Towards Scalable Multi-View Reconstruction of Geometry and Materials

Carolin Schmitt‡ Božidar Antić Andrei Neculai Joo Ho Lee‡ Andreas Geiger

Abstract—In this paper, we propose a novel method for joint recovery of camera pose, object geometry and spatially-varying Bidirectional Reflectance Distribution Function (svBRDF) of 3D scenes that exceed object-scale and hence cannot be captured with stationary light stages. The input are high-resolution RGB-D images captured by a mobile, hand-held capture system with point lights for active illumination. Compared to previous works that jointly estimate geometry and materials from a hand-held scanner, we formulate this problem using a single objective function that can be minimized using off-the-shelf gradient-based solvers. To facilitate scalability to large numbers of observation views and optimization variables, we introduce a distributed optimization algorithm that reconstructs 2.5D keyframe-based representations of the scene. A novel multi-view consistency regularizer effectively synchronizes neighboring keyframes such that the local optimization results allow for seamless integration into a globally consistent 3D model. We provide a study on the importance of each component in our formulation and show that our method compares favorably to baselines. We further demonstrate that our method accurately reconstructs various objects and materials and allows for expansion to spatially larger scenes. We believe that this work represents a significant step towards making geometry and material estimation from hand-held scanners scalable.

1 Introduction

As AR/VR technologies are emerging, there is an increasing demand for scanning casual scenes for immersive and interactive virtual experiences. Both, for a high level of realism and to increase robustness to varying illumination, it is beneficial to reconstruct material and reflectance properties alongside geometry. Yet, 3D shape and appearance estimation is very challenging due to sparse measurements, large computation and memory requirements and the interlaced and complex correlation of geometric and photometric entities. Therefore, most works target only small objects which are often captured using complex light stages in a laboratory. However, in order to generate photo-realistic replica of real environments for training embodied agents or telepresence, this is insufficient. Instead, we require accurate geometry and material reconstructions for large, complex scenes and the ability to work with data from mobile scanners that capture scenes from arbitrary viewpoints.

Ideally, object geometry and material properties are inferred jointly: a good model of light transport allows for recovering geometric detail using shading cues. An accurate shape model, in turn, facilitates the estimation of material properties. This is particularly relevant for shiny surfaces and detailed geometries. Yet joint optimization of geometry and material from a handheld device poses an inverse rendering problem and is ill-posed and under-determined. Existing approaches assume fixed camera poses [1], [2] or leverage sophisticated pipelines [3], [4], [5], [6] which decompose the problem into smaller problems using multiple decoupled objectives and optimization algorithms that treat geometry and materials separately. In this work, we provide a novel formulation for this problem which does not rely on sophisticated pipelines or decoupled objective functions.

In order to process larger multi-object scenes, a scalable scene representation is mandatory. Unfortunately, reconstructing large scenes captured from many viewpoints at high resolution (e.g., 4K) quickly becomes intractable. We therefore propose to use local 2.5D scene representations and an optimization scheme that encourages global consistency between them. By optimizing in 2.5D, the proposed model has a constant memory footprint independent of the scene size and allows for reconstructing geometry and materials at larger scales, see Fig. 1.
We summarize the contributions of this paper as follows:

- We demonstrate that joint optimization of camera pose, object geometry and materials is possible using a single objective function and off-the-shelf gradient-based solvers.
- We propose a distributed optimization scheme over a set of 2.5D scene representations that enables accurate integration of 2.5D reconstructions to full 3D models. We show that despite overlapping fields of view, regularizing multi-view consistency is crucial to attain globally accurate reconstructions without visual artifacts.
- We provide a study on the importance of each component in our formulation and a comparison to multiple baselines.
- We demonstrate that our model can be used to reconstruct scenes exceeding object-level that include multiple objects with various different materials.
- We provide videos of our reconstructed models and make our source code and dataset publicly available at https://sites.google.com/view/material-fusion/.

This journal paper is an extension of a conference paper published at CVPR 2020 [7] which jointly estimates pose, geometry and svBRDF from handheld data in 2.5D. In Section 7.3 we demonstrate that simple fusion of the 2.5D parameter maps

- *For this project, all authors were with the Autonomous Vision Group, University of Tübingen and Max Planck Institute for Intelligent Systems, Tübingen, Germany. E-mail: {firstname.lastname}@tue.mpg.de*
- *Joo Ho Lee is now with Sogang University, Seoul, South Korea.*
- ‡ *denotes corresponding authors.*
  Emails: carolin.schmitt@tue.mpg.de, jhleecs@sogang.ac.kr

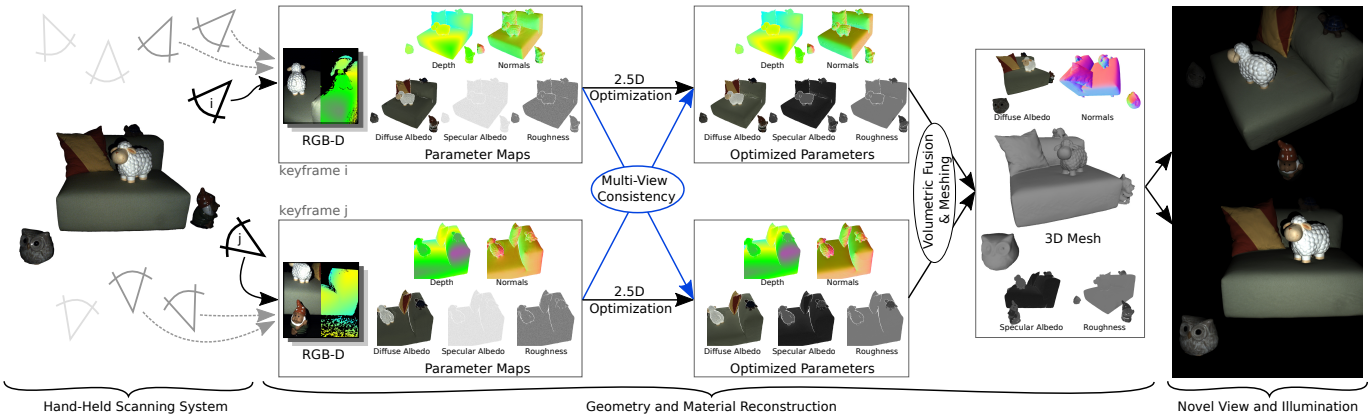


Fig. 1: **Globally Consistent Material and Geometry Reconstruction.** Given RGB-D images from a mobile hand-held scanner (left), the proposed method uses local 2.5D representations to iteratively reconstruct globally consistent poses, geometry and material parameter maps that can be integreated into a 3D representation which features per voxel normals and material parameters (middle). This allows for rendering novel views under unseen illumination (right). Our approach can handle both multiple-object scenes and very specular materials.

produced by [7] is insufficient to obtain an accurate and consistent 3D model. We therefore extend our previous model with a distributed multi-view optimization stage which enables fusion of 2.5D representations of geometry and materials into consistent 3D models.

Further, in comparison to [7], we 1) model the specular BRDF parameters per pixel for a richer and more flexible material representation, 2) refine the regularization terms to account for the new material model and better ensure proximity to the measurements, 3) provide a thorough ablation study of our global multi-view consistent optimization scheme, 4) add qualitative comparisons to [7] and [5], and 5) show reconstruction results for additional objects and scenes that demonstrate the scalability of our method.

2 Related Work

We now discuss the most related work on geometry, material as well as joint geometry and material estimation. We further provide an overview on geometry estimation at scale.

2.1 Geometry Estimation

Multi-View Stereo (MVS) reconstruction techniques [8], [9], [10], [11], [12], [13], [14] recover the 3D geometry of an object from multiple input images by matching feature correspondences across views or by optimizing photo-consistency. As they ignore physical light transport, they cannot recover material properties. Furthermore, they are only able to recover geometry for surfaces which are sufficiently textured.

Shape from Shading (SfS) techniques exploit shading cues for reconstructing [15], [16], [17], [18], [19] or for refining [20], [21], [22], [23] 3D geometry from one or multiple images by relating surface normals to image intensities through Lambert's law. While early SfS approaches were restricted to objects made of a single Lambertian material, modern reincarnations of these models [24], [25], [26] are also able to infer non-Lambertian materials and lighting. Unfortunately, reconstructing geometry from a single image is a highly ill-posed problem, requiring strong assumptions about the surface geometry. Moreover, textured objects often cause ambiguities as intensity changes can be caused by changes in either surface orientation or surface albedo.

Photometric Stereo (PS) approaches [27], [28], [29], [30], [31], [32], [33] assume three or more images captured with a static camera while varying illumination or object pose [34], [35] to resolve the aforementioned ambiguities. In contrast to early PS approaches which often assumed orthographic cameras and distant light sources, newer works have considered the more practical setup of near light sources [36], [37], [38], [39] and perspective projection [40], [41], [42]. To handle non-Lambertian surfaces, robust error functions have been suggested [43], [44] and the problem has been formulated using specularity-invariant image ratios [45], [46], [47], [48]. The advantages of PS (accurate normals) and MVS (global geometry) have also been combined by integrating normals from PS and geometry from MVS [49], [50], [51], [52], [53], [54], [55], [56] into a single consistent reconstruction. However, many classical PS approaches are not capable of estimating material properties other than albedo and most PS approaches require a fixed camera which restricts their applicability to lab environments. In contrast, here we are interested in recovering shape and surface materials of larger scenes using a *handheld* mobile scanner.

2.2 Material Estimation

Intrinsic Image Decomposition [24], [57], [58], [59] is the problem of decomposing an image into its material-dependent and light-dependent properties. However, only a small portion of the 3D physical process is captured by these models and strong regularizers must be exploited to solve the task. A more accurate description of the reflective properties of materials is provided by the Bidirectional Reflectance Distribution Function (BRDF) [60].

For **known 3D geometry**, the BRDF can be measured using specialized light stages or gantries [61], [62], [63], [64], [65]. While this setup leads to accurate reflectance estimates, it is typically expensive, stationary and only works for objects of limited size. In contrast, recent works have demonstrated that reflectance properties of flat surfaces can be acquired using an ordinary mobile phone [66], [67], [68], [69]. While data collection is easy and practical, these techniques are designed for capturing flat textured surfaces and do not generalize to objects with more complex geometries.

More closely aligned with our goals are approaches that estimate parametric BRDF models for scenes with known geometry based on sparse measurements of the BRDF space [70], [71], [72], [73], [74], [75], [76], [77], [78], [79]. While we also estimate a parametric BRDF model and assume only sparse measurements of the BRDF domain, we *jointly* optimize for camera pose, object geometry and material parameters. As our experiments show, joint optimization allows for recovering fine geometric structures not present in the initial reconstruction while at the same time improving material estimates compared to a sequential treatment of both tasks.

2.3 Joint Geometry and Material Estimation

Several works have addressed the problem of jointly inferring geometry and materials. By integrating shading cues with multi-view constraints and an accurate model of materials and light transport, this approach has the potential to deliver the most accurate results. However, joint optimization of all relevant quantities is a challenging task. Several works have considered extensions of the classic PS setting [1], [80], [81], [82], [83], [84], [85], [86], [87], [88], [89], [90]. While some of these approaches consider multiple viewpoints and/or estimate spatially varying BRDFs, all of them require multiple images from the **same or known viewpoints** as input. In contrast, we are interested in jointly estimating geometry and materials from **mobile scanning systems**, enabling applications outside laboratory environments.

In 2011, [91] proposed to exploit low-cost and handheld scanning devices such as a flash camera for reconstructing both BRDFs and geometry from multi-view images. Like subsequent works [92], [93], [94], [95] they are restricted to flat surfaces, simple shapes or uniform materials. For **single objects or small scenes** the following works estimate materials alongside geometry: Higo et al. [3] estimate a depth, normal and diffuse albedo map of a Lambertian object by graph-cut-based plane sweeping. Georgoulis et al. [4] optimize 3D geometry and a data-driven BRDF model in an alternating fashion. Nam et al. [5] refine a subdivided mesh by alternatively updating positions, normals, and material properties. Finally, Li et al. [6] iteratively optimize for 3D geometry, reflectance, camera pose and environment lighting. All these methods decompose the problem into smaller problems by splitting the optimization variables by their property (i.e. geometry, materials, poses) and alternate the optimization over those properties using multiple decoupled objectives. In contrast, we exploit that spatially separated regions naturally decouple the corresponding optimization variables and therefore, we decompose the problem based on spatial regions instead of separate properties. This has two advantages: 1) It enables us to optimize all parameters of each region jointly and to use a single objective function. Consequently, we can use all information encapsulated in the intricate interplay of geometry and materials and reach high accurate reconstructions. 2) The separation into spatial regions allows us to distribute the optimization over the local 2.5D representations of these regions and thus, facilitates scalability to larger scenes.

Following our conference paper [7], Luan et al. [96] proposed another method for jointly optimizing geometry and spatially-varying reflectance. They represent the geometry of a single object as a mesh and alternate the optimization over mesh vertices and reflectance with re-meshing in a coarse-to-fine process. Hereby, they use a co-located configuration of a hand-held camera and point light which greatly simplifies the rendering process but also restricts the sample space of the BRDF. In contrast, we use multiple, alternating light sources in conjunction with explicit shadow modeling to extract most information from the sparse set of samples that is captured with a handheld setup.

Recently, **neural scene representations** [97], [98], [99] , [100] have been trained to reconstruct surface normals and reflectance properties of complex and multi-colored objects. Most works are targeting object-centric scenarios with a fixed scanning volume. [97] uses voxel representation of deep features that encodes opacity, normals, and materials. Instead of storing deep features discretely, [98] trains a neural network with positional encoding to represent continuous 3D functions of scene properties. To track the 2D topology of the 3D surfaces of the object, [99] optimize the neural transform from the unit sphere to a 3D object. Allowing for arbitrary topologies, [100] train MLPs to predict SDF values and material parameters using a two-step hybrid optimization scheme. For a scene with multiple objects, all these methods either require to compromise reconstruction resolution due to a fixed encoding which limits scalability. Or they are designed to re-train the network individually for each object in an object-centered unit-volume. This maximizes reconstruction quality per object but at the cost of a global scene representation. In contrast, for our method the resolution does not depend on the scene size due to the local 2.5D representations and we optimize all local representations in a single global world coordinate system which does not require any reconfiguration of the representation after initialization.

With our proposed method, we make a step towards reconstructions of geometry, materials and poses beyond object-level. For scalability, we optimize a set of 2.5D representations instead of a single 3D representation since each local 2.5D representation has constant requirements in terms of memory and optimization variables, independent of the scene size. Furthermore, in contrast to methods that assume watertight 3D shapes, our model is able to reconstruct partially scanned 3D environments as it doesn't require closed shapes.

2.4 Differentiable Rendering

Differentiable rendering describes a rendering pipeline that allows for computing image pixel changes wrt. scene parameters. It is at the heart of most methods that aim to synthesize photo-realistic images from real-world observations. The approaches are manifold but many share a similar structure: Based on real 2D or 2.5D observations, inverse rendering is used to infer a parametric representation of the 3D scene (e.g. for geometry, illumination or BRDF). This scene representation is then rendered into images by a forward rendering engine. In the following, we discuss four method classes implementing this.

Classical inverse rendering approaches use non-learning-based methods to optimize 2D or 3D scene parameters from observation images via gradient descent. [101], [102] use 'soft' rasterization to differentiate a rasterizer, while [96], [103], [104], [105], [106], [107], [108] propose solutions to differentiate through ray casting. All these works use hand-designed rendering functions. This limits the flexibility of the renderer (as compared to learning-based approaches, discussed afterwards) but has the advantage of physically-based rendering functions which are interpretable and enable rendering a scene under changed conditions (e.g. novel viewpoint, different illumination, or edited materials). In the proposed pipeline, we use such a classical optimization approach

with a hand-designed rendering engine since we aim for physically correct reconstructions.

Neural inverse rendering pipelines [94], [109], [110], [111], [112], [113], [114], [115], [116] train neural networks to predict scene parameters from observations and then use an analytical differentiable rendering layer to synthesize images. The network has the potential to learn to ignore transient objects, adapt to varying illumination conditions in the observations or disentangle the parameters of complex scenes from data. But it also introduces additional parameters and thus, requires data to train. Especially when considering more complex reflectance settings, this is not easy to obtain. In contrast, our approach does not require any large dataset but solely takes the captures of a scene as input.

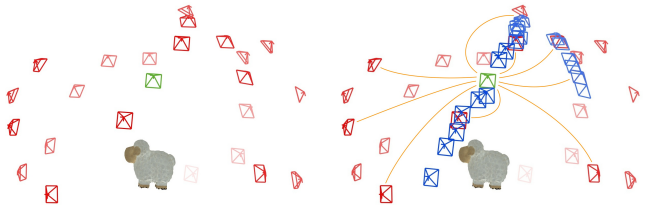
Neural rendering (or 2D neural rendering) refers to methods that use classical surface or volume representations and replace the differentiable rendering engine by a generative model to learn the image formation function. Exemplary tasks are changing the camera viewpoint [117], [118], [119], [120], [121], [122], [123] or relighting [124], [125], [126]. The generative network has the potential to synthesize high-quality novel images, learn visibility constraints, deal with incomplete or inconsistent input representations or depict complex illumination effects like inter-reflections or multiple bounces. However, the rendering network is non-deterministic and how to enforce physical plausibility of the reconstructions is unclear – which is the goal of this paper. Therefore, the proposed approach relies on a classical rendering engine instead of a neural renderer.

Last, **neural scene representations** (or 3D neural rendering, learnable 3D representations) encode the scene parameters in a neural network and combine them with classical differentiable rendering engines. A very well-known example is NeRF [127] and its follow-ups. E.g. [128], [129], [130] enable relighting and [98], [131], [132], [133], [134], [135], [136], [137], [138], [139] include full BRDFs to encode material reflectance. All these approaches are trained or fine-tuned per scene and require either prior knowledge on materials, like pre-trained reflectance or transmittance priors, or strong regularizers like compression to low dimensional latent spaces. While our method is also optimized per scene, we do not assume any prior knowledge on materials. Instead, we predict material parameters per pixel and use regularizers to propagate reflectance information across pixels. Additionally, our 2.5D representation is faster to optimize than an MLP, naturally scales to large scenes and distributes modeling capacity equally across all selected keyframes.

2.5 Geometry Estimation at Scale

To date, there exists no solution to reconstruct accurate geometry and materials at scale. In this paragraph, we therefore review existing work on scalable geometry-only reconstruction. Crucial for a scalable model is a memory efficient scene representation that allows for accurate and dense reconstructions.

One approach is to keep the **full reconstruction in memory** by supporting efficient compression of connected surface data. [140] represent the scanned environment by a light-weight mesh using plane primitives and [141] fits a multi-layer heightmap to a volume of occupancy votes. Those representations support scene completion and can scale efficiently to larger scenes, but fail to reconstruct complex 3D structures. In the context of Image Based Rendering (IBR), [142] calculate a global mesh from a pointcloud reconstruction but then refine per-view depth maps, sacrificing global consistency for local accuracy.



(a) Keyframes (b) Neighbor Views

Fig. 2: **Keyframe and Neighbor View Selection:** (a) Visualized are all keyframes K (red, green) for the object 'Sheep' - they represent the scene as a set of 2.5D maps for efficient optimization. (b) Our method optimizes the parameter set of one keyframe $k \in K$ (green) guided by photometric and geometric constraints from neighboring observation views N_k (blue cameras) and consistency constraints from neighboring keyframes $\bar{N}_k$ (orange lines).

Another approach is to use **scene representations that allow for subdivision** into parts/segments or to introduce hierarchies to facilitate memory efficient processing by keeping only relevant scene parts in memory. Hereby, a common strategy is to first reconstruct the geometry of individual and overlapping scene segments, parts or frames and then integrate those segments in 3D world space while employing sophisticated pose registration, alignment and outlier filtering techniques. Multiple **implicit volumetric models** extent the seminal works of [143] or [144] (which rely on memory inefficient regular voxel grids) to larger environments by introducing efficient volumetric data structures like volume windows [145], patch volumes [146], a hierarchical volume structure [147] or spatial hashing [148], [149]. This increases spatial efficiency. Non-volumetric approaches represent scene segments by, e.g., per frame **2.5D depth maps** [150], [151] or **3D mesh fragments** [152]. All aforementioned subdivision methods enable final reconstructions which would exceed memory limitations during processing. And most employ global pose or texture refinements similar to [140], [153], [154]. The common challenge of non-global geometry reconstruction methods is to assure consistency between local reconstructions.

Our method represents the scene as a collection of 2.5D parameter maps from multiple keyframe views. This representation is memory efficient and we actively encourage consistency between overlapping regions. As demonstrated in Section 7, this leads to accurate and well-aligned reconstructions, eliminating the need for post-processing or refinement.

In the following 4 sections we describe our method in detail. First, we introduce our scene representation and parameterizations of the optimization variables in Section 3. We then present the model formulation and optimization objective in Section 4 before discussing the multi-view consistent optimization scheme in Section 5. The mesh generation step that integrates the 2.5D optimization results into a full 3D model is explained in Section 6.

3 Scene Representation

Our goal is to reconstruct geometry, material properties and camera poses from RGB-D data. Unfortunately, representing an entire scene in memory is computationally demanding, in particular when using memory-limited but computationally efficient GPUs for optimization. Towards scalable scene reconstruction, we therefore exploit a keyframe-based 2.5D representation which

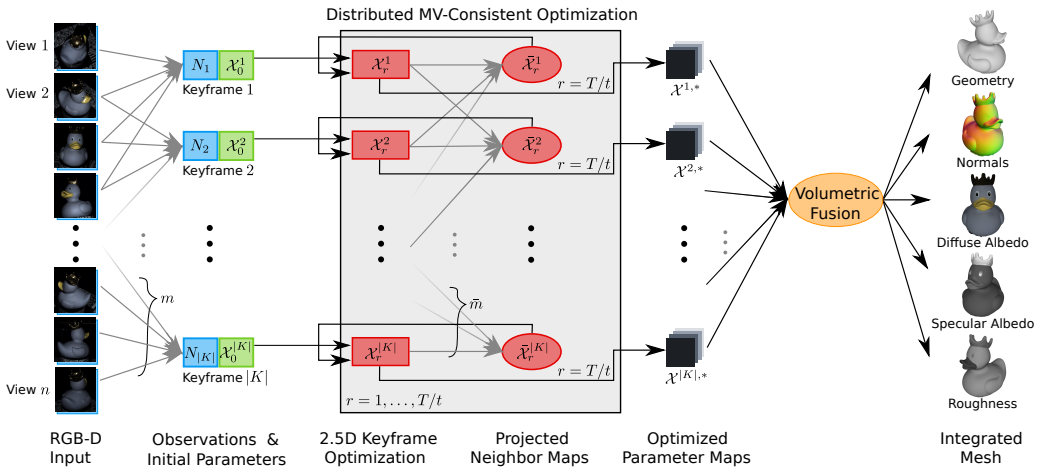


Fig. 3: **Pipeline Overview.** The input to our model are n RGB-D images from which we select a subset of well distributed keyframes K. For each keyframe $k \in K$, we select m neighboring observation views N_k and initialize the set of optimization parameters $\mathcal{X}_0^k$. During optimization, we then iterate the following rounds r (gray box): After optimizing each keyframe representation independently for t iterations, we project the current parameter maps of all neighboring keyframes $\bar{N}_k$ into each keyframe k and use the resulting set $\bar{\mathcal{X}}_r^k$ as additional constraint to the optimization of the next round $r+1$. The sets of $\bar{m}$ neighboring keyframes $\{\bar{N}_k\}_k$ are defined at the start for all keyframes k. After $r = T/t$ rounds, the resulting sets of optimized 2.5D parameter maps $\{\mathcal{X}^{k,*}\}_k$ are integrated into a full 3D model, represented by a mesh with per vertex normal, diffuse and specular albedo as well as roughness parameters.

locally describes and optimizes geometry, materials and poses. In particular, we adopt alternating block coordinate optimization of keyframes to minimize photometric errors while encouraging consistency between adjacent keyframes using soft constraints. An overview of our method is shown in Fig. 3.

The input to our model is an RGB-D sequence captured with a handheld scanner, as shown in Fig. 6, that consists of a color image $\mathcal{I}_i : \mathbb{R}^2 \to \mathbb{R}^3$ and a depth map $\mathcal{Z}_i : \mathbb{R}^2 \to \mathbb{R}$ at each frame $i \in N = \{1, \dots, n\}$. We assume that each image is illuminated by exactly one point light source and that global and ambient illumination effects are negligible. Moreover, we assume the images to be undistorted, de-vignetted and the black frame to be subtracted.

We represent the scene as a set of 2D parameter maps defined at several keyframes of the RGB-D sequence. More specifically, at each keyframe we store the geometry in terms of a depth and normal map, and the materials as BRDF parameter maps. Additionally, each keyframe is linked to a set of camera poses of its respective neighbor views. In the following, we first describe the process of keyframe and neighbor view selection, followed by the representations for poses, geometry and materials.

3.1 Keyframe and Neighbor Selection

To represent the scene, we define a set of keyframes $K \subseteq N$ that capture the scene tightly. For each keyframe $k \in K$ we define two sets of neighboring views: The first set is the set of neighboring observation views N_k which provide photometric and geometric constraints for the local 2.5D multi-view optimizations over the parameter set of keyframe k. Second, we define a set of neighboring keyframes $\bar{N}_k$ from which we project the parameter maps into keyframe k as a soft constraint during optimization to enforce consistency of the local 2.5D reconstructions. And since all keyframes are connected via the overall optimization graph, these pairwise consistency constraints propagate globally during optimization. As evidenced by our experiments, this term is crucial for obtaining a consistent result when fusing all 2.5D representations into a global 3D representation of the scene. All sets of views are visualized in Fig. 2 for the capture of the object 'Sheep'.

Keyframe Selection: To select a set of diverse keyframes $K \subseteq N$, we iteratively compute the pairwise 3D Euclidean distances between the camera centers of all views and remove the view with the minimum distance to its nearest neighbor until the desired number of keyframes has been reached. Generally, the number of keyframes is a tradeoff between accuracy and time and it grows with the scale of the scene. But increasing the number of keyframes is unproblematic for our method since most computations run per keyframe in parallel, with fixed memory requirements independent of the scene size. We ablate the number of keyframes in Section 7.2.

Neighboring Observation Views N_k: To optimize geometry, pose and spatially-varying material parameters, the set of keyframe observations contains too little samples. Therefore, we define m neighboring observation views $N_k \subset N$ with $m = |N_k|$ per keyframe $k \in K$ and minimize the photoconsistency error between these and the predictions of our model. To select the neighbor observation views we only consider views that are within a $40°$ cone around keyframe k with respect to (wrt.) the object center. For larger scenes, we additionally remove views with a view direction that deviates more than $45°$ from the keyframes' view direction. We then choose m views that cover the cone around keyframe k as uniformly as possible by removing the views which are closest to their neighbors.

Neighboring Keyframes $\bar{N}_k$: For each keyframe $k \in K$, we define a set of neighbor keyframes $\bar{N}_k \subseteq K \setminus \{k\}$ of size $\bar{m} = |\bar{N}_k|$. During optimization, we regularize the parameter maps of keyframe k against those of all neighbor keyframes $i \in \bar{N}_k$ projected into k. This enforces consistent parameter estimates across keyframes. To ensure that all neighbors $i \in \bar{N}_k$

share scene content with keyframe k, we sample them randomly from all keyframes that fulfill two conditions: For keyframes i and k, 1) define the *middle point* as the median of all initial geometry points for objects and the first intersection point of the principal ray of camera k with the initial geometry for scenes. Then the two lines connecting each views' camera position with the *middle point* should form an angle of $\leq 60°$. And 2) for scenes, both cameras' view directions form an angle of $\leq 45°$. Note that per keyframe, we sample up to $\bar{m}$ neighboring keyframes, depending on the availability of valid neighbors.

3.2 Keyframe Parameterizations

In this section, we formally describe the keyframe-based parameterization of our model in terms of poses, geometry and materials. For each keyframe $k \in K$, we define its pixels P_k as the set of all pixels of view k with a non-zero initial depth value. As we bound the depth to be non-negative, this implies $z_p^k > 0$.

3.2.1 Camera Parameterization

We use a perspective pinhole camera model and assume constant intrinsic camera parameters that have been calibrated in advance using established calibration procedures [155]. We denote the projective mapping for observation $i \in N_k$ and keyframe $k \in K$ as: $\pi_i^k : \mathbb{R}^3 \rightarrow \mathbb{R}^2$ and represent the extrinsic component (camera pose) of this mapping in world coordinates by a unit quaternion $\mathbf{q}_i^k \in SO(3)$ and a translation vector $\mathbf{t}_i^k \in \mathbb{R}^3$. Note that we use a redundant representation (i.e., the camera pose of an observation neighboring multiple keyframes is represented once per keyframe) to enable memory efficient optimization, one keyframe at a time, while enforcing consistency via additional soft constraints.

3.2.2 Geometry Parameterization

We parameterize geometry in terms of both depth and normal maps and enforce consistency between them using soft constraints.

Depth Map: For each pixel $p \in P_k$ of keyframe k at 2D location $\left(u_p^k, v_p^k\right)^T$ and associated depth z_p^k, the 3D point location $\mathbf{x}_p^k$ is given by

$$\mathbf{x}_p^k = \left(\pi_k^k\right)^{-1}\left(u_p^k, v_p^k, z_p^k\right) \tag{1}$$

where $\left(\pi_k^k\right)^{-1}$ denotes the inverse projection which takes a pixel coordinate and depth value and returns the 3D point in world coordinates.

Normal Map: We represent normals as 3D vectors $\{\mathbf{n}_p^k\}_{p \in P_k}$. During optimization, we only estimate an angular change wrt. the normal of the previous iteration to avoid both the unit vector constraint and the gimbal lock problem.

3.2.3 Material Parameterization

To model reflectance properties, we use a parametric version of the spatially varying Bidirectional Reflectance Distribution Function $f_p(\mathbf{n}_p, \boldsymbol{\omega}_{\text{in}}, \boldsymbol{\omega}_{\text{out}})$ and estimate its parameters per pixel/point $p \in P_k$ and keyframe k.

svBRDF: The svBRDF $f_p(\cdot)$ models the fraction of light that is reflected from incoming light direction $\boldsymbol{\omega}_{\text{in}}$ to outgoing light direction $\boldsymbol{\omega}_{\text{out}}$ given the surface normal $\mathbf{n}_p^k$ at each point $p \in P_k$. We use a modified version of the Cook-Torrance model [156]

$$f_p^k(\mathbf{n}_p^k, \boldsymbol{\omega}_{\text{in}}, \boldsymbol{\omega}_{\text{out}}) = \mathbf{d}_p^k + s_p^k \frac{D(r_p^k)\, G(\mathbf{n}_p^k, \boldsymbol{\omega}_{\text{in}}, \boldsymbol{\omega}_{\text{out}}, r_p^k)}{4(\mathbf{n}_p^k \cdot \boldsymbol{\omega}_{\text{in}})(\mathbf{n}_p^k \cdot \boldsymbol{\omega}_{\text{out}})} \tag{2}$$

with Disney's GTR model [157] for the microfacet slope distribution $D(\cdot)$ and Mitsuba's Smith's function [158] for the geometric attenuation factor $G(\cdot)$. The parameters of the svBRDF are given by the diffuse albedo $\mathbf{d}_p^k \in \mathbb{R}^3$, specular albedo $s_p^k \in \mathbb{R}$ and surface roughness $r_p^k \in \mathbb{R}$ for pixel/point $p \in P_k$ and keyframe k. As in prior work [5], we ignore the Fresnel effect which cannot be observed using an active handheld illumination setup.

4 Optimization Objective

To jointly optimize geometry, materials and pose parameters for each keyframe $k \in K$, we minimize the photometric error between rendered predictions and neighbor view observations while employing multiple additional loss functions for regularization.

For a single keyframe k and its pixels/points $p \in P_k$ and neighbor observation views $i \in N_k$, we wish to estimate the depth z_p^k, geometric surface normals $\mathbf{n}_p^k$, svBRDF parameters $\mathbf{d}_p^k, r_p^k, s_p^k$ as well as the camera poses π_i^k. Denoting the parameter set as

$$\mathcal{X} = \{\{z_p^k, \mathbf{n}_p^k, \mathbf{d}_p^k, r_p^k, s_p^k\}_{p \in P_k}, \{\pi_i^k\}_{i \in N_k}\}_{k \in K}$$

we define our objective function as follows

$$\mathcal{X}^* = \underset{\mathcal{X}}{\operatorname{argmin}}\ \psi_{\mathcal{P}} + \psi_{\mathcal{D}} + \psi_{\mathcal{C}} + \psi_{\mathcal{G}} + \psi_{\mathcal{M}} \tag{3}$$

The individual terms encourage photo-consistency $\psi_{\mathcal{P}}$, depth-consistency $\psi_{\mathcal{D}}$ and multi-view consistency $\psi_{\mathcal{C}}$, impose regularization on the geometry $\psi_{\mathcal{G}}$ and enforce material smoothness $\psi_{\mathcal{M}}$. Note that we omit the dependency on $\mathcal{X}$ and the relative weights between the individual terms for clarity. The full formulation can be found in the supplement.

4.1 Photo and Depth Consistency

We introduce the photo and depth consistency terms in the following. For better readability, we denote $\mathcal{I}_i(\pi_i^k(\mathbf{x}_p^k))$ as the observation $\mathcal{I}_i$ at the 2D location where 3D point $\mathbf{x}_p^k$ is observed in image i. For fractional image coordinates, we use bilinear interpolation. Similarly, we write $\mathcal{Z}_i(\pi_i^k(\mathbf{x}_p^k))$ for depth measurements.

Photo Consistency: The photo-consistency term ensures that the prediction of our model matches the observation $\mathcal{I}_i$ for every neighbor view $i \in N_k$ and all visible and illuminated ($\varphi_p^{ki} = 1$) pixels p:

$$\psi_{\mathcal{P}}(\mathcal{X}) = \sum_{i \in N_k} \sum_{p \in P_k} \left\| \varphi_p^{ki}\, w_p^i \left[\mathcal{I}_i(\pi_i^k(\mathbf{x}_p^k)) - \mathcal{R}_i(\mathbf{x}_p^k, \mathbf{n}_p^k, f_p^k)\right] \right\|_1 \tag{4}$$

Here, $\mathcal{R}_i$ denotes the rendering equation [159] for image i. Since we assume a single point light source, $\mathcal{R}_i$ simplifies to

$$\mathcal{R}_i(\mathbf{x}_p^k, \mathbf{n}_p^k, f_p^k) = f_p^k\left(\mathbf{n}_p^k, \boldsymbol{\omega}_{\text{in}}^i(\mathbf{x}_p^k), \boldsymbol{\omega}_{\text{out}}^k(\mathbf{x}_p^k)\right) \frac{a_i(\mathbf{x}_p^k)\, \mathbf{n}_p^{k^T} \boldsymbol{\omega}_{\text{in}}^i(\mathbf{x}_p^k)}{d_i(\mathbf{x}_p^k)^2}\, L \tag{5}$$

where $\boldsymbol{\omega}_{\text{in}}^i(\mathbf{x}_p^k)$ and $\boldsymbol{\omega}_{\text{out}}^k(\mathbf{x}_p^k)$ denote the in- and out-going light directions for the surface point $\mathbf{x}_p^k$, $a_i(\mathbf{x}_p^k)$ is the angle-dependent light attenuation, $d_i(\mathbf{x}_p^k)$ the distance between $\mathbf{x}_p^k$ and the light source and L denotes the radiant intensity of the light. To down-weight observations at grazing angles, we use a weight $w_p^i \propto \mathbf{n}_p^i \mathbf{l}_p^i$

proportional to the angle between the surface normal $\mathbf{n}_p^i$ and light direction $\mathbf{l}_p^i$.

We calculate the visibility term $\varphi_p^{ki} \in \{0, 1\}$ of surface point $\mathbf{x}_p^k$ in observation view i by reconstructing a rough 3D model of the scene using volumetric fusion of the depth observations and performing a zbuffer test to validate if 3D point $\mathbf{x}_p^k$ is both visible in view i and illuminated by the (calibrated) light source corresponding to keyframe k.

Depth Consistency: We further constrain the depth estimate $\{z_p^k\}_p$ against the depth measurements $\mathcal{Z}_i$ of all neighboring views $i \in N_k$:

$$\psi_{\mathcal{D}}(\mathcal{X}) = \sum_{i \in N_k} \sum_{p \in P_k} \varphi_p^{ki} \left\| z_p^i - \mathcal{Z}_i(\pi_i^k(\mathbf{x}_p^k)) \right\|_2^2 \tag{6}$$

Here, z_p^i denotes the depth of the 3D point $\mathbf{x}_p^k$ of keyframe k when projected to the neighbor view i via $\pi_i^k(\mathbf{x}_p^k)$. As above, φ_p^{ki} ensures that surface point $\mathbf{x}_p^k$ is visible in image i.

Note that our model is able to significantly improve upon the initial coarse geometry provided by the structured light sensor by exploiting shading cues. However, as these cues are related to depth variations (i.e., normals) rather than absolute depth, they do not fully constrain the 3D shape of the object. Our experiments demonstrate that combining complementary depth and shading cues yields reconstructions which are both locally detailed and globally consistent.

4.2 Multi-View Consistency

Since our representation is composed of multiple 2.5D views, we must ensure consistency between them. Towards this goal, we augment our objective with a multi-view consistency term which encourages the current parameter estimates $\{z_p^k, \mathbf{d}_p^k, r_p^k, s_p^k\}_{p \in P_k}$, $\{\mathbf{t}_j^k, \mathbf{q}_j^k\}_{j \in N_k}$ of keyframe k to agree with those of neighboring keyframes $\{\bar{z}_p^k, \bar{\mathbf{d}}_p^k, \bar{r}_p^k, \bar{s}_p^k\}_{p \in P_k}, \{\bar{\mathbf{t}}^i, \bar{\mathbf{q}}^i\}_{i \in \bar{N}_k}$ projected into the current keyframe:

$$\begin{aligned}\psi_{\mathcal{C}}(\mathcal{X}) = \frac{1}{|P_k|} \sum_{p \in P_k} & \left\| \mathbf{x}_p^k - \bar{\mathbf{x}}_p^k \right\|_2 + \left\| \mathbf{d}_p^k - \bar{\mathbf{d}}_p^k \right\|_1 \\ & + \left| r_p^k - \bar{r}_p^k \right| + \left| s_p^k - \bar{s}_p^k \right| \\ + \frac{1}{n} \sum_{i \in \bar{N}_k} \sum_{j \in N_i \cap N_k} & \left\| \mathbf{t}_j^k - \mathbf{t}_j^i \right\|_1 + \left\| \left(\left(\mathbf{q}_j^i \right)^{-1} \otimes \mathbf{q}_j^k \right)_v \right\|_1 \end{aligned} \tag{7}$$

Hereby, the projected neighbor parameters $\{\bar{z}_p^k, \bar{\mathbf{d}}_p^k, \bar{r}_p^k, \bar{s}_p^k\}_{p \in P_k}$ for each surface point $\mathbf{x}_p^k$ are computed as follows

$$(\bar{\cdot})_p^k = \frac{1}{\sum_i \varphi_p^{ki} \, w_p^i} \sum_{i \in \bar{N}_k} \varphi_p^{ki} \, w_p^i \, \text{interp}\left(\{(\cdot)_q^i\}_q, \pi_i(\mathbf{x}_p^k) \right) \tag{8}$$

where $\bar{N}_k$ is the set of neighboring keyframes, φ_p^{ki} denotes visibility as defined above, and $w_p^i = \mathbf{n}_p^i \mathbf{l}_p^i$ downweights estimates at grazing angles. The mapping interp : $\mathbb{R}^{|P|} \times \mathbb{R}^2 \to \mathbb{R}$ takes a neighboring parameter map $\{(\cdot)_q^i\}_{q \in P_i}$ and a projected 2D pixel location $\pi_i(\mathbf{x}_p^k)$ and outputs the bilinearly interpolated parameter value. The poses of all neighboring keyframes $\{\bar{\mathbf{t}}^i, \bar{\mathbf{q}}^i\}_{i \in \bar{N}_k}$ are defined as

$$\bar{\mathbf{t}}^i = \left\{ \mathbf{t}_j^i \mid j \in N_i \cap N_k \right\} \quad \text{and} \quad \bar{\mathbf{q}}^i = \left\{ \mathbf{q}_j^i \mid j \in N_i \cap N_k \right\} \tag{9}$$

The $\otimes$ operator in the last term of (7) denotes the Hamiltonian product for quaternions and calculates the composed rotation. Furthermore, $(\cdot)_v$ denotes the vector part of the quaternion which equals zero for the identity. Note that we only calculate the pose loss term for cameras which are part of the observations for both the current keyframe k and the neighboring keyframes i, i.e., $j \in N_i \cap N_k$.

4.3 Geometry Regularization

Our geometry regularizers encourage geometric consistency $\psi_{\mathcal{GC}}$ and normal smoothness $\psi_{\mathcal{N}}$ as follows:

$$\psi_{\mathcal{G}} = \psi_{\mathcal{GC}} + \psi_{\mathcal{N}} \tag{10}$$

As above, we omitted the dependency on $\mathcal{X}$ and relative weights of the individual terms for clarity.

Geometric Consistency: We enforce consistency between depth $\{z_p^k\}$ and normals $\{\mathbf{n}_p^k\}$ by maximizing the inner product between the estimated normals $\{\mathbf{n}_p^k\}$ and the cross product of the surface tangents at $\{\mathbf{x}_p^k\}$:

$$\psi_{\mathcal{GC}}(\mathcal{X}) = -\sum_p \left(\mathbf{n}_p^k \right)^T \frac{\frac{\partial z_p^k}{\partial x} \times \frac{\partial z_p^k}{\partial y}}{\left\| \frac{\partial z_p^k}{\partial x} \times \frac{\partial z_p^k}{\partial y} \right\|_2} \tag{11}$$

where the surface tangent $\frac{\partial z_p^k}{\partial x}$ is given by

$$\frac{\partial z_p^k}{\partial x} \propto \left[1, 0, \nabla \mathcal{Z}_k(\pi_k(\mathbf{x}_p^k))^T \left[f / z_p, 0 \right]^T \right]^T \tag{12}$$

with $\nabla \mathcal{Z}_k(\pi_k(\mathbf{x}_p^k))$ the gradient of the depth map estimated using finite differences. We refer to [7] for details.

Normal Smoothness: We further encourage normals of adjacent pixels $p \sim q$ to be similar:

$$\psi_{\mathcal{N}}(\mathcal{X}) = \sum_{p \sim q} e_{pq}^k \left\| \mathbf{n}_p^k - \mathbf{n}_q^k \right\|_1 \tag{13}$$

Here, e_{pq}^k is an edge-aware weighting term based on a Canny filter [160] to reduce the smoothing at pixels close to edges in the albedo map and facilitate detailed geometry reconstruction.

4.4 Material Smoothness

To enforce propagation of reflectance parameters across pixels, we constrain the specular albedo $\{s_p^k\}_p$ and roughness $\{r_p^k\}_p$ maps against a bilaterally smoothed version of themselves:

$$\begin{aligned}\psi_{\mathcal{M}}(\mathcal{X}) = \sum_p & \left\| s_p^k - \frac{\sum_q s_q^k \, w_q^k g_{pq}^k}{\sum_q w_q^k g_{pq}^k} \right\|_1 \\ & + \left\| r_p^k - \frac{\sum_q r_q^k \, w_q^k g_{pq}^k}{\sum_q w_q^k g_{pq}^k} \right\|_1 \end{aligned} \tag{14}$$

Assuming that nearby pixels with similar diffuse behavior also exhibit similar specular behavior, we use a Gaussian kernel g_{pq}^k with both the 3D location $\mathbf{x}$ and diffuse albedo $\mathbf{d}$ at pixels p and q as features:

$$g_{pq}^k = \exp\left(-\frac{(\mathbf{x}_p^k - \mathbf{x}_q^k)_2^2}{2\sigma_1^2} - \frac{(\mathbf{d}_p^k - \mathbf{d}_q^k)_2^2}{2\sigma_2^2} \right) \tag{15}$$

The weight $w_q^k = \max_i \cos^{-1}(\mathbf{n}_q^{k^T} \mathbf{h}_q^{ki})$ with half-vector $\mathbf{h}_q^{ki}$ increases the contribution of pixels q which are observed close to perfect mirror reflection in any view i and are therefore most informative for specular material estimation. We use the permutohedral lattice [161] for efficient evaluation of Eq. (14).

Data: Color and depth images $\{\mathcal{I}_i, \mathcal{Z}_i\}_{i \in N}$.
Result: Mesh $\mathcal{M}$ featuring per-vertex normals and BRDF parameters.

Initialize $\forall k \in K$: // Section 5.3
$\mathcal{X}_0^k = \{z_p^k, \mathbf{n}_p^k, \mathbf{d}_p^k, r_p^k, s_p^k\}_{p \in P_k}$ and $\{\pi_i^k\}_{i \in N_k}$
$\bar{\mathcal{X}}_0^k = \textit{None}$

$t = 100,\ T = 2000$
$\textit{rounds} = T/t$

Multi-View Consistent Optimization:
for $r = 1$ **to** *rounds* **do**
  for *keyframe* $k \in K$ **do**
    Optimize $\mathcal{X}_{r-1}^k$ given $\bar{\mathcal{X}}_{r-1}^k$ for t iterations:
    $\mathcal{X}_r^k = \mathcal{X}_{r-1}^{k,*}$ // Eq. (3)
    Project neighbor parameter maps: // Eq. (8)
    $\bar{\mathcal{X}}_r^k = \{\bar{z}_p^k, \bar{\mathbf{d}}_p^k, \bar{r}_p^k, \bar{s}_p^k\}_{p \in P_k}$ and $\{\bar{\pi}_i^k\}_{i \in \bar{N}_k}$
  end
end

Mesh Generation: // Section 6
Fuse all final keyframe parameter maps $\{\mathcal{X}_{r=T/t}^k\}_{k \in K}$ into a mesh $\mathcal{M}$ by volumetric fusion and marching cubes.

Algorithm 1: Pseudo-Code of the proposed Algorithm.

5 Optimization

Direct optimization of the global objective in Eq. (3) does not scale to larger scenes due to the large amount of data (variables and observations) that need to be stored and GPU memory limitations. Instead, we decompose the global reconstruction into multiple keyframe reconstructions and perform decentralized, frame-wise block coordinate descent in parallel on multiple processes. With this distributed optimization strategy, we drastically reduce the memory footprint since we only ever need to store one block in memory at a time per process. To keep locally adjacent blocks consistent, we periodically share the state of optimization variables between neighboring keyframes and regularize differences in the reconstructed models. An overview of the full optimization algorithm is given in Algorithm 1.

In the following, we first motivate our decentralized optimization strategy and then elaborate on the block coordinate descent algorithm. Subsequently, we provide details about the parameter initialization and our implementation.

5.1 Decentralized Optimization

For large computational problems, a **distributed optimization strategy** that allows for multiple processes and parallelization is essential. As per [162], distributed methods can be categorized into centralized and decentralized algorithms, depending on whether the processes read and update a central copy of the optimization variables or work on independent local copies. Centralized algorithms require consistent transaction management such as semaphores, cause additional computational cost for centralization and distribution, and exhibit less stable optimization behavior due to potentially contradicting updates to the central optimization variables by different processes. Therefore we implement a decentralized algorithm that facilitates accurate local reconstructions. As this might result in multiple different estimates per variable, we introduce a soft regularizer to establish synchronization between processes and encourage consistency across spatially nearby regions. We reduce the communication overhead by employing a strategy similar to [163] and letting each process independently perform a set of base optimization steps in between synchronization. In the following, we call this set of optimization steps performed for all processes a "round".

5.2 Block Coordinate Descent Optimization

For optimization, we represent the target scene as a set of 2.5D parameter maps induced by the keyframes. The keyframes can be viewed as blocks of variables over which we iterate as described in the following:

We start the first round by optimizing over every block/keyframe for t iterations independently using gradient descent. Given these intermediate optimization states $\{\mathcal{X}_1^k\}_k$ of all keyframes in K, the current parameter map estimates are projected into neighboring keyframes. We refer to them as $\{\bar{\mathcal{X}}_1^k\}_k$. They are used in the subsequent round when the parameters of all blocks/keyframes are re-optimized for t iterations with the additional multi-view consistency regularizer. We iterate these rounds of optimizing for $\{\mathcal{X}_r^k\}_k$ and calculating $\{\bar{\mathcal{X}}_r^k\}_k$ for a total of T/t rounds and T iterations. During the first half of optimization, we use a higher geometric smoothing regularizer to help bootstrapping the parameter maps. Thereafter, the smoothness regularization is reduced to allow for carving out fine geometric details and modeling sharp specular highlights during the remaining iterations.

5.3 Initialization

We initialize the **poses** using the SfM pipeline COLMAP [14], [164]. For the **depth** initialization we pre-integrate the rather coarse data of an active stereo setup into a fused 3D geometry using volumetric fusion [165] and render an initial depth map per keyframe $k \in K$, see Fig. 10 for an example of the initial geometry. Initial **diffuse albedo** and **normal** maps can be computed in closed form assuming a Lambertian scene. Towards this goal, we follow [3] and robustly filter outliers due to specularities using RANSAC. Both **specular albedo** and **roughness** parameter maps are initialized by sampling randomly and uniformly from the intervals $[0.05, 0.25]$ and $[0.1, 0.9]$, respectively.

5.4 Implementation details

We have implemented the rendering function $\mathcal{R}_i$ using PyTorch [166], exploiting PyTorch's GPU acceleration and auto-differentiation capabilities. For our experiments we use $|K| = 24$ keyframes, $m = 20$ neighboring observation views and $\bar{m} = 10$ neighboring keyframes. We project parameter maps into neighboring keyframes every $t = 100$ iterations and optimize for $T = 2000$ iterations in total. Please see the supplement for more details about the optimization.

6 Mesh Generation

In order to obtain a full 3D reconstruction of geometry and materials we use a memory efficient, voxel hashing based implementation of volumetric fusion [143] as seen in [148]. Since predictions are most accurate for pixels observed frontally, we weight each pixel contribution by the cosine between surface normal and viewing direction. We extract the final mesh from

	Fixed Poses	Full Model
Photometric Test Error	1.210	1.138

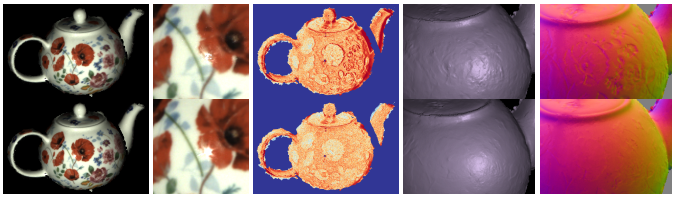


Fig. 4: **Pose Optimization in 2.5D,** results from Schmitt et al. [7]. Compared to using the input poses (top), optimizing the poses (bottom) improves reconstruction quality significantly.

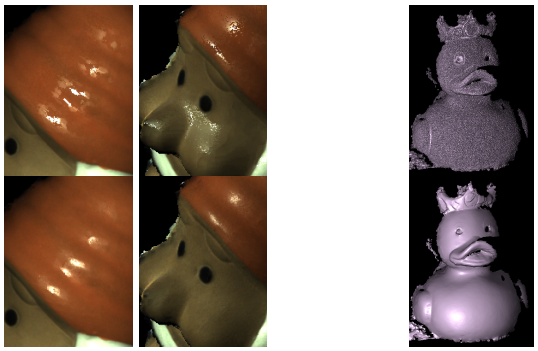


Fig. 5: **Loss Regularizers in 2.5D,** results from Schmitt et al. [7]. Shown are Reconstructions of held-out test views. Without regularization (top), appearance and geometry is inconsistent or noisy. Using the regularization terms (bottom), information is propagated across the object, successfully generalizing to new illumination conditions on the test set.

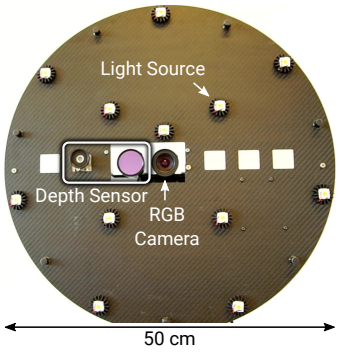


Fig. 6: **Sensor Rig.** Our custom-made handheld capture device features a high resolution RGB camera, a Kinect-like active depth sensor and 12 high-power LEDs (modeled as point light sources) that surround the camera in two circles (with radii 10 cm and 25 cm).

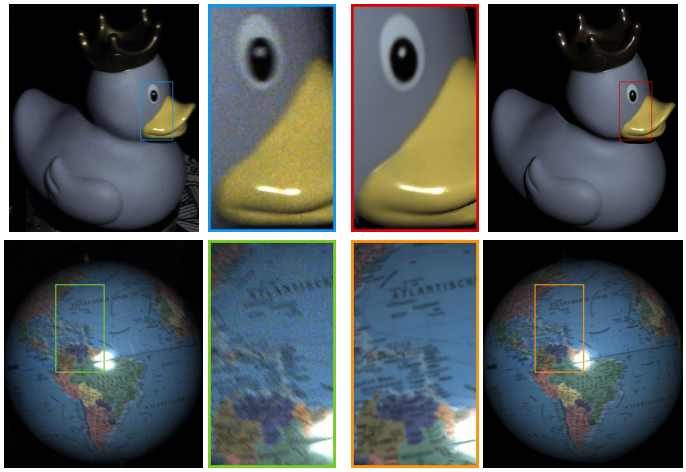


Fig. 7: **Super-Resolution and Denoising (3D).** With a hand-held capture system, the measured observations exhibit image noise and motion blur (left) which our model is able to remove. The resulting reconstructions appear denoised and sharpened (right).

the TSDF via marching cubes [167]. As we impose consistency across keyframes during optimization, the fused parameter maps are consistent without need for extra post-processing/alignment steps. Finally, the resulting mesh allows for extracting a texture map for each svBRDF parameter using Blender and exporting the mesh into the OBJ file format. During all our experiments, we use a voxel size of 0.5 - 2mm.

7 Experimental Evaluation

The proposed method is an extension of the 2.5D reconstruction algorithm presented by Schmitt et al. [7] to full 3D models. Due to space limitations, we do not repeat all experiments conducted in our conference paper, but instead review the main conclusions and insights. For further details, we kindly refer the reader to [7].

In [7], we propose a formulation for joint recovery of camera pose, object geometry and spatially-varying BRDF from handheld capture data. Instead of using multiple decoupled objectives and treating materials and geometry separately as done before, we demonstrate in the conference paper that this problem can be formulated using a single objective function and off-the-shelf gradient-based solvers. Except for a few minor differences described at the end of this section, this model corresponds to the model described in Section 4 when excluding the multi-view consistent optimization and instead optimizing only a single keyframe, yielding a 2.5D result. As shown in the ablation study of Schmitt et al. [7], two components are particularly important for accurate appearance and geometry reconstructions: 1) Optimizing poses jointly with the other parameters is crucial for disambiguating geometric properties from materials. We repeat the ablation results in Fig. 4. 2) The proposed regularization terms for the geometric parameters and material maps enable joint optimization over all parameters using a single objective function. Hereby, the material smoothness term is able to propagate material information over large distances and compensates for the sparse measurements of the BRDF per pixel. And the geometric consistency term enforces consistency between depth and normals and prevents high-frequency structure artifacts. The results are shown in Fig. 5.

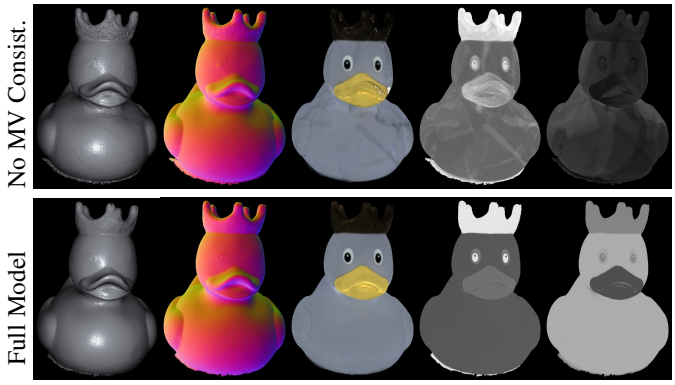
In this paper, we demonstrate that simple fusion of multiple 2.5D reconstructions obtained using [7] is insufficient to obtain accurate 3D models of an object. In particular, the geometry from different keyframes does not align well enough and material predictions from different viewpoints often differ noticeably due to the ambiguities present in this inverse problem. We therefore introduce a multi-view consistent optimization scheme in this extension, and demonstrate that this enables consistent and accurate reconstructions of 3D models. We further demonstrate that this allows for modeling larger scenes beyond single objects. In comparison to [7], we also make several minor improvements to the model which we empirically found to be useful: 1) We change the material model to a more flexible and practical

	Fixed Poses	Full Model
Photometric Test Error	18.767	17.8035

TABLE 1: **Pose Optimization (3D).** Similar to our conclusions in 2.5D (Schmitt et al. [7] and Fig. 4), pose optimization also improves the fused 3D results of our full model.

	No MV Consistency	Full Model
Photometric Test Error	14.65	13.295

(a) RMSE on held-out test views, average over 3 objects.



(b) Qualitative Comparison of Estimated Parameter Maps.

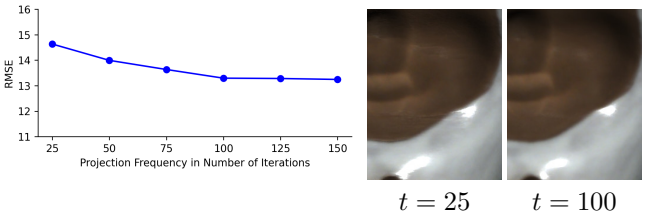
Fig. 8: **The Multi-View Consistency Loss (3D)** facilitates consistent parameter predictions across keyframes resulting in more accurate reconstructions (a). In (b) we show parameter maps of the 3D fused mesh. Without the multi-view consistency regularizer (top), geometric artifacts are visible and the BRDF parameter maps show patch-like structures as well as baked-in shading information on very glossy object parts (e.g. beak and crown). In contrast, with our loss (bottom), the mesh is clean and reflectance maps are homogeneous per object part.

solution for larger scenes that does not require material clustering and model selection, 2) we regularize the depth maps against all neighboring depth maps instead of a single one to stay closer to the measurements and 3) we include an edge-aware weighting term in the normal smoothness loss to facilitate reconstruction of small details.

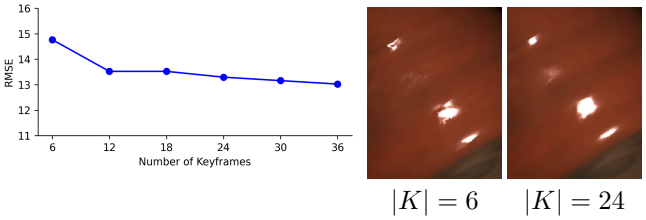
In the following, we present the results of our extended model in 3D and provide an evaluation on captures of real objects and scenes from a custom built handheld sensor rig. We first introduce our hardware system and the data capture procedure and then provide details on our evaluation protocol. Afterwards, we conduct an ablation study of the components of our method. We then provide qualitative and quantitative comparisons to related approaches and conclude with reconstruction results for our captured dataset. Note that unlike in [7] and in Fig. 4, 5, all of the following results show fused 3D models unless explicitly stated otherwise. Further, we present results of our method on synthetic data in the supplement.

7.1 Setup

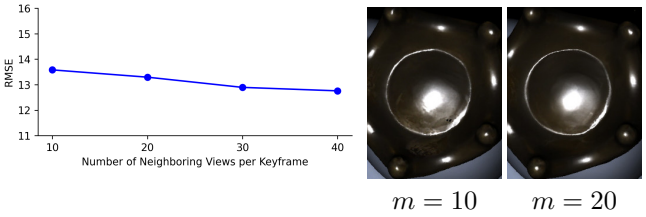
For capturing data, we use a custom-built handheld sensor rig as shown in Fig. 6. We thoroughly calibrate the system in advance,



(a) Multi-View Projection Frequency



(b) Number of Keyframes



(c) Number of Neighboring Observation Views per Keyframe

Fig. 9: **Multi-View Optimization (3D)**. For multiple parameters we show the average error over 3 objects wrt. the parameter on the left, and example predictions on the right. Hereby, the left image is a rendered result for the worst choice of parameters and the right image shows a rendered result for our chosen parameters.

both geometrically and photometrically. We estimate camera instrinsics, response, distortion and vignetting, the relative positions of the depth sensor and light sources wrt. the RGB camera, and the angular attenuation behavior and radiant intensities of the light sources.

We slowly move our sensor around the scene and alternate the illumination such that each image is illuminated by exactly one light source. We assume a completely darkened room with negligible ambient light. Examples of captured raw data is shown in Fig. 7 (left). Note that due to the handheld setup, we need to accept a certain amount of image noise to trade-off motion blur. But we show in Fig. 7 (right) that our model is able to predict denoised and sharpened reconstructions. If the scene is not sufficiently textured, we additionally add texture patterns to the scene to ensure enough feature points and obtain more reliable initial pose estimates.

We use full image resolution (4K) for all objects and half image resolution (2K) for scenes due to GPU memory limitations. For all our objects and scenes, we captured between 800 - 1400 images.

Evaluation Protocol: For quantitative evaluation, we render the final model in 10 held-out test views and compute the photometric loss with respect to the observation.

As our model can deviate from the coarse initial COLMAP poses during optimization, we first align the test view poses to the predicted model. Next, we compute the Root Mean Square Error over all pixels that have a non-zero color prediction for each test view and report the mean loss. Since for real capture data

there exist no ground truth object masks, we define valid image pixels as pixels with non-zero prediction (prediction mask). For validation, we draw the observation masks by hand for multiple objects and find that the RMS errors for both evaluation masks differ by $< 5\%$. Therefore, we use the prediction mask in the following for all experiments for simplicity.

7.2 Ablation Study

In this section, we ablate the important parts of our model, both qualitatively and quantitatively. An additional ablation on the loss weights can be found in the supplementary.

Multi-View Consistency Loss: Multiple local reconstructions of our method share the coarse but consistent captured depth maps, but observe different samples of the reflectance function since the captured images contain only sparse measurements thereof. That implies that while the 2.5D keyframe optimizations lead to consistent geometry with respect to immediate neighbors, they may lead to inconsistencies wrt. keyframes that are not optimized. And these inconsistencies cannot be resolved reliably using volumetric fusion which merely averages multiple geometries. Thus, there is no guarantee for consistent reconstruction results without explicitly enforcing consistency between keyframes.

Fig. 8 demonstrates the effectiveness of the multiview consistency loss. It encourages both regularization and propagation between keyframes. Synchronization of parameter estimates during optimization enables our method to find an equilibrium of the variables which is consistent with not only its neighboring observation views but also the neighboring observation views of nearby keyframes. Hereby, these connections between neighboring keyframes form a connected graph over all keyframes. Therefore, consistency between any two keyframes can be penalized during optimization, which is enforcing global consistency. Without the multi-view consistency loss term, the fused model is not aligned well enough to form one coherent surface or consistent BRDF parameter maps. In contrast, we observe that our distributed multi-view consistent optimization leads to globally consistent results without blending artifacts as illustrated in Fig. 8b. We note that, in particular, specular properties (specular albedo and roughness) are robustly reconstructed despite the sparsely sampled reflectance function and the high sensitivity of specular highlights to angular configurations. This leads to lower reconstruction errors as evidenced in Table 8a.

Pose Optimization: Misaligned camera poses yield wrong correspondences between view pairs and can cause various reconstruction artifacts such as ghosting, blur, and texture/geometry bleeding between front and back surfaces. With respect to material reconstruction, wrong pose estimates lead to errors in the prediction of angular relations between the surface normals, view and light directions. This causes estimated specular highlights to not align with the mirror reflection direction and hence leads to highlights not being recovered and bake-in effects of specular appearance into predicted texture and normal maps. Such pose alignment problems are particularly crucial when working with a moving handheld scanner. Therefore, we optimize the camera poses jointly with the other parameters leading to more consistent results and lower reconstruction errors. We show these findings quantitatively for our fused 3D models in Table 1, confirming the 2.5D results in Fig. 4 from Schmitt et al. [7].

Multi-View Parameter Projection Frequency: In a single optimization round, we optimize the parameters of all keyframes for t iterations before synchronizing with neighboring keyframes. Therefore, t balances local reconstruction quality and global parameter consistency. A low number of t or high synchronization frequency hinders the local optimizations to fit the neighboring observations as shown in Fig. 9a (right), whereas a low frequency or no synchronization ($t = T$) prevents consistency among the *current* parameter estimates of neighboring keyframes, as discussed in Fig. 8. We found that $t = 100$ leads to both accurate parameter estimates and consistent results across keyframes.

Number of Keyframes: Fig. 9b plots test accuracy against the number of keyframes $|K|$. We observe that generally, more keyframes lead to more accurate reconstructions and most affected by a small number of keyframes is the quality of the predicted highlights. This offers several insights as it indicates that 1) local keyframe results are globally consistent also for larger numbers of keyframes, 2) details are preserved and the geometry is not noticeably blurred during mesh fusion and 3) synchronization with neighboring keyframes is important for correct reflectance estimation (with a reduced number of keyframes, the number of possible neighboring keyframes decreases as well). We use $|K| = 24$ in the following for all single objects as the performance gain becomes very small thereafter.

Number of Neighboring Observation Views: Our goal is to estimate the spatially varying BRDF but we only observe a very sparse set of samples for each surface point $\mathbf{x}$. As expected, we see in Fig. 9c that reducing the number of neighboring observation views m worsens this problem. However, interestingly, this effect is quite small and the error degrades gracefully. We attribute this to our multi-view consistent optimization scheme which regularly provides information from neighboring keyframes during optimization. Therefore, given a sufficiently large number of keyframes, our method produces accurate predictions already for $m = 20$.

7.3 Comparisons to Existing Approaches

We compare our model with TSDF fusion [165], the 2.5D optimization approach by Schmitt et al. [7] and the 3D reconstruction method from Nam et al. [5]. Hereby, we qualitatively evaluate our reconstructions in terms of geometric details, material modeling as well as overall appearance prediction.

Geometry Reconstruction: In Fig. 10 we compare the geometry reconstruction capabilities of our 3D model to the 2.5D method by Schmitt et al. [7] and naïve 3D TSDF fusion [165] of the raw depth maps. We show that both photometric approaches are able to recover fine geometric structures that are not present in the initial reconstruction. Further, in contrast to Schmitt et al. [7], the proposed method recovers the geometry for shiny and dark surfaces like the eyes of the 'Owl'. Such materials are very challenging for photometric approaches since the signal-to-noise ratio of the diffuse component is low and the signal from specular highlights is very sparse. Since our multi-view consistent optimization shares information between keyframes, it is often able to reconstruct such problematic regions.

Comparison to Schmitt et al. [7]: We present a qualitative and quantitative comparison to [7], the conference paper that we extend in this paper. Since the method jointly reconstructs pose,

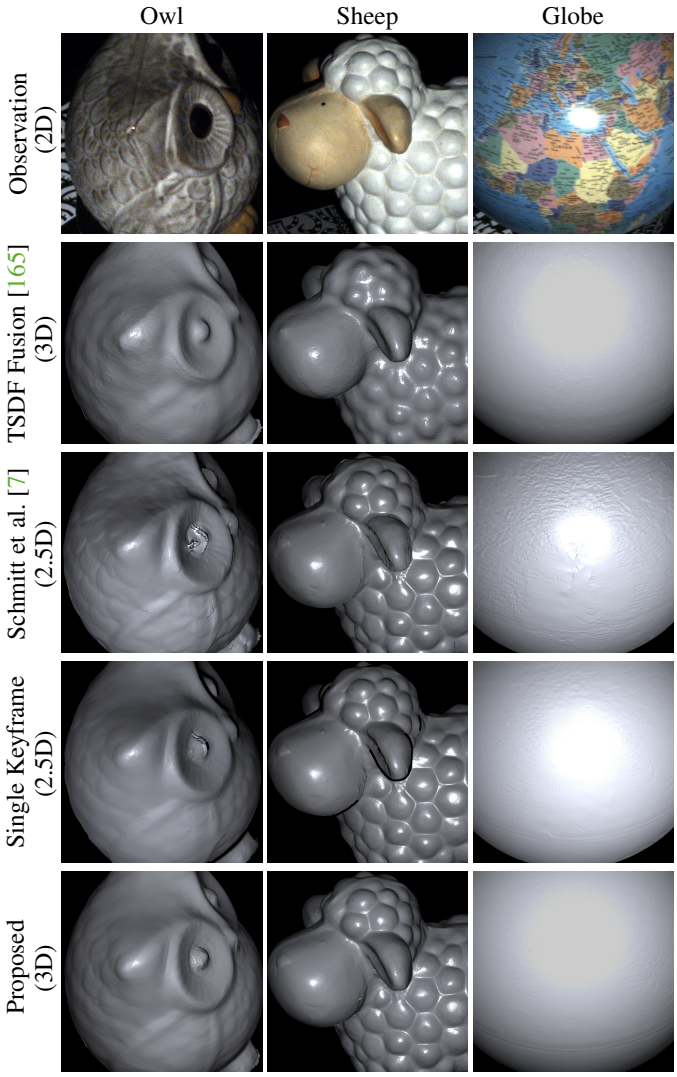
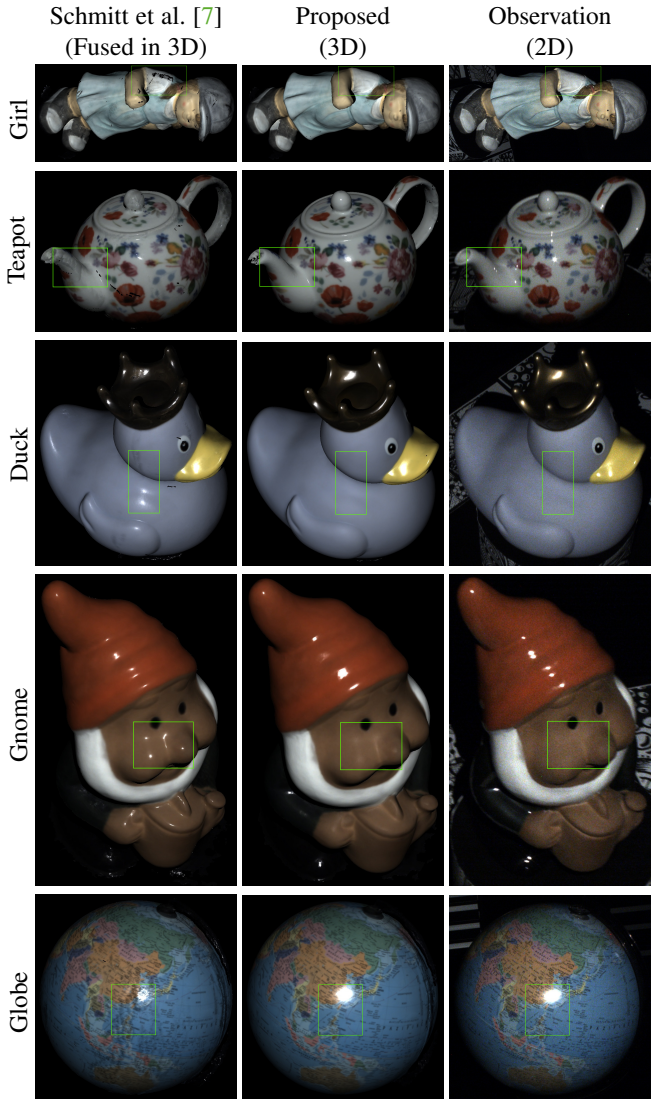


Fig. 10: **Qualitative Geometry Comparison (2.5D vs. 3D).** We show, for each object, the rendered depth map (shaded based on estimated surface normals) for the 3D model after TSDF fusion, the 2.5D model by Schmitt et al. and both a 2.5D keyframe as well as the full 3D model of our proposed method. We observe that the photometric approaches recover more details than naïve TSDF fusion of the input geometry. Thanks to our multi-view consistent optimization scheme, the single keyframe result of the proposed approach contains less textural artifacts in the geometry than Schmitt et al. (see e.g. the 'Globe'). Further, we observe that for our model the resulting global 3D geometry is as detailed as the geometry of the 2.5D keyframe, and additionally resolves artifacts present in the local reconstruction, i.e. the eye of the 'Owl'.



(a) Qualitative Results for Held-Out Test Views

	Girl	Teapot	Duck	Gnome	Globe
Schmitt et al.	15.196	43.755	31.212	64.032	92.765
Proposed	12.281	18.128	10.366	17.483	10.510

(b) Quantitative Results: Photometric Test Error

Fig. 11: **Comparison to Schmitt et al. [7] (3D):** For the same set of keyframes, we executed the proposed method and the independent 2.5D reconstructions as presented in the conference paper [7]. We show both models after volumetric fusion, qualitatively (a) and quantitatively (b) on held-out test views. While our method leads to realistic appearance reconstructions, the predictions of the method by Schmitt et al. (independent optimizations) cannot resolve inconsistencies between keyframes and tend to over-estimate specular parameters (see highlighted regions of 'Duck', 'Gnome' and 'Girl'). Please see Fig. 12 for details.

geometry and materials for local 2.5D scene representations, we execute it independently on all keyframes in K and demonstrate that integration of the results to a fused, global 3D mesh is insufficient: The 2.5D parameter estimates are inconsistent, causing patching artifacts and wrong appearance in the predictions of the integrated 3D models, see Fig. 11a and Fig. 13. The integration in 3D also reveals ambiguities in the estimation of geometric and photometric parameters. Our case study in Fig. 12 shows that the 2.5D model is indeed prone to over-estimate specular reflection: since the appearance is highly sensitive to angular changes in the normals, very small deviations of the normals are sufficient to strongly alter the appearance e.g. by removing specular highlights from the predictions. Therefore, the model is able to "cheat" by tilting normals away instead of decreasing the glossiness of the material. In contrast, our method resolves such ambiguities by incorporating information from neighboring keyframes and encouraging multi-view consistency. Specifically, regularization

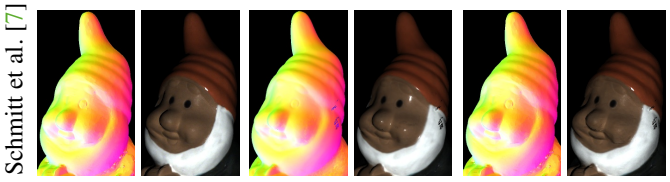


Fig. 12: **Reconstruction Ambiguities for Schmitt et al. [7] (2.5D vs. 3D):** Integration of independent 2.5D reconstructions, as presented in the conference paper [7], into a fused 3D mesh leads to incorrect appearance predictions, see Fig. 11. This can be attributed to unresolved ambiguities between geometry and materials: Shown are the normal maps and rendered predictions for (left) a single 2.5D reconstruction as presented in [7], (middle) the fused 3D mesh after integration of independent 2.5D reconstructions as shown in Fig. 11a, left column and (right) the fused 3D mesh (as in the middle) with only the normal map loaded from the 2.5D keyframe (from the left). We observe that the appearance of the 3D mesh (middle) shows artifacts. But when using the noisier normal map from the 2.5D keyframe reconstruction, these artifacts are reduced noticeably (right). That indicates that the model in Schmitt et al. [7] cannot resolve ambiguities in the normal and specular material estimation. It tends to over-estimate specular parameters (e.g. on the face of the gnome) and slightly perturbs the normal maps as compensation, resulting in good appearance.

against aggregated neighboring parameter maps prevents a bias in the normals and leads to better material estimates. Fig. 11b confirms significantly lower reconstruction errors for our model on held-out test views.

Comparison to Nam et al. [5]: We compare our method with that of Nam et al. [5] on a scene with multiple objects. As shown in Fig. 13, their method reconstructs 3D appearance components such as normals and diffuse albedo properly but does not manage to recover the specular reflections of the given scene well. Since they use base materials for reflectance modeling, this indicates that the clustering into surface regions with distinct materials fails, potentially due to an erroneous estimate of the number of base materials. This leads to non-smooth predictions even for regions with similar appearance and results in uneven specular highlights and high-frequency artifacts. In contrast, our method reconstructs materials and specular highlights well because our pixel-wise material representation does not involve a model selection step.

7.4 Reconstruction Results

We show results of our method on captured objects in Fig. 15 and the supplement and demonstrate the capabilities of our method to reconstruct accurate geometry and materials for a variety of real objects, scenes and materials. Please see videos of our reconstructions here: https://sites.google.com/view/material-fusion/ And results on synthetic data can be found in the supplement.

Towards Scalable Scene Reconstructions: We demonstrate the scalability of the proposed approach in Fig. 14. As shown, our method reconstructs scenes on the scale of several meters at a resolution of $\leq$ 2mm and recovers accurate appearance and geometry, leading to realistic renderings of novel viewpoints and illumination. But the 'Office' scene shows 2 limitations of our model: Since we do not recover missing geometry or fill holes but rely on the completeness of the input geometry, artifacts on the 'Teapot' and the 'Mug' can not be resolved. Additionally, we do not model global illumination effects leading to small artifacts in the diffuse albedo map.

7.5 Limitations

The proposed method refines depth maps but does not compete missing geometry. Therefore, larger holes in the initial geometry can not be filled, see e.g. the wall in the 'Office', Fig. 14. Further, we decided on a renderer that models a single light bounce to keep computation tractable. That means that global illumination cannot be modeled and inter-reflections cause local errors in the material maps, as can be seen in Fig. 15. And last, the expressiveness of our BRDF model is limited. While it is able to represent most objects common in indoor rooms, it does not support anisotropic reflections or subsurface scattering.

8 Conclusion

We have proposed a practical approach to estimating geometry and materials from a handheld sensor for full 3D models that exceed object-scale. Accurate camera poses are crucial to this task, but are not readily available. To tackle this problem, we propose a novel formulation which enables joint optimization of poses, geometry and materials using a single objective. Towards large-scale appearance and geometry reconstructions, we represent the scene as 2.5D parameter maps for a set of keyframes and introduce a distributed optimization scheme. We demonstrate that our multi-view consistency regularization is key to enable accurate integration of the local 2.5D reconstruction results into a consistent 3D model. Our approach recovers accurate geometry and material properties that are globally consistent across the local representations. We demonstrate on multiple scenes with larger compositions of multiple objects that our method takes a step towards scalable multi-view reconstruction of geometry and materials. In future work, we plan to extend our model to handle ambient light and global illumination effects.

Acknowledgments

We thank Giljoo Nam for providing the results of his method [5] on our data. We thank the International Max Planck Research School for Intelligent Systems (IMPRS-IS) for supporting Carolin Schmitt. This work was supported by the Intel Network on Intelligent Systems (NIS). Joo Ho Lee and Andreas Geiger were supported by the ERC Starting Grant LEGO-3D (850533) and the DFG EXC number 2064/1 - project number 390727645. Joo Ho Lee acknowledges the support of the MSIT/NRF of Korea (RS-2022-00165910 and 2023-00212828).

References

[1] S. Bi, Z. Xu, K. Sunkavalli, D. Kriegman, and R. Ramamoorthi, "Deep 3d capture: Geometry and reflectance from sparse multi-view images," in *CVPR*, 2020. 1, 3

[2] D. Lichy, J. Wu, S. Sengupta, and D. W. Jacobs, "Shape and material capture at home," in *CVPR*, 2021. 1

[3] T. Higo, Y. Matsushita, N. Joshi, and K. Ikeuchi, "A hand-held photometric stereo camera for 3-d modeling," in *ICCV*, 2009. 1, 3, 8

[4] S. Georgoulis, M. Proesmans, and L. J. V. Gool, "Tackling shapes and brdfs head-on," in *3DV*, 2014. 1, 3

[5] G. Nam, J. H. Lee, D. Gutierrez, and M. H. Kim, "Practical SVBRDF acquisition of 3d objects with unstructured flash photography," *ACM Trans. on Graphics*, vol. 37, no. 6, pp. 267:1–267:12, 2018. 1, 2, 3, 6, 11, 13, 14

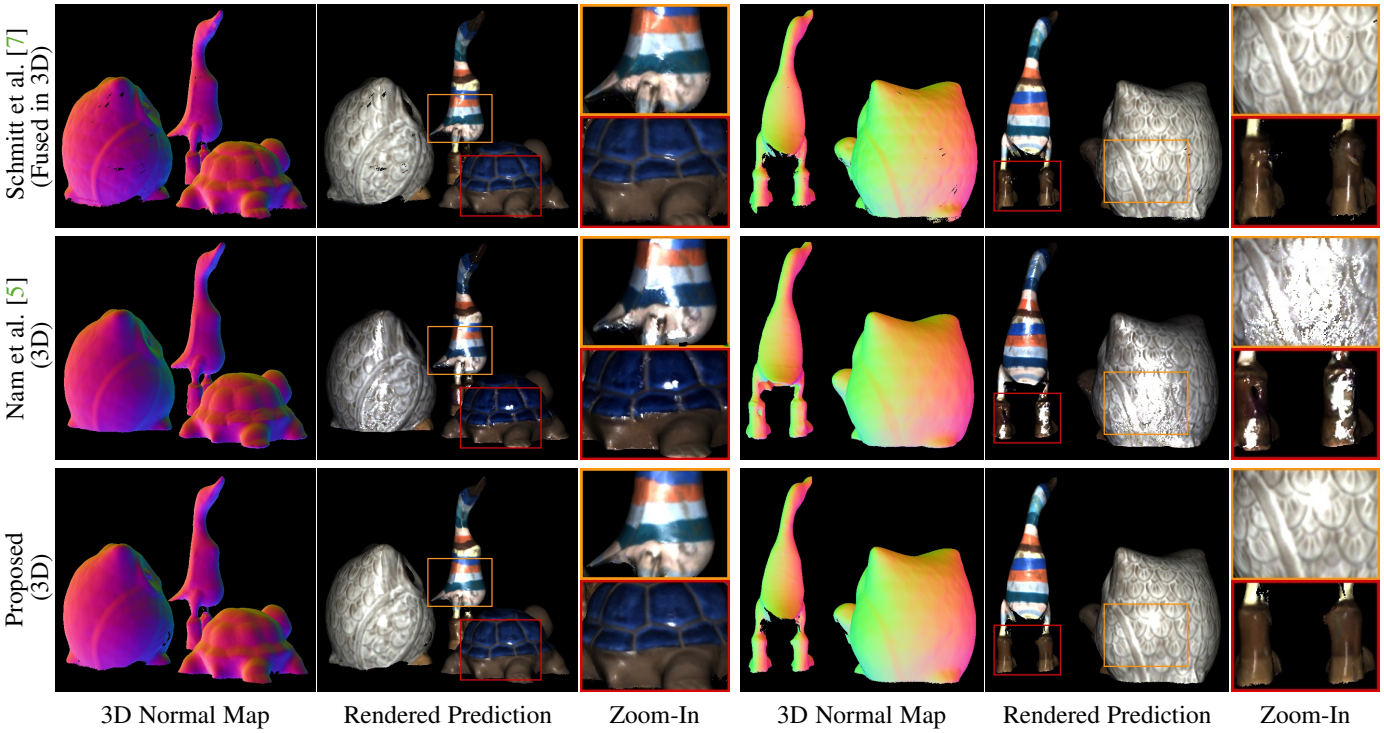


Fig. 13: **Qualitative Comparison to Baselines (3D).** Simple fusion of the 2.5D reconstruction results of Schmitt et al. [7] results in artifacts in the geometry and appearance. While the model from Nam et al. [5] recovers detailed normal maps, the material reconstruction fails to capture the appearance of the object which leads to high-frequency artifacts in the predictions. Our model estimates normals that are on par with the results of Nam et al. [5] and predicts consistent appearance with realistic specular reflections.

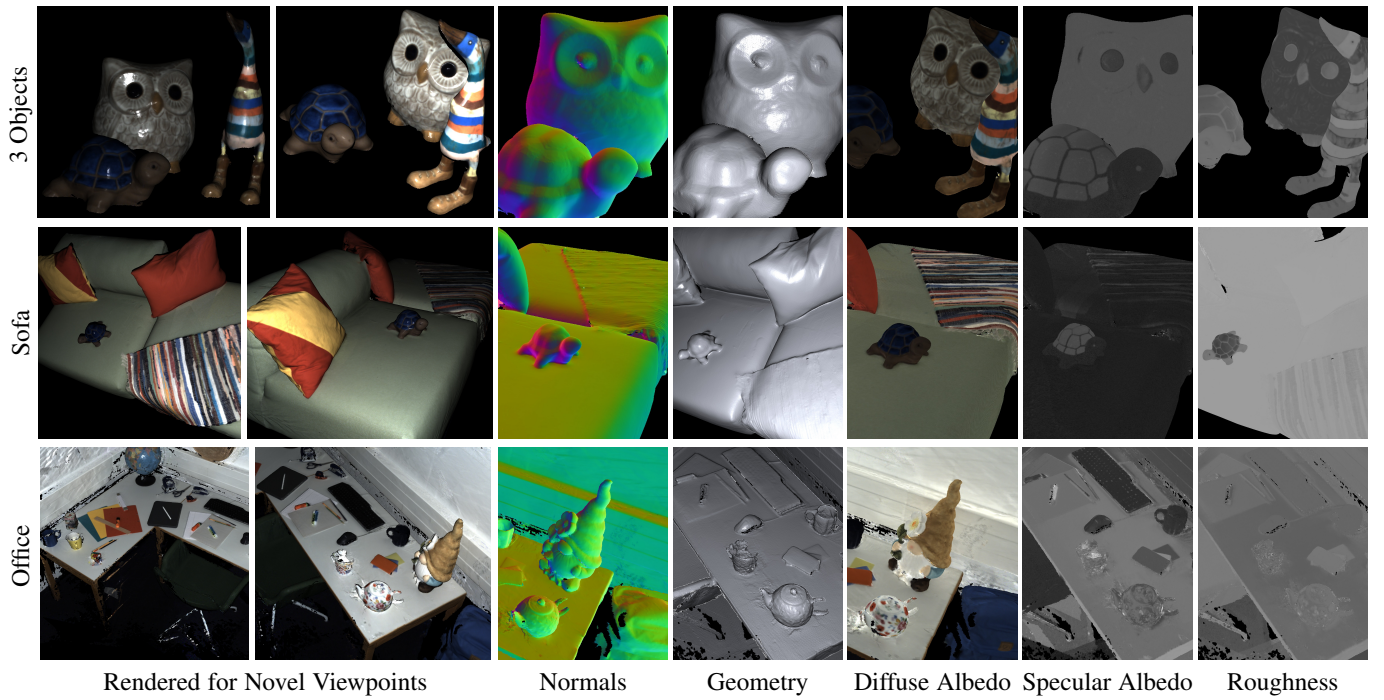


Fig. 14: **Reconstruction Beyond Object-Scale (3D).** We present reconstructions for three challenging scenes that are composed of multiple objects with non-convex shapes, detailed geometries, various different materials, many occlusions and shadows and a spatial size of up to $2 \times 3m$, e.g. the scene 'Office'. Our method reconstructs detailed geometry and accurate materials leading to realistic renderings under new illumination and unseen viewpoints. Hereby, we observe a clean separation of illumination effects and geometric properties, as the material maps are homogeneous per object parts (see e.g. the 'Turtle' in '3 Objects' and 'Sofa').

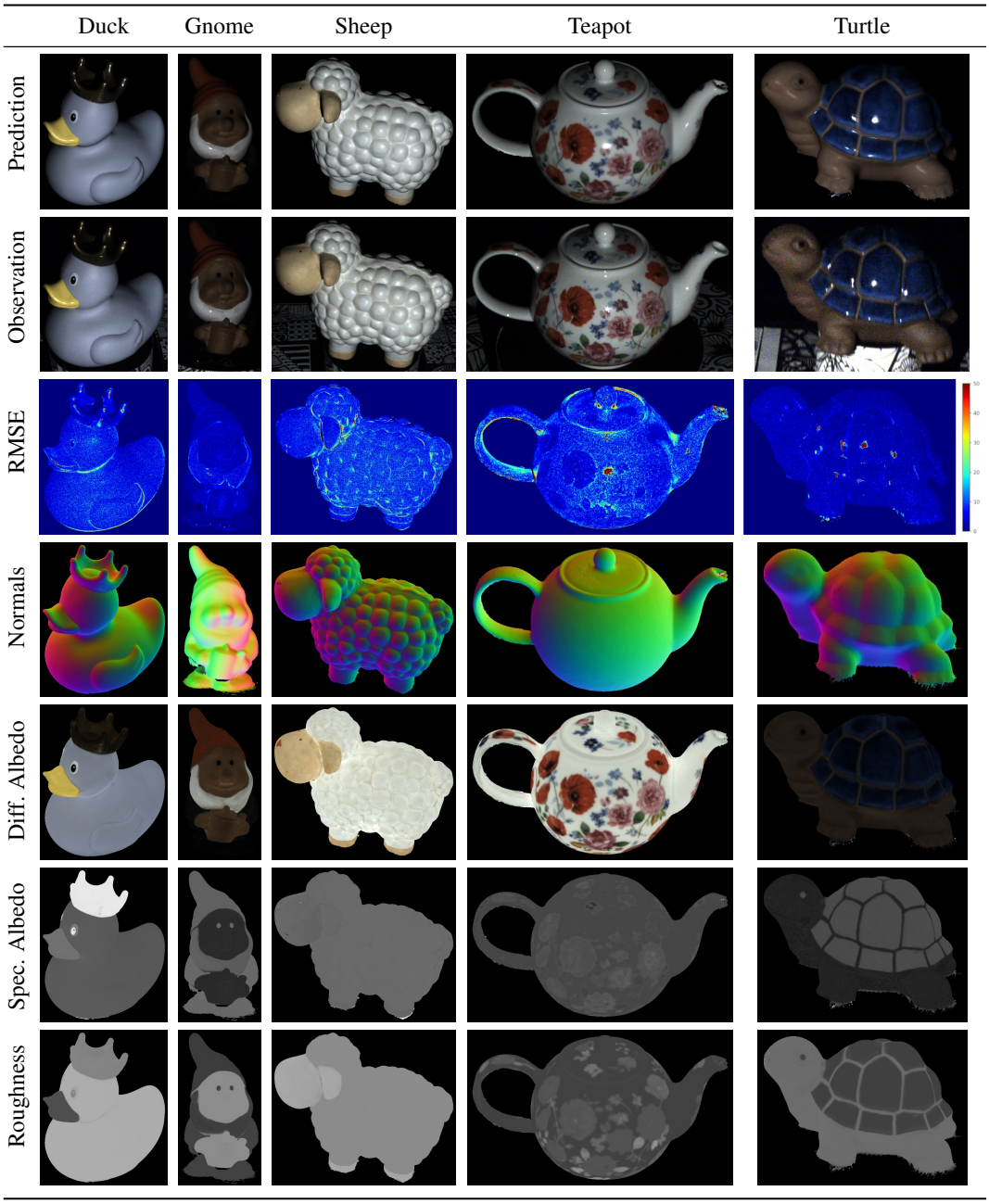


Fig. 15: **Reconstruction Results (3D)** for the objects 'Duck', 'Gnome', 'Sheep', 'Teapot' and 'Turtle'.

[6] Z. Li, T. Yu, S. Sang, S. Wang, S. Bi, Z. Xu, H. Yu, K. Sunkavalli, M. Hasan, R. Ramamoorthi, and M. Chandraker, "Openrooms: An end-to-end open framework for photorealistic indoor scene datasets," in *CVPR*, 2021. 1, 3

[7] C. Schmitt, S. Donne, G. Riegler, V. Koltun, and A. Geiger, "On joint estimation of pose, geometry and svbrdf from a handheld scanner," in *CVPR*, 2020. 1, 2, 3, 7, 9, 10, 11, 12, 13, 14

[8] S. Seitz and C. Dyer, "Photorealistic scene reconstruction by voxel coloring," in *CVPR*, 1997. 2

[9] F. Lafarge, R. Keriven, M. Bredif, and H.-H. Vu, "A hybrid multiview stereo algorithm for modeling urban scenes." *PAMI*, vol. 35, no. 1, pp. 5–17, 2013. 2

[10] A. O. Ulusoy, A. Geiger, and M. J. Black, "Towards probabilistic volumetric reconstruction using ray potentials," in *3DV*, 2015. 2

[11] G. Vogiatzis, P. H. S. Torr, and R. Cipolla, "Multi-view stereo via volumetric graph-cuts," in *CVPR*, 2005. 2

[12] V. Kolmogorov and R. Zabih, "Multi-camera scene reconstruction via graph cuts," in *ECCV*, 2002. 2

[13] Y. Furukawa and J. Ponce, "Accurate, dense, and robust multi-view stereopsis," *PAMI*, vol. 32, no. 8, pp. 1362–1376, 2010. 2

[14] J. L. Schönberger, E. Zheng, M. Pollefeys, and J.-M. Frahm, "Pixelwise view selection for unstructured multi-view stereo," in *ECCV*, 2016. 2, 8

[15] K. Ikeuchi and B. K. P. Horn, "Numerical shape from shading and occluding boundaries," *AI*, vol. 17, no. 1-3, pp. 141–184, 1981. 2

[16] R. Zhang, P. Tsai, J. E. Cryer, and M. Shah, "Shape from shading: A survey," *PAMI*, vol. 21, no. 8, pp. 690–706, 1999. 2

[17] B. K. Horn, "Shape from shading: A method for obtaining the shape of a smooth opaque object from one view bibtex," Tech. Rep., 1970. 2

[18] Y. Quéau, J. Mélou, F. Castan, D. Cremers, and J. Durou, "A variational approach to shape-from-shading under natural illumination," in *EMMCVPR*, 2017. 2

[19] Y. Quéau, J. Mélou, J. Durou, and D. Cremers, "Dense multi-view 3d-reconstruction without dense correspondences," *arXiv.org*, vol. 1704.00337, 2017. 2

[20] B. Haefner, Y. Quéau, T. Möllenhoff, and D. Cremers, "Fight ill-posedness with ill-posedness: Single-shot variational depth super-resolution from shading," in *CVPR*, 2018. 2

[21] C. Wu, M. Zollhöfer, M. Nießner, M. Stamminger, S. Izadi, and C. Theobalt, "Real-time shading-based refinement for consumer depth cameras," in *ACM Trans. on Graphics*, 2014. 2

[22] M. Zollhöfer, A. Dai, M. Innmann, C. Wu, M. Stamminger, C. Theobalt, and M. Nießner, "Shading-based refinement on volumetric signed distance functions," *ACM Trans. on Graphics*, vol. 34, no. 4, pp. 96:1–96:14, 2015. 2

[23] R. Maier, K. Kim, D. Cremers, J. Kautz, and M. Nießner, "Intrinsic3d: High-quality 3d reconstruction by joint appearance and geometry optimization with spatially-varying lighting," in *ICCV*, 2017, pp. 3133–3141. 2

[24] J. T. Barron and J. Malik, "Shape, illumination, and reflectance from shading," *PAMI*, vol. 37, no. 8, pp. 1670–1687, 2015. 2

[25] G. Oxholm and K. Nishino, "Multiview shape and reflectance from natural illumination," in *CVPR*, 2014. 2

[26] S. Lombardi and K. Nishino, "Single image multimaterial estimation," in *CVPR*, 2012. 2

[27] R. J. Woodham, "Photometric method for determining surface orientation from multiple images," *OE*, vol. 19, no. 1, p. 191139, 1980. 2

[28] Y. Quéau, R. Mecca, and J. Durou, "Unbiased photometric stereo for colored surfaces: A variational approach," in *CVPR*, 2016. 2

[29] Y. Quéau, F. Lauze, and J. Durou, "Solving uncalibrated photometric stereo using total variation," *JMIV*, vol. 52, no. 1, pp. 87–107, 2015. 2

[30] T. Papadhimitri and P. Favaro, "Uncalibrated near-light photometric stereo," in *BMVC*, 2014. 2

[31] M. Holroyd, J. Lawrence, G. Humphreys, and T. E. Zickler, "A photometric approach for estimating normals and tangents," *ACM Trans. on Graphics*, vol. 27, no. 5, pp. 133:1–133:9, 2008. 2

[32] Z. Zhou and P. Tan, "Ring-light photometric stereo," in *ECCV*, 2010. 2

[33] B. Tunwattanapong, G. Fyffe, P. Graham, J. Busch, X. Yu, A. Ghosh, and P. E. Debevec, "Acquiring reflectance and shape from continuous spherical harmonic illumination," *ACM Trans. on Graphics*, vol. 32, no. 4, pp. 109:1–109:12, 2013. 2

[34] J. Lim, J. Ho, M. Yang, and D. J. Kriegman, "Passive photometric stereo from motion," in *ICCV*, 2005. 2

[35] D. Simakov, D. Frolova, and R. Basri, "Dense shape reconstruction of a moving object under arbitrary, unknown lighting," in *ICCV*, 2003. 2

[36] F. Logothetis, R. Mecca, and R. Cipolla, "Semi-calibrated near field photometric stereo," in *CVPR*, 2017. 2

[37] Y. Quéau, T. Wu, and D. Cremers, "Semi-calibrated near-light photometric stereo," in *SSVM*, 2017. 2

[38] W. Xie, C. Dai, and C. C. L. Wang, "Photometric stereo with near point lighting: A solution by mesh deformation," in *CVPR*, 2015. 2

[39] C. Liu, S. G. Narasimhan, and A. W. Dubrawski, "Near-light photometric stereo using circularly placed point light sources," 2018. 2

[40] R. Mecca, A. Wetzler, A. M. Bruckstein, and R. Kimmel, "Near field photometric stereo with point light sources," *SIAM*, vol. 7, no. 4, pp. 2732–2770, 2014. 2

[41] Y. Quéau, B. Durix, T. Wu, D. Cremers, F. Lauze, and J. Durou, "Led-based photometric stereo: Modeling, calibration and numerical solution," *JMIV*, vol. 60, no. 3, pp. 313–340, 2018. 2

[42] R. Mecca, A. Tankus, A. Wetzler, and A. M. Bruckstein, "A direct differential approach to photometric stereo with perspective viewing," *SIAM*, vol. 7, no. 2, pp. 579–612, 2014. 2

[43] Y. Quéau, T. Wu, F. Lauze, J. Durou, and D. Cremers, "A non-convex variational approach to photometric stereo under inaccurate lighting," in *CVPR*, 2017. 2

[44] Y. Quéau, F. Lauze, and J. Durou, "A l^1 -tv algorithm for robust perspective photometric stereo with spatially-varying lightings," in *SSVM*, 2015. 2

[45] R. Mecca, E. Rodolà, and D. Cremers, "Realistic photometric stereo using partial differential irradiance equation ratios," *Computers & Graphics*, vol. 51, pp. 8–16, 2015. 2

[46] R. Mecca and Y. Quéau, "Unifying diffuse and specular reflections for the photometric stereo problem," in *WACV*, 2016. 2

[47] R. Mecca, Y. Quéau, F. Logothetis, and R. Cipolla, "A single-lobe photometric stereo approach for heterogeneous material," *SIAM*, vol. 9, no. 4, pp. 1858–1888, 2016. 2

[48] M. K. Chandraker, J. Bai, and R. Ramamoorthi, "A theory of differential photometric stereo for unknown isotropic brdfs," in *CVPR*, 2011. 2

[49] Z. Lu, Y. Tai, M. Ben-Ezra, and M. S. Brown, "A framework for ultra high resolution 3d imaging," in *CVPR*, 2010. 2

[50] D. Nehab, S. Rusinkiewicz, J. Davis, and R. Ramamoorthi, "Efficiently combining positions and normals for precise 3d geometry," *ACM Trans. on Graphics*, vol. 24, no. 3, pp. 536–543, 2005. 2

[51] N. Joshi and D. J. Kriegman, "Shape from varying illumination and viewpoint," in *ICCV*, 2007. 2

[52] H. Fan, L. Qi, J. Dong, G. Li, and H. Yu, "Dynamic 3d surface reconstruction using a hand-held camera," in *IECON*, 2018. 2

[53] J. Park, S. N. Sinha, Y. Matsushita, Y. Tai, and I. S. Kweon, "Robust multiview photometric stereo using planar mesh parameterization," *PAMI*, vol. 39, no. 8, pp. 1591–1604, 2017. 2

[54] B. Shi, K. Inose, Y. Matsushita, P. Tan, S. Yeung, and K. Ikeuchi, "Photometric stereo using internet images," in *3DV*, 2014. 2

[55] F. Logothetis, R. Mecca, and R. Cipolla, "A differential volumetric approach to multi-view photometric stereo," in *ICCV*, 2019. 2

[56] Y. Yoshiyasu and N. Yamazaki, "Topology-adaptive multi-view photometric stereo," in *CVPR*, 2011. 2

[57] H. Barrow, "Recovering intrinsic scene characteristics from images," *CVS*, pp. 3–26, 1978. 2

[58] P. V. Gehler, C. Rother, M. Kiefel, L. Zhang, and B. Schölkopf, "Recovering intrinsic images with a global sparsity prior on reflectance," in *NIPS*, 2011, pp. 765–773. 2

[59] Q. Chen and V. Koltun, "A simple model for intrinsic image decomposition with depth cues," in *ICCV*, 2013. 2

[60] F. E. Nicodemus, J. C. Richmond, J. J. Hsia, I. W. Ginsberg, and T. Limperis, "Geometrical considerations and nomenclature for reflectance," in *Radiometry*. Jones and Bartlett Publishers, Inc., 1992, ch. Geometrical Considerations and Nomenclature for Reflectance, pp. 94–145. 2

[61] W. Matusik, H. Pfister, M. Brand, and L. McMillan, "A data-driven reflectance model," *ACM Trans. on Graphics*, vol. 22, no. 3, pp. 759–769, Jul. 2003. 2

[62] J. B. Nielsen, H. W. Jensen, and R. Ramamoorthi, "On optimal, minimal BRDF sampling for reflectance acquisition," *ACM Trans. on Graphics*, vol. 34, no. 6, pp. 186:1–186:11, 2015. 2

[63] H. P. A. Lensch, J. Kautz, M. Goesele, W. Heidrich, and H. Seidel, "Image-based reconstruction of spatial appearance and geometric detail," *ACM Trans. on Graphics*, vol. 22, no. 2, pp. 234–257, 2003. 2

[64] C. Schwartz, R. Sarlette, M. Weinmann, and R. Klein, "DOME II: A parallelized BTF acquisition system," in *EUROGRAPHICS*, 2013. 2

[65] M. Holroyd, J. Lawrence, and T. E. Zickler, "A coaxial optical scanner for synchronous acquisition of 3d geometry and surface reflectance," *ACM Trans. on Graphics*, vol. 29, no. 4, pp. 99:1–99:12, 2010. 2

[66] R. A. Albert, D. Y. Chan, D. B. Goldman, and J. F. O'Brien, "Approximate svbrdf estimation from mobile phone video," in *EUROGRAPHICS*, 2018. 2

[67] J. Rivière, P. Peers, and A. Ghosh, "Mobile surface reflectometry," *Computer Graphics Forum*, vol. 35, no. 1, pp. 191–202, 2016. 2

[68] M. Aittala, T. Weyrich, and J. Lehtinen, "Two-shot SVBRDF capture for stationary materials," *ACM Trans. on Graphics*, vol. 34, no. 4, pp. 110:1–110:13, 2015. 2

[69] Z. Xu, J. B. Nielsen, J. Yu, H. W. Jensen, and R. Ramamoorthi, "Minimal BRDF sampling for two-shot near-field reflectance acquisition," *ACM Trans. on Graphics*, vol. 35, no. 6, pp. 188:1–188:12, 2016. 2

[70] Y. Dong, G. Chen, P. Peers, J. Zhang, and X. Tong, "Appearance-from-motion: recovering spatially varying surface reflectance under unknown lighting," *ACM Trans. on Graphics*, vol. 33, no. 6, pp. 193:1–193:12, 2014. 3

[71] J. J. Park, R. A. Newcombe, and S. M. Seitz, "Surface light field fusion," in *3DV*, 2018. 3

[72] H. Wu and K. Zhou, "Appfusion: Interactive appearance acquisition using a kinect sensor," *Computer Graphics Forum*, vol. 34, no. 6, pp. 289–298, 2015. 3

[73] Z. Wu, S. Yeung, and P. Tan, "Towards building an RGBD-M scanner," *arXiv.org*, vol. 1603.03875, 2016. 3

[74] H. Wu, Z. Wang, and K. Zhou, "Simultaneous localization and appearance estimation with a consumer RGB-D camera," *VCG*, vol. 22, no. 8, pp. 2012–2023, 2016. 3

[75] J. Mélou, Y. Quéau, J. Durou, F. Castan, and D. Cremers, "Beyond multi-view stereo: Shading-reflectance decomposition," in *SSVM*, 2017, pp. 694–705. 3

[76] ——, "Variational reflectance estimation from multi-view images," *JMIV*, vol. 60, no. 9, pp. 1527–1546, 2018. 3

[77] Y. Yu, P. E. Debevec, J. Malik, and T. Hawkins, "Inverse global illumination: Recovering reflectance models of real scenes from photographs," in *ACM Trans. on Graphics*, 1999, pp. 215–224. 3

[78] Z. Zhou, G. Chen, Y. Dong, D. Wipf, Y. Yu, J. Snyder, and X. Tong, "Sparse-as-possible svbrdf acquisition," *ACM Trans. on Graphics*, vol. 35, no. 6, pp. 189:1–189:12, 2016. 3

[79] B. Haefner, S. Green, A. Oursland, D. Andersen, M. Goesele, D. Cremers, R. Newcombe, and T. Whelan, "Recovering real-world reflectance properties and shading from hdr imagery," in *3DV*, 2021. 3

[80] D. B. Goldman, B. Curless, A. Hertzmann, and S. M. Seitz, "Shape and spatially-varying brdfs from photometric stereo," *PAMI*, vol. 32, no. 6, pp. 1060–1071, 2010. 3

[81] N. G. Alldrin, T. E. Zickler, and D. J. Kriegman, "Photometric stereo with non-parametric and spatially-varying reflectance," in *CVPR*, 2008. 3

[82] A. Hertzmann and S. M. Seitz, "Example-based photometric stereo: Shape reconstruction with general, varying brdfs," *PAMI*, vol. 27, no. 8, pp. 1254–1264, 2005. 3

[83] J. Ackermann, M. Ritz, A. Stork, and M. Goesele, "Removing the example from example-based photometric stereo," in *ECCV Workshops*, 2010. 3

[84] Z. Hui and A. C. Sankaranarayanan, "Shape and spatially-varying reflectance estimation from virtual exemplars," *PAMI*, 2017. 3

[85] S. Peng, B. Haefner, Y. Quéau, and D. Cremers, "Depth super-resolution meets uncalibrated photometric stereo," in *ICCV Workshops*, 2017. 3

[86] N. Birkbeck, D. Cobzas, P. F. Sturm, and M. Jägersand, "Variational shape and reflectance estimation under changing light and viewpoints," in *ECCV*, 2006. 3

[87] C. H. Esteban, G. Vogiatzis, and R. Cipolla, "Multiview photometric stereo," *PAMI*, vol. 30, no. 3, pp. 548–554, 2008. 3

[88] X. Zuo, S. Wang, J. Zheng, and R. Yang, "Detailed surface geometry and albedo recovery from RGB-D video under natural illumination," in *ICCV*, 2017. 3

[89] Z. Zhou, Z. Wu, and P. Tan, "Multi-view photometric stereo with spatially varying isotropic materials," in *CVPR*, 2013. 3

[90] R. Xia, Y. Dong, P. Peers, and X. Tong, "Recovering shape and spatially-varying surface reflectance under unknown illumination," *ACM Trans. on Graphics*, vol. 35, no. 6, pp. 187:1–187:12, 2016. 3

[91] P. Ren, J. Wang, J. Snyder, X. Tong, and B. Guo, "Pocket reflectometry," *ACM Trans. on Graphics*, vol. 30, no. 4, Jul. 2011. [Online]. Available: https://doi.org/10.1145/2010324.1964940 3

[92] Z. Hui, K. Sunkavalli, J. Lee, S. Hadap, J. Wang, and A. C. Sankaranarayanan, "Reflectance capture using univariate sampling of brdfs," in *ICCV*, 2017. 3

[93] K. Kim, J. Gu, S. Tyree, P. Molchanov, M. Nießner, and J. Kautz, "A lightweight approach for on-the-fly reflectance estimation," in *ICCV*, 2017. 3

[94] Z. Li, K. Sunkavalli, and M. Chandraker, "Materials for masses: SVBRDF acquisition with a single mobile phone image," in *ECCV*, 2018. 3, 4

[95] Z. Cheng, H. Li, Y. Asano, Y. Zheng, and I. Sato, "Multi-view 3d reconstruction of a texture-less smooth surface of unknown generic reflectance," in *CVPR*, 05 2021. 3

[96] F. Luan, S. Zhao, K. Bala, and Z. Dong, "Unified shape and svbrdf recovery using differentiable monte carlo rendering," *Eurographics Symposium on Rendering*, 2021. 3

[97] S. Bi, Z. Xu, K. Sunkavalli, M. Hašan, Y. Hold-Geoffroy, D. Kriegman, and R. Ramamoorthi, "Deep reflectance volumes: Relightable reconstructions from multi-view photometric images," 2020. 3

[98] S. Bi, Z. Xu, P. Srinivasan, B. Mildenhall, K. Sunkavalli, M. Haan, Y. Hold-Geoffroy, D. Kriegman, and R. Ramamoorthi, "Neural reflectance fields for appearance acquisition," 2020. 3, 4

[99] Z. Cheng, H. Li, R. I. Hartley, Y. Zheng, and I. Sato, "One ring to rule them all: a simple solution to multi-view 3d-reconstruction of shapes with unknown brdf via a small recurrent resnet," *arXiv.org*, 2021. 3

[100] K. Zhang, F. Luan, Z. Li, and N. Snavely, "IRON: Inverse rendering by optimizing neural sdfs and materials from photometric images," in *CVPR*, 2022. 3

[101] S. Liu, W. Chen, T. Li, and H. Li, "Soft rasterizer: Differentiable rendering for unsupervised single-view mesh reconstruction," in *ICCV*, 2019. 3

[102] N. Ravi, J. Reizenstein, D. Novotny, T. Gordon, W.-Y. Lo, J. Johnson, and G. Gkioxari, "Accelerating 3d deep learning with pytorch3d," *arXiv.org*, vol. abs/2007.08501, 2020. 3

[103] M. M. Loper and M. J. Black, "Opendr: An approximate differentiable renderer," in *ECCV*, 2014. 3

[104] G. Loubet, N. Holzschuch, and W. Jakob, "Reparameterizing discontinuous integrands for differentiable rendering," *ACM Trans. on Graphics*, vol. 38, no. 6, 12 2019. 3

[105] T. Li, M. Aittala, F. Durand, and J. Lehtinen, "Differentiable monte carlo ray tracing through edge sampling," *ACM Trans. on Graphics*, 2018. 3

[106] M. Niemeyer, L. Mescheder, M. Oechsle, and A. Geiger, "Differentiable volumetric rendering: Learning implicit 3d representations without 3d supervision," in *CVPR*, 2020. 3

[107] M. Nimier-David, D. Vicini, T. Zeltner, and W. Jakob, "Mitsuba 2: A retargetable forward and inverse renderer," *ACM Trans. on Graphics*, vol. 38, no. 6, 12 2019. 3

[108] D. Vicini, S. Speierer, and W. Jakob, "Differentiable signed distance function rendering," *ACM Trans. on Graphics*, vol. 41, no. 4, pp. 125:1–125:18, 2022. 3

[109] Z. Li, M. Shafiei, R. Ramamoorthi, K. Sunkavalli, and M. Chandraker, "Inverse rendering for complex indoor scenes: Shape, spatially-varying lighting and svbrdf from a single image," in *CVPR*, 2020, pp. 2475–2484. 4

[110] V. Deschaintre, M. Aittala, F. Durand, G. Drettakis, and A. Bousseau, "Single-image SVBRDF capture with a rendering-aware deep network," *ACM Trans. on Graphics*, vol. 37, no. 4, pp. 128:1–128:15, 2018. 4

[111] M. Boss, V. Jampani, K. Kim, H. P. Lensch, and J. Kautz, "Two-shot spatially-varying brdf and shape estimation," in *CVPR*, 2020. 4

[112] F. Wimbauer, S. Wu, and C. Rupprecht, "De-rendering 3d objects in the wild," in *CVPR*, 2022. 4

[113] Z. Li, J. Shi, S. Bi, R. Zhu, K. Sunkavalli, M. Hašan, Z. Xu, R. Ramamoorthi, and M. Chandraker, "Physically-based editing of indoor scene lighting from a single image," *ECCV*, 2022. 4

[114] Z. Li, L. Wang, X. Huang, C. Pan, and J. Yang, "Phyir: Physics-based inverse rendering for panoramic indoor images," in *CVPR*, 2022. 4

[115] R. Zhu, Z. Li, J. Matai, F. Porikli, and M. Chandraker, "Irisformer: Dense vision transformers for single-image inverse rendering in indoor scenes," in *CVPR*, 2022. 4

[116] Z. Wang, J. Philion, S. Fidler, and J. Kautz, "Learning indoor inverse rendering with 3d spatially-varying lighting," in *ICCV*, 2021. 4

[117] S. M. A. Eslami, D. J. Rezende, F. Besse, F. Viola, A. S. Morcos, M. Garnelo, A. Ruderman, A. A. Rusu, I. Danihelka, K. Gregor, D. P. Reichert, L. Buesing, T. Weber, O. Vinyals, D. Rosenbaum, N. C. Rabinowitz, H. King, C. Hillier, M. M. Botvinick, D. Wierstra, K. Kavukcuoglu, and D. Hassabis, "Neural scene representation and rendering," *Science*, vol. 360, pp. 1204–1210, 2018. 4

[118] M. Meshry, D. B. Goldman, S. Khamis, H. Hoppe, R. Pandey, N. Snavely, and R. Martin-Brualla, "Neural rerendering in the wild," in *CVPR*, 2019. 4

[119] J. Thies, M. Zollhöfer, C. Theobalt, M. Stamminger, and M. Nießner, "Image-guided neural object rendering," in *ICLR*, 2020. 4

[120] T. Nguyen-Phuoc, C. Li, S. Balaban, and Y. Yang, "Rendernet: A deep convolutional network for differentiable rendering from 3d shapes," in *NIPS*, 2018. 4

[121] P. Hedman, J. Philip, T. Price, J.-M. Frahm, G. Drettakis, and G. Brostow, "Deep blending for free-viewpoint image-based rendering," vol. 37, no. 6, pp. 257:1–257:15, 2018. 4

[122] Z. Xu, S. Bi, K. Sunkavalli, S. Hadap, H. Su, and R. Ramamoorthi, "Deep view synthesis from sparse photometric images," *ACM Trans. on Graphics*, vol. 38, no. 4, jul 2019. 4

[123] V. Sitzmann, J. Thies, F. Heide, M. Nießner, G. Wetzstein, and M. Zollhöfer, "Deepvoxels: Learning persistent 3d feature embeddings," in *CVPR*, 2019. 4

[124] D. Gao, G. Chen, Y. Dong, P. Peers, K. Xu, and X. Tong, "Deferred neural lighting: free-viewpoint relighting from unstructured photographs," *ACM Trans. on Graphics*, vol. 39, no. 6, p. 258, 12 2020. 4

[125] Z. Xu, K. Sunkavalli, S. Hadap, and R. Ramamoorthi, "Deep image-based relighting from optimal sparse samples," *ACM Trans. on Graphics*, vol. 37, no. 4, jul 2018. 4

[126] J. Philip, M. Gharbi, T. Zhou, A. Efros, and G. Drettakis, "Multi-view relighting using a geometry-aware network," *ACM Trans. on Graphics*, vol. 38, no. 4, July 2019. 4

[127] B. Mildenhall, P. P. Srinivasan, M. Tancik, J. T. Barron, R. Ramamoorthi, and R. Ng, "NeRF: Representing scenes as neural radiance fields for view synthesis," in *ECCV*, 2020. 4

[128] J. Zhang and E. Ohn-Bar, "Learning by watching," in *CVPR*, June 2021. 4

[129] M. Shafiei, S. Bi, Z. Li, A. Liaudanskas, R. O. Cayon, and R. Ramamoorthi, "Learning neural transmittance for efficient rendering of reflectance fields," in *BMVC*, 2021, p. 45. 4

[130] M. Boss, R. Braun, V. Jampani, J. T. Barron, C. Liu, and H. P. Lensch, "Nerd: Neural reflectance decomposition from image collections," in *ICCV*, 2021. 4

[131] X. Zhang, P. P. Srinivasan, B. Deng, P. Debevec, W. T. Freeman, and J. T. Barron, "Nerfactor: Neural factorization of shape and reflectance under an unknown illumination," *ACM Trans. on Graphics*, vol. 40, no. 6, 12 2021. 4

[132] P. Srinivasan, B. Deng, X. Zhang, M. Tancik, B. Mildenhall, and J. T. Barron, "NeRV: Neural reflectance and visibility fields for relighting and view synthesis," in *CVPR*, 2021. 4

[133] M. Boss, V. Jampani, R. Braun, C. Liu, J. T. Barron, and H. P. Lensch, "Neural-pil: Neural pre-integrated lighting for reflectance decomposition," in *NeurIPS*, 2021. 4

[134] M. Boss, A. Engelhardt, A. Kar, Y. Li, D. Sun, J. T. Barron, H. P. Lensch, and V. Jampani, "SAMURAI: Shape And Material from Unconstrained Real-world Arbitrary Image collections," in *NeurIPS*, 2022. 4

[135] Z. Kuang, K. Olszewski, M. Chai, Z. Huang, P. Achlioptas, and S. Tulyakov, "NeROIC: Neural object capture and rendering from online image collections," vol. abs/2201.02533, 2022. 4

[136] J. Munkberg, J. Hasselgren, T. Shen, J. Gao, W. Chen, A. Evans, T. Mueller, and S. Fidler, "Extracting Triangular 3D Models, Materials, and Lighting From Images," *arXiv.org*, 2021. 4

[137] Y. Zhang, J. Chen, and D. Huang, "Cat-det: Contrastively augmented transformer for multi-modal 3d object detection," in *CVPR*, 2022. 4

[138] Y. Yao, J. Zhang, J. Liu, Y. Qu, T. Fang, D. McKinnon, Y. Tsin, and L. Quan, "Neilf: Neural incident light field for physically-based material estimation," in *ECCV*, 2022. 4

[139] J. Y. Zhang, G. Yang, S. Tulsiani, and D. Ramanan, "NeRS: Neural reflectance surfaces for sparse-view 3d reconstruction in the wild," in *NeurIPS*, 2021. 4

[140] J. Huang, A. Dai, L. Guibas, and M. Nießner, "3dlite: Towards commodity 3d scanning for content creation," *ACM Trans. on Graphics*, 2017. 4

[141] D. Gallup, M. Pollefeys, and J.-M. Frahm, "3d reconstruction using an n-layer heightmap," in *Joint Pattern Recognition Symposium*. Springer, 2010, pp. 1–10. 4

[142] P. Hedman, T. Ritschel, G. Drettakis, and G. Brostow, "Scalable Inside-Out Image-Based Rendering," vol. 35, no. 6, pp. 231:1–231:11, 2016. 4

[143] B. Curless and M. Levoy, "A volumetric method for building complex models from range images," in *ACM Trans. on Graphics*, 1996. 4, 8

[144] R. A. Newcombe, S. Izadi, O. Hilliges, D. Molyneaux, D. Kim, A. J. Davison, P. Kohli, J. Shotton, S. Hodges, and A. Fitzgibbon, "Kinectfusion: Real-time dense surface mapping and tracking," in *ISMAR*, 2011. 4

[145] T. Whelan, M. Kaess, M. F. andH. Johannsson, J. Leonard, and J. McDonald, "Kintinuous: Spatially extended KinectFusion," in *RSSWORK*, 2012. 4

[146] P. Henry, D. Fox, A. Bhowmik, and R. Mongia, "Patch volumes: Segmentation-based consistent mapping with rgb-d cameras," in *3DV*, 2013, pp. 398–405. 4

[147] J. Chen, D. Bautembach, and S. Izadi, "Scalable real-time volumetric surface reconstruction," *ACM Trans. on Graphics*, vol. 32, no. 4, pp. 1–16, 2013. 4

[148] M. Nießner, M. Zollhöfer, S. Izadi, and M. Stamminger, "Real-time 3d reconstruction at scale using voxel hashing," in *ACM Trans. on Graphics*, 2013. 4, 8

[149] A. Dai, M. Nießner, M. Zollöfer, S. Izadi, and C. Theobalt, "Bundlefusion: Real-time globally consistent 3d reconstruction using on-the-fly surface re-integration," 2017. 4

[150] M. Pollefeys, "Detailed real-time urban 3d reconstruction from video," *IJCV*, vol. 78, no. 2-3, pp. 143–167, July 2008. 4

[151] S. Donne and A. Geiger, "Learning non-volumetric depth fusion using successive reprojections," in *CVPR*, 2019. 4

[152] S. Choi, Q. Zhou, and V. Koltun, "Robust reconstruction of indoor scenes," in *CVPR*, 2015. 4

[153] Q. Zhou and V. Koltun, "Color map optimization for 3d reconstruction with consumer depth cameras," in *ACM Trans. on Graphics*, 2014. 4

[154] S. Bi, N. K. Kalantari, and R. Ramamoorthi, "Patch-based optimization for image-based texture mapping," *ACM Trans. on Graphics*, vol. 36, no. 4, 2017. 4

[155] Z. Zhang, "Flexible camera calibration by viewing a plane from unknown orientations," in *ICCV*, 1999. 6

[156] R. L. Cook and K. E. Torrance, "A reflectance model for computer graphics," *ACM Trans. on Graphics*, vol. 1, no. 1, pp. 7–24, 1982. 6

[157] B. Burley, "Physically-based shading at Disney," Walt Disney Animation Studios, Tech. Rep., 2012. 6

[158] W. Jakob, "Mitsuba renderer," 2010, http://www.mitsuba-renderer.org. 6

[159] J. T. Kajiya, "The rendering equation," in *ACM Trans. on Graphics*, 1986. 6

[160] J. Canny, "A computational approach to edge detection," *PAMI*, vol. 8, no. 6, pp. 679–698, November 1986. 7

[161] A. Adams, J. Baek, and M. A. Davis, "Fast high-dimensional filtering using the permutohedral lattice," *Computer Graphics Forum*, vol. 29, no. 2, pp. 753–762, 2010. 7

[162] M. Assran, A. Aytekin, H. Feyzmahdavian, M. Johansson, and M. Rabbat, "Advances in asynchronous parallel and distributed optimization," *arXiv.org*, 2020. 8

[163] J. Wang, V. Tantia, N. Ballas, and M. G. Rabbat, "Slowmo: Improving communication-efficient distributed SGD with slow momentum," in *ICLR*, 2020. 8

[164] J. L. Schönberger and J.-M. Frahm, "Structure-from-motion revisited," in *CVPR*, 2016. 8

[165] A. Zeng, S. Song, M. Nießner, M. Fisher, J. Xiao, and T. Funkhouser, "3dmatch: Learning local geometric descriptors from rgb-d reconstructions," in *CVPR*, 2017. 8, 11, 12

[166] A. Paszke, S. Gross, S. Chintala, G. Chanan, E. Yang, Z. DeVito, Z. Lin, A. Desmaison, L. Antiga, and A. Lerer, "Automatic differentiation in PyTorch," in *NIPS Workshops*, 2017. 8

[167] W. E. Lorensen and H. E. Cline, "Marching cubes: A high resolution 3d surface construction algorithm," in *ACM Trans. on Graphics*, 1987. 9

Carolin Schmitt is a research engineer with the Optics and Sensing Laboratory at the Max Planck Institute for Intelligent Systems (MPI-IS). Previously, she was a Ph.D. student with the Autonomous Vision Group led by Prof. Andreas Geiger at the University of Tübingen and the MPI-IS, as a scholar of the IMPRS-IS Research School. Her research interests include computer vision, computer graphics and 3D.

Božidar Antić is a Ph.D. student with the Autonomous Vision Group led by Prof. Andreas Geiger at the University of Tübingen, as part of the ELLIS PhD and PostDoc program. He received his M.Sc. degree in Machine Learning at the University of Tübingen, and his B.Sc. degree in Informatics at the University of Belgrade. His research interests are 3D reconstruction, inverse rendering, and deep representation learning.

Andrei Neculai is a Machine Learning Engineer at Booking.com in Amsterdam, working on cancellation and revenue prediction. In the past, he was a Software Development Engineer for Amazon.com, working on projects ranging from developer tools to the detection of hazardous products. He received his M.Sc. degree in Machine Learning at the University of Tübingen and his B.Eng. degree in Computer Engineering at the Technical University of Iași.

Joo Ho Lee is an assistant professor of Sogang university and supervises the Visual Computing Laboratory. He worked as a postdoctoral researcher at University of Tuebingen and Max Planck Institute before. He received his Ph.D. in computer science from KAIST in 2020. He served reviewers of conference programs such as SIGGRAPH and CVPR. His research interests include computer graphics, 3D reconstruction and computer vision.

Andreas Geiger is a professor at the University of Tübingen. Prior to this, he was a visiting professor at ETH Zürich and a group leader at the Max Planck Institute for Intelligent Systems. He studied at KIT, EPFL and MIT, and received his PhD degree in 2013 from the Karlsruhe Institute of Technology (KIT). He is an ELLIS fellow and coordinates the ELLIS PhD and PostDoc program. His research interests are at the intersection of computer vision, machine learning and robotics, with a particular focus on 3D scene perception, deep representation learning, generative models and reconstruction of 3D geoametry and materials.