Computational Visual Media
https://doi.org/10.1007/s41095-023-0352-6

Research Article

Delving into high-quality SVBRDF acquisition: A new setup and method

Chuhua Xian[1] (✉), **Jiaxin Li**[1], **Hao Wu**[2], **Zisen Lin**[2], **and Guiqing Li**[1]

Abstract In this study, we present a new and innovative framework for acquiring high-quality SVBRDF maps. Our approach addresses the limitations of the current methods and proposes a new solution. The core of our method is a simple hardware setup consisting of a consumer-level camera, 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 an object, our network uses different subnetworks to train different property maps and employs appropriate loss functions for each of them. To further enhance the quality of the maps, we improved 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 (SVBRDF), modeled as a function of 6-dimensional space (light-view directions (4D) and spatial location (2D)), describes how 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 a GGX normal distribution function, which is mostly used in physical-based rendering, SVBRDFs can be parameterized using four parameter maps: diffuse, specular, normal, and glossiness. The traditional acquisition of these SVBRDF parameters tends to be densely sampled over a 6D space to obtain plausible results, but their procedures are inefficient and often limited by expensive hardware [1–3].

Recent studies have demonstrated how deep learning can be conveniently applied to obtain SVBRDF parameters [4–10]. These studies aimed to recover the reflectance properties of a material from one or more flash photographs captured using 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. Figure 1 shows reconstruction using the method of Guo et al. [8] under different illuminations.

As a critical factor in the acquisition task, illumination always changes: indoor or outdoor, sunny or cloudy, noon, night, etc. Therefore, owing to miscellaneous illumination, the results of these studies could only meet the entertainment needs of ordinary users while failing to meet the needs of professional designers who have strict requirements for the accuracy of reconstructing the SVBRDF maps of the material. To delve into the relationship between acquisition quality and lighting, it is necessary to establish a stable illumination environment. Recently, Kang et al. [11] proposed a framework for the joint acquisition of physicallly based rendering (PBR)

1 School of Computer Science and Engineering, South China University of Technology, Guangzhou 510006, China. E-mail: C. Xian, chhxian@scut.edu.cn (✉); J. Li, 202020143933@mail.scut.edu.cn; G. Li, ligq@scut.edu.cn.

2 Guangdong Shidi Intelligence Technology., Ltd., Guangzhou 510000, China. E-mail: H. Wu, wuh@4dstc.com; Z. Lin, antonio@4dstc.com.

Manuscript received: 2023-03-02; accepted: 2023-04-20

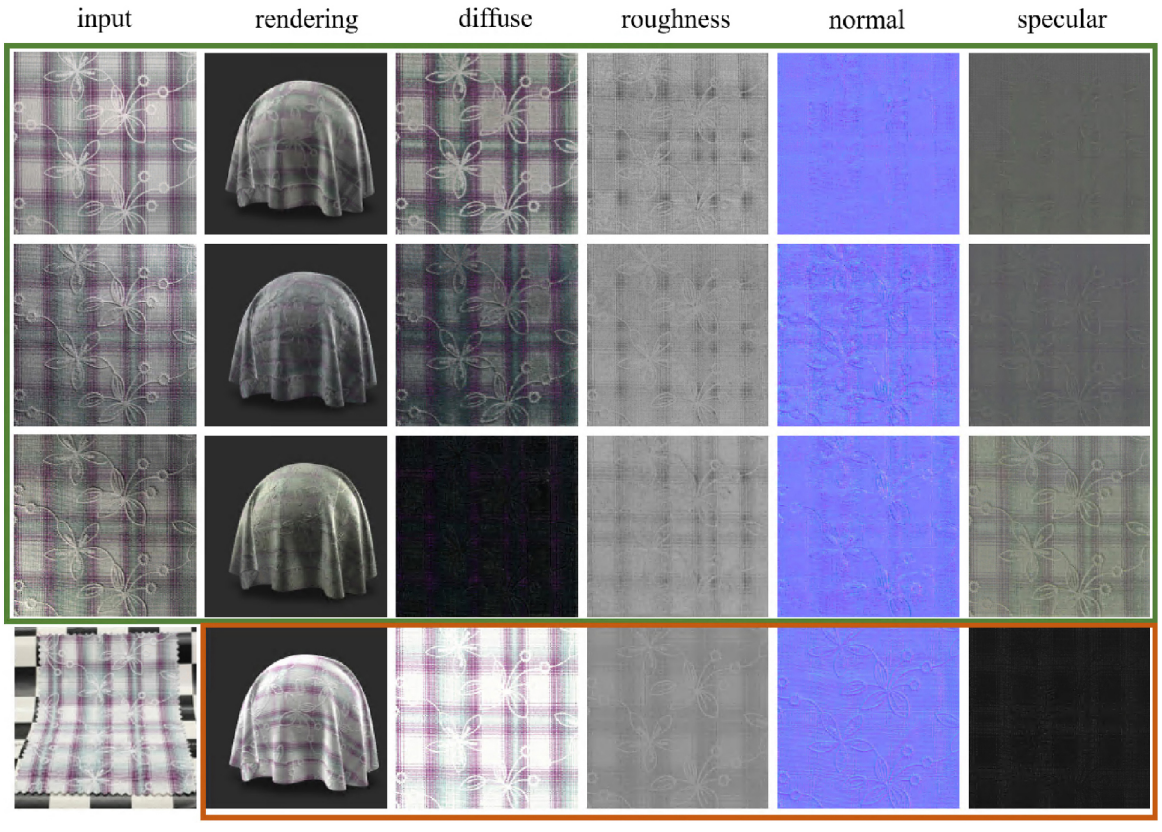


Fig. 1 Examples of SVBRDF acquisitions under different illuminations. The first three rows show the results generated by the same method using cell phone cameras. The 4-th row shows our result with stable illuminations. The bottom-left shows the photo of the real material captured by an SLR camera under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

maps and the shape of the 3D model. The device can generate a stable illumination environment by controlling different LEDs. However, the setup of their method requires 24,576 white LEDs and an Intel Cyclone 10 FPGA, which makes the hardware complex and expensive.

In this study, 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 conditions. First, we designed easy-to-use equipment to control the stray light interference. With this setup, the photographs were taken under stable illumination. This provides a considerable advantage for the input: the testing illuminations are almost the same as the training illuminations. Thus, our network can learn the illumination from all the training samples. By using prior knowledge of illumination, our network can generate a rather accurate inference result. We then analyzed the characteristics of the different maps using the rendering function. Based on these analyses, we built 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 were performed using both synthetic and real data. The results show that our method performs better than the previous methods, even at a resolution of up to 3072×3072. Moreover, we investigated the acquisition quality with different numbers of inputs using our proposed setup.

The main contributions of this study are as follows:

- We propose a novel simple setup for high-quality SVBRDF acquisition. Using our setup, the illuminations between the training and test samples can be maintained, 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 perform extensive 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 map acquisition can be classified into two categories: nearly planar and 3D objects. Studies on early plane objects can be further classified into single-image-based methods and multi-image-based methods, according to the number of inputs. In this section, we briefly review related works on single-image-based near-plane appearance acquisition, multi-images based nearly-plane appearance acquisition, and 3D object appearance acquisition.

2.1 Nearly-plane object appearance acquisition

Single-image-based methods input only one image into a network. Thus, the choice of photography method is important for obtaining the final result. It is common to select an image captured under a flashlight emitted by a handheld device [4, 5, 8, 12]. Under these lighting conditions, the entire material is illuminated, and the light and shadow information on the surface is recorded in a photo. At the same time, the input image can be easily obtained through a mobile phone. Owing to the limitations of the input information, single-image-based methods often show less accuracy than multi-image methods and sometimes fail to produce plausible results.

Multi-image-based methods require several images to be captured under different illumination conditions [6, 13, 14]. It is more complicated than single-image estimation, but works better in terms of accuracy. Deschaintre et al. [6] demonstrated that their method obtained better results with an increase in the number of input images. In addition to deep-learning methods, traditional optimization methods benefit from the addition of images. Gao et al. [14] and MaterialGAN [13] also performed better with more images as optimization targets.

Optimization methods place high demands on users because they must record many complicated parameters of light and cameras [13–15]. Albert et al. [16] proposed a method that utilizes videos to estimate these parameters. However, this requires large amounts of storage space. In addition, deep-learning methods fail when the captured light does not match the training images.

2.2 3D object appearance acquisition

In addition to acquiring the appearance of nearly planar objects, methods have been proposed for 3D objects. To address this problem, a special device such as a camera with a specific linear polarizer [17, 18] or an RGB LED array [19] is used. Holroyd et al. [1] designed a spherical gantry equipped with a projector–camera pair on two mechanical arms using phase-shift patterns for 3D geometry. Tunwattanaponget et al. [20] built a structure with an LED arm that rapidly orbits to create continuous spherical illumination with harmonic patterns and obtained SVBRDF parameters of the object. Other similar dome structures of multiple cameras have also been proposed using structured light patterns for 3D geometry and representing reflectance as a bidirectional texture function (BTF). To eliminate the dependence on structural light, Nam et al. [21] used conventional 3D reconstruction techniques, including SfM, MVS, and mesh reconstruction. Xia et al. [22] proposed the recovery of the 3D shape and isotropic SVBRDF parameters from a captured video sequence of a rotating object. Recently, Kang et al. [11] proposed the construction of a cube-shaped device light stage to capture several photos under different light fields and designed a deep-learning-based framework to capture both the reflectance and 3D shape of the object. However, the proposed device is complex and contains thousands of LEDs and complex control circuit boards. In contrast, in this study, we designed a simpler device that contained only several LEDs to form an illumination environment.

3 Proposed method

3.1 Problem overview

A spatially varying material can be rebuilt using the pixel-level reflectance properties stored in SVBRDF maps. Assuming the Cook–Torrance microfacet

specular shading model and GGX normal distribution function, the reflectance model used in this study was formulated as f_{r}:

$$f_{\mathrm{r}}(v,l,\rho,\alpha,n,F_0) = \underbrace{\frac{\rho}{\pi}}_{\mathcal{P}_{\mathrm{d}}} + \underbrace{\frac{\mathcal{D}(v,l,\alpha)\mathcal{G}(v,l,n)\mathcal{F}(v,l,F_0)}{4(v\cdot n)(l\cdot n)}}_{\mathcal{P}_{\mathrm{h}}} \quad (1)$$

where v and l are the unit vectors of the camera and light directions, respectively; and ρ, α, n, and F_0 are the spatially varying diffuse albedo, roughness, and normal, and specular albedo of the material surface, respectively. f_{r} has two terms: the first term is the diffused part $\mathcal{P}_{\mathrm{d}}$ and the second term is the highlighted part $\mathcal{P}_{\mathrm{h}}$. Our goal was to estimate ρ, α, n, and F_0 from a set of images $\mathcal{I} = \{I_i\}$.

Illumination significantly influences photo I_i. From Eq. (1), it is clear that the light and viewing directions are two significant factors for image I_i. Our observations show that different illuminations cause a well-trained network to fail, yielding erroneous results. Figure 1 shows a failed example by highlight-aware network [8]. Because of illumination mismatch, the diffuse map generated by the network is darker and uneven in brightness. Because it is highly affected by color variance, the predicted normal diverges from reality. Our solution to these problems is to provide a stable illumination in the capture environment. By fixing v and l between the 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

$$f_{\mathrm{r}}(\rho,\alpha,n,F_0) = \mathcal{P}_{\mathrm{d}}(\rho) + \mathcal{P}_{\mathrm{h}}(\mathcal{D}(\alpha)\mathcal{G}(n)\mathcal{F}(F_0)) \quad (2)$$

As image I_i is the combined effect of lights and all four SVBRDF maps, multiple sets of SVBRDF maps might reach the same radiance under special lighting conditions, making it insufficient to infer an accurate map from a single image. Mutual complementary information contained in multiple images of the same material under different lights is essential to alleviate the ambiguities in this problem. Experiments were conducted to demonstrate the effect of the number of input images on the training results in Section 4.3. In our method, we define the number of images $|\mathcal{I}|$ as N. Our task is to determine the generator network $\mathcal{G}$:

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

By training network $\mathcal{G}$, we expect to find an optimal network weight θ_{opt} that minimizes loss $\mathcal{L}$:

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

3.2 Acquisition setup

As shown in Fig. 2, our capture system is a combination of a 20 mega-pixel industrial camera, a material stage, and LED lights distributed on a hemispherical shell with a 225 mm radius.

The hemispherical shell provides a fixed position for the LEDs and cameras. It also minimizes light interference from the 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 placed vertically on top of the hemispherical shell facing the material stage.

Figure 3 shows an example of the captured images of a leather material. The LED positions in the equipment are shown in Fig. 4. The LEDs were distributed at three different levels on a hemispherical shell. In the polar coordinate system originating at the center of the hemisphere, eight equidistant LEDs were installed at each level, with the angle between each level being 22.5°. When the system starts working, the LEDs are lit up to create illumination in different directions. Meanwhile, the camera captures the material on the stage when an LED is turned on. By the end of the capture procedure, we obtained 24 images, each illuminated by a single LED.

A bottom LED was used at the material stage. It turns on to provide blue or green light to the material stage when the material is transparent. When the LED at the bottom was in operation, it first emitted green light, allowing the camera to capture

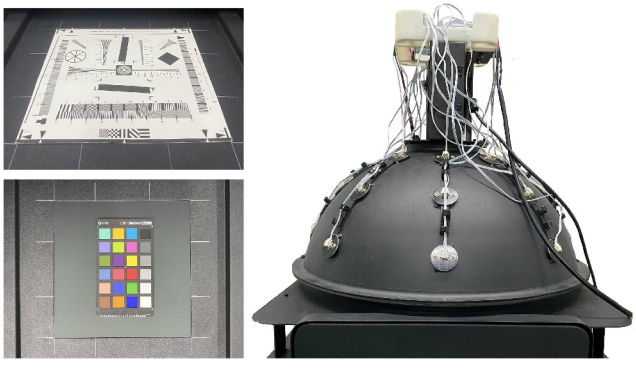


Fig. 2 Appearance of our acquisition device.

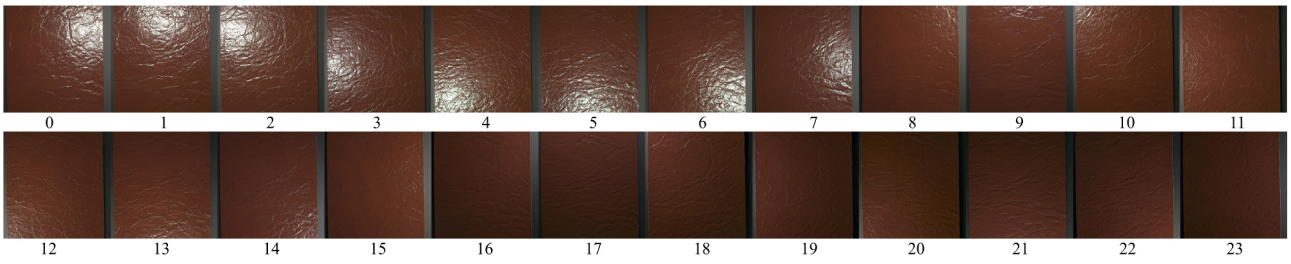


Fig. 3 Images of a leather material captured by our device under the 24 LED lights.

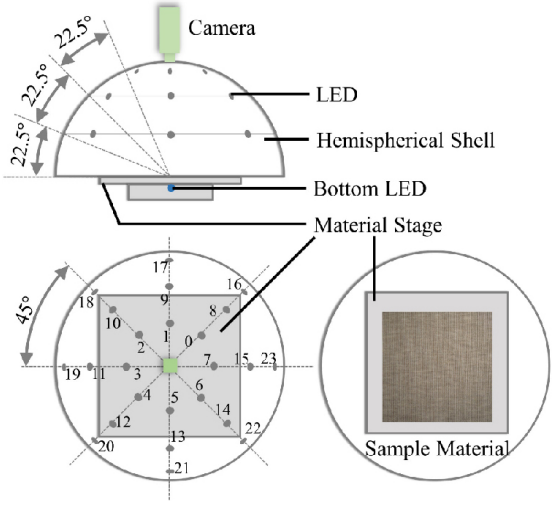


Fig. 4 Positions of lighting LED.

an image of the material with a green background. Subsequently, blue light was emitted to capture an image on a blue background. In Section 4.5, we explain how to determine the transparency of a material using these two special images. The material stage scatters the light emitted by the bottom LED, thereby evenly illuminating it.

Prior to acquisition, we calibrated the camera in our setup with an X-Rite ColorChecker Passport to guarantee high color accuracy during capture. The light intensity was also adjusted between the hardware and the rendering environment using an 18% gray card. By minimizing the L1 distance between the captured photo and its corresponding rendering image, we get a scale parameter, to a total of 24 parameters. 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 [23] as the baseline owing to its ability to solve image-to-image problems. Figure 5 presents an overview of our acquisition method, with the top half showing our training procedure and the bottom half showing how we make inferences using real materials. In the training phase, pairwise training samples $(R_1, ..., R_n)$ for our supervised learning network were generated by rendering with known ground-truth SVBRDF parameters (denoted as d_t,

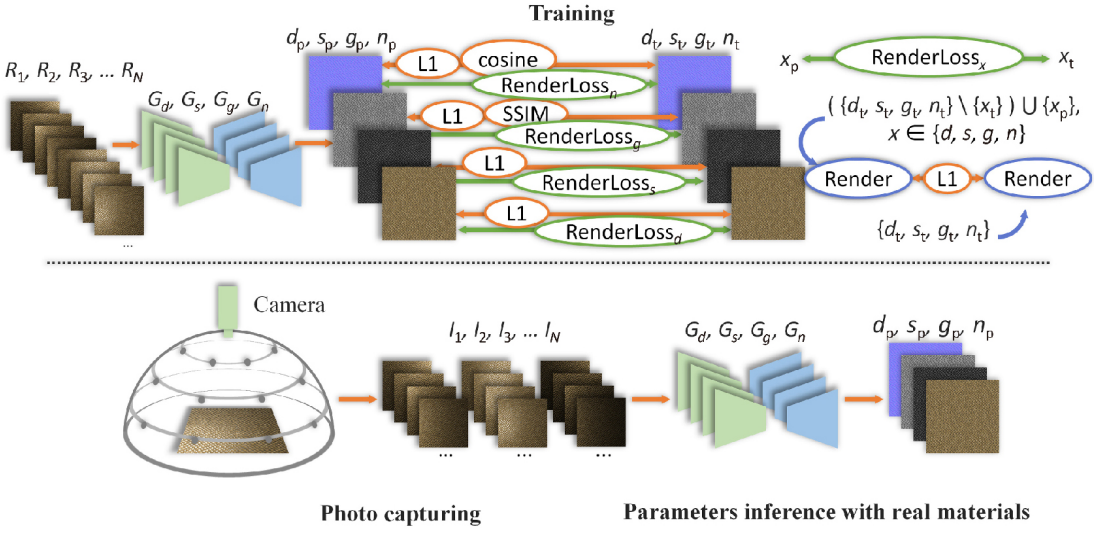


Fig. 5 Overview of our proposed method.

s_t, g_t, and n_t) under the same light settings as our acquisition equipment using the reflectance model defined in Eq. (2): When making inferences about real materials, captured images of the material $(I_1, ..., I_n)$ under different light positions by our acquisition device are input to four different networks (G_d, G_s, G_g, and G_n) to generate the corresponding SVBRDF maps (d_p, s_p, g_p, and n_p). In the following section, we introduce the details of our network architecture and loss functions.

3.3.1 Separated generation networks for four maps

A key distinguishing feature of our framework compared with other works is that we employ four independent networks to generate four different maps separately. Many recent works have adopted the "one-to-four" architecture [8, 12] for the acquisition task of SVBDRF by having a shared encoder for extracting compact features from input images and four separate decoder branches to recover the per-pixel diffuse albedo, specular albedo, normal, and roughness from the learned features. The rationale behind this network design pattern is straightforward. Because the four maps have different emphases on different features, the synthesis of different maps requires four decoders to decode the feature maps differently.

However, this architecture has several limitations. First, in our experiments, we noticed that the four maps could hardly achieve accurate results simultaneously. Because the four decoders share the same encoder, the gradients received by the encoder are related to all the four maps. Suppose a network has already learned to correctly predict three of the four maps. In this case, the nonzero gradient produced by the rest will impose changes on the encoder, indirectly affecting the correctness of the other maps. Second, because the four decoders decode from the same feature maps, the gradients of the four branches tend to reward the encoder that extracts the features required by all four branches.

In summary, this network architecture results in a high degree of entanglement in the feature space. Accordingly, we suggest using a separate network for

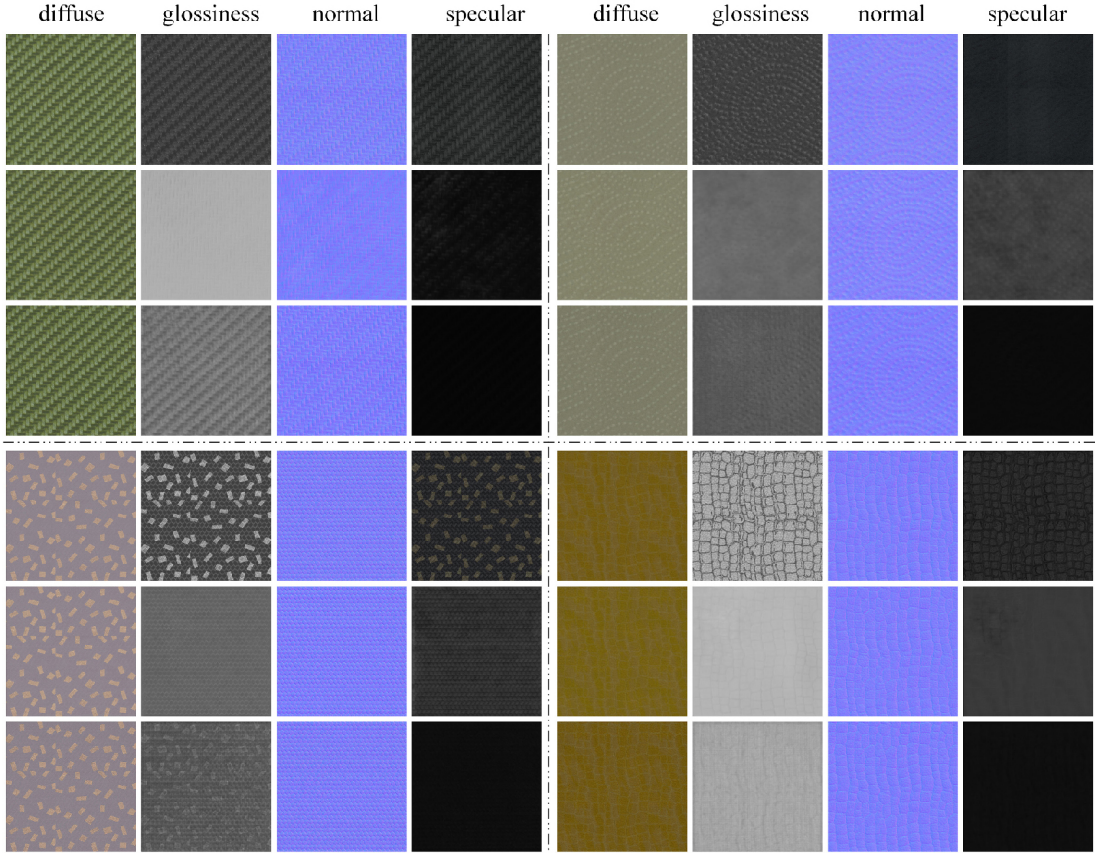


Fig. 6 Comparison between one network strategy and 4 networks strategy. For each example, from top to bottom, the first row shows the reference maps; the second row shows the maps generated with the one network strategy; the third row shows the maps generated with the 4 networks strategy.

each map because each network can better predict a specific map. To mitigate the inconsistencies in different maps, we used a render loss calculated from the estimated map and ground truth maps during the training of the networks. Figure 6 compares the maps generated using one and four networks.

Figure 7 shows the average feature maps over the channels of the four encoders in the second downsampling layer. As shown in the figure, our diffuse network G_d tends to extract features that follow the material pattern, thereby eliminating the interference caused by height changes and uneven light. Similarly, the features extracted by our glossiness network G_g were immune to height differences, presenting the map in a nearly flat manner. By contrast, because the network is most sensitive to changes in height, the normal network pays more attention to the information extracted from moving shadows and brightness. Unlike the other three models, specular network G_s focuses more on the microreflection highlights on the fabric surface. These results further prove the different emphases on the features of the four maps.

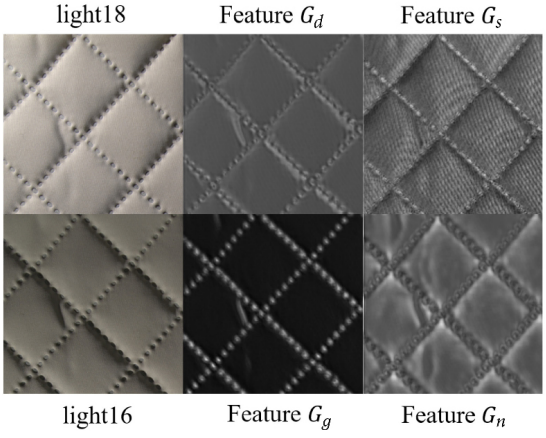


Fig. 7 Feature maps generated by the four networks.

The details of the proposed network architecture are presented in Fig. 8. Before being input to the encoder, the N captured images $I_1, ..., I_N$ are stacked as 72-channel inputs. Subsequently, a single convolution layer is utilized to map the 72-channel input layer into an abstract feature map with the same resolution as the captured images $\mathcal{I}$, but with 64 more condensed (compressed) channels.

Our downsampling block consists of three consecutive 3×3 convolution layers activated by LeakyRelu. We performed downsampling in the first convolution layer with a stride of 2. In this layer, we increased the number of feature channels by 32 and reduced the feature size by half. We set the stride of the following two convolution layers to one and maintained the number of feature channels and their size. Symmetrical to the downsampling block, keeping the last two convolution layers the same, our upsampling block replaces the first convolution layer with a transposed convolution with a stride of two. It also reduces the number of feature channels to the same number as the corresponding encoder layer and doubles the size of the feature maps.

To fully exploit the features at different scales, our generation network downsamples the input images

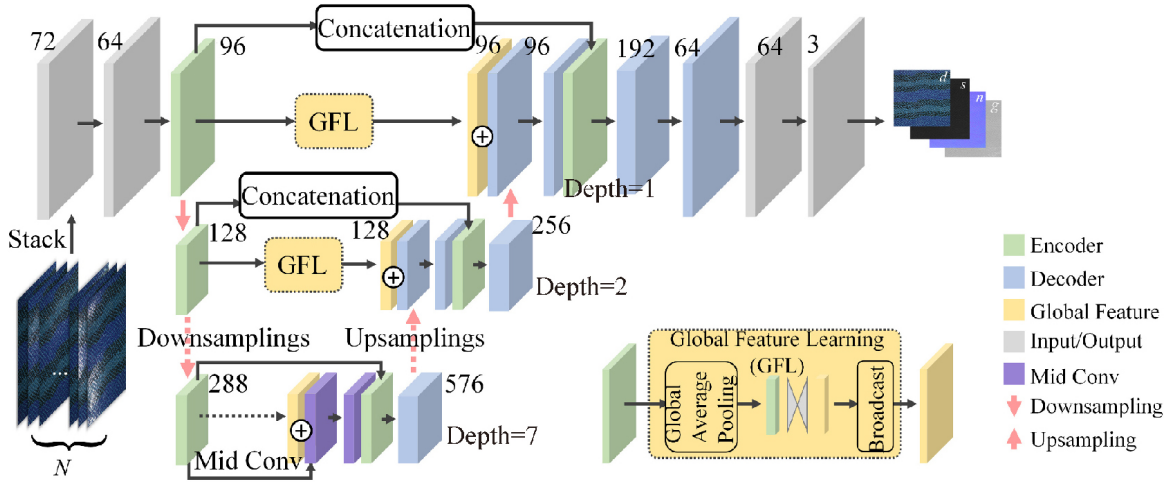


Fig. 8 Architecture of our network with global feature skip connections. Four maps are generated through separate networks, and this figure only shows one of them as an example. The four networks share a similar architecture with only a slight difference at the final convolutional layers. G_d and G_s 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 the sigmoid and tanh functions as active functions, respectively.

seven times through a series of seven downsampling blocks and recovers the SVBRDF maps at the same resolution as $\mathcal{I}$ with seven upsampling blocks. An additional middle convolution layer (bottleneck layer) is employed between the encoder and decoder to refactorize the features learned from the encoder.

Skip connections [23] are made between the encoder and decoder at the same depth to preserve the details at different scales. However, our experiments revealed that skip connections via concatenation were not sufficiently capable of producing pleasant results, leaving unevenness on the generated maps. A skip connection with a global feature learning block was introduced to mitigate this problem by learning common features that span the planar material. The design details and further analysis of the structure are described in Section 3.3.2.

3.3.2 Global skip connection

Although a plain skip connection [23] between the encoder and decoder layers through concatenation can produce generally acceptable results, our observations show that unevenness in brightness can pollute the generated maps, leaving stains on them even if the material surface has evidently uniform reflectance properties (see Fig. 9). The concatenation fusion mechanism between the lower-level features from the encoder and high-level features from the decoder potentially produces a semantic gap [24]. Inspired by Hu et al. [25], we introduced a newly designed global feature skip connection to U-net to address this issue. These connections allow the decoder to be aware of information from other regions. In this manner, some information-lacking regions can infer their information through the data of the information-full regions.

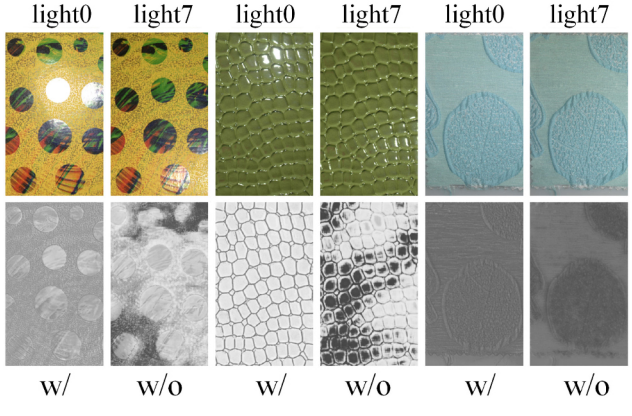


Fig. 9 Three examples of the glossiness map generated w/ or w/o global skip connections.

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**

Our global feature skip connection starts by abstracting a condensed channel-wise global feature vector from the encoder E_i at level i 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 original size by a broadcast operation, as illustrated in Fig. 10. The final output of this module is later fused with the corresponding layer from the decoder D_i using element-wise addition. With D_i and E_i being the ith layers in the decoder and encoder, respectively, this process can be expressed by Eq. (5):

$$D_{i+1} = \mathcal{U}\left(\mathcal{F}_{\mathrm{c}}\left(\mathcal{M}\left(E_i\right)\right) + D_i, E_i\right) \tag{5}$$

where $\mathcal{U}(\cdot)$ indicates an up-sample operation, $\mathcal{F}_{\mathrm{c}}(\cdot)$ indicates a full connection, and $\mathcal{M}(\cdot)$ represents the global average pooling.

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

3.4 Loss function

Having four generation networks to reproduce SVBRDF maps in high quality, we carefully designed specialized joint loss functions L_d, L_s, L_n, and L_g,

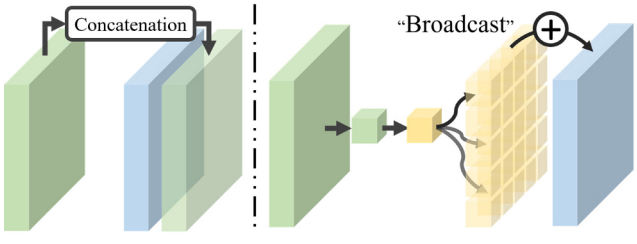


Fig. 10 Comparison of the general skip connection and our global skip connection. The encoder features are first compressed to a value with unit size, and the global skip connection broadcasts it to a complete map. In a general skip connection (left), every field in the decoder can only get information 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.

respectively, for G_d, G_s, G_n and G_g, depending on the different characteristics of the maps they generate. Sharing some common regularizing terms in all loss functions, the loss functions for all four maps consist of a map loss $\mathcal{L}_{\mathrm{m}}$ and a rendering loss $\mathcal{L}_{\mathrm{r}}$ calculated by averaging the mean absolute error between the images rendered with the predicted material maps and the ground truth map using N novel lightings.

This is slightly different from a conventional rendering loss because the SVBRDF parameters in our method are generated separately by four networks. The rendering loss for each network G_x uses one predicted map and three other ground-truth maps, as shown in Eq. (6) where the ground-truth maps $\theta = \{d_{\mathrm{t}}, s_{\mathrm{t}}, g_{\mathrm{t}}, n_{\mathrm{t}}\}$ and $x \in \{d, s, g, n\}$.

$$\mathcal{L}_{\mathrm{r},x} = \sum_{i=1}^{N} \mathrm{MAE}(\mathcal{R}_{\mathrm{l,v}}((\theta \backslash \{x_{\mathrm{t}}\}) \cup \{x_{\mathrm{p}}\}), \mathcal{R}_{\mathrm{l,v}}(\theta)) \quad (6)$$

The map loss $\mathcal{L}_{\mathrm{m}}$ in our method is computed as the $l1$ norm between the predicted maps and the ground-truth maps using the MAE, denoted as $\mathcal{L}_1$. Finally, two weighted factors, λ_{m} and λ_{r}, were applied to the map and rendering losses, respectively, which were set to 1 and 1/24 in our experiments. At this stage, we formally define four joint loss functions, L_d, L_s, L_n, and L_g as Eqs. (7)–(10):

$$L_d = \mathcal{L}_1(d_{\mathrm{t}}, d_{\mathrm{p}}) + \mathcal{L}_{\mathrm{r},d} + (1 - \mathrm{SSIM}(d_{\mathrm{p}}, d_{\mathrm{t}})) \quad (7)$$

$$L_s = \mathcal{L}_1(s_{\mathrm{t}}, s_{\mathrm{p}}) + \mathcal{L}_{\mathrm{r},s} \quad (8)$$

$$L_g = \mathcal{L}_1(g_{\mathrm{t}}, g_{\mathrm{p}}) + \mathcal{L}_{\mathrm{r},g} \quad (9)$$

$$L_n = \mathcal{L}_1(n_{\mathrm{t}}, n_{\mathrm{p}}) + \mathcal{L}_{\mathrm{r},n} + \mathcal{L}_{\mathrm{c}} \quad (10)$$

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 $\mathrm{SSIM}(\cdot)$ are the SSIM values between the two maps.

As a directional value, we use an additional cosine loss L_{c} to evaluate the orientation difference between the predicted normal n_{p} and the ground truth normal n_{t}:

$$\mathcal{L}_{\mathrm{c}} = -\frac{n_{\mathrm{t}}}{|n_{\mathrm{t}}|} \cdot \frac{n_{\mathrm{p}}}{|n_{\mathrm{p}}|} + 1 \quad (11)$$

4 Experiments

4.1 Dataset and training

In this study, we collected 352 real materials including cloth, leather, fabric with a metallic luster or pattern, and fluorescent materials. We first generated SVBRDF maps of these real materials using a commercial material scanner device X-Rite TAC7 Appearance Scanner [26], and then calibrated these maps by professional technical artists under standard illumination in a D65 light box in a dark room. We also expanded the dataset by mixing SVBRDF maps from public datasets. Finally, the newly constructed dataset consisted of 3184 examples. Each example contains SVBRDF maps and 24 rendered images. To obtain the 24-rendered images, we used 3D software to create a virtual digital twin model of our acquisition device, as shown in Fig. 2, and then generated 24 images using Blender Cycles [27] for each example. The resolution of the SVBRDF maps and virtual images is 512×512. In our experiments, we used 2184 for training and 1000 for testing.

We implemented our method using TensorFlow

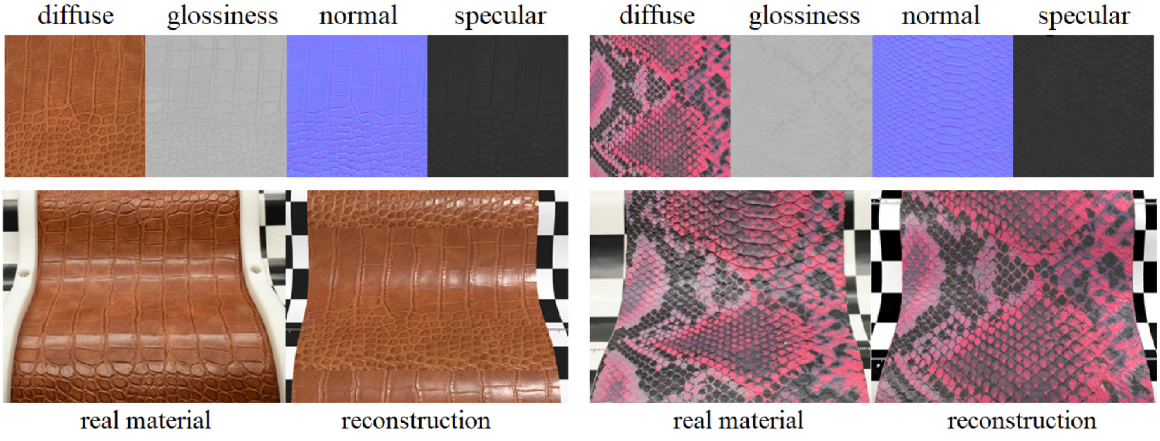


Fig. 11 SVBRDF maps of two real materials generated by our method. The left-bottom is a photo of the material captured by an SLR camera under the standard illumination in D65, while the right-bottom shows the rendering result using the generated maps. The resolution of the maps is 3072×3072.

version 2.4. We used the Adam optimizer [28]. The learning rate was started at 10^{-4}. All other hyperparameters were set to their default values. During training, the batch size was set to 4 for 2000 epochs. Figure 11 shows the results generated by the proposed method.

4.2 Results

We conducted the experiment using two images (Nos. 0 and 16) as inputs. Figure 12 presents an example using the proposed method and those of Deschaintre et al. [6], Guo et al. [13], and Guo et al. [8]. Our method yields results closer to the ground truth, especially for the normal map, whereas the other methods generate incorrect normal results (the normal maps of the flower shape of the cloth are wrong), which leads to incorrect re-rendering results. We conducted numerical experiments using our dataset. For a fair comparison, we fine-tuned the methods using our dataset. Table 2 lists the numerical results for the synthetic data, and Table 3 for real data. Compared to other methods, our method has a significant advantage in terms of diffusion, normality, glossiness, and rendering loss (both synthetic and real). For specular maps, our results are not as good as those of the method proposed by Deschaintre et al. [6]. This was mainly because the materials used were mostly fabrics, which have less prominent specular properties. Therefore, the specular maps produced by our method are not as accurate as those produced by the method proposed by Deschaintre et al. [6].

As described by Deschaintre et al. [6] and materialGAN [13], the recovered SVBRDF maps

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

	Deschaintre et al.	Guo J et al. (materialGAN)	Guo Y et al.	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 and our method on real materials with 2 images input. The first row shows the metrics on re-renderings under 24 lights using our device, while the second row is under novel light

	Deschaintre et al.	Guo J et al. (materialGAN)	Guo Y et al.	Ours
24	0.066352	0.017784	0.068793	**0.013180**
Novel	0.196887	0.087135	0.202598	**0.053856**

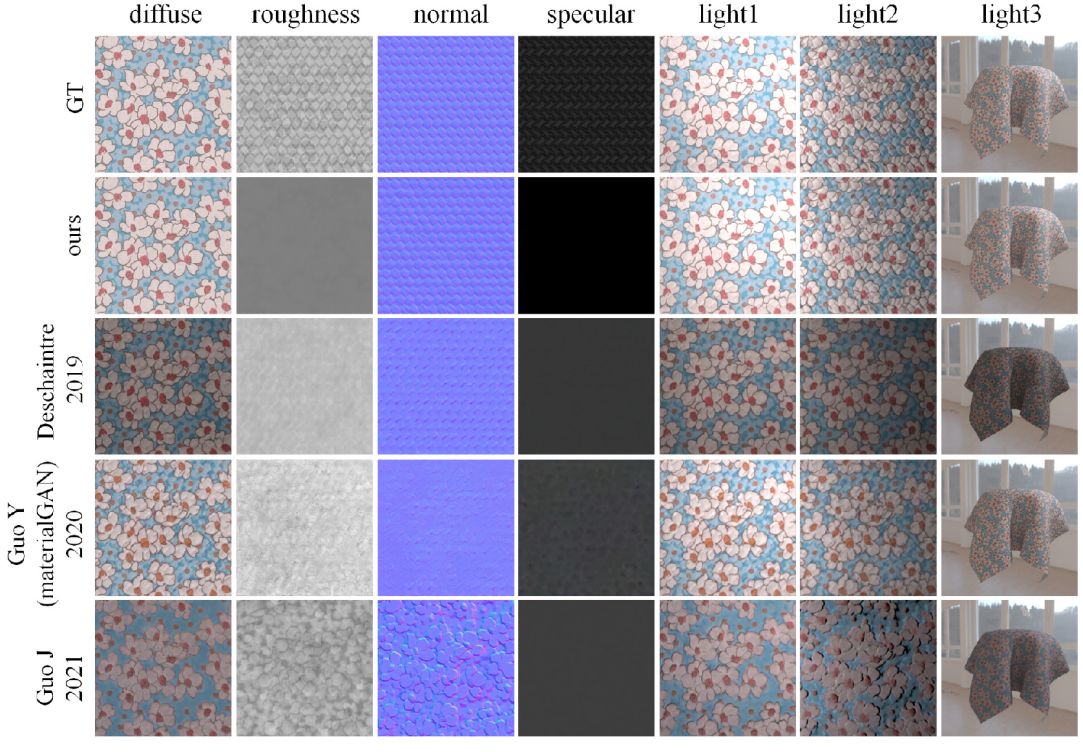


Fig. 12 Reconstruction results using two images as input. Under the side lighting conditions, our rendering results can clearly see the shadow texture generated by the surface bump. The other methods have only blurred dim or almost invisible surface shadows.

improved with an increasing number of inputs. We also conducted experiments and validated this conclusion using our hardware setup and the proposed method. We used the photos captured by our device and trained our network on the training dataset using $24, 16, 10, 6, 3, 2, 1$ photos as inputs (the numbers of selected photos are listed in Table 4). The RMSEs of the diffuse, glossy, normal, and specular maps were computed using the test dataset. The hyperparameters used for training were the same for all inputs. The numerical results are drawn as line graphs, as shown in Fig. 13. These four line graphs show that the RMSE decreases as the number of input images increases. For the specular maps, the curve oscillates when the input number is small, but tends to decline steadily when the number increases. The diffuse, normal, and glossiness maps showed significant improvements at the beginning. When the input number increases to 6, the decline in RMSE slows. Thus, if only the quality of the maps is considered, networks can be trained using as many inputs as possible. The more images that are input, the closer the details of the results are to the ground truth. Figure 14 shows a comparison of the normal maps using different numbers of images as inputs, and we can observe that the details can be obtained well when the number is larger than six. Thus, if the training cost/quality ratio is considered, selecting 6 images as the input is a better choice.

Table 4 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

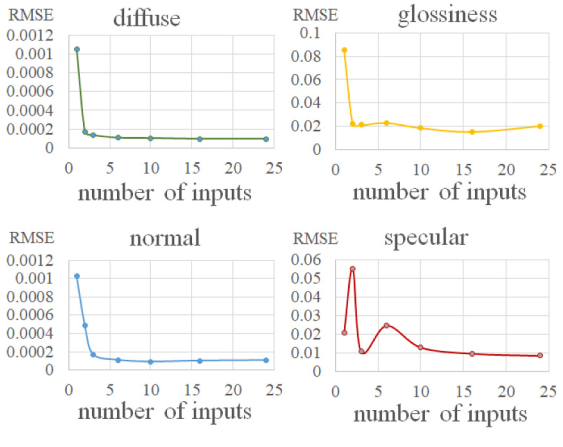


Fig. 13 RMSE in different numbers of inputs.

4.3 Comparisons with more images as input

We compared our method with those proposed by Deschaintre et al. [6], MaterialGAN [13], and Guo et al. [8]. Because the inputs of Deschaintre et al. [6] or materialGAN [13] are multiple images, for fair comparison, we directly utilized the 24 rendered images (or the images cropped from the photos captured by our device) as input. For the method of Guo et al. [8], we traverse the results of 24 images (or cropped photos captured by our device for real materials) and choose the best one for comparison. Qualitative and quantitative experiments are conducted to evaluate the proposed method using our dataset and real materials.

4.3.1 Comparisons on our dataset

Figure 15 presents an example of a comparison using our dataset. The diffuse maps acquired by

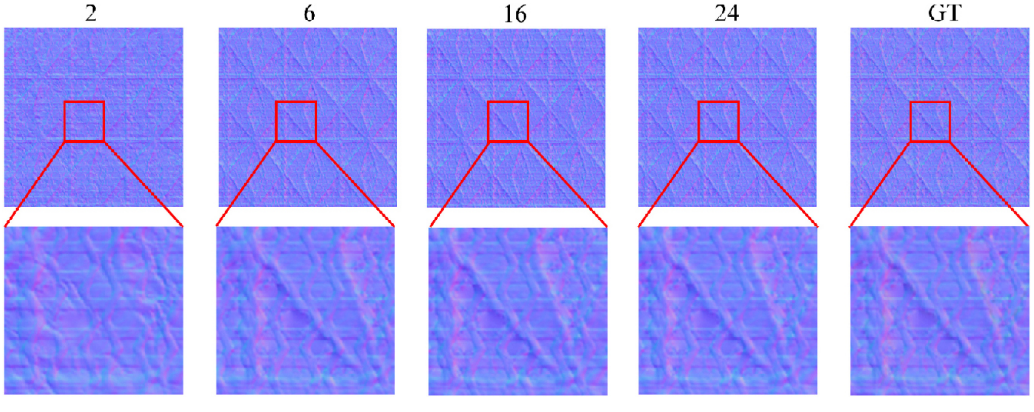


Fig. 14 Comparisons of the normal maps with different numbers of input images.

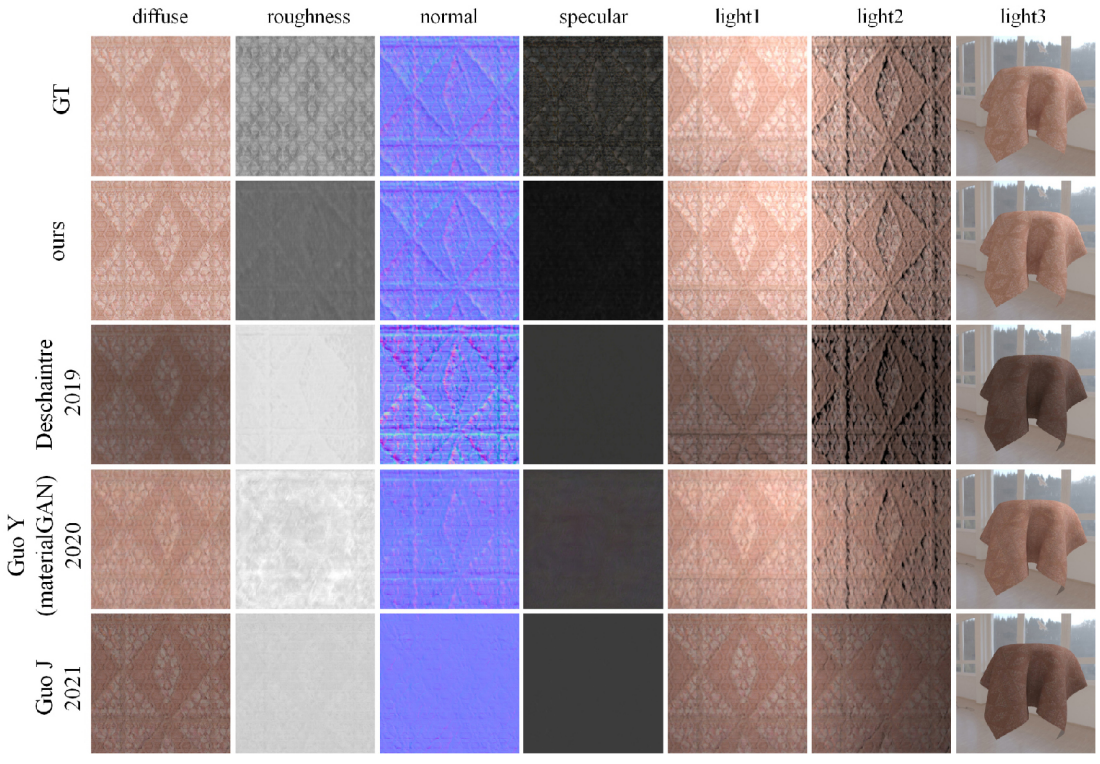


Fig. 15 An example of comparison with prior methods. The diffuse maps, specular maps, and rendering results are shown in Gamma space, while the glossiness maps are turned to roughness maps for more clear visualization.

Deschaintre et al. [6] and Guo et al. [8] are darker than the ground truth, resulting in darker re-rendering results. By contrast, the diffuse maps acquired by our method and MaterialGAN [13] were compatible with the ground truth. For normal maps, our results contain more details and are the closest to the ground truth, whereas the results obtained by Guo et al. [8] and materialGAN [13] tend to be flatter. For the results obtained by Deschaintre et al. [6], the direction of the edge changed significantly.

Note that materialGAN [13] must record the precise parameters of the camera and illumination, making obtaining SVBRDF maps become more complicated. By contrast, our method does not require complex parameters because the illumination of the input images is under fixed control. Our network can learn stable illumination between training samples and use the learned parameters for inference. Thus, the quality of the maps can be guaranteed by the stable illumination provided by our device, which does not require additional optimization.

Table 5 and Table 6 show numerical comparisons of our dataset. The numerical results demonstrate that our method achieves the best results in diffuse, normal, and glossiness maps, whereas the method of Guo et al. [8] has the lowest RMSE and Guo et al. [13] has the lowest LPIPS in specular maps. Although our method does not obtain the best results for specular maps, it obtains the best diffuse and normal maps, which have more important effects on re-rendering.

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

	Deschaintre et al.	Guo J et al. (materialGAN)	Guo Y et al.	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**

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

	Deschaintre et al.	Guo J et al. (materialGAN)	Guo Y et al.	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**

4.3.2 Comparisons on real materials

We validated our method on 85 real materials that were not included in our dataset. The input photographs were captured using the proposed device. We used 3 novel lights to evaluate the results and captured photographs of real materials using an SLR camera. We then rendered the materials using the recovered SVBRDF maps in digital twin illuminations. Figure 16 shows an example of our results. This indicates that the normals generated by the methods of Guo et al. [13] and Guo et al. [8] are incorrect. In fact, the regions of the heart shapes were flat; however, the normals recovered using these two methods were concave. In addition, the recovered diffuse maps do not contain the heart-shaped pattern, which means that these two methods cannot distinguish the color and shadow information of planar exemplar materials. For the maps generated by Deschaintre et al. [6], the recovered diffuse map is gray, while the color of the material is white. Thus, the re-rendering results were significantly different from those of the captured photographs. Table 7 and Table 8 list the RMSE and LPIPS comparisons of the rerendering results with captured photos from previous studies and our method for 85 real materials, respectively. Figure 17 shows the numerical details. Because Guo et al.'s method [8] is based on a single image, we did not compare its performance with that of statistics. This demonstrates that our method can achieve the best results for SVBRDF acquisition of real materials.

Table 7 RMSE comparisons between previous works and our method on real materials. The first row shows the metrics on re-renderings under 24 lights using our device, while the second row shows the metrics under novel lights

	Deschaintre et al.	Guo J et al. (materialGAN)	Guo Y et al.	Ours
24	0.069047	0.013157	0.068793	**0.012600**
Novel	0.204769	0.072937	0.202598	**0.067823**

Table 8 LPIPS comparisons between previous works and our method on real materials. The first row shows the metrics on re-renderings under 24 lights using our device, while the second row shows the metrics under novel lights

	Deschaintre et al.	Guo J et al. (materialGAN)	Guo Y et al.	Ours
24	0.470221	0.377422	0.498515	**0.284123**
Novel	0.630545	0.516387	0.611689	**0.422746**

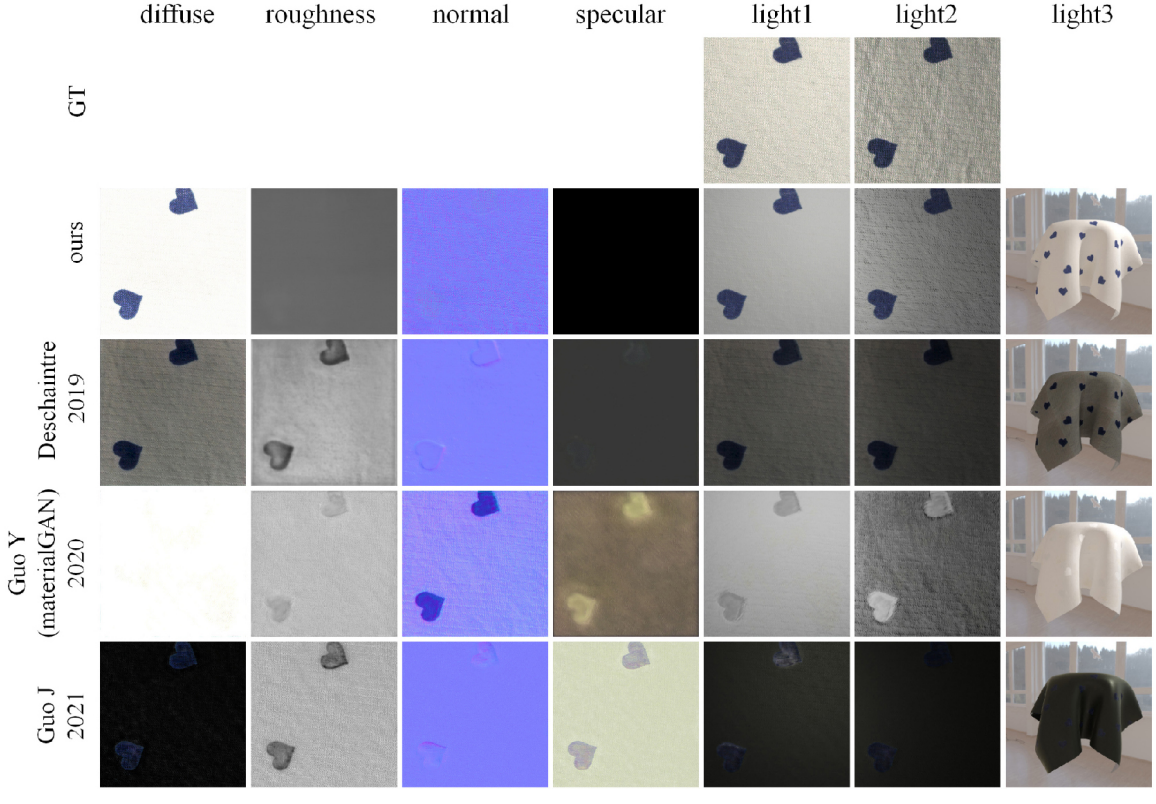


Fig. 16 An example of a real material.

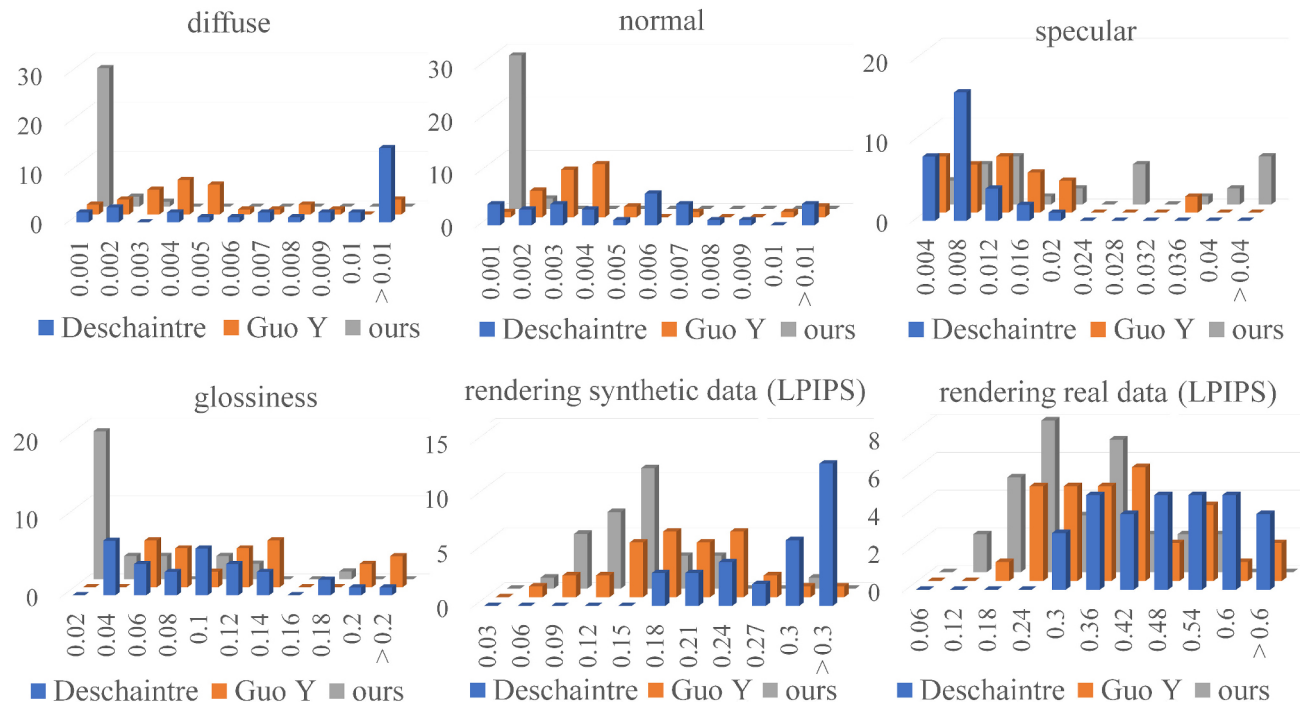


Fig. 17 Statistics from 31 synthetic examples and 31 real materials. We computed the learned perceptual image patch similarity (LPIPS) on the re-rendering images and the root mean square error (RMSE) on SVBRDF maps. In the metrics, a lower value indicates a higher accuracy. Our outputs are more concentrated in the areas with lower values which means that we get more accurate results on most examples.

4.3.3 Performance

We evaluated the runtime performance on a PC with a 3.0 GHz Intel Core i7 processor and an NVIDIA GeForce RTX 3090 GPU. For an input image with a resolution 512×512, it takes approximately 0.11 s using the method of Guo et al. [8] because it only takes one image as the input. For the input from 2 to 24 images, our method takes between 0.300 and 0.480 s, whereas it takes approximately 2.93 s using the method proposed by Deschaintre et al. [6]. In comparison, materialGAN [13] required approximately 660 s with the same input on the same platform because of its lengthy optimization.

4.4 Ablation study

As discussed in Section 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 have affected the accuracy of the one that was correct. We performed ablation studies to validate the performance of the single-encoder architecture. In addition, we compared the performances of our global skip connection and global track [5] to prove the superiority of our method. The results are presented in Table 9.

Table 9 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 be used to acquire PBR maps of special materials, such as mesh, metallic, and fluorescent materials, as shown in Figs. 18 and 21.

An alpha map α was required to simulate the hollow mesh. As illustrated in Fig. 4, the hardware contained a bottom LED light. Before capturing the transparent material, the light stage emits blue and green light, and the camera captures the background images B_{b} and G_{b}. Images of materials B_{c} (with blue light) and G_{c} (with green light) were captured. According to Alvys' method [29], we have

$$\begin{cases} B_{\mathrm{c}} = \alpha F + (1-\alpha) B_{\mathrm{b}} \\ G_{\mathrm{c}} = \alpha F + (1-\alpha) G_{\mathrm{b}} \end{cases} \tag{12}$$

where F denotes the color of the object. As $B_{\mathrm{b}}, G_{\mathrm{b}}, B_{\mathrm{c}}$, and G_{c} are known, by solving Eq. (12),

α can be obtained. Figures 18 and 21 show the alpha maps reconstructed using our method.

For materials with a metallic luster or pattern, we trained the networks using the same proposed method with a metallic workflow (rendered by the base color, metallic, normal, and roughness maps). Figures 18 and 19 show two examples of reconstructed metallic maps. For fluorescent materials, the emissive map can be obtained in a similar manner. Two examples of reconstructed fluorescent materials are shown in Figs. 20 and 21. It should be noted that the displacement map was converted from a normal map during reconstruction.

4.6 Comparisons with handheld devices

Compared with handheld devices such as mobile phones, our setup can achieve better results, albeit with a slightly more complex setup. Using our setup, we can generate alpha maps of the materials by controlling the bottom LED, which is not possible using mobile phones. Figure 22 shows the differences in the maps generated using photos taken with our

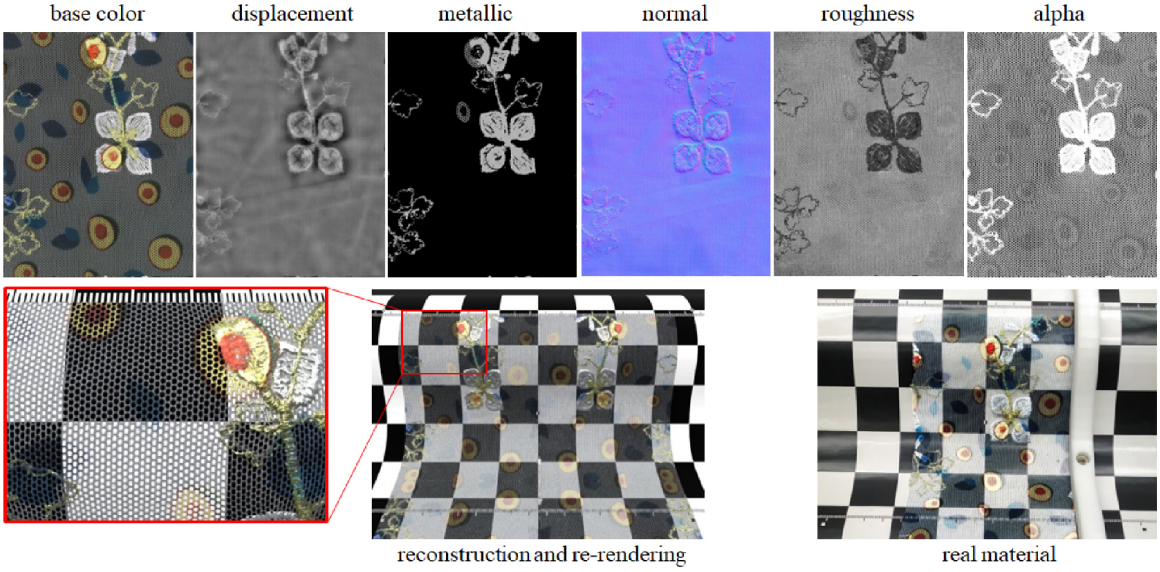


Fig. 18 Reconstruction of a mesh material with metallic patterns. The first row shows the maps obtained using our device and method.

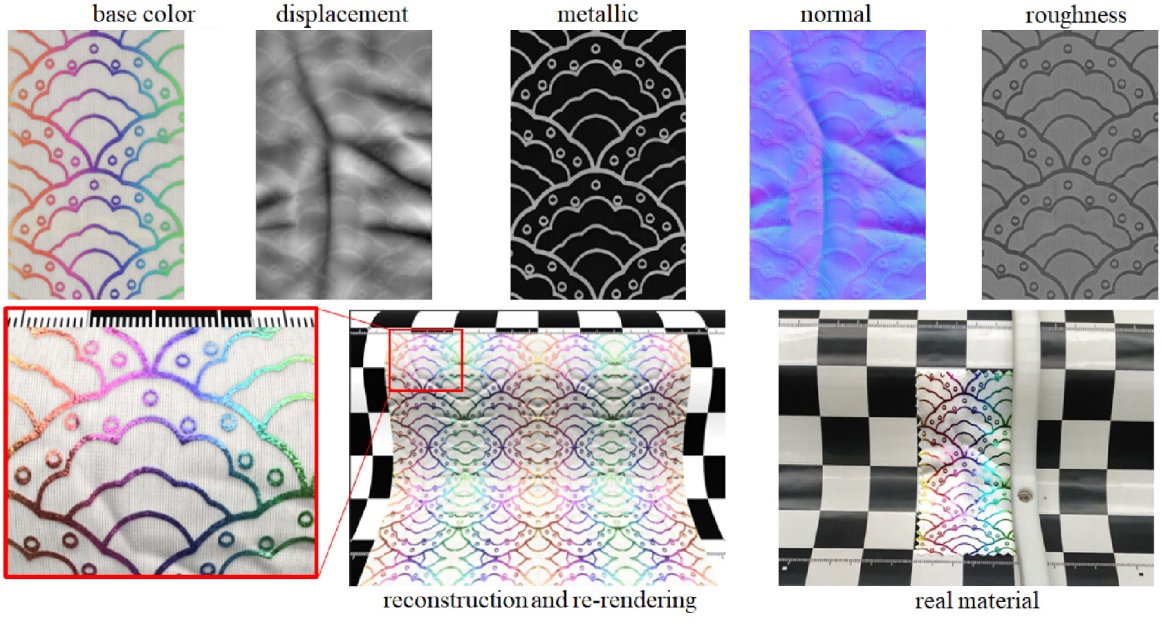


Fig. 19 Reconstruction of a fabric material with metallic patterns. To better express the metallic luster of materials, the workflow for reconstructing such materials employs the metallic workflow.

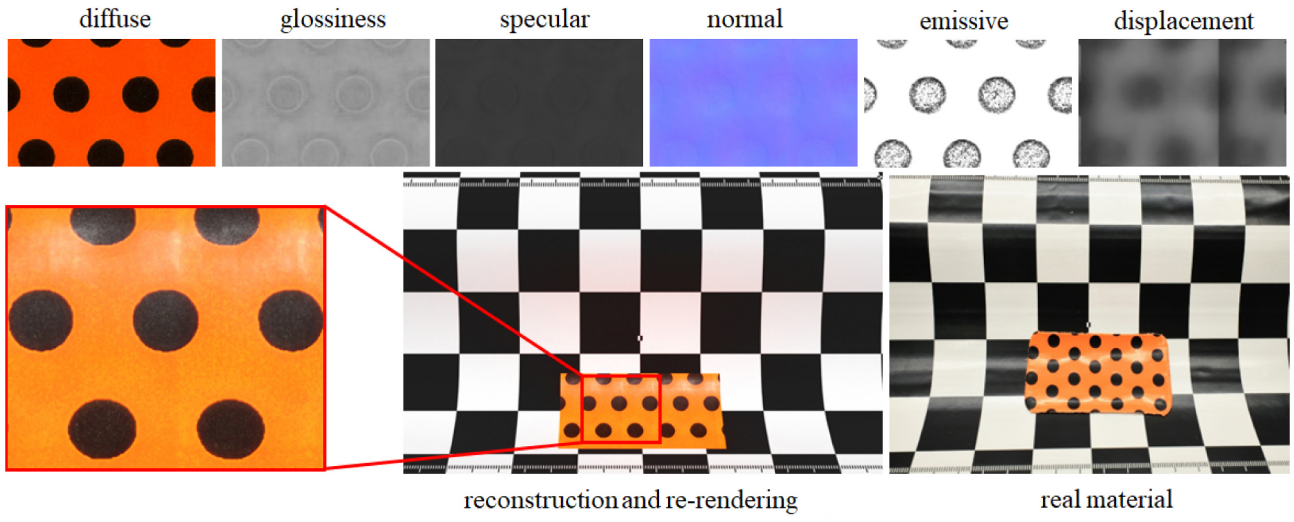


Fig. 20 Reconstruction of a fluorescent material.

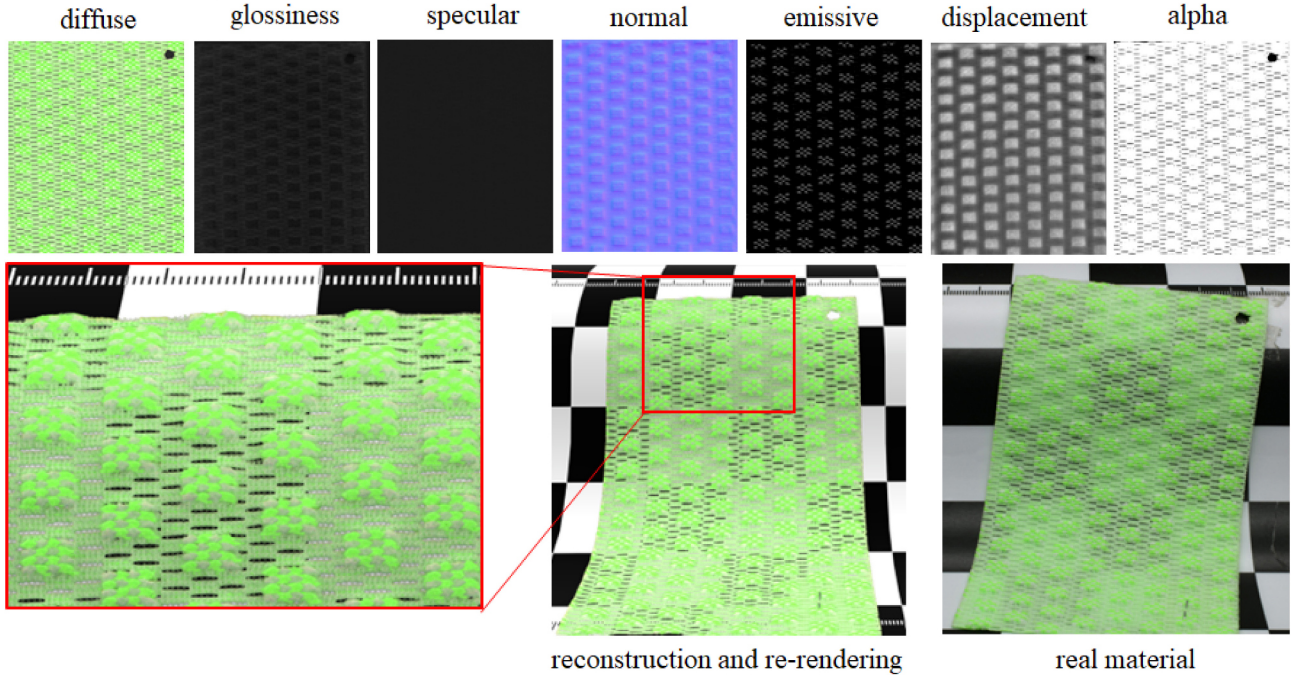


Fig. 21 Reconstruction of a mesh-fluorescent fabric.

setup versus those captured by a handheld device. The left image features a mesh fabric, and the second row demonstrates the maps and renderings captured by the mobile phone. Without our device, it is impossible to calculate alpha maps, and the background color cannot be seen through the hole in the mesh.

Furthermore, when using deep-learning methods, it is challenging to ensure consistency in the generated maps across different illumination conditions for the same material on handheld devices. This difficulty arises because it is difficult to guarantee that the illumination conditions in the photos captured by handheld devices are 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 by Guo et al. [8], which relies on handheld devices. In addition, as an example of a fabric with a cartoon-like pattern shown in Fig. 23, our proposed method exhibits color bias without the environmental control provided by our setup.

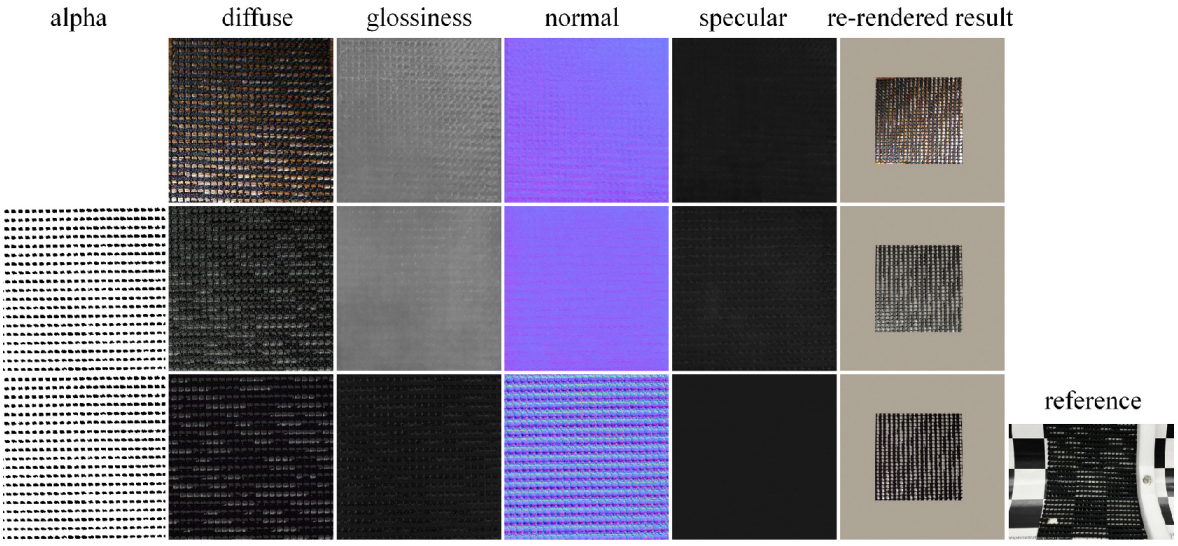


Fig. 22 An example of a mesh fabric material. The right image is the reference mesh fabric captured using DLR under the D65 light box. The first row shows the maps and re-rendered images generated using photos captured by a mobile phone. The second and third rows show the maps and re-rendered results generated using our setup, with 2 and 24 captured images respectively.

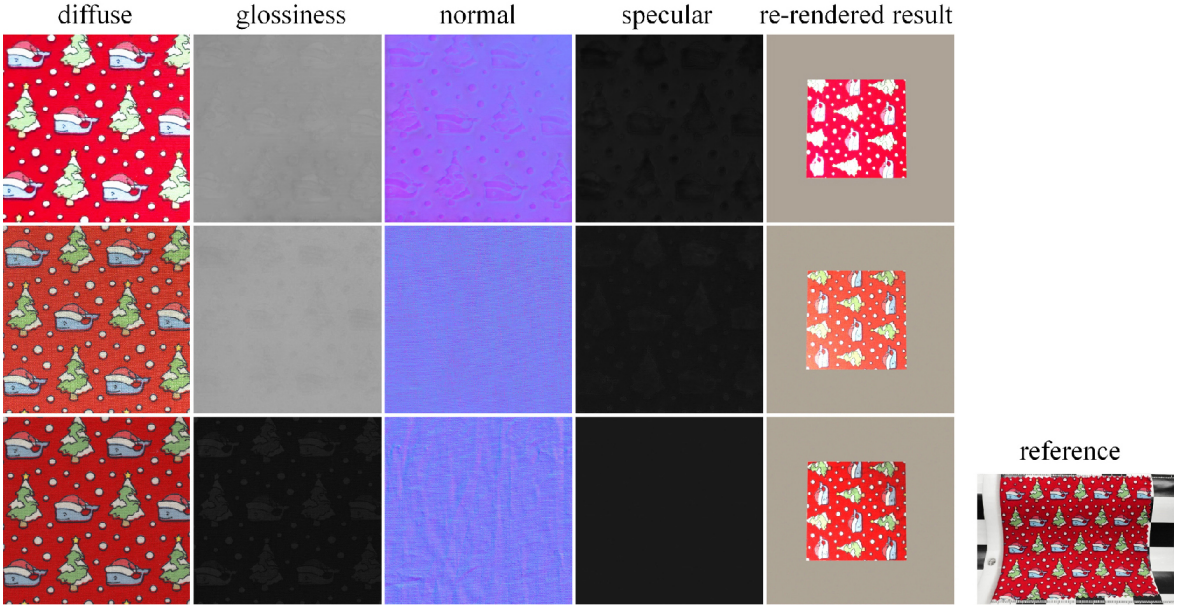


Fig. 23 An example of a fabric with a cartoon pattern. The right image is the reference fabric captured using a DLR under the D65 light box. The first row shows the maps and re-rendered images generated using photos captured by a mobile phone. The second and third rows show the maps and re-rendered results generated using our setup, with 2 and 24 captured images respectively.

5 Conclusions

In this study, we propose a novel setup and network for obtaining high-quality SVBRDF maps. We highlighted the importance of stable lighting patterns for deep-learning-based methods and delved into studying the relationship between the acquisition quality of different numbers of images as input. We also described the necessity of separating generation networks for each map. Then, we show that our naive global skip connection can pass global and local information between the decoder and encoder. We also experimentally investigated the effect of the number of input images. Our results show that our method outperforms existing methods on both our dataset and real materials. Our proposed method can also reconstruct 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 efficiently acquired using the proposed hardware setup.

Acknowledgements

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

Declaration of competing interest

The authors have no competing interests to declare that are relevant to the content of this article.

Electronic Supplementary Material

We have provided a video displaying our results. This video is available at `https://github.com/hgjljx/Delving-High-quality-SVBRDF-Acquisition-a-New-Setup-and-Method.git`.

References

[1] Holroyd, M.; Lawrence, J.; Zickler, T. A coaxial optical scanner for synchronous acquisition of 3D geometry and surface reflectance. *ACM Transactions on Graphics* Vol. 29, No. 4, Article No. 99, 2010.

[2] Dana, K. J.; van Ginneken, B.; Nayar, S. K.; Koenderink, J. J. Reflectance and texture of real-world surfaces. *ACM Transactions on Graphics* Vol. 18, No. 1, 1–34, 1999.

[3] Lawrence, J.; Ben-Artzi, A.; DeCoro, C.; Matusik, W.; Pfister, H.; Ramamoorthi, R.; Rusinkiewicz, S. Inverse shade trees for non-parametric material representation and editing. *ACM Transactions on Graphics* Vol. 25, No. 3, 735–745, 2006.

[4] Li, Z. Q.; Sunkavalli, K.; Chandraker, M. Materials for masses: SVBRDF acquisition with a single mobile phone image. In: *Computer Vision – ECCV 2018. Lecture Notes in Computer Science, Vol. 11207*. Ferrari, V.; Hebert, M.; Sminchisescu, C.; Weiss, Y. Eds. Springer Cham, 74–90, 2018.

[5] Deschaintre, V.; Aittala, M.; Durand, F.; Drettakis, G.; Bousseau, A. Single-image SVBRDF capture with a rendering-aware deep network. *ACM Transactions on Graphics* Vol. 37, No. 4, Article No. 128, 2018.

[6] Deschaintre, V.; Aittala, M.; Durand, F.; Drettakis, G.; Bousseau, A. Flexible SVBRDF capture with a multi-image deep network. *Computer Graphics Forum* Vol. 38, No. 4, 1–13, 2019.

[7] Asselin, L. P.; Laurendeau, D.; Lalonde, J. F. Deep SVBRDF estimation on real materials. In: Proceedings of the International Conference on 3D Vision, 1157–1166, 2020.

[8] Guo, J.; Lai, S. C.; Tao, C. Z.; Cai, Y. L.; Wang, L.; Guo, Y. W.; Yan, L. Q. Highlight-aware two-stream network for single-image SVBRDF acquisition. *ACM Transactions on Graphics* Vol. 40, No. 4, Article No. 123, 2021.

[9] Zhao, Y. Z.; Wang, B. B.; Xu, Y. N.; Zeng, Z.; Wang, L.; Holzschuch, N. Joint SVBRDF recovery and synthesis from a single image using an unsupervised generative adversarial network. In: Proceedings of the Eurographics Symposium on Rendering, 53–66, 2020.

[10] Wen, T.; Wang, B. B.; Zhang, L.; Guo, J.; Holzschuch, N. SVBRDF recovery from a single image with highlights using a pre-trained generative adversarial network. *Computer Graphics Forum* Vol. 41, No. 6, 110–123, 2022.

[11] Kang, K. Z.; Xie, C. H.; He, C. G.; Yi, M. Q.; Gu, M. Y.; Chen, Z. M.; Zhou, K.; Wu, H. Z. Learning efficient illumination multiplexing for joint capture of reflectance and shape. *ACM Transactions on Graphics* Vol. 38, No. 6, Article No. 165, 2019.

[12] Zhou, X. L.; Kalantari, N. K. Adversarial single-image SVBRDF estimation with hybrid training. *Computer Graphics Forum* Vol. 40, No. 2, 315–325, 2021.

[13] Guo, Y.; Smith, C.; Hasan, M.; Sunkavalli, K.; Zhao, S. MaterialGAN: Reflectance capture using a generative SVBRDF model. *ACM Transactions on Graphics* Vol. 39, No. 6, Article No. 254, 2020.

[14] 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 Transactions on Graphics* Vol. 38, No. 4, Article No. 134, 2019.

[15] Ye, W. J.; Dong, Y.; Peers, P.; Guo, B. N. Deep reflectance scanning: Recovering spatially-varying material appearance from a flash-lit video sequence. *Computer Graphics Forum* Vol. 40, No. 6, 409–427, 2021.

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

[17] Baek, S. H.; Jeon, D. S.; Tong, X.; Kim, M. H. Simultaneous acquisition of polarimetric SVBRDF and normals. *ACM Transactions on Graphics* Vol. 37, No. 6, Article No. 268, 2018.

[18] Deschaintre, V.; Lin, Y. M.; Ghosh, A. Deep polarization imaging for 3D shape and SVBRDF acquisition. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 15562–15571, 2021.

[19] Ma, X. H.; Kang, K. Z.; Zhu, R. S.; Wu, H. Z.; Zhou, K. Free-form scanning of non-planar appearance with neural trace photography. *ACM Transactions on Graphics* Vol. 40, No. 4, Article No. 124, 2021.

[20] Tunwattanapong, B.; Fyffe, G.; Graham, P.; Busch, J.; Yu, X. M.; Ghosh, A.; Debevec, P. Acquiring reflectance and shape from continuous spherical harmonic illumination. *ACM Transactions on Graphics* Vol. 32, No. 4, Article No. 109, 2013.

[21] Nam, G.; Lee, J. H.; Gutierrez, D.; Kim, M. H. Practical SVBRDF acquisition of 3D objects with unstructured flash photography. *ACM Transactions on Graphics* Vol. 37, No. 6, Article No. 267, 2018.

[22] Xia, R.; Dong, Y.; Peers, P.; Tong, X. Recovering shape and spatially-varying surface reflectance under unknown illumination. *ACM Transactions on Graphics* Vol. 35, No. 6, Article No. 187, 2016.

[23] Ronneberger, O.; Fischer, P.; Brox, T. U-net: Convolutional networks for biomedical image segmentation. In: *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2015. Lecture Notes in Computer Science, Vol. 9351*. Navab, N.; Hornegger, J.; Wells, W.; Frangi, A. Eds. Springer Cham, 234–241, 2015.

[24] Zhou, Z. W.; Rahman Siddiquee, M. M.; Tajbakhsh, N.; Liang, J. M. UNet++: A nested U-net architecture for medical image segmentation. In: *Deep Learning in Medical Image Analysis and Multimodal Learning for Clinical Decision Support. DLMIA ML-CDS 2018. Lecture Notes in Computer Science, Vol. 11045*. Stoyanov, D., et al. Eds. Springer Cham, 3–11, 2018.

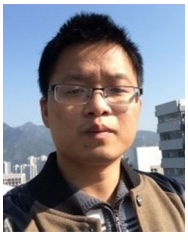
[25] Hu, J.; Shen, L.; Sun, G. Squeeze-and-excitation networks. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 7132–7141, 2018.

[26] X-Rite: TAC7 appearance scanner. 2017. Available at https://www.xrite.com/categories/appearance/total-appearance-capture-ecosystem/tac7

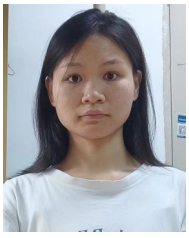
[27] Cycles: Open source production rendering. 2022. Available at https://www.cycles-renderer.org/

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

[29] Smith, A. R.; Blinn, J. F. Blue screen matting. In: Proceedings of the 23rd Annual Conference on Computer Graphics and Interactive Techniques, 259–268, 1996.

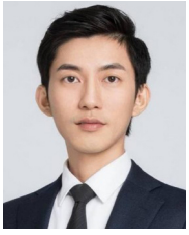


Chuhua Xian is currently an associate professor with the School of Computer Science and Engineering at South China University of Technology. He received his Ph.D. degree in computer science from the State Key Lab of CAD&CG at Zhejiang University in 2012. He was a postdoctoral researcher at CUHK from 11/2013 to 05/2014 and 09/2015 to 04/2016. His research interests include computer graphics, computer vision, image processing, and geometric processing.



Jiaxin Li is currently a graduate student at the School of Computer Science and Engineering at South China University of Technology. Her research interests include computer graphics, vision, and inverse rendering.

Hao Wu is currently the leader of the AI Lab of Guangdong Shidi Intelligence Technology, Ltd. She got her Ph.D. degree in computer science from Hong Kong University in 2019. Her research interests include computer graphics, rendering, computer vision, and generative models.



Zisen Lin is currently the CEO of Guangdong Shidi Intelligence Technology, Ltd. His research interests include computer graphics and computer vision.

Guiqing Li is currently a full-time professor at the School of Computer Science and Engineering, South China University of Technology in Guangzhou, China. His research interests include computer graphics, and geometric and image processing.