Delving High-quality SVBRDF Acquisition: a New Setup and Method

Chuhua Xian
South China University of Technology
Guangzhou China
chhxian@scut.edu.cn

Jiaxin Li
South China University of Technology
Guangzhou China

Hao Wu
Guangdong Shidi Intelligence Technology., Ltd
Guangzhou China

Zisen Lin
Guangdong Shidi Intelligence Technology., Ltd
Guangzhou China

Guiqing Li
South China University of Technology
Guangzhou China

Abstract

In this paper, we present a new and innovative framework for acquiring high-quality SVBRDF maps. Our approach addresses the limitations of current methods and proposes a new solution. The core of our method is a simple hardware setup, consisting of a consumer-level camera and LED lights, and a carefully designed network that can accurately obtain the high-quality SVBRDF properties of a nearly planar object. By capturing a flexible number of images of the object, our network uses different sub-networks to train different property maps and employs appropriate loss functions for each of them. To further enhance the quality of the maps, we also improve the network structure by adding a novel skip connection that connects the encoder and decoder with global features. Through extensive experimentation using both synthetic and real-world materials, our results demonstrate that our method outperforms previous methods and produces superior results. Furthermore, our proposed setup can also be used to acquire physically-based rendering maps of special materials.

Keywords: ***Acquisition Setup, SVBRDF Acquisition, Material Capture, Global Skip Connection.***

1. Introduction

The spatially-varying bidirectional reflectance distribution function (*abbr.* SVBRDF), modeled as a function of 6-dimensional space (light-view directions (4D) and spatial location (2D)), describes how the incident light is distributed in various exit directions after being reflected by a particular surface. Under the assumption of the Cook-Torrance BRDF model with GGX normal distribution function, which is mostly used in physical-based rendering, SVBRDFs can be parameterized using the four parameter maps: diffuse, specular, normal and glossiness. Traditional acquisition of these SVBRDF parameters tend to densely sample over the 6D space to obtain plausible results, but their procedures are low in efficiency and often limited by expensive hardwares [13, 6, 17].

Recent studies have demonstrated how deep learning can be applied to obtain SVBRDF parameters in a convenient way [18, 7, 8, 4, 11]. These studies aim to recover the reflectance properties of a material from one or several flash photographs captured by a cell phone camera. Such methods make estimations based on prior knowledge that the network has received and show that photographs of the same material captured under different illuminations may lead to contrasting results.

As a critical factor in the acquisition task, the illumination is always changed: indoor or outdoor, sunny or cloudy, noon or night, etc. Therefore, suffering from miscellaneous illuminations Fig. 1, the results of these studies could only meet the entertainment needs of ordinary users while failing to the needs of professional designers who have strict requirements on the accuracy of reconstruct the SVBRDF maps of the material. In order to delve into the relationship of the acquisition quality and lighting, it is necessary to set up a stable illumination environment. Recently, Kang [15] proposed a framework for joint acquisition of the Physicallly-Based Rendering (*abbr.* PBR) maps and the shape of a 3D model. By controlling different LEDs, the de-

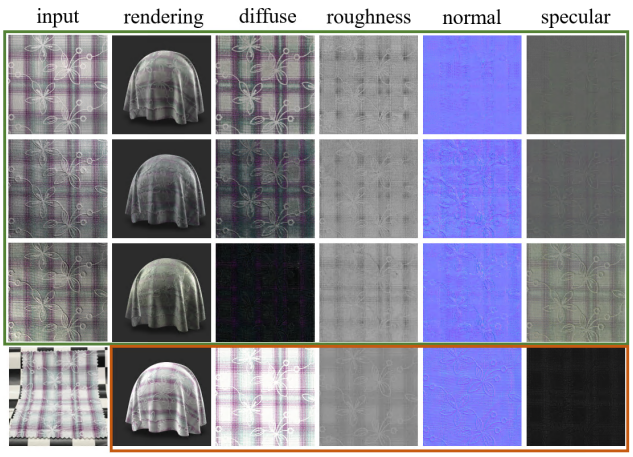

Figure 1. Examples of SVBRDF acquisitions under different illuminations. The first three rows show the results generated by the method in [11]. The 4-th row shows our result with stable illuminations. The bottom-left shows the photo of the real material captured by a SLR camera under the standard illumination of D65 light box in a dark room.

vice can generate a stable illumination environment. However, the setup of their method requires $24,576$ white LEDs and an Intel Cyclone 10 FPGA, which makes the hardware quite complex and expensive.

In this work, we propose a consumption level setup to obtain high-quality SVBRDF maps, and develop a novel network to delve into the effects of different lighting. We first design a piece of easy-to-use equipment to control stray light interference. With this setup, the photos we take are under stable illuminations. This brings a considerable advantage for the input: the testing illuminations are almost the same as the training illuminations. Thus, our network can learn the illuminations by all training samples. By taking the prior knowledge of the illuminations, our network can generate rather accurate inference result. Then, we analyzed the characteristics of different maps in the rendering function. Based on these analyses, we build our network as four independent networks to eliminate the entanglements between maps, trained with properly designed loss functions. We also propose a novel skip connection structure to learn the local and global features. Extensive experiments on both synthesis and real data have been carried out. The results show that our method works better than prior works, even at the resolution up to 3072×3072. Moreover, we delve into the acquisition quality with different numbers of the input using our proposed setup.

In summary, the main contributions of our work are as follows:

- We propose a novel simple setup for high-quality SVBRDF acquisition. Using our setup, the illuminations between training and test samples can keep well, which helps to study the relationship between acquisition quality and the number of input images.
- We design a novel skip connection that passes the global information learned from encoders to decoders. Global skip connection makes up for the shortcomings of general skip connection that can only pass local information.
- We make extensively studies on the reconstructed results with different numbers of input images. Using our proposed hardware setup, we can get up to 24 images under different illuminations. We test and analyze the effect of different number of image inputs on the reconstruction results, and give a relevant comparison.

2. Related work

Depending on the subject of interest, SVBRDF maps acquisition can be classified into two categories: nearly-plane and 3D objects. Works about the Nearly-plane objects can be further classified into single-image based methods and multi-images based methods according to the number of inputs. In this section, we will briefly review the related works of single-image based nearly-plane appearance acquisition, multi-images based nearly-plane appearance acquisition, and 3D object appearance acquisition.

2.1. Nearly-plane Objects Appearance Acquisition

Single-image based method only inputs one image to the network. Thus, the choice of photography is fairly important for the final result. It is common to choose the image that was captured under the flashlight emitted by the handheld device [18, 7, 26, 11]. Under such lighting conditions, the entire material will be illuminated, and the light and shadow information on the surface will be recorded in the photo. At the same time, the input image can be easily obtained through a mobile phone. Because of the limitation of input information, single-image based methods often show less accuracy than multi-image methods and sometimes fail to produce plausible results.

Multi-images based method requires several images which are captured in different illuminations [8, 12, 10]. It is more complicated than single-image estimation but works better in terms of accuracy. Deschaintre *et al*. [8] show that their method gets better results with the increasing of input images. In addition to deep learning methods, traditional optimization methods also benefit by adding images. Results in Gao *et al*. [10] and MaterialGan [12] also perform better with more images as optimization targets.

Optimization methods put forward high demands on users because they need to record many complicated parameters of the light and camera [12, 10, 25]. Rachel *et al*. [3] propose a method that utilizes video to estimate these parameters. Nevertheless, it requires a large amount of storage

space. In addition, deep learning methods will fail when the captured light does not match the training images.

2.2. 3D Object Appearance Acquisition

In addition to acquiring of the appearance of the nearly-plane objects, some methods have also been proposed to this purpose for 3D objects. To tackle this task, a special device should be applied, such as a camera with a specific linear polarizer [5, 9] or with the RGB LED array [19]. Holroyd *et al*. [13] design a spherical gantry equipped with a projector-camera pair on two mechanical arms, using phase-shift patterns for 3D geometry. Tunwattanaponget *et al*. [23] built a structure with an LED arm that orbits rapidly to create a continuous spherical illumination with harmonics patterns to obtain SVBRDF parameters of the object. Other similar dome structures of multiple cameras are also proposed, using structured light patterns for 3D geometry and representing reflectance as bidirectional texture functions (BTF). To get rid of the dependence of structure light, Giljoo *et al*. [20] use conventional 3D reconstruction technique, including SfM, MVS and mesh reconstruction. Xia *et al*. [24] propose to recover the 3D shape and isotropic SVBRDF parameters from a captured video sequence of a rotating object. Recently, Kang *et al*. [15] propose to build a cube-shape device light stage to capture many photos under different light fields and design a deep-learning based framework to capture both the reflectance and 3D shape of the object. However, the proposed device is quite complex, containing thousands of LEDs and complex control circuit boards. In contrast, we design a simpler device that only contains many LEDs to form the illumination environment in this work.

3. Proposed Method

3.1. Problem Overview

A spatially varying material can be well rebuilt by the pixel-level reflectance properties stored in SVBRDF maps. With the assumption of the Cook-Torrance microfacet specular shading model and GGX normal distribution function, the reflectance model used in this paper is formulated as f_r:

$$f_r(v,l,\rho,\alpha,n,F_0) = \underbrace{\frac{\rho}{\pi}}_{\mathcal{P}_d} + \underbrace{\frac{\mathcal{D}(v,l,\alpha)\mathcal{G}(v,l,n)\mathcal{F}(v,l,F_0)}{4(v\cdot n)(l\cdot n)}}_{\mathcal{P}_h}, \tag{1}$$

where v and l indicate the unit vectors of the camera and light directions; ρ, α, n, and F_0 are the spatial-varying diffuse albedo, roughness, normal and specular albedo of the material surface. f_r has two terms, with the first term being the diffuse part $\mathcal{P}_d$ and the second term being the highlight part $\mathcal{P}_h$. Our goal is to estimate ρ, α, n and F_0 from a set of images $\mathcal{I} = \{I_i\}$.

The illumination greatly influences the photo I_i. From Eq. 1, it is clear that lights and the view directions are two significant factors for an image I_i. Our observation shows that different illuminations will make a well-trained network fail and yield an erroneous result. Fig. 1 shows a failed example by highlight-aware network[11]. Because of the illumination mismatch, the diffuse map generated by their network is darker and uneven in brightness. Being highly affected by the color variance, the predicted normal diverges from reality. Our solution to these problems would have a stable illumination in the capture environment. By fixing the v and l between training samples, we expect our network to concentrate more on estimating the SVBRDF maps (ρ, α, n and F_0). Thus, our problem is simplified to estimate the SVBRDF parameters from a reflectance model modeled as:

$$f_r(\rho,\alpha,n,F_0) = \mathcal{P}_d(\rho) + \mathcal{P}_h(\mathcal{D}(\alpha)\mathcal{G}(n)\mathcal{F}(F_0)) \tag{2}$$

As an image I_i is the combined effects of lights, and all the four SVBRDF maps, multiple sets of SVBRDF maps might reach the same radiance under a special lighting condition, making it insufficient to infer an accurate map from merely a single image. Mutually complementary information contained in multiple images of the same material under different lights can be essential to alleviate the ambiguities in this problem. Experiments have been done to show how the number of input images affects the training results in Sec. 4.3. In our method, we define the number of images $|\mathcal{I}|$ to be N. Then, our task becomes to find a generator network $\mathcal{G}$:

$$\{\hat{\rho},\hat{\alpha},\hat{n},\hat{F_0}\} = \mathcal{G}(\{I_1,...,I_N\}) \tag{3}$$

Through training the network $\mathcal{G}$, we expected to find an optimal network weight θ_{opt} that minimize the loss $\mathcal{L}$:

$$\theta_{opt} = argmin_\theta \sum_{i=1} \mathcal{L}(\mathcal{G}(\{I_1^i,...,I_N^i\},\theta),\rho^i,\alpha^i,n^i,F_0^i) \tag{4}$$

3.2. Acquisition Setup

The hemispherical shell provides a fixed position for the LEDs and the camera. It also minimizes light interference from outside, ensuring that only LEDs light the material. The material stage is at the center of the hemisphere and provides a flat surface for the real material. The camera is vertically placed on the top of the hemispherical shell, facing the material stage.

The LED positions of our equipment are illustrated in Fig. 2. These LEDs are distributed at three different levels on the hemispherical shell. In a polar coordinate system originating at the center of the hemisphere, eight equidistant LEDs are installed at each level, with the angle between

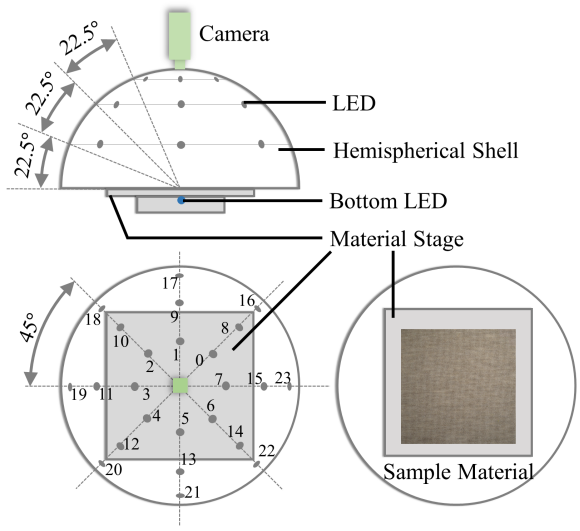

Figure 2. The positions of lighting LED.

each level being 22.5 degrees. When the system starts working, the LEDs will be lighted up in turns to create illumination in different directions. Meanwhile, the camera will capture the material on the stage when one LED is turned on. By the end of capturing procedure, we get 24 images, each lit by only one LED. Fig. 3 shows an example of the captured images of a leather material.

Under the material stage, there is a bottom LED. It will turn on to provide the blue or green light for the material stage when the material has a transparent property. When the bottom LED is in operation, it will first emit a green light, allowing the camera to capture an image of the material with a green background. After that, it will emit a blue light to capture an image with a blue background. In Sec.4.5,we explain how to determine the transparency of a material by using these two special images. The material stage scatters the light emitted by the bottom LED, making it evenly illuminate the material stage.

Prior to the acquisition, we calibrated the camera in our setup with an X-Rite ColorChecker Passport to guarantee a high color accuracy during capturing. The light intensity is also adjusted between the hardware and rendering environment with an 18% grey card. By minimizing the L1 distance between the captured photo and its corresponding rendering image, we get a scale parameter, a totally of 24 parameters. The color and light intensity calibration can further narrow the illumination gap between the training and testing dataset.

3.3. Proposed Network

Our property map generation networks leverage the classical U-net[21] as the baseline for its ability on image-to-image problems. Fig. 5 shows an overview of our acquisition method, with the top-half showing our training procedure and the bottom-half showing how we make an inference with real materials. In the training phase, pairwise training samples $(R_1, ..., R_n)$ for our supervised learning network are generated through rendering with known ground truth SVBRDF parameters (denoted as d_t, s_t, g_t, and n_t) under the same light settings as our acquisition equipment, using the reflectance model we defined in Eq. 2. When making inference on real materials, captured images of material $(I_1, ..., I_n)$ under different light positions by our acquisition device are input to the four different networks (G_d, G_s, G_g, G_n) to generate the corresponding SVBRDF maps (d_p, s_p, g_p, n_p). In the following, we will introduce the details of our network architecture and the loss functions.

3.3.1 Separated Generation Networks for Four Maps

A key distinguishing feature of our framework to other works is that we employ four independent networks to generate the four different maps separately. Many recent works have adopted the "one-to-four" architecture[11, 26] for the acquisition task of SVBDRF by having a shared encoder for extracting compact features from input images and four separate decoder branches to recovery the per-pixel diffuse albedo, specular albedo, normal and roughness from the learned features. The rationale behind such network design pattern is straightforward. Since the four maps have different emphases on different features, the synthesis of different maps requires four decoders to decode feature maps differently. However, such architecture has several drawbacks. Firstly, in our experiments, we notice that the four maps could hardly reach accurate results simultaneously. Because the four decoders share a same encoder, gradients received by the encoder are related to all four maps. Suppose a network has already learned to predict three maps out of four correctly. In that case, the non-zero gradient produced by the rest will impose changes to the encoder, indirectly affecting the correctness of other maps. Secondly, as the four decoders decode from the same feature maps, the gradients of four branches tend to reward the encoder that extracts features needed by all four branches. In summary, this network architecture leads to a high degree of entanglement in the feature space. Accordingly, we suggest using a separate network for each map, and then each network can be better at predicting a specific map. To mitigate inconsistencies in different maps, we use a render loss, calculated from the estimated map and the ground truth maps, during the training of the networks. Fig. 6 shows a comparison of maps generated by one network and four networks.

Fig. 7 visualizes the averaged feature maps over channels of the four encoders at the second downsampling layer. As shown in the figure, our diffuse network G_d tends to ex-

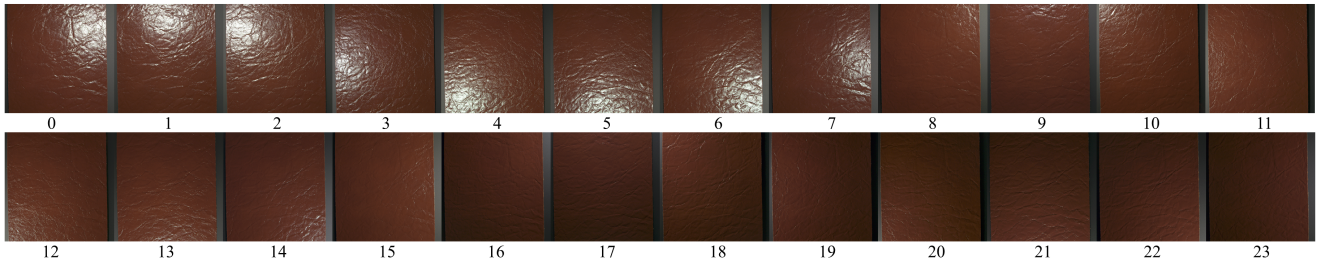

Figure 3. The images of a leather material captured by our device under the 24 LED lights.

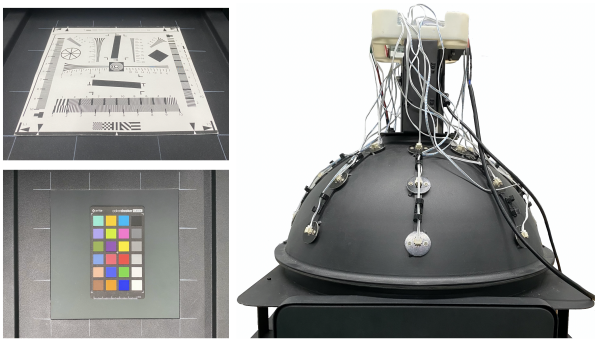

Figure 4. The appearance of our acquisition device.

tract features that follow the material pattern, eliminating the interference caused by height changes and uneven light. Similarly, features extracted by our glossiness network G_g are immune to the height differences, presenting the map in a nearly flat manner. In contrast, as the network most sensitive to the changes in height, the normal network pays more attention to information extracted from moving shadows and brightness. Unlike the other three, the specular network G_s focuses more on the micro-reflection highlights on the surface of the fabric. These results have further proven the different emphasises on features of the four maps.

The expanded details of our network architecture are shown in Fig. 8. Before being inputted to the encoder, the N captured images $I_1, ..., I_N$ are stacked as a 72-channel input. Subsequently, a single convolution layer is utilized to map the 72-channel input layer into an abstract feature map with the same resolution as $\mathcal{I}$, but with more condensed (compressed) 64 channels. Our downsampling block consists of three consecutive 3×3 convolution layers activated by LeakyRelu. We do the downsampling at the first convolution layer with a stride to be 2. At this layer, we also increase the number of feature channels by 32 and reduce the feature size to one half. We set the stride of the following two convolution layers to 1 and keep the number of feature channels and size. Symmetrical with the downsampling block, keeping the last two convolution layers the same, our upsampling block replaces the first convolution layer by a transposed convolution with a stride to 2. It also reduces the feature channels to the same number of channels as the corresponding encoder layer and doubles the size of feature maps.

To fully exploit features at different scales, our generation network downsamples the input images seven times through a sequential of 7 downsampling blocks and recoveries the SVBRDF maps in the same resolution as $\mathcal{I}$ with 7 upsampling blocks. An additional middle convolution layer (bottleneck layer) is employed between encoder and decoder to refactorize the features learned from the encoder.

Skip connections [21] are made between encoder and decoder at the same depth to preserve details at different scales. However, our experiments have revealed that skip connection via concatenation is not sufficiently capable of producing pleasant results for leaving unevenness on the generated maps. A skip connection with a global feature learning block is introduced to mitigate this problem for learning common features that span across the entire planar material. Design details and our further analysis about this structure will be explained in Sec. 3.3.2.

3.3.2 Global Skip Connection

Although a plain skip connection [21] between encoder and decoder layers through concatenation could produce generally acceptable results, our observations show that unevenness in brightness could pollute the generated maps, leaving stains on the generated maps even the material surface has evidently uniform reflectance properties (See Fig. 9). Concatenation fusion mechanism between lower-level features from encoder and high-level features from the decoder potentially produces a semantic gap [27]. Inspired by [14], we introduce a new designed global feature skip connection to U-net to tackle this issue. These connections allow the decoder to be aware of the information of other regions. In this way, some information-lack regions can infer their information through data of information-full regions.

Our global feature skip connection starts with abstracting a condensed channel-wise global feature vector from the encoder E_i at level i by using global average pooling. Subsequently, a multi-layer perceptron with one layer of hidden units activated by SeLU is leveraged to blend the condensed features at different channels before expanding back to their

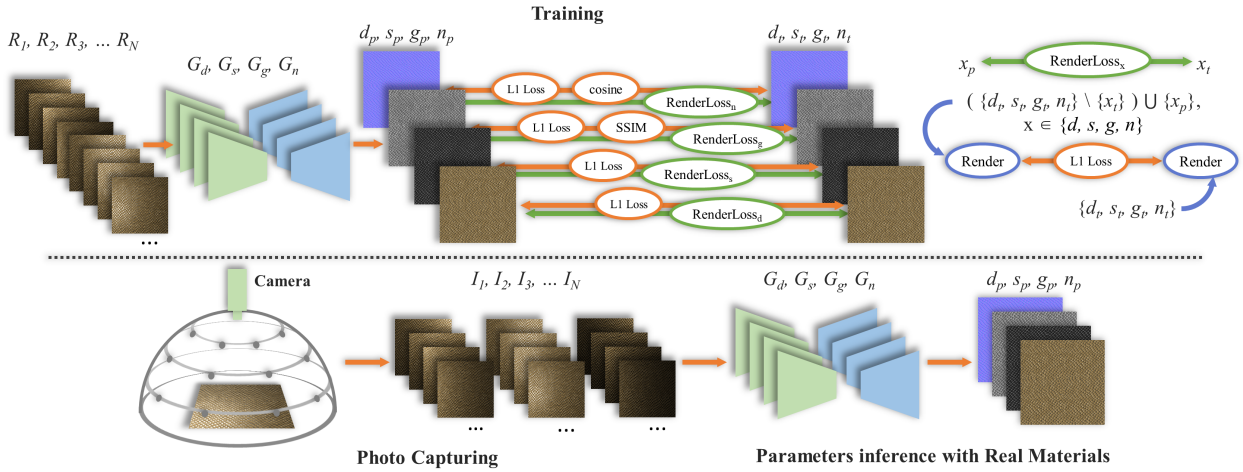

Figure 5. Overview of our proposed method.

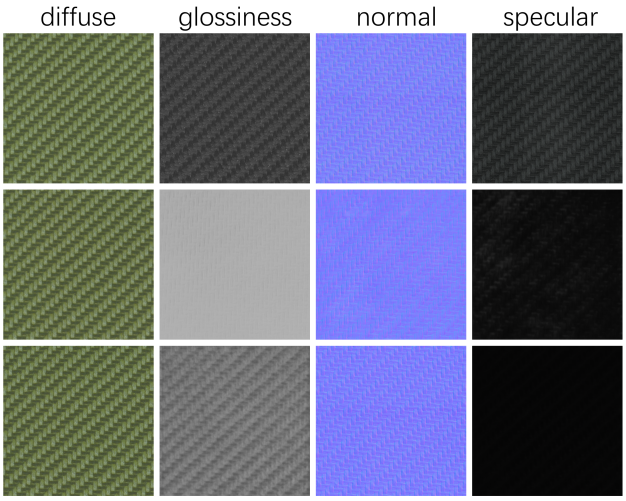

Figure 6. Comparison between one network strategy and 4 networks strategy. From top to bottom, the first row is the reference maps; the second row is the maps generated with one network strategy; the third row is the maps generated with 4 networks strategy.

Table 1. RMSE comparisons between glossiness map generated w/ and w/o global skip connections (GloSkip).

	w/o GloSkip	w/ GloSkip
RMSE	0.052626	**0.040709**

original size by a broadcast operation illustrated in Fig. 10. The final output of this module will later be fused with the corresponding layer from decoder D_i using element-wise addition. With D_i and E_i being the i_{th} layer in decoder and encoder, this process can be expressed by Eq. 5:

$$D_{i+1} = \mathcal{UP}\left(\mathcal{FC}\left(\mathcal{M}\left(E_i\right)\right) + D_i, E_i\right) \quad (5)$$

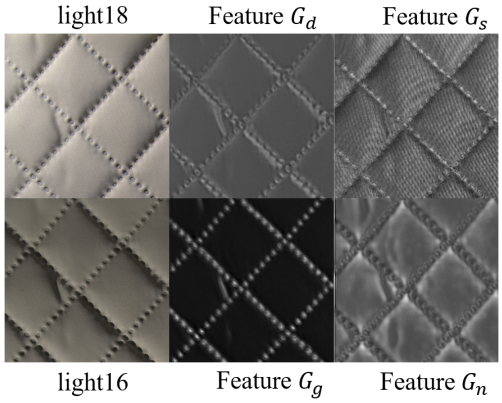

Figure 7. Feature maps generated by the four networks.

where $\mathcal{UP}(\cdot)$ indicates a upsample operation, and $\mathcal{M}(\cdot)$ represents global average pooling.

Although reaching outstanding performance in generating clear and uniform results, global skip connections are not employed to generate normal maps after careful consideration. By broadcasting an average value across the planar and enforcing such a feature to the decoder, global skip connections blur the final result, especially for normal maps, as its estimation requires high-frequency information.

3.4. Loss Function

Having four generation networks to reproduce SVBRDF maps in high quality, we carefully design specialized joint loss functions L_d, L_s,L_n and L_g respectively for G_d, G_s, G_n and G_g, depending on the different characteristics of maps they generate. Sharing some common regularizing terms in all loss functions, loss functions for all four maps consist of a map loss $\mathcal{L}_{map}$ and a rendering loss $\mathcal{L}_{render}$ calculated by averaging the mean absolute error between

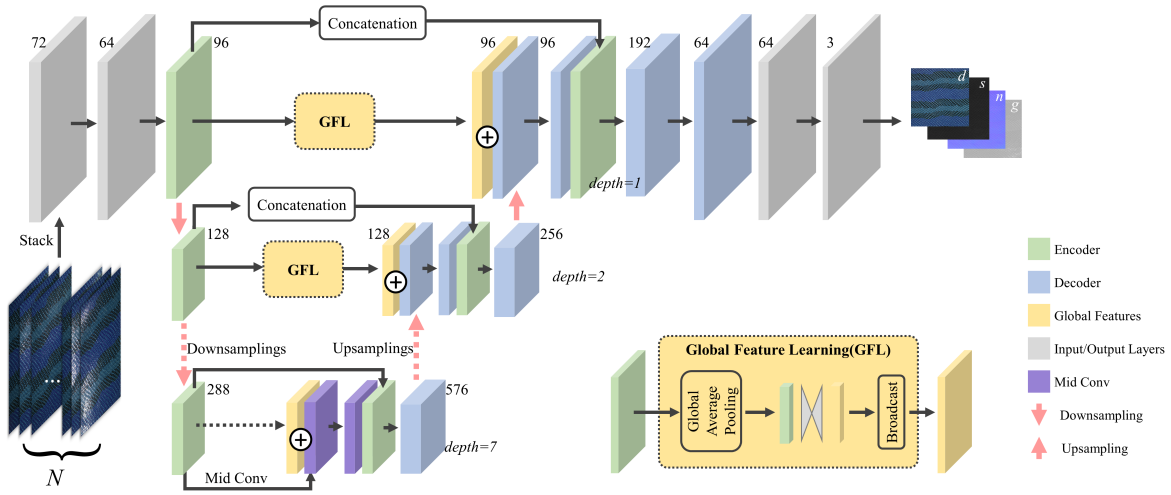

Figure 8. The architecture of our network with global feature skip connections.Four maps are generated through separate networks, and this figure only shows one of them as an example. The four networks share a similar architecture with only a slight difference at the final convolutional layers. G_d and G_s have no additional processing modules to the network structure shown in the figure. In contrast, both G_g and G_n nets undergo an extra layer of convolution with sigmoid and tanh function as active functions, respectively.

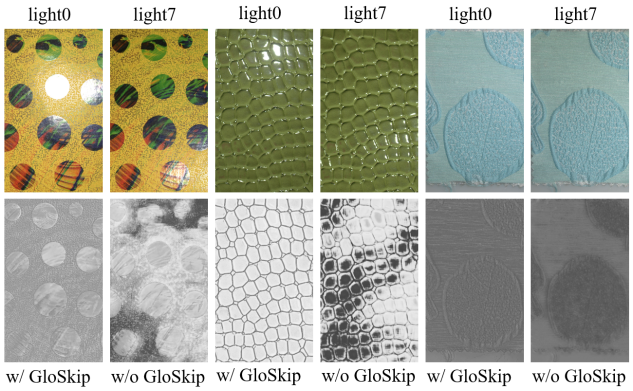

Figure 9. Three examples of the glossiness map generated w/ or w/o global skip connections (GloSkip).

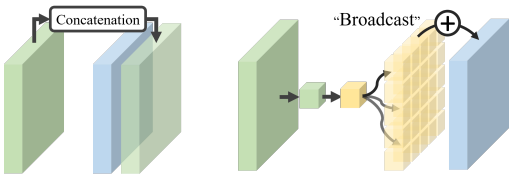

Figure 10. The comparison of the general skip connection and our global skip connection. The encoder features is first compressed to a value with unit size, and the global skip connection broadcasts it to a complete map. In a general skip connection (left), every field in the decoder can only get information in the corresponding encoder field. It only passes local information, but our global skip connection (right) broadcast the global information to every field in the decoder.

images rendered with the predicted material maps in comparison to the ground truth map using N novel lightings.

Being a slightly different with a conventional rendering loss, since the SVBRDF parameters in our method are generated separately by four networks, the rendering loss for each network G_x uses one predicted map with 3 other ground truth maps as shown in Eq. 6 where the ground truth maps $\theta = \{d_t, s_t, g_t, n_t\}$, and $x \in \{d, s, g, n\}$.

$$\mathcal{L}_{render,x} = \sum_{i=1}^{N} MAE(\mathcal{R}_{l,v}((\theta \backslash \{x_t\}) \cup \{x_p\}), \mathcal{R}_{l,v}(\theta)) \tag{6}$$

The map loss $\mathcal{L}_{map}$ in our method is computed as the $l1$ norm between predicted maps and ground truth maps using MAE. Finally, two weighted factors λ_{map} and λ_{render} are applied to the map loss and rendering loss respectively, which are set to be 1 and $1/24$ in our experiments. At this stage, we could formally defined the four joint loss functions L_d, L_s,L_n and L_g as following:

$$\mathcal{L}_d = \mathcal{L}_1(d_t, d_p) + \mathcal{L}_{render,d} + (1 - SSIM(d_p, d_t)) \tag{7}$$

$$\mathcal{L}_s = \mathcal{L}_1(s_t, s_p) + \mathcal{L}_{render,s} \tag{8}$$

$$\mathcal{L}_g = \mathcal{L}_1(g_t, g_p) + \mathcal{L}_{render,g} \tag{9}$$

$$\mathcal{L}_n = \mathcal{L}_1(n_t, n_p) + \mathcal{L}_{render,n} + \mathcal{L}_c \tag{10}$$

where x_p represents one of the predicted maps and x_t represents one of the ground truth maps, for $x \in \{d, s, g, n\}$. x_t and $SSIM(\cdot)$ is the SSIM between these two maps.

As a directional value, we use an additional cosine loss L_c to evaluate the orientation difference between predicted normal n_p and ground truth normal n_t:

$$\mathcal{L}_c = \left(-\frac{n_t}{|n_t|} \cdot \frac{n_p}{|n_p|} + 1\right). \tag{11}$$

4. Experiments

4.1. Dataset and Training

In this work, we collected 352 real materials, including cloth, leather, fabric with metallic luster or pattern, and fluorescent materials. We first generated the SVBRDF maps of these real materials using the commercial material scanner device **X-Rite TAC7 Appearance Scanner** [1], and then calibrated these maps by professional technical artists under the standard illumination in D65 light box in a dark room. We also expand the dataset by mixing many SVBRDF maps from public datasets. And finally, the new constructed dataset consists of 3184 examples. Each example contains SVBRDF maps and 24 rendered images. To get the 24 rendered images, we used 3D software to create a virtual digital twin model of our acquisition device shown in Fig. 4, then generated the 24 images using Blender Cycles [2] for each example. The resolution of the SVBRDF maps and virtual images is 512×512. In our experiments, we use 2184 for training and 1000 for test.

We implement our method using Tensorflow 2.4. For the loss optimization, we use the Adam optimizer [16]. The learning rate starts with 10^{-4}. All other hyperparameters are set as default values. When training, the batch size is set to 4 for 2000 epochs. Fig. 11 shows two results generated by our method.

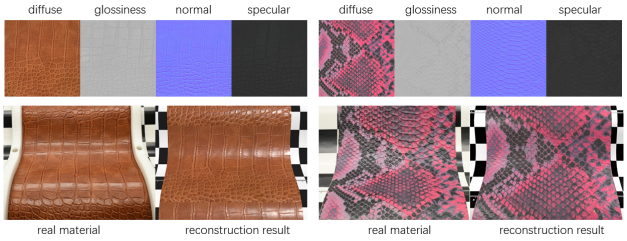

Figure 11. SVBRDF maps of two real materials generated by our method. The left-bottom is the photo of the material captured by a SLR camera under the standard illumination in D65, while the right-bottom shows the rendering result using the generated maps. The resolution of the maps is 3072×3072.

4.2. Results

We conducted our experiment using two images(No.0 and No.16) as input. Fig. 12 shows one example using our method and the methods in [8], [12], and [11]. Our method yields closer results to the ground truth, especially for the normal map, while the other methods generate wrong normal results (the normal maps of the flower shape of the cloth are wrong), which will lead to incorrect re-rendering results. We also conduct a numeric experiment on our dataset. For a fair comparison, we fine-tune these methods on our dataset. Tab. 2 lists the numeric results. Compared to other methods, our method has a significant advantage in diffuse, normal, glossiness, and rendering loss. For specular maps, our results are not as good as the method proposed in [8]. This is mainly because the materials we used are mostly fabrics, which have less prominent specular properties. Therefore, the specular maps produced by our method are not as good as those produced by the method presented in [8].

Table 2. RMSE comparisons on our dataset using two images as input. Here, d, n, s, g , and r indicate the diffuse, normal, specular, glossiness, and the rendered image, respectively.

	[8]	[12]	[11]	Ours
d	0.082861	0.006006	0.054556	**0.000306**
n	0.004437	0.005079	0.005232	**0.000791**
s	**0.007402**	0.013387	0.010090	0.046552
g	0.088811	0.132340	0.095743	**0.044373**
r	0.077482	0.009083	0.044927	**0.000602**

Table 3. RMSE comparison between previous works([8], [12], and [11]) and our method on real materials with 2 images input. The second row shows the metrics on re-renderings under 24 lights using our device, while the third row are under novel lights.

RMSE	[8]	[12]	[11]	Ours
24 lights	0.066352	0.017784	0.068793	**0.013180**
Novel lights	0.196887	0.087135	0.202598	**0.053856**

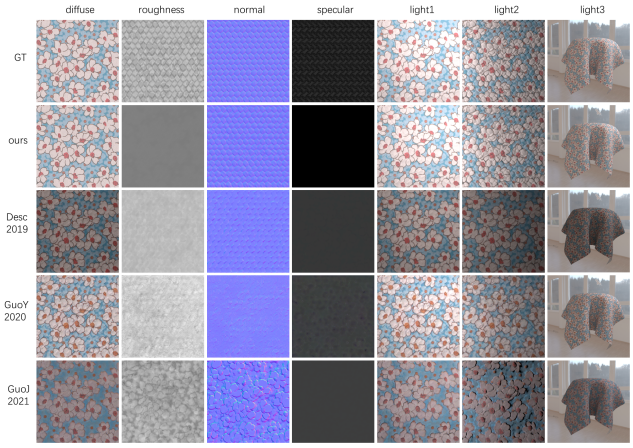

Figure 12. The reconstruction results using two images as input. Under the side lighting conditions, our rendering results can clearly see the shadow texture generated by the surface bump. [8] and [12] have only blurred dim or almost invisible surface shadows.

As pointed out in [8] and MaterialGan [12], the recovered SVBRDF maps will become better with the increasing number of the inputs. We also conduct experiments and validate this conclusion on our hardware setup using our proposed method. We use the photos captured by our device

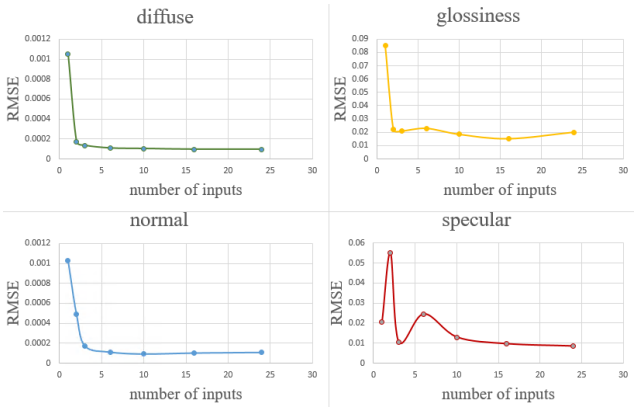

Figure 13. RMSE in different number of input.

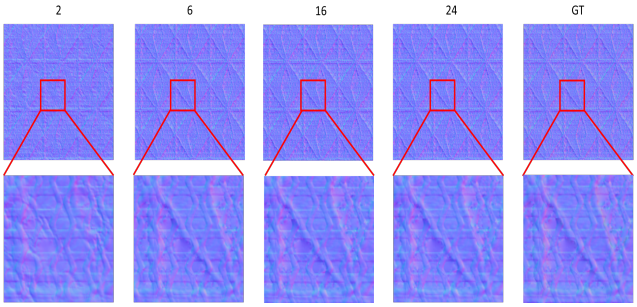

Figure 14. Comparisons of the normal maps with different numbers of input images.

Table 4. The image numbers we input in the experiment.

input	No.
1	0
2	0, 16
3	0, 16, 23
6	0, 4, 8, 12, 16, 20
10	4, 6, 8, 10, 12, 14, 16, 18, 20, 22
16	0, 2, 4, 5, 6, 8, 10, 11, 12, 14, 16, 17, 18, 20, 22, 23

and respectively train our network on the training dataset using $24, 16, 10, 6, 3, 2, 1$ photos as input (the numbers of the selected photos are listed in Tab. 4). The RMSEs of diffuse, glossiness, normal, and specular maps are computed on our test dataset. The hyperparameters used for training are the same for all different inputs. The numeric results are drawed as line graphs shown in Fig. 13. From these four line graphs, it can be seen that the RMSE decreases as input images increase. For the specular maps, the curve oscillates when the input number is small, but it tends to decline steadily when the number increases. Diffuse, normal, and glossiness maps get a significant improvement at the beginning. When the input number rises to 6, the decline of RMSE slows down. Thus, if considering the qualities of the maps only, one can train the networks with as many inputs as possible. And the more images are input, the more details of the result closer to the ground truth. Fig. 14 shows the comparison of the normal maps using different number images as input, and we can see that the details can be obtained well when the number is bigger than 6. Thus, if considering the training cost / quality ratio, selecting 6 images as input is a better choice.

4.3. Comparisons with More Images as Input

We compare our method with the methods in [8], MaterialGan [12], and [11]. Because the input of [8] or MaterialGan [12] are multiple images, for fair comparisons, we directly utilize the 24 rendered images (or the images cropped from the photos captured by our device) as input. For the method in [11], we traverse the results of the 24 images (or the cropped photos captured by our device for real materials) and choose the best one for comparison. We conduct qualitative and quantitative experiments to evaluate our method on our dataset and real materials.

Comparisons on our dataset. Fig. 15 shows an example of the comparisons in our dataset. The diffuse maps acquitted by [8] and [11] are darker than the ground truth, which will cause darker results for re-rendering. In contrast, the diffuse maps acquitted by our method and MaterialGan [12] are compatible with the ground truth. For the normal maps, our result contains more details and is closest to the ground truth, while the results acquitted by [11] and [12] tend to be flatter. For the result acquitted by [8], the direction of the edge changes more intensely.

Note that MaterialGan [12] needs to record precise parameters of the camera and illumination, making obtaining SVBRDF maps become more complicated. In contrast, our method does not require complex parameters because the illuminations of our input images is under fixed control. Our network can learn the stable illuminations between training samples and use the learned parameters for inference. Thus, the quality of the maps can be guaranteed by the stable illuminations provided by our device, and it does not need additional optimization.

Tab. 5 and Tab. 6 show the numerical comparisons on our dataset. The numeric results demonstrate that our method acquits the best results in diffuse, normal, and glossiness maps, while the method of Guo Jie *et al*. [11] has the lowest RMSE and Guo Yu *et al*. [12] has the lowest LPIPS in specular maps. Although our method does not get the best result of specular maps, it obtains the best diffuse and normal maps, which play more important effects for re-rendering.

Comparisons on real materials. We validate our method on 85 real materials which are not in our dataset. The input photos are captured using our device. We use 3 novel lights to evaluate the results, and capture the photos of real materials using a SLR camera. Then, we render the materials with the recovered SVBRDF maps in the digital twin

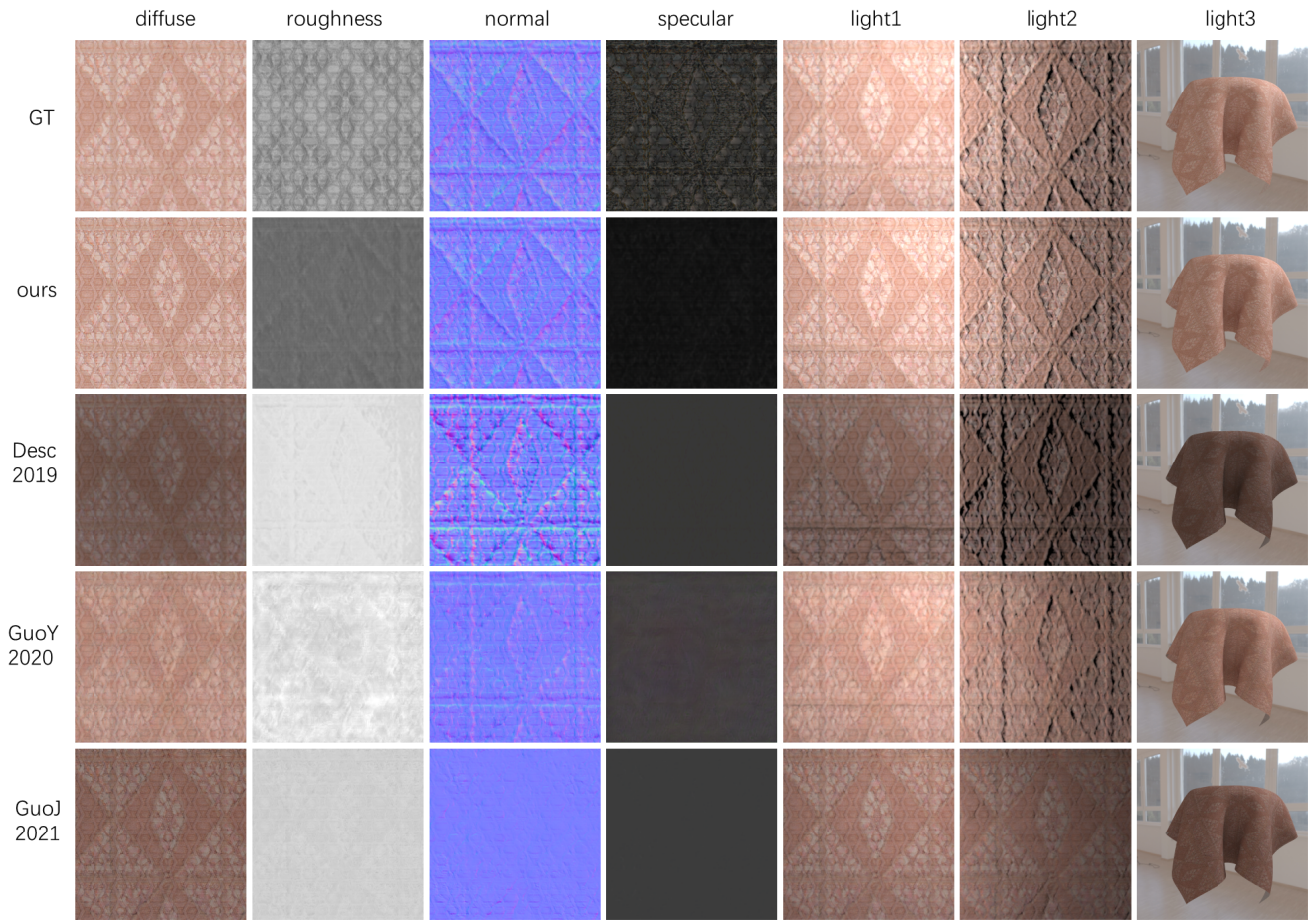

Figure 15. An example of comparison with the methods in [8], [12], and [11]. The diffuse maps, specular maps, and the rendering results are shown in Gamma space, while the glossiness maps are turned to roughness maps for more clearly visualization.

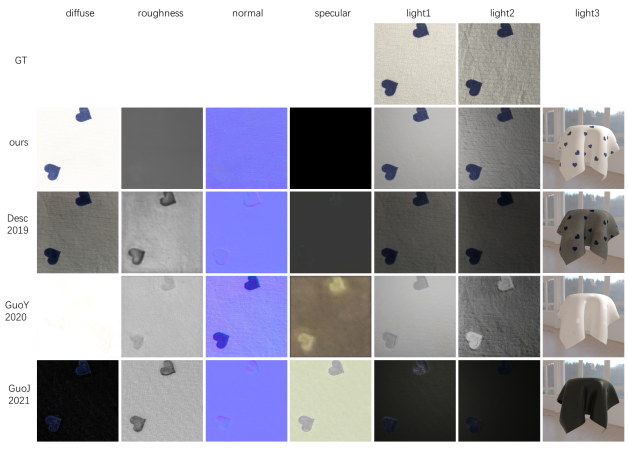

Figure 16. An example of a real material.

illuminations. Fig. 16 shows an example of our results and the comparisons. It shows that the normals generated by the methods in [12] and [11] are wrong. Actually, the re-

Table 5. RMSE comparisons on our dataset. d, n, s, g , and r indicate the diffuse, normal, specular, glossiness, and the rendering image, respectively.

	[8]	[12]	[11]	Ours
d	0.102023	0.005124	0.054556	**0.000423**
n	0.006479	0.004360	0.005232	**0.000247**
s	0.033632	0.012154	**0.010090**	0.025878
g	0.102216	0.140183	0.095743	**0.040709**
r	0.087551	0.006833	0.044927	**0.000301**

gion of the heart shapes are flat, but the normals recovered by these two methods are concave. Besides, the recovered diffuse maps do not contain the heart shape pattern, which means that these two methods cannot distinguish the color and shadow information for planar exemplar materials. For the maps generated by [8], the recovered diffuse map is gray while the color of the material is white. Thus, the re-rendering results are very different from captured pho-

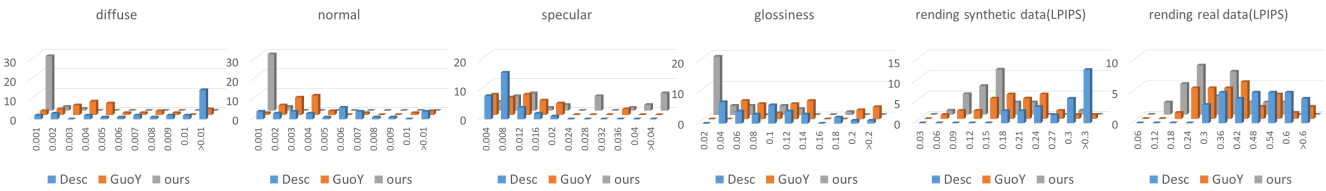

Figure 17. Statistics from 31 synthetic examples and 31 real materials. We compute the Learned Perceptual Image Patch Similarity(LPIPS) on the re-rendering images and the Root Mean Square Error(RMSE) on SVBRDF maps. In the metrics, a lower value indicates a higher accuracy. Our outputs are more concentrated in the areas with lower values which means that we get more accurate results on most examples.

Table 6. LPIPS comparisons on our dataset. d, n, s, g , and r indicate the diffuse, normal, specular, glossiness, and the rendering image, respectively.

	[8]	[12]	[11]	Ours
d	0.306619	0.133610	0.286742	**0.063140**
n	0.205903	0.246912	0.364797	**0.151345**
s	0.716021	**0.692563**	0.747934	0.721130
g	0.656656	0.579437	0.519371	**0.495239**
r	0.299803	0.179935	0.318218	**0.130414**

tos. Tab. 7 lists the RMSE comparisons of re-rendering results with captured photos between previous work and our method for 85 real materials. Fig. 17 shows the numerical details. Since GuoJie *et al.*'s method [11] is based on a single image, we did not compare the performance with it in statistics. It demonstrates that our method can achieve the best results on SVBRDF acquisitions for real materials.

Performance. We evaluate the runtime performance on a PC with 3.0 GHz Intel Core i7 processor and NVIDIA GeForce RTX 3090 GPU. For the input image of the resolution 512×512, it takes around 0.11s using the method in [11] because it only takes one image as input. For the input from 2 to 24 images, our method takes between 0.300s and 0.480s, while it takes about 2.93s using the method in [8] . In comparison, MaterialGan [12] requires around 660s with the same input on the same platform because of its lengthy optimization.

4.4. Ablation Study

As discussed in Sec.3.3.1, gradients received by the encoder in a one-to-four architecture are related to all four decoders. Gradients from the other three maps could affect the correctness of the one that has been correct. We do ablations studies on how the one encoder architecture performs to validate it. In addition, we compared the performance of our global skip connection and global track[7] to prove the superiority of our method. The result is shown in the Table.8.

Table 7. RMSE and LPIPS comparisons between previous works([8], [12], and [11]) and our method on real materials. The second row shows the metrics on re-renderings under 24 lights using our device, while the third row are under novel lights.

RMSE	[8]	[12]	[11]	Ours
24 lights	0.069047	0.013157	0.068793	**0.012600**
Novel lights	0.204769	0.072937	0.202598	**0.067823**
LPIPS	[8]	[12]	[11]	Ours
24 lights	0.470221	0.377422	0.498515	**0.284123**
Novel lights	0.630545	0.516387	0.611689	**0.422746**

Table 8. RMSE comparisons of the ablation study.

	1 encoder	Global track	Ours
d	0.017557	0.021936	0.011774
s	0.019679	0.022555	0.018615
n	0.023288	0.036720	0.020182
g	0.037449	0.052994	0.032916

4.5. Acquisition of Special Materials

In addition to leather and fabric, our simple hardware setup can also be used to acquit the PBR maps of some special materials, such as mesh, metallic, and fluorescent materials, as shown in Fig. 18 to 21.

In order to simulate the hollow of the mesh, an alpha map α is needed. As illustrated in Fig. 2, our hardware has a bottom LED light. Before capturing transparent material, the light stage emits blue and green light respectively and camera takes the background images B_b and G_b. The images with the material B_c (with blue light) and G_c (with green light) is taken when capturing. According to Alvys' method[22], we have:

$$\begin{cases} B_c = \alpha F + (1-\alpha)B_b, \\ G_c = \alpha F + (1-\alpha)G_b, \end{cases} \tag{12}$$

where F is the color of the object. Since B_b, G_b, B_c and G_c are known, by solving Eq. 12, α can be obtained. Fig. 18 and Fig. 21 show the reconstructed alpha maps using our

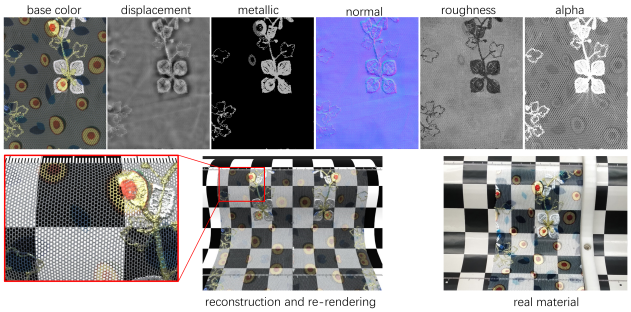

Figure 18. Reconstruction of a mesh material with metallic patterns. The first row shows the maps obtained using our device and method.

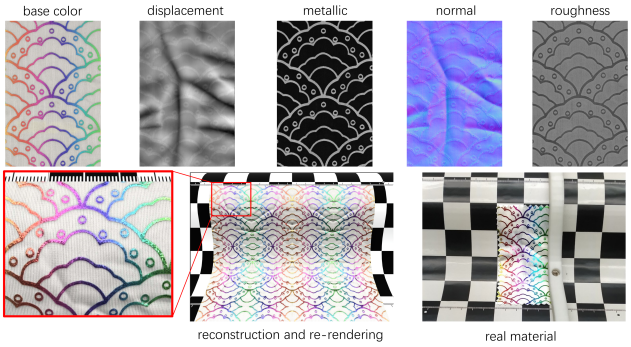

Figure 19. Reconstruction of a fabric material with metallic patterns. To better express the metallic luster of materials, the workflow for reconstructing such materials employs the metallic workflow.

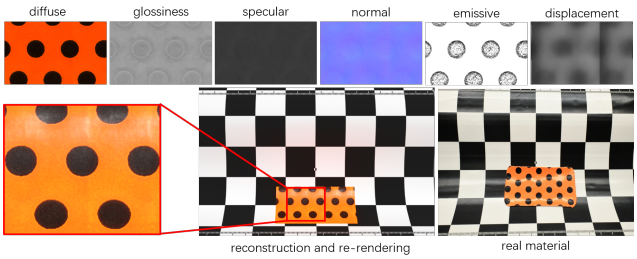

Figure 20. Reconstruction of a fluorescent material.

method.

For the materials with metallic luster or pattern, we train the networks using the same proposed method with the metallic workflow (render by the base color, metallic, normal, and roughness maps). Fig. 18 and 19 show two examples of the reconstructed metallic maps. For fluorescent materials, the emissive map can be also obtained in a similar way. Two examples of the reconstructed fluorescent materials are shown in Fig. 20 and 21. It should point out that the displacement map is converted from the normal map in the reconstruction.

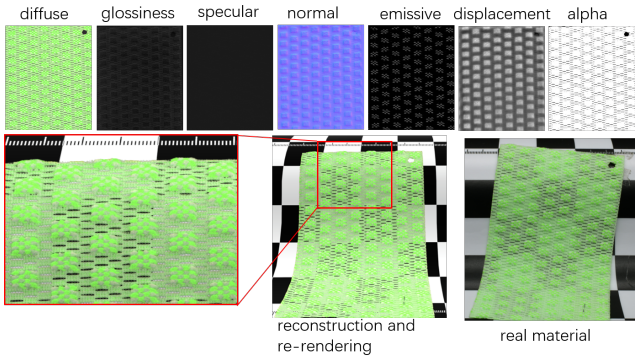

Figure 21. Reconstruction of a mesh-fluorescent fabric.

4.6. Compare with Handheld Devices

Compared to handheld devices like smartphones, our setup is capable of achieving better results, albeit with a slightly more complex setup. By utilizing our setup, we can generate alpha maps of materials by controlling the bottom LED, which is not possible with smartphones. Fig. 22 showcases the difference in the maps generated using photos taken with our setup versus those captured by a handheld device. The left image features a mesh fabric, with the second row demonstrating maps and renderings captured by a smartphone. Without our device, calculating alpha maps is impossible, and the background color cannot be seen through the hole in the mesh.

Furthermore, it is challenging to ensure consistency in the generated maps across different illumination conditions for the same material on handheld devices when using deep learning methods. This difficulty arises because it is difficult to guarantee that the illumination conditions in photos captured by handheld devices will be consistent with those in the training dataset. As demonstrated in Fig. 1, inconsistent maps result from different illuminations when capturing photos, using the deep learning method described in [11], which relies on handheld devices. Additionally, as shown on the right of Fig. 22, our networks exhibit color bias without the environment control provided by our setup.

5. Conclusions

In this work, we propose a novel setup and network to obtain high-quality SVBRDF maps. We have highlighted the importance of stable lighting patterns for deep learning based methods, and delved into studying the relationship of acquisition quality of different number images as input. We also described the necessity of separating the generation network for each map. Then, we have shown that our naive global skip connection can pass the global and local information between decoder and encoder. We also explore the effects of the input image number experimentally. Our results show that our method outperforms existing methods

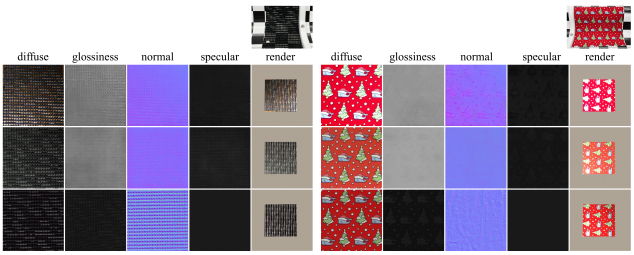

Figure 22. The top images in each example are the reference fabric captured using DLR under the D65 light box. The second row shows the maps and re-rendered images generated using a mobile phone. The third row displays the generated results using our setup with 2 images, while the fourth row shows the results generated with 24 images captured by our setup.

in performance both on our dataset and real materials. And our proposed method can also reconstruct the PBR maps for special materials, such as mesh, metallic, and fluorescent materials. We believe that high-quality PBR maps of more types of materials can be acquitted efficiently using our proposed hardware setup.

Acknowledgement

This paper is supported by the Nature Science Fund of Guangdong Province (No.2021A1515011849) and the Key-Area Research and Development of Guangdong Province (2022A0505050014).

References

[1] X-rite: Tac7 appearance scanner. https://www.xrite.com/categories/appearance/total-appearance-capture-ecosystem/tac7, 2017. 8

[2] Cycles: Open source production rendering. https://www.cycles-renderer.org/, 2022. 8

[3] R. A. Albert, D. Y. Chan, D. B. Goldman, and J. F. O'Brien. Approximate svbrdf estimation from mobile phone video. In *Proceedings of the Eurographics Symposium on Rendering: Experimental Ideas & Implementations*, pages 11–22, 2018. 2

[4] L.-P. Asselin, D. Laurendeau, and J.-F. Lalonde. Deep svbrdf estimation on real materials. In *2020 International Conference on 3D Vision (3DV)*, pages 1157–1166, 2020. 1

[5] S.-H. Baek, D. S. Jeon, X. Tong, and M. H. Kim. Simultaneous acquisition of polarimetric svbrdf and normals. *ACM Trans. Graph.*, 37(6):268–1, 2018. 3

[6] K. J. Dana, B. Van Ginneken, S. K. Nayar, and J. J. Koenderink. Reflectance and texture of real-world surfaces. *ACM Transactions On Graphics (TOG)*, 18(1):1–34, 1999. 1

[7] V. Deschaintre, M. Aittala, F. Durand, G. Drettakis, and A. Bousseau. Single-image svbrdf capture with a rendering-aware deep network. *ACM Transactions on Graphics (ToG)*, 37(4):1–15, 2018. 1, 2, 11

[8] V. Deschaintre, M. Aittala, F. Durand, G. Drettakis, and A. Bousseau. Flexible svbrdf capture with a multi-image deep network. In *Computer Graphics Forum*, volume 38, pages 1–13, 2019. 1, 2, 8, 9, 10, 11

[9] V. Deschaintre, Y. Lin, and A. Ghosh. Deep polarization imaging for 3d shape and svbrdf acquisition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 15567–15576, 2021. 3

[10] D. Gao, X. Li, Y. Dong, P. Peers, K. Xu, and X. Tong. Deep inverse rendering for high-resolution svbrdf estimation from an arbitrary number of images. *ACM Transactions on Graphics (TOG)*, 38(4):1–15, 2019. 2

[11] J. Guo, S. Lai, C. Tao, Y. Cai, L. Wang, Y. Guo, and L.-Q. Yan. Highlight-aware two-stream network for single-image svbrdf acquisition. *ACM Transactions on Graphics (TOG)*, 40(4):1–14, 2021. 1, 2, 3, 4, 8, 9, 10, 11, 12

[12] Y. Guo, C. Smith, M. Hašan, K. Sunkavalli, and S. Zhao. Materialgan: reflectance capture using a generative svbrdf model. *ACM Transactions on Graphics (TOG)*, 39(6):1–13, 2020. 2, 8, 9, 10, 11

[13] M. Holroyd, J. Lawrence, and T. Zickler. A coaxial optical scanner for synchronous acquisition of 3d geometry and surface reflectance. *ACM Transactions on Graphics (TOG)*, 29(4):1–12, 2010. 1, 3

[14] J. Hu, L. Shen, and G. Sun. Squeeze-and-excitation networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7132–7141, 2018. 5

[15] K. Kang, C. Xie, C. He, M. Yi, M. Gu, Z. Chen, K. Zhou, and H. Wu. Learning efficient illumination multiplexing for joint capture of reflectance and shape. *ACM Trans. Graph.*, 38(6):165–1, 2019. 1, 3

[16] D. P. Kingma and J. Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014. 8

[17] J. Lawrence, A. Ben-Artzi, C. DeCoro, W. Matusik, H. Pfister, R. Ramamoorthi, and S. Rusinkiewicz. Inverse shade trees for non-parametric material representation and editing. *ACM Transactions on Graphics (TOG)*, 25(3):735–745, 2006. 1

[18] Z. Li, K. Sunkavalli, and M. Chandraker. Materials for masses: Svbrdf acquisition with a single mobile phone image. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 72–87, 2018. 1, 2

[19] X. Ma, K. Kang, R. Zhu, H. Wu, and K. Zhou. Free-form scanning of non-planar appearance with neural trace photography. *ACM Transactions on Graphics (TOG)*, 40(4):1–13, 2021. 3

[20] G. Nam, J. H. Lee, D. Gutierrez, and M. H. Kim. Practical svbrdf acquisition of 3d objects with unstructured flash photography. *ACM Transactions on Graphics (TOG)*, 37(6):1–12, 2018. 3

[21] O. Ronneberger, P. Fischer, and T. Brox. U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical image computing and computer-assisted intervention*, pages 234–241, 2015. 4, 5

[22] A. R. Smith and J. F. Blinn. Blue screen matting. In *Proceedings of the 23rd annual conference on Computer graphics and interactive techniques*, pages 259–268, 1996. 11

[23] B. Tunwattanapong, G. Fyffe, P. Graham, J. Busch, X. Yu, A. Ghosh, and P. Debevec. Acquiring reflectance and shape

from continuous spherical harmonic illumination. *ACM Transactions on graphics (TOG)*, 32(4):1–12, 2013. 3

[24] R. Xia, Y. Dong, P. Peers, and X. Tong. Recovering shape and spatially-varying surface reflectance under unknown illumination. *ACM Transactions on Graphics (TOG)*, 35(6):1–12, 2016. 3

[25] W. Ye, Y. Dong, P. Peers, and B. Guo. Deep reflectance scanning: Recovering spatially-varying material appearance from a flash-lit video sequence. In *Computer Graphics Forum*, volume 40, pages 409–427, 2021. 2

[26] X. Zhou and N. K. Kalantari. Adversarial single-image svbrdf estimation with hybrid training. In *Computer Graphics Forum*, volume 40, pages 315–325, 2021. 2, 4

[27] Z. Zhou, M. M. R. Siddiquee, N. Tajbakhsh, and J. Liang. Unet++: A nested u-net architecture for medical image segmentation. pages 3–11, 2018. 5