GR-PSN: Learning to Estimate Surface Normal and Reconstruct Photometric Stereo Images

Yakun Ju, *Member, IEEE,* Boxin Shi, *Senior Member, IEEE,* Yang Chen, Huiyu Zhou,
Junyu Dong, *Member, IEEE,* and Kin-Man Lam, *Senior Member, IEEE*

Abstract—In this paper, we propose a novel method, namely GR-PSN, which learns surface normals from photometric stereo images and generates the photometric images under distant illumination from different lighting directions and surface materials. The framework is composed of two subnetworks, named GeometryNet and ReconstructNet, which are cascaded to perform shape reconstruction and image rendering in an end-to-end manner. ReconstructNet introduces additional supervision for surface-normal recovery, forming a closed-loop structure with GeometryNet. We also encode lighting and surface reflectance in ReconstructNet, to achieve arbitrary rendering. In training, we set up a parallel framework to simultaneously learn two arbitrary materials for an object, providing an additional transform loss. Therefore, our method is trained based on the supervision by three different loss functions, namely the surface-normal loss, reconstruction loss, and transform loss. We alternately input the predicted surface-normal map and the ground-truth into ReconstructNet, to achieve stable training for ReconstructNet. Experiments show that our method can accurately recover the surface normals of an object with an arbitrary number of inputs, and can re-render images of the object with arbitrary surface materials. Extensive experimental results show that our proposed method outperforms those methods based on a single surface recovery network and shows realistic rendering results on 100 different materials. Our code can be found in https://github.com/Kelvin-Ju/GR-PSN.

Index Terms—Photometric stereo, surface normal estimate, 3D reconstruction, deep neural networks, photometric image reconstruction.

1 Introduction

Recovering the 3D shape of an object is a pivotal problem in many computer graphics and vision applications because it can further improve the understanding of images and scenes [1], [2], [3], [4]. Photometric stereo aims to recover the dense 3D surface normals of an object under changing light directions, with a fixed camera [5]. Theoretically, changing illuminations will provide varying shading cues for recovering the surface normals, while the shading cues are affected by the non-Lambertian surface reflectance. Traditional photometric stereo methods attempt to solve these problems by approximating bidirectional reflectance distribution functions (BRDFs) [6], [7], [8] or rejecting non-Lambertian outliers [9], [10], [11]. However, these models are accurate for limited categories of materials and suffer from unstable optimization. Fortunately, photometric stereo based on deep learning has recently been introduced, which can better estimate surface normals and relax the non-Lambertian constraint [12], [13], [14], [15], [16], [17]. Meanwhile, deep learning-based rendering methods have also been proposed to generate photorealistic appearance from a 3D shape [18], [19], [20], which is a core problem in computer graphics. These rendering techniques relax the dense sampling requirement, noise, and fuzzy conditions, due to the powerful learning ability of deep neural networks. So far, surface-normal reconstruction and photometric-image generation tasks have entered the era of deep neural networks.

To further improve the accuracy, however, blindly increasing the model complexity and training dataset for training the photometric stereo networks may not be effective. Different from most existing methods, which solely focus on the constraint in the surface normal domain, we consider both the surface normal supervision and the image reconstruction supervision. In this paper, we aim to study the relations between these two tasks and devise a framework for training these two tasks simultaneously to reinforce the learning of each other.

To achieve the above objectives, we propose a deep learning framework consisting of two associated subnetworks, called GeometryNet and ReconstructNet, which are connected in cascade, as shown in Fig. 1. In GeometryNet, we apply a bilateral extraction module and a top-k pooling fusion module in a shallow-deep framework to extract features from input images. In short, GeometryNet reconstructs the surface normals of an object from calibrated photometric stereo images, while ReconstructNet uses the predicted normals to reproduce the photometric images of an object under different lighting conditions. In fact, ReconstructNet can be regarded as the inverse task of surface-normal predic-

- *Y. Ju and KM. Lam are with the Department of Electronic and Information Engineering, The Hong Kong Polytechnic University, Hong Kong (e-mail: kelvin.yakun.ju@gmail.com, enkmlam@polyu.edu.hk).*
- *B. Shi is National Key Laboratory for Multimedia Information Processing and National Engineering Research Center of Visual Technology, School of Computer Science, Peking University, Beijing, China (e-mail: shiboxin@pku.edu.cn).*
- *Y. Chen is with the Tsinghua Shenzhen International Graduate School, Tsinghua University (e-mails: cy21@mails.tsinghua.edu.cn).*
- *H. Zhou is with the School of Computing and Mathematical Sciences, University of Leicester, Leicester, UK (e-mail: hz143@leicester.ac.uk).*
- *J. Dong is with the Department of Computer Science and Technology, Ocean University of China, Qingdao, China (e-mail: dongjunyu@ouc.edu.cn).*
- *Corresponding authors: J. Dong and KM. Lam*

tion and provides additional supervision for GeometryNet to reduce the potential space of surface normals and to form a closed-loop structure.

However, the aim of the rendering process is not to reproduce the same image from the input, but it should produce a photorealistic appearance under an arbitrary lighting direction and the surface reflectance properties [21]. We address this issue in two ways. First, we explicitly combine the encoded lighting with high-dimensional features to output arbitrarily specified rendered images under different illumination directions. Second, we further encode the 100 different materials in the MERL BRDF dataset [22] to form 100-dimensional one-hot features, following the encoding way in ConditionalGAN [23], where the material information is treated as a condition.

To better learn the characteristics of the material properties and improve the utilization of training data from available datasets, we propose a parallel framework to learn the rendering of an object with two different materials. To achieve this, we simultaneously input two sets of photometric stereo images and render the reconstructed object with swapped materials. ReconstructNet regresses the two materials according to different encoding from the material, forming the reconstruction loss and transform loss. As shown in Fig. 1, the proposed method is trained by minimizing these two losses, in addition to the surface-normal loss.

To stabilize training, we employ a training strategy: ReconstructNet is alternately trained with the predicted normal map by GeometryNet and the ground-truth normal map. Specifically, after training the whole pipeline described in Fig. 1 (input $A \rightarrow$ reconstruct $\tilde{A}$, input $A \rightarrow$ reconstruct $\tilde{B}$, input $B \rightarrow$ reconstruct $\tilde{A}$, and input $B \rightarrow$ reconstruct $\tilde{B}$), we train ReconstructNet additionally using the ground-truth surface normal twice (generating $\tilde{A}$ and $\tilde{B}$). Experiments show that this strategy is beneficial to the convergence of both GeometryNet and ReconstructNet.

Our method employs widely used synthetic datasets for training [24], [25]. Concretely, we render every sample of an object with two randomly BRDFs from the MERL BRDF dataset [22]. We provide a thorough ablation experiment using the synthetic test dataset [15]. We also demonstrate the performance of our method on the widely used DiLiGenT benchmark dataset [26], the synthetic test data [27], and the real photoed Light Stage Data Gallery dataset [28]. We show that the proposed method outperforms state-of-the-art deep learning-based methods, as well as traditional methods, on surface-normal estimation. Additionally, our GR-PSN can generate images with 100 different materials [22].

In summary, this paper focuses on how to unify 3D reconstruction and rendering in a single framework and how to further improve the accuracy of surface-normal estimation. Our contributions are as follows:

- The proposed GR-PSN puts additional reconstruction loss and transform loss on reconstructed images, by the use of ReconstructNet, which forms a closed-loop structure and improves the learning of surface normals for shape recovery.
- We propose two simple but effective bilateral extraction and top-k pooling modules, to efficiently fuse features from a variable number of extracted features.
- The proposed method can simultaneously estimate surface normals and render photometric images under distant lighting from different directions and surface materials.

2 Related Work

In order to understand our contributions and how our method relates to those in the literature, in this section, we briefly review two areas, namely photometric stereo based on deep learning and 3D recovery by reconstructing images.

2.1 Photometric stereo based on deep learning

Photometric stereo [5] perceives the 3D shape of an object through changing shading cues, based on the Lambertian assumption. However, ideal Lambertian surfaces barely exist in the real world, therefore, many methods have been proposed to deal with the non-Lambertian surface problems. Traditional methods always treat non-Lambertian surfaces as outliers [9], [10], [11], [29], [30], or approximate non-Lambertian reflectance observations by using sophisticated BRDF models [6], [7], [8], [31], [32]. However, these handcrafted reflectance models are effective for limited classes of surface reflectance and suffer from unstable optimization.

Meanwhile, deep learning techniques [33], [34], [35] have shown powerful fitting ability in photometric stereo networks. The deep learning-based photometric stereo was first proposed by Santo *et al.* [12]. Then, various deep learning-based methods were proposed, which better relax the input constraints to further improve the estimation accuracy. These methods can be divided into two main categories, according to how the input images are processed [36], [37]. The methods in the first category use the intensity of every pixel as input. DPSN [12] estimates the per-pixel surface normal based on a fixed number of observations, which requires the training and testing samples to have the same pre-defined lighting conditions. To relax this limitation, CNN-PS [13] first proposes an observation map to merge all observations pixel by pixel, having the ability to handle inputs with order-agnostic lighting. SPLINE-Net [14] and LMPS [38] then apply a lighting interpolation strategy and a critical illumination strategy to relax the limitation of sparsity in the number of input images. Recently, PX-Net [16] proposes an observation map-based method that considers the effect of global illumination, while other methods, such as GPS-Net [39] and HT21 [40], learn global information by combining the per-pixel and all-pixel strategies. More detailed surveys about deep learning-based photometric stereo can be found in Refs [36], [37].

The methods of the second category use all the pixels of images and their corresponding light directions as input, attempting to learn shapes from various appearances. PS-FCN [41] employs the max-pooling operation to process an arbitrary number of input images. PS-FCN (Norm.) [15] further applies observation normalization to handle spatially varying materials. Under the framework of max-pooling, Attention-PSN [42] and NormAttention-PSN [17] propose an adaptive attention-weighted loss to improve

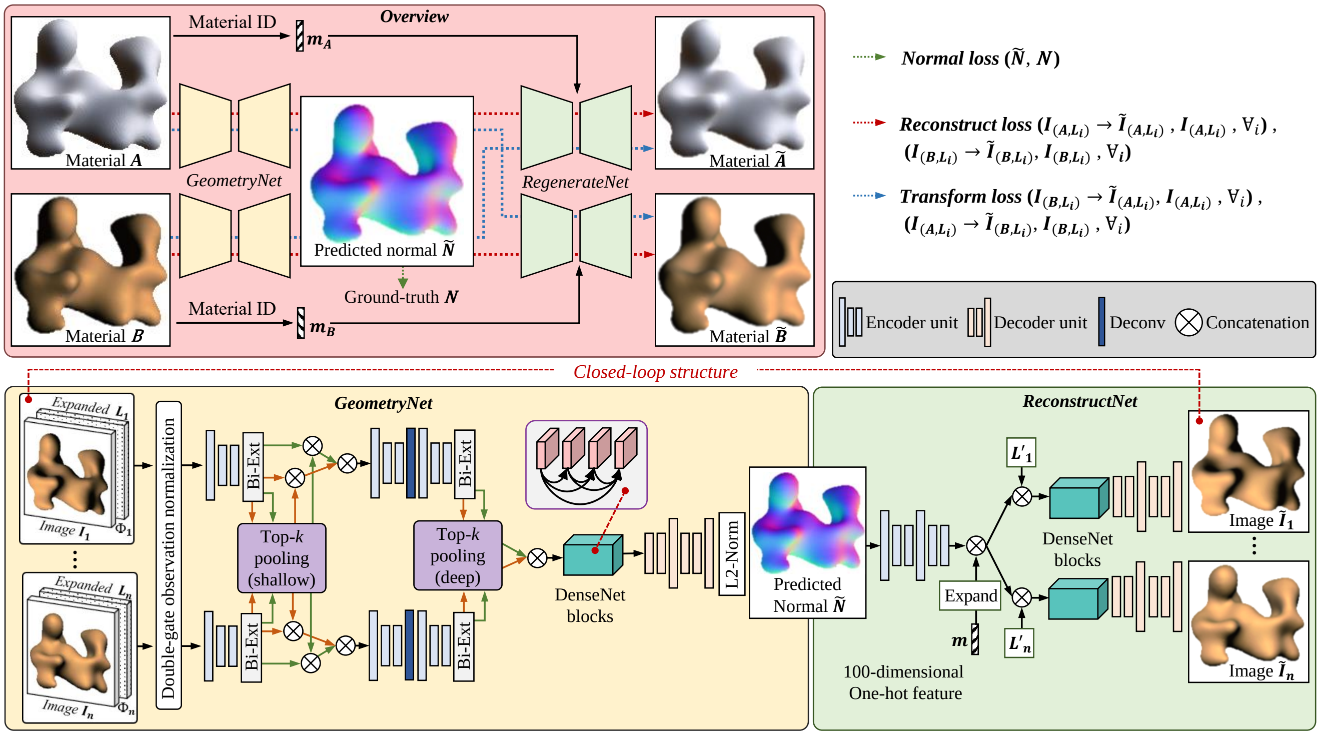

Overview
Material ID
m_A
Material A
GeometryNet
Predicted normal $\tilde{N}$
Ground-truth N
RegenerateNet
Material $\tilde{A}$
Material ID
m_B
Material B
Material $\tilde{B}$
Normal loss $(\tilde{N}, N)$
Reconstruct loss $(I_{(A,L_i)} \rightarrow \tilde{I}_{(A,L_i)}, I_{(A,L_i)}, \forall_i)$, $(I_{(B,L_i)} \rightarrow \tilde{I}_{(B,L_i)}, I_{(B,L_i)}, \forall_i)$
Transform loss $(I_{(B,L_i)} \rightarrow \tilde{I}_{(A,L_i)}, I_{(A,L_i)}, \forall_i)$, $(I_{(A,L_i)} \rightarrow \tilde{I}_{(B,L_i)}, I_{(B,L_i)}, \forall_i)$
Encoder unit
Decoder unit
Deconv
Concatenation
Closed-loop structure
GeometryNet
Expanded L_1
Image I_1
Φ_1
Expanded L_n
Image I_n
Φ_n
Double-gate observation normalization
Bi-Ext
Top-k pooling (shallow)
Top-k pooling (deep)
DenseNet blocks
L2-Norm
Predicted Normal $\tilde{N}$
ReconstructNet
L'_1
L'_n
Expand
m
100-dimensional One-hot feature
DenseNet blocks
Image $\tilde{I}_1$
Image $\tilde{I}_n$

Fig. 1. The overall structure of our method (red box), composed of GeometryNet (yellow box, see Section 3.3) and ReconstructNet (green box, see Section 3.4), which are trained by minimizing the three loss functions (see Section 3.5) shown in the upper right corner. In GeometryNet, we propose the bilateral extraction (Bi-Ext) and top-k pooling modules (see Section 3.2). A and B represent two arbitrary materials of an object.

the reconstruction performance on different surface regions. IRPS [43] proposes an unsupervised learning framework to estimate surface normals by minimizing the reconstruction loss. Recently, some methods have also introduced advanced feature aggregation methods [44], [45] and feature extraction modules [46], [47], [48]. More recently, Ikehata [49] introduced a novel approach that departs from physical lighting models and instead extracts a universal lighting representation through image interactions, termed UniPS. In this way, UniPS can effectively handle a wide range of lighting variations, including parallel, spatially-varying, near-field, and even complex outdoor lighting scenarios.

In fact, the above-mentioned methods are all based on a single constraint, while our method combines both reconstruction and rendering, to form a closed-loop architecture, trained with additional supervision. Recently, our previous work first uses a dual regression network to realize the relighting of photometric stereo [21], namely DR-PSN. However, DR-PSN can only render images with an arbitrary light direction, but without changing the material, and the performance of DR-PSN is limited. In this paper, we propose GR-PSN to simultaneously learn surface normal and regenerate the photometric images with arbitrary surface reflectance and light direction. Furthermore, we explore advanced feature extraction modules for better surface normal estimations.

2.2 3D recovery by reconstructing image

Recently, some unsupervised methods have been proposed because of the lack of 3D ground-truth [43], [50], [51], [52]. Taniai and Maehara [43] proposed an unsupervised method to recover surface normals and albedo by minimizing the reconstruction loss. Tiwari and Raman [53] pursued a holistic approach by simultaneously learning lighting estimation, image relighting, and surface normal estimation to tackle the challenge of uncalibrated photometric stereo. Additionally, they proposed an inverse rendering-based deep learning framework, called DeepPS2 [54], that jointly performs surface normal, albedo, lighting estimation, and image relighting, leveraging only two different illuminated photometric stereo images. Similarly, some later works [50], [51], [52] were proposed to learn disentangling the 3D shape, albedo, and lighting by rendering these components from the original input. However, these methods require the assumption of symmetric structures or Lambertian surfaces, which are valid in limited applications. Furthermore, these methods only use the reconstruction loss, without the ability to render photorealistic appearance under arbitrary lighting and materials. Compared with these methods, we realize 3D reconstruction and rendering in series, *i.e.*, using GeometryNet followed by ReconstructNet. There are two advantages with our model. (1) Our method uses the deep neural network, ReconstructNet, to approximate the reconstructed images, avoiding the inherent deviation of illumination model-based physical renderer. (2) Our method uses the one-hot feature [23] and light vector to explicitly encode surface reflectance and incident light in ReconstructNet, forming an additional transform loss, for realizing the reconstruction of arbitrary materials and illumination directions.

3 Methodology

In this section, we present a novel deep framework for photometric stereo. Our goal is to estimate the surface normals $\tilde{N}$ of an object, and regenerate specific images $\tilde{I}_{(M,L)}$ of the object under arbitrary conditions, where M and L represent the material and illumination direction, respectively. Our proposed GR-PSN approximates the image formation model in both the forward and inverse processes, as follows:

$$I^{p}_{(m,L_i)} = \rho_M\left(e_i, \boldsymbol{N^p}, \boldsymbol{l_i}\right) \max\left(\boldsymbol{N^{p\top} l_i}, 0\right), \tag{1}$$

where $\rho_M\left(e_i, \boldsymbol{N^p}, \boldsymbol{l_i}\right)$ is the bidirectional reflectance distribution function (BRDF) of the surface material M, e_i and $\boldsymbol{l_i}$ are the intensity and directions of the illumination light, respectively, $\max\left(\boldsymbol{N}^{p\top}\boldsymbol{l}_i, 0\right)$ accounts for the attached shadows, and $I^{p}_{(M,L_i)}$ and $\boldsymbol{N^p}$ denote the values of $I_{(M,L_i)}$ and $\boldsymbol{N}$, respectively, at the pixel position p. GR-PSN will use GeometryNet to learn the estimated surface normal $\tilde{\boldsymbol{N}}$ in the forward process and ReconstructNet to learn the reconstructed image $\tilde{\boldsymbol{I}}_{\boldsymbol{(M,L)}}$ in the reverse process, to approximate the imaging model shown in Eq. (1).

As discussed above, we propose a deep learning model, as shown in Fig. 1, which contains two associated subnetworks, called GeometryNet and ReconstructNet. They are connected in a cascade and trained by minimizing the three loss functions discussed in Section 3.5. GeometryNet reconstructs the surface normals of an object from calibrated photometric stereo images. At the same time, ReconstructNet uses the predicted surface normals to reproduce photometric stereo images under different surface materials and illuminations. In fact, ReconstructNet can be viewed as the inverse process of GeometryNet, providing additional supervision for surface normal prediction and forming a closed-loop structure.

To achieve a realistic appearance under arbitrary surface materials, the proposed method takes the surface material information as a condition and further encodes 100 materials in the MERL BRDFs dataset [22] to form 100-dimensional one-hot features, following the encoding method in conditional GAN [23]. In order to better learn the properties of surface materials, we propose a parallel framework to learn the rendering process of an object using two different materials. As shown in Fig. 1, A and B represent two arbitrary materials of an object, the model simultaneously inputs two sets of photometric stereo images and renders the reconstructed objects with the encoded surface material features swapped. ReconstructNet regresses the two materials according to different encoded one-hot features, forming the reconstruct loss and transform loss.

This section first presents the baseline operations in deep learning-based photometric stereo networks. Then, we show the structure of GeometryNet and the ReconstructNet. Finally, we introduce the triple-supervised loss function and training method.

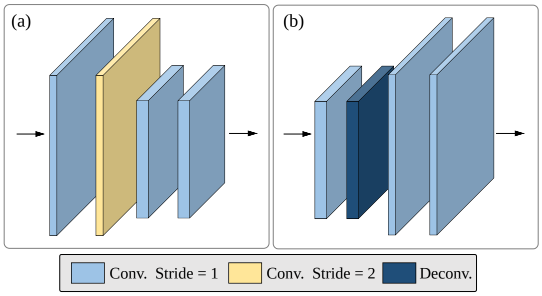


Fig. 2. Network architecture of (a) the encoder unit, and (b)the decoder unit.

3.1 Baseline operations

3.1.1 Double-gate observation normalization

A CNN-based network may fail to estimate an input with spatially varying colors, because it handles inputs in terms of patches, and patches with different colors may cause mutual influence. For photometric stereo mages, multi-materials can also cause drastic changes in color, which can impact the feature extraction of convolutional layers. Therefore, the observation normalization method [15] is employed to remove the impact of spatially varying surface materials, which normalizes each observation by all n observations, as follows:

$$o'_i = \frac{o_i}{\sqrt{o_1^2 + o_2^2 + \cdots + o_n^2}}, \; i \in \{1, 2, \cdots, n\}, \tag{2}$$

where o_i and o'_i represent a pixel value in the i_{th} original observation $\boldsymbol{I}$ and the normalized observation $\boldsymbol{I'}$, respectively. Under the assumption of Lambertian reflectance, the reflectance $\boldsymbol{\rho}$ in Eq. (1) can be totally removed. However, this observation normalization method is not applicable to non-Lambertian conditions. While most of the regions are close to the Lambertian model, those with specular highlights may be impacted after the normalization process, because the highlights enlarge the denominator of Eq. (2) and suppress the normalized pixel intensity. Therefore, our previous method, NormAttention-PSN [17], proposed a double-gate observation normalization method, as follows:

$$o'_i = \frac{o_i}{\sqrt{\sum_k o_k^2}}, \; i \in \mathcal{T}, \; k \in \mathcal{S}, \tag{3}$$

where the set $\mathcal{S}$ is a subset of $\mathcal{T} = \{o_1, o_2, \cdots, o_n\}$, which is controlled by the two gate (thresholds), such that $o_i \in \mathcal{S}$ if $Gate(P_{10}) < o_i < Gate(P_{90})$, for $i = 1, 2, \cdots, n$. The percentile P denotes a positional indicator and divides all observations into two parts. With this baseline operation, our GR-PSN can avoid extracting erroneous feature information from non-Lambertian surfaces with changing materials.

3.1.2 Light direction embedding

As a calibrated photometric stereo method, the light direction of each input image should be known and input into the network. However, an incident light direction is a vector $\boldsymbol{l_i} \in \mathbb{R}^3$, which cannot be fused with the input images $\in \mathbb{R}^{C\times H\times W}$. Therefore, following the widely used operation, we duplicate each light direction $\boldsymbol{l_i}$ to form 3-channel features $\boldsymbol{L_i}$ and $\boldsymbol{L'_i}$, having the same spatial dimension as the input image and fused feature $\in \mathbb{R}^{3\times H\times W}$ and $\in \mathbb{R}^{3\times\frac{1}{4}H\times\frac{1}{4}W}$. In this case, the expanded light direction $\boldsymbol{L_i}$ and $\boldsymbol{L'_i}$ can be concatenated with images and features.

3.1.3 Network units

To largely exclude the influence of other factors and verify the effectiveness of the main modules in our proposed network, *i.e.*, the proposed closed-loop structure, and the

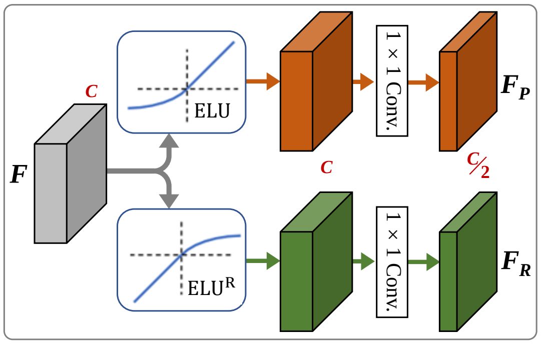


Fig. 3. Architecture of the bilateral extraction module.

bilateral extraction and top-k pooling modules, all other parts of our network are simple fully convolutional layers, as depicted in Fig. 2. For the encoder unit, our structure includes four convolutional layers, with the stride = 2 in the second layer to down-sample the features. In GeometryNet (the deep feature extraction stage), we use the encoder unit twice with a deconvolutional layer to increase the receptive field and preserve spatial information with a small memory burden. For the decoder unit, we design a deconvolutional layer to up-sample the features. All convolutional layers also use 3 × 3 kernels. It is worth noting that the channel and spatial dimensions of convolutional layers are adapted to the inputs.

3.2 Bilateral extraction and top-k pooling modules

We propose a bilateral extraction module and a top-k pooling module to efficiently fuse an arbitrary number of extracted features, in GeometryNet(see Section 3.3). In this Section, we will introduce these two modules.

3.2.1 *Bilateral extraction module*

Conventional activation functions, such as ReLU, LeakyReLU, and ELU, truncate or attenuate negative inputs. This may result in unwanted loss or distortion of information. The non-activated part of the input may contain important features. While some activation functions, such as Sigmoid and Tanh, respond to both positive and negative inputs, the gradients in saturated regions are small and may cause vanishing gradients. To address this limitation, we adopt bilateral extraction, inspired by [55], to maximize the use of the negative part of the features, while retaining the non-linearity. As illustrated in Fig. 3, the input feature $\boldsymbol{F}$ of the bilateral extraction module is activated by the original and the 180° rotated ELU activation functions (bilateral activation), resulting in the positive feature $\boldsymbol{F_P}$ and the rotated negative feature $\boldsymbol{F_R}$. We also add a 1×1 convolutional layer after the bilateral activation to reduce the channel dimension by half.

3.2.2 *Top-k module*

Convolutional neural networks (CNNs) are known to be incapable of handling a variable number of inputs during training and testing. Previous photometric stereo networks mainly apply the max-pooling operation to aggregate an arbitrary number of extracted features [15], [41]. However, max-pooling can only retain the maximum response for each position, thus discarding a large amount of information in the inputs. To relax this limitation, we further adopt a top-k pooling module instead of max-pooling, which keeps the top-k maximum responses from all input features at the same position. The advantages of top-k are twofold. First, more features are retained in this module, which can be beneficial to surface-normal regression. Second, the maximum response feature may face the problem of overexposure (the pixel value is 255 only), when meeting the specular highlights condition, resulting in the loss of all information. Nevertheless, this issue can be alleviated by learning from a number of maximum features. In our method, the number of top feature values k is 3, *i.e.*, the three largest features at each pixel position are considered. However, in this case, the dimension of the fused feature is k times that of the input feature.

3.3 GeometryNet

In this Section and 3.4, we illustrate GeometryNet and ReconstructNet in detail, as shown in Fig. 1.

Given n arbitrary normalized observations $\{\boldsymbol{I'}_{(\boldsymbol{L_1})}, \boldsymbol{I'}_{(\boldsymbol{L_2})}, \cdots, \boldsymbol{I'}_{(\boldsymbol{L_n})}\}$, where $\boldsymbol{I'}_{(\boldsymbol{L_i})} \in \mathbb{R}^{3\times H\times W}$, $i \in \{1, 2, \cdots, n\}$, concatenated with the expanded illumination direction $\boldsymbol{L_i} \in \mathbb{R}^{3\times H\times W}$ (discussed in Section 3.1.2), GeometryNet outputs surface normals $\tilde{\boldsymbol{N}}$, as follows:

$$\tilde{\boldsymbol{N}} = f_{ge}(\boldsymbol{I'}_{(\boldsymbol{L_i})}, \boldsymbol{L_i}; \theta_{ge}), \tag{4}$$

where f_{ge} is a feed-forward network with learnable parameters θ_{ge}, *i.e.*, the yellow box in Fig. 1.

Different from PS-FCN [41], we propose GeometryNet, with a deep-shallow and global-local multiple feature fusion framework [45]. Our network structure adopts the above-mentioned bilateral extraction and top-k pooling modules (see Section 3.2) to fuse local and global features of the inputs in the shallow layers, so as to generate comprehensive features for predicting surface normals. We argue that (1) the global selection mechanism only extracts the most k salient representations of each feature, while the discarded local features may still be important for estimation of surface normals, and (2) deep and shallow feature fusion has an irreplaceable impact on the extracted features, due to the fact that the deep and shallow features are extracted with different receptive fields, so the features should contain unique information. Therefore, GeometryNet integrates global-local features and deep-shallow features, and these fused features can further improve the estimated surface normals, compared to the original framework.

To better estimate the surface normals at the output, a DenseNet-based module [56] is also employed. In our GeometryNet, three dense blocks, with 2, 4, and 3 layers, are used.

3.4 ReconstructNet

With the predicted surface normals $\tilde{\boldsymbol{N}}$ for an encoded material from GeometryNet, ReconstructNet can produce images of the object with the specific material under different specified light directions $\boldsymbol{l_i}$. Therefore, we encode the materials, using a one-hot feature vector $\boldsymbol{M} \in \mathbb{R}^{100}$.

Given the predicted surface normals $\tilde{\boldsymbol{N}}$, the specific material vector $\boldsymbol{M}$, and the embedded lighting direction $\boldsymbol{L}_i'$ (see Section 3.1.2), ReconstructNet produces the reconstructed image $\tilde{\boldsymbol{I}}_{(\boldsymbol{M},\boldsymbol{L_i})}$, as follows:

$$\tilde{\boldsymbol{I}}_{(\boldsymbol{m},\boldsymbol{L_i})} = f_{re}(\tilde{\boldsymbol{N}}, \boldsymbol{M}, \boldsymbol{L}_i'; \theta_{re}), \tag{5}$$

where f_{re} represents a feed-forward encoder with learnable parameters θ_{re}. As shown in the green box in Fig. 1, the material vector $\boldsymbol{M} \in \mathbb{R}^{100}$ is first expanded to the spatial resolution of 100 $\times \frac{1}{4}H \times \frac{1}{4}W$ (same as the expansion method for the lighting direction), and the number of feature-map channels is increased to 256, via a 1×1 convolutional layer. This generated material feature is then concatenated with the surface-normal feature extracted by the two encoder units and the specific expanded illumination direction $\boldsymbol{L}_i'$, which can then generate the reconstructed image $\tilde{\boldsymbol{I}}_{(\boldsymbol{M},\boldsymbol{L_i})}$ via the DenseNet-based module [56] (same as the one in GeometryNet) and two decoder units, following the image formation model described in Eq. (1).

3.5 Loss function

We optimize the parameters θ_{gr} and θ_{re} by minimizing the following joint loss function $\mathcal{L}$, as follows:

$$\mathcal{L} = \mathcal{L}_{\text{normal}} + \lambda(\mathcal{L}_{\text{recon}} + \mathcal{L}_{\text{trans}}), \tag{6}$$

where $\mathcal{L}_{\text{normal}}$ defines the surface-normal loss between the predicted surface normal $\tilde{\boldsymbol{N}}$ and the ground truth $\boldsymbol{N}$, given as follows:

$$\mathcal{L}_{\text{normal}} = \frac{1}{HW}\sum_{p}^{HW}(2 - \tilde{\boldsymbol{N}}_{(A)}^{p} \odot \boldsymbol{N}^{p} - \tilde{\boldsymbol{N}}_{(B)}^{p} \odot \boldsymbol{N}^{p}), \tag{7}$$

where $\tilde{\boldsymbol{N}}_{(A)}^{p}$ and $\tilde{\boldsymbol{N}}_{(B)}^{p}$ denote the estimation via observations with materials A and B, and $\odot$ represents the dot-product operation. If the estimated surface normal $\tilde{\boldsymbol{N}}^{p}$ has a similar orientation to that of the ground truth $\boldsymbol{N}^{p}$ at pixel p, $\tilde{\boldsymbol{N}}^{p} \odot \boldsymbol{N}^{p}$ will be close to 1 and the corresponding loss will approach 0.

The remaining two terms define the two losses between the reconstructed images $\{\tilde{\boldsymbol{I}}_{(\boldsymbol{A},\boldsymbol{L_1})}, \tilde{\boldsymbol{I}}_{(\boldsymbol{A},\boldsymbol{L_2})}, \cdots, \tilde{\boldsymbol{I}}_{(\boldsymbol{A},\boldsymbol{L_n})}\}$, $\{\tilde{\boldsymbol{I}}_{(\boldsymbol{B},\boldsymbol{L_1})}\tilde{\boldsymbol{I}}_{(\boldsymbol{B},\boldsymbol{L_2})}, \cdots, \tilde{\boldsymbol{I}}_{(\boldsymbol{B},\boldsymbol{L_n})}\}$ and the real observation images $\{\boldsymbol{I}_{(\boldsymbol{A},\boldsymbol{L_1})}, \boldsymbol{I}_{(\boldsymbol{A},\boldsymbol{L_2})}, \cdots, \boldsymbol{I}_{(\boldsymbol{A},\boldsymbol{L_n})}\}$, $\{\boldsymbol{I}_{(\boldsymbol{B},\boldsymbol{L_1})}, \boldsymbol{I}_{(\boldsymbol{B},\boldsymbol{L_2})}, \cdots, \boldsymbol{I}_{(\boldsymbol{B},\boldsymbol{L_n})}\}$, with two different materials A and B, and different lighting directions $\boldsymbol{L_i}$, $i \in \{1, 2, \cdots, n\}$. $\mathcal{L}_{\text{recon}}$ denotes the reconstruction loss between the input images and the reconstructed images with the same material (*i.e.*, the red line in Fig. 1), which is defined, as follows:

$$\begin{aligned}\mathcal{L}_{\text{recon}} = \frac{1}{n}\sum_{i}^{n}(&\|\tilde{\boldsymbol{N}}_{(A)} \rightarrow \tilde{\boldsymbol{I}}_{(\boldsymbol{A},\boldsymbol{L_i})}, \boldsymbol{I}_{(\boldsymbol{A},\boldsymbol{L_i})}\|_2^2 \\ &+\|\tilde{\boldsymbol{N}}_{(B)} \rightarrow \tilde{\boldsymbol{I}}_{(\boldsymbol{B},\boldsymbol{L_i})}, \boldsymbol{I}_{(\boldsymbol{B},\boldsymbol{L_i})}\|_2^2),\end{aligned} \tag{8}$$

Another loss, denoted as $\mathcal{L}_{\text{trans}}$, is the reconstruction loss between the input images and the reconstructed images of different materials (*i.e.*, the blue arrows in Fig. 1), which is defined as follows:

$$\begin{aligned}\mathcal{L}_{\text{trans}} = \frac{1}{n}\sum_{i}^{n}(&\|\tilde{\boldsymbol{N}}_{(B)} \rightarrow \tilde{\boldsymbol{I}}_{(\boldsymbol{A},\boldsymbol{L_i})}, \boldsymbol{I}_{(\boldsymbol{A},\boldsymbol{L_i})}\|_2^2 \\ &+\|\tilde{\boldsymbol{N}}_{(A)} \rightarrow \tilde{\boldsymbol{I}}_{(\boldsymbol{B},\boldsymbol{L_i})}, \boldsymbol{I}_{(\boldsymbol{B},\boldsymbol{L_i})}\|_2^2).\end{aligned} \tag{9}$$

Algorithm 1 GR-PSN Training Algorithm

```
for j = 1 : Num_of_samples
Input: Images of the sample with material $A$ $\{\boldsymbol{I}_{(\boldsymbol{A},\boldsymbol{L_1})}, \boldsymbol{I}_{(\boldsymbol{A},\boldsymbol{L_2})}, \cdots, \boldsymbol{I}_{(\boldsymbol{A},\boldsymbol{L_n})}\}$ and material $B$ $\{\boldsymbol{I}_{(\boldsymbol{B},\boldsymbol{L_1})}, \boldsymbol{I}_{(\boldsymbol{B},\boldsymbol{L_2})}, \cdots, \boldsymbol{I}_{(\boldsymbol{B},\boldsymbol{L_n})}\}$, with illuminations $\boldsymbol{L_1}$, $\boldsymbol{L_2}, \cdots, \boldsymbol{L_n}$, encode one-hot feature $\boldsymbol{m}$, hyperparameter $\lambda$.
1. Obtain $\tilde{\boldsymbol{N}}_{(A)}$ and $\tilde{\boldsymbol{N}}_{(B)}$, via training GeometryNet, using Eq. (7);
2. Obtain $\tilde{\boldsymbol{I}}_{(\boldsymbol{A},\boldsymbol{L_i})}$ from $\tilde{\boldsymbol{N}}_{(A)}$, $\tilde{\boldsymbol{I}}_{(\boldsymbol{B},\boldsymbol{L_i})}$ from $\tilde{\boldsymbol{N}}_{(B)}$, via training ReconstructNet, using Eq. (8);
3. Obtain $\tilde{\boldsymbol{I}}_{(\boldsymbol{A},\boldsymbol{L_i})}$ from $\tilde{\boldsymbol{N}}_{(B)}$, $\tilde{\boldsymbol{I}}_{(\boldsymbol{B},\boldsymbol{L_i})}$ from $\tilde{\boldsymbol{N}}_{(A)}$, via training ReconstructNet, using Eq. (9);
4. Obtain $\tilde{\boldsymbol{I}}_{(\boldsymbol{A},\boldsymbol{L_i})}$ and $\tilde{\boldsymbol{I}}_{(\boldsymbol{B},\boldsymbol{L_i})}$ from $\boldsymbol{N}$, via training ReconstructNet, using Eq. (11);
5. Minimize the parameters $\theta_{ge}$, $\theta_{re}$, via the combined loss Eq. (10);
Output: Estimated surface normal map $\tilde{\boldsymbol{N}}_{(A)}$ and $\tilde{\boldsymbol{N}}_{(B)}$, reconstructed images $\tilde{\boldsymbol{I}}_{(\boldsymbol{A},\boldsymbol{L_i})}$ and $\tilde{\boldsymbol{I}}_{(\boldsymbol{B},\boldsymbol{L_i})}$, $i \in \{1, 2, \cdots, n\}$.
end for
```

For $\mathcal{L}_{\text{recon}}$ and $\mathcal{L}_{\text{trans}}$, the hyperparameter λ in Eq. (6) is set at 0.1 during training, to balance the normal loss and the reconstruction loss.

3.6 Training & testing procedures

In our framework, the learning of ReconstructNet requires a surface-normal map as input. However, the normal map predicted by GeometryNet is inaccurate at the beginning of training. Using an inaccurate input will make ReconstructNet converge to an incorrect local minimum.

Therefore, we propose an effective strategy for GR-PSN, which alternately uses the predicted and the ground-truth surface-normal maps as input to ReconstructNet. This strategy can train GeometryNet and ReconstructNet in an end-to-end manner, while ReconstructNet can be trained properly. Concretely, we additionally train ReconstructNet using the ground-truth surface-normal maps $\boldsymbol{N}$ twice, for each training sample: one for generating the rendered image with material A and the other with material B, after $\mathcal{L}_{\text{recon}}$ and $\mathcal{L}_{\text{trans}}$. Actually, the joint loss function $\mathcal{L}$ in Eq. (6) should be written as follows:

$$\mathcal{L} = \mathcal{L}_{\text{normal}} + \lambda(\mathcal{L}_{\text{recon}} + \mathcal{L}_{\text{trans}} + \mathcal{L}_{\text{assist}}), \tag{10}$$

where $\mathcal{L}_{\text{assist}}$ is defined as follows:

$$\begin{aligned}\mathcal{L}_{\text{assist}} = \frac{1}{n}\sum_{i}^{n}(&\|\boldsymbol{N} \rightarrow \tilde{\boldsymbol{I}}_{(\boldsymbol{A},\boldsymbol{L_i})}, \boldsymbol{I}_{(\boldsymbol{A},\boldsymbol{L_i})}\|_2^2 \\ &+\|\boldsymbol{N} \rightarrow \tilde{\boldsymbol{I}}_{(\boldsymbol{B},\boldsymbol{L_i})}, \boldsymbol{I}_{(\boldsymbol{B},\boldsymbol{L_i})}\|_2^2.\end{aligned} \tag{11}$$

The training algorithm for the proposed GR-PSN is summarized in Algorithm 1.

In testing, the trained GR-PSN can take a set of calibrated photometric stereo images with arbitrary materials as input, whether the object has homogeneous material or spatially varying materials. Since ReconstructNet does not extract material information from the input images, but extracts from the specified one-hot feature in testing, *i.e.*, a number (from 0 to 99) representing the specified material is needed, to determine the rendered material of the object. Of course, we can realize multi-material reconstructed images indirectly, via masks for different materials.

Our model was implemented in PyTorch. We use Adam as the optimizer, setting $\beta_1 = 0.9$ and $\beta_2 = 0.999$. The learning rate is initially set to 0.001, and then divided by 2 every 5 epochs. We train the model using a batch size of 96 for 40 epochs, and choose a fixed $n = 32$ as the number of input images, with a spatial resolution $H = W = 32$ (without random crop). It takes about 43.5 hours to train the framework, using a single RTX 3090 GPU with 24 GB memory. It is worth noting that we use the same illumination directions as the input images in training, but any directions can be used in testing. Similarly, the material can also be arbitrary.

4 Experiments

To verify the quantitative accuracy of the predicted surface normals, we use the mean angular error (MAE) in degrees, calculated by MAE = $\frac{1}{U}\sum_p^U \cos^{-1}\left(\tilde{\boldsymbol{N}}^p \odot \boldsymbol{N}^p\right)$, where U is the total number of pixels in the area where the surface normals are considered. For reconstructed images, we adopt the commonly used relative error (REL), calculated as REL = $\frac{1}{nU}\sum_i^n \sum_p^U \frac{\left|\tilde{\boldsymbol{I}}^p_{M,L_i} - \boldsymbol{I}^p_{M,L_i}\right|}{\boldsymbol{I}^p_{M,L_i}}$.

4.1 Datasets

The training synthetic dataset is the same as the widely used setup [15], [17], [21], [38], [39], [41], [45], [46], [57], [58], [59], including two shape datasets, named Blobby [25] and Sculpture [24], rendered by the MERL BRDFs dataset [22] (containing 100 different real-world materials). However, the way of using the datasets is different. In our experiments, we render each object with two randomly selected materials (from the MERL BRDFs dataset), denoted the materials as A and B in Fig. 1.

To evaluate our method, we apply several commonly used real and synthetic datasets. For real datasets, we first test our method on the widely used DiLiGenT benchmark dataset [26] and DiLiGenT10^2 [60], which contains 10 objects and 100 objects with ground truth. Therefore, we can quantitatively compare our GR-PSN with other methods on this dataset. Furthermore, we employ the Light Stage Data Gallery [61], which contains six samples without ground truth. Each object has 253 images under different known illumination directions.

For the synthetic dataset, we use the synthetic objects "Dragon" and "Armadillo" of the synthetic test set [62] in our experiments. These two objects are rendered with 100 different BRDFs from the MERL dataset [22] under 100 random illumination directions in the upper hemisphere. In this synthetic dataset, we can quantitatively test both the accuracy of the estimated surface normals and the rendered images.

4.2 Ablation studies

To quantitatively evaluate the effectiveness of our proposed modules, framework, and training strategy, a standard validation set with 852 samples [15], with all 64 input images and two rendered materials for each sample, is employed. We report the average MAE computed across these 852 samples, marked as MAE (V). It is worth noting that the REL metric includes two parts, REL (O) and REL (C), which represent the REL of the rendered images with the original material and changed material, respectively. We also report the average MAE on the real-photoed DiLiGenT dataset with 10 objects [26], marked as MAE (D). However, we cannot measure the REL metric on the DiLiGenT dataset, because it lacks reconstructed images with changed materials.

TABLE 1
Performance of adding ReconstructNet to form a closed-loop structure, in terms of MAE and REL.

Manners	MAE (V) ↓	REL (O / C) ↓	MAE(D) ↓
Proposed	**5.62**	0.070 / 0.071	**6.55**
w/o ReconstructNet	6.17	- / -	7.03

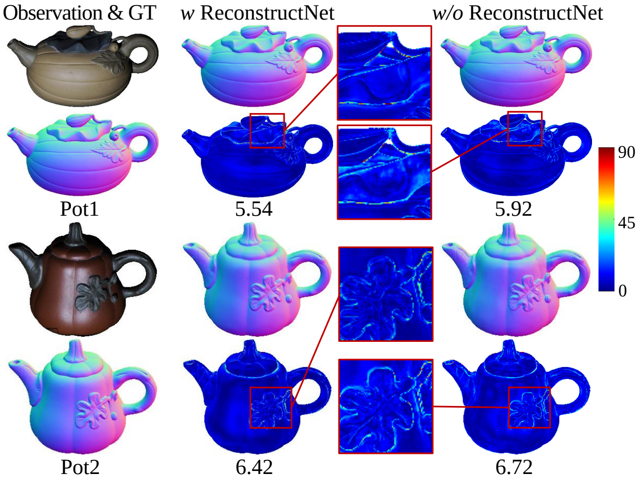


Fig. 4. Visualized examples on the DiLiGenT dataset. The first row of each sample represents the estimated normal maps, while the second row represents the error maps, based on with or without ReconstructNet. The values under error maps are MAE in degrees.

4.2.1 *Effectiveness of ReconstructNet and hyperparameter λ*

We first test the effectiveness of the proposed ReconstructNet (Table 1, Fig. 4) and its different weights for the hyperparameter λ in Eq. (10) (Fig. 5).

As tabulated in Table 1, our GR-PSN obviously outperforms the baseline structure (w/o ReconstructNet), which only uses the constraint of surface normals. This illustrates that ReconstructNet can provide additional supervision on the image domain, and the joint task of image reconstruction can reinforce the learning of normal estimation. Fig. 4 further illustrates the effectiveness of the proposed ReconstructNet. It can be seen that the details in the shadow region and complex structure region (red box in Fig. 4) can be better recovered with the use of full model GR-PSN. This may be due to the fact that the inverse stage, *i.e.*, ReconstructNet, can reduce the original search space and relax the optimization.

As shown in Fig. 5, to determine the optimal hyperparameter λ, we experimentally evaluate our GR-PSN with different values of λ from 0 to 1. It can be seen that the best performance is achieved when λ = 0.1, which achieves the smallest MAE on both the validation set and the DiLiGenT

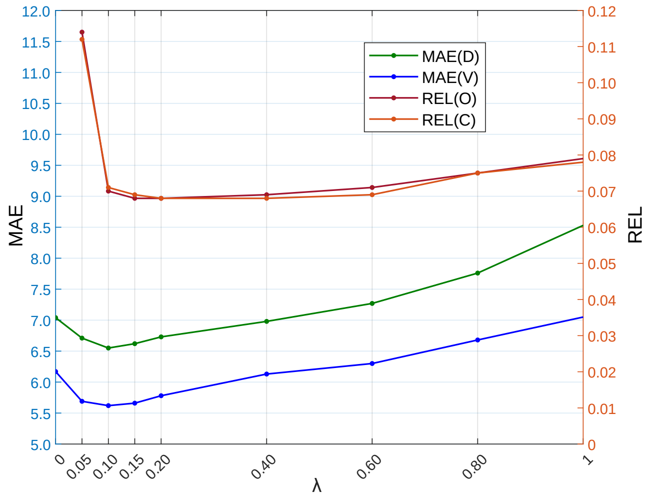


Fig. 5. Results of our GR-PSN, trained with different hyperparameter values λ. The left Y axis represents the MAE of the predicted surface normals, while the right Y axis represents the REL of the rendered images. Note that λ = 0 equals w/o ReconstructNet in Table 1, which is trained with GeometryNet only.

TABLE 2
Performance of using different training strategies, in terms of MAE and REL.

Manners	MAE (V)↓	REL (O / C) ↓	MAE(D) ↓
Proposed	**5.62**	**0.070** / **0.071**	**6.55**
w/o $\mathcal{L}_{\text{assist}}$	5.84	0.75 /0.75	6.87

TABLE 3
Effectiveness of Double-gate normalization, bilateral extraction, top-k fusion, and deep-shallow feature fusion, in terms of MAE and REL.

ID	Manners	MAE (V) ↓	REL (O / C) ↓	MAE(D) ↓
(0)	Proposed	**5.62**	**0.070** / 0.071	**6.55**
(1)	w/o Normalization	6.36	0.073 / 0.072	7.11
(2)	w/o Bilateral extraction	5.75	0.071 / 0.071	6.75
(3)	w/o Top-k fusion	5.69	0.071 / **0.070**	6.70
(4)	w/o Deep-shallow fusion	5.85	0.071 / 0.071	6.92

TABLE 4
Performance on the DiLiGenT benchmark [26] with 96 images, in terms of MAE (degrees). The values in red and blue represent the best performance and the second-best performance, respectively.

Method	Ball	Bear	Buddha	Cat	Cow	Goblet	Harvest	Pot1	Pot2	Reading	Avg.
Least Square [5]	4.10	8.39	14.92	8.41	25.60	18.50	30.62	8.89	14.65	19.80	15.39
IW12 [10]	2.54	7.32	11.11	7.21	25.70	16.25	29.26	7.74	14.09	16.17	13.74
GC10 [63]	3.21	6.62	14.85	8.22	9.55	14.22	27.84	8.53	7.90	19.07	12.00
WG10 [11]	2.06	6.50	10.91	6.73	25.89	15.70	30.01	7.18	13.12	15.39	13.35
IA14 [31]	3.34	7.11	10.47	6.74	13.05	9.71	25.95	6.64	8.77	14.19	10.60
ST14 [8]	1.74	6.12	10.60	6.12	13.93	10.09	25.44	6.51	8.78	13.63	10.30
SPLINE-Net [14]	4.51	5.28	10.36	6.49	7.44	9.62	17.93	8.29	10.89	15.50	9.63
DPSN [12]	2.02	6.31	12.68	6.54	8.01	11.28	16.86	7.05	7.86	15.51	9.41
IRPS [43]	1.47	5.79	10.36	5.44	6.32	11.47	22.59	6.09	7.76	11.03	8.83
LMPS † [38]	2.40	5.23	9.89	6.11	7.98	8.61	16.18	6.54	7.48	13.68	8.41
PS-FCN [41]	2.82	7.55	7.91	6.16	7.33	8.60	15.85	7.13	7.25	13.33	8.39
Attention-PSN [42]	2.93	4.86	7.75	6.14	6.86	8.42	15.44	6.92	6.97	12.90	7.92
LERPS [53]	2.41	6.93	8.84	7.43	6.36	8.78	11.57	8.32	7.01	11.51	7.92
DR-PSN [21]	2.27	5.46	7.84	5.42	7.01	8.49	15.40	7.08	7.21	12.74	7.90
GPS-Net [39]	2.92	5.07	7.77	5.42	6.14	9.00	15.14	6.04	7.01	13.58	7.81
JJ21 [57]	2.51	5.77	7.88	6.56	6.29	8.40	14.95	7.21	7.40	11.01	7.80
CNN-PS † [13]	2.12	8.30	8.07	4.38	7.92	7.42	14.08	5.37	6.38	12.12	7.62
PS-FCN (Norm.) [15]	2.67	7.72	7.53	4.76	6.72	7.84	12.39	6.17	7.15	10.92	7.39
MF-PSN [45]	2.07	5.83	6.88	5.00	5.90	7.46	13.38	7.20	6.81	12.20	7.27
HT21† [40]	2.49	8.96	7.23	4.69	4.89	6.89	12.79	5.10	4.98	11.08	6.91
NormAttention-PSN [17]	2.93	5.48	7.12	4.65	5.99	7.49	12.28	5.96	6.42	9.93	6.83
WZ20 ‡ [58]	1.78	5.26	6.09	4.66	6.33	7.22	13.34	6.46	6.45	10.05	6.76
PX-Net† [16]	2.03	4.13	7.61	4.39	4.69	6.90	13.10	5.08	5.10	10.26	6.33
GR-PSN (Ours)	2.22	5.61	6.73	4.33	6.17	6.78	12.03	5.54	6.42	9.65	6.55

benchmark dataset [26]. It is worth noting that our model is only trained with GeometryNet when the hyperparameter λ = 0, which shows a worse MAE for surface-normal estimation. This reflects the effectiveness of the proposed ReconstructNet, which can provide additional supervision for recovering surface normals. From the REL perspective, the performance continuously improves until the hyperparameter λ is larger than 0.2. This is because a larger weight for reconstruction naturally benefits the learning of reconstructed images. However, the REL metric becomes obviously worse when λ is larger than 0.6. We conjecture that this may be due to the fact that a large weight for the reconstruction loss impacts the accuracy of the estimated surface normal, which in turn affects ReconstructNet.

In fact, the optimal value of the hyperparameter λ also depends on the training strategy. From Eq. (10), in each iteration, we can see that ReconstructNet is trained three times while GeometryNet is only trained once. Therefore, a smaller weight, such as 0.1, is good for stable training.

4.2.2 Effectiveness of training strategy

We evaluate the effectiveness of the proposed training strategy, *i.e.*, alternately using predicted and the ground-truth surface normals as input to ReconstructNet, as described in Section 3.6. As tabulated in Table 2, the proposed GR-PSN uses the additional loss $\mathcal{L}_{\text{assist}}$ as in Eq. (10), while the ablation study discards this loss as in Eq. (6). Note that the experiment results are under the setting of λ = 0.1, as discussed in Section 4.2.1.

The loss function $\mathcal{L}_{\text{assist}}$ for ReconstructNet, with the use of ground-truth surface-normal maps, can be viewed as simple supervised task, guiding ReconstructNet to reach an optimal performance. Actually, it is very difficult to train the entire process, with two cascaded networks. Therefore, we use the easier task ($\mathcal{L}_{\text{assist}}$) to assist the learning of the harder tasks ($\mathcal{L}_{\text{recon}}$ and $\mathcal{L}_{\text{trans}}$). In Table 2, we can see the results indicate that our learning strategy is effective.

4.2.3 Effectiveness of the used modules

As shown in Table 3, we tested the modules used in GeometryNet, as discussed in Sections 3.1.1, 3.2, and 3.3. All experiments in Table 3 are conducted under the setting λ = 0.1 and the proposed training strategy discussed in Sections 4.2.1 and 4.2.2.

For ID (1), we remove the double-gate observation normalization [17]. It achieves lower accuracy in terms of MAE and REL. ID (2) does not use the bilateral extraction module. In this case, the top-k pooling modules only receive n features extracted from n input images. ID (3) represents the method without using top-k pooling, but the original max-pooling [41]. Comparing IDs (2) and (3) with the proposed method, we can see that bilateral extraction and top-k pooling can improve the learning ability of surface-normal estimation to some extent. In ID (4), we analyze the effectiveness of the deep-shallow and global-local multiple-feature fusion framework. For testing, we only retain the deep fusion module, while discarding the shallow fusion module. Therefore, the extraction stage of GeometryNet will be three consecutive encoding units with one deconvolutional layer. Due to the lack of shallow fusion, the fused features are not concatenated with every feature extracted from the input image (without the deep-shallow and global-local multiple-feature fusion framework). It can be found that the performance, in terms of MAE, drops by

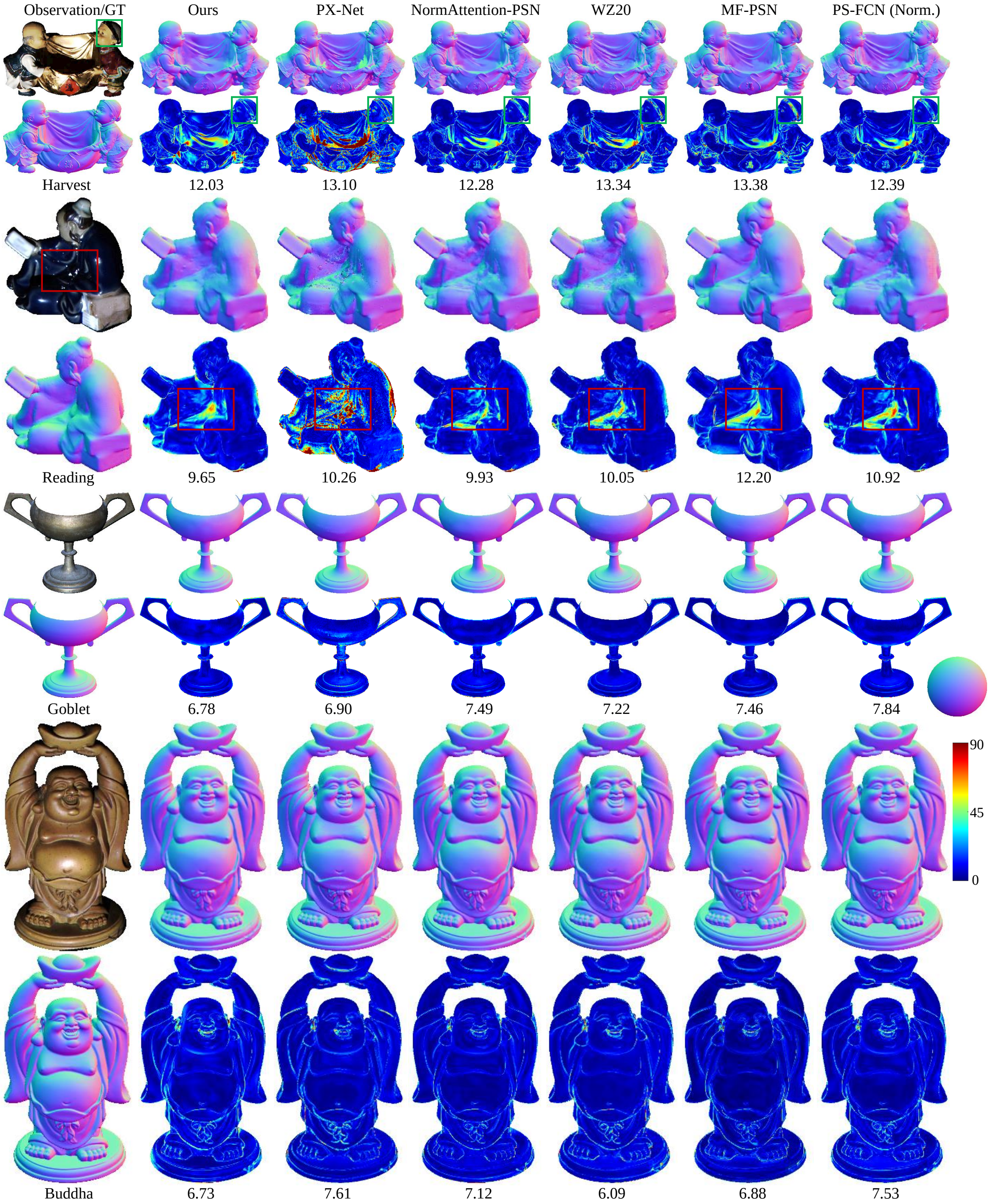


Fig. 6. Quantitative results on the DiLiGenT dataset, with 96 input images. The first row of each sample represents the estimated normal maps, while the second row represents the error maps, based on the different methods. The values represent MAE in degrees. The contrast of the images is adjusted for easy visualization. It can be seen that our GR-PSN achieves better performance on surfaces with cast shadows (red boxes) and spatially varying materials (green boxes).

0.23 degrees and 0.37 degrees on the validation set and the DiLiGenT benchmark dataset [26], respectively. However, the performance in terms of REL, which reflects the accuracy of the reconstructed images, is hardly affected by the ablated GeometryNet modules in IDs (1) to (4).

4.3 Evaluation on the DiLiGenT benchmark

The DiLiGenT benchmark [26] is a widely used real-photoed dataset for photometric stereo. It contains ten real-world objects, and is challenging for its strong non-Lambertian surfaces and complex structures. Each object has 96 images under different illumination directions. Table 4 tabulates the experimental results for all 96 input images, in terms of MAE, of our GR-PSN and other state-of-the-art methods. Most of the learning-based methods are represented by their network names. For non-learning methods and those learning-based methods without a network name, we present them by the first letter of the author's name, followed by the year of publication. We use † to represent the networks trained by CyclePS [13] rendered by Disney's prin-

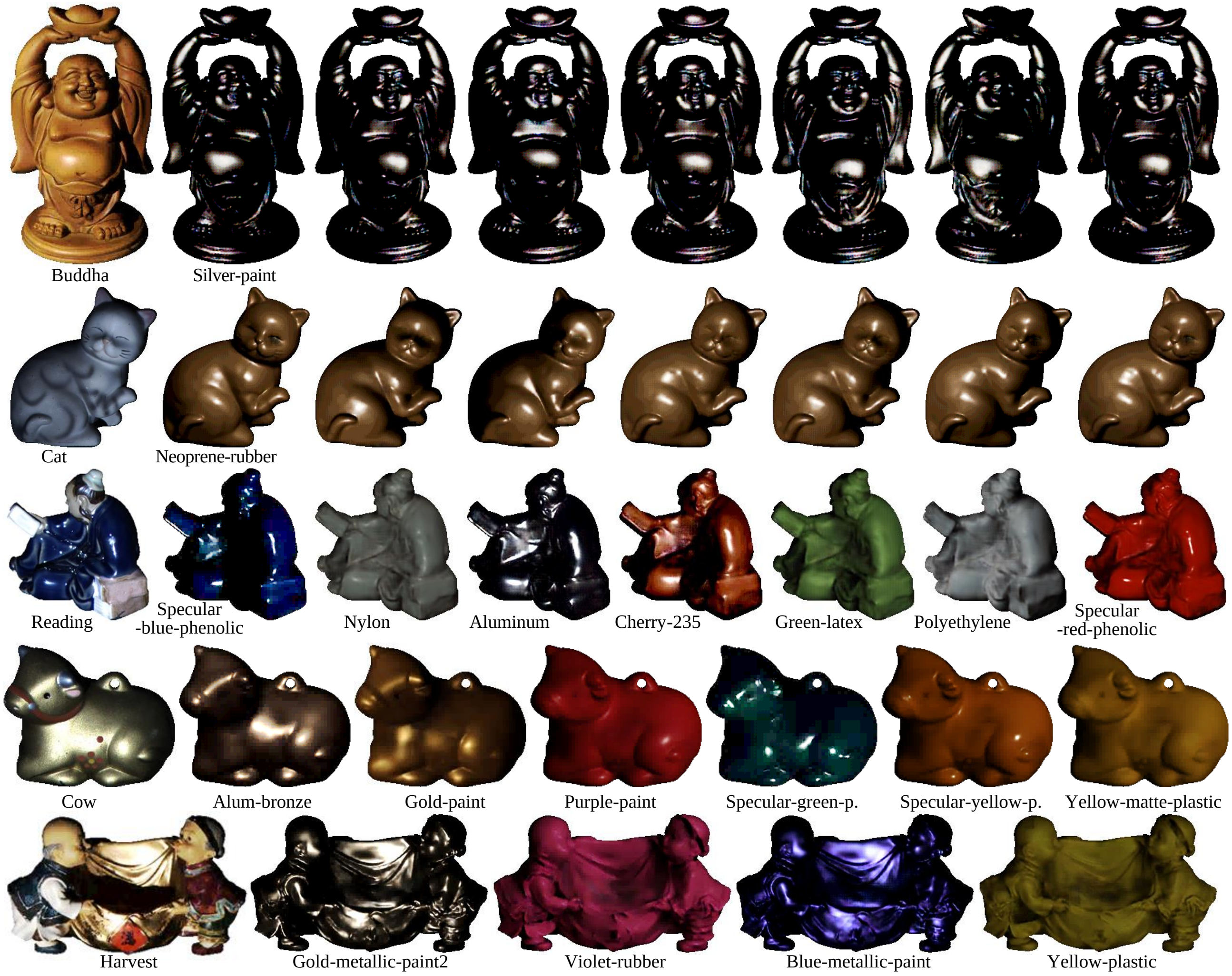


Fig. 7. Results of rendered objects from the DiLiGenT benchmark. The first two rows show the rendered images with different illuminated lights. The last three rows show examples of the objects rendered for different materials. The contrast of the images is adjusted for easy visualization.

cipled BSDFs [64]. For example, PX-Net [16], HT21 [40], and CNN-PS [13] train their models using Disney's principled BSDFs [64], which may lead to better performance because it contains unlimited reflectance. The Disney's principled BSDFs used contains unlimited reflectance, as they integrate different BRDFs controlled by 11 parameters. This makes the reflectance distributions more similar in real-world scenarios. Conversely, the MERL BRDFs dataset contains only 100 kinds of reflectance, which barely span all materials that exist in nature. However, this dataset is not suitable for most all-pixel methods (discussed in Section 2.1), because it is designed for the per-pixel processing strategy rather than the all-pixel networks (the number of samples required is very large for all-pixel methods). We also use ‡ to represent the method WZ20 [58], which uses a specific collocated illumination constraint in the augmented synthetic dataset rendered by the MERL BRDFs dataset [22]. Therefore, the comparison with † and ‡ is not entirely fair.

As shown in Table 4, our method achieves the second-best MAE (state-of-the-art compared to the methods trained with MERL BRDFs dataset [22]), averaged over ten objects. For those complicated and strong non-Lambertian objects, such as "Buddha", "Goblet", "Harvest", and "Reading", which contain shadows and inter-reflections, as shown in Fig. 6, the proposed method achieves the best or the second-best performances. It can be seen that our method achieves better results in those regions with cast shadows (red boxes), such as the "crotch" of the object "Reading", and those regions with spatially varying materials (green boxes), such as the "headband" of the object "Harvest". This can be explained by the fact that our GR-PSN can learn the entire representation of BRDFs, which constrains the learning of surface normals. These results illustrate the effectiveness of our method, which receives additional supervision, performed by ReconstructNet.

Furthermore, Fig. 7 shows some reconstructed images on the DiLiGenT benchmark [26]. We first show the reconstructed images of "Buddha" and "Cat" under different light directions, but keeping the same material. We then show the rendered images of complicated objects with different materials, *i.e.*, "Harvest" and "Reading". It can be seen that specularities and shadows are obvious on those objects with complicated structures when rendered with metal materials, and the reconstructed images are not affected by spatially varying materials on the original surfaces. Although the ground-truth of reconstructed images

TABLE 5
Performance on the DiLiGenT benchmark [26] with different numbers of input images. The values in red and blue represent the best performance and the second-best performance, respectively.

Methods	Number of input images				
	10	16	32	64	96
Least Square [5]	16.10	15.73	15.51	15.42	15.39
SPLINE-Net [14]	10.35	10.12	9.93	9.72	9.63
IRPS [43]	10.79	9.87	9.38	8.98	8.83
LMPS [38]	10.01	9.66	9.38	9.15	8.41
PS-FCN [41]	10.19	9.20	8.74	8.47	8.39
DR-PSN [21]	9.94	9.06	8.32	8.03	7.90
GPS-Net [39]	9.43	8.71	8.05	7.84	7.81
CNN-PS [13]	13.53	10.40	8.18	7.56	7.62
PS-FCN (Norm.) [15]	10.40	8.23	7.59	7.40	7.39
NormAttention-PSN [17]	10.50	7.86	7.08	6.89	6.83
GR-PSN (Ours)	9.98	8.02	7.25	6.86	6.55

with the MERL BRDFs dataset [22] is not available in the DiLiGenT benchmark [26], the re-reconstructed images can display the details of the objects. This demonstrates the accurate performance achieved by ReconstructNet.

4.4 Evaluation with an arbitrary number of input images

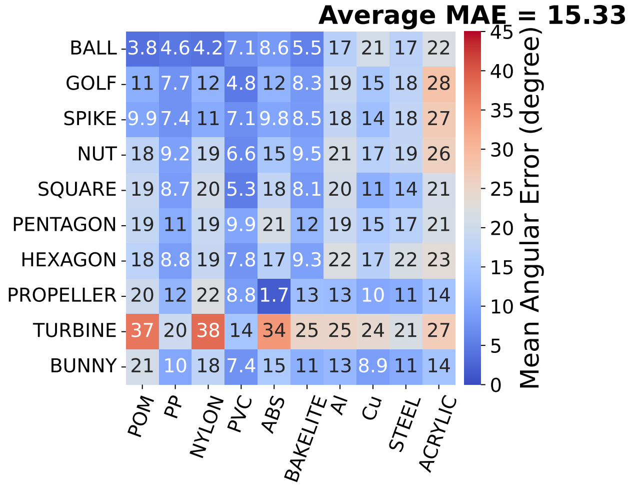
In fact, many practical applications involve sparse photometric stereo. We evaluate our GR-PSN and flexible input methods on the DiLiGenT benchmark using varying numbers of input images. We perform the testing on all ten objects and average the results. Specifically, we train our proposed method with a fixed 32 input images and test with 10, 16, 32, 64, and 96 input images of the DiLiGenT benchmark [26]. The results are shown in Table 5.

As illustrated in Table 5, the proposed GR-PSN method demonstrates superior performance compared to other approaches when utilizing 64 and 96 images. Additionally, it maintains sub-optimal estimation results in cases of 16 and 32 images. These outcomes highlight the effectiveness of our method, particularly in scenarios involving dense input images. Furthermore, GR-PSN keeps a promising performance when 10 images are used. Note that some methods in the comparison were trained using only 10 input images, specifically designed for sparse input conditions, such as LMPS [38] and SPLINE-Net [14]. Similarly, some recent technologies pay more attention to the sparse input photometric stereo with notably improved results, such as PS-Transformer [48] and DeepPS2 [54]. However, it's important to mention that using more than 10 input images in [48] might result in insufficient GPU memory due to large parameter requirements.

4.5 Evaluation on the DiLiGenT10^2 dataset

To conduct an in-depth analysis of our GR-PSN regarding its generalization capability across various objects and materials, we test GR-PSN on the challenging DiLiGenT10^2 dataset [60]. The results acquired by the online evaluation website [1] are shown in Fig. 8.

Referring to DiLiGenT10^2 [60], our GR-PSN achieves the average MAE of 15.33 and outperforms all the reported methods in their paper. These encompass traditional methods Lease square [5] (18.13), TH28 [26] (19.66), TH46 [26] (18.82), WG10 [11], ST14 [8] (18.34), PF14 [65] (30.63), and learning-based methods PS-FCN [41] (16.21), CNN-PS [13] (15.78), IRPS [43] (17.10), SPLINE-Net [14] (16.42), GPS-Net [39] (19.98). It can be seen that GR-PSN achieves the best performance for surface normal estimation, and especially works well for anisotropic materials such as STEEL, Cu, and Al. It owes to the effectiveness of our method, which receives additional supervision, performed by ReconstructNet.



	POM	PP	NYLON	PVC	ABS	BAKELITE	Al	Cu	STEEL	ACRYLIC
BALL	3.8	4.6	4.2	7.1	8.6	5.5	17	21	17	22
GOLF	11	7.7	12	4.8	12	8.3	19	15	18	28
SPIKE	9.9	7.4	11	7.1	9.8	8.5	18	14	18	27
NUT	18	9.2	19	6.6	15	9.5	21	17	19	26
SQUARE	19	8.7	20	5.3	18	8.1	20	11	14	21
PENTAGON	19	11	19	9.9	21	12	19	15	17	21
HEXAGON	18	8.8	19	7.8	17	9.3	22	17	22	23
PROPELLER	20	12	22	8.8	1.7	13	13	10	11	14
TURBINE	37	20	38	14	34	25	25	24	21	27
BUNNY	21	10	18	7.4	15	11	13	8.9	11	14

Fig. 8. Shape-material error matrix for our GR-PSN. A number in each element of the matrix indicates an MAE in degrees according to a shape and material index.

4.6 Evaluation on the synthetic test data

Due to the lack of quantitative results of reconstructed images in the DiLiGenT benchmark [26], we use the synthetic test data from the Stanford 3D dataset [62]. Each type of material of the objects "Dragon" and "Armadillo" have 100 random illumination directions in the upper hemisphere. Fig. 9 shows the MAE of predicted normal maps of "Dragon" and "Armadillo" under 100 kinds of materials rendered by MERL BRDFs [22]. Fig. 10 further shows the REL of rendered images of "Dragon" and "Armadillo", under 100 kinds of materials.

As shown in Fig. 9, our method achieves promising results under 100 kinds of materials, with the average MAE of 5.63° and 6.11° for the objects "Dragon" and "Armadillo", respectively. It can be seen that the average MAE of the object "Armadillo" is larger than "Dragon", which may be caused by the more complex surface structure. In contrast, as shown in Fig. 10, the average REL of the re-rendered images are 0.072 and 0.081 for "Dragon" and "Armadillo", respectively. It shows more similar performance for the reconstructed images, which illustrates the robustness of ReconstructNet in the proposed GR-PSN.

4.7 Evaluation on the Light Stage Data Gallery

We further evaluated our method on the more complex Light Stage Data Gallery [61], with general non-Lambertian materials. Due to the lack of ground truth, we show in

[1] The online evaluation website can be found in https://photometricstereo.github.io/diligent102.html

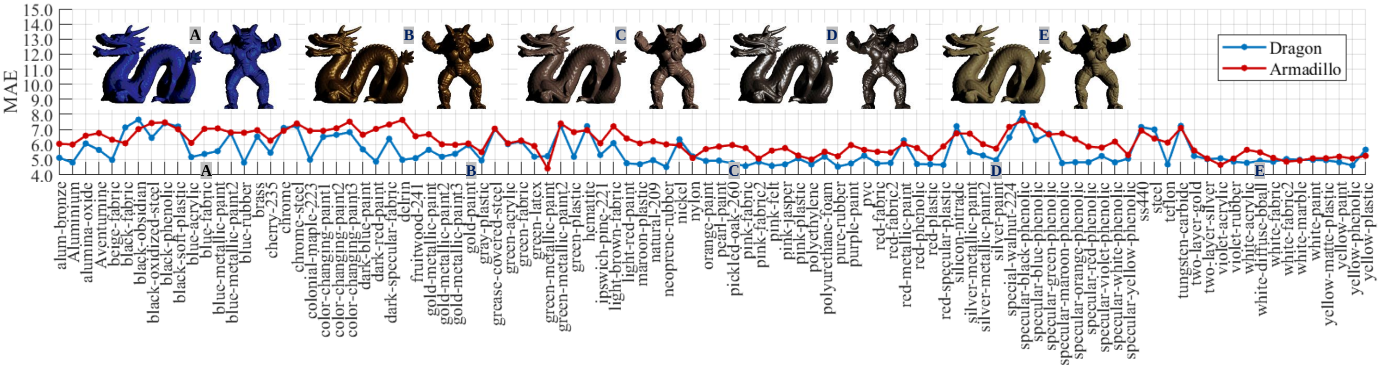


Fig. 9. MAE of the predicted surface normals for the samples “Dragon” and “Armadillo” with 100 kinds of materials in MERL BRDF [22]. Some examples are shown in the upper-left corner.

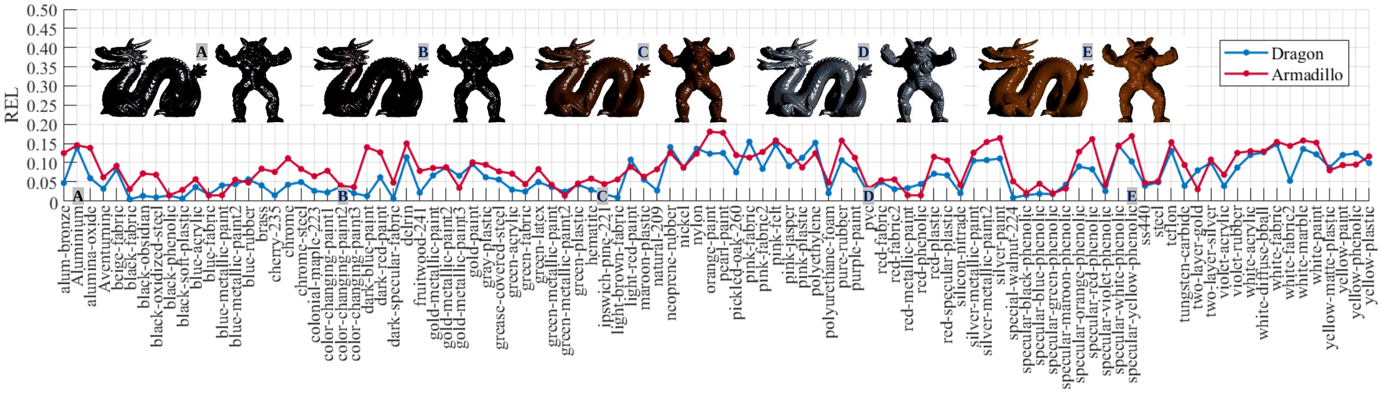


Fig. 10. REL of the reconstructed images for the samples “Dragon” and “Armadillo” with 100 kinds of materials in MERL BRDF [22]. Some examples are shown in the upper-left corner.

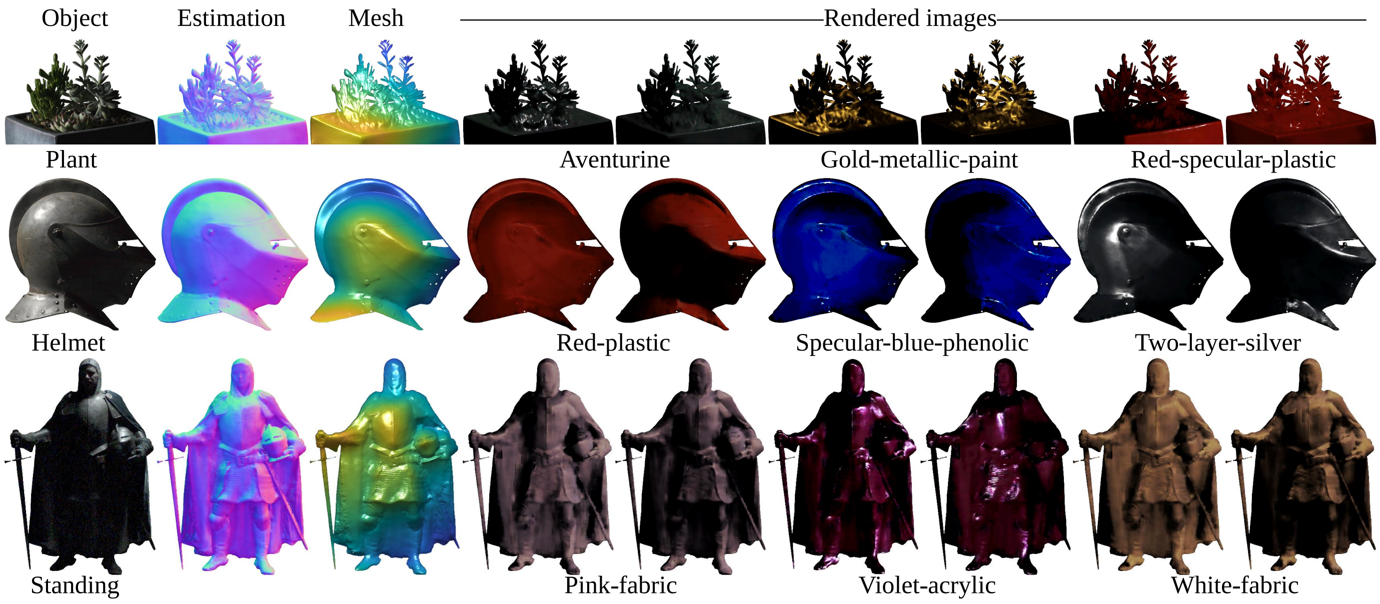


Fig. 11. Evaluation on the Light Stage Data Gallery, with 64 input images. We qualitatively show estimated surface normals and reconstructed images with arbitrary materials. Due to the lack of ground truth of the surface normals, we further show the 3D reconstruction results of our estimated surface normal maps using [66], to clearly show the details. The contrast of the images is adjusted for easy visualization.

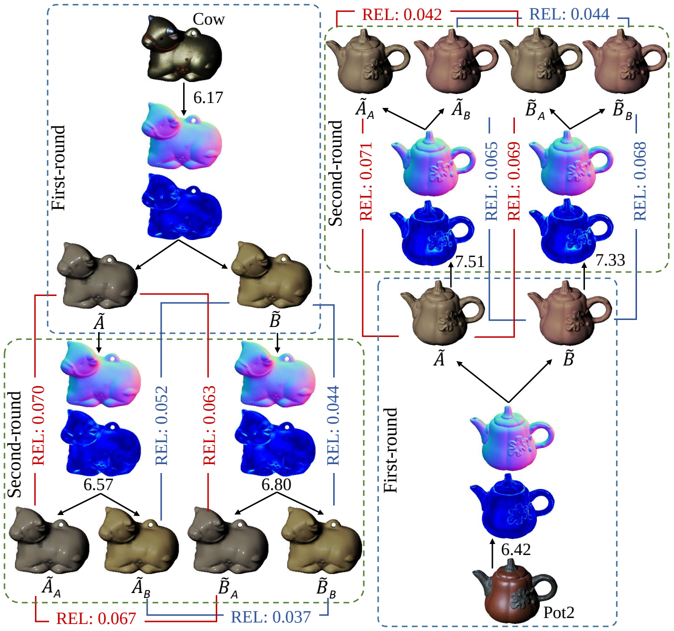


Fig. 12. Results of the multi-time reconstructed images and surface-normal maps. For the object "Cow", material A is set to Alumina-oxide, and material B is set to White-diffuse-bball, while for the object "Pot2", the materials A and B are Neoprene-rubber and Pink-felt, respectively. In the second-round of rendering, the subscript represents the material of the first-round rendering.

Fig. 11 the qualitative results of our method, on estimating surface normals and reconstructing images. Similar to the above experiments, our GR-PSN was trained with 32 input images, and tested with 64 input images randomly selected from all 253 images of the Light Stage Data Gallery. Note that the objects "Helmet", "Plant", and "Fighting" are down-sampled to half the spatial resolution, because the original resolution (1024 × 1024) is too large to process.

As shown in Fig. 11, the estimated surface normals and their Mesh reconstructions can accurately report the shapes of the objects, such as the screws of the object "Helmet", and the fiber skirt of the object "Standing". However, we also observe that some noise exists on the cloak of the object "Standing". The compromised surface normal estimations can be attributed to the down-sampling of the input images and camera noise resulting from high sensitivity (ISO) settings when capturing images in low-light conditions. Furthermore, it can be seen that the reconstructed images with different surface materials show reasonable highlights and shadows. The details are still clear in the reconstructed images, such as the leaves of the object "plant", and the rivet of the object "Helmet". We note that the reconstructed images with some metal-like materials are quite dark, which appears to lack realism (also see the reconstructed images of the object "Buddha" in Fig. 7). This is because our ReconstructNet generates the reconstructed images from the surface normals, encoded materials, and light directions explicitly, without the impact of global noise. In contrast, a real-photoed cannot totally avoid the global noise, such as natural illuminations and inter-reflections from other objects (*e.g.*, the background), which may influence the calculation of surface normals.

TABLE 6
Performance on the DiLiGenT dataset [26] using 96 images, in terms of average MAE across ten objects. We conduct tests with our GR-PSN and PS-FCN (Norm.) [15], which are trained with full and few samples respectively, using different augmented rendered data via our method.

	GR-PSN				PS-FCN (Norm.) [15]			
Augmentation	-	50%↑	100%↑	300%↑	-	50%↑	100%↑	300%↑
Full (84,360)	6.55	6.55	6.56	6.59	7.39	7.38	7.42	7.44
Few (844)	8.05	7.50	7.44	7.41	8.66	8.18	8.12	8.02

4.8 Further exploration

In the experiments, it is interesting to note the performance of using the reconstructed images as the inputs of our model. The results are illustrated in Fig. 12, which show the predicted normal map and rendered images based on our method, by using the reconstructed images of the objects "Cow" and "Pot2" as the inputs. As shown in Fig. 12, different generation paths may cause different results (*e.g.*, for simple description, the two different materials are denoted as A and B): original → $\tilde{A}$ → $\widetilde{A_A}$, original → $\tilde{A}$ → $\widetilde{B_A}$, original → $\tilde{B}$ → $\widetilde{A_B}$, and original → $\tilde{B}$ → $\widetilde{B_B}$. Compared with the predicted surface-normal maps with the original images, our surface-normal maps, generated by inputting the reconstructed images in the second round, only show slight differences. Furthermore, the REL between rendered images with the same material (from different paths, *i.e.*, $\tilde{A}$ and $\widetilde{A_A}$, $\widetilde{A_A}$ and $\widetilde{A_B}$, $\tilde{B}$ and $\widetilde{B_A}$, $\widetilde{B_B}$ and $\widetilde{B_A}$) is promising.

With the results of Fig. 12, we further test the results of the re-rendered images. We first use GR-PSN to render images with new random materials in the training dataset. Specifically, we augmented the training dataset from each object with two materials to three (50%↑), four (100%↑), and eight (300%↑) materials in MERL BRDFs [22], while maintaining the original lighting conditions across the rendered images. Note that the original training dataset already contains a substantial 84,360 samples, which is generally sufficient for training a photometric stereo network. Therefore, we also randomly selected 1% of the original samples (844) to examine the augmented training dataset via GR-PSN. In cases where the training data is scarce, the training epochs are adjusted to 400, and the learning rate is halved every 40 epochs. Notably, the number of epochs across different augmented datasets is also adjusted to ensure the equivalent total training processing. Detailed results can be found in Table 6.

As shown in Table 6, the augmented training dataset (few samples) effectively enhances the surface normal estimation performance for both GR-PSN and PS-FCN (Norm.) [15]. It demonstrates the augmented samples rendered through our method contribute to a more diverse representation of materials, which may mitigate overfitting and help the training process on the condition of inadequate training samples. However, the augmented training dataset with full samples shows little changes. This could be explained by the fact that the training samples are already sufficient to optimize both GR-PSN and PS-FCN (Norm.) [15] while bringing additional rendered samples does not increase the knowledge of surface normals but potentially introducing errors in shading cues. The experiments described above indicate that the re-rendered samples generated by GR-PSN could potentially benefit the learning process of networks when the available training samples are insufficient.

5 Conclusion

In this paper, we have proposed a cascaded framework, for learning both the surface normals and the reconstructed images of objects, using the proposed GeometryNet and ReconstructNet, respectively. ReconstructNet provides additional supervision for surface-normal estimation, forming a closed-loop structure, which can improve the performance of both tasks. Furthermore, our method can render images with different materials. The ablation studies illustrate the effectiveness of the additional ReconstructNet, as well as our network architecture. Extensive experiments on the most widely used DiLiGenT benchmark dataset have demonstrated that our method outperforms other calibrated photometric stereo methods. Experiments on the synthetic test data and the real-photoed datasets also show that our method can produce photorealistic rendered photometric images.

A future work of our method is that all trained samples are rendered by the MERL dataset [22], which only includes 100 kinds of real-world materials, hardly spans all materials existing in nature. Therefore, how to expand our method into the dataset with more materials in the natural world and whether it can further improve the performance of GR-PSN is still an open question.

Acknowledgments

The work was supported by the National Key Scientific Instrument and Equipment Development Projects of China (41927805), the National Natural Science Foundation of China (62136001, 62372306), and the Project of Strategic Importance Fund from The Hong Kong Polytechnic University (No. ZE1X)

References

[1] Jundan Luo, Zhaoyang Huang, Yijin Li, Xiaowei Zhou, Guofeng Zhang, and Hujun Bao, "Niid-net: adapting surface normal knowledge for intrinsic image decomposition in indoor scenes," *IEEE Transactions on Visualization and Computer Graphics*, vol. 26, no. 12, pp. 3434–3445, 2020.

[2] Ganzhangqin Yuan, Qiancheng Fu, Zhenxing Mi, Yiming Luo, and Wenbing Tao, "Ssrnet: Scalable 3d surface reconstruction network," *IEEE Transactions on Visualization and Computer Graphics*, 2022.

[3] Xingbin Yang, Liyang Zhou, Hanqing Jiang, Zhongliang Tang, Yuanbo Wang, Hujun Bao, and Guofeng Zhang, "Mobile3drecon: real-time monocular 3d reconstruction on a mobile phone," *IEEE Transactions on Visualization and Computer Graphics*, vol. 26, no. 12, pp. 3446–3456, 2020.

[4] Jingbo Zhang, Ziyu Wan, and Jing Liao, "Adaptive joint optimization for 3d reconstruction with differentiable rendering," *IEEE Transactions on Visualization and Computer Graphics*, 2022.

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

[6] Tian-Qi Han and Hui-Liang Shen, "Photometric stereo for general brdfs via reflection sparsity modeling," *IEEE Transactions on Image Processing*, vol. 24, no. 12, pp. 4888–4903, 2015.

[7] Qian Zheng, Ajay Kumar, Boxin Shi, and Gang Pan, "Numerical reflectance compensation for non-lambertian photometric stereo," *IEEE Transactions on Image Processing*, vol. 28, no. 7, pp. 3177–3191, 2019.

[8] Boxin Shi, Ping Tan, Yasuyuki Matsushita, and Katsushi Ikeuchi, "Bi-polynomial modeling of low-frequency reflectances," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 36, no. 6, pp. 1078–1091, 2014.

[9] Kevin HM Cheng and Ajay Kumar, "Revisiting outlier rejection approach for non-lambertian photometric stereo," *IEEE Transactions on Image Processing*, vol. 28, no. 3, pp. 1544–1555, 2018.

[10] Satoshi Ikehata, David Wipf, Yasuyuki Matsushita, and Kiyoharu Aizawa, "Robust photometric stereo using sparse regression," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2012, pp. 318–325.

[11] Lun Wu, Arvind Ganesh, Boxin Shi, Yasuyuki Matsushita, Yongtian Wang, and Yi Ma, "Robust photometric stereo via low-rank matrix completion and recovery," in *Proceedings of the Asian Conference on Computer Vision*, 2010, pp. 703–717.

[12] Hiroaki Santo, Masaki Samejima, Yusuke Sugano, Boxin Shi, and Yasuyuki Matsushita, "Deep photometric stereo network," in *Proceedings of the IEEE International Conference on Computer Vision Workshops*, 2017, pp. 501–509.

[13] Satoshi Ikehata, "Cnn-ps: Cnn-based photometric stereo for general non-convex surfaces," in *Proceedings of the European Conference on Computer Vision*, 2018, pp. 3–18.

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

[15] Guanying Chen, Kai Han, Boxin Shi, Yasuyuki Matsushita, and Kwan-Yee Kenneth Wong, "Deep photometric stereo for non-lambertian surfaces," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 44, no. 1, pp. 129–142, 2020.

[16] Fotios Logothetis, Ignas Budvytis, Roberto Mecca, and Roberto Cipolla, "Px-net: Simple and efficient pixel-wise training of photometric stereo networks," in *Proceedings of the International Conference on Computer Vision*, 2021, pp. 12757–12766.

[17] Yakun Ju, Boxin Shi, Muwei Jian, Lin Qi, Junyu Dong, and Kin-Man Lam, "Normattention-psn: A high-frequency region enhanced photometric stereo network with normalized attention," *International Journal of Computer Vision*, 2022.

[18] Anpei Chen, Minye Wu, Yingliang Zhang, Nianyi Li, Jie Lu, Shenghua Gao, and Jingyi Yu, "Deep surface light fields," *Proceedings of the ACM on Computer Graphics and Interactive Techniques*, vol. 1, no. 1, pp. 1–17, 2018.

[19] Justus Thies, Michael Zollhöfer, and Matthias Nießner, "Deferred neural rendering: Image synthesis using neural textures," *ACM Transactions on Graphics*, vol. 38, no. 4, pp. 1–12, 2019.

[20] Cen Wang, Minye Wu, Ziyu Wang, Liao Wang, Hao Sheng, and JIngyi Yu, "Neural opacity point cloud," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 42, no. 7, pp. 1570–1581, 2020.

[21] Yakun Ju, Junyu Dong, and Sheng Chen, "Recovering surface normal and arbitrary images: A dual regression network for photometric stereo," *IEEE Transactions on Image Processing*, vol. 30, pp. 3676–3690, 2021.

[22] Wojciech Matusik, Hanspeter Pfister, Matt Brand, and Leonard McMillan, "A data-driven reflectance model," *ACM Transactions on Graphics*, pp. 759–769, 2003.

[23] Mehdi Mirza and Simon Osindero, "Conditional generative adversarial nets," *arXiv preprint arXiv:1411.1784*, 2014.

[24] Olivia Wiles and Andrew Zisserman, "Silnet: Single-and multi-view reconstruction by learning from silhouettes," in *Proceedings of the British Machine Vision Conference*, 2017, pp. 99.1–99.13.

[25] Micah K Johnson and Edward H Adelson, "Shape estimation in natural illumination," in *Proceedings of the IEEE International Conference on Computer Vision*, 2011, pp. 2553–2560.

[26] Boxin Shi, Zhipeng Mo, Zhe Wu, Dinglong Duan, Sai-Kit Yeung, and Ping Tan, "A benchmark dataset and evaluation for non-lambertian and uncalibrated photometric stereo.," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 41, no. 2, pp. 271–284, 2019.

[27] Guanying Chen, Michael Waechter, Boxin Shi, Kwan-Yee K Wong, and Yasuyuki Matsushita, "What is learned in deep uncalibrated photometric stereo?," in *European Conference on Computer Vision*. Springer, 2020, pp. 745–762.

[28] Charles-Félix Chabert, Per Einarsson, Andrew Jones, Bruce Lamond, Wan-Chun Ma, Sebastian Sylwan, Tim Hawkins, and Paul

Debevec, "Relighting human locomotion with flowed reflectance fields," in *ACM SIGGRAPH 2006 Sketches*, pp. 76–es. 2006.

[29] Frank Verbiest and Luc Van Gool, "Photometric stereo with coherent outlier handling and confidence estimation," in *2008 IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, 2008, pp. 1–8.

[30] Daisuke Miyazaki, Kenji Hara, and Katsushi 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, 2010.

[31] Satoshi Ikehata and Kiyoharu Aizawa, "Photometric stereo using constrained bivariate regression for general isotropic surfaces," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2014, pp. 2179–2186.

[32] Manmohan Chandraker, Jiamin Bai, and Ravi Ramamoorthi, "On differential photometric reconstruction for unknown, isotropic brdfs," *IEEE transactions on pattern analysis and machine intelligence*, vol. 35, no. 12, pp. 2941–2955, 2012.

[33] Dehuan Zhang, Jingchun Zhou, Weishi Zhang, Zifan Lin, Jian Yao, Kemal Polat, Fayadh Alenezi, and Adi Alhudhaif, "Rex-net: A reflectance-guided underwater image enhancement network for extreme scenarios," *Expert Systems with Applications*, p. 120842, 2023.

[34] Jingchun Zhou, Boshen Li, Dehuan Zhang, Jieyu Yuan, Weishi Zhang, Zhanchuan Cai, and Jinyu Shi, "Ugif-net: An efficient fully guided information flow network for underwater image enhancement," *IEEE Transactions on Geoscience and Remote Sensing*, 2023.

[35] Yakun Ju, Muwei Jian, Cong Wang, Cong Zhang, Junyu Dong, and Kin-Man Lam, "Estimating high-resolution surface normals via low-resolution photometric stereo images," *IEEE Transactions on Circuits and Systems for Video Technology*, 2023.

[36] Qian Zheng, Boxin Shi, and Gang Pan, "Summary study of data-driven photometric stereo methods," *Virtual Reality & Intelligent Hardware*, vol. 2, no. 3, pp. 213–221, 2020.

[37] Yakun Ju, Kin-Man Lam, Wuyuan Xie, Huiyu Zhou, Junyu Dong, and Boxin Shi, "Deep learning methods for calibrated photometric stereo and beyond: A survey," *arXiv preprint arXiv:2212.08414*, 2022.

[38] Junxuan Li, Antonio Robles-Kelly, Shaodi You, and Yasuyuki Matsushita, "Learning to minify photometric stereo," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019, pp. 7568–7576.

[39] Zhuokun Yao, Kun Li, Ying Fu, Haofeng Hu, and Boxin Shi, "Gps-net: Graph-based photometric stereo network," *Proceedings of the Advances in Neural Information Processing Systems*, vol. 33, 2020.

[40] David Honzátko, Engin Türetken, Pascal Fua, and L Andrea Dunbar, "Leveraging spatial and photometric context for calibrated non-lambertian photometric stereo," in *Proceedings of the International Conference on 3D Vision*, 2021, pp. 394–402.

[41] Guanying Chen, Kai Han, and Kwan-Yee K Wong, "Ps-fcn: A flexible learning framework for photometric stereo," in *Proceedings of the European Conference on Computer Vision*, 2018, pp. 3–18.

[42] Yakun Ju, Kin-Man Lam, Yang Chen, Lin Qi, and Junyu Dong, "Pay attention to devils: A photometric stereo network for better details," in *Proceedings of the International Joint Conference on Artificial Intelligence*, 2020, pp. 694–700.

[43] Tatsunori Taniai and Takanori Maehara, "Neural inverse rendering for general reflectance photometric stereo," in *Proceedings of the International Conference on Machine Learning*, 2018, pp. 4857–4866.

[44] Yakun Ju, Muwei Jian, Junyu Dong, and Kin-Man Lam, "Learning photometric stereo via manifold-based mapping," in *Proceedings of the IEEE International Conference on Visual Communications and Image Processing*, 2020, pp. 411–414.

[45] Yanru Liu, Yakun Ju, Muwei Jian, Feng Gao, Yuan Rao, Yeqi Hu, and Junyu Dong, "A deep-shallow and global–local multi-feature fusion network for photometric stereo," *Image and Vision Computing*, vol. 118, pp. 104368, 2022.

[46] Huiyu Liu, Yunhui Yan, Kechen Song, and Han Yu, "Sps-net: Self-attention photometric stereo network," *IEEE Transactions on Instrumentation and Measurement*, vol. 70, pp. 1–13, 2020.

[47] Yakun Ju, Yuxin Peng, Muwei Jian, Feng Gao, and Junyu Dong, "Learning conditional photometric stereo with high-resolution features," *Computational Visual Media*, vol. 8, pp. 105–118, 2022.

[48] Satoshi Ikehata, "Ps-transformer: Learning sparse photometric stereo network using self-attention mechanism," in *Proceedings of the British Machine Vision Conference*, 2021, vol. 2, p. 11.

[49] Satoshi Ikehata, "Universal photometric stereo network using global lighting contexts," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2022, pp. 12591–12600.

[50] Ye Yu and William AP Smith, "Inverserendernet: Learning single image inverse rendering," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019, pp. 3155–3164.

[51] Soumyadip Sengupta, Angjoo Kanazawa, Carlos D Castillo, and David W Jacobs, "Sfsnet: Learning shape, reflectance and illuminance of facesin the wild'," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 6296–6305.

[52] Shangzhe Wu, Christian Rupprecht, and Andrea Vedaldi, "Unsupervised learning of probably symmetric deformable 3d objects from images in the wild," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2020, pp. 1–10.

[53] Ashish Tiwari and Shanmuganathan Raman, "Lerps: lighting estimation and relighting for photometric stereo," in *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing*, 2022, pp. 2060–2064.

[54] Ashish Tiwari and Shanmuganathan Raman, "Deepps2: Revisiting photometric stereo using two differently illuminated images," in *Proceedings of the European Conference on Computer Vision*, 2022, pp. 129–145.

[55] Zi-Rong Jin, Liang-Jian Deng, Tian-Jing Zhang, and Xiao-Xu Jin, "Bam: Bilateral activation mechanism for image fusion," in *Proceedings of the ACM International Conference on Multimedia*, 2021, pp. 4315–4323.

[56] Gao Huang, Zhuang Liu, Laurens Van Der Maaten, and Kilian Q Weinberger, "Densely connected convolutional networks," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 4700–4708.

[57] Yakun Ju, Muwei Jian, Shaoxiang Guo, Yingyu Wang, Huiyu Zhou, and Junyu Dong, "Incorporating lambertian priors into surface normals measurement," *IEEE Transactions on Instrumentation and Measurement*, vol. 70, pp. 1–13, 2021.

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

[59] Yanlong Cao, Binjie Ding, Zewei He, Jiangxin Yang, Jingxi Chen, Yanpeng Cao, and Xin Li, "Learning inter-and intraframe representations for non-lambertian photometric stereo," *Optics and Lasers in Engineering*, vol. 150, pp. 106838, 2022.

[60] Jieji Ren, Feishi Wang, Jiahao Zhang, Qian Zheng, Mingjun Ren, and Boxin Shi, "Diligent102: A photometric stereo benchmark dataset with controlled shape and material variation," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2022, pp. 12581–12590.

[61] Per Einarsson, Charles-Felix Chabert, Andrew Jones, Wan-Chun Ma, Bruce Lamond, Tim Hawkins, Mark Bolas, Sebastian Sylwan, and Paul Debevec, "Relighting human locomotion with flowed reflectance fields," in *Proceedings of the Eurographics conference on Rendering Techniques*, 2006, pp. 183–194.

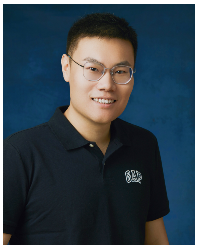
[62] Brian Curless and Marc Levoy, "A volumetric method for building complex models from range images," in *Proceedings of the SIGGRAPH*, 1996, pp. 303–312.

[63] Dan B Goldman, Brian Curless, Aaron Hertzmann, and Steven 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.

[64] Stephen McAuley, Stephen Hill, Naty Hoffman, Yoshiharu Gotanda, Brian Smits, Brent Burley, and Adam Martinez, "Practical physically-based shading in film and game production," in *ACM SIGGRAPH 2012 Courses*, pp. 1–7. 2012.

[65] Thoma Papadhimitri and Paolo Favaro, "A closed-form, consistent and robust solution to uncalibrated photometric stereo via local diffuse reflectance maxima," *International journal of computer vision*, vol. 107, pp. 139–154, 2014.

[66] Tal Simchony, Rama Chellappa, and Min Shao, "Direct analytical methods for solving poisson equations in computer vision problems," *IEEE transactions on pattern analysis and machine intelligence*, vol. 12, no. 5, pp. 435–446, 1990.



Yakun Ju is currently a Research Fellow of the Rapid-Rich Object Search (ROSE) Lab, School of Electrical and Electronic Engineering, Nanyang Technological University. Kot. Before that, he was the Postdoctoral Fellow of the Department of Electronic and Informa Engineering, The Hong Kong Polytechnic University. He received the B.Eng. degree from Sichuan University, Chengdu, China, in 2016 and Ph.D. degree from Ocean University of China, Qingdao, China, in 2022. His research interests include computational photography, 3D reconstruction, low-level vision, and image processing.

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 Lab. Before joining PKU, he did research with MIT Media Lab, Singapore University of Technology and Design, Nanyang Technological University, National Institute of Advanced Industrial Science and Technology, from 2013 to 2017. His papers were awarded as Best Paper Runner-Up at ICCP 2015 and selected as Best Papers from ICCV 2015 for IJCV Special Issue. He is an associate editor of TPAMI/IJCV and an area chair of CVPR/ICCV. He is a senior member of IEEE.

Yang Chen received the B.Eng. degree in computer science and technology from Ocean University of China in 2021, and is currently studying for a master's degree in Tsinghua University. His research interests include 3D reconstruction, medical image processing, and weak supervised learning.

Huiyu Zhou received a Bachelor of Engineering degree in Radio Technology from Huazhong University of Science and Technology of China, and a Master of Science degree in Biomedical Engineering from University of Dundee of United Kingdom, respectively. He was awarded a Doctor of Philosophy degree in Computer Vision from Heriot-Watt University, Edinburgh, United Kingdom. Dr. Zhou currently is a full Professor at School of Computing and Mathematical Sciences, University of Leicester, United Kingdom. He has published over 400 peer-reviewed papers in the field. His research work has been or is being supported by UK EPSRC, ESRC, AHRC, MRC, EU, Royal Society, Leverhulme Trust, Invest NI, Puffin Trust, Alzheimer's Research UK, Invest NI and industry.

Junyu Dong received the B.Sc. and M.Sc. degrees from the Department of Applied Mathematics, Ocean University of China, Qingdao, China, in 1993 and 1999, respectively, and the Ph.D. degree in image processing from the Department of Computer Science, Heriot-Watt University, U.K., in 2003. He joined Ocean University of China in 2004. He is currently a Professor and the Dean of the Faculty of Information Science and Engineering, Ocean University of China. His research interests include computer vision, underwater image processing, and machine learning, with more than ten research projects supported by the NSFC, MOST, and other funding agencies.

Kin-Man Lam received his Associateship in Electronic Engineering with distinction from The Hong Kong Polytechnic University (formerly called Hong Kong Polytechnic) in 1986, his M.Sc. degree in communication engineering from the Department of Electrical Engineering, Imperial College, U.K., in 1987, and his Ph.D. degree from the Department of Electrical Engineering, University of Sydney, Australia, in 1996. From 1990 to 1993, he was a lecturer at the Department of Electronic Engineering of The Hong Kong Polytechnic University. He joined the Department of Electronic and Information Engineering, The Hong Kong Polytechnic University again as an Assistant Professor in October 1996. He became an Associate Professor in 1999, and has been a Professor since 2010. Currently, he is also an Associate Dean of the Faculty of Engineering. He was actively involved in professional activities. He has been a member of the organizing committee or program committee of many international conferences. Prof. Lam was the Chairman of the IEEE Hong Kong Chapter of Signal Processing between 2006 and 2008, and was the Director-Student Services and the Director-Membership Services of the IEEE SPS between 2012 and 2014, and between 2015 and 2017, respectively. He was also the VP-Member Relations and Development and VP-Publications of the Asia-Pacific Signal and Information Processing Association (APSIPA) between 2014 and 2017, and between 2017 and 2021, respectively. He was an Associate Editor of IEEE Trans. on Image Processing between 2009 and 2014, and Digital Signal Processing between 2014 and 2018. He was also an Editor of HKIE Transactions between 2013 and 2018, and an Area Editor of the IEEE Signal Processing Magazine between 2015 and 2017. Currently, he is the IEEE SPS VP-Membership and the Member-at-Large of APSIPA. Prof. Lam also serves as a Senior Editorial Board member of APSIPA Trans. on Signal and Information Processing and an Associate editor of EURASIP International Journal on Image and Video Processing. His current research interests include image and video processing, computer vision, and human face analysis and recognition.