High-Quality RGB-D Reconstruction via Multi-View Uncalibrated Photometric Stereo and Gradient-SDF

Lu Sang[1,2]   Björn Häfner[1,2]   Xingxing Zuo[1,*]

Daniel Cremers[1,2]

[1]Technical University of Munich

[2]Munich Center for Machine Learning

{lu.sang, bjoern.haefner, Xingxing.Zuo, cremers}@tum.de

Abstract

Fine-detailed reconstructions are in high demand in many applications. However, most of the existing RGB-D reconstruction methods rely on pre-calculated accurate camera poses to recover the detailed surface geometry, where the representation of a surface needs to be adapted when optimizing different quantities. In this paper, we present a novel multi-view RGB-D based reconstruction method that tackles camera pose, lighting, albedo, and surface normal estimation via the utilization of a gradient signed distance field (gradient-SDF). The proposed method formulates the image rendering process using specific physically-based model(s) and optimizes the surface's quantities on the actual surface using its volumetric representation, as opposed to other works which estimate surface quantities only near the actual surface. To validate our method, we investigate two physically-based image formation models for natural light and point light source applications. The experimental results on synthetic and real-world datasets demonstrate that the proposed method can recover high-quality geometry of the surface more faithfully than the state-of-the-art and further improves the accuracy of estimated camera poses[1].

1. Introduction

Detailed surface reconstruction from 2D images and depth is a challenging topic in computer vision, with increased attention in recent years. It is required to not only reconstruct the rough shape of an object but also to recover the rich texture and fine geometric details of the surface. The reconstructed results can be used in many applications,

---

*corresponding author

[1]source code available https://github.com/Sangluisme/PSgradientSDF

input RGB   init reconstruction   init texture

estimated albedo   refined reconstruction   refined texture
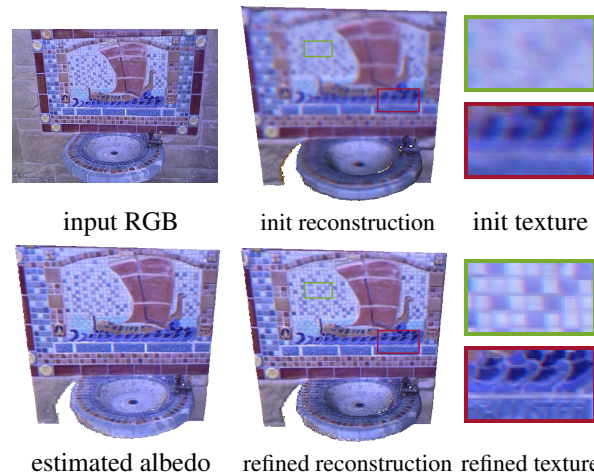
Figure 1. First row: one example input RGB image [45] and the initial reconstruction with zoomed-in detail texture. Second row: estimated albedo, refined reconstruction with zoom-in detail textures of the proposed method. We jointly estimate camera pose, surface normal, albedo and lighting to enable a fine-scaled 3D reconstruction.

such as 3D printing [10], virtual reality [14], digital media [31], etc. To recover a fully detailed 3D model, camera poses and surface quantities such as normal and reflectance (albedo) must be estimated. Instead of assuming known camera poses and directly optimizing surface quantities only *near* the surface, in this paper, we propose to tackle the surface refinement problem together with camera poses, lighting, and albedo via utilization of gradient-SDF [38], a volumetric representation, allowing us to *optimize surface quantities directly on the surface*. In summary, we propose the following main contributions:

- A novel formulation of physically realistic image model(s) compatible with a volumetric representation which enables effective optimization on actual surface

points.

- A complete reconstruction pipeline with the capability of camera pose, geometry refinement, albedo and lighting estimation.

- A method that enables camera pose refinement while taking the lighting condition into account.

- A method that deals with natural and point light source scenarios in an uncalibrated setting which are convenient to adopt in real applications.

Moreover, extensive examination of our approach on both synthetic and real-world datasets demonstrates that it can perform high-quality reconstruction of fine-scaled geometry, albedo, lighting, and camera tracking.

2. Background and Related work

Two critical problems must be considered when recovering a 3D model: the choice of surface representation and the underlying image formation model together with its inverse rendering techniques.

A surface is a 2D manifold and the surface representation strategy is how to discretize, store and update the continuous surface. It can mainly be divided into two categories, explicit representation and implicit representation. The explicit representation, such as point clouds [37], surfels [36], or meshes [21], store the exact surface point locations, allowing for operations on the surface point itself. The implicit representation, such as signed distance function (SDFs) [27], volume density [24] or occupancy [29], on the other hand, only store related properties of each unit, e.g., the distance to the surface [25]. Different representations are appreciated by different goals. For instance, camera tracking and refining, i.e., bundle adjustment, as well as image rendering models benefit from the explicit representation because the models are built on the actual surface. Geometry-related reconstruction methods, such as Kinect-Fusion [25] prefer implicit representation because a uniform and smooth surface can be extracted from it, while easily allowing topological changes during optimization. Therefore, many works alternate between two presentations to deal with different parameters such as camera pose and surface quantities [26, 17].

For the surface modeling, nowadays, there are different methods for recovering a rough 3D model, such as depth fusion-based methods [25] or RGB-image-based structure-from-motion technique [34]. However, the reconstructed 3D model lacks desired geometry detail. To further recover the fine-scale geometry, several strategies are applied. For example, by improving the accuracy of the input, e.g., depth quality [32, 13] or camera poses accuracy [36, 26]. Alternatively, by increasing the resolution of the chosen surface

representation, e.g., the work from Lee el at. [17] introduces a texture map attached to each voxel and subdivides the texture map for a higher texture representation, but the geometry resolution is not improved. Nevertheless, these methods do not use the underlying physical relation between the RGB images, lighting condition, and the surface geometry.

In recent years, *Photometric Stereo* (PS) [41] methods are widely applied in different research fields, such as geometry recovery [22, 8, 32] and image rendering [33, 44]. A PS model describes the physical imaging process, which reflects the irradiance and the outgoing radiance of surfaces. The irradiance is affected by lighting conditions, while the radiance typically depends on the surface material and normal. Therefore, using the image formation model to formulate the rendering equation, the desired surface properties such as surface normal, texture, and material can be recovered [4]. However, using one single RGB image to restore the desired quantities is an ill-posed problem [7]. To overcome the ill-posedness, one can use several RGB images as input [12, 22, 32]. Those images can be at the same view point [28], but to recover a complete 3D model, one needs the images from different camera positions. Existing algorithms that use PS and RGB-D sequences to recover a full 3D model [8, 22, 46] need pre-calculated camera poses. These works follow the pipeline that first integrates the depth with known camera poses to a volumetric cube, then re-projects each voxel back to the images and minimizes the PS energy. They archive good results; still, there are two drawbacks. First, as Figure 2 shows, they actually evaluate all optimization quantities at the voxel center instead of the surface point. Even with smaller voxel sizes, this gap might be reduced, but smaller voxel sizes limit the 3D model's size or introduce heavy computational overhead when the grid size needs to be enlarged. Most works use different regularizers to reduce the artifices caused by this gap. For instance, re-projecting a voxel to the depth map [8, 22, 46] to constraint the voxel distance updating, or constrainting the total variations of reflectance [8, 22]. The second drawback is that most of the works rely on an independent camera tracking algorithm for real-world datasets to get the initialized camera poses. These tracking algorithms either use pure depth information [25] or assume consistent light conditions across the input RGB images [26, 9, 34]. Thus, they do not employ color information, or their assumption contradicts the image formation model.

Notably, there is another class of methods: neural rendering. The success of the neural rendering methods leads to a boom of learning-based methods for image-based synthesis and geometry recovery. Some neural rendering methods also employ PS methods, but most of these approaches [43, 44] focus on view synthesis rather than geometry refinement, and their inputs are only RGB images with accurate camera poses. The work of Lin et al. [18] enables
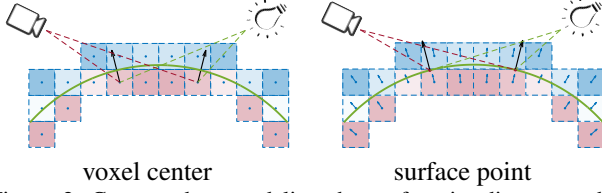
| voxel center | surface point |

Figure 2. Compared to modeling the surface irradiance on the voxel center in a volumetric surface representation (left), the proposed method models the scene more precisely on the actual surface (right) by moving from the voxel center along the surface normal direction.

camera pose refinement, still it does not focus on recovering geometry details or estimating the reflectance and lighting. The recent RGB-D neural rendering work [2] does not apply to most real-world datasets since it assumes single focal length camera model. To the best of our knowledge, only these two papers [42, 40] concentrate on the recovery of the geometry and filling the gap between voxel representation and surface points representation in a slightly different way. They propose to transfer the signed distance value to the density of the point and then use the accumulated color along the ray to recover a good 3D geometry, hence the surface representation as points and volume are combined. Yet, they learn each frame independently rather than explicitly model the lighting and surface normal as PS methods, Even so, they do not refine camera poses and have a long training time.

In this paper, we propose a method that tackles the fine-detailed 3D reconstruction problem together with camera pose estimation in a complete pipeline that stays in one consistent surface representation and assumption. Moreover, we estimate the surface quantities at the actual surface and jointly optimize camera pose after initializing the camera pose and coarse surface volume using gradient-SDF [38].

3. Image Formation Model

3.1. Surface Reflectance

Photometric stereo techniques study the reflected light from the surface together with the environment lighting, the surface material, and normal based on the physical laws to model the produced color images. It describes the outgoing radiance of a surface point $\mathbf{x} \in \mathbb{R}^3$ by integrating over the upper hemisphere $\mathbb{S}^2_+$ around the surface normal $\mathbf{n} \in \mathbb{S}^2$. The pixel intensity that is conjugated with this point on the image is proportional to its irradiance. The outgoing direction can be regarded as the camera view direction. Therefore, for an image $\mathbf{I}$, the color of the pixel $\mathbf{p} \in \mathbb{R}^2$ conjugated with 3D point $\mathbf{x} \in \mathbb{R}^3$ is

$$\mathbf{I}(\mathbf{p}(\mathbf{x})) \approx \int_{\mathbb{S}^2_+} \boldsymbol{\rho}(\mathbf{x}, \mathbf{i})\mathbf{L}(\mathbf{i}, \mathbf{x}) \max(\langle \mathbf{i}, \mathbf{n}(\mathbf{x})\rangle, 0)\mathrm{d}\mathbf{i}, \quad (1)$$

where $\boldsymbol{\rho} : \mathbb{R}^3 \times \mathbb{S}^2 \times \mathbb{S}^2 \rightarrow \mathbb{R}^3$ is the bidirectional reflectance distribution (BRDF) function with 3 color channels, and $\mathbf{L} : \mathbb{S}^2 \times \mathbb{R}^3 \rightarrow \mathbb{R}^3$ is the incoming lighting radiance from direction $\mathbf{i} \in \mathbb{S}^2$ at point $\mathbf{x} \in \mathbb{R}^3$ [15]. Assuming a constant BRDF, i.e., a Lambertian surface, (1) can be simplified as

$$\mathbf{I}(\mathbf{p}(\mathbf{x})) = \boldsymbol{\rho}(\mathbf{x}) \int_{\mathbb{S}^2_+} \mathbf{L}(\mathbf{i}, \mathbf{x}) \max(\langle \mathbf{i}, \mathbf{n}(\mathbf{x})\rangle, 0)\mathrm{d}\mathbf{i}. \quad (2)$$

Given enough color images, the surface normal $\mathbf{n}$ and the surface reflectance $\boldsymbol{\rho}$ can be therefore restored. The crucial challenge is to approximate the integral over the hemisphere reasonably. Different models are proposed to simplify the integral to solve the equation. Here we introduce two models that deal with two application scenarios: natural lighting and point light source illumination.

Natural Light Spherical Harmonics Model A *natural light* source situation is, for example, the light source is the sun, or when the light sources are in a distance. The lighting directions that reach the surface are nearly parallel, then the environment lighting can be well modeled using Spherical Harmonics (SH) functions. The integral part is approximated by the sum of SH basis [4, 5].

$$\mathbf{I}(\mathbf{p}(\mathbf{x})) \approx \boldsymbol{\rho}(\mathbf{x})\langle \mathbf{l}, \mathrm{SH}(\mathbf{n}(\mathbf{x}))\rangle, \quad (3)$$

where $\mathbf{l} \in \mathbb{R}^4$ is an 4-dimensional lighting vector for the current view and $\mathrm{SH}(\mathbf{n}(\mathbf{x})) \in \mathbb{R}^4$ are the first-order SH basis functions for a fixed $\mathbf{n}(\mathbf{x})$. The model is simply formulated and still can reach relatively high accuracy with a low order model.

Point Light Source Model Apart from the natural light, another commonly encountered scenario is the point light source situation, mainly when focusing on small object reconstruction. The object is usually illuminated by a *point light source*, e.g., a LED light that is close to the object. The lighting can hardly be regarded as a set of parallel lines, thus point light source provides more changes to the object illumination, which is preferred to deal with the ill-posedness of the PS model. One widely used point light source-light model is [19, 23, 30].

$$\mathbf{I}(\mathbf{p}(\mathbf{x})) = \Psi^s \boldsymbol{\rho}(\mathbf{x})(\frac{\langle \mathbf{n}^s, \mathbf{l}^s\rangle}{\|\mathbf{l}^s\|})^{\mu^s} \frac{\max(\langle \mathbf{n}(\mathbf{x}), \mathbf{l}^s\rangle, 0)}{\|\mathbf{l}^s\|^3}, \quad (4)$$

where Ψ^s is the light source intensity, $\mathbf{n}^s$ is the principal direction of the light source, and $\mathbf{l}^s$ is the vector pointing from the light location to the surface point. The denominator term $\|\mathbf{l}^s\|^3$ describes the attenuation of the light intensity when it reaches the surface point. $\mu^s \geq 0$ is the anisotropy parameter. Typically, when using point light source model,

other than the fact that the model is highly non-linear and non-convex, which makes it hard to optimize. Another challenge is that many parameters need to know or optimize, such as light source location and principal direction. Most works with point light source model have additional steps to calibrate the light sources intensity, location, and principal direction [20, 30]. The work of Logothetis el at. [20] designs a device with LED lights arranged circularly on a board with a centered camera and switches on and off the LED lights to capture the images in different lighting. Then they refine a pre-generated SDF model. The work of Quéau et al. [30] only takes RGB images as input and jointly optimizes the depth and reflectance. However, their output is a single depth image and needs strictly calibrated camera poses and light sources. In the next section, we will explain how our proposed model works without tedious lighting calibration, thus making our approach completely uncalibrated in the point light source and natural lighting setting.

3.2. Multi-view PS Models

To recover the complete 3D model, a sequence of RGB images $\{\mathbf{I}_i\}_i$ for $i \in \mathcal{I}$ are given with camera-to-world pose pairs $\{R_i, \mathbf{t}_i\}_i$. Denote the 3D point $\mathbf{x}$ warped to image $\mathbf{I}_i$ as $\mathbf{x}_i = R_i^\top \mathbf{x} - \mathbf{t}_i$, and plugging this into (3) and (4), we write the multi-view models residual as

$$\mathbf{r}_i = \mathbf{I}_i(\mathbf{p}(\mathbf{x}_i)) - \boldsymbol{\rho}(\mathbf{x})\mathcal{M}(\mathbf{x}, \mathcal{X}_i), \qquad (5)$$

$\mathcal{M}.(\cdot, \cdot)$ stands for two different image formation models and $\mathcal{X}_i$ is the variables in the two image formation models introduced in Section 3.1. Hence, the SH model is

$$\mathcal{M}_{\mathrm{SH}}(\mathbf{x}, \mathcal{X}_i) = \langle \mathbf{l}_i, \mathrm{SH}(R_i^\top \mathbf{n}(\mathbf{x})) \rangle. \qquad (6)$$

Point $\mathbf{x}$, as well as its normal is transferred to the image coordinates by camera pose R_i and $\mathbf{t}_i$. Here $\mathcal{X}_i = (R_i, \mathbf{t}_i, \mathbf{l}_i)$. For the point light source model, we propose a setup similar to [32], i.e. attaching an LED light to the camera when capturing the images, see Figure 3. Since the camera poses are known, the light position in (4) is at the origin of the camera coordinates, and the light source principal direction can be

Figure 3. The simple set-up we used to record the datasets, an Intel Realsense D415 camera with an LED light attached to it.

regarded as the $-z$ direction, i.e., we assume we have a collocated light-camera setup for our point light source model. The $\mathbf{l}^s$ in Equation (4) is then pointing from origin to the surface point $-\mathbf{x}_i$ under the camera coordinates. If additionally, we assume an isotropic light source, i.e., $\mu^s = 0$, the point light source model with multi-view setting reads

$$\mathcal{M}_{\mathrm{PLS}}(\mathbf{x}, \mathcal{X}_i) = \Psi_i^s \boldsymbol{\rho}(\mathbf{x}) \frac{\max(\langle R_i^\top \mathbf{n}(\mathbf{x}), -\mathbf{x}_i \rangle, 0)}{\|\mathbf{x}_i\|^3}, \quad (7)$$

where $\mathcal{X}_i = (R_i, \mathbf{t}_i, \Psi_i^s)$.

3.3. Geometry and Camera Poses Initialization

Geometric and photometric error are the two error terms that are typically utilized in camera tracking techniques. While the photometric error evaluates the color constancy when projecting one point to another RGB image, the geometric error determines the depth displacement when warping points to another depth frame. The majority of earlier surface refinement techniques [17, 11, 22, 8] use tracking methods such as [35] or [26, 9], where photometric error terms are adapted. Therefore, colors are assumed to be consistent across images when initializing the camera poses, but are assumed to be different later in the image formation model. To avoid this inconsistency, the proposed method initializes the SDF volume and estimated camera pose only using depth information, then optimize the camera poses together with surface properties taking lighting conditions into account later. The camera pose for i-th frame is optimized by aligning the point cloud with points $\mathbf{x}^k$ from depth i to the global shape $\mathcal{S}$ by [9, 38]

$$\min_{R_i, \mathbf{t}_i} E(R_i, \mathbf{t}_i) = \sum_k w_i^k d_{\mathcal{S}}(R_i \mathbf{x}^k + \mathbf{t}_i)^2, \qquad (8)$$

where $w_i^k = \max(\min(1 + \frac{d_i^k}{T}, 1), 0)$ is the truncated SDF weight of points k at frame i, T is the truncated distance and $d_{\mathcal{S}}(\mathbf{x})$ is the distance from point $\mathbf{x}$ to the shape $\mathcal{S}$. Please refer [9] or supplementary material for more details.

4. Voxel Based Photometric Modeling

4.1. From Voxel to Surface

The key idea of the proposed method is describing the explicit surface under an implicit representation, i.e., under a volumetric cube. To perform all the operations on the actual surface, we must find the corresponding surface point to each voxel. To combine the advantage of surface point representation with volumetric representation, gradient-SDF [38] stores the signed distance ψ^j for each voxel $\mathbf{v}^j$ for $j \in \mathcal{V}$, together with the distance gradient $\mathbf{g}^j$ of this voxel. It allows us to easily compute the surface point $\mathbf{x}^j$ by moving along the gradient direction $\mathbf{g}^j$ with the

voxel distance ψ^j,

$$\mathbf{x}^j = \mathbf{v}^j - \mathbf{g}^j \psi^j \,. \tag{9}$$

We then can develop our model precisely on the surface points. We will show it is not only theoretically more precise, but also leads to better quantitative results in Section 5. To retrieve the color of a surface point in the image, we can project the actual surface back to the image domain by

$$\mathbf{I}_i(\mathbf{p}(\mathbf{x}_i^j)) = \mathbf{I}_i(\pi(R_i^\top(\mathbf{v}^j - \psi^j \mathbf{g}^j - \mathbf{t}_i))) \,, \tag{10}$$

where $\pi : \mathbb{R}^3 \to \mathbb{R}^2$ is the projection operator that maps 3D points to 2D pixels on the RGB image and $\mathbf{x}_i^j$ is j-th point projected on i-th image. The stored voxel gradient $\mathbf{g}$ is the surface normal $\mathbf{n}(\mathbf{x})$ as described in [38], and $\mathbf{I}_i$ is the i-th color image. To recover the reflectance field (albedo) of the surface, for each voxel we directly save the estimated albedo of the surface point, i.e., $\boldsymbol{\rho}^j = \boldsymbol{\rho}(\mathbf{v}^j - \psi^j \mathbf{g}^j)$.

4.2. Multi-view PS Energy

We now present the multi-view volumetric-based photometric stereo model. For a set of input RGB images $\{I_i\}_i$ and a set of gradient-SDF voxels $\{\mathbf{v}^j\}_j$, the following energy function is minimized to recover the high-quality texture and jointly perform the bundle adjustment for camera pose refinement.

$$\min_{\{\boldsymbol{\rho}^j, \psi^j\}_j, \{\mathcal{X}_i\}_i} \mathbf{E}(\boldsymbol{\rho}^j, \psi^j, \mathcal{X}_i)$$
$$= \sum_{i \in \mathcal{I}, j \in \mathcal{V}} \nu_i^j \Phi(\mathbf{I}_i^j - \boldsymbol{\rho}^j \mathcal{M}(\mathbf{x}^j, \mathcal{X}_i))$$
$$+ \lambda \sum_{j \in \mathcal{V}} |\|\nabla \psi^j\|^2 - 1|^2 \,, \tag{11}$$

where $\Phi(\cdot)$ is a robust M-estimator [30]. We choose Cauchy estimator $\Psi(x) = \log(1 + \frac{x^2}{\sigma^2})$. The parameters $\sigma = 0.2$. ν_i^j is the visibility map of the j-th voxel in image i which we stored during camera tracking stage. $\mathbf{I}_i^j = \mathbf{I}_i(\mathbf{p}(\mathbf{x}_i^j))$ as described in equation (10). $\mathcal{M}(\mathbf{x}^j, \mathcal{X}_i)$ is as stated in (7) and (6) with point $\mathbf{x}^j$ is adapted from voxel $\mathbf{v}^j$ using (9). For each voxel two variables $\boldsymbol{\rho}^j$ and ψ^j are optimized. Both models have camera poses embedded; therefore, the camera poses can also be improved by minimizing the model. Joint optimization of camera poses and surface geometry, albedo, and lighting is a superior choice and leads to better results, as we will show in Section 5.

Note that we only need one regularizer in (11) on the distance gradient to ensure the distance satisfies the Eikonal equation. The regularizer needs to be made aware of two important things. First, because there is no voxel-center-to-surface gap, our energy function is straightforward and elegant. Contrary to the previous works [8, 22], our formulation (11) is not empirical but has a physical meaning: for

2D manifolds embedded in 3D being represented with an SDF, the signed distance field is differentiable and its gradient satisfies the Eikonal equation [27] on the iso-surface. The regularizer guarantees that the updated distance is still within the distance field. Second, constraining only the distance field itself detaches the energy from depth images. We do not need to store depth images during image formation model optimization of (11).

Note that the geometry is refined by optimizing the normal as well as the distance of a gradient-SDF voxel. Using the fact that close to the iso-surface, the gradient of the distance and surface normal satisfy

$$\mathbf{g}^j = \frac{\nabla \psi^j}{\|\nabla \psi^j\|} \,. \tag{12}$$

4.3. Voxel Up-sampling

A smaller voxel size is preferable for representing fine-scaled geometry details. However, for the previous SDF-based method [22], the up-sampling is possible by interpolating between neighborhood voxels. Several voxels need to be accessed for one subdivision. Our proposed method can effectively up-sample and only one voxel is needed for up-sampling $2 \times 2 \times 2 = 8$ sub-voxels to reach $(\frac{v^s}{2})$ voxel size using Taylor expansion similar as (9)

$$\mathbf{v}_{1...8}^j = \mathbf{v}^j + \frac{v^s}{4} \mathbf{s}_{1...8} \,, \tag{13}$$

$$\Rightarrow \quad \psi_{1...8}^j = \psi^j + \frac{v^s}{4} (\mathbf{s}_{1...8})^\top \mathbf{g}^j \,, \tag{14}$$

where $\mathbf{s}^{1...8} = (\pm 1, \pm 1, \pm 1)^\top$ indicate the 8 different directions from the coarse voxel. The distances of the sub-voxels are therefore re-initialized using the coarse voxel. The gradient is set to be the same of the coarse voxels because it will be updated in the next optimization steps. We include the up-sample around surface feature in our algorithm to enable higher resolution reconstructions.

4.4. Optimization Pipeline

We alternatingly update the surface quantities and camera poses during optimization. The $\{\boldsymbol{\rho}^j\}_j$ is initialized using the average intensities of voxel j. The optimization pipeline is as shown in Algorithm 1. The convergence condition is satisfied when the relative difference of k-th step energy and $(k-1)$-th step energy is smaller than the convergence threshold.

5. Evaluation

To demonstrate the results of the proposed method, due to the fact that it lacks method that offers both camera pose and surface geometry refining estimation simultaneously, we divide our evaluation into two parts. The first part is a

Algorithm 1: Optimization Pipeline

input: $\{(\psi^j, \rho^j)\}_j, \{R_i, \mathbf{t}_i\}_i, \{I_i\}_i$

while $k < max\ iter$ and not converge **do**

  if *up-sample* **then**
    $\mathcal{V} \leftarrow \mathcal{V}_{up}, \psi^j \leftarrow \psi_{up}^j$
  end

  for ($j \in \mathcal{V}$)
    $\rho^j \leftarrow \min \mathbf{E}(\psi^{j,(k)}, \rho, R_i^{(k)}, \mathbf{t}_i^{(k)}, \mathbf{l}_i^{(k)})$,

  for ($i \in \mathcal{I}$)
    $\mathbf{l}_i \leftarrow \min \mathbf{E}(\psi^{j,(k)}, \rho^{j,(k+1)}, R_i^{(k)}, \mathbf{t}_i^{(k)}, \mathbf{l}_i)$,

  for ($j \in \mathcal{V}$)
    $\psi^j \leftarrow \min \mathbf{E}(\psi, \rho^{j,(k+1)}, R_i^{(k)}, \mathbf{t}_i^{(k)}, \mathbf{l}_i^{(k+1)})$,
    $\hat{\mathbf{g}}^j \leftarrow \nabla \psi^j$,

  for ($i \in \mathcal{I}$)
    $R, \mathbf{t}_i \leftarrow \min \mathbf{E}(\{\rho, \psi\}^{j,(k+1)}, R_i, \mathbf{t}_i, \mathbf{l}_i^{(k+1)})$,

end

	SDF-Fusion [9]	BAD SLAM [36]	gradient-SDF [38]	Ours w/o pose refine	Ours (voxel)	Ours
fr1/xyz	2.3	**1.8**	2.0	2.1	2.0	2.0
fr1/plant	4.3	1.9	11.2	**1.5**	2.5	2.5
fr3/household	4.0	1.5	1.5	2.8	1.1	**0.7**
fr2/desk	-	1.8	1.6	1.1	0.9	**0.7**
fr2/rpy	2.2	0.9	7.0	1.3	0.6	**0.4**
fr2/xyz	1.8	1.3	1.3	1.0	0.9	**0.6**

Table 1. Root Mean Square Error (RMSE) of the absolute trajectory error (ATE) in cm on sequences from [39] compared against three baseline SLAM methods. From the right side, the third column represents the initial camera pose after tracking, the second column shows the error when modeling and optimizing the voxel center, and the last column shows the errors of the proposed method.

quantitative evaluation of the refined camera poses. We test our method on the TUM RGB-D [39] benchmark where the ground truth camera poses are provided. We compare the refined camera pose with three other state-of-the-art tracking methods [9, 36, 38]. The second part is the evaluation of surface refinement. We compared against four state-of-the-art methods; two classical approaches [22, 8] and two learning based approaches [40, 42] on both synthetic and real world datasets.

Set-up and Run Time Our data structure is implemented in C++ with single-precision `floats`. All experiments are performed on an Intel Xeon CPU @ 3.60 GHz without GPU. For both surface quantities optimization and pose refinement, we use the damped Gauss-Newton method [6] with $\lambda = 0.1$. The convergence threshold is 10^{-3}. We only enable the up-sampling once after 5 iterations. More mathematical and experiments details are stated in the supplementary material. We use an LED light attached to an Intel Realsense D415 RGB-D camera with a hand-held bar, see Figure 3 to record the point light source datasets. For synthetic datasets we use 256^3 voxel grid with 2cm voxel size, resulting in an initialSDF of 512MB, after integrating the depth maps. We only store voxels that contain surface points for optimization (for the synthetic bunny dataset [1], it is around 20k points). The camera tracking needs around 300ms per frame, and each optimization iteration needs around 8s per variable. The method converges typically after ∼20 iterations.

5.1. Camera Poses Refinement

We use the first-order SH model to refine the poses together with surface geometry because the datasets satisfy the natural light assumption and compare against two baseline SLAM methods [9, 36] and the method [38] which re-

fines camera poses without PS loss. We take maximum 300 frames to initialize the SDF volume when the whole sequence length is more than 300, then take 10% as keyframes using a sharpness detector [3] for the follow-up optimization. Table 1 shows that the proposed method further improves the camera pose after the tracking phase and gets better accuracy than other state-of-the-art methods. We superior to [38] by introducing the image formation model to refine camera poses. We test directly optimizing (11) on the voxel center to confirm that the step (9) leads to more accurate results. The failing case (*fr1/plant*) is due to absence of geometric information in the first few frames, causing tracking and PS optimization to fail. The results also indicate that PS method can deal with the wider baseline between two selected frames. Nevertheless, camera tracking, thus PS optimization might fail if the baseline is too large. The visualization of the refined surface compared to the initial surface also verified the improvement, see Figure 4. We obtain sharper and clearer textures compared to the state-of-the-art approaches. More visualization results are presented in the supplementary material.

5.2. Surface Geometry Improvement

We evaluate the improvement of the surface geometry on both synthetic datasets and real-world datasets for two proposed image formation models: SH model and point light source (PLS) model. The synthetic data contains 90 RGB images and corresponding depth images which are augmented with Kinect-like noise [16]. We show the quantitative comparison on synthetic datasets and qualitative evaluation on real-world datasets. See supplementary material for more details.
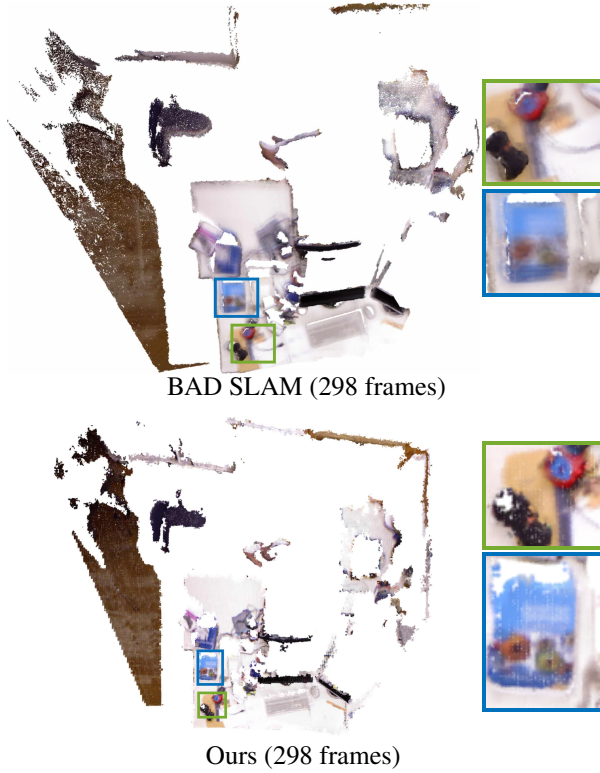
BAD SLAM (298 frames)

Ours (298 frames)

Figure 4. Point cloud reconstruction on *fr1/xyz* [39] by BAD SLAM [36] (top) and our method (bottom).The proposed method reduces the blurry effect and recovers the clear texture.

Figure 5. The figure shows the distance error of the reconstruction at different stages of the proposed method.The percentage with error less than 0.010% of the point cloud size increases from 75.45% to 82.12% after optimization for SH model and from 73.95% to 79.27% for the PLS model.

Figure 6. Quantitative evaluation on synthetic datasets. The point number percentage for the points with distance error less than 0.015% of point cloud size is 84.32% ([8]), 89.50% (volSDF [42]), 90.04% (Intrinsic3d [22]) 92.09% (NeuS [40]), 93.33% (ours)

Figure 5 shows the 3 stage errors: error of initial point cloud, i.e., after camera tracking, error after voxel up-sampling and error after the optimization of the energy (11). To eliminate the influence of object size, as small object size leads to a small absolute point cloud distance error under the same voxel grid size, we compute the distance error and object size ratio as the measurement. The x-axis shows the point-to-point distance error d_{c-c} w.r.t. the object size, i.e., $e = \frac{d_{c-c}}{d_{max}}$, where the d_{max} is the point cloud bounding box size. The y-axis is the percentage of points: points number with error less than e divided by the total points number of the point cloud.

SH model Figure 6 shows the quantitative evaluation compared to the related methods [22, 8, 42, 40] under same voxel size setting. The curve shows the proposed method performs best. Figure 7 shows the visualization of the reconstructed error on the real-world datasets [46] where the laser-scanned ground truth is available. The neural rendering methods volSDF [42], NeuS [40] perform well on the synthetic dataset but not well on the real-world datasets; even the mask setting is enabled in NeuS and the complex background setting is enabled in volSDF. For more experiments please refer to the supplementary material.
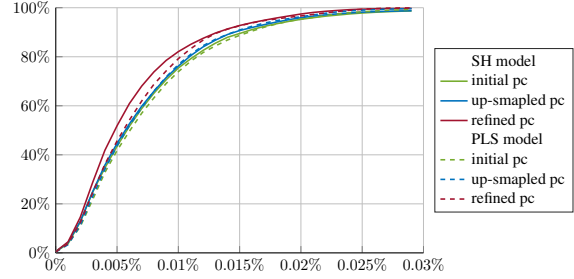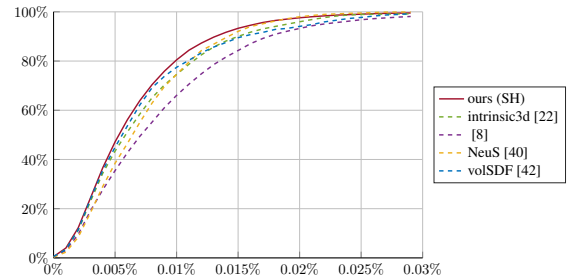
Point light source model We do not find existing work performs a non-calibrated point light source method that enables a complete 3D model. The work of Logothetis [20] requires a specific setting and calibration, but the data and code are not publicly available. So we only present the visualization results of the datasets recorded and refined using the proposed setup mentioned in Section 4. The sequences contain RGB and depth images of size 648×480, 15fps frame rates. We plug in the sequences directly in our method without pre-processing. The quantitative and qualitative results are showed in Figure 5 and Figure 8.

6. Ablation Study

Voxel center vs. surface point To verify that formulating the image formation model on the surface point is theoretically more accurate and leads to better performance, we test the voxel-center formulation on the synthetic data as well. Figure 9 shows that the surface point model results in more minor distance errors than the on-voxel-center approach. For camera pose refinement, the proposed method achieves the pose RMSE 6.3cm for the SH model and 8.1cm for the PLS model, while the on-voxel-center approach gives 7.8cm and 15.9cm, respectively.

RGB image   NeuS [40]   volSDF [42]   [8]

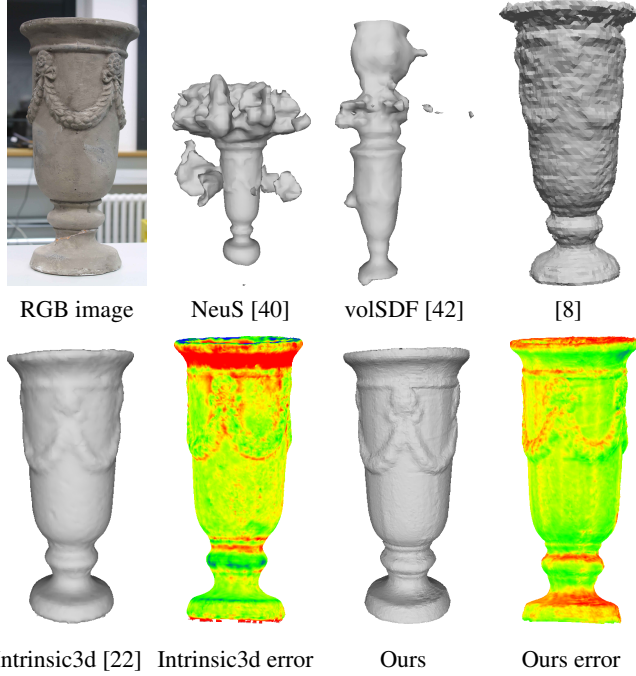Intrinsic3d [22]   Intrinsic3d error   Ours   Ours error

Figure 7. Comparison with Intrinsic3d [22] on the vase dataset [46]. In the error map, from the yellow to red color transaction indicates a larger positive distance to the ground truth, and the blue direction indicates the negative distance. The standard deviation of our result and the laser scan is 4.5mm, while [22] results in 5.8mm.
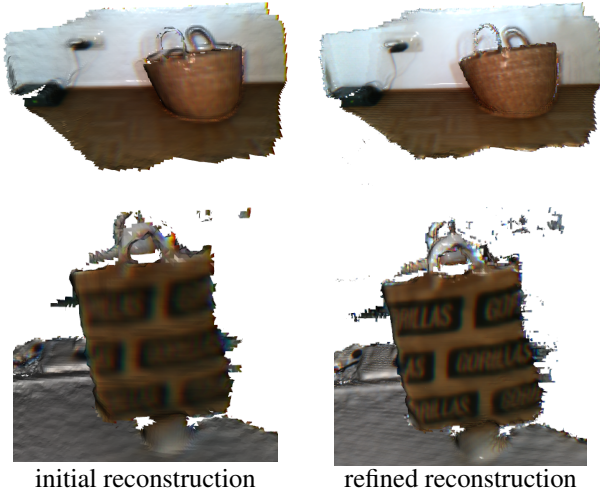

initial reconstruction   refined reconstruction

Figure 8. The example reconstruction of a sequence concentrated on a single object. The sequence contains around 200 RGB images and depth images that are recorded using the setting in Figure 3.

Eikonal constraint We include Eikonal regularizer in (11) in the multi-view PS energy. To verify the necessity of the regularizer not only from a theoretical but also from a heuristical point-of-view. We optimize our algorithm with
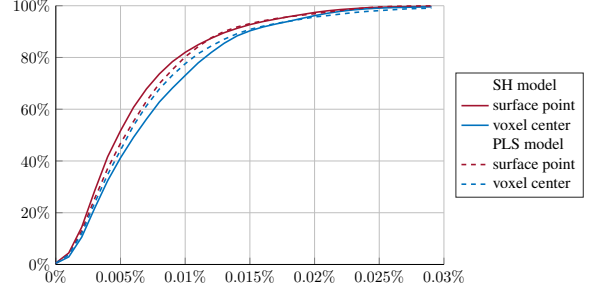

Figure 9. The figure shows the number of points with certain distance error range optimized on the voxel center (blue) and on surface points (red). Only less than 90.31% and 90.87% points have an error less than 0.015% percentage of the object size for the SH model and PLS model, while they increase to nearly 92.72% and 93.10% points when optimizing on the true surface points.
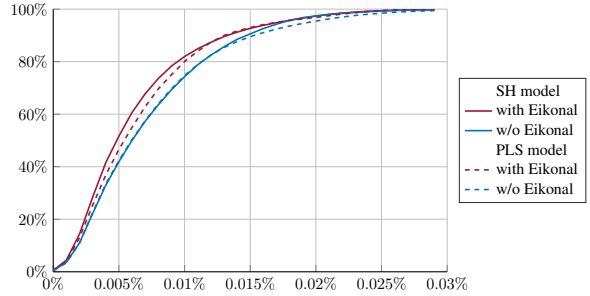

Figure 10. The number of points whose percentage distance error include the regularizer (red) and without the regularizer (blue). The points percentage of error less than 0.010% object size increase from 74.30% to 81.97% SH model and 74.77% to 80.09% for PLS model.

and without Eikonal regularizer for SH and PLS models. As Figure 10 shows, Eikonal regularizer improves the accuracy of the results in both natural light and PLS settings.

7. Conclusion and Future Work

We perform camera pose tracking and 3D surface recovery and refinement in a complete high-quality reconstruction pipeline using the gradient-SDF representation. We achieve good reconstruction quality and accurate pose tracking by enforcing the PS image formation model on the actual physical surface instead of the voxel center. Our method enables an easy and practical dense object 3D reconstruction without pre-processing or any other calibrations in natural light or point light source scenarios. We demonstrate that our method achieves superior results quantitatively and qualitatively on both synthetic and real-world datasets. Yet, the limitation of the work is, it might fail for degenerated surface due to the absence of the geometry and shading information.

In the future, we plan to include a general image formation model that can deal with non-Lambertian surfaces.

References

[1] The stanford models. http:http://graphics. stanford.edu/data/3Dscanrep/. Accessed: 2021-09-19.

[2] Dejan Azinović, Ricardo Martin-Brualla, Dan B Goldman, Matthias Nießner, and Justus Thies. Neural rgb-d surface reconstruction. *arXiv preprint arXiv:2104.04532*, 2021.

[3] Raghav Bansal, Gaurav Raj, and Tanupriya Choudhury. Blur image detection using laplacian operator and open-cv. In *2016 International Conference System Modeling and Advancement in Research Trends (SMART)*, pages 63–67, 2016.

[4] R. Basri and D. Jacobs. Photometric stereo with general, unknown lighting. In *Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition. CVPR 2001*, volume 2, pages II–II, 2001.

[5] R. Basri and D.W. Jacobs. Lambertian reflectance and linear subspaces. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 25(2):218–233, 2003.

[6] Åke Björck. *Numerical Methods for Least Squares Problems*. Society for Industrial and Applied Mathematics, 1996.

[7] M Brahimi, Y Quéau, B Haefner, and D Cremers. *On the Well-Posedness of Uncalibrated Photometric Stereo Under General Lighting*, chapter Advances in Photometric 3D-Reconstruction, pages 147–176. Springer International Publishing, Cham, 2020.

[8] Erik Bylow, Robert Maier, Fredrik Kahl, and Carl Olsson. Combining depth fusion and photometric stereo for fine-detailed 3d models. In *Scandinavian Conference on Image Analysis*, pages 261–274. Springer, 2019.

[9] Erik Bylow, Jürgen Sturm, Christian Kerl, Fredrik Kahl, and Daniel Cremers. Real-time camera tracking and 3D reconstruction using signed distance functions. In *Robotics: Science and Systems*, volume 2, page 2, 2013.

[10] Vinod G Gokhare, DN Raut, and DK Shinde. A review paper on 3d-printing aspects and various processes used in the 3d-printing. *Int. J. Eng. Res. Technol*, 6(06):953–958, 2017.

[11] Hyunho Ha, Joo Ho Lee, Andreas Meuleman, and Min H Kim. Normalfusion: Real-time acquisition of surface normals for high-resolution rgb-d scanning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 15970–15979, 2021.

[12] B. Haefner, Z. Ye, M. Gao, T. Wu, Y. Quéau, and D. Cremers. Variational uncalibrated photometric stereo under general lighting. In *International Conference on Computer Vision (ICCV)*, Seoul, South Korea, October 2019.

[13] Lingzhi He, Hongguang Zhu, Feng Li, Huihui Bai, Runmin Cong, Chunjie Zhang, Chunyu Lin, Meiqin Liu, and Yao Zhao. Towards fast and accurate real-world depth super-resolution: Benchmark dataset and baseline. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 9229–9238, June 2021.

[14] Jonathan J. Hull, Berna Erol, Jamey Graham, Qifa Ke, Hidenobu Kishi, Jorge Moraleda, and Daniel G. Van Olst. Paper-based augmented reality. In *17th International Conference on Artificial Reality and Telexistence (ICAT 2007)*, pages 205–209, 2007.

[15] James T. Kajiya. The rendering equation. In *Proceedings of the 13th Annual Conference on Computer Graphics and Interactive Techniques*, SIGGRAPH '86, page 143–150, New York, NY, USA, 1986. Association for Computing Machinery.

[16] Kourosh Khoshelham and Sander Oude Elberink. Accuracy and resolution of Kinect depth data for indoor mapping applications. *Sensors*, 12(2):1437–1454, 2012.

[17] Joo Ho Lee, Hyunho Ha, Yue Dong, Xin Tong, and Min H Kim. Texturefusion: High-quality texture acquisition for real-time rgb-d scanning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1272–1280, 2020.

[18] Chen-Hsuan Lin, Wei-Chiu Ma, Antonio Torralba, and Simon Lucey. Barf: Bundle-adjusting neural radiance fields. In *IEEE International Conference on Computer Vision (ICCV)*, 2021.

[19] Fotios Logothetis, Roberto Mecca, and Roberto Cipolla. A differential volumetric approach to multi-view photometric stereo, 2018.

[20] Fotios Logothetis, Roberto Mecca, and Roberto Cipolla. A differential volumetric approach to multi-view photometric stereo. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 1052–1061, 2019.

[21] William E Lorensen and Harvey E Cline. Marching cubes: A high resolution 3d surface construction algorithm. *ACM siggraph computer graphics*, 21(4):163–169, 1987.

[22] 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 *International Conference on Computer Vision (ICCV)*, Venice, Italy, October 2017.

[23] Roberto Mecca, Aaron Wetzler, Alfred M. Bruckstein, and Ron Kimmel. Near field photometric stereo with point light sources. *SIAM Journal on Imaging Sciences*, 7.

[24] Ben Mildenhall, Pratul P. Srinivasan, Matthew Tancik, Jonathan T. Barron, Ravi Ramamoorthi, and Ren Ng. Nerf: Representing scenes as neural radiance fields for view synthesis. In *ECCV*, 2020.

[25] Richard A Newcombe, Shahram Izadi, Otmar Hilliges, David Molyneaux, David Kim, Andrew J Davison, Pushmeet Kohli, Jamie Shotton, Steve Hodges, and Andrew Fitzgibbon. KinectFusion: Real-time dense surface mapping and tracking. In *10th IEEE International Symposium on Mixed and Augmented Reality*, pages 127–136. IEEE, 2011.

[26] Matthias Nießner, Michael Zollhöfer, Shahram Izadi, and Marc Stamminger. Real-time 3D reconstruction at scale using voxel hashing. *ACM Transactions on Graphics (ToG)*, 32(6):1–11, 2013.

[27] Stanley J. Osher and Ronald Fedkiw. *Level set methods and dynamic implicit surfaces.*, volume 153 of *Applied mathematical sciences*. Springer, 2003.

[28] S. Peng, B. Haefner, Y. Quéau, and D. Cremers. Depth super-resolution meets uncalibrated photometric stereo. In *International Conference on Computer Vision Workshops (ICCVW)*, 2017.

[29] Songyou Peng, Michael Niemeyer, Lars Mescheder, Marc Pollefeys, and Andreas Geiger. Convolutional occupancy networks, 2020.

[30] Yvain Quéau, Bastien Durix, Tao Wu, Daniel Cremers, François Lauze, and Jean-Denis Durou. Led-based photometric stereo: Modeling, calibration and numerical solution, 2017.

[31] Jorge Reyna, Jose Hanham, and Peter Meier. The internet explosion, digital media principles and implications to communicate effectively in the digital space. *E-Learning and Digital Media*, 15(1):36–52, 2018.

[32] L. Sang, B. Haefner, and D. Cremers. Inferring super-resolution depth from a moving light-source enhanced RGB-D sensor: A variational approach. In *IEEE Winter Conference on Applications of Computer Vision (WACV)*, Colorado, USA, March 2020.

[33] Sara Fridovich-Keil and Alex Yu, Matthew Tancik, Qinhong Chen, Benjamin Recht, and Angjoo Kanazawa. Plenoxels: Radiance fields without neural networks. In *CVPR*, 2022.

[34] Johannes Lutz Schönberger and Jan-Michael Frahm. Structure-from-Motion Revisited. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

[35] Johannes Lutz Schönberger, Enliang Zheng, Marc Pollefeys, and Jan-Michael Frahm. Pixelwise View Selection for Unstructured Multi-View Stereo. In *European Conference on Computer Vision (ECCV)*, 2016.

[36] Thomas Schops, Torsten Sattler, and Marc Pollefeys. Bad slam: Bundle adjusted direct rgb-d slam. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 134–144, 2019.

[37] Amir Arsalan Soltani, Haibin Huang, Jiajun Wu, Tejas D Kulkarni, and Joshua B Tenenbaum. Synthesizing 3d shapes via modeling multi-view depth maps and silhouettes with deep generative networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1511–1519, 2017.

[38] C Sommer, L Sang, D Schubert, and D Cremers. Gradient-SDF: A semi-implicit surface representation for 3d reconstruction. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022.

[39] Jürgen Sturm, Nikolas Engelhard, Felix Endres, Wolfram Burgard, and Daniel Cremers. A benchmark for the evaluation of rgb-d slam systems. In *2012 IEEE/RSJ international conference on intelligent robots and systems*, pages 573–580. IEEE, 2012.

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

[41] Robert J. Woodham. Reflectance map techniques for analyzing surface defects in metal castings. 1977.

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

[43] Alex Yu, Ruilong Li, Matthew Tancik, Hao Li, Ren Ng, and Angjoo Kanazawa. PlenOctrees for real-time rendering of neural radiance fields. In *ICCV*, 2021.

[44] Kai Zhang, Fujun Luan, Qianqian Wang, Kavita Bala, and Noah Snavely. Physg: Inverse rendering with spherical gaussians for physics-based material editing and relighting. In *The IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021.

[45] Qian-Yi Zhou and Vladlen Koltun. Color map optimization for 3d reconstruction with consumer depth cameras. *ACM Trans. Graph.*, 33(4), jul 2014.

[46] Michael Zollhöfer, Angela Dai, Matthias Innmann, Chenglei Wu, Marc Stamminger, Christian Theobalt, and Matthias Nießner. Shading-based refinement on volumetric signed distance functions. *ACM Transactions on Graphics (TOG)*, 34(4):1–14, 2015.