WildLight: In-the-wild Inverse Rendering with a Flashlight

Ziang Cheng, Junxuan Li, Hongdong Li
Australian National University
{ziang.cheng,junxuan.li,hongdong.li}@anu.edu.au

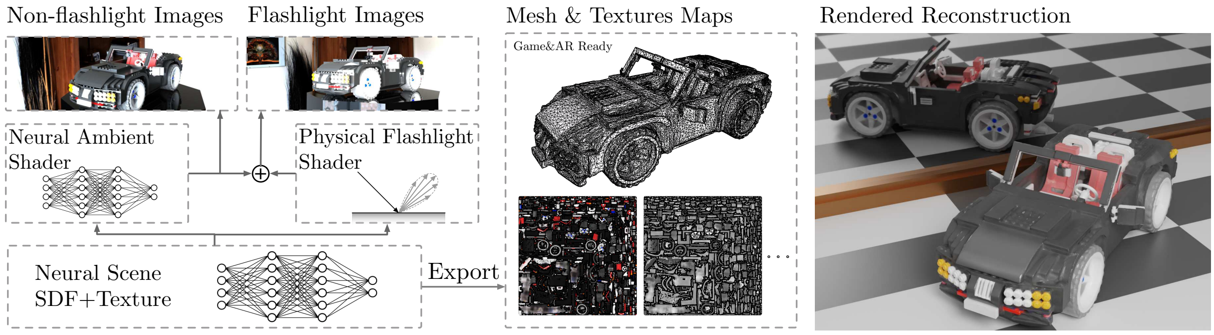


Figure 1. Our method reconstructs object geometry and reflectance from unstructured flashlight and non-flashlight images captured under unknown environment lights. We use a co-located camera-light configuration that is available with most smartphones. Reconstructed objects can be easily converted to industry-ready triangle meshes and PBR textures. The image to the right is rendered from our mesh export.

Abstract

This paper proposes a practical photometric solution for the challenging problem of in-the-wild inverse rendering under unknown ambient lighting. Our system recovers scene geometry and reflectance using only multi-view images captured by a smartphone. The key idea is to exploit smartphone's built-in flashlight as a minimally controlled light source, and decompose image intensities into two photometric components – a static appearance corresponds to ambient flux, plus a dynamic reflection induced by the moving flashlight. Our method does not require flash/non-flash images to be captured in pairs. Building on the success of neural light fields, we use an off-the-shelf method to capture the ambient reflections, while the flashlight component enables physically accurate photometric constraints to decouple reflectance and illumination. Compared to existing inverse rendering methods, our setup is applicable to non-darkroom environments yet sidesteps the inherent difficulties of explicit solving ambient reflections. We demonstrate by extensive experiments that our method is easy to implement, casual to set up, and consistently outperforms existing in-the-wild inverse rendering techniques. Finally, our neural reconstruction can be easily exported to PBR textured triangle mesh ready for industrial renderers. Our source code and data are released to https://github.com/za-cheng/WildLight.

1. Introduction

Rendering in computer graphics refers to computer generating photo-realistic images from known properties of a scene including scene geometry, materials, lighting, as well as camera parameters. In contrast, inverse rendering is regarded as a computer vision task, whose aim is to recover these unknown properties of a scene from images. Due to the ill-posed nature of inverse rendering, most existing methods concede the fullest solution and instead tackle only a simplified, partial problem; for instance, assuming simplified, known lighting conditions, initial geometry, or with diffuse or low specular materials.

Traditional photometric methods for solving inverse rendering (such as photometric stereo) are often restricted to laboratory settings: image intensities are measured by high dynamic range cameras, under controlled illumination conditions and without ambient light contamination. Moreover, many methods rely on the availability of a good initial estimation to start the optimization process (*e.g*. [19,26]) or assume purely diffuse (Lambertian) reflections (*e.g*. [21,31]).

While recent neural-net methods are able to handle specular highlights and complex geometries, *e.g*. [2, 10, 20, 39]), they are still restricted to a laboratory darkroom environment, hindering their practicality.

Conversely, neural rendering and appearance learning techniques have the ability to work outside darkroom. This is achieved by bypassing the physical reflection model. They instead learn the illumination-specific appearance (*i.e*. light field) of the scene conditioned on a geometric representation (*e.g*. [24,27,38]). While these representations are empirically powerful in recovering complex scene geometry and appearance, they cannot separate reflective properties from illumination. Furthermore, the underlying geometry is provably ill-constrained. Attempts have been made to physically decompose the appearance into reflectance and environment illumination to support in-the-wild inverse rendering [5, 25, 33, 40, 43]. However, this presents a much more complex and similarly ill-posed problem due to unknown ambient illumination. Consequently, these methods still trail behind traditional photometric methods in terms of accuracy and robustness.

This paper aims to fill the gap between conventional darkroom methods and in-the-wild inverse rending, and offer a solution that combines the best of both worlds, *i.e*. being practical, well-posed and easy-to-solve at the same time. Instead of attempting to directly decouple reflectance from the unknown ambient illumination, we learn the ambient reflection with a neural light field, and exploit an additional, minimally controlled light source, being the smartphone's flashlight, for physical constraints on reflectance. During the image capture stage, we take some images with the flashlight turned on, and others with the flashlight off, all at free viewpoints. Images without the flashlight describe the ambient reflections only, while the images with flashlight are the photometric summation of both ambient and flashlight reflections. We learn the ambient component with an off-the-shelf neural novel-view-synthesis technique [36], and delegate the flashlight component to a physically-based reflection model for inferring reflectance. Both ambient and flashlight components are conditioned on a unified scene intrinsic network that predicts scene geometry and reflectance distribution. Our method is easy to set up, easy to implement, and consistently outperforms competing state-of-the-arts. The reconstructed objects can be directly plugged into game/rendering engines as high fidelity virtual assets in the standard format of textured meshes.

2. Related work

Inverse rendering is a long-standing and highly researched topic. To limit the scope of discussion, below we only give a partial review of RGB camera-based multi-view 3D reconstruction methods.

Darkroom methods. Traditional photometric methods assume a darkroom environment where the lighting condition is controlled and calibrated. Earlier methods assume a mostly diffuse appearance to simplify the reflection model [11, 14, 21, 29, 31, 34, 37]. Recent papers are able to deal with moderately glossy objects by incorporating explicit specular reflectance, typically controlled by a roughness parameter. However, the underlying problem becomes highly non-convex under this setting, and many methods require an initial geometry to bootstrap the optimization process. Initial geometry is acquired either from multi-view stereo/structure-from-motion [19, 26, 44], or from RGB-D sensors [8, 13, 23, 32]. Notably, many recent methods are based on a co-located camera/light scanner to simplify imaging setup and reflection model. Nam *et al*. [26] propose a optimization pipeline that iteratively refines initial geometry and reflectance under co-located reflection. [2–4] use deep neural networks to learn representations of geometry and reflectance, supervised by ground truth, or directly by multi-view images. Cheng *et al*. [9] proposed an initialization-free optimization framework for highly specular surfaces, and recently developed a neural spherical parameterization method for learning shape and reflectance [10]. Luan *et al*. [22] used an Monte Carlo edge sampling approach for joint optimization of shape and reflectance. Recently, Zhang *et al*. [39] recovers objects as neural SDF and materials, and use an edge-aware renderer for refinement. All above methods are however restricted to a darkroom environment, and require per-view object masks to bootstrap geometry optimization.

Neural appearance/light field. Under ambient illumination, scene radiance distribution features strong co-relation with geometry. Neural implicit methods exploit this prior to model appearance as a light field conditioned on geometry. DVR [27] proposed a differentiable renderer for implicit occupancy networks. NeRF [24] use an MLP to represent density and view-dependent radiance fields supervised by input images. However, the underlying geometry is provably under-constrained and often of low quality. IDR [38] alleviates this problem by employing an signed distance field (SDF) representation that defines hard geometrical surfaces, and use a neural network to learn reflected lights. NeuS [36] proposes an alternative rendering technique for SDF based on volumetric rendering. In a similar spirit, UniSurf [28] unifies surface and volumetric rendering for an occupancy network representation. Kaya *et al*. [15] extract surface normal from photometric stereo, on which a NeRF-based neural shader is trained. While empirically these methods achieved high quality results for geometry and appearance reconstructions, they do not follow a physically based reflection model and cannot separate reflectance from illumination conditions to support relighting.

Inverse rendering under ambient light. Unlike the darkroom setup, in-the-wild inverse rendering is an ill-posed and much more challenging problem. Part of the challenge arises from the fact that reflected lights need to be integrated over all incident directions, which is a highly expensive operation. As a result, the ambient lighting is approximated by low resolution or low frequency environment maps, and all light sources are assumed to be infinitely far away. Early work of Zhang *et al.* [42] is restricted to Lambertian objects and small camera motions. Oxholm *et al.* [30] proposed an energy based approach to solve for simple geometries initialized from visual hull. NeRFactor [43] first extracts geometry from pretrained NeRF then decomposes radiance into visibility, reflectance and illumination for joint optimization. NeRV [33] learns a neural visibility field for efficiently computing secondary reflections. PhySG [40] sidesteps the numerical integration using Spherical Gaussian BRDF and illumination maps. NerD [5] adds support for varying illuminations, and NeuralPIL [6] uses an MLP to learn the light integration. Recently, Munkberg *et al.* [25] combine neural representation with mesh-based rasterizer for differentiable rendering. While these methods are flexible in imaging setup, their accuracy trails behind conventional darkroom based approaches due to the intrinsic difficulty of factorizing ambient reflections. Furthermore, NeRF-based methods [5, 6, 33, 43] cannot uphold a hard, smooth surface boundary.

Our approach. We adopt a hybrid approach that combines traditional darkroom photometric methods and neural light fields. Compared to existing in-the-wild inverse rendering methods, our approach requires an additional light source, but is not restricted to the distant ambient lighting assumption, can work without object masks, and avoids the difficulties of factorizing ambient reflections. On the other hand, the flashlight still offers sufficient photometric constraint to solve for reflectance and regulate geometry, achieving comparable accuracy to darkroom-based methods while being arguably more flexible.

3. Photometric image formulation

Our method approximates the scene as surfaces of opaque appearance. The scene is under a static but unknown ambient illumination (*i.e.*, the environment map), and may be further illuminated by a moving point light source that is co-located with the camera. With such assumptions in mind, scene radiance captured by the camera is the summation of two photometric components: an ambient component corresponding to flux transmitted from all light emitters within the scene, and an optional flashlight component that accounts for flashlight energy reflected off the surface.

More formally, with a smartphone device, we assume its flashlight is a point light source co-located with the camera. The raw image intensity of surface point $\mathbf{x} \in \mathbb{R}^3$ at view direction $\mathbf{v}$ and viewing distance t is expressed as the summation of an ambient term $\mathcal{A}$ and a flashlight term $\mathcal{L}$:

$$\mathcal{I}(\mathbf{x}, \mathbf{v}, t, s) = \mathcal{A}(\mathbf{x}, \mathbf{v}) + s\gamma\mathcal{L}(\mathbf{x}, \mathbf{v}, t), \tag{1}$$

where $s \in \{0, 1\}$ is a binary switch indicating whether the flashlight is on, and γ is the unknown flashlight intensity. Ambient term defines a scene appearance independent of the flashlight constituting non-flash images. Neural appearance/novel-view synthesis methods learn $\mathcal{A}$ as a black box function to supervise an underlying scene geometry that $\mathcal{A}$ is conditioned on (*i.e.* spatial distribution of $\mathbf{x}$) [24, 38]. This problem however, is well known to be under-constrained due to the view dependency of $\mathcal{A}$, in which case the geometry cannot be well recovered [41]. An alternative approach is to explain $\mathcal{A}$ with an ambient reflection model to obtain physical constrains, but this involves a much more complicated problem that is also often ill-posed [30, 33].

To mitigate above issues, we incorporate an additional flashlight reflection term. It follows real world physics but is much simpler and better posed than ambient reflections

$$\mathcal{L}(\mathbf{x}, \mathbf{v}, t) = \frac{\mathcal{E}_\mathbf{v}}{t^2}\rho_\mathbf{x}(\mathbf{n}_\mathbf{x}, \mathbf{v}, \mathbf{v}) \max(\mathbf{n}_\mathbf{x}^\top \mathbf{v}, 0). \tag{2}$$

$\mathcal{E}_\mathbf{v}$ denotes the flashlight radiant intensity at direction $\mathbf{v}$, and $\rho_\mathbf{x}$ is the material's reflectance function of three directions: surface normal vector $\mathbf{n}_\mathbf{x}$, and view and light directions joint at $\mathbf{v}$.[1] . The reflectance function can be further parameterized as

$$\rho_\mathbf{x}(\mathbf{n}, \mathbf{v}, \mathbf{l}) = \rho(\mathbf{n}, \mathbf{v}, \mathbf{l}; \Theta_\mathbf{x}) \tag{3}$$

such that it is defined by a set of parameters $\Theta_\mathbf{x}$. Without loss of generality, we further assume the flashlight exhibits an isotropic distribution of intensity within the field of view.[2] That is, we may simply define $\mathcal{E}_\mathbf{v} = 1$ as one unit of radiant intensity.

The co-located reflection model in (2) provides strong photometric constraints on the scene geometry and reflection, allowing them to be recovered jointly if flashlight reflection $\mathcal{L}$ is made known [9, 22, 26]. In reality, however, the flashlight images contain both flashlight reflections $\mathcal{L}$ and ambient reflections/emissions $\mathcal{A}$, and it is non-trivial to separate one from another. Many photometric methods are therefore limited to darkroom environment where there is no ambient light contamination. In this paper, we sidestep this problem by employing a neural appearance model to learn the ambient component.

[1] Actual difference between and view and light angles is around 1 degree for an iPhone at $0.5m$ view distance.

[2] Any radial anisotopicity can be corrected by applying per-pixel anti-vignetting compensation factors, as long as light and camera are co-located.

4. Scene representation and methodology

We parameterize target object's geometry and reflectance within an *intrinsic network*, and use an *ambient network* to shade its ambient reflections $\mathcal{A}$ as a neural light field. Our method supports both masked and mask-less object reconstruction. When object masks are not available, we only reconstruct the partial scene within a given spherical Region of Interest (ROI) recognized as foreground region. The flashlight and ambient lights outside this ROI are instead approximated by a *background NeRF* inspired by Zhang *et al.* [41]. With such foreground-background separation in mind, we rewrite (1) as

$$I_{\mathbf{x}} = \begin{cases} \hat{\mathcal{A}} + s\gamma\hat{\mathcal{L}} & \text{if } \mathbf{x} \in \text{Foreground} \\ \hat{A}_{\text{NeRF}} + s\gamma\hat{L}_{\text{NeRF}} & \text{if } \mathbf{x} \in \text{Background} \end{cases} \tag{4}$$

4.1. Foreground model

The foreground geometry and reflectance are represented by the intrinsic network, on which ambient and flashlight reflections are both conditioned.

Intrinsic network The *intrinsic network* recovers object intrinsic properties (*i.e.* geometry and reflectance) independent of any illumination condition. The intrinsic network learns geometry as a signed distance field, and outputs reflectance parameters at given location.

$$\hat{\mathcal{N}} : \{\mathbf{x}\} \to \{(distance, \Theta, \mathbf{f})\}. \tag{5}$$

Apart from the signed distance and reflectance parameters Θ, the network also outputs a feature vector $\mathbf{f}_{\mathbf{x}}$ as a general descriptor for the local scene around $\mathbf{x}$. Furthermore, the network derivative defines the normal direction of a surface point:

$$\mathbf{n}_{\mathbf{x}} = \nabla_{\mathbf{x}} distance. \tag{6}$$

Ambient appearance network The ambient network acts as a neural shader that learns ambient reflections as a view-dependent neural light field defined on scene geometry.

$$\hat{\mathcal{A}} : \{(\mathbf{x}, \mathbf{v}, \mathbf{n}_{\mathbf{x}}, \Theta_{\mathbf{x}}, \mathbf{f}_{\mathbf{x}})\} \to \{RGB\}. \tag{7}$$

Compared to (1), we incorporate the normal direction and reflectance parameters as additional inputs since ambient reflection is empirically co-related with them. This formulation is inherited from previous work in neural rendering [28, 36, 38], except here the network also receives reflectance parameters $\Theta_{\mathbf{x}}$ as additional input.

Flashlight reflection While the ambient network regularizes scene geometry by the MLP's inherent prior, there exist many solutions of scene geometry and radiance field that lead to the same multi-view images. We overcome this ambiguity by lighting a subset of images with a flashlight. The flashlight reflection $\mathcal{L}$ provides important physical constraints that disambiguate geometry, and further disentangle reflectance from illumination.

$$\hat{\mathcal{L}}(\mathbf{n}_{\mathbf{x}}, \mathbf{v}, t, \Theta_{\mathbf{x}}) = \frac{1}{t^2}\rho(\mathbf{n}_{\mathbf{x}}, \mathbf{v}, \mathbf{v}; \Theta_{\mathbf{x}}) \max(\mathbf{n}_{\mathbf{x}}^{\top}\mathbf{v}, 0) \tag{8}$$

We use a physically based reflectance model to estimate the flashlight reflection. Specifically, we parameterize $\rho(\cdot)$ using Disney's principled BRDF model (also known as the PBR texture model) [7], that is, a linear mix of two diffuse lobes and three specular lobes[3]:

- Unlike the simpler Lambertian model, the diffuse lobe accounts for varying retro-reflections at grazing angles. The diffuse model also blends in a secondary subsurface lobe for modeling scattering effects observed in translucent materials (*e.g.* human skin).
- The main specular model follows the GGX microsurface distribution [35] and has two lobes: a tinted lobe for metals, and an achromatic lobe for dielectric materials.
- Another achromatic specular lobe is included for clearcoat materials. This lobe follows Berry's microsurface distribution, and has a roughness independent of other lobes.

Since the flashlight and camera are co-located, we slightly modify [7] to use a joint mask-shadowing term for all specular lobes and include the flashlight intensity γ as a trainable parameter. The reader is referred to the Appendix for the exact formulation of BRDF $\rho(\cdot; \Theta_{\mathbf{x}})$.

4.2. Background NeRF

The scene outside the foreground ROI is instead learned by a modified NeRF model [24, 41]. There are two NeRFs responsible for ambient and flashlight components respectively:

$$\hat{I}_{\text{NeRF}}(\mathbf{x}, \mathbf{v}, t, s) = \hat{A}_{\text{NeRF}}(\mathbf{x}, \mathbf{v}) + s\gamma\hat{L}_{\text{NeRF}}(\mathbf{x}, \mathbf{v}), t) \tag{9}$$

$$\text{where } \hat{L}_{\text{NeRF}}(\mathbf{x}, \mathbf{v}), t) = \frac{1}{t^2}\rho_{\text{NeRF}}(\mathbf{x}, \mathbf{v})). \tag{10}$$

The network ρ_{NeRF} is analogue to the reflectance function in (1), except here the normal direction is not explicitly given but conditioned on the input $\mathbf{x}$. While $\hat{A}_{\text{NeRF}}$ and ρ_{NeRF} have different physical significance (radiance versus reflectance), implementation-wise we merge them into a single network with two RGB output branches: one for ambient radiance and one for reflectance, under the unified name NeRF ('R' stands for radiance or reflectance). Unlike with the foreground model, the background NeRF does not

[3] We disable the transmission/refraction lobe in the original Disney's model since we only consider opaque objects.

necessarily define a geometrical surface, nor does it conform to a physically-based reflectance function. Instead, it is solely purposed to reproduce the background appearance from which the foreground can be easily separated.

4.3. Training by rendering

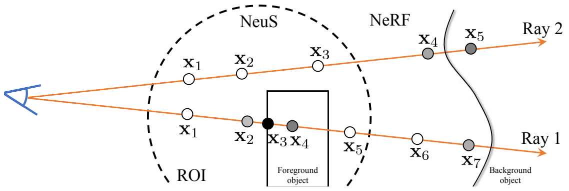


Figure 2. The hybrid rendering approach based on volumetric rendering: along each camera ray we densely trace multiple points $\mathbf{x}_1, ..., \mathbf{x}_n$ in space, and render the RGB values as the weighted mean of per-point RGBs. Alpha distributions on the rays are colored in different shades (darker is greater). RGB values are aggregated using alpha composition, where RGB and alpha values are computed from NeuS [36] for points inside the pre-defined ROI, or from the background NeRF [41] for points outside.

Volumetric renderer We adopt an off-the-shelf SDF rendering technique called NeuS [36]. Compare to conventional SDF rendering algorithms that trace a single intersection point per viewing ray, NeuS samples multiple points per ray while only the points closer and more orthogonal to the surface are assigned greater weights. This enables gradients to be traced back not only on rays hitting the surface, but also on those tangent to self-occlusion boundaries (*e.g.* for thin, concave geometries).

Denote the camera center by $\mathbf{o}$, a viewing ray is parameterized as $\mathbf{x}(t) = \mathbf{o} - \mathbf{v}t$. We render its RGB intensity by sampling multiple points at $t_i \in [0, \infty)$ along the ray, and use the alpha-composition

$$\hat{\mathcal{I}}(\mathbf{o}, \mathbf{v}, s) = \sum_i \omega_i \, \hat{\mathcal{I}}\big(\mathbf{x}(t_i), \mathbf{v}, t_i, s\big) \tag{11}$$

$$\text{where } \omega_i = \alpha_i \prod_{t_j < t_i} (1 - \alpha_j). \tag{12}$$

When foreground masks are available, we turn off the background NeRF and render foreground rays only. Conversely, in the mask-less case, we use a hybrid rendering approach where α values and RGB intensities are obtained from NeuS [36] for points inside the foreground ROI, and from the background NeRF (9) for points outside. Figure 2 illustrates the α distribution along viewing rays in the mask-less scenario.

Training loss All networks are directly supervised from multi-view images, and the intrinsic network is regularized by an Eikonal loss and an optional foreground mask loss on the SDF. The overall training loss is

$$Loss = Loss_{\text{RGB}} + w_E Loss_{\text{Eikonal}} + \underbrace{w_M Loss_{\text{mask}}}_{\text{optional}} \tag{13}$$

where $Loss_{\text{RGB}}$ is a color saturation aware L1 loss for all pixels p.

$$Loss_{\text{RGB}} = \frac{1}{|P|} \sum_{p \in P} M_p |\mathcal{I}_p - \hat{\mathcal{I}}_p| \tag{14}$$

The binary variable M_p is set to 1 if and only if $\mathcal{I}_p$ or $\hat{\mathcal{I}}_p$ does not exceed the RGB color range of $[0, 1)$. Otherwise $M_p = 0$, so that the loss is turned off where both predicted and actual pixel values are saturated (*e.g.* on specular highlights).

The Eikonal loss modulates the gradients of SDF. To compute this loss, we uniformly sample points inside the ROI where the SDF is defined.

$$Loss_{\text{Eikonal}} = \mathop{\mathbb{E}}_{\mathbf{x} \sim U} (1 - \|\nabla_{\mathbf{x}} distance\|)^2 \tag{15}$$

Finally, when available, foreground masks can be used to supervise SDF to regulate geometry. The optional foreground mask loss is defined as:

$$Loss_{\text{mask}} = \frac{1}{|P|} BCE(mask_p, \sum_i \omega_i^p), \tag{16}$$

where BCE is the binary cross entropy, and ω_i^p is defined as in (12).

Mesh and texture extraction After training, scene geometry and surface reflectance can be extracted from the intrinsic network $\hat{\mathcal{N}}$ and stored into a standard mesh format supported by industrial renderers (*e.g.* Blender, Unreal Engine *etc.*). We follow the procedure described below for mesh and PBR texture extraction:

1. A mesh is extracted from the zero isosurface of intrinsic SDF using marching cubes algorithm [18], followed by a mesh simplification step [12].
2. Mesh vertices are assigned UV coordinates. A 3D atlas is interpolated from vertex UVs, that records for each pixel in UV space, its corresponding 3D location on the mesh surface.
3. We feed the 3D atlas to the intrinsic network again to generate texture and normal maps.

We note that many other methods lack the portability to external industrial renderers. Examples include NeRF-based methods [5, 6, 33, 43] based on the particle cloud model that is not commonly supported, and [40] that uses a custom Spherical Gaussian reflectance model.

5. Experiments

5.1. Training routine and hyper-parameters

We use fixed weight $\omega_E = 0.1$, and set non-zero $\omega_M = 0.1$ only if masks are available. We train our networks by randomly sampling 256 rays per batch, where 128 points per ray are sampled within the ROI sphere. If the background NeRF is enabled, an extra 32 points per ray is sampled behind the ROI for training NeRF. Network weights are optimized with Adam [17] at a learning rate of $5e-4$, which is gradually reduced to $2.5e-5$ after 5000 iterations. The networks are trained for a total 400,000 iterations (or 2 epochs) on synthetic scenes, and double that iterations on real scenes. Total training time varies between 12 to 24 hours on a single RTX 3090. The intrinsic network weights are initialized such that initial SDF roughly corresponds to the ROI sphere [1].

For each input image, we assume the camera parameters are known, and define the ROI sphere in world space such that the objects of interest are well contained in it.

5.2. Dataset and image acquisition

We evaluate our methods on both synthetic and real-world images in indoor settings. We will publish our datasets for reproducible research.

Synthetic Image. Our synthetic images are generated with Blender, using a co-located camera and flashlight configuration. We use an imaging setup consistent with most real world smartphones, where the virtual flashlight emits a cone of light that roughly covers the camera's field of view. The flashlight is set to operate at 0.2 to 0.8 watt, and objects are located at viewing distance of 0.3 to 0.5 meter. For the virtual environment setup, we put the objects on a virtual pedestal in indoor environments. For each object we generate 75 images under ambient lighting alone, and another 75 images with flashlight turned on. Viewpoints are randomly sampled from a half dome above the object with camera always facing object center. The synthetic dataset consists of a total of 150 HDR images at 1K resolution, the corresponding foreground masks, as well as ground truth camera parameters. Example images are shown in Figure 3. Half of total images are lit by flashlight, and the other half by environment light only.

Real Image. For real world image acquisition, we used an iPhone and the "ProCam" app to take images in RAW format with a linear camera response. During each capture, we maintained a fixed camera exposure time, focus, and white balance for all images. The camera response ratio was automatically adjusted per image and recorded in file metadata, which we later used to scale all images back to a unified intensity scale. The camera pose are acquired from an AR board behind the object. Images are taken inside an apartment unit under interior lighting. We place the camera at 0.2 to 0.5 meter viewing distance, and move the camera in a spiral pattern around the object. We take 56 to 112 HDR images per object. In lack of object segmentation masks, we enable the background NeRF model and turn off $Loss_{\text{mask}}$. Around half of total images are lit by flashlight, and the other half by environment light only.

5.3. Evaluation and comparison

To the best of our knowledge, there is no previous method specifically engineered for our imaging configuration (*i.e.* in-the-wild images partly lit by flashlight). Therefore direct comparison with state-of-arts would not be fair or feasible. Instead, we keep all but a few imaging conditions identical, but allow following exceptions in favor of each competing method's own configuration:
NeRD [5] assumes varying global illumination. Therefore we replace the flashlight with a moving distant light source whose direction is always aligned with camera's optical axis. We adjust the light power to a similar intensity to our flashlight.
PhySG [40] assumes static global illumination. Therefore we remove the flashlight altogether and render images solely under ambient illumination.

Due to the diversity of imaging conditions, maintaining control variables (*e.g.* camera pose and environment lighting) is difficult. We therefore compare with above methods on the synthetic data. All methods received HDR images, object segmentation masks, as well as camera poses. Specifically, we evaluate the commutative surface-to-surface distance between predicted and ground truth shapes on the visible regions.[4] Additionally, we compute median and mean normal errors for all training view normal maps and depth maps. Results are listed in Table 1.

Each method is different in their own BRDF and illumination parameterization, therefore we compare the reflectance accuracy by rendering 20 novel-view and novel-lighting images under both flashlight and environment illumination. The results are listed in Table 2. Due to difference in setup, we swap the flashlight for a distant light source when evaluating [5] and [40]. A visual comparison is given in Figure 4. Spatially varying BRDFs are better visualized with varying lighting conditions, and we refer the reader to our website for more visualizations.

5.4. Varying flashlight intensities

The flashlight luminosity directly impacts how well the corresponding reflection model $\hat{\mathcal{L}}$ is supervised. When the

[4] See Appendix for the definition of surface-to-surface distance.

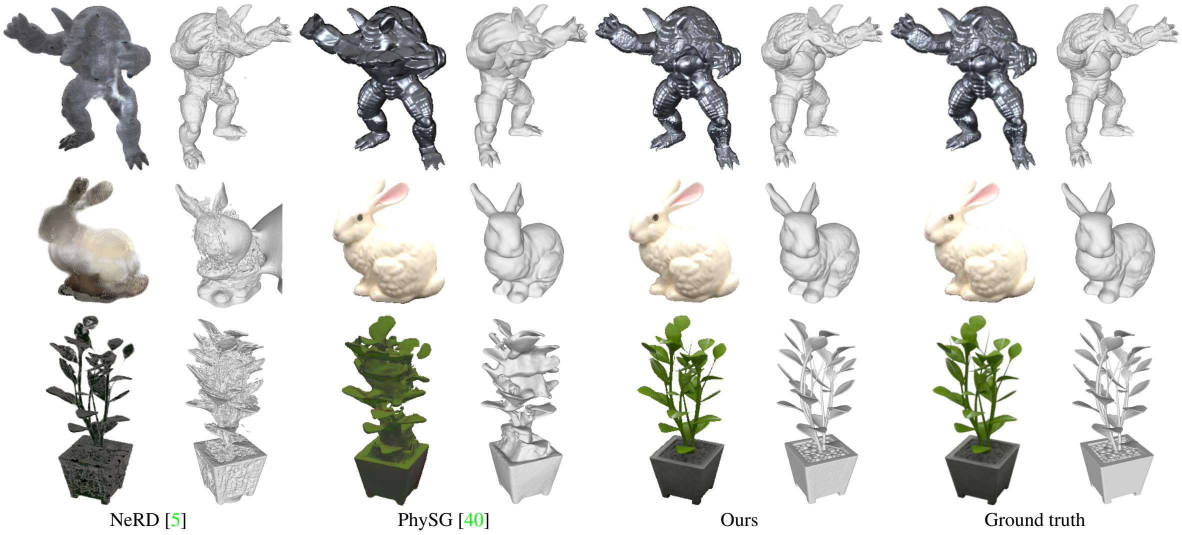
Figure 3. Example synthetic images under ambient lighting.

Table 1. Quantitative comparison of geometry estimations on synthetic scenes. Distance errors are defined as the mesh-to-mesh distance on the visible parts of surface. **Distance and depth errors are measured in ratio to 10^{-3} of object lengths.** Normal errors are measured in degrees. Our method consistently outperforms both NeRD and PhySG by a substantial margin.

Objects	Mean/Median errors per method per metric								
	NeRD [5]			PhySG [40]			Ours		
	Distance*	Depth	Normal	Distance	Depth	Normal	Distance	Depth	Normal
Armadillo	5.0 / 2.9	14.0 / 4.3	22.1 / 18.3	9.0 / 3.1	23.4 / 3.4	15.5 / 9.8	**1.0 / 0.7**	**2.1 / 0.7**	**6.4 / 4.5**
Bunny	64.9 / 48.7	168.2 / 79.4	69.8 / 68.5	3.8 / 1.7	7.4 / 1.7	7.7 / 5.0	**1.5 / 1.0**	**2.6 / 1.4**	**4.5 / 3.3**
LegoCar	-	-	-	34.1 / 26.6	64.1 / 29.3	39.2 / 29.2	**16.2 / 8.6**	**10.6 / 3.1**	**19.3 / 9.1**
Plant	15.4 / 13.4	47.0 / 15.3	26.0 / 18.5	16.7 / 13.6	69.1 / 23.4	31.5 / 17.6	**1.2 / 0.9**	**4.3 / 1.2**	**7.1 / 3.3**

*NeRD does not define a surface geometry for direct comparison. To obtain a surface from NeRD, we extract a point cloud by tracing camera rays' expected depths, followed by outlier removal and Poisson surface reconstruction [16].

-NeRD failed on the LegoCar sequence and produced an empty scene with zero density.



NeRD [5]

PhySG [40]

Ours

Ground truth

Figure 4. Visual comparison of novel rendering and surface geometry versus ground truth. NeRD uses a particle cloud geometry representation that results in bumpy surfaces. PhySG produces a hard, smooth surface but is unable to deal with complex geometries. Our method produces arguably better results than both baselines.

flashlight suppresses ambient lighting, $\hat{\mathcal{L}}$ becomes dominant, effectively yielding darkroom images. On the other hand, if the flashlight is dimmed out, the flashlight model $\hat{\mathcal{L}}$, in particular reflectance predictions Θ, lose all physical significance, and our method reverts to vanilla neural light field with NeuS [36] as backbone.

Figure 5 compares our result, produced with flashlight/non-flashlight images, to NeuS [36] trained solely from non-flashlight images. NeuS can generate inconsistent geometry on specular regions due to inherent ambiguity between appearance and geometry. We were able to disambiguate this case by adding a physically-based flashlight reflection model and jointly solving BRDF and shape.

Figure 6 illustrates the reconstructed objects at different

Table 2. Quantitative novel view and relighting results on synthetic scenes.

	NeRD [5]*		PhySG [40]		Ours	
	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM
Armadillo	21.27	0.811	33.86	0.957	**40.39**	**0.987**
Bunny	15.69	0.799	19.33	0.953	**21.12**	**0.964**
LegoCar	-	-	29.12	0.935	**38.40**	**0.986**
Plant	20.66	0.819	15.15	0.924	**37.54**	**0.965**

*Results of NeRD were obtained where ground truth images were used to solve for environment illumination maps.

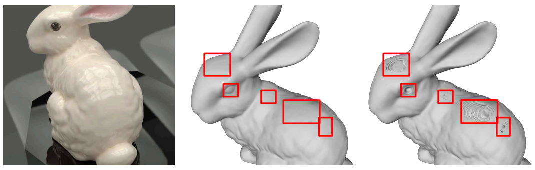

Input (HDR tone mapped) Ours NeuS [36]

Figure 5. Without flashlight, NeuS [36] creates holes on specular parts while our method is robust to specularities. Better viewed on screen zoom-in.

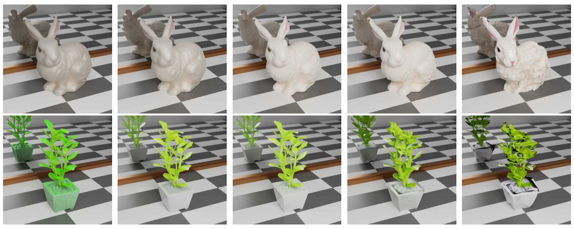

(a) $ratio$ 0.2 (b) $ratio$ 0.4 (c) $ratio$ 0.6 (d) $ratio$ 0.8 (e) $ratio$ 1

Figure 6. Reconstruction quality with varying flashlight intensities. Here $ratio \in [0.2, 1]$ defines the strength of flashlight reflections over image intensities in the input images. That is, 6a shows reconstructed objects with the least bright flashlight, while 6e shows results in darkroom setting. Our method works best when the flashlight component makes up around half of total incoming lights.

ratios of flashlight component over total image intensities (*i.e*. ratio $= 0$ means ambient-only, and ratio $= 1$ means darkroom). When flashlight is dim, the reflectance parameters are insufficiently supervised and only geometry can be recovered. Interestingly, we see a considerable performance drop in geometry estimations when the flashlight intensity overwhelms ambient illuminations. We attribute this to three reasons: (a) previous work showed neural light fields harness empirically powerful priors for regularizing geometry [38], however this prior is lost when ambient illumination becomes weak (b) the darkroom inverse rendering setup, although well-posed, is known to be highly nonconvex [9, 26] and hence sensitive to network initialization, and (c) in the darkroom setting, any non-zero output from ambient network becomes noise that disrupts optimization. Some potential solutions are to pre-train the intrinsic network on the object masks alone, and/or to involve additional geometry priors (*e.g*. surface smoothness), and/or to scale down or disable ambient network. However, considering in real world the flashlight is unlikely to suppress environment illumination, we leave such improvement for future work.

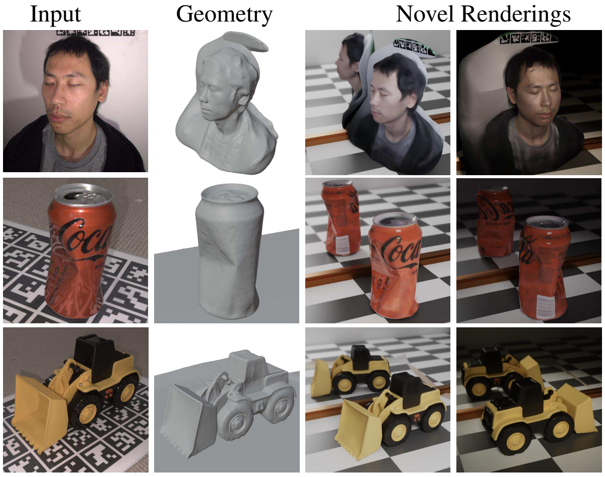


Figure 7. Visualization of reconstruction quality for three real indoor scenes.

5.5. Real world experiments

Figure 7 illustrates our results on real world scenes. Unlike with the synthetic experiments before, here we do not provide object segmentation masks, but rely on the background NeRF to differentiate foreground objects from the background without manual intervention. The results show that partial scene inside the foreground regions of interest is well recovered with high fidelity.

6. Conclusion

In this paper we proposed a practical setup for obtaining scene geometry and reflectance. Compared to other in-the-wild inverse rendering methods, our approach does not explicitly decompose ambient reflections, but instead rely on the ambient and flashlight shaders for regularizing geometry and reflectance; we achieved arguably better reconstruction quality than competing state-or-arts. Two future directions to be explored are: (a) to extend our solution to large scenes with adjustable indoor lighting, and (b) to reconstruct translucent objects.

References

[1] Matan Atzmon and Yaron Lipman. Sal: Sign agnostic learning of shapes from raw data. In *Proceedings of*

the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pages 2565–2574, 2020. 6

[2] Sai Bi, Zexiang Xu, Pratul Srinivasan, Ben Mildenhall, Kalyan Sunkavalli, Miloš Hašan, Yannick Hold-Geoffroy, David Kriegman, and Ravi Ramamoorthi. Neural reflectance fields for appearance acquisition. *arXiv preprint arXiv:2008.03824*, 2020. 2

[3] Sai Bi, Zexiang Xu, Kalyan Sunkavalli, Miloš Hašan, Yannick Hold-Geoffroy, David Kriegman, and Ravi Ramamoorthi. Deep reflectance volumes: Relightable reconstructions from multi-view photometric images. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part III 16*, pages 294–311. Springer, 2020. 2

[4] Sai Bi, Zexiang Xu, Kalyan Sunkavalli, David Kriegman, and Ravi Ramamoorthi. Deep 3d capture: Geometry and reflectance from sparse multi-view images. In *CVPR*, pages 5960–5969, 2020. 2

[5] Mark Boss, Raphael Braun, Varun Jampani, Jonathan T Barron, Ce Liu, and Hendrik Lensch. Nerd: Neural reflectance decomposition from image collections. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 12684–12694, 2021. 2, 3, 5, 6, 7, 8

[6] Mark Boss, Varun Jampani, Raphael Braun, Ce Liu, Jonathan T. Barron, and Hendrik P.A. Lensch. Neural-pil: Neural pre-integrated lighting for reflectance decomposition. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2021. 3, 5

[7] Brent Burley. Physically-based shading at disney. In *ACM SIGGRAPH Course Notes. Practical physically-based shading in film and game production.*, volume 2012, pages 1–7, 2012. 4

[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. 2

[9] Ziang Cheng, Hongdong Li, Yuta Asano, Yinqiang Zheng, and Imari Sato. Multi-view 3d reconstruction of a textureless smooth surface of unknown generic reflectance. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 16226–16235, 2021. 2, 3, 8

[10] Ziang Cheng, Hongdong Li, Richard Hartley, Yinqiang Zheng, and Imari Sato. Diffeomorphic neural surface parameterization for 3d and reflectance acquisition. In *ACM SIGGRAPH*. 2022. 2

[11] Amaël Delaunoy and Marc Pollefeys. Photometric bundle adjustment for dense multi-view 3d modeling. In *CVPR*, pages 1486–1493, 2014. 2

[12] Michael Garland and Paul S Heckbert. Surface simplification using quadric error metrics. In *Proceedings of the 24th annual conference on Computer graphics and interactive techniques*, pages 209–216, 1997. 5

[13] Hyunho Ha, Seung-Hwan Baek, Giljoo Nam, and Min H Kim. Progressive acquisition of svbrdf and shape in motion. In *Computer Graphics Forum*, volume 39, pages 480–495. Wiley Online Library, 2020. 2

[14] Tomoaki Higo, Yasuyuki Matsushita, Neel Joshi, and Katsushi Ikeuchi. A hand-held photometric stereo camera for 3-d modeling. In *ICCV*, pages 1234–1241. IEEE, 2009. 2

[15] Berk Kaya, Suryansh Kumar, Francesco Sarno, Vittorio Ferrari, and Luc Van Gool. Neural radiance fields approach to deep multi-view photometric stereo. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 1965–1977, 2022. 2

[16] Michael Kazhdan, Matthew Bolitho, and Hugues Hoppe. Poisson surface reconstruction. In *Proceedings of the fourth Eurographics symposium on Geometry processing*, volume 7, 2006. 7

[17] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014. 6

[18] Thomas Lewiner, Hélio Lopes, Antônio Wilson Vieira, and Geovan Tavares. Efficient implementation of marching cubes' cases with topological guarantees. *Journal of graphics tools*, 8(2):1–15, 2003. 5

[19] Min Li, Zhenglong Zhou, Zhe Wu, Boxin Shi, Changyu Diao, and Ping Tan. Multi-view photometric stereo: A robust solution and benchmark dataset for spatially varying isotropic materials. *IEEE Transactions on Image Processing*, 29:4159–4173, 2020. 1, 2

[20] Zhengqin Li, Zexiang Xu, Ravi Ramamoorthi, Kalyan Sunkavalli, and Manmohan Chandraker. Learning to reconstruct shape and spatially-varying reflectance from a single image. *ACM Transactions on Graphics (TOG)*, 37(6):1–11, 2018. 2

[21] Fotios Logothetis, Roberto Mecca, and Roberto Cipolla. A differential volumetric approach to multi-view photometric stereo. In *ICCV*, pages 1052–1061, 2019. 1, 2

[22] Fujun Luan, Shuang Zhao, Kavita Bala, and Zhao Dong. Unified shape and svbrdf recovery using differentiable monte carlo rendering. In *Computer Graphics Forum*, volume 40, pages 101–113. Wiley Online Library, 2021. 2, 3

[23] Robert Maier, Kihwan Kim, Daniel Cremers, Jan Kautz, and Matthias Nießner. Intrinsic3d: High-quality 3d reconstruction by joint appearance and geometry optimization with spatially-varying lighting. In *ICCV*, pages 3114–3122, 2017. 2

[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 *European conference on computer vision*, pages 405–421. Springer, 2020. 2, 3, 4

[25] Jacob Munkberg, Jon Hasselgren, Tianchang Shen, Jun Gao, Wenzheng Chen, Alex Evans, Thomas Mueller, and Sanja Fidler. Extracting Triangular 3D Models, Materials, and Lighting From Images. *arXiv:2111.12503*, 2021. 2, 3

[26] Giljoo Nam, Joo Ho Lee, Diego Gutierrez, and Min H Kim. Practical svbrdf acquisition of 3d objects with unstructured flash photography. *ACM Transactions on Graphics (TOG)*, 37(6):1–12, 2018. 1, 2, 3, 8

[27] Michael Niemeyer, Lars Mescheder, Michael Oechsle, and Andreas Geiger. Differentiable volumetric rendering: Learning implicit 3d representations without 3d supervision. In *CVPR*, pages 3504–3515, 2020. 2

[28] Michael Oechsle, Songyou Peng, and Andreas Geiger. Unisurf: Unifying neural implicit surfaces and radiance fields for multi-view reconstruction. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 5589–5599, 2021. 2, 4

[29] Geoffrey Oxholm and Ko Nishino. Shape and reflectance from natural illumination. In *ECCV*, pages 528–541. Springer, 2012. 2

[30] Geoffrey Oxholm and Ko Nishino. Multiview shape and reflectance from natural illumination. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2155–2162, 2014. 3

[31] Jaesik Park, Sudipta N Sinha, Yasuyuki Matsushita, Yu-Wing Tai, and In So Kweon. Robust multiview photometric stereo using planar mesh parameterization. *IEEE transactions on pattern analysis and machine intelligence*, 39(8):1591–1604, 2016. 1, 2

[32] Carolin Schmitt, Simon Donne, Gernot Riegler, Vladlen Koltun, and Andreas Geiger. On joint estimation of pose, geometry and svbrdf from a handheld scanner. In *CVPR*, pages 3493–3503, 2020. 2

[33] Pratul P Srinivasan, Boyang Deng, Xiuming Zhang, Matthew Tancik, Ben Mildenhall, and Jonathan T Barron. Nerv: Neural reflectance and visibility fields for relighting and view synthesis. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7495–7504, 2021. 2, 3, 5

[34] Daniel Vlasic, Pieter Peers, Ilya Baran, Paul Debevec, Jovan Popović, Szymon Rusinkiewicz, and Wojciech Matusik. Dynamic shape capture using multi-view photometric stereo. In *ACM SIGGRAPH Asia*, pages 1–11. 2009. 2

[35] Bruce Walter, Stephen R Marschner, Hongsong Li, and Kenneth E Torrance. Microfacet models for refraction through rough surfaces. *Rendering techniques*, 2007:18th, 2007. 4

[36] 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. *arXiv preprint arXiv:2106.10689*, 2021. 2, 4, 5, 7, 8

[37] Chenglei Wu, Yebin Liu, Qionghai Dai, and Bennett Wilburn. Fusing multiview and photometric stereo for 3d reconstruction under uncalibrated illumination. *IEEE transactions on visualization and computer Graphics*, 17(8):1082–1095, 2010. 2

[38] Lior Yariv, Yoni Kasten, Dror Moran, Meirav Galun, Matan Atzmon, Basri Ronen, and Yaron Lipman. Multiview neural surface reconstruction by disentangling geometry and appearance. *Advances in Neural Information Processing Systems*, 33:2492–2502, 2020. 2, 3, 4, 8

[39] Kai Zhang, Fujun Luan, Zhengqi Li, and Noah Snavely. Iron: Inverse rendering by optimizing neural sdfs and materials from photometric images. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2022. 2

[40] 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 *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5453–5462, 2021. 2, 3, 5, 6, 7, 8

[41] Kai Zhang, Gernot Riegler, Noah Snavely, and Vladlen Koltun. Nerf++: Analyzing and improving neural radiance fields. *arXiv preprint arXiv:2010.07492*, 2020. 3, 4, 5

[42] Li Zhang et al. Shape and motion under varying illumination: Unifying structure from motion, photometric stereo, and multiview stereo. In *Proceedings Ninth IEEE International Conference on Computer Vision*, pages 618–625. IEEE, 2003. 3

[43] Xiuming Zhang, Pratul P Srinivasan, Boyang Deng, Paul Debevec, William T Freeman, and Jonathan T Barron. Nerfactor: Neural factorization of shape and reflectance under an unknown illumination. *ACM Transactions on Graphics (TOG)*, 40(6):1–18, 2021. 2, 3, 5

[44] Zhenglong Zhou, Zhe Wu, and Ping Tan. Multi-view photometric stereo with spatially varying isotropic materials. In *CVPR*, pages 1482–1489, 2013. 2