

# NPL-MVPS: Neural Point-Light Multi-View Photometric Stereo

Fotios Logothetis  
 Toshiba Europe  
 Cambridge UK

fotios.logothetis@toshiba.eu

Ignas Budvytis  
 Independent researcher  
 Cambridge UK

ignas.budvytis@gmail.com

Roberto Cipolla  
 University of Cambridge  
 Cambridge UK

rc10001@cam.ac.uk

## Abstract

*In this work we present a novel multi-view photometric stereo (MVPS) method. Like many works in 3D reconstruction we are leveraging neural shape representations and learnt renderers. However, our work differs from the state-of-the-art multi-view PS methods such as PS-NeRF [47] or Supernormal [4] in that we explicitly leverage per-pixel intensity renderings rather than relying mainly on estimated normals.*

*We model point light attenuation and explicitly raytrace cast shadows in order to best approximate the incoming radiance for each point. The estimated incoming radiance is used as input to a fully neural material renderer that uses minimal prior assumptions and it is jointly optimised with the surface. Estimated normals and segmentation maps are also incorporated in order to maximise the surface accuracy.*

*Our method is among the first (along with Supernormal [4]) to outperform the classical MVPS approach proposed by the DiLiGenT-MV benchmark and achieves average 0.2mm Chamfer distance for objects imaged at approx 1.5m distance away with approximate  $400 \times 400$  resolution. Moreover, our method shows high robustness to the sparse MVPS setup (6 views, 6 lights) greatly outperforming the SOTA competitor (0.38mm vs 0.61mm), illustrating the importance of neural rendering in multi-view photometric stereo.*

## 1. Introduction

Photometric Stereo (PS) is a long standing and important problem in the field of Computer Vision. While early PS works [15, 17, 31, 33, 46] primarily tackled the estimation of normals from single view images, the value of PS was unlocked by binocular [8, 22, 29, 44] and multi-view [9, 25, 32, 40, 47, 53] stereo setups as it allowed for accurate recovery of shape and not only normals. This, in turn, opened many applications such as general 3D re-

construction, novel-view rendering, relighting and material editing [47], as well as robot interaction, quality control in manufacturing and industrial conveyor belt scanning.

Along with increasing the number of views Photometric Stereo undergone another important change by moving from classical non-linear optimisation enabled inverse graphics approaches (for single view [14], binocular [22], multi-view [25, 32]) to neural network (e.g. [15, 30]) and in particular neural shape representation enabled inverse graphics approaches (for single view [12], binocular [29] and multi-view [4, 47]). However, despite the latter methods, especially [4], achieving impressive accuracy on DiLiGenT-MV [25] benchmark their approach to MVPS is somewhat incomplete as they do not attempt to directly explain (and learn to match) observed pixel-wise intensities. In particular, [4] does not explicitly use image intensities to optimise for shape and is fully reliant on per-view normal maps. Whereas, PS-NeRF [47] only uses average intensity during the surface optimisation stage and thus leaves most of the photometric information unused. It is important to note that Brahimi et. al. [1], attempts to re-render the images however does not model cast shadows and uses the simplified Dinsey BRDF [3] which may not model all materials accurately.

In this work we provide the first neural multi-view photometric stereo approach which fully leverages the availability of pixel intensity information for estimating 3D shape from Photometric Stereo images (see Figure 5). We achieve this by explicitly modeling the incident light from point light sources to leverage intensity based shape optimisation over purely normal driven shape optimisation [4, 47] which is fragile to incorrectly estimated normals especially in cases of very few available lights as shown in Figure 1 and Section 5.

In more detail, we model point light attenuation and explicitly raytrace cast shadows in order to best approximate the incoming radiance for each point. The estimated incoming radiance is used as input to a fully neural material renderer that uses minimal prior assumptions and it is jointly

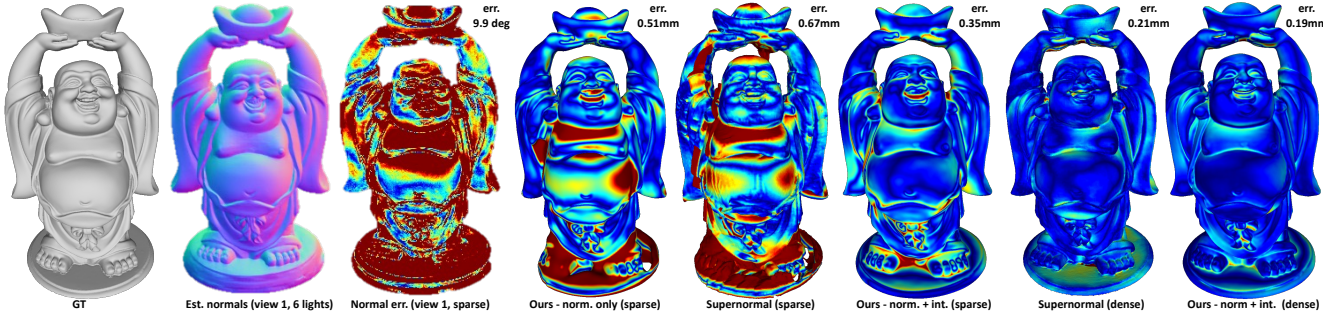


Figure 1. In this figure we demonstrate the fragility of relying mainly on estimated normals (using [13]) for deep learning based *sparse* multi-view photometric stereo when 6 views out of 20 and 6 light sources out of 96 available are used. The first column shows the ground truth of Buddha object from *dense* MVPS DiLiGenT-MV benchmark. The following two columns show the estimated normals and corresponding error maps using only 6 out of 96 lights available (view 1 mean average normal error is  $9.9^\circ$ , saturated red color corresponds to  $5^\circ$  error). Using such normals leads to a large reconstruction error using our method when pixel intensities are not leveraged (0.51mm) and previous SOTA *dense* MVPS method Supernormal [4] (0.67mm). If pixel intensities are used along with estimated normals (column 6) a significantly smaller error of 0.35mm is achieved. The final two columns show the error maps of estimated shapes when all available views and lights are used. In this setting Supernormal [4] achieves a similar reconstruction error as our method (0.21mm vs 0.19mm). Similar dynamics apply to other DiLiGenT-MV objects as shown in Tables 2 and 3, strongly motivating for explicit pixel intensity modeling in MVPS methods. Note here the errors are computed as Chamfer distance while the visualisation only shows errors from reconstruction to ground truth mesh for each reconstructed mesh surface point. Note dark red corresponds to  $\geq 1mm$  error in the shape error illustrations (columns 3-8).

optimised with the surface. Estimated normals and segmentation maps are also incorporated in order to maximise the surface accuracy. This allows us to achieve SOTA reconstruction accuracy (0.2mm) on original *dense* (20 views, 96 lights) DiLiGenT-MV [25] benchmark and significantly outperform (0.38mm vs 0.61mm) the best MVPS method [4] in a *sparse* setup of 6 views and 6 lights.

The remainder of this paper is organised as follows. Section 2 discusses the related work in Photometric Stereo and Multi-View Stereo. It is followed by a description of our method in Section 3. The experimental setup and experiment results are described in Sections 4 and 5 respectively.

## 2. Related Work

There is an extensive literature on single and multi-view photometric stereo and we review the following cases:

**Single view photometric stereo.** The first successful deep learning based single view PS was CNN-PS [15] which was extended by [30] and [31] to be applicable to general calibrated point like configurations. Other works like [9, 11, 18] have used material reflectance priors (using specific BRDFs like Lambertian or Ward) for single view normal prediction. Other recent approaches have leveraged the power of recent transformer models and big synthetic datasets (often of tens of thousand of images) to tackle a weakly uncalibrated setting like [6, 23, 24, 48] and more recently fully uncalibrated single-view PS [16, 17] and [13]. However, despite the success of these methods in producing single view normal maps (as well as material maps), accurate shape reconstruction is still challenging.

**Multi-view photometric stereo.** To overcome the ill-posedness of single view photometric surface reconstruction, multi-view photometric stereo (MVPS) methods have leveraged information from multiple view and multiple lights. Classical optimisation approaches have used triangle meshes [40] or sign distance function based parameterisations [32, 39, 54] to tackle the multi-view PS problem, under diffuse reflectance. Methods, e.g. [25], were also developed for more general materials as well.

**Neural surfaces.** Recently, neural surface approaches have become very popular in tackling the 3D reconstruction problem. Early approaches include NeRF [38] and its first extensions to neural SDF parameterisations [45, 49, 50]. The first methods which used neural SDFs specifically for the multi-view PS problem include [19–21, 52]. However, contrary to the direction of the neural inverse rendering literature, these approaches do not attempt to re-render the original photometric stereo images but rather some 2D derivatives of the images such as normals or albedo maps. For example, Supernormal [4] only renders normal maps but achieves very fast training speed though patch parameterisation, as well as the use of the NERFACC [26] framework. PS-NeRF [47] renders normal and average intensity maps whereas RNb-Neus [2], uses normal and albedo maps to render virtual light images. Thus, all these methods are reliant on single view PS networks and have no way to circumvent noisy estimates that are likely to happen in case of sparse lights and number of views as demonstrated in Figure 1.

Other recent neural rendering approaches have advanced the sophistication of the rendering methods to be more

structured and thus respect the physics of light reflection more, like Ref-NeRF [43], Neuralangelo [27], NERO [28] and NeILF++ [51] but none of these methods has yet to be applied to PS problem, especially lacking the ability to model point light illumination. Finally it is worth mentioning [12] who introduced the idea of a infinitely differentiable surface (SIREN [42]) with Lambertian rendering for the single view PS and [29], which extended this method to the binocular setting and also added a fully learnable general material renderer.

We borrow the material renderer from [29] while extending the approach to work in the multi-view setting using an SDF parametrisation similar to [49, 50]. It is worth noting that Brahimi et. al, [1], also tackles the MVPS problem though physics-based per point rendering (using the Disney BRDF [3]). This approach is the most similar to us with major differences being that we use a fully neural material model, we explicitly ray-trace cast shadows and also employ supervision signal from single view photometric stereo normals. Thus we are able to outperform them with 0.25mm vs 0.34mm error (see Table 2).

### 3. Method

This section describes our method for solving the point light multi-view photometric stereo. A high level overview is also shown in Figure 2. Our method is primarily an inverse neural rendering method. Section 3.1 describes the assumed irradiance model. Sections 3.2 and 3.3 describe the underlying neural surface parametrisation and its initialisation, respectively. Section 3.4 describes training losses used.

#### 3.1. Irradiance equation

We now explain the assumed irradiance equation of a world point  $\mathbf{X}$  with surface normal  $\mathbf{N}$  and albedo  $\rho$ . We assume point light sources  $m$  at positions  $\mathbf{P}_m$  which generate variable lighting vectors  $\mathbf{L}_m = \mathbf{P}_m - \mathbf{X}$ . In addition, point light propagation results to the following attenuation factor  $a_m = \frac{\phi_m}{\|\mathbf{L}_m\|^2}$  where  $\phi_m$  is the intrinsic brightness of the light source. We note that the literature [37] usually also assumes angular dissipation factor but these calibration numbers are unavailable for DiLiGenT-MV [25] therefore we opt for a simpler, perfect point light source model. Thus, the reflected intensity of the point  $\mathbf{X}$  for the  $m$ -th light source  $i_m$  is modelled as:

$$i_m = s_m a_m \rho B(\mathbf{N}, \mathbf{L}_m, \mathbf{V}) \quad (1)$$

Note, here  $B(\cdot)$  is assumed to be a general BRDF,  $s_m \in \{0, 1\}$  is an indicator variable to account for cast shadows that completely block direct reflectance. We assume that indirect reflectance (i.e. self reflections and ambient light) are negligible and can be ignored. Also note that analytic

BRDF models like [3, 7] often (but not always [36]) separate diffuse and specular components and may include separate albedos. However, a completely learned BRDF as proposed in Section 3.2 does not need to follow this structure.

#### 3.2. Neural SDF

**Geometry parameterisation.** Following the work of other neural volumetric approaches [49, 50], we parameterise the scene geometry as the zeroth level set of an implicit function  $F$  corresponding to the Signed Distance Field,  $d = F(\mathbf{X})$ , parameterised by a deep neural network. We note that the SDF of any arbitrary geometry is always continuous, almost everywhere differentiable and satisfies the Eikonal equation of unit magnitude gradient  $\|\nabla F(\mathbf{X})\| = 1$ . Finally, we note that for surface points (where  $F(\mathbf{X}) = 0$ ), the surface normal is the gradient of the SDF, i.e.  $\mathbf{N}(\mathbf{X}) = \nabla F$ . This allows to train the SDF through rendering loss from the initial photometric stereo images. In addition, if surface normal maps are available (e.g. from single view estimation networks) they can also be used as an addition training signal. We use the SIREN architecture [42] which is a MLP with sinusoidal activation functions and that guarantees that the surface is infinitely differentiable thus can be easily recovered from its derivatives.

**Ray sampling.** We follow a volumetric sampling and rendering method similar to VolSDF [49] where the neural SDF is queried in multiple samples on outgoing rays from each image foreground pixel. For each pixel an estimate of the depth is available (initialised from single view PS and occasionally updated during training) and thus most of the samples are concentrated around that depth. However, to allow the surface to evolve and to minimise free space artefacts, additional samples are also sampled in a wider depth range. Following VolSDF [49], we use the Laplace density function to convert from SDF values  $d$  to density  $t$  as follows:

$$t(d) = (0.5 + 0.5 \cdot \text{sign}(d)(\exp(-|\beta d|) - 1)) / \beta \quad (2)$$

with  $\beta$  being a trainable scalar constant controlling the sharpness of the distribution.

Finally, alpha blending is used to accumulate depth, normals and rendered intensity at each ray using the standard approach, i.e. transparency  $\alpha = \exp(-t\delta r)$  with  $\delta r$  being the distance along the ray and so the surface-ray intersection point  $\mathbf{X}_I$  is computed as a weighted sum of ray samples  $\mathbf{X}_I = \sum_i (w_i \mathbf{X}_{r_i})$ , with the sample weights  $w_i$  corresponding to the accumulated opacity. Note that intersection points  $\mathbf{X}_I$  are further used to compute cast shadows but are not directly rendered. Similar averaging is used to obtain the rendered intensities ( $i_I = \sum_i (w_i i_i)$ ) and surface normals ( $\mathbf{N}_I = \sum_i (w_i \mathbf{N}_{r_i})$ ) both of which are used to compute losses. The ray sampling process is further explained in Figure 3.

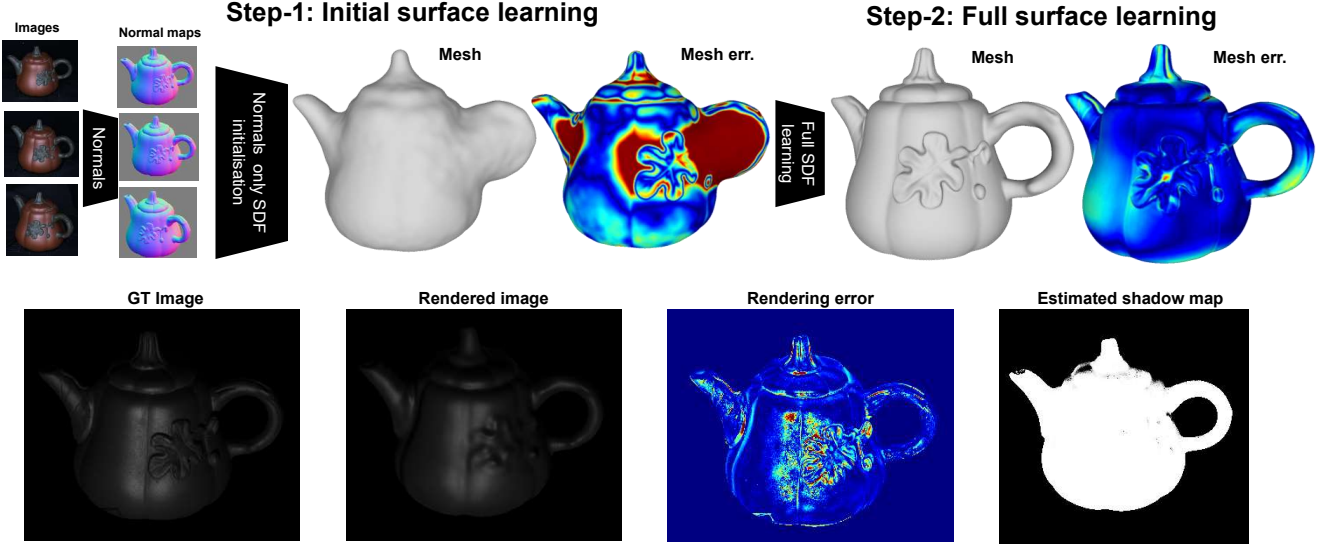


Figure 2. Schematic of our overall method. Single view PS is used to obtain normal maps. Training the SDF with normal and silhouette loss (for 3 epochs only, see Section 3.3) obtains a rough surface which is then refined with full volumetric rendering, explained in Figure 3. The second row also shows the GT and render images (as grayscale), the rendering error (with red  $\geq 0.1$ ) as well as the computed shadow map.

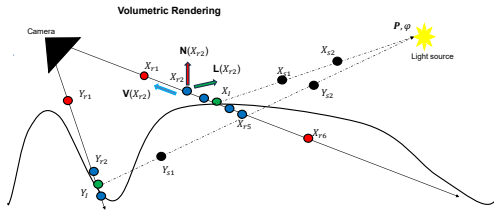


Figure 3. Visualisation of our volume rendering approach. Two rays with multiple ray samples  $\mathbf{X}_{r_i}$ , and  $\mathbf{Y}_{r_i}$  are shown. The surface-ray intersection points  $\mathbf{X}_I$ , and  $\mathbf{Y}_I$  are also shown as they are used to ray trace cast shadows (towards the light source at position  $\mathbf{P}$  with brightness  $\phi$ ). Cast shadow samples are marked as  $\mathbf{X}_{s_i}$ , and  $\mathbf{Y}_{s_i}$  respectively. Note that points that significantly contribute to the total rendering (through the accumulated opacity) are coloured blue and points that do not (because they are outside of the surface or occluded) are marked red. For shadow sample points rendering is not performed and so are marked black. Note that the intersection points ( $\mathbf{X}_I$ , and  $\mathbf{Y}_I$ ) are only used to guide shadows so they are not rendered either. Finally, for the  $\mathbf{X}_{r_2}$  ray sample point, normal  $\mathbf{N}$ , lighting  $\mathbf{L}$  and viewing vectors  $\mathbf{V}$  (that are used for rendering) are shown with respective colors of red, green and blue.

**Learned BRDF.** We follow the approach of [29] where the BRDF is also parameterised as another SIREN network and thus is completely learned from the data. This assumes that the material properties are uniform around the target scene except for a scalar albedo variation. We emphasise that we chose to perform grayscale intensity rendering instead of

full RGB rendering as this is expected to minimise the synthetic to real gap. Real RGB images are usually acquired with demosaicing of single intensity values and this procedure is usually optimised to best recover intensity not colour (e.g see [10] used by OpenCV).

To minimise over-fitting, the material BRDF network receives as input only the relative angles between  $\mathbf{N}$ ,  $\mathbf{L}$  and  $\mathbf{V}$ . In addition, to simplify the learning problem, we follow the principles described in the MERL database [35]. To achieve this, the half vector  $\mathbf{H} = \frac{\mathbf{L} + \mathbf{V}}{|\mathbf{L} + \mathbf{V}|}$  is first computed and the input to the network is the relative angles between  $\mathbf{N}$ ,  $\mathbf{L}$  and  $\mathbf{H}$ . Finally, we note that the final activation of SIREN part of the BRDF network is exponential and there is a post multiplication with an  $\mathbf{N} \cdot \mathbf{L}$  factor so that the BRDF network learns a multiplicative factor over the diffuse reflectance, parameterised as follows:

$$B(\mathbf{N}, \mathbf{L}_m, \mathbf{V}) = (\mathbf{N} \cdot \mathbf{L}_m) \exp(\text{SIREN}(\mathbf{N} \cdot \mathbf{H}_m, \mathbf{N} \cdot \mathbf{L}_m, \mathbf{H}_m \cdot \mathbf{L}_m)) \quad (3)$$

**Albedo.** The scalar albedo  $\rho$  is learned with another SIREN network which is queried for every sample point. We note that having the BRDF network constant throughout the volume and only varying a scalar albedo may sacrifice quality in objects with significantly varied materials, but this does not seem to be the case in the DiLiGenT-MV [25] as shown in Figure 4. We note some competitors like Neuralangelo [27] learn a fully-varied rendering network parameterised by position, normal, lighting and viewing vectors, but this approach is a lot more prone to over-fitting and would struggle to extrapolate the rendering into com-



pletely unseen viewing angles, which is not the case for our approach (see Figure 4).

**Shadow estimation.** To estimate cast shadows for a ray-surface intersection point  $\mathbf{X}_I$ , we raytrace from that point to the light source following the direction of the lighting vectors  $\mathbf{L}_m$  computed above. For each ray we take 16 random samples  $j \in [2\text{mm}, 50\text{mm}]$ . For all these points, we query the SDF network and accumulate opacity following the same volumetric rendering procedure, i.e  $s = \prod_j (1 - \alpha_j)$ . We note that this shadow computation procedure is very computationally expensive therefore it cannot be computed for all the rendered points. Instead, we only compute it for the intersection point of each ray ( $\mathbf{X}_I$  is Figure 3) and assume it is the same for all other ray samples.

Visualisation of the generated renderings of synthetic and real data are shown in Figure 4.

### 3.3. Initialisation

As it is standard in MVPS approaches, e.g. [47], we can use single view PS (at each view) in order to obtain normal and depth estimates. We start by computing per view normal maps using the state-of-the-art PS normal estimation network [13] and also use numerical integration [41] in order to get approximate depth maps. The normal maps are used in order to provide an additional training signal.

The depth maps do need to be accurate as they are only used to initialise the ray sampling space. More specifically, classical ray marching (e.g. NeRF [38]) uses fixed near/far planes for each pixel which is inefficient while newer approaches use an occupancy grid (e.g. NERFACC [26]) to guide the search space. We opt for a simpler solution where for each pixel, near/far planes are centered around the pixels depth estimate; this is initialised with single view depth estimate and updated every 5 epochs.

The SDF network is initialised with weights that approximate the SDF of a perfect sphere. To speed up convergence, we always run 3 epochs with normal and silhouette loss without rendering.

**Final surface calculation.** After the optimisation is completed, the SDF network can be sampled in a regular grid of points and a triangle mesh surface can be recovered using the standard Marching Cubes [34] algorithm. For recovered surface points, the albedo network is queried in order to obtain a textured reconstruction.

### 3.4. Losses

We use the following losses.

**Rendering loss.** We use L1 loss on the rendered intensities, i.e.  $L_{rend} = |i_{rend} - i_{gt}|$ . We note that to better balance the rendering data, each light source is scaled so as the maximum GT intensity is 1 (saturated). This is performed because some of the DiliGenT-MV lights are very dim. The relative weighting of this loss when used is 1000.

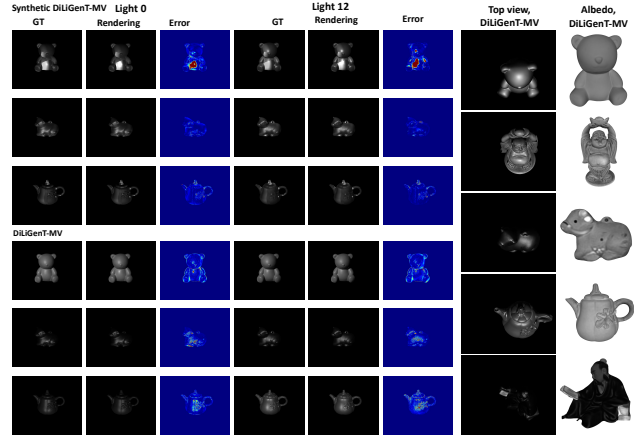


Figure 4. Qualitative visualisation of re-rendering and rendering errors for synthetic and real data (left side, top and bottom 3 rows). The scaling of the error map sets red to  $\geq 0.1$ . We note that most of the error is concentrated on the middle of concavities as self reflection is not modeled. On the right side we see renderings in a novel angle of objects recovered from real data as well as recovered albedo maps.

**Silhouette loss.** Similar to previous approaches [27, 45], we also apply binary cross entropy loss,  $L_{sil} = \text{BCE}(\alpha_{total}, mask)$ , between the predicted silhouettes and the ground truth ones. We generate predicted silhouettes with the total accumulated opacity  $\alpha_{total} = \prod_j (1 - \alpha_j)$  (which should be 1 for foreground and 0 for background points). We consider a 10 pixel band outside the provided segmentation masks for background points. The relative weighting of this loss is 10.

**Normal loss.** We apply angular normal loss to match SDF computed normals  $\mathbf{N}_s$  to the network predicted normals  $\mathbf{N}_n$ . We follow [29, 30] and use angular error (as opposed to L2 loss used in PS-NeRF [47])  $L_n = |\text{atan2}(\|\mathbf{N}_n \times \mathbf{N}_s\|, \mathbf{N}_n \cdot \mathbf{N}_s)|$ . For rays where the accumulated opacity is less than 0.01 (i.e the ray does not intersect any surface), no normal loss is applied. In addition, following previous works, the normal loss is weighted by the obliqueness of each point ( $\mathbf{N} \cdot \mathbf{V}$ ) and that stops the optimisation to try to fit occlusion boundaries which are numerically unstable. The relative weighting of this loss when used is 1.

**Eikonal loss.** To enforce the Eikonal equation for all ray samples, we apply L1 loss which is the standard in most SDF approaches i.e.  $L_{eik} = |||\nabla d|| - 1|$ . The relative weighting of this loss is 10.

**Curvature regulariser.** To minimise floater artefacts (especially on the inside of objects) and encourage the optimisation to recover the minimum surface, it important to include some curvature regulariser encourage smoothness on the volumetric normals (i.e. SDF gradients). Computing exact analytic curvatures (via auto-differentiation) has a high computational cost an is not really required, as the objective

Method	Bear	Buddha	Cow	Pot2	Reading	Avg. SE
GT Normals	0.06	0.08	0.03	0.04	0.02	0.05
N	0.17	0.14	0.10	0.14	0.16	0.14
I	0.13	0.23	0.06	0.08	0.12	0.12
I-S	0.20	0.27	0.04	0.08	0.18	0.15
N + I	0.11	0.15	0.03	0.07	0.11	0.09

Table 1. Ablation study of our method on a synthetic replica of DiLiGenT-MV [25] benchmark. We first compute our method performance using ground truth normals in order to highlight potential issues with real DiLiGenT-MV [25] benchmark (first two rows of Table 2) where recovered shape is significantly less accurate both for our and Supernormal [4] methods (0.05mm vs 0.11mm). We also include comparison of four versions of our method named (N), (I), (I-S) and (N+I). (N) corresponds to only applying normals loss, where (I) and (I-S) correspond to only using rendering loss with and without shadows respectively. (N+I) combines all losses. The combined approach achieves the best error and particularly note that it outperforms both other configurations most objects indicating the the combined approach is better than a simple interpolation between the two.

is to only use them as a regulariser. Instead, inspired by SuperNormal [4], we use the ray samples  $\mathbf{X}_i$  and compute finite differences along the ray as:  $curv(\mathbf{X}_i) \approx \frac{\|\mathbf{N}_i - \mathbf{N}_{i+1}\|}{\|\mathbf{X}_i - \mathbf{X}_{i+1}\|}$ . We note that an exact curvature would require finite differences along all 3-axis, but for regularisation purposes this definition is adequate, and comes with no additional SIREN queries. Note that NeILF++ [51] and Neuralangelo [27] also include a similar regulariser. The relative weighting of this loss is 1.

## 4. Experiment Setup

This section describes the datasets as well as the training and evaluation protocol.

### 4.1. Datasets

**DiLiGenT-MV.** Our main evaluation is performed on DiLiGenT-MV [25] benchmark containing 5 objects with 96 lights in 20 views. Images are of  $612 \times 512$  px resolution with objects actually occupying a maximum of  $400 \times 400$ px. Ground truth meshes, camera intrinsics, extrinsics and normal maps are provided, together with point light positions and far-field light brightnesses ( $\phi$ ). We note that these brightnesses were measured from the intensity of a flat calibration target roughly positioned at the location of the imaged objects, so intrinsic brightness is recovered by multiplying with inverse distance square. We note that as such calibration data is unavailable, the  $\phi$  is expected to be fairly inaccurate and thus it is optimised during training.

**Dataset anomalies.** In order to have the most fair assessment, we report the following dataset anomalies on DiLiGenT-MV and our attempts to overcome them.

Firstly, the provided GT normal maps and masks are incompatible with renderings of ground truth meshes when provided intrinsic and extrinsic parameters are used. Note, some of the provided rotation matrices are not orthonormal and have non-unitary determinant. We follow the approach of RNB [2] and Supernormal [4] by first performing an explicit projection matrix computation ( $P = K[R|T]$ ) followed by QR decomposition (using OpenCV) to obtain orthonormal rotation matrix.

Secondly, the provided segmentation masks in Bear and Cow contain holes that need to be closed manually in order to prevent the silhouette loss from introducing large holes in the reconstructed meshes.

Finally, Ikehata et al [15] first noticed that the first 20 images of the Bear appear to be corrupted. We also found more similarly corrupted images on other views (more visualisations in the supplementary) and did our best effort to manually mark and ignore them however it is possible that more image corruptions are still unnoticed.

**Synthetic DiLiGenT-MV.** To better demonstrate the effectiveness of our method without the real data corruptions discussed above, we rendered a synthetic version of DiLiGenT-MV with Blender. See Figure 4. We use the exact same objects, with the exact same poses and rendered the 96 points lighth. The objects materials were chosen to loosely mimic real objects and the albedo was set to a random pattern. Finally, we note that this synthetic data can be used to visualise shadow and indirect reflection maps which are really hard to correctly evaluate on real data.

### 4.2. Hyperparameters and Training

We use a tensorflow port of SIREN for all the experiments. The SDF MLP is set to  $5 \times 512$  layers (1.05M parameters) and the albedo MLP to  $3 \times 256$  (133K parameters). The BRDF MLP is set to  $3 \times 32$  layers (2.5K parameters). We use 64 ray samples in a 100mm ray range and an additional 64 around the average intersection (in a shrinking distance range up to 10mm) and 64 extra samples computed with one step of Newton method (for approximating the 0 of the SDF). For each shadow ray we used 16 samples. We train with batch size of 512 rays for 100 epochs which takes approx. 20h on a NVIDIA TITAN RTX and 17GB of RAM when rendering all 96 lights. Note that the 6 lights 6 views version completes in only 2h.

### 4.3. Evaluation protocol

We evaluate our method by computed Chamfer distance (marked as surface error SE) of the reconstructions and the ground truth. This is computed as the average of asymmetric Hausdorff distance from reconstruction to ground truth and the opposite, with the distances computed with Meshlab. We note that in order to have a fair comparison and not bias the error with unseen bottom of the objects, the

bottom 6mm of GTs and all reconstructions are removed. All DiLiGenT-MV objects are aligned to be touching the XY plane so the cropping is straightforward. This cropping also avoids large error at some parts of the bottom of the objects that are occluded by the background (e.g. the feet of Reading, see Supplementary Material). Thus our reported error (in Table 2) is generally *lower* than the numbers originally reported in other works.

**Competing approaches.** We compare against DiLiGenT-MV [25], PS-NeRF [47], MVAS [5] Brahimi et. al, [1] RNB [2] and Supernormal [4]. DiLiGenT-MV [25] and Brahimi et. al, [1] are closed source so we use the meshes computed by the original authors; for all other methods we use the meshes reproduced from their original codes. We note that Supernormal [4] offers the best performance and by far the least computational cost so it is used for ablating the sparse lights and views scenario as well. Also Brahimi et. al, [1] was only computed on the sparse scenario therefore that is the only available comparison for them.

## 5. Experiments

We describe two sets of experiments on synthetic data and real data in Sections 5.1 and 5.2 correspondingly.

### 5.1. Synthetic data

As mentioned in Section 4.1 the original DiLiGenT-MV [25] dataset contains several anomalies making hard to correctly ablate the several steps of our method. Thus additional ablation experiments are performed on Synthetic-DiLiGenT-MV dataset.

We first show that our network can achieve a very low error of 0.05mm when using the ground truth normals which is not the case for the real DiLiGenT-MV dataset as shown in Table 2. We also show that if predicted normals are used the performance is worse: 0.14mm. Using the rendering of intensities only achieves overall error of 0.12mm. In addition, using the rendering loss on intensities only (I) significantly outperforms the normals loss only (N) experiments on all objects except Buddha. The reason for such a differing performance is that the presence of strong self reflection effects presents a more difficult task to the rendering network than for the normal estimation network. Note that not computing the shadow maps while using rendering only loss (I-S) increases the error from 0.12mm to 0.15mm, with the very concave object Reading being affected the most (0.12m to 0.18mm). The combined normals and intensity rendering experiment achieves the best accuracy of 0.09mm average error.

### 5.2. Real data

In this section we report our results on DiLiGenT-MV [25] benchmark in both original *dense* setup (20 views, 96 lights) and various *sparse* setups.

Method	Bear	Buddha	Cow	Pot2	Reading	Avg.
SpN [4] GT N	0.16	0.12	0.06	0.10	0.13	0.11
Ours GT N	0.13	0.15	0.10	0.12	0.06	0.11
DiLiGenT-MV [25]	0.22	0.33	0.08	0.21	0.25	0.22
PS-NeRF [47]	0.27	0.33	0.27	0.26	0.36	0.30
MVAS [5]	0.25	0.37	0.21	0.20	0.52	0.31
SpN [4]	0.19	0.21	0.21	0.14	0.22	<b>0.20</b>
RNB [2]	0.25	0.21	0.31	0.18	0.27	0.24
Ours N	0.29	0.19	0.17	0.19	0.29	0.22
Ours I	0.25	0.24	0.18	0.34	0.26	0.26
Ours I-S	0.34	0.33	0.25	0.32	0.30	0.31
Ours N + I	0.21	0.19	0.17	0.20	0.22	<b>0.20</b>

Table 2. Results on original *dense* DiLiGenT-MV [25] benchmark. For all objects we report the Chamfer distance error as well as average error on all objects. Normals are computed using the universal PS method of [13]. We evaluate four versions of our method, including using only normal (N) loss, rendering loss with (I) and without shadows (I-S) as in Table 1. We also include comparison with Supernormal (SpN [4]) with ground truth normals (GT N).

Method	Bear	Buddha	Cow	Pot2	Reading	Avg.
SpN [4] [6,6]	0.48	0.67	0.27	0.39	1.21	0.61
Ours N + I [6,6]	0.32	0.35	0.28	0.33	0.63	<b>0.38</b>
Brah. et al [1] [6, 75]	0.38	0.32	0.24	0.29	0.47	0.34
SpN [4] [6, 30]	0.29	0.25	0.17	0.17	0.75	0.33
Ours N + I [6, 30]	0.28	0.29	0.16	0.25	0.25	<b>0.25</b>
SpN [4] [20, 6]	0.28	0.53	0.28	0.30	0.36	0.35
Ours N + I [20, 6]	0.25	0.38	0.24	0.52	0.33	<b>0.34</b>

Table 3. Results on *sparse* DiLiGenT-MV [25] benchmark. We include three sparse cases (marked with [views, lights]) and compare with Supernormal [4] which is the best performing dense competitor. We also include the comparison with Brahimi et al [1] which was computed with 6 views and 75 lights and is comparable to our 6 views 30 lights case. We note that our method significantly outperform Supernormal [4] (0.38mm vs 0.61mm) on the most sparse setup [6 views, 6 lights] as well as Brahimi et. al [1] (0.25mm vs 0.34mm) in [6 views, 30 lights] setup. Supernormal [4] matches our performance when more views are used.

**Original (*dense*) DiLiGenT-MV setup.** Our method achieves (0.2mm) state of the art results (see Table 2), only matching to the original DiLiGenT-MV benchmark (0.22mm) and Supernormal [4] (0.2mm). Note we significantly outperform most deep learning based competitors (ie. PS-NeRF [47], MVAS [5] and RNB [2]). Also note in PS-NeRF [47] the performance of various methods was compared by including the bottom (invisible) part of the object which gave misleading results of the deep learning method outperforming the classical one proposed originally with the DiLiGenT-MV [25] benchmark. We believe that the classical method performs so well as the objects have relatively simple geometry (especially the Cow where is



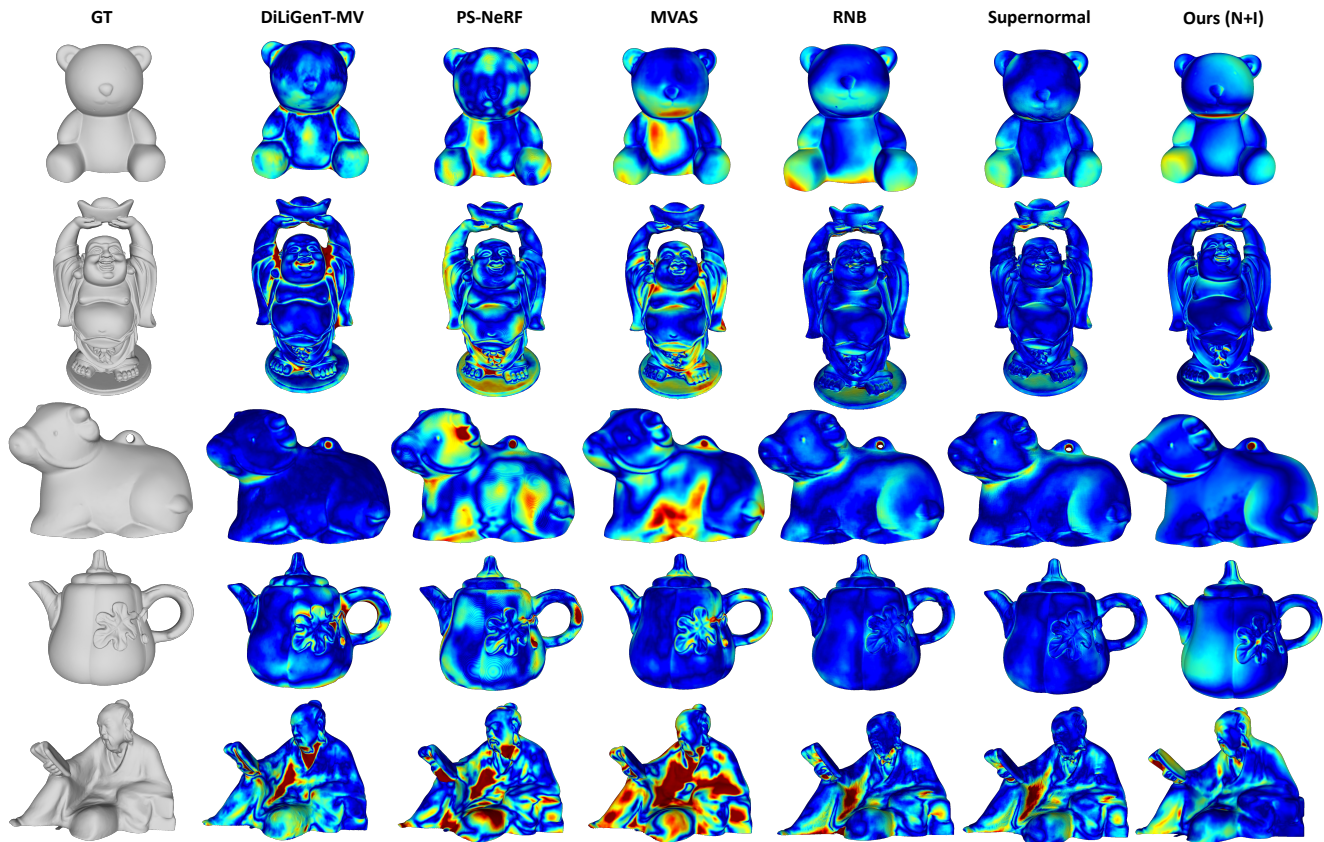


Figure 5. Qualitative results on real DiLiGenT-MV [25] benchmark. For each mesh vertex, the minimum distance to the GT mesh is shown with the error bars set to red corresponding to 1mm. We note that we are achieving consistent, uniform accuracy on all regions of all objects, including the concavity in the middle of Reading.

mostly convex and has the least amount of shadows and self reflection). The reconstructed meshes are shown in Figure 5.

**Sparse DiLiGenT-MV setup.** In Table 3 provide our results on various *sparse* setups of DiLiGenT-MV [25] benchmark. In particular, we include three sparse cases (marked with [views, lights]). In all cases we compare with Supernormal [4] which is the best competitor on *dense* DiLiGenT-MV benchmark and has code available. We also include the comparison with Brahimi et al [1] (code is not available) which was originally computed with 6 views and 75 lights which is comparable to our 6 views 30 lights case. Our method significantly outperforms Supernormal [4] (0.38mm vs 0.61mm) on the most sparse setup [6 views, 6 lights] where the normal estimates are very inaccurate (see Figure 1) and significantly outperforms Brahimi et. al [1] (0.25mm vs 0.34mm) despite having a significantly smaller number of lights in [6 views, 30 lights] setup. Supernormal [4] matches (0.34 mm vs 0.35mm) our performance when more views (20) are used. Visualisation of sparse results can be found in the supplementary material.

It is also noteworthy that only using 6 lights and 6 views (around 2% of the total data) only increases the total error from 0.2mm to 0.38mm signifying the need for a more challenging multi-view photometric stereo benchmark.

## 6. Conclusions

In this work we proposed a novel multi-view photometric stereo method. Unlike most MVPS methods our approach explicitly leverages per-pixel intensity renderings rather than relying mainly on estimated normals. We believe such approach is required for truly applicable and robust MVPS as the estimated normals are likely to fail on complex materials or geometries. We clearly demonstrate the benefit of leveraging intensities on a synthetic and real DiLiGenT-MV benchmark and the applicability of our method on the minimal 6 lights case.

Finally, it is important to note that improving computational efficiency has been beyond the scope of this project, however future work can improve it significantly by following strategies proposed by Supernormal [4] and integrating with the NERFACC [26] framework.



## References

- [1] Mohammed Brahimi, Bjoern Haefner, Zhenzhang Ye, Bastian Goldluecke, and Daniel Cremers. Sparse views near light: A practical paradigm for uncalibrated point-light photometric stereo. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 11862–11872, June 2024. [1](#), [3](#), [7](#), [8](#)
- [2] Baptiste Brument, Robin Bruneau, Yvain Quéau, Jean Mélou, François Lauze, Jean-Denis Durou, and Lilian Calvet. Rnb-neus: Reflectance and normal-based multi-view 3d reconstruction. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024. [2](#), [6](#), [7](#)
- [3] B. Burley. Physically-based shading at disney. In *SIGGRAPH Course Notes*, 2012. [1](#), [3](#)
- [4] Xu Cao and Takafumi Taketomi. Supernormal: Neural surface reconstruction via multi-view normal integration. *CVPR*, 2024. [1](#), [2](#), [6](#), [7](#), [8](#)
- [5] Okura F. Cao X., Santo H. and Matsushita Y. Multi-view azimuth stereo via tangent space consistency. In *Proc. of Computer Vision and Pattern Recognition (CVPR)*, 2023. [7](#)
- [6] Guanying Chen, Kai Han, Boxin Shi, Yasuyuki Matsushita, and Kwan-Yee K. Wong. Sdps-net: Self-calibrating deep photometric stereo networks. In *CVPR*, 2019. [2](#)
- [7] R. L. Cook and K. E. Torrance. A reflectance model for computer graphics. *ACM Trans. Graph.*, 1(1):7–24, jan 1982. [3](#)
- [8] Hao Du, Dan B. Goldman, and Steven M. Seitz. Binocular photometric stereo. In *BMVC*, 2011. [1](#)
- [9] Carlos Hernández Esteban, George Vogiatzis, and Roberto Cipolla. Multiview photometric stereo. *PAMI*, 2008. [1](#), [2](#)
- [10] Pascal Getreuer. Malvar-he-cutler linear image demosaicing. *Image Processing on Line*, 1:83–89, 2011. [4](#)
- [11] Dan B. Goldman, Brian Curless, Aaron Hertzmann, and Steven M. Seitz. Shape and spatially-varying brdfs from photometric stereo. *PAMI*, 2010. [2](#)
- [12] Heng Guo, Hiroaki Santo, Boxin Shi, and Yasuyuki Matsushita. Edge-preserving near-light photometric stereo with neural surfaces. *arXiv*, 2022. [1](#), [3](#)
- [13] Clément Hardy, Yvain Quéau, and David Tschumperlé. Uni MS-PS: a Multi-Scale Encoder Decoder Transformer for Universal Photometric Stereo. working paper or preprint, Feb. 2024. [2](#), [5](#), [7](#)
- [14] Zhuo Hui and Aswin C. Sankaranarayanan. Shape and spatially-varying reflectance estimation from virtual exemplars. *IEEE Trans. Pattern Analysis and Machine Intelligence (PAMI)*, 39(10):2060–2073, 2017. [1](#)
- [15] S. Ikehata. Cnn-ps: Cnn-based photometric stereo for general non-convex surfaces. In *ECCV*, 2018. [1](#), [2](#), [6](#)
- [16] Satoshi Ikehata. Universal photometric stereo network using global lighting contexts. *CVPR*, 2022. [2](#)
- [17] Satoshi Ikehata. Scalable, detailed and mask-free universal photometric stereo. *CVPR*, 2023. [1](#), [2](#)
- [18] Yakun Ju, Cong Zhang, Songsong Huang, Yuan Rao, and Kin-Man Lam. Learning deep photometric stereo network with reflectance priors. In *ICME*, 2023. [2](#)
- [19] Berk Kaya, Suryansh Kumar, Carlos Oliveira, Vittorio Ferrari, and Luc Van Gool. Uncertainty-aware deep multi-view photometric stereo. In *CVPR*, 2022. [2](#)
- [20] Berk Kaya, Suryansh Kumar, Carlos Oliveira, Vittorio Ferrari, and Luc Van Gool. Multi-view photometric stereo revisited. In *WACV*, 2023. [2](#)
- [21] Berk Kaya, Suryansh Kumar, Francesco Sarno, Vittorio Ferrari, and Luc Van Gool. Neural radiance fields approach to deep multi-view photometric stereo. In *WACV*, 2021. [2](#)
- [22] Hui Kong, Pengfei Xu, and Eam Khwang Teoh. Binocular uncalibrated photometric stereo. In *ISCV*, 2006. [1](#)
- [23] Junxuan Li and Hongdong Li. Neural reflectance for shape recovery with shadow handling. In *CVPR*, pages 16221–16230, 2022. [2](#)
- [24] Junxuan Li and Hongdong Li. Self-calibrating photometric stereo by neural inverse rendering. In *ECCV*. Springer, 2022. [2](#)
- [25] 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 Trans. Image Process.*, 2020. [1](#), [2](#), [3](#), [4](#), [6](#), [7](#), [8](#)
- [26] Ruilong Li, Hang Gao, Matthew Tancik, and Angjoo Kanazawa. Nerfacc: Efficient sampling accelerates nerfs. *arXiv*, 2023. [2](#), [5](#), [8](#)
- [27] Zhaoshuo Li, Thomas Müller, Alex Evans, Russell H Taylor, Mathias Unberath, Ming-Yu Liu, and Chen-Hsuan Lin. Neuralangelo: High-fidelity neural surface reconstruction. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023. [3](#), [4](#), [5](#), [6](#)
- [28] Yuan Liu, Peng Wang, Cheng Lin, Xiaoxiao Long, Jiepeng Wang, Lingjie Liu, Taku Komura, and Wenping Wang. Nero: Neural geometry and brdf reconstruction of reflective objects from multiview images. In *SIGGRAPH*, 2023. [3](#)
- [29] Fotios Logothetis, Ignas Budvytis, and Roberto Cipolla. A neural height-map approach for the binocular photometric stereo problem. *WACV*, 2024. [1](#), [3](#), [4](#), [5](#)
- [30] Fotios Logothetis, Ignas Budvytis, Roberto Mecca, and Roberto Cipolla. PX-NET: Simple, Efficient Pixel-Wise Training of Photometric Stereo Networks. In *ICCV*, 2021. [1](#), [2](#), [5](#)
- [31] Fotios Logothetis, Roberto Mecca, Ignas Budvytis, and Roberto Cipolla. A cnn based approach for the point-light photometric stereo problem. *IJCV*, 2022. [1](#), [2](#)
- [32] Fotios Logothetis, Roberto Mecca, and Roberto Cipolla. A differential volumetric approach to multi-view photometric stereo. In *ICCV*, 2019. [1](#), [2](#)
- [33] Fotios Logothetis, Roberto Mecca, Yvain Quéau, and Roberto Cipolla. Near-field photometric stereo in ambient light. In *BMVC*, 2016. [1](#)
- [34] William E Lorensen and Harvey E Cline. Marching cubes: A high resolution 3d surface construction algorithm. In *Seminal graphics: pioneering efforts that shaped the field*, pages 347–353. 1998. [5](#)
- [35] W. Matusik, H. Pfister, M. Brand, and L. McMillan. A data-driven reflectance model. *ACM TOG*, 2003. [4](#)
- [36] Roberto Mecca, Yvain Quéau, Fotios Logothetis, and Roberto Cipolla. A single-lobe photometric stereo approach for heterogeneous material. *SIAM Journal on Imaging Sciences*, 2016. [3](#)

- [37] Roberto Mecca, A. Wetzler, A. Bruckstein, and R. Kimmel. Near Field Photometric Stereo with Point Light Sources. *SIAM Journal on Imaging Sciences*, 2014. 3
- [38] 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. 2, 5
- [39] Matthias Nießner, Michael Zollhöfer, Shahram Izadi, and Marc Stamminger. Real-time 3d reconstruction at scale using voxel hashing. *ACM Trans. Graph.*, 2013. 2
- [40] 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, 2017. 1, 2
- [41] Y. Quéau and J.-D. Durou. Edge-preserving integration of a normal field: Weighted least squares, TV and L1 approaches. In *SSVM*, 2015. 5
- [42] Vincent Sitzmann, Julien N.P. Martel, Alexander W. Bergman, David B. Lindell, and Gordon Wetzstein. Implicit neural representations with periodic activation functions. In *NeurIPS*, 2020. 3
- [43] Dor Verbin, Peter Hedman, Ben Mildenhall, Todd Zickler, Jonathan T. Barron, and Pratul P. Srinivasan. Ref-nerf: Structured view-dependent appearance for neural radiance fields. *CVPR*, 2022. 3
- [44] Chaoyang Wang, Lijuan Wang, Yasuyuki Matsushita, Bojun Huang, Magnetro Chen, and Frank K. Soong. Binocular photometric stereo acquisition and reconstruction for 3d talking head applications. In *INTERSPEECH*, 2013. 1
- [45] Peng Wang, Lingjie Liu, Yuan Liu, Christian Theobalt, Taku Komura, and Wenping Wang. Neus: Learning neural implicit surfaces by volume rendering for multi-view reconstruction. *NeurIPS*, 2021. 2, 5
- [46] Robert J. Woodham. Determining surface curvature with photometric stereo. In *ICRA*, 1989. 1
- [47] Wenqi Yang, Guanying Chen, Chaofeng Chen, Zhenfang Chen, and Kwan-Yee K. Wong. Ps-nerf: Neural inverse rendering for multi-view photometric stereo. In *ECCV*, 2022. 1, 2, 5, 7
- [48] Wenqi Yang, Guanying Chen, Chaofeng Chen, Zhenfang Chen, and Kwan-Yee K. Wong. S3-NeRF: Neural reflectance field from shading and shadow under a single viewpoint. In Alice H. Oh, Alekh Agarwal, Danielle Belgrave, and Kyunghyun Cho, editors, *NeurIPS*, 2022. 2
- [49] Lior Yariv, Jiatao Gu, Yoni Kasten, and Yaron Lipman. Volume rendering of neural implicit surfaces. In *Thirty-Fifth Conference on Neural Information Processing Systems*, 2021. 2, 3
- [50] 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
- [51] Jingyang Zhang, Yao Yao, Shiwei Li, Jingbo Liu, Tian Fang, David McKinnon, Yanghai Tsin, and Long Quan. Neif++: Inter-reflectable light fields for geometry and material estimation. In *International Conference on Computer Vision (ICCV)*, 2023. 3, 6
- [52] Dongxu Zhao, Daniel Lichy, Pierre-Nicolas Perrin, Jan-Michael Frahm, and Soumyadip Sengupta. Mvpsnet: Fast generalizable multi-view photometric stereo. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 12525–12536, October 2023. 2
- [53] Zhenglong Zhou, Zhe Wu, and Ping Tan. Multi-view photometric stereo with spatially varying isotropic materials. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2013. 1
- [54] Michael Zollhöfer, Angela Dai, Matthias Innmann, Chenglei Wu, Marc Stamminger, Christian Theobalt, and Matthias Nießner. Shading-based refinement on volumetric signed distance functions. *ACM Trans. Graph.*, 2015. 2