Deep Photometric Stereo for Non-Lambertian Surfaces

Guanying Chen, Kai Han, Boxin Shi, Yasuyuki Matsushita, and Kwan-Yee K. Wong

Abstract—This paper addresses the problem of photometric stereo, in both calibrated and uncalibrated scenarios, for non-Lambertian surfaces based on deep learning. We first introduce a fully convolutional deep network for calibrated photometric stereo, which we call PS-FCN. Unlike traditional approaches that adopt simplified reflectance models to make the problem tractable, our method directly learns the mapping from reflectance observations to surface normal, and is able to handle surfaces with general and unknown isotropic reflectance. At test time, PS-FCN takes an arbitrary number of images and their associated light directions as input and predicts a surface normal map of the scene in a fast feed-forward pass. To deal with the uncalibrated scenario where light directions are unknown, we introduce a new convolutional network, named LCNet, to estimate light directions from input images. The estimated light directions and the input images are then fed to PS-FCN to determine the surface normals. Our method does not require a pre-defined set of light directions and can handle multiple images in an order-agnostic manner. Thorough evaluation of our approach on both synthetic and real datasets shows that it outperforms state-of-the-art methods in both calibrated and uncalibrated scenarios.

Index Terms—photometric stereo, non-Lambertian, uncalibrated, convolutional neural network.

1 Introduction

Photometric stereo aims at recovering the surface normals of a static scene from a set of images captured under different light directions with a fixed camera [1], [2]. Based on the availability of calibrated lighting conditions, photometric stereo can be categorized into *calibrated* and *uncalibrated* photometric stereo settings. Early calibrated photometric stereo methods assumed a simplified reflectance model, such as the ideal Lambertian model [1], [2] or analytical reflectance models [3], [4], [5]. However, most of the real-world objects are non-Lambertian, and a specific analytical model is only valid for a small set of materials. A bidirectional reflectance distribution function (BRDF) is a general form for describing the reflectance property of a surface, but it is difficult to directly use a non-parametric form of BRDFs for photometric stereo.

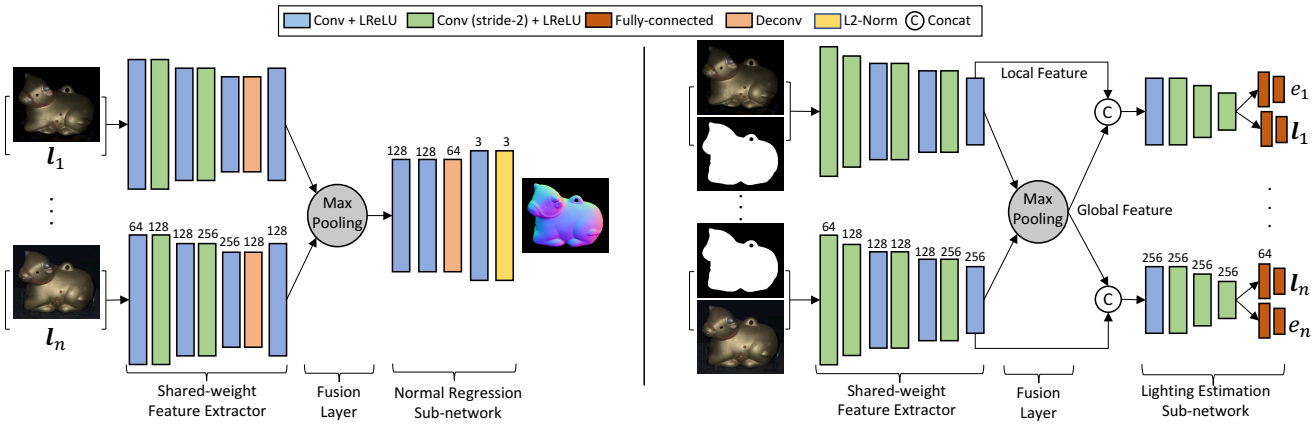
Recently, with the great success of deep learning in various computer vision tasks, deep learning based methods have been introduced to calibrated photometric stereo to handle surfaces with general and unknown isotropic reflectance [6], [7], [8]. Instead of explicitly modeling complex surface reflectances, they directly learn the mapping from reflectance observations to surface normals given known light directions. However, the method in [6] depends on a pre-defined set of light directions during training and testing. The methods in [6], [7] estimate the surface normals in a pixel-wise manner, making them not possible to account for the local context information of a surface point (*e.g.*, surface smoothness prior). Taniai and Maehara [8] introduced an optimization framework based on deep neural network, but their method suffers from complex scenes and requires a long processing time. Moreover, all of these methods assume known light directions.

On the other hand, the problem of uncalibrated photometric stereo still remains an open challenge, and a reliable method is desired for this relevant setting, because it eliminates the need for tedious light source calibration. Most of the existing methods for uncalibrated photometric stereo [9], [10], [11] assume a simplified reflectance model, such as the Lambertian model, and focus on resolving the shape-light ambiguity, such as the Generalized Bas-Relief (GBR) ambiguity [12]. Although methods of [13], [14] can handle surfaces with general BRDFs, they rely on the assumption of a uniform distribution of light directions for deriving a solution.

In this work, we study the problem of photometric stereo (PS) for surfaces with general and unknown isotropic reflectance. Following the conventional practice, we assume an orthographic camera with a linear radiometric response, directional lightings coming from the upper-hemisphere range, and the viewing direction being parallel to the z-axis pointing towards the origin of the world coordinates.

We first introduce a deep fully convolutional network (FCN), named PS-FCN, for *calibrated photometric stereo*. PS-FCN takes an arbitrary number of images with their associated light directions as input, and predicts a surface normal map of the scene in a fast feed-forward pass (see Fig. 1 (a)). Compared with previous learning based methods, our method does not depend on a pre-defined set of light directions during training and testing, and can handle multiple images in an order-agnostic manner. Moreover, convolutional neural network (CNN) can naturally

- *G. Chen and K-Y. K. Wong are with The University of Hong Kong, Hong Kong, China.*
  E-mail: {gychen,kykwong}@cs.hku.hk
- *K. Han is with University of Oxford, Oxford, United Kingdom.*
  E-mail: khan@robots.ox.ac.uk
- *Boxin 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
- *Y. Matsushita is with Osaka University, Osaka, Japan.*
  E-mail: yasumat@ist.osaka-u.ac.jp



(a) Network architecture of PS-FCN (b) Network architecture of LCNet

Fig. 1: Overview of the proposed method. Values above the layers indicate the number of feature channels.

incorporate information of the observations at neighboring pixels for computing feature maps, allowing our method to take advantage of local context information. To handle *uncalibrated photometric stereo* where light directions are unknown, one may consider to directly learn the mapping from images to surface normals without taking the light directions as input. However, as will be shown in Sec. 5, the performance of such a naïve model lags far behind those which take both images and light directions as input. Instead, we introduce another CNN, named Lighting Calibration Network (LCNet), to estimate light directions from input images (see Fig. 1 (b)). The estimated light directions and the input images can then be used by PS-FCN to estimate the surface normals.

To simulate complex non-Lambertian surfaces that are close to real-world scenes for training, we create two large-scale synthetic datasets using shapes from the blobby shape dataset [15] and the sculpture shape dataset [16], and BRDFs from the MERL BRDF dataset [17]. Once trained on the synthetic data, we show that our method can generalize well on real datasets, such as the DiLiGenT benchmark [18]. Extensive experiments on both synthetic and real datasets show that our approach outperforms existing methods in both calibrated and uncalibrated photometric stereo settings, clearly demonstrating its effectiveness.

We have presented preliminarily results of this work in [19], [20], and this paper extends them in several aspects. First, we extend PS-FCN to handle surfaces with spatially-varying BRDFs (SVBRDFs) by introducing a simple yet effective data normalization strategy. Second, we present a more detailed network analysis, experimental results, and discussion of how our method handles cast shadow. Third, we provide a comprehensive comparison between our method and the recent state-of-the-art methods. Last, we discuss how LCNet resolves the ambiguity in lighting estimation and its limitation. Our code, datasets, and models can be found at https://guanyingc.github.io/SDPS-Net.

2 Related Work

In this section, we review representative calibrated photometric stereo for non-Lambertian surfaces and uncalibrated photometric stereo methods. We also briefly review the loosely related work on learning based lighting estimation. Readers are referred to [18], [21] for more comprehensive surveys of photometric stereo methods.

Consider a non-Lambertian surface whose appearance is described by a general isotropic BRDF ρ. Given a surface point with a unit surface normal vector $\boldsymbol{n} \in \mathcal{S}^2$, $\mathcal{S}^2 = \{\boldsymbol{v} \in \mathbb{R}^3 : \|\boldsymbol{v}\|_2 = 1\}$ illuminated by the j-th incoming lighting with direction $\boldsymbol{l}_j \in \mathcal{S}^2$ and intensity $e_j \in \mathbb{R}_+$, the image formation model from a fixed viewpoint can be written as

$$m_j = e_j \rho(\boldsymbol{n}, \boldsymbol{l}_j) \max(\boldsymbol{n}^\top \boldsymbol{l}_j, 0) + \epsilon_j, \quad (1)$$

where m is the measured intensity, $\max(:, 0)$ accounts for attached shadows, and ϵ represents global illumination effects (*e.g.*, cast shadows and inter-reflections) and noise.

Calibrated photometric stereo For a Lambertian surface, the BRDF ρ reduces to an unknown constant. Theoretically, the albedo scaled surface normal can be uniquely determined from the shadow-free observations captured under three non-coplanar light directions [1]. However, perfect Lambertian surfaces barely exist. Many photometric stereo algorithms have been proposed to handle non-Lambertian surfaces. Outlier rejection based methods assume non-Lambertian observations to be local and sparse such that they can be treated as outliers. Various outlier rejection methods have been proposed so far. They are based on rank minimization [22], RANSAC [23], taking median values [24], expectation maximization [25], and sparse Bayesian regression [26]. These outlier rejection methods generally require lots of input images and have difficulty in handling objects with non-sparse non-Lambertian observations (*e.g.*, materials with broad and soft specular highlights).

Instead of rejecting specular observations as outliers, methods based on analytical reflectance models have been proposed. They adopt analytical models like Blinn-Phong model [3], Ward model [4], and Cook-Torrance model [5], to approximate the non-Lambertian reflectances. These methods require solving complex optimization problems, and can only handle limited classes of materials. Recently, bivariate BRDF representations [27], [28] were adopted to approximate isotropic BRDF, and a symmetry-based approach [29] was proposed to handle anisotropic reflectance without explicitly estimating a reflectance model.

More recently, a few deep learning based methods have been introduced to calibrated photometric stereo [6], [8], [7]. Santo *et al.* [6] proposed a fully-connected network to learn the mapping from reflectance observations captured under a pre-defined set of light directions to surface normal in a pixel-wise manner. Ikehata [7] introduced a fixed shape representation, called observation map, that is invariant to the number and permutation of the images. For each surface point of the object, all its observations are merged into an observation map based on the given light directions, and the observation map is then fed to a CNN to regress a normal vector. Compared with [6], [7], our method can take advantage of local context information in predicting the surface normals, which results in a more robust behavior. Taniai and Maehara [8] introduced an unsupervised learning framework that predicts both the surface normals and reflectance images of an object. Their model is "trained" at test time for each test object by minimizing the reconstruction loss between the input images and the rendered images, while our model is trained with supervised learning and achieves better performance on complex surfaces.

Uncalibrated photometric stereo Ignoring shadows and inter-reflections, the image formation model of a Lambertian surface is simplified to $m_j = e_j \rho \boldsymbol{n}^\top \boldsymbol{l}_j$. When light directions and intensities are unknown, the surface normals of a Lambertian object can only be estimated up to a 3×3 linear ambiguity [30], given by

$$m_j = e_j \rho (\mathbf{G}^{-\top} \boldsymbol{n})^\top (\mathbf{G} \boldsymbol{l}_j), \ \mathbf{G} \in \mathbb{R}^{3\times 3}. \tag{2}$$

This ambiguity can be reduced to a 3-parameter GBR ambiguity using the surface integrability constraint, which also holds true at the presence of attached and cast shadows [12], [31]. Previous work used additional clues like albedo priors [9], [10], inter-reflections [32], specular spikes [33], Torrance and Sparrow reflectance model [34], reflectance symmetry [35], [36], multi-view images [37], and local diffuse maxima [11], to resolve the GBR ambiguity. Cho *et al.* [38] considered a semi-calibrated case where the light directions are known but not their intensities. There are a few works that can handle non-Lambertian surfaces under unknown lighting. Hertzmann and Seitz [39] proposed an exemplar based method by inserting an additional reference object to the scene. Methods based on clues like similarity in radiance changes [40], [13] and attached shadow [41] were also introduced, but they require the light sources to be uniformly distributed on the whole sphere. Recently, Lu *et al.* [42] introduced a method based on the "constrained half-vector symmetry" to work with non-uniform lightings. Different from these traditional methods, our method can deal with surfaces with general and unknown isotropic reflectance without the need of explicitly utilizing any additional clues or reference objects, solving a complex optimization problem at test time, or making assumptions on the light source distribution.

Learning based lighting estimation Recently, learning based single-image lighting estimation methods have attracted a considerable attention. Gardner *et al.* [43] introduced a CNN for estimating HDR environment lighting from an indoor scene image. Hold-Goeffroy *et al.* [44] learned outdoor lighting using a physically-based sky model. Weber *et al.* [45] estimated indoor environment lighting from an image of an object with a known shape. Zhou *et al.* [46] estimated lighting, in the form of Spherical Harmonics, from a human face image by assuming a Lambertian reflectance model. Different from the above methods, our method can estimate accurate directional lightings from multiple images of a static object with general shape and non-Lambertian surface.

3 Learning Photometric Stereo

In this section, we first introduce our strategy for adapting CNNs to handle a variable number of inputs, and then introduce a deep fully convolutional network, named PS-FCN, for learning calibrated photometric stereo. For learning uncalibrated photometric stereo, we introduce another CNN, named Lighting Calibration Network (LCNet), to estimate lightings from input images. LCNet can be seamlessly integrated with PS-FCN to predict accurate surface normals. For the rest of this paper, we refer to light direction and intensity as "lighting."

3.1 Max-pooling for multi-feature fusion

CNNs have been successfully applied to dense regression problems like depth estimation [47] and surface normal estimation [48], where the number of input images is fixed and identical during training and testing. Note that adapting CNNs to handle a variable number of inputs during testing is not straightforward, as convolutional layers require the input to have a fixed number of channels during training and testing. Given a variable number of inputs, a shared-weight feature extractor can be used to extract features from each of the inputs (*e.g.*, siamese networks [49]), but an additional fusion layer is required to aggregate such features into a representation with a fixed number of channels. A convolutional layer is applicable for multi-feature fusion only when the number of inputs is fixed. Unfortunately, this is not practical for photometric stereo where the number of inputs often varies.

One possible way to tackle a variable number of inputs is to arrange the inputs sequentially and adopt a recurrent neural network (RNN) to fuse them. For example, Choy *et al.* [50] introduced a RNN framework to unify single- and multi-image 3D voxel prediction. The memory mechanism of RNN enables it to handle sequential inputs, but at the same time also makes it sensitive to the order of inputs. This order sensitive characteristic is not desirable for photometric stereo as it will restrict the illumination changes to follow a specific pattern, making the model less general.

More recently, order-agnostic operations (*e.g.*, pooling layers) have been exploited in CNNs to aggregate multi-image information. Wiles and Zisserman [16] used max-pooling to fuse features of silhouettes from different views for novel view synthesis and 3D voxel prediction. Hartmann *et al.* [51] adopted average-pooling to aggregate features of multiple patches for learning multi-patch similarity. In general, max-pooling operation can extract the most salient information from all the features, while average-pooling can smooth out the salient and non-activated features.

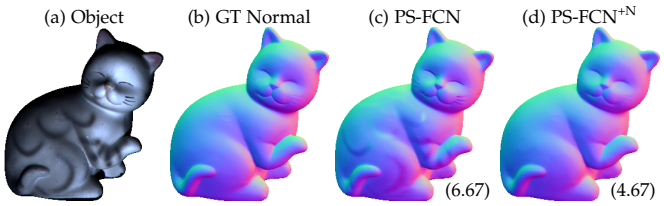


Fig. 2: Comparison between PS-FCN and PS-FCN^{+N} on an object with spatially-varying BRDFs. Numbers in the parentheses denote mean angular error (MAE) in degree.

For photometric stereo, we argue that max-pooling is a better choice for aggregating features from multiple inputs. Our motivation is that, under a certain light direction, regions with high intensities or specular highlights provide strong clues for surface normal inference (*e.g.*, for a surface point with a sharp specular highlight, its normal is close to the bisector of the viewing and light directions). Max-pooling can naturally aggregate such strong features from images captured under different light directions. Besides, max-pooling can ignore non-activated features during training, making it robust to cast shadow. As will be seen in Sec. 5, our experimental results do validate our arguments. We observe from experiments that each channel of the feature map fused by max-pooling is highly correlated to the response of the surface to a certain light direction. Strong responses in each channel are found in regions with surface normals having similar directions. The feature map can therefore be interpreted as a decomposition of the images under different light directions (see Fig. 8).

3.2 PS-FCN for calibrated photometric stereo

Network architecture PS-FCN is a multi-input-single-output (MISO) network consisting of three components, namely a shared-weight *feature extractor*, a *fusion layer*, and a *normal regression sub-network* (see Fig. 1 (a)). It can be trained and tested using an arbitrary number of images with their associated light directions as input[1].

For each light direction, we have a 3-channel input image with the dimensions of $3 \times h \times w$, where h and w are the image height and width, respectively. Concatenating images taken under q different lightings $\{\boldsymbol{l}_1, ..., \boldsymbol{l}_q\}$, we have the data with the dimensions of $q \times 3 \times h \times w$. In addition, we represent the light vectors $\{\boldsymbol{l}_1, ..., \boldsymbol{l}_q\}$ as 3-channel images having the same spatial resolution as the input images, resulting in another $q \times 3 \times h \times w$ data. Putting them together, we finally have $q \times 6 \times h \times w$ dimensional inputs to our model. We separately feed the image-light pairs to the shared-weight feature extractor to extract a feature map from each of the inputs, and apply a max-pooling operation in the fusion layer to aggregate these feature maps. Finally, the normal regression sub-network takes the fused feature map as input and estimates a normal map of the object.

The shared-weight feature extractor has seven convolutional layers, where the feature map is down-sampled twice and then up-sampled once, resulting in a down-sample factor of two. This design can increase the receptive field and preserve spatial information with a small memory consumption. The normal regression sub-network has four convolutional layers and up-samples the fused feature map to the same spatial dimension as the input images. An L2-normalization layer is appended at the end of the normal regression sub-network to produce the normal map.

As PS-FCN is a fully convolutional network, it can be applied to datasets with different image sizes. Thanks to the max-pooling operation in the fusion layer, PS-FCN possesses the order-agnostic property.

Loss function Training of our PS-FCN is supervised by the estimation error between the predicted and the ground-truth normal maps. We formulate our loss function as the commonly used cosine similarity loss, given by

$$\mathcal{L}_{\text{Normal}} = \frac{1}{hw} \sum_{i}^{hw} \left(1 - \boldsymbol{n}_i^\top \tilde{\boldsymbol{n}}_i\right), \tag{3}$$

where $\boldsymbol{n}_i$ and $\tilde{\boldsymbol{n}}_i$ denote the predicted normal and the ground-truth normal, respectively, at pixel i. If the predicted normal has a similar orientation as the ground truth, the dot-product $\boldsymbol{n}_i \cdot \tilde{\boldsymbol{n}}_i$ will be close to 1 and the loss becomes small, and vice versa. Other losses like mean squared error can also be alternatively adopted.

Extension to handle surfaces with SVBRDFs As PS-FCN is a fully-convolutional network that processes the input images in a patch-wise manner and is trained on surfaces with homogeneous BRDF, it may have difficulties in dealing with steep color changes caused by surfaces with SVBRDFs, as shown in Fig. 2 (c). A straightforward idea to tackle this problem is to train a model on surfaces with SVBRDFs. However, creating a large-scale training dataset for this purpose is not trivial, since modeling surface appearance with realistic SVBRDFs requires manual editing from artists. Even someone can collect a large-scale dataset of objects with SVBRDFs, the created dataset may not be able to faithfully cover the distribution of real data. In this paper, we introduce a simple yet effective data normalization strategy to enable PS-FCN to handle surfaces with SVBRDFs robustly. We will show that with the proposed data normalization strategy, our method can generalize well to surfaces with SVBRDFs after training only on surfaces with homogeneous BRDF.

During training, given q observations of a surface point[2], we concatenate all the observations and normalize them to a unit length vector by

$$(m'_1, ..., m'_q) = \left(\frac{m_1}{\sqrt{m_1^2 + ... + m_q^2}}, ..., \frac{m_q}{\sqrt{m_1^2 + ... + m_q^2}}\right), \tag{4}$$

where m and m' represent the original and normalized observations, respectively (for RGB images, we perform normalization on each channel separately). The intuition behind this operation is as follows. Consider a Lambertian model, the BRDF $\rho(\boldsymbol{n}, \boldsymbol{l})$ degenerates to a constant albedo

1. For calibrated photometric stereo, the input images are normalized by light intensities, and each light direction is represented by a unit 3-vector.

2. Note that the observations are already normalized by the light intensities.

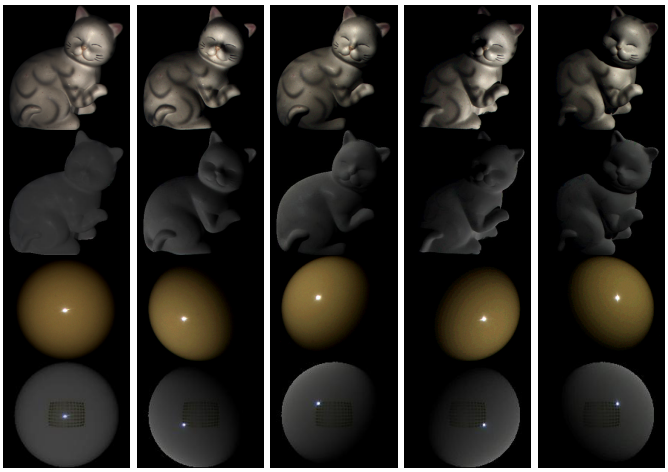


Fig. 3: Illustration of the introduced data normalization operation on CAT and BALL in the DiLiGenT benchmark. The first and third rows show the original images, while the second and last rows show the normalized images. Only 5 out of 96 images for each object are shown.

ρ and $m = \rho \max(\boldsymbol{n}^\top \boldsymbol{l}_j, 0)$. After the data normalization operation, we have

$$m'_i = \frac{\max(\boldsymbol{n}^\top \boldsymbol{l}_i, 0)}{\sqrt{\max(\boldsymbol{n}^\top \boldsymbol{l}_1, 0)^2 + \cdots + \max(\boldsymbol{n}^\top \boldsymbol{l}_q, 0)^2}}. \quad (5)$$

Equation (5) shows that the effect of albedo in Lambertian surfaces can be removed after performing data normalization, as shown in the first example in Fig. 3.

However, the above conclusion is not true for non-Lambertian surfaces, because for regions with specular highlights under some light directions, the observations under other light directions will be suppressed after data normalization (see the example of BALL in Fig. 3). Nevertheless, we experimentally found that such a normalization strategy works equally well for non-Lambertian surfaces under the PS-FCN framework. This might be explained by the fact that for a non-Lambertian surface under directional lighting, the low-frequency observations are quite close to Lambertian reflectance [27]. For observations exhibiting specular highlights under some light directions, the max-pooling operation in the fusion layer can naturally ignore the non-activated features (*i.e.*, features extracted from the suppressed observations) and aggregate the most salient features. Note that this normalization strategy has also been adopted in [40], [13] to compute the similarity between two pixel intensity profiles of non-Lambertian surfaces, while we use this normalization strategy as a preprocessing for CNNs to handle surfaces with SVBRDFs.

When the number of input images at test time t is different from that in training q, the magnitude of the normalized observations will be different, which leads to decreased performance (*e.g.*, when all observations have the same values, we have $m'_{\text{train}} = 1/\sqrt{q}, m'_{\text{test}} = 1/\sqrt{t}$). We experimentally verified that multiplying the normalized observations with the scalar $\sqrt{t/q}$ at test time solves this problem. We trained a variant model of PS-FCN, denoted as PS-FCN$^{+\text{N}}$, using the proposed data normalization strategy. Figure 2 (d) shows an example result that PS-FCN$^{+\text{N}}$ performed better than PS-FCN on surfaces with SVBRDFs.

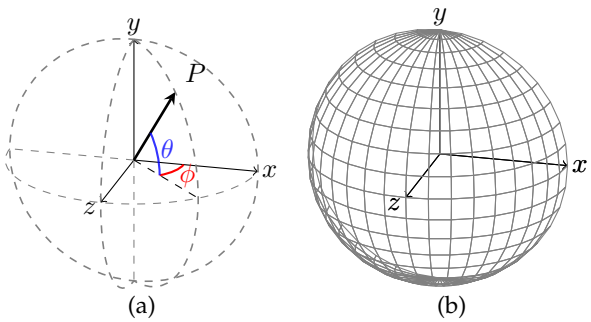


Fig. 4: (a) Illustration of the coordinate system (z axis is the viewing direction). $\phi \in [0°, 180°]$ and $\theta \in [-90°, 90°]$ are the azimuth and elevation of the light direction, respectively. (b) Example discretization of the light direction space when $K_d = 18$.

3.3 LCNet for lighting estimation

So far, we have assumed that the light intensities and directions are known. However, this assumption does not always hold in real applications. PS-FCN can be extended to handle the case where light directions are unknown by simply removing the light directions during training. However, as will be shown in Sec. 5, such a model is not optimal. To handle uncalibrated photometric stereo where both light intensities and directions are unknown, a more preferable solution is to learn the lightings from the input images, and then take the estimated lightings as part of the input for PS-FCN to estimate accurate normals. To estimate lightings from the images, one straightforward idea would be to directly regressing the light direction vectors and intensity values, but we propose that formulating the lighting estimation as a classification problem is a superior choice, as will be verified by our experiments in Sec. 5. Our arguments are as follows. First, classifying a light direction into a certain range is easier than regressing the exact value(s), and this will reduce the learning difficulty. Second, taking discretized light directions as input allows PS-FCN to better tolerate small errors in the estimated light directions, as verified by the experimental results.

Discretization of lighting space Since we cast our lighting estimation as a classification problem, we need to discretize the continuous lighting space. Note that a light direction in the upper-hemisphere can be described by its azimuth $\phi \in [0°, 180°]$ and elevation $\theta \in [-90°, 90°]$ (see Fig. 4 (a)). We can discretize the light direction space by evenly dividing both the azimuth and elevation into K_d bins, resulting in K_d^2 classes (see Fig. 4 (b)). Solving a K_d^2-class classification problem is not computationally efficient, as the softmax probability vector will have a very high dimension even when K_d is not large (*e.g.*, $K_d^2 = 1,296$ when $K_d = 36$). Instead, we estimate the azimuth and elevation of a light direction separately, leading to two K_d-class classification problems. Similarly, we evenly divide the range of possible light intensities into K_e classes (*e.g.*, $K_e = 20$ for a possible light intensity range of $[0.2, 2.0]$).

Local-global feature fusion A straightforward approach to estimate the lighting for each image would be simply taking a single image as input, encoding it into a feature map using a CNN, and feeding the feature map to a lighting prediction layer. It is not surprising that the result of such a simple solution is far from satisfactory. Note that the appearance of an object is determined by its surface geometry, reflectance model and the lighting. The feature map extracted from a single observation obviously does not provide sufficient information for resolving the shape-light ambiguity. Thanks to the nature of photometric stereo where multiple observations of an object are considered, we propose a local-global feature fusion strategy to extract more comprehensive information from multiple observations.

Specifically, we separately feed each image into a shared-weight feature extractor to extract a feature map, which we call *local feature* as it only provides information from a single observation. All local features of the input images are then aggregated into a *global feature* through a max-pooling operation. Such a global feature is expected to convey implicit surface geometry and reflectance information of the object which help to resolve the ambiguity in lighting estimation. Each local feature is concatenated with the global feature, and fed to a shared-weight lighting estimation sub-network to predict the lighting for each individual image. By taking both local and global features into account, our model can produce much more reliable results than using local features alone. We also include the object mask as input, as it allows the network to focus on extracting useful features inside the object region.

Network architecture LCNet is a multi-input-multi-output (MIMO) network that consists of a shared-weight *feature extractor*, a *fusion layer* (*i.e.*, max-pooling layer), and a shared-weight *lighting estimation sub-network* (see Fig. 1 (b)). It takes the observations of the object together with the object mask as input, and outputs the light directions and intensities in the form of softmax probability vectors of dimension K_d (azimuth), K_d (elevation) and K_e (intensity), respectively. We convert the output of LCNet to 3-vector light directions and scalar intensity values by simply taking the middle value of the range with the highest probability[3].

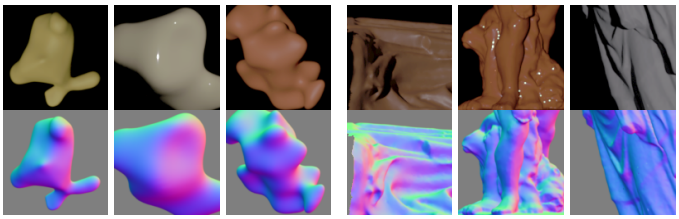
Loss function Multi-class cross-entropy loss is adopted for both light direction and intensity estimation, and the overall loss function is

$$\mathcal{L}_{\text{Light}} = \lambda_{l_a}\mathcal{L}_{l_a} + \lambda_{l_e}\mathcal{L}_{l_e} + \lambda_e\mathcal{L}_e, \tag{6}$$

where $\mathcal{L}_{l_a}$ and $\mathcal{L}_{l_e}$ are the loss terms for azimuth and elevation of the light direction, and $\mathcal{L}_e$ is the loss term for light intensity. During training, weights λ_{l_a}, λ_{l_e} and λ_e for the loss terms are set to 1.

Integration with PS-FCN Given the light directions and intensities estimated by LCNet, PS-FCN can be directly applied to estimate the surface normals of an object. For uncalibrated photometric stereo, we found that training PS-FCN from scratch with the estimated lighting of LCNet instead of the ground-truth lighting can lead to a more robust behavior over noise in the lighting.

3. We have experimentally verified that alternative ways like taking the expectation of the probability vector or performing quadratic interpolation in the neighborhood of the peak value do not improve the result.



(a) Blobby dataset (b) Sculpture dataset

Fig. 5: Examples of the synthetic training data (images are adjusted with gamma correction for visualization purpose).

4 Dataset

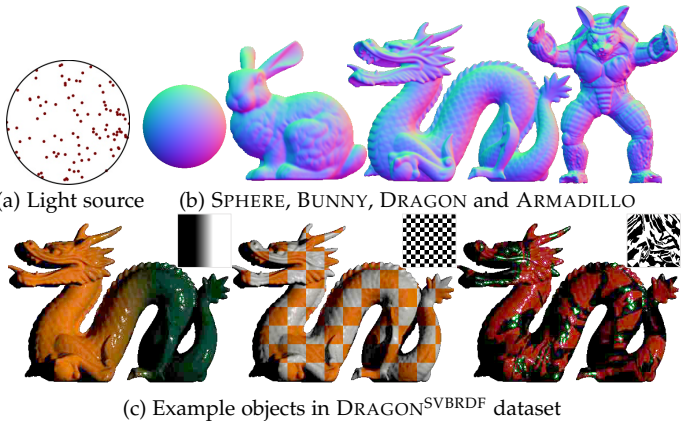
The training of our models requires ground-truth normal maps of the objects. However, obtaining ground-truth normal maps of real objects is a difficult and time-consuming task. Hence, we create two synthetic datasets for training and one synthetic dataset for testing. The publicly available real photometric stereo datasets are reserved to validate the generalization ability of our models.

4.1 Synthetic data for training

We used shapes from two existing 3D datasets, namely the blobby shape dataset [15] and the sculpture shape dataset [16], to generate our training data using the physically based raytracer Mitsuba [52]. Following SS17 [6], we employed the MERL dataset [17], which contains 100 different BRDFs of real-world materials, to define a diverse set of surface materials for rendering these shapes. Note that our datasets explicitly consider cast shadows during rendering.

Blobby dataset We first followed [6] to render our training data using the blobby shape dataset [15], which contains 10 blobby shapes with various normal distributions. For each blobby shape, $1,296$ regularly-sampled views (36 azimuth angles $\times$ 36 elevation angles) were used, and for each view, 2 out of 100 BRDFs were randomly selected, leading to $25,920$ samples ($10 \times 36 \times 36 \times 2$). For each sample, we rendered 64 images with a spatial resolution of 128×128 under light directions randomly sampled from a range of $180° \times 180°$, which is more general than the range ($74.6° \times 51.4°$) used in the real data benchmark [18]. We randomly split this dataset into $99:1$ for training and validation (see Fig. 5 (a)).

Sculpture dataset The surfaces in the blobby shape dataset are usually largely smooth and lack of details. To provide more complex (realistic) normal distributions for training, we employed 8 complicated 3D models from the sculpture shape dataset introduced in [16]. We generated samples for the sculpture dataset in exactly the same way we did for the blobby shape dataset, except that we discarded views containing holes or showing uniform normals (*e.g.*, flat facets). The rendered images are with a size of 512×512 when a whole sculpture shape is in the field of view. We then regularly cropped patches of size 128×128 from the rendered images and discarded those with a foreground ratio less than 50%. This gave us a dataset of $59,292$ samples, where each sample contains 64 images rendered



(a) Light source (b) SPHERE, BUNNY, DRAGON and ARMADILLO

(c) Example objects in DRAGON$^{\text{SVBRDF}}$ dataset

Fig. 6: (a) Lighting distribution of SynTest$^{\text{MERL}}$ dataset. The light direction is visualized by mapping a 3-d vector $[x, y, z]$ to a point $[x, y]$. (b) Ground-truth normals of SPHERE, BUNNY, DRAGON, and ARMADILLO. (c) Visualization of the selected material maps (Ramp, Checker, Irregular) and examples in DRAGON$^{\text{SVBRDF}}$ dataset.

under different light directions. Finally, we randomly split this dataset into $99 : 1$ for training and validation (see Fig. 5 (b)).

Training Details During training, we applied noise perturbation in the range of $[-0.025, 0.025]$ for data augmentation. To train PS-FCN for calibrated photometric stereo, given an image of size 128×128, we randomly performed image rescaling (with the rescaled width and height within the range of $[32, 128]$, without preserving the original aspect ratio). Image patches of size 32×32 were then randomly cropped for training. At test time, PS-FCN can take images of different sizes as input.

As the training data is rendered with uniform light intensity, to train LCNet for uncalibrated photometric stereo, we simulate images under different light intensities by randomly generated light intensities in the range of $[0.2, 2.0]$ to scale the magnitude of the images (*i.e.*, the ratio of the highest light intensity to the lowest one is 10)[4]. Note that this selected range contains a wider range of intensity values than the public photometric stereo datasets like DiLiGenT benchmark [18] and Gourd&Apple dataset [53]. As LCNet contains fully-connected layers and requires the input to have a fixed spatial dimension, the input image size for LCNet during both training and testing is 128×128.

4.2 Synthetic data for analysis

To quantitatively evaluate the performance of our method on different materials and shapes, we rendered a synthetic test dataset including Sphere, Bunny, Dragon, and Armadillo shapes. Hereafter, we denote this test dataset as SynTest$^{\text{MERL}}$ and these shapes as SPHERE, BUNNY, DRAGON, ARMADILLO respectively. Each shape was rendered with 100 isotropic BRDFs from MERL dataset [17] under 100 light directions randomly sampled from the upper-hemisphere, leading to 400 test objects (see Fig. 6 (a)-(b)).

4. Note that the ratio (other than the exact value) matters, since light intensity can only be estimated up to a scale factor.

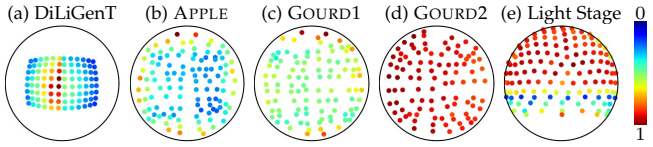


Fig. 7: Lighting distributions of the real testing datasets. The color of the point indicates the light intensity (value is divided by the highest intensity to normalize to $[0, 1]$).

Cast shadows and inter-reflections were considered during rendering using the physically based raytracer Mitsuba [52].

To analyze how surfaces with SVBRDFs affect the performance of our method, we created another synthetic test dataset with SVBRDFs, denoted as DRAGON$^{\text{SVBRDF}}$, following [54]. Specifically, we blended two BRDFs from 100 MERL dataset for DRAGON using 3 materials maps, namely the *Ramp*, *Checker*, and *Irregular*, as shown in Fig. 6 (c). Note that for each material map, there are $C(100, 2) = 4,950$ combinations of two BRDFs, leading to $14,850$ test objects.

4.3 Real data for testing

We employed three challenging real non-Lambertian photometric stereo datasets for testing, namely the *DiLiGenT benchmark* [18], *Gourd&Apple dataset* [53], and *Light Stage Data Gallery* [55]. Note that none of these datasets were used in the training.

DiLiGenT benchmark [18] is a public dataset containing 10 real objects, and each object was captured under 96 predefined light directions (see Fig. 7 (a)). Both ground-truth lighting conditions and normal maps are provided. We quantitatively evaluated the performance of our method on both lighting and normal estimation.

Gourd&Apple dataset [53] consists of three objects, namely APPLE, GOURD1, and GOURD2, with 112, 102 and 98 images, respectively. Figures 7 (b)-(d) visualize the lighting distributions of this dataset. Light Stage Data Gallery [55] is composed of six objects, and 253 images are provided for each object. We only used 133 images with the front side of the object under illumination. Figure 7 (e) visualizes the lighting distribution of the selected images. Since these two datasets only provide calibrated lightings (without ground-truth normal maps), we quantitatively evaluated our method on lighting estimation but only qualitatively evaluated it on normal estimation.

5 EXPERIMENTAL RESULTS

In this section, we present network analysis for our method, and compare our method with the previous state-of-the-art methods on both synthetic and real datasets.

Implementation details Our framework was implemented in PyTorch [56] and Adam optimizer [57] was used with default parameters ($\beta_1 = 0.9$ and $\beta_2 = 0.999$). PS-FCN contains 2.2 million learnable parameters. We trained PS-FCN using a batch size of 32 for 30 epochs, and it only took a few hours for training to converge using a single NVIDIA Titan X Pascal GPU (*e.g.*, about 9 hours using 32 image-light pairs per sample on both the blobby and sculpture datasets). Learning rate was initially set to 0.001 and divided by 2 every 5 epochs.

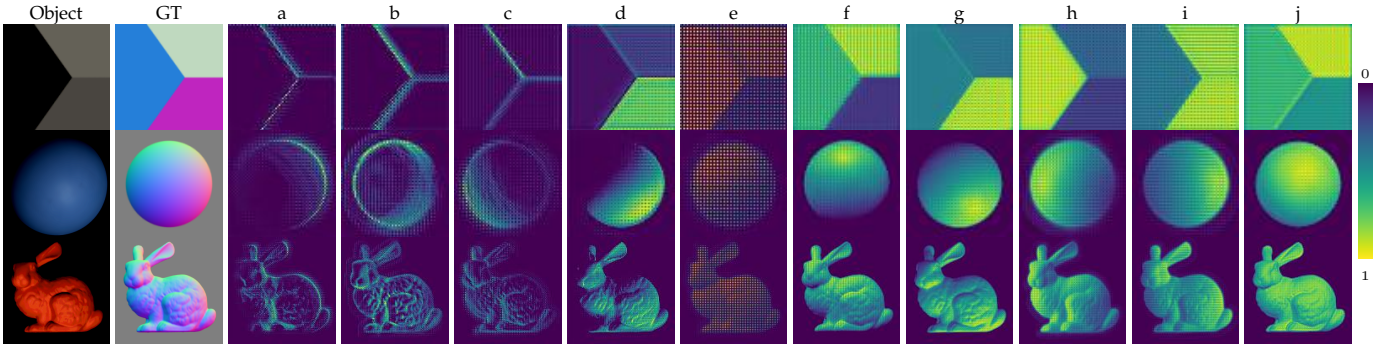


Fig. 8: Visualization of the learned feature map after fusion (the features were normalized to $[0, 1]$). The first two columns show the objects and ground-truth normals. The subsequent columns (a-j) visualize 10 out of 128 channels of the fused feature map. Note that different regions with similar normal directions are fired in different channels. Each channel can therefore be interpreted as the probability of the normal belonging to a certain direction (or alternatively as the object shading rendered under a certain light direction).

TABLE 1: Normal estimation results on SynTest$^{\text{MERL}}$ dataset. The results are averaged over samples rendered with 100 BRDFs. B and S stand for the blobby and sculpture training datasets, respectively.

Model Variants					Test Objects			
ID	Data	Fusion	Train #	Test #	SPHERE	BUNNY	DRAGON	ARMOD.
A0	B	Conv	32	32	4.54	6.74	9.57	9.87
A1	B	Max-p	32	32	3.65	5.33	7.86	8.09
A2	B	Avg-p	32	100	3.71	5.36	8.17	7.92
A3	B	Max-p	32	100	3.40	4.80	7.23	7.21
A4	B+S	Max-p	32	100	**2.66**	**3.80**	**4.83**	**5.24**

LCNet contains 4.4 million parameters. We trained LCNet using a batch size of 32 for 20 epochs, and it took about 22 hours to train LCNet on a single GPU with a fixed input image number of 32. The learning rate was initially set to 0.0005 and halved every 5 epochs.

5.1 Results on calibrated photometric stereo

To measure the accuracy of the predicted normal maps, mean angular error (MAE) in degree was used.

5.1.1 *Network analysis of PS-FCN with synthetic data*

We quantitatively analyzed PS-FCN on the synthetic dataset. For all the experiments in network analysis, we performed 100 random trials (save for the experiments using all 100 image-light pairs per sample during testing) and reported the average results.

Effectiveness of max-pooling We first validated the effectiveness of max-pooling in multi-feature fusion by comparing it with convolutional layers and average-pooling. Experiments with IDs A0 & A1 in Table 1 show that fusion by convolutional layer on the concatenated features was sub-optimal. This could be explained by the fact that the weights of the convolutional layer are related to the order of the input features, while the order of the input image-light pairs is random in our case, thus increasing the difficulty for the convolutional layer to find the relations among multiple features. Experiments with IDs A2 & A3 compared the performance of average-pooling and max-pooling for multi-feature fusion. It can be seen that max-pooling performs consistently better than average-pooling on SynTest$^{\text{MERL}}$ dataset. Figure 8 visualizes the fused features by max-pooling for three objects with different shapes and reflectances. We can see that each channel of the fused features can be interpreted as the probability of the normal belonging to a certain direction, and max-pooling can naturally aggregate such information from multiple observations.

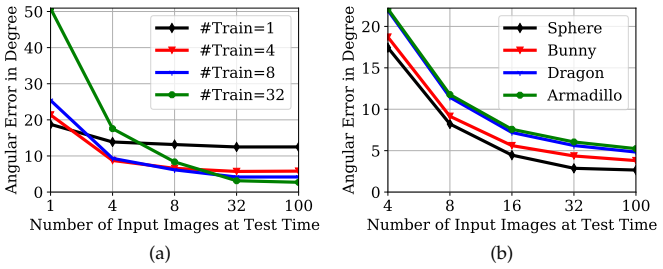


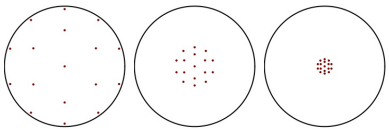
Fig. 9: (a) Results of PS-FCN trained and tested with different numbers of input images on SPHERE. (b) Results of PS-FCN trained with a fixed number of 32 input images and tested with different numbers of input images.

Effects of training data and input image number By comparing experiments with ID A3 & A4 in Table 1, we can see that training with the additional sculpture dataset that has a more complex normal distribution helped to boost the performance of PS-FCN. This result suggests that the performance of PS-FCN could be further improved by introducing more complex and realistic training data.

Figure 9 (a) shows that for a fixed number of inputs during testing, PS-FCN performs better when the number of inputs during training is close to that during testing. It is worth noting that when there is only one input image, the problem reduces to the more challenging shape-from-shading problem. Figure 9 (a) shows that PS-FCN performs best when the training image number is also 1, with an average MAE of 18.75° for SPHERE. However, this result is moderately inaccurate, indicating that PS-FCN has difficulties in resolving the ambiguity in the problem of shape from shading.

Figure 9 (b) shows that for a fixed number of inputs

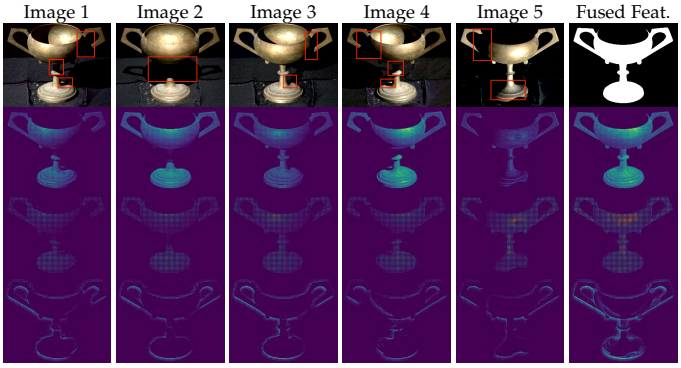
TABLE 2: Results of PS-FCN on BUNNY rendered using three different lighting distributions.



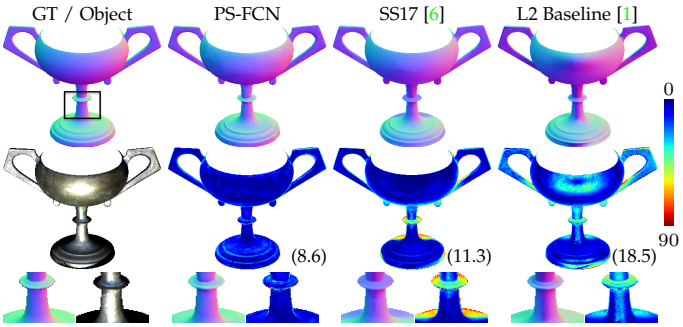
(a) (b) (c)

Type	Range	MAE
(a)	144°×144°	4.21
(b)	37°×37°	10.90
(c)	22°×22°	18.72

(d) Normal estimation



(a) The first five columns show the input images and the extracted features for each image (only 3 out of 128 feature channels are shown). The last column shows the object mask and the fused features by max-pooling. Red boxes in the images indicate regions with cast shadows.

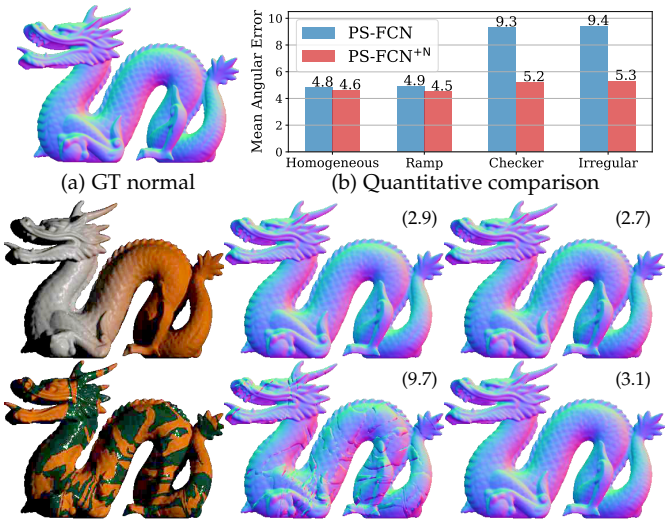


(b) Comparison between PS-FCN, SS17 [6] and L2 Baseline [1] on GOBLET. The first row shows the ground-truth and estimated normals, and the second row shows the object and the error maps.

Fig. 10: Illustration of how max-pooling fusion layer handles surface regions with cast shadow using GOBLET from the DiLiGenT benchmark. (Note that the provided object mask and ground-truth normal map do not include the concave interior of GOBLET.)

during training, the performance of PS-FCN increases with the number of inputs during testing. This is a desired property for photometric stereo as we can simply capture more images for robust estimation. For the rest of this paper, we refer PS-FCN as the model trained on both datasets and with an input of 32 image-light pairs per sample.

Effects of lighting distributions We tested PS-FCN on BUNNY rendered with three different lighting distributions, as shown in Table 2. These three distributions have the same number of light source (*i.e.*, 17), but with different spanning ranges. We can see that PS-FCN performs better when lightings are more diversely distributed. For the highly clustered distribution (see Table 2 (c)), the results of PS-FCN drops notably. Since the lightings are randomly sampled from the upper-hemisphere (*i.e.*, spanning range of



(a) GT normal (b) Quantitative comparison

(c) Qualitative results of PS-FCN (middle) and PS-FCN^{+N} (right).

Fig. 11: Comparison between PS-FCN and PS-FCN^{+N} on DRAGON^{SVBRDF} dataset.

180° × 180°) during training, it is therefore not surprising to see PS-FCN with decreased performance under this extreme lighting distribution.

Results on surface with cast shadows The presence of cast shadow is almost inevitable when the geometry of the object is non-convex, and is one of the major difficulties in photometric stereo. Given the observation that a real surface point is unlikely to be shadowed under all light directions, we argue that max-pooling fusion can naturally overcome the effect of cast shadow when determining the surface normals. This is because even a surface point is shadowed under some light directions, it can be observed under other light directions, and max-pooling can ignore those non-activated features and aggregate those activated features. Figure 10 (a) visualizes how max-pooling aggregates features from multiple observations and handles cast shadow. Compared with L2 baseline [1] and SS17 [6], our method is more robust in regions with cast shadow (see Fig. 10 (b)).

Results on surfaces with SVBRDFs To analyze how PS-FCN deteriorates in dealing with surfaces with SVBRDFs and verify the effectiveness of the proposed data normalization strategy, we compared PS-FCN and PS-FCN^{+N} on DRAGON^{SVBRDF} dataset and the results are summarized in Fig. 11. We can see that both models perform well on surfaces with homogeneous materials or surfaces with smooth BRDF changes (*e.g.*, surfaces blended with Ramp). However, PS-FCN has difficulty in dealing with steep color changes caused by SVBRDFs (*e.g.*, surfaces blended with Checker and Irregular). In contrast, PS-FCN^{+N} is robust in handling surfaces with different types of SVBRDFs, which clearly demonstrates the effectiveness of the proposed data normalization strategy.

5.1.2 *Evaluation on real datasets*

We compared our method against the recently proposed learning based methods [6], [8], [7] and other previous state-of-the-art methods on the DiLiGenT benchmark, as shown in Table 3. After training with the data normalization

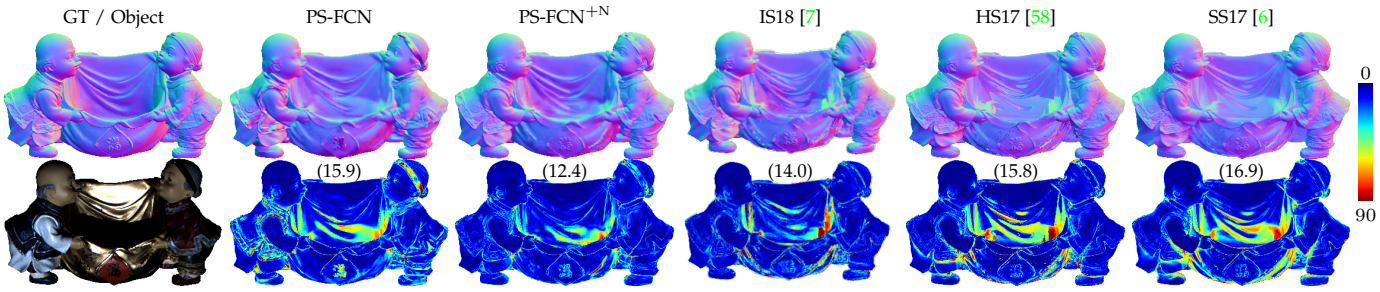


Fig. 12: Qualitative results on HARVEST in the DiLiGenT benchmark. Compared with PS-FCN, PS-FCN^{+N} performs better for surfaces with SVBRDFs. In contrast with the per-pixel normal estimation methods [7], [58], [6], PS-FCN^{+N} can take advantage of the surface smooth prior and estimate a smoother normal map with less noise artifacts.

TABLE 3: Quantitative comparison of calibrated photometric stereo on the DiLiGenT benchmark.

Method	BALL	CAT	POT1	BEAR	POT2	BUDD.	GOBL.	READ.	COW	HARV.	Avg.
L2 [1]	4.1	8.4	8.9	8.4	14.7	14.9	18.5	19.8	25.6	30.6	15.4
AZ08 [53]	2.7	6.5	7.2	**6.0**	11.0	12.5	13.9	14.2	21.5	30.5	12.6
WG10 [22]	2.1	6.7	7.2	6.5	13.1	10.9	15.7	15.4	25.9	30.0	13.4
IA14 [28]	3.3	6.7	6.6	7.1	8.8	10.5	9.7	14.2	13.1	26.0	10.6
ST14 [27]	1.7	6.1	6.5	6.1	8.8	10.6	10.1	13.6	13.9	25.4	10.3
SS17 [6]	2.0	6.5	7.1	6.3	7.9	12.7	11.3	15.5	8.0	16.9	9.4
TM18 [8]	1.5	5.4	6.1	5.8	7.8	10.4	11.5	11.0	**6.3**	22.6	8.8
HS17 [58]	**1.3**	4.9	**5.2**	**5.6**	6.4	8.5	7.6	12.1	8.2	15.8	7.6
IS18* [7]	2.2	**4.6**	5.4	8.3	**6.0**	7.9	**7.3**	12.6	8.0	14.0	7.6
PS-FCN	2.8	6.2	7.1	7.6	7.3	7.9	8.6	13.3	7.3	15.9	8.4
PS-FCN^{+N}	2.7	4.8	6.2	7.7	7.2	**7.5**	7.8	**10.9**	6.7	**12.4**	**7.4**

* indicates that the results of IS18 [7] on BEAR was computed using all of the 96 images. The results reported in IS18 [7] (BEAR: 4.1, Avg.: 7.2) was evaluated by discarding the first 20 images. When discarding the first 20 images, our results are PS-FCN (BEAR: 5.0, Avg.: 8.1) and PS-FCN^{+N} (BEAR: 5.0, Avg.: 7.1).

strategy, PS-FCN^{+N} performs better than PS-FCN on almost all of the ten objects, except for BEAR. Compared with the other state-of-the-art methods, PS-FCN^{+N} performs particularly well on surface with complexed geometry and/or SVBRDFs (*e.g.*, BUDDHA, READING, and HARVEST), and achieves state-of-the-art results with an average MAE of 7.4. Qualitative comparison on HARVEST is shown in Fig. 12. Note that PS-FCN did not outperform previous methods on all the 10 objects. We hypothesize that this might be caused by the limited training data. Different from pixel-wise approaches like IS18 [7] and HS17 [58], our method relies on diverse surface patches for training, while the current training data are only generated from 18 objects.

5.2 Results on uncalibrated photometric stereo

To measure the accuracy of the predicted light directions, the widely used mean angular error (MAE) in degree is adopted. Since the light intensities among the testing images can only be estimated up to a scale factor s, we introduce the scale-invariant relative error (RE)

$$RE_{scale} = \frac{1}{q}\sum_{i}^{q}\left(\frac{|se_i - \tilde{e}_i|}{\tilde{e}_i}\right), \quad (7)$$

where q is the number of images, e_i and $\tilde{e}_i$ are the estimated and ground-truth light intensities, respectively, for image i. The scale factor s is computed by solving $\underset{s}{\operatorname{argmin}} \sum_i^n (se_i - \tilde{e}_i)^2$ with least squares.

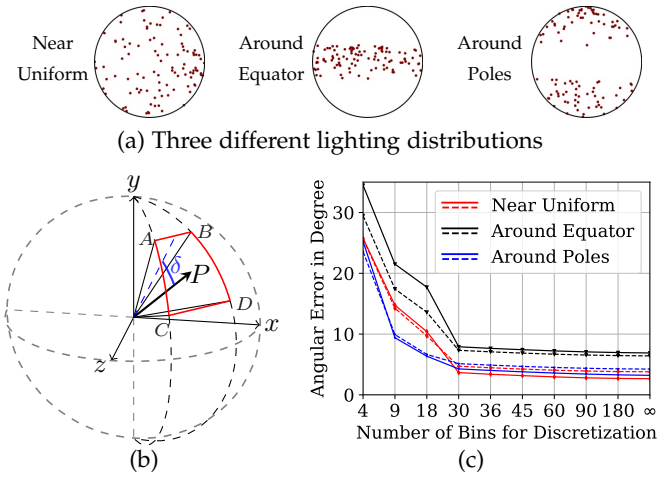


Fig. 13: (a) Three different lighting distributions. (b) Light directions A, B, C, and D have the maximum deviation angles with the actual light direction P after discretization. (c) Normal estimation error of PS-FCN on SPHERE (solid lines) and BUNNY (dashed lines) under different light direction space discretization levels (∞ indicates no discretization).

5.2.1 *Network analysis of LCNet with synthetic data*

For all experiments on synthetic dataset involving input with unknown light intensities, we randomly generated light intensities in the range of $[0.2, 2.0]$. Each experiment was repeated five times and the average results were reported.

Discretization of lighting space For a given number of bins K_d, the maximum deviation angle for azimuth and elevation of a light direction is $\delta = 180°/(K_d \times 2)$ after discretization (*e.g.*, $\delta = 2.5°$ when $K_d = 36$). Note that discretizing azimuth and elevation angles independently indicates that lighting space is more densely discretized around the poles and less around the equator. This suggests that the link between the quantization of the lighting space and surface normal estimation error correlates with the lighting distribution. To investigate how the light direction discretization affects the surface normal estimation accuracy, we tested PS-FCN on SPHERE and BUNNY rendered under three different lighting distributions, namely, *Near Uniform*, *Around Equator*, and *Around Poles* (see Fig. 13 (a)).

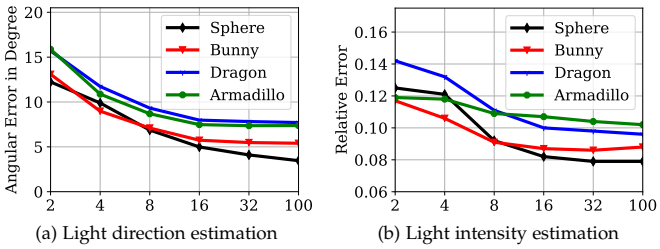
We divided the azimuth and elevation angles of light directions into different numbers of bins ranging from 4 to

TABLE 4: Lighting estimation results (MAE in degree for light direction and relative error for intensity) on SynTest^{MERL} dataset. The results are averaged over samples rendered with 100 BRDFs. (Value the lower the better)

Model	SPHERE		BUNNY		DRAGON		ARMADILLO	
	Dir.	Int.	Dir.	Int.	Dir.	Int.	Dir.	Int.
LCNet	**3.47**	**0.082**	**5.38**	**0.089**	**7.85**	**0.096**	**7.50**	**0.103**
LCNet$_{w/o\ mask}$	5.46	0.104	8.85	0.144	11.81	0.176	13.02	0.166
LCNet$_{local}$	6.87	0.198	9.98	0.255	10.58	0.264	9.50	0.266

TABLE 5: Results of LCNet and LCNet$_{reg}$ on SPHERE and BUNNY rendered under different lighting distributions.

Model	Near Uniform				Around Equator				Around Poles			
	SPHERE		BUNNY		SPHERE		BUNNY		SPHERE		BUNNY	
	Dir.	Int.	Dir.	Int.	Dir.	Int.	Dir.	Int.	Dir.	Int.	Dir.	Int.
LCNet	**3.47**	**0.082**	**5.38**	**0.089**	**3.32**	**0.079**	**5.33**	**0.077**	**4.82**	**0.088**	**6.34**	**0.095**
LCNet$_{reg}$	4.10	0.104	5.46	0.094	3.72	0.091	5.85	0.092	5.57	0.104	7.47	0.102



(a) Light direction estimation (b) Light intensity estimation

Fig. 14: Lighting estimation results of LCNet on SynTest^{MERL} dataset with varying input image numbers.

TABLE 6: Lighting estimation results of LCNet on DRAGON^{SVBRDF} dataset.

Model	Homogeneous		Ramp		Checker		Irregular	
	Dir.	Int.	Dir.	Int.	Dir.	Int.	Dir.	Int.
LCNet	7.85	0.096	7.44	0.076	8.19	0.111	8.68	0.103

180. For a specific bin number, we perturbed the azimuth and elevation of each ground-truth light direction by the maximum deviation angle, leading to four light directions that have the maximum possible angular deviations after discretization (see Fig. 13 (b)). We then used these light directions as input for PS-FCN to infer surface normals. The normal estimation error reported in Fig. 13 (c) is the upper-bound error for PS-FCN caused by discretization. We can see that the error increase caused by discretization is marginal for all three lighting distributions when $K_d \geq 30$. We chose a relatively sparse discretization of lighting space in this paper as it allows PS-FCN to learn to better tolerate small errors in the estimated lighting at test time.

Effectiveness of LCNet We first investigated the effect of object mask input and local-global feature fusion. Table 4 shows that taking the object mask as input and adopting the proposed local-global feature fusion strategy can effectively improve the lighting estimation results.

We then compared LCNet with a regression based baseline, denoted as LCNet$_{reg}$, to validate the effectiveness of the classification based model (please refer to the supplementary material for implementation details). Specifically, we tested LCNet and LCNet$_{reg}$ on three different lighting distributions illustrated in Fig. 13 (a). The results are shown in Table 5. The proposed classification based LCNet consistently outperforms LCNet$_{reg}$ on both light direction and intensity estimation. This echoes our hypothesis that classifying a light direction to a certain range is easier than regressing an exact value. Thus, solving the classification problem reduces the learning difficulty and improves the performance. It can also be seen that both methods perform better on *Around Equator* and worse on *Around Poles*. This suggests that lightings around the poles are more difficult to estimate due to their extremely directions, independent of the lighting space discretization.

Figure 14 shows that the performance of LCNet increases with the number of input images. This is expected, since more useful information can be used to infer lightings with more input images.

To analyze the effect of SVBRDFs in lighting estimation, we tested LCNet on DRAGON^{SVBRDF} dataset and reported the result in Table 6. We can see that LCNet is robust to surfaces with SVBRDFs, which indicates that unlike surface normal estimation, the features used by LCNet for lighting estimation are robust to SVBRDFs (*e.g.*, attached shadow).

Integration with PS-FCN For uncalibrated photometric stereo, we train a variant of PS-FCN, denoted as PS-FCN†, using the lighting estimated by LCNet. Note that the weights of LCNet was fixed during the training of PS-FCN†, as we found that end-to-end fine-tuning did not improve the performance. Experiments with IDs C1 & C2 in Table 7 show that after training with the discretized lighting estimated by LCNet, PS-FCN† performs better than PS-FCN given possibly noisy lightings at test time. Besides, experiments with IDs C1 & C3 show that PS-FCN† coupled with the classification based LCNet consistently outperforms that with the regression based LCNet$_{reg}$.

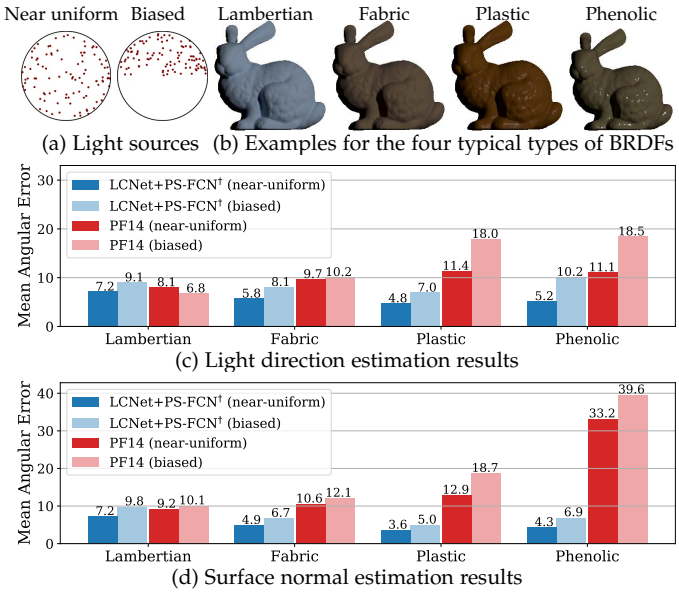
However, it is not straightforward to integrate LCNet with PS-FCN^{+N} to handle surface with SVBRDFs. Experiments with IDs C2 & C4 show that integrating LCNet with PS-FCN^{+N} decreases the performance of normal estimation. This is because when the estimated light intensities are noisy, the data normalization operation will magnify this error. We experimentally found that retraining PS-FCN^{+N} with the estimated lighting of LCNet cannot improve the result. In the future, we will investigate better strategies for handling surfaces with SVBRDFs under the uncalibrated setup.

Comparison with single-stage models To validate the effectiveness of the proposed two-stage framework, we compared our method with two different single-stage baseline models. We first train a variant of PS-FCN, denoted as UPS-FCN, without taking the light direction as input during training and testing. We then increased the model capacity of UPS-FCN by training a deeper network, denoted as UPS-FCN$_{deep+mask}$, that takes both the images and object mask as input. Please refer to our supplementary material for detailed network architectures.

Experiments with IDs C5 & C6 in Table 7 show that utilizing a deeper network and taking the object mask as input can improve the performance of single-stage model. However, experiments with IDs C1 & C5 show that the proposed method significantly outperforms the single-stage

TABLE 7: Normal estimation results on SynTest[MERL] dataset. PS-FCN† was trained given lightings estimated by LCNet or $\text{LCNet}_{\text{reg}}$.

ID	Model	# Param	SPHERE	BUNNY	DRAGON	ARMAD.
C0	PS-FCN	2.2 M	2.66	3.80	4.83	5.24
C1	LCNet + PS-FCN†	6.6 M	**2.71**	**4.09**	**6.41**	**7.09**
C2	LCNet + PS-FCN	6.6 M	3.19	4.67	6.92	7.70
C3	$\text{LCNet}_{\text{reg}}$ + PS-FCN†	6.6 M	3.22	4.99	6.63	7.54
C4	LCNet + PS-FCN^{+N}	6.6 M	4.53	5.35	7.36	7.99
C5	$\text{UPS-FCN}_{\text{deep+mask}}$	6.1 M	3.65	6.41	9.68	11.26
C6	UPS-FCN	2.2 M	7.44	12.34	14.44	15.93



(a) Light sources (b) Examples for the four typical types of BRDFs

(c) Light direction estimation results

(d) Surface normal estimation results

Fig. 15: Comparison between LCNet+PS-FCN† and PF14 [11] on BUNNY rendered with four different types of BRDFs under a near uniform lighting distribution and a biased lighting distribution.

model, when the input as well as the number of parameters are comparable. This result indicates that simply increasing the depth of the network cannot produce optimal results.

Comparison with the non-learning method [11] To further verify the effectiveness of our method over non-learning method, we compared our method with the existing uncalibrated method PF14 [11], which achieves state-of-the-art results on the DiLiGenT benchmark [18]. Figures 15 (c)-(d) show that our method consistently outperforms PF14 on BUNNY rendered using different types of BRDFs (*i.e.*, Lambertian, Fabric, Plastic, and Phenolic) under two different lighting distributions (one near uniform and one biased), especially on surfaces exhibiting specular highlights.

5.2.2 Evaluation on real datasets

Table 8 (a)-(b) show that LCNet outperforms the regression based baseline $\text{LCNet}_{\text{reg}}$ and achieves highly accurate results on both light direction and intensity estimation on DiLiGenT benchmark, with an average MAE of 4.92 and an average relative error of 0.068, respectively. Table 8 (c) compares the normal estimation results of LCNet+PS-FCN† with previous state-of-the-art methods on DiLiGenT benchmark. LCNet+PS-FCN† achieves state-of-the-art results on

TABLE 8: Quantitative results on the DiLiGenT benchmark.

(a) Results on light direction estimation.

Method	BALL	CAT	POT1	BEAR	POT2	BUDD.	GOBL.	READ.	COW	HARV.	Avg.
$\text{LCNet}_{\text{reg}}$	4.94	5.82	5.62	7.19	4.82	**3.90**	12.89	7.90	**4.19**	9.50	6.68
LCNet	**3.27**	**4.08**	**5.44**	**3.47**	**2.87**	4.34	**10.36**	**4.50**	4.52	**6.32**	**4.92**

(b) Results on light intensity estimation.

Method	BALL	CAT	POT1	BEAR	POT2	BUDD.	GOBL.	READ.	COW	HARV.	Avg.
$\text{LCNet}_{\text{reg}}$	**0.032**	**0.051**	**0.048**	0.167	0.074	0.080	0.075	0.141	**0.044**	0.085	0.080
LCNet	0.039	0.095	0.058	**0.061**	**0.048**	**0.048**	**0.067**	**0.105**	0.073	**0.082**	**0.068**

(c) Results on normal estimation.

Method	BALL	CAT	POT1	BEAR	POT2	BUDD.	GOBL.	READ.	COW	HARV.	Avg.
AM07 [9]	7.3	31.5	18.4	16.8	49.2	32.8	46.5	53.7	54.7	61.7	37.3
SM10 [10]	8.9	19.8	16.7	12.0	50.7	15.5	48.8	26.9	22.7	73.9	29.6
WT13 [36]	4.4	36.6	9.4	**6.4**	14.5	13.2	20.6	59.0	19.8	55.5	23.9
LM13 [13]	22.4	25.0	32.8	15.4	20.6	25.8	29.2	48.2	22.5	34.5	27.6
PF14 [11]	4.8	9.5	9.5	9.1	15.9	14.9	29.9	24.2	19.5	29.2	16.7
LC18 [42]	9.3	12.6	12.4	10.9	15.7	19.0	18.3	22.3	15.0	28.0	16.3
$\text{UPS-FCN}_{\text{deep+mask}}$	4.0	12.2	11.1	7.2	11.1	13.1	18.1	20.5	11.8	27.2	13.6
$\text{LCNet}_{\text{reg}}$+PS-FCN†	3.9	9.00	**8.0**	16.0	8.4	9.4	**11.5**	17.0	8.8	18.4	11.0
LCNet+PS-FCN†	**2.8**	**8.1**	**8.1**	6.9	**7.5**	**9.00**	**11.9**	**14.9**	**8.5**	**17.4**	**9.5**

almost all objects with an average MAE of 9.5, except for BEAR. Please refer to our supplementary material for more qualitative comparisons.

5.2.3 LCNet and the GBR ambiguity

Equation (2) in Sec. 2 indicates that when lightings are unknown, theoretically, the surface normal for a Lambertian surface can only be estimated up to the GBR transformation. This indicates that multiple combinations of albedo, normal, and lightings can result in the same set of images. However, the albedo of a GBR transformed surface becomes smoothly changing according to the surface normal, and such a spatially-varying albedo distribution seldom exists in the real-world [12]. Figure 16 shows that LCNet estimates the same lightings for two surfaces differed by GBR transformations, since the input images are the same. The estimated lightings are very close to lightings that correspond to the shape with uniform albedo (top row). This result suggests that LCNet is trained to predict the most probable lightings by implicitly assuming the albedo distribution is not GBR transformed, which is similar to previous non-learning methods relying on albedo distribution to resolve the GBR ambiguity [9], [10], [11]. As LCNet was trained using realistic surfaces (*i.e.*, surfaces without GBR transformed albedo distributions), it tries to predict realistic surfaces that are learned rather than other possible but unrealistic surfaces, which are not included in the training set (bottom row).

Figure 17 shows that LCNet fails to estimate reliable lightings on two special ambiguous cases, where a planar and a piecewise planar surfaces are rendered with uniform albedos. For planar surface (*i.e.*, the surface normal is constant over the whole surface), the ambiguity cannot be resolved even when assuming uniform albedos. Although there are three different surface normals in the piecewise planar surface, the surface integrability constraint [12] cannot be applied to reduce the linear ambiguity to GBR, making this case unsolvable as well. Note that existing methods also fail to handle these two ambiguous cases.

6 CONCLUSIONS

In this paper, we proposed a deep fully convolutional network, named PS-FCN, for calibrated photometric stereo. PS-FCN can accept an arbitrary number of images and their

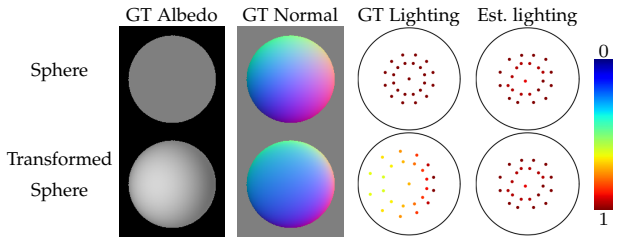


Fig. 16: Results of LCNet on surfaces differed by GBR transformations. The GT albedo, normal map, and lightings in the first and second rows give the same input images.

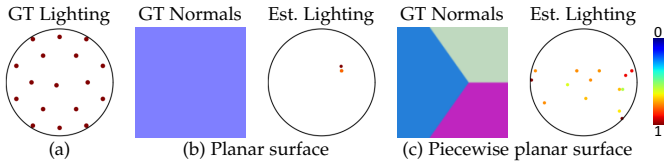


Fig. 17: Lighting estimation results of LCNet on a planar and a piecewise planar surfaces with uniform albedo.

associated light directions as input and estimate an accurate normal map in a fast feed-forward pass. We then introduced a simple yet effective data normalization strategy to allow PS-FCN better deal with surfaces with spatially-varying BRDFs. To handle the uncalibrated scenario, we introduced a convolutional network, named LCNet, to estimate lightings from input images. The estimated lightings and the input images can then be utilized by PS-FCN to estimate the surface normals. Our method does not require a predefined set of light directions during training and testing. It can handle multiple images and light directions in an order-agnostic manner. In order to train our model, two large-scale synthetic datasets with various realistic shapes and materials have been created. After training, our model can generalize well on challenging real datasets. Extensive results on both synthetic and real datasets have clearly shown that our method outperforms previous state-of-the-art methods on both calibrated and uncalibrated photometric stereo.

Acknowledgments

The work of Kai Han was supported by EPSRC Programme Grant Seebibyte EP/M013774/1. The work of Boxin Shi was supported by National Natural Science Foundation of China under Grant No. 61872012, National Key R&D Program of China (2019YFF0302902), and Beijing Academy of Artificial Intelligence (BAAI). The work of Yasuyuki Matsushita was supported by JSPS KAKENHI Grant Number JP19H01123. The work of Kwan-Yee K. Wong was supported by a grant from the Research Grant Council of the Hong Kong (SAR), China, under the Project HKU17203119.

References

[1] R. J. Woodham, "Photometric method for determining surface orientation from multiple images," *Optical engineering*, 1980. 1, 2, 9, 10
[2] W. M. Silver, "Determining shape and reflectance using multiple images," Ph.D. dissertation, Massachusetts Institute of Technology, 1980. 1
[3] S. Tozza, R. Mecca, M. Duocastella, and A. Del Bue, "Direct differential photometric stereo shape recovery of diffuse and specular surfaces," *Journal of Mathematical Imaging and Vision*, 2016. 1, 2
[4] H.-S. Chung and J. Jia, "Efficient photometric stereo on glossy surfaces with wide specular lobes," in *CVPR*, 2008. 1, 2
[5] R. Ruiters and R. Klein, "Heightfield and spatially varying BRDF reconstruction for materials with interreflections," in *Computer Graphics Forum*, 2009. 1, 2
[6] H. Santo, M. Samejima, Y. Sugano, B. Shi, and Y. Matsushita, "Deep photometric stereo network," in *ICCV Workshops*, 2017. 1, 3, 6, 9, 10
[7] S. Ikehata, "CNN-PS: CNN-based photometric stereo for general non-convex surfaces," in *ECCV*, 2018. 1, 3, 9, 10
[8] T. Taniai and T. Maehara, "Neural inverse rendering for general reflectance photometric stereo," in *ICML*, 2018. 1, 3, 9, 10
[9] N. G. Alldrin, S. P. Mallick, and D. J. Kriegman, "Resolving the generalized bas-relief ambiguity by entropy minimization," in *CVPR*, 2007. 1, 3, 12
[10] B. Shi, Y. Matsushita, Y. Wei, C. Xu, and P. Tan, "Self-calibrating photometric stereo," in *CVPR*, 2010. 1, 3, 12
[11] T. Papadhimitri and P. Favaro, "A closed-form, consistent and robust solution to uncalibrated photometric stereo via local diffuse reflectance maxima," *IJCV*, 2014. 1, 3, 12
[12] P. N. Belhumeur, D. J. Kriegman, and A. L. Yuille, "The bas-relief ambiguity," *IJCV*, 1999. 1, 3, 12
[13] F. Lu, Y. Matsushita, I. Sato, T. Okabe, and Y. Sato, "Uncalibrated photometric stereo for unknown isotropic reflectances," in *CVPR*, 2013. 1, 3, 5, 12
[14] F. Lu, I. Sato, and Y. Sato, "Uncalibrated photometric stereo based on elevation angle recovery from BRDF symmetry of isotropic materials," in *CVPR*, 2015. 1
[15] M. K. Johnson and E. H. Adelson, "Shape estimation in natural illumination," in *CVPR*, 2011. 2, 6
[16] O. Wiles and A. Zisserman, "SilNet: Single-and multi-view reconstruction by learning from silhouettes," in *BMVC*, 2017. 2, 3, 6
[17] W. Matusik, H. Pfister, M. Brand, and L. McMillan, "A data-driven reflectance model," in *SIGGRAPH*, 2003. 2, 6, 7
[18] B. Shi, Z. Mo, Z. Wu, D. Duan, S.-K. Yeung, and P. Tan, "A benchmark dataset and evaluation for non-Lambertian and uncalibrated photometric stereo," *IEEE TPAMI*, 2018. 2, 6, 7, 12
[19] G. Chen, K. Han, and K.-Y. K. Wong, "PS-FCN: A flexible learning framework for photometric stereo," in *ECCV*, 2018. 2
[20] G. Chen, K. Han, B. Shi, Y. Matsushita, and K.-Y. K. Wong, "SDPS-Net: Self-calibrating deep photometric stereo networks," in *CVPR*, 2019. 2
[21] S. Herbort and C. Wöhler, "An introduction to image-based 3d surface reconstruction and a survey of photometric stereo methods," *3D Research*, 2011. 2
[22] L. Wu, A. Ganesh, B. Shi, Y. Matsushita, Y. Wang, and Y. Ma, "Robust photometric stereo via low-rank matrix completion and recovery," in *ACCV*, 2010. 2, 10
[23] Y. Mukaigawa, Y. Ishii, and T. Shakunaga, "Analysis of photometric factors based on photometric linearization," *JOSA A*, 2007. 2
[24] D. Miyazaki, K. Hara, and K. Ikeuchi, "Median photometric stereo as applied to the segonko tumulus and museum objects," *IJCV*, 2010. 2
[25] T.-P. Wu and C.-K. Tang, "Photometric stereo via expectation maximization," *IEEE TPAMI*, 2010. 2
[26] S. Ikehata, D. Wipf, Y. Matsushita, and K. Aizawa, "Robust photometric stereo using sparse regression," in *CVPR*, 2012. 2
[27] B. Shi, P. Tan, Y. Matsushita, and K. Ikeuchi, "Bi-polynomial modeling of low-frequency reflectances," *IEEE TPAMI*, 2014. 2, 5, 10
[28] S. Ikehata and K. Aizawa, "Photometric stereo using constrained bivariate regression for general isotropic surfaces," in *CVPR*, 2014. 2, 10
[29] M. Holroyd, J. Lawrence, G. Humphreys, and T. Zickler, "A photometric approach for estimating normals and tangents," in *ACM TOG*, 2008. 2
[30] H. Hayakawa, "Photometric stereo under a light source with arbitrary motion," *JOSA A*, 1994. 3
[31] A. L. Yuille, D. Snow, R. Epstein, and P. N. Belhumeur, "Determining generative models of objects under varying illumination: Shape and albedo from multiple images using SVD and integrability," *IJCV*, 1999. 3
[32] M. K. Chandraker, F. Kahl, and D. J. Kriegman, "Reflections on the generalized bas-relief ambiguity," in *CVPR*, 2005. 3

[33] O. Drbohlav and M. Chaniler, "Can two specular pixels calibrate photometric stereo?" in *ICCV*, 2005. 3
[34] A. S. Georghiades, "Incorporating the torrance and sparrow model of reflectance in uncalibrated photometric stereo." in *ICCV*, 2003. 3
[35] P. Tan, S. P. Mallick, L. Quan, D. J. Kriegman, and T. Zickler, "Isotropy, reciprocity and the generalized bas-relief ambiguity," in *CVPR*, 2007. 3
[36] Z. Wu and P. Tan, "Calibrating photometric stereo by holistic reflectance symmetry analysis," in *CVPR*, 2013. 3, 12
[37] C. H. Esteban, G. Vogiatzis, and R. Cipolla, "Multiview photometric stereo," *IEEE TPAMI*, 2008. 3
[38] D. Cho, Y. Matsushita, Y.-W. Tai, and I. Kweon, "Photometric stereo under non-uniform light intensities and exposures," in *ECCV*, 2016. 3
[39] A. Hertzmann and S. M. Seitz, "Example-based photometric stereo: Shape reconstruction with general, varying brdfs," *IEEE TPAMI*, 2005. 3
[40] I. Sato, T. Okabe, Q. Yu, and Y. Sato, "Shape reconstruction based on similarity in radiance changes under varying illumination," in *ICCV*, 2007. 3, 5
[41] T. Okabe, I. Sato, and Y. Sato, "Attached shadow coding: Estimating surface normals from shadows under unknown reflectance and lighting conditions," in *ICCV*, 2009. 3
[42] F. Lu, X. Chen, I. Sato, and Y. Sato, "Symps: BRDF symmetry guided photometric stereo for shape and light source estimation," *IEEE TPAMI*, 2018. 3, 12
[43] M.-A. Gardner, K. Sunkavalli, E. Yumer, X. Shen, E. Gambaretto, C. Gagné, and J.-F. Lalonde, "Learning to predict indoor illumination from a single image," *ACM TOG*, 2017. 3
[44] Y. Hold-Geoffroy, K. Sunkavalli, S. Hadap, E. Gambaretto, and J.-F. Lalonde, "Deep outdoor illumination estimation," in *CVPR*, 2017. 3
[45] H. Weber, D. Prévost, and J.-F. Lalonde, "Learning to estimate indoor lighting from 3d objects," in *3DV*, 2018. 3
[46] H. Zhou, J. Sun, Y. Yacoob, and D. W. Jacobs, "Label denoising adversarial network (LDAN) for inverse lighting of faces," in *CVPR*, 2018. 3
[47] D. Eigen, C. Puhrsch, and R. Fergus, "Depth map prediction from a single image using a multi-scale deep network," in *NIPS*, 2014. 3
[48] X. Wang, D. Fouhey, and A. Gupta, "Designing deep networks for surface normal estimation," in *CVPR*, 2015. 3
[49] J. Bromley, I. Guyon, Y. LeCun, E. Säckinger, and R. Shah, "Signature verification using a "Siamese" time delay neural network," *International Journal of Pattern Recognition and Artificial Intelligence*, 1993. 3
[50] C. B. Choy, D. Xu, J. Gwak, K. Chen, and S. Savarese, "3d-r2n2: A unified approach for single and multi-view 3d object reconstruction," in *ECCV*, 2016. 3
[51] W. Hartmann, S. Galliani, M. Havlena, L. Van Gool, and K. Schindler, "Learned multi-patch similarity," in *ICCV*, 2017. 3
[52] W. Jakob, "Mitsuba renderer," 2010. 6, 7
[53] N. Alldrin, T. Zickler, and D. Kriegman, "Photometric stereo with non-parametric and spatially-varying reflectance," in *CVPR*, 2008. 7, 10
[54] D. B. Goldman, B. Curless, A. Hertzmann, and S. M. Seitz, "Shape and spatially-varying brdfs from photometric stereo," *IEEE TPAMI*, 2010. 7
[55] P. Einarsson, C.-F. Chabert, A. Jones, W.-C. Ma, B. Lamond, T. Hawkins, M. Bolas, S. Sylwan, and P. Debevec, "Relighting human locomotion with flowed reflectance fields," in *EGSR*, 2006. 7
[56] A. Paszke, S. Gross, S. Chintala, and G. Chanan, "PyTorch: Tensors and dynamic neural networks in python with strong gpu acceleration," 2017. 7
[57] D. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *ICLR*, 2015. 7
[58] Z. Hui and A. C. Sankaranarayanan, "Shape and spatially-varying reflectance estimation from virtual exemplars," *IEEE TPAMI*, 2017. 10

Guanying Chen is currently a Ph.D. student in the Department of Computer Science, The University of Hong Kong. He received his B.E. degree from Sun Yat-sen University in 2016. His research interests are learning based methods for computer vision.

Kai Han is a Postdoctoral Researcher at Visual Geometry Group (VGG), Department of Engineering Science, University of Oxford. He received his PhD degree from the Department of Computer Science, The University of Hong Kong, in 2018. He was a Research Intern at the WILLOW group of INRIA / École Normale Supérieure in 2016. His research interest includes computer vision and machine learning.

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 (IJCV) 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.

Kwan-Yee K. Wong received the BEng degree (Hons.) in computer engineering from The Chinese University of Hong Kong, in 1998, and the MPhil and PhD degrees in computer vision (information engineering) from the University of Cambridge, in 2000 and 2001, respectively. Since 2001, he has been with the Department of Computer Science at The University of Hong Kong, where he is currently an associate professor. His research interests are in computer vision and machine intelligence. He is currently an editorial board member of International Journal of Computer Vision (IJCV).