Improved Photometric Stereo through Efficient and Differentiable Shadow Estimation

Po-Hung Yeh[1, 2]
pohungyeh@citi.sinica.edu.tw
Pei-Yuan Wu[1]
peiyuanwu@ntu.edu.tw
Jun-Cheng Chen[2]
pullpull@citi.sinica.edu.tw

[1] Department of Electrical Engineering, National Taiwan University
[2] Research Center for Information Technology Innovation, Academia Sinica

Abstract

Traditional photometric stereo approaches, although valuable in various applications, have faced limitations due to lack of considering accurate shadow estimation under different object geometry and varying lighting conditions. To address this issue, we propose a fast and accurate shadow estimation algorithm based on a dynamic programming-based sampling method with a differentiable temperature function. The proposed method can be easily used to improve existing photometric stereo methods for better estimation of shadow estimation results. In addition, we further improve the performance with our proposed higher-order derivation loss configuration. To assess the effectiveness of our method, we conduct comprehensive experiments and compare our results with diverse unsupervised and supervised approaches. The results demonstrate that our method consistently outperforms other state-of-the-art unsupervised methods in terms of mean angular error (MAE) while remaining competitive with supervised techniques.

1 Introduction

Photometric stereo requires reconstructing geometrical information from images under varying lighting conditions. This technique has proven valuable in cases where traditional methods struggle, such as when objects are partially occluded or have complex surfaces. Photometric stereo has applications in various fields, including object recognition, scene reconstruction, medical imaging, face reconstruction, and surface inspection.

The primary goal of photometric stereo is to extract geometrical information from images. While surface normal estimation maps are a standard output, depth information is often of higher interest. In some cases, multi-view data is used for triangulation or representing scenes as Neural Radiance Fields (NeRF) [13] applications. However, uncalibrated photometric stereo cannot determine normal and depth from a nearly Lambertian surface without considering additional factors such as specular effects and shadows [7]. This highlights the need for a comprehensive approach when reconstructing geometrical information from photometric stereo.

Traditional photometric stereo studies have attempted to construct depth maps by integrating or solving difference equations based on surface normal inference [28]. However, this approach has faced limitations due to the predicted normal vector field not being a conservation vector field. Previous work has introduced a novel formulation that integrated depth, normal, and re-rendered images using the rendering equation to address this challenge. Despite achieving state-of-the-art results, this approach has limitations, including a tradeoff between accuracy vs sampling number and the lack of end-to-end differentiability.

In this paper, to address the tradeoff issue between the speed and mIOU accuracy, we propose a Dynamic Programming Shadow Estimation (DPSE), which optimizes the photometric stereo shadow estimation by using a line sweeping algorithm and parallel prefix [7]. Moreover, we further develop a differentiable version of DPSE (DDPSE), resulting in a more accurate solution for the surface normal estimation by approximating the step function with an asymptotic exponential function with a temperature hyperparameter in the denominator. Our framework improves upon existing formulations by explicitly predicting multiple aspects of the output, such as lighting direction and intensity, surface normal estimation, shading and attached/cast shadow of the image, a depth map of the image, Bidirectional Reflectance Distribution Function (BRDF) material at each pixel and the re-rendered image. The proposed approaches achieve promising improvement in terms of speed and shadow estimation accuracy for the cast shadow rendering during the inference time.

Our main contributions are summarized as follows:

1. We introduce a novel parallel prefix-based shadow estimation algorithm, resulting in faster and more accurate shadow estimation than prior methods.

2. We further develop the formulation of the algorithm using a temperature function, which is proven to improve the accuracy of both normal and depth prediction. The proposed approach achieves state-of-the-art performance in photometric stereo.

3. We demonstrate the potential of shadow estimation in enhancing depth estimation through direct evidence in our experiments, introducing an aspect rarely addressed in previous photometric stereo studies.

Our framework offers a multi-purpose solution for photometric stereo that balances speed and shadow/normal estimation accuracy, making it suitable for real-time applications with geometrical output. By addressing current limitations in the field, our method represents a significant advancement in photometric stereo research and has the potential to expand its applications in various domains further.

2 Related Work

In this section, we briefly review the recent relevant works related to photometric stereos, including supervised deep photometric stereo, neural inverse rendering-based photometric stereo, and NeRF-based photometric stereo.

Supervised Deep Photometric Stereo: Woodham [26] was the first to introduce the photometric stereo problem. This work assumes a linear relationship between the cosine similarity of the surface normal and the direction of the light source, which is referred to as a Lambertian surface in subsequent work. However, this unrealistic assumption poses challenges

for real-world data, as it does not account for non-Lambertian surfaces or complex lighting effects. This leads to further investigations where the non-Lambertian BRDF(Bidirectional Reflectance Distribution Function) [9, 27] is considered. Santo *et al*. [20] pioneered the application of deep neural networks to the realm of photometric stereo, but their method was limited to a fixed number of observations under various lighting conditions, thereby losing the flexibility adopting to applications where the number of lighting directions varies. Ikehata *et al*. [8] overcame this limitation by introducing the innovative observation map encoding information into an CNN-readable image, which enabled normal map prediction without relying on a fixed number of observations. Chen *et al*. then took the torch, developing a suite of CNN-based techniques [4, 5, 6] that catered to both calibrated and uncalibrated photometric stereo. They introduced lighting space discretization for calibration tasks in uncalibrated photometric stereo and effectively leveraged the power of max-pooling to aggregate features from multiple observed images and directions. However, these methods still relied on the availability of ground truth for normal and lighting calibration, which is not always feasible for real-world data.

Neural Inverse Rendering in Photometric Stereo: To tackle missing ground truth in datasets, self-supervised techniques through inverse rendering have gained prominence in photometric stereo. Taniai and Maehara [22] initially introduced reprojection-based guidance. They were followed by Kaya *et al*. [10], who considered inter-reflections. Tiwari and Raman then developed LERPS [24] using GAN loss and DeepPS2 [23] for self-calibration with two images.

NeRF Based Methods on Photometric Stereo: NeRF [18] represents a cutting-edge deep learning technique for capturing a scene as a dataset. Unlike traditional deep learning frameworks that strive to generalize, a NeRF model's objective is to encode by fitting a particular scene for improved rendering performance. NeRF-based approaches have produced state-of-the-art results in normal estimation, eliminating the requirement for ground truth during training in the context of photometric stereo. In a recent development, Li [12, 13] adapted the NeRF concept, training a MLP network on a scene under diverse lighting conditions. Building on this foundation, subsequent research by Yang *et al*. [29] merged the advantages of NeRF with those of multi-view stereo. While optimization using NeRF-based methods typically takes just a few minutes on GeForce RTX 30 Series, these approaches have significantly impacted the photometric stereo field, opening new possibilities for future research and applications.

3 Method

In this section, we introduce the details of the proposed DPSE and DDPSE algorithms.

3.1 Problem Formulation

We model the image rendering process using the approaches presented in previous works [3, 4, 5, 6, 8, 9, 10, 12, 14, 22]. Here we assume the light intensity is a constant by normalizing the observed images. Consider a set of photometric stereo images $X^{(0)}, X^{(1)}, \cdots, X^{(I-1)}$, while each image $X^{(i)} \in \mathbb{R}^{C\times H\times W}$ has C channels and dimensions H, W. Each image is associated with a point light source placed an infinite distance away, having direction $l^{(i)} \in$

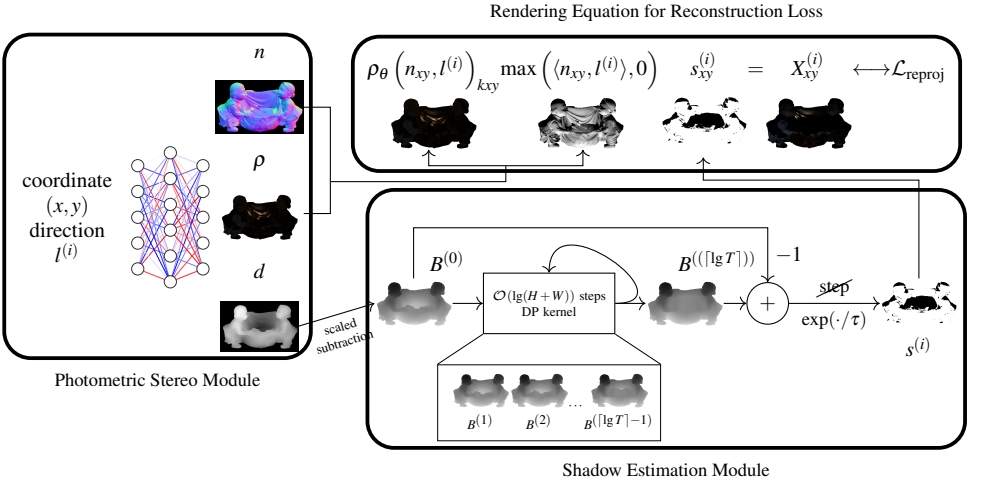


Figure 1: This diagram provides an overview of our contributions. Our system begins by processing input data through a Photometric Stereo module, which outputs normal, BRDF, and depth mappings. These depth mappings are utilized to compute shadows using our method. By inferring this information, we can render a predicted image for the process of inverse rendering.

$\mathbb{S}^2$. A dense normal mapping is denoted as $n_{xy} \in \mathbb{S}^2$, where xy is the pixel index. We assume n_{xy} and $l^{(i)}$ are of unit norm.

We account for the shadowing effect caused by geometric constraints, denoted by $s_{xy}^{(i)} \in \{0,1\}$ [10, 12, 14]. A value of 0 indicates that the location (x,y) is occluded by shadow. The effect of the BRDF (Bidirectional Reflectance Distribution Function, parameterized by θ) ρ_θ, which depends on the pixel's material, is also considered. The observed image is modeled as Figure 1.

3.1.1 Photometric Stereo Module

Photometric stereo networks, as highlighted in several studies [10, 12, 13, 29], leverage both global and local features for reverse rendering. With these works, we start with a model designed to predict the normal map $n_{xy} = N(x,y)$ and the depth map $d_{xy} = f(x,y)$, based on the stereo coordinate (x,y). Additionally, for each light direction $l^{(i)}$, the model is assumed to predict the BRDF as $\rho_{xy}^{(i)} = \rho_\theta\left(n_{xy}, l^{(i)}\right)$.

Our methodology distinguishes itself, especially from Li's [12] approach, by the synergistic application of these features. Rather than manually setting a threshold for shadow labeling, we incorporate these features in an end-to-end framework, enabling the model to autonomously determine shadow pixel labeling.

3.1.2 Shadow Estimation Module

With a lighting direction $l = (l_x, l_y, l_z)$, the shadow term $s_{xy}^{(i)} = s(x,y)$ is the occlusion along the light direction:

$$s(x,y) = \text{step}\left(\min_{t\in\mathbb{R}_+} f(x,y) + l_z t - f(x + l_x t, y + l_y t)\right) \tag{1}$$

where $f(x,y)$ is the depth map and step$(\cdot)$ represents the Heaviside step function. The shadow term $s(x,y)$ is determined by identifying any occlusion along a trajectory that starts from $f(x,y)$ and follows the lighting direction l.

3.1.3 Optimization Loss

Geometric Constraint and Autograd Losses. Given the predicted normal map n and predicted depth map $d = f(x,y)$, we define a loss $\mathcal{L}_{\text{geo}}$ to align these two outputs. To convert the normal map from a depth map, we use finite difference with step Δ to obtain the estimated normal vector. We also employ higher-order differentiation [1] loss $\mathcal{L}_{\text{autograd}}$ on the network's output f to take advantage of automatic differentiation frameworks available in popular deep learning packages.

$$\mathcal{L}_{\text{geo}} = 1 - \hat{n}^\top \left(\frac{f(x+\Delta,y) - f(x,y)}{\Delta}, \frac{f(x,y+\Delta) - f(x,y)}{\Delta}\right), \mathcal{L}_{\text{autograd}} = 1 - \hat{n}^\top \nabla f \tag{2}$$

Reprojection Loss. In the case of inverse rendering, given light direction l, we employ Li's [12] models to calculate $\hat{n}, \hat{\rho}, \hat{d}$. We then evaluate $\hat{s}$ using our shadow estimation algorithm with L1 loss.

$$\mathcal{L}_{\text{reproj}} = \left| X - \hat{s}\hat{\rho} \max\left(\hat{n}^\top l, 0\right)\right| \tag{3}$$

Here, X represents the ground truth image, $\hat{\rho}$ refers to the BRDF, and $\hat{s}$ is the shadow map estimated by our dynamic programming-based method. The overall loss is the weighted sum of all the individual losses:

$$\mathcal{L} = \lambda_{\text{reproj}}\mathcal{L}_{\text{reproj}} + \lambda_{\text{geo}}\mathcal{L}_{\text{geo}} + \lambda_{\text{autograd}}\mathcal{L}_{\text{autograd}}, \tag{4}$$

where $\lambda_{\text{reproj}}, \lambda_{\text{geo}}, \lambda_{\text{autograd}}$ are the weighting coefficients that control the relative importance of each loss term.

3.2 Closed-form Differentiable Shadow Estimation

The shadow estimation Eq. 1 can be accelerated using dynamic programming techniques, such as a line sweeping algorithm [11] on a discretized sample over the real line t. A naive approach in one dimension would take linear time based on the sample space size.

As our contribution, we present a technique to embed a parallelized line sweeping algorithm into a convolutional neural network. This shadow estimation module estimates the cast shadow from the depth map and lighting direction. Furthermore, this module component is independent of photometric stereo and is differentiable with only one parameter, making optimizing within a deep neural network easier.

3.2.1 Shadow Estimation from Depth Map

The shadow casting as shown in Eq. 1 of a given pixel $v(0) = (x,y)$ and lighting direction $l = (l_x, l_y, l_z)$ can be modeled as a one-dimensional sweep line problem along a discretized trajectory. The surface depth is represented by $z(t) = f(v(t))$, where $v(t) = (x + tl_x, y + tl_y)$ and x, y are pixel indices. The shadow estimate is then given by:

$$s = \text{step}\left(\min_{t \geq 0} z(0) + tl_z - z(t)\right) \tag{5}$$

We discretize the trajectory with a step size w and define the array $F[i] = f(v(w \cdot i))$ for $i = 0, 1, 2, \cdots$. The shadow estimation can then be written as:

$$s' := \text{step}\left(F[0] + \min_{i \in \mathbb{N}_+ \cup \{0\}} (il_z w - F[i])\right) = \text{step}\left(-B^{(0)}[0] + \min_{i \in \mathbb{N}_+ \cup \{0\}} B^{(0)}[i]\right) \tag{6}$$

We use dynamic programming to compute Eq. 6 efficiently. By defining $B^{(0)}[i] := il_z w - F[i]$, we can compute $B^{(l)}[i]$ through parallel prefix [7]:

$$B^{(l+1)}[i] := \min\left(B^{(l)}[i], B^{(l)}[i + 2^l]\right) \tag{7}$$

$$= \min\left(B^{(l-1)}[i], B^{(l-1)}[i + 2^{l-1}], B^{(l-1)}[i + 2 \cdot 2^{l-1}], B^{(l-1)}[i + 3 \cdot 2^{l-1}]\right) \tag{8}$$

$$\vdots \tag{9}$$

$$= \min_{0 \leq t < 2^l} B^{(0)}[i + t] \tag{10}$$

Note that the computation depicted in Eq. 7 corresponds to an image operation, which is inherently optimized by the majority of hardware and software systems, thus ensuring rapid execution on GPUs.

We use an exponential function with temperature τ to approximate the non-differentiable step function in our implementation. Since s' from Eq. 6 is always non-positive, the exponential function $\exp(\cdot/\tau)$ will approximately converge to the step function when τ is small. The temperature τ can be a trainable or a fixed parameter. We set τ as a trainable parameter (refer to Table 4). A smaller value of τ is indicative of a model that is more confident in its predictions, although this might introduce potential biases. With the replacement of temperature function, Eq. 6 can then be rewritten as:

$$s'' = \exp\left(\frac{B^{(\lceil \lg T \rceil)}[0] - B^{(0)}[0]}{\tau}\right), \tag{11}$$

where $T = \mathcal{O}(H + W)$ is the range of image size. The following lemma analysis the time complexity of the proposed shadow estimation module.

Lemma 1 (Module Complexity for Shadow Estimation)**.** *The shadow estimate can be computed with at most $\mathcal{O}(\log \frac{T}{w})$ differentiable image shift operations, where T is the range of the image dimension and w is the sampling interval.*

On top of that, Eq. 11 also inherits a numerical accuracy limitation:

Table 1: We compare our method with other state-of-the-art method in the MAE metric. The bold font means the best normal estimation accuracy.

Methods	GT Normal	Ball	Bear	Buddha	Cat	Cow	Goblet	Harvest	Pot1	Pot2	Reading	Avg
PX-Net [ ]	YES	2.00	**3.50**	7.60	**4.30**	**4.70**	**6.70**	**13.30**	**4.90**	**5.00**	**9.80**	**6.17**
WJ20 [ ]	YES	**1.78**	4.12	**6.09**	4.66	6.33	7.22	13.34	6.46	6.45	10.05	6.65
CNN-PS [ ]	YES	2.20	4.10	7.90	4.60	8.00	7.30	14.00	5.40	6.00	12.60	7.20
GPS-Net [ ]	YES	2.92	5.07	7.77	5.42	6.14	9.00	15.14	6.04	7.01	13.58	7.81
PS-FCN [ ]	YES	2.82	7.55	7.91	6.16	7.33	8.60	15.85	7.13	7.25	13.33	8.39
TM18 [ ]	NO	**1.47**	5.79	10.36	5.44	6.32	11.47	22.59	6.09	7.76	11.03	8.83
BK21 [ ]	NO	3.78	5.96	13.14	7.91	10.85	11.94	25.49	8.75	10.17	18.22	11.62
L2 [ ]	NO	4.10	8.40	14.90	8.40	25.60	18.50	30.60	8.90	14.70	19.80	15.40
Li22 [ ]	NO	2.43	3.64	8.04	4.86	4.72	6.68	14.90	**5.99**	**4.97**	8.75	6.50
DDPSE (ours)	NO	2.00	**3.61**	**7.48**	**4.75**	**4.64**	**6.53**	**13.79**	6.11	5.51	**8.14**	**6.26**

Lemma 2 (Numerical Error for Shadow Estimation)**.** *The numerical error of the shadow estimation module, which differs from Equation 1, is bounded by:*

$$\left|s'' - s\right| \leq \underbrace{\frac{w}{\tau} \sup_{t \in \mathbb{R}_+} \left|l_z - z'(t)\right|}_{\text{dominant when } \tau \text{ is small}} + \underbrace{\exp\left(\frac{\min_{t \in \mathbb{R}_+} z(0) + t l_z - z(t)}{\tau}\right)}_{\text{dominant when } \tau \text{ is large}} \tag{12}$$

The first term is related to the smoothness of the surface, while the second term is related to the approximation of the step function using the exponential function.

4 Experiments

In this section, we present the evaluation results of the proposed method with other state-of-the-art methods. We first describe the implementation and training details as follows.

Implementation Details: We adhere to the frameworks proposed by Li [12, 13] when implementing our core algorithm. Moreover, we synthesize a Lambertian-based dataset using common surface functions and include ground truth depth and cast shadow information. We utilize this dataset in specific experiments to evaluate depth estimation accuracy. We conduct experiments to measure the speedup achieved using dynamic programming. For detailed information, please refer to our supplementary material.

Training Details: To train our model, we begin with Li's pre-trained model [12] and integrate our shadow estimation algorithm for inverse rendering, making the process end-to-end. As shown in Table 1, our proposed model is implemented with τ as a trainable parameter. In addition, the differentiable image shifting operation is implemented using kornia [19]. We train the model in 800 epochs, with two iterations per epoch and batch size as half of the total data samples per iteration. Each instance takes approximately 5-8 minutes on an Nvidia 2080 Ti GPU. The initial learning rate is set as 3.5×10^{-4}. We employ the cosine annealing learning rate schedule with the AdamW optimizer [17] for optimization. The experiments are conducted three times, and we compute the standard deviation of the average MAE on the results that we achieve to be less than 0.01 (i.e., which is statistically significant). We carefully configure the hyperparameters of our model. We also set the weight for autograd loss, $\lambda_{\text{autograd}} = 2.0 \times 10^{-3}$ and the weight for geometrical loss, $\lambda_{\text{geo}} = 1.0 \times 10^{-3}$, the weight for projection loss $\lambda_{\text{reproj}} = 1$. These hyperparameters are chosen to ensure the optimal model performance.

Table 2: Qualitative comparison results on normal estimation.

GT Normal	DDPSE	Li [12]	PS-FCN [5]
			
MAE	**8.14°**	8.75°	13.33°
			
MAE	**13.79°**	14.90°	15.85°
			
MAE	**0.36°**	7.18°	0.47°

4.1 Enhancing Self-Supervised Methods

In this study, we aim to enhance the accuracy of estimating surface normal without any supervised guidance. To achieve this, we compare our proposed improvements with Li's method [12] from the source model and other recent approaches [5] using the widely-accepted mean angular error (MAE) metric.

Table 1 compares MAE values for various supervised and unsupervised methods on the common DiLiGenT [21] photometric stereo benchmark, including our proposed improvements. Our findings demonstrate that our improvements consistently outperform other unsupervised methods while remaining competitive with supervised methods. We provide qualitative results in Table 2, highlighting the key differences in the qualitative results for each instance. The bounding box in the table indicates the areas where our method outperforms others, particularly in regions with cast shadows. This demonstrates the effectiveness of our proposed DDPSE in handling cast shadows, leading to improved normal estimation results. By incorporating our novel shadow estimation algorithm into the photometric stereo framework, we can perform better in challenging scenarios, making our method a more robust and reliable solution for various applications.

4.2 Evaluation Results with Ground-Truth Depth

We measure the predicted depth estimation using simply rendered surfaces with ground truth depth and cast shadow. The shapes used include Cube (step-like function), Cone, Hemisphere, Gaussian, Ripple (sinusoidal function in polar coordinates), and Bumps (2D trigonometric functions), as shown in Table 3. For depth estimation, we measure the accuracy using root-mean-square error (RMSE) and peak signal-to-noise ratio (PSNR) metrics. We also provide the MAE result for normal estimation of the instances. Moreover, we achieved bet-

Table 3: Comparisons of Li [17] and the proposed DDPSE on the photometric stereo datasets. The bold font indicates the better shadow estimation accuracy.

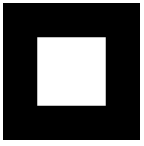
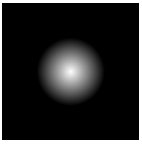
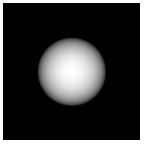
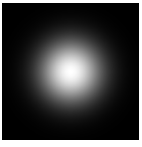
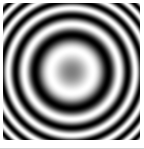
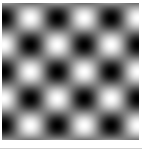
Instance	Method	Est Depth	Metric	Value	Instance	Method	Est Depth	Metric	Value
	Li		RMSE PSNR MAE	61.10 2.66 7.18		Li		RMSE PSNR MAE	0.20 49.21 0.88
	DDPSE		RMSE PSNR MAE	**8.11** **21.47** **0.36**		DDPSE		RMSE PSNR MAE	**0.09** **56.76** **0.86**
	Li		RMSE PSNR MAE	1.85 29.02 0.68		Li		RMSE PSNR MAE	2.25 33.02 5.08
	DDPSE		RMSE PSNR MAE	**0.36** **43.75** **0.53**		DDPSE		RMSE PSNR MAE	**0.12** **59.03** **4.75**
	Li		RMSE PSNR MAE	**0.68** **25.10** 6.38		Li		RMSE PSNR MAE	0.47 30.57 4.40
	DDPSE		RMSE PSNR MAE	1.79 16.64 **3.67**		DDPSE		RMSE PSNR MAE	**0.17** **39.23** **4.13**

Table 4: Ablation study of the proposed method using the MAE metric.

Setting	Methods	Ball	Bear	Buddha	Cat	Cow	Goblet	Harvest	Pot1	Pot2	Reading	Avg
S1	[17]	2.43	3.64	8.04	4.86	4.72	6.68	14.90	**5.99**	**4.97**	8.75	6.50
S2	S1 + DPSE	2.03	3.75	8.38	4.81	**4.55**	6.66	15.48	6.58	5.78	9.26	6.73
S3	S2 + step $\rightarrow \exp(\cdot/\tau)$	2.03	3.62	**7.48**	4.79	4.70	**6.52**	13.85	6.28	5.82	8.16	6.33
proposed	S3 + τ tunable	**2.00**	**3.61**	**7.48**	**4.75**	4.64	6.53	**13.79**	6.11	5.51	**8.14**	**6.26**

ter results compared to our baseline. Our method outperforms in all instances, except for the Ripple case, which indicates that our method may not be robust for certain instances. However, our baseline outputs a flat surface instead in the Cube instance (Table 3). It is important to note that the Cube instance is a pure shadow instance and does not provide any clue of normal from Woodham's [26] formulation. The fact that our method can achieve non-trivial RMSE and PSNR results for the Cube instance suggests that our method is not only robust to cast shadows but can also leverage shadow information to enhance geometric output.

4.3 Ablation Study

As described in the main paper, we have introduced several improvements to Li's method [17], which include:

- Incorporating dynamic programming for accelerated sampling.
- Employing a softer step function (the $\exp(\cdot/\tau)$) to enable differentiability.
- Making τ a trainable parameter.

Table 4 presents the impact of these modifications on the Mean Absolute Error (MAE) of normal estimation. We conduct an ablation study using Li's method [17] as a starting point. As expected, adding dynamic programming does not affect normal estimation accuracy since its primary contribution is to speed up the process. The crucial improvements come from making the operations differentiable, as evidenced by the results in Table 4. For the configuration that replaces the step function with the exponential function with a fixed

temperature, we set $\tau = 5 \times 10^3$. We then further make τ learnable and reaches the best result. Finally, we adopt this configuration for all the experiments in this paper.

5 Conclusion

In this paper, we propose a novel approach to enhance self-supervised methods for normal, depth, and cast shadow estimation. Our method has been shown to be effective in improving the performance of the previous state-of-the-art Li's framework [12], incorporating a dynamic programming-based sampling method and a differentiable temperature function, which jointly improves computational efficiency and shadow estimation accuracy. Through extensive experiments, we demonstrate that our proposed method outperforms other unsupervised techniques in terms of mean angular error (MAE), while remaining competitive with supervised methods. One of the significant contributions of our method is the ability to estimate depth using only shadow information (Table 3).

6 Acknowledgement

This research is supported by Taiwan Semiconductor Manufacturing Company under the grant number of TUP-20211108-2570, National Science and Technology Council, Taiwan (R.O.C), under the grant number of NSTC-111-2634-F-002-022, and Academia Sinica under the grant number of AS-CDA-112-M09. In addition, we would like to express our gratitude for the valuable contributions and guidance from these organizations, which have been instrumental in achieving the goals of this research.

References

[1] Martin Arjovsky, Soumith Chintala, and Léon Bottou. Wasserstein generative adversarial networks. In *Proceedings of the International Conference on Machine Learning (ICML)*, 2017.

[2] P.N. Belhumeur, D.J. Kriegman, and A.L. Yuille. The bas-relief ambiguity. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 1997.

[3] Manmohan Krishna Chandraker, Fredrik Kahl, and David J Kriegman. Reflections on the generalized bas-relief ambiguity. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2005.

[4] Guanying Chen, Kai Han, Boxin Shi, Yasuyuki Matsushita, and Kwan-Yee K. Wong. Sdps-net: Self-calibrating deep photometric stereo networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019.

[5] Guanying Chen, Kai Han, Boxin Shi, Yasuyuki Matsushita, and Kwan-Yee K. Wong. Deep photometric stereo for non-Lambertian surfaces. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 44(1):129–142, 2020.

[6] Guanying Chen, Michael Waechter, Boxin Shi, Kwan-Yee K Wong, and Yasuyuki Matsushita. What is learned in deep uncalibrated photometric stereo? In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2020.

[7] Ananth Grama, George Karypis, Vipin Kumar, and Anshul Gupta. *Introduction to Parallel Computing*. Addison-Wesley, 2003.

[8] Satoshi Ikehata. Cnn-ps: Cnn-based photometric stereo for general non-convex surfaces. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018.

[9] Satoshi Ikehata, David Wipf, Yasuyuki Matsushita, and Kiyoharu Aizawa. Robust photometric stereo using sparse regression. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2012.

[10] Berk Kaya, Suryansh Kumar, Carlos Oliveira, Vittorio Ferrari, and Luc Van Gool. Uncalibrated neural inverse rendering for photometric stereo of general surfaces. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021.

[11] Jeff Lane and Loren Carpenter. A generalized scan line algorithm for the computer display of parametrically defined surfaces. *Computer Graphics and Image Processing*, 11(3):290–297, 1979.

[12] Junxuan Li and Hongdong Li. Neural reflectance for shape recovery with shadow handling. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022.

[13] Junxuan Li and Hongdong Li. Self-calibrating photometric stereo by neural inverse rendering. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2022.

[14] Junxuan Li, Antonio Robles-Kelly, Shaodi You, and Yasuyuki Matsushita. Learning to minify photometric stereo. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019.

[15] Xin Lin, Changxing Ding, Jinquan Zeng, and Dacheng Tao. Gps-net: Graph property sensing network for scene graph generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020.

[16] Fotios Logothetis, Ignas Budvytis, Roberto Mecca, and Roberto Cipolla. Px-net: Simple and efficient pixel-wise training of photometric stereo networks. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2021.

[17] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *Proceedings of the International Conference on Learning Representations (ICLR)*, 2019.

[18] Ben Mildenhall, Pratul P Srinivasan, Matthew Tancik, Jonathan T Barron, Ravi Ramamoorthi, and Ren Ng. Nerf: Representing scenes as neural radiance fields for view synthesis. *Communications of the ACM*, 65(1):99–106, 2021.

[19] Edgar Riba, Dmytro Mishkin, Daniel Ponsa, Ethan Rublee, and Gary Bradski. Kornia: an open source differentiable computer vision library for pytorch. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, 2020.

[20] Hiroaki Santo, Masaki Samejima, Yusuke Sugano, Boxin Shi, and Yasuyuki Matsushita. Deep photometric stereo network. In *Proceedings of the IEEE/CVF International Conference on Computer Vision Workshop (ICCVW)*, 2017.

[21] Boxin Shi, Zhe Wu, Zhipeng Mo, Dinglong Duan, Sai-Kit Yeung, and Ping Tan. A benchmark dataset and evaluation for non-lambertian and uncalibrated photometric stereo. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

[22] Tatsunori Taniai and Takanori Maehara. Neural inverse rendering for general reflectance photometric stereo. In *Proceedings of the International Conference on Machine Learning (ICML)*, 2018.

[23] Ashish Tiwari and Shanmuganathan Raman. Deepps2: Revisiting photometric stereo using two differently illuminated images. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 129–145, 2022.

[24] Ashish Tiwari and Shanmuganathan Raman. Lerps: lighting estimation and relighting for photometric stereo. In *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2022.

[25] Xi Wang, Zhenxiong Jian, and Mingjun Ren. Non-lambertian photometric stereo network based on inverse reflectance model with collocated light. *IEEE Transactions on Image Processing (TIP)*, 29:6032–6042, 2020.

[26] Robert J Woodham. Photometric stereo: A reflectance map technique for determining surface orientation from image intensity. In *Proceedings of the Image Understanding Systems and Industrial Applications I*, volume 155, pages 136–143. SPIE, 1979.

[27] Lun Wu, Arvind Ganesh, Boxin Shi, Yasuyuki Matsushita, Yongtian Wang, and Yi Ma. Robust photometric stereo via low-rank matrix completion and recovery. In *Proceedings of the Asian Conference on Computer Vision (ACCV)*, 2011.

[28] Zhongquan Wu and Lingxiao Li. A line-integration based method for depth recovery from surface normals. *Computer Vision, Graphics, and Image Processing*, 43(1):53–66, 1988.

[29] Wenqi Yang, Guanying Chen, Chaofeng Chen, Zhenfang Chen, and Kwan-Yee K Wong. Ps-nerf: Neural inverse rendering for multi-view photometric stereo. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2022.