Occlusion-aware Multifocal Displays

Submitted in partial fulfillment of the requirements for
the degree of

Doctor of Philosophy
in
Department of Electrical and Computer Engineering

Jen-Hao Rick Chang

B.S., Electrical Engineering, National Taiwan University
M.S., Electrical Engineering, National Taiwan University

Carnegie Mellon University
Pittsburgh, PA

December 2019

Acknowledgments

I feel tremendously fortunate to have Aswin Sankaranarayanan and Vijayakumar Bhagavatula as my PhD advisors, and I cannot thank them enough for their support and guidance. I still remember the day I first met Aswin. It was in a meeting a few days after my arrival of the United States, and I had never seen a kick-down doorstop before. During my struggle to close the door, Aswin smiled and said:

"You are an engineer. You can figure this out."

The sentence has given me the confidence to tackle all the problems I faced during the thesis research, and I will always treasure it in my heart. Kumar taught me how to be a successful researcher and how to capture both the high-level ideas and low-level details of research. I have the luck to work with Kumar even after he became the director of Carnegie Mellon University Africa and moved to Kigali. I am sincerely grateful for all the freedom and support Aswin and Kumar provide. My research interest lies in the two distinct worlds of machine learning and optics. After spending my first three years working on machine learning, I got fascinated by all the wonderful capabilities that co-designing algorithm and optics can achieve, so I decided to dive into the world of optics. I had already prepared to be turned down, but Aswin and Kumar not only supported my decision but also have learned with me. Their positive attitude is the greatest present they gave me during my PhD. I could not have asked for finer and more amazing mentors.

I am grateful to Anat Levin for valuable advice, encouragement, and collaborations. Anat has installed in me a high standard of research and taught me how to use phase-only spatial modulators. I would like to thank Anthony Rowe for serving on my committee. Anthony's delightful and astute comments made my proposal exam an enjoyable experience. I want to thank Ioannis Gkioulekas for time-to-time optical advice and kindly lending his cameras and optical instruments.

I feel lucky to have the delightful, nutty, crazily smart lab mates, and I want to thank them all for making my PhD experience wonderful. Vishwanath Saragadam has made enjoyable all the late-night working in the lab. I will always remember the mystery of jumping-chair during Christmas 2017, and Vishwa, if you are reading, "How's life?" Yi Hua provides insightful artistic comments, advice and support. Chia-Yin Tsai is the best listener and friend to chat with. Zhuo Hui set the highest standard in the lab. Jian Wang showed me the way to building optical setups and taught me how to use digital micro-mirror devices. Byeongjoo Ahn keeps reminding me to graduate, and Michael De Zeeuw was a great travel companion for our data-capturing trip to the Marine Biological Laboratory. Eric He, Stephen

Siena, Zhiding Yu, Yongjune Kim, Andrew Fox, Yang Zou, Ibrahim Alkanhal, and Denis Guo gave me valuable comments in our group presentations. Anqi yang, Wei-Yu Chen, Kuldeep Kulkarni, Vijay Rengarajan, Shigeki Nakamura, Tejas Gokhale, Suren Jayasuriya, Joe Bartels, Chao Liu, and Shumian Xin all made my PhD life colorful.

I want to thank Yu-Hsuan Kuo for all the supports and my friends Ting-Yao Hu, Chuya Cheng, Chun-Liang Li, Shih-En Wei, Yen-Chia Hsu, Shao-Wen Chiu, Olive Ho, Jean Chen, Emily Huang, Allie Chang, Chen-Hsuan Lin, Ju-Yi Lu, Evelyn Yang, and Peiyu Chen for the wonderful time in Pittsburgh. Ting-Yao and Chuya were the best roommates, Chun-Liang was a great half roommate and collaborator, and Shao-Wen is the best cook I know in Pittsburgh. Thanks to Fred Lin, Stephanie Lai, Mao-Pang Lin, Ting-Po Lee, Ko-Wei Ma, Tsung-Chuan Chen, Annie Chen, Erh-Kang Tsao, Ming-Po Chang, Yu-Han Lay, Yuhan Chen, Chiu-Yi Wu for the great time in California.

I want to thank my mentors during my internship in Nvidia and Oculus. Sean Pieper and Seungseok Oh patiently taught me details of the camera pipeline and the auto-focusing mechanism. Ningfeng Huang taught me about light guides and Fourier modal method. Steve Lansel and Jennifer Gille taught me about human vision system and how to conduct user studies.

The dissertation and my PhD study are supported by Dean fellowship of Department of Electrical and Computer Engineering, Bertucci fellowship, Adobe Research and Systems, ARO (grant number W911NF-16-1-0441), and National Science Foundation (grant number CCF-1652569). Their support enables me to focus on research, and I am truly grateful.

Finally, I would like to thank my family, Yung-Chi Chang, Feng-Yueh Lin, and Yi-Wen Chang. Dad's passion from medical to mechanics has inspired my curiosity on a wide range of topics. Mom's regular reminders on health and Sis's constant care has always been my support. They have provided me steady encouragement and love through high and low. This dissertation is dedicated to them.

Abstract

The goal of three-dimensional (3D) displays is to recreate reality by satisfying all perceptual cues used by the human visual system. While many perceptual cues can be replicated by showing 2D images to our eyes, the accommodation cue, or the change of the focal length of the ocular lens, is very difficult to satisfy with today's 3D displays. This inability to support the focusing of the eyes causes a problem called the vergence-accommodation conflict, which results in visual discomfort after long periods of use.

Multifocal displays satisfy the accommodation cue by displaying content on multiple virtual planes, each at a different depth. However, current designs of multifocal displays suffer from a limited number of focal planes and their inability to block light. The small number of focal planes significantly reduces the supported depth range of multifocal displays. The light leaking from the far focal planes also dramatically reduces the contrast of the image formed on the retina and weakens the occlusion cue — another important perceptual cue used by the human visual system to estimate depth.

This dissertation focuses on solving the two limitations of multifocal displays — the paucity of focal planes and the weak occlusion cue. Specifically, we design and build a multifocal display that can generate a dense focal stack — with an order of magnitude increase in the number of focal planes over existing works. To create proper occlusion cues, we endow multifocal displays with a novel capability to tilt the light emitted by each pixel. We show that the capability enables multifocal displays to generate occlusion cues without losing spatial resolution. The dissertation also contributes to the theoretical understanding of multifocal displays. We analyze the domain of light fields that can be generated by multifocal displays and characterize multifocal displays in terms of their depth-of-field, spatial resolution, and the required number of focal planes.

The proposed methods enable natural accommodation and occlusion cues that are critical for an immersive virtual world. Virtual and augmented reality (VR/AR) devices stand to benefit significantly from the advancements made in the dissertation. Moreover, all of the proposed methods require only simple modifications to existing AR/VR displays and are computationally and bandwidth-efficient. In this sense, the technologies are timely and could pave the way to a more immersive AR/VR experience.

For Mom, Dad, and Sis

Contents

List of Figures

List of Tables

1 Introduction

Reality leaves a lot to the imagination.

— John Lennon

Have you ever felt dizzy or nauseous after watching a 3D movie or using a virtual reality headset? This was probably because you were subconsciously able to detect the subtle differences between the virtual 3D scene presented to you and the real world.

The holy grail for 3D displays is to produce a scene that, to our eyes, is indistinguishable from reality. To achieve the goal, the display would need to deceive all perceptual cues that the human visual system uses to sense the world. Our eyes perceive a wide gamut of colors, a broad range of intensities, and numerous cues to perceive our surroundings. Among all perceptual cues, arguably the hardest to deceive is the depth perception capability. Despite significant advances in display technology, simultaneous deception of all perceptual depth cues is still beyond the reach of most displays. As it turns out, this has immense implications for 3D TVs, movies, virtual and augmented reality (VR/AR) devices.

Human visual system perceives depth with multiple cues. These cues utilize the information hidden in the perceived images and the states of our eyes. Each of the cues provides vital information about our surroundings and together form a coherent view of the world. If a 3D display fails to generate any depth cue or generates cues that are incoherent with others, not only does the visual immersion deteriorate, our ability to perceive depth is also adversely affected . Often, this results in physical discomfort and temporary loss of our depth perception [Hoffman *et al.*, 2008, Vishwanath and Blaser, 2010, Watt *et al.*, 2005, Zannoli *et al.*, 2016].

1.1 The Missing Accommodation Cue in 3D Displays

Most depth cues can be generated by showing 2D images to our eyes. We render the virtual scene from the two perspectives of our eyes and show the rendered images to each of them. This is how existing 3D

displays work — by tracking our position and rendering the images accordingly. The recent technology development in tracking, imaging, and efficient rendering have made possible the VR industry, which has become a billion-dollar market since 2016 [Zion Market Research, 2019].

While most depth cues are easily satisfied by showing 2D images to our eyes, the accommodation cue, which refers to the change of the focal length of the ocular lens, is very difficult to deceive. Even though existing 3D displays fail to generate the single depth cue, studies have shown that failing to produce the accommodation cue not only reduces the immersion of the virtual world but also can lead to physical discomfort [Kooi and Toet, 2004, Lambooij *et al.*, 2009].

The importance of the accommodation cue can be understood from Figure 1.1. In the real world, when our eye focuses on a particular depth, all objects at the depth will become sharp, and those at the other depths will be blurred. The focusing and defocusing of objects at different depths provide important visual cues to our brain for inferring the depths of the objects [Held *et al.*, 2012]. When a 3D display fails to generate the accommodation cue, everything in the scene comes in-and-out of focus simultaneously. This phenomenon creates a false illusion that they all lie on the same depth, like the images shown in Figure 1.1(b). In this case, focusing on virtual objects does not necessarily make them look sharp.

The missing accommodation cue often causes discomfort when using 3D displays. When our brain subconsciously detects the difference between the virtual world and the reality or the conflict between

Focus on the tree Focus on the house Focus on the tree Focus on the house

(a) In the real world (b) 3D display without the accommodation cue

Figure 1.1: **Effect of the missing accommodation cue.** The figure compares between (a) the real world and (b) a virtual world created by a 3D display that cannot produce the accommodation cue. Without the accommodation cue, everything in the scene comes into and out of focus together and focusing on an object does not necessarily make it sharp.

the accommodation cue, and the other depth cues, we will often feel dizzy and our capability to sense depth can temporally degrade. This problem is known as the vergence-accommodation conflict [Hoffman *et al.*, 2008, Vishwanath and Blaser, 2010, Watt *et al.*, 2005, Zannoli *et al.*, 2016].

In addition to the lack of focus support and physical discomfort, we will also show that supporting the accommodation cue is crucial for achievable resolution in 3D displays. In summary, the inability to generate the accommodation cue not only makes the virtual world very different from the real world but also causes visual discomfort that prevents us from using the 3D displays.

1.2 Accommodating the Human Eyes with Multifocal Displays

Existing methods to create the accommodation cue can roughly be categorized into light-field displays, holographic displays, varifocal displays, focal-surface displays, and multifocal displays. We will introduce each of the methods in Section 2.2.2 along with other methods that try to bypass the accommodation of our eyes. This dissertation builds upon multifocal displays, and the end result is a VR display that not only supports accommodation cues but also provides unprecedented immersion in the VR experience.

1.2.1 Multifocal Displays

Multifocal displays [Johnson *et al.*, 2016, Konrad *et al.*, 2016, Liu *et al.*, 2008, Liu and Hua, 2009, Llull *et al.*, 2015, Love *et al.*, 2009] have been shown to be effective in producing the accommodation cue [Koulieris *et al.*, 2017, MacKenzie *et al.*, 2012, 2010]. They present multiple focal planes, or the virtual images of the display panel, at different depths simultaneously in front of our eyes. By displaying virtual objects at their closest focal plane, multifocal displays effectively reduce potential focus mismatches that cause the vergence-accommodation conflict.

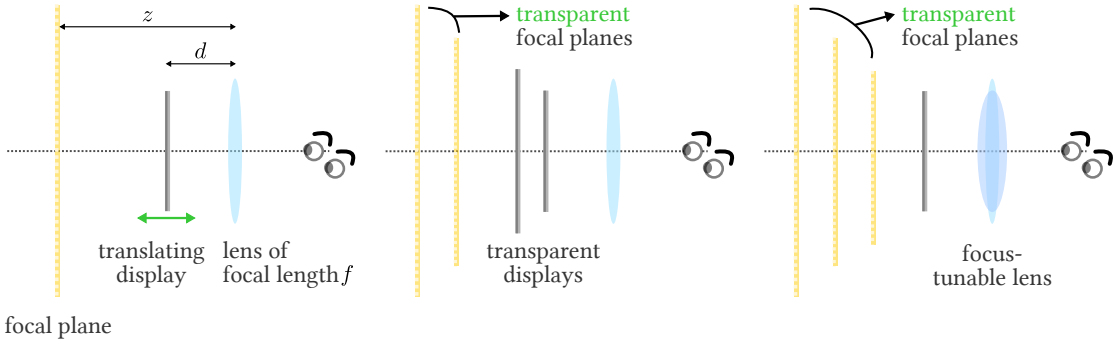
Consider the example setup shown Figure 1.2a. Here, focal planes are generated by placing a display panel in front of a lens. The depth of a focal plane is controlled by the thin-lens formula.

Let the distance between display panel and the lens be d and the focal length of the lens be f. The depth z can be calculated by

$$\frac{1}{d} - \frac{1}{z} = \frac{1}{f}. \tag{1.1}$$

By adjusting the distance d and the focal length f, we can control the depth of a focal plane.

The principle is used to design different kinds of multifocal planes. Figure 1.2 illustrates different examples of multifocal displays. Some multifocal displays use a translation stage to adjust the distance d [Akşit *et al.*, 2017, Shiwa *et al.*, 1996, Sugihara and Miyasato, 1998]; some setup multiple (transparent) display panels at different distances from the lens [Akeley *et al.*, 2004, Love *et al.*, 2009, Rolland *et al.*,



(a) Display with one focal plane (b) Transparent displays (c) Focus-tunable lens

Figure 1.2: Examples of multifocal displays

1999], some change the focal length of the lens using a focus-tunable lens [Chang *et al.*, 2018, Johnson *et al.*, 2016, Konrad *et al.*, 2016, Lee *et al.*, 2019, Liu *et al.*, 2008, Liu and Hua, 2009, Llull *et al.*, 2015, Love *et al.*, 2009, Padmanaban *et al.*, 2017, Rathinavel *et al.*, 2018], a deformable mirrors [Hu and Hua, 2014], a waveplate lens [Tabiryan *et al.*, 2015], a liquid-crystal lens [Jamali *et al.*, 2018a,b], or a variable-focus Moiré lens [Bernet and Ritsch-Marte, 2008]. Despite various methods to build multifocal displays, all of them show multiple focal planes simultaneously within a short duration to show a single VR frame.

1.2.2 Limitations of Existing Multifocal displays

Despite their ability to support the focus of our eyes, existing multifocal displays suffer from the following two limitations.

Insufficient Focal Planes

Despite the various ways to build a multifocal display, existing multifocal displays only have 3 to 5 focal planes. When using layers of transparent displays, the number of focal planes is limited by the weight and the energy consumption of the VR displays. When using a translating display panel or a focus-tunable lens to generate multiple focal planes, the speed of the translation stage and the tunable lens determines the number of the focal planes we can generate. The speed of a translation stage is often limited by the mass of the display panel and the heat it generates in order to avoid damaging the display or the user. For this reason, we are going to focus on focus-tunable lens, which requires much less moving parts than translation stages.

The speed of a focus-tunable lens is determined by its settling time. A typical focus-tunable lens has a settling time of 5 ms [Optotune, 2017, Varioptic, 2017]. Thereby existing multifocal displays can only show 200 focal planes per second or 3 to 5 focal planes per frame when operating at 40 to 60 frames per second. This is far too sparse to cover a depth range from 25 cm to infinity. As we will show in Chapter 4, the sparsely-separated focal planes significantly lowers the resolution of a multifocal display across the depth range.

Lack of Occlusion Cue and Low Contrast

In order for our eyes to simultaneously see the contents at different depths, the focal planes are transparent. However, this transparency of focal planes has two adverse effects. First, the display is incapable of satisfying occlusion cues since even small displacements of the eye will readily produce overlapping content. Second, the contrast of the display is significantly reduced, since the defocused far contents can bleed into near objects. Both of these effects are undesirable, in that, they reduce the immersion of the virtual scene.

Figure 1.3 shows an example of the lack occlusion in multifocal displays. We render images of a dinosaur standing in front of a grid and focus a camera on the dinosaur. In reality, light from the grid is blocked by the dinosaur, so we see a sharp image of the dinosaur. In a multifocal display, the content is shown on transparent focal planes. Therefore, we can see the light from the grid on a far focal plane leaking through the dinosaur, and the contrast is significantly reduced.

1.3 Main Contributions of the Dissertation

The dissertation enables a design for multifocal displays that can show dense focal stacks, produce occlusion cues, and have high contrast. Specifically, our multifocal display builds upon the following contributions.

- *Light-field analysis for multifocal displays.* The dissertation analyzes the space of realizable light fields of multifocal displays. It provides a mathematical understanding of multifocal displays. The findings enable us to determine the depth-of-field of a focal plane, bound the minimum and the maximum number of focal planes we need to deceive our eyes, and create the occlusion cue with free-form lenses.

- *Light intensity modulation for high frame-rate and high bit-depth projection.* The dissertation enables a design of multifocal displays that display thousands of focal planes per second. We propose a method

(a) Reality (b) Multifocal display (c) Proposed display

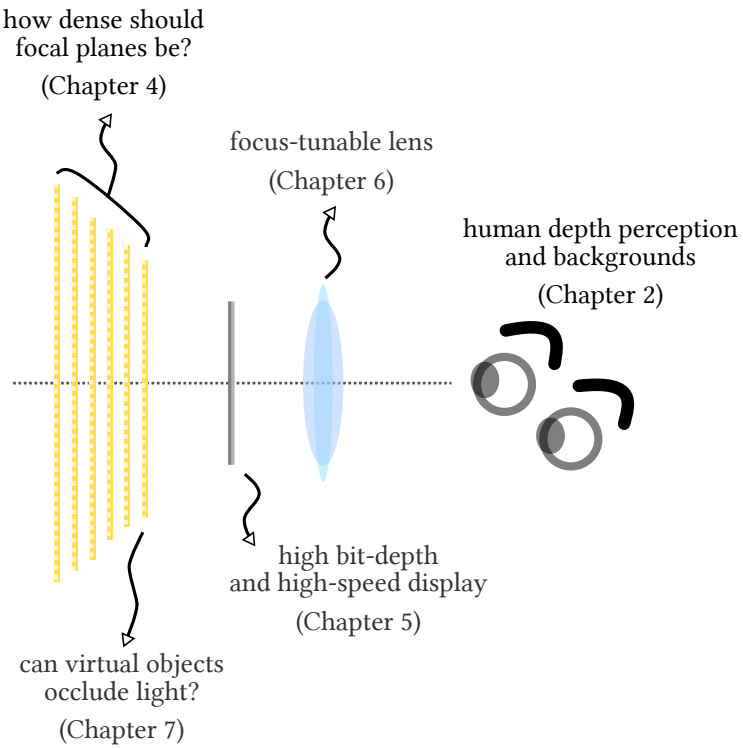
Figure 1.3: **Lack of occlusion in multifocal displays.** The figure shows rendered images of a scene composed of a dinosaur at 2 m and a grid at infinity. The camera focuses on the dinosaur with $f/22$. The proposed display uses the same contents as the multifocal display.

that utilizes light intensity modulation to efficiently achieve the goal.

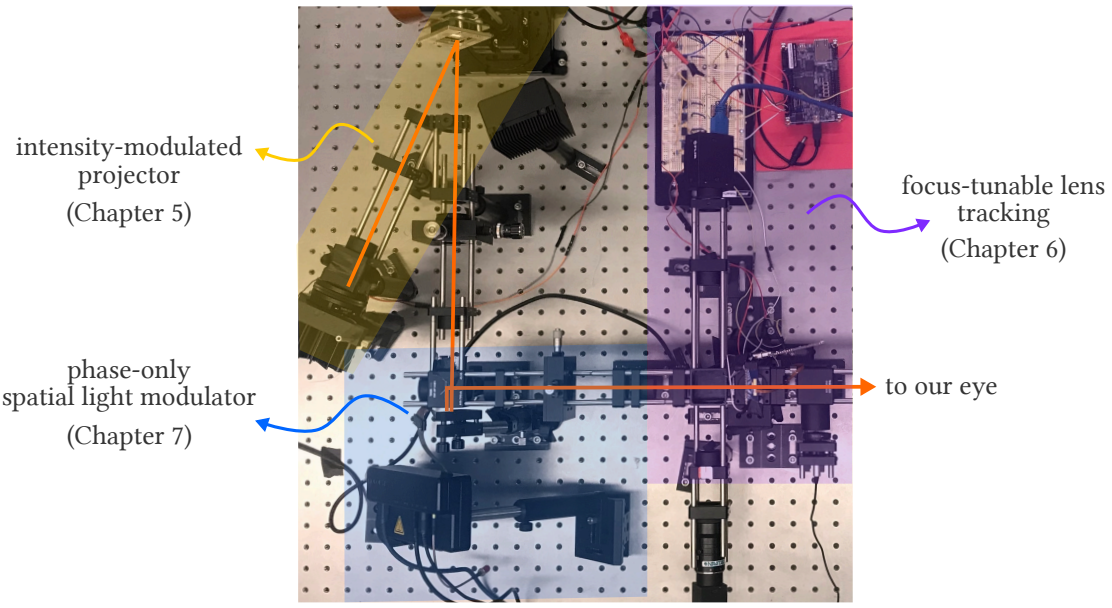
- *Dense focal stacks for multifocal displays.* The dissertation enables multifocal displays with dense focal planes by increasing the number of focal planes by an order of magnitude. We show that the ability to display dense focal stacks significantly improves the resolution of a multifocal display across a broad depth range.
- *Enabling the occlusion cue for multifocal displays.* The dissertation demonstrates how to generate occlusion cues in a multifocal display. The method uses free-form lenses created by a phase-only spatial light modulator (phase SLM) to manipulate the light field emitted by the display. This significantly improves the contrast and generates an illusion that the front focal plane can occlude light.
- *Lab prototype.* The dissertation presents a next-generation multifocal display which incorporates all the technologies developed in the dissertation. The photo of the prototype is shown in Figure 1.4b. The prototype provides a platform for validating all proposed ideas in the dissertation.

1.4 Roadmap of the Dissertation

This dissertation is written in a manner that it would interest researchers, students, and VR enthusiasts. I hope that the dissertation can serve as a useful manual/reference to building multifocal displays and the analysis of the tradeoffs when designing a multifocal display.



(a) Schematic view



(b) Lab prototype

Figure 1.4: Roadmap of the dissertation

Figure 1.4 gives an overview of the chapters in the dissertation and their connections to a multifocal display. The chapters can be read either sequentially or in any order.

- Chapter 2 introduces the perceptual depth cues used by the human visual system and how a typical 3D display fails to support the accommodation cue and causes the vergence-accommodation conflict. The chapter also introduces existing accommodation-supporting 3D display technologies and compares their advantages and limitations.
- Chapter 3 raises the questions that we need to answer before we design a multifocal display that can show dense focal stacks, produce occlusion cues, and has high contrast. The chapter also serves as a road map to the answers provided in the following chapters.
- Chapter 4 analyzes the light fields created by multifocal displays. The analysis will tell us why increasing the number of focal planes is important for multifocal displays. The chapter also establishes the relationship between spatial resolution and the density of focal planes.
- Chapter 5 shows how to enhance a typical projector to achieve a higher frame rate, higher bit-depth, and more vivid colors. The proposed projector will be a building block for the display panel of our multifocal display.
- Chapter 6 shows how to display thousands of focal planes per second, an order of magnitude improvement on existing multifocal display.
- Chapter 7 shows how we can design multifocal displays whose focal planes can block light, and hence, be occlusion-aware.
- Chapter 8 summarizes the lessons we have learned and points directions for future research on multifocal displays.

2 Depth Perception and 3D Displays

Before we dive into the details of the dissertation, let us understand how the human visual system perceives 3D information. For the dissertation, we are going to look at a simplified model that allows us to model the depth perception of the human visual system effectively. Nevertheless, this will help us understand the importance to generate the accommodation cue in VR displays and why it is challenging. At the end of the chapter, we will introduce typical VR displays and existing accommodation-supporting VR displays.

2.1 Human Depth Perception

In its most simplified analogy, we can think of the human visual system as two cameras connected by a brain. As a camera, human eyes has a lens and a sensor/retina. Our eyes are capable of fixating on the objects of interest, focusing on them, and sensing the images formed on the retina. The brain uses the image contents and the states of the eyes, *e.g.*, the angles of their rotation and the focal length of the ocular lens, to infer depth. The cues provided by the image contents form psychological depth cues, and those provided by the states of the eyes form physical depth cues.

2.1.1 Psychological Depth Cues

Based on the images seen by our two eyes, our brains extract various depth cues, which capture the interactions between objects at different depths [Geng, 2013].

- *Binocular disparity* is the difference in the images seen by each of our two eyes. The disparity between the two images increases as an object moves closer.
- *Occlusion* happens when a near object blocks parts of a distant object in the line of sight.

- *Motion parallax* is the relative motion of objects at different depths. When the eye moves, near objects move faster than far objects.
- *Perspective* is the projection of the 3D world on 2D images, like parallel lines meet at infinity.
- *Shading* is produced when the objects interact with the light sources. This cue enables us to infer both the lighting conditions and the shapes of the objects.
- *Relative size* is the change of the size in the appearance of an object when it moves. The same object looks larger when it is closer. The cue also helps most when similar objects appear at different depths in the scene.
- *Prior knowledge* about the size, the shape, the structure, or the relationship of common objects also provide strong depth information to the brain.

Since these depth cues are inferred from the images themselves, they can be evoked even when we are looking at 2D images or videos.

2.1.2 Physical Depth Cues

The state of our eyes is also driven by our understanding of the depth of the scene, real or virtual.

- *Vergence* refers to the rotation of our two eyes to fixate on the same point. Looking at close objects requires larger rotation, while staring at far objects require less rotation of our eyes.
- *Accommodation* is the action of focusing the ocular lens inside our eyes. Our eyes tighten/relax the muscles that control the focal length of its lens in order to form sharp images on the retina, and the depth of the object determines the required focal length.

The vergence cue can be generated simply by showing *2D images* with proper binocular disparity to each of our eyes. On the other hand, the accommodation cue is much more difficult to generate, since the focusing mechanism involves the object appearance at various depths. In order to generate accommodation cues, we need to simulate the *light rays* entering our eyes.

2.1.3 Strength of the Depth Cues

The human visual system relies on different depth cues when looking at objects at different depths. Figure 2.1 outlines their relative strength at different depths [Cutting and Vishton, 1995, Geng, 2013]. As can be seen, the occlusion, motion parallax, and binocular disparity cues have strong influences on

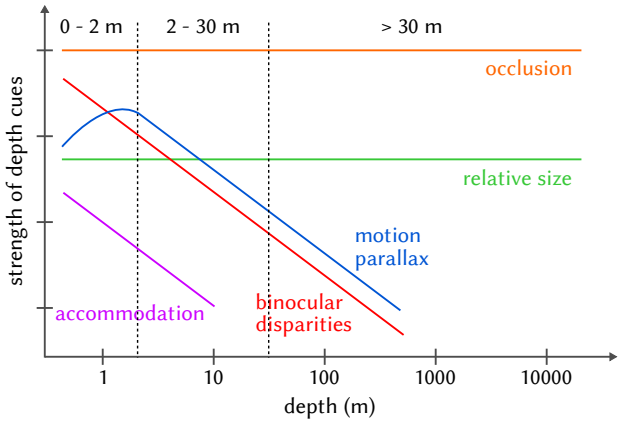


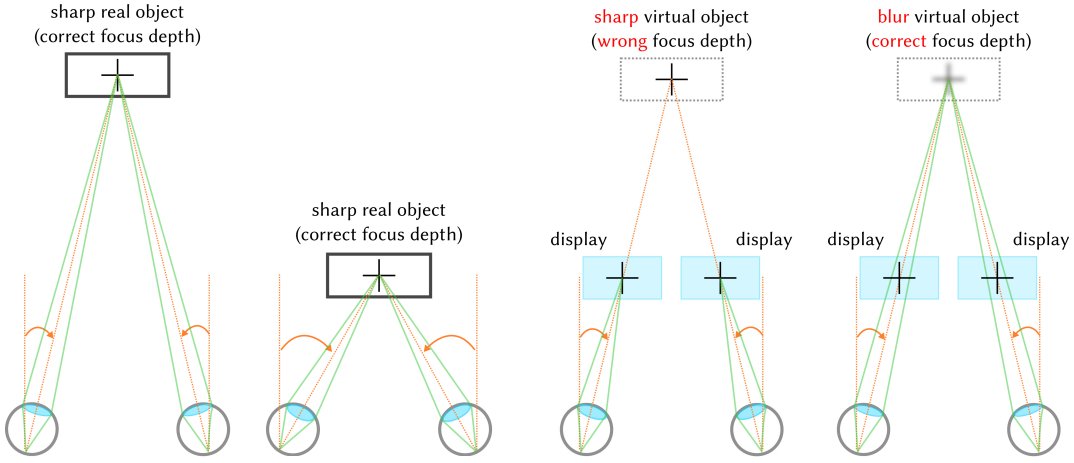
Figure 2.1: **Strength of depth perceptual cues.** A notional diagram of the relative strength of depth cues at different depth. Figure remade from [Cutting and Vishton, 1995, Geng, 2013].

our depth perception. Since these cues can be easily generated by showing 2D images to our eyes, most 3D displays primarily utilize these cues and ignore the accommodation cue, which is more challenging to generate. However, as we will see next, ignoring the accommodation cue creates a significant problem that often causes visual discomfort.

2.1.4 Vergence-Accommodation Conflict

In the real 3D world, all the depth cues are coherent with each other. For example, close objects produce large motion parallax, large eye vergence, and near eye accommodation. However, this is not the case in the virtual world created by typical 3D displays.

Typical 3D displays ignore the accommodation cue and drive our depth perception only with the binocular disparity, motion parallax, and the vergence cues, *etc.* However, the accommodation cue, *i.e.*, the focusing of our eyes, is the key to see sharp images, as shown in Figure 2.2. The inability to generate the accommodation cue either decouples our focusing to the virtual depth or create low-resolution images. In the first scenario, to see sharp images, our eyes focus on the display, where the images of the virtual object are shown, instead of the virtual depth indicated by other depth cues. This creates a problem called the vergence-accommodation conflict [Kooi and Toet, 2004], which is known to deteriorate our depth perception and cause eye fatigue, blurry vision, and other visual discomforts after a long period of use of the 3D display. In the second scenario, our brain drives the focus of our eyes to the correct depth of the virtual object. Even though the focused depth is correct, since the image of the



(a) In the real world, the vergence and the accommodation cues agree.

(b) In a virtual world created by a typical 3D display, the images look sharp only when the eye accommodation is wrong.

Figure 2.2: The vergence (angles indicated in orange color) and the accommodation (focusing of the light rays indicated in green color) in the real world and in a virtual world created by a typical 3D display that does not support the accommodation cue.

object is shown on the displays at another depth, we will see blurry images. In summary, if a 3D display does not generate accommodation cues, no matter where we focus, we will suffer from one of the two problems — visual discomfort or low resolution.

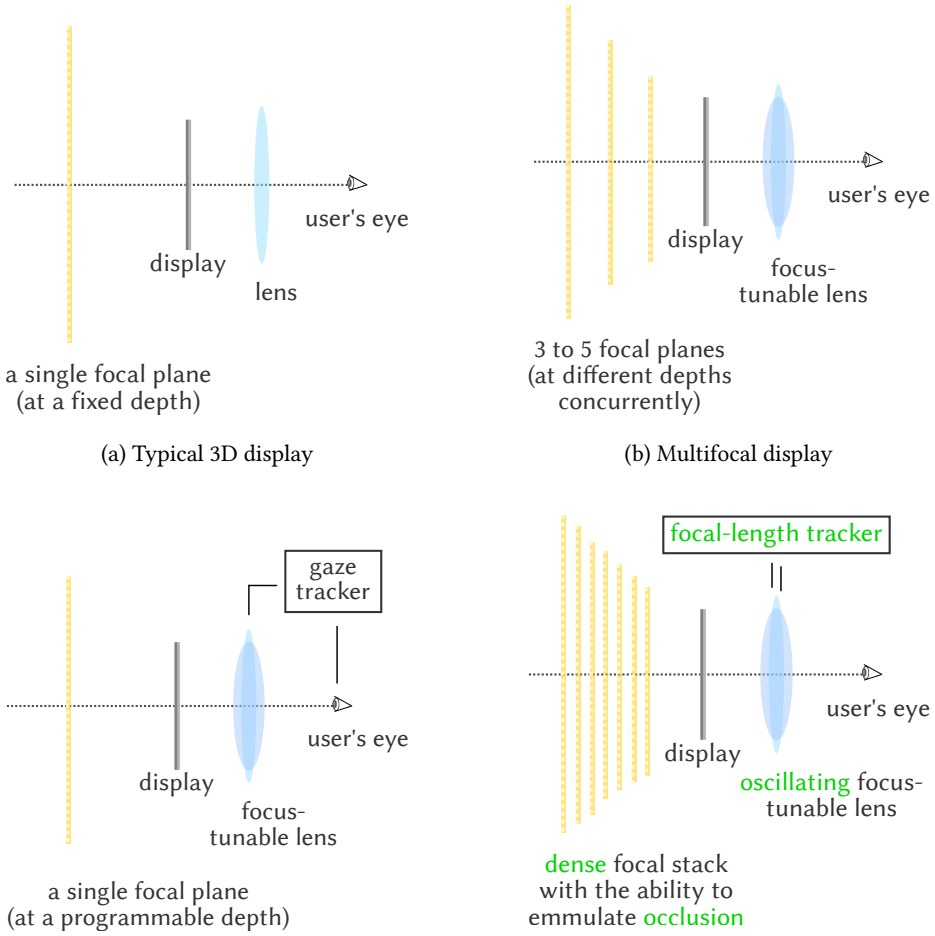
2.2 Overview of 3D Displays

Let us first understand how to build typical 3D displays, and we will see how existing accommodation-supporting displays work.

2.2.1 Typical 3D Displays

Typical 3D displays, like 3D televisions, movies, and AR/VR displays, convey depth information by showing two separate images to our two eyes. As we have discussed, the method can generate most of the depth cues but the accommodation cue [Cakmakci and Rolland, 2006].

The simplest way to show one image to one eye is to put a display in front of each of our eyes, as adopted by most AR/VR displays shown in Figure 2.3a [Wheatstone, 1838]. To reduce the form factor



(a) Typical 3D display

(b) Multifocal display

(c) Varifocal display

(d) Proposed multifocal display

Figure 2.3: **Various VR displays.** Most VR displays are composed of a lens and a display panel. The lens creates a virtual image of the display panel , or focal plane, for the user to look at. The figure illustrates the displays for one eye. For two eyes, the display is simply duplicated.

of the device, micro-displays like small OLED panels are put behind a lens in front of our eyes. The lens generates magnified images of the displays which locate inside the normal accommodation range of human eyes. For 3D televisions or movies, where only one display or screen is available, polarization [Kang *et al.*, 2010, Kim *et al.*, 2009] or color coding [Rollmann, 1853] can be used to separate the content on the screen to each eye. However, this requires viewers to wear specialized eyeglasses. To avoid any additional equipment, a lenticular array [Okoshi, 1980] or a parallax barrier [Son *et al.*, 2003] can be placed on top of the display to separate the pixels for each of our eyes.

By rendering images with all the psychological depth cues, these methods can drive vergence and create a sense of 3D efficiently. However, since the (magnified) displays are at a fixed distance, our eyes will tend to focus on the same depth regardless of the depth of the virtual objects. As a result, these displays cannot drive accommodation properly and suffer from the aforementioned vergence-accommodation conflict.

2.2.2 Displays that can drive Accommodation

The accommodation cue can only be generated when the light rays from a virtual object are oriented as if they originate from the depth of the object. In this section, we introduce accommodation-supporting 3D displays other than the multifocal displays that we have introduced in Section 1.2.1.

Light-Field Displays

Light-field displays, also known as integral displays, aim to control both the position and angle of the emitted light rays [Lanman and Luebke, 2013, Lippmann, 1908]. Each pixel on a light-field display is capable of sending light rays of different intensities toward different directions. This is in contrast to typical displays where each pixel sends the light of the same intensity toward all directions. With the angular control, light field displays are able to alleviate the vergence-accommodation conflict [Hiura *et al.*, 2017].

The goal of light field displays is to reproduce all the light rays that will enter our eyes. Imagine two invisible planes in front of your eyes, as shown in Figure 2.4. The two planes characterize every light ray that enters our eyes by their intersections. By placing a light field display on the first plane

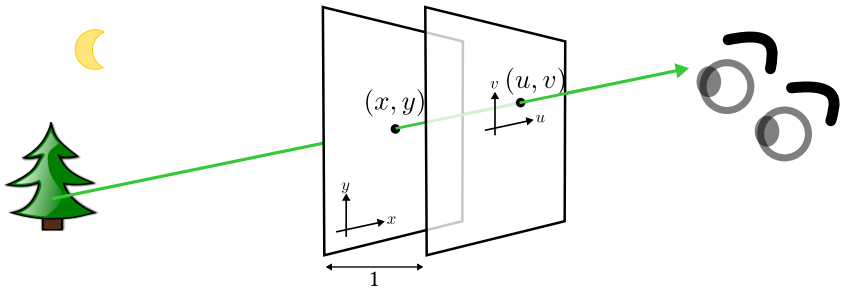


Figure 2.4: **Two-plane light field parameterization.** Each light ray can be characterized by its intersects on two frontal-parallel planes, which are separated by 1 unit. The origin of the second plane is relative to the intersect on the first plane. This makes (u, v) the tangent of the angles of the light ray.

and controlling the light that the display emits toward the second plane, we can reproduce the light rays that enter our eyes. If we can reproduce every light ray that enters our eyes, our eyes will think they are looking at a real scene, and all depth cues will be satisfied.

One method to build a light-field display is to put a microlens array on display panel [Lanman and Luebke, 2013, Lippmann, 1908]. The microlens array is placed one focal length away from the pixels, and thus the light emitted by the pixels is collimated and sent in different directions. For example, if three pixels are covered under a microlens along the horizontal direction, we divide the outgoing light rays into three groups, each group going in different directions. The number of pixels behind each microlens determines the angular resolution, whereas the size of a microlens determines the size of a spatial pixel. To create virtual objects across a wide depth range, we need high angular resolution, so we need large microlenses. However, this reduces the number of microlenses we can put on the display and lowers the spatial resolution. As a result, most light field displays have a low spatial resolution.

Instead of using a microlens array, tensor displays [Huang *et al.*, 2015, Maimone *et al.*, 2013, Wetzstein *et al.*, 2011] utilize multiple layers of transparent liquid crystal displays (LCD) to modulate the intensity of the light rays of different directions. Imagine the following setup where we have a transparent LCD in front of a typical display panel. When we switch on a pixel on the display panel, a region on the LCD will be lit. By setting the transmission rate of each LCD pixel in the region, we can modulate the intensity of each light ray of different directions. Putting multiple LCDs enables more complex modulation of the light rays. By exploiting the high correlation between the light rays from the same virtual object, these displays can achieve better spatial-angular tradeoff than those using a microlens array. However, using multiple layers of displays often deteriorates image quality due to the diffraction caused by the pixel grids. The size of the display pixels also limits their supported depth range.

Cossairt *et al.* [2007a] and Jones *et al.* [2007] produce light-field displays by coupling a projector with a rotating anisotropic diffuser. As the diffuser spins, the light emitted from the projector changes its direction. By synchronizing the projector and the rotation of the diffuser, the displays effectively increase their angular resolution in the horizontal direction without losing spatial resolution. However, the spinning diffuser makes the displays more geared towards 3D televisions and not VR.

Holographic Displays

Holographic displays also aim to reproduce the light entering our eyes. Unlike light field displays, which consider light as a collection of rays, holographic displays treat the light as electromagnetic *waves*. For any time instance, a monochromatic electromagnetic wave can be characterized by the magnitude and

the phase of the light at each location on a plane in front of our eyes. By reproducing these values for all wavelengths of light emitted by a real scene, a holography display can fully replicate reality.

Maimone *et al.* [2017] use a phase-only spatial light modulator (phase SLM) to build a holographic display. The phase SLM is a device that changes the phase of the incoming light without tampering its magnitude. Using a phase SLM, therefore, provides independent control of the phase of the light at each pixel location. To simultaneously control the magnitude and the phase, they first normalize the magnitude to be in the range of 0 and 1. Since any phasor whose magnitude is less than or equal to one can be decomposed into the sum of two phasors of magnitude one, by shining the phase SLM with uniform laser light and pairing adjacent phase SLM pixels, Maimone *et al.* can recreate the light from a scene composed by point light sources.

Even though holographic displays can reproduce the wavefront, their capabilities are often limited. First, since most holographic displays use phase SLM to control the phase, the authenticity of the generated wavefront is limited by the capability of the phase SLM like its resolution, the maximum amount of phase delay it can introduce, and its wavelength-dependency. The limitations significantly limit the resolution, field-of-view, and eye box of holographic displays. Second, the wavefront of a scene is computationally expensive to generate. In order to faithfully calculate the wavefront from a scene, we need to solve complicated Maxwell equations with boundary conditions. As a result, most computer-generated holograms are constructed using simplified algorithms and often does not reproduce occlusions faithfully. Finally, holographic displays requires coherent light, which can produce speckles that are often undesirable.

Varifocal Displays

Varifocal displays [Akşit *et al.*, 2017, Kim *et al.*, 2019, Konrad *et al.*, 2016, Liu and Hua, 2010, Padmanaban *et al.*, 2017, Sugihara and Miyasato, 1998] are very similar to multifocal displays that we introduced in Section 1.2.1. They both can be built with a translating display panel or a focus-tunable lens. While multifocal displays simultaneously show all focal planes with a frame, varifocal displays only has a single focal plane, but it dynamically adjusts its depth based on the user's gazing point. Figure 2.3 compares multifocal and varifocal displays.

The success of varifocal displays depends heavily on the capability of a gaze tracker. In an ideal varifocal display, the focal plane needs to follow the gaze of the user at all time. To achieve the goal, the gaze tracker needs to be very accurate (*e.g.*, with an angular resolution of less than 1 degree), which is very challenging to achieve in practice, and with very low latency, which is a hard system-design

problem.

There is an additional caveat when using varifocal displays. Since varifocal displays only have one focal plane, for any time instance, only a single depth can provide the correct accommodation cues. In other words, even though the accommodation is correct for the object at the gaze point, the accommodation cues for the objects in the peripheral area are wrong. To alleviate the effect, we need to render, in real-time, defocus blur in the peripheral area, which requires additional computations.

Other Types of Accommodation-supporting 3D Displays

Other types of 3D displays have been proposed to solve the vergence-accommodation conflict. Matsuda *et al.* [Matsuda *et al.*, 2017] use a phase-only spatial light modulator to create spatially-varying lensing based on the virtual content and the gaze of the user. Konrad *et al.* [Konrad *et al.*, 2017] operate a focus-tunable lens in an oscillatory mode. They use the focus-tunable lens to create a depth-invariant blur by using a concept proposed for extended depth of field imaging [Miau *et al.*, 2013]. Intuitively, since the content is displayed at all focal planes, the vergence-accommodation conflict is significantly resolved. However, there is a loss of spatial resolution due to the intentionally introduced defocus blur.

Summary of Accommodation-supporting 3D Displays

We summarize the advantages and limitations of the above solutions to drive accommodation in Table 2.1.

2.2.3 Depth-Filtering Methods for Multifocal Displays

When a multifocal display with sparse focal planes renders virtual scenes, it often causes aliasing artifacts as well as a reduction of spatial resolution on content that is to be rendered in-between focal planes. Akeley *et al.* [2004] show that such artifacts can be alleviated using linear depth filtering, a method that is known to be quite effective [MacKenzie *et al.*, 2010, Ravikumar *et al.*, 2011]. However, linear depth filtering produces artifacts near object boundaries due to the inability of multifocal displays to occlude light.

To produce proper occlusion cues with multifocal displays, Narain *et al.* [2015] propose a method that jointly optimizes the contents shown on all focal planes. By modeling the defocus blur of focal planes when an eye is focused at specific depths, they formulate a non-negative least-square problem that minimizes the mean-squared error between perceived images and target images at multiple depths. While this algorithm demonstrates promising results, the computational costs of the optimization are

Type of Display	Advantages	Limitations
Fixed focus	simple hardware	vergence-accommodation conflict, low resolution if not focused on the display
Light-field	supports all depth cues	low spatial resolution, shallow depth range
Holography	supports all depth cues	high computational cost, small field of view, speckles
Varifocal	high spatial resolution if focusing on the focal plane	latency, needs accurate gaze tracking and additional rendering of defocus blur
Multifocal	high spatial resolution if focusing on the focal planes	limited number of focal planes, poor occlusion cues

Table 2.1: Comparison of different types of 3D displays

often too high for real-time applications. Mercier *et al.* [2017] simplify the forward model of Narain *et al.* [2015] and significantly improve the speed to solve the optimization problem.

These filtering approaches assume perfect operating scenarios, *i.e.*, our pupils are at a known position with a fixed diameter, and our gaze direction can be accurately estimated. Most of the conditions are very challenging to achieve in practice. As we will see throughout the dissertation, introducing novel functionality through optics can increase the spatial resolution and produce the occlusion cue much more efficiently and effectively than merely rely on modifying the content.

3 Key Questions to Answer

Multifocal displays build a strong foundation to produce the accommodation cue. However, as we have discussed in Chapter 2, existing multifocal displays only have 3 to 5 focal planes and no occlusion cues. Thereby they cannot support a large depth range and suffer from loss of contrast. *Can we design a multifocal display that can produce occlusion cues and can show every object on its own focal plane?* Solving this problem requires solutions to a myriad of problems, both theoretical and practical, that we enumerate below. In the following chapters, I am going to answer each of the questions carefully, and at the end of the dissertation, we would have built a multifocal display that can generate natural accommodation cues, automatically create defocus blur, and support all perceptual depth cues used by the human visual system — all without the help of a gaze tracker or additional rendering.

Question 1: How Many Focal Planes Are Needed?

At this point, we may have an urge to design a multifocal display that can show thousands and thousands of focal planes, but we need to retain control and not hurry. Displaying focal planes can be costly, if not designed carefully. For a 3D display running at 60 frames per second (fps), for every additional focal plane that we want to show, we need to increase the refresh rate of the display panel by 60 fps. We may also need additional computational power to render the content on the focal planes. For our display to be efficient, we need to understand the benefit provided by each focal plane. For example, how does an extra focal plane change our capability to generate light fields? More importantly, since our eyes have finite depth resolution, is it possible that we need fewer focal planes than the number of virtual objects?

The question will be answered in Chapter 4.

Question 2: How Do We Display the Focal Planes in a High Frame Rate?

Displaying dense focal stacks requires a display with a very high frame rate. To show n focal planes per frame in F fps, the display panel needs to refresh at nF Hz. For example, a multifocal display with

10 focal planes per frame running at 60 fps already requires a refresh-rate of 600 Hz. Displays of such high frame rate are not widely available today; how do we achieve the capability efficiently and build a prototype ourselves?

The question will be answered in Chapter 5.

Question 3: How Do We Display Focal Planes Rapidly and Accurately in Depth?

The long settling time of the focus-tunable lenses hinders existing multifocal displays from creating dense focal stacks. The limitation lies in the physical nature of the tunable lenses. How do we overcome it? In addition, the capability to display many focal planes is meaningless if we cannot show each of them accurately in depth. Since focal planes are not real objects, how do we estimate their depths to display them accurately?

The question will be answered in Chapter 6.

Question 4: How Do We Generate the Occlusion Cue If Focal Planes Cannot Block Light?

Focal planes are virtual images of the display panel, and thereby, they cannot block light. Even when we remove the content in the occluded region of a far object, when the virtual object gets defocused, its blurred images can leak through objects on near focal planes, deteriorate the occlusion cue, and reduce the contrast of images. How do we create occlusion cues and increase the contrast of multifocal displays?

The question will be answered in Chapter 7.

4 Light Fields Generated by Multifocal Displays

A key factor underlying the design of multifocal displays is the number of focal planes required to support a target accommodation range. In order to be indistinguishable from the real world, a virtual world should enable human eyes to accommodate freely on arbitrary depths. In addition, the virtual world should have high spatial resolution anywhere within the target accommodation range. Simultaneously satisfying these two criteria for a large accommodation range is very challenging, since it requires generating light fields of high spatial and angular resolution. In this chapter, we are going to examine the light field generated by multifocal displays, and the conclusion of the analysis will lead us to designing the next-generation multifocal display.

4.1 Light-Field Parameterization and Assumptions

Light field is the collection of light rays. To perform analysis on light fields, we first need to characterize each light ray. A light ray can be described by its location in the space (x, y, z), its direction (θ, ϕ), its wavelength λ, and of course the time t we observe the light ray. Together these seven parameters define a function call plenoptic function:

$$P(x, y, z, \theta, \phi, \lambda, t) : \mathbb{R}^7 \to \mathbb{R}, \tag{4.1}$$

which gives the radiance of the light ray of interest. The plenoptic function provides every information of the light field.

For our purpose, to study the maximum space of light field a multifocal display can generate, we can ignore λ and t. Besides, we are only going to study the light field right before they enter our eye, so no object or light source will change the direction and the radiance of the light. As a result, one spatial parameter, say z, is redundant, because we can easily follow a light ray and calculate the values of the plenoptic function along the direction of the light ray. This leaves four parameters, (x, y, θ, ϕ), which describes the intersection of a light ray on a reference plane and its directions, respectively. We briefly

talked about how to parameterize light fields in Section 2.2.2, where we use tangent angles $u = \tan(\theta)$ and $v = \tan(\phi)$ to describe the angles. Here we are going to follow the same notation.

We are going to further simplify our scenario — we consider a flatland (*i.e.*, 2D world). The generalization to four-dimensional light fields can be conducted in a similar manner. In the flatland, the direction of a light ray is parameterized by its intercepts with two parallel axes, x and u, which are separated by 1 unit, and the origin of the u-axis is relative to each individual value of x such that u measures the tangent angle of a ray passing through x, as shown in Figure 4.1(a).

We model the human eye with a camera composed of a finite-aperture lens and a sensor plane d_e away from the lens, following the assumptions made in Mercier *et al.* [2017] and Sun *et al.* [2017]. We assume that the pupil of the eye is located at the center of the focus-tunable lens of the multifocal display and is smaller than the aperture of the tunable lens. We assume that the display and the sensor emits and receives light isotropically. In other words, each pixel on the display uniformly emits light rays toward every direction and vice versa for the sensor.

We further assume small-angle (paraxial) scenarios, since the distance d_o and the focal length of the tunable lens (or essentially, the depths of focal planes) are large compared to the diameter of the pupil. This assumption simplifies our analysis by allowing us to consider each pixel in isolation.

4.2 Light Field Generated by a Display

Let us decompose the optical path from the display to the retina (sensor) and examine the effect in frequency domain due to each component. The frequency-domain analysis reveals the maximum capability of multifocal displays, since it enables us to examine the bandwidth of the display.

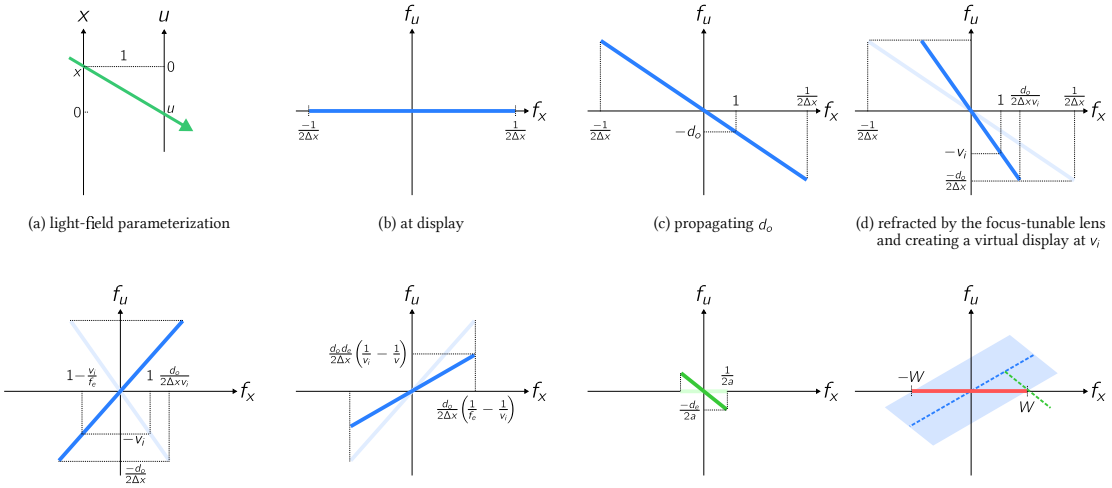
Due to the finite pixel pitch on the display panel, the light field creates by the display panel can be model as

$$\ell_d(x,u) = \left(\mathrm{rect}\left(\frac{x}{\Delta x}\right) * \ell_t(x, u=0)\right) \times \sum_{m=-\infty}^{\infty} \delta(x - m\Delta x),$$

where $*$ represents two-dimensional convolution, Δx is the pitch of the display pixel, and ℓ_t is the target light field. The Fourier transform of $\ell_d(x,u)$ is

$$L_d(f_x, f_u) = (\mathrm{sinc}(\Delta x f_x)\ \delta(f_u)\ L_t(f_x, f_u)) * \sum_{m=-\infty}^{\infty} \delta(f_x - \frac{m}{\Delta x}).$$

The finite pixel pitch acts as an anti-aliasing filter and thus we consider only the central spectrum replica ($m = 0$). Also, we assume $|L_t(f_x, f_u)| = 0$ for all $|f_x| \geq \frac{1}{2\Delta x}$ to avoid aliasing. Since the light field is



(a) light-field parameterization (b) at display (c) propagating d_o (d) refracted by the focus-tunable lens and creating a virtual display at v_i

(e) refracted by the lens in our eye, whose focal length is f_e (f) propagating d_e to the retina (g) aperture function propagated d_e to the retina (h) cross-correlation of (f) and (g) results in the light field at the retina (shaded blue). Our eye sees the slice along f_x.

Figure 4.1: **Light-field propagation from the display panel to the retina.** Fourier transform of the 2-dimensional light field at each stage of a multifocal display. The display is assumed to be isotropic and has pixels of pitch Δx. (a) Each light ray in the light field is characterized by its intercepts with two parallel axes, x and u, which are separated by 1 unit, and the origin of the u-axis is relative to each individual value of x. (b) With no angular resolution, the light field spectrum emitted by the display is a flat line on f_x. We focus only on the central part ($|f_x| \leq \frac{1}{2\Delta x}$). (c) The light field propagates d_o to the tunable lens, causing the spectrum to shear along f_u. (d) Refraction due to the lens corresponds to shearing along f_x, forming a line segment of slope $-v_i$, where v_i is the depth of the focal plane. (e,f) Refraction by the lens in our eye and propagation d_e to the retina without considering the finite aperture of the pupil. (g) The spectrum of the pupil function propagates d_e to the retina. (h) The light field spectrum on the retina with a finite aperture is the 2-dimensional cross-correlation between (f) and (g). According to Fourier slice theorem, the spectrum of the perceived image is the slice along f_x, shown as the red line. The diameter of the pupil and the slope of (f), which is determined by the focus of the eye and the virtual depth v_i, determine the spatial bandwidth, W, of the perceived image.

nonnegative, or $\ell_d \geq 0$, we have $|L_t(f_x, f_u)| \leq L_t(0,0)$. Therefore, we have

$$|L_d(f_x, f_u)| \leq L_t(0,0)\ |\mathrm{sinc}(\Delta x f_x)|\ \delta(f_u), \qquad |f_x| \leq \frac{1}{2\Delta x} \tag{4.2}$$

$$|L_d(f_x, f_u)| = 0, \qquad \text{otherwise.} \tag{4.3}$$

The inequalities provide a convenient upper-bound for the light field the display panel can generate,

regardless of individual pixel values shown on the display. Therefore, in the ensuing derivation, we will focus on the upper-bound

$$\widehat{L}_d = \operatorname{sinc}(\Delta x f_x)\, \delta(f_u)\, \operatorname{rect}\left(\frac{f_x}{\Delta x}\right).$$

The light field spectrum $\widehat{L}_d$ forms a line segment parallel to f_x, as plotted in Figure 4.1(b).

4.3 Propagation from Display to Retina

After leaving the display, the light field propagates d_o and get refracted by the focus-tunable lens before reaching the eye. Under first-order optics, there operations can be modeled by coordinate transformation of the light fields [Hecht, 2002]. Let $\mathbf{x} = [x\ u]^\top$. After propagating a distance d_o, the output light field is a reparameterization of the input light field and can be represented as

$$\ell_o(\mathbf{x}) = \ell_i\left(P_{d_o}^{-1}\mathbf{x}\right), \text{where } P_{d_o} = \begin{bmatrix} 1 & d_o \\ 0 & 1 \end{bmatrix}.$$

After refracted by a thin lens with focal length f, the output light field right after the lens is

$$\ell_o(\mathbf{x}) = \ell_i\left(R_f^{-1}\mathbf{x}\right), \text{where } R_f = \begin{bmatrix} 1 & 0 \\ \frac{-1}{f} & 1 \end{bmatrix}.$$

Since P_{d_o} and R_f are invertible, we can use the stretch theorem of d-dimensional Fourier transform to analyze their effect in the frequency domain. The general stretch theorem states that: Let $\mathbf{x} \in \mathbb{R}^d$, $\mathcal{F}(\cdot)$ be the Fourier transform operator, and $A \in \mathbb{R}^{d\times d}$ be any invertible matrix. We have

$$\mathcal{F}\left(\ell(A\mathbf{x})\right) = \frac{1}{|\det A|} L(A^{-\top}\mathbf{f}),$$

where L is the Fourier transform of ℓ, $\mathbf{f} \in \mathbb{R}^d$ is the variable in frequency domain, $\det A$ represents determinant of A, and $A^{-\top} = (A^\top)^{-1} = \left(A^{-1}\right)^\top$. By applying the stretch theorem to P_{d_o} and R_f, we can see that propagation and refraction shears the Fourier transform of the light field along f_u and f_x, respectively, as shown in Figure 4.1c-d.

4.4 Light Field Incident on the Retina

After reaching the eye, the light field ℓ_o is partially blocked by the pupil, refracted by the lens of the eye, propagates d_e to the retina, and finally integrated through all directions to form an image. The light field reaching the retina can be represented as

$$\ell_e(\mathbf{x}) = \ell_a\left(R_{f_e}^{-1}P_{d_e}^{-1}\mathbf{x}\right), \text{where } \ell_a(\mathbf{x}) = \operatorname{rect}\left(\frac{x}{a}\right)\ell_o(\mathbf{x}), \tag{4.4}$$

and a is the diameter of the pupil. The product in $\ell_a(\mathbf{x})$ is due to the blocking of the pupil on the light field, and it makes the light field difficult to analyze. In the following, we are going to examine the effect of the pupil closely and hopefully we can simplify Equation (4.4).

4.4.1 Effect of the Pupil

To understand the effect of the aperture of the eye, let us first analyze a more general situation where the light field is multiplied with a general function $h(\mathbf{x})$ and transformed by an invertible T with unit determinant. By multiplication theorem, we have

$$\ell_a(\mathbf{x}) = h(\mathbf{x}) \times \ell_o(\mathbf{x}) \overset{\mathcal{F}}{\longleftrightarrow} L_a(\mathbf{f}) = H(\mathbf{f}) * L_o(\mathbf{f}).$$

Thereby,

$$\begin{aligned} L_a(T\mathbf{f}) &= \int L_o(\mathbf{p}) H(\mathbf{p} - T\mathbf{f})\, \mathrm{d}\mathbf{p} = \int L_o(\mathbf{p}) H\left(T\left(T^{-1}\mathbf{p} - \mathbf{f}\right)\right) \mathrm{d}\mathbf{p} \\ &= \int L_o\left(T(\mathbf{q}+\mathbf{f})\right) H\left(T\mathbf{q}\right) \left|\frac{\partial \mathbf{p}}{\partial \mathbf{q}}\right| \mathrm{d}\mathbf{q} = L_o^{(T)} \otimes H^{(T)}(\mathbf{f}), \end{aligned} \tag{4.5}$$

where we use a change of variable by setting $\mathbf{q} = T^{-1}\mathbf{p} - \mathbf{f}$, and the last equation holds because $\left|\frac{\partial \mathbf{p}}{\partial \mathbf{q}}\right| = \det T = 1$. Equation (4.5) relates the effect of the aperture directly to the output light field at the retina: The spectrum of the output light field is the cross correlation between the transformed (refracted and propagated) input spectrum with full aperture and the transformed spectrum of the aperture function. The result is important since it significantly simplifies our analysis, and as a result, we are able to derive an analytical expression of spatial resolution and number of focal planes needed.

In our scenario, we have $T = \left(R_{f_e}^{-1} P_{d_e}^{-1}\right)^{-\top}$. For a virtual display at v_i, $\ell_o(\mathbf{x})$ is a line segment of slope $-v_i$ within $x \in [\frac{-1}{2\Delta x_i}, \frac{1}{2\Delta x_i}]$, where $\Delta x_i = \left|\frac{v_i}{d}\right| \Delta x$ is the magnified pixel pitch. According to Equation (4.5), $L_e(\mathbf{f}) = L_a(T\mathbf{f})$ is simply the cross correlation of $L_o(T\mathbf{f})$ and $\mathrm{sinc}(T\mathbf{f})$. After transformation, $L_a(T\mathbf{f})$ is a line segment of slope $\frac{d_e v_i - (d_e + v_i) f_e}{v_i - f_e}$, where $|x| \le \left|\left(\frac{v_i}{f_e} - 1\right) \frac{1}{\Delta x_i}\right|$. Similarly, $\mathrm{sinc}(T\mathbf{f})$ is a line segment with slope $-d_e$ within $|x| \le \frac{1}{2a}$. Note that we only consider $|x| \le \frac{1}{2a}$ because the cross-correlation result at the boundary has value $\mathrm{sinc}(0.5) \times \mathrm{sinc}(0.5) \approx 0.4$. Since $\mathrm{sinc}(x)$ function is monotonically decreasing for $|x| \le 1$, the half-maximum spectral bandwidth ($|L_e(\mathbf{f})| = 0.5$) must be within the region.

4.4.2 Focal Plane in Focus

Let the depth the eye is focusing at be v. We have $\frac{1}{v} + \frac{1}{d_e} = \frac{1}{f_e}$. When $v = v_i$, we can see from the above expression that $L_a(T\mathbf{f})$ is a flat segment within $|f_x| \le \frac{1}{2M\Delta x}$, where $M = \frac{d_e}{d_o}$ is the overall magnification

caused by the focus-tunable lens and the lens of the eye. From Fourier slice theorem, we know that the spectrum of the image is simply the slice $L_a(T\mathbf{f})$ along f_x.

In this case, the aperture has no effect to the final image, since the cross correlation does not extend or reduce the spectrum along f_x, and the final image has the highest spatial resolution $\frac{1}{2M\Delta x}$.

4.4.3 Defocused Focal Plane

Suppose the eye does not focus on the virtual display, or $v \neq v_i$. In the case of a full aperture ($a \to \infty$), the resulted image will be a constant DC term (completely blurred) because the slice along f_x is a delta function at $f_x = 0$. In the case of finite aperture diameter a, with a simple geometric derivation (see Figure 4.1h), we can show by simple geometry that the bandwidth of the f_x-slice of $L_e(\mathbf{f})$, or equivalently, the region $\{f_x | L_e(f_x, 0) \geq 0.5\}$, is bounded by $|f_x| \leq W$. And we have

$$W = \begin{cases} \frac{d_o}{2\Delta x d_e}, & \text{if} \left|\frac{1}{v_i} - \frac{1}{v}\right| \leq \frac{\Delta x}{a d_o} \\ \frac{1}{2a d_e}\left|\frac{1}{v} - \frac{1}{v_i}\right|^{-1}, & \text{otherwise.} \end{cases} \tag{4.6}$$

Thereby, based on Fourier slice theorem, the bandwidth of the retinal images is bounded by W.

4.5 Spatial Resolution of Retinal Images

We are now ready to characterize the spatial resolution of a multifocal display. Suppose the eye can accommodate freely on any depth v within a target accommodation range, $[v_a, v_b]$. Let $\mathcal{V} = \{v_1 = v_a, v_2, \ldots, v_n = v_b\}$ be the set of depth of the focal planes created by the multifocal display. When the eye focuses at v, the image formed on its retina has spatial resolution of

$$F_s(v) = \min\left\{\frac{d_o}{2d_e\Delta x}, \max_{v_i \in \mathcal{V}}\left(2ad_e\left|\frac{1}{v} - \frac{1}{v_i}\right|\right)^{-1}\right\}, \tag{4.7}$$

where the first term characterizes the inherent spatial resolution of the display unit, and the second term characterizes spatial resolution limited by accommodation, i.e. potential mismatch between the focus plane of the eye and the display. This bound on spatial resolution is a physical constraint caused by the finite display pixel pitch and the limiting aperture (*i.e.*, the pupil) — even if the retina had infinitely-high spatial sampling rate. *Any post-processing methods including linear depth filtering, optimization-based filtering, and nonlinear deconvolution cannot surpass this limitation.*

4.6 Minimum Number of Focal Planes Needed

As can be seen in (4.7), the maximum spacing between any two focal planes in diopter determines $\min_{v\in[v_a,v_b]} F_s(v)$, the lowest perceived spatial resolution within the accommodation range. If we desire a multifocal display with spatial resolution across the accommodation range to be at least F, $F \leq \frac{d_o}{2d_e\Delta x}$, the best we can do with n focal planes is to have a constant inter-focal separation in diopter. This results in an inequality that

$$\left(\frac{2ad_e}{2n}\left(\frac{1}{v_a}-\frac{1}{v_b}\right)\right)^{-1} \geq F, \tag{4.8}$$

or equivalently

$$n \geq ad_e\left(\frac{1}{v_a}-\frac{1}{v_b}\right)F. \tag{4.9}$$

Thereby, increasing the number of focal planes n (and distributing them uniformly in diopter) is required for multifocal displays to support higher spatial resolution and wider accommodation range.

4.6.1 Relationship to Prior Work.

There are many prior works studying the minimum focal-plane spacing of multifocal displays. Rolland *et al.* [1999] compute the depth-of-focus based on typical acuity of human eyes (30 cycles per degree) and pupil diameter (4 mm) and conclude that 28 focal planes equally spaced by $\frac{1}{7}$ diopter are required to accommodate from 25 cm to ∞. Both theirs and our analyses share the same underlying principle — maintaining the minimum resolution seen by the eye within the accommodation range, and thereby provide the same required focal planes. By taking $a = 4$ mm, $d_eF = 30 \times \frac{180}{\pi}$, $v_a = 25$ cm, and $v_b = \infty$, we have $n \geq 27.5$, which concurs with their result. MacKenzie *et al.* [2012, 2010] measure accommodation responses of human eyes during usage of multifocal displays with different plane-separation configurations under linear depth filtering [Akeley *et al.*, 2004]. Their results suggest that focal-plane separations as wide as 1 diopter can drive accommodation with insignificant deviation from the natural accommodation. However, it is also reported that smaller plane-separations provide more natural accommodation and higher retinal contrast — features that are desirable in any VR/AR display. By enabling dense focal stacks of focal-plane separation as small as 0.1 diopter, our prototype can simultaneously provide proper accommodation cues and display high-resolution images onto the retina.

4.7 Maximum Number of Focal Planes Needed

At the other extreme, if we have a sufficient number of focal planes, the limiting factor becomes the pixel pitch of the display unit. In this scenario, for a focal plane at virtual depth v_i, the retinal image of

an eye focuses on v will have maximal spatial resolution $\frac{d_o}{2d_e\Delta x}$ if

$$\left|\frac{1}{v} - \frac{1}{v_i}\right| \leq \frac{\Delta x}{ad_o}.$$

In other words, the depth-of-field of a focal plane — defined as the depth range that under focus provides the maximum resolution — is $\frac{2\Delta x}{ad_o}$ diopters. Since the maximum accommodation range of the multifocal display with a convex tunable lens is $\frac{1}{d_o}$ diopter, we need at least $\frac{a}{2\Delta x}$ focal planes to achieve the maximum spatial resolution of the multifocal display across the maximum supported depth range, or $\frac{D_o a d_o}{2\Delta x}$ focal planes for a depth range of D_o.

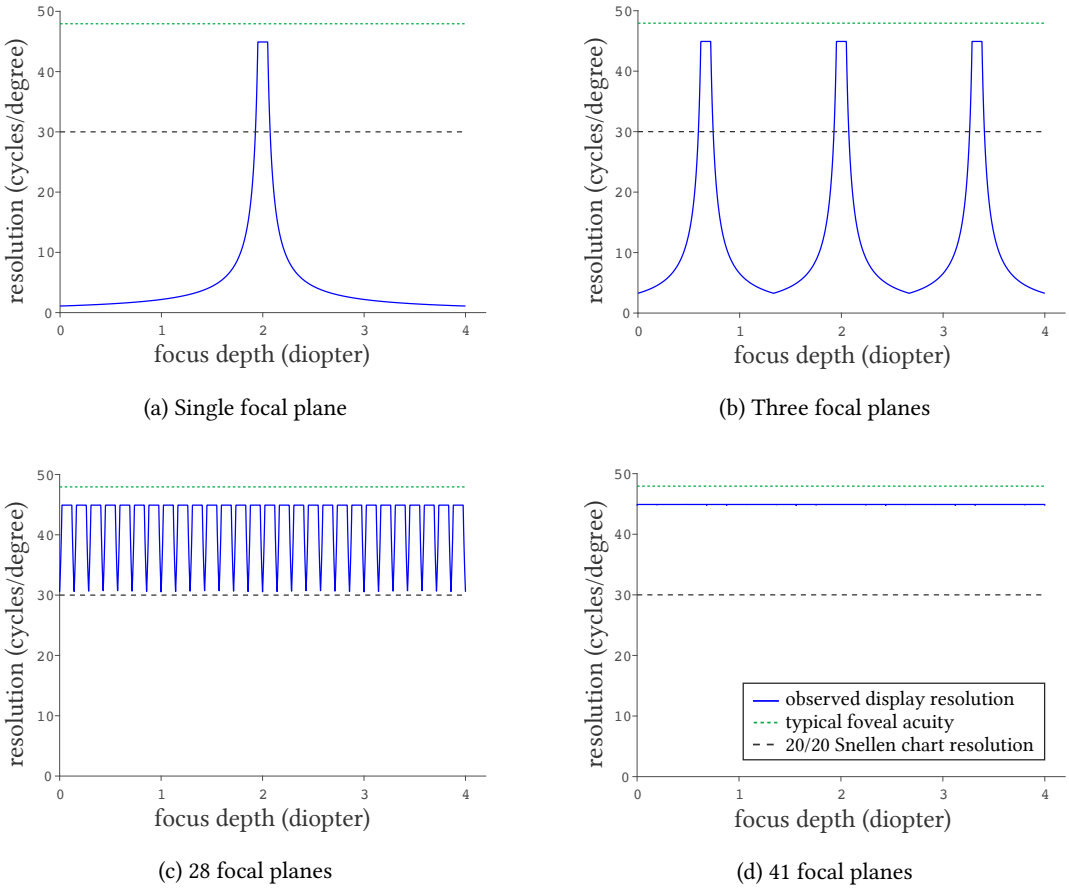
4.8 Conclusion of Our Analysis

Despite the lengthy analysis, our conclusion is very simple. The number of focal planes we need is simultaneously lower-bounded and upper-bounded. The lower bound is the smallest number of focal planes to achieve a certain resolution in the target depth range. The upper bound means that even if you have more focal planes than the number, you are not able to increase the resolution anywhere in the depth range. The upper bound is mainly due to the capability of the display. When we have a display with low resolution, we do not need many focal planes because the resolution is already bad. Nevertheless, if we have a nice display, we should increase the number of focal planes to enjoy the virtual world in high resolution.

Our analysis, following the derivation in Wetzstein *et al.* [2011] and Narain *et al.* [2015], characterizes the maximum capability of a multifocal display, regardless of the pixel values shown on each focal plane. The analysis is also similar to that of Sun *et al.* [2017] with the key difference that we focus on the minimum number of focal planes required to retain spatial resolution within an accommodation range, as opposed to efficient rendering of foveated light fields.

Our Target Number of Focal Planes

Our analysis gives an upper bound on the number of focal planes a multifocal display needs. Figure 4.2 shows the angular resolution ($F_s(v)d_e$) of our prototype multifocal display that we will introduce in Chapter 6. In the prototype, the pixel pitch $\Delta x = 13.6$ um, and $d_o = 7$ cm. Suppose that the pupil diameter is 4 mm. It requires 41 focal planes to achieve a consistent resolution of ~ 45 cycles per degree across a depth range of 4 diopters. Note that typical foveal acuity of health adults is around 48 cycles per degree, and the 20/20 Snellen eye chart has an angular resolution of 30 cycles per degree [Guenter *et al.*, 2012]. As a result, we set our goal on generating 40 focal planes per frame.



(a) Single focal plane

(b) Three focal planes

(c) 28 focal planes

(d) 41 focal planes

Figure 4.2: **Resolution of multifocal displays.** The figure shows the angular resolution of our prototype multifocal display if equipped with different numbers of focal planes.

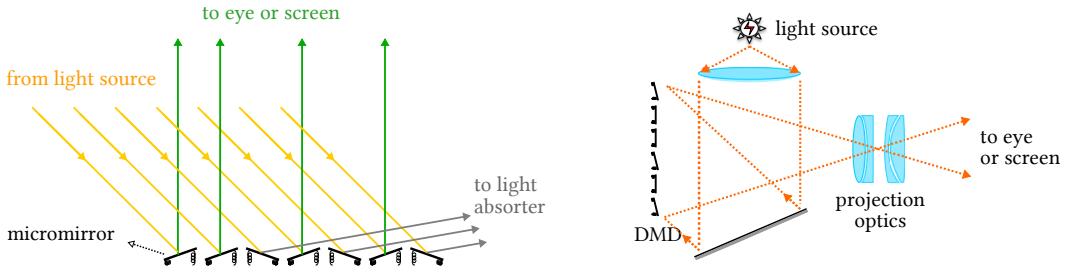
5 High-Speed Display with Light-intensity Modulation

Now that we have established a clear goal — display a dense focal stack in a multifocal display — how do we achieve this? In order to display dense focal stacks, we need a high-speed display and a high-speed focus-tunable lens. The speed of these components is critical. If the frame rate is too slow, our eyes will fail to fuse the content on the focal planes into a coherent scene, and we will see individual focal planes shifting in depth. In this chapter, we are going to focus on the high-speed display.

Let us first identify how fast the display needs to be. Suppose our multifocal display runs in F frames per second (fps), and we want to show n focal planes per frame. Since the content is different on different focal planes, the display panel in the multifocal display needs to refresh in nF Hz. From our discussion in Chapter 4, we have set our goal to show $n = 40$ focal planes per frame. If $F = 50$, we need a display with a refresh rate of 2000 Hz, which is far higher than what commodity displays can achieve. Most liquid crystal displays (LCD) or liquid crystal on silicon (LCoS) devices have refresh rates up to 240 Hz, due to the time to switch the state of the liquid crystals. While displays composed of organic light-emitting diodes (OLED) or micro-LEDs can in principle refresh at a very high speed, to the best of our knowledge, there is currently no such product available. As a result, we need to build a high-speed display ourselves with digital micromirror devices (DMD).

5.1 Digital Micromirror Devices

A digital micromirror device is composed of an array of micromirrors [Lee, 2018]. Each of the micromirrors can be programmed to flip in one of the two directions. As illustrated in Figure 5.1a, when flipped toward the light source, the micromirror directs light toward user's eyes or a screen and creates a bright pixel; otherwise, a light absorber will collect the light, and the pixel will be dark. Figure 5.1b illustrates the use of DMD in a projector. By installing projection optics in front of the DMD, we can form an image on a screen or user's retina and build a typical projector or a VR display, respectively.



(a) Working principle of DMD

(b) DMD projector

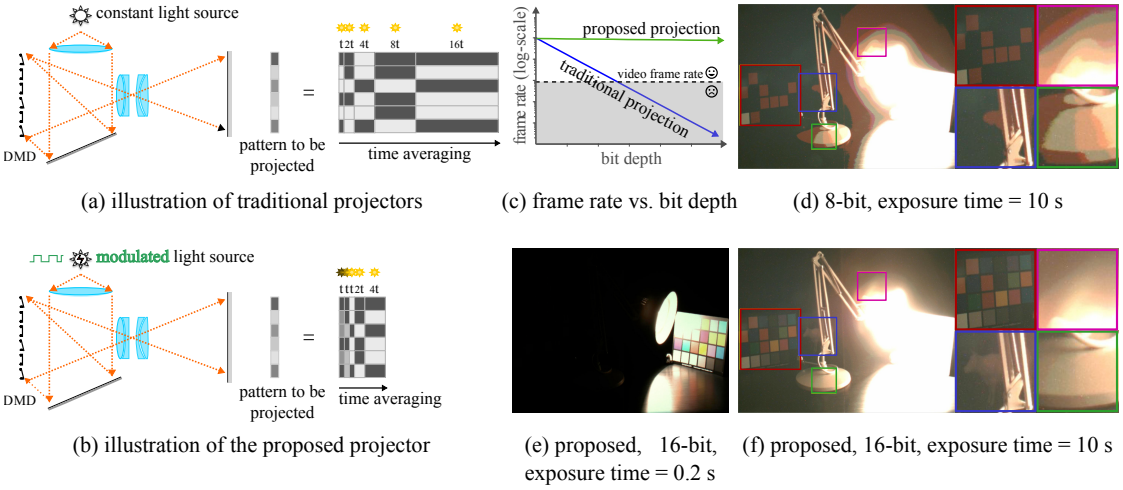
Figure 5.1: Digital micromirror device (DMD) and its use in projectors

The micromirrors in a DMD are very light-weight, so that they can be flipped rapidly, often more than thousands of times per second. This enables us to create a rapid sequence of *binary* images, or bitplanes, on the screen. In order to display gray-scale images in a DMD-based projector, these bitplanes are projected in rapid succession and subsequently averaged by the human eye due to the persistence of vision. The intensity observed at a pixel in the projected image is proportional to the number of bitplanes the pixel is illuminated in. The principle is called pulse-width modulation. The simplicity of pulse-width modulation has made DMD-based projectors the mainstream in the consumer projector industry [Markets and Markets, 2015].

5.2 Challenges for High-speed Projection

While pulse-width modulation enables DMD-based projectors to operate in typical frame rates, it fails to achieve high-speed projection. This is due to fundamental physical limitations to DMDs. One such limitation is the minimum time required to switch the micromirror array from one configuration to another.

To the best of our knowledge, due to the finite mass of the micromirrors, it takes at least 2 us for the fastest DMD to flip its micromirrors, To project a single RGB image at 8 bits per color channel, as we will see in the following section, we would require at least $(2^8-1)\times3\times2 = 1530$ us; this allows us to project in 653 fps, which is far from the 2000 fps that we need. Besides, if we instead wanted to display high bit-depth images, say 16 bits per color channel (16-bit), it would take a DMD-based projector 0.4 seconds to show a single image; this results in 2.4 fps. Clearly, there is a need for a design that can scale to the demands of modern applications.



(a) illustration of traditional projectors

(b) illustration of the proposed projector

(c) frame rate vs. bit depth

(d) 8-bit, exposure time = 10 s

(e) proposed, 16-bit, exposure time = 0.2 s

(f) proposed, 16-bit, exposure time = 10 s

Figure 5.2: **Overview of the proposed light-intensity modulated projection.** The proposed projector (b) uses an intensity-modulated light source instead of one with constant intensity in typical projectors (a). (c) The additional degree of freedom allows the proposed projector to achieve high bit-depth without losing frame rate. (e) and (f) are the photographs with two different exposure times of the same image projected by the proposed 16-bit projector; whereas (d) shows the results by a 8-bit projector. As can be seen, 16-bit projection successfully reproduces the image details like the checkerboard and the bull in the dark with fewer quantization errors. We denote the minimum DMD switching time by t.

What We Will Demonstrate in this Chapter

In this chapter, we propose a novel design for DMD-based projectors to achieve high bit-depth and high frame-rate projection. Our key innovation is in the form of light intensity control. Compared to traditional designs which use a light source with a constant intensity, our projector utilizes an intensity-modulated light source with a co-designed light-intensity coding scheme. This additional degree of freedom in the light source broadens the design space that enables us to increase both the frame rate and the bit-depth of the projector. We can also easily support the use of multiple color light sources to produce more vivid colors, wider color gamut, and higher brightness — all without sacrificing bit depth or frame rate. When we are interested in traditional levels of bit-depth projection, the proposed design reduces the operating speed of the DMD, which could potentially lower device costs. When incorporated into a VR display, the proposed projector not only can improve the immersion of the virtual world (by reducing the latency and producing more vivid colors) but also can enable us to build a multifocal display with dense focal stacks.

An overview of the proposed method is illustrated in Figure 5.2. As we will see in Figure 5.2(b), the proposed design requires a relatively small modification to existing projectors in the form of additional circuitry for enabling light intensity control. For LED sources, the intensity modulation can often be performed efficiently, without loss of energy, using pulse-width modulation. In all, this makes the proposed technology widely adoptable in most existing projector designs, including traditional 8-bit projectors and modern designs that enjoy high contrast ratio [Damberg *et al.*, 2016, Huang and Pan, 2014, Pan and Wang, 2013]. It also makes the proposed technology easily adoptable in our multifocal VR display.

Specifically, we make the following contributions.

- *Light intensity control for DMD-based projector.* We propose a novel approach to increasing the bit-depth and the frame rate of a projector by introducing intensity coding at light source. This light intensity coding is easily achieved using pulse-width modulation with little loss of energy and introduces little additional complexity to traditional projector designs.
- *Code design.* We chart out a design space that allows us to tradeoff the maximum brightness of the projected scene towards obtaining higher bit-depth, frame rate and/or wider color gamut. This design space is enabled by a novel hybrid code design that mixes light intensity control with traditional pulse-width modulation. A key observation is that for marginal loss in brightness (often, less than 4%) we can enable higher bit-depth and color gamut.
- *Hardware prototype.* We present a hardware prototype to showcase and validate the performance of our projector.

Note that the benefit of high bit-depth projection is often only perceptible by the human visual system when the projector has high contrast ratio and in a dark environment. The enclosure provided by a VR headset effectively isolate any environment light source. While the proposed design does not increase the contrast ratio of a projector, it can easily be incorporated into existing methods that increase contrast ratio using novel prism designs [Huang and Pan, 2014, Pan and Wang, 2013] or using a light modulator to spatially reallocate light [Damberg *et al.*, 2016].

5.3 Background

We start this section by introducing the concept of bit-depth and the dynamic range of a projector. Then we will dive into the details of DMD-based projection and pulse-width modulation. In the end, we will introduce prior works on light-intensity modulated projection.

5.3.1 Bit Depth and Dynamic Range

The output of a projector can be described by two factors — dynamic range and bit depth. Dynamic range determines the range of the projected light intensity and is measured by the contrast ratio, which is the ratio between the highest possible light intensity and the lowest intensity, *e.g.*, $C:1$. Bit depth, on the other hand, describes the intensity resolution and is usually represented in bits. For example, an 8-bit color channel means that its intensity range is uniformly divided into $2^8 = 256$ levels.

For a projector with a contrast ratio of C and n-bit intensity resolution, if we denote the intensity of j-th level by I_j, where $j \in [0, 2^n - 1]$, we can represent the ratio between the j-th intensity level I_j and the highest intensity $I_{\max}$ (or I_{2^n-1}) as

$$\frac{I_j}{I_{\max}} = \frac{j}{2^n-1} + \frac{1}{C} \times \frac{2^n-1-j}{2^n-1}, \quad \forall j \in [0, 2^n-1], \tag{5.1}$$

where the first term represents the ratio of intended intensity outputs and the second term represents the undesired outputs due to limited contrast ratio. Therefore, the intensity difference between two adjacent levels is

$$I_{j+1} - I_j = \frac{1}{2^n-1}\left(1 - \frac{1}{C}\right) I_{\max} := \frac{1}{2^n-1} I_{\text{eq}}, \ \forall j \in [0, 2^n-2], \tag{5.2}$$

where $I_{\text{eq}} = (1-1/C)I_{\max}$ is the equivalent intensity range. It can be seen that larger C allows us to utilize larger portion of $I_{\max}$, and increasing n allows subtle intensity differences (*i.e.*, image details) to be reproduced.

Contrast ratio of projectors is usually determined by their optical designs. For example, novel prism designs reduce stray light in the projectors and boost contrast ratio to $2^{15.5}:1$ [Huang and Pan, 2014, Pan and Wang, 2013]; light reallocation methods [Damberg *et al.*, 2016, Damberg and Heidrich, 2015, Hoskinson and Stoeber, 2008, Hoskinson *et al.*, 2010] redistribute light energies from darker regions in the images to the brighter ones and thereby directly increase the contrast ratio of the projected image. As these projector designs have achieved contrast ratios higher than $2^{15}:1$, in this paper, we propose a high bit-depth projection technique that can easily achieve bit depth of 16 bits, which fully utilizes the high dynamic-range of modern projector designs.

5.3.2 DMD-based Projection

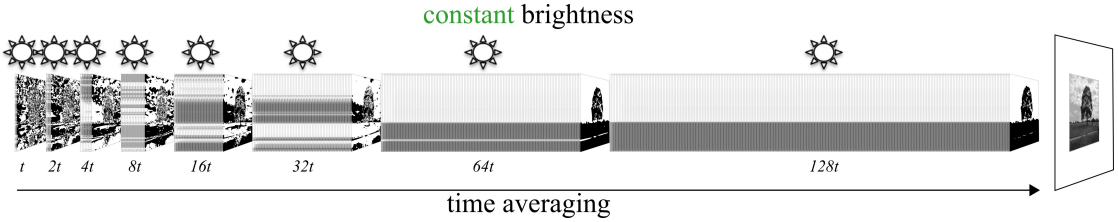
A DMD-based projector is typically composed of a light source, a DMD, and a projection lens, as shown in Figure 5.1b. The light source constantly shines light onto the micromirrors on the DMD, and the projection lens maps each micromirror to a pixel on the screen. Each micromirror has two states — when turned on, it directs light toward its corresponding pixel; when turned off, it directs light away

from the projection lens (usually toward a light collector in the projector.) Although the on-off operation allows simple micro-electromechanical design, the binary characteristic can only generate black-and-white images at any instance.

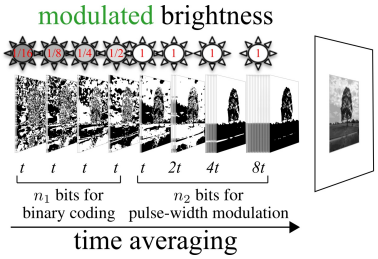
There are a few constraints underlying the operation of a DMD. There is a minimum amount of time required to transition from one micromirror configuration to another — a limitation that is imposed due to data transmission to the DMD. Typically, data transmission bus operates at 64 bits and 400 MHz. Hence, for a DMD with 1024×768 micromirrors, it takes $\frac{1024\times768}{64\times400} = 30.72$ us to transmit a full-frame binary image. The data transmission rates can be even larger for higher resolution DMDs. One approach to reducing the transmission time is to group micromirrors in blocks that are synchronized and change states jointly. This approach sacrifices spatial homogeneity within individual binary image similar to the trade-off between global and rolling shutters in cameras. Further, the electronics associated with block-based control is significantly complicated, and this makes the device expensive. However, the gains provided by this approach are often significant and it can reduce the latency between bitplanes to as little as 2 us for a 1024×768 array. For simplicity of analysis, we adopt an ideal DMD model that sends out bitplanes with a minimum exposure time of 2 us.

5.3.3 Pulse-width Modulated Projection

DMD projectors use pulse-width modulation (PWM) at each pixel (micromirror) to project a frame. Each micromirror encodes the binary representation of the desired intensity value at the corresponding pixel. Given a chosen minimum bitplane exposure time t ($t \geq 2$ us in our example), an n-bit grayscale image is produced by sequentially projecting n bitplanes (from the least significant bit (LSB) to the most significant bit), with exposure time $t, 2t, \ldots, 2^{n-1}t$ us. Hence, the total time required to project one image is $(2^n-1)t$ us. This concept is illustrated in Figure 5.3a and the analytical formulas for frame rate and contrast ratio are listed in Figure 5.1. One simple method to project color images is repeating this process for each color channel and, hence, for a three-color image, the total time to project a single image is $3(2^n-1)t$ us. The exponential dependence on intensity resolution, given in terms of the number of bits n, reflects the lack of scalability of traditional designs, as demonstrated by some examples in Figure 5.2. It is worth mentioning that more sophisticated color projection can be achieved by the technique called gamut reshaping [Majumder *et al.*, 2010], which fully utilizes the wider color gamut of modern light sources like LEDs and lasers to increase color fidelity and a maximum brightness of a projector. Since the gamut reshaping is achieved by temporally modulating the on-duration of the light sources, to increase the bit depth, the total frame exposure time still needs to be increased exponentially in term of n.



(a) An illustration for projecting a 8-bit, grayscale image with PWM projection.



(b) An illustration for projecting a 8-bit, grayscale image with the proposed HLM projection.

Figure 5.3: **Comparison between the PWM projection and proposed hybrid light modulated projection.** By incorporating light intensity control, the proposed method greatly reduces frame time.

5.3.4 Light-intensity Modulated Projection

Intensity-modulated light sources can be used to break the exponential relationship between total frame exposure time and intensity resolution. One approach of light-modulated projection is to perform binary-coding on the light source *intensity* [Hainich and Bimber, 2014]. Suppose that we seek to project an n-bit grayscale image. We project n bitplanes, each of a fixed exposure period t. Each bitplane is associated with a different light source intensity; specifically, the i-th bitplane is illuminated with the light intensity set at $2^{-(n-i)}$ times its maximum intensity. Hence, the light source intensity takes the values $2^{-(n-1)}, 2^{-(n-2)}, \ldots, 2^{-1}, 1$. Each micromirror is coded with the n-bit binary representation of the intensity that we seek to project at its corresponding pixel. We refer to this scheme as binary light modulation (BLM).

A key feature of BLM is that it can project an n-bit image using n bitplanes with the same exposure time and hence, the total time to project an image is simply nt us. This linear dependence of exposure time on n is in sharp contrast to traditional PWM projection whose exposure time is exponential in n, as compared in Figure 5.1 and Figure 5.2. Hence, we can achieve very high frame rate as well as

	pulse-width modulation (PWM)	**binary light modulation (BLM)**	**hybrid light modulation (HLM)**
frame rate	$\frac{f}{2^n-1}$	$\frac{f}{n}$	$\frac{f}{n_1+2^{n_2}-1}$
rel. brightness	1	$\frac{2}{n}\left(1-2^{-n}\right)$	$\frac{1-2^{-n}}{1+(n_1-1)\times 2^{-n_2}}$
contrast ratio	$C:1$	$C:1$	$C:1$
power efficiency (avg. power to project the LSB)	$\frac{p}{2^n-1}$	$\frac{p}{2^n-1}$	$\frac{p}{2^n-1}$

Table 5.1: **Expressions of frame rate, relative brightness, contrast ratio, and power efficiency of n-bit grayscale projection.** We denote the minimum bitplane switching time by $\frac{1}{f}$ and the power to project a frame with maximum brightness by p.

	PWM		**BLM**		**HLM**	
bit depth (bits)	frame rate	rel. brightness	frame rate	rel. brightness	frame rate	rel. brightness
12	40.7 fps	1	555 fps	0.17	50.5 fps	0.97
14	10 fps	1	476 fps	0.14	49.8 fps	0.96
16	2.5 fps	1	416 fps	0.12	49.0 fps	0.94

Table 5.2: **Examples of frame rate and relative brightness of RGB projection.** For PWM, we set $t = 2$ us, which is achieved with the help of block-based micromirror control. For BLM and HLM, we set $t = 50$ us as that used in our prototype (without the help of block-based micromirror control). We assign $n_2 = 7$ for HLM.

intensity resolution in BLM. However, BLM has one critical disadvantage — a significant reduction in the brightness of the projected image. The maximum brightness of BLM can be derived by

$$L^b_{\max} = \frac{1}{n}\left(1+\frac{1}{2}+\cdots+\frac{1}{2^{n-1}}\right)L = \frac{2}{n}\left(1-\frac{1}{2^n}\right)L \approx \frac{2}{n}L, \tag{5.3}$$

where n is the number of bitplanes and L is the full intensity of the light source. For example, to achieve a bit depth of 16-bits, BLM can only output 1/8 of the brightness achievable by PWM. This makes BLM rather impractical for most applications — a point noted in [Hainich and Bimber, 2014] as well.

5.4 Prior High Bit-depth Projection Techniques

Existing high bit-depth projectors utilize multiple spatial light modulators (SLM), *e.g.*, DMDs or liquid crystal displays (LCD), either in parallel or in series. Cinema projectors utilize three DMDs in parallel with each DMD associated with a single color channel, to achieve 15-bit projection at video rate [Texas Instruments, [n.d.]a]. Dual modulation techniques [Damberg *et al.*, 2007, Heide *et al.*, 2014, Kusakabe *et al.*, 2009, Pavlovych and Stuerzlinger, 2005, Seetzen *et al.*, 2004, Wetzstein *et al.*, 2011] are popularly used in high dynamic-range projectors, and they utilize two (or more) SLMs in series to modulate the outgoing light multiple times. For example, an LCD placed in front of a DMD-based projector can provide additional attenuation of the outgoing light intensity and thus reduces minimum brightness and increases dynamic range. SLMs like analog micromirror array or liquid crystal on silicon (LCoS) can also be used to redistribute light energy from dark pixels to bright ones to increase both energy efficiency and dynamic range of the projectors [Damberg *et al.*, 2016, Damberg and Heidrich, 2015, Hoskinson and Stoeber, 2008, Hoskinson *et al.*, 2010].

In addition to the increased device costs, utilizing serial SLMs leads to the following three challenges. First, every stage of light modulation loses energy. For example, light efficiency for DMDs is 68% [Texas Instruments, [n.d.]c,n] and those for LCDs and LCoSs are at most 50% (due to the polarization). To compensate for the lost light energy, higher-powered light sources are needed to achieve desired brightness. Second, despite the increased dynamic range, serial modulations usually produce nonuniform intensity levels and thus require additional preprocessing algorithms from the typical linear intensity values. In addition, even with the preprocessing, the non-uniformity in pixel intensity reduces the overall bit depth of the projectors. For example, serial modulated projectors composed of two 8-bit LCDs can achieve at best intensity resolution of 13.3 bits. Third, utilizing multiple SLMs requires sophisticated optical designs, including subtle calibrations like careful positioning the SLMs. Due to these factors, existing high bit-depth/high dynamic-range projectors are usually much more expensive than standard projectors.

Compared to these methods, our proposed projector requires only a single DMD and generates uniform intensity levels, and thereby no additional costs or calibration are needed. Besides, the system can be easily modified from commercial 8-bit DMD projectors or incorporated into existing high dynamic-range projectors or VR displays.

5.4.1 Other Types of High Bit-depth Displays and Projectors

The principle of dual modulation has also been utilized in a variety of high dynamic-range displays [Ferwerda and Luka, 2009, Guarnieri *et al.*, 2008, Seetzen *et al.*, 2004, Wanat *et al.*, 2012]. By coupling a second

SLM with a traditional projector or a LED panel, the local light intensity can be individually controlled to achieve high contrast ratio. Auto-iris technique dynamically controls the intensity of the light source based on the image content. For example, when showing a dark scene, the iris is reduced to increase the bit depth at the lower-intensity range. Therefore, the method cannot improve bit depth effectively if a scene contains both bright and dark regions. In comparison, the proposed method provides high bit-depth projection in every frame and requires no adaptation to the projected content. Multi-projector systems [Damera-Venkata and Chang, 2009, Majumder and Brown, 2007, Majumder and Welch, 2001] overlap the projected images to increase the maximum brightness as well as the spatial resolution; however, this requires going beyond a single light modulator and hence, has the same benefits and limitations as multi-DMD systems.

5.5 Hybrid Light Modulation

Recall that while traditional PWM projection has the maximum brightness output, it suffers from low frame rate due to the exponential dependence between the exposure time of an image and the desired bit depth. As BLM projection significantly increases the frame rate, it sacrifices maximum brightness and therefore is not useful in practice. In this section, we propose *hybrid light modulation*, which solves the disadvantages of PWM and BLM by carefully incorporating PWM with BLM.

Observe that with PWM projection, the bitplane exposure time grows exponentially as the number of bits increase, while with BLM, the bitplane exposure time remains identical. If we apply BLM to the least significant bitplanes to decrease the number of bits assigned to PWM (as in Figure 5.3b), we can dramatically reduce the total exposure time to project an image. Besides, if only a few bitplanes are assigned to BLM, the total exposure time of PWM will still be exponentially longer than that of BLM and thereby mitigate the loss of the maximum brightness due to BLM. We refer to this scheme as hybrid light modulation (HLM). In essence, in HLM, we use BLM for only the lesser significant bitplanes and use PWM on the rest (*i.e.*, more significant) bitplanes. As we will analyze next, HLM can provide a significant reduction in exposure time associated with a frame without a commensurate reduction in brightness.

Suppose that we require an intensity resolution of n bits. We breakup the intensity resolution into two buckets: n_1 bits that are assigned to BLM and n_2 bits assigned to PWM, with $n_1 + n_2 = n$. Further, the PWM is performed with the light source intensity set to its maximum. The total exposure time per image is

$$(n_1+2^{n_2}-1)t,$$

and its maximum achievable brightness is

$$L^h_{\max} = \frac{1}{n_1+2^{n_2}-1}\left(2^{n_2-1}+\ldots+1+\frac{1}{2}+\ldots+\frac{1}{2^{n_1}}\right)L = \frac{1-2^{-n}}{1+(n_1-1)\times 2^{-n_2}}L. \tag{5.4}$$

We can expect that 2^{n_2} is usually much larger than n_1, and hence, the maximum brightness $L^h_{\max} \approx L$ for small n_1 and the frame time is approximately $2^{n_2}t$. Hence, by optimizing over n_1 and n_2 while keeping their sum equal to n, we can achieve the desired tradeoff between max-brightness and speed of operation, in terms of images per second. Besides, the contrast ratio of HLM can be computed as

$$C^h = \frac{L^h_{\max}}{L^h_{\min}} = \frac{\left(2^{n_2-1}+\ldots+1+\frac{1}{2}+\ldots+\frac{1}{2^{n_1}}\right)L}{\left(2^{n_2-1}+\ldots+1+\frac{1}{2}+\ldots+\frac{1}{2^{n_1}}\right)\frac{L}{C}} = C, \tag{5.5}$$

which remains the same as the contrast ratio of the original PWM projection. We compare the frame rate, brightness, contrast ratio, and power efficiency between PWM, BLM, and the proposed HLM in Figure 5.1 and provide some example numbers in Figure 5.2. For the numbers in Figure 5.2, we allow for block-based micromirror control only for the PWM scheme. Yet, even without block-based control, HLM coding achieves 16-bit, RGB projection over 49 fps with only 6% of brightness reduction, when $n_2 = 7$. In contrast, traditional PWM with block-based control can only achieve 2.5 fps in spite of the efficiencies enabled by block-based control.

We illustrate the trade-offs among the bit depth, the frame rate, and the maximum brightness of PWM, BLM, and the proposed HLM in Figure 5.4. Note that when assigning all bitplanes to PWM (bottom right corner), HLM reduces to the traditional PWM, which has high brightness but low frame rate; on the other hand, when all bitplanes are assigned to use BLM, HLM becomes BLM and has high frame rate but low brightness. By designing the coding scheme (*i.e.*, judicious selection of n_1), HLM can achieve both high brightness and high frame rate for high bit-depth projection (upper-right corner).

5.5.1 Enhancing Color Gamut and Brightness

Recall that single DMD projectors use temporal dithering to achieve color perception and hence, the frame rate of projection is reduced by a factor equal to the number of colored light sources. For a small loss of brightness, we can exploit the efficiencies enabled by HLM to achieve wider color gamut without a commensurate loss in bit depth or frame rate. For example, Figure 5.5b shows the typical color gamut of projectors with RGB LEDs; by adding cyan and yellow LEDs, we can potentially expand the color gamut to be a pentagon, and thereby project deeper colors that are unable to be produced by ordinary RGB projectors. Similarly, if we add a bright-white LED, we can boost the brightness on the brighter spots in the images and make them more vivid; here, we rely on the higher luminance

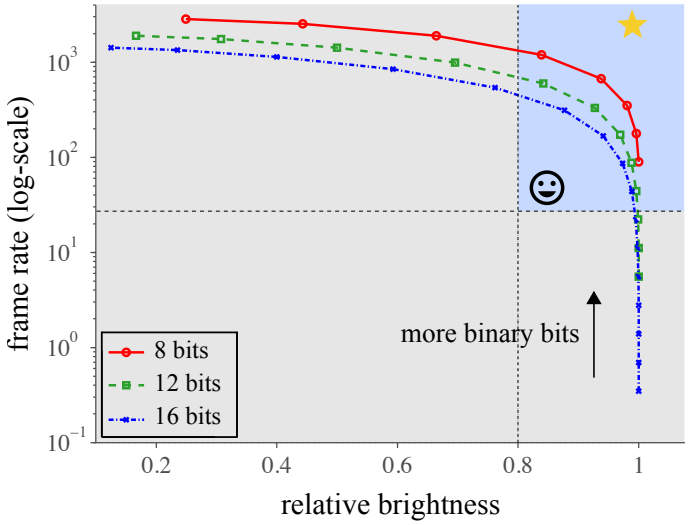


Figure 5.4: **Trade-off between brightness and frame rate of our proposed HLM with bit depth $n = 8, 12,$ and 16 bits and $t = 50$ us.** The bottom-right corner assigns all the bits to PWM ($n_1 = 0$), and by increasing n_1, the frame rate increases rapidly without losing much of the brightness. When all bits are assigned to BLM (upper-left corner), the projection has the highest frame rate but lowest brightness. It can be seen that our HLM effectively achieves high frame rate and high brightness with high bit-depth (upper-right corner).

output typical to commercial white LEDs. The proposed HLM can also be incorporated into the gamut reshaping technique [Majumder *et al.*, 2010] to concurrently increase bit depth and utilize the increased color gamut to increase color fidelity and maximum brightness.

5.6 Prototype and Experimental Results

We build a prototype to examine the benefits of the proposed light intensity modulation. We focus on demonstrating the capability of projecting high bit-depth images in the chapter, and the capability to perform high frame-rate projection will be demonstrated when we display dense focal stacks in the following chapters.

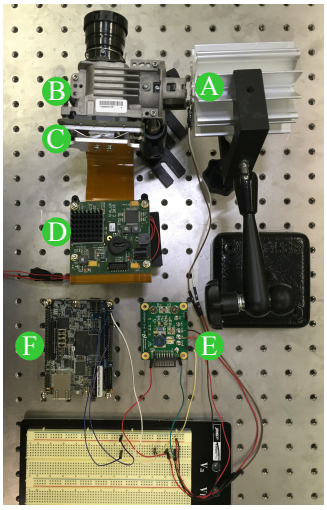
5.6.1 Light Intensity Control

In order to precisely control the intensity of the light source, we can use LEDs or laser diodes as light sources, both of which allow analog and digital controls of intensity at high frequencies. LEDs and

laser diodes are also able to achieve high luminous flux output, *e.g.*, above 4000 lm for LEDs and 10000 lm for laser diodes, and have been popularly used in commercial projectors due to their high energy efficiency. While their current-driven characteristic allows us to control intensity analogly, we choose to modulate their intensity digitally by switching the LEDs on and off rapidly, *e.g.*, in MHz, with pulse-width modulation. By controlling the 'on' duration within the exposure time of each bitplane, we can adjust the (averaged) light intensity. While more efficient electrical designs may be used to switch the LEDs in high frequency, for simplicity we use a current controller [Texas Instruments, [n.d.]e] that requires no capacitor connected to the LEDs (thus minimizes delays), and we use a MOSFET to shunt the current from the LED inputs to turn off the LEDs. Since the MOSFET can be turned on and off within a few nanoseconds, we can easily switch the LEDs in 20 MHz in our system prototype. This allows us to assign 10 bits to BLM with minimum bitplane exposure equal to 50 us.

5.6.2 System Prototype

Our system prototype is composed of three components — the projection optics, the DMD development kit, and the LED with associated circuits for intensity modulation (see Figure 5.5a). For the projection



A **LED**
Led-engin LZP-04MD00

B **Projection optics**
Vialux star core optics

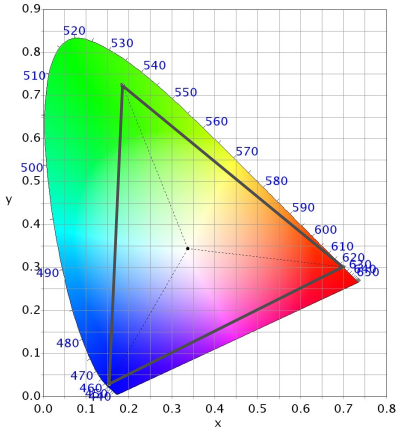
C **Digital micromirror device**
TI DLP7000

D **DMD control**
Vialux V-7000

E **LED driver**
TI LM3409HV

F **FPGA control**
Terasic DE0 nano soc

(a) System prototype.



(b) Color gamut.

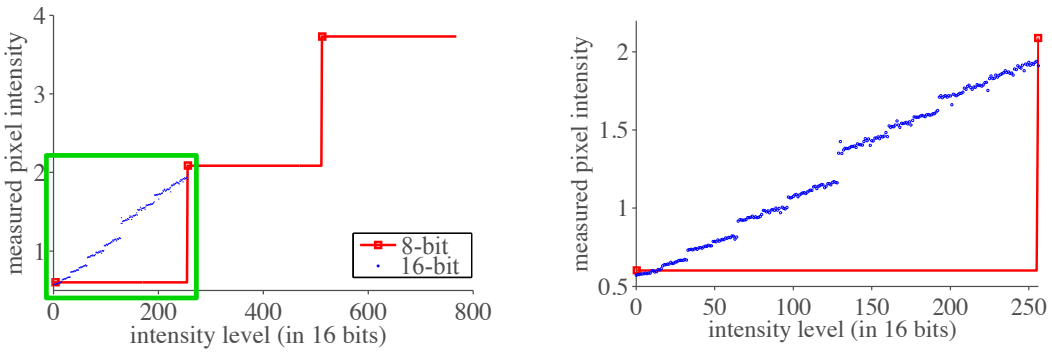
Figure 5.5: **Prototype of the proposed projector.** The prototype is composed of a LED, a DMD, and a projection optics. The color gamut of the prototype can be expanded by adding yellow or cyan LEDs to our system.

optics, we use Vialux STAR-07 core [Vialux, [n.d.]], whose full on/full off contrast ratio is 2000 and aperture is $f/2.6$. The projection optics module has two ports, one connecting the DMD and the other connecting the light source. The light entering the module is first spatially smoothed by an integration rod, to create homogeneous lighting, and directed onto the DMD by relay optics. For further details, we refer to the application report provided by Texas Instruments [Texas Instruments, [n.d.]b].

We use Texas Instruments DLP7000 DMD, which has a spatial resolution of 1024×768 pixels. For the light source, we use the LED-ENGIN LZP-04MD00 — an RGBW LED system whose red, green, blue, and white LEDs output 330, 820, 35, and 1785 lumens, respectively. We use Texas Instruments LM3409HV chip to drive the LED and program an FPGA (Terasic DE0-Nano) to synchronize the LEDs and DMD. For the sake of simplicity, we update all micromirrors jointly with minimum bitplane switching time set to t=50 us. With our system setup, we are able to modulate the LEDs with 20 MHz (with up to 10 bits of intensity control) and can achieve 16-bit, RGB projection at 49 fps and 6% loss of brightness (by assigning $n_1 = 9$ bits to BLM). We could also replace the FPGA and use a low-cost Arduino board to synchronize the LEDs and DMD, as shown in Code File 1 (Ref. [Rick Chang *et al.*, 2016]. With the speed of Arduino Uno board, we can easily achieve 12-bit projection ($n_1 = 4, n_2 = 8$).

5.6.3 Experimental Results

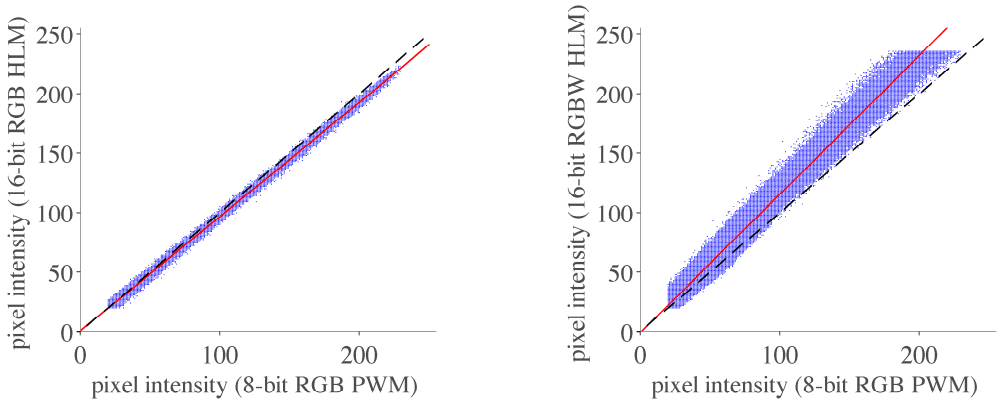
To validate the performance of our prototype, we use a PointGray Grasshopper camera to analyze the projected photograph.



(a) Measured pixel intensities.

(b) Zoom in of the green region.

Figure 5.6: Measured pixel intensities in grayscale projections of the traditional 8-bit PWM and the proposed 16-bit HLM.



(a) RGB projection (n_1=9, n_2=7). The slope of the regression line is 0.96.

(b) RGBW projection (n_1=10, n_2=6). The slope of the regression line is 1.17.

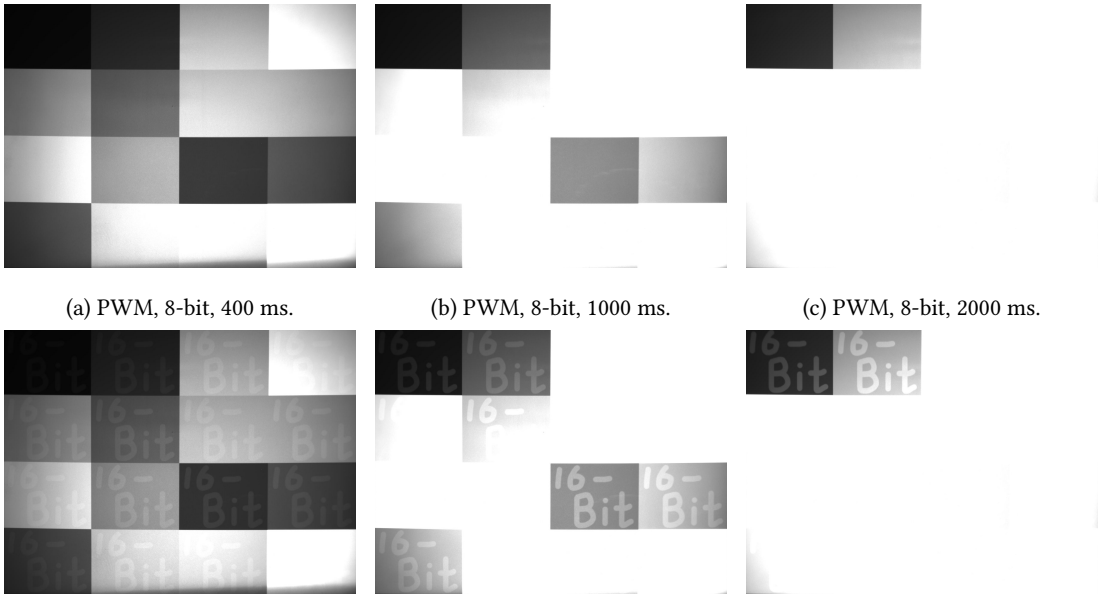
Figure 5.7: **Measured pixel intensities (blue dots) in RGB and RGBW projections of the traditional PWM and the proposed HLM.** The dashed line represents the line $y = x$, and the red line represents the least-squares regression line of the measured pixel intensities.

High Bit-depth Projection

We first validate our claim that proposed HLM coding can achieve 16-bit high bit-depth projection at a small loss of brightness. We project images in which all pixels having the same grayscale value with HLM ($n_1 = 8$) and average the captured pixel intensities in each photograph. As can be seen in Figure 5.6, the proposed 16-bit HLM successfully produce individual intensity levels which traditional 8-bit cannot produce. Note that the discontinuities of 16-bit HLM in Figure 5.6 are caused by the natural delay in LED switching and can be removed by precisely adjusting the pulse-width of the 'on' duration of the LED in each binary bit.

Loss in Brightness

We validate the claim that the proposed HLM coding causes only a marginal loss in brightness. We project multiple scenes (some of which are shown in Figure 5.8, Figure 5.9, and Figure 5.10) using both 8-bit PWM coding and 16-bit HLM coding. Figure 5.7 compares the intensity observed at a pixel (of a projected image) when we use the traditional 8-bit PWM to the intensity observed when we use HLM coding at 16-bits. We remove pixels that are either under-exposed or over-exposed. We observe that, on an average, HLM coding (RGB, n_1=9, n_2=7) loses only 4% of brightness.



(a) PWM, 8-bit, 400 ms. (b) PWM, 8-bit, 1000 ms. (c) PWM, 8-bit, 2000 ms.

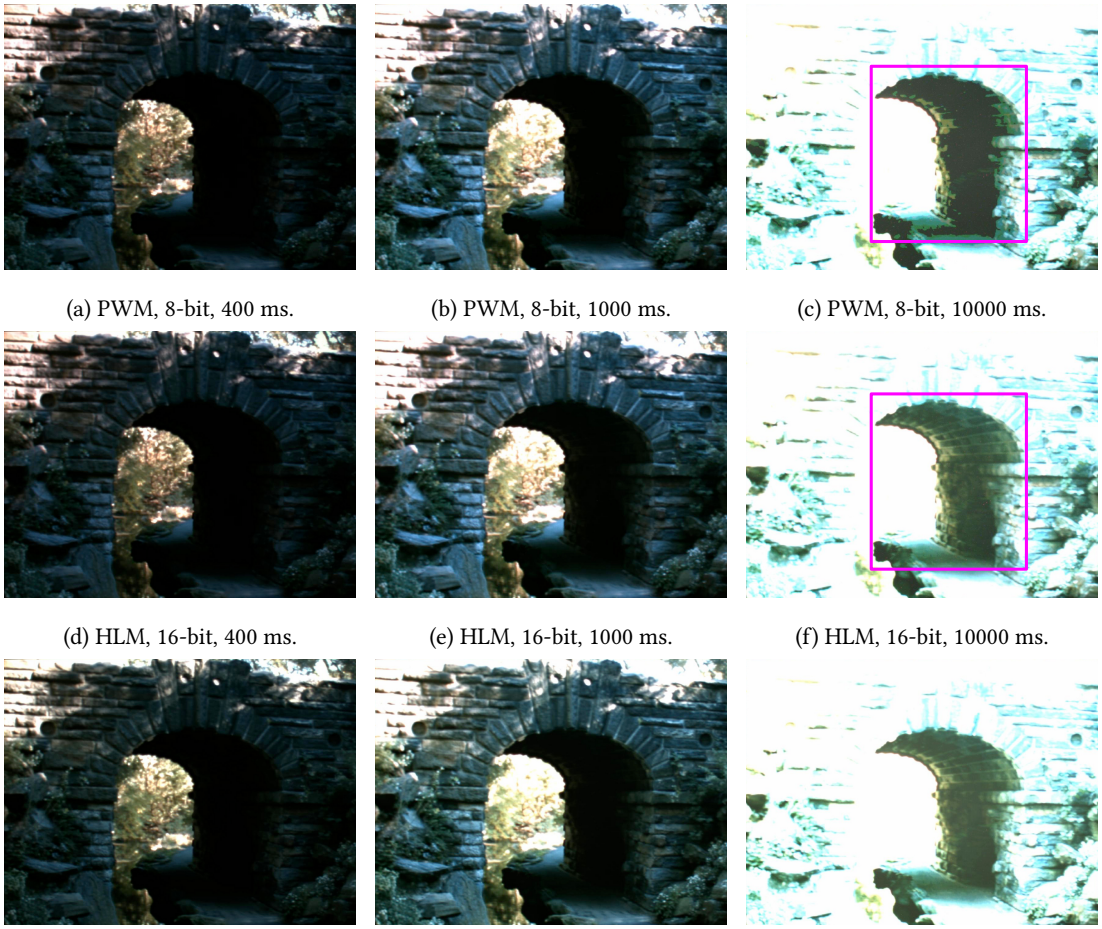
(d) HLM, 16-bit, 400 ms. (e) HLM, 16-bit, 1000 ms. (f) HLM, 16-bit, 2000 ms.

Figure 5.8: Unprocessed photographs of grayscale projection results.

We compare the traditional 8-bit coding to the 4-LED RGB+White (RGBW) system. Here, we observe that, despite using more LEDs and hence a 33% penalty in temporal dithering, HLM coding with 4 LEDs effectively increases brightness by 17%, compared to the traditional 8-bit RGB PWM. However, the RGBW projection mostly benefits the pixels that are gray-toned; this results in a higher variance of pixel intensities shown in Figure 5.7.

8-bit *vs.* 16-bit Projection

We now compare the tradition 8-bit projection to the 16-bit projection on our prototype. To generate the input for the 16-bit projector, given a high dynamic-range radiance image, we first perform primitive tone-mapping with exposure and contrast adjustment, then we quantize the radiance image to 16 bits and 8 bits per color channel. As 16-bit images have 256× more intensity levels per color channel and 16-million times more RGB colors than the 8-bit ones, subtle details are preserved in 16-bit projection and less quantization noise (*e.g.*, banding effects) is observed. We demonstrate grayscale projection in Figure 5.8, RGB and RGBW projection in Figure 5.9 and Figure 5.10. In the figures, we present unprocessed photographs of projected images under three different exposure time settings. We mark the regions where the difference between the 8-bit and the 16-bit projected results can be perceived in a



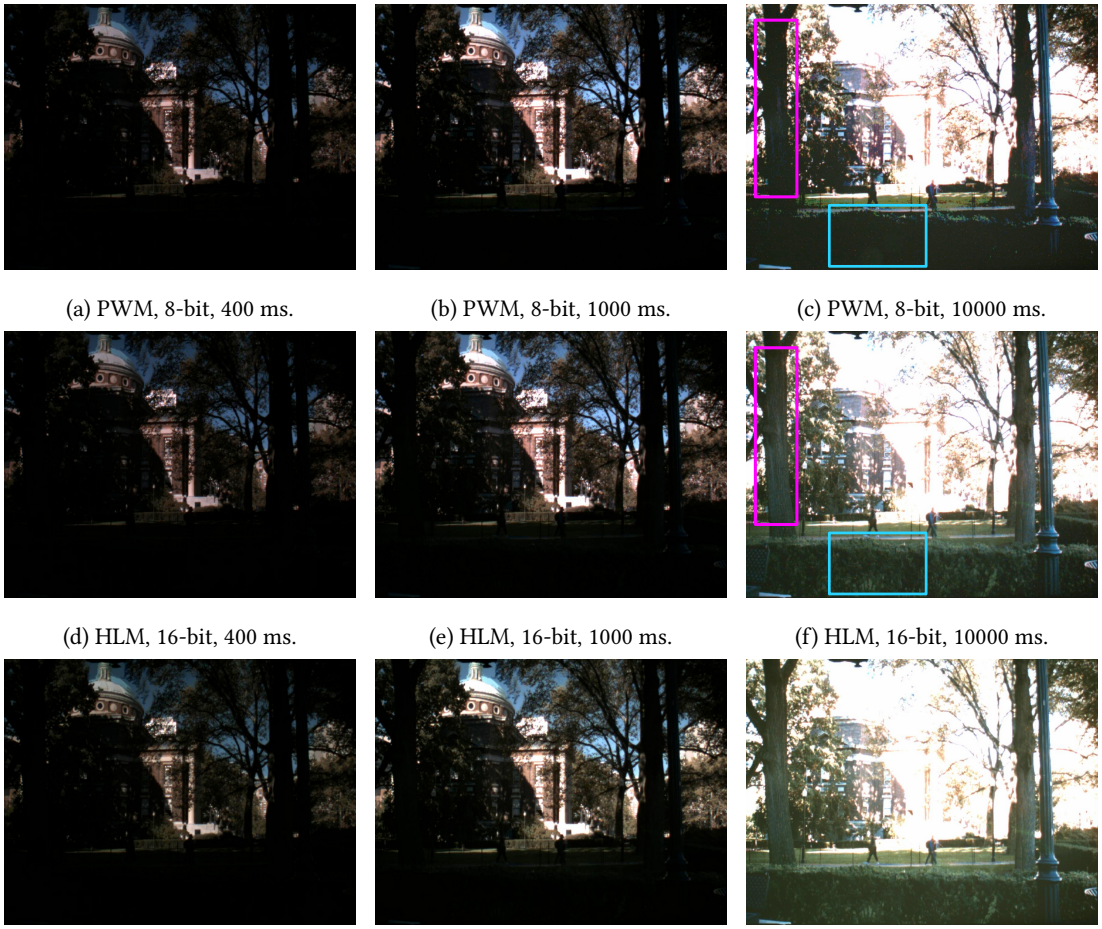
(a) PWM, 8-bit, 400 ms. (b) PWM, 8-bit, 1000 ms. (c) PWM, 8-bit, 10000 ms.

(d) HLM, 16-bit, 400 ms. (e) HLM, 16-bit, 1000 ms. (f) HLM, 16-bit, 10000 ms.

(g) HLM, 16-bit (RGBW), 0.4 s. (h) HLM, 16-bit (RGBW), 1 s. (i) HLM, 16-bit (RGBW), 10 s.

Figure 5.9: Unprocessed photographs of color projection results.

dark environment. Due to the higher bit-depth, we are able to reproduce image details that are otherwise lost with traditional 8-bit projection, for example, the '*16-bit*' in Figure 5.8, the inside of the bridge in Figure 5.9, the checkerboard in the darker area of Figure 5.2, and the bushes and tree trunks in Figure 5.10. To convert the original RGB images to RGBW, for each pixel location, we set the white value to be the minimum in the three channels and subtract the value from the original channels. Based on the assumption that brighter objects tend to have lower-saturated colors, in the RGBW projection, we deliberately set the luminance output of the white LED to be higher than the RGB ones. This highlights the bright spots in the scene and makes the images more vivid, which can be seen from the brighter white regions in Figure 5.9 and Figure 5.10.



(a) PWM, 8-bit, 400 ms. (b) PWM, 8-bit, 1000 ms. (c) PWM, 8-bit, 10000 ms.

(d) HLM, 16-bit, 400 ms. (e) HLM, 16-bit, 1000 ms. (f) HLM, 16-bit, 10000 ms.

(g) HLM, 16-bit (RGBW), 0.4 s. (h) HLM, 16-bit (RGBW), 1 s. (i) HLM, 16-bit (RGBW), 10 s.

Figure 5.10: Unprocessed photographs of color projection results.

5.7 Conclusion

We propose hybrid light modulation, whose novel light intensity control introduces a new design space for high bit-depth projection with wider color gamut and higher frame rates. When applied to high bit-depth projection, hybrid light modulation avoids the significant drop of frame rate in pulse-width modulated projection as well as a drop in brightness in binary light-modulated projectors. The proposed projector requires only a single DMD, projection optics, and a modulated LED light source — similar to most commercial DMD-based projectors; besides, no preprocessing algorithm is needed to adopt the proposed coding scheme. Thereby only a few modifications and additional costs are needed in

order to adopt this technology to existing projectors. Hybrid light modulation coding can also benefit existing high dynamic-range projection techniques, including novel prism designs and light reallocation projectors, to achieve higher bit depth and to expand the color gamut. The proposed method also benefits VR displays by improving their color reproduction and their frame rate. Specifically, the significantly improved frame rate enables us to display dense focal stacks in a multifocal VR display.

6 Building a Virtual World with Dense Focal Stacks

The human eye automatically changes the focus of its lens to provide sharp, in-focus images of objects at different depths. While convenient in the real world, for VR/AR applications, this focusing capability of the eye often causes the vergence-accommodation conflict (VAC) [Hua, 2017, Kramida, 2016] that prevents us to use the devices for a long period of time.

In the real world, the vergence cue and the accommodation cue act in synchrony. However, most commercial VR/AR displays render scenes by manipulating the disparity of the images shown to each eye. Given that the display is at a fixed distance from the eyes, the corresponding accommodation cues are invariably incorrect, leading to a conflict between vergence and accommodation that can cause discomfort, fatigue, and distorted 3D perception, especially after long durations of usage [Hoffman *et al.*, 2008, Vishwanath and Blaser, 2010, Watt *et al.*, 2005, Zannoli *et al.*, 2016]. While many approaches have been proposed to mitigate the VAC, it remains one of the most important challenges for VR and AR displays.

From Chapter 4, we have learned that the ability to generate dense focal stacks can effectively reduce VAC and increase the spatial resolution across the depth range. Our analysis also tells us the number of focal planes we need. We have designed and built a high frame-rate and high bit-depth display that enables us to rapidly refresh content on focal planes in Chapter 5. Now, we are ready to build a multifocal display with dense focal stacks. Specifically, our prototype system is capable of displaying 1600 focal planes per second, which can be used to display scenes with 40 focal planes per frame at 40 frames per second. As a consequence, we are able to render virtual worlds at a realism that is hard to achieve with current multifocal display designs.

What We Will Demonstrate in this Chapter

This chapter introduces the design of a novel multifocal display that produces three-dimensional scenes by displaying dense focal stacks. In this context, we make the following contributions:

- *High-speed focal-length tracking.* The core contribution of this chapter is a system for real-time tracking of the focal length of a focus-tunable lens at microsecond-scale resolutions. We achieve this by measuring the deflection of a laser incident on the lens.
- *Prototype.* We build a proof-of-concept prototype that is able to produce 40 8-bit focal planes per frame with 40 fps. This corresponds to 1600 focal planes per second — a capability that is an order of magnitude greater than competing approaches.

6.1 Generating Dense Focal Stacks

We now have a clear goal — designing a multifocal display supporting a very dense focal stack, which enables display high-resolution images across a wide accommodation range. A typical multifocal display is composed of a display panel and a focus-tunable lens. You may think that with the 2000-fps display we built in Chapter 5, we can build a multifocal display that can show 2000 focal planes per second. It turns out that thing is not as easy, and it is due to the focus-tunable lens.

6.1.1 Focus-tunable Lenses

The ability to change its focal length of a focus-tunable lens can be implemented in one of many ways; for example, by changing the curvature of a liquid lens [Optotune, 2017, Varioptic, 2017], the state of a liquid-crystal lens [Jamali *et al.*, 2018a,b], the polarization of a waveplate lens [Tabiryan *et al.*, 2015], or the relative orientation between two carefully designed phase plates [Bernet and Ritsch-Marte, 2008]. The response time of the devices inevitably constrains their ability to change the focal length rapidly.

Among the methods above, liquid-based focus-tunable lenses are the most matured and have widely been used in many multifocal and varifocal displays [Johnson *et al.*, 2016, Konrad *et al.*, 2016, Liu *et al.*, 2008, Padmanaban *et al.*, 2017]. Liquid-based focus-tunable lenses from Optotune [Optotune, 2019] is composed of a container filled with an optical fluid. The container is made of by an elastic polymer membrane whose shape changes according to the pressure applied from the optical fluid. By controlling the pressure with an actuator, the curvature of the membrane and hence the focal length of the lens can be programmed.

Most focus-tunable liquid lenses have a settling time longer than 5 ms [Optotune, 2019, Varioptic, 2017]. This means that every time we change the focal-length configuration, it takes 5 ms for the focal length to be set. Hence, in order to wait for the lens to settle so that the displayed image is rendered at the desired depth, we can output at most 200 focal planes per second. For a display operating with

30-60 frames per second (fps), this would imply anywhere between three and six focal planes per frame, which is woefully inadequate.

6.1.2 Oscillating Focus

The proposed display relies on the observation that, while focus-tunable lenses have long settling times, their frequency response is rather broad and has a cut-off upwards of 1000 Hz []optotune. This suggests that we can drive the lens with excitations that are radically different from a simple step edge (*i.e.*, a change in voltage). For example, we could make the lens sweep through its entire gamut of focal lengths at a high frequency simply by exciting it with a sinusoid or a triangular voltage of the desired frequency. If we can subsequently track the focal length of the lens in real-time, we can accurately display focal planes at any depth without waiting for the lens to settle. In other words, by driving the focus-tunable lens to periodically sweep the desired range of focal lengths and tracking the focal length at high-speed and in real-time, we can display numerous focal planes, as long as the display supports the required frame rate.

6.1.3 Focal-Length Tracking

While oscillating the focus-tunable lens helps avoid the long settling time, it sacrifices our ability to set the focal length manually. While the optical power of focus-tunable lenses is controlled by an input voltage or current, simply measuring these values only provides inaccurate and biased estimates of the focal length. This is due to the time-varying transfer functions of tunable lenses, which are known to be sensitive to operating temperature and irregular motor delays. Figure 6.1 shows the error in the focal length error of a tunable lens which is driven by its standard driver [Optotune, 2019]. Even though the driver uses the temperature of the device to compensate for the drift in the power of the lens, it does not eliminate the errors, especially when there is a rapid change in the input signal. Since we oscillate the focus-tunable lens, the errors can easily accumulate, and we will display focal planes at wrong depths.

Optically Probing the Tunable Lens

Instead of relying solely on controlling the input voltage, we propose to estimate the focal length by probing the tunable lens optically. This enables robust estimations that are invulnerable to the unexpected factors.

In order to measure the focal length, we send a collimated infrared laser beam through the edge of the focus-tunable lens. Since the direction of the outgoing beam depends on the focal length, the laser

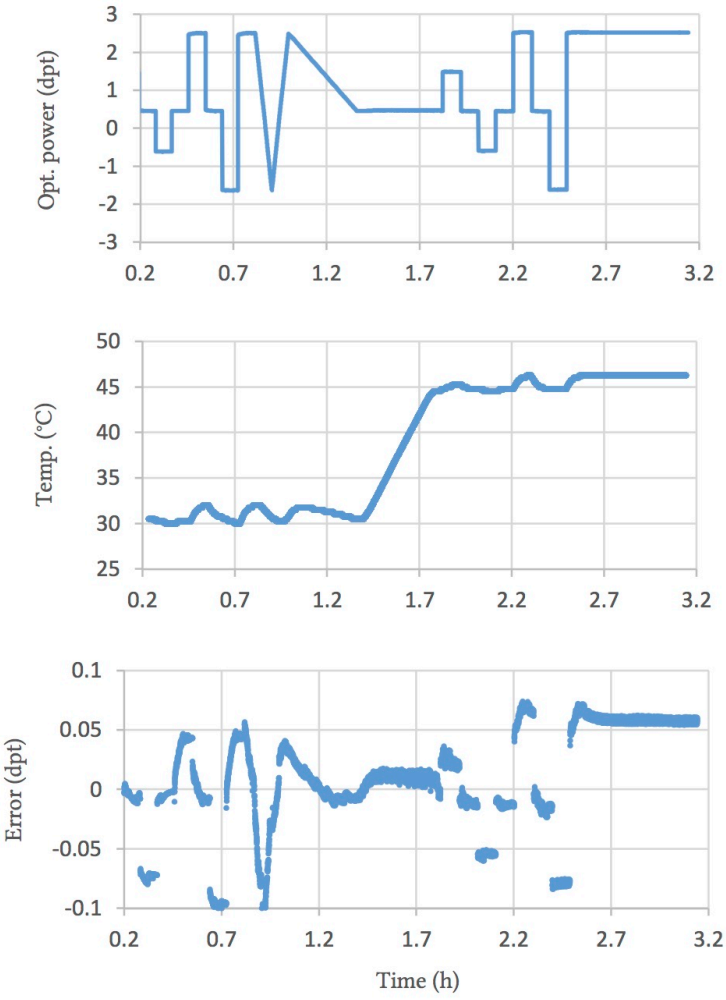


Figure 6.1: **Error in optical power of tunable lens.** The tunable lens is controlled by its standard driver. (a) shows the target optical power over time. (b) shows the device temperature. (c) is the error of optical power of the tunable lens. Figure courtesy Optotune [2019].

beam changes direction as the focal length changes. There are many approaches to measure this change in direction, including using a one-dimensional pixel array or an encoder system. In our prototype, we use a one-dimensional position sensing detector (PSD) to enable fast and accurate measurement of the location. The schematic is shown in Figure 6.2a.

The focal length of the laser is estimated as follows. We first align the laser so that it is parallel to the optical axis of the focus-tunable lens. After deflection by the lens, the beam is incident on a spot on the PSD whose position, as shown in Figure 6.2b, is given as

$$h = a\left(\frac{d_p}{f_x} - 1\right), \tag{6.1}$$

where f_x is the focal length of the lens, d_p is the distance measured along the optical axis between the lens and the PSD, and h is the distance between the optical center of the lens and the spot the laser is incident on. Note that the displacement h is an affine function of the optical power of the tunable lens.

We next discuss how the location of the spot is estimated from the PSD outputs. A PSD is composed of a photodiode and a resistor distributed throughout the active area. The photodiode has two connectors at its anode and a common cathode. Suppose the total length of the active area of the PSD is ℓ. When a light ray reaches a point at h on the PSD, the generated photocurrent will flow from each anode connector to the cathode with amount inversely proportional to the resistance in between. Since resistance is proportional to length, we have the ratio of the currents in the anode and cathode as

$$\frac{i_1}{i_2} = \frac{R_2}{R_1} = \frac{\frac{\ell}{2} - h}{\frac{\ell}{2} + h}, \text{ or } h = \frac{\ell}{2}\frac{i_2 - i_1}{i_2 + i_1}. \tag{6.2}$$

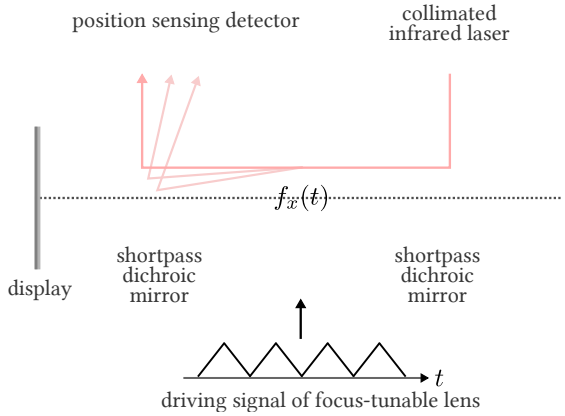
Combining (6.2) and (6.1), we have

$$\frac{1}{f_x} = \frac{\ell}{2ad_p}r + \frac{1}{d_p}, \text{ where } r = \frac{i_2 - i_1}{i_2 + i_1}. \tag{6.3}$$

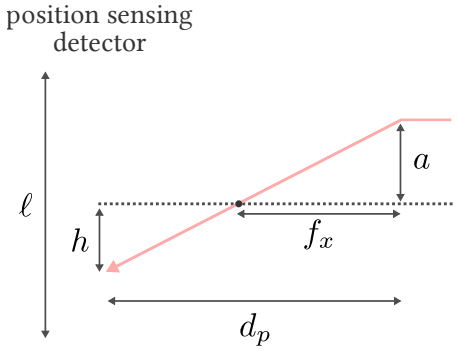
As can be seen, the optical power of the tunable lens $\frac{1}{f_x}$ is an affine function of r. With simple calibration (to get the two coefficients), we can easily estimate the value.

6.1.4 The Need for Fast Displays

In order to display multiple focal planes within one frame, we also require a display that has a frame rate greater than or equal to the focal-plane display rate. To achieve this, we use the digital micromirror device (DMD)-based projector that we built in Chapter 5 as our display. Commercially available DMDs can easily achieve upwards of 20,000 bitplanes per second. Following the design in [Chang *et al.*, 2016], we modulate the intensity of the projector's light source to display 8-bit images; this enables us to



(a) Schematic of focal-length tracking



(b) Optical layout

Figure 6.2: **Illustrations of the focal-length tracking module.** (a) The focal-length tracking system is composed of two shortpass dichroic mirrors and a position sensing detector. The dichroic mirror allows visible light to pass through but reflects the infrared light ray emitted from the collimated laser. (b) The position of the laser spot on the position sensing detector is an affine function of the optical power of the lens.

display each focal plane with 8-bits of intensity and generate as many as $20,000/8 \approx 2,500$ focal planes per second.

Other display technologies can be used with the proposed method. For example, OLED and microLED displays are in principle capable of refreshing at multiple thousands Hz. Since these displays are thinner and has higher contrast than DMD-based projectors, switching to these technologies can potentially improve the proposed display in terms of bulk and image quality.

6.1.5 Design Criteria and Analysis

We now analyze the system in terms of various desiderata and the system configurations required to achieve them. Figure 6.3 gives an overview of the proposed multifocal display. Given our estimation of the current focal length of the tunable lens, the depth $v(t) > 0$ of the currently-displaying focal plane can be calculated by the thin lens formula:

$$\frac{1}{d_o} + \frac{1}{-v(t)} = \frac{1}{f(t)}, \tag{6.4}$$

where d_o is the distance between the display and the lens, and $f(t)$ is the current focal length of the tunable lens.

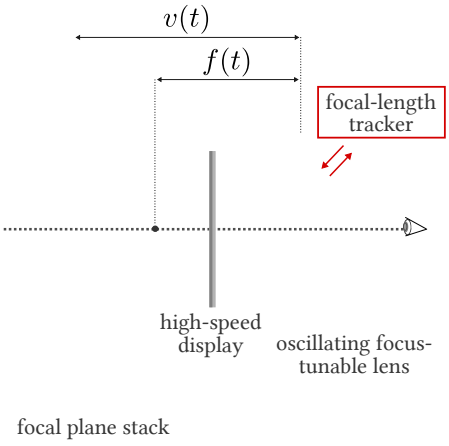


Figure 6.3: **Overview of the proposed multifocal display.** The proposed display outputs dense focal plane stacks by tracking the focal-length of an oscillating focus-tunable lens. The depths of the focal planes are independent to the viewer, and thereby eye trackers are optional.

Achieving a Full Accommodation Range

A first requirement is that the system be capable of supporting the full accommodation range of typical human eyes, *i.e.*, generate focal planes from 25 cm to infinity. Suppose the optical power of the focus-tunable lens ranges from $D_1 = \frac{1}{f_1}$ to $D_2 = \frac{1}{f_2}$ diopter. From (6.4), we have

$$\frac{1}{-v(t)} = \frac{1}{f_x(t)} - \frac{1}{d_o} = -\left(\frac{1}{d_o} - D_x(t)\right), \tag{6.5}$$

where d_o is the distance between the display unit and the tunable lens, $v(t)$ is the distance of the virtual image of the display unit from the lens, $f_x(t) \in [f_2, f_1]$ is the focal length of the lens at time t, and $D_x(t) = \frac{1}{f_x(t)}$ is the optical power of the lens in diopter. Since we want $v(t)$ to range from 25 cm to infinity, $1/v(t)$ ranges from 4 m^{-1} to 0 m^{-1}. Thereby, we need

$$4 - D_1 \leq \frac{1}{d_o} \leq D_2.$$

An immediate implication of this is that $D_2 - D_1 \geq 4$, *i.e.*, to support the full accommodation range of a human eye, we need a focus-tunable lens whose optical power spans at least 4 diopters. We have more choice over the actual range of focal lengths taken by the lens. A simple choice is to set $1/f_2 = D_2 = 1/d_o$; this ensures that we can render focal planes at infinity; subsequently, we choose f_1 sufficiently large to cover 4 diopters. By choosing a small value of f_2, we can have a small d_o and achieve a compact display.

Field-of-View

The proposed display shares the same field-of-view and eye box characteristics with other multifocal displays. The field-of-view will be maximized when the eye is located right near the lens. This will results in a field-of-view of $2\,\text{atan}\left(\frac{H}{2d_o}\right)$, where H is the height (or width) of the physical display (or its magnification image via lensing). When the eye is further away from the lens, the numerical aperture will limit the extent of the field-of-view. Since the apertures of most tunable lenses are small (around 1 cm in diameter), we would prefer to put the eye as close as the lens as possible. This can be achieved by embedding the dichroic mirror (the right one in Figure 6.2a) onto the rim of the lens. For our prototype that will be described in Section 6.2, we use a $4f$ system to relay the eye to the aperture of the focus-tunable lens. Our choice of the $4f$ system enables a 45-degree field-of-view, limited by the numerical aperture of the lens in the $4f$ system.

There are alternate implementations of focus tunable lenses that have the potential for providing larger apertures and hence, displays with larger field of views. Bernet and Ritsh-Marte [2008] design two phase plates that produce the phase function of a lens whose focal length is determined by the relative orientation of the plates; hence, we could obtain a large aperture focus tunable lens by rotating one of the phase plates. Other promising solutions to enable large-aperture tunable lensing include the Fresnel and Pancharatnam-Berry liquid crystal lenses [Jamali *et al.*, 2018a,b] and tunable metasurface doublets [Arbabi *et al.*, 2018]. In all of these cases, our tracking method could be used to provide precise estimates of the focal length.

Eye Box

The eye box of multifocal displays are often small, and the proposed display is no exception. Due to the depth difference of focal planes, as the eye shifts, contents on each focal plane shift by different amounts, with the closer ones traverse more than the farther ones. This will leave uncovered as well as overlapping regions at depth discontinuities. Further, the severity of the artifacts depends largely on the specific content being displayed. In practice, we observe that these artifacts are not distracting for small eye movements in the order of few millimeters. This problem can be solved by incorporating an eye tracker, as in Mercier *et al.* [2017].

6.1.6 Reduced Maximum Brightness and Energy Efficiency

Key limitations of our proposed design are the reduction in maximum brightness and, depending on the implementation, the energy efficiency of the device. Suppose we are displaying n focal planes per frame and T frames per second. Each focal plane is displayed for $\frac{T}{n}$ second, which is n-times smaller compared to typical VR displays with one focal plane. For our prototype, we use a high power LED to compensate for the reduction in brightness. Further, brightness of the display is not a primary concern since there are no competing ambient lights sources for VR displays.

Energy efficiency of the proposed method also depends on the type of display used. For our prototype, since we use a DMD to spatially modulate the intensity at each pixel, we waste $\frac{n-1}{n}$ of the energy. This can be completely avoided by adopted by using OLED displays, where a pixel can be completely turned off. An alternate solution is to use a phase spatial light modulator (SLM) [Damberg *et al.*, 2016] to spatially redistribute a light source so that each focal plane only gets illuminated at pixels that need to be displayed; a challenge here is the slow refresh rate of the current crop of phase SLMs. Another option is to use a laser along with a 2D galvo to selectively illuminate the content at each depth plane; however, 2D galvos are often slow when operated in non-resonant modes.

6.2 Proof-of-Concept Prototype

In this section, we present a lab prototype that generates a dense focal stack using high-speed tracking of the focal length of a tunable lens and a high-speed display.

6.2.1 Implementation Details

The prototype is composed of three functional blocks: the focus-tunable lens, the focal-length tracking device, and a DMD-based projector. All the three components are controlled by an FPGA (Altera DE0-nano-SOC). The FPGA drives the tunable lens with a digital-to-analog converter (DAC), following Algorithm 1. Simultaneously, the FPGA reads the focal-length tracking output with an analog-to-digital converter (ADC) and uses the value to trigger the projector to display the next focal plane. Every time a focal plane has been displayed, the projector is immediately turned off to avoid blur caused by the continuously changing focal-length configurations. A photo of the prototype is shown in Figure 6.4. In the following, we will introduce each component in detail.

Calibration

In order to display focal planes at correct depths, we need to know the corresponding PSD tracking outputs. From equations (6.3) and (6.5), we have

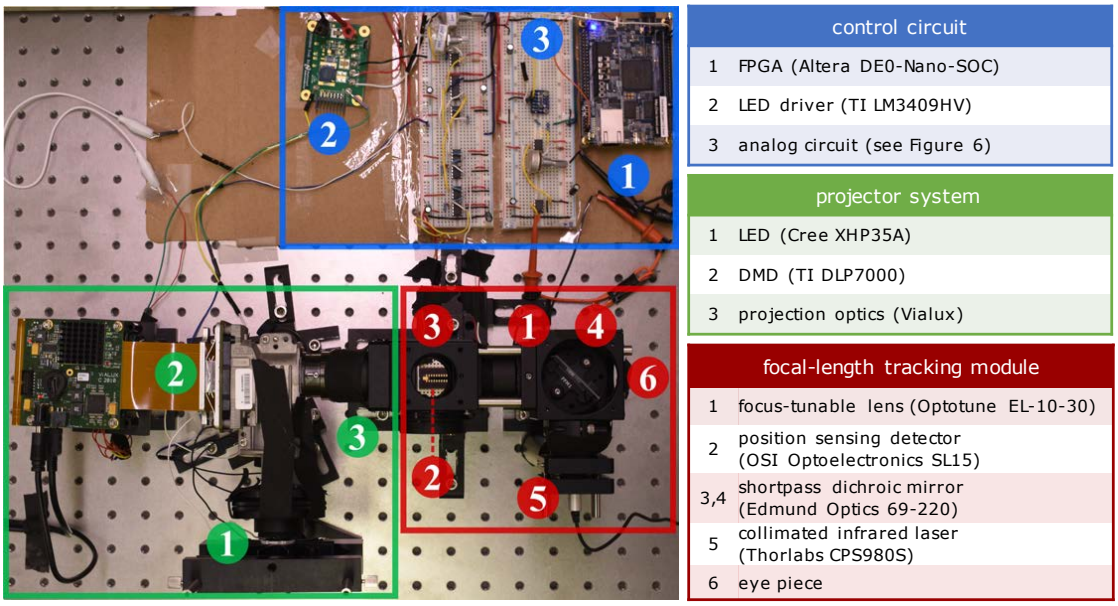
$$\frac{1}{v(t)} = \frac{1}{d_o} - \frac{1}{d_p} - \frac{\ell}{2ad_p} r(t) = \alpha + \beta r(t). \tag{6.6}$$

Thereby, we can estimate the current depth $v(t)$ if we know α and β, which only requires two measurements to estimate. With a camera focused at $v_a = 25$ cm and $v_b = \infty$, we get the two corresponding ADC readings r_a and r_b. The two points can be accurately measured, since the depth-of-field of the camera at 25 cm is very small, and infinity can be approximated as long as the image is far away. Since Equation (6.6) has an affine relationship, we only need to divide $[r_a, r_b]$ evenly into the desired number of focal planes.

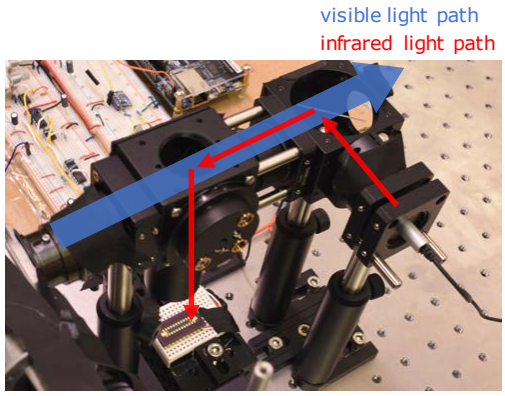
Control Algorithm

The FPGA follows Algorithm 1 to coordinate the tunable lens and the projector. On a high level, we drive the tunable lens with a triangular wave by continuously increasing/decreasing the DAC levels. We simultaneously detect the PSD's DAC reading r to trigger the projection of focal planes. When the last/first focal plane is displayed, we switch the direction of the waveform. Note that while Algorithm 1 is written in serial form, every module in the FPGA runs in parallel.

The control algorithm is simple yet robust. It is known that the transfer function of the tunable lens is sensitive to many factors, including device temperature and unexpected motor delay and errors [Optotune, 2017]. In our experience, even with the same input waveform, we observe different offsets, peak-to-peak values on the PSD output waveform for each period. Since the algorithm does not drive



(a) Photograph of the prototype and its component list



(b) Light path for infrared and visible light

Figure 6.4: **Prototype multifocal display with 40 focal planes.** The prototype is composed of a projector, the proposed focal-length tracking module, and the control circuits. (b) The two shortpass dichroic mirrors allow visible light to pass through and reflect infrared. The enables us to create individual light path for each of them.

the tunable lens with fixed DAC values and instead directly detect the PSD output (*i.e.*, the focal length of the tunable lens), it is robust to these unexpected factors. However, the robustness comes with a price.

ALGORITHM 1: Tunable-lens and focal-plane control

Data: n target PSD triggers $r_1, \ldots, r_n$

Input: PSD ADC reading r

Output: Tunable-lens DAC level L, projector display control signal

Initialize $L = 0$, $\Delta L = 1$, $i = 1$

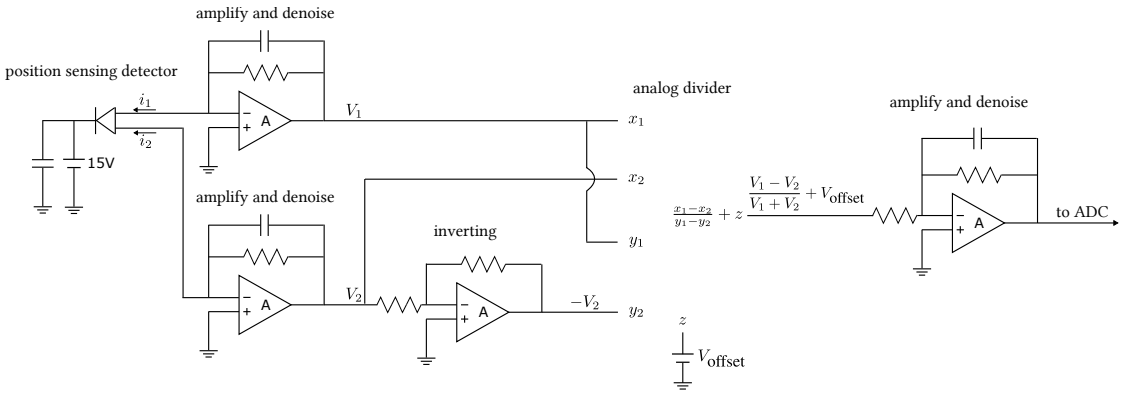
repeat
  $L \leftarrow L + \Delta L$
  if $|r - r_i| \leq \Delta r$ **then**
    Display focal plane i and turn it off when finished.
    $i \leftarrow i + \Delta L$
    if $\Delta L == 1$ *and* $i > n$ **then**
      Change triangle direction to down: $\Delta L \leftarrow -1$, $i \leftarrow n$
    else if $\Delta L == -1$ *and* $i < 1$ **then**
      Change triangle direction to up: $\Delta L \leftarrow +1$, $i \leftarrow 1$
  end
until *manual stop*;

Due to the motor delay, the peak-to-peak value $r_{\max} - r_{\min}$ is often a lot larger than $r_n - r_1$. This causes the frame rate of the prototype (1600 focal planes per second, or 40 focal planes per frame at 40 fps) to be lower than the highest display frame rate (2500 focal planes per second).

Note that since 40 fps is close to the persistence of vision, our prototype sometimes leads to flickering. However, the capability of the proposed device is to increase the number of focal planes per second and as such we can get higher frame rate by trading off the focal planes per frame. For example, we can achieve 60 fps by operating at 26 focal planes per frame.

Focus-Tunable Lens and its Driver

We use the focus-tunable lens EL-10-30 from Optotune [Optotune, 2017]. The optical power of the lens ranges from approximately 8.3 to 20 diopters and is an affine function of the driving current input from 0 to 300 mA. We use a 12-bit DAC (MCP4725) with a current buffer (BUF634) to drive the lens. The DAC provides 200 thousand samples per second, and the current buffer has a bandwidth of 30 MHz. This allows us to faithfully create a triangular input voltage up to several hundred Hertz. The circuit is drawn in Figure 6.5b.



(a) Analog circuit for processing focal-length tracking

(b) Analog circuit for driving focus-tunable lens

Figure 6.5: **Analog circuits used in the prototype.** All the operational amplifiers are TI OPA-37, the analog divider is TI MPY634, and the current buffer is TI BUF634. All denoising RC circuits have cutoff frequency at 47.7 kHz.

Focal-Length Tracking and Processing

The focal-length tracking device is composed of a one-dimensional PSD (SL15 from OSI Optoelectronics), two 800 nm dichroic short-pass mirrors (Edmundoptics #69-220), and a 980 nm collimated infrared laser (Thorlabs CPS980S). We drive the PSD with a reverse bias voltage of 15 V. This enables us to have 15 um precision on the PSD surface and rise time of 0.6 us. Across the designed accommodation range, the laser spot traverses within 7 mm on the PSD surface, which has a total length 15 mm. This allows us to accurately differentiate up to 466 focal-length configurations.

The analog processing circuit has three stages — amplifier, analog calculation, and an ADC, as shown in Figure 6.5a. We use two operational amplifiers (TI OPA-37) to amplify the two output current of the PSD. The gain-bandwidth of the amplifiers are 45 MHz, which can fully support our desired operating

speeds. We also add a low-pass filter with a cut-off frequency of 47.7 kHz at the amplifier, as a denoising filter. The computation of $r(t)$ is conducted with two operational amplifiers (TI OPA-37) and an analog divider (TI MPY634). We use a 12-bit ADC (LTC2308) with a rate of 200 thousand samples per second to port the analog voltage to the FPGA.

Overall, the latency of the focal-length tracking circuit is ~ 20 us. The bottleneck is the low-pass filter and the ADC; rest of the components have time responses in nanoseconds. Note that in 20 us the focal length of the tunable lens changes by 0.01 diopters — well below the detection capabilities of the eye [Campbell, 1957]. Also, the stability of the acquired focal stack (which took a few hours to capture) indicates that the latency was either minimal or at least predictable and can be dealt with by calibration.

DMD-based Projector

The projector is composed of a DLP-7000 DMD from Texas Instruments, projection optics from Vialux, and a high-power LED XHP35A from Cree. We control the DMD with a development module Vialux V-7000. We update the configuration of micro-mirrors every 50 us. Following Chang *et al.* [2016], we use pulse-width modulation, performed through a LED driver (TI LM3409HV), to change the intensity of the LED concurrently with the update of micro-mirrors. This enables us to display at most 2500 8-bit images per second.

For simplicity, we preload each of the 40 focal planes onto the development module. Each focal stack requires $40 \times 8 = 320$ bitplanes, and thereby, we can store up to 136 focal stacks on the module. The lack of video-streaming capability needs further investigation to make it practical; it could potentially be resolved by using the customized display controller in [Lincoln *et al.*, 2017, 2016] that is capable of displaying bitplanes with 80 us latency. This would enable us to display 1562 8-bit focal planes per second. We also note that whether we use depth filtering or not, the transmitted bitplanes are sparse since each pixel has content, at best, at a few depth planes. Thereby, we do not need to transmit the entire 320 bitplanes.

Note that we divide the 8 bitplanes of each focal planes into two groups of 4 bitplanes, and we display the first group when the triangular waveform is increasing, and the other at the downward waveform. From the results that will be presented in Section 6.3, we can see that the images of the two groups align nicely. This demonstrates the high accuracy of the focal-length tracking.

As a quick verification of the prototype, we used the burst mode on the Nikon camera to capture multiple photographs at an aperture of $f/4$, ISO 12,800 and an exposure time of 0.5 ms. Figure 6.6 shows six examples of displayed focal planes. Since a single focal plane requires an exposure time of $50{\times}4 = 0.2$

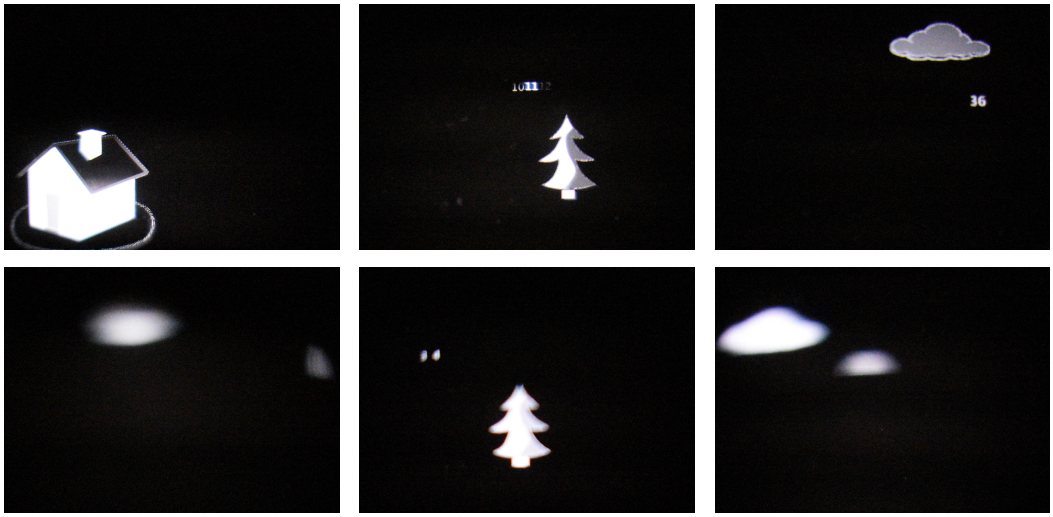


Figure 6.6: **Captured images of focal planes.** Example images that are captured in burst shooting mode with a $f/4$ lens, exposure time equal to 0.5 ms, and ISO equal to $12,800$. Note that in order to capture a single focal plane, we need exposure time of 0.2 ms. Thereby, these images are composed of at most 3 focal planes.

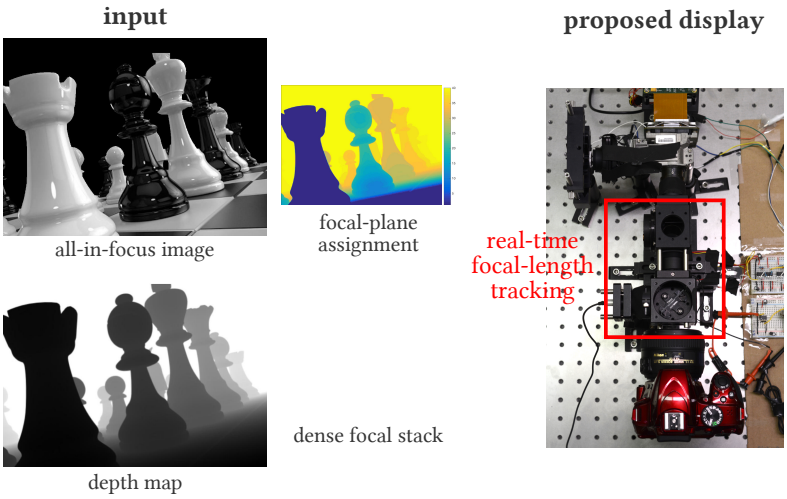
ms, the captured images are composed of at most 3 focal planes.

Putting it Together

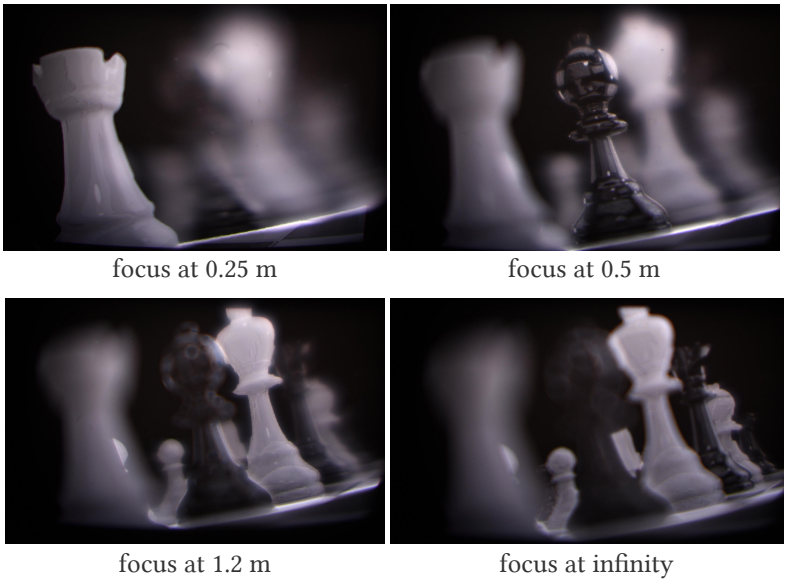
Figure 6.7 gives an overview of using the prototype. Given a virtual scene composed of an all-in-focus image and a per-pixel depth map, we first assign each pixel to their closest focal plane in diopter, and we preload the images onto the DMD memory. The FPGA then synchronizes the focus-tunable lens and the DMD-based projector to show the corresponding focal plane content at the right moment. We evaluate the prototype by placing a camera in front of the focus-tunable lens. Note that the camera is entire independent to the prototype — we are not tracking the focus of the camera.

6.3 Experimental Evaluations

We showcase the performance of our prototype on a range of scenes designed carefully to highlight the important features of our system. The supplemental material has video illustrations that contain full camera focus stacks of all results in this section.



(a) Input and the data flow



(b) Captured photos

Figure 6.7: **Example usage of the prototype.** Our system is capable of generating 1600 focal planes per second, which we use to render 40 focal planes per frame at 40 frames per second. Given a virtual scene composed of an all-in-focus image and a per-pixel depth map, we simply display each pixel in the all-in-focus image at its closest focal plane in diopter. Shown are images captured with a 50mm $f/2.8$ lens focused at different depths from the tunable lens.

6.3.1 Focal-Length Tracking

To evaluate the focal-length tracking module, we measure the input signal to the focus-tunable lens and the PSD output r from an Analog Discovery oscilloscope. The measurements are shown in Figure 6.8. As can be seen, the output waveform matches that of the input. The high bandwidth of the PSD and the analog circuit enables us to track the focal length robustly in real-time. From the figure, we can also observe the delay of the focus-tunable lens ($\sim$ 3 ms).

6.3.2 Depths of Focal Planes

As stated previously, measuring the depth of the displayed focal planes is very difficult. Thereby, we use a method similar to depth-from-defocus to measure their depths. When a camera is focusing at infinity, the defocus blur kernel size will be linearly dependent on the depth of the (virtual) object in diopter. This provides a method to measure the depths of the focal planes.

For each of the focal plane, we display a 3×3 pixels white spot at the center, capture multiple images of various exposure time, and average the images to reduce noise. We label the diameter of the defocus blur kernels and show the results in Figure 6.9. As can be seen, when the blur-kernel diameters can be accurately estimated, *i.e.*, largely defocus spots on closer focal planes, the values fit nicely to a straight line, indicating the depths of focal planes are uniformly separated in diopter. However, as the displayed spot size as a spot come into focus, the estimation of blur kernel diameters becomes inaccurate since we cannot display an infinitesimal spot due to the finite pixel pitch of the display. Since there were no special treatments to individual planes in terms of system design or algorithm, we expect these focal planes to be placed accurately as well.

6.3.3 Characterizing the System Point-Spread Function

To characterize our prototype, we measure its point spread function with a Nikon D3400 using a 50 mm $f/1.4$ prime lens. We display a static scene that is composed of 40 3×3 spots with each spot at a different focal plane. Using the camera, we capture a focal stack of 169 images ranging from 0 to 4 diopters away from the focus-tunable lens. For improved contrast, we remove the background and noise due to dust and scratches on the lens by capturing the same focal stack with no spot shown on the display. Figure 6.10 shows the point spread function of the display at four different focus settings, and a video of this focal stack is attached in the supplemental material. The result shows that the prototype is able to display the spots at 40 depths concurrently within a frame, verifies the functionality of the

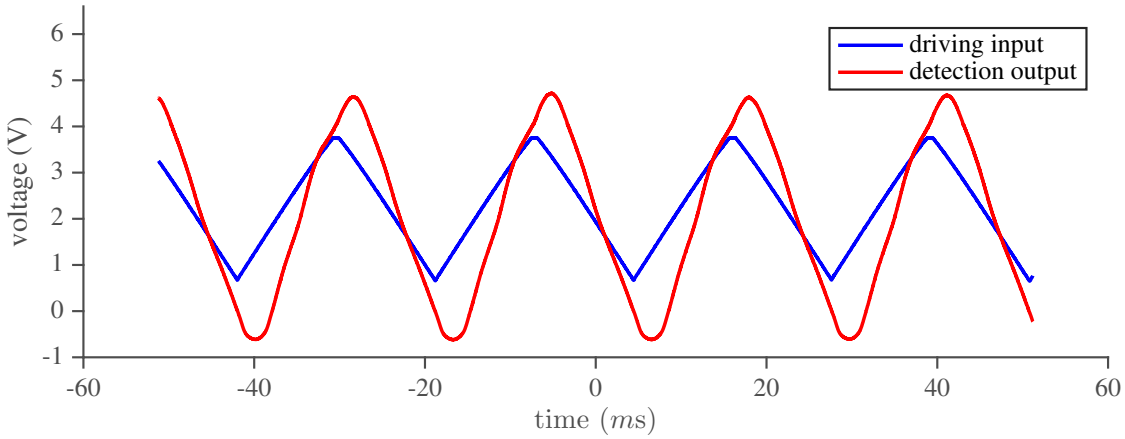


Figure 6.8: **Measurements of the input signal to the tunable lens and the output of the PSD after analog processing.** The output waveform matches that of the input. This shows that the proposed focal-length tracking is viable.

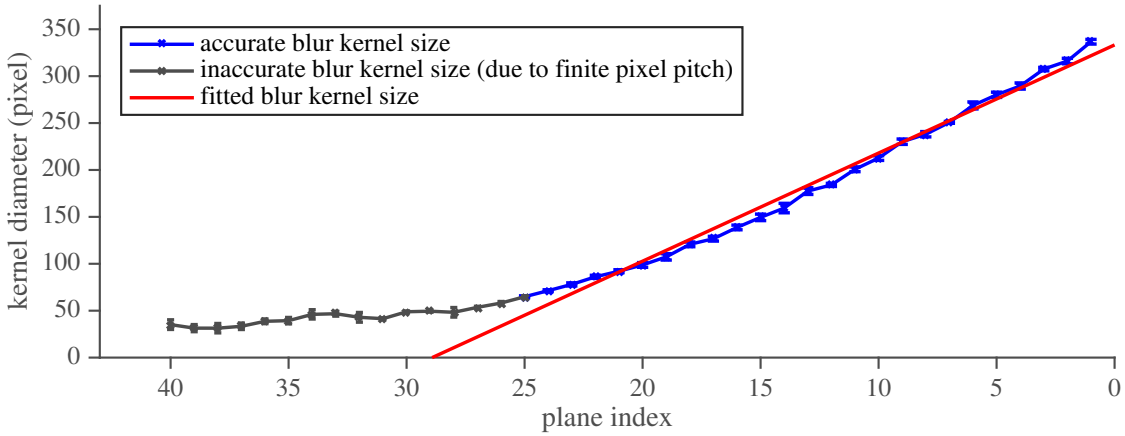
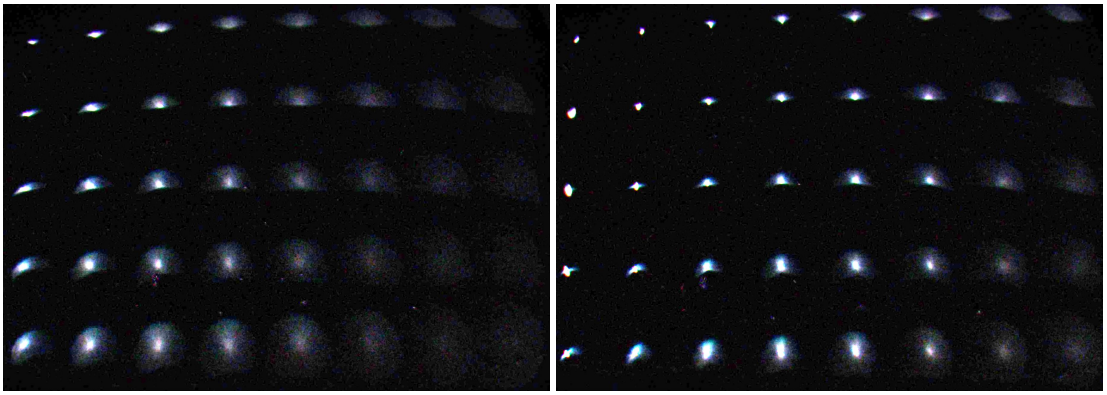


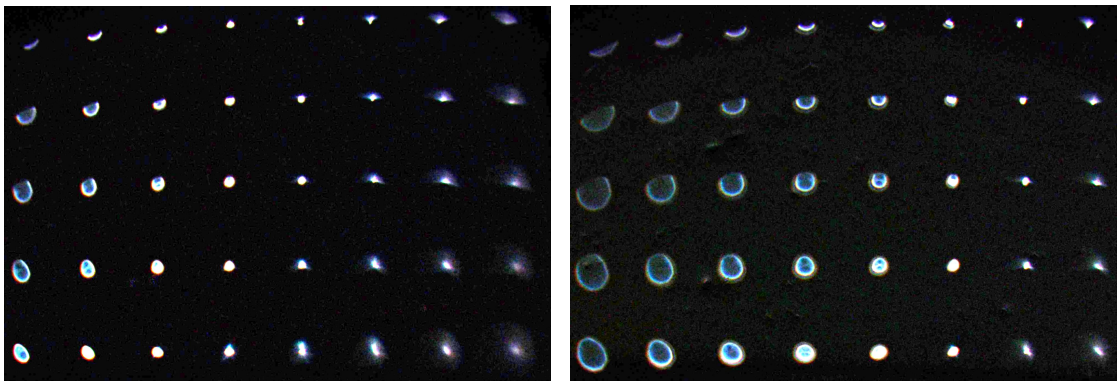
Figure 6.9: **Measured blur kernel diameter by a camera focusing at infinity (plane 40).** Due to the finite pixel pitch, the estimation becomes inaccurate when the spot size is too small (when the spots are displayed on focal planes close to infinity). When the blur kernel size can be accurately estimated, they fit nicely as a linear segment. This indicates the depth of the focal planes are distributed uniformly in diopter.

proposed method. The shape and the asymmetry of the blur kernels can be attributed to the spherical aberration of the focus-tunable lens as well as the throw of the projection lens on the DMD.



(a) Camera focuses at 25 cm

(b) Camera focuses at 30 cm



(c) Camera focuses at ~ 1.2 m

(d) Camera focuses at infinity

Figure 6.10: **Measured point spread function of the prototype.** Each of the 40 points is on a different focal plane — the top-left is closest to the camera and the bottom-right is farthest. For better visualization, we multiply the image by 10 and filter the image with a 4×4 median filter. The results show that the prototype is able to produce 40 distinct focal planes.

6.3.4 Benefits of Dense Focal Stacks

To evaluate the benefit provided by dense focal stacks, we simulate two multifocal displays, one with 4 focal planes and the other with 40 focal planes. The 40 focal planes are distributed uniformly in diopter from 0 to 4 diopters, and the 4-plane display has focal planes at the depth of the 5th, 15th, 25th, and 35th focal planes of the 40-plane display. The scene is composed of 28 resolution charts, each at a different depth from 0 to 4 diopters (please refer to the supplemental material for figures of the entire scene). The dimension of the scene is 1500×2000 pixels.

We render the scene with three methods:

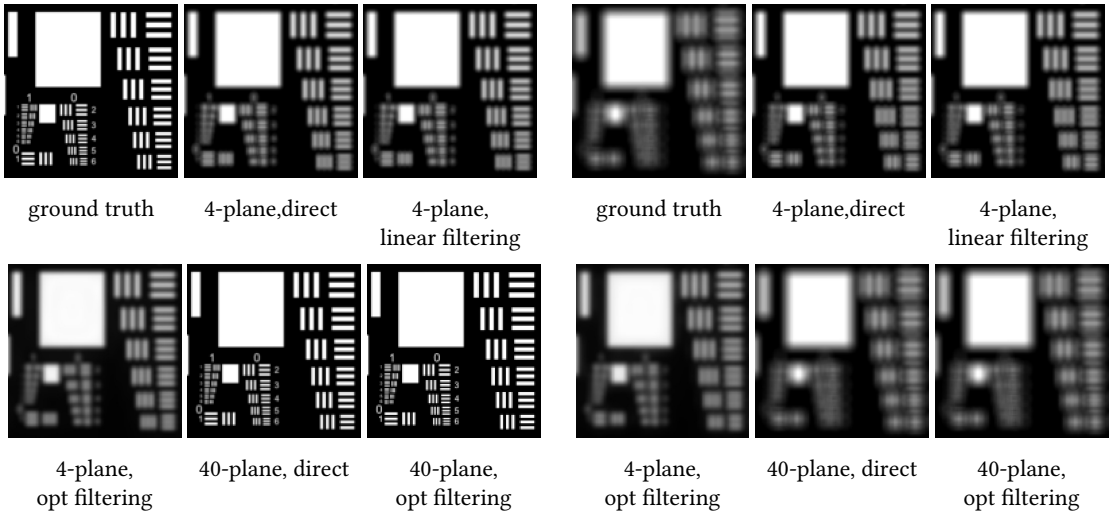
- *No depth filtering*: We directly quantize the depth channel of the images to obtain the focal planes of different depths.
- *Linear depth filtering*: Following [Akeley *et al.*, 2004], we apply a triangular filter on the focal planes based on their depths.
- *Optimization-based filtering*: We follow the formulation proposed in [Mercier *et al.*, 2017]. We first rendered normally the desired retinal images focused at 81 depths uniformly distributed across 0 to 4 diopters in the scene with a pupil diameter of 4 mm. Then we solve the optimization problem to get the content to be displayed on the focal planes. We initialize the optimization process with the results of direct quantization and perform gradient descent with 500 iterations to ensure convergence.

The perceived images of the resolution chart at 0.02 diopters are shown in Figure 6.11; a plane at 0.02 diopters is on a focal plane of the 40-plane display and is at the furthest inter-focal plane of the 4-plane display. Note that we simulate the results with pupil diameter of 4 mm, which is a typical value used to simulated retinal images of human eyes.

As can be seen from the results, the perceived images of the 40-plane display closely follow those of the ground truth — with high spatial resolution if the camera is focused on the plane (Figure 6.11a) and natural retinal blur when the camera is not focused (Figure 6.11b). In comparison, at its inter-plane location (Figure 6.11a), the 4-plane display has much lower spatial resolution than the other display, regardless of the depth filtering methods applied. These results verify our analysis in Chapter 4.

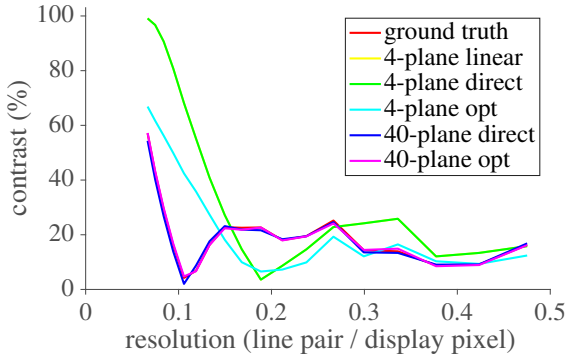
To evaluate the benefit provided by dense focal stacks in providing higher spatial resolution when the eye is focused at an inter-plane location, we implement four multifocal displays with 4, 20, 30 and 40 focal planes, respectively, on our prototype. The 4-plane display has its focal planes on the 5, 15, 25, 35th focal planes of the 40-plane display, and the 20-plane display has its focal planes on all the odd-numbered focal planes. We display a resolution chart on the fifth focal plane of the 40-plane display; this corresponds to a depth plane that all three displays can render.

To compare the worst-case scenario where an eye focuses on an inter-plane location, we focus the camera at the middle of two consecutive focal planes of each of the displays. In essence, we are reproducing the effect of VAC where the vergence cue forces the ocular lens to focus on an inter-focal plane. For the 40-plane display, this is between focal planes five and six. For the 20-plane display, this is on the sixth focal plane of the 40-plane display. And for the 4-plane display, this is on the tenth focal plane of the 40-plane display. We also focus the camera on the estimated inter-plane location of a 30-plane dis-



(a) Camera focuses at 0.02 diopters

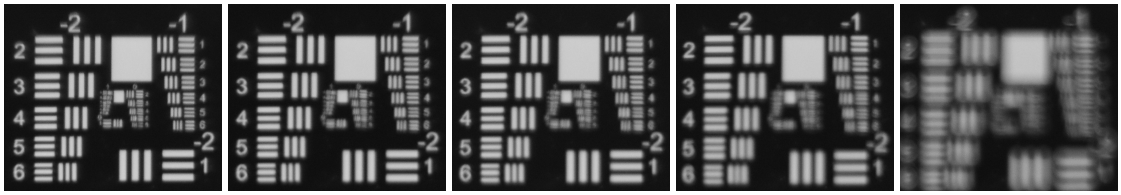
(b) Camera focuses at 0.9 diopters



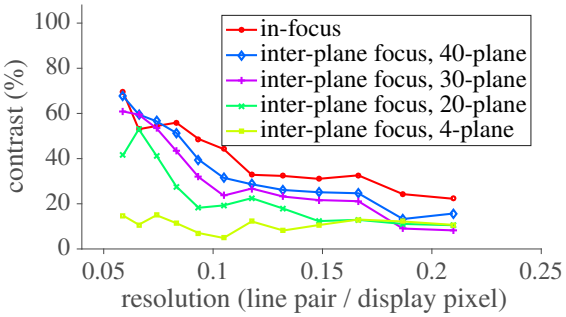
(c) Modulation transfer functions of (b)

Figure 6.11: **Simulation results of 4-plane and 40-plane multifocal displays with direct quantization, linear depth filtering, and optimization-based filtering.** The scene is at 0.02 diopters, which is an inter-plane location of the 4-plane display. (a) When the camera focuses at 0.02 diopters, the 40-plane display achieves higher spatial resolution than the 4-plane display, regardless of the depth filtering algorithm. (b) When the camera focuses at 0.9 diopters, the defocus blur on the 40-plane display closely follows that of the ground truth, whereas the 4-plane display fails to blur the low frequency contents. This can also be seen from the modulation transfer function plotted in (c).

play. The results captured by a camera with a 50 mm $f/1.4$ lens are shown in Figure 6.12. As can be seen, the higher number of focal planes (smaller focal-plane separation) results in higher spatial resolution at inter-plane locations.



(a) in-focus (b) 40-plane (c) 30-plane (d) 20-plane (e) 4-plane

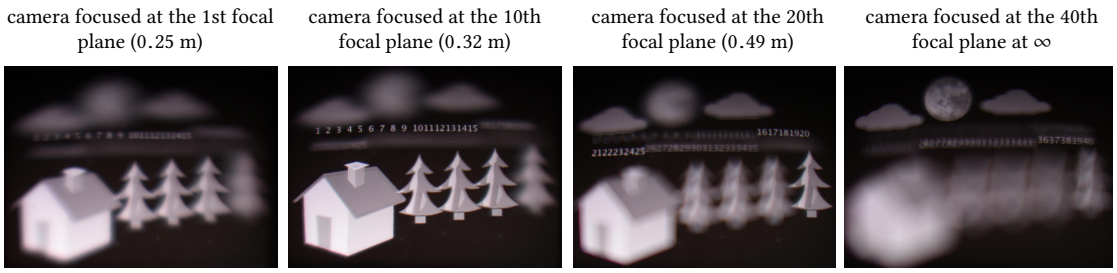


(f) modulation transfer function

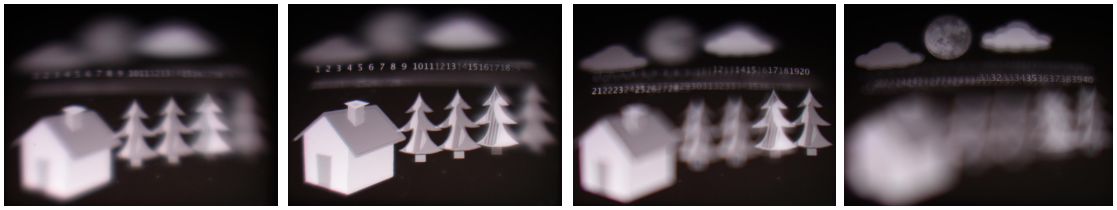
Figure 6.12: **Captured inter-plane focused images.** The resolution chart locates on the 5th focal plane of the 40-plane display. We emulate a 4-plane and a 20-plane display by putting their focal planes on the 5, 15, 25, 35th and on the odd focal planes of the 40-plane display, respectively. (a) Camera focuses at the 5th focal plane. (b,c) Cameras focus at the estimated inter-plane locations of the 40-plane display and the 30-plane displays, respectively. (d) Camera focuses at the 6th focal plane, an inter-plane location of a 20-plane display. (e) Camera focuses at the 10th focal plane, an inter-plane location of a 4-plane display. Their modulation transfer functions are plotted in (f). The images are captured with a 50 mm $f/1.4$ lens.

Next, we compare our prototype with a 4-plane multifocal display on a real scene. Note that we implement the 4-plane multifocal display with our 40-plane prototype by showing contents on the 10, 20, 30, 40th focal planes. The images captured by the camera are shown in Figure 6.13. For the 4-plane multifocal display, when used without linear depth filtering, virtual objects at multiple depths are focus/defocus as groups; when used with linear depth filtering, same objects appearing in two focal planes reduces the visibility and thereby lowers the resolution of the display. In comparison, the proposed method produces smooth focus/defocus cues across the range of depths, and the perceived images at inter-plane locations (*e.g.* 0.25 m) have higher spatial resolution than the 4-plane display.

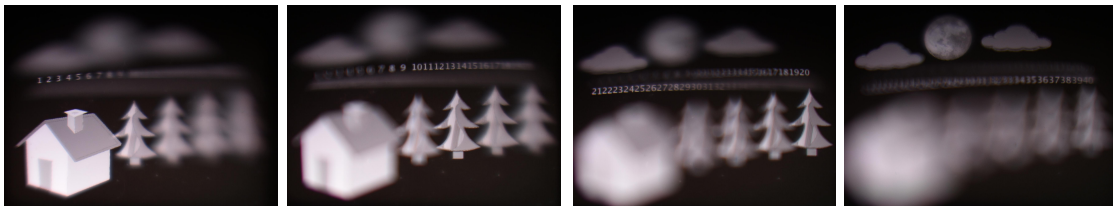
Finally, we render a more complex scene [eMirage] using Blender. From the rendered all-in-focus



(a) multifocal display with four focal planes and no depth filtering



(b) multifocal display with four focal planes and linear depth filtering



(c) proposed display with 40 focal planes and no depth filtering

Figure 6.13: **Comparison of a typical multifocal display with 4 focal planes and the proposed display with 40 focal planes.** The four focal planes of the multifocal display correspond to the 10th, 20th, 30th, and 40th focal plane. Images are captured with a 50 mm $f/1.4$ lens. Except for the first column, these focal planes are selected such that the 4-plane multifocal display (a) is in sharp focus. In the scene, the digits are at their indicated focal planes; the house is at the first focal plane; the trees from left to right are at 5, 10, 15, 20th focal planes; the clouds and the moon are at 30, 35, 40th, respectively.

image and its depth map, we perform linear filtering and display the results with the prototype. Focus stack images captured using a camera are shown in Figure 6.14. We observe realistic focus and defocus cues in the captured images.

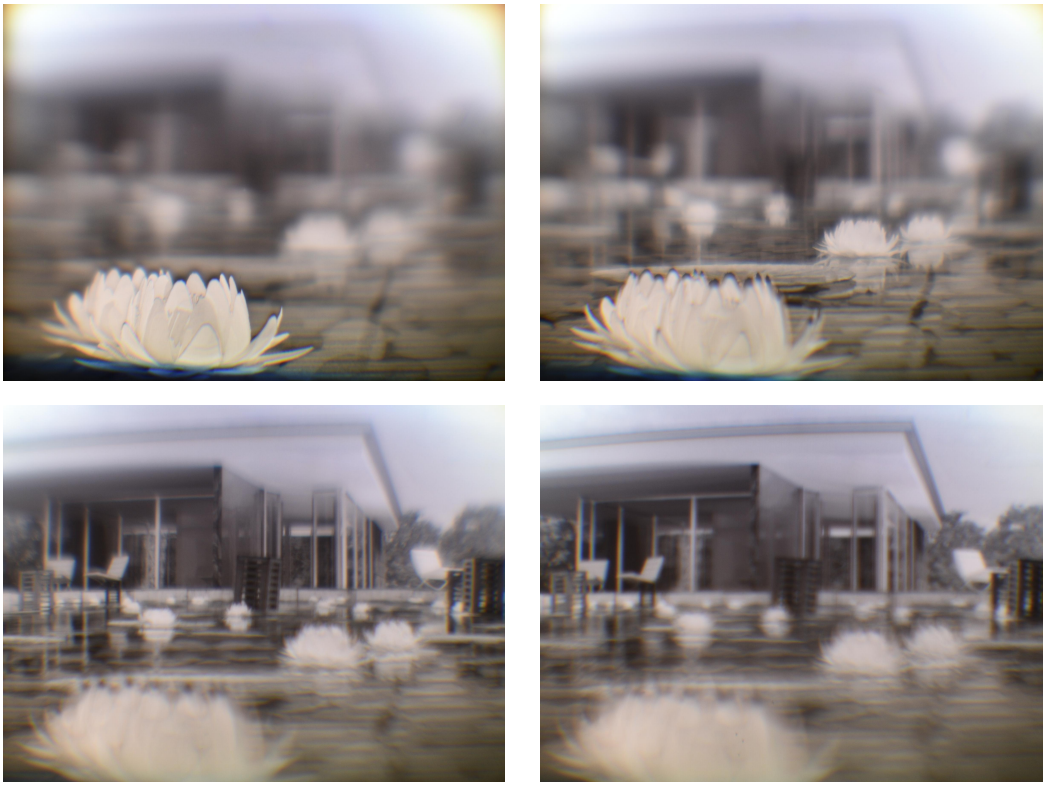


Figure 6.14: **Captured images with different focus settings of the camera.** From near (shown at the top left) to far (shown at the bottom right), the scene depth ranges from 50 cm (the flower at the bottom left) to infinity (the sky). The camera has a 50 mm $f/1.4$ lens. Three-dimensional scene courtesy eMirage.

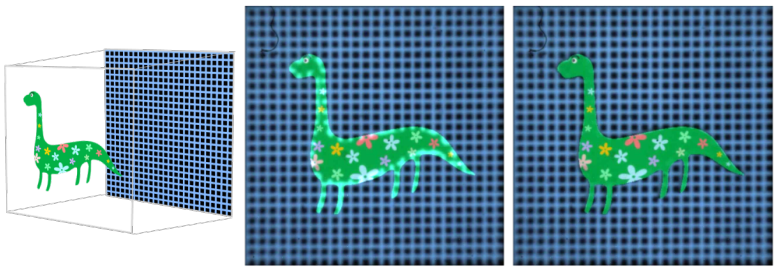
6.4 Conclusion

This chapter provides a simple but effective technique for displaying virtual scenes that are made of a dense collection of focal planes. Despite the bulk of our current prototype, the proposed tracking technique is fairly straightforward and extremely amenable to miniaturization. We believe that the system proposed in the chapter for high-speed tracking could spur innovation in not just virtual and augmented reality systems but also in traditional light field displays.

While the proposed multifocal display enables efficient and effective rendering of 3D worlds, as we have seen in the limitation, it does not render occlusion cues faithfully. The lack of occlusion cue has adversarial effects on the user experience of a virtual world, including deteriorated immersion and reduced contrast. In the next chapter, we will introduce how we are able to solve this problem.

7 Occlusion-Aware Multifocal Displays

Multifocal displays show three-dimensional (3D) content to a user by placing objects on different focal planes at different depths from the viewer. Having multifocal focal planes has a unique advantage that the display automatically renders the accommodation cues, *i.e.*, supports the focus of our eyes, provided there are a sufficient number of focal planes [Chang *et al.*, 2018, MacKenzie *et al.*, 2010, Rolland *et al.*, 1999, Watt *et al.*, 2012]. In order to display multiple focal planes at different depths, the focal planes are made transparent, often through time-multiplexing. However, as we have seen in the previous chapter, this transparency of focal planes has two adverse effects. First, the display is incapable of satisfying occlusion cues since even small displacements of the eye will readily produce overlapping contents. Second, the contrast of the display is significantly reduced. As illustrated in the example shown in Figure 7.1, when our eyes focus near, the content on far focal planes gets defocused. Since focal planes cannot block light from behind, the defocused far contents often bleed into near objects and reduce their contrast. Both effects are undesirable, in that, they reduce the immersive nature of the VR experience.



(a) Scene (b) Multifocal (c) Proposed

Figure 7.1: **Lack of occlusion cue and lowered contrast in multifocal displays.** This figure shows (a) the scene and the captured photos on (b) a typical multifocal display and (c) the proposed display when the camera/eye focuses on the dinosaur.

One potential approach for enabling occlusion cues and increasing contrast on a multifocal display is to enhance its angular resolution of the display pixels by replacing the display panel with a light-field display. The improved angular resolution allows us to control the intensity of the light rays that a pixel sends in different directions. The occlusion cue can then be produced by avoiding sending light through any virtual opaque object on front focal planes. In principle, this approach can generate photo-realistic occlusion cues. However, the additional angular resolution usually comes at the cost of loss in the spatial resolution [Huang *et al.*, 2015, Lanman and Luebke, 2013], which can be significant for precise handling of the occlusion cue.

This chapter provides a design for multifocal displays, capable of rendering occlusion cues, without any loss of spatial resolution. Our key idea is that to satisfy occlusion cues, for most scenes we do not need angular resolution in the physical display, but simply the ability to *tilt the light cone* emitted by display pixels. With appropriate tilts of the light cones, we can emulate the same effect as physical occlusion between real objects.

Figure 7.2 shows an example when we try to partially occlude a pixel on a far focal plane by a front occluder. Since the occluder is on the left, tilting the light cone emitted by the pixel to the right ensures that no light rays from the pixel pass through the occluder and thereby creates an illusion that the front occluder blocks light. More importantly, since the *entire* light cone is tilted, we do not need additional angular resolution on the display panel. As a result, no spatial resolution is traded for angular resolution.

To tilt the light cones emitted by display pixels, we place a phase-only spatial light modulator (phase

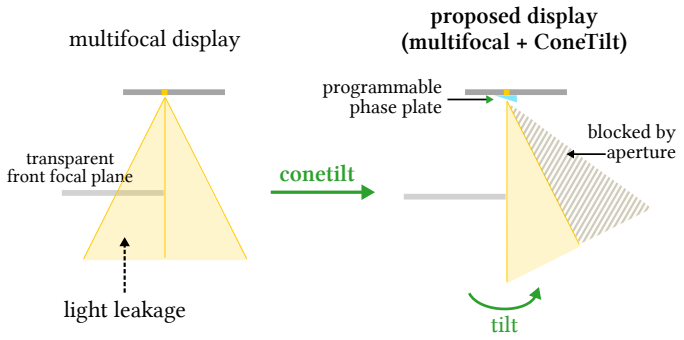


Figure 7.2: **Idea behind the ConeTilt operator.** (a) Since focal planes of multifocal displays are transparent, light from back focal planes easily leak through the front ones, causing deteriorated occlusion cues and reduced contrast. (b) By tilting the entire light cone to avoid the front occluder, the proposed display can prevent light leakage and thereby generate occlusion cues and retain contrast.

SLM) on the display panel. By programming the slope of the phase function at each display pixel, we can steer the light cone emitted by each pixel. The phase SLM acts as a *freeform field lens* that dynamically tilts each light cone based on the virtual scene.

What We Will Demonstrate in this Chapter

We make the following contributions in the chapter.

- *ConeTilt Multifocal Displays.* Our primary contribution is the use of the ConeTilt operation to endow occlusion cues in multifocal displays without loss of spatial resolution.
- *Implementation.* We provide a simple approach for implementing ConeTilt using phase SLMs. Given a virtual scene to be displayed, we derive the phase function to display on the SLM.
- *Design Space Analysis.* We derive important properties of the ConeTilt display including the fidelity of its occlusion cue, the contrast of the display, as well as the field-of-view and the size of the eye box.
- *Prototype.* We build a lab prototype using off-the-shelf components to characterize the improvements obtained in practice.

7.1 Prior Work

We briefly discuss the related research in producing occlusion cues in VR displays.

7.1.1 Role of Occlusion in Visual Perception

Among the numerous cues deployed by the human visual system to perceive the world, occlusion plays a dominant role [Cutting and Vishton, 1995, Geng, 2013]. When two opaque objects are at different depths, the object in front will occlude some light rays from the object behind. Moving our head and changing our perspective, even by small amount, will reveal parts of the back object that was originally hidden. The occluding and revealing of objects allows us to easily discover their relative depths even when the objects are close to each other. Further, when our eye focuses on objects at different depths, the subtle differences in the defocus blur at depth discontinuities are often sufficient to resolve their relative ordering [Zannoli *et al.*, 2016]. This makes occlusion one of the dominant cues for depth perception that works reliably across a wide depth range. As a consequence, it is of utmost importance that 3D displays, such as VR displays, generate occlusion cue properly.

7.1.2 Enabling Occlusion Cues in VR Displays

Most commercial VR displays generate occlusion cues by tracking the head/eye and regenerating content from the new perspective. This ensures that occlusion cues are faithfully produced and is only limited by the refresh rate of the display. However, as is often the case, the content is shown on a single plane and hence, there are gross accommodation errors. To alleviate the problem, gaze trackers are used to estimate users' gaze and pupil position, and the content is re-rendered based on the information. This increases both the hardware requirement and the computational cost. In the paper, we focus on enabling VR displays to simultaneously produce the accommodation cue and the occlusion cues created by subtle movements of our eyes, without the need of gaze tracking or content regeneration.

There are many display technologies that can produce occlusion cues without tracking. Cossairt *et al.* [2007a] and Jones *et al.* [2007] produce volumetric displays by rotating an anisotropic diffuser in synchrony with a projector. As the diffuser spins, the projector displays an image to be seen by a viewer in a specific direction. This results in realizing occlusion without knowing the position of the viewer. However, the spinning diffuser makes the displays more geared towards 3D televisions and not VR.

Light field displays [Huang *et al.*, 2014, Lanman and Luebke, 2013] provide angular control and, in principle, this is sufficient to produce rich occlusion cues. However, the gain in angular resolution is invariably accompanied by a loss in spatial resolution of the display. Further, the finite pixel pitch of the display greatly limits the depth range the displays can support, *i.e.*, only content whose depth is in the vicinity of the display depth can be faithfully rendered. While there are alternate implementations [Huang *et al.*, 2015, Wetzstein *et al.*, 2011] of light field displays that do not rely on microlens arrays, these do share the same challenges in obtaining a large depth range. In comparison, the depth range of multifocal displays is determined by the focus tunable lens and is often more than several diopters.

The importance of occlusion cues and methods to achieve it have been studied extensively in the context of augmented reality (AR) displays [Inami *et al.*, 2000, Kiyokawa *et al.*, 2000, Mulder, 2005]. However, these works concentrate on blocking light from real objects, wherein the challenges are different from those in VR displays.

7.2 ConeTilt Multifocal Displays

We start by studying the occlusion cues in the real world and what happens in its absence in a multifocal display. Subsequently, we introduce the concept of ConeTilt for producing occlusion cues.

7.2.1 Occlusion Cues in Real Scenes

Consider a scene consisting of two fronto-parallel planes, that are opaque and placed at different depths, as shown in Figure 7.3a. The front plane is red and the back plane is green; the camera/eye focuses on the front plane. Consider two points a and b, that are on either side of a depth discontinuity. At point a, all the light coming from the back plane is blocked, due to the opaqueness of the front plane. At point b, we get light from region $\overline{gh}$ on the back plane. Since the camera focuses on the front plane, light passing through point a and b will be collected by pixel A and B, respectively. Since no green light from the back object passes through a, pixel A is pure red.

7.2.2 Occlusion Cues in Multifocal Displays

Let us now consider the same scene, but rendered by a multifocal display. For simplicity, we will assume that the two planes are displayed on focal planes corresponding to their true depth. As with most multifocal designs, the focal planes are transparent, and as a result, light from the back focal plane can leak through the content shown on the front focal plane. In Figure 7.3b, pixel A receives not only light emitted by point a but also all light from $\overline{pq}$ passing through a, making A a yellow pixel (instead of red).

The light leakage has two consequences.

- *Loss of occlusion cue.* When two focal planes are in the depth of field of our eye, their contents will overlap even when we want to display an opaque front object.
- *Reduced contrast ratio.* When we focus on the front plane (and the back focal plane is defocused), the front focal plane will be overlaid with the blurred content from behind and thereby lose its contrast. The low contrast makes displaying dark objects on the front focal plane very difficult.

Removing occluded contents on the back focal plane cannot solve the leakage problem entirely. In Figure 7.3c, we remove the region behind the front object given the position of the eye; however, since each display pixel emits light toward a wide range of angles, light from the region $\overline{pq}$ still leaks through point a and reduces the contrast of pixel A. Removing occluded contents has another side effect — it decreases the intensity of defocused content near depth discontinuities. Let us use point b as an example. In reality, point b receives light from region $\overline{gh}$. Since we remove occluded region $\overline{gp'}$, we reduce the amount of light passing through point b and thereby make pixel B dimmer than the reality.

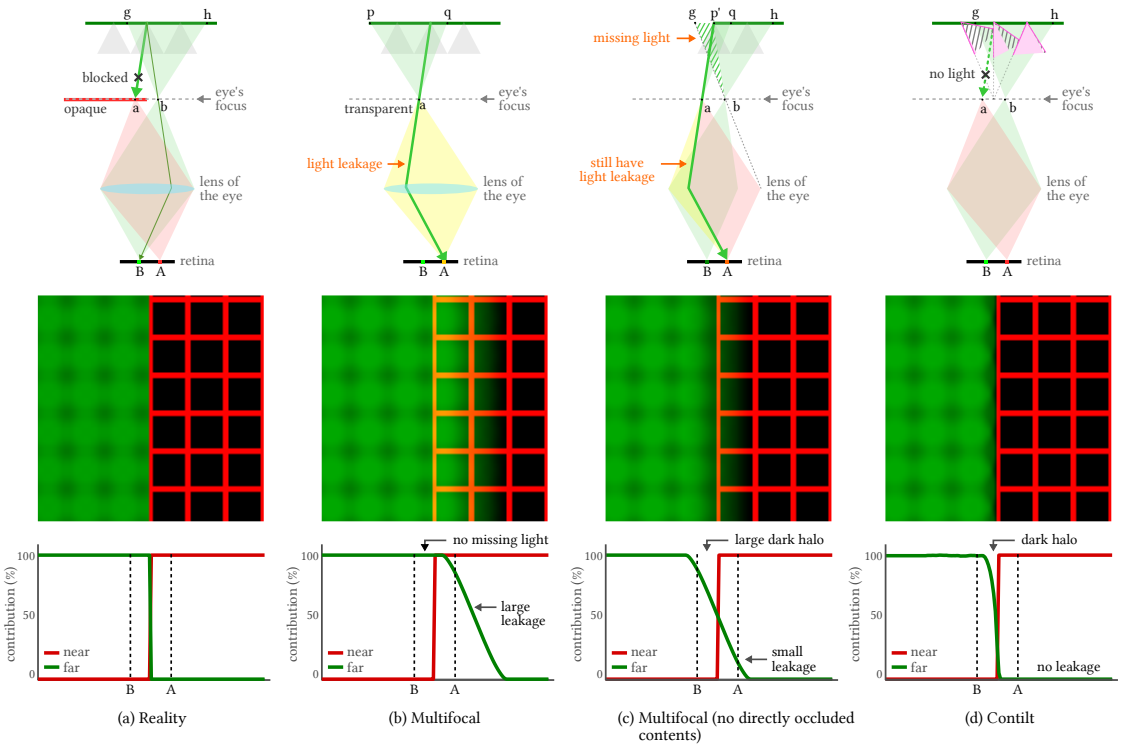


Figure 7.3: **The concept of ConeTilt.** We consider a scene consisting of two planes at different depths and show the image formation in (a) the real world, (b, c) multifocal displays with/without showing the overlapped part of the back plane, as well as (d) the proposed ConeTilt displays. In each case, the middle row shows a rendered image obtained when a camera/eye is focused on the front plane, and the bottom row shows the contribution from the front and the back planes (assuming all pixel values equal to 1). (a) In the real world, the front plane blocks the light from the back plane, and thus we see a sharp edge with no light from the back plane leaking onto the front. (b, c) In a multifocal display, the inherent transparent nature of focal planes leads to light leakage from the back focal plane. The light leakage cannot be prevented even when we remove the overlapped region from the content shown on the back plane. (d) In a ConeTilt display, the light cones are tilted to avoid emitting light rays that intersect with the content on the front focal plane, and thereby the display produces occlusion cues similar to those found in the real world. No light from the back plane leaks to the front plane, even when we do not remove the overlapping contents. Note that some light is missing from the back layer. We will explain the phenomenun in detail in Section 7.3.6.

7.2.3 Enabling Occlusion Cues via ConeTilt

The proposed display aims to produce occlusion cues on multifocal displays via a simple operation, that we refer to as *ConeTilt*. We discuss the basic idea of ConeTilt here and defer the details of its actual implementation and limitations to Section 7.3.

The ConeTilt operator enhances a multifocal display in the following way — *it allows for the cone of light emanated at a display pixel to be independently tilted.* That is, we endow the multifocal display with the freedom to independently tilt the cone that it emits at each pixel and at each focal plane. As we will describe next, for a large class of scenes, this operation is sufficient to produce occlusion cues as well as reduce the loss of contrast due to light leakage across focal planes.

Reducing Light Leakage

We consider the same scenario of a scene with two planes rendered by a multifocal display. However, on the back focal plane, we apply the ConeTilt operation at pixels near the occluding edge. For each pixel, we tilt the cone such that no emitted light ray intersects with the content shown on the front plane. As is to be expected, the resulting tilt is different across locations. Pixels that are occluded by the front focal plane need to be tilted the most, and the amount of tilt gradually reduces when a pixel moves away from the occluding edge, as shown in Figure 7.3d.

Despite its simplicity, ConeTilt effectively reduces light leakage across focal planes. Even though point *a* is transparent, ConeTilt ensures that no pixel on the back focal plane emits light toward point *a*, and thereby, we cannot see the far plane when we look at the front object. This effectively creates an illusion that the front object blocks light. In addition, contrast is preserved as no light leaks in the front object. Note that, since entire light cones are tilted, ConeTilt does not require additional angular resolution and in principle can have the same spatial resolution as a typical multifocal display.

7.3 Design of ConeTilt Displays

In this section, we describe an optical schematic to implement the ConeTilt operator and, subsequently, analyze the design and characterize the properties and limitations of a ConeTilt display.

7.3.1 Optical Schematic

The ConeTilt operation is implemented by optically attaching a phase SLM to the display panel, which is a digital micromirror device (DMD) in our prototype. Due to the reflective nature of our phase SLM,

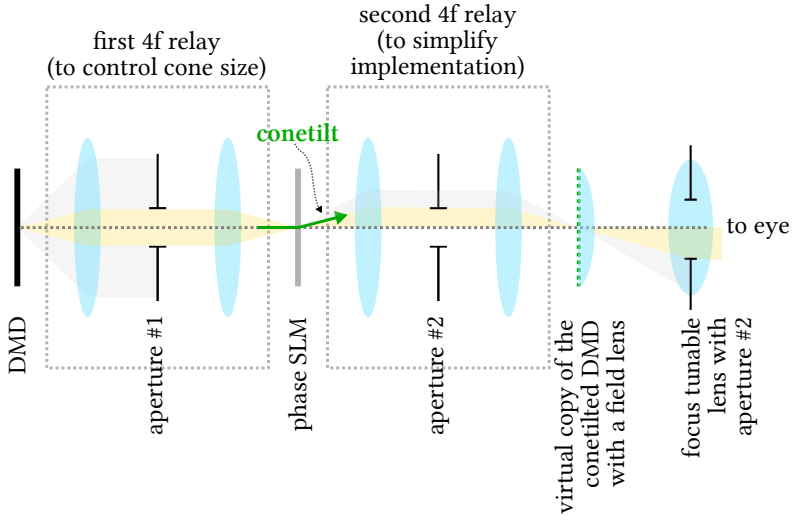


Figure 7.4: **Schematic of a ConeTilt display.** We implement the ConeTilt operation by optically colocating a phase SLM with a display panel (DMD). This is achieved by mapping the physical display onto the phase SLM using a 1:1 $4f$ relay. The phase SLM implements the ConeTilt operator. Subsequently, a second $4f$ relay is used to map the phase SLM onto the image plane of the focus tunable lens.

we cannot physically attach it on the DMD, and thereby we use $4f$ relays to optically attach the DMD to the phase SLM.

This optical setup is illustrated in Figure 7.4, which is composed of the DMD, the phase SLM, two one-to-one $4f$ relays, and a focus tunable lens that serves as the main lens of the multifocal display. The first $4f$ relay optically colocates the DMD and the phase SLM, and the second relay is used to provide additional room for calibration cameras (please see Section 7.4 for details). Conceptually, as the phase SLM is directly placed on the DMD, it serves as a free-form field lens and only controls the direction of the light from the DMD without introducing any magnification that will reduce the spatial resolution of the display. The aperture of the first $4f$ relay ensures a fixed angular cone arriving the SLM from all DMD pixels. Note that we need to crop any tilted light ray whose direction exceeds the angular range of the original light cone, otherwise any tilted pixel will appear to have a larger cone and seems brighter. To crop any light that exceeds the original angular range because of ConeTilt, we can place an aperture in the second $4f$ relay or on the focus tunable lens.

7.3.2 Use of Phase SLMs for ConeTilt Operations

Let us talk about how we implement ConeTilt with phase SLM. Intuitively, if we wish to tilt a cone of light, the simplest approach is to use a prism, which is a phase ramp (at least for monochromatic light). Hence, the ConeTilt operation at a given pixel is achieved by choosing an appropriate phase gradient that effectively acts as a local prism to steer the light. The phase gradient is determined by the direction and magnitude of the tilt, which in turn is determined by the occluding objects.

7.3.3 Deriving the Direction and Magnitude of the Cone Tilt

The parameters of the tilt, namely its direction and magnitude, are derived independently for each pixel on each focal plane. Our strategy for determining these parameters is illustrated in Figure 7.5a.

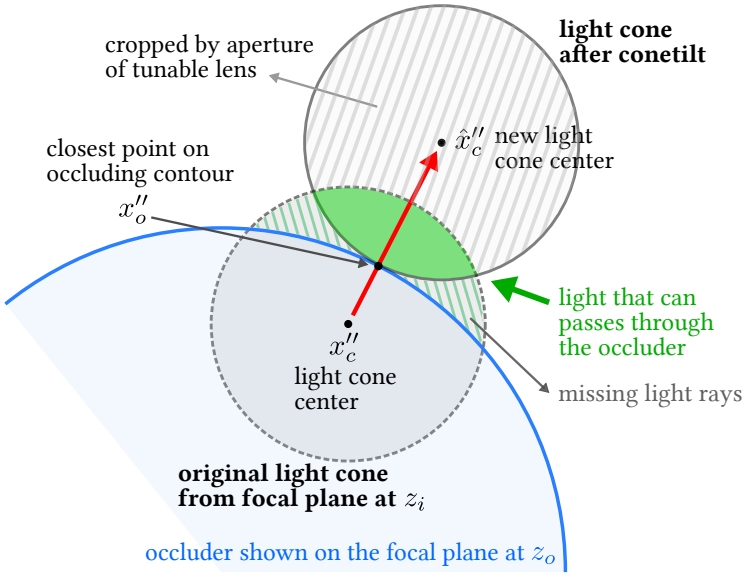
Suppose that a light cone is occluded (partially) by a virtual object on a front focal plane. The goal of ConeTilt is to ensure that no light rays in the light cone intersects with the occluder. Let the center of the light cone on the front focal plane be x_c''. We first identify the point x_o'' on the occluding contour that is closest to x_c''. Then we steer x_c'' towards (or away from) x_o'' such that the tilted light cone just touches the occluding contour. As can be seen from Figure 7.5a, using ConeTilt enables the display to approximate the occlusion caused be the virtual object. Since ConeTilt do not increase the angular resolution of the display panel, there are some missing light rays. We will discuss the limitation in more detail in Section 7.3.6.

Next, we proceed to derive the analytical expressions of the position of the light cone, for which we need to model the effect of the focus tunable lens.

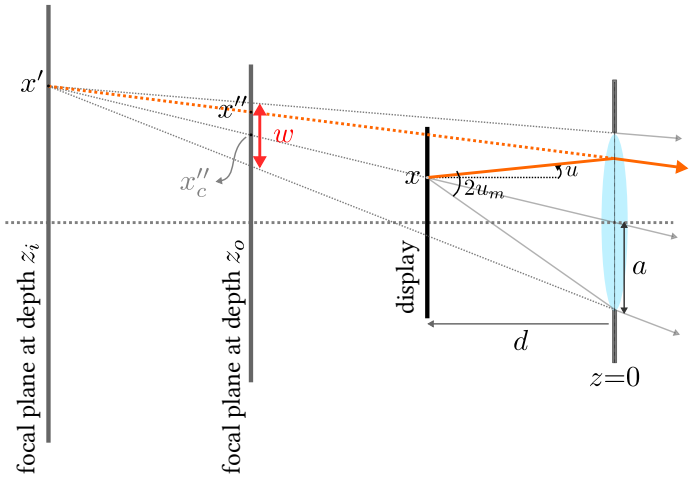
Image Formation in ConeTilt Displays.

Notation. Our notation is shown in Figure 7.5b. Consider a multifocal display which is composed of a focus-tunable lens and a display panel. The display panel is parallel to the tunable lens, and the distance between them is d. For simplicity, we assume small-angle (paraxial) scenarios. The radius of the aperture of the tunable lens is a, and the radius of the light cone u_m is set to $\frac{a}{d}$ by controlling the aperture of the first $4f$ relay. We use the prime symbol ($'$) and the double prime symbol ($''$) to denote positions on a far focal plane and a near focal plane, respectively.

Avoiding Vignetting with Default tilt. Let us look at the scenario when no tilt is applied, as shown in Figure 7.6a. Without any tilt, the light cone from each display pixel travels straight, and part of the cone will be blocked by the aperture of the tunable lens. This wastes energy and causes vignetting.

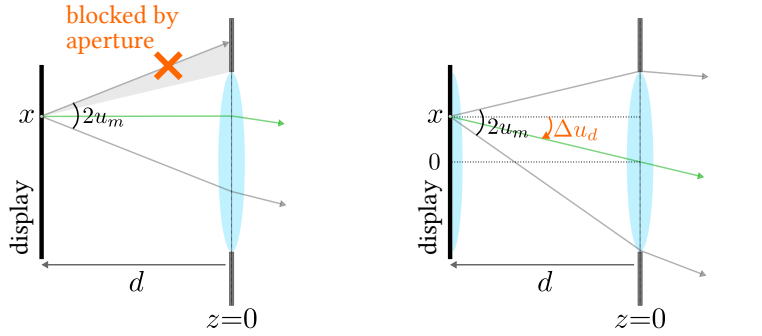


(a) View on the focal plane at depth z_o



(b) Ray diagram

Figure 7.5: **Determining ConeTilt parameters.** (a) shows the intersection of the light cone on the focal plane at depth z_o where the occluder (blue region) locates. The ConeTilt operator simply shifts the light cone to a position where it does not overlap with the occluder and is closest to the original location. Due to the aperture of the tunable lens, the slashed gray regions on the tilted light cone is cropped. Therefore, only the light in the solid green region is let through. Note that the slashed green regions represent the light that cannot be rendered by the display (see Section 7.3.6 for details). (b) shows the ray diagram and the notation used in the chapter.



(a) no default tilt

(b) adding default tilt $\Delta u_d = \frac{-x}{d}$ with a field lens to alleviate vignetting

Figure 7.6: **Avoiding vignetting with default tilts.** (a) Without the default tilt, the light cone travels perpendicular to the display panel. Therefore, the light cones of all off-center pixels will be cut by the aperture, causing light loss and vignetting. (b) By adding default tilt to each pixel (through a physical field lens or the phase SLM), the entire light cones can enter the aperture. This significantly reduces the vignetting of the multifocal display.

To avoid vignetting, we apply a default tilt so that the entire light cone enters the aperture without being blocked. In other words, the default tilt directs the chief ray of the light cone towards the center of the tunable lens, as shown in Figure 7.6b. We denote the default tilt by Δu_d, and its value can be calculated by

$$\Delta u_d = \frac{-x}{d}. \tag{7.1}$$

We implement the default tilt by attaching to the relayed display panel a physical convex field lens with focal length equal to d, but it can also be implemented by the phase SLM. In the later scenario, the phase SLM is responsible for implementing both the default tilt and the tilt required by ConeTilt. Note that as we will discuss in Section 7.3.6, implementing the default convex field lens on the phase SLM will confine the field-of-view of display, due to the limited capability of the phase SLM.

Ray Tracing. For simplicity, let us first consider a two-dimensional flatland. When the focal-length of the tunable lens is f_i, the multifocal display creates a focal plane at depth $z_i > 0$, where

$$\frac{1}{d} + \frac{1}{-z_i} = \frac{1}{f_i}. \tag{7.2}$$

The pixel x on the DMD forms a virtual pixel x' on the focal plane at depth z_i, where

$$x' = \frac{z_i}{d} x. \tag{7.3}$$

This means that after bent by the tunable lens, the light ray $(x, u + \Delta u_d + \Delta u)$ will intersect the focal plane at depth z_i on x' with angle u', where $u \in [-u_m, u_m]$ is the direction of the light ray, Δu_d is the default tilt, and Δu is the tilt introduced by ConeTilt. Given the focal length of the tunable lens f_i and Equation (7.1), we can calculate u' by simple ray tracing:

$$u' = \frac{x + (u + \Delta u_d + \Delta u)d - x'}{z_i} = -\frac{x}{d} + \frac{d}{z_i}(u + \Delta u). \tag{7.4}$$

We are interested in the intersection of the light cone on a front focal plane at depth $z_o < z_i$ where an occluder lies on. Let the intersection of the light ray (x', u') on the front focal plane be x''. By ray tracing, we have

$$x'' = x' + u'(z_i - z_o) = \frac{z_o}{d}x + dz_o\left(\frac{1}{z_o} - \frac{1}{z_i}\right)(u + \Delta u)\,. \tag{7.5}$$

From Equation (7.5), we can see that x'' is an affine function of u and Δu. This means that the light cone $\{(x, u) \mid u \in [-u_m, u_m]\}$ intersects continuously on the front focal plane, and if we tilt the light cone by Δu, the region simply shifts $\left(z_o^{-1} - z_i^{-1}\right) z_o\, d\Delta u$. Specifically, the intersection of the light cone on the front focal plane can be expressed by

$$k_i(x'') = k\left(\frac{x'' - \hat{x}_c''}{w}\right) k\left(\frac{x'' - x_c''}{w}\right), \tag{7.6}$$

where $k(x)$ is the aperture function which we assume to be a box function in the flatland, whose value is 1 for $|x| \le \frac{1}{2}$ and zero otherwise. In the equation, the first $k(\cdot)$ corresponds to the tilted light cone, and the second $k(\cdot)$ corresponds to the cropping caused by the aperture. The term $x_c'' = \frac{z_o}{d}x$ is the center of the light cone ($u = 0$) on the focal plane before the tilt; whereas $\hat{x}_c''$ is the center after the tilt, and

$$\hat{x}_c'' = \frac{z_o}{d}x + dz_o\left(\frac{1}{z_o} - \frac{1}{z_i}\right)\Delta u. \tag{7.7}$$

The term w is the diameter of the light cone:

$$w = 2du_m z_o\left(\frac{1}{z_o} - \frac{1}{z_i}\right), \tag{7.8}$$

which is independent to Δu_d and Δu.

Avoiding Occluders Equation (7.6) enables us to identify if a pixel on the focal plane z_o occludes a light cone. In particular, for a pixel x'' on the focal plane, $k_i(x'') \neq 0$ implies that the light cone is occluded and that we need to tilt the cone. Once we identify the closest point x_o'' on the occluding contour, we simply tilt the light cone such that the trailing edge of the cone is incident on x_o''. The exact expression of the tilt can be easily derived and is omitted here. Note that for scenes with compact objects,

the process can be greatly speeded up using a simple heuristic that most pixels remain unoccluded and only pixels near depth discontinuities need to perform ConeTilt.

In a 3D world, we apply the same principle, except that the points x, x', x'', x_c'', x_o'' are all 2D coordinates on the respective planes.

Examples. We showcase some examples in Figure 7.7 on scenes that are composed of two planes at different depths. Given a scene, we find the minimum tilt for each pixel on the back plane to avoid front objects. The input to the ConeTilt display is simply the content and the tilting vectors for both the front and the back plane. As can be seen from the figure, ConeTilt effectively avoids the light leakage from the back plane. Note that while ConeTilt performs well on simple occluding contours like the vertical edges and the smooth curve, it creates dark halo artifacts at the corner. The limitation will be discussed in details in Section 7.3.6.

7.3.4 Deriving the Phase Function

Having derived the desired tilt for each pixel, we now turn to derive the phase function to show on the phase SLM so that the tilts can be realized. We first derive the phase function without any restrictions on the phase SLM.

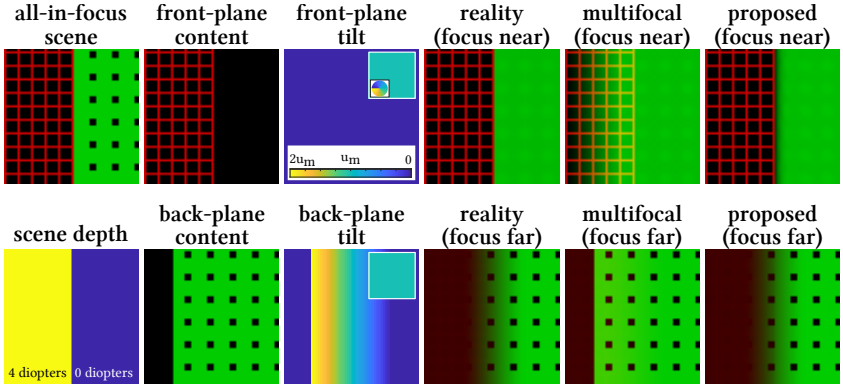
Let the phase function of the phase SLM be $\phi(x)$, where $x \in \mathbb{R}^2$, and the wave number be $k = \frac{2\pi}{\lambda}$, where λ is the wavelength of the emitted light, which is assumed to be monochromatic or narrowband. When a light ray reaches the phase SLM at x with direction $u \in \mathbb{R}^2$, the phase funtion delays the wavefront of the light and causes the light ray to change direction. Assuming all angles are small, the outgoing direction u_o can be calculated by

$$u_o = u + \frac{1}{k}\nabla\phi(x), \text{ or } \Delta u = \frac{1}{k}\nabla\phi(x). \tag{7.9}$$

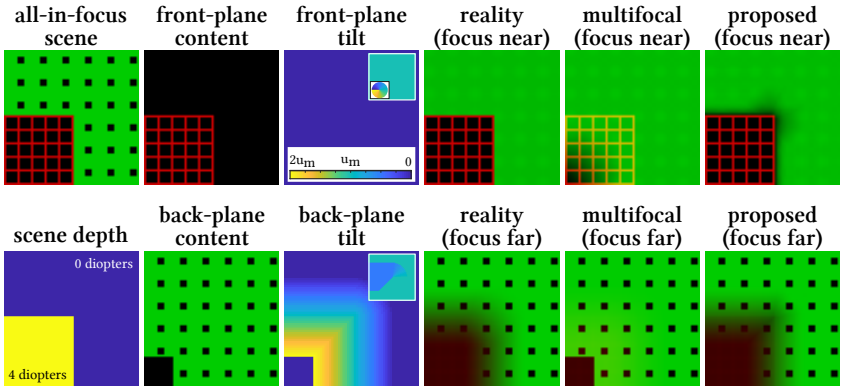
Thereby, our goal is to find a phase function that satisfies $\frac{1}{k}\nabla\phi(x) = \Delta u_t(x)$, where $\Delta u_t(x)$ is the desired tilt of the display pixel at x.

We find the phase function by solving a Poisson optimization problem. Let $\Delta\mathbf{u}_t^x \in \mathbb{R}^{n_x \times \mathbf{n}_y}$ and $\Delta\mathbf{u}_t^y \in \mathbb{R}^{n_x \times \mathbf{n}_y}$ be the vectorized target tilts at the center locations of all display pixels, where n_x and n_y are the number of pixels in the x and y direction, respectively. Let $\boldsymbol{\phi} \in \mathbb{R}^{(n_x+1)\times(\mathbf{n}_y+1)}$ be the discretized phase function that we try to find. We solve the following optimization problem

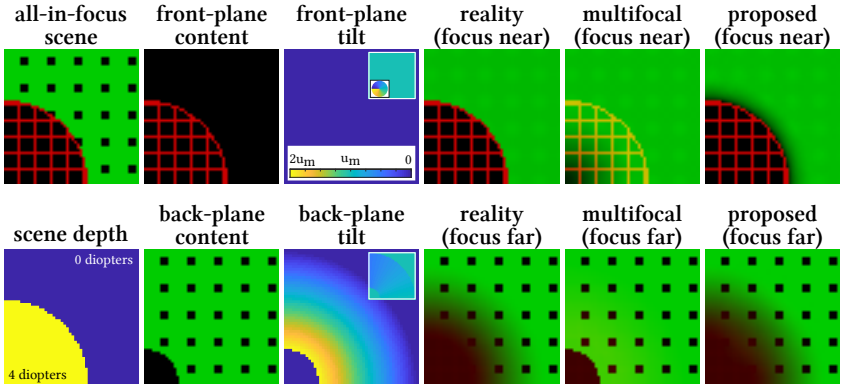
$$\min_{\boldsymbol{\phi}} \ \|D_x\boldsymbol{\phi} - \Delta\mathbf{u}_t^x\|^2 + ||D_y\boldsymbol{\phi} - \Delta\mathbf{u}_t^y||^2 + \epsilon||\boldsymbol{\phi}||^2, \tag{7.10}$$



(a) Vertical edge 1 (two planes separated by 4 diopters)



(c) Corner



(d) Curve

Figure 7.7: **ConeTilt examples.** This figure shows four example scenes, the content shown on the focal planes, the tilt vectors shown with the front and the back plane, the rendered scenes in reality, and the rendered results on a typical multifocal display and our ConeTilt display with the same parameters as our prototype. Note that we plot the tilting vectors in length and direction (insets). We can clearly see the loss of occlusion cue and light leakage in the typical multifocal display. The proposed display successful prevents light leakage and creates occlusion. However, it produces dark halo around the occluding contour.

where D_x and D_y represent taking derivative along x and y, respectively, and ϵ is a small constant used to control the smoothness of the phase function.

Incorporating Phase SLM Constraints

Due to the discretization, the phase functions that can be displayed on a phase SLM is limited by the Nyquist sampling theorem. Besides, since most phase SLMs can only delay the phase up to 2π for visible light, phase wrapping will create the phase aliasing artifacts [Spagnolini, 1993]. To avoid phase aliasing effect, we can only show phase functions that do not have high-frequency variations. Specifically, the maximum phase difference between two neighboring SLM pixels cannot be more than π. In other words, we require

$$\left|\frac{\mathrm{d}\phi(x)}{\mathrm{d}x}\right| \leq \frac{\pi}{\delta_x}, \tag{7.11}$$

where δ_x is the pixel pitch of the SLM pixels along the x direction. The same constraint applies to the y direction. The constraint (7.11) limits the maximum angle that we can shift the light cones using the phase SLM. From Figure 7.5a we can see that given the radius of a light cone u_m, the maximum amount of tilt is less than or equal to $2u_m$. Therefore, Equation (7.11) sets an upper bound of the radius of the light cone:

$$u_m \leq \frac{\pi}{2k\delta_x}. \tag{7.12}$$

By physically constraining the size of the light cone to satisfy Equation (7.12), the constraint (7.11) is automatically satisfied.

Our phase SLM has a pixel pitch $\delta_x = 6.4$ um; when $\lambda = 520$ nm, the radius u_m is upper-bounded by 1.2 degrees. The limited ability of the phase SLM to tilt light constrains the size of the light cone we can use, as we will see in Section 7.3.5, it also limits the size of the aperture, the field-of-view, and the eye box of the display. Nevertheless, these limitations can be greatly alleviated if we switch to a more powerful phase SLM.

7.3.5 Design Criteria and Analysis

We now analyze the properties of ConeTilt displays and the system configurations required to achieve them. For simplicity, all the analyses use the paraxial assumption and assume the eyes are close to the tunable lens.

Field-of-View

Field-of-view of a typical multifocal display depends on the size of the display panel and the distance d, when the eye is close to the tunable lens. When the default tilt is implemented by the phase SLM, the field-of-view is constrained, due to the limited tilt supported by the phase SLM. From Equation (7.1), the default tilt $\Delta u_d = \frac{-x}{d}$ for the pixel x on the DMD, and based on Equation (7.11), we have

$$\left|\frac{-x}{d}\right| \leq \frac{\pi}{k\delta_x}, \text{ or } |x| \leq \frac{\pi d}{k\delta_x}. \tag{7.13}$$

Therefore, when implementing the default tilt with the phase SLM

$$\text{field-of-view} \leq \max\left|\frac{2x}{d}\right| = \frac{2\pi}{k\delta_x} = \frac{\lambda}{\delta_x}. \tag{7.14}$$

For example, given $\lambda = 520$ nm, $\delta_x = 6.4$ um, $d = 58$ mm, and a DMD with 13.6-um pixel pitch, we have a field-of-view of 4.7 degrees, or 346 DMD pixels.

Our prototype implements the default tilt with a physical field lens and is capable of displaying content on the entire display panel without being constrained by the phase SLM. Thereby the field-of-view of our prototype is the same as a typical multifocal display.

Eye Box

Most multifocal displays have small eye boxes, due to the lack of occlusion cues (which causes virtual objects to overlap when the eye shifts). As a consequence, even though in principle multifocal displays do not require gaze tracking to provide accommodation cues, most implementations use gaze trackers are re-rendered the scene according to the location of the eye and the direction of the gaze [Mercier *et al.*, 2017].

In a ConeTilt display, eyes can move freely inside the aperture of the tunable lens without causing overlapping contents. This extends the effective eye box to the entire aperture without the help of a gaze tracker or re-rendering. In our prototype, the aperture size is only limited by the ability of the phase SLM and is equal to $u_m d = 2.1$ mm in diameter.

Contrast

With the ability to prevent light leakage, ConeTilt displays preserves the contrast of focal planes. Figure 7.3(b,d) compare the contrast when we display the same content on the focal planes on a typical multifocal display and on a ConeTilt display. As can be seen in the third row, ConeTilt not only significantly reduces the contribution from the back focal plane to the front focal plane, it also makes the

transition much sharper. Similar trends can be observed in Figure 7.7. These results demonstrate the ability of ConeTilt to preserve contrast.

7.3.6 Limitations

We provide a detailed understanding of the key limitations of ConeTilt displays and potential ways to mitigate them.

Inability to Handle Complex Occlusion Patterns

ConeTilt displays tilt entire light cones to mimic the effect of occlusion. While avoiding the loss of spatial resolution, this idea does not extend beyond simple occlusion scenarios where the occluding contours are smooth and well separated. For example, if the front focal plane has two occluding contours in close proximity, then ConeTilt would be insufficient to produce the occlusion cue. For such a scenario, we will need to "trim" the light cone, an operation that is beyond the simple tilt operation that we implement in this paper.

Complexity of the Occluding Contours. The minimum distance between two occluding edges on a focal plane is the size of the light cone on the front focal plane. From Equation (7.8), we have

$$\text{min distance} = \frac{2d^2 u_m}{\delta} \left| \frac{1}{z_o} - \frac{1}{z_i} \right| \text{ display pixels,} \tag{7.15}$$

where z_o and z_i is the depth of the focal planes, and δ is the pixel pitch of the display pixels. On our prototype, when the front and the back focal planes are separated by 4 diopters, the minimum distance between two occluders on the front focal plane can be 36 pixels. Note that Equation (7.15) decreases quadratically in d, whereas the eye box only decreases linearly in d. This provides an advantageous trade-off between the minimum distance and the size of the eye box. Specifically, we can make the occluding edges much closer if we are willing to slightly reduce the size of the eye box.

Dark Halo Near Occluding Edges

In typical multifocal displays, the shape of the defocus blur kernel is determined by the aperture of the tunable lens. In a ConeTilt display, the shape of the defocus blur kernel is determined by the tilts and is the intersection of the tilted light cone and the aperture of the tunable lens, as illustrated in Figure 7.5a and Figure 7.10. This functionality enables ConeTilt displays to avoid light leakage. However, a one-parameter tilt cannot produce photo-realistic defocus blur kernel. In the example shown in the

Figure 7.5a, in reality, the occluder will allows all the light in the crescentic region to pass. In contrast, a ConeTilt display can only render the light in the green region, and as a result, some light rays are missing in the virtual scene. The main effect of missing some light rays is that the defocused objects near the occluding boundaries are dimmer compared to the reality.

Other Limitations Due to Phase SLMs

In addition to the limited capability to tilt light, using a phase SLM induce the following limitations on a ConeTilt display.

Chromatic Aberration. Since the phase of the light depends on its wavelength, the phase function is color-dependent. To create a typical RGB display, we can use time-multiplexing and show each of the phase functions designed for each color sequentially. To alleviate the chromatic aberration caused by polychromatic light, the phase functions need to be smooth. Thereby, in the optimization problem (7.10) we use the ℓ_2-regularization to find a smooth solution. Nevertheless, since the phase SLM is attached to the display panel, the chromatic aberrations will only appear in the defocused regions, *i.e.*, on an out-of-focus content that has been tilted.

Ghosting Artifacts. Typical phase SLMs quantize the phase values to 8-bit, and the quantization will create ghosting artifacts [Laude, 1998], which reduces the contrast of tilted light cones. In our experience, it is usually more severe when the amount of tilt is large.

Phase Wrapping Artifacts. Since most phase SLMs can only achieve a phase delay of 2π, the phase function will be wrapped multiple times across the entire display. Due to the dramatic change in phase values, the wrapping creates dark seams in the images we see. While using smooth phase functions helps alleviate the problem, in our experience, the most effective solution is to rapidly change the global phase offset within the exposure time of a frame. Changing the offset shifts the dark seams without affecting the content, and thereby is effective in smoothing the dark seams. The functionality can be implemented by displaying the phase functions with different offsets rapidly on the phase SLM or with a single-cell liquid crystal rotator that can changes the phase offset globally. In our prototype, for the sake of simplicity, we display phase functions with different offsets within one camera exposure.

Refresh Rate. In an ideal scenario, each focal plane should be tilted individually with its own ConeTilt configurations. However, typical phase SLMs have a refresh rate of less than 200 Hz and thereby limits

the number of phase functions we can display within a frame. For example, if the display runs in 60 frames per second, we can at most use 3 phase functions within each 3D frame.

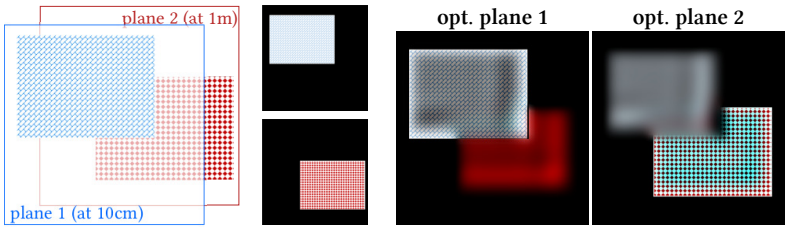
To account for the limited refresh rate of a phase SLM, in our experiments, we only calculate two phase functions — one for all the pixels in scene that can be seen by a centered eye and the other for the occluded contents directly behind occluders. In other words, we compute a phase function for contents that we can see directly, and we compute another phase function for all contents directly occluded by the objects shown in the first image. When displaying the 3D scene, we display the image-phase function pairs sequentially within one frame. This avoids the need to use a phase SLM with very high refresh rate and in our experience, it can still effectively reproduce the occlusion cues for most scenes.

Finally, we note that the ConeTilt operator need not be implemented on phase SLMs. We can use other technologies that can steer light locally, like the micro-prism proposed by Smith *et al.* [2006], which enables $\pm 7^\circ$ tilts. This can improve the size of the eye box of the display significantly.

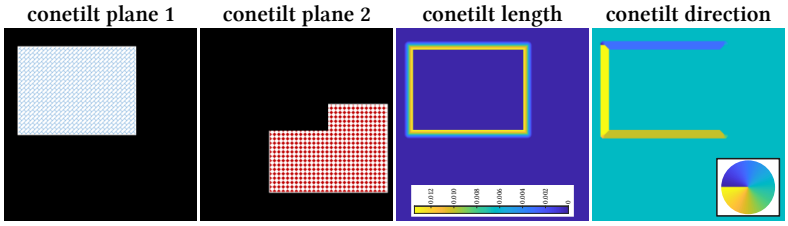
7.3.7 Comparison to Optimization-based Filtering

Narain *et al.* [2015] show that the dark and bright halos at depth discontinuities can be alleviated by optimizing the content to show on the focal planes, under the objective of producing the desired image at each focus setting of the eye. However, this optimization-based filtering approach aims to satisfy only the focus cue of the eye. The algorithm often renders an object at a single depth on multiple focal planes. As a consequence, the method requires precise placement of the eye with respect to the display, and small motions of the eye can lead to inconsistent motion parallax and occlusion cues unless the content is regenerated, which often requires a precise eye and head tracking system [Mercier *et al.*, 2017]. In the simulation results shown in Figure 7.8, even though the optimization-based filtering successfully reproduces the scene when the eye is centered, the quality of the results deteriorates dramatically when we change the viewpoint slightly. In comparison, the proposed display shows the contents at their original focal planes, and thereby its performance is more robust to the change of viewpoints, or other factors that can vary easily like the pupil diameter. Recently, Choi *et al.* [2019] propose to solve the problem by optimizing for small movements of the eye instead of different focus settings. However, the method is tied to specific multifocal displays and has high computational cost.

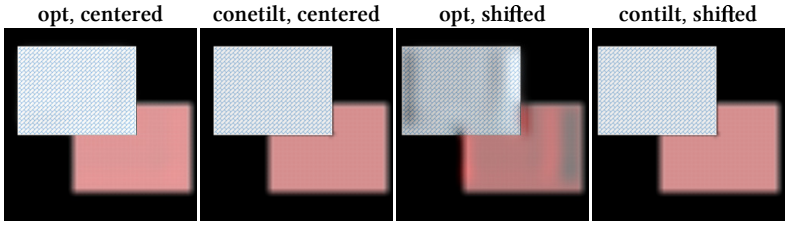
The proposed ConeTilt operator provides an alternate approach to achieve the goals of the optimization-based filtering. Instead of modifying the content, we concentrate on modifying the display hardware, and as we will demonstrate by real captured results in Section 7.5, the proposed method effectively and efficiently prevents light leakage, generates occlusion cues, and increases contrast. In principle, the



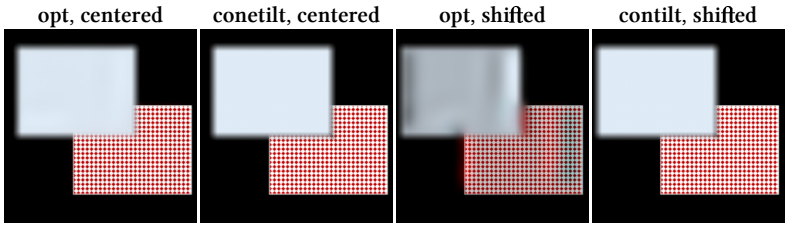
(a) Scene (b) Inputs from opt. filtering



(c) Inputs for the proposed ConeTilt display



(d) Camera focuses on front focal plane



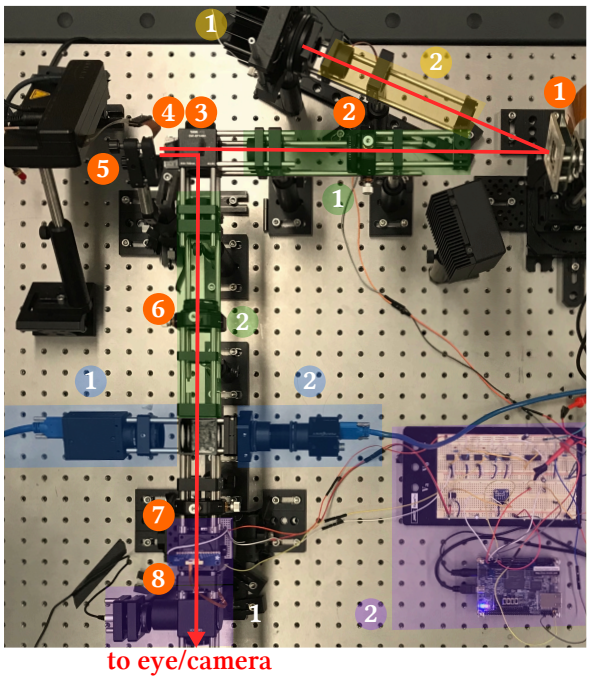
(e) Camera focuses on far focal plane

Figure 7.8: **Comparison between optimization filtering and ConeTilt.** (a) The scene contains two planes at 10 cm and 1 m. (b) shows the content created by optimization-based filtering, and (c) shows the inputs for the proposed ConeTilt display. We focus the camera (d) on the front plane and (e) on the back plane at both the center position and a slight shift to the right. The shift causes the back plane to move by 1 pixel and the front plane to move by 10 pixels in the same direction.

lessons underlying Narain *et al.* [2015] and Mercier *et al.* [2017] — namely, optimization-based content generation — could be extended to our hardware as well and, in this sense, the two approaches are complementary.

7.4 Proof-of-Concept Prototype

In the section, we provide details in building our proof-of-concept ConeTilt display, which is shown in Figure 7.9. Our prototype directly follows the schematic shown in Figure 7.4 and is built with off-the-shelf components.



1 DMD
2 1st circular aperture
3 pellicle beamsplitter
4 linear polarizer
5 phase SLM
6 2nd circular aperture
7 field lens (f=60mm)
8 focus tunable lens
1 LED
2 1:3 afocal relay
1 1:1 4f relay
2 1:1 4f relay
1 calibration camera
2 calibration camera
1 focal length tracking optics
2 focal length tracking circuits

Figure 7.9: **Lab Prototype.** The red line shows the path of a light ray. The DMD is TI DLP 7000, which has a pixel pitch of 13.6 um, the linear polarizer is Edmund Optics 86-178, the pellicle beamsplitter is Thorlabs CM1-BP145B1, the phase SLM is Holoeye LETO, which has a pixel pitch of 6.4 um, the focus tunable lens is Optotune EL-10-30, and the LED is LED Engin LZP-L4MD00. The distance between the relayed DMD (SLM) to the tunable lens is 58 mm. The depth range of the display is 35 cm to infinity.

7.4.1 System Overview

Our prototype implements a light cone of 1.2 degrees in radius, a field-of-view of 6.8 degrees in diameter, and an eye box of 2.4 mm in diameter. Note that since our prototype uses a physical field lens to implement the default tilt, the field of view is the same as a multifocal display of the same configuration. The small field-of-view is due to the simplicity of our implementation and can be increased by moving the tunable lens closer to the phase SLM, *i.e.*, reducing d, which is currently 58 mm.

The light comes from a green LED whose spectrum centers at 520 nm. We put a diffuser in front of the LED and build a 1:2.5 afocal system to make the light covers the DMD uniformly. After reflected by the DMD, the light is relayed by the first $4f$ system with $f = 100$ mm and passes through a light polarizer, which is used to enable the phase-only mode on the SLM. We set the target wavelength of the phase SLM to 520 nm. Since our phase SLM is reflective, we place a beamsplitter in front of the SLM. We use a pellicle beamsplitter to avoid the ghosting and optical-axis shift caused by cube- or plate-beamsplitters. This is only to simplify the implementation. The aperture of the second $4f$ system crops any light exceeds the original range of the light cone. Finally, the light goes through the focus tunable lens and reach in eye/camera. The distance between the relayed phase SLM/DMD and the tunable lens is 58 mm. The LED and the DMD are controlled by the proposed intensity-modulation we introduced in Chapter 5. The focus tunable lens is controlled by the focal-length tracking module introduced in Chapter 6. Note that the DMD micromirrors flip along their diagonal axes. To account for this, we rotated the DMD, the phase SLM, and the camera by 45 degrees.

7.4.2 Calibration and Alignment

To calibrate the system, we place a beamsplitter, between the second $4f$ system and the tunable lens. The beamsplitter forks the optical path for extra cameras without being affected by the tunable lens, and we remove the beamsplitter once calibration is completed. We connect two cameras (marked by blue in Figure 7.9) — one focuses on the phase SLM and the other on the infinity (*i.e.*, the aperture plane of the first $4f$ system).

In the following, we provide our calibration procedure.

1. *Virtually Attaching the Phase SLM and the DMD.* We focus the calibration camera (blue 1 in Figure 7.9) on the phase SLM and move the DMD till it is in sharp focus. Note that the phase SLM pixels are transparent, and this makes the calibration process more difficult. Fortunately, we find that when the input polarization of the SLM is in 45 degree with respect to its long axis, the SLM operates in amplitude mode and enables us to display visible patterns. The trick makes the calibration process more accurate.

2. *Adjust the Tilt of the Phase SLM.* We temporally mount a mirror on the unused side of the beamsplitter B1, focus the camera at infinity, and close the aperture of the first $4f$ system to the smallest. The small aperture enables us to send narrow beams toward the beamsplitter. When the phase SLM and the mirror have different tilting angles, the camera will see two copies of the beam (one from the mirror and the other from the SLM), and we adjust the tilt of phase SLM to overlap the two copies.

3. *Aperture of the first $4f$ System.* We use the calibration camera focusing at infinity to adjust the aperture size of the first $4f$ system (*i.e.*, the size of the light cone). Since the size of the light cone is limited by the capability of the phase SLM to tilt light, we show the phase SLM to help the calibration. We display an all white image on the DMD and tilt all pixels by $2u_m$ in the same direction. With the second aperture open, by changing the tilting direction, we adjust the aperture location and size till the tilted light cones touch the boundary of the original light cone in every direction.

4. *Aperture of the second $4f$ System.* After finishing the last step, we adjust the location and the size of the aperture so that the entire cone is cropped when tilted by $2u_m$ toward every direction.

5. *Placing the Field Lens.* Since the DMD is virtually relayed by the $4f$ relays, it makes adjusting the position of the field lens slightly trickier. We put a camera at the output side and focus the camera on the virtual copy of the DMD before placing the field lens. The position of the field lens is chosen to maximize the sharpness of DMD. The focal length of the field lens is 60 mm.

6. *Adjust the Position of the Tunable Lens.* The distance between the focus tunable lens and the field lens is determined by the focal length of the field lens and is very important. Since the default tilt implemented by the field lens makes light cones of all pixels to overlap at the focal plane, we first focus a camera on the output side on the focal plane where we see a sharp cone. We then place the tunable lens at the location where the cone is sharpest.

7. *Find Pixel Correspondence between DMD and Phase SLM.* To find the pixel correspondence between the DMD and the phase SLM, we focus the camera on the phase SLM (and the DMD, since they are colocated.) We label the patterns shown on the phase SLM and the DMD and use the results to calculate the pixel mapping. We then resample the phase function according to the correspondence to display on the phase SLM.

Note that the spatial resolution is preserved only when the phase SLM is perfectly colocated with the DMD and is optically thin. In our prototype, we observe a small loss ($\sim 1.5\times$) in spatial resolution.

7.4.3 Reducing the Bulk of the Prototype

The bulk of our prototype is mostly contributed by the off-the-shelf components which force us to *optically* colocate the DMD, the phase SLM, and the image plane of the tunable lens. In principle, the footprint of the overall system can be significantly reduced if customized components are used. For example, a transparent phase SLM can be attached directly to an OLED panel and be placed in front of a tunable lens. The angular range of the OLED can be controlled during the manufacture process by adding microlenses onto each pixel similar to the method used by camera pixels. The second aperture can be directly controlled by the aperture of the tunable lens. Compared to a typical multifocal display, a ConeTilt display only requires an additional transparent phase SLM attached to the display.

7.5 Experimental Results

We showcase the performance of ConeTilt on scenes designed to highlight the important features of the proposed method. Before we show the results, let us first introduce the inputs and the capturing process we used when conducting the experiments.

Inputs. Given a 3D scene, we first discretize the scene according to the depth of the focal planes (in diopters) and assign each point in the scene to its nearest focal plane. Given the size of the light cone, we remove all pixels that are completely occluded. We then follow the algorithm described in Section 7.3.3 to compute the tilt for each pixel and the phase function to show with each focal plane.

While the inputs can be generated very efficiently, the number of phase functions our prototype can show is limited by the refresh rate of the phase SLM. To circumvent the limitation, we find that most scenes need only two phase functions — one for unoccluded content and the other for the directly occluded content — to effectively create occlusion. As a result, we divide each frame into visible and directly overlapped contents, and we compute the phase function for each of them. This method significantly lessens the required refresh rate of the phase SLM. Note that while the proposed display is compatible with optimization-based content-generation methods [Akeley *et al.*, 2004, Choi *et al.*, 2019, Mercier *et al.*, 2017, Narain *et al.*, 2015, Xiao *et al.*, 2018], to evaluate the effectiveness of ConeTilt, we do not apply any filtering to the content and leave optimization-based content generation for future work.

Capturing Process. We use a FLIR Grasshopper grayscale camera with a Nikkor 35 mm prime lens set to $f/22$ to capture the photos. The camera is put on a linear translation stage in front of the tunable lens. We use a 1:1 $4f$ relay to map the camera to the aperture of the tunable lens. This provides enough space

for the translation stage and reduces the magnification caused by the unnecessary distance between the camera and the lens. To simplify the synchronization between the DMD and the phase SLM, we capture the directly visiable and overlapped content separately and sum the two images without moving the camera. Since our prototype is grayscale, to showcase RGB contents, we display and capture each color channel separately and sum the captured images.

In the following, we show the results on various scenes, and we encourage readers to check the videos where we move the camera left and right or change its focus in the supplemental materials. Note that during the capturing process, we do not re-render the scene based on the camera configurations, *i.e.*, the camera is entirely independent to the display.

7.5.1 Control the Light Cones with ConeTilt

First, we verify the ability of ConeTilt to tilt light. Figure 7.10 shows the light entering the tunable lens under different configurations of tilts. We show a full white image on the DMD and tilt every pixel in the same direction. The results are captured by focusing a camera on the aperture of the tunable lens (see Step 3 in Section 7.4.2). As can be seen from the results, ConeTilt effectively controls the light cones of all pixels.

7.5.2 Hiding Content Behind Occluders

We demonstrate the capability to hide content behind an occluder and reveal it when the camera/eye shifts — all without re-rendering the scene. As shown in Figure 7.11, the scene contains an opaque smiley face in the front and a question mark and the text “conetilt” in the back. We shift the camera with a translation stage from left to right; when the camera is at the left position the text should be occluded by the smiley head, and the text should be revealed when the camera shifts to the right. As can be seen from the results, the smiley face rendered by the typical multifocal display fails to occlude the text and even makes the text brighter due to the additive nature of the front and the back focal planes. In comparison, the text is occluded and revealed when ConeTilt is applied. The lower intensity of the text is as expected, since most of the light from the text are occluded by the smiley face, as it will happen in reality. The results showcase the capability of the proposed display to support small shifts of pupil without the help of a gaze tracker or any additional rendering. The property is useful, as it can lower the hardware and computation requirements of VR displays to create immersive virtual worlds.

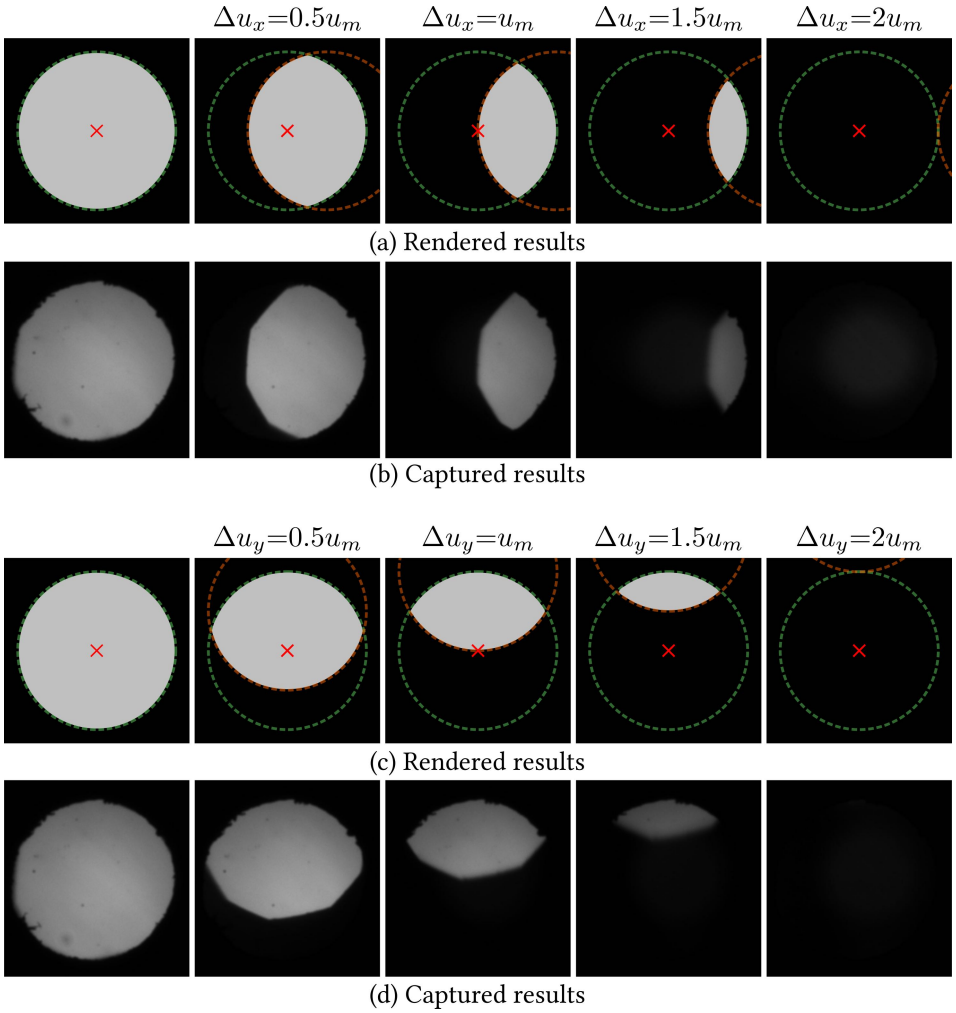


Figure 7.10: **Sum of all tilted light cones.** We focus the camera on the aperture of the tunable lens and show an all-one image on the DMD with different global tilting configurations. The camera sees the sum of the tilted light cones from all pixels.

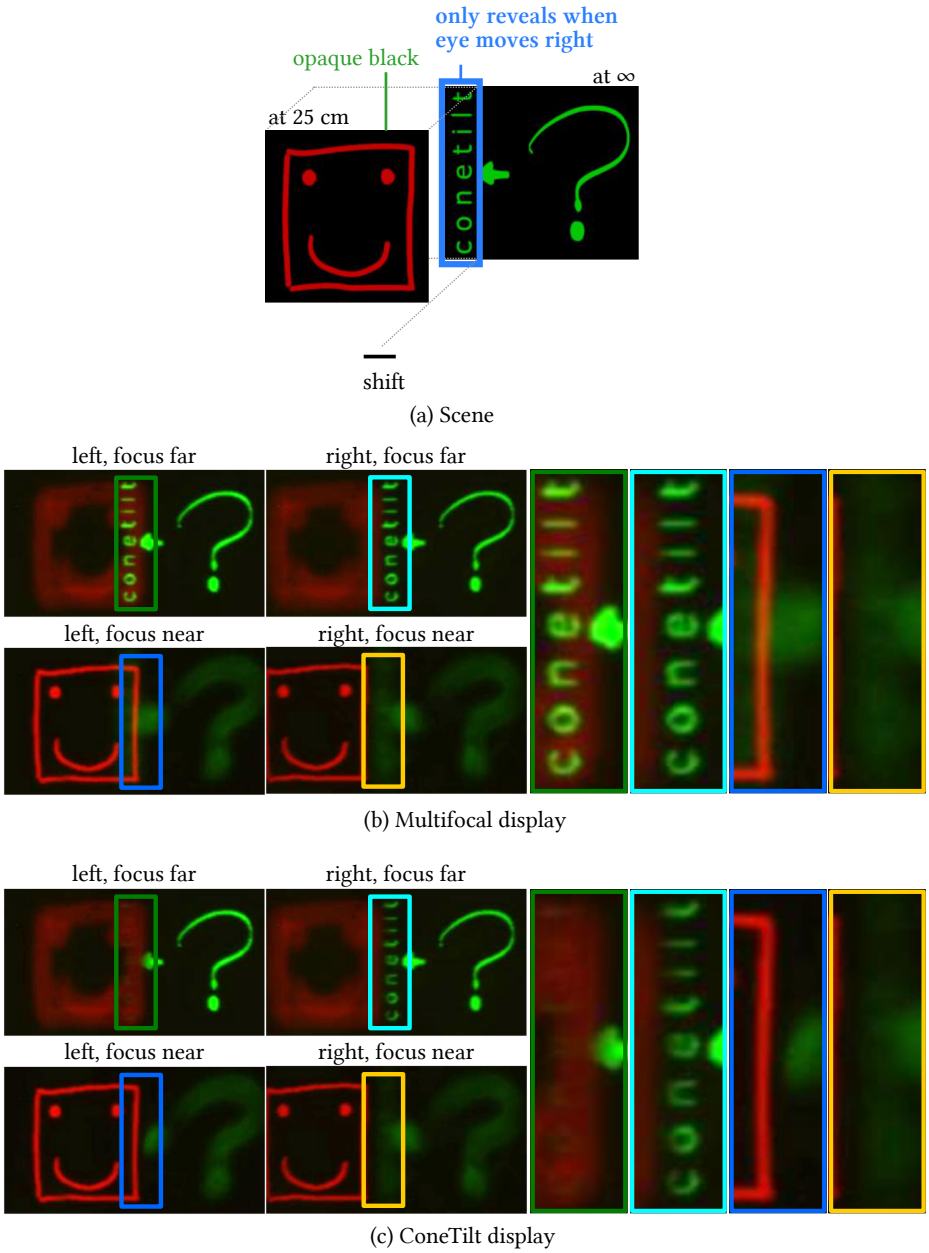


Figure 7.11: **Creating occlusion cue.** The figure shows the captured photo of the scene shown in (a) when the camera is at a left and a right position. On the left shows the whole images, and on the right shows the color-coded insets. The front smiley face is opaque and should occlude the text and part of the arrow when the camera is at the left position.

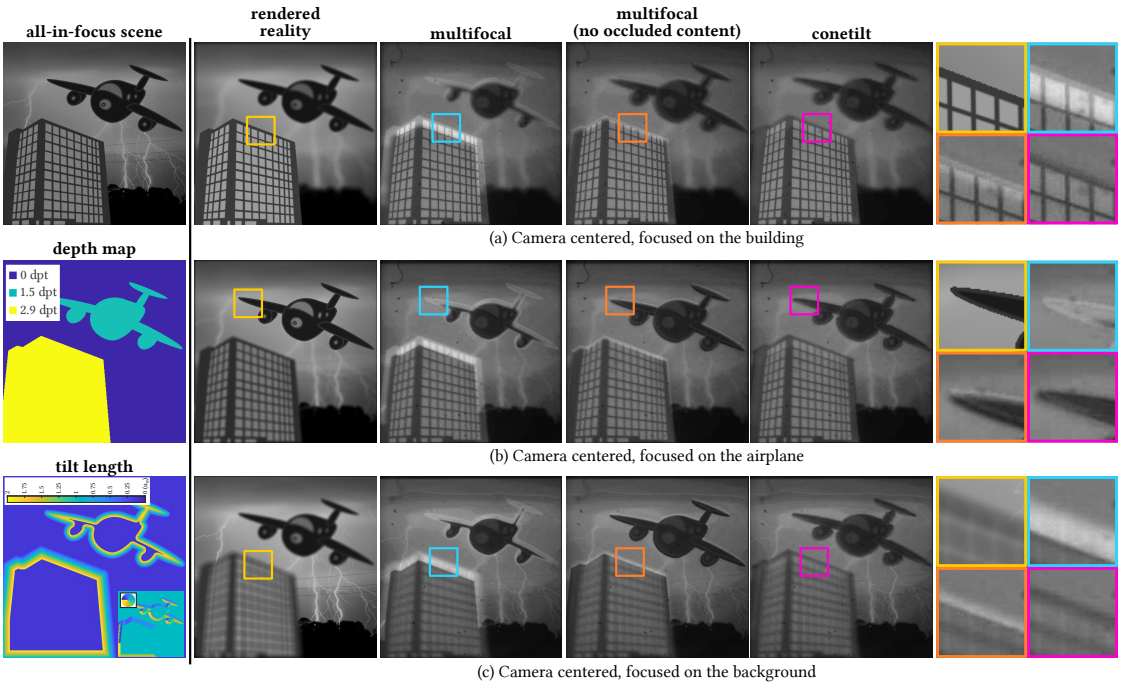


Figure 7.12: **Rendered and captured results on the lightning scene.** The figure shows the captured images of the scene shown on the top left. The tilting vectors are shown on the bottom left with the direction of the tilting vectors shown in the inset. "no occluded content" means that we remove the directly occluded regions in the background.

7.5.3 Generic Occluding Contours

We show captured results on scenes with more complicated occluding contours in Figure 7.12, Figure 7.13, Figure 7.14, and Figure 7.1. From the results, we have the following observations.

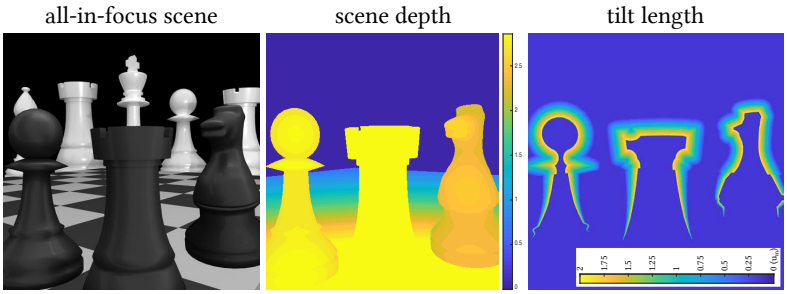
Reduced leakage. All results consistently demonstrate that the ConeTilt display effectively reduces light leaking from the background onto foreground occluders. Please see the boundaries of the building in Figure 7.12a, the top of the rock in Figure 7.13b, and the boundary of the leaf in Figure 7.14b for examples. As can be seen from the results, while removing the directly occluded regions in the background helps reduce the light leakage in multifocal displays, it does not completely solve the problem. This can be easily seen from the supplemented videos when the camera shifts left and right.

Improved contrast. To quantitatively characterize the effect of ConeTilt on the contrast of the foreground, for the same 3D scene shown in Figure 7.14, we capture an additional result shown in Figure 7.15 by removing the background of the scene while the camera focused on the foreground. This enables us to examine how showing the background affects the pixel values of the foreground.

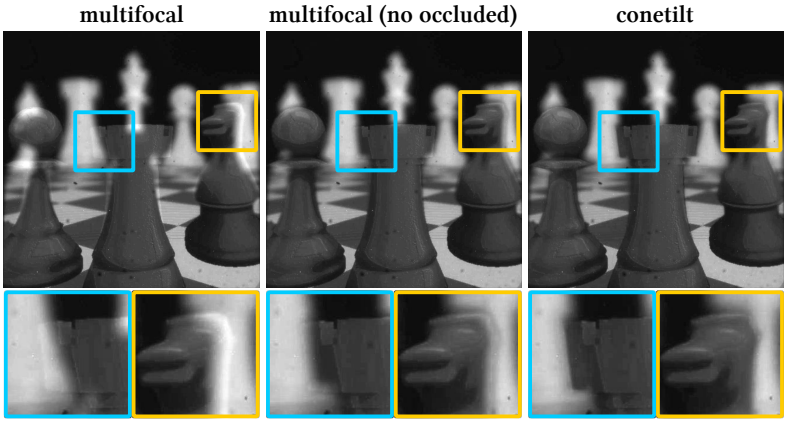
In reality, since the leaf is opaque and is in focus, showing the background does not affect its pixel values. Therefore, the correlation coefficient between the pixel values before and after showing the background should be one. As can be seen from Figure 7.15, in the multifocal display, due to the leakage from the background, we see an increase in the pixel values after showing the background and thereby a reduction in the contrast, which is reflected by the small correlation coefficient. While removing the directly occluded background helps reduce the leakage and increase the contrast, it has limited effects. In comparison, the ConeTilt display achieves a correlation coefficient closest to one. We want to point out that due to the ghosting effect discussed in Section 7.3.6, the ConeTilt display does not achieve a correlation coefficient of one.

Defocus cues. The captured results also demonstrate another advantage of ConeTilt displays over typical multifocal displays. When a multifocal display tries to reduce light leakages by removing directly occluded content on the background, it deteriorates the defocus cue of the occluder when the camera focuses on the background. As can be seen from Figure 7.13c and Figure 7.14c, the defocused foregrounds of the multifocal display (no overlap) look unnaturally sharp even though in reality they should be blurred due to defocus. In comparison, the ConeTilt display successfully renders blurred foregrounds. While subtle, it has been shown that successfully generating the defocus cue is important for improving the immersion of VR displays [Zannoli *et al.*, 2016].

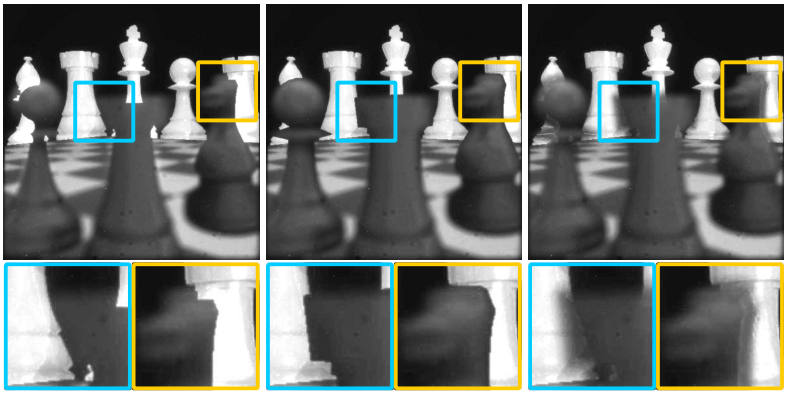
Artifacts. The captured results also faithfully shows the artifacts of ConeTilt displays. We can see the dark halo in Figure 7.13b around the rock and Figure 7.14b around the leaf. Note that when removing directly occluded background, the multifocal display also suffers from dark halo. The ConeTilt display also fails to prevent light leakage when two occluding boundaries are too close, as can be seen in Figure 7.14c at the narrow breaking of the leaf. We also want to point out that there are also light leakages at the tips of leaf and the stem. This is due to the smoothness constraint we apply when solving the phase function. For example, pixels at the upper part of the stem need to the tilted upward, whereas the bottom part needs to be tilted downward; this causes the center portion of the stem to be un-tilted. The artifact can be removed by not showing these background pixels, at a cost of increasing dark halo.



(a) Scene

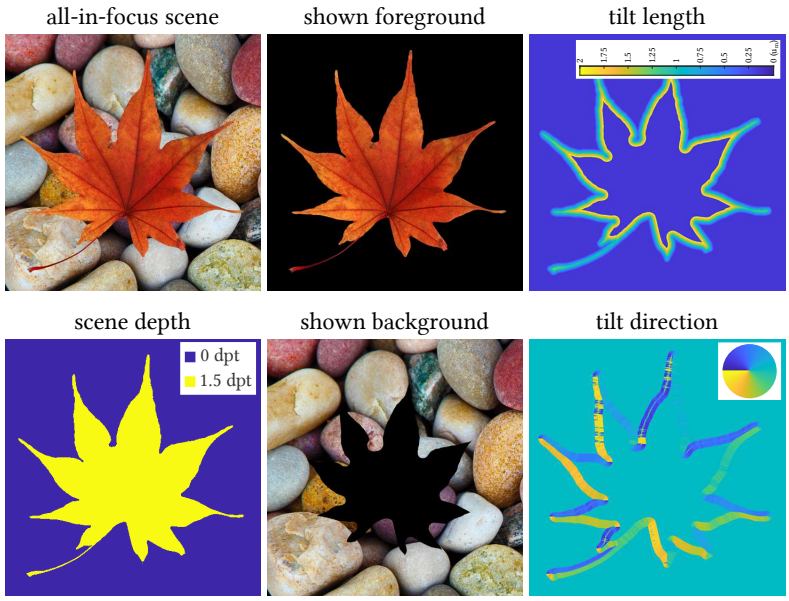


(b) Captured results, camera focused on the front rock

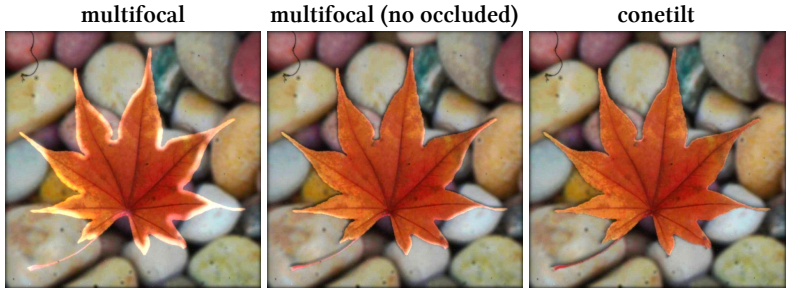


(c) Captured results, camera focused on the back king

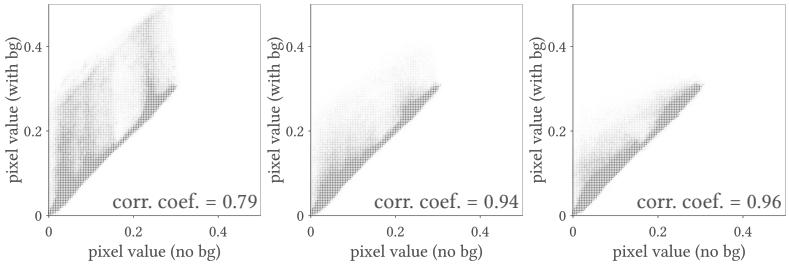
Figure 7.13: **Results on the chess scene.** This figure shows the captured photos of the scene (a) when the camera (b) focuses on the center rock piece and (c) focuses on the center king in the back.



(a) Scene and ConeTilt configurations



(b) Captured results, camera focuses on the leaf



(c) Pixel values of the leaf before and after showing the background, camera focuses on the foreground

Figure 7.14: **Results on the leaf scene.** This figure shows the results on (a) the scene when the camera (b)focuses on the leaf and (c) focuses on the background with the leaf colored as opaque black. (d) shows the pixel values of the leaf when the camera focuses on it. The x-axis represents the pixel values when the background is not shown, and the y-axis represents the pixel values after showing the background. 3D Scene modified from [Leaves].

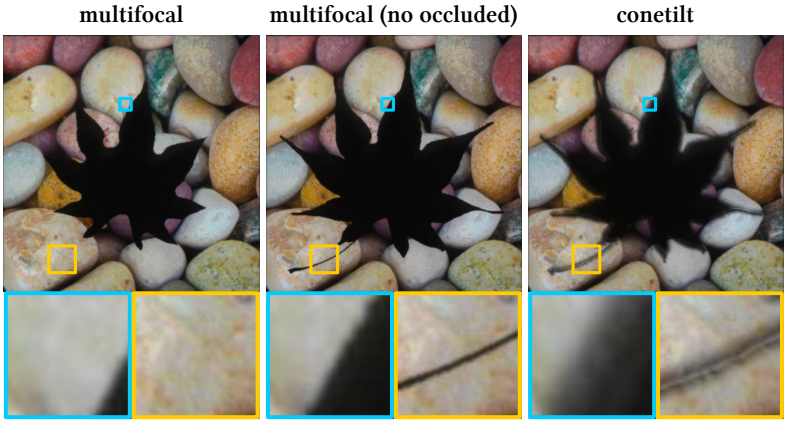


Figure 7.15: **Captured results on the modified leaf scene.** This figure shows the results on the scene shown in Figure 7.14 when the color of the foreground (leaf) is replaced by solid black. The camera focuses on the background.

7.6 Conclusions

This chapter proposes a simple but effective technology for displaying immersive virtual scenes on multifocal displays. While our current prototype is bulky and limited by the capability of our phase SLM, the proposed ConeTilt operator can be neatly incorporating into existing multifocal displays and can easily benefit from the rapid-evolving light modulation technologies. We believe the technology proposed in the paper for high spatial-resolution light manipulation could spur innovation in virtual and augmented reality systems and in traditional light-field displays.

8 Conclusion

If we say that photographs are the projection of a photographer's mind, virtual worlds should be the reflection of a director's imagination. Three-dimensional displays have come a long way since their debut in 1833 when Sir Charles Wheatstone developed the first stereoscope. Recent progresses in location and depth estimation, head and gaze tracking, and display technologies have popularized AR/VR devices in entertainment, education, and business industries.

That being said, AR/VR displays still have a long way to go. The research in the dissertation is motivated from my experience of playing VR games. After spending a wonderful afternoon with my friends in a virtual world, I felt the dizziness caused by the vergence-accommodation conflict, and I started to realize the implication of the lack of focus support in VR displays.

The main lesson that we have learned through the thesis research is that the generation of light can be made significantly more efficient with a small modification to display hardware. Each of the advancements carried out in the dissertation starts by introducing a new functionality into the display — an intensity-modulated light source for high bit-depth and high speed projection, an oscillating tunable lens for displaying dense focal stacks, and a programmable field lens for creating opaque virtual objects. The proposed optical and hardware designs not only enable novel functionalities but also relieve a significant amount of computational burdens if implemented via a software-only design.

Our ConeTilt display is far from perfect. While the customized DMD-based projector enables us to display focal planes in a high frame rate, it inevitably increases the footprint of the display. Next-generation micro-display technologies like OLEDs or microLEDs can significantly reduce the size of the proposed displays. We will also benefit from next-generation technologies that enables unprecedented capabilities of light modulation. For example, a SLM that can simultaneously controls both the intensity and the phase of a pixel, a programmable holographic optical element that can reshape the wavefront emitted by the display, or a nano-structure optical element that can significantly reduce the bulk of the display or manipulate light in an unimaginable manner.

The thesis research focuses on introducing novel hardware functionalities. Nevertheless, during the research, we have learned that the capability of the proposed designs have yet been maximized. For example, ConeTilt displays will significantly benefit from re-designed rendering and display pipelines. Optimization-based or deep-learning-based content generation can be used to solve the dark-halo artifacts created by ConeTilt display. A rendering process that efficiently returns an all-in-focus image and all occluded content near occluding boundaries will also make ConeTilt displays more efficient. A new display buffer that enables rapid refresh of pixel values and efficient storage of hidden contents will significantly improve the bandwidth requirement of ConeTilt displays.

The goal of 3D displays is to replicate reality by deceiving all perceptual cues used by the human visual system. The thesis research tries to mimic the physical process of light generating in the real world, but deceiving perceptual cues do not require following the conventional route. We imagine future 3D displays that generate light fields by boldly breaking the boundary set by the physical process. The displays will equip high contrast ratio, wide color gamut, fine bit-depth, high frame rate, and satisfies all perceptual cues. We look forward to immersing in the next-generation 3D displays.

Hope lies in dreams, in imagination, and in the courage of those who dare to make dreams into reality.

— Jonas Salk

Bibliography

Kaan Akşit, Ward Lopes, Jonghyun Kim, Peter Shirley, and David Luebke. 2017. Near-eye Varifocal Augmented Reality Display Using See-through Screens. *ACM Transactions on Graphics* 36, 6, Article 189 (Nov. 2017), 13 pages. https://doi.org/10.1145/3130800.3130892

Kurt Akeley. 2004. *Achieving Near-correct Focus Cues Using Multiple Image Planes.* Ph.D. Dissertation. Stanford University.

Kurt Akeley, Simon J. Watt, Ahna Reza Girshick, and Martin S. Banks. 2004. A Stereo Display Prototype with Multiple Focal Distances. *ACM Transactions on Graphics* 23, 3 (Aug. 2004), 804–813. https://doi.org/10.1145/1015706.1015804

Ehsan Arbabi, Amir Arbabi, Seyedeh Mahsa Kamali, Yu Horie, MohammadSadegh Faraji-Dana, and Andrei Faraon. 2018. MEMS-Tunable Dielectric Metasurface Lens. *Nature communications* 9, 1 (2018), 812.

BARB. 2018. Television ownership in private domestic households 1956-2018. https://www.barb.co.uk/resources/tv-ownership/.

Stefan Bernet and Monika Ritsch-Marte. 2008. Adjustable Refractive Power From Diffractive Moiré Elements. *Applied optics* 47, 21 (2008), 3722–3730.

J.B. Breckinridge and D.G. Voelz. 2011. *Computational Fourier Optics: A MATLAB Tutorial.* SPIE Press. https://books.google.com/books?id=2COGSQAACAAJ

Ozan Cakmakci and Jannick Rolland. 2006. Head-Worn Displays: a Review. *Journal of display technology* 2, 3 (2006), 199–216.

Fergus W Campbell. 1957. The Depth of Field of the Human Eye. *Optica Acta: International Journal of Optics* 4, 4 (1957), 157–164.

Pew Research Center. 2018. Mobile Fact Sheet. http://www.pewinternet.org/fact-sheet/mobile/.

Jen-Hao Rick Chang, BVK Vijaya Kumar, and Aswin C Sankaranarayanan. 2016. 2^{16} Shades of Gray: High Bit-depth Projection Using Light Intensity Control. *Optics express* 24, 24 (2016), 27937–27950.

Jen-Hao Rick Chang, B. V. K. Vijaya Kumar, and Aswin C. Sankaranarayanan. 2018. Towards Multifocal Displays with Dense Focal Stacks. *ACM Transactions on Graphics* 37, 6, Article 198 (Dec. 2018), 13 pages. https://doi.org/10.1145/3272127.3275015

Suyeon Choi, Seungjae Lee, Youngjin Jo, Dongheon Yoo, Dongyeon Kim, and Byoungho Lee. 2019. Optimal Binary Representation via Non-convex Optimization on Tomographic Displays. *Optics Express* 27, 17 (2019), 24362–24381.

Steven A. Cholewiak, Gordon D. Love, Pratul P. Srinivasan, Ren Ng, and Martin S. Banks. 2017. Chromablur: Rendering Chromatic Eye Aberration Improves Accommodation and Realism. *ACM Transactions on Graphics* 36, 6, Article 210 (Nov. 2017), 12 pages. https://doi.org/10.1145/3130800.3130815

Oliver S. Cossairt, Joshua Napoli, Samuel L. Hill, Rick K. Dorval, and Gregg E. Favalora. 2007a. Occlusion-capable Multiview Volumetric Three-dimensional Display. *Applied Optics* 46, 8 (Mar 2007), 1244–1250. https://doi.org/10.1364/AO.46.001244

Oliver S Cossairt, Joshua Napoli, Samuel L Hill, Rick K Dorval, and Gregg E Favalora. 2007b. Occlusion-Capable Multiview Volumetric Three-Dimensional Display. *Applied optics* 46, 8 (2007), 1244–1250.

James E Cutting and Peter M Vishton. 1995. Perceiving Layout and Knowing Distances: The Integration, Relative Potency, and Contextual Use of Different Information About Depth. In *Perception of space and motion*. 69–117.

Gerwin Damberg, Anders Ballestad, Eric Kozak, Johannes Minor, Raveen Kumaran, and James Gregson. 2015. High brightness HDR projection using dynamic phase modulation. In *SIGGRAPH Emerging Technologies*.

Gerwin Damberg, James Gregson, and Wolfgang Heidrich. 2016. High Brightness HDR Projection Using Dynamic Freeform Lensing. *ACM Transactions on Graphics* 35, 3, Article 24 (May 2016), 11 pages. https://doi.org/10.1145/2857051

Gerwin Damberg and Wolfgang Heidrich. 2015. Efficient freeform lens optimization for computational caustic displays. *Optics Express* 23, 8 (2015), 10224–10232.

Gerwin Damberg, Helge Seetzen, Greg Ward, Wolfgang Heidrich, and Lorne Whitehead. 2007. High dynamic range projection systems. *SID Symposium Digest of Technical Papers* 38, 1 (2007), 4–7.

Niranjan Damera-Venkata and Nelson L Chang. 2009. Display supersampling. *ACM Transactions on Graphics* 28, 1 (2009), 9.

Andrew T. Duchowski, Donald H. House, Jordan Gestring, Rui I. Wang, Krzysztof Krejtz, Izabela Krejtz, Radoslaw Mantiuk, and Bartosz Bazyluk. 2014. Reducing Visual Discomfort of 3D Stereoscopic Displays with Gaze-contingent Depth-of-field. In *Proceedings of the ACM Symposium on Applied Perception*. 39–46.

eMirage. 2017. Barcelona Pavillion. `https://download.blender.org/demo/test/pabellon_barcelona_v1.scene_.zip`.

Hewlett Packard Enterprise. 2017. Sales of VR Head-Mounted Displays. `https://www.theatlas.com/charts/rJr57t6C`.

JA Ferwerda and S Luka. 2009. A high resolution high dynamic range display for vision research. *Journal of Vision* 9, 8 (2009).

Jason Geng. 2013. Three-dimensional Display Technologies. *Advances in Optics and Photonics* 5, 4 (2013), 456–535.

Gabriele Guarnieri, Luigi Albani, and Giovanni Ramponi. 2008. Image-splitting techniques for a dual-layer high dynamic range LCD display. *Journal of Electronic Imaging* 17, 4 (2008), 043009–043009.

Brian Guenter, Mark Finch, Steven Drucker, Desney Tan, and John Snyder. 2012. Foveated 3D Graphics. *ACM Transactions on Graphics* 31, 6, Article 164 (Nov. 2012), 10 pages. `https://doi.org/10.1145/2366145.2366183`

Rolf R Hainich and Oliver Bimber. 2014. *Displays: Fundamentals and Applications*. CRC press.

Hamamatsu. [n.d.]. Optical Phase Modulator, LCOS-SLM X13138-01. `http://www.hamamatsu.com/jp/en/X13138-01.html`.

Eugene Hecht. 2002. *Optics*. Addison-Wesley.

Felix Heide, James Gregson, Gordon Wetzstein, Ramesh Raskar, and Wolfgang Heidrich. 2014. Compressive multi-mode superresolution display. *Optics Express* 22, 12 (2014), 14981–14992.

Robert T Held, Emily A Cooper, and Martin S Banks. 2012. Blur and Disparity are Complementary Cues to Depth. *Current biology* 22, 5 (2012), 426–431.

Hitoshi Hiura, Kazuteru Komine, Jun Arai, and Tomoyuki Mishina. 2017. Measurement of Static Convergence and Accommodation Responses to Images of Integral Photography and Binocular Stereoscopy. *Optics Express* 25, 4 (2017), 3454–3468.

David M Hoffman, Ahna R Girshick, Kurt Akeley, and Martin S Banks. 2008. Vergence–accommodation Conflicts Hinder Visual Performance And Cause Visual Fatigue. *Journal of vision* 8, 3 (2008), 33–33.

Reynald Hoskinson and Boris Stoeber. 2008. High-dynamic range image projection using an auxiliary MEMS mirror array. *Optics Express* 16, 10 (2008), 7361–7368.

Reynald Hoskinson, Boris Stoeber, Wolfgang Heidrich, and Sidney Fels. 2010. Light reallocation for high contrast projection using an analog micromirror array. *ACM Transactions on Graphics* 29, 6 (2010), 165.

Xinda Hu and Hong Hua. 2014. High-resolution Optical See-through Multi-focal-plane Head-mounted Display Using Freeform Optics. *Optics express* 22, 11 (2014), 13896–13903.

Hong Hua. 2017. Enabling Focus Cues in Head-mounted Displays. *Proc. IEEE* 105, 5 (2017), 805–824.

Fu-Chung Huang, Kevin Chen, and Gordon Wetzstein. 2015. The Light Field Stereoscope: Immersive Computer Graphics via Factored Near-eye Light Field Displays with Focus Cues. *ACM Transactions on Graphics* 34, 4, Article 60 (July 2015), 12 pages. `https://doi.org/10.1145/2766922`

Fu-Chung Huang, Gordon Wetzstein, Brian A. Barsky, and Ramesh Raskar. 2014. Eyeglasses-free Display: Towards Correcting Visual Aberrations with Computational Light Field Displays. *ACM Transactions on Graphics* 33, 4, Article 59 (July 2014), 12 pages. `https://doi.org/10.1145/2601097.2601122`

Yung-Chih Huang and Jui-Wen Pan. 2014. High contrast ratio and compact-sized prism for DLP projection system. *Optics Express* 22, 14 (2014), 17016–17029.

Masahiko Inami, Naoki Kawakami, Dairoku Sekiguchi, Yasuyuki Yanagida, Taro Maeda, and Susumu Tachi. 2000. Visuo-haptic Display Using Head-mounted Projector. In *Proceedings of the IEEE Virtual Reality*. 233–240.

Afsoon Jamali, Douglas Bryant, Yanli Zhang, Anders Grunnet-Jepsen, Achintya Bhowmik, and Philip J Bos. 2018a. Design of a Large Aperture Tunable Refractive Fresnel Liquid Crystal Lens. *Applied optics* 57, 7 (2018), B10–B19.

Afsoon Jamali, Comrun Yousefzadeh, Colin McGinty, Douglas Bryant, and Philip Bos. 2018b. A Continuous Variable Lens System to Address the Accommodation Problem in VR and 3D Displays. *Imaging and Applied Optics* (2018), 3Tu2G.5.

Youngjin Jo, Seungjae Lee, Dongheon Yoo, Suyeon Choi, Dongyeon Kim, and Byoungho Lee. 2019. Tomographic Projector: Large Scale Volumetric Display with Uniform Viewing Experiences. *ACM Transactions on Graphics* 38, 6, Article 215 (Nov. 2019), 13 pages. `https://doi.org/10.1145/3355089.3356577`

Paul V Johnson, Jared AQ Parnell, Joohwan Kim, Christopher D Saunter, Gordon D Love, and Martin S Banks. 2016. Dynamic Lens and Monovision 3D Displays to Improve Viewer Comfort. *Optics express* 24, 11 (2016), 11808–11827.

Andrew Jones, Ian McDowall, Hideshi Yamada, Mark Bolas, and Paul Debevec. 2007. Rendering for an Interactive 360&Deg; Light Field Display. *ACM Transactions on Graphics* 26, 3, Article 40 (July 2007). `https://doi.org/10.1145/1276377.1276427`

Hoon Kang, SuDong Roh, InSu Baik, HyunJoon Jung, WooNam Jeong, JongKeun Shin, and InJae Chung. 2010. A Novel Polarizer Glasses-type 3D Displays with a Patterned Retarder. In *SID Symposium Digest of Technical Papers*, Vol. 41. 1–4.

Jonghyun Kim, Youngmo Jeong, Michael Stengel, Kaan Akşit, Rachel Albert, Ben Boudaoud, Trey Greer, Joohwan Kim, Ward Lopes, Zander Majercik, *et al.* 2019. Foveated AR: Dynamically-foveated Augmented Reality Display. *ACM Transactions on Graphics* 38, 4 (2019), 99.

Sang Soo Kim, Bong Hyun You, Heejin Choi, Brian H Berkeley, Dong Gyu Kim, and Nam Deog Kim. 2009. World's First 240Hz TFT-LCD Technology for Full-HD LCD-TV and Its Application to 3D Display. In *SID Symposium Digest of Technical Papers*, Vol. 40. 424–427.

Hidei Kimura, Taro Uchiyama, and Hiroyuki Yoshikawa. 2006. Laser Produced 3D Display in the Air. In *ACM SIGGRAPH Emerging Technologies*. 20.

Kiyoshi Kiyokawa, Yoshinori Kurata, and Hiroyuki Ohno. 2000. An Optical See-through Display for Mutual Occlusion of Real and Virtual Environments. In *Proceedings IEEE and ACM International Symposium on Augmented Reality (ISAR 2000)*. 60–67.

Robert Konrad, Emily A Cooper, and Gordon Wetzstein. 2016. Novel Optical Configurations for Virtual Reality: Evaluating User Preference and Performance with Focus-tunable and Monovision Near-eye Displays. In *Conference on Human Factors in Computing Systems (CHI)*. 1211–1220.

Robert Konrad, Nitish Padmanaban, Keenan Molner, Emily A. Cooper, and Gordon Wetzstein. 2017. Accommodation-invariant Computational Near-eye Displays. *ACM Transactions on Graphics* 36, 4, Article 88 (July 2017), 12 pages. https://doi.org/10.1145/3072959.3073594

Frank L Kooi and Alexander Toet. 2004. Visual Comfort of Binocular and 3D Displays. *Displays* 25, 2-3 (2004), 99–108.

George-Alex Koulieris, Bee Bui, Martin S. Banks, and George Drettakis. 2017. Accommodation and Comfort in Head-mounted Displays. *ACM Transactions on Graphics* 36, 4, Article 87 (July 2017), 11 pages. https://doi.org/10.1145/3072959.3073622

Gregory Kramida. 2016. Resolving the Vergence-accommodation Conflict in Head-mounted Displays. *IEEE Transactions on visualization and computer graphics* 22, 7 (2016), 1912–1931.

Y Kusakabe, M Kanazawa, Y Nojiri, M Furuya, and M Yoshimura. 2009. A high-dynamic-range and high-resolution projector with dual modulation. In *SPIE Electronic Imaging*. 72410Q–72410Q.

Marc Lambooij, Marten Fortuin, Ingrid Heynderickx, and Wijnand IJsselsteijn. 2009. Visual Discomfort and Visual Fatigue of Stereoscopic Displays: A Review. *Journal of Imaging Science and Technology* 53, 3 (2009), 30201–1.

Douglas Lanman and David Luebke. 2013. Near-eye Light Field Displays. *ACM Transactions on Graphics* 32, 6, Article 220 (Nov. 2013), 10 pages. https://doi.org/10.1145/2508363.2508366

Vincent Laude. 1998. Twisted-nematic Liquid-crystal Pixelated Active Lens. *Optics Communications* 153, 1-3 (1998), 134–152.

Autumn Leaves. [n.d.]. Autumn Leaves HD PNG. http://pluspng.com/autumn-leaves-hd-png-5594.html.

Benjamin Lee. 2018. Introduction to Âś12 Degree Orthogonal Digital Micromirror Devices (DMDs). http://www.ti.com/lit/an/dlpa008b/dlpa008b.pdf.

Seungjae Lee, Youngjin Jo, Dongheon Yoo, Jaebum Cho, Dukho Lee, and Byoungho Lee. 2019. Tomographic Near-eye Displays. *Nature communications* 10, 1 (2019), 2497.

Peter Lincoln, Alex Blate, Montek Singh, Andrei State, Mary C. Whitton, Turner Whitted, and Henry Fuchs. 2017. Scene-adaptive High Dynamic Range Display for Low Latency Augmented Reality. In *Proceedings of the 21st ACM SIGGRAPH Symposium on Interactive 3D Graphics and Games*.

Peter Lincoln, Alex Blate, Montek Singh, Turner Whitted, Andrei State, Anselmo Lastra, and Henry Fuchs. 2016. From Motion to Photons in 80 Microseconds: Towards Minimal Latency for Virtual and Augmented Reality. *Transactions on visualization and computer graphics* 22, 4 (2016), 1367–1376.

G Lippmann. 1908. Épreuves Réversibles Donnant la Sensation du Relief. *Journal de Physique Théorique et Appliquée* 7, 1 (1908), 821–825.

Sheng Liu, Dewen Cheng, and Hong Hua. 2008. An Optical See-through Head Mounted Display with Addressable Focal Planes. In *IEEE/ACM International Symposium on Mixed and Augmented Reality*. 33–42.

Sheng Liu and Hong Hua. 2009. Time-multiplexed Dual-focal Plane Head-mounted Display with a Liquid Lens. *Optics letters* 34, 11 (2009), 1642–1644.

Sheng Liu and Hong Hua. 2010. A Systematic Method for Designing Depth-fused Multi-focal Plane Three-dimensional Displays. *Optics express* 18, 11 (2010), 11562–11573.

Patrick Llull, Noah Bedard, Wanmin Wu, Ivana Tosic, Kathrin Berkner, and Nikhil Balram. 2015. Design and Optimization of a Near-eye Multifocal Display System for Augmented Reality. In *Imaging and Applied Optics*. JTH3A.5.

Gordon D Love, David M Hoffman, Philip JW Hands, James Gao, Andrew K Kirby, and Martin S Banks. 2009. High-speed Switchable Lens Enables the Development of a Volumetric Stereoscopic Display. *Optics express* 17, 18 (2009), 15716–15725.

Kevin J MacKenzie, Ruth A Dickson, and Simon J Watt. 2012. Vergence and Accommodation to Multiple-image-plane Stereoscopic Displays. *Journal of Electronic Imaging* 21, 1 (2012), 011002.

Kevin J MacKenzie, David M Hoffman, and Simon J Watt. 2010. Accommodation to Multiple-focal-plane Displays: Implications for Improving Stereoscopic Displays and for Accommodation Control. *Journal of vision* 10, 8 (2010), 22–22.

Andrew Maimone, Andreas Georgiou, and Joel S. Kollin. 2017. Holographic Near-eye Displays for Virtual and Augmented Reality. *ACM Transactions on Graphics* 36, 4, Article 85 (July 2017), 16 pages. `https://doi.org/10.1145/3072959.3073624`

Andrew Maimone, Gordon Wetzstein, Matthew Hirsch, Douglas Lanman, Ramesh Raskar, and Henry Fuchs. 2013. Focus 3D: Compressive Accommodation Display. *ACM Transactions on Graphics* 32, 5, Article 153 (Oct. 2013), 13 pages. https://doi.org/10.1145/2503144

Aditi Majumder and Michael S Brown. 2007. *Practical multi-projector display design.* AK Peters.

Aditi Majumder, Robert G Brown, and Hussein S El-Ghoroury. 2010. Display gamut reshaping for color emulation and balancing. In *Computer Society Conference on Computer Vision and Pattern Recognition–Workshops.*

Aditi Majumder and Greg Welch. 2001. Computer Graphics Optique. In *Immersive Projection Technology and Virtual Environments.* Springer, 209–217.

Gary Mandle. 2010. OLED; What is it and How Does it Work?. In *SMPTE Tech Conference & Expo.* 1–17.

Markets and Markets. 2015. DLP Projector Market by Light Source (Lamp, LED, and Laser), Chip Model, by Brightness, Throw Distance (Normal Throw, Short Throw, and Ultra-short Throw), Application, and by Geography (The Americas, Europe, APAC, and RoW) - Global Forecast to 2020. https://www.marketsandmarkets.com/PressReleases/dlp-projector.asp.

Manuel Martínez-Corral and Bahram Javidi. 2018. Fundamentals of 3D Imaging and Displays: A Tutorial on Integral Imaging, Light-field, and Plenoptic Systems. *Advances in Optics and Photonics* 10, 3 (2018), 512–566.

Nathan Matsuda, Alexander Fix, and Douglas Lanman. 2017. Focal Surface Displays. *ACM Transactions on Graphics* 36, 4, Article 86 (July 2017), 14 pages. https://doi.org/10.1145/3072959.3073590

Olivier Mercier, Yusufu Sulai, Kevin Mackenzie, Marina Zannoli, James Hillis, Derek Nowrouzezahrai, and Douglas Lanman. 2017. Fast Gaze-contingent Optimal Decompositions for Multifocal Displays. *ACM Transactions on Graphics* 36, 6, Article 237 (Nov. 2017), 15 pages. https://doi.org/10.1145/3130800.3130846

Daniel Miau, Oliver Cossairt, and Shree K Nayar. 2013. Focal sweep videography with deformable optics. In *IEEE Conference on Computational Photography.*

Jurriaan D Mulder. 2005. Realistic Occlusion Effects in Mirror-based Co-located Augmented Reality Systems. In *IEEE Proceedings on Virtual Reality.* 203–208.

Rahul Narain, Rachel A. Albert, Abdullah Bulbul, Gregory J. Ward, Martin S. Banks, and James F. O'Brien. 2015. Optimal Presentation of Imagery with Focus Cues on Multi-plane Displays. *ACM Transactions on Graphics* 34, 4, Article 59 (July 2015), 12 pages. `https://doi.org/10.1145/2766909`

Shree K Nayar and Vijay N Anand. 2007. 3D Display Using Passive Optical Scatterers. *Computer* 7 (2007), 54–63.

Ren Ng. 2005. Fourier Slice Photography. *ACM Transactions on Graphics* 24, 3 (July 2005), 735–744. `https://doi.org/10.1145/1073204.1073256`

Takanori Okoshi. 1980. Three-Dimensional Displays. *Proc. IEEE* 68, 5 (1980), 548–564.

Optotune. 2017. Optotune Electrically Tunable Lens EL-10-30. `http://www.optotune.com/images/products/Optotune`.

Optotune. 2019. Optotune Focus Tunable Lens. `https://www.optotune.com/images/products/Optotune`.

Nitish Padmanaban, Robert Konrad, Tal Stramer, Emily A Cooper, and Gordon Wetzstein. 2017. Optimizing Virtual Reality for All Users Through Gaze-contingent and Adaptive Focus Displays. *Proceedings of the National Academy of Sciences* 114 (2017), 9.

Jui-Wen Pan and Hsiang-Hua Wang. 2013. High contrast ratio prism design in a mini projector. *Applied Optics* 52, 34 (2013), 8347–8354.

Jae-Hyeung Park. 2017. Recent Progress in Computer-Generated Holography for Three-Dimensional Scenes. *Journal of Information Display* 18, 1 (2017), 1–12.

Andriy Pavlovych and Wolfgang Stuerzlinger. 2005. A high-dynamic range projection system. In *Photonics North*. 59692O–59692O.

Yifan Peng, Qiang Fu, Hadi Amata, Shuochen Su, Felix Heide, and Wolfgang Heidrich. 2015. Computational imaging using lightweight diffractive-refractive optics. *Optics Express* 23, 24 (2015), 31393–31407.

Kishore Rathinavel, Hanpeng Wang, Alex Blate, and Henry Fuchs. 2018. An Extended Depth-at-field Volumetric Near-eye Augmented Reality Display. *IEEE Transactions on Visualization and Computer Graphics* 24, 11 (2018), 2857–2866.

Sowmya Ravikumar, Kurt Akeley, and Martin S Banks. 2011. Creating Effective Focus Cues in Multi-plane 3D Displays. *Optics express* 19, 21 (2011), 20940–20952.

Jen-Hao Rick Chang, BVK Vijaya Kumar, and Aswin C Sankaranarayanan. 2016. Arduino code for high bit-depth projection. `https://osapublishing.figshare.com/s/fc0524f9d3ea8a226742`.

Jannick P Rolland, Myron W Krueger, and Alexei Goon. 2000. Multifocal Planes Head-mounted Displays. *Applied Optics* 39, 19 (2000), 3209–3215.

Jannick P. Rolland, Myron W. Krueger, and Alexei A. Goon. 1999. Dynamic Focusing in Head-mounted Displays. *Proceeding of SPIE* 3639 (1999), 3639 – 3639 – 8. `https://doi.org/10.1117/12.349412`

Wilhelm Rollmann. 1853. Notiz zur Stereoskopie. *Annalen der Physik* 165, 6 (1853), 350–351.

Helge Seetzen, Wolfgang Heidrich, Wolfgang Stuerzlinger, Greg Ward, Lorne Whitehead, Matthew Trentacoste, Abhijeet Ghosh, and Andrejs Vorozcovs. 2004. High dynamic range display systems. *ACM Transactions on Graphics* 23, 3 (2004), 760–768.

Shinichi Shiwa, Katsuyuki Omura, and Fumio Kishino. 1996. Proposal for a 3-D Display with Accommodative Compensation: 3DDAC. *Journal of the Society for Information Display* 4, 4 (1996), 255–261.

DE Smalley, E Nygaard, K Squire, J Van Wagoner, J Rasmussen, S Gneiting, K Qaderi, J Goodsell, W Rogers, M Lindsey, *et al*. 2018. A Photophoretic-Trap Volumetric Display. *Nature* 553, 7689 (2018), 486.

Neil R Smith, Don C Abeysinghe, Joseph W Haus, and Jason Heikenfeld. 2006. Agile Wide-angle Beam Steering with Electrowetting Microprisms. *Optics Express* 14, 14 (2006), 6557–6563.

Jung-Yong Son, Vladmir V Saveljev, Yong-Jin Choi, Ji-Eun Bahn, Sung-Kyu Kim, and Hyun-Hee Choi. 2003. Parameters for Designing Autostereoscopic Imaging Systems Based on Lenticular, Parallax Barrier, and Integral Photography Plates. *Optical Engineering* 42, 11 (2003), 3326–3334.

Umberto Spagnolini. 1993. 2-D Phase Unwrapping and Phase Aliasing. *Geophysics* 58, 9 (1993), 1324–1334.

Statista. 2016a. 4K Ultra HD TV Unit Shipments Worldwide From 2013 to 2016. `https://www.statista.com/statistics/422402/4k-ultra-hd-tv-shipments-worldwide/`.

Statista. 2016b. Forecast High Dynamic Range (HDR) TV Shipments Worldwide From 2016 to 2020. https://www.statista.com/statistics/619685/global-hdr-tv-shipments-by-region/.

Toshiaki Sugihara and Tsutomu Miyasato. 1998. System Development of Fatigue-less HMD System 3DDAC (3D Display with Accommodative Compensation: System implementation of Mk. 4 in Lightweight HMD. In *ITE Technical Report 22.1*. The Institute of Image Information and Television Engineers, 33–36.

Qi Sun, Fu-Chung Huang, Joohwan Kim, Li-Yi Wei, David Luebke, and Arie Kaufman. 2017. Perceptually-guided Foveation for Light Field Displays. *ACM Transactions on Graphics* 36, 6, Article 192 (Nov. 2017), 13 pages. https://doi.org/10.1145/3130800.3130807

Nelson V Tabiryan, Svetlana V Serak, David E Roberts, Diane M Steeves, and Brian R Kimball. 2015. Thin Waveplate Lenses of Switchable Focal Length-new Generation in Optics. *Optics express* 23, 20 (2015), 25783–25794.

Texas Instruments. [n.d.]a. Digital light processing technology. http://www.ti.com/lsds/ti/dlp-technology/dlp-technology-home.page.

Texas Instruments. [n.d.]b. DLP system optics application note. http://www.ti.com/lit/an/dlpa022/dlpa022.pdf.

Texas Instruments. [n.d.]c. DLP7000 Data sheet. http://www.ti.com/lit/ds/symlink/dlp7000.pdf.

Texas Instruments. [n.d.]d. DLP9000 Data sheet. http://www.ti.com/lit/ds/symlink/dlp9000.pdf.

Texas Instruments. [n.d.]e. PFET Buck Controller for High Power LED Drivers, LM3409HV. http://www.ti.com/product/LM3409HV.

Varioptic. 2017. Varioptic Variable Focus Liquid Lens ARCTIC 25H. http://varioptic.com/media/cms_page_media/45/MADS_-_160429_-_Arctic_25H_family.pdf.

Vialux. [n.d.]. STAR-07 core optical module. http://www.vialux.de/en/core-optics.html.

Dhanraj Vishwanath and Erik Blaser. 2010. Retinal Blur and the Perception of Egocentric Distance. *Journal of Vision* 10, 10 (2010), 26–26.

Robert Wanat, Josselin Petit, and Rafal Mantiuk. 2012. Physical and perceptual limitations of a projector-based high dynamic range display. In *Theory and Practice of Computer Graphics*. 9–16.

Simon J Watt, Kurt Akeley, Marc O Ernst, and Martin S Banks. 2005. Focus Cues Affect Perceived Depth. *Journal of vision* 5, 10 (2005), 7–7.

Simon J Watt, Kevin J MacKenzie, and Louise Ryan. 2012. Real-world Stereoscopic Performance in Multiple-focal-plane Displays: How Far Apart Should the Image Planes Be?. In *Stereoscopic Displays and Applications XXIII*, Vol. 8288. 82881E.

Gordon Wetzstein, Douglas Lanman, Wolfgang Heidrich, and Ramesh Raskar. 2011. Layered 3D: Tomographic Image Synthesis for Attenuation-based Light Field and High Dynamic Range Displays. *ACM Transactions on Graphics* 30, 4, Article 95 (July 2011), 12 pages. https://doi.org/10.1145/2010324.1964990

Charles Wheatstone. 1838. Contributions to the Physiology of Vision.âĂŤPart the First. on Some Remarkable, and Hitherto Unobserved, Phenomena of Binocular Vision. *Philosophical Transactions of the Royal Society of London* 128 (1838), 371–394.

Lei Xiao, Anton Kaplanyan, Alexander Fix, Matthew Chapman, and Douglas Lanman. 2018. DeepFocus: Learned Image Synthesis for Computational Displays. *ACM Transactions on Graphics* 37, 6, Article 200 (Dec. 2018), 13 pages. https://doi.org/10.1145/3272127.3275032

Marina Zannoli, Gordon D Love, Rahul Narain, and Martin S Banks. 2016. Blur and the Perception of Depth at Occlusions. *Journal of Vision* 16, 6 (2016), 17–17.

Zion Market Research. 2019. Virtual Reality (VR) Market by Hardware and Software for (Consumer, Commercial, Enterprise, Medical, Aerospace and Defense, Automotive, Energy and Others): Global Industry Perspective, Comprehensive Analysis and Forecast, 2016 âĂŞ 2022. https://www.zionmarketresearch.com/report/virtual-reality-market.