NORTHWESTERN UNIVERSITY

Rendering-Based Optimization for a Near-Eye Display and Active 3D Scanning

A DISSERTATION

SUBMITTED TO THE GRADUATE SCHOOL

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS

for the degree

DOCTOR OF PHILOSOPHY

Field of Computer Science

By

Nathan Matsuda

EVANSTON, ILLINOIS

September 2018

ABSTRACT

Rendering-Based Optimization for a Near-Eye Display and Active 3D Scanning

Nathan Matsuda

Raytracing is a long-established means to simulate physically accurate light propagation. Increasing availability and power of highly-parallel computing, such as cloud-based clusters and dedicated graphics hardware, means that rendering algorithms can produce high resolution output very quickly. This means raytracing can now be used as a forward model in optimization algorithms to improve the performance of computational imaging systems. This thesis considers the hypothesis that this rendering-based optimization approach for computational imaging systems will gain increasingly-widespread use in the future. This thesis evaluates two distinct uses for rendering-based optimization: a near-eye display and a surface reconstruction algorithm for active 3D scanners. In the first, a novel display architecture produces spatially varying focus for the user. Rendering-based optimization corrects hardware-induced optical distortions to produce an improved retinal image. The second use case flexibly corrects for multibounce interference in active 3D scanners, including arbitrary scene reflectance, using rendering-based optimization.

Acknowledgements

My path leading to graduate school began with my grandparents. Gwendolyn Walters and William Walters instilled my fascination with visual and physical form before I can even remember. Amy Matsuda and Dr. Fujio Matsuda inspired me to follow a principle of lifelong learning and to get a PhD along the way.

My formal computer science education began in high school with Andrew Merrill who built a strong foundation for my future work. As an undergraduate I was fortunate to have Prof. Jack Tumblin as my advisor. Jack introduced me to the field of computational imaging and later shepherded me into graduate studies.

My peers and labmates, including Lukas Schmid, Roman Koller, Thomas Niederberger, Dr. Xiang Huang, Dr. Leonidas Spinoulas, Dr. Salman Asif, Dr. Manish Sharma, Dr. Jason Holloway, Chia-Kai Yeh, Adithuya Pediredla, Fengqiang Li, Dr. Marina Alterman, and Dr. Florian Willomitzer, have made many contributions to my work and PhD experience along the way.

During my time at Northwestern, I have had the priviledge of working on projects with Prof. Aggelos Katsaggelos, Prof. Marc Walton, and Prof. Melville Ulmer, as well as with Prof. Ashok Veeraraghavan of Rice University and Prof. Andreas Velten of the University of Wisconsin. Though the results of these collaborations do not appear in this thesis, these professors have greatly expanded my understanding of imaging science.

The analysis, writing, and results in Chapter 2 were a collaboration with Dr. Douglas Lanman and Dr. Alexander Fix of Oculus Research. Portions of Chapter 3 contain

analysis, writing, and results produced in collaboration with Prof. Oliver Cossairt of Northwestern University and Prof. Mohit Gupta of the University of Wisconsin.

I would like to thank Dr. Jason Holloway and Dr. Florian Willomitzer for guidance in preparing this thesis.

I am incredibly grateful for the mentorship of Prof. Oliver Cossairt. At every stage of this process, including the phone calls that convinced me to start a PhD in the first place, he has been a calm and supportive guide. I have learned a great deal about the natural world, working with colleagues, and about myself, thanks to the environment Ollie creates for his students. I have been able to explore freely while looking to Ollie for hands-on guidance when needed. I cannot imagine a better advisor for me.

None of this doctoral study would have been possible without my family. My parents and brothers are always thoughtful, loving supporters, even now that we are spread around the country. My wife, Jen, encouraged this undertaking from the outset despite the sudden change in location and lifestyle that would ensue. She has been my companion through various intellectual and emotional ups and downs. And, of course, she is my most treasured friend.

Just over two years ago, our son Adler was born. His first years have been fascinating and joyful, and have also solidified my sense of self in a way I hadn't anticipated. He won't remember any of this, but I look forward to reminiscing with him someday about working at my computer and having a toddler bring dozens of 'tatos to pile on my desk, or sitting on my lap in front of the open thesis document and gleefully shouting 'delete!' while pressing the key over and over. Adler, you're a blast, and I love you.

Funding: Chapter 2 was funded by Oculus Research. The remainder of this thesis was funded by National Science Foundation Graduate Research Fellowship DGE-1324585 and by the Todd and Ruth Warren Fellowship. Portions of Chapter 3 pertaining to Motion Contrast 3D Scanning were funded by the Office of Naval Research award 1(GG010550)//N00014-14-1-0741, and Department of Energy contract DE-AC02-06CH11357.

Nomenclature

I	An intensity vector
$I_{x,y}$	The intensity associated with pixel (x, y)
$\mathbf{I}$	An intensity image in matrix form
$\hat{\mathbf{I}}$	A measured or target intensity image
$\bar{\mathbf{I}}$	A complex phasor image
s	A point in space or on a surface
S	A surface
P	A light path contribution
$P_{\mathrm{s}_0 \to \mathrm{s}_1}$	The light path contribution from s_0 to s_1
θ	A ray angle
ϕ	A phase function
ρ	A reflectance function
$\mathbf{T}$	A light transport matrix
T	A rendering operator
$T_\Gamma(S)$	The renderer initialized by parameters Γ as a function of surface S

Table of Contents

ABSTRACT 3

Acknowledgements 4

Nomenclature 7

Table of Contents 8

List of Tables 10

List of Figures 11

Chapter 1. Introduction 24

1.1. Summary 24

1.2. Problem Statement 26

1.3. Light Transport Representations 27

1.4. Tractability 33

1.5. Rendering Based Optimization 35

1.6. Hypothesis 38

1.7. Thesis Statement 38

Chapter 2. Accomodation-Supporting Near-Eye Displays 39

2.1. Near-Eye Displays 39

2.2. Related Work 44

2.3. Focal Surface Displays 51

2.4. Designing Focal Surface Displays 67

2.5. Implementation and Results 70

2.6. Discussion 76

2.7. Conclusion 83

Chapter 3. Active 3D Scanning with Multibounce Interference 84

3.1. Introduction 84

3.2. Active 3D Scanning Systems and Limitations 86

3.3. Motion Contrast 3D Scanning 91

3.4. Image Formation Modeling and Rendering for Active 3D Scanning 104

3.5. Optimizing Depth Estimates with Gradient Descent 107

3.6. Implementation 112

3.7. Limitations and Future Directions 129

Chapter 4. Conclusion 133

References 135

List of Tables

3.1 **Dynamic Range of Simulated Results:** Dynamic range, calculated as the unitless ratio of unambiguous range to RMS Z-axis error in meters, is listed for each of the v-groove and BRDF combinations depicted in Figure 3.7. Optimized results always produce higher estimated dynamic range than the underlying measurements, and in most cases the single-frequency optimized results outperform the dynamic range of the Kinect measurements. 117

List of Figures

1.1 **Forward Model Path:** An example raytrace path starting at the camera origin s_0, then traversing $d = 5$ bounces through the scene, with the final segment directly connecting to the light source. An additional valid path can be created at each intersection point by connecting directly to the light source (shown as dotted lines) to cheaply reduce sampling noise. In the SL case, the value along this path is set to be the phasor given by the projection of s_{d-1} into the projector reference frame, where the column value is converted into a phase value. In the ToF case, the projector and camera are co-located, while the path value is set to the phasor with angle corresponding to the phase rotation over the total path length given a modulation wavelength. In both cases, the magnitude of the phasor is attenuated by each bounce along the path by the BRDF model used for the surface. 30

1.2 **Optimization Block Diagram:** A rendering-based forward model can be used in gradient descent optimization even when an analytical derivative or adjoint operator are not available for the target parameters. In these situations, a finite difference approach can be applied, where each target parameter is offset in the scene and then rerendered

to produce a gradient image with respect to that parameter. The parameter can then be updated to incrementally minimize the objective function. 37

2.1 **Near-Eye Display Continuum:** Focal surface displays generalize the concept of manipulating the optical focus of each pixel on an HMD. Configurations (d,e) augment a fixed-focus HMD (a) with a programmable phase modulator placed between the eyepiece and display. (b) Varifocal HMDs use a globally addressed tunable lens. (c) Multifocal displays may use a high-speed tunable lens and display to create multiple focal planes. (f) In contrast, certain light field HMD concepts fall at the other end of this spectrum, using a finely structured phase modulator (a microlens array) placed near the display. (d) This chapter considers designs existing between these extremes in which a phase modulator locally adjusts the focus to follow the virtual geometry, generalizing varifocal and multifocal concepts. (e) Similar to multifocal displays, multiple focal surfaces can be synthesized with high-speed phase modulators and displays. 40

2.2 **Simplified Optical Diagram and Labels:** A focal surface display is created by placing a phase-modulation element between an eyepiece and a display screen. This phase element and the eyepiece work in concert as a spatially programmable compound lens, varying the apparent virtual image distance across the viewer's field of view. 51

2.3 **A Toy Scene Illustration:** A focal surface decomposition for a simple scene, containing: a background fronto-parallel plane at 1.0 diopters, a foreground fronto-parallel plane at 4.0 diopters, and a slanted plane spanning 2.0 to 4.0 diopters. (a) A single image from the target focal stack and target depth map. (b) A two-surface decomposition is compared to the target depth map for a profile taken along the middle row of the target imagery. (c) The color images associated with each focal surface are shown, using the linear blending method of Akeley et al. [**6**]. (d) The color images associated with each focal surface, using the rendering-based optimized blending algorithm presented in Section 2.3.3. 55

2.4 **Depth Error Assessment:** Focal surface displays achieve lower depth map approximation errors, using less time multiplexing, than prior multifocal methods. The left column depicts depth decompositions ranging from 0.0 to 5.0 diopters, abbreviated "D". The right column depicts the resulting depth map approximation errors in diopters. For a fixed focus design, the virtual image is positioned at 0.5 D. Following Narain et al. [**96**], the fixed multifocal display employs four planes evenly spaced from 0.2 D to 2.0 D. The adaptive multifocal display and the focal surface display are optimized using k-means clustering, following Wu et al. [**142**], and the methods in Sections 2.3.1 and 2.3.2 to position planes across a 5.0 D span, respectively. Focal surface displays show significantly fewer depth errors, with errors decreasing as more

surfaces are used. (Source imagery courtesy Unity Asset Store publisher "VenCreations.") 56

2.5 **Depth Error Assessment, Middlebury Dataset:** Focal surface displays represent natural scene depths with few image components. Box plots compare the depth map errors $g_{\theta_x,\theta_y}(d)$ using the denoted methods with the Middlebury 2014 dataset [**122**]. The bottom and top of the whiskers indicate the 5^{th} and 95^{th} percentiles, respectively. The bottom, middle, and top of the boxes represent the 1^{st}quartile, the median, and the 3^{rd} quartile, respectively. Focal surface displays produce fewer depth errors, especially when fewer planes are used. 61

2.6 **Depth Error Assessment, Unity Dataset:** We repeat the assessment of Figure 2.5, but with the database of rendered scenes described in Section 2.3.2. Note that the trends are repeated, but due to the larger depth ranges in this database, additional virtual image surfaces are required with prior fixed and adaptive multifocal displays. 62

2.7 **Perceptual Assessment:** Focal surface displays depict near-correct retinal blur with fewer virtual image surfaces than prior multifocal architectures. Following Figures 2.4–2.6, focal surface displays produce virtual images that more closely align with the scene geometry. As a result, sharply focused imagery can be obtained throughout the scene, reducing focusing errors occurring with prior fixed and adaptive multifocal displays. In this figure, we quantitatively assess the focal stack reproduction error following the method of Narain et al. [**96**]: the

right column depicts the maximum per-pixel probability of detecting a difference between the target and reconstructed focal stacks, as quantified using the HDR-VDP-2 metric [**84**]. The corresponding quality predictor of the mean opinion score (MOS) is listed along the bottom. Note that focal surface displays achieve similar fidelity as prior adaptive multifocal displays, although with fewer virtual image surfaces. (Source imagery courtesy Unity Asset Store publisher "VenCreations.") 66

2.8 **Design Trade Space:** The accommodation range of a focal surface display depends critically on the SLM placement. Here we denote, via the labeled plot contours, the virtual image distance z_v achieved with an SLM, when used to represent a lens of focal length f_p and positioned a distance z_p from the eyepiece. Red lines indicate focal lengths beyond the dynamic range of the SLM. Note that these numbers correspond with the prototype described in Section 2.5.1. 68

2.9 **Hardware Prototype:** Our binocular focal surface display prototype incorporates commodity optical and mechanical components, as well as 3D-printed support brackets. (a) The prototype is mounted to an optical breadboard to support the comparatively large LCOS driver electronics. (b) A cutaway of of the prototype exposes the arrangement of the optical components. 72

2.10 **Experimental Results, Optimized Blending:** Our prototype focal surface display achieves high resolution with near-correct retinal blur. Photographs of the prototype are shown in the first three columns, as

taken by focusing the camera at the indicated distances. The last two columns depict the corresponding optimization outputs, including the phase functions and the color images. Note that optimized blending is applied with three time-multiplexed focal surfaces. The phase functions are wrapped assuming a wavelength of 532 nm. (Source imagery courtesy Unity Asset Store publisher "VenCreations".) 74

2.11 **Experimental Results, Linear Blending:** Experimental results using linear blending over three time-multiplexed focal surfaces, following Akeley et al. [**6**]. (Source imagery courtesy of Thomas Guillon.) 74

2.12 **System Resolution and Chromatic Aberration:** a) The measured modulation transfer function (MTF) of our prototype as the system varies focus from 0.0 to 4.0 diopters. Increasing contrast loss is expected away from the prototype's neutral focus of 3.0 diopter as the SLM synthesizes shorter focal lengths, due to the increased stray light from phase quantization and phase resets. b) The measured axial chromatic aberration (ACA) of our prototype is less than that of the typical human eye [**29**], confirming that focal cues are correctly rendered with field-simultaneous color presentation, in spite of polychromatic illumination. 78

2.13 **Field-sequential and Simultaneous Color:** Field-simultaneous color display minimizes time multiplexing. However, artifacts due to axial chromatic aberration (ACA) may appear in this case. (a) A

target focal stack image. (b,c) Simulations comparing field-simultaneous and field-sequential modes, using the geometric optics model from Section 2.3. (d,e) Corresponding experimental results. The contrast of experimental results differs from simulations due to stray light and misalignments that cannot be predicted without more accurate wave optics modeling and calibration, respectively. (Source imagery courtesy Ruggero Corridori.) 79

3.1 **Taxonomy of Active 3D Systems:** Active 3D depth imaging systems face trade-offs in acquisition speed, resolution, and light efficiency. Laser scanning (upper left) achieves high resolution at slow speeds. Single-shot triangulation methods (mid-right) obtain lower resolution with a single exposure. Time-of-Flight methods obtain higher resolution results, but not at conventional camera resolutions. Other methods such as Gray coding and phase shifting (mid-bottom) balance speed and resolution but have degraded performance in the presence of strong ambient light, scene inter-reflections, and dense participating media. Hybrid techniques from Gupta et al. [**41**] (curve shown in green) and Taguchi et al. [**132**] (curve shown in red) strike a balance between these extremes. 86

3.2 **Motion Contrast Events**: When a motion contrast pixel observes a large change in intensity (a), the output (c) consists of ON events (red circles) when the change in log intensity over time exceeds a preset threshold (red dashed line) and OFF events (blue triangles) when the

change in log intensity drops below a preset threshold (blue dotted line). Each of these events is followed by a fixed reset time (hatched box) that is a function of the internal amplifier and differencing circuit characteristics. When the change in observed intensity is low (b), no output events are produced (d). 93

3.3 **System Model:** A scanning source illuminates projector positions α_1 and α_2 at times t_1 and t_2, striking scene points s_1 and s_2. Correspondence between projector and camera coordinates is not known at runtime. The DVS sensor registers changing pixels at columns i_1 and i_2 at times t_1 and t_2, which are output as events containing the location/event time pairs $[i_1, \tau_1]$ and $[i_2, \tau_2]$. We recover the estimated projector positions j_1 and j_2 from the event times. Depth can then be calculated using the correspondence between event location and estimated projector location. 97

3.4 **Comparison with Laser Scanning:** Laser scanning performed with laser galvanometer and traditional sensor cropped to 128x128 with total exposure time of 28.5s. MC3D method captured with 1 second exposure at 128x128 resolution and median filtered. Object placed 1m from sensor under ∼150 lux ambient illuminance measured at object. 99

3.5 **Performance with Interreflections:** Comparison between Kinect 1, phase shifting, and MC3D. Experimental setup shown in (a). A 90° v-groove, assembled from foam core board shown in (b). (c) and (d) show 45° and 30° v-grooves, respectively. Kinect 1 (measurements

averaged over 1 second) produces comparable results to MC3D in the 90° and 45° cases as the block matching algorithm rejects interreflections. In the 30° case, however, the block matching algorithm fails completely due to interreflections. Phase shifting (16 phase offsets recorded over 64 seconds of total exposure time, using a low frequency period equal to the width of the projector), has severe multibounce interference even at 90°. MC3D (measurements averaged over 1 second) is not susceptible to these effects as it is a laser point scanning technique. 100

3.6 **Motion Comparison:** The top row depicts 4 frames of a pinwheel spinning at roughly 120rpm, captured at 60fps using MC3D. The bottom row depicts the same pinwheel spinning at the same rate, over the same time interval, captured with the Kinect. Only 2 frames are shown due to the 30fps native frame rate of the Kinect. 102

3.7 **Simulated V-Groove Depth Profiles:** Simulated depth profiles for walls with known BRDFs meeting at 45°, 60°, or 90°. Top row: a diffuse-only BRDF. Second row: a physically based rough plastic BRDF. Third row: a glossy plastic BRDF. Bottom row: an anisotropic material. Profiles for ground truth, 10Mhz and Kinect simulated measurements, and optimized results are shown in each plot. Box-and-whisker plots of the associated error values with these plots are shown in Figures 3.8 and 3.9. 115

3.8 **10MHz Depth Errors:** Box-and-whisker plots (5th and 95th percentiles, quartiles, and median values) for Z-axis error in each

of the 10Mhz single-frequency results in Figure 3.7 (diffuse, glossy plastic, rough plastic, and anisotropic BSDFs). Error is calculated as the absolute distance in meters relative to ground truth. Box-and-whiskers for simulated measurements and reconstructions are shown on alternating lines in red and blue, respectively. 116

3.9 **Kinect Frequency Depth Errors:** Box-and-whisker plots (5th and 95th percentiles, quartiles, and median values) for Z-axis error in each of the Kinect multi-frequency results in Figure 3.7 (diffuse, glossy plastic, rough plastic, and anisotropic BSDFs). Error is calculated as the absolute distance in meters relative to ground truth. Box-and-whiskers for simulated measurements and reconstructions are shown on alternating lines in red and blue, respectively. 116

3.10 **Dynamic Range Versus Global/Direct Ratio:** Dynamic range for the simulated 10Mhz, Kinect, and optimized measurements, plotted as a function of the ratio between direct and global contributions to the scene intensity. Dynamic range is calculated as the total unambiguous range divided by the RMS depth value error relative to ground truth. 118

3.11 **Random Surface Performance:** In a), per-pixel root mean squared Z-axis error, relative to a ground truth random heightmap, plotted against the surface height standard deviation. The optimized result outperforms the simulated 10Mhz measurement. b) A centerline profile with a standard deviation of 2.0m showing ground truth, 10Mhz measurement, and optimized result. 119

3.12 **Convergence with Measurement Noise:** Per-pixel root mean squared Z-axis error, relative to ground truth for a 90° v-groove, plotted versus optimization iteration. Profiles shown for noise magnitudes ranging from a standard deviation of 0m through 1.87m. The total z-axis range spanned by the v-groove is 2m. 120

3.13 **Accuracy and Precision:** The accuracy of a simulated 10MHz 90° v-groove measurement, measured as RMS Z-axis error, can be consistently improved with increasing levels of measurement noise (a). Rendering-based optimization is able to correct for large-scale multibounce intereference despite the presence of high frequency noise in this case. The precision of this measurement, measured as the standard deviation of Z-axis error, cannot be improved by correcting for multibounce interference alone (b). 121

3.14 **Accuracy and Precision - Example Profile:** In a), a simulated 10MHz 90° v-groove measurement, with low accuracy due to multibounce interference, but high precision due to lack of noise. In b), the noise-free measurement optimized, which shows an improvement in accuracy. In c), a simulated 10Mhz 90° v-groove measurement with added noise. In d), the optimized result given the noisy measurement. Here the systematic error has been reduced, improving accuracy, but the low precision due to measurement noise remains. To address this, an example using total variation regularization in the optimization loop is shown in (e). 122

3.15 **Experimental Setup:** Foam core boards aligned to marks on the floor form a 90° v-groove. A Kinect ToF center is placed along the centerline and aligned with the center of the v-groove and the horizon. Kinect output was captured with a PC laptop. Board angles were then moved to 60° and 45° angles and captured with the Kinect. 125

3.16 **Experimental V-Groove Depth Profiles:** Top Row: Experimental depth results using Microsoft Kinect for capture, shown as middle-row profiles plotted in scene space, recovered from a physical scene containing foam core boards meeting at an angle of 45°, 60°, or 90°. Ground truth is approximated with two separate captures, one for each of the left and right sides of the v-groove to eliminate multibounce interference. Middle Row: Measurements converted to rectilinear mesh and illuminated with a directional light source from the upper left. Bottom Row: Optimized results converted to mesh and rendered. Column A) shows a 90° v-groove. Column B) the 60° v-groove. Column C) the 45°. The optimized result (30 iterations) consistently outperforms the physical measurement due to the algorithm's ability to account for multibounce interference. 126

3.17 **Experimental Capture, Stairs:** Experimental depth results using Microsoft Kinect for capture, shown as middle-row profiles plotted in scene space (b), recovered from a physical set of varnished wooden stairs, shown in (a) with an inset showing the tread and riser profile highlighted. A lit, rendered mesh produced from the raw Kinect

measurement is shown in (c). The same treatment is applied to the optimized result in (d). 127

3.18 **Simulated SL Depth Profiles:** Simulated depth profiles for diffuse walls meeting at 45°, 60°, or 90°. Profiles for ground truth, phase shift, micro phase shift, and optimized results are shown in each plot. Like ToF results, the optimzed phase shift results approach the quality of micro phase shifting, which optically reduces multibounce interference. 130

CHAPTER 1

Introduction

1.1. Summary

Graphics researchers began developing rendering algorithms to simulate ray optics half a century ago. Since then, the community introduced accurate models for complex optical effects like surface scattering and multi-bounce illumination, albeit with increasing computational cost. Recently, electronics manufacturers have met growing consumer demand for interactive visual content on mobile phones and laptops by producing ever-faster processing hardware at higher volumes. Now, realistic rendering algorithms can operate at vastly higher speeds than even a decade ago.

When rendered optical simulation is physically accurate, we can use it to confirm the measurements or output of an optical device. The renderer serves as an independent but equivalent platform to reproduce light passing through the system. In the case of a display, a rendered model of emitted light can be used to verify that the user sees the intended image. In the case of a camera, a rendered model can verify scene properties associated with measured values. Any system, so long as it can be accurately modeled with a renderer, can be evaluated in this way.

Now that processing hardware is sufficiently fast, we can incorporate raytracing into an iterative gradient descent algorithm. The renderer becomes an efficient testbed for

progressively refining system parameters. The algorithm repeatedly updates target system parameters, estimates the resulting propagation of light using the renderer, compares this output to the desired output, and then computes an update to the target parameters. This rendering-based optimization approach can infer scene properties from captured measurements, or display parameters given a target output, taking into account the classical effects present in the system.

This tool has been used to recover surface orientation [**99**], to look around corners [**64**], and to estimate volumetric scattering parameters [**35**], amongst a variety of other complex reconstruction tasks.

This thesis evaluates rendering-based optimization in:

- **An Accommodation-Supporting Near-Eye Display:** Conventional near-eye displays produce a single plane of focus. This thesis proposes a system comprised of a conventional near-eye display and an inline phase modulator. A phase modulator warps the focus perceived by the viewer to approximate depth cues in the virtual scene, but in doing so induces spatially varying aberrations. Rendering-based optimization generates a color display image to best synthesize the desired retinal image through these distortions.
- **A Time-Of-Flight Reconstruction Algorithm:** Active depth imaging systems suffer from multibounce interference when observing concave geometry. This interference is also dependent on the surface reflectance of the scene. While hardware system design can mitigate some of these effects, this thesis demonstrates a rendering-based optimization to further improve depth estimates by eliminating systematic errors caused by interreflections.

1.2. Problem Statement

Many problems in computational imaging can be solved as linear systems which are easily tractable, but still powerful for modeling the appearance of a scene. These include deblurring [**118**], compressive sensing [**24**], dual photography [**125**], image relighting [**101**], and the estimation of environment lighting and reflectance [**88, 117**]. Yet there are many other inverse problems where appearance is poorly described by linear models.

This thesis is concerned with the following problem in this category:

Scene appearance is a highly nonlinear function of geometry and material.

Estimating geometry and material properties given measurements of scene appearance is the core challenge in 3D acquisition and display, which in turn plays a role in applications as diverse as virtual reality, visual effects, robotic navigation, metrology, and scientific imaging.

How can we estimate parameters to control a display when individual optical components produce nonlinear distortions? Or, how can we estimate scene geometry if it produces nonlinear distortions on measurements? This thesis will begin to address this question by examining the equivalency between linear image formation models and an alternative, nonliner image formation model: raytracing.

1.3. Light Transport Representations

The rendering equation [**60**], using notation inspired by [**114**] (see Nomenclature for reference), is given as:

$$I(\mathrm{s},\mathrm{s}') = G(\mathrm{s},\mathrm{s}')\left[I_e(\mathrm{s},\mathrm{s}') + \int_S \rho(\mathrm{s},\mathrm{s}',\mathrm{s}'')I(\mathrm{s},\mathrm{s}')\right] \tag{1.1}$$

where $I(\mathrm{s},\mathrm{s}')$ is the radiant transfer between points s and s′, $G(\mathrm{s},\mathrm{s}')$ is the mutual visibility between the two points, $I_e(\mathrm{s},\mathrm{s}')$ is the emissive contribution from s to s′, and $\rho(\mathrm{s},\mathrm{s}',\mathrm{s}'')$ is the intensity scattered from point s″ to s off of the point s′ (the reflectance). The integral is computed over the entire scene surface S.

This integral cannot be solved directly in all but the most trivial cases, but Kajiya gives a stochastic approximation using Monte Carlo integration in [**60**]. This approach, also known as path tracing, remains the standard for unbiased, physically based rendering in addition to extensions such as bidirectional path tracing [**70**] and Metropolis light transport [**135**]. This approach is implemented in the PBRT rendering engine and corresponding textbook [**114**], the spin-off Mitsuba renderer [**55**] (modified for the simulations and results in Chapter 3), the hardware-based Optix framework [**112**] (modified for the simulations and results in Chapter 2), as well as the open-source Blender rendering engine [**11**] which produced all visualization renders in this thesis.

The path tracing algorithm in [**114**] gives the intensity along a path P after recursion depth d:

$$P = \frac{I_e(\mathrm{s}_d \rightarrow \mathrm{s}_{d-1})\rho(\mathrm{s}_d \rightarrow \mathrm{s}_{d-1} \rightarrow \mathrm{s}_{d-2})G(\mathrm{s}_d \leftrightarrow \mathrm{s}_{d-1})}{p_A(\mathrm{s}_d)} \times \left(\prod_{j=1}^{d-2} \frac{\rho(\mathrm{s}_{j+1} \rightarrow \mathrm{s}_j \rightarrow \mathrm{s}_{j-1})|cos\theta_j|}{p_\omega(\mathrm{s}_{j+1} - \mathrm{s}_j)} \right) \tag{1.2}$$

Beginning with the camera center of projection s_0, the algorithm samples an outgoing ray direction which intersects the scene at point s_1. Each subsequent recursion depth d along the path P is sampled by selecting a direction from the bidirectional scattering distribution function (BSDF) at the previous point and tracing until a new surface is hit. Each of these intermediate points can be connected to an emitter to form a valid path.

In Eq. 1.2, the intermediate point connections are made by temporarily assigning s_d to a point on the light source surface S_A. The contribution from this point is the product of the radiance from s_d back to the previous point s_{d-1}, reflectance ρ at that previous point given the incoming and outgoing ray directions, and a geometric coupling term G (accounting for visibility and the surface normals at each endpoint of the ray), times the product of all previous reflectances in the path weighted by the solid angle associated with each reflection. After including the new contribution from this sub-path in the running total, a new surface point is selected for s_d and the algorithm moves on to the next iteration, until the desired maximum path depth is reached.

For clarity, the authors in [**114**] separate attenuation for all points prior to the most recent one into a term β, which we will also use in subsequent expressions.

$$\beta = \prod_{j=1}^{d-2} \frac{\rho(\mathrm{s}_{j+1} \rightarrow \mathrm{s}_j \rightarrow \mathrm{s}_{j-1})|cos\theta_j|}{p_\omega(\mathrm{s}_{j+1} - \mathrm{s}_j)} \tag{1.3}$$

One such path is shown as an example in Figure 1.1. The path shown traverses the scene to a depth of 5 bounces, starting at the camera center of projection s_0. The direction of the first segment is selected by the raytracer depending on the resolution of the camera and the pixel subsampling scheme used. Each subsequent bounce direction is determined depending on the probability distribution function of the surface reflectance model at that point. The final segment connecting s_{d-1} to s_d is a direct path to the light source. Each intermediate path point is also connected to the light source as this improves variance in the estimate of the light transport integral without incurring the computational cost of an additional full path trace. These subpath connections to the light source are shown as dotted lines. Attenuation along the path is determined iteratively according to Eq. 1.2, in a conventional manner.

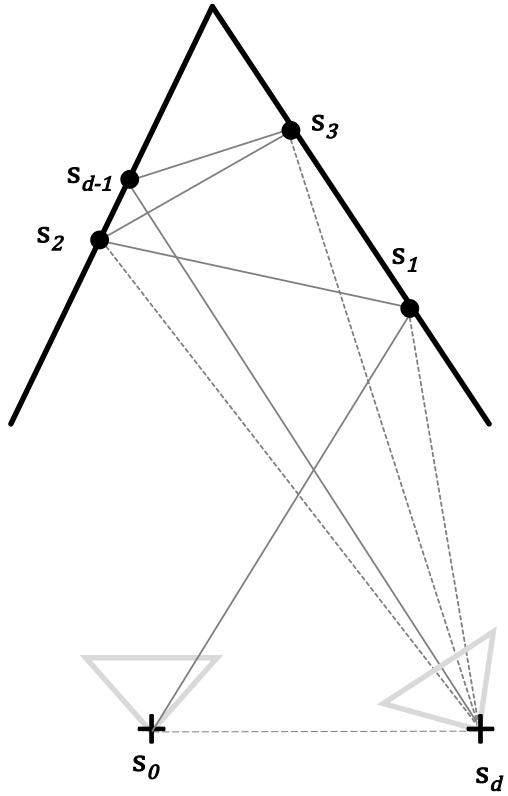


Figure 1.1. **Forward Model Path:** An example raytrace path starting at the camera origin s_0, then traversing $d = 5$ bounces through the scene, with the final segment directly connecting to the light source. An additional valid path can be created at each intersection point by connecting directly to the light source (shown as dotted lines) to cheaply reduce sampling noise. In the SL case, the value along this path is set to be the phasor given by the projection of s_{d-1} into the projector reference frame, where the column value is converted into a phase value. In the ToF case, the projector and camera are co-located, while the path value is set to the phasor with angle corresponding to the phase rotation over the total path length given a modulation wavelength. In both cases, the magnitude of the phasor is attenuated by each bounce along the path by the BRDF model used for the surface.

Despite the widespread use of this stochastic raytracer, many problems rely on linear image formation models in matrix form, particularly those involving linear optimization. An alternative notation for Eq. 1.1 is mentioned in [**60**], where $\mathbf{M}$ is the operator form of the integral over all surfaces. Then,

$$I = \mathbf{G}I_e + \mathbf{GM}I \tag{1.4}$$

Redistributing terms, the recursive I term can be eliminated:

$$(\mathbf{1} - \mathbf{GM})I = \mathbf{G}I_e \tag{1.5}$$

Here 1 is the identity operator. Expressing the rendering equation in operator terms is explored further by Arvo in [**8**]. Now, if we define an operator $\mathbf{T}$ as:

$$\mathbf{T} \equiv (\mathbf{1} - \mathbf{GM})^{-1} \tag{1.6}$$

We can establish a linear relationship between input intensities I_e and output intensities I in a scene.

$$I = \mathbf{T}I_e \tag{1.7}$$

This linear model now matches the form used in the linear systems approaches listed in the previous section. If the model is a raytrace operator rather than a matrix, the functional notation becomes:

$$I = \mathbf{T}(I_e) \tag{1.8}$$

where the rendering algorithm takes as an argument emissive points in the scene I_e.

Unlike a purely matrix-based representation of light transport, if the forward model is a raytracer, then higher order parameters can be controlled directly. The light transport operator can be specified with scene surface geometry S and reflectance parameters R, subsequently estimating the effects of those parameters as a function of input illumination.

$$I = \mathbf{T}_{S,R}(I_e) \tag{1.9}$$

These scene descriptors may be further parameterized for ease of use in individual applications, as in Chapter 2, where the renderer is initialized with an objective lens focal length and a spatial light modulator phase function, rather than the corresponding surface shapes and reflectance functions.

Before detailing these individual implementations, what are the circumstances that would lead us to use a stochastic renderer over a linear light transport representation? In the following section we will look at the consequences of this choice on problem tractability.

1.4. Tractability

The linear system in Eq. 1.7, while conceptually simple, can represent considerable computational complexity when representing camera and display systems with modern resolutions. For example, if we have a high definition projector-camera pair, each addressing 1920x1080 pixels with 3 color channels, the full light transport matrix $\mathbf{T}$ mapping the projector intensities to camera intensities is a square matrix with 6.2 million elements on a side. If each intensity value is digitized with 8 bits, then this matrix contains nearly 39 terabytes of information.

Now, for many scenes a given projector ray will only affect a local neighborhood of pixels, meaning that the distribution of energy in the light transport matrix will be sparse. Say each projector ray scatters to 10% of pixels in the camera on average. Using compressed sparse row formatting, with 32 bit row and column indices, this light transport matrix will still be approximately 19 terabytes. Of course, many matrix operations will require floating point precision — if the 8 bit intensity values are converted to a 32 bit floating point number, then this same sparse matrix will be approximately 31 terabytes.

Many image formation models can increase in size significantly beyond the above example. The focal surface deblurring algorithm in Chapter 2 uses a 13-slice focal stack optimization target and additive time multiplexing over 4 frames. Sparse representations of the light transport matrix corresponding to this system require approximately 20 petabytes to store, again in the case of 10% sparsity. This huge amount of information would need to be stored in memory to be used and manipulated in an optimization routine, possibly with multiple auxiliary representations. This exceeds the limits of large distributed computing systems, let alone the memory available on an embedded system.

This work explores two scenarios where some, but not all, of the generality of a light transport representation is needed to solve an inverse problem. The memory footprint described above is drastically reduced for these applications by using a raytracer that recreates light transport effects without an explicit light transport representation.

A striking example of the problem size and complexity that can be handled using rendering-based optimization is the recovery of volumetric scattering parameters by Gkioulekas et al. [**35**]. Though the authors use a 100 node cluster to perform inverse rendering, volumetric raytracing is a notoriously expensive computational class. The inverse rendering algorithms we introduce in each of Chapter 2 and 3 can operate on a single desktop PC.

Significant gains in efficiency would be required for these techniques to be used in low-latency, real-time scenarios like autonomous navigation. Nonetheless, a vast range of applications allow for offline processing, such as metrology, cultural heritage studies, and the many reality capture techniques for virtual reality, augmented reality, and visual effects. In these fields, processing times on the order of hours are typical for day-to-day computational tasks. Using rendering-based optimization to gain higher quality imaging results on these time scales is already feasible with modest desktop computing power in some cases. The parallelization of computing power, whether geared toward hardware-based deep learning or high-volume cloud-based services, will serve to make rendering algorithms increasingly efficient.

With this motivation for pursuing rendering-based optimization for real-world systems, we will now examine how a stochastic renderer is incorporated into a gradient descent framework.

1.5. Rendering Based Optimization

The equivalency between a linear light transport model and a stochastic raytracing model described in section 1.3, coupled with the problem size advantage of raytracing models over linear models described in 1.4, lead this thesis to address imaging systems using raytraced-based models.

For example, consider a motion deblurring problem. If we wish to recover a sharp image intensity vector I but measure blurred intensity vector $\hat{I}$, there is some light transport matrix $\mathbf{T}$ which defines the contribution of each sharp image point to each measured intensity. To recover I, we can calculate the inverse (or pseudoinverse if input and output dimensions are mismatched) of $\mathbf{T}$:

$$I = \mathbf{T}^{-1}\hat{I} \tag{1.10}$$

But, as the previous section highlighted, the measurement, storage, and inversion of $\mathbf{T}$ may be impractical. If we instead implement a renderer which contains a model of blur formation, $T(I)$, then we can avoid excessive problem size. Instead, we perform an operator-based optimization rather than using matrix inversion methods.

This thesis utilizes two approaches. In one case, an adjoint operator $T^*(I)$ is available, allowing the use of a conjugate gradient method [**109**]. Though this optimizer is designed to solve a linear problem in the form of Eq. 1.7, it does not invert $\mathbf{T}$ or even require direct access to that matrix. Instead, it requires the output of $\mathbf{T}I$ and $\mathbf{T}^T I$, which the forward and adjoint renderers provide, in order to solve a QR factorization and arrive at a solution for I.

For the problem in Chapter 3, where an adjoint operator is not available, this thesis employs a finite difference approach. This is similar to the iterative approach taken in the shape-from-interreflections algorithm [**99**], a key precursor to the inverse-rendering optimizations discussed in this thesis. Here, finite difference approximations to partial derivatives are calculated for each parameter Γ in the optimization as:

$$\frac{\partial I}{\partial \Gamma} \approx \frac{T(I + \delta_\Gamma) - T(I)}{\delta_\Gamma} \tag{1.11}$$

where δ_Γ is a small offset in parameter Γ. These partial derivatives are then used to update I. The operational flow of this approach is depicted as a block diagram in Figure 1.2. In [**78**], rendered finite difference approximations to partial derivatives are used to optimize body pose parameters to match depth camera measurements, using an energy minimization approach. Scene albedo and reflectance are iteratively recovered in [**13**]. A stochastic method is used by [**35**], while the results in Chapter 3 in this thesis use a batch method to improve parallelization.

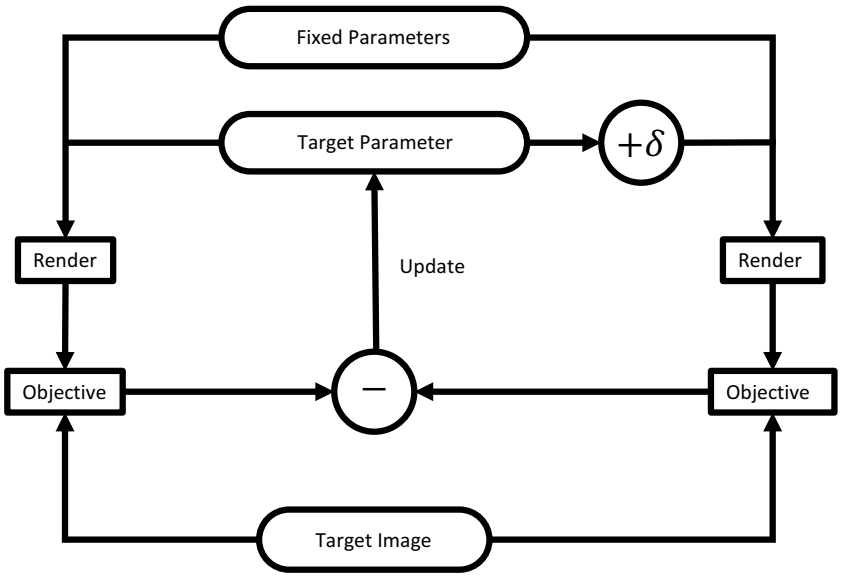


Figure 1.2. **Optimization Block Diagram:** A rendering-based forward model can be used in gradient descent optimization even when an analytical derivative or adjoint operator are not available for the target parameters. In these situations, a finite difference approach can be applied, where each target parameter is offset in the scene and then rerendered to produce a gradient image with respect to that parameter. The parameter can then be updated to incrementally minimize the objective function.

Deep neural networks have been proposed for inverse rendering problems including face geometry and reflectance recovery [**62**], as well as for time-of-flight depth reconstruction with multibounce interference ([**130**], [**85**]). These approaches are promising for their generality and processing speed. Like many deep learning techniques, though, they lack the foundation and interpretability of well-understood light transport models present in rendering-based optimizers.

Of these techniques, rendering-based optimization is appealing because of the long history and sophistication of raytrace renderers, the growing availability of parallel computing platforms, and the ease with which raytracers can be incorporated into gradient

descent algorithms. These qualities point toward a future where this approach is used more widely.

1.6. Hypothesis

Inverse rendering will play an increasingly important role in computational camera and display systems.

This thesis tests the hypothesis on two open problems in computational imaging, which leads to the following thesis statement.

1.7. Thesis Statement

Rendering-based optimization can:

(a) **Correct aberrations in a phase modulating near-eye display configuration to produce natural, spatially varying focus cues.**

(b) **Correct bias due to global light transport in an active 3D scanning surface recovery algorithm.**

CHAPTER 2

Accomodation-Supporting Near-Eye Displays

2.1. Near-Eye Displays

A modern head-mounted display (HMD), as designed for virtual reality (VR) applications, is a simple construction placing viewing optics (e.g., a magnifying lens) between the user's eye and a display screen. This configuration is replicated for binocular stereo configurations: one set of optics and one display, or portion of a display, is dedicated to each eye. In this manner, a binocular HMD depicts stereoscopic imagery such that the user perceives virtual objects with correct retinal disparity, which is the critical stimulus to vergence (the degree to which the eyes are converged or diverged to fixate a point) [**113**].

VR viewing optics typically create a virtual, erect, magnified image of the display screen, located at a fixed focal distance from the user [**17**]. Thus, current VR HMDs do not correctly depict retinal blur, which is the critical stimulus to accommodation (the eyes' focusing response). The resulting vergence-accommodation conflict (VAC) has been identified as a source of visual discomfort: viewers report eye strain, blurred vision, and headaches with prolonged viewing [**126**]. VAC has also been linked to perceptual consequences, affecting eye movements and the ability to resolve depth [**44**].

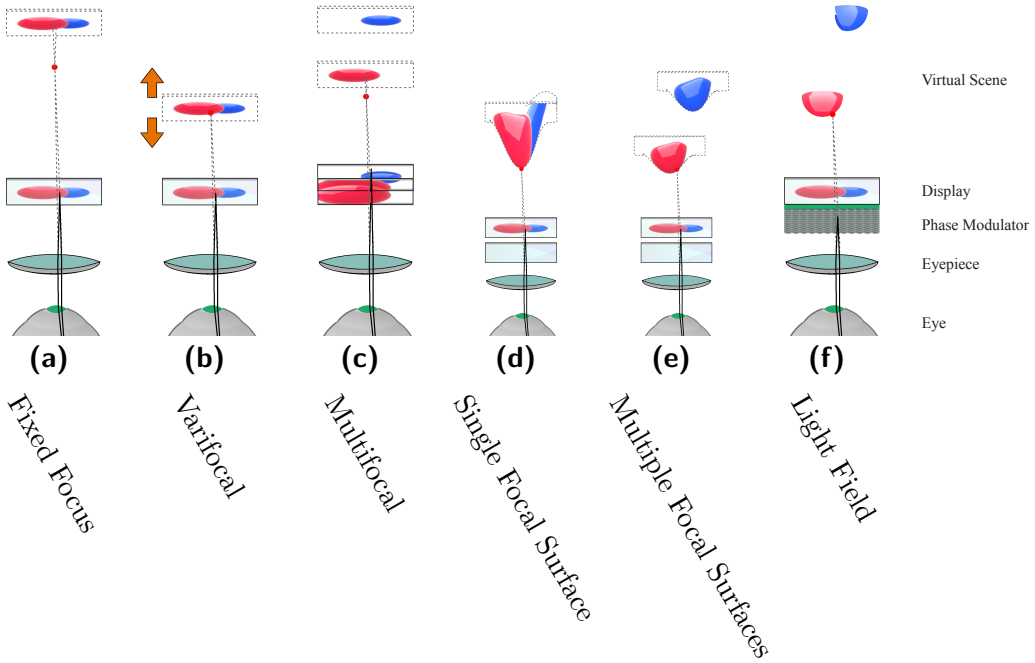


Figure 2.1. **Near-Eye Display Continuum:** Focal surface displays generalize the concept of manipulating the optical focus of each pixel on an HMD. Configurations (d,e) augment a fixed-focus HMD (a) with a programmable phase modulator placed between the eyepiece and display. (b) Varifocal HMDs use a globally addressed tunable lens. (c) Multifocal displays may use a high-speed tunable lens and display to create multiple focal planes. (f) In contrast, certain light field HMD concepts fall at the other end of this spectrum, using a finely structured phase modulator (a microlens array) placed near the display. (d) This chapter considers designs existing between these extremes in which a phase modulator locally adjusts the focus to follow the virtual geometry, generalizing varifocal and multifocal concepts. (e) Similar to multifocal displays, multiple focal surfaces can be synthesized with high-speed phase modulators and displays.

A multitude of "accommodation-supporting" HMD architectures have been proposed to depict correct or near-correct retinal blur, thereby mitigating VAC. As surveyed by Kramida [**68**], these architectures are distinguished by the fidelity to which they synthesize retinal blur. At one end of the spectrum are designs that effectively extend the user's

depth of focus (DOF), allowing the virtual image to remain sharp, independent of the user's accommodative state. This includes varifocal displays that dynamically adjust the focus of the HMD, contingent on the detected eye gaze. While addressing blurred vision induced by VAC, such displays cannot correctly depict retinal blur; instead, blur can only be synthetically rendered. At the other end of the spectrum are designs that correctly reproduce the optical wavefront of a physical scene, including holographic displays and, in certain circumstances, light field displays. As reported by Kramida, such displays are not yet practical, due to the limited resolution, field of view, and image quality achievable with today's hardware. As a result, a third category of accommodation-supporting HMDs is under active investigation: those that create "near-correct" (approximated) retinal blur. A visual overview of accommodation-supporting architectures is shown in Figure 2.1.

Approximate blur for multifocal displays has been studied extensively. Multifocal displays consist of a superposition of multiple virtual images spanning a range of focal depths. The first multifocal prototype employed three separate display elements per eye, prohibiting head-mounted configurations [**6**]. As reviewed in Section 2.2.1, multifocal HMDs increasingly exploit time-multiplexed presentation, wherein a single high-refresh-rate display and a fast varifocal element sequentially address the image planes [**76**]. Despite wide investigation, multifocal displays continue to present numerous practical challenges. First, as established by MacKenzie et al. [**80**], focal plane separation must be as close as 0.6 diopters to correctly stimulate accommodation. Thus, five focal planes are required to span a working range of 3.0 diopters (supporting virtual scenes extending from 33 cm to optical infinity). In practice, flickering is likely perceived with this many focal planes,

due to the refresh rates of microdisplays currently used in HMD designs. Second, as investigated by Narain et al. [**96**], the lateral spatial resolution of virtual objects presented between focal planes is restricted, demanding yet more planes to achieve the desired 3D resolution. Recently, Wu et al. [**142**] proposed dynamically adapting focal plane separations to virtual content, effectively combining the varifocal and multifocal concepts to reduce the number of required image planes.

This chapter expands on the concept of an adaptive multifocal display, introducing *focal surface displays* in which a *spatially-addressable* phase modulator is inserted into an otherwise conventional HMD. The phase modulator shapes focal surfaces to conform to the scene geometry, unlike multifocal displays with fixed, typically planar, focal surfaces. We produce a set of color images which are each mapped onto a corresponding focal surface using rendering-based optimization. Visual appearance is rendered by tracing rays from the eye through the optics and accumulating the color values for each focal surface. Our algorithm sequentially solves for first the focal surfaces, given the target depth map, and then the color images. Focal surfaces are adapted by nonlinear least squares optimization, minimizing the distance between the nearest depicted surface and the scene geometry. The color images, paired with each surface, are determined by linear least squares methods. Using databases of natural and rendered scenes, we demonstrate that focal surface displays depict more accurate retinal blur, with fewer multiplexed images, than prior multifocal displays, while maintaining high resolution throughout the user's accommodative range. Through focal surface displays, we aim to extend the technological development path beyond prior varifocal and multifocal concepts, opening a new point in the design tradespace of accommodation-supporting HMDs.

2.1.1. Contributions

- We introduce focal surface displays, capable of depicting near-correct focus cues in head-mounted displays, and assess capabilities relative to prior accommodation-supporting HMDs, including related multifocal architectures.
- We introduce an optimization framework that decomposes target focal stacks and depth maps into one or more pairs of focal surfaces and color images. Our pipelined approach finds focal surfaces through nonlinear least squares optimization and color images by linear least squares methods.
- Through a first-order optical analysis, we describe the optimal construction of focal surface displays, assessing trade-offs between resolution, field of view, and depth of focus. Furthermore, we identify the benefit of extending the SLM phase modulation range to enable high-resolution display.
- We implement a binocular focal surface display prototype, employing one LCOS spatial light modulator and one OLED panel per eye. We assess its experimental performance in relation to geometric optical simulations.

2.2. Related Work

Focal surface displays draw on insights spanning accommodation-supporting HMDs, goal-based caustics, as well as freeform and adaptive optics. Our prototype and development of this architecture is largely presented with regard to VR HMDs. However, as discussed in Section 2.6, there is a clear extension to certain augmented reality (AR) systems, particularly projector-based configurations. For reviews of existing VR and AR designs, consult Cakmakci and Rolland [**17**] and Kress and Starner [**69**], respectively.

2.2.1. Accommodation-Supporting Displays

An HMD can be evaluated relative to standard criteria, including resolution, field of view (FOV), and eye box dimensions. Today's VR HMDs exhibit FOVs around 100 degrees with resolutions better than 5 cycles per degree (cpd). Emerging designs must ultimately support such specifications and beyond. Accommodation-supporting HMDs may further be evaluated in regard to their depth of focus (DOF) and the fidelity to which retinal blur is reproduced. Many designs require eye tracking, which introduces concerns about reliability that must be weighed against others. Additionally, emerging HMDs increasingly exploit adaptive optics, particularly tunable lenses (see Figure 2.1). Some schemes may leverage computational display concepts and can be judged on additional axes, including image quality (which may be limited due to compression artifacts) and the failure modes and computational complexity resulting from content-dependent optimization. In this section, we review prior accommodation-supporting HMDs relative to these criteria, showing that focal surface displays expose a new, promising point in the design tradespace.

2.2.1.1. Monovision Displays

Marran and Schor [**87**] provide a prior survey of accommodation-supporting HMDs. One configuration they assess is that of *monovision*, wherein the virtual image distance differs between the eyes. This configuration is inspired by a related optometric application by which presbyopia is addressed by placing the focus of one eye closer than the other. Recently, Johnson et al. [**58**] and Konrad et al. [**66**] assessed the performance of monovision HMDs. The former study found viewer comfort and visual performance did not improve, whereas the latter found some benefit. However, not all viewers prefer or eventually adapt to monovision, motivating the need for more widely applicable methods.

2.2.1.2. Varifocal Displays

Varifocal HMDs augment a conventional design with two components: an eye tracker and a variable focusing element. Eye tracking is used in a feedback system to dynamically set the tunable lens focus to match vergence, thus ensuring VAC is minimized. Shiwa et al. [**127**] first demonstrated this concept using actuated lenses on an optical bench. Sugihara et al. [**131**] created the first varifocal HMD, wherein the display translated rather than a lens. Liu et al. [**76**] and Konrad et al. [**66**] demonstrated varifocal displays using electronically tunable lenses. Recently, Dunn et al. [**27**] and Padmanaban et al. [**107**] presented varifocal displays with integrated eye tracking.

Varifocal displays may reduce VAC, but they cannot directly reproduce retinal blur. Gaze-contingent depth of field (DOF) rendering must be applied. Hillaire et al. [**43**] and Mantiuk et al. [**83**] conclude that DOF blur is preferred with 2D displays. Duchowski et al. [**26**] found that visual discomfort was reduced when viewing a stereoscopic display with

gaze-contingent DOF blur, albeit with a statistically weak dislike for this blurring. Our interpretation of this result is that it highlights the limitations of rendered blur: latency and eye-tracking errors may create distracting artifacts, motivating the development of accommodation-supporting HMDs that support near-correct, rather than rendered, retinal blur. Perceptual studies by Maiello et al. [**81**] and Zannoli et al. [**145**] have found that synthetically rendered blur may not assist depth perception to the same degree as near-correct retinal blur.

2.2.1.3. Accommodation-Invariant (EDOF) Displays

For HMDs, the analogue of a pinhole camera is a Maxwellian view: a point light source is focused on the viewer's pupil, with an amplitude SLM modulating a focused image on the retina [**16**]. Von Waldkirch et al. [**140**] apply this principle to HMDs, showing a trade-off between DOF and resolution. Due to diffraction, DOF cannot extend above three diopters without restricting resolution below 30 cpd (i.e., 20/20 vision) [**52**]. Following Kramida [**68**], FOV is limited due to restricted eye movement.

Maxwellian-view HMDs exhibit an accommodation-invariant response. In computational photography, this is known as extended depth of focus (EDOF) [**144**]. Von Waldkirch [**139**] applied EDOF to HMDs, rapidly varying focus with a tunable lens; however, deconvolution was not considered and, as a result, image contrast was reduced. More recently, Huang et al. [**50**] applied pre-filtering to a multilayer EDOF display, although contrast remained low. Even if image quality can be improved, accommodation-invariant HMDs still rely on rendered retinal blur.

2.2.1.4. Multifocal Displays

To our knowledge, Neil et al. [**100**] proposed, and demonstrated, the first multifocal HMD. As they describe, the concept is preceded by decades of research into volumetric displays [**12**]. Rolland et al. [**120**] proposed a closely related architecture, assessing that a 2.0-diopter DOF requires up to 27 planes. Even this may not be sufficient: measurements by Sprague et al. [**128**] find an average 40-year-old or younger individual can accommodate in excess of 4.0 diopters.

MacKenzie et al. [**80**] show that wider plane separations can correctly drive accommodation; however, maintaining high resolution between planes and extending DOF can only be achieved, currently, with additional adaptive optical elements [**142**]. Multifocal adaptive optics include ferroelectric liquid crystal (FLC) SLMs [**100, 79**], tunable lenses [**76, 66, 142**], and deformable mirrors (DMs) [**47**]. Focal surface displays leverage this trend for increasing electro-optic control, preparing for a future in which spatially-varying phase modulation is widely available.

Akeley et al. [**6**] first considered the optimal presentation of imagery across multiple focal planes, introducing the "linear blending" algorithm. Ravikumar et al. [**119**] assessed alternative algorithms, concluding that, of those available at the time, linear blending was preferred. More recently, Narain et al. [**96**] introduced "optimized blending" to directly optimize the through-focus image, enhancing occluding, semi-transparent, and reflective objects. In this work, we generalize optimized blending to support adaptive focal surfaces.

2.2.1.5. Retinal Scanning Displays

Rather than using comparatively large screens, retinal scanning displays (RSDs) directly sweep a point of light across the viewer's retina [**136**]. McQuaide et al. [**92**] modify RSDs to additionally modulate focus using a deformable mirror (DM). Unlike varifocal HMDs, focus can be adjusted—in theory—independently per pixel. This concept is a precursor to focal surface displays; however, to our knowledge, it was never fully realized: deformable mirrors exhibit a modulation rate three orders of magnitude too slow for per-pixel focus control. Correspondingly, McQuaide et al. only demonstrate simple line images, albeit over a continuously-varying 3.0-diopter DOF.

Focal surface displays significantly differ from accommodation-supporting RSDs. First, we provide an optimization framework to tailor focal surfaces that respects the constraints of current phase SLM technology. Second, our framework allows multiple focal surfaces, yielding near-correct depictions of occlusions. Third, we leverage work on optimized blending for multifocal displays to account for limitations of focal surface control. Fourth, we demonstrate the first fully realized embodiment with a binocular LCOS-based prototype capable of depicting natural scenes.

2.2.1.6. Light Field Displays

Volumetric displays inspired multifocal displays. Similarly, near-eye light field displays originate from the autostereoscopic community. Lanman and Luebke [**72**] first applied integral imaging to VR HMDs, with a closely related AR HMD developed by Hua and Javidi [**48**]. While depicting near-correct retinal blur, these prototypes exhibit low resolution, albeit while additionally depicting correct parallax across the eye box. Maimone et al. [**82**] and Huang et al. [**49**] introduced computational near-eye light field displays for AR and VR, respectively, based on amplitude-only SLM stacks (i.e., multilayer LCDs). Such displays confront practical resolution limits due to diffraction and compression artifacts. Our multilayer focal surface display does not exhibit a similar limit due to the comparatively high fill factor and lack of color filters with LCOS panels.

2.2.1.7. Holographic Displays

Decades of research into direct-view holography has laid the groundwork for near-eye applications [**14**]. Today's digital holographic displays synthesize accurate wavefronts, and therefore correct retinal blur, by controlled illumination of a diffractive element. Moon et al. [**94**] describe a recent holographic HMD, showing practical limits on FOV (less than 20 degrees), eye box dimensions (a few millimeters wide), and image quality (degraded due to speckle). Focal surface displays, which may incorporate similar phase modulators, fundamentally differ: incoherent illumination is produced by an emissive display, with subsequent modulation by a phase-only SLM that produces piecewise smooth modulations. Furthermore, such displays require minimal modification to existing VR HMDs.

2.2.2. Caustics, Freeform Elements, and Adaptive Optics

Focal surface displays also trace their origin to recent progress in computational fabrication and adaptive optics. In a closely related work, Damberg et al. [**22**] use a phase-only SLM to create a freeform adaptive lens for the purpose of high dynamic range (HDR) projection. Damberg et al. adapt prior research into *goal-based caustics*, wherein freeform lenses are fabricated to project images under controlled illumination [**110, 143**]. Phase-only SLMs have been similarly adopted by the computational display community, with Glasner et al. [**36**] and Levin et al. [**74**] demonstrating their application to light-sensitive multiview displays. To our knowledge, focal surface displays are the first application of phase SLMs to locally adapt the focus of an HMD.

2.3. Focal Surface Displays

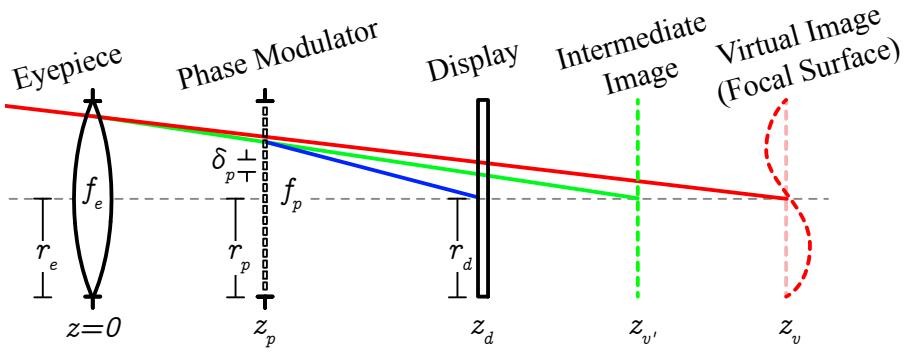


Figure 2.2. **Simplified Optical Diagram and Labels:** A focal surface display is created by placing a phase-modulation element between an eyepiece and a display screen. This phase element and the eyepiece work in concert as a spatially programmable compound lens, varying the apparent virtual image distance across the viewer's field of view.

A conventional VR HMD contains two primary optical elements: an eyepiece and an emissive display. This design delivers a single, fixed *focal surface*. As shown in Figure 2.2, a *focal surface display* adds a third element between the eyepiece and the display: a phase-modifying spatial light modulator (SLM). This SLM acts as a programmable lens with spatially varying focal length, allowing the virtual image of different display pixels to be formed at different depths. In this section, we present an optimization framework that decomposes a scene into one or more focal surfaces, and corresponding color images, to reproduce retinal blur consistent with natural scenes.

Inspired by related multifocal displays, we generalize our formulation to support multiple focal surfaces (as achieved by time multiplexing). The inputs to our algorithm are a depth map, representing the scene geometry, and a focal stack, modeling the variation of retinal blur with changes in accommodation. Both inputs are rendered from the

perspective of the viewer's entrance pupil. The outputs are k phase functions $\phi_1, \ldots, \phi_k$ and color images $c_1, \ldots, c_k$, to be presented by the SLM and underlying display, respectively. Ideally, we would jointly optimize the phase functions and color images. Because this results in a large, nonlinear problem, we introduce approximations that ensure the algorithm is computationally tractable. First, in Section 2.3.1, we decompose the target depth map into a set of smooth focal surfaces. Second, in Section 2.3.2, we optimize the phase functions to approximate these focal surfaces. Finally, in Section 2.3.3, we optimize the color images to reproduce the target focal stack.

While our formulation allows multiple focal surfaces, a single surface achieves similar retinal blur fidelity as prior multifocal displays. As with other computational displays, focal surface displays offer a trade-off between system complexity (the need for time multiplexing) and image quality (suppression of compression artifacts).

2.3.1. Approximating Depth Maps with Focal Surfaces

Given a target virtual scene, let $\hat{S}(\theta_x, \theta_y)$ be the depth (in diopters) along each viewing angle $(\theta_x, \theta_y) \in \Omega_\theta$, for *chief rays* passing through the center of the viewer's pupil and with Ω_θ being the discrete set of retinal image samples. If phase SLMs could render focal surfaces with arbitrary topology, then no further optimization would be required. As presented in Section 2.3.2, this is not the case: practically-realizable focal surfaces are required to be smooth. Correspondingly, we develop the following method for decomposing a depth map into k smooth focal surfaces $S_1, \ldots, S_k$.

For every viewing angle (θ_x, θ_y) we desire at least one focal surface $S_i(\theta_x, \theta_y)$ to be close to the target depth map $\hat{S}(\theta_x, \theta_y)$. If this occurs, then every scene element can be

depicted with near-correct retinal blur, as light from the underlying display will appear to originate from the correct scene depth. (As established by Narain et al. [**96**], optimized blending methods still benefit the rendition of occluding, semi-transparent, and reflective objects.) Given this goal, we formulate the following optimization problem.

$$\begin{aligned} &\min_{S_1,\ldots,S_k} \sum_{(\theta_x,\theta_y)\in\Omega_\theta} \left(\min_i |\hat{S}(\theta_x,\theta_y) - S_i(\theta_x,\theta_y)|\right)^2 \\ &\text{s.t.} \left(\frac{\partial^2 S_i}{\partial x^2}\right)^2 + \left(\frac{\partial^2 S_i}{\partial x \partial y}\right)^2 + \left(\frac{\partial^2 S_i}{\partial y^2}\right)^2 < \epsilon \end{aligned} \tag{2.1}$$

As analyzed in Section 2.3.2, synthesizing a focal surface using phase function ϕ may introduce some optical aberrations. Observationally, we find aberrations are minimized if the second derivatives of the focal surface are small. This observation is reflected by the bound constraints in our optimization problem. Note, however, that no explicit bound constraints are imposed on the optical powers S_i of the focal surfaces. This would appear to contradict our derivation of the minimum realizable focal length of a given phase SLM (see Section 2.3.2). Rather than adding these constraints directly, we simply truncate the target depth map $\hat{S}$ to the realizable range.

We apply nonlinear least squares (NLS) to solve Equation 2.1, which has high-quality implementations and scales to large problem sizes [**4**]. Note that our objective involves the nonlinear residual $g_{\theta_x,\theta_y}(S) := \min_i |\hat{S}(\theta_x,\theta_y) - S_i(\theta_x,\theta_y)|$ for each pixel (θ_x,θ_y). This residual is not differentiable, which is a problem for NLS. However, a close approximation is obtained by replacing the min with a "soft minimum" (soft-min), with the following

definition [**19**]:

$$\widetilde{g}_{\theta_x,\theta_y}(S) = -t \log \sum_i e^{-|\hat{S}(\theta_x,\theta_y) - S_i(\theta_x,\theta_y)|/t} \tag{2.2}$$

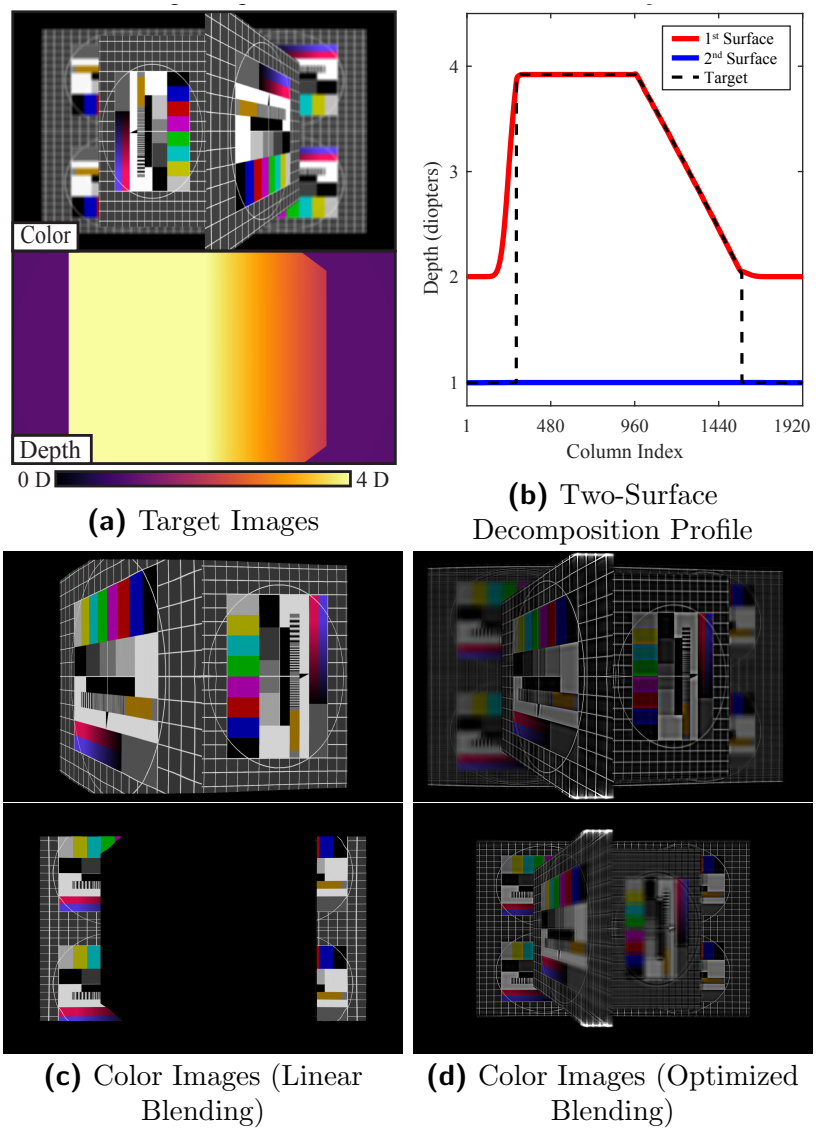
where t is a conditioning parameter to be tuned for a given application. Note that $\widetilde{g}$ is continuously differentiable and closely approximates g as $t \to 0$, with $|\widetilde{g}(\theta_x, \theta_y) - g(\theta_x, \theta_y)| \leq t \log k$.[1]

Applying Equation 2.2 to Equation 2.1, and re-expressing bound constraints as soft constraints, yields the following NLS problem:

$$\min_{S_1,\ldots,S_k} \sum_{(\theta_x,\theta_y)} (\widetilde{g}_{\theta_x,\theta_y}(S))^2 + \gamma \sum_{i,(\theta_x,\theta_y)} \|\partial^2 S_i(\theta_x, \theta_y)\|^2 \tag{2.3}$$

where $\partial^2 S_i(\theta_x, \theta_y)$ is the vector of second partial derivatives of S_i at (θ_x, θ_y) and γ is a weighting parameter. See Figures 2.3 and 2.4 for examples of applying this focal surface decomposition algorithm. As shown, locally adapted smooth focal surfaces offer an efficient representation of natural and artificially rendered depth maps.

[1]Note that when computing a soft-min, for numerical stability it is important to use the method described by Cook [**19**], wherein the minimum value is subtracted before evaluating the exponential functions.



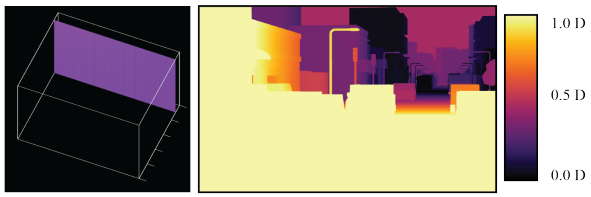
(a) Target Images

(b) Two-Surface Decomposition Profile

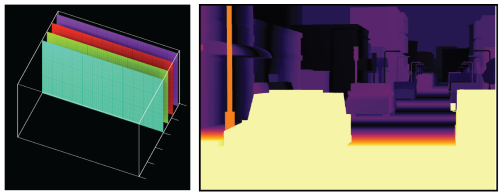
(c) Color Images (Linear Blending)

(d) Color Images (Optimized Blending)

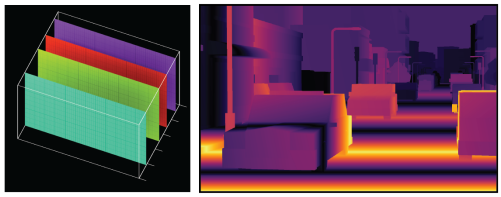
Figure 2.3. **A Toy Scene Illustration:** A focal surface decomposition for a simple scene, containing: a background fronto-parallel plane at 1.0 diopters, a foreground fronto-parallel plane at 4.0 diopters, and a slanted plane spanning 2.0 to 4.0 diopters. (a) A single image from the target focal stack and target depth map. (b) A two-surface decomposition is compared to the target depth map for a profile taken along the middle row of the target imagery. (c) The color images associated with each focal surface are shown, using the linear blending method of Akeley et al. [**6**]. (d) The color images associated with each focal surface, using the rendering-based optimized blending algorithm presented in Section 2.3.3.



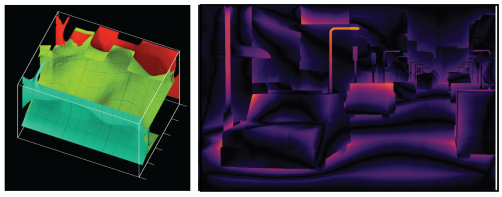
(a) Fixed Focus



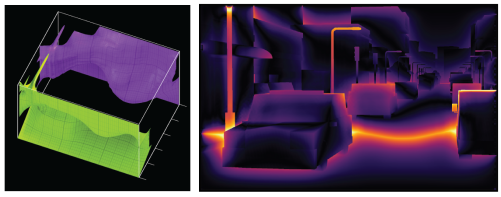
(b) Four Fixed Planes



(c) Four Adaptive Planes



(d) Three Focal Surfaces



(e) Two Focal Surfaces

Figure 2.4. **Depth Error Assessment:** Focal surface displays achieve lower depth map approximation errors, using less time multiplexing, than prior multifocal methods. The left column depicts depth decompositions ranging from 0.0 to 5.0 diopters, abbreviated "D". The right column depicts the resulting depth map approximation errors in diopters. For a fixed focus design, the virtual image is positioned at 0.5 D. Following Narain et al. [**96**], the fixed multifocal display employs four planes evenly spaced from 0.2 D to 2.0 D. The adaptive multifocal display and the focal surface display are optimized using k-means clustering, following Wu et al. [**142**], and the methods in Sections 2.3.1 and 2.3.2 to position planes across a 5.0 D span, respectively. Focal surface displays show significantly fewer depth errors, with errors decreasing as more surfaces are used. (Source imagery courtesy Unity Asset Store publisher "VenCreations.")

2.3.2. Synthesizing Focal Surfaces with Phase SLMs

Provided a set of focal surfaces S_i, the next stage in our pipeline requires solving for a set of phase functions ϕ_i to practically achieve them. To solve this problem, we first review the optical properties of phase SLMs and then present our phase optimization procedure.

2.3.2.1. Optical Properties of Phase SLMs

Variations in optical path length through a lens cause refraction. Similarly, differences in phase modulation across an SLM result in diffraction. Simulation of light propagation through a high-resolution SLM via wave optics modeling is currently computationally infeasible, but one can approximate these diffractive effects using geometric optics, similar to Glasner et al. [**36**] and Damberg et al. [**22**]. (Laude [**73**] provides additional details regarding the operation of phase SLMs.) We denote SLM locations by (p_x, p_y), with Ω_p being the discrete set of SLM pixel centers. Optical rays intersecting an SLM are redirected depending on the phase ϕ. For small angles (i.e., under the paraxial approximation), the deflection is proportional to the gradient of ϕ (see [**138**], Equation 6.1). If an incident ray has direction vector $(x, y, 1)$ and intersects the SLM at (p_x, p_y), then the outgoing ray has direction vector

$$\left(x + \frac{\lambda}{2\pi}\frac{\partial\phi}{\partial x}(p_x, p_y), y + \frac{\lambda}{2\pi}\frac{\partial\phi}{\partial y}(p_x, p_y), 1\right) \tag{2.4}$$

where λ is the illumination wavelength. Thus, if ϕ is a linear function, then the SLM operates as a prism, adding a constant offset to the direction of every ray. (Note that we assume monochromatic illumination in this derivation, with practical considerations for broadband illumination sources presented later in Section 2.6.) An SLM may also act

as a thin lens (see [**138**], Equation 6.8) by presenting a quadratically-varying phase as follows.

$$\phi(p_x, p_y) = -\frac{\pi}{\lambda f}(p_x^2 + p_y^2) \tag{2.5}$$

Note that these optical properties are local. The deflection of a single ray only depends on the first-order Taylor series of the phase (i.e., the phase gradient) around the point of intersection with the SLM. Similarly, the change in focus of an ϵ-sized bundle of rays intersecting the SLM only depends on the second-order Taylor series. Specifically, if the Hessian of ϕ at a point (p_x, p_y) is given by

$$H_\phi(p_x, p_y) = -\frac{2\pi}{\lambda f}\mathbf{1} \tag{2.6}$$

where $\mathbf{1}$ is the 2×2 identity matrix, then the ϵ-sized neighborhood around (p_x, p_y) functions as a lens of focal length f (i.e., Equation 2.6 is the Hessian of Equation 2.5).

To this point, we have allowed the phase to be any real-valued function. In practice, an SLM will have a bounded range, typically from $[0, 2\pi]$. Phases outside this range are "wrapped", modulo 2π. In addition, achievable phase functions are restricted by the Nyquist limit. The phase can change by no more 2π over a distance of $2\delta_p$, where δ_p is the SLM pixel pitch. Following Voelz [**138**], these factors bound the minimum focal length f such that $|f| \geq \frac{2r_p\delta_p}{\lambda}$, where r_p is the radius of the SLM (taken diagonally).

2.3.2.2. Adapting Focal Surfaces with Phase SLMs

With this paraxial model of an SLM, we can determine a phase function ϕ to best realize a given target focal surface d. First, we must determine how the SLM focal length f_p (synthesized via Equation 2.5) affects a focal surface distance z_v. As indicated in Figure 2.2, the SLM acts within a focal surface display that is parameterized by the eyepiece distance (z=0), the SLM distance z_p, and the display distance z_d. Ignoring the eyepiece, the SLM produces an intermediate image of the display at distance $z_{v'}$. This intermediate image is transformed to a virtual image of the display, located at z_v, depending on the eyepiece focal length f_e. These relations are compactly summarized by application of the thin lens equation (see [**138**], Equation 7.1):

$$\frac{1}{f_p} = \frac{1}{z_{v'} - z_p} + \frac{1}{z_d - z_p} \text{ and } \frac{1}{f_e} = \frac{1}{z_v} - \frac{1}{z_{v'}} \tag{2.7}$$

By casting viewing ray (θ_x, θ_y) from the viewer's pupil to the SLM, and then by applying Equation 2.7, a target focal length f_p can be assigned for each SLM pixel (p_x, p_y) to create a virtual image at the desired focal surface depth. To realize this focal length, Equation 2.6 requires a phase function ϕ with the Hessian

$$H_\phi(p_x, p_y) = -\frac{2\pi}{\lambda f(p_x, p_y)}\mathbf{1} \tag{2.8}$$

There may be no ϕ that exactly satisfies this expression. In fact, such a ϕ only exists when f is constant and ϕ is quadratic (i.e., the phase represents a uniform lens).

Since Equation 2.8 cannot be exactly satisfied, we solve the following linear least squares problem to obtain a phase function ϕ that is as close as possible:

$$\min_{\phi} \sum_{(p_x,p_y)\in\Omega_p} \|\hat{H}[\phi](p_x,p_y) - \frac{-2\pi}{\lambda f(p_x,p_y)}\mathbf{1}\|_F^2 \tag{2.9}$$

where $\|\cdot\|_F^2$ is the Frobenius norm and where $\hat{H}[\,\cdot\,]$ is the discrete Hessian operator, given by finite differences of ϕ. Note that the phase function ϕ plus any linear function $a+bx+cy$ has the same Hessian H, so we additionally constrain $\phi(0,0) = 0$ and $\nabla\phi(0,0) = 0$.

2.3.2.3. Representing Natural Scenes

Applying focal surface displays requires answering a key question: can natural scenes be well-approximated by smooth focal surfaces S_i, and if so, how many surfaces are required to accurately reproduce retinal blur? Following Wu et al. [**142**], we first consider the Middlebury 2014 dataset from Scharstein et al. [**122**], containing 33 depth maps from real-world environments. In Figure 2.5, we compare our depth approximation error with prior fixed and adaptive multifocal displays. *A single focal surface, as produced by our method, more closely follows scene geometry than prior fixed-focus multifocal displays (with four planes) and adaptive multifocal displays (with three planes).* In practice, two focal surfaces appear to be an effective representation, allowing occlusions, transparencies, and reflections to be captured, so long as two dominant surfaces are visible in each viewing direction. In this manner, our focal surface display technique significantly reduces the number of required surfaces and contributes to the practicality of time-multiplexed multifocal displays.

Relying solely on the Middlebury dataset could provide a misleading conclusion, as the depths in that collection only span an average range of 1.0 diopters. As a result, we created our own synthetically rendered database to span a range of 4.0 diopters, on average. Resulting depth approximation errors are shown in Figure 2.6. Note that focal surface displays continue to outperform prior multifocal displays.

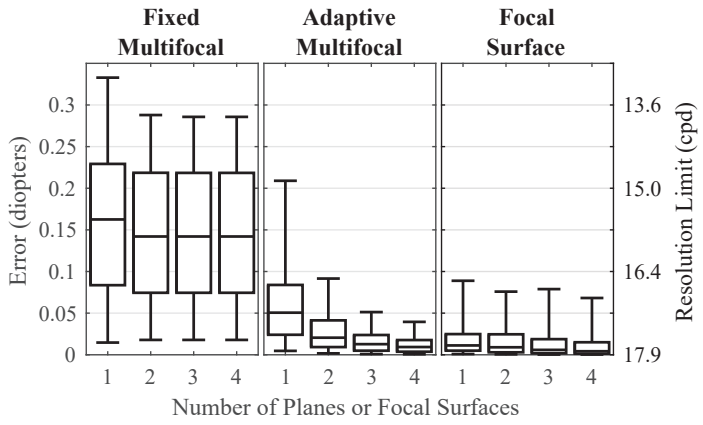


Figure 2.5. **Depth Error Assessment, Middlebury Dataset:** Focal surface displays represent natural scene depths with few image components. Box plots compare the depth map errors $g_{\theta_x,\theta_y}(d)$ using the denoted methods with the Middlebury 2014 dataset [**122**]. The bottom and top of the whiskers indicate the 5^{th} and 95^{th} percentiles, respectively. The bottom, middle, and top of the boxes represent the 1^{st}quartile, the median, and the 3^{rd} quartile, respectively. Focal surface displays produce fewer depth errors, especially when fewer planes are used.

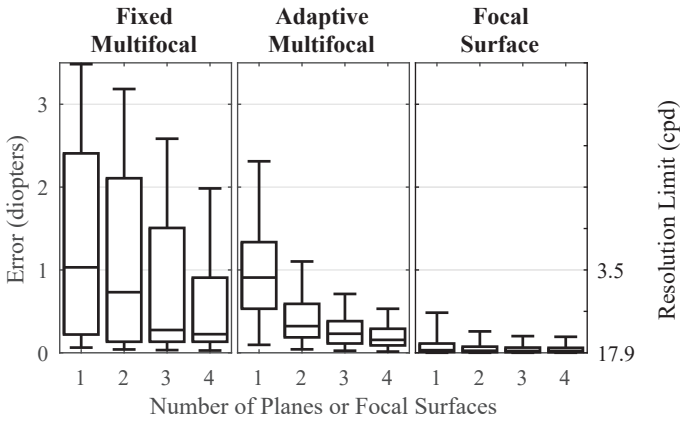


Figure 2.6. **Depth Error Assessment, Unity Dataset:** We repeat the assessment of Figure 2.5, but with the database of rendered scenes described in Section 2.3.2. Note that the trends are repeated, but due to the larger depth ranges in this database, additional virtual image surfaces are required with prior fixed and adaptive multifocal displays.

2.3.2.4. Focusing Errors Limit Visual Acuity

Reducing the number of planes, as with prior multifocal displays, is often achieved by increasing their separation. As noted by Narain et al. [**96**], this comes at the cost of reducing the maximum-supported resolution (measured in cycles per degree). For example, Narain et al. estimate that contrast falls below 50% for 11 cpd spatial frequencies or higher, with a plane separation of 0.6 diopters. For context, that would imply that a conventional fixed-focus multifocal display could not achieve resolutions, throughout the supported accommodation range, exceeding more than twice that of modern VR HMDs.

Based on the statistics in Figure 2.5 and 2.6, both multifocal and focal surface displays should achieve focusing errors less than 0.12 diopters, if operated over an appropriate accommodation range with a sufficient number of components. Following Kotulak and Schor [**67**], with this fidelity of focus, such systems should drive accommodation correctly.

In this circumstance, focusing errors can be directly translated to a spatial frequency (resolution) limit via the modulation transfer function (MTF) of the human eye. Narain et al. apply a similar analysis to assess contrast limits. In the technique described in this chapter, we apply the 35% through-focus MTF of the human eye, as estimated by Villegas et al. [**137**], to convert focusing errors to spatial frequency limits in Figures 2.5 and 2.6. Note that, with focal surface displays, a significantly higher resolution limit is predicted, opening a path to high-resolution HMDs, unlike prior multifocal displays.

2.3.2.5. Additional Metrics for Focal Surface Optimization

The paraxial approximation was applied to the phase optimization in Equation 2.9. However, a different criterion could be employed: find the phase ϕ minimizing the distance between the minimum-spot-size measured focus and the true depth d, summed over all angles Ω_θ. This metric accounts for higher-order aberrations (as it is inspired by similar analysis performed by optical design software), although it does not account for scene content (one may not care what the focus is in regions of uniform color). This metric requires evaluating the forward rendering operator from Section 2.3.3 and, as a result, would again produce a large nonlinear optimization problem — motivating our adoption of the paraxial model that, in practice, produces accurate focal surfaces. Efficiently leveraging the minimum-spot-size metric is a promising path for future work.

2.3.3. Optimized Blending with Focal Surfaces

Having determined k phase functions ϕ_i, corresponding to focal surfaces S_i, the last stage in our pipeline determines color images I_i, shown on the underlying display, to reproduce the target focal stack. This focal stack is represented by a set of l retinal images $\hat{I}r_1, \ldots, \hat{I}_l$. For this purpose, we generalize the optimized blending algorithm of Narain et al. [**96**]. In this section, we first describe a ray-traced model of retinal blur. Afterward, this model is applied to evaluate the forward and adjoint operators required to solve the linear least squares problem representing optimized blending.

2.3.3.1. Modeling Retinal Blur with Ray Tracing

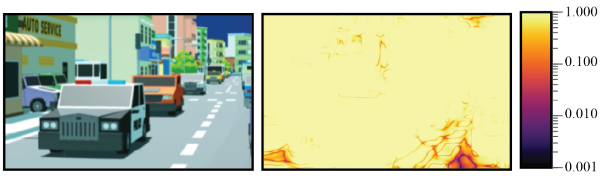
An optical ray is traced through our system under a geometric optics model. Following Figure 2.2, each ray originates at a point within the viewer's pupil. The ray then passes through the front and back of the eyepiece, the SLM, and then impinges on the display. At the eyepiece surfaces, rays are refracted using the radius of curvature of the lens, its optical index, and the paraxial approximation. Equation 2.4 models light transport through the SLM. Each ray is assigned the color interpolated at its coordinate of intersection with the display. We denote locations on the display by (q_x, q_y) and the set of display pixel centers by Ω_q. Note that any rays that miss the bounds of the eyepiece, SLM, or display are culled (i.e., are assigned a black color).

To model retinal blur, we accumulate rays that span the viewer's pupil, which we sample using a Poisson distribution. In this manner, we approximate the viewer's eye as an ideal lens focused at a depth z which changes depending on the viewer's accommodative state. For each chief ray (θ_x, θ_y) and depth z, we sum across a bundle of rays $R_{\theta_x,\theta_y,z}$ from

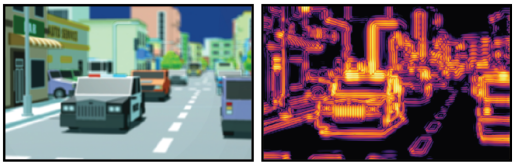
the Poisson-sampled pupil. This produces an estimate of the retinal blur when focused at a depth z. We define these preceding steps as the *forward operator* $\mathbf{T}_{z,\phi}(I)$, which accepts a phase function ϕ and color image I and predicts the perceived retinal image when focused at a distance z.

2.3.3.2. Depicting Focal Stacks with Optimized Blending

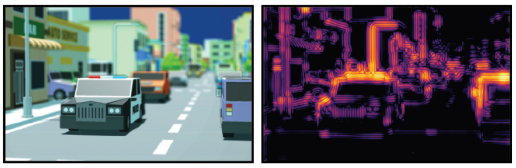
For a fixed phase function ϕ and accommodation depth z, the forward operator $\mathbf{T}_{z,\phi}(I)$ is linear in the color image I. The rendering operators $\mathbf{T}_{z,\phi_i}(I_i)$ combine additively, so our combined forward operator, representing viewing of multiple-component focal surface displays, is $\mathbf{T}_z(I_1, \ldots, I_k) = \sum_i \mathbf{T}_{z,\phi_i}(I_i)$. We can concatenate the forward renders for multiple accommodation depths $z_1, \ldots, z_l$ to estimate the reconstructed focal stack, with corresponding linear operator $\mathbf{T} = [\mathbf{T}_{z_1}; \ldots; \mathbf{T}_{z_l}]$. The forward operator, for a given set of color images c, gives the focal stack r that would be produced on the retina — minimizing $\|\mathbf{T}(I) - \hat{I}\|^2$ gives the color image best approximating the desired focal stack. We have already given an efficient algorithm for computing $\mathbf{T}_{z,\phi}$. Its transpose, mapping retinal image samples to display pixels, can be similarly evaluated with raytracing operations with accumulation in the color image I rather than the retinal image $\hat{I}$. In conclusion, these forward and adjoint operators are applied with an iterative least squares solver. (For implementation details, see Section 2.5.2.) Results of our full optimization pipeline are shown in Figure 2.7.



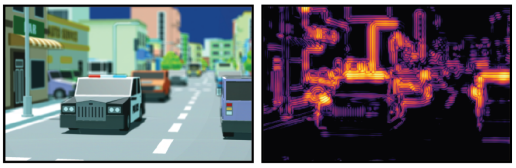
(a) Fixed Focus - $Q_{MOS} = 62.2$



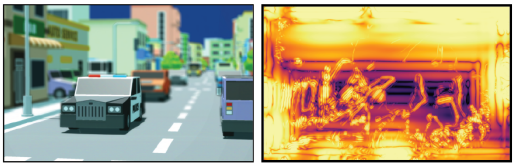
(b) Four Fixed Planes - $Q_{MOS} = 75.4$



(c) Four Adaptive Planes - $Q_{MOS} = 75.2$



(d) Three Focal Surfaces - $Q_{MOS} = 75.6$



(e) Two Focal Surfaces - $Q_{MOS} = 75.4$

Figure 2.7. **Perceptual Assessment:** Focal surface displays depict near-correct retinal blur with fewer virtual image surfaces than prior multifocal architectures. Following Figures 2.4–2.6, focal surface displays produce virtual images that more closely align with the scene geometry. As a result, sharply focused imagery can be obtained throughout the scene, reducing focusing errors occurring with prior fixed and adaptive multifocal displays. In this figure, we quantitatively assess the focal stack reproduction error following the method of Narain et al. [**96**]: the right column depicts the maximum per-pixel probability of detecting a difference between the target and reconstructed focal stacks, as quantified using the HDR-VDP-2 metric [**84**]. The corresponding quality predictor of the mean opinion score (MOS) is listed along the bottom. Note that focal surface displays achieve similar fidelity as prior adaptive multifocal displays, although with fewer virtual image surfaces. (Source imagery courtesy Unity Asset Store publisher "VenCreations.")

2.4. Designing Focal Surface Displays

In designing standard VR HMDs, there is a direct trade-off between field of view and resolution, which is largely determined by the placement of the display, the size and resolution of this display, and by the focal length of the eyepiece. For focal surface displays, there is a similar trade-off between the position of the SLM and the depth of focus (i.e., the supported accommodation range). In this section, we evaluate these trade-offs in terms of three metrics: field of view, depth of focus (DOF), and the degree of optical aberrations.

The field of view of a focal surface display is limited by the smallest of the display, the SLM, or the eyepiece (as appearing to the viewer). Wide eyepieces and displays are commonly available, so SLM dimensions currently limit the FOV. Ignoring variation with eye relief, the FOV is given by the angle subtended by the magnified SLM, or $2 \arctan \frac{r_p}{z_p}$, where r_p and z_p are the SLM radius and distance from the eyepiece, respectively. Thus, FOV is maximized by moving the SLM closer to the eyepiece.

Following Section 2.3.2, the SLM focal length is bounded such that $|f_p| \geq \frac{2r_p\delta_p}{\lambda}$. Substituting this range into Equation 2.7 gives a nonlinear expression mapping SLM focal length f_p and position z_p to the virtual image depth, and as such, bounds the depth of focus. The resulting trade-off between DOF and the system design parameters is illustrated in Figure 2.8: depth of focus for a given lens position is the difference in contour values between the red constraints. From this analysis, we conclude that DOF increases as we move the SLM closer to the eyepiece.

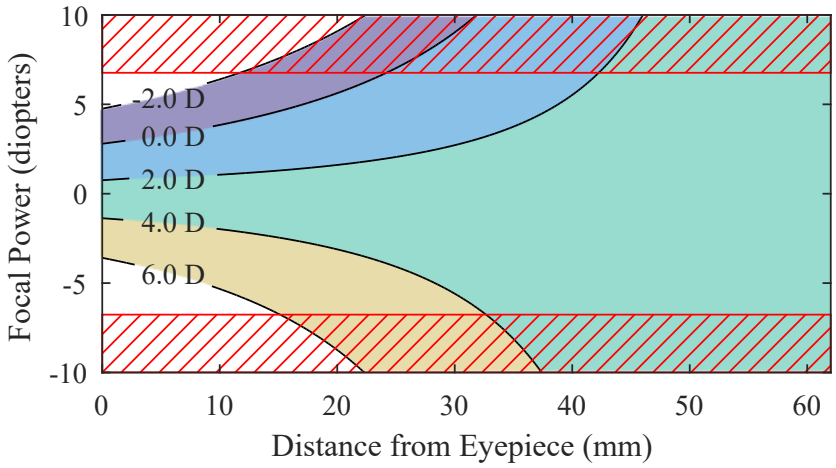


Figure 2.8. **Design Trade Space:** The accommodation range of a focal surface display depends critically on the SLM placement. Here we denote, via the labeled plot contours, the virtual image distance z_v achieved with an SLM, when used to represent a lens of focal length f_p and positioned a distance z_p from the eyepiece. Red lines indicate focal lengths beyond the dynamic range of the SLM. Note that these numbers correspond with the prototype described in Section 2.5.1.

Our final design metric is to minimize optical aberrations. As presented in Section 2.3.2, our method for generating phase functions optimizes phase curvature within small neighborhoods (since it is based on the discrete Hessian operator, which we evaluate using a 3×3 window). To estimate focus at angle (θ_x, θ_y) using our more accurate minimum-spot-size metric, we cast all rays in the bundle $R_{\theta_x,\theta_y,z}$ leaving the pupil. These rays intersect the SLM in a connected region P (i.e., the "circle of confusion"). The extent to which the rays intersect at a single point on the display depends on how close to quadratic the phase function is throughout all of P, not just the Hessian at a single point. It is easier to achieve this condition if the circle of confusion (i.e., P) is small, because the second-order Taylor series (i.e., the Hessian) is a better approximation in a

small neighborhood. The size of P is linearly proportional to the distance between the SLM and the display. We conclude that, for aberration control, we desire the SLM to be as close to the *display* as possible.

In summary, minimizing aberrations encourages moving the display in the opposite direction as required to increase DOF and FOV. As with all optical systems, the designer must balance between these trade-offs. For our prototype, we positioned the SLM as close to the display as possible, while supporting accommodation from 0.0 to 4.0 diopters. In practice, the hardware constrains the SLM position due to the volume occupied by the beamsplitter. Similarly, selecting from catalog lenses and SLMs limits the focal length, the SLM pixel pitch, and the SLM dimensions. Thus, only certain points in this design tradespace were readily accessible. However, the DOF of our prototype remains comparable to prior accommodation-supporting display prototypes.

2.5. Implementation and Results

A prototype is necessary to demonstrate the fundamental concepts presented in the preceding sections, as well as to identify practical limitations encountered with current-generation phase modulation hardware. In this section, we describe our hardware and software choices, and we evaluate the resulting experimental performance.

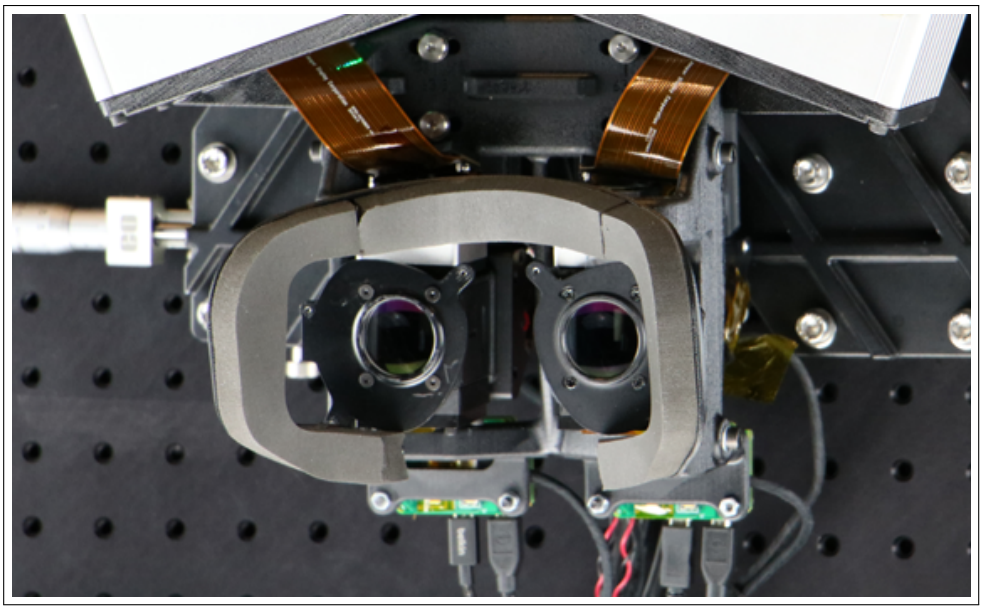
2.5.1. Hardware

Our prototype largely uses off-the-shelf optical and mechanical components, augmented with a handful of 3D-printed parts. The optical path begins, as shown in Figure 2.9b, with an eMagin WUXGA 1920×1200 60 Hz color OLED display, addressed via an MRA Digital HDMI driver board. The OLED is covered with an Edmund Optics 88-087 left-handed circular polarizer to suppress stray light reflections. Illumination from the display next encounters a Thorlabs 50:50 non-polarizing beamsplitter. The light reflected by the beamsplitter immediately impinges on a "beam dump" (i.e, a felt-covered, light-absorbing surface). Note that an eye tracking camera could be fitted to this side of the beamsplitter, as it allows imaging of viewer's pupil in a manner that bypasses the phase modulator. The transmitted path through the beamsplitter contains the phase modulator, a Jasper Display JD5552 1920×1080 60 Hz reflective LCOS SLM, addressed via the driver board supplied in the Jasper Display JD9554 Educational Kit. To operate this SLM in a phase-modulation mode, a Thorlabs LPVISE100-A polarizer is affixed in front of the SLM. The phase-modulated illumination propagates back through the beamsplitter, with the reflected path passing through a Thorlabs 75 mm lens (the eyepiece) and on to the viewer. The transmitted path returns towards the OLED, with the previously introduced circular

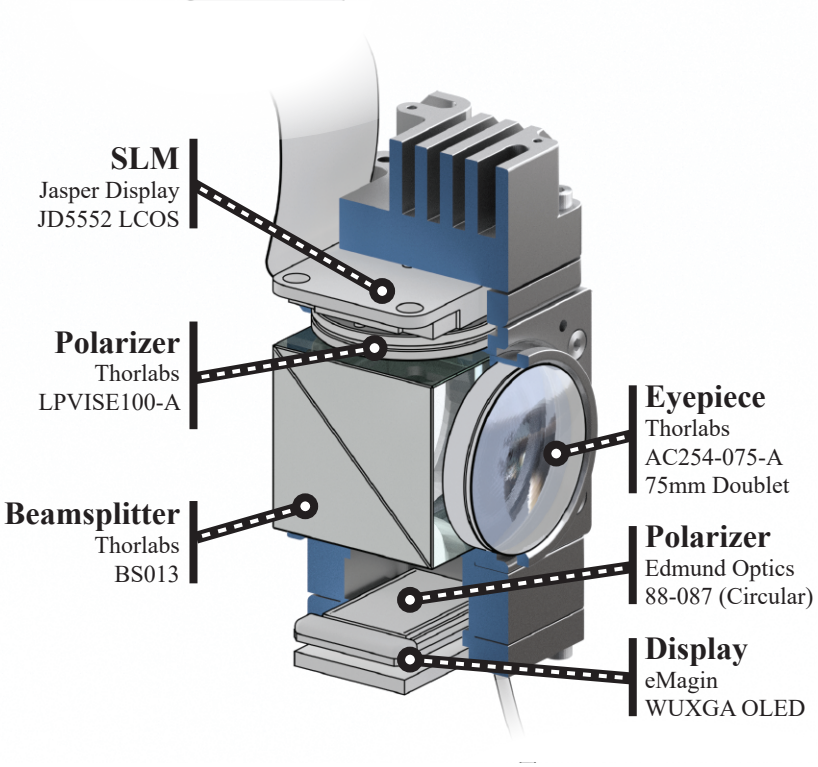
polarizer acting as an optical isolator and suppressing the return reflection. This entire assembly was duplicated, in a mirrored fashion, to enable binocular viewing, with each side mounted to a translation stage to adjust the interaxial distance (IAD) and with an optical breadboard supporting the LCOS drivers. A photograph of the assembled prototype is shown in Figure 2.9a.

Given the design considerations and the practical SLM limitations presented in Sections 2.4 and 2.6, respectively, the prototype has a measured DOF spanning 0.75−4.0 diopters (slightly less than the design specifications), assuming an eye relief of 10 mm. The field of view, limited by the size of the SLM, is 18° diagonally.

The HDMI inputs for the OLEDs and SLMs are connected to a host computer containing a pair of NVIDIA GTX Titan X (Maxwell) graphics cards with a 3.4 GHz Intel Core i7-3770 processor and 16 GB RAM. This computer was also used to run the focal surface decomposition, blending, and other rendering algorithms.



(a) Construction of the Prototype



(b) Arrangement of the Optical Components

Figure 2.9. **Hardware Prototype:** Our binocular focal surface display prototype incorporates commodity optical and mechanical components, as well as 3D-printed support brackets. (a) The prototype is mounted to an optical breadboard to support the comparatively large LCOS driver electronics. (b) A cutaway of of the prototype exposes the arrangement of the optical components.

2.5.2. Software

The forward rendering model from Section 2.3.3 was implemented using NVIDIA OptiX. Our scene database was rendered using Unity 5.5, assuming an ideal circular pupil. Focal stacks were evaluated offline with an accumulation buffer.

Focal surface decomposition is optimized using a cost function following Section 2.3.2, as implemented in C/C++ with Ceres Solver [**4**]. The LBFGS algorithm [**103**] was selected for iterative gradient descent. Depth map decompositions were evaluated on 192×108 downsampled images, with an average run time of 2.4 seconds (for three image components). Phase function optimization at the native SLM resolution took about 46 seconds per focal surface. Our optimized blending algorithm, again with three planes, took an average of 42 minutes (with 30 iterations), comparable to the run time reported by Narain et al. [**96**]. In contrast, linear blending required 17 seconds.

2.5.2.1. Calibration

Operation of a focal surface display requires understanding the alignment of optical components. Errors in assembly manifest as displacements in the focal surfaces, requiring calibration. For this purpose, we first employ a calibrated varifocal camera, using a Varioptic Caspian C-C-39N0-250-R33 tunable lens. With this camera, we measure the location of the rendered focal surfaces and, thereby, refine our estimates of the system parameters. Second, we position the camera so that it is located at the rendered center of projection. Third, we measure and correct transverse chromatic aberration using controlled illumination patterns.

2.5.3. Experimental Results

Experimental results are reported in Figures 2.10 and 2.11. In these examples, we apply time-multiplexed presentation with three focal surfaces, similar to prior simulations. We emphasize that color fields were displayed simultaneously in all cases (see Section 2.6 for details).

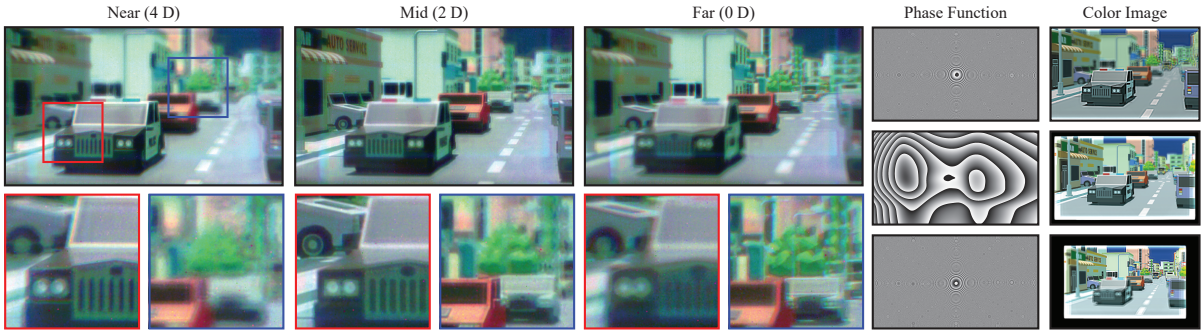


Figure 2.10. **Experimental Results, Optimized Blending:** Our prototype focal surface display achieves high resolution with near-correct retinal blur. Photographs of the prototype are shown in the first three columns, as taken by focusing the camera at the indicated distances. The last two columns depict the corresponding optimization outputs, including the phase functions and the color images. Note that optimized blending is applied with three time-multiplexed focal surfaces. The phase functions are wrapped assuming a wavelength of 532 nm. (Source imagery courtesy Unity Asset Store publisher "VenCreations".)

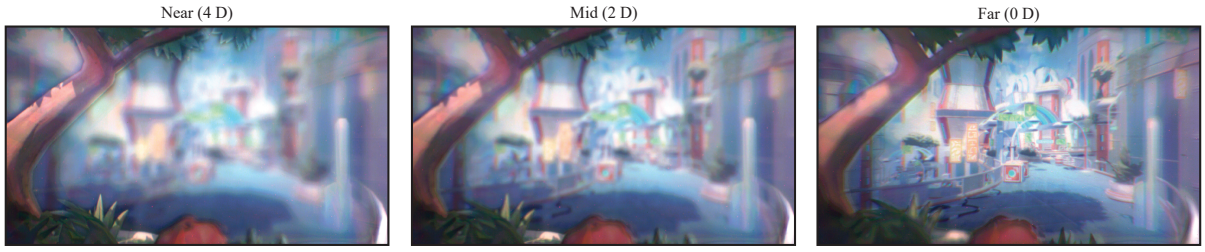


Figure 2.11. **Experimental Results, Linear Blending:** Experimental results using linear blending over three time-multiplexed focal surfaces, following Akeley et al. [**6**]. (Source imagery courtesy of Thomas Guillon.)

Our prototype addresses a key question: does diffraction degrade image quality to an extent prohibiting practical applications? To this end, we measured the modulation transfer function (MTF). Following Figure 2.12, MTF was assessed by displaying a series of sinusoids at a given focal distance, focusing a varifocal camera to that distance, and measuring the average contrast over the FOV. As predicted in prior sections, focal surface displays support high-resolution imagery. Specifically, our prototype achieves a resolution better than 5 cycles per degree throughout the accommodation range. As a result, our prototype is on par with modern VR HMDs, and considerably better in the center of its depth of focus.

Higher resolutions (exceeding 20 cpd) are possible when the SLM is used with longer focal lengths, as occurring for system focus near 3.0 diopters. In our prototype, the SLM creates shorter focal lengths as the focus approaches 1.0 and 4.0 diopters, resulting in reduced contrast (see Figure 2.12). The SLM may also exhibit chromatic aberration, further reducing contrast. Critically, diffraction-related issues often prohibit layered displays from achieving high resolutions. Focal surface displays are not similarly hindered. However, practical SLMs support finite, discrete phase modulation, typically limited to a range of 2π. Large phase gradients, as occurring with short focal lengths, produce quantization artifacts and frequent phase resets, resulting in unwanted energy in higher-order diffraction modes and stray light [**73**]. These effects reduce contrast, as shown in Figures 2.10 and 2.11. Thus, we observe a key direction for future work: extending the phase modulation range beyond 2π to allow higher resolutions and sharper variations in focal surfaces.

2.6. Discussion

Focal surface displays have been shown to achieve high-fidelity depictions of natural scenes. We now turn our attention to discussing the current and future practicality of this concept. As with any computational display, one must jointly consider issues regarding optical hardware, display technology, and optimization algorithms.

2.6.0.1. Supporting Multiple Focal Surfaces

The primary motivation for pursuing focal surface displays over simpler multifocal designs is to reduce the number of multiplexed images. Llull et al. [**77**] apply a 400 Hz tunable lens to achieve a 60 Hz multifocal display. We use a 60 Hz SLM, but this is not a fundamental limitation: Jasper Display JD4552 and HOLOEYE LETO support 720 Hz and 180 Hz, respectively. In terms of image quality, single focal surfaces arguably perform competitively. However, we strive to depict near-correct retinal blur, particularly at occlusions. As such, designs with two focal surfaces appear a viable and practically-realizable first step toward accommodation-supporting HMDs.

2.6.0.2. Resolving Phase Modulation Issues

Our use of phase SLMs is related to earlier work on dynamic freeform lensing. As previously assessed by Damberg et al. [**22**], using LCOS panels in imaging systems presents two primary concerns: stray light and chromatic aberration. We discuss each in turn.

As discussed in Section 2.5.3, stray light may result from inefficiencies of the phase SLM. However, LCOS phase SLMs are routinely applied with adaptive optics, including for retinal imaging and aberration correction. As such, LCOS panels have already benefited from extended research into suppressing stray light. A full assessment of these effects

is beyond the scope of this work. However, our MTF measurements[2] in Figure 2.12, as well as the experimental results, support that high-resolution imagery can be created.

Following Equation 2.5, the effective SLM focal length is wavelength dependent. As a result, the LCOS panel may introduce transverse and axial chromatic aberrations. While the former can be digitally corrected by warping displayed images, the latter cannot and manifests as focusing artifacts. The conventional solution is field-sequential color presentation. However, our goal is to reduce time multiplexing, and as a result, we aim for field-simultaneous color presentation. We emphasize that Laude [**73**], Márquez et al. [**86**], and Fernandez et al. [**28**] each report the successful operation of phase-only SLMs as focusing elements using polychromatic and broadband illumination. As summarized in Figure 2.12, we measure an average axial chromatic aberration (ACA)[3] of less than 0.25 diopters over the supported accommodation range. Simulations depicted in Figure 2.13 indicate modest benefits, in terms of minimizing color fringing, by employing field-sequential color (i.e., by using separately optimized phase functions for each color channel). Note that ACA is predicted with the geometric optics simulations, due to the dispersion introduced by Equation 2.4.

[2]The SLM optical power was optimized, following Equation 2.5, for $\lambda = 532$ nm.
[3]ACA is reported as the apparent optical distance in diopters, measured relative to the green channel. Focal distances are measured using a varifocal camera and a depth-from-focus metric (i.e., maximizing contrast for a high-frequency pattern).

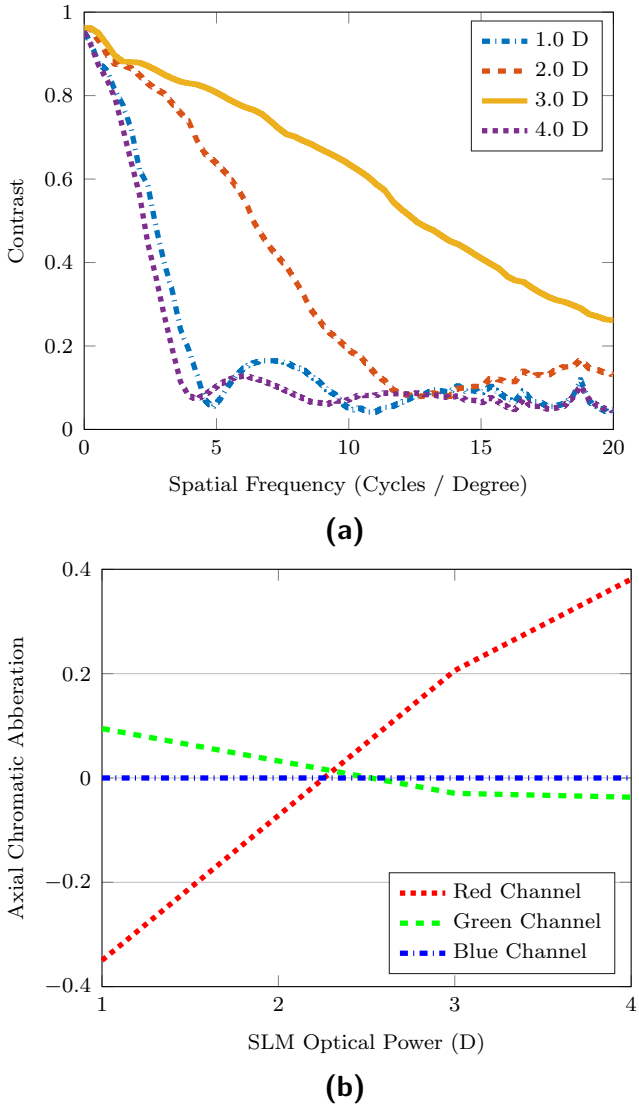


Figure 2.12. **System Resolution and Chromatic Aberration:** a) The measured modulation transfer function (MTF) of our prototype as the system varies focus from 0.0 to 4.0 diopters. Increasing contrast loss is expected away from the prototype's neutral focus of 3.0 diopter as the SLM synthesizes shorter focal lengths, due to the increased stray light from phase quantization and phase resets. b) The measured axial chromatic aberration (ACA) of our prototype is less than that of the typical human eye [**29**], confirming that focal cues are correctly rendered with field-simultaneous color presentation, in spite of polychromatic illumination.

(a) Target Focal Stack Image

(b) Field-simultaneous Color (Simulation)

(c) Field-sequential Color (Simulation)

(d) Field-simultaneous Color (Experiment)

(e) Field-sequential Color (Experiment)

Figure 2.13. **Field-sequential and Simultaneous Color:** Field-simultaneous color display minimizes time multiplexing. However, artifacts due to axial chromatic aberration (ACA) may appear in this case. (a) A target focal stack image. (b,c) Simulations comparing field-simultaneous and field-sequential modes, using the geometric optics model from Section 2.3. (d,e) Corresponding experimental results. The contrast of experimental results differs from simulations due to stray light and misalignments that cannot be predicted without more accurate wave optics modeling and calibration, respectively. (Source imagery courtesy Ruggero Corridori.)

Our simulations apply the geometric optics model from Section 2.3, which does not predict all experimental artifacts. First, we do not model wave optics effects, including stray light due to phase quantization and phase resets. As a result, the experimentally measured contrast loss, as reported in Figure 2.12, is not reproduced in the simulations. Second, the ACA of the physical SLM differs slightly from our model. Third, the calibration procedure in Section 2.5.2.1 does not account for vignetting and all sources of misalignment, introducing multifocal blending artifacts near the periphery.

While experimental results do not yet attain the quality of our geometric optics model, field-simultaneous color and mitigation of stray light appear realizable with practical SLMs, particularly by applying phase modulation exceeding the 2π range of our prototype, as described by Fernandez et al. [**28**]. We emphasize that all our experimental results, except for those in Figure 2.13, were captured while displaying all color fields simultaneously.

2.6.0.3. Optimizing Algorithm Performance

The algorithms that drive our prototype are not yet suitable for interactive content. A promising direction for future work is to explore efficient depth decomposition and optimized blending frameworks. In terms of the latter, the optimized blending algorithm of Narain et al. [**96**] poses a more significant hurdle, with reported minute-long run times. However, linear blending could be adopted to approach real-time refresh rates, albeit with diminished retinal blur fidelity.

2.6.0.4. Enabling Practical Applications

Here we turn our attention first to practical VR applications, and then to AR. Our prototype is not yet wearable, due to the large LCOS drivers. This is not a fundamental limitation, as attested by commercial pico projectors. However, VR applications do confront a current roadblock: LCOS panels are smaller than modern VR optics. As such, the field of view remains limited. Increasing the FOV requires three changes: using a shorter focal length eyepiece, eliminating the beamsplitter and replacing the reflective LCOS with a transmissive one, and reducing the overall optical stack height. Even if these measures were taken, a larger SLM would be required. Practical VR applications will require custom SLMs. However, we emphasize that most accommodation-supporting HMDs are similarly technologically limited to narrow FOVs.

Focal surface displays currently appear to be a forward-looking architecture requiring further maturation of SLM technology. While our prototype modifies a conventional VR architecture, largely due to the accessibility of catalog eyepieces, we believe focal surface displays can be equally applied to AR devices (e.g., those that substitute a projector and a combiner for the display and eyepiece). This configuration is a natural direction for focal surface displays: larger SLMs (our primary limitation) would not be required, as existing models would easily fit into a miniature projector. As such, focal surface displays continue the legacy of retinal scanning displays, providing a viable path to address refresh rate and multivalued depth limitations encountered by McQuaide et al. [**92**].

2.6.1. Future Work

Immediate extensions to this work include upgrading to wave optics modeling, generalizing to non-smooth focal surfaces, and exploring alternative depth map decompositions (e.g., those that penalize all focal surfaces, rather than just the closest.) However, the future work for focal surface displays largely overlaps with that required for all multifocal displays. As presented in Section 2.3, focal surface displays are a form of fixed-viewpoint volumetric display: rendering, optimization, and viewing are all assumed to occur relative to the viewer's entrance pupil. It is worth noting that Maxwellian view, retinal scanning, and other extended depth of focus concepts also share this assumption. A promising direction is to determine whether, through hardware or algorithms, eye movement can be supported. With eye tracking, focal surface displays may be driven in a gaze-contingent manner, similar to varifocal concepts. There is also an opportunity to leverage concepts from near-eye light field displays, rendering imagery to support limited eye movement. In this manner, we believe the challenges and research directions for all accommodation-supporting displays are closely tied.

2.7. Conclusion

Focal surface displays continue down the path set by varifocal and multifocal concepts, further customizing virtual images to scene content. We have demonstrated that emerging phase-modulation SLMs are well-prepared to realize this concept, having benefited from decades of research into closely-related adaptive imaging applications. We have demonstrated high-resolution focal stack reproductions with a proof-of-concept prototype, as well as presented a complete optimization framework addressing the joint focal surface and color image decomposition problems. Due to the complex and content-dependent nature of light propagation within this display architecture, coupled with the high resolutions required of near-eye displays, rendering-based optimization is uniquely well suited to producing natural retinal images in this case.

CHAPTER 3

Active 3D Scanning with Multibounce Interference

3.1. Introduction

Many applications in science and industry, such as robotics, bioinformatics, augmented reality, and manufacturing automation rely on capturing the 3D shape of scenes. Structured light (SL) methods, where the scene is actively illuminated to reveal 3D structure, provide the most accurate shape recovery compared to passive or physical techniques [**9, 121**]. We will consider triangulation-based SL techniques, which have been shown to produce the most accurate depth information over short distances [**123**], as well as time-of-flight (TOF) techniques, which have become commercially viable for robotics and entertainment applications because of their fast acquisition speeds [**30, 34, 90, 91**]. Most active depth imaging systems operate with practical constraints on sensor bandwidth and light source power. These resource limitations force concessions in acquisition speed, resolution, and performance in challenging 3D scanning conditions such as strong ambient light (e.g., outdoors) [**93, 41**], participating media (e.g. fog, dust or rain) [**53, 54, 97, 38**], specular materials [**111, 98**], and strong inter-reflections within the scene [**39, 37, 20, 106, 3**]. This chapter focuses on this last problem.

Recent computational imaging work addresses the inter-reflection problem, also known as multibounce interference, with novel SL measurement approaches [**105, 2, 40, 89**].

These acquisition systems are designed to reject information related to multibounce interference, treating it as noise. It is possible that by instead retaining these measurements, complementary information can be recovered that improves surface reconstruction. If an accurate forward model is available that can provide an unbiased, physically-correct simulation of the multibounce interference given a surface estimate, an optimization algorithm can use this model to find the correct surface that produced the experimental observation [**99, 32, 56, 95**].

This thesis chapter summarizes SL systems and their trade-offs, then examines a motion contrast hardware approach for rejecting multibounce interference. Then, this chapter describe a forward model, implemented using conventional raytracing techniques, and a simple optimization that, rather then rejecting multibounce interference, can correct these biased measurements to produce a more accurate surface estimate. We compare this approach to ground truth in simulation, and show experimental results using commercial 3D acquisition hardware.

3.1.1. contributions

- A technique to suppress multibounce interference in hardware using a motion contrast sensor
- An adaptation of conventional path tracing techniques to implement forward models for SL and ToF systems
- A simple gradient descent inverse rendering algorithm incorporating the raytraced forward model
- Simulated and experimental results demonstrating surface shape recovery in the presence of multibounce interference

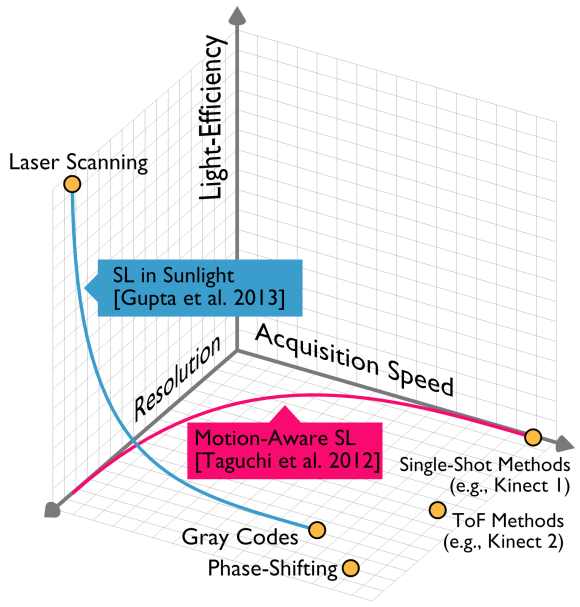


Figure 3.1. **Taxonomy of Active 3D Systems:** Active 3D depth imaging systems face trade-offs in acquisition speed, resolution, and light efficiency. Laser scanning (upper left) achieves high resolution at slow speeds. Single-shot triangulation methods (mid-right) obtain lower resolution with a single exposure. Time-of-Flight methods obtain higher resolution results, but not at conventional camera resolutions. Other methods such as Gray coding and phase shifting (mid-bottom) balance speed and resolution but have degraded performance in the presence of strong ambient light, scene inter-reflections, and dense participating media. Hybrid techniques from Gupta et al. [**41**] (curve shown in green) and Taguchi et al. [**132**] (curve shown in red) strike a balance between these extremes.

3.2. Active 3D Scanning Systems and Limitations

Speed-resolution trade-off: Most existing methods achieve either high resolution or high acquisition speed, but not both. This trade-off arises due to limited sensor bandwidth. On one extreme are the point/line scanning systems [**5**] (Figure 3.1, upper left), which achieve high-quality results. However, each image captures only one point (or line) of depth information, thus requiring hundreds or thousands of images to capture the entire scene. Improvements can be made in processing, such as the space-time analysis proposed

by Curless et al. [**21**] to improve accuracy and reflectance invariance, but ultimately traditional point scanning remains a highly inefficient use of camera bandwidth.

Methods such as Gray coding [**115**] and phase shifting [**129, 39**] improve bandwidth utilization but still require capturing multiple images (Figure 3.1, lower center). Single-shot triangulation methods [**146, 147**] enable depth acquisition (Figure 3.1, right) with a single image but achieve low resolution results due to a reliance on block matching to determine projector-camera correspondences.

Focal plane array time-of-flight techniques [**124, 71**] can perform a depth estimation per-pixel, so do not suffer from the resolution reduction of single-shot triangulation techniques. The complex pixel architecture, however, has thus far limited their resolution to a small fraction of conventional cameras.

Content-aware techniques improve resolution in some cases [**46, 65, 42**], but at the cost of reduced capture speed [**132**]. We have introduced a method [**89**] achieving higher scan speeds while retaining the advantages of traditional laser scanning, which will be summarized in the next section.

Speed-robustness trade-off: This trade-off arises due to limited light source power and is depicted by the green "SL in sunlight" curve in Figure 3.1. Laser scanning systems concentrate the available light source power in a smaller region, resulting in a large signal-to-noise ratio, but require long acquisition times. In comparison, the full-frame methods (phase-shifting, Gray codes, single-shot triangulation and focal plane array time-of-flight methods) achieve high speed by illuminating the entire scene at once but are prone to errors due to ambient illumination [**41, 2**] and indirect illumination due to inter-reflections and scattering [**37, 40**].

3.2.1. Ambient and Global Illumination in SL

Active systems rely on the assumption that light travels directly from source to scene to camera. However, in real-world scenarios, scenes invariably receive light indirectly due to inter-reflections and scattering, as well as from ambient light sources (e.g., sun in outdoor settings). In the following, we discuss how point scanning systems are the most robust in the presence of these undesired sources of illumination.

Point scanning and ambient illumination. Let the scene be illuminated by the active light source and an ambient light source. Full-frame SL methods (e.g., phase-shifting, Gray coding) spread the power of the structured light source over the entire scene. Suppose the brightness of the scene point due to the structured light source and ambient illumination are I_{active} and $I_{ambient}$, respectively. Since ambient illumination contributes to photon noise, the SNR of the intensity measurement can be approximated as $\frac{I_{active}}{\sqrt{I_{ambient}}}$ [**41**]. However, if the power of the structured controllable source is concentrated into only a fraction of the scene at a time, the effective source power increases and higher SNR is achieved. Since point scanning systems maximally concentrate the light (into a single scene point), they achieve the most robust performance in the presence of ambient illumination for any of these systems.

Point scanning and global illumination. The contributions of both direct and indirect illumination may be modeled by the light transport matrix $\mathbf{T}$ that maps a set of $R \times C$ projected intensities I from a projector onto the $M \times N$ measured intensities $\hat{I}$ from the camera.

$$\hat{I} = \mathbf{T}I \tag{3.1}$$

The component of light that is directly reflected to the i^{th} camera pixel is given by $\mathbf{T}_{i,\alpha}I_\alpha$ where the index α depends on the depth/disparity of the scene point. All other entries of $\mathbf{T}$ correspond to contributions from indirect reflections, which may be caused by scene inter-reflections, sub-surface scattering, or scattering from participating media. SL systems project a set of K patterns which are used to infer the index α that establishes projector-camera correspondence. For SL techniques that illuminate the entire scene at once, such as phase-shifting SL and binary SL, the sufficient condition for estimating α is that direct reflection must be greater than the sum of all indirect contributions:

$$\mathbf{T}_{i,\alpha} > \sum_{k \neq \alpha} \mathbf{T}_{i,k} \tag{3.2}$$

For scenes with significant global illumination, this condition is often violated, resulting in depth errors [**37**]. For point scanning, a set of $K = R \times C$ images are captured, each corresponding to a different column $\mathbf{T}_i$ of the matrix $\mathbf{T}$. In this case, a sufficient condition to estimate α is simply that direct reflection must be greater than each of the individual indirect sources of light, i.e:

$$\mathbf{T}_{i,\alpha} > \mathbf{T}_{i,k},\ \forall k \in \{1, \cdots R \times C\},\ k \neq \alpha \tag{3.3}$$

If this condition is met, α can be found by simply thresholding each column $\mathbf{T}_i$ such that only one component remains. Since Equation 3.3 is a significantly less restrictive requirement than Equation 3.2, point scanning systems are much more robust in the presence of significant global illumination (e.g. a denser $\mathbf{T}$ matrix).

3.2.2. Observations

We can summarize these characteristics of active scanning techniques to make the following two observations:

Observation 1: In order for the light source to be used with maximum efficiency, it should be concentrated on the smallest possible scene area. Point light scanning systems concentrate the available light into a single point, thus maximizing SNR.

Observation 2: In conventional scanning-based SL systems, most of the sensor bandwidth is not utilized. For example, in point light scanning systems, every captured image has only one sensor pixel that witnesses an illuminated spot under ideal conditions.

These observations open the path to a hardware-only approach to mitigating multi-bounce interference, which this thesis will explore prior to addressing the potential for rendering-based optimization to make use of this interference.

3.3. Motion Contrast 3D Scanning

As an exercise in improving a structured light system with hardware alone, consider the bandwidth inefficiency of laser scanning discussed in the previous section. The potential light efficiency and accuracy of laser scanning is high, but the conventional architecture performs an extremely comprehensive, and thus slow, light transport measurement. An intensity image is captured for every projector illumination angle. The resulting data can be arranged into the full M by N light transport matrix that maps the relationship between each addressable projector angle to each camera pixel. In conventional laser scanning, a peak finding operation selects a single entry in each row of the matrix corresponding to the most likely first-bounce camera coordinate for the laser position. After processing, the MxN light transport matrix produces a binary list of M pixel correspondences. Can we avoid measuring the discarded values in the first place?

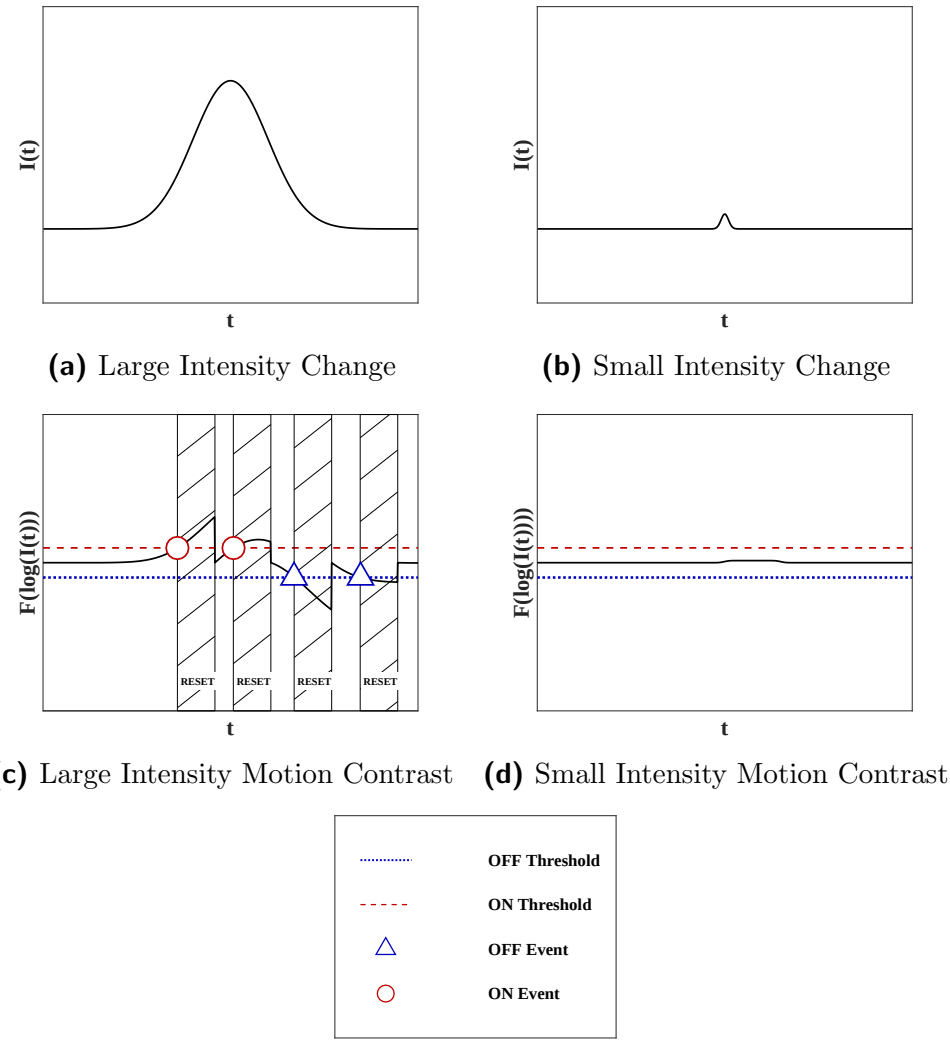
Ideally, we need a sensor that measures only the scene points that are directly illuminated by the scanning light source. Although conventional sensors do not have such a capability, we draw motivation from biological vision where sensors that only report salient information are commonplace. Organic photoreceptors respond to changes in instantaneous contrast, implicitly culling static information. If such a sensor observes a scene lit with scanning illumination, measurement events will only occur at scene points containing the moving spot. Digital sensors mimicking the differential nature of biological photoreceptors are now available as commercially packaged camera modules. Thus, we can use these off-the-shelf components to build a scanning system that utilizes both light power and measurement bandwidth in an efficient manner.

3.3.1. Motion Contrast Cameras

Lichtsteiner et al. [**75**] recently introduced the biologically inspired *Motion Contrast Camera*, in which pixels on the sensor independently and asynchronously generate output when they observe a temporal intensity gradient of magnitude greater than a predefined threshold. Given a time-varying irradiance received at the pixel photodetector, an amplifier produces a voltage proportional to the log intensity measurement. An analog finite-difference operation produces an approximation of the temporal derivative of the log intensity signal, albeit with some resonant behavior. A comparator performs a binary thresholding of this time differential given externally set reference voltages. An FPGA polls the output of each pixel's comparator at 1 MHz. A time stamp is assigned to the output events, which are queued and transmitted serially to the host computer.

The output of this pixel architecture on a time-varying intensity is illustrated in Figure 3.2. A large intensity change relative to predefined thresholds will produce on and off events separated by a minimum reset time governed by the electrical characteristics of the amplifier and differencing circuits. A small intensity change relative to the same thresholds will produce no output at all.

In a projector-camera system, the threshold voltage can be set so that the highest intensity direct path (identifying projector-camera correspondences) produces an event, but all global effects fall below the threshold. In this configuration, instead of measuring the full light transport matrix, a list of projector-camera pixel correspondences is produced directly. We propose applying this strategy to improve the performance of a laser scanning system.



(a) Large Intensity Change **(b)** Small Intensity Change

(c) Large Intensity Motion Contrast **(d)** Small Intensity Motion Contrast

Figure 3.2. **Motion Contrast Events**: When a motion contrast pixel observes a large change in intensity (a), the output (c) consists of ON events (red circles) when the change in log intensity over time exceeds a preset threshold (red dashed line) and OFF events (blue triangles) when the change in log intensity drops below a preset threshold (blue dotted line). Each of these events is followed by a fixed reset time (hatched box) that is a function of the internal amplifier and differencing circuit characteristics. When the change in observed intensity is low (b), no output events are produced (d).

3.3.2. Motion Contrast 3D Scanning

The key principle behind Motion Contrast 3D Scanning (MC3D) is the conversion of spatial projector-camera disparity to temporal events recorded by the motion contrast sensor. Interestingly, the idea of mapping disparity to time has been explored previously in the VLSI community, where several researchers have developed highly customized CMOS sensors with on-pixel circuits that record the time of maximum intensity [**7, 61, 104**]. The use of a motion contrast sensor in a 3D scanning system is similar to these previous approaches with two important differences: 1) The differential logarithmic nature of motion contrast cameras improves performance in the presence of ambient illumination and complex scene reflectance, and 2) motion contrast cameras are currently commercially available while previous techniques required custom VLSI fabrication, limiting access to only the small number of research labs with the requisite expertise.

MC3D consists of a laser point scanner that is swept relative to a DVS sensor. The event timing from the DVS is used to determine scan angle, establishing projector-camera correspondence for each pixel. The DVS was used previously for SL scanning by Brandli et al. [**15**] in a pushbroom setup that sweeps an affixed camera-projector module across the scene. This technique is useful for large area terrain mapping but ineffective for 3D scanning of dynamic scenes. Our focus is to design a SL system capable of 3D capture for exceptionally challenging scenes, including those containing fast dynamics, significant specularities, and strong ambient and global illumination.

For ease of explanation, we assume that the MC3D system is free of distortion, blurring, and aberration; that the projector and camera are rectified and have equal focal

lengths f; and are separated by a baseline b [1]. We use a 1D analysis that applies equally to all camera-projector rows. A scene point s maps to column i in the camera image and the corresponding column α in the projector image (see Figure 3.3). Referring to the right side of Equation 3.1, after discretizing time by the index t, the set of $K = R \times C$ projected patterns from a point scanner becomes:

$$\mathbf{I} = [I_1, \cdots I_K] = I_0 \delta_{i,t} + I_b \tag{3.4}$$

where δ is the Kronecker delta function, I_0 is the intensity of the focused laser beam, and I_b represents the small amount of background illumination introduced by the projector (e.g. due to scattering in the scanning optics). From Equation 3.1, the light intensity directly reflected to the camera is:

$$\hat{I}_{i,t} = \rho_{i,\alpha} I_{\alpha,t} = (I_0 \delta_{\alpha,t} + I_b)\rho_{i,\alpha} \tag{3.5}$$

where $\rho_{i,\alpha}$ denotes the fraction of light reflected in direction i that was incident in direction α (i.e. the BRDF) and the pair $[i, \alpha]$ represent a projector-camera correspondence. Motion contrast cameras sense the time difference of the logarithm of incident intensity [**75**]:

[1] Lack of distortion, equal focal lengths, etc., are not a requirement for the system and can be accounted for by calibration.

$$\hat{I}^{MC}_{i,t} = \log(\hat{I}_{i,t}) - \log(\hat{I}_{i,t+1}), \tag{3.6}$$

$$= \log\left(\frac{I_0 + I_b}{I_b}\right)\delta_{\alpha,t} \tag{3.7}$$

Next, the motion contrast intensity is thresholded and the set of space and time indices are transmitted asynchronously as tuples:

$$[i, \tau],\ s.t.\ \hat{I}^{MC}_{i,t} > \epsilon, \tau = t + \sigma \tag{3.8}$$

where σ is the timing noise that may be present due to pixel latency, multiple event firings, and projector timing drift. The tuples are transmitted as an asynchronous stream of events (Figure 3.3, middle) which establish correspondences between camera columns i and projector columns $j = \tau \cdot S$ (Figure 3.3, right), where ν is the projector scan speed in columns/sec. The depth is then calculated as:

$$z(i) = \frac{bf}{(i - \tau \cdot \nu)} \tag{3.9}$$

Fundamentally, MC3D is a scanning system, but it differs from conventional implementations because the motion contrast sensor implicitly culls unnecessary measurements. A conventional camera must sample the entire image for each scanned point, while the motion contrast camera samples only one pixel, drastically reducing the number of measurements required.

3.3.3. Motion Contrast 3D Scanning Implementation

For our prototype, we use the iniLabs DVS128 [**75**]. The camera module contains a 1st generation 128x128 CMOS motion contrast sensor, which has been used in research applications such as high-frequency tracking [**102**], unsupervised feature extraction [**10**], and neurologically inspired robotic control systems [**57**].

The DVS128 uses event time-stamps assigned using a 100kHz counter [**75**]. For our 128 pixel line scanning setup, this translates to a maximum resolvable scan rate of nearly 800Hz. The dynamic range of the DVS is more than 120dB due to the static background rejection discussed earlier [**75**].

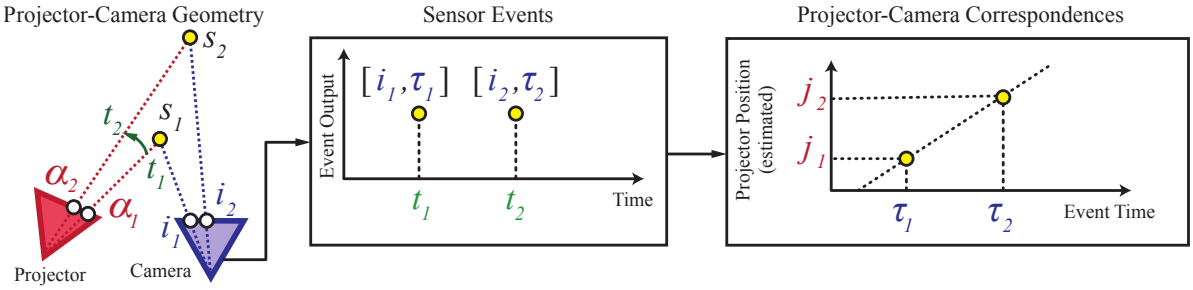


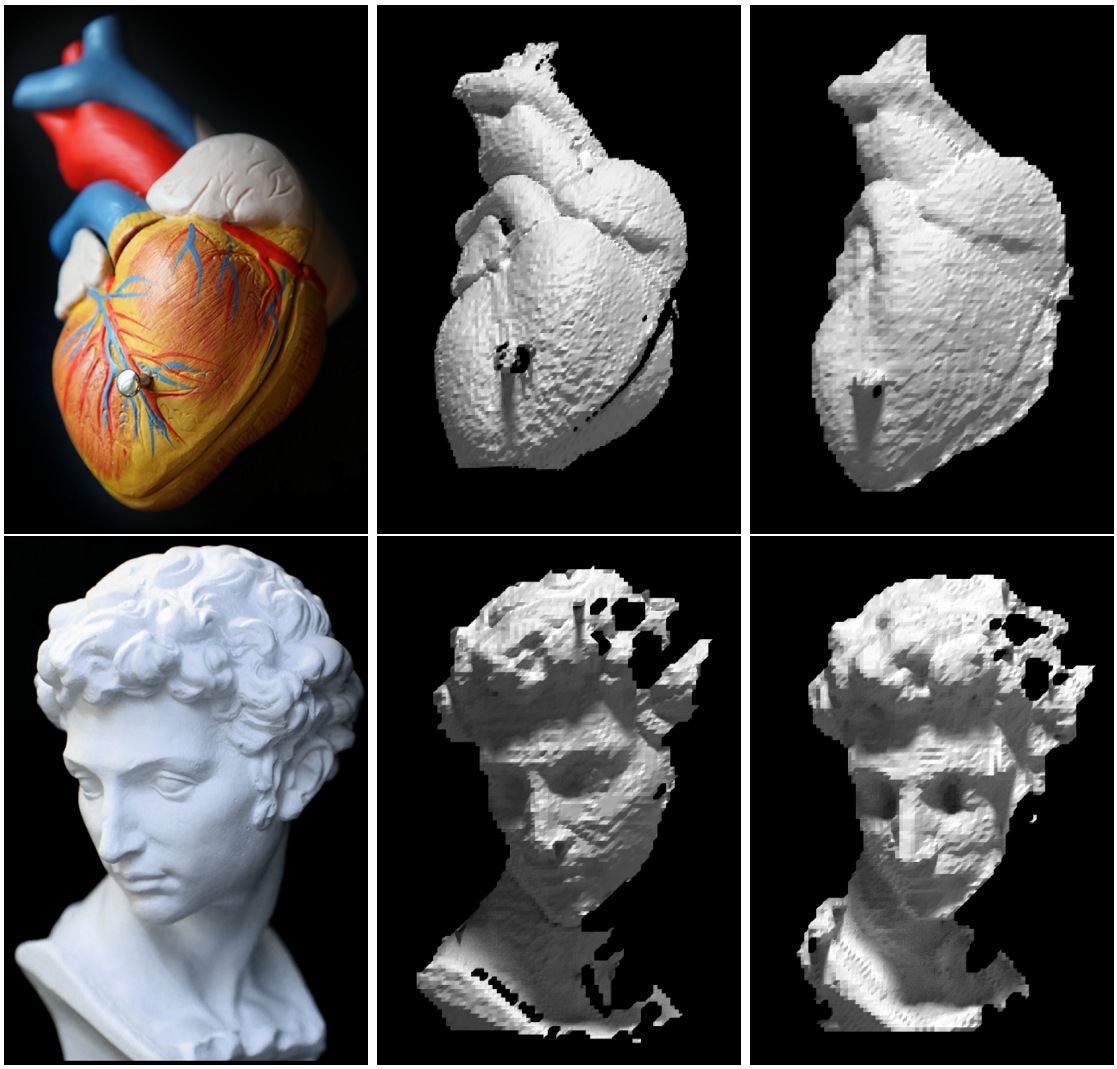
Figure 3.3. **System Model:** A scanning source illuminates projector positions α_1 and α_2 at times t_1 and t_2, striking scene points s_1 and s_2. Correspondence between projector and camera coordinates is not known at runtime. The DVS sensor registers changing pixels at columns i_1 and i_2 at times t_1 and t_2, which are output as events containing the location/event time pairs $[i_1, \tau_1]$ and $[i_2, \tau_2]$. We recover the estimated projector positions j_1 and j_2 from the event times. Depth can then be calculated using the correspondence between event location and estimated projector location.

We used two different sources in our prototype implementation: a portable, fixed-frequency point scanner and a variable-frequency line scanner. The portable scanner was a SHOWWX laser pico-projector from Microvision, which displays VGA input at 848x480 60Hz by scanning red, green, and blue laser diodes with a MEMS micromirror [**1**]. The

micromirror follows a traditional raster pattern, thus functioning as a self-contained 60Hz laser spot scanner. For the variable-frequency line scanner, we used a Thorlabs GVSM002 galvanometer coupled with a Thorlabs HNL210-L 21mW HeNe Laser and a cylindrical lens. The galvanometer is able to operate at scan speeds from 0-250Hz.

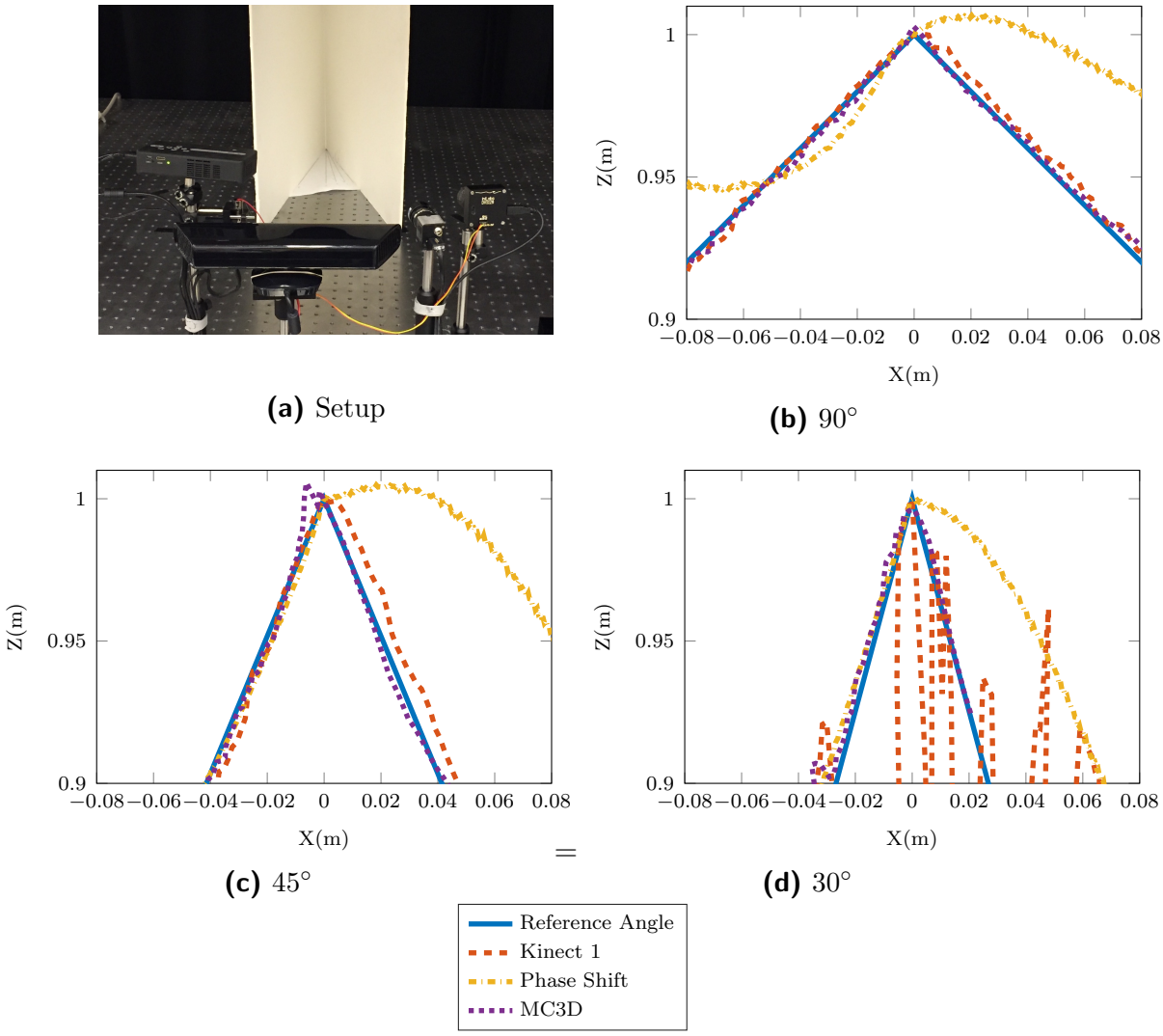
Evaluation of simple shapes: To quantitatively evaluate the performance of our system, we scanned a plane and a sphere. We placed the plane parallel to the sensor at a distance of 500mm and captured a single scan (one measurement per pixel). Fitting an analytic plane to the result using least squares, we calculated a depth error of 7.849mm RMSE. Similarly, for a 100mm diameter sphere centered at 500mm from the sensor, depth error was 12.680mm RMSE.

Evaluation of complex scenes: To demonstrate the advantages of our system in more realistic situations, we used two test objects: a medical model of a heart and a miniature plaster bust. These objects both contain smooth surfaces, fine details, and strong silhouette edges. We also captured the same scenes with traditional laser scanning using the same galvanometer setup and an IDS UI348xCP-M Monochrome CMOS camera. The image was cropped using the camera's hardware region of interest to 128x128. The camera was then set to the highest possible frame rate at that resolution, or 573fps. This corresponds to a total exposure time of 28.5s, though the real world capture time was 22 minutes. Note that MC3D, while requiring several orders of magnitude less capture time than traditional laser scanning, achieves similar quality results, shown in Figure 3.4.



(a) Reference Photo **(b)** Laser Scan **(c)** MC3D

Figure 3.4. **Comparison with Laser Scanning:** Laser scanning performed with laser galvanometer and traditional sensor cropped to 128x128 with total exposure time of 28.5s. MC3D method captured with 1 second exposure at 128x128 resolution and median filtered. Object placed 1m from sensor under ∼150 lux ambient illuminance measured at object.

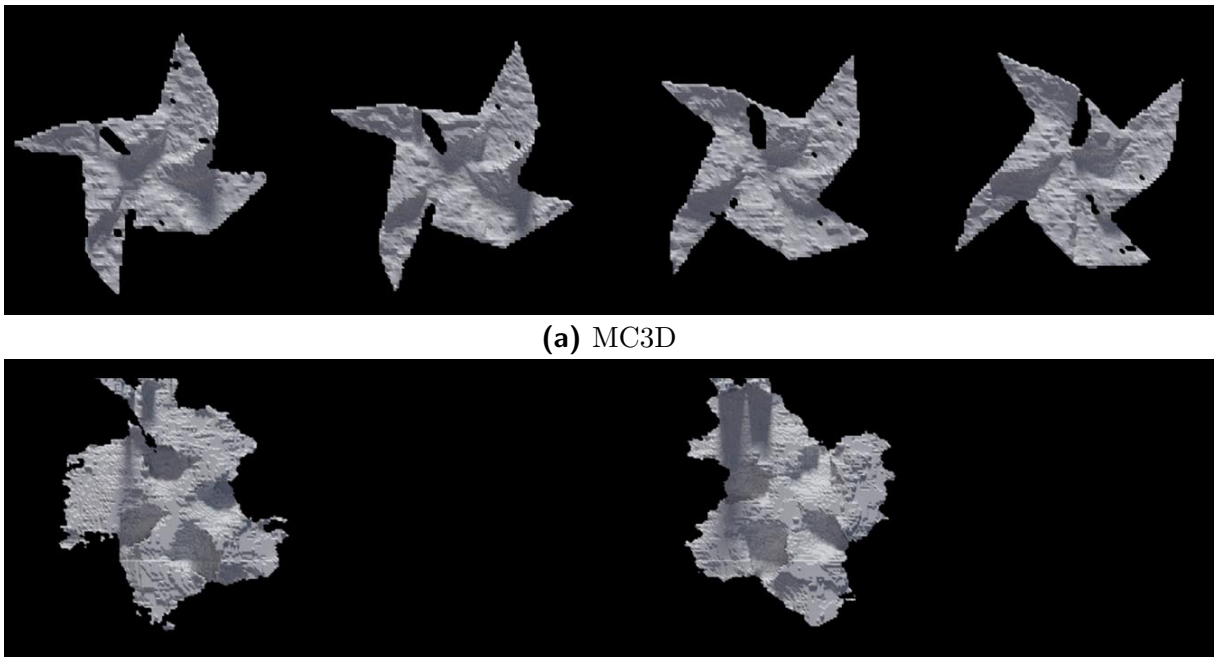


(a) Setup **(b)** 90° **(c)** 45° **(d)** 30°

Figure 3.5. **Performance with Interreflections:** Comparison between Kinect 1, phase shifting, and MC3D. Experimental setup shown in (a). A 90° v-groove, assembled from foam core board shown in (b). (c) and (d) show 45° and 30° v-grooves, respectively. Kinect 1 (measurements averaged over 1 second) produces comparable results to MC3D in the 90° and 45° cases as the block matching algorithm rejects interreflections. In the 30° case, however, the block matching algorithm fails completely due to interreflections. Phase shifting (16 phase offsets recorded over 64 seconds of total exposure time, using a low frequency period equal to the width of the projector), has severe multibounce interference even at 90°. MC3D (measurements averaged over 1 second) is not susceptible to these effects as it is a laser point scanning technique.

Strong scene inter-reflections: Figure 3.5 shows the performance of MC3D for a scene with significant multibounce interference. The test scene consists of two pieces of white foam board meeting at a 90, 45, and 30 degree angle. Kinect results in the image are accurate for 90° and 45°, but fail at 30° when interreflections obscure the projected random dot pattern to an extent that prevents block matching. Phase shifting results are significantly distorted for all v-groove angles, as any multibounce path contributes a bias to the radiometrically calibrated measurements. MC3D, being a true point laser scanning approach, passively rejects lower-energy multibounce paths via motion contrast thresholding, and consequently does not show significant distortions due to inter-reflections. Note that these depth-from-disparity reconstructions have been produced using approximate camera and projector calibration, with v-groove vertices manually aligned to provide a consistent frame of reference. Numerical error and dynamic range comparisons are not provided for this reason.

Motion comparison: We captured a spinning paper pinwheel using the SHOWWX projector to show the system's high rate of capture. Four frames from this motion sequence are shown at the top of Figure 3.6. Each image corresponds to consecutive 16ms exposures captured sequentially at 60fps. A Kinect capture at the bottom of the figure shows the pinwheel captured at the maximum 30fps frame rate of that sensor.



(a) MC3D

(b) Kinect

Figure 3.6. **Motion Comparison:** The top row depicts 4 frames of a pinwheel spinning at roughly 120rpm, captured at 60fps using MC3D. The bottom row depicts the same pinwheel spinning at the same rate, over the same time interval, captured with the Kinect. Only 2 frames are shown due to the 30fps native frame rate of the Kinect.

3.3.4. MC3D Limitations

There are several noise sources in our prototype system, such as uncertainty in event timing due to internal electrical characteristics of the sensor, multiple event firings during one brightness change event, or downsampling in the sensor's digital interface. As scan speed increases, timing errors are amplified, resulting in an increased amount of dropped events. These can be mitigated through updated sensor designs, further system engineering, and more sophisticated point cloud processing.

More fundamentally, rejecting measurements from low-energy multibounce paths inherently reduces the SNR of the system. Hardware multibounce rejection techniques such as those relying on epipolar constraints ([**105, 2**]) suffer from this issue. Multibounce paths manifest in conventional reconstruction techniques as noise. However, we know from light transport analysis that multibounce paths in fact produce systematic bias that can be corrected for with an appropriate optimization. This motivates the last contribution of this thesis, an rendering-based optimization depth recovery algorithm.

3.4. Image Formation Modeling and Rendering for Active 3D Scanning

Chapter 1 highlighted the equivalency between a linear model of light transport and the type of fixed-point algorithm employed in modern raytracers, then introduced the rendering equation in Equation 1.1 and its stochastic approximation using Monte Carlo integration. The expression for the contribution along an individual path, Equation 1.2, will be used throughout this section following [**114**].

3.4.1. SL Forward Model

In the SL case, the value of a ray exiting the projector will contain the coded pixel position originating the ray. In the case of sinusoidal projector patterns, we will propagate a phasor through the scene. The emission intensity I_e becomes $\bar{I}_e$, the emitted complex phasor[2]. The phase of the phasor to be projected is calculated using the pixel projection of the path endpoint onto the projector using projection matrix $\boldsymbol{P}_p$.

$$\bar{I}_e(\mathrm{s}_d \rightarrow \mathrm{s}_{d-1}) = e^{i\frac{2\pi}{\lambda}\boldsymbol{P}_p \mathrm{s}_x} \tag{3.10}$$

Here, λ is the period of the spatial encoding pattern in pixels. In the case of a projector-camera system whose baseline is parallel with the x-axis, the column component of $\bar{I}_e$ needs to be propagated through the scene.

[2]For raytracer compatibility the path value can be a tuple containing the real-valued phase and magnitude, or real and imaginary components, for later conversion back to a complex value prior to phasor summation.

We pass this phasor through the Monte Carlo estimation for path $\bar{P}$ (now complex valued) in place of the emission term. Since we are utilizing a single-point light source (the projector), the emitter PDF $p_A(\mathrm{s}_d)$ becomes unity.

$$\bar{P} = \bar{I}_e(\mathrm{s}_d \rightarrow \mathrm{s}_{d-1})\rho(\mathrm{s}_d \rightarrow \mathrm{s}_{d-1} \rightarrow \mathrm{s}_{d-2})G(\mathrm{s}_d \leftrightarrow \mathrm{s}_{d-1}) \times \beta \tag{3.11}$$

The raytracer initiates N ray samples from each camera pixel. If the SL technique being modeled involves thresholding, such as intensity thresholding in Gray coding or temporal contrast thresholding in MC3D, then the weight of each accumulated sub-path output is passed through a thresholding operator τ, which omits the current path if the throughput (or change in throughput, in the case of MC3D) is below the threshold. Each of these initiates the iterative path integral estimation above, iterating until maximum depth D is reached. The phasor observation for the current pixel, $\bar{I}_{x,y}$, is the accumulation of all N phasors contained in the sampled paths.

$$\bar{I}_{x,y} = \frac{1}{N}\sum_{n=1}^{N}\frac{1}{D}\sum_{d=1}^{D}\tau(\bar{P}_{x,y}) \tag{3.12}$$

The angle of this phasor is now the measurement estimate for the phase angle in the projector frame. The projector column C can be recovered for camera pixel (x, y) with the inverse projector projection matrix $\boldsymbol{P}_p^{-1}$.

$$C_{x,y} = \boldsymbol{P}_p^{-1}\angle\bar{I}_{x,y} \tag{3.13}$$

3.4.2. ToF Forward Model

While SL illumination is spatially coded, but not time resolved, ToF measurements are not spatially coded but modulated over time. ToF systems can be generalized to the phasor imaging model per Gupta et al. [**40**]. When using a phasor representation, the high-frequency time dependence of the illumination can be simplified to a steady-state phase measurement, which in turn allows conventional light transport analysis. We will again use the path tracing approach, but we will assume unity emission and accumulate the throughput and length of each path:

$$P^{throughput} = \rho(\mathrm{s}_d \rightarrow \mathrm{s}_{d-1} \rightarrow \mathrm{s}_{d-2}) G(\mathrm{s}_d \leftrightarrow \mathrm{s}_{d-1}) \times \left(\prod_{j=1}^{d-2} \frac{\rho(\mathrm{s}_{j+1} \rightarrow \mathrm{s}_j \rightarrow \mathrm{s}_{j-1}) |cos\theta_j|}{p_\omega(\mathrm{s}_{j+1} - \mathrm{s}_j)} \right) \tag{3.14}$$

$$P^{length} = \sum_{j=1}^{d} ||\mathrm{s}_j - \mathrm{s}_{j-1}||_2 \tag{3.15}$$

The contribution of each path to a measurement is still weighted in the conventional manner, but the phasor associated with the path length, given a modulation wavelength λ, is summed instead of the value:

$$\bar{I}_{x,y} = \sum_{n=1}^{N} \sum_{d=1}^{D} P_{x,y}^{throughput} e^{i \frac{2\pi}{\lambda} P_{x,y}^{length}} \tag{3.16}$$

The phase of this accumulated complex value now can be converted to a distance estimate using the modulation wavelength. In the case of multi-wavelength modulation, such as that employed by the Microsoft Kinect, each traced path can be modulated by that wavelength prior to phasor accumulation to avoid multiple renders. With multiple frequencies, the effective unambiguous range of the measurement can be increased using the phase unwrapping technique given in [**40**].

Now that we can produce an estimate of a phasor image including the effect of multi-bounce interference, we can include this renderer in an optimization problem.

3.5. Optimizing Depth Estimates with Gradient Descent

Inverse problems dealing with shape recovery are a longstanding area of research in computer vision. Though early work such as the Horn shape-from-shading algorithm [**45**] solves closed-form partial differential equations without any explicit modeling of light transport or reflectance, the extension by Ikeuchi [**51**] does include the use of a reflectance model operator (i.e., a renderer) within an iterative optimizer to recover surface shape. Nayar [**99**] proposed using a raytraced renderer to solve for errors in shape-from-shading surface reconstructions caused by interreflections. In such cases the observed intensity given known lighting is dependent on both the surface shape and the reflectance properties of the shape. These properties can be iteratively updated, passed through a renderer, and compared to a measurement until a match is found.

More recently, the inverse raytracing approach described by Gkiolekas [**35**] uses full Monte Carlo path tracing in a volume to model and recover scattering parameters from real-world measurements.

Approaches have been proposed to mitigate the effects of multibounce interference by carefully selecting illumination or sampling strategies [**2, 105**]. In doing so, however, these techniques treat multibounce interference as measurement noise, rather than valid information about the scene shape and its effect on light transport. By contrast, we propose retaining these paths as valid, modulated information about the scene to be demodulated in an offline optimization stage.

Others have proposed offline optimization techniques, including [**32**], [**23**], [**33**], [**56**], and [**31**]. These techniques all make assumptions such as specular-only (two-bounce) paths, diffuse-only surfaces, or otherwise use a simplified rendering model. Taking inspiration from these approaches, we aim to show that a conventional unbiased rendering engine can be used in a similar manner to optimize surface shape acquisition with the possibility of handling arbitrary BRDF models and other complex phenomenon.

Recently, [**130**] and [**85**] propose training a convolutional autoencoder to learn the equivalence between depth images corrupted by multipath interference and the correct depth image. This technique has the potential to run at real time rates, but does not explicitly model the effects of light transport in the scene.

The two sets of forward renderers described in the previous section can be used in a least-squares objective, as Monte-Carlo rendering is an unbiased estimator of the light transport integral. We wish to minimize the error E between measured depth image $\hat{I}$ and the output of our forward operator T, given the depth estimate S. All other scene parameters Γ are assumed to be fixed upon initializing the renderer.

$$E = \min_{S} \left\| \hat{I} - T_{\Gamma}(S) \right\|_2^2 \tag{3.17}$$

3.5.1. Calculating Gradients

Since we wish to use the raytraced forward models in a gradient descent framework, it would be ideal to efficiently compute the gradient of the forward operator. The gradient in both the SL and ToF cases is the partial derivative of the output phase value with respect to each surface parameter. We can make a simple modification to the iterative operator in Eq. 1.2 so that each sub-path is connected to the target point prior to sampling the light source. This eliminates all but the first entry in the series sum comprising β because all previous path points do not depend on the newly added target point in the path. The emission value for the SLM mode, calculated in Eq. 3.10, and the path segment distance for ToF mode in Eq. 3.15 can be differentiated with respect to the target point. If the BRDF assigned to the surface is a differentiable model, the reflectance terms for the surface intersections on either side of the target point in the path can also be differentiated. However, the geometric coupling G contains a non-differentiable visibility term. This can be ignored under the assumption that a delta offset in the target point does not alter path connectivity due to occlusion.

However, under this assumption, even the first-order expansion of Eq. 1.2 under the chain rule produces three times as many terms as the original expression. The full expansion depends on other implementation details such as the BRDF model and normal

estimation technique used, but will always result in a per iteration operation count many times greater than the original path integral estimation.

Due to the impact this has on computation time, as well as the potentially non-negligible contribution from the non-differentiable visibility term (depending on the scene's occlusion density), a finite difference approach is more appropriate.

During optimization, the gradient is estimated by rendering the scene once with the current surface estimate, than once again for each parameter of the surface representation offset by some small delta. To improve sample distribution, the target point can be connected to each sub-path sample in either a path or bidirectional path tracing renderer.

The gradient for the objective objective in Eq. 3.17 is calculated with respect to each point parameter s on the estimated surface.

$$\begin{aligned} \nabla E &= \left[\frac{\partial E}{\partial \mathrm{s}_k}, \frac{\partial E}{\partial \mathrm{s}_k}, \cdots, \frac{\partial E}{\partial \mathrm{s}_k} \right] \qquad S_k \in S \\ \frac{\partial E}{\partial \mathrm{s}_k} &\approx \frac{1}{\delta} \left(\left\| \hat{I} - T_\Gamma(S) \right\|_2^2 - \left\| \hat{I} - T_\Gamma(S + \delta(\mathrm{s}_k)) \right\|_2^2 \right) \end{aligned} \tag{3.18}$$

Where $\delta(\mathrm{s}_k)$ produces a small fixed offset at point s_k when added to surface estimate S.

We now have all the pieces necessary to perform gradient descent. A simple batch approach is presented in Algorithm 1. The fixed step size can also be replaced by any of the typical extensions such as momentum [**116**], Adagrad [**25**], or Adam [**63**]. Though these approaches can help avoid local minima in gradient descent optimizations, initialization quality is also an important factor. Because SL and ToF systems produce initial

estimates that are largely accurate, aside the bias due to multibounce interference, these measurements provide predictably high-quality initial estimates for the optimization as well.

Measurement noise also impacts convergence in gradient descent optimizations. Much research in the field of active 3D scanning, and imaging in general, has been devoted to reducing measurement noise, but no system will provide perfectly noise-free estimates with which to initialize the optimization. We test how convergence is affected by known noise levels in simulation in the following section.

Gradient descent problems of this sort have been explored at length in machine learning research, while the focus of this work is to highlight the fact that raytraced forward models of active 3D scanning system are well suited for this type of optimization. We leave further refinement of the optimization approach for this problem to others.

Algorithm 1 Gradient Descent Routine

$T \leftarrow \Gamma$
$S \leftarrow \text{Reproject}(\hat{I})$
for $e \in$ Epochs **do**
    current $\leftarrow T(S)$
    for $\text{s}_k \in S$ **do**
        offset $\leftarrow T(S + \delta(\text{s}_k))$
        $\text{gradient}[\text{s}_k] \leftarrow \left\|\hat{I} - \text{current}\right\|_2^2 - \left\|\hat{I} - \text{offset}\right\|_2^2$
    end for
    $S \leftarrow S$+gradient
end for
return S

3.6. Implementation

The forward models described in Section 3.4 are designed to be compatible with a variety of raytracing platforms, including those that are highly parallel or GPU based. Furthermore, since the model only relies on standard light transport integrator properties such as path length and projector radiance, it can be extended to volumetric, branched, and bidirectional path tracers. It can similarly handle any number of reflectance and scattering models, as well as geometric representations.

For the purposes of demonstrating this approach, we opted to use open-source tools that already support the capabilities we need and can be easily modified to operate in our proposed manner.

3.6.1. Rendering and Optimization Tools

We use the Mitsuba renderer [**55**], which is free, open-source software written in C++, with a Python interface. It is based on the algorithms presented in [**114**]. Though some performance gains could be realized by using a GPU renderer such as the Nvidia Optix framework used in Chapter 2, the ability to directly access intermediate results and support variables in Mitsuba while the renderer is in operation steered us in the CPU-based direction. Mitsuba also features a heightfield-defined surface representation based on [**134**] that is particularly well suited for our surface reconstruction purposes.

The algorithms in Section 3.4 were incorporated into the standard path and bidirectional path tracers in Mitsuba.

Scene setup, optimization, and profiling are implemented in Python using SciPy [**59**] and Mitsuba's Python interface. We set a heightfield surface to the target depth profile,

then run the renderer. After reprojecting this simulated measurement back to world space, we create a new scene with a heightfield surface set to the measurement. This surface has a bias relative to the target surface due to the multibounce interference in the measurement. We can now run the gradient descent algorithm, updating the surface iteratively until the measurement of the current surface estimate matches the initial measurement from ground truth. This procedure was illustrated in Chapter 1, Figure 1.2.

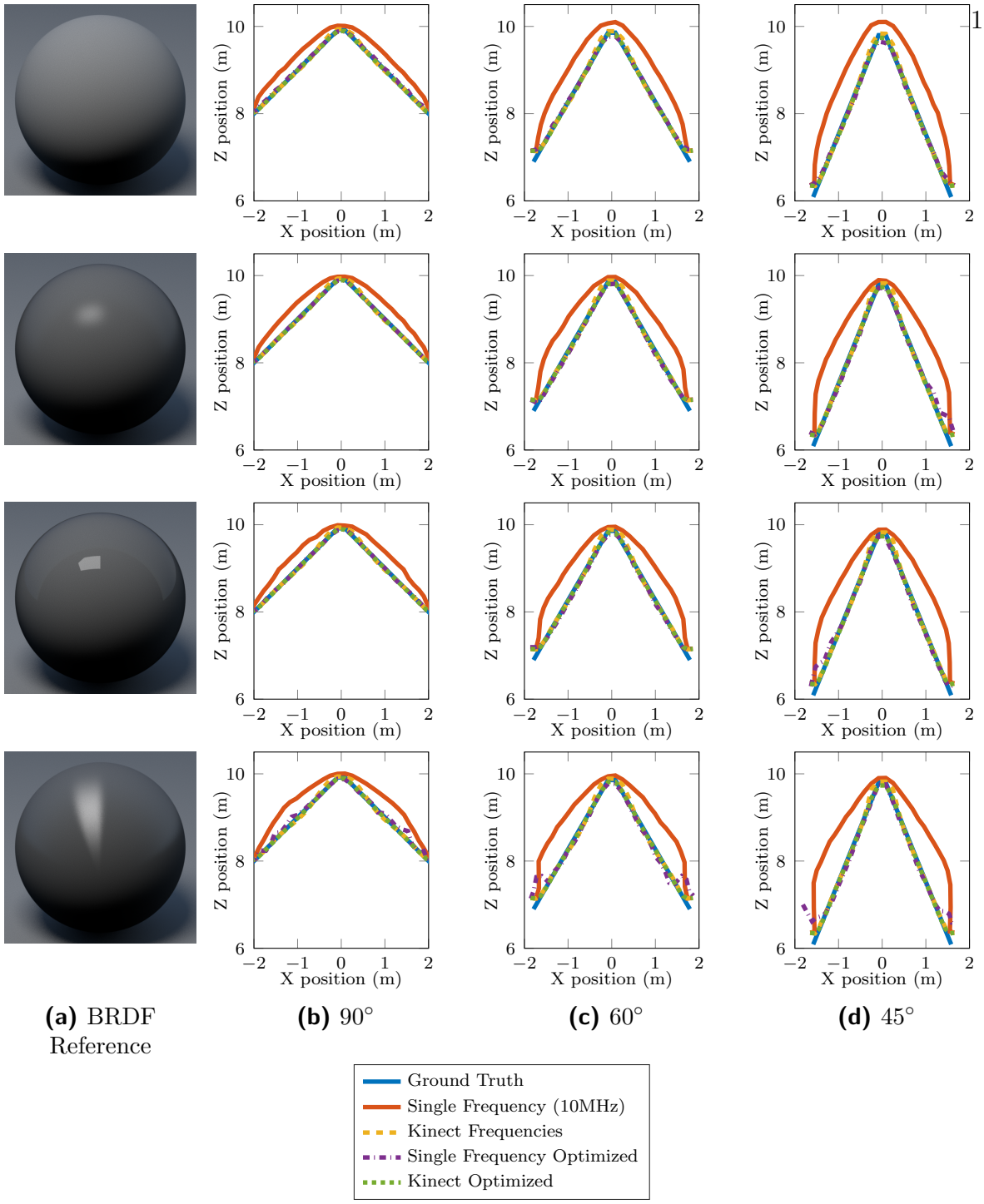
3.6.2. Simulated Results

Due to the difficulty in acquiring precise ground truth measurements over moderate-sized physical scenes, we conducted performance evaluations on two simulated time-of-flight systems, a single-frequency ToF sensor, and one with the three modulation frequencies used in the Microsoft Kinect.

Figure 3.7 shows the simulated measurements for 45°, 60°, and 90° diffuse concave v-groove shapes. The apex of the v-grooves are placed at the scene origin, and the camera is placed 10 meters away with a horizontal field of view of 30°. The solid lines depict the ground truth middle-row profile of the v-groove. Because of the long wavelength associated with the 10Mhz modulation (60 meters, for a scene with total depth of 10 meters), multibounce interference produces a large bias in the measurement. The simulated Kinect result, using multiple higher frequencies (16MHz, 80MHz, and 120Mhz, according to [**108**]), has reduced multibounce interference due to the filtering effects of the scene on higher frequencies described in [**40**]. We use the 10MHz simulated measurement as the initial estimate for our optimization, then run for 30 iterations with a fixed step size. The result of the simulation is similar to the quality of the measurement with the

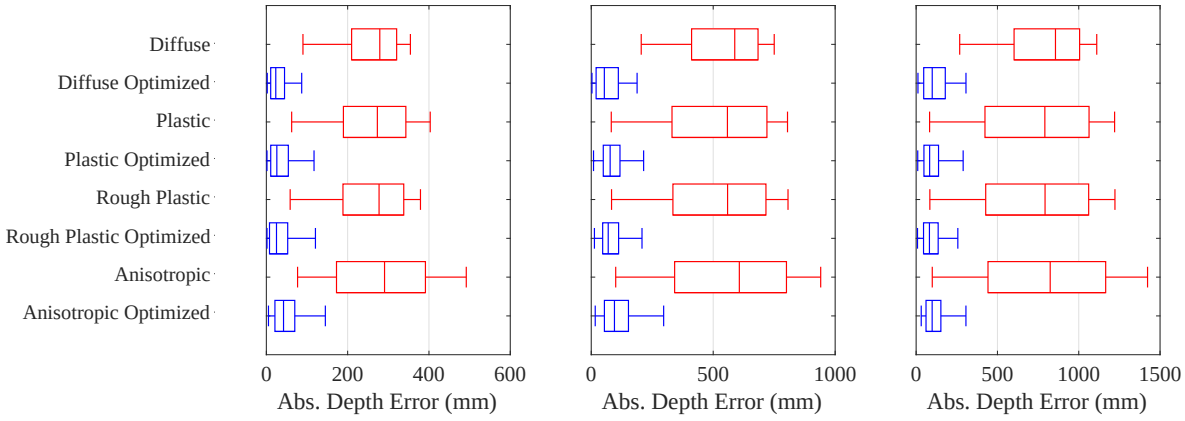
higher Kinect frequencies. We also use the simulated Kinect measurement to initialize the optimizer and again ran for 30 iterations. Though the improvement is not as great as with the single-frequency measurement, the optimization is still able to outperform the Kinect because of the remaining multibounce interference in the measurement. As these optimized results are all of high quality, it is more informative to refer to Figures 3.8 and 3.9 to compare relative error.

Evaluation of Dynamic Range: It is useful to evaluate the performance of a depth imaging system in terms of dynamic range, much like an intensity imaging system. A system that can resolve small depth displacements over a large unambiguous range has a high dynamic range, and is thus useful across a more general set of scenes. A low dynamic range system, conversely, can only resolve larger steps over a shorter range. The choice of modulation frequency directly affects the unambiguous measurement range, which is equal to half of the wavelength to account for the round trip distance to the scene. Disregarding measurement noise, longer wavelengths will result in stronger multibounce interference, however, so simply reducing modulation frequency is not a means to achieve higher dynamic range. Multifrequency techniques like the Kinect allow for a longer effective wavelength, the least common multiple of the individual wavelengths used. This approach retains the multibounce interference reduction associated with shorter wavelengths, while increasing the unambiguous range.



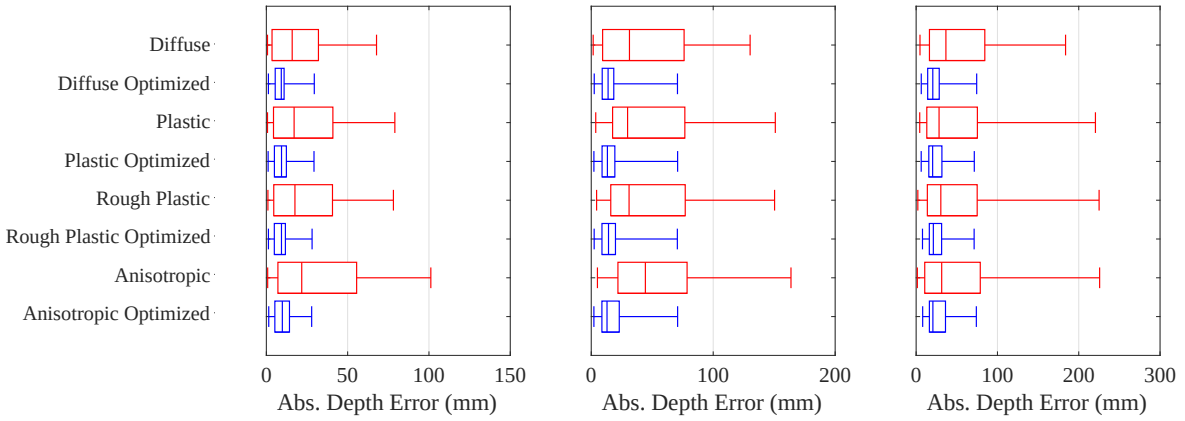
(a) BRDF Reference **(b)** 90° **(c)** 60° **(d)** 45°

Figure 3.7. **Simulated V-Groove Depth Profiles:** Simulated depth profiles for walls with known BRDFs meeting at 45°, 60°, or 90°. Top row: a diffuse-only BRDF. Second row: a physically based rough plastic BRDF. Third row: a glossy plastic BRDF. Bottom row: an anisotropic material. Profiles for ground truth, 10Mhz and Kinect simulated measurements, and optimized results are shown in each plot. Box-and-whisker plots of the associated error values with these plots are shown in Figures 3.8 and 3.9.



(a) 90° **(b)** 60° **(c)** 45°

Figure 3.8. **10MHz Depth Errors:** Box-and-whisker plots (5th and 95th percentiles, quartiles, and median values) for Z-axis error in each of the 10Mhz single-frequency results in Figure 3.7 (diffuse, glossy plastic, rough plastic, and anisotropic BSDFs). Error is calculated as the absolute distance in meters relative to ground truth. Box-and-whiskers for simulated measurements and reconstructions are shown on alternating lines in red and blue, respectively.



(a) 90° **(b)** 60° **(c)** 45°

Figure 3.9. **Kinect Frequency Depth Errors:** Box-and-whisker plots (5th and 95th percentiles, quartiles, and median values) for Z-axis error in each of the Kinect multi-frequency results in Figure 3.7 (diffuse, glossy plastic, rough plastic, and anisotropic BSDFs). Error is calculated as the absolute distance in meters relative to ground truth. Box-and-whiskers for simulated measurements and reconstructions are shown on alternating lines in red and blue, respectively.

		10Mhz	10Mhz Optimized	Kinect	Kinect Optimized
Diffuse	90°	49.34	601.97	448.82	1151.65
	60°	23.95	261.99	230.47	371.91
	45°	16.15	221.29	165.67	342.50
Glossy Plastic	90°	48.62	508.95	717.51	1134.53
	60°	24.69	212.05	175.16	284.49
	45°	16.72	210.96	106.04	342.10
Rough Plastic	90°	48.90	511.71	879.56	1130.79
	60°	24.54	210.36	199.65	284.14
	45°	16.71	209.31	118.22	341.85
Anisotropic	90°	41.45	391.59	216.16	1095.20
	60°	21.57	198.22	81.00	284.67
	45°	14.59	208.70	91.87	338.75

Table 3.1. **Dynamic Range of Simulated Results:** Dynamic range, calculated as the unitless ratio of unambiguous range to RMS Z-axis error in meters, is listed for each of the v-groove and BRDF combinations depicted in Figure 3.7. Optimized results always produce higher estimated dynamic range than the underlying measurements, and in most cases the single-frequency optimized results outperform the dynamic range of the Kinect measurements.

Dynamic ranges corresponding to the simulated results shown in Figure 3.7, calculated as unambiguous range divided by RMS Z-axis error in meters, are listed in Table 3.1. With Kinect simulation, the three modulation frequencies produce an effective wavelength equal to their least common multiple, or 37.5m. Many of the optimized single-frequency results have higher dynamic range than the underlying Kinect measurements, though the optimized Kinect measurements have the highest dynamic ranges by a large margin.

Evaluation of Multibounce Contribution: Varying amounts of interreflection will change the performance of our technique relative to the underlying measurements. The effect of this, as a function of ratio between direct and global illumination in the scene, is shown in Figure 3.10. Smaller angles (shown along the top horizontal axis in the figure) result in more multibounce interference, and thus a higher direct/global ratio. As the global/direct ratio increases, higher-frequency techniques like the Kinect produce higher dynamic range than single-frequency techniques, and our proposed optimization produces higher dynamic range than the technique used to initialize the optimization.

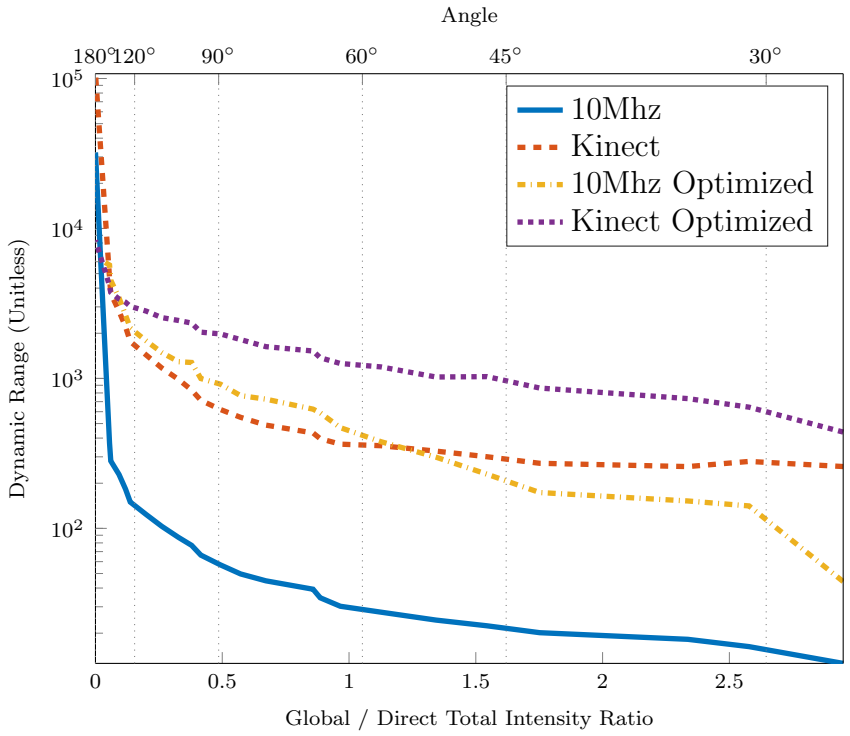
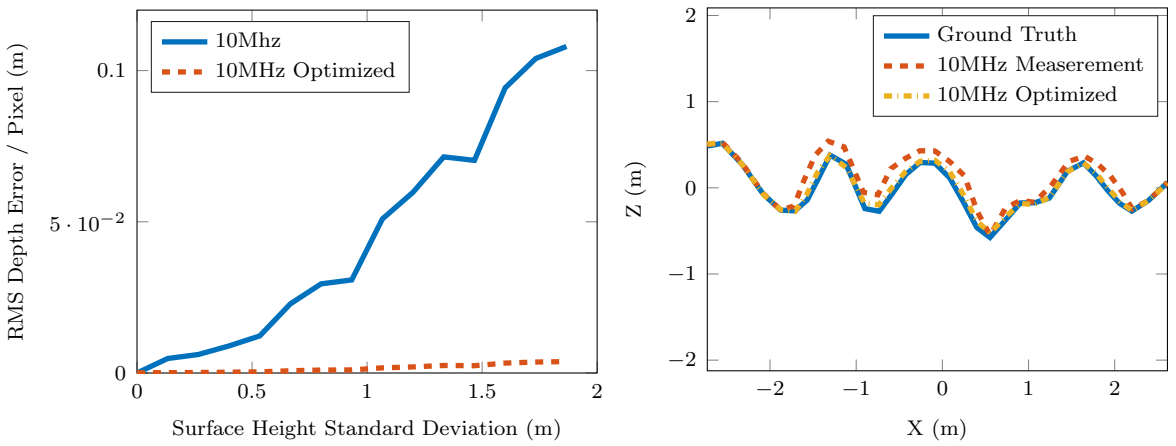


Figure 3.10. **Dynamic Range Versus Global/Direct Ratio:** Dynamic range for the simulated 10Mhz, Kinect, and optimized measurements, plotted as a function of the ratio between direct and global contributions to the scene intensity. Dynamic range is calculated as the total unambiguous range divided by the RMS depth value error relative to ground truth.



(a) Depth Error Versus Roughness **(b)** Centerline Profile

Figure 3.11. **Random Surface Performance:** In a), per-pixel root mean squared Z-axis error, relative to a ground truth random heightmap, plotted against the surface height standard deviation. The optimized result outperforms the simulated 10Mhz measurement. b) A centerline profile with a standard deviation of 2.0m showing ground truth, 10Mhz measurement, and optimized result.

Evaluation of Random Surfaces: The presence of concavities on a surface can be more generally described by the standard deviation of the surface. Surface deviations that are much smaller than the lateral resolution of a 3D scan are typically addressed with surface roughness models that take into account local scattering. Surface variations that can be spatially resolved also present challenges to accurate 3D acquisition at many scales from microscopic capture through terrain mapping. We can simulate the performance of our approach for surfaces of known standard deviations from a plane. Higher standard deviations will produce deeper concavities, and thus stronger multibounce interference. This effect is shown in Figure. 3.11. As surface height standard deviation increases, the error optimized results increases more slowly than non-optimized results because of the increasing presence of multibounce interference.

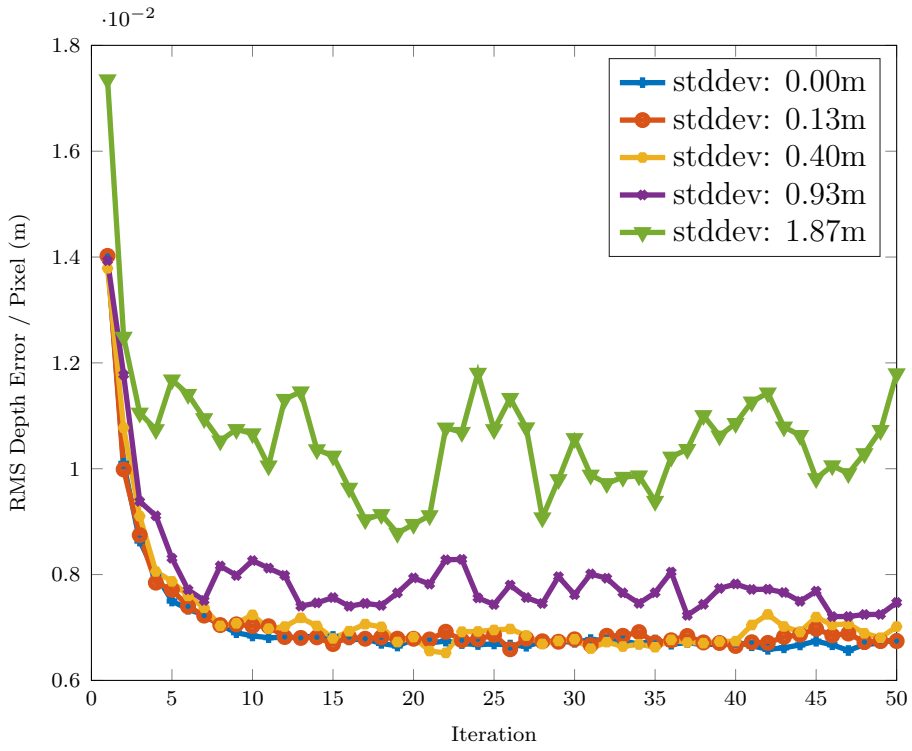
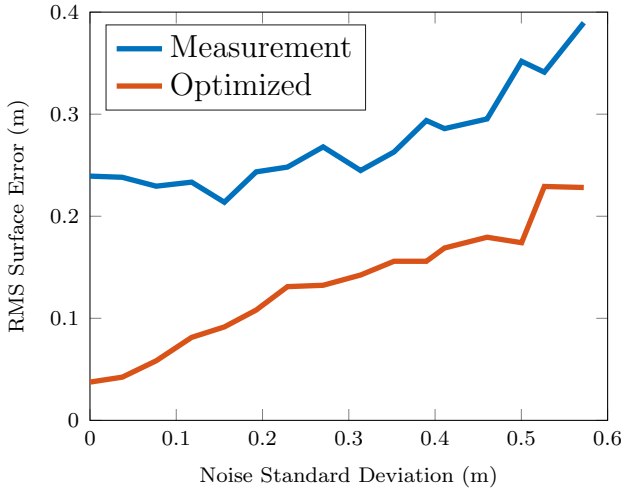
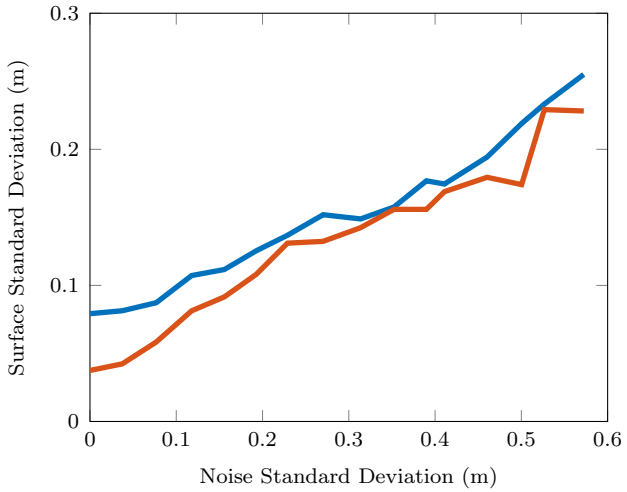


Figure 3.12. **Convergence with Measurement Noise:** Per-pixel root mean squared Z-axis error, relative to ground truth for a 90° v-groove, plotted versus optimization iteration. Profiles shown for noise magnitudes ranging from a standard deviation of 0m through 1.87m. The total z-axis range spanned by the v-groove is 2m.

Evaluation of Measurement Noise: We can also test the optimizer in simulation to check the effect of measurement noise on convergence. By approximating the effect of measurement noise as a random value drawn from a zero-mean Gaussian distribution added to the depth estimate of each pixel, we can control the standard deviation of the measurement noise relative to the depth range in the virtual scene. Figure 3.12 plots the per-pixel RMS Z-axis error relative to ground truth for a 90° v-groove over 30 iterations of the optimizer. The profiles show noise levels corresponding to 0% through 93% of the overall depth range in the image, or 2m. The value of the objective is consistently reduced during optimization, even for high levels of noise, which implies that this technique is effective for improving the accuracy of ToF estimates, but not necessarily the precision.

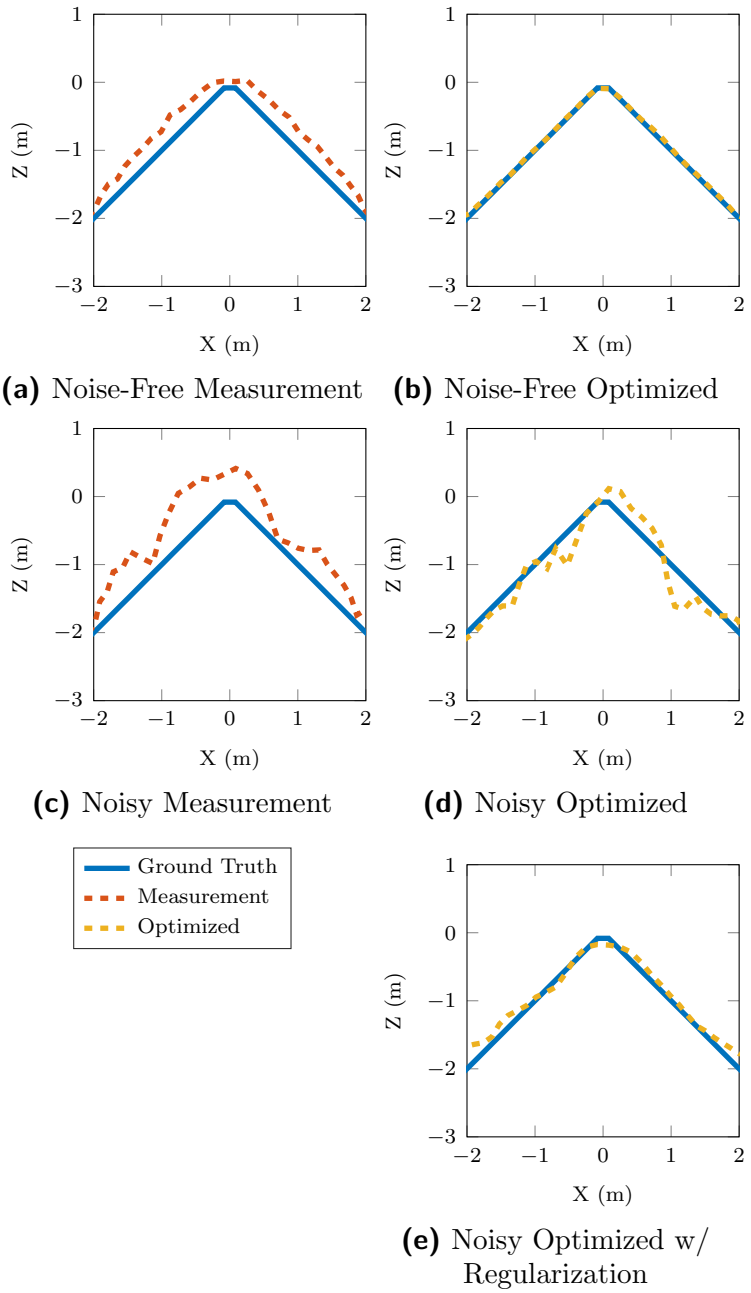


(a) Accuracy



(b) Precision

Figure 3.13. **Accuracy and Precision:** The accuracy of a simulated 10MHz 90° v-groove measurement, measured as RMS Z-axis error, can be consistently improved with increasing levels of measurement noise (a). Rendering-based optimization is able to correct for large-scale multibounce intereference despite the presence of high frequency noise in this case. The precision of this measurement, measured as the standard deviation of Z-axis error, cannot be improved by correcting for multibounce interference alone (b).



(a) Noise-Free Measurement **(b)** Noise-Free Optimized

(c) Noisy Measurement **(d)** Noisy Optimized

(e) Noisy Optimized w/ Regularization

Figure 3.14. **Accuracy and Precision - Example Profile:** In a), a simulated 10MHz 90° v-groove measurement, with low accuracy due to multi-bounce interference, but high precision due to lack of noise. In b), the noise-free measurement optimized, which shows an improvement in accuracy. In c), a simulated 10Mhz 90° v-groove measurement with added noise. In d), the optimized result given the noisy measurement. Here the systematic error has been reduced, improving accuracy, but the low precision due to measurement noise remains. To address this, an example using total variation regularization in the optimization loop is shown in (e).

Accuracy and Precision: Accuracy, measured as RMS Z-axis error, and precision, measured as the standard deviation of the Z-axis error, are shown for a range of noise levels added to simulated 10Mhz and 10Mhz optimized 90 v-groove measurements in Figure 3.13. The relative improvement in RMS error from the unoptimized to optimized simulation is notably consistent across increasing levels of noise. On the other hand, the precision of the optimized result shows no significant improvement. Both of these results can be explained intuitively: the optimization, by removing systematic error, can consistently improve accuracy, but does not denoise the result to improve precision. Centerline profiles demonstrating these effects in noise-free and noisy circumstances are shown in Figure 3.14.

There are numerous approaches to denoising that can be conjoined with the type of least squares objective function used in this optimization. As an example, Figure 3.14e shows a result obtained by including a total variation denoiser [**18**] executed as a separate step during each gradient descent iteration. In this case, the improvement in accuracy due to rendering-based optimization is retained while the high variance due to measurement noise is also reduced to improve precision. This result points to promising future directions exploring the many complementary optimization techniques that can provide additional advantages beyond the simple batch gradient descent employed in this thesis.

3.6.3. Experimental Results

To capture experimental equivalents to the v-groove simulations, we used a pair of 1m by 0.75m foam core boards aligned to marks measured on the floor corresponding to 45°, 60°, and 90° v-grooves. The Kinect ToF sensor was then placed along the centerline. This setup is shown in the 90° in Figure 3.15. Measurements were made using Matlab and a Kinect interface utility [**133**]. Reconstruction results are shown as profiles at the top of Figure. 3.16; renders of the measured surfaces are shown in the middle; and the resulting surface reconstructions are shown on the bottom.

For reconstruction speed, the Kinect sensor was aligned to the horizon so that vertical depth profiles in the scene were parallel to the image plane. This allows us to calculate a 1D gradient across the image. In order to obtain ground truth measurements, each of the two foam core boards was captured independently for each v-groove angle. Doing so removes multibounce interference, allowing depth error measurements relative to these accurately captured planes, which can in turn be used to estimate the dynamic range. Reconstructions shown here are downsampled to 1/8 of the native Kinect resolution.

The optimizer was run for 30 iterations using the Kinect measurement to initialize the surface estimate. The 90° optimization improves the Kinect dynamic range from approximately 474:1 to 1692:1. At 60° the optimization improves the approximate dynamic range from approximately 177:1 to 589:1. At 45° the optimization improves the approximate dynamic range from 148:1 to 773:1. These improvements are highly-dependent on scene content (and resulting amount of multibounce interference), but these results show that this technique can improve experimental measurements.

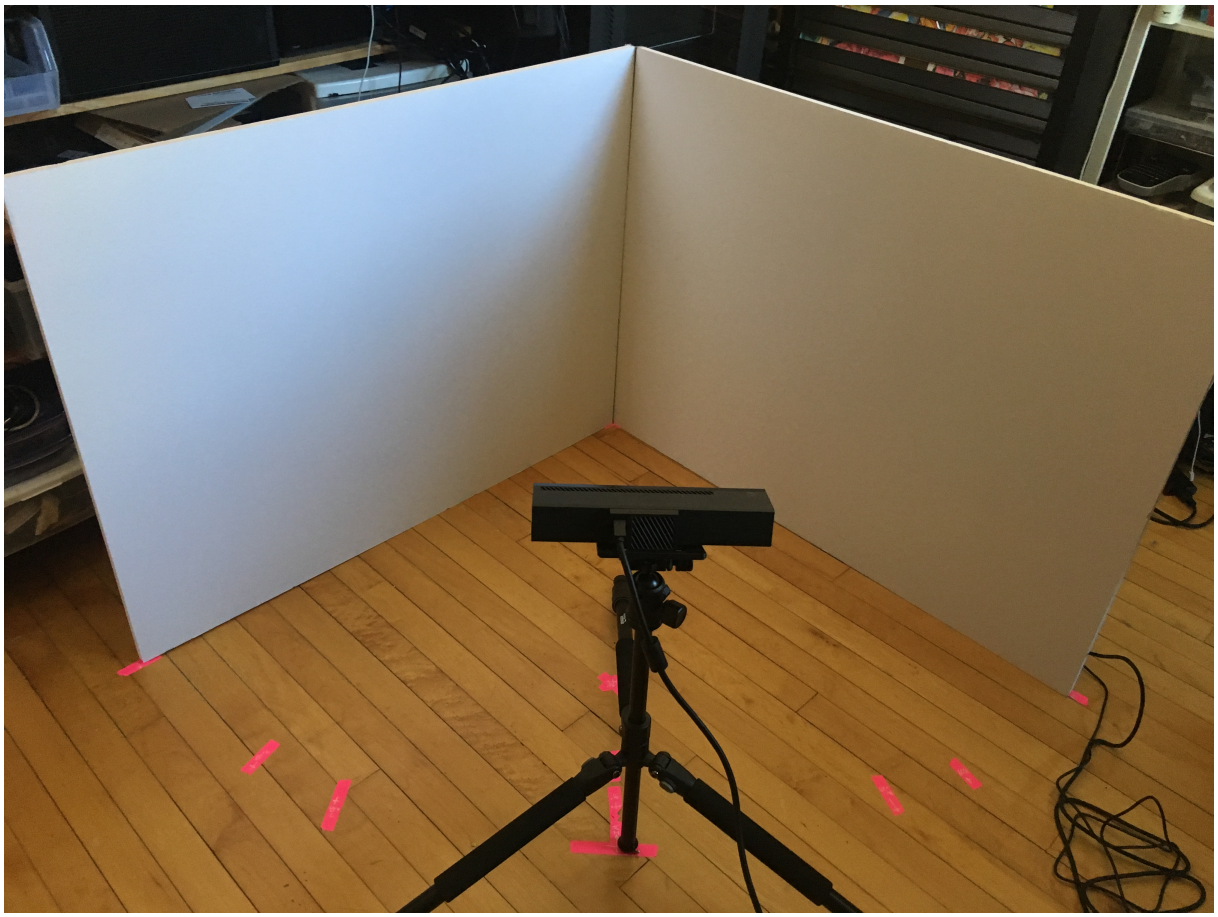
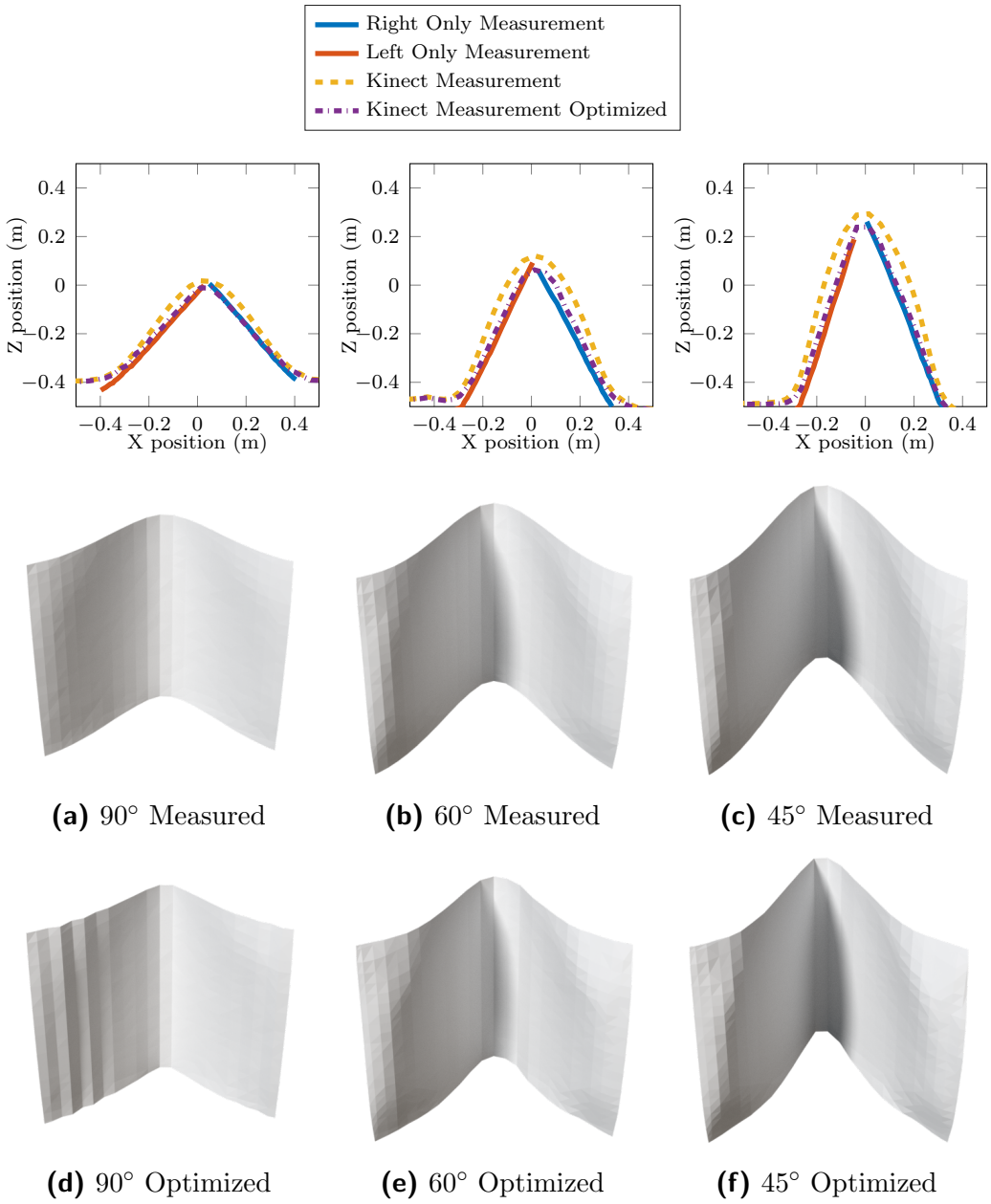


Figure 3.15. **Experimental Setup:** Foam core boards aligned to marks on the floor form a 90° v-groove. A Kinect ToF center is placed along the centerline and aligned with the center of the v-groove and the horizon. Kinect output was captured with a PC laptop. Board angles were then moved to 60° and 45° angles and captured with the Kinect.

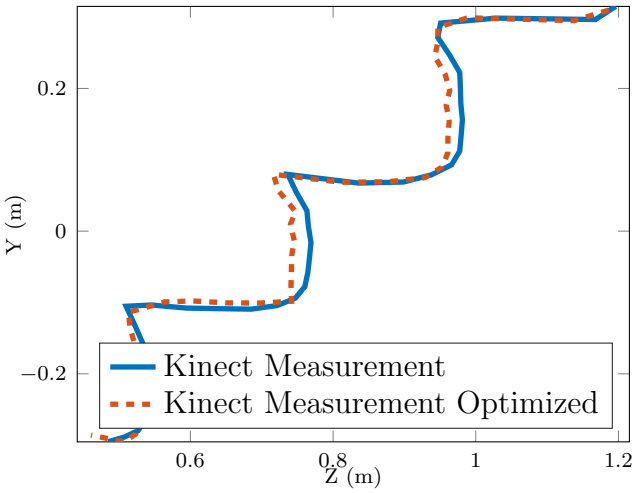


(a) 90° Measured **(b)** 60° Measured **(c)** 45° Measured

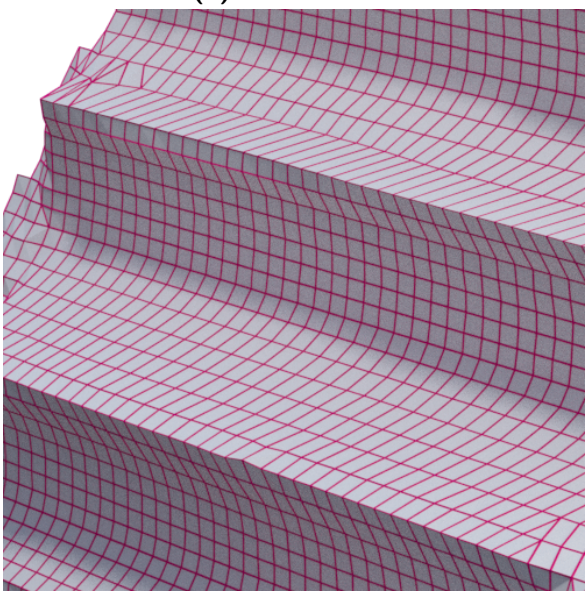
(d) 90° Optimized **(e)** 60° Optimized **(f)** 45° Optimized

Figure 3.16. **Experimental V-Groove Depth Profiles:** Top Row: Experimental depth results using Microsoft Kinect for capture, shown as middle-row profiles plotted in scene space, recovered from a physical scene containing foam core boards meeting at an angle of 45°, 60°, or 90°. Ground truth is approximated with two separate captures, one for each of the left and right sides of the v-groove to eliminate multibounce interference. Middle Row: Measurements converted to rectilinear mesh and illuminated with a directional light source from the upper left. Bottom Row: Optimized results converted to mesh and rendered. Column A) shows a 90° v-groove. Column B) the 60° v-groove. Column C) the 45°. The optimized result (30 iterations) consistently outperforms the physical measurement due to the algorithm's ability to account for multibounce interference.

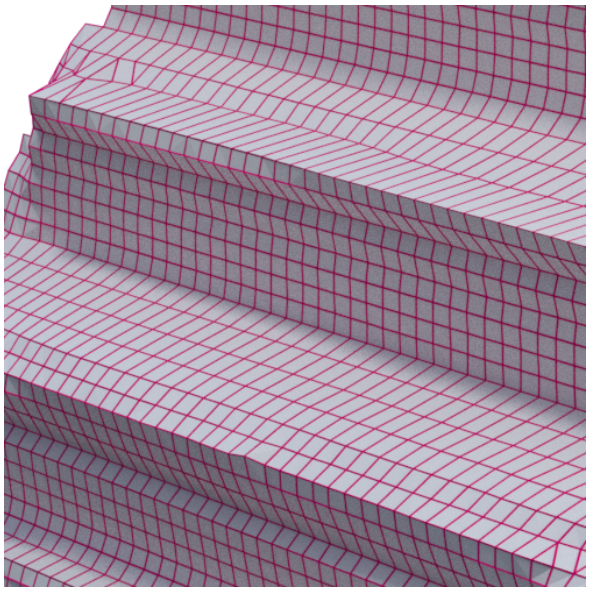
(a) Reference Photo



(b) Measured and Optimized Depth Profiles



(c) Rendered Mesh - Kinect Measurement



(d) Rendered Mesh - Optimized

Figure 3.17. **Experimental Capture, Stairs:** Experimental depth results using Microsoft Kinect for capture, shown as middle-row profiles plotted in scene space (b), recovered from a physical set of varnished wooden stairs, shown in (a) with an inset showing the tread and riser profile highlighted. A lit, rendered mesh produced from the raw Kinect measurement is shown in (c). The same treatment is applied to the optimized result in (d).

As a real-world test case, Figure 3.17 depicts the vertical centerline profile of a staircase and a reference photo. In the Kinect profile, the meeting point between each tread and riser is rounded out due to multibounce interference, while the optimized result recovers a more distinct right angle in these locations, while retaining the lip of each tread extending out past the riser. Furthermore, the stair material exhibits complicated reflectance characteristics. Parameters describing a semi-glossy BRDF were manually adjusted for this result. These types of parameters would be well suited for alternating minimization in future work.

3.6.4. Processing Time

The v-groove shapes used for Figures 3.7, 3.8, 3.10, and 3.16 were aligned to the vertical image axis so that depth values are uniform along columns in the image. The parameterization of each entire row can consequently be contained a single value. This drastically reduces computation time. Solutions were obtained on a 12-core Intel i7 desktop CPU. Single-frequency simulated results computed at 32x32 resolution with 1024 samples per pixel over 30 iterations had an average completion time of 5 minutes per result. Multifrequency Kinect simulations with the same parameters had an average completion time of 12 minutes per result.

Results in Figure 3.16 downsampled 8x from the 512x424 Kinect depth measurement, and were reconstructed with 30 iterations over a total wall-time duration of 11 minutes per result. The result in Figure 3.17 was downsampled 4x from the Kinect measurement.

To get a sense of processing times at full resolution, we can look to the work in [**35**], where an on-demand, cloud-based cluster comprising a total of 3200 cores was used. By

extrapolating the per-pixel, per-core runtime of our results, we can roughly estimate that a full resolution optimization of Kinect data (currently the highest-resolution time of flight sensor currently available), would take approximately 48 hours. While this is a significant amount of processing time, it is comparable to other offline simulation tasks used commercially. This thesis does not evaluate convergence for very low sample count renders to speed up solution time, nor does it evaluate more sophisticated gradient descent approaches to decrease convergence time. This thesis follows the hypothesis that inverse rendering optimization will increasingly be used in practice because of the recent and unprecedented availability and scale of parallel processing power, both in the form of cloud-based services and desktop GPUs.

3.7. Limitations and Future Directions

3.7.1. Generalized Active Scanning Optimization

This thesis looks toward future computational imaging devices that use rendering-based optimization as part of their core functionality. In order for this to happen, the methods described in the previous chapters will need to be generalized to subsume a broader variety of systems. The introduction of SL and ToF renderers, both requiring minimal modification to conventional path tracing, is a first step in this direction. Though a full comparison between the two models is outside the scope of this thesis, Figure 3.18 depicts the same v-groove reconstructions used in the previous section, but here with a projector-camera phase shifting setup. In this configuration, the period of the phase shift frequency is set to the column width of the projector. Here the equivalent to multifrequency Kinect

ToF is Micro Phase Shifting [**39**], which exchibits the same optical reduction of multi-bounce interference. Again, the optimized phase shift and Micro Phase Shifting results produce higher quality results.

These parallels can be directly extended to other multifrequency unwrapping techniques such as [**141**]. Other approaches, such as Gray coding or MC3D, will require additional thresholding operations as described in Section 3.4, but the basic path tracing approach and optimization principles will remain the same. The flexibility of the rendering-based optimization strategy presents a path toward developing a platform that generalizes to the full landscape of active 3D scanning techniques.

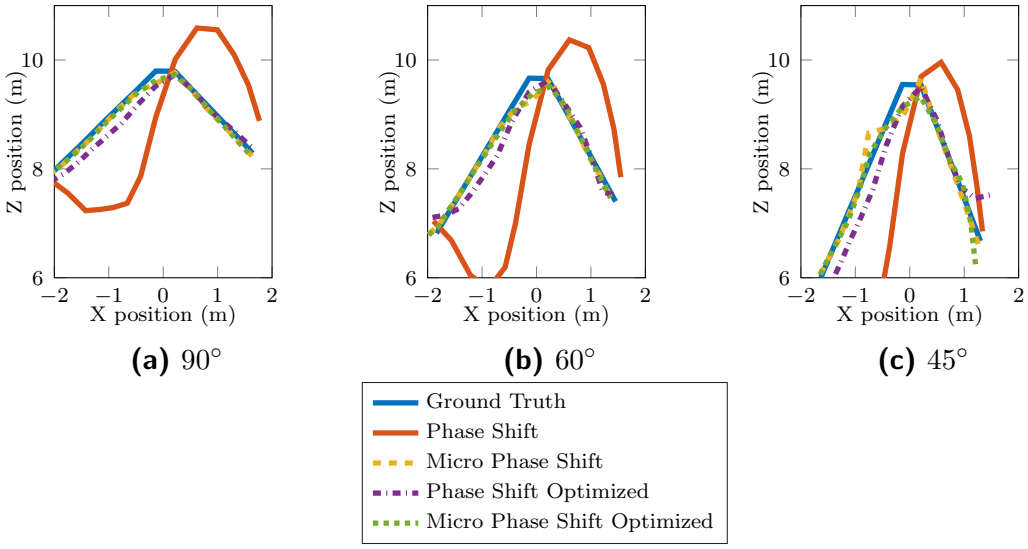


Figure 3.18. **Simulated SL Depth Profiles:** Simulated depth profiles for diffuse walls meeting at 45°, 60°, or 90°. Profiles for ground truth, phase shift, micro phase shift, and optimized results are shown in each plot. Like ToF results, the optimzed phase shift results approach the quality of micro phase shifting, which optically reduces multibounce interference.

3.7.2. BRDF Recovery

The forward model described in Section 3.6 is largely dependent on two groups of parameters: the surface shape and the BRDF of the surface. In order to support the full generality of the forward model, the optimizer would need to jointly solve for the BRDF. This problem is related to the volumetric scattering parameter recovery in [**35**]. In that work, the authors use inverse raytracing and gradient descent to recover scattering parameters by solving for the weights to be applied to a material dictionary. Such an approach can be applied to surface recovery as well, where the weighting of a dictionary of BRDFs is solved in an alternating minimization manner along with the surface shape. It is also possible that further estimation of parameters beyond surface scattering alone, such as subsurface scattering, could be approached in the same way. Of course, an increase in unknowns will require more computation to calculate the gradient with respect to each parameter, so low-dimensional reflectance models or dictionary-based approaches with a limited number of atoms will be beneficial in terms of computational complexity.

3.7.3. Occlusion and Multiple Viewpoints

A significant drawback of most SL and ToF systems is the inability to recover surface shapes that are occluded from the point of view of the camera or projector. Using multiple cameras and projectors can potentially measure the entirety of an object's visual hull, but in these cases, multibounce interference biases shape estimates in the same way as the single viewpoint case. Applying a raytracer-based optimization to multicamera systems can potentially alleviate this interference, as well.

3.7.4. Conclusion

This chapter summarized active 3D scanning techniques and their fundamental trade-offs. We introduced and experimentally tested a hardware-based approach to suppressing multibounce interference without significant increases in acquisition time. Then we modeled the multibounce interference problem in SL and ToF systems using conventional path tracing techniques from computer graphics. Because path tracing can account for physical effects that are ignored by canonical SL and ToF surface reconstruction algorithms, these raytracers can be used in an inverse rendering approach to recover surface shape estimates that are more accurate. We demonstrate this concept for time-of-flight using a simple gradient descent optimization on simulated and experimental measurements. We additionally quantify the performance gains over non-optimized ToF measurements in terms of depth error as a function of the ratio between global and direct intensities in the scene, as a function of measurement noise, and as a function of surface roughness. Together these simulations show that the inverse rendering approach can consistently improve the accuracy of 3D measurements because raytraced forward models can reliably account for the effect of multibounce interference in real-world scenes.

Though computation time remains a challenge, as it does for many optimization-based approaches in machine vision and computational imaging, we hope the benefits of rendering-based optimization used in conjunction with existing active 3D scanning systems enable new advances in the many fields that employ surface shape recovery, from cultural heritage studies to reality capture for cinema and virtual reality.

CHAPTER 4

Conclusion

In the previous two chapters, this thesis introduced a computational camera and display. The first of these, Focal Surfaces Displays, modify a conventional head-mounted display design by inserting a phase SLM in between an OLED screen and an objective lens. This phase modulator uses diffractive optics and a phase function optimizer to locally bend bundles of rays exiting the OLED so they converge on the user's retina at different apparent focal depths. Because of diffractive effects in the SLM's operation, however, the retinal image becomes distorted and aberrated when going through this process. To correct for this, the color OLED image must be transformed through the inverse of the SLM modulation. This distortion is spatially varying, so producing a linear mapping between the OLED and the retina would require a light transport matrix too vast to store in memory. This thesis solves this challenging problem with rendering-based optimization, which can accurately model the effects of the SLM distortion in an online manner that avoids the memory requirements of a light transport inversion approach. Chapter 2 concludes with experimental results showing that the focal surface display architecture can match or outperform competing techniques in terms of depth error and saliency metrics with less stringent time multiplexing requirements.

The second novel architecture, Motion Contrast 3D Scanning, approaches an entirely different set of computational imaging problems: efficient, robust 3D laser scanning. Laser

scanning is known to perform better than other techniques in the presence of ambient illumination and reflective surfaces, but is slow because conventional cameras are an inefficient measurement device for recovering first-bounce paths to the projector for triangulation. This thesis proposes a passive hardware technique to eliminate these extra measurements by using a motion contrast camera. Experimental results show that this configuration approaches laser scan quality while drastically reducing acquisition time. However, MC3D relies on specialized motion contrast hardware to achieve this, and in the process eliminates potentially useful multibounce measurements. A more widely-applicable method to address multibounce intereference should make use of these measurements across other active 3D methods.

This thesis again turns to rendering-based optimization for a solution. Using time-of-flight 3D depth sensors as a testbed due to their sensitivity to multibounce interference, this thesis introduces a raytraced image formation model designed for use in a gradient descent optimization to recover more accurate surface reconstructions for scenes with multibounce interference. Chapter 3 concludes with simulated and experimental results showing that the rendering-based optimization technique consistently improves surface estimates for concave scenes from single-frequency and multi-frequency time-of-flight measurements in terms of depth error and dynamic range.

This thesis uses these two distinct examples to highlight a path toward a more general method for improving the performance of computational cameras and displays by including rendering-based optimization in the design and operation of these systems.

References

[1] Microvision SHOWWX. https://web.archive.org/web/20110614205539/http://www.microvision.com/showwx/pdfs/showwx_userguide.pdf, 2011.

[2] Achar, S., Bartels, J. R., Whittaker, W. L., Kutulakos, K. N., and Narasimhan, S. G. Epipolar time-of-flight imaging. *ACM Trans. Graph. 36*, 4 (2017).

[3] Achar, S., and Narasimhan, S. G. Multi Focus Structured Light for Recovering Scene Shape and Global Illumination. In *ECCV* (2014).

[4] Agarwal, S., and Others. Ceres Solver. http://ceres-solver.org, 2012.

[5] Agin, G. J., and Binford, T. O. Computer description of curved objects. *IEEE Transactions on Computers 25*, 4 (1976).

[6] Akeley, K., Watt, S. J., Girshick, A. R., and Banks, M. S. A stereo display prototype with multiple focal distances. *ACM Trans. Graph. 23*, 3 (2004).

[7] Araki, K., Sato, Y., and Parthasarathy, S. High speed rangefinder. In *Optics, Illumination, and Image Sensing for Machine Vision II* (1988), vol. 850, International Society for Optics and Photonics, pp. 184–189.

[8] Arvo, J. *Analytic methods for simulated light transport*. PhD thesis, Yale University, 1995.

[9] Besl, P. Active, optical range imaging sensors. *Machine vision and applications 1*, 2 (1988).

[10] Bichler, O., Querlioz, D., Thorpe, S. J., Bourgoin, J.-P., and Gamrat, C. Unsupervised features extraction from asynchronous silicon retina through spike-timing-dependent plasticity. *IEEE IJCNN* (2011).

[11] Blender Foundation. Blender. http://www.blender.org, 2018.

[12] Blundell, B., and Schwartz, A. *Volumetric Three-Dimensional Display Systems*. Wiley-IEEE Press, 1999.

[13] Boivin, S., and Gagalowicz, A. Image-based rendering of diffuse, specular and glossy surfaces from a single image. In *Proceedings of the 28th annual conference on Computer graphics and interactive techniques* (2001), ACM, pp. 107–116.

[14] Bove, V. M. Display holography's digital second act. *Proceedings of the IEEE 100*, 4 (2012), 918–928.

[15] Brandli, C., Mantel, T. A., Hutter, M., Höpflinger, M. A., Berner, R., Siegwart, R., and Delbruck, T. Adaptive pulsed laser line extraction for terrain reconstruction using a dynamic vision sensor. *Frontiers in neuroscience 7* (2013).

[16] Burns, S. A., and Webb, R. H. Optical generation of the visual stimulus. In *Handbook of Optics, Third Edition Volume III*, M. Bass, Ed. McGraw-Hill, 2010.

[17] Cakmakci, O., and Rolland, J. Head-worn displays: A review. *Journal of Display Technology 2*, 3 (2006), 199–216.

[18] Chambolle, A. An algorithm for total variation minimization and applications. *Journal of Mathematical imaging and vision 20*, 1-2 (2004), 89–97.

[19] Cook, J. D. How to compute the soft maximum. `https://web.archive.org/web/20180528172422/https://www.johndcook.com/blog/2010/01/20/`, 2010.

[20] Couture, V., Martin, N., and Roy, S. Unstructured light scanning robust to indirect illumination and depth discontinuities. *IJCV 108*, 3 (2014).

[21] Curless, B., and Levoy, M. Better optical triangulation through spacetime analysis. In *IEEE ICCV* (1995).

[22] Damberg, G., Gregson, J., and Heidrich, W. High brightness HDR projection using dynamic freeform lensing. *ACM Trans. Graph. 35*, 3 (2016).

[23] Dorrington, A. A., Godbaz, J. P., Cree, M. J., Payne, A. D., and Streeter, L. V. Separating true range measurements from multi-path and scattering interference in commercial range cameras. In *Three-Dimensional Imaging, Interaction, and Measurement* (2011), vol. 7864, International Society for Optics and Photonics, p. 786404.

[24] Duarte, M. F., Davenport, M. A., Takhar, D., Laska, J. N., Sun, T., Kelly, K. F., and Baraniuk, R. G. Single-pixel imaging via compressive sampling. *IEEE signal processing magazine 25*, 2 (2008), 83–91.

[25] Duchi, J., Hazan, E., and Singer, Y. Adaptive subgradient methods for online learning and stochastic optimization. *Journal of Machine Learning Research 12*, Jul (2011), 2121–2159.

[26] Duchowski, A. T., and Others. Reducing visual discomfort of 3D stereoscopic displays with gaze-contingent depth-of-field. In *ACM Symposium on Applied Perception* (2014), pp. 39–46.

[27] Dunn, D., Tippets, C., Torell, K., Kellnhofer, P., Akşit, K., Didyk, P., Myszkowski, K., Luebke, D., and Fuchs, H. Wide field of view varifocal near-eye display using see-through deformable membrane mirrors. *IEEE TVCG 23*, 4 (2017), 1322–1331.

[28] Fernández, E. J., Prieto, P. M., and Artal, P. Adaptive optics binocular visual simulator to study stereopsis in the presence of aberrations. *J. Optical Society of America A 27*, 11 (2010).

[29] Fernandez, E. J., Prieto, P. M., Chirre, E., and Artal, P. Performance of a 6-Pi liquid crystal on silicon (LCoS) spatial light modulator under white light illumination for visual applications. *Imaging and Applied Optics* (2013).

[30] Foix, S., Alenya, G., and Torras, C. Lock-in time-of-flight (ToF) cameras: A survey. *IEEE Sensors Journal 11*, 9 (2011), 1917–1926.

[31] Freedman, D., Smolin, Y., Krupka, E., Leichter, I., and Schmidt, M. Sra: Fast removal of general multipath for tof sensors. In *ECCV* (2014).

[32] Fuchs, S. Multipath interference compensation in time-of-flight camera images. In *IEEE ICPR* (2010).

[33] Fuchs, S., Suppa, M., and Hellwich, O. Compensation for multipath in ToF camera measurements supported by photometric calibration and environment integration. In *ICVS* (2013).

[34] Ganapathi, V., Plagemann, C., Koller, D., and Thrun, S. Real time motion capture using a single time-of-flight camera. In *IEEE CVPR* (2010).

[35] Gkioulekas, I., Zhao, S., Bala, K., Zickler, T., and Levin, A. Inverse volume rendering with material dictionaries. *ACM Trans. Graph. 32*, 6 (2013).

[36] Glasner, D., Zickler, T., and Levin, A. A reflectance display. *ACM Trans. Graph. 33*, 4 (2014).

[37] Gupta, M., Agrawal, A., Veeraraghavan, A., and Narasimhan, S. G. A practical approach to 3D scanning in the presence of interre.ections, subsurface scattering and defocus. *IJCV 102*, 1-3 (2012).

[38] Gupta, M., Narasimhan, S. G., and Schechner, Y. Y. On controlling light transport in poor visibility environments. In *IEEE CVPR* (2008).

[39] Gupta, M., and Nayar, S. K. Micro phase shifting. *IEEE CVPR* (2012).

[40] Gupta, M., Nayar, S. K., Hullin, M. B., and Martin, J. Phasor imaging: A generalization of correlation-based time-of-flight imaging. *ACM Trans. Graph. 34*, 5 (2015).

[41] Gupta, M., Yin, Q., and Nayar, S. K. Structured light in sunlight. *IEEE ICCV* (2013).

[42] Hattori, K., and Sato, Y. Pattern shift rangefinding for accurate shape information. *MVA* (1996).

[43] Hillaire, S., Lécuyer, A., Cozot, R., and Casiez, G. Using an eye-tracking system to improve camera motions and depth-of-field blur effects in virtual environments. In *IEEE Virtual Reality* (2008), pp. 47–50.

[44] Hoffman, D. M., Girshick, R., Akeley, K., and Banks, M. S. Vergence-accommodation conflicts hinder visual performance and cause visual fatigue. *Journal of Vision 8*, 3 (2008), 33.

[45] Horn, B. K. Shape from shading: A method for obtaining the shape of a smooth opaque object from one view. *MIT Project MAC*, TR-79 (1970).

[46] Horn, E., and Kiryati, N. Toward optimal structured light patterns. *Image and Vision Computing 17*, 2 (1999).

[47] Hu, X., and Hua, H. High-resolution optical see-through multi-focal-plane head-mounted display using freeform optics. *Optics Express 22*, 11 (2014).

[48] Hua, H., and Javidi, B. A 3D integral imaging optical see-through head-mounted display. *Optics Express 22*, 11 (2014).

[49] Huang, F.-C., Chen, K., and Wetzstein, G. The light field stereoscope: Immersive computer graphics via factored near-eye light field displays with focus cues. *ACM Trans. Graph. 34*, 4 (2015).

[50] Huang, F.-C., Lanman, D., Barsky, B., and Raskar, R. Correcting for optical aberrations using multilayer displays. *ACM Trans. Graph. 31*, 6 (2012).

[51] Ikeuchi, K., and Horn, B. K. Numerical shape from shading and occluding boundaries. *Artificial intelligence 17*, 1-3 (1981), 141–184.

[52] Jacobs, R. J., Bailey, I. L., and Bullimore, M. A. Artificial pupils and maxwellian view. *Applied Optics 31*, 19 (1992).

[53] Jaffe, J. S. Computer modeling and the design of optimal underwater imaging systems. *IEEE Journal of Oceanic Engineering 15*, 2 (1990).

[54] Jaffe, J. S. Enhanced extended range underwater imaging via structured illumination. *Optics Express*, 12 (2010).

[55] Jakob, W. Mitsuba renderer, 2010. http://www.mitsuba-renderer.org.

[56] Jiménez, D., Pizarro, D., Mazo, M., and Palazuelos, S. Modeling and correction of multipath interference in time of flight cameras. *Image and Vision Computing 32*, 1 (2014), 1–13.

[57] Jimenez-Fernandez, A., Fuentes-del Bosh, J. L., Paz-Vicente, R., Linares-Barranco, A., and Jiménez, G. Neuro-inspired system for real-time vision sensor tilt correction. In *IEEE ISCAS* (2010).

[58] Johnson, P. V., Parnell, J. A., Kim, J., Saunter, C., Banks, M. S., and Love, G. D. Assessing visual discomfort using dynamic lens and monovision displays. *Imaging and Applied Optics* (2016).

[59] Jones, E., Oliphant, T., Peterson, P., et al. SciPy: Open source scientific tools for Python, 2018.

[60] Kajiya, J. T. The rendering equation. *ACM Trans. Graph. 20*, 4 (1986).

[61] Kanade, T., Gruss, A., and Carley, L. R. A very fast VLSI rangefinder. In *IEEE ICRA* (1991), pp. 1322–1329.

[62] Kim, H., Zollöfer, M., Tewari, A., Thies, J., Richardt, C., and Christian, T. InverseFaceNet: Deep Single-Shot Inverse Face Rendering From A Single Image. *arXiv preprint arXiv:1703.10956* (2017).

[63] Kingma, D. P., and Ba, J. Adam: A method for stochastic optimization. *ICLR* (2015).

[64] Klein, J., Peters, C., Martín, J., Laurenzis, M., and Hullin, M. B. Tracking objects outside the line of sight using 2d intensity images. *Scientific reports 6* (2016), 32491.

[65] Koninckx, T. P., and Van Gool, L. Real-time range acquisition by adaptive structured light. *IEEE PAMI 28*, 3 (2006).

[66] Konrad, R., Cooper, E. A., and Wetzstein, G. Novel optical configurations for virtual reality: Evaluating user preference and performance with focus-tunable and monovision near-eye displays. *ACM CHI* (2016), 1211–1220.

[67] Kotulak, J. C., and Schor, C. M. The accommodative response to subthreshold blur and to perceptual fading during the Troxler phenomenon. *Perception 15*, 1 (1986), 7–15.

[68] Kramida, G. Resolving the vergence-accommodation conflict in head-mounted displays. *IEEE TVCG 22*, 7 (2016), 1912–1931.

[69] Kress, B., and Starner, T. A review of head-mounted displays (HMD) technologies and applications for consumer electronics. *SPIE 8720* (2013).

[70] Lafortune, E. P., and Willems, Y. D. Bi-directional path tracing. *Compu-Graphics Proceedings* (1993), 145–153.

[71] Lange, R., and Seitz, P. Solid-state time-of-flight range camera. *IEEE Journal of quantum electronics 37*, 3 (2001), 390–397.

[72] Lanman, D., and Luebke, D. Near-eye light field displays. *ACM Trans. Graph. 32*, 6 (2013).

[73] Laude, V. Twisted-nematic liquid-crystal pixelated active lens. *Optics Communications 153* (1998), 134–152.

[74] Levin, A., Maron, H., and Yarom, M. Passive light and viewpoint sensitive display of 3d content. In *IEEE ICCP* (2016).

[75] Lichtsteiner, P., Posch, C., and Delbruck, T. A 128× 128 120 db 15 μs latency asynchronous temporal contrast vision sensor. *IEEE Journal of Solid-State Circuits 43*, 2 (2008).

[76] Liu, S., Hua, H., and Cheng, D. A novel prototype for an optical see-through head-mounted display with addressable focus cues. *IEEE TVCG 16*, 3 (2010), 381–393.

[77] Llull, P., Bedard, N., Wu, W., Tošić, I., Berkner, K., and Balram, N. Design and optimization of a near-eye multifocal display system for augmented reality. *Imaging and Applied Optics* (2015).

[78] Loper, M. M., and Black, M. J. OpenDR: An approximate differentiable renderer. In *ECCV* (2014).

[79] Love, G. D., Hoffman, D. M., Hands, P. J., Gao, J., Kirby, A. K., and Banks, M. S. High-speed switchable lens enables the development of a volumetric stereoscopic display. *Optics Express 17*, 18 (2009).

[80] MacKenzie, K. J., Dickson, R. A., and Watt, S. J. Vergence and accommodation to multiple-image-plane stereoscopic displays: "real world" responses with practical image-plane separations? *Journal of Electronic Imaging 21*, 1 (2012).

[81] Maiello, G., Chessa, M., Solari, F., and Bex, P. J. The (in)effectiveness of simulated blur for depth perception in naturalistic images. *PLOS ONE 10*, 10 (2015).

[82] Maimone, A., and Fuchs, H. Computational augmented reality eyeglasses. In *IEEE ISMAR* (2013).

[83] Mantiuk, R., Bazyluk, B., and Tomaszewska, A. Gaze-dependent depth-of-field effect rendering in virtual environments. In *Serious Games Development and Applications* (2011), Springer-Verlag, pp. 1–12.

[84] Mantiuk, R., Kim, K. J., Rempel, A. G., and Heidrich, W. HDR-VDP-2: A calibrated visual metric for visibility and quality predictions in all luminance conditions. *ACM Trans. Graph. 30*, 4 (2011).

[85] Marco, J., Hernandez, Q., Muñoz, A., Dong, Y., Jarabo, A., Kim, M. H., Tong, X., and Gutierrez, D. DeepToF: off-the-shelf real-time correction of multipath interference in time-of-flight imaging. *ACM Trans. Graph. 36*, 6 (2017).

[86] Márquez, A., Iemmi, C., Campos, J., and Yzuel, M. J. Achromatic diffractive lens written onto a liquid crystal display. *Optics Letters 31*, 3 (2006).

[87] Marran, L., and Schor, C. Multiaccommodative stimuli in VR systems: Problems & solutions. *Human Factors 39*, 3 (1997), 382–388.

[88] Marschner, S. R., and Greenberg, D. P. *Inverse rendering for computer graphics*. Cornell University, 1998.

[89] Matsuda, N., Cossairt, O., and Gupta, M. MC3D: Motion contrast 3d scanning. In *IEEE ICCP* (2015), IEEE.

[90] May, S., Droeschel, D., Holz, D., Fuchs, S., Malis, E., Nüchter, A., and Hertzberg, J. Three-dimensional mapping with time-of-flight cameras. *Journal of Field Robotics 26*, 11-12 (2009), 934–965.

[91] May, S., Werner, B., Surmann, H., and Pervolz, K. 3D time-of-flight cameras for mobile robotics. In *IEEE IROS* (2006).

[92] McQuaide, S. C., Seibel, E. J., Kelly, J. P., Schowengerdt, B. T., and Furness III, T. A. A retinal scanning display system that produces multiple focal planes with a deformable membrane mirror. *Displays 24*, 2 (2003).

[93] Mertz, C., Koppal, S. J., Sia, S., and Narasimhan, S. A low-power structured light sensor for outdoor scene reconstruction and dominant material identification. *IEEE International Workshop on Projector-Camera Systems* (2012).

[94] Moon, E., Kim, M., Roh, J., Kim, H., and Hahn, J. Holographic head-mounted display with RGB light emitting diode light source. *Optics Express 22*, 6 (2014), 6526–6534.

[95] Naik, N., Kadambi, A., Rhemann, C., Izadi, S., Raskar, R., and Bing Kang, S. A light transport model for mitigating multipath interference in time-of-flight sensors. In *IEEE CVPR* (2015).

[96] Narain, R., Albert, R. A., Bulbul, A., Ward, G. J., Banks, M. S., and O'Brien, J. F. Optimal presentation of imagery with focus cues on multi-plane displays. *ACM Trans. Graph. 34*, 4 (2015).

[97] Narasimhan, S. G., Nayar, S. K., Sun, B., and Koppal, S. J. Structured light in scattering media. In *IEEE ICCV* (2005).

[98] Nayar, S. K., and Gupta, M. Diffuse structured light. In *IEEE ICCP* (2012).

[99] Nayar, S. K., Ikeuchi, K., and Kanade, T. Shape from interreflections. *International Journal of Computer Vision 6*, 3 (1991), 173–195.

[100] Neil, M. A. A., Paige, E. G. S., and Sucharov, L. O. D. Spatial-light-modulator-based three-dimensional multiplanar display. *SPIE 3012* (1997), 337–341.

[101] Ng, R., Ramamoorthi, R., and Hanrahan, P. All-frequency shadows using non-linear wavelet lighting approximation. *ACM Trans. Graph. 22*, 3 (2003).

[102] Ni, Z., Bolopion, A., Agnus, J., Benosman, R., and Régnier, S. Asynchronous event-based visual shape tracking for stable haptic feedback in microrobotics. *IEEE Transactions on Robotics 28*, 5 (2012).

[103] Nocedal, J. Updating quasi-Newton matrices with limited storage. *Mathematics of Computation 35*, 151 (1980), 773–782.

[104] Oike, Y., Ikeda, M., and Asada, K. A CMOS image sensor for high-speed active range finding using column-parallel time-domain ADC and position encoder. *IEEE Transactions on Electron Devices 50*, 1 (2003), 152–158.

[105] O'Toole, M., Achar, S., Narasimhan, S. G., and Kutulakos, K. N. Homogeneous codes for energy-efficient illumination and imaging. *ACM Trans. Graph. 34*, 4 (2015).

[106] O'Toole, M., Mather, J., and Kutulakos, K. N. 3D Shape and Indirect Appearance By Structured Light Transport. In *IEEE CVPR* (2014).

[107] Padmanaban, N., Konrad, R., Stramer, T., Cooper, E. A., and Wetzstein, G. Optimizing virtual reality for all users through gaze-contingent and adaptive focus displays. *Proc. of the National Academy of Sciences 114*, 9 (2017).

[108] Pagliari, D., and Pinto, L. Calibration of kinect for xbox one and comparison between the two generations of microsoft sensors. *Sensors 15*, 11 (2015), 27569–27589.

[109] Paige, C. C., and Saunders, M. A. Lsqr: An algorithm for sparse linear equations and sparse least squares. *ACM transactions on mathematical software 8*, 1 (1982), 43–71.

[110] Papas, M., Jarosz, W., Jakob, W., Rusinkiewicz, S., Matusik, W., and Weyrich, T. Goal-based caustics. *Computer Graphics Forum 30*, 2 (2011).

[111] Park, J., and Kak, A. 3d modeling of optically challenging objects. *IEEE TVCG 14*, 2 (2008).

[112] Parker, S. G., Bigler, J., Dietrich, A., Friedrich, H., Hoberock, J., Luebke, D., McAllister, D., McGuire, M., Morley, K., Robison, A., and Stich, M. OptiX: A general purpose ray tracing engine. *ACM Trans. Graph. 29*, 4 (2010).

[113] Peli, E. Optometric and perceptual issues with head-mounted displays. In *Visual Instrumentation: Optical Design and Principles*, P. Mouroulis, Ed. McGraw-Hill, 1999.

[114] Pharr, M., Jakob, W., and Humphreys, G. *Physically based rendering: From theory to implementation*. Morgan Kaufmann, 2016.

[115] Posdamer, J., and Altschuler, M. Surface measurement by space-encoded projected beam systems. *Computer graphics and image processing 18*, 1 (1982).

[116] Qian, N. On the momentum term in gradient descent learning algorithms. *Neural networks 12*, 1 (1999), 145–151.

[117] Ramamoorthi, R., and Hanrahan, P. A signal-processing framework for inverse rendering. In *Proceedings of the 28th annual conference on Computer graphics and interactive techniques* (2001), ACM, pp. 117–128.

[118] Raskar, R., Agrawal, A., and Tumblin, J. Coded exposure photography: motion deblurring using fluttered shutter. *ACM Trans. Graph. 25*, 3 (2006).

[119] Ravikumar, S., Akeley, K., and Banks, M. S. Creating effective focus cues in multi-plane 3D displays. *Optics Express 19*, 21 (2011).

[120] Rolland, J. P., Krueger, M. W., and Goon, A. Multifocal planes head-mounted displays. *Applied Optics 39* (2000), 3209–3215.

[121] Salvi, J., Fernandez, S., Pribanic, T., and Llado, X. A state of the art in structured light patterns for surface profilometry. *Pattern Recognition 43*, 8 (2010).

[122] Scharstein, D., Hirschmüller, H., Kitajima, Y., Krathwohl, G., Nesic, N., Wang, X., and Westling, P. High-resolution stereo datasets with subpixel-accurate ground truth. In *GCPR* (2014), vol. 8753 of *LNCS*, Springer.

[123] Schwarte, R. *Handbook of Computer Vision and Applications, chapter Principles of 3-D Imaging Techniques*. Academic Press, 1999.

[124] Schwarte, R., Xu, Z., Heinol, H.-G., Olk, J., Klein, R., Buxbaum, B., Fischer, H., and Schulte, J. New electro-optical mixing and correlating sensor: facilities and applications of the photonic mixer device (pmd). In *Sensors, Sensor Systems, and Sensor Data Processing* (1997), vol. 3100, International Society for Optics and Photonics, pp. 245–254.

[125] Sen, P., Chen, B., Garg, G., Marschner, S. R., Horowitz, M., Levoy, M., and Lensch, H. Dual photography. *ACM Trans. Graph. 24*, 3 (2005).

[126] Shibata, T., Kim, J., Hoffman, D. M., and Banks, M. S. The zone of comfort: Predicting visual discomfort with stereo displays. *Journal of Vision 11*, 8 (2011), 11.

[127] Shiwa, S., Omura, K., and Kishino, F. Proposal for a 3-D display with accommodative compensation: 3DDAC. *J. Soc. Information Display 4*, 4 (1996).

[128] Sprague, W. W., Cooper, E. A., Tošić, I., and Banks, M. S. Stereopsis is adaptive for the natural environment. *Science Advances 1*, 4 (2015).

[129] Srinivasan, V., Liu, H.-C., and Halioua, M. Automated phase-measuring profilometry: a phase mapping approach. *Applied Optics 24*, 2 (1985).

[130] Su, S., Heide, F., Wetzstein, G., and Heidrich, W. Deep end-to-end time-of-flight imaging. In *IEEE CVPR* (2018).

[131] Sugihara, T., and Miyasato, T. System development of fatigue-less HMD system 3DDAC (3D display with accommodative compensation): System implementation of Mk.4 in light-weight HMD. *Image Engineering 97*, 467 (1998).

[132] Taguchi, Y., Agrawal, A., and Tuzel, O. Motion-aware structured light using spatio-temporal decodable patterns. *ECCV* (2012).

[133] Terven, J. R., and Córdova-Esparza, D. M. Kin2. a Kinect 2 toolbox for MATLAB. *Science of Computer Programming 130* (2016), 97–106.

[134] Tevs, A., Ihrke, I., and Seidel, H.-P. Maximum mipmaps for fast, accurate, and scalable dynamic height field rendering. In *Proceedings of the 2008 symposium on Interactive 3D graphics and games* (2008), ACM, pp. 183–190.

[135] Veach, E., and Guibas, L. J. Metropolis light transport. In *Proceedings of the 24th annual conference on Computer graphics and interactive techniques* (1997), ACM Press/Addison-Wesley Publishing Co., pp. 65–76.

[136] Viirre, E. S., Pryor, H., Nagata, S., and Furness III, T. A. The virtual retinal display. In *Proceedings of Medicine Meets Virtual Reality* (1998), pp. 252–257.

[137] Villegas, E. A., González, C., Bourdoncle, B., Bonnin, T., and Artal, P. Correlation between optical and psychophysical parameters as a function of defocus. *Optometry & Vision Science 79*, 1 (2002), 60–67.

[138] Voelz, D. G. *Computational Fourier Optics: A MATLAB Tutorial*. SPIE Press, 2011.

[139] von Waldkirch, M. *Retinal projection displays for accommodation-insensitive viewing*. PhD thesis, ETH Zurich, 2005.

[140] von Waldkirch, M., Lukowicz, P., and Tröster, G. Spectacle-based design of wearable see-through display for accommodation-free viewing. In *Pervasive Computing* (2004).

[141] Willomitzer, F., and Häusler, G. Single-shot 3D motion picture camera with a dense point cloud. *Optics express 25*, 19 (2017), 23451–23464.

[142] Wu, W., Llull, P., Tošić, I., Bedard, N., Berkner, K., and Balram, N. Content-adaptive focus configuration for near-eye multi-focal displays. In *IEEE Multimedia and Expo* (2016).

[143] Yue, Y., Iwasaki, K., Chen, B.-Y., Dobashi, Y., and Nishita, T. Poisson-based continuous surface generation for goal-based caustics. *ACM Trans. Graph. 33*, 3 (2014).

[144] Zalevsky, Z. Extended depth of focus imaging: A review. *Journal of Photonics for Energy* (2010).

[145] Zannoli, M., Love, G. D., Narain, R., and Banks, M. S. Blur and the perception of depth at occlusions. *Journal of Vision 16*, 6 (2016), 17.

[146] Zhang, L., Curless, B., and Seitz, S. M. Rapid shape acquisition using color structured light and multi-pass dynamic programming. *International Symposium on 3D Data Processing Visualization and Transmission* (2002).

[147] Zhang, S., Weide, D. V. D., and Oliver, J. Superfast phase-shifting method for 3-D shape measurement. *Optics Express 18*, 9 (2010).