RESEARCH IN LEARNING-BASED PHOTOMETRIC STEREO

Yiming Jia, Yuchen Xing, and Hao Zhao

University of California San Diego, La Jolla, CA 92093-0238,

ABSTRACT

Photometric stereo is a 3-D reconstruction technique, it is to estimate surface normal of objects from observations photographed under different lightings and single view point. Recent learning-based photometric stereo methods can be categorized into all-pixel methods and per-pixel methods. This paper proposes a data representation to combine the advantages of these two kinds of method, and also fix the problem they have.Performance of our method has been verified to be close to state-of-the art methods.

***Index Terms*—** photometric stereo, reconstruction, convolutional neural network

1. INTRODUCTION

Photometric stereo is a popular 3-D reconstruction approach in computer vision tasks, it is to estimate surface normal of 3-D objects from observations under different lightings (light intensity and direction) and single view point, as shown in Fig. 1a.

The main advantage of photometric stereo the surface normal map estimated has the same resolution with the original input, the shape information are retained at most. Compared to geometric approaches that generally reconstruct gross shape, photometric approach can get more detailed reconstruction, as shown in Fig. 1b[1].

Traditional photometric methods compute surface normal by solving the image formation model inversely,

$$I_i = L_i \rho(\mathbf{n}, \mathbf{l}_i, \mathbf{v}) max(\mathbf{n}^T \mathbf{l}_i, 0) \tag{1}$$

where I_i is the intensity of the observed pixel, L_i is the i^{th} light intensity, $\rho(\mathbf{n}, \mathbf{l}_i, \mathbf{v})$ is called Bidrectional Reflectance Distribution Function (BRDF) that describes reflectance of objects, $\mathbf{l}_i$ is the the i^{th} light direction, $\mathbf{v}$ is the fixed view point, $max(\mathbf{n}^T \mathbf{l}_i, 0)$ indicates objects can only be observed under certain lightings, that are actually attached shadows. Among these parameters, BRDF is unknown and affected my other complicated factors like materials except for $\mathbf{n}, \mathbf{l}_i, \mathbf{v}$, which makes the model hard to fit. And global illumination effects like cast shadow and inter-reflection are not included in this model, traditional methods just treat them as outliers.

[1]This figure is from ICIP 2019 Tutorial.

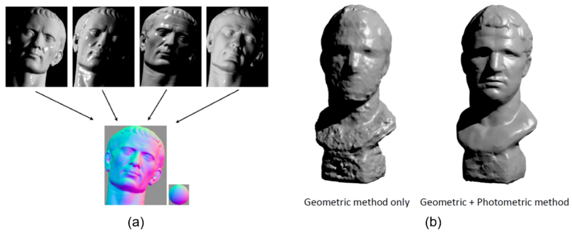

Fig. 1. a) Photometric stereo: Use observations under different lightings to reconstruct surface normal map, b) Photometric stereo approaches get more detailed shapes.

Thus, these methods are generally not accurate enough, especially on non-convex objects that has heavy global illumination effect.

Motivated by the modeling capability of deep learning methods, recent researchers have built some learning-based approaches and models [1, 2, 3, 4, 5, 6] that outperform traditional methods.

In this paper, we propose a new method mainly base on SPLINE-net [5], a CNN framework, input a image consists of observation maps[2] and output the surface normal map reconstructed.

2. RELATED WORKS

State-of-the-art learning-based photometric stereo methods[1, 2, 3, 4, 5, 6] can be divided into two categories according to how they process data, all-pixel methods[2, 3, 6] and per-pixel methods[1, 4, 5].

2.1. All-pixel methods

In all-pixel methods[2, 3, 6], surface normal map is seen as a common feature shared by all the observations under various lightings, researchers use CNN model to extract that feature automatically. This kind of method usually concatenates observed images and corresponding lightings as input, and output the surface normal map, as shown in Fig. 2. It's an end-to-end process since surface normal map extracted directly from the observations.

[2]More details in Section 3.

Fig. 2. PS-FCN[2], input several observations and lightings, output the surface normal map of the same resolution.

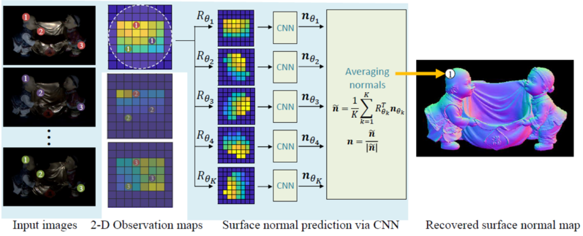

Fig. 3. CNN-PS[1], observations are converted into observation maps, there maps are used for normal estimation.

However, to assume surface normal map, that has certain spatial continuity, as a common feature of observations, requires observations also have some spatial continuity. That actually means this kind of method can be only apply to objects have uniform materials, hard to generalize into real-world.

2.2. Per-pixel methods

Per-pixel methods[1, 4, 5] process data pixel by pixel, each time input observed intensities of one pixel, output the normal vector. The input format is actually a better usage of CNN. Researchers convert original observations into a easy-learn data representation called observation map. The idea of observation map is to encode light direction as 2-D coordinates and for each pixel, put all the intensities into the map according to that coordinates. In this way, observation map has some easy-learn properties such as spatial continuity and symmetry. CNN-PS [1] directly uses observation maps to compute normal vectors, as shown in Fig. 3. LMPS[4] and SPLINE-Net[5] also use that observation map to investigate in a more pratical situation when the number of input is small.

This kind of method doesn't assume uniform materials since data is reorganized and processed pixel-wisely. But observation maps of different pixels have different pattern affected by global illumination, they may be asymmetric or discontinuous spatically, treat them equally is not good. Thus, per-pixel method is sensitive to global illumination effect and perform worse on complicated non-convex object than all-pixel method.

These two kinds of method both have their advantages and disadvantages. In section 3, we propose a method that combines the idea of these two and fixes the problems they have.

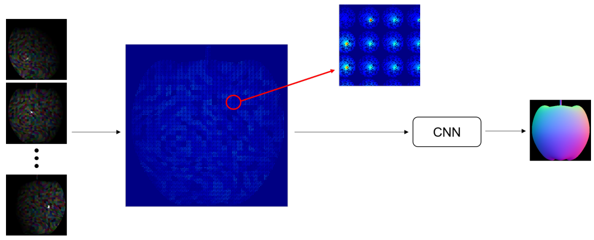

Fig. 4. Observation maps are put into one image, estimate surface normal map from this big image.

3. PROPOSED METHOD

In our method, we first generate observation maps as other per-pixel methods, but instead of processing seperately, we organize them into one big image[3]. The position each observation map takes is the position of the corresponding pixel. Then we input this big image to CNN, use that to compute the surface normal map, which is shown in Fig. 4. In this way, we take use of the easy-learn observation maps and shape information is also maintained.

The convolutional neural network we use is basically inherited from SPLINE-Net[5], it consists of two sub-networks, Lighting Interpolation Net (LI-Net, denoted as f) and Normal Estimation Net (NE-Net, denoted as g[4]).

Lighting Interpolation Net. LI-Net is an encoder-decoder, input of LI-Net is a matrix $\mathbf{S}$ consists of sparse observation maps, out is a matrix $\mathbf{D}$ consists of dense observation maps. Since observation map has some easy-learn properties such as symmetry and spatial continuity, it is feasible to increase the amount of data through interpolation. This process is formulated as,

$$f : \mathbf{S} \rightarrow \mathbf{D} \tag{2}$$

Loss function L_f of LI-Net is,

$$L_f = cos^{-1}(\mathbf{n}^{\mathbf{T}}\mathbf{n_{gt}}) + |\mathbf{D}-\mathbf{D_{gt}}|_1 + |\mathbf{M_s}\circ(\mathbf{D}-\mathbf{D_{gt}})|_1 \tag{3}$$

where $\mathbf{n}$ is the estimated normal vector, $\mathbf{n_{gt}}$ is ground truth of $\mathbf{n}$, $\mathbf{D_{gt}}$ is ground truth of $\mathbf{D}$, $\mathbf{M_s}$ is a binary mask that blocks the invalid area of observation map, $\circ$ represents element-wise multiplication. L1 loss is also the same set with SPLINE-Net.

Normal Estimation Net. NE-Net is a DenseNet of two dense blocks. It is to estimate surface normal map from $\mathbf{D}$. For better usage of the original input information, $\mathbf{S}$ and $\mathbf{D}$ are concatenated as the input, output is the estimated normal map.

$$g : \mathbf{S}, \mathbf{D} \rightarrow \mathbf{n} \tag{4}$$

[3]The original idea is from Yiming's project in ECE271b, the optimization of using Blobby & Sculpture dataset and further discussion are proposed in this project. Fig. 4 and Fig. 5 are also generated in Yiming's project in ECE271b.

[4]We use same notation and formula with SPLINE-Net[5]

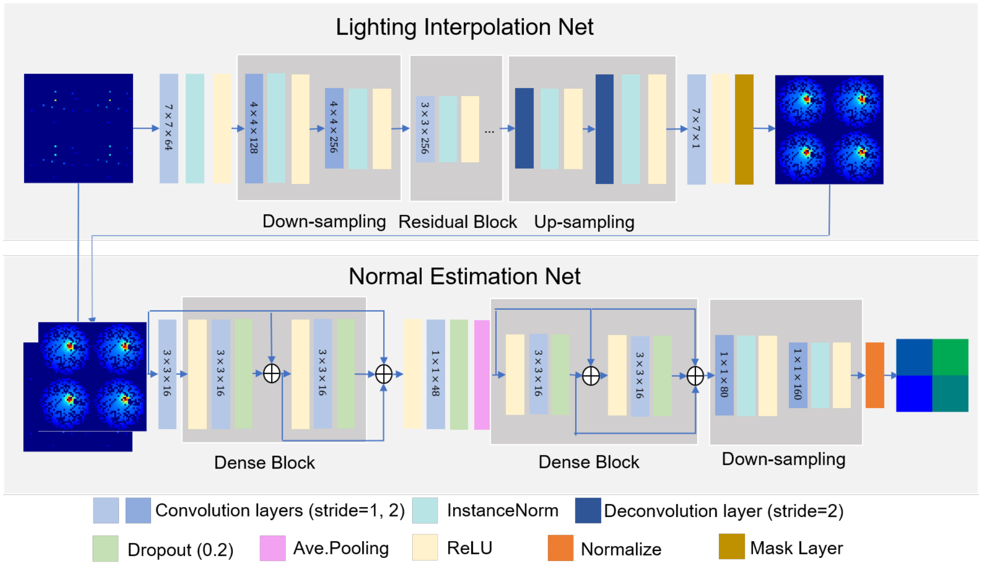

Fig. 5. Structure of our network. For better visualization, we use a small block (2×2) to show our input and output, real experiment setting is in Section 4.

Loss function L_g of NE-Net is,

$$L_g = cos^{-1}(\mathbf{n}^\mathbf{T}\mathbf{n_{gt}}) \tag{5}$$

In our method, we output a surface normal map of an object but not a normal vector of a single pixel like SPLINE-Net, so we change the output layer of NE-Net from fully-connected layer to convolutional layer.

Our network is shown in Fig. 5.

4. EXPERIMENT

4.1. Dataset

To train our model, we investigate in some popular datasets.

CyclesPS. CyclesPs is a synthetic train dataset proposed in [1] and generally used in per-pixel methods. It has 45 objects including 15 shapes times 3 kinds of reflectance (diffuse, specular and metallic), each one is rendered under 1000 different lightings. Per-pixel methods generate tens of thousands observation maps from one object. But since we put all the observation maps together, we get only 45 images after pre-processing, that is very insufficient.

Blobby & Sculpture. Blobby & Sculpture is a synthetic train dataset used in all-pixel methods, it has about 42000 shapes, but each one is rendered under 64 lightings only. With so few lightings, generated observation maps (32×32) will be sparse, thus the normal estimation will be inaccurate.

Neither of these two is perfect for us. Given that our method is a combination of per-pixel and all-pixel methods, we use them jointly to train our model, CyclesPS is used for the first 50000 iterations, then we use Blobby & Sculpture to train for another 40000 iterations.

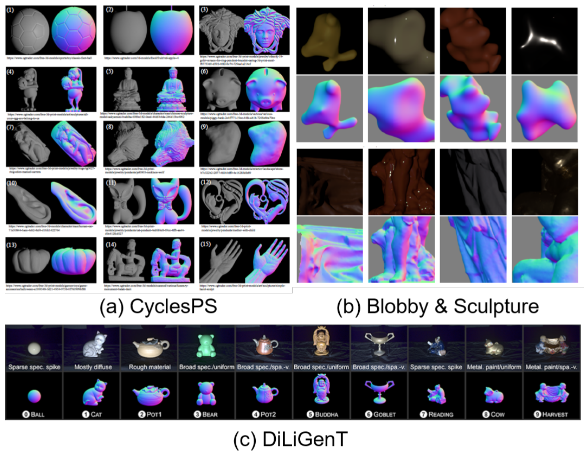

Fig. 6. Samples of the three datasets.

DiLiGenT. DiLiGenT is a real-world test dataset proposed in [7], it's the most popular benchmark for photometric stereo. It has 10 shapes with different reflectance, each is photographed under 96 different lightings.

Samples of these dataset are shown in Fig. 6.

4.2. Experiment settings

Follow settings of the other per-pixel methods, we use observation maps of resolution 32×32. And because of the limitation of GPU resources, we split all the input images into 512×512 blocks, thus the corresponding normal map block is 16×16. Emperically, we set the mini-batch size to be 8. Our network is implemented with Pytorch, and we use adam optimizer with parameters ($\beta_1 = 0.9$ and $\beta_2 = 0.999$) to

Table 1. Mean Angular Error on DiLiGenT dataset. Results of SPLINE-Net is from [5]. Ours: C means training with CyclesPS only. Ours: C + BS means training with CyclesPS, Blobby & Sculpture jointly.

Methods	BALL	BEAR	BUDDHA	CAT	COW	GOGLET	HARVEST	POT1	POT2	READING	Avg.
SPLINE-Net	5.0	6.0	10.1	7.5	8.8	10.4	19.1	8.8	11.8	16.1	10.4
Ours: C	4.4	7.3	12.3	9.4	12.0	13.8	17.6	11.5	15.6	16.3	12.0
Ours: C + BS	5.5	6.5	9.7	6.6	8.6	12.4	17.9	7.9	10.3	16.7	10.2

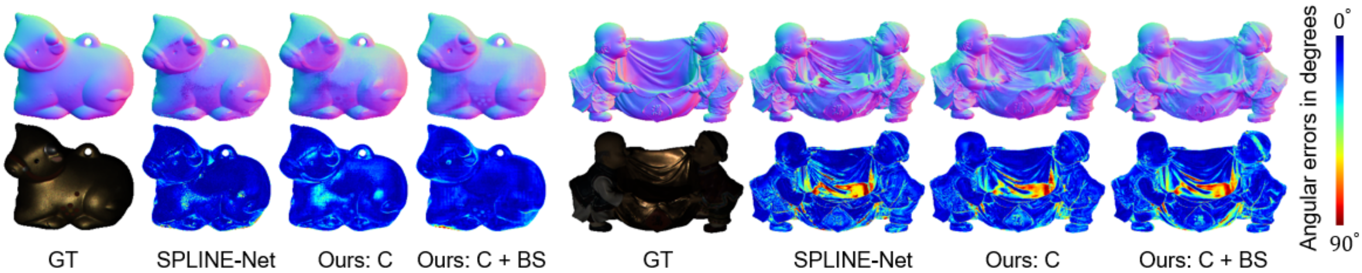

Fig. 7. Visualization of COW (left) and HARVEST (right) in DiLiGenT, comparision of reconstructed normal map (top) and error map (bottom).

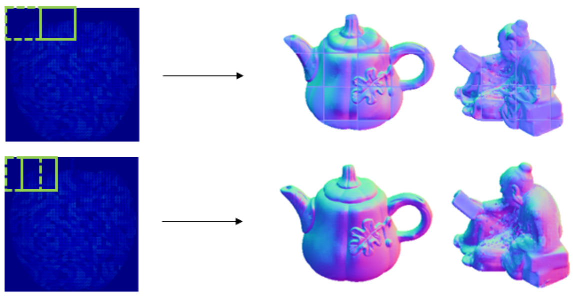

Fig. 8. Without sliding window (top), reconstructed normal maps are blocky; with sliding window (bottom), reconstructed normal maps are closer to ground truth.

optimize our network.

4.3. Test results

Similar to SPLINE-Net, our network accept observations under 10 lightings as input, but each object in DiLiGenT test dataset has 96 lightings. To fairly test the performance of our method, we generate 10 test cases, each time randomly select 10 lightings from the 96, and use the average as our final result.

As stated in Section 4.2, we split the test images into 512×512, if test these blocks seperately, the result seems blocky and has low accuracy. Thus, we come up with a sliding window strategy to average the border of the blocks, shown in Fig. 8. This strategy is verified to generate a better result.

Using sliding window and average the 10 test cases, final test results are reported in Table. 1. Metric we use is mean angular error in degree (unit omitted). Some visualization samples are shown in Fig. 7.

As we expected, when use CyclesPS only, the performance of our network is worse than SPLINE-Net, but with Blobby & Sculpture to fine tune, even the observation maps from Blobby & Sculpture are very sparse, performance of our network has been improved a lot and becomes close to SPLINE-Net.

We think the reason is per-pixel methods fit a model from observation maps (intensities) to normal vector, that requires more lightings, all-pixel methods directly extract surface normal map as a feature from observations, learning that feature requires more different shapes. Our method is a combination of these two, so it has both two requirement for the train dataset.

5. CONCLUSION

In this paper, we investigate in state-of-the-art photometric stereo methods, categorize them into all-pixel methods and per-pixel methods, and summarize their advantages and disadvantages. Base on the previous work, we propose a new method that is a combination of these two and try to fix the problems they have. Performance of our method has been verified by experiments to be close to state-of-the-art methods.

We found train datasets used in per-pixel methods and all-pixel methods have different properties. Neither of them is suit for us. Maybe the optimal way to train our model is to generate our own dataset in the future.

6. REFERENCES

[1] Satoshi Ikehata. Cnn-ps: Cnn-based photometric stereo for general non-convex surfaces. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 3–18, 2018.

[2] 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.

[3] Tatsunori Taniai and Takanori Maehara. Neural inverse rendering for general reflectance photometric stereo. *arXiv preprint arXiv:1802.10328*, 2018.

[4] Junxuan Li, Antonio Robles-Kelly, Shaodi You, and Yasuyuki Matsushita. Learning to minify photometric stereo. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 7568–7576, 2019.

[5] 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 International Conference on Computer Vision*, pages 8549–8558, 2019.

[6] 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.

[7] 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.

Individual Contributions

- Yiming Jia

  He did literature research, summarized the research results, collected all three datasets, and pre-processed the data, including generate observation maps and convert data into our input format.

- Yuchen Xing

  He implemented and modified our network from source code of SPLINE-Net. Set per parameters by repeating experiments.

- Hao Zhao

  He came up with the two strategies of testing our network, collected all the test results, visualized reconstructed surface normal map and error maps.

Reply to Critical Reviews

Critiques by group 37.

It is good to see that this presentation introduces two different methods and backgrounds of photometric stereo, and, more importantly, offers a combined solution that takes the advantages of two method and avoids the cons. The result looks pretty nice, even beats the SPLINE-Net method on some categories.

Some suggestions and improvements:

1. More information about the structure, it looks like a complex structure. How it helps to avoid the cons.
2. More information about the data extraction. How did you organize the observation map into a bigger image?

Our response:

1. By using observation maps, we don't assume uniform materials like all-pixel method, by organizing all the observation maps together, global information got maintained, intuitively global illumination effect can be somehow mitigated from neighbor's information.
2. It's really hard to state all the details in a few minutes. Basically, position of each observation map is the position of its corresponding pixel, from the figure we show, the big apple consists of observation maps.

Critiques by group 16.

This talk explained briefly the background of photometric stereo and offered a categorical view for the related literature, including the motivations, methods, pros, and cons of different work. The group 20 proposed a new method that combined the pros of previous work by organizing the observation map into a big image instead of processing pixel-wisely. Their current result is better than SPLINE-Net and they analyzed the current problems and proposed solutions as future work. The presentation is well formatted.

Some suggestions and improvements:

1. More information about the background and possible application of photometric Stereo.
2. More information about the dataset like train/test dataset size and what are the features in the original dataset.
3. What are the reasons for choosing the network structure?
4. Probably includes some more detail on training process in the presentation, like parameters and optimizers.
5. Maybe introduce the dataset before talking about the algorithm applied to help understand how the data is used by the neural network.

Our response:

1. Due to limitation of time, we put that in our paper report.
2. The most important information is the number of shapes, that can be found in our slides.
3. The most important reason is convenience, we use the same structure as SPLINE-Net with a little bit modification, it has been verified effective.
4. Our hyper parameters are either from SPLINE-Net or set experimentally.
5. Thanks for your advice.

Critiques by group 73

This project applied CNN models to reconstruct 3D images. The presentation explained the background, literature survey, datasets, model and result very clearly. We can get your motivation for this project and the novelty of this project by the presenters' logical speech. In addition, you described your neural network thoroughly with several details. The explanation of the coding part is also very clear. Your time management is extremely amazing which makes this presentation much more professional. Overall, the presentation is fluent and very detailed. The predicted results seem good as well. Very impressive project!

Some improvement:

● This presentation introduces background really well and is impressive, but it would help the listener to get your words faster if you write down some main points on slides rather than only figures. For example, in the first several slides from page 2 to 5, the text on the slide is a little bit messy. It is hard for listeners to catch the points at first glance. But if the presenter improves the composition of the slide to the same level as their content of speech, this slide would be wonderful.

● I love your presentation style of literature review, the highlight of pros and cons for each method is extremely clear and helpful. Could you simply remind the audience of the difference between all-pixel method and per-pixel method as well as common points after introducing them separately before getting into your method part? Moreover, writing down the full name of PS-FCN and CNN-PS would help the audience to understand your example.

● The input format of the proposed method seems novel. I am just a little confused about what you want to express by the rightest image on page five. Could you put in some word below the image?

● In your future work, does splitting the image into blocks face the same problem of per-pixel method that it is sensitive to global illumination effect?

● The only slight recommendation is that you can try to highlight some key words in the content of your slides, it would help listeners to read your slide along with your presentation.

Our response:

1. Thanks for your advices 1, 2, 5 on our slides, all good advices, we will take them.
2. The rightest image is just to show the format of our output is a whole surface normal map, different from SPLINE-Net, which is a single normal vector.
3. Good question! We think the answer is not, the idea is global illumination effect can be mitigated by neighbors' information, that means we can somehow find if the pixel we are looking at belong to certain shadow area from pixels nearby, we can get neighbors' information from even a small block, the whole image is not necessary. And of course, use the whole image seems more general and should be an optimal way, but we don't have enough resources to verify that.