Deep Photometric Stereo Networks for Determining Surface Normal and Reflectances

Hiroaki Santo, Masaki Samejima, Yusuke Sugano, Boxin Shi, and Yasuyuki Matsushita

Abstract—This paper presents a photometric stereo method based on deep learning. One of the major difficulties in photometric stereo is designing an appropriate reflectance model that is both capable of representing real-world reflectances and computationally tractable for deriving surface normal. Unlike previous photometric stereo methods that rely on a simplified parametric image formation model, such as the Lambert's model, the proposed method aims at establishing a flexible mapping between complex reflectance observations and surface normal using a deep neural network. In addition, the proposed method predicts the reflectance, which allows us to understand surface materials and to render the scene under arbitrary lighting conditions. As a result, we propose a deep photometric stereo network (DPSN) that takes reflectance observations under varying light directions and infers the surface normal and reflectance in a per-pixel manner. To make the DPSN applicable to real-world scenes, a dataset of measured BRDFs (MERL BRDF dataset) has been used for training the network. Evaluation using simulation and real-world scenes shows the effectiveness of the proposed approach in estimating both surface normal and reflectances.

Index Terms—Photometric stereo, surface normal, bidirectional reflectance distribution functions (BRDFs), deep learning

1 Introduction

Photometric stereo estimates surface normal of a scene from a set of measurements that are collected under different light conditions. The basic idea of photometric stereo was introduced in 1980's by Woodham [1] and Silver [2] based on the Lambertian [3] reflectance assumption. To make photometric stereo better applicable to real-world objects, it is of interest to use a more flexible reflectance model, for which in a general form it is represented by bidirectional reflectance distribution functions (BRDFs). In addition to determining surface normal, it is desired to estimate the reflectance function and its associated parameters rather than assuming that they are known a priori.

While an image formation model with a general reflectance representation based on BRDFs has great flexibility and descriptive power, it is difficult to directly work with general non-parametric BRDFs in the context of photometric stereo. This is due to that the photometric stereo is an inverse problem, in which surface normal is determined based on image measurements as input. The image formation is governed by reflectance functions, and it is desirable that the reflectance function is represented by a simple model so that the inverse estimation problem becomes tractable while retaining the representation power. Along with this direction, there have been studies to use parametric representations that are more flexible than the Lambert's model, *e.g.*, Torrance-Sparrow model [4], [5], microfacet-based model [6], and bi-polynomial model [7], to better approximate complex real-world reflectances. However, so far, parametric models have been only accurate for a limited class of materials, and the solution methods suffer from unstable non-convex optimization for estimating reflectance parameters, which prohibits obtaining accurate surface normal. Indeed, estimation of reflectance is an important task in photometric stereo because it is tightly coupled with the estimation of surface normal. In addition, correctly estimated reflectance provides important information for understanding the materials of the scene, and opens up a capability of re-rendering the scene under arbitrary light conditions that are unseen. Thus, it is needed a photometric stereo method that can recover surface normal of a scene with diverse reflectances as well as estimating the spatially-varying reflectances of the scene.

To achieve this goal, we propose an end-to-end learning approach to photometric stereo using a deep neural network (DNN). The proposed method, which we call a deep photometric stereo network (DPSN), uses a DNN for establishing a flexible mapping from observations to surface normal and reflectances. To make DPSN applicable to diverse real-world materials, a dataset of measured BRDFs (MERL BRDF dataset [8]) has been used for training the network. It allows the method to accurately estimate surface normal of a scene with complex reflectances and also to recover the reflectance in a per-pixel manner. We use a flexible expression of reflectances compared to the existing parametric models, which is represented as a linear combination of bases derived from the MERL BRDFs, allowing us to recover a full BRDF table for each pixel. Further, to deal with the non-local cast shadowing effect, we propose a shadow layer that is based on a conventional dropout strategy.

In this paper, we assume that the light directions are predefined and remain the same between training and prediction phases, which is the case in many photometric stereo apparatuses. DPSN operates in a per-pixel manner, by taking intensity observations of a surface point under varying light directions, and infers the surface normal and reflectance of the surface point. The result shows the effectiveness of the proposed method on both

- *H. Santo, M. Samejima, and Y. Matsushita are with Graduate School of Information Science and Technology, Osaka University, Osaka, Japan. E-mail: {santo.hiroaki, samejima, yasumat}@ist.osaka-u.ac.jp*
- *Y. Sugano is with Institute of Industrial Science, The University of Tokyo, Japan. E-mail: sugano@iis.u-tokyo.ac.jp*
- *B. Shi is with the National Engineering Laboratory for Video Technology, Department of Computer Science and Technology and Institute for Artificial Intelligence, Peking University, Beijing, China. E-mail: shiboxin@pku.edu.cn*

Manuscript received September 2, 2018; revised May 12, 2020.

simulation and real-world images for estimating surface normal and reflectances.

The preliminary version of this work appeared in [9], and we extend our previous work to achieve the simultaneous estimation of surface normal and reflectances while the previous method was limited to the surface normal estimation. To enable this new capability of reflectance estimation, we propose a new branch in the network trained by a rendering loss function.

2 Related work

In this section, we first describe previous works of photometric stereo. After that, we describe the reflectance modeling and methods for simultaneous estimation of both surface normal and reflectances based on physics-based vision. Finally, we describe the DNN-based approaches in photometric stereo and discuss their distinctions from this work.

2.1 Photometric stereo

Conventional photometric stereo [1], [2] is based on the Lambert's reflectance model. Because the Lambert's model is an ideal reflectance model that may not well represent real-world reflectances, extending photometric stereo to work with non-Lambertian surfaces has been of interest for its practical use. Existing studies on non-Lambertian photometric stereo can be classified into three categories.

The first category includes methods based on robust estimation, where the non-Lambertian reflectances are treated as outliers. They assume that the majority of reflectance observations obeys, or is close to, the Lambert's model so that the non-Lambertian reflectances, such as specular reflections, can be regarded as anomalies. Wu *et al*. [10] formulate the robust estimation problem as a rank minimization problem. They exploit the fact that the Lambertian observations form a low-rank subspace [11] and treat the non-Lambertian reflectances as sparse outliers. Mukaigawa *et al*. [12] use the random sample consensus (RANSAC) scheme [13], which essentially approximates the ℓ_0 residual minimization for discarding outliers. Other robust estimation methods, such as expectation maximization [14], taking the median values [15], ℓ_1 residual minimization and sparse Bayesian learning [16], are also shown to be effective for dealing with sparse outliers. Since the robust estimation methods are built upon statistical outlier rejection, they generally require many input images, *e.g*., 40 images in [10], recorded under distinct light directions.

The second category includes methods based on more sophisticated reflectance models than the Lambertian model to better approximate non-Lambertian reflectance observations. Georghiades [17] uses the Torrance-Sparrow model [18], and Ruiters *et al*. [19] use the Cook-Torrance model [20] along this direction. More recently, Shi *et al*. [21] propose a bi-polynomial BRDF model, which is capable of representing low-frequency non-Lambertian reflectances, and it shows greater accuracy in surface normal estimation. Holroyd *et al*. [22] propose another approach for generalizing reflectance properties based on the reflective symmetry of the halfway vector across the normal-tangent and normal-binormal planes, which does not require estimating a surface reflectance model, and performs well on anisotropic reflectance surfaces. All the above methods estimate the parameters of the reflectance model simultaneously with estimating the surface normal.

The third category includes example-based methods, which determines surface normal by the use of reference objects. Hertzmann and Seitz [23] propose an example-based method using a reference sphere that has the same reflectance as the target object. From the observations that are consistent between the target and reference objects, their method determines the surface normal of the target object by simply mapping the corresponding one from the reference object. The example-based method naturally avoids solving a complex optimization problem, but it requires a reference object, of which the shape is known and reflectance is the same as the target object. More recently, Hui *et al*. [24] proposed an example-based photometric stereo without any reference objects. They obtain a mapping from measurements to surface normal by rendering virtual spheres using given light conditions and various BRDFs instead of recording a reference sphere.

Our method is rooted somewhere between the second and third categories. As with the methods in the second category, our method is able to deal with diverse BRDFs. Instead of estimating both BRDF parameters and surface normal like in the previous approaches, our method directly establishes mappings from reflectance observations to surface normal using a deep learning framework. Our DPSN is trained using a dataset of measured BRDFs of various materials; therefore, it shares the spirit of the example-based methods in the third category, while DPSN does not require a reference object to be placed together with the target object.

2.2 Reflectance estimation

The reflectance estimation is one of the important problems in physically-based computer vision with a wide range of applications in computer graphics. Previously, Tominaga *et al*. [25] presented a reflectance estimation method based on the Phong model [26] from a single RGB image of a cylindrical object taken under a point light source. They use the known 3D shape information to estimate the parameters of the Phong model. Unlike their method, Sato *et al*. [27] developed a method for estimating parametric reflectances from a general shape object using the shape information obtained by laser scanning.

While these methods assume a given shape information for reflectances estimation, Goldman *et al*. [28] proposed the simultaneous estimation of shape and reflectance from multiple images taken under different illuminations. Their method models reflectances by an isotropic parametric reflectance model, namely the Ward model [29], and estimates its parameters via iterative alternating optimization. Wang *et al*. [30] also proposed an estimation method for surface reflectances and a classification method for the surface materials using multiple images taken under varying light conditions. To achieve the segmentation based on materials, they classify materials using support vector machines (SVMs) based on the observations under varying lightings.

Unlike these methods, Wang *et al*. [31] used light field images for the reflectance estimation. They formulate the reflectance estimation task as material classification using DNN, which takes as input a 4D light-filed image and outputs the score prediction for 12 classes of materials. Zhou *et al*. [32] also proposed to use multi-view images taken under varying light directions for shape and reflectance estimation. They used a flexible representation of reflectance using a data-driven BRDF model rather than a parametric reflectance model. The data-driven BRDF model uses discrete look-up tables indexed by light and viewing directions

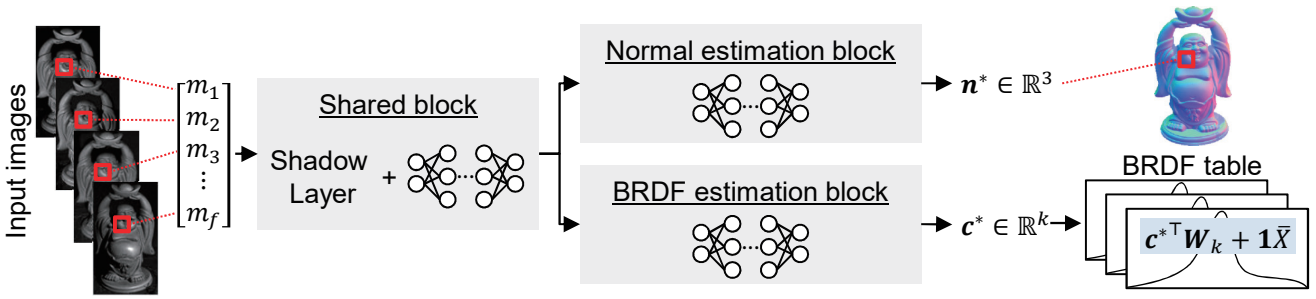


Fig. 1. Overview of the proposed deep photometric stereo network (DPSN). It consists of three components: shared, normal estimation, and BRDF estimation blocks. The shared block is formed by a shadow layer and dense layers. The shared block takes as input the per-pixel measurement vector $\mathbf{m} = [m_1, \ldots, m_f]^\top$. The normal estimation and BRDF estimation blocks consist of dense layers, which yield prediction of a surface normal vector $\mathbf{n}^*$ and reflectance parameters $\mathbf{c}^*$, respectively. From the reflectance parameters $\mathbf{c}^*$, a full BRDF table is reconstructed based on Eq. (2).

sampled by a certain interval. It has an excellent representation capability of reflectances, but it comes with a difficulty for estimation because of the large number of parameters. To make the estimation tractable, it has been shown a linear basis representation of BRDFs where reflectances are represented by a linear combination of a few basis vectors. This representation has been employed in [32]. Nielsen *et al*. [33] proposed a method for BRDF estimation from a few observations, which also used a linear basis representation of BRDFs. They use a log-relative mapping for reflectances to deal with the high dynamic range of reflectances. Xu *et al*. [34] also worked on the BRDF estimation from a few observations (as few as two images) using the same BRDF representation as [33]. Similarly, our method also represents reflectance by a linear combination of basis BRDFs. Unlike these methods, our DNN-based approach allows the use of many principal components without restriction and achieves a flexible representation of BRDFs.

2.3 Deep learning-based approaches for normal and reflectance predictions

Early works that use neural networks in the context of photometric stereo can be found in the 1990's. Iwahori *et al*. [35] used a neural network for determining surface normal. While effective, their method requires the pre-training with a reference sphere painted by the same material as the target object, akin to an example-based method. There are other methods that use neural networks afterwards, but they are restricted to Lambertian reflectances [36], [37] or specialized imaging setup [38]. In contrast to these early works, our work [9] is the first attempt to use a modern DNN architecture in the context of photometric stereo.

More recently, convolutional neural network (CNN) is shown to be effective for photometric stereo [39], [40]. Chen *et al*.'s method [39], called PS-FCN, extracts features from each input image and the corresponding light direction vector, and estimates a surface normal map from the feature map aggregated by max-pooling. With the feature map aggregation by pooling, PS-FCN can deal with an arbitrary number of photometric stereo images. Ikehata [40] proposed another approach, called CNN-PS, to allow the network to deal with an arbitrary number of inputs. It uses an intermediate representation of observed images, called observation map, to represent per-pixel observations under varying lightings. While these methods [39], [40] consider only the surface normal prediction, the proposed method aims to estimate both surface normal and reflectances. Unlike the supervised approaches [9], [39], [40], Taniai and Maehara [41] proposed an unsupervised learning approach based on a reconstruction loss, which is defined by the predicted surface normal, reflectances, and known light sources. Their method is able to accurately estimate surface normal and a slice of BRDF without requiring pre-training. While our method requires supervised training, it can estimate full BRDFs, not a slice of them, in the form of a linear combination of BRDF bases, which allows us to re-render the target scene under a new unseen illumination condition.

Along with the recent development of DNNs, methods for estimating surface normal and reflectance from a single RGB image have been proposed. Rematas *et al*. [42] proposed a convolutional neural architecture to estimate reflectance maps of specular materials in natural lighting conditions from a single RGB image. They treat the lighting conditions to be unknown and estimate a reflectance map, which is a slice of a BRDF, and surface normal simultaneously. Janner *et al*. [43] introduced an image decomposition method that factors a single input image into the reflectance, shape, and lighting condition by a convolutional encoder-decoder architecture. Deschaintre *et al*. [44] and Li *et al*. [45] also proposed DNN-based estimation methods for determining reflectances from a single image. They focus on the specular reflection on a near-planar surface and estimate the surface normal, diffuse albedo, and roughness using an encoder-decoder architecture. Their methods use the rendering loss based on the estimated parameters for training. Since these methods take only a single image as input, the accuracy of the surface normal prediction is rather limited. In addition, while they only consider the reflectance under a certain illumination condition, our method can estimate a full BRDF in a per-pixel manner.

In addition to an RGB image, depth images are useful for the purpose of reflectance estimation. Kim *et al*. [46] proposed a reflectance estimation method from RGBD images using DNN. They use the isotropic Ward BRDF model [29] for reflectances, which is characterized by three parameters. Unlike their method, we use the data-driven BRDF representation like [33], [34], which has a greater expression power.

3 Preliminaries

When a Lambertian surface with albedo-scaled surface normal $\mathbf{n} \in \mathbb{R}^3$ is illuminated by a directional light[1] $\mathbf{l} \in S^2 \subset \mathbb{R}^3$, the measurement $m \in \mathbb{R}_+$ can be described as

$$m = \mathbf{l}^\top \mathbf{n}.$$

1. Throughout this paper, we assume $\|\mathbf{l}\|_2 = 1$, and the input image intensities are normalized by the corresponding light intensity.

For a vector of measurements $\mathbf{m} \in \mathbb{R}_+^f$ observed under f distinct light directions, the above equation can be written with a light matrix $\mathbf{L} = [\mathbf{l}_1, \ldots, \mathbf{l}_f] \in \mathbb{R}^{3\times f}$ as

$$\mathbf{m} = \mathbf{L}^\top \mathbf{n}.$$

The conventional photometric stereo method [1], [2] determines surface normal $\mathbf{n}$ using the above image formation model by

$$\mathbf{n}^* = \mathbf{L}^{-\top}\mathbf{m},$$

when $f = 3$ and $\mathrm{rank}(\mathbf{L}) = 3$, or, with more than three distinct observations, a least-squares approximate solution $\mathbf{n}^*$ can be obtained by a pseudo-inverse of $\mathbf{L}$ as

$$\mathbf{n}^* = \left(\mathbf{L}\mathbf{L}^\top\right)^{-1}\mathbf{L}\mathbf{m}.$$

Unfortunately, a pure Lambertian surface rarely exists in the real world; hence, making photometric stereo work with non-Lambertian surfaces is one of the major interests in practice.

With a general BRDF function, the appearance of a surface under a local illumination model can be described more flexibly. The appearances of a surface observed from a fixed viewing direction $\mathbf{v}$ under varying distant light directions $\mathbf{L}$ can be written as

$$\mathbf{m} = \mathbf{b} \circ (\mathbf{L}^\top \mathbf{n}),$$

where $\mathbf{b} \in \mathbb{R}_+^f$ is a vector of reflectances sampled from the BRDF function ρ as $\mathbf{b} = \rho(\mathbf{L}, \mathbf{n}, \mathbf{v})$, the operator $\circ$ represents element-wise multiplication, and $\mathbf{L}^\top\mathbf{n}$ is the irradiance at the surface under the corresponding light directions.

The above equation assumes a shadow-free world, while in the real-world the surface patches facing away from the lighting direction are in attached shadow, and the light path being occluded causes cast shadow. Such a shadowing process can be written as

$$\mathbf{m} = \mathbf{s} \circ \left[\mathbf{b} \circ \max(\mathbf{L}^\top\mathbf{n}, \mathbf{0})\right], \tag{1}$$

where $\mathbf{s} \in \{0, 1\}^f$ is a boolean vector with 0 indicating observations in cast shadows and 1 otherwise. The effect of attached shadow is accounted by the element-wise $\max$ operator.

4 Proposed Method

The proposed DPSN is a differentiable multi-layer neural network, which learns a mapping from a measurement vector $\mathbf{m}$ obtained under different light directions to the surface normal $\mathbf{n}$ and reflectance. It operates in a per-pixel manner for both training and prediction. As stated in the introduction, we assume that the light directions $\mathbf{L}$ are known and consistent between the training and prediction phases. Our method uses simulated observations as training data that are generated using diverse surface normals rendered with the MERL BRDF dataset [8] that contains BRDFs of 100 different real-world materials. In what follows, we explain the representation of reflectance, the structure of the proposed network, and training and prediction procedures.

4.1 Reflectance representation

A non-parametric representation of BRDFs has high accuracy and expressiveness, while it comes with the cost of high storage requirement. In the context of photometric stereo, where we wish to determine the reflectance for the purpose of estimating surface normal, the direct use of a non-parametric BRDF is computationally intractable due to the large number of unknowns (*e.g.*, if using a MERL BRDF sampling rate, it becomes $90\times90\times180 = 1458000$ for each color channel).

We follow the procedure of linear analysis in Matusik *et al*. [8] to extract the basis from the MERL BRDF dataset so that the BRDFs can be expressed by the small number of coefficients associated with the basis vectors. The MERL BRDF dataset [8] stores BRDF tables for 100 different materials in 3 color channels. We treat each color channel independently and construct a BRDF matrix $\mathbf{X} \in \mathbb{R}_+^{300\times1458000}$ (100 materials in 3 color channels with 1458000 bins) by stacking each BRDF table as a row vector. Let $\bar{X} \in \mathbb{R}_+$ donate the mean of all the elements in $\mathbf{X}$, and $\tilde{\mathbf{X}}$ be $\mathbf{X} - \mathbf{1}\bar{X}$ computed by subtracting the mean $\bar{X}$ from $\mathbf{X}$ for each element, where $\mathbf{1}$ is an all-ones matrix. We compute a singular value decomposition (SVD) of $\tilde{\mathbf{X}}$ and obtain $\mathbf{U}$, $\mathbf{\Sigma}$, and $\mathbf{V}$, where $\mathbf{U}$ and $\mathbf{V}$ are left and right orthonormal singular vectors, and $\mathbf{\Sigma}$ is a diagonal matrix with singular values. As with a standard low-rank approximation, by using the top k singular vectors, $\mathbf{U}_k$ and $\mathbf{V}_k$, and corresponding singular values $\mathbf{\Sigma}_k$, $\tilde{\mathbf{X}}$ is approximately reconstructed as

$$\tilde{\mathbf{X}} \simeq \mathbf{U}_k\mathbf{\Sigma}_k\mathbf{V}_k^\top.$$

To make it easy for a deep neural network to learn, we apply normalization by the standard deviation σ of elements in $\mathbf{U}_k$ and regard $\frac{\mathbf{U}_k}{\sigma}$ as coefficient vectors and $\mathbf{W}_k = \sigma\mathbf{\Sigma}_k\mathbf{V}_k^\top$ as the basis matrix of the BRDF tables. With the basis representation, we can express a BRDF table $\mathbf{b}_t \in \mathbb{R}^{1458000}$ as a linear combination of the basis $\mathbf{W}_k$ with k-dimensional coefficients $\mathbf{c} \in \mathbb{R}^k$ as

$$\mathbf{b}_t^\top = \mathbf{c}^\top\mathbf{W}_k + \mathbf{1}\bar{X}, \tag{2}$$

where $\mathbf{1}$ is a vector whose every element is one. The proposed network estimates the coefficients $\mathbf{c}$ from the measurement vector $\mathbf{m}$. The number of singular vectors k is one of the hyper-parameters in our method and is set to 300, *i.e.*, we use all the singular vectors without compression, because the dimensionality reduction did not exhibit advantage in the estimation via DNN.

Once we obtain the full BRDF table $\mathbf{b}_t$, given a set of light directions $\{\mathbf{l}\}$ forming a light matrix $\mathbf{L}$, and a fixed surface normal $\mathbf{n}$ and viewing direction $\mathbf{v}$, we can sample BRDFs to generate a compact BRDF vector $\mathbf{b} \in \mathbb{R}_+^f$ for the measurement setting from a full BRDF table $\mathbf{b}_t$ as

$$\mathbf{b} = \rho(\mathbf{L}, \mathbf{n}, \mathbf{v}; \mathbf{b}_t), \tag{3}$$

using the function ρ that subsamples elements from $\mathbf{b}_t$. The sampled BRDF $\mathbf{b}$ can then be used as a part of the image formation model expressed in Eq. (1).

To deal with the high dynamic range of reflectances, previous methods [33], [34] used a log-relative mapping for reflectance $\mathbf{X}$. In our case, the log-scaling requires the exponential function to recover the original values when assessing the reconstruction loss. We have observed that it causes a negative effect on convergence in our preliminary experiments; therefore, we decided not to use it. Instead, we effectively neglect specular spikes that shows significantly larger values using ℓ_1 residual loss (more discussion in Sec. 4.2).

4.2 Network Architecture

The proposed DPSN learns a mapping from a measurement vector $\mathbf{m} \in \mathbb{R}_+^f$ at a pixel to the corresponding surface normal $\mathbf{n} \in \mathbb{R}^3$ and BRDF parameters $\mathbf{c} \in \mathbb{R}^k$ of the pixel using a fully connected

deep neural network. Figure 1 shows the overview of the proposed network architecture. DPSN consists of three blocks: Shared, surface normal estimation, and BRDF estimation blocks. The shared block takes as input a measurement vector $\mathbf{m}$, in which each element corresponds to an observation under a certain light direction, and outputs the extracted feature vectors. The normal estimation and BRDF estimation blocks take the feature vectors as input and output the prediction of the surface normal $\mathbf{n}$ and BRDF parameters $\mathbf{c}$, respectively. The measurement vector $\mathbf{m}$ is linearly normalized by a global scaling so that the maximum measurement value in the all f images becomes 1.0.

One of the major challenges in photometric stereo is cast shadow. Differently from attached shadow, cast shadow is caused by a global illumination effect, which cannot be modeled by a local illumination model regardless of the representation ability of a BRDF model. To simulate the cast shadow effect in the training phase, we introduce a *shadow layer* (Fig. 2) that is based on a variant of the dropout scheme [47], which randomly drops units from the network during training (or could be used for testing as well) to prevent learned weights from excessive adaptation. Our shadow layer applies dropout to input nodes to randomly drop a part of the input measurement vector, namely setting the selected measurements to 0, so that the dropped nodes can be regarded as shadowed observations. By training the network with the shadow layer, the proposed DPSN effectively learns mapping from observations to surface normal and BRDF parameters with accounting for cast shadow.

While, in conventional dropout, output from the dropout layer is scaled by $1/(1-r)$ with a dropout rate $r \in [0.0, 1.0)$ to avoid shrinkage of the output magnitude, our shadow layer does not apply the scaling but simply sets the selected elements of the measurement vector to 0 to mimic the shadowing effect. The dropout parameter r corresponds to the ratio of shadowed observations in our context. Obviously, the parameter r depends on the object shape and the light distribution, which is inaccessible in general; therefore, we use varying values of r for training. Specifically, we fluctuate the dropout rate by sampling from a binomial distribution $r \sim B(f, p)$, where the probability of each observation being shadowed p is set to $p = 0.05$.

The DPSN structure is summarized in Table 1. The shared block consists of three layers: One shadow layer and two dense layers. The normal estimation and BRDF estimation blocks respectively consist of three and four dense layers. The number of output nodes is three for the normal estimation block, and k for the BRDF estimation block. All the dense layers use ReLU and dropout during the training. We make the BRDF estimation block deeper than the normal estimation block because the number of output nodes is significantly larger (k is set to 300 in our experiment) in the BRDF estimation block. The existing methods for a similar task, such as shape estimation from a single image [42], [43], [44], [45], use an encoder-decoder architecture with CNNs rather than fully-connected layers to extract features from neighboring pixels. GeoNet [48] also proposed a CNN-based network architecture for joint estimation of depth and surface normal from a single image. Compared to these approaches, our task is better conditioned, allowing the choice of a simpler architecture of the network. Indeed, its computation cost is much smaller than that of CNNs. Generally, the disadvantage of fully-connected layers is the greater number of parameters, but in our case, the number of parameters is less than 4.3M (in the shared and surface normal estimation blocks) + 3.5M (in the BRDF estimation block) $\approx$ 8M, which is regarded a small model size compared to the modern CNN architectures such as AlexNet, VGG-16 (used in GeoNet as backbone), and ResNet-50 [49]. In addition, the per-pixel estimation is useful for scenes with spatially-varying BRDFs, and applicable to arbitrary resolution of the input images, without overly smoothing the surface normal estimates.

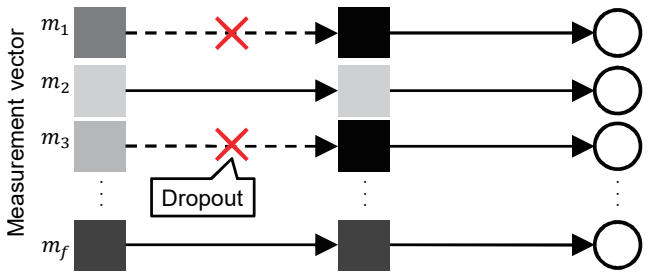


Fig. 2. Overview of shadow layer. Shadow layer randomly drops some of the measurement vector elements to simulate the cast shadow effect. In this illustration, m_1 and m_3 are dropped and corresponding values of the input vector are set to 0.

The DPSN is trained with the following loss function $\mathcal{L}$:

$$\mathcal{L} = (1-\alpha)\mathcal{L}_{\mathbf{n}} + \alpha\mathcal{L}_{\text{BRDF}}, \tag{4}$$

where

$$\begin{cases} \mathcal{L}_{\mathbf{n}} &= ||\mathbf{n} - \mathbf{n}^*||_2^2, \\ \mathcal{L}_{\text{BRDF}} &= ||\mathbf{m} - \mathbf{m}^*||_1, \\ \mathbf{m}^* &= \mathbf{s} \circ \left[\rho(\mathbf{L}, \mathbf{n}, \mathbf{v}; \mathbf{c}^{*\top}\mathbf{W}_k + \mathbf{1}\bar{X}) \circ \max(\mathbf{L}^\top \mathbf{n}, 0)\right]. \end{cases}$$

$\mathbf{n}$ is the ground truth surface normal vector, $\mathbf{n}^*$ is the predicted normal vector, $\mathbf{c}^*$ is the predicted BRDF parameters by the network, and α is a constant weight for balancing surface normal loss $\mathcal{L}_{\mathbf{n}}$ and reconstruction loss$\mathcal{L}_{\text{BRDF}}$. $\mathbf{m}^*$ is the reconstructed measurement vector with the predicted BRDF tables based on Eqs. (1), (2), and (3). The light directions $\mathbf{L}$ are predefined, and the viewing direction $\mathbf{v}$ is set to $[0, 0, 1]^\top$. The rendering loss has been used in previous methods [44], [45]. These methods render the target scene under new light directions in addition to the light direction of the input to constrain BRDFs. Unlike these methods, since we have the photometric stereo images taken under multiple light conditions, we simply render using these light directions to ensure the consistent appearance. We also tested the log-scaled loss for the rendering loss $\mathcal{L}_{\text{BRDF}}$ in Eq. (4) and compared the estimation accuracy of BRDFs. However, it did not exhibit improvement in either quantitative or qualitative evaluations. This is because we use ℓ_1 residual in the rendering loss, with which large but sparse reflectances are naturally neglected, and as a result, the log-scaled rendering loss became ineffective.

4.3 Training

The training set for DPSN consists of pairs of an observation vector and the corresponding surface normal, *i.e.*, $\{(\mathbf{m}, \mathbf{n})\}$. Instead of collecting real-world observations, we generate the observation vectors $\{\mathbf{m}\}$ using the MERL BRDF dataset [8] and a diverse set of surface normal $\{\mathbf{n}\}$ rendered under pre-defined light directions $\mathbf{L}$. The MERL BRDF dataset consists of measured BRDFs of 100 different materials, and we form reflectance vectors $\{\mathbf{b}\}$ in Eq. (1) from the set of surface normal $\{\mathbf{n}\}$ and light directions $\mathbf{L}$.

TABLE 1
The DPSN structure. The number in the parentheses represents the number of nodes in each layer. The shadow layer is used for training, but not for prediction. Dropout rates for training the dense layers are all set to 0.5. The input and output dimensions of the shadow layer is $96\ (=f)$ in our setting.

Layer		
	[Shared block]	
1	Shadow Layer	
2	Dense-(1024), ReLU, Dropout	
3	Dense-(1024), ReLU, Dropout	
	[Normal estimation block]	**[BRDF estimation block]**
1	Dense-(1024), ReLU, Dropout	Dense-(1024), ReLU, Dropout
2	Dense-(1024), ReLU, Dropout	Dense-(1024), ReLU, Dropout
3	Dense-(3)	Dense-(1024), ReLU, Dropout
4		Dense-(k)

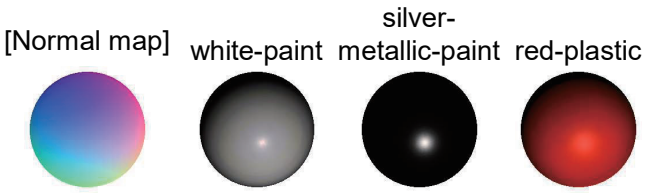


Fig. 3. Examples of rendered images and the corresponding normal map that are used for training. "white-paint", "silver-metallic-paint", and "red-plastic" are the material names in the MERL BRDF dataset [8]. Here, three are shown out of 100 different materials. As seen in the figures, the rendered images contain specularity and attached shadows.

While any distributions of surface normal can be used for generating the surface normal set $\{\mathbf{n}\}$, for this work, we used a simple sphere, which includes all surface normal directions on the scene surface. A sphere is rendered with MERL BRDFs under light directions $\mathbf{L}$, and observations at each pixel location of the rendered images for each color channel form a measurement vector $\mathbf{m}$. Figure 3 shows some of the images of training data generated under certain light directions. As shown in the figure, the rendered images contain complex reflectances that do not obey a simple parametric model.

In the loss function for training, the weight parameter α in Eq. (4) is set to 0.1 for our experiments based on the result of the preliminary experiments. The rendering loss $\mathcal{L}_{\text{BRDF}}$ uses an ℓ_1 loss defined in the rendered image intensity space to neglect outliers. The loss function $\mathcal{L}$ is minimized using Adam [50] with a learning rate set to 5×10^{-5} and other parameters to the suggested default settings ($\beta_1 = 0.9$ and $\beta_2 = 0.999$). The learning rate is decreased with 0.96 every 1000 steps.

After the above training, we perform fine-turning with the following cosine similarity loss (denoted as FT: fine-tuning loss):

$$\mathcal{L}_{\text{FT}} = 1 - \frac{\mathbf{n}^\top \mathbf{n}^*}{||\mathbf{n}||_2 ||\mathbf{n}^*||_2}, \tag{5}$$

to directly optimize based on the angular errors of surface normal. In this step, only the parameters in the normal estimation layers are updated while the parameters in the other branches are fixed. The loss function $\mathcal{L}_{\text{FT}}$ is minimized in the same setting with $\mathcal{L}$, but the learning rate decay is not used in this phase.

In the experiments, we trained the model with the loss function $\mathcal{L}$ for $30,000$ steps with the batch size set to 200 and α to 0.1. After that, it is further trained with $\mathcal{L}_{\text{FT}}$ for $5,000$ steps with the batch size set to $1,000$. We choose the model for the evaluation, which had the best performance in the validation data.

4.4 Prediction

In the prediction phase, given a set of observations under different light directions $\mathbf{L}$, DPSN estimates surface normal $\mathbf{n}$ and BRDF coefficients $\mathbf{c}$ in a per-pixel fashion. In a similar manner to the training phase, color channels are treated independently.

In the normal estimation, for an RGB image, DPSN estimates three surface normals per-pixel and consolidates them to obtain the final estimate. Namely, $\mathbf{n}_r$, $\mathbf{n}_g$, and $\mathbf{n}_b$ are the surface normal estimates from the RGB color channels that are independently estimated, we take the mean vector of the normalized surface normal estimates as

$$\bar{\mathbf{n}} = \frac{1}{3}\left(\frac{\mathbf{n}_r}{||\mathbf{n}_r||_2} + \frac{\mathbf{n}_g}{||\mathbf{n}_g||_2} + \frac{\mathbf{n}_b}{||\mathbf{n}_b||_2}\right).$$

Finally, the merged surface normal $\bar{\mathbf{n}}$ is further normalized to obtain the final surface normal estimate $\mathbf{n}^*$ as $\mathbf{n}^* = \bar{\mathbf{n}}/||\bar{\mathbf{n}}||_2$.

For BRDF estimation, the DSPN outputs k dimensional coefficient vector $\mathbf{c}$ per pixel per color channel, and we obtain the full BRDF table in a per-pixel manner for each color channel based on Eq. (2).

5 Experiments

We evaluate the proposed method using both simulation and real-world datasets. We will explain the training data and implementation details for the experiments before discussing the result.

Training data and implementation details: For training, we generate a synthetic dataset rendered under pre-defined light directions for each of 100 BRDFs from the MERL dataset. The pre-defined light directions are the same as the 96 light directions defined in the DiLiGenT dataset [51]. As a result, we render $100 \times 96 \times 3$ (RGB) $= 28,800$ grayscale images in total. The sphere scene consists of $31,400$ valid pixels with a 100-pixel radius, resulting in $31,400$ distinct surface normals $\mathbf{n}$. Each surface normal is associated with 100×3 measurement vectors $\{\mathbf{m}\}$ with the length of 96.

The DPSN is implemented using TensorFlow[2]. The training takes approximately five hours using two NVIDIA Tesla P100 GPUs. The training data generation can be performed concurrently with the training, *i.e.*, we generate a subset of data for creating a mini-batch, and during the training using the mini-batch, we prepare for the next subset of the training data. We set the batch size to 200, where one training data consists of a single measurement vector $\mathbf{m}$ and the surface normal $\mathbf{n}$. One iteration of training over a batch takes approximately 0.5 seconds. For the prediction, it takes about four seconds for an image with 612×512 pixels (same as the DiLiGenT setting) under 96 lights. Compared to CNN-PS [40], our method can predict surface normals and BRDFs about four times faster thanks to the simple network architecture.

Baseline methods: We compare the proposed DPSN with Lambertian photometric stereo based on conventional ℓ_2 residual minimization (L2) [1] and that with ℓ_1 residual minimization (L1) [16]. For these methods, a surface normal $\mathbf{n}^*$ for each pixel is computed by

$$\mathbf{n}^* = \begin{cases} \underset{\mathbf{n}}{\text{argmin}}\ ||\mathbf{m} - \mathbf{L}^\top \mathbf{n}||_2^2, \\ \underset{\mathbf{n}}{\text{argmin}}\ ||\mathbf{m} - \mathbf{L}^\top \mathbf{n}||_1, \end{cases}$$

2. TensorFlow: https://www.tensorflow.org

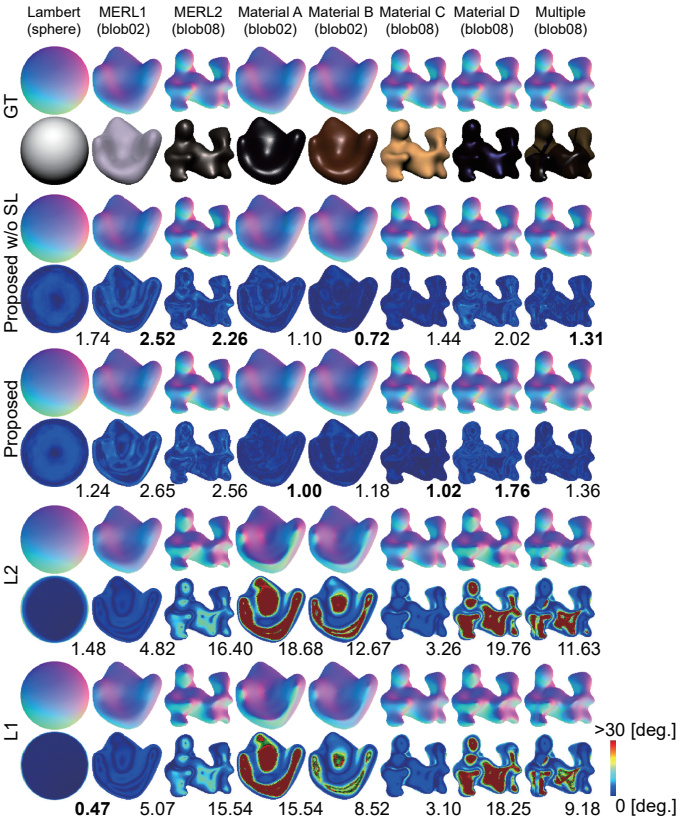


Fig. 4. Experimental result of synthetic scenes. In each row, a normal map is shown on top of the corresponding error map. The numbers represent Mean Angular Error (MAngE) in degree. In the top row, GT means the ground truth, the images below the normal maps are examples of observation images. On the top, material and shape names are shown. MERL1 and MERL2 correspond to "black-oxidized-steel" and "white-fabric2", respectively. Material A to D are the synthetic BRDFs that are created by linearly combining a pair of BRDFs in MERL BRDF dataset: ("blue-fabric", "silver-metallic-paint"), ("cherry-235", "natural-209"), ("beige-fabric", "yellow-paint"), and ("black-soft-plastic", "blue-metallic-paint"), respectively.

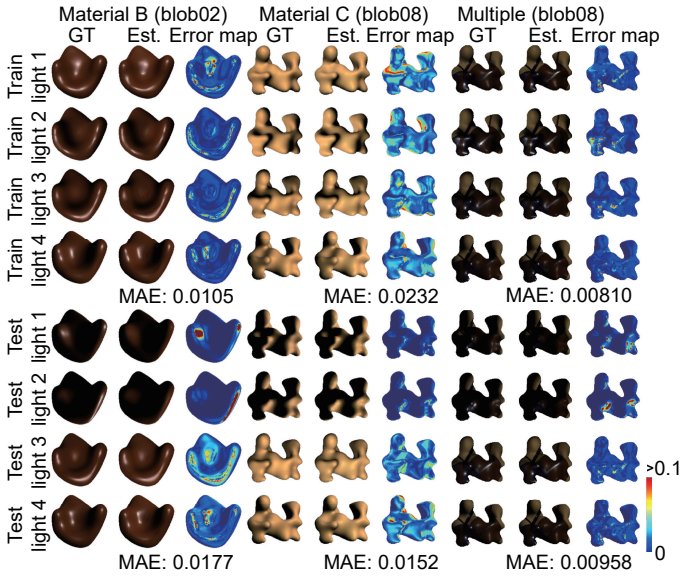


Fig. 5. Rendered results with the estimated BRDFs and the ground truth normal maps for synthetic scenes. We show the images rendered under eight different light conditions. "Train light 1" to "Train light 4" are selected from the 96 light conditions that are used for training, while "Test light 1" to "Test light 4" are from the test light conditions that are not included in the training data. We sampled 50 test light directions uniformly on a hemisphere. In "Multiple (blob08)", we apply Gamma correction with $\gamma = 0.8$ for visualization. "GT" and "Est." mean the ground truth and estimated respectively and are rendered in the same manner to the generation of training data. MAE is the mean absolute error calculated in the grayscale space with the intensities scaled in the range of $[0.0, 1.0]$. MAE for train light conditions is the mean of f $(= 96)$ images, and for test light conditions is the mean of 50 images.

from observations $\mathbf{m}$ and known lighting directions $\mathbf{L}$, respectively. For real-world scenes, we also assess the performance of our method using the DiLiGenT benchmark that covers the state-of-the-art methods of non-Lambertian photometric stereo. To see the effect of shadow layer, we compare DPSN without a shadow layer (denoted as "Proposed w/o SL") and with a shadow layer ("Proposed").

5.1 Evaluation using synthetic dataset

To assess the performance for unseen BRDFs, we split the MERL BRDF dataset into a training set and a testing set. MERL BRDF dataset includes 100 BRDFs and we randomly pick up 20 BRDFs as testing BRDFs. For test shapes, we use three objects, `sphere` and two shapes (`blob02` and `blob08`) from Blobby shape dataset [52]. We use the same light directions as the training for generating the input data.

In addition to the testing set of BRDFs, we also evaluate using synthesized unseen BRDFs. We synthesized new BRDFs by applying nonlinear transformation over a linearly combined BRDFs that are sampled from the MERL BRDF dataset. Specifically, a new BRDF $\tilde{\rho}(\mathbf{L}, \mathbf{n}, \mathbf{v})$ is generated as:

$$\tilde{\rho}(\mathbf{L}, \mathbf{n}, \mathbf{v}) = (\beta\rho_1(\mathbf{L}, \mathbf{n}, \mathbf{v}) + (1 - \beta)\rho_2(\mathbf{L}, \mathbf{n}, \mathbf{v}))^\gamma , \qquad (6)$$

where ρ_1 and ρ_2 are BRDFs sampled from the MERL BRDF dataset, and β and γ are constant parameters. In this experiment, ρ_1 and ρ_2 are selected randomly from the dataset, and the parameters are set to $\beta = 0.5$ and $\gamma = 0.8$. Furthermore, we used the object consisting of multiple materials (denoted as "Multiple" as the material name). We set the area of each materials randomly with combining lines and circles so that each material occupies a certain area.

The surface normal prediction results are summarized in Fig. 4. From testing BRDF sets, we show the result of "black-oxidized-steel" (MERL1) and "white-fabric2" (MERL2), which are the names in MERL BRDF dataset. "Lambert" indicates the Lambertian reflectance, for which the reflectance function ρ is a constant. "Material A" to "Material D" are the synthetic BRDFs. As the mean angular errors in the figure indicate, our method consistently yields accurate estimates of surface normal across different BRDFs. Since the generation of synthetic scenes neglects rendering of cast shadows, our method without a shadow layer ("Proposed w/o SL") shows superior performance to the one with the shadow layer ("Proposed") in some scenes.

Figure 5 shows the rendered images with the predicted BRDFs with the ground truth surface normal ("Rendered" columns) together with the ground truth appearance ("GT" columns). Here, we show the results for "Material B", "Material C", and "Multiple". For both GT and Rendered, we did not render cast shadows. Along the rows, "Train light 1" to "Train light 4" indicate the light directions that are used during the training, and "Test light 1" to "Test light 4" are the new unseen light directions that are not included in the training phase. For assessing the mean

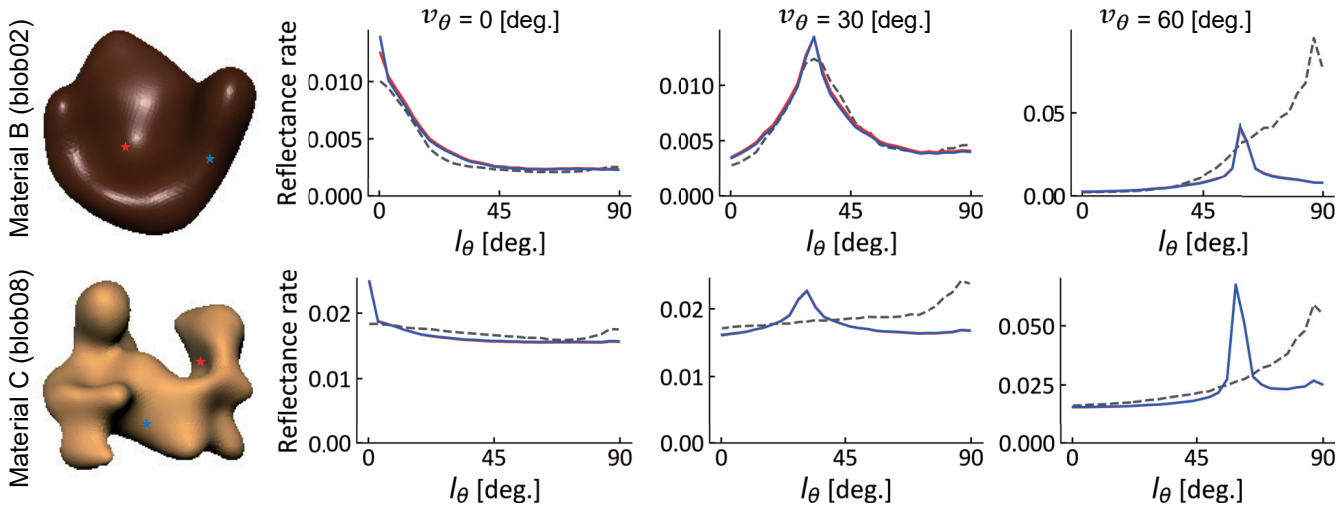


Fig. 6. Plots of estimated BRDFs. The plots show the reflectance $\rho(\mathbf{n}, \mathbf{l}, \mathbf{v})$ varying with the lighting direction $\mathbf{l}$. We use the polar coordinate system for the lighting direction $\mathbf{l}$ and viewing direction $\mathbf{v}$ as $[l_\theta, l_\phi]$ and $[v_\theta, v_\phi]$, where θ and ϕ represent the polar and azimuthal angles, respectively. We fix the normal direction to $\mathbf{n} = [0, 0, 1]^\top$, $v_\phi = 0°$, and $l_\phi = 180°$ and vary l_θ from $0°$ to $90°$. In each row, from left to right, it shows a scene image, and reflectance plots with three viewing directions $v_\theta = \{0°, 30°, 60°\}$. The dashed lines are the ground truth, and red and blue lines are the estimated BRDFs at pixels marked by red and blue stars in the leftmost figure. We show the grayscale reflectance by taking the average of three color channels. Note that we cannot predict the absolute values of reflectances from images, and there is a scaling ambiguity between the ground truth and estimated BRDFs. We align them by normalization based on the median of each BRDF table.

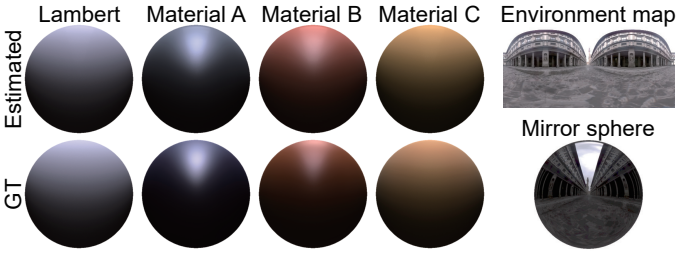


Fig. 7. Rendering of a sphere scene with estimated BRDFs under the natural lighting environment. The right-top figure shows the environment map. We also show the lighting condition by showing a rendered mirror sphere in the right-bottom. GT means the rendering with the ground truth BRDFs.

absolute error (MAE), we scale the intensity range to $[0, 1]$. "Test light 1" and "Test light 2" cases are particularly difficult ones, because their light directions are near perpendicular to the viewing direction, that are not covered by the training data. As a result, we can observe that specular highlights in the left-most example in "Test light 1" and "Test light 2" are not successfully rendered. Nevertheless, as seen in the figure, the renderings are plausible in most of the cases, and MAEs stay low. The result of the surface normal estimation and re-rendering of the multiple materials scene ("Multiple" column) in Figs. 4 and 5 demonstrates the our method's capability of per-pixel estimation as well.

Figure 6 shows the plots of estimated BRDFs by varying the light direction l_θ and viewing direction v_θ, which are the polar angles of light and viewing directions respectively in the polar coordinate system. We show the estimated BRDFs of the metallic material "Material B" with `blob02` and diffuse material "Material C" with `blob08`. While we achieve reasonably good performance for the viewing directions near $v_\theta = 0°$, it shows lower accuracy in steeper angles, such as $v_\theta = 60°$ and $l_\theta > 60°$, due to that the training light directions do not cover these regions.

To qualitatively assess the prediction performance of BRDFs, we show the rendered results under natural environment lightings. Figure 7 shows the rendered sphere with estimated BRDFs under natural lighting environment. We use "Uffizi Gallery, Italy" from High-Resolution Light Probe Image Gallery [3] as the environment map and the physically based renderer, Mitsuba [4]. The appearance of rendered images shows good agreement to the ground truth.

5.2 Evaluation using real-world dataset

For the evaluation with real-world images, we use the DiLiGenT dataset [51], which contains observations of ten different objects under 96 light directions and the ground truth surface normal maps measured by a laser scanner.

Figure 8 shows the estimated normal maps and corresponding error maps for "Proposed w/o SL", "Proposed", "L2", and "L1". Here, we show five objects out of ten in the DiLiGenT dataset (`buddha`, `goblet`, `harvest`, `pot2`, and `reading`). Among them, `buddha`, `pot2`, and `reading` are pottery objects, whose reflections are mostly Lambertian except for some sparse specular highlights; therefore, L2-, and especially L1-based methods work well. For an object like `goblet`, which is made of metallic materials and exhibit strong specular reflection in a wide area, the estimation error becomes larger with the Lambertian based methods, *i.e.*, L2 and L1, while the proposed method produces highly accurate estimates. For `harvest`, the estimation accuracy is poor for all methods due to the complexity of the shape of the object, particularly due to the large cast shadow region in the middle.

Figure 9 shows the rendered images with the predicted BRDFs and the ground truth surface normal map under light directions that are used for training. In the figure, we show two objects: One is mostly diffuse (`buddha`), and the other is highly specular

3. High-Resolution Light Probe Image Gallery: http://gl.ict.usc.edu/data/highresprobes/
4. Mitsuba: https://www.mitsuba-renderer.org/

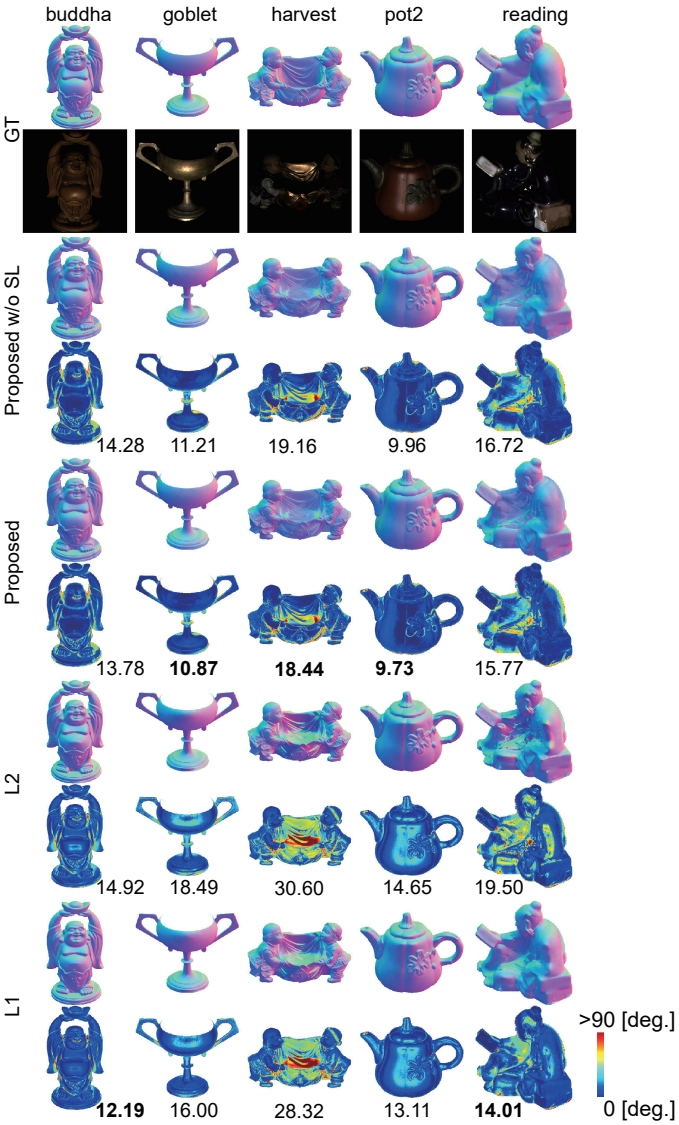


Fig. 8. Surface normal estimation result for real-world scenes from DiLiGenT [51]. In each row, a normal map is shown on top of an error map. The numbers represent Mean Angular Error (MAngE) in degree. GT means the ground truth, and figures under GT are examples of observed images. The contrast of observation images is adjusted for better visualization.

(goblet). For both cases, it can be seen that our method can relight the scenes well because of the accurate estimation of BRDFs. We also show a qualitative evaluation of the re-rendering results under novel light directions in Fig. 10 that are not included in the training dataset. The light directions (a) and (c) are more difficult conditions than (b) since they are off from the training light conditions. Even under such conditions, our method works well supported by the prior knowledge of MERL BRDFs and produces faithful predictions including specularities for the most of the parts. The accuracy is limited at some regions due to the lack of observations, where we can observe unnatural predictions; for example, a strong specular highlight in goblet–(a).

Hold-out validation for BRDF estimation in DiLGenT

To evaluate the BRDF estimation, we performed the light source hold-out validation using DiLiGenT dataset. We randomly choose six light directions out of 96 and keep them as test light directions without using them as a part of training data. Using the BRDF coefficients predicted from 90 input images, we render images with the predicted BRDF parameters under the test light directions and compare with the ground truth.

Figure 11 shows the rendered results of four objects; buddha, goblet, cat, and reading. Here, we show results under three light directions out of six for each object, and below each subfigure, MAE for the 90 training data (Train MAE) and six test data (Test MAE) are respectively shown. The accuracy of the test data shows is almost equivalent to that of training data, which shows the the accuracy of our BRDF prediction. The pixels with strong and sharp specular highlights, *e.g.* observed at the shoulder parts of reading, have less accurate estimates because such strong and sharp reflections cannot be expressed by a linear combination of MERL BRDFs. For cat scene, there is a large error around the foreleg. This is caused by the cast shadow, while our rendering does not account for the cast shadow.

Benchmark comparison

We compare the surface normal estimation accuracy of the proposed methods ("Proposed w/o SL" and "Proposed") with DiLiGenT benchmark results shown in [51], which includes the evaluation of the following ten methods: WG10 [53], IW12 [16], GC10 [28], AZ08 [54], ST12 [55], HM10 [56], ST14 [21], IA14 [57], CH18 [39], and SI18 [40]. Table 2 shows the Mean Angular Error (MAngE) in degree of each method for each objects and the average performance over all the scenes. Green color represents preferable results and red indicates poorer results. For each object, the best result is highlighted with a bold font. It can be seen that our method can achieve at a similar level of accuracy compared to other methods. Recent DNN-based methods [39], [40] exhibit smaller angular errors, which shows the benefit of a DNN-based approach. While our result is slightly less accurate than the newest ones, our method is able to predict BRDFs unlike other techniques.

5.3 Effect of the shadow layer

To observe the effect of shadow layer, we prepare Fig. 12 for depicting the difference of error maps in the surface normal prediction with and without the shadow layer. Here, we use four objects (ball, goblet, harvest, and pot2) as examples. For all these objects, the accuracy is generally improved in boundary areas where shadows often occur. In some areas, the accuracy does not improve for goblet and harvest. For metallic objects with strong interreflections like goblet and harvest, the measurement becomes greater than 0 even inside shadow, *e.g.*, Fig. 13. In such a case, the shadow layer may actually degrades the accuracy due to the wrong assumption that the measurement in shadow becomes 0.

The effect of shadowing probability p may depends on the shape of the target object. Figure 14 shows the variation of result in cases where $p = 0.05$ and $p = 0.9$ on ball. Comparing (a) and (b), we can see that although (b) improves more in the peripheral parts of the sphere (shadowed area) than (b), it deteriorates in the central part. With the shadow layer, since the model is optimized for shadow, the accuracy becomes lower in the area where shadow is never observed. In the case of $p = 0.05$, since the dropout rate r sampled in the manner of Sec. 4.2 can be 0, the estimation accuracy does not deteriorate in areas without shadow. On the other hand, in the case of $p = 0.9$, dropout would be applied to

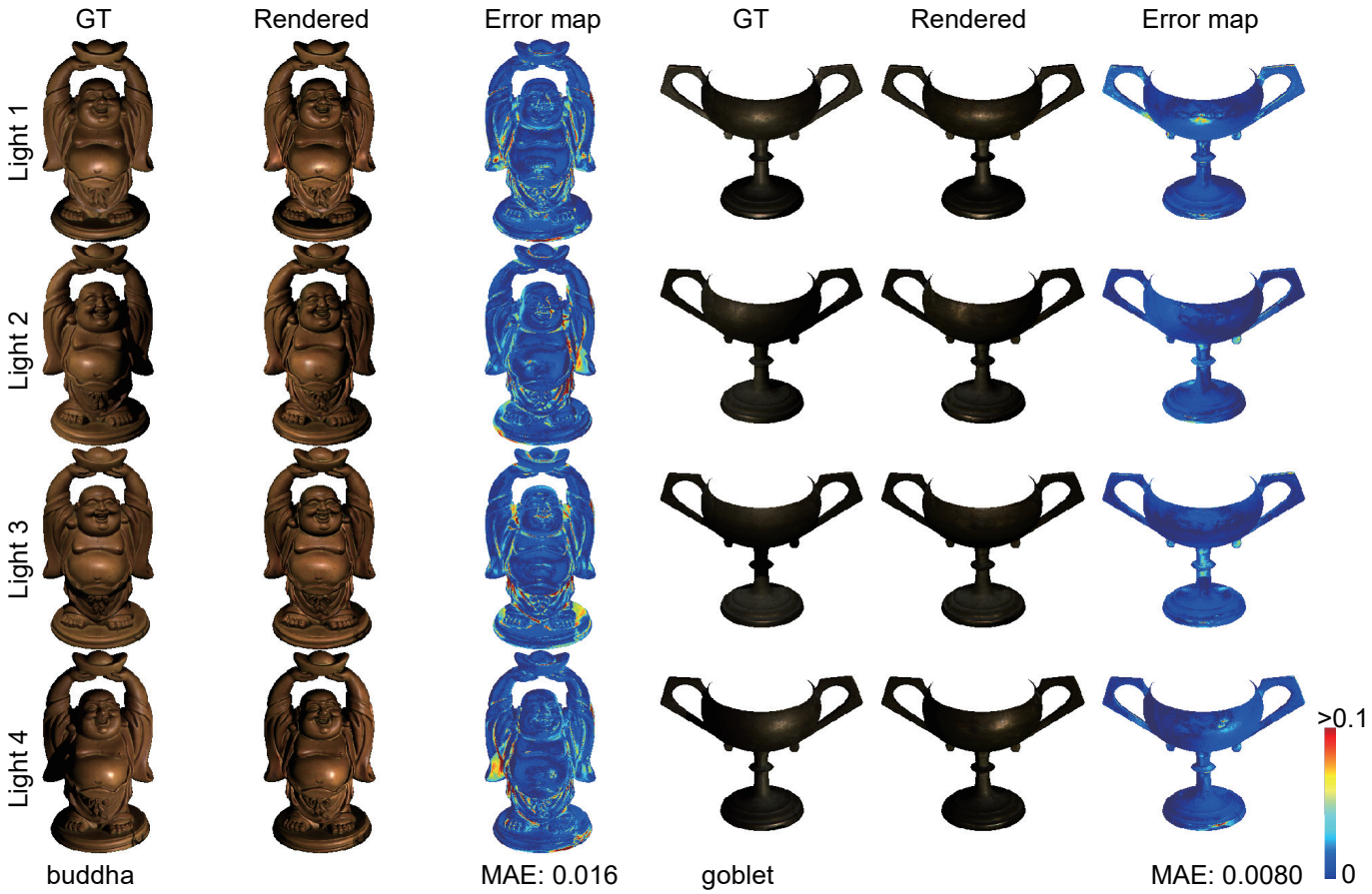


Fig. 9. Rendered results with the estimated BRDFs and the ground truth normal maps for DiLiGenT. `buddha` is the object with non-specular materials and `goblet` is made of metal materials. We apply Gamma correction with $\gamma = 0.8$ to the ground truth and rendered images for visualization. The light conditions Light 1 to Light 4 are from 96 DiLiGenT light conditions.

Fig. 10. Qualitative evaluation of re-rendering under novel light directions. We show the three images for each object rendered with the new light directions that are not included in the training data. Leftmost plot illustrates the light directions. Gray points represent the 96 light directions used in the training data, and red, green, and blue points are the new light directions that corresponds to images (a), (b), and (c), respectively.

TABLE 2
Evaluation results using the DiLiGenT Benchmark [51].

	ball	bear	buddha	cat	cow	goblet	harvest	pot1	pot2	reading	AVG.
Proposed w/o SL	2.63	7.22	14.3	7.10	7.96	11.2	19.2	8.91	9.6	16.7	10.5
Proposed	2.49	7.05	13.8	7.05	7.92	10.9	18.4	8.73	9.73	15.8	10.2
SI18	2.20	**4.10**	7.90	**4.60**	7.90	**7.30**	**13.9**	**5.40**	**6.00**	**12.6**	**7.20**
CH18	2.80	7.60	**7.90**	6.20	**7.30**	8.60	15.9	7.10	7.30	13.3	8.40
ST14	**1.74**	6.12	10.6	6.12	13.9	10.1	25.4	6.51	8.78	13.6	10.3
IA14	3.34	7.11	10.5	6.74	13.1	9.71	26.0	6.64	8.77	14.2	10.6
WG10	2.06	6.50	10.9	6.73	25.9	15.7	30.0	7.18	13.1	15.4	13.4
AZ08	2.71	5.96	12.5	6.53	21.5	13.9	30.5	7.23	11.0	14.2	12.6
HM10	3.55	11.5	13.1	8.40	15.0	14.9	21.8	10.9	16.4	16.8	13.2
IW12	2.54	7.32	11.1	7.21	25.7	16.3	29.3	7.74	14.1	16.2	13.7
ST12	13.6	19.4	18.4	12.3	7.62	17.8	19.3	10.4	9.84	17.2	14.6
GC10	3.21	6.62	14.9	8.22	9.55	14.2	27.8	8.53	7.90	19.1	12.0
BASELINE	4.10	8.39	14.9	8.41	25.6	18.5	30.6	8.89	14.7	19.8	15.4

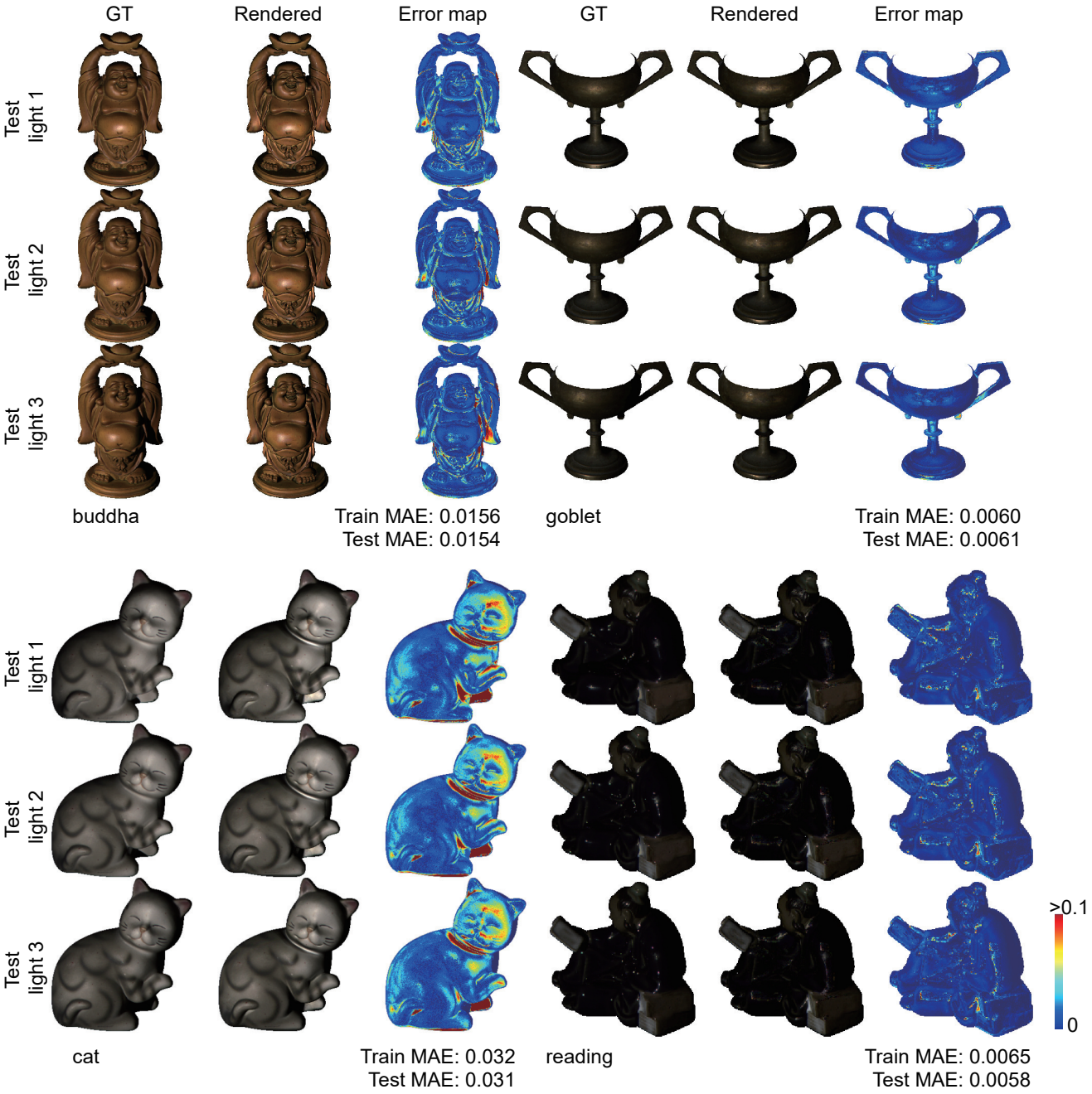


Fig. 11. Result of the hold-out validation of light directions. Among 96 light directions in DiLiGenT, we use 90 directions for training and keep the remaining six for testing. We train the model with the number of light directions $f = 90$ and estimated the BRDFs. Here, we show the three images for each objects rendered under test light directions. For `buddha`, `goblet`, and `reading`, we apply Gamma correct with $\gamma = 0.8$ for visualization. Train MAE is the mean absolute error for 90 light directions, and Test MAE is for the six test light directions.

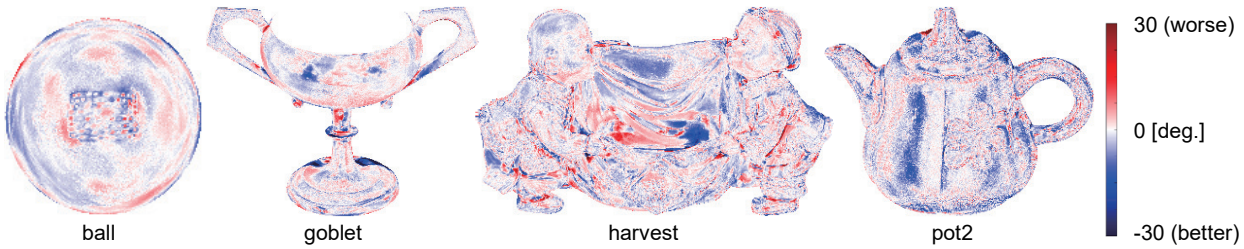


Fig. 12. Improvement by shadow layer. We show a difference map of the error maps of “Proposed w/o SL” and “Proposed”. Pixels whose surface normal estimation accuracy is improved by the shadow layer are colored in blue, otherwise in red.

Fig. 13. Effect of interreflections. We adjust the intensities in the left half in the red box by gamma correction. The shadowed area in the box exhibits measurements greater than 0 due to strong interreflections.

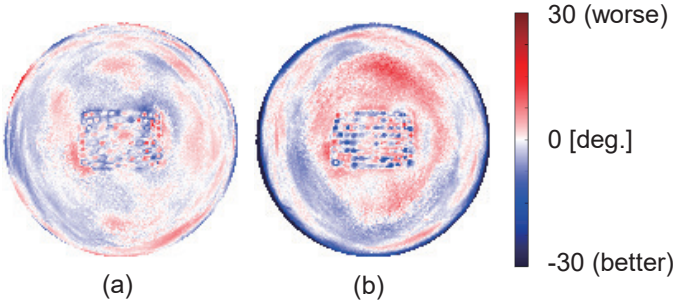


Fig. 14. Comparison of shadowing probability p on `ball` scene. (a) and (b) are difference of the error maps of "Proposed" and "Proposed w/o SL" in case $p = 0.05$ and $p = 0.9$, respectively. Pixels whose normal estimation accuracy is improved by the shadow layer are shown in blue, otherwise in red.

almost all the inputs, so the accuracy drops in the area where the shadow does not exist.

Including the cast shadow rendering directly in the training dataset generation is another option to deal with the cast shadow effect. Here, we use the dataset with the cast shadow rendering for training and study the effect of the shadow layer. We use a physically based renderer, Mitsuba, to render the training dataset with cast shadow and also prepare the test dataset including cast shadow. For this dataset, we use 10 shapes in the Blobby shape dataset instead of a sphere. We leave out 2 shapes (`blob02` and `blob08`) and use the remaining 8 shapes for the training. Figure 15 shows the surface normal prediction with and without the shadow layer (SL) both trained with dataset with cast shadow rendering. For the test shape, we use `blob08` which is not used in training dataset and has the most complex shape in Blobby shape dataset. We use "Lambert" and "Multiple" for the reflectances. As shown in "Diff map", the shadow layer improves the estimation accuracy around the shadowing area although the network has been training with cast shadows. It shows the effectiveness of the shadow layer in this setting as well. Our shadow layer effectively simulates the cast shadow effect and bypasses the preparation of dataset with cast shadow rendering. And it improves the robustness in both cases with and without the cast shadow rendering in the training dataset. It is of interest to extend the shadow layer to simulate a *structured* cast shadow, *i.e.*, neighboring pixels are likely shadowed together as the work of [58].

6 Discussion

We proposed a method that uses deep neural networks for establishing a flexible mapping from shading observations to surface normal and BRDFs. Since the proposed method can simultaneously estimate surface normal and BRDFs, it can be applied to

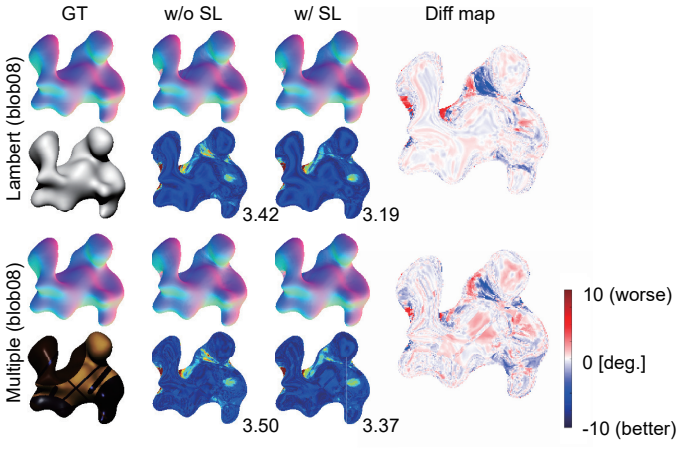


Fig. 15. Comparison of with and without the shadow layer trained with the dataset including the cast shadow rendering. Each row corresponds to one material, and the estimated normal maps are shown on top of the corresponding error map. GT means the ground truth, and one of the observation images is shown below it. "Diff map" shows the difference of the error maps of "w/o SL" and "w/ SL". In "Diff map", pixels whose surface normal estimation accuracy are improved by the shadow layer are colored in blue, otherwise in red. The numbers show the mean angular error in degree.

a wide range of applications such as a material prediction and relighting of target objects for virtual reality.

While early works that use neural networks in the context of photometric stereo can be found in 1990's, they have difficulties in applicability and accuracy. Even though neural networks have been recognized since then as a solution technique for the photometric stereo problem, no effective methods have been introduced until today. The factors that realized our DPSN are (1) availability of the BRDF dataset [8] collected from the real-world objects, which we used for generating training data, and (2) flexible expression power of modern DNNs.

Evaluations show the accurate estimation of the surface normal and BRDFs with our method. Indeed the surface normal estimation accuracy is comparable to the other tailored state-of-the-art methods based on the benchmark evaluation. In addition to estimating surface normal, our method is capable of predicting BRDFs that is expressed by a flexible linear basis representation, which allows us to re-render the scene under a new lighting condition.

One of the limitations of our method is the assumption that light directions are pre-defined and remain the same between training and test phases. It is designed for a photometric stereo device that has fixed light sources and a camera, so that we only need to conduct the training once for the device. Fortunately, many photometric stereo apparatuses use a fixed lamps with respect to the camera, allowing the use of our method. On the other hand, more recent works [39], [40] handle the input images taken under the (known) arbitrary light directions using the techniques such as the aggregated feature map or observation map. One of our future directions is to combine these techniques into our simultaneous estimation for surface normal and reflectances. Moreover, while our method and recent works [39], [40] still assume the given light directions, future venues include simultaneous estimation of the light conditions in addition to surface normal and reflectances so that it can deal with arbitrary light directions that are not predefined and potentially unknown, such as the data captured under a hand-held light source.

ACKNOWLEDGMENTS

This work was supported by JSPS KAKENHI Grant Number JP19H01123. Hiroaki Santo is grateful for support through a JSPS research fellowship for young scientists by the Japan Society for the Promotion of Science (JP19J10326). Boxin Shi is supported by the National Natural Science Foundation of China under Grants 61872012, National Key R&D Program of China (2019YFF0302902), and Beijing Academy of Artificial Intelligence (BAAI).

REFERENCES

[1] R. J. Woodham, "Photometric method for determining surface orientation from multiple images," *Optical engineering*, vol. 19, no. 1, pp. 139–144, 1980.

[2] W. M. Silver, "Determining shape and reflectance using multiple images," Master's thesis, Massachusetts Institute of Technology, Cambridge, MA, U.S., 1980.

[3] J. Lambert, "Photometria," *Augustae Vindelicorum*, 1760.

[4] A. S. Georghiades, "Incorporating the torrance and sparrow model of reflectance in uncalibrated photometric stereo," in *Proceedings of International Conference on Computer Vision (ICCV)*, 2003, pp. 816–825.

[5] K. Torrance and E. Sparrow, "Theory for off-specular reflection from roughened surfaces," *Journal of the Optical Society of America*, vol. 57, pp. 1105–1114, 1967.

[6] L. Chen, Y. Zheng, B. Shi, A. Subpa-Asa, and I. Sato, "A microfacet-based reflectance model for photometric stereo with highly specular surfaces," in *Proceedings of International Conference on Computer Vision (ICCV)*, 2017, pp. 3181–3189.

[7] B. Shi, P. Tan, Y. Matsushita, and K. Ikeuchi, "Bi-polynomial modeling of low-frequency reflectances," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 36, no. 6, pp. 1078–1091, 2014.

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

[9] H. Santo, M. Samejima, Y. Sugano, B. Shi, and Y. Matsushita, "Deep photometric stereo network," in *Proceedings of the International Workshop on Physics Based Vision meets Deep Learning (PBDL) in Conjunction with IEEE International Conference on Computer Vision (ICCV)*, 2017, pp. 501–509.

[10] L. Wu, A. Ganesh, B. Shi, Y. Matsushita, Y. Wang, and Y. Ma, "Robust photometric stereo via low-rank matrix completion and recovery," *Proceedings of Asian Conference on Computer Vision (ACCV)*, pp. 703–717, 2011.

[11] R. Basri and D. Jacobs, "Lambertian reflectance and linear subspaces," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 25, no. 2, 2003.

[12] Y. Mukaigawa, Y. Ishii, and T. Shakunaga, "Analysis of photometric factors based on photometric linearization," *JOSA A*, vol. 24, no. 10, pp. 3326–3334, 2007.

[13] M. A. Fischler and R. C. Bolles, "Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography," *Communications of the ACM*, vol. 24, no. 6, pp. 381–395, Jun. 1981.

[14] T.-P. Wu and C.-K. Tang, "Photometric stereo via expectation maximization," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 32, no. 3, pp. 546–560, 2010.

[15] D. Miyazaki, K. Hara, and K. Ikeuchi, "Median photometric stereo as applied to the segonko tumulus and museum objects," *International Journal of Computer Vision*, vol. 86, no. 2–3, pp. 229–242, 2010.

[16] S. Ikehata, D. Wipf, Y. Matsushita, and K. Aizawa, "Robust photometric stereo using sparse regression," in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2012, pp. 318–325.

[17] A. Georghiades, "Incorporating the torrance and sparrow model of reflectance in uncalibrated photometric stereo," 2003, pp. 816–823.

[18] K. E. Torrance and E. M. Sparrow, "Theory for off-specular reflection from roughened surfaces," *JOSA*, vol. 57, no. 9, pp. 1105–1114, 1967.

[19] R. Ruiters and R. Klein, "Heightfield and spatially varying BRDF reconstruction for materials with interreflections," *Computer Graphics Forum*, vol. 28, no. 2, pp. 513–522, Apr. 2009.

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

[21] B. Shi, P. Tan, Y. Matsushita, and K. Ikeuchi, "Bi-polynomial modeling of low-frequency reflectances," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 36, no. 6, pp. 1078–1091, 2014.

[22] M. Holroyd, J. Lawrence, G. Humphreys, and T. Zickler, "A photometric approach for estimating normals and tangents," *ACM Transactions on Graphics (Proceedings of SIGGRAPH Asia 2008)*, vol. 27, no. 5, p. 133, 2008.

[23] A. Hertzmann and S. M. Seitz, "Example-based photometric stereo: Shape reconstruction with general, varying BRDFs," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 27, no. 8, pp. 1254–1264, 2005.

[24] Z. Hui and A. C. Sankaranarayanan, "Shape and spatially-varying reflectance estimation from virtual exemplars," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 39, no. 10, pp. 2060–2073, 2017.

[25] S. Tominaga and N. Tanaka, "Estimating reflection parameters from a single color image," *IEEE Computer Graphics and Applications*, vol. 20, no. 5, pp. 58–66, 2000.

[26] B. T. Phong, "Illumination for computer generated pictures," *Communications of the ACM*, vol. 18, no. 6, pp. 311–317, Jun. 1975. [Online]. Available: http://doi.acm.org/10.1145/360825.360839

[27] Y. Sato, M. D. Wheeler, and K. Ikeuchi, "Object shape and reflectance modeling from observation," in *Proceedings of annual conference on Computer graphics and interactive techniques*. ACM Press/Addison-Wesley Publishing Co., 1997, pp. 379–387.

[28] D. B. Goldman, B. Curless, A. Hertzmann, and S. M. Seitz, "Shape and spatially-varying BRDFs from photometric stereo," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 32, no. 6, pp. 1060–1071, 2010.

[29] G. J. Ward, "Measuring and modeling anisotropic reflection," in *ACM SIGGRAPH Computer Graphics*, vol. 26, no. 2. ACM, 1992, pp. 265–272.

[30] O. Wang, P. Gunawardane, S. Scher, and J. Davis, "Material classification using BRDF slices," in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2009.

[31] T.-C. Wang, J.-Y. Zhu, E. Hiroaki, M. Chandraker, A. A. Efros, and R. Ramamoorthi, "A 4D light-field dataset and cnn architectures for material recognition," in *Proceedings of European Conference on Computer Vision (ECCV)*. Springer, 2016, pp. 121–138.

[32] Z. Zhou, Z. Wu, and P. Tan, "Multi-view photometric stereo with spatially varying isotropic materials," in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2013, pp. 1482–1489.

[33] J. B. Nielsen, H. W. Jensen, and R. Ramamoorthi, "On optimal, minimal BRDF sampling for reflectance acquisition," *ACM Transactions on Graphics*, vol. 34, no. 6, p. 186, 2015.

[34] Z. Xu, J. Boll Nielsen, J. Yu, H. Wann Jensen, and R. Ramamoorthi, "Minimal BRDF sampling for two-shot near-field reflectance acquisition," *ACM Transactions on Graphics*, vol. 35, pp. 1–12, 11 2016.

[35] Y. Iwahori, R. J. Woodham, H. Tanaka, and N. Ishii, "Neural network to reconstruct specular surface shape from its three shading images," in *Proceedings of International Joint Conference on Nueral Networks*, 1993, pp. 1181–1184.

[36] W.-C. Cheng, "Neural-network-based photometric stereo for 3D surface reconstruction," in *Proceedings of International Joint Conference on Nueral Networks*, 2006, pp. 404–410.

[37] D. Elizondo, S.-M. Zhou, and C. Chrysostomou, "Surface reconstruction techniques using neural networks to recover noisy 3D scenes," in *Proceedings of International Conference on Artificial Neural Networks*, 2008, pp. 857—-866.

[38] Y. Ding, Y. Iwahori, T. Nakamura, R. J. Woodham, L. He, and H. Itoh, "Self-calibration and image rendering using RBF neural network," in *Proceedings of International Conference on Knowledge-Based and Intelligent Information and Engineering Systems*, 2009, pp. 705—712.

[39] G. Chen, K. Han, and K.-Y. K. Wong, "PS-FCN: A flexible learning framework for photometric stereo," in *Proceedings of European Conference on Computer Vision (ECCV)*, 2018.

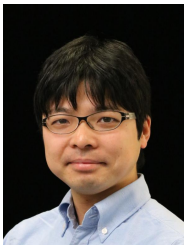
[40] S. Ikehata, "CNN-PS: CNN-based photometric stereo for general non-convex surfaces," in *Proceedings of European Conference on Computer Vision (ECCV)*, 2018.

[41] T. Taniai and T. Maehara, "Neural inverse rendering for general reflectance photometric stereo," in *Proceedings of International Conference on Machine Learning (ICML)*, 2018, pp. 4864–4873.

[42] K. Rematas, T. Ritschel, M. Fritz, E. Gavves, and T. Tuytelaars, "Deep reflectance maps," in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

[43] M. Janner, J. Wu, T. D. Kulkarni, I. Yildirim, and J. Tenenbaum, "Self-supervised intrinsic image decomposition," in *Proceedings of Advances in Neural Information Processing Systems (NIPS)*, 2017, pp. 5936–5946.
[44] V. Deschaintre, M. Aittala, F. Durand, G. Drettakis, and A. Bousseau, "Single-image SVBRDF capture with a rendering-aware deep network," *ACM Transactions on Graphics (TOG)*, vol. 37, no. 4, p. 128, 2018.
[45] Z. Li, K. Sunkavalli, and M. Chandraker, "Materials for masses: SVBRDF acquisition with a single mobile phone image," in *Proceedings of European Conference on Computer Vision (ECCV)*, 2018, pp. 72–87.
[46] K. Kim, J. Gu, S. Tyree, P. Molchanov, M. Nießner, and J. Kautz, "A lightweight approach for on-the-fly reflectance estimation," in *Proceedings of the IEEE International Conference on Computer Vision*, 2017, pp. 20–28.
[47] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Dropout: A simple way to prevent neural networks from overfitting," *The Journal of Machine Learning Research*, vol. 15, no. 1, pp. 1929–1958, 2014.
[48] X. Qi, R. Liao, Z. Liu, R. Urtasun, and J. Jia, "GeoNet: Geometric neural network for joint depth and surface normal estimation," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 283–291.
[49] V. Sze, Y.-H. Chen, T.-J. Yang, and J. S. Emer, "Efficient processing of deep neural networks: A tutorial and survey," *Proceedings of the IEEE*, vol. 105, no. 12, pp. 2295–2329, 2017.
[50] D. Kingma and J. Ba, "Adam: A method for stochastic optimization," *Proceedings of International Conference on Learning Representations (ICLR)*, 2014.
[51] B. Shi, Z. Wu, Z. Mo, D. Duan, S.-K. Yeung, and P. Tan, "A benchmark dataset and evaluation for non-lambertian and uncalibrated photometric stereo," in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.
[52] M. K. Johnson and E. H. Adelson, "Shape estimation in natural illumination," in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2011, pp. 2553–2560.
[53] T. Papadhimitri and P. Favaro, "A closed-form, consistent and robust solution to uncalibrated photometric stereo via local diffuse reflectance maxima," *International journal of computer vision*, vol. 107, no. 2, pp. 139–154, 2014.
[54] N. Alldrin, T. Zickler, and D. Kriegman, "Photometric stereo with non-parametric and spatially-varying reflectance," in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2008, pp. 1–8.
[55] B. Shi, P. Tan, Y. Matsushita, and K. Ikeuchi, "Elevation angle from reflectance monotonicity: Photometric stereo for general isotropic reflectances," *Proceedings of European Conference on Computer Vision (ECCV)*, pp. 455–468, 2012.
[56] T. Higo, Y. Matsushita, and K. Ikeuchi, "Consensus photometric stereo," in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2010, pp. 1157–1164.
[57] S. Ikehata and K. Aizawa, "Photometric stereo using constrained bivariate regression for general isotropic surfaces," in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2014, pp. 2179–2186.
[58] J. Li, A. Robles-Kelly, S. You, and Y. Matsushita, "Learning to minify photometric stereo," in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019.

Hiroaki Santo received his B.S. and M.S. degrees in computer science from Osaka University, Japan, in 2016 and 2018, respectively, where he is currently working toward the Ph.D. His research interests include physics-based vision and machine learning.



Masaki Samejima received the master's and Ph.D. degrees in information science from Osaka University, Japan, in 2007 and 2008, respectively. He was with Hitachi, Ltd., Japan, from 2007 to 2009. He is currently an Assistant Professor with the Department of Multimedia Engineering, Graduate School of Information Science and Technology, Osaka University. His current research interests include applied machine learning and mathematical optimization.

Yusuke Sugano is an associate professor at Institute of Industrial Science, The University of Tokyo. His research interests focus on computer vision and human-computer interaction. He received his Ph.D. in information science and technology from the University of Tokyo in 2010. He was previously an associate professor at Graduate School of Information Science and Technology, Osaka University, a postdoctoral researcher at Max Planck Institute for Informatics, and a project research associate at Institute of Industrial Science, the University of Tokyo.

Boxin Shi received the BE degree from the Beijing University of Posts and Telecommunications, the ME degree from Peking University, and the PhD degree from the University of Tokyo, in 2007, 2010, and 2013. He is currently a Boya Young Fellow Assistant Professor and Research Professor at Peking University, where he leads the Camera Intelligence Group. Before joining PKU, he did postdoctoral research with MIT Media Lab, Singapore University of Technology and Design, Nanyang Technological University from 2013 to 2016, and worked as a researcher in the National Institute of Advanced Industrial Science and Technology from 2016 to 2017. He won the Best Paper Runner-Up award at International Conference on Computational Photography 2015. He has served as an editorial board member of IJCV and an area chair of CVPR.

Yasuyuki Matsushita received his B.S., M.S. and Ph.D. degrees in EECS from the University of Tokyo in 1998, 2000, and 2003, respectively. From April 2003 to March 2015, he was with Visual Computing group at Microsoft Research Asia. In April 2015, he joined Osaka University as a professor. His research area includes computer vision, machine learning and optimization. He is/was an Editor-in-Chief for International Journal of Computer Vision and on editorial board of IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI), The Visual Computer journal, IPSJ Transactions on Computer Vision Applications (CVA), and Encyclopedia of Computer Vision. He served/is serving as a Program Co-Chair of PSIVT 2010, 3DIMPVT 2011, ACCV 2012, ICCV 2017, and a General Co-Chair for ACCV 2014 and ICCV 2021. He is a senior member of IEEE.