Neural Architecture Search for Efficient Uncalibrated Deep Photometric Stereo [Supplementary Material]

Francesco Sarno[1], Suryansh Kumar[1], Berk Kaya[1], Zhiwu Huang[1], Vittorio Ferrari[2], Luc Van Gool[1,3]
Computer Vision Lab, ETH Zürich[1], Google Research[2], KU Leuven[3]

Abstract

In this draft, we extended the experimental section of our main paper. It is organized as follows: First, we present the cell designs of our automatically searched neural architectures with complexity analysis. Next, we study the scope of our searched architecture and how it generalizes to diverse subjects with different surface profiles and material properties. To that end, we additionally performed experiments on Light Stage Data Gallery dataset [2] and Gourd&Apple [1]. Further, we conducted an extensive ablation study on our design choices. Specifically, we demonstrate the effect of using an auxiliary tower at train time, we study the variation in the performance with changes in the cell structure, we explore effects of modifications of the search space and we investigate different split percentages for search train set and search validation sets. Finally, we discuss the limitations and possible future extensions.

1. Analysis on Neural Architecture Search Design

1.1. Cell Designs

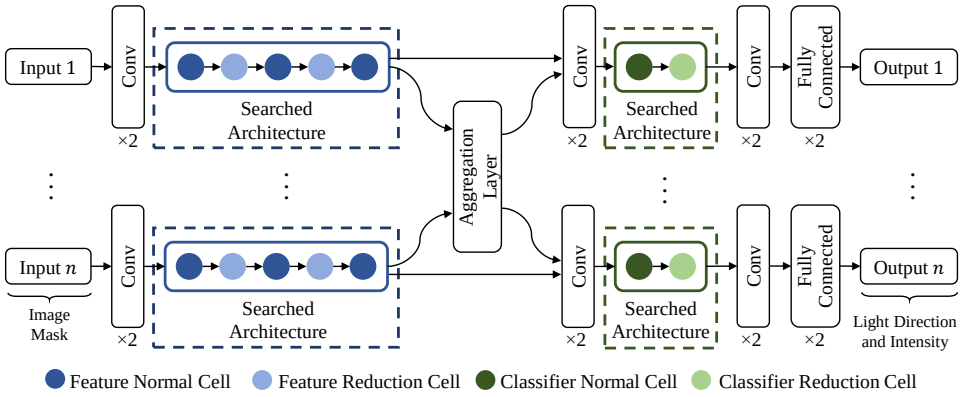
In Fig.1 and Fig.2, we show the obtained cells for light calibration network and normal estimation network respectively. The inclusion of diverse operations in the search space definition such as "zero", "$\mathrm{skip\_connnection}$", *etc*., allowed us to achieve commendable surface normal accuracy with a lighter architecture than the existing deep uncalibrated photometric stereo (PS) methods.

1.2. Complexity Analysis

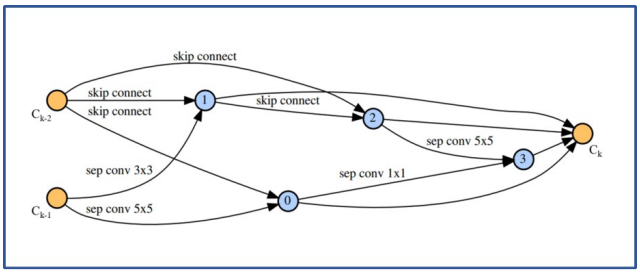
In this section, we show the effectiveness of the adopted one-shot differentiable NAS. In a reasonable amount of computational time, *i.e*. 2 GPU days on a single NVIDIA GPU with 12GB of RAM, the search phase manages to find optimal candidate architecture among a vast set of possible candidates. Here, we analyze the complexity of the search phase for both networks in detail.

Light Calibration Network: In this study, we consider cells with 4 intermediate nodes and 2 input nodes (see Fig.3). For each intermediate node, 2 strongest incoming operations are preserved in the final architecture. As we include 4 operations in the search space, excluding "zero", each normal or reduction cell is chosen among $\prod_{k=1}^{4} \frac{(k+1)k}{2} \times 4^2 \approx 10^7$ possible Directed Acyclic Graphs (DAGs). In the light calibration network, the search algorithm looks for 4 cells: normal and reduction cells for feature extractor and classifier. This leads to an total number of approximately $(10^7)^4 = 10^{28}$ possible discretized candidate architectures.

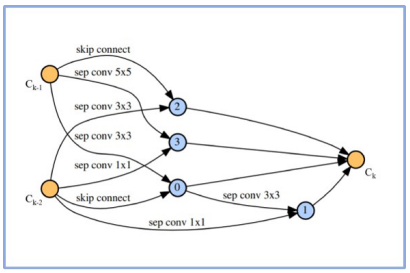
Normal Estimation Network: The same calculation can be extended to the normal estimation network either. In the search phase, the NAS algorithm looks for 3 cells: feature extractor normal and reduction cells and regressor normal cell. In the end, the complete number of architectures is about $(10^7)^3 = 10^{21}$ discretized candidates.



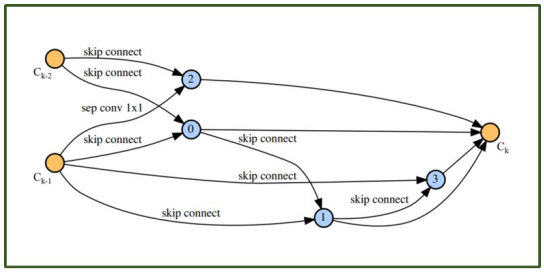
(a) Light Calibration Network



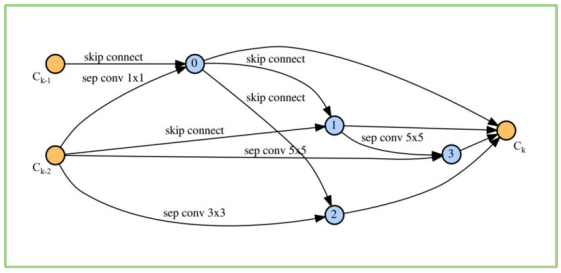
(b) Light Calibration Network Feature Extractor (Normal cell)



(c) Light Calibration Network Feature Extractor (Reduction cell)

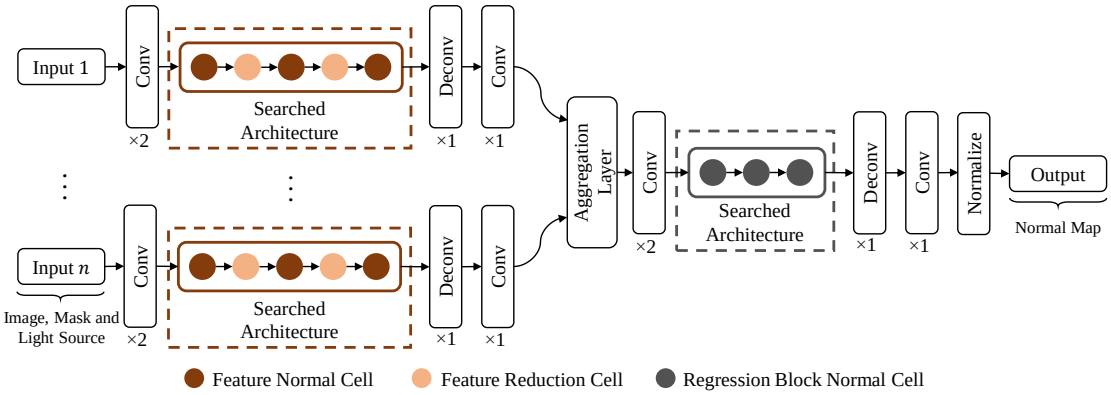


(d) Light Calibration Network Classifier (Normal cell)

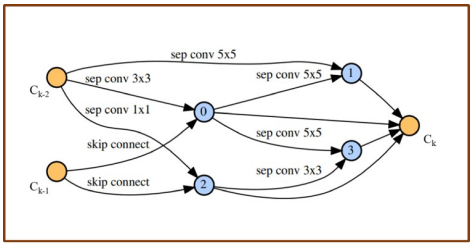


(e) Light Calibration Network Classifier (Reduction cell)

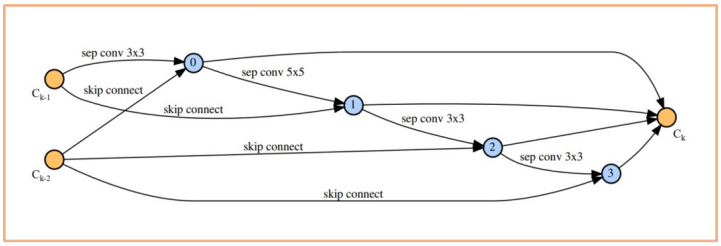
Figure 1: Illustration of the cells obtained with neural architecture search for the Light Calibration Network. (a) The backbone and the searchable parts. (b) Feature extractor normal cell. (c) Feature extractor reduction cell. (d) Classifier normal cell. (e) Classifier reduction cell.



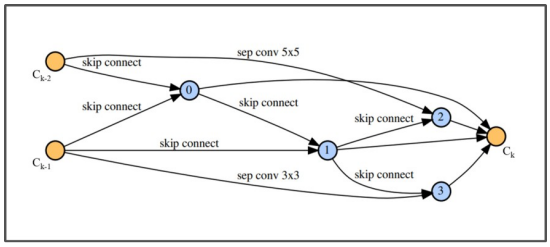
(a) Normal Estimation Network



(b) Normal Estimation Network Feature Extractor (Normal cell)



(c) Normal Estimation Network Feature Extractor (Reduction cell)



(d) Normal Estimation Network Regressor (Normal cell)

Figure 2: Illustration of the cells obtained with neural architecture search on the Normal Estimation Network. (a) The backbone and the searchable parts. (b) Feature extractor normal cell. (c) Feature extractor reduction cell. (d) Regressor normal cell.

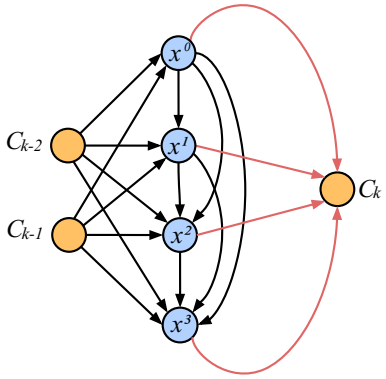


Figure 3: Illustration of a cell structure with 2 input nodes, 4 intermediate nodes and 1 output node. Black arrows are the operations to be learned by NAS algorithm. Red arrows represent the output of intermediate nodes to be concatenated to the output node.

2. More Results

2.1. Qualitative and Quantitative Performance Comparison on DiLiGenT Dataset

In this section, we compare our method against other uncalibrated photometric stereo methods on the DiLiGenT dataset [9]. The DiLiGenT dataset is composed of 10 real objects where each object is illuminated by 96 light sources. These objects are selected to cover a wide range of challenging real-world surfaces with different material properties. The dataset also provides ground-truth surface normals, light directions, and light intensities for extensive quantitative analysis.

In Fig.4-13 we visually compare our surface normal maps with Robust PS [8], Holistic PS [10], SDPS-Net [3] and UPS-FCN† [4]. Note that we use the deeper version of the UPS-FCN [4] for a fair comparison. For each method mentioned, we also provide angular error maps and MAE_{normal} values. The results show that our method outperforms the existing uncalibrated photometric stereo methods by estimating surface normals more accurately on various kinds of surfaces. For instance, objects such as BALL, POT1, BEAR, and POT2 have simple surfaces with limited specularity effects. On the other hand, BUDDHA and GOBLET have more complex and detailed surfaces. We also show results on scenes characterized by strong specular effects and interreflection phenomena such as READING and HARVEST. The results of these examples validate the efficiency of our method on complex shapes and challenging reflectance behaviors.

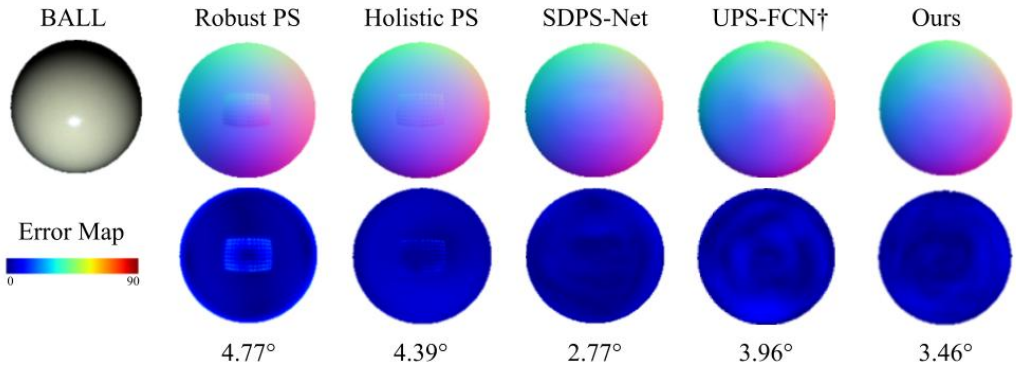


Figure 4: Visual comparison of uncalibrated photometric stereo results on **BALL** scene.

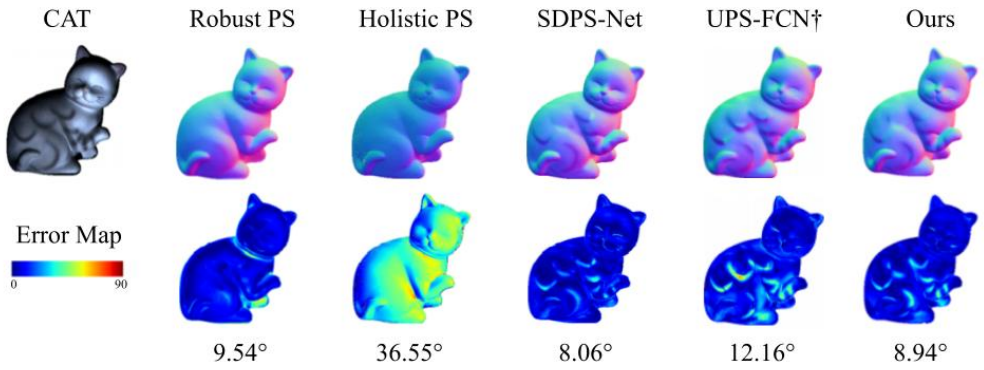


Figure 5: Visual comparison of uncalibrated photometric stereo results on **CAT** scene.

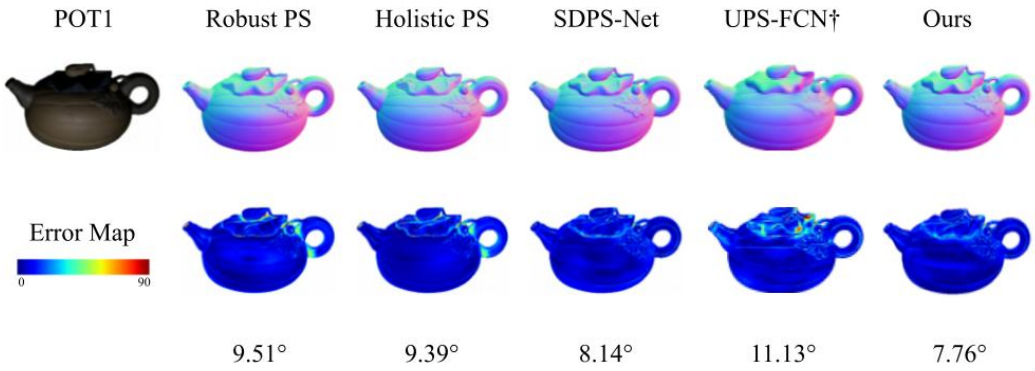


Figure 6: Visual comparison of uncalibrated photometric stereo results on **POT1** scene.

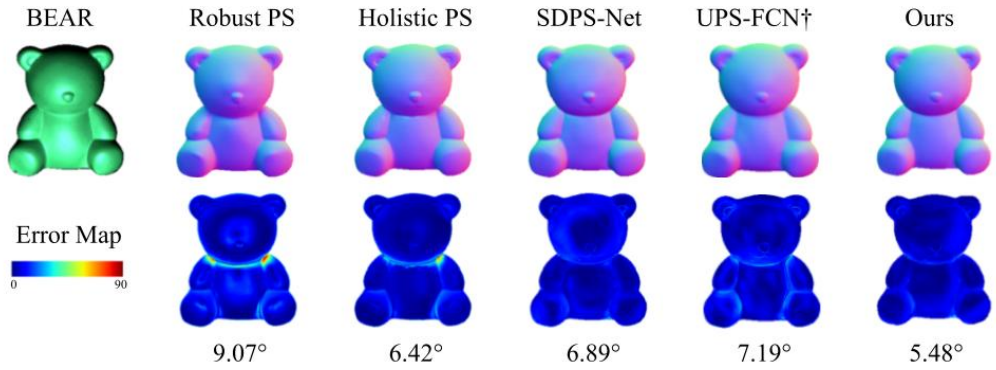


Figure 7: Visual comparison of uncalibrated photometric stereo results on **BEAR** scene.

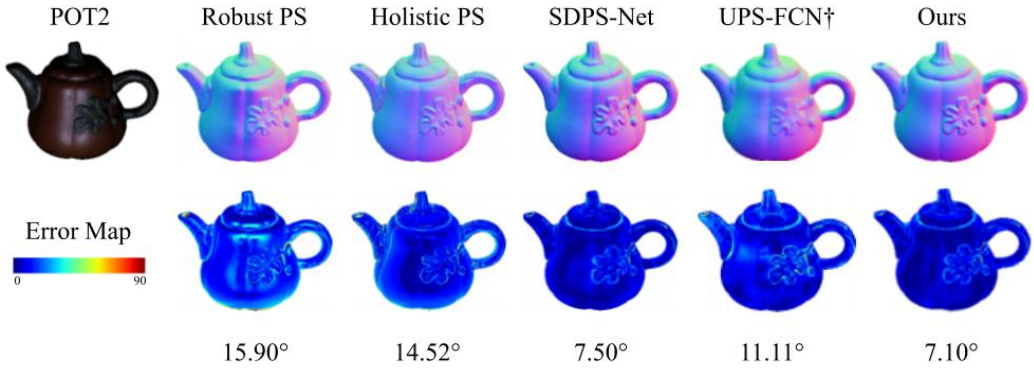


Figure 8: Visual comparison of uncalibrated photometric stereo results on **POT2** scene.

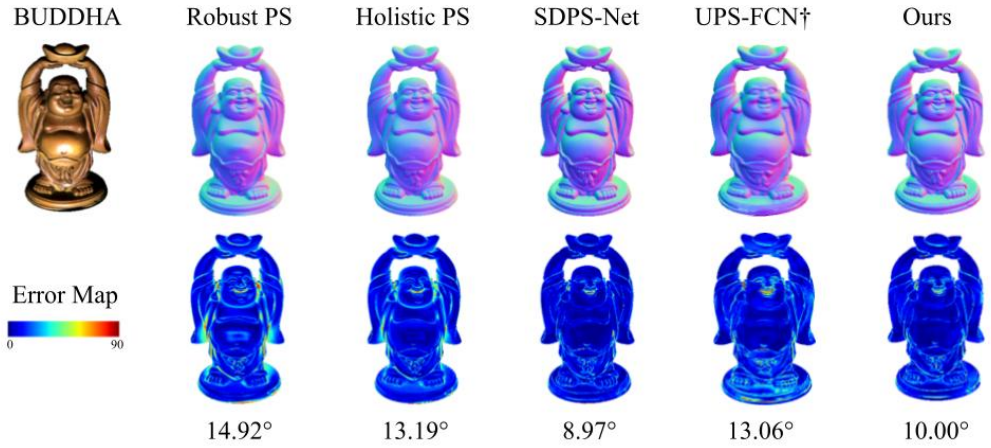


Figure 9: Visual comparison of uncalibrated photometric stereo results on **BUDDHA** scene.

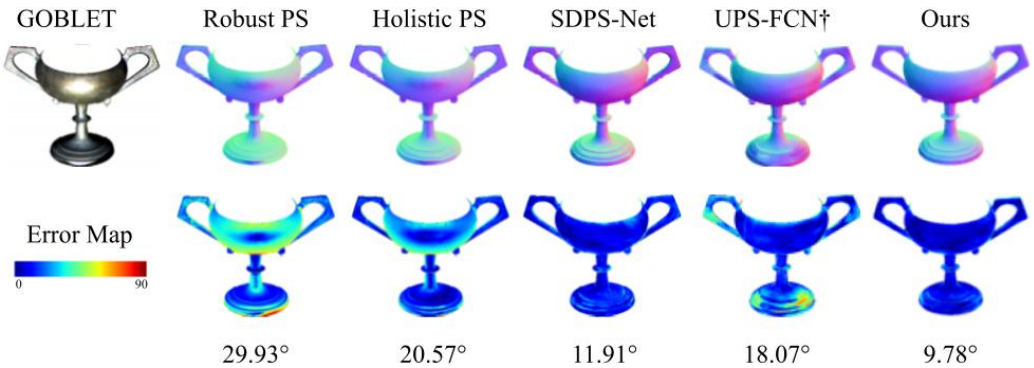


Figure 10: Visual comparison of uncalibrated photometric stereo results on **GOBLET** scene.

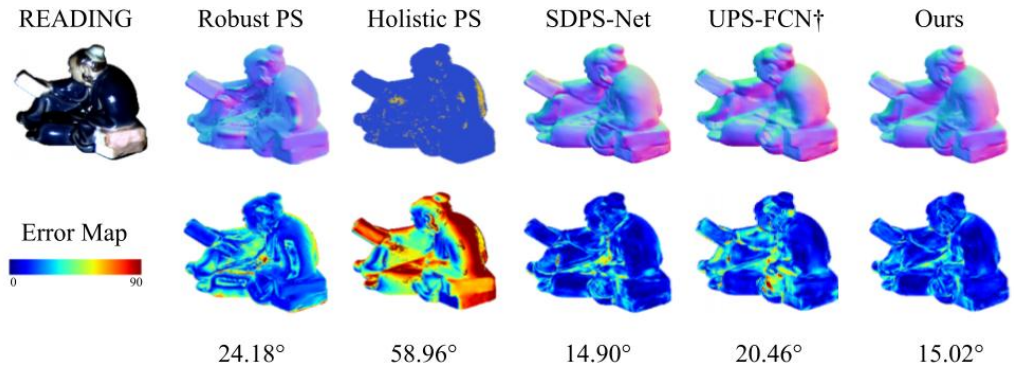


Figure 11: Visual comparison of uncalibrated photometric stereo results on **READING** scene.

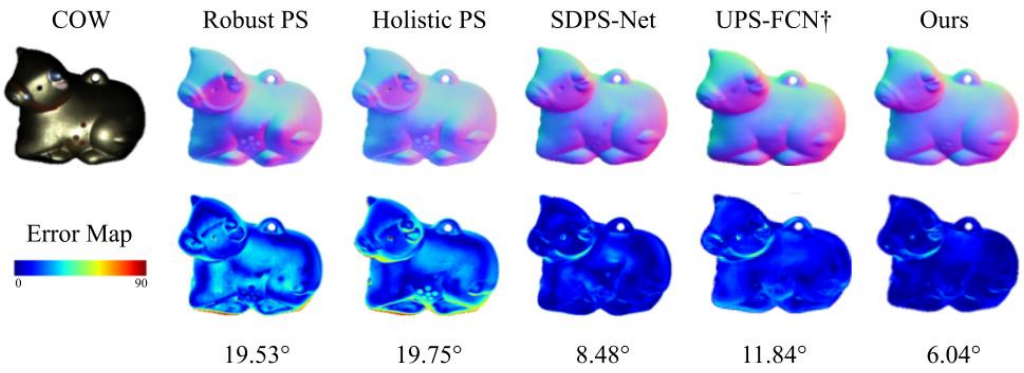


Figure 12: Visual comparison of uncalibrated photometric stereo results on **COW** scene.

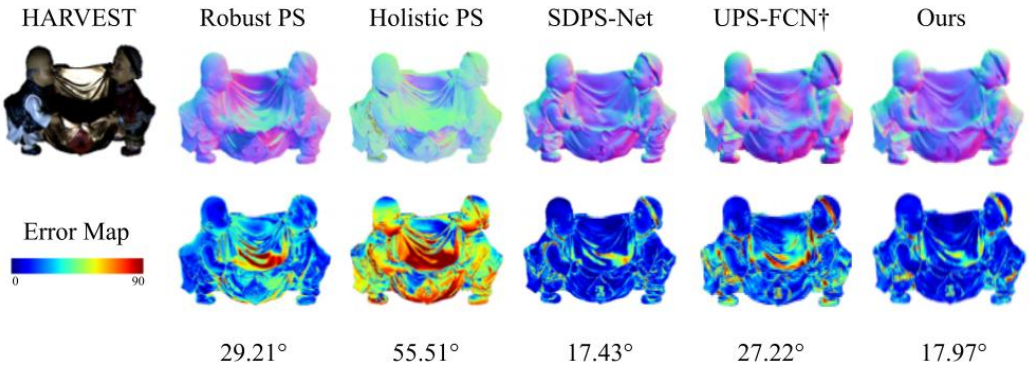


Figure 13: Visual comparison of uncalibrated photometric stereo results on **HARVEST** scene.

2.2. Qualitative Performance Comparison on Light Stage Data Gallery

To further analyze the performance of our method, we test it on additional datasets. Accordingly, we present a visual comparison of our results with the existing uncalibrated deep PS methods [4, 3] on the Light Stage Data Gallery [2]. The Light Stage Data Gallery is composed of 6 objects with challenging geometry and surface reflectance properties. As the dataset does not provide ground-truth normals, we limit our analysis to a qualitative comparison. In Fig.14-19, we compare the estimated surface normals. Although we cannot perform quantitative analysis due to lack of ground-truth normals, our method clearly performs better or comparable with other deep learning approaches [4, 3]. We also observed that our method estimates surface normals consistently on smooth surface regions while UPS-FCN†[4] results have serious artifacts. In addition to that, our method performs consistently well on fine-detailed surfaces (see Fig.18).

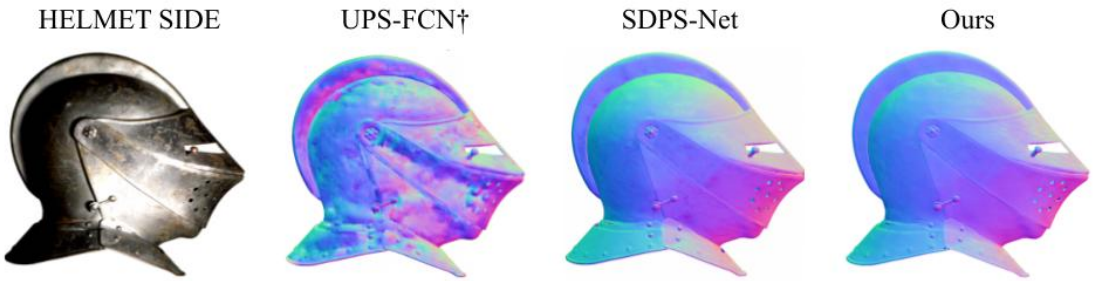


Figure 14: Visual comparison of uncalibrated photometric stereo results on **HELMET SIDE** scene.

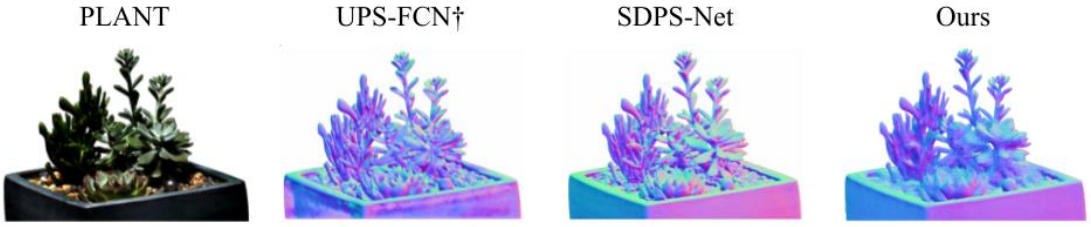


Figure 15: Visual comparison of uncalibrated photometric stereo results on **PLANT** scene.

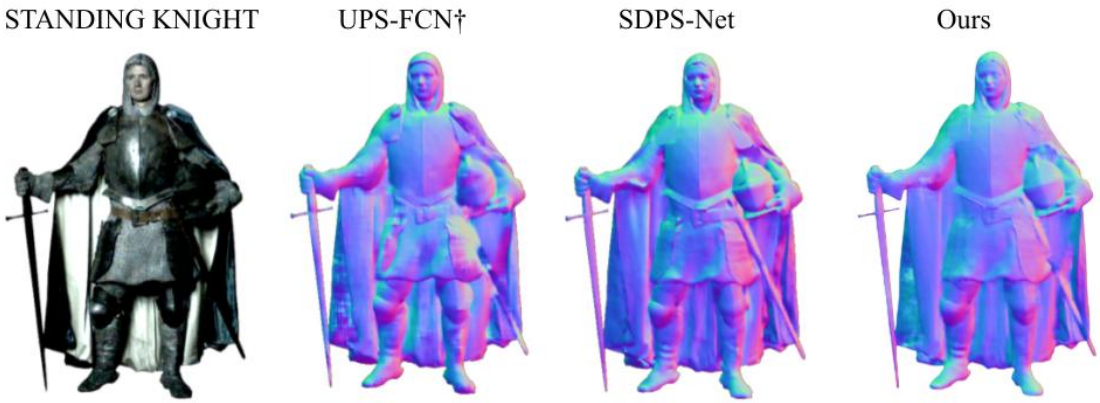


Figure 16: Visual comparison of uncalibrated photometric stereo results on **STANDING KNIGHT** scene.

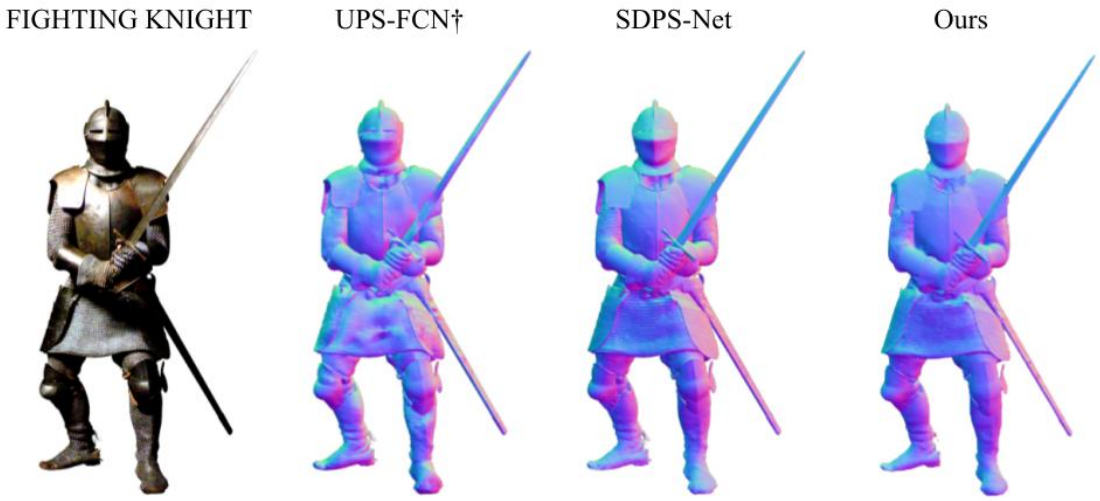


Figure 17: Visual comparison of uncalibrated photometric stereo results on **FIGHTING KNIGHT** scene.

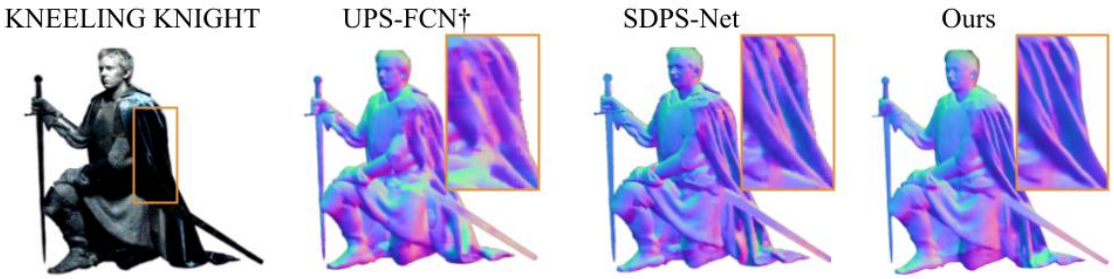


Figure 18: Visual comparison of uncalibrated photometric stereo results on **KNEELING KNIGHT** scene. The highlighted zone demonstrates the fine details achieved by our method.

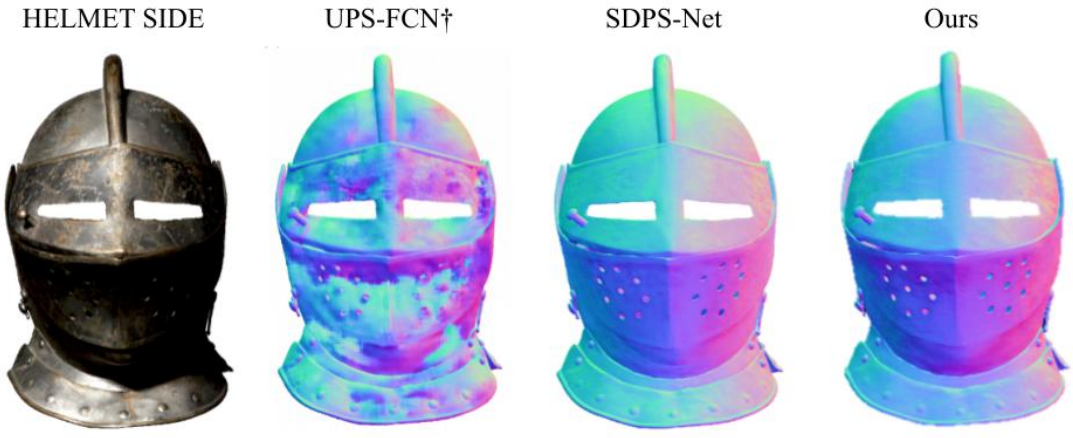


Figure 19: Visual comparison of uncalibrated photometric stereo results on **HELMET FRONT** scene.

2.3. Qualitative Performance Comparison on Gourd&Apple Dataset

The Gourd&Apple dataset [1] consists of 3 objects characterized by rough surfaces. Similar to Light Stage Data Gallery, this dataset does not provide ground-truth surface normals. Therefore, we provide a visual comparison of the estimated surface normals using our method and the existing deep PS methods [4, 3] (see Fig.20-22).

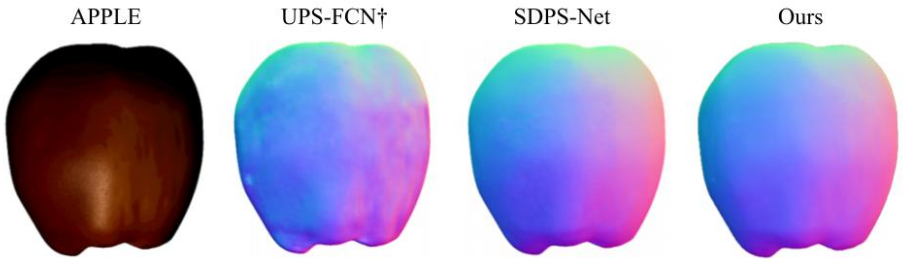


Figure 20: Visual Comparison among uncalibrated photometric stereo results on **APPLE** scene.

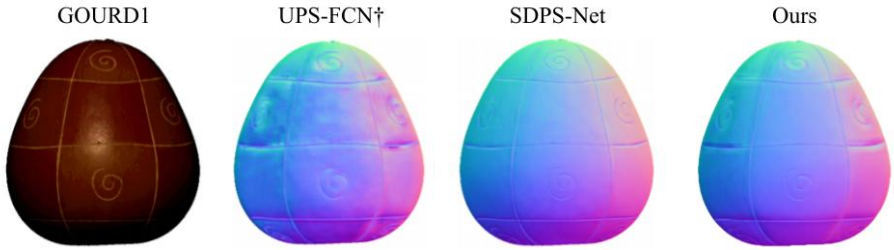


Figure 21: Visual Comparison among uncalibrated photometric stereo results on **GOURD1** scene.

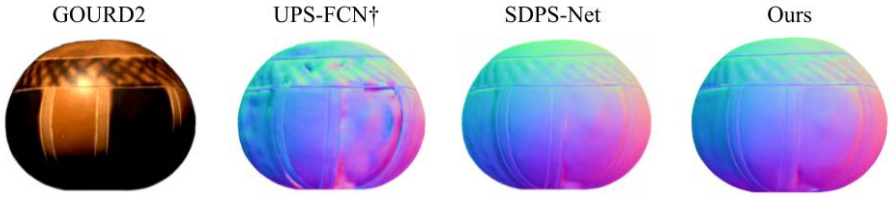


Figure 22: Visual Comparison among uncalibrated photometric stereo results on **GOURD2** scene.

3. Ablation Studies

In this section, we provide additional experimental results to validate our design choices. First, we show the effect of using an auxiliary tower at train time. Second, we study the consequences of changes in cell structures by reducing the number of intermediate nodes. We also analyze the selection of candidate operations and the search set.

3.1. Auxiliary Tower

In this study, we compare the results obtained with and without using an auxiliary tower at train time. As mentioned in the main paper, the performance of the normal estimation network obtained with architecture search is enhanced with the usage of an auxiliary tower. Our auxiliary tower applies two convolutional layers to the aggregated feature block as shown in Fig 23. Then, it uses a deconvolution, a convolution, and a normalization layer to get a surface normal estimate. In other words, the auxiliary tower is created by removing the searched part of our regressor module. Fig.24 shows the surface normal estimations obtained by removing the auxiliary tower at train time. The results clearly show that employing an auxiliary tower boosts performance. This behavior, which has already been observed in [7], demonstrates how the inclusion of a regularization structure enables a searched architecture to be robust in challenging testing scenarios.

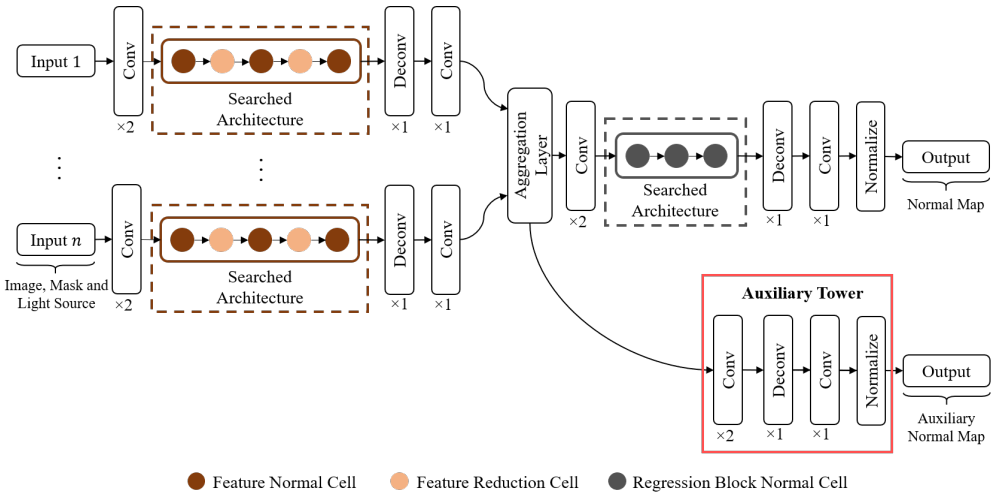


Figure 23: The pipeline of the Normal Estimation Network at train time. We employ the auxiliary tower (red box) to enhance the training.

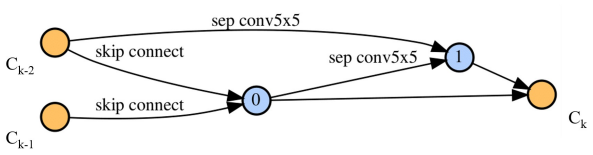
(a) Estimated normals and corresponding error maps **not** using Auxiliary Tower

(b) Estimated normals and corresponding error maps using Auxiliary Tower

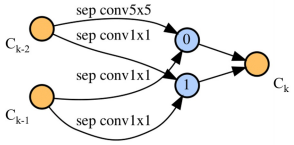
Figure 24: Estimated normals and error maps on DiLiGenT benchmark [9] obtained by (a) removing auxiliary tower in the normal estimation network at train time, (b) using the auxiliary tower at train time.

3.2. Reduced Number of Nodes in the Cell Structure

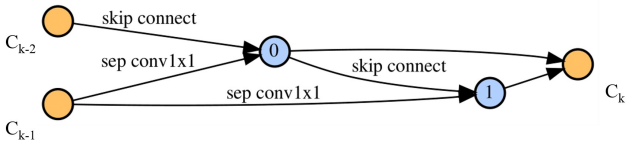
In the second case study, we investigate the skeleton structure of the searched cells. As mentioned in the main paper, we use cells with 2 input nodes, 4 intermediate nodes, and 1 output node as the cells used in DARTS work [7]. Here, we analyze the effect of reducing the number of intermediate nodes in the normal estimation network. To that end, we conduct experiments using cells with 2 intermediate nodes, while keeping all implementation details the same. The main motivation of this experiment is to see whether we can further reduce the number of parameters without a reduction in surface normal accuracy. Fig.25 shows the cell designs obtained with this reduced setting, where each cell has only 5-nodes. In Fig.26, we compare the surface normals obtained with 2 intermediate nodes and 4 intermediate nodes. As expected, reducing the number of nodes causes a degradation in the performance. So, we conclude that there exists a trade-off between the simplicity of the framework and the accuracy.



(a) Normal Estimation Network Feature Extractor (Normal cell)



(b) Normal Estimation Network Feature Extractor (Reduction cell)



(c) Normal Estimation Network Regressor (Normal cell)

Figure 25: Illustration of the cells obtained with neural architecture search on the Normal Estimation Network using reduced number of intermediate nodes. (a) Feature extractor normal cell. (b) Feature extractor reduction cell. (c) Regressor normal cell.

(a) Estimated normals and corresponding error maps using cells with 2 intermediate nodes

(b) Estimated normals and corresponding error maps using cells with 4 intermediate nodes

Figure 26: Estimated normals and error maps obtained by (a) employing cells with 2 intermediate nodes in the normal estimation network and (b) employing cells with 4 intermediate nodes in the normal estimation network.

3.3. Performance Variation with the Change in the Search Space

In the third ablation study, we demonstrate the effectiveness of the proposed search space. The search space presented in the main paper is composed of common operations adopted by deep PS methods. Namely: $\mathcal{O}^{normal} = \mathcal{O}^{light} = \{$"$1 \times 1$ separable conv.", "3×3 separable conv.", "5×5 separable conv.", "skip connection", "zero"$\}$. While the fundamental role of "skip connection", "zero" in NAS-based methods has been demonstrated by previous works such as [7, 5], the

(a) Estimated normals and corresponding error maps with search space $\mathcal{O}^1$.

(b) Estimated normals and corresponding error maps with search space $\mathcal{O}^2$.

(c) Estimated normals and corresponding error maps with search space $\mathcal{O}^3$.

(d) Estimated normals and corresponding error maps using the complete search space.

Figure 27: Estimated normals and error maps obtained with search space (a) $\mathcal{O}^1$ (b) $\mathcal{O}^2$ (c) $\mathcal{O}^3$ (d) $\mathcal{O}^{normal}$

variations in convolution kernel size has not been investigated. To that end, in Fig.27, we compare our method's performance with the performance of the other architectures obtained using the following three different candidate operations sets:

- $\mathcal{O}^1 = \{$"1×1 separable conv.", "skip connection", "zero"$\}$ Fig.27(a).
- $\mathcal{O}^2 = \{$"3×3 separable conv.", "skip connection", "zero"$\}$ Fig.27(b).
- $\mathcal{O}^3 = \{$"5×5 separable conv.", "skip connection", "zero"$\}$ Fig.27(c).

The ablation results are shown in Fig.27(a), 27(b), 27(c), 27(d). It demonstrates our NAS-based approach's effectiveness

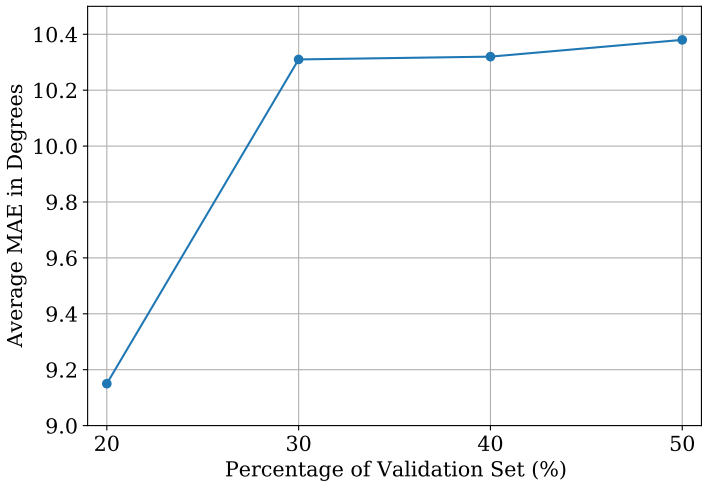


Figure 28: Average MAE [°] achieved on DiLiGenT dataset by architecture obtained with different percentages of search validation set.

on various search spaces by providing us meaningful surface normal. However, the search space described in the main paper provides better results (Fig.27(d)).

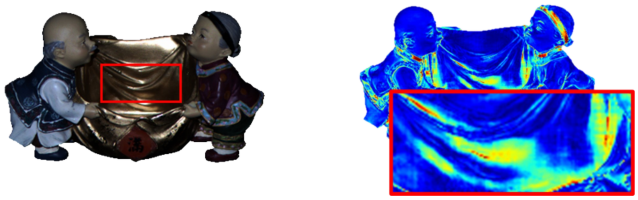
3.4. Effect of variation in the search train set, search validation sets

Utilizing a one-shot NAS framework such as [7] for an optimal architecture search for our task consumes a considerable amount of time. Having a large percentage of the dataset for searching for an optimal architecture can indeed provide a better architecture but, it can consume an enormous amount of time. Therefore, to have a better comprise between the search cost for an optimal architecture and performance, we used 10% of the total dataset for our search task. Using such a reduced dataset takes approximately two days with a single GPU (48 GPU hours) to search for the optimal PS architecture with a favorable performance accuracy. Further, we examined the searched architecture obtained using a higher percentage of the dataset for the network search. Still, it does not provide a good performance gain and consumes significantly more time, so we stick to the 10% stat.

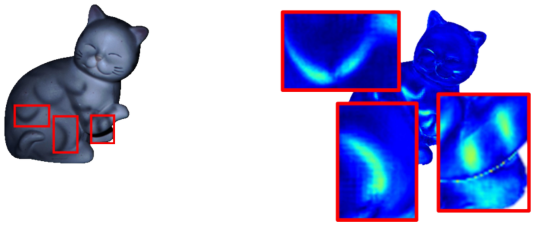
Secondly, we validate the separation of our search set into search train set and search validation set in 80%:20% split ratio. Fig.28 compares the performance of our searched architectures with different validation set percentages on the DiLiGenT benchmark [9]. The numerical values show that increasing the validation portion further degrades the performance, and therefore we stick to this 80%:20% split ratio.

4. Limitation and Further Study

In this work, we have successfully demonstrated a favorable way to exploit the differentiable NAS to the uncalibrated photometric stereo. By respecting the inherent constraint of uncalibrated PS, we utilize NAS that provides commendable performance and a lightweight neural network design. Still, we believe there are possible future directions to explore and handle the limitations of the current method. Firstly, our method considers a setup where each point is illuminated only by a directional light source. However, each surface element mutually illuminates each other on concave parts, and therefore, our method suffers on such interreflecting surfaces (see Fig.29(a)). Inspired by the Kaya *et al*. recent work [6], we aim to apply NAS on an inverse rendering pipeline with explicit interreflection modeling. Secondly, our method works well on the surfaces with homogeneous BRDF, and we observe degradation in performance on textured regions (see 29(b)). That is because the dataset used in training consists of texture-less surfaces. We believe that extending the dataset to spatially varying BRDFs will enhance the performance on such surfaces. As creating a large-scale dataset is not an easy task, it could be possible to improve the performance by exploring the applicability of techniques like channel-wise normalization.



(a) **HARVEST** scene



(b) **CAT** scene

Figure 29: Failure cases: (a) interreflecting surfaces, (b) textured surfaces.

References

[1] Neil Alldrin, Todd Zickler, and David Kriegman. Photometric stereo with non-parametric and spatially-varying reflectance. In *2008 IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–8. IEEE, 2008.

[2] Charles-Félix Chabert, Per Einarsson, Andrew Jones, Bruce Lamond, Wan-Chun Ma, Sebastian Sylwan, Tim Hawkins, and Paul Debevec. Relighting human locomotion with flowed reflectance fields. In *ACM SIGGRAPH 2006 Sketches*, pages 76–es. 2006.

[3] Guanying Chen, Kai Han, Boxin Shi, Yasuyuki Matsushita, and Kwan-Yee K Wong. Self-calibrating deep photometric stereo networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 8739–8747, 2019.

[4] Guanying Chen, Kai Han, and Kwan-Yee K Wong. Ps-fcn: A flexible learning framework for photometric stereo. In *Proceedings of the European conference on computer vision (ECCV)*, pages 3–18, 2018.

[5] Xiangxiang Chu, Tianbao Zhou, Bo Zhang, and Jixiang Li. Fair darts: Eliminating unfair advantages in differentiable architecture search, 2020.

[6] 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 Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2021.

[7] Hanxiao Liu, Karen Simonyan, and Yiming Yang. Darts: Differentiable architecture search. *arXiv preprint arXiv:1806.09055*, 2018.

[8] Thoma Papadhimitri and Paolo Favaro. A closed-form, consistent and robust solution to uncalibrated photometric stereo via local diffuse reflectance maxima. *International journal of computer vision*, 107(2):139–154, 2014.

[9] 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 Conference on Computer Vision and Pattern Recognition*, pages 3707–3716, 2016.

[10] Zhe Wu and Ping Tan. Calibrating photometric stereo by holistic reflectance symmetry analysis. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1498–1505, 2013.