Computational Visual Media
https://doi.org/10.1007/s41095-021-0223-y Vol. 8, No. 1, March 2022, 105–118
Research Article

Learning conditional photometric stereo with high-resolution features

Yakun Ju[1], Yuxin Peng[2], Muwei Jian[3], Feng Gao[1], and Junyu Dong[1] (✉)

Abstract Photometric stereo aims to reconstruct 3D geometry by recovering the dense surface orientation of a 3D object from multiple images under differing illumination. Traditional methods normally adopt simplified reflectance models to make the surface orientation computable. However, the real reflectances of surfaces greatly limit applicability of such methods to real-world objects. While deep neural networks have been employed to handle non-Lambertian surfaces, these methods are subject to blurring and errors, especially in high-frequency regions (such as crinkles and edges), caused by spectral bias: neural networks favor low-frequency representations so exhibit a bias towards smooth functions. In this paper, therefore, we propose a self-learning conditional network with multi-scale features for photometric stereo, avoiding blurred reconstruction in such regions. Our explorations include: (i) a multi-scale feature fusion architecture, which keeps high-resolution representations and deep feature extraction, simultaneously, and (ii) an improved gradient-motivated conditionally parameterized convolution (GM-CondConv) in our photometric stereo network, with different combinations of convolution kernels for varying surfaces. Extensive experiments on public benchmark datasets show that our calibrated photometric stereo method outperforms the state-of-the-art.

1 Department of Computer Science and Technology, Ocean University of China, Qingdao 266100, China. E-mail: Y. Ju, juyakun@stu.ouc.edu.cn; F. Gao, gaofeng@ouc.edu.cn; J. Dong, dongjunyu@ouc.edu.cn (✉).

2 Wangxuan Institute of Computer Technology, Peking University, Beijing 100871, China. E-mail: pengyuxin@pku.edu.cn.

3 School of Computer Science and Technology, Shandong University of Finance and Economics, Jinan 250002, China. E-mail: jianmuweihk@163.com.

Manuscript received: 2021-01-10; accepted: 2021-03-01

Keywords photometric stereo; normal estimation; deep neural networks; 3D reconstruction

1 Introduction

The goal of photometric stereo is to recover the dense surface orientation of a 3D object from varying shading cues, with a fixed camera, by establishing the relationship between two-dimensional images and the object geometry [1]. The earliest photometric stereo algorithm reconstructed the surface normal based on the Lambertian assumption [2]. Unfortunately, real-world objects hardly ever have the property of Lambertian reflectance, and therefore robust methods are needed to deal with objects with more general reflectance properties [3]. Traditional photometric stereo methods mainly address this problem by treating non-Lambertian regions as outliers [4, 5], or adopt bidirectional reflectance distribution functions (BRDFs) to model general reflectance [6, 7]. However, these traditional models are only accurate for limited categories of materials and suffer from unstable optimization.

Recently, deep learning frameworks have shown powerful capabilities for various tasks [8–10]. In particular, researchers have made efforts to learn general reflectance models through deep neural networks to solve the problem of photometric stereo. DPSN [11] first addressed non-Lambertian photometric stereo using a deep fully-connected network, to learn the surface normal in a per-pixel manner. Later, a series of methods employed convolutional neural networks (CNNs) to better utilize adjacent information embedded in images, such as PS-FCN [12], SDPS-Net [13], Manifold-PSN [14], and IRPS [15]. However, these methods suffer

from the blurring, especially in high-frequency regions (e.g., crinkles and edges). This phenomenon is caused by spectral bias [16], in which neural networks favor low-frequency representations so exhibit a bias towards smooth functions. Unfortunately, these regions are always those to which the human visual system pays attention and consequently should be reconstructed accurately. Existing photometric stereo networks pass the input through high-to-low resolution subnetworks that are connected in series, and then raise the resolution; these procedures cause the information loss and result in the blurring. Furthermore, existing photometric stereo networks employ the same learning strategy in all surface regions. The patterns we need to learn essentially vary from plain surfaces to high-frequency surfaces, and thus errors arise due to using the same learning strategy. Therefore, it remains urgent yet challenging to develop a robust and efficient photometric stereo method that can avoid blurring and accurately reconstruct objects' surface orientation.

In this paper, we propose a conditional deep neural network with a high-resolution structure, called CHR-PSN, for estimating the surface normals of objects. In contrast to existing methods, our framework reduces the error and blurring, especially for surfaces with high-frequency details. Extensive experiments on public datasets show that CHR-PSN achieves state-of-the-art performance. Our contributions are as follows.

Firstly, inspired by the High-resolution Net [17] for human pose estimation, we employ a parallel network structure for maintaining both deep features and high-resolution details of surface normals, for the first time. We show that high-resolution information in extracted features is essential to the per-pixel surface normal estimation task, a point which has not been explored in learning-based or data-driven photometric stereo.

Secondly, we investigate an improved gradient-motivated conditionally parameterized convolution module (GM-CondConv) [18] in the regression stage of our network, where frequency information in surface representations is integrated into the routing function. We show that the GM-CondConV module can regress the surface normal, with high-frequency details.

2 Related work

2.1 Background

The imaging model establishes the relationship between the surface normal $\boldsymbol{n} \in \mathbb{R}^3$ and visual observations $\boldsymbol{I}$ in a per-pixel manner. By introducing the general BRDF ρ of the object and illumination direction $\boldsymbol{l}$ with intensity e, photometric stereo recovers the surface orientation from a combination of multiple images with differing illumination directions, as follows:

$$I_j = e_j \rho\left(\boldsymbol{n}, \boldsymbol{l}_j\right) \max\left(\boldsymbol{n}^{\mathrm{T}} \boldsymbol{l}_j, 0\right) + \epsilon_j \tag{1}$$

where the subscript j indexes the input, $\max\left(\boldsymbol{n}^{\mathrm{T}} \boldsymbol{l}_j, 0\right)$ accounts for attached shadows, and ϵ accounts for noise (such as inter-reflections). To extend photometric stereo to work with unknown general BRDF ρ in practice, researchers have investigated different strategies. We divide them into non learning-based methods and deep leaning-based methods.

2.2 Non learning-based methods

Generally, traditional photometric stereo technologies aim to solve the ill-posed surface normal under unknown reflectance. Here, we briefly introduce these non learning-based photometric stereo techniques, divided as sophisticated reflectance methods and outlier rejection methods. More comprehensive surveys can be found in Refs. [19, 20]

Sophisticated reflectance methods are applied to model and approximate non-Lambertian reflectance. In this direction, many models have been proposed to fit nonlinear analytic BRDFs, such as bivariate functions [21, 22], the Ward reflectance model [23, 24], the specular spike reflectance model [25, 26], the Blinn–Phong reflectance model [27], and the Torrance–Sparrow reflectance model [28]. However, these sophisticated reflectance methods are generally useful for limited categories of surfaces as the reflectance properties significantly change from material to material.

Outlier rejection methods treat non-Lambertian regions (such as specularities and cast shadows) as outliers that should be discarded. A range of outlier rejection based photometric stereo algorithms have been proposed such as maximum-likelihood estimation [29], low rank approximation [4, 5], an RANSAC method [30], a maximum feasible

subsystem method [31], etc. However, these methods assume outliers to be local and sparse, and cannot handle surfaces with broad and soft specularities.

2.3 Deep learning-based methods

Inspired by the powerful fitting ability of deep neural networks [32, 33], deep learning-based methods have been introduced to solving the non-Lambertian photometric stereo problem. DPSN [11] first applied a fully-connected architecture for the non-Lambertian photometric stereo in a per-pixel manner. Some works use an observation map, which rearranges observed per-pixel intensities according to the light direction, to recover surface normals, such as CNN-PS [34], LMPS [35], and SPLINE-Net [36]. PS-FCN [12] and SDPS [13] employed a fully-convolutional network to learn the surface normal from input patches with neighborhood embedding. IRPS [15] further proposed an unsupervised learning framework that predicts surface normals by minimizing the loss of reconstructed images. However, existing networks pass the input through high-to-low resolution subnetworks connected in series, and then increase the resolution; these approaches cause blurring of predicted surface normals.

Recently, Attention-PSN [37] proposed an adaptive attention-weighted loss to improve the performance in various surface regions. Using the self-supervised weights of detail-preserving gradient loss, the method achieves better reconstruction results in high-frequency surface regions. However, we argue that the detail-preserving gradient loss can only constrain the high-frequency of surface structure but it is useless in terms of accuracy of predicted normal, i.e., the gradient loss dilutes supervision of the normal. Furthermore, Attention-PSN only uses the adaptive loss function to improve the details but ignores the impact of unsuitable kernels and receptive fields in the convolutional layers, which is the essential cause of blurring in high-frequency regions.

Other reconstruction approaches also address the frequency problem. Mildenhall et al. [38] proposed a method for synthesizing novel views of complex scenes by optimizing an underlying continuous volumetric scene function. This method represents high-frequency scene content, by using a positional encoding to map each input 5D coordinate into a higher dimensional space. Liu et al. [39] introduced a wavelet-based network to remove Moiré patterns, using the fact that high-frequency features may be highlighted in wavelet sub-bands.

3 Proposed method

In this section, we present our conditional deep photometric stereo network with high-resolution features. Our goal is to improve accuracy and reduce blurring of surface normal estimates. The architecture of the proposed CHR-PSN is shown in Fig. 1.

3.1 Network architecture

3.1.1 Feature extraction stage

As shown in Fig. 1, we first fuse the input images with their illumination direction in the feature fusion stage. For an object captured under j illumination directions, we expand each direction $\boldsymbol{l}_j$ to form a 3-channel image having the same spatial resolution as the input image ($H \times W \times 3$), and concatenate it with the corresponding image $\boldsymbol{I}_j$ as the $\boldsymbol{\Phi}_j \in \mathbb{R}^{H \times W \times 6}$.

The feature extraction stage of our network can be seen as the j-fold multi-branch shared-weight feature extraction network, which can be expressed as

$$\boldsymbol{\Psi}_j^{\mathrm{FR}}, \boldsymbol{\Psi}_j^{\mathrm{HR}}, \boldsymbol{\Psi}_j^{\mathrm{QR}} = F_{\mathrm{ext}}(\boldsymbol{\Phi}_j; \theta_{\mathrm{ext}}) \tag{2}$$

where F_{ext} is the multi-scale feature architecture with learnable parameters θ_{ext}, inspired by the High-resolution Net [17]. We employ a parallel network structure for extracting three scales of features, avoiding the feature map from low-resolution to high-resolution. Therefore, our feature extraction maintains both the deep features and high-resolution details of surface normals. As shown in Fig. 1, the down-sampling operations are executed through convolutional layers with stride 2 (double down-sampling) or 4 (twice double down-sampling), and the up-sampling operations are executed through bilinear-upsampling and 1×1 convolutional layers to adjust the channel of the feature to be the same as the high-resolution feature channel. The fusion of high-to-low and low-to-high processes into the same-resolution features is executed through skip connections. Therefore, our feature extraction method outputs three different resolution features, as full resolution (FR): $\boldsymbol{\Psi}_j^{\mathrm{FR}} \in \mathbb{R}^{H \times W \times 64}$, half resolution (HR): $\boldsymbol{\Psi}_j^{\mathrm{HR}} \in \mathbb{R}^{H/2 \times W/2 \times 128}$, and quarter resolution (QR): $\boldsymbol{\Psi}_j^{\mathrm{QR}} \in \mathbb{R}^{H/4 \times W/4 \times 256}$.

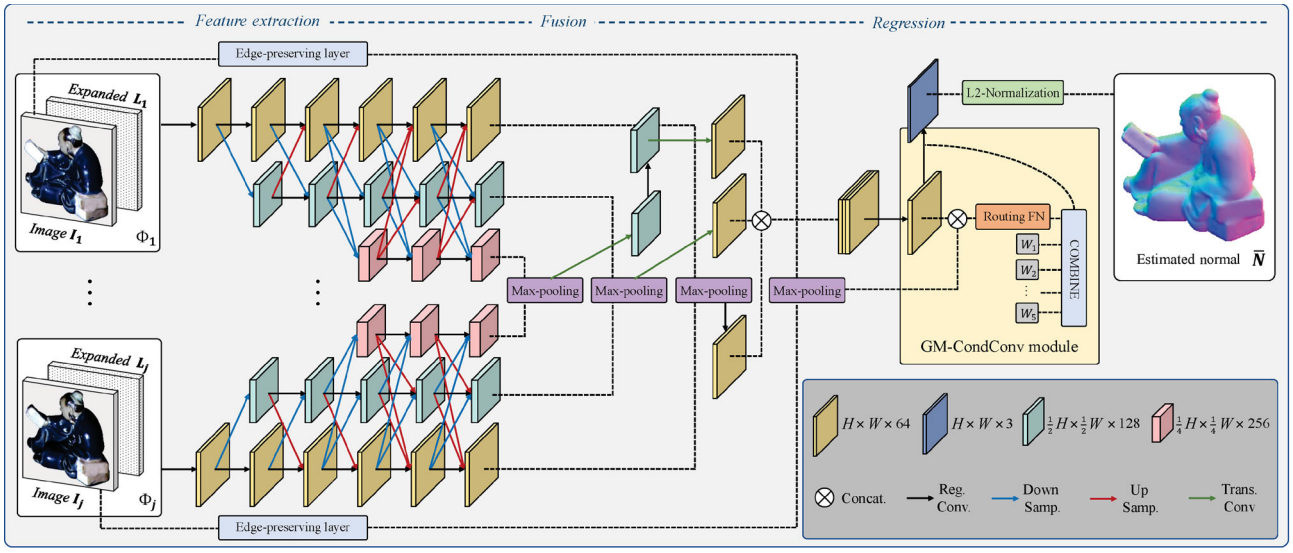


Fig. 1 Architecture of our CHR-PSN. Reg. Conv. = regular convolution, Down Samp. = down-sampling, Up Samp. = up-sampling, Trans. Conv = transposed convolution. Leaky-ReLU is the activation function for each layer. Our network has three stages, for feature extraction, fusion, and regression. Given an arbitrary number of images with different lighting directions, we first extract multi-scale features and represent the edge features. Multi-resolution max-pooling operations are applied for fusion. Finally, an improved GM-CondConv module infers surface normals.

We also introduce an edge-preserving layer for each $\boldsymbol{I}_j$ as follows:

$$\boldsymbol{\Omega}_j^{\mathrm{FR}} = F_{\mathrm{edge}}(\boldsymbol{I}_j) \tag{3}$$

where F_{edge} is the edge-preserving layer, calculated as the gradient of input image $\boldsymbol{I}_j$. $\boldsymbol{\Omega}_j^{\mathrm{FR}} \in \mathbb{R}^{H\times W\times 3}$ is the output with high-frequency edge information, which is used in the improved CondConv module of the regression stage.

3.1.2 Fusion stage

In the fusion stage, we apply multi-scale max-pooling operations [12, 37] to fuse the j features into one, so our network can handle an arbitrary number of inputs and backpropagate the parameters. We argue that max-pooling extracts the most salient information from all features, while average-pooling may smooth out useful features and be impacted by non-activated features. Here, the subscript p indexes position in the feature:

$$\boldsymbol{\Omega}_{\max}^{\mathrm{FR}} = \bigcup_{p}^{H\times W} \max(\boldsymbol{\Omega}_{1p}^{\mathrm{FR}}, \cdots, \boldsymbol{\Omega}_{jp}^{\mathrm{FR}}) \tag{4}$$

$$\boldsymbol{\Psi}_{\max}^{\mathrm{FR}} = \bigcup_{p}^{H\times W} \max(\boldsymbol{\Psi}_{1p}^{\mathrm{FR}}, \cdots, \boldsymbol{\Psi}_{jp}^{\mathrm{FR}}) \tag{5}$$

$$\boldsymbol{\Psi}_{\max}^{\mathrm{HR}} = \bigcup_{p}^{\frac{1}{2}H\times\frac{1}{2}W} \max(\boldsymbol{\Psi}_{1p}^{\mathrm{HR}}, \cdots, \boldsymbol{\Psi}_{jp}^{\mathrm{HR}}) \tag{6}$$

$$\boldsymbol{\Psi}_{\max}^{\mathrm{QR}} = \bigcup_{p}^{\frac{1}{4}H\times\frac{1}{4}W} \max(\boldsymbol{\Psi}_{1p}^{\mathrm{QR}}, \cdots, \boldsymbol{\Psi}_{jp}^{\mathrm{QR}}) \tag{7}$$

where $\boldsymbol{\Omega}_{\max}^{\mathrm{FR}}$, $\boldsymbol{\Psi}_{\max}^{\mathrm{FR}}$, $\boldsymbol{\Psi}_{\max}^{\mathrm{HR}}$, and $\boldsymbol{\Psi}_{\max}^{\mathrm{QR}}$ are the fused features.

3.1.3 Regression stage

The normal regression stage takes $\boldsymbol{\Omega}_{\max}^{\mathrm{FR}}$, $\boldsymbol{\Psi}_{\max}^{\mathrm{FR}}$, $\boldsymbol{\Psi}_{\max}^{\mathrm{HR}}$, and $\boldsymbol{\Psi}_{\max}^{\mathrm{QR}}$ as inputs and regresses the predicted surface normals $\bar{\boldsymbol{N}}$, by F_{reg} with learnable parameters θ_{reg}, as follows:

$$\bar{\boldsymbol{N}} = F_{\mathrm{reg}}(\boldsymbol{\Omega}_{\max}^{\mathrm{FR}}, \boldsymbol{\Psi}_{\max}^{\mathrm{FR}}, \boldsymbol{\Psi}_{\max}^{\mathrm{HR}}, \boldsymbol{\Psi}_{\max}^{\mathrm{QR}}; \theta_{\mathrm{ext}}) \tag{8}$$

In the regression stage, we first employ transposed convolution operations to up-sample the low-resolution feature $\boldsymbol{\Psi}_{\max}^{\mathrm{HR}}$ and $\boldsymbol{\Psi}_{\max}^{\mathrm{QR}}$ to the full resolution of $H \times W$ (twice transposed convolution and once regular convolution for $\boldsymbol{\Psi}_{\max}^{\mathrm{QR}}$, once transposed convolution for $\boldsymbol{\Psi}_{\max}^{\mathrm{HR}}$). As shown in Fig. 1, we employ concatenation to fuse the two up-sampled features and the full resolution feature, instead of using skip connections in the feature extraction stage.

To better reconstruct details of objects and remove blurring in high-frequency regions, we propose an improved GM-CondConv module in the regression stage [18], with the motivation that previous methods use the same learning strategy for all surface regions, causing blurring and error. By parameterizing the convolutional kernel conditionally on the input, the network can give accurate estimates for both simple surface regions and high-frequency surface regions (crinkles, edges). Particularly, we concatenate

the high-frequency edge information $\boldsymbol{\Omega}^{\mathrm{FR}}_{\max}$ with the previous layer feature $\boldsymbol{x}$. We argue that the frequency information is beneficial to the classification of each learned kernel, which is better used to predict different surface normal regions. Therefore, the convolutional kernels in GM-CondConv are parameterized as

$$\text{GM-CondConv}(\boldsymbol{x}, \boldsymbol{\Omega}^{\mathrm{FR}}_{\max}) = \sigma\left((\boldsymbol{\alpha}_1 \cdot \boldsymbol{W}_1 + \cdots + \boldsymbol{\alpha}_n \cdot \boldsymbol{W}_n) * (\boldsymbol{x}, \boldsymbol{\Omega}^{\mathrm{FR}}_{\max})\right) \quad (9)$$

where each $\boldsymbol{\alpha}_i = r_i(x, \boldsymbol{\Omega}^{\mathrm{FR}}_{\max})$ is an example-dependent scalar weight computed using a routing function with learned parameters, n is the number of weights ($n = 5$ in our default setting), and σ is the Leaky-ReLu activation function. Following CondConv [18], we compute example-dependent routing weights $\boldsymbol{\alpha}_i = r_i(x, \boldsymbol{\Omega}^{\mathrm{FR}}_{\max})$ from the layer input in three steps: global average pooling, a fully-connected layer, and sigmoid activation:

$$r(\boldsymbol{x}, \boldsymbol{\Omega}^{\mathrm{FR}}_{\max}) = \text{Sigmoid}(\text{GlobalAveragePool}(\boldsymbol{x}, \boldsymbol{\Omega}^{\mathrm{FR}}_{\max})\ \boldsymbol{R}) \quad (10)$$

where $\boldsymbol{R}$ is a matrix of learned routing weights mapping the pooled inputs to n expert weights. We finally employ $L2$ normalization of the predictions giving $\bar{\boldsymbol{N}}$.

3.2 Loss function and training procedure

Learning in our network is supervised by the angular error between the estimated and the ground-truth surface normals. We optimize network parameters θ_{ext} and θ_{reg} by minimizing the cosine similarity loss:

$$\mathcal{L}_{\mathrm{normal}} = \frac{1}{HW}\sum_{p}^{HW}\left(1 - \bar{\boldsymbol{N}}_p \cdot \boldsymbol{N}_p\right) \quad (11)$$

where $\bar{\boldsymbol{N}}_p$ and $\boldsymbol{N}_p$ denote the estimated and ground-truth normals respectively at pixel p. If the estimated normal $\bar{\boldsymbol{N}}_p$ at pixel p has similar orientation to the ground-truth $\boldsymbol{N}_p$, then $\bar{\boldsymbol{N}}_p \cdot \boldsymbol{N}_p$ will be close to 1 and the loss $\mathcal{L}_{\mathrm{normal}}$ will approach 0.

Our network is implemented in PyTorch [40] on an RTX 2080Ti GPU, and the Adam optimizer [41] is used with default settings, with the learning rate initially set to 0.001 and divided by 2 every 5 epochs. We train the model using a batch size of 32 for 40 epochs, with $j = 32$ for each sample in training, while our network can accept an arbitrary number of j in testing. Also, we set the resolution to $H = W = 32$ in training; an arbitrary resolution can be used in testing.

3.3 Datasets

3.3.1 Training and validation datasets

We adopt two public synthetic blobby shape [42] and sculpture shape datasets [43] to train our network. Following the setup in PS-FCN [12], we render these two shape datasets with the MERL dataset [44], which contains 100 different BRDFs of real-world materials, using the physically-based raytracer Mitsuba [45]. Their resolution is 128×128. Image patches of size 32×32 are randomly cropped for data augmentation. This results in 85,212 samples in total, each sample containing 64 images with different illumination directions (random directions across the upper hemisphere). We split the samples into a training set (84,360 samples) and a validation set (852 samples).

3.3.2 Testing datasets

We use public non-Lambertian photometric stereo datasets to evaluate our method. First, we employ the DiLiGenT benchmark dataset [19]. It contains 10 objects of various shapes with complex materials. For each object, the dataset provides 96 images under different illumination directions, at a resolution of 612×512. Then, we employ the Light Stage Data Gallery dataset [46]. It contains six complex objects with higher resolution. Each object has up to 253 images under different illumination directions. Note that this dataset lacks the ground-truth surface normal. Therefore we qualitatively evaluate our method on it.

4 Experimental results

We present experiments and analysis in this section.

4.1 Metrics

To verify the quantitative performance of our method, we employ widely used metrics to measure accuracy. We adopt the mean angular error (MAE) in degrees to evaluate the accuracy of the estimated surface normal:

$$\mathrm{MAE} = \frac{1}{HW}\sum_{p}^{H\times W}(\cos^{-1}(\bar{\boldsymbol{N}}_p \cdot \boldsymbol{N}_p)) \quad (12)$$

We also measure the percentage (%) of pixels with angular error less than 20°, which is denoted by $< err_{20^\circ}$. This metric better measures high-frequency error, as the normal error in high-frequency regions is bigger.

4.2 Ablation experiments and network analysis

4.2.1 Procedure

We performed quantitative ablation experiments on the validation set, reporting the average MAE of its 852 samples (tested with 32 images). Table 1 summarizes the results of the ablation experiments. Our default method is marked as D0, with full resolution features + $\frac{1}{2}$ resolution features + $\frac{1}{4}$ resolution features in the high-resolution feature extraction stage [17], as well as fusion of high-frequency edge information $\mathbf{\Omega}^{\mathrm{FR}}_{\max}$ and 5 weights in the GM-CondConv module of the regression stage. We first evaluate the effectiveness of multi-scale features: experiments D0, M1, M2, M3, and M4 combine different resolution features. For M1, M2, M3, and M4, we adjust the architecture of the feature extraction network, the corresponding multi-scale max-pooling fusion, and the number of concatenations in the regression stage, but maintain the GM-CondConv module unchanged. Note that the $\frac{1}{8}$ resolution feature in M4 has dimensions $\mathbb{R}^{\frac{1}{8}H\times\frac{1}{8}W\times 512}$. For the network without full resolution features, we down sample at the beginning. We then evaluate the effectiveness of the improved GM-CondConv module (experiments D0, C5, C6, C7, and C8). We test the impact of fusing edge information, and the number of weights of routing function in the GM-CondConv module. For C5, C6, C7, and C8, we only adjust the GM-CondConv module but maintain the architecture of the high-resolution network unchanged. Finally, we evaluate different methods of fusing illumination direction (experiments D0 and L9). Our default setting uses a concatenation operation to fuse the input images and illumination directions. For L9, we test the performance of adding the value of each element instead of concatenation. In this case, we adjust the number of input channels of the first convolutional layer from 6 to 3, in the feature extraction stage.

Table 1 MAE and $<err_{20^\circ}$ errors using different components of CHR-PSN on the validation set (with 32 input images)

ID	Variants	MAE	$<err_{20^\circ}$
D0	Our default settings	11.91	**85.38%**
M1	Full Resolution	12.15	83.49%
M2	Full Resolution + $\frac{1}{2}$ Resolution	11.97	84.13%
M3	$\frac{1}{2}$ Resolution + $\frac{1}{4}$ Resolution	12.69	80.83%
M4	Full Resolution + $\frac{1}{2}$ Resolution + $\frac{1}{4}$ Resolution+ $\frac{1}{8}$ Resolution	**11.90**	85.35%
C5	Without $\mathbf{\Omega}^{\mathrm{FR}}_{\max}$	11.99	84.65%
C6	Weight number = 1 (Regular Conv)	12.02	84.79%
C7	Weight number = 3	11.95	85.05%
C8	Weight number = 7	11.92	85.21%
L9	Element add	14.52	75.30%

4.2.2 Effectiveness of multi-scale features

Experiments D0, M1, M2, M3, and M4 compare the performance of different combinations of feature resolution. Note that M1 has only full resolution features, which can be seen as a fully convolutional network without up and down sampling. It can be seen that multi-scale resolution features are beneficial to the accuracy of prediction. Clearly, when the network has lower resolution features (M3), the performance is worse. It shows that the resolution of features has a crucial impact on the performance of the model in the per-pixel surface normal recovery task. Unfortunately, previous deep learning-based photometric stereo methods are similar to M3 in lacking high-resolution feature branches. Also, comparing D0 with M1, M2, and M4, we can see that deep features improve the performance of prediction to some extent. However, the improvement is quite slight after adding $\frac{1}{8}$ resolution features, while the $\frac{1}{8}$ resolution features significantly increase the number of parameters and training time. This might be because such deep features contain less detail information but high-level semantic information, which is useless for the per-pixel prediction task. Therefore, we select full resolution features + $\frac{1}{2}$ resolution features + $\frac{1}{4}$ resolution features in the high-resolution feature extraction stage [17].

4.2.3 Effectiveness of fusing high-frequency information in routing

Experiments D0, C5 show the influence of fusing high-frequency edge information $\mathbf{\Omega}^{\mathrm{FR}}_{\max}$ in the routing function of the GM-CondConv module. We can see that the angular error and $<err_{20^\circ}$ of the validation set are lower when edge information is taken into account. This might be explained by the fact that the improved routing function incorporates high-frequency information into the self-learned weights, which is beneficial to the GM-CondConv module for estimating different frequency surface regions

(such as crinkles and planar parts). We also show a "Buddha" example in Fig. 2. The comparison between CondConv (C5) and GM-CondConv (D0) shows that using GM-CondConv improves the performance in high-frequency areas.

4.2.4 Choice of number of weights in GM-CondConv

In experiments D0, C6, C7, our method increased the number of weights in GM-CondConv. Note that with one weight there is only one convolution kernel and no dynamic weight. These comparisons show the effectiveness of our improved GM-CondConv module. Also, compared with default settings, adding further weights to GM-CondConv does not continue to improve accuracy. Our method performs best when 5 weights are used, according to the above experiments.

4.2.5 Effectiveness of illumination direction fusion methods

Experiments D0, L9 show the influence of different fusion methods. Angular error and $<err_{20^\circ}$ for the validation set are best when using concatenation (our default). The performance of prediction severely decreases when using the add operation between the input image and the illumination direction. We argue that the network can hardly decouple features that are numerically added into image and illumination.

4.3 Evaluation on the DiLiGenT benchmark

4.3.1 Evaluation on 96 input images

We compare our method with both non learning-based methods and recent deep learning-based methods in terms of achieved MAE, on the DiLiGenT benchmark [19]. As non learning-based methods, we evaluate the least squares (baseline) method [2], rank minimization [4], and matrix rank = 3 [5] of the outlier rejection method. We also evaluate sophisticated reflectance methods, such as Multi-Ward models [23], bivariate BRDF [6], and a bi-polynomial method [47]. For deep learning-based methods, we compared our method to DPSN [11], IRPS [15], PS-FCN [12], and Attention-PSN [37] using 96 input images. Quantitative results are reported in Table 2. Figure 3 visualizes results for the four most accurate deep learning-based photometric stereo methods: Attention-PSN [37], PS-FCN [12], IRPS [15], and DPSN [11], as well as the baseline least squares method [2]. Figure 3 illustrates the performance of our method in high-frequency regions, such as the face of "Buddha" and the flower in "Pot2", and cast shadows regions, such as the shoulder of "Buddha" and the base of "Goblet". It can be seen that our method is more accurate in regions with cast shadows and crinkles.

We also show details in an enlargement of part of "Buddha" in Fig. 2. We can see that the last three comparisons, which take high-frequency information into consideration, achieve much better accuracy on crinkles and edges. Specifically, our default settings (using improved GM-CondConv) result in

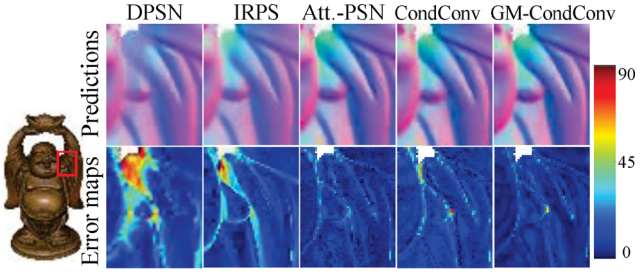


Fig. 2 An enlargement from "Buddha" from the DiLiGenT dataset [19]. Att.-PSN: Attention-PSN. CondConv represents using the original CondConv module [18] (ID = C5 in Table 1), while GM-CondConv represents our default model.

Table 2 MAE (in degree) for different methods on the DiLiGenT benchmark. All methods were evaluated with 96 images

Method	Ball	Bear	Buddha	Cat	Cow	Goblet	Harvest	Pot1	Pot2	Reading	Avg.
Baseline	4.10	8.39	14.92	8.41	25.60	18.50	30.62	8.89	14.65	19.80	15.39
Matrix rank = 3	2.54	7.32	11.11	7.21	25.70	16.25	29.26	7.74	14.09	16.17	13.74
Rank minimization	2.06	6.50	10.91	6.73	25.89	15.70	30.01	7.18	13.12	15.39	13.35
Multi-Ward models	3.21	6.62	14.85	8.22	9.55	14.22	27.84	8.53	7.90	19.07	12.00
Bivariate BRDF	3.34	7.11	10.47	6.74	13.05	9.71	25.95	6.64	8.77	14.19	10.60
Bi-polynomial	1.74	6.12	10.60	6.12	13.93	10.09	25.44	6.51	8.78	13.63	10.30
DPSN	2.02	6.31	12.68	6.54	8.01	11.28	16.86	7.05	7.86	15.51	9.41
IRPS	**1.47**	5.79	10.36	**5.44**	6.32	11.47	22.59	**6.09**	7.76	**11.03**	8.83
PS-FCN	2.82	7.55	7.91	6.16	7.33	8.60	15.85	7.13	7.25	13.33	8.39
Attention-PSN	2.93	**4.86**	7.75	6.14	6.86	8.42	15.44	6.92	6.97	12.90	7.92
CHR-PSN (ours)	2.26	6.35	**7.15**	5.97	**6.05**	**8.32**	**15.32**	7.04	**6.76**	12.52	**7.77**

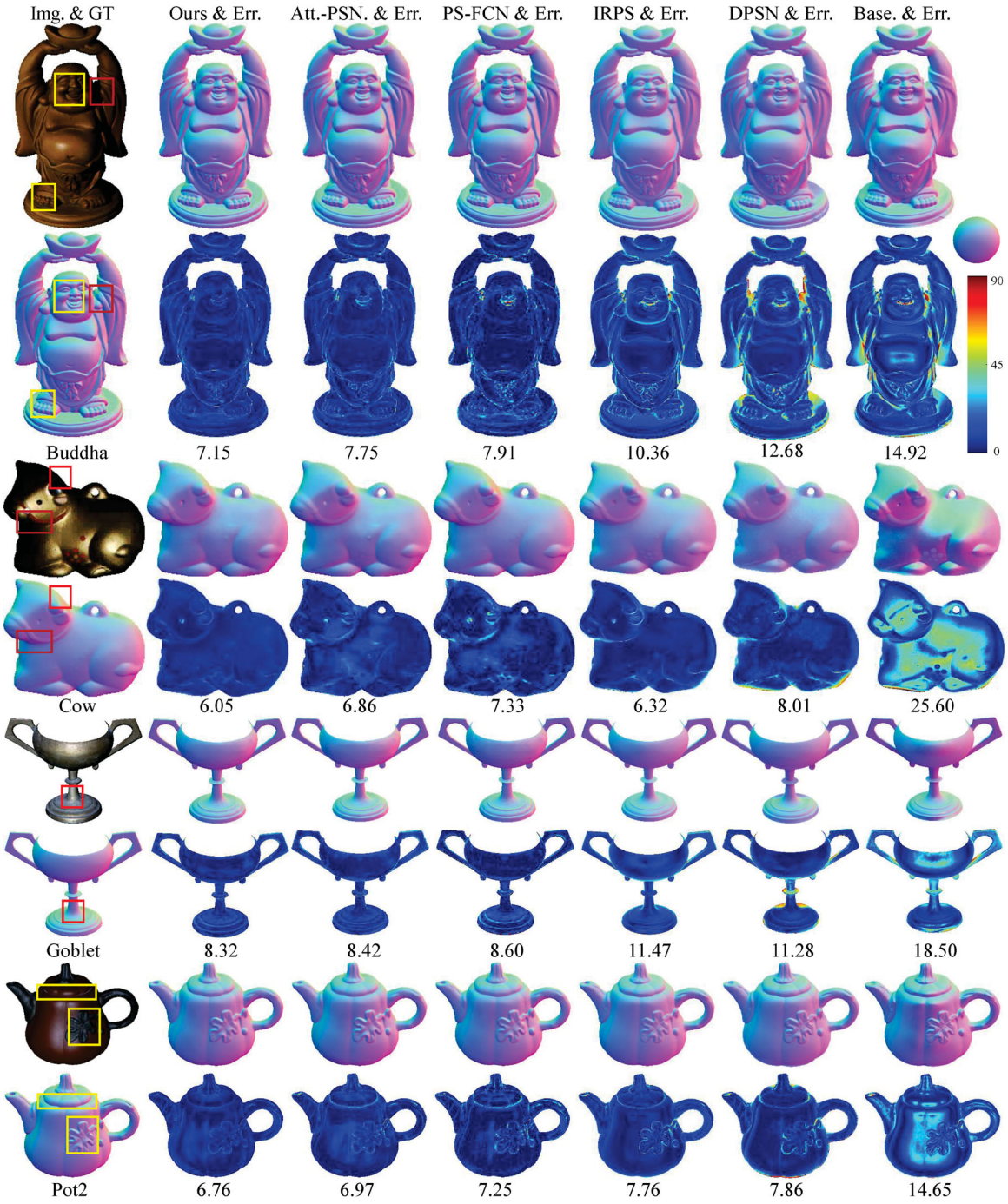


Fig. 3 Comparison using Buddha, Cow, Goblet, and Pot2 from the DiLiGenT benchmark. Yellow boxes: regions with high-frequency surfaces (such as crinkles). Red boxes: regions with cast shadows. Att.-PSN: Attention-PSN [37]. Base.: Baseline least squares method [2]. Contrast is adjusted for ease of viewing.

reduced error in high-frequency areas, compared to using CondConv module (without high-frequency information $\mathbf{\Omega}_{\max}^{\mathrm{FR}}$ in the routing function, C5 in Table 1).

4.3.2 Limitations

Our CHR-PSN method does not achieve the best performance on some objects, such as "Ball" and "Bear". We also illustrate some failures in Fig. 4. For these objects, our method provides sub-optimal performance. Objects like Ball and Bear have smooth surface normals and approximately Lambertian reflectance. In these cases, we argue that the high-resolution feature extraction of our method and GM-CondConv module are excessive. IRPS [15]

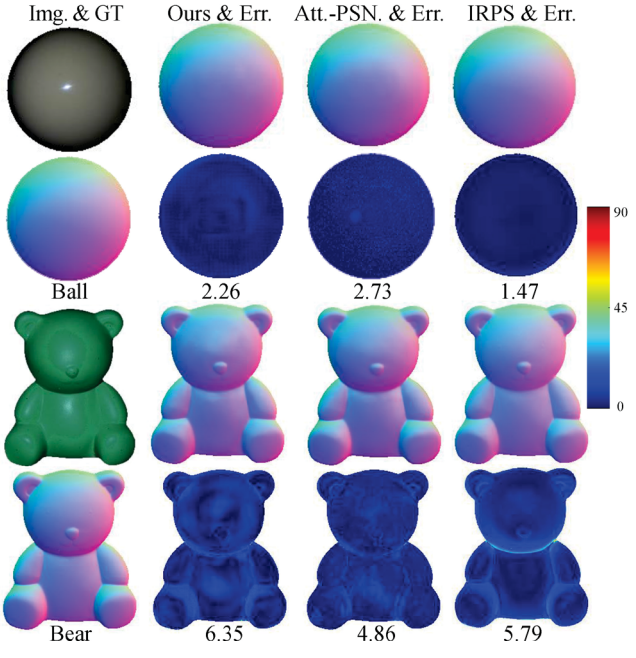


Fig. 4 Quantitative results on Ball and Bear from the DiLiGenT benchmark. Contrast is adjusted for ease of viewing.

performs very well on these objects because it introduces the reconstruction loss to learn the surface normal, where an approximate Lambertian surface and simple structure is beneficial to the inverse rendering. However, we can see that our method still outperforms Attention-PSN and IRPS in non-Lambertian regions (such as the specularity of "Ball") and cast shadows regions (such as the chin of "Bear").

4.3.3 Evaluation using fewer input images

We further evaluated our method against several methods with sparse inputs (10 input images). Our method employs max-pooling to handle an arbitrary number of input images, which is of practical use. For non learning-based methods, we evaluate the least squares baseline method [2], the bi-polynomial [47], and matrix rank = 3 [5]. For deep learning-based methods, we evaluate CNN-PS [34], SPLINE-Net [36], LMPS [35], and PS-FCN [12]. We summarize the comparisons in Table 3.

It can be seen that our method outperforms others on average MAE using the DiLiGenT dataset and achieves state-of-the-art accuracy on most objects. We also visualize the average MAE of the DiLiGenT dataset from sparse input (8) to dense input (96), as shown in Fig. 5. We compare our method to PS-FCN [12], which also uses max-pooling to handle differing numbers of input images with a single round of training. We can see that our method outperforms PS-FCN on all numbers of input images (both methods were trained with 32 input images).

4.3.4 Extension to uncalibrated photometric stereo

We next report the superior performance of our method in uncalibrated conditions. In actual applications, there are conditions where the directions of illuminations $\boldsymbol{l}_j$ are unknown. Our method can be easily extended to handle uncalibrated photometric stereo by removing the illumination direction from the input (as the $\boldsymbol{\Phi}_j \in \mathbb{R}^{H\times W\times 3}$, which only includes the RGB-channel image). To verify the potential of our method, we trained the model without illumination directions (also using 32 images for one

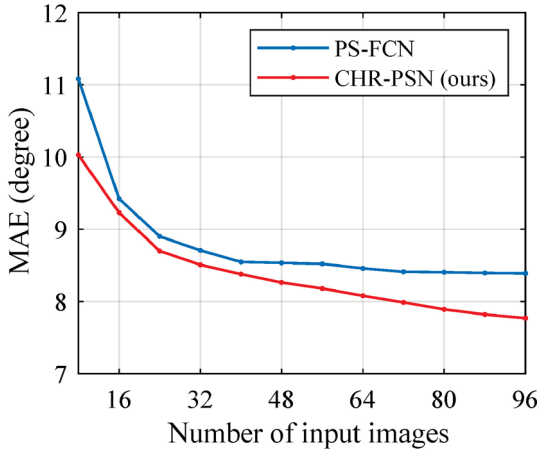


Fig. 5 MAE for different numbers of input images.

Table 3 MAE (in degree) for different methods using the DiLiGenT benchmark, with 10 input images

Method	Ball	Bear	Buddha	Cat	Cow	Goblet	Harvest	Pot1	Pot2	Reading	Avg.
Baseline	5.09	11.59	16.25	9.66	27.90	19.97	33.41	11.32	18.03	19.86	17.31
Bi-polynomial	5.24	9.39	15.79	9.34	26.08	19.71	30.85	9.76	15.57	20.08	16.18
Matrix rank = 3	**3.33**	7.62	13.36	8.13	25.01	18.01	29.37	8.73	14.60	16.63	14.48
CNN-PS	9.11	14.08	14.58	11.71	14.04	15.48	19.56	13.23	14.65	16.99	14.34
PS-FCN	4.02	7.18	9.79	8.80	10.51	11.58	18.70	10.14	9.85	15.03	10.51
SPLINE-Net	4.96	**5.99**	10.07	7.52	8.80	10.43	19.05	8.77	11.79	16.13	10.35
LMPS	3.97	8.73	11.36	**6.69**	10.19	10.46	17.33	**7.30**	9.74	14.37	10.02
CHR-PSN (ours)	3.91	7.84	**9.59**	8.10	**8.54**	**10.36**	**17.21**	9.65	**9.61**	**14.35**	**9.92**

Table 4 MAE (in degree) for uncalibrated photometric stereo on the DiLiGenT benchmark, using 96 images

Method	Ball	Bear	Buddha	Cat	Cow	Goblet	Harvest	Pot1	Pot2	Reading	Avg.
Entropy minimization	7.27	16.81	32.81	31.45	54.72	46.54	61.70	18.37	49.16	53.65	37.25
Self-calibrating	8.90	11.98	15.54	19.84	22.73	48.79	73.86	16.68	50.68	26.93	29.59
Reflectance symmetry	**4.39**	**6.42**	13.19	36.55	19.75	20.57	55.51	**9.39**	14.52	58.96	23.93
Diffuse maxima	4.77	9.07	14.92	**9.54**	19.53	29.93	29.21	9.51	15.90	24.18	16.66
UPS-FCN	6.62	11.23	15.87	14.68	**11.91**	20.72	27.79	13.98	14.19	23.26	16.02
Ours (uncalibrated)	5.61	10.80	**12.48**	13.95	12.44	**17.84**	**23.39**	13.62	**13.79**	**20.78**	**14.47**
SDPS-Net	2.77	6.89	8.97	8.06	8.48	11.91	17.43	8.14	7.50	14.90	9.51

sample) and tested it on the DiLiGenT benchmark [19] with 96 images. The results are reported in Table 4. We compare our method (uncalibrated) with several uncalibrated photometric stereo methods, such as entropy minimization [48], a self-calibrating method [49], reflectance symmetry [50], diffuse maxima [51], and UPS-FCN (for uncalibrated)[13]. Our method (uncalibrated) outperformed existing methods in terms of the average MAE, except for SDPS-Net [13]. SDPS-Net is specially designed for uncalibrated conditions (solely learning the illumination direction), while our method can be both used in both calibrated and uncalibrated conditions.

4.4 Evaluation on the Light Stage Data Gallery dataset

We further qualitatively evaluated our method on a more complex dataset with general non-Lambertian materials. Figure 6 shows the results of our method (tested with a random sample of 150 of 253 total images) on objects "Kneeling", "Helmet", and "Standing". We show qualitative outcomes in this experiment, due to the absence of ground-truth surface normals. Due to limited GPU memory, we tested the Light Stage Data Gallery with 64 input images (calibrated illumination directions).

As shown in Fig. 6, the estimated normal keeps the details without blurring, such as in the hair of Kneeling, and screws of the Helmet. The predicted surface normal and 3D reconstruction convincingly reflect the shapes of the objects, with accurate detail. The belt of Kneeling further illustrates our performance on cast shadows. However, we also note that the predicted surface normal of the object Kneeling has some blurring and noise. We argue that the poor quality observed in Kneeling is due to high-frequency noise, which may affect the GM-CondConv module of our method.

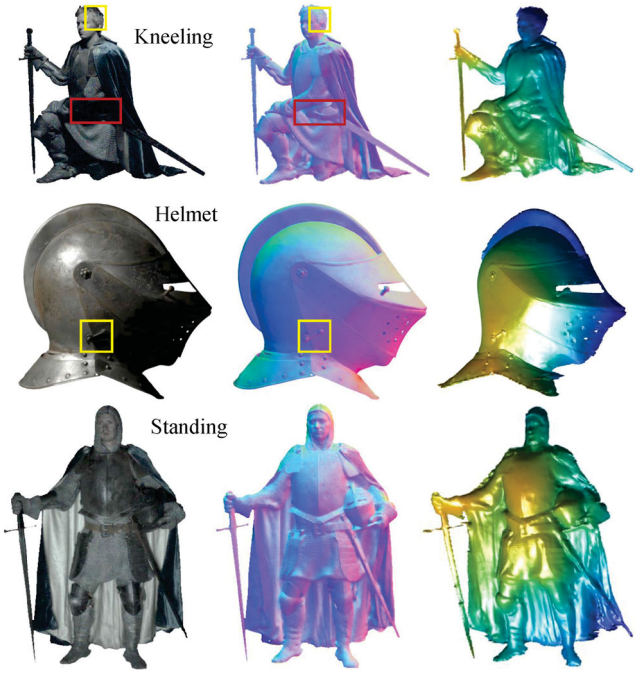


Fig. 6 Qualitative results of our method on objects Kneeling, Helmet, and Standing. Yellow boxes: regions with high-frequency surfaces (such as crinkles). Red boxes: regions with cast shadows. Contrast is adjusted for ease of viewing. After predicting surface normals, 3D reconstructions are recovered by Ref. [52].

5 Conclusions

In this paper, we have proposed a conditional photometric stereo network with a high-resolution feature extraction architecture. Compared to previous deep learning approaches which regress surface normals from a down-sampled feature map, we employ a multi-scale parallel architecture to enhance the details in predictions. Furthermore, we employ an improved GM-ConvCond module in the regression stage which considers the frequency of surfaces. As a result, our method outperforms others in high-frequency regions such as crinkles and edges. Ablation experiments have illustrated that our method performs more accurate reconstruction.

Extensive quantitative and qualitative comparisons on the DiLiGenT benchmark and the Light Stage Data Gallery have shown that our method outperforms state-of-the-art methods. Despite offering state-of-the-art performance, our method can be further improved. Firstly, our method provides sub-optimal results on some objects with very simple structure, in which cases the high-resolution feature extraction and GM-CondConv are excessive. Secondly, the training time of our method is longer than for other deep learning-based photometric stereo methods, due to our much bigger network architecture. In future, we will further design the feature extractor architecture to be better and predict the surface normal faster.

Acknowledgements

This work was supported by the National Key Scientific Instrument and Equipment Development Projects of China (41927805), the National Natural Science Foundation of China (61501417, 61976123), the Key Development Program for Basic Research of Shandong Province (ZR2020ZD44), and the Taishan Young Scholars Program of Shandong Province.

References

[1] Jian, M. W.; Dong, J. Y.; Gong, M. G.; Yu, H.; Nie, L. Q.; Yin, Y. L.; Lam, K.-M. Learning the traditional art of Chinese calligraphy via three-dimensional reconstruction and assessment. *IEEE Transactions on Multimedia* Vol. 22, No. 4, 970–979, 2020.

[2] Woodham, R. J. Photometric method for determining surface orientation from multiple images. *Optical Engineering* Vol. 19, No. 1, 191139, 1980.

[3] Khanian, M.; Boroujerdi, A. S.; Breuß, M. Photometric stereo for strong specular highlights. *Computational Visual Media* Vol. 4, No. 1, 83–102, 2018.

[4] Wu, L.; Ganesh, A.; Shi, B.; Matsushita, Y.; Wang, Y.; Ma, Y. Robust photometric stereo via low-rank matrix completion and recovery. In: Proceedings of the Asian Conference on Computer Vision, 703–717, 2010.

[5] Ikehata, S.; Wipf, D.; Matsushita, Y.; Aizawa, K. Robust photometric stereo using sparse regression. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 318–325, 2012.

[6] Ikehata, S.; Aizawa, K. Photometric stereo using constrained bivariate regression for general isotropic surfaces. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2187–2194, 2014.

[7] Higo, T.; Matsushita, Y.; Ikeuchi, K. Consensus photometric stereo. In: Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 1157–1164, 2010.

[8] Wei, K.; Yang, M. L.; Wang, H.; Deng, C.; Liu, X. L. Adversarial fine-grained composition learning for unseen attribute-object recognition. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, 3740–3748, 2019.

[9] Yang, X.; Deng, C.; Liu, T. L.; Tao, D. C. Heterogeneous graph attention network for unsupervised multiple-target domain adaptation. *IEEE Transactions on Pattern Analysis and Machine Intelligence* doi: 10.1109/TPAMI.2020.3026079, 2020.

[10] Wei, K.; Deng, C.; Yang, X. Lifelong zero-shot learning. In: Proceedings of the 29th International Joint Conference on Artificial Intelligence, 551–557, 2020.

[11] Santo, H.; Samejima, M.; Sugano, Y.; Shi, B. X.; Matsushita, Y. Deep photometric stereo network. In: Proceedings of the IEEE International Conference on Computer Vision Workshops, 501–509, 2017.

[12] Chen, G. Y.; Han, K.; Wong, K. Y. K. PS-FCN: A flexible learning framework for photometric stereo. In: *Computer Vision – ECCV 2018. Lecture Notes in Computer Science, Vol. 11213*. Ferrari, V.; Hebert, M.; Sminchisescu, C.; Weiss, Y. Eds. Springer Cham, 3–19, 2018.

[13] Chen, G.; Han, K.; Shi, B.; Matsushita, Y.; Wong, K.-Y. K. Self-calibrating deep photometric stereo networks. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 8739–8747, 2019.

[14] Ju, Y. K.; Jian, M. W.; Dong, J. Y.; Lam, K. M. Learning photometric stereo via manifold-based mapping. In: Proceedings of the IEEE International Conference on Visual Communications and Image Processing, 411–414, 2020.

[15] Taniai, T.; Maehara, T. Neural inverse rendering for general reflectance photometric stereo. In: Proceedings of the 35th International Conference on Machine Learning, 4857–4866, 2018.

[16] Rahaman, N.; Baratin, A.; Arpit, D.; Draxler, F.; Lin, M.; Hamprecht, F.; Bengio, Y.; Courville, A. On the spectral bias of neural networks. In: Proceedings of the International Conference on Machine Learning, 5301–5310, 2019.

[17] Sun, K.; Xiao, B.; Liu, D.; Wang, J. D. Deep high-resolution representation learning for human pose estimation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 5686–5696, 2019.

[18] Yang, B.; Bender, G.; Le, Q. V.; Ngiam, J. Condconv: Conditionally parameterized convolutions for efficient inference. In: Proceedings of the Advances in Neural Information Processing Systems, 1307–1318, 2019.

[19] Shi, B.; Mo, Z.; Wu, Z.; Duan, D.; Yeung, S.; Tan, P. A benchmark dataset and evaluation for non-Lambertian and uncalibrated photometric stereo. *IEEE Transactions on Pattern Analysis and Machine Intelligence* Vol. 41, No. 2, 271–284, 2019.

[20] Herbort, S.; Wöhler, C. An introduction to image-based 3D surface reconstruction and a survey of photometric stereo methods. *3D Research* Vol. 2, No. 3, 4, 2011.

[21] Alldrin, N.; Zickler, T.; Kriegman, D. Photometric stereo with non-parametric and spatially-varying reflectance. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 1–8, 2008.

[22] Shi, B. X.; Tan, P.; Matsushita, Y.; Ikeuchi, K. Bi-polynomial modeling of low-frequency reflectances. *IEEE Transactions on Pattern Analysis and Machine Intelligence* Vol. 36, No. 6, 1078–1091, 2014.

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

[24] Chung, H.-S.; Jia, J. Y. Efficient photometric stereo on glossy surfaces with wide specular lobes. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 1–8, 2008.

[25] Yeung, S. K.; Wu, T. P.; Tang, C. K.; Chan, T. F.; Osher, S. J. Normal estimation of a transparent object using a video. *IEEE Transactions on Pattern Analysis and Machine Intelligence* Vol. 37, No. 4, 890–897, 2015.

[26] Chen, T. B.; Goesele, M.; Seidel, H. P. Mesostructure from specularity. In: Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 1825–1832, 2006.

[27] Tozza, S.; Mecca, R.; Duocastella, M.; Del Bue, A. Direct differential photometric stereo shape recovery of diffuse and specular surfaces. *Journal of Mathematical Imaging and Vision* Vol. 56, No. 1, 57–76, 2016.

[28] Georghiades, A. S. Incorporating the Torrance and sparrow model of reflectance in uncalibrated photometric stereo. In: Proceedings of the 9th IEEE International Conference on Computer Vision, 816–823, 2003.

[29] Verbiest, F.; van Gool, L. Photometric stereo with coherent outlier handling and confidence estimation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 1–8, 2008.

[30] Sunkavalli, K.; Zickler, T.; Pfister, H. Visibility subspaces: Uncalibrated photometric stereo with shadows. In: *Computer Vision – ECCV 2010. Lecture Notes in Computer Science, Vol. 6312*. Daniilidis, K.; Maragos, P.; Paragios, N. Eds. Springer Berlin Heidelberg, 251–264, 2010.

[31] Yu, C.; Seo, Y.; Lee, S. W. Photometric stereo from maximum feasible Lambertian reflections. In: *Computer Vision – ECCV 2010. Lecture Notes in Computer Science, Vol. 6314*. Daniilidis, K.; Maragos, P.; Paragios, N. Eds. Springer Berlin Heidelberg, 115–126, 2010.

[32] Ju, Y. K.; Dong, X. H.; Wang, Y. Y.; Qi, L.; Dong, J. Y. A dual-cue network for multispectral photometric stereo. *Pattern Recognition* Vol. 100, 107162, 2020.

[33] Wang, C.; Wu, Y. T.; Su, Z. X.; Chen, J. Y. Joint self-attention and scale-aggregation for self-calibrated deraining network. In: Proceedings of the 28th ACM International Conference on Multimedia, 2517–2525, 2020.

[34] Ikehata, S. CNN-PS: CNN-based photometric stereo for general non-convex surfaces. In: *Computer Vision – ECCV 2018. Lecture Notes in Computer Science, Vol. 11219*. Ferrari, V.; Hebert, M.; Sminchisescu, C.; Weiss, Y. Eds. Springer Cham, 3–19, 2018.

[35] Li, J. X.; Robles-Kelly, A.; You, S. D.; Matsushita, Y. Learning to minify photometric stereo. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 7560–7568, 2019.

[36] Zheng, Q.; Jia, Y.; Shi, B.; Jiang, X.; Duan, L.-Y.; Kot, A. C. SPLINE-Net: Sparse photometric stereo through lighting interpolation and normal estimation networks. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, 8549–8558, 2019.

[37] Ju, Y. K.; Lam, K. M.; Chen, Y.; Qi, L.; Dong, J. Y. Pay attention to Devils: A photometric stereo network for better details. In: Proceedings of the 29th International Joint Conference on Artificial Intelligence, 694–700, 2020.

[38] Mildenhall, B.; Srinivasan, P. P.; Tancik, M.; Barron, J. T.; Ramamoorthi, R.; Ng, R. NeRF: Representing scenes as neural radiance fields for view synthesis. In: *Computer Vision – ECCV 2020. Lecture Notes in Computer Science, Vol. 12346*. Vedaldi, A.; Bischof, H.; Brox, T.; Frahm, J. M. Eds. Springer Cham, 405–421, 2020.

[39] Liu, L.; Liu, J. Z.; Yuan, S. X.; Slabaugh, G., Leonardis, A., Zhou, W. G.; Tian , Q. Wavelet-based dual-branch network for image demoiréing. In: *Computer Vision – ECCV 2020. Lecture Notes in Computer Science, Vol. 12358*. Vedaldi, A.; Bischof, H.; Brox, T.; Frahm, J. M. Eds. Springer Cham, 86–102, 2020.

[40] Paszke, A.; Gross, S.; Massa, F.; Lerer, A.; Bradbury, J.; Chanan, G.; Killeen, T.; Lin, Z.; Gimelshein, N.; Antiga, L. et al. PyTorch: An imperative style, high-performance deep learning library. In: Proceedings of the Advances in Neural Information Processing Systems, 8026–8037, 2019.

[41] Kingma, D. P.; Ba, J. Adam: A method for stochastic optimization. In: Proceedings of the 3rd International Conference on Learning Representations, 2015.

[42] Johnson, M. K.; Adelson, E. H. Shape estimation in natural illumination. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2553–2560, 2011.

[43] Wiles, O.; Zisserman, A. SilNet: Single- and multi-view reconstruction by learning from silhouettes. In: Proceedings of the British Machine Vision Conference, 2017.

[44] Matusik, W.; Pfister, H.; Brand, M.; McMillan, L. A data-driven reflectance model. *ACM Transactions on Graphics* Vol. 22, No. 3, 759–769, 2003.

[45] Jakob, W. Mitsuba renderer. 2010. Available at https://www.mitsuba-renderer.org/.

[46] Einarsson, P.; Chabert, C.-F.; Jones, A.; Ma, W.-C.; Lamond, B.; Hawkins, T.; Bolas, M.; Sylwan, S.; Debevec, P. Relighting human locomotion with owed reflectance fields. In: Proceedings of the 17th Eurographics Conference on Rendering Techniques, 183–194, 2006.

[47] Shi, B. X.; Tan, P.; Matsushita, Y.; Ikeuchi, K. Bi-polynomial modeling of low-frequency reflectances. *IEEE Transactions on Pattern Analysis and Machine Intelligence* Vol. 36, No. 6, 1078–1091, 2014.

[48] Alldrin, N. G.; Mallick, S. P.; Kriegman, D. J. Resolving the generalized bas-relief ambiguity by entropy minimization. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 1–7, 2007.

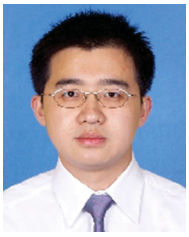
[49] Shi, B. X.; Matsushita, Y.; Wei, Y. C.; Xu, C.; Tan, P. Self-calibrating photometric stereo. In: Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 1118–1125, 2010.

[50] Wu, Z.; Tan, P. Calibrating photometric stereo by holistic reflectance symmetry analysis. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 1498–1505, 2013.

[51] Papadhimitri, T.; Favaro, P. A closed-form, consistent and robust solution to uncalibrated photometric stereo via local diffuse reflectance maxima. *International Journal of Computer Vision* Vol. 107, No. 2, 139–154, 2014.

[52] Simchony, T.; Chellappa, R.; Shao, M. Direct analytical methods for solving Poisson equations in computer vision problems. *IEEE Transactions on Pattern Analysis and Machine Intelligence* Vol. 12, No. 5, 435–446, 1990.

Yakun Ju received his B.Sc degree from Sichuan University, Chengdu, China, in 2016. He is currently pursuing his Ph.D. degree in computer application technology with the Department of Computer Science and Technology, Ocean University of China, Qingdao, China, supervised by Prof. Junyu Dong. His research interests include 3D reconstruction, deep learning, and image processing.



Yuxin Peng received his Ph.D. degree in computer application technology from Peking University in 2003. He is currently the Boya Distinguished Professor with the Wangxuan Institute of Computer Technology, Peking University. He has authored more than 160 articles in refereed international journals and conference proceedings. He has submitted 42 patent applications and been granted 24 of them. His current research interests include cross-media analysis and reasoning, image and video recognition and understanding, and computer vision.

Muwei Jian received his Ph.D. degree from the Department of Electronic and Information Engineering, Hong Kong Polytechnic University in 2014. He was a lecturer with the Department of Computer Science and Technology, Ocean University of China, from 2015 to 2017. He is currently a professor and Ph.D. supervisor with the School of Computer Science and Technology, Shandong University of Finance and Economics, Jinan, China. His current research interests include human face recognition, image and video processing, machine learning, and computer vision.

Feng Gao received his B.Sc. degree from the Department of Computer Science, Chongqing University, China, in 2008, and received his Ph.D. degree from the Department of Computer Science and Engineering, Beihang University, Beijing, China, in 2015. He is currently an associate professor in the Department of Computer Science and Technology, Ocean University of China. His research interests include computer vision and remote sensing.

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