Vision, Modeling, and Visualization (2021)
B. Andres, M. Campen, and M. Sedlmair (Eds.)

Capturing Anisotropic SVBRDFs

J. Kaltheuner, L. Bode, R. Klein

University of Bonn, Germany

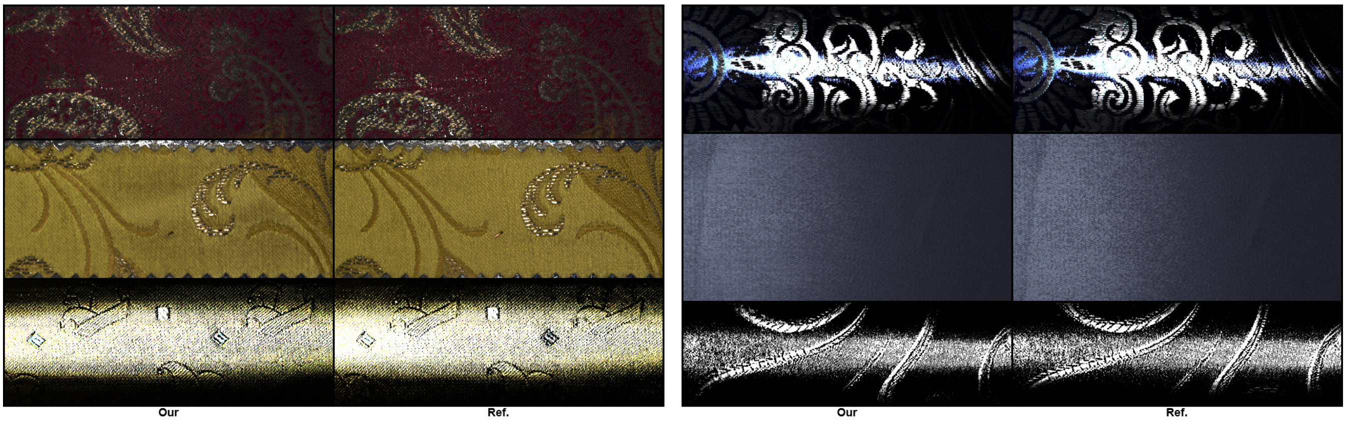

Figure 1: *Example renderings of materials with highly complex reflectance behavior containing anisotropy and Fresnel reflections. The SVBRDFs were reconstructed using our novel method from only 14 pairs of view- and light-configurations. The SVBRDFs of the rendered reference materials are provided by the Bonn SVBRDF dataset[MHRK19].*

Abstract

In this work, we adapt and improve recent isotropic material estimation efforts to estimate spatially varying anisotropic materials with an additional Fresnel term using a variable set of input images and are able to handle any resolution. We combine an initial estimation network with an auto-encoder to fine-tune the decoding of latent embedded appearance parameters on the input images to produce finely detailed SVBRDFs. For this purpose, the training must be adapted so that the determination is possible on the basis of a small number of images that still capture as much reflective behavior of materials as possible. The resulting appearance parameters are capable of capturing and reconstructing complex spatially varying features in detail, but place increased demands on the input images.

CCS Concepts

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

1. Introduction

The capturing and digital representation of spatially varying reflectance behavior of real-world materials is an important topic with many applications, such as entertainment, media and advertisement. The use of spatially varying bidirectional reflectance distribution functions (SVBRDFs) has proven to be advantageous as this parametric representation is very compact and efficient to render. However, the acquisition of this representation often requires specialized equipment and a laboratory environment as well as a large number of images of the material to capture the exact reflectance behaviour. An effective device designed specifically for this purpose is the TAC-7 scanner [XR18].

In recent years, advances with light-weight learning-based methods have softened the requirements and made it possible to capture SVBRDFs using simple devices such as mobile phone cameras, often times requiring as little as a single image, but they are often limited to a small resolution of 256 × 256. These deep-learning based methods are concerned with the detection of isotropic surfaces. However, some materials require more complex reflectance representations which need additional parameters, displayed in Figure 2. For these, the representation with isotropic models means a

DOI: 10.2312/vmv.20211372

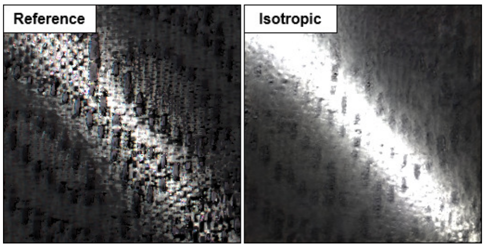

Figure 2: *Display of problematic reflections for isotropic estimation approaches produced by the MaterialGAN.*

loss of reflection behavior. We seek to estimate more complex reflections by building on existing methods for isotropic materials and refining and adapting their methodology to predict anisotropic reflectance behavior, as well as reflectances modeled by a Fresnel equation, from a few input images. Recovering these reflections of a material from a few input images is an ill-posed problem as many combinations of appearance parameters can represent the original images. As the complexity of the underlying appearance model increases, so does the set of possible parameter representations. In Figure 1 we present results obtained with our method for such complex materials. Our main contribution to the state-of-the-art is thereby:

- Estimation of anisotropic reflectance behavior and spatially varying Fresnel terms.
- Identification of rules to select view- and light-configurations necessary for a few shot capturing scenario and network training.
- Support for arbitrary image resolutions.

2. Related Work

For many years, the parameters of materials have been measured using specialized equipment in laboratory environments. For such procedures, devices such as the TAC7 scanner[XR18] are used, but they take many hours and are not applicable in the wild. The rise of learning-based methods in various fields has also brought great advances to material acquisition methods. One of the first learning based techniques[DAW01] is able to classify reflectance behaviour under unknown light conditions and arbitrary geometry of the surface. It is able to learn connections between the reflectance and certain statistics based on an image. Closer to our problem is a work by [LDPT17], which also attempts to use Convolutional Neural Networks (CNN) to estimate SVBRDF parameters based on a single image. However, it only deals with the prediction of normal and diffuse features and is therefore limited to represent only uniform specular reflections. A later learning-based method, [LXR*18], deals with the estimation of albedos, roughness, normal and depth from a single image and is thus already able to represent much more detailed reflections. A subsequent paper[BJK*20], which predicts the same reflection parameters, noted that a single image often does not contain enough information to reliably estimate all reflections, which also holds true for this work, where we need more informations to estimate the reflections.

One of the foundations of this work is the methodology of [DAD*18], in which a single flash-lit image is used to estimate an isotropic material. It uses a deep-learning U-Net architecture with an additional global feature track to combine local with spatially distant information. In a subsequent work [DAD*19] the method was extended to use multiple input images. We adapt and modify this architecture for our anisotropic materials while incorporating other ideas, as it is crucial for us to combine the information from multiple images into their appearance parameter representation. The combination of optimization methods with learning-based methods can also be used for the determination of SVBRDFs. Other methods, such as [BL19] improved on single image estimations using a Generative Adversarial Network (GAN) architecture.

In [BJLPS17] it has already been shown that optimizing a latent space of GANs can provide useful results in tasks such as sample interpolation and sample synthesis. The MaterialGAN network [GSH*20] showed that this latent space optimization can also be used to optimize the SVBRDF parameters of isotropic materials based on a few input images. The goal of the optimization is to be able to display the input images with the help of the generated appearance parameters. Previously, Gao [GLP*19] had shown that this optimization in the latent space of a simple auto-encoder is advantageous compared to direct optimization using a few input images for the appearance parameters of isotropic materials. We adapt this method for our work, where especially for our anisotropic materials the choice of input images is important to have a solid target for the optimization. Asselin's work [ALL20] further demonstrated the problem with many learning-based methods trained on synthetic materials and the weaknesses when applied to real-world materials. We therefore train on a dataset that does not directly use synthetic data, but instead uses traditional methods to obtain the SVBRDFs of anisotropic materials from the real world. In addition, the work of [BBJ*21] also showed the problems of shifting away from laboratory setups with a single point-light, considering rather uncontrolled environmental illumination, when dealing with the task of geometry and material property determination.

3. Methodology

Our method for SVBRDF estimation presented in Figure 3 uses an arbitrary number of input images, which are converted into appearance parameters using an estimation network introduced in Section 3.2. We use an encoder E that converts the appearance parameters into a latent vector z and a decoder D performing the back transformation with the procedure described in Section 3.3. We fine-tune this decoder for each material, using the input images as well as their view- and light-configuration, presented in Section 3.4. Additionally, we define a selection procedure for these configurations in Section 3.6, since their selection is crucial for the reconstruction and for keeping the number of required images low.

3.1. Rendering Model

We render the anisotropic materials using the Geisler-Moroder variant [GMD10] of the anisotropic Ward BRDF Model with Fresnel reflection term. The model is defined for a point p on the surface of the material. The geometric tangent frame is computed with the help of the heightmap $H(p)$. It is defined by a tangent $t_g(p)$, bitangent $b_g(p)$ and geometric normal $n_g(p)$ and computed using the

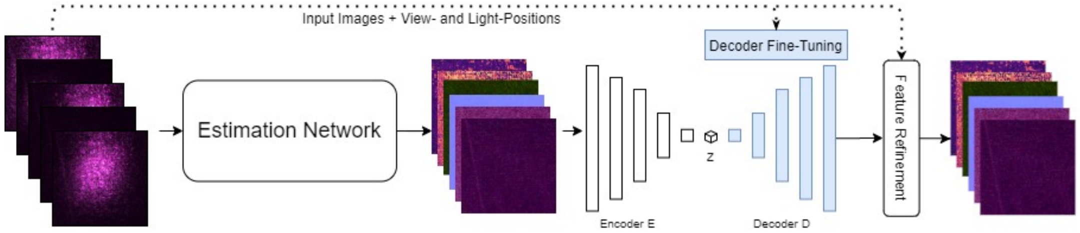

Figure 3: *Presentation of the pipeline of our anisotropic material estimation. All images, divided into 256×256 tiles are fed to an estimation network and recombined to the original resolution, creating an initial feature prediction. The appearance features are afterwards transformed to a latent embedding z with the help of a pre-trained encoder and use a pre-trained decoder to transform z back into feature predictions and fine-tune this decoder using the input images, thus improving the estimates. To further improve the resulting features, we refine them afterwards with the help of an traditional optimization of the estimate with the help of the input images.*

gradients of the heightmap. $t_g(p)$ and $b_g(p)$ are re-orthogonalized by a Gram-Schmidt step. The anisotropy map $\alpha(p)$ defines the angle of rotation of the tangent frames around an additional shading normal $n_s(p)$. Thus there are three basic vectors defining an orthonormal basis for each pixel.

The view- and light-direction v and l are transformed into the local coordinate frame and the un-normalized halfway vector h is created by adding these two directions up in $\mathbb{R}^3$. Additionaly the Fresnel term requires a normalized halfway vector h' computed as $h' = h/||h||$. The final anisotropic Ward model is detailed as

$$f(p,h,h') = \frac{a_d(p)}{\pi} + \frac{a_s(p)F(F_0(p),\langle v,h'\rangle)}{\pi\sigma_x(p)\sigma_y(p)h_z^4}e^d. \quad (1)$$

The parameters needed to compute the model are the diffuse albedo $a_d(p)$, specular albedo $a_s(p)$ and the Fresnel term at 0° incidence. The 2-dimensional lobes are defined as $\sigma = (\sigma_x, \sigma_y)^T$. The term d is computed as

$$d = -\frac{(h_x^2/\sigma_x(p)^2) + (h_y^2/\sigma_y(p)^2)}{h_z^2}. \quad (2)$$

Omitting the dependence on the single point p for our notation, we obtain our appearance parameters $s = \{a_d, a_s, n_s, \sigma, \alpha, F_0\}$ and make use of the Schlick Fresnel approximation [Sch94] defined by

$$F(F_0(p),\theta) = F_0(p)(1-F_0(p))(1-\cos\theta)^5. \quad (3)$$

It must be noted that the specular albedo a_s in our model, unlike in common Ward models and its variants, is not bounded by 1, but is divided by the Fresnel term $a_s = a_s'/F_0$. However, all qualitative comparisons of the specular properties in this paper remove the division of the Fresnel term from the specular albedo and display a_s'. Our methods also estimate a_s' and F_0 separately and a_s is calculated afterwards.

3.2. Estimation Network

We require an initial estimate for the appearance parameters and therefore alter the original multi-image network [DAD*19] for our purpose. the architecture follows the U-Net architecture by [RFB15] with an accompanied global feature track. The information is compressed by reducing the resolution of a 256×256 resolution input image in each encoder layer, while increasing the number of features and feeding information from the encoder to the corresponding decoder with the help of skip-connections. The global feature track feeds global information back into the network, as complementary information is present in distant regions. To account for the increased complexity of anisotropic materials the maximum feature size of the encoder is increased from 512 for isotropic materials to 1024 while the architecture of the six encoder layers is not changed any further. For the decoder we incorporate the idea of [LXR*18] using a single encoder track to encode the information, but splitting the decoder track into multiple ones. In contrast to the single image network of [DAD*18], we use two decoder tracks, one of which is responsible for the diffuse, specular, lobes, and Fresnel features, and the second decoder track for the normal and rotation features. The single encoder ensures consistent encoding of the input information, while the decoders can each interpret it separately depending on the features within the two paths. We do not split the decoder further to keep connections within the features per decoder track. The structure of each decoder layer is still kept the same as in the original multi-image architecture, where the resolution is doubled in each deocder layer while the feature size is halved. The skip connections from the encoder layers are forwarded to both corresponding decoder layers, where the output of the previous decoder is concatenated with the data of the skip-connection along the feature channel, displayed in Figure 5. The following convolution handles the reduction to the correct feature size. The output of the global feature track of the encoder is also routed to both decoder tracks, where each decoder track has its own global feature track. To match the number of encoder layers, each decoder track has six layers, but as each track is responsible for different amounts of features the channel sizes are adjusted. The first decoder track, responsible for four features, has a maximum output feature size of 512 and the second track, responsible for two features , has a maximum output feature size of 256. These are gradualy reduced to 64 and 32 channels. The output of each single image network therefore consists of two decoder track outputs, each consisting of a U-Net path and an associated feature path.

Since each input image is converted to such an output by a single-image network, maxpooling layers are used to combine all outputs into one output consisting of two U-Net paths with associated global feature paths, displayed in Figure 4. Since the estimation

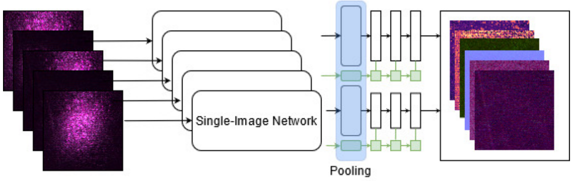

Figure 4: *Structure of the estimation network, handling a flexible amount of input images. Each output path of the single-image networks is responsible for different features and kept seperated from another. The outputs of the single-image networks are combined with maxpooling layers highlighted in blue.*

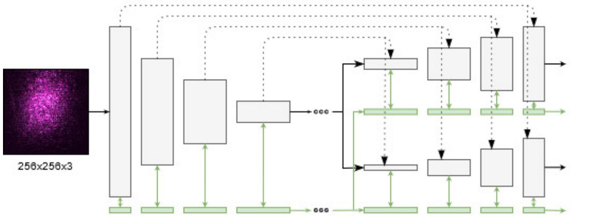

Figure 5: *The structure of our single-image network with two decoder tracks. The feature-track is highlighted in green and the skip connections forward the data to both decoder tracks and is highlighted using dotted lines.*

network is already supposed to generate a first estimation of parameters and still has 64 and 32 channels per pixel of information, three further convolutions are applied to generate the appropriate number of channels per pixel for each of our paths. The first path is responsible for the reconstruction of diffuse albedo, specular albedo, lobe and fresnel maps. Therefore, its information is reduced from 64 channels per pixel to 9 channels per pixel using the three convolutions. In the second path, responsible for the x and y directions of the normals, as well as the rotation values, the number of channels per pixel is reduced to 3.

3.3. Inverse Rendering Framework

Although the estimation network is able to provide an initial estimate of the SVBRDF parameters, it only uses its learned prior assumptions diluting the result. To make optimal use of the information contained in these input images, we adapt and improve the inverse rendering framework[GLP*19] for anisotropic materials, but keep the original network architecture. It improves the initial SVBRDF estimate with the help of an optimization in a latent embedded space provided by an auto-encoder. The encoder provides the latent embeddingt of the parameters and uses 5 layers each using a convolution to increase the feature size, while reducing the resolution. A single batch normalization is performed at the end of the encoder layers for regularization during training to balance the noise and smoothing effects of this regularization. The channel sizes of the encoder layers are 64, 128, 256, 512 and an increased 1024 channels, to account for the increased information necessary for anisotropic materials. The five decoder layers, decoding the latent space back to appearance features use transposed convolutions.

3.4. Optimization

Traditional optimization approaches try to find appearance parameters s which describe a set of input images $\{I\}_i$, minimizing the sum of a loss function L comparing an image I_i to the rendering R of the appearance parameters using the same view- and light-configuration C_i as the input image, displayed in Equation (4). This loss function computes the mean difference between an input image I_i and the rendered appearance parameters in log-space. This optimization of L with respect to s results in a per-pixel minimization, with no connection between the individual parameter maps, leading to parameters incorectly expressing characteristics of other parameter maps. To counter this problem of traditional optimization, Gao's original approach proposes to move the optimization into the latent space provided by an encoder of a pre-trained auto-encoder network. This results in better preservation of connectivity between features, but also suffers from some blurryness in the resulting features. We argue that this optimization of the latent space between the encoder and decoder is not strong enough, especially for our more complex scenario where some of the Fresnel and anisotropy features are only visible in fine details of the input images. We therefore fine-tune the decoder of the auto-encoder instead of optimizing the latent vector z to estimate the appearance parameters.

$$\arg\min_s L(I, R(s,C)) \tag{4}$$

$$L = \sum_i \text{mean}(|\log(I_i + 0.01) - \log(R(s,C_i) + 0.01)|) \tag{5}$$

We compute the loss in the same way as in the traditional optimization, except that we no longer directly optimize the appearance parameters, but rather train the decoder using the loss shown in Equation (6). We only use the input images and their configurations to evaluate the performance of our decoder.

$$L = \sum_i \text{mean}(|\log(I_i + 0.01) - \log(R(D(z),C_i) + 0.01)|) \tag{6}$$

Additionally we make use of the traditional optimization in order to display even fine details in the features that have been lost due to the smoothing effect of the architecture of the auto-encoder.

3.5. Image Resolutions

We enable our methods to predict SVBRDFs of materials with any resolution. Without any adjustments, our estimation network is only able to provide SVBRDFs with a resolution of 256 × 256. Therefore, we use a sliding window method to divide the input images into 256 × 256 slices. We overlap these slices up to 192 pixels, resulting in many estimates for one pixel. We average these estimates and merge all the estimates back together to regain the original resolution.

In the next step, we would like to optimize this initial SVBRDF estimate using the auto-encoder. However, due to its architecture it is limited to slices that are a multiple of 2^5. To solve even this limitation and allow any resolution, we add a padding to the input to change the resolution to a multiple of 2^5. This padding is then removed before being returned by the network, thus restoring the original resolution.

3.6. Image Configuration Selection

In order to capture the reflectance behavior of materials using only a few images, the selection procedure of view- and light-configurations is of great importance, as [LDPT17] has already stated. It aims to limit the large space of possible configurations to those that produce meaningful reflections. The configuration selection presented is not perfect for every material and only serves as a reference to achieve good results in general. Since the reflections caused by the Fresnel terms are only visible under special configurations, we define a Fresnel configuration that particularly tries to highlight them in limited region of the material and rely on the learned connections to apply these information to the other regions. It must also be noted that these fresnel configurations provoke extreme reflections and need to be balanced by the other configurations to avoid burn-in to other features. In addition, we define a configuration that causes anisotropic reflections, for which we need two images of a region on the material. To keep the total number of images low we also use independent configurations to view many areas of the material. For each configuration, we need to define the sampling method for the view- and light-positions. Since we allow materials with non-symmetric sizes, we need to define the configurations depending on the longest side n of a material. Additionally, the positions are distanced from the material according to n to ensure a sufficient number of highlights in the input images.

- **Fresnel-configuration:** A view-position is placed beside the material roughly along the center axes with a distance of up to $0.5n$ and a height of $0.1n$ to $0.4n$ to cause a strong Fresnel reflection across the material. The light-position is arranged in an approximate mirroring configuration.
- **Anisotropic-configuration:** The view-position is sampled from a cosine distribution and a light position is generated in a mirroring configuration. A second configuration is created by rotating the first configuration by 90°. The center of both configurations is moved together to a random position on the material and each view- and light-position is randomly distanced between n and $4n$ away from the material.
- **Independent-configuration:** Both view- and light-positions are moved independently to a random position on the material and are randomly distanced between n and $4n$ away from the material.

4. Implementation Details

After defining the methods, we now discuss the implementation details. We address the used dataset, the training of the learning-based approaches and the optimization implementation.

4.1. Dataset

Most other works, which estimate the appearance parameters of isotropic surfaces make use of SVBRDF datasets containing synthetic data created by designers and artists. While these display a wide variety of materials, they do not capture the variety of real-world objects. We make use of the Bonn SVBRDF Dataset [MHRK19], containing a wide variety of SVBRDFs of many different kinds of real-world fabrics. These SVBRDF parameters are fitted with the help of a highly accurate TAC-7 scanner using the Pantora[XR19] anisotropic textile preset. During the capturing process of the appearance parameters by the scanner, the textiles are rotated five times in 45° steps. Four cameras with different LED point lights turned on are used for each of the rotations. The result is 348 panchromatic images, 100 color images, and some line-lit images taken using a linear light source. All images are used for the fitting process.

The dataset is divided in three sets: A training, validation and an additional challenge dataset. The training dataset contains 300 fabrics and the validation dataset around 60 materials, which we also use as the training and validation data of our training process. The materials vary greatly in resolution and are described by the parameters of the Geisler-Moroder Ward Model with a spatially varying Fresnel reflection term.

4.2. Training Details

With the methods defined we need to train two seperate networks to estimate the final appearance parameters. The estimation network takes input images I with a resolution of 256×256 transformed into log-space using Equation (7) to reduce the dynamic range and scaled and shifted to range $[-1,1]$ as input. We have to set some requirements on the input images, in contrast to other works on isotropic materials, as not all features are visible under every input view- and light-configuration. We discussed the view-light configuration choices in Section 3.6 and use five images, one using the Fresnel-configuration, two images using the anistropic-configuration and two images sampled from the independent-configuration for the training of the estimation network.

$$I = \frac{\log(x+0.01) - \log(0.01)}{\log(1.01) - \log(0.01)} \tag{7}$$

We require some transformations of the features for the input of the auto-encoder and also return the transformed features. The rotation values are transformed from range $[-\pi/2, \pi/2]$ to range $[-1,1]$ and the normals are reduced to their x and y directions. Both networks are trained on 256×256 resolution patches. To judge the performance of the estimation network we make use of an L_1 and render-loss L_r with both weighted equally. The render-loss is crucial in capturing the perceptual performance of the parameters, comparing reference SVBRDF G and the estimated SVBRDF P in log-space using the same view- and light-configuration C and our rendering model R for i view- and light-configurations, displayed in Equation (8).

$$L_\mathrm{r} = \sum_i \mathrm{mean}(|\log(R(G,C_i)+0.01) - \log(R(P,C_i)+0.01)|) \tag{8}$$

We already mentioned that we have to choose the configurations C more carefully to make sure all the features are visible in the renderings, extending the ideas from the work of [DAD*18] for our anisotropic materials. We defined our Fresnel-, anisotropic- and independent-configurations in Section 3.6 and sample two

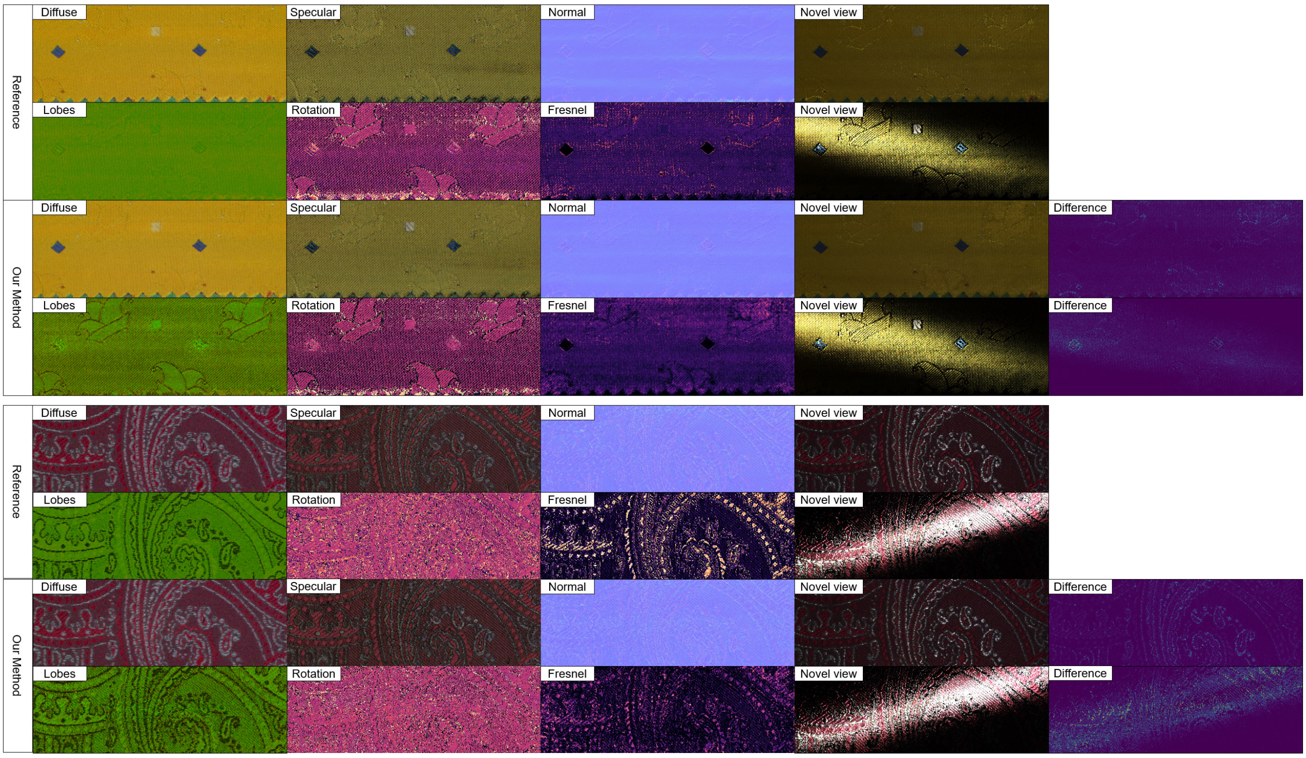

Figure 6: *Estimated features of two materials of the validation dataset using 12 input images, as well as two novel view images rendered with the help of the estimated features. The red material has a resolution of* 1024 × 350 *and the yellow material a resolution of* 1024 × 393*. The difference images show the mean absolute difference between the rendered images of the new view with the reference parameters and the predicted parameters, and show the largest difference in regions with inaccurate Frensel predictions. The x- and y-directions of the lobes are displayed in the R and G channel of the RGB space.*

	Fine-tuned Results configuration α						Refined Results configuration α						Refined Results configuration β					
N	a_d	a_s	n_s	σ	α	F_0	a_d	a_s	n_s	σ	α	F_0	a_d	a_s	n_s	σ	α	F_0
8	0.0004	0.0024	0.0023	0.0072	0.55	0.070	**0.0003**	**0.0022**	**0.0018**	**0.0068**	0.52	**0.070**	0.0008	0.0027	0.0033	0.0079	**0.51**	0.14
10	0.0005	0.0025	0.0023	0.0072	0.55	0.067	**0.0003**	**0.0023**	**0.0017**	**0.0068**	0.52	**0.066**	0.0008	0.0028	0.0033	0.0083	**0.51**	0.15
12	0.0004	0.0021	0.0022	0.0072	0.54	0.06	**0.0002**	**0.0019**	**0.0016**	**0.0068**	**0.50**	**0.059**	0.0006	0.0026	0.0028	0.0074	0.52	0.18

Table 1: *Comparison of fine-tuned results with and without a refinement step using a mean squared difference metric and N ranging from 8 to 12 input images. The evaluation is performed on 58 materials of the validation data set with the differences measured after re-transformation of the features to their original range of values.*

pairs from the anisotropic-configuration focussing on the highlight of specular and anisotropy features, one from the Fresnel-configuration and one from the independent-configuration to sample the configurations for the render-loss. The auto-encoder is trained using the same equally weighted L_1 and render-loss, but additionally requires a smoothness loss term, to make the latent space and the decoding of it less dependant on a perfect input SVBRDF, which we try to esitimate in the first place. We follow the definition and parameters of the smoothness loss of [GLP*19].

We train the estimation network for 300 epochs using a batchsize of 4 and the auto-encoder with a batchsize of 16 for 500 epochs. We make use of an Adam optimizer with a learning rate of 5e−5 for the estimation network and a learning rate of 1e−3 for the auto-encoder.

To fine-tune the decoder, we use an Adam optimizer with a learning rate of 7.5e−4 and 5000 optimization steps followed by 500 refinement steps with a learning rate of 1e−4. The selected learning rate is of crucial importance and must be chosen carefully, since a value which is too low leads to unoptimal results and a learning rate which is too high breaks the optimization.

5. Results

We focus our evaluation on materials with different resolutions. All evaluations are performed on a system with an RTX 2080ti and use the validation Bonn SVBRDF dataset by [MHRK19] further described in Section 4.1. The runtime varies depending on the number of input images as well as the resolution, but to give an estimate,

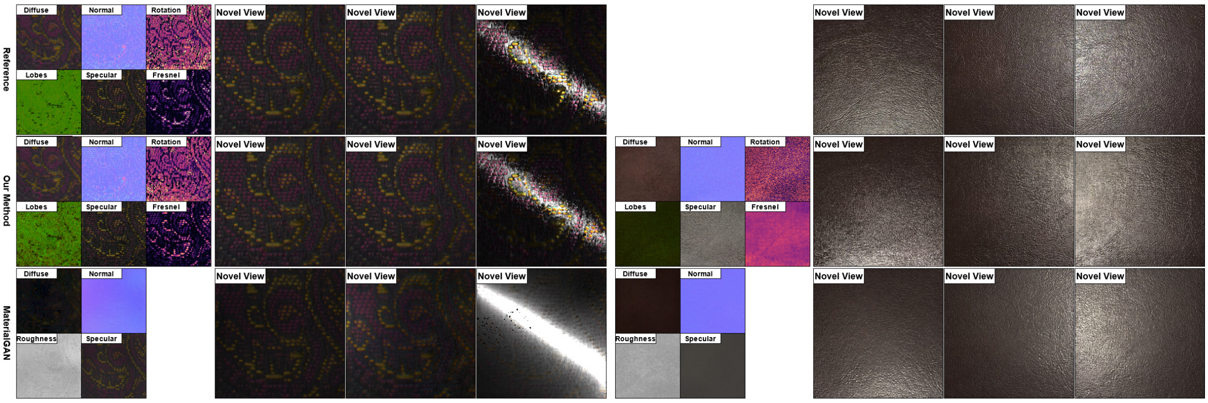

Figure 7: *A comparison between our method and the MaterialGAN method for a synthetic and a real material. For the synthetic material on the left, we use 10 input images and use 6 input images for the real-world material on the right. The MaterialGAN method is not able to reproduce reasonable novel view images for the synthetic material, while our method is able to produce reasonable feature predictions for both material.*

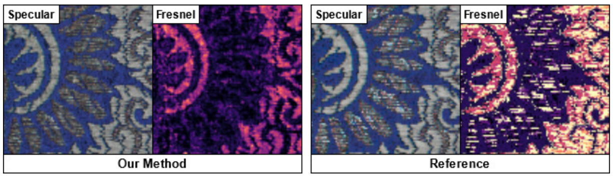

Figure 8: *Comparison of our estimated specular and Fresnel values with the reference values, showing more consistent predictions using our method.*

the evaluation for a 256 × 256 material tile takes about 7 minutes. The two-dimensional lobes are displayed in the R and G channels of the RGB images. We transform the rotation feature to the range [0, 1] and plot the rotation and Fresnel feature using a perceptualy uniform color map.

5.1. Synthetical Results

We use the materials of the validation dataset to generate 12 input images for each material. The choice of view- and light-configurations is crucial to estimate this amount of features with only these few input images, previously discussed in Section 3.6. We sample 2 inputs from the Fresnel-configuration, 2 from the independent-configuration and the rest from the anisotropic-configuration. We present two exemplar results in Figure 6, showing the great ability to predict the additional features, without compromising the other features predicted. It is noticeable across all results that the lobes in particular are estimated differently than specified in the reference and more clearly represent the structures of the materials in our estimation. Meanwhile the added rotation properties are very accurately captured and reconstructed with our method. The Fresnel values can also be reconstructed well, but do not quite reach the instensity of the reference values. Since we calculate the undivided specular value in our network and divide it by the predicted Fresnel value afterwards, the connection of the two parameters poses an additional challenge for the optimization. The Fresnel values in the dataset are not always optimal either, which again shows the difficulty of computing these properties. In Figure 8 we show how the computation of reflections with spatially varying Fresnel values challenges Pantora during the acquisition of the dataset, where a contiguous part within the material cutout undergoes drastic sudden changes in Fresnel values without any structural connection. However, our network is often able to unify this mapping and produce more coherent results. To show the results of our method with and without refinement and the effect of the input configuration selection, we compare the results over our validation dataset for different amounts of input images. The dataset contains 58 materials with an average resolution of 553 × 822 in Table 1. The α configuration uses the previously described combination of configurations, while the β configuration uses only the independent- and mirror- configurations, which leads to slightly worse results for most parameters.

5.2. Comparison

Here we compare our method to another state-of-the-art learning based method, with the features displayed in Figure 7. We provide both methods with 10 input images for the synthetical material and 6 input images for the real-world images. One problem with the comparison is the rendering models used. We use a variant of the Ward model and the MaterialGAN uses a microsurface BRDF with a GGX normal distribution. While other methods are concerned with capturing isotropic surfaces, the goal of this work is to extend them with additional parameters to be able to capture reflectance behavior that cannot be represented by the other methods. For the synthetic material, we provide input images to our method with the same combination of configurations as described in Section 5.1, but need to provide collocated view- and light-configurations for the MaterialGAN since ours tend to break the method. The real-world images are taken from the original MaterialGAN paper and represent colocated view- and light-configurations, on which our method has never been trained and yet is still able to achieve reasonable results. The MaterialGAN with its generator is able to generate SVBRDF parameters from its latent space which closely represent the input images. The problem with our synthetic material is that it does not fall into the space of SVBRDF parameters for which the MaterialGAN has been trained. Thus, there is no combination of SVBRDF parameters that the MaterialGAN generates which are able describe all of the input images for the left material. This is most obvious in the novel views, where a clear difference can be seen. Especially in the areas that are far away from the specular highlight, a difference in intensity is visible. In the novel views with configurations that focus on Fresnel effects, it is immediately apparent that the MaterialGAN is not able to represent this type of reflection with its model, while our method provides a much more convincing result with the estimation of the additional features.

5.3. Limitations

Although our method can achieve results ranging from plausible to near perfect for most materials depending on the input images, there is one type of material that has proven to be particularly difficult for our method. Such material has many small irregular complex reflective regions that are not coherent. For the human observer it is clear that such regions should have the same reflective behavior, but the irregularity in size and shape leads to the network not being able to determine these regions correctly. However, if the set of input images is large enough and captures the reflections of the small regions, these can also be accurately reproduced. Another factor that is crucial for determining the appearance parameters and can hinder the results are the selected input images. Since the estimated features do not have to be present in every image, our method can only recover them to a limited extent by making use of the learned priors if there is not enough information about these features in the input images.

6. Conclusion and Future Work

In this work, we have shown that we are able to detect even highly complex reflections using learning-based methods. In particular the reflections requiring an anisotropy and Fresnel feature. The training of the learning-based methods faces special challenges here, since not all features have to be recognizable in every image. We are able to define both the training conditions and the input data sufficiently to correctly estimate the features. But there is still room for improvement. Even though we are able to predict arbitrary sizes of materials, we are still limited by the GPU memory. Since this is mainly stressed by the render model used, more efficient models and hardware advances could remove this limitation as well. Furthermore, our method also depends on the choice of input images. If these are poorly chosen, the result can suffer. In the future, adapting a GAN to estimate these materials could be beneficial, but even greater care is needed during the training. Additionally, adapting our SVBRDF estimating methods for non-planar material samples and bidirectional texture functions [SSWK13] would be of interest in the future.

References

[ALL20] Asselin L.-P., Laurendeau D., Lalonde J.-F.: Deep svbrdf estimation on real materials. *2020 International Conference on 3D Vision (3DV)* (Nov 2020). 2

[BBJ*21] Boss M., Braun R., Jampani V., Barron J. T., Liu C., Lensch H. P. A.: Nerd: Neural reflectance decomposition from image collections, 2021. 2

[BJK*20] Boss M., Jampani V., Kim K., Lensch H. P., Kautz J.: Two-shot spatially-varying brdf and shape estimation. *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* (Jun 2020). 2

[BJLPS17] Bojanowski P., Joulin A., Lopez-Paz D., Szlam A.: Optimizing the latent space of generative networks, 2017. 2

[BL19] Boss M., Lensch H. P. A.: Single image brdf parameter estimation with a conditional adversarial network, 2019. 2

[DAD*18] Deschaintre V., Aittala M., Durand F., Drettakis G., Bousseau A.: Single-image svbrdf capture with a rendering-aware deep network. *ACM Transactions on Graphics (SIGGRAPH Conference Proceedings) 37*, 128 (August 2018), 15. 2, 3, 5

[DAD*19] Deschaintre V., Aittala M., Durand F., Drettakis G., Bousseau A.: Flexible svbrdf capture with a multi-image deep network. *Computer Graphics Forum (Proceedings of the Eurographics Symposium on Rendering) 38*, 4 (July 2019). 2, 3

[DAW01] Dror R. O., Adelson E., Willsky A.: Recognition of surface reflectance properties from a single image under unknown real-world illumination. 2

[GLP*19] Gao D., Li X., Peers P., Xu K., Tong X.: Deep inverse rendering for high-resolution svbrdf estimation from an arbitrary number of images. *ACM Transactions on Graphics 38*, 4 (July 2019). 2, 4, 6

[GMD10] Geisler-Moroder D., Dür A.: A new ward brdf model with bounded albedo. *Computer Graphics Forum 29*, 4 (2010), 1391–1398. 2

[GSH*20] Guo Y., Smith C., Hašan M., Sunkavalli K., Zhao S.: Materialgan. *ACM Transactions on Graphics 39*, 6 (Nov 2020), 1–13. 2

[LDPT17] Li X., Dong Y., Peers P., Tong X.: Modeling surface appearance from a single photograph using self-augmented convolutional neural networks. *ACM Transactions on Graphics 36*, 4 (Jul 2017), 1–11. 2, 5

[LXR*18] Li Z., Xu Z., Ramamoorthi R., Sunkavalli K., Chandraker M.: Learning to reconstruct shape and spatially-varying reflectance from a single image. *ACM Trans. Graph. 37*, 6 (Dec. 2018). 2, 3

[MHRK19] Merzbach S., Hermann M., Rump M., Klein R.: Learned fitting of spatially varying brdfs. *Computer Graphics Forum 38*, 4 (July 2019). 1, 5, 6

[RFB15] Ronneberger O., Fischer P., Brox T.: U-net: Convolutional networks for biomedical image segmentation, 2015. 3

[Sch94] Schlick C.: An Inexpensive BRDF Model for Physically-based Rendering. *Computer Graphics Forum* (1994). 3

[SSWK13] Schwartz C., Sarlette R., Weinmann M., Klein R.: Dome ii: a parallelized btf acquisition system. In *Proceedings of the Eurographics 2013 Workshop on Material Appearance Modeling: Issues and Acquisition* (2013), pp. 25–31. 8

[XR18] X-Rite: Tac7-scanner. http://web.archive.org/web/20180615015942/https://www.xrite.com/categories/appearance/tac7, 2018. Accessed: 2021-01-24. 1, 2

[XR19] X-Rite: Pantora material hub. https://web.archive.org/web/20190424232441/https://www.xrite.com/categories/appearance/pantora-software, 2019. Accessed: 2021-01-24. 5