Deep Inverse Rendering for Practical Object Appearance Scan with Uncalibrated Illumination

Jianzhao Zhang[1,3]([envelope]), Guojun Chen[2], Yue Dong[2], Jian Shi[4], Bob Zhang[5], and Enhua Wu[1,5]

[1] State Key Laboratory of Computer Science, Institute of Software, Chinese Academy of Sciences, Beijing, China
zhangjz@ios.ac.cn
[2] Microsoft Research Asia, Beijing, China
[3] University of Chinese Academy of Sciences, Beijing, China
[4] Institute of Automation, Chinese Academy of Sciences, Beijing, China
[5] University of Macau, Macao, China

Abstract. In this paper, we propose a practical method to estimate object appearance from an arbitrary number of images. We use a moving flashlight as light source, and encode surface reflectance properties in a pre-learned embedded latent space. Such lighting and appearance model combination enables our method to effectively narrow the solution space. Uncalibrated illumination requirement extremely simplifies our setup and affords it unnecessary to accurately locate light positions in advance. Moreover, our method automatically selects key frames before appearance estimation, which largely reduces calculation cost. Both synthetic and real experiments demonstrate that our method can recover object appearance accurately and conveniently.

Keywords: SVBRDF · Reflectance · Appearance · Flashlight

1 Introduction

Appearance capture is attractive but also challenging in both computer graphics and vision communities. It enables various applications in VR and AR, such as image relighting and virtual object insertion. Specially designed devices are used for accurate appearance capture [2,3,8]. Although these methods can reproduce high-resolution appearance, involved extensive scan effort prevents them from practical applications. In the past decade, consumer digital cameras have evolved a lot and it is quite convenient for non-expert users to capture high-quality images. For reflectance recovery, recent deep learning based methods learn shape, material priors from large-scale datasets, and take fewer images than traditional methods to infer appearance properties. It shows good prospects to design lightweight methods based on mobile phone cameras and deep learning technologies.

In this paper, we aim to capture object appearance from multiple input photographs. We use a neural network as an optimizer to estimate SVBRDF and normal under uncalibrated flashlight illumination. Object reflectance properties are encoded in a pre-learned latent space. Such a well-constructed latent space not only promises a reasonable SVBRDF but also provides an elegant search routine towards the final solution. During optimization, we sum reconstruction loss for each input photograph together as [7,10], and this provides flexibility about the number of input images. Experiments demonstrate that our method can recover SVBRDF from plausible to accurate with the increment of input image number.

For multiple images capture, one concern is how to take photographs efficiently. Since shooting videos of objects under a moving flashlight is quite simple, we select frames from videos as input. Generally there is a trade-off between the image number and recovery accuracy. Given the budget of the input image number, we propose to select the most valuable image collection via classic clustering. Experiments show that our strategy chooses reasonable images collection.

In summary: We propose a practical framework to estimate SVBRDF for objects with only off the shelf devices. Planar material latent space is adopted for object surface via normal decomposition. And we apply key frame selection strategy to promote algorithm efficiency.

2   Related Work

Intensive Measurement. One straightforward approach to capture appearance is brute-force measurement. Researchers design professional devices to control lighting and camera views for such purpose [15,19]. Dana *et al.* [3] used a robot arm to densely sample incident light and view directions for planar material samples. Another kind of common devices are light stages [2], they are mounted with a large number of lights and able to provide incident light from considerable directions. Linear light source reflectometry [8] is also broadly adopted for appearance recovery. Although those methods can recover vivid appearance, their dedicated devices hinder them from consumer applications.

Simplified Acquisition. In order to reduce the operating threshold for average users and simplify the acquisition process, some searchers capture appearance with hand-held commodity devices. Wu *et al.* [20] took Kinect sensors to scan object geometry and acquired illumination via a mirror sphere, then computed object appearance in an inverse rendering framework. With known geometry but unknown natural illumination, Dong *et al.* [6] estimated isotropic surface SVBRDF from a video of a rotating subject. Some methods jointly solve shape and materials with single or several images as input [1,17]. Comparing with these methods, we utilize neural networks to regularize SVBRDF in a reasonable space rather than relying on hand-crafted priors or specified heuristics.
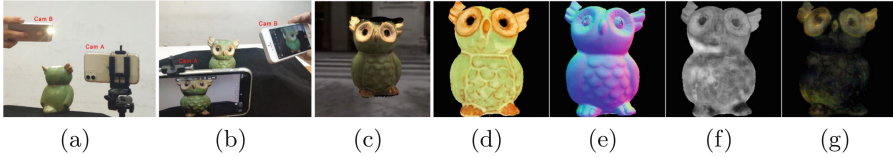
(a)   (b)   (c)   (d)   (e)   (f)   (g)

Fig. 1. (a)-(b): Capture setup. During capture, all lights in the room are turned off. Cam_A is fixed to shoot videos of target object. Cam_B serves as a light source. (c): Reconstructed object. (d)-(g): Estimated diffuse, normal, roughness and specular maps.

Deep Inverse Rendering. Li *et al.* [12] proposed a novel self-augment training scheme that effectively expanded training dataset. Deschaintre *et al.* [4] utilized in-network render layers to construct reconstruction loss and estimated reflectance properties from a flash-lit single image. Similar to [4], Li *et al.* [13] benefited from in-network render and added a dense CRF model to refine final results. Taking one image as input, these methods often fail when visible reflectance features are insufficient to distinguish ambiguities. Then novel frameworks [5,7] are proposed to infer SVBRDF from an arbitrary number of images as input. These methods focus on near-planar material sample, in contrast, ours method is able to recover object appearance.

Recently, Li *et al.* [14] proposed a learning-based method to jointly regress shape, SVBRDF and illumination from a single flash-lit image. However it may suffer from insufficient observations. Another related work is [9], they designed an asymmetric deep auto-encoder to model image formation and inverse rendering process. Extended from [9], Kang *et al.* [10] utilized learned lighting patterns to efficiently capture object appearance, and exploited diffuse and normal information from multiple views to reconstruct geometry.

3   Method

3.1   Preliminary

Our goal is to estimate object SVBRDF and normal in a single view. As showed in Fig. 1, a fixed camera is deployed to capture object-center images while a flashlight is moving. The power distribution of the flashlight is roughly concentrated in a solid angle. Therefore, we model the flashlight as a point light source as long as keep it facing the target object during the capture process. We assume that camera inner parameters are fixed, field of view fov is known, and flashlight intensity keeps constant as I_{int}. We adopt the Cook-Torrance microfacet BRDF model with the GGX normal distribution [18] and assign BRDF parameters for each point p: diffuse albedo $k_d(p)$, specular albedo $k_s(p)$ and monocular roughness $\alpha(p)$. Our method solves SVBRDF and normal $n(p)$ for target object in a fix view, with flashlight position l_i unknown for each input photograph I_i.
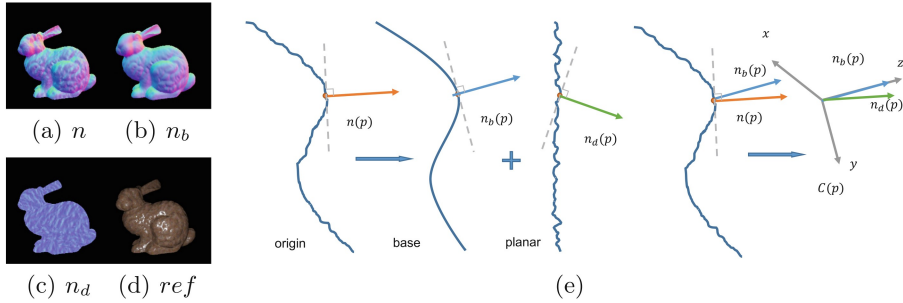
(a) n (b) n_b

(c) n_d (d) ref

(e)

Fig. 2. Decouple shape and surface material. From (a) to (d), we show normal n, base normal n_b and detail normal n_d of ref bunny. In (e): We show 2D slice of object surface. Origin object can be decomposed as base shape and planar material samples.

3.2   Decouple Shape and Surface Material

Demonstrated in [7], a pre-learned latent space can effectively model appearance of planar exemplars and benefits SVBRDF estimation. In our case, we argue that object surface material can be modeled like planar material samples too.

As showed in Fig. 2, object surface can be viewed as warped planar material samples. We decompose normal $n(p)$ as base normal $n_b(p)$ and detail normal $n_d(p)$. Base normal $n_b(p)$ relates with object shape, and detail normal $n_d(p)$ reflects material characteristics. For each point, a local coordinate system $C(p)$ can be constructed: base normal $n_b(p)$ direction is assigned as local z axis $z(p)$ and local y axis $y(p)$ is assigned to be perpendicular to direction $(1, 0, 0)$. In such a local space, detail normal $n_d(p)$ means deviation from base normal $n_b(p)$ and engraves detail variation. Therefore we get:

$$n(p) = n_b(p) \circ n_d(p) \tag{1}$$

Operation $\circ$ means transforming $n_d(p)$ from local space $C(p)$ (constructed according to $n_b(p)$) into global space. It crosses the gap between complex shape and planar material samples.

Once shape and surface material are decoupled, it is easy to convert lighting and viewing directions into the local space of each point $C(p)$. Solving object SVBRDF and detail normal $n_d(p)$ in local space is equal with planar material samples appearance recovery. We adopt local lighting model, object position is needed to calculate light direction and intensity attenuation for each point. But in fact, the distance d_i between the flashlight and object obj is much larger than the scale of object geometry variation. Thus, with known fov, a rough depth map in the camera view is enough to calculate object position.

3.3   Reflectance Recovery Under Uncalibrated Illumination

Object shape, material, and illumination jointly decide how the object looks like. If illumination is under control, SVBRDF estimation would be easier. In
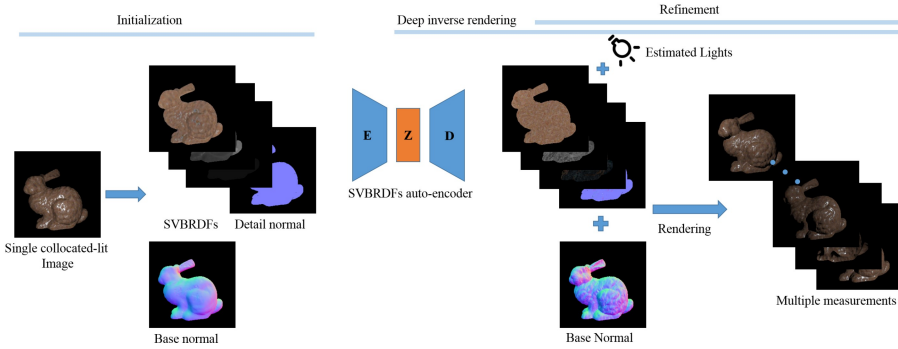
Fig. 3. Overview of the deep inverse rendering framework. We use the method [14] to estimate SVBRDF and base normal from single collocated-lit image as initialization. A specially designed SVBRDF auto-encoder is adopted for deep inverse rendering: we adjust z code to decode SVBRDF and simultaneously update base normal and point light positions. Finally, we directly refine all components in the post-processing step.

this paper, we drop fully-controlled illumination and each input photograph is lit by a flashlight with unknown position l_i. In addition, we assume flashlight intensity I_{int} is constant during the whole capture process.

As showed in Fig. 3, our method consists of three stages: initialization, deep inverse rendering and refinement. The core of our method is deep inverse rendering stage. We constrain object reflectance characteristics in a pre-trained latent space [7] and adjust the latent code z to decode reflectance parameters s:

$$s = D(z), \tag{2}$$

where $s = (n_d, k_d, \alpha, k_s)$.

We formulate the deep inverse rendering as a minimization that jointly updates the latent code z, base normal n_b and light positions $\{l_i\}$ to minimize the differences between input photograph I_i and corresponding rendering image $R(s, n_b, l_i)$:

$$\underset{z, n_b, \{l_i\}}{\arg\min} \sum_i \mathcal{L}(I_i, R(D(z), n_b, l_i)). \tag{3}$$

where we use the common loss function [4,7] as:

$$\mathcal{L}(x, y) = ||log(x + 0.01) - log(y + 0.01)||_1. \tag{4}$$

The whole pipeline of our method is as follows:

1. Use existing methods to initialize base normal, depth and SVBRDF with single collocated-lit image I_{col} as input.
2. Search for initialized light positions for each input photograph I_i.
3. Optimize latent code z, base normal n_b and light positions $\{l_i\}$ in deep inverse rendering stage.
4. Image space refinement for SVBRDF, detail normal n_d, base normal n_b and light positions $\{l_i\}$.

Reflectance Initialization. We capture a special image I_{col} whose correspondent flashlight is collocated with camera lens, as previous methods [13,14,16]. I_{col} can be used [14] to initialize our reflectance properties. The method takes I_{col} as input, and estimates object depth $\hat{d}$, SVBRDF $\hat{s} = (k_d, \alpha, k_s)$ and normal $\hat{n}$ in a cascaded network. We take their estimated normal $\hat{n}$ as base normal n_b, and initialize n_d as a flatten normal map $(n_d(p) = (0,0,1))$. In addition, given camera fov, we project depth map $\hat{d}$ into 3D coordinates as object position map Pos.

Light Initialization. Initialized SVBRDF $\hat{s}$ and normal $\hat{n}$ would help us initialize light position $\hat{l}_i$ for each photograph I_i. We use a try-and-compare strategy to search for light position candidates.

First, we define the metric to evaluate the possibility that a light position candidate lc_i can be used to initialize $\hat{l}_i$. For input image I_i, we use $\hat{s}$, $\hat{n}$ and lc_i to render image R_i, and calculate RMSE for R_i against I_i. Therefore, the goal of light initialization is to quickly find lc_i that has smaller RMSE for R_i.

Then we search for light position candidate $\acute{lc}_i$ in iterations. 1) At first, we construct a rectangular cuboid centered on the camera. During image capture, the flashlight is moving around the camera. Therefore, we set the cuboid size as $height = w_h * \bar{d}, length = w_l * \bar{d}, depth = w_d * \bar{d}$, where $\bar{d}$ is the average of depth map $\hat{d}$ and w_h, w_l, w_d are scale coefficients. We draw grids with step (w_s, l_s, d_s) in the cuboid and find the current best $\acute{lc}_i$ from all vertices. 2) Next we construct downsized cuboid centered on $\acute{lc}_i$, and draw downsized grids in the new cuboid. Similarly, we find new $\acute{lc}_i$ from all vertices. We iterate the search process until $\acute{lc}_i$ is not updated or cuboid size is below the threshold.

Finally, for each photograph I_i, $\acute{lc}_i$ is used as initialized light position $\hat{l}_i$.

Image Space Refinement. In our optimization network, latent code encodes reflectance properties in the bottle neck of auto-encoder. It usually decodes reflectance properties with details lost. Thus, we add post-process step to refine SVBRDF property maps pixel by pixel [7]. Instead of adjusting the latent code z, we directly update SVBRDF parameters k_d, α, k_s, detail normal n_d and light positions $\{l_i\}$ to minimize the differences between I_i and rendering image $R(s, n_b, l_i)$. We formulate the image space refinement as:

$$\underset{k_d, \alpha, k_s, n_d, n_b, \{l_i\}}{\arg\min} \sum_i \mathcal{L}(I_i, R(s, n_b, l_i)). \tag{5}$$

3.4   Key Frames Selection

We directly select images where the target object shows the distinctive appearance. When photographs look similar, they are possibly lit by flashlights close to each other. Thus, selecting different looking images means choosing different lighting directions. Here we rely on the classic k-means clustering method to divide all captured images into different clusters, and select centroids as picked images. Given recorded videos, our strategy can free users from tedious manual image selection and promotes efficiency.

4   Results

We implement our method in Tensorflow and take built-in layers to construct a differentiable render. For SVBRDF auto-encoder, we inherit trained model from [7]. We choose Adam [11] as optimizer, setting learning rate as 10^{-3} and β_1 as 0.5. In deep inverse rendering stage, we run $6k$ iterations; In refinement stage, we run $1k$ iterations.

At the beginning, we create synthetic datasets to validate proposed method. We randomly compose distorted elementary shapes into synthetic objects like [21] and apply texture from materials dataset [4]. In addition to composed shapes, we also select several models from the Standford 3D Scanning Repository. We render images with pre-defined point lights (used to approximate real flash-lights) in a rectangle area. To demonstrate the effectiveness of the whole method, we gradually relax the restriction from known lighting positions to unknown.

4.1   Known Lighting

For the synthetic experiments, we set camera fov as $60°$, image resolution as 256×256, and the distance between the camera and objects as $2units$. Suppose the camera center is C and the target object is at point O. All point lights are located insides the plane which is perpendicular to the line CO. In the 2×2 $unit^2$ rectangular area, we uniformly place point lights at vertices of the 5×5 grid. These point lights are used to render LDR input images I_i. For testing, we sample point light positions in the 4×4 grid, crossly among 5×5 grid. Given ground truth light positions l_i for each input photograph I_i, we take multiple images as inputs to estimate appearance properties.

To quantitatively evaluate our methods, we adopt metrics as follows: 1) RMSE(root mean square error) for estimated diffuse, specular and roughness albedos against ground truths. $s_{est} = (k_s, k_d, \alpha)$; 2) normal deviation between estimated normal n_{est} and ground truth n_{gt} in degree; 3) RMSE for rendering images $R'(n_{est}, s_{est}, \tilde{l}_i)$ under test lightings $\{\tilde{l}_i\}$. Note that SVBRDF albedos and rendering images are normalized in [0,1] to calculate RMSE.

We test on 21 objects with different materials and show average error in the Table 1. All SVBRDF property errors are smaller than initialization, and normal accuracy has been improved impressively. In general, our results are much closer to the reference than initialization. We show results of bunny in Fig. 4. For simplification, we adopt abbreviations: $diff$ for diffuse, nrm for normal, $spec$ for specular, rou for roughness, $Init$ for initialization, Opt for optimization and Ref for reference. Comparing with initialization, less highlight artifacts show in estimated diffuse albedo, and our estimated normal map contains more details.

4.2   Reflectance Recovery Under Uncalibrated Illumination

Uniformly Sampled Lights. At first, we take 25 images lit by uniformly sampled lights as input, and still test on synthetic objects mentioned in Sect. 4.1 (Without
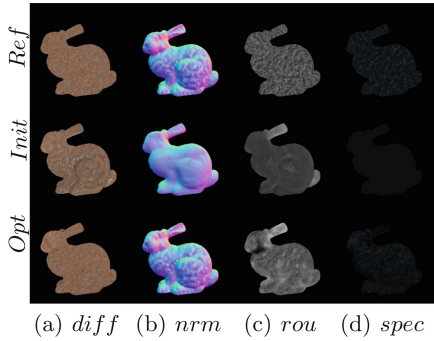
(a) *diff*   (b) *nrm*   (c) *rou*   (d) *spec*

Fig. 4. Results with known lighting.

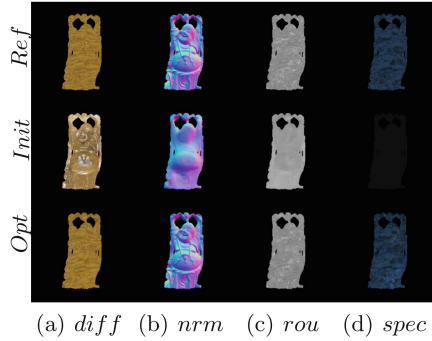
(a) *diff*   (b) *nrm*   (c) *rou*   (d) *spec*

Fig. 5. Result with unknown lighting.

Table 1. Result with 25 images. Opt_1: optimization with uniformly sampled known lighting. Opt_2: optimization with uniformly sampled unknown lighting. Opt_3: optimization with k-means selected unknown lighting.

Error	*diff*	*rou*	*spe*	*nrm*	Render	Light
Init	0.0811	0.1652	0.1096	19.47	0.0976	-
Opt_1	0.0189	0.0729	0.0706	1.2235	0.0548	-
Opt_2	0.0200	0.0716	0.0706	1.3001	0.0549	0.0398
Opt_3	0.0267	0.1048	0.0732	2.1573	0.0545	0.0448

special statement, we take such 21 objects for synthetic experiments by default). Light positions estimation will be measured in distance. We show average error in Table 1. Comparing with initialization, all refectance properties have been improved, and RMSE for rendering images is lower. At the same time, estimated light positions are close to the actual light positions. We show optimization result of buddha in Fig. 5 and light position estimation in Fig. 6. After optimization, light positions converge to ground truth dramatically.

K-means Frames Selection. In synthetic experiments, we render images with 400 uniformly sampled lights from a grid of 20 x 20 in the same rectangular area mentioned in Sect. 4.1. In Fig. 7, we show k-means clustering results for bunny. Each point represents a image lit by a flashlight. Since all lights in synthetic experiments are sampled in a rectangular planar, we take (x, y) from actual light position (x, y, z) as 2D coordinates to draw points in the figure. We observe that k-means clustering results are coincident with flashlight positions. If some flashlights are close, their correspondent images will be clustered in the same group.

Next, we select 25 images and summarize optimization results in Table 1. All SVBRDF properties, rendering images quality, and normal estimation have been improved significantly. As showed in Fig. 9, initialization method [14] cannot distinguish diffuse and specular components clearly. Their method misses normal
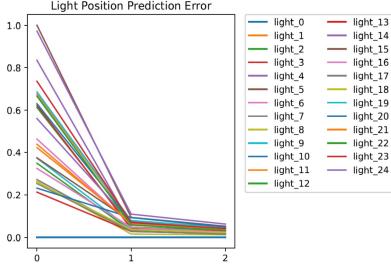
Fig. 6. Vertical: distance between predicted light position and ground truth. Horizontal: initialization, deep inverse rendering and refinement stages.
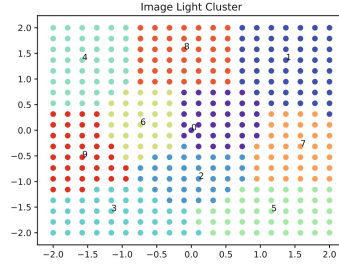
Fig. 7. Clustering result. 401 images are clustered into 10 groups. Images in groups share the same color and centroids are labeled with cluster id. (Color figure online)
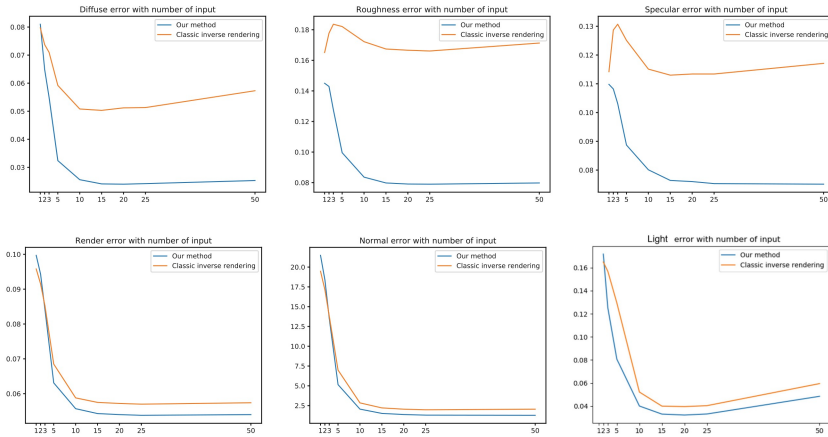

Fig. 8. Optimization result with different number of input.

details, leaving variations in diffuse albedo. In comparison, our method takes multiple images to estimate accurate SVBRDF, and rendering images under novel lighting look almost similar with ground truth.

Number of Input Images. We show how the number of input images affects optimization results. We use k-means strategy to select images with the number k ranging from 1 to 50 and show comparison in Fig. 8. In general, as the increment of input image number k, diffuse, specular and roughness estimation performance improves. Estimated normal becomes dramatically accurate with more input images. When k comes to 10, the rate of improvement slows. Another key point is 25, more images than 25 bring little benefits.

Comparison with Classic Inverse Rendering. We compare our method with classic inverse rendering that directly optimizes SVBRDF and normal in image

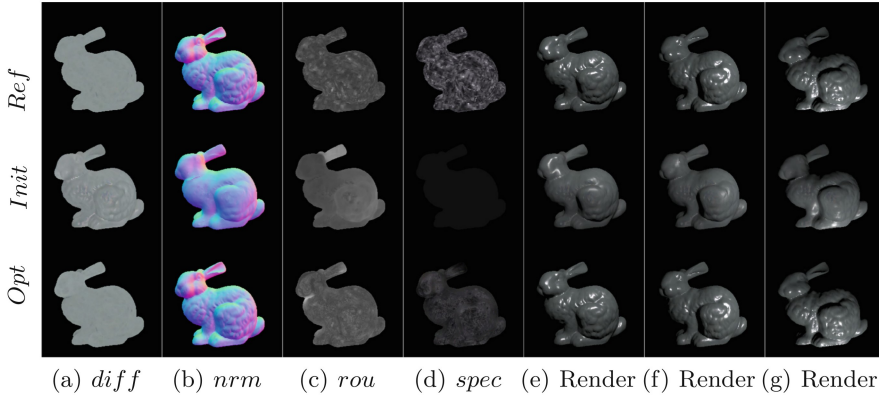(a) $diff$ (b) nrm (c) rou (d) $spec$ (e) Render (f) Render (g) Render

Fig. 9. Results for bunny with 25 selected images. Each row shows SVBRDF properties, normal map and 3 rendering images under novel lighting.

space. Similar with the post-process step introduced in Sect. 3.3, we take the differentiable render in our network to implement classic inverse rendering. Instead of updating both base normal n_b and detail n_d, classic inverse rendering directly adjusts global normal n. We formulate the classic inverse rendering as:

$$\underset{k_d,\alpha,k_s,n,\{l_i\}}{\arg\min} \sum_i \mathcal{L}(I_i, R(k_s, k_d, \alpha, n, l_i)). \qquad (6)$$

We provide the classic inverse rendering with the same initialization and run sufficient number of iterations to make sure convergence. Figure 8 shows that our method recovers more accurate SVBRDF and normal.

4.3 Real Acquisition Results

We use the ProCam app in iPhone 11 to capture all images and videos. We manually adjust ISO, white-balance, aperture and shutter speed parameters. During image capture, all camera configurations are fixed. For each object, we first shot the collocated-lit image, then turn video mode on while moving Cam_B manually around the target.

In Fig. 10, we show an example of real acquisition. Compared with initialized SVBRDF and normal, our method produces accurate diffuse and normal map with more details. Our rendering images are very close to references: highlight appears correctly and image intensity distribution is visually consistent with references. Realistic rendering images illustrate that our method can recover object appearance effectively in real scenarios.
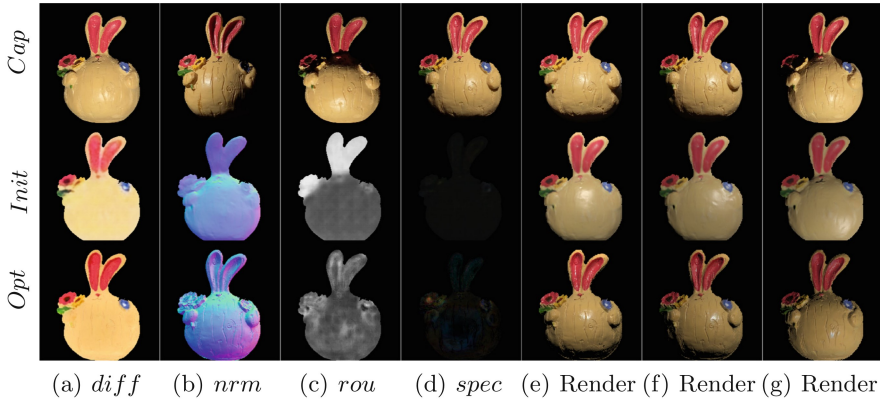
(a) *diff*   (b) *nrm*   (c) *rou*   (d) *spec*   (e) Render (f) Render (g) Render

Fig. 10. Results for real captured object with 25 selected images. First row includes captured images. Left four are part of input, and another three are reference images. In second and third rows: we display SVBRDF properties, normal map and 3 rendering images under novel lighting. Each render image column shares the same novel lighting.

5 Conclusions and Future Work

We propose a lightweight method to recover object reflectance with uncalibrated flash lighting. Modeling object surface material in a pre-learned latent space enables our method to always recover reasonable SVBRDF and constrain optimization routine to reduce ambiguity. Key frames selection strategy reduces both capture and calculation cost. Synthetic and real experiments show that our method can recover accurate SVBRDF and normal efficiently.

One limitation of our method is that we ignore inter-reflection among object components. Thus, we will add multiple bounce reflection estimation modules in the future. Currently, our method may fail if other indoor lights are not switched off. It is interesting to extend our methods under natural illumination.

References

1. Barron, J.T., Malik, J.: Shape, illumination, and reflectance from shading. IEEE Trans. Pattern Anal. Mach. Intell. **37**(8), 1670–1687 (2015)
2. Ben-Ezra, M., Wang, J., Wilburn, B., Li, X., Ma, L.: An LED-only BRDF measurement device. In: 26th IEEE Conference on Computer Vision and Pattern Recognition, CVPR (i) (2008)
3. Dana, K.J., Van Ginneken, B., Nayar, S.K., Koenderink, J.J.: Reflectance and texture of real-world surfaces. ACM Trans. Graph. **18**(1), 1–34 (1999)

4. Deschaintre, V., Aittala, M., Durand, F., Drettakis, G., Bousseau, A.: Single-image svbrdf capture with a rendering-aware deep network. ACM Trans. Graph. (SIGGRAPH Conf. Proc.) **37**(128), 1–15 (2018)

5. Deschaintre, V., Aittala, M., Durand, F., Drettakis, G., Bousseau, A.: Flexible svbrdf capture with a multi-image deep network. In: Computer Graphics Forum (Proceedings of the Eurographics Symposium on Rendering), vol. 38, no. 4, July 2019

6. Dong, Y., Chen, G., Peers, P., Zhang, J., Tong, X.: Appearance-from-motion: recovering spatially varying surface reflectance under unknown lighting. ACM Trans. Graph. **33**(6), 1–12 (2014)

7. Gao, D., Li, X., Dong, Y., Peers, P., Xu, K., Tong, X.: Deep inverse rendering for high-resolution svbrdf estimation from an arbitrary number of images. ACM Trans. Graph. **38**(4), 134:1–134:15 (2019)

8. Gardner, A., Tchou, C., Hawkins, T., Debevec, P.: Linear light source reflectometry. ACM Trans. Graph. **22**(3), 749–758 (2003)

9. Kang, K., Chen, Z., Wang, J., Zhou, K., Wu, H.: Efficient reflectance capture using an autoencoder. ACM Trans. Graph. **37**(4), 1–127 (2018)

10. Kang, K., et al.: Learning efficient illumination multiplexing for joint capture of reflectance and shape. ACM Trans. Graph. **38**(6), 1–165 (2019)

11. Kingma, D.P., Ba, J.: Adam: a method for stochastic optimization. In: Bengio, Y., LeCun, Y. (eds.) 3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, pp. 7–9. Conference Track Proceedings, May 2015

12. Li, X., Dong, Y., Peers, P., Tong, X.: Modeling surface appearance from a single photograph using self-augmented convolutional neural networks. ACM Trans. Graph. **36**(4), 45:1–45:11 (2017)

13. Li, Z., Sunkavalli, K., Chandraker, M.: Materials for masses: svbrdf acquisition with a single mobile phone image. In: Ferrari, V., Hebert, M., Sminchisescu, C., Weiss, Y. (eds.) Computer Vision - ECCV 2018, pp. 74–90. Springer International Publishing, Cham (2018)

14. Li, Z., Xu, Z., Ramamoorthi, R., Sunkavalli, K., Chandraker, M.: Learning to reconstruct shape and spatially-varying reflectance from a single image. ACM Trans. Graph. **37**(6), 1–11 (2018)

15. Matusik, W., Pfister, H., Brand, M., McMillan, L.: Efficient isotropic brdf measurement. In: Proceedings of the 14th Eurographics Workshop on Rendering. EGRW '03, Eurographics Association (2003)

16. Nam, G., Lee, J.H., Gutierrez, D., Kim, M.H.: Practical SVBRDF acquisition of 3D objects with unstructured flash photography. ACM Trans. Graph. **37**(6), 1–12 (2018)

17. Oxholm, G., Nishino, K.: Shape and reflectance estimation in the wild. IEEE Trans. Pattern Anal. Mach. Intell. **38**(2), 376–389 (2016)

18. Walter, B., Marschner, S., Li, H., Torrance, K.: Microfacet models for refraction through rough surfaces. Eurographics, pp. 195–206 (2007)

19. Ward, G.J.: Measuring and modeling anisotropic reflection. In: Proceedings of the 19th Annual Conference on Computer Graphics and Interactive Techniques. SIGGRAPH '92, Association for Computing Machinery (1992)

20. Wu, H., Zhou, K.: Appfusion: interactive appearance acquisition using a kinect sensor. Comput. Graph. Forum **34**(6), 289–298 (2015). https://doi.org/10.1111/cgf.12600

21. Xu, Z., Sunkavalli, K., Hadap, S., Ramamoorthi, R.: Deep image-based relighting from optimal sparse samples. ACM Trans. Graph. **37**(4), 1–13 (2018)