# Construction d'un jeu de données d'apprentissage adapté pour la reconstruction 3D par stéréophotométrie

Clément HARDY, Yvain QUÉAU, David TSCHUMPERLÉ

Normandie Univ, UNICAEN, ENSICAEN, CNRS, GREYC, Caen, France
{Clement.Hardy, Yvain.Queau, David.Tschumperle}@unicaen.fr

**Résumé –** La stéréophotométrie est une technique de reconstruction 3D de la surface d'un objet, à partir de plusieurs images de celui-ci, prises du même point de vue mais avec différents angles d'éclairage. Dans cet article, une revue des méthodes récentes de stéréphotométrie est d'abord proposée. Dans un second temps, nous définissons un ensemble de contraintes à suivre pour la génération d'un jeu de données d'apprentissage pertinent, utilisable pour l'apprentissage de réseaux de neurones convolutionnels dédiés à la stéréophotométrie. Nous illustrons l'importance de la variabilité de ce jeu de données, avec des exemples comparatifs de reconstruction de champs de normales.

**Abstract –** Photometric stereo is a technique for the 3D-reconstruction of the surface of an object, from several images of it, shot from the same point of view but with different lighting angles. In this paper, a review of recent photometric stereo methods is proposed. Then, a set of constraints is defined, to allow the generation of a relevant dataset for the training of convolutional neural networks dedicated to photometric stereo. The importance of the variability of such a dataset is finally illustrated, with comparative examples of normal fields reconstruction.

## 1 Contexte

La stéréophotométrie est une technique de reconstruction 3D permettant d'estimer la normale en chaque point de la surface d'un objet, à l'aide de $m \geqslant 3$ photographies de cet objet, prises depuis le même point de vue mais avec des directions d'éclairage différentes. La carte de normales ainsi estimée peut ensuite être intégrée en une carte de profondeur représentant le relief (altitude de chaque point), comme cela est présenté sur la Fig. 1. Les premiers travaux dans ce domaine [22] ont considéré le cas idéal d'une surface parfaitement lambertienne, c'est-à-dire dont la réflectance est caractérisée par un scalaire appelé l'albédo. Or, les images de la plupart des objets du monde réel exhibent des effets lumineux d'une grande diversité, qui ne sont pas bien prédits par la loi de Lambert. En particulier, leur réflectance comporte le plus souvent une composante spéculaire, conférant un aspect brillant à la surface imagée. Cette composante spéculaire met généralement en défaut la résolution classique de la stéréophotométrie.
Pour tenir compte des reflets spéculaires, trois grandes familles de méthodes ont été développées dans la littérature. La résolution classique de la stéréophotométrie a d'abord été étendue à des modèles de fonctions de distribution de la réflectance bidirectionnels (BRDF) plus généraux, comme par exemple le modèle de Torrance et Sparrow [8]. Une autre approche consiste à traiter les spécularités comme des données aberrantes et à recourir à des techniques d'estimation robuste [11]. Plus récemment, les méthodes fondées sur l'apprentissage profond par réseaux de neurones se sont imposées comme les plus efficaces pour gérer ce type de surface [4, 18, 20]. C'est dans ce contexte que s'inscrit le travail que nous présentons ici.

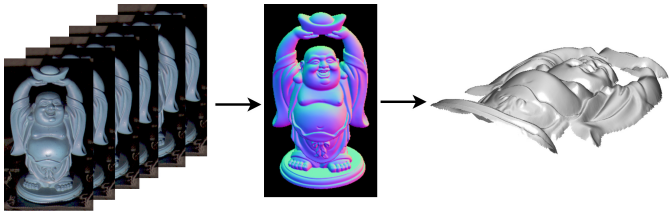


FIG. 1: Principe de la stéréophotométrie. À partir d'un ensemble d'images prises sous des directions d'éclairage différentes (à gauche), une carte de normales peut être estimée (au centre), de laquelle on peut déduire une carte de profondeur par intégration. Dans ce travail nous nous intéressons à l'amélioration des techniques neuronales pour l'estimation des normales.

La qualité des résultats obtenus par les approches fondées sur l'apprentissage profond repose sur deux facteurs principaux : 1. la qualité de la base d'apprentissage, qui doit être la plus représentative possible de la diversité des phénomènes lumineux observables ; 2. l'architecture du réseau, qui doit assurer une bonne capacité de généralisation sur de nouvelles données.
Ici, nous contribuons à améliorer ce premier facteur : nous élaborons un nouveau jeu de données synthétique adapté, et montrons qu'entraîner un des réseaux de l'état de l'art en stéréophotométrie sur ces nouvelles données induit un gain significatif dans la précision des cartes de normales estimées. Ce jeu de données est disponible en accès libre, pour favoriser une recherche ouverte et reproductible. En section 2, une vue d'ensemble des techniques existantes de stéréophotométrie par apprentissage est proposée. Nous introduisons les règles de génération de notre base d'apprentissage en section 3, suivie d'une comparaison quantitative des performances obtenues.

## 2 Revue de l'état de l'art

Les techniques d'apprentissage profond pour la stéréophotométrie reposent sur l'utilisation de réseaux de neurones convolutionnels. Généralement, une architecture de réseaux purement convolutionnels requiert un nombre donné (fixe) d'images d'entrées. Mais en stéréophotométrie, le nombre d'images varie car il dépend du mode opératoire de l'acquisition. Pour éviter d'avoir à entraîner un modèle de réseau différent pour chaque nombre possible d'images d'éntrées, deux alternatives ont été proposées dans la littérature : la première consiste à utiliser une carte d'observation (*observation map*) [16, 23], qui projette toutes les observations correspondant à un même pixel sous des éclairages différents dans un espace de taille fixe - typiquement un hémisphère échantillonné. Le but d'une carte d'observation est donc de faire un "résumé" de taille fixe des informations contenues dans un ensemble de taille variable d'images. La deuxième alternative consiste à recourir à un module de *pooling* [4, 12, 14, 21], qui aggrège les différentes caractéristiques de chaque image extraites par des couches préalables de convolution (Fig. 2). Cela permet d'obtenir des caractéristiques d'images de taille fixe à partir d'un nombre variable d'images d'entrées. Différentes méthodes de *pooling* peuvent être considérées. Il est montré dans [5] que le *max pooling* se montre plus performant que l'*average pooling* dès que le nombre d'images est supérieur à 16. Ce dernier a en effet tendance à sur-lisser les caractéristiques saillantes et à être trop sensible aux régions des images ayant peu d'intérêt. À l'inverse, il a été montré dans [13] que le *max pooling* peut parfois ignorer une grande partie des caractéristiques extraites des images, qui peuvent être pourtant pertinentes.

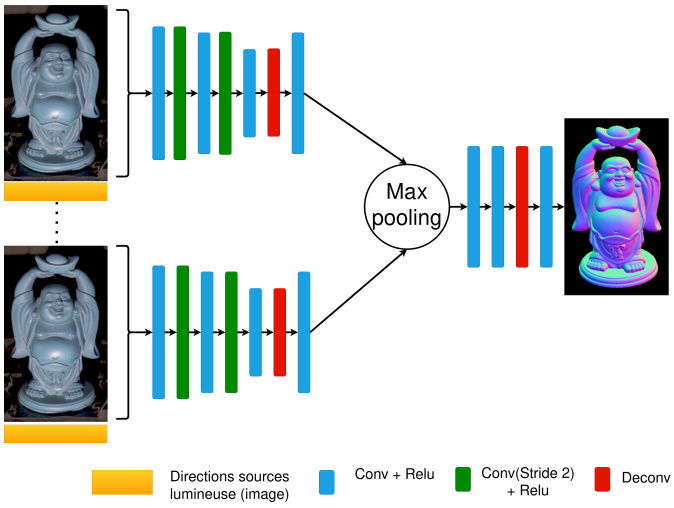


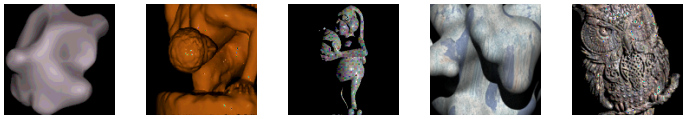

FIG. 2: Architecture neuronale utilisée dans [4] pour la stéréophotométrie calibrée. L'estimation d'une carte de normales (à droite) est réalisée à partir d'images d'entrées (à gauche) qui concatènent les images en couleurs (compensées au préalable par les intensités lumineuses) et les directions d'éclairage associées.

Ici, nous nous focalisons sur la résolution du problème de la stéréophotométrie *calibrée*, c'est-à-dire que nous supposons connues les différentes orientations de la source lumineuse, et que nous compensons les images par les intensités associées à chacune des images. Lorsque les orientations des lumières sont inconnues, on parle de stéréphotométrie *non calibrée*. Le problème de la stéréophotométrie non calibrée a été étudié dans [4] et [15], et partiellement résolu en définissant un réseau de neurones préalable de prédiction de l'orientation de l'éclairage associé à chaque image. Ces données estimées alimentent ensuire un second réseau qui, lui, résout le problème de la stéréophotométrie calibrée. Pour nos expériences, nous reprenons l'architecture neuronale de [4] pour l'estimation de la normale dans le cas *calibré*. Cette architecture, présentée en Fig. 2, est divisée en trois modules distincts : un module d'extraction de caractéristiques (à gauche), un module d'aggrégation de ces caractéristiques (au centre), et un module de régression qui prédit la carte de normales (à droite). Notons que le module d'extraction traite chaque couple image/direction de manière totalement indépendante. L'orientation de la lumière (vecteur de $\mathbb{R}^3$) est représentée sous forme d'une imagette couleur, concaténée à l'image associée. L'image résultante constitue l'entrée de l'extracteur de caractéristiques.

## 3 Une base d'apprentissage améliorée

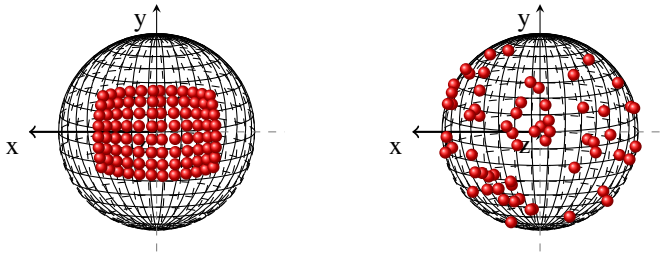
Il est en pratique difficile d'acquérir une grande base de données d'images *réelles*, accompagnées de vérités terrains 3D pour les objets photographiés. Pour cette raison, les techniques de stéréophotométrie par apprentissage statistique reposent en général sur l'utilisation de bases de données d'objets 3D *synthétiques*, notamment les bases "*Blobby*", "*Structure*" introduites dans [5] et "*CyclePS*" dans [10].

**Bases d'apprentissages existantes.** La base de données *Blobby* est composée de 10 formes géométriques, chacune observée depuis 1296 points de vue distincts. Comme son nom l'indique, les formes présentes dans *Blobby* sont lisses et régulières (Fig. 3a). Le jeu de données *Structure* est, quant à lui, constitué d'objets à géométries complexes contenant des détails fins (Fig. 3b). Il comporte 8 objets, imagés de 1387 à 6874 points de vue. Pour simuler des surfaces avec une réflectance lumineuse non lambertienne, un matériau de la base *MERL* [17] est tiré aléatoirement, et appliqué lors de chaque rendu, fournissant au total 25920 échantillons pour *Blobby* et 59292 pour *Structure*. Dans les deux cas, chaque échantillon est imagé sous 64 directions lumineuses différentes, sélectionnées aléatoirement sur l'hémisphère (Fig. 4b). Enfin, la base *CyclePS* [10] est également composée d'objets complexes, mais ne contient que 18 objets imagés depuis 10 vues (Fig. 3c). Le nombre de matériaux disponible est, par contre, conséquent, puisque la réflectance paramétrique *Disney's principled BSDF* [3] est utilisée, ce qui permet de faire varier la couleur, la rugosité ou la proportion de réflectance spéculaire, etc.), et ainsi de générer une quasi-infinité de matériaux différents pour habiller ces objets 3D.



(a) *Blobby* (b) *Structure* (c) *CyclePS* (d) *GREYC Blobby* (e) *GREYC Object*

FIG. 3: Exemples d'images des jeux de données de l'état de l'art *Blobby*, *Structure* et *CyclePS*, et des deux jeux de données proposés dans cet article (*GREYC Blobby* et *GREYC Object*).




(a) Répartition de *DiLiGenT* (b) Répartition aléatoire.

FIG. 4: Distribution des directions d'éclairage dans le jeu de données réelles *DiLiGenT*, et exemple d'une distribution aléatoire. L'axe $z$ correspond à l'axe optique de l'appareil photographique, l'objet imagé se situant aux coordonnées $(0, 0, 0)$.

**Variabilité de la géométrie et des matériaux.** En pratique, on peut remarquer que tous ces jeux de données synthétiques manquent de diversité en terme de géométrie et de textures. Par exemple, bien que la base *Structure* soit composée d'objets complexes, tous ces objets sont des statues. De même, le nombre de matériaux différents de la base de matériaux *MERL* n'est que de 100, ce qui est nettement insuffisant pour modéliser l'immense diversité des matériaux présents dans la nature. On peut donc penser qu'une plus grande diversité de formes et de matériaux dans les images de la base d'apprentissage s'avèrerait bénéfique pour l'entraînement de réseaux de neurones pour la stéréophotométrie. C'est ce que nous montrons dans la suite de cet article. Pour ce faire, nous créons notre propre *pipeline* de génération de données images pour l'apprentissage des réseaux. En pratique, nous utilisons le logiciel *Blender* [6] et son moteur de rendu *Cycles*. Nous générons deux jeux de données : *GREYC Blobby* contient des objets avec des surfaces lisses, et *GREYC Object* des objets complexes présentant des discontinuités fortes, des arêtes, des coins, etc. Notre version de *Blobby* comporte 3000 objets distincts, générés par la somme de potentiels gaussiens aléatoires, suivi d'une extraction d'iso-surfaces par l'algorithme du *Marching Cubes*. Les 56 objets plus détaillés sont des maillages 3D provenant du site *Sketchfab* [2]. Pour permettre l'apprentissage de surfaces non lambertiennes, plus de 1100 matériaux différents, extraits du site *Ambientcg* [1], sont appliqués aléatoirement sur les objets (soit bien plus que les 100 matériaux de *Structure* et *Blobby*). Des exemples d'images issues de ces bases sont illustrées en Fig. 3d,e. Dans un but de reproductibilité de nos résultats, nous mettons aussi à disposition le code de génération de ces deux bases [9].

**Sélection des éclairages.** Pour valider la pertinence de nos jeux de données, et pour vérifier que les modèles entraînés sur ces données synthétiques sont en capacité de généraliser sur des images réelles, nous évaluons les performances de reconstruction sur le jeu de données *réelles DiLiGenT* [19]. Cette base contient 10 objets différents, pris depuis le même point de vue sous 96 éclairages différents (Fig. 4a). Pour chaque objet photographié, la vérité terrain (cartes de normales) a été obtenue via la mesure de la profondeur par un scanner laser. Cela permet de mener une évaluation quantitative sur des données réelles. Nous aurions également pu utiliser le jeu de données de validation SynthPS [7], mais celui-ci est composé d'images synthétiques. Afin de limiter le biais lié à une différence de variabilité entre les éclairages utilisés pour la base d'apprentissage et pour le jeu de validation, les images de nos jeux de données sont générées avec les mêmes 96 directions d'éclairage que celles définies par *DiLiGenT*, en ajoutant cependant un bruit léger sur les coordonnées 3D de la source lumineuse, afin de rendre le modèle plus robuste à une possible imprécision de positionnement des sources. De même, pour permettre aux jeux de données d'être représentatifs de différentes conditions d'éclairages, l'intensité lumineuse de la source est également choisie de manière aléatoire. Le tableau 1 récapitule les caractéristiques des bases existantes, et de celles que nous proposons.

TAB. 1: Tableau récapitulatif des caractéristiques des différentes bases d'apprentissage utilisables en stéréophotométrie

| | # objets | # vues | # total échantillons | # éclairages | # matériaux |
|---|---|---|---|---|---|
| *Blobby* | 10 | 1 296 | 25 920 | 64 | 100 |
| *Structure* | 8 | 1387-6874 | 59 292 | 64 | 100 |
| *CyclePS* | 18 | 10 | 180 | 1 300 | 90 000 |
| *GREYC Blobby* | 3000 | 5 | 15 000 | 96 | 1 100 |
| *GREYC Object* | 56 | 267 | 15 000 | 96 | 1 100 |

# 4 Comparaison des performances

La mesure de performance de reconstruction 3D est réalisée en mesurant l'erreur angulaire moyenne (en degrés) sur l'orientation des normales estimées. Le tableau 2 résume les performances obtenues en entraînant le réseau avec les jeux de données de la colonne de gauche et en le validant sur chacun des jeux de données. On peut tout d'abord remarquer que l'erreur moyenne lors de la validation sur les jeux *Blobby* est plus faible que sur les autres jeux de données, ce qui paraît logique au vu de la simplicité des formes présentes dans *Blobby*. La validation est plus pertinente si elle est menée sur le jeu de données réelles *DiLiGenT*. Sur ce jeu de validation, les résultats semblent quantitativement similaires si l'on entraîne sur *Blobby* ou sur *Structure* (resp., sur *GREYC Blobby* ou *GREYC Object*). Mais en réalité, les résultats obtenus en apprenant uniquement sur les *Blobby* sont sur-lissés, comme cela est illustré qualitativement sur l'exemple du *Buddha* de *DiLiGenT* et sur une pièce de monnaie de *RealRTI* (Fig. 5). Enfin, les résultats s'améliorent lorsque les bases d'apprentissage sont combinées. En effet, l'entraînement en parallèle sur *GREYC Object* et *GR-*

*EYC Blobby* donne de meilleurs résultats qu'un entraînement séparé sur *GREYC Object* ou *GREYC Blobby* (et de même pour *Blobby* et *Structure*).

Surtout, le modèle entraîné sur l'ensemble des jeux de données (existants et proposés) est plus performant que ceux entrainés uniquement sur les bases existantes ou celles proposées. Ceci est clairement visible par l'évaluation sur les données réelles de *DiLiGenT* : ajouter nos données aux bases couramment utilisés permet de diminuer l'erreur moyenne de $9.25$ degrés à $8.79$ degrés, soit un gain appréciable de $6\%$.

TAB. 2: Erreur moyenne angulaire obtenue selon le jeu de données utilisé pour l'entraînement (première colonne) ou la validation (colonnes deux à six). L'écart type est donné entre parenthèses. Les jeux de données sont nommés par acronymes. DB1 : *Structure*, DB2 : *Blobby*, DB3 : *GREYC Blobby*, DB4 : *GREYC Object*. Dans l'état de l'art l'apprentissage est réalisé sur DB1 + DB2. Augmenter cette base avec nos jeux de données (DB1 + DB2 + DB3 + DB4) permet une amélioration significative des résultats sur données réelles (en gras).

| | *Structure* | *Blobby* | *GREYC Blobby* | *GREYC Object* | *DiLiGenT* |
|---|---|---|---|---|---|
| DB1 | 8.44 | 5.06 | 10.9 | 15.39 | 9.38 (4.05) |
| DB2 | 9.24 | 3.34 | 10.7 | 16.75 | 10.12 (4.09) |
| DB3 | 14.76 | 13.52 | 3.03 | 11.03 | 11.9 (3.82) |
| DB4 | 16.33 | 16.53 | 5.69 | 8.56 | 11.26 (3.42) |
| DB1+DB2 | 9 | 5.20 | 11.24 | 15.84 | 9.25 (3.80) |
| DB3+DB4 | 13.48 | 14.65 | 3.46 | 7.22 | 11.24 (3.30) |
| DB1+DB2 +DB3+DB4 | 8.74 | 4.91 | 4.72 | 9.48 | **8.79** (3.39) |

FIG. 5: De gauche à droite : une des images d'entrée, carte de normales obtenue lorsque l'apprentissage est réalisé sur *Structure*, et carte de normales obtenue lorsque l'apprentissage est réalisé sur *Blobby*, pour deux objets réels issues des jeux de données *DiLiGenT* (en haut) et RealRTI (en bas).

# References


[1] Contains assets from ambientCG.com, licensed under CC0 1.0 Universal. `https://ambientcg.com/`.

[2] Sketchfab. `https://sketchfab.com`.

[3] Brent Burley and Walt Disney Animation Studios. Physically-based shading at Disney, part of practical physically based shading in film and game production. *Proceedings of ACM SIGGRAPH Courses*, 2012.

[4] Guanying Chen, Kai Han, Boxin Shi, Yasuyuki Matsushita, and Kwan-Yee K. Wong. 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, and Kwan-Yee K. Wong. Ps-fcn: A flexible learning framework for photometric stereo. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018.

[6] Blender Online Community. *Blender - a 3D modelling and rendering package*. Blender Foundation, Stichting Blender Foundation, Amsterdam, 2018.

[7] Tinsae Gebrechristos Dulecha, Ruggero Pintus, Enrico Gobbetti, and Andrea Giachetti. Synthps: a benchmark for evaluation of photometric stereo algorithms for cultural heritage applications. In *GCH*, pages 13–22, 2020.

[8] Athinodoros S Georghiades. Incorporating the Torrance and Sparrow model of reflectance in uncalibrated photometric stereo. In *Proceedings of the IEEE Conference on Computer Vision (ICCV)*, 2008.

[9] Clément Hardy, Yvain Quéau, and David Tschumperlé. `https://github.com/Clement-Hardy/blender_data_generation`.

[10] Satoshi Ikehata. CNN-PS: CNN-based Photometric Stereo for General Non-Convex Surfaces. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018.

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

[12] Yakun Ju, Junyu Dong, and Sheng Chen. Recovering surface normal and arbitrary images: A dual regression network for photometric stereo. *IEEE Transactions on Image Processing*, 30:3676–3690, 2021.

[13] Yakun Ju, Muwei Jian, Junyu Dong, and Kin-Man Lam. Learning photometric stereo via manifold-based mapping. In *Proceedings of the IEEE International Conference on Visual Communications and Image Processing (VCIP)*, 2020.

[14] Yakun Ju, Kin-Man Lam, Yang Chen, Lin Qi, and Junyu Dong. Pay attention to devils: A photometric stereo network for better details. In *Proceedings of the International Conference on International Joint Conferences on Artificial Intelligence*, 2020.

[15] 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.

[16] 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.

[17] Wojciech Matusik, Hanspeter Pfister, Matt Brand, and Leonard McMillan. A data-driven reflectance model. *ACM Transactions on Graphics*, 22(3):759–769, July 2003.

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

[19] 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 (CVPR)*, 2016.

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

[21] 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*, 29:6032–6042, 2020.

[22] Robert J. Woodham. Photometric Method For Determining Surface Orientation From Multiple Images. *Optical Engineering*, 19(1):139 – 144, 1980.

[23] Qian Zheng, Yiming Jia, Boxin Shi, Xudong Jiang, Ling-Yu Duan, and Alex C. Kot. Spline-net: Sparse photometric stereo through lighting interpolation and normal estimation networks. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2019.