Neural Reflectance for Shape Recovery with Shadow Handling

Junxuan Li [1,2], Hongdong Li [1]
Australian National University [1] Data61, CSIRO [2]
{junxuan.li, hongdong.li}@anu.edu.au

Abstract

This paper aims at recovering the shape of a scene with unknown, non-Lambertian, and possibly spatially-varying surface materials. When the shape of the object is highly complex and that shadows cast on the surface, the task becomes very challenging. To overcome these challenges, we propose a coordinate-based deep MLP (multilayer perceptron) to parameterize both the unknown 3D shape and the unknown reflectance at every surface point. This network is able to leverage the observed photometric variance and shadows on the surface, and recover both surface shape and general non-Lambertian reflectance. We explicitly predict cast shadows, mitigating possible artifacts on these shadowing regions, leading to higher estimation accuracy. Our framework is entirely self-supervised, in the sense that it requires neither ground truth shape nor BRDF. Tests on real-world images demonstrate that our method outperform existing methods by a significant margin. Thanks to the small size of the MLP-net, our method is an order of magnitude faster than previous CNN-based methods.

1. Introduction

Recovering the 3D shape of a non-Lambertian object from its multiple photometric images taken by a fixed camera remains a challenging task. The diverse nature of real-world materials manifests a wide range of specularities on the surface, impeding traditional photometric methods [12, 20, 31, 32]. Moreover, shadows commonly appear in non-convex objects occluding part of the object surface, hindering surface normal estimation. Previous attempts to handle shadows often rely on a rather restrictive Lambertian assumption [5]. The problem becomes much complicated if both specularities and shadows appear on the surface.

With the recent advent of deep learning, tremendous progresses have been made in many computer vision problems, and there is no exception for photometric 3D reconstruction [6, 11, 15, 16, 23, 33]. Current existing deep learning methods often tackle the problem in a supervised training manner. The underlying physics principle of image formation are not duly utilized. In addition, the lack of interpretability of deep learning methods prevents leveraging the interactions between object appearance and surface normals. Despite various synthetic datasets with augmentation strategies [7, 11, 16, 23], it remains an open challenge to process real-world images with both specularities and shadows.

In this paper, we propose an unsupervised neural network method that overcomes the issues mentioned above. Our framework takes the image coordinates corresponding to a surface point as the input, and directly outputs the surface normal, reflectance parameters (*i.e*. diffuse albedo and specular parameters), and depth at that surface point. We proposed a series of neural specular basis functions to account for the different types of specularities in the real-world. Our neural bases provide the parameterization for the surface reflectance and fit the object's appearance to obtain the accurate surface normal. Furthermore, our framework explicitly parameterizes the shadowed regions by tracing through the estimated depth map. These shadowed regions are then excluded from computation in order to avoid possible rendering artifacts. Following the inverse graphics rendering idea, we use the estimated surface normal and neural reflectance to re-render the pixel intensities of the surface point under different light directions. Our framework is optimized by minimizing the difference between the reconstructed and observed images during the inference time. Therefore, there is no need for any ground truth data or pre-training. Our method outperforms both the supervised and self-supervised state-of-the-art methods on the challenging real-world dataset of DiLiGenT [24]. Compared to other self-supervised deep methods [13, 26], our framework is ten times faster.

2. Related Work

Conventional approaches: The photometric stereo is firstly introduced by Woodham [30], which assumes the surface of the objects to be Lambertian and convex to avoid the specular effects and shadows. This problem can therefore be solved in a closed-form manner by least-squares. The above strict assumptions were gradually liberalized by later studies [12, 20, 21, 31, 32]. These methods can tolerate the

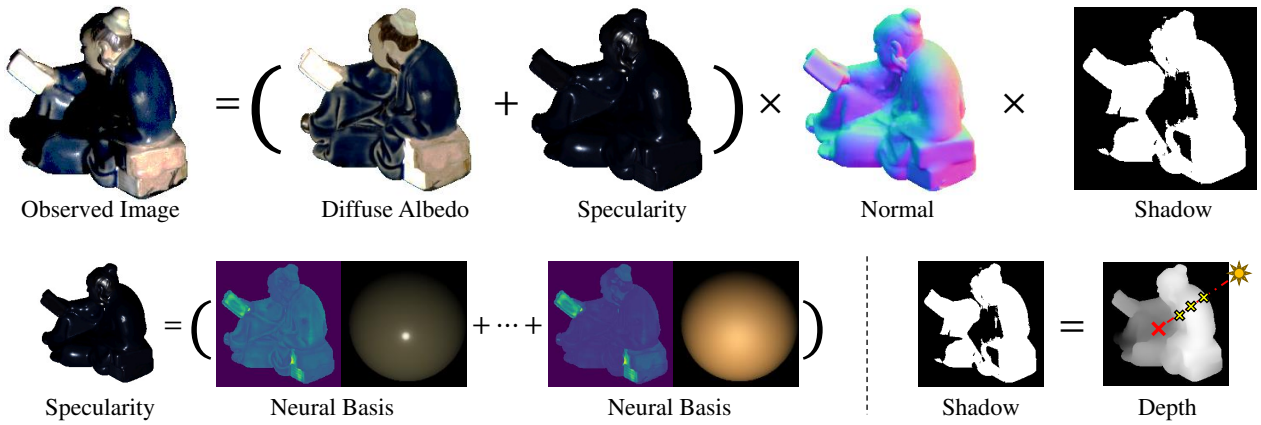

Figure 1. We propose an self-supervised framework that estimates the surface normal, diffuse albedo, specularity, and shadow of an object. Our method learns the neural basis to fit the observed specularities accurately and gives clues for normal estimation. We also explicitly parameterize the shadows based on the estimated depth, alleviating artifacts on these shadows.

existence of non-Lambertian effects by treating the specularities and cast shadows on the object as outliers. However, they may also erase other clues specularities can bring.

Supervised methods: With the progress of deep learning in many of the computer vision areas, the learning-based methods are the ones that have achieved the best performance in photometric stereo recently [6, 9, 11, 15, 16, 23, 29, 33, 36]. Santo *et al*. [23] proposed the first network-based method, which per-pixelly estimates the normal by taking observed pixels in a pre-defined order. Chen *et al*. [6, 7] proposed a feature-extractor and features-pooling strategy to obtain the spatial information for photometric stereo. Recently, more works [29, 33] exploited the local and global photometric clues for this problem. These learning-based methods require a large amount of data with ground truth surface normal at the training stage. The synthesized data with some augmentation strategies are commonly used as collecting a large-scale real-world dataset is exceptionally expansive and impractical.

Self-supervised methods: In contrast to the above-mentioned learning-based methods method, self-supervised methods do not require ground truth normal at supervision. Instead, the network is optimized by minimizing the difference between the reconstructed images and observed images. Taniai *et al*. [26] proposed a self-supervised network that takes the whole set of images at the input, directly output the surface normal, and aiming to reconstruct the observed images. Their network structure is further expanded by Kaya *et al*. [13] to deal with interreflection in the context of uncalibrated photometric stereo. Both of them implicitly encode specular components as features for the network and fail to consider shadows in the rendering equation.

Neural radiance fields: Recently, neural radiance fields introduced by NeRF [19] is widely adopt in many reconstruction tasks in computer vision. Many works also extend the neural radiance fields to recover both the shapes and materials of the object [2, 25, 34, 35]. These works are solving multi-view reconstruction problems. They generally assume the input being images of an object captured from multiple viewpoints under fix illumination. In contrast, the photometric stereo problem we are focusing in this paper assumes multiple images taking from the same viewpoint, but with different illuminations.

3. Proposed Method

As shown in Fig. 1, our framework aims at decoupling the surface into normal, diffuse albedo, specularity, and shadow. We model the specularity by learning a set of neural specular bases. Our method estimates the depth by querying the relative depth of the surface points. In the following subsections, we illustrate the details of each module in our framework.

3.1. Rendering Equation

Following the conventional calibrated photometric stereo problem, we assume that the light source is in distance over the images with known light direction $\boldsymbol{l} = [l_x, l_y, l_z]^T \in \mathcal{S}^2$ (the space of 3-dimensional unit vectors) and light intensity $L_i \in \mathbb{R}_+$. And the camera to be in orthographic position, hence, viewing direction $\boldsymbol{v} = [0, 0, -1]^T \in \mathcal{S}^2$. For simplicity, without any loss of generality, we omit the light intensity L_i in the following formulations by dividing the observations (*i.e*. images I_i) with the corresponding lighting intensities, $I = I_i / L_i$. We also assume that there are no inter-reflections between the surfaces so that the point light source is the only light source to illuminate the target object.

Given a light source from the direction $\boldsymbol{l}$ illuminates a surface point with surface normal $\boldsymbol{n} \in \mathcal{S}^2$. The observation

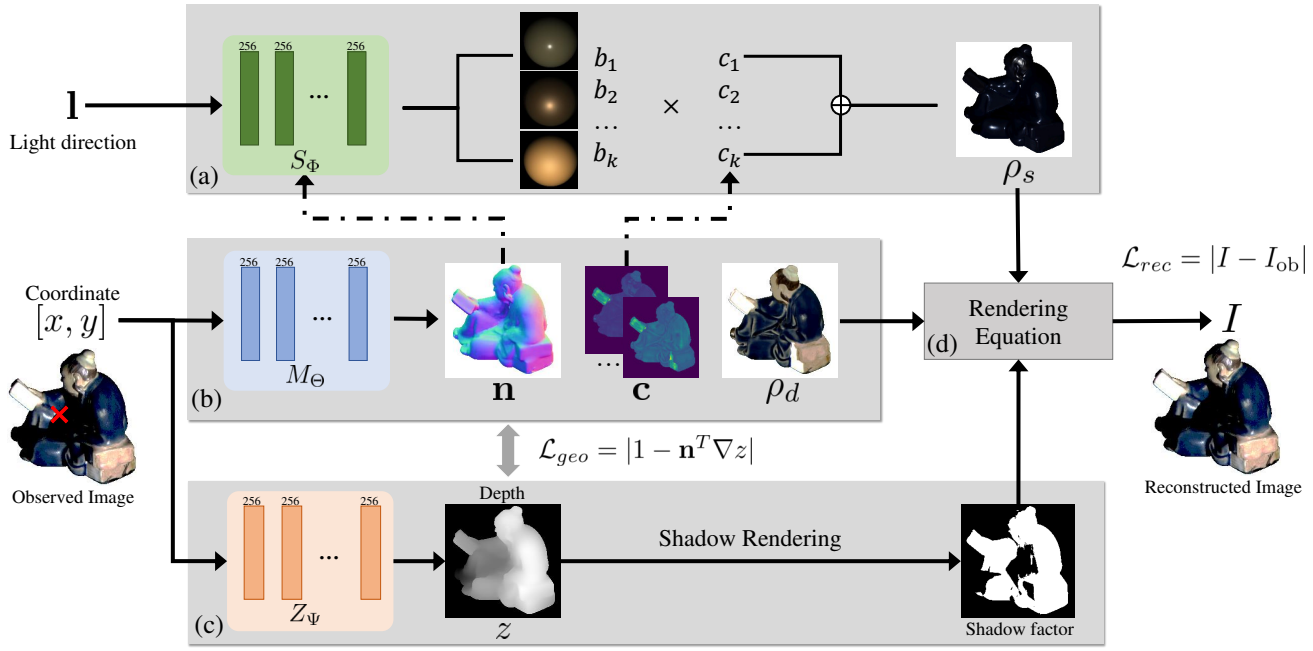

Figure 2. The four modules of our MLP-based deep photometric stereo framework: (a) neural specular bases modeling S_Φ (see Sec. 3.2) fits a suitable set of suitable BRDF bases to the target specularities; (b) surface modeling M_Θ (see Sec. 3.3) estimates the surface normal, as well as parameters of the BRDF given the image coordinates as input; (c) Z_Ψ estimates a dense depth map, which enables the shadow rendering (see Sec. 3.4) by checking the visibility of the light source at each surface point; and (d) the rendering equation (see Sec. 3.1). All MLPs are optimized in a self-supervised manner by minimizing the reconstruction error between reconstructed and observed images.

I viewing from direction $\boldsymbol{v}$ can be written as

$$I = s\rho(\boldsymbol{l}, \boldsymbol{v}, \boldsymbol{n}) \max(\boldsymbol{l}^T \boldsymbol{n}, 0), \tag{1}$$

where $s \in \{0, 1\}$ is a binary variable with a value of 0 at shadows, and 1 otherwise; $\rho(\boldsymbol{l}, \boldsymbol{v}, \boldsymbol{n})$ represents the BRDF of the surface point, which is a function of the light, view direction, and the surface normal; $\max(\boldsymbol{l}^T \boldsymbol{n}, 0)$ is the shading component.

3.2. Reflectance Modeling

The Lambertian surface assumes the BRDF $\rho(\boldsymbol{l}, \boldsymbol{v}, \boldsymbol{n}) = \rho_\mathrm{d}$ is always a positive constant. This unrealistic assumption fails to account for those materials with high specular effects. It can be beneficial to model the specular part in BRDF and leverage its information for photometric stereo. In order to take both the diffuse and specular effects into account, here we choose a more realistic way to model the surface reflectance, *i.e.* the microfacet BRDF models [27, 28], where the BRDF is separated into the diffuse and specular components

$$\rho(\boldsymbol{l}, \boldsymbol{v}, \boldsymbol{n}) = \rho_\mathrm{d} + \rho_\mathrm{s}(\boldsymbol{l}, \boldsymbol{v}, \boldsymbol{n}). \tag{2}$$

Neural Specular Basis Previous deep-learning-based approaches implicitly handle the specularity on images by feeding them as features into their neural network [13, 26], or processed by max-pooling [6, 7]. However, as the specularities, at the core, are reflections on the surface, explicitly model these effects by using clues from physical reflection constraints will certainly bring merits to the photometric stereo problem.

To relieve the burden of fitting such a neural specular BRDF, we need to introduce some reasonable and realistic assumptions. Recalling that the BRDF can be converted to a half-vector $\boldsymbol{h}$ based function with only four parameters [22], we assume that our specular BRDF is isotropic and is only the function of half-vector $\boldsymbol{h}$ and surface normal $\boldsymbol{n}$. This assumption omits the Fresnel reflection coefficient and the geometric attenuation, which only has limited effects at grazing angles [3]. Besides, observing the fact that many surface points in the real-world object are similar, if not identical, in the material. We further assume that the specular BRDF $\rho_\mathrm{s}(\boldsymbol{l}, \boldsymbol{v}, \boldsymbol{n})$ at each surface point lies on a non-negative linear combination of the atoms of specular basis. Similar approaches for simplifying the BRDF model to be the combination of different bases were also used in previous works [10, 17]. The specular BRDF can then be written as

$$\rho_\mathrm{s}(\boldsymbol{l}, \boldsymbol{v}, \boldsymbol{n}) = \mathbf{c}^T \, D(\boldsymbol{h}, \boldsymbol{n}), \quad \boldsymbol{h} = \frac{\boldsymbol{l} + \boldsymbol{v}}{||\boldsymbol{l} + \boldsymbol{v}||}, \tag{3}$$

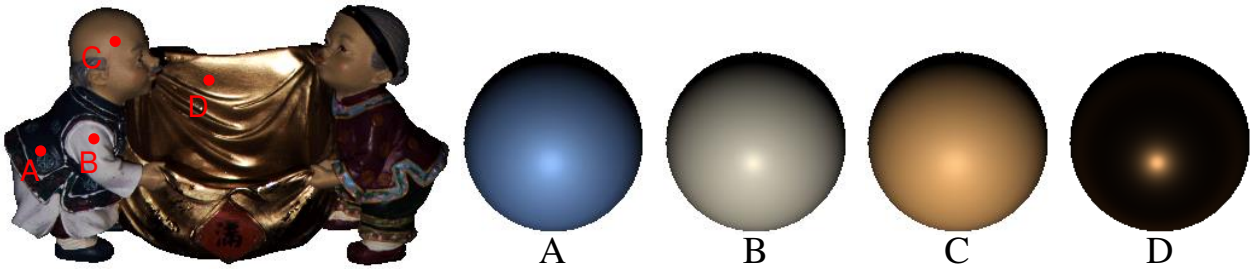

Figure 3. **Visualization on the estimated svBRDFs.** We select four different surface points on the object "Harvest" and showcase our estimated BRDF spheres on the right. The results demonstrate that our model can recover the metallic and diffuse materials. We scale up the observed images and normalize the BRDF spheres for better visualization.

where $\boldsymbol{h}$ is the half-vector between lighting and viewing direction; and $D(\boldsymbol{h}, \boldsymbol{n}) = [b_1, b_2, \cdots, b_k]^T$ is the underlying specular basis of the target object; $[c_1, c_2, \cdots, c_k]^T := \mathbf{c} \in \mathbb{R}_+^k$ represent the weights of each specular basis; k is the number of different bases. We assume that $\mathbf{c}$ is an element-wise non-negative vector, suggesting that the surface reflectance is represented by positive combination of a small number of basis materials.

We use an MLP to parameterize the specular basis by

$$D(\boldsymbol{h}, \boldsymbol{n}) = S_\Phi(\boldsymbol{h}, \boldsymbol{n}), \tag{4}$$

The network $S_\Phi(\boldsymbol{h}, \boldsymbol{n})$ only takes $\boldsymbol{h}, \boldsymbol{n}$ at the input, outputs the different specular basis in form of $[b_1, b_2, \cdots, b_k]^T$, as shown in Fig. 2. Φ are its weights that can be optimized during testing. It is well established that a variety of reflectance maps can be represented by a linear combination of a few basis functions [8,17,18]. We empirically set $k = 9$ when testing our model on real datasets. In Fig. 3, we re-rendered several spheres by using our estimation on the reflectance and neural basis of the surface points. As shown in Fig. 3, our neural reflectance modeling can approximate the spatially-varying and non-Lambertian materials very well. It can recover the diffuse surface, and also reliably construct the high-peak and long-tail metallic specularites.

3.3. Surface modeling

We model the surface normal, diffuse, and neural basis coefficients of an object by an MLP M_Θ. It takes the image coordinates of the pixels $\boldsymbol{x} = [x, y]^T \in \mathbb{R}^2$ as input. The output is the corresponding surface normal $\boldsymbol{n}$, diffuse albedo ρ_d, and the coefficients $\mathbf{c}$ of the bases at each coordinate $\boldsymbol{x}$.

$$\boldsymbol{n}, \rho_\mathrm{d}, \mathbf{c} = M_\Theta(\boldsymbol{x}), \tag{5}$$

where $\mathbf{c}$ represents the coefficients that can be used to reconstruct the specular component ρ_s in Sec. 3.2; and Θ is the weights of this MLP that can be optimized.

We use a similar MLP architecture and positional encoding strategy from NeRF [19] to build our network, and the embedding in input coordinates $\boldsymbol{x}$. The difference is that while NeRF also takes different viewing directions as input to model the view-dependent effects of the objects' appearance, our M_Θ network only estimates the "static" properties of the target object. Instead, we cover the "light-dependent" variance of the object by neural reflectance modeling. Our design will encourage the network to correctly decompose surface normal and material property of the object.

3.4. Shadow handling

We now look at the shadow factor s in the image rendering Eq. (1). Due to the rugged surface of the objects in the world, shadows may appear at the reflecting surface. As shown in Fig. 4, shadow occurs when the object itself occludes the surface. Rendering of the shadowed region relies on the relative geometry and depth of the object with respect to the light directions. Hence, we introduce a depth MLP Z_Ψ to model the object's depth value $z \in \mathbb{R}$ between the object surface points to the camera. The depth MLP takes images coordinates as input, outputs the corresponding depth value of the given coordinates $z = Z_\Psi(\boldsymbol{x})$.

To examine whether the object occludes the light source and hence causing the shadow, we can draw a line from the surface point $\boldsymbol{x}$ toward the light source. Denote this line in the world coordinates as $\mathbf{L} = \mathbf{X} - t\boldsymbol{l}$, where $t \in (0, +\infty)$; the $\mathbf{X} = [x, y, z]$ represent the surface points with its depth value z given by $Z_\Psi(\boldsymbol{x})$. We can further simplify the equation by using the function L_z to denote the z-axis value of $\mathbf{L}$. Now, by traveling along the light direction, *i.e.* $t \in (0, +\infty)$, we can compute the shadow factor by

$$s = \mathrm{step}\left(\min_{\boldsymbol{x}(t)}\left(Z_\Psi(\boldsymbol{x}(t)) - L_z(\boldsymbol{x}(t))\right)\right), \quad \boldsymbol{x}(t) = \boldsymbol{x} - t\boldsymbol{l}', \tag{6}$$

where the $\mathrm{step}(\cdot)$ denote the Heaviside step function, which outputs 1 if input is positive, and 0 otherwise; $\boldsymbol{l}' = [l_x, l_y]^T$ is the projection of light direction $\boldsymbol{l}$ at xy-plane. In implementation, we set the step size for shadow rendering to be 32 (with logspace intervals).

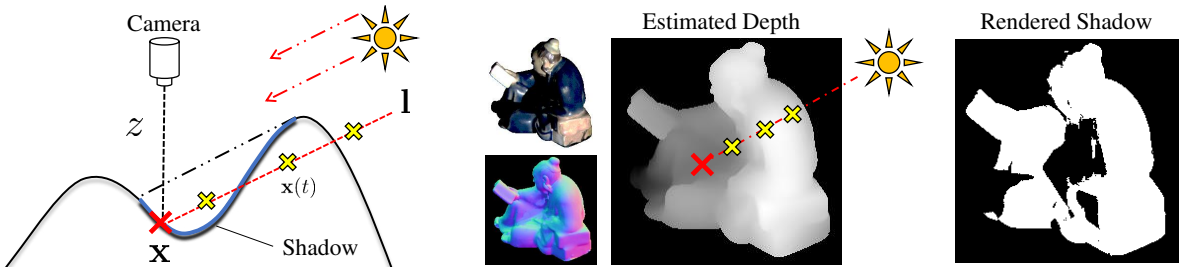

Figure 4. Shadow parameterization and rendering. As shown in the left figure, shadows are caused by self-occlusion. To determine whether a surface point $\boldsymbol{x}$ falls into the shadow region, we trace the point to the light source and sample multiple points $\boldsymbol{x}(t)$ along this ray. Given the light direction $\boldsymbol{l}$ and the estimated depth map $Z_\Psi(\boldsymbol{x})$, we can query the depth and compare the values to effectively parameterize and render the shadow by Eq. (6).

4. Implementation

We use the positional encoding [19] strategy to encode the input before inputting them into the MLP. For surface modeling net M_Θ, we encode the input with 10 levels of Fourier functions, the network M_Θ uses 12 fully-connected ReLU layers with 256 channels. The surface normal $\boldsymbol{n}$ is output at 8-th layer while the BRDF parameters are output at the last layer. We also use 10 encoding functions to embed the input of depth net Z_Ψ, which has 8 fully-connected ReLU layers with 256 channels. For the neural basis MLP S_Φ, we use only 3 encoding functions to embed the input. The network S_Φ consists of 3 fully-connected ReLU layers with 64 channels. Please refer to the supplementary material for more implementation details. Overall, the three MLP-networks are rather lightweight (*i.e.* small footprint) with total combined parameters of merely 1.1M. In contrast, the CNN-based self-supervised method [13] contains 3.7M parameters. Besides, our model is shallow and require less computation than previous works. Hence, our framework is much faster in inference time. The inference time in the 10 objects of DiLiGenT dataset range from 3 min to 9 min, with an average of 6 min per object. In contrast, CNN-based methods [13, 26] took about an hour per object.

Reconstruction loss. The reconstruction loss is defined as mean absolute errors between the observed intensity I_{ob} and reconstructed intensity:

$$\mathcal{L}_{rec} = \sum_{\text{all pixels}} |I - I_{\text{ob}}|. \tag{7}$$

Geometry Constraint. We introduce a geometry constraint between the estimated surface normal $\boldsymbol{n}$ and depth network Z_Ψ as below

$$\mathcal{L}_{geo} = \sum_{\text{all pixels}} (1 - \boldsymbol{n}^T \nabla Z_\Psi). \tag{8}$$

In the early stage of optimizing the network Z_Ψ, we introduce shadow guidance s_g to help with the training. Assume that observation under n different light direction is $[I_1, I_2, ..., I_n]$. We then set a threshold as $0.1\lambda_m$, where $\lambda_m = \frac{1}{n}\sum I_i$ is the mean intensity. Those pixel intensities that are smaller than the threshold will be discard. We use Eq. (6) for shadow rendering once the depth network Z_Ψ is stable.

Figure 5. Re-rendered image by our estimated svBRDFs. From left to right, we showcase the original image captured from "Harvest", the re-rendered image using our estimated neural svBRDFs, and the re-rendered image by ACLS [1], respectively. Our method achieves a better quality in reconstruction, being 2dB higher in peak signal-to-noise ratio (PSNR). ACLS failed to recover the spatially-varying materials (the red cloth and the human faces are all fainted in ACLS's result).

Smoothness constraint. Previous self-supervised methods suffered from poor network initialization [13,26]. Their networks required a pre-computed surface normal map as the early network guidance. In contrast, our model does not need any pre-computed surface normal as guidance. Instead, to cope with the poor network initialization problem, we use a smoothness constraint to guide the network in the early stages since the albedo and surface normal of real-world objects usually present a piece-wise smooth pattern

$$\mathcal{L}_{tv} = V_{l_1}(\rho_{\text{d}}, \mathbf{c}) + V_{l_2}(\boldsymbol{n}), \tag{9}$$

where V_{l_1} represents the total variation function with absolute loss and V_{l_2} with square loss.

To sum up, we optimize the parameters of the MLPs M_Θ, S_Φ, Z_Ψ by minimizing the following loss function: $\mathcal{L} = \mathcal{L}_{rec} + \mathcal{L}_{geo} + \beta\mathcal{L}_{tv}$, where β is the hyper-parameter controlling the total variation loss. We set it as $\beta = 0.01$; and it will then be set to 0 after the first half iterations.

Table 1. Quantitative comparison on the DiLiGenT dataset. The metric here is mean angular error (MAE); the lower MAE is preferred.

GT normal	Methods	Ball	Bear	Buddha	Cat	Cow	Goblet	Harvest	Pot1	Pot2	Reading	Avg.
No	Ours	2.43	**3.64**	**8.04**	**4.86**	**4.72**	**6.68**	**14.90**	**5.99**	**4.97**	**8.75**	**6.50**
No	TM18 [26]	**1.47**	5.79	10.36	5.44	6.32	11.47	22.59	6.09	7.76	11.03	8.83
No	BK21 [13]	3.78	5.96	13.14	7.91	10.85	11.94	25.49	8.75	10.17	18.22	11.62
No	L2 [30]	4.10	8.40	14.90	8.40	25.60	18.50	30.60	8.90	14.70	19.80	15.40
Yes	PX-NET [16]	2.00	**3.50**	7.60	**4.30**	**4.70**	**6.70**	**13.30**	**4.90**	**5.00**	**9.80**	**6.17**
Yes	WJ20 [29]	**1.78**	4.12	**6.09**	4.66	6.33	7.22	13.34	6.46	6.45	10.05	6.65
Yes	CNN-PS [11]	2.20	4.10	7.90	4.60	8.00	7.30	14.00	5.40	6.00	12.60	7.20
Yes	GPS-Net [33]	2.92	5.07	7.77	5.42	6.14	9.00	15.14	6.04	7.01	13.58	7.81
Yes	PS-FCN [7]	2.82	7.55	7.91	6.16	7.33	8.60	15.85	7.13	7.25	13.33	8.39

5. Experiments

In this section, we evaluate our method and its variants on the challenging real-world dataset DiLiGenT [24]. We used all the $n = 96$ images under different light directions for optimizing the network, except the object "Bear". We discard the first 20 images of "Bear", as they are found to be over-saturated in previous work [11]. The batch size is set as 8 images per batch. We iterate in total 6000 iterations when optimizing the network. We use Adam [14] optimizer with a learning rate of 5×10^{-4} and other parameters at their default settings. Our method is implemented in PyTorch and is running on a RTX 3090 GPU. The inference (*i.e.* training) time in the 10 objects of DiLiGenT dataset range from 3 min to 9 min, with an average of 6 min per object. In contrast, previous CNN-based methods [13, 26] took about an hour per object.

We also evaluate our method in two other challenging real world datasets: Gourd&Apple dataset [1], and Light Stage Data Gallery [4]. Please refer to supplementary material for more details.

5.1. Evaluation on real-world dataset

Surface normal evaluation. In Table 1, we present the quantitative comparison of our method against other methods on the DiLiGenT dataset. We use the mean angular error (MAE) as the metric in the paper. The lower MAE is preferred. We classify the previous methods into two categories: the supervised methods, which need ground truth surface normal at the training stage; and the self-supervised which does not need ground truth surface normal and directly estimates the normal at testing time. As reported in the Table 1, our method achieves the best performance over the other self-supervised methods at average MAE errors. Comparing to the previous self-supervised method [13, 26], our method is 2.33 degrees better in MAE errors. Thanks to our neural reflectance modeling, our method shows its significant advantages on shiny objects like "Reading", "Cow" and "Goblet". We present the visualization of "Cow" and "Pot2" in Fig. 6. "Cow" is a typical metallic-painted object with a high peak of specularities; while "Pot2" shows more broad and soft specular effects. Our method achieves the best performance in both two cases.

Table 2. Quantitative results on DiLiGenT with different number of images at the input. The average MAEs are shown in table.

GT Normal	# inputs	96	16	10	8
No	Ours	6.50	**6.82**	**7.47**	**7.70**
Yes	LMPS [15]	8.43	9.66	10.02	10.39
Yes	PX-Net [16]	**6.17**	–	8.37	–
Yes	SPLINE-NET [36]	–	–	10.35	–

svBRDF evaluation. In Fig. 3, we visualize the estimated svBRDFs on the challenging object "Harvest". "Harvest" contains many different type of materials over the surface. From diffuse (see point A), to specular (see point D), our model presents visually pleasing estimated BRDF spheres over these different points. To quantitatively evaluate our method, we re-rendered the observed image with our estimated reflectances, and ground truth lights. The results are shown in Fig. 5. We compare our re-rendered images with ACLS [1]. ACLS's BRDF fitting results are provided by Shi *et al*. [24], where it takes the ground truth surface normal when fitting the BRDF. By looking at the re-rendered images, our method achieve much higher reconstruction quality (2dB higher in peak signal-to-noise ratio (PSNR)). In comparison, ACLS [1] failed to faithfully recover the spatially-varying materials.

Results with Sparse Inputs. To evaluate how the performance changes with a different number of images at the input, we test our method on the DiLiGenT dataset. We follow the previous work LMPS [15] to use the same inputs for our method. The results and comparison are presented in Tab. 2. From left to right, our method takes 96 images, 16 images, 10 images and 8 images at the input separately. To our best knowledge, the trained model of SPLINE-Net [36] and PX-Net [16] is not publicly available. Hence, we report the value from their original paper. Although our method is not designed for sparse inputs, we still perform significantly better than previous work under a small number of inputs. It

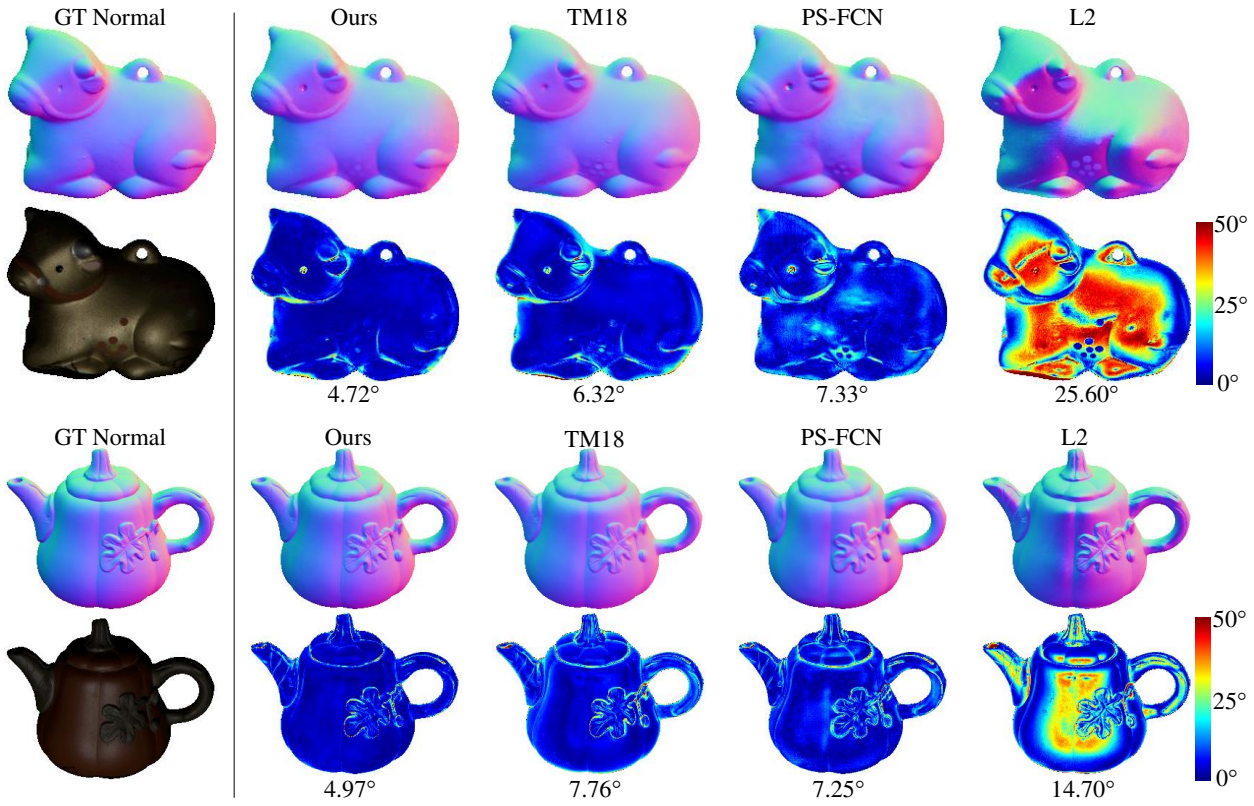

Figure 6. Qualitative results on "Cow" and "Pot2". For each object, the odd numbered rows show the observed image and estimated normal by different methods; the even numbered rows show the angular (normal) error in degrees by different methods.

demonstrates that our method is robust to the sparse inputs.

5.2. Ablation Study

Shadow handling: To show the efficacy of our shadow handling mechanism, we conduct ablation study by removing the shadow rendering module, denote as "w/o shadow". Quantitative comparisons are shown in Table 3, where one can see that the mean angular error on all objects increases 1.96 degrees. Notably, the performance degradation is majorly caused by objects "Buddha", "Harvest" and "Reading". This is as expected, because these objects have rather complex (concave) surface geometry, more susceptible to cast shadows. Our proposed shadow handling method attends to these shadowed regions better, achieving high recovery accuracy. In Fig. 7, we give the visualization of the effects of our shadowing module on the object "Reading". Observing the image and its ground truth normal of this object, we can see that "Reading" is a highly non-convex object with many specularities and shadows. The shadowed region is especially big when the light comes from the right direction, as shown in the lighting direction C and D in the figure. Our render shadows under these lighting directions, despite some minor errors, accurately predicting the shadowed regions. The error map shown at the right-most of the third row in Fig. 7 corresponds to the difference between the MAE yielded by our proposed model and its no-shadow-variant ("w/o s"). The negative areas, *i.e.*, blue regions in the error map, are those where our proposed model outperforms the alternative. The full model performs better in the region where the shadows are evident.

Table 3. Evaluations of the different variants of the proposed method. The second row is without using the early stage smoothness constraint; the third row is the method without the shadow factor s; the last row is without using specular component ρ_s. The metric here is MAE; lower is preferred.

Methods	Ball	Bear	Buddha	Cat	Cow	Goblet	Harvest	Pot1	Pot2	Reading	Avg.
Proposed	2.43	**3.64**	**8.04**	**4.86**	**4.72**	**6.68**	**14.90**	**5.99**	**4.97**	**8.75**	**6.50**
w/o $\mathcal{L}_{tv}$	2.44	3.66	8.56	4.93	5.27	6.77	21.67	6.73	6.88	9.19	7.61
w/o s	**2.13**	4.29	11.09	6.81	5.69	8.30	17.88	7.79	7.80	12.68	8.44
w/o ρ_s	3.13	6.48	10.58	6.93	27.23	15.19	29.65	8.27	14.14	11.41	13.30

Effectiveness of smoothness constraint: To show the effectiveness of proposed smoothness constraint, we conduct the experiments without using this loss, denote as "w/o $\mathcal{L}_{tv}$", shown in Table 3. The mean angular error on average is 1.11 degrees lower by leveraging this constraint.

Effectiveness of specular modeling: We further test the

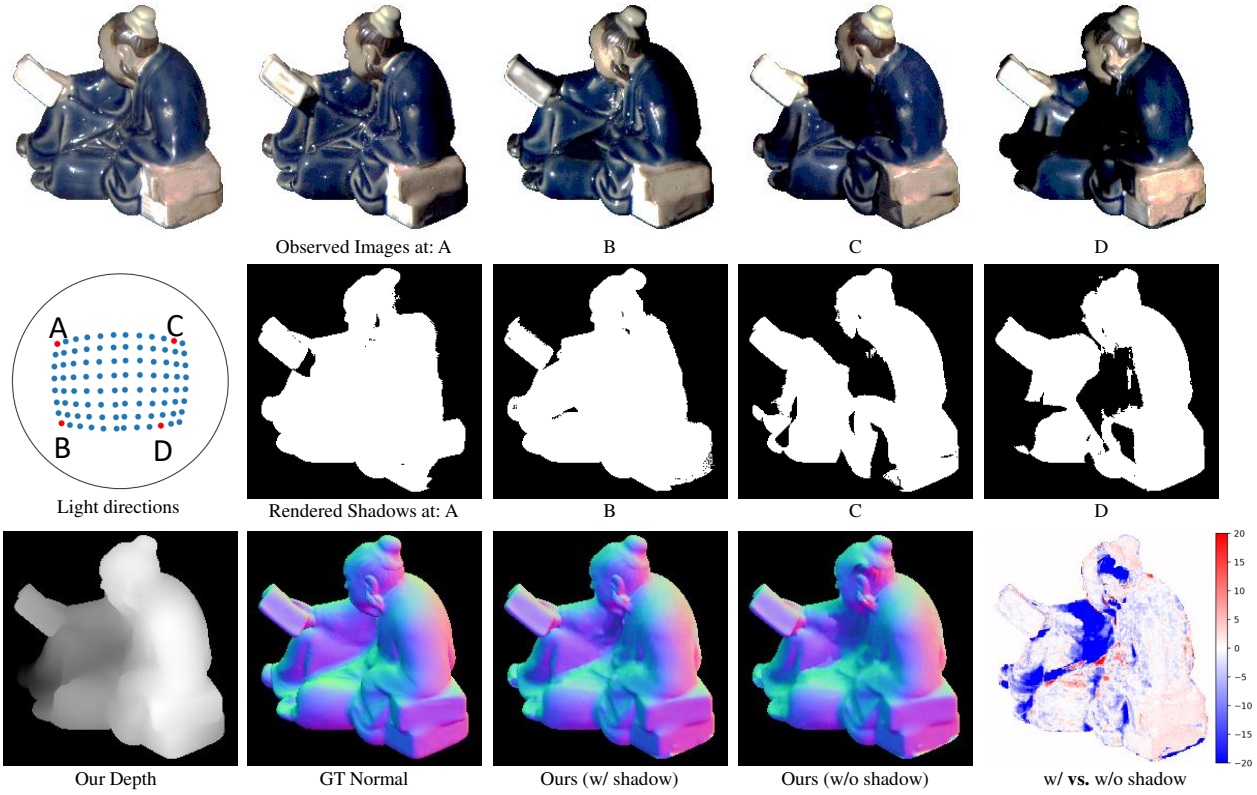

Figure 7. We select 4 different light directions. Their distribution is labeled as red points in the light distributions image in the second row. The first row shows the observed images under these 4 different light sources. The second row presents the results of our rendered shadow region under the corresponding illuminations. In the third row, we showcase the estimated depth, ground truth surface normal, estimated surface normal (with and without the shadow factor). In the right-most image on the third row, we also compare our estimated normal "w/ shadow" and "w/o shadow". The blue color in the comparison corresponds to the area where "w/ shadow" outperforms "w/o shadow".

model without using any specular modeling, denote as "w/o ρ_s", shown in Table 3. The performance is significantly worse without using the specular ρ_s. We can see that with specular components, our method improves a lot for the shiny objects like "Cow", "Goblet" and "Harvest".

6. Discussions and Conclusions

In this paper, we have proposed an MLP-based approach for non-Lambertian shape reconstruction. The key novelty of our method is the neural parameterizations of spatially-varying surface reflectances, and of surface geometry. By leveraging the physical principle of image rendering, we explicitly tackle the reflectance and cast shadows by neural network. Despite being an unsupervised method, our method outperforms existing state-of-the-art supervised methods on real-world datasets. Our method is inspired by NeRF [19], which uses a coordinate-based MLP to model the mapping from 3D coordinates to appearance. In contrast, we factorize the image appearance into multiple components: normal, diffuse albedos, neural specular bases, and shadows. The fitting on these physical-based rendering factors restores the object's surface properties faithfully. Besides, we explicitly parameterize diffuse, specularities, and shadows to ensure the inverse rendering follows a physically meaningful and explainable manner. Our method also relates to [13, 26], which aim at optimizing a CNN-based self-supervised architectures. Our MLP-based framework is significantly faster than those CNN-based methods. We will release the code and models.

Limitations and future work: Our estimation of depth is sensitive to the accuracy of normal estimation and surface discontinuities. Introducing more constraints for accurate depth estimation would certainly help to identify more accurate shadows. Our model may fail in the presence of strong inter-reflections. Finding an efficient model to trace secondary and tertiary rays bouncing between surfaces is also an interesting future direction.

Acknowledgments This research is funded in part by ARC-Discovery grants (DP 190102261 and DP220100800), a gift from Baidu RAL, as well as a Ford Alliance grant to Hongdong Li.

References

[1] Neil Alldrin, Todd Zickler, and David Kriegman. Photometric stereo with non-parametric and spatially-varying reflectance. In *2008 IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–8. IEEE, 2008. 5, 6

[2] Mark Boss, Raphael Braun, Varun Jampani, Jonathan T Barron, Ce Liu, and Hendrik Lensch. Nerd: Neural reflectance decomposition from image collections. *arXiv preprint arXiv:2012.03918*, 2020. 2

[3] Brent Burley and Walt Disney Animation Studios. Physically-based shading at disney. In *ACM SIGGRAPH*, volume 2012, pages 1–7. vol. 2012, 2012. 3

[4] Charles-Félix Chabert, Per Einarsson, Andrew Jones, Bruce Lamond, Wan-Chun Ma, Sebastian Sylwan, Tim Hawkins, and Paul Debevec. Relighting human locomotion with flowed reflectance fields. In *ACM SIGGRAPH 2006 Sketches*, pages 76–es. 2006. 6

[5] Manmohan Chandraker, Sameer Agarwal, and David Kriegman. Shadowcuts: Photometric stereo with shadows. In *2007 IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–8. IEEE, 2007. 1

[6] Guanying Chen, Kai Han, Boxin Shi, Yasuyuki Matsushita, and Kwan-Yee Kenneth Wong. Deep photometric stereo for non-lambertian surfaces. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2020. 1, 2, 3

[7] Guanying Chen, Kai Han, and Kwan-Yee K Wong. Ps-fcn: A flexible learning framework for photometric stereo. In *European Conference on Computer Vision*, pages 3–19. Springer, 2018. 1, 2, 3, 6

[8] Aaron Hertzmann and Steven M Seitz. Example-based photometric stereo: Shape reconstruction with general, varying brdfs. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 27(8):1254–1264, 2005. 4

[9] David Honzátko, Engin Türetken, Pascal Fua, and L Andrea Dunbar. Leveraging spatial and photometric context for calibrated non-lambertian photometric stereo. *arXiv preprint arXiv:2103.12106*, 2021. 2

[10] Zhuo Hui and Aswin C Sankaranarayanan. Shape and spatially-varying reflectance estimation from virtual exemplars. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 39(10):2060–2073, 2017. 3

[11] Satoshi Ikehata. Cnn-ps: Cnn-based photometric stereo for general non-convex surfaces. In *European Conference on Computer Vision*, pages 3–19. Springer, 2018. 1, 2, 6

[12] Satoshi Ikehata, David Wipf, Yasuyuki Matsushita, and Kiyoharu Aizawa. Robust photometric stereo using sparse regression. In *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*, pages 318–325. IEEE, 2012. 1

[13] Berk Kaya, Suryansh Kumar, Carlos Oliveira, Vittorio Ferrari, and Luc Van Gool. Uncalibrated neural inverse rendering for photometric stereo of general surfaces. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3804–3814, 2021. 1, 2, 3, 5, 6, 8

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

[15] Junxuan Li, Antonio Robles-Kelly, Shaodi You, and Yasuyuki Matsushita. Learning to minify photometric stereo. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7568–7576, 2019. 1, 2, 6

[16] Fotios Logothetis, Ignas Budvytis, Roberto Mecca, and Roberto Cipolla. Px-net: Simple and efficient pixel-wise training of photometric stereo networks. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 12757–12766, 2021. 1, 2, 6

[17] Wojciech Matusik, Hanspeter Pfister, Matt Brand, and Leonard McMillan. A data-driven reflectance model. *ACM Transactions on Graphics*, 22(3):759–769, July 2003. 3, 4

[18] Wojciech Matusik, Hanspeter Pfister, Matthew Brand, and Leonard McMillan. Efficient isotropic brdf measurement. 2003. 4

[19] 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, 4, 5, 8

[20] Yasuhiro Mukaigawa, Yasunori Ishii, and Takeshi Shakunaga. Analysis of photometric factors based on photometric linearization. *JOSA A*, 24(10):3326–3334, 2007. 1

[21] Yvain Quéau, Tao Wu, François Lauze, Jean-Denis Durou, and Daniel Cremers. A non-convex variational approach to photometric stereo under inaccurate lighting. In *Computer Vision and Pattern Recognition (CVPR), 2017 IEEE Conference on*, pages 350–359. IEEE, 2017. 1

[22] Szymon M Rusinkiewicz. A new change of variables for efficient brdf representation. In *Eurographics Workshop on Rendering Techniques*, pages 11–22. Springer, 1998. 3

[23] Hiroaki Santo, Masaki Samejima, Yusuke Sugano, Boxin Shi, and Yasuyuki Matsushita. Deep photometric stereo network. In *Computer Vision Workshop (ICCVW), 2017 IEEE International Conference on*, pages 501–509. IEEE, 2017. 1, 2

[24] Boxin Shi, Zhipeng Mo, Zhe Wu, Dinglong Duan, Sai Kit Yeung, and Ping Tan. A benchmark dataset and evaluation for non-lambertian and uncalibrated photometric stereo. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2018. 1, 6

[25] 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

[26] Tatsunori Taniai and Takanori Maehara. Neural inverse rendering for general reflectance photometric stereo. In *International Conference on Machine Learning*, pages 4864–4873, 2018. 1, 2, 3, 5, 6, 8

[27] Kenneth E Torrance and Ephraim M Sparrow. Theory for off-specular reflection from roughened surfaces. *Josa*, 57(9):1105–1114, 1967. 3

[28] Bruce Walter, Stephen R Marschner, Hongsong Li, and Kenneth E Torrance. Microfacet models for refraction through rough surfaces. *Rendering techniques*, 2007:18th, 2007. 3

[29] Xi Wang, Zhenxiong Jian, and Mingjun Ren. Non-lambertian photometric stereo network based on inverse reflectance model with collocated light. *IEEE Transactions on Image Processing*, 29:6032–6042, 2020. 2, 6

[30] Robert J Woodham. Photometric method for determining surface orientation from multiple images. *Optical engineering*, 19(1):191139, 1980. 1, 6

[31] Lun Wu, Arvind Ganesh, Boxin Shi, Yasuyuki Matsushita, Yongtian Wang, and Yi Ma. Robust photometric stereo via low-rank matrix completion and recovery. In *Asian Conference on Computer Vision*, pages 703–717. Springer, 2010. 1

[32] Tai-Pang Wu and Chi-Keung Tang. Photometric stereo via expectation maximization. *IEEE transactions on pattern analysis and machine intelligence*, 32(3):546–560, 2010. 1

[33] Zhuokun Yao, Kun Li, Ying Fu, Haofeng Hu, and Boxin Shi. Gps-net: Graph-based photometric stereo network. *Advances in Neural Information Processing Systems*, 33, 2020. 1, 2, 6

[34] 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

[35] 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. *arXiv preprint arXiv:2106.01970*, 2021. 2

[36] Qian Zheng, Yiming Jia, Boxin Shi, Xudong Jiang, Ling-Yu Duan, and Alex C Kot. Spline-net: Sparse photometric stereo through lighting interpolation and normal estimation networks. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 8549–8558, 2019. 2, 6