DOI: 10.1111/cgf.142635
EUROGRAPHICS 2021 / N. Mitra and I. Viola
(Guest Editors)

Volume 40 (2021), Number 2

Adversarial Single-Image SVBRDF Estimation with Hybrid Training

Xilong Zhou and Nima Khademi Kalantari

Texas A&M University

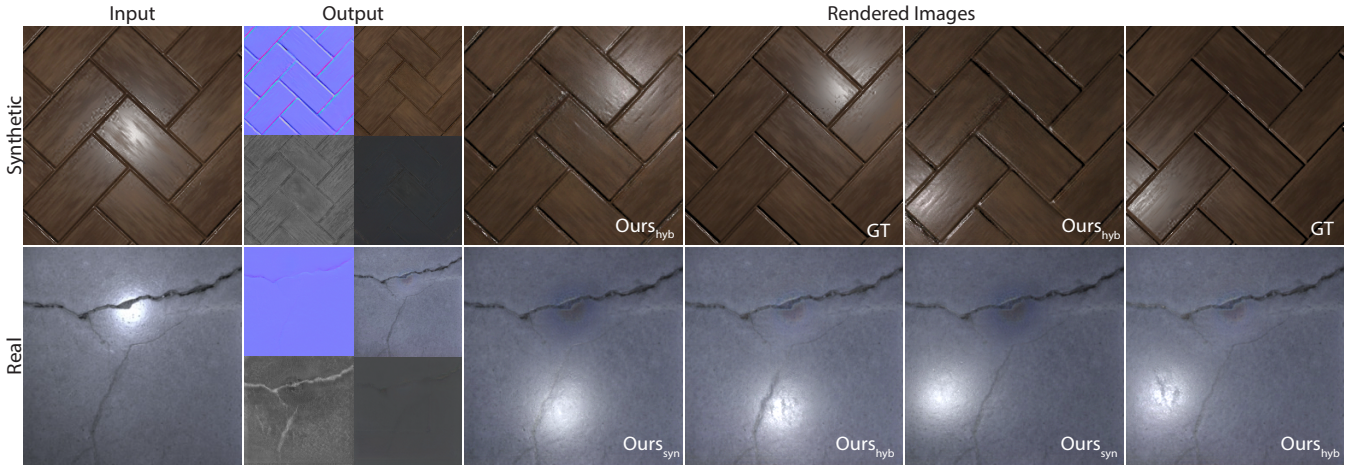

Figure 1: *We propose a novel adversarial framework to estimate SVBRDF parameters from a single image. We also introduce a simple strategy to train our network on real image pairs in addition to synthetically generated data. From the input images, shown on the left, our approach estimates a set of four parameters (middle). From top left to bottom right, the parameters are normal, diffuse albedo, roughness, and specular albedo. For the synthetic example (top row), our method produces a rendered image that closely matches the appearance of the ground truth. At the bottom, we show the results of our network trained only on synthetic and hybrid images. Compared to our synthetically trained network, our network trained on hybrid data is able to better capture the roughness of the scratches on the stone, producing a scratched appearance in the highlights.*

Abstract

In this paper, we propose a deep learning approach for estimating the spatially-varying BRDFs (SVBRDF) from a single image. Most existing deep learning techniques use pixel-wise loss functions which limits the flexibility of the networks in handling this highly unconstrained problem. Moreover, since obtaining ground truth SVBRDF parameters is difficult, most methods typically train their networks on synthetic images and, therefore, do not effectively generalize to real examples. To avoid these limitations, we propose an adversarial framework to handle this application. Specifically, we estimate the material properties using an encoder-decoder convolutional neural network (CNN) and train it through a series of discriminators that distinguish the output of the network from ground truth. To address the gap in data distribution of synthetic and real images, we train our network on both synthetic and real examples. Specifically, we propose a strategy to train our network on pairs of real images of the same object with different lighting. We demonstrate that our approach is able to handle a variety of cases better than the state-of-the-art methods.

CCS Concepts

• ***Computing methodologies*** → *Reflectance modeling; Image processing;*

1. Introduction

The appearance of real-world objects is the result of interactions between the light, geometries, and materials. Estimating the reflectance properties of a material from a single photograph, requires unraveling these complex interactions and, thus, is challenging. In recent years, several approaches have proposed to tackle this problem through deep learning [LDPT17; DAD*18; LSC18; YLD*18].

Specifically, these approaches train a convolutional neural network (CNN) to estimate the spatially-varying bidirectional reflectance distribution function (SVBRDF) from a single flash image of a planar surface.

However, these methods have two major problems. The first issue stems from the fact that all these techniques train their network by minimizing a pixel-wise loss (e.g., L1) between the estimated and ground truth SVBRDF parameters (and/or rendered images). However, since the problem of single-image SVBRDF estimation is ill-posed, there exist a large number of acceptable solutions. Therefore, forcing the network to adhere to the ground truth reduces its flexibility and negatively impacts the results. Second, since collecting a set of real images and their corresponding ground truth SVBRDF parameters is difficult, most of these methods [DAD*18; DAD*19; LSC18; LXR*18] train their system on synthetic images. Unfortunately, unlike real images, synthetic images are generated under perfect conditions. Therefore, the distribution of synthetic and real images are different, which limits the ability of these approaches to effectively generalize to real examples.

To address these issues, we propose an adversarial framework [GPM*14] for estimating SVBRDF parameters from a single image. Specifically, we use an encoder to extract a set of features from the input image and use four decoders to synthesize the SVBRDF parameters, i.e., normal, diffuse albedo, roughness, and specular albedo. To train the network, we use a set of five discriminators to evaluate the quality of the four estimated parameters as well as the re-rendered images. We condition all the discriminators on the input image to ensure the estimated parameters and re-rendered images follow the distribution of the input image.

Inspired by the hybrid intrinsic decomposition approaches [BKR18; LS18], we propose a mechanism to train our system on real images without ground truth SVBRDF parameters. Our key observation is that a pair of real images of the same object captured with different flash lights can be used for supervising the network. We use one of the images as the input and the other as the ground truth. Since the position of the light in the ground truth image is unknown, we estimate it through an auxiliary output. We use a discriminator, conditioned on the input image, to evaluate the quality of our rendered image. The hybrid training on both synthetic and real examples, helps our network to produce realistic results on real examples, as shown in Fig. 1. We show that our adversarial approach outperforms the state of the art on both synthetic and real images. In summary, our paper makes the following contributions:

- We propose an adversarial framework to address the ill-posed problem of single image SVBRDF estimation (Sec. 3).
- We propose a novel strategy to train our network on real image pairs without ground truth SVBRDF parameters (Sec. 4).
- We demonstrate that our approach outperforms state-of-the-art methods on both synthetic and real images.

2. Related Work

The problem of estimating the reflectance properties of real-world materials has been extensively studied in the past. A large number of approaches propose to do so using specialized hardware and many input images [GCHS09; WMP*06; DWT*10; GAHO07; GCP*10]. For brevity, we only focus on approaches that estimate reflectance parameters from a small number of images in the wild.

Aittala et al. [AWL*15] propose an approach to estimate the reflectance properties from a flash/no-flash image pairs. Zhou et al. [ZCD*16] present a sparse basis model to capture real-world materials from a few images of an object from different views. Hui et al. [HSL*17] propose an optimization strategy to reconstruct SVBRDF and normal from a set of collocated camera-flash images.

The recent approaches use a deep convolutional neural network (CNN) to handle this application. Aittala et al. [AAL16] propose an optimization-based neural texture transfer algorithm to estimate the SVBRDF parameters from a single image. However, their approach can only handle stationary textures and their output does not correspond to the input image.

Deschaintre et al. [DAD*18] propose a specifically designed network with a global and a local branch to estimate reflectance given an input image. Deschaintre et al. [DAD*19] propose several improvements and a flexible network architecture to be able to use arbitrary number of images as the input. Deschaintre et al. [DDB20] proposes a fine-tuning approach to improve the results on the test example at hand. Li et al. [LSC18] provides pixel coordinates along with the input image to the network to force the network to behave differently at different image locations. All these approaches use pixel-wise loss functions to train their network which limits the flexibility of the network in handling this underconstrained problem. Moreover, they use synthetically generated images for training, which have different data distribution compared to real-world examples.

Li et al. [LDPT17] attempt to address the gap in data distribution of synthetic and real images by proposing a self-augmented algorithm to train their network on a real images along with synthetic images. Ye et al. [YLD*18] improve this approach by proposing to replace the training on synthetic images with inexact supervision. The self-augmented training on real images in these methods, however, provides a weak supervision which is insufficient for producing high-quality results.

A couple of more recent techniques, pose the problem as energy optimization on the test example at hand. Gao et al. [GLD*19] propose to perform the optimization in the latent space of an encoder-decoder network. Guo et al. [GSH*20] propose to optimize the style and noise latent vector of the StyleGAN2 network [KLA*20]. While their approaches are highly effective when multiple images of the scene are provided as the input, they often struggle to produce high-quality results from a single input image, as the problem is highly underconstrained. Zhao et al. [ZWX*20] propose to estimate the diffuse map from the input image using a simple strategy and use it as ground truth. They then use the L1 loss between the estimated and ground truth diffuse map in combination with an adversarial loss on the rendered images to optimize the network. However, their method heavily relies on the initially estimated diffuse map. Moreover, their method is not able to handle surfaces with non-stationary texture. Additionally, the optimization in all of these approaches is time consuming and, thus, they are inefficient.

In contrast to all of these deep learning methods, we propose

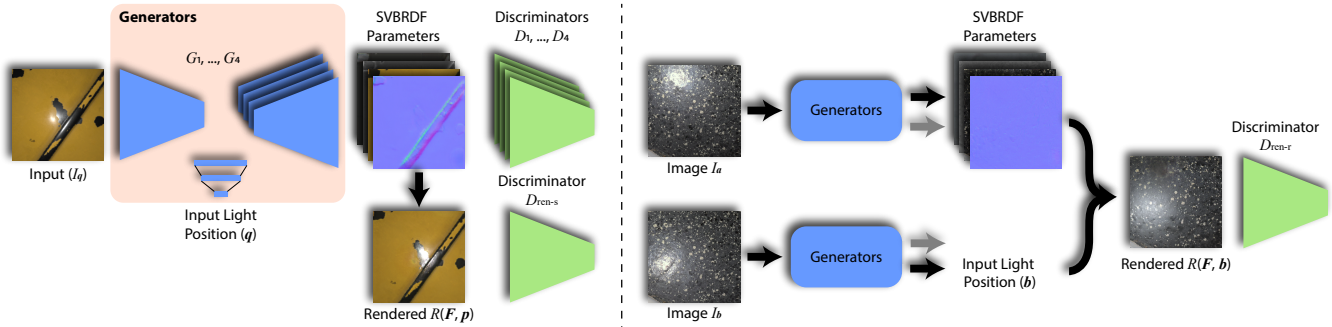

Figure 2: *On the left, we show an overview of our system for training on synthetic images with ground truth SVBRDF parameters. We use an encoder-decoder architecture with a shared encoder and four decoders to estimate the four parameters given an input image. We also use a set of fully connected layers to estimate the position of input light from the features extracted by the encoder. To train this network on synthetic images, we use a set of 4 discriminators to evaluate the quality of the estimated parameters. We also use an additional discriminator to estimate the quality of the rendered images using the estimated parameters. On the right, we show our novel strategy to train our generators on real image pairs without ground truth SVBRDF parameters. Given a set of two images of a real scene captured under two different lighting, we use our generators to estimate the parameters and light position. We then use the estimated parameters from image I_a and the estimated light position b to render an image. We use a discriminator to evaluate the quality of the rendered image with image I_b used as ground truth.*

an adversarial framework with a strong supervision on real images through a novel strategy to train the network on real image pairs.

3. Adversarial SVBRDF Estimation Framework

Given an input image I_q, captured with a flash light at position q, our goal is to recover four SVBRDF parameters, i.e., $F = \{F_i\}_{i=1}^4$ corresponding to normal, diffuse albedo, roughness, and specular albedo. We propose an adversarial framework to handle this application, as shown in Fig. 2. Specifically, we use a set of four generators, $G = \{G_i\}_{i=1}^4$, to estimate the SVBRDF parameters from the input image. Our generators are encoder-decoder convolutional neural networks (CNN) with shared encoder. In our system, the encoder is responsible for encoding the input image into a set of compact features. Each decoder then synthesizes an SVBRDF parameter from this feature representation. Note that, we use skip connections between the encoder and all the decoders to be able to preserve the high-frequency information. The goal of these generators is to produce results that are indistinguishable from ground truth. In addition to the four parameters, we also estimate the position of the light q from the encoded features through a set of fully connected layers G_l. This auxiliary output helps the network to understand the scene better and also is important for training the network on real images, as discussed in Sec. 4.

To effectively train the generators, we use a series of discriminators that are responsible for distinguishing the fake (estimated by the generators) from the real (ground truth) results. Specifically, we train our system by minimizing the following objective on a set of synthetically generated images:

$$E_{\text{syn}} = \sum_{i=1}^{4} E_i + \lambda_r E_{\text{ren}} + \lambda_l E_l, \quad (1)$$

where the first term E_i refers to the loss function for the four SVBRDF parameters. We also use a rendering loss E_{ren} to ensure the estimated feature maps are able to produce high-quality rendered images. Moreover, E_l ensures that our network can accurately estimate input light position. Finally, λ_r and λ_l are the weights for the rendering and light position loss terms and we set them to 5 and 10, respectively. Below we describe each loss in detail.

Parameter Loss We use this loss to enforce the estimated parameters $\hat{F}_i = G_i(I_q)$ to be indistinguishable from the ground truth parameters F. Note that, for synthetically generated data, we have access to the ground truth parameters. We define this loss as follows:

$$E_i = \mathcal{L}_{\text{adv}}(G_i, D_i) + \lambda_{1,p}\mathcal{L}_{\text{feat}}(G_i, D_i) + \lambda_{2,p}\|G_i(I_q) - F_i\|_1, \quad (2)$$

Here, $\mathcal{L}_{\text{adv}}(G_i, D_i)$ is the adversarial loss based on the least squared formulation (LSGAN) [MLX*17]. Our discriminator is conditional [IZZE17] and takes the input image in addition to the estimated or ground truth parameters. By minimizing this loss, the generator tries to estimate parameters that are indistinguishable from ground truth, while the discriminator tries to effectively distinguish the estimated parameter from ground truth.

We also use a feature matching loss [WLZ*18] to stabilize the training. This loss basically minimizes the L1 distance between the features at different layers of the discriminator, computed using the estimated and ground truth parameters. The last term in Eq. 2 is the L1 loss between the estimated parameter $\hat{F}_i = G_i(I_q)$ and ground truth F_i. This is a common term in adversarial frameworks and enforces the generator to not only fool the discriminator, but also produce results that are similar to the ground truth. Finally, λ_1 and λ_2 are the weight of the feature matching and L1 losses, respectively. We set $\lambda_{1,p} = 10$ and $\lambda_{2,p} = 10$ in our implementation.

Rendering Loss Although the parameter loss ensures that the network produces high-quality SVBRDF parameters, the estimated maps may not reproduce the appearance of the original material. Therefore, we use the rendering loss E_{ren} in Eq. 1 to ensure the estimated parameters can produce rendered images that are indistinguishable from the ground truth. Specifically, we use the estimated feature maps $\hat{F}$ along with a random light position p to render an image through the Cook-Torrance model [CT82] $R(\hat{F}, p)$ and minimize the following loss:

Figure 3: *We show three images illuminated with a flash light at different positions for six objects from our real training dataset.*

$$E_{\text{ren}} = \mathcal{L}_{\text{adv}}(G, D_{\text{ren-s}}) + \lambda_{1,r}\mathcal{L}_{\text{feat}}(G, D_{\text{ren-s}}) + \lambda_{2,r}\|R(\hat{F}, p) - I_p\|_1 + \lambda_{3,r}\mathcal{L}_{\text{vgg}}(R(\hat{F}, p), I_p), \tag{3}$$

where $\lambda_{1,r}$, $\lambda_{2,r}$ and $\lambda_{3,r}$ are the weight of the feature, L1 and VGG terms and we set them to 10, 50 and 10, respectively. Moreover, D_{ren} is a discriminator responsible for distinguishing the ground truth images from the ones rendered using the estimated features. Similar to the discriminators in the parameter loss, this is a conditional discriminator which takes the input image in addition to the estimated or ground truth images. Finally, $\mathcal{L}_{\text{vgg}}$ is the VGG-based perceptual loss, as proposed by Chen and Koltun [CK17]. Note that by minimizing this loss, we ensure that all the generators $G = \{G_i\}_{i=1}^{4}$ are able to produce consistent results.

Light Position As discussed, our network also estimates the position of the input light in the Cartesian coordinate (x, y, z) through an auxiliary output. To train the network for this task, we use last term in Eq. 1, E_l, which is the L1 distance between the estimated $\hat{q} = G_l(I_q)$ and ground truth light positions, q.

4. Real Training

As discussed, because of the large gap in data distribution of synthetic and real images, the network trained on only synthetic images can produce suboptimal results on real examples. To address this problem, we propose to train our system on a combination of real and synthetic examples. Unfortunately, obtaining ground truth SVBRDF parameters for real images is difficult.

To overcome this limitation, we use images of the same object with different lighting for supervising our generators. Specifically, given a pair of images, I_a and I_b, we use I_a as the input to our generators and I_b as ground truth. Using the estimated SVBRDF parameters from image a, $F = G(I_a)$, we render the object with the light position of the other image b. The major challenge is that the light position for the real image I_b is not known. To address this issue, we use our network to estimate the light position $\hat{b}$ from image I_b. We use the following objective function to train our system on real images:

$$E_{\text{real}} = \mathcal{L}_{\text{adv}}(G, D_{\text{ren-r}}) + \lambda_{1,r}\mathcal{L}_{\text{feat}}(G, D_{\text{ren-r}}) + \lambda_{2,r}\|R(\hat{F}, \hat{b}) - I_b\|_1, \tag{4}$$

where we use $\lambda_{1,r} = 10$ and $\lambda_{2,r} = 50$ in our implementation. By minimizing this loss on real images in combination with the loss function in Eq. 1 on synthetic images, we ensure the network is able to produce high-quality results on real examples.

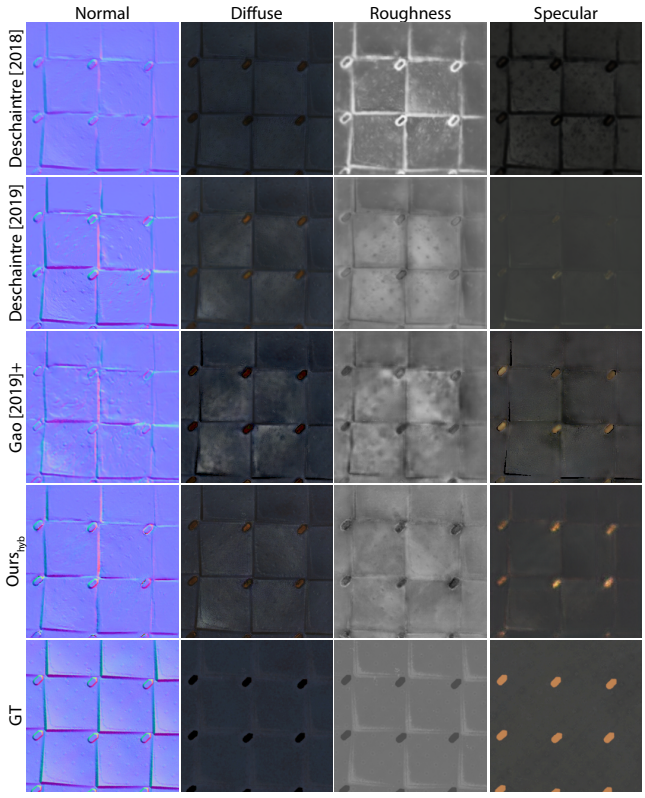

Figure 4: *Comparison of our estimated parameters against the other state-of-the-art approaches. See Fig. 5 (right) for the input image and comparison of the rendered images.*

In summary, we use the following loss function to train our system on both synthetic and real images:

$$E = E_{\text{syn}} + \lambda E_{\text{real}}, \tag{5}$$

where E_{syn} and E_{real} are defined in Eqs. 1 and 4, respectively, and λ_r is set to 5 in our implementation.

Note that, Li et al. [LDPT17] and Ye et al. [YLD*18] have also proposed a strategy to train their networks on real images without ground truth SVBRDF parameters. Specifically, they propose to first estimate a set of parameters from the input image and use it to render an image. They then use the rendered image to estimate another set of parameters. They train the network by minimizing the L1 loss between these two sets of parameters. However, their supervision is weak as there are many solutions that can minimize the loss. In contrast, by using a pair of images, our approach provides a stronger supervision.

5. Implementation

In this section, we discuss the details required for implementing our approach. We begin by explaining the architecture of our networks.

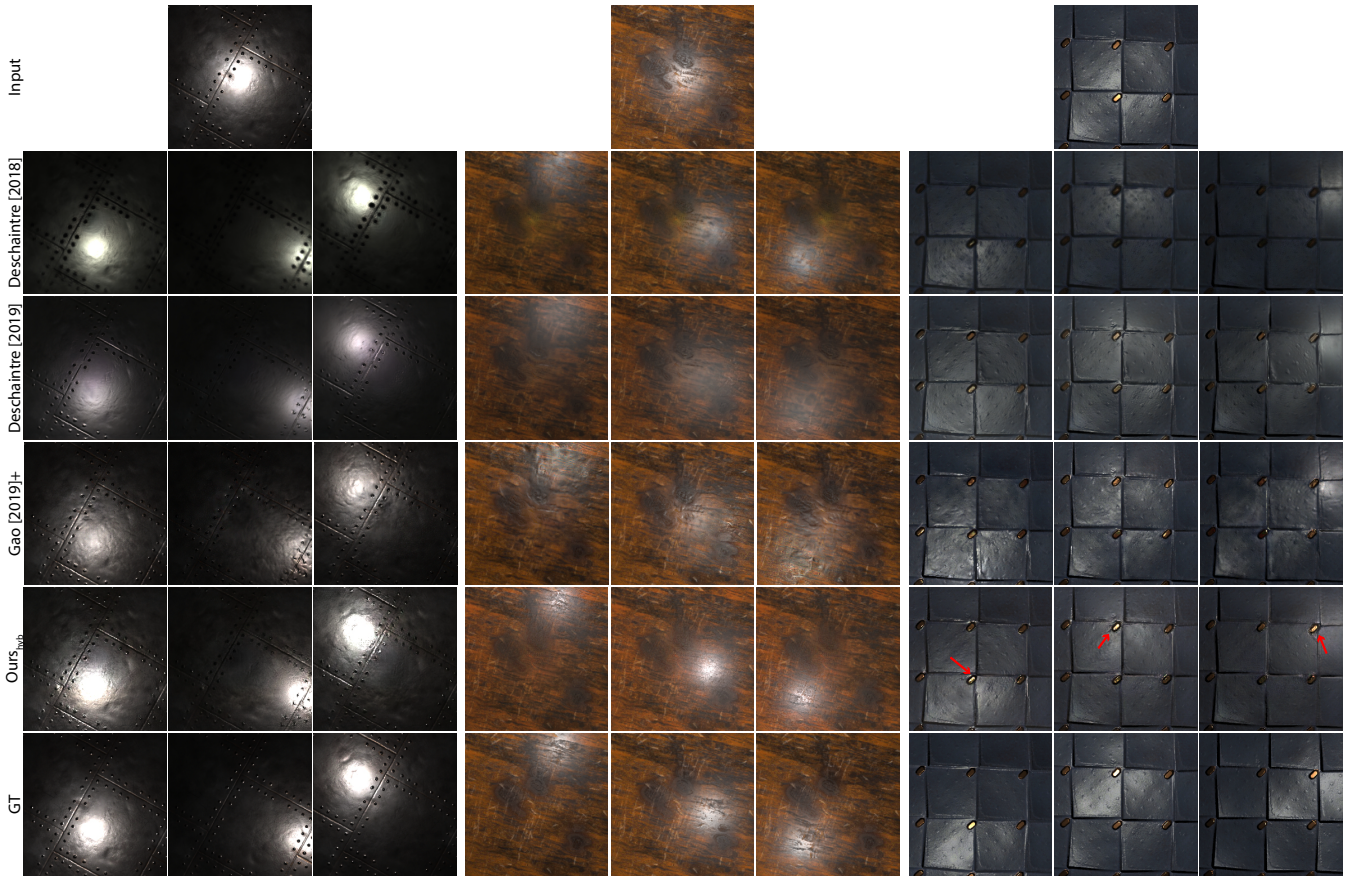

Figure 5: *Comparison of the rendered images with three different lighting using the estimated parameters from our method against the other approaches on three synthetic input images. Our method produces sharper images and reproduces the specular highlights better than the other techniques. Comparison of the estimated parameters for the image on the right is provided in Fig. 4.*

5.1. Architecture

Our architecture for the SVBRDF parameter generators is similar to the one proposed by Deschaintre et al. [DAD*18] with two differences. First, we only use their local branch which is a simple encoder-decoder network with skip connections. Second, instead of one decoder, we use four decoders with 3, 3, 1, and 3 output channels to estimate the normal, diffuse albedo, roughness and specular albedo. To estimate the position of light sources, we apply a set of fully connected layers to the features extracted using the encoder. Specifically, for a 256×256 image, the features at the end of the encoder are of size $1 \times 1 \times 512$. We pass these 512 features through 6 fully connected layers of size 256, 128, 64, 32, 16, 3. All the layers are followed by the leaky ReLU activation function except the last one which is linear. Finally, we use the discriminator architecture of Wang et al. [WLZ*18] for all of the discriminators.

5.2. Dataset

To train our system on synthetic images, we use the dataset of Deschaintre et al. [DAD*18]. From a set of SVBRDF parameters, we use the Cook-Torrance model [CT82] to synthesize the input image as well as the ground truth rendered image used in Eq. 3. For all the examples, we use orthographic view and randomly select the position of the point light based on cosine-weighted distribution on the upper hemisphere. We randomly select the distance of the light source to the center of the surface using $\exp(d)$, where d follows the normal distribution $\mathcal{N}(\mu = 1.0, \sigma = 0.75)$. We clamp the rendered image to one and gamma correct the result. Finally,

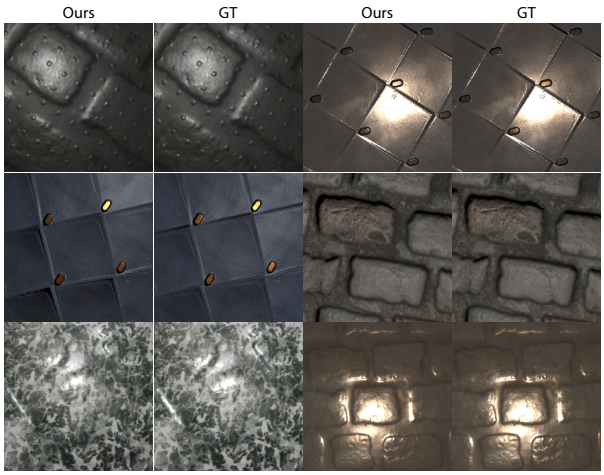

Figure 6: *We demonstrate the accuracy of our estimated light positions. The images on the left are obtained by re-rendering the ground truth images (right) using our estimated light positions and ground truth reflectance parameters.*

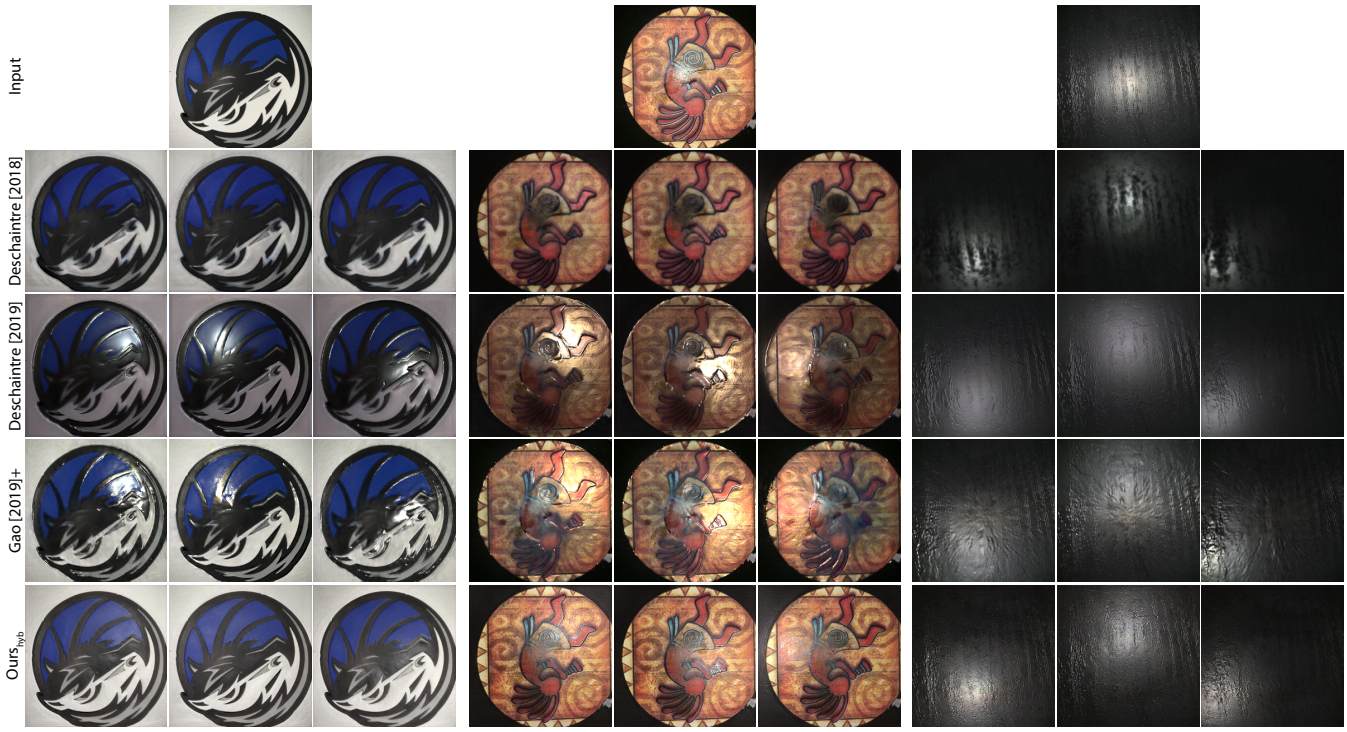

Figure 7: *Comparison of the rendered images with three different lighting using the estimated parameters from our method against the other approaches on three real input images. Our rendered images are overall sharper and have more realistic specular highlights than the other methods.*

we normalize the rendered images to [-1, 1] when passing them as the input to the network.

For the real images, we use a tripod mounted camera and take a video of a planar surface under a moving flash light. We collect a set of roughly 500 videos of a variety of different surfaces in this manner. We then select between one to three crops of each video containing reasonable lighting, producing around 1,000 videos. From each video, we take a set of 5 to 10 frames with sufficiently different light positions and resize them to 256×256. During training, we randomly select a pair of images from each scene to be used based on Eq. 4. A few real image pairs from our training set are shown in Fig. 3.

5.3. Training

We perform the training in three different stages. We start the training process on synthetic images and exclude the rendering discriminator in this stage, i.e., optimizing only Eq. 1 without the E_{ren} term. We perform this training for 250,000 iterations and then incorporate the rendering discriminator and continue the training for another 200,000 iterations. After this stage, we combine the synthetic with real examples and train our system for 100,000 iterations. We use a batch size of 4 during training on synthetic data. For hybrid training, we use a batch size of 5 consisting of 4 synthetic and one real data. We implement our approach in PyTorch and use Adam [KB14] with a learning rate of 2×10^{-5} and β_1 of 0.5; all other parameters are kept to the default. The training takes around 3 days on a single GeoForce RTX 2080 Ti GPU.

6. Results

We evaluate our approach on a set of synthetic and real images by comparing against the state-of-the-art techniques. Specifically, we compare against the approaches by Deschaintre et al. [DAD*18; DAD*19] and Gao et al. [GLD*19]. Note that, the two approaches by Deschaintre et al. are trained on synthetic images. On the other hand, the approach by Gao et al. [GLD*19] estimate the reflectance parameters of a test image through optimization. We initialize their optimization system using the results of Deschaintre et al.'s approach [DAD*19]. Moreover, the method of Deschaintre et al. [DAD*19] and Gao et al. [GLD*19] work on arbitrary number of input images, but we generate their results using a single image to compare against our single-image method. Finally, we use input images with centered light positions to ensure fair comparisons against the approaches by Deschaintre et al. [DAD*18] and Gao et al. [GLD*19]. We use the source code provided by the authors for all the comparisons. Here, we only show a few results, but in the supplementary materials, we compare our approach against other methods on a large number of images.

6.1. Comparison Against the Other Approaches

Synthetic Images We begin by visually comparing our estimated parameters against the other approaches in Fig. 4. Compared to the other methods, our approach is able to produce reflectance parameters that are closer to the ground truth. Note that, while Deschaintre et al.'s approach [DAD*18] properly separates the lighting from the diffuse component, it produces a blurry map which negatively impacts the quality of renderings (see Fig. 5). We further compare our

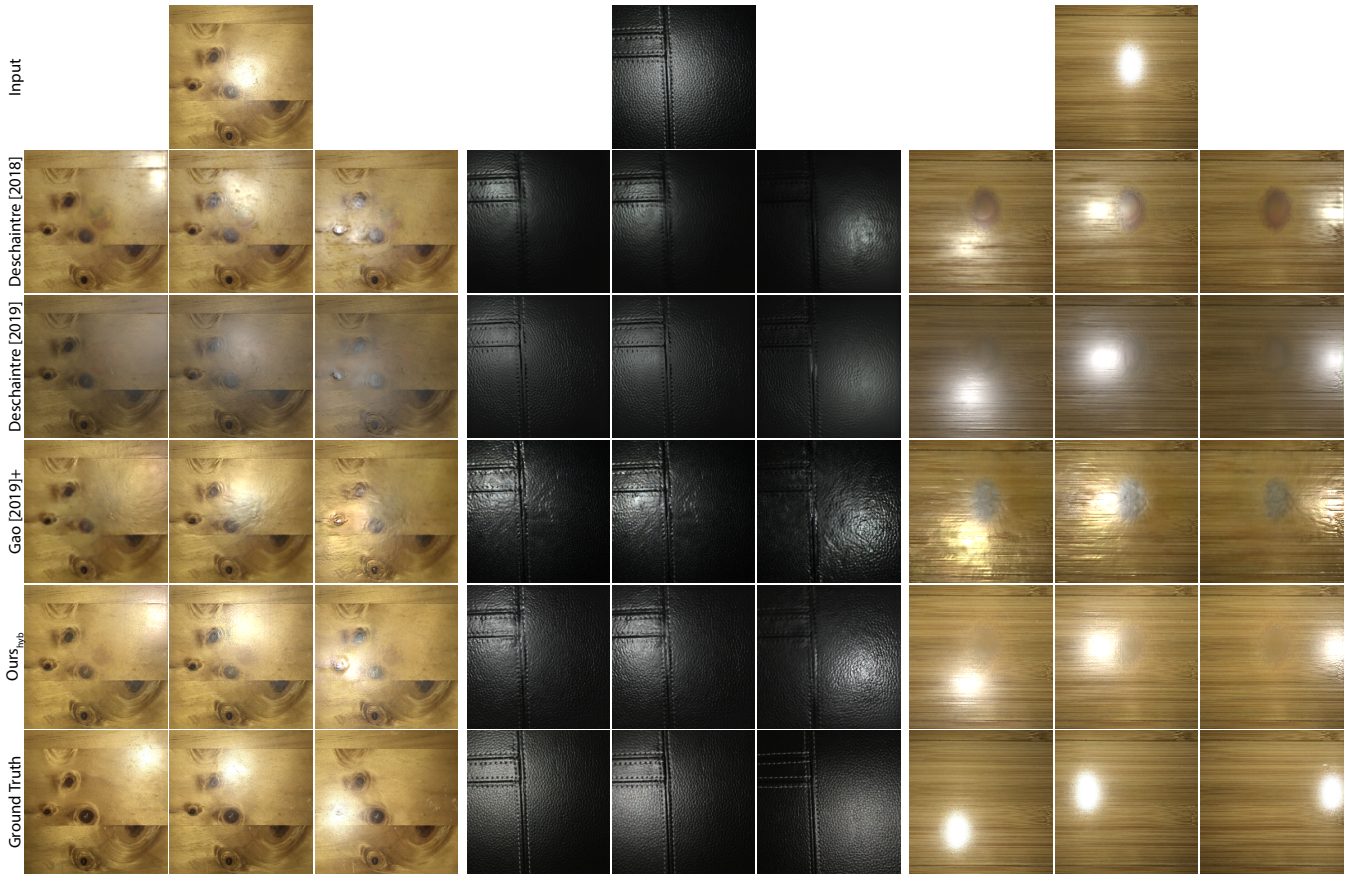

Figure 8: *Comparison of the rendered images produced by our method against the other approaches on three real input images with ground truth. The middle example is from Deschaintre et al. [DAD*19], while the rest are from Guo et al. [GSH*20]. Our rendered images are overall sharper, have fewer artifacts, and are closer to the ground truth images.*

rendered images against other methods on three synthetic images in Fig. 5. Overall, the other approaches are not able to properly estimate the color, texture details, and the specular highlights. Additionally, since the optimization system by Gao et al. using a single image is highly underconstrained, their method often introduces unnecessary details to the normal maps. In contrast, our method produces results with realistic appearance compared to the ground truth. Note that, for the scene on the right, only our approach is able to properly estimate the orange specular highlights as indicated by the arrows. Moreover, it is worth noting that Gao et al.'s method is slow as their optimization is expensive.

Next, we quantitatively compare our method against the other approaches on a set of 132 synthetic images in Table 1. We use root mean squared error (RMSE) to evaluate the quality of the reflectance parameters, while the rendered images are evaluated in terms of both RMSE and learned perceptual image patch similarity (LPIPS) [ZIE*18]. Moreover, we divide the table into two sections. At the top, we compare our network trained on synthetic (Ours$_{syn}$) and hybrid (Ours$_{hyb}$) data against the two approaches by Deschaintre et al. as well as Gao et al.'s method without refinement. As seen, our synthetic and hybrid networks produce comparable results which demonstrates that real training does not negatively impact the quality of the results on this synthetic test set. Moreover, our method is able to produce better or comparable results compared to the other approaches in terms of RMSE. However, because of using adversarial loss, our renderings have significantly better perceptual quality in terms of the LPIPS metric. Note that, while the

Table 1: *Numerical comparison on a set of 132 synthetic test images. N, D, R and S refer to normal, diffuse albedo, roughness, specular albedo. Ren refer to renderings for which the numerical values are obtained on a set of 20 images of each scene under different lights. Note that we evaluate the quality of rendering both in terms of RMSE and LPIPS, a perceptual metric.*

	RMSE					LPIPS
	N	D	R	S	Ren	Ren
Des18	0.065	**0.058**	0.175	0.129	0.086	0.278
Des19	0.092	0.062	0.129	0.066	0.100	0.223
Gao19	0.068	0.065	0.123	0.065	**0.072**	0.274
Ours$_{syn}$	**0.054**	0.062	**0.100**	**0.065**	0.078	0.193
Ours$_{hyb}$	0.056	0.062	0.104	0.066	0.074	**0.187**
Gao19+	0.069	0.065	0.123	**0.070**	0.068	0.177
Ours$_{syn}$+	**0.053**	**0.063**	**0.115**	0.079	**0.058**	**0.153**
Ours$_{hyb}$+	0.054	**0.063**	0.124	0.090	0.059	0.155

Table 2: *Numerical comparison on a set of 36 real test scenes from Guo et al. [GSH*20]. For each scene, one image is used as the input and the remaining 8 images are used as ground truth. We evaluate the quality of renderings in terms of RMSE and LPIPS.*

	LPIPS	RMSE
Des18	0.349	**0.131**
Des19	0.335	0.168
Gao19	0.419	0.132
Ours_{syn}	0.306	0.148
Ours_{hyb}	**0.279**	0.138
Gao19+	0.307	0.133
Ours_{syn}+	0.275	0.121
Ours_{hyb}+	**0.256**	**0.119**

approach by Deschaintre et al. [DAD*18] produces slightly better diffuse maps, they are usually blurry and lack fine details.

At the bottom of Table 1, we show the results of Gao et al.'s method after refinement. Here, we also use a similar refinement strategy for both versions of our approach where we further refine the reflectance parameters by minimizing the distance between rendered and input images. The main difference with respect to Gao et al.'s refinement is that during the initial iterations we only optimize the roughness and specular components using the average gradient across all the pixels, before performing per-pixel optimization for all the parameters. Overall, our method produces better results than Gao et al.'s approach. In particular, our renderings are significantly better than theirs in terms of the perceptual LPIPS metric.

Finally, we demonstrate the accuracy of our estimated light positions in Fig. 6. Here, we use our network to estimate the position of the light from an input image. We then use our estimated light position along with the ground truth feature maps to rerender the input image. The rerendered images (Ours) and the ground truth input images (GT) are shown in this figure. As seen, our rendered images closely match the ground truth, which demonstrates the ability of our network to accurately estimate the light position.

Real Images Figure 7 compares our rendering results against the other methods on three real images. We capture these scenes with a different camera from the one used to capture our real training set. Moreover, we provide our estimated light position to Gao et al.'s method approach, since they require it for the optimization. Approaches by Deschaintre et al. [DAD*19] and Gao et al. [GLD*19] produce results with strong highlights for the rubber material shown on the left. On the other hand, Deschaintre et al. [DAD*18] is not able to properly represent the texture details. Our method, however, reproduces the appearance of the material more realistically.

For the middle examples, Deschaintre et al. [DAD*18] produces a mostly diffuse appearance, while the other two approaches generate renderings with strong specular highlights. On the other hand, our approach is able to properly estimate the texture details and specular highlights. Finally, for the example on the right, Deschaintre et al. [DAD*18] and Gao et al. [GLD*19] generate results with severe artifacts. While Deschaintre et al. [DAD*19] avoid introducing artifacts, their results are over-smooth. Our method is able to produce renderings that better match the input image. We show the estimated reflectance parameters for all these images in the supplementary material.

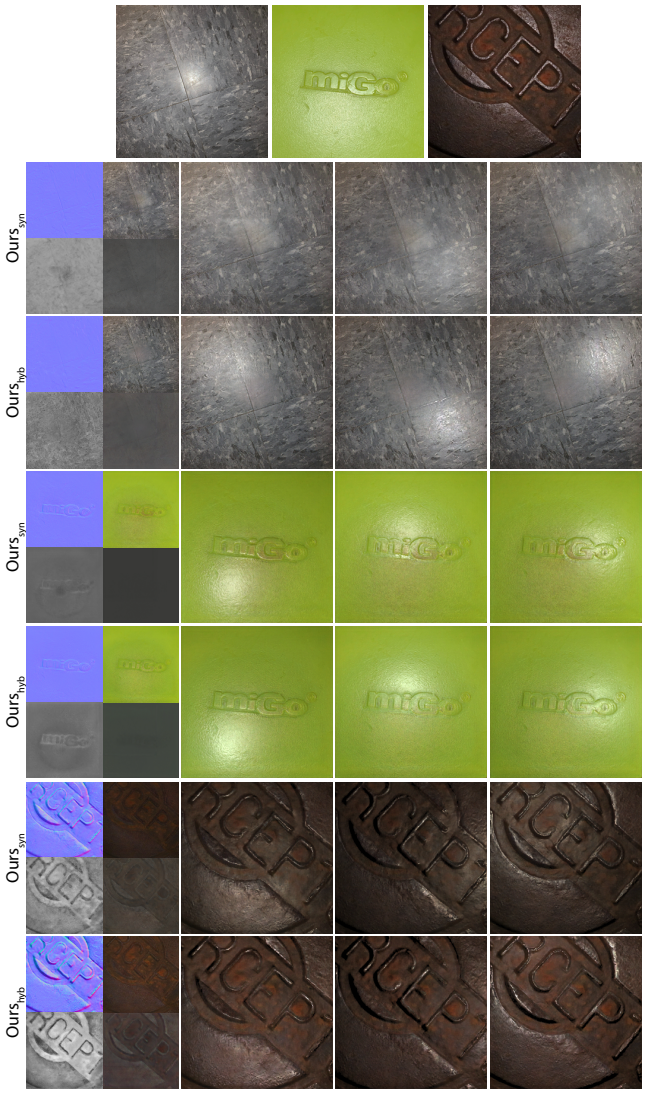

Figure 9: *Evaluating the effect of training on real images. Our network trained on both real and synthetic images can reconstruct the specular highlights and the diffuse map better than our network trained on only synthetic images.*

Furthermore, we compare our rendering results against other methods on three real images with ground truth in Fig. 8. Overall, our approach is able to produce sharper results with specular highlights that match the ground truth better than the other methods. Moreover, we quantitatively compare our results against other approaches on a set of 36 real scenes from Guo et al. [GSH*20] in Table 2. Our method produces results that are better or comparable in terms of RMSE, but are significantly better than the other techniques in terms of LPIPS. Additionally, our hybrid network produces significantly better results than the synthetic one, demonstrating the effectiveness of training on real images.

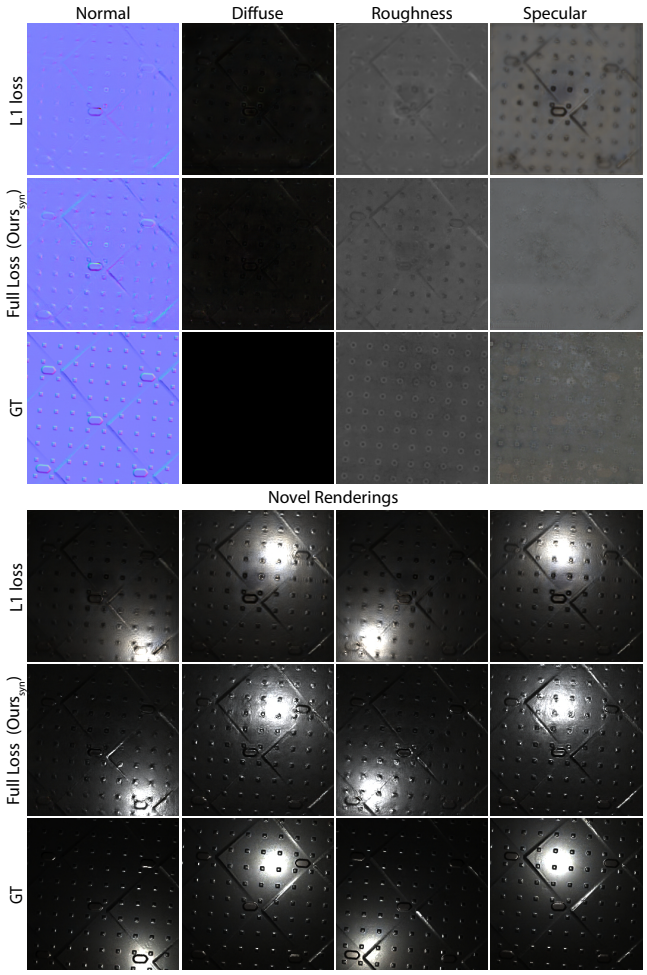

Figure 10: *Comparison of our network trained with only L1 loss against our full adversarial loss.*

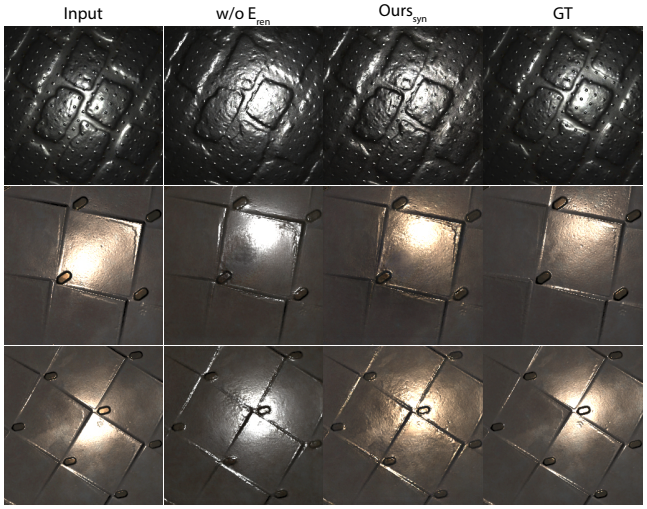

Figure 11: *Evaluating the effect of the rendering loss.*

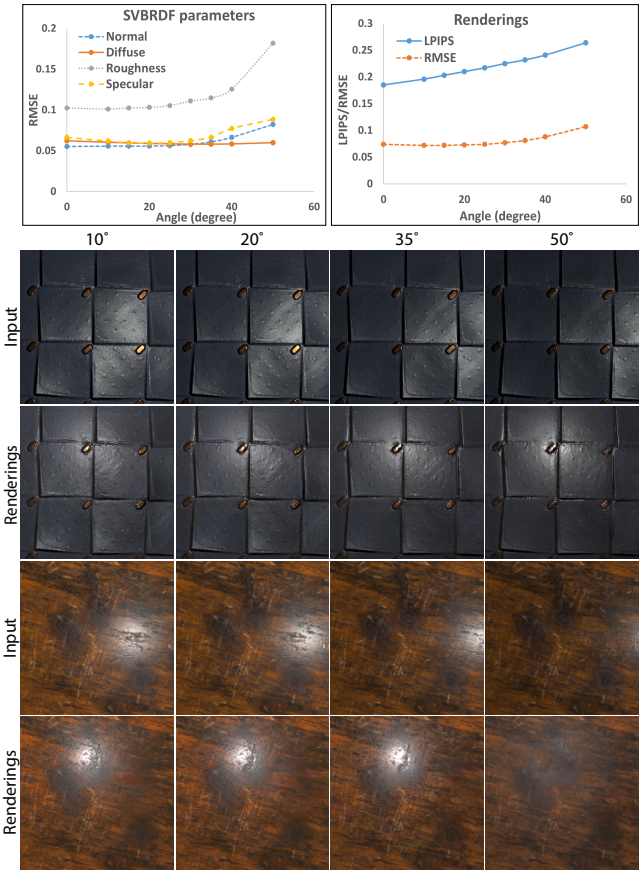

Figure 12: *Evaluating the robustness of our approach to the position of input light source. Top two diagrams show the RMSE of the estimated parameters (left) and LPIPS/RSME of renderings (right) for inputs with different light positions. Bottom rows show rendering results using estimated SVBRDF parameters from input images lit by different light positions.*

6.2. Ablation Studies

Here, we analyze the effect different components of our system. For all the ablation experiments, we follow the same training strategy discussed in Sec. 5.3.

Effect of the Hybrid Training Figure 9 compares our full approach (hybrid training) against our method trained only on synthetic images on three real images. For the example on the top, our hybrid network is better at reproducing the appearance of the specular highlights. In particular, it is able to properly capture the roughness of the scrateches on the tiles, producing detailed highlights. In the middle example, our synthetic network has difficulty estimating the diffuse map in the regions where the input image has highlights. Our hybrid network, on the other hand, produces results without noticeable artifacts. Finally, in the bottom example, the metal part between the letters is rusty and non-reflective. Our hybrid network correctly estimates the roughness for this area and produces the appearance of rusted metal.

Effect of Adversarial Loss To evaluate the effect of the adversarial loss, we compare the results of our generators using only the L1 loss with our approach trained with the full loss in Fig. 10.

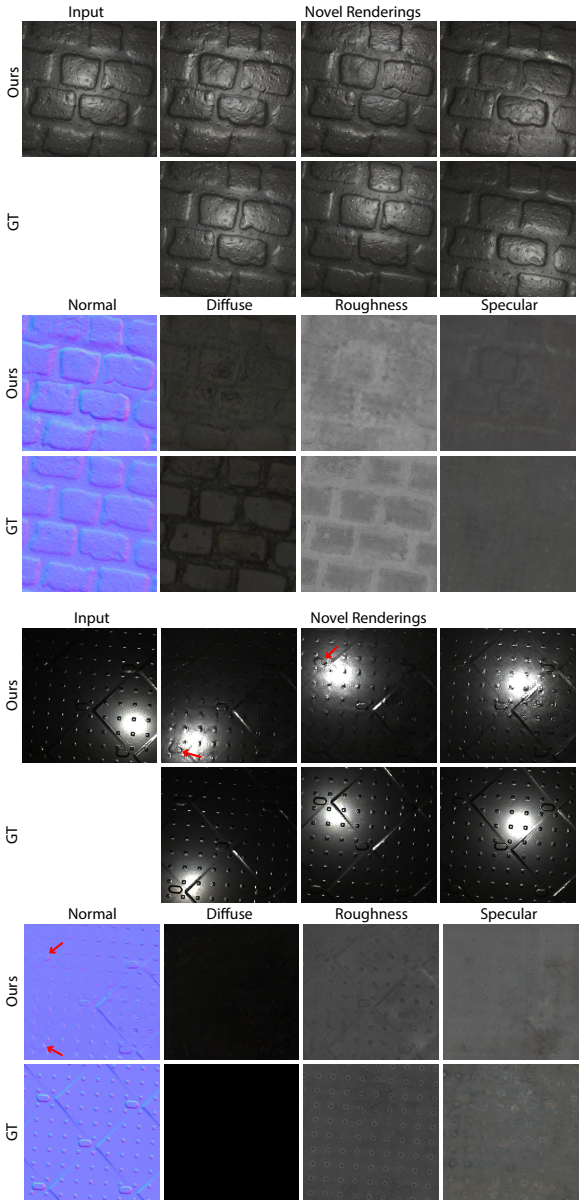

Figure 13: *The failure cases of our approach. On the top, we show a case where our approach produces fine details and structures that do not exist in the ground truth. On the bottom, our approach is not able to properly reconstruct all the details because of lack of sufficient information in the single input image*

Our approach trained with only L1 loss fails to capture details, specially in the normal map. On the other hand, our approach with the full loss function is able to produce sharper results with more details, demonstrating the benefit of the adversarial loss. Moreover, we quantitatively evaluate the effect of adversarial loss using renderings of 132 synthetic scenes. Our system trained with L1 loss produces LPIPS/RMSE value of 0.236/0.073, while our system trained with the full adversarial loss produces LPIPS/RMSE value of 0.193/0.078. Although training with L1 produces lower RMSE values, the results are generally blurry and have lower perceptual quality, as indicated by the LPIPS values.

Effect of Rendering Loss To analyze the effect of the rendering loss E_{ren}, we compare the results of our network trained with and without this loss in Fig. 11. Note that, we train the networks in both cases only on synthetic images to properly evaluate the effect of E_{ren} term in Eq. 1. As seen, while using the network trained with four discriminators is able to produce sharp images with fine details, the renderings are typically not consistent with the ground truth. Using the rendering discriminator, we are able to generate results that better match the ground truth renderings.

Effect of Light Position We also analyze the robustness of our approach to the position of light source in the input image. To do so, we change the angle of light source and compute the LPIPS/RMSE value of renderings on 132 synthetic scenes in Fig. 12. As seen, numerically our system is fairly robust to deviation from the center up to 35 degrees. This is also confirmed by the visual results where our approach consistently reproduces the highlights for input images with light angles up to 35 degrees.

6.3. Limitations and Future Work

In some cases with strong specular highlights, our approach fails to properly remove the highlights from the estimated parameters. These often show up in the rendered images in form of highlight removal artifacts. However, this is also an issue with all the other approaches. In fact, our adversarial framework is able to better reduce these artifacts by inpainting the lost content (see supplementary material).

Furthermore, since we use an adversarial loss function to train the network, our method in some cases produce fine details and structures that do not exist in the ground truth image, as shown in Fig. 13 (top). However, our results are still visually plausible. Finally, in some cases, we are not able to properly capture all the details of the reflectance parameters across the whole image, as shown in Fig. 13 (bottom). In this case, our method is not able to properly estimate the map in the regions away from the highlights, since the single input image does not provide useful information in these areas. In the future, it would be interesting to resolve this issue by extending our approach to use multiple images as the input.

7. Conclusion

We have presented an adversarial framework to estimate the four spatially-varying BRDF parameters from a single input photograph. We do so using an encoder-decoder network with shared encoder and four decoders and train our network using a set of discriminators to distinguish the estimated results from ground truth. In addition to training on synthetic images, we propose a novel strategy to provide a strong supervision for our network on real image pairs. We demonstrate that our approach produces better results on both synthetic and real images compared to the state-of-the-art methods.

References

[AAL16] AITTALA, MIIKA, AILA, TIMO, and LEHTINEN, JAAKKO. "Reflectance modeling by neural texture synthesis". *ACM Transactions on Graphics (ToG)* 35.4 (2016), 1–13 2.

[AWL*15] AITTALA, MIIKA, WEYRICH, TIM, LEHTINEN, JAAKKO, et al. "Two-shot SVBRDF capture for stationary materials." *ACM Trans. Graph.* 34.4 (2015), 110–1 2.

[BKR18] BI, SAI, KALANTARI, NIMA KHADEMI, and RAMAMOORTHI, RAVI. "Deep hybrid real and synthetic training for intrinsic decomposition". *arXiv preprint arXiv:1807.11226* (2018) 2.

[CK17] CHEN, QIFENG and KOLTUN, VLADLEN. "Photographic image synthesis with cascaded refinement networks". *Proceedings of the IEEE international conference on computer vision*. 2017, 1511–1520 4.

[CT82] COOK, ROBERT L and TORRANCE, KENNETH E. "A reflectance model for computer graphics". *ACM Transactions on Graphics (TOG)* 1.1 (1982), 7–24 3, 5.

[DAD*18] DESCHAINTRE, VALENTIN, AITTALA, MIIKA, DURAND, FREDO, et al. "Single-image svbrdf capture with a rendering-aware deep network". *ACM Transactions on Graphics (ToG)* 37.4 (2018), 1–15 1, 2, 5, 6, 8.

[DAD*19] DESCHAINTRE, VALENTIN, AITTALA, MIIKA, DURAND, FRÉDO, et al. "Flexible SVBRDF Capture with a Multi-Image Deep Network". *Computer Graphics Forum*. Vol. 38. 4. Wiley Online Library. 2019, 1–13 2, 6–8.

[DDB20] DESCHAINTRE, VALENTIN, DRETTAKIS, GEORGE, and BOUSSEAU, ADRIEN. "Guided Fine-Tuning for Large-Scale Material Transfer". *Computer Graphics Forum*. Vol. 39. 4. Wiley Online Library. 2020, 91–105 2.

[DWT*10] DONG, YUE, WANG, JIAPING, TONG, XIN, et al. "Manifold bootstrapping for SVBRDF capture". *ACM Transactions on Graphics (TOG)* 29.4 (2010), 1–10 2.

[GAHO07] GHOSH, ABHIJEET, ACHUTHA, SHRUTHI, HEIDRICH, WOLFGANG, and O'TOOLE, MATTHEW. "BRDF acquisition with basis illumination". *2007 IEEE 11th International Conference on Computer Vision*. IEEE. 2007, 1–8 2.

[GCHS09] GOLDMAN, DAN B, CURLESS, BRIAN, HERTZMANN, AARON, and SEITZ, STEVEN M. "Shape and spatially-varying brdfs from photometric stereo". *IEEE Transactions on Pattern Analysis and Machine Intelligence* 32.6 (2009), 1060–1071 2.

[GCP*10] GHOSH, ABHIJEET, CHEN, TONGBO, PEERS, PIETER, et al. "Circularly polarized spherical illumination reflectometry". *ACM SIGGRAPH Asia 2010 papers*. 2010, 1–12 2.

[GLD*19] GAO, DUAN, LI, XIAO, DONG, YUE, et al. "Deep inverse rendering for high-resolution SVBRDF estimation from an arbitrary number of images". *ACM Transactions on Graphics (TOG)* 38.4 (2019), 134 2, 6, 8.

[GPM*14] GOODFELLOW, IAN, POUGET-ABADIE, JEAN, MIRZA, MEHDI, et al. "Generative adversarial nets". *Advances in neural information processing systems*. 2014, 2672–2680 2.

[GSH*20] GUO, YU, SMITH, CAMERON, HAŠAN, MILOŠ, et al. "MaterialGAN: Reflectance Capture using a Generative SVBRDF Model". *ACM Trans. Graph.* 39.6 (2020), 254:1–254:13 2, 7, 8.

[HSL*17] HUI, ZHUO, SUNKAVALLI, KALYAN, LEE, JOON-YOUNG, et al. "Reflectance capture using univariate sampling of BRDFs". *Proceedings of the IEEE International Conference on Computer Vision*. 2017, 5362–5370 2.

[IZZE17] ISOLA, PHILLIP, ZHU, JUN-YAN, ZHOU, TINGHUI, and EFROS, ALEXEI A. "Image-to-image translation with conditional adversarial networks". *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2017, 1125–1134 3.

[KB14] KINGMA, DIEDERIK P and BA, JIMMY. "Adam: A method for stochastic optimization". *arXiv preprint arXiv:1412.6980* (2014) 6.

[KLA*20] KARRAS, TERO, LAINE, SAMULI, AITTALA, MIIKA, et al. "Analyzing and improving the image quality of stylegan". *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2020, 8110–8119 2.

[LDPT17] LI, XIAO, DONG, YUE, PEERS, PIETER, and TONG, XIN. "Modeling surface appearance from a single photograph using self-augmented convolutional neural networks". *ACM Transactions on Graphics (TOG)* 36.4 (2017), 1–11 1, 2, 4.

[LS18] LI, ZHENGQI and SNAVELY, NOAH. "Learning Intrinsic Image Decomposition From Watching the World". *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. June 2018 2.

[LSC18] LI, ZHENGQIN, SUNKAVALLI, KALYAN, and CHANDRAKER, MANMOHAN. "Materials for masses: SVBRDF acquisition with a single mobile phone image". *Proceedings of the European Conference on Computer Vision (ECCV)*. 2018, 72–87 1, 2.

[LXR*18] LI, ZHENGQIN, XU, ZEXIANG, RAMAMOORTHI, RAVI, et al. "Learning to reconstruct shape and spatially-varying reflectance from a single image". *ACM Transactions on Graphics (TOG)* 37.6 (2018), 1–11 2.

[MLX*17] MAO, XUDONG, LI, QING, XIE, HAORAN, et al. "Least squares generative adversarial networks". *Proceedings of the IEEE International Conference on Computer Vision*. 2017, 2794–2802 3.

[WLZ*18] WANG, TING-CHUN, LIU, MING-YU, ZHU, JUN-YAN, et al. "High-resolution image synthesis and semantic manipulation with conditional gans". *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2018, 8798–8807 3, 5.

[WMP*06] WEYRICH, TIM, MATUSIK, WOJCIECH, PFISTER, HANSPETER, et al. "Analysis of human faces using a measurement-based skin reflectance model". *ACM Transactions on Graphics (TOG)* 25.3 (2006), 1013–1024 2.

[YLD*18] YE, WENJIE, LI, XIAO, DONG, YUE, et al. "Single image surface appearance modeling with self-augmented cnns and inexact supervision". *Computer Graphics Forum*. Vol. 37. 7. Wiley Online Library. 2018, 201–211 1, 2, 4.

[ZCD*16] ZHOU, ZHIMING, CHEN, GUOJUN, DONG, YUE, et al. "Sparse-as-possible SVBRDF acquisition". *ACM Transactions on Graphics (TOG)* 35.6 (2016), 1–12 2.

[ZIE*18] ZHANG, RICHARD, ISOLA, PHILLIP, EFROS, ALEXEI A, et al. "The Unreasonable Effectiveness of Deep Features as a Perceptual Metric". *CVPR*. 2018 7.

[ZWX*20] ZHAO, YEZI, WANG, BEIBEI, XU, YANNING, et al. "Joint SVBRDF Recovery and Synthesis From a Single Image using an Unsupervised Generative Adversarial Network". (2020) 2.