Instrumentation for Estimating Surface Radiometry

Doest, Mads Emil Brix

Publication date:
2021

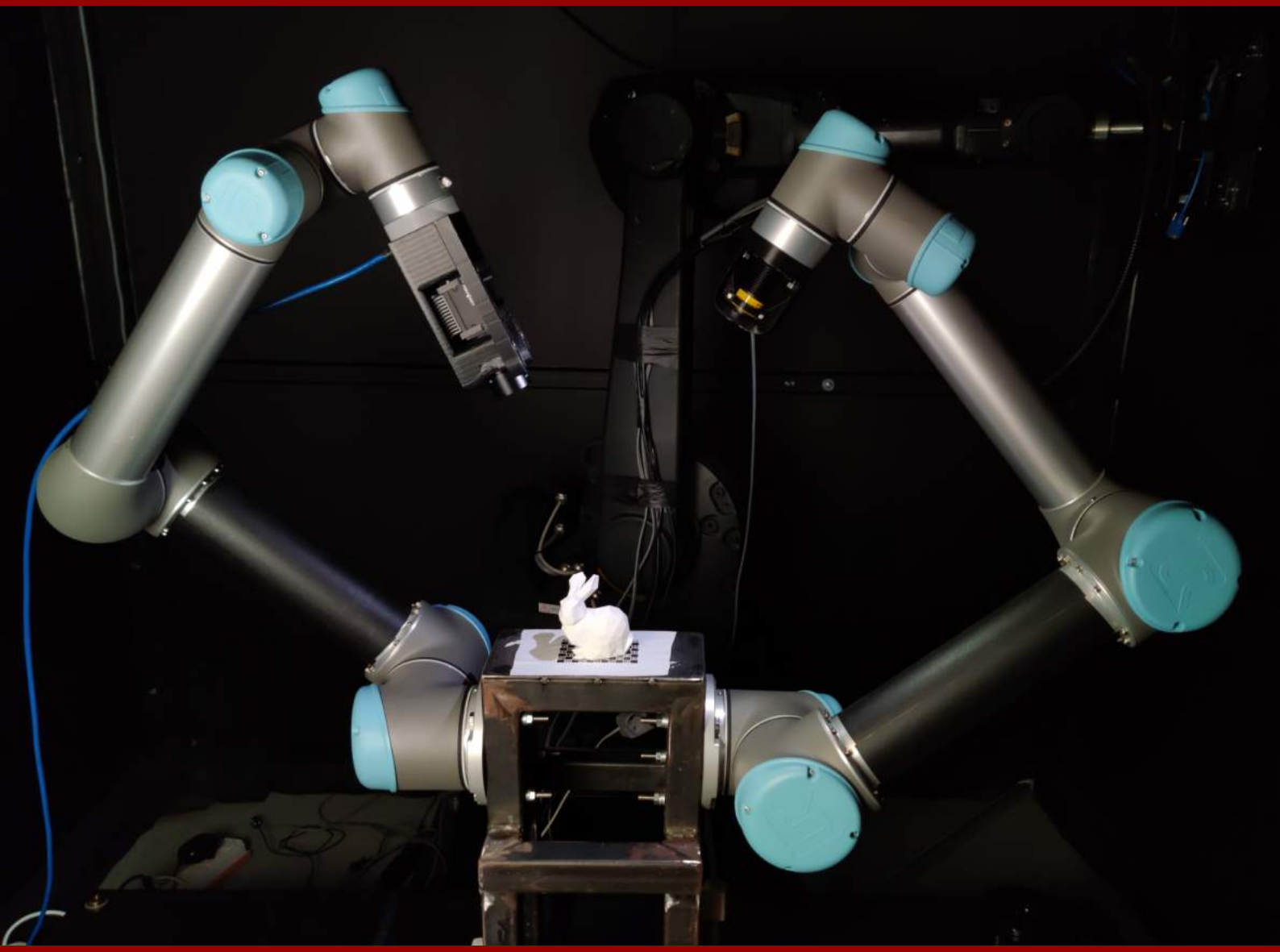
Document Version
Publisher's PDF, also known as Version of record

DTU

MADE
Manufacturing Academy of Denmark

Instrumentation for Estimating Surface Radiometry

Ph.D. Thesis



DTU Compute
Department of Applied Mathematics and Computer Science

Cover photo: Mads Doest, 2020
Published by: DTU, Department of Applied Mathematics and Computer Science, Richard Petersens Plads, Bygning 324, 2800 Kgs. Lyngby Denmark
www.compute.dtu.dk
ISSN: [0000-0000] (electronic version)
ISBN: [000-00-0000-000-0] (electronic version)

ISSN: [0000-0000] (printed version)
ISBN: [000-00-0000-000-0] (printed version)

Summary (English)

Automating visual quality assurance of product appearance is a very hot topic in industry as it is still primarily a manual task only carried out by humans. With a strong focus on digitization in *Industry 4.0*, it is also highly beneficial for companies to be able to include product appearance in their *digital twin*. In order to automate the visual quality assurance, we need to be able to model, measure, and compare appearance.

Our focus is combining computer vision and computer graphics to develop the instrumentation and methodology needed to achieve the goal of automating appearance assessment. This thesis is focused around industrial applications in collaboration with industrial partners through an organization named Manufacturing Academy of Denmark (MADE).

Specifically, we contribute by developing three robotic setups that utilize cameras and controllable light sources to obtain surface information from radiometric measurements, *ABB Setup*, *UR5 Setup* and *UR3 Setup*. These instruments has been key components in the contributions presented in this thesis, although they span a wide range of topics. One contribution is developing methods for measuring the contrast in engineered microsurfaces with the *UR5 Setup*. In relation to that, we developed a suitable BRDF model for simulating these structures. By utilizing the high repeatability of the *ABB Setup*, we created a digitization pipeline that inputs data from multiple modalities and create a digital scene, which can be rendered and compared pixel by pixel with a photograph. This allow for direct comparison of the digital and physical twin as images. We developed practical methods that can help introduce appearance in the digital twin such that it can be compared with the physical twin for Visual Quality Assurance.

Summary (Danish)

Automatisering af visual kvalitetets kontrol af product udseende er et meget efterspurgt emne i industrien, da det stadig primært er en manuel opgave udført af mennesker. Med et stærkt fokus på digitaliseringen i Industri 4.0, er det meget fordelagtigt for virksomheder at kunne inkludere et products udseende i dets digitale tvilling. For at kunne automatisere visual kvalitets kontrol, så er man nødt til at kunne modelere, måle og sammenligne udseende.

Vores fokus er at kombinere computer vision og computer grafik til at udvikle måleinstrumenter og metoder til at opnå automatiseret vurdering af et produkts udseende. Denne afhandling bygger på industrielle anvendelser i samarbejde med industrielle samarbejdspartnere gennem organisationen Manufacturing Academy of Denmark (MADE).

Specifikt, så bidrager vi ved at udvikle tre robotsystemer *ABB System*, *UR5 System*, og *UR3 System*, som bruger kameraer og kontrollerede lyskilder til at få informationer om overfladen ud fra radiometriske målinger. Disse instrumenter har været nøglekomponenter i bidragene som præsenteres i denne afhandling, selvom anvendelsesområderne spænder bredt. En del af bidragene er udviklingen af metoder, til at modelere og måle kontrasten i mikro fræsede overflader med vores *UR5 setup*. Ved at udnytte den høje repetiterbarhed i vores *ABB setup*, kunne vi udvikle en digitaliserings pipeline, der tager imod data fra forskellige modaliteter og udformer en digital scene.

Preface

This thesis was carried out at the Section for Visual Computing, under the department of Applied Mathematics and Computer Science (DTU Compute), at the Technical University of Denmark (DTU). It is done in fulfilment of the requirements for obtaining a Doctor of Philosophy degree (Ph.D) in Computer Science.

This project has been funded by Manufacturing Academy of Denmark (MADE) as part of Work Package 9 - Sensor Technologies and Production Data.

In this thesis we seek to combine robotics, computer vision, and computer graphics, to develop tools that allow for estimation of appearance, i.e. surface radiometry. To improve on the tools used in Industry 4.0 to introduce object appearance into the digital twin.

The project has been supervised by Associate Professor Jeppe Revall Frisvad as main supervisor, Associate Professor Claus Brøndsgaard Madsen as co-supervisor and Assistant Professor Søren Schou Gregersen as co-supervisor. The research has been conducted at the Section for Visual Computing at DTU Compute. Part of the experimental work has been done in collaboration with the Section of Manufacturing Engineering at the Department of Mechanical Engineering at the Technical University of Denmark. The external stay was conducted under the supervision of Alessio Del Bue and Arianna Traviglia at Istituto Italiano di Tecnologia - IIT Centre for Cultural Heritage Technology (CCHT@Ca'Foscari) in Venezia Italy.

Mads Doest

Acknowledgements

First I would like to thank my supervisor, Jeppe Revall Frisvad, who has been a great inspiration and motivator throughout my PhD studies. His excellent guidance and highly technical knowledge led to fruitful discussions of interesting topics. I would like to thank my supervisors, Jeppe Revall Frisvad, Søren K. S. Gregersen, and Claus Brøndsdgaard Madsen, for their guidance and our many interesting and fruitful discussions.

Thanks to MADE and DTU Compute for providing me with the funding to work with my passion for 3 years and MADE collaborators for feedback and discussions.

Thanks to LEGO and especially Otto H. A. Abildgaard for great collaboration, interesting discussions, and support.

Thanks to all my collaborators from IIT Center for Cultural Heritage Technology, especially, Alessio Del Bue, Stuart James, Arianna Traviglia, and Marco Fiorucci.

Thanks to my collaborators from DTU Mekanik, especially, Francesco Regi, Dongya Li, Macarena Ribo, Dario Loaldi and Yang Zhang.

A special thanks to all my colleagues and friends at DTU Compute and the Section for Visual Computing. It's been some interesting and exciting years with you all.

Lastly I will thank my friends, family and girlfriend Thea for their immense support.

List of Contributions

Peer Reviewed Contributions

A Jonathan Dyssel Stets, Alessandro Dal Corso, Jannik Boll Nielsen, Rasmus Ahrenkiel Lyngby, Sebastian Hoppe Nesgaard Jensen, Jakob Wilm, **Mads Emil Brix Doest**, Carsten Gundlach, Eythor Runar Eiriksson, Knut Conradsen, Anders Bjorholm Dahl, Jakob Andreas Bærentzen, Jeppe Revall Frisvad, and Henrik Aanæs. "Scene reassembly after multimodal digitization and pipeline evaluation using photorealistic rendering". In: *Applied Optics* 56.27 (Sept. 2017), pp. 7679–7690. DOI: 10.1364/AO.56.007679. [1]

B Andrea Luongo, Viggo Falster, **Mads Emil Brix Doest**, Dongya Li, Francesco Regi, Yang Zhang, Guido Tosello, Jannik Boll Nielsen, Henrik Aanaes, and Jeppe Revall Frisvad. "Modeling the Anisotropic Reflectance of a Surface With Microstructure Engineered to Obtain Visible Contrast After Rotation". In: *Proceedings of the IEEE International Conference on Computer Vision Workshops (ICCVW)*. Oct. 2017, pp. 159–165. DOI: 10.1109/ICCVW.2017.27. [2]

C Francesco Regi, **Mads Emil Brix Doest**, Dario Loaldi, Dongya Li, Jeppe Revall Frisvad, Guido Tosello, and Yang Zhang. "Functionality characterization of injection moulded micro-structured surfaces". In: *Precision Engineering* 60 (Nov. 2019), pp. 594 –601. DOI: 10.1016/j.precisioneng.2019.07.014. [3]

D Andrea Luongo, Viggo Falster, **Mads Emil Brix Doest**, Macarena Méndez Ribó, Eyþór Rúnar Eiríksson, David Bue Pedersen, and Jeppe Revall Frisvad. "Microstructure Control in 3D Printing with Digital Light Processing". In: *Computer Graphics Forum* 39.1 (2020), pp. 347–359. DOI: 10.1111/cgf.13807. [4]

E Morten Hannemose, **Mads Emil Brix Doest**, Andrea Luongo, Søren Kimmer Schou Gregersen, Jakob Wilm, and Jeppe Revall Frisvad. "Alignment of rendered images with photographs for testing appearance models". In: *Applied Optics* 59.31 (Nov. 2020), pp. 9786–9798. DOI: 10.1364/AO.398055. [5]

F Sebastian Hoppe Nesgaard Jensen, **Mads Emil Brix Doest**, Henrik Aanæs, and Alessio Del Bue. "A benchmark and evaluation of non-rigid structure from motion". In: *International Journal of Computer Vision* 129 (Dec. 2020). DOI: 10.1007/s11263-020-01406-y. [6]

Non Peer Reviewed Manuscripts

G **Mads Emil Brix Doest**, Stuart James, Alessio Del Bue, and Jeppe Revall Frisvad. "Reconstructing transparent glass objects from polarization". Unpublished Manuscript. 2021. [7]

Not Included Peer Reviewed Contributions

I Henrik Aanæs, Knut Conradsen, Alessandro Dal Corso, Anders Bjorholm Dahl, Alessio Del Bue, **Mads Emil Brix Doest**, Jeppe Revall Frisvad, Sebastian Hoppe Nesgaard Jensen, Jannik Boll Nielsen, Jonathan Dyssel Stets, and George Vogiatzis. "Our 3D vision data-sets in the making". In: *The Future of Datasets in Vision 2015: CVPR 2015 Workshop*. 2015. [8]

J Jakob Wilm, Daniel González Madruga, Janus Nørtoft Jensen, Søren Kimmer Schou Gregersen, **Mads Emil Brix Doest**, Maria Grazia Guerra, Henrik Aanæs, and Leonardo De Chiffre. "Effects of subsurface scattering on the accuracy of optical 3D measurements using miniature polymer step gauges". In: *Proceedings of the 18th International Conference of the European Society for Precision Engineering and Nanotechnology (euspen 2018)*. June 2018, pp. 449–450. [9]

Not Included Non Peer Reviewed Manuscripts

K Andrea Luongo, Jeppe Revall Frisvad, Alessandro Dal Corso, **Mads Emil Brix Doest**, and Henrik Wann Jensen. "Building Vision-Based Predictive Appearance Models for 3D Printing". Published as a Technical Report in Andrea Luongos PhD Thesis. 2019. [10]

Abbreviations

ADC Analog to Digital Converter

ANOVA ANalysis Of VAriance

BRDF Bi-directional Reflectance Distribution Function

BSDF Bi-directional Scattering Distribution Function

BSSRDF Bi-directional Surface Scattering Distribution Function

BTDF Bi-directional Transmittance Distribution Function

BTF Bi-directional Texture Function

BxDF Bi-directional x Distribution Function

CAD Computer Aided Design

CG Computer Graphics

CGI Computer-Generated Imagery

CNN Convolutional Neural Network

Cobot Collaborative Robot

CT Computed Tomography

DoF Degrees of Freedom

FPS Frames Per Second

GAN General Adversarial Network

GT Ground Truth

HDR High Dynamic Range

HSV Hue Saturation Value

IK Inverse Kinematics

LCD Liquid Crystal Display

LED Light Emitting Diode

LUT Look-Up Table

MoCap Motion Capture

MP Mega Pixels

NRSfM Non-Rigid Structure from Motion

QA Quality Assurance

RGB Red, Green, and Blue

RMSE Root Mean Squared Error

ROS Robot Operating System

SfM Structure from Motion

SLA Stereolithography Apparatus

SVBRDF Spatially Varying Bi-directional Reflectance Distribution Function

VQA Visual Quality Assurance

Contents

CHAPTER 1
Introduction

This thesis has been carried out in close collaboration with industrial partners through an organization named Manufacturing Academy of Denmark (MADE), and as such the research project focuses on applied research for industrial use. The industrial interest in this topic relates to visual quality assurance, more specifically to develop instruments able to quantify if the appearance of a produced part is within specification or not. Coupling this with the parameters used to control the machines and we would achieve, what in Industry 4.0 is referred to as *Closing the Loop*. As such the focus of this research project is developing instrumentation and methods that can be used for visual quality assurance with the goal of *Closing the Loop*. We seek to make the appearance of an object quantifiable, so that others can link it to machine parameters, this is an important and for appearance an unsolved task.

1.1 Scope

In this thesis we focus on developing instrumentation and methods for estimating and comparing the appearance of objects. The work is focused around measuring optical phenomena on a human perceivable scale and from those measurements estimate properties not directly observable. Specifically we use cameras and controlled environments to estimate the radiometric properties, by observing the light reflected off a surface. We seek to use photographs in combination with rendering techniques to infer these properties.

As the primary application of this research project is industrial, we work with materials used in industry, which are primarily metals, glass, and polymers. These materials are known for being difficult to scan using traditional 3D scanners and thus we rely on multiple modalities for acquisition of geometry. One modality being a Computer Aided Design (CAD) model, which is almost always available as part of the design process. Some of the methods in this thesis rely on a priori knowledge of geometry for estimation of material properties, and some include the acquisition of geometry.

The contributions in this thesis span a broad range of topics, from designing and building instruments, to object and light pose estimation, creating datasets for evaluation of methods, material modelling, and virtual scene representation for comparing physical objects with digital versions. But common for all of these contributions are that they

are steps towards being able to *close the loop* in production.

Because of the broad range of topics used in this research project, spanning over robotics, computer vision, computer graphics, and additive manufacturing, we cover the specifics for the development of the instrument, whereas for the other contributions we refer to the attached papers in Contributions A to G for the specifics. We kindly refer the reader to Section 1.5 for an in depth explanation of the expected background when reading the thesis and a list of suggested literature.

1.2 Motivation

The traditional production pipeline, as seen in Figure 1.1, is where a designer designs the product, the design is then given to a machine that produces a product. Then the Visual Quality Assurance (VQA) inspects the product to ensure that the product is of the desired quality. In the case of not passing the VQA, the machine operator needs to fix the machine, so that it can produce products that the VQA can approve. In relation to Visual Quality Assurance, this pipeline is, for many companies, a manual task, which leads to much variation in the products, due to variations in the VQA. In many productions, the visual quality assurance is still largely manual, this is primarily because it is a very complicated task, not easily automated. Since it is a manual task, it is a labour intensive and error prone process. This is one of the reasons why it is interesting to look into developing new tools to aid the process of appearance specification and verification. Tools that can assist the humans in visual quality assurance and increase robustness. The key component that is missing for this is instruments and methods to measure and quantify the appearance of an object.

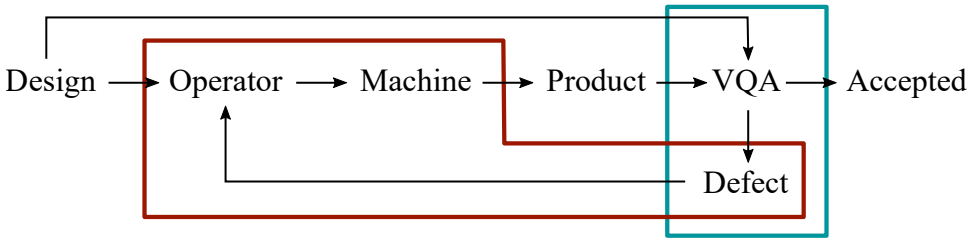


Figure 1.1: Figure showing a simplified overview of the traditional production pipeline. Scope of this project is related to VQA (marked in blue), and Closing the Loop (marked in red).

The field of Computer Graphics, is showing amazing results with their photorealistic renderings, creating digital models that many of us cannot tell are not real. For example companies like KeyShot[1], produce software for artists to create images for virtual product showcasing that look stunningly real. We see movies with digital special effects where it

[1] https://www.keyshot.com/

is nearly impossible to tell what is real and what is Computer-Generated Imagery (CGI), yet we can read that it was "More efficient" to buy a real Boeing 747 airplane and blow up for making a scene in the movie Tenet[2], than using CGI for it. Even though the statement is a publicity stunt, there is at least some truth to it, that creating digital models of real objects is expensive, increasing with requirement for realism. This makes it interesting to introduce the computer graphics concepts of photorealistic rendering into visual quality control, as a way to work with object appearance. But to be useful in practice, we need to develop instruments to acquire the geometry and optical properties.

The simple approach would be to use cameras to measure the appearance models. Often simpler models are used such as a Bi-directional Reflectance Distribution Function (BRDF) but to refer to the full range of models an umbrella term Bi-directional x Distribution Function (BxDF) is often used, where x is the placeholder for any specific model. In general BxDFs are not easily compared with each other, and even comparison between the parameters of two different materials modeled with the same model, being mathematically easy, does not correlate with the perceived difference. E.g. two materials with very different parameters might look quite similar in their perceived appearance, this makes BxDF parameters ill suited for comparing the perceived appearance.

Many appearance models exist [11] there are both physically accurate and approximate models for better artistic use. These models were not developed with the purpose of comparison and material analysis, but for producing the most realistic rendering. Most models have a high variance of the numerical scale of their parameters, and mathematical ambiguities, allowing very different values of parameters to result in the same or very similar output, this is sometimes called similarity relations [12, 13, 14]. This makes distance measures less meaning-full and complicate comparing the perceived output based solely on model parameters, thus such an approach based on current material models is sub optimal for quality assurance.

One of the reasons that we would like to make comparisons in image space rather than BxDF model space is that images make more sense to us humans for interpretation where most BxDF models have hard-to-interpret parameters. Images are the closest representation we can make of the real world, that we can also intuitively interpret. A problem that rises when working with images is that they show a final combination of many interactions, each pixel contains information on both environmental lighting, geometry and material properties.

The focus of this research project has been on developing instrumentation supporting the research in the methodology to compare the appearance of objects. In order to compare something, it first needs to be measured. In order to measure something, you first need to have a model, that explains what to measure and how to measure. Lastly, you need an instrument to perform the measurement. These are the four core problems addressed in this thesis.

[2]https://www.insider.com/christopher-nolan-blew-up-boeing-747-for-tenet-stunt-2020-5

1.3 Research Goals

The primary objective of the MADE Work-Package 9.4 - Estimation of Surface Radiometry, was to develop a proof of concept instrument, able to estimate optical properties of materials in an industrially relevant setting. As mentioned in Section 1.1, an industrially relevant setting is a controlled environment, where the environment, geometry and expectation of the appearance is known a priori.

This industrial interest in being able to compare a product with its digital twin for visual quality assurance, is the core motivation for the research described in this thesis. It is not obvious how to create a digital twin that is useful for quality assurance, thus one of the research questions is how to create appearance specifications, so that it can be used for quality assurance.

The research goals of this thesis is as follows:

1. Develop instrumentation for estimating surface radiometry.
2. Establish methodology for appearance specification
3. Develop methods to create a digital twin from measurements
4. Establish a framework to compare the physical and digital twins

1.4 Thesis Structure

In Chapter 2 we describe the background and related work to the central topics of the thesis. For related work on the individual contributions, we refer the reader to the related work sections in the respective papers.

In Chapter 3 we describe and discuss the development of the instruments used in the contributions listed in **??** . This chapter focus on information that has not made it to the published manuscripts due to space limitations. It is the intention that this chapter, will give the reader some insight into the challenges related to developing such an instrument, as well as the ability to recreate the instrument if one were to buy the equipment.

In Chapter 4 we present and discuss the contributions and put them into the perspective of research goals, related to comparison of the physical and digital twin and instrumentation for estimating the surface radiometry in an industrial setting.

1.5 Expected Background

For the full appreciation of the contents of this thesis, we recommend background knowledge within Linear Algebra, Image Analysis, Computer Vision and Computer Graphics.

Specifically for Computer Vision, the topics of Camera modeling, Multi-view Stereo and Structure From Motion are assumed known. The textbook *Computer Vision: Algorithms and Applications* [16], provides the needed background knowledge.

Regarding Computer Graphics and Light Transport topics, the textbook *Physically Based Rendering: From Theory to Implementation* [17], provides a strong background on the practicalities related to rendering. The textbook *Color Imaging: Fundamentals and Applications* [18], provides a good insight into the physical phenomena and a more physics-related angle on appearance.

CHAPTER 2
Background and Related Work

In this chapter we guide the reader through the definition of *appearance* as it is used in this thesis, the physical phenomena and how we can model it. Additionally, we discuss options for measuring the appearance and how it has been done previously, with focus on the instrumentation to do so. This leads to the final part of closing the loop, comparing appearance, ways to do that and how that has been done previously.

2.1 Radiometry

In order to describe an object's interaction with light, we start by briefly introducing the terminology for radiometry and the notation used. For an in depth explanation the reader is referred to the textbook *Color Imaging: Fundamentals and Applications* [18].

If we start with the example of buying a light bulb, the power consumption of the light bulb is usually defined in Watts. We refer to the part of this emitted as light as radiant flux

$$\Phi = \frac{dQ}{dt} \quad [\mathrm{W}]$$

where $Q\,[\mathrm{J}]$ is the radiant energy, which would correspond to the total energy emitted in a given time t. If we take a picture of the light bulb, each pixel would be an area illuminated by the radiant flux, we refer to this as irradiance

$$E = \frac{d\Phi}{dA} \quad \left[\frac{\mathrm{W}}{\mathrm{m}^2}\right]$$

which is radiant flux received per area. The exposure time of the camera integrates irradiance over time, giving us radiant exposure

$$H = \int_0^T E\, dt \quad \left[\frac{\mathrm{J}}{\mathrm{m}^2}\right]$$

which is the physical quantity that a camera measures.

When a camera chip is exposed to light, charge accumulates and is read by an Analog to Digital Converter (ADC), giving the final pixel value in either 8-bits or 12-bits for most cameras. This pixel value correspond to radiant exposure, at an unknown scale. Calibrating this scale requires special instruments and is not easily done. For our applications of comparing measurements, the scale is not needed to be known as long as it is constant between measurements. There are many sources of noise when using cameras to measure, most of those are addressed in the EMVA 1288 Standard [19], which is the *Standard for Characterization of Image Sensors and Cameras*. The two largest sources of noise in images are *dark current* and *shot noise*. The influence of Dark Current noise can be reduced by keeping low camera temperature and subtracting "dark" images from all acquired images. Shot noise is statistical noise in the number of photons exciting the sensor, and can be mitigated by acquiring multiple images and averaging pixel values.

To help us describe how light reflects in a surface, we can use the reflected radiance equation [20]

$$L_r(\mathbf{x}, \vec{\omega}_r) = \int_\Omega f_r(\mathbf{x}, \vec{\omega}_i, \vec{\omega}_r) L_i(\mathbf{x}, \vec{\omega}_i)(\vec{\omega}_i \cdot \vec{n}) \, \mathrm{d}\vec{\omega}_i \qquad \left[\frac{\mathrm{W}}{\mathrm{sr\,m^2}}\right] \tag{2.1}$$

where $L_r(\mathbf{x}, \vec{\omega_r})$ is the radiance reflected off a surface at any given point $\mathbf{x}$ and direction $\vec{\omega}_r$, can be modeled by a surface integral over the hemisphere $\int_\Omega d\vec{\omega}_i$ centered around $\mathbf{x}$. For each light direction $\vec{\omega_i}$ we multiply the BRDF term $f_r(\mathbf{x}, \vec{\omega_i}, \vec{\omega_r})$ with the incoming radiance $L_i(\vec{n}, \vec{\omega_i})$, weighted with the dot product between surface normal $\vec{n}$ and $\vec{\omega_i}$, these vectors must be of unit length. The function f_r describes the ratio of reflected radiance in direction $\vec{\omega_r}$ given the incident irradiance at direction $\vec{\omega_i}$, at a specific point on a surface $\mathbf{x}$, hence this function is generally used to model the optical properties of a material, it is further explained in Section 2.2.

The reflected radiance depends on wavelength and time, but when observing with a camera, we can only measure Red, Green, and Blue (RGB) which are spectrums of light and we assume that the light is constant over time, thus we leave out the wavelength and time dependency.

2.2 Appearance Definition

In this thesis we use the term appearance as how an object interacts with light. Typically, we represent the appearance of an object as a combination of optical properties (material) and geometry. We refer to optical properties as a separate term, but it is just a way to represent interaction with geometry at a micro- or nano-scale. We humans can only visually perceive the micro- and nano-geometry from its interaction with light, as such we consider the border between geometry and optical properties to be at the scales of our perception.

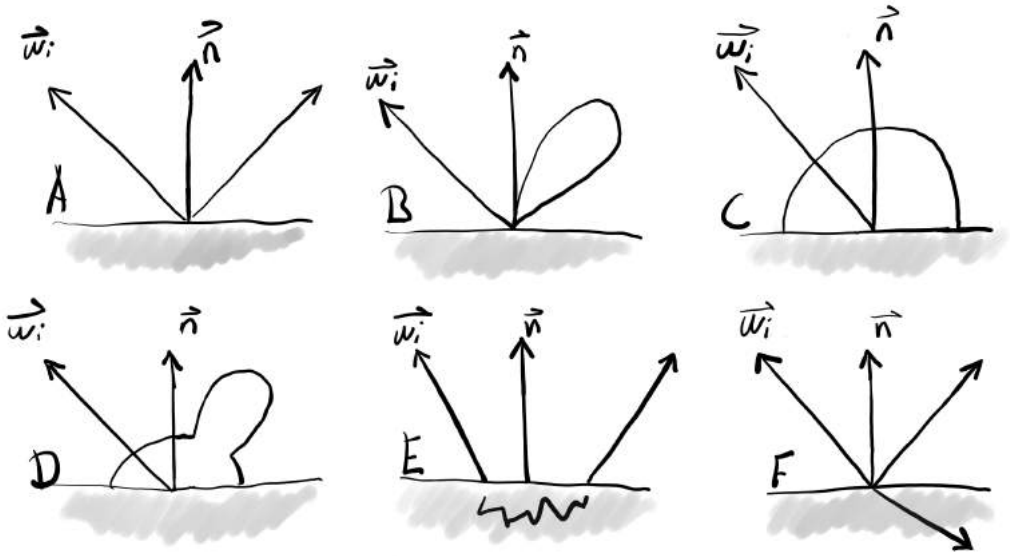


Figure 2.1: Showing different light material interactions. A being a perfectly specular material. B being a glossy material. C being a diffuse material. D is a combination of glossy and diffuse. E is sub-surface scattering. F is transparent.

In Figure 2.1 we see some different light material interactions, $\vec{n}$ is the surface normal and $\vec{\omega}_i$ is the incident light direction. If we start from a simpler-to-model point of view, we have A, B, and C being the basic light/material interactions for reflections. A being a perfectly specular material, where all light is reflected in the direct reflection. B being a glossy material, spreading the reflected light but still only around the direct reflection. C being a diffuse material, which reflects light equally in all directions on the hemisphere. Common for A, B, and C is that those are ideal reflectance representations and most materials behave as a combination of the three. D is a combination of glossy and diffuse, which is a much more realistic scenario for real world materials. E is sub-surface scattering where the light exits at a different place than it enters, this phenomenon is often modelled as a diffuse component. F is a transparent material, where the light is partially reflected off and transmitted through the surface. To acquire a digital representation of a real object, we have to deal with D, E and F or some combination thereof.

2.3 Instruments for Appearance Acquisition

In the introduction we motivated the usefulness of being able to compare a digital and physical twin. A way to enable this comparison, is to create an instrument that is reliable and precise enough to consistently perform measurements. In this chapter we investigate

the available research to better identify the problems already solved and further motivate the focus on sparse radiometric measurements of industrial samples using robot arms as instrumentation. The focus is primarily on instrumentation rather than methodology. Just to remind the reader, our ideal goal is to develop an instrument where an operator puts in a manufactured object, and a digital twin for comparison. Then the instrument can quantify the difference in perceived appearance between the manufactured object and its digital twin or in this case reference model. As such the points of interest in the previous work is instruments to measure appearance.

Ideal Requirements for Instrument

As we are interested in measuring the reflected light, we find that basing the instrument on cameras and controllable light sources, to be a logical solution. In order to work with samples of various shape and material, the system has to move around the object. Thus it is a requirement for the instrument that the object remains stationary during acquisition. This infer that the system must be able to position a camera and a light source such that the object can be observed and illuminated in any configuration, without moving the object. Making sure that the object is stationary makes it possible to scan fragile objects. Having no constraints on camera and light positioning, allows for better sampling of materials with complex optical properties, such as anisotropy and narrow specular lopes. A setup with limitations on camera positioning will suffer from aliasing effects from sampling the high frequency behavior from a sparse set of positions. Since we want to measure objects with a wide range of shapes and materials, it is important that the samples remain stationary.

Related Work on Instrumentation

A survey on the *Advances in Geometry and Reflectance Acquisition* by Weinmann and Klein [21] provide an in depth investigation of the previous work on both methods and instrumentation for acquiring geometry and reflectance models up until 2015. They divided the instruments into three groups: *Gonio-reflectometers*, *Camera Arrays* and *Mirror/Kaleidoscope* systems. These categories allow us to get an overview over the type of actuation. But we are also interested in identifying the versatility of the setups for changing experiment parameters, thus we also look into the actuation of the camera and light source for positioning. All methods seem to be based around cameras and light sources, and as such the variation lies within the choice of actuation and measurement procedures. A list of instruments and an overview over their methods of actuation can be seen in Table 2.1, which serves as an overview over a selected subset of relevant instruments from the survey and others.

If we start by looking into the dynamics of the instruments from Table 2.1. We have the instruments from [22, 23, 24, 25] that mount a flat sample on a robot arm, responsible for moving the sample, rather than the camera and light source. Sattler et al [22] use a fixed

Table 2.1: Comparisons of actuation approaches of the instruments used to estimate surface radiometry. The output class *Radiometry* is used for instruments that do not have the purpose of measuring a BxDF.

Method	**Year**	**Object**	**Sensor**	**Light Source**	**Geometry**	**Output**
Murray-Coleman & Smith [31]	1990	Stationary	Gantry	Gantry	Flat	BRDF
Ward [32]	1992	Stationary	Gantry	Gantry	Flat	BRDF
Dana et al [33]	1999	Robot arm	Fixed	Fixed	Flat	BTF
Marschner et al [34]	2000	Stationary	Fixed	Manual	Sphere/Cylinder	BRDF
Dana [35]	2001	Stationary	Fixed	Fixed	Flat	BRDF/BTF
Matusik et al [36]	2003	Stationary	Gantry	Gantry	Sphere	MERL BRDF
Sattler et al [22]	2003	Robot arm	Gantry	Fixed	Flat	BTF
Hünerhoff et al [23]	2006	Robot arm	Fixed	Gantry	Flat	BRDF
Kimachi et al [37, 24]	2006	Robot arm	Fixed	Robot arm	Flat	Radiometry
DOME I [28, 29]	2008	Stationary	Fixed	Fixed	Measured	BTF
Holroyd et al [38]	2010	Stationary	Gantry	Gantry	Measured	SVBRDF
Höpe et al [25]	2012	Robot arm	Fixed	Gantry	Flat	BRDF
DOME II [30]	2013	Turntable	Fixed	Fixed	Measured	BTF
UTIA [26]	2014	Stationary	Gantry	Controlled	Flat	BRDF
Nielsen et al [39]	2017	Turntable	Fixed	Fixed	Measured	BRDF
Lyngby et al [27]	2019	Stationary	Robot Arm	Fixed	Flat	BRDF
X-rite Tac7 [40]	2020	Turntable	Fixed	Fixed	Flat/Staircase	SVBRDF
UR5-Setup Chapter 3	2021	Stationary	Robot Arm	Robot Arm	CAD	Radiometry

light source and a moving camera on a rail, where the instruments from [23, 25] have fixed camera, movable light source and flat sample. Filip et al [26] use a gantry based gonioreflectometer setup made to measure the Bi-directional Texture Function (BTF) of flat samples put on a turntable, unlike the previous setups the sample seems to not need to be mechanically fixed. In the setup by Lyngby et al [27] a robot arm is used to move the camera with relation to a flat sample, but the light source is an arc with a fixed array of Light Emitting Diodes (LEDs) from 0° to 90° in increments of 7.5°. This has the advantage of being able to select an arbitrary observer direction, but like DOME I[28, 29] and DOME II[30], it is noticeable in the measurements that the light source positions are few and mechanically fixed.

We see multiple instruments utilizing robot arms to actuate the sample, and that provides many benefits with relation to motion planning, compared to our setups in Chapter 3. Having the robot constrained to a small portion of its practical work-area leads to less risk of intersection with other items and easier avoidance of singularities in the Inverse Kinematics (IK) when performing the motion planning. The limitations of such an approach is that the sample must be fixated on the end-effector of the robot arm, and that severely limits the range of shapes that can be evaluated in the system. The problems related to this are described in more detail in Chapter 3.

Besides the aforementioned methods that work only on flat samples, there is also a subset of the methods [36, 34] and X-Rite TAC7 [40], that rely on samples of special geometry in order to measure the appearance. Matusik et al [36] use spherical objects, Marschner et al [34] use both spheres and cylinder shapes to measure multiple light/camera/surface normal configurations in a single image.

While all of the instruments in Table 2.1 are created to measure radiometric quantities, almost all of them seem to have the purpose of estimating BxDF models for computer graphics uses. This is all except [23, 25], which seem to be developed to evaluate appearance-related quantities with high spectral sensitivity. The purpose of our instruments is not to acquire BxDFs, but to estimate the surface radiometry. While acquiring a BxDF involves measuring the surface radiometry, our focus is on using the radiometric measurements directly to generate and compare with the digital twin.

The instruments by [28, 29, 30, 38, 39] all estimate the geometry of the sample as well as their chosen BxDF model. In DOME I [28, 29] 151 cameras are used to estimate the geometry by shape from silhouette, where in DOME II [30] this is done by adding projectors to the system allowing for structured light 3D scanning. In [38, 39] the geometry is also estimated by using structured light scanners. All of these methods reproject the geometry into the images and estimate the ray paths from light source to object and then to camera. This allows for isolation of the BxDF parameters and dense measurements of BxDFs on arbitrary geometries. Our ABB-setup from Chapter 3 uses a structured light scanner to acquire geometry, but cannot estimate the BRDF simultaneous, as it requires a reconfiguration to use the light arc from [27]. Where as

the UR5-setup from Chapter 3 does not have any means of acquiring geometry, primarily due to weight and size constraints, and relies on other modalities to obtain the geometry.

Common for all these systems are that they have very long acquisition times, ranging from hours to days, and output complex appearance models suited for rendering. This is a direct consequence of trying to estimate BxDFs. For example, if we were to densely measure the 4-dimensional BRDF at a 1° resolution, that would require $90 \cdot 180 \cdot 90 \cdot 180 \approx 260 \cdot 10^6$ samples. Measuring 1 sample pr. second, that would take around 8 years. In an industrial setting it might not be beneficial to work directly with BxDF functions as they are high-dimensional and it is often not meaningful to perform parameter comparisons on these methods.

So far we have discussed the gonio-reflectometers, but taking a look at the Camera Array setups, we have DOME I [28, 29] and DOME II [30]. These setups are expensive, complex and super fast. This is due to the fact that they use 151 and 264 cameras for acquisition and have "no moving parts", which is mostly true, as they added a turntable to rotate the sample, but one could argue that the rotation time for that is negligible. This is also the downside to this setup, as the cameras and light sources are fixed in relation to the sample, resulting in a limited resolution. Especially the fact that the cameras are distributed very evenly across the sphere, is a limitation for BTF acquisition [21]. For companies, acquiring the BTF of an object often one would buy a system like TAC-7[40] from XRite, which is an industrial version similar to the DOME setups, and a breakdown can be found in this article [41] where they use the TAC-7 to extract a Spatially Varying Bi-directional Reflectance Distribution Function (SVBRDF) instead of the BTF normally measured by the instrument.

Theses instruments provide high acquisition speeds at the cost of versatility and resolution. They are the exact opposites of our setups, where we sacrifice the acquisition speed for high versatility and significantly increased resolution, theoretically the accuracy of the robot arms are the limiting factor in our setups.

Previous work on instrumentation have been focusing on very specific tasks, measuring either different BxDF models, geometry or both. This focus has resulted in some very specialized instruments, such as gonioreflectometers, spherical gantrys, mirror setups, and massive domes using many cameras. Common for these setups is that they are custom setups, that require custom built components and a significant amount of resources to be recreated or modified for slightly different tasks.

The gantry setups are ideal for positioning the camera and light source on a hemisphere with equal distance to the center. But have the downside that the focus is fixed at the center of the gantry, and does not allow for arbitrary camera and light placement in the space around an object. The ability to evaluate reflectance at specific regions could become a difficult task if that region is located significantly far away from the gantry center. This problem is also present in the dome setups, while they are also being limited

by resolution, as they have fixed cameras at 7.5° angles. For appearance evaluation it can be difficult to sample the spectral peak at such a coarse resolution.

All instruments in Table 2.1 are focused on measuring the radiometry to estimate BxDF models. But we do not see any of these papers focus on creating a controlled environment that allows us to take our measurements and evaluations a level up and address the radiometry on a scale of human perception. By doing this, we are able to compare a digital representation with its physical counterpart. And as that has not previously been the focus, the highly specialized setups are not built with that purpose in mind. Thus we see a need for a versatile setup, that can position a camera and a light source around an object, very precisely and accurately, so that we can compare our digital renderings with actual photographs in the setup as we do in Contributions A and E.

The current trend, for research within the Computer Vision and Computer Graphics communities, seems to be focused on developing new methodology rather than instrumentation. Most of which either relies on using one of the instruments from Table 2.1 or an *in the wild* setting. We see a strong focus on methods for handheld devices and Deep Learning methods e.g. [42, 43, 44, 45, 46, 47, 48]. To support the development of such methods, it is crucial to have proper reference data. Obtaining proper reference data is a core contribution in Contributions A, F and G.

Using an instrument to create repeatable input data with accompanying reference output, makes it possible to evaluate the performance of methods. The availability of good reference data is important for developing new methods. This was a big motivation for Contribution F, where we developed a dataset and benchmark for other researchers to compare their methods against the performance of other methods.

From the survey by Weinmann and Klein [21], we identify a few challenges. One is pose estimating light sources in acquisition setups, according to the survey it is primarily done by triangulating highlights in arrays of mirror spheres of known positioning and radii or clever optimization as in [30]. We found this topic to be open for improvements on its practicality, and in Contribution E we investigate using the shadow of an arbitrary object with known geometry to estimate the direction to one or more point light sources.

2.4 Appearance Comparison

Using the parameters of any BxDF model for comparison is not straight forward. Inter-model comparison or even quantification of the similarity of two materials, given their sets of parameters, is challenging. For many appearance models the scale of which the parameters operate are very different, complicating the process of quantifying the deviation. In microfacet models [49], for example, we have spectral complex indices of refraction, normal distributions, roughness and color. These parameters interact non-linearly and on different scales. So a change of 0.1 in any RGB parameter is much

different than a change of 0.1 in the index of refraction. These reflections have been researched previously by Havran et al [50]. While they provide important insight to the problem and an approach to this problem, the problem remains largely unsolved.

In [51], they discuss the importance of being able to compare rendered images with photograph of the same scene. They argue that it is important to further advance the field of computer graphics. We argue that it is also highly beneficial to perform VQA as well. We addressed this in Contributions A and E, where we aimed to make it more practical to compare photographs and renderings. The primary reason we are interested in measuring the appearance is for using it to compare a physical object with its digital twin. In the computer graphics community, this is sometimes done by comparing renderings with photographs. While the purpose of a large portion of the graphics research is to create renderings that are indistinguishable by humans from photographs. Most methods are often compared qualitatively and not quantitatively. The former would be a side-by-side visual comparison of two images as in manual VQA, while the latter would for example be investigation of pixel-to-pixel differences. An example of side-by-side qualitative comparison is the CornellBox [52, 53], where they have subjects looking at a photograph and a rendering on a screen and deciding which is real.

While being a very informative and an useful evaluation form, pixel-wise comparisons of renderings and photographs, are seldom used in literature. This quantitative way of describing the methods is rare, often authors chose to let the reader perform the qualitative evaluation, of comparing the rendering with an image. One of the main reasons for this, is that it is a highly complex task to create a physically accurate digital representation of a physical scene. Some of the first to do pixel-by-pixel comparisons was Rushmeier et al [54] and Pattanaik et al [55], which both found that the biggest source of error in their comparisons where misalignment of the geometry in the physical and digital representations. Solving this alignment from images is still an active research topic, especially in Differentiable Rendering. A recent contribution to this is by Loubet et al [56]. We also found this geometry alignment to be an issue, and in Contribution A we solved it by gluing on markers, and use marker based pose estimation methods, and optimization for refinement. Where in Contribution E we rely on a CAD model, to estimate the pose of the object, making for a more practically feasible solution, that could be used as an initial guess and further improved in a differentiable renderer like Mitsuba 2 [57].

CHAPTER 3

Development of Flexible Instruments for 3D and Appearance Acquisition

In this chapter we dive into the development of the instruments used in the contributions. In Contributions A and F we use an industrial robot with a mounted 3D scanner, focusing on geometry acquisition and structured light scanning, we will refer to this instrument as the *ABB Setup*. Where in Contributions B to E we use a dual robot system to move a camera and light source in relation to an object, with the focus to estimate surface radiometry, we will refer to this as the *UR5 Setup*. We will discuss the details of implementing the setups and the methods for evaluating the accuracy and precision of these two instruments here, as many of these details have been left out in the published papers, due to space constraints.

There are many reasons for using 6 Degrees of Freedom (DoF) robot arms for moving 3D scanners, cameras, and light sources in Computer Vision applications, besides the obvious of automating movement, industrial robots are built to have a very high repeatability. They also have a versatility in regards to which applications they can be used in, compared to specialized setups like the gantry setups often used in appearance acquisition. With the introduction of Collaborative Robots (Cobots), like the Universal Robots UR(3,5 and 10) series, they are no longer required to be behind security fences and proximity switches to be operated safely. This, and a significantly lower price compared to ”old” welding robots like the ABB IRB-1600 10/1.45m, have made them very approachable for introducing automation into applied research.

Mostly the work here related to radiometry measurements are very similar to related work on instrumentation for BTF measurements. While we do not perform dense measurements as for BTF estimations, there is simply a large overlap of methods and practical approaches.

A limitation that the robots solve for us is the need to pose estimate the camera every time we move the robot. The repeatability of the robot allows us to only calibrate each

position of the camera once, and then reuse previous calibration when revisiting the position. This allows for much more complex data acquisitions, allowing the work of Contributions A and F to be carried out.

The price of these robot arms for manufacturing have dropped significantly, and the ease of use have increased as well. The combination of the two are making the use of Cobots for this task a very interesting prospect. Previously the hardware for instrumentation needed to be custom designed and built, requiring an enormous amount of resources, both in terms of money and technical staff. Cobots appear to provide a way to overcome these limitations or at least most Cobot manufacturers claim that their robots can be easily used.

A big hurdle in doing physical experiments related to appearance measurements is the level of accuracy and precision needed to capture these complex light and matter interactions.

This allows for easier installations or even portable system as there is no need for large security systems. Further it allows for interacting with the scene while the robots are moving without endangering oneself. An example of this is that the UR5 weight is approximately 13kg aluminium and plastic casing, where the ABB IRB 1600 is a 250kg cast iron casing. While the UR5 still packs a punch if it were to collide with an operator, when it moves 1.0m/s the force is still small compared to 250kg cast iron at the same speed.

ABB Setup

The *ABB Setup* is a setup built for high precision movement, using an industrial welding robot ABB IRB 1600-10/1.45, meaning a payload of 10kg and a reach of 1.45m. The repeatability is 0.02mm, allowing for very high precision when positioning the 3D scanner. The *ABB Setup* is an upgrade of a previous system built around the same robot for acquiring the DTU MVS dataset [58]. The upgrade is improving the mounted structured light 3D scanner and the environment lighting. Photographs of the setup can be seen in Figure 3.1.

The 3D scanner is a stereo camera structured light setup, build of two PointGrey Grasshopper3 9.1MP color cameras on either side of a WinTech LightCrafter 4500Pro projector with a resolution of 1920x1080 pixels. We used the Phase Shifting Heterodyne principle [59] for our structured light patterns, and 3D reconstruction. The cameras are positioned slightly toe-in of approximately 10°, and while in theory the toe-in configuration increases the warping in the rectification step, we did not see an increase in error. Further by using the VDI-2634 [60] standard for measuring and reporting the precision and accuracy of 3D area scanners, we found the results in Table 3.1, to be acceptable for our applications.

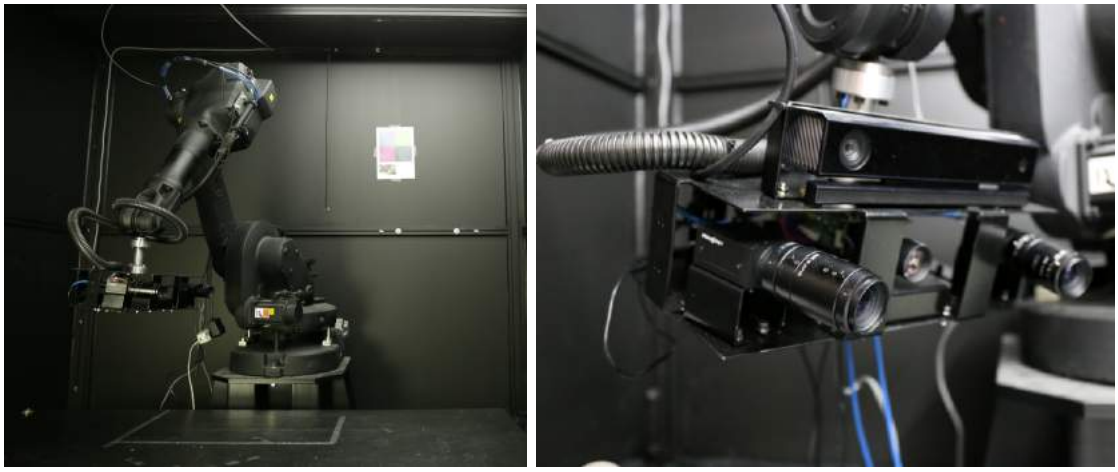


Figure 3.1: Left: Industrial robot system, ABB IRB 1600 1.45m. Right: 3D scanner rig, consisting of a Microsoft Kinect version 2, and a custom structured light 3D scanner. Using 2 PointGrey Grasshopper3 9.1MP RGB cameras, and a WinTech Lightcrafter 4500Pro. These images are also found in Contribution F.

Error type	μ[mm]	σ[mm]
Form	0.01	0.32
Sphere	-0.33	0.50
Flatness	0.29	0.56

Table 3.1: Results from the VDI-2634 [60] standard on geometric errors from 3D area-scanners. According to the standard results should be reported as bias μ and standard deviation σ.

Because of the high repeatability of the robot, we can create a predefined series of scanner positions and perform pose estimation as a separate step from data acquisition. This has two major benefits, first it allow us to use camera calibration targets to pose-estimate the scanner giving better point-cloud stitching. Secondly, we do not need to have trackers in the scene, giving much more freedom to the choice of scenes. This was crucial for Contributions A and F.

UR5 Setup

For our appearance related research we needed a setup to accurately position a camera and a light source around an object. The *ABB Setup* didn't have a second robot to move a light source separately from the 3D scanner, and the risk of colliding with another robot and severely damaging it was a strong motivator for not adding a second robot to that setup. Instead we put two Universal Robots UR5, 6-DoF robot arms on either side of a pedestal, see Figure 3.2. On the left robot is a PointGrey Grasshopper3 6MP color camera, we chose the 6MP over the 9.1MP used in the *ABB Setup*, due to the larger pixel size, giving better light sensitivity and noise levels. White light LEDs are known for a large intensity peak around 440nm, and the Thorlabs MNWHL4 4900K LED was chosen over a 6500K due to a reduced peak.

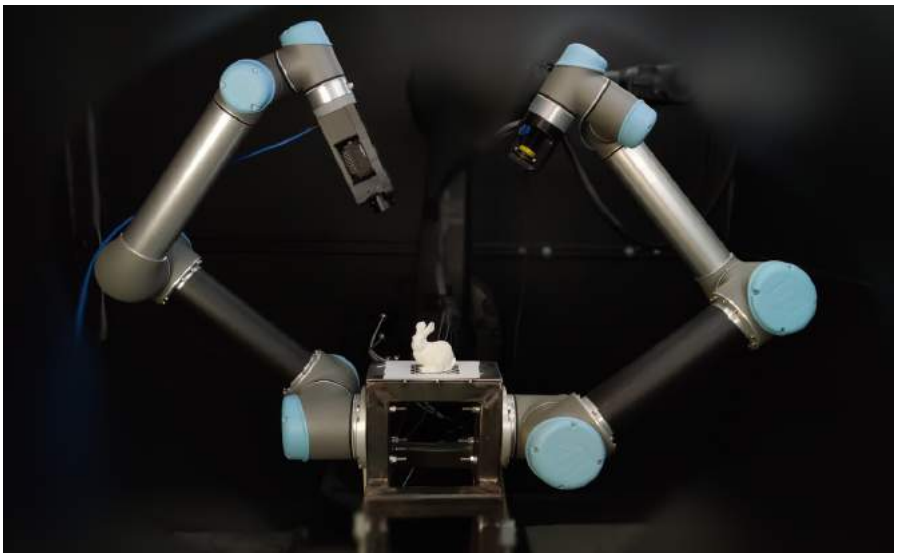


Figure 3.2: Image of UR5 Setup, with the two Universal Robot UR5. Right arm holding a PointGrey Grasshopper3 6MP color camera. Left arm holding a Thorlabs LED MNWHL4 4900K, for a reduced blue peak compared to a 6500K LED.

The repeatability of the Universal Robots UR5 robots are lower than the ABB IRB 1600, they state a worst case repeatability of 0.1mm. Thus we can use the same, calibrate once and run forever, approach as we did in the *ABB setup*. This made it possible for us to scan hundreds of different samples and objects without having to recalibrate.

UR3 Setup

In a collaboration with Center for Cultural Heritage Technology (CCHT@Ca'Foscari) at Istituto Italiano di Tecnologia (IIT), we looked into 3D reconstruction of glass objects. We developed an acquisition setup as seen in Figure 3.3, based around a Liquid Crystal Display (LCD) screen, a turntable and an Universal Robot UR3 with two FLIR BlackFly

S BFS-U3-51S5P-C polarization cameras mounted. The setup uses a light source with polarized light, the LCD screen, and then observe changes in polarization in the light transmitted through the glass object, measured by the polarization cameras. We found that the light transport paths were too complex to model with sufficient accuracy for this setup. As such we used Convolutional Neural Networks (CNNs) to estimate the shape of the glass objects. This is the physical setup used for generating data for the neural network in Contribution G.

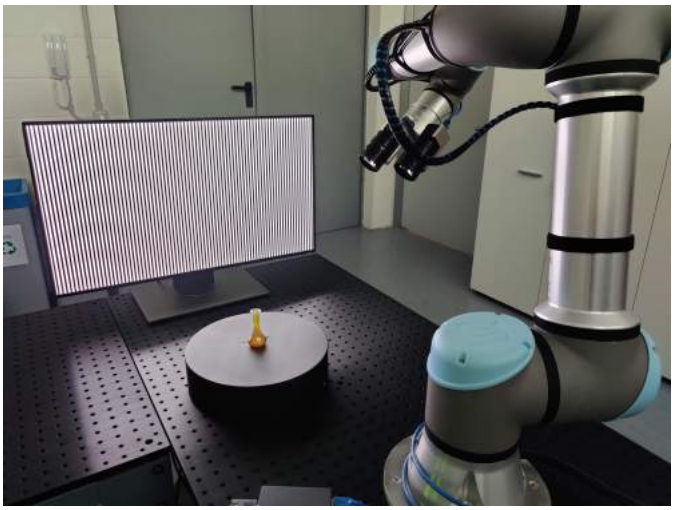


Figure 3.3: Image of UR3 Setup. Used as a research tool for 3D acquisition of glass objects, developed in collaboration with CCHT@Ca'Foscari IIT. Consists of a LCD screen, projecting light patterns, a turn table and an Universal Robot UR3 to move two FLIR BlackFly S BFS-U3-51S5P-C cameras. The glass objects on the turn table is a replica of an ancient glass bottle, thus it is not transparent. Photograph courtesy of Stuart James.

3.1 Practicalities of Camera Calibration

The basics behind camera calibration is described in most computer vision textbooks such as [16], and many good implementations are openly available online. We used the camera calibration implementation from OpenCV [61], which is based on checkerboard patterns like seen in Figure 3.4 and the method by Zhang [62].

Checkerboard corners are well defined in image space and very robust, making checkerboards ideal for intrinsic camera calibration. For intrinsic calibration only the relation between the points are needed, not their absolute 3D position in world space. On the other hand for extrinsic calibration the 3D position in world space must be known for all points in order to correctly estimate the pose of the camera. Only if the checkerboard is asymmetric, and fully visible in all images, then the extrinsic parameters can be estimated. This requirement is difficult to fulfil in a practical setup like the robot setups.

To circumvent this requirement we use ChArUco boards, which extend the checkerboard with the inclusion of ArUco markers [63, 64]. This gives a unique identification of each tile, and thus each corner, making calibration possible with only a subset of the markers visible in the camera.

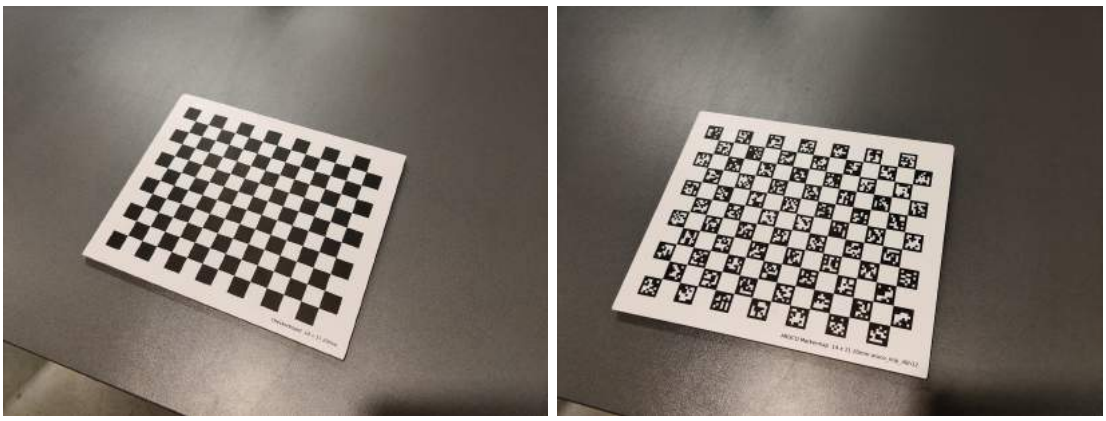


Figure 3.4: Left: image of a normal calibration checkerboard . Right: a ChArUco board. The calibration boards are bought from https://calib.io

We experienced that for high resolution images, the corner detection from OpenCV got stuck and struggled to find the pattern. Even in what we would see as perfect images. We found that using down-scaled images for finding the corners and full resolution images when finding sub-pixel position, provided both the fastest but also the most robust calibration.

When taking photographs there is radial distortion due to the lens of the camera. This is rarely modelled in computer graphics and to be able to compare photographs and renderings we have to un-distort the photographs first.

Most 3D modeling and rendering software e.g. Blender, KeyShot, Autodesk Maya etc. uses artist friendly representations of the camera parameters. Often the input is as a camera position $\mathbf{p}$ and orientation is often represented as a rotation in Euler angles or quaternions, some times it is even represented as a lookat vector and up direction vector. The internal parameters are often defined as the field of view, in either direction, in degrees and the size of the output image in pixels. These can all be found from the camera calibration.

The output of the camera calibration is the 3D location $\mathbf{t}$ and rotation $\mathbf{R}$ of the checkerboard in relation to the focal point of the camera. From that we can get the camera position by

$$\mathbf{p} = -\mathbf{R}^T\mathbf{t}$$

If we extract the three row vectors U, V, and W making up the rotation matrix as such,

$$\mathbf{R} = \begin{bmatrix} U & V & W \end{bmatrix}$$

We get W as the lookat direction, V as the up direction, and U as perpendicular to the two. This is a change from our notation using arrow overline, but it is to comply with the notation used in Computer Graphics (CG) and rendering resources. The up vector is often selected to be y-up, but that is not the case in most real world cases.

We can derive the field of view from the width of the sensor and focal point in pixels, as

$$\text{fov}_x = 2 \arctan \frac{w_{px}}{f_x}, \quad \text{fov}_y = 2 \arctan \frac{h_{px}}{f_y}$$

3.2 Mechanical Setup and Kinematics

In order to use a robot to accurately position a camera or light source in relation to an object, we need to know where the robot is located in relation to the object. While applicable to any system using actuators, it was only strictly needed in the *UR5 Setup*, as for the *ABB Setup* and *UR3 Setup* absolute positioning accuracy was not needed. For Contribution C we needed to have an accurate position of camera and light source in relation to the surface normal of the sample. This was because we were looking at the reflected radiance, and recalling from Equation (2.1), we are effectively sampling a single light direction $\vec{\omega}_\ell$ so that L_i can be described as a delta function

$$L_i(\mathbf{x}, \vec{\omega}_i) = \delta(\vec{\omega}_i - \vec{\omega}_\ell) V(\mathbf{x}, \mathbf{x}_\ell)(I_\ell / |\mathbf{x}_\ell - \mathbf{x}|^2)$$

where $V(\mathbf{x}, \mathbf{x}_\ell)$ is the visibility function, indicating if the light is visible or not at $\mathbf{x}$. Since a point light source does not have an area, it cannot emit radiance, but rather we divide the intensity I_ℓ with the distance squared to the point light $(I_\ell / |\mathbf{x}_\ell - \mathbf{x}|^2)$. Combining this into Equation (2.1), gives us

$$L_r(\mathbf{x}, \omega_r) = \int_{2\pi} f_r(\mathbf{x}, \omega_i, \omega_r) L_i(\mathbf{x}, \omega_i)(\omega_i \cdot n) \quad d\omega_i \tag{3.1}$$

$$= f_r(\mathbf{x}, \omega_\ell, \omega_r) V(\mathbf{x}, \mathbf{x}_\ell)(I_\ell / |\mathbf{x}_\ell - \mathbf{x}|^2)(\omega_\ell \cdot \vec{n}) \tag{3.2}$$

Due to the nature of BxDFs, small deviations in $\vec{\omega}_\ell$ and $\vec{\omega}_r$ can lead to large changes in the reflected radiance L_r. Thus the accuracy of the instrument largely depends on the accuracy for positioning the camera and light source.

To obtain a high accuracy we need to model the physical positioning of the various components and their actuation, in control logic using this information to control movement is called kinematics. In the field of robotics the mechanical system is often divided into groups of coordinate systems called frames [65]. We represent transformation between frames using the notation $\mathbf{T}^{to}_{from}$, so if we wanted to have a transform from world coordinates to robot tool coordinates we would write $\mathbf{T}^{tool}_{world}$. Applying this to the *UR5 Setup*, we have 7 important coordinate frames, as seen in Figure 3.5.

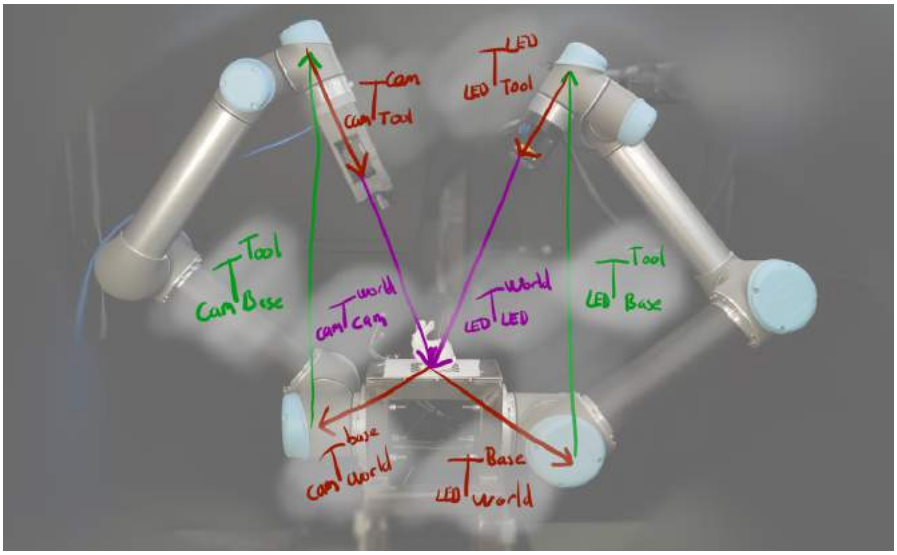


Figure 3.5: Drawing of the transformations related to camera and robot. Green arrows represent transformations based on robot movement, red are fixed offsets due to mounting, and purple are transformations are information needed for a CG pipeline to render images. The prefix of cam and LED is to avoid long suffixes on frame names.

The first frame is the *world* frame, which is a reference point on the pedestal. The two robot *base* frames are points inside the robot mounting bracket. The *tool* frame is the robots internal representation point for the end-effector mounting bracket. The *LED* is the physical location of the LED. The *camera* is the mathematical representation of the focal point of the camera. The locations of these make them hard to measure by any form of ruler. We have to resort to other methods described later to do that.

By using Homogeneous coordinates we can represent the transformation between two frames as a rigid motion

$$\mathbf{T} = \begin{bmatrix} \mathbf{R} & \mathbf{t} \\ 0 & 1 \end{bmatrix}, \quad \mathbf{R} \in \mathrm{SO3}, \quad \mathbf{t} \in \mathbb{R}^3 \tag{3.3}$$

This has the advantage that we can easily go back and forth between coordinate frames

as

$$\mathbf{T}_{world}^{tool\ -1} = \mathbf{T}_{tool}^{world}$$

In order to accurately use the kinematics we have to obtain the transformations between frames. If we look at Figure 3.5, the red arrows are fixed transformations as they represent how the robot is attached to other parts. These will not change during operation and small errors in the rotation part of these result in nonlinear large error in absolute positioning. Especially the $\mathbf{T}_{world}^{base}$ transformation, as 1° error in the *base* of the robot can propagate to errors in the *tool* position in the millimeter range. The green arrows represent robot motion, which is a black box in this system, but we rely on the fact that Universal Robots inform that the error is no larger than 0.1mm. The purple lines represent the position of the light and camera in relation to the *world* frame, $\mathbf{T}_{camera}^{world}$ can be estimated by doing camera calibration.

In the beginning of using the system, we had to rely on measuring the fixed transformations by hand and relying on dimensions in the CAD models of the system and manual tweaking. This is of course not optimal and we found that it could be calibrated by using hand-eye calibration. Hand-eye calibration is a well researched topic and a few methods can be found as implementations in OpenCV [61]. Hand-eye calibrations can be split into two categories, $\mathbf{AX} = \mathbf{XB}$ which is used when only the relation between robot and camera is needed, and $\mathbf{AX} = \mathbf{ZB}$ when the position of the robot is also needed.

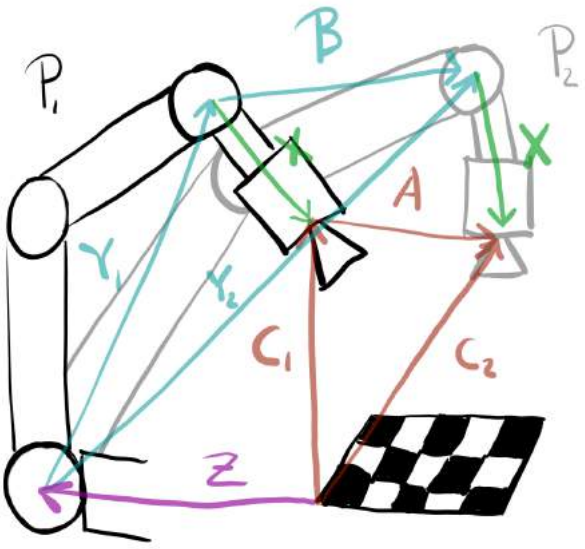


Figure 3.6: Showing the difference between $\mathbf{AX} = \mathbf{XB}$ and $\mathbf{AX} = \mathbf{ZB}$ hand-eye calibration. $\mathbf{A}$ and $\mathbf{B}$ are the movement of $\mathbf{T}_{base}^{tool}$ and $\mathbf{T}_{world}^{camera}$. $\mathbf{X}$ is the transformation from the robot tool to the camera $\mathbf{T}_{tool}^{camera}$. $\mathbf{Z}$ is the transformation from the world coordinate system to the robot base $\mathbf{T}_{world}^{base}$. Both $\mathbf{X}$ and $\mathbf{Z}$ are constant for all movement.

The basic concept of hand-eye calibration can be seen in Figure 3.6. To perform hand-eye calibration we rely on the camera being fixed to the robot, by the transform $\mathbf{X}$, and the robot *base* is fixed w.r.t. the checkerboard by transform $\mathbf{Z}$. Then by moving the robot from $\mathbf{P}_1$ to $\mathbf{P}_2$, we obtain the movement $\mathbf{A}$ for the *camera* and $\mathbf{B}$ for the robot *tool*. Recalling that we can estimate $\mathbf{C}_1$ and $\mathbf{C}_2$ from observing a checkerboard with a camera and $\mathbf{Y}_1$ and $\mathbf{Y}_2$ directly from the robot, we are able to obtain $\mathbf{A}$ and $\mathbf{B}$ quite easily. This can be formulates as a non-linear optimization problem, but it is prone to local minima and finding the optimal solution is not straight forward [66, 67, 68].

Obviously, since we cannot view images from the LED point of view we cannot use hand-eye calibration for calibrating the robot responsible for moving the light source. In Contribution C we noticed a slight misalignment of the highlights in the samples, and being able to perform hand-eye calibration for the light source, would greatly benefit the setup as a whole. With the light position calibration method in Contribution E, we might be able to perform hand-eye calibration of the LED by observing the shadows of a known shape.

3.3 Discussion and Further Improvements

The list of improvements we would like to make is long. The immediately obvious ones would be to apply the newly developed method in Contribution E to obtain the data to do hand-eye calibration for the light source. Next would be to use the same method to automatically estimate the pose of the sample scanned, and correct robot positions, just by supplying the CAD model of the sample. The ability to finely control position of light and camera in relation to the sample could be very useful for researching new methods.

It could be interesting to upgrade the system to perform photometric stereo, but it would be very slow in its current version, with just one light source that would have to be moved. But despite the slow acquisition time, it is versatile enough to be used to investigate the optimal positioning of light sources and cameras for a static setup with a specific product, a good use case for inline visual quality control systems in industry.

The original plan was to use two Universal Robots UR5 to move the camera and light source, as done in the gantry based gonioreflectometers in Table 2.1. We found that using 6 DoF robot arms for dense hemispherical sampling, requires movement in the full working area of the robots, which leads to a ”singularity hell”. Meaning that the robots often will encounter a singularity in their IK model, causing the robot to do undefined behavior, for example turning a joint 360° to move 0.1mm to the right. Whenever the robot encounters a singularity, it compensates by taking a detour at infinite speed in joint space to keep a constant speed in cartesian space. This often leads to cables to be torn over and collisions with whatever is within reach.

The IK singularities that we encountered doing motion-planning, proved to be cumbersome to work around. This is one thing that gantry systems do not suffer from, as their mechanical parts are aligned with a spherical coordinate system. Usually specializing in solving a single task well, is at the cost of versatility. We seek to have a versatile instrument, that we can use to aid in multiple research topics, as demonstrated in this thesis.

Making proper motion sequences for our measurements, was by far the biggest issue we had in regards to using 6-axis robot arms. Having to manually account for self-collisions, singularities and joint-limits was a very cumbersome task, that neither ABB nor UR provided any good software solutions to solve. This is not a new problem, as singularities in IK is a well known problem and there are many libraries to solve both IK and motion planning. But at the time we started, neither ABB nor UR were willing to provide us with the kinematic model or the calibrated dimensions of the robots, which is required to obtain any useful accuracy. This in order to achieve any accuracy better than a few millimeters, we had to use the systems shipped with the robots. Universal Robots now officially provide its users with access to the internal calibration file, allowing the use of third party IK solvers and motion planners.

Given that Universal Robots now share the robot calibration, an obvious choice of control platform would be to use Robot Operating System (ROS) [69]. ROS is an extensive framework for communicating between subsystems, made specifically to control automated systems. One of such subsystems for ROS is their RViz and IK solvers and motion planning software. At the time we started the project, it was not possible to control the robots reliably using the ROS framework, but at the current state of ROS, if we were to start over, we would build everything around ROS.

CHAPTER 4
Contributions

In this chapter we summarize the contributions from this thesis and put them into context of the research goals, see Section 1.3. For the full details on the contributions, we refer the individual papers Contributions A to G. All of the contributions are related to problems encountered when trying to acquire or compare object appearance in an industrial setting. Each setup has been used in different contributions, where they have served either as a core/critical component in the contribution or as a tool supporting the exploratory research, idea generation, method development and analysis of results. An overview over the relation between contributions and instruments can be seen in Figure 4.1.

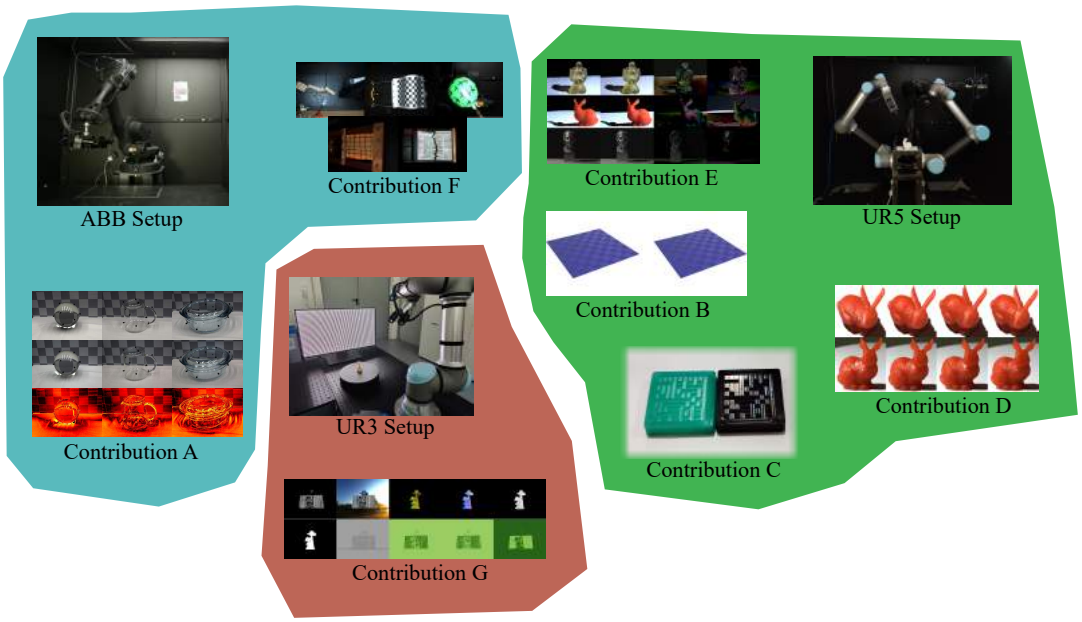


Figure 4.1: An overview over the relation between the instruments and the different contributions.

In Chapter 1 we discussed the industrial applications of being able to measure and compare appearance in a production setting. The contributions also address topics which current state of the art 3D acquisition methods struggle with, e.g. structured light scan-

ners struggle with scanning metals, glass, and polymers, which are all materials that are heavily used in industry. Further we address the application of using computer graphics methods for controlling the appearance of 3D prints. This is related to closing the loop, as we would like to change parameters in the productions based on measurements.

We start by discussing Contribution C where we quantify the perceived contrast of binary surface patterns, made by controlling the surface micro-structure in the injection molding process. We will briefly describe the surface structures and their characteristics and then address how we used the *UR5 Setup* to estimate the surface radiometry of different regions on the surface. This leads us to Contribution B, where we developed a BRDF model to create physically accurate renderings of these ridged micro-structures. We compared the response of our BRDF model with measurements of real samples made by [70]. This gives us instrumentation to measure the appearance of samples and the methodology to create renderings of the micro-milled ridged structures, the basic tools needed to investigate the creation of a digital twin.

This leads to the topic of creating digital twins, the digital twin is different from the normal digital model as the digital twin implies that the parameters are identical to, or obtained from, the physical twin. In Contribution A we develop a digitization pipeline around the *ABB Setup*, to acquire a digital twin and perform pixel-wise comparison to photographs of its physical twin. The ability to reliably position and re-position the camera at a later time, was crucial to creating the pipeline. One of the challenges in this study was placing the digital geometry at the correct pose in the virtual representation. We addressed this in Contribution E, where we developed a method to estimate the pose of an object of arbitrary shape and material, as long as a 3D mesh, a photograph of the object, segmentation and a camera calibration is available. Quite strong requirements, but in in an industrial setting, the environment can often be controlled to allow easy segmentation and an intrinsic camera calibration is easily obtained.

In Contribution A we also found that we needed af Computed Tomography (CT) scanner to obtain the 3D geometry of glass objects, which is a very expensive requirement. This led to Contribution G, which build on the work by Stets et al. [71]. Here, we decided to look into the use of polarization cameras to better capture the geometry. As in the work of Stets et al. [71], we needed training data for a network and implemented a rendering tool supporting polarization ray tracing. This tool renders the images as they would appear in polarization cameras. We used this engine to generate a dataset of polarized images of glass objects, that we then used to train a CNN to estimate the 3D geometry.

If we were to measure appearance in industries working with deformable objects, e.g. meat or clothing, we would need to obtain proper geometry first. This ties Contribution F to the geometric part of appearance, where we developed a dataset to benchmark state of the art non-rigid structure from motion methods. We wanted to evaluate the performance of Non-Rigid Structure from Motion (NRSfM) methods with structured light scanning as a reference. We need to have a better knowledge of the geometric errors in

NRSfM methods, if we want to use them to extend the methods from Contributions A and E to quantify the appearance of deformable objects. Thus the contribution here is a dataset and benchmark to help researchers in the NRSfM community when developing new methods.

Lastly in Contribution D, we looked into using computer graphic methods to control the appearance of objects 3D printed using a Stereolithography Apparatus (SLA). The UR5-Setup was heavily used in qualitative appearance verification throughout the development, but also for the final evaluation of the produced appearance using the inverse rendering methods that later led to Contribution E.

This links the parameters of computer graphics models to machine parameters to control surface finish on a 3D printed object, and brings us a step closer towards closing the loop.

4.1 Modelling and Measuring the Appearance of Surface Micro-Structures

In this section we will go over Contributions B and C, which were carried out in collaboration with a research group from the Department of Mechanical Engineering at DTU. They were working on embedding binary patterns, like a datamatrix or QR code, into the surface of injection-molded polymers. The approach was to control the radiometric response by modifying the micro-structures to obtain dark and light areas, without the use of paint or dyes. This were done by milling ridged micro-structures into steel inserts, which was then used for injection-molding the polymer samples. In Figure 4.2 we can see two of the measured samples and an example datamatrix code representing the text "DTU". The binary pattern comes from using either parallel or perpendicular ridges to direct the light towards or away from the viewer, giving the light and dark areas.

Figure 4.2: Left: Image showing two datamatrix samples used in Contribution C. They are made from injection molding with the micromilled ridged structures. Both specimens are translucent, but the green sample is see through, where the black sample have a higher absorption. This strongly influences the contrast, as can also be seen in the image. Right: Datamatrix code for "DTU".

Using binary codes to embed information into the part, be that either lot number, production parameters or information for the end-user, is a way to fix the information to the part. Being able to do this as a mold modification, allows production companies to embed this information at part creation, without adding additional steps to their production pipeline. This sees its usefulness for high-volume production, as modifying the pipeline is expensive.

In the work by Regi et al. [70] they used a modified microscope with and external LED and a turntable to measure the contrast of surface micro-structures. They wanted to improve their acquisition method and ideally have it automated. This was an early application of our *UR5 Setup*, which we used for measuring the perceived contrast of micro-ridged surfaces in Contribution C.

The iterative process of designing micro-structures for the machining inserts and making

injection molded samples is expensive and a cumbersome process. Thus, it was interesting to investigate the possibilities to develop a BRDF model of the ridge micro-structure and perform functionality experiments on renderings, to find the ridge parameters giving the best contrast between sample rotation. This is what let to Contribution B, where we developed an analytic BRDF model to model the an-isotropic behaviour of the ridged micro-structures. Renderings of the final BRDF model applied to a plane, can be seen in Figure 4.3. For the specifics of the ridged BRDF model, we refer the reader to the paper of Contribution B.

Modelling the ridged micro-structures

We find that having a BRDF model of the ridge micro-structures, is beneficial in two ways. One is finding the optimal parameters for the ridge structure, to achieve optimal contrast. Second is being able to create renderings, that we can compare with photographs for quality control.

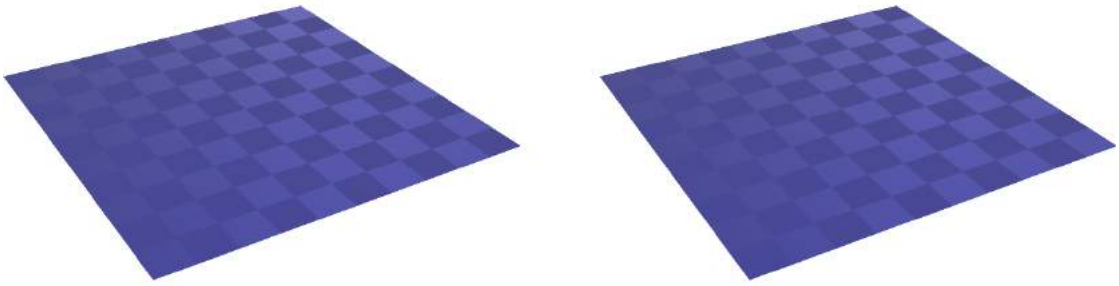


Figure 4.3: Renderings of a plane with the ridge BRDF model from Contribution B, showing the an-isotropic features from rotating the sample 90°. The right image is rotated 90° around the plane normal compared to the left, showing the desired functionality, that the binary pattern is inverted. It is easiest to see in the top corner.

For a qualitative evaluation of the BRDF model, we can see in Figure 4.3, that the rotation of 90° around the surface normal, yields the expected functionality: That dark checkers turn light and light checkers turn dark.

When comparing predictions made with our ridge BRDF model against measurements from the real world, provided by Regi et al. [70], we can see in Figure 4.4 that the trends align pretty well, indicating that our model is suitable for modeling the ridge micro-structures. Only when looking at the ridge angle θ_r we see a large deviation at 10°. We suspect the reason for the large deviation is due to inaccuracies of the physical samples, which were made using early versions of the micro-milling process and the

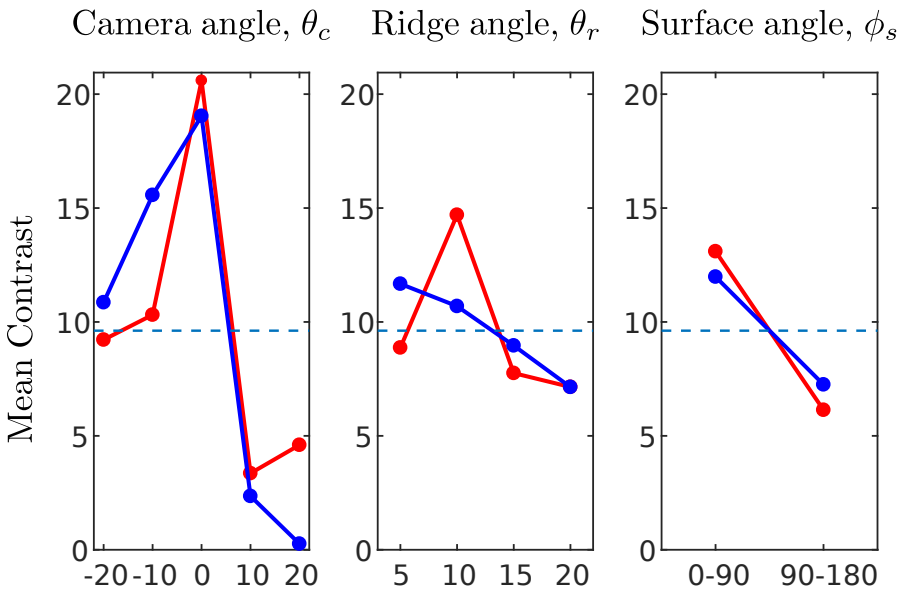


Figure 4.4: Measurements of mean contrast between 0° and 90° rotation around the surface normal. Comparison between predictions from our ridge BRDF model (blue) vs. measurements from [70] (red). The figure is from Contribution B.

observed issues related to the quality of the micro-structures. This resulted in small metal particles in the polymer samples, as seen in Figure 4.5, as well as many smaller defects in the ridge structure, which led to many undesired effects such as diffraction of light and significant geometrical deviations from the ridge edge structure. Thus it is not surprising that the comparisons between measurements and simulations differ a bit.

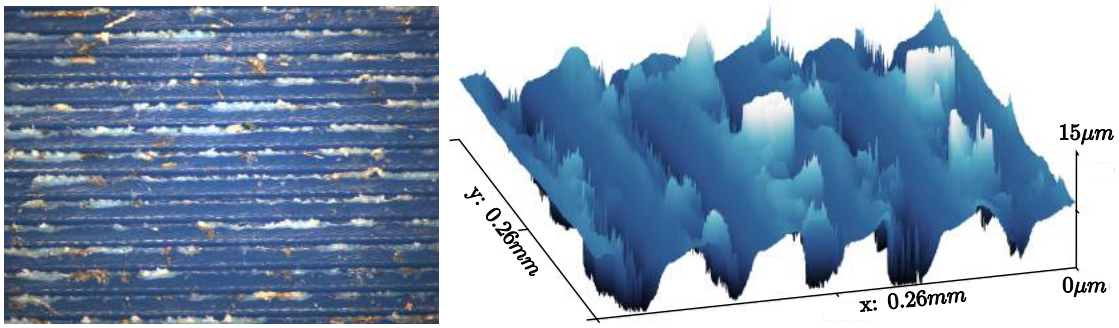


Figure 4.5: Left: Microscope image of a ridge sample similar to the one used for verification in Contribution B and Figure 4.4. Some of the ridge edges have significant defects, and the amount of small metal particles is quite high. We suspect that is a contributing factor in the deviation between BRDF model and measurements. Right: 3D visualization of a surface height map, acquired from a 3D laser confocal microscope. Both images are from Contribution B.

Measuring the functionality of surface micro-structures

It was of interest to develop an improved data acquisition for these samples, due to the development of new ridge micro-structure samples, and a significant amount of different samples. We automated this task using our *UR5 Setup* and were able to make an automated acquisition pipeline and data processing based on an initial user assisted sample localization method. This is published in Contribution C.

In Contribution C we used two orientations of ridge structure to direct the light in different directions, using either parallel or perpendicular ridges to create a bright or dark region, see Figure 4.2. We then looked into measuring the contrast, and thus the functionality of the surface. To evaluate the quality of the ridge structures, on a human perceivable scale, we used the *UR5 Setup* to control camera and light position in order to measure the radiometric response from the surface of the micro-structures.

The *UR5 Setup* has been crucial in this setup for a number of reasons. Versatility in the setup, endless possibilities for positioning of light source and camera.

Automation of acquisition. Each sample took approximately 90 minutes to acquire all data from a given rotation. A turntable would have been convenient for acquisition, but would have introduced an angular error around the surface normal. To avoid that, we decided to use LEGO bricks for mounting the samples, as the samples fit a 2x2 LEGO brick perfectly, this way we could neglect the rotational error around the surface normal. The LEGO mounting bracket, was securely mounted to the *UR5 Setup* frame.

Robot repeatability allowed for a ”calibrate once, run forever” approach. It made it possible for the operator to only select corner points in a subset of images, for the system to calculate sample position and make automatic homography extraction, even for views not annotated by the operator. We used this homography to segment the binary regions of the sample. The contrast could then be evaluated by comparing the black and white regions before and after a physical rotation of the sample as we see reported in the paper. It also made it possible to observe surface defects on a pr. block scale, by observing the changes in the signal w.r.t. camera movement.

Figure 4.6: Extracted homographies from the angles [10°, 20°, 30°, 40°, 50°]

By careful investigation its noticeable, from looking at the reflection in the images, that during these experiments the light source was not perfectly aligned 90° atop of the samples. See Figure 4.6, specifically. This misalignment most likely come from the fact

that the position of the light source could not be calibrated properly, and the positioning was based solely on the CAD models and manual adjustment.

If the methods developed in Contribution E had been available at the time, we could have done a hand-eye calibration of the light source, improving the precision of the light source positioning. And used the method for automatic pose estimation without the need for an operator to mark the corners. That would most likely significant improve both accuracy and precision of the measurements.

4.2 Comparing the Appearance of the Physical and Digital Twins

One of many challenges in Industry 4.0, is how to create a digital twin that has a meaningful relation to the physical twin. In production and particularly in visual quality assurance, it is interesting to use the digital twin to verify the quality of the physical part that is produced. To do this one first has to define a model to represent the twin, then a way to measure the parameters of the model and lastly a way to compare a measured model with a reference to measure deviation from the ideal part.

In Contribution A we address all of the aforementioned steps in a digitization pipeline. Even though the angle of the manuscript is focused on research applications, the concept is directly applicable in relation to comparing a physical and digital twin, as the pipeline represents all aspects of digitizing glass objects and a the environment around it, to perform comparisons between renderings and photographs, a result of this can be seen in Figure 4.7.

Previously we have been measuring the radiometric response on the surface of flat samples, and extending to objects of other shapes and materials, seems like the next natural step. An interesting area is glass objects, as they are inherently hard to work with, when using traditional computer vision methods. One of the reasons why glass is so hard to work with, is that it takes its appearance from the environment around it and generally doesn't absorb much light.

In Contribution A we build a digitization pipeline around the *ABB Setup*, for creating a digital twin of a full scene, using many different modalities for acquisition. A scene is defined here as the environment, objects, light sources and appearance models, everything that is needed to create a photo-realistic rendering. We then used the digital scene to create a rendering for pixel-wise comparison with a photograph of the same scene. While not being the original intend, the pipeline in Contribution A, contains all the steps needed to do visual quality control using data obtained from different modalities.

Objects in close proximity to the glass objects, were 3D scanned using the structured light scanner of the *ABB Setup*. While not directly addressed in the manuscript, the

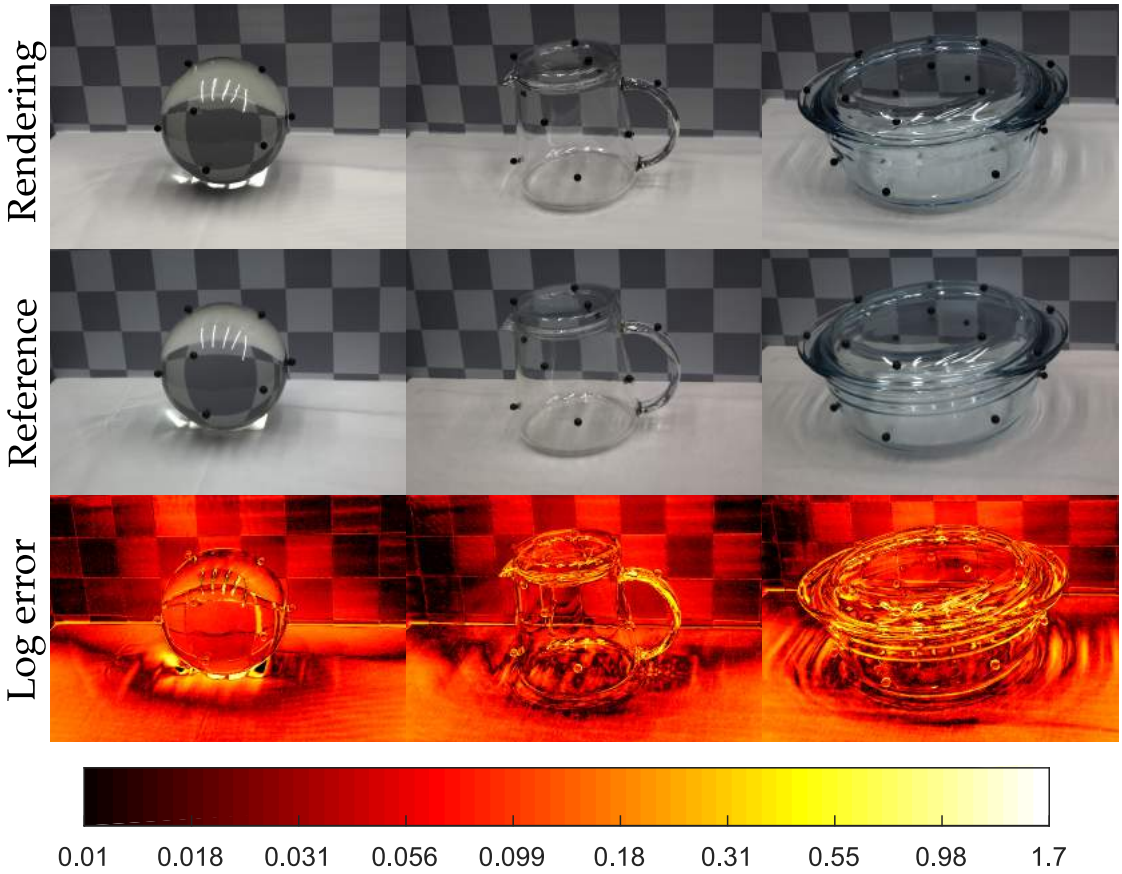


Figure 4.7: Comparison of rendering of digital scene with photograph. The log-error row is calculated by $\log_{10}(||\text{Rendering} - \text{Reference}||_2$. Figure from Contribution A.

ABB Setup is a core component. Making it possible to align all of these digital objects from different modalities robustly. The calibrate once and repeat forever approach, that was needed for this project to succeed, was only practically feasible because of the repeatability and versatility of the robot system. Besides 3D scanning of the scene and backdrop, it was also used for obtaining an environment map of the location of light sources, defining a common base frame for placing all objects in the virtual scene trough extrinsic camera calibration, and acquisition of the BRDF of the cloth and backdrop. Common for all of these are that they rely on the robots ability to reliably position and re-position itself in world space.

We also found the absorption coefficients, for the glass objects using the digitization pipeline and a rendering engine to perform *analysis by synthesis* also known as *inverse rendering*. The digitization pipeline presented in Contribution A makes it possible to use the Root Mean Squared Error (RMSE) distance between a rendered image and the photograph as a cost-function for optimization methods. Opening up the possibilities to estimate many different parameters solely based on their influence on the comparison

of the rendered image and its corresponding photograph. While we obtained quite good results, a downside to this approach is compensating for pixels not related to the object or parameters we optimize for. We encountered these issues, where the optimizer increased the absorption of the glass to compensate for a small increase of brightness in the tablecloth.

Another challenge we found in Contribution A, was that the pose estimation of the glass object had to be solved by physically gluing markers onto the glass objects that could both be seen and pose estimated by a camera as well as in the CT. This caused artifacts in the CT reconstructions and estimating the pose of these markers in the acquired images was not as robust as we had hoped. This was among other issues, a motivator for Contribution E.

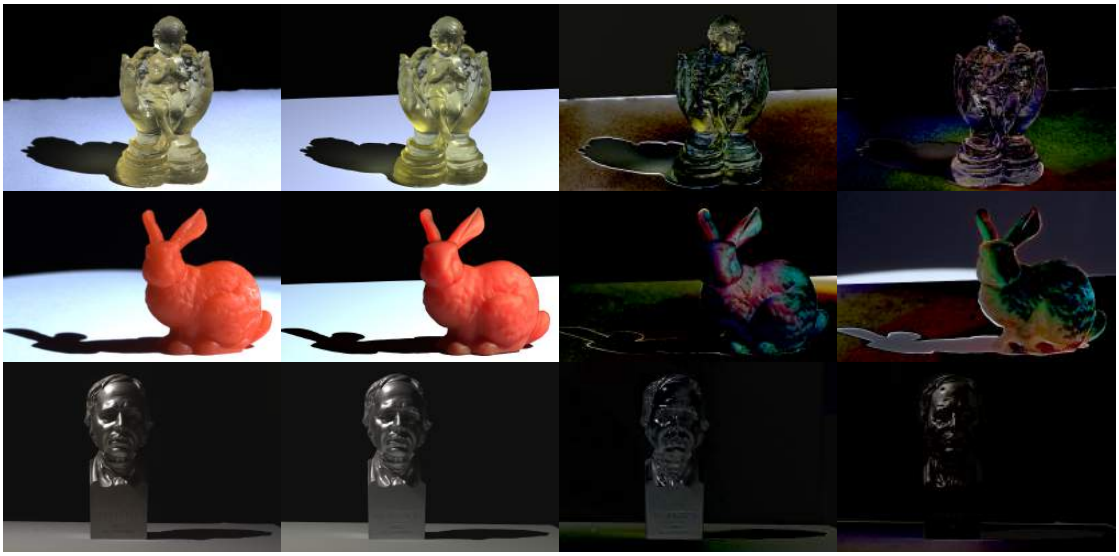
Estimating Object Pose and Appearance from Images

From the work in Contributions A, C and F we have encountered multiple issues that were complicated or impractical to solve. One of these issues is related to pose estimation, both of the object in the scene, but also the LED position in the *UR5 Setup*, this is what led to Contribution E. While we do not specifically cover the pose of glass objects in that contribution, we only need the segmentation of the glass object in order to estimate the pose. The groundwork leading to Contribution E is described in Contribution K, where we also looked into obtaining the material parameters by optimization rather than manual tweaking as was done in Contribution E.

We used the *UR5 Setup* to obtain images of the 3D printed Stanford Bunny from multiple positions under controlled lighting. An useful feature of using the robots, is a quite good initial guess for camera and light pose. Our pose estimation is based on CAD models and image segmentations. For obtaining the pose from images our first approach in Contribution K, was to use the binary operation exclusive or (XOR) as a loss function to define the overlap between two regions. We then minimized the loss using the Nelder-Mead minimization algorithm [72]. In Contribution E we improved this by using object silhouettes and Hu's moment invariants [73] with the Levenberg-Marquardt [74, 75] minimization method.

For obtaining the actual BxDF parameters we have a few options available, we have manual tweaking, which is a slow and tedious process. Then we have inverse rendering techniques in its broadest term, making the optimization automated, but generally lack proper gradients for the input parameters. Lastly, we have differentiable rendering, which is a subset of inverse rendering techniques that focus on having well defined gradients throughout the rendering process [57].

We investigated the general usefulness of Contribution E by using it together with readily



photograph (x) rendering (y) $(\max(y-x,0))^{1/\gamma}$ $(\max(x-y,0))^{1/\gamma}$

Figure 4.8: Pixelwise comparison of rendered images and photographs. Row-wise starting from the top, we have a rough transparent cupped angel figurine. Middle, a rough translucent 3D printed Stanford Bunny [76] and an aluminium bust of H. C. Ørsted. $\gamma = 2.2$. The images are from Contribution E, where the two right columns are gamma corrected versions of similar images found in the manuscript.

available rendering software, Blender[1] as Open Source and KeyShot[2] as Commercial, to create renderings that would be pixelwise comparable to photos taken with a less controlled setup than the *UR5 Setup*. We then used this comparison to manually adjust the parameters of the Principled Bi-directional Scattering Distribution Function (BSDF) [77, 78, 79] appearance model, this would arguable give better results if estimated using inverse rendering techniques.

In Contribution K we looked into finding the BxDF parameters from the images using inverse rendering. For this we found that using CIELAB color space worked best in the optimization pipeline. For the appearance model we used a directional dipole mode [80] and estimated the parameters by minimizing the difference in pixel intensities between a rendering and the photograph using the Nelder-Mead minimization method [72]. The parameters used to render the bunny in Figure 4.8 was obtained using this method, where the other entries were obtained by manual tweaking.

As seen in Figure 4.8 we do not use the absolute difference for visualizing the error, instead we look at the negative and positive errors separately clamping at 0. This help us interpret the direction to change the parameters to obtain a better rendering. This is

[1] www.blender.org
[2] www.keyshot.com

a simple tool that makes life much easier for manually tweaking parameters. Tweaking a parameter that causes the error to fluctuate around 0, would not be visible when looking at absolute error, while looking at the max images one would immediately see that the sign just flips, indicating the influence of the change. Further we apply a gamma mapping of $\gamma = 2.2$ to the error for visualization, as it makes it easier to identify changes.

One of the practical limitations of Contribution E for automated pose estimating objects with complex materials, is the required segmentation of object, shadow and background. This is still challenging to obtain for materials that take their appearance from their surroundings, such as low-absorbing transparent objects like glass and mirror-like smooth metal surfaces. Fortunately, neural networks such as [71, 81] and our Contribution G show promising results in making segmentations of these materials, increasing the practical usefulness of Contribution E as an automated approach for pose estimating objects of complex materials.

This is ultimately what is interesting for the industrial partners involved in this project: Being able to define a format to describe the appearance of their product and estimate the parameters from images taken by our instruments. And then compare their physical part with a digital version and quantify the visual deviation between the two. We have developed instrumentation that assisted us in the development of our contributions. Combining the methodology developed in Contribution E with the instrumentation developed in Contribution C we are able to acquire the data needed to generate the ideas for and to evaluate new methods.

4.3 Controlling the Surface Roughness in 3D printing

Our work with estimating the appearance of 3D objects led us into the field of trying to produce parts, where we introduced surface noise to influence the surface appearance of objects made with 3D printing, specifically SLA printing. In Contribution D we found that we could use computer graphics methods to modify the signal that was fed to the projector in such a system, to influence the printed surface roughness. The *UR5 Setup* was used heavily to inspect the parts and as a tool to verify and measure the appearance of our produced parts.

This was investigated using confocal microscopes able to measure the microstructures in the surface and as well as the *UR5 Setup*. Using the confocal microscopes was very time consuming, taking 3-5 hours for each scan, while only being able to scan a around 100µm. This was important for observing the appearance as geometry. But we as humans, cannot see features on that scale and thus we found a need for an instrument for evaluating the appearance on a human perceptible scale. This gave rise to some questions on as how can we evaluate that what we observe with the camera is in fact what we want it to be? This is where we started to look into using the robot setup in combination with inverse rendering techniques to perform pixel wise comparisons between produced parts

and their digital twin.

standard sinusoid noise ($A = 0.625$) noise ($A = 3$)

Figure 4.9: Images of the 3D printed Stanford Bunny, corrected with different versions of surface roughness models, taken under same lighting and camera conditions, in the UR5-Setup. Images are from Contribution D

When developing BRDF models in Contribution B and Contribution D the *UR5 Setup* has been of great use to allow investigations and comparing of renderings with real world measurements. By using the *UR5 Setup* we were able to obtain images from the exact same set of angles and light configurations, for multiple objects, see Figure 4.9. This allow us to still be able to perform pixel wise comparisons of the image. This proved to be very useful for the development of the methods, designing experiments and evaluating results. And while we only show a small subset of the produced samples in Contribution D, we produced many more samples as seen in Figure 4.10. While not being the only instrument used for inspection, the ability to use an automated setup, as the *UR5 Setup*, to compare a bunch of samples under the same conditions was very useful.

4.4 A Benchmark and Dataset for NRSfM Methods

If we are to extend these methods to deformable objects, where the ground truth geometry is not known, e.g. making sensory systems for surgery robots or meat quality assurance, then we need to obtain the geometry first. We already showed that we have the methodology to do this for known shapes of rigid objects, so the focus is now, how we can obtain the geometry of non-rigid objects. The concept of acquiring geometry of deformable objects using camera motion is also known as Non-Rigid Structure from Motion (NRSfM) methods, and there are many applications such as, surgery robotics, self driving cars and as sensory input for industrial automation. In Contribution F we

Figure 4.10: Image showing the large amount of different samples of various shapes and appearance, produced to reach the final results in Contribution D.

developed a dataset and a benchmark, which we used to analyze the NRSfM methods, to help identify research directions in the field.

One of the problems in the field of NRSfM is the difficulty to generate accurate reference data. Most high precision 3D scanners takes a few seconds to make a 3D scan and high Frames Per Second (FPS) 3D scanners, like the Microsoft Kinect, are very inaccurate. This forces researchers to use low accuracy scanners for their reference data.

The idea for our NRSfM dataset was to do stop motion, but instead of taking a single image, we acquired a full 3D scan from multiple views, pr. frame. To do this we developed a series of small robots, see Figure 4.11, capable of stepping through the non-rigid deformations, i.e. change the pose of the robot slightly between each frame, and keep steady for acquisition. The robots were positioned inside the *ABB setup*, see Chapter 3 for the details, to obtain full 3D scans pr. frame. The repeatability of the ABB robot made it possible to move the camera precisely, to multiple positions for 3D acquisition with sub-millimeter accuracy on positioning. This made it possible to only calibrate the system once, instead of for every camera position in every frame, which would have been infeasible. These small robots were designed to mimic a series of non-rigid deformations, that would occur either naturally or in an industrial setting.

Experimenting with different types of deformation and developing the actual animatronics, required a large amount of the experimentation, which is not shown in the paper. The previously most used datasets for NRSfM evaluation are based on human Motion Capture (MoCap) data, the CMU motion dataset [82], a flag dataset [83] and a synthetic dataset [84]. Multiple papers such as [85, 86] use the deformation of bending a piece of paper for qualitative evaluation, without ground truth, just to clarify that our choice of

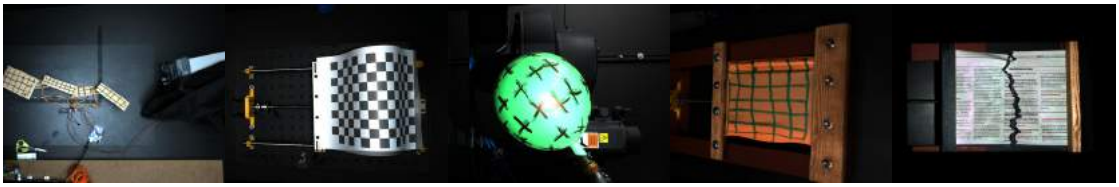


Figure 4.11: Images of the 5 different small robots made for the dataset in Contribution F. Each representing a different type of non-rigid deformation. Starting from left to right, Articulated, Bending, Deflation, Stretching, and Tearing.

deformations was influenced by previous choices of verification.

For the acquisition of ground truth data, it was not possible to use the same 3D scanner positions for all small robots, as they did not all occupy the same physical space. Thus there would have been empty frames, and most likely a bias introduced due to change in both camera motion and geometry simultaneously. We solved this by making virtual camera motions, based on the intrinsic parameters of the camera in the 3D scanner, and reprojecting the geometry into the virtual cameras. This allowed for complete separation of camera motion and geometry, while keeping the acquisition noise from the 3D scanner.

One of the strengths of the dataset is that, due to the nature of acquisition, it has realistic structured missing data due to occlusion. This is unlike most of the other datasets for NRSfM that rely on removing randomly selected points for missing data. As many of the methods rely on strong regularization factors and smoothing, they would perform well with randomly removed points. But when dealing with structured occlusion based missing data, we saw that most methods performed poorly.

The complete separation of geometry, camera motion and missing data makes it possible to perform a factorial analysis of the influence of the individual parameters on the performance of the NRSfM methods, this has to our knowledge not been done previously. According to our ANalysis Of VAriance (ANOVA) test, all the factors in our dataset have a significant influence on the reconstruction error, and as such this is a positive indication that the dataset does challenge the methods, and thus can be used as a tool for further research within NRSfM.

4.5 Obtaining 3D Information from Monocular Polarization Images

Taking a reference in the challenges experienced in our work in Contribution A, we found that having to use a CT scanner for scanning the geometry of glass objects is a very expensive prerequisite in order to reproduce the work. It was therefore of interest to be able to estimate the geometry of glass objects from images taken by a camera. This endeavour was started by Stets et al. [71], but while their results are promising, the

quality of the estimated depth maps are not accurate enough to be used for generating images. This led to Contribution G, as an extension of their work into using polarization cameras and using that for estimating the geometry.

In Contribution G we developed a rendering pipeline to perform physically accurate polarization raytracing that is directly comparable to real world images. We used it to render a dataset, see Figure 4.12, for estimating the geometry of glass objects using CNNs, both with VGG-16 and a hybrid General Adversarial Network (GAN) version. The results from this are shown in Table 4.1. The rendering pipeline is representing the digital version of the *UR3 setup* using a LCD screen, a turntable and a polarization camera. The concept is that the polarized light emitted by the screen, will interact with the glass object and each interaction will, due to effects described by the Fresnel equations, change the polarization. This change in polarization will be measured by the camera and polarization will change depending on the number of interfaces the ray interacts with before reaching the camera.

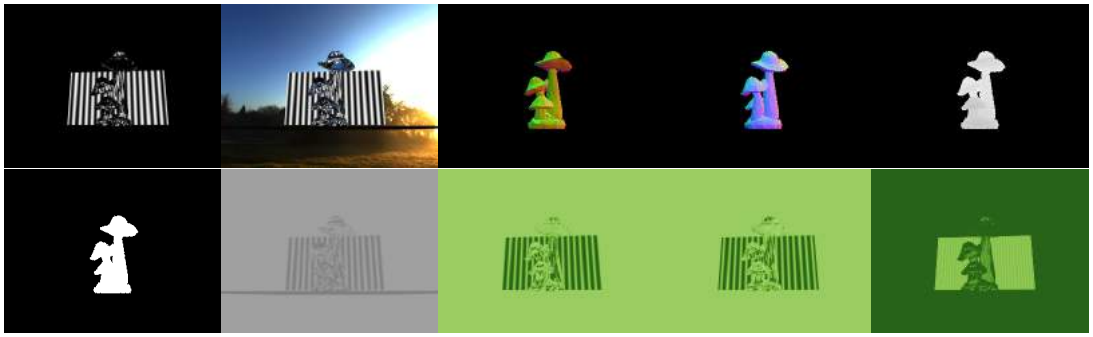


Figure 4.12: Showing an example output from our renderer. Here we render CAD model #53159 from the Thingi10k dataset. From top left we have: Rendering no environment, rendering with environment, front-facing normals, back-facing normals, depth, mask, s0, s1, s2, and s3. Figure from Contribution G

When developing the polarization ray tracing framework, we found that while in literature they are able to do polarization ray tracing for geometry estimation of transparent objects, no research was found on simulating a polarization camera for rendering. We made a polarization camera model by modelling the polarization mosaic pattern on the chip, similarly to the Bayer pattern for RGB cameras. Simulating such a camera requires more than just the filter pattern on the image sensor. It requires knowledge about the coordinate frames of which polarization is calculated in the simulated scene. This is important to know for the screen and the camera. The cameras coordinate frames are given as the angle of the polarization filter in each pixel. For the screen we made the assumption that the coordinate frame of the monitor was aligned to the output polarization, but this should be properly estimated for the optimal results.

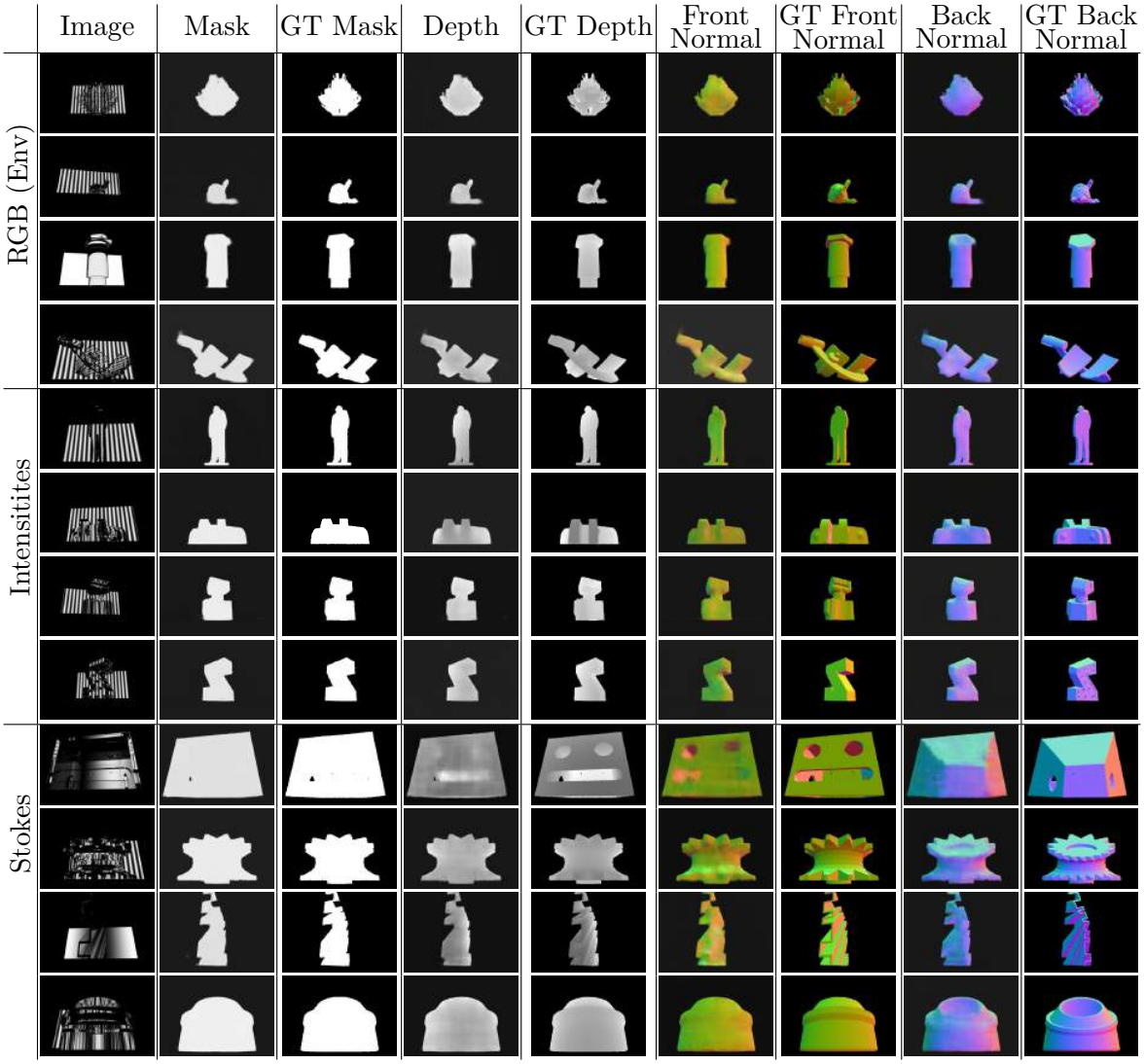


Table 4.1: Comparisons of mask, depth, normal and back face normal to GT across our three input modalities, RGB, Intensities and Stokes. For each modality we show four random models that demonstrate varying surface and geometry complexity. Table from Contribution G.

4.6 Future Work and Discussion

If the methods developed in Contribution E had been available at the time of Contribution C, we could have used hand-eye calibration of the light source, improving the precision of the light source positioning. Furthermore, the sample could have been automatically pose estimated, removing the need for an operator to mark the corners. That would give significant improvements in both accuracy and precision.

The primary challenge in applying Contribution E to pose estimate the glass object is that an image segmentation needs to be obtained, and that is (to our knowledge) not possible by conventional image analysis methods. However, Stets et al. [71] shows that the segmentation mask can be generated by using CNNs, and combined with our method in Contribution E we could estimate the pose of glass objects in images. A problem that could arise from this is caustics. Which breaks the assumption that a shadow is a dark flat color as it would be for non-transparent objects. This behaviour can already be seen in images of the cupped angel in Figure 4.8. With segmentation and pose estimation of marker-free glass objects, we could significantly improve the reassembly technique in Contribution A. The digitization procedure would then be non-invasive and the markers would then not generate issues in the CT scanning process and much better results would most likely be obtained.

CHAPTER 5
Conclusion

To sum up, the contributions from this PhD project is the development of an instrument Contribution C and methodology Contribution E to estimate the appearance of objects with unknown and arbitrary material but a known and arbitrary shape. We have developed the *UR5 Setup* with methodology as discussed in Chapter 3, is assisting researchers in their development of new methodology to quantify and evaluate the appearance of objects. The instrument has been used in Contributions C to E, both as a core instrument but also as an instrument used to evaluate and analyze results for idea generation and method evaluation. The *UR5 Setup* has been used by external researchers to evaluate results for their research projects.

We developed a BRDF model in Contribution B to represent the complex light interactions in ridged microstructures, which was difficult to model by previous BRDF models. The proposed BRDF model can be used for virtual experimentation to find optimal production parameters to obtain the best surface functionality.

In Contribution C we developed the *UR5 Setup* as described in Chapter 3, to estimate the surface radiometry of samples with engineered microsurfaces. We also identified areas for improvement, and developed methodology in Contribution E allowing us to in the future, make better measurements with the *UR5 Setup*. As it allows for automatic pose estimation of both the object and light source, which is currently difficult.

In Contribution A we developed a digitization pipeline for, but not limited to, glass objects around the *ABB Setup*. The pipeline demonstrates how to create a digital representation of a physical environment for use in comparing photographs to renderings of the digital reference model. This allows for a direct comparison of a digital and physical twin, and even a tool to generate the digital twin from a physical object. Further the pipeline can be used to evaluate the performance of new methods in each steps in the pipeline, making it useful for new research.

We found pose estimating objects for placing in a digital scenes, to be a particularly hard problem. In Contribution E we solved that problem to a large extend, relying only on CAD models and image segmentations for pose estimation. This could prove useful in making the pipeline in Contribution A for use cases where the destructive process of gluing on markers is not an option.

In Contribution D we used computer graphics principles to control the surface roughness of 3D prints to avoid aliasing artifacts. The *UR5 setup* with its ability to reproduce a previously used light-camera configuration was indispensable in this context when inspecting the macroscopic optical effects of actuating changes in the surface microstructure of the digital twin.

In Contribution F, we developed a dataset that captures a wider range of deformations and a more realistic representation of missing data, than previously available. Using this dataset we made a benchmark, that provided an exploration of the performance, trends and challenges of the current NRSfM methods and hopefully can act as a tool that will support fellow researchers in their research for future NRSfM methods.

In Contribution G we developed a rendering pipeline capable of rendering images as seen by polarization cameras. We used this renderer to generate data for training our Polarization Neural Network (PNN) that could generate masks, depth, front- and back-normal maps from images of glass objects.

With the contributions from this thesis, we believe that we have moved a step closer to bringing appearance into the digital twin, as we developed practical methods for estimating object pose and appearance by using combining computer vision with rendering techniques.

Bibliography

[1] Jonathan Dyssel Stets, Alessandro Dal Corso, Jannik Boll Nielsen, Rasmus Ahrenkiel Lyngby, Sebastian Hoppe Nesgaard Jensen, Jakob Wilm, Mads Emil Brix Doest, Carsten Gundlach, Eythor Runar Eiriksson, Knut Conradsen, Anders Bjorholm Dahl, Jakob Andreas Bærentzen, Jeppe Revall Frisvad, and Henrik Aanæs. "Scene reassembly after multimodal digitization and pipeline evaluation using photorealistic rendering". In: *Applied Optics* 56.27 (Sept. 2017), pp. 7679–7690. DOI: 10.1364/AO.56.007679.

[2] Andrea Luongo, Viggo Falster, Mads Emil Brix Doest, Dongya Li, Francesco Regi, Yang Zhang, Guido Tosello, Jannik Boll Nielsen, Henrik Aanaes, and Jeppe Revall Frisvad. "Modeling the Anisotropic Reflectance of a Surface With Microstructure Engineered to Obtain Visible Contrast After Rotation". In: *Proceedings of the IEEE International Conference on Computer Vision Workshops (ICCVW)*. Oct. 2017, pp. 159–165. DOI: 10.1109/ICCVW.2017.27.

[3] Francesco Regi, Mads Emil Brix Doest, Dario Loaldi, Dongya Li, Jeppe Revall Frisvad, Guido Tosello, and Yang Zhang. "Functionality characterization of injection moulded micro-structured surfaces". In: *Precision Engineering* 60 (Nov. 2019), pp. 594 –601. DOI: 10.1016/j.precisioneng.2019.07.014.

[4] Andrea Luongo, Viggo Falster, Mads Emil Brix Doest, Macarena Méndez Ribó, Eyþór Rúnar Eiríksson, David Bue Pedersen, and Jeppe Revall Frisvad. "Microstructure Control in 3D Printing with Digital Light Processing". In: *Computer Graphics Forum* 39.1 (2020), pp. 347–359. DOI: 10.1111/cgf.13807.

[5] Morten Hannemose, Mads Emil Brix Doest, Andrea Luongo, Søren Kimmer Schou Gregersen, Jakob Wilm, and Jeppe Revall Frisvad. "Alignment of rendered images with photographs for testing appearance models". In: *Applied Optics* 59.31 (Nov. 2020), pp. 9786–9798. DOI: 10.1364/AO.398055.

[6] Sebastian Hoppe Nesgaard Jensen, Mads Emil Brix Doest, Henrik Aanæs, and Alessio Del Bue. "A benchmark and evaluation of non-rigid structure from motion". In: *International Journal of Computer Vision* 129 (Dec. 2020). DOI: 10.1007/s11263-020-01406-y.

[7] Mads Emil Brix Doest, Stuart James, Alessio Del Bue, and Jeppe Revall Frisvad. "Reconstructing transparent glass objects from polarization". Unpublished Manuscript. 2021.

[8] Henrik Aanæs, Knut Conradsen, Alessandro Dal Corso, Anders Bjorholm Dahl, Alessio Del Bue, Mads Emil Brix Doest, Jeppe Revall Frisvad, Sebastian Hoppe Nesgaard Jensen, Jannik Boll Nielsen, Jonathan Dyssel Stets, and George Vogiatzis. "Our 3D vision data-sets in the making". In: *The Future of Datasets in Vision 2015: CVPR 2015 Workshop*. 2015.

[9] Jakob Wilm, Daniel González Madruga, Janus Nørtoft Jensen, Søren Kimmer Schou Gregersen, Mads Emil Brix Doest, Maria Grazia Guerra, Henrik Aanæs, and Leonardo De Chiffre. "Effects of subsurface scattering on the accuracy of optical 3D measurements using miniature polymer step gauges". In: *Proceedings of the 18th International Conference of the European Society for Precision Engineering and Nanotechnology (euspen 2018)*. June 2018, pp. 449–450.

[10] Andrea Luongo, Jeppe Revall Frisvad, Alessandro Dal Corso, Mads Emil Brix Doest, and Henrik Wann Jensen. "Building Vision-Based Predictive Appearance Models for 3D Printing". Published as a Technical Report in Andrea Luongos PhD Thesis. 2019.

[11] Jeppe Revall Frisvad, Søren Alkærsig Jensen, Jonas Skovlund Madsen, António Correia, Li Yang, Søren K. S. Gregersen, Youri Meuret, and Poul-Erik Hansen. "Survey of Models for Acquiring the Optical Properties of Translucent Materials". In: *Computer Graphics Forum* 39.2 (2020), pp. 729–755. DOI: 10.1111/cgf.14023.

[12] Douglas R. Wyman, Michael S. Patterson, and Brian C. Wilson. "Similarity relations for the interaction parameters in radiation transport". In: *Appl. Opt.* 28.24 (Dec. 1989), pp. 5243–5249. DOI: 10.1364/AO.28.005243.

[13] Jeppe Revall Frisvad, Niels Jørgen Christensen, and Henrik Wann Jensen. "Computing the Scattering Properties of Participating Media Using Lorenz-Mie Theory". In: *ACM SIGGRAPH 2007 Papers*. SIGGRAPH '07. Association for Computing Machinery, 2007, 60–es. DOI: 10.1145/1275808.1276452.

[14] Shuang Zhao, Ravi Ramamoorthi, and Kavita Bala. "High-Order Similarity Relations in Radiative Transfer". In: *ACM Trans. Graph.* 33.4 (July 2014). ISSN: 0730-0301. DOI: 10.1145/2601097.2601104.

[15] Richard Szeliski. *Computer Vision: Algorithms and Applications*. 2nd ed. 2020. URL: http://szeliski.org/Book.

[16] Richard Szeliski. *Computer Vision: Algorithms and Applications*. 1st ed. 2011. DOI: 10.1007/978-1-84882-935-0.

[17] Matt Pharr, Wenzel Jakob, and Greg Humphreys. *Physically Based Rendering: From Theory to Implementation*. 3rd. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 2016. ISBN: 0128006455.

[18] Erik Reinhard, Erum Arif Khan, Ahmet Oguz Akyz, and Garrett M. Johnson. *Color Imaging: Fundamentals and Applications*. USA: A. K. Peters, Ltd., 2008. ISBN: 1568813449.

[19] European Machine Vision Association. *EMVA 1288 - Standard for Characterization of Image Sensors and Cameras*. Tech. rep. European Machine Vision Association, 2016.

[20] Fred E Nicodemus. "Directional reflectance and emissivity of an opaque surface". In: *Applied optics* 4.7 (1965), pp. 767–775.

[21] Michael Weinmann and Reinhard Klein. "Advances in Geometry and Reflectance Acquisition (Course Notes)". In: *SIGGRAPH Asia 2015 Courses*. SA '15. Kobe, Japan: Association for Computing Machinery, 2015. DOI: 10.1145/2818143.2818165.

[22] Mirko Sattler, Ralf Sarlette, and Reinhard Klein. "Efficient and realistic visualization of cloth". In: *Rendering Techniques*. 2003, pp. 167–178.

[23] D Hünerhoff, U Grusemann, and A Höpe. "New robot-based gonioreflectometer for measuring spectral diffuse reflection". In: *Metrologia* 43.2 (Mar. 2006), S11–S16. DOI: 10.1088/0026-1394/43/2/s03.

[24] Akira Kimachi, Norihiro Tanaka, and Shoji Tominaga. "Development and calibration of a gonio-spectral imaging system for measuring surface reflection". eng. In: *Ieice Transactions on Information and Systems* E89-D.7 (2006), pp. 1994–2003. ISSN: 17451361, 09168532. DOI: 10.1093/ietisy/e89-d.7.1994.

[25] A. Höpe, T. Atamas, D. Hünerhoff, S. Teichert, and K.-O. Hauer. "ARGon3: "3D appearance robot-based gonioreflectometer" at PTB". In: *Review of Scientific Instruments* 83.4 (2012), p. 045102. DOI: 10.1063/1.3692755.

[26] Jiri Filip, Radomir Vavra, Michal Haindl, Pavel Zid, Mikulas Krupika, and Vlastimil Havran. "BRDF slices: Accurate adaptive anisotropic appearance acquisition". In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2013, pp. 1468–1473.

[27] Rasmus Ahrenkiel Lyngby, Jannik Boll Matthiassen, Jeppe Revall Frisvad, Anders Bjorholm Dahl, and Henrik Aanæs. "Using a Robotic Arm for Measuring BRDFs". In: *Scandinavian Conference on Image Analysis*. Springer. 2019, pp. 184–196.

[28] Gero Müller, Jan Meseth, Mirko Sattler, Ralf Sarlette, and Reinhard Klein. "Acquisition, synthesis, and rendering of bidirectional texture functions". In: *Computer Graphics Forum*. Vol. 24. 1. Wiley Online Library. 2005, pp. 83–109.

[29] Gero Müller, Gerhard H. Bendels, and Reinhard Klein. "Rapid Synchronous Acquisition of Geometry and BTF for Cultural Heritage Artefacts". In: *The 6th International Symposium on Virtual Reality, Archaeology and Cultural Heritage (VAST)*. Eurographics Association. Eurographics Association, Nov. 2005, pp. 13–20.

[30] Christopher Schwartz, Ralf Sarlette, Michael Weinmann, and Reinhard Klein. "DOME II: A Parallelized BTF Acquisition System." In: *Material Appearance Modeling*. 2013, pp. 25–31.

[31] JF Murray-Coleman and AM Smith. "The automated measurement of BRDFs and their application to luminaire modeling". In: *Journal of the Illuminating Engineering Society* 19.1 (1990), pp. 87–99.

[32] Gregory J Ward. "Measuring and modeling anisotropic reflection". In: *Proceedings of the 19th annual conference on Computer graphics and interactive techniques*. 1992, pp. 265–272.

[33] Kristin J Dana, Bram Van Ginneken, Shree K Nayar, and Jan J Koenderink. "Reflectance and texture of real-world surfaces". In: *ACM Transactions On Graphics (TOG)* 18.1 (1999), pp. 1–34.

[34] Stephen R Marschner, Stephen H Westin, Eric PF Lafortune, and Kenneth E Torrance. "Image-based bidirectional reflectance distribution function measurement". In: *Applied optics* 39.16 (2000), pp. 2592–2600.

[35] Kristin J Dana. "BRDF/BTF measurement device". In: *Proceedings Eighth IEEE International Conference on Computer Vision. ICCV 2001*. Vol. 2. IEEE. 2001, pp. 460–466.

[36] Wojciech Matusik, Hanspeter Pfister, Matt Brand, and Leonard McMillan. "A Data-Driven Reflectance Model". In: *ACM Transactions on Graphics* 22.3 (July 2003), pp. 759–769.

[37] Akira Kimachi, Norihiro Tanaka, and Shoji Tominaga. "A goniometric system for measuring surface spectral reflection using two robot arms". eng. In: *Cgiv 2006 - 3rd European Conference on Colour in Graphics, Imaging, and Vision, Final Program and Proceedings* (2006), pp. 378–381.

[38] Michael Holroyd, Jason Lawrence, and Todd Zickler. "A Coaxial Optical Scanner for Synchronous Acquisition of 3D Geometry and Surface Reflectance". In: *ACM SIGGRAPH 2010 Papers*. SIGGRAPH '10. Los Angeles, California: Association for Computing Machinery, 2010. ISBN: 9781450302104. DOI: 10.1145/1833349.1778836.

[39] Jannik Boll Nielsen, Jonathan Dyssel Stets, Rasmus Ahrenkiel Lyngby, Henrik Aanæs, Anders Bjorholm Dahl, and Jeppe Revall Frisvad. "A variational study on BRDF reconstruction in a structured light scanner". In: *Proceedings of the IEEE International Conference on Computer Vision Workshops*. 2017, pp. 143–152.

[40] X-Rite. *TAC7*. https://www.xrite.com/categories/appearance/tac7. Dec. 2020.

[41] Heinz Christian Steinhausen, Dennis den Brok, Sebastian Merzbach, Michael Weinmann, and Reinhard Klein. "Data-driven Enhancement of SVBRDF Reflectance Data." In: *VISIGRAPP (1: GRAPP)*. 2018, pp. 273–280.

[42] Miika Aittala, Tim Weyrich, Jaakko Lehtinen, et al. "Two-shot SVBRDF capture for stationary materials." In: *ACM Trans. Graph.* 34.4 (2015), pp. 110–1.

[43] Miika Aittala, Timo Aila, and Jaakko Lehtinen. “Reflectance Modeling by Neural Texture Synthesis”. In: *ACM Trans. Graph.* 35.4 (July 2016). DOI: 10.1145/2897824.2925917.

[44] Xiao Li, Yue Dong, Pieter Peers, and Xin Tong. “Modeling Surface Appearance from a Single Photograph Using Self-Augmented Convolutional Neural Networks”. In: *ACM Trans. Graph.* 36.4 (July 2017). DOI: 10.1145/3072959.3073641.

[45] Zhengqin Li, Zexiang Xu, Ravi Ramamoorthi, Kalyan Sunkavalli, and Manmohan Chandraker. “Learning to Reconstruct Shape and Spatially-Varying Reflectance from a Single Image”. In: 37.6 (Dec. 2018). DOI: 10.1145/3272127.3275055.

[46] Zhengqin Li, Kalyan Sunkavalli, and Manmohan Chandraker. “Materials for Masses: SVBRDF Acquisition with a Single Mobile Phone Image”. In: *Proceedings of the European Conference on Computer Vision (ECCV)*. Sept. 2018.

[47] Mark Boss, Varun Jampani, Kihwan Kim, Hendrik Lensch, and Jan Kautz. “Two-shot Spatially-varying BRDF and Shape Estimation”. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2020, pp. 3982–3991.

[48] Shen Sang and Manmohan Chandraker. “Single-Shot Neural Relighting and SVBRDF Estimation”. In: *European Conference on Computer Vision*. Springer. 2020, pp. 85–101.

[49] R. L. Cook and K. E. Torrance. “A Reflectance Model for Computer Graphics”. In: *ACM Trans. Graph.* 1.1 (Jan. 1982), pp. 7–24. DOI: 10.1145/357290.357293.

[50] V. Havran, J. Filip, and K. Myszkowski. “Perceptually Motivated BRDF Comparison using Single Image”. In: *Computer Graphics Forum* 35.4 (2016), pp. 1–12. DOI: 10.1111/cgf.12944.

[51] Donald P Greenberg, Kenneth E Torrance, Peter Shirley, James Arvo, Eric Lafortune, James A Ferwerda, Bruce Walter, Ben Trumbore, Sumanta Pattanaik, and Sing-Choong Foo. “A framework for realistic image synthesis”. In: *Proceedings of the 24th annual conference on Computer graphics and interactive techniques*. 1997, pp. 477–494.

[52] Cindy M. Goral, Kenneth E. Torrance, Donald P. Greenberg, and Bennett Battaile. “Modeling the Interaction of Light between Diffuse Surfaces”. In: *SIGGRAPH Comput. Graph.* 18.3 (Jan. 1984), pp. 213–222. ISSN: 0097-8930. DOI: 10.1145/964965.808601.

[53] Gary W. Meyer, Holly E. Rushmeier, Michael F. Cohen, Donald P. Greenberg, and Kenneth E. Torrance. “An Experimental Evaluation of Computer Graphics Imagery”. In: *ACM Trans. Graph.* 5.1 (Jan. 1986), pp. 30–50. DOI: 10.1145/7529.7920.

[54] H Rushmeier, G Ward, C Piatko, P Sanders, and B Rust. “Comparing real and synthetic images: Some ideas about metrics”. eng. In: *Eurographics* (1995), pp. 82–91.

[55] SN Pattanaik, JA Ferwerda, KE Torrance, and D Greenberg. “Validation of global illumination simulations through CCD camera measurements”. eng. In: *Fifth Color Imaging Conference: Color Science, Systems, and Applications* (1997), pp. 250–253.

[56] Guillaume Loubet, Nicolas Holzschuch, and Wenzel Jakob. “Reparameterizing discontinuous integrands for differentiable rendering”. In: *Transactions on Graphics (Proceedings of SIGGRAPH Asia)* 38.6 (Dec. 2019). DOI: 10.1145/3355089.3356510.

[57] Merlin Nimier-David, Delio Vicini, Tizian Zeltner, and Wenzel Jakob. “Mitsuba 2: A Retargetable Forward and Inverse Renderer”. In: *Transactions on Graphics (Proceedings of SIGGRAPH Asia)* 38.6 (Dec. 2019). DOI: 10.1145/3355089.3356498.

[58] Rasmus Jensen, Anders Dahl, George Vogiatzis, Engin Tola, and Henrik Aanæs. “Large scale multi-view stereopsis evaluation”. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2014, pp. 406–413.

[59] Carsten Reich, Reinhold Ritter, and Jan Thesing. “White light heterodyne principle for 3D-measurement”. In: *Sensors, Sensor Systems, and Sensor Data Processing*. Ed. by Otmar Loffeld. Vol. 3100. International Society for Optics and Photonics. SPIE, 1997, pp. 236 –244. DOI: 10.1117/12.287750.

[60] Deutsches Institut für Normung. *VDI 2634: Optical 3-D measuring systems. Optical systems based on area scanning*. Tech. rep. Deutsches Institut für Normung, 2012.

[61] OpenCV. *Open Source Computer Vision Library v. 4.3*. 2020.

[62] Z. Zhang. “A flexible new technique for camera calibration”. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 22.11 (2000), pp. 1330–1334. DOI: 10.1109/34.888718.

[63] Sergio Garrido-Jurado, Rafael Munoz-Salinas, Francisco José Madrid-Cuevas, and Rafael Medina-Carnicer. “Generation of fiducial marker dictionaries using mixed integer linear programming”. In: *Pattern Recognition* 51 (2016), pp. 481–491.

[64] Francisco J Romero-Ramirez, Rafael Muñoz-Salinas, and Rafael Medina-Carnicer. “Speeded up detection of squared fiducial markers”. In: *Image and vision Computing* 76 (2018), pp. 38–47.

[65] M.W. Spong, S. Hutchinson, and M. Vidyasagar. *Robot Modeling and Control*. Wiley, 2005. ISBN: 9780471649908. URL: https://books.google.dk/books?id=A0OXDwAAQBAJ.

[66] Radu Horaud and Fadi Dornaika. “Hand-eye calibration”. In: *The international journal of robotics research* 14.3 (1995), pp. 195–210.

[67] F. Dornaika and R. Horaud. “Simultaneous robot-world and hand-eye calibration”. In: *IEEE Transactions on Robotics and Automation* 14.4 (1998), pp. 617–622. DOI: 10.1109/70.704233.

[68] Klaus H Strobl and Gerd Hirzinger. "Optimal hand-eye calibration". In: *2006 IEEE/RSJ international conference on intelligent robots and systems*. IEEE. 2006, pp. 4647–4653.

[69] Morgan Quigley, Ken Conley, Brian Gerkey, Josh Faust, Tully Foote, Jeremy Leibs, Rob Wheeler, and Andrew Y Ng. "ROS: an open-source Robot Operating System". In: *ICRA workshop on open source software*. Vol. 3. 3.2. Kobe, Japan. 2009, p. 5.

[70] F. Regi, J. B. Nielsen, D. Li, Y. Zhang, J. R. Frisvad, H. Aanaes, and G. Tosello. "A method for the characterization of the reflectance of anisotropic functional surfaces". eng. In: *Surface Topography-metrology and Properties* 6.3 (2018). DOI: 10.1088/2051-672X/aac373.

[71] Jonathan Dyssel Stets, Zhengqin Li, Jeppe Revall Frisvad, and Manmohan Chandraker. "Single-Shot Analysis of Refractive Shape Using Convolutional Neural Networks". In: *IEEE Winter Conference on Applications of Computer Vision (WACV 2019)*. 2019, pp. 995–1003.

[72] Fuchang Gao and Lixing Han. "Implementing the Nelder-Mead simplex algorithm with adaptive parameters". In: *Computational Optimization and Applications* 51.1 (2012), pp. 259–277.

[73] Ming-Kuei Hu. "Visual pattern recognition by moment invariants". In: *IRE transactions on information theory* 8.2 (1962), pp. 179–187.

[74] Kenneth Levenberg. "A method for the solution of certain non-linear problems in least squares". In: *Quarterly of applied mathematics* 2.2 (1944), pp. 164–168.

[75] Donald W Marquardt. "An algorithm for least-squares estimation of nonlinear parameters". In: *Journal of the society for Industrial and Applied Mathematics* 11.2 (1963), pp. 431–441.

[76] Greg Turk. *The Stanford Bunny*. https://www.cc.gatech.edu/~turk/bunny/bunny.html. 2000.

[77] Brent Burley. "Physically-Based Shading at Disney". In: 2012.

[78] Brent Burley. "Extending the Disney BRDF to a BSDF with integrated subsurface scattering". In: *Physically Based Shading in Theory and Practice'SIGGRAPH Course* (2015).

[79] Blender. *Blender BSDF*. https://docs.blender.org/manual/en/latest/render/shader_nodes/shader/principled.html. Dec. 2020.

[80] Jeppe Revall Frisvad, Toshiya Hachisuka, and Thomas Kim Kjeldsen. "Directional dipole model for subsurface scattering". In: *ACM Transactions on Graphics (TOG)* 34.1 (2014), pp. 1–12.

[81] Zhengqin Li, Yu-Ying Yeh, and Manmohan Chandraker. "Through the Looking Glass: Neural 3D Reconstruction of Transparent Shapes". In: *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 2020.

[82] Carnegie Mellon University. *CMU Graphics Lab Motion Capture Database*. 2002. URL: http://mocap.cs.cmu.edu/ (visited on 10/18/2017).

[83] J. Fayad, L. Agapito, and A. Del Bue. "Piecewise Quadratic Reconstruction of Non-Rigid Surfaces from Monocular Sequences". In: *ECCV*. 2010.

[84] L. Torresani, A. Hertzmann, and C. Bregler. "Learning Non-Rigid 3D Shape from 2D Motion". In: *Advances in Neural Information Processing Systems 16*. Ed. by Sebastian Thrun, Lawrence Saul, and Bernhard Schölkopf. Cambridge, MA: MIT Press, 2004.

[85] Chris Russell, Joao Fayad, and Lourdes Agapito. "Dense non-rigid structure from motion". In: *2012 Second International Conference on 3D Imaging, Modeling, Processing, Visualization & Transmission*. IEEE. 2012, pp. 509–516.

[86] Suryansh Kumar, Anoop Cherian, Yuchao Dai, and Hongdong Li. "Scalable dense non-rigid structure-from-motion: A grassmannian perspective". In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2018, pp. 254–263.

CONTRIBUTION A

Scene reassembly after multimodal digitization and pipeline evaluation using photorealistic rendering

Scene reassembly after multimodal digitization and pipeline evaluation using photorealistic rendering

JONATHAN DYSSEL STETS[1,†], ALESSANDRO DAL CORSO[1,†], JANNIK BOLL NIELSEN[1], RASMUS AHRENKIEL LYNGBY[1], SEBASTIAN HOPPE NESGAARD JENSEN[1], JAKOB WILM[1], MADS BRIX DOEST[1], CARSTEN GUNDLACH[2], EYTHOR RUNAR EIRIKSSON[1], KNUT CONRADSEN[1], ANDERS BJORHOLM DAHL[1], JAKOB ANDREAS BÆRENTZEN[1], JEPPE REVALL FRISVAD[1,*], AND HENRIK AANÆS[1]

[1] *Department of Applied Mathematics and Computer Science, Technical University of Denmark, Richard Petersens Plads, 2800 Kongens Lyngby, Denmark*
[2] *Department of Physics, Technical University of Denmark, Fysikvej, 2800 Kongens Lyngby, Denmark*
[†] *Joint primary authors*
[*] *Corresponding author: jerf@dtu.dk*

Transparent objects require acquisition modalities that are very different from the ones used for objects with more diffuse reflectance properties. Digitizing a scene where objects must be acquired with different modalities, requires scene reassembly after reconstruction of the object surfaces. This reassembly of a scene that was picked apart for scanning seems unexplored. We contribute with a multimodal digitization pipeline for scenes that require this step of reassembly. Our pipeline includes measurement of bidirectional reflectance distribution functions and high dynamic range imaging of the lighting environment. This enables pixelwise comparison of photographs of the real scene with renderings of the digital version of the scene. Such quantitative evaluation is useful for verifying acquired material appearance and reconstructed surface geometry, which is an important aspect of digital content creation. It is also useful for identifying and improving issues in the different steps of the pipeline. In this work, we use it to improve reconstruction, apply analysis by synthesis to estimate optical properties, and to develop our method for scene reassembly.

OCIS codes: *(150.4232) Multisensor methods; (150.6910) Three-dimensional sensing; (150.1488) Calibration; (160.4760) Optical properties; (290.1483) BSDF, BRDF, and BTDF; (330.1690) Color.*

This is the authors' version of the work. The definitive version is available at https://doi.org/10.1364/AO.56.007679

1. INTRODUCTION

Several research communities work on techniques for optical acquisition of physical objects and their appearance parameters [1–5]. Thus, we are now able to acquire nearly any type of object and perform a computer graphics rendering of nearly any type of scene. The range of applications is broad and includes movie production [2], cultural heritage preservation [3], 3D printing [4], and industrial inspection [5]. A gap left by these multiple endeavors is a coherent scheme for acquiring a scene consisting of several objects that have very different appearance parameters, together with the reassembly of a digital replica of such a scene. Our objective is to fill this gap for the combination of transparent and opaque objects, as many real world scenarios exhibit this combination. An example is a living room, like the one rendered in Fig. 1 (right). We propose a pipeline for acquiring and reassembling digital scenes from this type of heterogeneous real-world scenes. In addition, our pipeline closes the loop by rendering calibrated images of the digital scene that are commensurable with photographs of the original physical scene (see Fig. 1, left). This allows for validation and fine-tuning of appearance parameters. The quantitative evaluation we get from pixelwise comparison of rendered images with photographs is a great improvement with respect to validation of the acquired digital representation of the physical objects.

When addressing the problem of acquiring a heterogeneous scene, there is an infinite variety of scenes and object types to choose from. So, to make our task feasible, we focus on scenes that combine glassware and non-transparent materials, more specifically, white tablecloth and cardboard with a checkerboard pattern. We made these choices as glass requires a different acquisition modality, the tablecloth bidirectional reflectance distribution function (BRDF) is spatially uniform but not necessarily simple, and the cardboard has simple two-color varia-

Fig. 1. To the left, we compare rendered images (top) with photographs (bottom). More views are available in Appendix A. The scenes to the left were digitized using our pipeline and include both glass objects and non-transparent objects (tablecloth and backdrop). To the right, we exemplify the use of our pipeline for virtual product placement using our digitized glass objects, with estimated optical properties and artifact-reduced removal of markers.

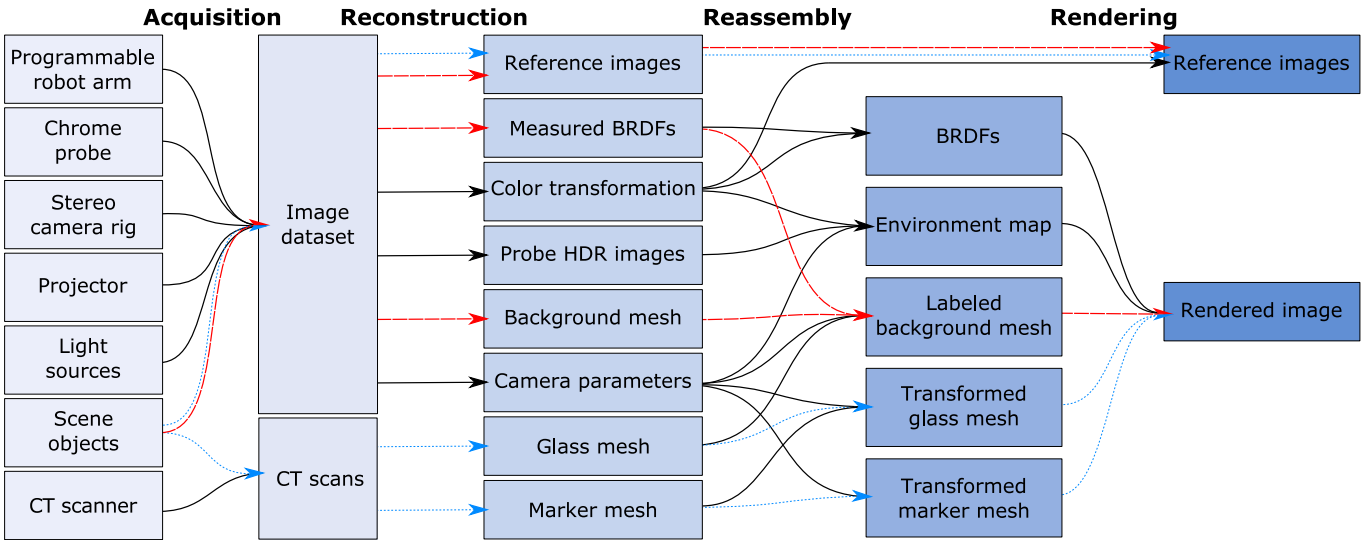


Fig. 2. Overview of our digitization pipeline in four main stages: acquisition, reconstruction, reassembly, and rendering. A video presentation of our pipeline is available in supplementary Visualization 1. Colored arrows show the path through the pipeline of transparent objects (dotted blue) and non-transparent objects (dashed red).

tion. The latter is particularly useful for observing how light refracts through the glass. The chosen case is also of particular interest, since glass is present in many intended applications of optical 3D acquisition. Considering the highly multidisciplinary nature of our work, we have released our dataset (http://eco3d.compute.dtu.dk/pages/transparency). This facilitates further investigation by other researchers of the different steps of our pipeline with the possibility of a quantitative feedback at the end of the process.

A. Related Work and Contributions

Researchers occasionally compare renderings with photographs to provide a qualitative verification of a presented rendering technique. The work by Phong [6], Goral et al. [7], and Takagi et al. [8] are early examples of this trend. A procedure to bring a rendered image close to a photograph was first presented by Meyer et al. [9]. In this work, likeness of images was evaluated perceptually by human observers. Pixelwise comparison of photographs with rendered images is surprisingly uncommon. The few examples we have found are by Rushmeier et al. [10], Karner and Prantl [11], Pattanaik et al. [12], and Jones and Reinhart [13, 14]. These examples build on the rendering framework described by Greenberg et al. [15]. Employing such a framework for more complex scenes is a long and tedious process [16]. The key issue is that a scene specification is expected as an input.

Several problems arise as a result of not having correspondence between the physical and the digital scene. Misalignment due to inaccurate scene and viewing geometry and inaccurate orientation of the lighting environment are some of the essential problems identified in previous work [17, 18]. One way to deal with this problem is to calculate error for image patches when evaluating results [13, 19, 20]. As opposed to this, our digitization pipeline (Fig. 2) provides both reference photographs and correspondingly calibrated scene and viewing geometry so that pixelwise comparison becomes meaningful.

Pixelwise comparison of rendered images with photographs is not only useful for quantifying the photorealism of a rendering in terms of error measurements. We find it particularly useful for improving the digitization pipeline. The fact that our pipeline enables quantitative evaluation led us to more specific contributions in its different steps. These contributions are mostly in the reassembly and are as follows. (a) A cross-modality marker-based placement approach, enabling accurate placement of objects scanned with one modality into scenes scanned with

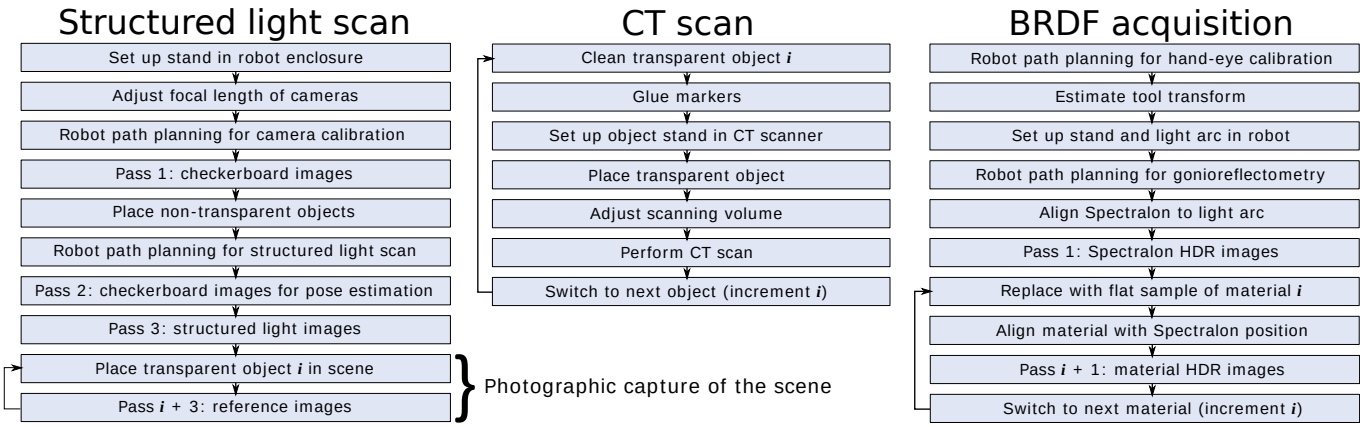


Fig. 3. Our workflow for scanning the geometry of non-transparent objects and collecting reference images (left), for scanning the geometry of transparent objects (middle), and for measuring material reflectance properties (right).

another modality. (b) A soft object deformation technique dealing with surface intersections after object placement, which is critical for scenes containing transparent or translucent objects. (c) A micropolygon labeling approach for assigning BRDFs to acquired geometry. (d) A color calibration scheme enabling use of spectral optical properties for calculating reflectance, transmittance, and absorption. (e) Perspective unwrapping of mirror probe images to improve precision when the environment is not very distant. (f) Use of analysis by synthesis for fine-tuning physics-based optical properties.

Digitization is most often unimodal and tailored toward objects with a specific type of surface reflectance behavior [1]. While unimodal techniques are becoming more versatile [21–23], objects with a transparent material like glass still pose challenging problems. Their reflectance behavior is so different that they require an entirely different modality, such as computed tomography (CT) [24]. The transparent object must then be removed from the scene to be scanned elsewhere. In the meantime, the surrounding scene can be scanned with a more common technique. However, as the transparent object takes most of its appearance from its surroundings, it must be repositioned in the surrounding scene (physically and digitally) if we are to take reference images for comparison with rendered images. The purpose of our scene reassembly is to address this type of issue.

Our digitization technique is multimodal. Currently, such techniques seem to exist only in the context of sensor fusion [25–27]. Here, the goal is to optimize reconstruction by fusing data from different sensor modalities with complementary characteristics. Even so, the different modalities see the same object and thus work for materials with a similar reflectance behavior. The challenge is then mostly in registration of the scans. In their final remarks and suggestions for future work, Weinmann and Klein [1] discuss possible ways of combining multiple techniques tailored to different types of surface reflectance. Our pipeline is a different way to take a step in this direction.

In summary, our work makes it possible to perform multimodal digitization and scene reassembly in such a way that rendered images of the reassembled scene can be quantitatively compared to photographs of the original. This enables us to provide the first empirically founded investigation of the appearance accuracy of objects digitized using a non-optical scanner.

2. DIGITIZATION PIPELINE

We divide our pipeline into four stages: (1) acquisition, (2) reconstruction, (3) reassembly, and (4) rendering. Figure 2 provides an overview. As illustrated, transparent objects (dotted blue arrows) and non-transparent objects (dashed red arrows) take different paths through the pipeline. The acquisition stage includes structured light scanning of non-transparent objects, CT scanning of transparent objects, gonioreflectometric reflectance measurements, and photographic capture of environment, color chart, and scene reference images. Figure 3 provides details of our workflow in these acquisition steps (except the simpler captures of environment and color chart). The second stage includes reconstruction of surface meshes, material BRDFs, and color space. The third stage is reassembly of the digital scene consisting of geometric objects, material appearance properties, and environment map. The fourth and final stage is rendering and comparison with reference images.

Our acquisition stage requires an elaborate hardware setup. We assemble the physical scene in a black light-proof enclosure. This has five LED light tubes for scene lighting, which we capture by high dynamic range (HDR) imaging of a light probe. To acquire non-transparent geometry inside this enclosure, we use a structured light scanner consisting of a toe-in stereo camera rig and a light projector mounted on a robotic arm [28, 29]. We chose a converging camera configuration (toe-in) to increase the overlap of the fields of view so that we get a denser point cloud per stereo view. Together with an LED based illumination arc, we also use this camera rig with exact control for measuring isotropic BRDFs. For transparent objects, we use a CT scanner. In the following subsections, we describe the individual steps of the pipeline with focus on details required for reproducibility and on non-standard techniques that we introduce.

A. Camera Calibration and Settings

The camera system is calibrated using a standard technique [30]. Our calibration board is an 11 by 12 black-and-white checkerboard. For the intrinsic calibration (Pass 1 of Fig. 3, left), we include a large variety of views to estimate good lens distortion coefficients. To facilitate stereo calibration, we also ensure that both cameras have the calibration board fully in view. For extrinsic calibration (Pass 2 of Fig. 3, left), we balance good coverage of the scene and good coverage of the calibration board. Since we cannot change the camera system while collecting data, we

chose a small aperture to ensure that background and projected structured light patterns are always in focus from all views. The full setup is in a dark room environment to eliminate external light, so we use a long shutter time (600 ms) to obtain sufficient exposure. A slight noise component is present in the images, but this is considered negligible. Finally, we use the estimated distortion coefficients to remove distortion from all images in the dataset so that subsequent algorithms may assume a pinhole camera model.

To avoid any compression or manipulation of the images by the camera software, in particular automatic color correction, we read the raw sensor data directly. We use bilinear interpolation to reconstruct RGB images from the raw Bayer pattern images. By doing this, we obtain a consistent RGB color space. Moreover, the raw sensor data is linear and correlates directly with radiometric quantities, which allows for better BRDF and environment map estimation in later stages of our pipeline.

We capture radiometrically relevant parts of our dataset in HDR by stacking multiple exposures [31]. More specifically, we stack 11 exposures at one-stop intervals ranging from 1 to 2048 ms. For the other parts of the dataset, we capture a single image at an exposure time of 600 ms.

B. Surface Reconstruction from Structured Light

We use a standard Gray code structured light approach to generate raw point clouds for a scene [32, 33]. With camera parameters from the calibration, we transform these point clouds into the same world coordinate system.

To reconstruct one connected triangle mesh from the point clouds, we merge them into a single point cloud and perform screened Poisson reconstruction with trimming and an octree depth of nine [34]. This technique requires point normals, so before the merging we generate normals for each point cloud as follows. We resample the point cloud down to 100,000 vertices via Poisson disk sampling [35] and then compute normals via planar fitting to a nearest neighborhood of 500 points ($\sim$16 mm radius). We then reorient all the normals according to the location of one of the cameras and transfer them back onto the original point cloud. This procedure ensures smooth continuous normals, necessary for a good performance of the mesh reconstruction algorithm. As we rely on smoothing, we cannot reconstruct features in the mesh with the same physical size as the alignment error accumulated from structured light and calibration. The aim of the chosen constants was to preserve features by striking a balance between too noisy and too smooth. The operability of the pipeline is however not sensitive to the choice of these constants.

C. Material BRDF Reconstruction

We assume that all non-transparent materials in the scene are opaque and isotropic, so we model their reflectance properties by BRDFs. To acquire a BRDF, we combine traditional canonical gonioreflectometric sampling [36] with a BRDF interpolation (reconstruction) technique [37]. We follow the workflow outlined in Fig. 3 (right). A light arc illuminates material samples from 11 unique inclinations, evenly distributed from 7.5° up to 90° with 7.5° steps. We place a flat material sample at the center of the circle partly traced by the light arc. Using the cameras mounted on the robot, we then measure radiance reflected by the sample across one octant of a sphere. The center of this sphere coincides with that of the light arc, while its radius is slightly larger to avoid collision between the robot and the arc. The robot moves in steps of 7.5° and captures 11 HDR images of the sample per step, one for each light direction. In total, this yields 2,783 HDR images per material. We avoid tangential and zenith viewing directions (90° and 0°, respectively). In the former case, no reflected radiance should be visible, while in the latter the light arc occludes the view of the sample.

The 2,783 observations are too few to faithfully represent the BRDF of a material in a photorealistic rendering. We need an interpolation scheme to fill the entire $(90 \times 90 \times 180)$ Mitsubishi Electric Research Laboratories (MERL) format BRDF look-up table [38]. The reconstruction method by Nielsen et al. [37] is our interpolation scheme. First, we use each of the 100 BRDFs in the MERL-dataset [38] as sample points in a $90 \cdot 90 \cdot 180 = 1{,}458{,}000$ dimensional space. The nonlinear mapping of Nielsen et al. [37] is then applied to each of the samples. The mapped samples are ordered as rows of a matrix $\mathbf{X} \in \mathbb{R}^{m \times d}$ where m is the number of BRDF samples and d is the dimension of the space. The zero-mean matrix is computed as $\mathbf{X} - \bar{\mathbf{x}}$, with $\bar{\mathbf{x}}$ being the sample mean. From this, the singular value decomposition $\mathbf{X} - \bar{\mathbf{x}} = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^T$ is used to compute the eigenvectors and eigenvalues of the covariance matrix of $\mathbf{X} - \bar{\mathbf{x}}$, which are given as the columns of $\mathbf{V}$ and the diagonal elements of $\mathbf{\Sigma}$, respectively. This is effectively a principal component analysis (PCA), where the eigenvectors are the principal components. A matrix composed of the scaled principal components as columns are computed as $\mathbf{Q} = \mathbf{V}\mathbf{\Sigma}$.

Now, the full BRDF can be reconstructed from this principal component space by projection. Let $\mathbf{x}' \in \mathbb{R}^n$ be n BRDF observations measured for a given material. Then, let $\bar{\mathbf{x}}' \in \mathbb{R}^n$ be the mean values and $\mathbf{Q}' \in \mathbb{R}^{n \times k}$ be the scaled eigenvectors corresponding to the direction pairs of those n observations. A vector $\mathbf{c}$ which spans the full space can be constructed by finding the linear combinations of principal components that best approximate the n observations. We do this by solving the linear least-squares optimization problem given by

$$\begin{aligned} \mathbf{c} &= \arg\min_{\mathbf{c}} \|(\mathbf{x}' - \bar{\mathbf{x}})' - \mathbf{Q}'\mathbf{c}\|^2 + \eta\|\mathbf{c}\|^2 \\ &= (\mathbf{Q}'^T\mathbf{Q}' + \eta\mathbf{I})^{-1}\mathbf{Q}'^T(\mathbf{x}' - \bar{\mathbf{x}}'). \end{aligned}$$

Note that by adding a penalty η to the norm of $\mathbf{c}$, this effectively becomes a Tikhonov regularized least squares. Now, the full, mapped BRDF is reconstructed as $\mathbf{x} = \mathbf{Q}\mathbf{c} + \bar{\mathbf{x}}$. The inverse of the nonlinear mapping applied to $\mathbf{X}$ is applied to $\mathbf{x}$ to get the actual, unmapped BRDF of the material. The described approach is applied to every single non-transparent material in the scene in order to obtain models of their reflectance properties.

This approach assumes that the MERL database encompasses the class of materials present in the scene. Effectively, this is a practical compromise between dense, unbiased, canonical BRDF sampling and fast, inferred BRDF sampling. This enables us to obtain high confidence BRDFs in a matter of a few hours.

D. Surface Reconstruction from CT

In our dataset, we have three glass objects: a sphere, a teapot (pot and lid) and a bowl (bowl and lid), for a total of five pieces. All objects have spherical plastic markers glued onto their outer surface. We CT scan each glass piece to obtain X-ray radiographs and use the CT PRO 3D reconstruction software from Nikon Metrology to obtain a volumetric image for each piece. The resolution of the reconstructed volume is up to 1000^3 voxels. Due to beam hardening, high density differences between materials lead to streak artifacts [39], especially around our markers and at the top and bottom of the objects (see Fig. 4). We account for these artifacts in the volumetric segmentation.

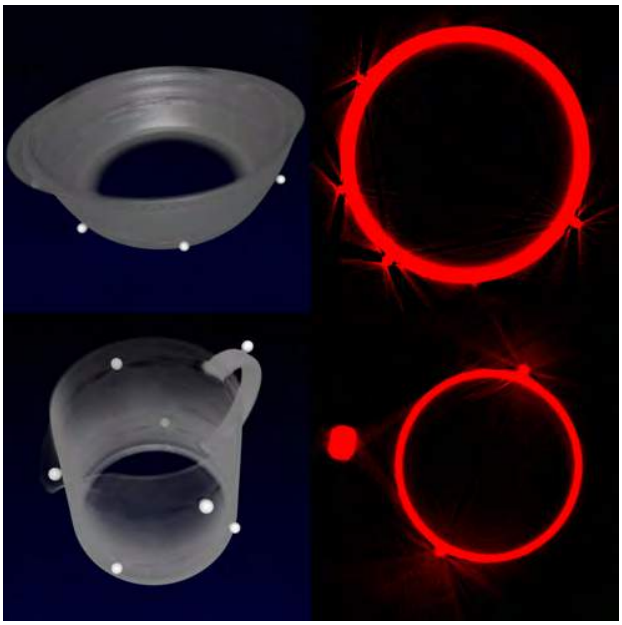


Fig. 4. CT scans of the bowl (top row) and the teapot (bottom row) with markers glued onto them. In the left column, visualized using a 1D transfer function. Note the different density of the markers. In the right column, a slice scaled to display streak artifacts.

From a CT scan, we generate two triangular meshes with vertex normals: one for the glass object and one the plastic markers. Figure 5 provides an overview of our procedure. We start with the markers, which appear as elements of higher density in the scan. We preprocess the scan by clamping all the values under a certain threshold to zero and then create a mesh using dual contouring [40]. Generating the glass mesh is more cumbersome. We also use dual contouring in this case, but because of the streak artifacts (Fig. 4) it is not possible to isolate the glass mesh via a threshold. Instead, we use a lower threshold that only removes noise, then estimate the marker positions, and use these to remove the markers from the glass mesh.

To estimate marker positions, we determine a series of center/radius pairs $(\mathbf{c}_i, r_i)$ by fitting a multi-sphere model to the marker mesh vertices using a tuned random sample consensus (RANSAC) algorithm [41]. We then carve a hole by excluding all the triangles that are inside a sphere with center $\mathbf{c}_i$ and radius $(1+\epsilon)r_i$, where ϵ is usually in the 0.5 to 0.75 range. We store the marker positions $\mathbf{c}_i$ so that we can use them to transform from the local coordinate system of the glass object to the world coordinate system (see Section F).

After removing the markers, the glass meshes still have aliasing artifacts. To deal with this issue, we first decimate the mesh down to 1% of the original vertices via quadric edge collapse. The holes are then easy to close by identifying the edge loops surrounding each hole and filling these with triangles. We then introduce a subdivision-decimation loop with alternating $\sqrt{3}$-subdivision [42] and decimation to 33% of the original vertices. We perform this subdivision-decimation operation four times to obtain a cleaned mesh. The decimation removes unwanted high frequency features from the mesh. Thus, we generate smooth meshes at the cost of some geometric precision. We are again trying to strike a balance between reconstruction error and too much smoothing. In Section 4, we compare our method with a different cleaning procedure that better preserves geometry.

Fig. 5. Reconstruction from CT with stages illustrated using Phong shading (top row) and wireframe shading (bottom row). After estimating the marker mesh (first column) and fitting spheres to the markers, we reconstruct the object mesh (second column). To eliminate noise, we first simplify the mesh (third column) and then close the holes and apply our subdivision-decimation loop to get the final object mesh (fourth column).

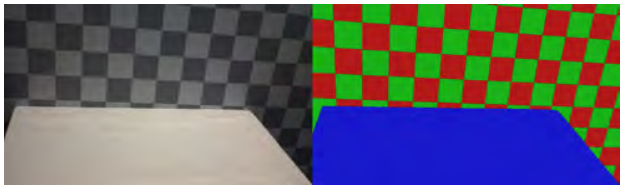


Fig. 6. Labeling of the image to the left results in the label image to the right. Each color in the label image represents a label that we assign a BRDF to. The black edges between labels indicate areas where we apply a nearest neighbor method.

E. Scene Reassembly for Non-Transparent Objects

Two operations are necessary to prepare the background mesh for rendering: labeling and deformation. In the labeling, our objective is to identify BRDFs and label each face of the mesh with a BRDF. Assuming a scene with a small number of known BRDFs, we apply edge detection and watershed on the images of the scene to segment BRDF boundaries. Shadows, specular highlights, and different viewing angles of the scene complicate fully automatic BRDF identification. Our approach gets us most of the way, but we manually correct any residual misclassification. Figure 6 shows a label image produced by our labeling technique.

The label images can be used in multi-view projective texturing of the background mesh. However, we would like to precompute the view and label selection instead of doing it millions and millions of times while rendering. To avoid uv-unwrapping of the mesh for storing precomputed labels, we take an approach inspired by micropolygon rendering [43]. We project each vertex of a face onto the label images of the scene and select the face BRDF according to the image label that most of the face vertices were projected to. If a vertex projects to an unknown label, we resolve it by a nearest neighbor search. Since faces around material boundaries overlap multiple materials,

No subdivision One subdivision Two subdivisions

Fig. 7. Subdividing the mesh dissolves unwanted boundary sawtooth artifacts that originate from the BRDF labeling.

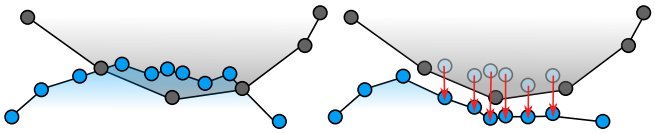


Fig. 8. Deformation of background mesh, where we push the background vertices down to avoid mesh intersection.

we get sawtooth artifacts. We dissolve these by subdividing the mesh until the rendered triangles are smaller than the surface area observed in a pixel, see Fig. 7.

When applying physically based rendering, we observed intersections between background scene and glass meshes. This could be due to small errors in reconstruction and positioning, or perhaps the harder glass objects press down the tablecloth when placed for reference imaging. It causes significant visual artifacts since the rendering exposes all surfaces of a transparent object. To eliminate these artifacts, we accommodate the hard object (glass) by deforming the soft object (tablecloth), see Fig. 8. To deform the soft object, we need a "down" direction in which to push the vertices. We first find contact vertices. These are vertices in each mesh that are close to any vertex of the other mesh. We consider vertices close if the distance between them is less than 7% of the bounding box diagonal of the hard object. Using least squares regression, we fit a contact plane to the contact vertices of the soft object. We set the sign of the contact plane normal so that the upper half-space contains the center of the hard object bounding box. Projection of a contact vertex to the normal of the contact plane then measures the height of the vertex. For each soft object contact vertex $\mathbf{x}$, we find the nearest hard object contact vertices and push $\mathbf{x}$ down below the lowest one of these.

F. Scene Reassembly for Transparent Objects

To reposition the glass objects in the scene, we rigidly transform the meshes reconstructed from CT to the world coordinate system of the background mesh. We obtain this transformation by matching markers in the stereo images with the marker coordinates $\mathbf{c}_i$ computed during reconstruction from CT (see Section D).

To find the markers, we employ a size invariant circle Hough transform [44]. This works well for our dataset, where the markers show high contrast against their surroundings. We match markers in the left and the right images via Sampson distance [45]. Using this technique, markers on the same epipolar line lead to false positives, so we manually inspect the result. We also manually discard detected markers that are visible through the glass, as the refraction would lead to incorrect positioning. Markers in both stereo images with no match are discarded. The result is a set of matched markers in image coordinates as seen in Fig. 9 (bottom left). We then triangulate the matched markers from the stereo views and gather them in clusters of 3D points. We remove outliers via their distance from the cluster centers, and for each cluster we select the point with the lowest reprojection error. An example of the points and clustering is shown in Fig. 9 (top middle).

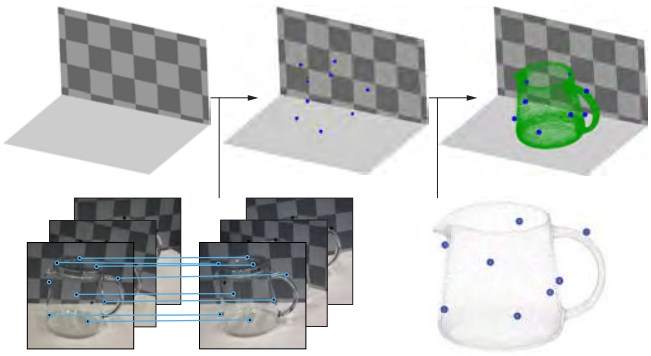


Fig. 9. Repositioning a CT scanned object in the background scene. We identify and match the markers in the stereo image pairs and calculate their corresponding 3D points. Pairing these with marker coordinates from the CT scans, we transform the CT scanned piece of an object into the world coordinate system.

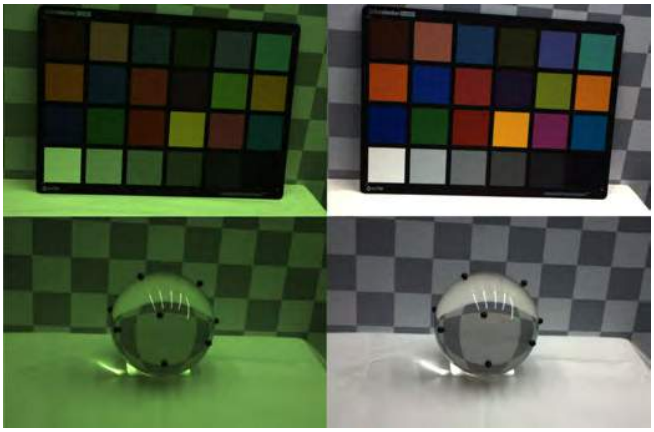


Fig. 10. Color calibration: raw images (left) and color corrected images (right). The camera sensor is particularly sensitive to green.

We manually pair the 3D marker coordinates from the images with the marker coordinates $\mathbf{c}_i$ from the CT scans. We perform Procrustes analysis [46] on the two point sets, excluding reflection, since we assume a rigid transformation applied to each vertex of the mesh. The bowl and the teapot are composed of multiple pieces. For these objects, we compute the transformation individually for each piece. The result of the object transformed into the scene is shown in Fig. 9 (top right). We found that in order to have low error in the transformation the chosen markers should sample the surface evenly and be visible from most views.

G. Color Calibration

Images are only quantitatively comparable if they live in the same color space. Thus, we must ensure that our radiometry-dependent data, namely reference images, environment map, and BRDFs, are in the same color space. We do this by imaging

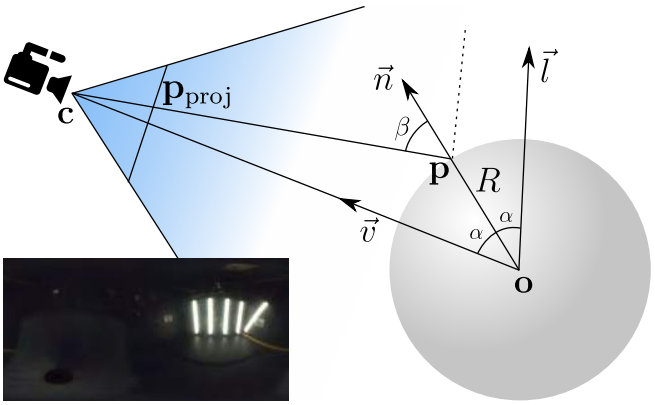


Fig. 11. Unwrapping of a spherical probe. We know the sphere radius R from specification, the camera position $\mathbf{c}$ through calibration, and the sphere center $\mathbf{o}$ by triangulation. Radiance at $\mathbf{p}_{\text{proj}}$ in our image then corresponds to the environment map direction $\vec{l}$. The result for the robot enclosure is in the lower left corner in latitude-longitude panoramic format (here tone-mapped).

a color chart of precisely known colors. More specifically, we use second degree root-polynomial color correction [47] based on a 24 patch ColorChecker Classic from X-Rite. This provides a matrix that transforms from camera RGB to XYZ, where we assume illuminant D50 when specifying the XYZ values of the colorchecker. With the assumption of illuminant D50, we can transform colors to the CIE L*a*b* color space and then compute color difference using the ΔE_{00} metric [48]. We use this to refine our result by minimizing ΔE_{00} using the Broyden-Fletcher-Goldfarb-Shanno (BFGS) algorithm [49]. The result is in Fig. 10. The average color difference is $\Delta E_{00} = 1.97 \pm 1.21$, which is larger than 1 JND (just noticeable difference) [50], but we find it acceptable.

Since we work with glass objects (and chrome, see Section H), we need refractive indices to determine reflectance, transmittance, and absorption properties. Refractive indices can be found per wavelength in tables of research papers. To use such spectral optical properties together with our trichromatic image data, we integrate them to CIE RGB using the CIE RGB color matching functions listed by Stockman and Sharpe [51]. It is important to normalize these functions [52] and to use RGB rather than XYZ [53]. This is because a refractive index is not a color, but rather a quantity that in trichromatic representation should resemble a sparse sampling of the spectrum. Thus, as recommended by other authors [54], we choose CIE RGB as our rendering color space. After transforming our image data from camera RGB to XYZ, we therefore convert them to CIE RGB [55]. As a final step, we apply Bradford chromatic adaptation [50], adapting to the originally assumed illuminant D50, so that renderings and reference images get closer to real life appearance.

H. Environment Lighting

To capture the lighting observed in the reference images, we use a method similar to the mirror probe technique [56]. However, we use a pinhole camera model for probe image unwrapping instead of the standard orthographic model. Our pipeline enables this as we have a calibrated camera and know its position relative to the photographed mirror probe. With the pinhole model, we obtain a more precise estimate of the environment lighting. The environment map is generated from HDR images and stored in latitude-longitude panoramic format [50]. We use a polished grade G100 chrome bearing ball as mirror probe.

An environment map represents an infinite area light and maps a direction to a texture element (a texel). To do unwrapping, we map each texel direction $\vec{l}$ to the corresponding pixel position $\mathbf{p}_{\text{proj}}$ in a light probe image. Given the configuration illustrated in Fig. 11, we have

$$\vec{v} = \frac{\mathbf{c}-\mathbf{o}}{\|\mathbf{c}-\mathbf{o}\|},\ \vec{n} = \frac{\vec{v}+\vec{l}}{\|\vec{v}+\vec{l}\|},\ \mathbf{p} = \mathbf{o} + R\vec{n},\ \mathbf{p}_{\text{proj}} = \mathbf{M}\ [\mathbf{p}^T\ \ 1]^T,$$

where camera matrix $\mathbf{M}$ and camera position $\mathbf{c}$ are available from our calibration. The radius of the sphere R is available from the bearing ball specification, and we find the center of the sphere $\mathbf{o}$ by manually annotating the sphere and then triangulating it. We assume that the distance to the actual light along $\vec{l}$ is equal to the distance between camera and sphere $\|\mathbf{c}-\mathbf{o}\|$. This assumption works well in practice, leading to an error smaller than the uncertainty of $\mathbf{o}$ caused by the triangulation. With the original orthographic camera model, we can reconstruct the lighting for all directions except one ($-\vec{v}$). In our model, we cannot reconstruct the lighting for a set of directions ($\vec{n} \cdot \vec{v} \leq R/\|\mathbf{c}-\mathbf{o}\|$), so we set them to black. Since we do our unwrapping in world space, we can combine contributions from multiple camera views with no need to align them afterwards.

The environment map is color corrected according to Section G, which enables us to correct for the angularly dependent reflectance of chrome. The correction is to divide by Fresnel reflectance, which we compute during unwrapping. As input for Fresnel's equations, we use the angle β between $\mathbf{c}-\mathbf{p}$ and $\vec{n}$ and the complex refractive index of chrome [57] converted from spectrum to CIE RGB. The result is shown in the inset of Fig. 11.

I. Rendering

We render images using progressive unidirectional path tracing [58, 59] implemented in OptiX [60]. The captured HDR environment map is the sole light source in our scene [56]. When rendering non-specular materials, we importance sample the environment map to get direct illumination and use sampling of a cosine-weighted hemisphere to get indirect illumination. From our labeling, we have one BRDF attached to each triangle in our scene. For non-transparent objects, we use our measured BRDFs tabulated in the MERL format [38]. To terminate paths probabilistically, we use Russian roulette based on the bihemispherical reflectance of each measured BRDF. This reflectance is calculated in a preprocessing step using Monte Carlo integration. We deal with transparent objects in the usual way, setting reflectance and transmittance according to Fresnel's equations of reflection and Bouguer's law of exponential attenuation. Given their small surface, we were unable to estimate a BRDF for the markers. Instead, we render them as glass with all refracted rays being absorbed.

3. ANALYSIS BY SYNTHESIS

The ability to render images comparable to photographs enables us to use our pipeline for improving parameter estimates through analysis by synthesis. As an example, we need a scaling factor for our HDR environment map as it measures relative radiance [31]. We estimate this factor by taking ratios of references

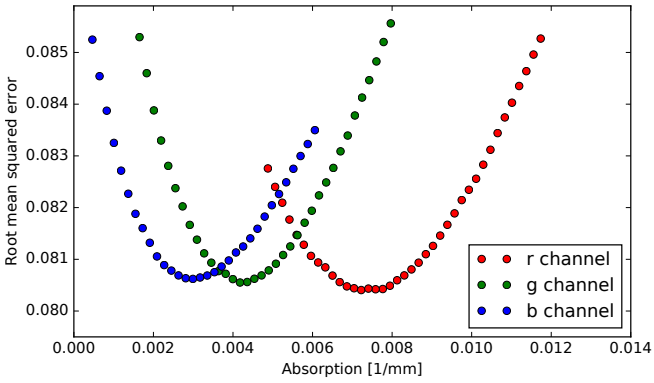


Fig. 12. Analysis by synthesis to estimate absorption of the glass bowl. We run renderings in low resolution and change the absorption in each color channel one at the time. In the case of the bowl, the blue channel is the most sensitive one.

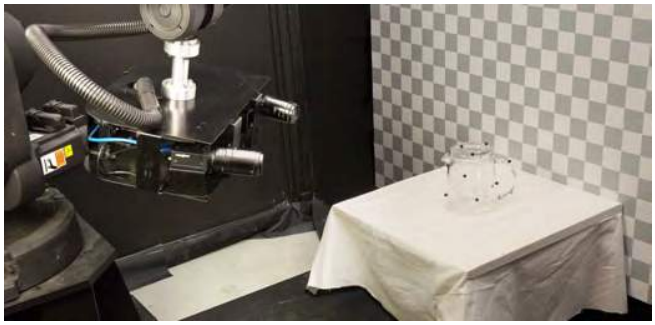


Fig. 13. Scene with checkerboard backdrop, lighting, glass teapot, and stand with table cloth observed by two cameras mounted on a 6-axis industrial robot arm.

and renderings with the background scene alone. Another example is estimating real and imaginary parts of glass refractive indices. As analysis by synthesis is fundamentally ill-posed [61], we take our outset in physics-based initial guesses such as Schott K5 crown glass (sphere and teapot) and soda lime glass (bowl). Spectral refractive indices for these glasses were obtained from an online database (http://refractiveindex.info) and converted to CIE RGB. All parameters were estimated using different views than the ones in our comparisons of renderings with references.

As an example of our analysis by synthesis, we plot the evolution of the root-mean-squared error (RMSE) for different renderings of the glass bowl in Fig. 12. For each rendering, we vary a trichromatic component of the absorption coefficient (which directly relates to the imaginary part of the refractive index). We identify a distinct minimum in the error for each channel, with a slightly larger uncertainty in the red channel. The minimum values in this figure were used in our renderings of the glass bowl. We apply the same analysis to the teapot and the sphere.

Given an initial guess for a parameter, we can employ standard optimization algorithms, defining the RMSE between the reference and the rendering as a cost function to minimize. To reduce rendering times, the evaluation of the cost function can be calculated on a downsampled image or limited to a specific patch of the images. Various general optimization algorithms exist for minimizing expensive cost functions [62].

Fig. 14. Markers rendered in blue and added to the reference image to validate marker positions by looking at pixel offsets.

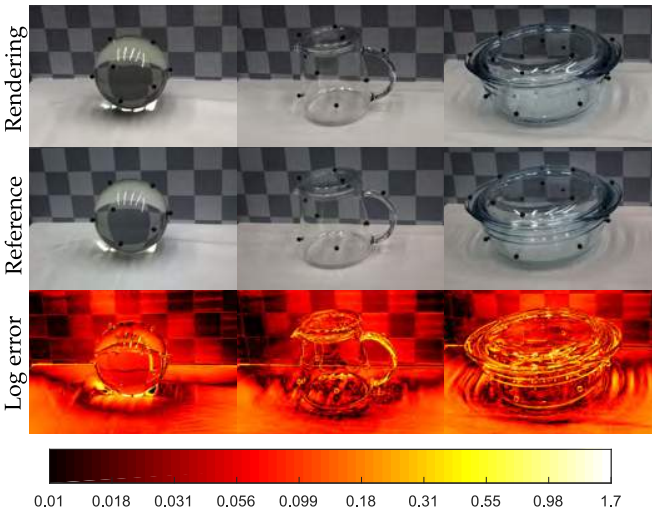


Fig. 15. Pixelwise error for three rendering-reference pairs. Error is the ℓ^2-norm of 32-bit per channel RGB images, visualized using a base 10 logarithmic scale.

4. RESULTS

Our scenes consist of a backdrop, a stand, and a glass object (with markers) placed on the stand. The backdrop is a 30 by 20 white-and-gray checkerboard print on 120 cm by 80 cm semi-matte cardboard and the stand is a tabletop with a white cloth. An example scene is depicted in Fig. 13. We implemented our reconstruction and reassembly procedures as a modular software pipeline and computed all rendered images using our path tracer. As illustrated in Fig. 2 and mentioned in Section G, we color correct both rendered images and reference images to have a meaningful perceptual comparison. Figure 14 compares markers in a reference image with rendered markers to validate our marker positioning. For the teapot, the average distance between the markers from stereo and the transformed markers from CT is 0.43 mm.

Figure 15 presents pixelwise comparisons of reference images and rendered images. The error images allow us to spot subtle differences not easily noticed in a perceptual comparison, such as the slight misalignments in geometry and highlights. As reference photographs were not captured in HDR, we clamp the renderings correspondingly. This means that areas of strong light intensity, such as highlights and intense caustics, appear black in the error images.

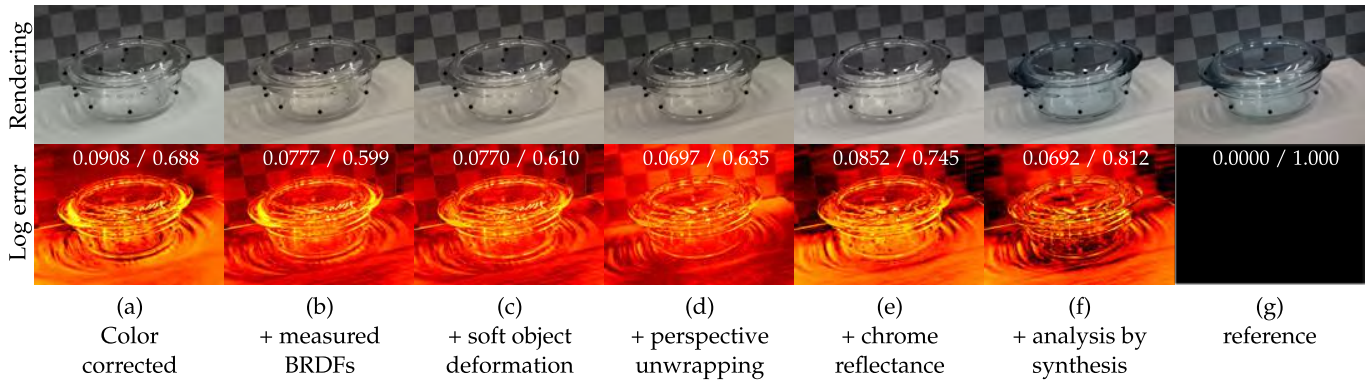


Fig. 16. Qualitative (top) and quantitative (bottom) step-by-step evaluation of our reassembly techniques. The log error images have the same format as in Fig. 15 and the reference photograph is in the rightmost column (g). In each column, we provide root-mean-squared error and structural similarity index (RMSE / SSIM). Both measures attain their best score in our final result (f).

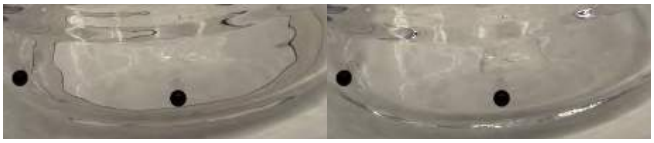


Fig. 17. Zoom-in of Figs. 16 (b) and (c) to emphasize the effect of our background deformation.

Fig. 18. Zoom-in of Fig. 16 (c) and (d) to emphasize the effect of our perspective unwrapping of the environment map.

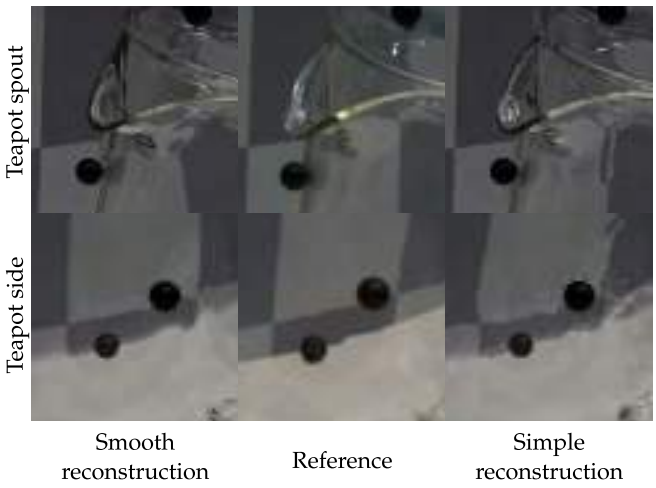


Fig. 19. Trade-off in mesh reconstruction. If we smooth more, we get less distortion in the refractions, but less precision in the mesh geometry. From left to right: Rendering with smoothing, reference image, rendering without smoothing.

Figure 16 exemplifies the impact on error images of some of our contributions. In Fig. 16 (a), we only reposition the glass object in the background scene and apply color correction (Sections F and G). This means that we use Lambertian materials (with bihemispherical reflectances from the measured BRDFs), an orthographic unwrapping model of the environment map, and no chrome reflectance correction or analysis by synthesis optimization. We compare to the reference image in Fig. 16 (g), with error images as in Fig. 15. Figure 16 (b) shows the impact of using measured BRDFs (Section C), resulting in a more accurate representation of the folds of the cloth in the background scene (top image) and an overall reduction of the error (bottom image). In Fig. 16 (c), we add deformation of the background mesh (Section E), which ensures that the background mesh does not poke through the glass surface (see a close-up in Fig. 17). Additionally, we can see how this improves the error on the lid of the bowl, because of refraction of light in the glass. The next step, Fig. 16 (d), shows the impact of our modified environment map unwrapping (Section H) against the standard orthographic unwrapping rotated according to our camera parameters. A close-up is available in Fig. 18. Our modified unwrapping provides a better shape and alignment of highlights and caustics. Partially due to the assumption of infinitely distant environment light, some alignment artifacts persist. In Fig 16 (e), we show the effect of correcting for chrome reflectance in our environment map reconstruction. Quantitatively, this changes the distribution of the error (bottom image). On the cloth, the exposure increases, exposing the caustics misalignment. On the backdrop, the error reduces. Interestingly, the structural similarity index (SSIM) improves while the RMSE worsens. Finally, in Fig. 16 (f), we use analysis by synthesis to adjust glass absorption. This improves the glass appearance, but it also leads to slight color changes in other parts of the scene due to indirect light paths. Because of this global influence, the analysis by synthesis introduces slightly too much absorption to compensate for the slightly too bright tablecloth.

As an example of how our pipeline can be used to validate existing algorithms, we investigate the case of glass object reconstruction. In Fig. 19, we compare two different reconstruction methods with focus on two parts of the teapot scene. Smooth reconstruction refers to the procedure described in Section D. The other procedure is to simply decimate the reconstructed mesh to 2.5% of the original vertices and apply Taubin smoothing [63]. This removes the high frequencies of the noise but much noise is still present in the midranges leading to wobbly refractions.

Rendering Reference Log error (as in 15)

Fig. 20. Material transitions: error lines along checker edges and along the boundary between tablecloth and backdrop.

Rendering Reference Log error (as in 15)

Fig. 21. Effect of separating markers from glass (refracted light close to marker) and of not accounting for subsurface scattering (dark areas close to caustics).

Our method in Section D reduces far more noise, but this is at the cost of greater changes to the overall shape. We note that a refractive object with a simple geometry is very hard to reconstruct automatically if fidelity and almost no noise are both required.

5. DISCUSSION

Since our pipeline enables us to compare renderings with photographs, we can identify problems in acquisition, reconstruction, and rendering that would otherwise have been hard to find. Camera calibration issues, for example, reveal themselves as error lines along edges (visible in Fig. 20). Color calibration issues reveal themselves as color shift. Such issues led us to more careful camera calibration procedures and the choice of root-polynomial color correction. Qualitative comparisons revealed artifacts in surface reconstruction, mesh intersections calling for deformation, misplacement of highlights, color shift due to chrome reflectance, and missing absorption in renderings (Figs. 16–19). Quantitative comparisons confirmed improvement due to perspective unwrapping of light probe images and led to analysis by synthesis.

The comparison with reference photographs before and after deformation (Fig. 17) to some extent validates our soft object deformation technique. Further validation would be desirable, but it is difficult to come up with a different experiment. Some kind of soft, durable memory foam with a scannable surface would be required as the soft object would otherwise change shape again once the hard object is removed. Our validation only supports that the cloth appearance (as observed through glass) is represented more faithfully after deformation.

We found analysis by synthesis useful for estimating parameters with an outset in physics-based initial guesses. The results in Fig. 12 show that we can estimate optical properties for a given material and use them in a different setting (right part of Fig. 1). The precision of the estimation varies with the impact of the property on the overall error, and the estimated parameters may compensate for unrelated errors. In this regard, specific scene configurations could be used to favor estimation of a particular parameter.

The most important limitation of our method is that we describe materials as large patches of isotropic BRDFs. In our renderings, this assumptions works well for the checkerboard backdrop but not for the cloth, where we both have subsurface scattering effects and probably anisotropy due to the weave structure of the cloth. Fig. 21 reveals that the rendered image is too dark in areas surrounding caustics. As seen in the light refracted through the sphere in the vicinity of the marker, our processing of the glass object to separate glass from markers causes some imprecision in the geometry. We believe this mainly influences the shape of the caustic. The bleeding of the caustic to areas that are much darker in the rendered images looks like backscattering from the table beneath the cloth. We refer to this as a kind of subsurface scattering.

Another limitation is seen at the transition between non-connected elements. It is visible in the renderings at the boundary between the cloth and the backdrop (see Fig. 20). The problem derives from the fact that the cloth and the backdrop were too close to each other during dataset acquisition. This resulted in the Poisson mesh reconstruction interpreting them as a continuous object instead of two separate ones. The problems around markers (Fig. 21) are also due to transition of materials. The marker removal and whole closing in the glass surface reconstruction interrupts the original shape of the surface. Furthermore, the markers are glued onto the glass surface, and the glue is not considered in the reconstruction and renderings. The marker glue problem is magnified by the glass refraction.

6. CONCLUSION

We have proposed a pipeline for multimodal scene digitization. Our work addresses the entire process from acquisition of the original objects, through reassembly of the digital scene, to accurate modeling of camera and environment. While the pipeline required several non-trivial steps, the benefits are correspondingly great since we can perform pixelwise comparisons between rendered images and photographs of the corresponding physical scene. This means that we have the means to quantitatively assess the accuracy of an acquired model based on comparison with empirical evidence. We believe this kind of quantitative assessment has not previously been possible for transparent objects. In applications like cultural heritage preservation and industrial inspection, where the accuracy of a digitization is important, such comparison with empirical evidence is crucial.

To the best of our knowledge, our work is also the first work to quantify the photorealism of a heterogeneous scene requiring multimodal acquisition.

Our dataset is publicly available so that others can test new techniques for the different steps of the pipeline with quantitative feedback based on photorealistic rendering. The fact that one can use off-the-shelf rendering techniques for improving the different steps of a multimodal digitization pipeline is perhaps the most important benefit of our work. An application of the full pipeline is the virtual product placement in Fig. 1. Another important application is the estimation of radiometric properties through analysis by synthesis. The ability to accurately estimate optical properties through computation rather than measurement, which might require specialized equipment, is likely to greatly simplify the digitization of radiometrically complex objects. In this paper, we estimated absorption and refractive indices of transparent objects, but analysis by synthesis could be equally useful for other materials with non-trivial BRDFs. This is another key benefit of our work that we believe is well worth exploring in the future.

Fig. 22. Comparison of renderings and photographs as in Fig. 1 (left), but with more views.

Funding. Innovation Fund Denmark (IFD) (75-2014-1, 3067-00001B, 5163-00001B, 5163-00003B).

A. APPENDIX

Figure 22.

REFERENCES

1. M. Weinmann and R. Klein, "Advances in geometry and reflectance acquisition (course notes)," in "Proceedings of SIGGRAPH Asia 2015 Courses," (ACM, 2015).
2. P. Debevec, "The light stages and their applications to photoreal digitial actors," in "SIGGRAPH Asia 2012 Technical Briefs," (2012).
3. L. Gomes, O. R. P. Bellon, and L. Silva, "3D reconstruction methods for digital preservation of cultural heritage: A survey," Pattern Recognition Letters **50**, 3–14 (2014).
4. L. Zhang, H. Dong, and A. E. Saddik, "From 3D sensing to printing: A survey," ACM Transactions on Multimedia Computing, Communications, and Applications **12**, 27:1–27:23 (2016).
5. J. B. Nielsen, E. R. Eiriksson, R. L. Kristensen, J. Wilm, J. R. Frisvad, K. Conradsen, and H. Aanæs, "Quality assurance based on descriptive and parsimonious appearance models," in "Workshop on Material Appearance Modeling (MAM 2015)," (The Eurographics Association, 2015), pp. 21–24.
6. B. T. Phong, "Illumination for computer generated pictures," Communications of the ACM **18**, 311–317 (1975).
7. C. M. Goral, K. E. Torrance, D. P. Greenberg, and B. Battaile, "Modeling the interaction of light between diffuse surfaces," Computer Graphics (Proceedings of SIGGRAPH 84) **18**, 213–222 (1984).
8. A. Takagi, H. Takaoka, T. Oshima, and Y. Ogata, "Accurate rendering technique based on colorimetric conception," Computer Graphics (Proceedings of SIGGRAPH 90) **24**, 263–272 (1990).
9. G. W. Meyer, H. E. Rushmeier, M. F. Cohen, D. P. Greenberg, and K. E. Torrance, "An experimental evaluation of computer graphics imagery," ACM Transactions on Graphics **5**, 30–50 (1986).
10. H. Rushmeier, G. Ward, C. Piatko, P. Sanders, and B. Rust, "Comparing real and synthetic images: Some ideas about metrics," in "Rendering Techniques '95 (Proceedings of EGWR 1995)," (Springer, 1995), pp. 82–91.
11. K. F. Karner and M. Prantl, "A concept for evaluating the accuracy of computer generated images," in "Proceedings of Spring Conference on Computer Graphics (SCCG 1996)," (1996).
12. S. N. Pattanaik, J. A. Ferwerda, K. E. Torrance, and D. P. Greenberg, "Validation of global illumination solutions through CCD camera measurements," in "Proceedings of Color Imaging Conference (CIC 1997)," (1997), pp. 250–253.
13. N. L. Jones and C. F. Reinhart, "Parallel multiple-bounce irradiance caching," Computer Graphics Forum (Proceedings of EGSR 2016) **35**, 57–66 (2016).
14. N. L. Jones and C. F. Reinhart, "Experimental validation of ray tracing as a means of image-based visual discomfort prediction," Building and Environment **113**, 131–150 (2017).
15. D. P. Greenberg, K. E. Torrance, P. Shirley, J. Arvo, J. A.Ferwerda, S. Pattanaik, E. Lafortune, B. Walter, S.-C. Foo, and B. Trumbore, "A framework for realistic image synthesis," in "Proceedings of SIGGRAPH 97," (ACM/Addison-Wesley, 1997), pp. 477–494.
16. F. Drago and K. Myszkowski, "Validation proposal for global illumination and rendering techniques," Computers & Graphics **25**, 511–518 (2001).
17. C. Ulbricht, A. Wilkie, and W. Purgathofer, "Verification of physically based rendering algorithms," Computer Graphics Forum **25**, 237–255 (2006).
18. J. Meseth, G. Müller, R. Klein, F. Röder, and M. Arnold, "Verification of rendering quality from measured BTFs," in "Proceedings of Applied Perception in Graphics and Visualization (APGV 2006)," (ACM, 2006), pp. 127–134.
19. A. I. Ruppertsberg and M. Bloj, "Rendering complex scenes for psychophysics using RADIANCE: How accurate can you get?" Journal of the Optical Society of America A **23**, 759–768 (2006).
20. A. Dal Corso, J. R. Frisvad, T. K. Kjeldsen, and J. A. Bærentzen, "Interactive appearance prediction for cloudy beverages," in "Workshop on Material Appearance Modeling (MAM 2016)," (The Eurographics Association, 2016), pp. 1–4.
21. B. Tunwattanapong, G. Fyffe, P. Graham, J. Busch, X. Yu, A. Ghosh, and P. Debevec, "Acquiring reflectance and shape from continuous spherical harmonic illumination," ACM Transactions on Graphics (Proceedings of SIGGRAPH 2013) **32**, 109:1–109:11 (2013).
22. T. Nöll, J. Köhler, G. Reis, and D. Stricker, "Fully automatic, omnidirectional acquisition of geometry and appearance in the context of cultural heritage preservation," Journal on Computing and Cultural Heritage **8**, Article 2 (2015).
23. H. Wu, Z. Wang, and K. Zhou, "Simultaneous localization and appearance estimation with a consumer RGB-D camera," IEEE Transactions on Visualization and Computer Graphics **22**, 2012–2023 (2016).
24. I. Ihrke, K. N. Kutulakos, H. P. A. Lensch, M. Magnor, and W. Heidrich, "Transparent and specular object reconstruction," Computer Graphics Forum **29**, 2400–2426 (2010).
25. A. Kolb, J. Zhu, and R. Yang, "Sensor fusion," in "Digital Representation of the Real World," M. A. Magnor, O. Grau, O. Sorkine-Hornung, and C. Theobalt, eds. (CRC Press, 2015), chap. 9, pp. 133–150.
26. V. Bhateja, H. Patel, A. Krishn, A. Sahu, and A. Lay-Ekuakille, "Multimodal medical image sensor fusion framework using cascade of wavelet and contourlet transform domains," IEEE Sensors Journal **15**, 6783–6790 (2015).
27. A. Pamart, O. Guillon, J.-M. Vallet, and L. De Luca, "Toward a multimodal photogrammetric acquisition and processing methodology for monitoring conservation and restoration studies," in "Eurographics Workshop on Graphics and Cultural Heritage," (The Eurographics Association, 2016), pp. 207–210.
28. H. Aanæs and A. B. Dahl, "Accuracy in robot generated image data sets," in "Proceedings of SCIA 2015," , vol. 9127 of *Lecture Notes in Computer Science* (Springer, 2015), pp. 472–479.
29. H. Aanæs, R. R. Jensen, G. Vogiatzis, E. Tola, and A. B. Dahl, "Large-scale data for multiple-view stereopsis," International Journal of Computer Vision **120**, 153–168 (2016).
30. Z. Zhang, "A flexible new technique for camera calibration," IEEE Transactions on Pattern Analysis and Machine Intelligence **22**, 1330–1334 (2000).
31. P. E. Debevec and J. Malik, "Recovering high dynamic range radiance maps from photographs," in "Proceedings of SIGGRAPH 97," (ACM/Addison-Wesley, 1997), pp. 369–378.
32. J. L. Posdamer and M. Altschuler, "Surface measurement by space-encoded projected beam systems," Computer Graphics and Image Processing **18**, 1–17 (1982).
33. J. Geng, "Structured-light 3D surface imaging: a tutorial," Advances in

Optics and Photonics **3**, 128–160 (2011).
34. M. Kazhdan and H. Hoppe, "Screened Poisson surface reconstruction," ACM Transactions on Graphics **32**, 29:1–29:13 (2013).
35. M. Corsini, P. Cignoni, and R. Scopigno, "Efficient and flexible sampling with blue noise properties of triangular meshes," IEEE Transactions on Visualization and Computer Graphics **18**, 914–924 (2012).
36. J. F. Murray-Coleman and A. M. Smith, "The automated measurement of BRDFs and their application to luminaire modeling," Journal of the Illuminating Engineering Society **19**, 87–99 (1990).
37. J. B. Nielsen, H. W. Jensen, and R. Ramamoorthi, "On optimal, minimal BRDF sampling for reflectance acquisition," ACM Transactions on Graphics (Proceedings of SIGGRAPH Asia 2015) **34**, 186:1–186:11 (2015).
38. W. Matusik, H. Pfister, M. Brand, and L. McMillan, "A data-driven reflectance model," ACM Transactions on Graphics (Proceedings of SIGGRAPH 2003) **22**, 759–769 (2003).
39. J. F. Barrett and N. Keat, "Artifacts in CT: Recognition and avoidance," RadioGraphics **24**, 1679–1691 (2004).
40. T. Ju, F. Losasso, S. Schaefer, and J. Warren, "Dual contouring of Hermite data," ACM Transactions Graphics (Proceedings of SIGGRAPH 2002) **21**, 339–346 (2002).
41. M. A. Fischler and R. C. Bolles, "Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography," Communications of the ACM **24**, 381–395 (1981).
42. L. Kobbelt, "$\sqrt{3}$-subdivision," in "Proceedings of SIGGRAPH 2000," (ACM/Addison-Wesley, 2000), pp. 103–112.
43. R. L. Cook, "The Reyes image rendering architecture," Computer Graphics (Proceedings of SIGGRAPH 87) **21**, 95–102 (1987).
44. T. Atherton and D. Kerbyson, "Size invariant circle detection," Image and Vision Computing **17**, 795 – 803 (1999).
45. P. D. Sampson, "Fitting conic sections to "very scattered" data: An iterative refinement of the Bookstein algorithm," Computer Graphics and Image Processing **18**, 97–108 (1982).
46. J. C. Gower, "Generalized Procrustes analysis," Psychometrika **40**, 33–51 (1975).
47. G. D. Finlayson, M. Mackiewicz, and A. Hurlbert, "Color correction using root-polynomial regression," IEEE Transactions on Image Processing **24**, 1460–1470 (2015).
48. G. Sharma, W. Wu, and E. N. Dalal, "The CIEDE2000 color-difference formula: Implementation notes, supplementary test data, and mathematical observations," Color Research & Application **30**, 21–30 (2005).
49. J. Nocedal and S. J. Wright, *Numerical Optimization* (Springer, 2006), 2nd ed.
50. E. Reinhard, G. Ward, S. Pattanaik, P. Debevec, W. Heidrich, and K. Myszkowski, *High Dynamic Range Imaging: Acquisition, Display and Image-Based Lighting* (Morgan Kaufmann/Elsevier, 2010), 2nd ed.
51. A. Stockman and L. T. Sharpe, "The spectral sensitivities of the middle- and long-wavelength-sensitive cones derived from measurements in observers of known genotype," Vision Research **40**, 1711–1737 (2000).
52. J. R. Frisvad, N. J. Christensen, and H. W. Jensen, "Computing the scattering properties of participating media using Lorenz-Mie theory," ACM Transactions on Graphics (Proceedings of SIGGRAPH 2007) **26**, 60:1–60:10 (2007).
53. C. Ulbricht and A. Wilkie, "A problem with the use of XYZ colour space for photorealistic rendering computations," in "Proceedings of Colour in Graphics, Imaging, and Vision (CGIV 2006)," (2006), pp. 435–437.
54. J. Meng, F. Simon, J. Hanika, and C. Dachsbacher, "Physically meaningful rendering using tristimulus colours," Computer Graphics Forum (Proceedings of EGSR 2015) **34**, 31–40 (2015).
55. H. S. Fairman, M. H. Brill, and H. Hemmendinger, "How the CIE 1931 color-matching functions were derived from Wright-Guild data," Color Research & Application **22**, 11–23 (1997).
56. P. Debevec, "Rendering synthetic objects into real scenes: Bridging traditional and image-based graphics with global illumination and high dynamic range photography," in "Proceedings of SIGGRAPH 98," (ACM, 1998), pp. 189–198.
57. A. D. Rakić, A. B. Djurišić, J. M. Elazar, and M. L. Majewski, "Optical properties of metallic films for vertical-cavity optoelectronic devices," Applied Optics **37**, 5271–5283 (1998).
58. J. T. Kajiya, "The rendering equation," Computer Graphics (Proceedings of SIGGRAPH 86) **20**, 143–150 (1986).
59. M. Pharr, W. Jakob, and G. Humphreys, *Physically Based Rendering: From Theory to Implementation* (Morgan Kaufmann/Elsevier, 2017), 3rd ed.
60. S. G. Parker, J. Bigler, A. Dietrich, H. Friedrich, J. Hoberock, D. Luebke, D. McAllister, M. McGuire, K. Morley, A. Robison, and M. Stich, "OptiX: A general purpose ray tracing engine," ACM Transactions on Graphics (Proceedings of SIGGRAPH 2010) **29**, 66:1–66:13 (2010).
61. M. Hejrati and D. Ramanan, "Analysis by synthesis: 3D object recognition by object reconstruction," in "Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR 2014)," (2014), pp. 2449–2456.
62. D. R. Jones, M. Schonlau, and W. J. Welch, "Efficient global optimization of expensive black-box functions," Journal of Global Optimization **13**, 455–492 (1998).
63. G. Taubin, "A signal processing approach to fair surface design," in "Proceedings of SIGGRAPH 95," (ACM Press, 1995), pp. 351–358.

CONTRIBUTION B

Modeling the Anisotropic Reflectance of a Surface With Microstructure Engineered to Obtain Visible Contrast After Rotation

Modeling the Anisotropic Reflectance of a Surface with Microstructure Engineered to Obtain Visible Contrast after Rotation

Andrea Luongo, Viggo Falster, Mads Brix Doest, Dongya Li, Francesco Regi, Yang Zhang, Guido Tosello, Jannik Boll Nielsen, Henrik Aanæs, Jeppe Revall Frisvad
Technical University of Denmark
aluo@dtu.dk

Abstract

Engineering of surface structure to obtain specific anisotropic reflectance properties has interesting applications in large scale production of plastic items. In recent work, surface structure has been engineered to obtain visible reflectance contrast when observing a surface before and after rotating it 90 degrees around its normal axis. We build an analytic anisotropic reflectance model based on the microstructure engineered to obtain such contrast. Using our model to render synthetic images, we predict the above mentioned contrasts and compare our predictions with the measurements reported in previous work. The benefit of an analytical model like the one we provide is its potential to be used in computer vision for estimating the quality of a surface sample. The quality of a sample is indicated by the resemblance of camera-based contrast measurements with contrasts predicted for an idealized surface structure. Our predictive model is also useful in optimization of the microstructure configuration, where the objective for example could be to maximize reflectance contrast.

1. Introduction

Engineering of surface microstructure to obtain custom reflectance properties, or so-called appearance printing, has many applications in product design and manufacturing. This research area has received significant attention [23, 11, 10, 7, 8, 14] and, recently, tooling was done with the objective of inserting a simple anisotropic surface microstructure into economic manufacturing processes [9]. The intended functionality of the anisotropic surface microstructure was to obtain high reflectance contrast for the surface when viewed from above at orthogonal angles. Using a microscope and a camera, the contrast was measured for different surface structure configurations to find the configuration revealing highest contrast [16].

In this work, we build an analytic bidirectional reflectance distribution function (BRDF) that models the anisotropic reflectance properties of the functional surface tested by previous authors [9, 16]. Our analytic BRDF model has two benefits. It is (1) useful for finding the surface structure configuration that theoretically produces optimal contrast. It also (2) enables estimation of surface quality from photographs. As an example, deviation in the contrast measured for a physical sample from the contrast predicted for an idealized surface corresponds to surface quality deviation, and contrast is measurable using simple computer vision [16].

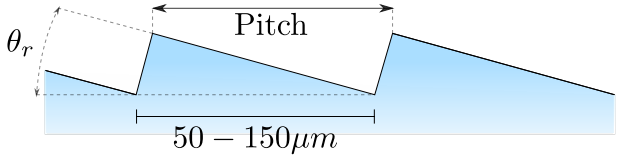


Figure 1. Engineered surface microstructure used by previous authors [9, 16] to obtain reflectance contrast when the surface is observed from above at orthogonal angles. We build an analytic BRDF model for this type of surface.

The simple anisotropic microstructure that produces visible contrast when viewed at orthogonal angles is created by having a sequence of small, parallel ridges as shown in Figure 1. The angle θ_r is a microstructure configuration referred to as the ridge angle. Based on how the structure is rotated around its macrosurface normal, it reflects light in different ways. Thus contrast can be generated by rotating the object by $90°$.

The reflectance properties of the ridged surface have only been studied experimentally [16]. The analytic BRDF we provide is built for this particular ridged surface structure, but a similar procedure could be used to derive an analytical model for a surface with a different engineered surface structure. Figure 2 shows an example of a quad rendered with our BRDF before and after rotation by $90°$. The contrast produced by the ridges having orthogonal orientation is clearly visible in this image.

Luongo, A., Falster, V., Doest, M. B., Li, D., Regi, F., Zhang, Y., Tosello, G., Nielsen, J. B., Aanæs, H., and Frisvad, J. R. Modeling the anisotropic reflectance of a surface with microstructure engineered to obtain visible contrast after rotation. In *Proceedings of IEEE International Conference on Computer Vision Workshop (ICCVW 2017)*, pp. 159-165. October 2017. https://doi.org/10.1109/ICCVW.2017.27

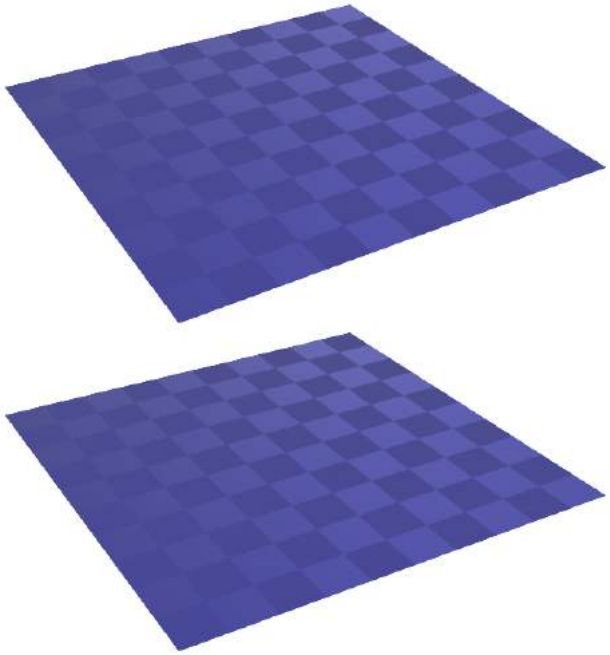


Figure 2. Quad rendered using our BRDF model before and after rotation by 90°. The orientation of the ridges follows a checkerboard pattern: two adjacent squares have ridges oriented in orthogonal directions.

2. Related Work

The work of Torrance and Sparrow [18, 19] is an early example of measuring the reflectance properties of a surface and subsequently developing a BRDF model for predicting the measured properties. Torrance and Sparrow [18] investigated metals and ceramics and processed the surfaces of their samples with the objective of having isotropic, random rough surfaces. They developed their BRDF model for this type of surface in order to explain surprising occurrences of off-specular peaks in the reflectance measurements [19]. Our work is similar, but we model the reflectance properties of plastic samples with an anisotropic, ridged surface.

With a similar approach, Ward [22] measured and modeled the bidirectional reflectance properties of anisotropic, random rough surfaces. He found good agreement between model and measurements for materials such as varnished wood and unfinished (rolled) or brushed metals. Our work is different in the sense that we model a ridged rough surface instead of a random rough surface.

Poulin and Fournier [15] presented one of the first BRDF models for an anisotropic surface with a specific microstructure. Their model assumes a microstructure consisting of half cylinders each with its axis lying in the surface tangent plane. More generically, Ashikhmin et al. [2] describe a methodology for generating a BRDF according to a given microfacet normal distribution function. Our ridged surface microstructure requires a slightly different approach as the microfacet normals are predominantly in two directions. The specific microstructure we model is interesting because it can be engineered. This enables us to compare reflectance properties predicted by our model with measured reflectance properties.

Using a generic BRDF model [2], it is possible to match observed reflectance properties by acquiring spatially varying microfacet normal distribution functions for an anisotropic surface [21]. This is impressive, but then deviations between predicted and measured properties cannot be used to assess how close an engineered microstructure is to the desired idealized microstructure.

Researchers working with techniques for BRDF printing have an opportunity to compare predicted reflectance properties with those of engineered surfaces. Weyrich et al. [23] use micro milling to obtain a surface structure with a specific microfacet normal distribution. This is similar to the tooling part of the manufacturing process that we are modeling [9]. Our added step of transferring the tool microstructure to a polymer component enables absorption (colored surfaces). In previous work [11, 7], absorption was added by applying different inks to the surface. This means that their BRDF model is a weighted average of differently oriented ink BRDFs, where we have a combination of surface and subsurface scattering effects. Other authors [10, 8] improve the microstructure resolution as compared with Weyrich et al. [23], but their techniques do not allow for absorption effects. In the work of Levin et al. [8], microfacets are at a scale that requires a BRDF model based on wave optics. Our pitches ranging from 50 to 150 microns can safely be modeled using geometrical optics. None of this previous work on BRDF printing includes shadowing and masking in their BRDF models. This is however important in the microstructure we investigate due to the steep slope of the ridge edges.

In recent work, Pereira et al. [14] show that magnetic microflakes can be used for anistropic BRDF printing. They measure the BRDFs printed by their technique but do not provide a predictive BRDF model.

Levin et al. [8] investigated the same kind of contrast that we are aiming at with our ridged surface structure. Their technique is however very different as it is based on wave interference effects. While they seem to achieve better contrast than ridged surfaces, they use photolithography which has high capital and operational cost and cannot easily be used with polymeric or curved substrates [1]. Nevertheless, it is noteworthy how easily their very small surface features produce contrast through wave interference.

McGunnigle [12] uses a bivariate Gaussian distribution (no Fresnel or geometrical attenuation effects) to model the anisotropic reflectance of a surface sandpapered in one direction. While his directional surface microstructure seems

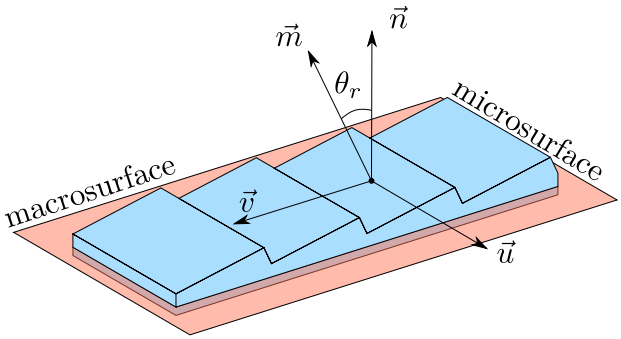


Figure 3. Microstructured surface and simplified macrosurface.

a bit like ours, his reflectance model is one-dimensional and considers only the azimuthal angle of the light source.

3. BRDF Model

The engineered surface that we consider is composed of many parallel ridges with a pitch of between 50 μm and 150 μm (see Figure 1). If viewed at a reasonable distance, these details can be assumed to be too small to be seen directly (at a distance of 0.5 m, humans can discern details of about 150 μm [13]). Thus we choose to model our surface by a macrosurface with an appropriate BRDF. In this way, the details of the microstructure are represented by the reflectance properties of the surface. This is analogous to other microfacet BRDF models [19, 3, 4, 2], where a rough surface with a complicated microgeometry is modeled by a simpler surface with an appropriate BRDF that can replicate the overall light scattering of the microsurface.

Figure 3 illustrates the macrosurface for our particular microsurface. In this model, we have a microsurface normal $\vec{m}$ and a macrosurface normal $\vec{n}$ (both are unit vectors). In addition, $\vec{u}$ is a vector parallel to the ridges and orthogonal to the normal $\vec{n}$, and $\vec{v}$ is a vector aligned with the direction of the pitches. Together, $\vec{u}, \vec{v}, \vec{n}$ is an orthonormal basis of the macrosurface tangent space.

Microfacet models represent the microsurface in terms of a microfacet reflectance function f_m, a geometrical attenuation function G, and a normal distribution function D. These are combined by integration over all microfacet normals to form a BRDF f for the macrosurface [20].

3.1. Geometrical Attenuation Function G

The portion of the microsurface with normal $\vec{m}$ visible from both directions $\vec{\omega}_i$ and $\vec{\omega}_o$ is described by the geometrical attenuation function $G(\vec{\omega}_i, \vec{\omega}_o, \vec{m})$. This means that the function models shadowing and masking effects.

An exact formulation of G is rarely available since it depends on the geometrical details of the particular surface. Most often, the function is approximated based on assumptions about the surface geometry (such as v-grooves [19]).

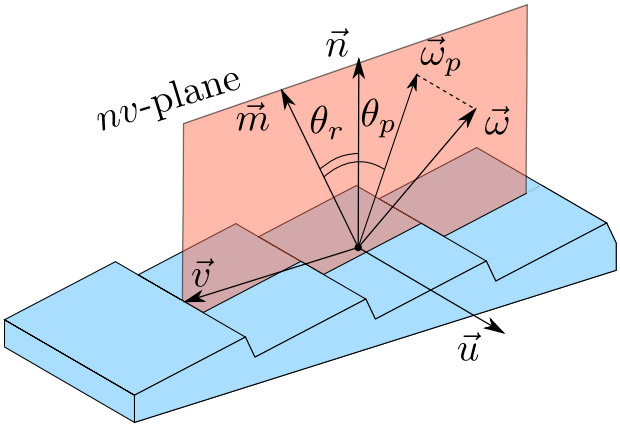


Figure 4. The angle θ_p between the microsurface normal $\vec{m}$ and the projection of the vector $\vec{\omega}$ on the nv-plane is used to evaluate the geometrical attenuation function.

Smith [17] derived an approximation of G for surfaces with Gaussian microfacet normal distribution. This has the useful property of being separable into the product of two mono-directional functions (one for shadowing and one for masking):

$$G(\vec{\omega}_i, \vec{\omega}_o, \vec{m}) \approx G_1(\vec{\omega}_i, \vec{m}) G_1(\vec{\omega}_o, \vec{m}). \quad (1)$$

Given the particular regularity of the microsurface we are dealing with, we have derived an expression for G_1 (details are provided in Appendix A) that is suitable for our model:

$$G_1(\vec{\omega}, \vec{m}) = \chi^+\left(\frac{\vec{\omega}_p \cdot \vec{m}}{\vec{\omega}_p \cdot \vec{n}}\right) \times \left[1 - \min\left(1, |\tan\theta_r \tan\theta_p|\right)\right], \quad (2)$$

where $\chi^+(a)$ denotes a Heaviside step function that is 1 for $a > 0$ and 0 otherwise. We let θ_p denote the angle between $\vec{m}$ and the projection $\vec{\omega}_p$ of $\vec{\omega}$ on the plane spanned by $\vec{n}$ and $\vec{v}$, see Figure 4. Thus,

$$\cos\theta_p = \frac{\vec{\omega}_p \cdot \vec{m}}{|\vec{\omega}_p|} = \frac{(\vec{\omega} - (\vec{\omega} \cdot \vec{u})\vec{u}) \cdot \vec{m}}{|\vec{\omega} - (\vec{\omega} \cdot \vec{u})\vec{u}|},$$

which reveals that the orientation of the macrosurface is required to evaluate the geometrical attenuation function. This is as expected since we are dealing with an anisotropic surface microstructure.

3.2. Microfacet Distribution Function D

The manufacturing process introduces irregularities on the ridges. The surface microstructure of physical samples is therefore not as regular as the idealized surface illustrated in Figure 1. In reality, it is rather rough as illustrated in Figure 5. Roughnesses have been measured for physical samples in previous work using optical profilometry [9, 16].

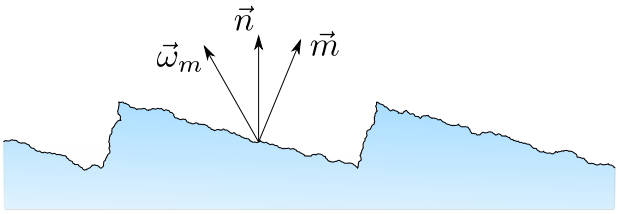


Figure 5. Rough surface.

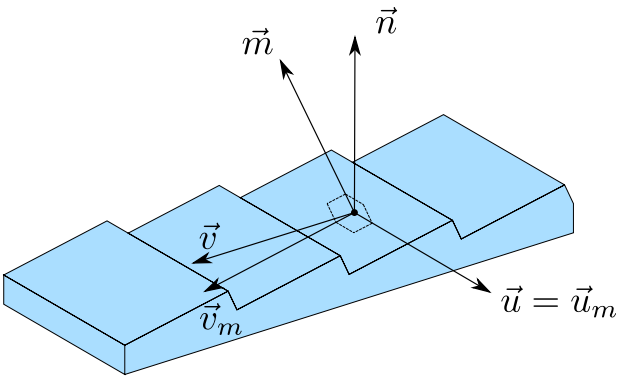


Figure 6. Orthonormal basis formed by $\vec{u}_m, \vec{v}_m, \vec{m}$.

These measurements reveal that the ridges are certainly not smooth. Thus, at a given point $\boldsymbol{x}$ on the microsurface the normal $\vec{\omega}_m$ is usually slightly different from the pitch normal $\vec{m}$.

The distribution of normals $D(\vec{\omega}_m)$ statistically describes the orientation of these irregularities across the microsurface. Many microfacet distribution functions have been defined over the years [19, 3, 4, 20]. In order to highlight the anisotropic nature of the surface we are working with, we use the anisotropic Beckmann distribution function [5], which is defined by

$$D(\vec{\omega}_m) = \frac{\chi^+(\vec{\omega}_m \cdot \vec{m})}{\pi \alpha_u \alpha_v \cos^4(\theta_m)} \times \exp\left(-\tan^2\theta_m \left(\frac{\cos^2\phi_m}{\alpha_u^2} + \frac{\sin^2\phi_m}{\alpha_v^2}\right)\right). \quad (3)$$

This distribution function is centred around the pitch normal $\vec{m}$, and the parameters a_u and a_v represent the stretching coefficients of the distribution along the $\vec{u}_m$ and $\vec{v}_m$ directions, respectively, see Figure 6. Together, $\vec{u}_m, \vec{v}_m, \vec{m}$ form an orthonormal basis and the microsurface normal $\vec{\omega}_m$ can be written in spherical coordinates as

$$\begin{aligned}\vec{\omega}_m &= \sin(\theta_m)\cos(\phi_m)\vec{u}_m \\ &\quad + \sin(\theta_m)\sin(\phi_m)\vec{v}_m + \cos(\theta_m)\vec{m}.\end{aligned}$$

3.3. Macrosurface and Microfacet BRDFs

The normal distribution function $D(\vec{\omega}_m)$ and the geometrical attenuation function $G(\vec{\omega}_i, \vec{\omega}_o, \vec{m})$ are combined into a macrosurface BRDF using [20]

$$\begin{aligned}f_M(\boldsymbol{x}, \vec{\omega}_i, \vec{\omega}_o) &= \int f_m(\vec{\omega}_i, \vec{\omega}_o, \vec{\omega}_m) D(\vec{\omega}_m) G(\vec{\omega}_i, \vec{\omega}_o, \vec{\omega}_m) \\ &\quad \times \left|\frac{\vec{\omega}_i \cdot \vec{\omega}_m}{\vec{\omega}_i \cdot \vec{n}}\right| \left|\frac{\vec{\omega}_o \cdot \vec{\omega}_m}{\vec{\omega}_o \cdot \vec{n}}\right| d\vec{\omega}_m. \quad (4)\end{aligned}$$

For the microfacet BRDF f_m, we assume that a microfacet is smooth so that it reflects and refracts light as a perfectly specular material. Reflection is described by one BRDF f_m^r and some of the refracted light returns due to subsurface scattering. We approximate this part by another BRDF f_m^{ss}. The function f_m is then defined by

$$f_m(\vec{\omega}_i, \vec{\omega}_o, \vec{\omega}_m) = f_m^r(\vec{\omega}_i, \vec{\omega}_o, \vec{\omega}_m) + f_m^{ss}(\vec{\omega}_i, \vec{\omega}_o, \vec{\omega}_m). \quad (5)$$

These BRDFs are based on a directional Dirac delta-function δ (just like the BRDF of a perfect mirror). We use Fresnel reflectance F_r as the specular reflectance and include a change of coordinates to enable integration over microfacet normals [20]. We then have

$$f_m^r(\vec{\omega}_i, \vec{\omega}_o, \vec{\omega}_m) = F_r(\vec{\omega}_i, \vec{\omega}_m) \frac{\delta(\vec{\omega}_h, \vec{\omega}_m)}{4(\vec{\omega}_i \cdot \vec{\omega}_h)^2},$$

where $\vec{\omega}_h = (\vec{\omega}_o + \vec{\omega}_i)/|\vec{\omega}_o + \vec{\omega}_i|$ is the half vector of reflection.

Although subsurface scattering happens for many BRDF inputs, we limit our model to only include subsurface scattering of the light that was lost to refraction in the reflection case. This light is certainly missing and including it is a first step. This makes our model similar to the BRDF approximation of subsurface scattering described by Jensen et al. [6]. We have

$$f_m^{ss}(\vec{\omega}_i, \vec{\omega}_o, \vec{\omega}_m) = F_t(\vec{\omega}_i, \vec{\omega}_m) F_t(\vec{\omega}_o, \vec{\omega}_m) \frac{\rho_d}{\pi} \frac{\delta(\vec{\omega}_h, \vec{\omega}_m)}{4(\vec{\omega}_i \cdot \vec{\omega}_h)^2},$$

where $F_t = 1 - F_r$ is the Fresnel transmittance, and ρ_d is the diffuse reflectance of the material.

By inserting Eq. 5 into Eq. 4, we arrive at our macrosurface BRDF:

$$f(\boldsymbol{x}, \vec{\omega}_i, \vec{\omega}_o) = f_r(\boldsymbol{x}, \vec{\omega}_i, \vec{\omega}_o) + f_{ss}(\boldsymbol{x}, \vec{\omega}_i, \vec{\omega}_o), \quad (6)$$

where the reflection term is

$$f_r(\boldsymbol{x}, \vec{\omega}_i, \vec{\omega}_o) = \frac{F_r(\vec{\omega}_i, \vec{\omega}_h)}{4\,|\vec{\omega}_i \cdot \vec{n}|\,|\vec{\omega}_o \cdot \vec{n}|} G(\vec{\omega}_i, \vec{\omega}_o, \vec{\omega}_h) D(\vec{\omega}_h)$$

and the subsurface scattering term is

$$\begin{aligned}f_{ss}(\boldsymbol{x}, \vec{\omega}_i, \vec{\omega}_o) &= \frac{\rho_d}{\pi} F_t(\vec{\omega}_o, \vec{\omega}_h) F_t(\vec{\omega}_i, \vec{\omega}_h) \\ &\quad \times \frac{G(\vec{\omega}_i, \vec{\omega}_o, \vec{\omega}_h) D(\vec{\omega}_h)}{4\,|\vec{\omega}_i \cdot \vec{n}|\,|\vec{\omega}_o \cdot \vec{n}|}.\end{aligned}$$

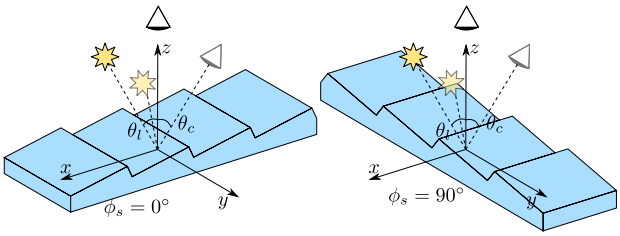


Figure 7. Configuration of the experiment for measuring contrast.

4. Experiments

We test our model by investigating its ability to predict the contrast measurements by Regi et al. [16]. These were conducted by photographing the surface before and after rotating the microstructure 90° around its macrosurface normal axis. Figure 7 illustrates the configuration of this experiment. They observed the samples with a digital microscope modified to hold an LED light source at a fixed position relative to the camera so that the angle between the camera and the light source was constant: $\theta_l = 10°$.

The parameters considered in the experiment are: the ridge angle θ_r which could assume the values 5°, 10°, 15°, and 20°; the camera tilting angle θ_c with values $-20°$, $-10°$, 0°, 10°, and 20°, and the azimuthal angle of rotation of the structure ϕ_s with values 0°, 90°, and 180°. The radiant exposure was measured under constant lighting conditions and varying parameters. The contrast was then evaluated as the difference between the measurements at positions 0° and 90° and between 90° and 180° for ϕ_s.

To predict these contrast measurements, we reproduced the same settings in a rendering framework and measured the radiant exposure $\left[\frac{J}{m^2}\right]$ (up to an unknown scaling factor k). Renderings were based on the BRDF described in the previous section and we compare our contrast measurements with the results presented by Regi et al. [16] in the following section.

5. Results

Our contrast predictions are compared with the measured contrasts in Figure 8. The mean contrast was evaluated by keeping one parameter constant and averaging all the contrasts obtained by varying the other parameters.

As in the measurements, we find maximum contrast for zero tilting angle ($\theta_c = 0°$) and decreasing contrast when the camera is tilted. We also find that the anisotropic structure of the surface makes the contrast between the azimuthal angles 0° and 90° stronger than the contrast between 90° and 180°. With respect to the ridge angle θ_r, our model predicts the highest contrast with a 5° angle. This is theoretically plausible as a five degrees ridge angle should leave most microfacets with a normal so that light is reflected in the macrosurface normal direction when $\theta_c = 0°$.

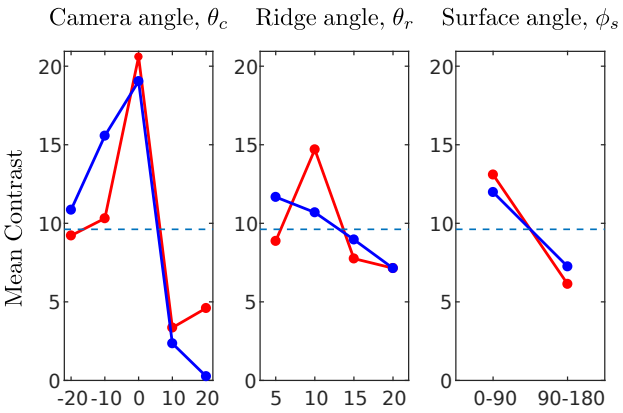


Figure 8. Comparison of mean contrasts for different values of the parameters θ_c, θ_r, and ϕ_s. Measured contrasts [16] are in red and our predicted contrasts are in blue.

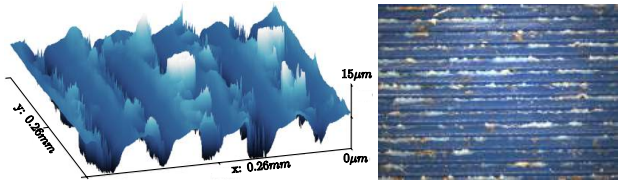


Figure 9. Small part of a manufactured sample. Visualization of a height map acquired with an industrial laser confocal microscope (left) and a microscope image (right).

The most significant difference between prediction and measurement is that measurements found highest contrast for a ridge angle of $\theta_r = 10°$. We think that this result might be caused by the presence of noise in the surface structure due to the manufacturing process. To support this conjecture, we have produced samples similar to the ones in previous work [9, 16] and investigated the microstructure of the tool and the manufactured plastic sample. Figure 9 shows a 3D visualization of height data captured with a 3D laser confocal scanner and a microscope image both of the plastic surface. While the original surfaces produced by Regi et al. [16] may have been higher quality, there is no doubt that the manufacturing process produces inaccuracies both in the tool and in the sample microstructure. In the tool, we have observed small burrs, especially along ridge edges. These burrs have a tendency to leave residues of material on the surface and create substantial artifacts. The white bulky peaks in Figure 9 are examples of such artifacts. These imperfections in the surface become more significant for small ridge angles and may easily hide the signal from the ridged structure in noise. We believe this is a plausible explanation for this deviation between prediction and measurement.

6. Discussion and Future Work

We have developed a new model for predicting the reflectance properties of an engineered anisotropic surface

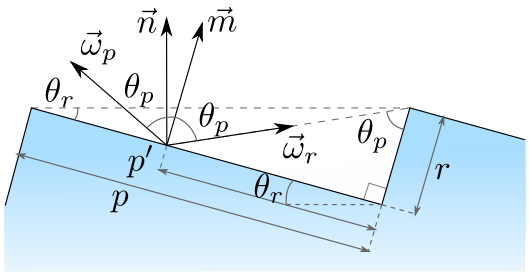


Figure 10. The surface fraction masked by the ridged structure is given by the ratio between p and p'.

made of parallel micro ridges. Our model provides a BRDF based on microfacet theory including an expression for the geometrical attenuation function. The BRDF describes our particular type of ridged surface, but a similar procedure could be employed to model other engineered surface microstructures. We validated our model by comparing with experimental measurements from previous work. Our results are quite similar to the measurements, but we observed some deviations. If deviations are due to manufacturing artifacts, as we conjecture, our model is useful as a tool for computer vision based quality inspection of optical functional surfaces of this kind. In addition, our model provides many opportunities for optimizing surface structure with the objective of maximizing contrast, for example. It is significantly easier to modify microstructure configuration in simulation as compared with experiment.

In the future, we would like to further support our conjecture that contrast measurements converge to predicted contrasts as sample quality improves. This will be investigated as tooling and manufacturing processes improve to provide higher quality samples. Moreover, comparison of anisotropic BRDF measurements with predicted values would also be interesting as an alternative to the more overall contrast measurements.

A. The G_1 Function for a Ridged Surface

This appendix provides some details about the derivation of the geometrical attenuation function described in Eq. 2.

The value of the function G_1 is given by the ratio between the portion of the pitch surface visible from a given direction $\vec{\omega}_p$ and the total pitch surface. Figure 10 provides some elements that are useful for the derivation of Eq. 2. The vector $\vec{\omega}_r$ represents the reflection of $\vec{\omega}_p$ around the surface normal $\vec{m}$, θ_r is the ridge angle and θ_p is the angle between $\vec{\omega}_p$ and $\vec{m}$, p and r represent respectively the length of the pitch and the length of the ridge, and p' represents the length of the portion of pitch surface for which the reflection vector $\vec{\omega}_r$ is blocked by the ridge.

Now, G_1 is described by

$$G_1(\vec{\omega}_p, \vec{m}) = 1 - \frac{p'(\vec{\omega}_p, \vec{m})}{p}, \tag{7}$$

and the value of p' is

$$p'(\vec{\omega}_p, \vec{m}) = r \tan\theta_p = p \tan\theta_r \tan\theta_p. \tag{8}$$

Then, by inserting Eq. 8 into Eq. 7, we have

$$G_1(\vec{\omega}_p, \vec{m}) = 1 - \tan\theta_r \tan\theta_p. \tag{9}$$

Since the value of p' might become greater than p for certain combinations of angles θ_r and θ_p, we modify Eq. 9 and get

$$G_1(\vec{\omega}_p, \vec{m}) = \chi^+\left(\frac{\vec{\omega}_p \cdot \vec{m}}{\vec{\omega}_p \cdot \vec{n}}\right) \times \left[1 - \min\left(1, |\tan\theta_r \tan\theta_p|\right)\right]. \tag{10}$$

In a similar way, it can be shown that for an arbitrary direction $\vec{\omega}$ not lying in the plane spanned by the $\vec{n}$ and $\vec{m}$ Eq. 10 is still valid and depends only on the projection vector $\vec{\omega}_p$ and the surface normal $\vec{m}$.

References

[1] C. Acikgoz, M. A. Hempenius, J. Huskens, and G. J. Vancso. Polymers in conventional and alternative lithography for the fabrication of nanostructures. *European Polymer Journal*, 47(11):2033–2052, November 2011.

[2] M. Ashikmin, S. Premože, and P. Shirley. A microfacet-based BRDF generator. In *Proceedings of SIGGRAPH 2000*, pages 65–74. ACM/Addison-Wesley, 2000.

[3] J. F. Blinn. Models of light reflection for computer synthesized pictures. *Computer Graphics (Proceedings of SIGGRAPH 77)*, 11(2):192–198, July 1977.

[4] R. L. Cook and K. E. Torrance. A reflectance model for computer graphics. *ACM Transactions on Graphics*, 1(1):7–24, January 1982.

[5] E. Heitz. Understanding the masking-shadowing function in microfacet-based brdfs. *Journal of Computer Graphics Techniques*, 3(2):48–107, June 2014.

[6] H. W. Jensen, S. R. Marschner, M. Levoy, and P. Hanrahan. A practical model for subsurface light transport. In *Proceedings of SIGGRAPH 2001*, pages 511–518. ACM, August 2001.

[7] Y. Lan, Y. Dong, F. Pellacini, and X. Tong. Bi-scale appearance fabrication. *ACM Transactions on Graphics (Proceedings of SIGGRAPH 2013)*, 32(4):145:1–145:11, July 2013.

[8] A. Levin, D. Glasner, Y. Xiong, F. Durand, W. Freeman, W. Matusik, and T. Zickler. Fabricating BRDFs at high spatial resolution using wave optics. *ACM Transactions on Graphics (Proceedings of SIGGRAPH 2013)*, 32(4):144:1–144:13, July 2013.

[9] D. Li, Y. Zhang, F. Regi, G. Tosello, M. H. Madsen, J. B. Nielsen, H. Aanæs, and J. R. Frisvad. Process chain for fabrication of anisotropic optical functional surfaces on polymer components. In *Proceedings of the 17th EUSPEN International Conference*, June 2017.

[10] T. Malzbender, R. Samadani, S. Scher, A. Crume, D. Dunn, and J. Davis. Printing reflectance functions. *ACM Transactions on Graphics*, 31(3):20:1–20:11, May 2012.

[11] W. Matusik, B. Ajdin, J. Gu, J. Lawrence, H. Lensch, F. Pellacini, and S. Rusinkiewicz. Printing spatially-varying reflectance. *ACM Transactions on Graphics (Proceedings of SIGGRAPH Asia 2009)*, 28(5):128:1–128:9, December 2009.

[12] G. McGunnigle. Shape recovery of anisotropic metal surfaces. *JOSA A*, 26(10):2235–2242, October 2009.

[13] D. Miller, P. Schor, and P. Magnante. Optics of the normal eye. In M. Yanoff and J. S. Duker, editors, *Ophthalmology*, chapter 2.6, pages 52–60. Mosby/Elsevier, 3rd edition, 2009.

[14] T. Pereira, C. L. Leme, S. Marschner, and S. Rusinkiewicz. Printing anisotropic appearance with magnetic flakes. *ACM Transactions on Graphics (Proceedings of SIGGRAPH 2017)*, 36(4):123:1–123:10, July 2017.

[15] P. Poulin and A. Fournier. A model for anisotropic reflection. In *Computer Graphics (Proceedings of SIGGRAPH 90)*, volume 24, pages 273–282, August 1990.

[16] F. Regi, D. Li, Y. Zhang, J. B. Nielsen, M. H. Madsen, G. Tosello, J. R. Frisvad, and H. Aanæs. A comparison of reflectance properties on polymer micro structured functional surface. In *Proceedings of the 17th EUSPEN International Conference*, June 2017.

[17] B. Smith. Geometrical shadowing of a random rough surface. *IEEE Transactions on Antennas and Propagation*, 15(5):668–671, September 1967.

[18] K. E. Torrance and E. M. Sparrow. Off-specular peaks in the directional distribution of reflected thermal radiation. *Journal of Heat Transfer*, 88(2):223–230, 1966.

[19] K. E. Torrance and E. M. Sparrow. Theory for off-specular reflection from roughened surfaces. *Journal of the Optical Society of America*, 57(9):1105–1114, September 1967.

[20] B. Walter, S. Marschner, H. Li, and K. Torrance. Microfacet models for refraction through rough surfaces. In *Proceedings of Eurographics Symposium on Rendering (EGSR 2007)*, pages 195–206. The Eurographics Association, 2007.

[21] J. Wang, S. Zhao, X. Tong, J. Snyder, and B. Guo. Modeling anisotropic surface reflectance with example-based microfacet synthesis. In *ACM Transactions on Graphics (Proceedings of SIGGRAPH 2008)*, volume 27, pages 41:1–41:9, August 2008.

[22] G. J. Ward. Measuring and modeling anisotropic reflection. *Computer Graphics (Proceedings of SIGGRAPH 92)*, 26(2):265–272, July 1992.

[23] T. Weyrich, P. Peers, W. Matusik, and S. Rusinkiewicz. Fabricating microgeometry for custom surface reflectance. *ACM Transactions on Graphics (Proceedings of SIGGRAPH 2009)*, 28(3):32:1–32:6, August 2009.

CONTRIBUTION C

Functionality characterization of injection moulded micro-structured surfaces

Link to the definitive published version which is preferred but cannot be inserted here: https://doi.org/10.1016/j.precisioneng.2019.07.014

Francesco Regi, Mads Emil Brix Doest, Dario Loaldi, Dongya Li, Jeppe Revall Frisvad, Guido Tosello, and Yang Zhang. "Functionality characterization of injection moulded micro-structured surfaces". In: *Precision Engineering* 60 (Nov. 2019), pp. 594 –601. DOI: 10.1016/j.precisioneng.2019.07.014

CONTRIBUTION D

Microstructure Control in 3D Printing with Digital Light Processing

Volume xx (*200y*), *Number z, pp. 1–12*

Microstructure Control in 3D Printing with Digital Light Processing

A. Luongo, V. Falster, M. B. Doest, M. M. Ribo, E. R. Eiriksson, D. B. Pedersen, and J. R. Frisvad

Technical University of Denmark

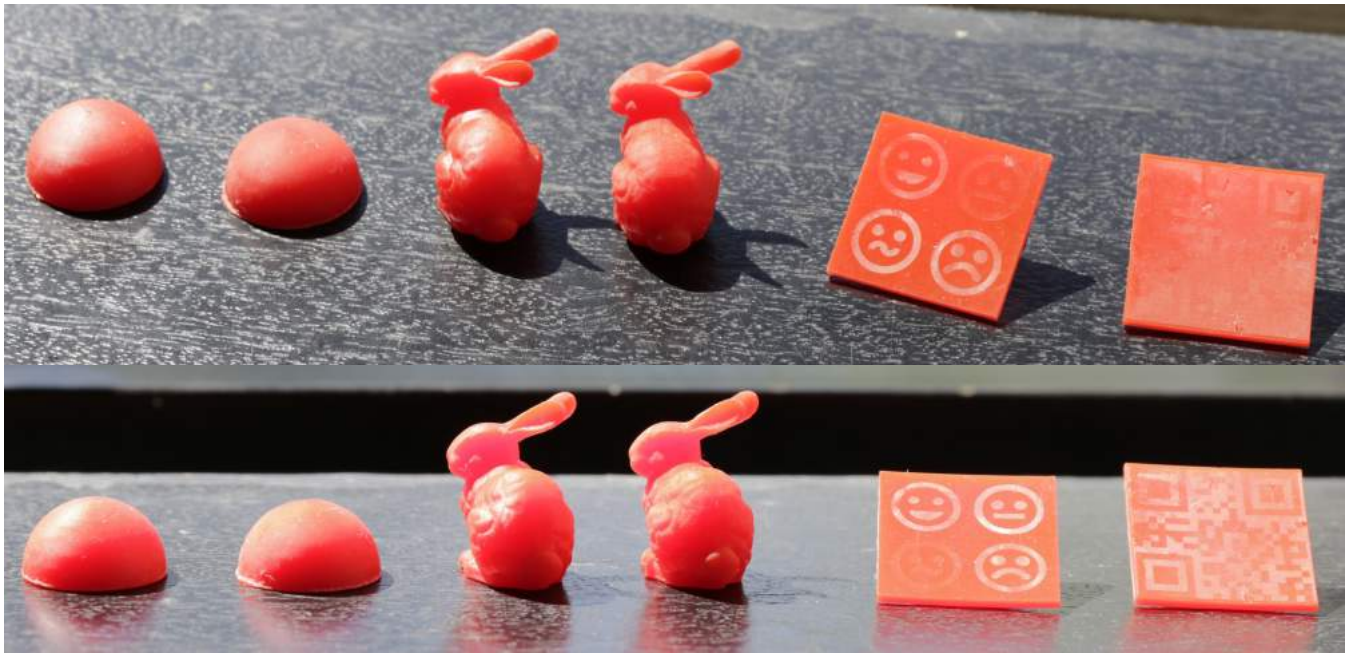


Figure 1: *Hemispheres and bunnies with smooth and rough surfaces, and flat samples (smileys and QR code) with spatially varying anisotropic reflectance. The scene is observed from two different directions to exhibit the anisotropy. The sun is used as a directional light source. Each item was printed in a one-step process using the presented technique.*

Abstract

Digital light processing stereolithography is a promising technique for 3D printing. However, it offers little control over the surface appearance of the printed object. The printing process is typically layered, which leads to aliasing artifacts that affect surface appearance. An antialiasing option is to use grayscale pixel values in the layer images that we supply to the printer. This enables a kind of subvoxel growth control. We explore this concept and use it for editing surface microstructure. In other words, we modify the surface appearance of a printed object by applying a grayscale pattern to the surface voxels before sending the cross-sectional layer images to the printer. We find that a smooth noise function is an excellent tool for varying surface roughness and for breaking the regularities that lead to aliasing. Conversely, we also present examples that introduce regularities to produce controlled anisotropic surface appearance. Our hope is that subvoxel growth control in stereolithography can lead 3D printing toward customizable surface appearance. The printing process adds what we call ground noise to the printed result. We suggest a way of modeling this ground noise to provide users with a tool for estimating a printer's ability to control surface reflectance.

CCS Concepts

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

1. Introduction

While 3D printers can often print geometric features in high quality, they lack the ability to control surface appearance by modifying roughness and reflectance properties. The ability to produce models with region-specific surface properties is crucial for artists and developers to properly design the appearance of a part. In the prototyping stage of product development, additive manufacturing is commonly used to produce parts in order to evaluate the final aesthetics of a product. For a part to look like a designed digital model, however, additional surface processing is often required. We propose a method for better control of printed surface properties to enable customization of the final appearance of a printed part.

The printing technology we work with is based on photopolymerization, which refers to the curing of liquid photo-reactive resins (photopolymers) using light. The light is usually in the ultraviolet range of wavelengths. This process is used for 3D printing with stereolithography, where a light source selectively illuminates a photopolymer to produce a solid object with a user-defined shape. If a digital light processing (DLP) projector is used as the source, the technique is referred to as DLP printing. In this case, we can specify the user-defined shape as a volume. The photopolymer is contained inside a vat and at each step a building platform is raised or lowered, depending on the setup of the DLP printer, in order to expose only a thin layer of liquid photopolymer to the projector. Each slice of the volume is then projected onto the photopolymer to produce a layer of the 3D print consisting of solidified polymer in all the pixels of the slice with value one (white voxels). In the context of DLP printing, we provide an investigation of the use of grayscale voxel values to control surface microstructure. Figure 1 displays some of our results.

Commercial 3D printers improve continually in terms of the resolution and the complexity of the geometries that can be printed. Nevertheless, the final surface appearance is typically controlled through the use of different print materials, deposition of different inks, and postprocessing of the surface. Samples with different reflectance properties can be printed directly in a one-step process, but the microstructure of the surface is then defined by the employed 3D printing technique. For example, in a material-extrusion based printer, the sample surface will exhibit layering artifacts, while a powder based print will have a grainy surface. A DLP printer can produce smooth flat surfaces, but on vertical and curved surfaces it will produce staircase artifacts. Even if the layers are so thin that we cannot see them with our naked eyes, the layered structure still produces moiré patterns and reflects light with a glean at certain angles. To get a different appearance, such as smooth or matte, the printer must produce a more detailed geometry with smaller features. The resolution of the 3D printer typically sets the limitation and prevents us from obtaining the desired result.

In this work, we show how the use of grayscale patterns greatly increases the capabilities of a DLP printer, and how it enables us to print microfeatures and patterns on the surface of a sample in a one step process without changing the macroscopic geometry of the printed part. By using this technique, we can modify the roughness and surface appearance of a print without changing materials or applying postprocessing to the sample.

2. Related Work

Fabrication of microgeometry to obtain custom surface reflectance was pioneered by Weyrich et al. [WPMR09]. They point at many interesting applications and fabricate custom microgeometry using a micro milling approach. In a 3D printing context, a 5-axis micro milling machine can produce free-form surfaces with fairly small features. However, due to the kinematics of the milling process, it is difficult to control the surface roughness [ABRK17]. In another early technique, Matusik et al. [MAG*09] use different inks in different halftoning patterns to print a surface with spatially-varying reflectance properties. This technique is however restricted to printing on planar surfaces, and the microstructure that can be printed depends on the reflectance properties of the employed inks.

Different ways of extending these early techniques have been tested. Malzbender et al. [MSS*12] print on a paper with a static microstructure and let the selective depositing of ink control the surface reflectance. More generally, Baar et al. [BBS15] study the link between variation of print parameters and local control of the gloss appearance in a printout. However, they only consider printing of flat images. Lan et al. [LDPT13] use a 3D printer based on material jetting to produce patches with oriented facets and then coat them with glossy inks using a flatbed UV printer. However, the facets in the patches are visible to the naked eye (140 μm by 140 μm) and the fabrication process requires two steps. The use of the flatbed printer puts a constraint on the curvature of the surface that the inks can be applied to. Thus, when applying this method to a 3D surface, the object is divided into several parts that are stitched together in a post-process after inks have been deposited using the flatbed printer. Another approach requiring two steps is by Rouiller et al. [RBK*13]. They use another 3D printer based on material jetting to print microfacetted transparent domes that they stick onto a colored model, which was 3D printed using a powder bed printer. In this way, each dome modifies the reflectance in the local area where it is attached. As opposed to these techniques, we present a one-step approach where the fabrication of surface microstructure is integrated into the 3D printing process. The material jetting printers (PolyJet technology) employed in this previous work can only print binary voxels (material or not). Consequently, they do not support the grayscale voxel values that we can use when employing a vat polymerization based DLP printer.

Levin et al. [LGX*13] present a technique for printing microstructure small enough to create reflectance functions based on wave interference effects. Their technique is based on photolithography, which is a very precise but also very costly process that requires a special wafer coated by photoresist. Photolithography is currently not available as a 3D printing technique.

Pereira et al. [PLMR17] propose an entirely different approach, where magnetic microflakes are embedded into a photopolymer and controlled during printing using electromagnets. While they obtain interesting results, the magnetic flakes are significantly harder to control than our surface microstructure based on grayscale values in the projected cross-sectional images.

Use of grayscale values in DLP printing is not entirely new. Mostafa et al. [MQM17] explore to what extent grayscale values can improve the dimensional accuracy of an Autodesk Ember printer. This use case has also been investigated internally

at Autodesk [Gre16], where they improve printing fidelity using grayscale values computed with antialiasing techniques. The work presented by Greene [Gre16] is the work most closely related to ours. Greene even mentions in passing that random noise can be used to break moiré patterns and to produce a matte surface. However, to the best of our knowledge, we are the first to more carefully modify surface roughness and reflectance properties of 3D printed objects by applying grayscale patterns across surface voxels.

Some work has been done to control the subsurface scattering and absorption properties of fabricated objects [DWP*10, HFM*10, PRJ*13, ESZ*17]. In our case, these properties are determined by the photopolymer selected for the print job. We consider it an interesting challenge for future work to investigate ways of controlling the scattering properties of a photopolymer.

3. Subvoxel Growth

The resolution of DLP printing is typically in the range from 15 to 100 μm [LCR*17]. It depends on the quality and pixel resolution of the digital micromirror device (DMD) chip of the projector and on the step-precision of the building plate. It is possible to use grayscale images as input for the projector to obtain subvoxel accuracy [Gre16, MQM17]. The principle behind this idea is that the solidification process of the resin depends on the amount of UV light received, and this amount can be changed by varying either the period of time for which an image is projected (exposure time) or the intensity of the light. With grayscale values as input for the projector, we vary the intensity and thus control the growth of each voxel. This approach can be used to produce very small features and patterns on the surface of a 3D printed sample. If applied properly, the grayscale values modify the microscopic surface properties of an object without modifying its macroscopic geometrical structure.

3.1. Subvoxel Control

The relation between grayscale values and voxel growth is crucial if we are to print an arbitrary microscopic pattern with high accuracy. If we project an even slope of all the grayscale values (pixel intensity values from black to white), we would ideally see the same even slope being printed. If this were the case, voxels would grow proportionally with the grayscale values.

Unfortunately, the photopolymerization is initiated only when a critical energy level is reached, and the cure depth then follows a logarithmic curve with increasing energy [Jac92, LPA01, Ben17]. Thus, we can determine the relationship between pixel intensity and voxel growth. With τ denoting the thickness of a print layer, the cure depth and thus the voxel growth height is

$$\tau f(I) = \begin{cases} \alpha + \beta \log(I-\gamma), & \text{for } I > e^{-\alpha/\beta} + \gamma, \\ 0 \quad , & \text{for } I \le e^{-\alpha/\beta} + \gamma, \end{cases} \tag{1}$$

where I is the pixel intensity, and α, β, and γ are parameters that need to be fitted for a particular photopolymer.

Through inversion of the function f, we obtain a mapping to the proportionality relation, which significantly eases control of the voxel growth. We have

$$f^{-1}(I) = \begin{cases} e^{\frac{\tau I - \alpha}{\beta}} + \gamma, & \text{for } I > 0, \\ 0 \quad , & \text{for } I = 0, \end{cases} \tag{2}$$

and using $f^{-1}(I)$ as the grayscale values of the pixels in a projection, the printer prints voxels of height τI. This is illustrated in Figure 2. Greene [Gre16] presented a similar result, but they used a quadratic f function while suggesting that a logarithmic function seems a better choice. We found the right f function by considering the photopolymerization cure depth.

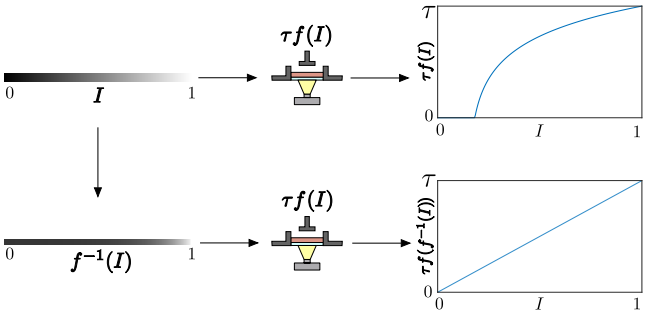


Figure 2: *Inversion of non-linear voxel growth to have printed voxel height proportional to grayscale pixel intensity, I.*

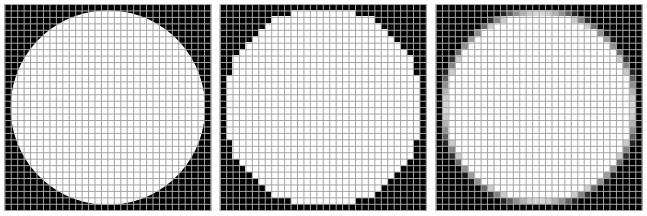


Figure 3: *A desired circular print layer geometry (left), its rasterization according to the resolution of the projector (middle), and the same layer with grayscale values for antialiasing (right).*

3.2. Grayscale Patterns

The ability to control voxel growth using a linear scale of grayscale values enables us not only to improve fidelity and reduce aliasing artifacts, as demonstrated by Greene [Gre16], it also enables us to print smooth microfeatures in a single layer and thereby modify the reflectance properties of the surface.

3.2.1. Antialiasing

When printing an object, we have to slice the geometry to generate an image for each layer. Slices are obtained by rasterizing the geometry, and if no measures are taken, spatial aliasing will be present along edges of the layers in the form of pixelated boundaries, see Figure 3. Grayscale values based on supersampling (in all three dimensions) can be used to counteract this effect and produce a smoother surface [Gre16]. However, this is not enough to completely remove staircase artifacts in a surface. These artifacts lead to visible reflectance anisotropy and moiré patterns.

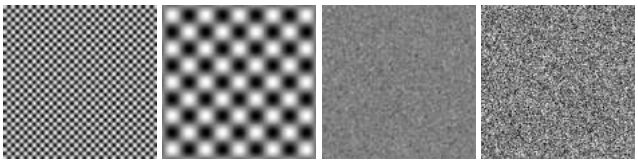


Figure 4: *Sinusoidal patterns with different wavelengths (leftmost with $\lambda_u = \lambda_v = 100$ μm and middle left with $\lambda_u = \lambda_v = 400$ μm). Sparse convolution noise with different amplitude and frequency factors (middle left with $A = 0.625$ and $B = 16$ and rightmost with $A = 3$ and $B = 32$). Both types of patterns are useful for controlling roughness. Due to its irregularities, the noise function is also useful for antialiasing.*

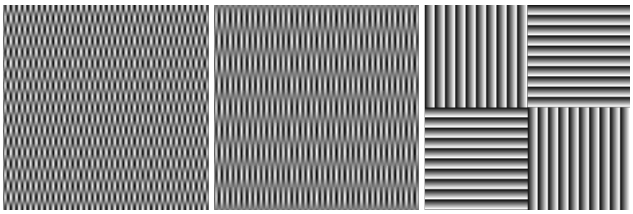


Figure 5: *Sinusoidal patterns with different wavelengths along the two axes (left with $\lambda_u = 50$ μm and $\lambda_v = 200$ μm and middle with $\lambda_u = 50$ μm and $\lambda_v = 400$ μm) and sequences of parallel ridges (right). These 2D patterns are useful for printing anisotropic surface roughness and reflectance contrast.*

Greene [Gre16] suggests the use of Gaussian smoothing that produces grayscale values in a thick band around the edges to further reduce aliasing. A broad Gaussian smoothing is however likely to also smoothen the macroscopic geometry of the object if the surface is not spherical. This would compromise object fidelity. Another suggestion by Autodesk [Gre16] is to add random noise to all the grayscale values. This breaks the moiré patterns, but it also leads to a matte surface. In other words, when printing in 3D, existing work leaves us with the choice of an aliased or a matte surface appearance. In the following, we demonstrate how a smooth low-amplitude solid noise function can be used to break moiré patterns while retaining surface smoothness. In addition, we explore the use of procedural methods for inserting grayscale values in surface voxels to control the surface microstructure.

3.2.2. Reflectance Properties

The roughness of a surface is given by its microstructure. The features are so small that they are only individually visible at the microscale, but they affect the macroscopic surface appearance. Our goal is to apply grayscale patterns along the surface of an object to print surfaces with different roughnesses, going from smooth to almost diffuse, and also to print spatially varying anisotropic reflectance properties.

As rough surfaces are characterized by having a distribution of microfacet normals pointing in various directions, one way to obtain isotropic roughness is to use a curved surface [TR75]. We therefore test a grayscale pattern with surface voxel values set according to a (2D) sinusoidal function running along the surface. The function is

$$I(u,v) = \frac{1}{2}\sin\left(\frac{2\pi}{\lambda_u}u\right)\sin\left(\frac{2\pi}{\lambda_v}v\right) + \frac{1}{2}, \tag{3}$$

where u and v are parameters measuring physical length in a uniform parametrization of the surface, so that λ_u and λ_v represent the wavelengths along these two dimensions. The wavelengths of the sinusoid then control the roughness of the surface, see Figure 4. This kind of grayscale pattern will generate a periodic sequence of micro-cavities and micro-bumps on the 3D printed object, and this structure will produce a rough surface when the frequency of the sinusoid is high (more bumps and cavities), and a smooth surface when the frequency is low.

An issue with the sinusoidal surface is its regularity. Since the function is regular, it does not entirely prevent the aliasing problems due to layered printing. We therefore decided to also use a smooth noise function, as it is irregular but produces a similar effect in terms of the microfacet normal distribution. To avoid the grid-aligned artifacts seen in Perlin noises [Per85, Per02, MSRG12], we employ a solid sparse convolution noise (Appendix A). The difference between sinusoidal patterns and noise slices is illustrated in Figure 4. By controlling the frequency and amplitude of the noise function, we are able to obtain smooth and rough surfaces with very few staircase artifacts (hemispheres and bunnies in Figure 1).

We print anisotropic reflectance properties using a 2D sinusoidal function with different frequencies along the two axes, or a sequence of parallel ridges, as described by Luongo et al. [LFD*17], see Figure 5. These patterns are useful for producing anisotropic reflectance contrast (smileys and QR code in Figure 1). While we only test these patterns on a 2D surface, they could be texture mapped onto a curved surface to obtain a 3D surface with anisotropic reflectance. Texture coordinates for a given model can be generated using a 3D modeling tool such as Maya or Blender. If we want to avoid this task, a solid noise function (Appendix A) can be stretched along the tangent space of a 3D surface using line integral convolution [BSH97]. To obtain a consistently oriented tangent space without use of texture coordinates, we can use the function for building an orthonormal basis by Frisvad [Fri12].

3.3. Assessing Reflectance Controllability

We assess how well our method controls the reflectance properties of a printed surface using two different approaches. For anisotropic microstructure, we predict the expected contrasts in light reflection when the surface is illuminated and viewed from different directions. We do this by rendering the surface appearance due to the varying microstructures using analytic BRDF models derived for those specific microstructures. For the ridged structure in Figure 5, we use the model presented by Luongo et al. [LFD*17]. For the anisotropic sinusoidal patterns, we derived a new model, which is described in Appendix B. We then qualitatively compare the rendered images with photographs of printed samples. The comparison is not in terms of photorealism, but in terms of contrast in light reflection. For irregular noise-based microstructure, such as the patterns generated using sparse convolution noise (Figure 4), we compute the corresponding bidirectional reflectance distribution function (BRDF) using a path tracer. We path trace a represen-

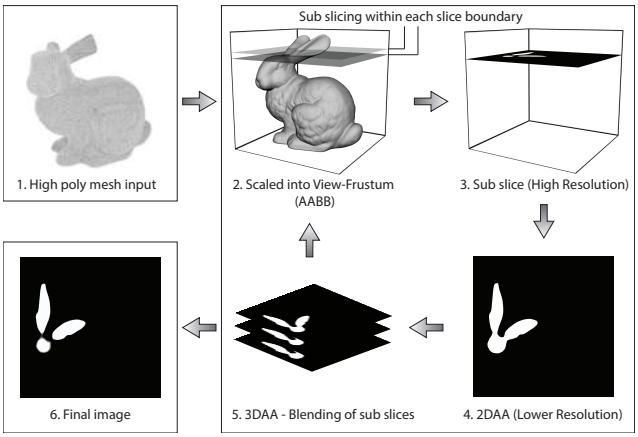


Figure 6: *Mesh slicing pipeline based on rasterization. Used for generating cross-sectional layer images for the DLP projector.*

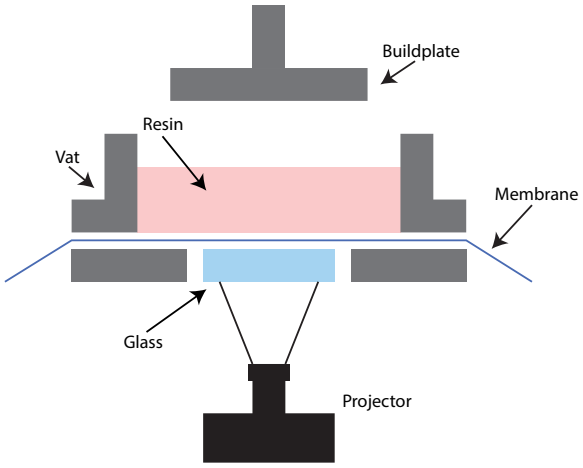


Figure 7: *Schematic of the homebuilt DLP printer.*

tative patch of the noise used as grayscale input for the printer. Measuring the printed microstructure using a microscope, we can then compare the BRDF of the desired microstructure with the BRDF of a corresponding printed microstructure.

Interestingly, Ribardière et al. [RBSM19] provide an algorithm for generating height fields with microstructure corresponding to the normal distributions used in popular analytic microfacet BRDF models [WMLT07]. These height fields can be used as grayscale maps in our printing process and would allow for assessments similar to ours but with the commonly used BRDFs. We leave this additional investigation for future work.

3.4. Mesh Slicing

To generate antialiased cross-sectional layer images for the DLP projector, we have tested two different approaches: one based on rasterization and one based on ray tracing, both running on the graphics processing unit (GPU). Our rasterization procedure is illustrated in Figure 6, and the different steps are described in the following paragraphs.

In both approaches, a closed triangle mesh is provided as input (step 1) and the print volume is represented by the view frustum of an orthographic camera placed above the mesh looking downwards. The background color is set to black and the frame buffer resolution is set to the projector resolution. The latter ensures that each pixel of a generated layer image corresponds to a voxel with physical dimensions as described in Section 4. To determine the number of slices that we need, we calculate the object height in number of voxels using the desired physical height of the printed object.

In rasterization, we slice the mesh by moving the near cutting plane of the camera through the print volume in steps of the print layer thickness (step 2). The far cutting plane is placed at the end of the print volume and depth testing is enabled. For frontfacing triangles, the color is based on a procedural texture (sinusoid or noise) but the fragment is only rendered to the color buffer if it is within the current layer. Frontfacing triangles behind the current layer are only rendered to the depth buffer. Backfacing triangles passing the depth test are rendered with a flat white color. For each slice, we generate a number of subslices (step 3) to include supersampling in the depth dimension.

In ray tracing, we trace a ray from the image plane through all surfaces until it reaches the front surface of the current layer. The ray keeps a counter for each intersection, so that the counter is even when the ray is outside the object, odd when inside. A ray is then traced in the same direction from the front to the back of the layer. The fraction of the distance traveled by this ray that was also inside the object provides a grayscale value for antialiasing in the depth direction. Combining this with jitter sampling of the ray origin in the camera pixel, we obtain grayscale values incorporating full 3D antialiasing. As in rasterization, the grayscale value is modulated by a procedural texture when the ray going through the layer intersects a frontfacing triangle.

In rasterization, antialiasing requires more passes. To have 2D antialiasing in each slice, we use hardware supported full screen antialiasing with four samples in each pixel (4xFSAA). This is done in eight times higher resolution and downsampled to the projector resolution (step 4). The subslices are then blended into the same frame buffer (step 5) to produce one antialiased cross-sectional layer image for the printer (step 6).

4. Experiments

We run our experiments on a homebuilt bottom-up DLP printer, which is based on the work of Jørgensen [Jør15]. A schematic of the printer is in Figure 7. The photopolymer resin is inside the vat. The building platform starts at the bottom of the vat and moves upwards during the printing process. The step precision of the building platform is 1 μm, which enables us to print very thin layers. A transparent membrane is placed at the bottom of the vat in order to separate the photopolymer from the glass. This is done to facilitate the peeling effect and the release of the sample from the vat when the platform is raised [PZNH16].

The DLP projector we use is a LUXBEAM Rapid System by Visitech equipped with a DMD chipset of the DLP9000 family by

Texas Instruments. It has an array of 2560×1600 micro-mirrors and pixel pitch of 7.54 μm. The projector is placed underneath the vat and can be raised and lowered to focus it. We use a projection lens from Visitech with a magnification factor of $1.0\times$, yielding an image pixel pitch of 7.54 μm, or alternatively a lens with a factor of $2.0\times$ and pixel pitch of 15.08 μm.

According to the manufacturer, the projector is more stable for high values of the UV LED amplitude, but even low values of UV LED amplitude can overcure the photopolymer in our setup. This would ruin the quality of the prints, so we equipped the projector with two absorptive neutral density filters from Thorlabs. Each filter transmits 10% of the incoming light, so that the amount of light reaching the photopolymer is 1% of the light emitted by the projector. In this way, we can use higher values of UV LED amplitude for our prints, which means that we get a more stable behavior from the projector (less flickering, for example).

The photopolymer we use is Industrial Blend (red) resin from Fun To Do. In order to inspect and measure the properties of our prints we used an optical measuring device based on focus-variation, Infinite Focus by Alicona, which can produce high-quality 3D measurements of the surfaces and measure the surface roughness with nanometer precision.

After the printing process, the sample is cleaned with isopropanol in an ultrasonic cleaner in order to remove any residual resin from the surface. We then do additional curing in a UV curing box to ensure that the sample has solidified properly, and to remove the risk of contamination when touching the sample.

Our setup enables us to print high resolution samples. However, the presence of the membrane, which mitigates peeling forces, is a source of some defects: when the membrane is installed on the glass, some wrinkles may be present and air can be trapped between the membrane and the glass causing the formation of bubbles. Such issues affect the final quality of the sample, where we sometimes observe bumps and scratches on the surface. Scratches start appearing as the membrane gets worn.

4.1. Parameter Calibration

The photopolymer curing process is determined by the intensity of the projected UV light, by the exposure time, and by the amount of resin that we want to cure (layer thickness). All these parameters vary for different materials, and a calibration operation is required in order to find the optimal configuration for a certain setup.

Based on previous experiments performed on the same printer [Rib17], we decided to use a value of $\tau = 18$ μm for the layer thickness. This value is small enough to give us microfeatures, which can affect the reflectance properties of an object without being visible to the naked eye, and it is thick enough so that the features created with grayscale images are not overexposed.

To calibrate the projector intensity and exposure time, we created a calibration sample with the same pattern repeated 36 times on the top surface, see Figure 8. For each of these 36 patterns, we use a different value of intensity or exposure time. One out of the 36 patterns has a physical size of 1920×1920 μm^2 and consists of four black-and-white checkerboards with different scales for the size of the squares. We first print a calibration sample with increasing UV LED amplitude for each pattern repetition while keeping the exposure time constant. The same experiment is then repeated with increasing exposure time while keeping the UV LED amplitude constant. A good combination of parameters is found when a pattern shows sharp features which are neither underexposed nor overcured. With this experiment, we found that for a layer thickness of $\tau = 18$ μm the optimal parameters of our setup are an UV LED amplitude of 230 and an exposure time of 3 seconds.

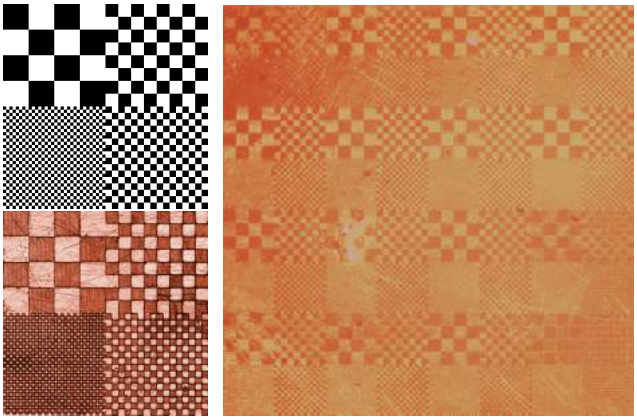


Figure 8: *Pattern used to calibrate projector parameters (top left) and microscope image of a printed pattern (bottom left). The pattern is composed of four black-and-white checkerboards at different scales, and it is repeated 36 times in a calibration sample. On the right, a microscope image with 16 of the 36 checkerboard pattern repetitions in a calibration sample.*

4.2. Voxel Height Measurements

As mentioned in Section 3.1, the relation between pixel intensity and growth of the corresponding voxel is logarithmic, Eq. 1. In order to apply our correction, Eq. 2, we need to find the values of the parameters α, β, and γ.

We printed several samples with a repeated linear grayscale gradient containing all the values from black to white, the upper left part of Figure 9 shows two examples. We then examined the samples with the Infinite Focus microscope and measured the surface with a vertical resolution of 0.4 μm. The collected data were used to find a fit for Eq. 1, see the lower left part of Figure 9, and we estimated the parameter values to be $\alpha = 17.71$ μm, $\beta = 10.24$ μm, and $\gamma = -0.01$. By having the same pattern repeated multiple times we got a better estimate and were able deal with some of the noise introduced by the printing process.

The corrected grayscale pattern and the corresponding printed samples are shown in the upper right part of Figure 9. The surface of the sample now looks more smooth and the resin solidifies everywhere on the surface. The blue plot in the lower right part of Figure 9 is a measurement of the surface height, while the red plot is the ideal linear behavior that we would like to have when printing with grayscale images. Even though the blue plot shows some irregularities, it proves that by applying Eq. 2 to our patterns we

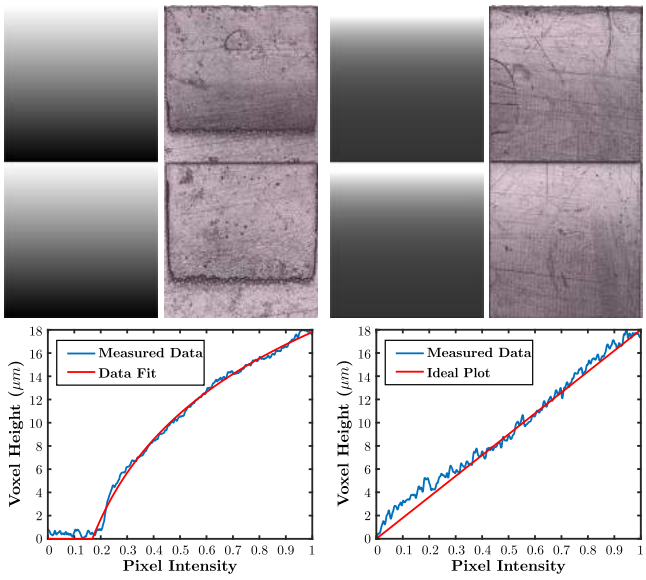


Figure 9: *Grayscale layer images and microscope images of printed results used for estimating α, β, and γ to control voxel growth (two repetitions). The linear gradient (left) is used for fitting to Eq. 1. The logarithmic gradient (right) is used for testing the linearity of the printed gradient after correction with Eq. 2.*

Table 1: *Average roughness measured as S_a and S_q for samples with sparse convolution noise applied using different amplitudes A and frequencies B.*

	$A = 0.625$	$A = 2$	$A = 3$	
S_a (μm)	2.21	3.30	5.53	$B = 16$
S_q (μm)	2.82	4.20	6.94	
S_a (μm)	2.90	4.49	7.95	$B = 32$
S_q (μm)	3.64	5.71	10.10	

obtain the desired geometry, and we therefore have the ability to control subvoxel-sized surface microstructure.

4.3. Roughness Measurements

To verify that we can print surfaces with different roughnesses by applying sparse convolution noise with varying amplitude and frequency parameters (Appendix A), we printed several samples and measured their surface roughness with the microscope. The parameters used in this experiment and the corresponding results are in Table 1. These results show quantitatively that by increasing the amplitude A and the frequency B of the noise function the area roughness parameters S_a (arithmetic average height) and S_q (root mean square roughness) increase as well. Thus, we obtain a smoother surface if we apply a grayscale pattern with sparse convolution noise using lower values of A and B, and more diffuse-like surfaces if we use higher values of these two parameters.

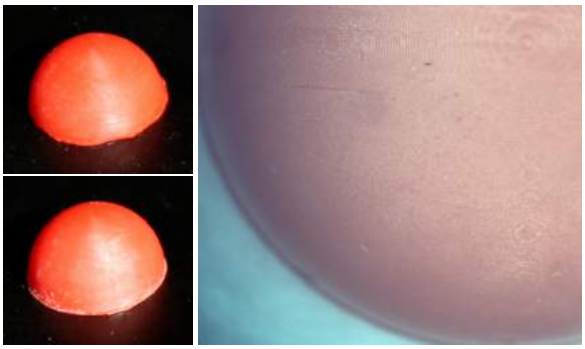


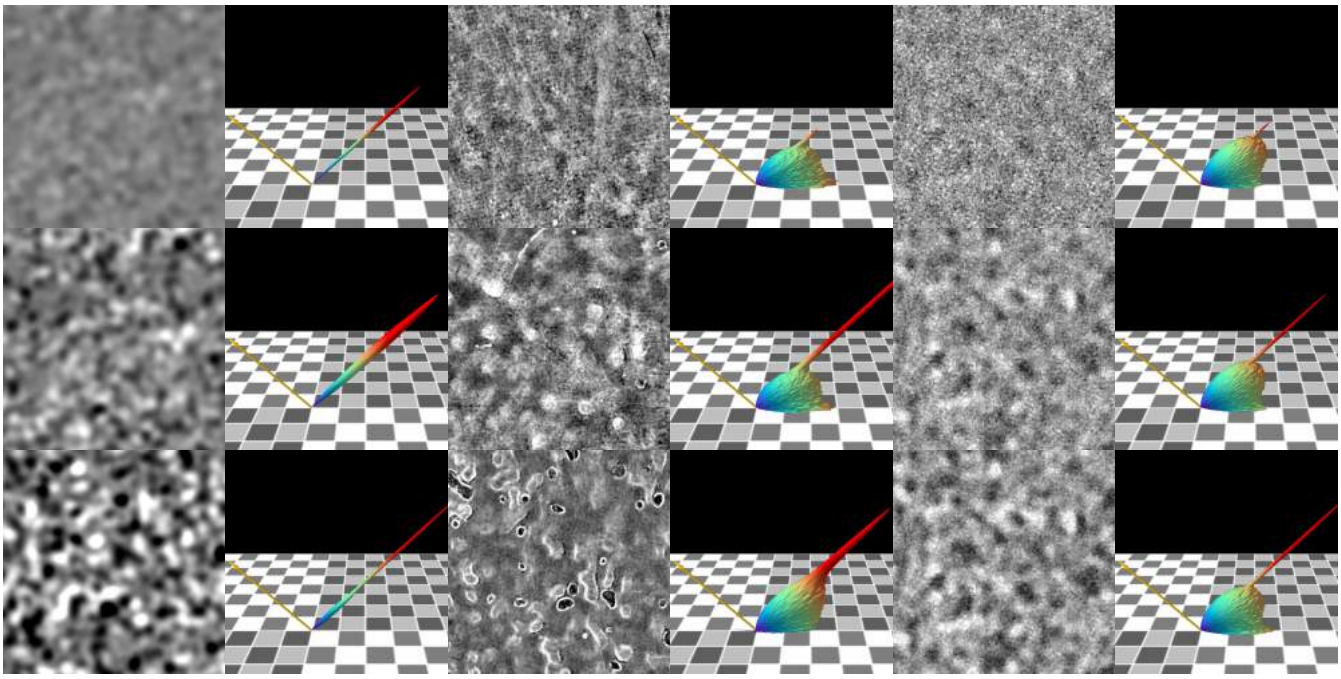
Figure 10: *Hemispheres printed with grayscale values calculated using supersampling. On the left, the hemispheres were printed using a 2× magnifying lens: one with supersampling only (top left) and one with both supersampling and sparse convolution noise (bottom left, parameters $A = 0.625$ and $B = 32$). On the right, the hemisphere was printed with supersampling and 1× magnifying lens. Even at a scale this small, moiré patterns are still visible when the surface is observed in a microscope.*

4.4. Antialiasing Abilities of Supersampling

As discussed by Greene [Gre16] and in Section 3.2.1, we can use supersampling to calculate grayscale values for spatial antialiasing during the slicing process. However, we find (as did Greene [Gre16]) that the surface will still exhibit reflectance anisotropy and moiré patterns. The hemisphere in Figure 10 (top left) was printed using 2× magnifying lens and supersampling for antialiasing. Nevertheless, it still has an elongated highlight that we would only expect to see when the surface exhibits anisotropic reflectance [AS00]. Even if printed with 1× magnifying lens and supersampling, we still see staircases and moiré patterns when looking through a microscope (Figure 10, right). On the other hand, we find a smooth irregular noise function (like the one presented in Appendix A) useful for obtaining improved antialiasing and more isotropic reflectance properties. The hemisphere in Figure 10 (bottom left) includes sparse convolution noise with parameters $A = 0.625$ and $B = 32$. While this sample is not completely free of aliasing artifacts, it does exhibits a more rounded highlight and, thus, more isotropic reflectance properties. The same hemisphere is illuminated by a more directional source in Figure 1.

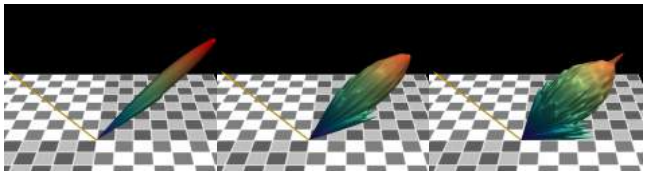
5. Results

Let us compare printed surface microstructure with the surface microstructure given as input grayscale values for the printing process. The first column of Figure 11 is examples of input noise at amplitudes $A = 0.625, 2, 3$ and the third column is examples of printed surface microstructure for input noise at the same amplitudes. It is clear that the printing process introduces additional noise, let us call it ground noise, caused by the membrane and the cleaning process. We can now use path tracing of a specular surface patch with geometry given by these height maps to calculate a corresponding BRDF lobe (second column of Figure 11). The input noises produce a highly specular lobe, so we also draw these using a logarithmic scale in Figure 12 to make their differences more easily



(a) input (b) input BRDF (c) measured (d) printed BRDF (e) input + ground (f) model BRDF

Figure 11: *(a) Input grayscale noise values of amplitudes $A = 0.625, 2, 3$, (c) surface microstructure printed using input of the same amplitudes and measured using a microscope, (e) ground noise added to the input noise. (b, d, f) Lobe images showing the BRDF values for a 45 degrees angle of incidence. The lobes were computed using path tracing.*



$A = 0.625$ $A = 2$ $A = 3$

Figure 12: *Log transformed versions of the BRDF lobes based on the input noise values alone (second column of Figure 11).*

observable. We observe that the shape of the lobe broadens with increasing amplitude. The height maps obtained by imaging printed surfaces using the Infinite Focus microscope result in a much more broadly scattering lobe that we visualize in the fourth column of Figure 11. The reflectance properties of the input surfaces and the printed surfaces are so different that they are hard to compare. However, the results are important as we can use them to build a model of the printer's added ground noise.

Through inspection of the measured height maps and using the noise function in Appendix A, we manually found that the following function is a good model for our printer's ground noise:

$$\begin{aligned}&\text{ground}(\boldsymbol{x})\\ &= \frac{2}{3}\text{noise}\left(\frac{\boldsymbol{x}}{50\ \mu\text{m}}\right) + \frac{1}{9}\text{noise}\left(\frac{\boldsymbol{x}}{25\ \mu\text{m}}\right) + \frac{1}{12}\text{noise}\left(\frac{\boldsymbol{x}}{2\ \mu\text{m}}\right).\end{aligned}$$

We believe this is useful as an example if one were to build a similar model for the ground noise of another printer. Finding an expression for the ground noise of a printer is important as it models the imprecision of the printing process. Since the printer adds noise similar to the ground noise to the input grayscale values, the ground noise function provides us with an outline of the printer's limitations in terms of reflectance control. If the printer is improved, we can repeat the experiment and see if the ground noise has diminished. To model the BRDF output of the printer, we add the ground noise to the input grayscale values and flatten the result a bit by clamping to include the membrane in the model. The fifth column of Figure 11 is examples of the surface microstructure estimated by this model, and the sixth column indicates that the resulting BRDF lobes come fairly close to the printed BRDF in the fourth column.

Figure 1 displays some of the visual effects enabled by our technique. It is remarkable that the rather small difference in the BRDFs that we estimated (Figure 11) produces a fairly obvious visual difference. In the following, we explore different techniques for printing surfaces with anisotropic reflection, and we demonstrate why the irregular noise function is important when printing 3D surfaces. Regarding the quality of antialiasing and the rate at which slices are generated, both techniques introduced in Section 3.4 perform similarly, and either one can be used to obtain the following results.

Figure 13 (top row) shows the grayscale patterns used for printing the smiley sample displayed in Figure 1. The figure also shows microscope images of the printed result (bottom row). We printed this sample with the $1\times$ lens to test how well we can print surfaces with anisotropic reflectance properties. In this example, we used the grayscale pattern for the last layer of the printing process only. We used the 2D sinusoid to generate the patterns in the main diagonal of Figure 13, with parameters $\lambda_u = 150\ \mu$m and $\lambda_v = 50\ \mu$m

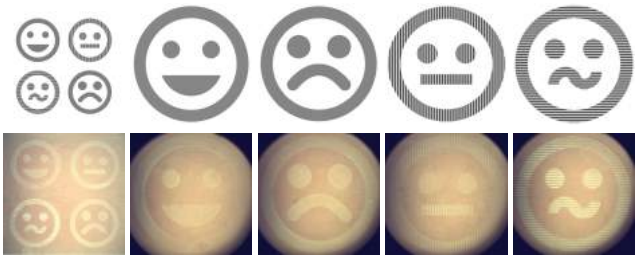


Figure 13: *Sample generated using two different anisotropic patterns with orthogonal orientation. The first two smileys have been printed with anisotropic sinusoidal patterns but with two different orientations. The last two have been printed with a ridged pattern with two different orientations.*

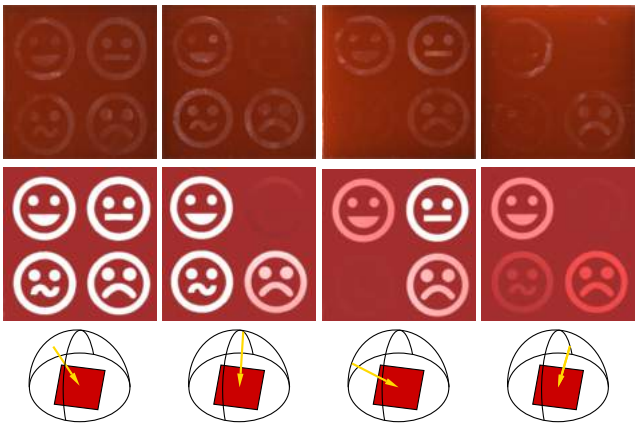


Figure 14: *Photos of the anisotropic smiley samples of Figure 13 (top row) with light incident from the directions shown in the bottom row. Reflection contrast predictions based on our analytic BRDF models are in the middle row. While the contrast seen in the printed samples is not as clear as in the predictions, the intensity variations are qualitatively similar.*

respectively $\lambda_u = 50\ \mu$m and $\lambda_v = 150\ \mu$m. In the antidiagonal, we used ridged patterns [LFD*17] with an inclination of 10° and pitch length of 100 μm. The ridges of the patterns in these two smileys have orthogonal orientations. The QR code in Figure 1 is another example of a surface with orthogonal ridged structures, but this was printed using the $2\times$ magnifying lens.

Printing these anisotropic patterns, we obtain a sample with spatially varying reflectance properties without adding any extra step to the DLP printing process. Figure 14 exemplifies how the different parts of the sample reflect light differently under different lighting conditions. The ridged structure generates different contrasts as the light rotates around the sample. The sinusoid structure also results in anisotropic properties, but the difference in contrast between the two different pattern orientations is not as strong as for the ridged pattern. On the other hand, the 2D sinusoid structure results in a more diffuse-like anisotropic effect. We validated these results by comparing the photographs in the top row of Figure 14 with images rendered using the corresponding BRDF models (as explained in Section 3). The printed samples present light reflection contrast that is qualitatively similar to the rendered images. The difference in reflection contrast between the printed samples and the rendered images are mainly due to our choice of using BRDF models (no subsurface scattering), and due to the ground noise introduced by the printing process.

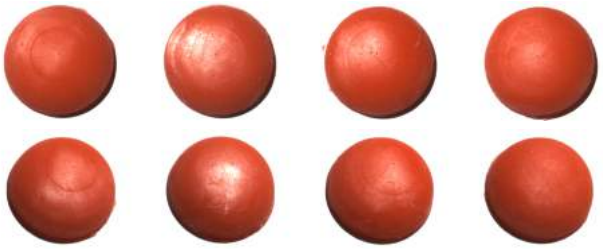


Figure 15: *Hemisphere printed without applying a grayscale pattern to the surface, leftmost, and from second to rightmost when using noise with amplitude $A = 0.625, 2, 3$, respectively, and frequency $B = 32$. Mesh slicing was done with ray tracing. The light-view configuration is the same within each row.*

In Figure 15, we compare a hemisphere printed without applying any grayscale pattern to the surface (leftmost) with samples where we applied sparse convolution noise of different amplitudes (A). The presence of a grayscale pattern produced by a smooth irregular noise function with low amplitude makes the surface smoother and removes the majority of the staircase aliasing artifacts introduced by the layered printing process. As the amplitude increases, the specular highlight becomes less visible and the surface appears to be more diffuse. This is a visual indication that the noise function enables us to control roughness not only in flat samples (as measured in Section 4.3) but also in curved 3D printed surfaces.

Finally, we applied grayscale patterns to a more complex geometry, namely the Stanford Bunny. The results are in Figure 1 and in Figure 16. In the leftmost column of Figure 16, the bunny was printed without applying a pattern to the surface. It exhibits an anisotropic specular highlight which is caused by the staircase that is a by-product of the layered printing. In the middle left column, we tried to remove the anisotropy and smoothen the printed surface by applying a low-frequency 2D sinusoid. While this approach to some extent reduces staircase artifacts in the highlights, a line-like reflection is still visible across the back of the bunny (bottom image). In addition, the regularity of the sinusoid pattern makes it visible on the back and the ears of the bunny (top image). A better result was achieved by using sparse convolution noise (middle right column and rightmost column). With a value of $A = 0.625$, we obtained a smoother surface with highlights similar to the ones obtained with the sinusoid pattern specular highlight but without introducing visible sinusoidal features. With $A = 3$, the bunny is more rough and the appearance is more diffuse-like. In Figure 1, we used the sun as the light source. This somewhat resembles a directional light and makes the difference between the rough and the smooth bunny stand out clearly.

We observed that the effect of our technique is less visible at the bottom of the ears of the Stanford Bunny. This is the case for surface voxels that are backfacing as seen from the projector. These

standard sinusoid noise ($A = 0.625$) noise ($A = 3$)

Figure 16: *Stanford Bunny printed and photographed in two different light-view configurations (rows). The bunny was printed without any grayscale pattern applied (*standard*), with an isotropic 2D sinusoid function applied (*sinusoid*, $\lambda_u = \lambda_v = 400$ μm*), and using sparse convolution noise with low and high amplitudes (A) and frequency* $B = 64$ *(*noise*). The glean due to anisotropic reflection caused by layering artifacts is clearly observable for the standard technique. The sinusoid pattern reduces the problem but introduces regularity artifacts. The noise function more effectively reduces the problem. As compared with the rough bunny (*$A = 3$*), the smooth bunny (*$A = 0.625$*) is brighter in the highlight regions and darker outside those regions as expected. Mesh slicing was done with rasterization.*

may have a different ground noise due to being cured without adhesion (interlaminar bonding) to an existing solidified layer. A technique such as monitoring the photopolymerization process using a photorheometer [HOBS18] might be used to improve the precision of a 3D printer for backfacing surface voxels.

6. Conclusion

In this work, we presented a one-step technique for controlling surface appearance in DLP printing. Our technique is based on projection of grayscale images to control the voxel growth and enable printing of subvoxel sized microstructure. We provided a procedure for correcting the nonlinearity of the photopolymerization process, and the validity of this procedure was experimentally verified. We also demonstrated that application of different grayscale patterns to surface voxels is useful for modifying the microstructure of a surface and for printing spatially varying anisotropic reflectance properties. An important discovery in our work is that a smooth irregular noise function (sparse convolution noise, in our case) is useful both for antialiasing to obtaining a smooth surfaces without staircase artifacts and for controlling surface roughness. We have described a pipeline for applying grayscale patterns to surface voxels during the slicing of mesh geometry. Finally, we included a procedure for calibrating the parameters of a DLP printer and for estimating the ground noise added to the surface by the printing process. Our results demonstrate that by modulating the UV light intensity of a DLP projector with grayscale images we can print samples with spatially varying reflectance properties, such as anisotropic effects and surface roughness.

As an addendum, Mark Wheadon has presented a webpage that describes an interesting experimental technique called velocity painting (www.velocitypainting.xyz). This technique enables use of grayscale values in fused deposition modeling (FDM) printing. The grayscale input images modify and control the print speed of an FDM 3D printer. This enables printing of patterns on the sample surface without modifying the filament or using multiple extruders. We leave investigation of the microstructure controllabilities of such a technique to future work. Nevertheless, we find it exciting that our calibration and grayscale microstructure control techniques can perhaps be transferred to the more commonly available nozzle-based 3D printers.

Acknowledgments. This work was funded by Innovation Fund Denmark (MADE Digital, 6151-00006B; QRprod, 5163-00001B; 3DIMS). The Stanford Bunny appearing in Figures 1, 6, and 16 is based on data from the Stanford Computer Graphics Laboratory, http://graphics.stanford.edu/data/3Dscanrep/.

Appendix A: Sparse Convolution Noise

We use sparse convolution noise [Lew84, Lew89] in the version presented by Frisvad and Wyvill [FW07], but implemented as a closed function. This is a solid noise function in the classical sense [Per85], but without the grid-aligned regularity artifacts seen

in Perlin noise and with no need for tabulated data. The noise function uses a simple linear congruential pseudo-random number generator:

$$\begin{aligned} t_{n+1} &= (b\,t_n + c) \bmod m \\ \mathrm{rnd}(t_n) &= t_{n+1}/m\,, \end{aligned}$$

where we use $b = 3125$, $c = 49$, and $m = 65536$, and a cubic filter kernel function

$$\mathrm{cubic}(\mathbf{v}) = \begin{cases} (1 - 4\,\mathbf{v}\cdot\mathbf{v})^3 & \text{for } \mathbf{v}\cdot\mathbf{v} < \frac{1}{4}\,, \\ 0 & \text{otherwise}\,. \end{cases}$$

A sparse distribution of randomly placed random impulses are then blended using this cubic filter to obtain the noise function. As the filter radius is $\frac{1}{2}$, we can use a regular grid offset by half a unit, so that we only need to consider the impulses in the eight nearest grid cells. Suppose i is the neighbor index of the grid cell, j is the impulse index, and N is the number of impulses per cell. We let $\alpha_{i,j}$ denote the value of the impulse, $\boldsymbol{\xi}_{i,j}$ the local position of the impulse in its grid cell, and $n_{i,j}$ the seed of the pseudo-random number generator for an impulse. The noise function is then

$$\mathrm{noise}(\mathbf{p}) = \frac{4}{5\sqrt[3]{N}} \sum_{i=0}^{7} \sum_{j=1}^{N} \alpha_{i,j}\,\mathrm{cubic}(\mathbf{x}_{i,j} - \mathbf{p})\,,$$

$$\begin{aligned} \mathbf{x}_{i,j} &= \mathbf{q}_i + \boldsymbol{\xi}_{i,j} \\ \alpha_{i,j} &= \mathrm{rnd}(t_{n_{i,j}})(1 - 2\,(j \bmod 2)) \\ \boldsymbol{\xi}_{i,j} &= \left(\mathrm{rnd}(t_{n_{i,j}+1}), \mathrm{rnd}(t_{n_{i,j}+2}), \mathrm{rnd}(t_{n_{i,j}+3})\right) \\ n_{i,j} &= 4(N\,\mathbf{q}_i \cdot \mathbf{a} + j) \\ \mathbf{q}_i &= \left\lfloor \mathbf{p} - \left(\frac{1}{2}, \frac{1}{2}, \frac{1}{2}\right) \right\rfloor + \left(i \bmod 2, \left\lfloor \frac{i}{2} \right\rfloor \bmod 2, \left\lfloor \frac{i}{4} \right\rfloor \bmod 2\right), \end{aligned}$$

where N should be an even number to avoid a bias toward negative impulse values. We use $N = 30$ and $\mathbf{a} = (1, 1000, 576)$.

To generate a noise function for procedural texturing with values in $[0, 1]$, we use

$$\min\left(\max\left(0, \frac{A}{2}\,\mathrm{noise}(B\,\mathbf{p}) + \frac{1}{2}\right), 1\right),$$

where the parameters A and B control the amplitude and the frequency (also called the scale) of the noise, respectively.

Appendix B: Masking and Shadowing for a Sinusoidal Structure

This appendix briefly describes the BRDF model that we used to predict the reflection contrast produced by a 2D sinusoidal microstructure. The microstructure is described by Equation 3. The model that we used is similar to the one presented by Luongo et al. [LFD*17] for the ridged surface microstructure, with the main difference that we here use a different geometrical attenuation function, G. As Walter et al. [WMLT07], we use the separation

$$G(\omega_i, \omega_o, \mathbf{n}) = G_1(\omega_i, \mathbf{n}) G_1(\omega_o, \mathbf{n})\,,$$

where ω_i and ω_o are incoming and outgoing light directions and $\mathbf{n}$ is the surface normal.

We consider a generic 2D sinusoidal function

$$f(x, y) = A \sin\left(\frac{2\pi}{\lambda_x} x\right) \sin\left(\frac{2\pi}{\lambda_y} y\right),$$

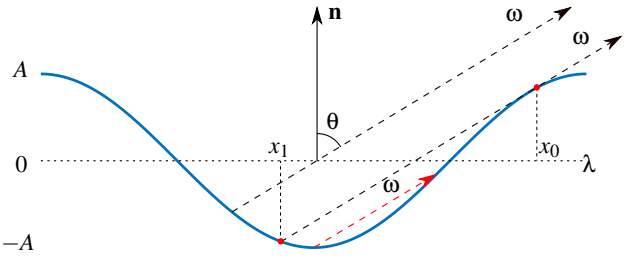


Figure 17: *The surface fraction masked by the sinusoidal structure is given by the ratio between $|x_1 - x_0|$ and λ.*

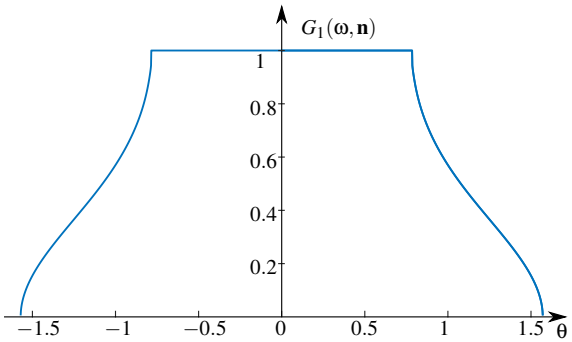


Figure 18: *Plot of the masking function G_1 for $A = 1$ and $\lambda = 2\pi$.*

where A represents the amplitude, and λ_x and λ_y are the wavelengths along the x and y axes. For simplicity, we derive the geometrical attenuation function for the 1D function

$$f(x) = A\cos(kx)$$

with $k = \frac{2\pi}{\lambda}$, and we then extend it to the 2D case.

For a given direction ω forming an angle θ with the surface normal $\mathbf{n}$, as shown in Figure 17, we would like to determine if this direction is tangent to $f(x)$. This is determined by solving

$$f'(x) = -Ak\sin(kx) = m \tag{4}$$

with $m = \tan\left(\frac{\pi}{2} - \theta\right)$. Equation 4 admits

$$x_0 = \arcsin\left(-\frac{m}{Ak}\right)\frac{1}{k}$$

as solution only if $\left|\frac{m}{Ak}\right| < 1$. We can now define the function G_1 by

$$G_1(\omega, \mathbf{n}) = \begin{cases} 1 - \frac{|x_1 - x_0|}{\lambda}\,, & \left|\frac{m}{Ak}\right| < 1\,, \\ 1\,, & \left|\frac{m}{Ak}\right| > 1\,, \end{cases} \tag{5}$$

where x_1 is the intersection point between $f(x)$ and the tangent line

$$f_t(x) = f(x_0) + m(x - x_0)\,,$$

as shown in Figure 17, and this is found by numerically solving the equation $f_t(x) - f(x) = 0$.

Equation 5 is plotted in Figure 18 for the parameters $A = 1$ and $\lambda = 2\pi$. The function G_1 is extended to the 2D sinusoidal case by considering projections of ω on the planes spanned by $\mathbf{n}$ and the

x-axis as well as $\mathbf{n}$ and the y-axis. We refer to these projections as ω_x and ω_y and define G_1 by

$$G_1(\omega, \mathbf{n}) = G_1(\omega_\mathbf{x}, \mathbf{n}) G_1(\omega_\mathbf{y}, \mathbf{n}).$$

References

[ABRK17] AURICH J. C., BOHLEY M., REICHENBACH I. G., KIRSCH B.: Surface quality in micro milling: Influences of spindle and cutting parameters. *CIRP Annals 66*, 1 (2017), 101–104. 2

[AS00] ASHIKHMIN M., SHIRLEY P.: An anisotropic Phong BRDF model. *Journal of graphics tools 5*, 2 (2000), 25–32. 7

[BBS15] BAAR T., BRETTEL H., SEGOVIA M. V. O.: Towards gloss control in fine art reproduction. In *Measuring, Modeling, and Reproducing Material Appearance* (2015), vol. 9398 of *Proceedings of SPIE Electronic Engineering 2015*, p. 93980T. 2

[Ben17] BENNETT J.: Measuring UV curing parameters of commercial photopolymers used in additive manufacturing. *Additive Manufacturing 18* (December 2017), 203–212. 3

[BSH97] BATTKE H., STALLING D., HEGE H.-C.: Fast line integral convolution for arbitrary surfaces in 3D. In *Visualization and Mathematics*. Springer, 1997, pp. 181–195. 4

[DWP*10] DONG Y., WANG J., PELLACINI F., TONG X., GUO B.: Fabricating spatially-varying subsurface scattering. *ACM Transactions on Graphics (SIGGRAPH 2010) 29*, 4 (July 2010), 62:1–62:10. 3

[ESZ*17] ELEK O., SUMIN D., ZHANG R., WEYRICH T., MYSZKOWSKI K., BICKEL B., WILKIE A., KŘIVÁNEK J.: Scattering-aware texture reproduction for 3d printing. *ACM Transacions on Graphics (SIGGRAPH Asia 2017) 36*, 6 (November 2017), 241:1–241:15. 3

[Fri12] FRISVAD J. R.: Building an orthonormal basis from a 3d unit vector without normalization. *Journal of Graphics Tools 16*, 3 (August 2012), 151–159. 4

[FW07] FRISVAD J. R., WYVILL G.: Fast high-quality noise. In *Proceedings of GRAPHITE 2007* (December 2007), ACM, pp. 243–248. 10

[Gre16] GREENE R.: High-fidelity 3D printing techniques. Additive Manufacturing Today, Videos, April 2016. 3, 4, 7

[HFM*10] HAŠAN M., FUCHS M., MATUSIK W., PFISTER H., RUSINKIEWICZ S.: Physical reproduction of materials with specified subsurface scattering. *ACM Transactions on Graphics (SIGGRAPH 2010) 29*, 4 (July 2010), 61:1–61:10. 3

[HOBS18] HOFSTETTER C., ORMAN S., BAUDIS S., STAMPFL J.: Combining cure depth and cure degree, a new way to fully characterize novel photopolymers. *Additive Manufacturing 24* (December 2018), 166–172. 10

[Jac92] JACOBS P. F.: *Rapid prototyping & manufacturing: fundamentals of stereolithography*. Society of Manufacturing Engineers, 1992. 3

[Jør15] JØRGENSEN A. R.: *Design and development of an improved direct light processing (DLP) platform for presion and additive manufacturing*. Master's thesis, Technical University of Denmark, 2015. 5

[LCR*17] LOW Z.-X., CHUA Y. T., RAY B. M., MATTIA D., METCALFE I. S., PATTERSON D. A.: Perspective on 3D printing of separation membranes and comparison to related unconventional fabrication techniques. *Journal of Membrane Science 523* (February 2017), 596–613. 3

[LDPT13] LAN Y., DONG Y., PELLACINI F., TONG X.: Bi-scale appearance fabrication. *ACM Transactions on Graphics (SIGGRAPH 2013) 32*, 4 (July 2013), 145:1–145:11. 2

[Lew84] LEWIS J. P.: Texture synthesis for digital painting. *Computer Graphics (SIGGRAPH '84) 18*, 3 (July 1984), 245–252. 10

[Lew89] LEWIS J. P.: Algorithms for solid noise synthesis. *Computer Graphics (SIGGRAPH '89) 23*, 3 (July 1989), 263–270. 10

[LFD*17] LUONGO A., FALSTER V., DOEST M. B., LI D., REGI F., ZHANG Y., TOSELLO G., NIELSEN J. B., AANÆS H., FRISVAD J. R.: Modeling the anisotropic reflectance of a surface with microstructure engineered to obtain visible contrast after rotation. In *Proceedings of International Conference on Computer Vision Workshop (ICCVW 2017)* (October 2017), IEEE, pp. 159–165. 4, 9, 11

[LGX*13] LEVIN A., GLASNER D., XIONG Y., DURAND F., FREEMAN W., MATUSIK W., ZICKLER T.: Fabricating BRDFs at high spatial resolution using wave optics. *ACM Transactions on Graphics (SIGGRAPH 2013) 32*, 4 (July 2013), 144:1–144:13. 2

[LPA01] LEE J. H., PRUD'HOMME R. K., AKSAY I. A.: Cure depth in photopolymerization: experiments and theory. *Journal of Materials Research 16*, 12 (December 2001), 3536–3544. 3

[MAG*09] MATUSIK W., AJDIN B., GU J., LAWRENCE J., LENSCH H., PELLACINI F., RUSINKIEWICZ S.: Printing spatially-varying reflectance. *ACM Transactions on Graphics (SIGGRAPH Asia 2009) 28*, 5 (December 2009), 128:1–128:9. 2

[MQM17] MOSTAFA K., QURESHI A. J., MONTEMAGNO C.: Tolerance control using subvoxel gray-scale DLP 3D printing. In *Proceedings of ASME International Mechanical Engineering Congress and Exposition (IMECE17)* (2017), p. V002T02A035. 2, 3

[MSRG12] MCEWAN I., SHEETS D., RICHARDSON M., GUSTAVSON S.: Efficient computational noise in GLSL. *Journal of Graphics Tools 16*, 2 (2012), 85–94. 4

[MSS*12] MALZBENDER T., SAMADANI R., SCHER S., CRUME A., DUNN D., DAVIS J.: Printing reflectance functions. *ACM Transactions on Graphics 31*, 3 (May 2012), 20:1–20:11. 2

[Per85] PERLIN K.: An image synthesizer. *Computer Graphics (SIGGRAPH '85) 19*, 3 (July 1985), 287–296. 4, 10

[Per02] PERLIN K.: Improving noise. *ACM Transactions on Graphics (SIGGRAPH 2002) 21*, 3 (July 2002), 681–682. 4

[PLMR17] PEREIRA T., LEME C. L., MARSCHNER S., RUSINKIEWICZ S.: Printing anisotropic appearance with magnetic flakes. *ACM Transactions on Graphics (SIGGRAPH 2017) 36*, 4 (July 2017), 123:1–123:10. 2

[PRJ*13] PAPAS M., REGG C., JAROSZ W., BICKEL B., JACKSON P., MATUSIK W., MARSCHNER S., GROSS M.: Fabricating translucent materials using continuous pigment mixtures. *ACM Transactions on Graphics (SIGGRAPH 2013) 32*, 4 (July 2013), 146:1–146:12. 3

[PZNH16] PEDERSEN D. B., ZHANG Y., NIELSEN J. S., HANSEN H. N.: A self-peeling vat for improved release capabilities during DLP materials processing. In *Proceedings of the 2nd International Conference on Progress in Additive Manufacturing (Pro-AM 2016)* (2016), pp. 241–245. 5

[RBK*13] ROUILLER O., BICKEL B., KAUTZ J., MATUSIK W., ALEXA M.: 3D-printing spatially varying BRDFs. *Computer Graphics and Applications 33*, 6 (2013), 48–57. 2

[RBSM19] RIBARDIÈRE M., BRINGIER B., SIMONOT L., MENEVEAUX D.: Microfacet BSDFs generated from NDFs and explicit microgeometry. *ACM Transactions on Graphics 38*, 5 (2019), 143:1–143:15. 5

[Rib17] RIBO M. M.: *3D Printing of Bio-inspired Surfaces*. Master's thesis, Technical University of Denmark, 2017. 6

[TR75] TROWBRIDGE T. S., REITZ K. P.: Average irregularity representation of a rough surface for ray reflection. *Journal of the Optical Society of America 65*, 5 (1975), 531–536. 4

[WMLT07] WALTER B., MARSCHNER S., LI H., TORRANCE K.: Microfacet models for refraction through rough surfaces. In *Proceedings of Eurographics Symposium on Rendering (EGSR 2007)* (2007), The Eurographics Association, pp. 195–206. 5, 11

[WPMR09] WEYRICH T., PEERS P., MATUSIK W., RUSINKIEWICZ S.: Fabricating microgeometry for custom surface reflectance. *ACM Transactions on Graphics (SIGGRAPH 2009) 28*, 3 (August 2009), 32:1–32:6. 2

CONTRIBUTION E

Alignment of rendered images with photographs for testing appearance models

Alignment of rendered images with photographs for testing appearance models

MORTEN HANNEMOSE[1], MADS EMIL BRIX DOEST[1], ANDREA LUONGO[1], SØREN KIMMER SCHOU GREGERSEN[1], JAKOB WILM[2], AND JEPPE REVALL FRISVAD[1,*]

[1] Technical University of Denmark, Richard Petersens Plads, Building 321, 2800 Kongens Lyngby, Denmark
[2] University of Southern Denmark, Campusvej 55, 5230 Odense M, Denmark
[*] Corresponding author: jerf@dtu.dk

We propose a method for direct comparison of rendered images with a corresponding photograph in order to analyze the optical properties of physical objects and test the appropriateness of appearance models. To this end, we provide a practical method for aligning a known object and a point-like light source with the configuration observed in a photograph. Our method is based on projective transformation of object edges and silhouette matching in the image plane. To improve the similarity between rendered and photographed objects, we introduce models for spatially varying roughness and a model where the distribution of light transmitted by a rough surface influences direction-dependent subsurface scattering. Our goal is to support development toward progressive refinement of appearance models through quantitative validation.

http://dx.doi.org/10.1364/ao.XX.XXXXXX

1. INTRODUCTION

Photorealistic rendering has many applications: product appearance prediction, digital prototyping, inverse rendering to acquire optical properties, 3D soft proofing, etc. In most of these applications, it is important to validate the photorealism of the employed rendering technique. In graphics, side-by-side visual comparison of rendered and photographed images has traditionally been the validation method of choice. Phong [1], for example, qualitatively compared a rendered sphere with a photographed sphere as a final evaluation of his shading and lighting models. Similarly, the Cornell box [2, 3] was presented as a test scene for qualitative comparison of photographs and rendered images. Rushmeier [4] was seemingly the first to discuss quantitative comparison of photographed and rendered images, and Pattanaik et al. [5] then presented a difference image for rendering versus photograph of a version of the Cornell box. Differences in scene geometry and the view-light configuration tend to be the main difficulty in setting up such pixel-by-pixel comparisons [4, 6].

Alignment of rendered and photographed images has reached good precision in controlled setups for geometry and reflectance acquisition [7]. For images captured in less controlled settings, the main difficulties are pose estimation of an object from a given CAD model and light source estimation. These are most often considered two separate problems. For pose estimation, a large dataset is usually employed to train a statistical model [8, 9]. A multitude of techniques exist for light source estimation [10, 11]. However, as we estimate the object pose, we may as well use the pose for light source estimation. Moreover, if we use the cast shadow for estimating the light position, we can use it to improve the estimate of the object pose as well.

Inverse rendering [12] enables recovery of both lighting and reflectance properties but often assumes a known object with a known pose. More recent inverse rendering techniques [13–15] allow pose estimation and deformation of object geometry too. These techniques are based on differentiable rendering, where per pixel derivatives are computed as part of the rendering. While this is a powerful approach for estimating surface displacements and spatially varying reflectance [13], it is also a gradient-based optimization based on per pixel derivatives that requires careful initialization to avoid local minima [14]. In this landscape, we missed a practical method for estimation of both object pose and light source position to enable pixel-by-pixel comparison of a photograph with a rendering. We propose such a method and find that it delivers a good starting point for validating rendering techniques, estimating optical properties, and testing appearance models. In addition, our method is useful for initialization of inverse rendering techniques.

Our outset is a photograph of a single object of known geometry that has been captured with a known camera. We assume that the object is placed on a diffuse planar surface and illuminated by a point-like light source. We let the term *point-like* refer to a small source with a uniform far-field radiant intensity distribution within the part of the scene observed by the camera. In this scene configuration, we let the user approximately initialize the orientation of the object relative to the planar surface

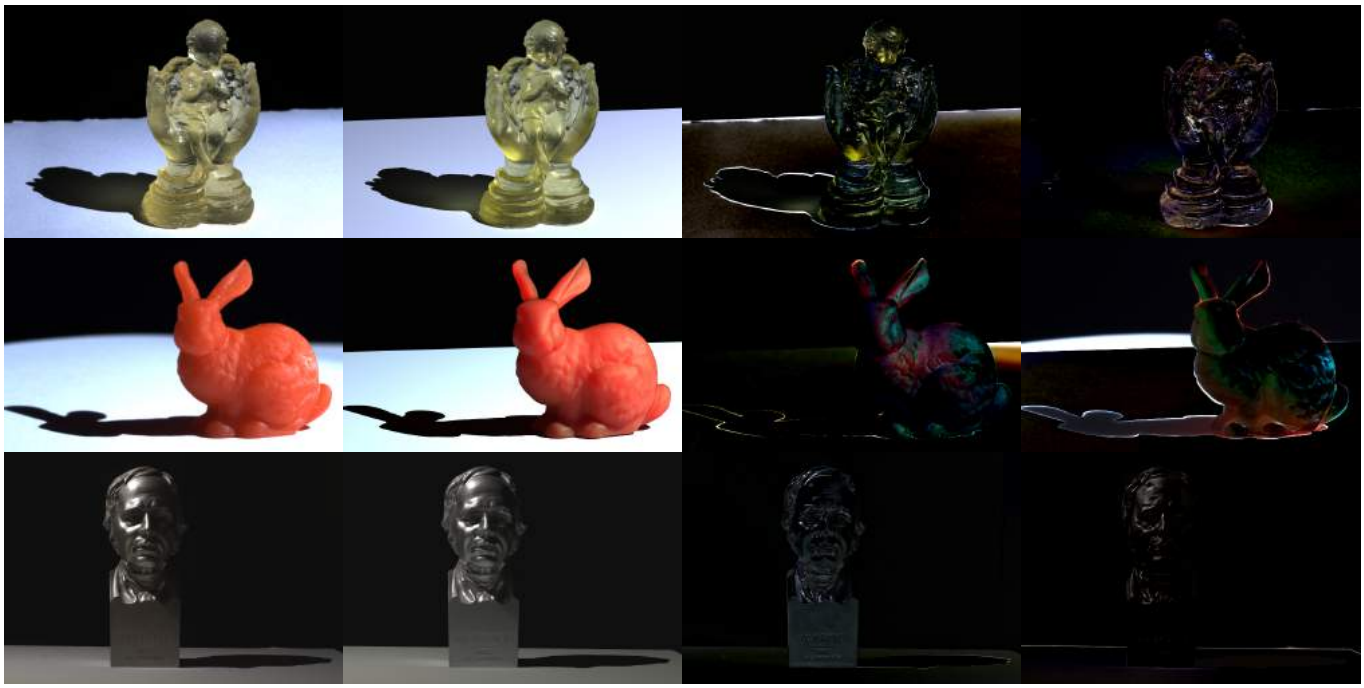


Fig. 1. Pixel-by-pixel comparison of renderings with a photograph enables a detailed investigation of the virtues and deficiencies of an appearance model. Our practical alignment technique is here used for testing different models: rough transparent (top), rough translucent (middle), and metallic (bottom). The signed difference images to the right have been scaled by a factor of 2.

(this could be done using a physics engine), or we use a camera calibration. Our method then estimates the light source position and the camera and object poses. We do this by segmenting the photograph and matching the object and the shadow silhouettes to the silhouettes of the virtual object found by projective transformation of the edges.

We exemplify our method using three scanned objects (see Figure 1): the Stanford bunny [16], a cupped angel figurine, and an aluminium bust of H. C. Ørsted (the scientist who discovered electromagnetism and who was also the first to isolate aluminium). The Stanford bunny was scanned by Greg Turk using a technique for zippering several range scans [17], and we 3D scanned the other two objects using structured light and stereo vision [18]. We use a translucent 3D printed version of the Stanford bunny, the angel figurine was 3D printed using an almost transparent photopolymer, while we used the aluminium bust as is. This enabled us to take photographs and test appearance models for both subsurface scattering, rough refraction, and metallic rough reflection. We quantitatively test the ability of such models to match the appearance of object samples from the real world (Figure 1), and we suggest improved models based on our findings. Notably, we for the first time integrate rough surface scattering [19] with the directional dipole model for subsurface scattering [20].

2. RELATED WORK

In many side-by-side comparisons of renderings with photographs [1–3, 6, 12, 21], alignment is done manually. This is usually a time-consuming process with an imprecise result. When a comparison is done in the context of 3D acquisition, alignment is given with good accuracy because the object geometry was acquired in a calibrated setup [22, 23]. We are however looking for an alignment method that does not require concurrent 3D scanning of the object. Differentiable rendering [13, 15, 24, 25] is an option, but the aim of such a technique is usually more than alignment. We think of our technique as an enabler for an inverse (differentiable) rendering system, which is then free to focus on estimation of parameters not related to alignment. In Sec. 6.B, we compare our object pose estimation with that of a differentiable rendering method [15] to demonstrate the advantages of our specialized technique.

Our work is related to CAD-based vision [26], where the CAD model of a 3D object is used to recognise the physical version of the object in an image. An important part of such recognition is pose estimation of the object. In a view-based approach [27, 28], multiple views of the object are used for the training of a statistical model to recognise the object and suggest an initial pose. The views can be obtained from photographs captured in a calibrated robot setup [27] or from rendered images of object edges [8, 28, 29]. After estimating an initial pose using a statistical model, the pose is typically refined using iterative shape matching [28, 30]. We combine some of these ideas. Petit et al. [29] suggest a method based on foreground/background segmentation in the case of a moving object. Our method is also based on such a segmentation but for a static object. As in the discussed previous work, we use the edges of the CAD model for pose estimation, specifically the silhouette [8], but we avoid the training of a statistical model based on a dataset with many views.

Iterative methods for pose estimation [30] are good for pose refinement but also prone to local minima if not carefully initialised. An exhaustive search for initial parameters is then needed if we want to avoid the training of a statistical model,

but such a search is infeasible for the full 6D pose of an object. An option is then to limit the dimensionality of the search space using invariants [31, 32]. Hu's moment invariants [33] are for example invariant to scale, rotation, and translation. For a 2D shape, this reduces the search space in pose estimation to two angular dimensions [31]. We use this concept for 3D shapes by applying it to the object silhouette found in the image plane.

If one is willing to generate a dataset of object silhouettes (for example) as observed across a view sphere, the pose estimation can be accomplished using shape descriptors even for cluttered scenes [34]. After image segmentation and initial pose estimation, refinement is still required using an iterative method. Several other learning-based techniques are available as well [35–38]. These all require a large dataset for training and pose refinement after estimating the initial pose. Interestingly, Tekin et al. [39] report a fast learning-based method that does not require pose refinement, but then Li et al. [40] present an iterative learning-based method for pose refinement with improvements over Tekin et al. Peng et al. [9] present an improved method inspired by Tekin and others that indeed seems not to require *a posteriori* pose refinement. This is based on an extensive dataset augmented with 20,000 synthetic images of each object. These learning-based techniques contribute robustness with respect to object detection. This is however not important for our scenes which must, in any case, be uncluttered to enable photorealistic rendering of a corresponding digital scene.

A distinctive advantage of our silhouette matching approach is that we can estimate the light source position too. In this way, we avoid the traditional calibration of a point light by observing highlights in mirroring spheres [7]. Our method employs the shadow silhouette, which we find using Blinn's projection shadows [41]. In some related work [42], the shadow silhouette was detected in an input image with depth information (RGB-D) and used for estimating the position of one or more light sources. However, since we estimate the pose of a known object together with the position of the light, we do not need the depth information. In addition, our treatment of pose and light as a joint problem enables us to refine the estimation of both.

3. ALIGNMENT METHOD

Our method is based on the following input:

- image of an object on a uniform ground plane illuminated by a point-like light source
- segmentation of the image into object, shadow, and background
- 3D model of the object
- camera intrinsics (focal length / camera constant / field of view)
- approximate rotation of the object relative to the ground plane.

Any camera can be used to capture the input image, but we need to know the field of view. If this is not known for a given camera, we can obtain it through camera calibration, but we exclude images captured with an unknown and unavailable camera. In most cases, the segmentation can be accomplished by appropriate thresholds of the input image. In harder cases, such as transparent objects, a good segmentation can be obtained through background subtraction based on one image with and one without the object.

Although we work with one light source per view, we also illuminate a static object with multiple light sources in different positions one at a time. In this case, we use the additional information to improve the object pose and light source positions in a final refinement step.

Algorithm 1. Computing a silhouette from edges of a mesh projected to a plane. Each edge exists once in each direction.

$\mathbf{p} := \mathbf{p}_0$ (the leftmost point)
$\mathbf{e}$:= edge from $\mathbf{p}$ with the largest slope
repeat
  from $\mathbf{p}$ follow $\mathbf{e}$ until next intersection, $\mathbf{p}_{\text{new}}$
  $\mathbf{e}_{\text{new}}$:= choose from edges intersecting $\mathbf{p}_{\text{new}}$
    such that angle$(\mathbf{e}_{\text{new}}, \mathbf{e})$ is minimized
  $\mathbf{p} := \mathbf{p}_{\text{new}}$, $\mathbf{e} := \mathbf{e}_{\text{new}}$
until $\mathbf{p} = \mathbf{p}_0$

To obtain object pose and light source position (in $\mathbb{R}^3$), we project the 3D model into the image plane of the camera and extract the silhouette. Our method aligns the silhouette in this plane with the corresponding silhouette in the input image. We obtain the latter from the segmentation of the input image. The silhouette is a useful representation that enables different comparisons of two silhouettes with options for being either exact or invariant to various measures such as rotation and translation, all while being differentiable.

We define a silhouette as a list of 2D point pairs each representing an edge with a direction. In analogy with a triangle mesh, we can use an indexed edge set to represent a silhouette or a set of lists of 2D points, where the points in each list are connected by edges. This works in general, as we can describe objects with holes (nonzero genus) by having both outer and inner perimeters. An inner perimeter should then be in the opposite direction.

A. Silhouette Computation

To compute the silhouette of the real object, we enlarge the segmentation resolution by a factor of two using nearest neighbor sampling. We then use the algorithm by Suzuki and Abe [43, 44] to trace the perimeter of the object. We downscale the traced perimeter and round the coordinates so that they lie exactly on the border between object and background. After tracing the perimeter, we have an optional step to simplify the perimeter to accelerate computations later on. The optional simplification is done using the Ramer-Douglas-Peucker algorithm [45, 46]. If the lens distortion of the camera that captured the ground truth image is known, the silhouette points can be undistorted, removing the need to undistort the segmentation itself.

We compute silhouettes of the 3D models without rasterization. This makes the silhouettes directly differentiable with respect to scene parameters, which is an advantage in a gradient-based optimization. Given a CAD model, we extract a polygonal mesh and build a half-edge representation of this for easy queries. For a given view matrix, we project the vertex positions to the image plane and connect them using the edges of the mesh polygons. To compute the silhouette, we traverse these edges using Algorithm 1. This algorithm assumes a fully connected object silhouette without holes. Extension to objects with holes is done by restarting the algorithm inside each hole.

For the silhouette computation in Algorithm 1, we find the signed angle between two vectors in 2D using

$$\text{angle}\left(\begin{bmatrix} a_1 \\ a_2 \end{bmatrix}, \begin{bmatrix} b_1 \\ b_2 \end{bmatrix}\right) = \text{atan2}\left(a_1 b_2 - a_2 b_1,\ a_1 b_1 + a_2 b_2\right). \quad (1)$$

The majority of time in Algorithm 1 is spent computing edge intersections [47]. Computational complexity thus depends on the number of edges. We significantly reduce this number by exploiting that an edge can only be part of the silhouette if it is shared by one face facing the camera and another facing away [48]. If we let $\vec{n}_{e,1}$ and $\vec{n}_{e,2}$ denote the 3D surface normals of the faces bordering an edge e, the edge e can only be part of the silhouette if

$$\left(\vec{n}_{e,1} \cdot (\mathbf{v}_e - \mathbf{c})\right)\left(\vec{n}_{e,2} \cdot (\mathbf{v}_e - \mathbf{c})\right) \leq 0\,, \tag{2}$$

where $\mathbf{v}_e$ is any point on the edge and $\mathbf{c}$ is the position of the camera. After removing all edges that cannot be part of the silhouette and building a bounding volume hierarchy for the remaining edges, intersection testing is inexpensive.

Another way to reduce the computation time of this algorithm is to use a mesh with a lower polygon count for the silhouette while retaining the original mesh for rendering. A modest mesh simplification often has a negligible influence on the silhouette.

We have several options when computing silhouette derivatives. For simplicity, we use finite differences. Exact derivatives can be obtained with automatic differentiation.

B. Shadow Contours

To include the shadow of an object when considering its silhouette, we assume that the object is placed on a planar surface and use projection shadows [41]. This is also done without rasterization to keep our method valid for the entire image plane. We project the edges of the mesh to the ground plane to generate shadow edges. We then project both object and shadow edges to the image plane of the camera. After this, we use Algorithm 1 to compute the silhouette of the object including its shadow. The number of edges in the shadow that we need to consider is reduced early in the procedure by substituting $\mathbf{c}$ with the light position in Eq. (2).

C. Silhouette Matching

To be able to align silhouettes, we introduce a silhouette similarity metric. We refer to the silhouette of the real object observed by camera c as $\mathbf{R}_c$, and the union of object and shadow silhouettes as $\mathbf{R}_{c,\ell}$, where ℓ is the light source causing the shadow. Equivalently, we define for the virtual object $\mathbf{V}_c$ and $\mathbf{V}_{c,\ell}$. We now let $\mathrm{P}(\mathbf{X}, t)$ denote a parameterization of the silhouette $\mathbf{X}$ with $t \in [0,1]$. We measure the similarity of two silhouettes by using a function (d) that finds the shortest distance from a point to a silhouette. Taking n equidistantly sampled points on the silhouettes, we find the shortest distance to the other silhouette and take the sum. The similarity is then computed by

$$\mathrm{sim}(\mathbf{R}, \mathbf{V}, n) = \sum_{i=1}^{n} \left(\mathrm{d}\left(\mathbf{R}, \mathrm{P}\left(\mathbf{V}, \tfrac{i}{n}\right)\right)^2 + \mathrm{d}\left(\mathbf{V}, \mathrm{P}\left(\mathbf{R}, \tfrac{i}{n}\right)\right)^2 \right), \tag{3}$$

A visualization of what sim computes is in Figure 2. We can again use a spatial data structure to obtain an efficient implementation of the dist function [49]. Our similarity metric (sim) has the advantage that it has a nonzero gradient even for non-intersecting silhouettes, which enables the use of our method with a poor initial guess.

Our final goal is to minimize the difference between the silhouettes of the real and virtual objects. For a silhouette without shadow, we measure the similarity by

$$E_c = \mathrm{sim}(\mathbf{R}_c, \mathbf{V}_c, \lceil \|\mathbf{R}_c\| \rceil)\,, \tag{4}$$

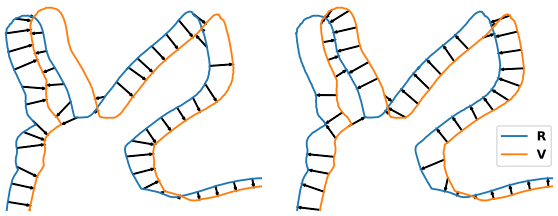


Fig. 2. Illustration of how $\mathrm{sim}(\mathbf{R}, \mathbf{V}, n)$ is computed for a small value of n. The arrows illustrate evaluations of $\mathrm{dist}(\cdot, \cdot)$.

where $\|\cdot\|$ denotes the length of a silhouette in pixels. Ideally, we would like to sample as many points as possible. In this performance vs. accuracy trade-off, we have chosen $n = \lceil \|\mathbf{R}_c\| \rceil$ to place the sampled points approximately one pixel apart.

To compare silhouettes including shadows, we introduce a similarity measurement $E_{c,\ell}$. As mentioned previously, we would like to refine estimates using multiple cameras and light sources as long as only one is active per image. We compute the sum of comparisons of silhouettes over one or more configurations as follows:

$$E_s = \sum_{\ell} \sum_{c} \Big(\underbrace{\mathrm{sim}(\mathbf{R}_{c,\ell}, \mathbf{V}_{c,\ell}, \lceil \|\mathbf{R}_{c,\ell}\| \rceil)}_{E_{c,\ell}} + E_c \Big). \tag{5}$$

In the following, we describe how we estimate object pose and light source position using these silhouette similarity measurements.

D. Pose Estimation

We compute the pose of the object independently for each camera. We do this in camera space, where the camera is fixed at the origin. In the end, we can then use the known relation between object and ground plane to position each camera in world space.

Starting in camera space, the first step of the pose estimation is to find an initial guess for the position of the object. We do this by minimizing E_c with respect to the position, which places the virtual object approximately in the same position as the real object.

To find a good initial guess of the rotation, we randomly sample rotations. For each rotation, we compare the silhouette of the digital object to the real object using Hu's moment invariants [33]. These are calculated from image moments but are invariant to scale, rotation, and translation. For an image of pixel values $I(x, y)$, the image moments are defined by

$$M_{pq} = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} x^p y^q I(x, y)\, dx\, dy, \tag{6}$$

where the p and q exponents are the moment orders and integration is across the image plane. Since the silhouette can be considered a polygon, the image moments can be computed efficiently by applying Green's theorem [50]. Hu's moment invariants are seven polynomial combinations of image moments that we store in a vector and compare using the sum of squared differences. Using the Hu moment invariants, the search space of the rotation is practically reduced to two dimensions. The rotation giving the silhouette that best matches the Hu moment invariants of the real silhouette is chosen as the initial guess of the rotation. We parameterize the rotation using quaternions and use the centroid of the object as the rotation centre.

With these initial guesses for position and rotation, we minimize E_c, which gives an object pose for each view. We use

Levenberg-Marquardt [51, 52] for the minimization. This is possible as E_c is a sum of squares. As part of our input, we know the object pose in relation to the ground plane. We use this to convert the per camera object poses into camera poses in world space.

E. Light Positions and Final Refinement

To estimate the position of each light, we randomly sample positions and then choose the one with the lowest $E_{c,\ell}$ for each light. Following this, $E_{c,\ell}$ is minimized using Levenberg-Marquardt.

The last step of our method is a joint optimization where we minimize E_s with respect to object pose, camera pose(s), light position(s), and a non-uniform scaling of the mesh. The non-uniform scaling of the mesh is to compensate for some of the shrinkage that may occur during 3D printing. The final optimization of the object pose is beneficial as the inclusion of the shadow silhouette(s) enables us to use more information from the input image.

F. Known Camera Poses

If camera poses are known in advance, for example from a stereo calibration of the camera rig, we can use the same steps as in Sec. 3.D to find the pose for all cameras jointly. When finding the rotation, it is then no longer desirable to have rotational invariance for all cameras. Instead, we propose to rotate the object to align the normalized image moments of the virtual and digital objects in the best way possible along a randomly chosen camera's viewing direction. The rotation is found by aligning the principal components of the two silhouettes [53]. We choose the rotation that best matches the normalized image moments across all cameras as the best rotation.

An initial guess of the object's scale is required, but if the camera poses are known in relation to the ground plane, we need not know the rotation of the object relative to the ground plane. The method for light source estimation is as with unknown camera poses.

4. APPEARANCE MODELS FOR REAL OBJECTS

Rendering systems provide a multitude of rendering techniques that we need to choose among when composing an appearance model for a real object. We start from a very approximate model at the most macroscopic scale. We then gradually increase complexity by reconsidering the involved optical properties [54] and what types of materials and visual effects that they can model.

At the most macroscopic scale, we have the bidirectional reflectance/transmittance distribution function (BRDF/BTDF) and the simplest models at this scale are the ones for perfectly diffuse and perfectly specular materials [55]. To cover a broad spectrum of different material types, we consider three different starting points: (a) diffuse, (b) metallic, or (c) transparent. In the following, we describe existing appearance models for these material types as well as model extensions (Secs. 4.A–4.C).

The perfectly diffuse (or Lambertian) material is a good starting point for objects that exhibit a significant amount of subsurface scattering (a). The BRDF of a perfectly diffuse material is $f_{r,d} = \rho_d/\pi$, where ρ_d is the bihemispherical diffuse reflectance, which we can set in an RGB renderer using a color vector in $[0,1]^3$. This reflectance represents the subsurface scattering of the material. We can then add an interface to model highlights and switch to a bidirectional scattering-surface reflectance distribution function (BSSRDF) to model translucency. The Fresnel equations for reflection are an excellent starting point for metallic and transparent objects (b-c).

The BRDF/BTDF of a perfectly smooth or a rough interface are available from Walter et al. [19]. The BRDFs presented by these authors work just as well for metals as long as we use the complex index of refraction of the metals to find the Fresnel factor. The key difficulty in use of the Fresnel equations is that indices of refraction are physical parameters that are defined as a spectrum rather than colors. We can convert a spectrum to a representative RGB vector using weighted averages based on RGB color matching functions [56, 57]. Assuming known (complex) index of refraction, the key parameter for metallic and transparent objects is the surface roughness (which is different for different surface microfacet distributions [19, 58]).

A natural extension of the diffuse model (a) is to introduce a refractive interface. The BRDF then becomes a sum of a specular and a diffuse component [59]. We can think of the specular term as in-surface scattering and of the diffuse term as subsurface scattering. The Fresnel equations are then useful for ensuring energy conservation (and reciprocity) both for smooth surfaces [60] and for rough surfaces [61, 62]. The trick is to sample the BRDF/BTDF of a transparent surface [19] and then let incident light that refracts into the material reflect diffusely before it refracts back out of the material using the BTDF of the surface again but this time for the outgoing direction. This enables addition of glossy reflections and highlights to an object with an otherwise matte appearance.

A natural extension of the transparent model (b) is to account for absorption based on the distance d that a ray travels through the interior of the object. This is done using an (RGB) absorption coefficient σ_a and Bouguer's law of exponential attenuation of light (attenuation factor $e^{-\sigma_a d}$). The absorption coefficient is directly linked to the imaginary part of the index of refraction [63]. The index of refraction was assumed known, and for metals σ_a is very large. We can thus assume that all light transmitted into a metal is absorbed. However, for transparent objects, σ_a is often very small and may need some adjustment to account for dissolved substances [56] or impurities [57]. The absorption coefficient then becomes an RGB parameter in the model that controls the color of transmitted light.

A further extension of the diffuse model (a) is to replace $f_{r,d}$ with proper subsurface scattering, where light may be incident at one surface position and observed at another. In terms of input parameters, this requires knowledge of the (RGB) scattering coefficient σ_s and the phase function. The latter is the distribution of the scattered light, which is often represented by an analytical model taking an (RGB) asymmetry parameter (g) as input. Several rendering techniques are available for evaluating the volumetric light transport between two surface positions [64]. For highly scattering materials, however, a full-fledged unbiased path tracing technique [65] is unpractical due to long rendering times. We need faster rendering when tuning parameters based on comparison of renderings with a reference photograph. A more practical rendering technique for subsurface scattering is then to use an analytical approximation of the BSSRDF [20, 21].

The standard dipole approximation for subsurface scattering [21] does not model how the direction of the incident light influences the subsurface scattering. To include this component, we can use a directional dipole approximation [20]. However, these models use Fresnel terms that assume a perfectly smooth interface. Donner and Jensen [66] explained how to account for a rough surface with a distribution of microfacet normals [58, 59]. In the following, we describe how to account for a rough surface

in the case of a model that accounts for the directional dependency of the subsurface scattering. We also describe simplistic models that we use to account for spatial variation in the surface roughness of our example objects.

A. Directional Subsurface Scattering for Rough Surfaces

The BSSRDF depends on the object geometry X, the position $\mathbf{x}_i$ and the direction $\vec{\omega}_i$ of the incident light as well as the position $\mathbf{x}_o$ and the direction $\vec{\omega}_o$ of the observed light. The normals at the points of incidence and observation $\vec{n}_i$ and $\vec{n}_o$ are known from the object geometry. An analytic BSSRDF model developed for a material with a smooth surface then usually has the form

$$S(X;\mathbf{x}_i,\vec{\omega}_i;\mathbf{x}_o,\vec{\omega}_o) = F_t(\vec{\omega}_o\cdot\vec{n}_o)(S_d+S^*)F_t(\vec{\omega}_i\cdot\vec{n}_i)\,, \quad (7)$$

where $F_t = 1-F$ is Fresnel transmittance, S_d is the diffusive part, which is typically modeled by a dipole, and S^* is the remaining light transport. The number of arguments used with S_d and S^* is different for different models.

To incorporate a rough surface in a BSSRDF model of this kind, we add a BRDF in the special case where the point of incidence equals the point of emergence, and we insert hemispherical transmittance integrals in place of the Fresnel terms:

$$\begin{aligned} S(X;\mathbf{x}_i,\vec{\omega}_i;\mathbf{x}_o,\vec{\omega}_o) &= \delta(\mathbf{x}_o-\mathbf{x}_i)f_r(\mathbf{x}_o,\vec{\omega}_i,\vec{\omega}_o) \\ &+ \int_{2\pi}\int_{2\pi} f_t(\mathbf{x}_o,\vec{\omega}_{21},\vec{\omega}_o)(-\vec{n}_o\cdot\vec{\omega}_{21})(S_d+S^*)\,d\omega_{21} \\ &\quad f_t(\mathbf{x}_i,\vec{\omega}_i,\vec{\omega}_{12})(-\vec{n}_i\cdot\vec{\omega}_{12})\,d\omega_{12}\,, \end{aligned} \quad (8)$$

where f_r is the BRDF and f_t is the BTDF of the surface, δ is a Dirac delta function, $\vec{\omega}_{12}$ is the direction of a ray transmitted into the volume, and $\vec{\omega}_{21}$ is the direction of a ray to be transmitted out of the volume. The directions $\vec{\omega}_{12}$ and $\vec{\omega}_{21}$ would thus be the ones to use as arguments for the S-functions.

The S^* term is usually fully directional, and the integrations over BTDFs at $\mathbf{x}_i$ and $\mathbf{x}_o$ are evaluated using regular volume path tracing with rough refraction at the interfaces. In the case of the standard dipole [21], $S^* = S^{(1)}$ includes evaluation of single scattering in the volume. In the case of the directional dipole [20], $S^* = S_{\delta E}$ is evaluated in the same way as absorption in a transparent material, but with a modified coefficient in the exponential attenuation. One should note that analytic expressions are available for the Fresnel transmittance integrals in cases where S_d is independent of $\vec{\omega}_i$ and/or $\vec{\omega}_o$ [66, 67]. Some care must be taken as some models [20, 67] assume a diffuse distribution of the light at $\mathbf{x}_o$ and then include the integration over $\vec{\omega}_{21}$ in their formulation. In the case of the directional dipole, our expression becomes

$$\begin{aligned} S(X;\mathbf{x}_i,\vec{\omega}_i;\mathbf{x}_o,\vec{\omega}_o) &= \delta(\mathbf{x}_o-\mathbf{x}_i)f_r(\mathbf{x}_o,\vec{\omega}_i,\vec{\omega}_o) + S^*_{\delta E} \\ &+ \int_{2\pi} S_{d,\mathrm{dir}}(\mathbf{x}_i,\vec{\omega}_{12};\mathbf{x}_o)f_t(\mathbf{x}_i,\vec{\omega}_i,\vec{\omega}_{12})(-\vec{n}_i\cdot\vec{\omega}_{12})\,d\omega_{12}\,, \end{aligned} \quad (9)$$

where $S_{d,\mathrm{dir}}$ is the diffusive part of the BSSRDF in the directional dipole model, but taking the transmitted direction directly as input instead of $\vec{\omega}_i$, and $S^*_{\delta E}$ is the modified reduced intensity term appearing in this model, but here including the BTDF integrations (rough refractions at the interfaces).

Comparing Eq. (8) to common illumination models [1, 59], the first term corresponds to the specular term and the second term corresponds to the diffuse term. The BRDF f_r to be used for the first term should therefore not include an added diffuse term. The BSDF (collective name for BRDF and BTDF) used in Eq. (8) should rather depend only on surface properties, such as a distribution of microfacet normals, see the work of Walter et al. [19] for examples. In particular, we use the so-called GGX distribution developed by these authors. This distribution has a width parameter α_g that we refer to as the GGX roughness.

B. Surface Roughness of a 3D Printed Object

Since most 3D printers print in layers, the surface of a printed object is usually rougher when the intended surface normal points in a direction aligned with layer edges in the voxel cubes of the print volume. If the z-axis is the print direction, we can use the following function to control the GGX roughness (α_g) based on the z-component of the surface normal (n_z):

$$\alpha_g = \rho + (1-\rho)\frac{|\sin(2\theta)|^s}{s} = \rho + (1-\rho)\frac{\left(2|n_z|\sqrt{1-n_z^2}\right)^s}{s}, \quad (10)$$

where θ is the angle of the surface normal $\vec{n}$ with the z-axis. We can think of the user parameters as follows: $\rho \in [0,1]$ is the minimum roughness and $s > 0$ is the shininess, which controls the height and width of the bumps in the curve around angles of $\pm 45°$, $\pm 135°$.

C. Surface Roughness of a Polished Metal Object

Quick hand polishing of a metallic object can result in an object with a rougher surface in curved areas and a smoother surface in flat areas. One way to specify the curvature of an object is using the mean curvature normal $\mathbf{H}$ [68]. This is a quantity that we can precompute for a triangle mesh using vertex circulators and store as a vertex attribute. The dot product of the outward-pointing surface normal $\vec{n}$ and the mean curvature normal $\mathbf{H}$ provides a signed measure of the curvature, where positive is a concavity and negative is a convexity. We use the absolute value of this dot product as an indicator of areas that were maybe not as easy to polish. To reduce noise from the surface scan and set a high roughness for curved areas, we employ a sigmoid function. Our use of the mean curvature normal is demonstrated in Figure 3, and the formula is

$$\alpha_g = \rho + \frac{1-\rho}{1+\exp(s\,(1-30\,|\mathbf{H}\cdot\vec{n}|))}\,, \quad (11)$$

where ρ is again minimum roughness and s is a sort of shininess while $\mathbf{H}$ is the mean curvature normal after division by the length of the longest mean curvature normal in the triangle mesh.

5. RENDERING

We implemented a progressive unidirectional path tracer using OptiX [69]. To include subsurface scattering, we sample a new set of surface positions for each progressive update. For each update and within each pixel, the ray tracer generates a random position $\mathbf{x}_p$ in pixel coordinates. With the rotation of the camera relative to the object $\mathbf{R}$ and the camera intrinsic matrix $\mathbf{K}$, we get the direction of the corresponding ray using

$$\vec{\omega} = (\mathbf{KR})^{-1}\mathbf{S}\,\mathbf{x}_p = \mathbf{R}^T\mathbf{K}^{-1}\mathbf{S}\,\mathbf{x}_p\,. \quad (12)$$

Since the intrinsic matrix $\mathbf{K}$ is locked to the resolution of the camera ($W_c \times H_c$), which is usually very high, we use the scaling matrix $\mathbf{S} = \mathrm{diag}(W_r/W_c, H_r/H_c, 1)$ to enable rendering in a different resolution ($W_r \times H_r$).

Fig. 3. Model of spatially varying roughness α_g for an aluminium bust that has from time to time been subjected to hand polishing. We use the dot product of the mean curvature normal $\mathbf{H}$ and the surface normal $\vec{n}$. The model correctly marks eyes, hair, nostrils, and engraved letters as rough, but also incorrectly marks edges along the box-like base of the bust as being very rough.

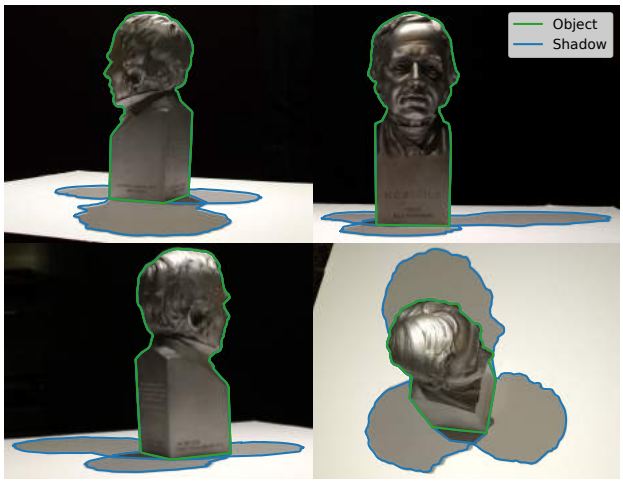


Fig. 4. Each image is an additive blend of three photos of the bust illuminated by the light source at different positions and overlaid with aligned silhouettes of the digital object.

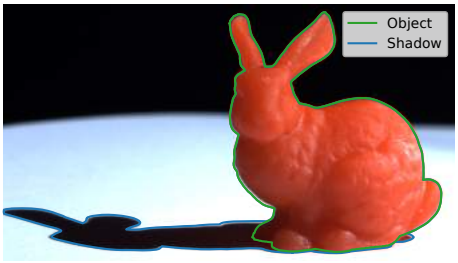


Fig. 5. Photo of the bunny overlaid with the aligned silhouette of the digital object.

6. RESULTS

The three objects of interest are (a) a translucent 3D print of the Stanford bunny, (b) an aluminium bust, and (c) a cupped angel figurine 3D scanned and printed using almost transparent resin. Two of our test objects (b-c) were 3D scanned using structured light based on phase shifting [18]. The employed 3D scanner has a precision of around 100 μm [70]. Our 3D printed objects (a, c) were produced using vat photopolymerization additive manufacturing processes. In our pose estimation and renderings, we used the geometry of these objects without correction for print artifacts. The Stanford bunny was printed by Luongo et al. [71] using red Industrial Blend resin (manufactured by Fun To Do) and a digital light processing (DLP) printer developed for research. The vertical resolution of this printer is 18 μm and the horizontal resolution is 15.08 μm. The angel was printed using general-purpose resin IM2.0 GP1 (manufactured by AddiFab) and a Peopoly Moai stereolithography (SLA) printer. The laser spot size (horizontal resolution) of this printer is 70 μm, and we used a vertical resolution of 50 μm. In simulation, we use a real index of refraction of 1.54 for the printed objects as this is in the middle of the range of commercial acrylic resins with low shrinkage after photopolymerization [72]. Our three objects all have a rough surface and exhibit different types of spatial variation in this roughness.

We used our method to align renderings of the objects of interest with their photographs. We then tested different appearance models following the presented guidelines, where we started from a simplistic model and gradually added complexity. In each case, our end result is an appearance model and a rendering paired with a photograph for validation that would serve as a suitable starting point for an inverse rendering technique. The optical properties that we estimated for our different objects are in Table 1. The reference photographs and the associated CAD files and relative camera and light source alignments will be available as a supplement. We encourage the reader to use this dataset for testing preferred appearance models and rendering software. Another option is to use the dataset for finding better optical properties including better spatial variation of surface roughness by means of inverse rendering.

A. Acquisition

The objects were placed on a flat piece of paper and illuminated by a Thorlabs MNWHL4 LED light source. This source is neutral white with a point-like radiation distribution within an angular diameter of 10°. The bunny (a) and the angel (c) were captured using a FLIR Grasshopper3 GS3-U3-60QS6C-C camera, while the bust (b) was captured using D3200, D7000, D7500 and D750 cameras from Nikon. We used four cameras to cover all angles of the object while also taking multiple images from the same positions with different light positions. As different cameras were used, the images of the bust were color calibrated using a ColorChecker from X-Rite. All images were captured with a small aperture so that all parts of the object and shadow were in focus. We performed camera calibration [44, 73] using a ChArUco board which is a checkerboard with ArUco markers [74]. For the bunny (a), we did not use the estimated extrinsics and only used the estimated focal length from the intrinsics. Lens distortion from the camera calibrations were used to undistort the reference photographs and ground truth silhouettes.

B. Alignment

To segment the photographs as required by our alignment method, we used thresholding followed by hole closing and selected the largest connected component. For the images of the bunny and the angel, some manual cleaning of the segmentation was necessary due to caustics.

Our test cases span different setups to showcase the flexibility of our alignment method. For the bunny (a), we use just a single

Table 1. Estimated optical properties.

Material	n	σ_a	σ_s	ρ	s
Bunny (FTD, red Industrial Blend)	1.54	$(0.33, 25, 67) \cdot 10^3$ m^{-1}	$(10, 21, 0.083) \cdot 10^3$ m^{-1}	0.20	2.4
Angel (AddiFab, IM2.0 GP1)	1.54	$(0.032, 32, 640)$ m^{-1}	0	0.15	5.0
Bust (aluminium)	$(1.04, 0.76, 0.49) + i\,(6.45, 5.73, 4.76)$	$1.3 \cdot 10^8$	n/a	0.22	4.5

Fig. 6. Photos of the angel overlaid with the aligned silhouette of the digital object.

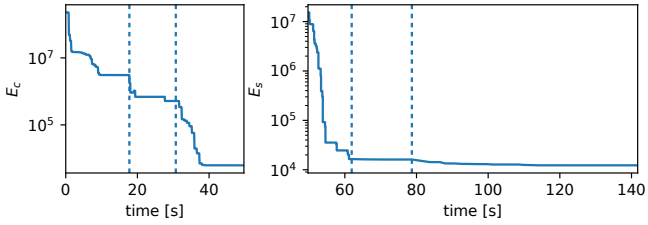


Fig. 7. Convergence plots of our pose estimation method applied to the bunny, showing the total time elapsed, with vertical dashed lines separating the different steps of our method. Left: as described in Sec. 3.D, steps: translation optimization, random rotation search, pose optimization. Right: as described in Sec. 3.E, steps: random light search, light position optimization, joint optimization.

picture with unknown camera pose to align the scene. For the angel (c), we use two camera poses and a single light position to do the estimation. Finally, the bust (b) was captured from four camera poses, each with four different light source positions, yielding a total of 16 images that we used to do the alignment. The more light source positions, the more information we have available for the pose estimation. This comes at the small cost of increasing the dimensionality of the optimization problem. If we again consider our method an enabler for inverse rendering, it is an advantage to have multiple light positions as these provide additional samples for estimation of BRDFs, for example.

Outputs from our alignment method are in Figures 4 to 6. We achieve good alignment of the outlines of the bust, which makes sense as this is the only object in our collection for which the geometry is directly from the photographed object. Both the angel and the bunny have a quite good alignment, but especially the bunny has noticeable differences between the rendered silhouette and the object. We presume these mostly stem from non-linear shrinkage during printing that our method cannot account for. For the angel, our method estimated shrinkage of 3%, 6%, 1% in the x, y, z directions as compared to the size of an ideal 3D print.

As the bunny (a) is the more difficult case (with only one view and light source position to constrain the problem), we have analyzed the performance of our method more closely for this case. Convergence plots in Figure 7 show that each step

RMSE: 0.1315 SSIM: 0.8065 Liu et al. [15] for object pose

RMSE: 0.1304 SSIM: 0.7405 ours w/o joint estimation

RMSE: 0.1124 SSIM: 0.8181 ours final

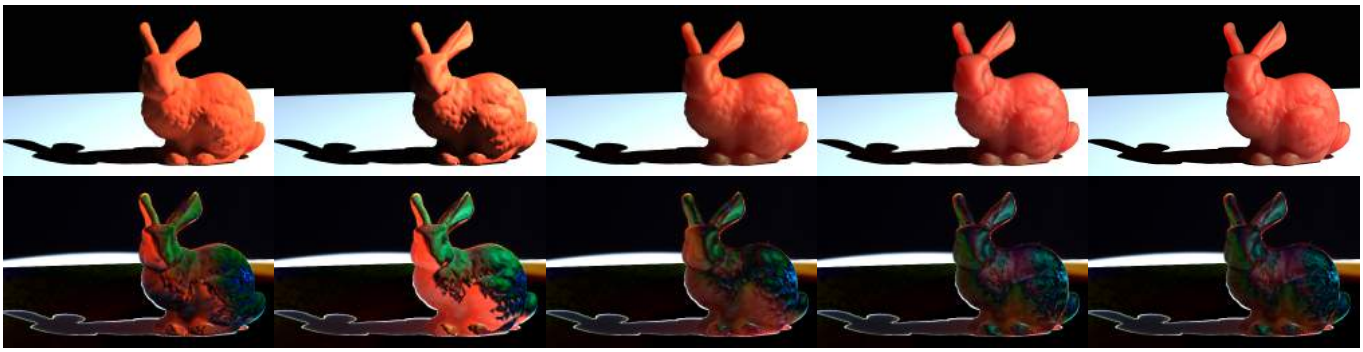
Fig. 8. Ablation study shown with signed difference images $\times 2$. Blue and red indicate positive and negative differences (for rendering minus photograph): average of the color bands in the third and the fourth column of Figure 1, respectively.

improves the similarity (reduces E_c and E_s). While the joint optimization in the last step gives a smaller improvement of E_s than other steps, the improvement of the final rendered result is significant as seen in Figure 8. We also compare our alignment result with an object pose obtained using the differentiable rendering method of Liu et al. [15]. We observe that the performance of this related work is similar to ours without joint optimization and we needed many random initial guesses with this method too in order for it to converge to a good solution. With other methods than ours, we do not get the advantages of jointly estimating light source position and mesh shrinkage. In the result found using the method of Liu et al. (Figure 8, left), we used the camera pose and light source position from our final result. The key benefit of our work is thus collective extraction of information available in projected silhouettes (object pose, light source position, mesh shrinkage), and that we can use joint optimization to collectively improve each part of the result.

C. Appearance

Since our objects are placed on a piece of paper assumed to be flat, we place a quad in the ground plane and resize it manually to approximately fit the paper observed in the photograph. Precise alignment of the paper could be part of the object alignment, but we find that it is not so important with respect to testing the appearance model applied to the object. To start simple, we consider the paper to be a diffuse surface. More complexity could easily be added to the paper appearance model [75], but we focus our attention on the objects of interest.

We initialise the diffuse reflectance of the paper to $\rho_d = (0.8, 0.8, 0.8)$ and select the simplest shading model for the material category of the object in question. We then use the intensity of the light reflected from the paper to estimate the intensity of the point light. Since our source is neutral white, we use the same intensity in all color bands. An easy way to do a comparison is using two colored difference images: one for positive difference and one for negative difference (see examples in Figure 1). Once the light intensity has been set, we modify the reflectance values until each color appears equally in the positive and the negative difference image. We also evaluate our results



RMSE: 0.1152	RMSE: 0.1237	RMSE: 0.1136	RMSE: 0.1127	RMSE: 0.1124
SSIM: 0.8108	SSIM: 0.7931	SSIM: 0.8145	SSIM: 0.8177	SSIM: 0.8181
Lambertian	interfaced	standard SSS	directional SSS	SV roughness

Fig. 9. Renderings (top) and absolute difference images $\times 2$ (bottom) to test appearance models for the rough translucent bunny. The interfaced model adds a rough surface with a GGX microfacet normal distribution [19, 60, 61]. The standard subsurface scattering (SSS) model is the standard dipole including path traced single scattering and a rough surface [21, 66]. The directional SSS model uses the directional dipole [20] and incorporates a rough surface (Sec. 4.A). The model with spatially varying (SV) roughness uses Eq. (10). Further comparison of the input image with the end result is in Figure 1. The not quite so flat paper worsens both RMSE and SSIM by approximately 0.05.

quantitatively using root-mean-squared error (RMSE) (lower is better) and structural similarity (SSIM) index [76] (higher is better). The initial results for each of our three test cases are leftmost in Figures 9 to 11.

To estimate absorption and scattering coefficients (σ_a and σ_s), we need the physical size of the object as these optical properties are measured per distance unit that a ray has travelled through the material. Using the physical dimensions of the object, we get the coefficients in Table 1. We decided to leave the phase function as isotropic ($g = 0$) since the analytic BSSRDF models mostly use the reduced scattering coefficient $\sigma_s' = \sigma_s(1 - g)$ and thus do not distinguish much between a reduction in σ_s and an increase of g. The directional dipole is not exclusively based on the reduced scattering coefficient, but the role of g seems limited. When estimating the coefficients, 10 over the length of the bounding box diagonal is usually a good value to start with for the absorption or the scattering to have a reasonable effect.

Refinement of the model for the rough translucent bunny (a). Figure 9. We first add an interface to the model [60, 61] to enable rendering of highlights. However, this also directs a lot of energy into a glossy reflection lobe meaning that the missing transport of light from the point of incidence to a different point of emergence becomes apparent and RMSE and SSIM both worsen. As soon as we switch to a model that accounts for this subsurface light transport [21], the result becomes better than the Lambertian model. This is true even without single scattering and assuming that the surface is perfectly smooth. The directional dipole [20] and our spatially varying roughness from Sec. 4.B further improve the result. However, the models cannot fully represent the scattering process. This is probably due to limiting assumptions such as diffuse emergent light and a locally flat, convex object. It should be mentioned that the bunny was printed using greyscale values to reduce staircasing artefacts [71]. These staircasing artefacts due to layered printing are significantly less pronounced for the bunny as compared with the angel (which was not printed using greyscale values). Nevertheless, the bunny object still exhibits some spatial variation in its roughness that we have modelled.

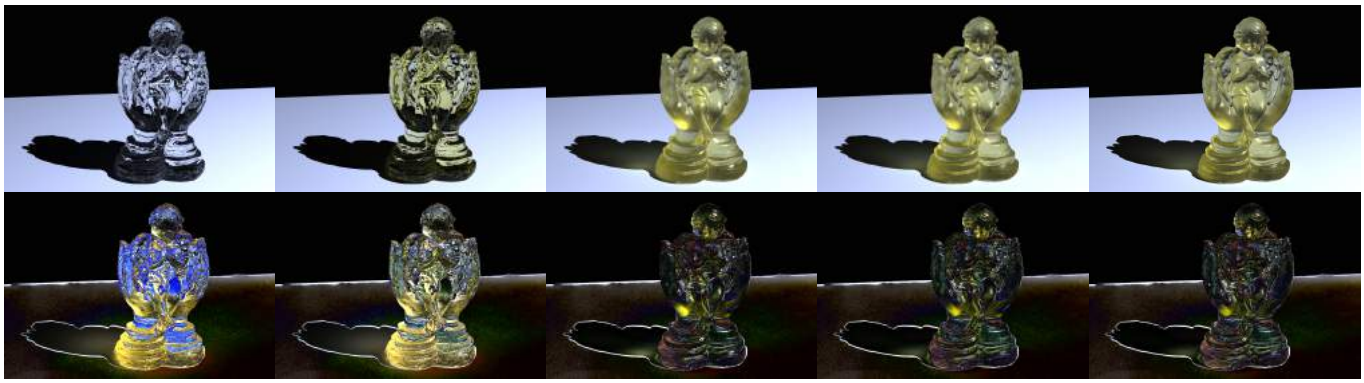
Refinement of the model for the aluminium bust (b). Figure 11. We use the complex index of refraction of aluminium from McPeak et al. [77] (this is available for download at refractiveindex.info). Since we have a dark scene with a point light, the appearance is off without surface roughness (as highlights then disappear). Adding a microfacet normal distribution was thus essential for this case, and we found that the GGX distribution [19] provided a good result. When adding spatially varying roughness based on the curvature, we found that SSIM would improve for a larger shininess s at the cost of a poorer RMSE. The SSIM-improved result is in Figure 1. The RMSE probably suffers from a slight misplacement of the highlight peak in the forehead of the bust.

Refinement of the model for the rough transparent angel (c). Figure 10. Using the convention that surface normals always point outwards, absorption is easily included by applying Bouguer's law of exponential attenuation to all rays that hit the surface from the inside. Accounting for absorption and a rough interface is highly important when modelling the appearance of the angel. Apart from this, the print layers are visually obvious, especially in highlights. We, therefore, tried to model the layers by calculating a layer index based on the point of intersection and using an increased roughness for every second layer. This represents the rougher layer edges more explicitly. Visually, we find this layered result more convincing and it also has lower RMSE, but SSIM disagrees. We tried adding single scattering to the material, but this only seemed to worsen RMSE and SSIM. Thus, it seems that the remaining deviations from the reference are mostly due to geometric print artefacts and inaccuracies in the spatial variation of the surface roughness.

7. DISCUSSION

Although our method is able to quantify the differences between a rendering and a photograph, it does not provide a direct way of determining what the source of these differences are. However, when a change of reflectance model leads to a smaller error, it is very likely that the previous model was a source of error.

While we use a pinhole camera model, one should note that our method can also work for more advanced camera models as



RMSE: 0.1267	RMSE: 0.1135	RMSE: 0.0741	RMSE: 0.0733	RMSE: 0.0730
SSIM: 0.7699	SSIM: 0.8051	SSIM: 0.8876	SSIM: 0.8870	SSIM: 0.8858
glass	absorbing	rough	SV roughness	layered

Fig. 10. Renderings (top) and absolute difference images $\times 2$ (bottom) to test appearance models for the rough transparent angel. We use the GGX microfacet normal distribution [19] and add absorption through analysis by synthesis [57] and spatially varying (SV) roughness (Sec. 4.B). We also tested a layered variation of the roughness in the print direction (every second layer is rougher to model a staircase). SSIM is sensitive to structure and takes a hit because the layers do not perfectly match the real layers. Further comparison of the input image with the end result is in Figure 1.

RMSE: 0.0311	RMSE: 0.0159	RMSE: 0.0134
SSIM: 0.9339	SSIM: 0.9693	SSIM: 0.9713
metallic	rough	SV roughness

Fig. 11. Renderings (top) and absolute difference images $\times 2$ (bottom) to test appearance models for the aluminium bust. We test spatially varying (SV) roughness as depicted in Figure 3 and use high dynamic range when computing differences. The input image is in Figure 1, where it is compared with an SSIM-improved result (RMSE: 0.0148, SSIM: 0.9725).

we can apply the necessary transformations to the object edges before computing the silhouette. Extending to area lights is however challenging and left for future work.

A disadvantage of using silhouettes is their simplicity. In some cases, they describe the features of an object inadequately, which can cause ambiguities in the pose estimation. An example of this could be a bowl with contents, where the silhouette only contains information enough to pose estimate the bowl. To have more information, some methods [29] also use features on the object itself. In cases where the segmentation has inaccuracies and our pose may have small errors, our method is still useful for obtaining a good initial guess that can be refined by other methods (such as differentiable rendering).

8. CONCLUSION

We presented a practical method for aligning photographs with rendered images. Our method is based on silhouette matching and estimates both object pose and the position of a point-like light source. If multiple images have been captured from different views and/or with light sources in different positions, our method can include this added information in the pose estimation. As opposed to differentiable rendering techniques, our method works not only in pixel space but in the entire image plane. This means that we can estimate a pose from a very poor initial guess. Thus we find our work a practical enabling technique for inverse rendering that could be based on differentiable rendering.

Given an alignment, we proposed a procedure for composing an appearance model suitable for the photographed object. The concept is to start from a simplistic model and gradually increase the complexity of appearance models guided by difference images and quantitative metrics such as RMSE and SSIM. As a consequence of this approach, we presented extensions of existing models providing improved photorealism. One extension was the combination of a rough surface with directional subsurface scattering. We believe that practical alignment of photographs with renderings is an important step in furthering the predictive abilities of appearance models.

Funding. Innovationsfonden (6151-00006B, 6151-00005B); Poul Due Jensen Foundation (2018-017).

Acknowledgments. Thanks to Macarena Mendez Ribo for 3D printing the transparent angel figurine and to Andreas Bærentzen for useful discussions on finding a silhouette given projected polygon edges.

Disclosures. Mads Emil Brix Doest: LEGO Group (F). Jakob Wilm: Euler3D ApS (I), Calib.io I/S (I).

Disclosures. The authors declare no conflicts of interest.

REFERENCES

1. B. T. Phong, "Illumination for computer generated pictures," Commun. ACM **18**, 311–317 (1975).
2. C. M. Goral, K. E. Torrance, D. P. Greenberg, and B. Battaile, "Modeling the interaction of light between diffuse surfaces," Comput. Graph. (SIGGRAPH '84) **18**, 213–222 (1984).
3. G. W. Meyer, H. E. Rushmeier, M. F. Cohen, D. P. Greenberg, and K. E. Torrance, "An experimental evaluation of computer graphics imagery," ACM Transactions on Graph. **5**, 30–50 (1986).
4. H. Rushmeier, G. Ward, C. Piatko, P. Sanders, and B. Rust, "Comparing real and synthetic images: Some ideas about metrics," in *Rendering Techniques '95,* (Springer, 1995), pp. 82–91.
5. S. N. Pattanaik, J. A. Ferwerda, K. E. Torrance, and D. P. Greenberg, "Validation of global illumination solutions through CCD camera measurements," in *Proceedings of Color Imaging Conference (CIC 1997),* (1997), pp. 250–253.
6. C. Ulbricht, A. Wilkie, and W. Purgathofer, "Verification of physically based rendering algorithms," Comput. Graph. Forum **25**, 237–255 (2006).
7. M. Weinmann and R. Klein, "Advances in geometry and reflectance acquisition (course notes)," in *Proceedings of SIGGRAPH Asia 2015 Courses,* (ACM, 2015).
8. C. Reinbacher, M. Ruther, and H. Bischof, "Pose estimation of known objects by efficient silhouette matching," in *Proceedings of International Conference on Pattern Recognition (ICPR 2010),* (IEEE, 2010), pp. 1080–1083.
9. S. Peng, Y. Liu, Q. Huang, X. Zhou, and H. Bao, "PVNet: Pixel-wise voting network for 6DoF pose estimation," in *Proceedings of CVPR 2019,* (2019), pp. 4561–4570.
10. A. Panagopoulos, C. Wang, D. Samaras, and N. Paragios, "Illumination estimation and cast shadow detection through a higher-order graphical model," in *Proceedings of CVPR 2011,* (IEEE, 2011), pp. 673–680.
11. J. Lopez-Moreno, E. Garces, S. Hadap, E. Reinhard, and D. Gutierrez, "Multiple light source estimation in a single image," Comput. Graph. Forum **32**, 170–182 (2013).
12. R. Ramamoorthi and P. Hanrahan, "A signal-processing framework for inverse rendering," in *Proceedings of SIGGRAPH 2001,* (ACM, 2001), pp. 117—128.
13. G. Loubet, N. Holzschuch, and W. Jakob, "Reparameterizing discontinuous integrands for differentiable rendering," ACM Transactions on Graph. **38**, 228:1–228:14 (2019).
14. M. Nimier-David, D. Vicini, T. Zeltner, and W. Jakob, "Mitsuba 2: a retargetable forward and inverse renderer," ACM Transactions on Graph. **38**, 203:1–203:17 (2019).
15. S. Liu, W. Chen, T. Li, and H. Li, "Soft rasterizer: Differentiable rendering for unsupervised single-view mesh reconstruction," in *Proceedings of ICCV 2019,* (2019), pp. 7708–7717.
16. G. Turk, "The Stanford bunny," https://www.cc.gatech.edu/~turk/bunny/bunny.html (2000).
17. G. Turk and M. Levoy, "Zippered polygon meshes from range images," in *Proceedings of SIGGRAPH '94,* (1994), pp. 311–318.
18. J. Geng, "Structured-light 3D surface imaging: a tutorial," Adv. Opt. Photonics **3**, 128–160 (2011).
19. B. Walter, S. R. Marschner, H. Li, and K. E. Torrance, "Microfacet models for refraction through rough surfaces," in *Proceedings of Eurographics Symposium on Rendering (EGSR 2007),* (The Eurographics Association, 2007), pp. 195–206.
20. J. R. Frisvad, T. Hachisuka, and T. K. Kjeldsen, "Directional dipole model for subsurface scattering," ACM Transactions on Graph. **34**, 5:1–5:12 (2014).
21. H. W. Jensen, S. R. Marschner, M. Levoy, and P. Hanrahan, "A practical model for subsurface light transport," in *Proceedings of SIGGRAPH 2001,* (ACM, 2001), pp. 511–518.
22. H. Lensch, J. Kautz, M. Goesele, W. Heidrich, and H.-P. Seidel, "Image-based reconstruction of spatial appearance and geometric detail," ACM Transactions on Graph. **22**, 234–257 (2003).
23. M. Holroyd, J. Lawrence, and T. Zickler, "A coaxial optical scanner for synchronous acquisition of 3D geometry and surface reflectance," ACM Transactions on Graph. **29**, 99:1–99:12 (2010).
24. M. M. Loper and M. J. Black, "OpenDR: An approximate differentiable renderer," in *Proceedings of ECCV 2014,* (Springer, 2014), pp. 154–169.
25. T.-M. Li, M. Aittala, F. Durand, and J. Lehtinen, "Differentiable Monte Carlo ray tracing through edge sampling," ACM Transactions on Graph. **37**, 222:1–222:11 (2018).
26. B. Bhanu, "CAD-based robot vision," Computer **20**, 13–16 (1987).
27. J. Byne and J. A. D. W. Anderson, "A CAD-based computer vision system," Image Vis. Comput. **16**, 533–539 (1998).
28. M. Ulrich, C. Wiedemann, and C. Steger, "CAD-based recognition of 3D objects in monocular images," in *Proceedings of ICRA 2009,* (IEEE, 2009), pp. 2090–2097.
29. A. Petit, E. Marchand, R. Sekkal, and K. Kanani, "3D object pose detection using foreground/background segmentation," in *Proceedings of ICRA 2015,* (IEEE, 2015), pp. 1858–1865.
30. B. Rosenhahn, T. Brox, D. Cremers, and H.-P. Seidel, "A comparison of shape matching methods for contour based pose estimation," in *International Workshop on Combinatorial Image Analysis,* (Springer, 2006), pp. 263–276.
31. O. Tahri and F. Chaumette, "Complex objects pose estimation based on image moment invariants," in *Proceedings of ICRA 2005,* (IEEE, 2005), pp. 436–441.
32. O. Tahri, H. Araujo, Y. Mezouar, and F. Chaumette, "Efficient iterative pose estimation using an invariant to rotations," IEEE Transactions on Cybern. **44**, 199–207 (2013).
33. M.-K. Hu, "Visual pattern recognition by moment invariants," IRE transactions on information theory **8**, 179–187 (1962).
34. M. Zhu, K. G. Derpanis, Y. Yang, S. Brahmbhatt, M. Zhang, C. Phillips, M. Lecce, and K. Daniilidis, "Single image 3D object detection and pose estimation for grasping," in *Proceedings of ICRA 2014,* (IEEE, 2014), pp. 3936–3943.
35. Z. Cao, Y. Sheikh, and N. K. Banerjee, "Real-time scalable 6DOF pose estimation for textureless objects," in *Proceedings of ICRA 2016,* (IEEE, 2016), pp. 2441–2448.
36. E. Brachmann, F. Michel, A. Krull, M. Ying Yang, S. Gumhold *et al.*, "Uncertainty-driven 6D pose estimation of objects and scenes from a single RGB image," in *Proceedings CVPR 2016,* (2016), pp. 3364–3372.
37. W. Kehl, F. Manhardt, F. Tombari, S. Ilic, and N. Navab, "SDD-6D: Making RGB-based 3D detection and 6D pose estimation great again," in *Proceedings of ICCV 2017,* (2017), pp. 1521–1529.
38. M. Rad and V. Lepetit, "BB8: A scalable, accurate, robust to partial occlusion method for predicting the 3D poses of challenging objects without using depth," in *Proceedings of ICCV 2017,* (2017), pp. 3828–3836.
39. B. Tekin, S. N. Sinha, and P. Fua, "Real-time seamless single shot 6D object pose prediction," in *Proceedings CVPR 2018,* (2018), pp. 292–301.
40. Y. Li, G. Wang, X. Ji, Y. Xiang, and D. Fox, "DeepIM: Deep iterative matching for 6D pose estimation," in *Proceedings of ECCV 2018,* (2018), pp. 683–698.
41. J. Blinn, "Me and my (fake) shadow," IEEE Comput. Graph. Appl. **8**, 82–86 (1988).
42. N. Chotikakamthorn, "Near point light source location estimation from shadow edge correspondence," in *Proceedings of Cybernetics and Intelligent Systems (CIS) and Robotics, Automation and Mechatronics (RAM),* (IEEE, 2015), pp. 30–35.
43. S. Suzuki and K. Abe, "Topological structural analysis of digitized binary images by border following," Comput. Vision, Graph. Image Process. **30**, 32–46 (1985).
44. G. Bradski, "The OpenCV Library," Dr. Dobb's J. Softw. Tools (2000).
45. U. Ramer, "An iterative procedure for the polygonal approximation of plane curves," Comput. Graph. Image Process. **1**, 244–256 (1972).
46. D. H. Douglas and T. K. Peucker, "Algorithms for the reduction of the number of points required to represent a digitized line or its caricature," Cartogr. The Int. J. for Geogr. Inf. Geovisualization **10**, 112–122 (1973).
47. F. Antonio, "Faster line segment intersection," in *Graphics Gems III,*

D. Kirk, ed. (Academic Press, 1992), pp. 199–202.
48. P. Bénard and A. Hertzmann, "Line drawings from 3d models: a tutorial," Foundations Trends Comput. Graph. Vis. **11**, 159 (2019).
49. P. Alliez, S. Tayeb, and C. Wormser, "3D fast intersection and distance computation," CGAL user reference manual **3** (2016).
50. X. Y. Jiang and H. Bunke, "Simple and fast computation of moments," Pattern Recognit. **24**, 801–806 (1991).
51. K. Levenberg, "A method for the solution of certain non-linear problems in least squares," Q. Appl. Math. **2**, 164–168 (1944).
52. D. W. Marquardt, "An algorithm for least-squares estimation of nonlinear parameters," J. Soc. for Ind. Appl. Math. **11**, 431–441 (1963).
53. R. Candelier, "Tracking object orientation with image moments," http://raphael.candelier.fr/?blog=Image%20Moments (2016).
54. J. R. Frisvad, S. A. Jensen, J. S. Madsen, A. Correia, L. Yang, S. K. S. Gregersen, Y. Meuret, and P.-E. Hansen, "Survey of models for acquiring the optical properties of translucent materials," Comput. Graph. Forum **39** (2020). To appear.
55. F. E. Nicodemus, J. C. Richmond, J. J. Hsia, I. W. Ginsberg, and T. Limperis, "Geometrical considerations and nomenclature for reflectance," Tech. Rep. NBS MN-160, National Bureau of Standards (1977).
56. J. R. Frisvad, N. J. Christensen, and H. W. Jensen, "Computing the scattering properties of participating media using Lorenz-Mie theory," ACM Transactions on Graph. **26**, 60:1–60:10 (2007).
57. J. D. Stets, A. Dal Corso, J. B. Nielsen, R. A. Lyngby, S. H. N. Jensen, J. Wilm, M. B. Doest, C. Gundlach, E. R. Eiriksson, K. Conradsen, A. B. Dahl, J. A. Bærentzen, J. R. Frisvad, and H. Aanæs, "Scene reassembly after multimodal digitization and pipeline evaluation using photorealistic rendering," Appl. Opt. **56**, 7679–7690 (2017).
58. R. L. Cook and K. E. Torrance, "A reflectance model for computer graphics," ACM Transactions on Graph. **1**, 7–24 (1982).
59. K. E. Torrance and E. M. Sparrow, "Theory for off-specular reflection from roughened surfaces," J. Opt. Soc. Am. **57**, 1105–1114 (1967).
60. P. Shirley, B. Smits, H. Hu, and E. Lafortune, "A practitioners' assessment of light reflection models," in *Proceedings Pacific Graphics 97,* (1997), pp. 40–49.
61. M. Ashikmin, S. Premože, and P. Shirley, "A microfacet-based BRDF generator," in *Proceedings of SIGGRAPH 2000,* (ACM/Addison-Wesley, 2000), pp. 65–74.
62. L. Simonot, "Photometric model of diffuse surfaces described as a distribution of interfaced Lambertian facets," Appl. Opt. **48**, 5793–5801 (2009).
63. M. Born and E. Wolf, *Principles of Optics: Electromagnetic Theory of Propagation, Interference and Diffraction of Light* (Cambridge University Press, 1999), seventh (expanded) ed.
64. M. Pharr, W. Jakob, and G. Humphreys, *Physically Based Rendering: From Theory to Implementation* (Morgan Kaufmann/Elsevier, 2017), 3rd ed.
65. M. Raab, D. Seibert, and A. Keller, "Unbiased global illumination with participating media," in *Monte Carlo and Quasi-Monte Carlo Methods 2006,* (Springer, 2008), pp. 591–605.
66. C. Donner and H. W. Jensen, "Light diffusion in multi-layered translucent materials," ACM Transactions on Graph. **24**, 1032–1039 (2005).
67. E. d'Eon and G. Irving, "A quantized-diffusion model for rendering translucent materials," ACM Transactions on Graph. **30**, 56:1–56:13 (2011).
68. J. A. Bærentzen, J. Gravesen, F. Anton, and H. Aanæs, *Guide to Computational Geometry Processing: Foundations, Algorithms, and Methods* (Springer, 2012).
69. S. G. Parker, J. Bigler, A. Dietrich, H. Friedrich, J. Hoberock, D. Luebke, D. McAllister, M. McGuire, K. Morley, A. Robison, and M. Stich, "OptiX: A general purpose ray tracing engine," ACM Transactions on Graph. **29**, 66:1–66:13 (2010).
70. E. R. Eiríksson, J. Wilm, D. B. Pedersen, and H. Aanæs, "Precision and accuracy parameters in structured light 3-D scanning," The Int. Arch. Photogramm. Remote. Sens. Spatial Inf. Sci. **40**, 7 (2016).
71. A. Luongo, V. Falster, M. B. Doest, M. M. Ribo, E. R. Eiriksson, D. B. Pedersen, and J. R. Frisvad, "Microstructure control in 3D printing with digital light processing," Comput. Graph. Forum **39**, 347–359 (2020).
72. F. Aloui, L. Lecamp, P. Lebaudy, and F. Burel, "Refractive index evolution of various commercial acrylic resins during photopolymerization," eXPRESS Polym. Lett. **12**, 966–971 (2018).
73. Z. Zhang, "A flexible new technique for camera calibration," IEEE Transactions on pattern analysis machine intelligence **22**, 1330–1334 (2000).
74. S. Garrido-Jurado, R. Muñoz-Salinas, F. J. Madrid-Cuevas, and M. J. Marín-Jiménez, "Automatic generation and detection of highly reliable fiducial markers under occlusion," Pattern Recognit. **47**, 2280–2292 (2014).
75. M. Papas, K. de Mesa, and H. W. Jensen, "A physically-based BSDF for modeling the appearance of paper," Comput. Graph. Forum **33**, 133–142 (2014).
76. Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, "Image quality assessment: From error visibility to structural similarity," IEEE Transactions on Image Process. **13**, 600–612 (2004).
77. K. M. McPeak, S. V. Jayanti, S. J. Kress, S. Meyer, S. Iotti, A. Rossinelli, and D. J. Norris, "Plasmonic films can easily be better: rules and recipes," ACS Photonics **2**, 326–333 (2015).

CONTRIBUTION F

A benchmark and evaluation of non-rigid structure from motion

Noname manuscript No.
(will be inserted by the editor)

A Benchmark and Evaluation of Non-Rigid Structure from Motion

Sebastian Hoppe Nesgaard Jensen,
Mads Emil Brix Doest,
Henrik Aanæs,
Alessio Del Bue

Received: date / Accepted: date

Abstract Non-Rigid structure from motion (NRS*f*M), is a long standing and central problem in computer vision and its solution is necessary for obtaining 3D information from multiple images when the scene is dynamic. A main issue regarding the further development of this important computer vision topic, is the lack of high quality data sets. We here address this issue by presenting a data set created for this purpose, which is made publicly available, and considerably larger than the previous state of the art. To validate the applicability of this data set, and provide an investigation into the state of the art of NRS*f*M, including potential directions forward, we here present a benchmark and a scrupulous evaluation using this data set. This benchmark evaluates 18 different methods with available code that reasonably spans the state of the art in sparse NRS*f*M. This new public data set and evaluation protocol will provide benchmark tools for further development in this challenging field.

Sebastian Hoppe Nesgaard Jensen
DTU Compute, Denmark
E-mail: snje@dtu.dk

Mads Emil Brix Doest
DTU Compute, Denmark
E-mail: mebd@dtu.dk

Henrik Aanæs
DTU Compute, Denmark
E-mail: aanes@dtu.dk
Last author with equal contribution.

Alessio Del Bue
Pattern Analysis and Computer Vision (PAVIS)
Visual Geometry and Modelling (VGM) Lab
Istituto Italiano di Tecnologia (IIT), Genova, 08028, Italy.
E-mail: alessio.delbue@iit.it
Last author with equal contribution.

Keywords Non-Rigid Structure from Motion · Dataset · Evaluation · Deformation Modelling

1 Introduction

The estimation of structure from motion (SfM) using a monocular image sequence is one of the central problems in computer vision. This problem has received a lot of attention, and truly impressive advances have been made over the last ten to twenty years [38,63,52]. It plays a central role in robot navigation, self-driving cars, and 3D reconstruction of the environment, to mention a few. A central part of maturing regular SfM is the availability of sizeable data sets with rigorous evaluations, e.g. [48][1].

The regular SfM problem, however, primarily deals with rigid objects, which is somewhat at odds with the world we see around us. That is, trees sway, faces express themselves in various expressions, and organic objects are generally non-rigid. The issue of making this obvious and necessary extension of the SfM problem is referred to as the non-rigid structure from motion problem (NRS*f*M). A problem that also has a central place in computer vision. The solution to this problem is, however, not as mature as the regular SfM problem. A reason for this is certainly the intrinsic difficulty of the problem and the scarcity of high quality data sets and accompanying evaluations. Such data and evaluations allow us to better understand the problem domain and better determine what works best and why.

To address this issue, we here introduce a high quality data set, with accompanying ground truth (or reference data to be more precise) aimed at evaluating non-rigid structure from motion. To the best of our knowledge, this data set is significantly larger and more

diverse than what has previously been available – c.f. Section 3 for a comparison to previous evaluations of NRS*f*M. The presented data set better capture the variability of the problem and gives higher statistical strength of the conclusions reached via it. Accompanying this data set, we have conducted an evaluation of 18 state of the art methods, hereby validating the suitability of our data set, and providing insight into the state of the art within NRS*f*M. This evaluation was part of the competition we held at a CVPR 2017 workshop, and still ongoing. It is our hope and belief that this data set and evaluation will help in furthering the state of the art in NRS*f*M research, by providing insight and a benchmark. The data set is publicly available at `http://nrsfm2017.compute.dtu.dk/dataset` together with the description of the evaluation protocol.

This paper is structured by first giving an overview of the NRS*f*M problem, followed by a general description of related work, wrt. other data sets. This section is then followed by a presentation of our data set, including an overview of the design considerations, c.f. Section 3, which is followed by a presentation of our proposed protocol for evaluation, c.f. Section 4. This leads to the result of our benchmark evaluation in Sections 5. The paper is rounded off by a discussion and conclusions in Section 6.

2 The NRS*f*M Problem

In this section, we will provide a brief introduction of the NRS*f*M problem, followed by a more detailed overview of the ways this problem has been addressed. The intention is to establish a taxonomy to base our experimental design and evaluation upon. In particular, we review sparse NRSfM methods as these approaches are the one evaluated in our benchmark.

The standard/rigid SfM problem, c.f. e.g. [38], is an inverse problem aimed at finding the camera positions (and possibly internal parameters) as well as 3D structure – typically represented as a static 3D point set, Q – from a sequence of 2D images of a rigid body. The 2D images are typically reduced to a sparse set of tracked 2D point features, corresponding to the 3D point set, Q. The most often employed observation model, linking 2D image points to 3D points and camera motion is either the *perspective camera model*, or the *weak perspective* approximation hereof. The weak perspective camera model is derived from the full perspective model, by simplifying the projective effect of 3D point depth, i.e. the distance between the camera and 3D point.

The extension from rigid structure from motion to the non-rigid case is by allowing the 3D structure, here points $\mathbf{Q}_f$, to vary from frame to frame, i.e.

$$\mathbf{Q}_f = \begin{bmatrix} \mathbf{Q}_{f,1} & \mathbf{Q}_{f,2} & \cdots & \mathbf{Q}_{f,P} \end{bmatrix} \quad , \tag{1}$$

Where $\mathbf{Q}_{f,p}$ is the 3D position of point p at frame f. To make this NRS*f*M problem well-defined, a prior or regularization is often employed. Here most of the cases target the spatial and temporal variations of $\mathbf{Q}_f$. The fitness of the prior to deformation in question is a crucial element in successfully solving the NRS*f*M problem, and a main difference among NRS*f*M methods is this prior.

In this study, we denote NRS*f*M methods according to a three category taxonomy, i.e. the **deformable model** used (statistical or physical), the **camera model** (affine, weak or full perspective) and the ability to deal with **missing data**. The remainder of this section will elaborate this taxonomy by relating it with the current literature, leading up to a discussion of how the NRS*f*M methods we evaluate, c.f. Table 1, span the state of the art.

2.1 Deformable Models

The description of our taxonomy will start with the underlying structure deformation model category, divided into statistical and physical based models.

2.1.1 Statistical

This set of algorithms apply a statistical deformation model with no direct connection to the physical process of structure deformations. They are in general heuristically defined a priori to enforce constraints that can reduce the ill-posedness of the NRS*f*M problem. The most used low-rank model in the NRS*f*M literature falls into this category, utilizing the assumption that 3D deformations are well described by linear subspaces (also called basis shapes). The low-rank model was first introduced almost 20 years ago by Bregler et al. [13] solving NRS*f*M through the formalisation of a factorization problem, as analogously proposed by Tomasi and Kanade for the rigid case [65]. However, strong nonlinear deformations, such as the one appearing in articulated shapes, may drastically reduce the effectiveness of such models. Moreover, the first low-rank model presented in [13] acted mainly as a constraint over the spatial distribution of the deforming point cloud and it did not restrict the temporal variations of the deforming object.

Differently, Gotardo and Martinez. [31] had the intuition to use the very same DCT bases to model camera and deformation motion instead, assuming those

factors are smooth in a video sequence. This approach was later expanded on by explicitly modeling a set of complementary rank-3 spaces, and to constrain the magnitude of deformations in the basis shapes [33]. An extension of this framework, increased the generalization of the model to non-linear deformations, with a kernel transformation on the 3D shape space using radial basis functions [32]. This switch of perspective addressed the main issue of increasing the number of available DCT bases, allowing more diverse motions, while not restricting the complexity of deformations. Later, further extension and optimization have been made to low-rank and DCT based approaches. Valmadre and Lucey [70] noticed that the trajectory should be a low-frequency signal, thus laying the ground for an automatic selection of DCT basis rank via penalizing the trajectory's response to one or more high-pass filters. Moreover, spatio-temporal constraints have been imposed both for temporal and spatial deformations [8].

A related idea proposed by Li et al. [45] attempts at grouping recurrent deformations in order to better describe deformations. At its core, the method has an additional clustering step that links together similar deformations. Recently a new prior model, related to the Kronecker-Markov structure of the covariance of time-varying 3D point, very well generalizes several priors introduced previously [62]. Another recent improvement is given by Ansari et al.'s usage of DCT basis in conjunction with singular value thresholding for camera pose estimation [19].

Similar spatial and temporal priors have been introduced as regularization terms while optimizing a cost function solving for the NRS*f*M problem, mainly using a low-rank model only. Torresani et al. [67] proposed a probabilistic PCA model for modelling deformations by marginalizing some of the variables, assuming Gaussian distributions for both noise and deformations. Moreover, in the same framework, a linear dynamical model was used to represent the deformation at the current frame as a linear function of the previous. Brand [11] penalizes deformations over the mean shape of the object by introducing sensible parameters over the degree of flexibility of the shape. Del Bue et al. [22] instead compute a more robust non-rigid factorization, using a 3D mean shape as a prior for NRS*f*M [20]. In a non-linear optimization framework, Olsen et al. [50] include l_2 penalties both on the frame-by-frame deformations and on the closeness of the reconstructed points in 3D given their 2D projections. Of course, penalty costs introduce a new set of hyper-parameters that weights the terms, implying the need for further tuning, that can be impracticable when cross-validation is not an option. Regularization has also been introduced in formulations of Bundle Adjustment for NRS*f*M [3] by including smoothness deformations via l_2 penalties mainly [25] or constraints over the rigidity of pre-segmented points in the measurement [24].

Another important statistical principal is enforcing that low-rank bases are independent. In the coarse to fine approach of Bartoli et al. [9], base shapes are computed sequentially by adding the basis, which explains most of the variance in respect to the previous ones. They also impose a stopping criteria, thus, achieving the automatic computation of the overall number of bases. The concept of basis independence clearly calls for a statistical model close to Independent Component Analysis (ICA). To this end, Brandt et al. [12] proposed a prior term to minimize the mutual information of each basis in the NRS*f*M model. Low-rank models are indeed compact but limited in the expressiveness of complex deformations, as noted in [82]. To solve this problem, Zhu et al. [82] use a temporal union of subspace that associate at each cluster of frames in time a specific subspace. Such association is solved by adopting a cost function promoting self-expressiveness [28]. Similarly, both spatial and temporal union of subspaces was used also to account for independently deforming multiple shapes [4,42]. Interestingly, such union of subspaces strategy was previously adopted to solve for the multi-body 3D reconstruction of independently moving objects [81]. Another option is to use an over-complete representation of subspaces that can still be used by imposing sparsity over the selected bases [40]. In this way, 3D shapes in time can have a compact representation, and they can be theoretically characterized as a block sparse dictionary learning problem. In a similar spirit, Hamsici et al. propose to use the input data for learning spatially smooth shape weights using rotation invariant kernels [36].

All these approaches for addressing NRS*f*M with a low-rank model have provided several non-linear optimization procedures, mainly using Alternating Least Squares (ALS), Lagrange Multipliers and alternating direction method of multipliers (ADMM). Torresani et al. first proposed to alternate between the solution of camera matrices, deformation parameters and basis shapes. This first initial solution was then extended by Wang et al. [75] by constraining the camera matrices to be orthonormal at each iteration, while Paladini et al. [54] strictly enforced the matrix manifold of the camera matrices to increase the chances to converge to the global optimum of the cost function. All these methods were not designed to be strictly convergent, for this reason, a Bilinear Augmented Multiplier Method (BALM) [26] was introduced to be convergent while implying all the problems constraints being satisfied. Furthermore, ro-

bustness in terms of outlying data was then included to improve results in a proximal method with theoretical guarantees of convergence to a stationary point [77].

Despite the non-linearity of the problem, it is possible to relax the rank constraint with the trace norm and solve the problem with convex programming. Following this strategy, Dai et al. provided one of the first effective closed form solutions to the low-rank problem [18]. Although their convex solution, resulting from relaxation, did not provide the best performance, a following iterative optimization scheme gave improved results. In this respect, Kumar et al. proposed a further improvement on their previous approach, where deformations are represented as a spatio-temporal union of subspaces rather than a single subspace [42]. Thus complex deformation can be represented as the union of several simple ones as already described in the previous paragraphs. To notice that evaluation is performed with synthetic generated data only.

Later Kumar [41] proposed a set of improvements over Dai et al. approach [18]. Namely, metric rectification was performed using incomplete information by choosing arbitrarily a triplet of solutions among the one available. The solution in [41] proposes a method to select the best among the available triplets using a rotation smoothness heuristic as a decision criteria. Then, a further improvement is algorithmic. Instead of using Dai et al. strategy with a matrix shrinkage operator that equally penalizes all the singular values, the method in [41] introduces a weighted nuclear norm function during optimisation. More recently Ornhag et al. [51] proposed a unified optimization framework for low-rank inducing penalties that can be readily applied to solve for NRS*f*M. The main advantage of the approach is the ability to combining bias reduction in the estimation and nonconvex low-rank inducing objectives in the form of a weighted nuclear norm.

On the one hand, the Procrustean Normal Distribution (PND) model was proposed as an effective way to implicitly separate rigid and non-rigid deformations [44,56]. This separation provides a relevant regularization, since rigid motion can be used to obtain a more robust camera estimation, while deformations are still sampled as a normal distribution as done similarly previously [67]. Such a separation is obtained by enforcing an alignment between the reconstructed 3D shapes at every frame. This should in practice factor out the rigid transformations from the statistical distribution of deformations. The PND model has been then extended to deal with more complex deformations and longer sequences [17].

2.1.2 Physical

Physical models represent a less studied class wrt. NRSfM, which should ideally be the most accurate for modelling NRS*f*M. Of course, applying the right physical model requires a knowledge of the deformation type and object material, which is information not readily available a priori.

A first class of physical models assume that the non-rigid object is a piecewise partition into parts, i.e. a collection of pre-defined or estimated patches that are mostly rigid or slightly deformable. This observation is certainly true for objects with articulated deformations, as it naturally models natural and mechanical shapes connected into parts. One of the first approaches to use this strategy is given by Varol et al. [71]. By preselecting a set of overlapping patches from the 2D image points, and assuming each patch is rigid, homography constraints can be imposed at each patch, followed by global 3D consistency being enforced using the overlapping points. However, the rigidity of a patch, even if small, is a very hard constraint to impose and it does not generalise well for every non-rigid shape. Moreover, dense point-matches over the image sequence are required to ensure a set of overlapping points among all the patches. A relaxation to the piece-wise rigid constraint was given by Fayad et al. [29], assuming each patch deforming with a quadratic physical model, thus, accounting for linear and bending deformations. These methods all require an initial patch segmentation and the number of overlapping points, to this end, Russel et al. [58] optimize the number of patches and overlap by defining an energy based cost function. This approach was further extended and generalised to deal with general videos [59] and energy functional that includes temporal smoothing [30]. The method of Lee et al. [43] instead use 3D reconstructions of multiple combinations of patches and define a 3D consensus between a set of patches. This approach provides a fast way to bypass the segmentation problem and robust mechanism to prune out wrong local 3D reconstructions. The method was further improved to account for higher degrees of missing data in the chosen patches so to generalise better the capabilities of the approach in challenging NRS*f*M sequences [14].

Differently from these approaches, Taylor et al. [64] constructs a triangular mesh, connecting all the points, and considering each triangle as being locally rigid. Global consistency is here imposed to ensure that the vertexes of each triangle coincide in 3D. Again, this approach is to a certain extent similar to [71], which requires a dense set of points in order to comply with the local rigidity constraint.

A strong prior, which helps dramatically to mitigate the ill-posedness of the problem, is obtained by considering the deformation isometric, i.e. the metric length of curves does not change when the shape is subject to deformations (e.g. paper and metallic materials to some extent). A first solution considering a regularly sampled surface mesh model was presented in [60]. Using an assumption that a surface can be approximated as infinitesimally planar, Chhatkuli et al. [15] proposed a local method that frame NRS*f*M as the solution of Partial Differential Equations (PDE) being able to deal with missing data as well. As a further update [55] formalizes the framework in the context of Riemannian geometry, which led to a practical method for solving the problem in linear time and scaling for a relevant number of views and points. Furthermore, a convex formulation for NRS*f*M with inextensible deformation constraints was implemented using Second-Order Cone Programming (SOCP), leading to a closed form solution to the problem [16]. Vincente and Agapito implemented soft inextensibility constraints [73] in an energy minimization framework, e.g. using recently introduced techniques for discrete optimization.

Another set of approaches try to directly estimate the deformation function using high order models. Del Bue and Bartoli [21] extended and applied 3D warps such as the thin plate spline, to the NRS*f*M problem. Starting from an approximate mean 3D reconstruction, the warping function can be constructed and the deformation at each frame can be solved by iterating between camera and 3D warp field estimation. Finally, Agudo et al. introduced the use of Finite Elements Models (FEM) in NRS*f*M [6]. As these models are highly parametrized, requiring the knowledge of the material properties of the object (e.g. the Young modulus), FEM needs to be approximated in order to be efficiently estimated, however, in ideal conditions it might achieve remarkable results, since FEM is a consolidated technique for modelling structural deformations. Lately, Agudo and Moreno-Nouger presented a duality between standard statistical rank-constrained model and a new proposed force model inspired from the Hooke's law [5]. However, in principle, their physical model can account for a wider range of deformations than rank-based statistical approaches.

2.2 Missing Data

The initial methods for NRS*f*M assumed complete 2D point matches among views when observing a deformable object. However, given self and standard occlusions, this is rarely the case. Most approaches for dealing with such missing data in NRS*f*M were framed as a matrix completion problem, i.e. estimate the missing entries of the matrix storing the 2D coordinates obtained by projecting each deforming 3D point.

Torresani et al. [68] first proposed removing rows and lines of the matrix corresponding to missing entries in order to solve the NRS*f*M problem. However, this strategy suffers greatly from even small percentages of missing data, since the subset of completely known entries can be very small. Most of the iterative approaches indeed include an update step of the missing entries [54, 26] where the missing entries become an explicit unknown to estimate. Gotardo et al. [31] instead strongly reduce the number of parameters by estimating only the camera matrix explicitly under severe missing data. This variable reduction is known as VARPRO in the optimization literature. It has been recently revisited in relation to several structure from motion problems [39].

2.3 Camera Model

Most NRS*f*M methods in the literature assume a weak perspective camera model. However, in cases where the object is close to the camera and undergoing strong changes in depth, time-varying perspective distortions can significantly affect the measured 2D trajectories.

As low-rank NRS*f*M is treated as a factorization problem, a straightforward extension is to follow best practices from rigid SfM for perspective camera. Xiao and Kanade [80] have developed a two step factorization algorithm for reconstruction of 3D deformable shapes under the full perspective camera model. This is done using the assumption that a set of basis shapes are known to be independent. Vidal and Abretske [74] have also proposed an algebraic solution to the non-rigid factorization problem. Their approach is, however, limited to the case of an object being modelled with two independent basis shapes and viewed in five different images. Wang et al. [76] proposed a method able to deal with the perspective camera model, but under the assumption that its internal calibration is already known. They update the solutions from a weak perspective to a full perspective projection by refining the projective depths recursively, and then refine all the parameters in a final optimization stage. Finally, Hartley and Vidal [37] have proposed a new closed form linear solution for the perspective camera case. This algorithm requires the initial estimation of a multifocal tensor, which the authors report is very sensitive to noise. Llado et al. [46, 47] proposed a non-linear optimization procedure. It is based on the fact that it is possible to detect nearly rigid points in the deforming shape, which can provide the basis for a robust camera calibration.

Table 1 Methods included in our NRS*f*M evaluation with annotations of how they fit into our taxonomy.

Method	Citation	Deformable Model	Camera Model	Missing Data
BALM	[26]	Statistical	Orthographic	Yes
Bundle	[25]	Statistical	Weak Perspective	Yes
Compressible	[40]	Statistical	Weak Perspective	-
Consensus	[43]	Physical	Orthographic	-
CSF	[31]	Statistical	Weak Perspective	Yes
CSF2	[33]	Statistical	Orthographic	Yes
EM PPCA	[67]	Statistical	Weak Perspective	Yes
KSTA	[32]	Statistical	Orthographic	Yes
MDH	[16]	Physical	Perspective	Yes
MetricProj	[54]	Statistical	Orthographic	Yes
MultiBody	[42]	Statistical	Orthographic	-
PTA	[7]	Statistical	Orthographic	-
RIKS	[36]	Statistical	Orthographic	-
ScalableSurface	[19]	Statistical	Orthographic	Yes
SoftInext	[73]	Physical	Perspective	Yes
SPFM	[18]	Statistical	Orthographic	-
CMDR	[30]	Physical	Orthographic	-
F-consensus	[14]	Physical	Orthographic	yes

2.4 Evaluated Methods

We have chosen a representative subset of the aforementioned methods, which are summarized according to our taxonomy in Table 1. This gives us a good representation of recent works, distributed according to our taxonomy with a decent span of deformation models (statistical/physical) and camera models (orthographic, weak perspective or perspective). This also takes into account in-group variations such as DCT basis for statistical deformation and isometry for physical deformation. Even lesser used priors, such as compressibility, are represented. While this is not a full factorial study, we think this reasonably spans the recent state of the art of NRS*f*M. Our choice has, of course, also been influence by method availability, as we want to test the author's original implementation, to avoid our own implementation bias/errors. All in all, we have included 18 methods in our evaluation. Note that we have chosen not to include the method of Taylor et al. [64], even if code is available, the approach failed approximately two thirds of the time when tested on our data set.

3 Dataset

As stated, in order to compare state of the art methods for NRS*f*M, we have compiled a larger data set for this purpose. Even though there is a lack of empirical evidence w.r.t. NRS*f*M, it does not imply, that no data sets for NRS*f*M exist.

As an example in [43], [31], [33], [32], [42], [7], [36] and [18], a combination of two data sets are used. Namely seven sequences of a human body from the CMU motion capture database [69], two MoCap sequences of a deforming face [66,23], a computer animated shark [66] and a challenging flag sequence [29]. To the best of our knowledge, this list in Table 2 represents the most used evaluation data sets for NRS*f*M with available ground truth.

The CMU data set [69] captures the motion of humans. Since the other frequently used data sets are also related to animated faces [66,23], this implies that there is a high over representation of humans in this state of the art and that a higher variability in the deformed scenes viewed is deemed beneficial. In addition, the shark sequence [66] is not based on real images and objects but on computer graphics and pure simulation. As such, there is a need for new data sets, with reliable ground truth or reference data,[1] and a higher variability in the objects and deformations used.

As such, we here present a data set consisting of five widely different objects/scenes and deformations. The physical object motions are generated mechanically using animatronics, therefore assuring experimental repeatability. Furthermore, we have defined six different camera motions using orthographic and full perspective camera models. This setup, all in all, gives 60 different sequences organized in a factorial experimental design, thus, enabling a more stringent statistical analysis. In addition to this, since we have tight 3D surface models of our objects or scenes, we are able to determine occlusions of all 2D feature points. This in turn gives a realistic handling of missing data, which is often due to object self occlusion. Given this procedure of generating occlusions, missing data always follow a more realistic structured pattern in contrast with the most common,

[1] With real measurements like ours the 'ground truth' data also include noise, why 'reference data' is a more correct term.

Table 2 A description of the previous data set sequences with available ground truth. The table shows the number of frames and points, the way to generate the sequence (mainly with motion capture data) and the type of shape used.

Name	Citation	Frames×Points	Type	Shape
shark	[67]	240 × 91	Synthetic	Animal motion
face1	[67]	74 × 37	Mocap	Face motion
face2	[67]	316 × 40	Mocap	Face motion
cubes	[79]	200 × 14	Synthetic	ToyProblem
face_occ	[54]	70 × 37	Mocap	Face motion
flag	[29]	540 × 50	Mocap	Cloth deformation
yoga	[7]	307 × 41	Mocap	Human motion
drink	[7]	1102 × 41	Mocap	Human motion
stretch	[7]	307 × 41	Mocap	Human motion
dance	[7]	264 × 41	Mocap	Human motion
pickup	[7]	357 × 41	Mocap	Human motion
walking	[7]	260 × 41	Mocap	Human motion
capoeira	[31]	250 × 41	Mocap	Human motion
jaws	[31]	321 × 49	Synthetic	Animal motion

and unrealistic, random process of removing 2D measurement entries used in previous evaluation dataset.

As indicated, these data sets are achieved by stop-motion using mechanical animatronics. These are recorded in our robotic setup previously used for generating high quality data sets c.f. e.g. [2]. We will here present details of our data capture pipeline, followed by a brief outline and discussion of design considerations.

The goal of the data capturing is to produce 3 types of related data:

Ground Truth: A series of 3D points that change over time.
Input Tracks: 2D tracks used as input.
Missing Data: Binary data indicating the tracks that are occluded at specific image frames.

We record the step-wise deformation of our animatronics from K static views, obtaining both image data and dense 3D surface geometry. We obtain 2D point features by applying standard optical flow tracking [10] to the image sequence obtained from each of the K views, which is then reprojected onto the recorded surface geometry. The ground truth is then the union of these 3D tracks. By using optical flow for tracking instead of MoCap markers, we obtain a more realistic set of ground truth points. We create input 2D points by projecting the recorded ground truth using a virtual camera in a fully factorial design of camera paths and camera models.

In the following, we will detail some of the central parts of the above procedure.

3.1 Animatronics & Recording Setup

Our stop-motion animatronics are five mechatronic devices capable of computer controlled gradual deformation. They are shown in Fig. 2, and they cover five types of deformations: Articulated Motion, Bending, Deflation, Stretching, and Tearing. We believe this covers a good range of interesting and archetypal deformations. It is noted, that NRS*f*M has previously been tested on bending and tearing [64, 73, 16, 43], but without ground truth for quantitative comparison. Additionally, elastic deformations, like deflation and stretching, are quite commonplace but did not appear in any previous data sets, to the best of our knowledge.

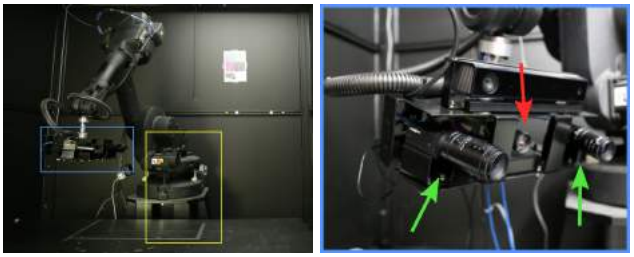
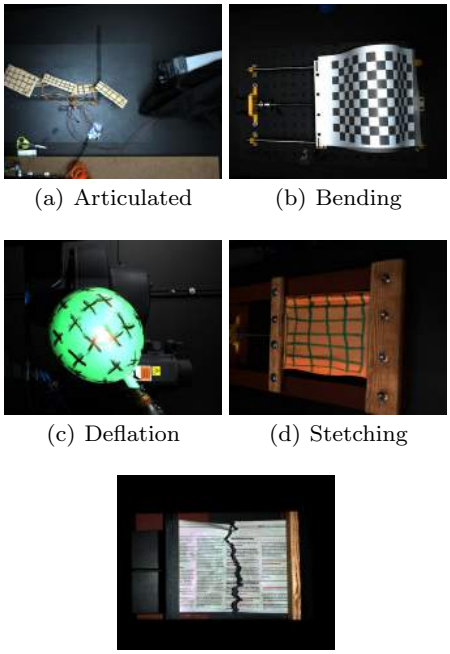


Fig. 1 Images of the robot cell for dataset acquisition. Left image shows the robot with the structured light scanner (blue box) and the area where the animatronic systems are positioned (yellow box). Right image shows the structured light scanner up close, green arrows show the position of the Point-Grey Grasshopper3 cameras, and the red arrow marks the Lightcrafter 4500 projector.

The animatronics can hold a given deformation or pose for a large extent of time, thus, allowing us to record accurately the object's geometry. We, therefore, do not need a real-time 3D scanner or elaborate multi-scanner setup. Instead, our recording setup consists of an in-house built structured light scanner mounted on an industrial robot as shown in Fig. 1. This does not only provide us with accurate 3D scan data, but the robot's mobility also enables a full scan of the object at each deformation step.



(a) Articulated (b) Bending

(c) Deflation (d) Stetching

(e) Tearing

Fig. 2 Animatronic systems used for generating specific types of non-rigid motion.

The structured light scanner utilizes two PointGrey Grasshopper3 9.1MP CCD cameras and a projector WinTech Lightcrafter 4500 Pro projecting patterns onto the scene and acquiring images. Then, we use the Heterodyne Phase Shifting method [57] to compute the point clouds using 16 periods across the image and 9 shifts. We verified precision according to standard VDI 2634-2 [27], and found that the scanner has a form error of [0.01mm, 0.32mm], a sphere distance error of [-0.33mm 0.50mm] and a flatness error of [0.29mm, 0.56mm]. This is approximately 2 orders of magnitude better than the results we see in our evaluation of the NRS*f*M methods.

3.2 Recording Procedure

The recording procedure acquires for each shape a series of image sequences and surface geometries of its deformation over F frames. We record each frame from K static views with our aforementioned structured light scanner. As such we obtain K image sequences with F images in each. We also obtain F dense surface reconstructions, one for each frame in the deformation. The procedure is summarized in pseudo code in Algorithm 1. Fig. 3 illustrates sample images of three views obtained using the above process.

Algorithm 1: Process for recording image data for tracking and dense surface geometry for an animatronic.

```
Let F be the number of frames
Let k be the number of static scan views K
for f ∈ F do
    Deform animatronic to pose f
    for k ∈ K do
        Move scanner to view k
        Acquire image I_{f,k}
        Acquire structured light scan S_{f,k}
    end
    Combine scans S_{f,k} for full, dense surface S_f
end
```

3.3 3D Ground Truth Data

The next step is to take acquired images $I_{f,k}$ and surfaces S_f, and extract the ground truth points. We do this by applying optical flow tracking [10] as implemented in OpenCV 2.4 to obtain 2D tracks, which are then reprojected onto S_f. The union of these reprojected tracks gives us the ground truth, Q. This process is summarized in pseudo code in Algorithm 2.

Algorithm 2: Process for extracting the ground truth Q from recorded images and surface scans.

```
Let F be the number of frames
Let k be the number of static scan views K
Let S_f be the surface at frame f
Let I_{f,k} be the image from view k, frame f
S = {S_1 ... S_F}
for k ∈ K do
    I_k = {I_{1,k} ... I_{F,k}}
    Apply optical flow [10] to I_k to get 2D tracks T_k
    Reproject T_k onto S to get 3D tracks Q_k
end
Q = {Q_1 ... Q_K}
```

3.4 Projection using a Virtual Camera

To produce the desired input, we project the ground truth $\mathbf{Q}$ using a virtual camera, similar to what has been done in [43,31,18,23]. This step has two factors related to the camera that we wish to control for: Path and camera model. To keep our design factorial, we define six different camera paths, which will all be used to create the 2D input. They are illustrated in Fig. 4. We believe these are a good representation of possible camera motion with both linear motion and panoramic panning. The Circle and Half Circle paths correspond well

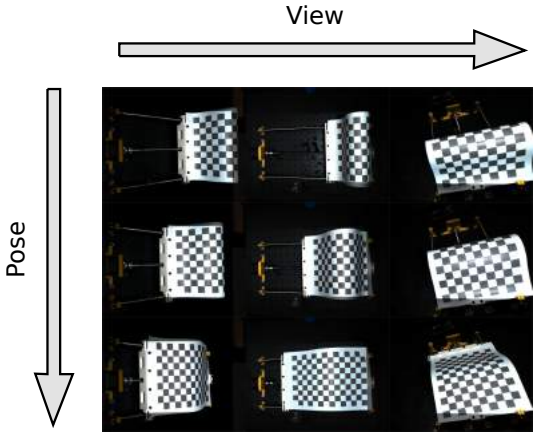


Fig. 3 Illustrative sample of our multi-view, stop-motion recording procedure. Animatronic pose evolves vertically and scanner view change horizontally.

to the way scans are performed in SfM and structured light methods: By moving around the target object we try to cover most of its shape. Line and Flyby are to simulate a scenario where instead the camera move linearly as in the automotive and drone-alike movements respectively. Zigzag and Tricky motions are about having depth variations in the camera movement, which is important for perspective camera, where each frame will have different projective distortions. Tricky camera path resembles more a critical motion in the direction of the optical ray of the camera as expected, for instance, in medical imaging. To conclude, as mentioned earlier, the camera model can be either orthographic or perspective.

The factorial combination of these elements yields to 12 input sequences for each ground truth. Additionally, as we have previously recorded the dense surface for each frame (see Sec. 3.2), we estimate missing data via self-occlusion. Specifically, we create a triangular mesh for each S_f and estimate occlusion via raycasting into the camera along the projection lines. Vertices whose ray intersects a triangle on the way to the camera are removed, from the input for the given frame, as those vertices would naturally be occluded. In this way, we ensure as realistic as possible structured missing data by modelling self-occlusion given the different camera paths. This process is summarized in pseudo code in Algorithm 3.

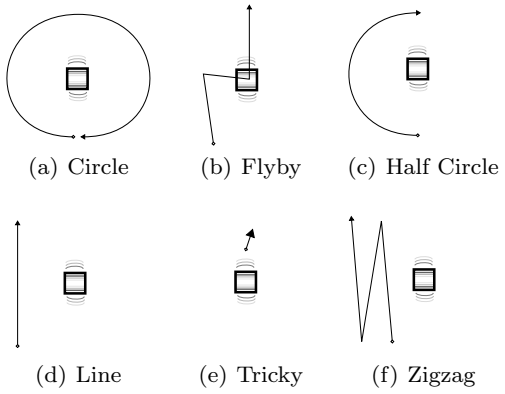


Fig. 4 Camera path taxonomy. The box represents the deforming scene and the wiggles illustrates the main direction of deformation, e.g. the direction of stretching.

Algorithm 3: Creation of input tracks $W_{c,p}$ and missing data $D_{c,p}$ from ground truth Q for each combination of camera path p and model c.

```
Let F be the number of frames
Let P be the set of camera paths shown in Fig. 4
Let C be either perspective or orthographic
Let Q_f be the ground truth at frame f
Let S_f be the surface at frame f
for S_f ∈ {S_1 ... S_F} do
    Estimate mesh M_f from S_f
end
for c ∈ C do
    for p ∈ P do
        for f ∈ F do
            Set camera pose to p_f
            Project Q_f using model c to get points
              w_f
            Do occlusion test q_f against M_f to get
              missing data d_f
        end
        W_{c,p} = {w_1 ... w_F}
        D_{c,p} = {d_1 ... d_F}
    end
end
```

3.5 Discussion

While stop-motion does allow for diverse data creation, it is not without drawbacks. Natural acceleration is easily lost when objects deform in a step-wise manner and recordings are unnaturally free of noise like motion blur. However, without this technique, it would have been prohibitive to create data with the desired diversity and accurate 3D ground truth.

The same criticism could be levied against the use of a virtual camera, it lacks the shakiness and acceleration of a real world camera. On the other hand, it allows us to precisely vary both the camera path and camera model. This enables us to perform a factorial analysis, in which we can study the effects of different configurations on NRS*f*M. As we show in Sec. 5 some interesting conclusions are drawn from this analysis. Most NRS*f*M methods are designed with an orthographic camera in

mind. As such investigating the difference between data under orthographic and perspective projection is of interest. Such an investigation is only practically possible using a virtual camera.

4 Evaluation Metric

In order to compare the methods of Table 1 w.r.t. our data set, a metric is needed. The purpose is to project the high dimensional 3D reconstruction error into (ideally) a one dimensional measure.

Several different metrics have been proposed for NRS*f*M evaluation in the past literature, e.g. the Frobenius norm [53] mean [36], variance normalized mean [33] and RMSE [64].

All of the above mentioned evaluation metrics are based on the $L2$-norm in one form or another. A drawback of the $L2$-norm is its sensitive to large errors, often letting a few outliers dominating the evaluation. To address this, we incorporate robustness into our metric, by introducing truncation of the individual 3D point reconstruction errors. In particular, our metric is based on a RMSE measure similar used in Taylor et al. [64].

Given the visualisation effectiveness and general adoption of box plots [72], we propose to use their whisker function to identify and to model outliers in the error distribution. Such a strategy will enable the inclusion of outliers in the metric with the additional benefit of reducing their influence in the RMSE. Consider E being the set of point-wise errors ($||\mathbf{X}_{f,p} - \mathbf{Q}_{f,p}||$) and E_1, E_3 as the first and third quartile of that set. As described in [78], we define the whisker as $w = \frac{3}{2}(E_3 - E_1)$, then any point that is more than a whisker outside of the interquantile range ($IQR = E_3$ - E_1) is considered as an outlier. Those outliers are then truncated at $E_3 + w$ allowing them to be included in a RMSE without dominating the result. This strategy works well for approximately normally distributed data. With this in mind, our truncation function is defined as follows,

$$t\left(\mathbf{x}, \mathbf{q}\right) = \begin{cases} ||\mathbf{x} - \mathbf{q}||, & ||\mathbf{x} - \mathbf{q}|| < E_3 + w \\ E_3 + w, & \text{otherwise} \end{cases} \tag{2}$$

Thus the robust RMSE is defined as,

$$m\left(\mathbf{Q}, \mathbf{X}\right) = \sqrt{\frac{1}{FP} \sum_{f,p}^{F,P} t\left(\mathbf{X}_{f,p}, \mathbf{Q}_{f,p}\right)}. \tag{3}$$

A NRS*f*M reconstruction is given in an arbitrary coordinate system, thus we must align the reference and reconstruction before computing the error metric. This is typically done via Procrustes Analysis [34], but as it minimizes the distance between two shapes in a $L2$-norm sense it is also sensitive to outliers. Therefore, we formulate our alignment process as an optimization problem based on the robust metric of Eq. 3. Thus the combined metric and alignment is given by,

$$m(\mathbf{X}, \mathbf{Q}) = \min_{s,\mathbf{R},\mathbf{t}} \sqrt{\frac{1}{FP} \sum_{f,p} t\left(s\left[\mathbf{R}\mathbf{X}_{fp} + \mathbf{t}\right], \mathbf{Q}_{fp}\right)}, \tag{4}$$

where s = scale,
$\mathbf{R}$ = rotation and reflection,
$\mathbf{t}$ = translation.

An implication of using a robust, as opposed to a $L2$-norm, is that the minimization problem of (4) cannot be achieved by a standard Procrustes alignment, as done in [64]. As such, we optimize (4) using the Levenberg-Marquardt method, where s, $\mathbf{R}$ and $\mathbf{t}$ have been initialized via Procrustes alignment [35]. In summary, (4) defines the alignment and metric that has been used for the evaluation presented in Sec. 5.

Notice also that this registration procedure estimates a single rotation and translation for the entire sequence. In this way, we avoid the practise of registering the GT 3D shape at every frame of the reconstructed 3D sequence. Such frame-by-frame procedure does not account for the global temporal consistency of the reconstructed 3D sequence and in particular regarding possible sign flips of the 3D shape, scale variations, or reflections that might happen abruptly from one frame to the other during reconstruction. Registering the 3D ground truth frame-by-frame is also unrealistic, because in general, it is not feasible to do in a real operative reconstruction scenario where 3D GT is not available.

To conclude, the choice of an evaluation metric always has a streak of subjectivity and for this reason, we investigated the sensitivity of choosing a particular one. We did this by repeating our evaluation with another robust metric, where the minimum track-wise distance between the ground truth and reconstruction was used. By just using the n-th percentile, instead of our truncation, the magnitude of the RMSE significantly decreases, but the major findings and conclusions, as presented in Sec. 5, were the same. As such we conclude that our conclusions are not overly sensitive to the choice of metric.

5 Evaluation

With our data set and robust error metric, we have performed a thorough evaluation and analysis of the state-of-the-art in NRS*f*M, which is presented in the following. This is done in part as an explorative analysis and in part to answer some of what we see as most pressing, open questions in NRS*f*M. Specifically:

- Which algorithms perform the best?
- Which deformable models have the best performance or generalization?
- How well can the state-of-the-art handle data from a perspective camera?
- How well can the state-of-the-art handle occlusion-based missing data?

To answer these questions, we perform our analysis in a factorial manner, aligned with the factorial design of our data set. To do this, we view a NRS*f*M reconstruction as a function of the following factors:

Algorithm a_i:	Which algorithm was used.
Camera Model m_j:	Which camera model was used (perspective or orthographic).
Animatronics s_k:	Which animatronics sequence was reconstructed.
Camera Path p_l:	How the camera moved.
Missing Data d_n:	Whether occlusion based missing data was used.

We design our evaluation to be almost fully crossed, meaning we obtain a reconstruction for every combination of the above factors.

The only missing part is that the authors of MultiBody [42] only submitted reconstructions for orthographic camera model.

Our factorial experimental design allows us to employ a classic statistical method known as ANalysis Of VAriance (ANOVA) [61]. The ANOVA not only allow us to deduce the precise influence of each factor on the reconstruction but also allows for testing their significance. To be specific, we model the reconstruction error in terms of the following bilinear model,

$$\begin{aligned} y = \mu + a_i + m_j + s_k + p_l + d_n \\ + as_{ik} + ap_{il} + ad_{in} + ms_{jk} \\ + mp_{jl} + md_{jn} + sp_{kl} + sd_{kn} + pd_{ln}, \end{aligned} \tag{5}$$

where,

y = reconstruction error,

μ = overall average error,

$xy_{i,j}$ = interaction term between factor x_i and y_j.

This model, Eq. (5), contains both linear and interaction terms, meaning the model reflects both factor influence as independent and as cross effects, e.g. as_{ik} is the interaction term for 'algorithm' and 'animatronics'. For each term, we test for significance by choosing between two hypotheses:

$$\begin{aligned} &\mathcal{H}_0 : c_0 = c_1 = \ldots = c_N \\ &\mathcal{H}_1 : c_0 \neq c_1 \neq \ldots \neq c_N \end{aligned} \tag{6}$$

Table 3 ANOVA table for NRS*f*M reconstruction error without missing data with sources as defined in (5). All factors are statistically significant at a 0.0005 level except ms_{jk} and mp_{jl}.

Factor	Sum Sq.	DoF	Mean Sq.	F	p-value
a_i	3.6×10^5	15	2.4×10^4	204.8	5.5×10^{-242}
m_j	1.1×10^4	1	1.1×10^4	90.4	3.2×10^{-20}
s_k	1.0×10^5	4	2.6×10^4	219.0	3.6×10^{-121}
p_l	1.5×10^4	5	3.0×10^3	25.6	9.3×10^{-24}
as_{ik}	4.1×10^4	60	6.9×10^2	5.9	2.9×10^{-33}
ap_{il}	4.1×10^4	75	5.5×10^2	4.7	2.3×10^{-28}
ms_{jk}	1.3×10^3	4	3.2×10^2	2.7	0.03
mp_{jl}	1.8×10^3	5	3.6×10^2	3.1	0.0086
sp_{kl}	1.1×10^4	20	5.7×10^2	4.9	2.3×10^{-11}
Error	8×10^4	689	1.2×10^2		
Total	7×10^5	878			

with c_n being a term from (5) e.g. a_i or md_{jn}. Typically, $\mathcal{H}_0$ is referred to as the null hypothesis, meaning the term c_n has no significant effect. ANOVA allows for estimating the probability of falsely rejecting the null hypothesis for each factor. This statistic is referred to as the p-value. A term is referred to as being statistically significant if its p-value is below a certain threshold. In this paper we consider a significance threshold of 0.0005 or approximately 3.5σ. As such, we clearly evaluated which factors are important for NRS*f*M and which are not.

Another interesting property of the ANOVA is that all coefficients in a given factor sums to zero,

$$\sum_{i=0}^{N} c_i = 0. \tag{7}$$

So each factor can be seen as adjusting the predicted reconstruction error from the overall average. It should be noted that the "algorithm"/"camera model" interaction am_{ij} has been left out of (5) due to MultiBody [42] only being tested with one camera model.

The error model of (5) is not directly applicable to the error of all algorithms as not all state-of-the-art methods from Table 1 can deal with missing data. As such we perform the evaluation in two parts. One where we disregard missing data and include all available methods from Table 1, and one where we use the subset of methods that handles missing data and utilize the full model of (5). The former is covered in Sec. 5.1 and the latter is covered in Sec. 5.2.

5.1 Evaluation without missing data

In the following, we discuss the results of the ANOVA without taking 'missing data' into account, using the

Table 4 Linear term $\mu + a_i$ sorted in ascending numerical order, this is the average error for the given algorithm. Algorithms are referred to by their alias in Table 1. All numbers are given in millimeters.

MultiBody 29.36	**KSTA** 31.94	**RIKS** 32.21
CSF2 32.83	**MetricProj** 34.09	**CSF** 41.19
Bundle 46.66	**PTA** 46.80	**F-Consensus** 53.17
ScalableSurface 53.88	**CMDR** 53.91	**EM PPCA** 59.21
SoftInext 61.94	**BALM** 66.34	**MDH** 70.34
Compressible 79.18	**SPFM** 85.34	**Consensus** 94.61

model as in Eq. (5) without terms related to d_n:

$$y = \mu + a_i + m_j + s_k + p_l + as_{ik} + ap_{il} + ms_{jk} + mp_{jl} + sp_{kl}. \quad (8)$$

The results of the ANOVA using Eq. (8) is summarized in Table 3. All factors except ms_{jk} and mp_{jl} are statistically significant. As such, we can conclude that all the aforementioned factors have a significant influence on the reconstruction error. Therefore, we will explore the specifics of each factor in the following, starting with 'algorithm'.

Table 4 shows the average reconstruction error for each algorithm. The method MultiBody [42] has the lowest average reconstruction error over all experiments followed by KSTA [32] and RIKS [36]. For more detailed insights refer to Table 5 showing the 'algorithm' vs 'animatronic' effect on the reconstruction error. As it can be seen, MultiBody [42] does not have the lowest error for all animatronics, as e.g. KSTA [32] has a significantly lower error on the Tearing and Articulated deformations. Both of these can roughly be described as rigid bodies moving relative to each other, and it would seem KSTA [32] is the best at handling these deformations.

Methods with a physical prior, like MDH [16] and SoftInext [73] have in general lower performance, as it is evident from Tables 1, 5 and 6. MDH [16] is designed with an isometry prior, therefore one would expect it to perform well in the bending deformation. Indeed, while its interaction term as_{ik} has its lowest value for the bending deformation, denoting the fitness of the chosen prior, the average reconstruction error is higher. On a more careful inspection of the reconstructed 3D sequences, it is evident that for a few frames MDH and SoftInext struggle to obtain an accurate 3D reconstruction and this affects the whole evaluation. Moreover, the 3D reconstruction shows intermittent sign flips of the 3D reconstructed shape. To this end, a stronger temporal consistency may help to reduce this negative effect and improve the method performance.

Table 5 Interaction term $\mu+a_i+s_k+as_{ik}$. This is equivalent to the algorithms average error on each animatronic. Lowest error for each animatronic is marked with bold text. Algorithms are referred to by their alias in Table 1. All numbers are given in millimeters.

	Deflation	Tearing	Bending	Stretching	Articulated
MultiBody	***15.20***	24.82	***25.21***	***25.12***	56.44
KSTA	27.60	***20.78***	36.66	29.62	***45.05***
RIKS	24.10	21.37	35.04	32.07	48.49
CSF2	23.55	21.55	36.21	32.33	50.51
MetricProj	27.75	25.93	35.93	33.22	47.63
CSF	34.92	40.93	40.10	39.96	50.03
Bundle	39.36	29.47	43.07	49.96	71.44
PTA	35.75	34.49	51.81	47.93	63.99
F-Consensus	34.86	48.45	50.22	57.96	74.33
ScalableSurface	34.60	47.95	53.82	59.40	73.65
CMDR	40.28	51.95	54.43	61.20	61.68
EM PPCA	40.18	59.60	65.29	73.88	57.09
SoftInext	46.60	54.07	64.05	65.49	79.48
BALM	52.51	58.28	74.85	67.76	78.29
MDH	56.87	63.75	69.00	75.02	87.06
Compressible	61.62	71.06	79.66	79.08	104.47
SPFM	54.85	76.19	80.05	89.93	125.68
Consensus	66.96	83.07	83.51	95.62	143.90

Table 6 Interaction term $\mu + a_i + p_l + ap_{il}$. Algorithms are referred to by their alias in Table 1. All numbers are given in millimeters.

	Zigzag	Line	Half Circle	Flyby	Tricky	Circle
MultiBody	***19.48***	***28.52***	30.88	***29.71***	52.18	***15.37***
KSTA	24.35	33.56	29.36	34.65	43.17	26.57
RIKS	25.68	30.24	***26.76***	37.59	***41.21***	31.81
CSF2	28.22	28.96	28.25	36.58	43.96	31.02
MetricProj	26.48	32.37	30.67	34.88	48.79	31.36
CSF	31.90	46.39	40.17	34.53	59.49	34.65
Bundle	47.30	39.27	45.55	39.68	55.30	52.84
PTA	35.51	48.34	42.67	43.91	60.53	49.82
F-Consensus	37.89	37.42	50.52	52.73	48.76	91.68
ScalableSurface	39.64	41.88	52.68	52.64	48.49	87.98
CMDR	38.95	45.89	53.35	52.91	51.90	80.46
EM PPCA	52.88	58.40	54.68	55.70	57.49	76.11
SoftInext	51.38	49.13	58.32	62.58	61.17	89.06
BALM	62.61	72.22	59.87	56.73	73.55	73.06
MDH	75.09	71.77	60.50	67.90	67.46	79.33
Compressible	73.61	80.08	80.78	83.84	84.24	72.49
SPFM	85.53	82.53	86.09	88.33	86.88	82.68
Consensus	94.70	94.81	94.52	94.35	94.42	94.88

Table 7 Linear term $\mu + m_j$ sorted in ascending numerical order, this is the average error for the given camera model. All numbers are given in millimeters.

Orthographic	**Perspective**
50.45	57.66

Table 8 Linear term $\mu + s_k$ sorted in ascending numerical order, this is the average error for the given animatronic. All numbers are given in millimeters.

Deflation	**Tearing**	**Bending**
39.86	46.32	54.38
Stretching	**Articulated**	
56.42	73.29	

Table 9 Linear term $\mu + p_l$ sorted in ascending numerical order, this is the average error for the given camera path. All numbers are given in millimeters.

Zigzag	**Line**	**Half Circle**
47.29	51.21	51.42
Flyby	**Tricky**	**Circle**
53.29	59.94	61.18

A similar trend can be observed in Table 6, which shows the 'algorithm' vs 'camera path' effect on the reconstruction error. While MultiBody [42] has the lowest average error, it is surpassed in the Half Circle and Tricky 'camera path' by RIKS [36]. On the other hand, MultiBody has the lowest error under the Circle path by quite a significant margin.

From this analysis we can conclude that MultiBody performs the best on average, but is surpassed w.r.t. to certain camera paths and animatronic deformations by algorithms such as RIKS [36] and KSTA [32]. This also clearly indicates that one needs to control for both deformation type and camera motion in future NRS*f*M comparisons, as the above conclusion could be changed by choosing the right combination of camera path and deformation. On the other hand, these findings show that NRS*f*M performance can be optimized by choosing the right camera path (e.g. Zigzag) and the right algorithm for the deformation in question.

The camera model and its path have a significant impact on reconstruction error, a trend that can be observed from Table 6. Table 9 shows that there is a significant difference in average error w.r.t. 'camera path'. It is interesting to note, that the Circle path has one of the highest average errors, only surpassed by the Tricky camera path. The latter was specifically designed to be challenging, as such, it is surprising to find that the Circle and Tricky path's average error only differ by 3.08mm. In fact, MultiBody [42] seems to be the only method that benefits from the circle type of camera path, as can be seen in Table 6. Table 7 shows the average error of reconstructions for an orthographic and a perspective camera model. As it can be seen, there is a difference of 7.20mm, which is significant but not as large as the difference w.r.t. 'algorithm' (Table 4) or 'camera path' (Table 9). This suggests that, while the error increases the state-of-the-art in NRS*f*M can still operate under a perspective camera model. This is quite interesting as most NRS*f*M approaches are not designed with a perspective camera in mind. It would seem that an orthographic or weak-perspective camera acts a reasonable approximation given the perspective distortions and the scale of the object deformation.

Table 10 ANOVA table for NRS*f*M reconstruction error with missing data. Factors are as defined in (5) and described at the beginning of this section. All factors are statistically significant at a 0.0005 level except ms_{jk}, mp_{jl} and md_{jn}.

Factor	Sum Sq.	DoF	Mean Sq.	F	p-value
a_i	1.3×10^5	8	1.6×10^4	90.9	7.7×10^{-108}
m_j	1.4×10^4	1	1.4×10^4	81.6	1.2×10^{-18}
s_k	7.5×10^4	4	1.9×10^4	106.5	3.8×10^{-73}
p_l	4.1×10^4	5	8.2×10^3	47.0	8.8×10^{-43}
d_n	1.6×10^4	1	1.6×10^4	89.8	2.7×10^{-20}
as_{ik}	1.6×10^4	32	5.0×10^2	2.9	3.4×10^{-7}
ap_{il}	5.6×10^4	40	1.4×10^3	8.0	6.4×10^{-37}
ad_{in}	1.1×10^4	8	1.3×10^3	7.5	1.1×10^{-9}
ms_{jk}	2.6×10^3	4	6.5×10^2	3.7	0.0052
mp_{jl}	2.5×10^3	5	5.1×10^2	2.9	0.013
md_{jn}	2.9×10^2	1	2.9×10^2	1.6	0.2
sp_{kl}	2.7×10^4	20	1.4×10^3	7.8	6.7×10^{-21}
sd_{kn}	3.6×10^3	4	8.9×10^2	5.1	0.00048
pd_{ln}	8.1×10^3	5	1.6×10^3	9.3	1.4×10^{-8}
Error	1.4×10^5	824	1.8×10^2		
Total	5.7×10^5	962			

There is also a significant difference between the average reconstruction error of each animatronic which Table 8 shows. Articulated has by far the highest average reconstruction error, making it the most difficult to reconstruct for the current state-of-the-art in NRS*f*M. Since most approaches use low-rank methods, a highly structured motion such as an Articulated is difficult to handle with a low-rank prior, especially if points are densely sampled on all joints. On the other hand, Deflation seems to be quite easy to handle for most of the state-of-the-art methods.

5.2 Evaluation with Missing Data

As previously mentioned, we are interested in 'missing data' and its effect on NRS*f*M. We, thus, here use Eq. (5), which is used to evaluate the subset of methods capable of handling missing data, as shown in Table 1.

It should be noted that while MDH [16] is nominally capable of handling missing data, it has not been included in this part of the study. The reason being that the code provided only reconstructs frames with minimum ratio of visible data, thus our error metric cannot

be applied. As such, we have 9 methods in total in this category.

We treat 'missing data' as a categorical factor having two states: with or without missing data. This is because the missing percentage of our occlusion-based missing data is dependent on the 'animatronic', 'camera path' and 'camera model' factors. Additionally, there is a significant sampling bias in the occlusion-based missing data. For example, in-plane motion, like Articulated and Tearing, rarely get a missing percentage above 25% and more volumetric motion such as Deflation rarely go below 40% missing data. This would make it difficult to distinguish between the influence of the 'missing data' factor and the animatronic factor.

The results of the ANOVA is summarized in Table 10 and all factors except ms_{jk}, mp_{jl} and md_{jn} are statistically significant. This means that 'missing data' has a significant influence on the reconstruction error. Table 13 shows the interaction between 'algorithm' and 'missing data'. As expected, the mean error without missing data is very similar to the averages in Table 4 with KSTA [32] having the lowest expected error. However, with missing data, MetricProj [54] actually has a lower average reconstruction error. This is due to its low increase in error of 5.85mm when operating under occlusion-based missing data. In comparison, KSTA [32], CSF2 [33] and CSF [31] are much more unstable with average increases in error of 9.65mm, 18.15mm and 13.49mm respectively. Common among the three methods is the fact that they assume a Discrete Cosine Transform (DCT) as their prior. Indeed, we see a similar increase for ScalableSurface of 16.52mm and this method also uses a DCT basis.

These results suggest that while DCT-based approaches are quite accurate without missing data, they are not very robust when operating under occlusion-based missing data. Thus, they would likely not be very robust when applied to real-world deformations, where occlusion-based missing data is unavoidable. This indicates that future research should focus on making DCT basis methods more robust or to modify the DCT model to better generalize for 'missing data'. Finally, BALM [26] method exhibit some peculiar behavior as its average error actually decreases by 3.33mm, contrary to expectation. A likely cause is a different computational structure of the algorithm, since the full data case uses mainly SVD for factorisation while the missing data approach has a more elaborated algorithmic approach with manifold projections and matrix entries imputation.

Table 12 shows the average error as an interaction between 'animatronic' and 'missing data', i.e. the average reconstruction error of each animatronic with and without missing data. It is interesting to note that the in-plane deformations, i.e. Tearing, Stretching and Articulated, generally have a smaller increase in error with missing data compared to the more volumetric deformation, i.e. Deflation and Bending, compared to the error without missing data. The increase is respectively 3.96mm, 4.65mm and 8.38mm versus 12.27mm and 13.47mm. The main difference between the two groups is that the ratio of missing data is consistently low for the in-plane deformations. This would suggest that the ratio of missing data has an impact on the reconstruction error.

Table 11 Interaction between 'camera path'/'missing data'; $\mu + p_l + d_n + pd_{ln}$. Numbers are given in milimeters.

	Without Missing	With Missing
Zigzag	***42.82***	***46.48***
Half Circle	45.59	52.41
Line	46.25	52.10
Flyby	47.22	53.47
Circle	58.96	63.39
Tricky	54.24	75.26

Table 12 Interaction between 'animatronic'/'missing data'; $\mu + s_k + d_n + sd_{kn}$. Numbers are given in milimeters.

	Without Missing	With Missing
Deflation	***36.94***	48.66
Tearing	41.53	***45.06***
Stretching	52.30	56.70
Bending	50.33	63.12
Articulated	64.79	72.39

Table 11 shows the average error as interaction between 'camera path' and 'missing data'. The Tricky path has by far the highest average error. This is expected, as the small camera movement ensures that a portion of the tracked points is consistently hidden. As such, while Tricky and Circle were almost equally difficult without missing data, this is no longer the case with missing data as Circle's average error only increases by 4.9mm. Indeed, all other camera paths have approximately the same increase in error with missing data. These paths also ensure that all observed points are equally visible. What differs consistently is the spatio-temporal distribution of missing data, which has a physical plausible structured pattern. the missing data distributions in our dataset are in contrast with previous evaluations where often missing entries were generated randomly, thus not reflecting a real 3D modelling scenario. These results also suggest that the distribution of missing data is as important as the ratio in affecting the reconstruction error. Indeed this is in line with the observations made by Paladini et al. [54].

Table 13 Interaction between 'algorithm'/'missing data'; $\mu + a_i + d_n + ad_{in}$. This is the average error for each algorithm either with or without occlusion-based missing data.

	KSTA	MetricProj	CSF2	CSF	Bundle	F-Consensus	ScalableSurface	EM PPCA	BALM	MDH
Without Missing	***31.94***	34.09	32.83	41.19	46.66	53.00	53.88	61.33	66.34	70.51
With Missing	41.59	***39.76***	50.98	54.68	52.95	56.43	70.40	64.11	62.98	77.97

The aforementioned observations demonstrate the importance of testing against occlusion-based missing data as it contains a spatio-temporal structure of missing data that a randomly removed subset lacks. Many NRS*f*M methods treat missing data as a matrix fill-in problem, meaning recreating missing values from interpolation of spatio-temporally close observations. Thus, it is clear that conceptually it is much easier to interpolate random, evenly distributed missing data, compared to the spatio-temporally clustered structure of occlusion-based missing data. It is noted, that KSTA [32] and CSF [31] were both evaluated using random subset missing data in the original works, and was found to approximately have the same performance whether from 0% to 50% missing data. These results are obviously quite different from the conclusion of our study and we hypothesize, that the spatio-temporal structure of our occlusion-based missing is probably the primary cause for the drop in performance of many approaches.

6 Discussion and Conclusion

To summarize our findings, we would like to firstly mention that, the algorithm with the lowest error on average without missing data was found to be MultiBody [42].

There is, however, a large variation between the different algorithms performance depending on the factors chosen. As such our study does not conclude that Multibody [42] is definitively better than all other methods in general. As an example, for some camera paths RIKS [36] had lower average error than MultiBody [42]. Also, with missing data MetricProj [53] has the lowest reconstruction error. Other observations include that methods with a DCT basis were found to have a great increase in error with occlusion-based missing data. In general, the evaluated methods stay about two orders of magnitude behind the accuracy of the ground truth, showing that there is a need of improving current approaches.

Our study also shows findings that support hypotheses of where NRS*f*M research could head in the future. Even though some of these hypotheses have been stated before in related work, the strength of our data set and evaluation is able to confirm these. Firstly, it is clear that methods using the weak perspective approximation to the perspective camera model only incur a small penalty for doing so on average. This camera model seems like a good approximation, although it should be noted, that our data set does not challenge the algorithms extremely in this regard, with only an average 1.6 fold change in the depth variations. In particular, NRS*f*M applied in the medical domain, e.g. endoscopic imaging, may better benefit from a perspective camera model as the deforming body can be imaged at different depths while approaching with the endoscope to the regions of interest. Providing an in vivo data set for this scenario is a complex task requiring medical staff support. Some initial and promising efforts have been done for evaluating deformable registration methods [49] that could lead to a related NRS*f*M evaluation.

Moreover, given continuously deforming shapes, global temporal consistency should be enforced in order to avoid frame-by-frame sign flips, reflections and other ambiguities given the stronger geometrical expressiveness of deformable models. This is truly necessary in an operative scenario where such a problem might drastically reduce the effectiveness of the NRS*f*M approaches.

Another main avenue of investigation was the effect of missing data. Here we found, that that this aspect has a large impact on the reconstruction error. This is somewhat at odds with previous findings, and we speculate that this has to do with our missing data having structure originating from object self occlusion, as opposed to generate missing data with random sampling. In particular, occlusion-based missing data increases the reconstruction error of all methods except BALM [26]. Our study thus indicates this area to be a fruitful area of investigation for NRS*f*M research.

Another observation is that the physical based methods did quite poorly compared to the methods using a statistically based deformation model. This is in a sense counter intuitive, provided that the physical models capture the deformation physics well. This, in turn, leads us to the observation that stronger efforts could be beneficial as far as better physical based deformation models.

As stated, many of these observations, support hypothesis held in the NRS*f*M community, and it strengths them, that we have here provided empirical support for them. On the other hand, this study also helps to validate the suitability of our compiled data set. In regard to which, it should be noted, both deformation types and camera paths have a statistically significant

impact on reconstruction error, regardless of the algorithm used. This indicated that our proposed taxonomy and the data set design has value.

All in all, we have here presented a state of the art data set for NRS*f*M evaluation. We have applied 18 different NRS*f*M method to this data set. Methods that span the state of the art of NRS*f*M. This evaluation validates the usability of our proposed, and publicly available data set, and gives several insights into the current state of the art of NRS*f*M, including directions for further research.

References

1. Aanæs, H., Dahl, A., Steenstrup Pedersen, K.: Interesting interest points. International Journal of Computer Vision **97**, 18–35 (2012)
2. Aanæs, H., Jensen, R., Vogiatzis, G., Tola, E., Dahl, A.: Large-scale data for multiple-view stereopsis. International Journal of Computer Vision pp. 1–16 (2016)
3. Aanæs, H., Kahl, F.: Estimation of deformable structure and motion. In: In Workshop on Vision and Modelling of Dynamic Scenes, ECCV'02 (2002)
4. Agudo, A., Moreno-Noguer, F.: Dust: Dual union of spatio-temporal subspaces for monocular multiple object 3d reconstruction. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 6262–6270 (2017)
5. Agudo, A., Moreno-Noguer, F.: Force-based representation for non-rigid shape and elastic model estimation. IEEE Transactions on Pattern Analysis and Machine Intelligence (2017)
6. Agudo, A., Moreno-Noguer, F., Calvo, B., Montiel, J.M.M.: Sequential non-rigid structure from motion using physical priors. IEEE Transactions on Pattern Analysis and Machine Intelligence **38**(5), 979–994 (2016)
7. Akhter, I., Sheikh, Y., S., Kanade, T.: Trajectory space: A dual representation for nonrigid structure from motion. IEEE Transactions on Pattern Analysis and Machine Intelligence **33**(7), 1442–1456 (2011)
8. Akhter, I., Simon, T., Khan, S., Matthews, I., Sheikh, Y.: Bilinear spatiotemporal basis models. ACM Transactions on Graphics (TOG) **31**(2), 17 (2012)
9. Bartoli, A., Gay-Bellile, V., Castellani, U., Peyras, J., Olsen, S., Sayd, P.: Coarse-to-fine low-rank structure-from-motion. In: International Conference on Computer Vision and Pattern Recognition (2008)
10. Bouguet, J.Y.: Pyramidal implementation of the affine lucas kanade feature tracker description of the algorithm. Intel Corporation **5**(1-10), 4 (2001)
11. Brand, M., Bhotika, R.: Flexible flow for 3d nonrigid tracking and shape recovery. In: International Conference on Computer Vision and Pattern Recognition, pp. 315–22 (2001)
12. Brandt, S., ad J. Kannala, P.K., Heyden, A.: Uncalibrated non-rigid factorisation with automatic shape basis selection. In: Workshop on Non-Rigid Shape Analysis and Deformable Image Alignment (2011)
13. Bregler, C., Hertzmann, A., Biermann, H.: Recovering non-rigid 3D shape from image streams. In: International Conference on Computer Vision and Pattern Recognition, pp. 690–696 (2000)
14. Cha, G., Lee, M., Cho, J., Oh, S.: Reconstruct as far as you can: Consensus of non-rigid reconstruction from feasible regions. IEEE Transactions on Pattern Analysis and Machine Intelligence (2019)
15. Chhatkuli, A., Pizarro, D., Bartoli, A.: Non-rigid shape-from-motion for isometric surfaces using infinitesimal planarity. In: BMVC (2014)
16. Chhatkuli, A., Pizarro, D., Collins, T., Bartoli, A.: Inextensible non-rigid structure-from-motion by second-order cone programming. IEEE Transactions on Pattern Analysis and Machine Intelligence (2017)
17. Cho, J., Lee, M., Oh, S.: Complex non-rigid 3d shape recovery using a procrustean normal distribution mixture model. International Journal of Computer Vision **117**(3), 226–246 (2016)
18. Dai, Y., Li, H., He, M.: A simple prior-free method for non-rigid structure-from-motion factorization. International Journal of Computer Vision **107**(2), 101–122 (2014)
19. Dawud Ansari, M., Golyanik, V., Stricker, D.: Scalable dense monocular surface reconstruction. International Conference on 3DVision (2017)
20. Del Bue, A.: Adaptive non-rigid registration and structure from motion from image trajectories. International Journal of Computer Vision **103**, 226–239 (2013). URL `http://dx.doi.org/10.1007/s11263-012-0577-9`
21. Del Bue, A., Bartoli, A.: Multiview 3d warps. In: International Conference on Computer Vision, pp. 675–682 (2011)
22. Del Bue, A., Lladó, X., Agapito, L.: Non-rigid face modelling using shape priors. In: S.G. W. Zhao, X. Tang (eds.) IEEE International Workshop on Analysis and Modelling of Faces and Gestures, *Lecture Notes in Computer Science*, vol. 3723, pp. 96–107. Springer-Verlag (2005)
23. Del Bue, A., Lladó, X., Agapito, L.: Non-rigid face modelling using shape priors. In: AMFG, pp. 97–108. Springer (2005)
24. Del Bue, A., Llado, X., Agapito, L.: Non-rigid metric shape and motion recovery from uncalibrated images using priors. In: International Conference on Computer Vision and Pattern Recognition (2006)
25. Del Bue, A., Smeraldi, F., Agapito, L.: Non-rigid structure from motion using ranklet–based tracking and non-linear optimization. Image and Vision Computing **25**(3), 297–310 (2007)
26. Del Bue, A., Xavier, J., Agapito, L., Paladini, M.: Bilinear modeling via augmented lagrange multipliers (balm). Pattern Analysis and Machine Intelligence, IEEE Transactions on **34**(8), 1496 –1508 (2012). DOI 10.1109/TPAMI.2011.238. URL `http://users.isr.ist.utl.pt/~adb/publications/2012_PAMI_Del_Bue.pdf`
27. Deutsches Institut für Normung: VDI 2634: Optical 3-D measuring systems. Optical systems based on area scanning. Tech. rep., Deutsches Institut für Normung (2012)
28. Elhamifar, E., Vidal, R.: Sparse subspace clustering: Algorithm, theory, and applications. IEEE transactions on pattern analysis and machine intelligence **35**(11), 2765–2781 (2013)
29. Fayad, J., Agapito, L., Del Bue, A.: Piecewise quadratic reconstruction of non-rigid surfaces from monocular sequences. In: European Conference on Computer Vision (2010)
30. Golyanik, V., Jonas, A., Stricker, D.: Consolidating segmentwise non-rigid structure from motion. In: Machine Vision Applications (MVA) (2019)

31. Gotardo, P.F.U., Martinez, A.M.: Computing smooth time-trajectories for camera and deformable shape in structure from motion with occlusion. IEEE Transactions on Pattern Analysis and Machine Intelligence **33**(10), 2051–2065 (2011)
32. Gotardo, P.F.U., Martinez, A.M.: Kernel non-rigid structure from motion. In: IEEE International Conference on Computer Vision (2011)
33. Gotardo, P.F.U., Martinez, A.M.: Non-rigid structure from motion with complementary rank-3 spaces. In: IEEE Conference on Computer Vision and Pattern Recognition (2011)
34. Gower, J.C.: Generalized procrustes analysis. Psychometrika **40**(1), 33–51 (1975)
35. Gower, J.C., Dijksterhuis, G.B.: Procrustes problems, vol. 30. Oxford University Press on Demand (2004)
36. Hamsici, O.C., Gotardo, P.F., Martinez, A.M.: Learning spatially-smooth mappings in non-rigid structure from motion. pp. 260–273. Springer (2012)
37. Hartley, R., Vidal, R.: Perspective nonrigid shape and motion recovery. In: European Conference on Computer Vision, pp. 276–289 (2008)
38. Hartley, R.I., Zisserman, A.: Multiple View Geometry in Computer Vision. Cambridge University Press (2000)
39. Hyeong Hong, J., Zach, C., Fitzgibbon, A.: Revisiting the variable projection method for separable nonlinear least squares problems. In: The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2017)
40. Kong, C., Lucey, S.: Prior-less compressible structure from motion (2016)
41. Kumar, S.: Non-rigid structure from motion: Prior-free factorization method revisited. In: The IEEE Winter Conference on Applications of Computer Vision, pp. 51–60 (2020)
42. Kumar, S., Dai, Y., Li, H.: Spatio-temporal union of subspaces for multi-body non-rigid structure-from-motion. Pattern Recognition (2017)
43. Lee, M., Cho, J., Oh, S.: Consensus of non-rigid reconstructions. pp. 4670–4678 (2016)
44. Lee, M., Cho, J., Oh, S.: Procrustean normal distribution for non-rigid structure from motion. IEEE Transactions on Pattern Analysis and Machine Intelligence **39**(7), 1388–1400 (2017). DOI 10.1109/TPAMI.2016.2596720
45. Li, X., Li, H., Joo, H., Liu, Y., Sheikh, Y.: Structure from recurrent motion: From rigidity to recurrency. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 3032–3040 (2018)
46. Lladó, X., Del Bue, A., Agapito, L.: Euclidean reconstruction of deformable structure using a perspective camera with varying intrinsic parameters. In: Proc. International Conference on Pattern Recognition. Hong Kong (2006)
47. Lladó, X., Del Bue, A., Agapito, L.: Non-rigid metric reconstruction from perspective cameras. Image and Vision Computing **28**(9), 1339–1353 (2010)
48. Menze, M., Geiger, A.: Object scene flow for autonomous vehicles. In: Conference on Computer Vision and Pattern Recognition (CVPR) (2015)
49. Modrzejewski, R., Collins, T., Seeliger, B., Bartoli, A., Hostettler, A., Marescaux, J.: An in vivo porcine dataset and evaluation methodology to measure soft-body laparoscopic liver registration accuracy with an extended algorithm that handles collisions. International journal of computer assisted radiology and surgery **14**(7), 1237–1245 (2019)
50. Olsen, S.I., Bartoli, A.: Implicit non-rigid structure-from-motion with priors. Journal of Mathematical Imaging and Vision **31**(2), 233–244 (2008)
51. Ornhag, M.V., Olsson, C.: A unified optimization framework for low-rank inducing penalties. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 8474–8483 (2020)
52. Özyeşil, O., Voroninski, V., Basri, R., Singer, A.: A survey of structure from motion*. Acta Numerica **26**, 305–364 (2017)
53. Paladini, M., Del Bue, A., Stosic, M., Dodig, M., Xavier, J., Agapito, L.: Factorization for non-rigid and articulated structure using metric projections. In: International Conference on Computer Vision and Pattern Recognition (2009). DOI 10.1109/CVPRW.2009.5206602
54. Paladini, M., Del Bue, A., Stosic, M., Dodig, M., Xavier, J., Agapito, L.: Optimal metric projections for deformable and articulated structure-from-motion. International Journal of Computer Vision (IJCV) **96**, 252–276 (2012). DOI 10.1007/s11263-011-0468-5. URL `http://dx.doi.org/10.1007/s11263-011-0468-5`
55. Parashar, S., Pizarro, D., Bartoli, A.: Isometric non-rigid shape-from-motion with riemannian geometry solved in linear time. IEEE Transactions on Pattern Analysis and Machine Intelligence (2017)
56. Park, S., Lee, M., Kwak, N.: Procrustean regression: A flexible alignment-based framework for nonrigid structure estimation. IEEE Transactions on Image Processing **27**(1), 249–264 (2018). DOI 10.1109/TIP.2017.2757280
57. Reich, C., Ritter, R., Thesing, J.: White light heterodyne principle for 3D-measurement. In: O. Loffeld (ed.) Sensors, Sensor Systems, and Sensor Data Processing, vol. 3100, pp. 236 – 244. International Society for Optics and Photonics, SPIE (1997). DOI 10.1117/12.287750. URL `https://doi.org/10.1117/12.287750`
58. Russell, C., Fayad, J., Agapito, L.: Energy based multiple model fitting for non-rigid structure from motion. In: IEEE Conference on Computer Vision and Pattern Recognition (2011)
59. Russell, C., Yu, R., Agapito, L.: Video pop-up: Monocular 3d reconstruction of dynamic scenes. In: European conference on computer vision, pp. 583–598. Springer (2014)
60. Salzmann, M., Pilet, J., Ilic, S., Fua, P.: Surface deformation models for nonrigid 3d shape recovery. IEEE Transactions on Pattern Analysis and Machine Intelligence **29**(8), 1481–1487 (2007)
61. Seber, G.A., Lee, A.J.: Linear regression analysis, vol. 936. John Wiley & Sons (2012)
62. Simon, T., Valmadre, J., Matthews, I., Sheikh, Y.: Kronecker-markov prior for dynamic 3d reconstruction. IEEE transactions on pattern analysis and machine intelligence **39**(11), 2201–2214 (2017)
63. Szeliski, R.: Computer vision: algorithms and applications. Springer Science & Business Media (2010)
64. Taylor, J., Jepson, A.D., Kutulakos, K.N.: Non-rigid structure from locally-rigid motion. In: IEEE Conference on Computer Vision and Pattern Recognition (2010)
65. Tomasi, C., Kanade, T.: Shape and motion from image streams under orthography: A factorization approach. International Journal of Computer Vision **9**(2), 137–154 (1992)
66. Torresani, L., Hertzmann, A., Bregler, C.: Learning non-rigid 3D shape from 2D motion. In: S. Thrun, L. Saul, B. Schölkopf (eds.) Advances in Neural Information Processing Systems 16. MIT Press, Cambridge, MA (2004)
67. Torresani, L., Hertzmann, A., Bregler, C.: Non-rigid structure-from-motion: Estimating shape and motion with hierarchical priors. IEEE Transactions on Pattern Analysis and Machine Intelligence **30**(5), 878–892 (2008)

68. Torresani, L., Yang, D., Alexander, E., Bregler, C.: Tracking and modeling non-rigid objects with rank constraints. In: International Conference on Computer Vision and Pattern Recognition (2001)
69. University, C.M.: Cmu graphics lab motion capture database (2002). URL `http://mocap.cs.cmu.edu/`
70. Valmadre, J., Lucey, S.: General trajectory prior for non-rigid reconstruction. In: IEEE Conference on Computer Vision and Pattern Recognition (2012)
71. Varol, A., Salzmann, M., Tola, E., Fua, P.: Template-free monocular reconstruction of deformable surfaces. In: International Conference on Computer Vision, pp. 1811–1818 (2009)
72. Velleman, P.F., Hoaglin, D.C.: Applications, basics, and computing of exploratory data analysis. Duxbury Press (1981)
73. Vicente, S., Agapito, L.: Soft inextensibility constraints for template-free non-rigid reconstruction. In: European Conference on Computer Vision, pp. 426–440 (2012)
74. Vidal, R., Abretske, D.: Nonrigid shape and motion from multiple perspective views. In: European Conference on Computer Vision, pp. 205–218. Springer (2006)
75. Wang, G., Tsui, H., Wu, Q.: Rotation constrained power factorization for structure from motion of nonrigid objects. Pattern Recognition Letters **29**(1), 72–80 (2008)
76. Wang, G., Tsui, H.T., Hu, Z.: Structure and motion of nonrigid object under perspective projection. Pattern Recognition Letters **28**(4), 507–515 (2007)
77. Wang, Y.X., Lee, C.M., Cheong, L.F., Toh, K.C.: Practical matrix completion and corruption recovery using proximal alternating robust subspace minimization. International Journal of Computer Vision **111**(3), 315–344 (2015)
78. Williamson, D.F., Parker, R.A., Kendrick, J.S.: The box plot: a simple visual method to interpret data. Annals of internal medicine **110**(11), 916–921 (1989)
79. Xiao, J., Chai, J., Kanade, T.: A closed-form solution to non-rigid shape and motion recovery. International Journal of Computer Vision **67**(2), 233–246 (2006)
80. Xiao, J., Kanade, T.: Uncalibrated perspective reconstruction of deformable structures. In: IEEE International Conference on Computer Vision, pp. 1075–1082 (2005)
81. Zappella, L., Del Bue, A., Lladó, X., Salvi, J.: Joint estimation of segmentation and structure from motion. Computer Vision and Image Understanding **117**(2), 113–129 (2013)
82. Zhu, Y., Huang, D., De La Torre, F., Lucey, S.: Complex non-rigid motion 3d reconstruction by union of subspaces. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 1542–1549 (2014)

CONTRIBUTION G

Reconstructing transparent glass objects using polarized light

Reconstructing transparent glass objects using polarized light

Mads Doest
Technical University of Denmark - DTU
mebd@dtu.dk

Stuart James
Istituto Italiano di Tecnologia - IIT
stuart.james@iit.it

Alessio Del Bue
Istituto Italiano di Tecnologia - IIT
alessio.delbue@iit.it

Jeppe Revall Frisvad
Technical University of Denmark - DTU
jerf@dtu.dk

Abstract

Reconstruction of transparent objects (e.g. glass) traditionally relies on modelling very complex light interaction phenomena, but recently deep learning approaches have shown to be a viable alternative to traditional model-based methods. In this paper, we show that polarized images can provide a further boost to this compelling problem. This aspect is especially true because the polarization provides different image cues after reflection off external and internal surfaces. We therefore introduce PNN that uses polarized images as input and it outputs an estimation of the surface in terms of their *mask, normals, depth*. PNN is a fully supervised approach and we bypass the complexity of generating data using our CG-based engine for massively generating polarized images. We extensively experiment on the different numerical input formats and show X out performs the rest. We also provide an extensive ablation study on CNN techniques for the inference of the surface and provide insights over the method performance over varying complexity of glass surfaces.

1 Introduction

Reconstructing the surface geometry of a non-lambertian or transparent objects from images is one of the few inverse problems in computer vision still challenging the computer vision and graphics communities [10]. In the most unconstrained case, solving this problem is overly complex given several unknowns that relates to highly non-linear physical phenomena: The number of interfaces (e.g. glass-air) in the object are generally unknown, every interface usually causes both reflection and refraction, the Index of Refraction (IOR) of the different media depends on the wavelength, and interreflections are hard to deal with if no knowledge of the environment is available. To exacerbate further the problem, it is very expensive or in some cases not viable to obtain real 3D ground truth from standard objects. Examples of dataset are limited to few objects scanned with Computed Tomography (CT) [32] and then acquired with a camera. This has been a limiting aspect for learning approaches that often rely on advanced photorealistic renderers to provide enough data for training deep approaches [32, 28].

Still, the research community has attempted to solve 3D surface reconstruction of transparent objects, often with custom built setup [24, 38], a single fixed template [8, 24], RGBD sensors [11, 28] and using polarization [17, 3, 39]. In this paper we propose a new approach that leverage the recent availability of affordable polarized cameras, a new CG engine to generate polarized images for training data and a new model named PNN to to compute segmentation mask, depth and front & back normals of a transparent object from a single image. Although our approach can work with any images with arbitrary background, we also specialize the method to the case where the fixed background is an LCD screen. This choice is due to the fact that this monitor is a linearly polarized source [4] thus improving results over other less controlled polarized lights.

The contributions of the paper is therefore as follows:

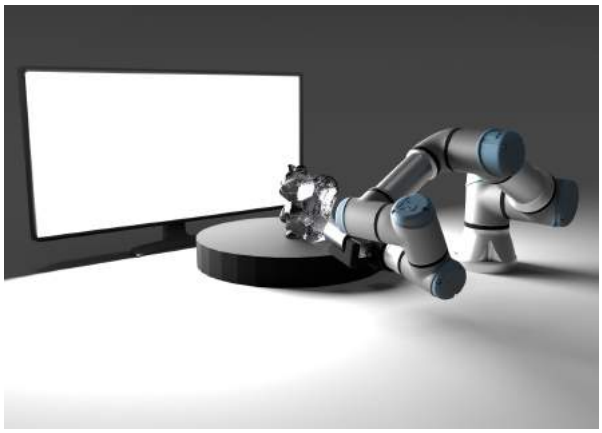


Figure 1: Exemplar capture setup for capturing polarized images. Light from a monitor passes through a transparent object and is captured by the polarized cameras. A robotic arm and turn table provide variation in views.

- Fast ray tracer for polarized camera including a model of the polarized emission of an LCD monitor;
- Synthetic dataset of polarized and RGB images;
- PNN custom model for estimation of Mask, Normal and Depth from Polarized Intensities and Stokes.

2 Related work

The reference method for 3D reconstruction of a transparent object is using Computed Tomography (CT) to acquire a volumetric representation of the object and then reconstructing the surface from this volume [14, 10, 31]. However, this reference technique requires slow and expensive CT scanning equipment and is limited by smoothing (and possibly other artifacts) introduced by the surface reconstruction technique available in standard scanners [31]. These shortcomings have pushed researchers to search for other techniques that are faster and more "light-weight" than a CT scanner. In this section, we review both general purpose reconstruction of transparent objects as well as related approaches to our PNN model.

Transparent Object Reconstruction Accurate 3D glass reconstruction is one of the most complex vision problems yet to solve and initial approaches relied mostly on custom-built lab setups. Within the field of transparent object reconstruction, the setup contains one or more cameras, the object in question, and a structured pattern display. This type of capture setup was originally presented by Kutulakos and Steger [12] and further explored by Chari and Sturm [2]. Qian et al. [24] also employ such a setup but focus on computing normals using two cameras to capture both the front and the back surface of the object. Given images taken from multiple configurations they are able to reconstruct glass objects of geometry such as spherical shapes and reading glasses. A more recent version of this setup [38] uses a screen and a turn table to expedite the capture. All these in-the-lab approaches are limited to two interfaces only and assume a known IOR, which is a common assumption in references on transparent object reconstruction. An exception to this is the work of Han et al. [8, 9] who proposed a fixed setup for dense reconstruction of transparent objects in high detail allowing them to handle various indices of refraction. The main drawback of their custom setup is that it requires moving a reference pattern to *a priori* known positions in order to infer the surface.

An interesting technique by Morris and Kutulakos [21] performs good quality reconstruction by capturing the trajectory of highlights for a moving diffuse point-like source. This enables reconstruction of heterogeneous transparent objects. The key difficulty in this technique is to get full coverage of the surface as trajectories are only available where highlights were observed.

Polarization-Based Reconstruction The Fresnel equations [5] describe the angularly dependent amount of reflection and refraction of light at an interface, including the polarization of the light. The fact that specular reflection causes polarization has multiple uses in vision. We can use it to separate highlight and diffuse reflection components [22]. This is useful for finding source directions and for removing highlights in shape from shading and photometric stereo. Saito et al. [27] used the degree of polarization revealed by the Fresnel equations to measure the orientation of the surface normals of a transparent surface. In turn, this led to the concept of shape from polarization [25].

Using the degree of polarization for surface normal reconstruction leads to an ambiguity because a given degree of polarization can be due to two different surface normals [19]. Different methods have been proposed for

disambiguation: using two wavelengths [19, 33] or two views [18] or an inverse rendering approach [20]. For transparent objects, such inverse rendering is based on polarization ray tracing [16, 17], where each ray carries a Stokes vector and a reference frame to represent the polarization of the light. Emission of light and light-matter interaction then requires a model describing polarization effects. This is usually accomplished using Mueller matrices that are applied to the Stokes vector.

The usefulness of polarization in vision has led to the development of polarimetric cameras [37, 15, 35]. Moreover, a standard liquid crystal display (LCD) monitor emits linearly polarized light. The combination of polarization imaging and use of an LCD monitor as a polarized light source has resulted in a solid technique for reconstructing surface normals [4] and for calibrating polarimetric cameras [36]. With a camera measuring the degree of polarization and through use of polarization ray tracing [17] in combination with shape from distortion [29, 34], Drouet et al. [3] developed a technique for acquiring both the front surface and the first internal surface (back surface, if the geometry is simple) of a transparent object. Their method however requires that reflections of the light source in the external and the internal surface do not overlap in an acquired image.

A circularly polarized light source in combination with a camera that provides more complete information of the Stokes vector is another way to address the disambiguation in surface normal reconstruction using a single view [7]. A combination of shape from polarization (this section) with triangulation methods for refractive surfaces [12, 2] (previous section) was presented by Xu et al. [39]. This technique is limited to simple object geometries due to refractive light-path triangulation being limited to one or two light bounces. Nevertheless, This work clearly indicates an advantage in use of the information provided by the partial polarization of the light when interacting with a specular/refractive surface.

3 Method

We aim to utilize multi-view and polarized information to inform a data-driven approach for 3D reconstruction. Our setup uses a robotic arm to capture multi-view images of the glass object (sec. 3.1). For each view we independently predict a mask, normal and depth to understand for the object in sec. 3.3, using a adaption on U-Net, then it would then be possible to integrate this information into a single model using Shape-from-Silhouette [13]. To achieve this we develop a ray-trace for polarized camera to synthetically generate a large dataset in Sec. 2.

3.1 Physical setup

Our setup consists of a robotic arm, stereo cameras screen and capture object. We use a single Universal Robots UR5 6-DOF robotic arm to move the cameras into a variety of positions. The end-effector is a custom frame housing two FLIR BFS-U3-51S5P-C polarization cameras from positioned in stereo. We use a 27" monitor to provide the light patterns to be passed through the sample object. The object is placed on a platform (turntable –although not used), where the center of the platform is placed 35cm from the screen. A selection of views of the full setup is shown in Figure 2.

We manually teach the robot a set of positions within the focal range of the object. This setup is performed once by manually moving the arm into a variety of positions. After the robotic arm then performs the same capture of views for an array of sample objects.

3.2 Polarized Dataset Creation

The total size of the dataset is 132k entries. Uncompressed this is roughly 2TB of data, and can be quite impractical to work with we also provice a smaller subset of the dataset of just 40k entries. Each entry consists of a photorealistic rendering with and without environment map, normals of the front and back surface, depth image, stokes vectors, and polarized intensity images simliar to the FLIR Polarization cameras. An example of this can be seen in Fig. 3.

3.2.1 Choice of 3D models

The dataset is based on a subset of objects from the Thingi10K [40] dataset. It consists of CAD models designed for 3D printing, thus the geometric properties of the dataset is very diverse and similar to objects produced in industry. Statistics on the geometrical properties of the dataset can be found as a spreadsheet on their website.

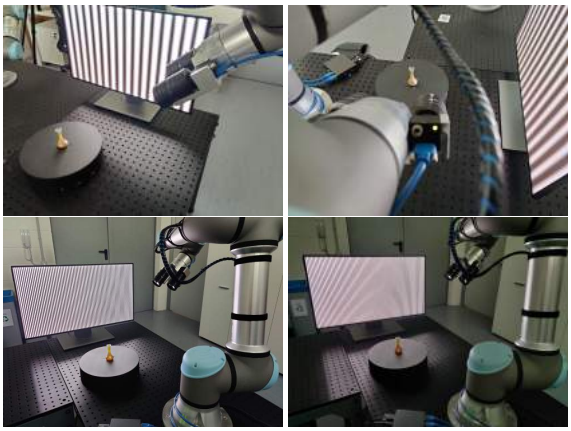


Figure 2: Top a synthetic setup, bottom a real capture setup showing a sample object from different perspectives. Setup uses a robotic arm (UR5) and the patterns displayed on a LCD display. The capture object is placed on a platform for easy identification and contrast.

Since Thingi10K is a massive dataset containing many different types of geometries we have limited the set using their search tool, to include only objects which have a single component, are manifold, are not degenerate and are closed. Further all mesh files have been decimated to only include up to 20k triangles. This was done to avoid cases of significantly increased rendering times. Due to the CAD models having a rather big size range, mm to m objects were scaled to be approximately the size of the table and screen.

Most of the previous work focus on reducing the complexity of the geometry supported by their method e.g. maximum of 2 interfaces, convex objects, no self occlusions, or even prior knowledge about parts of the geometry. While using Thingi10K the only assumption made is that the mesh is not degenerative, is closed and does not contain multiple objects. Also the genus of the objects varies from 1 to as high as 4886, allowing for very long and complex ray paths.

3.2.2 Polarization Renderer

We evaluate two different types of input the first being standard RGB data as proposed in [32], which we treat as grayscale given the capture setup, as well as polarized data. In this section, we outline the method for generating polarized training data where we use NVIDIA OptiX [23] for implementing a Monte Carlo ray tracer to render polarization images similar to images taken by the FLIR Polarization cameras mentioned in Sec. 3.1.

To include polarization in our renderer, we carry a Stokes vector and a reference frame together with each ray. In our shader (closest hit program), we account for the polarization caused by reflection and refraction of light by using the Fresnel equations. The plane on which the objects are placed is modelled by a Lambertian material. Lambertian materials work as a depolarization filter due to the light scatterings, this is why the plane is visible in s_0 but not in s_1, s_2 or s_3.

Polarization is due to the fact that photons have positive or negative spin. This property of photons is in the outset not represented by the wave theory of light. However, due to the principle of superposition, we can decompose any electromagnetic wave into two independent wave components and let these represent the polarization of the light (the spin preference of the photons). We can do the decomposition using an orthonormal frame of reference, where we can for example use the direction of wave propagation as one of the basis vectors. The directions of the other two basis vectors are not important, since it is just a mathematical tool to have two wave components for representing the polarization of the light. To obtain a reference frame, we use an efficient method for building an orthonormal basis from a unit direction vector [6]. A Stokes vector

$$\mathbf{s} = \begin{bmatrix} s_0, s_1, s_2, s_3 \end{bmatrix}^T \tag{1}$$

is then used to describe the amount of polarization in the wave components that the wave has been decomposed into.

The ratio of reflected radiance is given by the Fresnel equations [1]:

$$R_{\parallel} = \left| \frac{n_2 \cos\theta_i - n_1 \cos\theta_t}{n_2 \cos\theta_i + n_1 \cos\theta_t} \right|^2 \tag{2}$$

$$R_{\perp} = \left| \frac{n_1 \cos\theta_i - n_2 \cos\theta_t}{n_1 \cos\theta_i + n_2 \cos\theta_t} \right|^2 \tag{3}$$

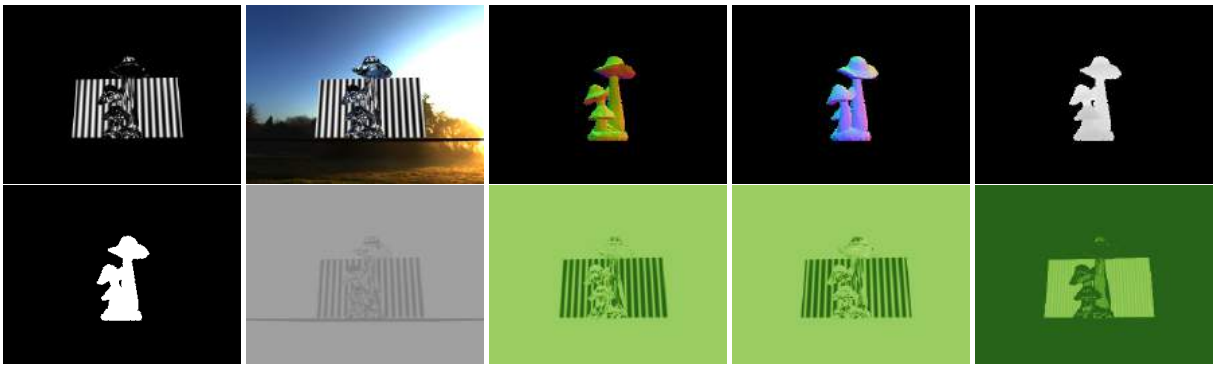


Figure 3: Showing an example from the rendered dataset, CAD model #53159. From top left we have: PB rendering no environment, PB rendering with environment, front-facing normals, back-facing normals, depth, mask, s0, s1, s2, and s3

$$T_{\parallel} = 1 - R_{\parallel} \quad (4)$$
$$T_{\perp} = 1 - R_{\perp}\,, \quad (5)$$

where n_1 is the index of refraction (IOR) of the medium with the incident and reflected light and n_2 is the IOR of the medium with the transmitted light, while θ_i and θ_t are the angles of incidence and transmission. The angle of reflection is equal to the angle of incidence.

It is important to realize that the reference frame and the Stokes vector belong together. If we rotate the reference frame, the Stokes vector must change accordingly. The change in a Stokes vector upon rotation of the reference frame is given by a so-called rotation Mueller matrix [16]. When light reflects and refracts at an interface, the amount of polarization is given by the Fresnel equations but these assume a reference frame with the basis vectors of the two wave components being perpendicular to and parallel with the plane of incidence. Thus, the rotation Mueller matrix is needed for every light bounce in a ray tracing in order to rotate the reference frame to the plane of incidence (which is spanned by the surface normal and the direction of wave propagation of the light). A reflection or transmission Mueller matrix can then be applied, and we can rotate the resulting Stokes vector back to the original reference frame. This procedure is described in previous work [16, 17].

Modelling LCD screens as a polarization light source. In order to do the Mueller-Calculus the stokes vector needs to have a reference frame, it seems logical to chose this frame as the xy-plane expanded by the screen. An interesting feature of LCD screens is that the light intensity in a pixel is controlled by changing the polarization on one of two filter with orthogonal polarization [26, p.751]

The basics for polarization raytracing are explained by Miyazaki et al. [17]. Where as a practicality in the implementation that is not explained there, is how to define reference frames for the Mueller-Calculus.

3.3 View-based Glass Estimation

We extend the work of [32] which constructs three independent networks to estimate the Depth, Normals and Mask from a single RGB Image. We implement a U-Net style architecture with skip connections. The convolutional blocks (ConvBlock) are as a standard 3×3 with 128 filter convolutional operation followed by batch normalization and a non-linear ReLU function. As input we explore both polarized or RGB image are of fixed size (w, h). The encoder follows the VGG-16 [30] architecture with 6 layers. In contrast to [32] we use a single encoder as opposed to three independent networks. For each of the output branches (mask, depth, normal) we use the standard deconvolution block which combines a bilinear up-sampling interpolation and convolutional filters, where the up-sampling as a power of two increase per layer, for the w, h this is 7 layers of up-sampling. Our network learns the ground truth mask (M), depth (D), normal foreground (N_f) and background (N_b), where the

outputs are therefore M', D', N'_f, N'_b respectively. We opt for standard architecture as opposed to optimizing for any specific traits like image sharpness allowing the network to learn a useful representation.

We combine the loss of the different output into a single loss

$$\mathcal{L} = \mathrm{L}_1(M, M') + \mathrm{L}_1(D, D') + \mathrm{L}_2(N_f, N'_f) + \mathrm{L}_2(N_b, N'_b) \quad (6)$$

where L_x refers to L1 and L2 loss. We opt for L2 loss on the normal as it allows the network to learn the complementary object mask with vector length of zero as opposed to an arbitrary unit length vector. We found this improves stability during training.

4 Evaluation

We evaluate the proposed approach using a quantitative studies of train test performance. As the proposed dataset is the first to synthetically render polarized images (See sec. 3.2 for dataset details). Firstly, we evaluate our model for different inputs RGB and two forms of Polarized images, Stokes and Intensities, in section 4.1.

4.1 View-based Estimation

As in [32] we evaluate based on the loss over multiple runs on a test set. We split the dataset in a random $80:20$ train:test split, where the results are shown in table 2 for the respective channel inputs. We compare training with environment map and without for both standard RGB and Polarized input. It is worth noting that in the case of the polarized network, the first layer is retrained from scratch to account for the four channel input in the case of Polarized Intensities, however, the standard VGG-16 is used to initialize RGB and the three Stokes channels (S_0, S_1, S_2).

From table 2 we see that the environment map is less useful in the case of polarized input. This is in strong contrast to prior work on transparency [31, 32] which greatly benefited from an environment map for RGB input. Given the additional information encoded in polarized images, it is intuitive that the environment map, which doesn't provide polarized light emission, does not contribute into an improved performance. Although it could be beneficial for mixed RGB and Polarized cameras.

4.1.1 Back face discussion

Incorporating the back face normals has an improvement in the outcome of the model. Although it isn't explicit it can imply the network is learning the relationship between the front face and the back face of the medium. Interestingly the network find better backnormals using only RGB data rather than polarization, which it contrary to our expectations. When observing the images from a polarization camera there is a noticeable difference between the different interfaces in the different polarization channels. One would think this would increase normal estimation in general for polarized images, but it might introduce difficult to model noise for areas with many interface interactions.

5 Conclusion

We have presented a method that takes advantage of polarized input for the reconstruction of transparent mediums and training a novel network PNN to achieve improved performance over the more traditional RGB with a slight improvement for mask and depth. But with a lower performance for back normals, which is quite interesting as we would expect the back normals to be significantly improved when using polarization information. With this network we are able to predict the mask, depth, and front- and back-normal maps, using images from polarization cameras.

References

[1] M. Born and E. Wolf. *Principles of Optics: Electromagnetic Theory of Propagation, Interference and Diffraction of Light*. Cambridge University Press, seventh (expanded) edition, 1999. 4

[2] V. Chari and P. Sturm. A theory of refractive photo-light-path triangulation. In *Proceedings of CVPR 2013*, pages 1438–1445, 2013. 2, 3

[3] F. Drouet, C. Stolz, O. Laligant, and O. Aubreton. 3D measurement of both front and back surfaces of transparent objects by polarization imaging. In *Reflection, Scattering, and Diffraction from Surfaces IV*, volume 9205, page 92050N. International Society for Optics and Photonics, 2014. 1, 3

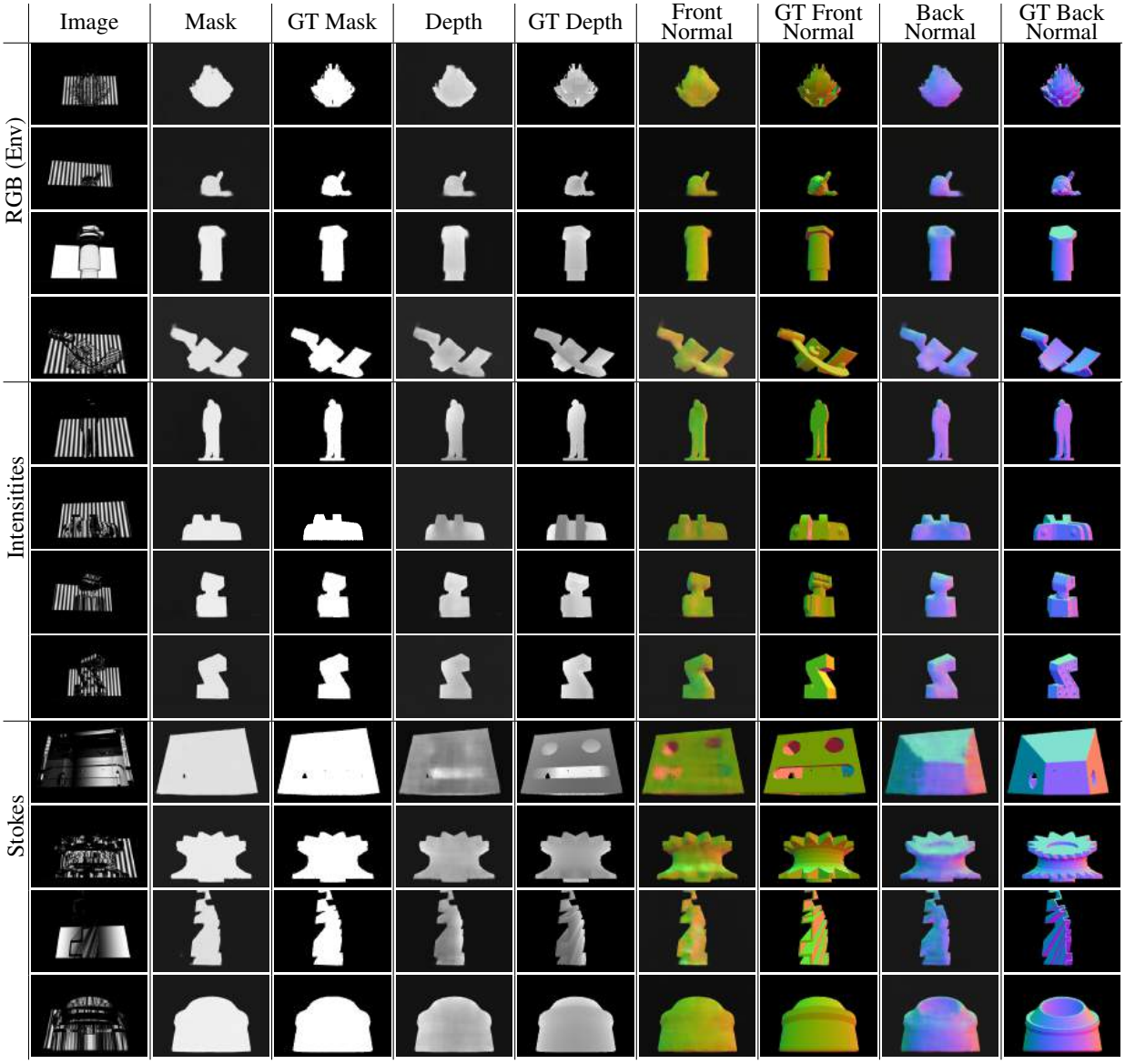


Table 1: We compare the mask, depth, normal and back face normal to GT across our three input modalities. As the environment map has not been shown to help in the case of polarized input we opt to show without. For each modality we show four random models that demonstrate varying surface and geometry complexity.

[4] Y. Francken, C. Hermans, T. Cuypers, and P. Bekaert. Fast normal map acquisition using an LCD screen emitting gradient patterns. In *Canadian Conference on Computer and Robot Vision (CRV 2008)*, pages 189–195. IEEE, 2008. 1, 3

[5] A. Fresnel. Mémoire sur la loi des modifications que la

	RGB				Polarized Stokes				Polarized Intensities			
	Mask	Depth	Normal	Back Normal	Mask	Depth	Normal	Back Normal	Mask	Depth	Normal	Back Normal
$SEMD + N_b$	0.0070	0.0233	0.0245	**0.0299**	0.0063	0.0193	0.0245	0.0305	**0.0059**	**0.0186**	0.0245	0.0305
$SEMD + N_b + Env$	0.0067	0.0204	0.0245	0.0305	0.0072	0.0219	0.0245	0.0298	0.0141	0.0404	0.0245	0.0302

Table 2: Comparison on PolSet40K of multiple models on RGB, Stokes and Intensities input data. We explore using environment map (as per prior papers [31]) and the use of back face normals.

réflexion imprime a la lumière polarisée. *Mémoires de l'Académie des sciences de l'Institut de France*, 11:393–434, 1832. Presented 7 January 1823. 2

[6] J. R. Frisvad. Building an orthonormal basis from a 3D unit vector without normalization. *Journal of Graphics Tools*, 16(3):151–159, August 2012. 4

[7] N. M. Garcia, I. De Erausquin, C. Edmiston, and V. Gruev. Surface normal reconstruction using circularly polarized light. *Optics Express*, 23(11):14391–14406, 2015. 3

[8] K. Han, K.-Y. K. Wong, and M. Liu. A fixed viewpoint approach for dense reconstruction of transparent objects. In *Proceedings of CVPR 2015*, pages 4001–4008, 2015. 1, 2

[9] K. Han, K.-Y. K. Wong, and M. Liu. Dense reconstruction of transparent objects by altering incident light paths through refraction. *International Journal of Computer Vision*, 126(5):460–475, 2018. 2

[10] I. Ihrke, K. N. Kutulakos, H. P. A. Lensch, M. Magnor, and W. Heidrich. Transparent and specular object reconstruction. *Computer Graphics Forum*, 29(8):2400–2426, 2010. 1, 2

[11] Y. Ji, Q. Xia, and Z. Zhang. Fusing depth and silhouette for scanning transparent object with rgb-d sensor. *International Journal of Optics*, 2017, 2017. 1

[12] K. N. Kutulakos and E. Steger. A theory of refractive and specular 3d shape by light-path triangulation. *International Journal of Computer Vision*, 76(1):13–29, 2008. 2, 3

[13] A. Laurentini. The visual hull concept for silhouette-based image understanding. *IEEE Trans. Pattern Anal. Mach. Intell.*, 16(2):150–162, Feb. 1994. 3

[14] W. E. Lorensen and H. E. Cline. Marching cubes: A high resolution 3D surface construction algorithm. *Computer Graphics (SIGGRAPH '87)*, 21(4):163–169, 1987. 2

[15] F. Meriaudeau, M. Ferraton, C. Stolz, O. Morel, and L. Bigué. Polarization imaging for industrial inspection. In *Image Processing: Machine Vision Applications*, volume 6813, page 681308. International Society for Optics and Photonics, 2008. 3

[16] D. Miyazaki and K. Ikeuchi. Inverse polarization raytracing: estimating surface shapes of transparent objects. In *Proceedings of CVPR 2005*, volume 2, pages 910–917. IEEE, 2005. 3, 5

[17] D. Miyazaki and K. Ikeuchi. Shape estimation of transparent objects by using inverse polarization ray tracing. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 29(11):2018–2030, November 2007. 1, 3, 5

[18] D. Miyazaki, M. Kagesawa, and K. Ikeuchi. Polarization-based transparent surface modeling from two views. In *Proceedings of ICCV 2003*, volume 3, page 1381, 2003. 3

[19] D. Miyazaki, M. Saito, Y. Sato, and K. Ikeuchi. Determining surface orientations of transparent objects based on polarization degrees in visible and infrared wavelengths. *Journal of the Optical Society of America A*, 19(4):687–694, 2002. 2, 3

[20] D. Miyazaki, R. T. Tan, K. Hara, and K. Ikeuchi. Polarization-based inverse rendering from a single view. In *Proceedings of ICCV 2003*, page 982. IEEE, 2003. 3

[21] N. J. Morris and K. N. Kutulakos. Reconstructing the surface of inhomogeneous transparent scenes by scatter-trace photography. In *Proceedings of ICCV 2007*, pages 1–8. IEEE, 2007. 2

[22] S. K. Nayar, X.-S. Fang, and T. Boult. Separation of reflection components using color and polarization. *International Journal of Computer Vision*, 21(3):163–186, 1997. 2

[23] S. G. Parker, J. Bigler, A. Dietrich, H. Friedrich, J. Hoberock, D. Luebke, D. McAllister, M. McGuire, K. Morley, A. Robison, and M. Stich. OptiX: A general purpose ray tracing engine. *ACM Transactions on Graphics*, 29(4):66:1–66:13, 2010. 4

[24] Y. Qian, M. Gong, and Y. Hong Yang. 3D reconstruction of transparent objects with position-normal consistency. In *Proceedings of CVPR 2016*, pages 4369–4377, 2016. 1, 2

[25] S. Rahmann and N. Canterakis. Reconstruction of specular surfaces using polarization imaging. In *Proceedings of CVPR 2001*, pages I–149–I–155. IEEE, 2001. 2

[26] E. Reinhard, E. A. Khan, A. O. Akyz, and G. M. Johnson. *Color Imaging: Fundamentals and Applications*. A. K. Peters, Ltd., USA, 2008. 5

[27] M. Saito, Y. Sato, K. Ikeuchi, and H. Kashiwagi. Measurement of surface orientations of transparent objects by use of polarization in highlight. *Journal of the Optical Society of America A*, 16(9):2286–2293, 1999. 2

[28] S. S. Sajjan, M. Moore, M. Pan, G. Nagaraja, J. Lee, A. Zeng, and S. Song. ClearGrasp: 3D shape estimation of transparent objects for manipulation. arXiv:1910.02550 [cs.CV], October 2019. 1

[29] S. Savarese, M. Chen, and P. Perona. Local shape from mirror reflections. *International Journal of Computer Vision*, 64(1):31–67, 2005. 3

[30] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. In *International Conference on Learning Representations*, 2015. 5

[31] J. D. Stets, A. Dal Corso, J. B. Nielsen, R. A. Lyngby, S. H. N. Jensen, J. Wilm, M. B. Doest, C. Gundlach, E. R. Eiriksson, K. Conradsen, A. B. Dahl, J. A. Bærentzen, J. R. Frisvad, and H. Aanæs. Scene reassembly after multimodal digitization and pipeline evaluation using photorealistic rendering. *Applied Optics*, 56(27):7679–7690, September 2017. 2, 6, 8

[32] J. D. Stets, Z. Li, J. R. Frisvad, and M. Chandraker. Single-shot analysis of refractive shape using convolutional neural networks. In *IEEE Winter Conference on Applications of Computer Vision (WACV 2019)*, pages 995–1003, 2019. 1, 4, 5, 6

[33] C. Stolz, M. Ferraton, and F. Meriaudeau. Shape from polarization: a method for solving zenithal angle ambiguity. *Optics Letters*, 37(20):4218–4220, 2012. 3

[34] M. Tarini, H. P. A. Lensch, M. Goesele, and H.-P. Seidel. 3D acquisition of mirroring objects using striped patterns. *Graphical Models*, 67(4):233–259, 2005. 3

[35] M. Vedel, S. Breugnot, and N. Lechocinski. Full stokes polarization imaging camera. In *Polarization Science and Remote Sensing V*, volume 8160, page 81600X. International Society for Optics and Photonics, 2011. 3

[36] Z. Wang, Y. Zheng, and Y.-Y. Chuang. Polarimetric camera calibration using an LCD monitor. In *Proceedings of CVPR 2019*, pages 3743–3752, 2019. 3

[37] L. B. Wolff and A. G. Andreou. Polarization camera sensors. *Image and Vision Computing*, 13(6):497–510, 1995. 3

[38] B. Wu, Y. Zhou, Y. Qian, M. Gong, and H. Huang. Full 3D reconstruction of transparent objects. *ACM Transactions on Graphics (SIGGRAPH 2018)*, 37(4):103:1–103:11, 2018. 1, 2

[39] X. Xu, Y. Qiao, and B. Qiu. Reconstructing the surface of transparent objects by polarized light measurements. *Optics Express*, 25(21):26296–26309, 2017. 1, 3

[40] Q. Zhou and A. Jacobson. Thingi10k: A dataset of 10,000 3d-printing models. *arXiv preprint arXiv:1605.04797*, 2016. 3

Technical
University of
Denmark

Richard Petersens Plads
Building 324
2800 Kgs. Lyngby
Tlf. 4525 3031

www.compute.dtu.dk