3D Lensless Imaging – Theory, Hardware, and Algorithms

Submitted in partial fulfillment of the requirements for

the degree of

Doctor of Philosophy

in

Department of Electrical and Computer Engineering

Yi Hua

B.S., Computer Science, Rice University

M.S., Computer Vision, Carnegie Mellon University

Carnegie Mellon University

Pittsburgh, PA

January 2023

Acknowledgments

First and foremost, I would like to thank my advisor Aswin Sankaranarayanan for all his help and advice with this PhD. I have gained an appreciation and confidence in getting a deep understanding of systems, finding their limitations and using the insight to improve them.

I would like to thank my boyfriend, Jen-Hao Rick Chang, for the constant feedback and support. I would also like to thank my lab mates and classmates who made my PhD life memorable: Vishwanath Saragadam, Chia-Yin Tsai, Harry Hui, Jian Wang, Byeongjoo Ahn, Anqi Yang, Michael De Zeeuw, Wei-Yu Chen, Leron Julian, Tyler Nuanes, Natalie Janosik, Yingsi Qin, Haejoon Lee, Kuldeep Kulkarni, Vijay Rengarajan, Shumian Xin, Alankar Kotwal, Satwik Kottur, and Chaojing Duan. In addition to those in Pittsburgh, I would also like to thank people who have given me advice, encouragement, or fellowship on my journey into research: Adithya Pediredla, Jason Holloway, Adam Samaniego, Qiong Huang, Xuaner Zhang, Vivek Boominathan, Huaijin George Chen, Beidi Chen, and Yan Ai among others.

I would like to thank my academic mentors for showing me what is good research and what makes a good researcher: Ioannis Gkioulekas, Simon Lucey, Ashok Veeraraghavan, Ashu Sabharwal and Luay Nakhleh. I would like to additionally thank my thesis committee for their questions and comments that improved this thesis: Matthew O'Toole, Laura Waller, and Soummya Kar. I would like to thank my collaborators for the illuminating discussions: Salman Asif, Yucheng Zheng, Shigeki Nakamura, Yash Belhe, Hossein Baktash, Matteo Giuseppe Scopelliti, and Maysamreza Chamanzar. I would like to thank my internship hosts at Google for a wonderful summer working on a product that feels like science fiction: Supreeth Achar, Harris Nover, Dan Goldman. I would like to thank the many brilliant researchers in computational photography and imaging; your well-written and creative papers have brought me much joy.

The dissertation and my PhD work were supported by a Dean's fellowship, a Sony research contract, and the National Science Foundation (grant numbers 1618823, 1652569, 1730147). I am grateful for their support in enabling my doctoral work.

Finally, I would like to thank my parents, Lingli Li and Xianming Hua. They have unequivocally supported me in exploring my many interests. This thesis is a direct result of that support.

Abstract

Lensless cameras enable us to see in many challenging scenarios. They can image in narrow spaces where lenses do not fit, operate at wavelengths where lenses do not work, and make ultra-wide field-of-view microscopes. Despite their novel capabilities, current lensless cameras have limited imaging quality that restricts their practicality. These limitations can be attributed to the conditioning and complexity of the inverse problem that lensless imagers must solve to obtain the scene.

A common design in lensless imaging is that of a thin attenuating mask placed before a sensor. For a scene restricted to a front-parallel plane, the image formation model can be approximated as a 2D convolution between the plane's texture and a scaled version of the mask pattern, and the ensuing inverse problem has efficient solutions. However, scenes of more complex geometry, such as those spanning a large depth range, pose a difficult and under-determined inverse problem. This thesis aims to develop lensless imaging techniques to effectively and efficiently photograph 3D scenes with an extended depth range. To that end, we make the following contributions to the theory, hardware, and algorithms of 3D lensless imaging.

First, we present a theoretical analysis of the spatial and axial resolution limits of a mask-based lensless camera, which provides an understanding of the performance of various camera designs. Specifically, we derive the closed-form expression of a 3D modulation transfer function as a function of the mask pattern, and connect the parameters of the mask to the camera's achievable spatio-axial resolution.

Second, we introduce programmable masks in lensless imagers to increase the number of measurements by capturing multiple frames while displaying different mask patterns. This upgrade in hardware allows computational focusing at a given depth, such that the resulting measurements are well approximated as a result of 2D convolution, even when the scene extends over a large depth range. As a result, the texture corresponding to a specific depth can be recovered with an efficient deconvolution method with fewer artifacts.

Finally, we present an inverse rendering approach to the reconstruction problem, which requires a joint solution of the texture and shape of the scene. This approach solves the inverse problem under a physically realistic and differentiable forward model. It allows us to faithfully represent scenes as surfaces instead of volumetric albedo functions as is commonly used in previous works, and avoids reconstruction artifacts arising from model mismatch.

Together, those three contributions provide a fundamental advance to 3D lensless imaging.

Contents

List of Figures

List of Tables

1 Introduction

If you want to see a better world, change the lens through which you see it ...

— Jeffrey G. Duarte

When we try to expand the capabilities of modern cameras, we often find that the lens is the limiting factor. It occupies volume and weight that prevents miniaturization. Its rigidity prevents flexible camera designs. It operates on a limited range of wavelengths. And it requires a manufacturing process that is separate from the sensor. All these constraints raise the question: Can we image without a lens?

The short answer is yes. The concept of lensless imaging dates back thousands of years to the camera obscura, where a small pinhole in a wall projects an image of the outside scene into a dark room. About a century ago, X-ray and gamma-ray imaging techniques were developed for medical applications; computing became an integral part of imaging and produced images from tomographic measurements [Kak and Slaney, 2001]. In the past decade, many researchers leveraged the advancements in computing power to produce lightweight and light-efficient lensless cameras that challenge the limits of imaging [Boominathan *et al.*, 2016]. Unlike lens-based designs, these imagers record a very blurred version of the scene that is unrecognizable to the human eye. However, the measurements preserve information that can be inverted with computation. Recent lensless imagers have proven many advantages, including thin form factor [Adams *et al.*, 2017, Asif *et al.*, 2016, Boominathan *et al.*, 2020, Yamaguchi *et al.*, 2019], wide field-of-view microscopic imaging [Adams *et al.*, 2017], 3D imaging [Adams *et al.*, 2017, Antipa *et al.*, 2018, Boominathan *et al.*, 2020, Hua *et al.*, 2020, Zheng and Asif, 2020, Zheng *et al.*, 2021], hyperspectral imaging [Monakhova *et al.*, 2020], and video from a single frame [Antipa *et al.*, 2019].

While lensless imagers have shown great promise at imaging in diverse scenarios, their current imaging quality is not comparable to that of a traditional lens-based camera, because the reconstruction problem is rather challenging, as we explain in the following sections.

1.1 Why image without lenses?

Lenses conveniently focus an image of a scene onto a sensor, so why design cameras without lenses? Replacing the lens with a thin, light-modulating element brings numerous advantages, both in camera form-factors and its imaging abilities. This section illustrates many benefits of lensless imaging with recent examples; a detailed survey of the benefits of lensless imaging can be found in Boominathan *et al.* [2016, 2022].

Thin form-factor. Miniature cameras are useful for endoscopy and mobile devices that benefit from thin size. Since lensless imagers do not require focusing the scene on the sensor, they allow thin light-modulating elements, such as amplitude masks or gratings, to be placed very closely or directly manufactured on top of the sensor. PicoCam [Gill and Stork, 2013]'s experimental hardware has only a 0.45mm thick grating on top of a sensor and expects the design to produce ∼100um ultra-miniature imagers. FlatCam [Asif *et al.*, 2016] and FlatScope [Adams *et al.*, 2017] are amplitude mask-based designs with prototypes that measure less than 0.6mm thick and weigh 0.2g. PhlatCam [Boominathan *et al.*, 2020] is a more light-efficient phase mask-based design, producing a prototype that is 2mm thick.

Wide field-of-view microscopy. In a lens-based microscope, increasing the magnification trades off the field of view and the depth of field. Lensless cameras can resolve fine details on objects that are very close to them without requiring magnification, and therefore makes wide field of view and depth of field microscopes. FlatScope [Adams *et al.*, 2017] is a lensless microscope with 2 µm lateral resolution and less than 15 µm axial (or depth) resolution, and ultrawide field of view that is 10 times larger than a lens-based microscope.

3D and light field imaging. Since lensless imagers have depth-dependent PSF, they encode the scene's depth information in their measurements. FlatScope [Adams *et al.*, 2017], DiffuserCam [Antipa *et al.*, 2018], PhlatCam [Boominathan *et al.*, 2020], Fresnel zone aperture based imager [Shimano *et al.*, 2018, Yamaguchi *et al.*, 2019] show 3D reconstructions or refocused images from light field from measurements captured on their lensless imager prototype. However, it is difficult to analyze the depth resolution of each imager.

Hyperspectral imaging. Lensless imagers with large and random PSF implements random linear projections of the scene. Thus compressive sensing techniques can reconstruct structured high dimensional signals, such as hyperspectral images, from the lensless measurements. Spectral DiffuserCam

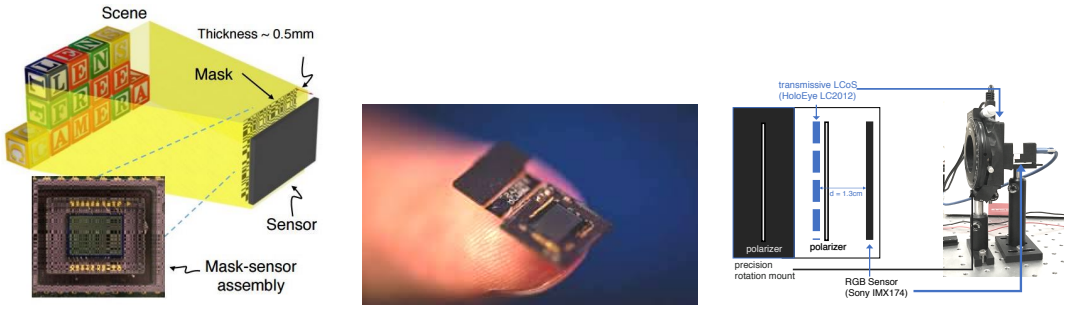


Figure 1.1: **Amplitude mask-based lensless camera prototypes.** From left to right: the prototypes for FlatCam [Asif *et al.*, 2016], FlatScope [Adams *et al.*, 2017], and programmable mask prototype in SweepCam [Hua *et al.*, 2020].

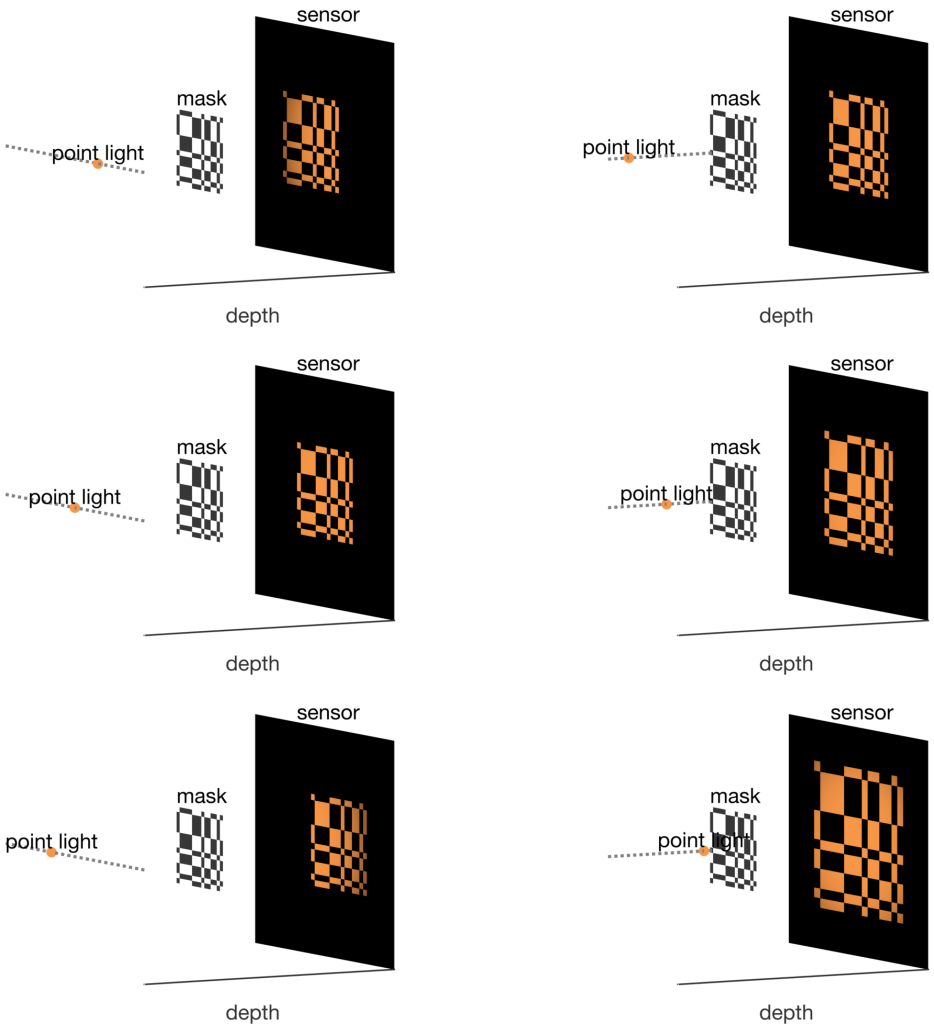
uses a diffuser to create large and random PSF, and produces a hyperspectral imaging prototype with 64 color channels and 48 µm multi-point spatial resolution [Monakhova *et al.*, 2020].

High frame-rate video. Similarly, video is a structured high-dimensional signal that can be reconstructed from lensless measurements. Recording compressive measurements instead of the original signal shortens the time for transmitting and saving measurements, increasing the frame rate. An extreme case of this is a prototype with DiffuserCam design and a rolling shutter sensor that uses each row to recover a video frame [Antipa *et al.*, 2019]. It records video at 4500 frames per second.

1.2 Amplitude mask-based lensless imagers

Amplitude mask-based lensless imagers consists of a thin amplitude mask that attenuates light with pattern placed a short distance away from the sensor. They record a blurry image of the scene, where the blur is designed to preserve information so that a clear image of the scene can be computationally reconstructed from the blurry measurements.

Prototypes. Some examples of lensless camera prototypes are shown in Figure 1.1. Building an amplitude-mask lensless camera is simple since the fabrication of the mask is easy. The amplitude mask can be a mask printed on glass, with the smallest feature size around a few micrometers. In the future, it maybe possible to combine the manufacturing of the sensor with the mask in the same CMOS process. This thesis introduces a programmable mask, similar to the programmable spatial light modulators used in displays.



(a) translation in sensor plane (b) translation orthogonal to sensor plane

Figure 1.2: **Measurement translates as the point source translates in sensor plane, and scales as the point source translates orthogonal to sensor plane.**

Measurements. If a point light is placed in front of the lensless camera, it will cast a shadow of the mask on the sensor, as shown in Figure 1.2. A closer look reveals that if you move the light source parallel to the sensor, the shadow will translate in the opposite direction. With this observation, we model the lensless measurements of a single-depth scene as a convolution between the scene albedo with the shadow of the mask, *i.e.* the point spread function (PSF) of the system. This convolution model allows us to analyze the imaging resolution of lensless cameras and design good mask patterns that preserve scene information in measurements. However, if you move the light source orthogonal to the sensor, the shadow will scale. The fact that the PSF changes as a function of the point's distance to the sensor, *i.e.* the point's depth, complicates the reconstruction problem. Specifically, this means we cannot predict the lensless cameras' imaging resolution of 3D scenes, and the reconstruction of 3D scenes from lensless measurements remains a challenging problem. This thesis addresses this problem of imaging 3D scenes with lensless cameras, by understanding the role of depth in lensless imaging.

1.3 Resolving depth in lensless imaging

Resolving a scene from lensless measurements requires separating the scene content from their PSFs. For a thin mask-based lensless imager, scene points at the same depth have PSFs that are translated copies of each other; scene points at different depths result in different PSFs that are scaled copies of each other. Therefore resolving the scene requires an estimate of the depth of scene points so that the correct PSF can be used in the reconstruction. This is the central problem addressed by this thesis.

Resolving depth from a lensless measurement is a challenging problem for several reasons. First, it is an under-determined problem. This means it is nearly impossible to distinguish the actual scene from an infinitude of other solutions. Additionally, it is a non-convex problem, so it is difficult to navigate the solution space to find the true solution.

While the actual forward model that maps the scene to lensless measurements is complex, previous research places assumptions on the scene and the imager to obtain simplified approximations that can be inverted. There are two kinds of simplified representations of the scene. The first approach assumes the scene consists of a 2D array of point light sources, and the inverse problem solves each point's brightness and depth (Section 2.3.2). This approach tries to solve the difficult inverse problem by alternating between estimating the brightness and depth on each iteration, but requires good initialization to succeed. The second approach avoids estimating the depth of scene points by representing the whole scene as a volume, typically consisting of multiple transparent depth planes, so the inverse problem solves only the brightness or color at every point in the 3D volume (Section 2.3.3). The downside to representing

the scene as a 3D volume is that typical 3D scenes need to be represented by a large number of depth planes, and the reconstruction problem requires solving a 3D unknown from a 2D measurement, which makes it severely underdetermined.

The two approaches share some common limitations. Both solve under-determined systems which makes it difficult to evaluate the lateral and axial resolution of the imager. Both require iterative methods for solving the 3D scenes, which makes the reconstruction time-consuming. Both place restrictive assumptions on the scene to obtain a simple forward model. This thesis addresses each of the problems with proposed imaging methods and reconstruction algorithms in the following section.

1.4 Thesis Contributions

This thesis makes the following contribution to improve 3D lensless imaging:

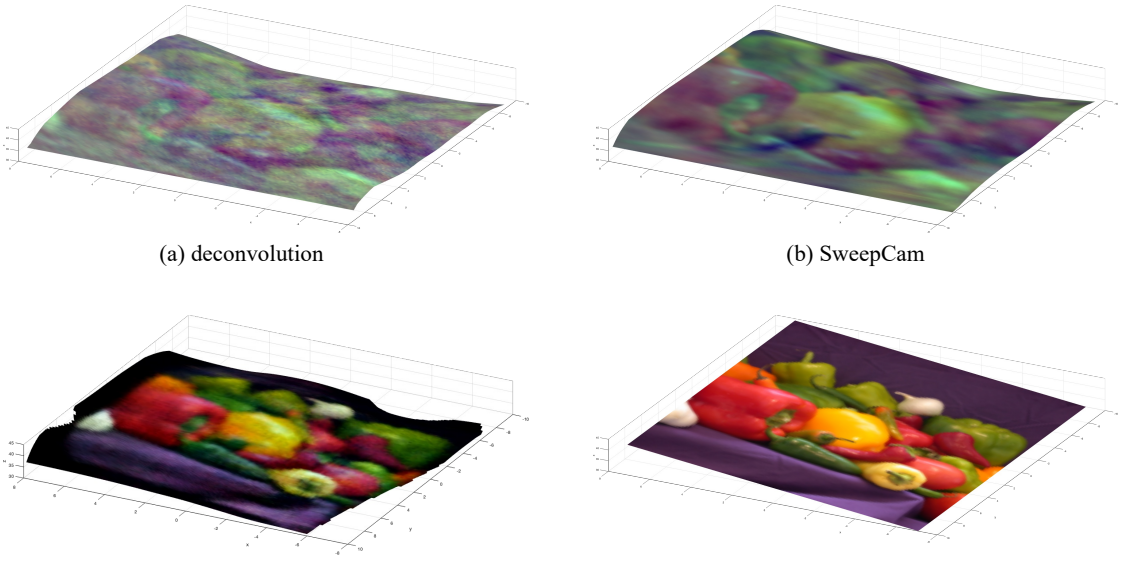
- **Spatial and depth resolution analysis (Chapter 3, [Hua *et al.*, 2023]).** The key question this thesis tackles is, what is the resolution of a lensless camera, when it images a 3D scene? Previous studies only answered this question with empirical observations. This is because the measurement model underlying prior lensless imagers lacks special structures that facilitate deeper analysis. This work provides a theoretical framework for studying the achievable spatial-axial resolution of lensless cameras. We obtain a theoretical limit on the spatial and axial resolving power of the camera, in the form of 3D modulation transfer function (MTF), as a function of the mask parameters. This framework also allows us to reason about the general performance of amplitude mask-based lensless cameras, by deriving the decay rate for MTFs.

- **Depth-aware reconstruction with programmable masks (Chapter 4, [Hua *et al.*, 2020]).** Previous works reconstruct a 3D scene from a single 2D lensless measurement. This inverse problem is under-determined and hence difficult to solve. To mitigate this, we obtain more measurements by introducing programmable masks to enable imaging multiple frames with different mask patterns. However, naively solving the problem of reconstruction from a large number of measurements is still time and memory prohibitive, as the measurement matrix for such system is large, containing more than 10^{12} entries for a camera sensing with a moderate resolution of 150×240 pixels and 32 mask patterns. We present an efficient method for imaging a single depth with few artifacts, by imaging with many translated versions of the same mask pattern, which allows us to computationally "focus" on a limited depth range of the scene, resulting in reconstruction that is two orders of magnitude faster.

- **Inverse rendering for joint reconstruction of texture and depth from lensless measurements (Chapter 5).** One of the critical ingredients in solving the inverse problems in computational imaging is a forward model that balances the fidelity to image formation against the computational efficiency of the ensuing inverse technique. This careful balancing act between precision and complexity can be seen in many problems. For example, early work in photometric stereo made assumptions in the form of convex objects with Lambertian reflectance and distant lighting [Woodham, 1980], which permits a simple solution for the shape of the object. Over time these constraints have been progressively relaxed to incorporate inter-reflections, specularities, and uncalibrated, near-field lighting [Ackermann and Goesele, 2015]. A more recent example can be found in non-line-of-sight imaging, where there is a clear progression from diffuse volumetric scene representations [Velten *et al.*, 2012], that permit solutions in the form linear inverses, to complex forward modeling using wave [Liu *et al.*, 2020] and geometric optics [Tsai *et al.*, 2019]. While previous methods simplify the forward model to make the inverse problem feasible, the model mismatches resulted in artifacts in the reconstruction result. We introduce ideas from differential rendering to solve the inverse problem with a physically realistic forward model using stochastic gradient descent.

We show an example of how the contributions from this thesis improve the 3D imaging quality of a lensless camera in Figure 1.3.

The rest of the document is structured in the following way: Chapter 2 provides the mathematical forward model under ray optics, examples of the challenges in depth reconstruction from lensless measurements, and finally previous 3D reconstruction methods. Chapters 3-5 are dedicated to the three contributions. Chapter 6 concludes the proposed ideas as well as provide outlook on the future of 3D lensless imaging.



(a) deconvolution

(b) SweepCam

(c) surface-based inverse rendering

(d) ground truth

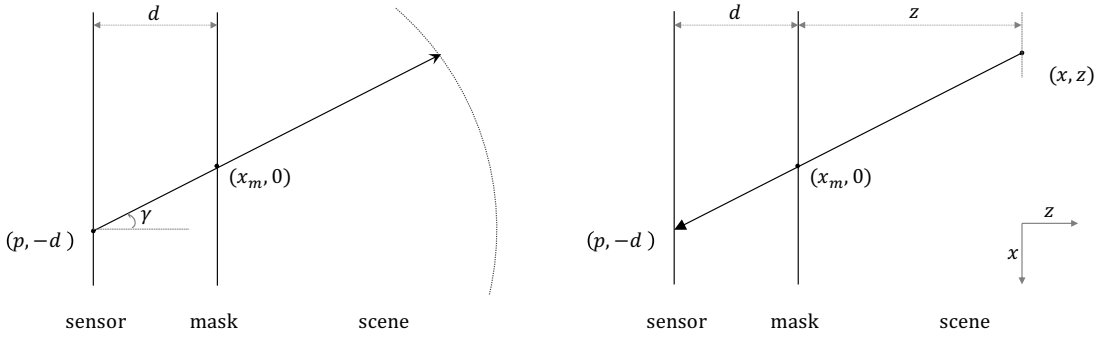
Figure 1.3: **Improvements to imaging 3D scenes with lensless cameras.** A plane textured with a image of peppers is placed at an angle to the camera as shown in (d). (a) The 3D scene inferred from 2D devolution of the measurement. (b) The 3D scene inferred from our proposed programmable mask camera, named SweepCam. (c) The 3D scene as inferred by our proposed surface-based inverse rendering algorithm.

2 3D Lensless Imaging

This chapter develops the mathematical forward models and introduces prior reconstruction techniques for 3D lensless imaging.

2.1 Forward model under ray optics

To understand the lensless measurements of 3D scenes, we first derive the mathematical expression of lensless measurements as a function of the 3D scene, under ray optics.



(a) spherical coordinates

(b) Cartesian coordinates

Figure 2.1: **Geometry of mask-based lensless imager illustrated in 2D.** Light ray from (x, z) in scene reaches sensor location (p, d) via mask location $(x_m, 0)$.

Let the mask have attenuation function $m(\cdot)$ and mask-to-sensor distance d. As shown in Figure 2.1(a), the ray passing $(\mathbf{p}, -d)$ in the direction of $-\boldsymbol{\gamma}$ has radiance $L(\mathbf{p}, -\boldsymbol{\gamma})$ before it reaches the imager and is attenuated by mask $m(\mathbf{x}_m)$. The sensor measurement for pixel centered at $\mathbf{p}$, $i(\mathbf{p})$, integrates radiance of rays from all directions reaching the pixel area, modulated by the angular efficiency of pixel

$a(\cdot)$, which decreases at large angles.

$$i(\mathbf{p}) = \int_{\substack{\widehat{\mathbf{p}} \text{ in} \\ \text{pixel}}} \int_{\boldsymbol{\gamma} \in \mathbb{S}^2} L(\widehat{\mathbf{p}}, -\boldsymbol{\gamma}) m(\mathbf{x}_m) a(\boldsymbol{\gamma}) d\boldsymbol{\gamma} d\widehat{\mathbf{p}} \tag{2.1}$$

We proceed to simplify the model with the common assumptions shared in previous research. First, we assume each point in the scene emits light uniformly in all directions, and the scene can be represented as a volumetric albedo function $\rho(\mathbf{x}, z)$. Next, we assume that the pixel area is small, and it is in current commercial sensors, $L(\mathbf{p}, -\gamma)$ is approximately constant for $\mathbf{p}$ in pixel with area $\Delta_{\mathbf{p}}$. Finally, we note that the pixel angular efficiency $a(\cdot)$ and solid angle $\omega(\cdot)$ are both functions of ray angle. We can consider those effects together as effective pixel angular response, $\widetilde{a}(\theta) = a(\theta)\omega(\theta)\Delta_p$. Then the sensor measurement of pixel centered at $\mathbf{p}$ can be obtained from

$$i(\mathbf{p}) = \int_z \int_{\mathbf{x}} \rho(\mathbf{x}, z) m(\mathbf{x}_m) \widetilde{a}\left(\frac{\mathbf{x} - \mathbf{p}}{z + d}\right) d\mathbf{x} dz \tag{2.2}$$

2.1.1 Sum of Convolution Model

By placing the additional assumption of small incident ray angles so that the effective angular response is constant over the field of view, *i.e.* $\widetilde{a}\left(\frac{\mathbf{x}-\mathbf{p}}{z+d}\right) = \widetilde{a}_0$, the forward model of a 3D scene becomes a sum of convolutions. This model is commonly assumed in lensless imaging literature [Antipa *et al.*, 2018, Asif, 2018, Boominathan *et al.*, 2020, Hua *et al.*, 2020, Yamaguchi *et al.*, 2019]. It is a useful model that allows simple calibration and reconstruction.

For discrete depth planes, a re-parameterization $\widetilde{\mathbf{x}} = -\frac{d}{z}\mathbf{x}$ of Eq. (5.2) allows the sensor measurement to be written as a result of sum of convolutions,

$$i(\mathbf{p}) = \sum_z \int_{\widetilde{\mathbf{x}}} \widetilde{\rho}_z(\widetilde{\mathbf{x}}) \widetilde{m}(\mathbf{p} - \widetilde{\mathbf{x}}) d\widetilde{\mathbf{x}} = \sum_z (\widetilde{\rho}_z *_{1D} \widetilde{m}_z)(\mathbf{p}), \tag{2.3}$$

where re-parameterized scene $\widetilde{\rho}_z(\widetilde{\mathbf{x}}) = \widetilde{a}_z(\frac{z}{d})^2 \rho(-\frac{z}{d}\widetilde{\mathbf{x}}, z)$ is convolved with a kernel which is a scaled version of the mask modulation pattern $\widetilde{m}_z(\widetilde{\mathbf{x}}) = m(\frac{z}{z+d}\widetilde{\mathbf{x}})$.

Equation (2.3) can be expressed in frequency as a sum of multiplications,

$$I(\boldsymbol{\omega}) = \sum_z \widetilde{\mathrm{P}}_z(\boldsymbol{\omega})\, \widetilde{M}_z(\boldsymbol{\omega}), \tag{2.4}$$

where $I, \widetilde{\mathrm{P}}_z, \widetilde{M}_z$ are the Fourier transform of $i, \widetilde{\rho}_z, \widetilde{m}_z$ respectively.

Note that finite sensor area result in cropped measurements, and the cropping is modeled in Antipa *et al.* [2018] and Boominathan *et al.* [2020].

2.2 Challenges in reconstructing depth from lensless measurements

It is challenging to resolve the depth of the scene from lensless measurements for a number of reasons:

1. the function that maps scene to lensless measurements is surjective;
2. the optimization problem is non-convex.

This section shows examples that illustrate those two problems.

Example of depth ambiguity. Under the sum of convolution model, it is easy to produce an example of two scenes, each consisting of one frontal-parallel plane, at different depth, that result in the same measurements. Consider a lensless imager, whose PSFs are $p_1(\mathbf{x})$, $p_2(\mathbf{x})$ for points at two different depths z_1, z_2 respectively. Consider a scene consisting of a plane with texture $p_2(\mathbf{x})$ at depth z_1, whose measurement is $\propto (p_1 * p_2)(\mathbf{x})$. Consider another scene consisting of a plane with texture $p_1(\mathbf{x})$ at depth z_2, whose measurement is $\propto (p_1 * p_2)(\mathbf{x})$. The two scenes consisting of single planes at different depth result in the same measurements. We can capture more measurements with different masks to break this ambiguity, and we present a method for depth-aware reconstruction with programmable masks in Chapter 4.

Example of non-convex inverse problem. Figure 2.2 shows an example of mean squared error of sensor measurements corresponding to changes in texture and depth of a point on surface. This is the data term commonly used in the inverse optimization problem. As shown, the function is not convex. In this example, there is a local minimum corresponding to a closer but darker point. Besides using prior terms that enforce sparsity or continuity of the scene, obtaining multiple measurements with different masks also helps disambiguate the depth of scene points.

2.3 Prior work

The inverse problem of 3D reconstruction of scene $\mathbf{s}$ from lensless measurements $\mathbf{i}$ with forward model Ψ is posed as an optimization problem that minimizes the differences between captured and rendered measurements plus some regularization term on the scene,

$$\arg\min_{\mathbf{s}} \|\mathbf{i} - \Psi(\mathbf{s})\|_2^2 + \lambda \,\text{prior}(\mathbf{s}). \tag{2.5}$$

Different lensless imager designs lead to different simplification of the forward model, resulting in different reconstruction problems. This section summarizes the previous research in 3D reconstruction from lensless measurements, detailing the forward models, assumptions used for simplifcation, and runtime.

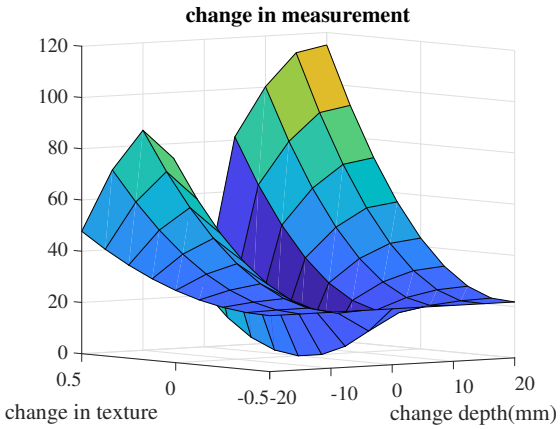


Figure 2.2: **MSE of sensor measurements with texture and depth change.** Measurements are simulated for SweepCam prototype with 8×8 binned pixel and mask pattern from calibration. A small plane with depth 26.9mm and 0.01 square degree is perturbed in texture and depth.

2.3.1 Reconstruction of a single depth plane

When the scene consists of a single plane of known depth, or is far away enough that the PSF is the same for all scene points, resolving the scene from lensless measurements is simple and runs in real time.

Sum of convolution model. Under the sum-of-convolution model as described in Section 2.1.1 , reconstructing a single plane of known depth is a deconvolution problem.

Optimization objective. For a single plane of known depth, its texture can be recovered under the sum of convolution model equation (2.3) by solving

$$\arg\min_{\boldsymbol{\rho}} \|\mathbf{i} - \widetilde{\boldsymbol{\rho}} *_{1D} \widetilde{\mathbf{m}}\|_2^2 + \lambda \|\boldsymbol{\rho}\|_2^2. \tag{2.6}$$

Reconstruction. This objective equation (2.6) can be solved very fast with Wiener deconvolution with regularization λ,

$$\widetilde{\mathrm{P}}(\boldsymbol{\omega}) = \frac{\overline{\widetilde{M}(\boldsymbol{\omega})}}{|\widetilde{M}(\omega)|^2 + \lambda} I(\boldsymbol{\omega}). \tag{2.7}$$

Separable model. A different simplification of the forward model has been used in amplitude-mask based designs that focused on producing thin imagers [Adams *et al.*, 2017, Asif *et al.*, 2016] for imaging in tight spaces (resulting in large incident ray angles), by using mask patterns that can be computed from sum of outer product of two 1D functions, resulting in a forward model that is separable in $x - y$ dimension.

Assumptions. Besides the shared ones in equation (5.2), the mask attenuation pattern is assumed to be separable, *i.e.*, $m(x,y) = \sum_{k=1}^{K} m^{x,k}(x) m^{y,k}(y)$, and the angular efficiency function of the pixels are separable, as it typically is on common sensors.

Forward model. Explicitly expanding out the x, y dimension from $\mathbf{x}$ and p, q dimension from $\mathbf{p}$ in equation (5.2),

$$i(p,q) = \int_z \int_{x,y} \rho(x,y,z)\Phi(x,y,p,q,z)dxdydz, \tag{2.8}$$

where

$$\Phi(x,y,p,q,z) = m\left(\frac{z}{z+d}p + \frac{d}{z+d}x, \frac{z}{z+d}q + \frac{d}{z+d}y\right)\widetilde{a}\left(\frac{\sqrt{(x-p)^2+(y-q)^2}}{z+d}\right) \tag{2.9}$$

is separable when the mask is separable,

$$\Phi(x,y,p,q,z) = \sum_{k=1}^{K} \Phi^{x,k}(x,p,z)\Phi^{y,k}(y,q,z). \tag{2.10}$$

Equation (2.8) becomes

$$i(p,q) = \int_z \sum_{k=1}^{K} \left(\int_y \left(\rho(x,y,z)\Phi^{x,k}(x,p,z)dx\right)\Phi^{y,k}(y,q,z)dy\right)dz. \tag{2.11}$$

With discretized x, y, z volume,

$$i[p,q] = \sum_z \sum_{k=1}^{K} \Phi_z^x[p,x]\,\rho_z[x,y]\,\Phi_z^y[y,q]\Delta_z; \tag{2.12}$$

it is usually written as a sum of 3 matrix multiplications

$$\mathbf{I} = \sum_z \sum_{k=1}^{K} \mathbf{\Phi}_z^x \boldsymbol{P}_z \mathbf{\Phi}_z^y. \tag{2.13}$$

FlatScope [Adams *et al.*, 2017] considers the non-negative mask pattern separable with $K = 2$, while previous work [Asif *et al.*, 2016] considers a (-1, 1) mask pattern with $K = 1$.

Optimization objective. For single plane of known depth z, its texture can be recovered under the separable model equation (2.13) by solving

$$\arg\min_{\boldsymbol{P}} \|\mathbf{I} - \sum_{k=1}^{K} \mathbf{\Phi}_z^{x,k} \boldsymbol{P} \mathbf{\Phi}_z^{y,k}\|_2^2 + \lambda\|\boldsymbol{P}\|_2^2 \tag{2.14}$$

Reconstruction. This problem has analytical solution for $K = 1$,

$$\boldsymbol{P} = \mathbf{V}_x[\left(\Sigma_x \mathbf{U}_x^T \mathbf{I} \mathbf{U}_y \Sigma_y\right)./\left(\boldsymbol{\sigma}_x \boldsymbol{\sigma}_y^T + \lambda \mathbf{1}\mathbf{1}^T\right)]\mathbf{V}_y^T,$$

and can be computed within a few matrix multiplications using pre-computed matrix decompositon results

$$\begin{aligned}\mathbf{\Phi}^{x,1} &= \mathbf{U}_x \Sigma_x \mathbf{V}_x^T \\ \mathbf{\Phi}^{y,1} &= (\mathbf{U}_y \Sigma_y \mathbf{V}_y^T)^T \\ \Sigma_x &= \text{diag}(\Sigma_x) \\ \Sigma_y &= \text{diag}(\Sigma_y).\end{aligned}$$

For $K = 2$, FlatScope solves it with Nesterov's gradient method.

2.3.2 Reconstruction of depth and texture

Asif [2018] along with Zheng [2020] jointly resolve the scene brightness $\rho(\mathbf{x})$ and depth $z(\mathbf{x})$ from a non-linear inverse problem.

Assumptions. 1) the scene consists of point sources without self-occlusion with brightness $\rho(\mathbf{x})$ at 3D location $(\mathbf{x}, z(\mathbf{x}))$; 2) sensor pixel pitch is small; 3) the imager sees a small field of view, *i.e.* the effective angular response $\widetilde{a}(\frac{\mathbf{x}-\mathbf{p}}{z+d})$ is constant over the field of view.

Forward model. By discretizing the scene into a collection point sources $\mathbf{x}_1, \ldots, \mathbf{x}_n$, the forward model simplifies to sum of each point rendered by its corresponding PSF,

$$i(\mathbf{p}) = \sum_{\mathbf{x}_1,\ldots,\mathbf{x}_n} \rho\left(\mathbf{x}, z(\mathbf{x})\right) m\left(\mathbf{x}_m\right) \widetilde{a}\left(\frac{\mathbf{x}-\mathbf{p}}{z(\mathbf{x})+d}\right) = \sum_{\mathbf{x}_1,\ldots,\mathbf{x}_n} \Psi\left(z(\mathbf{x})\right) \rho(\mathbf{x}) \tag{2.15}$$

where $\Psi(z)$ renders the PSF of point at depth z.

Optimization objective. The reconstruction is posed as the non-linear optimization problem

$$\arg\min_{\rho,z} \|\mathbf{i} - \sum_{\mathbf{x}_1,\ldots,\mathbf{x}_n} \Psi\left(z(\mathbf{x})\right) \rho(\mathbf{x})\|_2^2 \tag{2.16}$$

Solution. A initial solution can be obtained by reconstructing single depth planes as described in Section 2.3.1 or using greedy depth pursuit algorithm [Asif, 2018]. The solution to equation (2.16) can be refined using alternating gradient descent as proposed in Zheng and Asif [2020].

2.3.3 Reconstruction of volume albedo

Many works solve the 3D lensless imaging problem with this approach, and a comparison can be found in Table 2.1. Solving for albedo of voxels in a scene volume $\rho(\mathbf{x}, z)$ results in a linear inverse problem. Since the 3D unknown makes the problem under-determined, most methods that reconstruct volume albedo regularize the optimization with sparsity prior in either voxels or gradient of voxels.

name	**forward model**	**prior**	**optimization algorithm**	**reported runtime**
Antipa *et al.* [2018]	cropped sum of convolution (Section 2.1.1)	3D TV semi-norm	ADMM	"512×512×128≈33.5 million voxels takes 3 min (1 s per iteration) on a four-core laptop with 16 gigabytes of RAM"
Adams *et al.* [2017]	separable (Section 2.3.1)	ℓ_1	FISTA	"640× 480 pixels" "converge in 15 min"
Boominathan *et al.* [2020]	cropped sum of convolution (Section 2.1.1)	2D TV + ℓ_1	ADMM	"high-quality reconstruction within a few iterations"
Zheng *et al.* [2021]	sum of circular convolution	ℓ_2	matrix inversions	"eight depth planes with eight programmable masks" in "0.33 seconds"

Table 2.1: Comparison of projects that reconstruct 3D volume.

In summary, the prior methods for 3D lensless imaging

- solve under-determined systems;
- are difficult to analyze the resolution of their 3D reconstructions, especially in depth;
- require time-consuming solutions;
- ignores effects such as specularity, occlusion, and sensors' non-uniform angular responses.

This thesis will improve on those limitations of prior methods in the following chapters.

3 Spatial and Axial Resolution Limits for Mask-based Lensless Cameras

This chapter provides a theoretical analysis for the achievable resolutions of 3D lensless imaging.

From a theoretical perspective, 3D imaging with lensless cameras is poorly understood. While prior works have produced ample empirical observations for specific prototypes, they are difficult to generalize and interpret. In particular, the measurement operators associated with lensless imagers are hard to analyze for scenes with extended depth. As a consequence, there has been little work in analyzing the fundamental limits in achievable spatial and axial resolutions with a lensless camera; nor is there a clear understanding of how the various parameters of the imager—including the mask pattern and the scene/sensor to mask distances—affect its ability to recover texture and depth.

One of our primary observation is that the analysis of performance of lensless cameras is complicated by the dimensionality gap between the scene, which is three-dimensional (3D), and the lensless measurements, which are invariably two-dimensional (2D) or, at best, a collection of such 2D images. To alleviate this gap, we draw inspiration from z-stacking or focus stacking and consider, *as a thought experiment*, the 3D space of measurements formed by moving the image sensor axially, i.e., by changing the mask to sensor distance. We show that, under assumptions that are commonplace in prior work, the z-stacked measurements for an amplitude mask-based lensless camera are the result of a 3D convolution of the scene, represented as a volumetric albedo function, with a 3D kernel that is dependent of the mask. This convolutional structure of the measurement operator is immensely consequential, since it provides the foundation for characterizing the lensless camera's spatial and axial resolving power by simply computing the modulation transfer function (MTF) of the associated PSF. This result further enables us to compare and evaluate various masks and parameters of the lensless camera, thereby answering the previously elusive questions. Finally, since z-stacked measurements encompass those made by a traditional system with fixed sensor-to-mask distance, our results provide an upper bound on their performance.

Contributions. We make the following contributions in this chapter.

- *Analysis via lens-free z-stacks.* Our main technical result shows that the measurements obtained with z-stacking are related to the 3D scene, described as a volumetric albedo function, with a convolutional measurement operator.
- *Derivation of the MTF.* We show that the 3D MTF of the convolutional operator has a closed-form expression in terms of the attenuating mask pattern of the lensless camera.
- *Dependence of the mask on achievable spatio-axial resolution.* As a consequence of the MTF derivation, we connect the parameters of the mask to its achievable spatio-axial resolution. In particular, we derive an upper bound to the axial or depth resolution given the spatial resolution of the camera and the spatial extent of the mask.

The analysis in this chapter is meaningful for typical lensless cameras with static sensors. Since the static sensor measurement is a subset of the z-stack measurement, the 3D resolution analysis derived in this chapter serves as an upper bound to the performance of a typical static sensor lensless cameras.

Limitations. From a theoretical perspective, the derivation of the main results require a number of assumptions on the scene and the image formation that, in principle, reduce their applicability. These include the use of a volumetric model that ignores occlusion, shading, and specularities and the use of a ray tracing approach that ignores diffraction caused by the small features in the mask. Volumetric modeling is a commonly-made assumption in this literature (for example, see FlatCam [Asif *et al.*, 2016], FlatScope [Adams *et al.*, 2017] or SweepCam [Hua *et al.*, 2020]); hence, these results characterize their performance with the caveat that there is an additional mismatch between the actual measurements and the assumed measurement model. Lastly, the MTF analysis only characterizes the effectiveness of the measurement operator (or the imaging system), but does not consider the use of sophisticated computational techniques for solving the inverse problem; here, the use of scene priors can potentially offer reconstruction that is better than what our analysis predicts.

3.1 Prior Work

The theory and analysis proposed here builds upon two largely independent topics: lensless imaging, and focus stack photography.

3.1.1 Lensless measurement operators

For a lensless camera with an amplitude mask [Asif *et al.*, 2016, Busboom *et al.*, 1998, Dicke, 1968, Fenimore and Cannon, 1978], its measurement operator is convolutional when all scene points are restricted to lie on a single front-parallel plane. Hence, for a scene not restricted to a single depth, the measurement operator can be described as a sum of 2D convolutions; see Figure 3.1(a). To stabilize the reconstruction process, previous work use priors in the form of sparsity [Adams *et al.*, 2017] and data-driven models [Khan *et al.*, 2020, Monakhova *et al.*, 2019a, Rego *et al.*, 2021].

One approach to improve the conditioning of the operator is to capture multiple measurements with different mask patterns. The measurements corresponding to the different masks can simply be concatenated, as in Figure 3.1(b), to obtain a joint system with improved conditioning and invertibility. However, the computational burden in implementing this operator can be quite formidable, especially for high-resolution sensors. Hua *et al.* [2020] capture multiple measurements with a translating mask to facilitate computational refocusing to different depths *in the measurement space*; the resulting operation approximates imaging points only from focused depth with the rest in severe defocus, resulting in the measurement operator shown in Figure 3.1(c). Ignoring boundary effects of the convolution, Zheng *et al.* [2021] formulate the measurement operator in the frequency domain of the measurement; here, the sum of convolution operator reduces to a block diagonal structure, as seen in Figure 3.1(d), which can be implemented very efficiently.

In this chapter, we show that the z-stacked measurements, under an appropriate re-parameterization, are convolutional in the scene's volumetric albedo. Antipa *et al.* [2018] make a similar observation, implementing the sum of 2D convolution as a 3D convolution. However, there are notable differences including their use of a phase mask, which has a different image formation from ours. Further, our use of z-stacked measurements introduces important re-parameterizations of the scene and measurements that are critical to the derivation of the convolutional model. Finally, we detail a number of important consequences of the convolution model, which goes significantly beyond prior work.

3.1.2 Focus stack photography

Focus stack photography acquires multiple images of a scene by sweeping the focus plane of the imaging system [Kutulakos and Hasinoff, 2009, Nayar and Nakagawa, 1994]; typically, this is achieved via axial movement of the sensor with respect to the imaging lens or by using focus-tunable optics [Miau *et al.*, 2013]. Focus stacks have been studied for 3D scene estimation, using focus [Nayar and Nakagawa, 1990] and defocus [Favaro and Soatto, 2005] cues, as well as obtaining extended depth-of-field images [Pieper

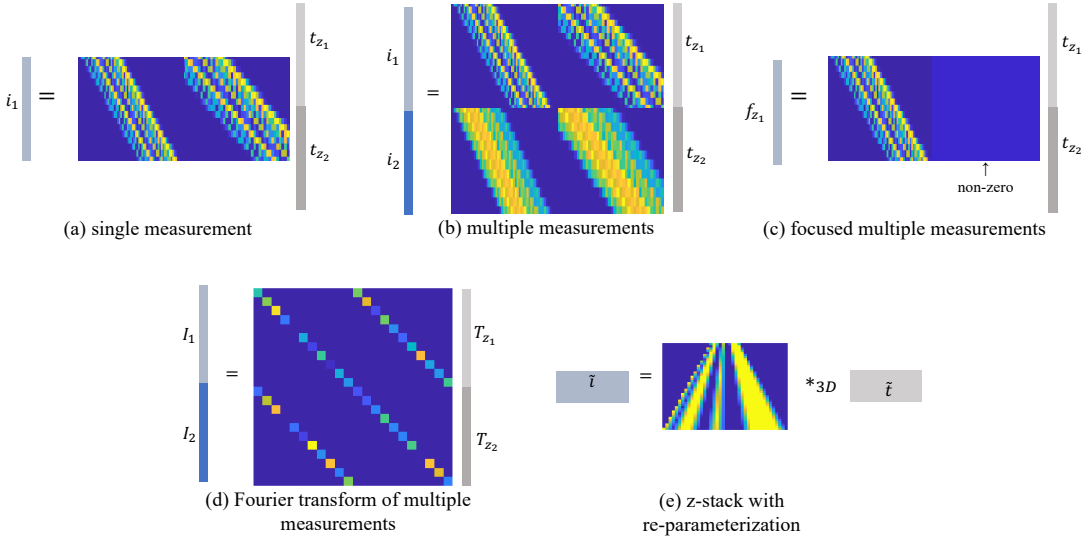


Figure 3.1: **Structure of 3D lensless imaging operators for a scene with two depth planes.** The planes have albedo t_{z_1}, t_{z_2} and the lensless camera captures measurements i. (a) Single measurement; (b) Multiple measurements; (c) Hua *et al.* [2020] computationally focus measurements on specific depths to obtain systems that approximate single 2D convolution with low-frequency residual. (d) Zheng *et al.* [2021] represent the Fourier Transform of multiple measurement matrix as a block-diagonal matrix. (e) Z-stack measurements, after re-parameterization, can be obtained by a 3D convolution between the 3D scene volume and a 3D kernel defined by the mask pattern.

and Korpel, 1983]—the latter being useful in microscopy [Streibl, 1985] where the extremely shallow depth of field is often a challenge.

Sundaram and Nayar [1997] study recoverability of depth of a textureless scene from a focus stack, and present a theoretical analysis of MTFs that is similar to this work. Specifically, they show that the volumetric measurement formed by a scene point in a tele-centric system is shift-invariant. They derive the optical transfer function associated with this system and establish conditions when the depth of a textureless scene is resolvable. The theory presented in this chapter can be interpreted as the lensless counterpart to their work; in spite of the conceptual similarity between the two results, the differences in image formation between a lens-based and lens-less imager lead to distinct results and consequences. For example, we discuss the ability to design the PSF via designing the mask pattern of the lensless camera and gain insights on mask design from the viewpoint of its achievable resolutions.

3.2 Z-Stacking and The Convolutional Model

We begin with a recap of the image formation model for imaging a 3D scene with a lensless camera, followed by derivation for obtaining the z-stacked measurements with a 3D convolutional model, and finally address its implication with static sensor prototypes.

3.2.1 Measurement model for a 3D scene

Consider an image sensor placed behind an amplitude mask defined with an attenuation function $m(\mathbf{x})$ with $\mathbf{x} = (x, y)$; we assume that the mask is aligned with the $z = 0$ plane (see Figure 4.2). Following prior work [Hua *et al.*, 2020], we model the scene as a volumetric albedo function $t(\mathbf{x}, z)$. When the sensor is placed on the plane $z = d < 0$ (*i.e.* a distance d behind the mask), the intensity observed at a point $\mathbf{p} = (p, q)$ on the sensor is given as

$$i(\mathbf{p}, d) = \int_{z=z_{\min}}^{z_{\max}} \iint_{\mathbf{x}=-\infty}^{\infty} t\,(\mathbf{x}, z)\, m\left(\mathbf{p} + \frac{-d}{z-d}(\mathbf{x} - \mathbf{p})\right) d\mathbf{x}\, dz \tag{3.1}$$

where the scene occupies the depth range $[z_{\min}, z_{\max}]$, with $0 < z_{\min} < z_{\max} < \infty$. Equation (3.1) suggests that a scene point produces a sensor measurement that is a *scaled* and *translated* copy of the mask; in particular, the scaling parameter is depth dependent. This image formation ignores the effects of light fall-off, occlusion between scene points, shading, specular reflections, and the sensor's angular response; it also ignores the effects of diffraction. In practice, the effects of diffraction causes the PSF to no longer be a scaled copy of the mask; this changes equation (3.1) by changing the mask function $m(\cdot)$ to its diffracted counterpart.

We observe that there is a reasonable depth range where the scaling assumption is valid. For example, the correlation between scaled PSF patterns remain above 0.86 when we use PSF from 11cm as template, and evaluate it in the depth range from 6cm to 34cm on a SweepCam prototype, as shown in Figure 3.2. This validates the shift-invariance in space and scaling in depth holds for the diffracted mask, as has been commonly used in most lensless camera prototypes [Antipa *et al.*, 2018, Boominathan *et al.*, 2020, Hua *et al.*, 2020, Zheng and Asif, 2020, Zheng *et al.*, 2021] to which this work provides an analysis for.

3.2.2 3D Convolution model for a z-stack

We now consider a lensless camera where the sensor captures multiple measurements while moving axially (*i.e.* with varying sensor-to-mask distances). An important distinction is that these z-stacked

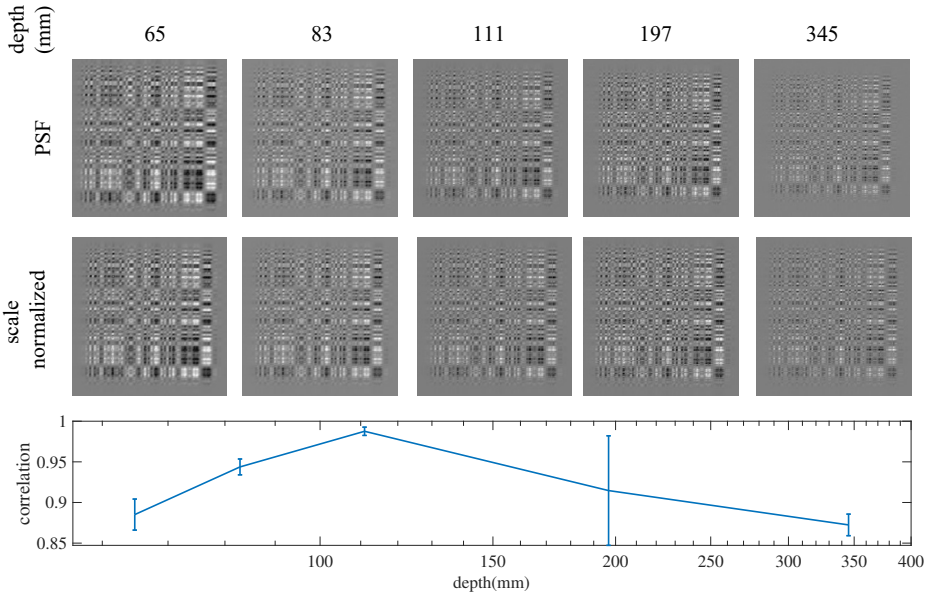


Figure 3.2: **Prototype PSF from points at different depth.** Top row: a point light source captured at 5 different depth and with three translated mask patterns (±0.14 mm) at each depth. Middle row: each measurement scaled to match the template at 111mm (center column). Bottom plot: Computed correlation of scaled measurements against the template. The plot shows the average and standard deviation of correlation for each depth.

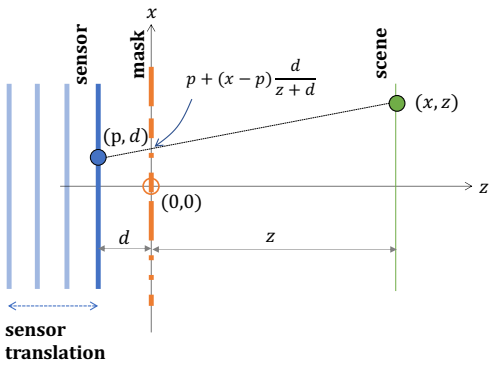
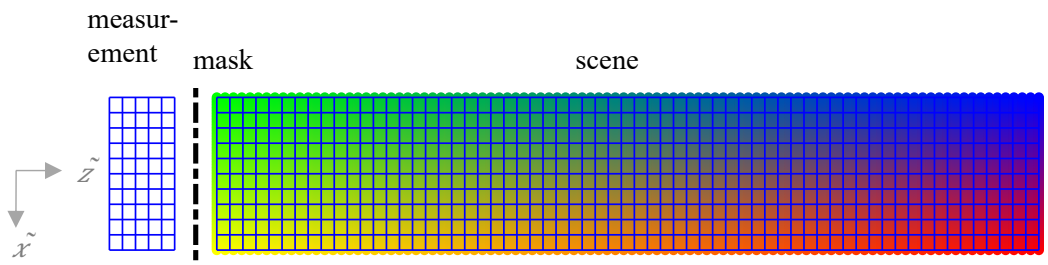
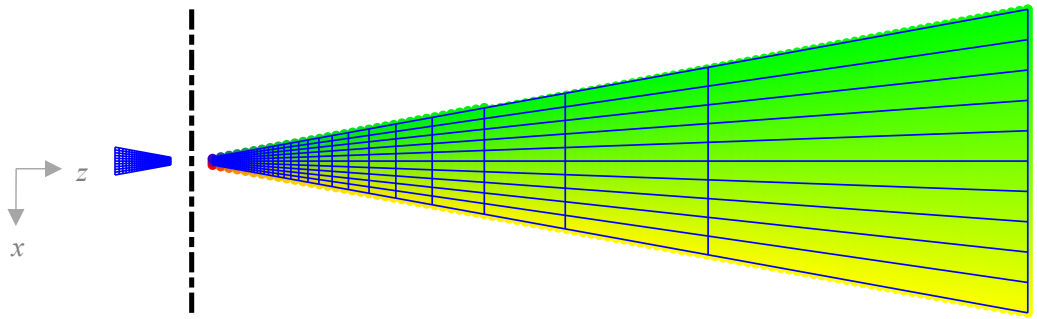


Figure 3.3: **Z-stacked lensless measurements.** The sensor is translated axially, along z, to different sensor-to-mask distances d to obtain a z-stack.



(a) uniform sampling in proposed parameterization $(\tilde{x}, \tilde{z})$



(b) uniform sampling in proposed parameterization shown in in Cartesian coordinates (x, z)

Figure 3.4: **Illustration of proposed parameterizationof the scene and measurement volume in 2D.** The grids show uniform sampling in the proposed parameterization, *i.e.* in tangent of angle subtendend $\widetilde{x}$ and diopter $\widetilde{z}$.

measurements are volumetric; specifically, we can extend equation (3.1) to write the measurements, $i(\mathbf{p}, d)$, to be explicitly dependent on the spatial locations $\mathbf{p}$ as well as axial location d.

Change of variables. We now show that the z-stack $i(\mathbf{p}, d)$ is related to a re-parameterized volumetric scene albedo $t(\mathbf{x}, z)$ via a convolution operator. Starting from equation (3.1), we can rewrite $i(\mathbf{p}, d)$ by rearranging the terms in $m(\cdot)$ as follows:

$$i(\mathbf{p}, d) = \int_{z=z_{\min}}^{z_{\max}} \iint_{\mathbf{x}=-\infty}^{\infty} t(\mathbf{x}, z) m\left(\frac{\frac{\mathbf{p}}{d} - \frac{\mathbf{x}}{z}}{\frac{1}{d} - \frac{1}{z}}\right) d\mathbf{x} dz \tag{3.2}$$

The equation above is significantly simplified if we change the variables from Cartesian coordinates to the following choice:

$$\widetilde{\mathbf{x}} = \frac{\mathbf{x}}{z}, \quad \widetilde{z} = \frac{1}{z}, \quad \widetilde{\mathbf{p}} = \frac{\mathbf{p}}{d}, \quad \widetilde{d} = \frac{1}{d}. \tag{3.3}$$

This new parameterization changes (x, y) to the tangent of the angle at the origin, and depth z to its reciprocal. An illustration of this parameterization is shown in Figure 3.4. Finally, we use $\widetilde{i}(\widetilde{\mathbf{p}}, \widetilde{d})$ and $\widetilde{t}(\widetilde{\mathbf{x}}, \widetilde{z})$ to denote the z-stack measurements and the scene albedo, respectively, in their new variables. With this, we can rewrite equation (3.2) as

$$\widetilde{i}(\widetilde{\mathbf{p}}, \widetilde{d}) = \int_{\widetilde{z}=z_{\max}^{-1}}^{z_{\min}^{-1}} \iint_{\widetilde{\mathbf{x}}=-\infty}^{\infty} \frac{1}{\widetilde{z}^4} \widetilde{t}(\widetilde{\mathbf{x}}, \widetilde{z}) m\left(\frac{\widetilde{\mathbf{p}} - \widetilde{\mathbf{x}}}{\widetilde{d} - \widetilde{z}}\right) d\widetilde{\mathbf{x}} d\widetilde{z}. \tag{3.4}$$

Here, the $1/\widetilde{z}^4$ is the modulus of determinant of the Jacobian underlying the change of variables. Defining the 3D kernel $k(\widetilde{\mathbf{x}}, \widetilde{z})$ and depth-normalized texture $\widetilde{t}'(\widetilde{\mathbf{x}}, \widetilde{z})$ as

$$k(\widetilde{\mathbf{x}}, \widetilde{z}) = m\left(\frac{\widetilde{\mathbf{x}}}{\widetilde{z}}\right), \quad \widetilde{t}'(\widetilde{\mathbf{x}}, \widetilde{z}) = \frac{1}{\widetilde{z}^4} \widetilde{t}(\widetilde{\mathbf{x}}, \widetilde{z}), \tag{3.5}$$

we can express equation (3.4) as

$$\widetilde{i}(\widetilde{\mathbf{p}}, \widetilde{d}) = \int_{\widetilde{z}=z_{\max}^{-1}}^{z_{\min}^{-1}} \iint_{\widetilde{\mathbf{x}}} \widetilde{t}'(\widetilde{\mathbf{x}}, \widetilde{z})\ k(\widetilde{\mathbf{p}} - \widetilde{\mathbf{x}}, \widetilde{d} - \widetilde{z}) d\widetilde{\mathbf{x}} d\widetilde{z}. \tag{3.6}$$

To proceed further and obtain the desired convolutional model, we need to make an additional assumption pertaining to the limits of integration. Specifically, the integral in equation (3.6) is physically meaningful—i.e., consistent with what a sensor would measure—only when the sensor and the scene are on opposite sides of the mask. After all, the mask plays no role when the sensor and scene are on its same side. We can implicitly enforce the sensor-mask-scene configuration with the following two conditions: first, the scene albedo $\widetilde{t}'(\widetilde{\mathbf{x}}, \widetilde{z}) = 0$ when $\widetilde{z} \leq 0$; and second, the output $\widetilde{i}(\widetilde{\mathbf{p}}, \widetilde{d})$ is evaluated *only* for $\widetilde{d} < 0$. We can now change the limits of integration to get

$$\widetilde{i}(\widetilde{\mathbf{p}}, \widetilde{d}) = \iiint_{\{\widetilde{z}, \widetilde{\mathbf{x}}\}=-\infty}^{\infty} \widetilde{t}'(\widetilde{\mathbf{x}}, \widetilde{z})\ k(\widetilde{\mathbf{p}} - \widetilde{\mathbf{x}}, \widetilde{d} - \widetilde{z}) d\widetilde{\mathbf{x}} d\widetilde{z} = (\widetilde{t}' *_{3D} k)(\widetilde{\mathbf{p}}, \widetilde{d}). \tag{3.7}$$

Hence, *the z-stack of measurements is related to the scene's volumetric albedo, both under their respective re-parameterizations, via a 3D convolution operator whose kernel is dependent only on the mask pattern.*

The convolutional model presented in equation (3.7) is the centerpiece of our contributions. As we show in the next section, it can be leveraged to characterize basic properties of the imager, including the MTF that is defined as the modulus of the Fourier transform of the convolutional kernel. Before we delve into this analysis, a few observations are in order.

Connection to non-line-of-sight imaging. The interested reader is referred to a recent work in non-line-of-sight imaging that formed the inspiration for the derivation above. The measurement models in non-line-of-sight imaging are similar to lensless imaging in their lack of structures that facilitate

fast implementation or analysis. O'Toole *et al.* [2018] show that an appropriate re-parameterization of the variables describing the scene and measurements lead to a convolution operator. A notable difference is that non-line-of-sight operator is 5D and O'Toole *et al.* [2018] select a 3D subset to match the dimensionality of the scene and measurements; in contrast, our work matches the dimensionality by enhancing the space of the measurements by z-stacking.

3.3 Analysis

We now derive an expression for the Fourier transform of the PSF, and use it to characterize the lateral and axial resolution of the device in terms of its mask.

3.3.1 Derivation of the MTF

A key feature of an imaging system whose measurement operator is convolutional is that we can easily characterize the fundamental limits of achievable resolution. This is often done by computing the MTF of the system, which is the magnitude of the Fourier transform of the PSF.

In our case, the PSF is a 3D function $k(\widetilde{\mathbf{x}}, \widetilde{z})$ that is defined in (3.5). Let $K(f_x, f_y, f_z)$ be the 3D Fourier Transform of 3D kernel $k(\widetilde{x}, \widetilde{y}, \widetilde{z})$.

$$K(f_x, f_y, f_z) = \iiint k(\widetilde{x}, \widetilde{y}, \widetilde{z}) e^{-j2\pi\left(\widetilde{x}f_x + \widetilde{y}f_y + \widetilde{z}f_z\right)} d\widetilde{x}\, d\widetilde{y}\, d\widetilde{z} \tag{3.8}$$

Substituting the expression for the 3D kernel in (3.5),

$$k(\widetilde{x}, \widetilde{y}, \widetilde{z}) = m\left(\frac{\widetilde{x}}{\widetilde{z}}, \frac{\widetilde{y}}{\widetilde{z}}\right),$$

$K(f_x, f_y, f_z)$ can be written as

$$\int_{\widetilde{z}} \underbrace{\left[\iint_{\widetilde{y},\widetilde{x}} m\left(\frac{\widetilde{x}}{\widetilde{z}}, \frac{\widetilde{y}}{\widetilde{z}}\right) e^{-j2\pi\left(\widetilde{x}f_x + \widetilde{y}f_y\right)} d\widetilde{x} d\widetilde{y}\right]}_{\text{2D Fourier Transform of } m(\widetilde{x}/\widetilde{z}, \widetilde{y}/\widetilde{z})} e^{-j2\pi\widetilde{z}f_z} d\widetilde{z}. \tag{3.9}$$

Let $M(f_x, f_y)$ be the 2D Fourier transform of $m(\widetilde{x}, \widetilde{y})$; using scaling properties, we can derive the 2D Fourier transform of $m(\widetilde{x}/\widetilde{z}, \widetilde{y}/\widetilde{z})$ is

$$|\widetilde{z}|^2 M\left(\widetilde{z}f_x, \widetilde{z}f_y\right).$$

Substituting this into (3.9), we get

$$K(f_x, f_y, f_z) = \int_{\widetilde{z}=-\infty}^{\infty} \widetilde{z}^2 M\left(\widetilde{z}f_x, \widetilde{z}f_y\right) e^{-j2\pi\widetilde{z}f_z} d\widetilde{z} \tag{3.10}$$

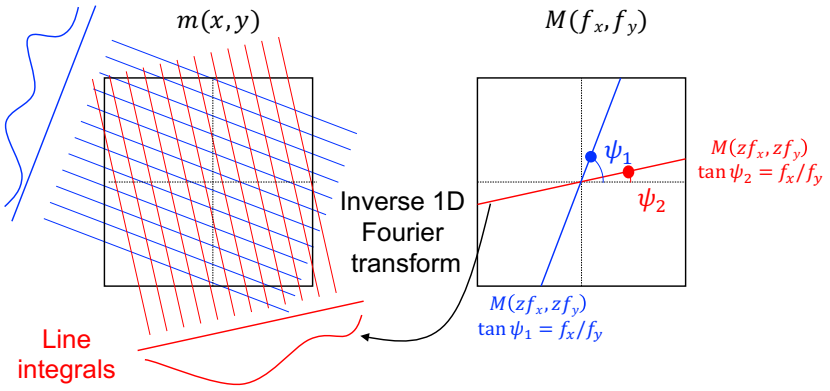


Figure 3.5: **Illustration of Fourier slice theorem.** $M(zf_x, zf_y)$ for any constant $f_y/f_x = \tan\psi$ represents a radial slice in the mask 2D Fourier space. A 1D inverse Fourier transform of any such slice is equivalent to the integral projection of the mask at angle ψ.

The integral in equation (3.10) evaluates to the 1D Fourier transform of $\widetilde{z}^2 M\left(\widetilde{z}f_x, \widetilde{z}f_y\right)$ over the variable $\widetilde{z}$.

Defining the following terms,

$$\rho = \sqrt{f_x^2 + f_y^2}, \ \tan\psi = \frac{f_y}{f_x}, \tag{3.11}$$

we can express

$$M\left(\widetilde{z}f_x, \widetilde{z}f_y\right) = M\left(\widetilde{z}\rho\cos\psi, \widetilde{z}\rho\sin\psi\right). \tag{3.12}$$

This suggests that for a fixed $(f_x, f_y) \neq (0,0)$, $M\left(\widetilde{z}f_x, \widetilde{z}f_y\right)$—as a function of $\widetilde{z}$—is a radial slice of $M(\cdot,\cdot)$ at an angle ψ to the f_x-axis. We can now apply the Fourier slice theorem to simplify the expression in equation (3.10); recall, that the Fourier Slice theorem suggests that the inverse 1D Fourier transform of $M\left(\widetilde{z}f_x, \widetilde{z}f_y\right)$, is a line integral of $m(x,y)$ (see an illustration in Figure 3.5). Finally, the term $\widetilde{z}^2$ can be accounted for using the differentiation property of Fourier transforms.

We now have all the components to simplify equation (3.10). Let $r_m(\alpha,\theta)$ be the Radon transform of the mask $m(x,y)$; the Fourier slice theorem suggests that

$$r_m(\alpha,\psi) \overset{\text{1D FT}}{\longleftrightarrow} M(\widetilde{z}\cos\psi, \widetilde{z}\sin\psi), \tag{3.13}$$

where $\widetilde{z}$ is the frequency domain variable. We can invoke the differentiation property twice, to get

$$\frac{-1}{4\pi^2}\frac{\partial^2}{\partial\alpha^2} r_m\left(\alpha,\psi\right) \overset{\text{1D FT}}{\longleftrightarrow} \widetilde{z}^2 M(\widetilde{z}\cos\psi, \widetilde{z}\sin\psi) \tag{3.14}$$

Now, from the scaling property, we get

$$\frac{-1}{4\pi^2}\frac{1}{\rho}\frac{\partial^2}{\partial\alpha^2} r_m\left(\frac{\alpha}{\rho},\psi\right) \overset{\text{1D FT}}{\longleftrightarrow} \rho^2\widetilde{z}^2 M(\rho\widetilde{z}\cos\psi, \rho\widetilde{z}\sin\psi) \tag{3.15}$$

The last technicality to resolve is that equation (3.10) requires us to calculate the *Fourier transform* of $\widetilde{z}^2 M(\rho\widetilde{z}\cos\psi, \rho\widetilde{z}\sin\psi)$; the expression above provides us with its *inverse Fourier transform.* However, duality suggests that we only need to negate α to get that expression.

Putting these together, for $(f_x, f_y) \neq (0,0)$, we can show that the 3D Fourier transform of the PSF is given as

$$K(f_x, f_y, f_z) = \frac{-1}{4\pi^2 (f_x^2 + f_y^2)^{\frac{3}{2}}} \left.\frac{\partial^2}{\partial \alpha^2} r_m(\alpha, \psi)\right|_{\substack{\alpha = -f_z/\sqrt{f_x^2+f_y^2} \\ \tan\psi = f_y/f_x}}, \tag{3.16}$$

with $r_m(\alpha, \psi)$ being the Radon transform of the mask $m(x, y)$. At the origin, $f_x = f_y = f_z = 0$, K has a Dirac delta function. Finally, if we define $\ell(x, y)$ to be the Laplacian of the mask, *i.e.*

$$\ell(x, y) = \left(\frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2}\right) m(x, y),$$

then, using the Fourier slice theorem, we can show that the Radon transform of $\ell(x, y)$, $r_\ell(\alpha, \psi)$, can be expressed as

$$r_\ell(\alpha, \psi) = \frac{\partial^2}{\partial \alpha^2} r_m(\alpha, \psi).$$

With this, we can obtain the following expression:

$$K(f_x, f_y, f_z) = \frac{-1}{4\pi^2 (f_x^2 + f_y^2)^{\frac{3}{2}}} r_\ell\left(-\frac{f_z}{\sqrt{f_x^2 + f_y^2}}, \tan^{-1}\left(\frac{f_y}{f_x}\right)\right). \tag{3.17}$$

Subsequent analysis is simplified if we define $\rho = \sqrt{f_x^2 + f_y^2}$ as the magnitude of angular frequency and $\psi = \tan^{-1}(f_y/f_x)$ as its directionality. The Fourier transform of the 3D PSF in polar coordinates for (f_x, f_y), denoted as K^P, can now be written as

$$K^P(\rho, \psi, f_z) = \frac{-1}{4\pi^2 \rho^3} r_\ell\left(-\frac{f_z}{\rho}, \psi\right). \tag{3.18}$$

Numerical verification. We observe from equation (3.18) that if ρ is fixed to some value, say $\rho = \rho_0$, then

$$K^P(\rho = \rho_0, \psi, f_z) \propto r_\ell(-f_z/\rho_0, \psi). \tag{3.19}$$

Hence, the 2D slice of K^P for a fixed ρ is a scaled copy of r_ℓ, the Radon transform of the Laplacian of the mask pattern. Figure 3.6 provides a numerical verification of this; for a collection of masks commonly used in lensless imaging, we compute the Laplacian of the mask, its Radon transform, and a ψ-f_z slice of the magnitude spectra $|K^P(\rho, \psi, f_z)|$, for a fixed value of ρ. We observe that the Laplacian matches well with the 2D slice of the magnitude spectra with two notable sets of artifacts: aliasing of the DFT which results in repeating copies, and the sinc-like decay, especially along the f_z axis, which we attribute to the windowing of the kernel; we elaborate on this in the next section.

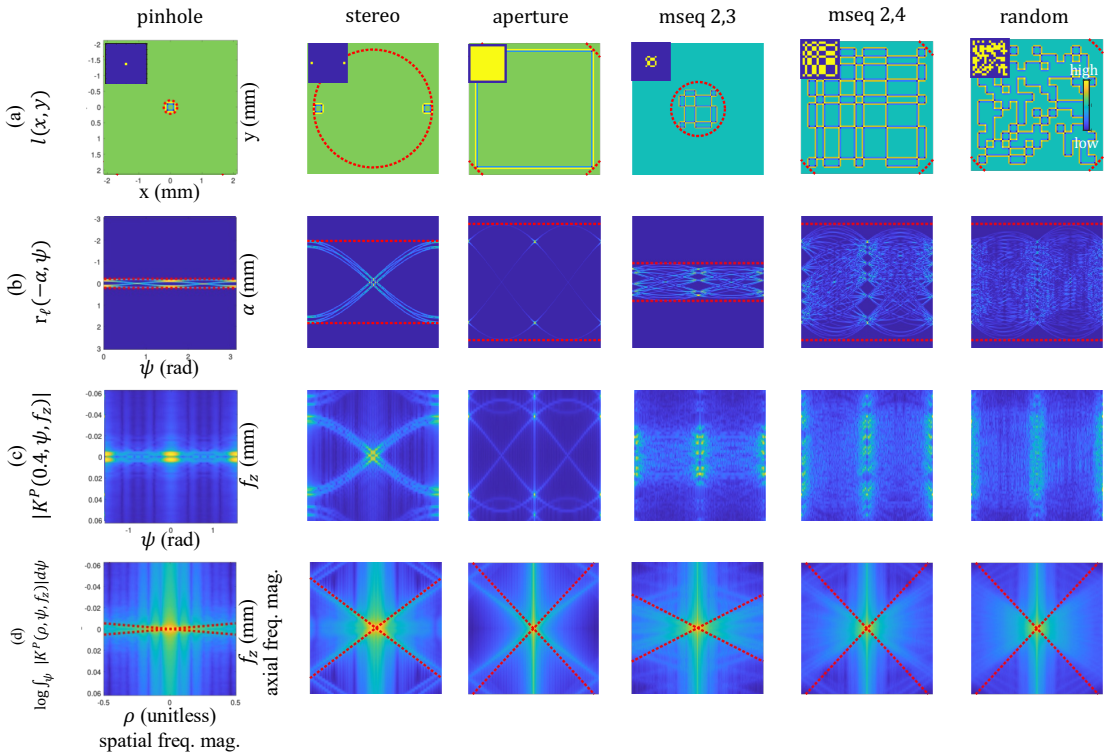


Figure 3.6: **Comparison of mask patterns and their MTFs.** Multiple mask patterns (a, inset) are shown with their Laplacian (a) and the Radon transform of Laplacian (b). (c) shows $|K^P(\rho = 0.4, \psi, f_z)|$, a ψ-f_z slice, obtained numerically; the similarity between row (b) and (c) verifies our analytical expression of K. (d) shows $\log \int_\psi |K^P(\rho, \psi, f_z)|d\psi$ obtained numerically. The red lines marked in (d) correspond to the butterfly structure defined in equation (3.20); this is a consequence of the compact spread of the mask, indicated by the red dotted circles in (a). The structures of the MTF in (d) are constrained within the butterfly structure, except for the leakage due to the windowing of the PSF in the spatial domain.

3.3.2 Dependence of the MTF on the mask

The expressions for the Fourier transform of the PSF also provide a direct way to understand how the mask affects the MTF of the resulting system. We study this next.

Spatial extent of the mask. Many of the masks that we use have a finite spatial extent. The MTFs of such masks exhibit an important symmetry. Specifically, suppose that the mask $m(x, y)$ is zero outside of a disc of radius R_m, i.e., $m(x, y) = 0,\ \forall (x, y)$ s.t. $x^2 + y^2 \geq R_m^2$, then the Radon transform of Laplacian satisfies

$$r_\ell(\alpha, \psi) = 0,\ \forall |\alpha| > R_m.$$

This implies that the Fourier transform can be non-zero only when $|f_z/\rho| \leq R_m$, or

$$-R_m\rho \leq f_z \leq R_m\rho. \tag{3.20}$$

Hence, if we visualize a different 2D cross-section of K^P, one corresponding to a fixed ψ, then we expect to see non-zero values only with in the "butterfly" shape defined by the set defined above. Figure 3.6(d) visualizes this for a number of masks.

The butterfly structure places an important constraint on achievable spatio-axial resolution[1]. Given measurements that resolve in tangent of angle subtended with a resolution of δ_p (the ratio of sensor pixel pitch to sensor-to-mask distance) , we can only resolve frequencies corresponding to $\rho \in [-\frac{1}{2\delta_p}, \frac{1}{2\delta_p}]$. Hence, for a mask with support restricted within a disc of radius R_m, the maximum resolvable axial resolution is

$$|f_z| \leq \frac{R_m}{2\delta_p}. \tag{3.21}$$

This naturally explains the lack of depth resolution in a pinhole mask and the improvement in performance with multiple pinholes, as well as larger-sized masks based on M-sequences or random constructions, since they have a larger R_m than pinholes. It is worth noting that the bound discussed above is expected to be loose, since it only considers the diameter of the mask and ignores the specific pattern within.

Example. This analysis allows us to compute the upper bound of 3D resolution for lensless camera prototypes. For example, in the FlatScope prototype [Adams *et al.*, 2017], the spatial extent of mask is contained within a disc of radius $R_m = 1.84\,\text{mm}$. The spatial resolution is bounded by $\delta_p = \frac{\Delta p}{d} = 1.12 \times 10^{-2}$, where $\Delta p = 2.24\,\mu\text{m}$ is the effective pixel pitch and $d = 0.2\,\text{mm}$ is the mask-to-sensor distance. FlatScope reports lateral resolution of less than $2\,\mu\text{m}$. After converting the sensor angular resolution (δ_p) to the scene spatial resolution ($\delta_p \times z$), our analysis predicts the $2\,\mu\text{m}$ spatial resolution holds true for scenes that are farther than $z = 178\,\mu\text{m}$ from the mask. This is the depth range in which FlatScope have shown experimental results ($z > 170\,\mu\text{m}$). The axial resolution of the prototype has an upper bound from equation (3.21): $f_{zmax} = \frac{R_m}{2\delta_p} = 82.1\,\text{mm}$ in frequency of diopters. FlatScope shows an axial resolution of less than $15\,\mu\text{m}$ at depth $z = 270\,\mu\text{m}$ from experiments. Our analysis predicts that diopter resolution is bounded by $\Delta\tilde{z} = \frac{1}{2f_{zmax}} = \frac{\delta_p}{R_m}$. The resulting depth resolution at $z = 270\,\mu\text{m}$ can be computed as $\Delta z = |\frac{\partial z}{\partial \tilde{z}}\Delta\tilde{z}| = \frac{\Delta\tilde{z}}{\tilde{z}^2} = z^2\frac{\delta_p}{R_m} = 0.44\,\mu\text{m}$. The predicted theoretical upper bound of axial resolution is better than the reported empirical axial resolution, as it does not account of the limitation

[1] Similar structures also arise in the analysis of spatio-temporal resolutions in videos [Park and Wakin, 2013]; as in our analysis, such structures serve to strongly couple achievable resolutions across the two domains.

further imposed by the subsampling operator discussed in Section 3.3.4 or sensor non-idealities. This suggest further improvements to the axial resolution may be possible with obtaining z-stack and better handling of measurement noise and quantization.

A counter-intuitive consequence of the butterfly structure is that the depth resolving power of a pinhole, modeled as a disc, seems to change when we increase its radius. We discuss this next.

Pitch of the mask. Given a mask pattern—for example, a pinhole—changing its pitch scales the mask pattern, and consequently, the Radon transform of its Laplacian only along its shift axis. This suggests that we can get better depth resolution simply by scaling a mask pattern. Intuitively, this makes sense as the defocus blur is enlarged and hence, given a sensor pitch, we can distinguish smaller changes in depth better.

Sparsity of the Laplacian. Another significant factor that detrimentally affects performance of reconstruction is the presence of nulls in the MTF. We observe such nulls in the simple masks consisting of one or a few pinholes; the Laplacian of such masks have positive and negative intensities which cancel out during the line-integrals, and so their Radon transform has sparse structures as well (as seen in Figure 3.6(c)). As a consequence, while increasing the size of the pinhole enlarges the butterfly structure, the sparsity of Radon transform results in the same amount of frequency measurements, albeit those that can reach higher axial resolutions.

DC term and the effect of windowing. The expressions in equations (3.17) and (3.18) also indicate that the Fourier transform tends to infinity when we approach the origin, i.e., the DC term. This is not surprising since the DC term is equal to the integral area under the curve (AUC). Since the PSF $k(\widetilde{\mathbf{x}}, \widetilde{z})$ is simply a projection of the mask m along the depth dimension, its AUC is infinite. Practical considerations, however, force us to work with a windowed kernel when we do the simulations of Figure 3.6. In particular, when we seek to recover a scene and sensor measurements that both have finite extents in tangent of the angle and reciprocal of the depth, then the effective kernel that we see is a windowed version of the theoretical kernel.

Windowing has two distinct effects on K. The term at the origin is now finite; K approaches the AUC of the windowed PSF as we approach the origin. Next, windowing in spatial domain results in convolution with a sinc function in the frequency domain; this results in the vertical and horizontal spread of the MTF, especially in regions outside the butterfly structure in Figure 3.6(d).

Effect of axial sampling density and range. When the z-stack is uniformly sampled within some range in the diopter space, the effect of sampling density follows the traditional trade-offs characterized by the Nyquist–Shannon sampling theorem [Shannon, 1949]; specifically, to avoid aliasing, this model would only be able to handle scenes which are band-limited along depth as determined by the sampling rate. The effect of the sampling range is to be interpreted as windowing of the ideal sampling of the z-stack, which has been discussed in the previous paragraph.

3.3.3 Lateral and axial resolution

We are interested in expressions for the MTF as functions of ρ and f_z, respectively, corresponding to characterizing the lateral and axial resolution. To get such expressions, we start with the modulus of $K^P(\rho, \psi, f_z)$ and integrate/marginalize the variables that we want to exclude. Such a marginalization assumes that all frequency components are equally important, and can be interpreted as an average over an ensemble of 2D signals, without any priors.

Lateral resolution. For lateral resolution, we are interested in characterizing the MTF, purely as a function of ρ, which we can obtain by summing over the modulus of K over f_z and ψ.

$$\mathrm{MTF}(\rho) = \frac{1}{4\pi^2\rho^3}\int_{f_z}\int_{\psi}\left|r_\ell\left(-\frac{f_z}{\rho},\psi\right)\right| df_z d\psi \qquad = \underbrace{\frac{1}{\rho^2}\left[\frac{1}{4\pi^2}\int_{f_z}\int_{\psi}|r_\ell(f_z,\psi)|df_z d\psi\right]}_{\text{constant that is mask dependent}}. \tag{3.22}$$

This suggests that the lateral resolution for different masks is similar, except for a constant. Further, it also suggest that the tail of $\mathrm{MTF}(\rho)$ decays as $1/\rho^2$; therefore, visualizing $\mathrm{MTF}(\rho)$ in a log-log plot should produce linear profiles with a slope of -2. We verify this in Figure 3.7 for the masks shown earlier in Figure 3.6.

Axial resolution. To obtain MTF as a function of f_z, we can marginalize the modulus of $K(f_x, f_y, f_z)$ over f_x and f_y, or equivalently, ρ and ψ.

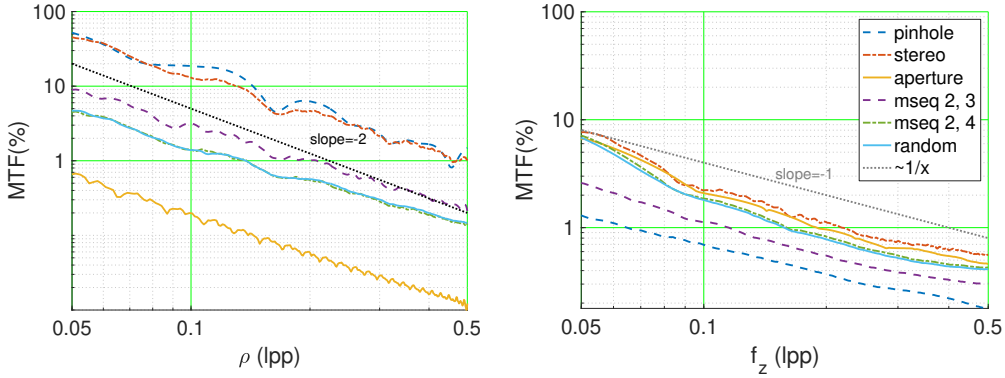
$$\mathrm{MTF}(f_z) = \frac{1}{4\pi^2}\int_{\rho}\int_{\psi}\frac{1}{\rho^3}\left|r_\ell\left(-\frac{f_z}{\rho},\psi\right)\right|\rho d\rho d\psi. \tag{3.23}$$

Suppose we define $h(f_z)$ as

$$h(f_z) = \frac{1}{4\pi^2}\int_{\psi}|r_\ell(f_z,\psi)|d\psi,$$

then

$$\mathrm{MTF}(f_z) = \int_{\rho}\frac{1}{\rho^2}h\left(\frac{f_z}{\rho}\right)d\rho = \underbrace{\frac{1}{f_z}\left[\int_{\tau}h(\tau)\,d\tau\right]}_{\text{constant that is mask dependent}}. \tag{3.24}$$



(a) spatial MTF (b) axial MTF

Figure 3.7: **Lateral and axial MTF of different masks..** The lateral and axial MTF of different masks are numerically obtained via FFT of the 3D kernel. The slopes of the lines agree with that of ρ^{-2} and f_z^{-1} respectively, as shown in black dotted lines. This validates the analytical expression in equations (3.22) and (3.24).

This suggests that the axial MTF has a tail decay of $1/f_z$, or linear with a slope of -1 in a log-log plot. We confirm this using simulations in Figure 3.7.

Remarks. The expressions on the slices of the MTF in equations (3.22) and (3.24) are obtained by integrating out the variables that are not of immediate interest. Implicitly, this assumes that all frequencies are equally important; in reality, natural scenes have their energy concentrating on low-frequencies and hence, the plots have to be interpreted with this distinction in mind. This also explains why the open aperture appears at an higher value in the MTF(ρ) plot; Figure 3.6(d) shows that the open aperture samples higher depth frequencies compared to the pinhole so de-emphasizing the high frequencies – to match their distribution in natural scenes – will likely result in an aperture MTF curve that is worse off as compared to the pinhole.

3.3.4 Reduction to the static sensor scenario

The analysis of the MTF of the z-stacked measurements have immediate consequences for traditional lensless cameras, such as FlatCam [Asif *et al.*, 2016], FlatScope [Adams *et al.*, 2017], and SweepCam [Hua *et al.*, 2020], that rely on a single (or multiple) image measurements without sensor movement; after all, we can simply choose to retain a single image in the z-stack, which would correspond to a static sensor scenario. The measurement operator associated with a static sensor can be written as a sub-sampling

operator applied to that of the z-stacked measurements; specifically, if we denote the measurement operators associated with the z-stacked sensor and the static sensor as $\mathcal{A}_{\text{zcam}}$ and $\mathcal{A}_{\text{static}}$, respectively, then they are related as follows:

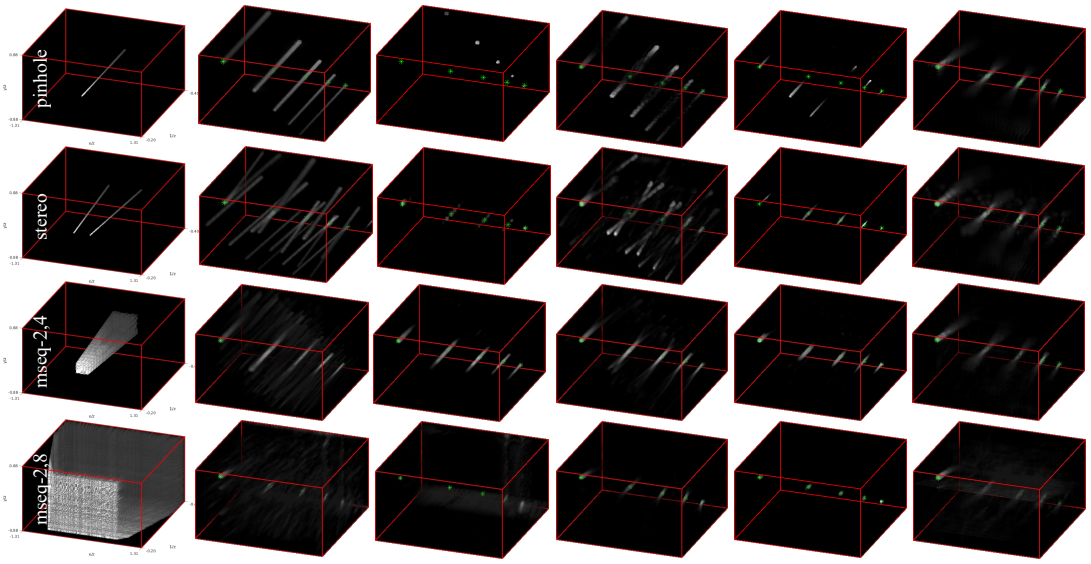
$$\mathcal{A}_{\text{static}} = \mathcal{S}_z \circ \mathcal{A}_{\text{zcam}},$$

where $\mathcal{S}_z$ is the subsampling operator along the z-axis that selects the measurements corresponding to the sensor-to-mask distance associated with the static case. As a consequence, the operator in the static scenario $\mathcal{A}_{\text{static}}$ inherits all the disadvantages of the z-stacked operator $\mathcal{A}_{\text{zcam}}$; for example, the null space of the latter is necessarily a subset of the former. In the case of SweepCam, the mask is translated laterally (i.e., along $x - y$); each of those masks is associated with a different z-stacked system. So the nulls in the MTF in one mask can potentially be alleviated by its translation.

Remarks. The predictions we make are in the reparameterized space, where the spatial coordinates are represented as tangent of angles subtended and the axial coordinates are in reciprocal of depth. These transformed coordinates have natural intepretability in the context of imaging. In traditional lens-based camera, spatial coordinates on the sensor are linear in the tangent of the angle subtended by scene points. Similarly, the size of the defocus blur, with or without a lens, is linear in the reciprocal of depth. Thus, an analysis in the transformed space is meaningful once we evaluated it through this perspective.

3.4 Simulation Results

Setup. Since the discussion above was heavily based on a volumetric representation of the scene, we aim to test the proposed method in realistic situations that involve light fall-off, occlusion, foreshortening, and sensor angular response; such effects can be significant for scenes very close to the mask. Specifically, we render the simulated measurements by ray-tracing a mesh-based scene so that it captures the above mentioned effects. However, the renderer does not model diffraction and other wave effects. Details of the renderer and its comparison to the volumetric modeling can be found in supplementary material. Additionally, we add Gaussian noise based on the dynamic range of a typical machine vision sensor (71.95 dB). For z-stacking, we translate a 13.13mm×8.75mm sensor from 5mm to 10mm to the mask, in 64 steps linear in $\frac{1}{d}$, and render a measurement at each location. The dimension of our measurements is given by $114 \times 171 \times 64$. The dimension of reconstructed volume is the same. Each of the reconstructions were finished within 5 minutes. For single measurement results, we use the measurement captured at the furthest location in the z-stack.



(a) 3D kernel (b) single: ℓ_2 reg. (c) single: ℓ_1 reg. (d) z-stack:ℓ_2 reg. (e) z-stack:ℓ_1 reg. (f) sweepcam: ℓ_2 reg.

Figure 3.8: **Different masks and their reconstructions.** The 3D kernel and reconstructed volumes are all plotted in re-parameterized space as given in Section 3.2.2. The scene consists of five points aligned on a diagonal line; the ground truth location of points are overlayed with green asterisks in the reconstructions. In single or z-stack measurements, shown in column (b) to (e), mask pattern with larger spread result in higher depth resolution – mseq-2,8 reconstruct points as points instead of line streaks in the other masks under ℓ_2 regularization. While a sparse prior (ℓ_1) results in sparse points in the reconstruction, they can be located at the wrong depth for masks with poor depth resolution (pinhole); thus it is essential to characterize the cameras' resolution theoretically in addition to empirical observations. The sweepcam [Hua *et al.*, 2020] results, shown in column (f), capture the scene with translated masks, which alleviates the nulls in the MTF.

Performance of different masks. We experimentally validate our resolution analysis of mask patterns in Figure 3.8. We image a scene with different mask patterns under the proposed z-stacking. The scene contains five points of diameter 200 µm evenly spaced on a line between point (-2mm, 2mm, 5mm) and (4mm, -4mm, 10mm). Figure 3.8 visualizes the 3D kernels for a set of masks, and reconstructions using regularized least squares (ℓ_2) and a sparse ℓ_1 prior (with FISTA [Beck and Teboulle, 2009]). The results reflect the observations made in Section 3.3.3 about the spatial and axial resolution of the masks. As expected, pinhole has almost no depth resolution, and the result reflects this by producing long streaks instead of points. Stereo masks have better depth resolution, and results in shorter streaks in the ℓ_1 reconstructions. Note the stereo mask ℓ_2 regularization results show long streaks; this suggests typical

stereo systems observe depth with strong dependence on sparsity of the scene. The longer M-sequence mask has the best depth resolution, and results in sparse points.

Additionally, our prediction in Section 3.3.4 states that using translated masks in SweepCam [Hua *et al.*, 2020] alleviates nulls in the MTF of one mask by translation, and our numerical results from column (f) validates the prediction as we observe similar resolution from all four masks, though we observe the SweepCam-fast results are blurrier than mseq-2,8 single ℓ_2 results, because SweepCam-fast algorithm trades off accurate modeling for speed.

Finally, It is also instructive to note from column (c) and (e) that using a sparse prior always produces isolated points in the volume; however, for the masks with poor depth resolution, the location of those points are incorrectly reconstructed. This highlights the importance of theoretical analysis, as the use of strong priors makes it difficult to analyze the performance of lensless cameras.

3.5 Discussion

This chapter provides a theoretical characterization of the performance of lensless images, in terms of spatio-axial resolution, and the central role played by the mask. Our primary result relies on the construction of a measurement operator that is convolutional; this involves two steps: using z-stacked measurements to obtain a 3D space of measurements, and a reparameterization of the space. This construction connects the MTF of the system to a simple transformation of the mask. More importantly, it makes a concrete set of predictions on the achievable spatial and axial resolutions: a butterfly structure that limits the depth resolution based on sensor pitch and the mask extend, and tail decays on the marginal MTFs over spatial and axial frequencies. Finally, we verify these predictions on a set of commonly-used masks. We envision the impact of this work and its relevance to the imaging community to be two-fold. First, it analyzes spatial and axial resolutions for prior art in lensless imaging [Adams *et al.*, 2017, Asif, 2018, Hua *et al.*, 2020, Zheng and Asif, 2020, Zheng *et al.*, 2021]. Second, it provides a pathway for the design of lensless cameras, in mask design and in acquiring z-stacked measurements, which we detail in the following paragraphs.

Design of masks. The theory developed in this work, especially equations (3.17) and (3.18), provide a crisp connection between the mask used and the MTF of the resulting system; specifically, that the MTF is a resampling of the Radon transform of the Laplacian of the mask used. Radon transforms are invertible, and basic Fourier analysis suggests that the Laplacian of the mask specifies the mask completely, except for the DC offset and slope. This suggests that mask design can be formulated as an

optimization problem in r_ℓ under some desired cost function.

Acquiring z-stacked measurements. The analysis in this chapter also raises the intriguing possibility of building lensless cameras with axial sensor motion, so as to provide a richer space of measurements. Perhaps, the most significant among these stems from the challenges in acquiring z-stack images. Axial motion of a sensor invariably results in lateral motion as well, which needs to be accounted for, via careful calibration. Axial motion can potentially be in conflict with the primary motivation of lensless cameras, namely the need for imaging with a compact footprint. Yet, such mechanisms for translation are routinely used in cellphone lenses for autofocusing, and so there is the possibility that such a feature can be implemented for the sensor as well. Finally, the resampling of sensor measurements required to enable the convolutional model is non-uniform and, hence, results in a loss in sensor resolution.

4 Sweepcam – Depth-aware Lensless Imaging using Programmable Masks

This chapter presents a hardware upgrade and its associated fast reconstruction algorithm for depth-aware lensless imaging.

As discussed in Chapter 2, one approach to simplify the non-linear reconstruction problem of 3D lensless imaging is to represent the scene as an intensity function over a 3D volume, instead of texture and depth map; this "lifting" of the unknown variables results in a linear inverse problem [Adams *et al.*, 2017, Antipa *et al.*, 2018]. This approach is especially promising given the extensive studies on linear inverse problem and it benefits from a rich suite of tools for analyzing and solving them. Unfortunately, for scenes with dense textures, spread over a large depth range, the resulting inverse problem is severely under-determined, i.e., the number of unknowns vastly outnumbers that of measurements. The dimensionality gap between number of unknowns and measurements can be resolved by obtaining more measurements, which this chapter facilitates via the use of a programmable amplitude mask.

We propose the use of programmable masks to improve the conditioning of the image and depth estimation problem (see Figure 4.1). Borrowing ideas from light field cameras, we translate a single mask pattern which in effect provides with us with coded images from novel viewpoints. We analyze the resulting system and show that the main operations underlying reconstruction are identical to producing a coded focus stack of the scene. A volumetric texture of the scene is subsequently obtained using simple deconvolution techniques.

Contributions. This section proposes *SweepCam* which advances lensless imaging via the use of programmable masks. Our main contributions are as follows.

- *Choice of multiple mask patterns with efficient forward model.* Exploiting ideas in plane-sweep stereo [Collins, 1996], we propose to regularize the depth recovery using measurements made from a translating mask and processed by a computational focusing operator.

Figure 4.1: **Lensless focal stack.** Images reconstructed at three different depths using our proposed SweepCam technique, which is a lensless camera with a programmable mask.

- *Fast reconstruction via the focusing operator.* We show that a computationally intensive multi-image recovery procedure can be decoupled into a collection of single image deconvolutions. This provides significant computational benefits especially when the scene has content on a large number of depths.
- *Validation using an experimental prototype.* On a lab prototype, we demonstrate that programmability of the mask enhances the quality of image reconstructions, especially when compared to state-of-the-art lensless imagers and their associated algorithms.

Limitations. The improvements provided by SweepCam come at the cost of taking multiple measurements and, hence, a loss in the time resolution of the device. Further, our implementation suffers from the poor contrast of the device that we use to implement the programmable masks.

4.1 Prior Work

4.1.1 Lensless Imaging with Static Masks

This work builds upon the core ideas from previous lensless imagers, especially FlatCam [Asif *et al.*, 2016] and DiffuserCam [Antipa *et al.*, 2018]. FlatCam covers a bare sensor with a coded mask printed on film and significantly reduces the thickness of imagers. There has been subsequent work in extending FlatCam for applications in face-detection [Tan *et al.*, 2018], privacy protection[Nguyen Canh and Nagahara, 2019], and fluorescent microscopy [Adams *et al.*, 2017]. More recent work has focused on mitigating inadequacies of the calibration and reconstruction procedure by including a deep neural network in the reconstruction pipeline [Khan *et al.*, 2019].

DiffuserCam places a diffuser that produces a caustic pattern on the sensor, and establishes the forward model as 3D convolution with cropping. We adopt the same forward model as DiffuserCam. However, reconstructing a 3D volume from a single measurement is severely under-determined, and only

possible under a sparse signal prior. To avoid such priors, we focus on obtaining more measurements so that reconstruction of 3D volume from lensless measurements is viable even for densely occupied scenes.

Another line of work [Asif, 2018, Zheng and Asif, 2020] jointly estimates depth and texture of the scene from one or more FlatCam measurements. Each scene point is assumed to be opaque, resulting in a model that suggests that there is only one scene point along each ray. Simulations show that, under this assumption, rough depth of the scene points can be recovered by a greedy depth-pursuit algorithm [Asif, 2018] and then refined by an alternating descent algorithm [Zheng and Asif, 2020]. It is also shown that, when the scene is imaged from multiple view points, the reconstruction quality is better than that from a single view point. Instead of measuring from multiple sensors as in [Asif, 2018], we propose to image with a shifted mask pattern on top of a single sensor, which effectively provides multiple viewpoints, but results in a simpler reconstruction algorithm.

4.1.2 Lensless Imaging with a Programmable Mask

Zomet and Nayar [2006] use multiple liquid crystal displays (LCDs) as a programmable aperture whose field-of-view can be changed without mechanical movements. While Zomet and Nayar implemented an “flexible pinhole" to form images of regions of interest on image sensor, the proposed design allows a more general programmable coded aperture, and reconstructs the scene computationally, which additionally allows estimation of depth.

4.1.3 Multiple Capture Imagers

The ideas in this chapter are closely related to prior work on multiple-capture imagers proposed in the context of compressive sensing; example include the single pixel camera [Duarte *et al.*, 2008], the CASSI system [Kittle *et al.*, 2010, Wagadarikar *et al.*, 2008] for hyperspectral imaging, and CACTI imager [Llull *et al.*, 2013] for high-speed imaging. These systems are similar to SweepCam in that they capture multiple coded images of a scene; however, in a broad sense, our system is different primarily because of its lensless nature, which leads to a different set of challenges when it comes to implementation and reconstruction.

4.2 Image Formation Model with Programmable Mask

We review the basic image formation models underlying lensless imaging systems, building up to the image formation model for a lensless imager with programmable amplitude mask. For brevity, the equa-

tions are presented in two dimensions on the $x-z$ plane, where z axis is perpendicular to the sensor; all conclusions generalize trivially to the three-dimensional case. Note this chapter uses different notation for albedo, mask attenutation function, and measurements, but describes the same sum of convolution model as in Section 2.1.1.

Consider a lensless imager consisting of a sensor and a programmable amplitude mask, placed at a distance d in front of the sensor, as illustrated in Figure 4.2. We will first derive a simplified image formation model under a single *static* amplitude mask for a scene confined to a single plane (parallel to the sensor) and subsequently extend the model to scenes on multiple depths as well as programmable masks. We also assume that the origin of the coordinate axes is at the center of the amplitude mask.

4.2.1 Scene on a Single Depth Plane

If the mask attenuation function is given as $a(x)$, then a point light source with effective brightness $t_0(x_0)$ placed at (x_0, z_0) produces a image measurement that is a scaled version of its PSF,

$$b(x) = t_0(x_0)\; a\left(x + (x_0 - x)\frac{d}{z_0 + d}\right). \tag{4.1}$$

This expression is true under a small angle approximation, specifically, that different pixels on the sensor measure the same intensity from the point light source.

We define a textured scene as a collection of point light sources, each inducing a measurement according to equation (4.1). When the scene is constrained to a single depth at $z = z_0$, the intensity formed at a sensor pixel x can be written as

$$b(x) = \int_{x_0} t_0(x_0) a\left(x + \frac{x_0 - x}{z_0 + d} d\right) dx_0. \tag{4.2}$$

We can simplify this expression in equation (4.2) to obtain the convolution model:

$$b(x) = \widetilde{t_0}(x) * \widetilde{k_0}(x), \tag{4.3}$$

where

$$\widetilde{t_0}(x) = \frac{z_0}{d} t_0\left(-\frac{z_0}{d}\, x\right) \text{ and } \widetilde{k_0}(x) = a\left(\frac{z_0}{z_0 + d}\, x\right).$$

The convolutional model uses a reparameterization of the scene and the mask that is depth dependent. While we ignored effects of diffraction in modeling of PSF in equation (4.1), in our experiments, we directly measure the kernel $\widetilde{k}(\cdot)$ which includes the effects of diffraction as shown in Figure 4.3(a). More experiments verifying the convolutional model can be found in our supplementary material.

Upon discretization, the image formation model in (4.3) can be written as

$$\mathbf{b} = K_{z_0,a}\; \mathbf{t}_0, \tag{4.4}$$

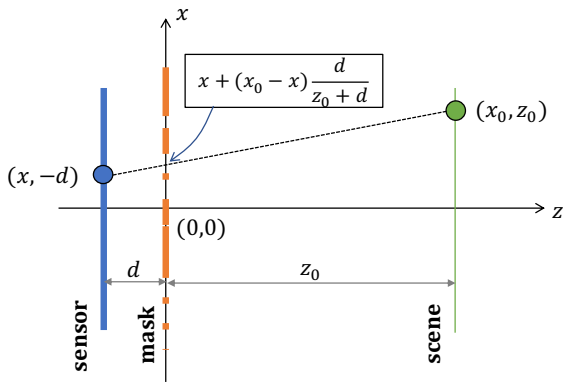


Figure 4.2: **Schematic of a lensless imager..** A mask is placed at a distance d from the sensor. Ray from point (x_0, z_0) reaches sensor pixel $(x, -d)$ after crossing the mask at $(x + \frac{x_0 - x}{z_0 + d} d)$.

where $\mathbf{b}$ and $\mathbf{t}_0$ are the vectorized image measurements and scene points texture, respectively, and $K_{z_0,a}$ is a Toeplitz matrix, representing a linear convolution operator, associated with the mask $a(\cdot)$ and the scene depth z_0.

Image recovery. Given the measurements $\mathbf{b}$, the depth z_0 and the mask $a(\cdot)$, or equivalently the Toeplitz matrix $K_{z_0,a}$, we can reconstruct $\mathbf{t}_0$ by solving the linear inverse problem in equation (4.4). Classic mask designs based on URA [Fenimore and Cannon, 1978], MURA [Gottesman and Fenimore, 1989] and M-sequences [Golomb, 1967] are designed to provide an inverse that is convolutional, at least as an approximation[1]. For the approach in this chapter, we use a small-sized mask pattern that is an outer product of two M-sequences, and it allows us to solve the system of equations using fast deconvolutional techniques including, for example, Wiener deconvolution. For the sake of simplified exposition, we assume the existance of a deconvolutional operator $K^{-1}_{z_0,a}$ that can invert the operator $K_{z_0,a}$.

4.2.2 Scene on Multiple Depth Planes

The image formation model in (4.3) and (4.4) is easily extended to a non-planar scene if we discretize the scene depths as well as assume that the effects of occlusion are minimal. Given a scene with content of D depth planes with depths $\{z_\ell, \ell = 1, \ldots, D\}$ and textures $\{\mathbf{t}_\ell, \ell = 1, \ldots, D\}$, the (discretized) image

[1]The nature of this approximation comes from replacing linear convolution with circular convolution, which is acceptable when the sensor area is larger than the mask.

formation can be written as

$$\mathbf{b} = \sum_{\ell=1}^{D} K_{z_\ell,a}\, \mathbf{t}_\ell = \begin{bmatrix} K_{z_1,a} & \cdots & K_{z_D,a} \end{bmatrix} \begin{bmatrix} \mathbf{t}_1 \\ \vdots \\ \mathbf{t}_D \end{bmatrix}. \tag{4.5}$$

Image recovery. As before, solving for the unknown scene texture at each depth, given the single image measurement $\mathbf{b}$, is a linear inverse problem. However, this system can be severely under-determined for large number of depths. One approach regularizes the inverse problem with signal priors by solving an optimization problem

$$\min_{\mathbf{t}_1,\ldots,\mathbf{t}_D} \|\mathbf{b} - \sum_{\ell=1}^{D} K_{z_\ell,a}\mathbf{t}_\ell\|^2 + \rho(\mathbf{t}_1,\ldots,\mathbf{t}_D), \tag{4.6}$$

where $\rho(\cdot)$ is a regularizing penalty function. For example, in DiffuserCam, an ℓ_1-penalty is used as the prior to promote sparsity in the scene textures. Solving such optimization problems require joint estimation of a large number of unknowns, and is computationally intensive even when we use efficient implementations for $K_{z,a}$.

A different approach is to first solve the texture at each depth in isolation, assuming that the contributions from the rest are absorbed into noise, and then in post-processing, reason about which pixels belong to which depths. That is, for $\ell \in \{1,\ldots,D\}$, we solve for

$$\widehat{\mathbf{t}}_\ell = \arg\min_{\mathbf{t}_\ell} \|\mathbf{b} - K_{z_\ell,a}\mathbf{t}_\ell\|^2 + \rho(\mathbf{t}_\ell) \tag{4.7}$$

and use contrast-based cues to clean up the reconstructions. For example, suppose that a deconvolutional kernel for $K_{z_1,a}$ existed, then an estimate for t_1 can be obtained as:

$$\widehat{\mathbf{t}}_1 = K_{z_1,a}^{-1}\mathbf{b} = \mathbf{t}_1 + \underbrace{\sum_{\ell=2}^{D} K_{z_1,a}^{-1} K_{z_\ell,a}\mathbf{t}_\ell}_{\text{(cross-plane interference)}}.$$

We observe that the reconstruction can suffer from interference across planes, and we can hope to recover high quality reconstructions only if copies of the mask a under scaling are sufficiently incoherent with each other, or equivalently, $K_{z_1,a}^{-1}K_{z_\ell,a}$ has very small spectral norm. Unfortunately, this is generally not true, as shown in Figure 4.3, especially since depth planes in close proximity will likely have very similar PSFs. Further, the artifacts arising out of this interference are generally sharp, which makes subsequent post-processing non-trivial. We aim to improve the conditioning of the imaging system by acquiring multiple images using a programmable mask.

4.2.3 Programmable Masks

Suppose that we collect N measurements $\mathbf{b}_n$ with mask a_n for $n = 1, \ldots, N$, then each measurement is

$$\mathbf{b}_n = \sum_{\ell=1}^{D} K_{z_\ell, a_n}\, \mathbf{t}_\ell. \tag{4.8}$$

We can now formulate a single linear system

$$\begin{bmatrix} \mathbf{b}_1 \\ \vdots \\ \mathbf{b}_N \end{bmatrix} = \begin{bmatrix} K_{z_1,a_1} & \cdots & K_{z_D,a_1} \\ \vdots & \ddots & \vdots \\ K_{z_1,a_N} & \cdots & K_{z_D,a_N} \end{bmatrix} \begin{bmatrix} \mathbf{t}_1 \\ \vdots \\ \mathbf{t}_D \end{bmatrix}. \tag{4.9}$$

We can write the image formation model in (4.16) as

$$\mathbf{b} = \mathbb{K}\mathbf{t},$$

and there are numerous approaches to recovering $\mathbf{t} = [\mathbf{t}_1, \cdots, \mathbf{t}_D]^\top$. There are two important considerations that determine the efficacy of using programmable masks: the choice of the mask patterns and the computational complexity of the recovery algorithm.

Choice of mask patterns. The choice of mask patterns $a_n(x)$ is extremely important and has important implications in the conditioning of the matrix $\mathbb{K}$. In the case of static masks, popular choices include codes based on URA, MURA, Hadamard and M-Sequences — all of which have many desirable properties. In contrast, the design of similar mask patterns for multi-image recovery is relatively unexplored.

Computational complexity. A second consideration is the computational complexity of the recovery procedure, which can be effectively characterized by the amount of time required to implement the operator $\mathbb{K}^\top\mathbb{K}$. The operator $\mathbb{K}$ is comprised of operators K_{z_ℓ,a_n} which are all convolutional operators; the associativity property of convolutions can be invoked to reduced the total number of computations. Therefore, we can implement $\mathbb{K}^\top\mathbb{K}$ with $\min(2ND, D^2)$ convolutional operators, which can be prohibitive for large values of N and D.

In the next section, we describe a simple technique that addresses both of these concerns.

4.3 SweepCam

We now provide a simple design for mask patterns that leads to a computationally efficient solution to the inverse problem. Specifically, we emulate a camera array using the programmability of the mask and use techniques inspired from plane-sweep stereo to simplify the complexity of the recovery procedure. We refer to this technique as *SweepCam*.

4.3.1 Mask Design for Fast Computation of $\mathbb{K}^\top\mathbb{K}$

Digging deeper into equation (4.16), we can derive the expression for the Gram matrix $\mathbb{K}^\top\mathbb{K}$ as

$$\begin{bmatrix} \sum_n K_{z_1,a_n}^\top K_{z_1,a_n} & \cdots & \cdots & \cdots & \sum_n K_{z_1,a_n}^\top K_{z_D,a_n} \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ \sum_n K_{z_D,a_n}^\top K_{z_1,a_n} & \cdots & \cdots & \cdots & \sum_n K_{z_D,a_n}^\top K_{z_D,a_n} \end{bmatrix}.$$

This Gram matrix has a block structure with diagonal blocks given as

$$\sum_n K_{z_\ell,a_n}^\top K_{z_\ell,a_n},$$

and off-diagonal blocks given as

$$\sum_n K_{z_\ell,a_n}^\top K_{z_r,a_n}, \text{ for } \ell \neq r.$$

We make two observations that motivate the choice of the mask patterns that we use. First, since K_{z_ℓ,a_n} is a convolutional operator for some kernel, say k_ℓ, the operator $K_{z_\ell,a_n}^\top K_{z_\ell,a_n}$ is a convolution with the autocorrelation function of k_ℓ. It is well-known that autocorrelations are invariant to translations. Hence, if we had a well-designed mask a_0 that has desirable properties for the single mask case, including robust inverses and fast implementations, we can reuse it simply by translating it. In this case, the diagonal block becomes N multiplied by convolution with the autocorrelation function:

$$NK_{z_\ell,a_0}^\top K_{z_\ell,a_0}.$$

In essence, it enriches the space of measurements we can obtain without having to redesign the masks. Second, as we will show in this section, translating the mask serves to decouple contributions from different depths. This forms the motivation for our use of a translating mask.

4.3.2 Translating Masks

SweepCam relies on taking multiple image measurements by translating the mask pattern, i.e., the displayed mask patterns are shifted versions of each other. Consider imaging with mask functions a_n for $n \in \{1, \ldots, N\}$, shifted in steps of Δ, where

$$a_n(x) = a_0(x - n\Delta) = a(x) * \delta(x - n\Delta). \tag{4.10}$$

Translating the mask patterns effectively changes the camera's viewpoint; this leads to a depth-dependent translation of the measurements that is referred to as disparity, following standard convention from stereo. Specifically, for scene points at depth z, the measurement corresponding to mask $a_n(x)$

translate by nv_z, where the disparity v_z can be computed from equation (4.10) ,

$$v_z = \Delta(1 + d/z). \tag{4.11}$$

Thus, we can selectively focus on measurements from a single known depth if we can align the contributions from this depth plane by undoing this translation. Such a focusing operation constructively adds measurements from a single plane while blurring out those from other depth planes.

4.3.3 Focusing

For simplicity in exposition, we first describe the concept in continuous domain. Given image measurements $\{b_1(x), \ldots, b_N(x)\}$ taken with translated mask pattern a, and a focus disparity parameter v, the focused measurement corresponding to this disparity v is given as

$$f_v(x) = \frac{1}{N} \sum_{n=1}^{N} b_n(x + nv). \tag{4.12}$$

The focusing operation aligns the contribution from a specific depth while blurring out those from other depths. Conceptually, this is similar to focus sweep operation used in plane-sweep stereo and multi-camera arrays. An example of focusing operation is shown in Figure 4.4.

To better understand the effect of the focusing operator, suppose that the scene is restricted to a single depth $z = z_0$. Starting from equation (4.3), we can derive a simplified expression for the focused image. The captured image,

$$b_n(x) = \widetilde{t}_0(x) * a_n\left(\frac{z_0}{z_0 + d}x\right) = \widetilde{t}_0(x) * a\left(\frac{z_0}{z_0 + d}x\right) * \delta\left(\frac{z_0}{z_0 + d}x - n\Delta\right) \propto \widetilde{t}_0(x) * \widetilde{k}_0(x) * \delta\left(x - nv_{z_0}\right).$$

After translation by nv becomes

$$b_n(x + nv) \propto \widetilde{t}_0(x) * \widetilde{k}_0(x) * \delta\left(x - n\Delta v_{z_0} + nv\right).$$

Thus, the focused image is filtered by $\beta_{z_0}(x)$,

$$f_v(x) \propto \widetilde{t}_0(x) * \widetilde{k}_0(x) * \beta_{z_0}(x),$$

where

$$\beta_{z_0}(x) = \frac{1}{N} \sum_{n=1}^{N} \delta\left(x - n\left(v_{z_0} - v\right)\right). \tag{4.13}$$

When $v = v_{z_0}$ then the shift applied to the image measurements cancels out that of the mask pattern, $\beta(x) = \delta(x)$. In contrast, when $v \neq v_{z_0}$, then we filter the measurement with the filter $\beta(x)$ that progressively suppresses more frequencies as N increases.

In the discretized setting, the focusing operation can be expressed easily if we introduce a shift operator $\mathcal{S}_v$ which translates the input by v pixels, where v is real valued. With some basic algebraic manipulation we can show that

$$\mathbf{b}_n = \sum_{\ell=1}^{D} \mathcal{S}_{n v_{z_\ell}} K_{z_\ell,a}\, \mathbf{t}_\ell. \tag{4.14}$$

Hence, the focused measurements for some focus disparity v_0 can be written as

$$\mathbf{f}_{v_0} = \frac{1}{N}\sum_{n=1}^{N} \mathcal{S}_{-n v_0}\mathbf{b}_n = \frac{1}{N}\sum_{n=1}^{N} \mathcal{S}_{-n v_0} \sum_{\ell=1}^{D} \mathcal{S}_{n v_{z_\ell}} K_{z_\ell,a}\, \mathbf{t}_\ell = \sum_{\ell=1}^{D} K_{z_\ell,a} \left(\frac{1}{N}\sum_{n=1}^{N} \mathcal{S}_{n v_{z_\ell} - n v_0}\right) \mathbf{t}_\ell \tag{4.15}$$

The last step in the expression above is a consequence of both K and $\mathcal{S}$ being convolutions, and therefore commute with each other. Hence, we observe that the focused measurement is identical to the single image, multi-depth model of equation (4.5) with the key difference that the texture at depth z_ℓ is now blurred by multiple translations:

$$\mathbf{t}^*_\ell = \left(\frac{1}{N}\sum_{n=1}^{N} \mathcal{S}_{n v_{z_\ell} - n v_0}\right) \mathbf{t}_\ell.$$

Hence, while the depth z_0 corresponding to the disparity v_0 observes no blurring, other depths are progressively blurred depending on the values of N, Δ and v_0.

4.3.4 Reconstruction from Full Measurements

Consider a scene $\{t_1, \ldots, t_D\}$ consisting of depth planes $\{z_1, \ldots, z_D\}$, with measurements $\{b_1, \ldots, b_N\}$ obtained from masks translated in steps of Δ. We can directly solve equation (4.16), which we refer to later as 'SweepCam-full' reconstruction, with an efficient implementation of $\mathbb{K}^\top\mathbb{K}$ in D^2 convolutions. The rest of this section describes the algorithm for 'SweepCam-full' reconstruction.

We model the forward process as a sum of convolutions,

$$\begin{bmatrix} \mathbf{b}_1 \\ \vdots \\ \mathbf{b}_N \end{bmatrix} = \begin{bmatrix} K_{z_1,a_1} & \cdots & K_{z_D,a_1} \\ \vdots & \ddots & \vdots \\ K_{z_1,a_N} & \cdots & K_{z_D,a_N} \end{bmatrix} \begin{bmatrix} \mathbf{t}_1 \\ \vdots \\ \mathbf{t}_D \end{bmatrix} \equiv \mathbf{SK} \begin{bmatrix} \mathbf{t}_1 \\ \vdots \\ \mathbf{t}_D \end{bmatrix} \equiv \mathbf{A} \begin{bmatrix} \mathbf{t}_1 \\ \vdots \\ \mathbf{t}_D \end{bmatrix}, \tag{4.16}$$

where $\mathbf{K}$ is a block-diagonal matrix containing $D \times D$ blocks, and the (d, d) block effectively convolves with PSF at depth z_d; $\mathbf{S}$ is a matrix containing $N \times D$ blocks, and the block (n, d) effectively shifts by nv_{z_d} (i.e., convolves with $\delta(x - nv_{z_d})$).

Let us consider the following least squares problem with an ℓ_2-norm regularization term:

$$l(\mathbf{t}) = \|\mathbf{A}\boldsymbol{t} - \mathbf{b}\|_2^2 + \frac{\lambda}{2}\|\mathbf{t}\|_2^2. \tag{4.17}$$

We can write the solution in the closed form as

$$\mathbf{t} = (\mathbf{A}^T\mathbf{A})^{-1}\boldsymbol{A}^T\mathbf{b}, \tag{4.18}$$

which can be computed using an iterative method like conjugate gradients by supplying the operator $\boldsymbol{A}^T\boldsymbol{A}$.

Fast implementation of convolutions. To achieve a well-determined system, the number of measurements N should be equal to or greater than the number of depth planes D. Applying $\boldsymbol{A}$ and $\boldsymbol{A}^T$ separately requires $2ND$ convolutions, while applying $\boldsymbol{A}^T\boldsymbol{A}$ directly requires D^2 convolutions and $2ND > D^2$. We implement the (i, j)-th block of $\boldsymbol{A}^T\boldsymbol{A}$ as a convolution in the following manner:

$$\begin{aligned}(\mathbf{A}^T\mathbf{A})_{ij} &= \sum_p (\mathbf{A}^T)_{ip}(\mathbf{A})_{pj} = \sum_{p=1}^{N} (\mathbf{A}_{pi})^T(\boldsymbol{A})_{pj} = \sum_{p=1}^{N} \mathbf{K}_{ii}^T\mathbf{S}_{pi}^T\mathbf{S}_{pj}\mathbf{K}_{jj} \\ &= \mathbf{K}_{ii}^T\mathbf{K}_{jj}\sum_{p=1}^{N}\mathbf{S}_{pi}^T\mathbf{S}_{pj} = \mathbf{K}_{ii}^T\mathbf{K}_{jj}\sum_{p=1}^{N}\mathcal{S}_{-p\nu_{z_i}}\mathcal{S}_{p\nu_{z_j}} = \mathbf{K}_{ii}^T\mathbf{K}_{jj}\sum_{p=1}^{N}\mathcal{S}_{p(\nu_{z_j}-\nu_{z_i})}.\end{aligned}$$

Thus we can implement $(\boldsymbol{A}^T\boldsymbol{A})_{ij}$ operator as a convolution with a kernel k_{ij}. If the PSF at depth z_i, z_j are a_i and a_j, respectively, then k_{ij} is a kernel formed by cross-correlating a_j with a_i, and then summing its copies translated by $p(\nu_{z_j} - \nu_{z_i})$, for $p = 1, \ldots, N$.

4.3.5 Reconstruction from Focused Measurements

When N and Δ are designed well, the effect of the focusing operation is to make the focused image measurement depend minimally on all depths, except one. This allows us to decouple the optimization problem of joint texture recovery on D depths, and solve D deconvolution on individual depth planes instead; this results in very fast recovery.

Without loss of generality, let's consider the effect of focusing on the closest depth z_1, with disparity ν_1.

$$\mathbf{f}_{\nu_1}(x) = K_{z_1,a}\mathbf{t}_1 + \sum_{\ell=2}^{D} K_{z_\ell,a}\mathbf{t}_\ell^*.$$

If we had an inverse in the form of a deconvolution kernel $K_{z_1,a}^{-1}$, then we can obtain an estimate

$$\widehat{\mathbf{t}}_1 = K_{z_1,a}^{-1}\mathbf{f}_{\nu_1} = \mathbf{t}_1 + \underbrace{\sum_{\ell=2}^{D} K_{z_1,a}^{-1}K_{z_\ell,a}\mathbf{t}_\ell^*}_{\text{(reduced interference)}}. \tag{4.19}$$

This decoupling of the inverse problems associated with each depth vastly reduces the complexity of the recovery procedure. Figure 4.3 shows the effect of focusing on the interference terms.

The suppression of interference due to focusing leads to an algorithm, that we call 'SweepCam fast', where we implement $K_{z_l,a}^{-1}$ by Wiener deconvolution for its speed. We compute it as

$$\widehat{\mathbf{t}}_\ell = K_{z_l,a}^{-1}\mathbf{f}_{v_\ell} = \mathscr{F}^{-1}\left(\frac{\kappa_\ell^*\mathscr{F}\left(\mathbf{f}_{v_\ell}\right)}{|\kappa_\ell|^2\mathscr{F}\left(\mathbf{f}_{v_\ell}\right)+\lambda}\right), \tag{4.20}$$

where $\mathscr{F}(\cdot)$ is the Fourier transform operator and κ_ℓ is the Fourier transform of PSF at depth z_ℓ.

Comparison between 'full' and 'fast'. 'SweepCam fast' and 'full' offer two distinct operating points. While the 'full' algorithm provides a more accurate solution by accurately modeling the inter-plane interference, it is computationally expensive. In Section 4.5.3, we consider a scene with 34 depth planes, each with 600×960 spatial resolution. For this scene, the 'full' reconstruction algorithm requires solving a problem with 19.58 million unknowns and further, each application of the forward operator or its adjoint involves $34^2 = 1156$ convolutional operators with fairly large (300×300 pixel) kernels. In contrast, the 'fast' algorithm deconvolves each depth plane in isolation, each of which only requires Fourier-domain filtering that is computationally light. This enables us to reconstruct otherwise infeasible volumes with dense depth planes, at the cost of the model misfit introduced by the interference term; however, the use of the focusing operator suppresses this interference and permits a robust solution to the inverse problem.

4.4 Properties of SweepCam

To find optimal hardware design and operating parameters for SweepCam, we analyze how various parameters affect the properties of SweepCam.

4.4.1 Spatial Resolution

Let p be the smallest feature size on the programmable mask, which is the pixel pitch of spatial light modulator in our prototype. The continuous attenuation function can be written as

$$a(x) = \beta(x) * \mathrm{rect}\left(x/p\right), \tag{4.21}$$

with some discrete pattern $\beta(x) = \sum_k \beta_k \delta\left(x - pk\right)$. Combining equations (4.3) and (4.21), we observe that the effective PSF at depth z is given as

$$\widetilde{k}_z(x) = \beta\left(\frac{z}{z+d}\,x\right) * \mathrm{rect}\left(\frac{z}{(z+d)p}\,x\right). \tag{4.22}$$

Thus, resolution at depth z is limited by the first null of $\mathscr{F}\left(\mathrm{rect}\left(\frac{z}{(z+d)p}\,x\right)\right)$, which occurs at the frequency $z/((z+d)p)$. Spatial frequencies of the texture, at depth z, outside of this cutoff can not be

reliably reconstructed. On our prototype with mask pitch p = 36um and d = 1.31cm, the resolution limit is 20.14 line pairs per millimeter (lp/mm) at a depth of 2cm, and 32.90 lp/mm at 1m.

4.4.2 Effects of Δ and N

We next analyze the dependence of the reconstruction on the number of captured images N as well as the amount of translation Δ, between each capture.

A closer examination of $\beta(x)$ from equation (4.13) in the frequency domain shows the effect of Δ and N at suppressing interference from other depth. Let us re-number translated patterns for $n = 0, \pm 1, \ldots, \pm(N-1)/2$ for odd N. Then, focusing with a disparity v_{z_0} modifies PSF of points at depth z by

$$\beta_z(x) = \frac{1}{N} \sum_{n=-\frac{N-1}{2}}^{\frac{N-1}{2}} \delta\left(x - n\Delta d\left(\frac{1}{z_0} - \frac{1}{z}\right)\right).$$

The Fourier transform of $\beta_z(x)$,

$$\beta_z(\omega) = \frac{1}{N} + \sum_{n=1}^{\frac{N-1}{2}} \cos\left(2\pi\omega n\Delta d\left(\frac{1}{z_0} - \frac{1}{z}\right)\right). \tag{4.23}$$

To suppress contribution from depth $z \neq z_0$ when we focus on z_0, $|\beta_z(\omega)|$ should be as small as possible on the resolvable frequencies, defined by the imager's spatial resolution at depth z_0. When $N \to \infty$, $\beta_z(x)$ is an impulse train, whose Fourier transform is also an impulse train. When N is small, $|\beta_z(\omega)|$ are N-slit diffraction patterns [Hecht, 2017]. The periodicity of $|\beta_z(\omega)|$ is determined by Δ, and decides how many peaks fit in the resolvable frequency range.

However, in practice we are constrained by a limited frame budget for capturing a scene, as well as a minimum translation defined by mask pitch, and a maximum baseline limited by mask size and the angular response of the mask and sensor pixels. The practical question is how to factor Δ and N within a limited baseline ΔN. Choosing a small N with large Δ results in secondary peaks that are outside the resolvable frequencies; this provides effective separation of measurements from depth z and z_0. Figure 4.5 shows an example of choosing different N and Δ with narrow and wide baseline.

Simulation Setup We quantized depth from 5 scenes in the 2001 Middlebury stereo dataset [Scharstein and Szeliski, 2002], so that generating many measurements with translated mask pattern is fast and scalable for our simulation. The number of depth planes we used into are listed in Table 4.2, with the threshold values for quantization. The furthest plane mapped mapped to 12.7 cm.

Additionally, we pad each scene with zero boundary so that contribution from each pixel in the scene does not go out side sensor boundary with maximum amount of translation of $p$mm on the mask in each

Table 4.1: **Depth quantization thresholds used on Middlebury dataset for simulations.**

scene	depth planes	quantization thresholds
sawtooth	3	[0.15, 0.285]
bull	3	[.1437, .2440]
tsukuba	7	[0.3138, 0.3766, 0.4393, 0.5021, 0.6276, 0.6903]
poster	6	[.13735 .23 .3 .44 .51365]
venus	3	[.1453 .2549]

direction. The ratio of the scene occupying the field of view is calculated by

$$1 - 2p\frac{d + z_{min}}{wz_{min}}, \tag{4.24}$$

where d is sensor to mask distance, z_{min} is the depth of closest plane in the scene, and w is the width of sensor in mm. The maximum amount of translation is calculate for all operating points in each plot, with the maximum being 96 LCoS pixels, or equivalently $p = 3.45$mm.

Photon noise and read noise are simulated in all the measurements with parameters taken from the sensor used in our hardware experiments, Sony IMX174, with full well capacity $F = 30500$ electrons and $R = 71.7$dB.

Performance with different number of measurements We evaluate in simulation how image quality of SweepCam changes over number of measurements N in Figure 4.6(a). We simulate photon and read-out noise in our measurements using sensor parameters from our prototype. For each method, we report the best structural similarity index (SSIM) score for the all-in-focus scene, generated by compressing the 3D volume using the ground truth depth map, across different regularization parameter $\lambda \in \{0.001, 0.01, 0.1, 1, 10\}$.

Figure 4.6(a) compares the performance of SweepCam fast and full reconstructions against a static mask; for fairness in comparisons, we repeat and average the static mask measurements so that the number of measurements for all three methods is the same. The baseline is kept the same, at a spread of 96 pixels, while the number of measurements is changed; further, the translation of masks is purely horizontal. While averaging multiple static measurements mitigates noise, it does not change the SSIM

score significantly. SweepCam-full reconstruction is severely under-determined for single measurement, but improves as number of measurements increases, and peaks when number of measurements match number of unknown depth planes. SweepCam-fast reconstruction has increasing SSIM as the number of measurements increase, since the interference between depth planes is reduced due to focusing.

Performance at different baselines Figure 4.6(b) shows the performance of the 'SweepCam-fast' and 'SweepCam-full' reconstructions over different baselines $N\Delta$. We perform this by keeping the number of measurements fixed at $N = 9$ and varying Δ. As we expect, the reconstruction accuracy of both techniques increase with increasing baseline.

Qualitative performance We also show qualitatively the reconstruction performance of the techniques in Figure 4.6. Here we show SweepCam at three operating conditions: *p*arameter A as a default setting, *p*arameter B as a setting with fewer measurements and *p*arameter C with small baseline. We also show the reconstruction from the static mask for comparison. With *p*arameter B, the texture suffers from the reconstruction artifacts, which is caused by the interference from other depth planes, as discussed in Section 4.4.2. By comparing *p*arameter A and *C*, we find that the small baseline also makes it difficult to reconstruct.

4.4.3 Arranging Aperture Locations in 2D

We also evaluate the effect of sweep pattern, i.e. the spatial arrangement of aperture locations on the 2D mask. We use two different types of arrangements, 1D and 2D, where both of them use the same number of measurements (e.g. 9×1 versus 3×3 for N=9) for the uniform step sweep patterns under the same baseline of 96 pixels. Three scenes from the dataset of [Scharstein and Szeliski, 2002] are employed for the evaluation.

The effect of spatial arrangement is scene dependent, as shown in Figure 4.7. For the Bull scene, which has a vertical and horizontal edges in its depth variations, 2D arrangement performs better than 1D with sufficient number of measurements ($N \geq 25$). This is because 2D arrangement can effectively mitigate the cross-plane interference over edges with various angle, while 1D arrangement acquires only the horizontal parallax. Although 1D arrangement scores better than 2D for Sawtooth and Tsukuba, this is because these scenes have mainly vertical edges and thus fewer measurements of 2D for the horizontal parallax causes the performance drops. In practice, we do not have prior knowledge on the scene, therefore it is desirable to acquire 2D measurements with sufficient sampling along both directions.

4.4.4 Length of M-sequence

Length of M-sequence affect the area of aperture, which determines the light efficiency of proposed camera. Therefore, we evaluate how the length of M-sequence effects on the quality of reconstructed image in simulation. We observe the performance transition by using various types of M-sequence, whose size is 15, 31, 63, 127 and 255 respectively, while other experimental setups follow those of *p*arameter A. The measurement noise is applied considering the light efficiency which is decided by the size of each aperture size.

The result is shown in Fig. 4.8, with the averaged scores among 5 different scenes, and errorbar showing the standard deviation for each M-sequence length. We can observe that we have a peak on SSIM score at the length of 63 and 127. The transition is not monotonic due to two conflicting nature on the size of aperture. The longer the length of M-sequence is, the flatter the power spectrum becomes, which is desirable for the reconstruction performance. But this theory holds when we can ignore the effect of sensor boundary. In practice, too large an aperture leads to the performance deterioration since a significant portion of the measurement is cropped at the sensor boundary.

4.4.5 Depth Resolution

Depth of scene points are inferred from their difference in disparity in SweepCam measurements. The change in disparity in sensor pixels as a result of change in depth can be computed from equation (4.11),

$$\partial v = d\Delta \;\; \partial(1/z). \tag{4.25}$$

Since focusing provides explicit control over disparity, we observe that SweepCam, much like other depth estimation techniques, resolves depth uniformly in diopters or in $1/z$ space. A uniform sampling in diopters results in a depth tiling that is highly non-uniform, with a dense sampling of depth in close proximity to the device and very sparse sampling at far away depths. Further, the resolution in diopters is inversely proportional to the mask-to-depth distance d. For example, when $d = 2$mm, a focus disparity in the range of $[10, \infty)$ pixels maps to a depth range $z \in (0, 1]$ mm; in contrast, when $d = 13.1$mm, the same focus disparity range maps to a depth range $z \in (0, 10]$ mm.

4.4.6 Field of View

SweepCam aims to recover an image formed by a pinhole placed at the center of the mask, d away from the sensor. The field of view of the reconstructed image is given by

$$2\tan^{-1}(s/(2d)).$$

Table 4.2: **Average run time and quality comparison between reconstruction methods.** The experiments operate under *param A* of Figure 4.6

Reconstruction technique	Run time in sec.	SSIM
Static mask fast	2	0.46
SweepCam-fast	3	0.66
SweepCam-full	1635	0.67

On our prototype with d = 1.31 cm and s = 0.71 cm, it sees about 30°. In addition to the geometric spacing of the mask and sensor, the field of view is also limited by the combined effects of mask attenuation and sensor pixel angular response.

4.4.7 Computational Time

The average run time and average SSIM over all scenes operating with *parameter A* is shown in Table 4.2. SweepCam fast reconstruction achieves better quality than static mask reconstruction with similar run time, and runs two orders of magnitude faster than full volume reconstruction with a small loss in quality. The reduced run time of the 'fast' algorithm can be traced to the decoupling of reconstruction at different depths.

4.4.8 Light Efficiency

Light efficiency of SweepCam is primarily dependent on the size of the coded aperture. However, when the aperture is too large, the convolutional model underlying SweepCam is violated due to the cropping of the mask boundary by the finite sensor. Hence, we trade-off light efficiency for the simplicity of the convolution model and choose the largest aperture for which the model holds reliably. For the experiments with our prototype, we use an aperture of size 2.27mm, within which half of the light is blocked.

Performance under noise. We simulate different sensor noise on SweepCam and static mask measurements, and compare their performance quantitatively in Figure 4.9. We scale the maximum measurement to different percentage of full well capacity of the sensor, and reconstruct from measurements

with different amount of noise. Photon noise and read noise are generated via

$$\widetilde{\mathbf{b}} = \frac{G}{F}\left(\text{Poisson}\left(\frac{F}{G}\mathbf{b}\right) + \text{Normal}(0, \sigma^2)\right), \tag{4.26}$$

where F is full well capacity of the sensor, gain G is one over light level, and $\sigma = F \times 10^{-R/20}$ with R being the dynamic range. As shown in Figure 4.9, SweepCam methods are more robust to measurement noise induced by low light level. SweepCam averages out non-idealities such as dust particles and dead pixels in focused measurements, since light from each scene point is observed multiple times at different pixels.

4.4.9 Reconstruction with Different Priors

Finally we show simulation results for different reconstruction methods in Figure 4.10. We implemented solutions for traditional least squares as well as canonical and wavelet sparsity by choosing appropriate regularizing penalty function $\rho(\cdot)$ in equation (4.6). We do this for solving single depth planes separately as well as for the whole volume simultaneously. For the least squares solutions, we use Wiener deconvolution when working with individual depth planes and the conjugate gradient squared method for volume reconstruction. Sparse priors, both in the canonical and wavelet bases, were implemented using backtracking FISTA [Beck and Teboulle, 2009] with initialization at the zero solution. While more sophisticated image prior result in sharper reconstructions, we observe that this comes at a cost of increased runtime.

4.5 Experiments on Hardware Prototype

We conduct several experiments on hardware prototype to address details in implementation as well as validate the proposed model.

Figure 4.13 shows the prototype hardware. It consists of two parts: a programmable amplitude mask and a image sensor. The programmable amplitude mask consists of a Holoeye LC2012 spatial light modulator sandwiched between two cross polarizers, one of which is placed on a precision rotation stage to maximize contrast. Our prototype's amplitude mask has a effective contrast ratio of 200:1, pixel pitch of 36µm, and fill factor of 58%. We use a Sony IMX174 RGB sensor in our prototype; its pixel pitch is 5.86µm. We calibrate for angle between programmable mask and sensor, PSF at different depth, and distance between mask and sensor after building the prototype.

Angle between programmable mask and Sensor. The focusing operator requires knowledge of the direction of mask rows and columns in the sensor coordinate. We try to align the mask parallel to the sensor, and estimate those directions via calibration. An LED is placed before the mask, and an image is captured when each row of the mask is turned on to transmit light. The direction of mask rows in sensor coordinate can be calculated from the lines detected in those images. The direction of mask columns in sensor coordinate is similarly obtained.

Point spread function (PSF). We display a pattern on the mask and capture its PSF by moving a point light source, an LED, on a rail for different depth. Two more measurements were captured while translated patterns were displayed, and those were used to produce a focused image, which is cropped and used for reconstruction. We capture these images at six depths, and obtain the PSF at other depths by scaling the image captured at the nearest depth.

Distance between mask and sensor. The distance between mask and sensor can be solved from the scale of the PSF captured at different depths. We calibrate by setting up a ruler rail on the z-axis of the camera, moving a point light source at $z_1, \ldots, z_m$ on the rail, and recording the physical mask size l as well as corresponding PSF size $l_1, \ldots, l_m$. The first measurement gives a equation from similar triangle,

$$\frac{z_1 + z_0}{l} = \frac{z_1 + z_0 + d}{l_1}. \tag{4.27}$$

The distance between mask and sensor, d ,and the distance from start of ruler to mask, z_0, can be solved from the equation formed from m similar triangles for $m \geq 2$,

$$\begin{bmatrix} l_1 - l & -l \\ \vdots & \vdots \\ l_m - l & -l \end{bmatrix} \begin{bmatrix} z_0 \\ d \end{bmatrix} = \begin{bmatrix} -(l_1 - l)z_1 \\ \vdots \\ -(l_m - l)z_m \end{bmatrix} \tag{4.28}$$

The distance between mask and sensor affects observed disparity from the same depth. Calibration on our prototype yields 1.31cm between mask and sensor; its disparity from depth is plotted in yellow dashed line in Figure 4.11. Decreasing the distance will make the prototype more suitable for microscopic applications.

Validation of convolutional model. We validate the convolutional model by placing an 8×8 LED in 1 inch array 5cm in front of the hardware prototype, displaying a mask pattern that is the outer product

of M-sequence of length 31, and capturing a measurement while one LED is turned on for each LED in the odd rows in the array. We annotate the center of PSF from LED in row 5 column 5, predict the center of PSFs in other measurements based on disparity, and crop patches with those predicted centers. Those patches are shown in Figure 4.12. The maximum value exceed 1 because cubic interpolation is used. The difference between patches extracted from other measurements and that from LED in row 5 column 5 is shown on the bottom image in Figure 4.12. The small intensity in difference verify that translating a point light source results in a measurement with corresponding translated PSF, and the convolutional model holds.

Other details. Unless noted otherwise, all SweepCam results included are produced with 13×13 aperture locations; the aperture codes are outer product of M-sequence of length 63. The positive and negative parts are separately captured and subtracted computationally. Static masks comparisons are produced with the same number of captures but without changing the mask pattern.

4.5.1 Scenes with Two Depth Planes

We now show results on a real scene captured with our hardware prototype in Fig. 13. The scene consists of two printed transparencies, in Figure 4.14(a). With static mask measurements, directly deconvolving with PSFs at near and far planes as [Asif *et al.*, 2016] results in artifacts in reconstruction, as shown in Figure 4.14(b). Figure 4.14(c) shows the reconstruction of Asif [2018], a technique that estimates both the depth map and textures jointly. We also report results of texture estimation using Asif [2018] when the depth at each pixel is known; this is shown in Figure 4.14(d). Finally, Figure 4.14(e) shows the SweepCam reconstructions, which provides the highest quality results with the least artifacts.

4.5.2 Resolution Chart on Two Planes

We image two printed USAF charts located at different depth to demonstrate how SweepCam improves of resolution of lensless images, shown in Figure 4.15. The near chart is 6.6cm away containing group 0 and 1; the far one is 28 cm away containing group from -2 to 1. The static mask reconstructions is able to resolve 1.78 lp/mm on near chart and 0.44 lp/mm on far chart. The SweepCam full and fast reconstructions resolve 2.24 lp/mm on near chart and 0.70 lp/mm on far chart, as they can distinguish contributions from different depth planes.

4.5.3 Continuous Depth Scenes

We image objects with dense textures and continuously-varying depth profiles, as shown in Figure 4.16. The three objects correspond to a tilted plane, a corner of a box, and a cylinder. A focal stack with 600×960 spatial resolution and 3 channels and 34 depth planes is generated within 8 minutes with MATLAB code running on 12 core CPU following the reconstruction described in equation (4.19) thanks to the decoupling of depths provided by the SweepCam measurements. Without decoupling of depth, solving the full estimation problem would result in larger difference in reconstruction time than that shown in Table 4.2 because of the increase in the number of depth planes. The full focal stacks can be found in our supplementary video.

Additionally, Figure 4.16 shows depth map recovered by using depth-from-defocus algorithms on the SweepCam reconstructions, in comparison to that from static measurement reconstructions.We assign each pixel to the depth plane where the local contrast of textures reaches its maximum value as we sweep across focus planes. Additionally, we show result from joint estimation of texture and depth [Asif, 2018] with 10 depth planes in the depth range for comparison. The depth of the textured regions are correctly resolved for SweepCam reconstructions because interference from other depths are suppressed at high frequencies as explained in Section 4.3.3.

4.5.4 General Scenes

SweepCam is able to resolve general scenes with depth variation as shown in Figure 4.1 and 4.17. Figure 4.17 shows some challenging scenes that deviate from the convolutional model. While some artifacts are produced by the model mismatch, the SweepCam reconstructions can still resolve content reasonably at each depth.

4.6 Discussion

We present a method for distinguishing depth of scene points on lensless imagers using a translating mask implemented using a programmable LCoS device.

Occlusion modeling. Consider the light cone that a scene point casts on the sensor. In the presence of occlusions, each scene point will have a different visibility to the sensor and this breaks the shift-invariance of the convolution. One way of modeling occlusion is to introduce a visibility term [**??**]. We could augment equation (4.2) in our forward model to be

$$b(x) = \int_{x_0} t_0(x_0) v_0(x, x_0)\, a\left(x + \frac{x_0 - x}{z_0 + d} d\right) dx_0. \tag{4.29}$$

where $v_0(x, x_0)$ indicates visibility of scene point at (x_0, z_0) from sensor location x. In addition to $v_0(x, x_0)$ being high dimensional, solving for both t_0 and v_0 is no longer a linear problem. However, an important benefit of this modeling is that secondary effects that break the convolution model, including occlusion and specularity, can be accounted for in the ensuing non-linear optimization. This would invariably require iterative solutions and good initializations; perhaps SweepCam-fast results can serve as a good initialization point since it is fast to compute.

Loss of time resolution. The main limitation of using programmable mask in lensless cameras arise from the fact that multiple images need to be captured corresponding to multiple modulation pattern. Capturing multiple images introduce limitations such as long capture time, low frame rate, and the inability to deal with moving scenes; however, this is a well-studied problem with potential solutions that can borrowed from research on multi-image fusion [Ma *et al.*, 2017].

Limitations of the implementation. Our prototype implements the programmable amplitude mask with a transmissive LCoS. Its limited contrast ratio results in a low SNR in captured measurements; its large pixel pitch limits the spatial resolution and depth resolution of the imager as discussed in Section 4.4. Previous compressive temporal imagers [Llull *et al.*, 2013] have used translating mask for time-modulation and use this to obtain improved time resolution. The proposed design can be similarly implemented by piezo actuators for mechanically translation of a mask on film or glass, which comes with higher contrast ratio, smaller minimum feature size, and finer control over translation.

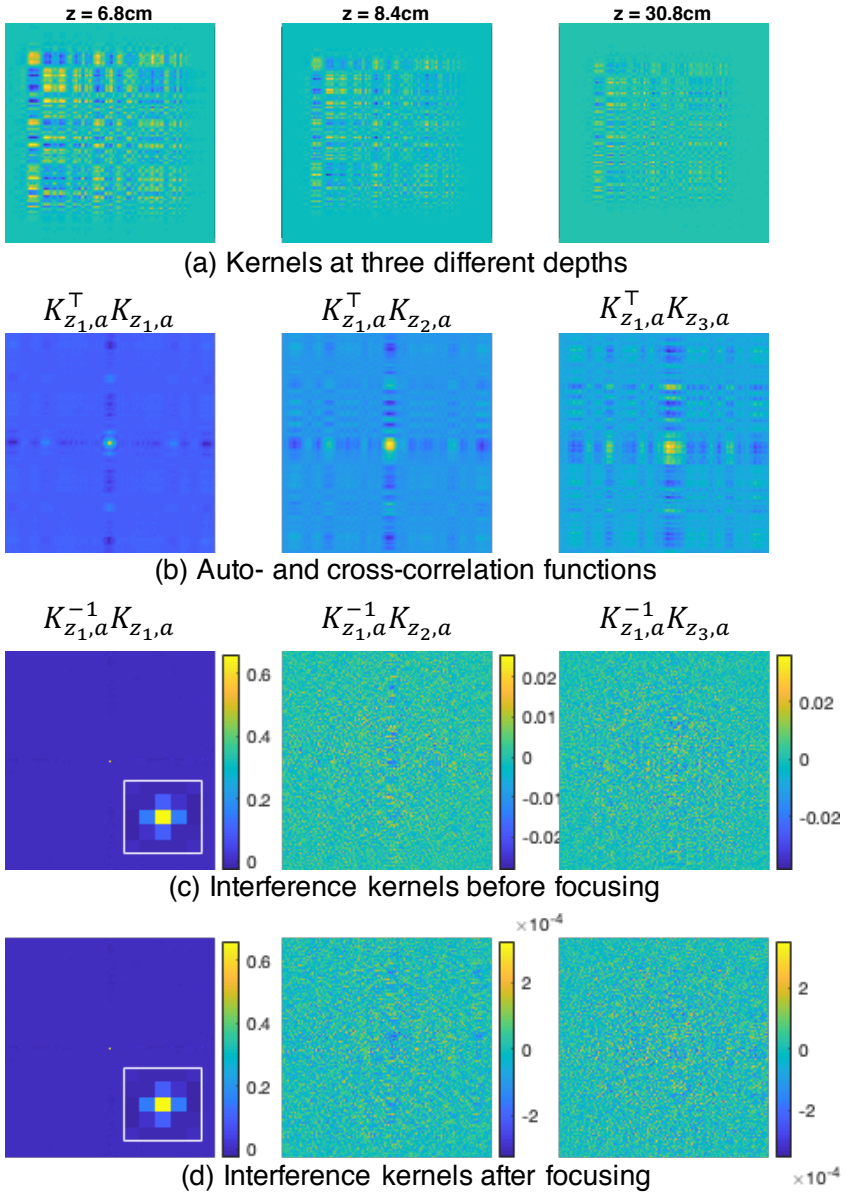


Figure 4.3: **Kernels and their evolution.** Top row shows PSF for three different depth. Second row shows each PSF correlated with PSF from $z_0 = 6.8$cm, as kernels underlying blocks in the Gram matrix from Section 4.3.1. Third row shows applying deconvolution kernel for PSF at z_0 on PSFs of different depth; the result is high frequency artifacts for directly applying deconvolution kernel on captured measurements. Last row shows applying deconvolution kernel for PSF at z_0 on PSFs of focused measurements; the artifacts are reduced by two orders of magnitude when reconstructing with focused measurements. Focused measurements are generated as described in Section 4.3.3 with 13×13 aperture locations across baseline area 0.78cm×0.78cm.

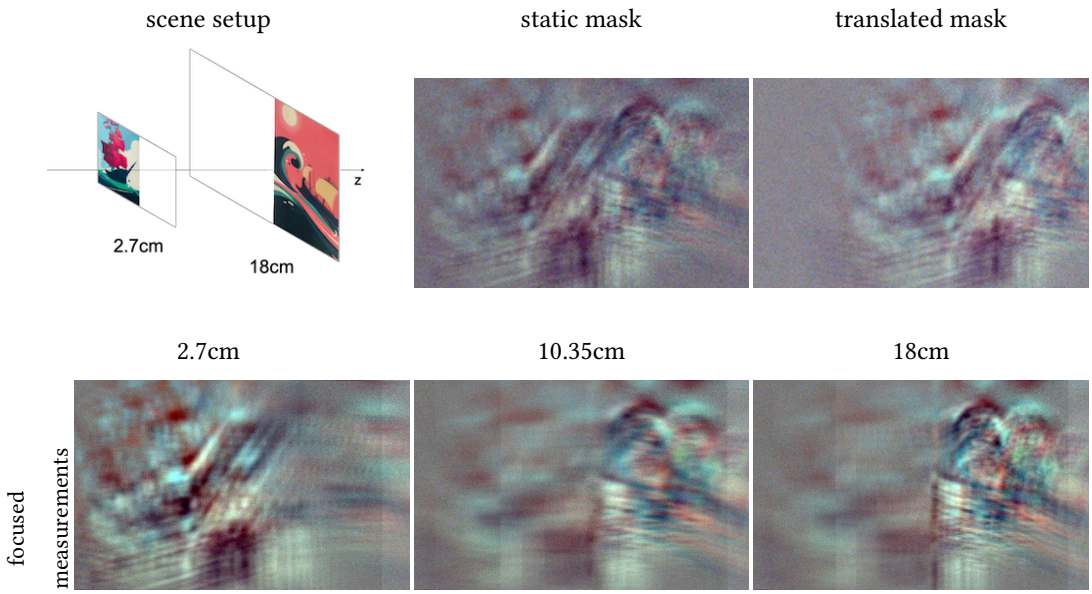


Figure 4.4: **Captured and focused measurements from our lab prototype for scene with content on two planes.** Focused measurements in both are generated with 13 captures with total baseline of 0.78cm.

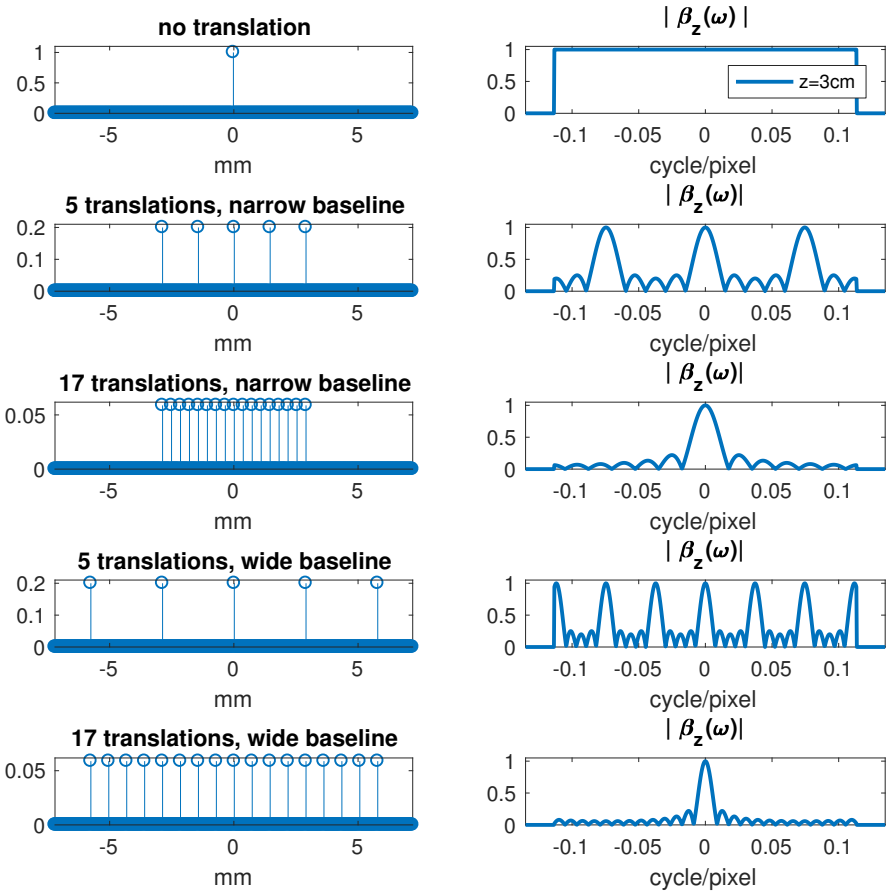
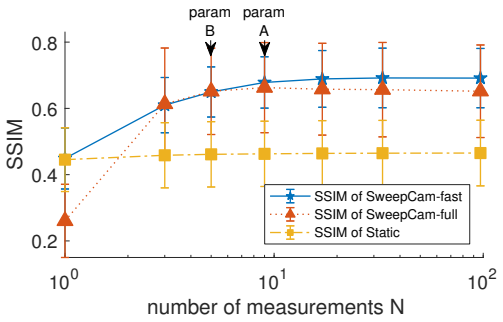
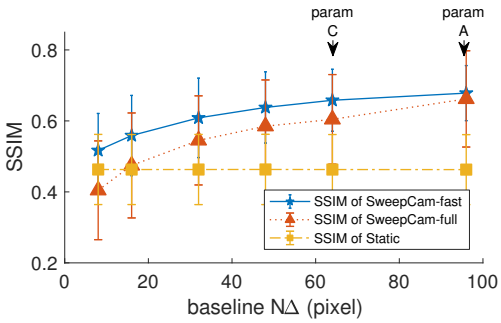


Figure 4.5: **Reducing interference from other depth via focusing..** The left column shows translation patterns of the mask, while the right column shows of $|\beta_z(\omega)|$ in equation (4.23) for depth $z = 3cm$ and $\nu_{z_0=4cm}$. Row two and three show more effective of suppression of interference from $z = 3cm$ as number of translations increase. Imaging parameters such as mask pixel pitch are taken from our hardware prototype, given in Section 4.5.



(a) Performance for different number of measurements at a fixed baseline of 96 pix



(b) Experiments over different baseline at fixed N=9

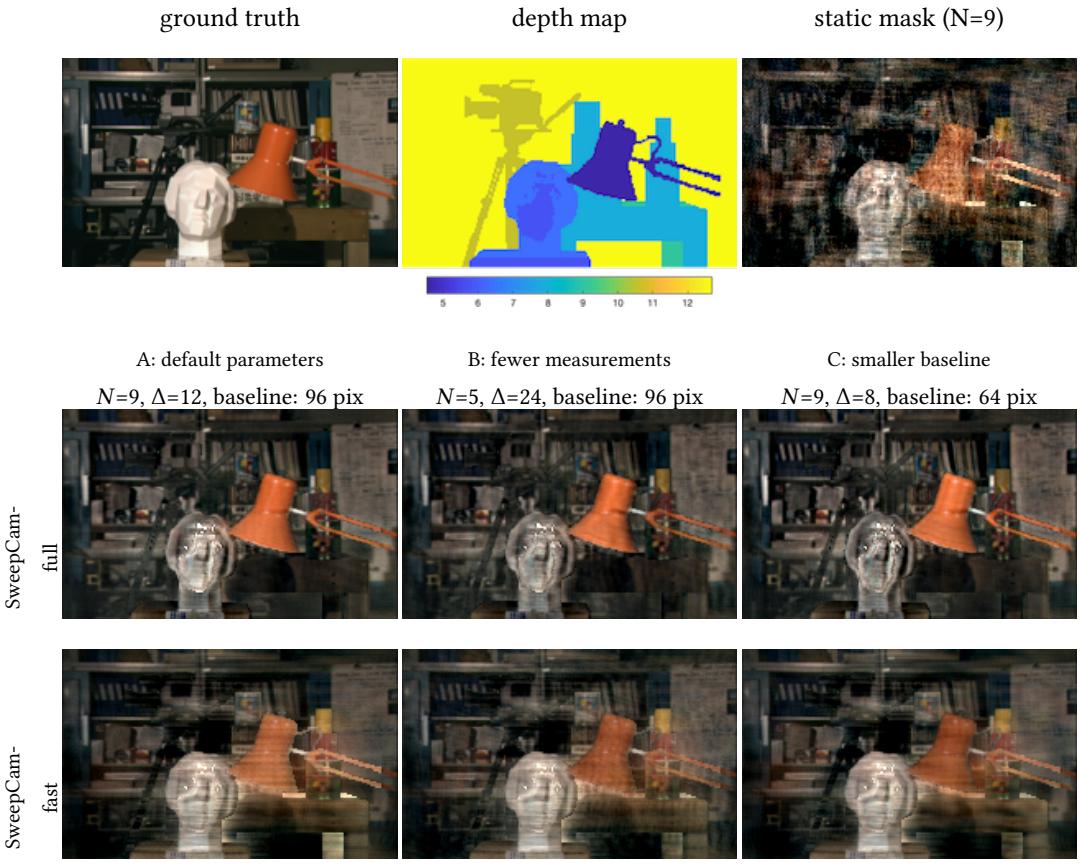


Figure 4.6: **Comparison of different number of measurements and baseline on simulated data..** Top figure quantitative evaluates reconstruction performance in terms of SSIM. (a) shows results for different number of measurements: increasing number of static measurements mitigates noise in the measurement but results in little changes in SSIM. (b) shows results for different baseline: small baseline degrades performance. Bottom images shows one of the reconstructed all-in-focus images at three operating points noted in the top plots.

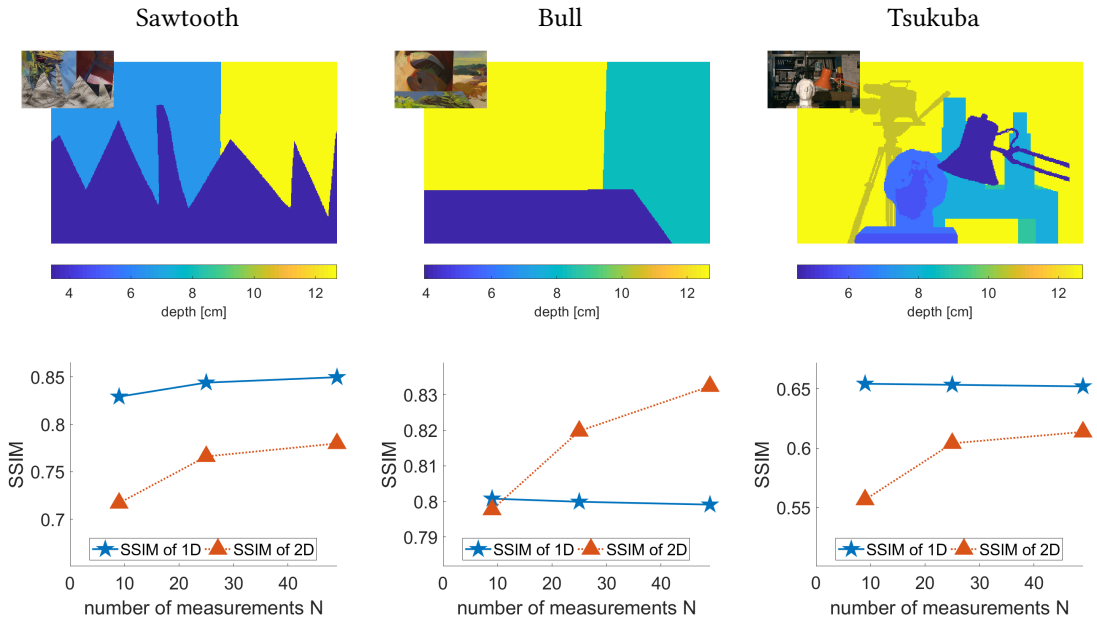


Figure 4.7: **Reconstruction quality of SweepCam with 1D and 2D sweep pattern.** Top row shows the depth maps of scenes, with their texture in insets.Bottom row shows the transition of SSIM for each scene. Effects of different sweep patterns are scene dependent.

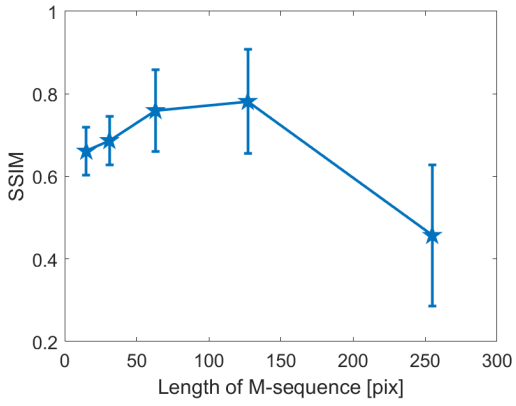


Figure 4.8: **Image quality of SweepCam over different length of M-sequence.**

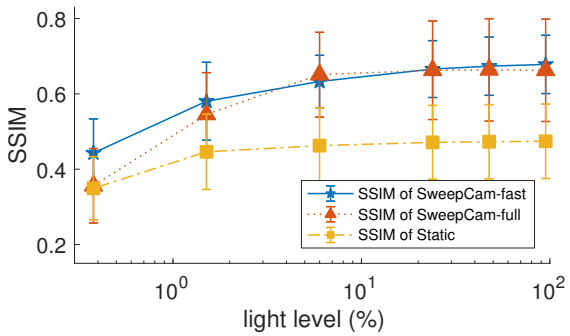
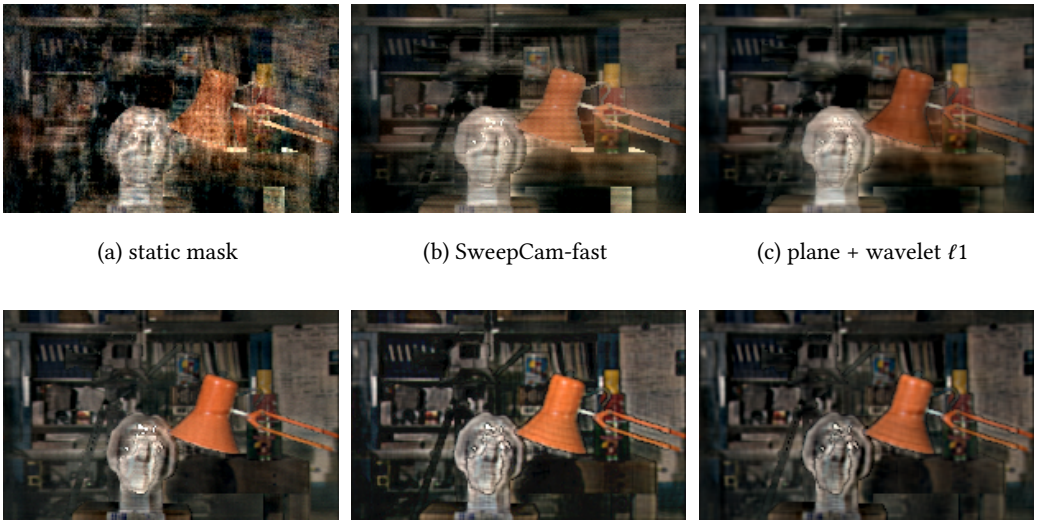


Figure 4.9: **Image quality with varying light levels.** We simulate light levels in terms of the fraction of the full well capacity at the brightest pixel on the sensor. Shot noise and read noise are simulated with sensor full well capacity and dynamic range for the Sony IMX174 sensor. We observe that SweepCam fast achieves better performance under noisy conditions.



(a) static mask (b) SweepCam-fast (c) plane + wavelet $\ell 1$

(d) SweepCam-full (e) volume + $\ell 1$ (f) volume + wavelet $\ell 1$

Figure 4.10: **Comparison of reconstructing with different image priors.** "Tsukuba" scene from Middlebury dataset (ground truth shown in Figure 4.6) is imaged at "param A" described in Figure 4.6. Here we show results of reconstruction using different image priors. Images on the top row are reconstructed at each depth plane separately following equation (4.7): using static mask measurements with ℓ_2 norm squared, SweepCam measurements with ℓ_2 norm squared, and SweepCam measurements with ℓ_1 norm of wavelet coefficients, respectively. Images on the bottom row are reconstructed from all depth planes in the volume from SweepCam measurements following (4.6), using ℓ_2 norm squared, ℓ_1 norm, ℓ_1 norm of wavelet coefficients on each image plane as priors, respectively. As shown, the SweepCam-fast algorithm achieves reasonable quality while it runs significantly faster than the other algorithms using more sophisticated priors.

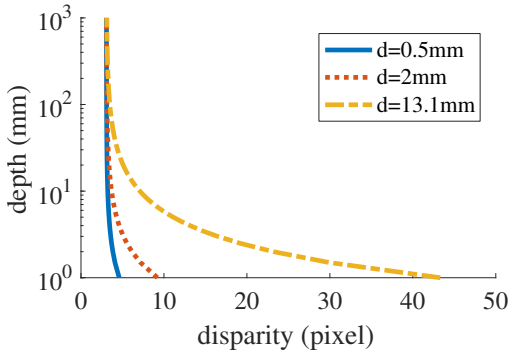


Figure 4.11: **Scene point depth v.s. disparity for different distance d.** The vertical arrows indicate range of depth corresponding to 1 pixel change in disparity. Note larger d results in an larger range of indistinguishable depth, and close depth has smaller range of indistinguishable depth.

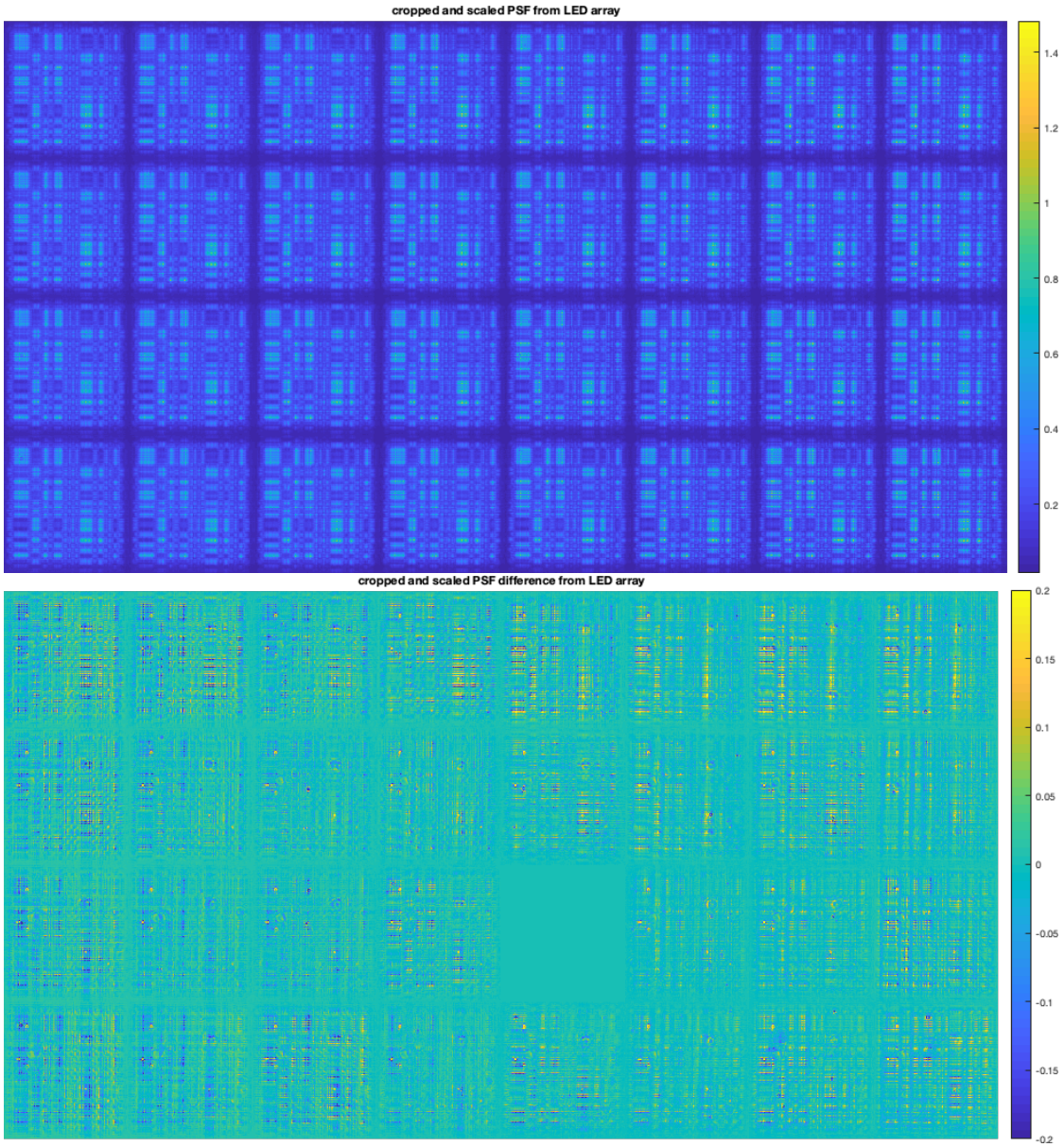


Figure 4.12: **Measurements from an LED array, aligned with predicted disparity.** Small intensity in difference image verifies the convolution model.

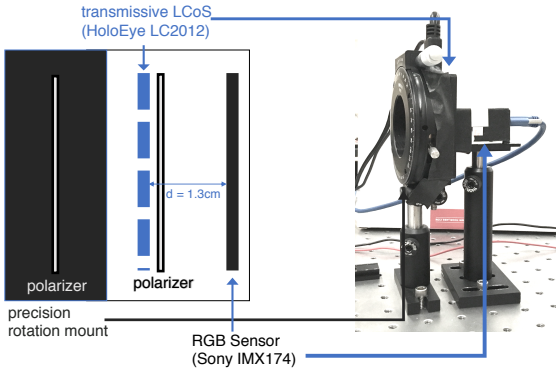
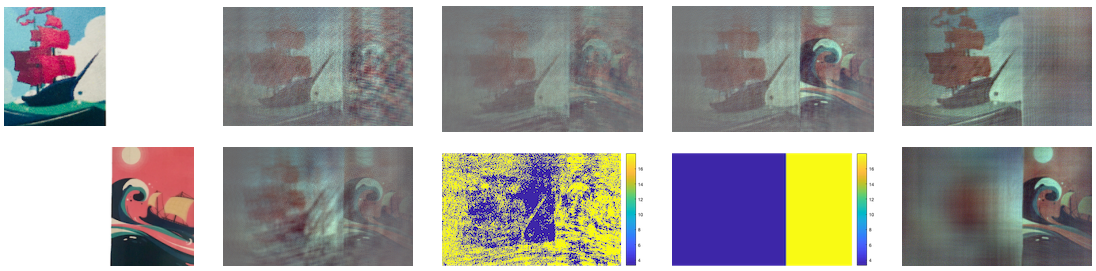


Figure 4.13: **Prototype hardware setup.** The proposed design includes a programmable amplitude mask and a sensor. Our programmable mask is made of a transmissive LCoS sandwiched between two cross polarizers, one of which is mounted on precision rotation mount of optimal contrast.



(a) scene setup (b) static fast (c) static joint (d) static texture only (e) SweepCam-fast

Figure 4.14: **Comparison of different reconstruction methods on real data.** As shown in (a), the scene contains two transparencies printed with boat pattern. White is printed to be transparent. Near plane is at 2.8cm while the far plane is at 18cm. (b)(c)(d) show various reconstruction techniques from static mask measurements. (b) deconvolves static mask measurements with PSFs at near and far planes, as Asif *et al.* [2016]; (c) jointly estimates texture and depth of each pixel in the scene, as Asif [2018]. (d) is given per pixel depth as input and only solves for texture. (e) reconstructs from SweepCam measurements with the same number of frames using the fast algorithm.

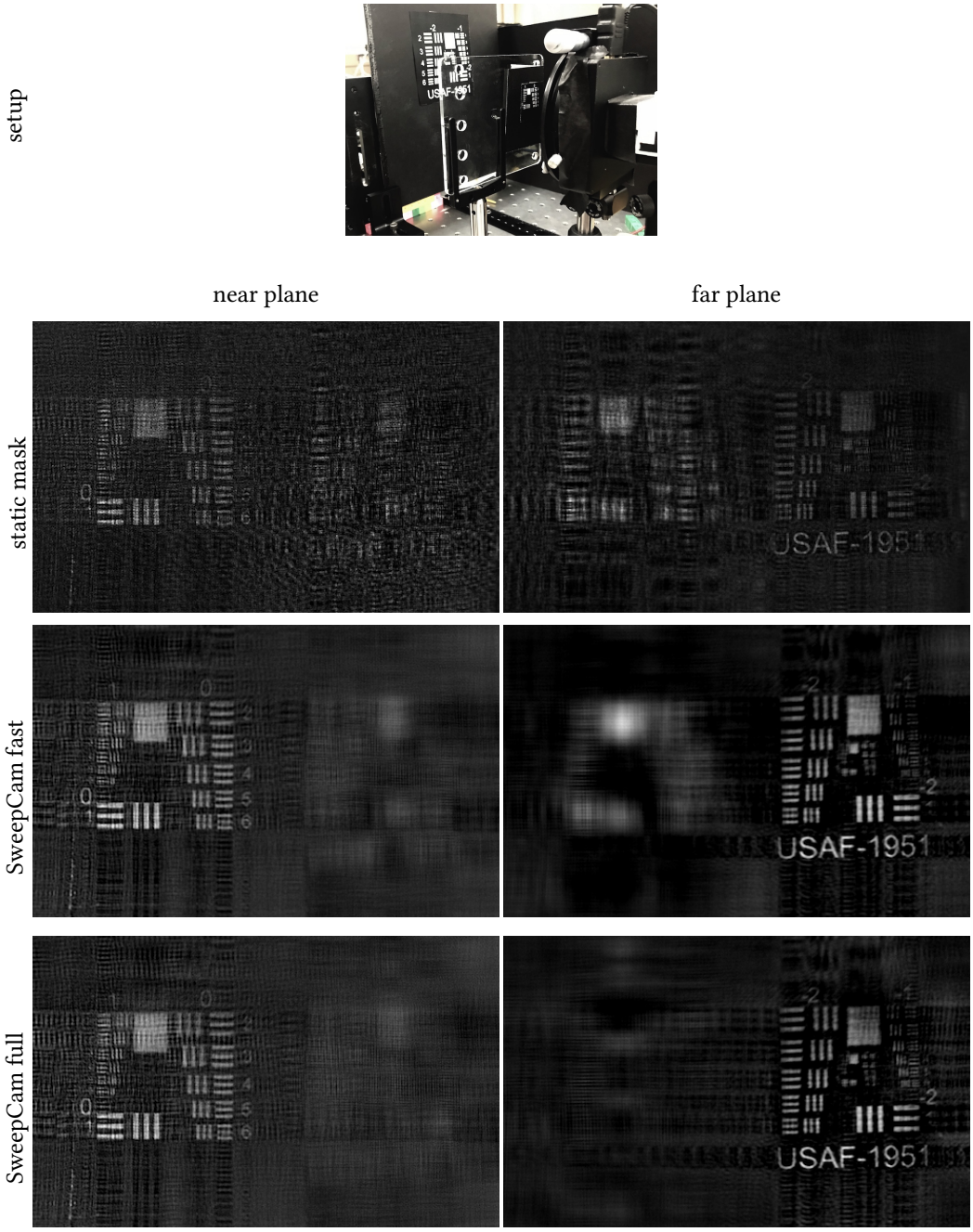
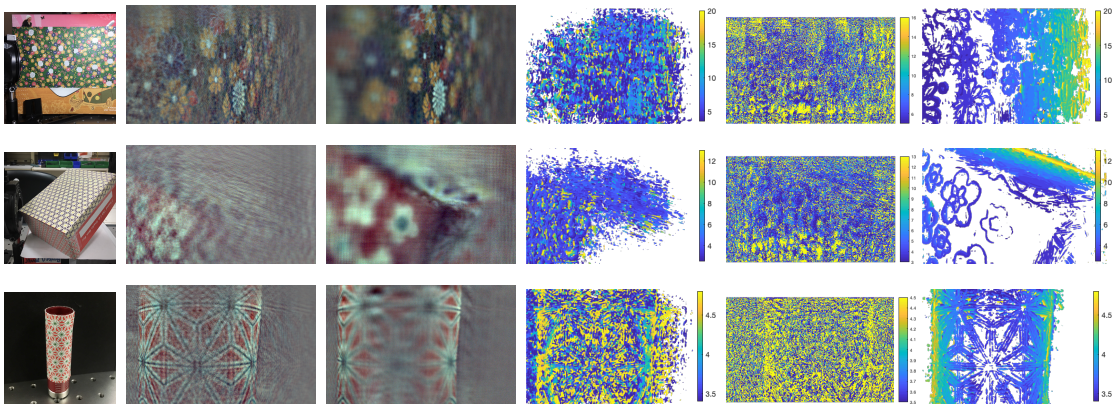


Figure 4.15: **Two USAF resolution charts at different depths.** SweepCam results are captured with 9×9 aperture locations across 0.4cm × 0.4cm.



(a) setup | (b) static mask reconstruction | (c) SweepCam-fast reconstruction | (d) static mask estimated depth | (e) Asif [2018] estimated depth | (f) SweepCam-fast estimated depth

Figure 4.16: **Estimated depth for objects with known geometry.** From top to bottom: a slanted plane, corner of a box, and a cylinder. Objects are covered with patterned paper to produce dense texture. (b)-(c) show a image from the focal stack at the same depth; column (d)-(f) show estimated depth estimated from focal stack with corresponding method. (d) and (f) estimates depth from lensless focal stacks by assigning each pixel to the focal distance with maximum local contrast. Local contrast is computed by standard deviation of pixel intensity in 11×11 neighborhood. Contrast below threshold indicates untextured region and has no depth estimation. In (e), depth is estimated as part of reconstruction algorithm in Asif [2018]. Removing high frequency artifacts in SweepCam fast reconstruction significantly improves depth estimation, as (f) demonstrate more reliable estimation against (d) and (e).

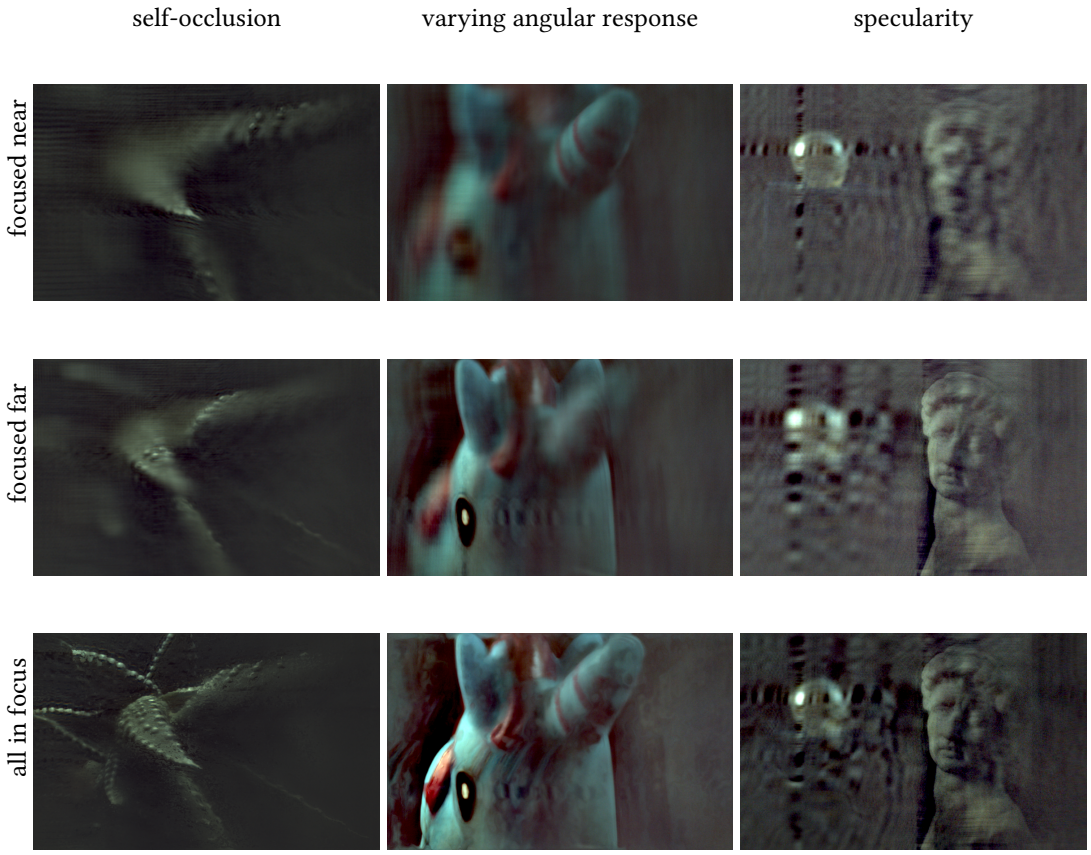


Figure 4.17: **General scenes that deviate from the convolution model.** Each of the three scenes violet the assumptions underlying the image formation model, either in the form of occlusions between depth planes, or due to materials with non-Lambertian reflectance and directional lighting in the scene. In spite of these model mismatches our techniques work reliably, except perhaps for artifacts that are spatially localized.

5 Inverse Rendering for Lensless Imaging

This chapter advances the field of lensless imaging by incorporating forward models that are more precise and better describe the image formation process. To understand the nature of our contributions, it is worth looking the state of the art in lensless imaging and identifying gaps in modeling of image formation. In a lensless camera, a sensor is placed in behind an amplitude or phase mask that modulates the light incident on it [Boominathan *et al.*, 2016]. When the scene is sufficiently far away, and the mask aperture is sufficiently smaller than the sensor area, the forward model can be written as one of convolution between the scene's angular radiance [Asif *et al.*, 2016]; this simple setting provides fast and quick inverses via the Weiner filter. However, when we perturb the assumptions underlying this model, either by considering with a scene at a finite depth range or a mask whose size is commensurate to the sensor, then we need to revisit the forward model and consequently, the inverse algorithm.

Using a larger mask changes the forward model to one of convolution followed by cropping by the sensor [Antipa *et al.*, 2018], which enjoys an efficient implementations via the FFT. In FlatCam [Asif *et al.*, 2016] and FlatScope [Adams *et al.*, 2017], a separable mask aligned with the sensor simplifies the forward model to left and right multiplication of the scene image with two smaller matrices. In both cases, the nature of the forward modeling permits the solution to be a linear inverse problem which can be solved using iterative techniques.

Incorporating scene depth variations, on the hand, has proved to be more challenging. The vast majority of lensless imaging techniques represent the scene as a volumetric albedo function [Adams *et al.*, 2017, Antipa *et al.*, 2018, Hua *et al.*, 2020]; this retains the flavor of the forward model to be linear in the scene unknowns, namely the volumetric albedo, and hence allows the use of regularized least squares techniques for recovery. The use of volumetric albedo representation does present challenges in the form of a dramatic increase in the dimensionality of the unknown signal. SweepCam [Hua *et al.*, 2020] resolves this by collecting multiple images with a translating mask. This result is extended in Zheng *et al.* [2021] where a sequence of programmable masks is designed to better condition the depth

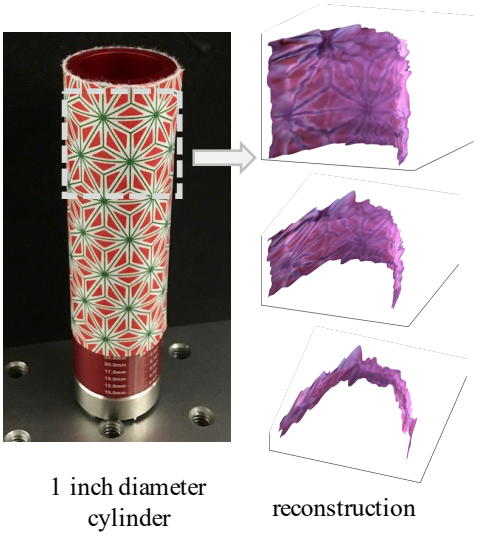


Figure 5.1: **Reconstruction of cylinder.** This chapter presents an physically-realistic and differentiable forward model for lensless imager, which results in improved reconstruction of the scene texture and depth.

recovery process. Zheng and Asif [2020] avoid the dimensionality gap by modeling the scene with a texture and depth map model.

When we consider the landscape of these techniques, there are critical gaps in forward modeling that restrict the complexity of scene that can be captured. First, for most scenes, the volumetric approximation is not physically accurate. Image formation for most opaque objects requires understanding of the surface orientation and reflectance, which are both beyond the scope of the volumetric model. Second, most prior work ignore the effects of the sensor's angular response whose influence on the measurements can be quite significant when the scene is in close proximity to the camera. When a scene point is sufficiently far away, the angle it subtends to the sensor is small and hence, the effect of the angular response can be ignored. However, when the sensor area is large and scene points are sufficiently close to the device, this term can dramatically influence the contributions arising from a scene point across the sensor. Not modeling these effects, as we show later, leads to poor reconstruction quality.

This chapter provides a new techniques for scene recovery using techniques from inverse rendering [Kato *et al.*, 2020, Marschner, 1998, Patow and Pueyo, 2003]. We base our work on recent work on the successful use of such inverse rendering techniques for scattering [Gkioulekas *et al.*, 2016], non-line-of-sight shape estimation [Tsai *et al.*, 2019] and reflectometry [Shem-Tov *et al.*, 2020]. Specifically, we use a precise forward model based on a physically-accurate and *differentiable* renderer where we model

the scene as a triangulated mesh in 3D space whose vertices encode texture properties. The forward model now involves a Monte-Carlo renderer that accurate reflects image formation including scene modeling that treats objects as surfaces with accurate modeling of foreshortening terms, and sensor effects like angular sensitivity at each pixel. The forward model is differentiable with respect to the unknown parameters of the mesh, which allows us to use stochastic gradient techniques and associated optimization toolboxes that have efficient GPU implementation. This precision of this forward model provides a significant advance in our ability to recover textures and depth from lensless cameras.

Contributions. This chapter proposes a inverse rendering approach to reconstruction which advances lensless imaging by building a more physically accurate measurement model. Our main contributions are as follows.

- *A differentiable and physically-realistic forward model.* We build a Monte-Carlo renderer using this model, which maps a 3D scene represented by triangle mesh to its lensless measurements, so that complex effects such as foreshortening and sensor's angular sensitivity can be accurately captured in the rendered measurements.
- *Reconstruction algorithm for the texture and shape of a scene using inverse rendering.* Under the proposed differentiable model, we can compute the texture and shape gradient with respect to a loss function defined on the measurement using differentiable rendering.
- *Validation on synthetic and real experiments.* We conduct multiple experiments on both synthetic measurements as well as real measurements from two hardware prototypes [Asif *et al.*, 2016, Hua *et al.*, 2020] and show results on texture-only as well as joint texture and shape recovery.

Limitations. The implementation that we provide has a few limitations. First, the proposed method is sensitive to initialization of depth. Triangles whose normals are nearly perpendicular to the optical axis can result in numerically unstable depth gradients. This is a consequence of the Monge model that we adopt and can be circumvented by using freeform meshes. Since we use Monte Carlo sampling to render lensless measurements, the proposed need to render a large number of rays to reduce variation in rendered measurements. For high resolution reconstruction, this results in high GPU memory requirements and long compute time.

5.1 Prior Work

5.1.1 Forward Models for Lensless Imaging

It is essential for all reconstruction methods to accurately model the forward process of the lensless imager, i.e. how scenes map to lensless measurements. While learning-based methods improve the perceptual quality of reconstructions, they perform best when the knowledge of forward process is integrated [Khan *et al.*, 2019, Monakhova *et al.*, 2019b]. Monakhova *et al.* [2019b] found better reconstruction quality from combining a unrolled ADMM that uses the foward process and a denoising network than using only a U-Net, and Khan *et al.* [2019] uses the separable property of the forward operator in their reconstruction network. Thus, let's take a close look at popular models in prior research.

Convolution Model. A convolutional model is a popular chouce for the forward process for both amplitude mask [Asif, 2018, Hua *et al.*, 2020] and phase mask [Antipa *et al.*, 2018, Boominathan *et al.*, 2020] based imagers. When the point spread function (PSF) of the lensless imager is shift-invariant for points at the same depth, and scales with depth change, we can parameterize the scene as the brightness of point sources at a few depth $\mathbf{i}_{z_1}, \ldots, \mathbf{i}_{z_N}$, so that the measurement $\mathbf{b}$ on a large sensor is a sum of convolutions,

$$\mathbf{b} = \sum_{n=1}^{N} \mathbf{m} * \mathbf{i}_{z_n} \tag{5.1}$$

where $\mathbf{m}$ is the PSF, and in the case of amplitude-mask imagers, the mask pattern.

The benefits of modeling the forward process as well-understood convolution are many: we can apply Fourier analysis on the PSF to understand the resolution bound of the imager, which helps the design of of well-conditioned lensless imagers; the forward process can be computed very fast in the Fourier domain, if we approximate linear convolution with circular convolution; single-depth scene reconstructions can be very quickly computed by deconvolution; calibration of this model need only one observation of a well-placed point light source.

The drawback of the convolution model is that it ignores many effects and they show up as artifacts degrading the reconstruction quality. On the scene side, specular reflections and opaque surfaces often found in common environments are poorly modeled by point light sources. On the sensor side, the finite size of the sensor means we only observe a cropped version of the measurement, and the convolution is not circular. Furthermore, the sensor pixels are less efficient when light reaches them at a large angle; this results in heavy vignetting in almost all lensless reconstructions.

Separable Matrices Model. Another line of work produces very thin lensless imagers with printed amplitude masks [Adams *et al.*, 2017, Asif *et al.*, 2016]. They show that separable 2D masks can be constructed from 1D MURA and M-sequence codes, and the number of parameters in the result imagers forward process is significantly reduced compared to a imager built with non-separable 2D mask. For example, for a plane consisting of $n \times n$ point light sources and a sensor of $p \times p$ pixels, instead of $\mathbf{b} = \mathbf{\Phi i}$, where $\mathbf{\Phi}$ has size $p^2 \times n^2$, the imager with separable amplitude mask can be modeled as $\mathbf{B} = \mathbf{\Phi}_l \mathbf{I} \mathbf{\Phi}_r^T$, where $\mathbf{\Phi}_l, \mathbf{\Phi}_r$ are of size $p \times n$.

This model captures most of the sensor-side effects, including linear convolution borders, finite sensor area, and separable angular response functions. However, it is tedious to calibrate this model, since we need to record the image of each row and column of point sources in the imaged volume, requiring $2nD$ captures for D depths.

5.1.2 Differentiable Rendering

Differential rendering has been proven useful for reconstructing the 3D geometry from 2D observations in the past few years [Kato *et al.*, 2020]. Tsai *et al.* [2019] used differential rendering to solve for surface geometry in non-line-of-sight (NLOS) imaging setting. They obtain higher level of details in the reconstructed NLOS scene, by representing the scene as a surface instead of previous volumetric approaches. We adopt the same approach, modeling a scene by surfaces and computes derivatives of the lensless measurements with respect to the surface geometry and reflectance. However, unlike the NLOS measurements which captures time (and therefore length of each light path), the lensless imager measurements have weaker constraint on the surface geometry. So we adopt the more restrictive Monge surface representation, instead of full mesh representation as in Tsai *et al.* [2019].

A number of work has explored the proper shape gradients under realistic settings, including complex geometric discontinuities and light transport phenomena [Delaunoy and Prados, 2011, Zhang *et al.*, 2020]. Delaunoy and Prados [2011] derive vertex gradient for two generic energy functionals with rigorous account for visibility. There are some differences to this work that stem from features specific to lensless imagers. First, the lensless imager pixels receive light rays from more than one point in the scene, making the problem more challenging. Second, this work also adds a "sensor angular response" term, as lensless imagers, especially thin ones, receive rays at a large angle to the optical axis. Thus the angular efficiency of sensor pixels become visible in the measurements, and result in heavy vignetting.

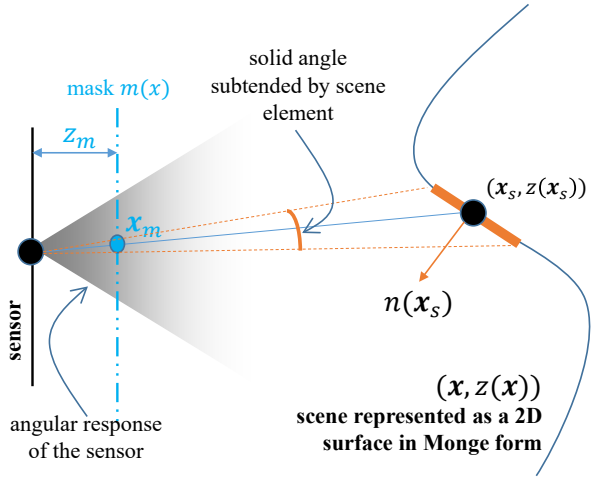


Figure 5.2: **Forward model from modeling the scene as a surface.**

5.2 Method

We now describe a Monte Carlo-based inverse rendering model for lensless cameras.

5.2.1 Basics of Image Formation

Suppose that we have a lensless camera comprising of an image sensor aligned with the xy-plane at $z = 0$, with its optical axis oriented along the z-axis. An amplitude mask with attenuation pattern $m(\mathbf{x})$, where $\mathbf{x} = (x, y)$, is placed at $z = z_m$ and parallel to the sensor plane. The scene is described in the Monge form as a height map $z(\mathbf{x})$ which describes a 2D surface $(\mathbf{x}, z(\mathbf{x}))$ embedded in 3D space. Figure 5.2 provides a schematic of this setup. Our implementation uses a slightly different model that we refer to as scaled Monge form, where the surface is represented as $(\mathbf{x}_s, 1)z(\mathbf{x}_s)$.

Now consider a location $\mathbf{x}_0$ on the sensor and a point $(\mathbf{x}_s, z(\mathbf{x}_s))$ on the scene surface indexed by xy location $\mathbf{x}_s$. The flux received at an infinitesimal area at the sensor location $\mathbf{x}_0$ due to an infinitesimal area on the scene at $\mathbf{x}_s$ is given as

$$\mathcal{L}(\mathbf{x}_s \rightarrow \mathbf{x}_0) m(\mathbf{x}_m) a\left(\frac{\mathbf{x} - \mathbf{x}_0}{z(\mathbf{x})}\right) d\omega(\mathbf{x}, \mathbf{x}_0) d\mathbf{x}_0,$$

where $\mathcal{L}(\mathbf{x}_s \rightarrow \mathbf{x}_0)$ is the radiance of the light ray at $\mathbf{x}_s$ towards the sensor location $\mathbf{x}_0$, and $a(\cdot)$ is the sensor angular response as a function of the tangent of the angle made with the optical axis. The term

$$\mathbf{x}_m = \mathbf{x}_0 + \frac{z_m}{z(\mathbf{x}_s)}(\mathbf{x}_s - \mathbf{x}_0)$$

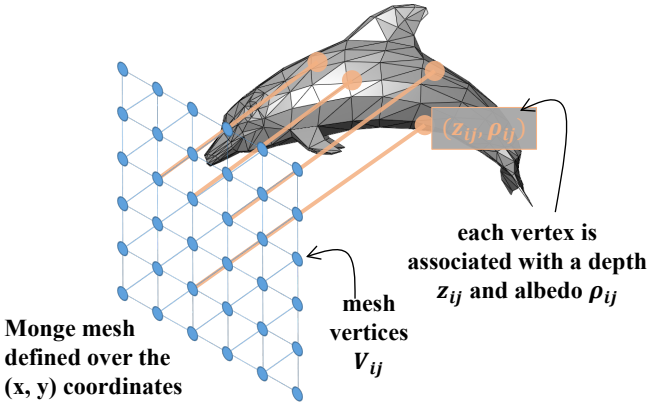


Figure 5.3: **A Monge mesh model.** We start with a mesh with uniformly placed vertices over the xy plane. Each vertex is endowed with a depth and albedo value that describes the scene scene and its texture.

is the location where the ray $\mathbf{x}_s \rightarrow \mathbf{x}_0$ intersects the the mask plane. Finally, the solid angle subtended by the scene surface element is given as

$$d\omega(\mathbf{x}_s, \mathbf{x}_0) = \frac{dA(\mathbf{x}_s)}{\|\mathbf{x}_s - \mathbf{x}_0\|_2^2 + z(\mathbf{x}_s)^2}.$$

Hence, the flux measured at a pixel of area Δ_0 centered at location $\mathbf{x}_0$ is approximately

$$\Delta_0 \int_{\mathbf{x}_s} \mathcal{L}(\mathbf{x}_s \rightarrow \mathbf{x}_0) m(\mathbf{x}_m) a\left(\frac{\mathbf{x}_s - \mathbf{x}_0}{z(\mathbf{x}_s)}\right) d\omega(\mathbf{x}, \mathbf{x}_0), \tag{5.2}$$

where the approximation arises from the assumption that the irradiance of incident light is constant over the sensor pixel. We are also ignoring the effect of self-occlusion by the surface, which can be easily incorporated by adding in a binary term indicating the visibility of the scene point to the sensor pixel.

5.2.2 Monge Mesh Parametrization of the Scene

We represent the scene using a **fixed** triangulated mesh over the xy plane. Specifically, given the surface described as $(\mathbf{x}, z(\mathbf{x}))$, we uniformly sampling $\mathbf{x} = (x, y)$ over the over the expected footprint of the scene to create a 2D mesh with $N \times N$ vertices denoted as $\{V_{ij}, 0 \leq i, j < N\}$. Given a sampling distance δ along each direction, the vertex V_{ij} corresponds to a 2D location

$$\mathbf{x}_{ij} = (i\delta - N\delta/2, j\delta - N\delta/2).$$

We can now represent the surface by endowing each vertex with a depth $z_{ij} = z(\mathbf{x}_{ij})$, and triangulating it in a fixed regular fashion as shown in Figure 5.3. Triangulation enables us to propagate the depth

values from the vertices to the faces by linear interpolating the vertex depth using local barycentric coorindates defined the vertices defining the triangle. Each face of the triangle also as a well-defined normal that is computed by fitting a plane to the three vertices defining it and computing its normal.

To compute the image intensities using equation (5.2), we also need to model the radiance field emitted by the surface. As with depth, we can model the radiance field at the vertices and propagate it to the faces. As a starting point, we consider Lambertian surfaces and hence, model the radiance field to be constant at each vextex location, i.e., $\forall \mathbf{x}_0,\ \mathcal{L}(\mathbf{x}_{ij} \rightarrow \mathbf{x}_0) = \mathcal{L}(\mathbf{x}_{ij}) = \rho_{ij}$.

With this definition of the Monge mesh, we can now describe the forward rendering approach.

5.2.3 Forward Model via Monte Carlo Rendering

To compute the flux observed at a pixel at $\mathbf{x}_0$, we need to calculate the integral in equation (5.2), which can be done efficiently using Monte Carlo simulations. We first generate Q samples $\{\mathbf{x}^{(k)}\}$ on the mesh by uniformly sampling over its 2D extent. For each sample, we identical the face of the mesh it belongs to. This allows us to calculate the depth $z^{(k)}$, albedo $\rho^{(k)}$, and surface normal $\mathbf{n}^{(k)}$ which we calculate from the depth and albedo of the vertices defining the face. Specifically, if the vertices defining the face are V_{k1}, V_{k2}, V_{k3}, then the depth and albedo associated with the sample are given as

$$\begin{aligned} z^{(k)} &= \alpha_1 z_{k1} + \alpha_2 z_{k2} + (1 - \alpha_1 - \alpha_2) z_{k3} \\ \rho^{(k)} &= \alpha_1 \rho_{k1} + \alpha_2 \rho_{k2} + (1 - \alpha_1 - \alpha_2) \rho_{k3}, \end{aligned} \tag{5.3}$$

where α_1 and α_2 are the barycentric coordinates of the point $\mathbf{x}^{(k)}$ in terms of vertex locations $\mathbf{x}_{k1}, \mathbf{x}_{k2},$ and$\mathbf{x}_{k3}$. The surface normal $\mathbf{n}^{(k)}$ is assigned to the face normal which is calculated from the 3D location corresponding to the vertices.

Now the flux at the pixel at $\mathbf{x}_0$ can be calculated as

$$\frac{A}{Q} \sum_{k=1}^{Q} \rho^{(k)} m\left(\mathbf{x}_m^{(k)}\right) a\left(\frac{\mathbf{x}^{(k)} - \mathbf{x}_0}{z^{(k)}}\right) d\omega(\mathbf{x}^{(k)}, \mathbf{x}_0) \tag{5.4}$$

Note that this calculation can be done extremely fast since the contribution of each Monte Carlo sample can be calculated in parallel. The expression in equation (5.4) gives us our forward model.

5.2.4 Rendering with Scaled Monge Mesh

The Monge form allows us to parameterize surfaces in a hyperrectangle. However, since the sensor pixels have reduced efficiency for light rays with large incident angles, the lensless camera sees a cone-

shaped volume instead of a hyperrectangle. Thus, we adopt a scaled Monge form that parameterizes surfaces in the pyramid volume.

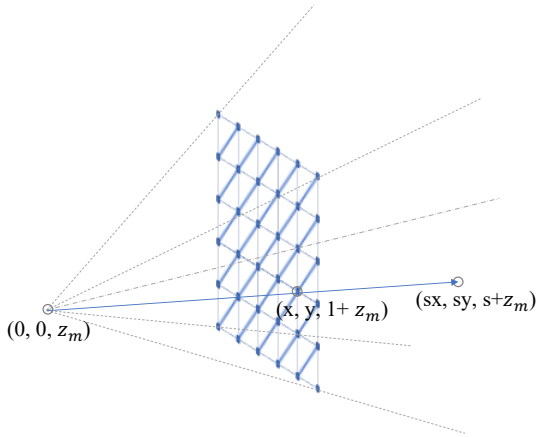


Figure 5.4: **Scaled Monge Parameterization of scene surface.**

As shown in Figure 5.4, we choose fixed points (x, y) on the $x - y$ plane 1 unit distance away from the mask center, and parameterize the surface by $s(x, y)$. The vertices on the surface have Cartesian coordinate $(sx, sy, s + z_m)$ w.r.t. center of the sensor.

In our implementation of Monte Carlo rendering, we choose points (x, y) with uniform probability on the $x - y$ plane 1 unit distance away from the mask center. This means we have to infer $A_{surface}$, the area of triangle around $\mathbf{x} = (sx, sy, s + z_m)$ on the scene surface from A_{grid}, area of triangles on the grid around $(x, y, 1 + z_m)$:

$$\frac{A_{surface}}{A_{grid}} = \frac{s^2}{\cos(\theta)\sqrt{x^2 + y^2 + 1}} \tag{5.5}$$

where θ is the angle between surface normal of triangle around $\mathbf{x}$ and ray $\mathbf{s} = (x, y, 1)$.

5.2.5 Inverse Rendering

Our goal is to recover the 2D surface defining the scene and its associated albedo map from the measurements made by the lensless imager. Since the vertices of the mesh are fixed, we need to optimize over $\Theta = (\mathbf{z}, \mathbf{r})$ where $\mathbf{z} = \{z_{ij}\}$ and $\mathbf{r} = \{\rho_{ij}\}$. To recover these we can setup an optimization problem

$$\min_{\Theta} \ell(\Theta) = \min_{\mathbf{z},\mathbf{r}} \frac{1}{2} \sum_{\mathbf{x}_0} \|y(\mathbf{x}_0) - b(\mathbf{x}_0; \Theta)\|^2,$$

where $y(\mathbf{x}_0)$ is the measured flux at the pixel $\mathbf{x}_0$ and $b(\mathbf{x}_0; \Theta)$ is the *rendered* flux using equation (5.4). We can solve for the mesh parameters by applying gradient descent or its stochastic version, both of

which require us to compute the derivative of the loss function ℓ with respect to the mesh variables $\mathbf{z}$ and $\mathbf{r}$. This derivative can be written as

$$\frac{\partial}{\partial \Theta}\ell(\Theta) = \sum_{\mathbf{x}_0}(b(\mathbf{x}_0;\Theta) - y(\mathbf{x}_0))\frac{\partial b}{\partial \Theta}$$

This derivative can also be computed via Monte Carlo rendering, following well established ideas in inverse rendering [Tsai *et al.*, 2019].

Texture gradient. Similar to equation (5.4), The texture gradient for vertex ij w.r.t. measurement at $\mathbf{x}_0$ can be calculated by

$$\frac{\partial b(\mathbf{x}_0)}{\partial \rho_{ij}} = \frac{A}{Q}\sum_{k=1}^{Q}\frac{\partial \rho^{(k)}}{\partial \rho_{ij}}\left(m\left(\mathbf{x}_m^{(k)}\right)a\left(\frac{\mathbf{x}^{(k)}-\mathbf{x}_0}{z^{(k)}}\right)d\omega(\mathbf{x}^{(k)},\mathbf{x}_0)\right),$$

where $\frac{\partial \rho^{(k)}}{\partial \rho_{ij}}$ is Bayercentric coordinate of point $\mathbf{x}^{(k)}$ in terms of vertex $\mathbf{x}_{ij}$ if $\mathbf{x}^{(k)}$ lies on a triangle which contains vertex $\mathbf{x}_{ij}$, and 0 otherwise.

Depth Gradient. The depth gradient for vertex ij w.r.t. measurement at $\mathbf{x}_0$ can be calculated by

$$\begin{aligned}\frac{\partial b(\mathbf{x}_0)}{\partial z_{ij}} = \frac{1}{Q}\sum_{k=1}^{Q}\Bigg(&\frac{\partial m\left(\mathbf{x}_m^{(k)}\right)}{\partial z_{ij}}\left(\rho^{(k)}a\left(\frac{\mathbf{x}^{(k)}-\mathbf{x}_0}{z^{(k)}}\right)Ad\omega(\mathbf{x}^{(k)},\mathbf{x}_0)\right)+\\ &\frac{\partial a\left(\frac{\mathbf{x}^{(k)}-\mathbf{x}_0}{z^{(k)}}\right)}{\partial z_{ij}}\left(\rho^{(k)}m\left(\mathbf{x}_m^{(k)}\right)Ad\omega(\mathbf{x}^{(k)},\mathbf{x}_0)\right)+\\ &\frac{\partial Ad\omega(\mathbf{x}^{(k)},\mathbf{x}_0)}{\partial z_{ij}}\left(\rho^{(k)}m\left(\mathbf{x}_m^{(k)}\right)a\left(\frac{\mathbf{x}^{(k)}-\mathbf{x}_0}{z^{(k)}}\right)\right)\Bigg)\end{aligned}$$

We use stochastic gradient descent to solve the optimization problem.

5.3 Simulations

5.3.1 Intensity-only Reconstruction

To illustrate how model mismatch degrades the imaging quality under simple forward models, we show the reconstruction results under convolution model ("conv"), separable model ("separable") [Asif *et al.*, 2016], and the proposed differentiable surface model assumption ("proposed") in Figure 5.5.

Sensor angular response function is a factor that degrades imaging quality when it is unmodeled in simple forward models. We simulate measurements for two types of sensor angular response functions:

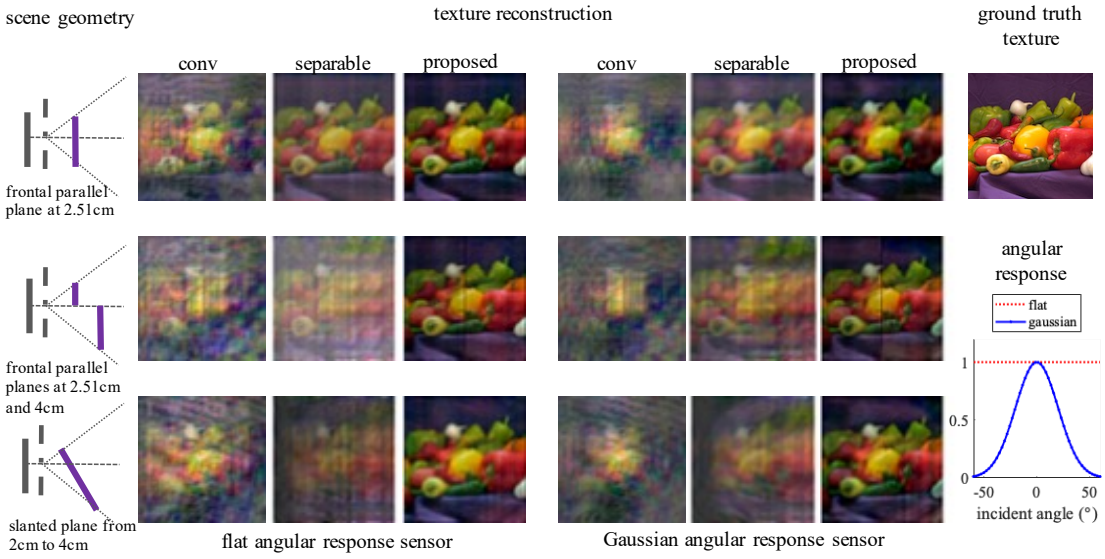


Figure 5.5: **Texture reconstruction results under various geometry and sensor angular response functions under different forward models.** The left column shows the geometry of the setup, with sensor (solid grey line), mask (dash grey line), and scene surface (purple lines) drawn roughly to scale. Each row of the table shows result of a different scene geometry: single frontal parallel plane, multiple frontal parallel planes, and a slanted plane. The convolution forward model is broken by cropping of the measurements as a result of finite sensor area, and only manages to recover the center portion of the image in frontal parallel plane and flat angular response setting. The measurements for three different scene geometry are simulated for a lensless imager similar to FlatCam prototype [Asif *et al.*, 2016], consisting of a 128×128 pixels on a 3.36mm by 3.36mm RGB sensor, and a binary amplitude mask with length 255 M-sequence outer product pattern printed at 20 µm feature size, and mask-to-sensor distance of 0.95mm. The measurements are obtained by ray tracing from a Lambertian surface with texture as shown on the top-left. The images are recovered at 128×128 resolution. Convolution model and separable model reconstruct the scene at depth 2.51cm. The proposed model recovers the scene with the knowledge of the scene geometry.

flat, and a more realistic Gaussian of standard deviation of 0.35 radians; the angular response functions are shown on the bottom right of Figure 5.5. In the first row, non-flat angular response function results in vignetting in both reconstruction from convolution model and separable models, but the proposed method can model it and avoid degraded image boarders.

Another factor that degrades imaging quality is scene geometry. Both convolution model and separable model assumes scenes are consisted of points emit light uniformly across the angle seen by the sensor. In realistic situations, scenes are often piecewise continuous. The second and third row shows that common scenarios such as multiple planes at different depth, and a slanted plane results in poor image quality from convolution and separable model, while the proposed method still recovers the scene quite well.

Loss functions. We find the scene by optimizing a loss function consisting of measurement term, texture prior term, and depth prior term

$$l(\Theta, b) = \frac{1}{N}\sum_{u}^{N} \|b_u - f(\Theta, u)\|_2^2 \tag{5.6}$$

$$+ \lambda_1 \text{cross-channel}(\mathbf{r}) + \lambda_2 \|\text{TV}(\mathbf{z})\|_1, \tag{5.7}$$

where $f(\Theta, u)$ defines the forward rendering, calculated via equation (5.4). We use basic cross-channel prior as proposed in Heide *et al.* [2013]:

$$\text{cross-channel}(\mathbf{r}) = \sum_{a=1}^{5} \|\mathbf{H}_a \mathbf{r}_c\|_1 \tag{5.8}$$

$$+ \lambda_3 \sum_{l \neq c} \sum_{a=1}^{2} \|\mathbf{H}_a \mathbf{r}_c \cdot \mathbf{r}_l - \mathbf{H}_a \mathbf{r}_l \cdot \mathbf{r}_c\|_1 \tag{5.9}$$

Implementation details. We solve the optimization problem with optimizers implemented in Pytorch, using Adam optimizer for texture $\mathbf{r}$ and SGD optimizer with momentum weight 0.1 for depth $\frac{z_m}{\mathbf{z}}$, where z_m is sensor-to-mask distance. For the real data results shown in Figure 5.9, we use $\lambda_1 = 5 * 10^{-3}, \lambda_2 = 10^{-5}, \lambda_3 = 0.3$, initial texture learning rate 10^{-3}, initial depth learning rate 10^{-1}, and decrease the learning rate of all variables by 0.985 every 250 steps. We render 65536 rays from the surface for each pixel in the optimization, and render 1048576 rays for each pixel for producing measurements in simulation experiment. We use the value from 128 pixels to estimate gradient on each step of gradient descent. The optimizer typically run for several hours to convergence.

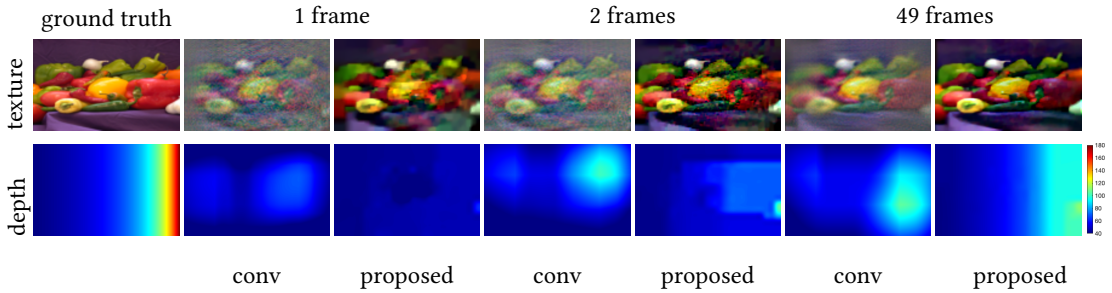


Figure 5.6: **Comparison of reconstructing both texture and depth under different models.** The measurements are simulated for a scene consisting of a slanted plane with depth ranging from 40mm to 180mm. The surface is Lambertian and has texture as shown on the left. The imager is chosen with the same parameters as SweepCam prototype [Hua *et al.*, 2020], with 300×480 pixels across 7.032mm by 11.25mm, mask-to-sensor distance of 13.1mm, and use same mask pattern as SweepCam prototype. We simulate multiple measurements captured with translated mask pattern: 2 frames capture horizontal translation of length 2.54mm, 49 frames capture both horizontal and vertical translation in 7×7 grid in a window of size 7.78mm×7.78mm.

5.3.2 Joint intensity and shape reconstruction

Often, we do not know the shape of the scene we want to recover. We show the results of joint reconstruction of intensity and depth from lensless measurements in Figure 5.6.

The convolution model and separable model can recover the depth of scene by reconstructing scene texture at many depth planes, a focus stack, and use a local contrast measure to determine which planes are occupied by a texture surface patch with the highest local contrast. We use the focusing operation introduced in SweepCam [Hua *et al.*, 2020] to make use of additional frames captured with translated mask pattern. The depth obtained from the textured regions are propagated to the textureless regions to obtain depth map. The depth map can help us blend frames from the stack to form a all-in-focus texture image. We use the proposed method to refine the depth map and texture image obtained from the focus stack. We observe that the proposed method enhanced the contrast of texture image, and improved the depth estimation. Additionally, increasing the number of measurement captured with translated mask pattern resulted in better depth reconstruction.

5.4 Hardware Experiments

Next, we validate the proposed method on real data from two different lensless prototypes.

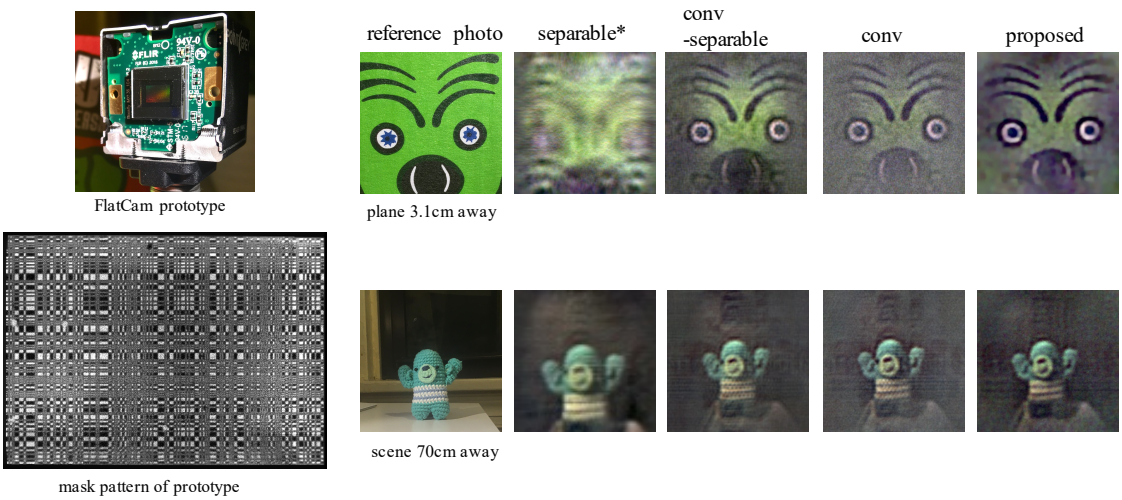


Figure 5.7: **Flatcam prototype results.** The prototype shown on top-left of the figure is a thin lensless imager consisting of a printed binary mask pattern affixed on top of a Sony IMX 136 RGB sensor. The mask pattern obtained from calibration is shown in bottom left. Top right shows texture reconstruction under various forward models of a near scene, a plane 3.1cm away. Bottom right shows texture reconstruction of a far scene, 70cm away. Separable model requires a laborous calibration procedure repeated for each depth, and it was only calibrated for far scenes; it fails on the near scene. "conv-separable" assumes convolution with a separable mask. "conv" is convolution with calibrated mask pattern as shown.

5.4.1 Calibration

The goal of calibration is to obtain a precise forward model of the lensless imager. Specifically, calibration estimates mask pattern $m(\mathbf{x})$, mask to sensor distance z_m, and angular response function $a(\theta)$. The calibration problem can be posed as

$$\arg\min_{m,z_m,a} \|f_{m,z_m,a}(i,z) - b\|_2^2 \tag{5.10}$$

Since f is differentiable with respect to the calibration parameters, the same stochastic gradient descent method used for reconstructing the scene can be similarly applied to solve equation (5.10) if we know the scene (i,z) and have a rough estimation for initialization.

We propose a calibration procedure similar to standard camera calibration [Zhang, 2000]: observe a planar target of known dimension at many poses. Since we can get an image of the mask from imaging point light sources, we use a target composed of them: an LED array. We move the LED array to multiple different poses in front of the lensless imager, separately turn on the LEDs one by one and capture a measurement for each. Registering the projected shadow of each LED provides us with an

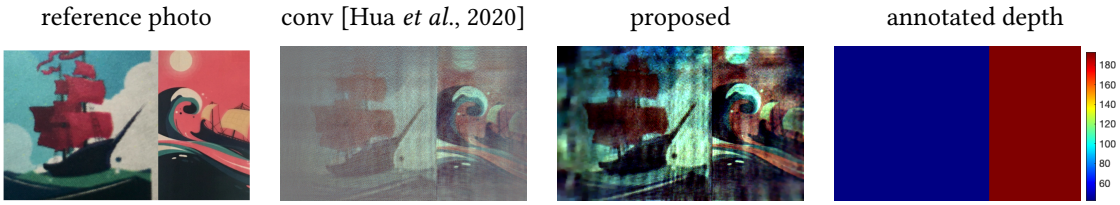


Figure 5.8: **Texture reconstruction from measurements captured with single mask pattern and depth map from annotation .**

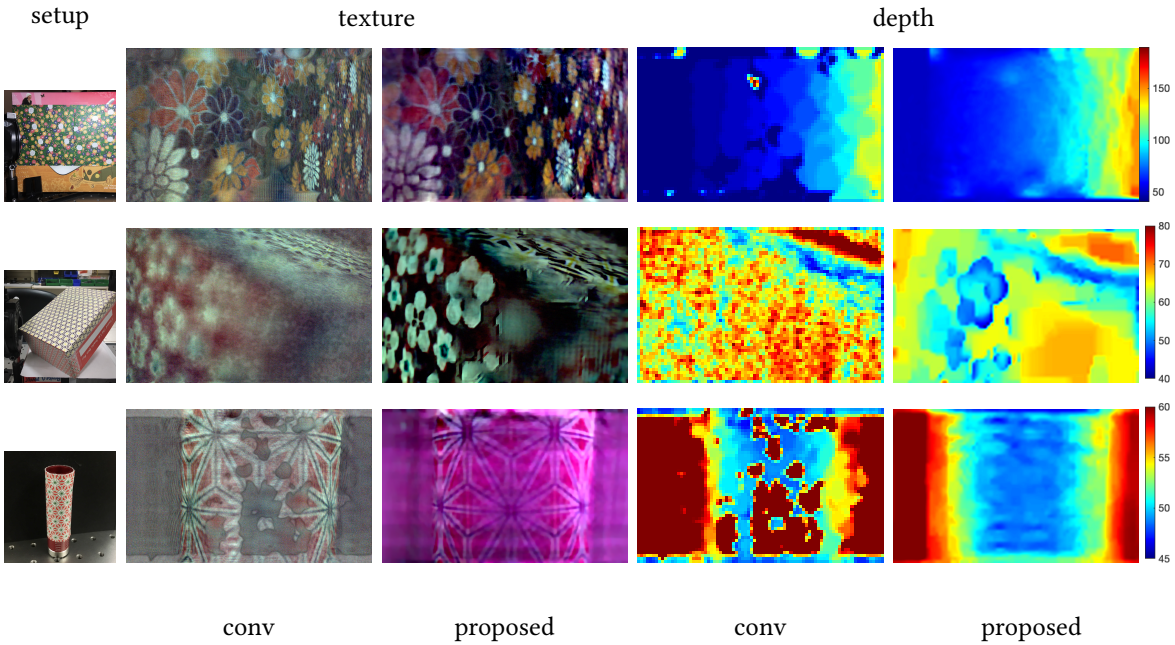


Figure 5.9: **Texture and depth reconstruction from measurements captured with 49 translated mask pattern.** "conv stack" are obtained by selecting depth with maximum local contrast from focus stack [Hua *et al.*, 2020]. Proposed method initializes from "conv stack" texture and box-filtered "conv stack" depth map. Texture reconstructions are brightness adjusted for easier comparison.

initial estimate, which we refine using stochastic gradient descent on the camera parameters, namely the mask.

5.4.2 Static seperable mask lensless imager

Flatcam prototype, shown in Figure 5.7, is a thin imager with a large separable binary amplitude mask. It consists of a Sony IMX136 RGB sensor (1920×1200 pixels with pitch of 2.8μm) and a mask with $20u$m features covering the whole sensor. The mask-to-sensor distance is 0.95mm from calibration.

Figure 5.7 shows reconstruction under different models on measurements captured on the FlatCam prototype. Calibrating for the mask pattern as described here is advantageous over the separable model as it allows us to reconstruct for scene of any depth, unlike the separable model which can only reconstruct on depth it is has been calibrated. We observe that while the imager is designed to be separable, the mask pattern is no longer full separable, and the non-separable pattern ("conv") produced sharper results with less vignetting compared to a separable mask pattern ("conv-separable"). The proposed method improves the contrast and reduces artifacts seen in the convolution models.

5.4.3 Programmable mask lensless imager

Recovering texture and depth information from a single measurement yields a underdetermined problem. Therefore, we explore programmable mask lensless imagers, which allow us to capture multiple multiplexed measurements of the scene. SweepCam [Hua *et al.*, 2020] implements programmable mask lensless imager with a programmable amplitude mask, displaying translated versions of the same mask. We reconstruct data from SweepCam to compare the proposed method against the convolution model.

Single frame. We show that the proposed surface model with angular response is a better model than convolution in Figure 5.8. The "conv" forward model treats the measurement as a sum of two convolutions from two depths, and solves the problem via conjugate gradient descent. The proposed model performs stochastic gradient descent while rendering each pixel by ray tracing. The result of proposed model has higher contrast, and results in better details around the image border.

Translated masks. Finally we use multiple measurements captured with translated mask patterns to reconstruct both texture and depth in Figure 5.9. The proposed method refines the texture and depth from estimates from the focus stack. The proposed method recovers correct depth at textured regions, and correctly fills texture near those patches.

5.5 Discussion

This chapter proposed an inverse rendering technique for surface estimation for lensless cameras. We show successful recovery of shape and texture that are significantly better, if not comparable, to state-of-the-art. Many differentiable functions can be incorporated into the proposed frame work to more closely model the forward process of lensless imager. This method can be expanded to model the forward process of other high-dimensional data being recorded by the lensless imager, such as video, light field, and hyperspectral image.

6 Conclusion

6.1 Thesis Contributions

The main barrier to practical adaptation of lensless cameras is that the images reconstructed from lensless cameras currently have lower quality when compared to those of lens-based cameras. This thesis tackles the most common challenge in the reconstruction of scenes from lensless measurements: the PSF's depth dependency makes the inverse problem difficult to solve and analyze.

This thesis contributes to improving the imaging quality of 3D scenes with lensless cameras in the following ways:

- We provided the first theoretical framework for studying the achievable spatial-axial resolution of amplitude mask-based lensless cameras; this allows us to derive the upper bound of spatial and axial resolution as a function of the mask pattern.
- We introduced a hardware modification, *i.e.* a programmable mask, and show that imaging with many translated versions of the same mask pattern allows fast reconstruction of 3D scenes with few artifacts.
- We explored reconstructing the scene from lensless measurements under a physically-realistic forward model by utilizing techniques in inverse rendering, and show reconstructions with better contrast and details.

6.2 Future Work

The work from this thesis brings to light some future directions for improving lensless imaging.

6.2.1 Extension to Phase Mask-based Lensless Cameras

This thesis discusses amplitude mask-based lensless cameras. Phase masks-based lensless cameras are more light efficient because the phase masks do not block any light and often produce better conditioned

image systems.

The PSF from phase mask-based lensless cameras behave similarly to amplitude mask-based cameras when light enter the camera at small angles with respect to the optical axis due to the memory effect; the PSF translates when the point source translates parallel to the sensor plane and scales when the point source translates orthogonal to the sensor plane. Current phase mask-based lensless cameras [Antipa *et al.*, 2018, Boominathan *et al.*, 2020] operate within this range. In this case, the analysis and discussions in this thesis can be applied to the phase mask-based lensless cameras, by replacing the mask pattern with an appropriately-scaled version of their PSF.

The behavior of PSF of phase-mask based lensless camera changes when light enters the camera at large angles with respect to the optical axis. The modeling and analysis of this scenario remains a future work.

6.2.2 Effect of Diffraction

This thesis models light in ray optics and for the most part, ignore the effects of diffraction. Adapting the methods and analysis presented in this thesis to handle diffraction is similar to handling phase mask-based cameras. The method in Chapter 4 and 5 can both be extended to replace the mask by looking up a pre-calibrated PSF accounting for diffraction effects from a specific depth. The presence of diffraction causes model misfit for our analysis for scenes of extended depth range (~40 cm), as the PSF are no longer exactly scaled copies of each other; some examples of such PSFs are shown in Figure 3.2.

6.2.3 Designing Mask Patterns

We derive the 3D MTF, which represents the resolutions of the lensless camera, as a function of the mask pattern in Chapter 3. This allows us to design the mask pattern from the desired spatial and axial resolution performances of the camera. Specifically, we could formulate mask design as an optimization problem on r_l, the Radon Transform of the Laplacian of the mask pattern. For example, we could find an amplitude mask $m(\cdot)$ that maximizes the smallest 3D MTF value, $K^P(\rho, \psi, f_z)$ in target resolution range using Eq. 3.18,

$$\arg\max_{m(\cdot)} \min_{\substack{\rho,\psi,f_z \text{ in} \\ \text{target resolution}}} r_l\left(\frac{f_z}{\rho}, \psi\right) \text{ s.t. } m(\mathbf{x}) \in [0, 1].$$

6.2.4 Neural Networks for 3D Reconstruction from Lensless Measurements

This thesis explored mesh-based representation for modeling physically-realistic effects such as occlusion and angular effects in Chapter 5. This work can be extended to incorporate neural networks to better handle shape optimization, which has many local optima. Some concurrent works use neural networks for reconstructing intensity and depth maps from lensless measurements [Bagadthey *et al.*, 2022, Zheng *et al.*, 2021], but it is challenging to extend them to account for those physically-realistic effects. Neural implicit surfaces [Wang *et al.*, 2021] offers a promising method for reconstructing 3D scenes from lensless measurements.

One important barrier for using neural networks for 3D reconstruction from lensless measurements is the lack of training data – it is difficult to obtain ground truth geometry for 3D imaging scenarios calling for lensless cameras at large scale. A possible solution for this is to setup such configurations in simulation and render those lensless measurements. The work in Chapter 5 builds a renderer that can be used for this purpose.

6.2.5 Other Limiting Factors on Image Quality

In working with lensless measurements, we identified other limiting factors on the imaging quality of lensless cameras. One problem is typical lensless measurements encode scene information in small variations across pixels, and the small variations are hard to measure due to quantization and noise. Perhaps novel sensors and light-modulating technologies will alleviate this problem and further improve the imaging quality of lensless cameras.

6.3 Conclusion

We imagine the work presented in this thesis, along with other concurrent research in lensless cameras, will improve the imaging quality of lensless cameras so that they enable practical applications, such as *in vivo* imaging [Adams *et al.*, 2022], which may lead to discoveries in biology research and less invasive medical procedures. Looking further into the future, we envision this line of research will produce cameras that are light-weight, flexible, have real-time previews, and allow 3D or light field reconstruction.

Bibliography

Jens Ackermann and Michael Goesele. 2015. A survey of photometric stereo techniques. *Foundations and Trends® in Computer Graphics and Vision* 9, 3-4 (2015), 149–254.

Jesse K Adams, Vivek Boominathan, Benjamin W Avants, Daniel G Vercosa, Fan Ye, Richard G Baraniuk, Jacob T Robinson, and Ashok Veeraraghavan. 2017. Single-frame 3D fluorescence microscopy with ultraminiature lensless FlatScope. *Science Advances* 3, 12 (2017).

Jesse K Adams, Vivek Boominathan, Sibo Gao, Alex V Rodriguez, Dong Yan, Caleb Kemere, Ashok Veeraraghavan, and Jacob T Robinson. 2022. In vivo fluorescence imaging with a flat, lensless microscope. *Nature Biomedical Engineering* (2022).

Nick Antipa, Grace Kuo, Reinhard Heckel, Ben Mildenhall, Emrah Bostan, Ren Ng, and Laura Waller. 2018. DiffuserCam: Lensless single-exposure 3D imaging. *Optica* 5, 1 (Jan 2018), 1–9. `https://doi.org/10.1364/OPTICA.5.000001`

Nick Antipa, Patrick Oare, Emrah Bostan, Ren Ng, and Laura Waller. 2019. Video from stills: Lensless imaging with rolling shutter. In *2019 IEEE International Conference on Computational Photography (ICCP)*.

M Salman Asif. 2018. Lensless 3D Imaging Using Mask-Based Cameras. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*.

M Salman Asif, Ali Ayremlou, Aswin Sankaranarayanan, Ashok Veeraraghavan, and Richard G Baraniuk. 2016. Flatcam: Thin, lensless cameras using coded aperture and computation. *IEEE Transactions on Computational Imaging* 3, 3 (2016), 384–397.

Dhruvjyoti Bagadthey, Sanjana Prabhu, Salman S. Khan, D Tony Fredrick, Vivek Boominathan, Ashok Veeraraghavan, and Kaushik Mitra. 2022. FlatNet3D: intensity and absolute depth from single-shot

lensless capture. *J. Opt. Soc. Am. A* 39, 10 (Oct 2022), 1903–1912. https://doi.org/10.1364/JOSAA.466286

Amir Beck and Marc Teboulle. 2009. Fast gradient-based algorithms for constrained total variation image denoising and deblurring problems. *IEEE Transactions on Image Processing* 18, 11 (2009), 2419–2434.

Vivek Boominathan, Jesse Adams, Jacob Robinson, and Ashok Veeraraghavan. 2020. PhlatCam: Designed phase-mask based thin lensless camera. *IEEE Transactions on Pattern Analysis and Machine Intelligence* (2020).

Vivek Boominathan, Jesse K Adams, M. Salman Asif, Benjamin W. Avants, Jacob T. Robinson, Richard Baraniuk, Aswin C. Sankaranarayanan, and Ashok Veeraraghavan. 2016. Lensless Imaging: A computational renaissance. *IEEE Signal Processing Magazine* 33, 5 (2016), 23–35.

Vivek Boominathan, Jacob T Robinson, Laura Waller, and Ashok Veeraraghavan. 2022. Recent advances in lensless imaging. *Optica* 9, 1 (2022), 1–16.

Axel Busboom, Harald Elders-Boll, and Hans Dieter Schotten. 1998. Uniformly redundant arrays. *Experimental Astronomy* 8, 2 (1998), 97–123.

Robert T Collins. 1996. A space-sweep approach to true multi-image matching. In *CVPR*.

Amaël Delaunoy and Emmanuel Prados. 2011. Gradient flows for optimizing triangular mesh-based surfaces: Applications to 3d reconstruction problems dealing with visibility. *International journal of computer vision* 95, 2 (2011), 100–123.

Robert H. Dicke. 1968. Scatter-hole cameras for x-rays and gamma rays. *The Astrophysical Journal* 153 (1968), L101.

Marco F Duarte, Mark A Davenport, Dharmpal Takhar, Jason N Laska, Ting Sun, Kevin F Kelly, and Richard G Baraniuk. 2008. Single-pixel imaging via compressive sampling. *IEEE Signal Processing Magazine* 25, 2 (2008), 83–91.

Paolo Favaro and Stefano Soatto. 2005. A geometric approach to shape from defocus. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 27, 3 (2005), 406–417.

Edward E Fenimore and Thomas M Cannon. 1978. Coded aperture imaging with uniformly redundant arrays. *Applied Optics* 17, 3 (1978), 337–347.

Patrick R. Gill and David G. Stork. 2013. Lensless Ultra-Miniature Imagers Using Odd-Symmetry Spiral Phase Gratings. In *Imaging and Applied Optics*.

Ioannis Gkioulekas, Anat Levin, and Todd Zickler. 2016. An evaluation of computational imaging techniques for heterogeneous inverse scattering. In *European Conference on Computer Vision*. 685–701.

Solomon W Golomb. 1967. *Shift register sequences*. Aegean Park Press.

Stephen R Gottesman and EE Fenimore. 1989. New family of binary arrays for coded aperture imaging. *Applied Optics* 28, 20 (1989), 4344–4352.

Eugene Hecht. 2017. *Optics* (fifth edition ed.). Pearson.

Felix Heide, Mushfiqur Rouf, Matthias B Hullin, Bjorn Labitzke, Wolfgang Heidrich, and Andreas Kolb. 2013. High-quality computational imaging through simple lenses. *ACM Transactions on Graphics (TOG)* 32, 5 (2013), 1–14.

Yi Hua, M Salman Asif, and Aswin C Sankaranarayanan. 2023. Spatial and axial resolution limits for mask-based lensless cameras. *Optics Express* 31, 2 (Jan 2023), 2538–2551. `https://doi.org/10.1364/OE.480025`

Yi Hua, Shigeki Nakamura, M Salman Asif, and Aswin C Sankaranarayanan. 2020. SweepCam – Depth-aware Lensless Imaging using Programmable Masks. *IEEE Transactions on Pattern Analysis and Machine Intelligence* (2020).

Avinash C Kak and Malcolm Slaney. 2001. *Principles of computerized tomographic imaging*. SIAM.

Hiroharu Kato, Deniz Beker, Mihai Morariu, Takahiro Ando, Toru Matsuoka, Wadim Kehl, and Adrien Gaidon. 2020. Differentiable rendering: A survey. *arXiv preprint arXiv:2006.12057* (2020).

Salman S. Khan, V. R. Adarsh, Vivek Boominathan, Jasper Tan, Ashok Veeraraghavan, and Kaushik Mitra. 2019. Towards Photorealistic Reconstruction of Highly Multiplexed Lensless Images. In *ICCV*.

Salman Siddique Khan, Varun Sundar, Vivek Boominathan, Ashok Veeraraghavan, and Kaushik Mitra. 2020. Flatnet: Towards photorealistic scene reconstruction from lensless measurements. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 44, 4 (2020), 1934–1948.

David Kittle, Kerkil Choi, Ashwin Wagadarikar, and David J Brady. 2010. Multiframe image estimation for coded aperture snapshot spectral imagers. *Applied Optics* 49, 36 (2010), 6824–6833.

Kyros Kutulakos and Samuel W Hasinoff. 2009. Focal Stack Photography: High-Performance Photography with a Conventional Camera. In *Proceedings of the IAPR Conference on Machine Vision Applications*. 332–337.

Xiaochun Liu, Sebastian Bauer, and Andreas Velten. 2020. Phasor field diffraction based reconstruction for fast non-line-of-sight imaging systems. *Nature communications* 11, 1 (2020), 1–13.

Patrick Llull, Xuejun Liao, Xin Yuan, Jianbo Yang, David Kittle, Lawrence Carin, Guillermo Sapiro, and David J Brady. 2013. Coded aperture compressive temporal imaging. *Optics Express* 21, 9 (2013), 10526–10545.

Kede Ma, Hui Li, Hongwei Yong, Zhou Wang, Deyu Meng, and Lei Zhang. 2017. Robust multi-exposure image fusion: a structural patch decomposition approach. *IEEE Transactions on Image Processing* 26, 5 (2017), 2519–2532.

Stephen Robert Marschner. 1998. *Inverse rendering for computer graphics*. Ph.D. Dissertation. Cornell University.

Daniel Miau, Oliver Cossairt, and Shree K Nayar. 2013. Focal sweep videography with deformable optics. In *IEEE International Conference on Computational Photography (ICCP)*. IEEE, 1–8.

Kristina Monakhova, Kyrollos Yanny, Neerja Aggarwal, and Laura Waller. 2020. Spectral DiffuserCam: Lensless snapshot hyperspectral imaging with a spectral filter array. *Optica* 7, 10 (2020), 1298–1307.

Kristina Monakhova, Joshua Yurtsever, Grace Kuo, Nick Antipa, Kyrollos Yanny, and Laura Waller. 2019a. Learned reconstructions for practical mask-based lensless imaging. *Optics express* 27, 20 (2019), 28075–28090.

Kristina Monakhova, Joshua Yurtsever, Grace Kuo, Nick Antipa, Kyrollos Yanny, and Laura Waller. 2019b. Learned reconstructions for practical mask-based lensless imaging. *Opt. Express* 27, 20 (Sep 2019), 28075–28090.

Shree K Nayar and Yasuo Nakagawa. 1990. Shape from focus: An effective approach for rough surfaces. In *IEEE Intl. Conf. Robotics and Automation (ICRA)*.

Shree K Nayar and Yasuo Nakagawa. 1994. Shape from focus. *IEEE Trans. Pattern analysis and machine intelligence* 16, 8 (1994), 824–831.

Thuong Nguyen Canh and Hajime Nagahara. 2019. Deep Compressive Sensing for Visual Privacy Protection in FlatCam Imaging. In *CVPR*.

Matthew O'Toole, David B Lindell, and Gordon Wetzstein. 2018. Confocal non-line-of-sight imaging based on the light-cone transform. *Nature* 555, 7696 (2018), 338–341.

Jae Young Park and Michael B Wakin. 2013. Multiscale algorithm for reconstructing videos from streaming compressive measurements. *Journal of Electronic Imaging* 22, 2 (2013), 021001.

Gustavo Patow and Xavier Pueyo. 2003. A survey of inverse rendering problems. In *Computer graphics forum*, Vol. 22. 663–687.

Ron J. Pieper and Adrianus Korpel. 1983. Image processing for extended depth of field. *Applied Optics* 22, 10 (1983), 1449–1453.

Joshua D Rego, Karthik Kulkarni, and Suren Jayasuriya. 2021. Robust Lensless Image Reconstruction via PSF Estimation. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*. 403–412.

Daniel Scharstein and Richard Szeliski. 2002. A Taxonomy and Evaluation of Dense Two-Frame Stereo Correspondence Algorithms. *International Journal of Computer Vision* 47, 1 (01 Apr 2002), 7–42.

Claude E Shannon. 1949. Communication in the Presence of Noise. *Proceedings of the IRE* 37, 1 (jan 1949), 10–21. https://doi.org/10.1109/jrproc.1949.232969

Kfir Shem-Tov, Sai Praveen Bangaru, Anat Levin, and Ioannis Gkioulekas. 2020. Towards Reflectometry from Interreflections. In *IEEE International Conference on Computational Photography (ICCP)*. 1–12.

Takeshi Shimano, Yusuke Nakamura, Kazuyuki Tajima, Mayu Sao, and Taku Hoshizawa. 2018. Lensless light-field imaging with Fresnel zone aperture: quasi-coherent coding. *Applied optics* 57, 11 (2018), 2841–2850.

Norbert Streibl. 1985. Three-dimensional imaging by a microscope. *JOSA A* 2, 2 (1985), 121–127.

Hari Sundaram and Shree Nayar. 1997. Are textureless scenes recoverable? In *IEEE Conference on Computer Vision and Pattern Recognition*. 814–820.

Jasper Tan, Li Niu, Jesse K Adams, Vivek Boominathan, Jacob T Robinson, Richard G Baraniuk, and Ashok Veeraraghavan. 2018. Face Detection and Verification Using Lensless Cameras. *IEEE Transactions on Computational Imaging* 5, 2 (2018), 180–194.

Jun Tanida, Tomoya Kumagai, Kenji Yamada, Shigehiro Miyatake, Kouichi Ishida, Takashi Morimoto, Noriyuki Kondou, Daisuke Miyazaki, and Yoshiki Ichioka. 2001. Thin observation module by bound optics (TOMBO): concept and experimental verification. *Applied Optics* 40, 11 (2001), 1806–1813.

Chia-Yin Tsai, Aswin C Sankaranarayanan, and Ioannis Gkioulekas. 2019. Beyond Volumetric Albedo–A Surface Optimization Framework for Non-Line-Of-Sight Imaging. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 1545–1555.

Andreas Velten, Thomas Willwacher, Otkrist Gupta, Ashok Veeraraghavan, Moungi G Bawendi, and Ramesh Raskar. 2012. Recovering three-dimensional shape around a corner using ultrafast time-of-flight imaging. *Nature communications* 3, 1 (2012), 1–8.

Ashwin Wagadarikar, Renu John, Rebecca Willett, and David Brady. 2008. Single disperser design for coded aperture snapshot spectral imaging. *Applied Optics* 47, 10 (2008), B44–B51.

Peng Wang, Lingjie Liu, Yuan Liu, Christian Theobalt, Taku Komura, and Wenping Wang. 2021. NeuS: Learning Neural Implicit Surfaces by Volume Rendering for Multi-view Reconstruction. *Advances in Neural Information Processing Systems* 34 (2021), 27171–27183.

Robert J Woodham. 1980. Photometric method for determining surface orientation from multiple images. *Optical engineering* 19, 1 (1980), 191139.

Keita Yamaguchi, Yusuke Nakamura, Kazuyuki Tajima, Toshiki Ishii, Koji Yamasaki, and Takeshi Shimano. 2019. Lensless 3D sensing technology with Fresnel zone aperture based light-field imaging. In *ODS 2019: Industrial Optical Devices and Systems*, Vol. 11125. International Society for Optics and Photonics, 111250F.

Cheng Zhang, Bailey Miller, Kai Yan, Ioannis Gkioulekas, and Shuang Zhao. 2020. Path-space differentiable rendering. *ACM Trans. Graph.(Proc. SIGGRAPH)* 39, 6 (2020), 143.

Zhengyou Zhang. 2000. A flexible new technique for camera calibration. *IEEE Transactions on pattern analysis and machine intelligence* 22, 11 (2000), 1330–1334.

Yucheng Zheng and M Salman Asif. 2020. Joint image and depth estimation with mask-based lensless cameras. *IEEE Transactions on Computational Imaging* 6 (2020), 1167–1178.

Yucheng Zheng, Yi Hua, Aswin C Sankaranarayanan, and M Salman Asif. 2021. A Simple Framework for 3D Lensless Imaging with Programmable Masks. In *ICCV*.

Assaf Zomet and Shree K Nayar. 2006. Lensless Imaging with a Controllable Aperture. In *CVPR*, Vol. 1. 339–346.