



UNIVERSITÄT
DES
SAARLANDES



max planck institut
informatik

Saarland University
Faculty of Natural Sciences and Technology I
Department of Computer Science

Master Thesis

Reflectance measurement and Fabrication tools for gloss editing

Sebastian Cucerca

May 6th, 2019

Advisors

Michal Piovarci
Dr. Vahid Babaei

Supervisor

Prof. Dr. Piotr Didyk

Reviewers

Prof. Dr. Piotr Didyk
PD Dr. Karol Myszkowski

Declarations

Eidesstattliche Erklärung / Statement in Lieu of an Oath:

Ich erkläre hiermit an Eides Statt, dass ich die vorliegende Arbeit selbstständig verfasst und keine anderen als die angegebenen Quellen und Hilfsmittel verwendet habe.

I hereby confirm that I have written this thesis on my own and that I have not used any other media or materials than the ones referred to in this thesis.

Unterschrift / Signature:

Sebastian Cucerca

Einverständniserklärung / Declaration of Consent:

Ich bin damit einverstanden, dass meine (bestandene) Arbeit in beiden Versionen in die Bibliothek der Informatik aufgenommen und damit veröffentlicht wird.

I agree to make both versions of my thesis (with a passing grade) accessible to the public by having them added to the library of the Computer Science Department.

Unterschrift / Signature:

Sebastian Cucerca

Abstract

Real-world objects have different surface properties. Their appearance can range from highly glossy to matt. Furthermore, surfaces can exhibit isotropic or anisotropic features. Unfortunately, ordinary printers are usually capable of reproducing only colors. They are not able to vary the surface reflectances within a scene as the substrate material predetermines it. Additionally, the reflectance properties for materials in an image are not always given.

For classifying the BRDF of surfaces, an image-based measuring device is built in this thesis. The cylindrical setup can measure anisotropic BRDFs. Both the hardware and software for the implemented system are explicitly described. It is intended to make it easy for other researchers to rebuild and use it. Additionally, a method for gloss fabrication is implemented. It can produce anisotropic BRDFs by scratching one-dimensional lines with a scalpel into surfaces. Both the measuring setup and fabrication method are used to build a model for predicting fabrication parameters to reproduce the appearance of a given material.

To my parents, Aurora and Valentin. I could not have done this without you.
Thank you for all of your love and support along the way.

Acknowledgements

I especially want to thank Piotr Didyk, Vahid Babaei and Michal Piovarci, who always helped me with their advice and supported me with any problem I faced. I value the vast amount of time and effort they put into this thesis and express my most profound appreciation to all of them.

Furthermore, I would like to thank Karol Myszkowski for being my second reviewer.

At last, I want to thank Alexander Speicher, Gian-Luca Kiefer, Marc Habermann, and Maximilian Augustin for their support during the whole time.

Contents

1	Introduction	1
2	Background	5
2.1	BRDF	5
2.2	HDRI	6
2.3	Camera Geometry	7
3	Related Work	9
3.1	Reflectance Measurements	9
3.2	Gloss Fabrication	12
4	BRDF Measurements	15
4.1	Hardware Setup	15
4.2	Image Acquisition	21
4.3	Data Acquisition	24
4.4	Diffuse Normalization	48
4.5	Function Fitting	50
5	Fabrication	53
5.1	Hardware	53
5.2	Software	60
6	Modeling	67
6.1	Reproduction Model	67
6.2	Model Computation	69
7	Evaluation and Discussion	73
7.1	BRDF Measurements	73
7.2	BRDF Reproduction	77
8	Conclusion	87
9	Future Work	89
9.1	BRDF Space Expansion	89
9.2	Alternative Fabrication Methods	89
	Bibliography	91
	Appendices	93

List of Figures

1.1	Example for different reflectance properties	1
2.1	Visualization of BRDF directions	6
3.1	Measuring setup by Karner et al.	9
3.2	Measuring setup by Marschner	10
3.3	Measuring setup by Havran et al.	11
4.1	Image of the proposed measuring setup	16
4.2	Effect of the diffuser on the LED pattern	17
4.3	Image of the sample part	18
4.4	Circuit diagram of the measuring setup	20
4.5	Example captures for a specular material	22
4.6	Concept of a CFA	24
4.7	Bayer CFA and corresponding color indices	25
4.8	Examples of the color index arrays	26
4.9	Processing pipeline for calibration images	28
4.10	Steps of corner detection	30
4.11	Result of raytracing	36
4.12	Demonstration of the LED fluctuations	38
4.13	Result of an HDR regression	41
4.14	Processing pipeline for sequence images	42
5.1	Example for concept of line geometry	53
5.2	Image of scalpel holder	59
5.3	Example for juxtaposed halftoning	61
5.4	Differently filled lookup screens	62
5.5	Example for concept of line extension and shifting	63
5.6	Example for concept of line pattern	64
5.7	Example for extraction of line segments	65
6.1	Pipeline of reproduction model	67
6.2	Rendered base BRDFs	71
7.1	Illustration of measured example materials	74
7.2	Visualization of measured BRDF subspace	76
7.3	Fitted ETN curves for second base and different methods	77
7.4	Rendered images of the fitted sample BRDFs	78
7.5	Fabricated gradient image	79
7.6	Rendered images of original and reproduced materials	80
7.7	Fabricated rendered image	81
7.8	Fabricated continuous-tone image	82
7.9	Sequence images of sample 6/6	83
7.10	Comparison of images from samples 4/6 and 5/6	84
9.1	Anisotropic effect of FDM print	90

List of Tables

4.1	Camera parameters for captures	21
4.2	Optimization parameters for Ward fitting	51
5.1	List with tested pens	55
5.2	List with tested varnishes	56
5.3	List with tools used for scratching	58
5.4	List with parameters for juxtaposed halftoning	62
6.1	List with fabrication parameters for sample generation	69
6.2	List with nominal coverages	69
6.3	Optimization parameters for model fitting	71
6.4	List with fitted effective and nominal coverages	72
6.5	Optimization parameters for ETN fitting	72
7.1	Cross-validation errors for different modeling methods	77
9.1	Fitted Ward parameters of materials used in the thesis	93
9.2	Fitted Ward parameters of second base during cross-validation	93

List of Symbols

$CI_{x,y}^{raw}$	Raw calibration image
$n_{\#}$	Number of sequences
$m_{\#}$	Number of exposure times
$SI_{n,m,x,y}^{raw}$	Raw sequence images
$LI_{x,y}^s$	Saturation level per pixel
$LI_{x,y}^b$	Black level per pixel
$CI_{x,y}^{lin}$	Linearized calibration image
$SI_{n,m,x,y}^{lin}$	Linearized sequence images
$ID_{x,y}^{cid}$	Measured color index
$CI_{x,y,c}^{col}$	Split CFA into color channels
K_c	Kernel for interpolating color c
$CI_{x,y,c}^{rgb}$	Demosaiced calibration image
P_d^{ul}	Upper left corner of RoI
P_d^{lr}	Lower right corner of RoI
$CI_{x,y,c}^{roi}$	RoI image of normalization marker
$CI_{x,y,c}^{hsv}$	HSV image of normalization marker
t_c^u	Upper threshold limit
t_c^l	Lower threshold limit
$CI_{x,y}^{thr}$	Binary image threshold image
C_i	Contours
C_i^a	Contour areas
$c_{\#}$	Number of contours
C^m	Contour of normalization maker
$CI_{x,y}^{cont}$	Image containing clean normalization marker
$P_{i,j}^{hc}$	All detected harris corners
P_i^{cp}	Corner points of normalization marker
P_i^{2D}	Sorted corner points of normalization marker
w^{nm}	Width of normalization marker
h^{nm}	Height of normalization marker
d^{nm}	Distance between holder center and normalization marker
P_i^{3D}	Corresponding 3D points to P_i^{2D}

TV_{hold}^{cam}	Translation vector of holder in camera space
RM_{hold}^{cam}	Rotation matrix of holder in camera space
TV_{cam}^{hold}	Translation vector of camera in holder space
RM_{cam}^{hold}	Rotation matrix of camera in holder space
r^c	Cylinder radius
$w^c m$	Distance between two cylinder markers
h^u	Height of the cylinder markers upper end
h^l	Height of the cylinder markers lower end
TV_{cyl}^{hold}	Translation vector of cylinder in holder space
RM_{cyl}^{hold}	Rotation matrix of cylinder in holder space
h^{img}	Height of the image in pixels
w^{img}	Width of the image in pixels
$P_{i,d}^{ip}$	Ray directions in image plane
$P_{i,d}^{cp}$	Ray directions in camera space
$P_{i,d}^{ud}$	Undistorted ray directions in camera space
$r^{\{2,4,6\}}$	Tangential distortion coefficients
$k_{\{1,2,3\}}$	Radial distortion coefficients
P^{ro}	Ray origin in holder space
P_i^{ro}	Ray directions in holder space
P_i^{sp}	Intersection points for rays
$SI_{n,x,y}^{comb}$	Combined sequence images
ID^{nb}	Pairs of IDs for neighboring sequence images
t^p	Threshold for neighbor correction
\bar{R}	Mean ratio between two neighboring sequence images
fac	Correction factor for two neighboring sequence images
$SI_{b,x,y}^{corr}$	Correction sequence images
$CI_{x,y}^{cl}$	Binary image containing patch contour
K_4	Kernel for morphological dilation
$CI_{x,y}^{morph}$	Binary image containing thickened patch contour
$CI_{x,y}^{ic}$	Binary image containing normalization patch interior
avg^{in}	Average HDR values of normalization patch interior
$SI_{x,y}^{hdrcorr}$	Corrected HDR image
I_i^{hdr}	HDR values of intersecting pixels
d^{ls}	Distance between light source and cylinder center axis
TV_{ls}^{hold}	Translation vector of light source

f^w	Ward BRDF
$f^{w'}$	Extended Ward BRDF
\angle^l	Rotation of lobe
f^{lamb}	Lambert BRDF
ρ^{diff}	Albedo of Lambert material
ρ^d	Diffuseness of material
ρ^s	Magnitude of lobe
α^x	Width of lobe in x direction
α^y	Width of lobe in y direction
f_m^{lc}	Linear combination of two bases
cn_m^{b1}	Nominal coverage for base 1
cn_m^{b2}	Nominal coverage for base 2
ce_m^{b1}	Effective coverage for base 1
ce_m^{b2}	Effective coverage for base 2
f^{clip}	Peak reflection clipping function
f_m^{etn}	ETN function for second base
L	Lines for scratching
\angle^a	Rotation angle for scratching lines
a	Line width in juxtaposed halftoning
b	Line height in juxtaposed halftoning
T	Screen thickness in juxtaposed halftoning
ls	Vector consisting of a and b

List of Abbreviations

LDR	Low dynamic range
HDR	High dynamic range
HDRI	High dynamic range imaging
BRDF	Bidirectional reflectance distribution function
DSLR	Digital single-lens reflex camera
LED	Light-emitting diode
RoI	Region of interest
PNP	Perspective-n-Point

List of Operands

$\ \cdot\ _2^2$	L2-norm of a vector
\oplus	Dilation operator
$\langle \cdot, \cdot \rangle$	Dot product between two vectors

1 Introduction

Nowadays inexpensive 2D printers are widespread and available for most people. They are used to reproduce digital models of the real world like images captured by cameras. Here, the focus usually lies on the high-quality reproduction of the scene colors. However, object surfaces can have a wide range of reflectance properties independent on their colors. For example, metallic surfaces appear more glossy while organic ones are usually rather matt. Figure 1.1 shows examples of objects with different appearances in the same scene.

Unfortunately, printers are not able to accurately reproduce reflectance properties. That is because the substrate, i.e., the material we print on, defines most of the reflectance. The same problem also arises for other kinds of devices like 3D printers. Again the reflectance is given by the material used for printing. Even though many 3D printers can print with a couple of different materials, it is still challenging to model the objects accurately. Especially anisotropic surfaces are hard to reproduce. The reflection of these materials is not rotationally symmetric and thus directionally dependent. Velvet is an example of such material. Producing anisotropy requires introducing special geometries during fabrication and thus further increases the challenge.



Figure 1.1: *Example scene containing objects with different reflectance properties.*

Furthermore, when we take an image of a scene, it contains only implicit information about the reflectances of objects. Unfortunately, it is often not possible to explicitly define the reflectance property for a particular object. The reason is that usually, there is not enough information about the lighting setup. Additionally, the material of an object is captured only from a limited amount of angles which makes it hard approximating the reflectance accurately. Especially, making any point about the properties of anisotropic materials requires even more information.

Thus, in this thesis, a device was built with the capability to measure anisotropic reflectance properties of any bendable material. More specifically, it has a cylindrical setup which uses a digital camera to capture the amount of light reflected by the material from different angles. For the illumination, several LED light sources placed at various locations were used. With the aid of the material captures and knowledge about the controlled environment, it is possible to approximate the bidirectional reflectance distribution function. Additionally, a framework was implemented for this purpose. The device is low-cost and easy to reproduce for anyone.

Moreover, several fabrication methods for gloss editing of materials were investigated. Experiments have shown that scratching one-dimensional lines into a substrate generates the most promising results regarding anisotropy and the change of appearance. For predicting the effect of scratching a suitable model was developed. More specifically, multiple samples generated with the fabrication method are classified with the reflection measuring device. The model is built on top of these measurements and can approximate the required amount of scratches to reproduce the appearance of other materials. Here, the focus lies primarily on anisotropic ones. Even though the presented approach demonstrates the reproduction in 2D, it is possible to extend it to 3D.

Contributions The main contributions of this thesis are both the building of a BRDF measurement system and the development of a reproduction method for fabricating gloss. More detailed the contributions are as follows.

- Building, a low-cost BRDF measurement system, consisting of multiple light sources, a DSLR and the necessary electronic part to control the components.
- Implementing a framework that estimates anisotropic BRDFs from images captured by the setup. The framework includes a geometric and radiometric calibration as well as the final fitting of the BRDF parameters.
- Investigating multiple methods for gloss fabrication.
- Building a model that uses scratching as a fabrication method and predicts how to use it for reproducing a given set of BRDF parameters.

Overview The thesis is divided as follows. First, related work is presented to give an overview of associated works that already tackled the mentioned problems. The reproduction pipeline consists of three different parts. The first part focuses on measuring the reflectance properties of materials. Chapter 4 explains the device in a way that anyone can rebuild and use it. The second part is about fabrication methods for editing gloss properties. Chapter 5 provides an overview of the investigated fabrication tools together with their advantages and disadvantages. Chapter 6 is about the third part of the pipeline. It explains how the model was built for reproducing different material reflectances. In the end, both the measurement setup and reproduction methods are evaluated in chapter 7. Finally, the last chapters give a summary of the presented work and an outline over future work.

2 Background

This chapter provides a reminder regarding the necessary background for the concepts used in this thesis. The explanations include bidirectional reflectance distribution functions, high dynamic range imaging, and camera geometry.

2.1 BRDF

In the real world, various materials behave differently under identical lighting conditions. For example, a metallic and a paper surface. Illuminated from the same direction with equal light intensity, reflects a different amount of light to a specific direction. This comes from the fact that the surface absorbs some of the incident light and reflects it differently. Another critical factor is the viewing direction. The reflection is dependant on the viewer and the light source position relative to the surface normal. The bidirectional reflectance distribution function (BRDF) describes the reflection property of a material. More precisely, it describes how much light is reflected from an incident direction to another one. A BRDF is usually defined as

$$f(\theta^{in}, \phi^{in}, \theta^{out}, \phi^{out}) \quad (2.1)$$

It specifies for each pair of incident light and outgoing viewing directions the reflected amount of the reflected light energy. Some BRDFs additionally make use of the halfway vector which is defined as the vector between the light and viewing directions[1]. Each direction is defined by its zenith θ and azimuth ϕ angles. The azimuth is depending on a tangent vector. A visualization of the concept is depicted in figure 2.1. A physically-based BRDF must fulfill the following properties.

$$f(\theta^{in}, \phi^{in}, \theta^{out}, \phi^{out}) \geq 0 \quad (2.2)$$

$$f(\theta^{in}, \phi^{in}, \theta^{out}, \phi^{out}) = f(\theta^{out}, \phi^{out}, \theta^{in}, \phi^{in}) \quad (2.3)$$

$$\int_{\Omega} f(\theta^{in}, \phi^{in}, \theta^{out}, \phi^{out}) \cos(\theta_o) d\omega_o \leq 1 \quad (2.4)$$

The so-called Helmholtz reciprocity from equation 2.3 is one of the main characteristics of physically based BRDF's. It describes the symmetry between both the incident and reflected directions. This means the directions can be swapped without changing the function's value. Equation 2.4 ensures energy conservation such that there is not more energy reflected than incoming. Additionally, a BRDF must be non-negatively valued according to equation 2.2.

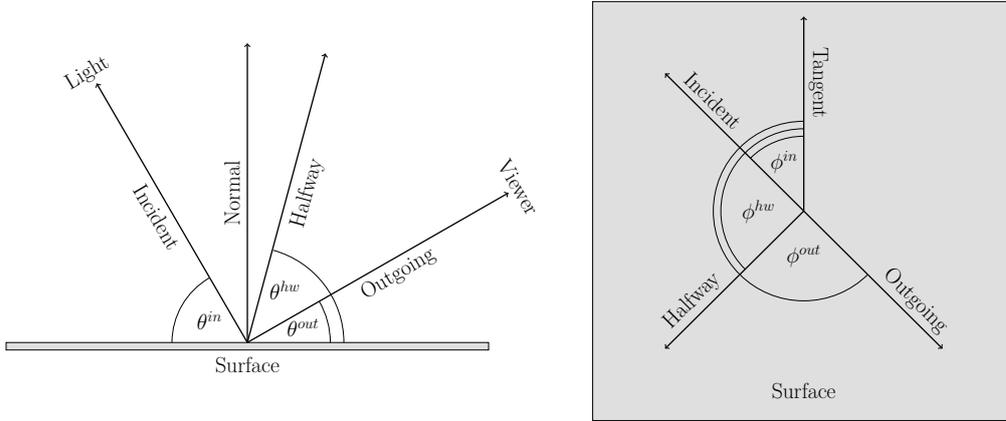


Figure 2.1: Visualization of BRDF directions. Side view(left) depicts θ angles while top view(right) shows ϕ angles of the respective directions.

Ward Model There exist many analytical BRDF models. This thesis focuses on Ward model[2]. It is widely used in the computer graphics community and can be defined by only a few parameters. Furthermore, it can model anisotropic surfaces. This is one of the main requirements since the goal of this thesis is the reproduction of anisotropic reflectances. The ward model is defined as follows.

$$f^w(\theta^{in}, \theta^{out}, \theta^{hw}, \phi^{hw}) = \frac{\rho^d}{\pi} + \frac{\rho^s}{4\pi\alpha^x\alpha^y\sqrt{\cos\theta^{in}\cos\theta^{out}}} e^{-\tan^2\theta^{hw}\left(\frac{\cos^2\phi^{hw}}{(\alpha^x)^2} + \frac{\sin^2\phi^{hw}}{(\alpha^y)^2}\right)} \quad (2.5)$$

where the first term denotes the diffuseness. A Gaussian lobe defines the specularity. Here, ρ^s is the magnitude of the lobe while α^x and α^y control its width in each of the two principle directions[3]. Note that it expresses an isotropic BRDF in case of $\alpha^x = \alpha^y$.

2.2 HDRI

The dynamic range of cameras is limited. More specifically, it can be problematically to take a well-illuminated photo of a scene that has a wide range of different brightness levels. For example, choosing a proper exposure time for a landscape photo is often not easy. In most of the cases, the sky is brighter than the ground. An exposure time that does not overexpose the sky probably underexposes the ground. Conversely, an exposure time that fits for the ground overexposes the sky.

High Dynamic Range Imaging (HDRI) can help to handle this problem [4]. The basic idea is to capture multiple low dynamic range (LDR) images at different exposures. Here, each part of the scene is well-illuminated at least in one exposure. In the end, the LDR images have to be combined into one HDR image.

2.3 Camera Geometry

Cameras are used to generate two-dimensional images of the real three-dimensional world. This thesis uses the model of a so-called pinhole camera. It projects 3D points from the real world onto the camera's image plane using a perspective transformation. First of all the 3D world coordinates are translated into the camera coordinate space by using the extrinsic parameters of the camera. The extrinsic parameters describe the camera location in the world coordinate system and are defined as follows

$$TV_{cam}^{world} = \begin{pmatrix} t_x \\ t_y \\ t_z \end{pmatrix} \quad (2.6)$$

$$RM_{cam}^{world} = \begin{pmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{pmatrix} \quad (2.7)$$

where TV is the translation vector and RM the rotation matrix. The superscript denotes the reference coordinate system while the subscript is the object. Additionally, the camera intrinsic parameters are necessary for the projection of the camera coordinates to pixel coordinates in the image frame. The intrinsic parameters are not dependent on the viewed scene and just the internal specifications of the camera. Furthermore, they are defined for each camera lens pair and have to be recomputed when one of both is changed. They are described as

$$A = \begin{pmatrix} f_x & 0 & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{pmatrix} \quad (2.8)$$

where f_x and f_y denote the focal lengths expressed in pixel units. Furthermore, c_x and c_y is the principal point which is usually at the image center and express in pixel units as well. The full projection of a point P^w from the 3D world coordinate system to its corresponding point in the 2D image plane is defined as

$$A \cdot (RM_{cam}^w \cdot P^w + TV_{cam}^w) \quad (2.9)$$

Further information to the procedure and an additional illustration of the pinhole camera model can be found on the respective OpenCV documentation¹.

¹https://docs.opencv.org/2.4/modules/calib3d/doc/camera_calibration_and_3d_reconstruction.html

3 Related Work

The field of image-based BRDF measurement has been widely researched. Many previous works successfully solved the problem. I present a couple of them in this chapter and discuss their strengths and weaknesses. Different components of the methods are combined and used as a base for the proposed measuring setup. Furthermore, several approaches for gloss fabrication are presented. In addition to the works presented here, some others are discussed in the further chapters.

3.1 Reflectance Measurements

One of the first image based measurement systems was proposed by Ward et al. [2]. They used an inexpensive CCD camera and a fish-eye lens for measuring BRDF data of planar material. More precisely, the camera captures an image of a hemispherical half-silvered mirror which scatters light reflected from the flat sample material. Taking images with a camera is time-saving since it captures many data points with only one shot. Furthermore, using a fish-eye lens captures the entire hemisphere at once. They additionally presented an empirical BRDF model which uses only a few parameters and is capable of modeling anisotropic surfaces. Nevertheless, the system is limited by the used optics. Vignetting degrades the image quality and distorts the measurements. Furthermore, it does not use any reference material which makes it hard to estimate the actual BRDF values considering that the hemisphere is just an approximation of an ideal ellipsoid. However, the proposed BRDF model can represent anisotropic BRDFs reliably while having a low parameter space. Thus, it is used in this thesis for all BRDF fittings.

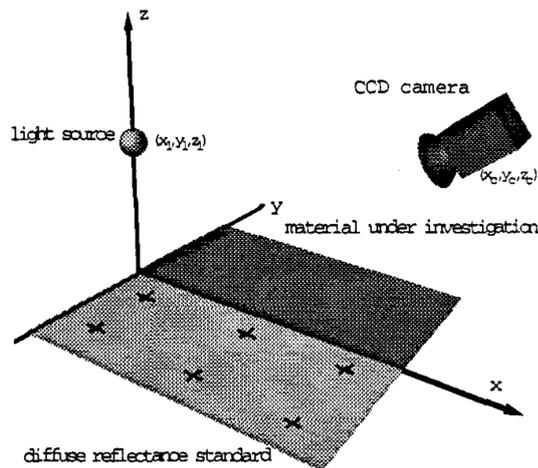


Figure 3.1: *Measuring setup by Karen et al. [5]*

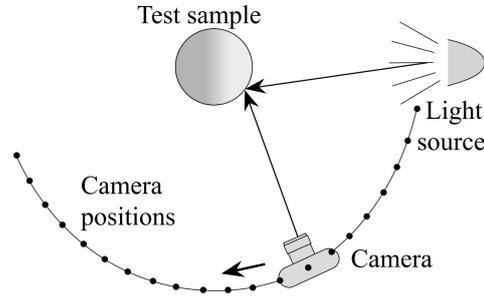


Figure 3.2: *Measuring setup by Marschner et al. [6]*

Karner et al. [5] describes a setup using an inexpensive CCD camera and an incandescent light source for measuring BRDF data of planar materials. More specifically, they put both the material and an additional diffuse material besides each other. The diffuse material is used as reference material. The light source symmetrically illuminates both while the camera captures a single image of the scene. The BRDF is fitted to data extracted from this image. They use Ward’s BRDF model. The geometric calibration, i.e., finding the camera and light positions, is performed by annotating the images manually. An image of their setup is depicted in figure 3.1. The approach has many limitations. First of all, the measurements are based on data from planar materials. While this simplifies the geometric calibration, it provides little data per captured image. Thus, the sampled BRDF space is rather small and does not provide reliable fitting of the BRDF model. Additionally, by capturing the diffuse material in the same image the possible dynamic range is limited. Consequently, highly specular materials cannot be measured since it is not possible to capture both diffuse and specular materials in a well-illuminated way with only one shot. One of them would be over or undersaturated in any case. Annotating positions by hand is not convenient and can be time-consuming especially when measuring many materials. Still, the system is a good starting point, and thus, it is taken as a base with room for improvement. Especially, the idea of using a diffuse reference in combination with a camera and a point-like light source is inherited.

The system by Marschner et al. [6] is an extension to the system implemented by Marschner in his previous work [7]. Similarly, to the other described methods it uses a camera and light source. However, they used markers for the geometric calibration to automatize the position detection. Additionally, the test sample does not have to be flat anymore which allows measuring a much wider BRDF space within a capture. More precisely, they use convex surfaces for this purpose. To further increase the measured space, several captures are taken while moving the camera between them. It increases the amount of sampled outgoing directions. As a reference, they use a diffuse white sample. The system proposed in this thesis is inspired by the idea of Marschner et al. to extend the shape of the samples. Furthermore, the geometric calibration is also inspired

by the usage of markers since it is a convenient solution for the automatization. Still, the second camera makes the system unnecessarily complicated. Thus, an alliterative approach for finding the light position is developed. Moving the light source also makes the system inconvenient. Therefore, multiple static light sources are used in this thesis to still have a wide range of different incident directions without incorporating further mechanic components.

Another system proposed by Havran et al. [8] uses a hemispherical dome. It is portable and measures the BRDF of planar surfaces. For illumination 139, light sources are distributed around the dome. They use six different cameras mounted onto a circular arc which can be moved along the arc. Additionally, the whole dome can be automatically rotated around the sample. Thus, the system has two degrees of freedom providing a high accuracy of measurements. An image of the system is depicted in figure 3.3. The main drawback of the system is the limited ability to measure highly specular surfaces. The reason for the cameras big field of view is the small dome size. This thesis solves this problem by using another approach regarding the camera setup. Although their construction differs significantly from the proposed one, the idea of using multiple light sources was taken over.

Unfortunately, only Havran et al. explain their method explicitly enough to rebuild it. Therefore, this thesis provides a complete and detailed guideline about both the design of the hardware and software for the implemented system. It is intended to make it easy for other researchers rebuilding and using it.

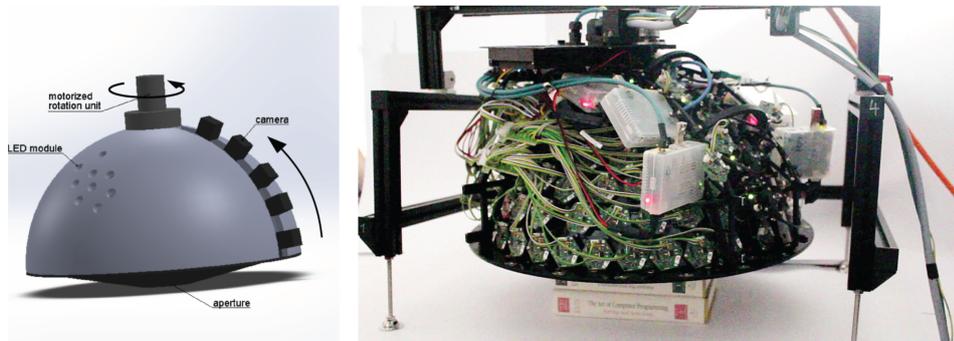


Figure 3.3: *Measuring setup by Havran et al. [8]*

3.2 Gloss Fabrication

In general, there are two approaches to gloss fabrication. The first one is based on the microfacet theory [9, 10]. It assumes that material surfaces are not perfectly smooth but rather consist of small mirroring facets. The composition of many differently orientated microfacets can change the surface reflection. This approach is used by Weyrich et al. [11]. They reproduced spatially-varying BRDFs by milling small highly reflective facets into aluminum. In this way, complex reflectance shapes are possible such that also anisotropy becomes reproducible. However, the spatial details are limited by the device resolution. The method of fabricating gloss by introducing geometries is investigated in this thesis as well. More precisely, the novel approach of affecting reflectance by scratching a material was successfully used for this purpose.

Rouiller et al. [12] proposed a similar system. Rather than microfacets, they use small domes which reproduce the BRDF. The domes are placed on the object and thus affect its reflectance properties. The method can handle anisotropic BRDFs and has a higher spatial resolution compared to the method by Weyrich et al. Still, it has the disadvantage that it cannot produce a continuous surface.

Levin et al. [13] use a method based on wave optics. More specifically, they control the reflections by modulating the phase of incident light waves and thus the diffraction effects. The fabrication procedure relies on Photolithography [14] which etches patterns onto silicon wafers. The method can produce high-resolution and anisotropic samples with micron-sized features. Due to the used fabrication method, only planar facets at two distinct levels are possible which limits the achievable reflectance. Another drawback is the expense factor for fabricating the wafers.

The second approach is the usage of inks as proposed by Matusik et al. [15]. They combined different inks and foils to print with them and thus reproduced a wide range of spatially-varying BRDFs. This method provides a rich spatial resolution. Unfortunately, it is not possible to generate anisotropic effects due to the low spatial resolution and isotropic nature of inks. In this thesis, a similar approach is investigated. Instead of inks, varnishes were used to edit the appearance of flat objects and reproduce spatially varying BRDFs.

Lan et al. [16] combine both microgeometries and inks to a bi-scale approach. More precisely, they first fabricate a small scale height field using a 3D printer. The print is coated with an additional layer of metallic paint to obtain a glossy finish. By further adding inks on top of the paint it is possible to vary the final BRDF. With this procedure, they combine the rich possible spatial details of microgeometries with the high-resolution of ink printing.

Malzbender et al. [17] also propose a combined method. They first introduce a facet microgeometry to a reflective substrate. An opaque ink is printed on top of chosen areas to reduce specular highlights. Thus, it is possible to vary between specular and matt reflectances. The method can fabricate anisotropic BRDFs. Due to the used inks, it is not possible to produce colored samples. Furthermore, the method requires a manual alignment of substrate and printing mask which is inconvenient.

4 BRDF Measurements

The main goal of this thesis is to reproduce materials with a fabrication system. Therefore, characterization of fabricated samples is necessary to build a model. Additionally, BRDFs of unknown materials have to be determined to reproduce them. For this purpose, an image-based measuring system is built which can acquire BRDFs of anisotropic materials.

Building such a setup is a challenging task since it requires a proper geometric calibration for a precise definition of the exact BRDF vectors and angles. Furthermore, we need a radiometric calibration for characterizing the light intensities and keeping them comparable over measurements of different materials. Some materials are highly specular and thus capturing them is challenging due to the limited dynamic range of the camera.

The system uses a fixed DSLR camera for capturing images of a sample which is wrapped around a cylinder. Due to the curvature, it provides continuous variations for two of the four angles of the BRDF, i.e., θ^{in} and θ^{out} . By rotating the sample on the cylinder, it is possible to vary also the other two angles, i.e., ϕ^{in} and ϕ^{out} . To further extend the captured BRDF space, multiple LEDs are used for illuminating the sample from different angles. Since the sample has to be wrapped around the cylinder, the system is capable of measuring the BRDF of every bendable surface. The whole capturing process is automated and run by a micro-controller. Furthermore, a framework for computing the BRDF from the captured images was implemented.

In the following, the hardware setup for the different parts are explained. Additionally, the capturing process and subsequent data acquisition are described. In the end, an overview of the BRDF computation is given.

4.1 Hardware Setup

The setup of the measuring system consists of four base parts which are explained in this chapter together with their assembling. Different pieces of the measuring setup are designed and printed with an Objet Connex 260 and a MakerBot Replicator 2. The Objet printer is loaded with the rigid resin VeroBlackPlus while the Makerbot uses standard black PLA filament.

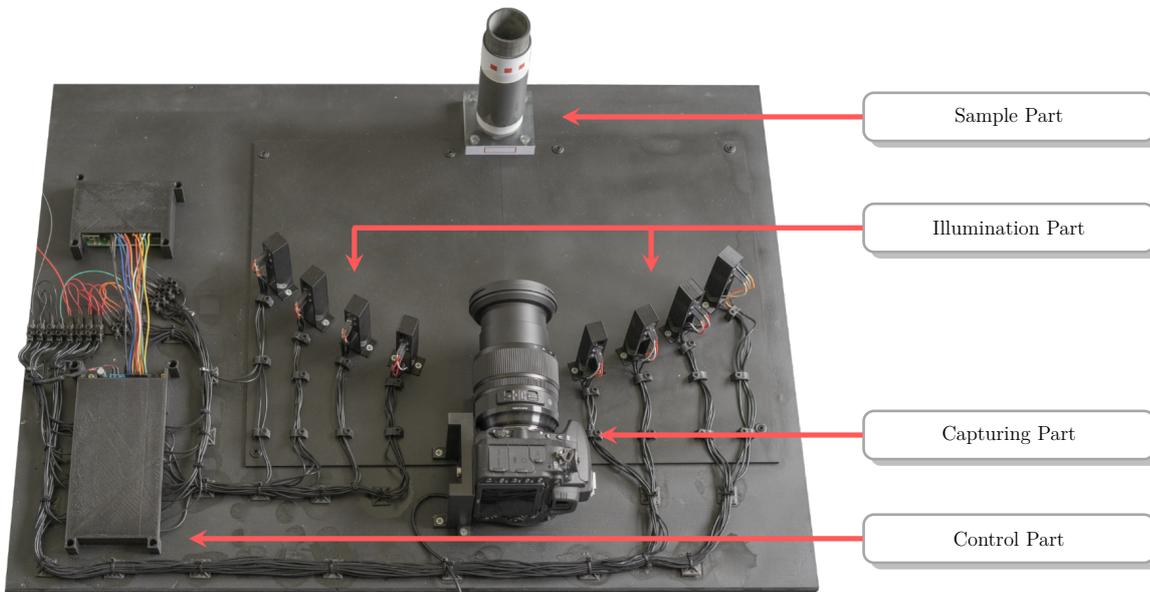


Figure 4.1: *Annotated image of the measuring setup. It shows the wooden board with all assembled parts. The black cover around the board was omitted for better visibility.*

4.1.1 Control part

For an automatic process, parts have to be controlled by one device. For this, a Raspberry Pi controller together with a relay module are used. Unfortunately, they have multiple small integrated LEDs. This additional light would distort the measurements. Here, a black box is printed with the FDM printer and put over both the relay module and controller to block emitted light. All LEDs are additionally covered with black tape to block the light further.

A 12 Volt mains adapter powers the relay module with 6 Ampere. It redirects power from the LEDs mains adapter to the currently used LED. Powering multiple LED at the same time is possible. This is necessary in some cases as explained in 4.2.1. The relay shield has overall 16 channels. Thus, it is capable of controlling 16 different single LEDs.

A 5.1 Volt main adapter powers the Raspberry Pi controller with 2.5 Ampere. The controller runs the software for capturing the images and controls the LEDs over the relay module. Furthermore, it triggers the camera and downloads the images to its memory. In this thesis' setup a 64 gigabyte SD Card is inserted into the controller with Raspbian Stretch Desktop V 4.14 installed on it.

4.1.2 Capturing part

The second part is responsible for capturing the images that are later used for data extraction. A Nikon D750 digital single-lens reflex camera is used as capturing device. It is chosen because it has a full-frame sensor that produces sharp images with little noise at a high ISO. More specifically, the sensor has a size of 35.9 mm x 24 mm with a resolution of 24.93 Megapixel. A Sigma 24-105mm f/4 L lens is attached to the camera. Additionally, the camera is connected to the Raspberry PI with a USB cable to be able to control the capturing process by software.

A holder is designed and printed for fixing both the camera and lens. The holder fixes the camera in a vertical position and is attached to the cameras tripod thread. Additionally, a small ramp is printed and put beneath the center of the lens. It helps stabilizing it and reducing the weight that the camera holder has to withstand. The whole construction helps to stabilize the camera components and prevents unintended movement of the camera sensor. This could happen, for example, when the shutter triggers because it applies kinetic movement to the whole camera body.

The holder makes it easier to remove and remount the camera back in the same position. Nevertheless, the dimensions of the stage allow accessing the battery compartment of the camera for a quick and easy battery change.

4.1.3 Illumination part

The third part is responsible for the illumination of the sample. For capturing the reflections of the samples for different incoming light directions the system has multiple illumination sources. More precisely, there are sixteen different light sources where always two are mounted at each of the angles -50, -40, -30, -20, 15, 25, 35, 45 degrees to the camera sensor. Furthermore, the distance between the cylinders center axis and light sources is always 30 cm.

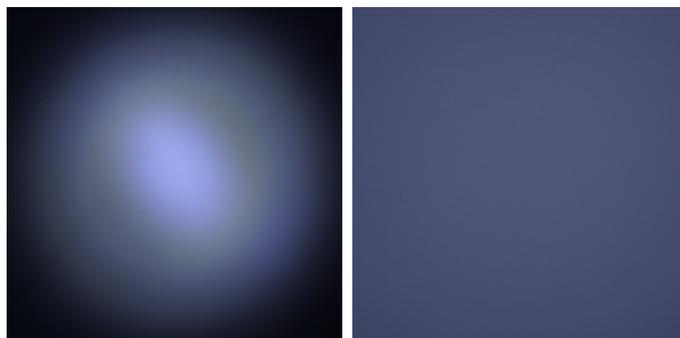


Figure 4.2: *LED pattern without (left) and with (right) the diffuser. Note that the diffuser reduces the emitted light.*

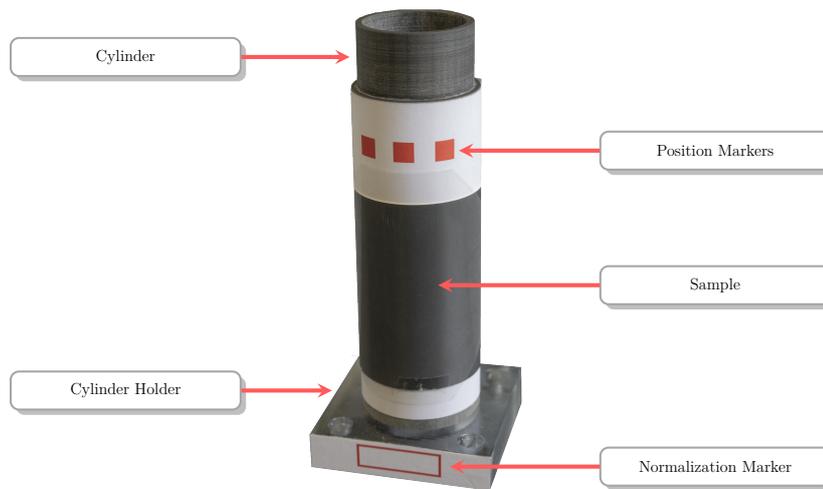


Figure 4.3: *Annotated image of the sample part.*

The light sources are LEDs. Since multiple of them are required and LEDs are in general cheap, they are well-suited for this purpose. A holder is printed with an FDM printer for each of the LED pairs. It has three holes at different heights where one LED can be plugged in. The center of the middle hole is at the same height as the camera sensor and the cylinder center. The other two holes are located 13.5 mm above and below. They provide the possibility to add more LEDs or change the position of the mounted ones to measure other parts of the BRDF space.

The setup used in this thesis has two LEDs attached to every holder. Always one in the middle hole while the other is alternating between the upper and lower hole. The arrangement ensures that the sample part is illuminated uniformly. This is important for a proper capturing of the calibration image which is explained in section 4.2.1. Furthermore, having LEDs at different heights for the same angle provides the possibility to measure a wider BRDF space.

Additionally, the holder has a small channel with a diameter of 0.75 mm in front of the LED. This reduces the area of the emitted light and makes it more like a point light source. The emitted light of LEDs is not uniform and rather forms a pattern. The pattern can produce errors if it is too significant. The mentioned channel already reduces the effect. It bundles the light and thus reduces the pattern, but does not obliterate it.

For removing the pattern, a diffusor is installed between the LED and the channel. The diffusor is an opaque circle-shaped Plexiglas of 3 mm thickness. Figure 4.2 shows the illumination of a diffuse material with and without it. The effect can be significantly reduced but is still visible. Chapter 4.3.3 provides a software solution for removing the remaining pattern.

The used LEDs have a color temperature of 10000 Kelvin and are manufactured by the company Thomsen. A mains adapter powers them with 12.5 Volt and 6 Ampere. Since all LEDs have to be turned on at the same time for capturing the calibration image, the mains adapter should have enough Ampere to power all LEDs. The used pre-resistor has 300 ohms. This results for the LED in a total voltage of 3.6 Volt at a forward current of 30 milliamperes. With this configuration, it has a luminous intensity of 80 candelas.

4.1.4 Sample part

This area includes the sample and the required markers for the geometric calibration and is depicted in figure 4.3. First, a cylinder holder is printed and mounted to the wooden plate. It has the purpose of making the handling of the cylinder more flexible. More precisely, the holder has a circular hole where the cylinder can be plugged in. The hole makes the cylinder removable and thus the changing of the sample easier. Simultaneously, the cylinder is still at the same position even if it is removed and plugged in again.

The cylinder itself is also printed for achieving a constant diameter and thus a higher precision. The sample is wrapped around it. This solution is more convenient than putting the sample on a flat surface. More precisely, due to the cylinder curvature measurements of multiple lamps do not generate redundant data. A more detailed explanation on this concept is given in chapter 7.1.3. The cylinder has a diameter of 50mm which is big enough to provide a proper resolution for the chosen camera distance. A sheet of matte black paper is first wrapped around the cylinder. It creates a uniform matte surface in case that the sample is transparent. The samples are later fixed on top of the black paper.

Additionally, a marker is placed on the front part of the holder. The marker consists of a white patch which is framed by a red square. This marker is used for light normalization which is further described in chapter 4.3.3. The front side of the holder is facing the camera. This ensures that the marker are captured adequately on every image.

Localization of the cylinder is needed to define proper geometric information for points on the cylinder surface. For this purpose, markers are used for the position detection of the cylinder. More specifically, three red colored squares are printed on a white paper. The markers are wrapped around the top part of the cylinder. This part of the cylinder must be faced to the camera to capture the markers. Here, it is not necessary that they perfectly face the camera in each image. They can be moved by rotating the cylinder as long as the camera still sees all of them. This range of rotation makes it easier to plug in the cylinder at the right position. A more detailed explanation of the localization system is given in chapter 4.3.2.

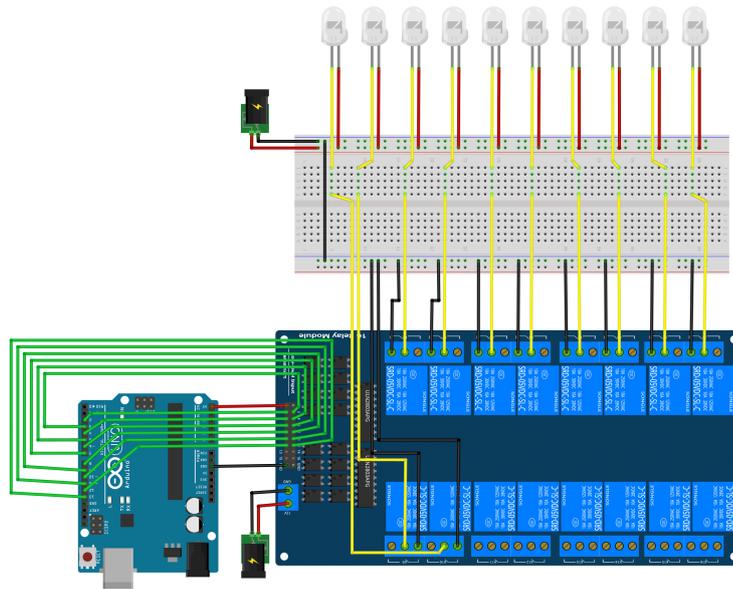


Figure 4.4: *The circuit diagram for Raspberry Pi controller, relay shield and 10 of the LEDs. For sake of simplicity the wiring of the other 6 LEDs is omitted as it works analogously.*

4.1.5 Assembling

All these parts are mounted together on a wooden board of the size 80 cm x 60 cm. It is important to ensure that all holders are in the correct position. Especially for LED and cylinder holders, the positions have to be exact. For this purpose, the holder shapes were laser-cut from a Plexiglas sheet of 3 mm thickness. This sheet is fixed on the wooden board with screws. With this construction, the different holders can be easily screwed at the correct positions without the need for any manual measuring.

The LEDs are connected to the relay via black cables with a diameter of 0.75 mm. The cables are fixed at several points with cable holders that are glued to the wooden board. The Plexiglas sheet and the cable holders were sprayed in matt black to keep the reflections as little as possible. The circuit diagram for assembling controller, relay module and LEDs is shown in figure 4.4.

Additionally, we built a box around the setup to reduce light from other sources than the LEDs such as sunlight or reflections from the wall. These would affect the measurements. The frame is made of aluminum sticks with a size of 100 cm x 100 cm x 100 cm. For preventing light from outside getting into the box, pieces of cloth were attached to each side. The cloth on one side can be pulled up to make the setup accessible even with the box on top of it.

4.2 Image Acquisition

Extracting the BRDF data requires multiple images taken with different parameters. One image is used for the geometric calibration. Furthermore, a sequence of images is used for the radiometric calibration. They are used for computing an HDR image which requires images at different exposures. In the following, the capturing procedure and details for both types are explained.

4.2.1 Calibration Image

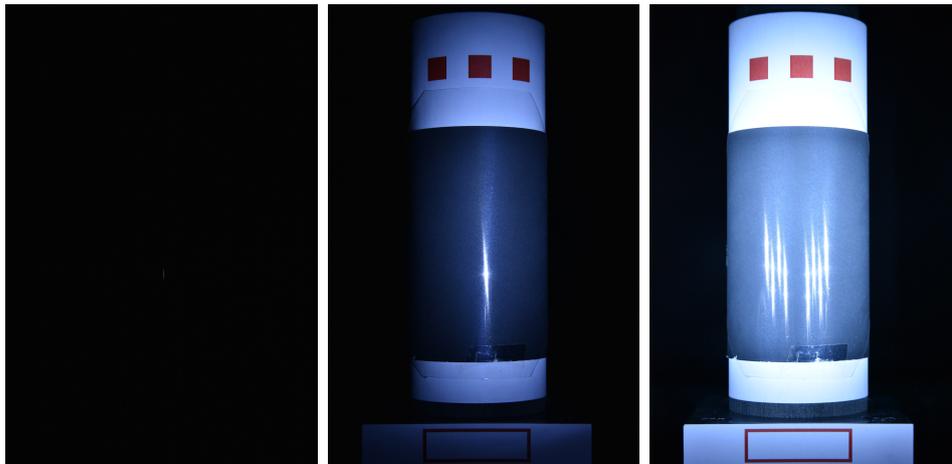
For the geometric calibration, the positions of the different markers in the image need to be known. They are used for computing the position of objects such as the cylinder. Unfortunately, it is not possible to use one of the LDR images from the sequences or the HDR image computed from them. That is because for sequence images there is always only one LED turned on. Depending on the currently used LED it may be possible that not all markers are fully visible. In that case, a detection would not be possible.

To make sure that all markers are well illuminated a separate image is taken under special conditions. More specifically, all LEDs are turned on during capturing. The image is used to compute the marker positions in the image. It is important that each of them and especially their edges are bright enough. Thus, one correctly illuminated LDR image suffices because at this point it is not important to capture a high dynamic range. The raw calibration image is denoted as $CI_{x,y}^{raw}$ where x and y are the 2D pixel indices. An example is shown in figure 4.5c.

The camera parameters for the calibration image are depicted in table 4.1. It is important to mention that these parameters are producing usable images for this particular hardware setup. They have to be adjusted if either other light sources are used or the distance between the camera and cylinder is changed. First, the aperture is chosen such that the depth of field is big enough to guarantee all markers to be in focus. The ISO is relatively high to produce a bright image. Noisy pixels do not affect the detection since the image is blurred during detection. Last, the exposure time is adjusted to produce a bright image. Using other combinations of ISO and exposure time is possible. These values are a good trade-off for this particular setup.

Image Type	#	Aperture	ISO	Exposure Time (in s)
Calibration	1	11.0	3200	30
Sequence	1 ... 18	11.0	320	1/4000, 1/2000, 1/1000, 1/500 1/250, 1/125, 1/60, 1/30, 1/15 1/8, 1/4, 1/2, 1, 2, 4, 8, 15, 30

Table 4.1: Camera parameters for captures.



(a) *Sequence Image 1* (b) *Sequence Image 18* (c) *Calibration Image*

Figure 4.5: *Example captures for a specular material. Note that the images are rotated to the right for a better visualization.*

4.2.2 Sequence

Specular samples are hard to capture. Let's illustrate this with the example of a mirror-like material. If the angle of reflection is similar to the angle of incidence, this material reflects most of the incoming light. It leads to overexposed parts in the image for specular materials since bright LEDs are used as light sources. For properly capturing these parts illuminated, a minimal exposure time should be used.

Even though the peak reflection is captured properly, the material reflects lesser light for the case that the outgoing ray deviates from the perfect reflection. These parts are not captured by such a short exposure time. Additionally, it can happen that even the peak reflection is not bright enough to get captured at this short exposure time. Especially for diffuse materials, longer exposure time is needed because less light is reflected even at the peak reflection. As we see, it is hard to find an exposure time that fits all materials. Furthermore, it is even impossible to choose one that fits all parts of one material. Here the computation of an HDR image is essential.

For computing, the HDR values, 18 LDR images are captured at different exposures. The first exposure is the camera's shortest possible exposure. It ensures that the peak of highly specular materials is not overexposed. The exposure time is roughly doubled for every subsequent capture which makes sure not to underexpose the peak for less specular materials. If the difference in exposures is too high, the peak might be underexposed in the first one while it is already overexposed in the next one. The longest exposure time is the camera's longest possible exposure. It guarantees that dark areas are still captured properly.

Two example sequence images of a highly specular material are shown in figure 4.5a and 4.5b. The peak in the left image is properly exposed while the outer area around it is underexposed. However, the right image provides more of the missing information. The same applies to all captured images between those two exposures. They contribute valuable information that is necessary to compute the HDR image. Additionally, figure 4.5b shows that not even the brightest sequence image is bright enough to extract the marker data. It emphasizes the need for the calibration image.

The camera parameters for the LDR sequence are depicted in table 4.1. Here, again these parameters have to be adjusted when using other light sources or different distances. Since the image captures a cylindrical surface, a certain depth of field is needed. The small aperture ensures that the whole visible surface of the cylinder is in focus. A larger one could blur parts of the cylinder while a smaller one can lead to an underexposed peak. The shortest exposure time would not be short enough anymore in that case.

The smallest native ISO of the camera is 160. It is capable of using ISO 100, but due to the involved interpolation, it can produce more noise than the native ISO. Even with this ISO value, it is possible that the sensor captures noise because long exposure times are used. Since it was more important to have a high aperture, the ISO had to be reduced to 320 to get well-illuminated images. Unfortunately, it further increases the noise which can lead to errors since the HDR computation is performed pixel-wise. To account for this problem multiple consecutive sequences of the same scene are taken and combined later. The number of sequences and exposure times are denoted as $m_{\#}$ and $n_{\#}$ respectively. The raw LDR images are defined as $SI_{m,n,x,y}^{raw}$ where $m \in \{0, \dots, m_{\#} - 1\}$ is the sequence index and $n \in \{0, \dots, n_{\#} - 1\}$ the exposure time index. x and y denote the 2D pixel index within the image.

4.2.3 Full Process

The capturing process runs fully automated. More specifically, after specifying the capturing parameters, the Software takes all needed images for acquiring the BRDF. The whole process for one sample works as follows. First, all LEDs are turned on to take the calibration image. For the sequence capturing, it is possible to define which LEDs should be used for capturing. In this thesis, all LEDs at the center row of the LED holders were used. Thus, the sequences are taken for eight single LEDs. Furthermore, three sequences were captured per LED. Each of these LEDs is turned on once while the LDR sequences are taken. The capturing time for one LDR sequence is on average 4 minutes. It results in a total capturing time of 120 minutes (8 LEDs x 3 sequences x 4 minutes) per material. Overall 540 images (8 LEDs x 3 sequences x 18 images) are taken.

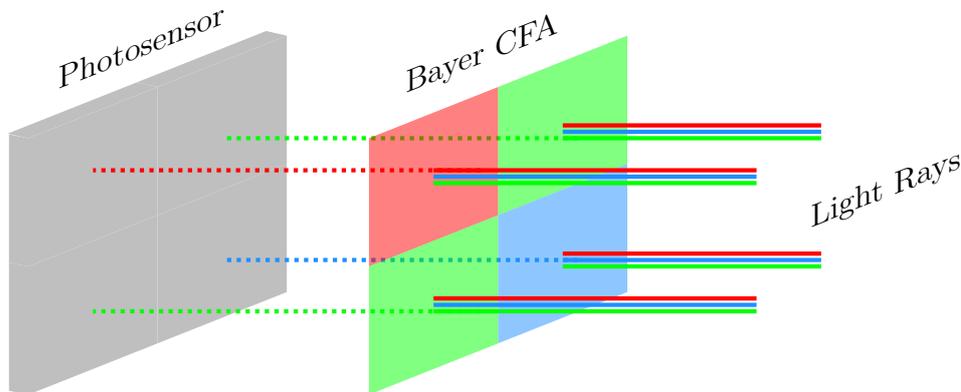


Figure 4.6: Concept of a color filter array (CFA). The CFA consists of multiple filters that are put in front of the photosensors pixels. Each lets only pass a specific wavelength and makes it possible to measure different colors with each pixel. The depicted arrangement of the CFA is called a Bayer CFA.

4.3 Data Acquisition

For fitting a proper BRDF, multiple pairs of input and output values of the function are required. More precisely, we need for a given input of θ^{in} , θ^{out} , ϕ^{in} , ϕ^{out} the corresponding reflected amount of light. To get these measurements, we take the pixels of the captured images that belong to the material surface. Here, each pixel corresponds to one point of the BRDF space.

To get these measurements, we have to find out which pixels lie on the sample surface and which input values of the BRDF they belong to. This step is called the geometric calibration. Additionally, the corresponding reflected amount of light has to be computed from the captured pixel value. This is done in the radiometric calibration. Additionally, we need to preprocess the RAW images of the camera to make them usable for the data extraction. The image processing and both types of calibrations are explained in the following. Additionally, it is explained how the BRDF data is composed.

4.3.1 Image Preprocessing

The camera stores raw sensor data. Several preprocessing steps are required to get a viewable output image. Keep in mind that there are cameras that apply a non-linear transformation to compress the raw data. If this is the case, the data has to be uncompressed first before the next steps. Luckily, in our case, it is not necessary.

The camera's photosensor detects light intensities. It cannot specify any color or wavelength. Thus, a color filter is applied on top of the photosensor. It lets pass only a specific wavelength and filters any others. As a consequence, each pixel is sensitive to only one color specified by its filter. The mosaic-like structure is called a color filter array (CFA). Nikon uses the most common CFA which is the so-called Bayer filter. It consists of filters for the colors red, green and blue. They are arranged in a repetitive pattern of 2x2 of the sequence red, green, green, blue. It contains twice as many green filters since the human visual system is more sensitive to green colors. The concept of a CFA is visualized in figure 4.6. The resulting captured array by the sensor is called a CFA image.

The sensor output has 14 bit per sensor pixel while it is stored as 16-bit values in the CFA image. These values are linearized to get rid of offset and scaling. We perform this step for all captured images, i.e., calibration and sequence images. Here, the darkest value does not start at 0. It starts at a predefined position and can vary for each filter. These so-called black levels are stored in the metadata of the raw image file. The brightest value is also located at a specific value. This position is constant for all filters and stored as the saturation level in the metadata. Still, it is possible that values below the black level or above the saturation level occur in the CFA image due to sensor noise. These values have to be clipped since they are not meaningful. In the following, we denote the 2D arrays storing black levels and saturation levels for each pixel as LI^b and LI^s respectively. SI^{lin} and CI^{lin} are the linearized raw images and computed as following

$$CI_{x,y}^{lin} = \min(\max(\frac{CI_{x,y}^{raw} - LI_{x,y}^b}{LI_{x,y}^s - LI_{x,y}^b}, 1.0), 0.0) \quad (4.1)$$

$$SI_{n,m,x,y}^{lin} = \min(\max(\frac{SI_{n,m,x,y}^{raw} - LI_{x,y}^b}{LI_{x,y}^s - LI_{x,y}^b}, 1.0), 0.0) \quad (4.2)$$

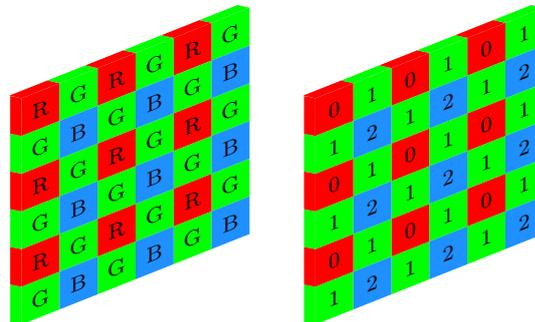


Figure 4.7: The figure shows the color filter array of the raw image CI^{raw} (left) and the corresponding color indices ID^{cid} (right).

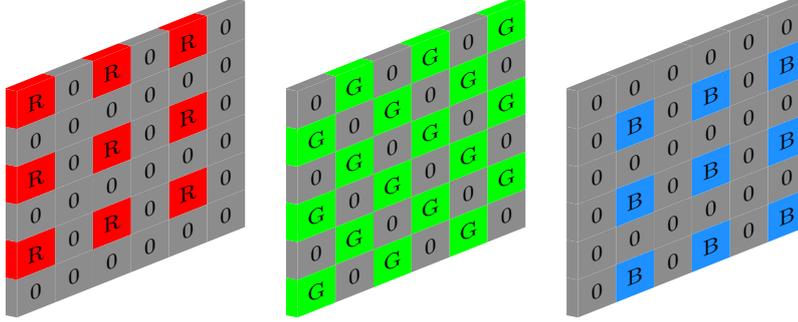


Figure 4.8: The figure depicts the slices of $CI_{x,y,c}^{col}$ for its third dimension. More precisely, it shows $CI_{x,y,0}^{col}$ (left), $CI_{x,y,1}^{col}$ (middle) and $CI_{x,y,2}^{col}$ (right).

4.3.2 Geometric Calibration

The geometric calibration is responsible for computing the relative 3D positions of the camera, light source, and sample. We use these positions to find out which pixels lie on the sample surface. Additionally, it helps to define the corresponding angles for the different points. The whole geometric calibration procedure uses the calibration image.

First of all, we need to define a proper reference coordinate system. All other positions are relative to it. It has to be a coordinate system whose position can be estimated in each calibration image. Additionally, it should move as little as possible relative to the light source. This is important since the light source position is computed once and should stay fixed in the reference space. If the reference system moves even slightly the light source position is not correct anymore.

We choose the cylinder holder space as a reference coordinate system. It is visible in all calibration images and fixed to the wooden board. Alternatively, we could have used the camera space. This would result in a less robust system since the camera can move and simultaneously the reference of all other points. Even if it is fixed by a holder, it has mechanical parts like the shutter which is moving during the capturing process.

Demosaicing For the calibration image, there is only one single capture. This image is responsible for extracting positions of markers which are used to compute different extrinsic parameters. Thus, it is necessary to demosaic the image in order to get an ordinary RGB image from the CFA. This step is only performed for the calibration image. We do not need to demosaic the sequence images since their CFA is used as it is for further computations.

The naive approach would be to always combine groups of four pixels to one. With this method, each group contains one red, two green and one blue measurement which can be combined to an RGB pixel. The problem is that it shrinks the image by a factor of 2 in each dimension because of the grouping. For preserving the image resolution, we use interpolation to get RGB values for each pixel. One color is already given for each pixel. The missing two colors are a bilinear interpolation of the adjacent pixels.

The procedure works as follows. First, the CFA image is split into three different 2D arrays where each array contains the measured color information for one of the three RGB colors. We need to know which color was measured for a specific pixel. For this purpose, we define a 2D array as ID^{cid} . It contains for each pixel position the index of its measured color. Note that it has the same size as CI^{lin} . The CFA pattern has the size 2 x 2 and thus four different indices. But since the color green is represented twice, its indices can be combined. In the end, there are three color indices c which are 0 for red, 1 for green and 2 for blue (see figure 4.7).

$$CI_{x,y,c}^{col} = \begin{cases} CI_{x,y}^{lin} & \text{if } c = ID_{x,y}^{cid} \text{ with } c \in \{0, 1, 2\} \\ 0 & \text{otherwise} \end{cases} \quad (4.3)$$

$CI_{x,y,c}^{col}$ is a 3D array that contains for each pixel its captured value at color index c . All other positions with other color indices are set to zero. For example, we take the particular pixel at the index (4,4). The pixel has the color index 1 and a measured value of 0.42. Thus, $CI_{4,3,1}^{col}$ contains the value 0.42 while $CI_{4,3,0}^{col}$ and $CI_{4,3,2}^{col}$ have value zero since their color was never measured. The resulting arrays are illustrated in figure 4.8.

Still, the values for positions where a specific color was not captured are missing. We compute the missing RGB information by applying a bilinear interpolation to each of CI^{col} . More specifically, the interpolation is implemented as a 2D convolution. The convolution and its corresponding kernels are defined as follows.

$$CI_{x,y,c}^{rgb} = K_c * CI_{x,y,c}^{col} \text{ with } c \in \{0, 1, 2\} \quad (4.4)$$

$$K_0 = K_2 = \frac{1}{4} \cdot \begin{bmatrix} 1 & 2 & 1 \\ 2 & 4 & 2 \\ 1 & 2 & 1 \end{bmatrix} \quad K_1 = \frac{1}{4} \cdot \begin{bmatrix} 0 & 1 & 0 \\ 1 & 4 & 1 \\ 0 & 1 & 0 \end{bmatrix} \quad (4.5)$$

where K_c is the kernel for the specific color index. The stride is set 1, and the borders are extended by a zero padding of 1 to preserve the image size. The convolution results in an RGB image that defines a color value at each pixel. The complete pipeline for processing the calibration image is shown in figure 4.9.

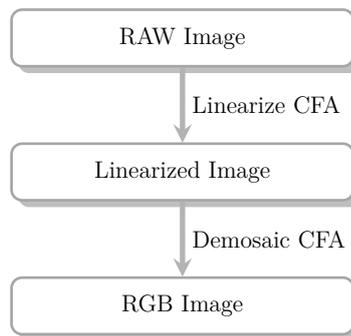


Figure 4.9: *The figure shows the image processing pipeline for calibration images.*

Intrinsic Parameters The intrinsic parameters are necessary to know how the camera maps points from its own space to the image plane. These parameters are different for every camera-lens combination. They have to be recomputed if one of both changes.

The parameters are computed using multiple images of a checkerboard. It works by using the coordinates of the crossings on the checkerboard. Their 2D coordinates can be detected in the images. Furthermore, the 3D coordinates are also known as the checkerboard structure is also known. We use these 2D-3D correspondences to estimate how the camera maps them which actually describes the intrinsic parameters. A more detailed explanation for the computation of the intrinsic parameters is given on the OpenCV documentation².

We took 30 images of a checkerboard pattern that are passed to the algorithm. It is important that all images are different in terms of distance and viewing direction to the checkerboard. Keeping the distance fixed resulted in a higher remapping error. The checkerboard pattern is printed on a standard sheet. It is helpful to fix the sheet on an underlay of uniform thickness to be sure that it stays planar. The size of the checkerboard on the print can be arbitrary. This is because the calibration software uses the edge size of one square for the estimation which can be measured manually.

Cylinder Holder Extrinsic We need to know the cylinder holder position before computing any of the others. With the intrinsic parameters, we can simulate how the camera maps world 3D points to the 2D image plane. In addition, corresponding pairs of 2D points in the image domain and 3D points in the local space of the cylinder holder are necessary. It is possible to approximate the 3D coordinates of the cylinder holder using these correspondences and the mapping process.

²https://docs.opencv.org/2.4/modules/calib3d/doc/camera_calibration_and_3d_reconstruction.html

First, we need to find the coordinates of the 2D point on the cylinder holder in the image domain. As the points have to be visible in all images, we take points from the marker which is located at the front side of the cylinder holder. More specifically, we take the corners of the marker since their position is known in the holder space.

Unfortunately, the markers on the cylinder are red as well. They have to be distinguished from the normalization marker. This is done by defining an area called the region of interest (RoI). It is a rectangular area in the image which is used to isolate parts from each other. The RoI is defined by the two corner points of the rectangle. They are denoted as P^{rul} for the upper left and P^{rlr} for the lower right points of the RoI.

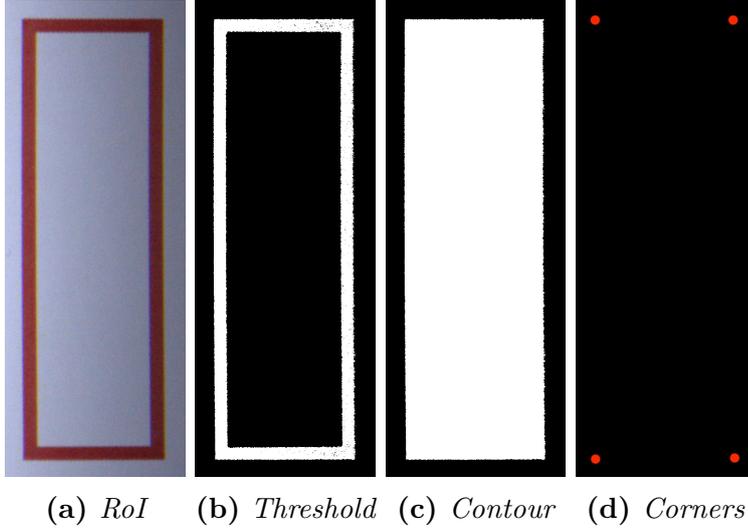
$$CI_{x,y,c}^{roi} = \begin{cases} CI_{x,y,c}^{rgb} & \text{if } P^{rul} \leq (x, y) \leq P^{rlr} \\ 0 & \text{otherwise} \end{cases} \quad (4.6)$$

The isolation of the RoI is denoted in equation 4.6 where $CI_{x,y,c}^{roi}$ is the image that contains only the RoI. It is depicted in figure 4.10a. For finding the exact pixels that belong to the red area of the marker a threshold is applied to $CI_{x,y,c}^{roi}$. We perform the thresholding in the HSV color domain. The reason to use HSV over RGB is that it allows a more precise definition of the thresholds. The HSV image is denoted as $CI_{x,y,c}^{hsv}$ where c stands for the different channels h, s and v.

The thresholds are defined as t_c^l and t_c^u for the lower and upper limit respectively. C denotes the channel, i.e., h, s and v. Applying the threshold results in a binary image $CI_{x,y}^{thr}$ that indicates if a pixel lies within the threshold limits. More specifically, each red pixel of the marker has a value of 1 while all other pixels are 0.

$$CI_{x,y}^{thr} = \begin{cases} 1 & \text{if } \forall c \in \{0, 1, 2\} : t_c^l \leq CI_{x,y,c}^{hsv} \leq t_c^u \\ 0 & \text{otherwise} \end{cases} \quad (4.7)$$

The resulting image is depicted in figure 4.10b. It is an outline of the marker area and thus not yet useful for corner detection. It could happen that the corner detection finds corners inside the marker even though we are interested in the outside corners. To make the procedure more robust, we want to have a rectangular-like shape that is completely filled. This is done by first finding the marker contour and then filling it.



(a) *ROI* (b) *Threshold* (c) *Contour* (d) *Corners*

Figure 4.10: Image of the different steps for the corner point extraction. The corner points in (d) are overemphasized for better visibility. Originally, they are just one pixel thick.

We compute the contours in the binary image CI^{thr} according to Suzuki et al[18]. It provides multiple contours that can occur within the marker or even in other parts of the ROI due to erroneous threshold application. We argue that the biggest shape in the ROI is the marker while all erroneously detected shapes are in comparison smaller. Thus, the contour that belongs to the full marker is the contour that covers the biggest area. We define C_i as an array with all detected contours and C_i^a as the corresponding contour areas. Here, $i \in \{0, \dots, c_{\#}\}$ is the contour index where $c_{\#}$ is the number of detected contours. The marker contour C^m is computed as follows

$$S_1 = \{0, \dots, c_{\#} - 1\} \quad (4.8)$$

$$C^m = \{C_i \mid i \in S_1 \wedge \forall j \in S_1 : C_j^a \leq C_i^a\} \quad (4.9)$$

Using C^m the contours of the marker can be drawn and filled up. This results in a binary image $CI_{x,y}^{cont}$ that contains the clean shape of the marker. It is shown in figure 4.10c and computed as following

$$CI_{x,y}^{cont} = \begin{cases} 1 & \text{if } (x, y) \text{ inside } C^m \\ 0 & \text{otherwise} \end{cases} \quad (4.10)$$

In the next step, we use the Harris corner detector[19] to find potential candidate corner points in the image $CI_{x,y}^{cont}$. The Harris corner detector provides multiple corner points $P_{i,j}^{hc}$ where $i \in \{0, \dots, hc_{\#} - 1\}$ denotes the point index, $j \in \{0, 1\}$ the dimension and $hc_{\#}$ the number of detected Harris corner points. The four corner points of the marker are the ones that have the largest distance to each other.

$$D = (a_{ij}) \in \mathbb{R}^{hp_{\#} \times hp_{\#}} \quad (4.11)$$

$$a_{ij} = \|P_i^{hp} - P_j^{hp}\|_2^2 \quad (4.12)$$

For separating the four desired points from the others a distance matrix D is constructed. It contains for each pair of points in P^{hp} the euclidian distance to each other. The first two corner points are the ones who have the highest distance, thus points with the indices where D has its maximum. Those two points are located in two opposite corners. Their indices are denoted as ID_1^{cp} and ID_2^{cp} . We compute them as following

$$S_2 = \{0, \dots, hp_{\#} - 1\} \quad (4.13)$$

$$ID_0^{cp}, ID_1^{cp} = \{i, j \in S_2 \mid \forall k, l \in S_2 : D_{kl} \leq D_{ij}\} \quad (4.14)$$

The third point is found by taking a point whose distance to both first ones is the largest. It is computed as

$$ID_2^{cp} = \{i \in S_2 \mid \forall j \in S_2 : \sum_{k=0}^1 D_{ID_k^{cp}, j} \leq \sum_{k=0}^1 D_{ID_k^{cp}, i}\} \quad (4.15)$$

A similar procedure applies to find the fourth point. Here, the ID of the point which has the largest distance to the first three points is wanted. It is computed with the following formula

$$ID_3^{cp} = \{i \in S_2 \mid \forall j \in S_2 : \sum_{k=0}^2 D_{ID_k^{cp}, j} \leq \sum_{k=0}^2 D_{ID_k^{cp}, i}\} \quad (4.16)$$

We use the indices to gather the corner points from the list of Harris points. The final corner points are defined as

$$P_i^{cp} = P_{ID_i^{cp}}^{hp} \text{ with } i \in \{0, \dots, 3\} \quad (4.17)$$

Despite the point coordinates are known they are not ordered yet. The different points have to be assigned to their respective corners. The camera is rotated clockwise by 90 degrees. This means that the bottom side of the cylinder is on the right side in the image. That is also where the cylinder holder is located. With decreasing pixel index in the x-direction, we move upwards along the cylinder. Thus, the two points with the

highest x-coordinates are the two lower points of the marker. The two extra points with lower x-coordinates are the two top points. It works similarly for the left or right points. The two points on the left side have higher y-coordinates while the ones on the right side have lower y-coordinates. This information can be used to sort the points. We define the points P_1^{2D} , P_2^{2D} , P_3^{2D} , P_4^{2D} as the upper left, upper right, lower left, and lower right points of the marker.

In addition to the 2D coordinates of the corners, we need the corresponding 3D points in the local coordinate system of the cylinder holder. The width and height of the normalization marker are measured and denoted as w^{nm} and h^{nm} . We also need the depth of the holder to know the distance between the marker plane and the center of the holder. It is also the center of the cylinder since it is mounted to the holder center. This distance d^{nm} is half of the holder depth. Thus, the 3D coordinates are defined as follows.

$$\begin{aligned}
 P_1^{3D} &= \begin{pmatrix} -w^{nm}/2 \\ h^{nm}/2 \\ -d^{nm} \end{pmatrix} \\
 P_2^{3D} &= \begin{pmatrix} w^{nm}/2 \\ h^{nm}/2 \\ -d^{nm} \end{pmatrix} \\
 P_3^{3D} &= \begin{pmatrix} -w^{nm}/2 \\ -h^{nm}/2 \\ -d^{nm} \end{pmatrix} \\
 P_4^{3D} &= \begin{pmatrix} w^{nm}/2 \\ -h^{nm}/2 \\ -d^{nm} \end{pmatrix}
 \end{aligned} \tag{4.18}$$

With the corresponding points P^{3D} and P^{2D} we can estimate the position of the cylinder holder. This is done by defining a Perspective-n-Point (PNP) problem. We solve it with the algorithm proposed by Gao et al. [20]. It optimizes for the position by minimizing the reprojection error from 3D to 2D points. The intrinsic parameters are required for this operation since they define a major part of the projection. The procedure estimates a translation vector and rotation matrix which are denoted as TV_{hold}^{cam} and RM_{hold}^{cam} respectively.

Since we want the cylinder holder to be the reference space, we translate the position to the cylinder holder space as follows.

$$\begin{aligned} TV_{hold}^{hold} &= TV_{hold}^{cam} - TV_{hold}^{cam} = \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix} \\ RM_{hold}^{hold} &= (RM_{hold}^{cam})^{-1} \cdot RM_{hold}^{cam} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} \end{aligned} \quad (4.19)$$

Camera Extrinsic The camera position in its own space is located in the origin. Furthermore, the rotation matrix is the identity matrix. Both are defined as following

$$\begin{aligned} TV_{cam}^{cam} &= \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix} \\ RM_{cam}^{cam} &= \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} \end{aligned} \quad (4.20)$$

They are translated to the cylinder holder space by using the previously computed TV_{hold}^{cam} and RM_{hold}^{cam} and is performed as follows

$$\begin{aligned} TV_{cam}^{hold} &= TV_{cam}^{cam} - TV_{hold}^{cam} \\ RM_{cam}^{hold} &= (RM_{hold}^{cam})^{-1} \cdot RM_{cam}^{cam} \end{aligned} \quad (4.21)$$

Cylinder Extrinsic Each pixel that pictures the sample on the cylinder corresponds to exactly one BRDF value. Thus, the position of these points needs to be known in the reference coordinate system. To compute these positions, we need to know the position of the cylinder object. The method uses three square-sized colored markers. These markers are located at the top of the cylinder.

The computation of the cylinder extrinsic works similar to those for the cylinder holder. Thus, I briefly explain the main steps and give more detailed information for steps that differ from the other procedure. The first step is again to compute the 2D corner points of the markers. To find the markers we use an RoI defining their area in the image. The contours are computed by applying a threshold to the RoI image. Since there are three markers, we also take the three contours with the biggest areas instead of one. The corner points are computed for each contour individually. They are sorted as before with the difference that now we end up with two rows where each contains six points. They are ordered from left to right on the cylinder or bottom to top in the image space.

For the computation of the 3D points, we assume that the cylinder is centered in its local coordinate system. This means that its center axis is simultaneously the y-axis. It is important to keep in mind that their distance is extended by the thickness of the paper depending on the used paper to print the markers on. We define r^c as the cylinder radius including the paper thickness. The marker corners are always at the same distance from the center axis since they lie on the cylinder hull. Thus, x and z coordinates are the same for both rows. Their position can be computed by rotating the vector that points from the cylinder center to the cylinder surface. In fact, rotating this particular vector around the y-axis moves it along the cylinder hull. Here, we use the assumptions that each marker has the same width, i.e., w^{cm} . The distance between the two markers is also w^{cm} . Thus, the x-coordinates of the 3D points are computed as following

$$\cup_{i \in \{1, \dots, 6\}} [\sin(\frac{w^{cm}}{r^c} \cdot i) \cdot r^c] \quad (4.22)$$

Similarly, the z-coordinates are computed as

$$\cup_{i \in \{1, \dots, 6\}} [\cos(\frac{w^{cm}}{r^c} \cdot i) \cdot r^c] \quad (4.23)$$

The height of the lower row of corners is the distance between the sample center on the cylinder and the lower side of the markers. The height of the upper row is the distance to the upper side of the markers. They are defined as h^u and h^l respectively. These distances can be measured by hand as they do not need to be perfectly accurate. A small offset does not affect the 2D-3D correspondences of the corner points. If the heights are longer, for example, the center of the cylinder is estimated lower by the same amount. Nevertheless, it is important that the difference between h^u and h^l exactly matches the height of a marker, i.e., h^{cm} . Otherwise, the relationship between upper and lower corners is not preserved anymore such that an estimation of extrinsics is not possible. In the end, the y coordinates of the 3D points are h^u for the first six points and h^l for the last six ones. They are defined as following

$$[\cup_{i \in \{1, \dots, 6\}} h^u] \cup [\cup_{i \in \{7, \dots, 12\}} h^l] \quad (4.24)$$

The computation of the extrinsic parameters works as described in the previous chapter. Again it is defined and solved as a PNP problem. The resulting extrinsics of the cylinder in camera space are defined as TV_{cyl}^{cam} and RM_{cyl}^{cam} . In order to bring the extrinsics to the cylinder holder space, we apply the following translation

$$\begin{aligned} TV_{cyl}^{hold} &= TV_{cyl}^{cam} - TV_{hold}^{cam} \\ RM_{cyl}^{hold} &= (RM_{hold}^{cam})^{-1} \cdot RM_{cyl}^{cam} \end{aligned} \quad (4.25)$$

2D-3D Correspondences In this step, we want to find for each pixel that pictures the sample, the corresponding world coordinate. This is necessary to put the measured sensor response in relation to the geometric values. With the aid of the camera intrinsics we are able to map any visible 3D point from the camera space to the image plane. Thus, it would be possible to sample the cylinder surface and check to which pixel it maps. The problem with this approach is that multiple points might map to the same pixel. Additionally, the computational overhead is huge since most of the points would not map to any pixel in the image plane.

It is a more natural approach to go the other direction and compute the corresponding surface point for each pixel. Of course, some pixel are also not mapped to any surface point, but the overhead is much lower. Unfortunately, the images captured by the camera do not contain any depth information. They are lost during the mapping process from 3D world coordinates to image plane. We can just construct a ray for each pixel. By intersecting these with the cylinder surface, we get the missing depth information. This is possible since we know the cylinder position in camera space. For the computation, a ray-tracing approach is used which I explain in the following.

We construct the rays in the camera space and then translate them to the cylinder space. The origin of all rays is the camera position. In general, the ray direction is given by the vector between camera position the pixel center. Since the camera position is the zero vector in its own space, we can directly take the pixel center as ray direction. Initially, the pixel positions are defined in the image plane space as

$$P_{i,d}^{ip} = \{(x, y) \mid x \in \{0, \dots, w^{img}\} \wedge y \in \{0, \dots, h^{img}\}\} \quad (4.26)$$

where i is the point index and d the dimension. In fact, P^{ip} is a list of all possible pixel coordinates in the image. They have to be translated to the 3D space of the camera by using the focal lengths f_x, f_y and principal points c_x, c_y as follows.

$$\begin{aligned} P_{i,0}^{cs} &= P_{i,0}^{ip} \cdot \left(\frac{P_{i,0}^{ip}}{f_x} - \frac{c_x}{f_x} \right) \\ P_{i,1}^{cs} &= P_{i,1}^{ip} \cdot \left(\frac{P_{i,1}^{ip}}{f_y} - \frac{c_y}{f_y} \right) \\ P_{i,2}^{cs} &= 1 \end{aligned} \quad (4.27)$$

Furthermore, the x and y coordinates are scaled by the z coordinate. In fact, the z coordinate expresses for each point the distance between the object it depicts and the sensor. As we do not know this distance yet, we choose it to be 1 for all points.

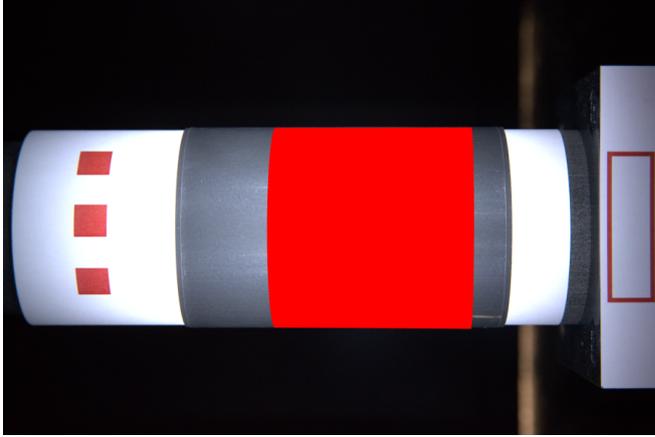


Figure 4.11: Result of raytracing. All pixels that have intersection points are marked red.

Note that it is necessary to scale x and y proportional to z. This is because each ray that is not parallel to the optical axis diverges from it. Thus, as a point moves along its ray, it simultaneously moves away from the optical axis. The divergence is expressed by the scaling.

Unfortunately, the pixel positions are distorted. Especially, for points that are far away from the center their position is off in the image. We need to perform an undistortion using the distortion coefficients which were computed during the camera calibration.

$$\begin{aligned}
 P_{i,0}^{ud} &= P_{i,0}^{cs} + (1 + k_1 r^2 + k_2 r^4 + k_3 r^6) + 2p_1 P_{i,0}^{cs} P_{i,1}^{cs} + p_2 (r^2 + 2(P_{i,0}^{cs})^2) \\
 P_{i,1}^{ud} &= P_{i,1}^{cs} + (1 + k_1 r^2 + k_2 r^4 + k_3 r^6) + 2p_2 P_{i,0}^{cs} P_{i,1}^{cs} + p_1 (r^2 + 2(P_{i,1}^{cs})^2) \\
 &\text{with } r = \sqrt{(P_{i,0}^{cs})^2 + (P_{i,1}^{cs})^2}
 \end{aligned} \tag{4.28}$$

where k_1, k_2, k_3 are the radial distortion coefficients and p_1, p_2 are the tangential distortion coefficients of the lens. The ray origin and directions are still in the camera space. They are translated to the cylinder space by first subtracting camera translation vector. Additionally, we have to apply the inverse of the camera rotation matrix.

$$\begin{aligned}
 P^{ro} &= (RM_{cyl}^{cam})^{-1} \cdot (TV_{cam}^{cam} - TV_{cyl}^{cam}) \\
 P_i^{rd} &= (RM_{cyl}^{cam})^{-1} \cdot (P_i^{ud} - Trans_{cyl}^{cam})
 \end{aligned} \tag{4.29}$$

The ray origin is the same for all pixels and denoted as P^{ro} . The ray directions of each pixel are defined as P_i^{rd} . They are intersected with the cylinder using the ray-cylinder intersection algorithm proposed by Pharr et al. [21]. Additionally, we are not intersecting with an infinite cylinder. Since any point outside the sample must not be considered, we have to limit the height of the cylinder we are intersecting with.

More specifically, we constrain the value for the y-axis of the intersection points to be within the boundaries $(-h^s/2, h^s/2)$. h^s denotes the sample height. The resulting intersection points that fulfill the height constraint are defined by P_i^{ip} . Note that the list is smaller than P^{rd} . The intersection points are still in the cylinder space. We need to transform it back to the camera space and then to the cylinder holder space.

$$P_i^{sp} = Rot_{cam}^{hold} \cdot (RM_{cyl}^{cam} \cdot P_i^{ip} + TV_{cyl}^{cam}) + Trans_{cam}^{hold} \quad (4.30)$$

P_i^{sp} are the 3D coordinates of the intersection points in the cylinder holder space. Figure 4.11 shows an example for the result of the concept. All pixels that have intersection points are marked red.

4.3.3 Radiometric Calibration

Besides the geometric information of the data points, we need to know the corresponding reflected amount of light for each of them. The radiometric calibration is responsible for the processing of the measured sensor response.

Sequence combination It was already mentioned in 4.2.2 that we take multiple sequences of the same scene to account for sensor noise. To combine the sequences, we average images with identical exposure times as follows.

$$SI_{n,x,y}^{comb} = \frac{1}{m_{\#}} \sum_{i=0}^{m_{\#}-1} SI_{m,n,x,y}^{lin} \quad (4.31)$$

Neighbor correction The used LED light sources have both high and low-frequency intensity fluctuations. To demonstrate this effect I took images of the light at different points in time. For each image, I extracted the same patch and averaged the intensity within the patch for each color individually. The resulting time-intensity plot is shown in figure 4.12. Here, we can see that intensity changes locally rapidly. These are the high-frequency fluctuations. Additionally, the intensity changes slowly over a longer period of time. These are the low-frequency fluctuations.

The high-frequency fluctuations are producing errors in the images within images of a sequence. The illumination between two neighboring images changes fast enough to affect the expected energy ratio of these images. More specifically, sequence images should be taken under identical illumination conditions for computing proper HDR values. Due to the high-frequency fluctuations the conditions are differently. Thus, the fluctuations have to be removed.

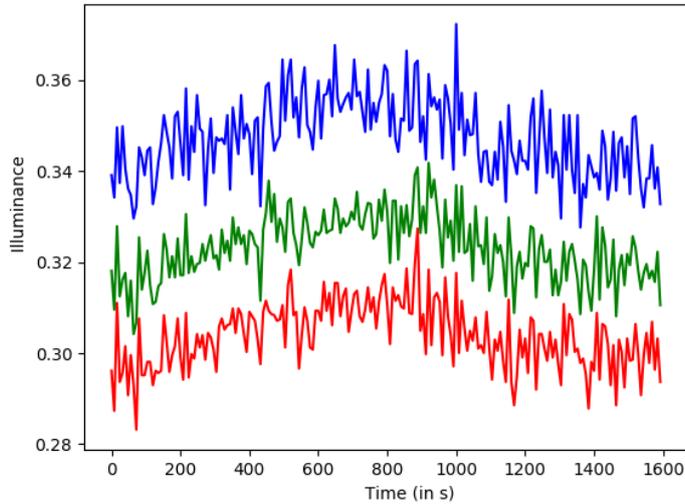


Figure 4.12: *Demonstrates low and high frequency fluctuations of the LED light source for the different color channels. The illuminance was measured by averaging the pixel intensities for the same patch in each image.*

Before we can compute the HDR image, it is necessary to rectify errors between the sequence images. This is done with the aid of the assumption that doubling the exposure time simultaneously doubles the response of the sensor. More specifically, if we take pairs of two neighbored exposure times, the corresponding sensor response should have the same ratio as the two exposure times. We compensate for deviations from the expected ratio and the actual ratio between each neighbored sequence images. Note that the following correction does not affect the error produced by the low-frequency fluctuations of the light source.

First, we construct pairs of indices for neighboring sequence images. These are corrected against each other. During the correction, we do not use pixels that are over- or underexposed. Thus, images with very long or short exposure times have less usable pixels. The most reliable correction can be performed for the images in the center of the sequence images array. If we began the correction from one end of the array, we would start with a high error and propagate it through the whole correction progress. To make the procedure more robust, we start in the center of the sequence array and correct towards each direction.

$$\begin{aligned}
 ID^{nb} = & [(i, i + 1) \mid i \in \{\lfloor \frac{n_{\#}}{2} \rfloor, \dots, n_{\#} - 2\}] \\
 & \cup [(1, i - 1) \mid i \in \{\lfloor \frac{n_{\#}}{2} \rfloor, \dots, 1\}]
 \end{aligned} \tag{4.32}$$

ID^{nb} contains neighboring pairs of indices that start from the center index of the sequence images and go to the right. Additionally, it includes all pairs of indices that go to the left. The correction is conducted for each of the index pairs in the array ID^{nb} . In the following, the correction is explained exemplarily for one pair. Here, a and b are the two indices of the specific pair.

We do not use all pixels for the correction but rather those that are within a threshold. This excludes unreliable pixel values and thus makes the correction more adequate. Note that the second image can have a longer or shorter exposure time than the first depending on the current direction. Thus, we have to use different thresholds depending on the direction. When we correct from left to right, it means that the second image has higher values than the first. Here, it is valid to take into account all reliable values, i.e., everything between 0.1 and 0.9. By contrast, if we correct from right to left, the second image has lower values than the first. As a consequence, we cannot take into account values whose multiplication with the expected ratio would be greater than 1. This would lead to incomparable values since we never have a value greater than 1 in the array of the first image.

For example, let's say we have in the first image a pixel with value 0.8. If the exposure times of the pair are 4 seconds and 2 seconds, we get an expected ratio of 2.0. In order to check if the ratio is preserved between the image pair, we would expect in the second image a value of 1.6 which is not possible due to the limitation of the camera. Thus, we chose for this case the threshold of 0.4 to cover even the biggest ratio between exposure times. The threshold t^p for a pair is defined as

$$t^p = \begin{cases} 0.9 & \text{if } a < b \\ 0.4 & \text{otherwise} \end{cases} \quad (4.33)$$

Now take all pixels positions of the second image that lies within the thresholds.

$$ID^2 = \{(x, y) \mid x \in \{0, \dots, w^{img}\} \wedge y \in \{0, \dots, h^{img}\} \wedge 0.1 < SI_{b,x,y}^{comb} < t^p\} \quad (4.34)$$

It can happen that there are too little points within the thresholds. If the number of ratios $|ID^2|$ is lower than 100, we do not perform the correction for this pair of images since the mean ratio is not expressive. In that case, the pair is skipped. If we have more than 100 pixels, we compute the ratios between first and second image and take the mean \bar{R} of the resulting ratios.

$$\bar{R} = \frac{1}{|ID^2|} \sum_{(x,y) \in ID^2} \frac{SI_{a,x,y}^{comb}}{SI_{b,x,y}^{comb}} \quad (4.35)$$

The correction factor fac for the current pair is computed as follows

$$fac = \frac{E_a}{E_b} \cdot \frac{1}{\bar{R}} \quad (4.36)$$

$$SI_{b,x,y}^{corr} = \begin{cases} fac \cdot SI_{b,x,y}^{comb} & \text{if } SI_{b,x,y}^{comb} < 1 \\ SI_{b,x,y}^{comb} & \text{otherwise} \end{cases} \quad (4.37)$$

The correction factor fac is the deviation of the ratio mean from the ratio of their exposure times. Each pixel from the second image of the pair is multiplied by F except for overexposed ones. This is because correcting them does not make sense since we do not know their real value. This is crucial especially in the case that the correction factor F is smaller than 1. Here, we would decrease the sensor response to a number smaller than one even though the actual response was potentially higher.

$$SI_{b,x,y}^{corr} = \min(\max(SI_{b,x,y}^{corr}, 1.0), 0.0) \quad (4.38)$$

We have to clip the resulting values of the corrected images to the range between 0 and 1. It is possible that multiplication between the correction factor and a pixel value generated a value greater than 1 which is an overexposure of the pixel. SI^{corr} is a 3D array that contains the corrected sequence images.

The sequence image in the middle is taken as a starting point for the computations of the other images. Its corrected version is not computed yet. Since it is used as a reference, there is no correction for it, and we can take the unchanged original image.

HDRI Regression The energy for all pixels in an HDR image, should be measured under the same conditions. Unfortunately, this is not true for the sequence images. Even though there are the same exposure times available for all pixels, some of them are over- or underexposed so that we have to discard these measurements. Thus, we have a different set of exposure times for each pixel. To have comparable energies, we need to be able to sample each pixel at the same exposure time. The solution is to artificially generate further values by fitting a line through the available data points using linear regression. As a result, we get a function that is independent of the measured points and thus can be sampled at any position.

In this step, we compute the HDR image. It is a pixel-wise computation. First, we have to discard the data points where the pixel value is over- or underexposed. These measurements have no validity. For example, if the value is 1, it can be overexposed. In that case, we cannot tell anymore what the real sensor response was since it is clipped to the maximum value of 1.

The reduction of both the set of exposure times and the corresponding sensor responses is performed as follows

$$ID^{xy} = [n \in \{0, \dots, n_{\#} - 1\} \mid 0 < SI_{n,x,y}^{norm} < 1] \quad (4.39)$$

$$X^{xy} = \cup_{i \in ID^{xy}} E_i \quad (4.40)$$

$$Y^{xy} = \cup_{i \in ID^{xy}} SI_{i,x,y}^{corr} \quad (4.41)$$

ID^{xy} contains the indices for each pixel xy where the sensor response is not over- or underexposed, i.e., in the range between zero and one. Note that the extremes of the range are excluded. These indices are used to reduce the exposure times and the corresponding sensor responses to the subsets X^{xy} and Y^{xy} . With the reduced set of points, we perform a linear regression. Slope and y-intercept of the linear function are computed as following

$$a^{xy} = \frac{\sum_{i=0}^{|X^{xy}|-1} (X_i^{xy} - \bar{X})(Y_i^{xy} - \bar{Y})}{\sum_{i=0}^{|X^{xy}|-1} (X_i^{xy} - \bar{X})^2} \quad (4.42)$$

$$b^{xy} = \bar{Y} - a\bar{X} \quad (4.43)$$

where a is the slope, b the y-intercept, \bar{X} the mean of the X values, \bar{Y} the mean of the Y values. An example of the fitting of one pixel is depicted in figure 4.13. Here, the x-axis shows the exposure times while the y-axis shows the sensor response.

$$SI_{x,y}^{hdr} = \begin{cases} a^{xy} \cdot 1 + b^{xy} & \text{if } |ID^{xy}| \geq 4 \\ 0 & \text{otherwise} \end{cases} \quad (4.44)$$

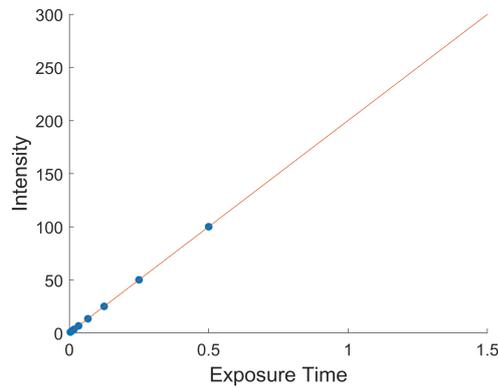


Figure 4.13: Result of linear regression for a particular pixels. Note that depending on the LDR values of the pixel both slope and y-intercept change.

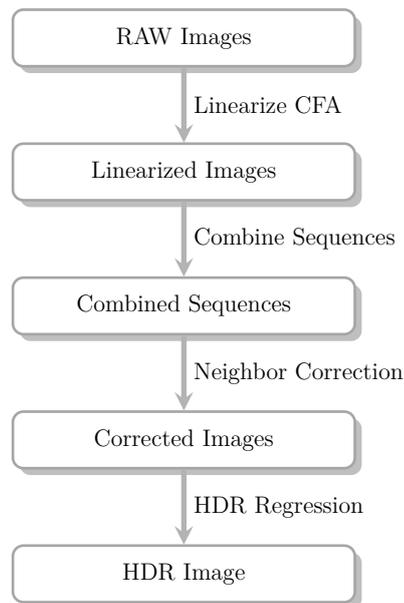


Figure 4.14: *The figure shows the image processing pipeline for sequence images.*

We chose to sample the regressed function for each pixel at the position 1. This is done by using the previously computed slope and y-intercept. Some materials have a high light absorption, or they are non-reflective for specific angles. It can happen that for such materials some pixels are underexposed for most exposure times. This results in a small set of sample points such that the reliability of the regression cannot be guaranteed. In case that a pixel is not properly exposed in at least four different images, the fitted function is not used and HDR value becomes 0. An overview of the complete procedure for processing sequence images to the final HDR image is given in figure 4.14.

Patch Normalization In addition to the high-frequency fluctuations of the LED we need to reduce the low-frequency ones. These fluctuations cause the lamp to have different illumination intensities over a longer period of time. Thus, the same scene can have different HDR images even if they should be at least very similar to each other. For normalizing the HDR image, we need an object which is visible in all images. As we want each HDR image to have the same base illumination, this particular object should also have the same intensity among them. The marker on the cylinder holder suits well for this purpose.

First, we need to know which pixels in the HDR image belong to the white area within the red rectangle. The white area is computed with the aid of the marker contour $Cont^{mark}$ from equation 4.9. We build an image that contains only the thin contour lines called CI^{cl} .

$$CI_{x,y}^{cl} = \begin{cases} 1 & \text{if } (x, y) \text{ on } Cont^{mark} \\ 0 & \text{otherwise} \end{cases} \quad (4.45)$$

The contour lines in the image are expanded using a morphological dilation as

$$K_4 = \begin{bmatrix} 1 & \dots & 1 \\ \vdots & \ddots & \vdots \\ 1 & \dots & 1 \end{bmatrix}^{70 \times 70} \quad (4.46)$$

$$CI^{morph} = CI^{cl} \oplus K_4 \quad (4.47)$$

where \oplus denotes the dilation with the kernel K_4 . It thickens the contour line such that it overlays the red area of the marker. Now the white area pixels can be extracted as follows.

$$CI^{ic} = \max(CI^{cont} - CI^{morph}, 0.0) \quad (4.48)$$

where CI^{cont} was defined in equation 4.10 and contains all pixels inside the markers contour. The subtraction is necessary to make sure that we have only pixels inside the marker. Since we do not want to average together red and white pixels, we have to separate them. The average of the inner pixels is computed as

$$avg^{in} = \frac{1}{h^{img} \cdot w^{img}} \sum_{x=0}^{w^{img}} \sum_{y=0}^{h^{img}} SI_{x,y}^{hdr} \quad (4.49)$$

We want the average illumination of the white area to be 1 in every HDR image. This is achieved by dividing each pixel in the HDR image by the previously computed average.

$$SI_{x,y}^{hdr_{corr}} = \frac{1}{avg^{in}} \cdot SI_{x,y}^{hdr} \text{ with } avg^{in} \neq 0 \quad (4.50)$$

$SI^{hdr_{corr}}$ is the image with the corrected HDR values. We are not interested in the whole image but rather in the values of the pixels whose rays intersect the cylinder. Thus, these points are extracted from $SI^{hdr_{corr}}$ and stored in a linear array denoted as I_i^{hdr} where i is the point index.

4.3.4 Lamp Position

The lamp position is needed for defining further geometric information about the surface points. Before we can compute it, we have to first capture the calibration and sequence images for a highly specular material. For this purpose, a transparency sheet is used. The whole pipeline as explained until now including the geometric and radiometric calibration is applied to these captures.

Furthermore, I make the assumption that the brightest spot in the HDR points I^{hdr} of the transparency sheet is the position where we have the perfect reflection of the light source. It is extracted as following

$$ID^m = \{i \mid i, j \in \{0, \dots, |I^{hdr}| - 1\}^2 \wedge \forall j : I_j^{hdr} \leq I_i^{hdr}\} \quad (4.51)$$

We can use the geometric information of the pixel to compute the mirrored vector at the normal. It is defined as follows. Note that the vectors V^{norm} and V^{out} are defined in the next section.

$$V^{mirr} = 2 \cdot \langle V_{ID^m}^{norm}, V_{ID^m}^{out} \rangle \cdot V_{ID^m}^{norm} - V_{ID^m}^{out} \quad (4.52)$$

Additionally, we have to measure the distance between the light source and cylinder center by hand. It is used to approximate the position of the light source along the mirrored vector. Unfortunately, it cannot be computed and is an unknown factor in the equation. The final lamp position is defined as follows.

$$TV_{ls}^{hold} = P_{ID^m}^{sp} + d^{ls} \cdot V^{mirr} \quad (4.53)$$

where d^{ls} is the light source distance and TV_{ts}^{hold} the light source position in the cylinder holder space. I found out that taking the maximum value of the HDR image is not reliable enough for the position computation. Thus, the prior computation is performed for the 50 brightest HDR pixels individually. The resulting positions are averaged to one.

4.3.5 Geometric Values

In this step, we combine all previously gathered information to define the vectors and angles which are used as input to the BRDF for the respective HDR values. We need them to form input-output pairs when computing the BRDF. Each of the following vectors is computed individually for every visible surface point of the sample. Here, the subscript i denotes the point index within the corresponding geometric and radiometric information, i.e., within P^{sp} and I^{hdr} .

Axis position First, we need to define for each point on the sample its closest point on the cylinder axis. The computation of these points works as following. First, we take the cylinder axis from its local space and translate it to the cylinder holder space using its extrinsics as following.

$$\hat{V}^{ax} = RM_{cyl}^{hold} \cdot \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix} + TV_{cyl}^{hold} \quad (4.54)$$

Next, we define the vector V_i^1 between the sample point and cylinder center. Note that it is not normalized since the length is important for future projections.

$$V_i^1 = P_i^{sp} - TV_{ls}^{hold} \quad (4.55)$$

We compute the vector along the cylinder axis that is as long as the height of the sample point.

$$V_i^2 = \langle V_i^1, V^{ax} \rangle \cdot V^{ax} \quad (4.56)$$

The final point on the cylinder axis is the sum of the cylinder center and the vector along the axis.

$$V_i^{ap} = TV_{cyl}^{hold} + V_i^2 \quad (4.57)$$

Normal Vector The normal vector is the surface normal for each visible point on the sample surface. The direction is given by the vector between the point itself and the closest point on the cylinder axis. Note that it is normalized to have a length of 1.

$$\hat{V}_i^{norm} = P_i^{sp} - V_i^{ap} \quad (4.58)$$

In Vector The incident vector is the direction from each visible point on the sample surface towards the light source position. It denotes the direction where the light comes from.

$$\hat{V}_i^{in} = T_{ls}^{hold} - V_i^{ap} \quad (4.59)$$

Out Vector The outgoing vector is the direction from each visible point on the sample surface towards the camera position. It denotes the direction where the scene is viewed from by the camera.

$$\hat{V}_i^{out} = T_{cam}^{hold} - V_i^{sc} \quad (4.60)$$

Tangent Vector The tangent vector is used as a reference direction for computing the phi angles. It lies in the plane with the normal vector as normal. Thus, it is the vector that is perpendicular to the point normal and cylinder axis. As each surface point has another normal, we have to define the tangent vector for each point individually.

$$V_i^3 = \hat{V}^{ax} \times \hat{V}_i^{norm} \quad (4.61)$$

Since we want to measure the samples from a different direction we rotate them on the cylinder. Simultaneously, we have to rotate the reference vector, i.e., the tangent vector, according to the sample rotation. The rotation is performed as following

$$V_i^{tang} = V_i^3 (\curvearrowright^{s^{ori}}) \hat{V}_i^{norm} \quad (4.62)$$

where \curvearrowright denotes the rotation of the first vector around the second one by the angle s^{ori} .

Bitangent Vector The bitangent vector is a vector that is perpendicular to the tangent vector and the normal vector of each point.

$$V_i^{bitang} = \hat{V}_i^{norm} \times \hat{V}_i^{tang} \quad (4.63)$$

Halfway Vector The halfway vector is the direction between the incident and outgoing vectors.

$$\hat{V}_i^{hw} = \frac{1}{2} \cdot (\hat{V}_i^{in} + \hat{V}_i^{out}) \quad (4.64)$$

Theta In Theta-in is the angle between the surface normal vector and the incident vector.

$$\theta_i^{in} = \arccos \langle \hat{V}_i^{norm}, \hat{V}_i^{in} \rangle \quad (4.65)$$

Theta Out Theta-out is the angle between the surface normal vector and the outgoing vector.

$$\theta_i^{out} = \arccos \langle \hat{V}_i^{norm}, \hat{V}_i^{out} \rangle \quad (4.66)$$

Theta H Theta-h is the angle between the surface normal vector and the halfway vector.

$$\theta_i^{hw} = \arccos \langle \hat{V}_i^{norm}, \hat{V}_i^{hw} \rangle \quad (4.67)$$

Phi In For computing the phi angles, we need to first project the respective vector onto the surface with the normal \hat{V}_i^{norm} .

$$\hat{V}_i^{pin} = \langle \hat{V}_i^{in}, \hat{V}_i^{norm} \rangle \cdot \hat{V}_i^{norm} - \hat{V}_i^{in} \quad (4.68)$$

The phi in angle is the angle between the tangent vector and the projected vector in the counterclockwise direction.

$$\phi_i^{in} = \begin{cases} 2\pi - \arccos \langle \hat{V}_i^{tang}, \hat{V}_i^{pin} \rangle & \text{if } pass_i^{in} < 0 \\ \arccos \langle \hat{V}_i^{tang}, \hat{V}_i^{pin} \rangle & \text{otherwise} \end{cases} \quad (4.69)$$

where $pass_i^{in}$ tells us if the projected incident vector passes the tangent line and thus ϕ_i^{in} is greater than π . We have to distinguish if it does since we need the angle in the counterclockwise direction. Otherwise we would compute the smaller angle without respecting the direction.

$$pass_i^{in} = \langle \hat{V}_i^{in} \times \hat{V}_i^{norm} \rangle \cdot \hat{V}_i^{norm} \quad (4.70)$$

Phi Out The phi out angle is the angle between the projected outgoing vector and the tangent vector. It is defined as follows. The computation works equivalent to the one of the phi in angle. Thus, I omit the explanation for \hat{V}_i^{pout} and $pass_i^{out}$ at this point.

$$\phi_i^{out} = \begin{cases} 2\pi - \arccos \langle \hat{V}_i^{tang}, \hat{V}_i^{pout} \rangle & \text{if } pass_i^{out} < 0 \\ \arccos \langle \hat{V}_i^{tang}, \hat{V}_i^{pout} \rangle & \text{otherwise} \end{cases} \quad (4.71)$$

Phi H The phi halfway angle is the angle between the projected halfway-vector and the tangent-vector. As before the explanation for \hat{V}_i^{phw} and $pass_i^{hw}$ are omitted.

$$\phi_i^{hw} = \begin{cases} 2\pi - \arccos \langle \hat{V}_i^{tang}, \hat{V}_i^{phw} \rangle & \text{if } pass_i^{hw} < 0 \\ \arccos \langle \hat{V}_i^{tang}, \hat{V}_i^{phw} \rangle & \text{otherwise} \end{cases} \quad (4.72)$$

4.4 Diffuse Normalization

The corrected HDR values in I^{hdr} are in fact sensor responses for reflected light at different positions on the sample. All points have a different distances to the light source. Hence, they are illuminated by different amounts of energy and thus neither comparable nor do they express BRDF values yet. We need to interrelate the points and make them independent of the light source.

For this purpose, we need a reference material that has a diffusely reflecting surface. More specifically, it should fulfill Lambert's cosine law. The BRDF values of these so-called Lambert materials are constant and independent of the BRDF parameters. More specifically, the Lambert BRDF is defined as

$$f^{lamb}(\theta^{in}, \phi^{in}, \theta^{out}, \phi^{out}) = \frac{\rho^{diff}}{\pi} \quad (4.73)$$

where ρ^{diff} denotes the albedo of the Lambert material. It is necessary to express the deviation from a perfect diffusely Lambertian surface which would have an albedo of 1. Unfortunately, there is no such material in the real world. We decided to take white paper as reference material since it comes close enough to a Lambert material. It has an albedo of 0.8. The reference values are gathered by measuring and extracting the data of the white paper according to the pipeline. In the following, we denote these data by the subscript *ref*. The data of the sample material is marked by the subscript *samp*.

The reason for choosing a Lambert material as reference is that we can alter its BRDF to be 1 by multiplying all values with $\frac{\pi}{0.8}$. Since viewing direction along the cylinder is not perpendicular to the plane, the measured values of the reference material still contain the cosine factor. Additionally, they are affected by the light pattern that was mentioned in chapter 4.1.3. By dividing sample values by reference values with the same geometric properties, both the cosine factor and the light pattern are cancelled out due to the fact that they are contained in sample values as well. Thus, we express the sample values as scaled version of the reference BRDF which is 1. With this procedure we transform the measured sample values to pure BRDF values.

Before normalizing, we need to find for each sample point a point on the diffuse material that has nearly identical geometric properties. Since both materials are measured under similar conditions, the existence of such a point can be presumed. First, we construct a distance matrix D^2 that contains for each sample point the distance to each reference point.

$$samp_{\#} = |I_{smp}^{hdr}| \quad (4.74)$$

$$ref_{\#} = |I_{ref}^{hdr}| \quad (4.75)$$

$$D^2 = (a_{sr}) \in \mathbb{R}^{smp_{\#} \times ref_{\#}} \quad (4.76)$$

with

$$a_{sr} = \sum_{ang \in \{\theta^{in}, \theta^{out}, \phi^{in}, \phi^{out}\}} \|ang_s - ang_r\|_2^2$$

We calculate for each sample point i the ID of the closest point. Here, the ID denotes the index of the respective point in the list of the reference points.

$$ID_i^{norm} = \{j \in \{0, \dots, ref_{\#} - 1\} \mid \forall k \in \{0, \dots, ref_{\#} - 1\} : D_{i,j}^2 < D_{i,k}^2\} \quad (4.77)$$

In the end, we compute the list I^{norm} which contains the normalized BRDF value for each sample point. As mentioned before, the BRDF value is acquired by dividing the respective sample value by the respective closest reference value. Additionally, a factor is applied to compensate for the deviation from the lambertian material.

$$I_i^{norm} = \frac{0.8}{\pi} \frac{I_{smp,i}^{hdr}}{I_{ref,ID_i^{norm}}^{hdr}} \quad (4.78)$$

4.5 Function Fitting

The normalized BRDF data is rather sparse compared to the full BRDF space. Furthermore, it is inconvenient to use it for further computations due to the high dimensionality and amount of data points. Thus, we fit an analytical BRDF model to the data for getting a more compact representation. The function also provides the possibility to sample the BRDF space outside our measurements. Additionally, measuring errors, i.e., outlier points, are smoothed out.

4.5.1 Extended Ward Model

We fit the data to the anisotropic Ward model which was already explicitly explained in chapter 2.1. However, I adjusted the model slightly to help it matching the measured data better. The problem with the original Ward function is that it expresses the expansion of the anisotropic gloss lobe only in x and y direction. It can happen that the measured data have their maximum expansion in another direction than x or y. Fitting such data to the original model is not optimal. Thus, I added a parameter \angle_l which expresses the rotation of the the tangent vector and the phi angles accordingly. As a consequence, the lobe can be adjusted to the data which provides better fits. The extended Ward model $f^{w'}$ is defined as following

$$f^{w'}(\theta^{in}, \theta^{out}, \theta^{hw}, \phi^{hw}) = \frac{\rho^d}{\pi} + \frac{\rho^s}{4\pi\alpha^x\alpha^y\sqrt{\cos\theta^{in}\cos\theta^{out}}} e^{-\tan^2\theta^{hw}\left(\frac{\cos^2\alpha^n}{(\alpha^x)^2} + \frac{\sin^2\alpha^n}{(\alpha^y)^2}\right)} \quad (4.79)$$

where

$$\alpha^n = \begin{cases} \phi^{hw} + \angle^l & \text{if } (\phi^{hw} + \angle^l) \leq 2\pi \\ \phi^{hw} + \angle^l - 2\pi & \text{otherwise} \end{cases} \quad (4.80)$$

Note that by adding \angle^l it can happen that the angle becomes greater than 2π , i.e. 360 degrees. In that case it has to be adjusted which is expressed in equation 4.80.

4.5.2 Optimization

For the optimization I use a first-order iterative optimization algorithm called gradient descent. Furthermore, I use the Matlab implementation of `fmincon` for the computation. The settings for `fmincon` are not changed and can be looked up in the respective Matlab documentation³.

Parameter Name	Min	Max	Initial Point
ρ^d	0	∞	Random
ρ^s	0	∞	Random
α^x	0	0.5	Random
α^y	0	0.5	Random
\angle^l	0	360	0

Table 4.2: List with optimization parameters for Ward fitting. Min and max denote the boundaries for the parameter while initial point is the parameters starting value for the optimization.

The parameters we are optimizing for are depicted in table 4.2. The algorithm finds a parameter set that minimizes the following objective function.

$$\|w \cdot (I^{norm} - f^{w'}(\theta^{in}, \theta^{out}, \theta^{hw}))\|_2^2 \quad (4.81)$$

with

$$w = \frac{1}{2} \cdot \left(1 - \frac{|\phi^d - \pi|}{\pi}\right) + \frac{1}{2} \cdot \left(1 - \frac{\theta^d}{0.5 \cdot \pi}\right) \quad (4.82)$$

The closer values are to the peak reflection the higher their weight. The weighting is necessary as we have little data for the peak reflection compared to the peripheral areas. Without a proper weighting the Ward model would overfit to the insignificant outer points with less considering the peak.

³<https://de.mathworks.com/help/optim/ug/fmincon.html>

5 Fabrication

In this chapter, I investigate hardware devices for processing a substrate surface in different ways. The goal is to find a fabrication method that can be used to fabricate prescribed appearance in a controlled way. Additionally, the software pipeline for using them to fabricate gloss is explained.

5.1 Hardware

For finding a proper fabrication method, I chose two devices which are used in different ways to fabricate a surface with prescribed appearance. Even though this thesis focuses on the fabrication of 2D materials, the devices are chosen such that they can also process 3D surfaces. This makes it easier to extend the proposed methods to 3D in future work.

As a substrate, I use a transparency sheet for all of the following experiments. The sheet has a highly glossy surface. Thus, it represents already one extreme of the gamut in its original state. By processing the surface, it is possible to make it more matt and go towards the other gamut extreme as much as desired. It is suitable for experiments since it provides a wide range of different reflectances. Additionally, it has similar material characteristics as plastic-like 3D objects as like pieces printed with Objet printer. This makes it possible to transfer the results also to 3D prints and extend the processable range of objects.

For generating anisotropic BRDFs, I decided to process the substrate with line structures. The idea is motivated by the principle behind the microfacet theory. More precisely, we want light rays to hit different geometries from different directions. In theory, this provides different reflectance properties depending on the direction. A geometry with different BRDFs at two heights also has the described characteristic. For better understanding an example is given in figure 5.1. It shows the side view of such a geometry. Here, the light rays are hitting the surface perpendicular to the lines such that less of them are reaching the lower floor. Thus, the reflection is mostly given by the BRDF of the top part. It is obvious that the effect is stronger the steeper the angle of the light rays. Of course, it additionally depends on the viewer's position. By contrast, if the ray directions are parallel to the lines, more of them are reflected by the lower material. Note that the effect gets emphasized the longer a line is in comparison to its thickness.

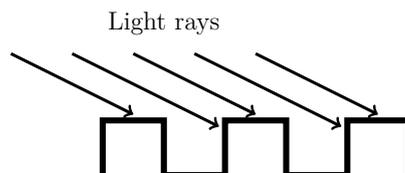


Figure 5.1: *Example for concept of line geometry.*

For measuring and comparing the respective resolutions of each method, we need to know the line thickness. It is measured by fabricating parallel lines. More specifically, we start with a wide distance of 2mm to be sure that the lines are not touching. The distance is decreased by 10 μ m for each new line until we reach a distance where the lines are touching. This particular distance is the line thickness.

Fabricating anisotropic surface reflectances is a challenging task. The fabrication methods need to process the geometry of the substrate at a tiny scale. Furthermore, the positioning system requires a high resolution to provide the necessary precision. For making the results reproducible, the method should also have high repeatability which is not apparent as we will see. In the following, I first explain the fabrication devices. After, I present three fabrication methods which make use of them.

5.1.1 Devices

In this section, both devices used for the different fabrication methods are explained together with their respective specifications.

Engraving Machine The first device is a Roland CNC engraving machine⁴. The machine has three axes and provides a good precision at high speed. The resolution in each direction is 10 μ m. The maximum speed is 100 mm/s in x-z direction and 50 mm/s in the y-direction. Each component of the engraver can be controlled individually by its proprietary machine code language. This provides full control over the whole machine. We can vary both the speed and position at each step as we need.

In the first method it serves as a positioning system. More specifically, different pens are attached to the device which moves them over the substrate. The pens are filled with different varnishes that are deposited on the substrate. Furthermore, in the third method the engraver is used to scratch the substrate surface with several attached tools.

Laser Cutter The second device is an Epilog Zing 20W⁵ laser cutter. The laser is operated in vector mode. This provides the possibility to mark lines. The device has three axes, but just two of them can be affected during marking. The third axis, i.e., the y-axis, is used for moving the material into the laser focus. The laser cutter has three parameters that can be varied. More specifically, the parameters are power, speed, and frequency. Both the power and speed are in the range [1 and 100]. The frequency is defined in Hertz within the range [1,5000] where a value of 1 Hz means that the laser is turned on permanently.

⁴<https://www.rolanddga.com/products/engraving-machines/egx-400-and-egx-600-professional-cnc-engraving-machine>

⁵<https://www.epiloglaser.de/Lasermaschinen/zing-engraver-cutter/>

5.1.2 Methods

In this section, the different investigated fabrication methods are explained and evaluated regarding their strengths and weaknesses.

Varnish The first method is an additive approach. It works by applying different varnishes to the transparency sheet. The varnishes affect the reflectance properties of the substrate as they are applied as an additional layer to it. They are deposited by a pen loaded with the respective liquid. The pen is attached to the engraving machine from chapter 5.1.1 using a 3D printed holder. Since we have many different sized pens, we also need an own holder for each of them. The machine moves the pen and thus deposits the varnish as it touches the surface. It is enough for the pen if it slightly touches the substrate to draw on it. Thus, the fabrication depth for the pens is one step down, i.e., 10 μ m.

Pen Name	Tip Type	Tip Size (in mm)
Schmincke Aero Color Liner	Plastic Fibres	0.8
Montana Acrylic Marker	Plastic Fibres	0.7
Rotring Isograph	Solid Plastic	0.7

Table 5.1: *List with tested pens.*

In general, there are not many refillable pens on the market that can draw with varnishes. The ones I tested are depicted in table 5.1. There were a few more pens, but I did not include the ones that did not work at all with varnishes. The problem with these pens is that they are made for ink. We have pens with two different types of tips. The first type is made of plastic fibers. Here, the working principle is capillary action. This physical effect lets a liquid flow into a narrow space without any additional external force like gravity or pressure. The space between the fibers fills with varnish due to the capillary action. As soon as the bottom of the tip touches the surface, the varnish is deposited on the substrate, and new varnish flows from top to bottom. Additionally, there is a space incorporated where air can get to the varnish tank. Otherwise, the varnish would not flow due to underinflation in the tank. The second type of tips works similarly. They use the capillary action as well. By contrast, these tips are solid and have a small channel in the middle. The varnish flows through the channel where the effect of capillary action applies.

I bought different water-based, oil-based and resin-containing varnishes from various fabricators. A list of the varnishes together with their properties is given in table 5.2. Each pen-varnish combination was tested. The experiments have shown the following outcome.

Varnish Name	Base	Type
Lascaux 2031	Water	Glossy
Schmincke Aero 50610	Water	Glossy
Air Vallejo 70.510	Water	Glossy
Swingcolor 2in1 5961	Water	Glossy
Kreul Solo Goya 169	Oil	Glossy
Schmincke 50065	Synthetic Resin	Glossy
Lascaux 2061	Water	Matt
Schmincke Aero 50611	Water	Matt
Air Vallejo 70.520	Water	Matt
Swingcolor 2in1 5960	Water	Matt
Kreul Solo Goya 170	Oil	Matt
Sennelier V�ernis	Oil	Matt
Schmincke 50044	Synthetic Resin	Matt

Table 5.2: *List with tested varnishes.*

All pens stop drawing after a certain amount of time independent on the loaded varnish. Especially when using matt varnishes the time not predictable and lies between a couple of seconds and a minute. This is because matt varnishes contain small particles which make them matt in the first place. These particles stuck in the small space between the fibers or in the channel, preventing the fluid from flowing. The tip size does not affect the time because the space between the fibers or within the channel is independent on the tip size.

Even though the glossy varnishes contain less or no particles at all, the drawing duration is limited to a couple of minutes. Again there is no observable pattern regarding the exact time. This makes reliability unpredictable. Here, the problem is that the varnish dries out after some time and the tip becomes clogged. A deposition of varnish is then not possible anymore. There is a further observation regarding the glossy varnishes. In comparison to the water-based ones, the oil-based and resin-containing varnishes lasted longer. I assume that it is because the contained water makes the varnish drying faster. This does not apply to the matt varnishes since it clogs the tip with the particles such that there is no time for it to dry. Due to the unpredictable drying out, it is hard to make a point about the effect of the tip size on glossy varnishes.

Due to the two stated problems and missing alternative pens, we started designing our pens. The goal was to have a pen with a channel that is large enough to let the varnish flow including its particles. Additionally, it should be narrow enough to make thin lines and prevent the varnish from dripping on the substrate. To prevent underinflation in the tank, I left the top of the pen open. There is a further design restriction. The pressure that is applied at the bottom of the pen is depending on the height of the head of liquid. This, in turn, is defined by the height of both channel and tank. Therefore the tank has to be designed as wide as possible to provide enough space for a fair amount of varnish. The channel should not be too long.

By experimenting, I found out that there is no trade-off for the width of the channel such that it neither drips nor becomes clogged. Thus, I made the channel thicker and added a cap with a small hole to the pen top. The hole size controls the flow of varnish. Unfortunately, the channel is still too thin such that the particles clog it after some time. Making the channel even broader is not an option because then it starts letting air in. Thus, the flow does not depend on the hole in the cap anymore, and the pen begins dripping.

With the designed pens we could not find any pen-varnish combination that works as desired. Even if we would accept the short durability per pen, the approach has one main drawback. As all pens have at least a tip size of 700 μm and the fluid spreads after deposition, the drawn lines are thick. Thus, we cannot get the desired anisotropic effect since the necessary microgeometry is not present. Additionally, the line thickness restricts the possible resolution.

It should be mentioned that it could be possible to mix new varnishes that have smaller or fewer particles. Additionally, changing viscosity could provide better results. Properly testing these approaches requires further chemical experimenting. However, such experiments are out of the scope of this thesis.

Laser Marking The second method uses a laser cutter which makes it a subtractive approach. It marks the top of the substrate by heating and melting the surface locally. This makes it appear more matt and thus affects the reflection properties. The transparency sheet is very sensitive to the laser beam. By experimentally exploring the parameter space I found out that there is just a limited set of parameters we can use. All parameter sets outside this space destroy the substrate or do not affect it at all. I set the power to 3% and the traveling speed to 100%. The frequency was chosen as 350 Hz.

The laser approach has some advantages over the varnish approach. With the stated parameters, it can generate lines that have a thickness of about 100 μm . Narrower lines provide more anisotropy and a better resolution. Also, the speed is higher because the power-up delay of the laser is low. Additionally, the laser cutter offers better reliability. However, there are also drawbacks. Since the surface is processed by heating it with the laser beam, the substrate deforms. Even if fixed at the borders, the tension is high enough to distort the transparency sheet still. The deformation has the consequence that parts of the substrate are getting more and more out of focus. Thus, it happens that parts are not lasered properly or not affected by the laser beam. The deformation also limits the possible line depth. It sets a limit to the power which does not allow us to generate deeper lines. This also limits the anisotropic effect microgeometry is not deep enough.

Tool
Milling Bit
Drilling Bit
Diamond Engraving Tip
Scriber
Milia Knife
Needle
Scalpel

Table 5.3: *List with tools used for scratching.*

Furthermore, fume arises due to the heating process. Even though there is a ventilation system in the laser area, some of the fumes set on the substrate and changes the glossiness. The amount and effect of the fume are not predictable and thus problematically regarding the repeatability of the system.

Scratching This subtractive method uses the engraving machine. I attached different tools to the machine which are used to scratch the surface of the substrate and thus introduce the mentioned line-shaped microstructures. Furthermore, I tested the tools listed in table 5.3. They were attached to the machine by a 3D printed holder printed for each of them separately. This approach is inspired by the work of Regg et al. [22]. They use the engraver to fabricate microgeometries into acrylic substrate.

By contrast to the laser cutter, the fabrication process with the engraving machine uses tools. The processing of the substrate surface is controlled by physical contact. More specifically, the strength and thus the line thickness is depending on the applied force. We define a fabrication depth for each tool. The starting position for the depth is defined as the lowest height where the tool is not affecting the surface.

Except for the needle and the scalpel, all other tools have one common problem. They are not uniformly round shaped and thus blunt on some of their sides. Depending on the current mounting, it is possible that the tool is not sharp enough to cut through the substrate. As a consequence, it instead sweeps over the surface and wrecks it. Since the sharpness cannot be seen by eye, it is not possible to know at which rotation the tool should be mounted to its holder. This unpredictable behavior makes these tools unusable. By contrast, the needle is sharp enough towards each direction. However, even thicker needles bend to fast. Thus, they rather bend before they cut because it requires less force. Additionally, I tried to tilt each of the tools without significant improvement.

The scalpel is an exceptionally sharp bladed instrument which is initially used for surgeries. An image of the attached scalpel to the engraver is depicted in figure 5.2. It is the most promising tool I tested since it fits our needs perfectly. It can scratch thin lines into soft materials. Since it is made to cut in one direction, we know in which direction to attached it to its holder such that it can cut through the material. The fabrication depth for the scalpel is $20\mu\text{m}$. Experiments have shown that the scratches at $10\mu\text{m}$ are barely visible while $30\mu\text{m}$ is too deep and produces messy lines. Due to the resolution of the engraving machine, we cannot use other values in between. The scratch thickness is $90\mu\text{m}$.

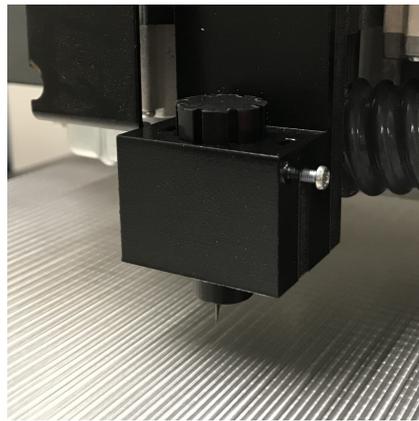


Figure 5.2: *Scalpel holder attached to the engraving machine.*

Scratching with the scalpel combines the advantages of both the varnish and laser approaches. Due to the sharpness of the blade, it has a high resolution while providing a reliable system. If we use the same parameters, the tools produce the same scratch on the substrate. Furthermore, we can increase the line depth with the scalpel further than with the other tools because the blade is thin to get deeper into the material. Thus, we can get a higher anisotropic effect. Due to the stated advantages of scratching in combination with the scalpel was chosen as the fabrication method of this thesis. It is used for building and evaluating the model.

5.2 Software

During fabrication, we vary the intensity of the induced effects by the parameters of the line coverage. More specifically, the line coverage is the number of lines per area on the substrate surface. In the following, I explain the necessary steps for fabricating a sample with spatially-varying line coverages. As input, we take a grayscale image with different gray values in the range between 0 and 1. Here, the gray value denotes the line coverage in percent.

Halftoning It is not possible to continuously vary the intensity of a scratch itself. Any space on the substrate can be scratched or left blank. Thus, the continuous-tone input image of line coverages has to be halftoned.

The naive approach is to fabricate a pixel on a bigger scale. More specifically, its size is expanded such that multiple lines are fitting into its space. By filling the pixel with more or fewer, the continuous tone can be fabricated. For example, if a pixel has the size of six lines, there would be seven different levels. This approach has a significant disadvantage. There is a trade-off between the image resolution and a reasonable amount of BRDF levels. The scratching method produces lines of $90\mu\text{m}$ thickness. If a pixel is six lines thick, it would have a final size of $450\mu\text{m}$. One could increase the possible levels by making the lines within the pixel shorter. However, this would reduce the anisotropic effect since the anisotropy is increased if the ratio between line width and length is high. Furthermore, the pixel size would still be quite high.

As we can see, even a limited amount of lines per pixel already produces a reduced resolution and does not provide many BRDF levels. Expanding the pixel size by more lines would worsen the resolution even more. Thus, a method is needed that offers many BRDF levels at a proper resolution. For this purpose, juxtaposed halftoning is used as proposed by Babaei et al. [23]. The algorithm translates a greyscale image to a binary bitmap. In the end, the bitmap tells the fabrication method if a specific area should contain a scratch or not. As we will see, the technique provides many levels at a proper resolution in contrast to naive approach. The algorithm is explained in the following.

Figure 5.3 shows a simplified example of the juxtaposed halftoning algorithm. First, the algorithm divides the continuous-tone input image into screens which are formed as parallelograms. One screen is marked by red borders in the left image of the figure. The main idea of the algorithm is to use lookup screens to get the corresponding halftoned value for each pixel. At this point, it is essential to know that there are many different lookup screens. More specifically, different gray values have to use other lookup screens depending on their value.

To clarify the concept, I explain the procedure on the basis of the upper right gray pixel from the example image. More specifically, it is the pixel in the fourth row and seventh column which has a value of 0.5. The respective lookup screen for the value of 0.5 is depicted in the right image of the figure. This lookup screen is half-filled and thus corresponds to the pixel value. The halftoned value for this particular pixel is given by the value of the pixel in the lookup screen that is at the same relative position within the screen. In the case of the example pixel, the corresponding pixel has a value of 1 as it is filled in the lookup screen. In contrast, the halftoned value of the pixel in the eighth row and the third column is zero. It has the same continuous-tone value as the other pixel and thus uses the same lookup screen. However, its relative position within the screen is different. More specifically, the pixel of its relative position in the lookup screen is not filled. Thus, it also gets the halftoned value zero. The same procedure applies for pixels with other values.

In this way, the algorithm determines the final halftoned screen by mixing different pixels of different lookup screens together depending on original continuous-tone values. Note that the more pixels within the screen in the input image are mapped to the same lookup screen, the more the halftoned screen looks like the respective lookup screen. In the extreme case, the lookup screen is mapped 1:1 to the halftoned image.

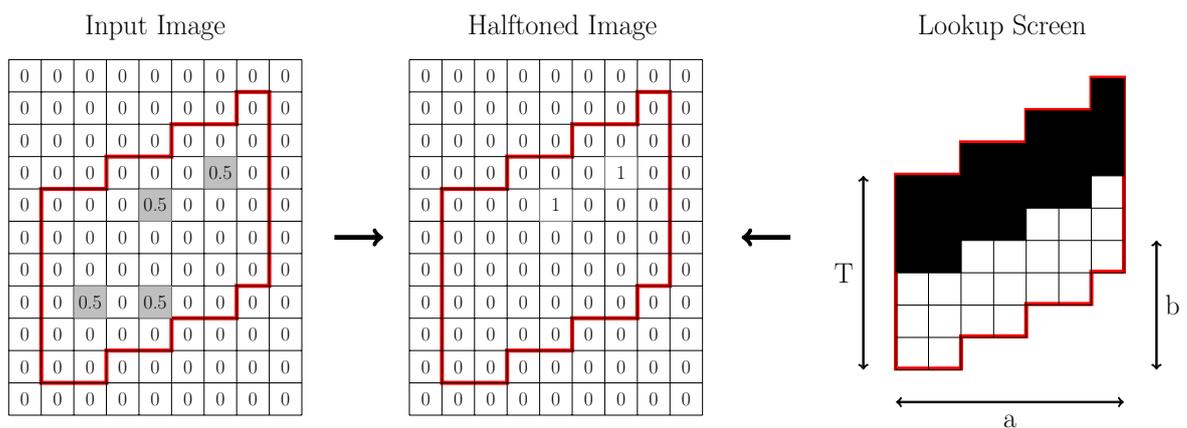


Figure 5.3: Example for juxtaposed halftoning.

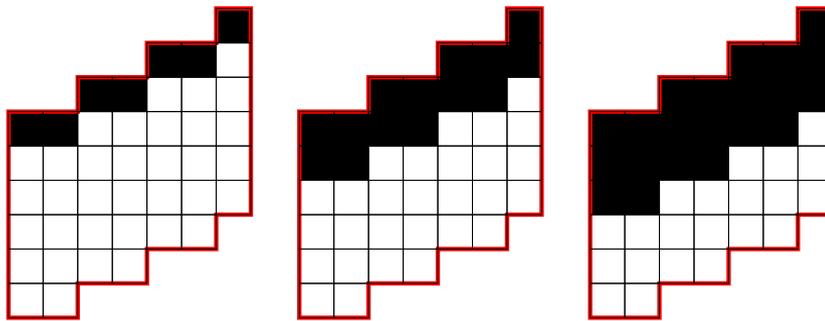


Figure 5.4: Example for differently filled lookup screens with the levels $7/42$ (left), $14/42$ (middle) and $21/42$ (right).

The limit of producible levels is given by the number of pixels in a screen. In the example, the screen consists of 42 pixels and thus provides 43 levels. For each of these 43 levels, a differently filled screen is defined. Three of them are depicted in figure 5.4. They have the levels $7/42$, $14/42$ and $21/42$. Of course, the input image can have a lot more levels than we have screens, i.e., each possible floating point number between 0 and 1. In that case, the continuous input has to be mapped to the lookup screen with the closest level.

The pixels in the lookup screens are filled with a given pattern. More specifically, the pattern is defined as a discrete line and fills the screens along the lines pixel by pixel. Thus, juxtaposed halftoning suits well for our purpose since it generates line-like structures that can be fabricated by the presented devices.

The screens are filled along the line has with the slope of a/b which also denotes the orientation of the screen element [24]. The screen thickness T expresses its height and thus the screen period. For all the following computations, the parameters are defined as depicted in table 5.4. The corresponding screen has like in the example 42 pixels and thus 43 different levels. I chose this specific parameter set since it provides a good trade-off between screen size and levels.

Parameter Name	Value
a	4
b	7
T	6

Table 5.4: List with parameters for juxtaposed halftoning.

Vectorization The bitmap is not yet ready to be fabricated. It contains only pixels with values 0 and 1. Since the fabrication device needs lines, the halftoned image has to be translated into lines with explicit start and end points. As already mentioned, the screens that were used to create the halftoned image, are filled by discrete lines. Thus, it makes sense also to use them for the definition of the fabrication lines. For this purpose, it is necessary to have information about the line pattern for checking which parts of the respective line should be scratched.

Each screen is consisting of T same stacked discrete lines. Thus, it is enough to know the pattern of one line within the screen. Here, pattern denotes the structure that forms a discrete line. Since the discrete line is slanted, it is not apparent which pixels belong to the line. The concept is best understood by examining the example depicted in figure 5.5. It shows a screen marked by a red border. The first discrete line is emphasized by black pixels. Here, the number in each pixel denotes the pattern, i.e., the y-coordinate of the line at a specific x-coordinate. Note that these are relative coordinates and thus start at 0. The numbering makes it easier to translate the pattern to other parts of the image.

The pattern can be extracted by halftoning an image that fills exactly one full discrete line per screen. We start at the first upper left pixel of the screen and follow the black pixels while keeping a record of the pixels heights. The pattern has a length of b since this is the width of a screen. We can generate the necessary binary image by halftoning a continuous-tone image with the following gray value

$$\frac{\text{Line Length}}{\text{Number of Screen Pixels}} = \frac{b}{b \cdot T} = \frac{1}{T} \quad (5.1)$$

For our parameters, the required gray value is $7/42$. We can see that this gray value fills 7 out of the 42 available pixels in the lookup screen. Thus, it contains exactly one discrete line. As already mentioned, a grayscale image with identical values produces a halftoned image containing screens equally to the lookup screen. Thus, the halftoned image contains only full discrete lines.

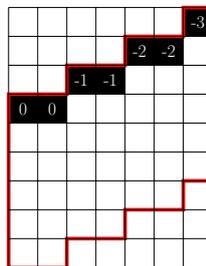


Figure 5.5: Example for concept of line extension and shifting for different heights.

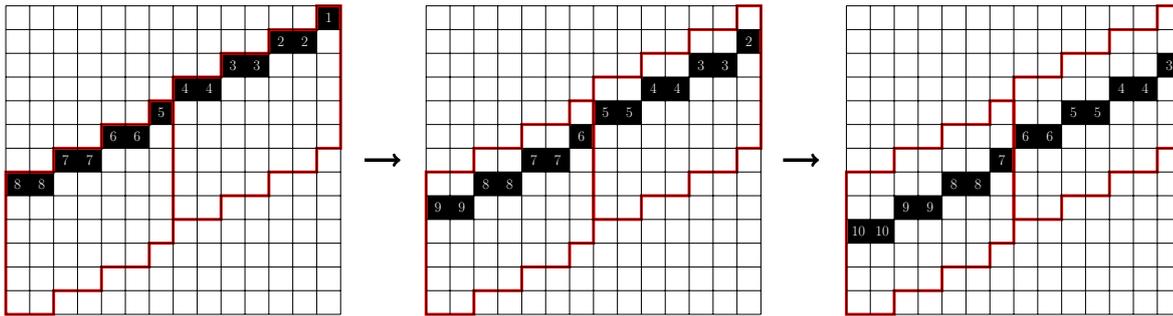


Figure 5.6: *Example for concept of line pattern.*

Fortunately, the screens are tiled continuously. Thus, the line of one screen is continued with the same pattern consecutively. With this knowledge, it is possible to extend the line until it is long enough to cover the whole width of the image. This behavior is shown in the left image of figure 5.6. Additionally, the two images on the right show how the line can be shifted downwards to cover the next line. This is repeated for each y-coordinate of the image.

The next step is to check which parts of a discrete line have to be scratched. For supporting the explanation the procedure is illustrated in figure 5.7. The idea is to construct for each y-coordinate a mask that contains only the discrete line for this specific height. The mask is put over the halftoned image as it contains only one pixel at each x-coordinate the resulting image does as well. Thus, it can be reduced to one dimension since the second dimension does not provide useful information.

In the end, the 1D array can be split into line segments. They define the parts that have to be scratched along the vector of the discrete line that was used in the mask. Note that this procedure has to be repeated for each y-coordinate of the halftoned image.

Postprocessing At this stage, the line coordinates are in the image domain. Postprocessing of their coordinates is required to make sure that the distance between two lines to equals to the scratch thickness. The lines coordinates are rotated such that the lines are aligned to the y-axis. This has two advantages. First of all, it is necessary that the lines are aligned to the scratching direction of the scalpel. The scalpel holder is designed such that it can be attached in 45-degree steps to the machine. Thus, aligning its scratching direction with one of the main axes is the most convenient way. Here, the y-axis was chosen as scratching direction.

A further effect of the rotation is that the lines can be spread easier to have the correct distances to each other. After the rotation, it is achieved by multiplying them with the scratch thickness which transforms their metric system to millimeter. For the unaligned lines, this would not work. Their distance is given along the perpendicular vector of each line. To get proper line distances, the coordinate system has to be extended such that the scratch thickness scales the perpendicular vector. This is indeed possible but requires a more complicated computation. The rotation angle is computed by

$$\angle^a = -\arccos\left(\left\langle \begin{bmatrix} 0 \\ 1 \end{bmatrix}, \frac{ls}{\|ls\|} \right\rangle\right)$$

with

$$ls = \begin{bmatrix} a \\ b \end{bmatrix}$$
(5.2)

The lines are rotated as following

$$\begin{bmatrix} \cos \angle^a & -\sin \angle^a \\ \sin \angle^a & \cos \angle^a \end{bmatrix} \cdot L$$
(5.3)

where L are the previously extracted lines. Due to the rotation, it is possible that lines were moved out of the fabrication area of the engraving machine. Thus, they are moved back to be as close as possible to the zero position by subtracting from each dimension its minimum. More specifically, the smallest x/y-value is subtracted from each x/y-coordinate respectively. In the end, the coordinate system is translated to millimeter by multiplying each component with the scratch thickness, i.e., 0.9 mm.

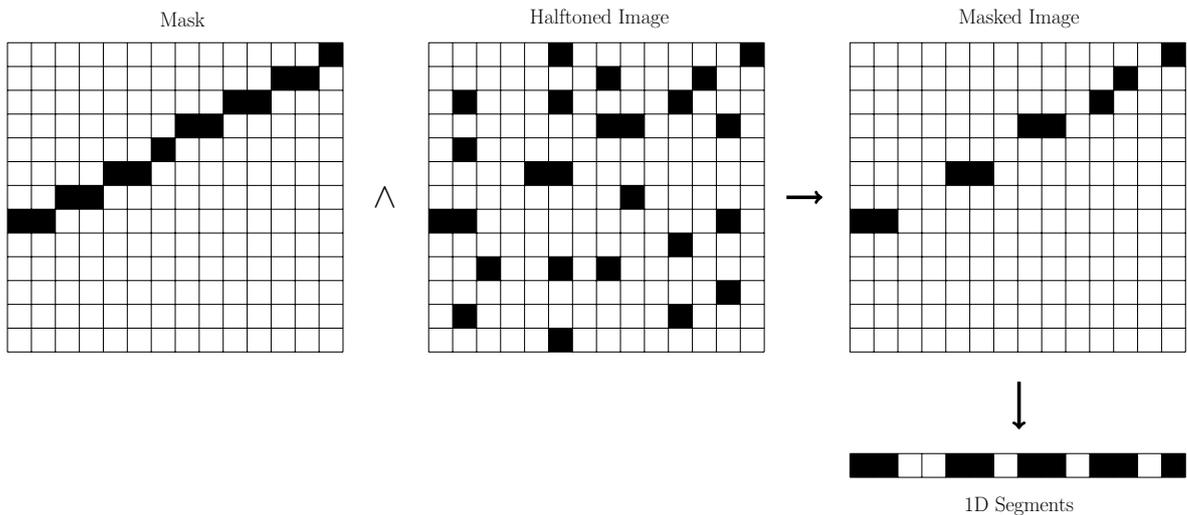


Figure 5.7: Example for extraction of line segments.

6 Modeling

The goal of this thesis is the reproduction of existing materials. Using the measuring setup we can compute the Ward BRDF for one of these materials. Furthermore, the previously presented fabrication method can edit the BRDF of our substrate material. The missing piece is a model that can tell us how the fabrication needs to be controlled for reproducing the measured Ward BRDF of the material on the substrate. More specifically, we need to know how much the substrate needs to be scratched such that the surface BRDF gets as close as possible to the one of the given material that should be reproduced.

For this purpose, we put the line coverage per area in relation to the induced anisotropy and gloss properties. This is a challenging task since the relation is non-proportional and the model needs to express this. Furthermore, fabricating a huge amount of samples for looking up the relationship is not possible due to the fabrication time. Thus, the model is built on top of a small number of fabricated samples. In this chapter, I explain how the necessary model is defined and computed.

6.1 Reproduction Model

The fabrication method edits the appearance of the substrate by scratching the substrate. Thus, it transforms the BRDF of scratched parts to another BRDF. In other words, the surface is mixed from two different materials, i.e., scratched and non-scratched areas. The model is based on the same principle. It uses two base materials each defined by a set of Ward parameters. Given a set of Ward parameters for a material that should be reproduced, it computes the mixing ratio of the two bases that is necessary to get as close as possible to the given material. Thus, an interpolation between the two bases is performed.

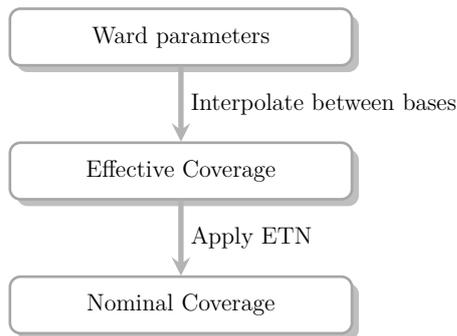


Figure 6.1: *Pipeline of reproduction model.*

The bases express the two extremes of the gamut and are denoted by the indices b1 and b2 respectively. It is explained later how they are defined. For a given set of Ward parameters of a material m , we need to estimate the mixing ratio of the two bases. The ratio of the first base for the material is defined as ce_m^{b1} . Since the ratios have to sum up to one and we have only two bases, the ratio for the second base is

$$ce_m^{b2} = 1 - ce_m^{b1} \quad (6.1)$$

Thus, it is enough to only optimize for ce_m^{b1} because the second ratio is given implicitly. The objective function is defined as follows.

$$\|f_m^{lc} - f_m^{w'}(\theta_m^{in}, \theta_m^{out}, \theta_m^{hw})\|_2^2 \quad (6.2)$$

with

$$f_m^{lc} = ce_m^{b1} \cdot f_{b1}^{w'}(\theta_m^{in}, \theta_m^{out}, \theta_m^{hw}) + ce_m^{b2} \cdot f_{b2}^{w'}(\theta_m^{in}, \theta_m^{out}, \theta_m^{hw}) \quad (6.3)$$

where the subscript m denotes the Ward parameters of the material. f^{lc} computes the linear combination of the two bases with their ratios. I want to emphasize that f^{lc} linearly combines the BRDF values that are sampled from the Ward model using the parameter sets of the bases. The sampling at this point is crucial. A linear combination of the bare Ward parameters is not possible as experiments have shown.

A further step is necessary before the material can be fabricated. The computed mixing ratio of the two bases is not the actual line coverage. That is because the two base BRDFs do not perfectly correspond to the two BRDFs that are mixed on the substrate. However, the non-linear relation between mixing ratio and line coverage can be expressed as a function. In the following, we denote the mixing ratio of the two base BRDFs as effective coverage. Furthermore, the mixing ratio of scratched and non-scratched area on the substrate is the nominal coverage. The translation function from effective to nominal is defined as the effective-to-nominal (ETN) function f_m^{etn} . It is explained later how to compute it. The actual nominal coverage for the second base is defined as

$$cn_m^{b2} = f_m^{etn}(ce_m^{b2}) \quad (6.4)$$

With the ETN function and the effective coverage of the second base, we can compute its nominal coverage cn_m^{b2} . Note that the ETN function is defined for the translation of the second base. A dedicated ETN function for the first base is not necessary. The nominal of the first base can be computed by subtracting the nominal of the second base from 1. This is possible because the nominals of two bases sum up to one as well as their effectives. The model pipeline is depicted in figure 6.1.

6.2 Model Computation

For computing the model components, multiple samples with different nominal coverages are fabricated. Their Ward parameters are estimated with the measuring setup. Both the two bases and the ETN function are optimized using the Ward parameters of the samples and their corresponding nominal coverages. In the following, a detailed explanation of this procedure is given.

Sample fabrication For exploring the gamut and computing bases and ETN function, I fabricated a couple of samples. The process is as previously described, i.e., using the scalpel for scratching lines into a transparency sheet. The scratching parameters are

Parameter Name	Value
Scratching Depth	20 μm
Scratch Thickness	90 μm

Table 6.1: *List with fabrication parameters for sample generation.*

Note that computed model is adapted to the fabrication parameters and type of transparency sheet. Changing one of them requires a re-computation of the model. To explore as much as possible from the gamut I fabricated a total of nine samples with different nominal coverages. More precisely, the chosen coverages are depicted in table 6.2. They are given for the second base since it depicts the number of scratches. The coverages for the first seven samples are distributed uniformly between 0 and 1. I fabricated two further samples to explore how scratching behaves between the uniform coverages.

Sample Name	cn_m^{b2}
0/6	0.0
1/6	0.17
2/6	0.33
4/10	0.4
3/6	0.5
4/6	0.67
7/10	0.7
5/6	0.83
6/6	1.0

Table 6.2: *List with nominal coverages for second base.*

Sample BRDFs The BRDF of the samples is measured and computed using the measuring setup from chapter 4. For covering as much as possible of the BRDF space, the eight LED lights at the center row were used. Furthermore, the samples are captured at the rotation with the angles -45, 0, 45 and 90 degrees. It ensures that both the highest and lowest anisotropic reflections are captured and thus represented in the BRDF.

A further preprocessing step is necessary before the BRDF fitting. Unfortunately, the data of the peak reflection is not reliable. Due to the narrow peak reflection and the camera resolution, we get an aliasing effect. More precisely, we cannot distinguish if it was sampled from a scratched or non-scratched area. Thus, the sampled data within and around the perfect reflection is not comparable among different samples or even different rotations of one sample. Therefore, the peak reflection has to be eliminated. The fitting procedure focuses more on the peripheral reflections and compares the samples using them. We smooth out the peak by applying a clipping to the BRDF values. The function is defined as follows.

$$f^{clip}(x) = \begin{cases} x & \text{if } x \leq 1.0 \\ 1.0 & \text{otherwise} \end{cases} \quad (6.5)$$

Furthermore, \angle^l , i.e., the rotation of the lobe in the Ward model, is initialized to be 2.6 rad. I found out that the lobe is otherwise differently rotated for several samples. More precisely, if the initial rotation is set to 0 rad, the lobe of some samples have a rotation of 2.6 rad while others have a rotation of 1 rad. In this case, also α^x and α^y are swapped. Thus, the lobe is the same but rotated by 90 degrees. Even though the BRDF is also the same but rotated, it is hard for the model to perform a proper interpolation due to the differently turned lobes. By settings the initial point of the lobe rotation close to the final value, it helps the optimization to fit the same lobe rotation for all samples. Additionally, no weighting is applied. It would not make sense to weight the peak higher after it was smoothed out.

Base Parameters The first base expresses one extreme of the gamut and thus the BRDF with the least scratches. Therefore, the first base is fixed and predefined as the BRDF with the Ward parameters of the sample 0/6. Unfortunately, it is not possible to take sample 6/6 since even for the fully scratched sample because the lines are not always perfectly touching. Thus, we have to optimize for a BRDF that suits like a second base.

During the fitting, I found out that the sample 4/6 is an outlier. The BRDF of this particular sample does not fulfill the assumption to lie between 3/6 and 5/6. According to the fitted BRDF, it is rather close to 6/6. The reason is an error during the fabrication process. Therefore, this particular sample is not included during base fitting such that the optimization is run for the eight leftover samples. Chapter 7 evaluates on this

problem and shows results that confirm that proper fitting is not possible if 4/6 is used. Furthermore, it explains more detailed why the fabrication error arises.

The effective coverages are dependant on the bases and thus have to be fitted jointly with the parameters for the second base. In the end, we optimize for 13 parameters, i.e., five Ward parameters for second base and eight effective coverages. The fitted parameters are depicted in table 6.3. The objective function is defined as follows.

$$\frac{1}{8} \cdot \sum_{i \in \{1, \dots, 8\}} \|f_i^{lc} - f_i^{w'}(\theta_i^{in}, \theta_i^{out}, \theta_i^{hw})\|_2^2 \quad (6.6)$$

The resulting base BRDFs are rendered and depicted in figure 6.2. Here, we can see that the second base is more matt and anisotropic. Note that the Ward parameters are fitted based on the clipped data. Thus, the rendered images do not depict gamut extremes. They rather show the extremes that are used for interpolation. Furthermore, the fitted effective and nominal coverages for the bases are shown in table 6.4. As expected they are monotonically increasing and decreasing respectively.

Parameter Name	Min	Max	Initial Point
ρ_{b2}^d	0	∞	Random
ρ_{b2}^s	0	∞	Random
α_{b2}^x	0	0.5	Random
α_{b2}^y	0	0.5	Random
\angle_{b2}^l	0	2π	2.6
ce_i^{b2} with $i \in \{1, \dots, 8\}$	0	1	Random

Table 6.3: List with optimization parameters for model fitting. Min and max denote the boundaries for the parameter while initial point is the parameters starting value for the optimization.

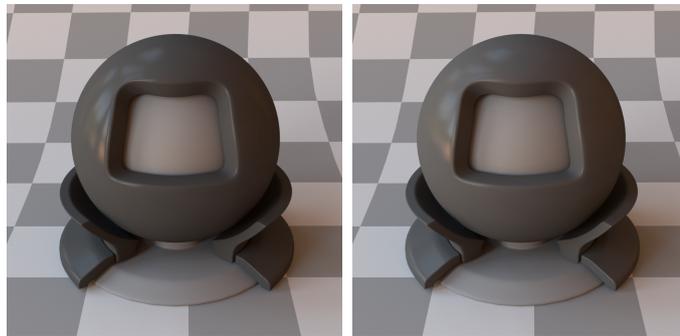


Figure 6.2: Rendered image of BRDFs for base 1 (left) and base 2 (right). The images are rendered with the Mitsuba renderer [25].

Sample Name	c_{b1}^{nom}	c_{b1}^{eff}	c_{b2}^{nom}	c_{b2}^{eff}
0/6	1.0	1.0	0.0	0.0
1/6	0.83	0.61	0.17	0.39
2/6	0.67	0.37	0.33	0.63
4/10	0.6	0.28	0.4	0.72
3/6	0.5	0.24	0.5	0.76
7/10	0.3	0.14	0.7	0.86
5/6	0.17	0.0	0.83	1.0
6/6	0.0	0.0	1.0	1.0

Table 6.4: List with fitted effective and nominal coverages.

ETN Function Knowing BRDF coverages and thus the effective coverage for each sample, it is possible to fit the ETN function. The nominal coverages are already known since they are the scratch amount that the samples were fabricated with. As effective-to-nominal we use a sigmoid-like function. Experiments with other functions have shown that it suits best to the actual data points. The function is defined as follows.

$$f_m^{etn}(x) = \frac{\gamma_1}{(\gamma_3 + e^{-\gamma_2 x}) + \gamma_4} \quad (6.7)$$

The respective parameters with their constraints and initialization points are depicted in table 6.5. Note that the initialization points are chosen such that the function forms a standard sigmoid. The ETN for the second base is fitted using the following objective function. Fitting the ETN function for the first base works analogously.

$$\|cn_i^{b2} - f^{etn}(ce_i^{b2})\|_2^2 \text{ with } i \in \{1, \dots, 8\} \quad (6.8)$$

Parameter Name	Min	Max	Initial Point
γ_1	$-\infty$	∞	1
γ_2	$-\infty$	∞	-1
γ_3	$-\infty$	∞	1
γ_4	$-\infty$	∞	0

Table 6.5: List with optimization parameters for effective-to-nominal function fitting. Min and max denote the boundaries for the parameter while initial point is the parameters starting value for the optimization.

7 Evaluation and Discussion

This section describes various experiments and investigations carried out to evaluate the performance and reliability of the presented methods. It is split into two parts where the first focuses on the BRDF measurement system while the second one is about the BRDF reproduction.

7.1 BRDF Measurements

In the following, the BRDF measurement system is evaluated. I present its repeatability, function space coverage and some measured example materials. In the end, the advantages and limitation are resumed.

7.1.1 Repeatability

High repeatability is necessary to guarantee the reliability of measurements across different materials. For evaluating the repeatability of the system, I measured the transparency sheet ten times. It is highly specular and thus a rather challenging material. That is because the peak of this material mostly depends on the captures with short exposure times. Fluctuations of the light source or even sensor errors are affecting especially these images and thus the resulting peak. Therefore, this test evaluates the radiometric calibration and quality of the normalization steps. Furthermore, the peak of the transparency sheet is narrow such that even a small spatial distortion can lead to different results. The material is removed and reapplied between the measurements. Thus, it also tests the geometric calibration.

For evaluating the difference between measurements, I compute the root mean squared error between each pair of measurements. Note that the data are weighted by the equation 4.82 to consider the peak more. Furthermore, the values for each pair are divided by the maximum illumination among them to get errors between 0 and 1.

The resulting errors of all pairs are averaged which results in a total error of 0.008 with a standard deviation of 0.002. This is an acceptable error for such a challenging material and proves the repeatability of the system.

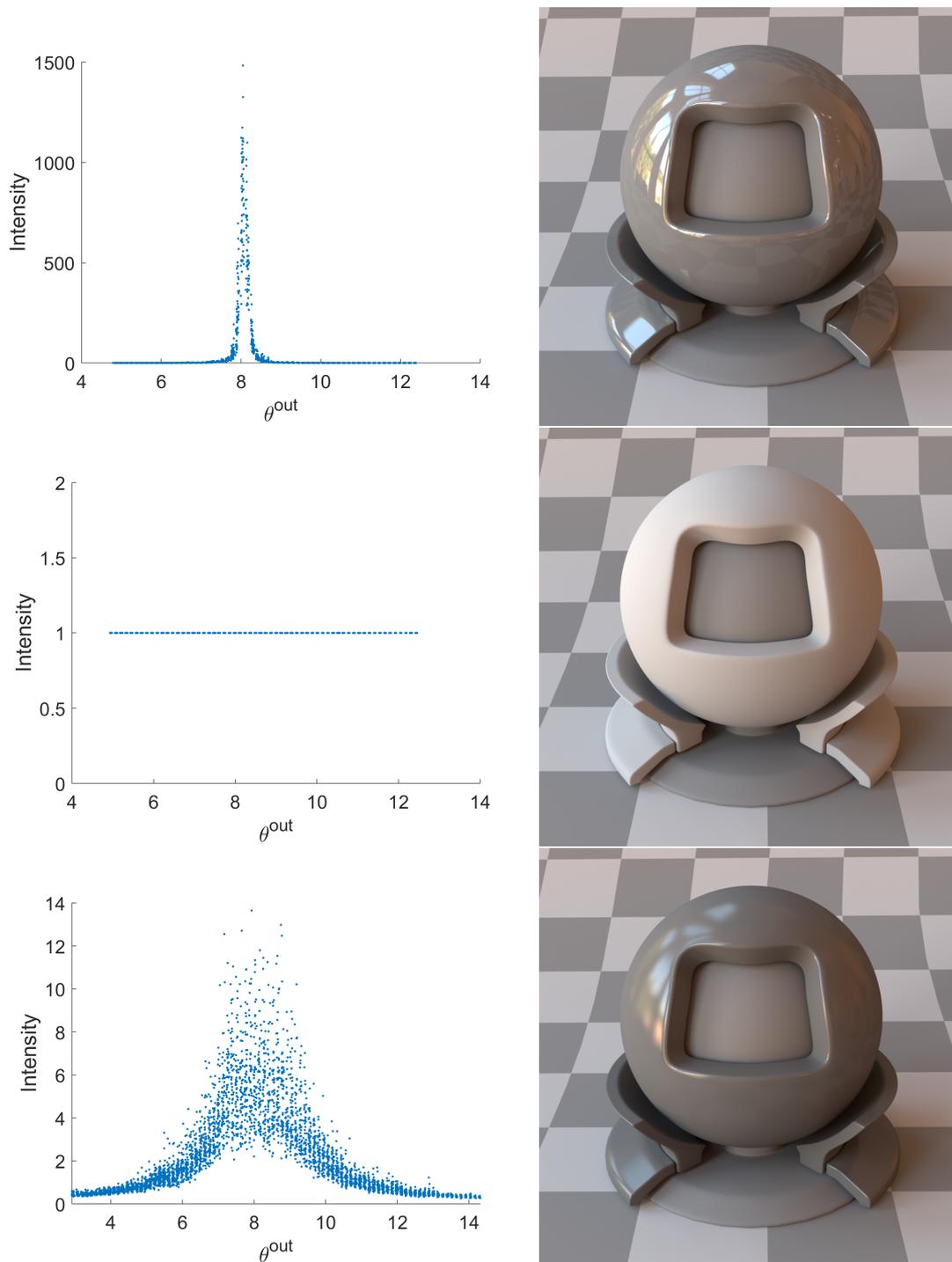


Figure 7.1: *Slice plots of intensities (left) and rendered BRDFs (right) for measured example materials transparency sheet, white paper and polyethylen (top to bottom). The images are rendered with the Mitsuba renderer [25].*

7.1.2 Example Materials

A couple of example materials are measured and rendered to demonstrate their appearance. They are shown in figure 7.1. The plot on the left side shows their function values for different θ^{out} . Here, ϕ^{in} and ϕ^{out} are limited to the incident plane. Furthermore, θ^{in} is fixed to 7.5 degrees to plot only the peak of the first lamp which is located at 15 degrees from the camera.

As expected and according to the real material, the transparency sheet is highly glossy. In comparison to the other two materials, it has a high peak. By contrast, the white paper is very diffuse and not glossy at all. Remember that this material is used as a reference such that all other materials are a scale of it. Thus, it represents a constant function with the value one. This is obvious since the scale of itself is one. Furthermore, the polyethylene has a blurry appearance. Its function peak is therefore rather low.

7.1.3 BRDF Space Coverage

The measured subspace of the BRDF function space is crucial for a proper fitting of the original BRDF. The definition of the space differs depending on the type of BRDF. For anisotropic ones it is defined by all possible θ^{in} , θ^{out} , ϕ^{in} , ϕ^{out} combinations. In contrast, for isotropic BRDFs the exact ϕ -angles are not needed since only their difference is enough. Thus, for isotropic BRDFs is defined by all possible combinations of θ^{in} , θ^{out} and $|\phi^{in} - \phi^{out}|$. In both cases, a binning of the BRDF data is performed with a resolution of 1 degree. For computing the coverage, the count of bins that contain measured data is divided by the total amount of bins. For isotropic BRDFs the resulting space coverage is 0.98% while for anisotropic BRDF it is 0.1%.

Furthermore, figure 7.2 shows a plot of θ^{in} and θ^{out} for all measured points. The depicted material is measured with eight lamps. Due to the curved surface of the cylinder, the angles vary for each point. Still, their sum is always the angle between lamp and camera. This is because the normal either moves towards the camera or the lamp direction while moving along the cylinder surface. Simultaneously, one of the θ -angles decreases while the other increases by the same amount. Thus, all points that are measured with the same lamp lie on a line. More specifically, since eight lamps are used during measuring of this material, the plot also shows eight lines. Note that this principle applies only if the normal lies between camera and light vectors. Otherwise, the sum would start increasing at some point. In our case, this does not happen because these points are not considered in the measurements. Furthermore, the disjoint data points per lamp demonstrate the advantage of the cylinder. The measured data points would overlap for a planar sample which leads to redundant measurements. Thus, the function space coverage would be smaller.

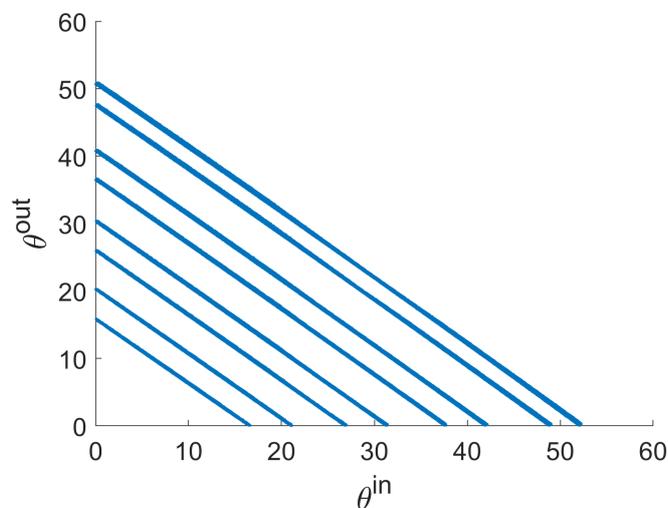


Figure 7.2: *BRDF subspace of θ -angles for measured BRDF points.*

7.1.4 Advantages and Limitations

The BRDF measuring setup can compute isotropic and anisotropic BRDFs of any bendable material. The capturing process requires only one click and is thus unsupervised. It is low-cost and easy to reproduce. Furthermore, the required amount of space is rather short in comparison to most of the other available setups.

Still, the system has some limitations. The biggest one is the limited amount of measurable materials. Since a cylindrical setup is used, the materials have to be wrapped around the cylinder and thus should be bendable. Furthermore, the proposed lighting setup only allows measuring of a limited range for different angles. Extending the angles of the light source would increase the BRDF space coverage.

7.2 BRDF Reproduction

In the following, the reproduction pipeline consisting of model and fabrication method is evaluated. First, the model is validated regarding its functionality. The gamut of the system is shown together with two reproduced materials. Furthermore, the anisotropy is evaluated separately. In the end, the advantages and limitations are resumed.

7.2.1 Model Validation

For evaluating the model, I use cross-validation. It is performed by taking out one of the samples. The model is then optimized to the leftover samples and used to estimate the nominal coverage for the left-out sample. The distance between the estimated and real nominal is the error for this particular iteration. This procedure is applied to each sample once except for the two extremes, i.e., 0/6 and 6/6. In the end, the cross-validation error is computed by averaging the errors of all iterations.

Method #	Used Samples	Preprocessing	Error Mean	Error Std
1	All but 4/6	Clipping	0.049	0.04
2	All	Clipping	0.074	0.086
3	All but 4/6	None	0.142	0.153

Table 7.1: *Cross-validation errors for different modeling methods.*

Table 7.1 shows the mean error and standard deviation of the cross-validation iterations for different modeling methods. Here, method one which excludes the sample 4/6 and smooths out the peak reflection by clipping performs best. The fitted ETN function for the second base is depicted in figure 7.3a. As expected the fitted effective coverages are monotonically increasing for this base. Furthermore, the assumption that with an increasing amount of scratches the anisotropy and diffuseness increase is also fulfilled. Renderings of the fitted sample BRDFs are depicted in figure 7.4.

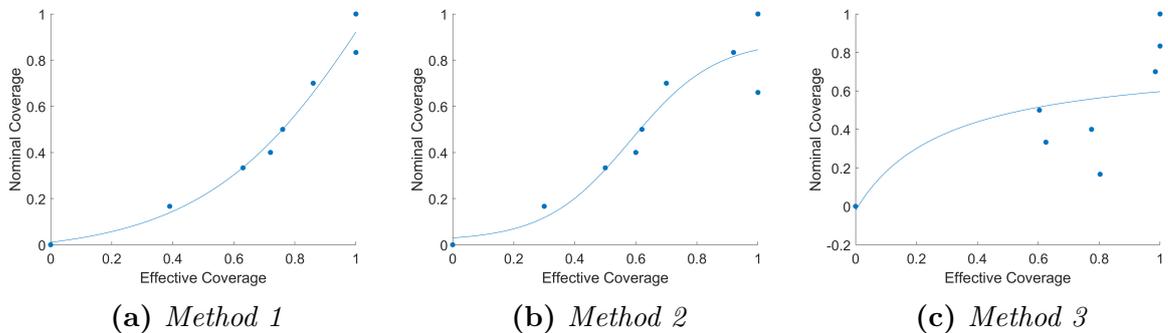


Figure 7.3: *Fitted ETN curves for second base.*

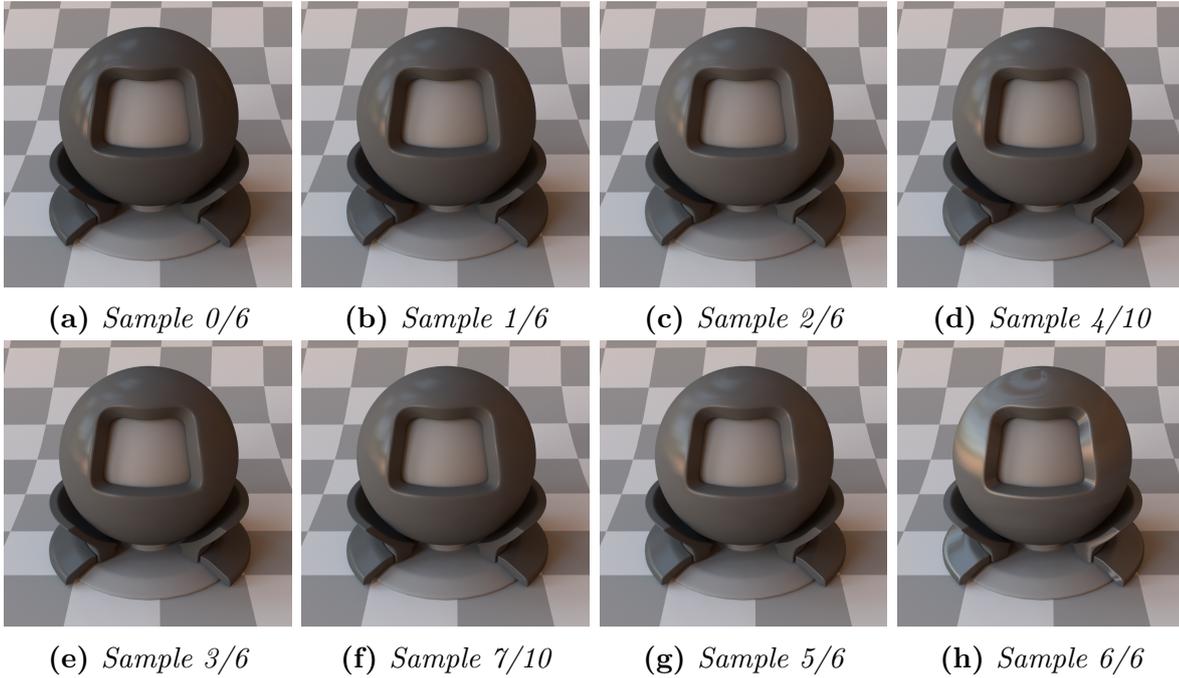


Figure 7.4: Rendered images of the fitted sample BRDFs. The images are rendered with the Mitsuba renderer [25].

In contrast, including the sample 4/6 as in method 2 increases the error and especially the standard deviation since the nominal prediction for this particular sample fails and generates a huge error. Furthermore, the sample 4/6 is included during the validation of other samples and thus negatively affects the computation of the second base. The corresponding ETN curve is depicted in figure 7.3b. An explanation for the negative effect of sample 4/6 is given later in this chapter.

Method 3 does not clip the input data which increases the error significantly. The ETN curve from figure 7.3c shows that the model is not able to find a proper fitting for the effective coverages. They are rather randomly distributed and have no visible trend. Thus, clipping is necessary for the model to be able to generalize and make correct predictions.

To further prove the generalization of the model the fitted second base should be roughly the same during the cross-validation. A fluctuation of the ward parameters means that taking one sample out affects the computation of the base. Thus, the base is too dependent on specific single samples which indicates overfitting to them. The ward parameters of the second base do not change significantly and remain nearly constant. The list with the parameters can be found in the appendix.

7.2.2 Gamut Exploration

For showing the extremes of the gamut an image of increasing nominal coverages is fabricated using the halftoning method. More precisely, the values start on the left side at one and uniformly decrease to the coverage of zero. Note that before halftoning the image the ETN function has to be applied to the values to preserve the relations of the gray tones. The resulting sample is depicted in figure 7.5. Both images show the same sample captured with identical camera parameters. However, they are illuminated from different directions to show the anisotropic effect. Although the gradient is clearly visible, the transition is not smooth at some points. This comes from the fact that the anisotropic effect is stronger the more connected lines an image contains. As the halftoning algorithm fills the screens pixel-wise and not by full lines, the anisotropy increases radically as soon as a line is suddenly filled.

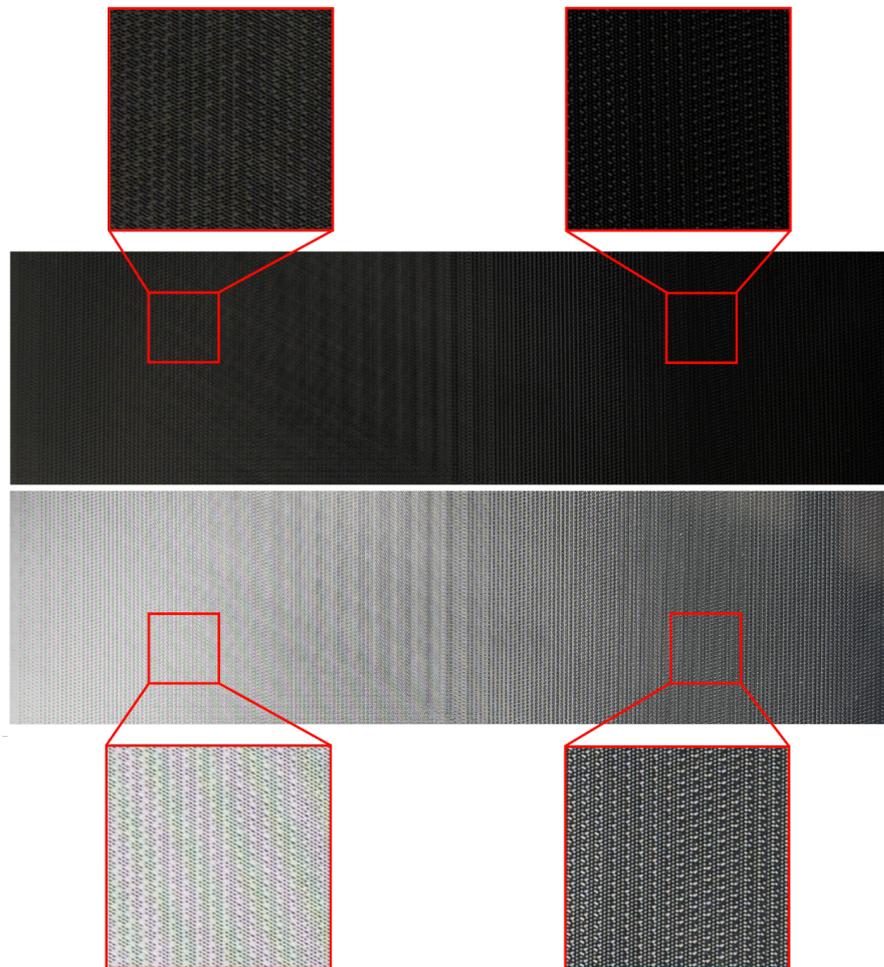


Figure 7.5: *Fabricated gradient sample illuminated from different directions for demonstrating gamut extremes.*

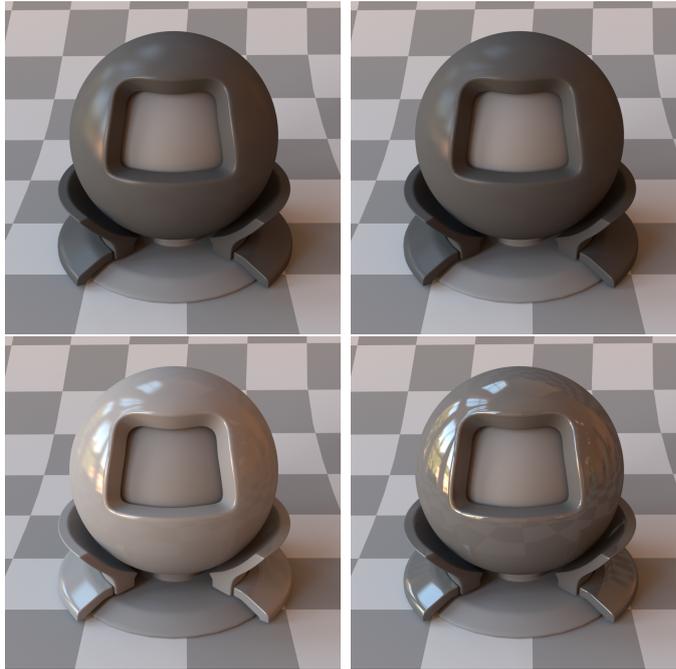


Figure 7.6: *Rendered images of original(left) and reproduced materials(right). The upper row shows the sanded transparency sheet while the lower one depicts the brushed varnish. The images are rendered with the Mitsuba renderer [25].*

7.2.3 Sample Reproduction

The goal of the model is to reproduce materials. The cross-validation tested already the ability of the model to predict the correct nominal value for materials fabricated with the scratching method. Additionally, it is necessary to show that it can predict the nominal for other materials. This is tested by measuring materials that are fabricated with other techniques. The measured data is reproduced using the model and fabrication method. In the end, the fabricated samples are measured and compared against the original one.

Two materials are created for this purpose. The first one is a transparency sheet processed with sandpaper. It is sanded in one direction to get an anisotropic effect. The second material is created by applying a gloss varnish to the transparency sheet. Furthermore, lines are introduced by sweeping with a hard brush over the material while the varnish is drying. Rendering for both materials and their reproductions are depicted in figure 7.6.

As we can see, the first sample is properly reproduced. In contrast, the second one is similar but differs slightly regarding its gloss. That is because it lies outside the gamut of the fabrication method.

7.2.4 Processing Pipeline

As mentioned in section 1, we want a model that is able to reproduce a scene with objects that have different reflectance properties. This is demonstrated by fabricating the scene from figure 7.7a. For each of the visible objects, the specific Ward parameters that were used for rendering is first processed through the model to get the nominal values for each pixel. The resulting image of nominal values is depicted in figure 7.7b. Note that the left sphere has the lowest nominal value since it is specular and should consequently contain a little number of scratches. In contrast, the walls are very diffuse and thus have the highest nominal value. Both the cube and the other sphere are in between. After, the image is fabricated using the halftoning method as explained in chapter 5. Figure 7.7c and 7.7d show the resulting sample illuminated from different directions but under the same conditions. We can observe that the objects have different gloss properties. Furthermore, the relative distances regarding appearance were preserved.

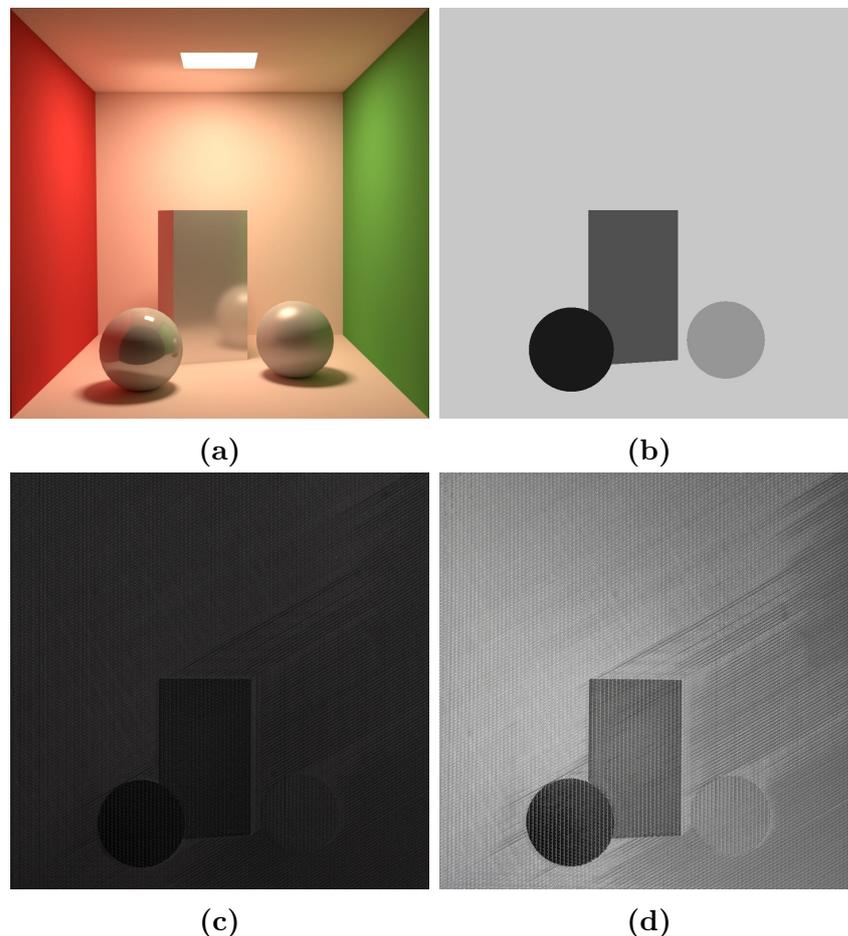


Figure 7.7: Rendered image (a), the corresponding nominals (b) and the fabricated sample (c,d).

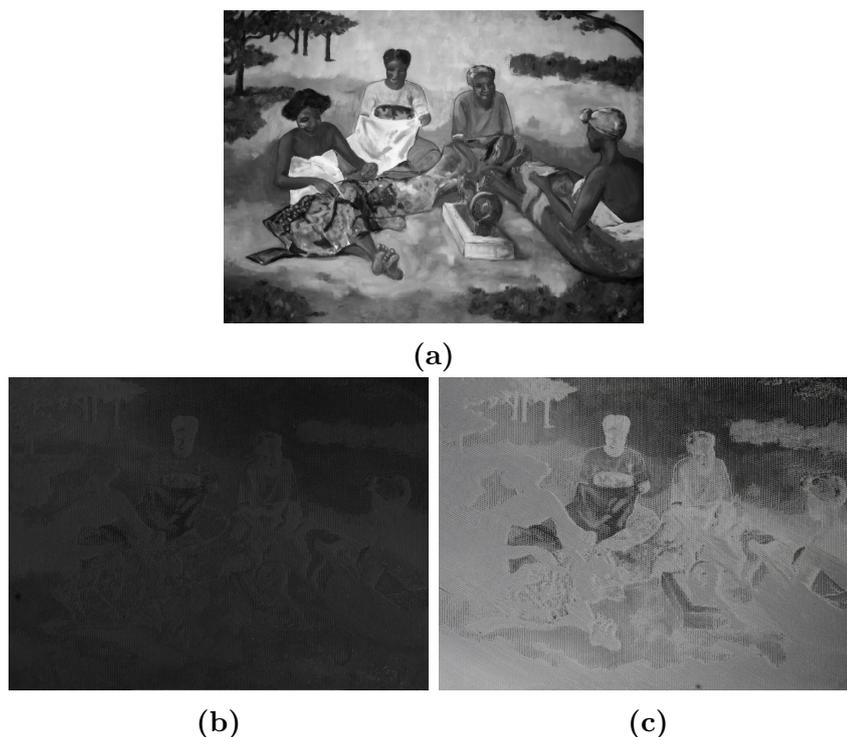


Figure 7.8: *Original continuous-tone image drawn by Paul Gauguin (a) and reproduced sample (b, c).*

7.2.5 Level of details

An image with many different gray values is fabricated to show the possible details with the proposed fabrication method. The grayscale image is depicted in figure 7.8a. It contains many different gray levels which express the nominal coverages. Thus, each gray level is fabricated as a specific amount of scratches and should also demonstrate the possible levels of details.

Before it can be fabricated, we first have to apply the ETN function to each pixel value. This ensures that the relations between the gray tones are preserved during fabrication. After, the resulting image is fabricated using the halftoning method as explained in chapter 5.

Figure 7.8b and 7.8c show the same fabricated sample illuminated from different directions but under the same conditions. Furthermore, the capturing parameters are kept the same to make the results comparable. We can see that depending on the position of the light source the reflection intensities vary strongly. Furthermore, the details of the original image are perceptible and the scene can be recognized.

7.2.6 Anisotropic effect

The anisotropic effect can already be observed in figures 7.5 and 7.7. Nevertheless, I want to present images of the fabricated samples to demonstrate the anisotropy further. Figure 7.9 shows sequence images of the material 6/6 at different rotations. More specifically, it is rotated by 90 degrees between the images. This sample is chosen since it has the strongest anisotropic effect. We can clearly see that the images are differently bright. Especially, the peripheral areas show different reflections. Furthermore, the reflections are differently tilted which happens due to the anisotropy.

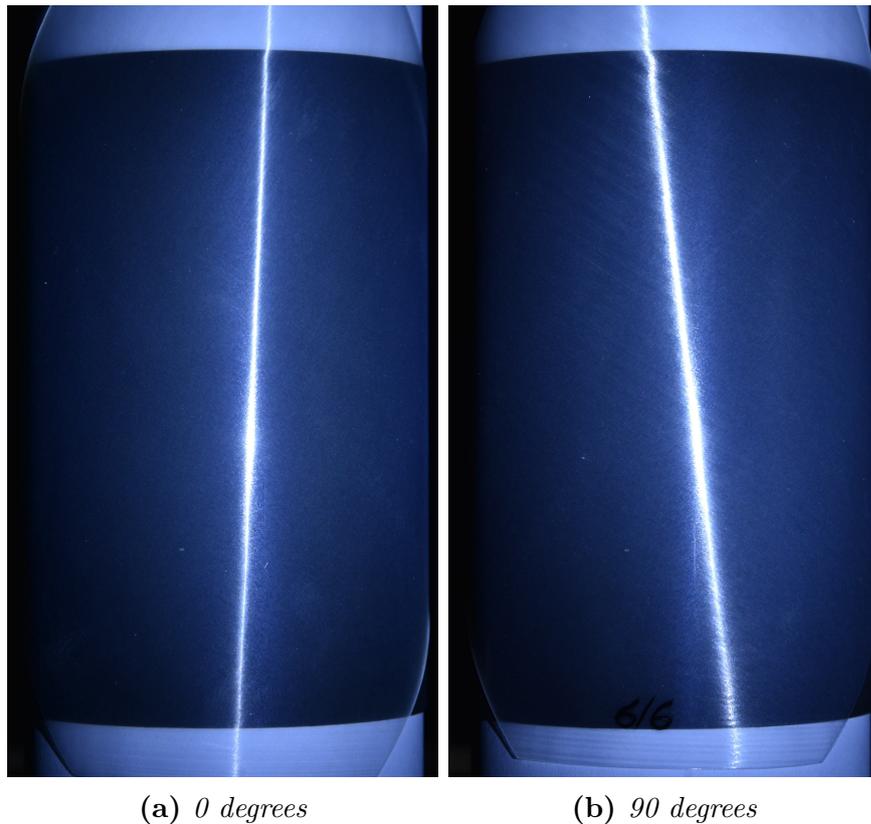


Figure 7.9: Sequence images of sample 6/6 at different rotations.

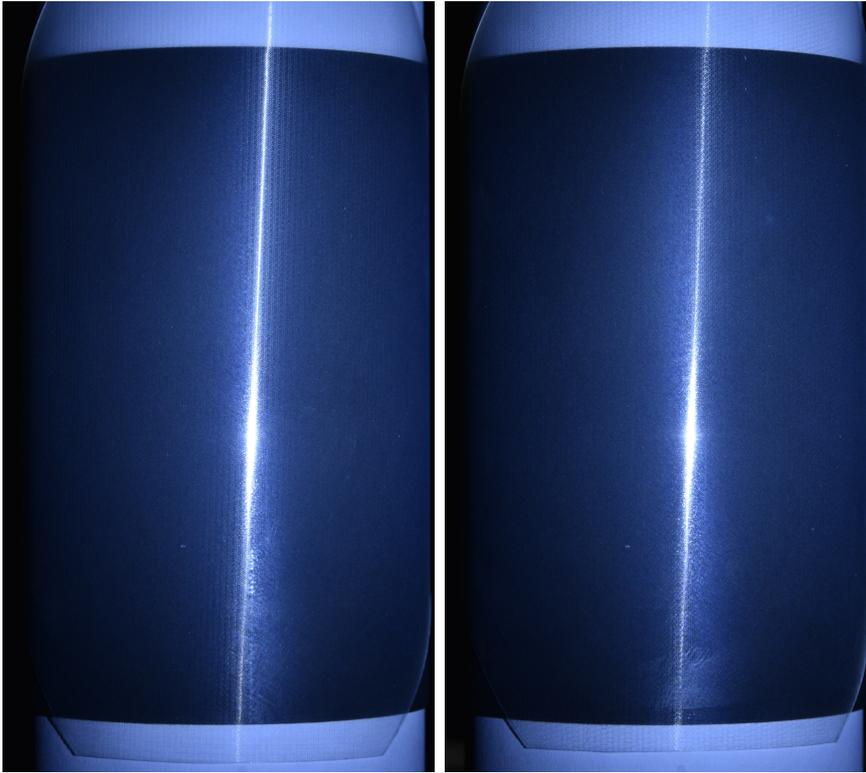


Figure 7.10: Comparison of sequence images from samples 4/6 (left) and 5/6 (right) at 30 seconds exposure time illuminated by lamp at 15 degrees.

7.2.7 Fabrication Errors

During the fabrication of the presented samples, I experienced irregularly emerging fabrication errors. More specifically, even though the height of the scalpel is set properly, the scratching depth within one sample seems to vary. The effect can be observed in the sample from figure 7.7d. Note that it is stronger visible in areas with higher line coverage. Furthermore, slight errors are also present in the sample from figure 7.8c. Again, it is only visible in strongly scratched areas. The effect cannot be observed in figure 7.5 since it does not contain a fully scratched area which is big enough to see it.

The reason for the errors seems to be irregularities in the scratching depth which is caused by unevenness of one component. An uneven bed of the engraving machine can be ruled out. The transparency sheet is put on an aluminum plate which is leveled by milling. Thus, it can be assumed to be perfectly flat. It seems like a varying thickness of transparency sheet is the problem. The experiments for investigating a proper scratching depth have shown that even ten μm more or less can make a big difference. It is possible that the transparency sheet thickness varies only by a few μm which is already enough to change the expected fabrication results.

The errors are only visible in the stronger scratched areas. Scratching deeper than expected in such regions is problematical as it generates thicker lines. The line distance is chosen such that the lines at the predefined depth are barely touching. Now that the lines get thicker they start overlapping. If the thickness additionally fluctuates the effect is emphasized since in some areas the lines are touching while in others they are much thinner and thus covering even less area.

I expect that the sample 4/6 also exhibits the irregularities due to fabrication errors which is why it does not fit into the model. Unfortunately, this sample does not have enough coverage to see the effect by eye. However, by comparing the sequence images of 4/6 and 5/6 we can clearly see that against the assumption 4/6 is brighter than 5/6. The images are depicted in figure 7.10. This also confirms the prediction of the model that the Ward parameters of 4/6 are rather close to 6/6 than to 5/6.

7.2.8 Advantages and Limitations

The presented system can reproduce anisotropic BRDFs of given materials. The method of scratching is a novel approach that has not been used for this purpose before. Furthermore, it is cheap since it does not require expensive materials like inks.

However, the system is limited to the available gamut. The materials from chapter 7.2.3 are reproduced properly because they are both close to the gamut of the fabrication method. Additionally, they are both anisotropic. The system is not able to reliably reproduce isotropic materials. Also, it is hard to reproduce highly glossy materials since the peak is smoothed out during the prediction. Thus, the peak cannot be fully considered anymore.

Furthermore, the fabrication method uses only one variable parameter, i.e., the nominal coverage. It has to encode both the desired amount of anisotropy and diffuseness. Changing anisotropy without simultaneously changing the diffuseness as well is not possible. It always has to be a trade-off between both.

Fully covering an area with scratches is also problematically due to the stated fabrication error. It leads to unpredictable behaviour and thus generates unreliable results. However, this problem could be solved by using transparency sheets of higher quality.

8 Conclusion

The thesis presented a novel approach for editing and reproducing surface appearances of flat objects. It combines the principles of BRDF measuring and appearance fabrication to build a model that can estimate and fabricate the reflectance properties of a given material.

The data-driven model is based on BRDF measurements of fabricated samples. For measuring and estimating the BRDF of surfaces a measuring system is built. More precisely, the material is bent around a cylinder while a camera captures the reflected amount of light. Multiple LED light sources at different positions are used to illuminate the material. The measuring system can estimate the anisotropic BRDF of any bendable material. Furthermore, using the detailed explanation given in this thesis everybody can reproduce the system due to its simple structure and affordable components.

For the fabrication method of the samples, multiple different methods were investigated. Among drawing with varnish and laser marking, the method of scratching surfaces with a scalpel provided the best results. More specifically, it is possible to make a substrate appear more matt and increase the anisotropic reflection by scratching one-dimensional lines into a glossy surface.

The final model is based on measurements of multiple samples fabricated by scratching. It expresses the relation between varyingly strong scratched surfaces and can estimate the needed amount of scratches for simulating the surface reflection properties of a given material. The evaluation has shown that materials inside the gamut of the model can be reliably reproduced.

9 Future Work

In this chapter, I present further possible improvements to the method. First, I explain how the captured BRDF space can be expanded. Additionally, some alternative fabrication methods are shown that provide several advantages.

9.1 BRDF Space Expansion

Compared to a planar setup, using a cylinder for reflectance measurements of materials already increases the captured part of the BRDF space. Still, the BRDF subspace can be expanded by using a spherical setup. It provides more different angles in both directions due to the curvature. However, it is harder to wrap a material around a sphere. A similar approach as proposed by Konakovic et al. [26] could be used for this purpose. They used cuts to make inextensible materials stretchable and thus bendable around doubly-curved surfaces like spheres.

Still, experiments have shown that the current setup only captures around 0.1 % of the anisotropic BRDF space. A bigger space would improve the fitting to the BRDF function. Havran et al. [27] presented an optimization-based approach for finding the best coverage of incoming and outgoing light directions to preserve important features of anisotropic BRDFs. A similar approach could be used to optimize for the best light source positions regarding the fitting of anisotropic materials.

9.2 Alternative Fabrication Methods

At the moment, the direction of lines is limited. By being able to rotate the scalpel one could make multi-directional scratches and thus expand the possible directions of anisotropy. For this purpose, the holder design is extended such that the scalpel can be rotated by hand in discrete steps. Consequently, samples would be fabricated layer-wise since the scalpel has to be rotated manually between two different scratching directions. Alternatively, one could extend the holder by a motor which rotates it in continuous steps. This solution would offer the possibility to change the scratching direction seamlessly, and thus it could generate continuous curves.

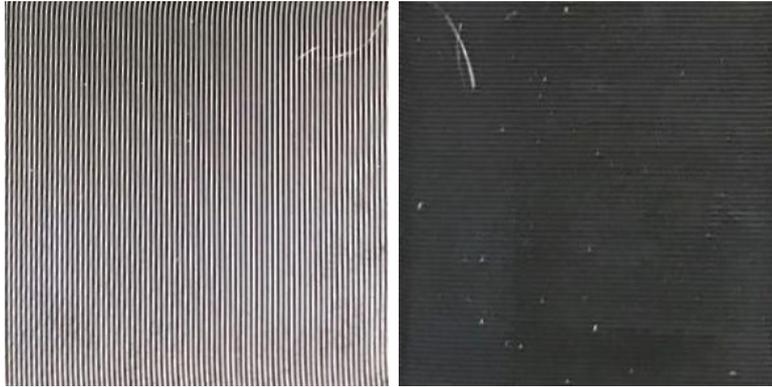


Figure 9.1: *The anisotropic effect produced by an FDM printer. The images depict the same sample identically illuminated but rotated by 90 degrees.*

With the proposed fabrication method the choice of substrates is limited to flat surfaces. However, the presented engraving machine can scratch heightfields due to its z-axis. In that case, a proper 2.5D object as a substrate is necessary. Furthermore, the model has to be refitted for this particular material. Fabricating fully 3D objects is rather hard with the engraver as the object has to be rotated at some point. Here, registration is problematical. Alternatively, a 5 DoF milling machine would make it simpler to extend the presented approach to 3D. The problem with such devices is that the scalpel has to be always aligned to its scratching direction. Thus, the machine needs to rotate the fabrication tool additionally.

Another approach is using FDM printers. Experiments have shown that they generate anisotropic surface reflectances due to layer-wise printing. The anisotropic effect of an FDM print is shown in figure 9.1. The appearance of the lines in FDM prints is usually not desired and reduce the aesthetics of the printed objects. Controlling the lines and additionally use them for changing the reflectance properties of an object would increase the quality of FDM prints.

Bibliography

- [1] Dave Edwards, Solomon Boulos, Jared Johnson, Peter Shirley, Michael Ashikhmin, Michael Stark, and Chris Wyman. The halfway vector disk for brdf modeling. In *ACM Transactions on Graphics*, 2006.
- [2] Gregory J Ward. Measuring and modeling anisotropic reflection. In *ACM SIGGRAPH Computer Graphics*, 1992.
- [3] Bruce Walter. Notes on the ward brdf. In *Program of Computer Graphics, Cornell University, Technical report PCG-05*, 2005.
- [4] Erik Reinhard, Wolfgang Heidrich, Paul Debevec, Sumanta Pattanaik, Greg Ward, and Karol Myszkowski. High dynamic range imaging: acquisition, display, and image-based lighting. In *Morgan Kaufmann*, 2010.
- [5] Konrad F Karner, Heinz Mayer, and Michael Gervautz. An image based measurement system for anisotropic reflection. In *Computer Graphics Forum*, 1996.
- [6] Stephen R Marschner, Stephen H Westin, Eric PF Lafortune, Kenneth E Torrance, and Donald P Greenberg. Image-based BRDF measurement including human skin. In *Rendering Techniques' 99: Proceedings of the Eurographics Workshop*, 1999.
- [7] Stephen Robert Marschner and Donald P Greenberg. Inverse rendering for computer graphics. In *Cornell University*, 1998.
- [8] Vlastimil Havran, Jan Hošek, Šárka Němcová, Jiří Čáp, and Jiří Bittner. Light-drum—portable light stage for accurate btf measurement on site. In *Multidisciplinary Digital Publishing Institute*, 2017.
- [9] Robert L Cook and Kenneth E Torrance. A reflectance model for computer graphics. In *ACM Transactions on Graphics*, 1982.
- [10] James F Blinn. Models of light reflection for computer synthesized pictures. In *ACM SIGGRAPH computer graphics*, 1977.
- [11] Tim Weyrich, Pieter Peers, Wojciech Matusik, and Szymon Rusinkiewicz. Fabricating microgeometry for custom surface reflectance. In *ACM Transactions on Graphics*, 2009.
- [12] Olivier Rouiller, Bernd Bickel, Jan Kautz, Wojciech Matusik, and Marc Alexa. 3d-printing spatially varying brdfs. In *IEEE computer graphics and applications*, 2013.
- [13] Anat Levin, Daniel Glasner, Ying Xiong, Frédo Durand, William Freeman, Wojciech Matusik, and Todd Zickler. Fabricating brdfs at high spatial resolution using wave optics. In *ACM Transactions on Graphics*, 2013.

-
- [14] Stefan Sinzinger, Jürgen Jahns, and Jürgen Jahns. Microoptics. In *Wiley Online Library*, 2003.
- [15] Wojciech Matusik, Boris Ajdin, Jinwei Gu, Jason Lawrence, Hendrik Lensch, Fabio Pellacini, and Szymon Rusinkiewicz. Printing spatially-varying reflectance. In *ACM Transactions on Graphics*, 2009.
- [16] Yanxiang Lan, Yue Dong, Fabio Pellacini, and Xin Tong. Bi-scale appearance fabrication. In *ACM Transactions on Graphics*, 2013.
- [17] Tom Malzbender, Ramin Samadani, Steven Scher, Adam Crume, Douglas Dunn, and James Davis. Printing reflectance functions. In *ACM Transactions on Graphics*, 2012.
- [18] Satoshi Suzuki and Keiichi Abe. Topological structural analysis of digitized binary images by border following. In *Computer vision, graphics, and image processing*, 1985.
- [19] Christopher G Harris, Mike Stephens, et al. A combined corner and edge detector. In *Alvey vision conference*, 1988.
- [20] Xiao-Shan Gao, Xiao-Rong Hou, Jianliang Tang, and Hang-Fei Cheng. Complete solution classification for the perspective-three-point problem. In *IEEE transactions on pattern analysis and machine intelligence*, 2003.
- [21] Matt Pharr, Wenzel Jakob, and Greg Humphreys. Physically based rendering: From theory to implementation. In *Morgan Kaufmann*, 2016.
- [22] Christian Regg, Szymon Rusinkiewicz, Wojciech Matusik, and Markus Gross. Computational highlight holography. In *ACM Transactions on Graphics*, 2010.
- [23] Vahid Babaei and Roger D Hersch. Juxtaposed color halftoning relying on discrete lines. In *IEEE Transactions on Image Processing*, 2013.
- [24] Vahid Babaei and Roger D Hersch. Spectral prediction of juxtaposed halftones relying on the two-by-two dot centering model. In *Journal of Imaging Science and Technology*, 2013.
- [25] Jakob Wenzel. Mitsuba renderer. 2010.
- [26] Mina Konaković, Keenan Crane, Bailin Deng, Sofien Bouaziz, Daniel Piker, and Mark Pauly. Beyond developable: computational design and fabrication with auxetic materials. In *ACM Transactions on Graphics*, 2016.
- [27] Vlastimil Havran, Jiri Filip, and Karol Myszkowski. Perceptually motivated brdf comparison using single image. In *Computer Graphics Forum*, 2016.

Appendices

Ward parameters

Table 9.1 depicts the fitted Ward parameters for fabricated samples, computed bases, reproduced materials and example materials. Furthermore, table 9.2 shows the different fitted Ward parameter sets during cross-validation.

Name	ϕ^d	ϕ^s	α^x	α^y	\angle^l
Sample 0/6	0.374	0.0064	0.035	0.0492	2.52
Sample 1/6	0.044	0.006	0.0479	0.0352	2.32
Sample 2/6	0.0518	0.0069	0.0595	0.034	2.47
Sample 4/10	0.051	0.0075	0.0679	0.0335	2.49
Sample 3/6	0.0476	0.0069	0.0684	0.029	2.53
Sample 4/6	0.0513	0.0218	0.213	0.0341	2.58
Sample 7/10	0.0492	0.0083	0.0835	0.0311	2.56
Sample 5/6	0.0505	0.0133	0.1363	0.0327	2.60
Sample 6/6	0.0474	0.0477	0.4572	0.0314	2.58
Base 1	0.0374	0.0064	0.035	0.0492	2.52
Base 2	0.052	0.01	0.103	0.03	2.55
Transparency Sheet	0.1198	0.0708	0.0031	0.006	3.26
White paper	0.3183	0.00001	0.0.3753	0.3580	2.53
Polyethylen	0.0921	0.0303	0.0436	0.0378	2.87
Sanded original	0.0523	0.0153	0.0755	0.0652	2.25
Sanded reproduced	0.0492	0.0083	0.0835	0.0311	2.19
Brushed original	0.173	0.045	0.0354	0.0106	2.19
Brushed reproduced	0.0751	0.0583	0.0089	0.0079	2.51

Table 9.1: *Fitted Ward parameters of materials used in the thesis.*

Excluded Sample	ϕ^d	ϕ^s	α^x	α^y	\angle^l
1/6	0.052	0.011	0.113	0.030	2.57
2/6	0.051	0.011	0.120	0.030	2.57
4/10	0.052	0.012	0.125	0.030	2.58
3/6	0.052	0.012	0.129	0.031	2.58
7/10	0.052	0.013	0.137	0.030	2.58
5/6	0.051	0.008	0.110	0.030	2.54

Table 9.2: *Fitted Ward parameters of second base during cross-validation.*