PS-FCN: A Flexible Learning Framework for Photometric Stereo

Guanying Chen[1(✉)], Kai Han[2], and Kwan-Yee K. Wong[1]

[1] The University of Hong Kong, Pokfulam, Hong Kong
{gychen,kykwong}@cs.hku.hk
[2] University of Oxford, Oxford, UK
khan@robots.ox.ac.uk

Abstract. This paper addresses the problem of photometric stereo for non-Lambertian surfaces. Existing approaches often adopt simplified reflectance models to make the problem more tractable, but this greatly hinders their applications on real-world objects. In this paper, we propose a deep fully convolutional network, called PS-FCN, that takes an arbitrary number of images of a static object captured under different light directions with a fixed camera as input, and predicts a normal map of the object in a fast feed-forward pass. Unlike the recently proposed learning based method, PS-FCN does not require a pre-defined set of light directions during training and testing, and can handle multiple images and light directions in an order-agnostic manner. Although we train PS-FCN on synthetic data, it can generalize well on real datasets. We further show that PS-FCN can be easily extended to handle the problem of uncalibrated photometric stereo. Extensive experiments on public real datasets show that PS-FCN outperforms existing approaches in calibrated photometric stereo, and promising results are achieved in uncalibrated scenario, clearly demonstrating its effectiveness.

Keywords: Photometric stereo · Convolutional neural network

1 Introduction

Given multiple images of a static object captured under different light directions with a fixed camera, the surface normals of the object can be estimated using photometric stereo techniques. Early photometric stereo algorithms often assumed an ideal Lambertian reflectance model [1,2]. Unfortunately, most of the real-world objects are non-Lambertian, and therefore more general models are needed to make photometric stereo methods more practical. Bidirectional reflectance distribution function (BRDF) is a general form for describing the reflectance property of a surface. However, it is difficult to handle general non-parametric BRDFs in non-Lambertian photometric stereo. Many researchers therefore adopted analytical reflectance models [3–5] to simplify the problem. However, a specific analytical model is only valid for a small set of materials.

V. Ferrari et al. (Eds.): ECCV 2018, LNCS 11213, pp. 3–19, 2018.
https://doi.org/10.1007/978-3-030-01240-3_1

Besides, fitting an analytical model to all the captured data requires solving a complex optimization problem. Hence, it remains an open and challenging problem to develop a computationally efficient photometric stereo method that can handle materials with diverse BRDFs.

Fig. 1. Given an arbitrary number of images and their associated light directions as input, our model estimates a normal map of the object in a fast feed-forward pass.

Deep learning frameworks [6,7] have shown great success in both high-level and low-level computer vision tasks. In the context of photometric stereo, Santo *et al.* [8] recently proposed a deep fully-connected network, called DPSN, to learn the mapping between reflectance observations and surface normals in a per-pixel manner. For each pixel, DPSN takes observations under 96 pre-defined light directions as input and predicts a normal vector. Note that since DPSN depends on a pre-defined set of light directions during training and testing, its practical use is sort of limited. Besides, DPSN predicts a normal vector based solely on the reflectance observations of a single pixel, it cannot take full advantage of the information embedded in the neighborhood of a surface point.

In this paper, we propose a flexible fully convolutional network [9], called PS-FCN, for estimating a normal map of an object (see Fig. 1). Convolutional network inherently takes observations in a neighborhood into account in computing the feature map, making it possible for PS-FCN to take advantage of local context information (e.g., surface smoothness prior). PS-FCN is composed of three components, namely a *shared-weight feature extractor* for extracting feature representations from the input images, a *fusion layer* for aggregating features from multiple input images, and a *normal regression network* for inferring the normal map (see Fig. 3).

Unlike [8], PS-FCN does not depend on a pre-defined set of light directions during training and testing, and allows the light directions used in testing different from those used in training. It takes an arbitrary number of images with their associated light directions as input, and predicts a normal map of the object in a fast feed-forward pass. It can handle multiple images and light directions in an order-agnostic manner. To simulate real-world complex non-Lambertian surfaces for training PS-FCN, we create two synthetic datasets using shapes from the blobby shape dataset [10] and the sculpture shape dataset [11], and BRDFs from the MERL BRDF dataset [12]. After training on synthetic data, we show that PS-FCN can generalize well on real datasets, including the

DiLiGenT benchmark [13], the Gourd&Apple dataset [14], and the Light Stage Data Gallery [15]. We further demonstrate that PS-FCN can be easily extended to handle the problem of uncalibrated photometric stereo, which reiterates the flexibility of our model. Extensive experiments on public real datasets show that PS-FCN outperforms existing approaches in calibrated photometric stereo, and promising results are achieved in uncalibrated scenario, clearly demonstrating its effectiveness.

2 Related Work

In this section, we briefly review representative non-Lambertian photometric stereo techniques. More comprehensive surveys of photometric stereo algorithms can be found in [13,16]. Non-Lambertian photometric stereo methods can be broadly divided into four categories, namely outlier rejection based methods, sophisticated reflectance model based methods, exemplar based methods, and learning based methods.

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 based on rank minimization [17], RANSAC [18], taking median values [19], expectation maximization [20], sparse Bayesian regression [21], etc. Outlier rejection methods generally require lots of input images and have difficulty in handling objects with dense non-Lambertian observations (e.g., materials with broad and soft specular highlights).

Many sophisticated reflectance models have been proposed to approximate the non-Lambertian model, including analytical models like Torrance-Sparrow model [3], Ward model [4], Cook-Torrance model [5], etc. Instead of rejecting specular observations as outliers, sophisticated reflectance model based methods fit an analytical model to all observations. These methods require solving complex optimization problems, and can only handle limited classes of materials. Recently, bivariate BRDF representations [22,23] were adopted to approximate isotropic BRDF, and a symmetry-based approach [24] was proposed to handle anisotropic reflectance without explicitly estimating a reflectance model.

Exemplar based methods usually require the observation of an additional reference object. Using a reference sphere, Hertzmann and Seitz [25] subtly transformed the non-Lambertian photometric stereo problem to a point matching problem. Exemplar based methods can deal with objects with spatially-varying BRDFs without knowing the light directions, but the requirement of known shape and material of the reference object(s) limits their applications. As an extension, Hui and Sankaranarayanan [26] introduced a BRDF dictionary to render virtual spheres without using a real reference object, but at the cost of requiring light calibration and longer processing time.

Recently, Santo *et al.* [8] proposed a deep fully-connected network, called DPSN, to regress per-pixel normal given a fixed number of observations (e.g., 96) captured under a pre-defined set of light directions. For each image point of the object, all its observations are concatenated to form a fixed-length vector, which

is fed into a fully-connected network to regress a single normal vector. DPSN can handle diverse BRDFs without solving a complex optimization problem or requiring any reference objects. However, it requires a pre-defined set of light directions during training and testing, which limits its practical uses. In contrast, our PS-FCN does not depend on a pre-defined set of light directions during training and testing, and allows the light directions used in testing to be different from those used in training. It takes an arbitrary number of images with their light directions as input, and predicts a normal map of the object in a fast feed-forward pass. It can handle multiple images and light directions in an order-agnostic manner.

Typically, photometric stereo methods require calibrated light directions, and the calibration process is often very tedious. A few works have been devoted to handle uncalibrated photometric stereo (e.g., [27–32]). These methods can infer surface normals in the absence of calibrated light directions. Our PS-FCN can be easily extended to handle uncalibrated photometric stereo, by simply removing the light directions during training. Afterwards, it can solely rely on the input images without known light directions to predict the normal map of an object.

3 Problem Formulation

In this paper, we follow the conventional practice by assuming orthographic projection, directional lights, and the viewing direction pointing towards the viewer. Given q color images of an object with p pixels captured under different light directions[1], a normal matrix $\mathbf{N}_{3\times p}$, a light direction matrix $\mathbf{L}_{3\times q}$, and an observation matrix $\mathbf{I}_{3\times p\times q}$ can be constructed. We further denote the BRDFs for all observations as $\boldsymbol{\Theta}_{3\times p\times q}$, where each 3-vector $\boldsymbol{\Theta}_{:,i,j}$ is a function of the normal, light direction, and viewing direction at (i, j). The image formation equation can be written as

$$\mathbf{I} = \boldsymbol{\Theta} \circ \text{repmat}(\mathbf{N}^\top \mathbf{L}, 3), \tag{1}$$

where $\circ$ represents element-wise multiplication, and $\text{repmat}(\mathbf{X}, 3)$ repeats the matrix $\mathbf{X}$ three times along the first dimension.

For a Lambertian surface, the BRDF for a surface point degenerates to an unknown constant vector. Theoretically, with three or more independent observations, the albedo scaled surface normal can be solved using linear least squares [1]. However, pure Lambertian surfaces barely exist. We therefore have to consider a more complex problem of non-Lambertian photometric stereo, in which we estimate the normal matrix $\mathbf{N}$ from an observation matrix $\mathbf{I}$ and light direction matrix $\mathbf{L}$ under unknown general BRDFs $\boldsymbol{\Theta}$.

We design a learning framework based on (1) to tackle the problem of non-Lambertian photometric stereo. Different from previous methods which approximate $\boldsymbol{\Theta}$ with some sophisticated reflectance models, our method directly learns the mapping from $(\mathbf{I}, \mathbf{L})$ to $\mathbf{N}$ without explicitly modeling $\boldsymbol{\Theta}$.

[1] Images are normalized by light intensities, and each light direction is represented by a unit 3-vector.

4 Learning Photometric Stereo

In this section, we first introduce our strategy for adapting CNNs to handle a variable number of inputs, and then present a flexible fully convolutional network, called PS-FCN, for learning photometric stereo.

4.1 Max-Pooling for Multi-feature Fusion

CNNs have been successfully applied to dense regression problems like depth estimation [33] and surface normal estimation [34], 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), 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, [35] 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.

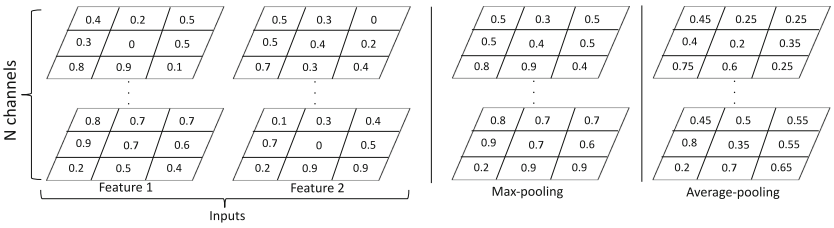

Fig. 2. A toy example for max-pooling and average-pooling mechanisms on multi-feature fusion.

More recently, order-agnostic operations (e.g., pooling layers) have been exploited in CNNs to aggregate multi-image information. Wiles and Zisserman [11] used max-pooling to fuse features of silhouettes from different views for novel

view synthesis and 3D voxel prediction. Hartmann *et al.* [36] 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. Figure 2 illustrates how max-pooling and average-pooling operations aggregate two features with a toy example.

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 Sect. 6, 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. 5).

4.2 Network Architecture

PS-FCN is a multi-branch siamese network [37] consisting of three components, namely a *shared-weight feature extractor*, a *fusion layer*, and a *normal regression network* (see Fig. 3). It can be trained and tested using an arbitrary number of images with their associated light directions as input.

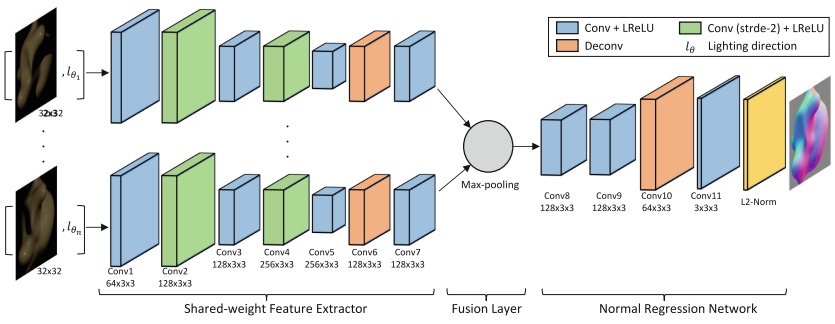

Fig. 3. Network architecture of PS-FCN.

For an object captured under q distinct light directions, we repeat each light direction (i.e., a 3-vector) to form a 3-channel image having the same spatial dimension as the input image ($3 \times h \times w$), and concatenate it with the input

image. Hence, the input to our model has a dimension of $q \times 6 \times h \times w$. 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 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 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 network to produce the normal map.

PS-FCN is a fully convolutional network, and it can be applied to datasets with different image scales. Thanks to the max-pooling operation in the fusion layer, it possesses the order-agnostic property. Besides, PS-FCN can be easily extended to handle uncalibrated photometric stereo, where the light directions are not known, by simply removing the light directions during training.

4.3 Loss Function

The learning of our PS-FCN is supervised by the estimation error between the predicted and the ground-truth normal maps. We formulate our loss function using the commonly used cosine similarity loss

$$L_{normal} = \frac{1}{hw} \sum_{i,j} (1 - \mathbf{N}_{ij} \cdot \tilde{\mathbf{N}}_{ij}), \tag{2}$$

where $\mathbf{N}_{ij}$ and $\tilde{\mathbf{N}}_{ij}$ denote the predicted normal and the ground truth, respectively, at the point (i, j). If the predicted normal has a similar orientation as the ground truth, the dot-product $\mathbf{N}_{ij} \cdot \tilde{\mathbf{N}}_{ij}$ will be close to 1 and the loss will be small, and vice versa. Other losses like mean square error can also be adopted.

5 Dataset

The training of PS-FCN requires the 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 model. Experimental results show that our PS-FCN trained on the synthetic datasets generalizes well on the challenging real datasets.

5.1 Synthetic Data for Training

We used shapes from two existing 3D datasets, namely the blobby shape dataset [10] and the sculpture shape dataset [11], to generate our training data using the physically based raytracer Mitsuba [38]. Following DPSN [8], we employed the MERL dataset [12], 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. For the sake of data loading efficiency, we stored our training data in 8-bit PNG format.

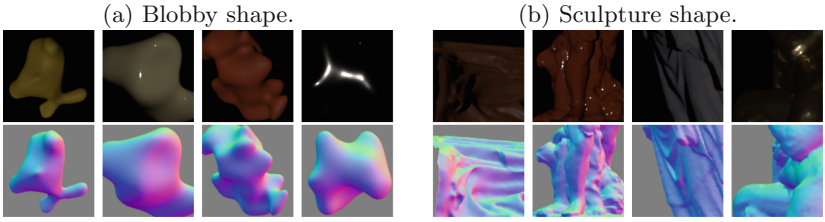

Fig. 4. Examples of the synthetic training data.

Blobby Dataset. We first followed [8] to render our training data using the blobby shape dataset [10], 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 [13]. We randomly split this dataset into 99 : 1 for training and validation (see Fig. 4(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 [11]. 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 under different light directions. Finally, we randomly split this dataset into 99 : 1 for training and validation (see Fig. 4(b)).

Data Augmentation. To narrow the gap between real and synthetic data, data augmentation was carried out on-the-fly during training. 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) and noise perturbation (in a range of $[-0.05, 0.05]$). Image patches of size 32×32 were then randomly cropped for training.

5.2 Synthetic Data for Testing

To quantitatively evaluate the performance of our model on different materials and shapes, we rendered a synthetic test dataset using a *Sphere* shape and a *Bunny* shape. Each shape was rendered with all of the 100 BRDFs from MERL dataset under 100 randomly sampled light directions. Similarly, the light directions were sampled from a range of $180^\circ \times 180^\circ$. As a result, we obtained 200 testing samples, and each sample contains 100 images.

5.3 Real Data for Testing

We employed three challenging real non-Lambertian photometric stereo datasets for testing, namely the DiLiGenT benchmark [13], the Gourd&Apple dataset [14], and the Light Stage Data Gallery [15]. Note that none of these datasets were used during training.

The DiLiGenT benchmark [13] contains 10 objects of various shapes with complex materials. For each object, 96 images captured under different predefined light directions and its ground-truth normal map are provided. We quantitatively evaluated our model on both the main and test datasets of this benchmark.

The Gourd&Apple dataset [14] and the Light Stage Data Gallery [15] are two other challenging datasets that without ground-truth normal maps. The Gourd&Apple dataset is composed of three objects, namely *Gourd1*, *Gourd2*, and *Apple*. They provide 102, 98 and 112 image-light pairs, respectively. The Light Stage Data Gallery [15] is composed of six objects, and 253 image-light pairs are provided for each object.[2] We qualitatively evaluated our model on these two datasets to further demonstrate the transferability of our model.

6 Experimental Evaluation

In this section, we present experimental results and analysis. We carried out network analysis for PS-FCN on the synthetic test dataset, and compared our method with the previous state-of-the-art methods on the DiLiGenT benchmark [13]. Mean angular error (MAE) in degree was used to measure the accuracy of the predicted normal maps. We further provided qualitative results on the Gourd&Apple dataset [14] and the Light Stage Data Gallery [15].

[2] In our experiment, for each object in the Light Stage Data Gallery, we only used the 133 pairs with the front side of the object under illumination.

6.1 Implementation Details

Our framework was implemented in PyTorch [39] with 2.2 million learnable parameters. We trained our model 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 1 hour for 8 image-light pairs per sample on the blobby dataset, and about 9 hours for 32 image-light pairs per sample on both the blobby and sculpture datasets). Adam optimizer [40] was used with default parameters ($\beta_1 = 0.9$ and $\beta_2 = 0.999$), where the learning rate was initially set to 0.001 and divided by 2 every 5 epochs. Our code, model and datasets are available at https://guanyingc.github.io/PS-FCN.

Table 1. Results of network analysis on the synthetic test dataset. The numbers represent the average MAE of all the objects (the lower the better). B and S stand for the blobby and sculpture training datasets respectively. († indicates the number of per-sample image-light pairs used is identical during training and testing.)

	Variants			Tested with # images				
ID	Data	Fusion Type	Train #	1	8	16	32	100
0	B	Avg-p	16	38.60	8.96	6.70	6.13	5.61
1	B	Avg-p	32	45.04	10.94	7.28	6.00	5.52
2	B	Conv	†	-	-	7.09	6.49	-
3	B	Max-p	1	**22.47**	14.58	13.95	13.88	13.67
4	B	Max-p	8	27.96	7.40	6.24	5.87	5.82
5	B	Max-p	16	46.85	8.44	6.24	5.64	5.43
6	B	Max-p	32	45.17	11.84	6.64	5.50	5.30
7	B + S	Max-p	8	26.65	**7.20**	6.17	5.71	5.66
8	B + S	Max-p	16	36.07	7.71	**5.94**	5.29	5.03
9	B + S	Max-p	32	51.18	9.12	6.01	**4.91**	**4.55**

6.2 Network Analysis

We quantitatively analyzed PS-FCN on the synthetic test dataset. In particular, we first validated the effectiveness of max-pooling in multi-feature fusion by comparing it with average-pooling and convolutional layers. We then investigated the influence of per-sample input number during training and testing. Besides, we investigated the influence of the complexity of training data. Last, we evaluated the performance of PS-FCN on different materials. 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 which are summarized in Table 1.

Effectiveness of Max-Pooling. Experiments with IDs 0, 1, 5 & 6 in Table 1 compared the performance of average-pooling and max-pooling for multi-feature fusion. It can be seen that max-pooling performed consistently better than

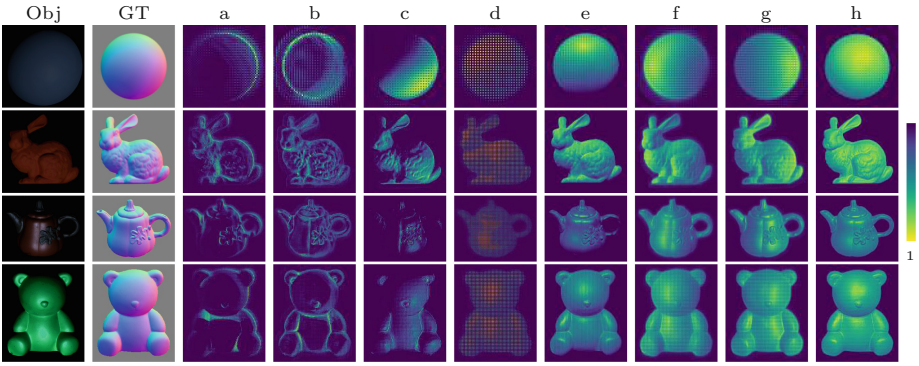

Fig. 5. Visualization of the learned feature map after fusion. The first two columns show the images and ground-truth normal maps. Each of the subsequent columns (a–h) shows one particular channel of the fused feature map. 8 out of the 128 channels of the feature map are presented. Note that different 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). Accurate normal maps can then be inferred from these probability distributions.

average-pooling, when the per-sample input number during testing was $\geq$16. Similarly, experiments with IDs 2, 5 & 6 showed that fusion by convolutional layers on the concatenated features was sub-optimal. This could be explained by the fact that the weights of the convolutional layers are related to the order of the concatenated features, while the orders of the input image-light pairs are random in our case, thus increasing the difficulty for the convolutional layers to find the relations among multiple features. Figure 5 visualizes the fused features (by max-pooling) of *Sphere* (blue-rubber) & *Bunny* (dark-red-paint) in synthetic test dataset, and *pot2* & *bear* in DiLiGenT main dataset. Note that all the image-light pairs were used as input and the features were normalized to $[0, 1]$.

Effects of Input Number. Referring to the experiments with IDs 3–6 in Table 1, for a fixed number of inputs during training, the performance of PS-FCN increased with the number of inputs during testing. For a fixed number of inputs during testing, PS-FCN performed better when the number of inputs during training was close to that during testing.

Effects of Training Ddata. By comparing experiments with IDs 4–6 (where the models were trained only on the blobby dataset) with experiments with IDs 7–9 (where the models were trained on both the blobby dataset and the sculpture dataset), we can see that the additional sculpture dataset with a more complex normal distribution helped to boost the performance. This suggests that the performance of PS-FCN could be further improved by introducing more complex and realistic training data.

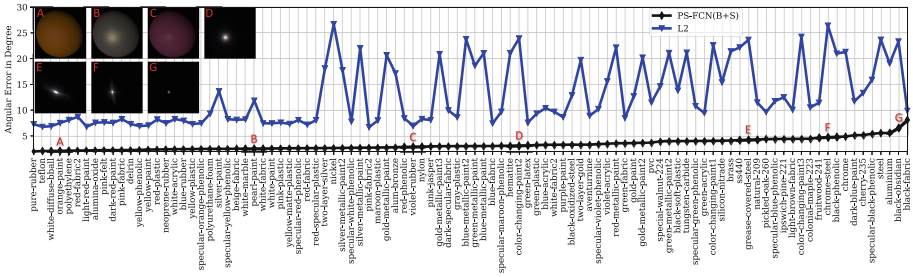

Fig. 6. Quantitative comparison between PS-FCN and L2 Baseline [1] on the samples of *Sphere* rendered with 100 different BRDFs. Images in the upper-left corner show the corresponding samples.

Results on Different Materials. Fig. 6 compares PS-FCN (trained with 32 per-sample inputs on both synthetic datasets) with L2 Baseline [1] on samples of *Sphere* that were rendered with 100 different BRDFs. It can be seen that PS-FCN significantly outperformed L2 Baseline. Note that PS-FCN generally performed better on materials with a light color than those with a dark color. This might be explained by the fact that max-pooling always tries to aggregate the most salient features for normal inference, and the image intensities of objects with a dark color are mostly very small. As a result, fewer useful features could be extracted to infer normals for objects of dark materials.

Table 2. Comparison of results on the DiLiGenT benchmark main dataset. The numbers represent the MAE (the lower the better). Results of PS-FCN under two different testing settings are reported, e.g., PS-FCN (B+S+32, 16) indicates the model trained on both the blobby dataset and the sculpture dataset with a per-sample input number of 32, and tested with a per-sample input number of 16. (Note that the result of PS-FCN (B+S+32, 16) is the average of 100 random trials.)

Method	ball	cat	pot1	bear	pot2	buddha	goblet	reading	cow	harvest	Avg.
L2 [1]	4.10	8.41	8.89	8.39	14.65	14.92	18.50	19.80	25.60	30.62	15.39
AZ08 [14]	2.71	6.53	7.23	**5.96**	11.03	12.54	13.93	14.17	21.48	30.50	12.61
WG10 [17]	2.06	6.73	7.18	6.50	13.12	10.91	15.70	15.39	25.89	30.01	13.35
IA14 [23]	3.34	6.74	6.64	7.11	8.77	10.47	9.71	14.19	13.05	25.95	10.60
ST14 [22]	**1.74**	**6.12**	**6.51**	6.12	8.78	10.60	10.09	13.63	13.93	25.44	10.30
DPSN [8]	2.02	6.54	7.05	6.31	7.86	12.68	11.28	15.51	8.01	16.86	9.41
PS-FCN (B+S+32, 16)	3.31	7.64	8.14	7.47	8.22	8.76	9.81	14.09	8.78	17.48	9.37
PS-FCN (B+S+32, 96)	2.82	6.16	7.13	7.55	**7.25**	**7.91**	**8.60**	**13.33**	**7.33**	**15.85**	**8.39**

6.3 Benchmark Comparisons

DiLiGenT Benchmark Main Dataset. We compared PS-FCN against the recently proposed learning based method DPSN [8] and other previous

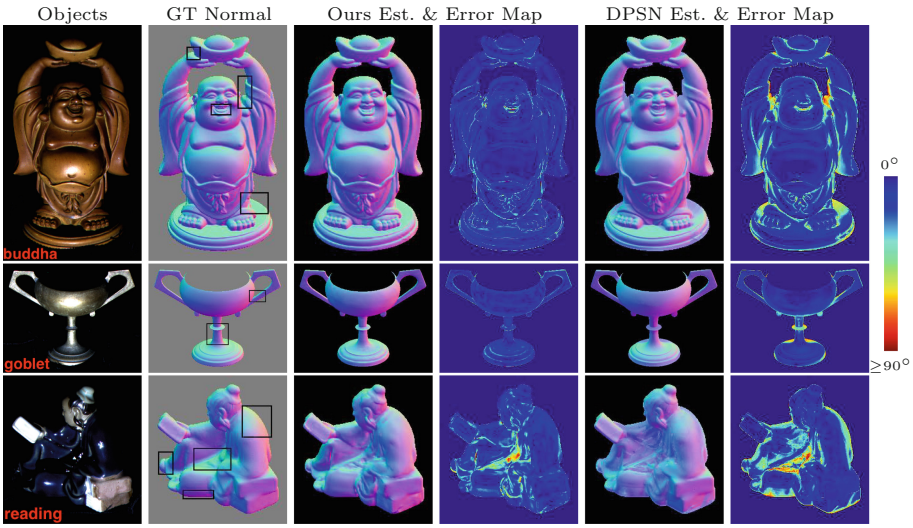

Fig. 7. Qualitative results on the DiLiGenT benchmark main dataset. The black boxes in the ground-truth normal maps are regions with cast shadows. Our method can produce more robust estimations in those regions compared with DPSN [8].

state-of-the-art methods. Quantitative results on the main dataset of the DiLiGenT benchmark are shown in Table 2. Compared with other methods, PS-FCN performed particularly well on objects with complicated shapes (e.g., *buddha*, *reading*, and *harvest*) and/or spatially varying materials (e.g., *pot2*, *goblet* and *cow*). Our best performer, which achieved an average MAE of 8.39°, was trained with 32 per-sample inputs on both synthetic datasets and tested with all 96 inputs for each object. With only 16 inputs per object during testing, PS-FCN still outperformed the previous methods in terms of the average MAE. Figure 7 presents the qualitative comparison between PS-FCN and DPSN. It can be seen that PS-FCN is more robust in regions with cast shadows.

DiLiGenT Benchmark Test Dataset. We further evaluated our model on the test dataset of the DiLiGenT benchmark, with the ground-truth normal maps withheld by the original authors (see. Table 3). Similar to the results on the main dataset, PS-FCN outperformed other methods on the test dataset. More results of the other methods can be found on the benchmark website[3] for comparison.

Uncalibrated Photometric Stereo Extension. PS-FCN can be easily extended to handle uncalibrated photometric stereo by simply removing the light directions from the input. To verify the potential of our framework towards uncalibrated photometric stereo, we trained an uncalibrated variant of our model, denoted as UPS-FCN, taking only images as input (note that we assume the

[3] https://sites.google.com/site/photometricstereodata/home/summary-of-benchmarking-results.

Table 3. Comparison of results on the DiLiGenT benchmark test dataset. The numbers represent the MAE (the lower the better).

Method	cat	pot1	bear	pot2	buddha	goblet	reading	cow	harvest	Avg.
IA14 [23]	**5.61**	**6.33**	**5.12**	8.83	11.00	10.54	**13.27**	11.18	24.82	10.74
ST14 [22]	6.43	6.64	6.09	8.94	10.92	10.33	14.16	10.82	25.43	11.08
DPSN [8]	5.82	8.26	6.32	9.02	12.80	12.04	16.11	8.00	17.78	10.68
PS-FCN (B+S+32, 96)	6.24	7.59	5.42	**7.11**	**8.30**	**8.62**	13.43	**7.98**	**15.93**	**8.96**

images were normalized by the light intensities). UPS-FCN was trained on both synthetic datasets using 32 image-light pairs as input. We compared our UPS-FCN with the existing uncalibrated methods. The results are reported in Table 4, our UPS-FCN outperformed existing methods in terms of the average MAE, which demonstrates the effectiveness and flexibility of our model.

Table 4. Comparison of results for uncalibrated photometric stereo on the DiLiGenT benchmark main dataset. The numbers represent the MAE (the lower the better).

Method	ball	cat	pot1	bear	pot2	buddha	goblet	reading	cow	harvest	Avg.
AM07 [27]	7.27	31.45	18.37	16.81	49.16	32.81	46.54	53.65	54.72	61.70	37.25
SM10 [28]	8.90	19.84	16.68	11.98	50.68	15.54	48.79	26.93	22.73	73.86	29.59
WT13 [29]	**4.39**	36.55	**9.39**	**6.42**	14.52	**13.19**	20.57	58.96	19.75	55.51	23.93
PF14 [30]	4.77	**9.54**	9.51	9.07	15.90	14.92	29.93	24.18	19.53	29.21	16.66
LC18 [32]	9.30	12.60	12.40	10.90	15.70	19.00	**18.30**	**22.30**	15.00	28.00	16.30
UPS-FCN	6.62	14.68	13.98	11.23	**14.19**	15.87	20.72	23.26	**11.91**	**27.79**	**16.02**

6.4 Testing on Other Real Datasets

Due to absence of ground-truth normal maps, we qualitatively evaluated our best-performing model PS-FCN (B+S+32) on the Gourd&Apple dataset [14] and the Light Stage Data Gallery [15]. Figure 8 shows the estimated normal maps and surfaces reconstructed using [41]. The reconstructed surfaces convincingly reflect the shapes of the objects, demonstrating the accuracy of the normal maps predicted by PS-FCN.

7 Conclusions

In this paper, we have proposed a flexible deep fully convolutional network, called PS-FCN, that accepts an arbitrary number of images and their associated light directions as input and regresses an accurate normal map. Our PS-FCN does not require a pre-defined set of light directions during training and testing, and

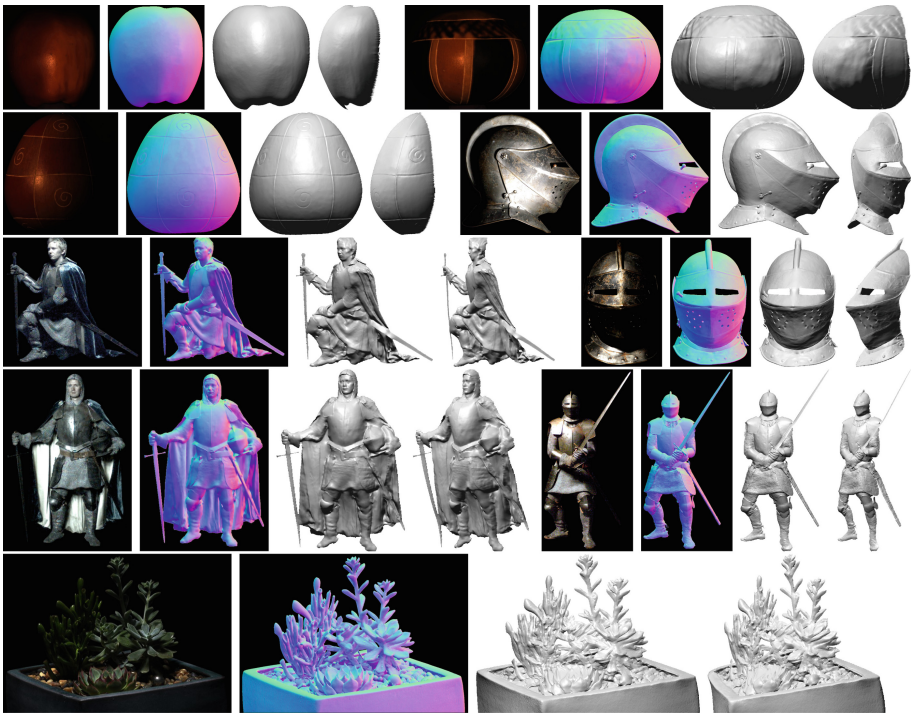

Fig. 8. Qualitative results for the Gourd&Apple dataset and Light Stage Data Gallery. For each shape, a sample input image, the estimated normal map, and two views of the reconstructed surfaces are shown. (Best viewed in PDF with zoom.)

allows the light directions used in testing different from that used in training. It can handle multiple images and light directions in an order-agnostic manner. In order to train PS-FCN, two synthetic datasets with various realistic shapes and materials have been created. After training, PS-FCN can generalize well on challenging real datasets. In addition, PS-FCN can be easily extended to handle uncalibrated photometric stereo. Results on diverse real datasets have clearly shown that PS-FCN outperforms previous calibrated photometric stereo methods, and promising results have been achieved in uncalibrated scenario.

Acknowledgments. We thank Hiroaki Santo for his help with the comparison to DPSN. We also thank Boxin Shi and Zhipeng Mo for their help with the evaluation on the DiLiGenT benchmark. We gratefully acknowledge the support of NVIDIA Corporation with the donation of the Titan X Pascal GPU used for this research. Kai Han is supported by EPSRC Programme Grant Seebibyte EP/M013774/1.

References

1. Woodham, R.J.: Photometric method for determining surface orientation from multiple images. Opt. Eng. **19**, 191130 (1980)

2. Silver, W.M.: Determining shape and reflectance using multiple images. Ph.D. thesis, Massachusetts Institute of Technology (1980)
3. Georghiades, A.S.: Incorporating the torrance and sparrow model of reflectance in uncalibrated photometric stereo. In: ICCV (2003)
4. Chung, H.S., Jia, J.: Efficient photometric stereo on glossy surfaces with wide specular lobes. In: CVPR (2008)
5. Ruiters, R., Klein, R.: Heightfield and spatially varying BRDF reconstruction for materials with interreflections. In: Computer Graphics Forum (2009)
6. Krizhevsky, A., Sutskever, I., Hinton, G.E.: Imagenet classification with deep convolutional neural networks. In: NIPS (2012)
7. LeCun, Y., Bottou, L., Bengio, Y., Haffner, P.: Gradient-based learning applied to document recognition. In: Proceedings of the IEEE (1998)
8. Santo, H., Samejima, M., Sugano, Y., Shi, B., Matsushita, Y.: Deep photometric stereo network. In: ICCV Workshops (2017)
9. Long, J., Shelhamer, E., Darrell, T.: Fully convolutional networks for semantic segmentation. In: CVPR (2015)
10. Johnson, M.K., Adelson, E.H.: Shape estimation in natural illumination. In: CVPR (2011)
11. Wiles, O., Zisserman, A.: SilNet: single-and multi-view reconstruction by learning from silhouettes. In: BMVC (2017)
12. Matusik, W., Pfister, H., Brand, M., McMillan, L.: A data-driven reflectance model. In: SIGGRAPH (2003)
13. Shi, B., Wu, Z., Mo, Z., Duan, D., Yeung, S.K., Tan, P.: A benchmark dataset and evaluation for non-lambertian and uncalibrated photometric stereo. IEEE TPAMI (2018)
14. Alldrin, N., Zickler, T., Kriegman, D.: Photometric stereo with non-parametric and spatially-varying reflectance. In: CVPR (2008)
15. Einarsson, P., et al.: Relighting human locomotion with flowed reflectance fields. In: EGSR (2006)
16. Herbort, S., Wöhler, C.: An introduction to image-based 3D surface reconstruction and a survey of photometric stereo methods. 3D Res. **2**, 4 (2011)
17. Wu, L., Ganesh, A., Shi, B., Matsushita, Y., Wang, Y., Ma, Y.: Robust photometric stereo via low-rank matrix completion and recovery. In: Kimmel, R., Klette, R., Sugimoto, A. (eds.) ACCV 2010 Part III. LNCS, vol. 6494, pp. 703–717. Springer, Heidelberg (2011). https://doi.org/10.1007/978-3-642-19318-7_55
18. Mukaigawa, Y., Ishii, Y., Shakunaga, T.: Analysis of photometric factors based on photometric linearization. JOSA A **24**, 3326–3334 (2007)
19. Miyazaki, D., Hara, K., Ikeuchi, K.: Median photometric stereo as applied to the segonko tumulus and museum objects. IJCV **86**, 229 (2010)
20. Wu, T.P., Tang, C.K.: Photometric stereo via expectation maximization. IEEE TPAMI **32**, 546–560 (2010)
21. Ikehata, S., Wipf, D., Matsushita, Y., Aizawa, K.: Robust photometric stereo using sparse regression. In: CVPR (2012)
22. Shi, B., Tan, P., Matsushita, Y., Ikeuchi, K.: Bi-polynomial modeling of low-frequency reflectances. IEEE TPAMI (2014)
23. Ikehata, S., Aizawa, K.: Photometric stereo using constrained bivariate regression for general isotropic surfaces. In: CVPR (2014)
24. Holroyd, M., Lawrence, J., Humphreys, G., Zickler, T.: A photometric approach for estimating normals and tangents. In: ACM TOG (2008)
25. Hertzmann, A., Seitz, S.M.: Example-based photometric stereo: shape reconstruction with general, varying BRDFs. IEEE TPAMI **27**, 1254–1264 (2005)

26. Hui, Z., Sankaranarayanan, A.C.: A dictionary-based approach for estimating shape and spatially-varying reflectance. In: ICCP (2015)
27. Alldrin, N.G., Mallick, S.P., Kriegman, D.J.: Resolving the generalized bas-relief ambiguity by entropy minimization. In: CVPR (2007)
28. Shi, B., Matsushita, Y., Wei, Y., Xu, C., Tan, P.: Self-calibrating photometric stereo. In: CVPR (2010)
29. Wu, Z., Tan, P.: Calibrating photometric stereo by holistic reflectance symmetry analysis. In: CVPR (2013)
30. Papadhimitri, T., Favaro, P.: A closed-form, consistent and robust solution to uncalibrated photometric stereo via local diffuse reflectance maxima. IJCV **107**, 139–154 (2014)
31. Lu, F., Matsushita, Y., Sato, I., Okabe, T., Sato, Y.: From intensity profile to surface normal: photometric stereo for unknown light sources and isotropic reflectances. IEEE TPAMI **37**, 1999–2012 (2015)
32. Lu, F., Chen, X., Sato, I., Sato, Y.: SymPS: BRDF symmetry guided photometric stereo for shape and light source estimation. IEEE TPAMI **40**, 221–234 (2018)
33. Eigen, D., Puhrsch, C., Fergus, R.: Depth map prediction from a single image using a multi-scale deep network. In: NIPS (2014)
34. Wang, X., Fouhey, D., Gupta, A.: Designing deep networks for surface normal estimation. In: CVPR (2015)
35. Choy, C.B., Xu, D., Gwak, J., Chen, K., Savarese, S.: 3D–R2N2: a unified approach for single and multi-view 3D object reconstruction. In: ECCV (2016)
36. Hartmann, W., Galliani, S., Havlena, M., Van Gool, L., Schindler, K.: Learned multi-patch similarity. In: ICCV (2017)
37. Bromley, J., Guyon, I., LeCun, Y., Säckinger, E., Shah, R.: Signature verification using a "siamese" time delay neural network. Int.J. Pattern Recogn. Artif. Intell. (1993)
38. Jakob, W.: Mitsuba renderer (2010)
39. Paszke, A., Gross, S., Chintala, S., Chanan, G.: PyTorch: tensors and dynamic neural networks in python with strong GPU acceleration (2017)
40. Kingma, D., Ba, J.: Adam: a method for stochastic optimization. In: ICLR (2015)
41. Frankot, R.T., Chellappa, R.: A method for enforcing integrability in shape from shading algorithms. IEEE TPAMI **10**, 439–451 (1988)