Title

SweepCam — Depth-Aware Lensless Imaging Using Programmable Masks

Journal

IEEE Transactions on Pattern Analysis and Machine Intelligence, 42(7)

ISSN

0162-8828

Authors

Hua, Yi

Nakamura, Shigeki

Asif, M Salman

et al.

Publication Date

2020-07-01

DOI

10.1109/tpami.2020.2986784

Peer reviewed

SweepCam — Depth-aware Lensless Imaging using Programmable Masks

Yi Hua *Student Member, IEEE*, Shigeki Nakamura, M. Salman Asif,
and Aswin C. Sankaranarayanan, *Senior Member, IEEE*

Abstract—Lensless cameras, while extremely useful for imaging in constrained scenarios, struggle with resolving scenes with large depth variations. To resolve this, we propose imaging with a set of mask patterns displayed on a programmable mask, and introduce a computational focusing operator that helps to resolve the depth of scene points. As a result, the proposed imager can resolve dense scenes with large depth variations, allowing for more practical applications of lensless cameras. We also present a fast reconstruction algorithm for scene at multiple depths that reduces reconstruction time by two orders of magnitude. Finally, we build a prototype to show the proposed method improves both image quality and depth resolution of lensless cameras.

Index Terms—Lensless Imaging, Computational Photography

Fig. 1. *Lensless focal stack.* We show images reconstructed at three different depths using our proposed SweepCam technique, which is a lensless camera with a programmable mask.

1 Introduction

Recent advances in lensless cameras [1] have produced numerous designs for imaging in scenarios where there is a need to avoid the bulk and the standoff distances associated with a lens. In these designs, the lens is replaced with a modulation element placed in close proximity to the sensor; common choices for this element include amplitude masks [2], [3], [4], phase masks in the form of thin diffusers [5] and lenslet arrays [6], as well as diffraction gratings [7]. Under incoherent light, the sensor measurements can be modeled as being linear in the scene intensities and an image can be computationally reconstructed.

The operating principle of a lensless camera is not entirely different from that of a lens-based camera. In both cases, a scene point produces a point spread function (PSF) that is depth dependent. However, the size of the PSF is remarkably different in both cases. In a lens-based camera, points on the focal plane have a highly compact PSF that is restricted within a few pixels. When the scene is planar and in focus, the sensor measurements produce an image of the scene directly. In a lensless camera, the PSF is large for all depths, often covering a large portion of the sensor, and thus requires a deconvolution procedure to reconstruct the image from the sensor measurements. Further, given the depth dependence of the PSF, precise reconstruction requires that we estimate the depth as well which results in a highly non-linear inverse problem.

One approach to simplify the non-linear reconstruction problem is to represent the scene as an intensity function over a 3D volume, instead of texture and depth map; this "lifting" of the unknown variable results in a linear inverse problem. 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 underdetermined, 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 paper 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 Fig. 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 paper 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 [8], we propose to regularize the depth recovery using measurements made from a translating mask and processed by a computational focusing operator.
- *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 deconvolution. This provides significant computational benefits especially when the scene has content on a large number of depths.
- *Validation using an experimental prototype.* We demonstrate programmability provides improvement in image quality over state-of-the-art lensless imagers and their associated algorithms on a lab prototype.

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.

2 Prior Work

We provide a brief overview of lensless imaging techniques. For a detailed overview, please refer to [1].

2.1 Lensless Imaging with Static Masks

This work builds upon the core ideas from previous lensless imagers, especially FlatCam [2] and DiffuserCam [5]. 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 in terms of exploring applications in face-detection [9], privacy protection [10], and fluorescent microscopy [4]. More recent work has focused on mitigating inadequacies of the calibration and reconstruction procedure by including a deep neural network in the reconstruction pipeline [11].

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 underdetermined in both of those methods, 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 [3], [12] 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 sparsity 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 proposed greedy depth-pursuit estimation algorithm [3] and then refined by alternating descent algorithm [12]. They also observe 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 [3], 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.

2.2 Lensless Imaging with a Programmable Mask

Zomet and Nayar [13] use multiple liquid crystal displays (LCDs) as a programmable aperture whose field-of-view can be changed without mechanical movements. However, while Zomet and Nayar use a small aperture, the proposed design uses a large coded aperture followed by depth-aware deconvolution, so that the signal to noise ratio is significantly increased in the captured measurements.

2.3 Multiple Capture Imagers

The ideas in this paper are closely related to prior work on multiple-capture imagers proposed in the context of compressive sensing; example include the single pixel camera [14], the CASSI system [15], [16] for hyperspectral imaging, and CACTI imager [17] for high-speed imaging. All of 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.

3 Basics of Lensless Imaging

We present the basic image formation models underlying lensless imaging systems. For brevity, the equations 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.

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 Fig. 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.

3.1 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,

$$b(x) = t_0(x_0)\; a\left(x + (x_0 - x)\frac{d}{z_0 + d}\right). \tag{1}$$

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 (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

$$b(x) = \int_{x_0} t_0(x_0) a\left(x + \frac{x_0 - x}{z_0 + d} d\right) dx_0. \tag{2}$$

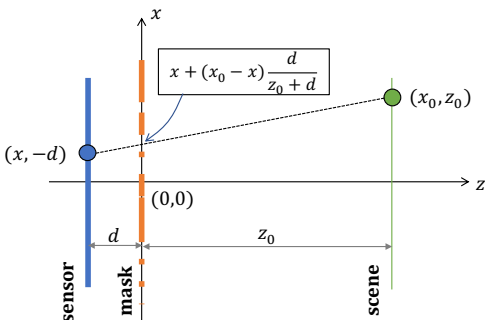


Fig. 2. *Schematic of lensless imager.* A mask is placed d distance away from the sensor. Ray from point (x_0, z_0) reaches sensor pixel $(x, -d)$ after crossing the mask at $(x + \frac{x_0 - x}{z_0 + d} d)$.

We can simplify this expression in (2) to obtain the convolution model:

$$b(x) = \widetilde{t}_0(x) * \widetilde{k}_0(x), \tag{3}$$

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 (1), in our experiments, we directly measure the kernel $\widetilde{k}(\cdot)$ which includes the effects of diffraction as shown in Fig. 3(a). More experiments verifying the convolutional model can be found in our supplementary material.

Upon discretization, the image formation model in (3) can be written as

$$\mathbf{b} = K_{z_0,a}\, \mathbf{t}_0, \tag{4}$$

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 (4). Classic mask designs based on URA, MURA and M-sequences are designed to provide an inverse that is convolutional, at least as an approximation[1]. For the approach in this paper, we use a small sized mask whose pattern 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 $K_{z_0,a}$.

3.2 Scene on Multiple Depth Planes

The image formation model in (3) and (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 formation can be written as

$$\mathbf{b} = \sum_{\ell=1}^{D} K_{z_\ell,a}\, \mathbf{t}_\ell = [K_{z_1,a} \ \cdots \ K_{z_D,a}] \begin{bmatrix} \mathbf{t}_1 \\ \vdots \\ \mathbf{t}_D \end{bmatrix}. \tag{5}$$

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

$$\min_{\mathbf{t}_1, \ldots, \mathbf{t}_D} \|\mathbf{b} - \sum_{\ell=1}^{D} K_{z_\ell,a} \mathbf{t}_\ell\|^2 + \rho(\mathbf{t}_1, \ldots, \mathbf{t}_D), \tag{6}$$

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 that requires joint estimation of a large number of unknowns are 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

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

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:

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

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, $K^{-1}_{z_1,a} K_{z_\ell,a}$ has very small spectral norm. Unfortunately, this is generally not true, as shown in Fig. 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 and high-frequency, which makes subsequent post-processing non-trivial.

It is worth nothing, that learning-based techniques could be used here given recent successes in mitigating such artifacts in the context of lensless imaging [11]. In contrast to learning-based techniques, we adopt a different strategy of improving the conditioning by acquiring multiple images using a programmable mask.

3.3 Programmable Masks

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

$$\mathbf{b}_n = \sum_{\ell=1}^{D} K_{z_\ell,a_n}\, \mathbf{t}_\ell. \tag{7}$$

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.

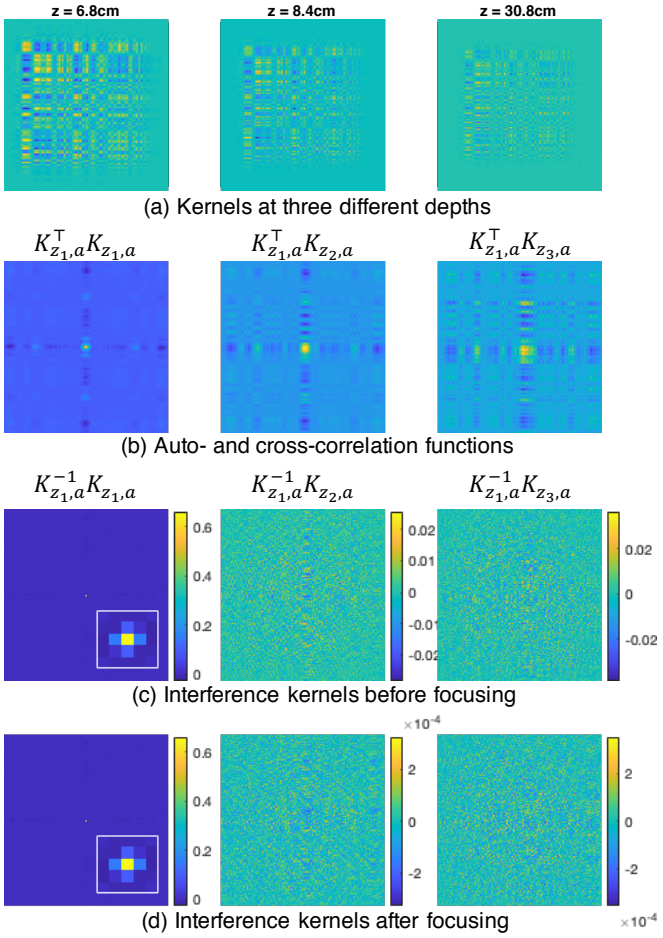


Fig. 3. *Kernels and their evolution.* Top row shows PSF for three different depth. Second row shows each PSF correlated with PSF from z_0 =6.8cm, as kernels underlying blocks in the Gram matrix from Section 4.1. Third row shows applying deconvolution kernel for PSF at z_0 on PSFs of different depth; 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 with 13×13 aperture locations across baseline area 0.78cm×0.78cm.

We can now formulate a single linear system

$$\begin{bmatrix} \mathbf{b}_1 \\ \vdots \\ \mathbf{b}_N \end{bmatrix} = \begin{bmatrix} K_{z_1,a_1} & \cdots & K_{z_D,a_1} \\ \vdots & \ddots & \vdots \\ K_{z_1,a_N} & \cdots & K_{z_D,a_N} \end{bmatrix} \begin{bmatrix} \mathbf{t}_1 \\ \vdots \\ \mathbf{t}_D \end{bmatrix}. \quad (8)$$

We can write the image formation model in (8) 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. 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 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*.

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

Digging deeper into (8), 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 our choice of the mask patterns that we use. First, since K_{z_ℓ,a_n} is a convolutional operator with some kernel, say k_ℓ, the operator $K_{z_\ell,a_n}^\top K_{z_\ell,a_n}$ is 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.

4.2 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. For simplicity in exposition, we first describe the concept in continuous domain. For $n \in \{1, \ldots, N\}$, the mask pattern is translated in steps of Δ, and $a_n$can be written as

$$a_n(x) = a_0(x - n\Delta) = a(x) * \delta(x - n\Delta). \tag{9}$$

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. That is, for scene points at depth z, their measurements translate by $n\nu_z$, where the disparity ν_z can be computed from (9) ,

$$\nu_z = \Delta(1 + d/z). \tag{10}$$

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 Focusing

Given image measurements $\{b_1(x), \ldots, b_N(x)\}$ taken with translated mask pattern a, and a focus disparity parameter ν, the focused measurement corresponding to this disparity ν is given as

$$f_\nu(x) = \frac{1}{N} \sum_{n=1}^{N} b_n(x + n\nu). \tag{11}$$

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 Fig. 4.

To better understand the effect of the focusing operator, consider the scene is restricted to a single depth $z = z_0$. Starting from (3), we can derive a simplified expression for the focused image. The captured image,

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

After translation by $n\nu$ becomes

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

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

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

where

$$\beta_{z_0}(x) = \frac{1}{N} \sum_{n=1}^{N} \delta\left(x - n\left(\nu_{z_0} - \nu\right)\right). \tag{12}$$

When $\nu = \nu_{z_0}$ then the shift applied to the image measurements cancels out that of the mask pattern, $\beta(x) = \delta(x)$. In contrast, when $\nu \neq \nu_{z_0}$, then we filter the measurement

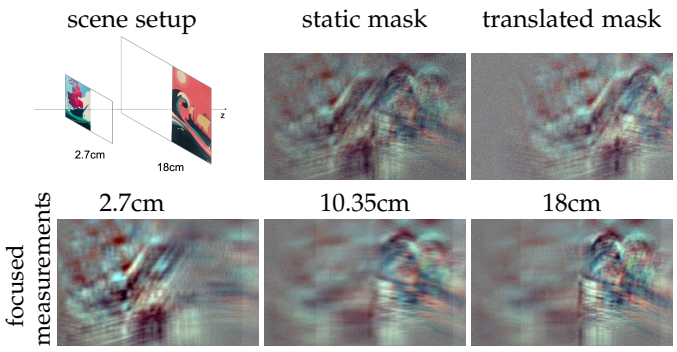


Fig. 4. *Captured and focused measurements* from our lab prototype for scene with content on two planes. Focused measurements in both are generated with 13 captures with total baseline of 0.78cm. Notice that the measurements focused at 2.7cm suppresses contribution from 18cm, and vice versa.

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 $\mathcal{S}_\nu$ which translates the input by ν pixels, where ν is real valued. With some basic algebraic manipulation we can show that

$$\mathbf{b}_n = \sum_{\ell=1}^{D} \mathcal{S}_{n\nu_{z_\ell}} K_{z_\ell,a}\ \mathbf{t}_\ell. \tag{13}$$

Hence, the focused measurements for some focus disparity ν_0 can be written as

$$\begin{aligned} \mathbf{f}_{\nu_0} &= \frac{1}{N} \sum_{n=1}^{N} \mathcal{S}_{-n\nu_0} \mathbf{b}_n = \frac{1}{N} \sum_{n=1}^{N} \mathcal{S}_{-n\nu_0} \sum_{\ell=1}^{D} \mathcal{S}_{n\nu_{z_\ell}} K_{z_\ell,a}\ \mathbf{t}_\ell \\ &= \sum_{\ell=1}^{D} K_{z_\ell,a} \left(\frac{1}{N} \sum_{n=1}^{N} \mathcal{S}_{n\nu_{z_\ell} - n\nu_0}\right) \mathbf{t}_\ell \end{aligned} \tag{14}$$

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 (5) with the key difference that the texture at depth z_ℓ is now blurred by multiple translations:

$$\mathbf{t}^*_\ell = \left(\frac{1}{N} \sum_{n=1}^{N} \mathcal{S}_{n\nu_{z_\ell} - n\nu_0}\right) \mathbf{t}_\ell.$$

Hence, while the depth z_0 corresponding to the disparity ν_0 observes no blurring, but other depths are progressively blurred depending on the values of N, Δ and ν_0.

4.4 Reconstruction from Focused Measurements

Suppose that we seek to recover the scene textures corresponding to a set of D known depth values $\{z_1, \ldots, z_D\}$. We can directly solve (8), which we refer to later as 'SweepCam full' reconstruction, and an efficient implementation of $\mathbb{K}^\top \mathbb{K}$ in D^2 convolutions is described in details in the supplementary material.

However, 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, which 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

$$\widehat{\mathbf{t}}_1 = K_{z_1,a}^{-1}\mathbf{f}_{\nu_1} = \mathbf{t}_1 + \underbrace{\sum_{\ell=2}^{D} K_{z_1,a}^{-1}K_{z_\ell,a}\mathbf{t}_\ell^*}_{\text{(reduced interference)}}. \tag{15}$$

This decoupling of the inverse problems associated with each depth vastly reduces the complexity of the recovery procedure. Fig. 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 by

$$\widehat{\mathbf{t}}_\ell = K_{z_l,a}^{-1}\mathbf{f}_{\nu_\ell} = \mathscr{F}^{-1}\left(\frac{\kappa_\ell^*\mathscr{F}\left(\mathbf{f}_{\nu_\ell}\right)}{|\kappa_\ell|^2\mathscr{F}\left(\mathbf{f}_{\nu_\ell}\right)+\lambda}\right), \tag{16}$$

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 6.2, 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.

5 Properties of SweepCam

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

5.1 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

$$a(x) = \beta(x) * \text{rect}\left(x/p\right), \tag{17}$$

with some discrete pattern $\beta(x) = \sum_k \beta_k \delta\left(x - pk\right)$. Combining (3) and (17), we observe that the effective PSF at depth z is given as

$$\widetilde{k}_z(x) = \beta\left(\frac{z}{z+d}\,x\right) * \text{rect}\left(\frac{z}{(z+d)p}\,x\right). \tag{18}$$

Thus, resolution at depth z is limited by the first null of $\mathscr{F}\left(\text{rect}\left(\frac{z}{(z+d)p}\,x\right)\right)$, 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 = 36µm and d = 1.31cm, the resolution limit is 20.14 line pairs per mm (lp/mm) at a depth of 2 cm, and 32.90 lp/mm at 1 m.

5.2 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 (12) 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 ν_{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 $\beta_z(x)$,

$$\beta_z(\omega) = \frac{1}{N} + \sum_{n=1}^{\frac{N-1}{2}} \cos\left(2\pi\omega n\Delta d\left(\frac{1}{z_0} - \frac{1}{z}\right)\right). \tag{19}$$

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 [18]. 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 5 shows an example of choosing different N and Δ with narrow and wide baseline.

5.2.1 Performance with different number of measurements

We evaluate five 3D scenes from the Middlebury 2001 stereo dataset [19], with depth quantized into 3 - 7 planes and the furthest plane mapped mapped to 12.7 cm. We simulate photon and read-out noise in our measurements using sensor parameters from our prototype. Details of preprocessing of each scene and noise generation can be found in our supplementary material. For each method, we report the

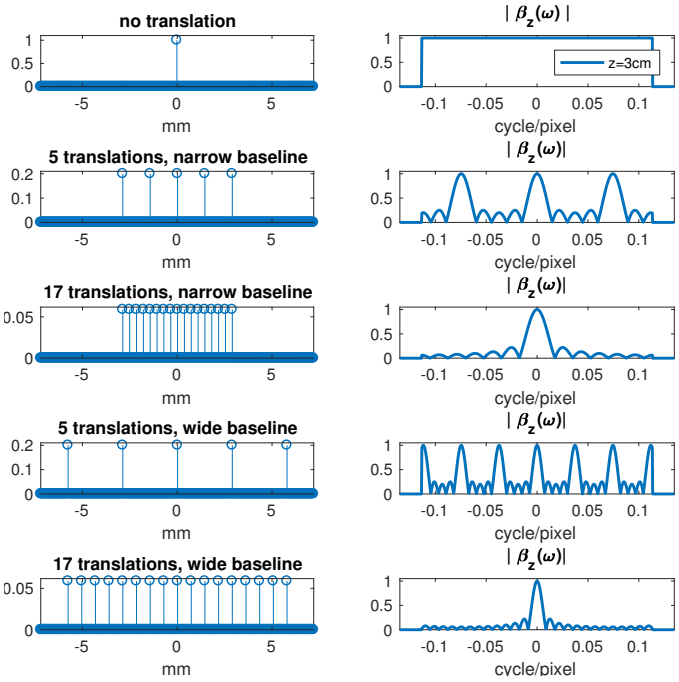


Fig. 5. *Reducing interference from other depth via focusing.* The left column shows translation patterns of the mask, while the right column shows of $|\beta_z(\omega)|$ in (19) for depth $z = 3$cm and $\nu_{z_0=4\text{cm}}$. Row two and three show more effective of suppression of interference from $z = 3cm$ as number of translations increase. Imaging parameters such as mask pixel pitch are taken from our hardware prototype, given in Section 6.

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 $\lambda \in \{0.001, 0.01, 0.1, 1, 10\}$.

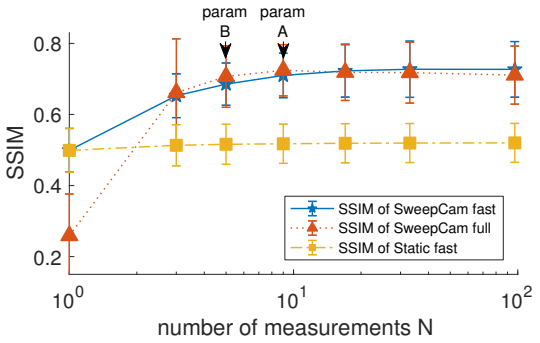
Figure 6(a) compares the performance of SweepCam fast and full reconstructions against a static mask; for fairness in comparisons, we repeat and average the static mask measurements so that the number of measurements for all three methods is the same. The baseline is kept the same, at a spread of 96 pixels, while the number of measurements is changed; further, the translation of masks is purely horizontal. While averaging increasing number of static measurements mitigates noise in the measurement, capturing more frames results in little changes in SSIM. Both SweepCam reconstructions have poor quality reconstruction for $N = 1$ as the problem of reconstructing a volume containing multiple depth planes from a single measurement is severely underdetermined, but increasing number of measurements results in a substantial increase in performance.

5.2.2 Performance at different baselines

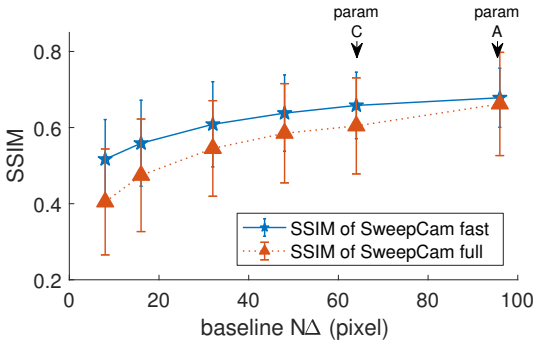
Figure 6(b) shows the performance of the 'fast' and 'full' reconstructions over different baselines $N\Delta$. We perform this by keeping the number of measurements fixed at $N = 9$ and varying Δ. As we expect, the reconstruction accuracy of both techniques increase with increasing baseline.

5.2.3 Qualitative performance

We also show qualitatively the reconstruction performance of the techniques in Fig. 6. Here we show SweepCam at three operating conditions: *parameter A* as a default setting,



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



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

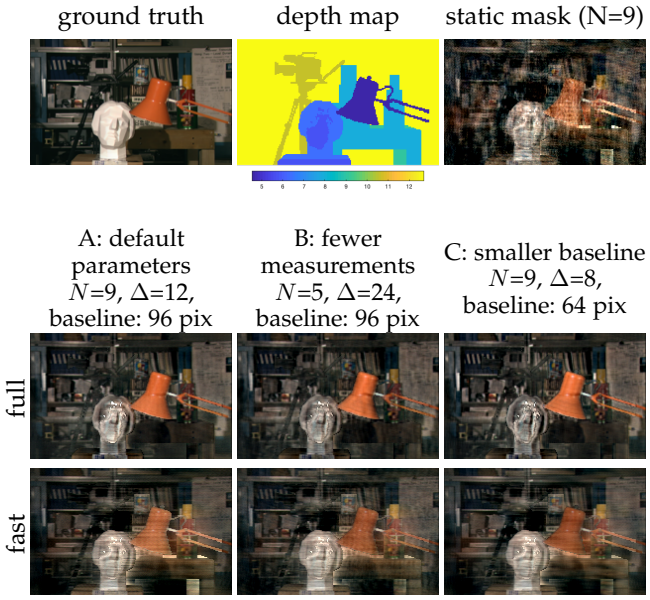


Fig. 6. *Comparison of different number of measurements and baseline on simulated data.* Left column quantitative evaluates reconstruction performance in terms of SSIM. (a) shows results for different number of measurements: increasing number of static measurements mitigates noise in the measurement but results in little changes in SSIM. SweepCam full reconstruction is severely underdetermined 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. (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 left plots. The all-in-focus images are generated by selecting pixels from the reconstructed volume with ground truth depth map.

parameter B as a setting with fewer measurements and *parameter C* with small baseline. We also show the reconstruction from the static mask for comparison. With *parameter B*, the texture suffers from the reconstruction artifacts, which is caused by the interference from other depth planes, as discussed in Section 5.2. By comparing *parameter A* and *C*, we find that the small baseline also makes it difficult to reconstruct.

We also evaluate the effect of sweep pattern, i.e. effect of different 2D translation of mask patterns, as well as provide RSNR and PSNR plots in the supplementary material; we observe that the choice of 1D vs 2D sweep patterns is scene dependent and thus, in our real experiments, we translate the mask pattern in both dimensions.

5.3 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 as (10),

$$\partial\nu = d\Delta \;\; \partial(1/z). \tag{20}$$

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 $d = 2mm$, 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.

5.4 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

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

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.

5.5 Computational Time

The average run time and average SSIM over all scenes operating with *parameter A* is shown in Table 1. 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.

TABLE 1
Average run time and quality comparison between reconstruction methods operating under *param A* of Figure 6

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

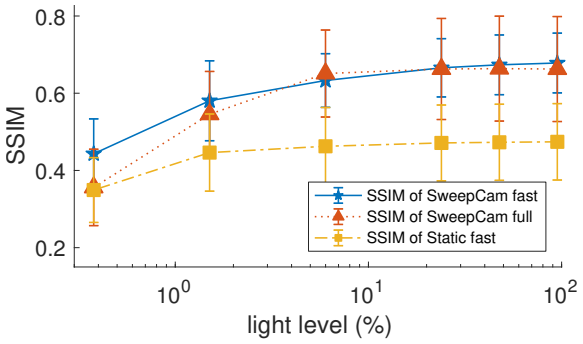


Fig. 7. *Image quality with varying light levels.* We simulate light levels in terms of the fraction of the full well capacity at the brightest pixel on the sensor. Shot noise and read noise are simulated with sensor full well capacity and dynamic range for the Sony IMX174 sensor. We observe that SweepCam fast achieves better performance under noisy conditions.

5.6 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.

5.6.1 Performance under noise

We simulate different sensor noise on SweepCam and static mask measurements, and compare their performance quantitatively in Fig. 7. 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

$$\tilde{\mathbf{b}} = \frac{G}{F}\left(\text{Poisson}\left(\frac{F}{G}\mathbf{b}\right) + \text{Normal}(0, \sigma^2)\right), \tag{21}$$

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 Fig. 7, 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.

(a) static mask (b) SweepCam fast (c) plane + wavelet l1 (d) SweepCam full (e) volume + l1 (f) volume + wavelet l1

Fig. 8. *Comparison of reconstructing with different image priors.* "Tsukuba" scene from Middlebury dataset (ground truth shown in Fig. 6) is imaged with "param A" described in Fig. 6. From left to right, we show results of reconstruction using different image priors.The left three reconstructs each depth plane separately: using static mask measurements with l2 norm, SweepCam measurements with l2 norm, and SweepCam measurements with l1 norm of wavelet coefficients respectively. The right three reconstructs the full volume jointly from SweepCam measurements, using l2 norm, l1 norm, l1 norm of wavelet coefficients on each image plane as priors respectively. As shown, the SweepCam fast algorithm achieves reasonable quality while significantly faster than the other algorithms using more sophisticated priors.

5.7 Reconstruction with Different Priors

Finally we show simulation results for different reconstruction methods in Fig. 8. We implemented solutions for traditional least squares as well as canonical and wavelet sparsity by choosing appropriate regularizing penalty function $\rho(\cdot)$ in (6). We do this for solving single depth planes separately as well as for the whole volume simultaneously. For the least squares solutions, we Wiener deconvolve when working with individual depth planes and use the conjugate gradient squared method for volume. Sparse priors, both in the canonical and wavelet bases, were implemented using backtracking FISTA [20] with zero initialization. While more sophisticated image prior result in sharper reconstructions, we observe that this comes at a cost of increased runtime.

6 Experiments on Hardware Prototype

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

Figure 9 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. Details of the calibration can be found in supplementary material.

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.

6.1 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 Fig. 10(a). With static mask measurements, directly deconvolving with PSFs at near and far planes as [2] results in artifacts in reconstruction, as

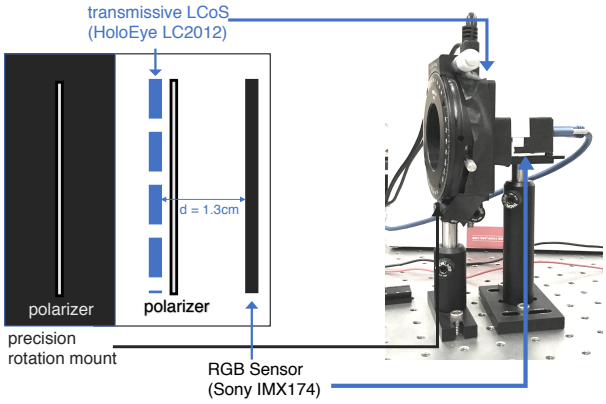


Fig. 9. *Prototype hardware setup.* The proposed design includes a programmable amplitude mask and a sensor. Our programmable mask is made of a transmissive LCoS sandwiched between two cross polarizers, one of which is mounted on precision rotation mount of optimal contrast.

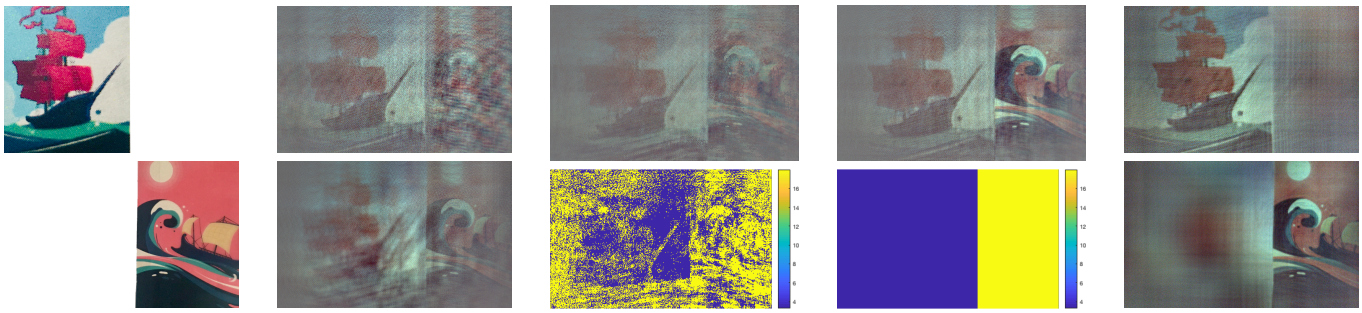
shown in Fig. 10(b). Figure 10(c) shows the reconstruction of [3], a technique that estimates both the depth map and textures jointly. We also report results of texture estimation using [3] when the depth at each pixel is known; this is shown in Fig. 10(d). Finally, Fig. 10(e) shows the SweepCam reconstructions, which provides the highest quality results with the least artifacts.

Additionally, we include the result of imaging two USAF resolution charts with static mask and SweepCam masks in the supplementary material.

6.2 Continuous depth scenes

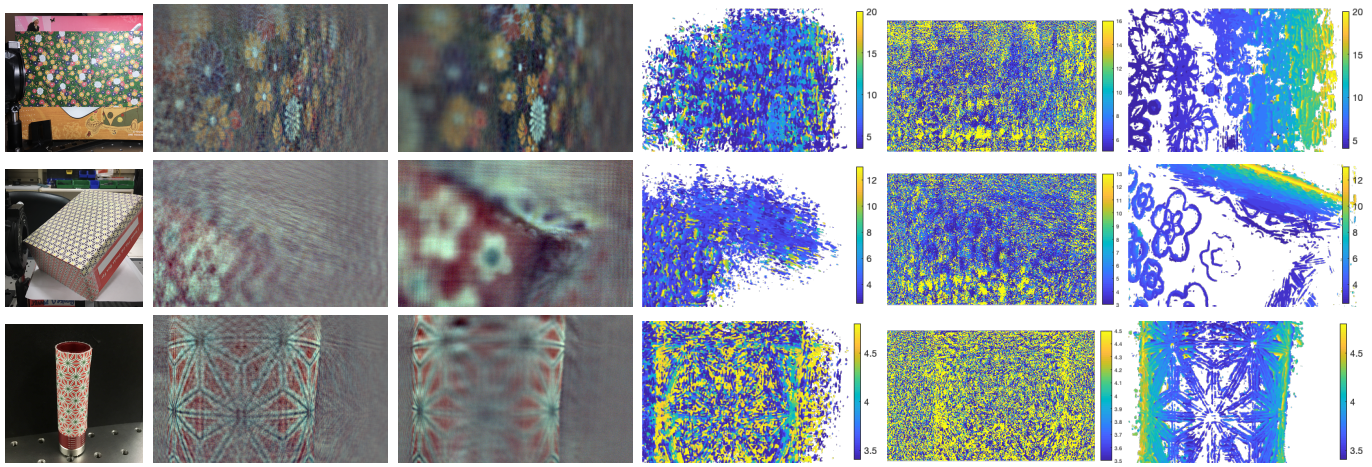
We image objects with dense textures and continuously-varying depth profiles, as shown in Fig. 11. The three objects correspond to a tilted plane, a corner of a box, and a cylinder. A focal stack with 600×960 spatial resolution and 3 channels and 34 depth planes is generated within 8 minutes with MATLAB code running on 12 core CPU following the reconstruction described in (15) 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 1 because of the increase in the number of depth planes. The full focal stacks can be found in our supplementary video.

Additionally, Fig. 11 shows depth map recovered by using depth-from-defocus algorithms on the SweepCam reconstructions, in comparison to that from static measure-



(a) scene setup (b) static fast (c) static estimate depth (d) static known depth (e) SweepCam fast

Fig. 10. *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 [2]; (c) jointly estimates texture and depth of each pixel in the scene, as [3]. (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.



(a) setup (b) static mask reconstruction (c) SweepCam fast reconstruction (d) static mask estimated depth (e) Asif [3] estimated depth (f) SweepCam fast estimated depth

Fig. 11. *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 [3] from focused measurements from SweepCam.Removing high frequency artifacts in SweepCam fast reconstruction significantly improves depth estimation, as (f) demonstrate more reliable estimation against (d) and (e).

ment 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 following algorithm in [3] 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.

6.3 General Scenes

SweepCam is able to resolve general scenes with depth variation as shown in Fig. 1 and 12. Fig. 12 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.

7 Discussion and Conclusion

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 [21], [22]. We could augment (2) in our forward model to be

$$b(x) = \int_{x_0} t_0(x_0) v_0(x, x_0) a\left(x + \frac{x_0 - x}{z_0 + d} d\right) dx_0. \qquad (22)$$

where $v_0(x, x_0)$ indicates visibility of scene point at (x_0, z_0) from sensor location x. In addition to $v(x, x_0)$ being high dimensional, solving for both t_0 and v is no longer a linear problem. However, an important benefit of this modeling is that secondary effects that break the convolution model,

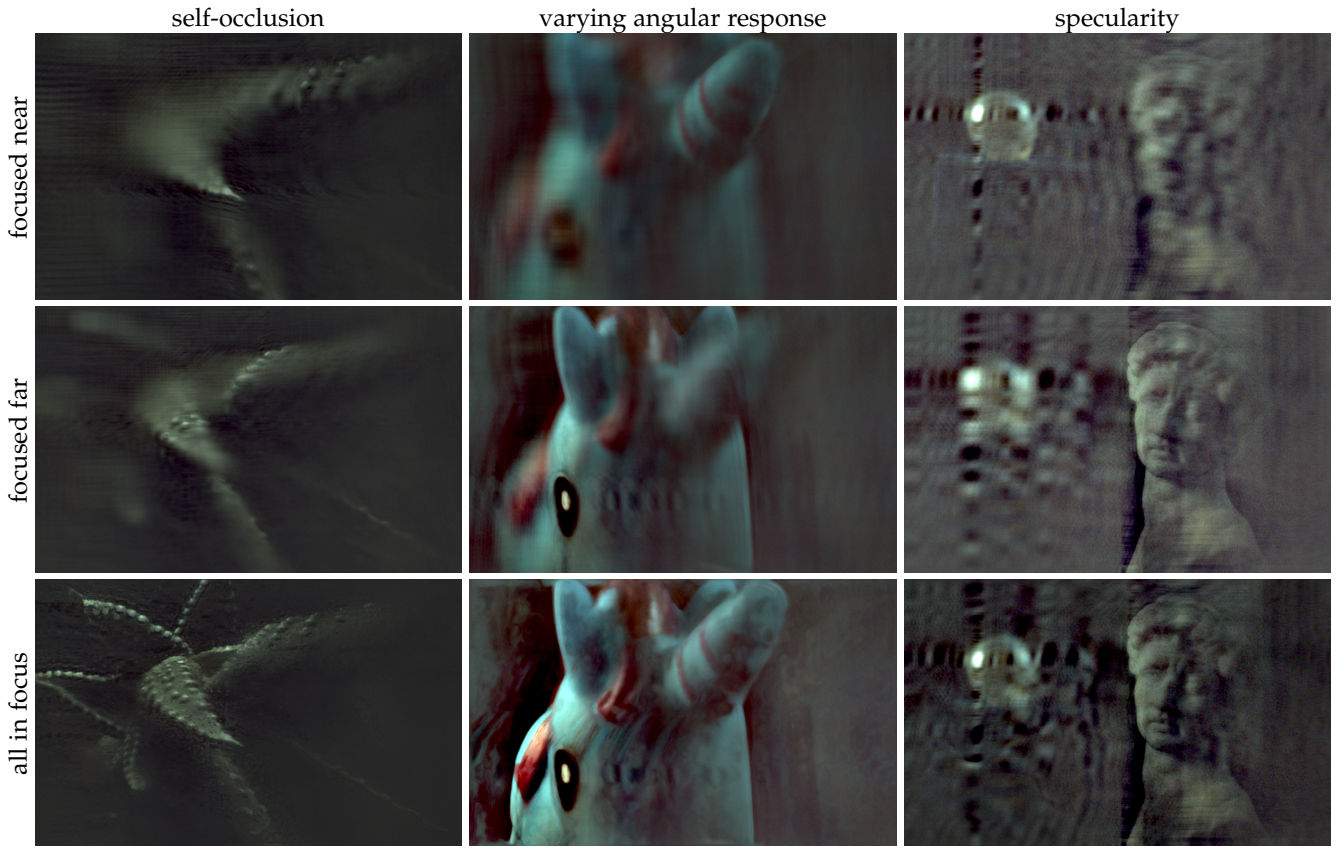


Fig. 12. *General scenes* that deviate from the convolution model. Each of the three scenes violet the assumptions underlying the image formation model, either in the form of occlusions between depth planes, or due to materials with non-Lambertian reflectance and directional lighting in the scene. In spite of these model mismatches our techniques work reliably, except perhaps for artifacts that are spatially localized. The lensless focal stacks can be found in supplementary video.

including occlusion and specularity, can be accounted for in the ensuing non-linear optimization. This would invariably require iterative solutions and good initializations, perhaps using the proposed method.

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 [23].

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 5. Previous compressive temporal imagers [17] 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.

Acknowledgments

We thank Ashok Veeraraghavan for helpful discussions and for generously donating the transmissive SLM used in the prototype. This work was supported by a Sony research contract, the NSF award IIS-1618823, and the NSF Expeditions award 1730147.

References

[1] V. Boominathan, J. K. Adams, M. S. Asif, B. W. Avants, J. T. Robinson, R. G. Baraniuk, A. C. Sankaranarayanan, and A. Veeraraghavan, "Lensless imaging: A computational renaissance," *IEEE Signal Processing Magazine*, vol. 33, no. 5, pp. 23–35, 2016.

[2] M. S. Asif, A. Ayremlou, A. C. Sankaranarayanan, A. Veeraraghavan, and R. G. Baraniuk, "Flatcam: Thin, lensless cameras using coded aperture and computation," *IEEE Transactions on Computational Imaging*, vol. 3, no. 3, pp. 384–397, Sept 2017.

[3] M. S. Asif, "Lensless 3d imaging using mask-based cameras," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2018, pp. 6498–6502.

[4] J. K. Adams, V. Boominathan, B. W. Avants, D. G. Vercosa, F. Ye, R. G. Baraniuk, J. T. Robinson, and A. Veeraraghavan, "Single-frame 3d fluorescence microscopy with ultraminiature lensless flatscope," *Science advances*, vol. 3, no. 12, p. e1701548, 2017.

[5] N. Antipa, G. Kuo, R. Heckel, B. Mildenhall, E. Bostan, R. Ng, and L. Waller, "Diffusercam: lensless single-exposure 3d imaging," *Optica*, vol. 5, no. 1, pp. 1–9, Jan 2018.

[6] J. Tanida, T. Kumagai, K. Yamada, S. Miyatake, K. Ishida, T. Morimoto, N. Kondou, D. Miyazaki, and Y. Ichioka, "Thin observation module by bound optics (tombo): concept and experimental verification," *Applied optics*, vol. 40, no. 11, pp. 1806–1813, 2001.

[7] P. R. Gill and D. G. Stork, "Lensless ultra-miniature imagers using odd-symmetry spiral phase gratings," in *Imaging and Applied Optics*, 2013, p. CW4C.3.

[8] R. T. Collins, "A space-sweep approach to true multi-image matching," in *Proceedings CVPR IEEE Computer Society Conference on Computer Vision and Pattern Recognition*. IEEE, 1996, pp. 358–363.

[9] J. Tan, L. Niu, J. K. Adams, V. Boominathan, J. T. Robinson, R. G. Baraniuk, and A. Veeraraghavan, "Face detection and verification using lensless cameras," *IEEE Transactions on Computational Imaging*, vol. 5, no. 2, pp. 180–194, 2018.

[10] T. Nguyen Canh and H. Nagahara, "Deep compressive sensing for visual privacy protection in flatcam imaging," in *Proceedings of the IEEE International Conference on Computer Vision Workshops*, 2019, pp. 0–0.

[11] S. S. Khan, V. R. Adarsh, V. Boominathan, J. Tan, A. Veeraraghavan, and K. Mitra, "Towards photorealistic reconstruction of highly multiplexed lensless images," in *Proceedings of the IEEE International Conference on Computer Vision*, 2019, pp. 7860–7869.

[12] Y. Zheng and M. S. Asif, "Joint image and depth estimation with mask-based lensless cameras," *arXiv preprint arXiv:1910.02526*, 2019.

[13] A. Zomet and S. K. Nayar, "Lensless imaging with a controllable aperture," in *2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'06)*, vol. 1, June 2006, pp. 339–346.

[14] M. F. Duarte, M. A. Davenport, D. Takhar, J. N. Laska, T. Sun, K. F. Kelly, and R. G. Baraniuk, "Single-pixel imaging via compressive sampling," *IEEE signal processing magazine*, vol. 25, no. 2, pp. 83–91, 2008.

[15] A. Wagadarikar, R. John, R. Willett, and D. Brady, "Single disperser design for coded aperture snapshot spectral imaging," *Applied optics*, vol. 47, no. 10, pp. B44–B51, 2008.

[16] D. Kittle, K. Choi, A. Wagadarikar, and D. J. Brady, "Multiframe image estimation for coded aperture snapshot spectral imagers," *Applied optics*, vol. 49, no. 36, pp. 6824–6833, 2010.

[17] P. Llull, X. Liao, X. Yuan, J. Yang, D. Kittle, L. Carin, G. Sapiro, and D. J. Brady, "Coded aperture compressive temporal imaging," *Optics express*, vol. 21, no. 9, pp. 10 526–10 545, 2013.

[18] E. Hecht, *Optics*, fifth edition ed. Pearson, 2017.

[19] D. Scharstein and R. Szeliski, "A taxonomy and evaluation of dense two-frame stereo correspondence algorithms," *International Journal of Computer Vision*, vol. 47, no. 1, pp. 7–42, Apr 2002. [Online]. Available: https://doi.org/10.1023/A:1014573219977

[20] A. Beck and M. Teboulle, "A fast iterative shrinkage-thresholding algorithm for linear inverse problems," *SIAM journal on imaging sciences*, vol. 2, no. 1, pp. 183–202, 2009.

[21] J. T. Kajiya, "The rendering equation," in *Proceedings of the 13th annual conference on Computer graphics and interactive techniques*, 1986, pp. 143–150.

[22] P. Dutre, P. Bekaert, and K. Bala, *Advanced global illumination*. AK Peters/CRC Press, 2006.

[23] K. Ma, H. Li, H. Yong, Z. Wang, D. Meng, and L. Zhang, "Robust multi-exposure image fusion: a structural patch decomposition approach," *IEEE Transactions on Image Processing*, vol. 26, no. 5, pp. 2519–2532, 2017.

Yi Hua is a Ph.D. student in ECE Department at Carnegie Mellon University. She received her B.Sc in computer science in 2015 from Rice University, Houston, TX, USA, and M.S. in Computer Vision from Carnegie Mellon University, PA, USA in 2016. Her research interests are computational imaging and computer vision.

Shigeki Nakamura received the B.E. and M.E. degrees from the University of Tokyo, Japan, in 2010 and 2012, respectively. In 2012, he joined Sony Corporation, Japan, where he is involved in research and development of image processing algorithms. Since 2016, he has worked for Sony Semiconductor Solutions Corporation, Japan. He studied as a visiting researcher at Carnegie Mellon University in 2019. His research interests include computational photography, computer vision, and image processing.

M. Salman Asif received his B.Sc. degree in 2004 from the University of Engineering and Technology, Lahore, Pakistan, and the M.S.E.E degree in 2008, and the Ph.D. degree in 2013 from the Georgia Institute of Technology, Atlanta, GA, USA. He was a Senior Research Engineer at Samsung Research America, Dallas, TX, USA, from August 2012 to January 2014, and a Postdoctoral Researcher at Rice University from February 2014 to June 2016. He is currently an Assistant Professor in the Department of Electrical and Computer Engineering at the University of California, Riverside. He received Hershel M. Rich Outstanding Invention Award in 2016, UC Regents Faculty Fellowship Award in 2017, and Google Faculty Award in 2019. His research interests broadly lie in the areas of information processing and computational sensing with applications in signal processing, machine learning, and computational imaging.

Aswin C. Sankaranarayanan is an Associate Professor in the ECE Department at Carnegie Mellon University (CMU). He earned his Ph.D. from University of Maryland, College Park and was a post-doctoral researcher at the DSP group at Rice University before joining CMU. Aswin's research spans topics in imaging, vision, and image processing. He is the recipient of the CVPR 2019 best paper award, the CIT Dean's Early Career Fellowship in 2018, the NSF CAREER award in 2017, the Eta Kappa Nu (Sigma chapter) Excellence in Teaching award in 2017, the 2016 Herschel M. Rich invention award, and the distinguished dissertation fellowship from the ECE Department at University of Maryland in 2009.