Exploiting Randomness in Computational Cameras and Displays

Grace Kuo

Electrical Engineering and Computer Sciences
University of California at Berkeley

Technical Report No. UCB/EECS-2020-218
http://www2.eecs.berkeley.edu/Pubs/TechRpts/2020/EECS-2020-218.html

December 18, 2020

Exploiting Randomness in Computational Cameras and Displays

by

Grace E Kuo

A dissertation submitted in partial satisfaction of the

requirements for the degree of

Doctor of Philosophy

in

Engineering – Electrical Engineering and Computer Sciences

in the

Graduate Division

of the

University of California, Berkeley

Committee in charge:

Associate Professor Laura Waller, Co-chair
Assistant Professor Ren Ng, Co-chair
Associate Professor Hillel Adesnik

Fall 2020

Exploiting Randomness in Computational Cameras and Displays

Abstract

Exploiting Randomness in Computational Cameras and Displays

by

Grace E Kuo

Doctor of Philosophy in Engineering – Electrical Engineering and Computer Sciences

University of California, Berkeley

Associate Professor Laura Waller, Co-chair

Assistant Professor Ren Ng, Co-chair

Despite its desirability, capturing and displaying higher dimensional content is still a novelty since image sensors and display panels are inherently 2D. A popular option is to use scanning mechanisms to sequentially capture 3D data or display content at a variety of depths. This approach is akin to directly measuring (or displaying) the content of interest, which has low computational cost but sacrifices temporal resolution and requires complex physical hardware with moving parts. The exacting specifications on the hardware make it challenging to miniaturize these optical systems for demanding applications such as neural imaging in animals or head-mounted augmented reality displays.

In this dissertation, I propose moving the burden of 3D capture from hardware into computation by replacing the physical scanning mechanisms with a simple static diffuser (a transparent optical element with pseudorandom thickness) and formulating image recovery as an optimization problem. First, I highlight the versatility of the diffuser by showing that it can replace a lens to create an easy-to-assemble, compact camera that is robust to missing pixels; although the raw data is not intelligible by a human, it contains information that we extract with optimization using an efficient physically-based model of the optics. Next, I show that the randomness of the diffuser makes the system well-suited for compressed sensing; we leverage this to recover 3D volumes from a single acquisition of raw data. Additionally, I extend our lensless 3D imaging system to fluorescence microscopy and introduce a new diffuser design with improved noise performance. Finally, I show how incorporating the diffuser in a 3D holographic display expands the field-of-view, and I demonstrate state-of-the-art performance by using perceptually inspired loss functions when optimizing the display panel pattern. These results show how randomness in the optical system in conjunction with optimization-based algorithms can both improve the physical form factor and expand the capabilities of cameras, microscopes, and displays.

To Danny

Contents

List of Figures

Acknowledgments

This thesis is a culmination of work that wouldn't have been possible without a large number of people. First, my advisors, Laura Waller and Ren Ng, have been a huge part of this work. Throughout my years in graduate school, they taught me about many technical topics (for example, how to work in an optics lab, how image sensors work, and how calibration is done in industry), but possibly more importantly, they've taught me how to shape my ideas into complete projects and how to present my work to the broader research community. They've connected me with collaborators, resources, and many opportunities to give seminars and share my work. I'm really grateful to have not one, but two, amazing advisors who value and support me.

In addition to my official advisors, I've been lucky to get mentorship from several others. Nico Pégard supported me making the first random microlens diffuser prototype, connected me with many collaborators in the neuroscience world, and was always happy to explain his ideas to me. Eric Jonas, always incredibly enthusiastic, taught me about compressed sensing and gave lots of life advice about grad school, faculty jobs, and beyond. Andrew Maimone hosted me during my internship at Facebook Reality Labs, and he taught me all about SLMs, holographic displays, and the importance of good image quality.

One of the best things about graduate school was the group of incredible peers that I've had the pleasure of working with. Nick Antipa was critical in designing our original DiffuserCam, and his patience for tuning hyperparamters made all of our results better. Li-hao Yeh taught me to take matrix derivatives and always provided honest and thoughtful feedback, and Zack Phillips helped me create expert automated lab setups. I've also gotten many new insights on diffuser microscopy from talking to Linda Liu and Kryollos Yanny, and our conversations were consistently valuable and rewarding. Kristina Monakhova has constantly inspired my with her innovative uses of machine learning in computational imaging, which I've been lucky to be able to help with, and Pratul Srinivasan was always happy to provide a different perspective on my work.

Many of the people I've met in graduate school have become much more than just peers or collaborators: I've spent countless evenings hanging out with Regina Eckert, Coline Devin, and Ben Mildenhall, and our many food adventures and late night conversations have kept things fun over the years.

I'd also like to thank my parents who instilled in me curiosity about the world, excitement about learning, and an interest in science, math, and engineering. I definitely wouldn't be where I am today without their constant love and support.

Finally, this work wouldn't be possible without my partner Danny, who supported me in choosing Berkeley even though it was across the country, who listened during the challenging times and helped me make difficult decisions, and who's constantly expressed his thoughtfulness, love, kindness, and humor in every action.

Chapter 1

Introduction

From the first photographic system in the mid-nineteenth century to the ubiquitous cell phone cameras of today, the structure of an imaging system has remained relatively constant, relying primarily on a set of lens elements to focus light onto a photosensitive material. Historically, it was necessary that the lens form a good quality image, due to the limited ability to edit film-based photographs after capture. Now, with the advent of the digital image sensor, post-processing has become commonplace and is used regularly for tasks such as distortion correction [58], high dynamic range [40], synthetic depth of field [154], noise removal [21], and low light photography [90]. However, despite the prevalence of digital post-processing, the optical design of most imaging systems still has the same basic elements.

In this work, we dramatically change the conventional architecture: we show that adding a random optical element into the system can enable capture and display of additional content beyond the native abilities of the sensor or display panel. Furthermore, our random optic (a transparent diffuser with pseudorandom thickness) can replace the traditional lens, resulting in a simple and compact system, well-suited for miniaturization. Although the diffuser scrambles the incoming light, we can computationally compensate for the diffuser using optimization algorithms with physically-based models of the system. In this thesis, we demonstrate how the combination of the diffuser with well-designed algorithms can expand the capabilities of cameras, microscopes, and displays.

1.1 Related Work

This work falls into a class of imaging system where information is *indirectly* encoded in the measurement, then computationally extracted. This computational imaging framework has resulted in many new imaging modalities over the years [109], and here, we summarize some key prior work in this area, highlighting places where randomness has been employed for higher dimensional capture, faster acquisition, higher resolution, or reduced system complexity.

Photography: Our work has a lot in common with computational imaging techniques for recovering depth in a photograph. For example, Levin et al. [87] encode depth information by modifying camera hardware with a pseudorandom occluder in the camera aperture, which creates a point spread function that varies with depth, and they recover the image and depth map with sparsity-constrained optimization. Integral photography [91, 66] is another example, in which a microlens array is added in front of a detector plane for both capture and viewing of images from different perspectives. This multi-view capture seeks to record a full 4D light field that contains information about the position and direction of each incoming ray. When recorded digitally (for example, with a microlens array [110] or an array of cameras at different positions [158, 152]), recorded light fields can enable refocusing in post-processing, depth estimation, and extended depth of field. However, in these systems, each pixel on the 2D sensor measures one ray in the 4D light field space, resulting in coarse resolution. To overcome this, Marwah et al. [103] use a quasi-random amplitude mask inside a traditional camera and employ compressed sensing to recover a higher resolution light field. Like these prior works, we use a pseudorandom encoding element and similar image recovery algorithms, but our physical system removes the main lens entirely to create a compact lensless system.

Compact lensless imagers have also been explored previously. Asif et al. [10] used a pseudorandom amplitude mask in place of all other optics, exploiting the randomness to gain form factor, and other systems have used encoding elements such diffractive gratings [140, 53], randomly oriented mirrors [48], Fresnel zone apertures [68], and lenslets separated with baffles to prevent cross talk [144]. However, each of these previous lensless cameras required precise fabrication of the encoding element, and some systems, for example [10], require precise alignment between the mask and the sensor. In contrast, our imaging system uses only an off-the-shelf diffuser, which can even be replaced with everyday items such as scotch tape [7], and the system does not require precise alignment during assembly. We further show that, by using principles of compressed sensing, our system is robust to random pixel erasures and capable of 3D imaging from a single acquisition.

Compressed sensing [25, 42] gained wide-spread attention around 2008, and since then there have been several examples applying the concept to photographic systems. The Rice single-pixel camera [43] uses a micromirror array to sequentially create random projections of a scene which are measured by a single pixel detector; by applying compressed sensing theory, one can reconstruct an image with more pixels than the number of acquisitions. However, the time-cost and physical complexity of the system make it more of an exploratory demonstration than a practical imaging device. An alternative approach is to place an optical mask in the Fourier plane of a $4f$ system to create random projections which are simultaneously measured by a 2D sensor. Romberg [126] and Marcia and Willett [102] both suggest this setup for pixel super-resolution, and the setup also has a lot in common with the coded aperture work of [87]. A key advantage of our design is that it dramatically simplifies the optical setup needed for compressive photography (no $4f$ system or micromirror array) and in Chapter 4 we demonstrate how compressed sensing can be applied for 3D imaging.

Our work is also similar to photographic systems which encode temporal information into a single exposure. Raskar, Agrawal, and Tumblin [125] used a single pseudo-randomly coded shutter to encode linear motion into a single image, and Hitomi et al. [60] expanded

on the idea using per-pixel shutters to encode an arbitrary video. With these cameras, one can also remove motion blur robustly in low light regimes [37]. In this thesis we focus on 3D acquisition rather than video, but Antipa et al. [9] expanded on the lensless camera design presented in Chapter 3 to create a single-shot video method which takes advantage of the built-in rolling shutter resulting in very simple, compact hardware.

Medical Imaging: Although we do not explore any applications in medical imaging in this work, it's worth mentioning since it is a field where indirectly measuring information is critical as it enables non-invasive measurements in the body. For example, in computed tomography (CT), an x-ray beam rotates around a patient, and a detector array collects the projection image at a variety of angles. Similar to our work, this data doesn't look like an image, but can be reconstructed into an internal slice of the body. Since excessive x-ray radiation can be harmful, compressive measurements and reconstruction schemes are useful in this domain to reduce the radiation dose [31].

Magnetic resonance imaging (MRI) is a medical imaging modality that collects samples in Fourier space, rather than directly in image space. MRI is notoriously slow, since samples are collected sequentially and acquisition speed is limited for patient safety; therefore, reducing the number of measurements is desirable. In perhaps the most famous application of compressed sensing, Lustig et al. [99] showed that collecting a random subset of samples and formulating image recovery as an optimization problem enables recover of MRI images with a fraction of the acquisition time. In this thesis, we apply these same principles in optical imaging; however, in the optical domain, samples are generally acquired simultaneously in an array, so instead of changing the sampling pattern, we add randomness by changing the optical design.

Microscopy: Microscopy is filled with imaging tasks that cannot be accomplished with a traditional imaging system and therefore require computational techniques. Some examples include optical super-resolution [129], phase imaging [29], polarization imaging [113], and imaging through scattering media [15]. In this thesis, the microscopy applications are focused primarily on 3D fluorescence imaging in which we seek to determine the location and intensity of fluorophores in a volume. 3D fluorescence can be measured directly with scanning mechanisms using two-photon [41] or light sheet [64] microscopy, but these strategies require complex hardware and face a trade-off between acquisition speed and field-of-view. To capture 3D in a single acquisition, the light distribution must be encoded into the 2D sensor measurement, then computationally decoded. For example, Pavani and Piestun [118] used an additional phase plate in the pupil plane of the microscope to encode depth when the sample is ultra-sparse, and Lu et al. [97] projected a 3D volume into a 2D image with a high resolution Bessel beam, then recovered the sample using temporal priors on a long video. As in photography, the addition of a lenslet array [88, 20, 119] into the system can create a light field microscope, enabling 3D recovery at the expense of either lateral resolution or field-of-view, depending on the design. In Chapter 6 of this thesis, we also use a flat, refractive element (similar to a lenslet array) to project 3D content onto the sensor,

but we do it in a compact package with no microscope objective, and the randomness of our encoding element enables compressed sensing so field-of-view and resolution do not need to be sacrificed.

Displays: Displaying 3D content with accurate focal cues can improve realism and reduce eye fatigue. A brute-force approach is to rapidly move either the sensor or lens (or change the lens focal length) while displaying content at different depths [143]. However, this requires complex moving parts and very good synchronization between the motion and display panel content. Another strategy is a light field display, in which a lenslet array is placed in front of a display panel and each pixel corresponds to a single ray [85]. However, as with image capture, spreading the 2D array of display pixels over a 4D space results in low resolution, much worse than the eye can perceive. Although resolution can be improved by replacing the lenslet array with multiple layers of controllable attenuating masks, diffraction still limits the ultimate resolution [157].

An alternative is a holographic display, which takes advantage of diffraction instead of fighting it. In a holographic display, one uses a display panel which controls the phase of light, and images are formed through interference, enabling 3D or light field content at high resolution without moving part [137]. Although images from these displays often appear grainy with coherent speckle noise, recent prototypes [101] have shown greatly improved image quality. However, holographic displays suffer from a trade-off between the field-of-view and eyebox size, which is the area where the image is viewable. This can be mitigated with a combination of eyetracking and eyebox steering, but these require complex moving parts that are difficult to miniaturize [67, 80].

Random optics were first suggested for this problem by Buckley et al. [22], then built into dynamic display prototypes by Yu et al. [165] and Park, Lee, and Park [116]. However, the algorithms used in these works couldn't generate dense, photorealistic scenes; in Chapter 7 of this thesis, we apply our optimization-based algorithm to this problem and introduce new perceptually inspired loss functions, greatly improving image quality beyond prior work.

1.2 Dissertation Outline

In the rest of this thesis we demonstrate how optical randomness can be exploited in several practical imaging systems spanning photography, microscopy, and displays.

Chapter 2: We begin by providing background that is referenced throughout the rest of the dissertation. We describe the linear model we'll use throughout this work, introduce compressed sensing and the advantages of randomness, and provide details on image recovery using optimization.

Chapter 3: Next, we demonstrate how a random optical element, a diffuser, can replace the traditional camera lens. Here, we introduce DiffuserCam, our easy-to-assemble, compact

lensless camera which uses a pseudorandom diffuser as the only optical element. We propose a simple physically-based convolution model of the system, describe its advantages for both computation and calibration, and explore compressed sensing with DiffuserCam in the context of pixel erasures.

Chapter 4: Without changing the system hardware, we extend DiffuserCam to 3D imaging from a single acquisition with no moving parts. This problem is highly underdetermined and relies heavily on compressed sensing and sparsity priors. To quantify the performance of our system, we introduce a local condition number metric based on compressed sensing theory, which does a better job of characterizing the system than tradition metrics.

Chapter 5: The convolution model used in Chapters 3 and 4 is very efficient but doesn't capture all effects in many systems. In this chapter, we discuss causes of spatial variance and introduce practical algorithms for modeling and calibrating systems that are not convolutional.

Chapter 6: In this chapter we extend our lensless camera to fluorescence microscopy, which requires a spatially varying model. To improve performance in low light conditions, we introduce the random microlens diffuser, an alternative design with better noise performance, and we demonstrate its effectiveness both in simulation and experiment.

Chapter 7: In this chapter, we turn the ideas of DiffuserCam around into a display system. We show that placing a diffuser in front of a holographic display panel can break the trade-off between eyebox and field-of-view that plagues holographic display. We solve for the display panel pattern with the optimization-based approaches used in prior chapters and show that this strategy outperforms prior state-of-the-art.

Chapter 8: Finally, we reflect on themes throughout this dissertation and present possible directions for future work.

Chapter 2

Background

This chapter provides background including an introduction to linear models for optical systems, the theoretical advantages of randomness for compressed sensing, and background on practical algorithms for image recovery.

2.1 Linear Models of Cameras

Rather than independently designing optics and algorithms for imaging systems, in this thesis, we jointly design both elements to improve overall performance. Therefore, we need a mathematical model of the optical system in order to understand how changes to the physical design effect the measurement and algorithm. In this section, we present a linear model describing the physical mapping between the real world object and the sensor response.

For simplicity in describing the model, we initially consider a 2D object a distance z away from the camera, and we extend the model to 3D objects in Chapter 4. We use $\vec{x}$ to denote the coordinates at the object, and $\vec{u}$ to denote the coordinates at the sensor, as shown in Fig. 2.1a. If the object consists of only a single point of unit intensity, we measure a deterministic pattern on the sensor, called the point spread function (PSF), which we denote $h(\vec{u}, \vec{x})$. Note that the PSF is a 2D image that depends on the point source location, making it a function of both the sensor and object coordinates.

What happens when the object is a full scene? We model the object as a collection of independent point sources with varying intensity; therefore, our total sensor measurement is the sum of all the PSFs, with each PSF weighted by the intensity of the corresponding point. If $y(\vec{x})$ is the intensity of the object, then our total sensor measurement, $b(\vec{u})$, is a linear combination of the PSFs as follows:

$$b(\vec{u}) = \sum_{\vec{x}} y(\vec{x})\, h(\vec{u}, \vec{x}). \tag{2.1}$$

Equation 2.1 is linear, and it will be convenient to write it in matrix-vector form. To do this, we discretize the b and y by defining a 2D discrete grid of points in both the sensor and object coordinates; then, we create vectors containing the function values at each of those

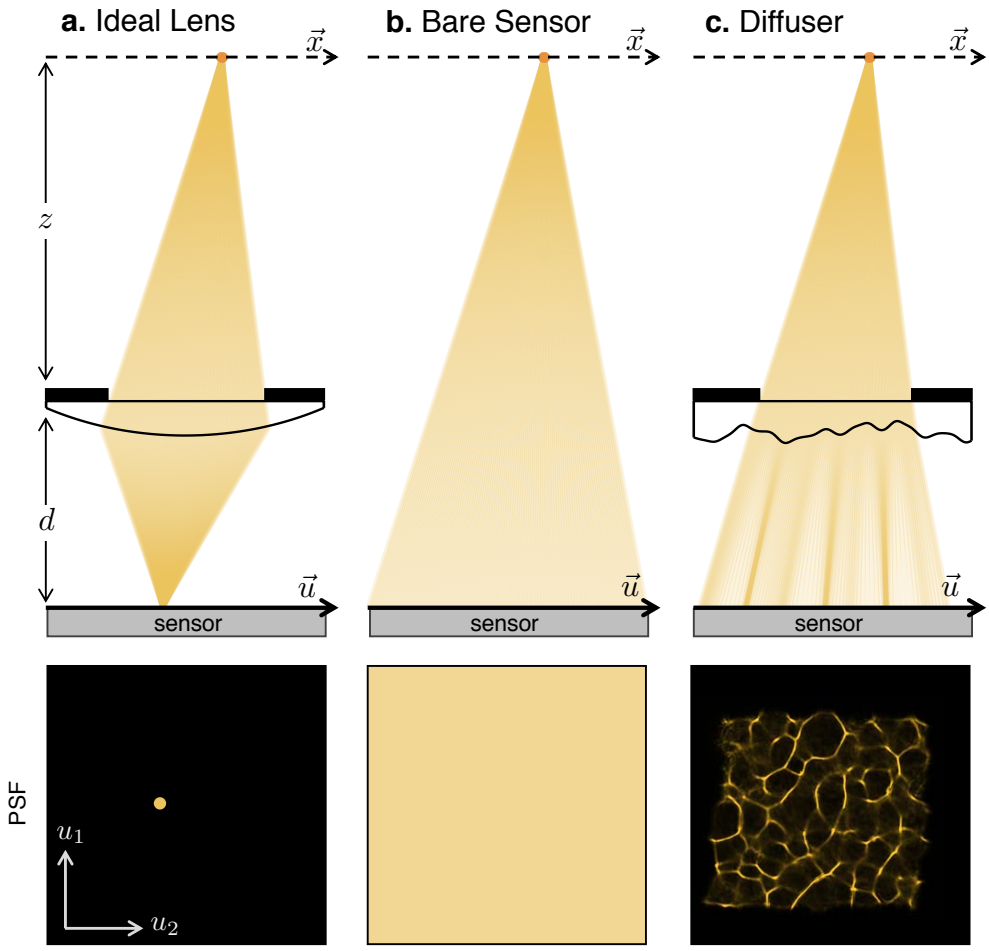

Figure 2.1: Schematic showing three possible camera architectures and their corresponding point spread functions (PSFs). Each architecture corresponds to a different system matrix based on the PSFs produced by the optics.

points. Defining discrete grid points is very natural in the sensor coordinates where we can choose each point to correspond to a pixel location on the sensor. Thus, we let $\vec{\boldsymbol{b}} \in \mathbb{R}^p$ be a vector containing the values of $b(\vec{u})$ at each of the p pixels on the sensor. Defining the discrete grid in object space is more arbitrary, and we'll discuss later how we choose that grid. For now, assume we define n object points, and $\vec{\boldsymbol{y}} \in \mathbb{R}^n$ will be the vector containing the values of $y(\vec{x})$ at those points. Finally, we let $\boldsymbol{A} \in \mathbb{R}^{p \times n}$ be the discretized version of $h(\vec{u}, \vec{x})$. This system matrix that defines the linear mapping between our unknown object $\vec{\boldsymbol{y}}$ and our known sensor measurement $\vec{\boldsymbol{b}}$:

$$\vec{\boldsymbol{b}} = \boldsymbol{A}\vec{\boldsymbol{y}} \tag{2.2}$$

Here, each column of $\boldsymbol{A}$ contains the vectorized PSF from a single point in the world.

This linear matrix model is powerful because we can model a wide range of scenarios and analyze the invertibility of $\boldsymbol{A}$ with well-established techniques from linear algebra. For example, consider a camera with an ideal thin lens, as shown in Fig. 2.1a. Assuming the lens is in focus, the sensor measurement is an image of the world, and $\boldsymbol{A}_{\text{thin lens}} = \boldsymbol{I}$, where $\boldsymbol{I}$ is the identity matrix. We consider this a direct measurement scheme since no computation is needed to recover $\vec{\boldsymbol{y}}$ from the measurement.

Another extreme example is shown in Fig. 2.1b, where the lens is removed entirely and the bare sensor is the only element in the system. In this case, each sensor pixel measures the average intensity of light in the world, which is represented by

$$\boldsymbol{A}_{\text{bare sensor}} \approx \frac{1}{n} \begin{bmatrix} 1 & 1 & \dots & 1 \\ \vdots & & & \vdots \\ 1 & 1 & \dots & 1 \end{bmatrix}.$$

Although it's appealing to have such a simple camera structure with no alignment, $\boldsymbol{A}_{\text{bare sensor}}$ clearly is not invertible since every row is identical and spatial information is lost. In reality, dust and micro-structures on the sensor can create enough variance between different PSFs to make $\boldsymbol{A}$ invertible, but attempts at imaging with a bare sensor [78, 79] have shown limited results: the condition number of the resulting matrix is poor, so the resulting camera requires very controlled lighting conditions and each measurement must be averaged with over 100 acquisitions to reduce noise.

From a matrix-invertibility perspective, the thin lens camera is ideal; however, a camera with no lens has the physical advantages of being thin, lightweight, and simple to manufacture. Figure 2.1c shows our intermediate design, in which we replace the traditional lens with a thin diffuser, a bumpy piece of plastic. (Chapter 3 describes the system in more detail.) Light passing through the diffuser is refracted to form a pseudorandom PSF that changes with point source position. The resulting system matrix $\boldsymbol{A}$ has much better inversion properties than the bare sensor case, and the physical system is substantially simpler than the lens case. In addition, $\boldsymbol{A}$ has elements that are close to random, which gives the matrix good properties for compressed sensing.

2.2 Advantages of Randomness: Compressed Sensing

At first glance, it may appear that an optical system described by an identity matrix, such as an ideal lens, will always have the best inversion properties. However, direct mapping from inputs to outputs in not always ideal when the number of unknowns exceeds the number of measurements, $p < n$. For instance, consider the case where a subset of the camera pixels don't record a measurement; we can model this as deleting rows of $\boldsymbol{A}$ to form a wide matrix representing an undetermined system of linear equations. Another underdetermined case occurs when capturing 3D content from a 2D measurement, and we explore this in depth in Chapter 4.

In these situations, there are infinitely many objects $\vec{\boldsymbol{y}}$ that match the measurement; therefore, traditional linear solvers will fail. However, when the sensing matrix and object

meet certain conditions, *compressed sensing* [42, 24, 25] can be applied to reconstruct an object with more unknowns than the number of measurements. Here we give a brief overview of compressed sensing, the conditions needed on $\boldsymbol{A}$ and $\vec{\boldsymbol{y}}$, and common techniques for recovering the object.

Sparsity and Incoherence

Solving for more components in $\vec{\boldsymbol{y}}$ than the number of measurements is not uniquely defined without further constraints. Compressed sensing specifically seeks to solve underdetermined linear problems when the unknown vector $\vec{\boldsymbol{y}}$ is *sparse*, meaning that most of the elements in $\vec{\boldsymbol{y}}$ are zero. The intuition is that, instead of needing to recover n unknown values (when $n > p$), one only needs to find a small number of non-zero coefficients and their locations in the vector; the remaining elements are zero. When the number of non-zero coefficients is much smaller than p, the problem intuitively seems much more tractable.

However, if compressed sensing could only recover sparse images, where most pixels are black (or zero), then its applications would be rather limited. Luckily, $\vec{\boldsymbol{y}}$ does not need to be sparse itself as long as there is a predetermined linear transformation $\boldsymbol{\Psi}$ such that $\boldsymbol{\Psi}\vec{\boldsymbol{y}}$ is sparse. For example, a wavelet transformation results in images that are almost sparse, with most coefficients close to zero. Perhaps the most commonly used transformation is termed *Total Variation* [153] in which $\boldsymbol{\Psi} = \begin{bmatrix}\nabla_1 \dots \nabla_k\end{bmatrix}^T$ where ∇_i is the finite difference operator along the i^{th} spatial dimension of the signal. (For example, $k = 2$ when solving for a 2D photograph, $k = 3$ when solving for a 3D volume, etc.) Total variation assumes that the spatial gradient of most images is sparse, in other words, that images are composed of regions of constant intensity broken up by relatively few sharp edges.

In addition to a sparse signal in some domain, its also necessary for the system matrix $\boldsymbol{A}$ to be *incoherent*. Formally, the *coherence* of a matrix is

$$\max_{i \neq j} \left| \left\langle \boldsymbol{A}_i, \boldsymbol{A}_j \right\rangle \right|, \tag{2.3}$$

where $\boldsymbol{A}_i$ is the i^{th} column of $\boldsymbol{A}$, and $\langle \cdot, \cdot \rangle$ denotes the inner product. For compressed sensing, we want the matrix coherence to be low, meaning that any two columns should be close to orthogonal.[1] Note, that the columns cannot all be exactly orthogonal since the number of columns exceeds the number of rows. With high probability two random vectors are close to orthogonal, so it follows that a good system matrix for compressed sensing consists of randomly generated entries. Using the diffuser in the optical system is one way to achieve this physically. However, it is important to note that our system matrices generally represent

[1] More rigorously, for compressed sensing we want the matrix $\boldsymbol{A}\boldsymbol{\Psi}^{-1}$ to have low matrix coherence, to account for the sparsifying transform. However, in practice we frequently choose $\boldsymbol{\Psi}$ to be total variation, which is non-invertible, making it challenging to analyze this scenario. Therefore, we we restrict our analysis to the incoherence of $\boldsymbol{A}$ which is fully accurate when the image $\vec{\boldsymbol{y}}$ is natively sparse (i.e. $\boldsymbol{\Psi} = \boldsymbol{I}$). Furthermore, Candès and Wakin [25] show that if $\boldsymbol{A}$ is random, then $\boldsymbol{A}$ is likely to be incoherent with any fixed orthonormal basis $\boldsymbol{\Psi}$.

light intensity, as described in Sec. 2.1, so they will only have positive entries since intensity cannot be negative. In Chapter 6 we consider this limitation in more depth and discuss how it informs a different diffuser design.

Signal Recovery

Given an incoherent system matrix $\boldsymbol{A}$ and sparsifying transform $\boldsymbol{\Psi}$, we would like to solve for the sparest vector $\boldsymbol{\Psi}\vec{\boldsymbol{y}}$ that matches the data. The is equivalent to minimizing the ℓ_0 norm of $\boldsymbol{\Psi}\vec{\boldsymbol{y}}$ while maintaining data consistency. However, minimizing the ℓ_0 norm is NP-hard. In 2008, Donoho [42] showed that when the matrix is sufficiently incoherent and the vector is sufficiently sparse, then minimizing the ℓ_1 norm instead gives and accurate reconstruction. In the presence of noise, we can therefore recover the unknown vector by solving the following convex problem:

$$\underset{\vec{\boldsymbol{y}}}{\operatorname{argmin}} \|\boldsymbol{\Psi}\vec{\boldsymbol{y}}\|_1 \quad \text{subject to} \quad \tfrac{1}{2}\|\boldsymbol{A}\vec{\boldsymbol{y}} - \vec{\boldsymbol{b}}\|_2^2 \leq \epsilon, \tag{2.4}$$

where ϵ is based on the quantity of noise in the measurement. This problem can can be written in Lagrangian form as

$$\underset{\vec{\boldsymbol{y}}}{\operatorname{argmin}} \tfrac{1}{2}\|\boldsymbol{A}\vec{\boldsymbol{y}} - \vec{\boldsymbol{b}}\|_2^2 + \tau\|\boldsymbol{\Psi}\vec{\boldsymbol{y}}\|_1, \tag{2.5}$$

where τ is a tuning parameter that determines the sparsity of the signal. We'll refer to the first term of the loss function as the data fidelity term, denoted $\mathcal{L}_{\text{data}}$, and we'll refer to the second term as the regularization term, denoted $\mathcal{L}_{\text{reg}}$. In the next section, we discuss a practical algorithm for solving this minimization problem at image scales.

2.3 Inverse Problem: Recovering the Image from the Measurement

Given a known system matrix $\boldsymbol{A}$ and sensor measurement $\vec{\boldsymbol{b}}$, we'd like to recover the unknown object $\vec{\boldsymbol{y}}$ by solving Eq. 2.5. In general, this problem has no closed-form solution, and therefore must be solved with iterative techniques. Luckily, since Eq. 2.5 is convex, we can initialize an iterative algorithm anywhere and still converge to a good solution. However, $\vec{\boldsymbol{y}}$ represents an image and therefore contains many elements ($n \approx 10^6$). As a result, interior point methods and second-order methods that require computing the Hessian quickly become computationally intractable. Instead, for imaging problems, first-order methods that only require the gradient are desirable.

Iterative Shrinkage and Thresholding Algorithm (ISTA)

A simple and popular approach is the *Iterative Shrinkage and Thresholding Algorithm*, abbreviated ISTA [39], in which the estimate of $\vec{\boldsymbol{y}}$ is iteratively updated by first taking a

gradient step based on the data fidelity term, then applying a soft-thresholding operator in the sparse domain:

$$\vec{\boldsymbol{y}}_{k+1} \leftarrow \mathcal{T}_{\alpha,\boldsymbol{\Psi}}\big(\vec{\boldsymbol{y}}_k - \mu\nabla\mathcal{L}_{\text{data}}\big). \tag{2.6}$$

Here $\vec{\boldsymbol{y}}_k$ is the best estimate of $\vec{\boldsymbol{y}}$ at iteration k, μ is the step size, and $\nabla\mathcal{L}_{\text{data}}$ is the gradient of the data fidelity term with respect to $\vec{\boldsymbol{y}}$. $\mathcal{T}_{\alpha,\boldsymbol{\Psi}}$ represents the soft-thresholding in the (invertible) domain $\boldsymbol{\Psi}$,

$$\mathcal{T}_{\alpha,\boldsymbol{\Psi}}(\vec{\boldsymbol{y}}) = \boldsymbol{\Psi}^{-1}\mathcal{T}_{\alpha}(\boldsymbol{\Psi}\vec{\boldsymbol{y}}), \tag{2.7}$$

where the soft-thresholding operator can be defined element-wise as

$$\mathcal{T}_{\alpha}(\vec{\boldsymbol{y}})_i = \begin{cases} y_i - \alpha, & \text{if } y_i > \alpha \\ 0, & \text{if } -\alpha \geq y_i \leq \alpha \\ y_i + \alpha, & \text{if } y_i < -\alpha. \end{cases} \tag{2.8}$$

Here, α is a tuning parameter related to τ in Eq. 2.5.

To implement total variation sparsity, in which $\boldsymbol{\Psi}$ is not invertible, we capitalize on the connection between total variation and Haar-wavelet shrinkage. Here we apply the soft-thresholding operator in the invertible Haar-wavelet domain on several integer shifts of the signal, as described by Kamilov, Bostan, and Unser [71]. Details of this method are included in the Appendix A.1.

For most imaging problems, the vector $\vec{\boldsymbol{y}}$ represents the intensities of an object or scene, and we know that physically none of the intensities can be negative. Therefore, we frequently wish to apply the additional constraint that $y_i \geq 0$ for all i. When this is the case, we can project $\vec{\boldsymbol{y}}_{k+1}$ onto the non-negative set after soft-thresholding by setting all negative elements to zero.

The last piece needed to completely define the update step is $\nabla\mathcal{L}_{\text{data}}$, the gradient of the data fidelity term with respect to $\vec{\boldsymbol{y}}$. This can be derived using vector calculus from the definition of $\mathcal{L}_{\text{data}}$ as follows:

$$\begin{aligned} \mathcal{L}_{\text{data}} =& \tfrac{1}{2}\|\boldsymbol{A}\vec{\boldsymbol{y}} - \vec{\boldsymbol{b}}\|_2^2 \\ =& \tfrac{1}{2}(\boldsymbol{A}\vec{\boldsymbol{y}} - \vec{\boldsymbol{b}})^T(\boldsymbol{A}\vec{\boldsymbol{y}} - \vec{\boldsymbol{b}}) \\ =& \tfrac{1}{2}(\vec{\boldsymbol{y}}^T\boldsymbol{A}^T\boldsymbol{A}\vec{\boldsymbol{y}} - 2\vec{\boldsymbol{b}}^T\boldsymbol{A}\vec{\boldsymbol{y}} + \vec{\boldsymbol{b}}^T\vec{\boldsymbol{b}}) \end{aligned}$$

$$\begin{aligned} \nabla\mathcal{L}_{\text{data}} =& \boldsymbol{A}^T\boldsymbol{A}\vec{\boldsymbol{y}} - \boldsymbol{A}^T\vec{\boldsymbol{b}} \\ =& \boldsymbol{A}^T(\boldsymbol{A}\vec{\boldsymbol{y}} - \vec{\boldsymbol{b}}) \end{aligned} \tag{2.9}$$

We can think of the gradient calculation as a two-step process where, first, we compute the error, $\boldsymbol{A}\vec{\boldsymbol{y}} - \vec{\boldsymbol{b}}$, and then we pass the error through the operator $\boldsymbol{A}^T$, which is sometimes referred to as the *adjoint*. Note that, if $\boldsymbol{A}$ is complex valued, than all of the transpose operations should also take the complex conjugate.

Fast Iterative Shrinkage-Thresholding Algorithm (FISTA)

ISTA is easy to implement with very few tuning parameters, but it has relatively slow convergence, $O(1/k)$. Throughout this work, we use an equally simple but faster algorithm with convergence $O(1/k^2)$ called the *Fast Iterative Shrinkage-Thresholding Algorithm* (FISTA) [12]. This algorithm follows the same structure as ISTA with first a gradient step based on the data fidelity term, then a soft-thresholding operation. However, FISTA includes an additional step, sometimes considered a *momentum* term, shown in the last two lines in the complete algorithm below.

ALGORITHM 1: Fast Iterative Shrinkage-Thresholding Algorithm (FISTA)

Inputs: $\boldsymbol{A}$ (system matrix), $\vec{\boldsymbol{b}}$ (measurement), μ (step size), α (sparsity parameter)
Initialize: $\vec{\boldsymbol{y}}_0 = \vec{\boldsymbol{0}}, \quad t_0 = 1, \quad \vec{\boldsymbol{y}}'_0, = \vec{\boldsymbol{y}}_0$
Repeat:

$$\nabla \mathcal{L}_{\text{data}} \leftarrow \boldsymbol{A}^T(\boldsymbol{A}\vec{\boldsymbol{y}}_k - \vec{\boldsymbol{b}})$$
$$\vec{\boldsymbol{y}}'_{k+1} \leftarrow \mathcal{T}_\alpha\big(\vec{\boldsymbol{y}}_k - \mu \nabla \mathcal{L}_{\text{data}}\big)$$
$$t_{k+1} \leftarrow \frac{1+\sqrt{1+4t_k^2}}{2}$$
$$\vec{\boldsymbol{y}}_{k+1} \leftarrow \vec{\boldsymbol{y}}'_k + \frac{t_k - 1}{t_{k+1}}\left(\vec{\boldsymbol{y}}'_{k+1} - \vec{\boldsymbol{y}}'_k\right)$$

The intuition of FISTA's momentum term is that it keeps track of both our current estimate of $\vec{\boldsymbol{y}}$ as well as our estimate from the previous iteration. We use the difference between the two estimates to smooth out the oscillations between gradient steps, improving converge speed.

Efficient Computation of $\boldsymbol{A}$ and $\boldsymbol{A}^T$

An final challenge is that, in practice, the vectors $\vec{\boldsymbol{b}}$ and $\vec{\boldsymbol{y}}$ each contain on the order of a million elements for a megapixel sensor, making it computationally impractical to explicitly store or do matrix multiplication with $\boldsymbol{A}$. Therefore, throughout this thesis, we approximate $\boldsymbol{A}$ with a variety of operations that can be performed on an image without actually instantiating a large matrix. Although we avoid instantiating $\boldsymbol{A}$, it is very helpful to have a matrix representation of all operators since it enables easy computation of $\boldsymbol{A}^T$ which is necessary for the gradient step described previously. We summarize several common operations and their matrix representations in Table 2.1.

Table 2.1: Summary of common image operations and their corresponding matrix operators. Note that the matrix operators are never instantiated, instead the operations are implemented on the 2D image directly.

Operation	**Matrix Description**	**Symbol**
2D Fourier transform	2D discrete Fourier transform (DFT) matrix, unitary	$\boldsymbol{F}$
2D inverse Fourier transform	2D inverse DFT matrix, unitary	$\boldsymbol{F}^{-1} = \boldsymbol{F}^{H}$
Point-wise multiplication with $\vec{\boldsymbol{v}}$	Square matrix with $\vec{\boldsymbol{v}}$ along the diagonal	$\text{diag}(\vec{\boldsymbol{v}})$
Crop	Identity matrix with rows corresponding to cropped pixels removed (wide matrix)	$\boldsymbol{C}$
Pad with zeros	Identity matrix with additional rows of zeros added corresponding to locations of new pixels	$\boldsymbol{P} = \boldsymbol{C}^{T}$
Convolution with $\vec{\boldsymbol{h}}$ (with circular boundary conditions)	Square Toeplitz matrix, which can be implemented as point-wise multiplication in Fourier Space	$\boldsymbol{H} = \boldsymbol{F}^{-1}\,\text{diag}(\boldsymbol{F}\vec{\boldsymbol{h}})\boldsymbol{F}$

2.4 Imaging in Reverse: Computational Displays

So far, we've discussed how randomness is advantageous in camera systems, in which we capture a sensor measurement and use a model of the camera to reconstruct an unknown object or scene. In this section, we discuss how we can reverse the ideas presented above and apply them to display systems to create computational displays that exploit randomness. As depicted in Fig. 2.2, camera and display systems are duals of each other: instead of an unknown object and known sensor measurement, in a display, we have a known object (a target image) and an unknown display panel. Given a model of the optical system, we aim to solve for the display panel pattern that best creates the target image after light passes through the system optics.

We use our system matrix $\boldsymbol{A}$ to represent the optical system in the direction of the light's propagation: in the case of a camera, this was from the object to the sensor. In the case of a

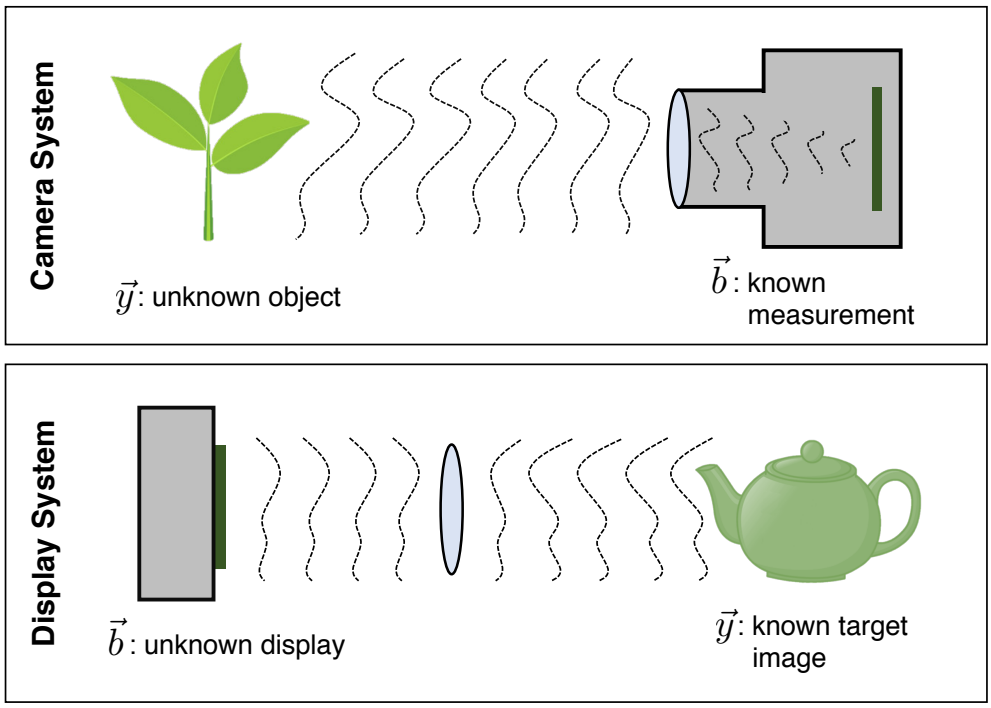

Figure 2.2: Computational cameras and displays are duals of each other. In a camera system, we attempt to reconstruct an unknown object from the known sensor measurement. In a display system, the object is known, since it is a target image that we want the user to see. We attempt to solve for the pattern to place on the display panel to best create the known target image.

display, it's from the display panel to the virtual object. Therefore, we'll represent a linear display as

$$\boldsymbol{A}\vec{\boldsymbol{b}} = \vec{\boldsymbol{y}}. \tag{2.10}$$

By using a linear mapping, we've implicitly assumed that each pixel on our display panel acts independently, and our displayed image is a linear combination of the contributions from each pixel. This is a good assumption for something like an LCD or OLED display. Later, in Chapter 7 we'll discuss how we can model the electric field, rather than intensity, to use a linear model for for propagation through a coherent holographic system.

In a camera system, we frequently want to reconstruct more data than measurements, resulting in the challenging case of an underdetermined linear system. In contrast, in a display system, if $\boldsymbol{A}$ is underdetermined this means there are multiple $\vec{\boldsymbol{b}}$ that create the same output image $\vec{\boldsymbol{y}}$, which isn't a problem at all! This is because any solution that creates the target $\vec{\boldsymbol{y}}$ is equally good, since the user will see the desired picture.

The more challenging case in a display system is when we wish to display content with more degrees of freedom than the number of pixels on the display panel. In this case, $\vec{\boldsymbol{y}}$ has

more elements than $\vec{\boldsymbol{b}}$ and $\boldsymbol{A}$ has more columns than rows resulting in an overdetermined system. Here, it may be impossible to accurately reproduce the target image, and instead we'd like to solve for the display pattern that achieves the closest image possible.

This leads to the following question: What properties do we want in $\boldsymbol{A}$ so our displayed images are similar to the target for a wide range content? Let p be the number of pixels on our display panel and n be the number of pixels in each target image. We'll consider the underdetermined case where $p < n$ and $\boldsymbol{A} \in \mathbb{R}^{p \times n}$ is a tall matrix. Suppose we have a collection of k target images that we wish to display where $\vec{\boldsymbol{y}}_i$ denotes the i^{th} target image. Each target image has an associated optimal display pattern $\vec{\boldsymbol{b}}_i$ that best recreates the target. Let $\boldsymbol{Y} \in \mathbb{R}^{p \times k}$ be the matrix of all target images, $\boldsymbol{Y} = \left[\vec{\boldsymbol{y}}_1 \dots \vec{\boldsymbol{y}}_k\right]$, and let $\boldsymbol{B} \in \mathbb{R}^{n \times k}$ be the matrix of all the associated displayed patterns, $\boldsymbol{B} = \left[\vec{\boldsymbol{b}}_1 \dots \vec{\boldsymbol{b}}_k\right]$.

We would like to find the optimal system matrix $\boldsymbol{A}$ that will best display the collection of target images. However, the optimal displayed patterns $\boldsymbol{B}$ depend on the system matrix, so finding the optimal system requires finding both $\boldsymbol{A}$ and $\boldsymbol{B}$ as by solving

$$\hat{\boldsymbol{A}}, \hat{\boldsymbol{B}} = \underset{\boldsymbol{A},\boldsymbol{B}}{\operatorname{argmin}} \|\boldsymbol{A}\boldsymbol{B} - \boldsymbol{Y}\|_F \tag{2.11}$$

where $\|\cdot\|_F$ is the Frobenius norm. Solving Eq. 2.11 is equivalent to finding the best rank-p approximation of $\boldsymbol{Y}$ in an ℓ_2 sense, which can be solved exactly with principle component analysis (PCA) by computing the singular value decomposition (SVD) of $\boldsymbol{Y}$ and keeping the singular vectors associated with the largest p singular values. The resulting $\hat{\boldsymbol{A}}$ represents the optimal system matrix for the input collection of images.

Unfortunately, this method of system design is impractical for several reasons. First, the collection of target images, $\boldsymbol{Y}$, can be very large, making PCA impractical due to its high computational complexity. Second, even if one finds the optimal system matrix $\hat{\boldsymbol{A}}$, one would need to determine how to physically implement this matrix in an optical system. Finally, the system may perform poorly if one attempts to display a new target image with different statistics from $\boldsymbol{Y}$, since the calculated matrix is optimal only over the input targets.

To overcome these problems, we once again turn to randomness as a computationally tractable solution that is agnostic to the image content and can be well approximated with simple optical hardware. Random projections have been proposed as a computationally efficient alternative to PCA for dimensionality reduction [74, 16] and classification/clustering tasks [65, 49]. In these works, $\hat{\boldsymbol{A}}$ in Eq. 2.11 is approximated by a random matrix, and $\hat{\boldsymbol{B}}$ is estimated as $\hat{\boldsymbol{A}}^T\boldsymbol{Y}$. The success of these techniques suggest that a random system matrix is a good choice for a display in which we want to create content with more degrees of freedom than the number of controllable pixels. In Chapter 7, we demonstrate a computational display system with a random system matrix and show how it can be used to create imagery with larger field-of-view than natively supported by the display panel.

2.5 Limitations of the Linear Model

Throughout this work, we model camera and display optics as a linear system, as described above. However, there are some key limitations to this model that we expand on here:

Self-Occlusions: In Chapter 4 we expand the 2D camera model introduced here to 3D, in which point sources representing the object can exist at multiple depths. In this situation, a point source towards the front of the volume could partially block light coming from a point source at the back of the volume. When the existence of a point source changes the PSF of a *different* point, it breaks the assumption that all points act independently. Note that a point source which is *fully* occluded is not an issue, since this is equivalent to the point having zero intensity, which matches the model. The challenge occurs when the PSF is changed by other points, but not fully removed (for example, if half of the PSF is blocked).

Specularities and Angle-Dependent Effects: The linear model assumes we can break an image or volume into a collection of isotropically emitting point sources. Therefore, points that emit or reflect light non-isotropically, such as specular highlights that change dramatically with angle, may not be captured well with this model.

Optical Interference and Partial Coherence: In addition, we assume that all points are optically incoherent[2] with each other; that is, there are no wave interference effects when light from two different points in the scene interact. The linear model can be extended to coherent illumination by creating a model that describes the complex electric field, rather than intensity [123]. However, these *transmission matrices* still do not account for partial coherence, which is when there are several set of points that each interfere with themselves but not with each other (e.g. several different colored lasers).

[2]Optical coherence/incoherence, which is a property of the wave-nature of light, should not be confused with matrix coherence/incoherence (described in Sec. 2.2), which describes how good a matrix is for compressed sensing.

Chapter 3

DiffuserCam: A Diffuser-Based Lensless Camera

In this chapter[1], we introduce a compact and easy-to-build lensless camera, called DiffuserCam, in which we replace the lens of a traditional camera with a diffuser. Unlike a standard lens, the diffuser does not directly produce an image on the sensor. Instead it indirectly encodes the object intensities, and we algorithmically recover the image from the sensor data. As described in Chapter 2, we model our computational camera as a matrix, and recover the object by solving an optimization problem. However, it's computationally impractical to calibrate and store the entire system matrix, so we introduce a physically-based *convolution model* that leads to a simple calibration scheme with efficient computation.

Replacing the lens with a diffuser has several advantages: The resulting system is low-cost, lightweight, and has potential for scaling to larger sensor formats. Assembly is easy and doesn't require precise alignment. Finally, unlike a traditional lens, the system matrix of DiffuserCam is pseudorandom, enabling recovery of more object pixels than measurements through compressed sensing.

3.1 Methods

The DiffuserCam architecture is simple: the only optical element is a diffuser (a pseudo-random phase mask) placed a small distance in front of a traditional image sensor. The diffuser is a piece of clear polymer with a smooth, slowly-varying surface, and when illuminated by a point source, light refracts at the diffuser's surface based on Snell's law. We set the distance between the diffuser and sensor, d, so that the convex bumps on the diffuser concentrate rays to create a high contrast caustic pattern at the sensor, shown in Fig. 3.1. An aperture physically constrains the PSF to have support smaller than the sensor size. In

[1]This chapter is in part based on the published conference paper titled "DiffuserCam: Diffuser-Based Lensless Cameras" and is joint work with Nick Antipa, Ren Ng, and Laura Waller [82].

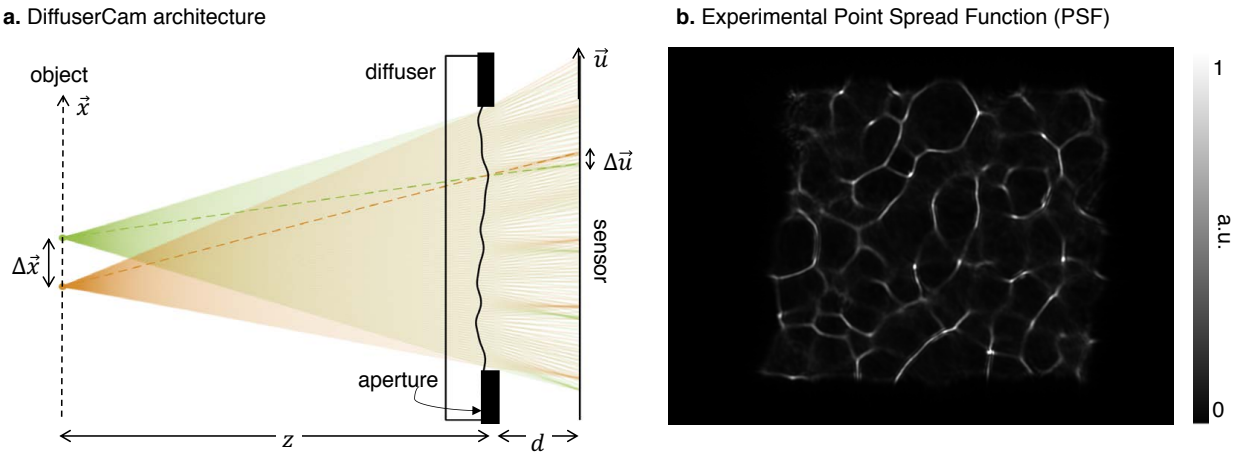

Figure 3.1: (a) DiffuserCam architecture. Bumps on the diffuser surface cause rays of light to bend inward, creating a high contrast point spread function (PSF), shown in (b). When the angle of incidence at the diffuser is small, a lateral translation of the point source causes a corresponding translation of the PSF. We use this *shift-invariance* to create an efficient convolutional model of the system.

combination with the convolution model presented next, the aperture enables calibration from a single image.

Convolution Model

Next we derive our convolution model which efficiently describes the system. The diffuser surface is slowly varying with low angles (around 0.5°), and when the point source is far from the camera (compared to the sensor) the incident angle of incoming rays is close to normal. In this regime, we can apply the small angle approximation ($\sin\theta \approx \theta$) to Snell's law, resulting in a linear model. In this small-angle (*paraxial*) regime, a lateral translation of the point source by $\Delta\vec{x}$ does not change the PSF structure; it merely translates the PSF by $\Delta\vec{u}$, as shown in Fig. 3.1. Based on the system geometry,

$$\frac{\Delta\vec{u}}{\Delta\vec{x}} = \frac{d}{z} = m, \tag{3.1}$$

where z is the distance between the object and the sensor. We refer to m as the paraxial magnification.

Using this shift-invariance property, we can determine the PSF at any location in the 2D field-of-view from only a single image of the on-axis PSF. Let $h_0(\vec{u}) = h(\vec{u}, \vec{x} = \vec{0})$ be the on-axis PSF. Then, the PSF from an arbitrary location can be written as,

$$h(\vec{u}, \vec{x}) = h_0(\vec{u} + m\vec{x}). \tag{3.2}$$

Plugging this into Eq. 2.1 yields the following expression for the sensor response, $b(\vec{u})$:

$$\begin{aligned} b(\vec{u}) &= \sum_{\vec{x}} y(\vec{x})\, h_0(\vec{u} + m\vec{x}) \\ &= y(\tfrac{\vec{x}}{m}) * h_0(\vec{u}) \end{aligned} \tag{3.3}$$

where $*$ represents a 2D linear convolution. In summary, the sensor response can be modeled as a 2D convolution between the object intensities and the on-axis PSF, which we'll calibrate by directly measuring a single image of a point source.

However, there are still a few details needed for practical implementation; specifically, we need to discretize Eq. 3.3. As discussed in Chapter 2, it is very natural to discretize $b(\vec{u})$ at the pixel resolution of the sensor, and we let $\vec{\boldsymbol{b}}$ represent the resulting vectorized sensor response. Eq. 3.1 suggests a natural discretization of $\vec{\boldsymbol{y}}$: if Δp is the pixel size at the sensor, then we should sample the object coordinates at a spacing of $\frac{1}{m}\Delta p$. With this sampling, a translation of one pixel at the sensor is equivalent to a translation of one sample at the object.[2]

For computational efficiency, we would like to implement Eq. 3.3 in the Fourier domain by taking the fast Fourier transform (FFT) of both y and h_0, multiplying them together, and then taking the inverse FFT. However, FFT-based convolutions have circular boundary conditions, in which content at the edge of the field-of-view "wraps around" to the other side, shown in Fig. 3.2a. Since this effect does not happen physically in the camera, it causes model mismatch.

We eliminate the non-physical effects of the circular boundary conditions by padding the PSF to twice the sensor size (denoted $\boldsymbol{P}\vec{\boldsymbol{h}}_{\boldsymbol{0}}$ where $\boldsymbol{P}$ is the pad operator) and increasing the extent of the object $\vec{\boldsymbol{y}}$. We convolve these larger vectors and then crop the result to the original sensor size, as depicted in Fig. 3.2b. The result is the same as convolution without the circular boundary conditions and can be implemented efficiently with FFT-based convolutions.

Notice that points outside the original field-of-view (shown by the dashed box in Fig. 3.2b) can contribute to the measurement since a portion of the PSF will still appear on the sensor. Therefore, when solving for the object, we let $\vec{\boldsymbol{y}}$ be larger than the sensor size so that it can explain all parts of the sensor measurement. The physical aperture on the diffuser (Fig. 3.1a) constrains the PSF to be smaller than the sensor size, so the maximum size of $\vec{\boldsymbol{y}}$ is twice the sensor size in each direction.

Putting this all together, we can write the complete forward model as follows:

$$\begin{aligned} \vec{\boldsymbol{b}} &= \boldsymbol{A}\vec{\boldsymbol{y}} \\ &= \boldsymbol{C}\boldsymbol{F}^{-1}\,\mathrm{diag}(\boldsymbol{F}\boldsymbol{P}\vec{\boldsymbol{h}}_{\boldsymbol{0}})\boldsymbol{F}\vec{\boldsymbol{y}} \end{aligned} \tag{3.4}$$

where $\boldsymbol{C}$ is an operator cropping to the sensor size, $\boldsymbol{P}$ is the pad operator that pads the image to $2\times$ the sensor size in each direction, and $\boldsymbol{F}$ and $\boldsymbol{F}^{-1}$ are the forward and inverse

[2] When the object is far away, it makes more sense to represent the object as a function of angle. In this case, we sample the object every $\Delta\alpha = \tan^{-1}(\frac{\Delta p}{z})$

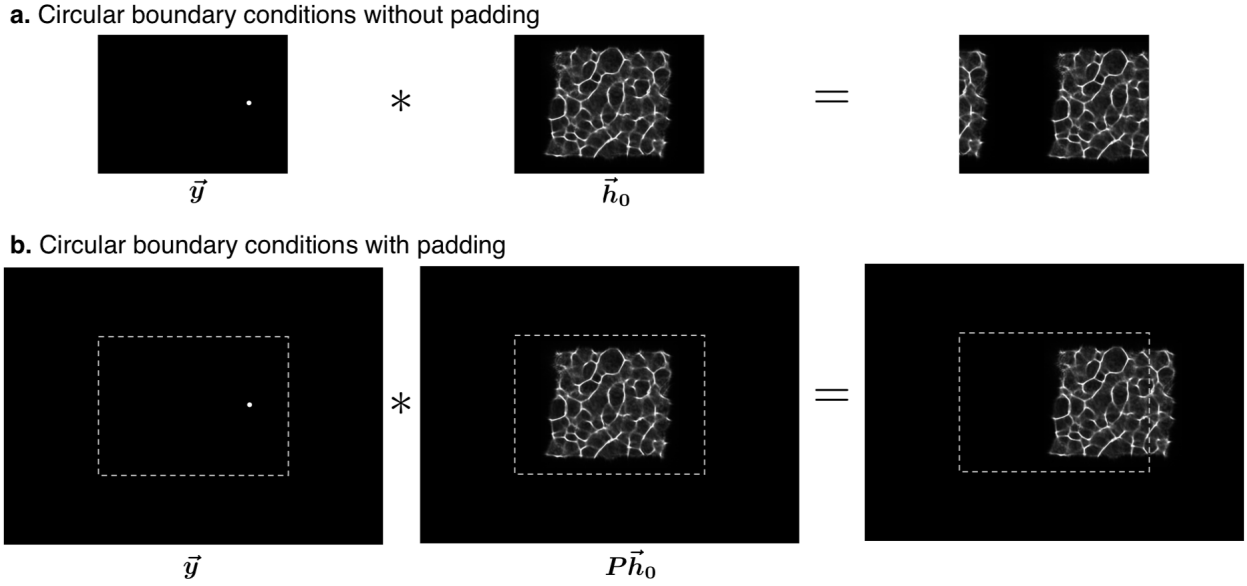

Figure 3.2: Padding the object and PSF before convolving removes the non-physical effects of circular boundary conditions. (a) Without padding, the circular boundary conditions cause the PSF to wrap around to the other side of the image, which does not happen in the physical system, causing model-mismatch. (b) We pad the PSF and increase the extent of the object to give the effect of convolution without circular boundary conditions. After the convolution, the result must be cropped back to the original sensor size (dashed box).

Fourier transform matrices, respectively. If `h` and `y` are 2D matrices, we can easily implement the forward model with the following MATLAB code:

```
b = crop(ifftshift(ifft2(fft2(pad(h)).*fft2(y))))
```
(3.5)

Calibration and Image Reconstruction

The only calibration needed for Eq. 3.4 is the on-axis PSF, $\vec{\boldsymbol{h}}_{\mathbf{0}}$. We can directly measure the PSF by collecting a single image of a point source, such as an LED, placed a distance z from the sensor. The physical aperture on the diffuser constrains the PSF extent so the entire PSF can be captured in one acquisition. (Without the aperture, lateral shifts of a point source could cause new parts of the PSF to hit the sensor that were not captured in the on-axis measurement, necessitating a more complex calibration procedure.) The PSF exhibits slight variations with depth, so they system works best if the calibration source is at approximately the same depth as the objects to be photographed. However, if the calibration PSF is taken at a different depth, we can use a paraxial model of the depth-varying PSF to digitally refocus the PSF after capture.

After calibration, we can use DiffuserCam for photography. We capture the raw data, $\vec{\boldsymbol{b}}$, and then recover the image by solving the regularized least-squares problem from Eq. 2.5

using FISTA, presented in Algorithm 1. FISTA begins with an estimate of the scene, and each iteration requires computing the error, $\vec{\boldsymbol{e}}$,

$$\vec{\boldsymbol{e}} = \boldsymbol{A}\vec{\boldsymbol{y}} - \vec{\boldsymbol{b}}, \tag{3.6}$$

which is used to calculate the gradient,

$$\nabla \mathcal{L} = \boldsymbol{A}^T \vec{\boldsymbol{e}}. \tag{3.7}$$

We can efficiently calculate $\boldsymbol{A}\vec{\boldsymbol{y}}$ (needed for calculating the error) using Eq. 3.5. Based on the matrix representation in Eq. 3.4, we can derive an expression for $\boldsymbol{A}^T$ (needed for calculating the gradient):

$$\nabla \mathcal{L} = \boldsymbol{F}^{-1} \operatorname{diag}^*(\boldsymbol{F}\boldsymbol{P}\vec{\boldsymbol{h}}_{\boldsymbol{0}})\boldsymbol{F}\boldsymbol{P}\vec{\boldsymbol{e}} \tag{3.8}$$

where the * superscript denotes the complex conjugate. This can be implemented as

```
grad = ifftshift(ifft2(conj(fft2(pad(h))).*fft2(pad(e)))) .
```
(3.9)

For regularization, we apply either non-negativity alone, or a combination of non-negativity and total variation.

Extension to Color

In prior sections, we've represented the image as a set of grayscale intensities. To extend to color, we use the built-in demosaicing algorithm on the camera sensor to output an RGB measurement. We apply the image reconstruction algorithm above to each of the three color channels separately and, at the end, recombine the reconstructions into a single RGB image. Although there may be some color dispersion in the PSF, we find that it's effective to use the same PSF measurement, taken with a white LED, for each color channel.

A more complete approach would be to simultaneously solve for all three color channels, allowing implementation of cross-channel priors which could take advantage of similarities between the red, green, and blue images [59]. However, this approach means that it is no longer possible to parallelize over color, potentially slowing down processing speeds.

Compressed Sensing with Pixel Erasures

Unlike a traditional lens, the system matrix $\boldsymbol{A}$ of DiffuserCam has a large degree of randomness from the pseudorandom PSF, making it well-suited for compressed sensing. Although $\boldsymbol{A}$ is not a true random Gaussian matrix (ideal for compressed sensing), the inner products between PSFs from different locations is low, which is the definition of matrix incoherence (see Sec. 2.2), making it likely that compressed sensing techniques will succeed. To test this idea, we explore the scenario in which some random fraction of the sensor pixels do not record a measurement, and we attempt to recover the full image including the missing pixels.

The intuition is that each point on the object maps to many pixels on the sensor, so even if some sensor pixels are removed, the measurement still contains information about

the whole object. We can update the forward model in Eq. 3.4 by adding a point-wise multiplication with a binary mask, $\vec{\boldsymbol{m}} \in \mathbb{R}^p$. The mask is the same size as the sensor and contains ones at measured pixels and zeros at pixels the are erased.

$$\begin{aligned}\vec{\boldsymbol{b}} &= \boldsymbol{A}_{\text{erasures}}\vec{\boldsymbol{y}} \\ &= \text{diag}(\vec{\boldsymbol{m}})\boldsymbol{C}\boldsymbol{F}^{-1}\,\text{diag}(\boldsymbol{F}\boldsymbol{P}\vec{\boldsymbol{h}}_{\boldsymbol{0}})\boldsymbol{F}\vec{\boldsymbol{y}}. \end{aligned} \tag{3.10}$$

We implement image recovery with FISTA, as described in Chapter 2, using both non-negativity and total variation as priors. In Sec. 3.3, we show that we can recover the full image even in the presence of pixel erasures. In contrast, a traditional camera with direct measurements does not handle missing pixels as well; here, missing pixels simply correspond to missing information. Although the holes could be filled in with in-painting or interpolation techniques, interpolation results in reduced resolution and in-painting [14] generally requires that only small fraction of the image pixels are missing.

3.2 System Analysis

In this section, we provide a theoretical framework for determining the resolution, field-of-view, and depth-of-field of DiffuserCam.

Resolution

If the PSFs of neighboring point sources are very similar, it is challenging to distinguish between the sources. This causes blurring in the reconstruction and limits the optical resolution of the camera. Consider two point sources located $\Delta\vec{x}$ apart with PSFs $h(\vec{u},\vec{x})$ and $h(\vec{u},\vec{x}+\Delta\vec{x})$. We define *similarity*, denoted $\mu(\Delta\vec{x})$, as the inner product of the PSFs:

$$\mu(\Delta\vec{x}) = \big\langle h(\vec{u},\vec{x}), h(\vec{u},\vec{x}+\Delta\vec{x})\big\rangle. \tag{3.11}$$

Notice that the similarity $\mu(\Delta\vec{x})$ goes to zero when the PSFs occupy a completely disjoint set of pixels.

Because our system is shift invariant, $h(\vec{u},\vec{x})$ and $h(\vec{u},\vec{x}+\Delta\vec{x})$ are translated copies of the same PSF. Therefore, we can rewrite Eq. 3.11 as $\mu(\Delta\vec{x}) = \big\langle h_0(\vec{u}), h_0(\vec{u}+m\Delta\vec{x})\big\rangle$ which is the autocorrelation of $h_0(\vec{x})$. For normalized PSFs, $\mu(0) = 1$ by definition, and ideally, $\mu(\Delta\vec{x})$ should decrease quickly as $|\Delta\vec{x}|$ increases. The normalized similarity at which two points can be distinguished from each other depends on the camera bit depth and sensor noise. We find empirically

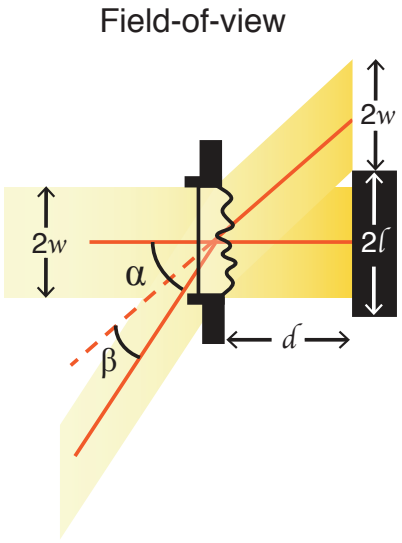

Figure 3.3: System geometry determines the field-of-view.

that the autocorrelation half-width at 70% of maximum is a good predictor of resolution for many photographic sensors.

Field-of-view

Theoretically, the angular field-of-view (FoV) of our camera, denoted α, is determined by the maximum illumination angle that contributes to the sensor measurement. Since the diffuser bends light, we take into account the diffuser's maximum deflection angle, denoted β. Based on the geometry shown in Fig. 3.3, we calculate that the angular FoV α satisfies $l + w = d \tan(\alpha - \beta)$ where $2l$ is the sensor width, $2w$ is the width of the PSF support, and d is the distance between the diffuser and sensor. Finally, real-world sensor pixels cannot detect light from arbitrarily high angles, so we include their maximum angle of acceptance, α_c, in our final FoV equation:

$$\alpha = \beta + \min\left[\alpha_c, \tan^{-1}\left(\tfrac{l+w}{d}\right)\right]. \tag{3.12}$$

Depth-of-field

Consider two on-axis point sources at different depths, z_1 and z_2. We define the depth-of-field (DoF) to be the minimum detectable separation, $\Delta z = z_1 - z_2$. Treating the diffuser paraxially, the corresponding on-axis PSFs, $h_0(\vec{x}; z_1)$ and $h_0(\vec{x}; z_2)$, are related by a coordinate scaling with parameter s: $h_0(s\vec{x}; z_1) = h_0(\vec{x}; z_2)$. Plugging this into the similarity definition in Eq. 3.11, we can determine the depth sensitivity of the camera in terms of a single PSF measurement and s with the expression $\mu(s) = \left\langle h_0(s\vec{x}), h_0(\vec{x}) \right\rangle$. Similar to our resolution analysis, we determine the values of s for which μ is sufficiently low. Then, we relate s geometrically to the corresponding DoF.

3.3 Experimental Results

To demonstrate the ease with which our method can be adapted to any existing sensor and how sensor parameters affect imaging characteristics, we built two prototype cameras. One uses a PCO Edge 5.5 Color camera, and the other a Point Grey Flea3 with Sony IMX036 monochrome CMOS chip. We placed a 0.5° Luminit Light Shaping Diffuser [98] at $d = 8.8$ mm $d = 6.4$ mm, respectively. We measured the experimental PSF of each prototype using an LED.

Using the measured PSFs in conjunction with the analysis presented in Section 3.2, we compute the theoretical system parameters for each prototype. A line from each autocorrelation is shown in Fig. 3.4a. For the PCO camera, the resolution, defined by the autocorrelation peak half-width at 70% of maximum, is 0.16°. The DoF is shown in Fig. 3.4b (blue), with a hyperfocal distance of 2929 mm. For the Point Grey prototype, the resolution is also 0.16°. The DoF is plotted in Fig. 3.4b (red), with a hyperfocal distance of 571 mm. To validate resolution, we took images of a single point source with each prototype, and reconstructions

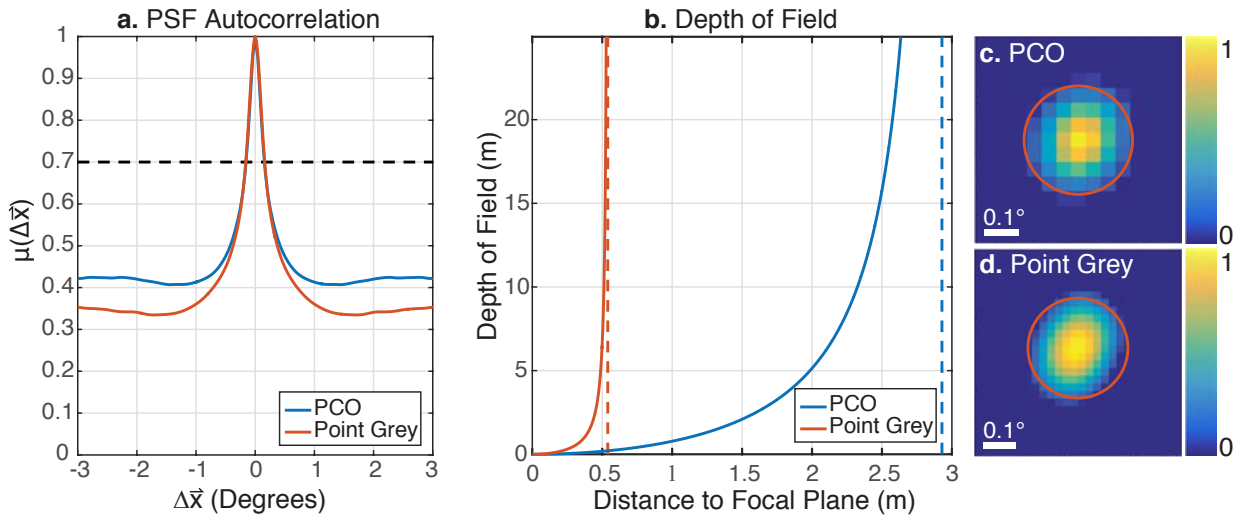

Figure 3.4: Analysis of DiffuserCam resolution and depth-of-field. (a) Autocorrelation of diffuser PSF for the two prototypes, which sets the optical resolution limit. (b) Depth of field (solid) and hyperfocal distance (dotted) for the two prototypes. (c) Zoom-in on the reconstruction of a single point source captured with each camera to illustrate resolution. The red circles represent estimated spot size based on autocorrelation width at 70% of maximum.

of each are shown in Figure 3.4c and 3.4d. The spot-size radius matches our theoretical resolution for each camera. Note that the smaller sensor size of the Point Grey camera results in significantly larger depth of field. However, since both prototypes use the same diffuser, the angular resolution is the same, despite differences in pixel size.

We validate the theoretical field-of-view on the PCO prototype. First we measured the sensor's angular acceptance by translating the calibration LED; we define the angular cutoff, α_c, to be when the brightness falls to 20% of the intensity at normal incidence, shown in Fig. 3.5. For front-side illuminated sensors, such as the PCO camera, the angular acceptance is different along the vertical and horizontal axes resulting in an asymmetry in the FoV. In addition to the sensor response, Eq. 3.12 also requires the diffuser's maximum deflection angle, β, so we measured the diffuser phase using differential phase contrast [29] shown in Appendix A.2; Figure 3.5b shows a histogram of the diffuser deflection angle, calculated based on the diffuser phase and index of refraction. We determine that $\beta = 0.5°$ for the diffuser used in our prototype. We calculate the horizontal FoV at $\pm 42°$ and the vertical FoV at $\pm 30.5°$, and we see that these values are a good match of the experimental FoV of a generic scene (Fig. 3.5c). Note that the diffuser phase and value of β are not necessary for general reconstructions; we only measure them here to demonstrate the validity of Eq. 3.12.

Next we demonstrate our camera by capturing photographs of several real world objects. Figure 3.6a shows four sample reconstructions from both prototypes, with the corresponding raw data shown in the inset. Since each acquisition is captured in a single exposure, just

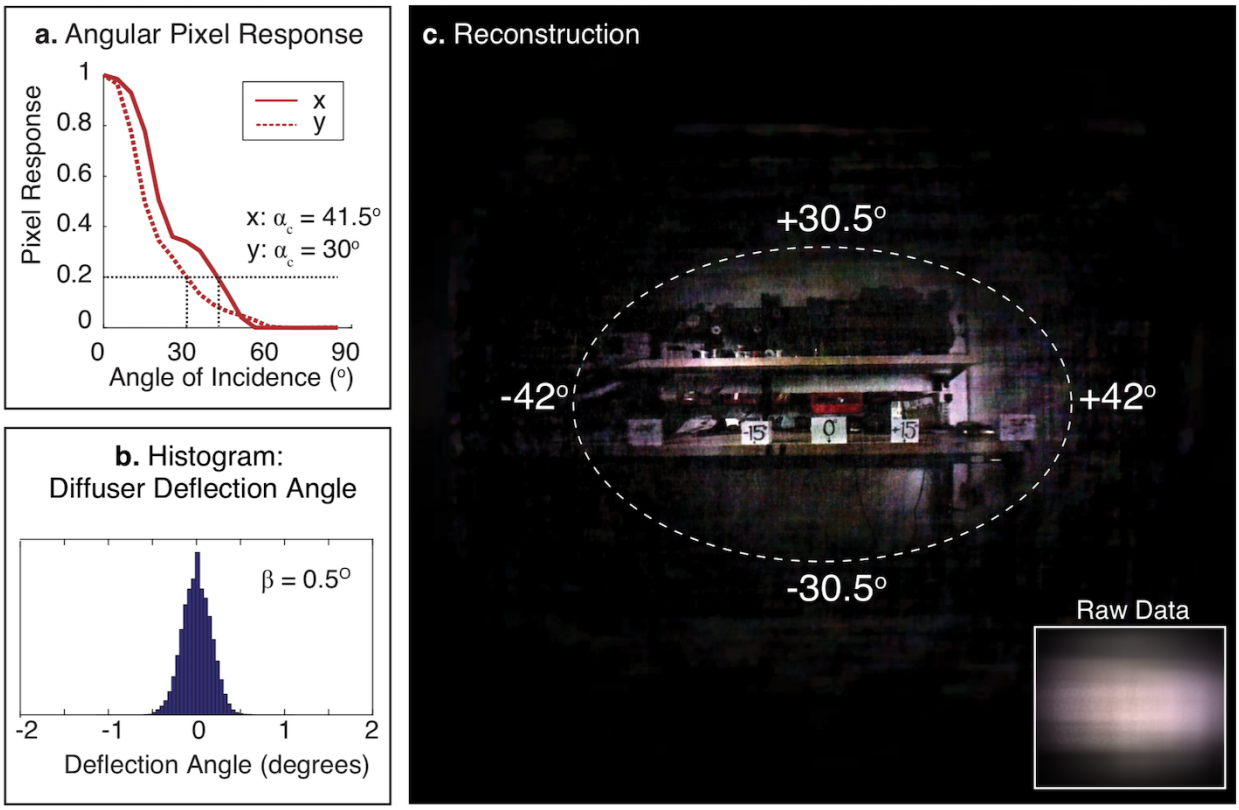

Figure 3.5: Experimental field-of-view (FoV) validation. To verify the theoretical FoV, we experimentally measure the the angular pixel response of the sensor (a) and determine the angular cutoff, α_c, where the intensity falls to below 20%. We also measure the diffuser profile and use it to get the histogram of deflection angles (b). Together with the system geometry, these parameters should determine the FoV. We experimentally measure the FoV by photographing a large, well-lit scene and determining the range of angles visible in the reconstruction. We find that the experimental FoV matches the theory, and in this case, the angular pixel response is the limiting factor on the FoV.

like a tradition camera, we can capture video as well, shown in Fig. 3.6b. The raw data is captured in real time, and then each frame is reconstructed separately before recombining. As with color, reconstructions could potentially be improved by simultaneously reconstructing all frames and including temporal priors, but this dramatically increases the memory requirements and reduces parallizability of the computations.

Finally, we experimentally use DiffuserCam for compressed sensing by recovering missing pixels that are synthetically removed, as described in Sec. 3.1. We start with experimental DiffuserCam raw data, and reconstruct the image using all the data as a baseline (Fig. 3.7, left). Then, we apply a binary pixel mask to the raw data to synthetically erase 80% of the pixels. We perform the reconstruction using only the remaining 20% of pixels with the

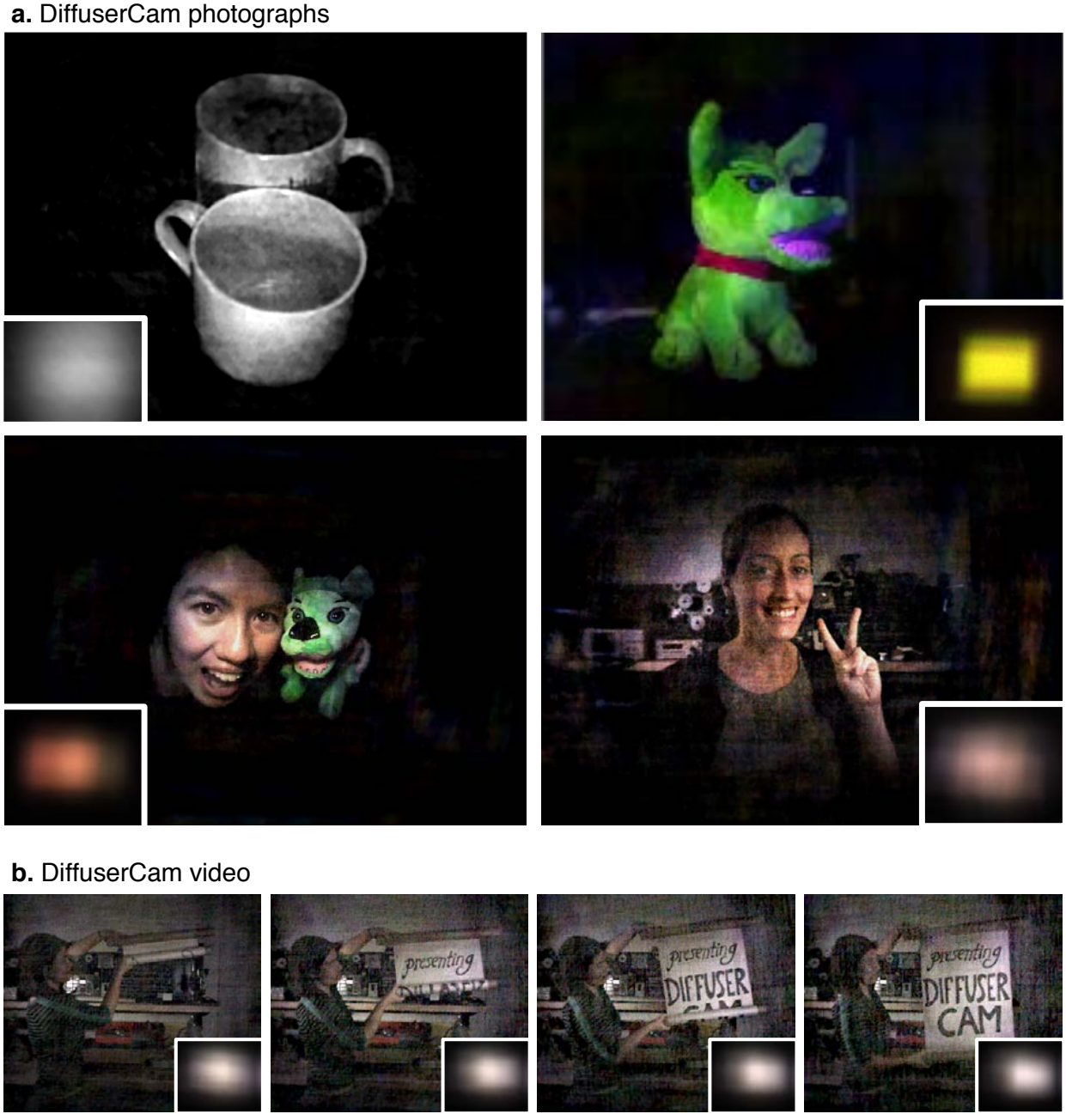

Figure 3.6: 2D photographs (a) and video (b) captured with DiffuserCam. The top left image was taken with the Point Grey prototype, and the remaining images were taken with the PCO prototype. The raw data is shown in the insets.

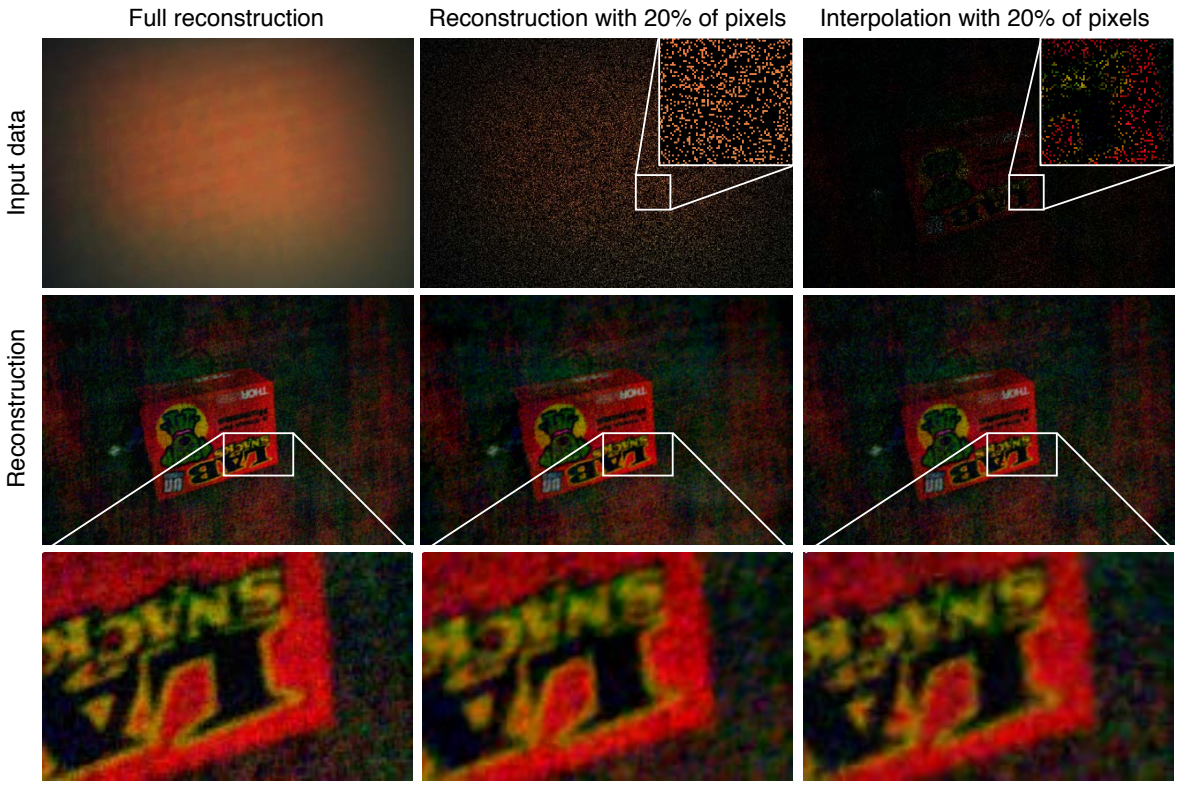

Figure 3.7: The pseudorandom structure of the diffuser enables compressed sensing with DiffuserCam. To demonstrate we erase 80% of the pixels in the raw DiffuserCam data, then reconstruct the image with only 20% of the data (middle column). We compare with erasing pixels of the reconstructed image (right column) and then linearly interpolating between the samples. This would be similar to applying the erasure pattern to the sensor of a traditional camera with a lens. Close ups of the reconstructions, bottom row, show that erasing pixels in the DiffuserCam data yields a more faithful reconstruction than removing pixels in the reconstructed data.

model described in Eq. 3.10; the resulting reconstruction, which uses non-negativity and total variation regularization, is shown in the center panel of Fig. 3.7 and demonstrates that compressed sensing can be applied to DiffuserCam. Finally, we compare with erasing pixels in the image domain, which is what would happen without the random sensing matrix from the diffuser. Starting with the full reconstruction, we erase the same pattern of pixels to get a sparse sampling of the image. We then linearly interpolate between the pixels to get the reconstruction shown on the right of Fig. 3.7. Close ups of the reconstructions show that erasing DiffuserCam pixels yields a reconstruction closer to the baseline than erasing pixels in the image directly. This highlights the utility of DiffuserCam for compressed sensing on a simple example; in Chapter 4, we use compressed sensing with the diffuser for the powerful application of recovering 3D information from a single image.

Chapter 4

Single Exposure 3D Imaging with DiffuserCam

In this chapter[1], we demonstrate how we can use a diffuser to project 3D information onto a 2D sensor and then computationally recover the 3D volume. Unlike scanning and multi-shot methods, which can achieve high spatial resolution 3D imaging but sacrifice capture speed [41, 61], our method is single-shot. Prior single-shot 3D methods [20, 119] are fast but may have low resolution or small field-of-view (FoV) and often require bulky hardware and complicated setups. Here, we introduce a compact and inexpensive single-shot lensless optical system that is capable of 3D imaging. We show how it can reconstruct a large number of voxels by leveraging compressed sensing.

The system architecture is the same as in Chapter 3: the only optical component is the diffuser, a thin phase mask, which is placed a few millimeters in front of an image sensor. Each point source in 3D space creates a unique pseudorandom caustic pattern that covers a large portion of the sensor. Because of this, compressed sensing algorithms can be used to reconstruct more voxels than pixels captured, provided that the 3D sample is sparse in some domain. We solve the inverse problem via a sparsity-constrained optimization procedure. Extending the convolution model from Chapter 3 to 3D enables efficient computation and calibration, allowing us to reconstruct several orders of magnitude more voxels than related previous work [43, 94].

This system, like many computational cameras, uses a nonlinear reconstruction algorithm, resulting in object-dependent performance. To quantify, we experimentally measure the resolution of our prototype with different objects. We show that the standard two-point resolution criterion is misleading and should be considered a best-case scenario. To better explain the variable resolving power of our system, we propose a new local condition number analysis that is consistent with our experiments.

[1]This chapter is based on the published journal paper titled "DiffuserCam: Lensless Single Exposure 3D Imaging" and is joint work with Nick Antipa, Reinhard Heckel, Ben Mildenhall, Emrah Bostan, Ren Ng, and Laura Waller [8].

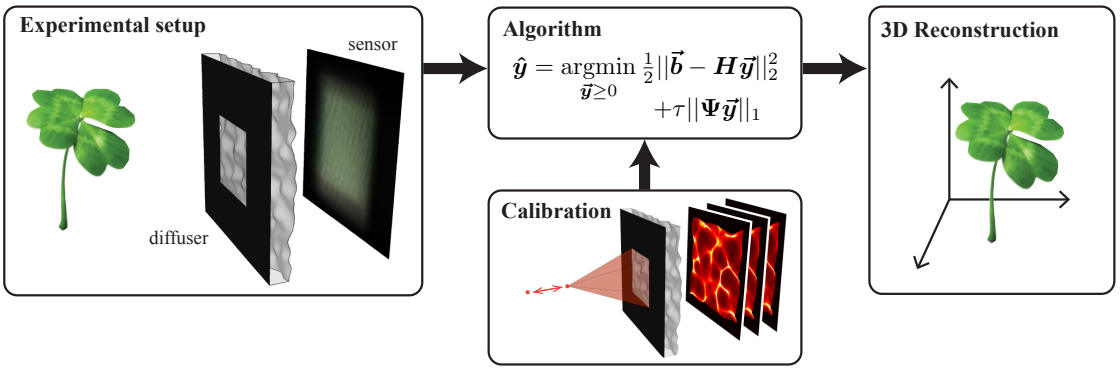

Figure 4.1: DiffuserCam setup and reconstruction pipeline. Our lensless system consists of a diffuser placed in front of a sensor (bumps on the diffuser are exaggerated for illustration). The system encodes a 3D scene into a 2D image on the sensor. A one-time calibration consists of scanning a point source axially while capturing images. Images are reconstructed computationally by solving a nonlinear inverse problem with a sparsity prior. The result is a 3D image reconstructed from a single 2D measurement.

4.1 Related Work

Lensless cameras for 2D photography have shown great promise because of their small form factors. Unlike traditional cameras, in which a point in the scene maps to a pixel on the sensor, lensless cameras map a point in the scene to many points on the sensor, requiring computational reconstruction. A typical lensless architecture replaces the lens with an encoding element placed directly in front of the sensor. 2D lensless cameras have demonstrated passive incoherent imaging using amplitude masks [10], diffractive masks [140, 53], random reflective surfaces [48, 142], and modified microlens arrays [144]. Our system uses a similar architecture with a diffuser as the encoding element, and also extends the design and image reconstruction to enable 3D capture.

Light field cameras, also called integral imagers, passively capture 4D space-angle information in a single-shot [110], which can be used for 3D reconstructions. This concept can be built into a thin form factor with microlens arrays [63] or Fresnel zone plates [68]. Lenslet array-based 3D capture schemes have also been used in microscopy [88], where wave-optical [20, 93] or scattering [119, 93] effects can be included. All of these systems, however, must trade resolution (or field-of-view) for single-shot capture, limiting the number of useful voxels. DiffuserCam improves upon this tradeoff, capturing large 3D volumes with high voxel counts in a single exposure.

Lensless imaging has also been demonstrated with coherent systems in both 2D [57, 30, 134, 135] and 3D [19, 86, 17, 46, 132], but these methods require active (coherent) illumination, limiting applications. Further, many coherent methods do not generate unambiguous 3D reconstructions, but rather use digital refocusing to estimate depth. DiffuserCam, on the other hand, exhibits actual depth sectioning (in the absence of occlusions) for "true 3D."

Since methods for imaging through scattering often use diffusers as a proxy for general scattering media [75, 44, 133], our mathematical models will be similar. However, instead of trying to mitigate the effects of unwanted scattering, here we use the diffuser as an optical element in our system design. We choose a thin, optically smooth diffuser that refracts pseudorandomly, producing high contrast patterns under incoherent illumination. Such diffusers have been used in light field imaging [6] and coherent holography [86, 73]. Coherent multiple scattering has been demonstrated as an encoding mechanism for 2D compressed sensing [94], but necessitates a transmission matrix approach that does not scale well past a few thousand pixels. We achieve similar benefits without needing coherent illumination, and we reconstruct 3D objects, rather than 2D. Finally, an important benefit of our system over previous work is the simple calibration and efficient computation that allow for 3D reconstruction at megavoxel scales with superior image quality.

4.2 System Overview

As decribed in Chapter 3, DiffuserCam is part of the class of mask-based passive lensless imagers in which a phase or amplitude mask is placed a small distance in front of a sensor, with no main lens. Our mask (the diffuser) is a thin transparent phase object with smoothly varying thickness (see Fig. 4.1). When a temporally incoherent point source is placed in the scene, we observe a high-frequency pseudorandom caustic pattern at the sensor. The caustic patterns, termed Point Spread Functions (PSFs), vary with the 3D position of the source, thereby encoding 3D information.

To illustrate how the caustics capture 3D information, Fig. 4.2 shows simulations of the PSFs for a point source at different locations in object space. A lateral shift of the point source causes a lateral translation of the PSF [47, 47]. An axial shift of the point source causes (approximately) a scaling of the PSF. Hence, each 3D position in the volume generates a unique caustic pattern. The structure and spatial frequencies present in the PSFs determine our reconstruction resolution. By using a phase mask (which concentrates light better than an amplitude mask) and designing the system to retain high spatial frequencies over a large range of depths, DiffuserCam attains good lateral resolution across the volumetric field-of-view.

By assuming that all points in the scene are incoherent with each other, the measurement can be modeled as a linear combination of PSFs from different 3D positions. We represent this as matrix-vector multiplication:

$$\vec{\boldsymbol{b}} = \boldsymbol{A}\vec{\boldsymbol{y}}, \tag{4.1}$$

where $\vec{\boldsymbol{b}}$ is a vector containing the 2D sensor measurement and $\vec{\boldsymbol{y}}$ is a vector representing the intensity of the object at every point in the 3D FoV, sampled on a user-chosen grid. $\boldsymbol{A}$ is the forward model matrix whose columns consist of each of the caustic patterns created by the corresponding 3D points on the object grid. The number of entries in $\vec{\boldsymbol{b}}$ and the number of rows of $\boldsymbol{A}$ are equal to the number of pixels on the image sensor, but the number of columns in $\boldsymbol{A}$ is set by the choice of reconstruction grid (discussed in Sec. 4.4). Note that this model does not account for partial occlusion of sources, as discussed in Sec. 2.5 of Chapter 2.

In order to reconstruct the 3D object, $\vec{\boldsymbol{y}}$, from the measured 2D image, $\vec{\boldsymbol{b}}$, we must solve Eq. 4.1 for $\vec{\boldsymbol{y}}$. However, if we solve on a 3D reconstruction grid that corresponds to the full optical resolution of our system (measured in Sec. 4.4), $\vec{\boldsymbol{y}}$ will contain more voxels than there are sensor pixels. In this case, $\boldsymbol{A}$ has more columns than rows, so the problem is underdetermined and we cannot uniquely recover $\vec{\boldsymbol{y}}$ simply by inverting Eq. 4.1. To remedy this, we rely on sparsity-based principles [25]. We exploit the fact that many 3D objects are sparse in some domain, meaning that the majority of coefficients are zero after a linear transformation. We enforce this sparsity as a prior and solve the ℓ_1 regularized nonnegativity-constrained inverse problem:

$$\hat{\vec{\boldsymbol{y}}} = \underset{\vec{\boldsymbol{y}} \geq 0}{\operatorname{argmin}} \tfrac{1}{2} \|\vec{\boldsymbol{b}} - \boldsymbol{A}\vec{\boldsymbol{y}}\|_2^2 + \tau \|\Psi \vec{\boldsymbol{y}}\|_1. \tag{4.2}$$

Here, Ψ maps $\vec{\boldsymbol{y}}$ into a domain in which it is sparse ($\Psi\vec{\boldsymbol{y}}$ is mostly zeros), and τ is a tuning parameter that adjusts the degree of sparsity. For objects that are sparse in voxels, such as fluorescent particles in a volume, Ψ is the identity matrix. In our results we show reconstruction of objects that are not sparse in voxels but are sparse in the gradient domain. Hence, we choose Ψ to be the finite difference operator and $\|\Psi\vec{\boldsymbol{y}}\|_1$ to be the 3D Total Variation (TV) semi-norm [127]. In general, any linear sparsity transformation may be used (e.g. wavelets), but we utilize only identity and gradient representations in this work.

Equation 4.2 is the basis pursuit problem in compressed sensing [25]. For this optimization procedure to succeed, $\boldsymbol{A}$ must have distributed, uncorrelated columns. Since our diffuser creates high spatial frequency caustics that spread across many pixels in a pseudorandom fashion, any shift or magnification of the caustics leads to a new pattern that is uncorrelated with the original one. As discussed in Sec. 4.3, these properties allow us to reconstruct 3D images via compressed sensing, and we quantify this effect in Appendix A.2.

4.3 Methods

System Architecture

The hardware setup for our prototype DiffuserCam (Fig. 4.3a) is exactly the same as in Chapter 3: it consists of an off-the-shelf diffuser (Luminit 0.5°) placed at a fixed distance in front a sensor (PCO.edge 5.5 Color camera, 6.5 µm pixels). The diffuser has a flat input surface and an output surface that is described statistically as Gaussian lowpass-filtered white noise with an average spatial feature size of 140 µm and average slope magnitude of 0.7° (see Appendix A.2). The convex bumps on the diffuser surface can be thought of as randomly-spaced microlenses that have statistically-varying focal lengths and f-numbers. The average focal length determines the distance at which the caustics have highest contrast (the *caustic plane*), which is where we place the sensor [6]. This distance, measured experimentally, is 8 mm for our diffuser. However, the high average f-number of the bumps (8 mm/140 µm=57) means that the caustics maintain high contrast over a large range of propagation distances. Therefore, the diffuser need not be placed precisely at the caustic plane (in our prototype,

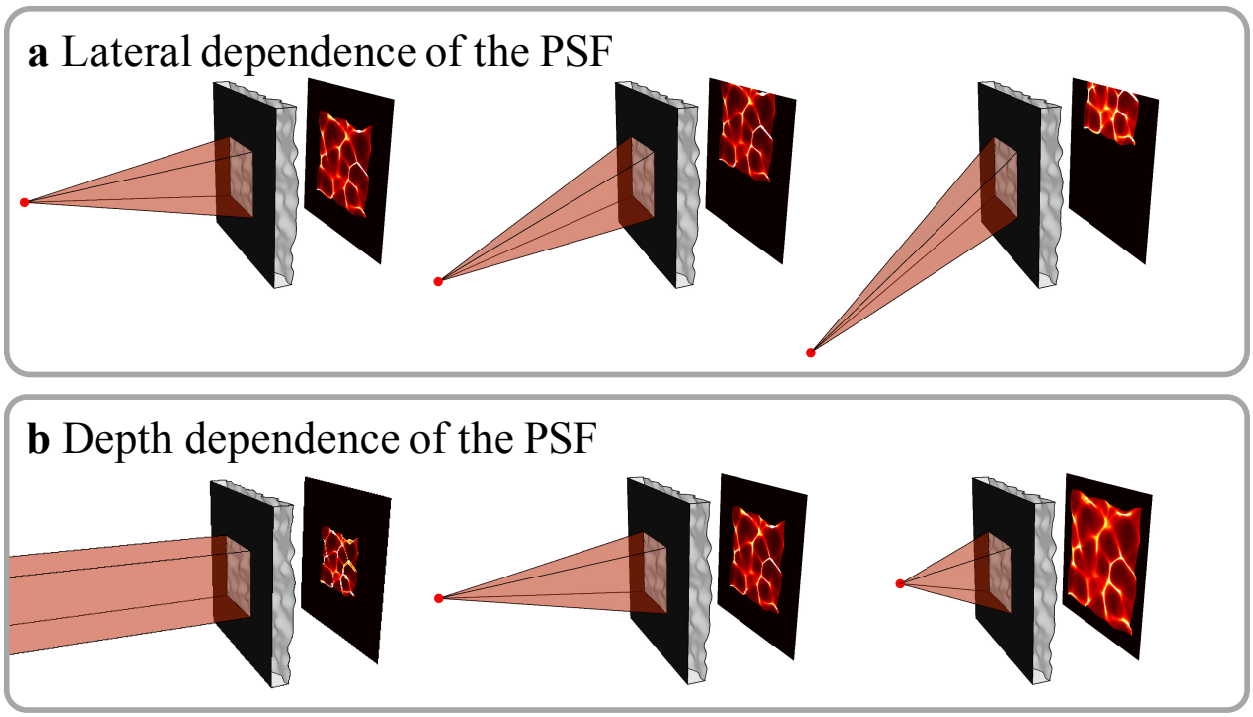

Figure 4.2: The caustic pattern shifts with lateral shifts of a point source in the scene and scales with axial shifts. (a) Ray-traced renderings of caustics as a point source moves laterally. For large shifts, part of the pattern is clipped by the sensor. (b) The caustics magnify as the source is brought closer.

d=8.9 mm). We also affix a 5.5×7.5 mm aperture on the textured side of the diffuser to limit the support of the caustics.

Similar to a traditional camera, the sensor's pixel pitch should Nyquist sample the minimum features of the PSF. Since the f-number of the smallest bumps on the diffuser determine the minimum feature size of the caustics, it will also set the lateral optical resolution. In our case, the smallest features generated by the caustic patterns are roughly twice the pixel pitch of our sensor, so we perform 2×2 binning on the data, yielding 1.3 megapixel images, before applying our reconstruction algorithm.

Convolutional Forward Model

Recovering a 3D image requires knowing the system matrix, $\boldsymbol{A}$, which is extremely large. Measuring or storing the full $\boldsymbol{A}$ would be impractical, requiring millions of calibration images and operating on multi-Terabyte matrices. Instead, we extend the convolution model from Chapter 3 to 3D, drastically reducing complexity of both calibration and computation.

We describe the object, $\vec{\boldsymbol{y}}$, as a set of point sources located at $(\vec{x}, z)$ on a non-Cartesian 3D grid, where $\vec{x}$ represents the 2D lateral coordinates as in Chapters 2 and 3 and z is

the axial distance, measured from the diffuser. The relative radiant power collected by the aperture from each source is $y(\vec{x}, z)$. The caustic pattern at pixel $\vec{u}$ on the sensor due to a unit-powered point source at $(\vec{x}, z)$ is the PSF, $h(\vec{u}; \vec{x}, z)$. Thus, $b(\vec{u})$ is the sum of all 2D sensor measurements for each non-zero point in y after propagating through the diffuser and onto the sensor. This lets us explicitly write the matrix-vector multiplication $\boldsymbol{A}\vec{\boldsymbol{y}}$ by summing over all voxels in the FoV:

$$b(\vec{u}) = \sum_{(\vec{x},z)} y(\vec{x}, z)\, h(\vec{u}; \vec{x}, z). \tag{4.3}$$

Recall that $\vec{\boldsymbol{b}}$ is the discrete vector version of $b(\vec{u})$, $\vec{\boldsymbol{y}}$ is the discrete version of $y(\vec{x}, z)$, and $\boldsymbol{A}$ is the system matrix containing each $h(\vec{u}; \vec{x}, z)$ as a column.

Our convolution model amounts to a shift invariance (or infinite memory effect [75, 44]) assumption, which greatly simplifies the evaluation of Eq. 4.3. Consider the caustics created by point sources at a fixed distance, z, from the diffuser. Because the diffuser surface is slowly varying and smooth, the paraxial approximation holds. As described in detail in Chapter 3, this implies that a lateral translation of the source by $\Delta\vec{x}$ leads to a lateral shift of the caustics on the sensor by $\Delta\vec{u} = m\Delta\vec{x}$, where m is the paraxial magnification. We validate this behavior in both simulations (see Fig. 4.2) and experiments (see Fig. 5.1), and in Chapter 5 we further discuss shift-varying system. For notational convenience, we define the on-axis caustic pattern at depth z as $h_0(\vec{u}; z) := h(\vec{u}; \vec{x} = 0, z)$. Thus, the off-axis caustic pattern is given by $h(\vec{u}; \vec{x}, z) = h_0(\vec{u} + m\vec{x}; z)$. Plugging into Eq. 4.3, the sensor measurement is then given by:

$$\begin{aligned} b(\vec{u}) &= \sum_{z} \sum_{\vec{x}} y(\vec{x}, z) h_0(\vec{u} + m\vec{x}; z) \\ &= \boldsymbol{C} \sum_{z} \left[y\left(\frac{-\vec{u}}{m}, z\right) * h_0\left(\vec{u}; z\right) \right]. \end{aligned} \tag{4.4}$$

Here, $*$ represents 2D discrete convolution over $\vec{u}$, which returns arrays that are larger than the originals. Hence, we crop to the original sensor size, denoted by the linear operator $\boldsymbol{C}$ as described in Chapter 3. For an object discretized into N_z depth slices, the number of columns of $\boldsymbol{A}$ is N_z times larger than the number of elements in $\vec{\boldsymbol{b}}$ (i.e. the number of sensor pixels), so our system is underdetermined.

The cropped convolution model provides three benefits. First, it allows us to compute $\boldsymbol{A}\vec{\boldsymbol{y}}$ as a linear operator in terms of N_z images, rather than instantiating $\boldsymbol{A}$ explicitly (which would require petabytes of memory to store). In practice, we evaluate the sum of 2D cropped convolutions using a single circular 3D convolution, implemented with 3D FFTs, which scale well to large arrays. Second, it provides a theoretical justification of our system's capability for compressed sensing; derivations in [81] show that translated copies of a random pattern provide close-to-optimal performance.

The third benefit of our convolution model is that it enables simple calibration. Rather than measuring the system response for every voxel (hundreds of millions of images), we only need to capture a single calibration image of the caustic pattern from an on-axis point

source. Though the scaling effect shown in Fig. 4.2 suggests that we could use only one image for calibrating the entire 3D space (by scaling it to predict PSFs at different depths), we obtain better results when we calibrate the PSF at each depth. A typical calibration thus consists of capturing images as a point source is moved axially. This takes minutes, but need only be performed once. The added aperture at the diffuser ensures that a point source at the minimum z distance generates caustics that just fill the sensor, so that the entire PSF is captured in each image (see Appendix A.2).

Inverse Algorithm

Our inverse problem is extremely large in scale, with millions of inputs and outputs. Even with the convolution model described above, projected gradient techniques, like FISTA, are extremely slow. In this work, we instead use the Alternating Direction Method of Multipliers (ADMM) [18] and derive a variable splitting that leverages the specific structure of our problem.

Our algorithm uses the fact that Ψ can be written as a circular convolution for both the 3D TV and native sparsity cases. Additionally, we factor the forward model in Eq. 4.4 into a diagonal component, $\boldsymbol{D}$, and a 3D convolution matrix, $\mathbf{M}$, such that $\boldsymbol{A} = \boldsymbol{D}\mathbf{M}$ (details in Appendix A.3). Thus, both the forward operator and the regularizer can be computed in 3D Fouier space. This enables us to use variable-splitting [5, 104, 3] to formulate the constrained counterpart of Eq. 4.2:

$$\begin{aligned} \hat{\vec{\boldsymbol{y}}} = \underset{w\geq 0,u,v}{\operatorname{argmin}} \tfrac{1}{2}\|\vec{\boldsymbol{b}} - \boldsymbol{D}v\|_2^2 + \tau\|u\|_1 \\ \text{s.t. } v = \mathbf{M}\vec{\boldsymbol{y}}, u = \Psi\vec{\boldsymbol{y}}, w = \vec{\boldsymbol{y}}, \end{aligned} \tag{4.5}$$

where v, u, and w are auxiliary variables. We solve Eq. 4.5 by following the augmented Lagrangian arguments [112]. Using ADMM, this results in the following scheme at iteration k:

$$\begin{aligned} u^{k+1} &\leftarrow \mathcal{T}_{\frac{\tau}{\mu_2}}\left(\Psi\vec{\boldsymbol{y}}^k + \eta^k/\mu_2\right) \\ v^{k+1} &\leftarrow (\boldsymbol{D}^\intercal\boldsymbol{D} + \mu_1 I)^{-1}\left(\xi^k + \mu_1\mathbf{M}\vec{\boldsymbol{y}}^k + \boldsymbol{D}^\intercal\vec{\boldsymbol{b}}\right) \\ w^{k+1} &\leftarrow \max\left(\rho^k/\mu_3 + \vec{\boldsymbol{y}}^k, 0\right) \\ \vec{\boldsymbol{y}}^{k+1} &\leftarrow \left(\mu_1\mathbf{M}^\intercal\mathbf{M} + \mu_2\Psi^\intercal\Psi + \mu_3 I\right)^{-1} r^k \\ \xi^{k+1} &\leftarrow \xi^k + \mu_1(\mathbf{M}\vec{\boldsymbol{y}}^{k+1} - v^{k+1}) \\ \eta^{k+1} &\leftarrow \eta^k + \mu_2(\Psi\vec{\boldsymbol{y}}^{k+1} - u^{k+1}) \\ \rho^{k+1} &\leftarrow \rho^k + \mu_3(\vec{\boldsymbol{y}}^{k+1} - w^{k+1}), \end{aligned}$$

where

$$r^k = (\mu_3 w^{k+1} - \rho^k) + \Psi^\intercal\left(\mu_2 u^{k+1} - \eta^k\right) + \mathbf{M}^\intercal\left(\mu_1 v^{k+1} - \xi^k\right).$$

Note that $\mathcal{T}_\nu$ is a vectorial soft-thresholding operator with a threshold value of ν [156]. ξ, η and ρ are the Lagrange multipliers associated with v, u, and w, respectively. The scalars

μ_1, μ_2 and μ_3 are penalty parameters which we compute automatically using the tuning strategy in [18]. A MATLAB implementation of our algorithm is available at [7].

Although our algorithm involves two large-scale matrix inversions, both can be computed efficiently and in closed form. Since $\boldsymbol{D}$ is diagonal, $(\boldsymbol{D}^\intercal \boldsymbol{D} + \mu_1 I)$ is itself diagonal, requiring complexity $\mathcal{O}(n)$ to invert using point-wise multiplication. Additionally, all three matrices in $(\mu_1 \mathbf{M}^\intercal \mathbf{M} + \mu_2 \Psi^\intercal \Psi + \mu_3 I)$ are diagonalized by the 3D discrete Fourier transform (DFT) matrix, so inversion of the entire term can be done using point-wise division in 3D frequency space. Therefore, its inversion has good computational complexity, $\mathcal{O}(n^3 \log n)$, since it is dominated by two 3D FFTs being applied to n^3 total voxels. We parallelize our algorithm on the CPU using C++ and Halide [124], a high performance programming language for image processing.

A typical reconstruction requires at least 200 iterations. Solving for 2048×2048×128=537 million voxels takes 26 minutes (8 seconds per iteration) on a 144-core workstation and requires 85 Gigabytes of RAM. A smaller reconstruction (512×512×128=33.5 million voxels) takes 3 minutes (1 second per iteration) on a 4-core laptop with 16 Gigabytes of RAM.

4.4 System Analysis

Unlike traditional cameras, the performance of computational cameras depends on properties of the scene being imaged (e.g. the number of sources). As a consequence, standard two-point resolution metrics may be misleading, as they do not predict resolving power for complex objects. To address this, we propose a new local condition number metric that better predicts performance. We analyze resolution, FoV and the validity of the convolution model, then combine these analyses to determine the appropriate sampling grid for our experiments.

Field-of-View

At every depth in the volume, the angular half-FoV is determined by the most extreme lateral position that contributes to the measurement. There are two possible limiting factors. The first is the geometric angular cutoff, α, set by the aperture size, w, the sensor size, l, and the distance from the diffuser to the sensor, d (see Fig. 4.3a). Since the diffuser bends light, we also take into account the diffuser's maximum deflection angle, β. This gives a geometric angular half-FoV at every depth of $l + w = 2d\tan(\alpha - \beta)$. The second limiting factor is the angular response of the sensor pixels. Real-world sensor pixels may not accept light at the high angles of incidence that our lensless camera accepts, so the sensor angular response (shown in Fig. 4.3b) may limit the FoV. Defining the angular cutoff of the sensor, α_c, as the angle at which the camera response falls to 20% of its on-axis value, we can write the overall FoV equation as:

$$\text{FoV} = \beta + \min[\alpha_c, \tan^{-1}(\tfrac{l+w}{2d})]. \tag{4.6}$$

Since we image in 3D, we must also consider the axial FoV. In practice, the axial FoV is limited by the range of calibrated depths. However, the system geometry creates bounds

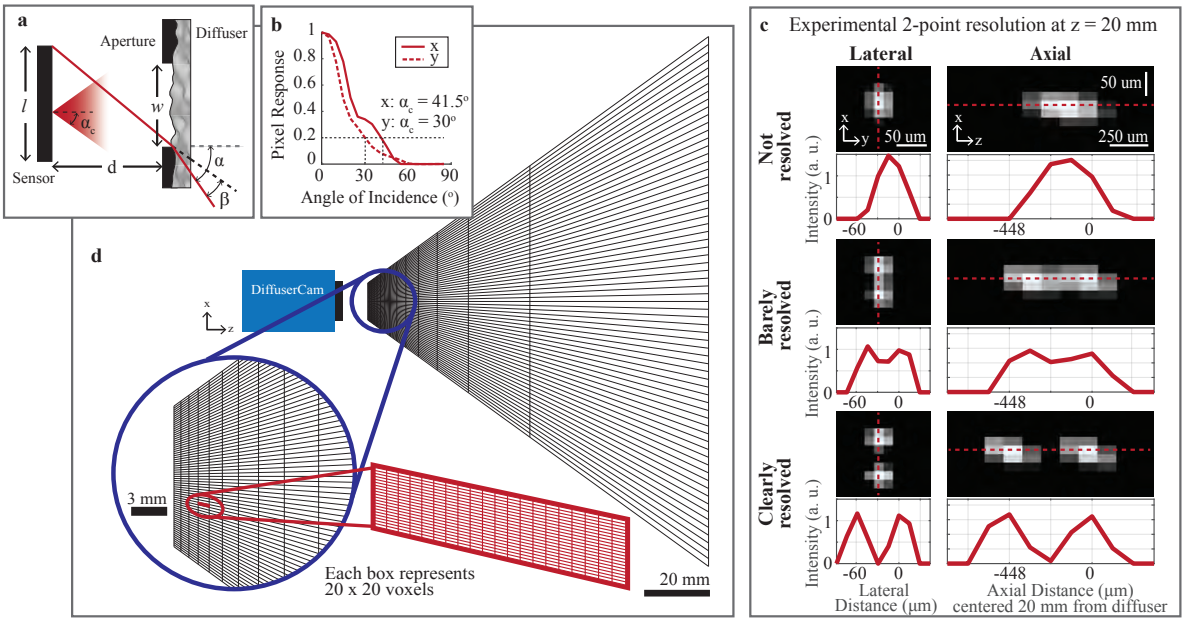

Figure 4.3: Experimentally determined field-of-view (FoV) and resolution. (a) System architecture with design parameters. (b) Angular pixel response of our sensor. We define the angular cutoff (α_c) as the angle at which the response falls to 20%. (c) Reconstructed images of two points (captured separately) at varying separations laterally and axially, near the $z = 20\,\text{mm}$ depth plane. Points are considered resolved if they are separated by a dip of at least 20%. (d) To-scale non-uniform voxel grid for 3D reconstruction. The chosen voxel grid is based on the system geometry and Nyquist-sampled two-point resolution over the entire FoV. For visualization purposes, each box represents 20×20 voxels, as shown in red.

on possible calibration locations. Point sources arbitrarily close to the sensor would produce caustic patterns that exceed the sensor size. To avoid this complication, we impose a minimum object distance at which an on-axis point source creates caustics that fill the sensor. Point sources arbitrarily far from the sensor theoretically can be captured, but axial resolution degrades with depth. The hyperfocal plane represents the axial distance beyond which no depth discrimination is available, establishing an upper bound. Objects beyond the hyperfocal focal plane can still be reconstructed to create 2D images for photographic applications [82], without any hardware modifications.

In our prototype, the axial FoV ranges from the minimum calibration distance (7.3 mm) to the hyperfocal plane (2.3 m). The angular FoV is limited by the pixel angular acceptance ($\alpha_c = 41.5°$ in x, $\alpha_c = 30°$ in y). Combined with our diffuser's maximum deflection angle ($\beta = 0.5°$) this yields an angular FoV of $\pm 42°$ in x and $\pm 30.5°$ in y. We validated the lateral FoV experimentally by capturing a scene at optical infinity and measuring the angular extent of the result (see Fig. 3.5 in Chapter 3).

Resolution

Investigating optical resolution is critical for both quantifying system performance and choosing our reconstruction grid. Although the raw data is collected on a fixed sensor grid, we can choose the non-uniform 3D reconstruction grid arbitrarily. This choice of reconstruction grid is important. When the grid is chosen with voxels that are too large, resolution is lost, and when they are too small, extra computation is performed without resolution gain. In this section we explain how to choose the grid of voxels for our reconstructions, with the aim of Nyquist sampling the two-point optical resolution limit.

Two-point resolution

A common metric for resolution analysis in traditional cameras is two-point distinguishablity. We measure our system's two-point resolution by imaging scenes containing two point sources at different separation distances, built by summing together images of a single point source (1 µm pinhole, wavelength 532 nm) at two different locations. We reconstruct the scene using our algorithm, with $\tau = 0$ to remove the influence of the regularizer. To ensure best-case resolution, we use the full 5 MP sensor data (no binning). The point sources are considered distinguishable if the reconstruction has a dip of at least 20% between the sources, as in the Rayleigh criterion. Figure 4.3c shows reconstructions with point sources separated both laterally and axially.

Our system has highly non-isotropic resolution (Fig. 4.3d), but we can use our model to predict the two-point distinguishability over the entire volume from localized experiments. Due to the shift invariance assumption, the lateral resolution is constant within a single depth plane and the paraxial magnification causes the lateral resolution to vary linearly with depth. For axial resolution, the main difference between two point sources is the size of their PSF supports. We find pairs of depths such that the difference in their support widths is constant:

$$c = \frac{1}{z_1} - \frac{1}{z_2}. \tag{4.7}$$

Here, z_1 and z_2 are neighboring depths and c is a constant determined experimentally.

Based on this model, we set the voxel spacing in our grid to Nyquist sample the 3D two-point resolution. Figure 4.3d shows a to-scale map of the resulting voxel grid. Axial resolution degrades with distance until it reaches the hyperfocal plane ($\sim$2.3 m from the camera), beyond which no depth information is recoverable. Due to the non-telecentric nature of the system, the voxel sizes are a function of depth, with the densest sampling occurring close to the camera. Objects within 5 cm of the camera can be reconstructed with somewhat isotropic resolution; this is where we place objects in practice.

Multi-point resolution

In a traditional camera, resolution is a function of the system and is independent of the scene. In contrast, computational cameras that use nonlinear reconstruction algorithms may incur degradation of the effective resolution as the scene complexity increases. To

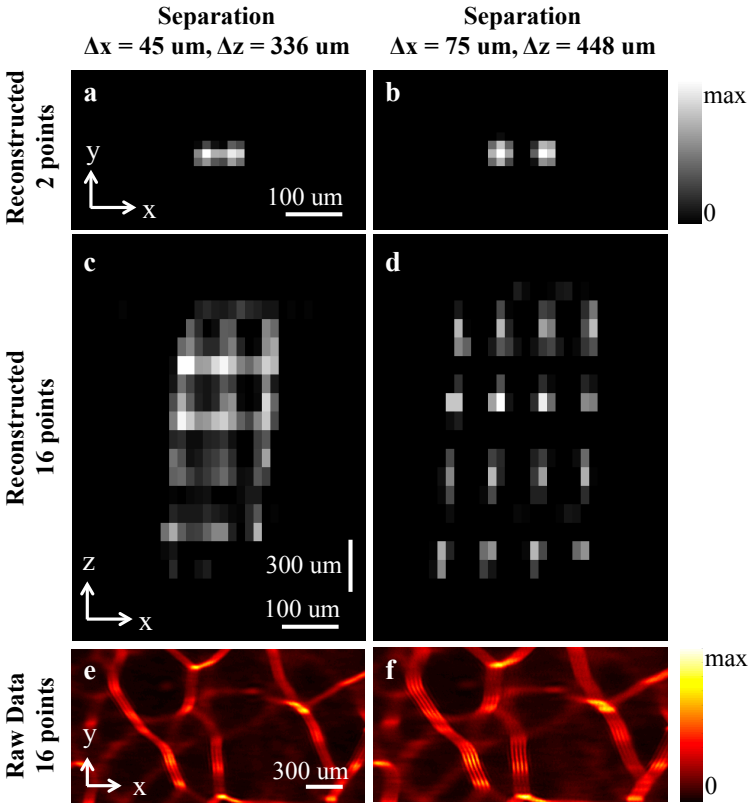

Figure 4.4: Our computational camera has object-dependent performance, such that the resolution depends on the number of points. (a) To illustrate, we show here a situation with two points successfully resolved at the two-point resolution limit $(\Delta x, \Delta z) = (45\mu m, 336\mu m)$ at a depth of approximately 20 mm. (c) However, when the object consists of more points (16 points in a 4×4 grid in the $x - z$ plane) at the same spacing, the reconstruction fails. (b,d) Increasing the separation to $(\Delta x, \Delta z) = (75\mu m, 448\mu m)$ gives successful reconstructions. (e,f) A close-up of the raw data shows noticeable splitting of the caustic lines for the 16 point case, making the points distinguishable. Heuristically, the 16 point resolution cutoff is a good indicator of resolution for real-world objects.

demonstrate this in our system, we consider a more complex scene consisting of 16 point sources. Figure 4.4 shows experiments using 16 point sources arranged in a 4×4 grid in the (x, z) plane at two different spacings. The first spacing is set to match the measured two-point resolution limit (Δx=45μm, Δz=336μm). Despite being able to separate two points at this spacing, we cannot resolve all 16 sources. However, if we increase the source separation to (Δx=75μm, Δz=448μm), all 16 points are distinguishable (Fig. 4.4d). In this example, the usable lateral resolution of the system degrades by approximately 1.7× due to the increased scene complexity. As we show in the next section, the resolution loss does not become arbitrarily worse as the scene complexity increases.

This experiment demonstrates that existing resolution metrics cannot be blindly used to determine performance of computational cameras like ours. How can we then analyze resolution if it depends on object properties? In the next section, we introduce a general theoretical framework for assessing resolution in computational cameras like ours.

Local condition number theory

Our goal is to provide new theory that describes how the effective reconstruction resolution of computational cameras changes with object complexity. To do so, we introduce a numerical analysis of how well our forward model can be inverted.

First, note that recovering the image $\vec{\boldsymbol{y}}$ from the measurement $\vec{\boldsymbol{b}} = \boldsymbol{A}\vec{\boldsymbol{y}}$ entails simultaneous estimation of the locations of all nonzeros within our image reconstruction, $\vec{\boldsymbol{y}}$, as well as the values at each nonzero location. To simplify the problem, suppose an oracle tells us the exact locations of every source within the 3D scene. This corresponds to knowing *a priori* the support of $\vec{\boldsymbol{y}}$, so we then need only determine the *values* of the nonzero elements in $\vec{\boldsymbol{y}}$. This can be done by solving a least squares problem using a sub-matrix consisting of only the columns of $\boldsymbol{A}$ that correspond to the indices of the nonzero voxels. If this problem fails, then the more difficult problem of simultaneously determining the nonzero locations *and* their values will certainly fail.

In practice, the measurement is corrupted by noise. The maximal effect this noise can have on the least-squares estimate of the nonzero values is determined by the condition number of the sub-matrix described above. We therefore say that the reconstruction problem is ill-posed if any sub-matrices of $\boldsymbol{A}$ are very ill-conditioned. In practice, ill-conditioned matrices result in increased noise sensitivity and longer reconstruction times, as more iterations are needed to converge to a solution.

In general, finding the worst-case sub-matrix is a hard problem. However, because our system measurements vary smoothly for inputs within a small neighborhood, the worst-case scenario is when multiple sources are in a contiguous block (*i.e.* nearby measurements are most similar, either by shift or scaling). Therefore, we compute the condition number of sub-matrices of $\boldsymbol{A}$ corresponding to a group of point sources with separation varying by integer numbers of voxels. We repeat this calculation for different numbers of sources. The results are shown in Fig. 4.5. As expected, the conditioning is worse when sources are closer together. In this case, increased noise sensitivity means that even small amounts of noise could prevent us from resolving the sources. This trend matches what we saw experimentally in Figs. 4.3 and 4.4.

Figure 4.5 also shows that the local condition number increases with the number of sources in the scene, as expected. This means that resolution will degrade as more and more sources are added. We see in Fig. 4.5, however, that as the number of sources is increased, the conditioning approaches a limiting case. Hence, the resolution does not become arbitrarily worse with increased number of sources. Therefore we can estimate the system resolution for complex objects from distiguishability measurements with a limited number of point sources. This is experimentally validated in Sec. 4.5, where we find that the experimental 16-point resolution is a good predictor of the resolution for a USAF target.

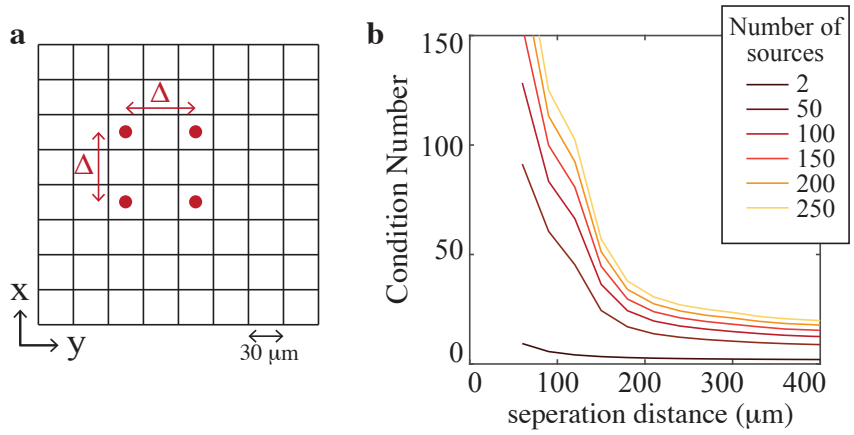

Figure 4.5: Our local condition number theory shows how resolution varies with object complexity. (a) Virtual point sources are simulated on a fixed grid and moved by integer numbers of voxels to change the separation distance. (b) Local condition numbers are plotted for sub-matrices corresponding to grids of neighboring point sources with varying separation (at depth 20 mm from the sensor). As the number of sources increases, the condition number approaches a limit, indicating that resolution for complex objects can be approximated by a limited number (but more than two) sources.

Unlike the traditional two-point resolution metric, our new local condition number theory explains the resolution loss we observe experimentally. Since many optical systems are locally shift invariant, we believe that it is sufficiently general to be applicable to other computational cameras that use nonlinear algorithms, which likely exhibit similar performance loss.

4.5 Experimental Results

Images of two objects are presented in Fig. 4.6. Both were illuminated using broadband white light and reconstructed with a 3D TV regularizer. We choose a reconstruction grid that approximately Nyquist samples the two-point resolution (by 2×2 binning the sensor pixels to yield a 1.3 megapixel measurement). Calibration images are taken at 128 different z-planes, ranging from z=10.86mm to z=36.26mm (from the diffuser), with spacing set according to conditions outlined in the resolution analysis of Sec. 4.4. The 3D images are reconstructed on a 2048×2048×128 grid, but the angular FoV restricts the usable portion of this grid to the center 100 million voxels. Note that the resolvable feature size on this reconstruction grid can still vary based on object complexity.

The first object is a negative USAF 1951 fluorescence test target, tilted 45° about the y-axis (Fig. 4.6a). Slices of the reconstructed volume at different z planes are shown in order to highlight the system's depth sectioning capabilities. As described in resolution

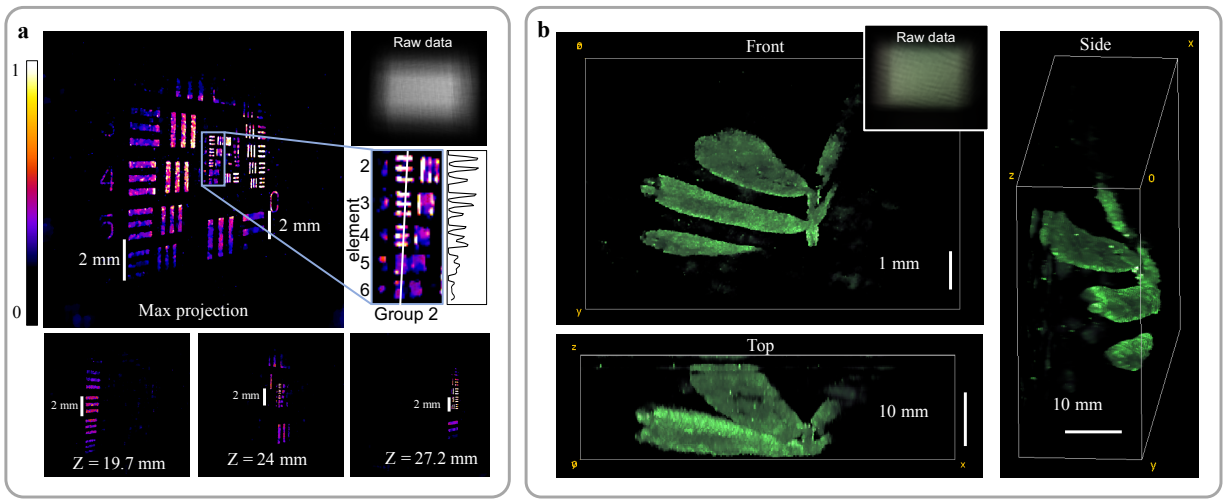

Figure 4.6: Experimental 3D reconstructions. (a) Tilted resolution target, which was reconstructed on a 4.2 MP lateral grid with 128 z-planes and cropped to 640×640×50 voxels. The large panel shows the max projection over z. Note that the spatial scale is not isotropic. Inset is a magnification of group 2 with an intensity cutline, showing that we resolve element 5 at a distance of 24 mm, which corresponds to a feature size of 79 μm (approximately twice the lateral voxel size of 35μm at this depth). The degraded resolution matches our 16-point distinguishability (75 μm at 20 mm depth). Lower panels show depth slices from the recovered volume. (b) Reconstruction of a small plant, cropped to 480×320×128 voxels, rendered from multiple angles.

portion of Sec. 4.4, the spatial scale changes with depth. Analyzing the resolution in the vertical direction (Fig. 4.6a inset), we can easily resolve group 2 element 4 and barely resolve group 2 element 5 at z=24mm. This corresponds to resolving features 79μm apart on the resolution target. This resolution is significantly worse than the two-point resolution at this depth (50μm), but similar to the 16-point resolution (75μm). Hence, we reinforce our claim that two-point resolution is a misleading metric for computational cameras, but multi-point distinguishability can be extended to more complex objects.

Finally, we demonstrate the ability of DiffuserCam to image natural objects by reconstructing a small plant (Fig. 4.6b). Multiple perspectives of the 3D reconstruction are rendered to demonstrate the ability to capture the 3D structure of the leaves.

Chapter 5

Efficient Modeling and Calibration of Spatial Variance

In Chapter 3, we introduced the *convolution model* which enabled efficient computation of the DiffuserCam forward model. This efficiency was critical for practically solving the large-scale inverse problem when we used DiffuserCam for 3D volume recovery, and it allowed easy calibration with just a single PSF for 2D photography or a z-sweep of PSFs for 3D imaging. However, the shift-invariance assumption that leads to the convolution model is only valid at small angles of incidence since it is based on the paraxial (small angle) approximation. In this chapter, we consider additional causes of spatial variance and present two strategies for efficient modelling: an approach we'll call the *local convolution model* and an approach based on a low rank approximation using principal component analysis (PCA). We'll also explore an efficient strategy based on blind deconvolution for practical calibration of systems with spatial variance.

Validity of the Convolution Model

We start by exploring the validity of the convolution model. Figure 5.1a-c shows registered close-ups of experimentally measured DiffuserCam PSFs from plane waves incident at 0°, 15° and 30°. The convolution model assumes that these are all exactly the same, though, in reality, they have subtle differences. To quantify the similarity across the FoV, we plot the inner product between each off-axis PSF and the on-axis PSF (see Fig. 5.1d). The inner product is greater than 75% across the entire FoV and particularly good within ±15° of the optical axis, indicating that the convolution model holds relatively well, but not exactly, for the DiffuserCam system presented in Chapter 4.

To investigate how the spatial variance of the PSF impacts system performance, we use the peak width of the cross-correlation between the on-axis and off-axis PSFs to approximate the spot size off-axis. Figure 5.1e (solid) shows that we retain the on-axis resolution up to ±15°. Beyond that, the resolution gradually degrades. The rest of this chapter discusses procedures for bridging the gap between exhaustive calibration and the convolution model in a computationally efficient way.

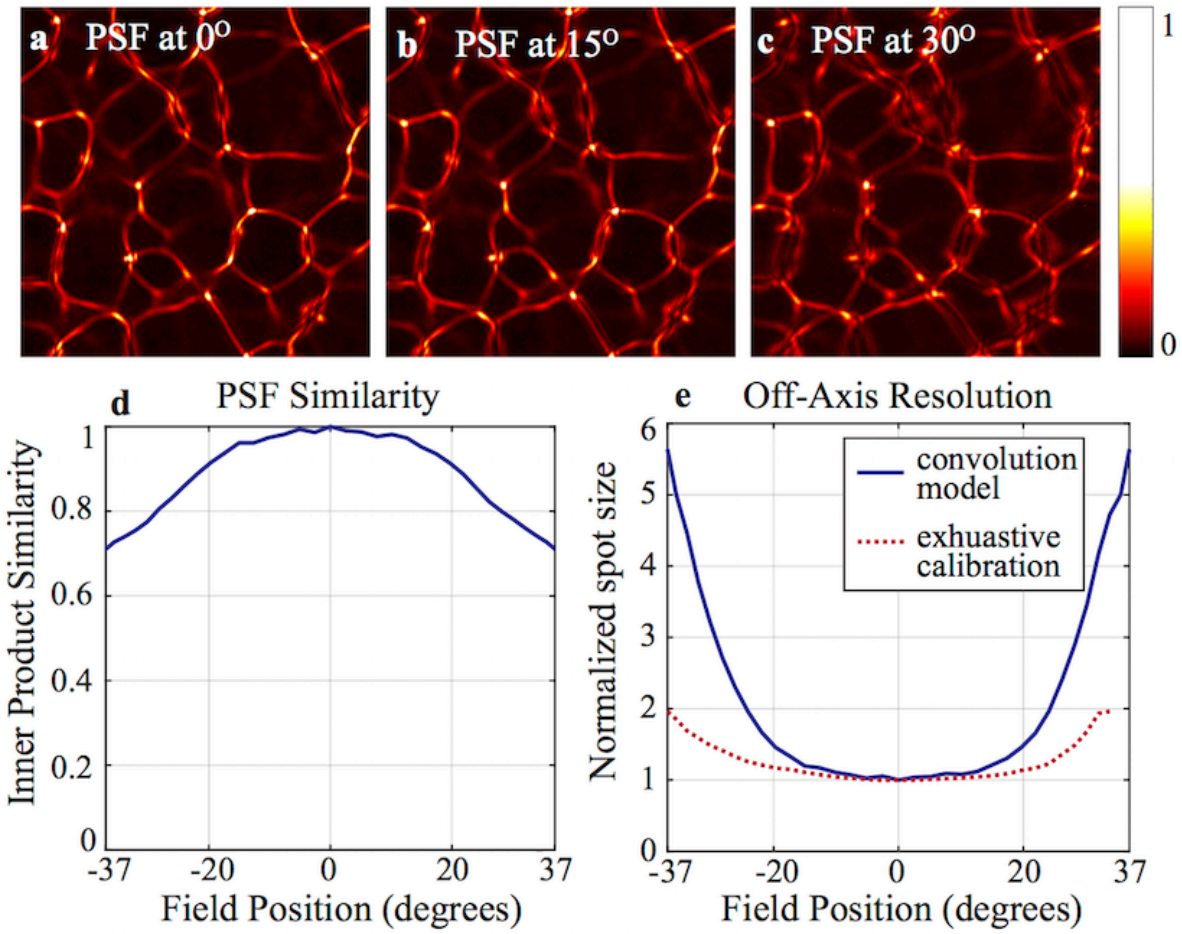

Figure 5.1: Experimental validation of the convolution model. (a)-(c) Close-ups of registered experimental PSFs for sources at 0°, 15° and 30°. The PSF at 15° is visually similar to that on-axis, while the PSF at 30° has subtle differences. (d) Inner product between the on-axis PSF and registered off-axis PSFs as a function of source position. (e) Resulting spot size (normalized by on-axis spot). The convolution model holds well up to ±15°, beyond which resolution degrades (solid). Exhaustive calibration would improve the resolution (dashed), at the expense of complexity in computation and calibration.

5.1 Causes of Spatial Variance

Aberrations

The shift-invariance assumption arises from the linearization of Snell's law via the small angle approximation. However, as one can see in Fig. 5.1a-c, Snell's law is not linear, resulting in differences in the PSF at different angles, which we'll refer to as aberrations. Furthermore, higher angles also result in a longer propagation distance between the diffuser and sensor,

contributing to changes in the PSF analogous to field curvature, which explains why Fig. 5.1c looks similar to the on-axis PSF at a longer propagation distance.

In addition, any other optics in the system (e.g. color filters for fluorescence, lenses when the diffuser is integrated into a microscope [92, 160]) can also have abberations. As described by Liu et al. [92], if the exit pupil of a micrscope objective does not stay stationary, than the PSF will not be shift-invariant, since different parts of the diffuser are illuminated for different field locations. As described by Yanny et al. [160], if the exit pupil has phase aberrations (e.g. coma), we see warping and distortion of the PSF.

Sensor fall-off

Another component that can cause spatially varying PSFs is the response of the image sensor. Digital image sensors exhibit angle-sensitive responses due to cosine falloff, microlenses on the pixels, and circuitry that blocks light [146]. Furthermore, not all pixels necessarily behave the same way: some sensors have deliberately different microlens locations to account for the specific design inside a camera. In addition, many designs have circuity that is grouped together for neighboring pixels, resulting in shadowing effects that are not the same for each location. The most complete model would account for the change in sensor response over angle (2D) and position on the sensor (2D). Due to the challenge in calibrating the 4D function, we assume a simplified model in which the sensor response is summarized by $f(\theta)$ where θ is the angle between the incident light and the normal to the sensor. This model is best for backside-illuminated sensors, in which circuitry is beneath the photodiode and does not create spatial variation in the sensor response. Based on the geometry of the system, the PSF due to the sensor response alone is

$$h_{\text{sensor}}(\vec{u};\vec{x},z) = f\left(\tan^{-1}\left(\frac{||\vec{u}-\vec{x}||}{d+z}\right)\right).$$

where $||\cdot||$ is the euclidean distance.

Assuming that the diffuser does not bend light by a substantial amount compared to the frequencies in $f(\theta)$, we can model the total PSF, $h(\vec{u};\vec{x},z)$, as the product of the sensor and paraxial PSF, yielding the following "two-part" model:

$$h(\vec{u};\vec{x},z) = h_0\left(\vec{u}+\frac{d}{z}\vec{x};z\right) f\left(\tan^{-1}\left(\frac{||\vec{u}-\vec{x}||}{d+z}\right)\right). \tag{5.1}$$

Note that when the distance to the object, $d+z$, is much larger than the sensor size (recall that $\vec{u}$ is constrained to be within the sensor size), the sensor falloff term reduces to

$$h_{\text{sensor}}(\vec{u};\vec{x},z) = f\left(\tan^{-1}\left(\frac{||\vec{x}||}{d+z}\right)\right),$$

which is independent of $\vec{u}$. Therefore, when the object is far from the sensor, as it was in Chapter 3 and Chapter 4, the falloff term can be implicitly grouped with the sample, and the total PSF consists only of the shift-invariant component due to the diffuser.

The factorization in Eq. 5.1 suggests that all PSFs can be synthesized from the two underlying functions $h_0(\vec{u}; z)$ and $f(\theta)$. However, recall that $h_0(\vec{u}; z)$ is an approximation which assumes no spatially-varying aberrations in the diffuser. In practice, due to spatially-varying aberrations, there is no single $h_0(\vec{u}; z)$ that accurately describes the diffuser's pattern for all field positions. Despite its inaccuracies, we'll see shortly that this "two-part" model is still useful for analysis purposes.

5.2 Interpolating PSFs for Efficient Spatial Variance Modeling

If we measured the PSFs $h(\vec{u}; \vec{x}, z)$ at every possible 3D location in the volume it would capture all aspects of spatial variance in the system. However, this is excessive and unnecessary due to the slowly-varying nature of the diffuser aberrations and sensor falloffs. Furthermore, it is incredibly impractical to capture (or simulate) a PSF at every location since, for 3D, there may be many million possible positions. Instead, we take advantage of the slowly-varying aberrations by collecting only a sparse grid of calibration measurements and interpolating between them.

We denote the i-th calibration measurement taken at $(\vec{x}_i, z_i)$ as $h_i(\vec{u}) = h(\vec{u}; \vec{x}_i, z_i)$. Our goal is to interpolate between these calibration measurements to synthesize every PSF in the volume. However, naive pixel-wise averaging of neighboring calibration measurements will result in inaccurate blurring of the high-frequency diffuser pattern since it fails to account for the dominant effects on the PSF (lateral translation and axial scaling) when the source moves. Therefore, instead of interpolating between the raw measurements, we first computationally register the calibration measurements to the on-axis PSF taken at depth z by applying the following shifts and scales:

$$\tilde{h}_i(\vec{u}; z) = h_i(s_i\vec{u} - s_i\Delta\vec{u}_i) \tag{5.2}$$

where

$$s_i = \frac{z(d + z_i)}{z_i(d + z)}, \qquad \Delta\vec{u}_i = \frac{d}{zs_i}\vec{x}_i.$$

The registered calibration measurements, denoted $\tilde{h}_i(\vec{u}; z)$, have the paraxial component of the diffuser pattern aligned. After this transformation, pixel-wise interpolation between neighboring calibration measurements preserves the high-frequency features of the diffuser caustics. In a shift-invariant system, all of the registered PSFs are identical and no interpolation is necessary; in a system with spatial variance, the variations appear as smoothly changing deviations between the registered PSFs. Therefore, we synthetically generate intermediate PSFs by linearly interpolating between the registered calibration measurements. We first consider the case where all calibration PSFs are at a single depth, z, which yields $s_i = 1$ for all i. The synthetically generated PSF from an arbitrary point $(\vec{x}, z)$ is approximated

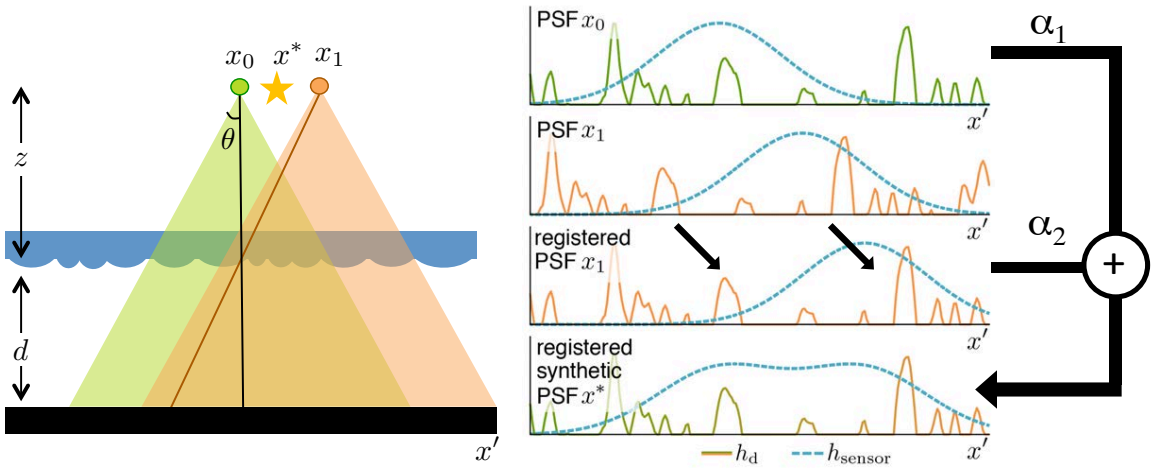

Figure 5.2: Schematic depicting interpolation between PSFs under the assumption that all spatial variance is due to sensor fall-off. Consider two calibration points located on-axis at x_0 and off-axis at x_1. For clarity, we assume a simplified two-part model, where each PSF consists of a high frequency signal due to the diffuser (h_{d}) multiplied by the angular-dependent sensor response (h_{sensor}). In the off-axis PSF, h_{d} is translated to the left while h_{sensor} remains directly under the point source. To interpolate between the two calibration measurements, first we register the component due to the diffuser. Then we approximate the PSF at x^* by taking a linear combination of the registered PSFs where the weights (α) are based on the distance between the calibration location and x^*. We extend this to 2D using bilinear interpolation.

from the calibration measurements:

$$h\left(\vec{u};\vec{x},z\right) \approx \sum_i \alpha_i(\vec{x},z)\,\tilde{h}_i(\vec{u}+\frac{d}{z}\vec{x};z). \tag{5.3}$$

Here $\alpha_i(\vec{x},z)$ is a weighting factor determined by the distance to the i-th calibration measurement. We choose $\alpha_i(\vec{x},z)$ to correspond to bilinear interpolation between the four nearest PSFs. The procedure for generating synthetic PSFs is visually summarized in Fig. 5.2.

It is also possible to synthesize new PSFs at different depths using an analogous procedure. To generate PSFs at a new depth z^*, nearby calibration measurements are first scaled based on Eq. 5.2 to generate $\tilde{h}_i(\vec{u};z^*)$. Then, the synthetic PSF at z^* is calculated from a linear combination of these scaled calibration measurements, where the weight is once again based on the synthetic PSF's proximity to the calibration points.

Although Eq. 5.2 requires the precise locations of the calibration measurements, we can circumvent this requirement by directly determining $\Delta\vec{u}_i$ from the measurements themselves. We find that cross-correlation between neighboring calibration measurements is maximized when the diffuser component of the PSF is aligned. Therefore, we choose a central PSF to act as the on-axis PSF, then calculate $\Delta\vec{u}_i$ by determining the translation that maximizes the

cross-correlation with this central PSF. We find this approach more robust than physically measuring the translation.

Local convolution model

By applying Eq. 5.3 we can generate the complete set of calibration measurements needed for the system matrix $\boldsymbol{A}$. However, its large size makes $\boldsymbol{A}$ computationally inefficient to generate and store . Luckily, the linear structure of Eq. 5.3 allows us to form what we call the *local convolution model* which models the raw data without explicitly generating every PSF. When we plug Eq. 5.3 into Eq. 4.3, the convolutional structure becomes apparent:

$$\begin{aligned} b(\vec{u}) &= \boldsymbol{C} \sum_{(z,i)} \sum_{\vec{x}} \alpha_i(\vec{x}, z)\, y(\vec{x}, z)\, \tilde{h}_i(\vec{u} + \frac{d}{z}\vec{x}; z) \\ &= \boldsymbol{C} \sum_{(z,i)} \left[\alpha_i\left(-\frac{z}{d}\vec{u}, z\right)\; y\left(-\frac{z}{d}\vec{u}, z\right)\right] * \tilde{h}_i\left(\vec{u}; z\right). \end{aligned} \tag{5.4}$$

Here $*$ denotes a 2D convolution in the sensor coordinates; recall that the sensor coordinates, $\vec{u}$, and world coordinates, $\vec{x}$, are related by the system magnification, $m = d/z$. In Eq. 5.4 we assume that we have calibration measurements at every depth of interest since we can generate PSFs at new depths using the procedure outlined in the previous section.

As before, we can vectorize the elements of Eq. 5.4 and write the forward model using matrix operators.

$$\begin{aligned} \vec{\boldsymbol{b}} =& \boldsymbol{A}\vec{\boldsymbol{y}} \\ & \boldsymbol{C} \sum_{(z,i)} \vec{\boldsymbol{h_i}} * \operatorname{diag}(\vec{\boldsymbol{\alpha}}_{\boldsymbol{i,z}})\vec{\boldsymbol{y}} \\ =& \boldsymbol{C} \sum_{(z,i)} \boldsymbol{F}^{-1} \operatorname{diag}(\boldsymbol{F}\vec{\boldsymbol{h}}_{\boldsymbol{i,z}}) \boldsymbol{F} \operatorname{diag}(\vec{\boldsymbol{\alpha}}_{\boldsymbol{i,z}})\vec{\boldsymbol{y}} \end{aligned} \tag{5.5}$$

See Table 2.1 in Chapter 2 for a summary of basic operators as matrices.

With this model we can efficiently interpolate between the calibration measurements as we compute the forward model. We refer to this as the *local convolution model* because we can think of $\alpha_i(\vec{x}, z)$ as choosing a region around the i-th calibration measurement where the measurement is valid, then performing a convolution in this region. If the support of the object is known, computational efficiency can be further improved by using the subset of the calibration measurements corresponding to the 3D object support.

Any interpolation scheme that can be written in the form of Eq. 5.3 is compatible with the local convolution model, which has the distinct advantage of only requiring that the calibration measurements themselves be stored in memory. Although other interpolation schemes could be used in place of Eq. 5.3, they would require pre-computing and storing every possible PSF in the volume, using on the order of 10,000$\times$ more memory. Computing PSFs on-the-fly is too computationally expensive for practical use in an iterative optimization process.

Low rank approximation with PCA

In the local convolution model, the $\vec{\boldsymbol{\alpha}}_i$'s each represent a pre-determined selection function that chooses a local region of the FoV in which a given PSF is valid. However, PSFs on opposite sides of the FoV may have similarities, which might be exploited for faster computation. Here, we describe how to use principle component analysis (PCA) to take advantage of this scenario. Note, this is a computational trick in which we do some pre-prossessing to reduce computation in the iterative inverse problem – it does not change the calibration costs since we still need to collect calibration measurements across the FoV.

Let's assume we collect a series of N calibration measurements $\vec{\boldsymbol{h}}_1 \dots \vec{\boldsymbol{h}}_N$ at distributed over the FoV (but all at a single depth z). We could directly use all of these measurements in the local convolution model descibed above, but each pass of the forward model then requires computing N convolutions for each depth (Eq. 5.4). However, since the PSFs are slowly-varying, we expect that a matrix containing all of the calibration measurements will be low rank. We first register the calibration measurements using Eq. 5.2, then we create a matrix $\boldsymbol{H}$ with each vectorized and registered calibration measurement as a column. Note that $\boldsymbol{H}$ has rank N since the there are N columns.

Using principle component analysis (PCA) we can approximate $\boldsymbol{H}$ as

$$\boldsymbol{H} \approx \sum_{i=1}^{k} \vec{\boldsymbol{p}}_i \vec{\boldsymbol{w}}_i^T, \tag{5.6}$$

where $k < N$. We calculate the $\vec{\boldsymbol{p}}_i$'s and $\vec{\boldsymbol{w}}_i^T$'s by taking the singular value decomposition (SVD) of $\boldsymbol{H}$; then, the $\vec{\boldsymbol{p}}_i$'s are the left-singular vectors associated with the k largest singular values, and the $\vec{\boldsymbol{w}}_i^T$'s are the corresponding right-singular vectors, multiplied by their singular values. There is a nice interpretation of this decomposition: the $\vec{\boldsymbol{p}}_i$'s represent the "principle PSFs" which form a basis for the PSFs across the FoV, and we'll denotes these as $p_i(\vec{u}; z)$. The $\vec{\boldsymbol{w}}_i^T$'s are spatially varying weights, which describe what linear combination of the principle PSFs to use at each spatially location. Note that PCA solves for the weights *only* at the locations in the FoV where the calibration points were acquired, but we can interpolate the weights between these points, to get a function $w_i(\vec{x}; z)$ that outputs the weight corresponding to the i^{th} component at field location $(\vec{x}, z)$. With this interpolated function, we can approximate the PSF from any location in the FoV as

$$h(\vec{u}; \vec{x}, u) = \sum_{i=1}^{k} w_i(\vec{x}; z) p_i(\vec{u} + \frac{d}{z}\vec{x}; z). \tag{5.7}$$

Since the $p_i(\cdot)$'s are, by definition, registered to the center PSF, we include the translation term $\frac{d}{z}\vec{x}$ to move the principle PSFs to their true location on the sensor.

We can see that Eq. 5.7 is exactly analogous to Eq. 5.3, but we've replaced the measured PSFs, $\tilde{h}_i$, with the principle PSFs found with PCA, p_i, and we're replaced the pre-defined weights, α_i, with the weights output by PCA, w_i. We can plug Eq. 5.7 into Eq. 5.4 to get the same convolutional structure as the local convolution model. However, importantly, if $\boldsymbol{H}$

is indeed low rank, the PCA model enables the same computation with only k convolutions instead of N convolutions.

We find that this PCA model works best when there is a lot of similarity between PSFs are different locations in the field-of-view, as in [160]. However, in the flat microscope presented in Chapter 6, the dominant spatially varying factor is the sensor fall-off, resulting in very different PSFs at different locations, so their is little gain from the PCA approach. Therefore, in Chapter 6, we will use the local convolution model presented above.

5.3 Sampling Requirements of the Local Convolution Model

The local convolution model relies on experimentally measuring the PSF at discrete calibration points, then interpolating to synthesize the remaining PSFs. If we known something about the rate that the PSFs are changing as a function of field position, than we can apply Nyquist sampling theory to determine the appropriate spacing between calibration measurements which fully captures the variance.

Lateral Sampling

To illustrate this idea, we consider the case where there are no aberrations in the diffuser (or any other optics) and the dominant factor creating spatial variance is the sensor response. For this analysis, we'll assume the "two-part" model from Eq. 5.1. Although this is a simplified model, it captures the dominant factors in the flat, lensless microscope that we'll describe in detail in Chapter 6, and it is convenient since we can characterize the aberrations with just a single 1D function, $f(\theta)$.

For this analysis, we assume that all calibration PSFs are captured at the same axial location, $z_i = z$ for all i. Registering the calibration measurements based on Eq. 5.2 aligns the diffuser component of the PSF, but not the sensor falloff component. When we plug the two-part model from Eq. 5.1 into Eq. 5.2, we get the following expression for the aligned PSF:

$$\tilde{h}_i(\vec{u}; z) = h_0(\vec{u}; z) f\left(\tan^{-1}\left(\frac{||s_i\vec{u} - \frac{d}{z}\vec{x}_i - \vec{x}_i)||}{d+z}\right)\right) \tag{5.8}$$

Here, the second term, $f(\ldots)$, describes the spatial variance of the falloff function after registration. This is the function that we are interpolating, so we must Nyquist sample this term to enable robust interpolation of PSFs. Since we assumed the sensor component is rotationally symmetric and all pixels have the same falloff response (a good assumption for backside-illuminated sensors), we reduce our analysis to the 1D case and only consider a single representative pixel at $\vec{u} = 0$. This yields the registered falloff function $\tilde{f}(x)$:

$$\tilde{f}(x) = f\left(\tan^{-1}\left(\frac{x}{z}\right)\right). \tag{5.9}$$

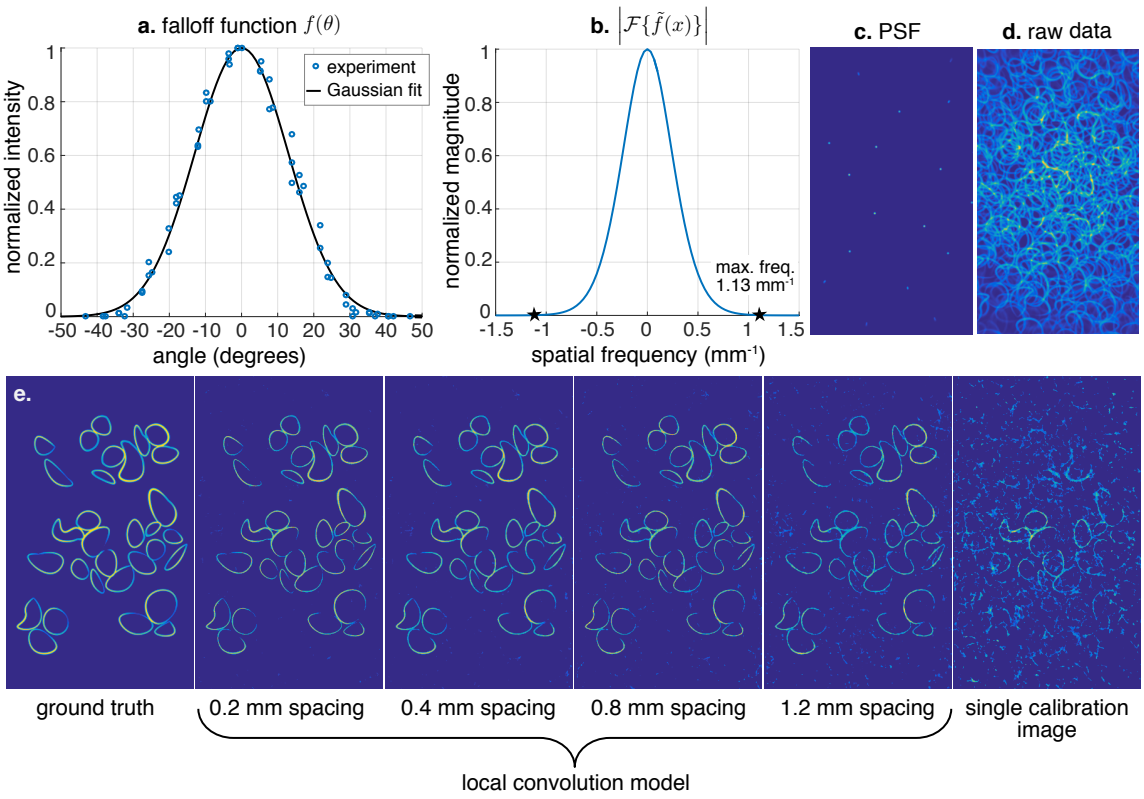

Figure 5.3: When the sensor response is the primary source of spatial variance, the number of calibration measurements needed for our local convolution model is determined by the angular falloff of the sensor. (a) Experimentally measured angular falloff, $f(\theta)$, fit to a Gaussian curve with $\sigma = 13°$. (b) Fourier transform after $f(\theta)$ is transformed based on Eq. 5.9, which is used to determine the maximum frequency, and thus the Nyquist sampling, in order to determine the necessary calibration sampling. (c) To test this, we simulate PSF measurements and (d) raw data using the Gaussian approximation of $f(\theta)$. (e) We then reconstruct the sample using the local convolution model with varying spacing between the calibration measurements. When calibration images satisfy the Nyquist sampling (0.4 mm apart), the model performs well, but when samples are spaced further apart, the reconstruction degrades.

If $\tilde{f}(x)$ is sampled at the Nyquist frequency, we can robustly interpolate between samples at different positions. To determine the lateral sampling requirements, we experimentally measure $f(\theta)$, estimate its bandwidth, and use Eq. 5.9 to determine the Nyquist sampling period.

To test this, we simulate raw data using an experimentally measured $f(\theta)$ (shown in Fig. 5.3), then deconvolve using the local convolution model. We simulate calibration measurements at varying spacings and find that when the calibration measurements are at the Nyquist sampling rate or closer together, we get good reconstruction quality. However,

as the measurements move further apart, substantial artifacts appear. Finally, if a single calibration measurement is used, without any model-based interpolation as in Chapters 3 and 4, the reconstruction fails, demonstrating the necessity of the local convolution model when there is significant sensor fall-off.

Axial Sampling

For 3D imaging, we also need to capture PSFs from different depths. Here we consider the axial sampling for the local convolution model. Axial changes in the PSF are primarily due to aberrations, particularly defocus, and therefore cannot be explained with the "two-part model" alone. However, by assuming the diffuser has a microlens structure with lenslets of focal length f, we can determine the axial sampling based on the depth-of-field. If the radius of the circle of confusion, $\frac{p(d-f)}{2f}\left|1-\frac{f\,d}{z(d-f)}\right|$, changes by no more than the diffraction-limited spot size, $\frac{\lambda}{\mathrm{NA}}$, between neighboring samples, then we have fully sampled the axial defocus function. This yields the condition

$$\begin{aligned}\frac{pd}{2}\left(\frac{1}{z_1}-\frac{1}{z_2}\right) &\leq \frac{\lambda}{\mathrm{NA}} \approx \frac{2d\lambda}{p} \\ \frac{1}{z_1}-\frac{1}{z_2} &\leq \frac{4\lambda}{p^2},\end{aligned} \tag{5.10}$$

where z_1, z_2 are the axial locations of neighboring calibration images, p is the microlens diameter, and λ is the wavelength of light.

5.4 Compressive Calibration using Blind Deconvolution

Here, we present a more data-efficient alternative in which the spatially-varying PSFs are recovered from a small number of calibration images containing multiple fluorescent beads. These multiplexed measurements simultaneously sample many PSFs from across the FoV in each acquisition which enables recovery of more PSFs than measurements, resulting in a more accurate model of the system. Furthermore, the measurements are physically easy to acquire since the beads can be randomly distributed and their locations do not need to be known, eliminating the need for a precision motion stage. We computationally acquire both the bead locations and spatially-varying PSFs from just the raw calibration measurements by solving a blind deconvolution problem with a rank constraint.

We once again assume we collect N calibration measurements, but this time, each measurement is of a small number of fluorescent beads instead of a single point source (Fig. 5.4b). In each measurement the beads are in a different, random location and the number of beads is not know *a priori*. We model the i-th calibration measurement, denoted $\vec{\boldsymbol{b}}_i$, using the local convolution model as follows

$$\vec{\boldsymbol{b}} = f(\boldsymbol{H}, \vec{\boldsymbol{y}}_i) = \sum_j \vec{\boldsymbol{h}}_j * (\boldsymbol{M}_j\, \vec{\boldsymbol{y}}_i)\,. \tag{5.11}$$

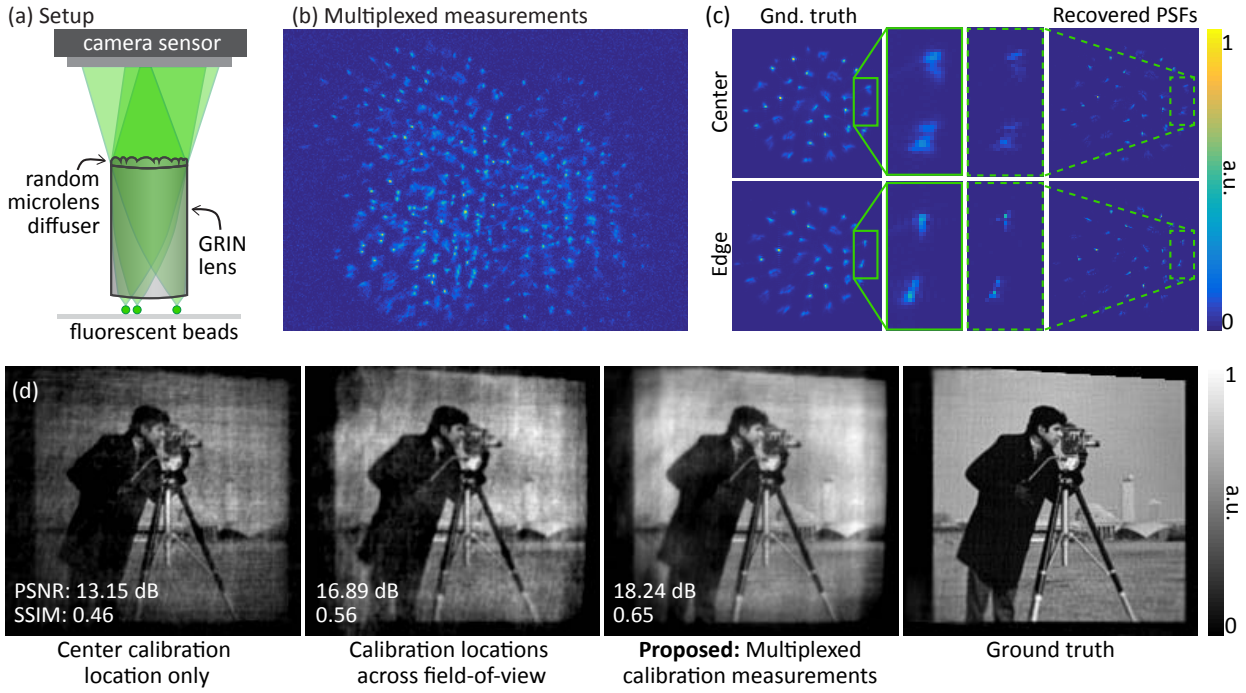

Figure 5.4: Spatially varying PSFs can be recovered from a set of multiplexed calibration measurements of several point sources at unknown locations. We simulate a random microlens diffuser at the pupil plane of a GRIN lens (a). Rather than capturing calibration images of a single point source, we envision capturing images of many randomly placed fluorescent bead, simulation shown in (b). From the multiplexed measurements we recover the spatially-varying PSFs (c), which show good correspondence to the ground truth PSFs. To test if the recovered PSFs are sufficiently accurate, we simulate raw data of a dense image and compare deconvolution results using the PSF at the center only (averaged over nine images to reduce noise), PSFs captured at nine locations over the FoV, and our recovered spatially-varying PSFs from nine multiplexed calibration measurements.

Here, $\vec{\boldsymbol{h}}_j$ is the PSF at position $\vec{x}$ in the FoV, $\boldsymbol{M}_j$ is a mask that selects a region around the point $\vec{x}_j$, and $\vec{\boldsymbol{y}}_i$ is the i-th vectorized calibration sample, specifically, the randomly positioned fluorescent beads. $\boldsymbol{H}$ is a matrix containing the PSFs, $\vec{\boldsymbol{h}}_j$, as its columns.

Both the sample, $\vec{\boldsymbol{y}}_i$, and PSFs, $\boldsymbol{H}$, are unknown, which leads to a multichannel blind deconvolution problem [138]. We solve for both unknowns simultaneously by alternating between solving the following minimization problems.

$$\hat{\boldsymbol{y}} = \underset{\vec{\boldsymbol{y}}}{\operatorname{argmin}} \, || \sum_i \vec{\boldsymbol{b}}_i - f(\boldsymbol{H}, \vec{\boldsymbol{y}}_i) ||_2^2 + \tau ||\vec{\boldsymbol{y}}||_1 \tag{5.12}$$

$$\hat{\boldsymbol{H}} = \underset{\boldsymbol{H}}{\operatorname{argmin}} \, || \sum_i \vec{\boldsymbol{b}}_i - f(\boldsymbol{H}, \vec{\boldsymbol{y}}_i) ||_2^2 \quad \text{s.t.} \quad \operatorname{rank}(\vec{\boldsymbol{H}}) \leq k \tag{5.13}$$

Here, $\vec{\boldsymbol{y}}$ is a vector containing all $\vec{\boldsymbol{y}}_i$, τ is a tuning parameter on the object sparsity, and k determines the degree of spatial variance of the PSFs. Solutions to Eqs. 5.12 and 5.13 are approximated using the fast iterative shrinkage threshholding algorithm for 100 iterations. After each alternating solve, k is increased. We initialize the PSF by aligning all the $\vec{\boldsymbol{b}}_i$ using cross correlation, then applying rank one approximation to the aligned stack to obtain a noisy PSF approximation. The $\vec{\boldsymbol{y}}_i$ are initialized at zero.

We use the "3D-Randoscope" introduced by Yanny et al. [160] as our model system to test our calibration method (Fig. 5.4a). This miniature 3D microscope consists of a random microlens diffuser attached to the back of a gradient index (GRIN) lens, which contains aberrations that create spatial variance. Starting with experimental calibration measurements from a Randoscope prototype, we build a simulation of the fully spatially-varying system at a single depth. With our simulation, we generate $N = 9$ images each containing 20 fluorescent beads at random locations in the FoV, and we simulate read noise such that the raw measurement PSNR is 34 dB (Fig. 5.4b).

From just these 9 measurements, we solve the blind deconvolution problem for 25 different $\vec{\boldsymbol{h}}_j$. An example of two recovered PSFs at different field locations is shown in Fig. 5.4c, displaying good but imperfect correspondence with ground truth. However, the true test is whether the recovered PSFs are capable of reconstructing images. Therefore, we simulate raw data of a dense image, then use our recovered spatially-varying PSFs to reconstruct the image (Fig. 5.4d). We compare with two other approaches used in prior work: (1) PSF calibration only at the center of the FoV, which implicitly assumes no spatial variance; for fair comparison, we average 9 PSFs in this case. (2) PSF calibration at 9 different locations across the FoV, as in [84], with smooth interpolation between measurements. Our proposed approach only uses 9 measurements, but recovers 25 unique PSFs, resulting in the highest quality reconstruction.

Chapter 6

Fluorescence Microscopy with a Random Microlens Diffuser

In this chapter[1], we extend DiffuserCam to fluorescence microscopy, to create a compact portable on-chip microscope capable of 3D imaging when the sample is sparse. To achieve microscopic resolution, the object is placed much closer to the sensor than in Chapter 4, creating high angles of incidence at the sensor that break the shift-invariant assumption used previously. Drawing on the results of Chapter 5, we describe our diffuser microscope using the local convolution model, which captures the spatial variance using $40,000\times$ fewer calibration measurements and $10,000\times$ less memory than a brute force approach. Furthermore, the local convolution model does not impose any restrictions on the diffuser; this gives us the freedom to design the diffuser for improved noise performance. Specifically, we introduce a new diffuser design, the *random microlens diffuser* which consists of many small lenslets arranged on the diffuser surface. We show that this design has better noise performance than the smooth diffuser or a regularly space microlens array. We fabricate a random microlens diffuser for our experimental prototype microscope and demonstrate $10\times$ resolution improvement compared to the prototype in Chapter 4.

6.1 Related Work

On-chip microscopy is a powerful imaging modality in which a digital image sensor captures information about the sample without using a traditional microscope objective. These lensless microscopes can be very compact and lightweight for portable or *in vivo* applications, and they typically have simpler hardware than their lensed counterparts. However, many on-chip microscopes are limited to bright-field microscopy [55, 108, 17] rather than fluorescence imaging, a critical modality for probing structure and function in a wide range of samples.

[1]This chapter is based on the published journal paper titled "On-chip fluorescence microscopy with a random microlens diffuser" and is joint work with Fanglin Linda Liu, Irene Grossrubatscher, Ren Ng, and Laura Waller [84].

As summarized in Greenbaum et al. [55], on-chip fluorescence imaging is challenging for several key reasons. First, fluorophores are incoherent with each other and with background illumination. As a result, digital holography [108, 17] and other interferometric methods cannot be applied. Shadow-based techniques [167, 38] are also not applicable because fluorescent samples do not necessarily block light. Furthermore, fluorophores emit light uniformly in all directions; in an on-chip system without a main lens, fluorophores therefore become dim and defocused as they move further from the sensor. This results in degradation of both signal-to-noise ratio (SNR) and resolution with increasing distance from the sensor. Prior on-chip microscopes for fluorescence [115, 35, 36, 130, 1] mitigate this effect by using very short working distances (less than 500 µm), limiting their applications to samples that can be placed directly on the sensor. In this work, we demonstrate an on-chip fluorescence microscope featuring a practical working distance of over 1.5 mm, suitable for imaging samples on slides or in microfluidic channels.

Our strategy for on-chip fluorescence microscopy involves placing a thin mask between the sample and the sensor. The mask modulates incoming light, indirectly encoding information about the sample, which can then be recovered computationally. Since the mask is placed close to the sensor (3.8 mm), it does not greatly increase the system form factor or hardware complexity (as compared to [134]). Such designs maintain the advantages of an on-chip lensless system and have been demonstrated successfully in both microscopy [1, 134] and photography [10, 82, 68, 140, 144, 63, 48], and have been shown to capture higher-dimensional information, such as three-dimensional (3D) [8] or temporal information [9], in a single acquisition.

Here, our mask is a random microlens diffuser [9, 92, 160] which has many small lenslets randomly arranged in 2D. Since the lenslets have focusing power, the best performance occurs when the object is in imaging condition with the sensor, enabling practical working distances, over 1.5 mm. In contrast, similar architectures with amplitude masks that have no focusing power [1, 10] have the best performance when the object is close to the mask, resulting in short working distances ($< 500\,\mu m$). Furthermore, unlike amplitude masks, our random microlens diffuser does not block light, making it better suited for fluorescent samples which are typically dim. As in [8], our system can recover 3D structures from a single acquisition; in this work, we demonstrate 8 µm lateral resolution and 50 µm axial resolution, an order of magnitude higher than in [8].

The architecture of our system has many parallels to a 4D light field camera [2, 110, 20] or an integral photography system [91, 66], which instead uses a periodic microlens array. Similar to a light field camera, each lenslet of our random microlens diffuser can be thought of as imaging the object from a different perspective. However, because our proposed system uses random rather than regular arrays of lenslets, cross-talk between the lenslets can be disambiguated computationally. This allows for increased flexibility in the design of the micro-optics, eliminates the need for a main objective lens, and enables a simple flat architecture that does not require physically isolating each lenslet, as in [144, 63, 69]. As a result, our system is easy to assemble, compact and portable (total size of $3.5\,cm \times 3.5\,cm \times 1\,cm$, limited by the board size of the sensor), and the architecture can easily be extended to larger sensor sizes.

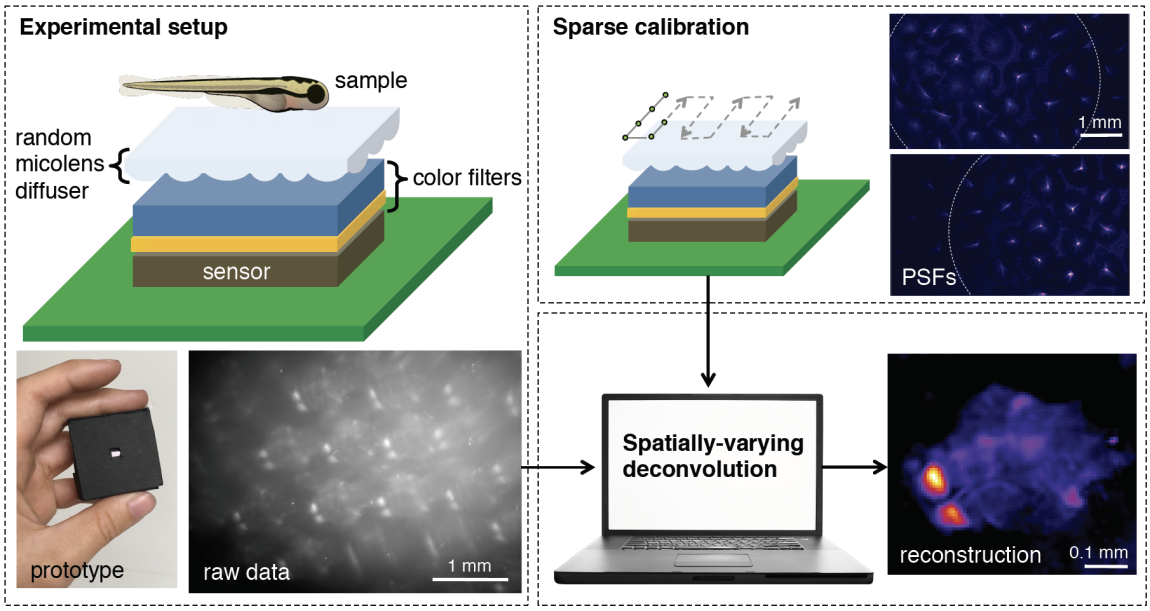

Figure 6.1: Our light-weight and portable on-chip microscope consists of a random microlens diffuser placed a few millimeters above an image sensor. Using only a sparse grid of calibration measurements, 3D images are reconstructed with a local convolution model that accounts for the spatially-varying PSFs.

6.2 Methods

Like DiffuserCam (Chapters 3 and 4), our microscope consists of a thin refractive diffuser placed a few millimeters in front of a traditional image sensor, with additional color filters between the diffuser and sensor to block excitation light (Fig. 6.1). Emitted light from each fluorophore in the sample is refracted by the diffuser surface to form a high-contrast pattern on the sensor. Every position in 3D creates a unique pattern, or PSF. We model our scene as a collection of point sources with varying intensity and no occlusions, so we can describe the imaging system with the following linear equation:

$$b(\vec{u}) = \mathbf{C} \sum_{(\vec{x},z)} h(\vec{u};\vec{x},z)\, y(\vec{x},z) + g(\vec{u}). \tag{6.1}$$

As before, $\vec{u}$ represents the 2D coordinates at the sensor, $\vec{x}$ represents the 2D lateral coordinates in the world, and z is the axial distance in the world. $b(\vec{u})$ is the measurement at position $\vec{u}$ on the sensor, $h(\vec{u};\vec{x},z)$ is the PSF taken at position $(\vec{x},z)$ in the world, $y(\vec{x},z)$ is the sample intensity, $\mathbf{C}$ is a crop operator that accounts for the finite sensor size, and $g(\vec{u})$ is the background due to unattenuated excitation light and autofluorescence from the color filters. Vectorizing the sample, measurement, and background allows Eq. 6.1 to be written compactly in matrix form: $\vec{\boldsymbol{b}} = \boldsymbol{A}\vec{\boldsymbol{y}} + \vec{\boldsymbol{g}}$. To recover the object from the sensor measurement,

we jointly estimate the sample fluorescence, $\vec{\boldsymbol{y}}$, and background, $\vec{\boldsymbol{g}}$, by solving the regularized least squares problem described in Sec. 6.2.

Forward model

Reconstructing the sample requires knowing the matrix $\boldsymbol{A}$, or equivalently, the PSFs for every point in 3D space. Prior work [8] assumed that the distance between the object and sensor was large relative to the sensor size, making the PSF shift-invariant at each depth; this enabled $h(\vec{u}; \vec{x}, z)$ to be fully characterized by only one calibration measurement per axial location and enabled $\vec{\boldsymbol{b}}$ to be efficiently computed with convolutions. In this work, we place objects closer to the sensor in order to achieve microscopic resolution; however, this breaks the shift-invariance assumption and necessitates accounting for the angular dependence of the sensor.

To capture the shift-varying effects, we use the local convolution model described in Chapter 5 and we calibrate the system by measuring a sparse grid of PSFs in 3D and interpolating between them. Using the sampling guidelines described in Sec. 5.3, we find that the local convolution model enables image reconstruction with $40,000\times$ fewer calibration samples than a brute force approach. We also tried the PCA model to further improve computation, but found little gain for the PSFs in this system, which suggests there is little similarity between PSFs at different FoV locations. Therefore, we stick with the local convolution framework with pre-defined interpolation kernels based on bilinear interpolation.

Inverse problem with background estimation

To recover the object from the raw data, we formulate a regularized inverse problem using the vectorized forward model from Eq. 5.5. Since many fluorescent samples have a sparse structure, we use an ℓ_1 loss on the object for regularization, enforcing sparsity in the native domain without a sparifying transformation. In addition, there is frequently autofluorescence and unattenuated excitation light hitting the sensor, which does not match our model and corrupts the reconstruction. Therefore we jointly estimate a low-frequency background component along with the object by solving the following minimization problem:

$$\begin{aligned} \hat{\boldsymbol{y}}, \hat{\boldsymbol{g}} = &\underset{\vec{\boldsymbol{y}}, \vec{\boldsymbol{g}}}{\operatorname{argmin}} \frac{1}{2} \|\vec{\boldsymbol{b}} - (\boldsymbol{A}\vec{\boldsymbol{y}} + \vec{\boldsymbol{g}})\|_2^2 + \tau \|\vec{\boldsymbol{y}}\|_1 \\ &\text{s.t.} \quad \vec{\boldsymbol{g}} \geq 0, \vec{\boldsymbol{y}} \geq 0, \boldsymbol{D}\vec{\boldsymbol{g}} = 0 \text{ outside low frequency support.} \end{aligned} \tag{6.2}$$

Here $\vec{\boldsymbol{g}}$ is the estimated background that cannot be well-explained by the forward model, $\boldsymbol{D}$ is the 2D discrete cosine transform (DCT) operator, and τ is a tuning parameter. Without constraints on $\vec{\boldsymbol{g}}$, a trivial solution to Eq. 6.2 is $\vec{\boldsymbol{y}} = 0$ and $\vec{\boldsymbol{g}} = \vec{\boldsymbol{b}}$. To prevent this, we constrain $\vec{\boldsymbol{g}}$ such that its DCT coefficients are zero outside some low-frequency support, typically the 5×5 lowest frequency components. We jointly solve for $\vec{\boldsymbol{y}}$ and $\vec{\boldsymbol{g}}$ using the fast iterative shrinkage-thresholding algorithm (FISTA) [12]. We find that it helps convergence to initialize the estimated background with a low-pass filtered version of the raw data.

Solving for a 3D sample from a single 2D measurement is an under-determined problem, and compressed sensing theory [42, 25, 24] can provide guidance regarding what samples will be accurately reconstructed. Without regularization, there are infinite possible 3D distributions that match the raw data. Therefore, the ℓ_1 regularization term and non-negativity constraints are critical to guide the optimization. As a result, we expect the highest-quality results when the sample matches the underlying assumptions, mainly that the sample is natively sparse. For dense samples, the object could be transformed into a different basis, as described in [25].

6.3 Random Microlens Diffuser

The local convolution model and the calibration scheme described above apply to a wide variety of diffuser designs; the only restrictions are that the spatially-varying aberrations in the PSF change smoothly as a function of position, and that no two points in the volumetric FoV generate identical PSFs. In contrast with prior works [1, 68] that require specific mask designs for image reconstruction, we have the freedom to design the diffuser to improve other aspects of the system, in particular, resolution, working distance, and noise sensitivity. Specifically, we propose using a random microlens diffuser, as in [9, 92, 160], which consists of small lenslets randomly arranged in 2D. To illustrate the advantages of the random microlens diffuser, we compare with a traditional microscope objective and two types of flat transparent masks: a smooth diffuser [8, 6] and a regular microlens array. Figures 6.2 and 6.3 show simulations of a small patch of the FoV for PSFs from each type of mask. Raw data for each PSF was simulated based on the two-part model in Eq. 5.1. In Fig. 6.2, the same quantity of Gaussian noise was added to each simulated measurement, and in Fig. 6.3, Poisson noise (shot noise) was simulated for varying number of collected photons. Each mask spreads photons differently over the sensor, so the number of photons per pixel at the sensor, and thus the shot noise performance, depends on the PSF. In both simulations, the noisy raw data was processed using the method described in Sec. 6.2. Background estimation was omitted and we assumed that the FoV was small enough that one calibration PSF per depth was sufficient.

Traditional microscope objectives are carefully optimized to capture high-quality images at the focal plane. They have good noise performance even at low photon counts for a 2D scene (Fig. 6.3, top row). However, recovering depth information from a single image is a poorly posed problem; as objects become defocused they lose high-frequency detail which may not be recovered, even with deconvolution (Fig. 6.2, top row).

Off-the-shelf diffusers [98] are convenient, inexpensive, and can easily be extended to larger sensor sizes. Lensless imagers made with these diffusers have a caustic pattern PSF (Fig. 6.2, second row). Due to the pseudorandom diffuser surface, any translation or scaling of the PSF *should* result in a substantially different pattern (i.e. the caustics should have a low inner product). However, the large amount of background light between the caustics causes an increased inner product which results in higher noise amplification during deconvolution (Figs. 6.2 and 6.3, second row). Other masks with low contrast patterns (e.g. amplitude

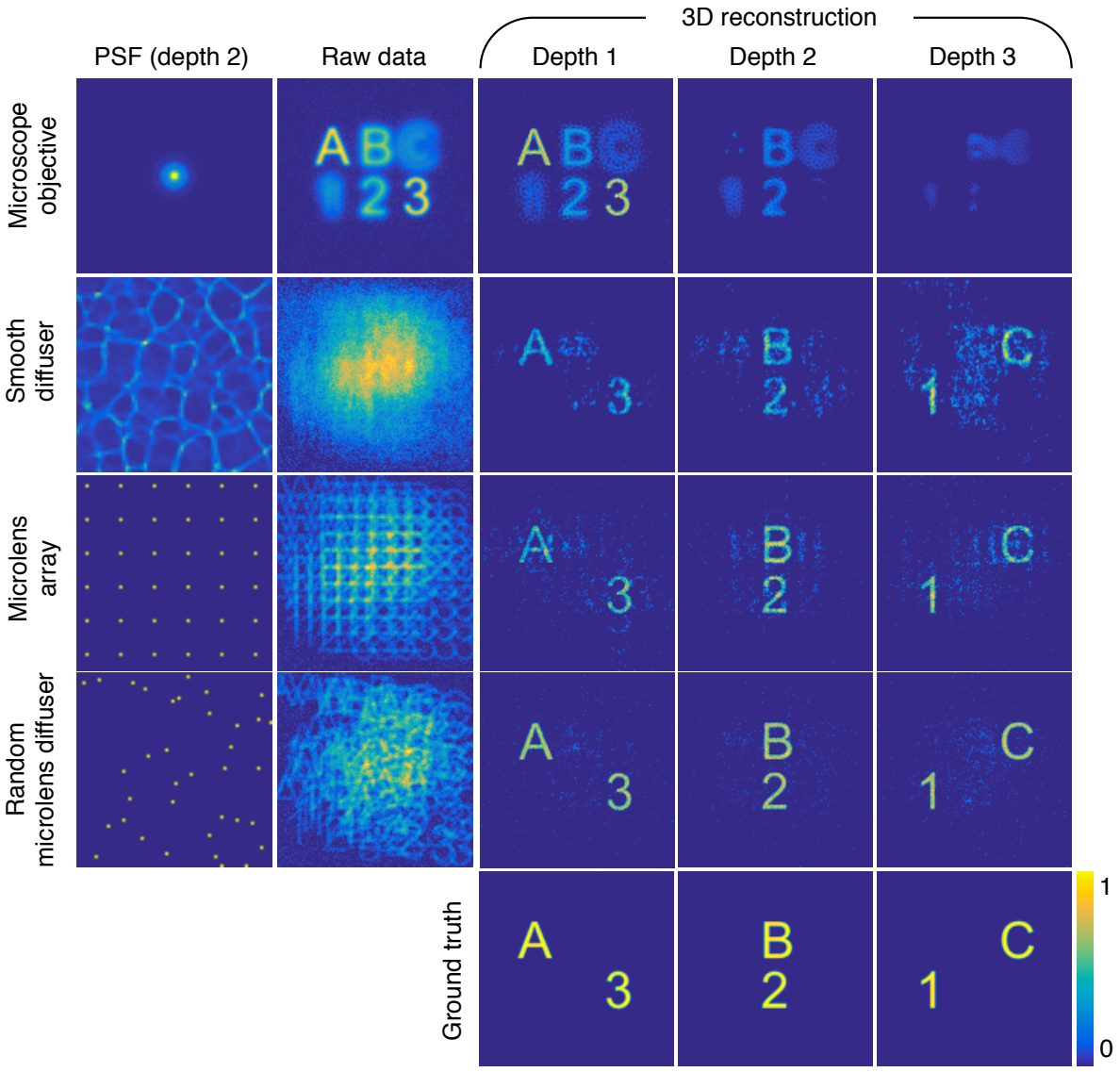

Figure 6.2: Simulation comparing PSFs for depth-resolved imaging. A microscope objective has good noise performance but fails to capture 3D information. A smooth diffuser's PSF has significant background light causing noise amplification, as does the periodicity of the regular microlens array. Our random microlens diffuser has a non-periodic PSF with high contrast, resulting in good noise performance and 3D reconstructions. All simulations have the same quantity of Gaussian noise added to the raw data.

masks, far field speckle) will suffer from similar noise amplification.

In comparison, a microlens array, which is widely used in light field microscopy, is designed to concentrate all incoming light into diffraction-limited spots beneath each lenslet, resulting in very little background light and a high-contrast pattern. However, if the PSF from a

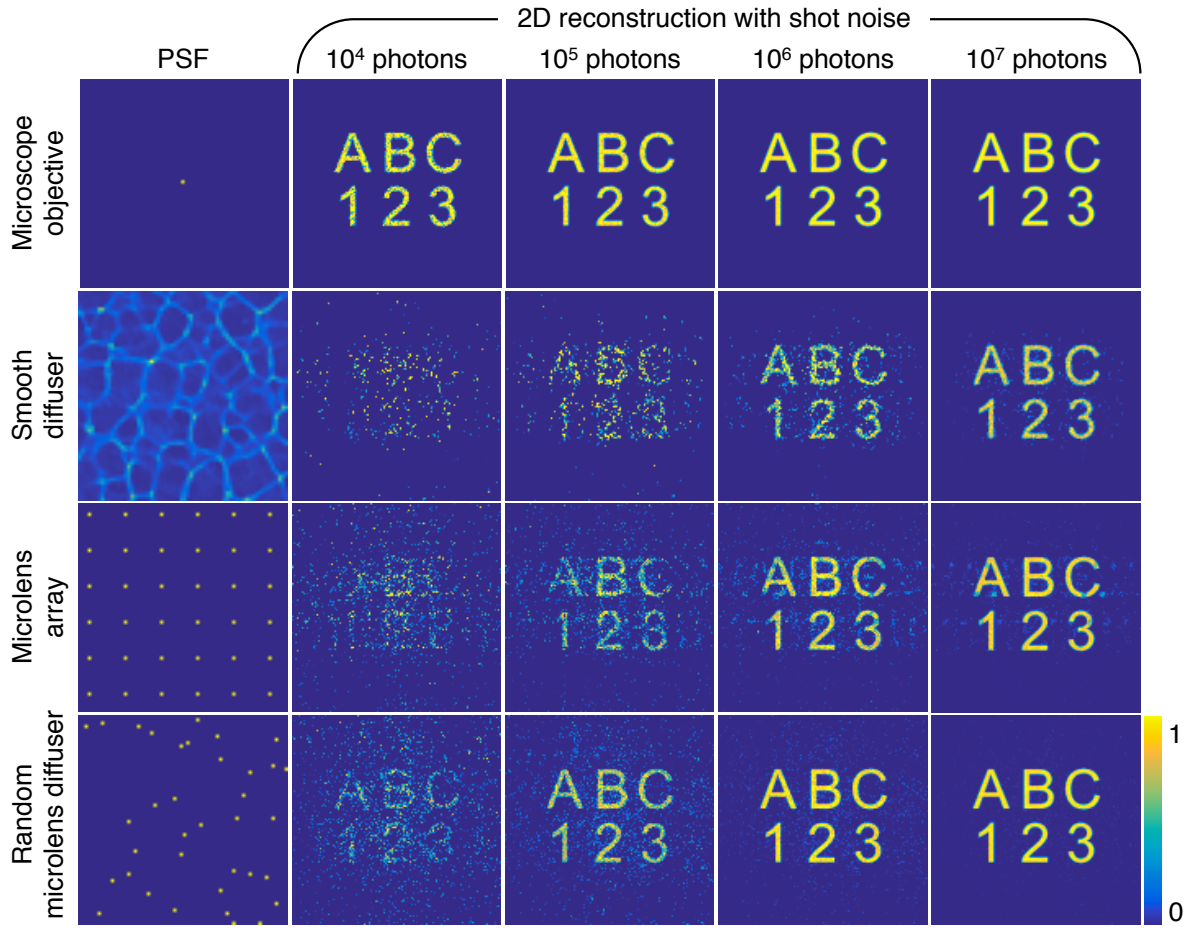

Figure 6.3: Simulation comparing PSF robustness to shot noise at a single depth. A microscope objective has the best noise performance for 2D imaging, but it does not extend to depth-resolved imaging nor to miniaturized systems. Of the PSFs with 3D capabilities, the random microlens diffuser is most robust to shot noise.

regularly-spaced microlens array is translated by exactly one period, the shifted PSF is a duplicate of the on-axis PSF, resulting in an increased inner product and higher noise amplification (Figs. 6.2 and 6.3, third row). Notice that the reconstruction shows periodic ghosting due to the regularity of the microlens array, and these artifacts are present even in low noise scenarios.

In our system, we use a random microlens diffuser which combines the best properties of the phase masks described above. Like the regular microlens array, our microlens diffuser has a high contrast PSF with low background, and, like the smooth diffuser, our PSF is pseudorandom without periodic ambiguity. The result is reduced noise sensitivity compared to the other flat masks (Figs. 6.2 and 6.3, third row). Although a traditional microscope objective has better noise performance at a single plane, our system is better suited for

miniaturization and enables reconstruction of 3D information from a single acquisition.

In addition, our microlens-based design is well-suited for resolution enhancement since the focal spot of each lenslet contains high spatial frequencies in all directions. Furthermore, it is easier to design diffraction-limited lenses when the diameter is small [95], allowing each lenslet to have nearly diffraction-limited performance with only a single spherical surface. Finally, since the microlenses focus light, the best performance is obtained when the object is in imaging condition with the sensor, so the lenslet focal length and distance to the sensor can be used to set a practical working distance, over 1.5 mm in our prototype.

6.4 Experimental Results

We built a prototype system using a backside-illuminated monochrome CMOS sensor (UI-3862LE with Sony IMX290 chip) and two color filters (Kodak Wratten #12 and Chroma ET525/50m) designed for green fluorescent probes ($\lambda = 520\,\text{nm}$). As described in [130], the combination of an absorption and an interference-based color filter is well-suited for removing excitation light at the high angles of incidence potentially present in our system, and any unfiltered light is removed with our computational background estimation (Sec. 6.2). We fabricated our random microlens diffuser with a droplet-based technique, similar to [100, 70], since these methods are known for good surface quality. Drops of optical epoxy (Norland 63) were cured on a hydrophobic surface, then transferred onto a glass coverslip to form the diffuser, generating lenslets with approximately $p = 250\,\mu\text{m}$ diameter. The diffuser was index-matched with polydimethylsiloxane (PDMS) to increase the microlens focal length to about 1.5 mm, and it was placed $d = 3.8\,\text{mm}$ away from the sensor. Based on these physical parameters and the sampling requirements outlined in Sec. 5.3, we require lateral samples every 400 µm and axial samples with $1/z_1 - 1/z_2 \leq 33\,\text{m}^{-1}$. Rather than sampling dioptrically, we choose to calibrate every 100 µm axially, which satisfies the axial sampling condition for objects $z = 1.7\,\text{mm}$ or further from the diffuser. Calibration images are captured with a 15 µm fluorescent bead, and the lowest frequency 10×10 DCT coefficients were set to zero for all calibration images before further processing to remove background light. Negative values after background subtraction were set to zero. For each calibration point, four measurements were taken and averaged to reduce noise. All images were downsampled by $2\times$ in each direction such that the equivalent pixel size is 5.8 µm.

To characterize the resolution of our system, we solve Eq. 6.2 on images of two points at varying separation distances, each generated by summing images of a single fluorescent bead. For axial characterization, images are of a 15 µm bead, moved in 10 µm increments; for lateral characterization, we use a 5 µm bead, moved in 2 µm increments. We define the resolution to be the minimum spacing at which there is at least a 20% dip in intensity between neighboring points in the reconstruction. Figure 6.4a summarizes our results, demonstrating 8 µm lateral resolution and under 50 µm axial resolution. We further test our system with a fluorescent USAF resolution target, $z = 2.56\,\text{mm}$ from the diffuser, shown in Fig. 6.4b. We can clearly resolve group 6, element 1 with 7.8 µm bars, which matches our two-point resolution experiments and demonstrates an order of magnitude improvement over our previous

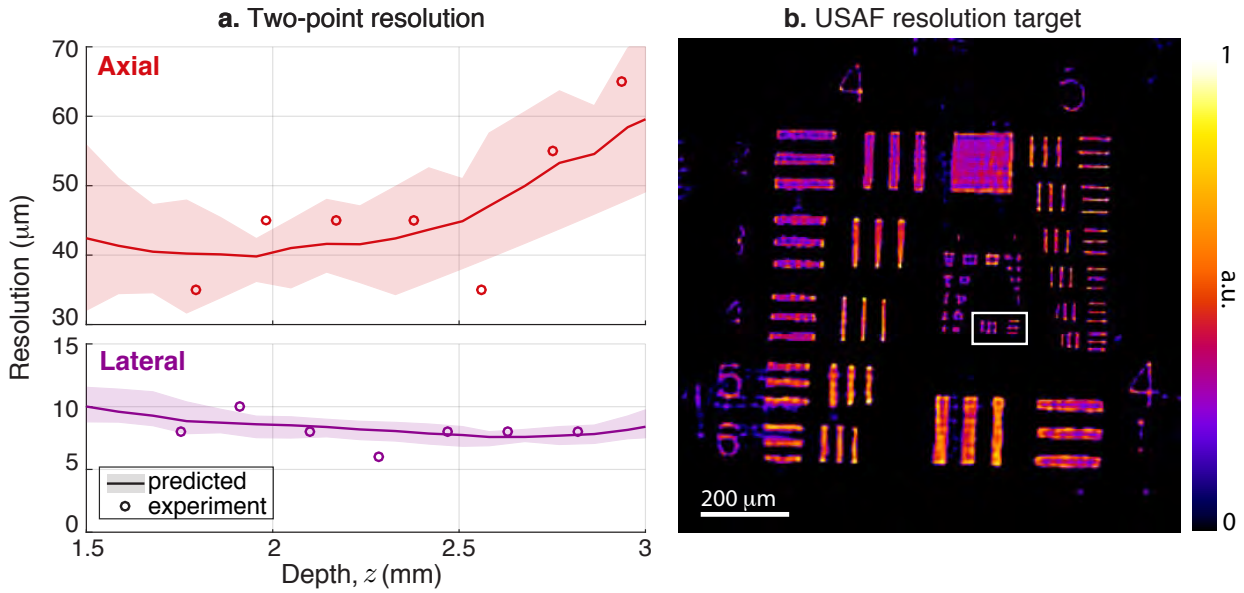

Figure 6.4: Experimental resolution characterization. (a) Resolution is measured by determining the minimum separation distance at which two fluorescent beads are resolvable. The experimental resolution can be predicted by calculating inner products of a PSF with shifts and scales of itself; points are considered resolvable when the inner product is below 0.8. The predicted resolution is calculated at 12 different lateral locations in the FoV. The range of values is depicted by the filled area in the plot, and the solid line is the mean. (b) USAF resolution target shows group 6, element 1 containing 7.8 µm bars (boxed) is clearly resolvable, which matches the two-point resolution.

work [8]. Brute-force calibration at every resolvable location in the volume would require Nyquist sampling the two-point resolution, necessitating samples every 4 µm laterally and every 25 µm axially. This is $100 \times 100 \times 4 = 40{,}000$ times more calibration measurements than with our sparse calibration scheme and local convolution model, demonstrating the large savings achieved with our model.

In addition, we show that we can predict the two-point resolution from the PSF measurements, without running a full reconstruction. To do this, we shift (for lateral resolution) or scale (for axial resolution) a central PSF and calculate the inner product with the original. A low inner product indicates that neighboring measurements are sufficiently different to be distinguished. For the noise levels in our system, we find that a normalized inner product of 0.8 is a good predictor of the resolution, plotted by the solid line in Fig. 6.4a (average over 12 field positions). This process can be used for system design by simulating PSFs (for example, using Fresnel propagation) then using this inner product metric to predict the final resolution.

Due to fabrication errors and the non-uniform distribution of lenslets, there will be variation between the focal spots under each microlens, which can result in resolution that

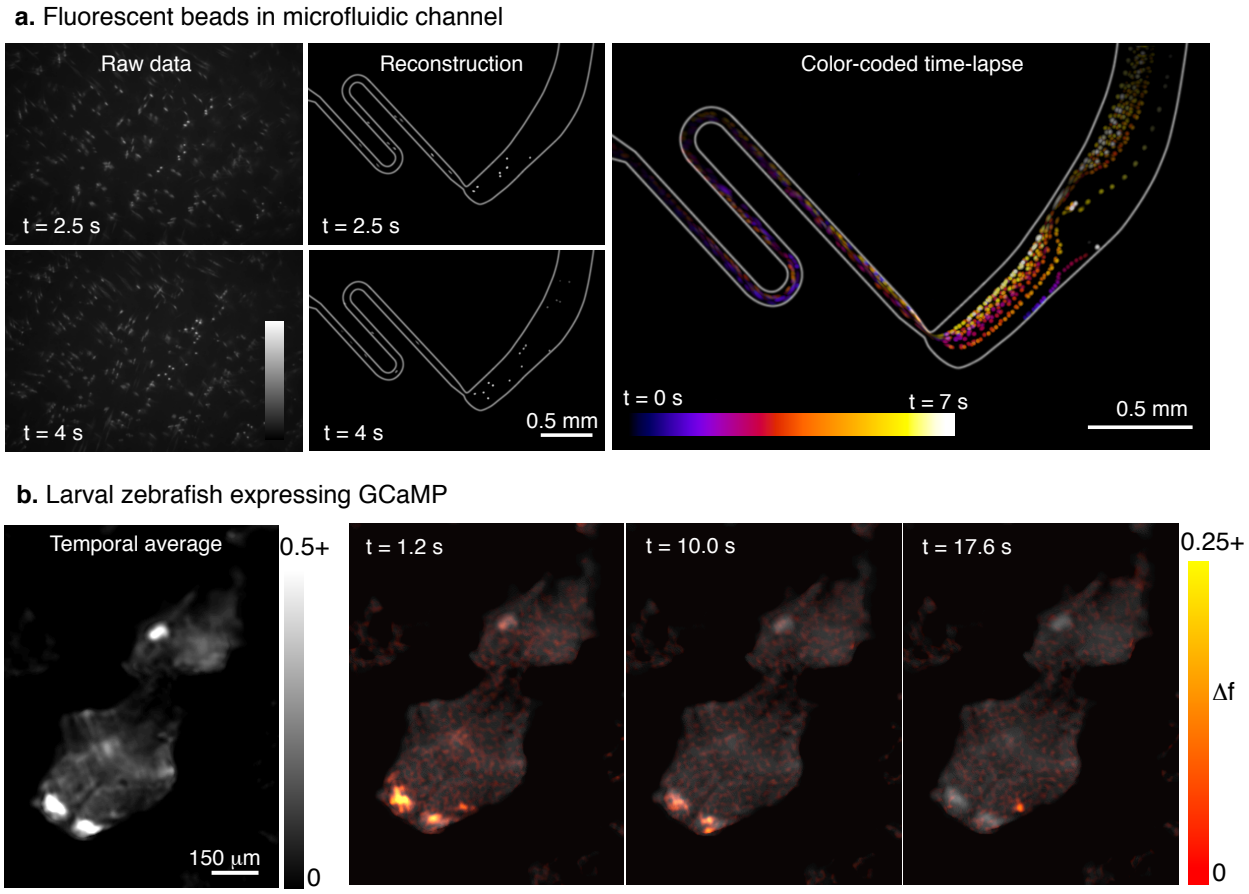

Figure 6.5: Experimental videos captured with our diffuser microscope at 10 fps. (a) Fluorescent beads flowing in a microfluidic channel. Channel outlines are superimposed for visualization purposes, and the full video is in Visualization 1. (b) NeuroD:GCaMP6f larval zebrafish, 6 days old. Change in fluorescence (Δf) compared to a 20th percentile baseline is shown in red and indicates neural activity.

depends on the object's lateral location. However, since each PSF includes focal spots from many lenslets (about 15 in our prototype), the effect of individual variations is averaged. To quantify this in our system, for each depth we calculate the predicted resolution at 12 locations in the FoV and plot the range of predicted values in Fig. 6.4a. We find that the variation in resolving power is low for lateral resolution and modest for axial resolution. We believe resolution variation can be reduced substantially by fabricating the diffuser with more precise methods (e.g. injection moulding).

Since our method is single-shot, the frame rate is only limited by the sensor. To demonstrate, we capture a 10 fps video of 15 µm fluorescent beads flowing through a microfluidic channel, shown in Fig. 6.5a. Beads were reconstructed at a single depth plane, $z = 2.42$ mm, and the full video is available in Visualization 1. We also test our system on a live 6-day-old

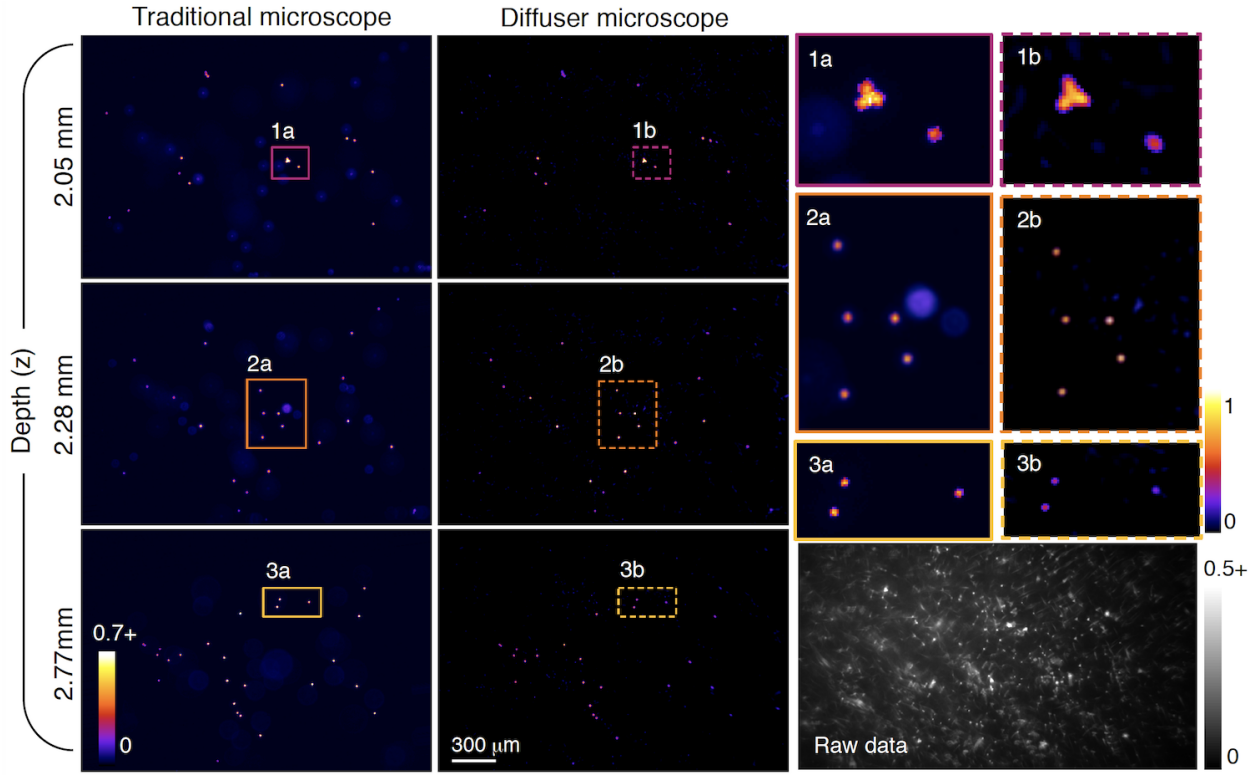

Figure 6.6: 3D reconstruction of 15 µm fluorescent beads, axially separated by coverslips. The focal stack from a traditional fluorescence microscope (5×, 0.15 NA) is shown for comparison, with close-ups on the right. Our diffuser microscope reconstructs all depth planes from a single acquisition (bottom right) and removes out-of-focus light.

NeuroD:GCaMP6f larval zebrafish [128] captured at 10 fps and reconstructed at a single depth plane, $z = 2.19\,\text{mm}$. Fig. 6.5b shows the change in fluorescence (compared to a 20th percentile baseline). Our results qualitatively match the expected neural activity of a larval zebrafish. However, determining whether the reconstructed fluorescence signal is a linear function of the true fluorescence is still an open problem. Compressed sensing theory [24] proves that if the matrix $\mathbf{A}$ fulfills the *restricted isometry property* and the sample is sufficiently sparse, then the signal can be recovered with perfect accuracy. Our design matrix $\mathbf{A}$ is pseudorandom which is expected to fulfill the restricted isometry property with high probability, but the conditions are notoriously hard to verify, and a more rigorous proof of linearity in general cases is the subject of future work.

To highlight the 3D capacity of our system, we created sample containing layers of 15 µm fluorescent beads separated by coverslips. The sample was reconstructed at the three depth planes containing beads, shown in Fig. 6.6. A focal stack from a traditional fluorescent microscope is shown to validate the bead locations. Note that, unlike with a traditional microscope, our prototype reconstructs the complete 3D distribution of beads from a single

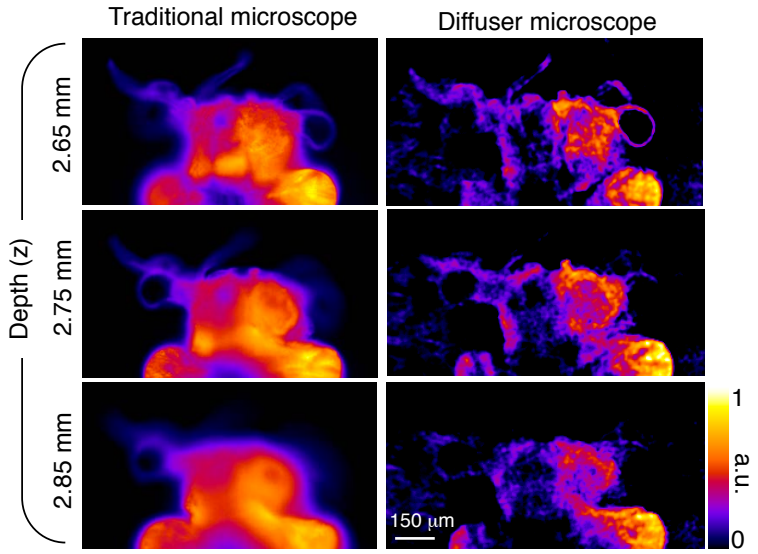

Figure 6.7: 3D reconstruction of a fixed brine shrimp tagged with eosin, shown at three different axial planes. The focal stack from a traditional fluorescence microscope (10×, 0.45 NA) is shown for comparison. Our diffuser microscope reconstructs 3D from a single acquisition and recovers the thin antenna structures at the correct depths.

acquisition of raw data. Finally, since our system requires that the object be sparse for accurate reconstruction, we test on non-sparse samples to demonstrate that our system still captures the edges and sharp regions of dense samples. We image a fixed brine shrimp sample (Carolina Biological) stained with eosin and reconstructed at 10 depth planes spaced 100 µm apart, processed from a single acquisition of raw data (Fig. 6.7). In the dense regions of the head of the brine shrimp, we find some inconsistencies between the traditional microscope focal stack and the diffuser microscope reconstruction. In regions where the sample is sparse, especially the shrimp's antennae, our reconstruction matches the 3D locations captured in the traditional microscope focal stack.

Chapter 7

Scattering Diffuser for Étendue Expansion in Holographic Displays

In this chapter[1], we reverse the ideas of DiffuserCam to create capable of showing imagery with more degrees of freedom than the number of pixels on the display panel. Here, we use a diffuser-like optic in front of the display panel in a holographic display. Instead of the smooth, refractive diffuser of the prior chapters, in this chapter, we choose a *diffractive* diffuser, which has small features, on the size scale of the wavelength of light. To distinguish from the smooth diffuser, we refer to the diffuser in this chapter as a "scattering mask."

As before, we use a physically-based model of our system and solve for the unknown display panel pattern by setting up and solving an optimization problem. However, there are several differences between this work and the imaging work presented earlier. First, calibration is more challenging as there is no sensor inherently in the system. Next, we can't "fix up" an image later, like one can with a camera; what is displayed is what the user sees. Finally, since this work is based on holographic displays, all of our models are based on *coherent* light propagation. Therefore, rather than modeling intensity only, as in prior chapters, we keep track of the complex electric field, only taking the magnitude (intensity) at the end.

In the coming sections, we describe the details of how we overcome these challenges. The result is a simple system in which we use the scattering mask to break the trade-off between field-of-view and eyebox size, a limiting factor in modern holographic displays. Our novel perceptually-inspired constraints enable improved image quality compare to prior work, and we further show a path forward towards practical head-mounted holographic displays with sunglasses like form-factor.

[1]This chapter is based on the published conference paper titled "High resolution étendue expansion for holographic displays" and is joint work with Laura Waller, Ren Ng, and Andrew Maimone [83].

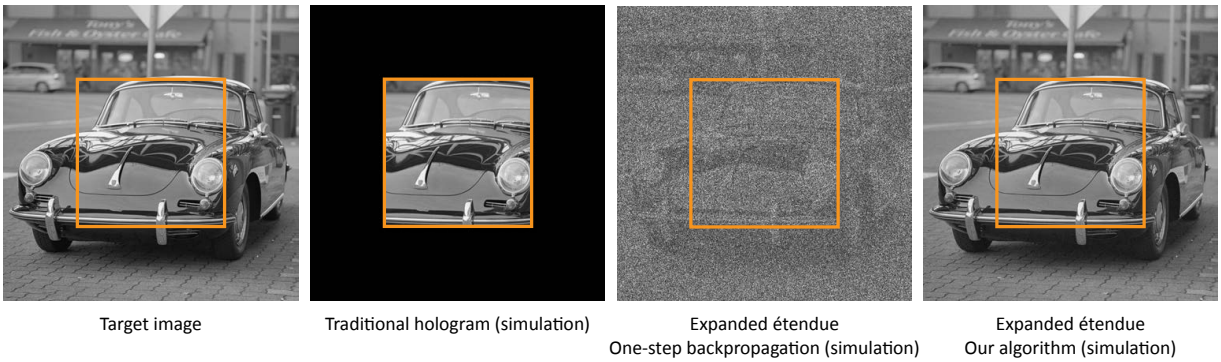

Figure 7.1: Traditional holographic displays have limited étendue resulting in a tradeoff between field-of-view (FoV) and eyebox size. If the eyebox is held constant, unique imagery cannot be displayed outside of the native FoV (orange box). The addition of a thin scattering mask into the system increases the diffraction angles, and thus the FoV, without sacrificing the eyebox. The scattering mask is taken into account during computation of the hologram through an iterative algorithm that outperforms the one-step backpropagation approach used in prior work. Car source image by Bill Newton (CC BY 2.0).

7.1 Holographic Displays

Computer generated holography allows generation of an arbitrary light distribution from a flat, programmable spatial light modulator (SLM) by controlling the wavefront of a coherent beam of light. This technique is particularly promising for near-eye displays since it enables per-pixel focus control, computational correction of optical aberrations, and simple optical components suitable for miniaturization [101].

However, current holographic displays suffer from a unique challenge: a tradeoff between field-of-view (FoV) and the size of the viewing eyebox, the area in which the eye must be located to see the image. Together, these two quantities describe the *étendue* of the display, a quantity which measures the product of the area and solid angle of emitted light from a surface in an optical system. In conventional, non-holographic displays (e.g., the Oculus Rift S) obtaining large étendue is generally not a challenge and can be provided, for example, by a display panel backlight that has large area and range of emission angles. However, in a holographic display the étendue is determined by the number of degrees of freedom (*i.e.* pixels) on the SLM.

For an immersive display, one generally desires a large FoV of $\geq 90°$. With current modulators, a holographic display with such a FoV would only afford an eyebox of approximately 1 mm. Small eyeboxes cause the image to disappear if the eye deviates slightly from the design position, including deviations from eye rotation. The brute force solution is to increase the pixel count of the SLM; however, a solution providing a 10 mm eyebox would require approximately one billion pixels. This solution is two orders of magnitude away from

current technology and is inefficient since the pixel count far exceeds what can be resolved with the human eye.

Another proposed solution is to expand the étendue by augmenting the display with a static scattering mask, which can increase the angles of light diffracted from the display. The known or inferred mask pattern is taken into account when computing the hologram so that a coherent image can still be formed after scattering. However, past efforts using the approach [165, 116, 22] have been limited to very simple scenery, consisting, for example, of only tens of spots. To make this approach practical, it must scale several orders of magnitude to achieve the resolution expected of modern displays.

In this work, we present a new algorithmic approach to scattering-based étendue expansion that preserves the native, high resolution of modern spatial light modulators. After being scattered by a mask, the wavefront from a holographic display has many more degrees-of-freedom than one can control with the spatial light modulator, resulting in very high resolution output, but also extreme noise. Our key innovation is to constrain the holographic image to the number of spatial frequencies that can be controlled by the modulator, so that noise is pushed to higher frequencies than can be resolved by a human viewer. The process essentially decouples étendue and pixel count in holographic displays and results in high quality output with a small to moderate loss in contrast. Unlike prior work, we show that our method scales well to complex, full-resolution, photographic images. We also demonstrate that spatial constraints can be used to programmatically redistribute the image quality and resolution in a holographic image, for example, to increase fidelity in the area around the user's fovea. We present a mathematical framework for optimizing étendue expanded holograms with scattering masks, provide optical simulations and characterize performance, and provide preliminary experimental results on a benchtop prototype. We also discuss current limitations and describe a potential path for implementing our design in a sunglasses-like form factor.

Contributions and Limitations

We provide algorithms for generating high-quality étendue-expanded holograms and evaluate results in simulation and on a hardware prototype. Specifically, we make the following contributions:

- An algorithm for generating holographic images through a scattering mask based on constrained non-convex optimization that significantly outperforms prior state-of-the-art methods
- The addition of frequency and spatial constraints to significantly improve image quality
- The first demonstration, both in simulation and experiment, of dense, photorealistic holograms with higher étendue than the native SLM

Our approach also suffers from some limitations and challenges. Image contrast is reduced as the étendue of the display is increased beyond the native support of the SLM, limiting the

practical range of étendue expansion. Computation time is greater than past methods [22, 165, 116] as we rely on iterative optimization, rather than a one-step method. Additionally, our current hardware prototype operates in a single color channel only, although full color operation has been demonstrated in past related works [116]. As with other prior work featuring scattering masks with small features, our approach is sensitive to alignment and has not yet been demonstrated in a compact form factor suitable for the proposed virtual and augmented reality applications. We address these challenges in Section 7.5.

7.2 Related Work

Holographic Displays Holographic displays have shown promising results for virtual and augmented reality in a series of recent papers [161, 155, 89, 101, 131]. To highlight a few, Maimone, Georgiou, and Kollin [101] demonstrated a holographic display for augmented reality with wide FoV and sunglasses-like form factor, and demonstrated high quality, full-color holograms with real-time computation in a benchtop form factor. Shi et al. [131] demonstrated the rendering of light field data as holograms to capture view-dependent effects. However, these systems were constrained by the low étendue of current SLMs, limiting either the FoV or eyebox of the displays.

More recently, several works have proposed methods for more effective use of the étendue of a holographic display by tracking the viewers' eyes and dynamically moving around a small eye box, also known as pupil steering. Jang et al. [67] show pupil steering by changing the angle of light incident on an SLM with a mechanical mirror and an arrayed hologram. Kim et al. [80] create several copies of the hologram and used a reflective display to control which copy is shown. Choi, Ju, and Park [32] also create copies of the hologram but effectively control which copy is used computationally. While showing promise, these pupil steered methods require precise and low-latency eye tracking, have complex and difficult to miniaturize optics, and have lower performance than non-pupil steered holographic displays. Our approach does not demand eye tracking and requires only the addition of a scattering mask in the optical system.

Focusing through Scattering Media Our approach builds on prior work using wavefront shaping to focus light through an unknown scattering element. This concept was first described by Vellekoop and Mosk [151] who formed a focal spot on the far side of a scattering material by optimizing the phase of a deformable mirror via a feedback loop. Popoff et al. [123] describes a more efficient calibration technique in which the scattering material's transmission matrix is pre-characterized using a wavefront modulator, enabling computational creation of focal spots at any location without re-calibrating; variations on this approach are prevalent in the literature [33, 163, 145]. Focusing through scattering has also been demonstrated with binary amplitude modulation [4, 164] and has been successfully employed for imaging through translucent materials [26, 34]. However, these works all assume that the scatterer is an unknown and undesirable obstacle.

In contrast, our proposed approach exploits properties of the scattering element, namely that it can diffract light to higher angles than natively supported by the spatial light modulator. This property was first used by Vellekoop, Lagendijk, and Mosk [150] to generate sharper foci than achievable without the scattering media, and Yeh et al. [162] used the concept for optical superresolution in imaging. Holographic displays that take advantage of increased angle from a scattering element have also been proposed, as we discuss below.

Étendue Expansion for Displays Perhaps most conceptually similar to our work are prior holographic displays that use a diffractive mask in front of the SLM for the purpose of increasing étendue. Buckley et al. [22] describe using a diffractive phase mask in front of a binary phase modulator to remove the twin image and simultaneously increase the viewing angle of the display. While a compelling idea, their experimental results are limited to a static prototype of a single very simple and sparse scene (a few letters of text) and temporal averaging is required to produce visually-pleasing results.

Yu et al. [165] created the first dynamic display with expanded étendue. Using an off-the-shelf diffuser as the scattering element, they demonstrated a large increase in étendue but could only create up to 15 foci simultaneously and required an intensive calibration that scaled with the number of pixels on the SLM, limiting use to low resolution modulators. Park, Lee, and Park [116] improved on the idea by replacing the unknown diffuser with a known diffractive amplitude mask or "photon sieve", thus eliminating the calibration step and enabling use of higher resolution SLMs. The resulting display could also generate a large increase in étendue, but could only generate up to 75 focal spots simultaneously. In contrast to these approaches, our proposed method scales to dense, photo-realistic holograms at the native resolution of the SLM. We choose a thin transparent mask for our scattering element, which has better light efficiency and a smaller DC term compared to the "photon sieve", and we introduce spatial and frequency-based weighting to étendue expanded holograms.

Algorithms for Computer Generated Holography A key component in any holographic display is the algorithm used to determine the pattern to display on the SLM. This is a particular challenge if the SLM affords phase-only control, which we assume in most of this work. To generate phase-only holograms, one option is to simply discard the amplitude, but iterative approaches, such as the popular algorithm by Gerchberg and Saxton [52], increase image fidelity by allowing the phase at the image plane to vary. Georgiou et al. [51] augmented this algorithm with "don't care" regions which improve image quality at the expense of dedicating a high-noise region outside of the active part of the field of view. However, both these algorithms do not have explicitly defined cost functions, making it challenging to tune parameters to specific applications. In contrast, Zhang et al. [166] explicitly define the problem in an optimization framework, allowing custom, application-specific loss functions, and they demonstrate improved results for optogenetic stimulation. Chakravarthula et al. [27] use a similar framework and target their work towards displays, demonstrating high quality experimental results on color images. Similar to this prior work, our algorithm is based on explicitly solving an optimization problem, but we extend the

approach to compensate for the scattering mask and introduce new loss functions based on perceptual metrics.

7.3 Methods

Étendue of holographic displays

The étendue of a display is defined as the product of the display area with the solid angle of emitted light,

$$G = 4A \sin^2 \theta, \tag{7.1}$$

where G is the étendue of a planar display with area A emitting light confined to a square pyramid of width 2θ around the display's normal. Étendue is conserved through reflections, refraction, and free space propagation [28].

In a non-holographic display (e.g. a backlit LCD panel, OLED panel, etc.), each pixel on the display emits light over a large cone of angles, so the étendue is usually quite large and does not present major limitations to the optical design. However, in a holographic display, the maximum deflection angle of the light, θ, is determined by the pixel size, Δ,

$$\sin \theta = \frac{\lambda}{2\Delta}, \tag{7.2}$$

where λ is the wavelength of light. Substituting this equation into Eq. 7.1 yields the étendue of a holographic display,

$$G = \frac{\lambda^2 A}{\Delta^2} = \lambda N_x \times \lambda N_y, \tag{7.3}$$

where N_x and N_y are the number of pixels along each dimension of the SLM. Therefore, the étendue of a traditional holographic display is proportional to the total number of pixels.

Field-of-View and Eyebox Tradeoff

We will now consider why large étendue is desirable and how much is needed. For simplicity, we'll center the discussion around "Fourier holography" in which a virtual image or volume is produced by the SLM at the Fourier (pupil) plane of a lens, assumed to be ideal with focal length f_1. (The same conclusions holds in the regime of "Fresnel holography" in which the virtual image is created directly in front of the SLM without additional optics.) In a near-eye display, there is typically an additional lens (focal length f_2) that projects the virtual image to optical infinity before the light enters the eye. Figure 7.2a shows a schematic of this scenario.

Consider an SLM with width w and maximum diffraction angle $\pm\theta$. Based on geometry, this results in a one-dimensional (1D) FoV and eyebox size given by:

$$\text{FoV} = 2 \tan^{-1} \left(\tfrac{f_1}{f_2} \tan \theta \right) \approx 2 \tfrac{f_1}{f_2} \theta \tag{7.4}$$

$$\text{eyebox} = \tfrac{f_2}{f_1} w. \tag{7.5}$$

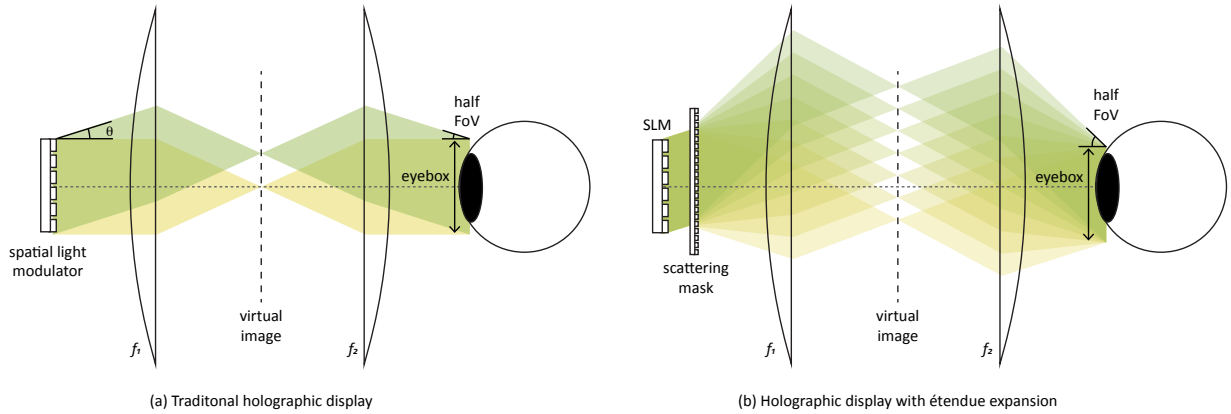

Figure 7.2: In a traditional holographic display (a), the diffraction angle of pixels on the SLM determines the nominal FoV and the extent of the SLM determines the eyebox. These quantities can be exchanged by modifying the ratio of f_1/f_2, but the product is fixed and determined by the number of pixels on the SLM. (b) To overcome this trade-off, a scattering mask is placed in front the SLM. The wavefront coming off the SLM is scattered by the mask to a larger range of angles, thus increasing the FoV without decreasing the eyebox size.

Therefore, we can easily trade off between the FoV and eyebox size for a given SLM by choosing the ratio f_1/f_2. However, the FoV-eyebox product is fixed. To illustrate this, we apply the small angle approximation ($\sin\theta \approx \tan\theta \approx \theta$), and plug Eq. 7.2 into Eq. 7.4 to give the following relationship:

$$\text{FoV} \times \text{eyebox} \approx 2\theta w \approx \frac{\lambda w}{\Delta} = \lambda N_x. \tag{7.6}$$

Thus, the product of 1D FoV and eyebox is equal to the 1D étendue, and the same relationship extends to 2D as well.

To give an idea of the achievable design space, the highest pixel count commercial SLM today (Holoeye GAEA-2) has 4160×2464 pixels. To provide an immersive experience for the user with a horizontal FoV of 120° at wavelength $\lambda = 532\,\text{nm}$, the eyebox size is only 1.05 mm. Therefore, even a small rotation of the eye will cause the pupil to leave the display's viewing eyebox, and the user will no longer be able to see the image. If we instead prioritize an eyebox of 10 mm, large enough to accommodate reasonable eye movement, the resulting FOV is only 12.7°. Simultaneously achieving both the 120° FoV and 10 mm eyebox would require $N_x \approx 32,500$ pixels, resulting in a total of over one billion pixels needed over the SLM area.

The brute force solution is to develop SLMs with more pixels, a pathway that is actively being researched. Most current phase SLMs use liquid crystal on silicon (LCoS) technology, and the highest resolution LCoS device known has 8192×4320 pixels [139]. However, this still leaves a gap of more than an order of magnitude between the achievable étendue of

current technology and that required for the ideal display described above. Continuing to increase pixel count will encounter challenges in display bandwidth, power usage, pixel cross-talk [107], and device size.

In addition, a billion-pixel SLM is inefficient since it generates much higher resolution images than can be resolved with the human eye. An SLM with a horizontal pixel count of $N_x = 32,500$ and 120° FoV, as described above, would create visual stimuli with 270 pixels/degree angular resolution, far beyond the 60 pixels/degree limit of normal 6/6 or 20/20 vision. Furthermore, we know that humans have higher visual acuity in the center of the retina, a observation that has been exploited for improved computational efficiency through foveated rendering [72, 117, 56]. As a result of both foveation and the limited resolution of the eye, the total number of degrees of freedom actually perceived by the user is far smaller than the number of SLM pixels needed for étendue purposes. What we truly desire is to decouple the étendue, and therefore the FoV and eyebox, from the number of SLM pixels and the display resolution. In the next section, we describe a strategy to achieve this by placing a static scattering mask in front of the SLM and computationally generating holograms that account for the limited resolution of human vision.

Scattering-Based Étendue Expansion

In a traditional holographic display, shown in Figure 7.2a, the image perceived by the viewer is the same as the intensity distribution at the virtual image plane, $I(\vec{x})$, which is described by

$$I(\vec{x}) = |y(\vec{x})|^2 = |\mathcal{F}\{s(\vec{u})\}|^2. \tag{7.7}$$

Here, $s(\cdot)$ is the complex field at the SLM, $y(\cdot)$ is the complex field at the image plane, $\mathcal{F}\{\cdot\}$ denotes 2D Fourier transform, and $\vec{x}$ and $\vec{u}$ are the coordinates at the image plane and SLM plane, respectively, which are related by $\vec{x} = \vec{u}/\lambda f_1$ [54]. Since the SLM pixel size, Δ, determines the maximum frequency displayable on the SLM, the virtual image has finite controllable extent corresponding to $\vec{x} \in [-\frac{1}{2\Delta}, \frac{1}{2\Delta}]^2$. Due to the discrete nature of the SLM pixels, the SLM also generates higher-order terms which manifest as replicas of diminishing intensity. However, these replicas cannot be controlled independently of the central region so they are not included in the FoV calculation; in fact, many holographic display systems physically filter them out.

To expand the étendue of the holographic display, we place a static scattering mask in front of the SLM, shown in Figure 7.2b. Unlike past systems that focused light through unknown media (e.g. biological tissue), here we can choose the scattering mask to have desirable properties, as discussed in Sec. 7.3. We specifically use a thin transparent mask with a known phase profile, $\alpha(\vec{u})$, resulting in a corresponding complex modulation function $m(\vec{u}) = \exp(j\alpha(\vec{u}))$. Since the mask is thin by design, with an optical path length deviation of at most one wavelength, we assume that the mask only affects the electric field at one plane, and using a relay system we set that plane to be directly conjugate to the SLM. Therefore, with the addition of the scattering mask, the intensity at the image plane is

$$I(\vec{x}) = |\mathcal{F}\{s(\vec{u})m(\vec{u})\}|^2, \tag{7.8}$$

which can also be written as a convolution between the far-field patterns of the SLM and the mask:

$$I(\vec{x}) = |\mathcal{F}\{s(\vec{u})\} * \mathcal{F}\{m(\vec{u})\}|^2. \tag{7.9}$$

Here, $*$ denotes a 2D convolution.

Assuming the mask is defined on a discrete grid with pixel size Δ_m, the far field pattern of the mask, $\mathcal{F}\{m(\vec{u})\}$, has unique content over the extent $\vec{x} \in [-\frac{1}{2\Delta_m}, \frac{1}{2\Delta_m}]^2$. As with the SLM, higher-order terms due to the discrete nature of the mask create replicas of the central region, allowing us to think of the convolution in Eq. 7.9 as having circular boundary conditions. Therefore, the total extent of $I(\vec{x})$, which directly corresponds to the FoV, is equal to the extent of the far-field pattern of the mask itself. Using standard micro/nano fabrication techniques, it is straightforward to create a mask with smaller pixels than those on the SLM, thus increasing the display's FoV.

Importantly, adding the scattering mask into the system in this way does not change the total SLM size, and therefore the eyebox is unchanged. This means that the increase in FoV described above corresponds directly to an increase in étendue by a factor, q, determined by the ratio of the SLM and mask pixel sizes,

$$q = \left(\frac{\Delta}{\Delta_m}\right)^2. \tag{7.10}$$

Throughout this paper, we'll visualize the increase in étendue as a FoV expansion, but the total expanded étendue can easily be redistributed between FoV and eyebox by choosing appropriate focal length lenses, as described in Sec. 7.3.

Mask Design

One important parameter of the mask design is the pixel size, Δ_m, which determines the étendue expansion factor (Eq. 7.10). In addition, if we assume no prior knowledge about the content to be displayed, Eq. 7.9 suggests that we want a uniform far-field mask pattern, $\mathcal{F}\{m(\vec{u})\}$, so that the intensity distribution, $I(\vec{x})$, is not biased towards any specific location in the FoV. To achieve this, we choose a random binary phase profile, with phase $\alpha(\vec{u})$ either 0 or π (equivalent to $m(\vec{u}) = \pm 1$) at each mask pixel. This design is a type of white noise and thus yields a flat far-field pattern. Although we chose a binary pattern for ease of fabrication, other mask designs with uniform far-field patterns will generate similar results. However, an amplitude-only mask, such as the "photon sieve" used by Park, Lee, and Park [116], will always have a strong zero-order (DC) term due to lack of "negative" values in the modulation function, which results in a uneven far-field distribution. The DC term diminishes if more of the mask is opaque, but this greatly reduces light efficiency compared to our transparent phase mask.

A practical factor in the mask design is our ability to computationally model the effect of the mask on the electric field. Theoretically, a multiple-scattering element with uniform far-field intensity could replace the thin mask in our system. However, calibration of multiple-scattering effects is intensive and, for fixed q, the number of parameters in the model scales

quadratically with the number of pixels on the SLM [123]. In contrast, our choice of a thin flat scattering mask enables efficient modeling by a single point-wise multiplication (Eq. 7.8), and the number of parameters describing the mask scales linearly with the number of SLM pixels for fixed q, allowing the model to be efficiently used in an iterative computational framework.

Our final practical consideration in the mask design is also related to computational tractability: to compute the Fourier transform of Eq. 7.8 with an FFT, it is necessary that both the SLM and mask be represented digitally on a uniform grid. Therefore, a mask with a pixel structure is convenient as it is easily represented in a discrete form. A non-pixelated mask with smoothly varying features could also be represented on a uniform grid but to accurately capture all features, the mask must be over-sampled, increasing compute time.

Although our mask choice is justified based on the reasoning above, improvements to the design may be revealed through end-to-end optimization [136], which is a topic for future work.

Image Calculation Algorithm

Simply adding our thin scattering mask in front of an SLM is not sufficient to make a display. In fact, the mask would scramble the wavefront such that the viewer only sees a speckle field. Therefore, to generate an image after the mask, we must pre-compensate for the scattering in the SLM pattern. Furthermore, the number of degrees-of-freedom in the output image, $I(\vec{x})$ is higher than the number of controllable modes on the SLM by a factor of q, so it is impossible to generate arbitrary images. This can be thought of as an over-determined data fitting problem, in which we can only attempt to create the closest possible image (by some metric), but there will always be uncontrollable noise creating deviations between the target image and output image. We propose an optimization-based algorithm to reduce unwanted noise by allowing the algorithm to control phase at the image plane, and we further direct residual noise into perceptually less noticeable regions or spatial frequencies through custom loss functions based on human vision. We begin by describing the algorithm used in prior work, which we call one-step backpropagation, then introduce our improved version.

One-step Backpropagation

Despite physical differences in their systems, prior state-of-the-art [123, 22, 165, 116] all use the same underlying algorithm for computing the SLM pattern to generate target images through the scattering element.

Due to the reciprocity of light, a given target electric field, $\hat{y}(\vec{x})$, can be propagated backwards to the calculate the electric field at the SLM plane. For example, in our model (Eq. 7.8), the electric field at the SLM plane is

$$\hat{s}(\vec{u}) = \mathcal{F}^{-1}\{\hat{y}(\vec{x})\}m^*(\vec{u}), \tag{7.11}$$

where $m^*(\vec{u})$ is the complex conjugate of the mask function and $\mathcal{F}^{-1}\{\cdot\}$ is the inverse 2D Fourier transform. If this electric field were displayed on the SLM, it would exactly recreate

the target electric field after the mask. However, this is impossible for two reasons. First, the SLM is phase-only, meaning that it can only display the phase of the complex field. Second, the resolution of the SLM pixels is limited and only one phase value can be displayed per pixel. To generate a valid SLM pattern, the one-step backpropagation method integrates the field over each SLM pixel, then throws away the amplitude to get a phase-only value:

$$p_i = \arg\left(\int_{\Delta_i} \mathcal{F}^{-1}\left\{\hat{y}(\vec{x})\right\} m^*(\vec{u})\, d\vec{u}\right), \tag{7.12}$$

where p_i is the phase value at pixel i on the SLM, the $\arg(\cdot)$ operator takes the phase of the complex field, and the integral is over the area corresponding to the i-th pixel.

On the surface this appears to be an optimal approach without much room for improvement. However, this model optimizes for a target *electric field*, despite the fact that humans can only detect the intensity of light. To create a target intensity, $\hat{I}(\vec{x})$, with this approach, first an arbitrary phase $\phi(\vec{x})$ is assigned such that

$$\hat{y}(\vec{x}) = \sqrt{\hat{I}(\vec{x})}\, \exp\left(j\phi(\vec{x})\right), \tag{7.13}$$

and after the phase is assigned, it stays fixed when calculating the SLM pattern from Eq. 7.12. In contrast, in our approach we let the phase at the image plane be a free variable, since it is not detectable by the eye, which greatly improves image quality at the expense of increased computation. This idea has been previously applied to computer generated holography without scattering masks [52, 166, 27]. In addition, we introduce a flexible framework that allows us to incorporate different loss functions that can be tailored to the specific application of near-eye displays.

Our Approach

To calculate the SLM pattern, we solve the following optimization problem to find the phase values $\vec{p}$ to be displayed on the SLM.

$$\begin{aligned} \vec{p} = \arg\left(\underset{\vec{s}}{\operatorname{argmin}}\, \mathcal{L}\left(I(\vec{x}), \hat{I}(\vec{x})\right)\right) \\ \text{subject to } \ |\vec{s}| = \vec{1}, \end{aligned} \tag{7.14}$$

where $\vec{s}$ is the discrete complex field at the SLM pixels, the $\arg(\cdot)$ operator takes the phase of the complex field, $|\cdot|$ denotes element-wise magnitude, $\hat{I}(\vec{x})$ is the target intensity at the image plane, and $I(\vec{x})$ is the output intensity calculated using Eq. 7.8. Finally, $\mathcal{L}(\cdot,\cdot)$ is a differentiable custom loss function that outputs a single real-valued similarity metric between the output and target intensities.

To take the discrete nature of the SLM into account, we solve directly for a single phase value at each pixel. Before computing the output intensity using Eq. 7.8, we generate a

continuous representation from the discrete field, $\vec{s}$, as follows

$$\begin{aligned} s(\vec{u}) &= \sum_i s_i \, R(\vec{u} - \vec{u}_i) \\ R(\vec{u}) &= \begin{cases} 1 & \vec{u} \in [-\Delta/2, \Delta/2]^2 \\ 0 & \text{otherwise} \end{cases} \end{aligned} \tag{7.15}$$

where s_i is the complex field at the i-th pixel and $\vec{u}_i$ is the pixel's location on the SLM. Note that, although the equations here are presented as continuous, they must be discretized to solve Eq 7.14 digitally; for practical purposes Eq. 7.15 corresponds to upsampling the SLM pattern with a box filter. Since the mask has smaller pixels than the SLM, the necessary grid resolution to describe the system is always finer than the SLM pixel size, making this upsampling a critical step. See the Appendix A.4 for more details.

In addition to specifically solving for a phase-only discrete SLM pattern, we also improve on one-step backpropagation with our custom loss function, which acts only on the image intensities. Unlike the prior algorithm, the phase of the target image is never defined, and the phase of the output image can vary freely. If we simply want the output image to match the target as closely as possible, we can use the following sum of squares loss function:

$$\mathcal{L}\left(I(\vec{x}), \hat{I}(\vec{x})\right) = \tfrac{1}{2} \sum_{\vec{x}} \left(I(\vec{x}) - \hat{I}(\vec{x})\right)^2. \tag{7.16}$$

However, since the number of degrees-of-freedom on the SLM is significantly smaller than the number of free variables in the output image, we cannot perfectly match the target and output. Next, we introduce two different loss functions that prioritize features of the image that are more perceptually relevant. Other loss functions (for example, based on salient features in the image) could be readily incorporated in this optimization framework.

Frequency Constraints The resolution of the human eye is limited to about 60 pixels/degree for normal vision. As described in Section 7.3, the achievable resolution of a large étendue holographic display can be several times higher than human perception. Therefore, we can improve the output by constraining the loss function to only penalize the lower spatial frequencies resolvable by the eye:

$$\mathcal{L}\left(I(\vec{x}), \hat{I}(\vec{x})\right) = \tfrac{1}{2} \sum_{\vec{u}} \left(c_f(\vec{u}) \, \mathcal{F}\left\{I(\vec{x}) - \hat{I}(\vec{x})\right\}\right)^2, \tag{7.17}$$

where $c_f(\vec{u})$ is a low-pass filter. Setting the cutoff frequency of the low-pass filter to match human vision depends on the total FoV of the system (determined by the focal lengths of the relay optics, described in Sec. 7.3). To abstract away the choice of how to distribute eyebox and FoV, we set the cutoff frequency to correspond to the resolution achievable by the native SLM. Conveniently, this means that the resolution of the display remains constant as the étendue expansion factor increases, as long as the native SLM resolution is at least

that of the human eye. To minimize ringing artifacts, we choose a fifth order Butterworth filter

$$c_f(\vec{u}) = \left(1 + \left(\frac{||\vec{u}||^2}{r_0^2}\right)^5\right)^{-1} \tag{7.18}$$

where $||\cdot||^2$ denotes the squared magnitude, and the cutoff frequency is $r_0 = \Delta_m\sqrt{N_x \times N_y/\pi}$ such that the number of controlled frequencies in the filter's passband matches the number of degrees-of-freedom on the SLM.

Spatial Constraints It is well known that humans have foveated vision with the most visual acuity at the center of our gaze direction. This can be used to improve image quality by non-uniformly weighting the image loss based on the mostly likely regions viewed by the user, the most important content in the image, or the center of gaze as determined from eye tracking data. We combine this idea with the frequency constraints described above through the following loss function

$$\mathcal{L}\left(I(\vec{x}), \hat{I}(\vec{x})\right) = \tfrac{1}{2}\sum_{\vec{x}}\left(c_s(\vec{x})\mathcal{F}^{-1}\left\{c_f(\vec{u}')\,\mathcal{F}\left\{I(\vec{x}) - \hat{I}(\vec{x})\right\}\right\}\right)^2, \tag{7.19}$$

where $c_s(\vec{x})$ is a grayscale weight map describing the spatial importance of different regions of the image. This is conceptually similar to the work of Georgiou et al. [51], in which noise is moved into "don't care" regions outside of the active FoV. Here we use spatial constraints to control the importance of imagery seen by the user and also provide non-binary weighting.

Extension to Multiple Focal Planes

One of the key advantages of holographic displays is the ability to display 3D content, which improves the realism of the display while helping alleviate visual fatigue from the vergence-accommodation conflict that plagues stereoscopic displays. In this section, we extend our image calculation algorithm to create content at multiple focal planes simultaneously. The intensity at distance z from the focal plane of the lens is

$$I_z(\vec{x}) = \left|\mathcal{P}_z\mathcal{F}\left\{s(\vec{u})m(\vec{u})\right\}\right|^2, \tag{7.20}$$

where $\mathcal{P}_z$ is the Frensel propagation operator which can be defined in Fourier space as $\mathcal{P}_z\{\cdot\} = \mathcal{F}^{-1}\{h_z(\vec{u})\mathcal{F}\{\cdot\}\}$ where

$$h_z(\vec{u}) = \exp\left(2\pi j z/\lambda\right)\exp\left(j\pi\lambda z\|\vec{u}\|^2\right), \tag{7.21}$$

and $\|\cdot\|^2$ is magnitude squared. By noting that $h_z(\vec{u}) = h_z(-\vec{u})$, we can efficiently calculate $I_z(\vec{x})$ as follows

$$I_z(\vec{x}) = \left|\mathcal{F}\left\{s(\vec{u})m(\vec{u})h_z(\vec{u})\right\}\right|^2. \tag{7.22}$$

As done by Zhang et al. [166], we simultaneously optimize the intensity at all focal planes of interest by solving

$$\vec{p} = \arg\left(\underset{\vec{s}}{\operatorname{argmin}} \sum_{z} \mathcal{L}\left(I_z(\vec{x}), \hat{I}_z(\vec{x})\right)\right) \quad \text{subject to } |\vec{s}| = \vec{1}, \tag{7.23}$$

where $\hat{I}_z(\vec{x})$ is the target intensity at the focal plane at distance z, the summation is over the discrete number of z planes of interest, and the loss function can be set to any of those described for a single plane. Extensions to loss functions that account for interactions between planes are also possible but not explored in this work. As with the single plane version, $\vec{s}$ is converted to $s(\vec{u})$ via Eq. 7.15.

7.4 Results

Simulation

We test our étendue expansion concept in a simulation implemented in MATLAB running on an Nvidia GeForce GTX 1060 GPU. We assume an SLM with 16 µm pixels and resolution 960×540. Our scattering mask is modeled as a thin phase element with binary phase (either 0 or π) randomly assigned to each pixel. The mask is the same physical size as the SLM and the pixel size of the mask, Δ_m, determines the étendue expansion factor, q, based on Eq. 7.10 with $\Delta_m = 8\,\mu\text{m}$, $4\,\mu\text{m}$, or $2.66\,\mu\text{m}$ for $4\times$, $16\times$, and $36\times$ étendue expansion, respectively. For a given target image, we solve Eq. 7.14 using projected gradient descent with Nesterov acceleration [12]; the details of the algorithm are summarized in the Appendix A.4. We use Eq. 7.8 to simulate the intensity at the image plane from the SLM pattern, then apply the low-pass filter described in Eq. 7.18 to simulate the perceptual effect of limited retinal resolution.

Figure 7.3 shows a simulation comparison of the image formation algorithms presented in Section 7.3. We implemented the one-step backpropagation algorithm used in prior work (Sec. 7.3) by solving Eq. 7.12 after generating a target electric field based on Eq. 7.13 where the phase, $\phi(\vec{x})$, is random from a uniform distribution, which we found yielded superior results compared to a constant phase. By not allowing optimization of the phase or additional constraints, one-step backpropagation results in low contrast images, even for only a $4\times$ expansion factor, and contrast degrades rapidly as the expansion factor increases. We conclude that one-step backpropagation is better suited for the ultra-sparse scenes demonstrated in prior work [165, 116] than for dense high resolution images.

We improve on prior state-of-the-art with our iterative optimization approach in which phase at the image plane is a free variable, and even without additional perceptual constraints, the improvement from such optimization is clearly apparent at all expansion factors. Adding frequency constraints, such that only low frequencies are optimized, further improves performance, yielding high quality, good contrast images for $4\times$ expansion. However, these results begin to lose more contrast at higher expansion factors. Our spatial constraints

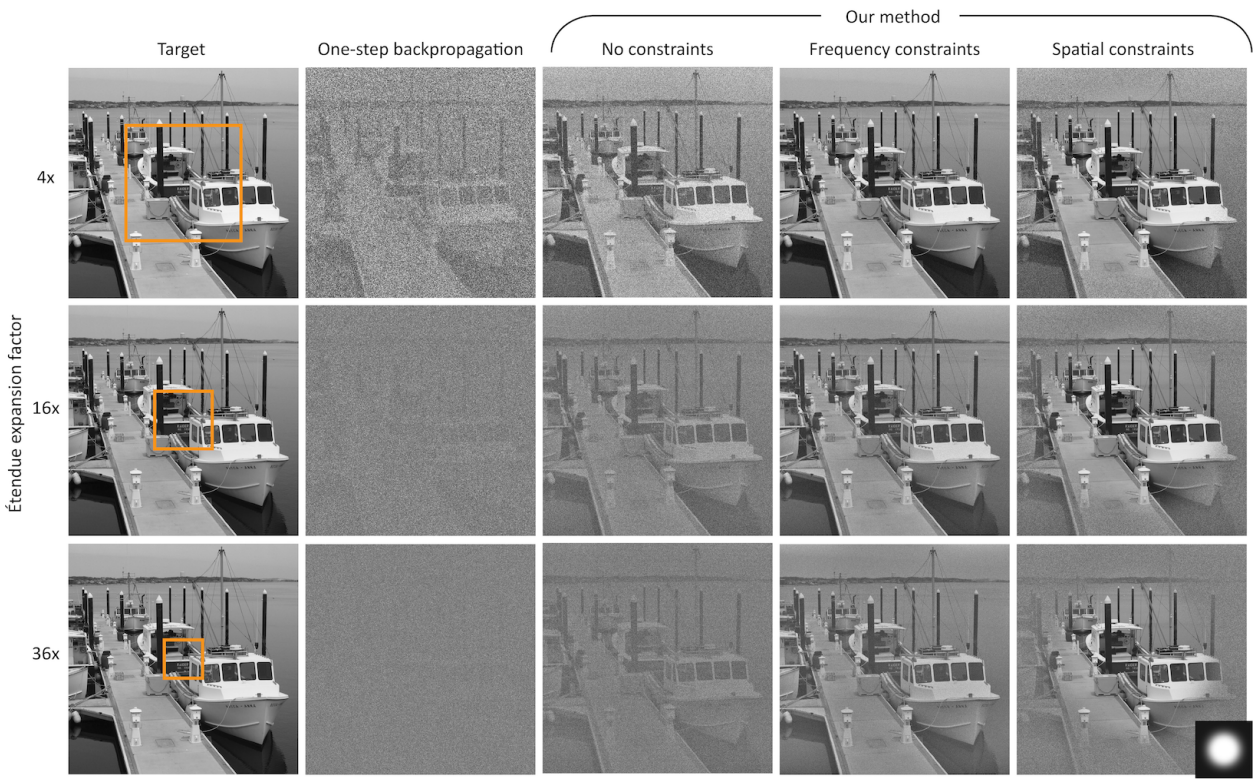

Figure 7.3: Simulations comparing image formation algorithms for 4×, 16×, and 36× étendue expansion. The orange box indicates the FoV addressable by the native SLM without the scattering mask. With one-step backpropagation, the phase at the image plane is fixed and no perceptual constraints can be added, resulting in low contrast results that scale poorly to higher expansion factors. By allowing phase at the image plane be a free variable (i.e. "no constraints"), image quality is improved but still shows noisy results. With the addition of frequency constraints that prioritize the range of frequencies detectable by the visual system, noise and contrast are further improved. However, contrast degrades as the expansion factor increases. This can be mitigated by applying additional spatial constraints via a spatial weighting map (lower right inset). The spatial map used here is designed to approximately correspond to those used by foveated renderers [117], assuming our total FoV is set to 80°. Adding the spatial weights improves contrast and noise performance in the prioritized central region at the expense of the periphery, and the region of interest can easily be moved based on the viewer's gaze direction (see supplemental video for an example). All images are low-pass filtered to simulate the limited resolution of the visual system. Quantitative metrics are found in Table 7.1. Boat source image by Erick Bee (CC BY-SA 2.0).

Table 7.1: Numerical comparison of image formation algorithms, averaged over 40 different natural images, where higher values indicate more similarity to the target. To quantify the effects of the spatial constraints, metrics were calculated over both the whole FoV (top) and over a cropped region one quarter the size of the FoV corresponding to the area prioritized by the spatial constraints (bottom). All results and target images are filtered according to Eq. 7.18 before calculating metrics, and images are normalized to have the same mean. The best performing algorithm in each category is in bold; when considering the whole FoV, our method with frequency constraints performs best, but can be improved in a subregion of the FoV by applying spatial constraints.

4× expansion

		PSNR (dB)	SSIM
Full FoV	One-step backpropagation	12.059	0.064
	Ours: No constraints	18.510	0.286
	Ours: Frequency constraints	**26.958**	**0.624**
	Ours: Spatial constraints	19.267	0.472
Partial FoV	One-step backpropagation	12.163	0.069
	Ours: No constraints	18.996	0.316
	Ours: Frequency constraints	27.412	0.655
	Ours: Spatial constraints	**36.811**	**0.903**

16× expansion

		PSNR (dB)	SSIM
Full FoV	One-step backpropagation	12.720	0.197
	Ours: No constraints	15.128	0.363
	Ours: Frequency constraints	**18.486**	**0.552**
	Ours: Spatial constraints	15.341	0.455
Partial FoV	One-step backpropagation	13.044	0.194
	Ours: No constraints	15.883	0.373
	Ours: Frequency constraints	19.773	0.572
	Ours: Spatial constraints	**27.266**	**0.829**

36× expansion

		PSNR (dB)	SSIM
Full FoV	One-step backpropagation	12.891	0.389
	Ours: No constraints	14.217	0.509
	Ours: Frequency constraints	**16.167**	**0.633**
	Ours: Spatial constraints	14.371	0.541
Partial FoV	One-step backpropagation	13.272	0.383
	Ours: No constraints	14.912	0.515
	Ours: Frequency constraints	17.344	0.650
	Ours: Spatial constraints	**21.500**	**0.806**

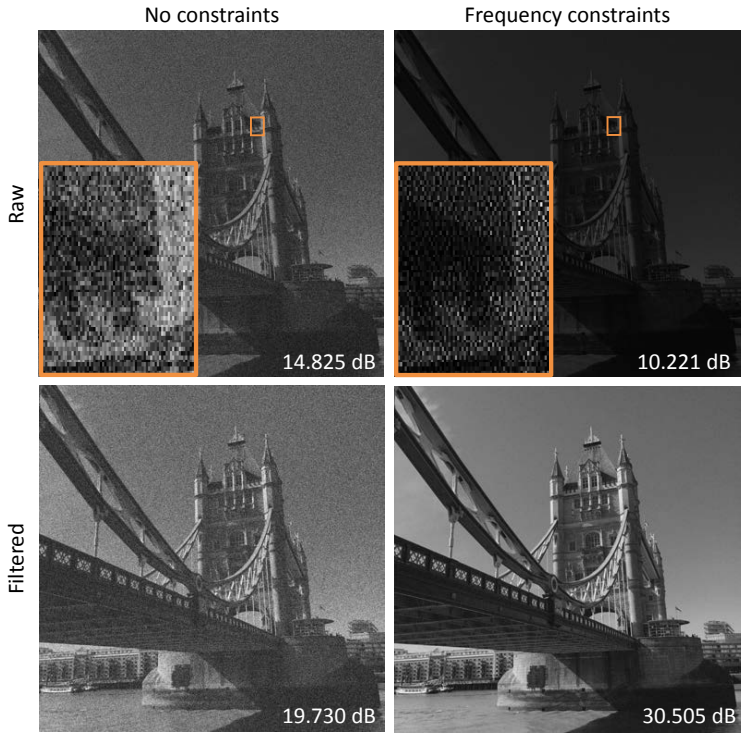

Figure 7.4: By applying frequency constraints during optimization of the SLM pattern, noise is moved into higher frequencies that are imperceptible to the viewer, except through contrast loss. The unconstrained version is more similar to the target image before filtering; after low-pass filtering the output to simulate the viewer's experience (bottom), the frequency constrained result is much closer to the target. This example has 4× étendue expansion and PSNR is reported in the bottom right of each image.

can restore some of the contrast by prioritizing image fidelity in a subset of the FoV, thus improving contrast in this region at the expense of image quality in the periphery. Importantly, moving the spatial constraints to different regions of the FoV is easily accomplished algorithmically by changing the spatial weights, $c_s(\vec{x})$. The spatial constraints work equally well over any position in the FoV, and shifting position does not require physically moving components. An example of dynamically changing the spatial constraints is shown in the supplemental video. As discussed in Section 7.5, the spatial weights can be used statically or dynamically in conjunction with eye tracking.

A quantitative comparison of the methods is shown in Table 7.1. We calculate two metrics, peak signal to noise ratio (PSNR) and structural similarity (SSIM) [62]. When simulating images with spatial constraints, we apply a spatial map with a central region of value 1 that smoothly transitions into a peripheral region with value 0.1, shown in the inset of Figure 7.3. These regions approximately correspond to the viewing zones used by Patney et al. [117] for foveated rendering if the FoV of our display is set to 80° (although

there is no direct mapping between the values in our spatial map and the sampling factors used by Patney et al. [117]). To fairly compare the images with spatial constraints, metrics are calculated over both the whole FoV and over a subregion (the central quarter of the image) corresponding to the area prioritized by the spatial constraints. Quantitatively, our method with frequency constraints performs best when considering the whole FoV, and our method with spatial constraints performs best when considering only a subregion. As visible in Figure 7.3, the main difference between the target and output images is due to contrast reduction.

The frequency constraints are critical to achieving low-noise images. In Figure 7.4 we compare the unconstrained loss function (Eq. 7.16) and the loss function with frequency constraints (Eq. 7.17) *before* applying the low-pass filter that approximates the visual system. Without constraints, the raw unfiltered output is more faithful to the target image, as evidenced by its higher PSNR. With frequency constraints, the unconstrained noise is moved into higher frequencies that are filtered out. The filtered result is visually less noisy and higher contrast. In this example, the frequency constraints improve PSNR by over 10 dB in the filtered result.

Experimental Prototype

We validate our simulations with a benchtop prototype built with a 1080p phase only LCoS SLM (Holoeye PLUTO-2). To reduce sensitivity to alignement errors, we bin the SLM pixels 2×2 resulting in an effective resolution of 960×540 with 16 µm pixels. For our scattering mask, we use a binary phase mask with a random pattern of 4 µm pixels with phase values of either 0 or π, resulting in an étendue expansion of $q = 16\times$. The mask was fabricated with lithography, and we assume the pattern is known. One of the challenges of LCoS SLMs is the relatively low diffraction efficiency that results in a strong reflection of un-modulated light, called the DC term. When the DC term passes through the mask, it scatters and creates background haze that reduces contrast. Therefore, we add a $4f$ system consisting of two Pentax FA 645 lenses (75 mm focal length, F/2.8) and place an opaque DC block (chrome on glass mask) at the Fourier plane to remove the DC term. This also allows us to relay the SLM directly onto the mask, to match the model in Eq. 7.8. The SLM is illuminated by a collimated beam from a laser with $\lambda = 660$ nm. We pre-calibrate the SLM phase [13] and measure and compensate for flatness deviations on the SLM [159].

After the mask, a relay system de-magnifies the image by a factor of $2\times$; finally, images of our display are captured with a monochromatic camera (FLIR Blackfly S BFS-U3-200S6M) with a f = 16 mm, F/1.4 C-mount lens. As with the simulations, captured experimental images are low-pass filtered to simulate the effect of low-pass filtering in the visual system. All non-linear processing in the camera, such as gamma and black level, are turned off and there are no adjustments to the black level in post processing. Figure 7.5 shows a schematic of the experimental prototype.

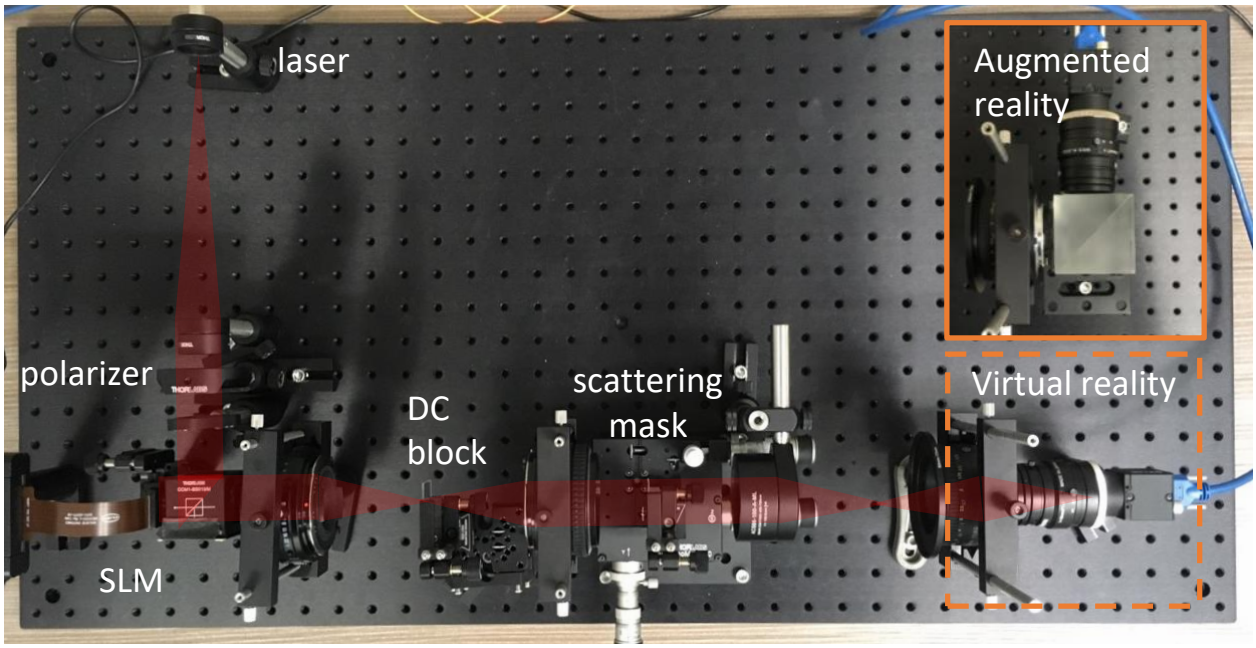

Figure 7.5: Our benchtop prototype can be arranged either in a virtual reality configuration, shown above, or in an augmented reality configuration by including a beamsplitter in the imaging path, shown in inset.

Mask Alignment and Calibration

The alignment of the scattering mask is critical for good performance since the SLM pattern is only valid when the mask position matches the simulation. Our custom designed mask includes coarse alignment markers (three 600 µm squares of constant phase) that are visible to the human eye and are used to approximately position the mask in the system. Since the mask pattern in known *a priori*, we can use our algorithm to calculate an SLM pattern that generates a single focal spot on the camera. The mask position, which is controlled by a 6-axis motion stage (Thorlabs Max313D, APY002, KM100C), is fine-tuned over the six degrees of rigid transformation to maximize the spot intensity on the camera.

In addition, geometric distortion from the $4f$ system can cause non-rigid misalignment between the mask and SLM. We coarsely compensate for this effect with the following procedure. First, we split the mask area in 3×5 subsections. For each subsection, we computationally modify our simulated mask pattern, $m(\vec{u})$, by translating the subsection of interest. We then calculate a new SLM pattern to produce a focal spot based on the modified mask. Using the camera for feedback, we optimize each subsection's translation to maximize the spot intensity, and we combine the optimal translations in a piecewise fashion to form a new mask pattern which is used for all future images. Note that we only apply translations of integer pixel values since we find interpolation of the mask pattern results in artifacts in the displayed image.

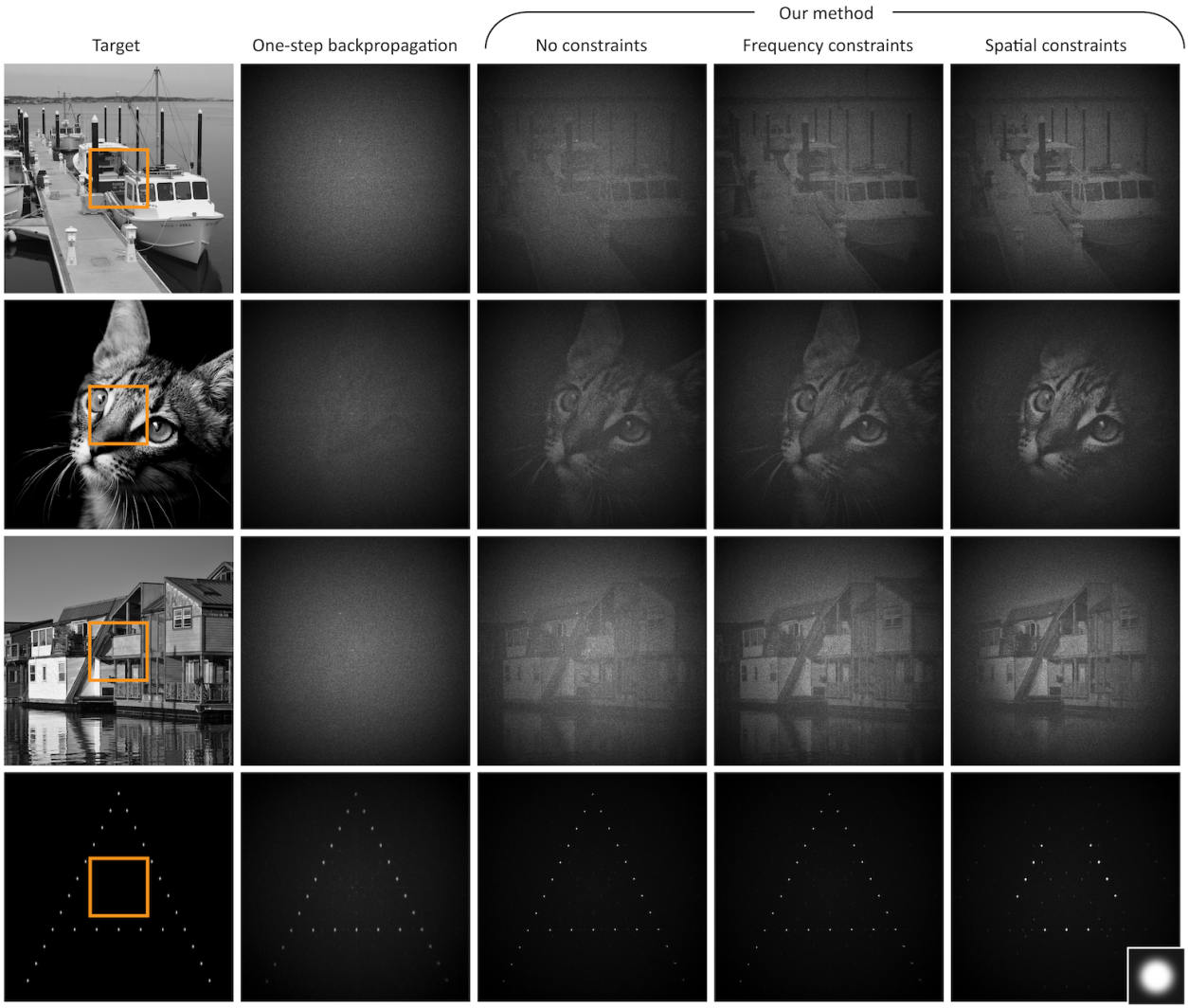

Figure 7.6: Experimental results from our benchtop prototype with 16× étendue enhancement beyond the native SLM. Although the one-step backpropagation algorithm is effective on sparse scenes (bottom row), it has very low contrast on dense imagery. Contrast is improved by our iterative method and further enhanced by applying frequency and spatial constraints in the loss function. The orange box shows the native FoV of the SLM. Boat source image by Erick Bee (CC BY-SA 2.0); cat source image by Lali Masriera (CC BY 2.0); floating houses source image by Madeleine Deaton (CC BY 2.0).

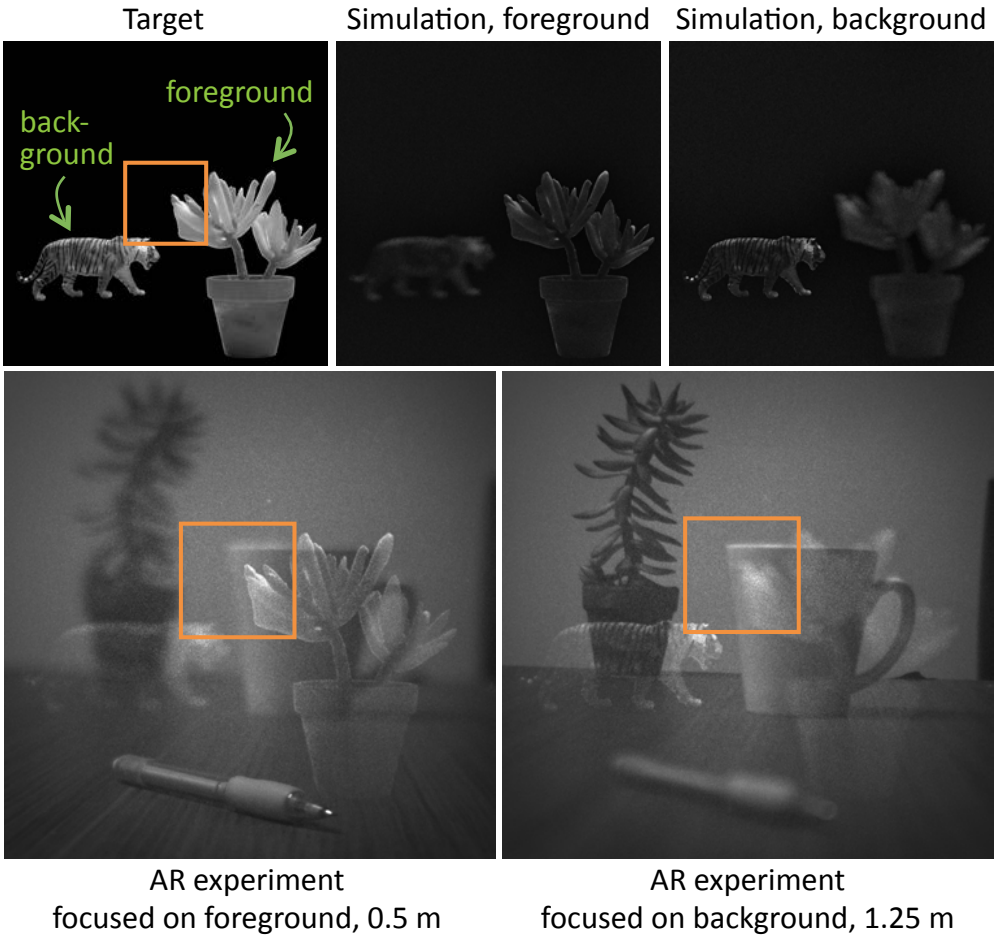

Figure 7.7: Augmented reality prototype demonstration with multi-plane content shown at two different focal distances. The orange box represents the native FoV of the SLM without étendue expansion.

Experimental Results

Figure 7.6 shows captured images from our experimental prototype comparing the image calculation algorithms described in Section 7.3. All images are at 16× expansion. Although the one-step backpropagation algorithm works well for sparse scenes, it does not extend to dense, photographic imagery. As with the simulation results, we see strong improvement in contrast when using our algorithm compared to prior work, and contrast is further enhanced when using the frequency constraints. Applying spatial constraints creates higher contrast but only over a limited region determined by our provided spatial weighting map. Areas that are not prioritized by the spatial map may appear to be "missing" content due to reduced contrast in these regions, but recall that the spatial map is user-specified and can easily be translated to any location, enabling high quality content anywhere in the FoV. Although

Figure 7.8: We propose three methods to further improve contrast for 16× étendue enhancement beyond the performance of our baseline frequency constrained method (shown in second panel). First, with emerging complex-valued SLM technology, image contrast can be improved over the whole FoV. Second, since the eye tends to rotate only within a limited range before the head moves, in a 120° display a large portion of the FoV is almost always in the periphery. Therefore, we can improve contrast in the center by applying our spatial constraints in a fixed fovation pattern that does not require eye tracking. Finally, if eye tracking is available, our spatial constraints can be used to improve quality in a dynamically changing subregion, highlighted by the green circle. The orange box indicates the native FoV, and the spatial maps used in the foveated simulations are shown in the insets. Floating houses source image by Madeleine Deaton (CC BY 2.0).

our experimental results do not yet match the quality of the simulations (discussed more in Sec. 7.5), ours is the first prototype to demonstrate dense, higher resolution imagery outside of the SLM's native FoV.

We further demonstrate the multi-plane capabilities of our system in an augmented reality (AR) prototype. We re-arrange the imaging path of our benchtop prototype to include a beamsplitter, creating a see-through path (see inset of Figure 7.5). Figure 7.7 shows a multi-plane image captured through our AR setup, displaying a small plant in the foreground and a tiger in the background. This highlights the advantages of a holographic display compared to a stereoscopic display: the holographic images contain correct monocular focal cues, which can be seen when the hologram is defocused. However, as with other holographic displays, we cannot easily display occlusions and all virtual objects appear transparent.

7.5 Discussion

Our work represents a step towards practical holographic near-eye displays by breaking the trade-off between FoV and eyebox size. As an illustrative example, an ideal display may have a 120° FoV and a 1 cm eye box, such that the eye can rotate freely and maintain view of the image. Such a configuration could theoretically be achieved with our method by using emerging 8K SLMs (which have been demonstrated [139]) and a scattering mask with 16× expansion. Note that without the expansion mask, an 8K SLM scaled to have a 120°

FoV would provide 66 pixels/degree of resolution, just above what normal human vision can perceive. Therefore, after applying our frequency constraints, which limits the final output resolution to the native SLM resolution, noise is pushed into imperceptible frequencies.

However, based on our simulations in Figure 7.3, there is a visible reduction in contrast at 16× étendue expansion. This might be acceptable in some scenarios, in particular for augmented reality in which content tends to be sparse, and light from the world changes the perception of contrast. For cases where the content is dense and the contrast loss is too severe, we propose three potential solutions, simulated in Figure 7.8. Although current commercial SLMs have phase-only modulation, complex modulation is a potential solution to increase performance, and is often achieved with cascaded modulators [131]. If complex SLMs are not available, another solution is based on fixed foveation: although a large FoV is important for an immersive display, the eye tends to only rotate within ±18° on average before the head moves [45]. Conservatively we will assume that half of the 120° FoV is almost always in peripheral vision and only the central 60° must be highly optimized. In Fig. 7.8, we simulate a fixed foveation falloff using our spatial constraints and show restored contrast in the center region, without the use of eye tracking. However, if eye tracking is available, the contrast can be further improved by moving around a smaller tracked foveal region based on the viewer's gaze direction.

Challenges and Future Work

Although our work demonstrates progress toward more practical holographic displays, there is still additional work to be done to achieve a full-color display with high resolution, complete focal depth cues, and a sunglasses-like form factor. We discuss some key challenges below.

Model Mismatch Although our experimental system shows the potential of scattering-based étendue expansion, the contrast and quality of the early prototype is noticeably lower than that of the simulation. We conjecture that this is due to the high sensitivity of the system to alignment errors, particularly if the scattering mask is not at the correct location. Our alignment procedure removes misalignment that can be represented as a rigid transformation, but can only coarsely account for non-rigid distortion. Since the image of the SLM is relayed through a $4f$ system, we encounter geometric distortion from the lenses, even though the lenses are well corrected for aberrations. In Figure 7.9, we simulate the effect of a small amount of geometric distortion in the SLM pattern, less than a 30 µm translation at the edge of the SLM. Even after applying our coarse correction procedure, we see that the simulations with model mismatch yield qualitatively similar results to the experiment. This problem could be mitigated in future work by omitting the $4f$ system and using a volume grating to remove the DC term [11]. We also observe vignetting in experimental results, which can be alleviated with improved design of the imaging relay system. We find that using all the SLM pixels without binning results in further degradation in image quality, which we suspect is due to higher tolerances on alignment and increased cross-talk between the SLM pixels [107].

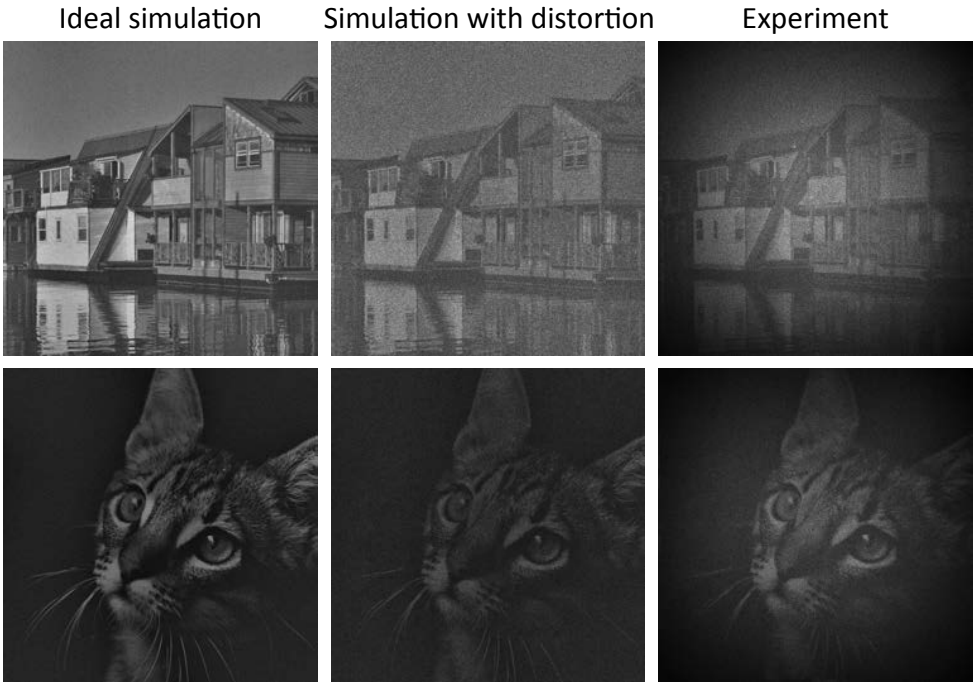

Figure 7.9: Performance of our experimental prototype does not yet match the simulations. We ascribe this mismatch to the high sensitivity of the mask alignment, and although we precisely align the mask through the 6 degrees of freedom of rigid transformation, we do not accurately account for effects such as geometric distortion. Here, we simulate the effect of a small amount of distortion and observe that this model mismatch creates qualitatively similar contrast to the experimental results. Additional vignetting from the imaging system is also apparent in the experimental images. Floating houses source image by Madeleine Deaton (CC BY 2.0); cat source image by Lali Masriera (CC BY 2.0).

Miniaturization The prototype presented in this work is intended as a proof-of-concept; the final design is ideally a wearable display with a sunglasses-like form factor. Starting with the design presented by Maimone, Georgiou, and Kollin [101], which had promising form factor and FoV but very limited eyebox, we propose integrating our scattering mask into the holographic optical element that acts as an image combiner. Figure 7.10 shows a simplified schematic of this idea. A display with a traditional holographic image combiner, shown on the left, is recorded by interfering two beams to create a volume hologram that relays the projected light to the eye box. To fabricate the holographic image combiner with the encoded scattering mask (Fig. 7.10 right), we propose placing a lithography-printed phase mask, like the one used in our benchtop prototype, in front of the holographic optical element during recording. We expect this will result in an image combiner that both relays the projector light to the eye box and implements the scattering mask in one compact optical element.

As in our benchtop system, we need an accurate model of the effect of the scattering

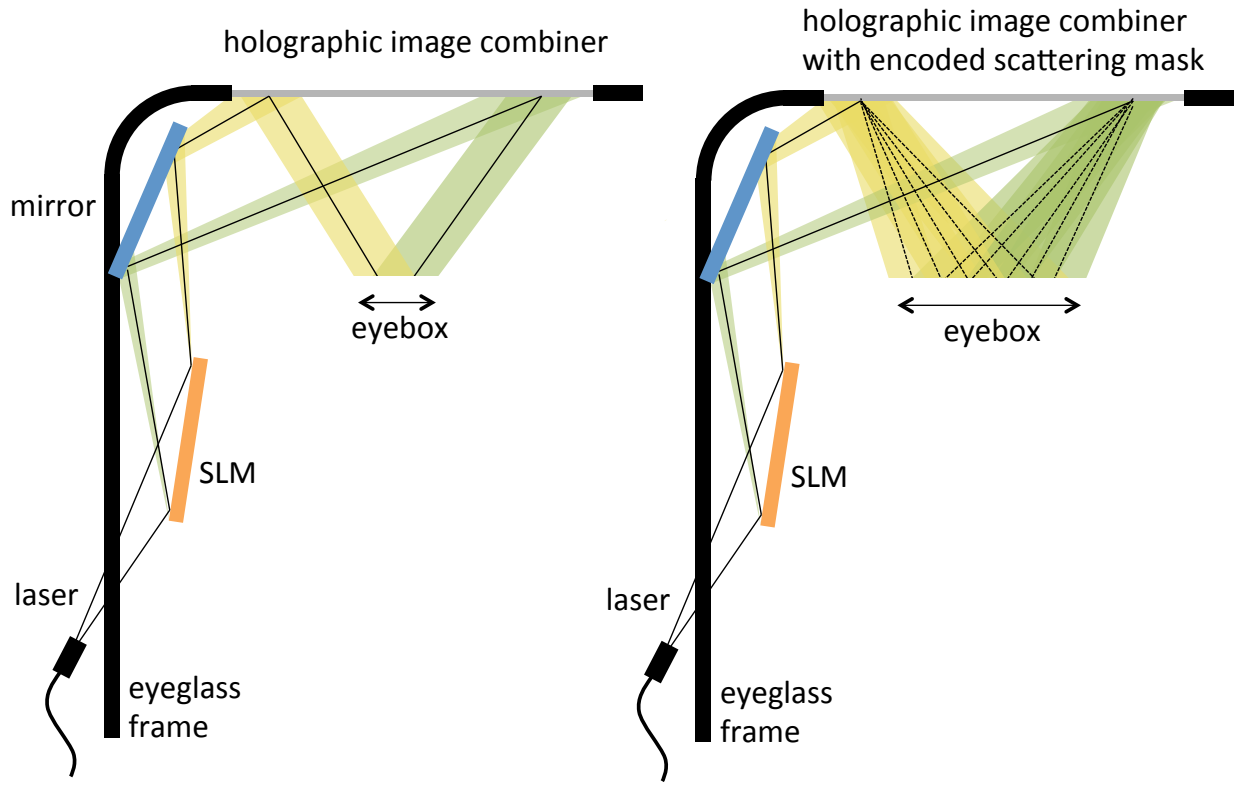

Figure 7.10: Proposed future scheme for integrating our étendue expansion mask into an sunglasses-like form factor display. (Left) Schematic of compact holographic display prototype based on the work of Maimone, Georgiou, and Kollin [101]. (Right) By encoding the scattering mask in the holographic image combiner, the eyebox can be increased without sacrificing FoV, adding additional optical components, or compromising form factor.

mask such that our algorithm can compensate for the scattering in the SLM pattern. In the miniaturized off-axis configuration, we can no longer assume that the SLM and mask are parallel or at the same plane. However, these effects can be physically modeled with minimal additional compute by including free-space propagation using Fresnel or angular spectrum methods [54] and modelling electric fields at non-parallel planes using the fast approach described by Matsushima, Schimmel, and Wyrowski [105].

Color Our prototype displays images in the red channel only; there are several additional considerations for full color display. Maimone, Georgiou, and Kollin [101] provide full color by field sequential operation, i.e., displaying holograms for the red, green, and blue channels in rapid succession on the SLM in conjunction with synchronized laser sources. Wavelength-multiplexed volume holograms are used for the static optical components, which allow independent operation for each color with very little crosstalk. A per-channel system calibration process is also used to reduce any residual differences between color channels. We could also

apply the strategy of Maimone, Georgiou, and Kollin [101] by using field sequential operation on our SLM, replacing our binary phase scattering mask with a wavelength-multiplexed volume hologram optimized for each color channel, and performing the calibration procedure of Section 7.4 for each channel. We expect that his method will be successful for our proposed display as each color channel can be optimized and calibrated independently; however, we have yet to experimentally validate this method.

Compute Time Currently, compute is performed offline and takes about 3 minutes to generate the hologram for the binned 960×540 SLM with $16\times$ étendue expansion in MATLAB code on a GeForce GTX 1060 GPU. Accelerating compute was not prioritized in this work, but will be necessary to make a practical real-time display. Improved hardware acceleration will be critical, and future work incorporating temporal consistency into the hologram calculation might further reduce compute time for each frame.

Perceptual effects In this work, we model the human visual system simply as a low-pass filter that removes high spatial frequencies. However, it is possible human subjects may actually perceive high frequencies in the image in more complicated ways. In order to ensure that the viewer experiences the desired effect, future work includes perceptual studies of the display architecture and development of corresponding biologically-inspired loss functions.

Chapter 8

Conclusion

This dissertation explored how joint design of optical hardware and processing algorithms can enable new imaging systems. By relying more heavily on data processing, these computational imaging systems can be physically simpler while capturing (or displaying) more information than their traditional counterparts. In particular, we break away from the idea that the sensor measurement (or display panel) should *look* like an image; instead, we treat each pixel as just a data point and apply optimization-based approaches to recover an image. In this chapter, we reflect on some common themes and challenges in computational imaging systems, and suggest directions for future work.

8.1 Themes and Challenges

Computational imaging systems can provide remarkable capabilities by leveraging domain-specific information, but this restricts the system to a particular domain. In DiffuserCam, we achieved compact, easy-to-assemble hardware with 3D capture, but we relied on a sparsity prior on the object, limiting the device to samples that match the prior. In our holographic display, we similarly create a domain-specific device through our perceptually inspired constraints: here, we've assume a particular model of perception, and if a "viewer" (for example, a high resolution camera) deviates too far from the model, the image will not be perceived correctly. A general trend is that we can continue to improve system performance by increasing specificity with more targeted priors. For example, if we restrict our diffuser microscope to the task of imaging neurons, we can incorporate additional constraints, and prior work suggests this added information can improve 3D reconstructions in the presence of background fluorescence and scattering [111]. Due to the increasing prevalence of imaging systems today, it's common for a device to be used for only one task, and there are many applications where it's desirable to have a very specific yet powerful imaging system – computational imaging can provide this!

One challenge for computational imaging systems are the large data sizes of images and videos; efficient computation is critical since brute force approaches quickly become unwieldy. In this work, we purposefully use a thin diffuser in both the camera and display

systems since the diffuser only modulates the light in a single 2D plane. Therefore, the diffuser can be characterized by a single 2D function (the PSF in DiffuserCam and the scattering mask phase in the holographic display), greatly reducing the computational burden of these systems. In addition, throughout this work we consistently choose optimization algorithms like ADMM and FISTA that are memory efficient and do not require computing the Hessian. Unfortunately, the compute times are still slow compared to traditional imaging systems. However, recent work using feed-forward neural networks has shown close to real-time performance on some similar systems [106, 77, 120], and leveraging cloud computing can reduce compute times for parallizable problems (e.g. video).

Finally, calibration is a critical element of most computational imaging systems since solving the inverse problem relies on an accurate model of the optics. In this work, we use two different calibration strategies. In DiffuserCam, we first assembled the system and then captured calibration measurements to characterize the diffuser, which included all misalignment and non-idealities. This procedure worked well because the calibration measurements were experimentally straight-forward to acquire and we did not know anything about the diffuser profile beforehand. In contrast, in the holographic display system, we assumed we knew the diffuser (or scattering mask) profile and instead of calibrating the system after the fact, we carefully aligned the diffuser in the system to match the forward model. This approach worked well since the diffuser was fabricated with a well-known procedure (lithography) and we could count on the accuracy of the profile. Furthermore, calibration measurements are more challenging in a display system, since there is no sensor inherent in the design. An alternative intermediate approach (not employed in this dissertation), would be to pre-calibrate the diffuser profile in a separate metrology system (e.g. a white light interferometer) and then take calibration measurements after assembly to determine just the alignment of the diffuser. I don't believe there is a single "best" approach to calibration; however, it is worthwhile to consider a range of approaches for any given system because often the success of the device depends on reliable calibration.

8.2 Extensions and Future Work

Beyond this work, there are several potential extensions that could improve the capabilities of diffuser-based imaging systems. These include modifications to the algorithms for improved image quality (ex. higher dimensional priors) or faster computation (ex. computation in the diffuser domain) as well as modifications to the optical hardware (ex. engineered diffusers, active illumination) for improved robustness and resolution. We expand on these ideas below.

Higher dimensional priors: In this work, we apply several sparsity priors to enable recovery of images in ill-posed or underdetermined inverse problems. However, all of the priors used here are in the 2D or 3D image domain, for example using total variation to enforce a sparse spatial gradient. A natural extension is to apply priors along either color (for photography applications) or time (for applications involving video). For instance, color priors might take advantage of the general trend that edges are usually aligned between

color channels [59] and that the basic structure of each color channel is similar. Temporal priors for general scenes could include directly applying total variation along the temporal direction [9], estimating non-rigid models of motion [76], or assuming low rank models, potentially at multiple scales [114].

A specific case where temporal priors can be particularly strong is for the application of neural imaging. Here, a test animal is genetically modified so that its neurons fluoresce when they fire [141]. Microscopes for neural imaging aim to localize the neurons and measure the temporal changes in fluorescence, which correspond to neural activity. In this scenario, the video can be registered so neurons do not move between frames, resulting in very strong priors on the signal. For instance, we know the video is low rank, with rank equal to the number of active neurons; furthermore, specific fluorophores like GCaMP [149] have well-documented temporal dynamics which can act as an additional prior on the temporal signal [122].

Engineered diffusers: We touched on the idea of designing the diffuser in Chapter 6 where we demonstrated that the random microlens diffuser outperforms the smooth off-the-shelf diffuser of Chapter 4. However, there are still many open questions about the optimal diffuser design. For example, even within the realm of randomly spaced lenslets, the locations and focal lengths of the lenslets have an impact on the final result; optimization of the lenslet placement based on heuristic metrics has shown good results [160], and on-going work on end-to-end optimization may further outperform the heuristics. However, this still relies on the underlying assumption that the diffuser is composed of lenslets – perhaps there is a better design.

One drawback of the random microlens diffuser is that each lenslet has a small diameter compared to the overall aperture. As a result, each lenslet's diffraction-limited spot has significantly worse resolution than if a single lens covered the whole sensor size. Although the pixel-superresolution can be achieved by combining the images from each lenset [144], the diffraction-limited resolution cannot be improved in this way. In contrast, the "scatter-plate microscope" presented by Singh et al. [134] has much longer propagation distance between the diffuser and sensor, resulting in a speckle PSF with diffraction-limited resolution based on the *entire* aperture. The key difference is that this system allows light from across the entire aperture to interfere creating higher frequencies in the PSF; the downside is that the resulting PSF is very dense and low contrast, making it poorly suited for noise amplification (Chapter 6), which is a major drawback. If I were to redesign the diffuser, I'd aim for a design that creates opportunities for interference between light from different diffuser locations (to improve diffraction-limited resolution) while simultaneously directing the light into only a few spots on the sensor to create a high contrast pattern (for improved noise performance). One possible option would be to engineer prisms under some of the lenslets in the diffuser such that the spots from several lenslets interfere at a single location. This could add high frequency content within each spot while maintaining the sparse, high contrast PSF. Although this design has the same components as the light field camera proposed by Georgiev and Intwala [50], rather than use the prisms to seperate images, here I propose using the prisms to combine images, creating opportunities for optical interference. However, this

just one design; extending end-to-end or optimization-based diffuser design beyond simple lenslets might yield superior results.

There is also potential in engineering the diffuser (scattering mask) used in our holographic display system. Our current design is based on the heuristic that it's desirable to evenly scatter light over the extended FoV, but performance may be improved with a carefully engineered design. One option is to optimize the scattering mask for robustness to misalignment, since alignment is critical in this system. Another is to explore mask designs that are specific to a class of images such as natural scenes, text, or augmented-reality images (where the majority of the image is black to allow the real world to pass through to the viewer). This could be achieved with end-to-end optimization or by heuristically determining good far-field patterns, then optimizing the scattering mask profile with phase retrieval techniques.

Co-design with active illumination: Throughout this work we assumed no control of the illumination on the object (beyond stable excitation illumination for fluorescence). However, particularly for microscopy applications, one can frequently control the illumination with simple hardware. For example, illumination control with an LED array [147, 148] can be very fast and contains no moving parts. One potential benefit of incorporating active illumination into the design of diffuser-based imaging systems is the ability to make a sample appear more sparse than it actually is. A series of acquisitions with changing illumination could capture all information about the sample while enforcing that only limited number of pixels are illuminated. Furthermore, if the illumination pattern is known, it adds an additional constraint in the reconstruction, which could greatly improve image fidelity. Although this approach might require multiple acquisitions, the lack of moving parts in both illumination and detection would help maintain high frame rates, and acquisition speeds could be increased further by encoding multiple illumination patterns as different color channels on a single camera, as in [121].

Head-mounted displays also provide a natural platform for active illumination since the illumination is integrated into the device. Our holographic display assumed a constant plane-wave illumination on the SLM, but modulating the intensity on different parts of the SLM could increase the number of degrees of freedom in the system and improve image contrast, since light can be removed (instead of only redirected). By changing the incoming illumination angle, one could also shift the imagery to different parts of the field-of-view (similar to [67]). Since design of a compact head-mounted display is an incredible engineering challenge, and I believe all aspects of the system, including illumination, will need to be optimized for success.

Direct computation in the diffuser domain: In most of this work, our end goal was a human-viewable image. However, in many applications (e.g. security, robotics, and metrology) one wishes to automatically extract higher-level information from the image; a reconstructed photograph is unnecessary. Therefore, it's computationally efficient if one can directly extract the desired information without the time-consuming full image reconstruc-

tion. Preliminary work has shown the effectiveness of neural networks for classification tasks directly on raw data that does not look like an image [79], eliminating the need for a reconstruction. Enabling tasks like structure from motion (SfM) directly on the raw data or adapting feature-finding algorithms (ex. SIFT [96]) into the diffuser domain could be invaluable for applications in robotics and autonomy.

A main challenge in adapting existing computer vision algorithms to the diffuser domain is lack of locality: unlike a lens, the diffuser spreads information over the whole sensor, which is not handled well by many algorithms. One option is to generate a fast (but less accurate) reconstruction to improve locality and then modify existing computer vision algorithms to account for the artifacts. For instance, Wiener deconvolution could be used to quickly solve for a low quality image reconstruction with DiffuserCam; however, streaky artifacts in the reconstructions may necessitate modifications to downstream algorithms.

The same problem exists in holographic displays as well. A user may want to edit imagery in a small region of the FoV, but this requires recomputing the entire SLM pattern. The ability to edit or combine images directly from their SLM patterns could dramatically speed-up computation times, especially for video where we expect many similarities between frames. However, just like in DiffuserCam, each pixel on the display panel effects every point in the scene, resulting in a challenging lack of locality. However, if the system is redesigned to have a shorter propagation distance between the SLM and the scene, some locality is preserved. Redesigning the scattering mask to maintain this locality may enable new algorithms where imagery is edited by directly modifying the SLM pattern.

Appendix A

Implementation Details

A.1 Total Variation with FISTA

Here, we briefly describe an efficient implementation of total variation regularization with FISTA. As described in Chapter 2, FISTA promotes sparsity in some basis by transforming a vector into that basis, performing soft-thresholding, and then reverting with an inverse transformation. Total variation sparsity seeks to promote a sparse spatial gradient. It is easy to compute the discrete spatial gradient by translating the image by one pixel then subtracting, and soft-thresholding is trivial to implement. However, the challenge comes when transforming back to the native domain since taking the gradient is not invertible.

The trick to making this work is to keep track, not only of the differences between neighboring pixels, but also the sum, which gives enough information to undo the transformation. Let $\vec{\boldsymbol{y}}$ be a vector, and $\vec{\boldsymbol{y}}^{(k)}$ represent a circularly translated version of the vector by k elements. We can compute the following difference term and sum term between neighboring pairs of pixels as follows:

$$\begin{aligned} \vec{\boldsymbol{y}}_{\text{diff}} &= \tfrac{1}{2}(\vec{\boldsymbol{y}}^{(-1)} - \vec{\boldsymbol{y}}) \\ \vec{\boldsymbol{y}}_{\text{sum}} &= \tfrac{1}{2}(\vec{\boldsymbol{y}}^{(-1)} + \vec{\boldsymbol{y}}). \end{aligned} \tag{A.1}$$

From these difference and sum terms, we can recompute the original vector:

$$\vec{\boldsymbol{y}} = \tfrac{1}{2}(\vec{\boldsymbol{y}}_{\text{sum}} - \vec{\boldsymbol{y}}_{\text{diff}}) + \tfrac{1}{2}(\vec{\boldsymbol{y}}_{\text{sum}} + \vec{\boldsymbol{y}}_{\text{diff}})^{(+1)}, \tag{A.2}$$

where once again the superscript notation denotes a circular translation of the elements in the vector.

With this formulation, we can use total variation in FISTA by first calculating the sum and difference terms in Eq. A.1, then applying soft-thesholding on just the difference term, and finally recombining the two terms based on Eq. A.2. For higher dimensional signals (ex. 2D images, 3D volumes), we can do this procedure separately along each dimension, and then average the result. This procedure highly related to wavelet denoising with Haar wavelets, and the connection is described in detail in [71].

In Chapter 3, we use total variation with ADMM, instead of FISTA, which is even more efficient since ADMM generally converges in fewer iterations. However, it is challenging to use ADMM efficiently on many forward models since it depends on the specific model structure, and there are many more hand-tuned parameters in ADMM, making FISTA a good choice in many applications.

A.2 Properties of the Smooth Diffuser

To quantify the properties of the smooth diffuser used in Chapters 2 and 3, we used an LED array microscope to capture a quantitative Differential Phase Contrast (DPC) [147] image of the diffuser phase. After using the index of refraction of the diffuser material (polycarbonate, $n = 1.58$) to convert phase into surface shape, we show in Fig. A.1 the measured relative height profile of a small patch on our 0.5° diffuser. The surface slope of the diffuser is Gaussian distributed with average magnitude of 0.7°. The deflection angle at the diffuser surface has a HWHM angle of 0.25°, which matches the manufacturer specifications. The maximum deflection angle is $\beta = 0.5°$, as shown in the histograms in Fig. A.1.

To illustrate the overall size and spread of the caustic PSF patterns in our system, we show in Fig. A.2 the full PSF patterns captured for the closest and farthest axial distances used. Note that the closest axial distance is the one at which the caustic pattern just fills the sensor, and therefore depends on the aperture size. The caustics contain high-frequency information in all orientation directions, as evidenced by the sharp lines randomly spread in all directions. This facilitates good resolution at all depths and a highly structured PSF for deconvolution. Our calibration point source is a $30\mu m$ pinhole illuminated by a planar RGB LED array (λ = 630 nm, 515 nm, and 460 nm, $\Delta\lambda$ = 20 nm, 35 nm, and 25 nm, respectively) placed behind a 80° diffuser. As shown in [6], the caustics from narrowband and broadband sources are indistinguishable, and we do not find problems with using narrowband calibration.

PSF similarity

We quantify the similarity of the PSF versus shift and scale across the volume to validate our claim that the resulting underdetermined matrix has good properties for sparse recovery techniques. Figure A.3 shows the autocorrelation of the PSFs acquired at the minimum and maximum object distances, as well as the cross-correlation between the two. Notice that the PSF autocorrelation maintains a sharp central peak and relatively low sidelobes for all depths within our calibration volume. This means that a shifted version of the PSF is roughly 50% similar to the un-shifted version. Importantly, the cross-correlation has no values greater than 50%, meaning that the scaled caustics are dissimilar to any shift of the unscaled caustics. To quantify this further, we plot the inner product between the central image in the calibration stack, corresponding to the orange dotted line in Fig. A.3b, with all other images in the stack. We again observe a relatively sharp peak and side lobes on

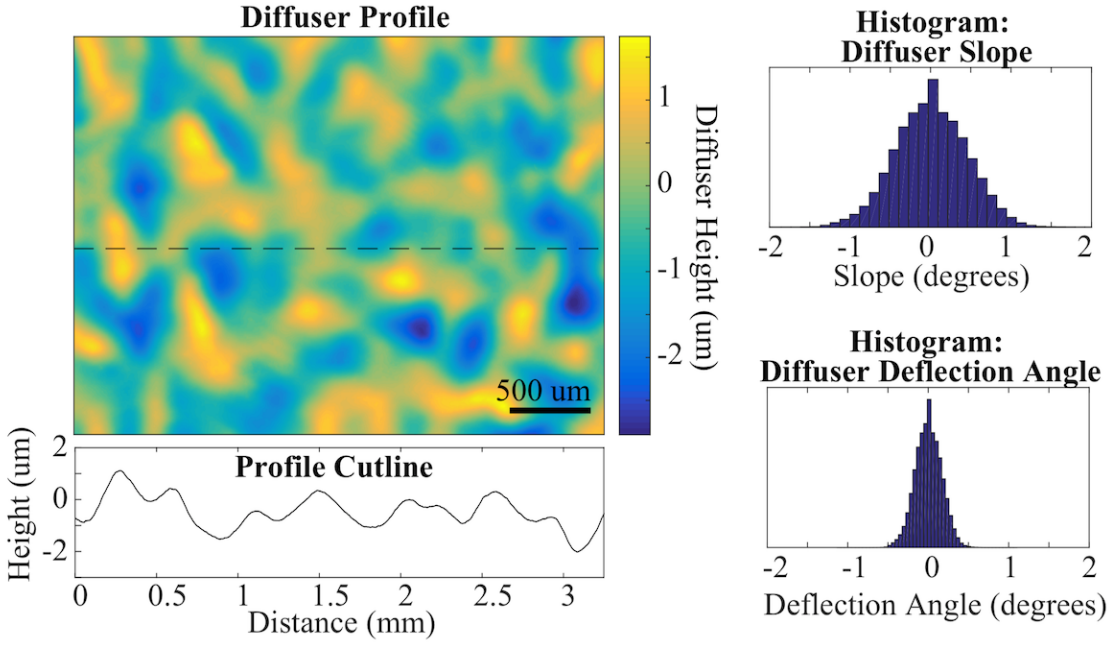

Figure A.1: Left: The thickness profile of a small patch of our diffuser, as measured by quantitative Differential Phase Contrast (DPC) microscopy. Below is a cut-line plot along the dashed line. Right: Histograms of the diffuser slope (top) and the deflection angle of a ray normally incident on the diffuser (bottom).

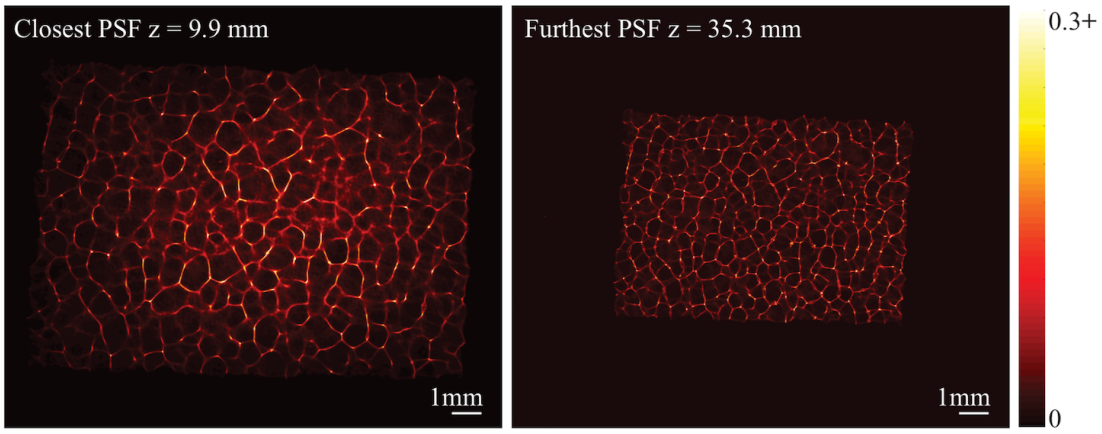

Figure A.2: Un-cropped, false color sensor measurements of PSFs for the closest and farthest planes used in our reconstructions. These were measured by placing a point source on-axis at the front and back of the volume. The closest PSF has a caustic pattern that fills the sensor. Both PSFs have been contrast stretched from 0 to 30% of the max value for visibility.

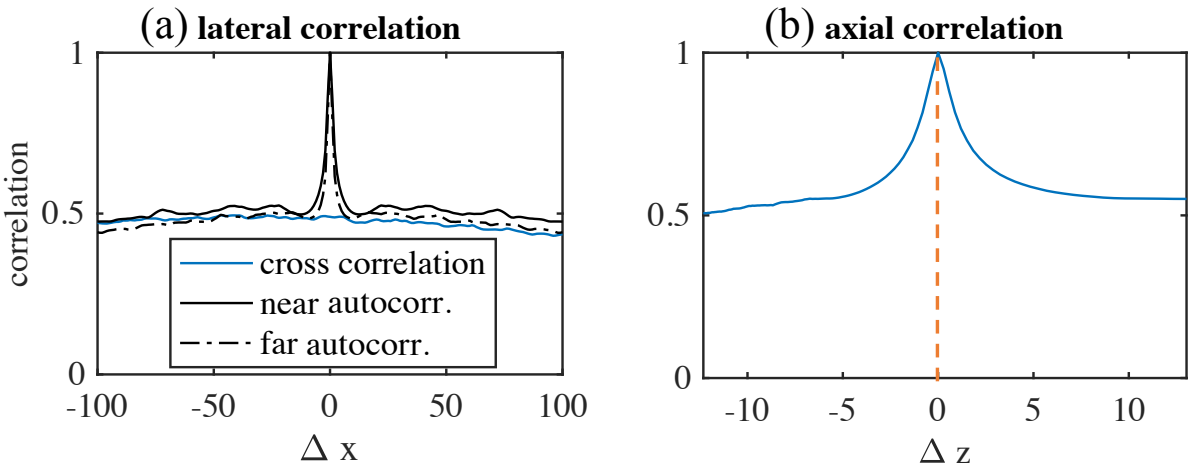

Figure A.3: Correlation of various caustics patterns. (a) The caustics at a given depth are unique over shifting, and caustics from two different depths are not similar to each other, even under translation. The solid black curve is a slice of the autocorrelation of a PSF for a point source near the front of the volume, and the dotted black line is the autocorrelation for a far away point source's PSF. The solid blue line is the cross-correlation between the two. (b) The inner product of the PSF from the middle of the volume (corresponding to the orange dotted line) with all other PSFs at varying depths. In both (a) and (b), shifting or scaling the caustics leads to an inner product of approximately 0.5 compared to a peak value of 1.

the order of 50% in the axial direction. This validates our claim that the caustics produced by any point in the volume are unique.

A.3 Derivation of ADMM Inverse Algorithm

Throughout this work, the problem we seek to solve is:

$$\hat{\vec{\boldsymbol{y}}} = \underset{\vec{\boldsymbol{y}} \geq 0}{\operatorname{argmin}} \tfrac{1}{2} \|\vec{\boldsymbol{b}} - \boldsymbol{A}\vec{\boldsymbol{y}}\|_2^2 + \tau \|\Psi \vec{\boldsymbol{y}}\|_1. \tag{A.3}$$

We can solve this with FISTA, as explained in the introduction. However, FISTA can be very slow since it requires a large number of iterations to converge. For specific formulations of $\boldsymbol{A}$, we can also solve this optimization problem with ADMM, which we do for the 3D DiffuserCam described in Chapter 4. To do this we transform Eq. A.3 this into the equivalent problem:

$$\begin{aligned}\hat{\vec{\boldsymbol{y}}} = \underset{w,u,v}{\operatorname{argmin}} & \tfrac{1}{2}\|\vec{\boldsymbol{b}} - \boldsymbol{D}v\|_2^2 + \tau\|u\|_1 + \mathbb{1}_+(w) \\ \text{s.t. } & v = \mathbf{M}\vec{\boldsymbol{y}} \\ & u = \Psi\vec{\boldsymbol{y}} \\ & w = \vec{\boldsymbol{y}},\end{aligned} \tag{A.4}$$

where $\mathbb{1}_+(\cdot)$ is the nonnegativity barrier function, which returns 0 when the argument is nonnegative, and ∞ when the argument is negative.

In order to compute the ADMM updates efficiently, we will see that it is useful for both $\mathbf{M}$ and Ψ to represent 3D convolutions. Clearly, when Ψ is the identity matrix, this holds. Additionally, when Ψ is the 3D finite difference operator, it can be expressed as a concatenation of 3D convolutions with the finite difference kernel, oriented in each of the 3 directions. In order to express $\mathbf{M}$ as a 3D convolution, we must choose the diagonal operator, $\boldsymbol{D}$, such that Eq. (4) can be written as $\boldsymbol{D}\left(m \overset{(x,y,z)}{*} \vec{\boldsymbol{y}}\right)$, where m is a 3D kernel, and $\overset{(x,y,z)}{*}$ represents convolution over the variables, x, y, and z. To accomplish this, we use the fact that a sum of 2D convolutions between an object, $\vec{\boldsymbol{y}}(x,y,z)$, and a stack of 2D kernels, $h(x,y;z)$, can be expressed as the first 2D (x,y)-slice in the 3D convolution between the object and a z-flipped version of the kernel stack:

$$\sum_z h(x,y;z) \overset{(x,y)}{*} \vec{\boldsymbol{y}}(x,y,z) = \left[h(x,y;-z) \overset{(x,y,z)}{*} \vec{\boldsymbol{y}}(x,y,z)\right]\Big|_{z=0}. \tag{A.5}$$

For proof, we can take the right hand side of (A.5) and apply the definition of discrete 3D convolution directly:

$$\begin{aligned}&\left[h(x,y;-z) \overset{(x,y,z)}{*} \vec{\boldsymbol{y}}(x,y,z)\right]\Big|_{z=0} \\ =&\sum_{z'=0}^{N_z-1}\sum_{y'=0}^{N_y-1}\sum_{x'=0}^{N_x-1} \vec{\boldsymbol{y}}(x',y',z')h(x-x',y-y';z'-z)|_{z=0} \\ =&\sum_{z'=0}^{N_z-1} \vec{\boldsymbol{y}}(x,y,z') \overset{(x,y)}{*} h(x,y;z').\end{aligned}$$

Using this identity, we can write the forward operator in Eq. (4) as:

$$\begin{aligned}&\boldsymbol{C}\sum_z\left[\vec{\boldsymbol{y}}\left(\frac{-x'}{m},\frac{-y'}{m},z\right) \overset{(x,y)}{*} h\left(x',y';z\right)\right] \\ =&\boldsymbol{C}\left[\vec{\boldsymbol{y}}\left(\frac{-x'}{m},\frac{-y'}{m},z\right) \overset{(x',y',z)}{*} h\left(x',y';-z\right)\Big|_{z=0}\right] \\ =&\boldsymbol{D}\left[\vec{\boldsymbol{y}}\left(\frac{-x'}{m},\frac{-y'}{m};z\right) \overset{(x',y',z)}{*} h\left(x',y';-z\right)\right],\end{aligned}$$

where $\boldsymbol{D}$ is a diagonal operator that simultaneously performs the 2D crop, $\boldsymbol{C}$, as well as selecting the $z = 0$ slice. Effectively, $\boldsymbol{D}$ comprises taking the center crop of the first layer of the 3D array resulting from the circular 3D convolution of $h(x', y'; -z)$ with $\vec{\boldsymbol{y}}$. Note that our definition of z is as a parameter indexing each slice in the 3D array h, not the physical distance to each slice. We assume circular boundary conditions for h, such that $h(\cdot, \cdot; -z) = h(\cdot, \cdot; N_z - z)$ is a z-stack that is flipped in the z-direction.

Using (A.5), we present an efficient method for solving (A.4). We begin by transforming (A.4) into an unconstrained augmented Lagrangian form, and consider the saddle-point problem:

$$\begin{aligned}\max_{\xi,\eta,\rho}\Bigg[\min_{u,v,w,\vec{\boldsymbol{y}}} \tfrac{1}{2}\left\|\vec{\boldsymbol{b}} - \boldsymbol{D}v\right\|_2^2 + \tfrac{\mu_1}{2}\left\|\mathbf{M}\vec{\boldsymbol{y}} - v + \tfrac{\xi}{\mu_1}\right\|_2^2 \\ + \tau\|u\|_1 + \tfrac{\mu_2}{2}\left\|\Psi\vec{\boldsymbol{y}} - u + \tfrac{\eta}{\mu_2}\right\|_2^2 \\ + \mathbb{1}_+(w) + \tfrac{\mu_3}{2}\left\|\vec{\boldsymbol{y}} - w + \tfrac{\rho}{\mu_3}\right\|_2^2\Bigg].\end{aligned}$$

To solve the above equation using ADMM, we first derive the optimality conditions for each primal variable, assuming the others are fixed:

$$\begin{aligned}u^{k+1} &\leftarrow \operatorname*{argmin}_u \quad \tau\|u\|_1 + \tfrac{\mu_2}{2}\left\|\Psi\vec{\boldsymbol{y}}^k - u + \tfrac{\eta^k}{\mu_2}\right\|_2^2 \\ v^{k+1} &\leftarrow \operatorname*{argmin}_v \quad \tfrac{1}{2}\left\|\vec{\boldsymbol{b}}^k - \boldsymbol{D}v\right\|_2^2 + \tfrac{\mu_1}{2}\left\|\mathbf{M}\vec{\boldsymbol{y}}^k - v + \tfrac{\xi^k}{\mu_1}\right\|_2^2 \\ w^{k+1} &\leftarrow \operatorname*{argmin}_w \quad \mathbb{1}_+(w) + \tfrac{\mu_3}{2}\left\|\vec{\boldsymbol{y}}^k - w + \tfrac{\rho^k}{\mu_3}\right\|_2^2 \\ \vec{\boldsymbol{y}}^{k+1} &\leftarrow \operatorname*{argmin}_{\vec{\boldsymbol{y}}} \quad \tfrac{\mu_1}{2}\left\|\mathbf{M}\vec{\boldsymbol{y}} - v^{k+1} + \tfrac{\xi^k}{\mu_1}\right\|_2^2 \\ &\qquad + \tfrac{\mu_2}{2}\left\|\Psi\vec{\boldsymbol{y}} - u^{k+1} + \tfrac{\eta^k}{\mu_2}\right\|_2^2 \\ &\qquad + \tfrac{\mu_3}{2}\left\|\vec{\boldsymbol{y}} - w^{k+1} + \tfrac{\rho^k}{\mu_3}\right\|_2^2.\end{aligned}$$

And update each dual variable as

$$\begin{aligned}\xi^{k+1} &\leftarrow \xi^k + \mu_1(\mathbf{M}\vec{\boldsymbol{y}}^{k+1} - v^{k+1}) \\ \eta^{k+1} &\leftarrow \eta^k + \mu_2(\Psi\vec{\boldsymbol{y}}^{k+1} - u^{k+1}) \\ \rho^{k+1} &\leftarrow \rho^k + \mu_3(\vec{\boldsymbol{y}}^{k+1} - w^{k+1}).\end{aligned}$$

The final result is the algorithm outlined in Chapter 4 of the main text.

A.4 Holographic Display Image Calculation Algorithm

Here we provide additional details on the iterative algorithm used in Chapter 7. Although there are several options for solving Eq. 7.14, we use projected gradient descent with Nesterov acceleration, which is based on the work of Beck and Teboulle [12] and summarized in Algorithm 1 in the introduction. In the algorithm, μ is the user-defined step-size, $\nabla\mathcal{L}$ is the gradient of the loss with respect to $\vec{s}$, and $\text{prox}\{\cdot\}$ is the proximal operator that constrains the SLM pattern to be phase-only:

$$\text{prox}\{\vec{s}\} = \vec{s}/|\vec{s}|. \tag{A.6}$$

The proximal operator is equivalent to setting the amplitude of every element of $\vec{s}$ to one.

Since implementation of the algorithm requires discretization of all variables, we summarize the complete loss function below in discrete vector representation.

$$\begin{aligned}
&\mathcal{L} = \tfrac{1}{2}||\vec{g}||^2, \\
&\vec{g} = \vec{c}_s \odot \mathcal{F}^{-1}\{\vec{c}_f \odot \mathcal{F}\{\vec{I} - \hat{I}\}\}, \\
&\vec{I} = |\vec{y}|^2, \\
&\vec{y} = \mathcal{F}\{\vec{m} \odot U\vec{s}\}.
\end{aligned} \tag{A.7}$$

Here, $\mathcal{L}$ is the scalar loss, $\vec{g}$ is a new intermediate variable, $\odot$ represents element-wise multiplication, and U is an upsampling operation with a box filter (the discrete version of Eq. 7.15). $\vec{c}_f, \vec{c}_s, \vec{I}, \hat{I}, \vec{y}$ and $\vec{m}$ are discrete vector versions of $c_f(\vec{u}'), c_s(\vec{x}), I(\vec{x}), \hat{I}(\vec{x}), y(\vec{x})$ and $m(\vec{u})$ respectively. The above equations use the custom loss function from Eq. 7.19, but the other loss functions can be achieved by setting $\vec{c}_s$ or $\vec{c}_f$ (or both) to all ones.

The gradient $\nabla\mathcal{L}$ is with respect to the complex variable $\vec{s}$, so we calculate the gradient using Wirtinger derivatives [23].

$$\nabla\mathcal{L} = \left(\frac{d\mathcal{L}}{d\vec{s}}\right)^* = \left(\frac{d\vec{I}}{d\vec{s}}\right)^* \left(\frac{d\mathcal{L}}{d\vec{I}}\right). \tag{A.8}$$

Note that $\mathcal{L}$ and $\vec{I}$ are both real valued, so there is no need for the complex conjugate on the second term. Since $\frac{d\vec{I}}{d\vec{s}}$ yields a matrix, we can think of it as an operator that acts on the vector $\frac{d\mathcal{L}}{d\vec{I}}$. This yields the gradient needed for Algorithm 1:

$$\begin{aligned}
\nabla\mathcal{L} &= U^T\left\{\vec{m}^* \odot \mathcal{F}^{-1}\left\{\vec{y} \odot \left(\frac{d\mathcal{L}}{d\vec{I}}\right)\right\}\right\}, \\
\left(\frac{d\mathcal{L}}{d\vec{I}}\right) &= \mathcal{F}^{-1}\{\vec{c}_f \odot \mathcal{F}\{\vec{c}_s \odot \vec{g}\}\},
\end{aligned} \tag{A.9}$$

where U^T is the downsampling (binning) operation such that $\nabla\mathcal{L}$ has the same number of elements as $\vec{s}$

Bibliography

[1] Jesse K Adams et al. "Single-frame 3D fluorescence microscopy with ultraminiature lensless FlatScope". In: *Science advances* 3.12 (2017), e1701548.

[2] Edward H Adelson and John Y. A. Wang. "Single lens stereo with a plenoptic camera". In: *IEEE Transactions on Pattern Analysis & Machine Intelligence* 14.2 (1992), pp. 99–106.

[3] M. V. Afonso, J. M. Bioucas-Dias, and M. A. T. Figueiredo. "Fast image recovery using variable splitting and constrained optimization". In: *IEEE Transactions on Image Processing* 19.9 (Sept. 2010), pp. 2345–2356.

[4] Duygu Akbulut et al. "Focusing light through random photonic media by binary amplitude modulation". In: *Optics express* 19.5 (2011), pp. 4017–4029.

[5] M. S. C. Almeida and M. Figueiredo. "Deconvolving images with unknown boundaries using the alternating direction method of multipliers". In: *IEEE Transactions on Image processing* 22.8 (2013), pp. 3074–3086.

[6] N. Antipa et al. "Single-shot diffuser-encoded light field imaging". In: *2016 IEEE International Conference on Computational Photography (ICCP)*. May 2016, pp. 1–11. DOI: 10.1109/ICCPHOT.2016.7492880.

[7] Nick Antipa et al. *DiffuserCam*. `https://waller-lab.github.io/DiffuserCam/`. Accessed: 2017-11-17. 2017.

[8] Nick Antipa et al. "DiffuserCam: lensless single-exposure 3D imaging". In: *Optica* 5.1 (2018), pp. 1–9.

[9] Nick Antipa et al. "Video from stills: Lensless imaging with rolling shutter". In: *2019 IEEE International Conference on Computational Photography (ICCP)*. IEEE. 2019, pp. 1–8.

[10] M Salman Asif et al. "Flatcam: Replacing lenses with masks and computation". In: *Computer Vision Workshop (ICCVW), 2015 IEEE International Conference on*. IEEE. 2015, pp. 663–666.

[11] Kiseung Bang, Changwon Jang, and Byoungho Lee. "Compact noise-filtering volume gratings for holographic displays". In: *Optics letters* 44.9 (2019), pp. 2133–2136.

[12] Amir Beck and Marc Teboulle. "A fast iterative shrinkage-thresholding algorithm for linear inverse problems". In: *SIAM journal on imaging sciences* 2.1 (2009), pp. 183–202.

[13] Alain Bergeron et al. "Phase calibration and applications of a liquid-crystal spatial light modulator". In: *Applied optics* 34.23 (1995), pp. 5133–5139.

[14] Marcelo Bertalmio et al. "Image inpainting". In: *Proceedings of the 27th annual conference on Computer graphics and interactive techniques*. 2000, pp. 417–424.

[15] Jacopo Bertolotti et al. "Non-invasive imaging through opaque scattering layers". In: *Nature* 491.7423 (2012), pp. 232–234.

[16] Ella Bingham and Heikki Mannila. "Random projection in dimensionality reduction: applications to image and text data". In: *Proceedings of the seventh ACM SIGKDD international conference on Knowledge discovery and data mining*. 2001, pp. 245–250.

[17] Waheb Bishara et al. "Lensfree on-chip microscopy over a wide field-of-view using pixel super-resolution". In: *Optics Express* 18.11 (2010), pp. 11181–11191.

[18] S. Boyd et al. "Distributed Optimization and Statistical Learning via the Alternating Direction Method of Multipliers". In: *Foundations and Trends in Machine Learning* 3.1 (2011), pp. 1–122.

[19] David Brady et al. "Compressive Holography". In: *Opt. Express* 17.15 (July 2009), pp. 13040–13049. DOI: 10.1364/OE.17.013040. URL: http://www.opticsexpress.org/abstract.cfm?URI=oe-17-15-13040.

[20] Michael Broxton et al. "Wave optics theory and 3-D deconvolution for the light field microscope". In: *Optics Express* 21.21 (2013), pp. 25418–25439. DOI: 10.1364/OE.21.025418. URL: http://www.opticsexpress.org/abstract.cfm?URI=oe-21-21-25418.

[21] Antoni Buades, Bartomeu Coll, and J-M Morel. "A non-local algorithm for image denoising". In: *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*. Vol. 2. IEEE. 2005, pp. 60–65.

[22] Edward Buckley et al. "Viewing angle enhancement for two-and three-dimensional holographic displays with random superresolution phase masks". In: *Applied optics* 45.28 (2006), pp. 7334–7341.

[23] Emmanuel J Candes, Xiaodong Li, and Mahdi Soltanolkotabi. "Phase retrieval via Wirtinger flow: Theory and algorithms". In: *IEEE Transactions on Information Theory* 61.4 (2015), pp. 1985–2007.

[24] Emmanuel J Candès. "The restricted isometry property and its implications for compressed sensing". In: *Comptes rendus mathematique* 346.9-10 (2008), pp. 589–592.

[25] Emmanuel J Candès and Michael B Wakin. "An introduction to compressive sampling". In: *IEEE Signal Processing Magazine* 25.2 (2008), pp. 21–30.

[26] Thomas Chaigne et al. "Light focusing and two-dimensional imaging through scattering media using the photoacoustic transmission matrix with an ultrasound array". In: *Optics letters* 39.9 (2014), pp. 2664–2667.

[27] Praneeth Chakravarthula et al. "Wirtinger holography for near-eye displays". In: *ACM Transactions on Graphics (TOG)* 38.6 (2019), p. 213.

[28] Julio Chaves. *Introduction to nonimaging optics*. CRC press, 2017.

[29] Michael Chen, Lei Tian, and Laura Waller. "3D differential phase contrast microscopy". In: *Biomedical optics express* 7.10 (2016), pp. 3940–3950.

[30] Wanli Chi and Nicholas George. "Optical imaging with phase-coded aperture". In: *Optics express* 19.5 (2011), pp. 4294–4300.

[31] Kihwan Choi et al. "Compressed sensing based cone-beam computed tomography reconstruction with a first-order method a". In: *Medical physics* 37.9 (2010), pp. 5113–5125.

[32] Myeong-Ho Choi, Yeon-Gyeong Ju, and Jae-Hyeung Park. "Holographic near-eye display with continuously expanded eyebox using two-dimensional replication and angular spectrum wrapping". In: *Opt. Express* 28.1 (Jan. 2020), pp. 533–547. DOI: 10.1364/OE.381277.

[33] Donald B Conkey, Antonio M Caravaca-Aguirre, and Rafael Piestun. "High-speed scattering medium characterization with application to focusing light through turbid media". In: *Optics express* 20.2 (2012), pp. 1733–1740.

[34] Donald B Conkey et al. "Super-resolution photoacoustic imaging through a scattering wall". In: *Nature communications* 6 (2015), p. 7902.

[35] Ahmet F Coskun et al. "Lensfree fluorescent on-chip imaging of transgenic Caenorhabditis elegans over an ultra-wide field-of-view". In: *PloS one* 6.1 (2011), e15955.

[36] Ahmet F Coskun et al. "Lensless wide-field fluorescent imaging on a chip using compressive decoding of sparse objects". In: *Optics express* 18.10 (2010), pp. 10510–10523.

[37] Oliver Cossairt, Mohit Gupta, and Shree K Nayar. "When does computational imaging improve performance?" In: *IEEE transactions on image processing* 22.2 (2012), pp. 447–458.

[38] Xiquan Cui et al. "Lensless high-resolution on-chip optofluidic microscopes for Caenorhabditis elegans and cell imaging". In: *Proceedings of the National Academy of Sciences* 105.31 (2008), pp. 10670–10675.

[39] Ingrid Daubechies, Michel Defrise, and Christine De Mol. "An iterative thresholding algorithm for linear inverse problems with a sparsity constraint". In: *Communications on Pure and Applied Mathematics: A Journal Issued by the Courant Institute of Mathematical Sciences* 57.11 (2004), pp. 1413–1457.

[40] Paul E Debevec and Jitendra Malik. "Recovering high dynamic range radiance maps from photographs". In: *ACM SIGGRAPH 2008 classes*. 2008, pp. 1–10.

[41] Winfried Denk, James Strickler, Watt Webb, et al. "Two-photon laser scanning fluorescence microscopy". In: *Science* 248.4951 (1990), pp. 73–76.

[42] David L Donoho. "Compressed sensing". In: *IEEE Transactions on information theory* 52.4 (2006), pp. 1289–1306.

[43] Marco F Duarte et al. "Single-pixel imaging via compressive sampling". In: *IEEE signal processing magazine* 25.2 (2008), pp. 83–91.

[44] Eitan Edrei and Giuliano Scarcelli. "Memory-effect based deconvolution microscopy for super-resolution imaging through scattering media". In: *Scientific Reports* 6 (2016).

[45] Yu Fang et al. "Eye-head coordination for visual cognitive processing". In: *PloS one* 10.3 (2015).

[46] HML Faulkner and JM Rodenburg. "Movable aperture lensless transmission microscopy: a novel phase retrieval algorithm". In: *Physical Review Letters* 93.2 (2004), p. 023903.

[47] Shechao Feng et al. "Correlations and fluctuations of coherent wave transmission through disordered media". In: *Physical review letters* 61.7 (1988), p. 834.

[48] Rob Fergus, Antonio Torralba, and William T. Freeman. *Random Lens Imaging*. Tech. rep. Massachusetts Institute of Technology, Sept. 2006. URL: `http://hdl.handle.net/1721.1/33962`.

[49] Dmitriy Fradkin and David Madigan. "Experiments with random projections for machine learning". In: *Proceedings of the ninth ACM SIGKDD international conference on Knowledge discovery and data mining*. 2003, pp. 517–522.

[50] Todor Georgiev and Chintan Intwala. "Light field camera design for integral view photography". In: *Adobe Technical Report* (2006).

[51] A Georgiou et al. "Aspects of hologram calculation for video frames". In: *Journal of Optics A: Pure and Applied Optics* 10.3 (2008), p. 035302.

[52] Ralph W Gerchberg and W. 0. Saxton. "A practical algorithm for the determination of phase from image and diffraction plane pictures". In: *Optik* 35 (1972), pp. 237–246.

[53] Patrick R Gill et al. "Thermal Escher Sensors: Pixel-efficient Lensless Imagers Based on Tiled Optics". In: *Computational Optical Sensing and Imaging*. Optical Society of America. 2017, CTu3B–3.

[54] Joseph W Goodman. *Introduction to Fourier optics*. Roberts and Company Publishers, 2005.

[55] Alon Greenbaum et al. "Imaging without lenses: achievements and remaining challenges of wide-field on-chip microscopy". In: *Nature methods* 9.9 (2012), p. 889.

[56] Brian Guenter et al. "Foveated 3D graphics". In: *ACM Transactions on Graphics (TOG)* 31.6 (2012), p. 164.

[57] Walter Harm et al. "Lensless imaging through thin diffusive media". In: *Optics Express* 22.18 (2014), pp. 22146–22156.

[58] Richard Hartley and Sing Bing Kang. "Parameter-free radial distortion correction with center of distortion estimation". In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 29.8 (2007), pp. 1309–1321.

[59] Felix Heide et al. "High-quality computational imaging through simple lenses". In: *ACM Transactions on Graphics (TOG)* 32.5 (2013), pp. 1–14.

[60] Yasunobu Hitomi et al. "Video from a single coded exposure photograph using a learned over-complete dictionary". In: *2011 International Conference on Computer Vision*. IEEE. 2011, pp. 287–294.

[61] Terrence F Holekamp, Diwakar Turaga, and Timothy E Holy. "Fast three-dimensional fluorescence imaging of activity in neural populations by objective-coupled planar illumination microscopy". In: *Neuron* 57.5 (2008), pp. 661–672.

[62] Alain Hore and Djemel Ziou. "Image quality metrics: PSNR vs. SSIM". In: *2010 20th International Conference on Pattern Recognition*. IEEE. 2010, pp. 2366–2369.

[63] Ryoichi Horisaki et al. "Three-Dimensional Information Acquisition Using a Compound Imaging System". In: *Optical Review* 14.5 (2007), pp. 347–350. ISSN: 1349-9432.

[64] Jan Huisken et al. "Optical sectioning deep inside live embryos by selective plane illumination microscopy". In: *Science* 305.5686 (2004), pp. 1007–1009.

[65] Piotr Indyk and Rajeev Motwani. "Approximate nearest neighbors: towards removing the curse of dimensionality". In: *Proceedings of the thirtieth annual ACM symposium on Theory of computing*. 1998, pp. 604–613.

[66] Herbert E Ives. "Parallax panoramagrams made with a large diameter lens". In: *JOSA* 20.6 (1930), pp. 332–342.

[67] Changwon Jang et al. "Holographic Near-Eye Display with Expanded Eye-Box". In: *ACM Trans. Graph.* 37.6 (Dec. 2018). ISSN: 0730-0301. DOI: 10.1145/3272127.3275069.

[68] K.Tajima et al. "Lensless light-field imaging with multi-phased fresnel zone aperture". In: *2017 IEEE International Conference on Computational Photography (ICCP)*. May 2017, pp. 76–82.

[69] Keiichiro Kagawa et al. "A three-dimensional multifunctional compound-eye endoscopic system with extended depth of field". In: *Electronics and Communications in Japan* 95.11 (2012), pp. 14–27.

[70] Tahseen Kamal et al. "Design and fabrication of a passive droplet dispenser for portable high resolution imaging system". In: *Scientific reports* 7 (2017), p. 41482.

[71] Ulugbek Kamilov, Emrah Bostan, and Michael Unser. "Wavelet shrinkage with consistent cycle spinning generalizes total variation denoising". In: *IEEE Signal Processing Letters* 19.4 (2012), pp. 187–190.

[72] Anton S Kaplanyan et al. "DeepFovea: neural reconstruction for foveated rendering and video compression using learned statistics of natural videos". In: *ACM Transactions on Graphics (TOG)* 38.6 (2019), pp. 1–13.

[73] Yuval Kashter, A. Vijayakumar, and Joseph Rosen. "Resolving images by blurring: superresolution method with a scattering mask between the observed objects and the hologram recorder". In: *Optica* 4.8 (Aug. 2017), pp. 932–939. DOI: 10.1364/OPTICA.4.000932. URL: http://www.osapublishing.org/optica/abstract.cfm?URI=optica-4-8-932.

[74] Samuel Kaski. "Dimensionality reduction by random mapping: Fast similarity computation for clustering". In: *1998 IEEE International Joint Conference on Neural Networks Proceedings. IEEE World Congress on Computational Intelligence (Cat. No. 98CH36227)*. Vol. 1. IEEE. 1998, pp. 413–418.

[75] Ori Katz et al. "Non-invasive single-shot imaging through scattering layers and around corners via speckle correlations". In: *Nature Photonics* 8.10 (2014), pp. 784–790.

[76] Michael Kellman et al. "Motion-resolved quantitative phase imaging". In: *Biomedical Optics Express* 9.11 (2018), pp. 5456–5466.

[77] Salman Siddique Khan et al. "FlatNet: Towards Photorealistic Scene Reconstruction from Lensless Measurements". In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* (2020).

[78] Ganghun Kim et al. "Lensless photography with only an image sensor". In: *Applied optics* 56.23 (2017), pp. 6450–6456.

[79] Ganghun Kim et al. "Lensless-camera based machine learning for image classification". In: *arXiv preprint arXiv:1709.00408* (2017).

[80] Mugeon Kim et al. "Expanded Exit-Pupil Holographic Head-Mounted Display With High-Speed Digital Micromirror Device". In: *ETRI Journal* 40.3 (2018), pp. 366–375. DOI: 10.4218/etrij.2017-0166.

[81] F. Krahmer, S. Mendelson, and H. Rauhut. "Suprema of Chaos Processes and the Restricted Isometry Property". In: *Commun. Pur. Appl. Math.* 67.11 (2014), pp. 1877–1904.

[82] Grace Kuo et al. "DiffuserCam: Diffuser-Based Lensless Cameras". In: *Computational Optical Sensing and Imaging*. Optical Society of America. 2017, CTu3B–2.

[83] Grace Kuo et al. "High resolution étendue expansion for holographic displays". In: *ACM Transactions on Graphics (TOG)* 39.4 (2020), pp. 66–1.

[84] Grace Kuo et al. "On-chip fluorescence microscopy with a random microlens diffuser". In: *Optics Express* 28.6 (2020), pp. 8384–8399.

[85] Douglas Lanman and David Luebke. "Near-eye light field displays". In: *ACM Transactions on Graphics (TOG)* 32.6 (2013), pp. 1–10.

[86] KyeoReh Lee and YongKeun Park. "Exploiting the speckle-correlation scattering matrix for a compact reference-free holographic image sensor". In: *Nature Communications* 7 (2016).

[87] Anat Levin et al. "Image and depth from a conventional camera with a coded aperture". In: *ACM transactions on graphics (TOG)* 26.3 (2007), 70–es.

[88] Marc Levoy et al. "Light Field Microscopy". In: *ACM Trans. Graph. (Proc. SIGGRAPH)* 25.3 (2006).

[89] Gang Li et al. "Holographic display for see-through augmented reality using mirror-lens holographic optical element". In: *Optics letters* 41.11 (2016), pp. 2486–2489.

[90] Orly Liba et al. "Handheld mobile photography in very low light". In: *ACM Transactions on Graphics (TOG)* 38.6 (2019), pp. 1–16.

[91] Gabriel Lippmann. "Epreuves reversibles donnant la sensation du relief". In: *J. Phys. Theor. Appl.* 7.1 (1908), pp. 821–825.

[92] Fanglin Linda Liu et al. "Fourier diffuserScope: single-shot 3D Fourier light field microscopy with a diffuser". In: *Optics Express* 28.20 (2020), pp. 28969–28986.

[93] Hsiou-Yuan Liu et al. "3D imaging in volumetric scattering media using phase-space measurements". In: *Opt. Express* 23.11 (June 2015), pp. 14461–14471. DOI: 10.1364/OE.23.014461. URL: http://www.opticsexpress.org/abstract.cfm?URI=oe-23-11-14461.

[94] Antoine Liutkus et al. "Imaging with nature: Compressive imaging using a multiply scattering medium". In: *Scientific Reports* 4 (2014).

[95] Adolf W Lohmann. "Scaling laws for lens systems". In: *Applied optics* 28.23 (1989), pp. 4996–4998.

[96] David G Lowe. "Object recognition from local scale-invariant features". In: *Proceedings of the seventh IEEE international conference on computer vision*. Vol. 2. Ieee. 1999, pp. 1150–1157.

[97] Rongwen Lu et al. "Video-rate volumetric functional imaging of the brain at synaptic resolution". In: *Nature neuroscience* 20.4 (2017), pp. 620–628.

[98] Luminit. *Technical Data and Downloads*. http://www.luminitco.com/downloads/data-sheets. [Online; accessed 19-July-2008]. 2017.

[99] M. Lustig et al. "Compressed Sensing MRI". In: *IEEE Signal Processing Magazine* 25.2 (Mar. 2008), pp. 72–82. ISSN: 1053-5888. DOI: 10.1109/MSP.2007.914728.

[100] DL MacFarlane et al. "Microjet fabrication of microlens arrays". In: *IEEE Photonics Technology Letters* 6.9 (1994), pp. 1112–1114.

[101] Andrew Maimone, Andreas Georgiou, and Joel S Kollin. "Holographic near-eye displays for virtual and augmented reality". In: *ACM Transactions on Graphics (TOG)* 36.4 (2017), p. 85.

[102] Roummel F Marcia and Rebecca M Willett. "Compressive coded aperture superresolution image reconstruction". In: *2008 IEEE International Conference on Acoustics, Speech and Signal Processing*. IEEE. 2008, pp. 833–836.

[103] Kshitij Marwah et al. "Compressive light field photography using overcomplete dictionaries and optimized projections". In: *ACM Transactions on Graphics (TOG)* 32.4 (2013), p. 46.

[104] A. Matakos, S. Ramani, and J. A. Fessler. "Accelerated edge-preserving image restoration without boundary artifacts". In: *IEEE Transactions on Image Processing* 22.5 (2013), pp. 2019–2029.

[105] Kyoji Matsushima, Hagen Schimmel, and Frank Wyrowski. "Fast calculation method for optical diffraction on tilted planes by use of the angular spectrum of plane waves". In: *JOSA A* 20.9 (2003), pp. 1755–1762.

[106] Kristina Monakhova et al. "Learned reconstructions for practical mask-based lensless imaging". In: *Optics express* 27.20 (2019), pp. 28075–28090.

[107] Simon Moser, Monika Ritsch-Marte, and Gregor Thalhammer. "Model-based compensation of pixel crosstalk in liquid crystal spatial light modulators". In: *Optics express* 27.18 (2019), pp. 25046–25063.

[108] Onur Mudanyali et al. "Compact, light-weight and cost-effective microscope based on lensless incoherent holography for telemedicine applications". In: *Lab on a Chip* 10.11 (2010), pp. 1417–1428.

[109] Shree K Nayar. "Computational cameras: Redefining the image". In: *Computer* 39.8 (2006), pp. 30–38.

[110] Ren Ng et al. "Light Field Photography with a Hand-held Plenoptic Camera". In: *Stanford University Computer Science Tech Report* (Apr. 2005), pp. 3418–3421. URL: `https://graphics.stanford.edu/papers/lfcamera/lfcamera-150dpi.pdf`.

[111] Tobias Nöbauer et al. "Video rate volumetric Ca 2+ imaging across cortex using seeded iterative demixing (SID) microscopy". In: *Nature methods* 14.8 (2017), p. 811.

[112] J. Nocedal and S. J. Wright. *Numerical Optimization*. Springer, 2006.

[113] Rudolf Oldenbourg. "A new view on polarization microscopy". In: *Nature* 381.6585 (1996), pp. 811–812.

[114] Frank Ong and Michael Lustig. "Beyond low rank+ sparse: Multiscale low rank matrix decomposition". In: *IEEE journal of selected topics in signal processing* 10.4 (2016), pp. 672–687.

[115] Efthymios P Papageorgiou et al. "Angle-insensitive amorphous silicon optical filter for fluorescence contact imaging". In: *Optics letters* 43.3 (2018), pp. 354–357.

[116] Jongchan Park, KyeoReh Lee, and YongKeun Park. "Ultrathin wide-angle large-area digital 3D holographic display using a non-periodic photon sieve". In: *Nature communications* 10.1 (2019), p. 1304.

[117] Anjul Patney et al. "Towards foveated rendering for gaze-tracked virtual reality". In: *ACM Transactions on Graphics (TOG)* 35.6 (2016), p. 179.

[118] Sri Rama Prasanna Pavani and Rafael Piestun. "Three dimensional tracking of fluorescent microparticles using a photon-limited double-helix response system". In: *Optics Express* 16.26 (2008), pp. 22048–22057.

[119] Nicolas C Pégard et al. "Compressive light-field microscopy for 3D neural activity recording". In: *Optica* 3.5 (2016), pp. 517–524.

[120] Yifan Peng et al. "Neural holography". In: *ACM SIGGRAPH 2020 Emerging Technologies*. 2020, pp. 1–2.

[121] Zachary F Phillips, Michael Chen, and Laura Waller. "Single-shot quantitative phase microscopy with color-multiplexed differential phase contrast (cDPC)". In: *PloS one* 12.2 (2017), e0171228.

[122] Eftychios A Pnevmatikakis et al. "Simultaneous denoising, deconvolution, and demixing of calcium imaging data". In: *Neuron* 89.2 (2016), pp. 285–299.

[123] SM Popoff et al. "Measuring the transmission matrix in optics: an approach to the study and control of light propagation in disordered media". In: *Physical review letters* 104.10 (2010), p. 100601.

[124] Jonathan Ragan-Kelley et al. "Halide: a language and compiler for optimizing parallelism, locality, and recomputation in image processing pipelines". In: *ACM SIGPLAN Notices* 48.6 (2013), pp. 519–530.

[125] Ramesh Raskar, Amit Agrawal, and Jack Tumblin. "Coded exposure photography: motion deblurring using fluttered shutter". In: *ACM SIGGRAPH 2006 Papers*. 2006, pp. 795–804.

[126] Justin Romberg. "Compressive sensing by random convolution". In: *SIAM Journal on Imaging Sciences* 2.4 (2009), pp. 1098–1128.

[127] Leonid I Rudin, Stanley Osher, and Emad Fatemi. "Nonlinear total variation based noise removal algorithms". In: *Physica D: Nonlinear Phenomena* 60.1-4 (1992), pp. 259–268.

[128] Peter Rupprecht et al. "Remote z-scanning with a macroscopic voice coil motor for fast 3D multiphoton laser scanning microscopy". In: *Biomedical optics express* 7.5 (2016), pp. 1656–1671.

[129] Michael J Rust, Mark Bates, and Xiaowei Zhuang. "Sub-diffraction-limit imaging by stochastic optical reconstruction microscopy (STORM)". In: *Nature methods* 3.10 (2006), pp. 793–796.

[130] Kiyotaka Sasagawa et al. "Highly sensitive lens-free fluorescence imaging device enabled by a complementary combination of interference and absorption filters". In: *Biomedical optics express* 9.9 (2018), pp. 4329–4344.

[131] Liang Shi et al. "Near-eye light field holographic rendering with spherical waves for wide field of view interactive 3D computer graphics". In: *ACM Transactions on Graphics (TOG)* 36.6 (2017), p. 236.

[132] Alok Singh et al. "Exploiting scattering media for exploring 3D objects". In: *Light: Science & Applications* 6.2 (2017), e16219.

[133] Alok Singh et al. "Looking through a diffuser and around an opaque surface: A holographic approach". In: *Optics Express* 22.7 (2014), pp. 7694–7701.

[134] Alok Singh et al. "Scatter-plate microscope for lensless microscopy with diffraction limited resolution". In: *Scientific Reports* 7.1 (2017), p. 10687.

[135] Ayan Sinha et al. "Lensless computational imaging through deep learning". In: *Optica* 4.9 (Sept. 2017), pp. 1117–1125. DOI: 10.1364/OPTICA.4.001117. URL: http://www.osapublishing.org/optica/abstract.cfm?URI=optica-4-9-1117.

[136] Vincent Sitzmann et al. "End-to-end optimization of optics and image processing for achromatic extended depth of field and super-resolution imaging". In: *ACM Transactions on Graphics (TOG)* 37.4 (2018), pp. 1–13.

[137] Chris Slinger, Colin Cameron, and Maurice Stanley. "Computer-generated holography as a generic display technology". In: *Computer* 38.8 (2005), pp. 46–53.

[138] Filip Sroubek and Peyman Milanfar. "Robust multichannel blind deconvolution via fast alternating minimization". In: *IEEE Transactions on Image processing* 21.4 (2011), pp. 1687–1700.

[139] Rod Sterling. "JVC D-ILA high resolution, high contrast projectors and applications". In: *Proceedings of the 2008 workshop on Immersive projection technologies/Emerging display technologiges*. ACM. 2008, p. 10.

[140] David G Stork and Patrick R Gill. "Optical, mathematical, and computational foundations of lensless ultra-miniature diffractive imagers and sensors". In: *International Journal on Advances in Systems and Measurements* 7.3 (2014), p. 4.

[141] Christoph Stosiek et al. "In vivo two-photon calcium imaging of neuronal networks". In: *Proceedings of the National Academy of Sciences* 100.12 (2003), pp. 7319–7324.

[142] Abigail Stylianou and Robert Pless. "SparkleGeometry: Glitter Imaging for 3D Point Tracking". In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*. 2016, pp. 10–17.

[143] Shiro Suyama, Munekazu Date, and Hideaki Takada. "Three-dimensional display system with dual-frequency liquid-crystal varifocal lens". In: *Japanese Journal of Applied Physics* 39.2R (2000), p. 480.

[144] Jun Tanida et al. "Thin observation module by bound optics: concept and experimental verification". In: *Applied Optics* 40.11 (2001), pp. 1806–1813.

[145] Xiaodong Tao et al. "High-speed scanning interferometric focusing by fast measurement of binary transmission matrix for channel demixing". In: *Optics express* 23.11 (2015), pp. 14168–14187.

[146] Nobukazu Teranishi et al. "Evolution of optical structure in image sensors". In: *2012 International Electron Devices Meeting*. IEEE. 2012, pp. 24–1.

[147] Lei Tian, Jingyan Wang, and Laura Waller. "3D differential phase-contrast microscopy with computational illumination using an LED array". In: *Optics letters* 39.5 (2014), pp. 1326–1329.

[148] Lei Tian et al. "Multiplexed coded illumination for Fourier Ptychography with an LED array microscope". In: *Biomedical optics express* 5.7 (2014), pp. 2376–2389.

[149] Lin Tian et al. "Imaging neural activity in worms, flies and mice with improved GCaMP calcium indicators". In: *Nature methods* 6.12 (2009), pp. 875–881.

[150] Ivo M Vellekoop, Aart Lagendijk, and AP Mosk. "Exploiting disorder for perfect focusing". In: *Nature photonics* 4.5 (2010), p. 320.

[151] Ivo M Vellekoop and AP Mosk. "Focusing coherent light through opaque strongly scattering media". In: *Optics letters* 32.16 (2007), pp. 2309–2311.

[152] Kartik Venkataraman et al. "PiCam: An ultra-thin high performance monolithic camera array". In: *ACM Transactions on Graphics (TOG)* 32.6 (2013), p. 166.

[153] Curtis R Vogel and Mary E Oman. "Iterative methods for total variation denoising". In: *SIAM Journal on Scientific Computing* 17.1 (1996), pp. 227–238.

[154] Neal Wadhwa et al. "Synthetic depth-of-field with a single-camera mobile phone". In: *ACM Transactions on Graphics (TOG)* 37.4 (2018), pp. 1–13.

[155] Koki Wakunami et al. "Projection-type see-through holographic three-dimensional display". In: *Nature communications* 7.1 (2016), pp. 1–7.

[156] Y. Wang et al. "A New Alternating Minimization Algorithm for Total Variation Image Reconstruction". In: *SIAM Journal on Imaging Sciences* 1.3 (2008), pp. 248–272.

[157] Gordon Wetzstein et al. "Tensor displays: compressive light field synthesis using multilayer displays with directional backlighting". In: (2012).

[158] Bennett S Wilburn et al. "Light field video camera". In: *Media Processors 2002*. Vol. 4674. International Society for Optics and Photonics. 2001, pp. 29–36.

[159] Xiaodong Xun and Robert W Cohn. "Phase calibration of spatially nonuniform spatial light modulators". In: *Applied optics* 43.35 (2004), pp. 6400–6406.

[160] Kyrollos Yanny et al. "Miniscope3D: optimized single-shot miniature 3D fluorescence microscopy". In: *Light: Science & Applications* 9.1 (2020), pp. 1–13.

[161] Fahri Yaraş, Hoonjong Kang, and Levent Onural. "Real-time phase-only color holographic video display system using LED illumination". In: *Applied optics* 48.34 (2009), H48–H53.

[162] Li-Hao Yeh et al. "Speckle-structured illumination for 3D phase and fluorescence computational microscopy". In: *Biomedical Optics Express* 10.7 (2019), pp. 3635–3653.

[163] Jonghee Yoon et al. “Measuring optical transmission matrices by wavefront shaping”. In: *Optics Express* 23.8 (2015), pp. 10158–10167.

[164] HyeonSeung Yu, KyeoReh Lee, and YongKeun Park. “Ultrahigh enhancement of light focusing through disordered media controlled by mega-pixel modes”. In: *Optics express* 25.7 (2017), pp. 8036–8047.

[165] Hyeonseung Yu et al. “Ultrahigh-definition dynamic 3D holographic display by active control of volume speckle fields”. In: *Nature Photonics* 11.3 (2017), p. 186.

[166] Jingzhao Zhang et al. “3D computer-generated holography by non-convex optimization”. In: *Optica* 4.10 (2017), pp. 1306–1313.

[167] Guoan Zheng et al. “The ePetri dish, an on-chip cell imaging platform based on subpixel perspective sweeping microscopy (SPSM)”. In: *Proceedings of the National Academy of Sciences* 108.41 (2011), pp. 16889–16894.