Correlation-aware Encoder-Decoder with Adapters for SVBRDF Acquisition

Di Luo*
College of Computer Science, Nankai University
China
diluo@mail.nankai.edu.cn

Hanxiao Sun*
College of Computer Science, Nankai University
China
hx.sun@mail.nankai.edu.cn

Lei Ma
School of Computer Science, Peking University
China
lei.ma@pku.edu.cn

Jian Yang
College of Computer Science, Nankai University
China
csjyang@njust.edu.cn

Beibei Wang†
School of Intelligence Science and Technology, Nanjing University
China
beibei.wang@nju.edu.cn

Figure 1: By modeling the correlation among input images with an encoder, together with an adapter-equipped decoder, our network achieves high-quality SVBRDF recovery on both isotropic and anisotropic (with roughness encoded in red and green channels) materials. Here we show re-rendered views for four materials under environment illumination. (Please use Adobe Acrobat and click the renderings to see the animation.)

ABSTRACT

Capturing materials from the real world avoids laborious manual material authoring. However, recovering high-fidelity Spatially Varying Bidirectional Reflectance Distribution Function (SVBRDF) maps from a few captured images is challenging due to its ill-posed nature. Existing approaches have made extensive efforts to alleviate this ambiguity issue by leveraging generative models with latent space optimization or extracting features with variant encoder-decoders. Albeit the rendered images at input views can match input images, the problematic decomposition among maps leads to significant differences when rendered under novel views/lighting. We observe that for human eyes, besides individual images, the correlation (or the highlights variation) among input images also serves as an important hint to recognize the materials of objects. Hence, our key insight is to explicitly model this correlation in the SVBRDF acquisition network. To this end, we propose a correlation-aware encoder-decoder network to model the correlation features among the input images via a graph convolutional network by treating channel features from each image as a graph node. This way, the ambiguity among the maps has been reduced significantly. However, several SVBRDF maps still tend to be over-smooth, leading to a mismatch in the novel-view rendering. The main reason

*Contribute equally.
†Corresponding author.

Authors' addresses: Di Luo, College of Computer Science, Nankai University, China, diluo@mail.nankai.edu.cn; Hanxiao Sun, College of Computer Science, Nankai University, China, hx.sun@mail.nankai.edu.cn; Lei Ma, School of Computer Science, Peking University, China, lei.ma@pku.edu.cn; Jian Yang, College of Computer Science, Nankai University, China, csjyang@njust.edu.cn; Beibei Wang, School of Intelligence Science and Technology, Nanjing University, China, beibei.wang@nju.edu.cn.

ACM 0730-0301/2024/9-ART
https://doi.org/10.1145/3680528.3687594

is the uneven update of different maps caused by a single decoder for map interpretation. To address this issue, we further design an adapter-equipped decoder consisting of a main decoder and four tiny per-map adapters, where adapters are employed for individual maps interpretation, together with fine-tuning, to enhance flexibility. As a result, our framework allows the optimization of the latent space with the input image feature embeddings as the initial latent vector and the fine-tuning of per-map adapters. Consequently, our method can outperform existing approaches both visually and quantitatively on synthetic and real data.

CCS CONCEPTS

• **Computing methodologies** → **Rendering**; **Reflectance modelings**.

KEYWORDS

SVBRDF, capture, graph convolution network

ACM Reference Format:
Di Luo, Hanxiao Sun, Lei Ma, Jian Yang, and Beibei Wang. 2024. Correlation-aware Encoder-Decoder with Adapters for SVBRDF Acquisition. *ACM Trans. Graph.* 1, 1 (September 2024), 11 pages. https://doi.org/10.1145/3680528.3687594

1 INTRODUCTION

Spatially-varying materials are crucial to raise realism in many applications, e.g., video games, virtual reality, etc. These materials are usually termed Spatially Varying Bidirectional Reflectance Distribution Function (SVBRDF) maps, where each pixel defines the parameters of the Bidirectional Reflectance Distribution Function (BRDF), including the roughness, normal, specular, and diffuse albedo. Designing these materials manually is time-consuming, even for experienced artists. The alternative is capturing materials from the real world to avoid the laborious design. However, recovering high-fidelity SVBRDF maps from a few captured images is challenging due to the ambiguities among different maps.

Deep learning has opened up extensive opportunities for material acquisition from one [Deschaintre et al. 2018; Guo et al. 2021; Zhang et al. 2023; Zhou and Kalantari 2022] or several [Deschaintre et al. 2019; Gao et al. 2019; Guo et al. 2020] captured images. Among these approaches, one group of works represent the SVBRDF space with a latent space and then recover SVBRDFs by latent space optimization [Gao et al. 2019; Guo et al. 2020]. Another group of works [Deschaintre et al. 2018, 2019; Guo et al. 2021] design variant encoder-decoder structures to extract features and predict SVBRDF maps from these features. Despite the renderings in the input views can match input images, significant differences are shown when rendering under novel light/view conditions, due to the ambiguous decomposition.

In this paper, we aim to alleviate the ambiguity among the maps. For that, our key insight is that the correlation among input images serves as an important hint for human eyes, which can provide more clues for SVBRDF recovery. To this end, we propose a novel correlation-aware encoder-decoder framework for SVBRDF recovery from multiple images. To model the correlation among input images, we introduce a graph convolution network (GCN) into the encoder to learn the adjacency matrix of channel features. This way, the feature representing the variation at each pixel under different light conditions provides strong hints for material recovery. While the correlation-aware network structure alleviates the ambiguity, leading to less light burn-in artifacts, the learned SVBRDFs tend to be over-smooth, as they are updated with a single decoder, which lacks flexibility to adjust each map. To address this issue, we design a new structure for the decoder, by introducing an extra tiny adapter to interpret each map, which allows fine-tuning during optimization. Finally, the features encoded by the correlation-aware encoder form a latent space, which enables optimization with the input image feature embeddings as the initialization and the per-map adapters allow fine-tuning, leading to a network with high capability and flexibility. The capability of our full solution is demonstrated on two datasets ([Deschaintre et al. 2018] and [Ma et al. 2023]) by comparing with the state-of-the-art (SOTA) single/multiple SVBRDF recovery approaches. Our method can outperform these methods on both synthetic data and real data.

In summary, our contributions include:

- We propose a correlation-aware encoder, which leverages graph convolutional network to model correlation among input images, leading to less ambiguity among SVBRDF maps.
- We design a novel decoder for SVBRDF interpretation, by introducing tiny per-map adapters, which address the over-smooth issues of maps.
- Finally, with our correlation-aware encoder-decoder framework, we introduce a new optimization strategy, combining the feature embedded latent space optimization and per-map adapter fine-tuning to achieve high-quality SVBRDF maps.

2 RELATED WORK

In this section, we review the work related to lightweight SVBRDF recovery and then provide a brief overview of works related to GCN.

2.1 Multi-image SVBRDF recovery

Recovering SVBRDFs from multiple images has been a long-standing problem. Due to its complexity, traditional methods for this task usually rely on some domain-specific priors or assumptions, such as known illumination conditions [Chandraker 2014; Hui and Sankaranarayanan 2015; Riviere et al. 2016], self-similarity [Aittala et al. 2015] or spatial relation [Xu et al. 2016] between images. Naturally, these assumptions limit their applicability. For more details on traditional methods, please refer to Guarnera et al. [2016].

Thanks to Deschaintre et al. [2018], a large synthetic dataset has been made available, allowing deep learning methods to be applied to material recovery. Deschaintre et al. [2019] design a network architecture to handle an arbitrary number of input images. They use the shared encoder to extract the features and aggregate them from different images by max-pooling, and then decode the features to obtain the final SVBRDF maps. Unlike their max-pooling operation, which keeps the maximum response for multiple features, our method extracts features with a GCN, which learns the adjacency matrix between channel features, leading to richer information. Gao et al. [2019] first propose to rationally utilize the network priors in the SVBRDF recovery task by employing the latent space optimization via an adaptive auto-encoder. Nevertheless, their work depends

on plausible initialization provided by other networks, which inevitably introduces bias (e.g., artifacts and over-smooth maps) and impacts the recovery quality. On the contrary, our method employs features extracted from input images as the initialization for optimization. Guo et al. [2020] develop a material-specific version of StyleGAN2 [Karras et al. 2020] that can alternatively optimize the intermediate latent vector and the noise vector. Meanwhile, it serves as a realistic material generator. However, they do not directly extract the features from input images.

2.2 Single-image SVBRDF recovery

Another group of works only uses a single image as input. Single image material recovery can be divided into two categories: direct prediction and optimization-based. Most of previous works are direct prediction methods. Aittala et al. [2016] employ Convolutional Neural Network (CNN) to extract neural Gram-matrix descriptor from a single image to acquire the SVBRDF maps of stationary textured materials. Li et al. [2017] use a self-augmented convolutional neural network training strategy to solve the problem of insufficient data labels. Deschaintre et al. [2018] design a U-Net network combined with a global branch to process global and local features of the input image. Vecchio et al. [2021] treat the SVBRDF recovery as an image translation task and reduce the domain shift between synthetic and real data distributions in an unsupervised way. Zhou et al. [2021] train an adversarial framework on both synthetic and real data to improve the generalization ability. Guo et al. [2021] propose highlight-aware convolution to predict the saturated pixels with adversarial training loss. Since these approaches obtain the SVBRDF maps by a forward pass through the network, there is no optimization, leading to an apparent difference between renderings of the reconstructed maps and the input images.

There are a few works based on optimization. Henzler et al. [2021] propose a method for fine-tuning a pre-trained network to enhance its performance on a test example. However, their approach is limited to stationary isotropic materials and needs multiple carefully captured photos of the same scene. Zhou et al. [2022] employ meta-learning, combining the testing process with the training process, in addition to using an auxiliary network to assess the quality of material recovery. Nevertheless, their training process is memory intensive and unstable. Wang et al. [2023] propose to predict basis materials and their blending weights, by treating the estimated SVBRDF as the linear combination of basis materials. Zhang et al. [2023] propose to learn the lighting pattern of a planar light source and optimize the lighting pattern. Sartor et al. [2023] employ the diffusion model to reconstruct SVBRDF maps under several different lighting conditions, but their reconstruction results have trouble generating pixel-perfect reproductions. Luo et al. [2024] use recurrent neural network to update reflectance parameters given reconstruction likelihoods. Unlike these methods, our framework employs latent space optimization and per-map adapters for fine-tuning during optimization, enabling high capability and flexibility.

Recently, a number of methods for material editing [Guerrero-Viu et al. 2024; Zhou et al. 2022] and generation [Vecchio et al. 2023; Xin et al. 2024; Zhou et al. 2023] based on generative adversarial network [Karras et al. 2020] and diffusion model [Rombach et al. 2022] have also received great attention. Furthermore, some methods are proposed to tackle multiple resolution [Kuznetsov 2021] or high-resolution [Guo et al. 2023; Rodriguez-Pardo et al. 2023] SVBRDF recovery. Instead of per-pixel map recovery, several works [Garces et al. 2023; Jin et al. 2022; Shi et al. 2020] resort to procedural material models, which can act as a strong prior to constrain the materials, at the cost of limited expressive ability.

2.3 Graph Convolutional Network

Graph Convolutional Networks [Kipf and Welling 2016] have proven to be effective in extending the concept of convolution to graphs, enhancing the representation of nodes and edges for improved scalability and feature extraction capability in large-scale graphs. Several variants of GCN [Chen et al. 2018; Hamilton et al. 2017; Veličković et al. 2017] have been developed to address different aspects of the model. ImageGCN [Mao et al. 2022] models the image as the node of the directed graph and the different relationships between the images as the edge, which serves as one of our inspirations. Different from ImageGCN, we model the relationship of the input images as a complete undirected graph. In addition, we propose a learnable matrix to represent the relationship between images.

3 OUR METHOD

Similar to previous works ([Guo et al. 2020], [Guo et al. 2021]), our work aims to recover material properties from a single image or several images taken from nearly planar surfaces, where the images are captured with a collocated light/view configuration. For the shading model, we use the Cook-Torrance microfacet BRDF [Cook and Torrance 1982] with the GGX [Walter et al. 2007] normal distribution function. Therefore, every SVBRDF can be represented as four maps: diffuse albedo k_d, specular albedo k_s, roughness r, and surface normal n.

In this paper, our aim is to alleviate the ambiguity among the maps. For that, we propose a correlation-aware encoder-decoder framework (see Fig. 2) that consists of a correlation-aware encoder (Sec. 3.1) and an adapter-equipped decoder (Sec. 3.2) for SVBRDF map recovery. We first train the network on the entire training set. Then, we introduce a new optimization strategy (backpropagation in a single material), combining the feature-embedded latent space optimization and the fine-tuning of the adapter per map to achieve high-quality SVBRDF maps (Sect. 3.3). "Latent space optimization" involves searching for the latent vector within the latent space, while "fine-tuning" updates the adapter parameters under the supervision of rendering loss.

3.1 Correlation-aware encoder

Our intuition is that there is a correlation among images of the same material in different views. And this correlation can be proven: the correlation of the radiance (or pixel color) of the same point across different input images can be modeled by the BRDF (f), as the material properties and light intensity are identical, while the main difference lies in the incoming/outgoing directions (ω_i and ω_o). Note that ω_i and ω_o are identical due to the camera/light collocated assumption. Thus, the input images are related to the viewpoints. Establishing this correlation can help the network capture the change in material attributes with incoming directions. For

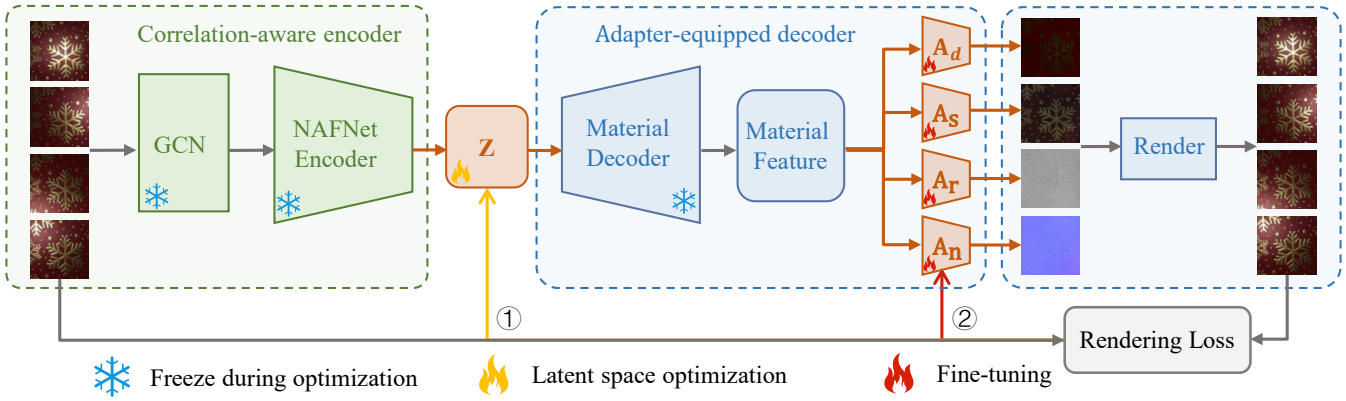

Figure 2: Structure: our network has an encoder-decoder structure, where the encoder consists of a graph convolutional network to learn correlation among the input images, followed by an encoder from Nonlinear Activation Free Network (NAFNet) to encode features into a latent vector z and the decoder includes a material decoder, together with several map adapters to output SVBRDFs. Training: the network is trained end-to-end with the rendering loss and map loss, and all the components are updated. Optimization: During the optimization, the network is optimized for each material with the rendering loss. The input images are fed into the encoder to obtain an initialized latent vector z_0, and then the decoder performs the latent space optimization (①) starting from z_0 for several iterations. Later, the map adapters are fine-tuned (②) for 1K iterations with the found latent vector (frozen) and frozen material decoder to output the final SVBRDFs.

example, the specular region has a sensitive response to the lighting changes, while the diffuse region is less sensitive. Recognizing these areas will provide useful clues for SVBRDF recovery. Inspired by ImageGCN [Mao et al. 2022], we introduce a graph convolutional network [Kipf and Welling 2016] to characterize this correlation.

Background of GCN. GCNs [Kipf and Welling 2016] has been proposed to learn the correlations between entities or nodes in a graph. An example is the social network, where each node is a person, and the edge between two nodes represents the relationship between two persons. By leveraging both node attributes and the relationships with neighboring nodes in the graph, the GCN is capable of learning robust and informative node embeddings. The node representations are learned with the layer-wise propagation rule:

$$H^{l+1} = \sigma(\tilde{D}^{-\frac{1}{2}}\tilde{A}\tilde{D}^{-\frac{1}{2}}H^{l}W^{l}), \tag{1}$$

where H^l is the feature matrix at the l^{th} layer. $\tilde{A} = A + E$ is the adjacency matrix A together with identity matrix E for self-connection. $\tilde{D}$ is a diagonal matrix, where $\tilde{D}_{ii} = \sum_j \tilde{A}_{ij}$. $\tilde{D}^{-\frac{1}{2}}\tilde{A}\tilde{D}^{-\frac{1}{2}}$ is a symmetrically normalized adjacency matrix, W^l is a layer-specific weight matrix, and $\sigma(\cdot)$ denotes an activation function. Note that the propagation rule defined in Eqn. (1) is considered a type of Laplacian smoothing [Li et al. 2018].

GCNs were introduced into the image domain by ImageGCN [Mao et al. 2022] later. The graph nodes represent the images, and the edges represent the relationships between the images.

Correlation-aware encoder. Inspired by ImageGCN, we also introduce a GCN into our framework to characterize the input image correlation, by treating channel features from each image as a graph node, and learning the adjacency matrix between these nodes, as shown in Fig. 3. In this way, the feature that represents the variation

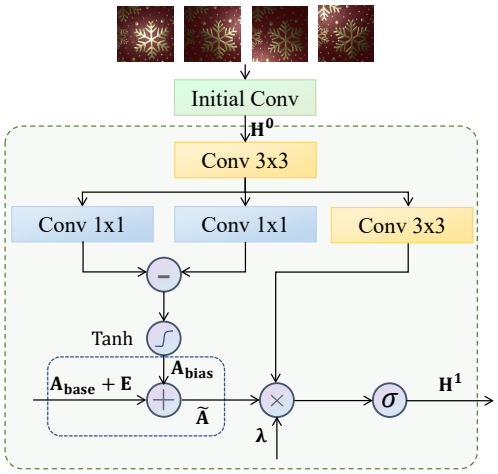

Figure 3: The architecture of GCN. We first extract the features of the input images using an initial convolution layer and concatenate them into the channel dimension. We then aggregate the features by our designed propagation rule defined in Eqn. (2).

at each pixel under different lighting conditions are learnt to help with the SVBRDF recovery.

We make several key changes to the GCN to suit the SVBRDF recovery task. First, we model the channel features of the captured image after initial convolution as the nodes of a complete undirected graph, rather than the directed graph in ImageGCN, since each image has the same impact on other input images. Second, the adjacency matrix between different channels is not explicit, unlike ImageGCN [Mao et al. 2022]. Hence, we propose learning this matrix rather than directly setting it to a specific input. Specifically,

we first encode the input images into an initial feature H^0 (with 64 channels) by a convolutional layer and then learn features with our designed propagation rule:

$$\begin{aligned} H^0 &= \text{Conv}(I_1, I_2, ...I_N), \\ H^1 &= \sigma(\lambda \tilde{A} H^0 W^0), \\ \tilde{A} &= A_{\text{base}} + E + A_{\text{bias}}. \end{aligned} \tag{2}$$

Here, the adjacency matrix consists of A_{base} and A_{bias}, where A_{base} is set as a 64×64 matrix with all diagonal zeros and all other elements one, and A_{bias} is learned during training. λ is a learnable normalization coefficient, and $\sigma(\cdot)$ is ReLU in our implementation. We set N as 4 and the light conditions of input images are random in practice.

Then, the features extracted with GCN are further encoded by an encoder into the latent vector z:

$$z = \text{Encoder}(H^1). \tag{3}$$

The choice of the encoder is crucial to the recovery quality. Thus, we choose the advanced image-to-image network using NAFNet [Chen et al. 2022]. At the core of their model is the layer normalization and the advanced channel attention mechanism. The former effectively stabilizes and accelerates network training while avoiding the artifacts from Batch Normalization (BN) [Gao et al. 2019; Guo et al. 2021] and difficulty of non-local information preservation from Instance Normalization (IN) [Guo et al. 2021] at the same time. The latter has been widely applied in various tasks in computer vision and natural language processing with great success. The details of NAFNet encoder are as shown in the supplementary.

Discussion. With our GCN and the designed propagation rule, channels that vary more with incoming direction retain stronger responses in the learned adjacency matrix. In this way, the GCN can focus more on regions that has a sensitive response to the lighting changes, helping to mitigate the ambiguity. Additionally, the GCN aggregates features from these channel features and effectively capturing the spatial and contextual relationships between different images. This process enhances the feature representation by integrating information across the graph, leading to more accurate recovery of the SVBRDF.

3.2 Adapter-equipped decoder

With the features extracted by our specialized encoder, we also need a decoder to produce the SVBRDF maps. One straightforward solution is to use a single decoder with the same structure as the NAFNet encoder. However, we notice a typical issue of this solution as previous works [Gao et al. 2019; Guo et al. 2020]: although the albedo map achieves enough details, the normal map is over-smooth, and the roughness map is fuzzy. The main reason behind this is that the SVBRDF maps are produced by a single decoder, which lacks the flexibility to adjust each map. Using a more extensive network might improve the recovery quality, but there is a risk of over-fitting. Alternatively, we propose a simple solution to solve this issue. Besides the main decoder, we introduce four tiny map adapters into our framework, allowing fine-tuning during optimization. This design can guarantee both consistency and flexibility: the prior material decoder is used to produce the material feature,

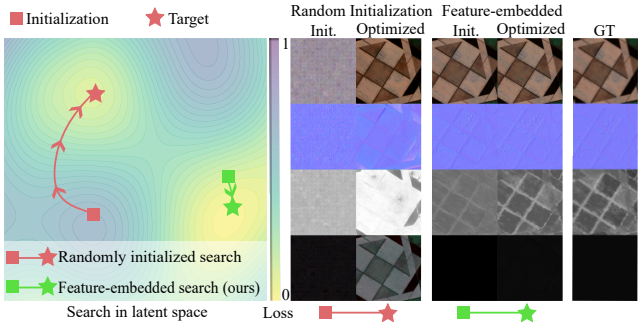

Figure 4: Comparison between the random-initialized latent space optimization (red line) and our feature-embedded optimization (green line) in a hypothetical latent space behinds Eqn. (6), where the brighter color indicates a lower loss. We use random values defined by a normal distribution for initialization, same as Guo et al. [2020]. Here, we show two local minimal points (shown as stars), while the right one (green star) is a better solution. With the feature-embedded initialization, it is more likely to converge to the optimal, while the optimization with a random initialization gets stuck in the local optimum.

ensuring consistency across maps, while fine-tuning four adapters enhances flexibility.

Specifically, we add four tiny map adapters to the main decoder, so-called *material decoder* (comes from NAFNet) to output each map, which is formulated as:

$$\begin{aligned} H &= \text{Decoder}(z), \\ k_d, k_s, r, n &= \text{A}_\text{d}(H), \text{A}_\text{s}(H), \text{A}_\text{r}(H), \text{A}_\text{n}(H), \end{aligned} \tag{4}$$

where H denotes the material feature map, which has the same resolution as the input image and has nine channels. Here, each adapter consists of a basic block from NAFNet, with about $\frac{1}{160}$ of the FLOPs of the material decoder. The output channels of four adapters are set to the channel count of each map.

Training. The correlation-aware encoder and adapter-equipped decoder form the complete network structure. We train our network with a joint loss function, consisting of a map loss $\mathcal{L}_{\text{map}}$ and a rendering loss $\mathcal{L}_{\text{render}}$:

$$\mathcal{L} = \lambda_1 \mathcal{L}_{\text{map}} + \lambda_2 \mathcal{L}_{\text{render}}, \tag{5}$$

where $\mathcal{L}_{\text{map}}$ is the L_1 loss on the SVBRDF maps, and $\mathcal{L}_{\text{render}}$ is the L_1 loss on nine different-view renderings similar to previous works [Deschaintre et al. 2018; Gao et al. 2019]. λ_1 and λ_2 are the weights to balance each loss, set as $\lambda_1 = 10$ and $\lambda_2 = 1$ in practice.

Discussion. In contrast to fine-tuning the adapters, directly fine-tuning the material decoder breaks the network prior and impairs consistency among SVBRDF maps. We provide the visual comparison in the supplementary. Our adapter-equipped decoder strategy can also be applied in other optimization-based works [Gao et al. 2019; Guo et al. 2020] to improve the recovery quality.

3.3 Feature-embedded latent space optimization

With the trained network, there are two typical ways to use it for SVBRDF recovery: latent space optimization and direct prediction. The latent space optimization can guarantee a better match with the input images, although it leads to a longer optimization time. One key question of latent space optimization is the initialization of the latent vector, as an adequate initialization could lead to a lower chance of local minimum. For this, we propose feature-embedded latent space optimization.

Feature-embedded latent space optimization. We use the trained correlation-aware encoder to encode the input images into a latent vector z and then perform optimization of the latent vector z in the latent space:

$$z^* = \text{argmin}_z \sum_{i=1}^{N} \mathcal{L}(R(M), I_i), \tag{6}$$

where M denotes four SVBRDF maps computed by Eqn. (4), $R(\cdot)$ is the rendering operator, and $\mathcal{L}$ is the L_2 loss. Note that the encoder is performed only once to obtain the initial latent vector, and then only the decoder is inferred (and frozen) every iteration to optimize.

We show a diagram of our feature-embedded latent space optimization in Fig. 4, by comparing against the random-initialized optimization. The random-initialized optimization gets stuck in the local optima and produces ambiguous SVBRDFs. In contrast, with the feature-embedded initialization, it is more likely to converge to a better solution and achieves a high-fidelity result.

Fine-tuning of four map adapters. As described in Sec. 3.2, for sufficient updates, after feature-embedded latent space optimization, the four map adapters are fine-tuned:

$$\theta^* = \text{argmin}_\theta \sum_{i=1}^{N} \mathcal{L}(R(M), I_i), \tag{7}$$

where M is four SVBRDF maps computed by Eqn. (4) and θ denotes the parameters of the four adapters. The per-map adapter component and fine-tuning enable finer and more accurate recovery, as shown in Fig. 6.

4 IMPLEMENTATION DETAILS

Data preparation. To train our network, we use the dataset by Deschaintre et al. [2018], which consists of 200,000 SVBRDFs. We randomly select about 96,000 SVBRDFs as our training dataset. The SVBRDFs are rendered into images with the Cook-Torrance model under several random light configurations, similar to previous work [Gao et al. 2019]. All input images are randomly cropped to 256×256. Each SVBRDF has 9 channels (3 for diffuse, 2 for normal, 1 for roughness, and 3 for specular).

Network training and optimization. We implement our model and train the network using the Adam optimizer in Pytorch [Paszke et al. 2017] with a fixed learning rate of $1e^{-5}$. All other hyper-parameters are set to Pytorch's default values. The network is trained with a batch size of 6 per GPU for 100 epochs, and it takes about 3 days on four RTX 3090 graphics cards. We optimize the latent vector 10 epochs and execute fine-tuning 1000 epochs.

Choice of the input image number. Our network requires a fixed number of input images and needs to be retrained on different ones. We experiment with three settings (single/four/eight images). Note that the light conditions of the input images are not fixed, exactly following the previous works (e.g., Gao et al. [2019]).

5 RESULTS

In this section, we compare our model with previous works with a single image or multiple images as inputs. For all of these methods, we use the source code and pre-trained models from the authors' websites, and both our model and theirs use the same dataset [Deschaintre et al. 2018] for training except MatFusion [Sartor and Peers 2023], which has a more extensive and diverse synthetic dataset than Deschaintre et al. [2018]. We use Mean Square Error (MSE), Root Mean Square Error (RMSE) and LPIPS [Zhang et al. 2018] metrics to measure the difference between maps/renderings and their ground truth. We only provide subset of the results in the paper, and more results are shown in the supplementary and video.

Table 1: Reconstruction and rendering error comparison between our method (with different variants) and previous works on 300 SVBRDFs from the dataset of Deschaintre et al. [2018]. These SVBRDFs are never used in training and none of the materials crossed in the training set. We evaluate the quality of renderings with RMSE and LPIPS and the quality of estimated maps with RMSE. Here, (D,N,R,S) means the four maps (diffuse, normal, roughness, specular). The re-renderings (Ren.) for each estimated SVBRDF are performed on 20 random light directions and evaluated by RMSE and LPIPS. The lowest errors are highlighted in bold.

		RMSE					LPIPS
Methods	Inputs	D	N	R	S	Ren.	Ren.
DIR	4	0.063	0.028	0.113	0.062	0.069	0.162
MaterialGAN		0.055	0.032	0.078	0.057	0.062	0.150
Ours		**0.024**	**0.015**	**0.061**	**0.016**	**0.032**	**0.108**
w/o GCN		0.027	0.027	0.069	0.035	0.036	0.128
w/o PMFT		0.025	0.024	0.072	0.033	0.034	0.121
max-pooling		0.039	0.031	0.078	0.048	0.046	0.138
DIR	1	0.072	0.064	0.124	0.069	0.119	0.186
MaterialGAN		0.066	0.065	0.116	0.080	0.112	0.191
LAT		0.054	0.048	0.091	0.065	0.074	0.174
DeepBasis		0.041	0.055	0.088	0.063	0.066	0.162
Ours		**0.030**	**0.034**	**0.082**	**0.045**	**0.055**	**0.145**

5.1 Comparison on multi-image SVBRDF recovery

We compare our model with Deep Inverse Rendering (DIR) by Gao et al. [2019] and MaterialGAN by Guo et al. [2020] on both synthetic images and captured photos. For all the methods, we take four random view images as inputs. For DIR, the model by Deschaintre et al. [2019] is used as an initialization, following Guo et al. [2020]. Regarding the optimization time, it takes 15 and 3 minutes for DIR and MaterialGAN, while our method only costs one minute on the same device.

In Table 1, we compare our method against DIR and MaterialGAN numerically on a set of 300 unseen (i.e., never used in training and none of the materials crossed in the training) synthetic SVBRDFs from Deschaintre et al. [2018]. Our method outperforms other methods in estimated maps and renderings, as indicated by RMSE and

Table 2: Numerical comparison between our method and others on real data with four images (top) and a single image (bottom) as input. RMSE and LPIPS are computed between the re-rendered and the reference images.

Inputs	Method	RMSE	LPIPS
4	DIR	0.114	0.224
	MaterialGAN	0.121	0.240
	Ours	**0.087**	**0.152**
1	DIR	0.138	0.272
	MaterialGAN	0.132	0.261
	MatFusion	0.129	0.231
	LAT	0.119	0.232
	DeepBasis	0.111	0.224
	Ours	**0.104**	**0.204**

LPIPS metrics. In Table 2, we conduct comparisons with DIR and MaterialGAN on captured photos. These samples are from Guo et al. [2020] and Zhou et al. [2022]. Our method achieves the highest quality numerically.

We also provide visual comparison with DIR and MaterialGAN, as shown in Fig. 5. For synthetic data, the SVBRDFs recovered by DIR and MaterialGAN show obvious artifacts, where the former has polluted maps, and the latter suffers from over-smooth maps. Our estimated SVBRDFs are the closest to the ground truth. For real data, the renderings from DIR and MaterialGAN exhibit different highlights from the input images, while the renderings of our method at both views agree more closely with the given images.

5.2 Comparison on single image SVBRDF recovery

As for the single input image, we compare our method with DIR, MaterialGAN, Look-Ahead Training (LAT) by Zhou et al. [2022] and DeepBasis by Wang et al. [2023] on both synthetic images and captured photos. We compare against MatFusion [Sartor and Peers 2023] on real data. Note that we train our network without GCN in this comparison, as there is a single image as input.

The numerical comparison is provided in Tables 1 and 2, showing that our method achieves the highest quality on both synthetic and real data in terms of RMSE and LPIPS metrics. The visual comparison is shown in Fig. 7. For synthetic data, the SVBRDFs estimated by previous works have problematic decomposition issues, leading to low-quality renderings. Our method outperforms these methods on both the SVBRDFs and renderings. Regarding the captured image, all the methods have a good agreement between renderings and the input image. However, previous works have apparent highlight burn-in in the recovered SVBRDF maps. As a result, the renderings of our method at the novel view have higher quality than previous works, thanks to the design in our network to alleviate ambiguity.

5.3 Experiments on anisotropic materials

We further validate the effectiveness of our approach on anisotropic material dataset from Ma et al. [2023], which consists of a thousand anisotropic SVBRDF maps with a resolution of 1024×1024. These anisotropic SVBRDF maps are rendered with an anisotropic GGX model [Walter et al. 2007] and each SVBRDF has 14 channels (3 for diffuse, 3 for normal, 2 for anisotropic roughness, 3 for tangent, and 3 for specular). We modify our network slightly to enable anisotropic materials, by setting the channel of material feature (Fig. 2) as 14 and using five map adapters. To train our network, we randomly select 900 SVBRDFs as our training set and use the remainings as the testing set. Then we augment the dataset by generating 256×256 crops at random positions, scales and rotations, leading to 20,000 SVBRDFs in total. We train our network on the training set with four images as input (N=4).

Table 3: Comparison between our method, DIR and MaterialGAN on eight images as inputs on 300 synthetic samples.

	RMSE					LPIPS
Methods	D	N	R	S	Ren.	Ren.
DIR	0.054	0.027	0.097	0.053	0.060	0.152
MaterialGAN	0.048	0.032	0.073	0.042	0.055	0.143
Ours	**0.021**	**0.015**	**0.039**	**0.015**	**0.025**	**0.098**

In Fig. 8, we show the recovered SVBRDF maps and the renderings of the input and novel views. We also provide the ground-truth maps and renderings. Since there is no pre-trained model on this dataset, we do not perform any comparison. The renderings closely match the GTs both visually and quantitatively, which confirms the powerful ability on the anisotropic materials.

5.4 Ablation study

The effect of the GCN. We validate the influence of the GCN in Table 1 and Fig. 6 by comparing our models designed without GCN and with GCN. We find that GCN highly improves the reconstruction quality for materials with highlights. This observation is reasonable since the correlation between the input images matters more for specular than diffuse materials. In addition, we conduct comparison between GCN and max-pooling applied in Deschaintre et al. [2019]. The number of parameters of these two variants is identical for fairness. As shown in Table 1, replacing GCN with max-pooling yields SVBRDF maps that have higher RMSE and LIPIPS errors. We provide more visual results on ablation of GCN and comparisons between GCN and max-pooling in the supplementary.

The effect of adapters and the latent space optimization. In Table 1 and Fig. 6, we compare our models without and with the adapters. A clearer normal map and a more detailed roughness map are recovered for sufficient updates to four maps thanks to the fine-tuning of adapters. Then we verify the influence of the latent space optimization by comparing our model (w/o optimization) and our model (w/ optimization). As shown in Fig. 9, introducing the latent space optimization improves the details of recovered SVBRDF maps, and leads to a lower MSE loss. When there is no latent space optimization, although the per-map adapter has some flexibility to update the SVBRDF maps, its capability is constrained by the latent vector. Therefore, both the latent space optimization and the per-map adapter fine-tuning are needed. More visual comparisons on adapters are shown in the supplementary.

The number of input images. In Table 3, we conduct a comparison with DIR and MaterialGAN on eight images as inputs. The errors in both reconstructed SVBRDF maps and novel-view renderings

decrease as the number of input images increases, as expected. Our method still outperforms the other methods in terms of RMSE and LPIPS. We provide visual comparisons of our method against MaterialGAN with different inputs (1, 4, 8) in the supplementary.

5.5 Discussion and limitations

We have identified several limitations of our method. First, our model needs fixed number of input images, since we need to extract features from them. It makes our method less flexible compared to Gao et al. [2019] and Guo et al. [2020], which support an arbitrary number of input images. Second, despite the high performance for SVBRDF recovery, our model can not be used for generations, unlike the previous StyleGAN2-based methods [Guo et al. 2020]. Lastly, our method might exhibit lower quality for highly-specular materials with a single input image than Guo et al. [2021] as shown in Fig. 10, since we do not have any specialized design to handle highlights. However, the highlight-aware design by Guo et al. [2021] is orthogonal to our contribution, which can be combined with our method to improve the performance further.

6 CONCLUSION

In this paper, we have presented a novel framework for high-fidelity SVBRDF acquisition. The core of our framework is a correlation-aware encoder and adapter-equipped decoder. We employ graph convolutional network to model the correlation features among the input images and adapters to the material decoder interpret and fine-tune individual maps. Our framework allows feature-embedded latent space optimization and fine-tuning of four map adapters. Thanks to these components, our framework has achieved state-of-the-art performance on both synthetic and real materials.

There are still many potential future researching directions. Despite the high capability of recovery, our model can not be used for generation. However, the core idea of the feature-embedded latent space optimization can be further used in a generative model for both material recovery and generation. Another interesting potential work is using material priors from the current existing large language models for material recovery.

ACKNOWLEDGMENTS

We thank the reviewers for the valuable comments. This work has been partially supported by the National Science and Technology Major Project under grant No. 2022ZD0116305 and National Natural Science Foundation of China under grant No. 62172220.

REFERENCES

Miika Aittala, Timo Aila, and Jaakko Lehtinen. 2016. Reflectance modeling by neural texture synthesis. *ACM Transactions on Graphics (TOG)* 35, 4 (2016), 65.

Miika Aittala, Tim Weyrich, Jaakko Lehtinen, et al. 2015. Two-shot SVBRDF capture for stationary materials. *ACM Trans. Graph.* 34, 4 (2015), 110–1.

Pontus Andersson, Jim Nilsson, Tomas Akenine-Möller, Magnus Oskarsson, Kalle Åström, and Mark D Fairchild. 2020. FLIP: A Difference Evaluator for Alternating Images. *Proc. ACM Comput. Graph. Interact. Tech.* 3, 2 (2020), 15–1.

Manmohan Chandraker. 2014. On shape and material recovery from motion. In *European Conference on Computer Vision*. Springer, 202–217.

Jie Chen, Tengfei Ma, and Cao Xiao. 2018. Fastgcn: fast learning with graph convolutional networks via importance sampling. *arXiv preprint arXiv:1801.10247* (2018).

Liangyu Chen, Xiaojie Chu, Xiangyu Zhang, and Jian Sun. 2022. Simple baselines for image restoration. In *European Conference on Computer Vision*. Springer, 17–33.

Robert L Cook and Kenneth E Torrance. 1982. A reflectance model for computer graphics. *ACM Transactions on Graphics (TOG)* 1, 1 (1982), 7–24.

Valentin Deschaintre, Miika Aittala, Fredo Durand, George Drettakis, and Adrien Bousseau. 2018. Single-image svbrdf capture with a rendering-aware deep network. *ACM Transactions on Graphics (TOG)* 37, 4 (2018), 128.

Valentin Deschaintre, Miika Aittala, Frédo Durand, George Drettakis, and Adrien Bousseau. 2019. Flexible svbrdf capture with a multi-image deep network. In *Computer graphics forum*, Vol. 38. Wiley Online Library, 1–13.

Duan Gao, Xiao Li, Yue Dong, Pieter Peers, Kun Xu, and Xin Tong. 2019. Deep inverse rendering for high-resolution SVBRDF estimation from an arbitrary number of images. *ACM Transactions on Graphics (TOG)* 38, 4 (2019), 134.

Elena Garces, Victor Arellano, Carlos Rodriguez-Pardo, David Pascual-Hernandez, Sergio Suja, and Jorge Lopez-Moreno. 2023. Towards Material Digitization with a Dual-scale Optical System. *ACM Transactions on Graphics (TOG)* 42, 4 (2023), 1–13.

D. Guarnera, G.C. Guarnera, A. Ghosh, C. Denk, and M. Glencross. 2016. BRDF Representation and Acquisition. *Computer Graphics Forum* 35, 2 (2016), 625–650.

Julia Guerrero-Viu, Milos Hasan, Arthur Roullier, Midhun Harikumar, Yiwei Hu, Paul Guerrero, Diego Gutierrez, Belen Masia, and Valentin Deschaintre. 2024. TexSliders: Diffusion-Based Texture Editing in CLIP Space. *arXiv preprint arXiv:2405.00672* (2024).

Jie Guo, Shuichang Lai, Chengzhi Tao, Yuelong Cai, Lei Wang, Yanwen Guo, and Ling-Qi Yan. 2021. Highlight-aware two-stream network for single-image SVBRDF acquisition. *ACM Transactions on Graphics (TOG)* 40 (2021), 1 – 14.

Jie Guo, Shuichang Lai, Qinghao Tu, Chengzhi Tao, Changqing Zou, and Yanwen Guo. 2023. Ultra-High Resolution SVBRDF Recovery from a Single Image. *ACM Transactions on Graphics* (2023).

Yu Guo, Cameron Smith, Miloš Hašan, Kalyan Sunkavalli, and Shuang Zhao. 2020. MaterialGAN: reflectance capture using a generative SVBRDF model. *ACM Transactions on Graphics (TOG)* 39, 6 (2020), 1–13.

Will Hamilton, Zhitao Ying, and Jure Leskovec. 2017. Inductive representation learning on large graphs. *Advances in neural information processing systems* 30 (2017).

Philipp Henzler, Valentin Deschaintre, Niloy J Mitra, and Tobias Ritschel. 2021. Generative modelling of BRDF textures from flash images. *arXiv preprint arXiv:2102.11861* (2021).

Zhuo Hui and Aswin C Sankaranarayanan. 2015. A dictionary-based approach for estimating shape and spatially-varying reflectance. In *2015 IEEE International Conference on Computational Photography (ICCP)*. IEEE, 1–9.

Wenhua Jin, Beibei Wang, Milos Hasan, Yu Guo, Steve Marschner, and Ling-Qi Yan. 2022. Woven fabric capture from a single photo. In *SIGGRAPH Asia 2022 conference papers*. 1–8.

Tero Karras, S. Laine, Miika Aittala, Janne Hellsten, J. Lehtinen, and Timo Aila. 2020. Analyzing and Improving the Image Quality of StyleGAN. *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* (2020), 8107–8116.

Thomas N Kipf and Max Welling. 2016. Semi-supervised classification with graph convolutional networks. *arXiv preprint arXiv:1609.02907* (2016).

Alexandr Kuznetsov. 2021. NeuMIP: Multi-resolution neural materials. *ACM Transactions on Graphics (TOG)* 40, 4 (2021).

Qimai Li, Zhichao Han, and Xiao-Ming Wu. 2018. Deeper insights into graph convolutional networks for semi-supervised learning. In *Proceedings of the AAAI conference on artificial intelligence*, Vol. 32.

Xiao Li, Yue Dong, Pieter Peers, and Xin Tong. 2017. Modeling surface appearance from a single photograph using self-augmented convolutional neural networks. *ACM Transactions on Graphics (TOG)* 36, 4 (2017), 45.

Xuejiao Luo, Leonardo Scandolo, Adrien Bousseau, and Elmar Eisemann. 2024. Single-Image SVBRDF Estimation with Learned Gradient Descent. In *Computer Graphics Forum (Proceedings of Eurographics)*, Vol. 43.

Xiaohe Ma, Xianmin Xu, Leyao Zhang, Kun Zhou, and Hongzhi Wu. 2023. OpenSVBRDF: A Database of Measured Spatially-Varying Reflectance. *ACM Trans. Graph* 42, 6 (2023).

Chengsheng Mao, Liang Yao, and Yuan Luo. 2022. Imagegcn: Multi-relational image graph convolutional networks for disease identification with chest x-rays. *IEEE Transactions on Medical Imaging* 41, 8 (2022), 1990–2003.

Adam Paszke, Sam Gross, Soumith Chintala, Gregory Chanan, Edward Yang, Zachary DeVito, Zeming Lin, Alban Desmaison, Luca Antiga, and Adam Lerer. 2017. Automatic differentiation in PyTorch. (2017).

Jérémy Riviere, Pieter Peers, and Abhijeet Ghosh. 2016. Mobile surface reflectometry. *Computer Graphics Forum* 35, 1 (2016), 191–202.

Carlos Rodriguez-Pardo, Henar Dominguez-Elvira, David Pascual-Hernandez, and Elena Garces. 2023. Umat: Uncertainty-aware single image high resolution material capture. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 5764–5774.

Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. 2022. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 10684–10695.

Sam Sartor and Pieter Peers. 2023. MatFusion: A Generative Diffusion Model for SVBRDF Capture. In *SIGGRAPH Asia 2023 Conference Papers*. 1–10.

Liang Shi, Beichen Li, Miloš Hašan, Kalyan Sunkavalli, Tamy Boubekeur, Radomir Mech, and Wojciech Matusik. 2020. Match: Differentiable material graphs for

procedural material capture. *ACM Transactions on Graphics (TOG)* 39, 6 (2020), 1–15.

Giuseppe Vecchio, Rosalie Martin, Arthur Roullier, Adrien Kaiser, Romain Rouffet, Valentin Deschaintre, and Tamy Boubekeur. 2023. ControlMat: A Controlled Generative Approach to Material Capture. *arXiv preprint arXiv:2309.01700* (2023).

Giuseppe Vecchio, Simone Palazzo, and Concetto Spampinato. 2021. Surfacenet: Adversarial svbrdf estimation from a single image. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 12840–12848.

Petar Veličković, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Lio, and Yoshua Bengio. 2017. Graph attention networks. *arXiv preprint arXiv:1710.10903* (2017).

Bruce Walter, Stephen R Marschner, Hongsong Li, and Kenneth E Torrance. 2007. Microfacet models for refraction through rough surfaces. In *Proceedings of the 18th Eurographics conference on Rendering Techniques*. 195–206.

Li Wang, Lianghao Zhang, Fangzhou Gao, and Jiawan Zhang. 2023. DeepBasis: Hand-Held Single-Image SVBRDF Capture via Two-Level Basis Material Model. In *SIGGRAPH Asia 2023 Conference Papers*. 1–11.

Linxuan Xin, Zheng Zhang, Jinfu Wei, Ge Li, and Duan Gao. 2024. DreamPBR: Text-driven Generation of High-resolution SVBRDF with Multi-modal Guidance. *arXiv preprint arXiv:2404.14676* (2024).

Zexiang Xu, Jannik Boll Nielsen, Jiyang Yu, Henrik Wann Jensen, and Ravi Ramamoorthi. 2016. Minimal BRDF Sampling for Two-Shot near-Field Reflectance Acquisition. *ACM Trans. Graph.* 35, 6 (Nov. 2016), 12.

Lianghao Zhang, Fangzhou Gao, Li Wang, Minjing Yu, Jiamin Cheng, and Jiawan Zhang. 2023. Deep SVBRDF Estimation from Single Image under Learned Planar Lighting. In *ACM SIGGRAPH 2023 Conference Proceedings*. 1–11.

Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang. 2018. The unreasonable effectiveness of deep features as a perceptual metric. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 586–595.

Xilong Zhou, Milos Hasan, Valentin Deschaintre, Paul Guerrero, Yannick Hold-Geoffroy, Kalyan Sunkavalli, and Nima Khademi Kalantari. 2023. PhotoMat: A Material Generator Learned from Single Flash Photos. In *ACM SIGGRAPH 2023 Conference Proceedings*. 1–11.

Xilong Zhou, Milos Hasan, Valentin Deschaintre, Paul Guerrero, Kalyan Sunkavalli, and Nima Khademi Kalantari. 2022. Tilegen: Tileable, controllable material generation and capture. In *SIGGRAPH Asia 2022 Conference Papers*. 1–9.

Xilong Zhou and Nima Khademi Kalantari. 2021. Adversarial Single-Image SVBRDF Estimation with Hybrid Training. In *Computer Graphics Forum*, Vol. 40. Wiley Online Library, 315–325.

Xilong Zhou and Nima Khademi Kalantari. 2022. Look-Ahead Training with Learned Reflectance Loss for Single-Image SVBRDF Estimation. *ACM Transactions on Graphics (TOG)* 41, 6 (2022), 1–12.

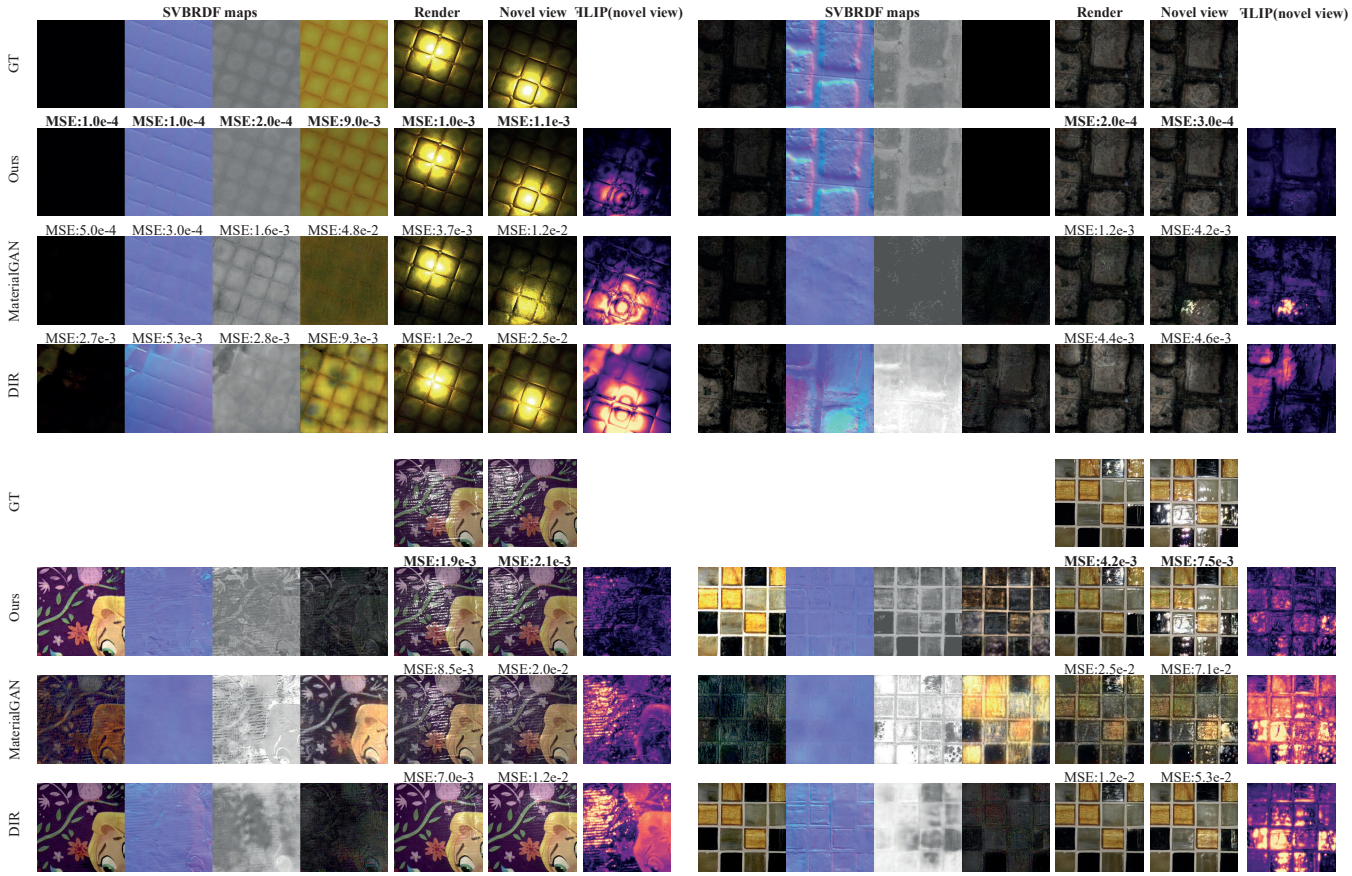

Figure 5: Comparison between our method, MaterialGAN and DIR on synthetic and real data, where the input image count is set as four. Our model outperforms the other methods in terms of the recovered SVBRDFs and renderings. We use the error map (ꟻLIP [Andersson et al. 2020]) to show the difference between the novel-view renderings and the reference images. The lowest error is marked in bold.

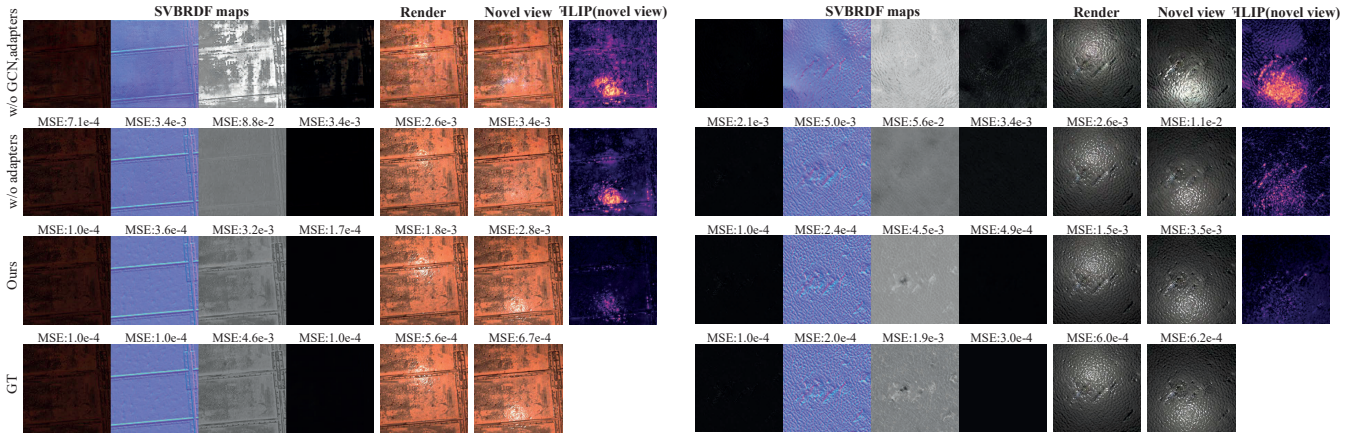

Figure 6: Ablation study of GCN and adapters. If without GCN and adapters, the estimated maps are coupled together and thus very different from GT. With GCN, the light bake-in issue has been greatly mitigated. With adapters, our framework avoids over-smooth maps and achieves better novel-view renderings.

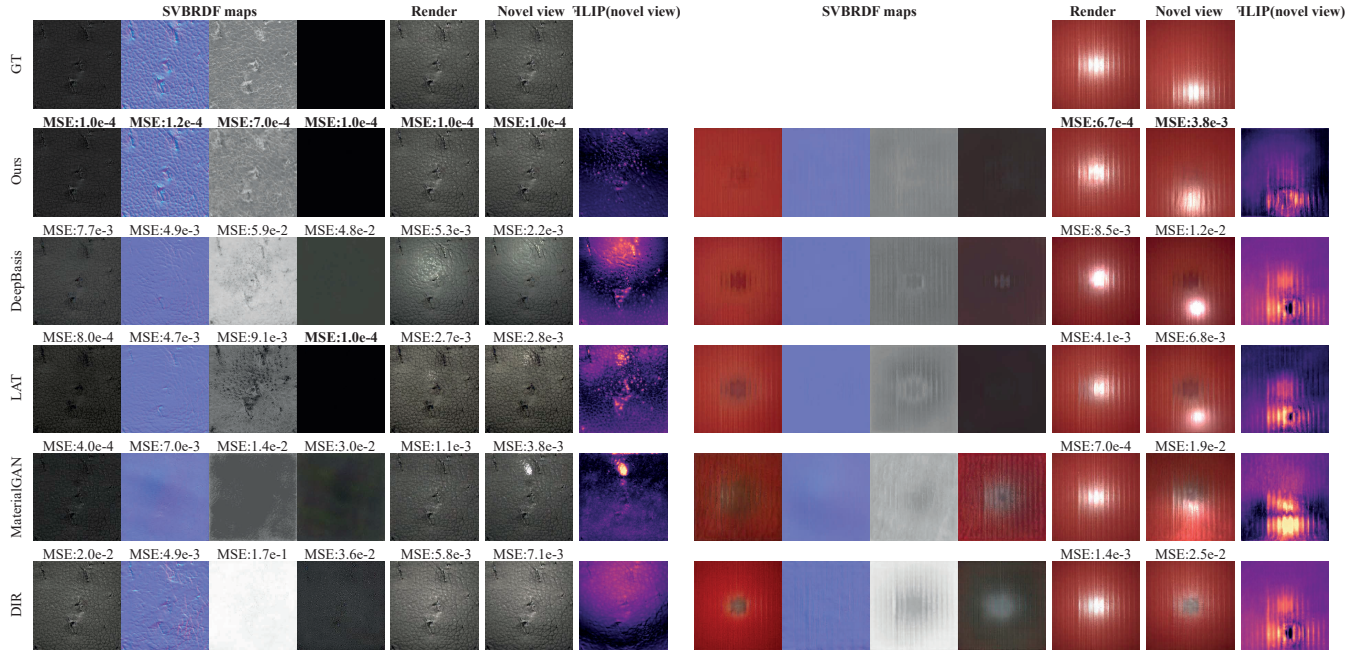

Figure 7: Comparison between our method, DIR, MaterialGAN, LAT and DeepBasis on synthetic and real data, with a single image as input. For the synthetic data, our model produces the closest SVBRDF maps in most cases, resulting in the highest-quality renderings at both input and novel views. For the real data, our method has less highlight burn-in than other methods, leading to the least error in the novel view renderings. We use ꟻLIP images to show the difference between the novel-view renderings and the reference images.

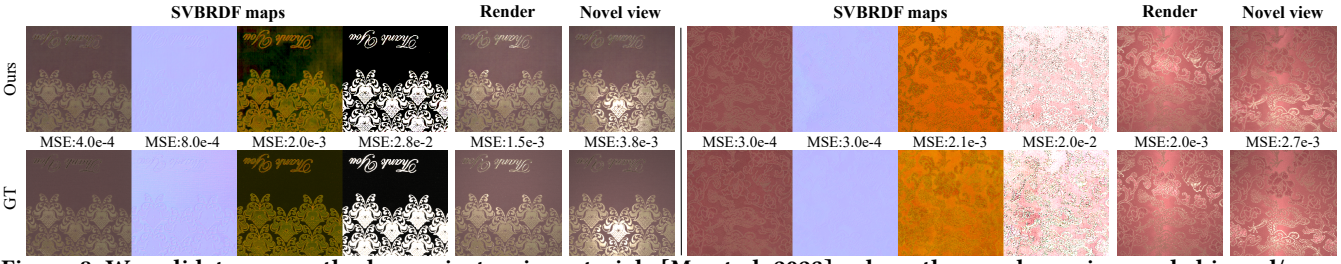

Figure 8: We validate our method on anisotropic materials [Ma et al. 2023], where the roughness is encoded in red/green channel, following OpenSVBRDF [Ma et al. 2023]. The renderings of both input and novel views can closely match the ground truth.

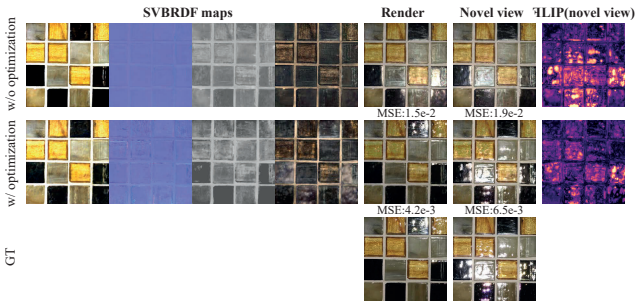

Figure 9: The influence of the latent space optimization. If without latent space optimization, SVBRDFs cannot be fully optimized and the highlights of the rendered image differ significantly from the reference image.

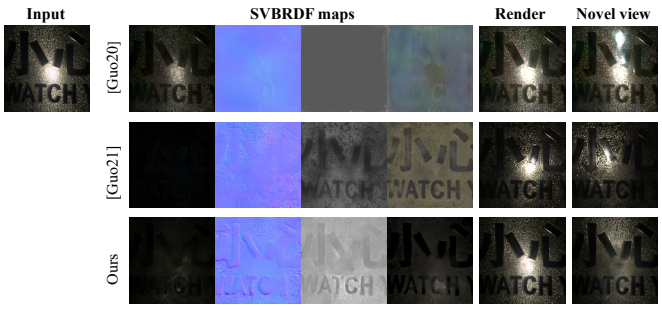

Figure 10: Failure case. Our method shows lower quality than Guo et al. [2021] for highly-specular materials with a single input image. Because there is no application of GCN for our single image recovery.