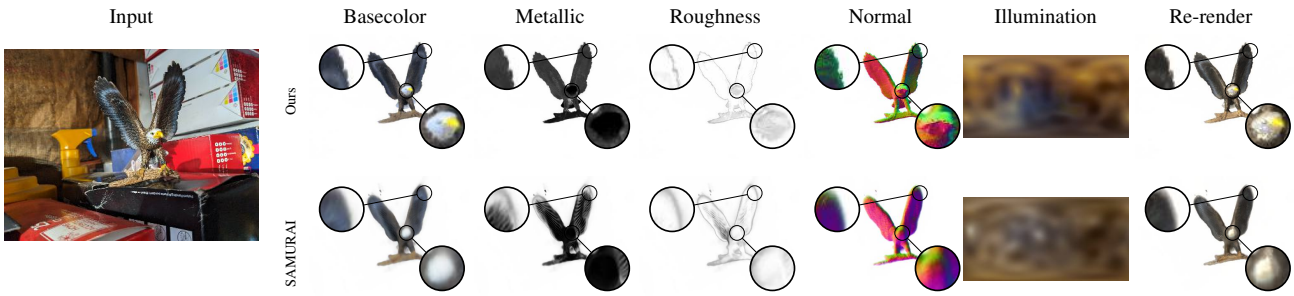


Figure 5. **Comparison with SAMURAI decomposition for joint pose and object reconstruction.** Due to the improved alignment and representation higher frequency details are reconstructed in shape and the BRDF components compared to SAMURAI. Notice the improved texture detail and silhouettes of ours. Both methods optimize camera poses jointly initialized from rough quadrants.

includes functionality to automatically generate segmentation masks. We run experiments on a single Nvidia A100 or V100 GPU per scene.

Results. Tab. 1 shows the performance of different methods for in-the-wild reconstruction when using GT poses from NAVI. Following NAVI [28], we divide the scenes into two subsets based on whether the COLMAP works (S_C) or not ($\sim S_C$) as some techniques like NeROIC need COLMAP poses to work on unposed image collections. Using the provided annotated poses SHINOBI clearly performs best on the view synthesis task (Tab. 1). This shows the advantage of our hybrid encoding scheme and the patch-based losses over previous methods for in-the-wild scenes. Optimization runtimes of different techniques show that we are 3 times faster than the next-best SAMURAI approach.

Tab. 2 shows results of joint shape and pose optimization from in-the-wild image collections when the GT camera poses are not given as input. SHINOBI outperforms both NeROIC and NeRS by a healthy margin while being on-par with SAMURAI. While PSNR of SHINOBI is similar to SAMURAI, our method is able to reconstruct scenes consistently with lower translation and rotation pose errors (with also lower standard deviation in pose metrics). This results in SHINOBI obtaining better LPIPS perceptual metrics compared to SAMURAI. The on-par mean PSNR compared to SAMURAI mostly stems from individual test cameras not being aligned properly. This also happens for other methods but seems to be emphasized by the faster optimization scheduling in SHINOBI. NeROIC can also achieve good results if camera poses are close to the ground truth but fails for many scenes where a COLMAP-based initialization is not possible. NeRS also succeeds in reconstructing all scenes. However, it achieves lower-quality camera alignments. Fig. 6 visually compares view synthesis results from different methods, which visually confirms that SHINOBI can produce sharper results that are more faithful to the input images. Further results on the NAVI dataset are shown in Fig. 7, where we show novel views predicted by SHINOBI initialized with either GT poses or rough quadrants. Visual results clearly show that SHINOBI can recover the pose and provide a consistent illumination w.r.t the ground-truth target views in both settings.

Decomposition results. Fig 5 compares the BRDF and illumination decomposition of SHINOBI to SAMURAI where the same output modalities are available. Visual results show significantly more high-frequency detail and plausible material parameters with SHINOBI compared to SAMURAI.

Ablation study. We ablate different aspects of SHINOBI in terms of reconstruction metrics using the "Keywest" and "School Bus", two in-the-wild sets from NAVI [28] of medium complexity. Metrics in Tab. 3 show that the res-

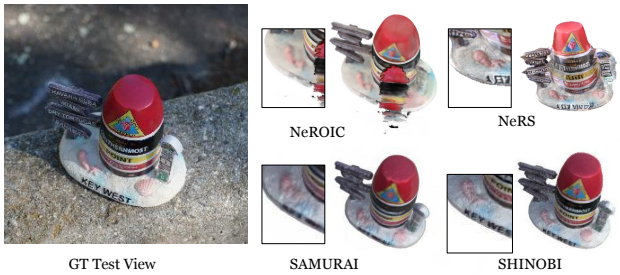


Figure 6. **Novel view synthesis compared to existing methods.** Compared to other methods on an example view from the NAVI [28] in-the-wild test set, SHINOBI preserves fine detail and recreates the lighting realistically.

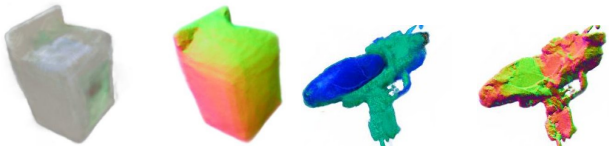
olution annealing coarse-to-fine scheme and the patch-based losses contribute most significantly to the final quality. The latter improves local details and registration accuracy compared to a simple pixel-wise loss. The view importance weighting is another important factor for improved sharpness. It helps to stabilize the optimization after the initial resolution annealing schedule has ended. While the hybrid encoding and camera multiplex consistency do not seem to have a large impact quantitatively, they play a critical role in stabilizing the optimization over different scene types and scales. Without them, the optimization might take longer or diverge depending on the initialization. Visual examples of the specific ablations are compared in the supplementary material.

Applications. In addition to novel view synthesis using the NeRF [44] representation, the parametric material model allows for controlled editing of the object's appearance. Also the illumination can be adjusted, e.g. for realistic composites. A mesh extraction allows further editing and integration in the standard graphics pipeline including real-time rendering. SHINOBI can help in obtaining relightable 3D assets for e-commerce applications as well as 3D AR and VR for entertainment and education. Refer to the supplementary material for sample visual results on relighting, material editing etc.

Limitations. Joint pose and shape reconstruction is an inherently ill-posed problem. While SHINOBI improves over previous work, especially symmetric objects and highly specular materials can lead to failure cases as shown in Fig. 8. The coarse-to-fine scheme is not able to resolve the disambiguities and the camera poses are stuck in a local minimum. All existing methods show these limitations to some extent. In some regions, high-frequency detail is still not reconstructed properly due to misaligned views and the band limited capabilities of the illumination representation [13]. Furthermore, our BRDF and illumination decomposition is not capable of modeling shadowing and inter-reflections. As

(a) GT Novel View (b) GT pose init. (c) Direction pose init.

Figure 7. **View synthesis on NAVI.** Renderings from SHINOBI using models initialized with camera pose quadrants only or the GT provided by NAVI [28] compared to the input image.



(a) Kitchen sink (rgb, normals) (b) Water gun (rgb, normals)

Figure 8. **Failure cases.** Unconstrained image collections featuring highly symmetric objects or homogenous surfaces still pose a challenge and potentially require additional assistance.

we are mainly concerned with single-object decomposition, these are not crucial. Extending this method to more complex light transport modeling forms an important future work.

5. Conclusion

We present SHINOBI, a framework for shape, pose, and illumination estimation of objects from unposed in-the-wild image collections. Using a hybrid hash grid encoding scheme we enable easier camera pose optimization using a multiresolution hash grid. Additionally, our choice of camera parameterization along with per-view importance weighting and patch-based alignment loss allows for a better image-to-3D alignment resulting in better reconstruction with high-frequency details. Although SHINOBI is able to recover the geometry of objects from any category, its performance is limited on thin/transparent structures and fails to recover high-frequency details under extreme illumination changes, which we leave as exploration for future work.

Acknowledgements

This work has been partially funded by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) under Germany's Excellence Strategy – EXC number 2064/1 – Project number 390727645 and SFB 1233, TP 02 - Project number 276693517.

References

[1] Martín Abadi, Ashish Agarwal, Paul Barham, Eugene Brevdo, Zhifeng Chen, Craig Citro, Greg S. Corrado, and Andy Davis et al. TensorFlow: Large-scale machine learning on heterogeneous systems, 2015. Software available from tensorflow.org. 6

[2] Miika Aittala, Timo Aila, and Jaakko Lehtinen. Reflectance modeling by neural texture synthesis. *ACM TOG*, 2018. 3

[3] Rachel Albert, Dorian Yao Chan, Dan B. Goldman, and James F. O'Brian. Approximate svBRDF estimation from mobile phone video. *Eurographics Symposium on Rendering*, 2018. 3

[4] Jonathan T. Barron, Ben Mildenhall, Dor Verbin, Pratul P. Srinivasan, and Peter Hedman. Zip-NeRF: Anti-Aliased Grid-Based Neural Radiance Fields. *ICCV*, 2023. 2, 6

[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*, 2020. 2, 3

[6] Sai Bi, Zexiang Xu, Kalyan Sunkavalli, Miloš Hašan, Yannick Hold-Geoffroy, David Kriegman, and Ravi Ramamoorthi. Deep reflectance volumes: Relightable reconstructions from multi-view photometric images. *ECCV*, 2020. 2

[7] Sai Bi, Zexiang Xu, Kalyan Sunkavalli, David Kriegman, and Ravi Ramamoorthi. Deep 3d capture: Geometry and reflectance from sparse multi-view images. *CVPR*, 2020. 3

[8] Wenjing Bian, Zirui Wang, Kejie Li, Jiawang Bian, and Victor Adrian Prisacariu. Nope-nerf: Optimising neural radiance field with no pose prior. *CVPR*, 2023. 2, 3

[9] Mark Boss and Hendrik P.A. Lensch. Single image brdf parameter estimation with a conditional adversarial network. *arXiv*, 2019. 3

[10] Mark Boss, Fabian Groh, Sebastian Herholz, and Hendrik P. A. Lensch. Deep Dual Loss BRDF Parameter Estimation. *Workshop on Material Appearance Modeling*, 2018. 3

[11] Mark Boss, Varun Jampani, Kihwan Kim, Hendrik P.A. Lensch, and Jan Kautz. Two-shot spatially-varying BRDF and shape estimation. *CVPR*, 2020. 3

[12] Mark Boss, Raphael Braun, Varun Jampani, Jonathan T. Barron, Ce Liu, and Hendrik P.A. Lensch. NeRD: Neural reflectance decomposition from image collections. *ICCV*, 2021. 2, 3, 4

[13] 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. *NeurIPS*, 2021. 2, 3, 4, 8

[14] Mark Boss, Andreas Engelhardt, Abhishek Kar, Yuanzhen Li, Deqing Sun, Jonathan T. Barron, Hendrik P.A. Lensch, and Varun Jampani. SAMURAI: Shape And Material from Unconstrained Real-world Arbitrary Image collections. *NeurIPS*, 2022. 2, 3, 4, 5, 6, 7

[15] Yue Chen, Xingyu Chen, Xuan Wang, Qi Zhang, Yu Guo, Ying Shan, and Fei Wang. Local-to-global registration for bundle-adjusting neural radiance fields. *CVPR*, pages 8264–8273, 2023. 3

[16] Zhiqin Chen and Hao Zhang. Learning implicit fields for generative shape modeling. *CVPR*, 2019. 2

[17] Robert L. Cook and Kenneth E. Torrance. A reflectance model for computer graphics. *ACM TOG*, 1982. 3

[18] Valentin Deschaintre, Miika Aittalla, Fredo Durand, George Drettakis, and Adrien Bousseau. Single-image SVBRDF capture with a rendering-aware deep network. *ACM TOG*, 2018. 3

[19] Valentin Deschaintre, Miika Aittalla, Fredo Durand, George Drettakis, and Adrien Bousseau. Flexible SVBRDF capture with a multi-image deep network. *Eurographics Symposium on Rendering*, 2019. 3

[20] Valentin Deschaintre, George Drettakis, and Adrien Bousseau. Guided fine-tuning for large-scale material transfer. *Eurographics Symposium on Rendering*, 2020. 3

[21] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. *ICLR*, 2021. 3

[22] Duan Gao, Xiao Li, Yue Dong, Pieter Peers, and Xin Tong. Deep inverse rendering for high-resolution SVBRDF estimation from an arbitrary number of images. *ACM Transactions on Graphics (SIGGRAPH)*, 2019. 3

[23] Shubham Goel, Angjoo Kanazawa, and Jitendra Malik. Shape and viewpoint without keypoints. *ECCV*, 2020. 5

[24] John C Gower and Garmt B Dijksterhuis. *Procrustes problems*. OUP Oxford, 2004. 6

[25] Philipp Henzler, Valentin Deschaintre, Niloy J Mitra, and Tobias Ritschel. Generative modelling of BRDF textures from flash images. *ACM Transactions on Graphics (SIGGRAPH ASIA)*, 2021. 3

[26] Hwan Heo, Taekyung Kim, Jiyoung Lee, Jaewon Lee, Soohyun Kim, Hyunwoo J. Kim, and Jin-Hwa Kim. Robust camera pose refinement for multi-resolution hash encoding. *ICML*, 2023. 2, 4

[27] Ajay Jain, Matthew Tancik, and Pieter Abbeel. Putting NeRF on a Diet: Semantically Consistent Few-Shot View Synthesis. *ICCV*, 2021. 2

[28] Varun Jampani, Kevis-Kokitsi Maninis, Andreas Engelhardt, Arjun Karpur, Karen Truong, Kyle Sargent, Stefan Popov, Andre Araujo, Ricardo Martin-Brualla, Kaushal Patel, Daniel Vlasic, Vittorio Ferrari, Ameesh Makadia, Ce Liu, Yuanzhen Li, and Howard Zhou. Navi: Category-agnostic image collections with high-quality 3d shape and pose annotations. *NeurIPS*, 2023. 2, 6, 7, 8

[29] Yoonwoo Jeong, Seokjun Ahn, Christopher Choy, Animashree Anandkumar, Minsu Cho, and Jaesik Park. Self-calibrating neural radiance fields. *ICCV*, 2021. 3

[30] Berk Kaya, Suryansh Kumar, Carlos Oliveira, Vittorio Ferrari, and Luc Van Gool. Uncalibrated neural inverse rendering for photometric stereo of general surfaces. *ICCV*, 2021. 3

[31] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv*, 2014. 6

[32] Zhengfei Kuang, Kyle Olszewski, Menglei Chai, Zeng Huang, Panos Achlioptas, and Sergey Tulyakov. NeROIC: Neural object capture and rendering from online image collections. *arXiv*, 2022. 2, 4, 6, 7

[33] Hendrik P. A. Lensch, Wolfgang Heidrich, and Hans-Peter Seidel. Automated texture registration and stitching for real world models. *Pacific Graphics*, 2000. 2, 6

[34] Zhengqin Li, Kalyan Sunkavalli, and Manmohan Chandraker. Materials for masses: SVBRDF acquisition with a single mobile phone image. *ECCV*, 2018. 3

[35] 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. *CVPR*, 2020. 3

[36] Zhaoshuo Li, Thomas Müller, Alex Evans, Russell H Taylor, Mathias Unberath, Ming-Yu Liu, and Chen-Hsuan Lin. Neuralangelo: High-fidelity neural surface reconstruction. *CVPR*, 2023. 4, 5

[37] Ruofan Liang, Huiting Chen, Chunlin Li, Fan Chen, Selvakumar Panneer, and Nandita Vijaykumar. ENVIDR: Implicit Differentiable Renderer with Neural Environment Lighting. *arXiv*, 2023. 2, 3

[38] Chen-Hsuan Lin, Wei-Chiu Ma, Antonio Torralba, and Simon Lucey. BARF: Bundle-Adjusting Neural Radiance Fields. *ICCV*, 2021. 3, 4, 5

[39] Sainan Liu, Shan Lin, Jingpei Lu, Shreya Saha, Alexey Supikov, and Michael Yip. BAA-NGP: Bundle-Adjusting Accelerated Neural Graphics Primitives. *arXiv*, 2023. 2, 4, 5

[40] Xiaoxiao Long, Cheng Lin, Peng Wang, Taku Komura, and Wenping Wang. Sparseneus: Fast generalizable neural surface reconstruction from sparse views. *ECCV*, 2022. 2

[41] Ricardo Martin-Brualla, Noha Radwan, Mehdi S. M. Sajjadi, Jonathan T. Barron, Alexey Dosovitskiy, and Daniel Duckworth. NeRF in the Wild: Neural Radiance Fields for Unconstrained Photo Collections. *CVPR*, 2021. 2

[42] Quan Meng, Anpei Chen, Haimin Luo, Minye Wu, Hao Su, Lan Xu, Xuming He, and Jingyi Yu. GNeRF: GAN-based Neural Radiance Field without Posed Camera. *ICCV*, 2021. 6, 7

[43] Lars Mescheder, Michael Oechsle, Michael Niemeyer, Sebastian Nowozin, and Andreas Geiger. Occupancy networks: Learning 3d reconstruction in function space. *CVPR*, 2019. 2

[44] Ben Mildenhall, Pratul Srinivasan, Matthew Tancik, Jonathan T. Barron, Ravi Ramamoorthi, and Ren Ng. NeRF: Representing scenes as neural radiance fields for view synthesis. *ECCV*, 2020. 2, 3, 4, 8

[45] Thomas Müller, Alex Evans, Christoph Schied, and Alexander Keller. Instant neural graphics primitives with a multiresolution hash encoding. *ACM TOG*, 2022. 2, 3, 4, 6

[46] Jacob Munkberg, Jon Hasselgren, Tianchang Shen, Jun Gao, Wenzheng Chen, Alex Evans, Thomas Mueller, and Sanja Fidler. Extracting Triangular 3D Models, Materials, and Lighting From Images. *CVPR*, 2022. 2

[47] Giljoo Nam, Diego Gutierrez, and Min H. Kim. Practical SVBRDF acquisition of 3d objects with unstructured flash photography. *ACM Transactions on Graphics (SIGGRAPH ASIA)*, 2018. 2, 3

[48] Michael Niemeyer, Jonathan T. Barron, Ben Mildenhall, Mehdi S. M. Sajjadi, Andreas Geiger, and Noha Radwan. Regnerf: Regularizing neural radiance fields for view synthesis from sparse inputs. *CVPR*, 2022. 2

[49] Michael Oechsle, Songyou Peng, and Andreas Geiger. Unisurf: Unifying neural implicit surfaces and radiance fields for multi-view reconstruction. *ICCV*, 2021. 2

[50] Jeong Joon Park, Peter Florence, Julian Straub, Richard Newcombe, and Steven Lovegrove. Deepsdf: Learning continuous signed distance functions for shape representation. *CVPR*, 2019. 2

[51] Keunhong Park, Philipp Henzler, Ben Mildenhall, Jonathan T. Barron, and Ricardo Martin-Brualla. Camp: Camera preconditioning for neural radiance fields. *ACM Trans. Graph.*, 2023. 2, 4

[52] Daniel Rebain, Mark Matthews, Kwang Moo Yi, Dmitry Lagun, and Andrea Tagliasacchi. LOLNeRF: Learn from One Look. *CVPR*, 2022. 2

[53] Shen Sang and Manmohan Chandraker. Single-shot neural relighting and SVBRDF estimation. *ECCV*, 2020. 3

[54] Johannes Lutz Schönberger and Jan-Michael Frahm. Structure-from-motion revisited. *CVPR*, 2016. 2, 3

[55] Johannes Lutz Schönberger, Enliang Zheng, Marc Pollefeys, and Jan-Michael Frahm. Pixelwise view selection for unstructured multi-view stereo. *ECCV*, 2016. 2, 3

[56] Soumyadip Sengupta, Jinwei Gu, Kihwan Kim, Guilin Liu, David W. Jacobs, and Jan Kautz. Neural inverse rendering of an indoor scene from a single image. *ICCV*, 2019. 3

[57] S. Sinha, J. Y. Zhang, A. Tagliasacchi, I. Gilitschenski, and D. B. Lindell. Sparsepose: Sparse-view camera pose regression and refinement. *CVPR*, 2023. 3

[58] 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. *CVPR*, 2021. 2

[59] 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. *NeurIPS*, 2020. 2, 4

[60] Prune Truong, Marie-Julie Rakotosaona, Fabian Manhardt, and Federico Tombari. Sparf: Neural radiance fields from sparse and noisy poses. *CVPR*, 2023. 2

[61] Itsuki Ueda, Yoshihiro Fukuhara, Hirokatsu Kataoka, Hiroaki Aizawa, Hidehiko Shishido, and Itaru Kitahara. Neural density-distance fields. *ECCV*, 2022. 2

[62] Peng Wang, Lingjie Liu, Yuan Liu, Christian Theobalt, Taku Komura, and Wenping Wang. NeuS: Learning neural implicit surfaces by volume rendering for multi-view reconstruction. *NeurIPS*, 2021.

[63] Yiming Wang, Qin Han, Marc Habermann, Kostas Daniilidis, Christian Theobalt, and Lingjie Liu. Neus2: Fast learning of neural implicit surfaces for multi-view reconstruction. *ICCV*, 2023. 2

[64] Zirui Wang, Shangzhe Wu, Weidi Xie, Min Chen, and Victor Adrian Prisacariu. NeRF−−: Neural radiance fields without known camera parameters. *arXiv*, 2021. 3, 4

[65] Jiamin Xu, Zihan Zhu, Hujun Bao, and Weiwei Xu. A Hybrid Mesh-neural Representation for 3D Transparent Object Reconstruction. *cvmj*, 2022. 2

[66] Jiawei Yang, Marco Pavone, and Yue Wang. FreeNeRF: Improving Few-shot Neural Rendering with Free Frequency Regularization. *CVPR*, 2023. 2

[67] 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. *ECCV*, 2022. 2

[68] Lior Yariv, Jiatao Gu, Yoni Kasten, and Yaron Lipman. Volume rendering of neural implicit surfaces. *NeurIPS*, 2021. 2

[69] Weicai Ye, Shuo Chen, Chong Bao, Hujun Bao, Marc Pollefeys, Zhaopeng Cui, and Guofeng Zhang. IntrinsicNeRF: Learning Intrinsic Neural Radiance Fields for Editable Novel View Synthesis. *ICCV*, 2023. 2, 3

[70] Jianzhao Zhang, Guojun Chen, Yue Dong, Jian Shi, Bob Zhang, and Enhua Wu. Deep inverse rendering for practical object appearance scan with uncalibrated illumination. *ACG*, 2020. 3

[71] Jason Zhang, Gengshan Yang, Shubham Tulsiani, and Deva Ramanan. NeRS: Neural reflectance surfaces for sparse-view 3d reconstruction in the wild. *NeurIPS*, 2021. 2, 3, 4, 6, 7

[72] Jiahui Zhang, Fangneng Zhan, Rongliang Wu, Yingchen Yu, Wenqing Zhang, Bai Song, Xiaoqin Zhang, and Shijian Lu. VMRF: View Matching Neural Radiance Fields. *ACM MM*, 2022. 3

[73] 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. *ICCV*, 2023. 3

[74] Kai Zhang, Fujun Luan, Qianqian Wang, Kavita Bala, and Noah Snavely. PhySG: Inverse rendering with spherical Gaussians for physics-based material editing and relighting. *CVPR*, 2021. 2, 3

[75] Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. *CVPR*, 2018. 6

[76] Xiuming Zhang, Pratul P. Srinivasan, Boyang Deng, Paul Debevec, William T. Freeman, and Jonathan T. Barron. Nerfactor: Neural factorization of shape and reflectance under an unknown illumination. *ACM Trans. Graph.*, 40(6), 2021. 3

[77] Yuanqing Zhang, Jiaming Sun, Xingyi He, Huan Fu, Rongfei Jia, and Xiaowei Zhou. Modeling indirect illumination for inverse rendering. *CVPR*, 2022. 2, 3

[78] Yi Zhou, Connelly Barnes, Jingwan Lu, Jimei Yang, and Hao Li. On the continuity of rotation representations in neural networks. *CVPR*, 2019. 4

[79] Hao Zhu, Fengyi Liu, Qi Zhang, Xun Cao, and Zhan Ma. Rhino: Regularizing the hash-based implicit neural representation. *arXiv*, 2023. 2, 4