**Title**

Unknown Motion Calibration and Dynamic Imaging Reconstruction



**Author**

Cao, Ruiming

**Publication Date**

2024

Peer reviewed|Thesis/dissertation

# Unknown Motion Calibration and Dynamic Imaging Reconstruction

by

Ruiming Cao

A dissertation submitted in partial satisfaction of the

requirements for the degree of

Joint Doctor of Philosophy
with University of California, San Francisco

in

Bioengineering

in the

Graduate Division

of the

University of California, Berkeley

Committee in charge:

Professor Laura Waller, Chair
Professor Daniel A. Fletcher
Professor Na Ji
Professor Bo Huang

Fall 2024

Unknown Motion Calibration and Dynamic Imaging Reconstruction

Abstract

Unknown Motion Calibration and Dynamic Imaging Reconstruction

by

Ruiming Cao

Doctor of Philosophy in Bioengineering

University of California, Berkeley

Professor Laura Waller, Chair

Most imaging systems were developed to capture images for static objects that do not move during the image acquisition time. As a result, motion is considered as a primary source of imaging artifacts, which limits the observation of fast moving samples. The most common way to suppress motion artifacts is to shorten the acquisition time, thereby minimizing motion during the observation. However, this comes with a cost of less signal and increased noise in the measurements. While most imaging systems assume static scenes and well-calibrated system motion during image acquisition, this thesis pioneers an alternative approach that algorithmically estimates unknown motion. With accurate motion estimation, we can computationally correct motion artifacts during the image reconstruction process, opening opportunities to design imaging systems specifically for dynamic scenes. The core idea of this thesis is to simultaneously reconstruct both the object and its motion using optimization techniques. We find this approach to be effective and versatile, demonstrating it across various imaging modalities, object scales, and applications. This joint object and motion optimization can be done in post-processing without altering image acquisition, enabling the reconstruction of dynamic scenes from existing datasets affected by motion artifacts.

In addition to recovering dynamic information from static imaging systems, we also explore an opposite problem: recovering static scenes using an event camera that only detects changes in the scene. By studying the triggering mechanism of noise events, we develop a statistical noise model for the event camera that explains its illuminance-dependent noise characteristics. With this understanding, we propose to form an image of a static scene using only noise events, providing rich contextual information about static scenes to the dynamic sensor, without requiring any change on its hardware.

This thesis demonstrates these concepts using various novel imaging systems, including an event camera, a lensless camera, a quantitative phase microscope, a 3D refractive-index microscope, and a super-resolution fluorescence microscope. By accurately modeling both

unknown motion and noise, we aim to demonstrate how computational methods can bridge the gap between static and dynamic imaging.

To my family

# Contents

# List of Figures

# List of Tables

## Acknowledgments

First and foremost, I would like to thank my PhD advisor Laura Waller for her support and optimism toward my study. Laura accepted me as a student even though she knew I knew absolutely nothing about optics when I started. Since then, I have learned so much and have been constantly inspired by her. Laura has shielded me very well from external pressure, so that I had the unparalleled privilege of exploring science freely throughout my time in graduate school.

I am also greatly indebted to Gokul Upadhyayula for his mentorship on research and beyond. Gokul persuaded me to apply neural space-time model to the existing super-resolution microscope method and offered me tremendous help in the experiment and computation, which eventually led to a big success. I would also like to thank Bo Huang, Dan Fletcher, Na Ji, who are in my thesis committee, and also Miki Lustig who is in my qualifying exam committee. They have offered invaluable guidance for my research. I had the pleasure to teach with Steve Conolly as a teaching assistant. I want to thank my fellow TA, Eric Markley, for carrying me through the class and teaching me a lot about analog circuits.

I joined Waller lab when there were many senior graduate students and postdocs around, and they gave me extremely valuable mentorship and advice. I want to thank Michael Kellman for mentoring me on my first project and sharing a lot of research ideas with me, David Ren for answering my questions about tomography and helping me to fix a lab computer, Regina for the weekly chat to keep me sane during the deep isolation of the year-long shelter-in-place order, Li-Hao for sharing valuable insights on speckle illumination work and making things extremely organized, and Kristina, Shweta, Yi for always sharing on their awesome research and career experiences. This list can go very long if I don't just stop here. I had wonderful memories in Cory 558 office. I really enjoyed the science and non-science conversations with my cubicle mates (Linda, David, and later Chaoying, Dekel) and also everyone else in the Waller lab. I also like that many conversations started with science/research related topics would become eventually a gossip chat in many long afternoons. So thank you to everyone from Waller lab for contributing to these memories! I was fortunate to join Laura's group together with Neerja and Eric, and we have shared a lot of good memories and graduate school/life grind. It is also memorable to have spent countless days and nights in our Cory 151M lab. I often struggled to even make the most basic things to work, but Linda and Guanghan helped me a lot on building and aligning the speckle structured illumination microscope. I had a lot of discussions and brainstorm with Neerja and Kevin Zhou who I often met in the lab. I had a great and productive time working with Dekel and Amit on the event camera project. I want to thank the brilliant undergraduate students who I've worked with, including Kevin Tandi, Jin Wei Wong, Vi Tran, Shamus Li, David Martinez, Quang Nguyen, and Qijun Li.

Outside of the Waller lab, I was also very grateful to overlap with many awesome researchers throughout the years. I would like to thank Jonathan Dong, Kuan-Chen Shen for letting me involved their dark-field LED project, Nikita Divekar and James Nuñez for engineering robust cells for imaging experiments, Hunter Johnson for cell samples, Xiongtao

Ruan for his help on Slurm, Gaoxiang Liu for teaching me how to dye cells, and a lot more people for inspiring discussions. I want to thank Alison Killilea and Sara Sosa from the Cell Culture Facility and Denise Schichnes and Steven Ruzin from the Biological Imaging Facility for their help on cell preparation and imaging experiments. I also want to thank Rocío and Kristin from Bioengineering Department who have always been super helpful and responsive. I was fortunate to spend great summers in Google Research and Meta Reality Labs Research, and I want to thank my mentors Michael Brenner and Hui Chen for recruiting and hosting me.

I want to thank my wonderful friends in Berkeley who made the journey fun and meaningful. Thank you to Ke, Rebekah, Yuhan for some fun hiking. Thank you to Geng and Zoey for planning awesome adventures. Thank you to Qianyi for driving me to Lake Tahoe to ski at 5am and driving me back at 10pm. Thank you to Qing for taking me to bike trips. Thank you to my BioE friends, Jiaang, Yuheng, Fei, Tao, Charlene for taking me to great dinner spots in SF. I want to thank my college and high school buddies for spending time with me on fun activities. Especially I want to thank Gengming for telling me that he thought using LED array in microscopy is fun before I started at Berkeley, which encouraged me to reach out to Laura despite knowing nothing about optics. I want to thank Jeff and Susan for inviting me to their house during every Thanksgiving and New Year Eve and being my family away from home.

In the end, I would like to thank Mom and Dad and my extended family for being my strongest advocates. My parents always answer my calls anywhere anytime despite the 15-hour time difference. They give me unconditioned love and support throughout different stages of my life and whenever it is the most needed. My achievement is really a celebration of their efforts and wisdom.

# Chapter 1

# Introduction

Imaging systems play a crucial role in various aspects of our lives. For example, people take photos to capture moments, doctors use medical imaging to diagnose diseases, and biologists observe sub-cellular components using fluorescent microscopy. A modern imaging system is typically designed to perform two core functions: image formation and signal collection. In the context of optical imaging systems, the former is achieved through a combination of a light source and optics, and the latter through imaging sensors. In a nutshell, light encodes the information of the target scene, passes into the optics to be collected by the sensor, and the photons arriving at the sensor pixels are consequently converted into electrical signals and recorded by a digital system.

Computational algorithms have become a fundamental component in many imaging systems over the past two decades. Once the signal is captured by the sensor, computational software processes the raw data to reconstruct the final image, which would be difficult to capture using optical hardware alone. Smartphone photography is a good example that highlights the success of these computational algorithms. Over the past decade, the image quality of smartphone cameras has significantly improved due to various algorithms such as motion metering [100], high-dynamic range [123] and low light [70], despite the physical constraints of compact optics and sensor pixels. Computational imaging involves the joint development of signal acquisition methods and image reconstruction algorithms. Beyond digital photography, computational imaging systems have significant impacts in scientific and biomedical imaging. For example, the scan time of magnetic resonance imaging (MRI) can be accelerated by multiple fold when using custom phase gradient and compressive sensing-based optimization [108].

Despite these advances, capturing fast dynamic scenes is often a challenge. Broadly speaking, most imaging systems are designed for static scenes, and they often produce artifacts when the scene changes during a single acquisition. A famous example is the Temple Boulevard photograph (Fig. 1.1), one of the earliest photographs ever taken. It captured only the shoeshiner and the customer (red box) while missing all other pedestrians and cars because they did not stay still during the long exposure of 4-5 minutes required by the daguerreotype photographic process. Digital CMOS sensors also produce distortions in

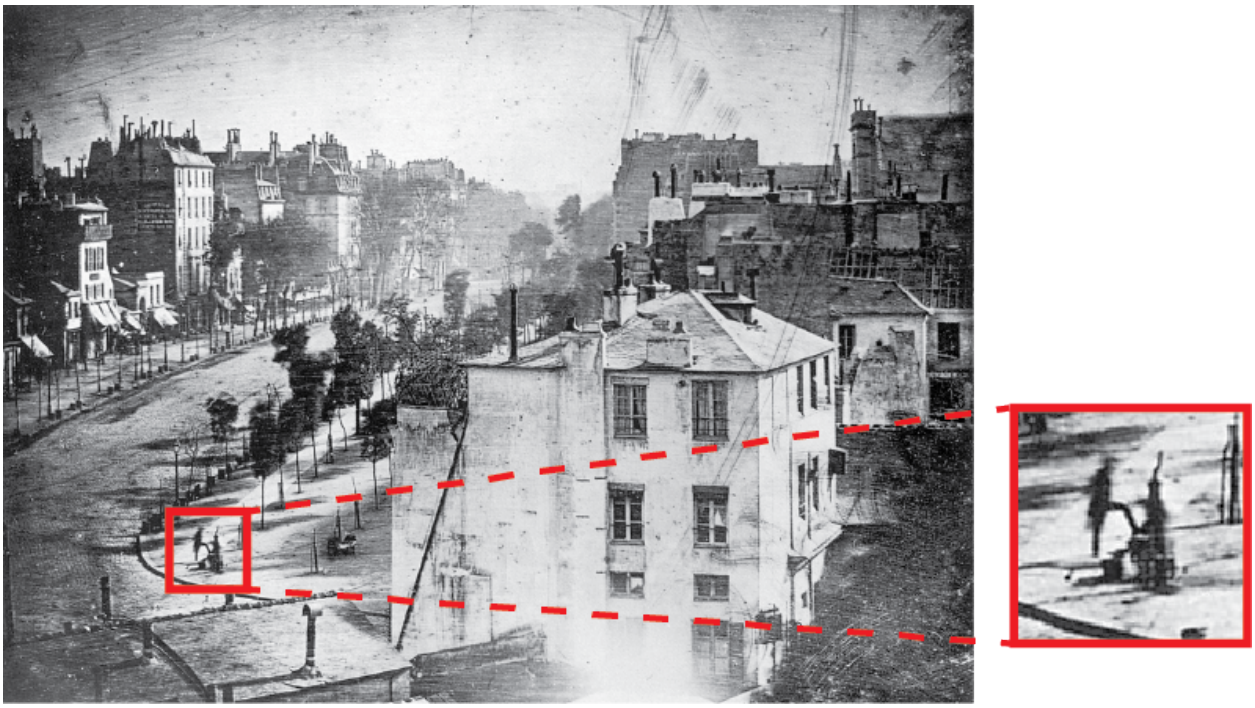

Figure 1.1: The famous Boulevard du Temple photograph taken by Louis Daguerre in 1838. Even though it was captured at a busy hour in the morning, the street appeared empty since the early photographic process requires a long exposure time (4-5 minutes in this case). Hence, the shoeshiner and the customer (red box) were the only persons captured by this photograph.

fast-moving scenes due to the sequential rolling-shutter readout of the camera circuit (see Fig. 1.2 and Fig. 1.3). One solution to motion-related artifacts is to reduce the acquisition time. While a shorter acquisition time is effective at reducing motion artifacts, it often limits the amount of information captured and can result in noisier raw data. In other words, reducing the acquisition time is constrained by physical limits, which can pose empirical challenges.

Motion also presents challenges for medical imaging techniques like computed tomography (CT) and magnetic resonance imaging (MRI). Patient movements, including voluntary motion (such as moving an arm or shifting position) and involuntary motion (such as breathing and heartbeat), can both introduce artifacts to the final image, affecting the clinical interpretations [207]. As the motion is often periodic, motion gating methods are commonly used to reject signals that can cause motion artifacts. Motion gating methods synchronize the scanning with patient's respiratory or cardiac motion cycle and only collect signal at a specific phase in multiple motion cycles [46]. External devices, such as chest motion sensors or electrocardiography (ECG) are often deployed to detect the specific phases of the mo-

Figure 1.2: A de Havilland Canada Dash 8 Q-400 six-blade propeller with severe rolling-shutter distortion. This photo was taken by Richard F. Lyon using a Google Pixel 3 camera [50].

Figure 1.3: Cessna 172 two-blade propeller with severe rolling-shutter distortion. This photo was taken by Laura Waller using an iPhone camera [182].

tion cycle [95]. Additionally, certain MRI techniques can measure the motion cycle without external sensors. These methods include an additional navigator echo to the MRI pulse sequence, and the signal from the navigator echo will help the system determine the best time to acquire imaging data [45, 191, 96].

This thesis takes a different approach to the motion challenge: instead of assuming a static scene and working to maintain this assumption, we acknowledge the unknown motion in the raw data and estimate it computationally. Chapter 2 introduces an illumination scheme to capture motion-aware signal and use that to calibrate for unknown axial motion of the imaging system. In Chapter 4, the motion is deemed as an encoding mechanism for diversified measurements. This approach eliminates the need to shorten acquisition time, instead framing the motion challenge as an optimization problem. Without assuming a static scene, this thesis addresses the challenge of simultaneously solving for both the image and its motion, as both variables are unknown. Resolving the motion not only eliminates artifacts in the final image but also allows us to observe dynamics at speeds beyond the normal capabilities of the imaging system, as demonstrated in Chapter 5.

Previous work has explored methods to estimate unknown global motion during image reconstruction. The global motion is often caused by sample drifting in the context of super-resolution microscopy. The drifting motion (global translation motion) can often be identified using spatial cross-correlation between different raw images, and the effect of the motion can then be corrected during the reconstruction [117, 137, 68]. In contrast to global motion, local deformable motion involves complex, nonlinear transformation, requiring a large number of parameters for accurate modeling. As a result, accounting for deformable motion during image reconstruction is particularly difficult due to its complexity to model and the large number of parameters to estimate for.

From the perspective of inverse problems, while global drifting motion can often be resolved, introducing unknown deformable motion parameters complicates the problem and can make it ill-posed. Since deformable motion occurs at the pixel level, the number of parameters needed to represent it is comparable to those required to represent the image. Thus, given the same set of measurements used for reconstructing a final image, jointly estimating the unknown motion during the image reconstruction makes the inverse problem less well-conditioned and cannot work for all cases. In Chapters 4 and 5, we explore the assumption of motion smoothness to mitigate the ill-posedness and enable the joint motion and image optimization.

This thesis features a number of optical imaging modalities, including quantitative phase contrast, 3D refractive index tomography, 2D/3D super-resolution fluorescent imaging, lensless photography, and event-based cameras. While they have different designs and uses, we often consider them under the same inverse problem framework:

$$y = Ax \tag{1.1}$$

The measured signal is denoted as $y$, and the scene we want to recover is $x$. $A$ is the physical forward model of the imaging system which connects the measured signal with the final reconstruction. When dealing with dynamic scenes, the inverse problem is rewritten as:

$$y\left(t\right) = Ax\left(t\right), \tag{1.2}$$

where both the signal and the reconstruction vary with time. There are two key aspects for effectively solving inverse problems: The first is encoding information into measurements through a favorable design of the forward model $A$. The second is decoding the information, particularly when the inverse problem is ill-posed, by establishing appropriate priors for the scene $x$.

The remainder of this chapter introduces the imaging modalities that will be discussed in greater detail in the following chapters.

## 1.1 Quantitative phase imaging

Quantitative phase imaging (QPI) measures a sample's phase delay, which is directly related to optical path length. Since phase cannot be measured directly, QPI methods use various phase contrast mechanisms to encode phase information into the captured intensity measurements. These measurements are then used to calculate the phase. Several established QPI methods include digital holographic microscopy [88], ptychography [149], and spatial light interference microscopy [187], each taking unique approaches. Most of these methods use coherent light. However, methods using partially spatially coherent light [183, 175, 127] can offer higher spatial resolution [14], reduced coherence-induced speckle [56], and are often less expensive.

QPI techniques provide high-contrast, label-free imaging, making them suitable for various scientific and industrial applications [142]. They enable observation of transparent

samples, such as living cells, without the need for exogenous contrast agents. This preserves the natural state of the specimens and allows for continuous real-time observation.

## 1.2 Super-resolution imaging

The spatial resolution of any far-field optical imaging system is fundamentally limited by diffraction, which depends on the system's numerical aperture (NA) and light wavelength. Super-resolution microscopy allows observation of sub-cellular and molecular structures at resolutions beyond the diffraction limit, providing unprecedented insights into biological processes. This breakthrough was recognized by the 2014 Nobel Prize in Chemistry [118, 199].

There are three fundamental approaches for super-resolution microscopy: stimulated emission depletion (STED) microscopy [71], single-molecule localization microscopy (SMLM) [12, 150], and structured illumination microscopy (SIM) [67]. STED microscopy reduces the effective point spread function by depleting fluorophores around the focal point. SMLM achieves super-resolution by precisely localizing sparsely located individual fluorophores over time. SIM enhances resolution by projecting patterned light onto the sample and using computational algorithms to reconstruct high-resolution images from the resulting moiré patterns.

These methods have revolutionized biological research by providing insight into the intricate workings of cells and their components [158]. For instance, super-resolution imaging allows scientists to visualize the spatial organization [78] and interactions of molecular structures in cells at nanometer-resolution [128]. This has led to breakthroughs in understanding cellular processes such as membrane organization [78], protein interactions [155], and intracellular transport mechanisms [6].

## 1.3 Event camera

Event cameras [109], also known as neuromorphic cameras or dynamic vision sensors, are an emerging imaging modality for capturing dynamic scenes. Their ability to capture data at a much faster rate than conventional cameras has led to their application in high-speed navigation, augmented reality, and real-time 3D reconstruction [57]. Unlike conventional CMOS cameras, which output intensity images at fixed intervals, event cameras detect brightness changes at each pixel asynchronously. When the brightness change at a pixel exceeds some threshold, an event is recorded. The output from an event consists of three elements: a timestamp, the spatial coordinate of the triggered pixel, and a binary polarity, indicating whether it is an increase or decrease of brightness.

## 1.4 Outline

Here is an outline of this thesis:

- In Chapter 2, we discuss a practical method for 3D differential phase contrast (DPC) phase microscopy to estimate the axial positions of a defocused image stack when a precise motion stage is not available. This chapter includes the research published in [23, 28].
- Chapter 3 is a detour from the motion estimation and explores a systematic way to optimize the illumination patterns to improve the information encoding efficiency of an imaging system. This chapter includes the research published in [24, 28].
- In Chapter 4, we propose a novel dynamic super-resolution microscopy method that uses fixed random speckle illumination and takes sample motion as a contrast mechanism to encode diversified information. This chapter covers the research published in [25, 22, 29].
- In Chapter 5, we propose a motion-resolved reconstruction method called the neural space-time model, which jointly estimates the scene and its motion dynamics for multi-shot imaging systems. This model effectively improves the temporal resolution of systems designed for static imaging. This chapter covers the research published in [26].
- In Chapter 6, we investigate the event camera, an emerging dynamic sensor that detects only brightness changes, and introduce a Noise2Image method to recover the absolute brightness of a scene from the noise characteristics. This chapter covers the research published in [27].
- Chapter 7 serves as a conclusion and provides some future directions and outlook.

# Chapter 2

# Self-calibrated 3D differential phase contrast microscopy

Differential phase contrast (DPC) microscopy is a practical QPI method that recovers quantitative phase from multiple images with different illumination source patterns [114, 175]. The illumination source diversity can be conveniently achieved with a programmable LED array [212, 175, 105]; thus, QPI is enabled by a simple and inexpensive modification to a commercial microscope. The LED array microscope is, in general, a powerful platform for computational illumination microscopy, enabling not only QPI, but also multi-contrast [212, 176], super-resolution [211, 177] and 3D imaging [174].

Since phase is a projected quantity related to both the refractive index (RI) and thickness of the sample, 3D phase imaging amounts to volumetric reconstruction of the sample's RI [196, 14]. Interferometric and diffraction tomography techniques [171, 89, 13, 174, 76, 35, 33, 98], as well as 3D Fourier Ptychography [174, 112], reconstruct 3D RI from projection measurements captured at different illumination angles with spatially coherent light. 3D DPC [32, 162], on the other hand, uses partially coherent illumination (e.g. LED array illumination with many LEDs on) to create strong depth sectioning effects [156] that blur out-of-focus planes [175]. The sample is then scanned axially to image the third ($z$) dimension. This approach is practical because it gives good signal and does not require well-aligned illumination [35, 98, 44]; however, it does require an axial motion stage, which increases hardware complexity and cost.

Axial scanning for 3D DPC is usually performed by an automated motion stage which stops at each defocus plane [32]. This "stop-and-stare" strategy limits the overall speed of capture [140] because the camera must wait for the motion stage to move and settle before capturing each frame. Fast focusing mechanisms like focus-tunable lenses [162] can improve capture speed, but are expensive to implement. For high-NA systems, motion stages are particularly expensive since the depth-of-field (DoF) is short, so high-precision axial motion is required [162].

---

This chapter includes the research I presented or published in [23, 28].

Here, we present an extension of 3D DPC that enables fast axial scanning without a dedicated motion stage. Instead, we hand-turn the focus knob of the microscope to scan through focus while capturing video measurements, and then use an algorithmic self-calibration procedure to solve for the defocus positions in post-processing. The system capture speed is increased because the images are taken during the scanning motion, and the overall cost of the system decreases significantly without the need for an axial motion stage.

## 2.1 Background on 3D differential phase contrast imaging

An inhomogeneous 3D volume can be written as a scattering potential, $V$, such that

$$V(\boldsymbol{r}) = k_0^2 \left(n_0^2 - n^2(\boldsymbol{r})\right), \tag{2.1}$$

where $\boldsymbol{r}$ denotes the 3D spatial coordinate, $k_0$ is the wave number, $n_0$ is the RI of the surrounding medium, and $n(\boldsymbol{r})$ is the complex RI of the sample (real part for phase and imaginary part for absorption). When coherent light propagates through the 3D volume, we can write the evolution of the electric field using the Lippmann-Schwinger equation:

$$U(\boldsymbol{r}) = U_{in}(\boldsymbol{r}) + U_{scat}(\boldsymbol{r}) = U_{in}(\boldsymbol{r}) + \iiint U(\boldsymbol{r}') V(\boldsymbol{r}') G(\boldsymbol{r} - \boldsymbol{r}') d^3\boldsymbol{r}', \tag{2.2}$$

where $U_{in}$, $U_{scat}$ are the incident and scattered light and $G$ is the 3D Green's function [14].

For a partially coherent source, we calculate the intensity distribution at the sensor by treating the source as a collection of different spatially-coherent sources and summing the intensity generated by each after coherent propagation:

$$I(\boldsymbol{r}) = \iint S(\boldsymbol{u}') \left|U(\boldsymbol{r}; \boldsymbol{u}')\right|^2 d^2\boldsymbol{u}', \tag{2.3}$$

where $S$ represents the 2D angular distribution of the incoherent source (assuming Kohler geometry) and $u'$ describes the spatial frequency of each spatially coherent source (*e.g.* each LED). The 3D spatial coordinates, $\boldsymbol{r}$, is dropped in future expressions for simplicity.

Previous work [32] simplified Eq. 2.3 by taking the first Born approximation [14] in order to obtain a linear model directly relating the intensity measurements to the scattering potential. The Born approximation assumes $U \approx U_{in}$ in the integral term of Eq. 2.2 and is valid for weakly-scattering objects where $U_{in} \gg U_{scat}$. The weak object approximation applies when the auto-correlation of $U_{scat}$ is negligible due to weak scattering, $|U_{scat}|^2 \approx 0$ [167]. If we separate the scattering potential into its real and imaginary parts, $\tilde{V} = \tilde{V}_{Re} + i \cdot \tilde{V}_{Im}$, the background subtracted 3D image stack under the $i$th illumination pattern, $I_i'$, can be written as

$$\tilde{I}_i' = H_{Re}^{(i)} \cdot \tilde{V}_{Re} + H_{Im}^{(i)} \cdot \tilde{V}_{Im}, \tag{2.4}$$

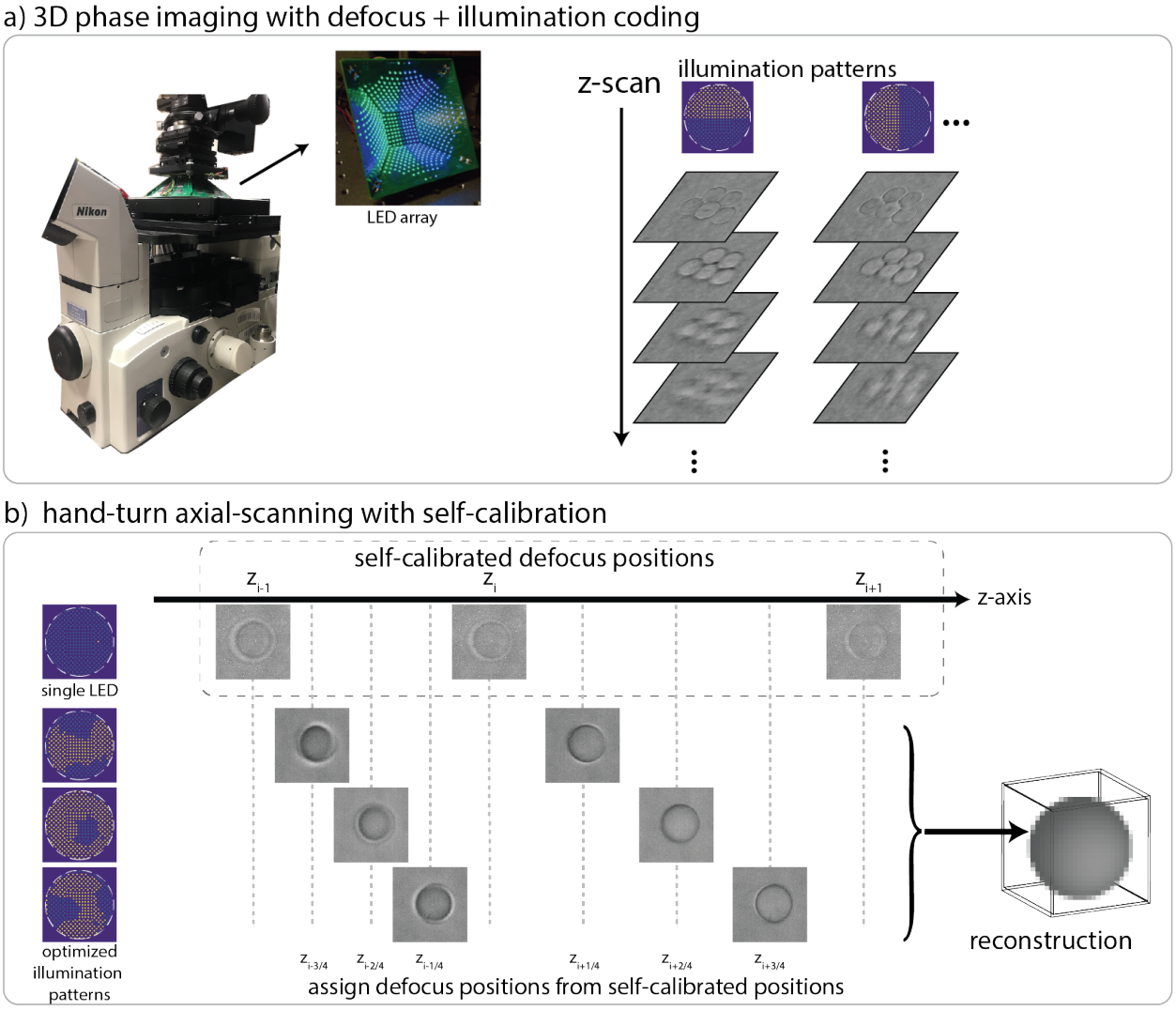


Figure 2.1: (a) Imaging setup on a commercial inverted microscope with a custom LED array illumination unit. The sample is imaged with various partially-coherent illumination patterns and at different focal planes; then a computational algorithm recovers the 3D refractive index map from the captured dataset. (b) We capture images continuously while cycling through different illumination patterns and scanning axially by hand-turning the focus knob. The single LED illumination pattern (spatially coherent) enables self-calibration of the defocus positions, such that the precise focus position need not be known.

where $\tilde{\cdot}$ denotes Fourier transform and $H_{Re}^{(i)}$ and $H_{Im}^{(i)}$ are the real and imaginary parts of the transfer functions corresponding to phase and absorption contrast, respectively, for the $i$th pattern $S_i$ [32]:

$$\begin{aligned} H_{Re}^{(i)} &= \mathcal{F}_z\, i[(S_i' \cdot P_z) \star (P_z \cdot \Gamma) - (P_z \cdot \Gamma) \star (S_i' \cdot P_z)], \\ H_{Im}^{(i)} &= \mathcal{F}_z[(S_i' \cdot P_z) \star (P_z \cdot \Gamma) + (P_z \cdot \Gamma) \star (S_i' \cdot P_z)], \end{aligned} \tag{2.5}$$

where $\mathcal{F}_z$ denotes Fourier transform along the $z$ axis, and $\star$ denotes cross-correlation. $S_i'$ is the flipped source distribution of $S_i$, $P_z$ is the pupil function with defocus kernel, and $\Gamma(\boldsymbol{u}) = \frac{1}{4\pi\sqrt{n_o^2\lambda^{-2} - |\boldsymbol{u}|^2}}$ for lateral spatial frequency $\boldsymbol{u}$.

Given defocus image stacks for each of $M$ different source patterns, the scattering potentials can be found by solving the following inverse problem:

$$\underset{\tilde{V}_{Re}, \tilde{V}_{Im}}{\arg\min} \sum_{i=1,\ldots,M} |\tilde{I}_i' - H_{Re}^{(i)} \cdot \tilde{V}_{Re} - H_{Im}^{(i)} \cdot \tilde{V}_{Im}|_2^2 + \eta \cdot R(V_{Re}, V_{Im})_2^2, \tag{2.6}$$

where $R(\cdot)$ is the regularization term. When Tikhonov regularization (L2 norm of $V_{Re}$ and $V_{Im}$) is used, we can find an analytical estimator for the scattering potential:

$$\begin{pmatrix} V_{Re}^* \\ V_{Im}^* \end{pmatrix} = \begin{pmatrix} \sum |H_{Re}^{(i)}|^2 + \eta & \sum H_{Im}^{(i)\,H} H_{Re}^{(i)} \\ \sum H_{Re}^{(i)\,H} H_{Im}^{(i)} & \sum |H_{Im}^{(i)}|^2 + \eta \end{pmatrix}^{-1} \begin{pmatrix} \sum H_{Re}^{(i)\,H} \tilde{I}_i' \\ \sum H_{Im}^{(i)\,H} \tilde{I}_i' \end{pmatrix}. \tag{2.7}$$

Thus, the 3D absorption and refractive index distributions can be recovered from the raw data.

## 2.2 Axial scanning and defocus self-calibration

In previous work, axial through-focus scanning was performed by a motion $z$-stage with closed-loop control. With the "stop-and-stare" strategy, the user moves the stage to each desired focal plane and acquires images with known defocus positions. Here, we instead hand-turn the built-in focus knob on a standard microscope while continuously updating the illumination patterns and capturing images at a fast enough frame rate such that there is negligible motion blur in each frame. This enables fast axial scanning without hardware dependency; however, the user can no longer specify the focal planes and the defocus position of each image is unknown.

We seek an algorithmic way to infer the unknown defocus positions in post-processing, known as *self-calibration*. With partially coherent illumination (many LEDs turned on at once), the system will have strong optical sectioning [156], meaning that images taken at a particular focal plane will have little information from other focal planes. Thus, the problem of jointly solving for the defocus positions and the 3D sample becomes very ill-posed. If spatially coherent illumination (a single LED) is used instead, there is no optical sectioning

and changing focus of the microscope only changes the defocus kernel. This makes 3D reconstruction difficult but gives good information to solve for defocus positions. Hence, we choose to use both partially coherent illumination patterns designed for DPC and a self-calibration pattern with only a single LED on. While the focus knob is swept by hand, the LED array quickly alternates between these patterns, with each lasting for one exposure.

Our capture strategy thus collects images for the single-LED illumination at different defocus positions, which are used for self-calibration. The LED array is placed sufficiently far from the sample such that an LED illuminates the sample with a plane wave (spatially coherent light), defined as $\exp\left(i2\pi \boldsymbol{u}' \cdot \boldsymbol{x}\right)$, where $\boldsymbol{u}'$ corresponds to the plane wave angle. Because the light is spatially coherent, we can model the single-LED self-calibration images as

$$I_{calib}\left(\boldsymbol{x}, z\right) = |\mathcal{F}^{-1}[P_z\left(\boldsymbol{u} - \boldsymbol{u}'\right) \cdot \tilde{o}\left(\boldsymbol{u}\right)]|^2, \tag{2.8}$$

where $o$ is the 2D complex-field, and $\boldsymbol{u}$ denotes the lateral spatial frequency. $P_z$ is the pupil function with a defocus distance $z$, modeled by angular spectrum propagation [14], $P_z\left(\boldsymbol{u}\right) = P\left(\boldsymbol{u}\right)\exp\left[i2\pi z\sqrt{\frac{1}{\lambda^2} - \boldsymbol{u}^2}\right]$. This single-LED illumination can be from any angle; we choose an off-axis LED because the image will shift laterally with defocus, providing stronger defocus contrast.

## Joint optimization for self-calibration

Once the dataset (with $N$ images) is captured, we then need to jointly solve for the field and the defocus positions. The problem can be written in a joint optimization form,

$$\arg\min_{o,z} \sum_{i=1}^{N} \|I_{calib}\left(\boldsymbol{x}, z_i\right) - |\mathcal{F}^{-1}[P_{z_i}\left(\boldsymbol{u} - \boldsymbol{u}'\right) \cdot \tilde{o}\left(\boldsymbol{u}\right)]|^2\|_2^2. \tag{2.9}$$

This formulation takes $N$ intensity images, $I_{calib}\left(z_i\right)$, to solve for one 2D complex-field, $o$, and defocus positions, $z_i$ for $i = 1, ..., N$, and thus is well-constrained even with only a few defocus planes [2]. The optimization problem is, however, non-convex, and the defocus positions $z_i$ and complex-field $o$ are dependent on each other. As a result, when we use gradient descent-based methods to optimize them, their gradients will be affected by one another, making it difficult to reach convergence. However, if the defocus positions are known, the field can be solved for; similarly, if the field is known, the defocus position can also be determined for each image. For a non-divergence solution, we alternate the optimization for these two unknowns, such that only one variable is updated at each time [213, 31].

To perform the optimization, we first initialize the defocus positions with a guess of the total range of defocus and equal spacing between images. Then, we use gradient descent, implemented with Adam [90] for fast convergence, to optimize the complex-field in Eq. 2.9 with $z$ fixed. Next, we fix the complex-field at the current estimate and update the defocus positions, $z$. We check the loss after each iteration and stop the update if it converges earlier. These alternating updates continue until the convergence of both variables. To

ensure a unique solution of defocus positions, a non-decreasing condition is assumed, such that the defocus motion is only in one direction. This condition is enforced by projecting the updated defocus positions into a non-decreasing sequence.

After joint optimization, we know the defocus position for each single-LED image. Then the defocus positions for images using other illumination patterns are bi-linearly interpolated by the nearest known defocus positions from single-LED images. We also note that, since image planes are no longer equally spaced, during the 3D reconstruction as in Eq. 2.4, a non-uniform discrete Fourier transform needs to be computed in the $z$ direction.

## 2.3 Experiment setup for DPC

Experiments were performed on a commercial inverted microscope (Nikon TE2000-U) with a 40× 0.65NA objective lens (Nikon). A customized LED quasi-dome array (SCI Microscopy) [139] was installed on the microscope to replace the conventional illumination unit. The top panel of the LED array was positioned 65mm above the focal plane and only the 'brightfield' LEDs were used (those that illuminate the sample from an angle within the NA of the objective lens). Green LEDs ($\lambda$ centered at 525nm) were used throughout the experiment. We used a sCMOS sensor (PCO Edge 5.5 monochromatic) to capture intensity images in global shutter mode. Each exposure was hardware triggered by the LED array's controller (Tenseey 3.2) after every illumination pattern update. We used an automated piezoelectric $z$-stage (Thorlabs MZS500-E) to defocus the sample for the comparison case of controlled axial scanning. For hand-turning defocus, the fine focus knob of the microscope was spun, and an off-axis LED at $NA_x = 0.36, NA_y = 0.0$ was used as the self-calibration single-LED pattern.

## 2.4 Validation of defocus self-calibration

We validate the self-calibration algorithm on experimental data from single-LED illumination. We use the precision $z$-stage to defocus and acquire an evenly-spaced image stack with 140 planes with a step size of 1µm. This image stack is acquired with the same optics and exposure setting as in the rest of study. Then, we randomly choose a number of images with defocus positions in a monotonic order, and those chosen images become an unevenly-spaced image stack with known defocus positions, which can be used to validate our self-calibration algorithm. We repeat this process to randomly generate 20 image stacks for each of 10 different average defocus spacing between images, from 1µm to 10µm. The self-calibration algorithm, blind to the knowledge of their ground truth positions, is performed on each image stack as follows. The self-calibration is initialized with a linear defocus estimation. Then, the joint updates are performed for 100 iterations, each of which contains 50 gradient descent steps to update the field and 10 steps to update the defocus positions.

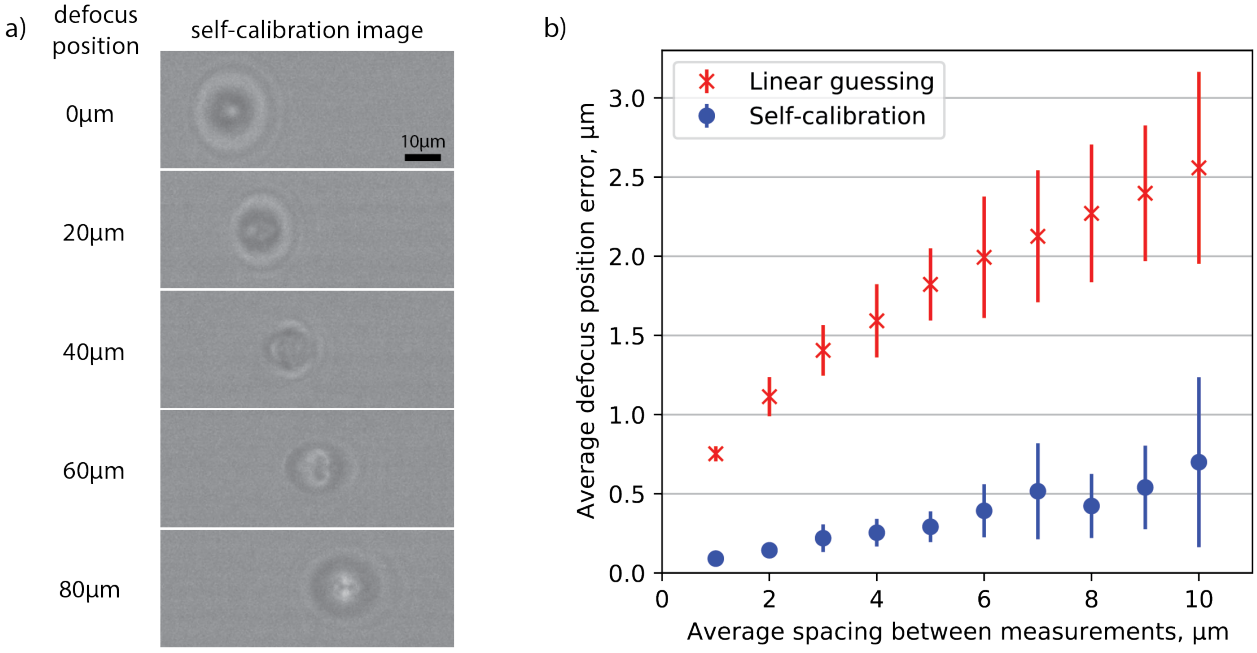


Figure 2.2: Algorithmic self-calibration for defocus positions. (a) Self-calibration images from a single off-axis LED at different depth planes for a simple object (a single polystyrene bead). (b) Experimental average defocus position errors before (linear guessing) and after self-calibration for different average defocus spacing between measurements. Each data point shows the average over 20 unevenly-spaced image stacks, and the error bars denote 2× the standard deviation.

The error of the self-calibration is quantified by comparing with the ground truth defocus positions and results are plotted in Fig. 2.2. With average defocus spacing of 1µm, the defocus position error was 91nm, which is close to the resolution of the $z$-stage (50nm). When the average defocus spacing is 5µm, the defocus position error is 0.29µm while the error of linear guessing jumps to 1.82µm.

# Chapter 3

# 3D DPC illumination pattern optimization

We further improve the practicality of 3D DPC by optimizing the illumination source patterns. Previous work [114, 32] used four half-circular illumination patterns at each focal plane, though later works have shown the ability to reduce the number of images captured [138, 162]. The half-circle designs were developed heuristically for use with analytical inversion methods. However, since arbitrary patterns can be used in our system, we aim to optimize for illumination patterns that best encode the 3D phase information in the raw images. To enable a systematic design of the LED patterns, *Hugonnet et al.* [80] defined an objective function to evaluate illumination patterns. However, the optimized design often becomes very sensitive to the choice of objective function (e.g., to balance high vs. low frequency, sensitivity vs. signal-to-noise ratio (SNR)). Recently, a class of data-driven methods called physics-based learning have been used to optimize the illumination design end-to-end to improve the final reconstruction without a crafted objective function [85, 83]. Here, we employ physics-based learning to optimize the illumination patterns for our 3D DPC setup, ensuring efficient and robust capture strategies. Combining this with our self-calibrated axial motion, we demonstrate practical high-quality refractive index reconstruction on a commercial microscope with LED array illumination.

## 3.1 Physics-based learning to optimize illumination pattern

The choice of illumination patterns will largely dictate the quality of the results. As described above, we time-interleave a single-LED illumination with multiple DPC patterns. Instead of using the traditional half-circle DPC patterns, we use new techniques in physics-based learning [85] to design better partially coherent DPC illumination patterns. The forward

This chapter includes the research I presented or published in [24, 28].

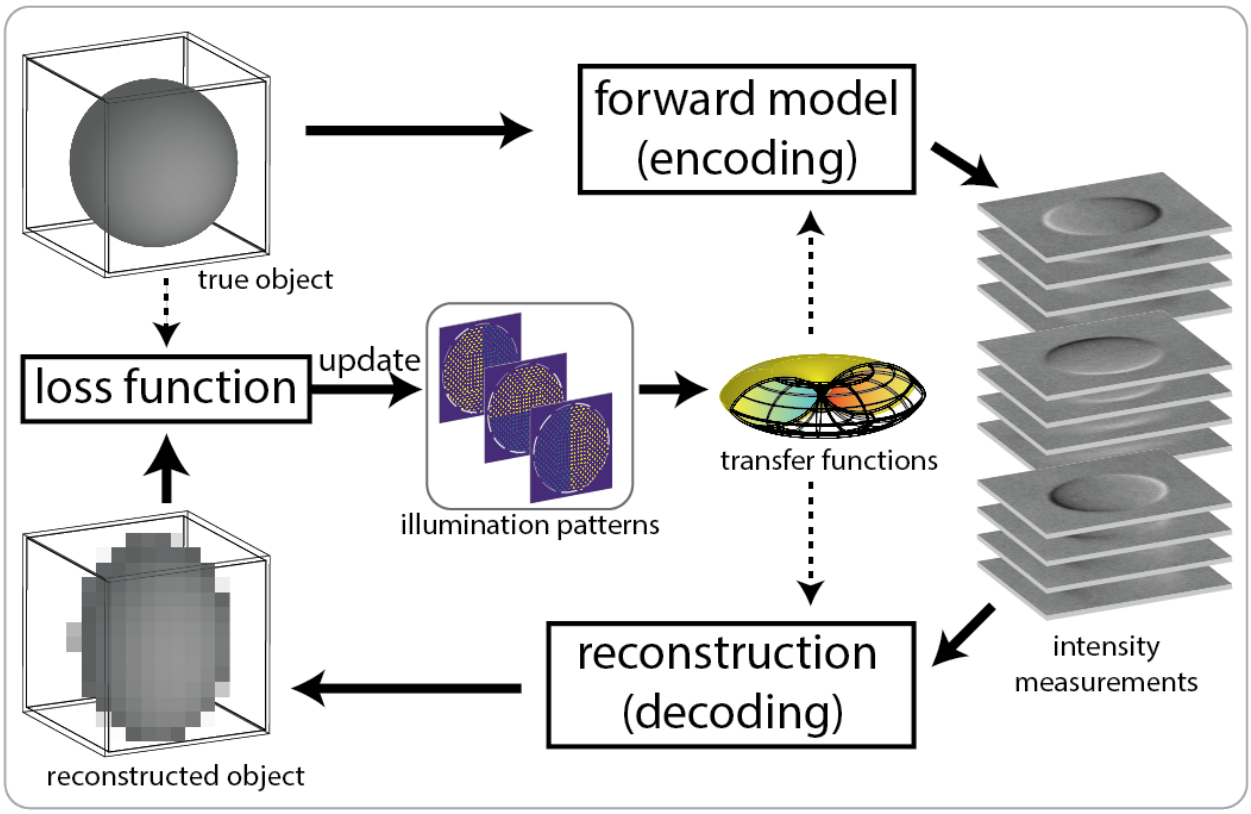


Figure 3.1: Illumination patterns were designed by physics-based learning to optimize encoding of phase information into intensity measurements.

model of image formation for 3D DPC 'encodes' the sample's scattering potential into 2D intensity measurements, and the reconstruction 'decodes' the 3D information from these measurements. As described in Section 2.1, the encoding and decoding processes are described by the system's transfer functions, which specify what information can be encoded into the measurements as well as how much of the encoded information can be recovered (without being overwhelmed by noise, etc.).

Since both the forward model and the reconstruction are differentiable in simulation, physics-based learning can optimize the patterns by forming the encoding-decoding pipeline, as in Fig. 3.1, then defining a loss function to measure the discrepancy between the the true scattering potential of a simulated sample and its reconstruction. The simulated samples, which the optimized illumination patterns will be tailored to, are expected to have a spatial frequency distribution similar to the experimental samples. The illumination patterns are updated iteratively to minimize the loss function. Our loss function consists of an object consistency loss and a source physical constraint term:

$$Loss\left(S\right) = \left\|V - rec\{fwd\left[H\left(S\right), V\right], H\left(S\right)\}\right\|_2^2 + \mu \cdot c\left(S\right), \tag{3.1}$$

where $S$ denotes a set of illumination patterns. The real and imaginary components of the transfer function and scattering potential are written together for conciseness. The object consistency loss measures the L2-distance between the reconstruction and the true scattering potential, $V$, where *fwd* and *rec* are the forward model and reconstruction as described in Eq. 2.4 and Eq. 2.6, respectively.

The source physical constraint term, $c$, enforces non-negative light intensity and limits the maximum intensity of each LED to one:

$$c\left(S\right)=\begin{cases}-S, & S<0\\ 0, & 0\leq S\leq 1\\ S-1, & S>1.\end{cases} \tag{3.2}$$

This term will give a reverse gradient when $S$ goes below 0 or above 1. The overall weight of this term is set to be large, so that its gradient also prevails over the gradient of object consistency when the light intensity goes beyond the range. We find this term effective to eliminate trivial solutions with very large or negative values for light sources, and the optimized patterns do not require a normalization or clipping at the end.

## 3.2 Practical considerations

We discuss a few practical considerations that we include in the optimization. First, it is important to simulate the noise in the forward model to discourage solutions that bring good phase contrast but sacrifice the overall SNR. Hence, we model the readout noise and the signal noise in the forward model. The readout noise is from a Gaussian distribution by its nature, and the signal noise is from a Poisson distribution, which is also approximated as Gaussian for the relatively high light levels in our system. During physics-based learning, both the readout and the signal noise are sampled from standard normal distributions and scaled with the total illumination intensity, $\sum S$, before being added to the simulated intensity images at the end of the forward model. The total noise, $I'_{noise}$, added to the normalized intensity images can be written as

$$I'_{noise}=\frac{\alpha\cdot N_{\text{readout}}}{t_{\text{exp}}\sum S}+\frac{\beta\cdot N_{\text{signal}}}{\sqrt{t_{\text{exp}}\sum S}},\ \text{where}\ N_{\text{readout}},N_{\text{signal}}\sim\mathcal{N}\left(0,1\right), \tag{3.3}$$

where $t_{\text{exp}}$ is the exposure time, and $\alpha$, $\beta$ are coefficients we obtained from experimentally-captured images. Since the additive noise carries no information about the sample, it will only negatively affect the reconstruction and raise the loss value. In this way, the physics-based learning optimization will have an incentive to use higher total illumination power.

Second, binary-valued illumination patterns (i.e., each LED is either on or off) are easier to implement in practice because: 1) the hardware delay time of binary-valued pattern updates are shorter, and 2) binary-valued patterns do not require per-LED illumination

intensity calibration. However, this binary-value constraint requires a combinatorial optimization, which is difficult to solve in practice. Instead, we still use gradient descent to optimize for continuous-valued patterns while promoting a binarized LED intensity value distribution (values to be close to either 0 (off) or 1 (on)) as follows. At each iteration, we feed in the binarized patterns to the forward model while keeping the continuous-valued patterns for the reconstruction. Since only the continuous-valued patterns are updated during the gradient descent optimization, the final optimized patterns will have more intensity values close to 0 or 1 to minimize the mismatch due to the binarized patterns in forward model. After the optimization, the binary-valued patterns can be obtained by thresholding the continuous-valued optimized patterns.

Third, we take into account the slight misalignment of the LED array to improve the robustness of the optimized patterns. We randomly add a small lateral shift to the source patterns during the forward model while assuming the original, not-shifted patterns in the reconstruction. This mismatch will deteriorate the reconstruction for illumination patterns sensitive to misalignment, and we find that the optimized patterns with this consideration are denser and more connected.

## 3.3 Optimization setup

To determine the optimal number of DPC illumination patterns, we compare the final object consistency losses for the optimizations with different numbers of patterns. When an additional pattern does not further reduce the loss, it is considered unnecessary. Fig. 3.2 shows the relationship between the object consistency loss and the number of DPC patterns. When noise is incorporated into the simulation (Fig. 3.2(b)), having more patterns in a fixed total acquisition time reduces the SNR. We find 2-4 DPC patterns gives the minimal loss value, and thus we choose to have four illumination patterns, as in the case of half-circular DPC illumination patterns.

Next, we use end-to-end optimization to find the patterns for our four DPC illuminations. The patterns are randomly initialized from a uniform distribution and then optimized by an Adam optimizer [90] for 250 iterations. The optimization is implemented in Tensorflow (Google), and we use a single GPU (NVIDIA Titan X) to accelerate the computing. The optimized patterns are shown in the first two rows of Fig. 3.3(c). The typical DPC half-circle patterns (Fig. 3.3(a)) and the optimized patterns without applying the practical considerations in Sec. 3.2 (Fig. 3.3(b)) are shown for comparison. The patterns without the practical considerations (Fig. 3.3(b)) have more emphasis on high-angle illumination; the patterns optimized with the practical considerations (Fig. 3.3(c)) have dense and connected patterns to cover more low-angle LEDs, which presumably give a better SNR and are more robust for misalignment.

Throughout the pattern optimization, we use ground truth simulated objects that are somewhat similar to the types of samples we expect in experimental applications. We simulate spherical objects with smaller high-RI spheres inside to mimic simple cells. A small

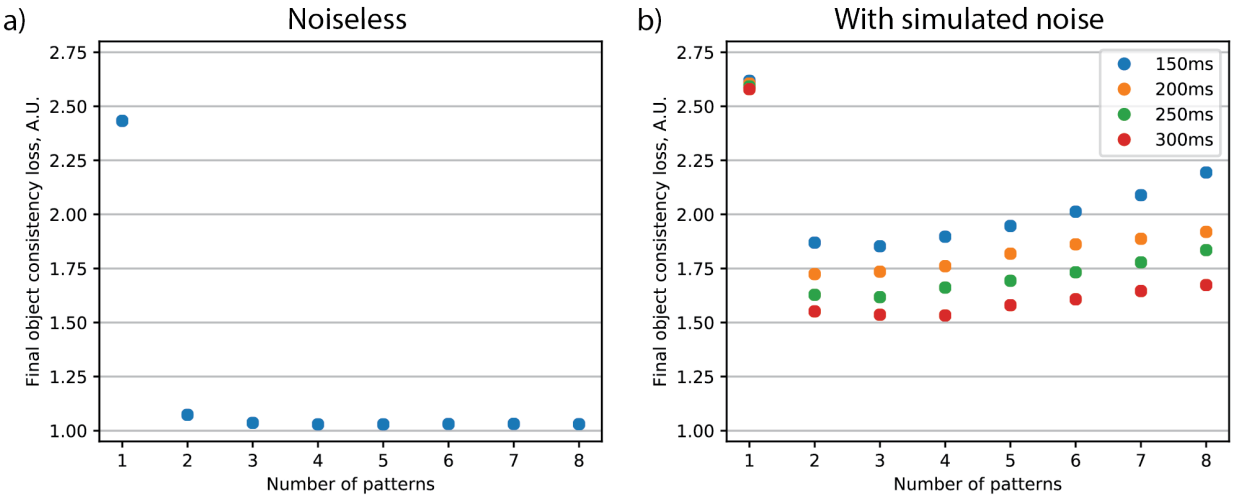


Figure 3.2: The final object consistency loss of physics-based learning as the number of illumination patterns varies, (a) without noise, and (b) with simulated noise. Marker colors indicate the acquisition time for each set of illumination patterns.

Gaussian noise with mean 0 and standard deviation 0.25 (which is about 3% of the real part of the maximum scattering potential value) is also added to each object's scattering potential to increase the spatial frequency diversity of simulated objects and to avoid over-fitting to particular spatial frequencies.

## 3.4 Experimental validation and transfer function analysis

The optimized DPC patterns were programmed on the experimental system for validation. We imaged 10µm polystyrene-based microsphere beads (Sigma-Aldrich) with RI 1.6 [168] in RI 1.584 index-matching oil (Cargille; RI 1.592 at $\lambda = 525nm$). Some microspheres were greater than 10µm in diameter, possibly due to their reaction to the index-matching oil or degradation during storage. We acquired an image stack for each illumination pattern, defocused by the precision $z$-stage with 1µm spacing between planes.

As a comparison, the RI reconstructions for the half-circular DPC patterns used in [32], patterns optimized without practical considerations, and patterns optimized with practical considerations are shown in the third row of Fig. 3.3 respectively. The patterns optimized with practical considerations (Fig. 3.3(c)), give more accurate RI values and reduce elongation artifacts [32] in the axial direction compared with the half-circular baseline (Fig. 3.3(a) and patterns optimized without the practical considerations (Fig. 3.3(b)). Therefore, the practical considerations helps to improve experimental robustness and the patterns opti-

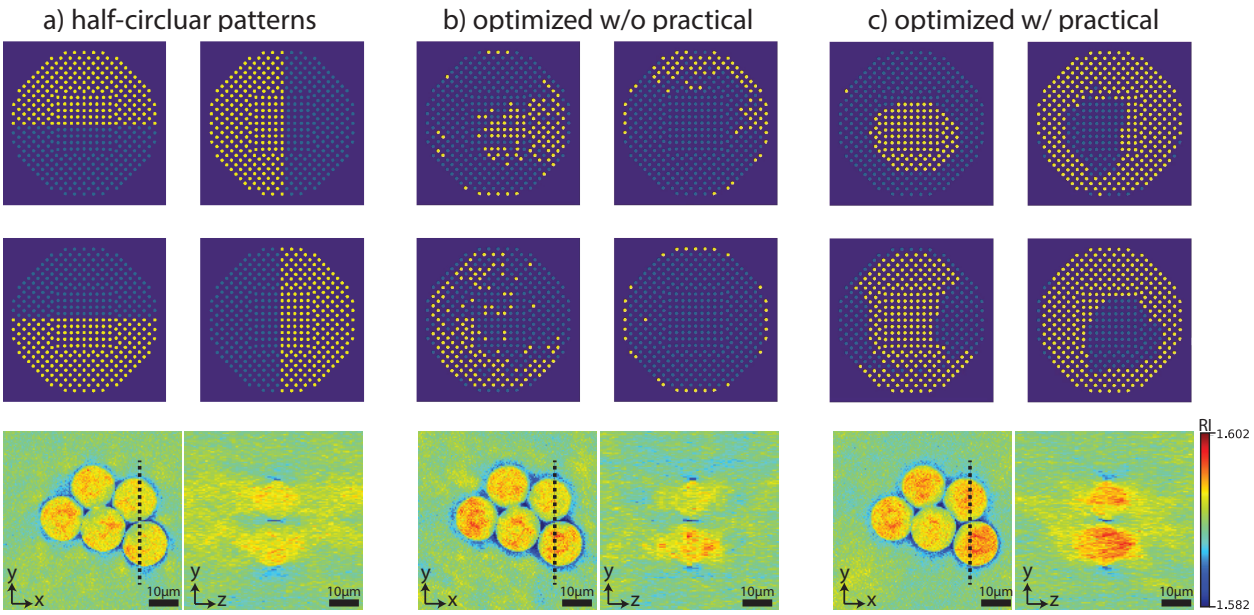


Figure 3.3: Experimental 3D refractive index volume reconstruction with different illumination pattern designs, for a polystyrene bead sample. (a) Half-circular differential phase contrast (DPC) patterns, (b) optimized patterns without practical considerations in Section 3.2, and (c) optimized patterns with practical considerations, which gives the best reconstructions.

mized with the practical considerations (Fig. 3.3(c)) are used as our final optimized DPC patterns. A 3D reconstruction of human embryonic stem cells using the final optimized DPC patterns is shown in Fig. 3.4.

To further investigate the optimized DPC patterns, we visualize the 3D phase transfer functions ($H_{Re}$ in Eq. 2.4) corresponding to the patterns (Fig. 3.5). Note that zeros in the transfer function indicate that no phase information is encoded in the intensity measurements for that spatial frequency. Therefore, a good set of illumination patterns should have as many non-zero regions as possible to recover 3D information. Dotted circles in 2D slices indicate the region of missing cones (also illustrated in 3D in Fig. 3.5(c)), in which the phase information cannot be encoded due to the limited NA [166, 167]. Fig. 3.5(b) shows the 3D transfer function for a traditional DPC half-circular pattern, for comparison. The transfer function of DPC pattern has good coverage at the zero axial frequency plane, but misses the low-frequency content (indicated by the arrows in Fig. 3.5(b)) outside of the missing cones at non-zero axial frequencies. This explains the RI underestimation of the DPC patterns in Fig. 3.3(a); the transfer functions for the optimized patterns have good coverage of low-frequency content across different axial frequencies and thus recover a more accurate quantitative RI value.

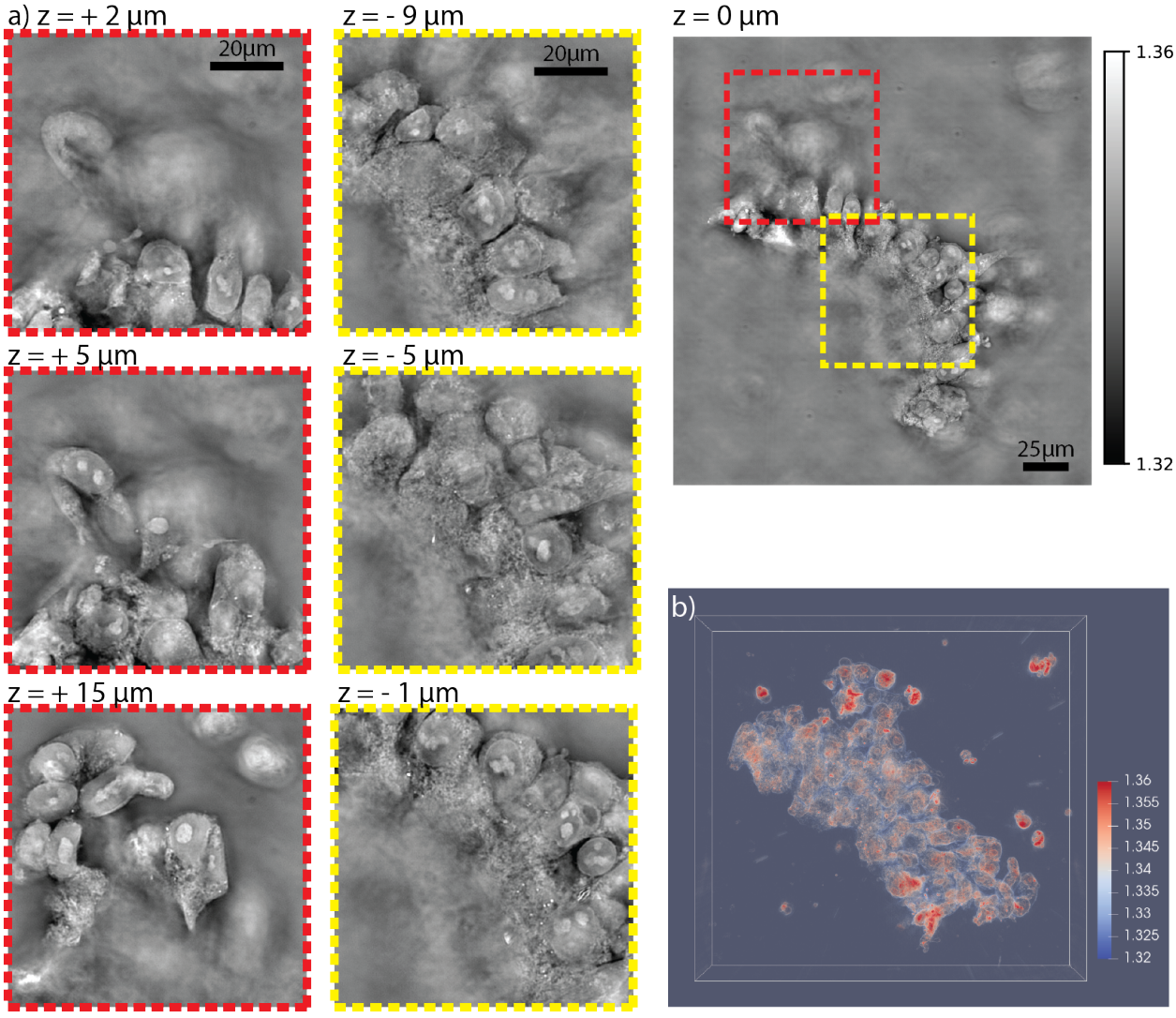


Figure 3.4: Experimental 3D refractive index volume reconstruction of human embryonic stem cells (hESC) using optimized illumination patterns. (a) A lateral slice of the refractive index reconstruction with two zoomed-in regions at different $z$ planes. (b) 3D rendering.

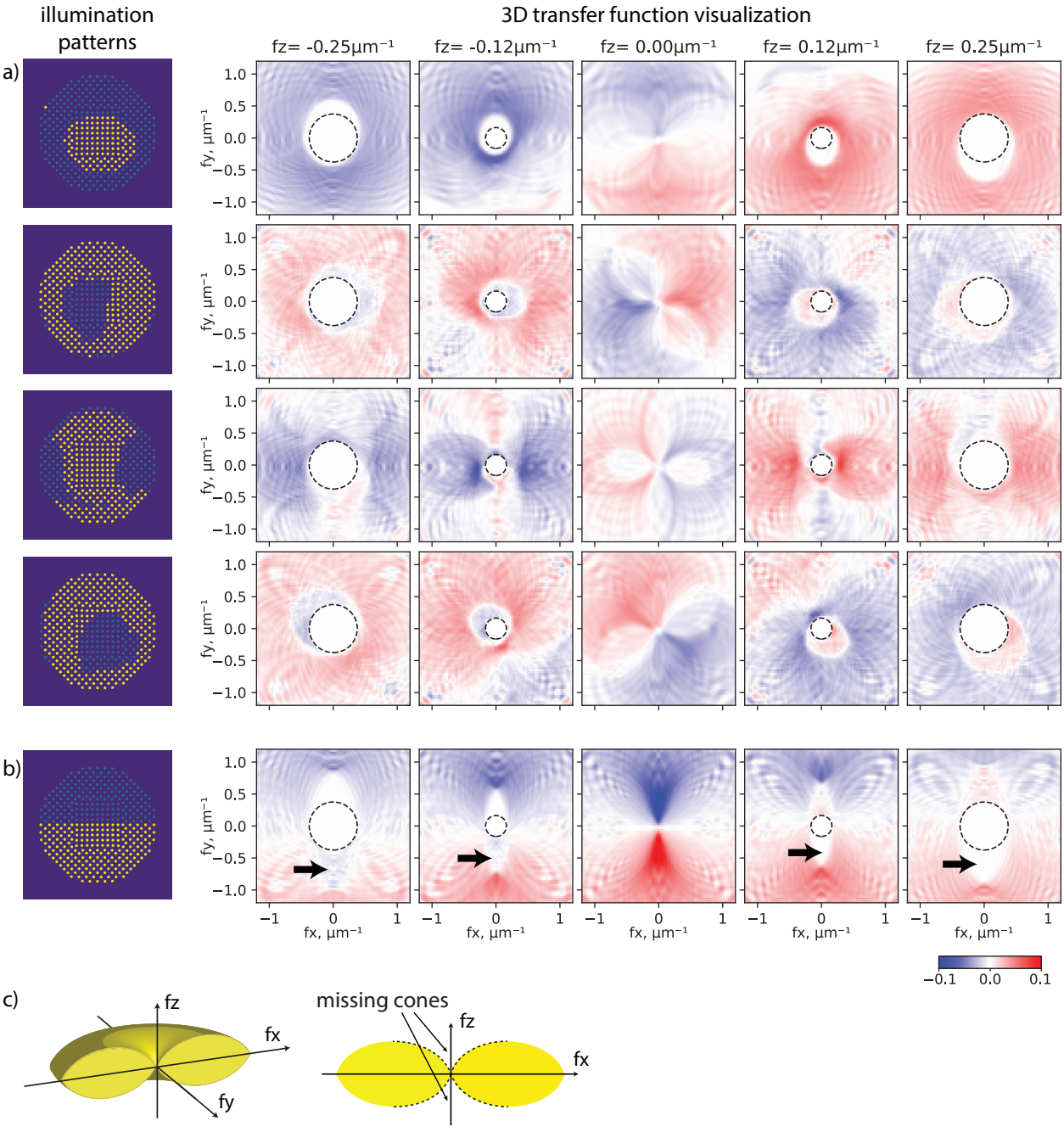


Figure 3.5: Comparison of the 3D phase transfer functions for (a) the 4 optimized illumination patterns from our physics-based learning, and (b) a half-circular DPC pattern. Dotted circles in (a) and (b) indicate the phase transfer function's missing cones, which can also be seen in (c) the 3D visualization of the theoretically feasible support regions of the transfer function (due to the limited NA).

## 3.5 Experimental validation with self-calibration + optimized patterns

We combined hand-turning axial scanning (Chapter 2) with the optimized illumination patterns into our final 3D DPC system. We imaged 10µm borosilicate glass (RI 1.56) microspheres (Duke Standards, Thermo Fisher) in RI 1.54 index-matching oil (Cargille; RI 1.546 at $\lambda = 525nm$). The illumination sequence consisted of the 4 optimized patterns in Fig. 3.3(c) and one single-LED illumination for defocus self-calibration. During the acquisition, we continuously turned the fine focus knob by hand at an approximately steady speed, while the LED array looped through the illumination patterns and sent triggers to the camera after each update. We acquired 375 frames (75 frames for each illumination) in about 25 seconds, and the total defocus was about one rotation of the fine focus knob, roughly 100µm of defocus. With single-LED illumination measurements, we performed defocus self-calibration and recovered the defocus positions shown in Fig. 3.6(a). The self-calibration optimization took about 7 minutes to finish 100 iterations using a single GPU (Nvidia Titan X). The self-calibration update reached its convergence after around 50 iterations (see Fig. 3.6(b)). The reconstructed 3D RI volume is visualized in Fig. 3.6(c), with zoom-in lateral and axial sections for the two insets. Many detailed features within microspheres (presumably due to the fabrication imperfection) can be observed with clear contrast in both insets, showing the efficacy of the optimized patterns. The quantitative RI values also match well to the labeled value of the glass microspheres (RI 1.56), with the exception of a negative relative RI and a zero relative RI microsphere in inset 2 of Fig. 3.6(c). We believe these two microspheres have different RI inherently after checking the measured image contrast. The halo artifact for phase imaging described in [195, 126] can also be observed for our 3D reconstruction on the in-focus plane of an object; a non-negativity constraint can be used during the Tikhonov reconstruction to suppress the halo artifact as described in [32].

## 3.6 Conclusion

We demonstrated an extension of 3D differential phase contrast (DPC) imaging with improved reconstruction and without a motion $z$-stage. In Chapter 2, we introduced a practical stage-free axial scanning by spinning the built-in focus knob while taking measurements and then self-calibrating for the actual defocus positions later using a joint update algorithm. In this chapter, we showed the illumination patterns can be optimized for a better refractive index reconstruction.

Future work could focus on tailoring the loss function for the optimization based on the application. For example, low-frequency contrast will be more important if we are interested in segmenting densely placed cells. It is also worthwhile to investigate the role of different regularization of the reconstruction. We used Tikhonov regularizer in this paper for its generality, while other regularizers, such as total variation (TV) and deep learning-related ones [62, 200], might bring other insights in particular imaging scenarios.

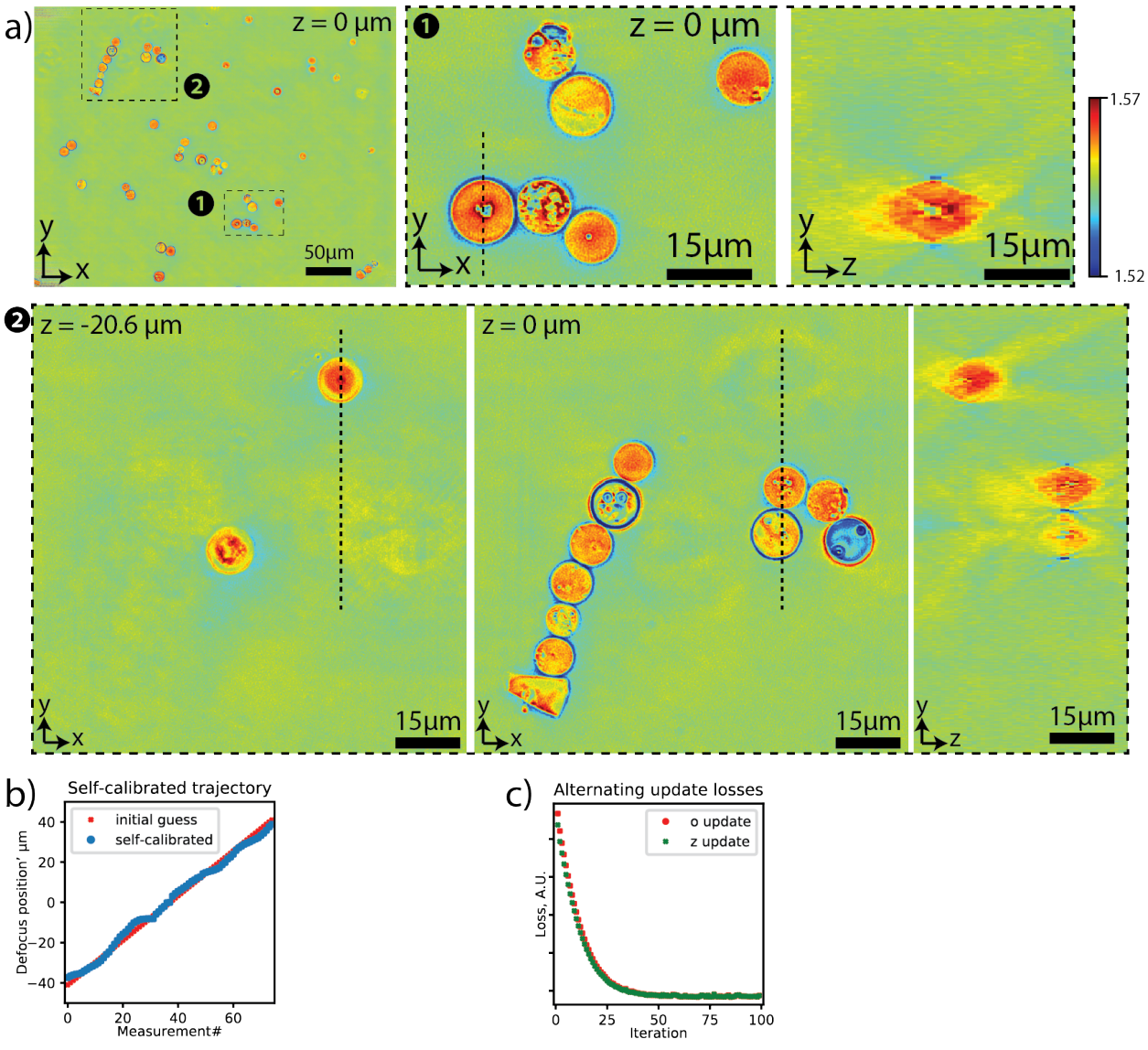


Figure 3.6: Experimental 3D refractive index reconstructions of borosilicate glass beads from a hand-tuned defocus stack. (a) Reconstructed refractive index at one depth slice, with two zoom-ins showing lateral and axial cross-sections and different depth slices. (b) Recovered defocus positions after self-calibration. (c) Self-calibration loss (defined in Eq. 2.9) for each iteration of the joint optimization, which converges after 80 iterations.

# Chapter 4

# Speckle Flow SIM: dynamic speckle structured illumination microscopy

Structured illumination microscopy (SIM) [67] is a practical super-resolution method that uses patterned illumination to encode high-frequency information from Moiré patterns, and then computationally decodes the image, enabling 2× better resolution than the diffraction limit. Compared with other super-resolution methods (e.g., STED [71], PALM [12]), SIM has faster frame rates, lower photo-toxicity [60], and is compatible with both brightfield [34] and generic fluorescence methods [67]. Although SIM usually uses sinusoidal illumination, it can also be implemented with random unknown speckle illumination, called speckle SIM [120]. The speckle is generated by a plane wave passing through a scattering layer and thus is easy to set up in experiment and does not require a calibration step for the structured pattern [194, 94].

In both sinusoidal and speckle SIM, multiple images must be captured to achieve super-resolution; hence, SIM trades off the system's temporal resolution for spatial super-resolution. Sinusoidal SIM methods usually take ∼10 diffraction-limited raw measurements, each with a different shift or rotation of the sinusoidal illumination pattern, to reconstruct a 2× super-resolved image [67]. Speckle SIM methods often require dozens of randomized speckle illuminations [120] or shifted speckle patterns [201, 203]. In both cases, the multi-shot nature of SIM effectively reduces the frame rate by at least ∼10-fold. If the scene contains motion during the multi-shot acquisition, the recovered super-resolved image will suffer from motion artifacts [54]. Here, we explore methods for modeling the spatio-temporal relationship of the dynamic scene in order to reconstruct images without motion artifacts.

A dynamic, super-resolved scene can be modeled using neural representations, which are a class of methods that encode high-dimensional information into an untrained deep neural network and store the information compressively in the neural network's weights [107]. A coordinate-based neural network is a type of neural representation which maps a matrix coordinate to its corresponding value on the matrix, usually via a multi-layer perceptron

This chapter covers the research I presented or published in [25, 22, 29].

(MLP) [164]. Coordinate-based MLPs were first demonstrated to model for 3D scenes [115] or 3D geometries [159, 134]. The spatio-temporal relationship can be similarly represented using coordinate-based MLPs by adding a time coordinate [135, 144].

In this chapter, we develop a new SIM method, called Speckle Flow SIM, that spatially super-resolves a dynamic scene using a neural space-time model. In Speckle Flow SIM, we modulate the scene with speckle illumination to encode high-frequency information into diffraction-limited images. Unlike previous speckle SIM systems that change the speckle to acquire sufficiently diversified information for a well-posed reconstruction, we maintain the speckle illumination unchanged but rely on the scene dynamics for the measurement diversity (see Fig. 4.1(a)). We model the spatio-temporal relationship of the dynamic scene using a neural space-time model with coordinate-based MLPs [164] and jointly recover the motion dynamics and the super-resolved scene. This allows Speckle Flow SIM to spatially super-resolve a dynamic scene with deformation motion. We validated Speckle Flow SIM for coherent imaging in simulation and experimentally demonstrated it using a simple, inexpensive setup.

## 4.1 Related Work

Sinusoidal SIM is widely used to achieve up to two times better than diffraction-limited resolution [67]. Saturated SIM exploits the nonlinear response of saturated fluorescence and has unlimited theoretical resolution in the noise-free case and typically 5× super-resolution in experiments [66]. SIM is compatible with both coherent [34] and fluorescence systems [67], but saturated SIM only works for fluorescence systems. Speckle illumination is an alternative implementation of sinusoidal structured illumination, which also modulates high-frequency information into the diffraction-limited system [120, 110, 202, 201, 203]. Unlike sinusoidal illumination, speckle illumination is random and requires either additional prior statistical information [120, 202] or joint speckle calibration [201, 203] for a super-resolved image reconstruction. Speckle SIM has been experimentally demonstrated with a 1.6× resolution gain in a high-NA system [120] and a 5× resolution gain in a low-NA system [201].

SIM collects information beyond the diffraction limit by acquiring multiple raw images with varying illumination, which reduces the temporal resolution by an order-of-magnitude or more (depending on the number of raw images required for each super-resolved reconstruction). Previous study achieved video rate (10-20 Hz) SIM using a ferroelectric liquid crystal spatial light modulator (SLM) for a rapid update of structured light and a electron-multiplying charge coupled device (EMCCD) for a high-frame rate acquisition [91, 185]. *Mangeat et al.* similarly demonstrated a high-frame rate version of speckle SIM using a SLM and a scientific CMOS camera [110]. While these methods improved the overall frame rate of the system, the scene is still assumed to be static for each of the raw images taken at different timepoints. To improve the SIM reconstruction by accounting for the motion of the scene, *Shroff et al.* estimated the phase shift of the sinusoidal illumination to correct for a small global translational motion of the sample [157]. *Turcotte et al.* achieved dynamic

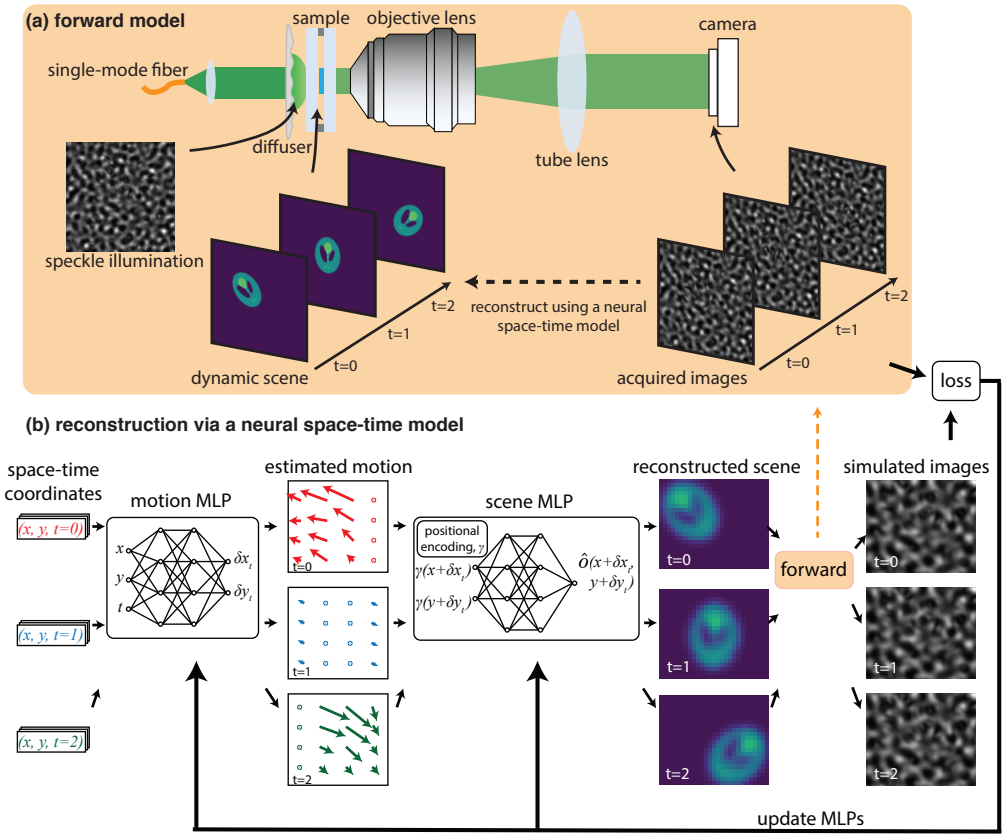


Figure 4.1: Overview of Speckle Flow Structured Illumination Microscopy (SIM). (a) A plane wave passes through a thin scattering layer to generate speckle-structured illumination at the sample. The microscope, a $4f$ system with an objective lens and a tube lens, then magnifies the image and the intensity is captured at the image plane by a CMOS sensor. The speckle illumination pattern is calibrated in advance and remains fixed while an image sequence of the dynamic scene is captured. The dynamic scene is modeled and simultaneously recovered with resolution beyond the diffraction limit. (b) The neural space-time model represents a dynamic scene using a motion multi-layer perceptron (MLP) and a scene MLP. The motion MLP takes a space-time coordinate, $(\boldsymbol{r}, t) = (x, y, t)$, corresponding to a pixel measured at a particular timepoint and estimates its displacement at $t$ relative to the time-independent scene stored in the scene MLP, $\delta\boldsymbol{r}_t = (\delta x_t, \delta y_t)$. The motion-accounted spatial coordinate, $\boldsymbol{r} + \delta\boldsymbol{r}_t$, is then fed into the scene MLP to query the corresponding value for the coordinate. This process is repeated for each coordinate to build up the entire scene replicated by the neural space-time model. During the reconstruction, the weights of the two MLPs are updated to minimize the difference (loss) between the acquired images and simulated images from the forward model.

in-vivo SIM imaging in the mouse brain by setting up a short exposure time and registering raw images to digitally remove motion artifacts [181]. However, these studies assume a simple motion (e.g., rigid global motion) and do not aim to retain the motion dynamics as a final product of the reconstruction. In this chapter, we improve the temporal resolution by embedding the spatio-temporal relationship into the forward model to account for and recover the dynamics for each raw image, including non rigid-body motion.

Space-time model is a lasting theme in the computational imaging community. This modeling is especially useful when a multi-shot system is used to image a dynamic scene, such that the raw images acquired at different timepoints capture the scene at different states of the dynamics. A common strategy is to meter the motion and co-register acquired images for a static image reconstruction unaffected by the motion. For example, to improve the noise performance in low-light, smartphone cameras take a burst of frames and align them to compensate for motion during the image signal processing pipeline [70]. Motion differential phase contrast (DPC) microscopy used a navigation color channel to detect motion and wraps raw images into a single reference frame for the recovery of quantitative phase [84]. However, the raw images of speckle SIM do not provide enough information to estimate the motion without reconstructing the object. We instead opt to model the motion dynamics of the scene in a compressive way and simultaneously recover the scene and the motion in our reconstruction. *Pnevmatikakis, et al.* expressed the fluorescence images of neural activity as a product of a spatial matrix and a temporal matrix to represent calcium imaging [141]. Instead of the matrix decomposition, we use neural representations to disentangle the spatio-temporal relationship.

Neural representations use untrained artificial neural networks to reproduce a high-dimensional matrix or tensor in a compressive way for image reconstruction [107]. The coordinate-based MLP [164] has been used successfully in recent years in novel view synthesis for static [115] and dynamic scenes[135, 144], 3D geometry representation [159, 160, 111, 134], and partial differential equation solution [146]. Coordinate-based MLP maps a coordinate to its corresponding value, and it is suitable to represent a smooth, continuous object. Recent works improve the coordinate-based MLP's capacity for high-frequency information with sinusoidal Fourier features [115, 172] or periodic activation functions [160, 111]. The coordinate-based MLP has also demonstrated an improved 3D reconstruction in the settings of computed tomography [169] and optical diffraction tomography [104].

## 4.2 Theory on Structured Illumination Microscopy

In this section, we review the theory of sinusoidal and speckle structured illumination for super-resolution in a coherent imaging system. In a diffraction-limited system with the incident illumination field $U_{\text{in}}$ and the scene $o$, the measured output field, $U_{\text{out}}$, can be expressed as

$$U_{\text{out}}(\boldsymbol{r}) = \mathcal{F}^{-1}\left[\mathcal{F}\left(U_{\text{in}}(\boldsymbol{r}) \cdot o(\boldsymbol{r})\right) \cdot P(\boldsymbol{u})\right], \tag{4.1}$$

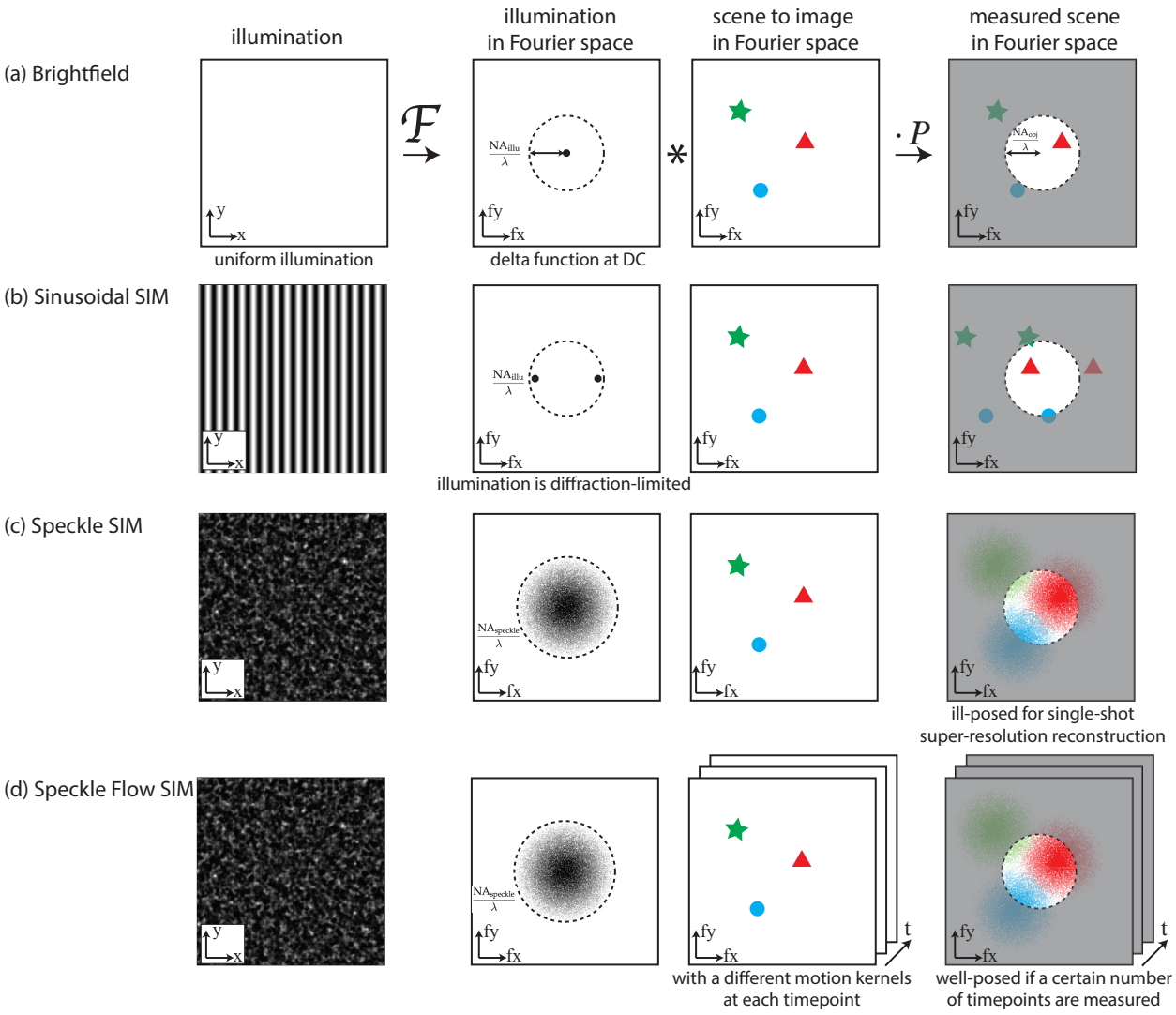


Figure 4.2: Illustrations of spatial frequency information for (a) brightfield microscopy, (b) sinusoidal structured illumination microscopy (SIM), (c) speckle SIM, and (d) Speckle Flow SIM. The first column shows the illumination intensity at the sample. The second column is the amplitude of the illumination in Fourier space. The third column illustrates the spatial frequency information of a sample scene we want to image (note that this scene is sparse in Fourier space only for simplicity of visualization). The last column shows the measured spatial frequency bandwidth in Fourier space after passing through a diffraction-limited microscope (the grayed-out areas cannot be measured). More details in Sec. 4.2.

where $\mathcal{F}$ denotes 2D Fourier transform, and $\boldsymbol{r}$ and $\boldsymbol{u}$ are spatial coordinates in real and frequency space. The magnification is assumed to be 1 for simplicity. The pupil function, $P$, is a circular binary mask defined as $P(\boldsymbol{u}) = 1$ for $|\boldsymbol{u}| \leq \mathrm{NA}_{\mathrm{obj}}/\lambda, P(\boldsymbol{u}) = 0$ if otherwise, where $\lambda$ is the wavelength of the illumination and $\mathrm{NA}_{\mathrm{obj}}$ is the NA of the objective lens. In the case of brightfield illumination, $U_{\mathrm{in}}$ is uniform and the pupil function directly low-pass filters the high-frequency information of the scene (Fig. 4.2(a)). The diffraction-limited resolution is set by the reciprocal of the pupil function bandwidth, i.e., $\lambda/\mathrm{NA}_{\mathrm{obj}}$.

In sinusoidal SIM, the incident illumination is sinusoidal such that $U_{\mathrm{in}}(\boldsymbol{r}) = \cos(2\pi \boldsymbol{v_0} \cdot \boldsymbol{r})$, where $\boldsymbol{v_0}$ is the spatial frequency of the illumination pattern. This frequency shifts the spectrum of the scene such that higher spatial frequencies can be measured (Fig. 4.2(b)), enabling super-resolution. The raw measurement of SIM can be expressed as

$$U_{\mathrm{SIM}}(\boldsymbol{r}) = \mathcal{F}^{-1}\left[\frac{\tilde{o}(\boldsymbol{u} - \boldsymbol{v_0}) + \tilde{o}(\boldsymbol{u} + \boldsymbol{v_0})}{2} \cdot P(\boldsymbol{u})\right], \tag{4.2}$$

where $\tilde{\cdot}$ denotes the quantity in 2D Fourier transform space. The frequency of the sinusoidal pattern, $\boldsymbol{v_0}$, is also diffraction-limited in the far field such that $|\boldsymbol{v_0}| \leq \mathrm{NA}_{\mathrm{illu}}/\lambda$. The final resolution is $\lambda/(\mathrm{NA}_{\mathrm{obj}} + \mathrm{NA}_{\mathrm{illu}})$ which gives a $2\times$ resolution gain when $\mathrm{NA}_{\mathrm{obj}} \approx \mathrm{NA}_{\mathrm{illu}}$. As each measurement $U_{\mathrm{SIM}}$ is band-limited by the pupil function $P$ and only observes a fraction of the super-resolved scene $\tilde{o}$, multiple measurements are needed to recover a single image.

Speckle SIM uses random speckle illumination, $U_{\mathrm{sp}}$, instead of sinusoidal illumination. The random speckle in Fourier transform space, $\tilde{U}_{\mathrm{sp}}$, also contains features from higher spatial frequencies, so super-resolved information can be encoded into a diffraction-limited measurement as in sinusoidal SIM (Fig. 4.2(c)). A speckle SIM measurement can be expressed mathematically as

$$\begin{aligned} U_{\mathrm{SpeckleSIM}}(\boldsymbol{r}) &= \mathcal{F}^{-1}\left[\mathcal{F}\left(U_{\mathrm{sp}}(\boldsymbol{r}) \cdot o(\boldsymbol{r})\right) \cdot P(\boldsymbol{u})\right] \\ &= \mathcal{F}^{-1}\left[\left(\tilde{U}_{\mathrm{sp}}(\boldsymbol{u}) * \tilde{o}(\boldsymbol{u})\right) \cdot P(\boldsymbol{u})\right], \end{aligned} \tag{4.3}$$

where $*$ denotes convolution operation. The speckle SIM can encode a frequency bandwidth of $(\mathrm{NA}_{\mathrm{obj}} + \mathrm{NA}_{\mathrm{speckle}})/\lambda$, where $\mathrm{NA}_{\mathrm{speckle}}$ is the effective NA of the speckle illumination. Similar to sinusoidal SIM, each raw measurement of speckle SIM is also band-limited by $P$, and multiple raw measurements are needed to decode the super-resolved information. In practice, a varying speckle illumination is often needed to collect raw measurements for a well-conditioned super-resolution reconstruction [120, 202].

In Speckle Flow SIM, we use a single static speckle illumination pattern and instead rely on the inherent motion dynamics of the scene to diversify the measured information (Fig. 4.2(d)). We acquire a sequence of frames for a dynamic scene that can be represented as $o(\boldsymbol{r}, t) = \mathit{motion}(o(\boldsymbol{r}), \boldsymbol{\mu}(\boldsymbol{r}, t))$. $\mathit{motion}(\cdot, \cdot)$ is a motion function transforming a time-independent scene, $o(\boldsymbol{r})$, to a timepoint by its spatially-varying motion kernel, $\boldsymbol{\mu}(\boldsymbol{r}, t)$. Speckle Flow SIM can be formally expressed as

$$\begin{aligned} &U_{\mathrm{SpeckleFlow}}(\boldsymbol{r}, t) \\ &\quad = \mathcal{F}^{-1}\left[\mathcal{F}\left(U_{\mathrm{sp}}(\boldsymbol{r}) \cdot \mathit{motion}(o(\boldsymbol{r}), \boldsymbol{\mu}(\boldsymbol{r}, t))\right) \cdot P(\boldsymbol{u})\right]. \end{aligned} \tag{4.4}$$

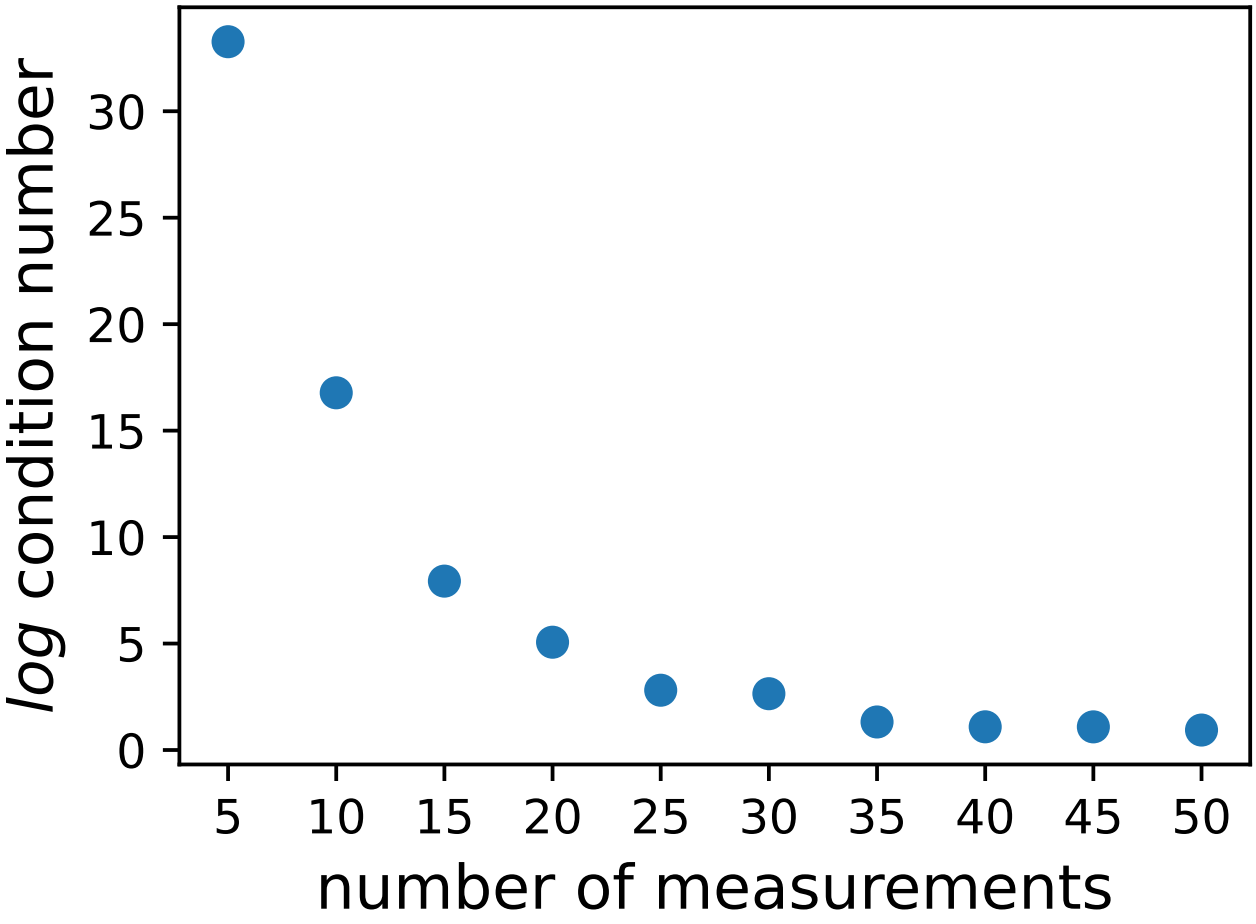


Figure 4.3: Condition number analysis for Speckle Flow SIM in 1D with increasing numbers of raw measurements. This suggests that Speckle Flow SIM becomes well-posed with a sufficient number of raw measurements.

It is difficult to analyze the well-posedness of the Speckle Flow SIM inverse problem for deformable motion. Hence, we show here only a simplified analysis for the case of translational motion, and assume the result will be similar for deformable motion. In the translational motion case, $motion\left(o\left(\boldsymbol{r}\right), \boldsymbol{\mu}\left(\boldsymbol{r}, t\right)\right) = o\left(\boldsymbol{r} + \delta\boldsymbol{r}_t\right)$, where $\delta\boldsymbol{r}_t$ is the relative displacement at timepoint $t$, and thus Eq. 4.4 can be written as a linear system:

$$\begin{aligned} U_{\text{SpeckleFlow}}\left(\boldsymbol{r}, t\right) &= \mathcal{F}^{-1}\left[\mathcal{F}\left(U_{\text{sp}}\left(\boldsymbol{r}\right) \cdot o\left(\boldsymbol{r} + \delta\boldsymbol{r}_t\right)\right) \cdot P\left(\boldsymbol{u}\right)\right] \\ &= \mathcal{F}^{-1}\left[\left(\tilde{U}_{\text{sp}}\left(\boldsymbol{u}\right) * \left(\tilde{o}\left(\boldsymbol{u}\right) \cdot e^{-2\pi i \delta\boldsymbol{r}_t \cdot \boldsymbol{u}}\right)\right) \cdot P\left(\boldsymbol{u}\right)\right]. \end{aligned} \tag{4.5}$$

Because of the translation property of Fourier transforms, the motion kernel in Fourier space can be expressed as $e^{-2\pi i \delta\boldsymbol{r}_t \cdot \boldsymbol{u}}$. If we rewrite this linear forward model into a transformation matrix, the transformation matrix changes at different timepoints as the scene moves. We

then perform a condition number analysis for using this forward model to solve for 2× super-resolution. In order to make the singular value decomposition (SVD) of the condition number calculation computationally feasible, we only consider the 1D case, where the scene and the measurement are both one-dimensional. The results (Fig. 4.3) show that the condition number of Speckle Flow SIM is dependent on the number of raw measurements captured, and the problem is better posed as more raw measurements are added, as expected. Beyond having a sufficient number of measurements, the super-resolution reconstruction is well-posed and the condition number becomes asymptotic.

## 4.3 Neural Space-Time Model

### Coordinate-based Neural Networks

The coordinate-based neural network is an alternative representation of a grid-based matrix using a MLP. The coordinate-based neural network takes an arbitrary coordinate of the matrix as the input and outputs the corresponding matrix value. To represent a 2D scene, a coordinate $\boldsymbol{r} = (x, y)$ within the domain of interest is used as the input coordinate. The weight, $\theta$, of the MLP, $f$, is optimized to fit into the given matrix [160, 111], such that

$$\arg\min_{\theta} \sum_{\boldsymbol{r} \in domain(o)} \left| f\left(\boldsymbol{r}; \theta\right) - o\left(\boldsymbol{r}\right) \right|^2. \tag{4.6}$$

Compared with a matrix representation, the coordinate-based MLP has a continuous form without the matrix grid, such that any off-grid coordinate values can be queried without additional rounding or interpolation. Once the MLP weights are optimized, we can retrieve a matrix with an arbitrary sampling grid by querying all the corresponding coordinates from the MLP. The coordinate-based MLP also tends to have a smoothing effect on the retrieved matrix because of the linearity of its fully-connected layers [172].

Positional encoding maps a coordinate into a vector of sinusoidal features at different frequencies before feeding into the network. This helps avoid over-smoothing and enables the representation of high-frequency details [172]. Positional encoding, $\gamma$, can be formally written as

$$\gamma\left(\boldsymbol{r}\right) = \left(\boldsymbol{r}, \cos\left(2^i \pi \boldsymbol{r}\right), \sin\left(2^i \pi \boldsymbol{r}\right), ...\right), \text{for } i = 0, ..., l-1. \tag{4.7}$$

$l$ is the order of positional encoding, which is a tunable parameter. As a result, positional encoding maps $\boldsymbol{r} \in \mathbb{R}^2$ to $\gamma\left(\boldsymbol{r}\right) \in \mathbb{R}^{4l+2}$ as the input for the coordinate-based MLP.

The neural space-time model is a compressive representation of a dynamic scene. The dynamic scene is split into two parts under the neural space-time model: motion kernels corresponding to different timepoints stored in the motion MLP, $f_{motion}$, and a time-independent scene represented by the scene MLP, $f_{scene}$. Both MLPs are coordinate-based.

The motion MLP acts as the motion kernel in Eq. 4.4, which transforms a dynamic scene into a time-independent scene. For any space-time coordinate, $(\boldsymbol{r}, t)$, the motion MLP

estimates the relative displacement with respect to the scene captured by the scene MLP, $\delta \boldsymbol{r}_t$. The dynamic scene can be expressed using the motion MLP, namely,

$$o(\boldsymbol{r}, t) = o(\boldsymbol{r} + \delta \boldsymbol{r}_t) = o(\boldsymbol{r} + f_{motion}(\boldsymbol{r}, t; \theta_{motion})), \tag{4.8}$$

where $\theta_{motion}$ is the weights of the motion MLP. Since the relative displacement returned from the motion MLP can vary spatially, the motion MLP can represent both global motion and locally deformable motion dynamics. When the motion MLP is queried for every spatial coordinate at a given timepoint, the obtained motion kernel can be used to map the current scene to the time-independent scene.

The scene MLP captures a time-independent, super-resolved scene. I.e., we can feed in any spatial coordinate to the scene MLP and obtain its corresponding value from the MLP output, $\hat{o}(\boldsymbol{r}) = f_{scene}(\boldsymbol{r}; \theta_{scene})$, where $\theta_{scene}$ is the weights of the scene MLP. The scene MLP does not take the timepoint as an input, as the time-dependency is already account in the motion MLP. We use $\hat{o}$ here since the scene MLP output is an approximated value from the coordinate-based MLP network, which might not be exact.

Combining these two parts together, as shown in Fig. 4.1(b), we can obtain the pixel value at any spatial and temporal coordinate by querying the estimated motion from the motion MLP first and then using the motion-accounted spatial coordinates to retrieve the super-resolved scene from the scene MLP. The positional encoding is applied to the motion-accounted coordinate before feeding into the time-independent scene MLP. The final approximated scene can be expressed as

$$\hat{o}(\boldsymbol{r}, t; \theta_{motion}, \theta_{scene}) = f_{scene}(\gamma(\boldsymbol{r} + f_{motion}(\boldsymbol{r}, t; \theta_{motion})); \theta_{scene}). \tag{4.9}$$

We repeat this process for each pixel of our sampling frame to retrieve a scene at a given timepoint, $\hat{o}(t; \theta_{motion}, \theta_{scene})$. We can input the retrieved scene into the forward model as Eq. 4.4 to simulate the raw image captured by the camera.

## Dynamic Scene Reconstruction

The neural space-time model recovers a dynamic scene from the weights of the motion and the scene MLPs. During the reconstruction, the model weights are optimized to minimize the loss function, which is the difference between the acquired intensity image at timepoint $t$, $I_t$, and the simulated intensity image using the forward model described in Sec. 4.3. This optimization can be formulated as

$$\underset{\theta_{motion}, \theta_{scene}}{\arg\min} \sum_t \left\| \sqrt{I_t} - \left| \mathcal{F}^{-1}\left[ \mathcal{F}\left( U_{\text{sp}} \cdot \hat{o}(t; \theta_{motion}, \theta_{scene}) \right) \cdot P \right] \right| \right\|^2. \tag{4.10}$$

We drop the spatial coordinates, $\boldsymbol{r}$ and $\boldsymbol{u}$, in this expression for simplicity. The complex-field of the speckle illumination, $U_{\text{sp}}$, is predetermined in a separate calibration process (described in Sec. 4.4) before the reconstruction. As both the forward model and the MLPs

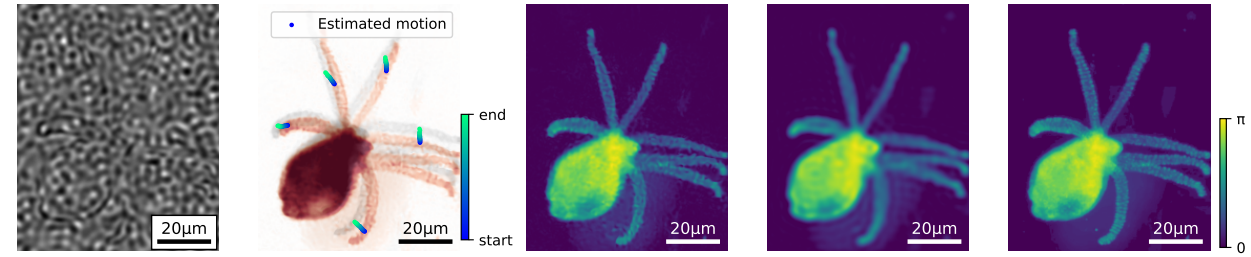


Figure 4.4: Simulation results for Speckle Flow SIM dynamic scene reconstruction of hydra with deformable motion. (a) The first frame of intensity measurements. (b) Recovered deformable motion trajectories for selected points are drawn as color gradient lines, where a point's color indicates its corresponding timepoint and the first (red) and last frame (grey) of the reconstruction are overlaid. (c) The reconstructed phase at the first timepoint of the dynamic scene. (d) Diffraction-limited phase obtained by low-pass filtering from (e) the true phase.

are differentiable, the gradients for those two MLPs' weights can be computed through back-propagation, and we then update the weights, $\theta_{motion}$ and $\theta_{scene}$, using gradient descent. We also note that $I_t$ can be replaced by the measured complex-field using an interferometric-based system, such as in [34].

While the reconstruction does not require any regularization terms, common regularizers for matrix-based inverse problem solution (e.g., L1, Tikhonov, total variation, etc.) are also available for our reconstruction. However, unlike in the matrix representation, applying a spatial filter directly to update the weights of a coordinate-based MLP is difficult. The experimental data reconstruction often generates some very high-frequency signals caused by imaging noise. To filter out these from the model weights, we have a high-frequency suppression term, $\mathcal{L}_{high\text{-}freq}$, to penalize the reconstruction of the signals beyond the theoretical super-resolution limit of Speckle Flow SIM. I.e., $\mathcal{L}_{high\text{-}freq} = \|\mathcal{F}(o(t))(1 - P')\|_1$, where $P'$ is the pupil function for the effective NA of Speckle SIM, $\mathrm{NA}_{speckle} + \mathrm{NA}_{obj}$.

## 4.4 Implementation Details

### Neural Space-Time Model Setup

The dynamic scene reconstruction is computation and memory intensive. As the scene is stored in the coordinate-based neural networks, obtaining the value for each pixel of the scene takes a query to the neural space-time model with its coordinate. Since the forward model requires the full matrix of a scene to perform a Fourier transform, we need to repeat this query for an entire scene (usually at the scale of a million pixels) to simulate

a measured image. Besides, the model's intermediate output values for each query are stored in the memory for an efficient gradient computation, and thus numerous copies of the intermediate output need to fit into the memory for the reconstruction in Eq. 4.10. To make this computationally feasible, we use a compact network configuration for the motion and scene MLPs. The motion MLP has the network depth of 4 and width of 32. While the spatial coordinates are fed into the motion MLP directly, the time coordinate is encoded with a positional encoding order of 4 for the motion MLP's input as in [135]. The scene MLP has a network depth of 8 and width of 64, which is larger than the motion MLP as we expect the scene to contain more information than the motion. The scene MLP uses a skip connect to concatenate the input to the fifth layer's output as in [115]. As the MLP outputs real values, the scene MLP has two output channels for phase and absorption of the scene respectively. Both MLPs use ReLU activation function after each fully-connected layer. The reconstruction using our network configuration can be performed in a single Titan Xp GPU (Nvidia) with 12 GB memory. Without an exhaustive testing of other configurations due to our limited computational resources, we also find our configuration robust for different scenes or dynamics. Nevertheless, having larger MLPs might still help to represent a more complex scene and motion as suggested in the universal function approximator theory [75].

## Reconstruction Procedure

We implement the neural space-time model and the reconstruction using Jax library (Google) [15] and in-house developed computational imaging toolbox[1]. We use ADAM, a fast version of the stochastic gradient descent, for the optimization of the neural space-time model. The motion and scene MLPs are updated concurrently under the same setting. The learning rate is set to $5 \times 10^{-4}$ for simulation and $5 \times 10^{-5}$ for experimental data, with a exponential delay to 0.1 of the starting value at the end. The reconstruction process takes 200k update steps for simulation data and 500k steps for experimental data. We only optimize for the data fidelity term as in Eq. 4.10 without the high-frequency suppression or other regularization terms for simulation data. For experimental data, we tried three different settings: vanilla reconstruction (no regularization), the reconstruction with high-frequency suppression, and the reconstruction with high-frequency suppression and speckle update (described in Sec. 4.4). The high-frequency suppression has a weighting factor of $1 \times 10^{-5}$ when included.

## Simulation Setup

We first perform a simulation study to validate Speckle Flow SIM. A speckle illumination is generated by a plane wave passing through a thin, random phase mask, and we low-pass filter the phase mask such that $\mathrm{NA}_{speckle} = \mathrm{NA}_{obj}$. The field of the speckle illumination is known during the reconstruction of the simulation data. We use two phase phantoms in the simulation: the Shepp-Logan phantom with a rigid motion we define using translation and

[1] Code will be released upon publication.

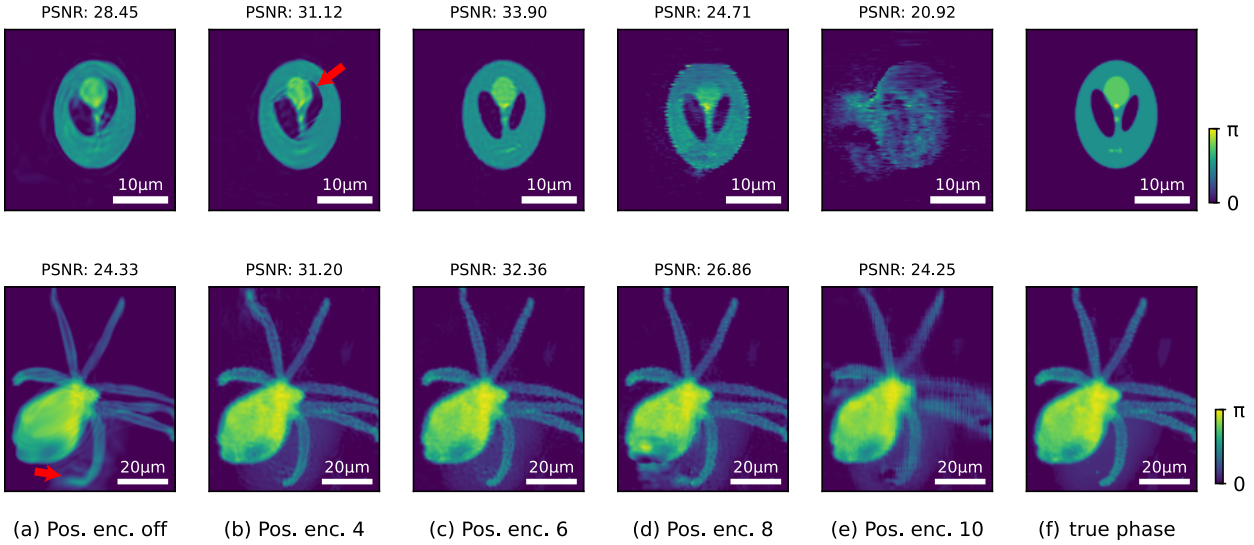


Figure 4.5: Dynamic scene reconstructions for Shepp-Logan phantom and hydra phantom with different positional encoding orders. The phase reconstruction at the first timepoint of the sequence is shown. The peak signal-to-noise ratio (PSNR) is calculated for each reconstructed sequence. The red arrows indicate the distortions caused by the inexact motion estimation.

rotation (40 frames total), and a video of hydra with a deformable motion (20 frames total). Then, we simulate a sequence of measurements frame-by-frame using the forward model of speckle SIM as in Eq. 4.3. The simulated sequence of intensity images is used as the input for the Speckle Flow SIM reconstruction.

## Experiment Setup

We build a custom microscope system as Fig. 4.1(a). A 532 nm green laser light (Thorlabs CPS532, 4.5 mW) outcoming from a single-mode fiber is collimated into a plane wave which then shines on a random diffuser for the speckle illumination. We use four layers of Scotch tape (3M 810 Scotch Tape, S-9783) as the random diffuser [201]. The layered Scotch tape is attached on a clear microscope slide for enhanced stability. After propagating through the sample, the transmitted light is magnified by a $4f$ system composed of a $10\times$ 0.25NA objective lens (Nikon) and a single lens (250 mm focal length) as the tube lens, resulting in an effective magnification of 12.5. The magnified image is captured by a monochromatic CMOS sensor (FLIR, BFS-U3-200S6M-C) placed at the back focal plane of the tube lens. The exposure time is set to maximize the dynamic range of the sensor.

### Speckle Calibration

We need to calibrate for the complex field of the speckle illumination in advance before the experimental data reconstruction. The calibration can be performed by simply imaging the background in a holographic setup. In our intensity-only system, however, we retrieve the phase of the speckle illumination by taking intensity images at different defocus planes [2]. The intensity image at each defocus plane, $I_{z_i}$, is modeled as the in-focus speckle field propagating by a distance of $z_i$ using the angular spectrum method [61], such that

$$I_{z_i}(\boldsymbol{r}) = \left| \mathcal{F}^{-1}\left( \tilde{U}_{sp}(\boldsymbol{u}) \cdot P(\boldsymbol{u}) \cdot \exp\left( j \frac{2\pi z_i \sqrt{1-\boldsymbol{u}^2}}{\lambda} \right) \right) \right|^2 . \tag{4.11}$$

We then solve for $U_{sp}$ from the acquired defocused images $I_{z_i}$.

In the experimental setup, we place the CMOS sensor on a 1-axis motorized stage (Thorlabs Z825) toward the $z$ direction. To retrieve the phase of the speckle illumination, we acquire intensity images at 10 $z$-planes in the image space with a step size of 200 µm, which is equivalent to 1.28 µm in the sample space. This defocus calibration process is performed on the background field-of-view without the scene. After the acquisition, the complex field of the speckle illumination is iteratively updated to minimize Eq. 4.11 using gradient descent.

The speckle field retrieved from this calibration process may not perfectly match the actual speckle illumination in the dynamic scene acquisition for experimental reasons, such as mechanical instability. Thus, the complex field of the speckle can also be jointly updated to minimize the objective function defined in Eq. 4.10 during the reconstruction of a dynamic scene. As the speckle update affects the scene reconstruction, the speckle update is performed only in the fine-tuning stage of the dynamic scene reconstruction, which is after the first 10k update steps in our case.

## 4.5 Simulation and Experimental Results

### Simulation Results

We first validate the reconstruction of the hydra phantom with deformable motion in simulation. As the hydra phantom in our simulation is phase-only, the simulated intensity measurement in Fig. 4.4(a) looks similar to the speckle image without the sample and does not contain much discernible contrast without the reconstruction. After the reconstruction, the motion trajectories recovered by the neural space-time model are a good fit for the actual scene dynamics as in Fig. 4.4(b), which demonstrates the motion MLP's capacity of representing deformable motion. The phase reconstruction for the first frame of the scene is shown in Fig. 4.4(c). Compared with the diffraction-limited reference scene in Fig. 4.4(d) (low-pass filtered from the true scene), the reconstruction recovers those finer features on both the hydra's gastrovascular cavity and tentacles, matching well to the original ground truth phantom. It is worth noting that the space-time model here successfully reconstructs

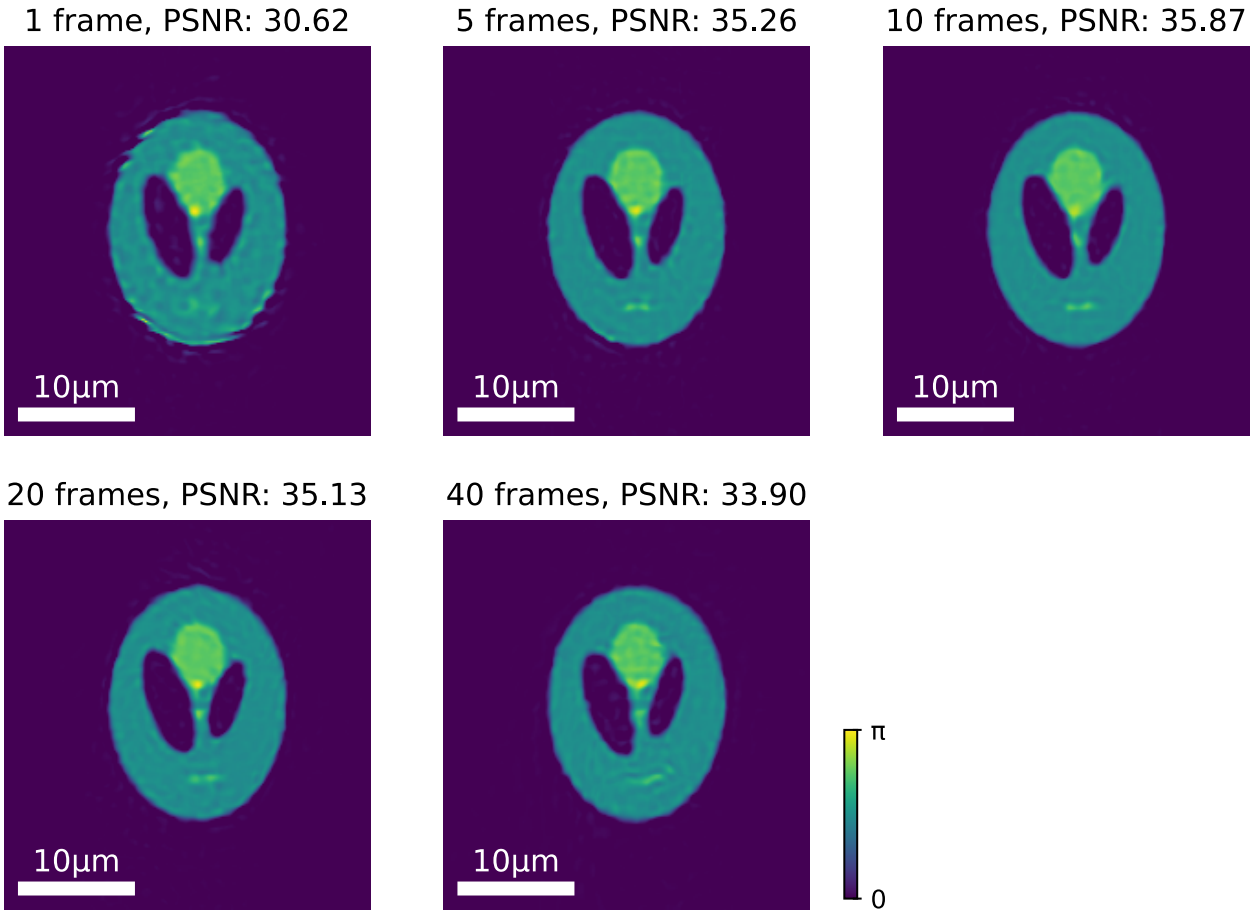


Figure 4.6: The phase reconstructed using the first 1, 5, 10, 20, 40 frames of the input intensity image sequence. The reconstructed phase at the first timepoint is shown here. The peak signal-to-noise ratio (PSNR) is calculated for the reconstruction over all timepoints. Based on the PSNR, the reconstruction quality is optimal using 10 frames.

the discontinuous and non-smooth features from the scene MLP, e.g. fine features on the tentacles, which can be credited to the non-linearity from the activation function and the high-frequency mapping from positional encoding.

We further analyze the effect of positional encoding on the reconstruction performance. By its definition in Eq. 4.7, the order of positional encoding, $l$, regulates the highest frequency of the positional encoding mapping, which in turn affects the reconstruction of high-frequency signal [172]. With the same set of input raw measurements and reconstruction settings, we reconstruct the dynamic scene with different orders of positional encoding of the spatial coordinates in the scene MLP, as shown in Fig. 4.5. When the positional encoding is turned off or the order is low (order of 4), the reconstructions are under-fitted and also have distortions as indicated by red arrows. The distortions are caused by the inexact motion

estimation for some small features. The positional encoding order of 6 gives the optimal reconstruction in terms of its peak signal-to-noise ratio (PSNR) and visual quality for both phantoms. When the order is set to 10, the reconstructed phase contains a considerable scene distortion and high-frequency artifacts. This suggests that while a high order of positional encoding gives the scene MLP extra degrees-of-freedom for the scene representation, it hinders the convergence of the motion MLP at the same time. The scene MLP with a high order of positional encoding tends to over-fit high-frequency content, before a good motion estimation is obtained by the motion MLP.

We also compare the reconstructions with different numbers of input intensity images, such that we reconstruct using the first 1, 5, 10, 20, 40 images of the simulated measurement sequence for the Shepp-Logan phantom. With more input images, the neural space-time model receives more encoded information of the scene, while it is also responsible for recovering the motion dynamics at more timepoints. As in Fig. 4.6, despite being ill-posed, the phase can be reconstructed from a single frame due to the implicit network smoothness prior from the scene MLP. The reconstruction improves with more raw images and reaches the optimal quality using 10 frames. If even more frames are used, the reconstruction slowly degrades as the motion MLP has to recover an extended scene dynamic from an increasing number of timepoints. The optimality of 10 raw images roughly coincide with the number of raw images required for sinusoidal SIM.

## Experimental Results

As a proof-of-concept experiment for Speckle Flow SIM, we create a dynamic scene by placing an amplitude USAF-1951 resolution target (Benchmark Technologies) on a 1-axis motorized stage (Thorlabs Z812) that travels laterally ($xy$-direction). A sequence of 40 intensity images is acquired while the resolution target moves continuously. The defocus images for speckle calibration are captured before the actual image acquisition. As the speckle evolves with a different optical path length, we calibrate the speckle using a background field-of-view on the resolution target slide to minimize the potential mismatch. A diffraction-limited brightfield image is also acquired using the same system without the scattering layer as a reference.

The reconstruction is performed with three different settings detailed in Sec. 4.4. The reconstruction is shown in Fig. 4.7(a)-(c). The brightfield image is shown in Fig. 4.7(e) for comparison. The vanilla reconstruction recovers the high-frequency information well despite being noisy, as demonstrated on the zoom-in and the line plot in Fig. 4.7(f). The noisy reconstruction is due to the mismatch between the calibrated speckle and the actual speckle for practical reasons, such as, subtle differences in optical path length, mechanical vibration and instability. Adding a joint speckle update reduces the noise in the reconstruction while still maintaining good contrast for fine features (Fig. 4.7(b)). When we add the high-frequency suppression regularization term, described in Sec. 4.3, to the reconstruction loss, the reconstruction in Fig. 4.7(c) becomes much smoother and less noisy at the cost of slightly reduced contrast.

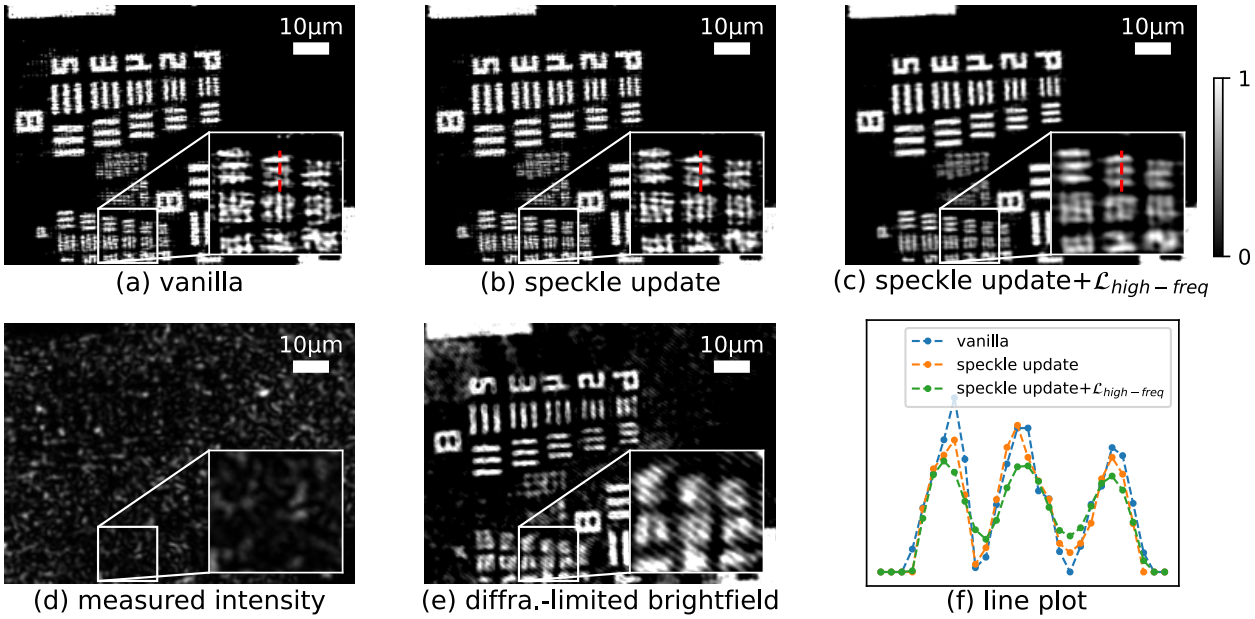


Figure 4.7: The experimental reconstruction of an amplitude USAF-1951 resolution target in continuous motion. (a)-(c) The reconstructed absorption coefficient at the first timepoint under different reconstruction settings described in Sec. 4.4. (d) The corresponding raw intensity image. (e) The diffraction-limited brightfield image as a reference. (f) Line plot for the red dotted lines in (a)-(c).

The diffraction-limited image has a theoretical Rayleigh resolution of $1.22 \cdot \lambda/\mathrm{NA} = 2.60$ µm, which corresponds to Group 8 Element 5 of the USAF resolution target (2.46 µm) and matches our observation in Fig. 4.7(e). In Speckle Flow SIM, our reconstruction resolves up to group 9 element 4 (1.38 µm), which is 1.88× of the diffraction-limited resolution. Since the speckle calibration is performed with the same objective lens such that $\mathrm{NA}_{speckle} \leq \mathrm{NA}_{obj}$, we expect a theoretical resolution gain of 2×, very close to our experimental demonstration. This experimental resolution improvement is similar to the reported gain from previous study of sinusoidal SIM [67] and speckle SIM [120] which are both close to 2×.

Similar to Sec. 4.5, we compare the reconstructions using different numbers of acquired intensity images. Fig. 4.8 shows the reconstruction using the first 8, 16, 24, 32, 40 acquired frames. The reconstruction is visually identical using 24, 32, 40 acquired frames, while some regions of the scene are missed when reconstructing with 8 or 16 frames. This is caused by the joint speckle update, such that the static speckle background becomes indistinguishable from the dynamic scene foreground when the number of raw frames is limited.

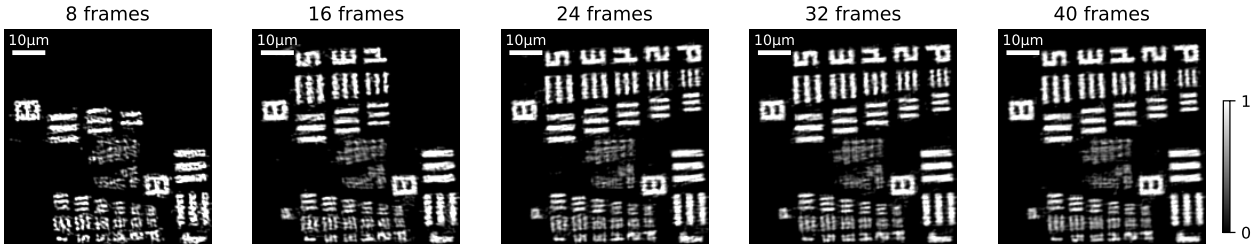


Figure 4.8: The experimental reconstruction of the moving USAF-1951 target using the first 8, 16, 24, 32, 40 frames of the acquired intensity image sequence. The reconstructed absorption coefficient at the first timepoint is shown here.

### Beyond 2× Super-resolution

Speckle SIM can achieve more than 2× better than diffraction-limited resolution when the illumination NA is higher than the objective NA, such that the final resolution is $\frac{\lambda}{\mathrm{NA_{obj}}+\mathrm{NA_{speckle}}} > 2 \cdot \frac{\lambda}{\mathrm{NA_{obj}}}$, as in Sec. 4.2. This setting can be useful for high-content imaging [201, 203] or for total internal reflection-based SIM settings [36]. Speckle Flow SIM can also similarly recover a dynamic scene with beyond 2× diffraction-limited resolution using a low-NA system and a fine speckle illumination.

We validate this in simulation, for an objective lens of 0.1 NA and known speckle illumination of 0.3 NA. The amplitude USAF-1951 resolution target with a constant-velocity translational motion is imaged, and we acquire the intensity images at 100 equally-spaced timepoints. The reconstruction is performed under the same optimization procedure as in Sec. 4.4. The reconstructed absorption coefficient as well as a few reference targets are shown in Fig. 4.9. The reference targets are obtained by directly low-pass filtering from the groundtruth target with the bandwidth of 2×, 3×, and 4× the diffraction limit. The reconstruction is able to resolve the second from the top element on the right group, which is close to the 4× diffraction-limited reference in Fig. 4.9, matching well with the theoretical limit in Sec. 4.2.

## 4.6 Discussion

### Limitations

Speckle Flow SIM has several limitations. First, the reconstruction process is computationally expensive and memory intensive as discussed in Sec. 4.4. The 500k iterations of experimental data reconstruction in Fig. 4.7 (matrix size: $400 \times 540$) takes around 13 hours on a Nvidia Titan Xp GPU using our current implementation. This reconstruction time is very long compared with other SIM methods for static scenes. E.g., the reconstruction of sinusoidal SIM takes 30s using one CPU core [67]); speckle SIM takes 5 hours using one CPU

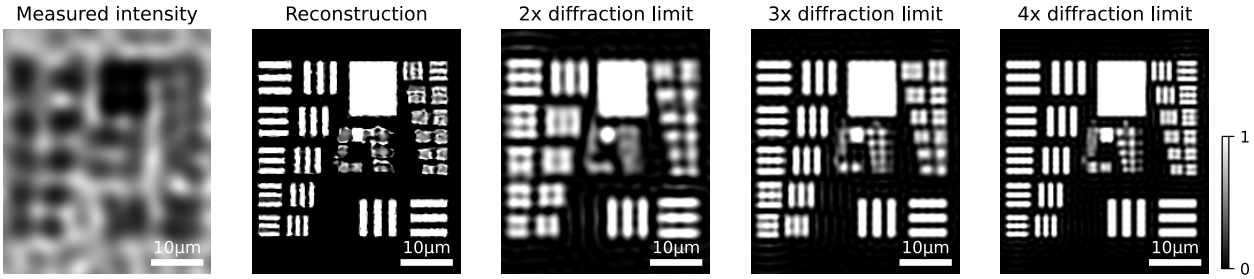


Figure 4.9: The reconstructed absorption coefficient for an amplitude USAF-1951 resolution target in motion achieves more than 2× better resolution than the diffraction limit in simulation. The numerical aperture (NA) of the speckle is 3× the NA of the objective lens. The target with 2×, 3×, and 4× the diffraction-limited resolution are shown as references. The reconstruction is close to 4× the diffraction-limited resolution.

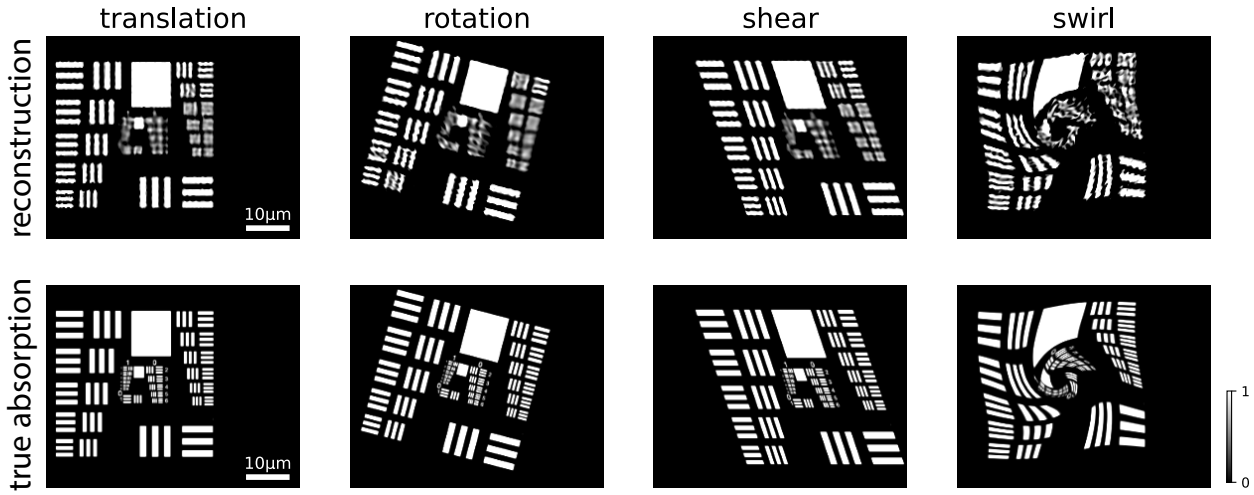


Figure 4.10: The reconstructed absorption coefficient for an amplitude USAF-1951 resolution target with four different types of motion in simulation. The performance of Speckle Flow SIM is motion-dependent and degrades with highly-deformable motion.

core [120] and 15 minutes on a GPU [201]). The main bottleneck for the reconstruction time is the coordinate-based MLPs in the neural space-time model, which requires one MLP's forward pass to know the corresponding value for each pixel coordinate. Speckle Flow SIM may benefit from two very recent studies suggesting a potential 100x speed-up with a more efficient sparse voxel representation and GPU programming [205, 122].

Second, the neural space-time model assumes that the scene at all timepoints can be wrapped into the time-independent frame represented by the scene MLP. This assumption is necessary to exploit the smoothness of motion using the motion MLP for a joint motion-

scene optimization and improve the SIM frame rate by an order of magnitude. However, this assumption also limits the applicability of Speckle Flow SIM, such that dynamics must be related by motion estimations, e.g., cells moving a microfluidic chamber. Speckle Flow SIM will fail to capture dynamics that are not smooth over time, e.g., random firing of neurons. Besides, the overall performance of Speckle Flow SIM is also motion-dependent. In Fig. 4.10, we reconstruct the USAF-1951 resolution target with four different types of motion dynamics (i.e., translation, rotation, shearing, and swirl) in simulation. The forward simulation and reconstruction settings are the same as in Sec. 4.4. As in Fig. 4.10, the reconstruction of the swirl motion contains more noise or artifacts than those of affine motion. This suggests that the performance of our method degrades for complex motion dynamics (e.g., highly deformable motion), presumably due to the inexact motion estimation from the motion MLP. As also discussed in Sec. 4.5 and Fig. 4.6, the representation capacity of a motion MLP is finite and limited, and thus a motion MLP with a compact network architecture may fail to fit into dramatic or highly deformable motion dynamics. A larger network architecture may help accommodate the estimation for more complex motion dynamics [75].

Third, Speckle Flow SIM requires a calibration process of the speckle illumination before the reconstruction, which is unlike previous speckle SIM methods. Previous methods either assumed a statistical prior on the speckle [120, 202, 110] or jointly resolved the speckle without any prior assumption [201, 203]. Having the full knowledge of the random speckle illumination allows Speckle Flow SIM to super-resolve a scene using much fewer raw images than other methods. Our current calibration process, however, limits Speckle Flow SIM to 2× the diffraction-limited resolution, as the calibration images are also captured in the same optical system and objective lens. A joint update of the speckle illumination during the reconstruction [201] could help to achieve more resolution gain with a low-NA objective for high-content imaging. Additionally, the calibrated speckle does not always match the actual illumination, making the reconstruction noisy as in Sec. 4.5. This mismatch can be prominent when the speckle contains features beyond the diffraction-limit of the system. Any changes between the calibration and the actual acquisition (e.g., phase shift on the illumination path) unaccounted for may also contribute to the speckle mismatch.

## Conclusion and Future Work

We demonstrate a super-resolution method for dynamic scenes, called Speckle Flow SIM, which illuminates the sample with speckle-structured illumination to observe high frequency information beyond the diffraction limit. Speckle Flow SIM does not change its illumination but relies on the dynamics of the scene to acquire diversified measurements. This enables a simple, inexpensive experimental setup. Without loss of generality, we model the spatio-temporal relationship of the dynamic scene using the neural space-time model with coordinate-based multi-layer perceptrons, which jointly recover the motion dynamics and the super-resolved scene from a temporal sequence of raw images. We validate the Speckle Flow SIM in simulation and experiment. We show that Speckle Flow SIM can reconstruct a

scene with deformable motion. We also experimentally demonstrate 1.88× of the diffraction-limited resolution for a dynamic scene.

Future work may extend Speckle Flow SIM into the fluorescence channel where super-resolution microscopy methods are more commonly used by biologists and neuroscientists. A high-NA system is also needed for ∼100 nm spatial resolution. Another future direction is to enable the dynamic imaging (i.e., improve the temporal resolution) for other multi-shot computational imaging systems using the neural space-time model. By plugging in different forward models, the space-time model may jointly estimate the scene and the motion dynamics when the motion is relatively smooth. Besides, the neural space-time model itself can be further optimized for better computational efficiency and less network-induced reconstruction artifacts. Even though our space-time model has a great flexibility in its model selection, it is not exploited in this chapter, and our model was determined ad-hoc from limited tries. A systematic search of the neural network architecture [47] and hyperparameters [52] may be beneficial.

# Chapter 5

# Neural space-time model for dynamic multi-shot imaging

Multi-shot computational imaging systems capture multiple raw measurements sequentially and combine them through computational algorithms to reconstruct a final image that enhances the capabilities of the imaging system (e.g., super-resolution [79, 68], phase retrieval [136], hyperspectral imaging [106]). Each raw measurement is captured under a different condition (e.g., illumination coding, pupil coding) and hence encodes a different subset of the information. The reconstruction algorithm must then decode this information to generate the final reconstruction.

If the sample is moving during the multi-shot capture sequence, the reconstruction may be blurry or suffer artifacts [54] since the system effectively encodes information from slightly different scenes at each timepoint. Thus, most methods require that the sample be static during the full acquisition time, which limits the types of samples that can be imaged. Approaches for imaging dynamic samples aim to reduce acquisition time by multiplexing measurements via hardware modifications [184, 138, 204], developing more data-efficient reconstruction algorithms [93, 69, 41], or deploying additional data priors with deep learning techniques [124, 198, 145, 192, 58, 163, 30, 143]. However, these methods may be impractical to implement, and usually are only applicable for a specific imaging system. Data priors, for example, are non-trivial to generate (e.g., due to the lack of access to groundtruth data) and may fail with out-of-distribution samples [9].

Here we take another approach for imaging moving samples, where we model the sample dynamics in order to account for it during the image reconstruction. Modeling sample dynamics in multi-shot methods is challenging for two reasons: First, each measurement has a different encoding, so we cannot simply register the raw images to solve for the motion. Second, the motion can be highly complex and deformable, necessitating a pixel-level motion kernel. Our approach is to use deep learning methods to develop flexible motion models that would be very difficult to express analytically. For example, recent work successfully used a

This chapter covers the research I presented or published in [26].

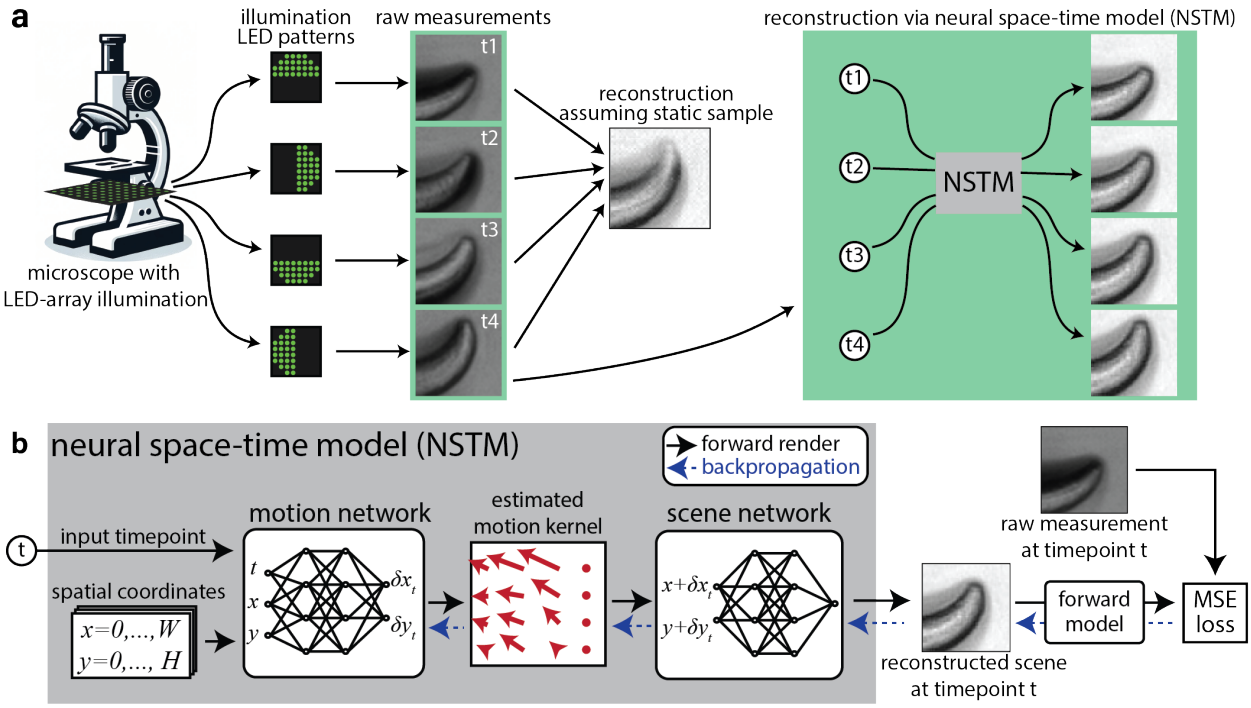


Figure 5.1: **a**, Multi-shot computational imaging systems capture a series of images under different conditions and then computationally reconstruct the final image. For example, differential phase contrast microscopy (DPC) captures four images with different illumination source patterns, and then uses them to reconstruct quantitative phase. Sequential capture of the raw data results in motion artifacts for dynamic samples, since the reconstruction algorithm assumes a static scene. Our proposed neural space-time model (NSTM) extends such methods to dynamic scenes, by modeling and reconstructing the motion at each timepoint. **b**, NSTM consists of two coordinate-based neural networks, one for the motion and one for the scene. Once the networks have been trained using the dataset of raw measurements, we can give the NSTM any timepoint as the input, and it will generate the reconstruction at that timepoint. The network weights of NSTM are trained to match the forward model-rendered measurement with the actual raw measurement at each timepoint.

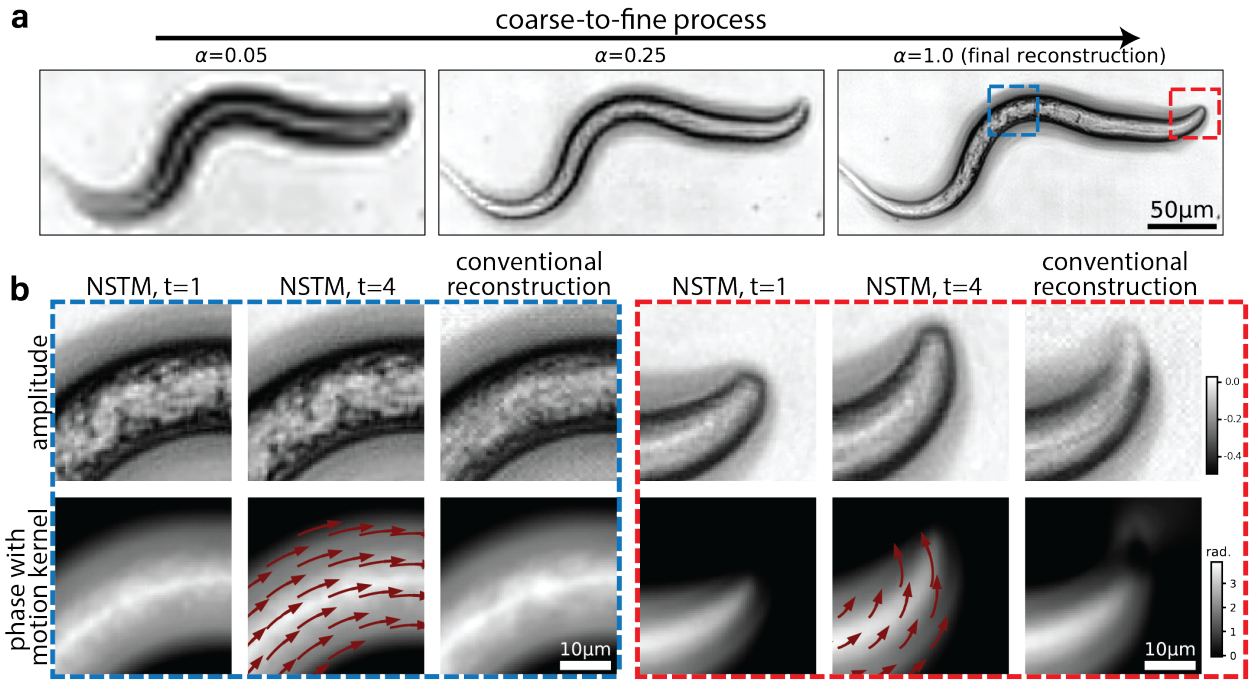


Figure 5.2: **a**, The coarse-to-fine process for the reconstruction of a live *C. elegans* worm imaged by DPC. **b**, Zoom-ins for NSTM reconstruction at different timepoints with the recovered motion kernel overlaid, along with a comparison to conventional reconstruction.

deep learning approach (with a robust data prior) to model dynamics in the case of single molecule localization microscopy [151].

We propose a neural space-time model (NSTM) that can recover a dynamic scene by modeling its spatiotemporal relationship in multi-shot imaging reconstruction. NSTM exploits the temporal redundancy of dynamic scenes. This concept, widely used in video compression, assumes that a dynamic scene evolves smoothly over adjacent timepoints. Specifically, NSTM models a dynamic scene using two coordinate-based neural networks; these networks store the multi-dimensional signal through their network weights, and are used for novel view-synthesis [116, 135], 3D object representation [160], and image registration [197, 19]. As illustrated in Fig. 5.1b, one network of NSTM represents the motion, and the other network represents the scene. The motion network outputs a motion kernel for a given timepoint, which estimates the motion displacement for each pixel of the scene. Subsequently, the scene network generates a scene using spatial coordinates that have been adjusted for motion by the motion network. Then, the generated scene is passed into the system's forward model to produce a rendered measurement. To train the weights of the two networks (which store the scene and its motion dynamics), we use gradient descent optimization to minimize the difference between the rendered measurements and the acquired measurements

(details in Section 5.1).

The motion and scene networks in NSTM are interdependent, and failing to synchronize their updates leads to poor convergence of the model. This poor convergence typically happens when the scene network overfits to the measurements before the motion is recovered, a situation common for scenes involving more complex motion (Fig. 5.3 & 5.4). To mitigate this issue, we developed a coarse-to-fine process (detailed in Section 5.1), which controls the granularity of the outputs from both networks. Specifically, the reconstruction starts by recovering only the low-frequency features and motion, and then gradually refines higher-frequency details and local deformable motion as illustrated in Fig. 5.2a.

NSTM is a general model for motion dynamics and can be plugged into any multi-shot system with a differentiable and deterministic forward model. It does not involve any pre-training or data priors; the learned network weights describe the final reconstructed video for each dataset individually, so it can be considered a type of 'self-supervised learning'. We demonstrate NSTM here for three different computational imaging systems: differential phase contrast microscopy (DPC) [175], 3D structured illumination microscopy (SIM) [68] and rolling-shutter DiffuserCam [4]. In future, we hope it will find use in other applications as well.

## 5.1 Implementation of Neural Space-Time Model

### Construction of NSTM

The motion and the scene network of NSTM are both coordinate-based neural networks [164, 160, 116], a type of multi-layer perceptrons that learn a mapping from coordinates to signals. A coordinate-based neural network can represent a multi-dimensional signal, *e.g.,* an image, a 3D scene, etc., through its network weights. To enhance the capacity and efficiency of the coordinate-based networks, we use hash embedding [122] to store multiple grids of features at different resolutions and transform a coordinate vector to a multi-resolution hash-embedded feature vector, $\boldsymbol{h} = [h_0, h_1, \cdots, h_{N-1}]$, before passing it into the network (details below). As the input coordinate varies, a fine resolution feature (e.g., $h_{N-1}$) changes more rapidly than a coarse resolution feature (e.g., $h_0$). During the coarse-to-fine process, we re-weight the output features of the hash embedding using a granularity value, $\alpha$, to control the granularity of the network. $\alpha$ is set by the ratio of the current epoch to the end epoch of the coarse-to-fine process, which is set to 80% of the total number of reconstruction epochs in practice. As in [135], each feature $f_i$ is weighted by $\frac{1}{2} - \frac{1}{2}\cos\left(\pi \cdot \text{trunc}\left(\alpha \cdot N - i\right)\right)$, where trunc truncates a value to $[0, 1]$. In this way, finer features will be weighted to 0 until $\alpha$ gets larger, as illustrated in Fig. 5.2a.

In the forward process of NSTM (Fig. 5.1b), every spatial coordinate of the scene, $\boldsymbol{x}$, is concatenated with the temporal coordinate $t$, and the hash-embedded features of the spatiotemporal coordinate, $hash(\boldsymbol{x}, t)$, are fed into the motion network. The motion network, $f\left(\cdot\,|\theta_{\text{motion}}\right)$, produces the estimated motion displacement vector, $\delta\boldsymbol{x}$, for each input

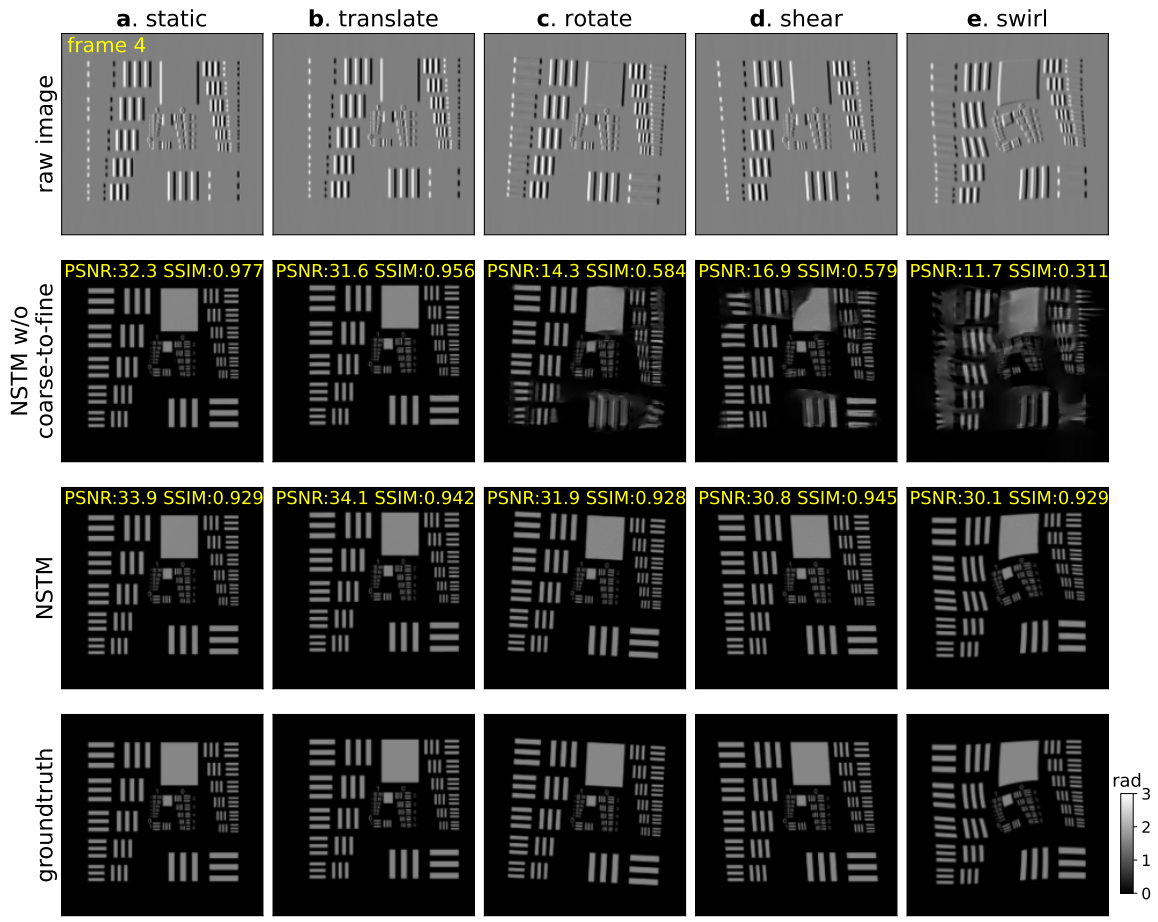


Figure 5.3: Simulations of differential phase contrast microscopy (DPC) using a phase-only USAF-1951 resolution target with various types of motion: **a**, no motion, **b**, rigid motion - translation, **c**, rigid motion - rotation, **d**, non-rigid global motion - shearing, and **e**, local deformable motion - swirl. We reconstruct the quantitative phase of the dynamic scene using NSTM with the set of four simulated DPC images. Two reconstruction quality metrics are calculated: peak signal-to-noise ratio (PSNR) and the structural similarity index measure (SSIM). The NSTM does well with all types of motion. However, without using our coarse-to-fine process ('NSTM w/o coarse-to-fine'), it is likely to fail as the motion gets complicated, due to poor convergence of the joint optimization of motion and scene.

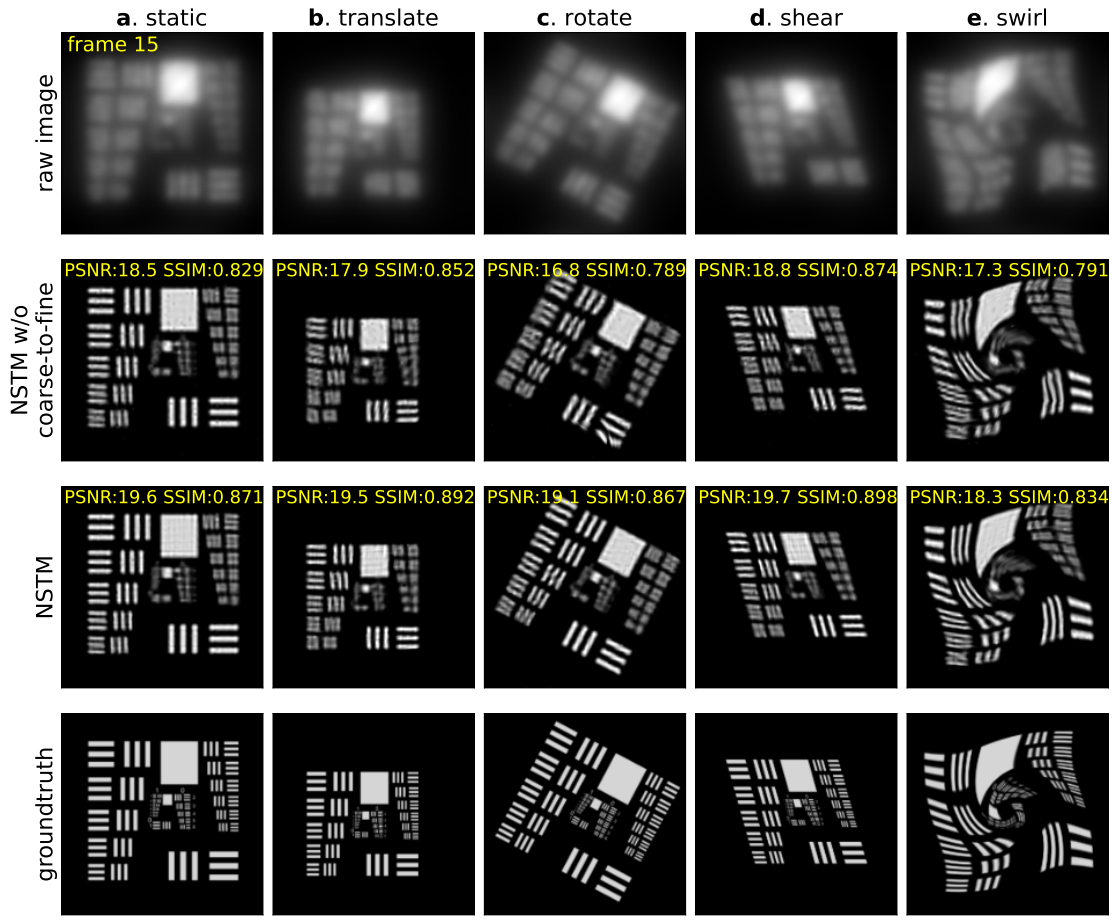


Figure 5.4: Simulations of structured illumination microscopy (SIM) using fluorescent USAF-1951 resolution target with various types of motion: **a**, no motion, **b**, rigid motion - translation, **c**, rigid motion - rotation, **d**, non-rigid global motion - shearing, and **e**, local deformable motion - swirl. The forward model of single-plane three-beam SIM is assumed for the simulation.

spatiotemporal coordinate:

$$\delta \boldsymbol{x} = f\left(hash\left(\boldsymbol{x}, t\right)|\theta_{\text{motion}}\right). \tag{5.1}$$

The motion-adjusted spatial coordinate, $(\boldsymbol{x} + \delta\boldsymbol{x})$ is then transformed into hash-embedded features and fed into the scene network, $f\left(\cdot\,|\theta_{\text{scene}}\right)$ for the reconstruction value, $o$, such that

$$o\left(\boldsymbol{x}, t\right) = f\left(hash\left(\boldsymbol{x} + \delta\boldsymbol{x}\right)|\theta_{\text{scene}}\right). \tag{5.2}$$

This process is repeated for all spatial coordinates to obtain the reconstructed scene at time $t$. As the scene network does not take the time as an input, it relies on the motion network to generate a dynamic scene. In our demonstrations, the scene network outputs a single channel as the fluorescent density for 3D SIM, two channels as the amplitude and phase for DPC, and three channels as RGB intensity for DiffuserCam. Since the hash embedding is always applied to the network input coordinate, we consider it a part of the network, $f$, and drop it from our expression for readability.

## NSTM reconstruction

To update the network weights of NSTM, the reconstructed scene is passed into the imaging system's forward model for a rendered measurement. Comparing the rendered measurement with the actual measurement acquired in the experiment, we compute the mean square error (MSE) loss and minimize it by back-propagating its gradient to update the network weights. Mathematically, the optimization becomes

$$\underset{\theta_{\text{motion}},\theta_{\text{scene}}}{\arg\min} \sum_{i\in\{0,\cdots,T-1\}} \left(forward_i\left(f\left(\boldsymbol{x} + f\left(\boldsymbol{x}, t_i\right)|\theta_{\text{motion}}\right)|\theta_{\text{scene}}\right) - I_i\right)^2, \tag{5.3}$$

where $forward_i$ is the forward model to render the $i$th measurement given the temporal coordinate $t_i$. The actual measurement captured at timepoint $t_i$ is denoted as $I_i$. Adapting NSTM to new computational imaging modalities thus amounts to simply dropping in the appropriate forward model.

In our implementation, the motion network has two hidden layers with a width of 32, and the scene network has two hidden layers with a width of 128. The gradient update is performed with Adam optimizer [90]. The initial learning rate is set to $1 \times 10^{-5}$ for motion network ($5 \times 10^{-5}$ for rolling-shutter DiffuserCam reconstruction) and $1 \times 10^{-3}$ for scene network, with a exponential decay schedule to a tenth of the initial learning rate at the end of the reconstruction. For the conventional reconstruction of NSTM without motion update (in Fig. 5.6a, Fig. 5.8c, and Fig. 5.11b), we keep all settings the same as the NSTM reconstruction except that the motion network is not updated and the input timepoints are set to zero.

## Hash embedding

Hash embedding store all features vector in its weights, *i.e.,* a list of feature storage array, $[\psi_0, \psi_1, \cdots, \psi_{N-1}]$. The hash embedding transforms a coordinate vector to a multi-resolution

feature vector, $\boldsymbol{h} = [h_0, h_1, \cdots, h_{N-1}]$. Starting from the basic case, when the coordinate vector $(x)$ is *1*-D. To obtain the feature at a particular resolution $h_i$, we first identify the nearest coordinate values on the resolution grid that are just greater or smaller than $x$, which are denoted as $\lceil x \rceil$ and $\lfloor x \rfloor$. Then, we define a fixed hash function to obtain the hash values for $\lceil x \rceil$ and $\lfloor x \rfloor$, and the hash values are used to retrieve the features corresponding to $\lceil x \rceil$ and $\lfloor x \rfloor$ from the $\psi_i$. Lastly, we linearly interpolate these retrieved features for the feature for $x$. Putting this mathematically,

$$h_i(x) = (x - \lfloor x \rfloor) \cdot \psi(hash(\lceil x \rceil)) + (\lceil x \rceil - x) \cdot \psi(hash(\lfloor x \rfloor)). \tag{5.4}$$

Generalizing this into $N$-D, we will find $2^N$ nearest coordinate vectors and perform $N$-D interpolation based on $2^N$ of the retrieved features. By repeating this process for each resolution, we concatenate features from all resolutions for the hash embedded features as the input of the coordinate-based neural network. In our notation, the hash embedding weights, $\psi$, are considered as a part of the network weights, $\theta$, and thus $\psi$ is not written out in Eqs. 5.1-5.3. They are updated together using the same learning setting the during the reconstruction.

## 5.2 Differential phase contrast microscopy

Our first multi-shot computational imaging system, differential phase contrast microscopy (DPC), captures four raw images, from which it reconstructs the amplitude and phase of a sample [175]. The images are captured with four different illumination source patterns, which are generated by an LED array microscope in which the traditional brightfield illumination unit is replaced by a programmable LED array [139].

The raw images of DPC are normalized by the background intensity, and then passed through the linear transfer functions derived in [175] as the forward model:

$$forward_i(o_u, o_p) = \mathcal{F}_{2D}^{-1}\left[H_u^i \cdot \mathcal{F}_{2D}(o_u) + H_p^i \cdot \mathcal{F}_{2D}(o_p)\right], \tag{5.5}$$

where $\mathcal{F}_{2D}$ is 2D Fourier transform, $H_u^i, H_p^i$ denote the absorption and phase transfer functions for the $i$th measurement, and $o_u, o_p$ are the absorption and quantitative phase of the scene. The conventional reconstruction is obtained by solving a Tikhonov regularization with a regularization weight of $10^{-4}$ for both amplitude and phase terms [175]. For ease of comparison, we add the same Tikhonov regularization to the loss term for NSTM reconstruction.

In Fig. 5.1a, we show the system and raw images captured for a live, moving *C. elegans* sample. The conventional reconstruction algorithm assumes a static scene over these four raw images. Consequently, unaccounted sample motion leads to artifacts in the reconstruction (Fig. 5.2b). Through the coarse-to-fine process (Fig. 5.2a), the NSTM recovers the motion of the *C. elegans* at each timepoint, giving both a clean reconstruction without motion artifacts and an estimate of the sample dynamics.

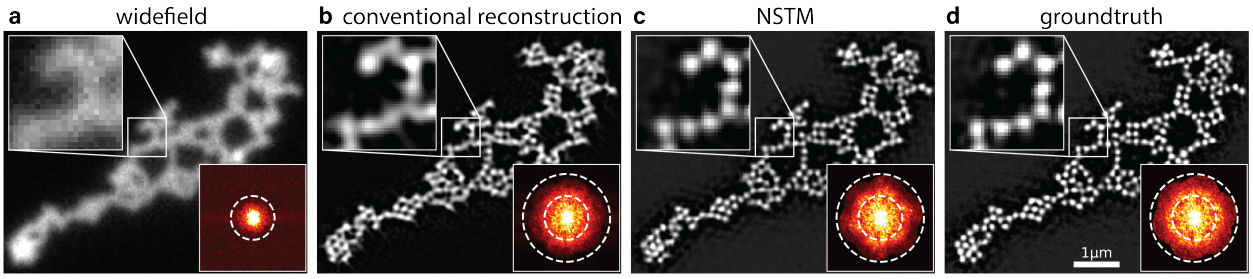


Figure 5.5: Structured illumination microscopy (SIM) of a dense microbead sample with vibrating motion. **a**, The diffraction-limited widefield image cannot resolve individual beads. **b**, The conventional SIM reconstruction algorithm (fairSIM [121]) assumes a static scene, so suffers from motion blur. **c**, Our NSTM reconstruction resolves all of the sub-resolution sized beads and gives a similar quality reconstruction as **d**, the groundtruth case, in which we collected the data without sample motion. Bottom right of each image shows the frequency spectra (with gamma correction power = 0.7).

### Data acquisition

The DPC images were obtained from [84] with a commercial inverted microscope (Nikon TE300) with 10× 0.25NA objective (Nikon) and an effective pixel size of 0.454μm. A LED-array [139] (SCI Microscopy) was attached to the microscope in place of the Köhler illumination unit. Four half circular illumination patterns, with the maximum illumination NA equal to the objective NA, were sequentially displayed on the LED array to capture four raw images as in [175]. Exposure time was 25ms.

## 5.3 3D structured illumination microscopy

Our second multi-shot system is 3D structured illumination microscopy (3D SIM) [68] which captures 15 raw measurements at each $z$ plane (three illumination orientations, five phase shifts for each orientation). The conventional 3D SIM reconstruction assumes there is no motion during the acquisition; thus, it is limited to fixed samples. Previous work in extending 3D SIM to live cells focuses on accelerating the acquisition through faster hardware [204, 185, 53] or assumes translation-only motion [68]. NSTM provides a strategy to recover and account for deformable motion. Because we model motion during the acquisition of a single volume, we can reconstruct both the super-resolved image and the dynamics.

### Implementation

The conventional 3D SIM reconstruction uses five measurements of different sinusoidal phase shifts to separate the complex spectra of three frequency bands and then shifts each band

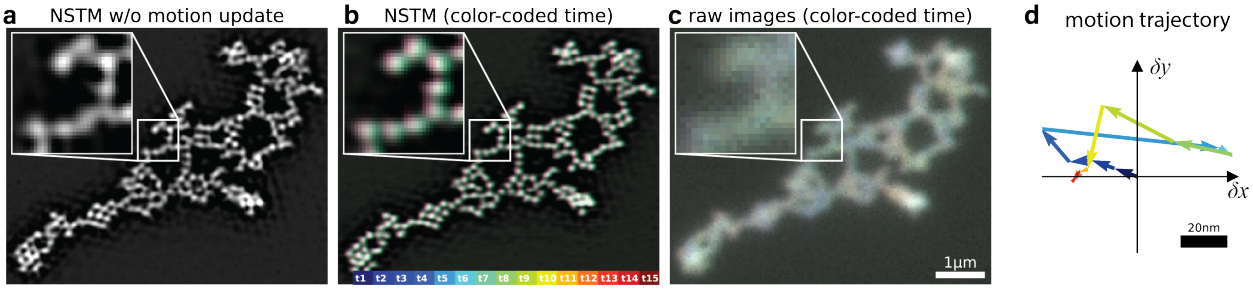


Figure 5.6: Additional results for the dense microbead sample from Fig. 5.5: **a**, Reconstruction using NSTM without the motion update results in motion blurring similar to the conventional reconstruction in Fig. 5.5b, since dynamics are not accounted for. **b**, NSTM reconstruction with color-coded time. **c**, The raw images with color-coded time. In the images with color-coded time, each timepoint of raw images or reconstruction is drawn in a distinct color as indicated by the color bar. The 'color dispersion' in the zoom-in reconstruction suggests that subtle motion is recovered by NSTM. **d**, The recovered motion trajectory of a pixel on the vibrating microbeads from NSTM reconstruction. Each arrow shows the motion displacement vector with respect to the previous timepoint as indicated by the color code (color bar in **b**).

accordingly based on its corresponding modulation frequency. The band separation process necessitates the assumption of a static scene over those five measurements. To avoid this static assumption and preserve the temporal information, we implement the 3D SIM forward model in real space without band separation, rendering each measurement independently from NSTM's reconstruction at the timepoint that the actual measurement is taken.

This forward model can be expressed mathematically as

$$forward_i\left(o\right) = \sum_{j\in\{0,1,2\}} \mathcal{F}_{3D}^{-1}\left[\mathrm{OTF}_j \cdot \mathcal{F}_{3D}\left(illum_{i,j} \cdot o\right)\right], \tag{5.6}$$

where $\mathcal{F}_{3D}$ denotes 3D Fourier transform. The super-resolved 3D fluorescent density, $o$, is first modulated by the corresponding illumination pattern, $illum_{i,j}$, at the $i$th measurement and band $j$. Then, the modulated signal is filtered by the optical transfer function, $\mathrm{OTF}_j$, for each band $j$, and the resulted signals for the three bands are summed to render the $i$th intensity measurement.

In the naive implementation, we need to feed the 3D fluorescent density, $o$, at hundreds of different timepoints[1] to the forward model to render a set of measurements, which is computationally inefficient. To improve the efficiency, we group together measurements with identical orientation and phase captured at different depth planes, and render them in one

[1]For example, a dataset with 20 depth planes has 20 planes $\times$ 3 orientations $\times$ 5 phases = 300 raw images, so 300 distinct timepoints.

forward model pass as if they were acquired at the same timepoint. This simple modification allows us to feed $o$ at only 15 timepoints to get the full set of raw images, regardless of the number of depth planes.

In our comparisons, we use the same illumination parameters estimated from measurements [68, 185] for both conventional reconstruction algorithms and NSTM. For the conventional reconstructions shown in Fig. 5.10b, we use the moving window approach to select a set of raw images around a certain timepoint to feed into the reconstruction algorithm, and we repeat this process to get the conventional reconstruction at every illumination orientation. For example, the conventional reconstruction at timepoint 3 in Fig. 5.10b uses raw images from illumination orientation 2 & 3 from the current acquisition[2] and also the illumination orientation 1 from the next acquisition, where there is no delay between two acquisitions.

## Results

Figure 5.5 shows results for a single-layer dense microbead sample in which we introduced motion by gently pushing and releasing the optical table during the acquisition. Using a conventional reconstruction algorithm (fairSIM [121]) results in a motion blurred image in which the individual beads cannot be resolved. In contrast, our NSTM reconstruction resolves individual beads with a quality comparable to the groundtruth reconstruction. In addition, we also recover the motion map (Fig. 5.6b and d). In this experiment, the groundtruth was reconstructed from a separate set of raw measurements captured without motion (Fig. 5.5d).

Applying this technique to live-cell imaging, Fig. 5.7 and Fig. 5.8 show 3D SIM reconstructions for a live RPE-1 cell expressing StayGold-tagged [73] mitochondrial matrix protein. In Fig. 5.7b, the conventional reconstruction appears to show a mitochondrion with a tubule branch (red arrow); however, our NSTM result recovers the sample dynamics (see Fig. 5.8b) and thus recognizes that it is actually a single tubule which is moving during the acquisition time. This can be further verified by the low-resolution widefield images (Fig. 5.7e) and by running our NSTM algorithm without the motion update (Fig. 5.8c). In addition to resolving motion, NSTM removes motion blur, recovering features that were blurred in the conventional reconstruction (blue arrows in Fig. 5.7b-c), and thus NSTM preserves more high-frequency content compared with conventional reconstructions (Fig. 5.9).

In another 3D SIM experiment, we imaged a live RPE-1 cell expressing StayGold-tagged endoplasmic reticulum (ER) (Fig. 5.10). The conventional reconstruction struggles to resolve clear ER network structures, likely due to their fast dynamics (see red arrows). Additionally, the motion artifacts in the conventional reconstruction are changing over time, making it difficult to visually track different features to see the ER dynamics. NSTM, on the other hand, recovers the motion kernels and the dynamic scene from the same set of raw images for a single volume reconstruction, and the ER structures it resolves are consistent over

[2]We use the term 'acquisition' to refer to 'timepoint' in a regular context of time-series acquisition, since 'timepoint' is already heavily used for time within a single acquisition of a scene.

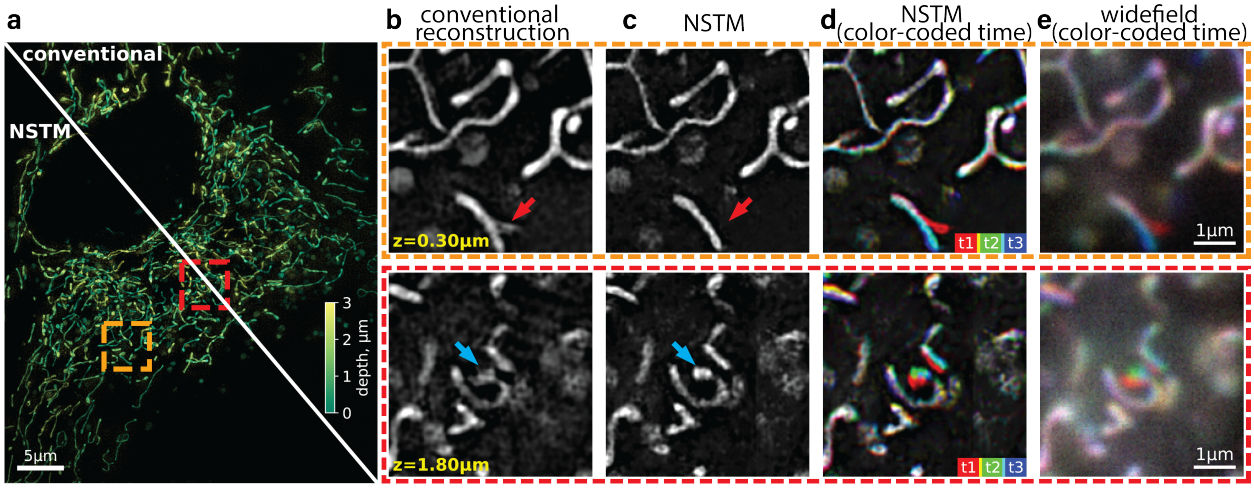


Figure 5.7: 3D SIM reconstruction of a live RPE-1 cell expressing StayGold-tagged mitochondrial matrix protein. **a**, Maximum projection of the volume with color-coded depth. **b-c**, Zoom-in of a slice from the 3D reconstruction, comparing the conventional 3D SIM algorithm (CUDA-accelerated three-beam SIM reconstruction software [68]) with our NSTM algorithm. The NSTM reconstruction disambiguates the artifacts induced by tubular motion (as indicated by the arrows). **d-e**, The NSTM reconstructions and widefield images at three timepoints coded by colors. Widefield images are obtained by summing the raw images from five phase shifts.

time. The recovered motion kernels reveal the dynamics happening at different timepoints within a single 3D SIM acquisition as shown in Fig. 5.10c. We also imaged a live RPE-1 cell tagged with F-Actin Halo-JF585 or MitoTracker Green to show NSTM's capability on dense subcellular structures (Fig. 5.12 and Fig. 5.13).

## Cell line generation

The RPE-1 cell lines used in 3D SIM experiments were cultured using Dulbecco's Modified Eagle Medium/Nutrient Mixture F-12 (Thermo Scientific 11320033) supplemented with 10% FBS (VWR Life Science 100% Mexico Origin 156B19), 2mM L-Glutamine, 100 Units/mL penicillin, and 100mg/mL streptomycin (Fisher Scientific 10378016). Trypsin-EDTA (0.25%) phenol red (Fisher Scientific 25200114) was used to detach cells for passaging. To generate the cell lines, we obtained the pCSII-EF/mt-(n1)StayGold (Addgene plasmid #185823) and pcDNA3/er-(n2)oxStayGold(c4)v2.0 (Addgene plasmid #186296) from Atsushi Miyawaki [73] to tag the mitochondrial matrix and the endoplasmic reticulum, respectively. We obtained the LifeAct-HaloTag from Dorus Gadella (Addgene #176105) to tag F-Actin. The er-(n2)oxStayGold(cr)v2.0, mt-(n1)StayGold, and the LifeAct-HaloTag sequences were PCR amplified and cloned into a lentiviral vector containing an EF1 alpha

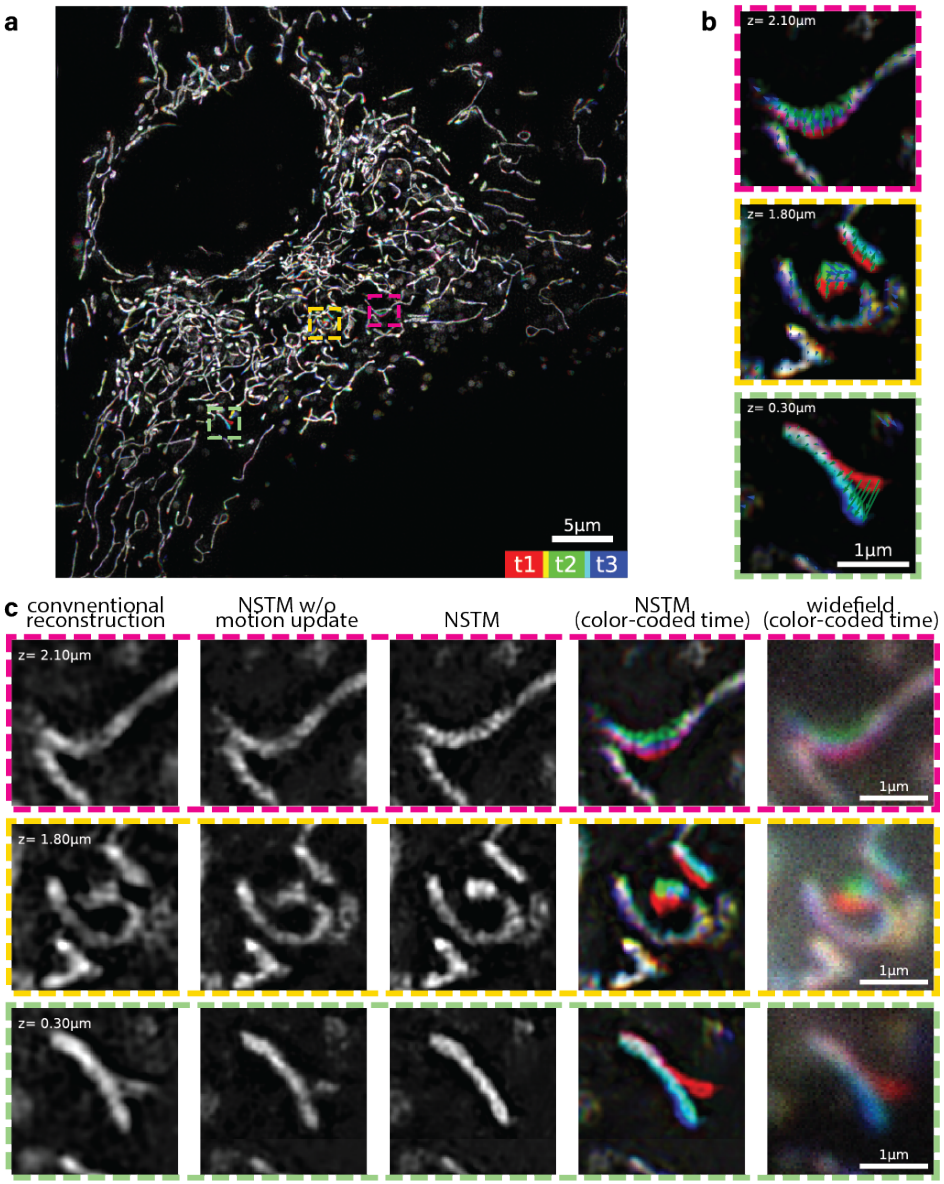


Figure 5.8: Additional 3D SIM results for the mitochondria-labeled RPE-1 cell from Fig. 5.7. **a**, Maximum projection of NSTM reconstruction volume, with three colors denoting the three timepoints that correspond to the three illumination orientations. **b**, Zoom-ins of a slice of NSTM 3D reconstruction, with color-coded time. The overlaid vector fields show the motion displacement recovered by NSTM, with their colors to indicate their corresponding timepoints. **c**, Zoom-in comparisons, from left to right: conventional reconstructions [68], NSTM without motion update, NSTM reconstruction, NSTM reconstruction with color-coded time (three colors for three illumination orientations), and widefield images with color-coded time.

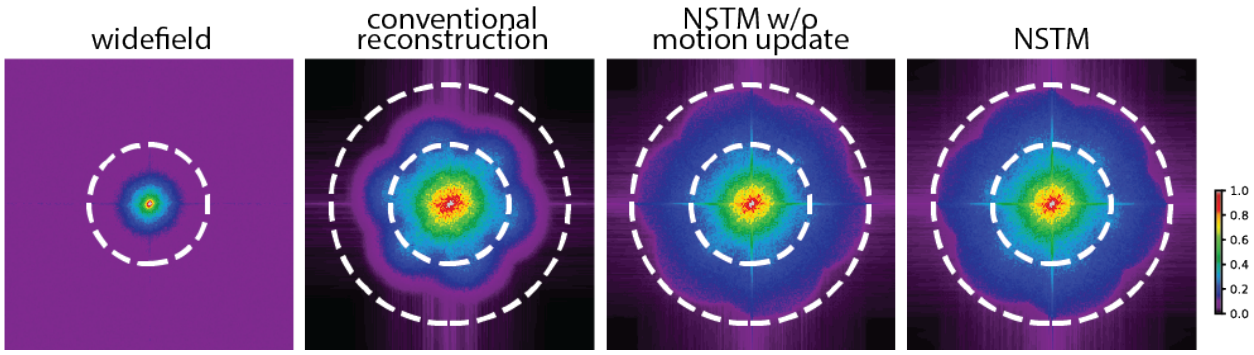


Figure 5.9: A comparison of the spatial frequency spectra for each method. The two dashed circles indicate the diffraction-limited bandwidth and SIM super-resolved bandwidth, respectively. Gamma correction with power of 0.5 is applied to all frequency spectra for better contrast.

promoter. The vector is a derivative of Addgene #60955 with the sgRNA sequence removed. Lentiviral particles containing each plasmid were produced by transfecting standard packaging vectors along with the plasmids into HEK293T cells using TransIT-LT1 Transfection Reagent (Mirus, MIR2306). Media was changed at 24 hours post-transfection, without disturbing the adhered cells, and the viral supernatant was harvested approximately 50 hours post transfection. The supernatant was filtered through 0.45 mm PVDF syringe filter and about 1 mL was used to directly seed a 10 cm plate of hTERT RPE-1 cells (ATCC CRL-4000). Two days post-infection, cells were analyzed on BD FACSAria Fusion Sorter and BDFACSDiva Software. The highest 5% of StayGold/GFP (FITC) fluorescence cells were sorted for the StayGold tagged- ER and mitochondrial matrix lines (gating strategy illustrated in Fig. 5.14). To prepare F-Actin Halo-tagged RPE-1 cells for sorting, Janelia Fluor HaloTag Ligand 503 was diluted at 1:20,000 from a 1mM stock in supplemented DMEM-F12. Then the original media was carefully aspirated off the cells, and replaced with DMEM-F12 media containing the ligand. The ligand and cells were incubated at 37 degree for 15 mins, then washed three times with PBS before trypsinization and subsequent sorting. For the LifeAct-Halo tagged RPE-1 line, the same gating strategy was used as described above for StayGold cells wherein highest 5% of Halo fluorescence cells were sorted (gating strategy illustrated in Fig. 5.15). All sorted cells were expanded for imaging experiments.

## Sample preparation

Janelia Fluor JF585 dye was used to label the F-Actin on the LifeAct-Halo tagged RPE-1 cells prior to imaging. The dense microbead sample was made with 0.19µm dyed microbeads (Bangs Laboratories, FC02F). The stock solution was diluted 1:100 with distilled water and placed on a glass-bottom 35mm dish coated by Poly-L-lysine solution (Sigma Aldrich,

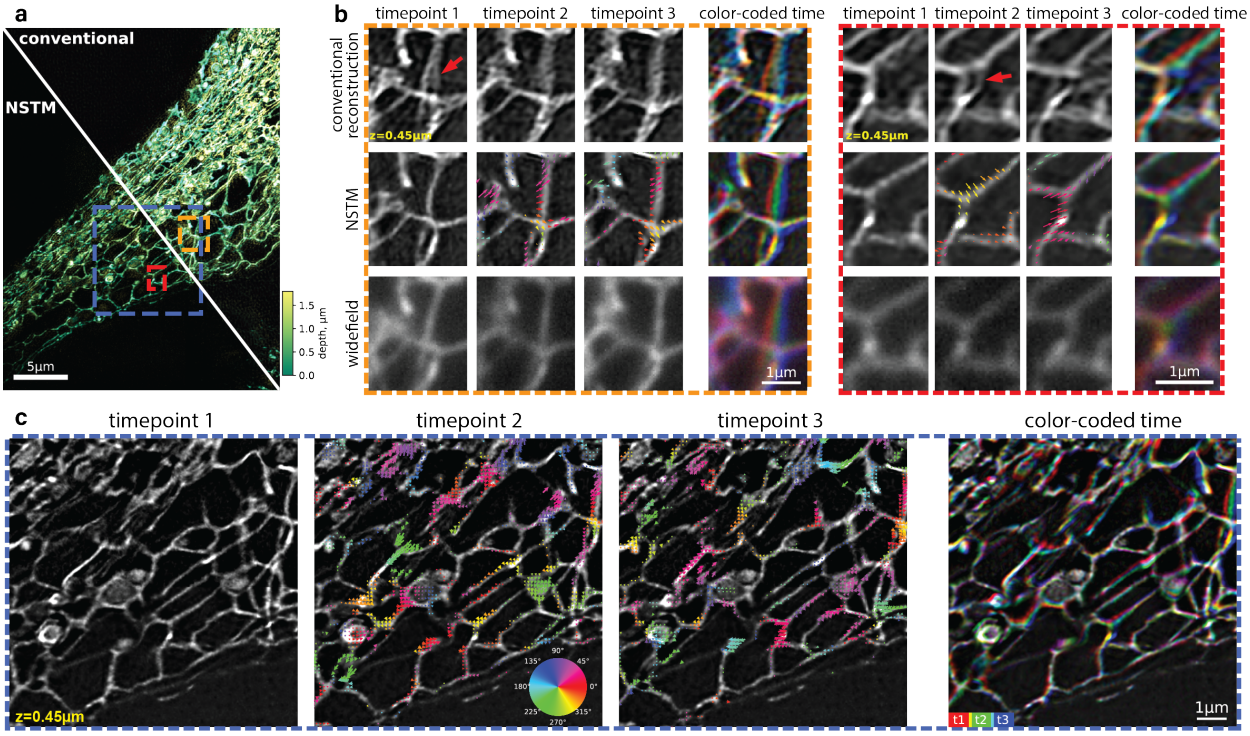


Figure 5.10: 3D SIM experimental results for a RPE-1 cell expressing StayGold-tagged endoplasmic reticulum (ER). **a** Maximum projection of the reconstructed volume with color-coded depth. **b**, Zoom-ins at three timepoints, each of which corresponds to a different illumination orientation, for the widefield, conventional and NSTM reconstructions. A moving window approach (see Methods) is used to compute the conventional reconstruction [68] at different timepoints. The NSTM reconstructions are overlaid with the recovered motion kernels which show the sample's motion displacements from the previous timepoint. The colors of the motion kernel indicate motion directions, according to colorwheel in **c**. **c**, Zoom-in NSTM reconstructions at three timepoints and the combined view with color-coded time. The motion kernels on the second and third timepoints show the structure's motion displacements from the previous timepoint, with color-coding to indicate motion directions.

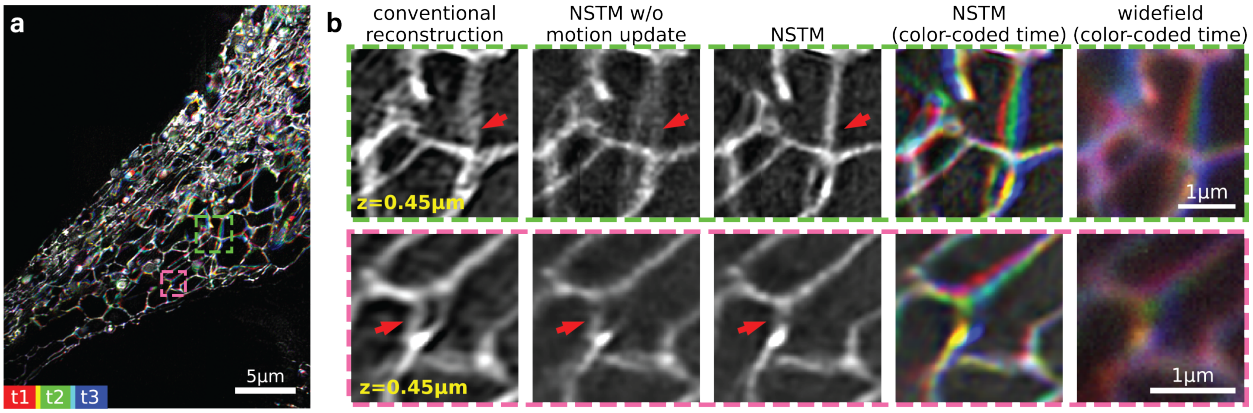


Figure 5.11: Additional 3D SIM results for the live endoplasmic reticulum-labeled RPE-1 cell. **a**, Maximum z-projection of NSTM reconstruction volume, with three colors denoting the three timepoints that correspond to the three illumination orientations. **b**, Zoom-in comparisons, from left to right: conventional reconstructions [68], NSTM without motion update, NSTM reconstruction, NSTM reconstruction with color-coded time (three colors for three illumination orientations), and widefield images with color-coded time.

P8920).

### Data acquisition

The 3D SIM datasets were acquired on a commercial three-beam SIM system (Zeiss Elyra PS.1) using an oil immersion objective (Zeiss, 100× 1.46 NA) and 1.6× tube lens. The effective pixel size was 40.6nm. The system captures 15 images at each depth plane, with 3 illumination orientations and 5 phase shifts for each orientation. A single image plane was acquired for the dense microbead sample. 20 planes with a step size of 150nm were captured for the RPE-1 cell expressing StayGold-tagged mitochondrial matrix protein, LifeAct-Halo tagged RPE-1 cell stained with Janelia Fluor JF585, and 12 planes with a step size of 150nm were captured for the RPE-1 cell expressing StayGold-tagged endoplasmic reticulum. 488 nm laser was used for all but the F-Actin Halo-JF585 tagged cell, for which we used a 561nm laser. The SIM system has a illumination update delay of around 20ms for each phase shift or $z$-position shift, and a delay of 300ms for each illumination orientation change. We set the exposure time to 20ms for the dense microbeads and 5ms for all cell experiments.

## 5.4 Rolling-shutter DiffuserCam lensless imaging

Our third multi-shot computational imaging example is rolling-shutter DiffuserCam [4], a lensless camera that compressively encodes a high-speed video into a single captured image.

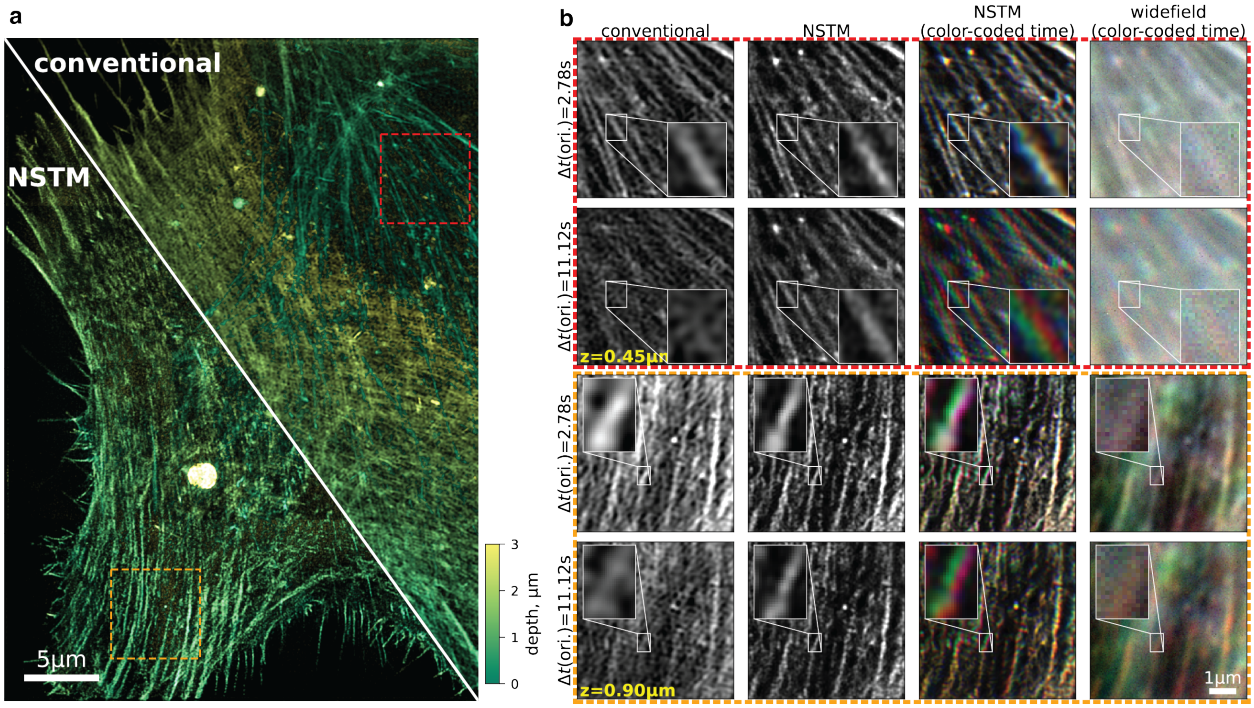


Figure 5.12: 3D SIM reconstruction of a live F-Actin labeled RPE-1 cell. **a**, Maximum z-projection of the reconstructed volume with color-coded depth. **b**, Zoom-in comparisons, from left to right: conventional reconstruction [68], NSTM reconstruction, NSTM reconstruction with color-coded time (three colors for three illumination orientations), and widefield images with color-coded time. The second row of each zoom-in assumes raw images with longer delay between orientations, $\Delta t(\text{ori.})$, and thus more motion (*i.e.*, the raw images of orientation 1 are from acquisition timepoint 1, orientation 2 from acquisition timepoint 2, and orientation 3 from timepoint 3 from a time-series measurement).

This method leverages the fact that each row of the image, captured sequentially by the rolling shutter, contains information about the whole scene at that timepoint, due to the system's large point-spread-function (PSF).

Each row of the raw image captured by rolling-shutter DiffuserCam is the time integral of the dynamic scene convolved with the caustic point-spread-function (PSF) over the rolling shutter exposure. Thus, its forward model can be written in a discrete-time sum of $T$ timepoints [4],

$$forward\left(o\right) = \sum_{t=0}^{T-1} \left(o\left(t\right) * \text{PSF}\right) \cdot S\left(t\right), \tag{5.7}$$

where $o$ is the dynamic scene, $S$ is a binary map of the shutter on/off state, and $*$ denotes 2D convolution operation. However, rendering the entire image at once requires obtaining

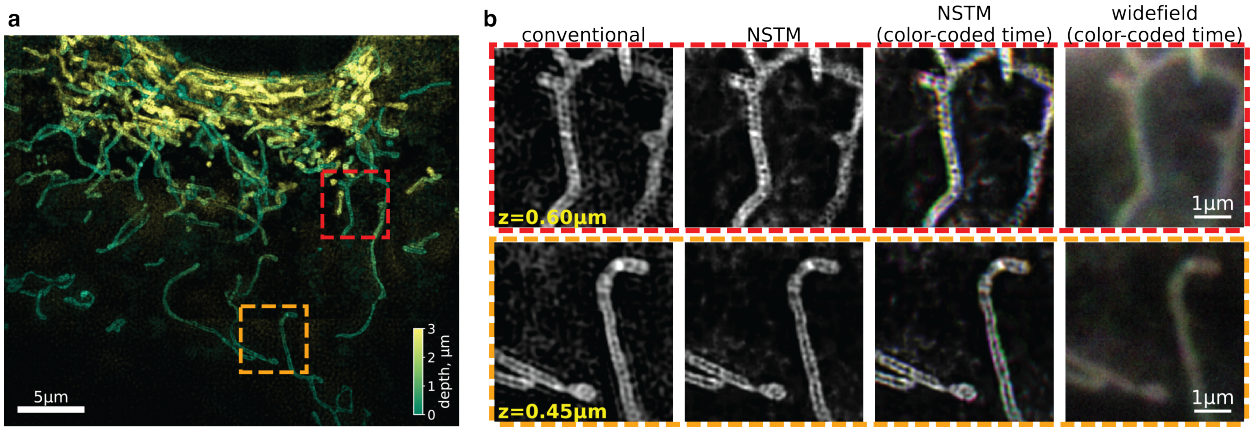


Figure 5.13: 3D SIM reconstruction of a live RPE-1 cell tagged with MitoTracker Green. **a**, Maximum z-projection of the NSTM reconstructed volume with color-coded depth. **b**, Zoom-in comparisons, from left to right: conventional reconstruction [68], NSTM reconstruction, NSTM reconstruction with color-coded time (three colors for three illumination orientations), and widefield images with color-coded time.

NSTM's reconstructed scenes at all timepoints, which will be intensive on GPU memory. To make this feasible on common GPUs, during each step of the reconstruction we render a subset of image rows by only obtaining the reconstructed scenes at timepoints which have contributed signal to these rows. The forward model for the $i$th row of the raw image can be written as

$$forward_i\left(o\right) = \sum_{t\in\{t|S(i,t)=1\}} \left(o\left(t\right) * \mathrm{PSF}\right) \cdot S\left(t\right). \tag{5.8}$$

In practice, to improve the efficiency, we render 20 consecutive rows in each forward pass.

To enable video reconstruction from the single raw image, the original algorithm [4] uses total variation (TV) regularization to promote smoothness. In contrast, by modeling for the motion explicitly, NSTM produces cleaner reconstructions without over-smoothing (Fig. 5.16b). As a byproduct of NSTM, the motion trajectory for any point can be queried directly from the motion network (Fig. 5.16c).

## Data acquisition

The rolling shutter DiffuserCam data is from the original work on the technique [4]. The raw image was taken by a color sCMOS (PCO Edge 5.5) in slow-scan rolling shutter mode (27.52μs readout time for each row) with dual shutter readout and 1320μs exposure time. The acquisition of the raw image took 31.0ms.

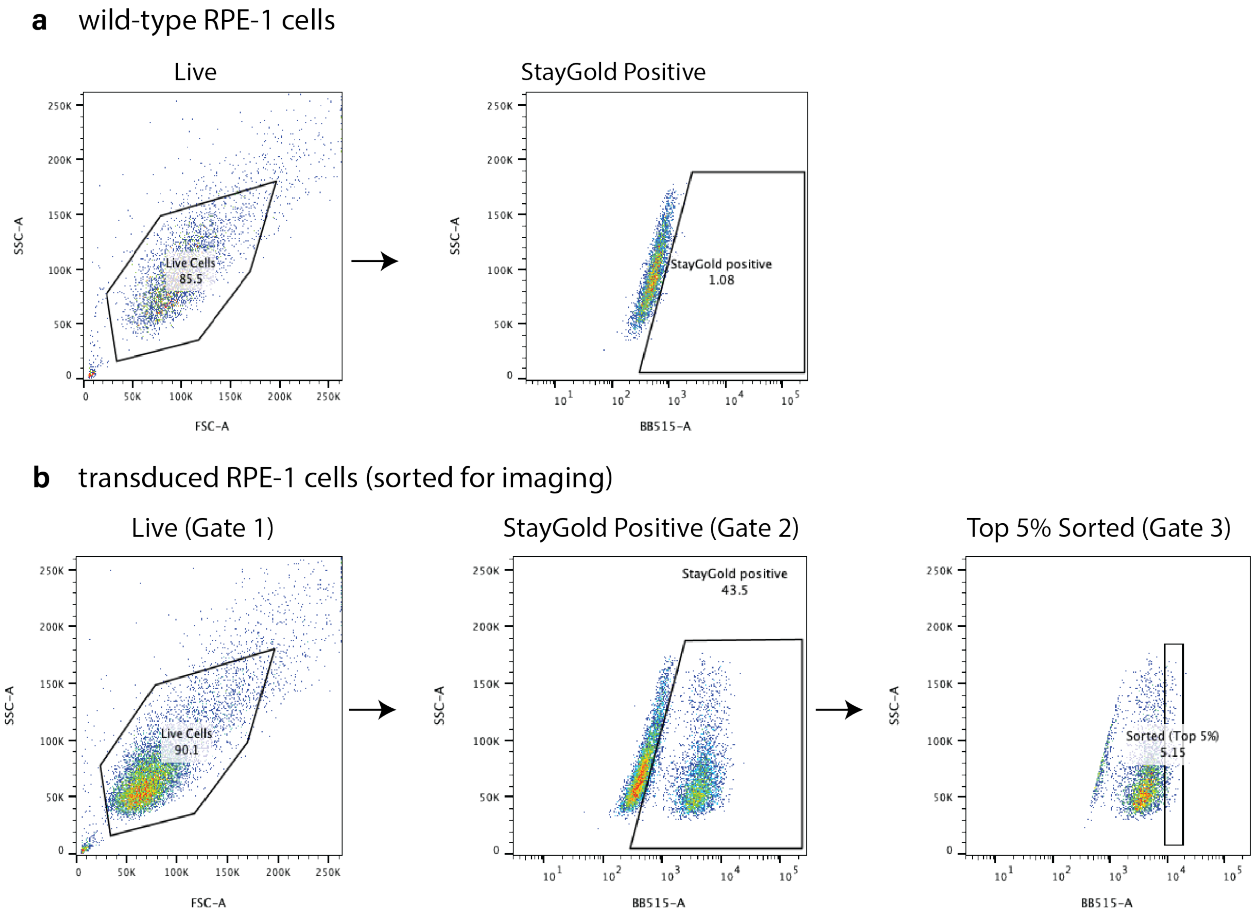


Figure 5.14: Gating strategy for sorting the StayGold tagged- ER and mitochondrial matrix lines. **a**, Wild-type RPE-1 cells were used to gate for Live Cells (Gate 1) and the StayGold negative cells were used to gate for the StayGold positive population (Gate 2). **b**, To sort samples that were transduced with StayGold expressing plasmids, Gate 1 (Live cells) was applied followed by Gate 2 (StayGold positive), and then top 5% of the StayGold positive cell population (Gate 3) was sorted using BDFACS Aria Fusion Sorter and expanded using DMEM-F12 for subsequent imaging experiments.

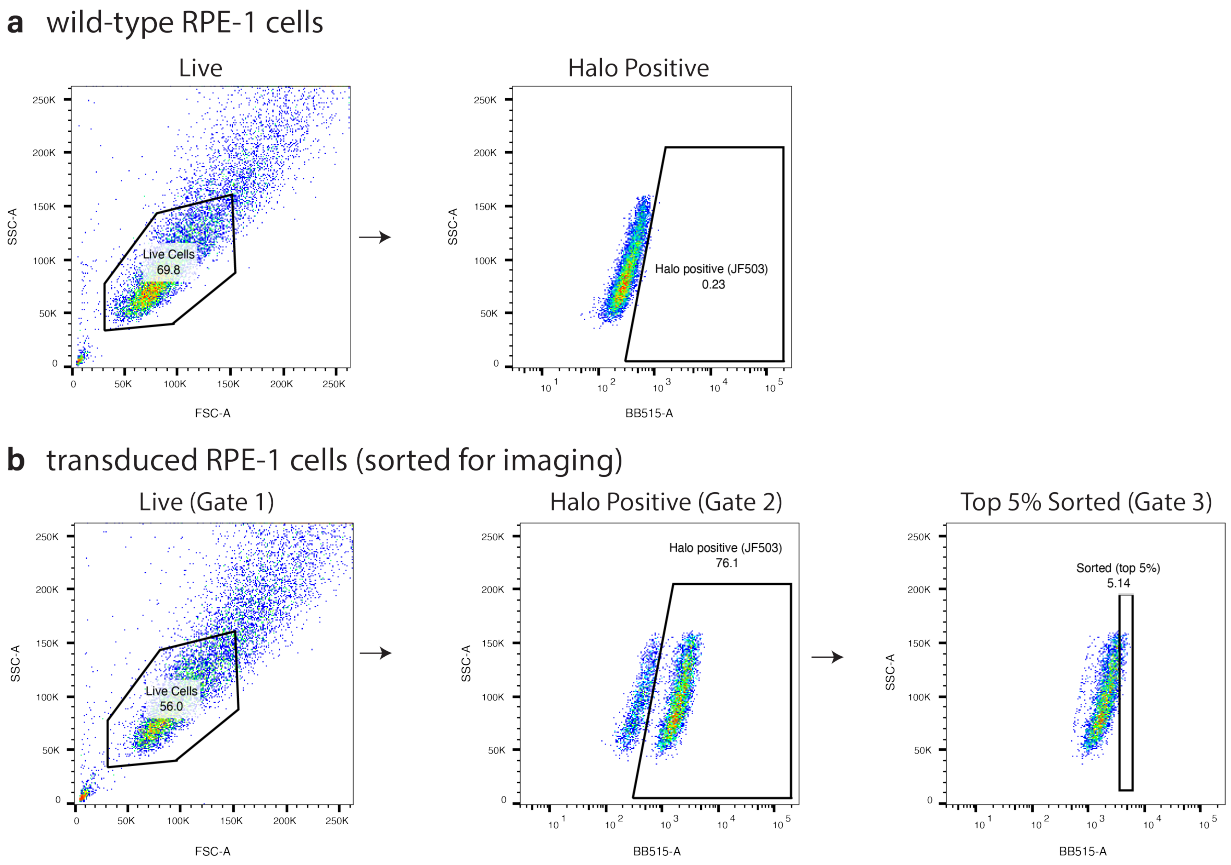


Figure 5.15: Gating strategy for sorting the LifeAct-Halo tagged RPE-1 line. **a**, Wild-type RPE-1 cells were used to gate for Live Cells (Gate 1) and the Halo negative cells were used to gate for the Halo positive population (Gate 2). **b**, To sort samples that were transduced with Halo expressing plasmids, Gate 1 (Live cells) was applied followed by Gate 2 (Halo positive), and then top 5% of the Halo positive cell population (Gate 3) was sorted using BDFACS Aria Fusion Sorter and expanded using DMEM-F12 for subsequent imaging experiments.

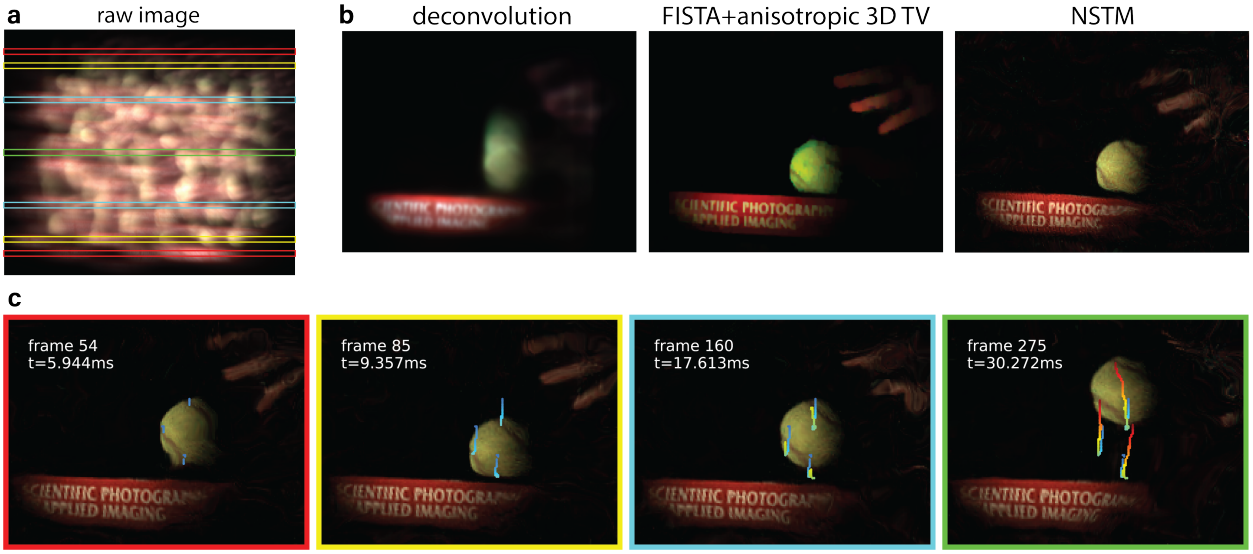


Figure 5.16: Results for rolling-shutter DiffuserCam. **a**, The raw image measurement. **b**, Comparisons of the reconstruction using basic deconvolution (assumes a static scene), FISTA with anisotropic 3D Total Variation regularization (TV) [4] (the original reconstruction method), and our NSTM algorithm. **c**, NSTM reconstruction at different timepoints, with their corresponding measurement rows indicated by colored boxes on the raw image. The colored curves show some selected motion trajectories recovered by the motion network.

## 5.5 Discussion

We demonstrated our neural space-time model (NSTM) for recovering motion dynamics and removing motion-induced artifacts in three different multi-shot imaging systems; however, the models are general and should find use in other multi-shot computational imaging methods. Notably, NSTM does not use any data priors or pre-training, such that the network weights are trained from scratch for each set of raw measurements. Hence, it is compatible with any multi-shot system with a differentiable and deterministic forward model. For multi-shot imaging systems like 3D SIM, which do not use gradient-based reconstruction, we can alternatively implement a forward model as part of the NSTM reconstruction as discussed in Section 5.3.

While NSTM is a powerful technique to resolve dynamic scenes from multiple raw images, it relies on temporal redundancy, *i.e., the smoothness of motion and correlatable scenes over adjacent timepoints*, to jointly recover the motion and the scene. As a consequence, this strategy tends to degrade or fail when the motion is less smooth. To demonstrate some failure modes, we provide several simulation examples. First, we simulate different amounts (magnitudes) of motion, showing that NSTM does well with large magnitudes of rigid-body or linear motion, presumably due to the effectiveness of coarse-to-fine process, but begins to

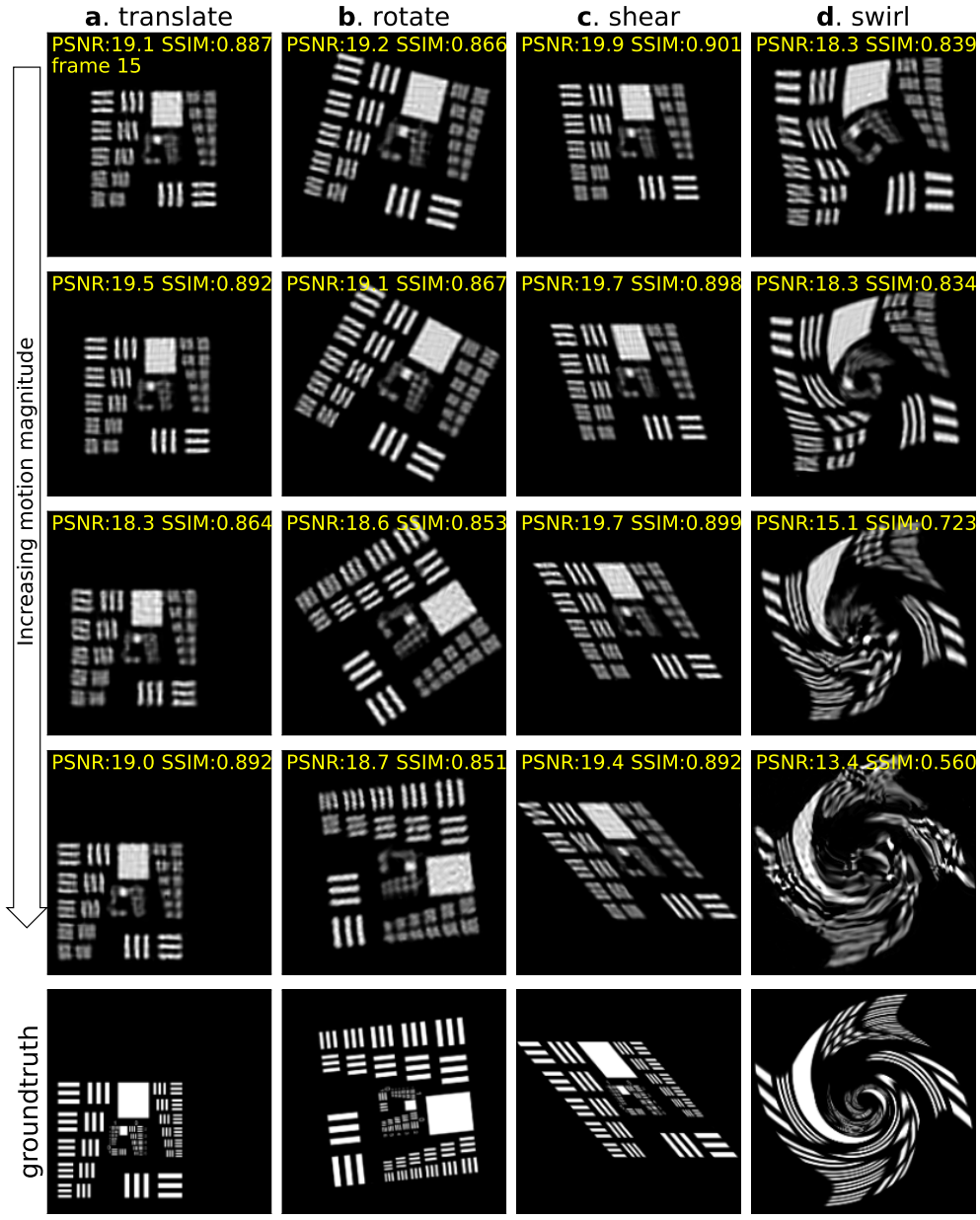


Figure 5.17: SIM simulations with various types and magnitudes of motion. From left to right: **a**, rigid motion - translation, **b**, rigid motion - rotation, **c**, non-rigid global motion - shearing, and **d**, local deformable motion - swirl. The first four rows show the NSTM reconstructions from simulated images with increasing magnitude of motion between frames, and the last row shows the groundtruth scenes. The reconstruction of local deformable motion is more likely to fail when the motion magnitude increases.

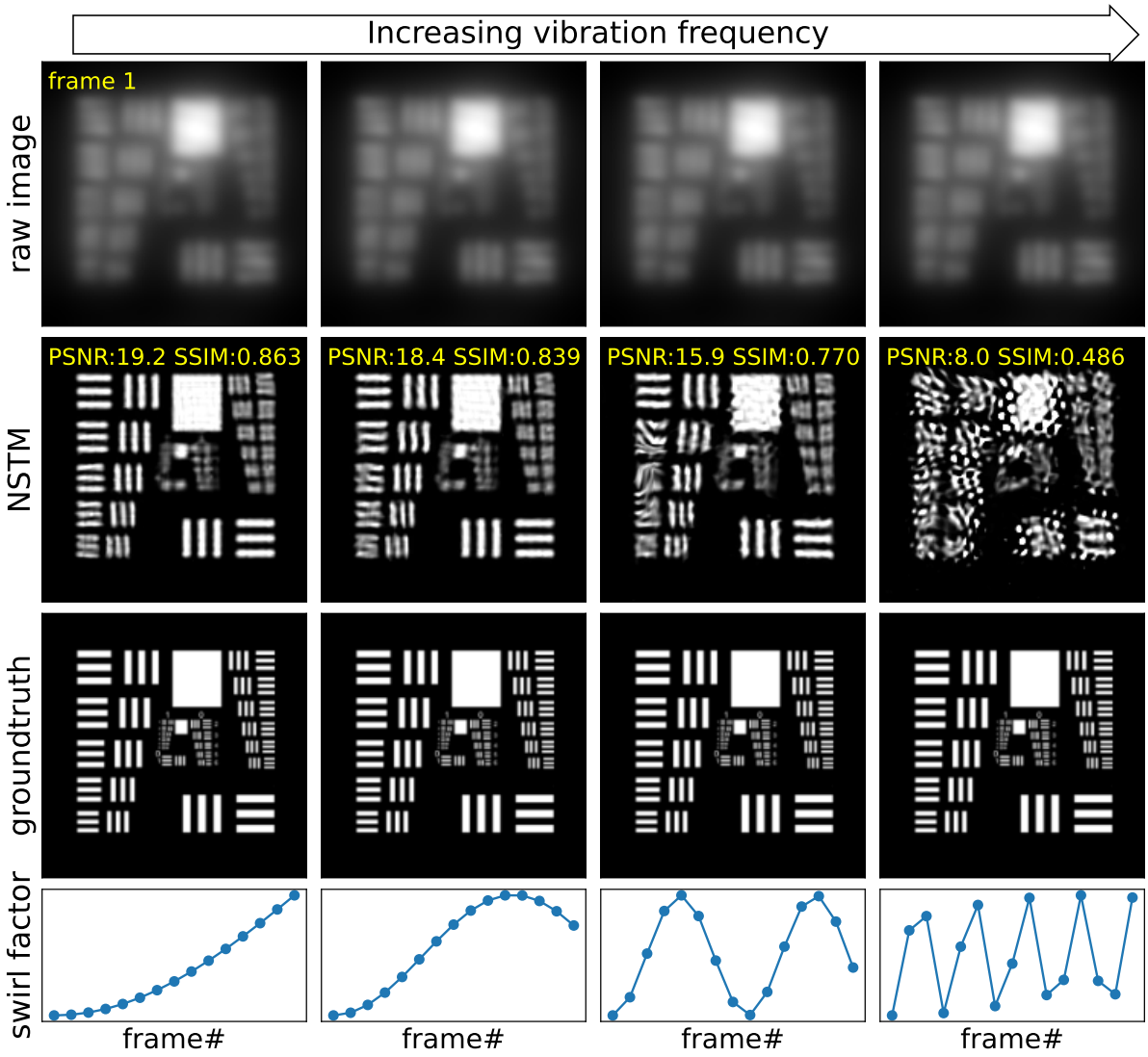


Figure 5.18: Simulations of SIM with local deformable vibration motion. The deformable swirl motion for each frame is generated using the swirl factor shown in the last row. The frequency of the swirl factor increases from left to right. As the frequency increases, there will be less temporal redundancy between adjacent frames, and hence NSTM will be more likely to fail.

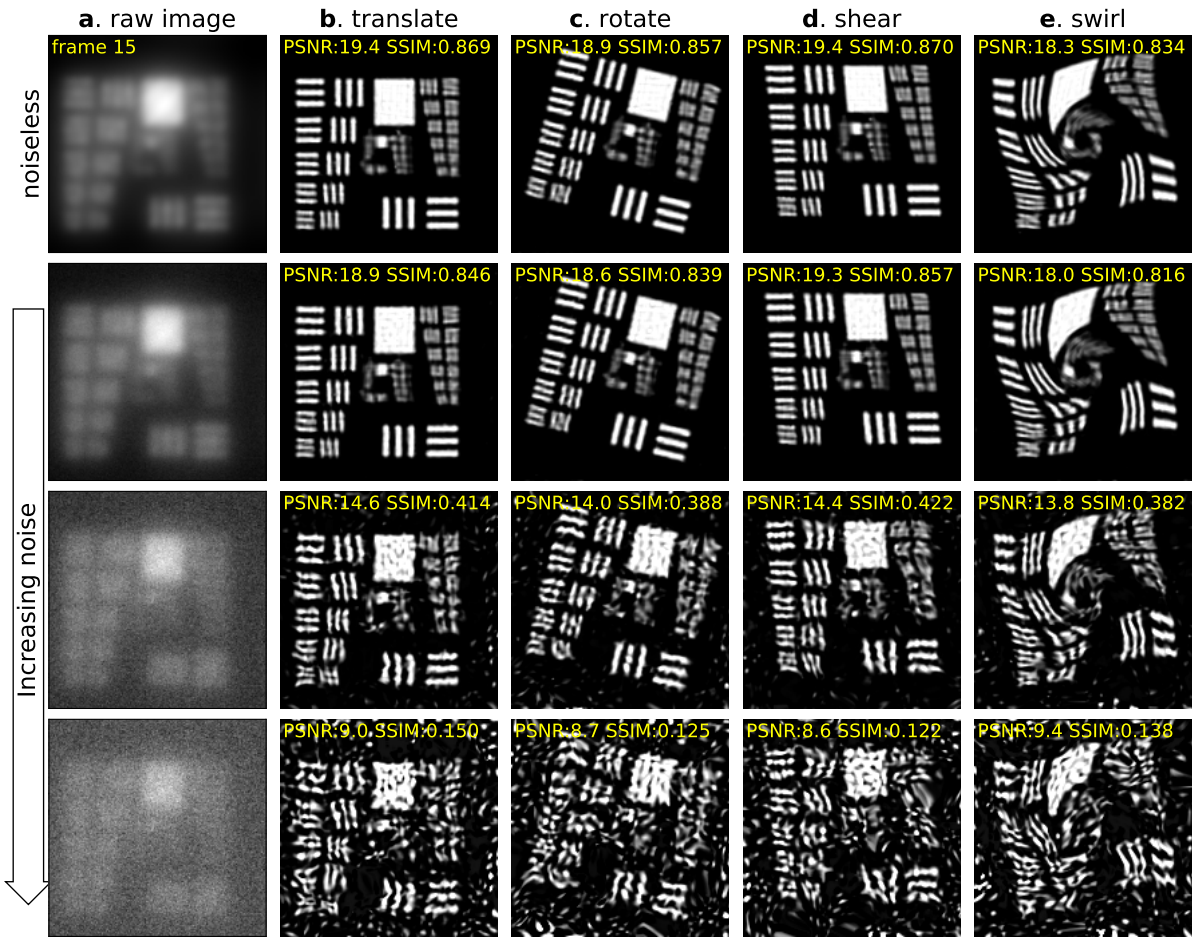


Figure 5.19: Simulations of SIM with increasing amounts of additive Gaussian noise. **a**, The simulated raw image. **b**-**e**, Various types of motion: **b**, rigid motion - translation, **c**, rigid motion - rotation, **d**, non-rigid global motion - shearing, and **e**, local deformable motion - swirl. NSTM reconstruction degrades as the noise gets stronger for all types of motion.

degrade with large magnitudes of local deformable motion (Fig. 5.17). Second, we simulate periodic local deformable motion with different vibration frequencies (Fig. 5.18). We find that since NSTM does not explicitly account for periodic motion, it cannot capture high-frequency vibrations when the motion is no longer smooth between adjacent frames. Third, we simulate additive Gaussian noise to the raw measurements (Fig. 5.19) to show how noise degrades the NSTM reconstruction.

One limitation of our method is that its two-network construction cannot accommodate for certain dynamics. Despite that this construction allows an explicit motion model and ensures reconstruction fidelity, it also introduces an additional constraint: since the scene network does not depend on the temporal coordinate, any frame of a dynamic scene has to be obtained by deforming a static reconstruction (from the scene network) with a motion kernel (from the motion network). As a result, NSTM is unable to recover dynamic scenes with appearing/disappearing features or switching on/off dynamics (such as neuron firing, or fluorescence photoactivation), which cannot be reproduced by a time-independent scene network. To overcome this limit, future work could modify the NSTM architecture to account for the different types of non-smooth dynamics and/or incorporate the time-dependency to the scene network.

Another limitation is that our NSTM reconstructions generally require more computation than conventional methods. For example, the dense microbead reconstruction using NSTM took about three minutes on a NVIDIA RTX 3090 GPU, in contrast to the conventional algorithm (fairSIM) which completed in less than 10 seconds on a CPU. The live cell 3D reconstructions (volume size $20 \times 512 \times 512$ with 15 timepoints) using NSTM took 40.5 minutes on a NVIDIA A6000 GPU (Table 5.5. Future work could improve the computational efficiency of NSTM by better initialization of network weights [173], hyper-parameter search for a faster convergence [206], using lower precision arithmetic, and data-driven methods to optimize a part of the model in a single pass [178].

One interesting advantage of using coordinate-based neural networks like NSTM is that it can accommodate arbitrary coordinates that may not be on a rectilinear grid. This is especially advantageous for modeling spatiotemporal relationships, as it can intuitively handle sub-pixel motion shifts and non-uniformly sampled measurements in both space and time, without requiring interpolation of a uniformly sampled matrix. For example, one can output a temporally interpolated video with any desired temporal resolution simply by querying the network at intermediate timepoints between actual measurement timepoints to render the corresponding frames. The resulting reconstructions are clean (no motion blur) and can faithfully represent the scene at those timepoints, provided that the dynamics are accurately modeled by the NSTM. We should not, however, expect to recover any dynamics happening at timescales faster than which can be learned from the measurements.

In summary, we showed that our NSTM method can recover motion dynamics and thus resolve motion artifacts in multi-shot computational imaging systems, using only the typical datasets used for conventional reconstructions. The ability to recover dynamic samples within a single multi-shot acquisition seems particularly promising for observing subcellular systems in live biological samples. By accounting for motion through NSTM's joint recon-

| Imaging system, reconstruction dimension, number of epochs | 1× NVIDIA RTX 3090 (24GB GPU RAM), Intel Xeon Gold 6226R | 1× NVIDIA A6000 (48GB GPU RAM), Intel Xeon Gold 6444Y | 1× NVIDIA A100 (80GB GPU RAM), Intel Xeon Gold 6144 |
|---|---|---|---|
| DPC (Fig. 5.2a) $320 \times 1000$, 4 timepoints 5000 epochs | 6.04 minutes | 4.40 minutes | 4.35 minutes |
| SIM (Fig. 5.5) $320 \times 320$, 15 timepoints 2000 epochs | 3.64 minutes | 2.71 minutes | 3.70 minutes |
| 3D SIM (Fig. 5.7) $20 \times 512 \times 512$, 15 timepoints 500 epochs | insufficient GPU RAM | 40.5 minutes | 43.2 minutes |
| Rolling-shutter DiffuserCam (Fig. 5.16) $540 \times 640$, 280 timepoints 200 epochs | 94.1 minutes | 81.7 minutes | 73.8 minutes |

Table 5.1: The runtime of NSTM reconstructions under different GPU models. The CPU model is also listed as a reference. All computations were performed using single-precision arithmetic and JAX library [16]. While this serves as a reference for the processing speed of NSTM, the actual runtime also varies based on other computer configurations (*e.g.* NVIDIA driver and CUDA software versions, CPU and RAM speed, computer I/O speed, etc.).

struction, NSTM reduces the risk of misinterpretations in the study of living systems caused by motion artifacts in multi-shot acquisitions. Further, it effectively increases the temporal resolution of the system when multi-shot data is captured.

# Chapter 6

# Noise2Image: Noise-Enabled Static Scene Recovery for Event Cameras

While event cameras can operate beyond conventional framerates, they are designed to be blind to the stationary components of a scene, which induce no brightness changes over time. This issue is especially prevalent when the camera is not moving and therefore provides no information about the static background. Even though event cameras are not designed to capture an intensity image, it is often needed for downstream applications such as initializing motion tracking algorithms [3]. To mitigate this issue, some event cameras include a conventional frame-based sensor in the pixel circuit to simultaneously image both events and traditional intensity images [18, 11]. Alternatively, when circuit-level modification of the event camera is not possible, a frame-based camera can be installed in parallel, using either a beam splitter [72, 216, 215] or additional view registration [189]. Both of these solutions introduce additional hardware, increasing cost, complexity, size, and power consumption.

Our work leverages the fact that, even when the scene is static, event cameras still produce noise events. We focus on low and moderate-brightness regimes (*e.g.*room light, outdoor sunset) [65, 64, 63] in which the dominant source of noise is photon noise — random fluctuations in the photon arrival process. In contrast, the high-brightness regime (*e.g.*outdoor daylight) includes leakage noise events [129], which we do not model here. While the photon noise is well-studied in the context of frame-based scenes, those events triggered by photon noise are commonly deemed as part of the general background noise activity of the event-based sensor, to be filtered out.

In this work we propose a method called Noise2Image that can reconstruct a static scene from its event noise statistics, with no hardware modifications and negligible computational overhead. First, we derive a statistical noise model describing how noise event generation correlates with scene intensity, which shows a good correspondence with our experimental measurements. Unlike in conventional sensors, where photon noise grows with the signal, we find that for event cameras, the number of events triggered by photon noise is mostly

This chapter covers the research I presented or published in [27].

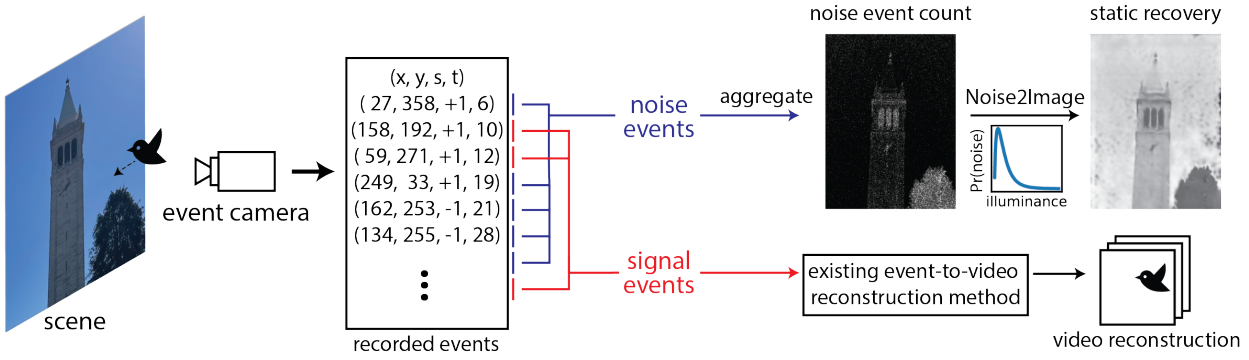


Figure 6.1: Schematic of the Noise2Image pipeline. Recorded events are first separated into noise events and signal events using an existing event denoiser. Signal events are triggered by intensity changes of the scene, which can be fed into existing event-to-video reconstruction methods. The noise events are then used to reconstruct the static scene intensity with our Noise2Image method. This relies on characterizing the relationship between noise events and using learned priors to resolve ambiguities. Data in this figure was captured in an outdoor environment late afternoon.

negatively correlated with the illuminance level due to the logarithmic sensitivity of the sensor. Imaging the static scene then amounts to inverting this intensity-to-noise process. However, the mapping is one-to-many, so not directly invertible; thus, we rely on a learned prior to resolve ambiguities. To train and validate our method, we experimentally collect a dataset of event recordings on static scenes. We demonstrate that Noise2Image can recover photo-realistic images from noise events alone, or can be used to recover the static parts of scenes with dynamics. In addition to testing on in-distribution data, we demonstrate the robustness of this approach with out-of-distribution testing data and live scenes (Fig. 6.1).

Our contributions are summarized as follows:

- We characterize noise event generation due to photon noise and derive a mathematical model describing the statistical relationship between noise events and pixel illuminance.
- We propose the Noise2Image method to recover the intensity image of a static scene from a recording of noise events using a learned prior.
- We collect a noise-events-to-image (NE2I) dataset with recordings of noise events paired with the corresponding intensity images to train and validate our method.
- We show that Noise2Image is complementary to event-to-video reconstruction methods (E2VID), enabling recovery of both static and dynamic parts of a scene.

## 6.1 Related Work

**Event-to-video reconstruction.**

Event-to-video reconstruction (E2VID) is a class of methods that recover high frame-rate video from event recordings. These methods can capture high-speed dynamics without the motion blur that plagues traditional frame-based cameras. Because event cameras only capture scene changes, many E2VID approaches include one or two traditional frame-based images, using events to deblur [152, 82, 132, 186], synthesize adjacent frames [188], or interpolate temporally [179, 180, 210, 190].

In contrast, a more challenging class of E2VID reconstruction only uses events for video reconstruction. Because the initial scene intensity is unknown and only the relative intensity changes are measured, reconstruction requires either explicit modeling of spatiotemporal relationships [7] or deep neural networks as a data prior to fill in the missing information [148, 165, 153, 193, 20, 133, 49]. Because it is difficult to collect event recordings paired with the corresponding frame-based videos at scale, the data prior is obtained through synthetic data generation [148, 165] using an event camera simulator [147]. While E2VID methods are effective for recordings with object motion or camera motion, the reconstruction of static scenes or regions is still out of reach since no events are thought to be triggered without motion. Our Noise2Image method can be considered complementary to E2VID; one might use Noise2Image to recover the static parts of the scene and E2VID to recover the dynamic parts.

**Event camera noise characterization.**

An event recording often contains a number of events not associated with intensity changes, which are termed noise events or background activity [38, 64]. Noise events are attributed to two main sources: photon noise and leakage current [129, 63, 65]. Photon noise is the dominant source of noise in low-brightness conditions [101, 63], while leakage noise events dominates in high-brightness conditions [129, 63]. Although the noise model of event cameras is less studied than traditional sensors, in the lower-light regime, it is generally believed that noise events become less likely to trigger as intensity increases [42, 77]. In event camera simulators, the events triggered by photon noise are modeled as a Poisson process, with the noise event rate linearly decreasing with intensity [77]. The intensity dependency of noise events has been experimentally measured in [63], and shows a non-monotonic relationship with intensity in low light. Our experimental results are consistent with this trend, and we derive a theoretical noise model which explains the relationship between intensity and noise events.

**Event denoising.**

While denoising methods for CMOS or CCD sensors often focus on building an accurate noise model, event camera denoising methods instead emphasize noise detection by iden-

tifying the "signal" events corresponding to changes in the scene and removing everything else. Because natural scene changes are inherently spatiotemporal, an event triggered by real signal should be accompanied by a number of other events at the neighboring spatial and temporal locations. With this observation, a background activity filter (BAF) (also called nearest neighbor filter) is used to identify real events by checking the time difference between a new event and the most recent previous event in its proximate spatial location and rejecting events when the time difference passes a threshold [39, 37, 131]. Similarly, the noise events can also be found through spatiotemporal local plane fitting [10]. BAF is simple to compute and can be implemented in hardware with limited computational resources [103, 8]. Guo and Delbruck further extended BAF by reducing its memory footprint and incorporating structural information from a data-driven signal-versus-noise classifier [65]. A recent study points out that events are often triggered simultaneously by noise and signal and therefore treats denoising as a regression problem to predict the noise likelihood [5]. Here, we assume access to a good event denoiser [51, 42] that isolates noise events resulting from static components of a scene.

**Event camera datasets.**

Unlike frame-based cameras, the collection and curation of event camera datasets is challenging, largely due to the inaccessible hardware and the scarcity of online data repositories (*e.g.*, flickr, Google Images for framed-based images). Nevertheless, since large datasets are critical for machine learning [92, 43], there are a handful of recent efforts to systematically generate large-scale event camera datasets. One common approach is to display images on a monitor and record them using an event camera [154, 130, 97, 87]. The motion in the scene can then be generated by either moving the displayed image on the monitor [154, 97] or moving the camera [130, 87]. Alternatively, event datasets can also be generated from frame-based video datasets by over-sampling in time and feeding into an event camera simulator [59, 77, 214]. In this work, we generate an NE2I dataset using experimental measurements and synthetic noise, described in Sec. 6.2.

## 6.2 Modeling Noise Event Statistics

First, we develop a model of noise event statistics for later use in our synthetic dataset generation and image reconstruction steps. An event $e$ is triggered when a pixel $(x, y)$ detects a change in logarithmic intensity, $\log(I)$, greater than the contrast threshold, $\epsilon$, at the time $t$. The polarity $s = +1$ if the change is positive, and $s = -1$ if the change is negative. The triggering condition for an event can be written as

$$(\log(I(x, y, t)) - \log(I(x, y, t_0))) \cdot s > \epsilon, \tag{6.1}$$

where $t_0$ is the timestamp for the most recent event at the same pixel. The logarithmic sensitivity ensures that this triggering condition is adaptive to the brightness level, *e.g.*a

tenfold increase from 1 lux to 10 lux would result in a similar response as a tenfold increase from 100 lux to 1000 lux. As pixels operate asynchronously, we will proceed by considering events at an arbitrary pixel and drop the $(x, y)$ notation for simplicity.

By design, there should be no events triggered by a static scene, because there will be no changes in intensity. However, the inherent randomness in the photon arrival process means that there will be fluctuations in the detected intensity and thus noise events triggered. Consider a static scene in which a pixel sees $n$ photons over a short period of time. $n$ follows a Poisson distribution with the average photon count $\lambda \propto I$, which can be further approximated as a Gaussian distribution for the light levels we work with ($\lambda > 10$). We similarly write the photon count during the last event trigger as $n_0$, and both $n, n_0 \overset{\text{i.i.d.}}{\sim} \mathcal{N}(\lambda, \lambda)$.

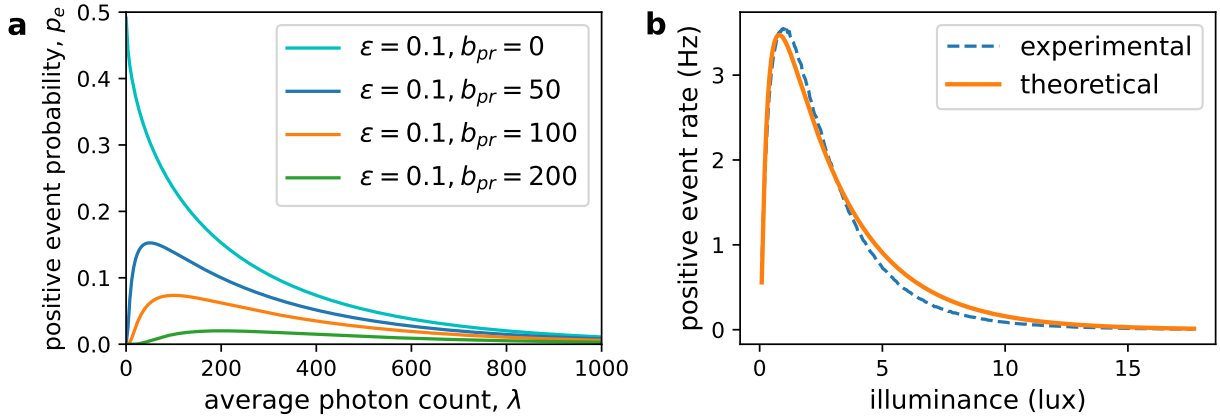


Figure 6.2: **a**. Theoretical noise event probability, $p_e$, versus the average photon count, $\lambda$, at different photoreceptor bias values, $b_{pr}$ ($\epsilon$ is the contrast threshold). **b**. Experimentally measured noise event rate versus illuminance (proportional to $\lambda$) matches well with our theoretical model after fitting parameters $\epsilon, b_{pr}$ and $N$.

To trigger a positive event, we need $\log(n) - \log(n_0) > \epsilon$, which can be re-written as $n - n_0 e^{\epsilon} > 0$. Under the Gaussian approximation, $n - n_0 e^{\epsilon} \sim \mathcal{N}(\lambda(1 - e^{\epsilon}), \lambda(1 + e^{\epsilon}))$. Thus, we can write the probability of triggering a positive event,

$$Pr(n - n_0 e^{\epsilon} > 0) = \frac{1}{2} - \frac{1}{2} erf\left(\frac{\lambda(e^{\epsilon} - 1)}{\sqrt{2\lambda(1 + e^{\epsilon})}}\right), \tag{6.2}$$

where *erf* denotes the error function. The derivation for negative events is similar. This function is plotted in Fig. 6.2a (cyan curve) and reveals a monotonically decreasing trend between noise event probability and illuminance. In other words, as the illuminance increases, it is less likely to find $n, n_0$ that satisfies the triggering condition. This trend matches with previous literature [42, 63, 64].

For low-light conditions, where $\lambda$ is relatively small, the relative signal fluctuation gets stronger, which can lead to more noise events. Consequently, the analog circuit of an event pixel is designed to have a photoreceptor bias voltage and source-follower buffer which help stabilize the photoreceptor's reading and filter out fluctuations beyond a certain bandwidth [38, 64]. To account for this in our model, we approximate the filtering effect by adding a photoreceptor bias term, $b_{pr}$, to the photon count before the logarithmic operation. As a result, the triggering condition of a positive event becomes $\log(n+b_{pr})-\log(n_0+b_{pr}) > \epsilon$, which can be re-written as $n + b_{pr} - e^{\epsilon}(n_0 + b_{pr}) > 0$. We can similarly write the probability of triggering an event, $p_e$, as a function of the average photon count (proportional to the illuminance):

$$p_e(\lambda) = \frac{1}{2} - \frac{1}{2} erf\left(\frac{(\lambda + b_{pr})(e^{\epsilon} - 1)}{\sqrt{2\lambda(1 + e^{\epsilon})}}\right). \tag{6.3}$$

With the bias term, the model shows fewer noise events at lower illuminance levels (Fig. 6.2a). As average photon count increases, the number of noise events will increase up to a point and then decrease. While this model is not monotonic, we will show that we can still recover illuminance from noise event counts.

To validate our derivation, we acquired experimental measurements of noise events at different illuminance levels (Fig. 6.2b). The measured positive event rate matches to our derived noise model after parameter fitting. Note that our formulation is a simplified noise model for the event circuit, which characterizes the events triggered by photon noise. There are more controllable bias parameters in the actual analog circuit [38, 64] that are beyond our formulation.

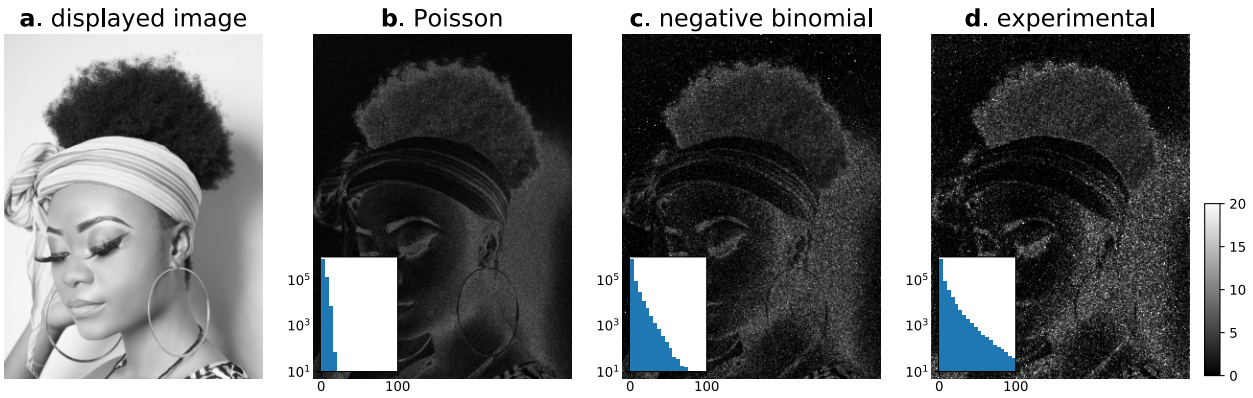


Figure 6.3: **a**. Displayed static scene. **b-c**. Synthetic noise event count sampled from the Poisson distribution and generalized negative binomial distribution, respectively. **d**. Noise event count from a experimental recording. The insets show the histogram for the noise event count.

**Sampling Noise Events.**

Once we know the probability of observing a single noise event, we can simulate the number of noise events over a fixed time window given the intensity of a scene, $I$. Viewing each potential noise event in this window as a Bernoulli trial with probability $p_e$, the noise event count has a binomial distribution with probability of $p_e(I)$ and $N$ trials. With the refractory period (*i.e.*minimum time between events) much smaller than the inter-event interval, $N$ will be large, and the binomial distribution can be approximated by a Poisson distribution with parameter $N \cdot p_e(I)$.

However, we observe that the empirical noise event count (Fig. 6.3d) has a higher variance than the Poisson distribution (Fig. 6.3b), referred to as overdispersion. The overdispersion of the experimental data is likely caused by some inter-pixel variabilities, such as the variation of contrast threshold values [101]. Thus, we instead sample the noise count from a generalized negative binomial distribution—which is often used in lieu of Poisson when there is overdispersion—with a mean of $N \cdot p_e(I)$ and an illuminance-dependent variance obtained empirically from the calibration data. The noise count sampled from the negative binomial distribution (Fig. 6.3c) resembles the experimental count.

## 6.3 Reconstructing the Scene

Once we have a model for the mapping between scene intensity and noise event count at each pixel, we can develop an algorithm for recovering the static scene. We first estimate the true noise event count $N \cdot p_e(I)$ from the empirical noise event count that the camera measures, and then estimate the intensity $I$ by inverting Eq. (6.3). By doing this for every pixel, we can form an image for a static scene.

This inverse problem is non-trivial to solve for two reasons. First, the problem is ill-posed since Eq. (6.3) is one-to-many, as in Fig. 6.2b, for nonzero biases and thus using the noise count alone is insufficient to solve for exact intensity values. Second, even though the empirical event count is the maximum likelihood estimator of the true event count, the estimate from the empirical event counts in a finite time window will have some statistical error leading to further downstream error after inverting $p_e(I)$.

We approach the first challenge by counting the positive and negative polarities of noise events separately. We found that event cameras often set different contrast threshold values to trigger positive and negative events, since the leakage current of the event circuit causes positive events to be triggered more easily [129]. Because of potentially asymmetric behavior of the two polarities, the correlation between light intensity and noise events is also different for each polarity. As shown in Fig. 6.4, the corresponding light intensity becomes more uniquely defined when counting positive and negative events separately.

The second issue — inexact event count estimation — can also be mitigated through the use of data priors in the inverse problem. This will make the resulting reconstruction robust to small errors in the event rate. Combining these two, we train a neural network to map the

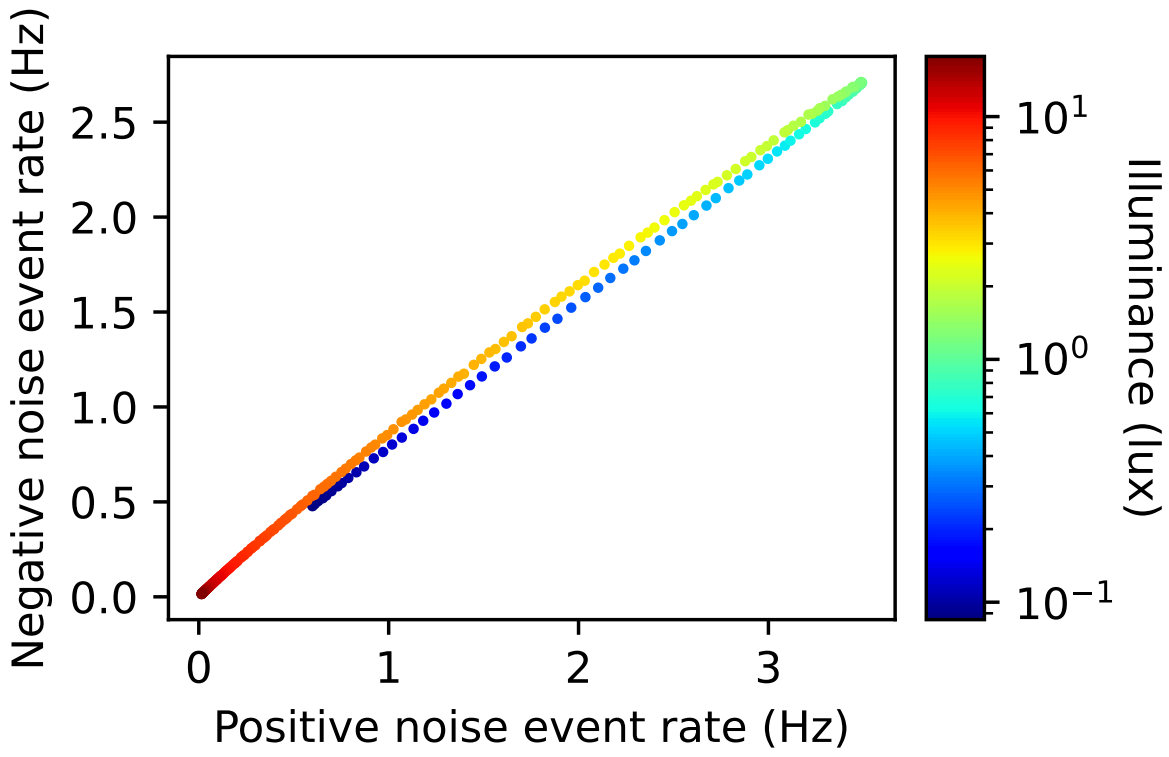


Figure 6.4: Noise event counts on positive events versus negative events. The color indicates the corresponding illuminance level.

estimated event count directly to the corresponding intensity image. The network accepts inputs with two spatial channels for positive and negative event counts. To further reduce the variance, we perform pixel binning across blocks of $2 \times 2$ pixels.

### Application on Dynamic Scenes

As in Fig. 6.1, the static scene reconstruction can also operate in parallel with the event-to-video reconstruction (E2VID) pipeline on recordings with both static and dynamic components, *e.g.*, scenes with moving foreground and static background. We first separate signal events triggered by dynamic changes from other recorded events using an event denoising algorithm [51, 42], and apply an E2VID method to reconstruct the dynamic components. The remaining events can be considered noise events triggered mostly by photon fluctuation,

which can be fed into the proposed Noise2Image model for static background reconstruction. For a variable-length recording, we aggregate noise events using a moving window with a fixed temporal width. Lastly, we stitch the dynamic and static scene reconstructions together using a binary motion mask identified from the signal events.

## 6.4 Experiments

The existing event camera datasets all have a significant amount of scene motion and/or camera motion. It is not possible to fully distinguish between events triggered by intensity changes and noise events [5], and thus they are not suitable for our goal of static scene reconstruction. Hence, we collect our own training/validation dataset, termed noise events-to-image (NE2I). NE2I contains pairs of high-resolution intensity images and noise event recordings from both experimental acquisition and synthetic noise based on the model presented in Sec. 6.2.

We image a 24.5 inch LCD monitor displaying static images. In order to calibrate the noise model, we display 256 grayscale values on the monitor and capture both their event response and light intensity. The illuminance of each grayscale value is measured using a light meter placed next to the camera[1]. Our event camera is a monochromatic Prophesee Metavision EVK3-HD. The default bias parameters were used for the camera, and no denoising was performed on the raw data. While the event camera has a high pixel count ($1280 \times 720$ pixels), it does not have an active pixel sensor (APS) that records frame-based intensity images. To register the event camera recording to the screen, we first imaged a standard checkerboard flashing on the monitor to induce events. The aggregated event count is used to establish a transformation matrix [17], which is applied to any event recordings to align it spatially with the monitor.

The full NE2I dataset consists of an in-distribution set and an out-of-distribution test set. The in-distribution data contains 1004 high-resolution images of artistic human portraits from Unsplash, split into 754 images for training, 100 for validation, and 150 for testing. The out-of-distribution test set has 100 high-resolution images from the validation set of DIV2K image super-resolution dataset [1], aiming to provide a variety of scenes much beyond the training data distribution. We intentionally chose a confined distribution for training data and a broad distribution for testing, so that the evaluation can test if the model learns local correlations instead of an image-level prior.

Given a sequence of noise events, we first aggregate them into a 2D-matrix of event count. To account for the positive and negative polarities, we separately aggregate them and store as two channels. We train a U-net to map an event count matrix into the corresponding intensity image. A modified U-net from [74] is used for enhanced performance. The network is trained using either experimental training data (described above) or synthetic event counts generated by the statistical noise model in Sec. 6.2. For experimental data training, we

[1]Since the light meter has optics different from the lens on the event camera, the measured illuminance value may not be equal to but is proportional to the actual illuminance level hitting the camera sensor.

augment the input noise count by choosing a random starting time for the 1-second window over a 10-second event recording. For synthetic data training, event counts are re-sampled each time on-the-fly.

We use E2VID as a baseline method for the evaluation of static scene recovery even though it was not trained by such data. We compare to the pre-trained model implementation [48] for the original E2VID [148], E2VID+ [165], FireNet [153], FireNet+ [165], ET-Net [193], SPADE-E2VID [20], SSL-E2VID [133], HyperE2VID [49]. A sequence of events within a 1-second window is fed into each method, and the last frame of predicted video (5 predicted frames in total) is used for evaluation.

**Metrics.**

Evaluation of all methods is performed using experimentally-collected testing data, with the quality of recovered intensity images being calculated for the in-distribution and out-of-distribution testing datasets using three common quantitative metrics [148]: peak signal-to-noise ratio (PSNR), structured similarity (SSIM) and perceptual similarity (LPIPS [209]). LPIPS is computed using pre-trained AlexNet with image intensity values normalized to [-1, 1].

## Results

For static-only scenes, the quantitative scores for Noise2Image as well as baseline methods using NE2I dataset are reported in Table 6.1. As shown in Fig. 6.5, the Noise2Image model trained by either experimental or synthetic data recovers detailed intensity images from the count of noise events. Both Noise2Image models generalized well to out-of-distribution testing data, providing a good level of contrast and details, suggesting that Noise2Image learns the correlation between noise event and intensity. The Noise2Image model trained by experimental data out-performed the one trained by the synthetic data by 4.1dB of PSNR. As the synthetic data-trained model often recovered speckle-like patterns for the uniform image background, we speculate that this performance margin is caused by the spatial correlation of the noise events, which is beyond our synthesis model but potentially captured by the experimental data training. Baseline E2VID methods did not perform well on static scene reconstruction for two reasons: they were only trained by scenes with various degrees of motion (not static scenes) and the training data of E2VID was generated without an intensity-dependent noise model [147].

In addition to testing data, we tested Noise2Image on real-world scenes, both indoor and outdoor. The Noise2Image model trained by images displayed on a monitor generalizes well for scenes outside of the laboratory setting, as shown in Fig. 6.1 and Fig. 6.7. We have implemented a real-time demo running on a laptop computer as well.

[2]Since E2VID models were trained by synthetic motion data generated from the MS-COCO dataset [102], both of our testing sets are considered as out-of-distribution.

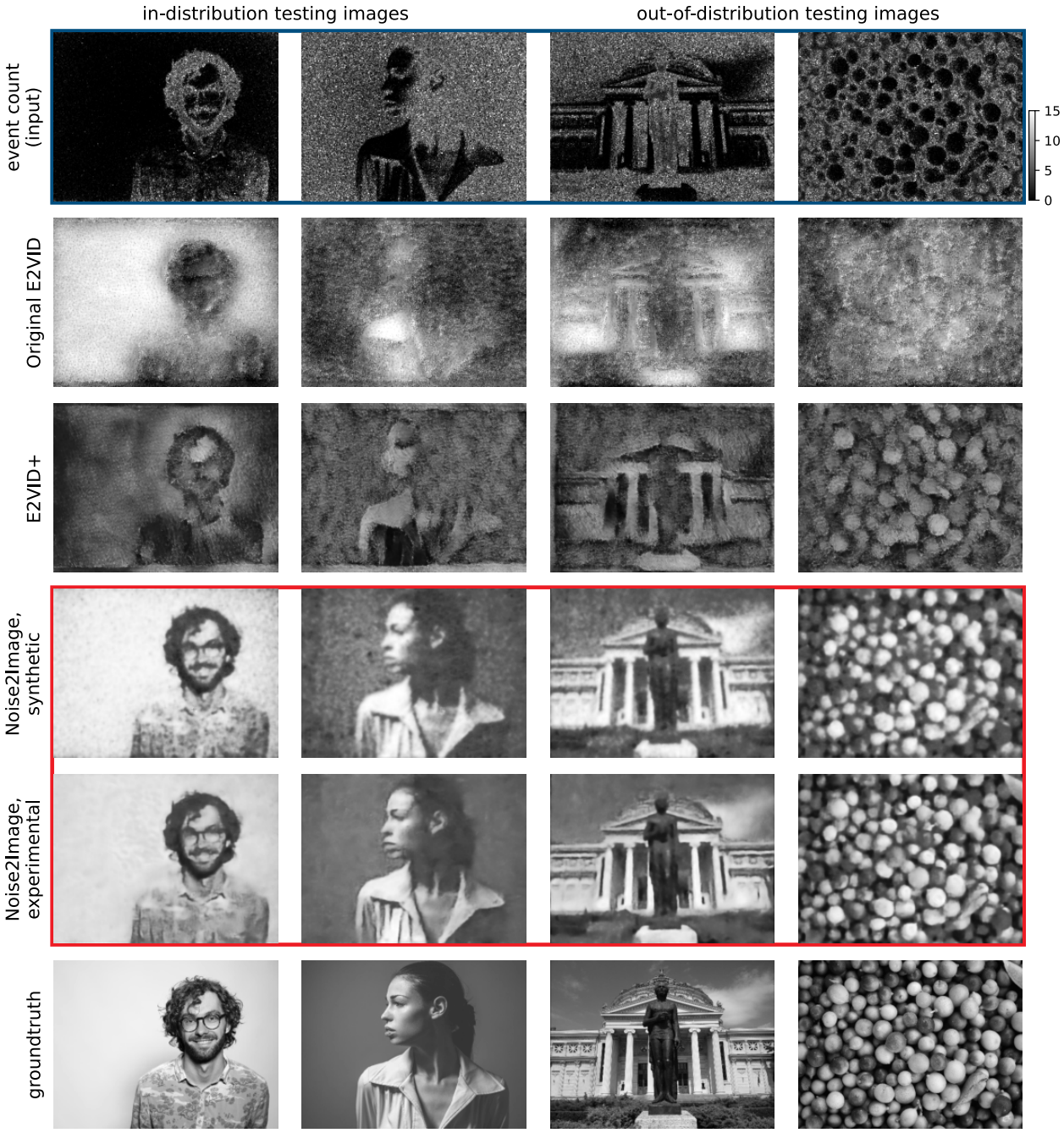


Figure 6.5: Comparison between Noise2Image and baseline pre-trained event-to-video (E2VID) methods on noise event-to-intensity reconstruction. The first row shows the input event count (captured in experiment) aggregated over a 1-second window. Noise2Image is trained using either synthetic or experimental data as specified. Full comparison with all baseline methods can be found in Fig. 6.6.

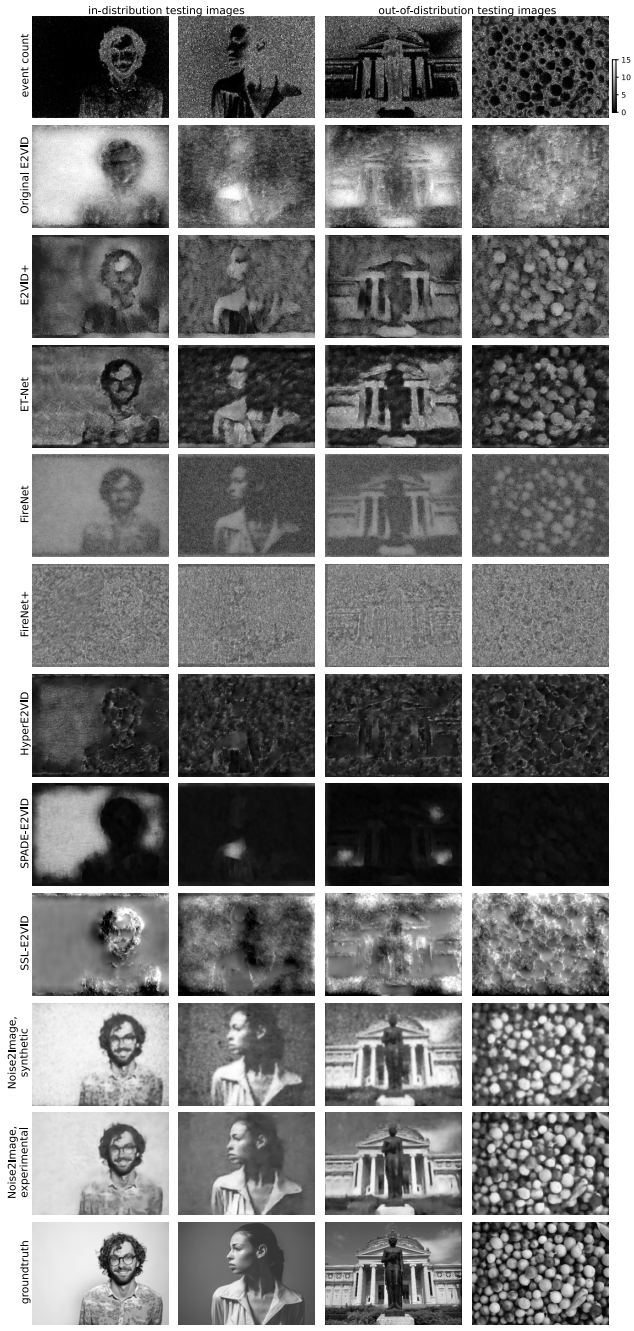


Figure 6.6: Additional comparison between Noise2Image and all baseline event-to-video (E2VID) methods on noise event-to-intensity reconstruction. The first row shows the input event count (captured in experiment) aggregated over a 1-second window.

| Methods | In-distribution test | | | Out-of-distribution test | | |
|---|---|---|---|---|---|---|
| | PSNR | SSIM | LPIPS | PSNR | SSIM | LPIPS |
| Original E2VID [148] | 7.76 | 0.043 | 1.131 | 10.1 | 0.055 | 1.032 |
| E2VID+ [165] | 8.72 | 0.148 | 0.872 | 9.26 | 0.108 | 0.768 |
| FireNet [153] | 9.35 | 0.089 | 1.047 | 10.5 | 0.098 | 0.970 |
| FireNet+ [165] | 9.78 | 0.063 | 0.959 | 9.77 | 0.057 | 0.800 |
| ET-Net [193] | 9.13 | 0.166 | 0.805 | 10.1 | 0.131 | 0.730 |
| SPADE-E2VID [20] | 7.74 | 0.229 | 0.856 | 8.74 | 0.149 | 0.870 |
| SSL-E2VID [133] | 8.81 | 0.133 | 0.764 | 8.95 | 0.123 | 0.720 |
| HyperE2VID [49] | 9.09 | 0.159 | 0.858 | 0.89 | 0.094 | 0.759 |
| Noise2Image, synthetic | 20.9 | 0.589 | 0.462 | 16.3 | 0.468 | 0.583 |
| Noise2Image, experimental | 25.0 | 0.742 | 0.349 | 19.3 | 0.552 | 0.509 |

Table 6.1: Quantitative results for static scene reconstruction with in-distribution and out-of-distribution testing data. Our Noise2Image models are trained with either synthetic or experimental noise data. Pre-trained events-to-video reconstruction (E2VID) methods[2]are used as baselines. We report peak signal-to-noise ratio (PSNR), structured similarity (SSIM), and perceptual similarity (LPIPS [209]).

**Effect of aggregation duration.**

Table 6.2 shows the effect of varying the aggregation window time for training and testing the Noise2Image method. As the aggregation duration shortens, fewer noise events will be triggered, and the estimation of the true event count becomes less accurate. Table 6.2 shows that the Noise2Image reconstruction works even with short aggregation duration (0.1 seconds), although longer integration will result in better reconstruction quality.

| Aggregation duration | In-distribution test | | | Out-of-distribution test | | |
|---|---|---|---|---|---|---|
| | PSNR | SSIM | LPIPS | PSNR | SSIM | LPIPS |
| 0.1 s | 23.8 | 0.714 | 0.391 | 18.4 | 0.513 | 0.586 |
| 0.25 s | 24.4 | 0.721 | 0.377 | 18.8 | 0.522 | 0.564 |
| 0.5 s | 24.7 | 0.738 | 0.358 | 19.1 | 0.545 | 0.537 |
| 1 s | 25.0 | 0.742 | 0.349 | 19.3 | 0.552 | 0.509 |
| 2 s | 25.4 | 0.765 | 0.324 | 19.5 | 0.598 | 0.473 |

Table 6.2: Noise2Image performance using noise event counts aggregated over different time windows. For each aggregation duration, the Noise2Image model is trained and evaluated using the experimental data with identical training setting.

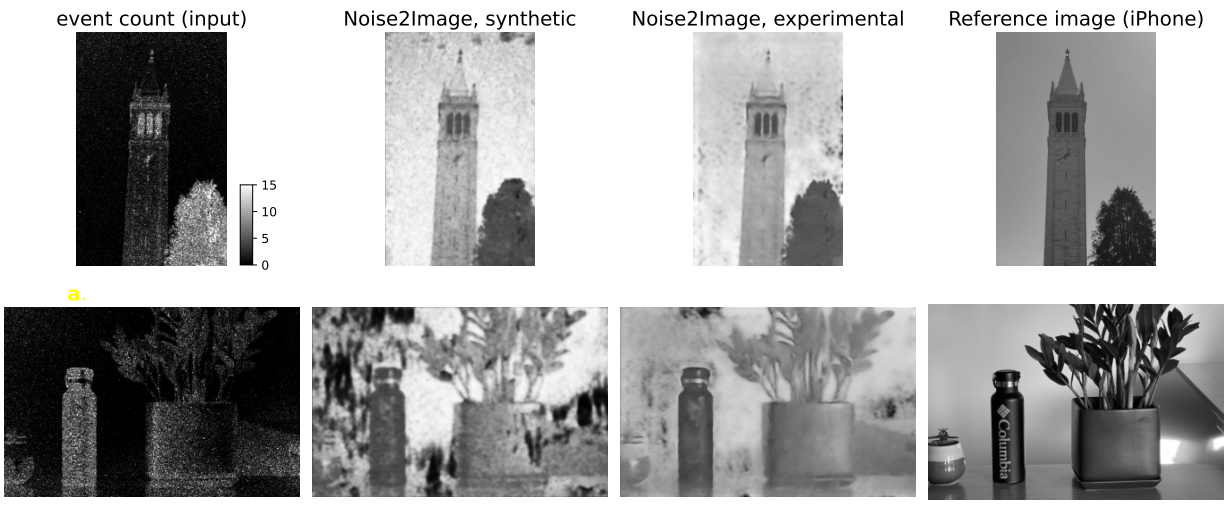


Figure 6.7: Real-world examples of Noise2Image taken outside of the laboratory setting. The Noise2Image model trained by synthetic data results in background artifacts on the in-door scene (second row, second column), presumably caused by the one-to-many relationship of Eq. 3. In contrast, the Noise2Image trained by experimental data predicts the background correctly, hinting that there exist spatial correlations in the experimental noise events, beyond our derived spatially independent noise event synthesis. The reference images were taken by an iPhone 12 plus back camera.

**Reconstructing scenes with dynamic components.**

Finally, we demonstrated that Noise2Image is complementary to E2VID reconstruction when the scene has both static and dynamic components. We imaged a fan rapidly moving in front of a static scene, and fed signal events into a pre-trained E2VID model [148]. While the E2VID method performed well on the moving object, it could not recover the background static scene (Fig. 6.8c). To incorporate Noise2Image, we first identified a motion mask at each timepoint (Fig. 6.8b) by thresholding signal events. We then aggregated noise events and fed the event count at pixels without motion to the Noise2Image model. Stitching together the moving foreground from E2VID and the static background from Noise2Image, we obtained the final high-quality reconstruction as in Fig. 6.8d.

## 6.5 Discussion

Our experiments were done using a single event camera, and there might be differences in the noise characteristics between different event circuit designs. However, our finding in Sec. 6.2 is based on the photon arrival process, which is generalizable to other event camera hardware.

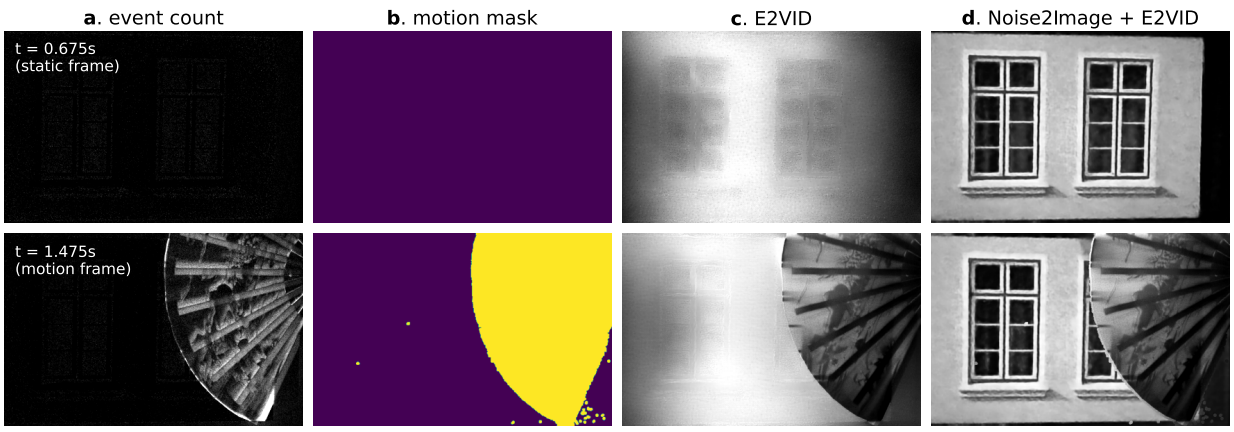


Figure 6.8: Dynamic scene reconstruction of a moving fan in front of a static scene. **a**. Aggregated event count. The first row is at a timepoint with no motion and only faint noise, and the second row has motion. **b**. Motion mask obtained from thresholding the signal events as determined by the event denoiser. **c**. E2VID reconstruction using pre-trained model from [148]. **d**. The dynamic foreground reconstructed by E2VID and the static background reconstructed by Noise2Image are stitched together.

This correlation between noise events and illuminance level was also independently reported in previous studies using different event cameras [63, 77, 42, 64].

Denoising and camera bias parameters [113] can also affect this illuminance dependency. Our data was acquired without denoising and using bias parameters default to our event camera. We also tested different high-pass filtering bias values which affect the correlation between noise event rate and illuminance (Fig. 6.9). For this reason, the trained Noise2Image model is bound to the camera model and bias parameters that were used in the training data collection. We hope that our modeling of noise events can be improved upon by incorporating the analog nature of event sampling and modeling additional event camera bias parameters, *e.g.*using non-parametric models to account for unknown camera behavior.

One practical constraint for our study is that the dynamic range of Noise2Image is limited by the monitor we used. As a result, the increase in noise events at high illuminance settings due to leakage current cannot be measured using our setup. This issue can likely be alleviated by changing to a high-dynamic range and high-brightness monitor or projector for data collection. We hypothesize that our Noise2Image approach will still be valid for brighter scenes since the statistics of leakage current-induced noise events is quite distinct [129].

Another future direction is to incorporate the proposed noise event modeling into existing event camera simulators. This will help synthesize data with realistic noise events, which can be used for training event-to-video recovery (E2VID) models. Overall, the model similarities between Noise2Image and E2VID suggest that in the future, Noise2Image can be

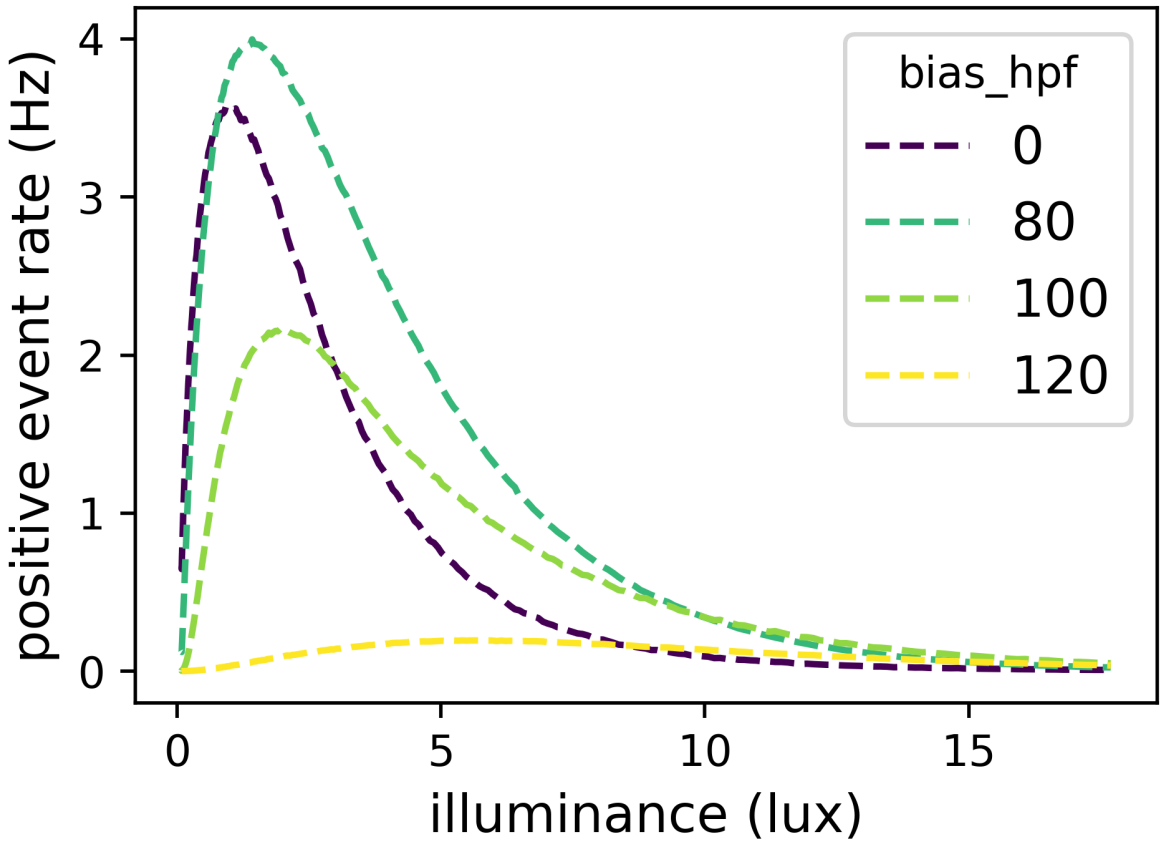


Figure 6.9: Experimentally captured noise event rate vs. illuminance levels using various sensor high-pass filter bias values which is called "bias_hpf" in Prophesee Metavision SDK. The correlation between illuminance and event rate changes with different bias_hpf values.

incorporated into the E2VID pipeline via more realistic training data, complementing each other's goals.

In summary, we demonstrated imaging of static intensity using an event camera with no additional hardware. This is made possible by the fact that photon-noise triggers events which are correlated with illuminance. We developed a statistical model of the noise event generation, and leveraged it to develop a strategy called Noise2Image, which maps noise events to an intensity image via a neural network that incorporates local priors. We demonstrate that our approach works well quantitatively and qualitatively on experimental event recordings, including ones taken in the wild. We also show that our method is compatible with scenes that have object dynamics. Our hope is that Noise2Image reduces the need for event cameras to have additional hardware for measuring static scenes.

# Chapter 7

# Conclusion

This thesis has explored methods to transform unwanted and unknown motion into useful information through the joint estimation of motion and scene. We find this concept powerful and widely applicable. We have applied it to various optical imaging systems for objects at different scales, ranging from sub-cellular super-resolution imaging to high-speed imaging of dynamic objects like bouncing tennis balls. We developed a neural space-time model to resolve dynamic scenes using data from imaging systems originally designed for static scenes. In Chapter 6, we address a problem opposite to Chapter 5, which is to recover static scenes using sensors designed exclusively for dynamic observation. We hope both problems have provided valuable insights for the future development of dynamic imaging systems. In this last chapter, we will discuss some open challenges and future research directions.

## 7.1 Challenges and future directions

### Computational efficiency

Computational efficiency is a significant challenge for many iterative image reconstruction algorithms. Often, the reconstruction runtime is much longer than the data acquisition time, even with powerful computing hardware. As a result, not many computational imaging systems, especially those used in biological or scientific imaging, can achieve real-time reconstructions. Consequently, biologists often prefer using widefield or confocal microscopes, even when a more powerful, less phototoxic commercial SIM system is available. This computational inefficiency becomes even more pronounced when we need to reconstruct motion as well. The joint reconstruction of unknown motion and scene requires either alternating the update steps (as described in Chapter 2) or coarse-to-fine process to avoid converging to local minimums (as described in Chapter 5), both of which make the reconstruction process slower.

Nevertheless, we have seen a lot of progress to improve algorithmic efficiency in recent years, largely due to the increasing popularity of novel view synthesis in computer graph-

ics community [116, 122, 55, 86]. We used the hash embedding method [122] to make the dynamic 3D reconstruction feasible in Chapter 5. We believe there are still numerous algorithmic improvements to be made to accelerate the joint reconstruction of scene and motion. In addition, the community is moving toward more efficient floating point arithmetic, such as half-precision or even 8-bit floating point. With the new computing hardware driven by the rapid development of foundation models, we envision that efficiency can be further improved by at least an order of magnitude in the next couple of years. This improvement will hopefully remove the computing barrier for wider applicability.

## Data prior for dynamic imaging

Advances in generative learning enable us to build a robust data priors for natural images using large-scale image datasets like ImageNet [40]. The learned data prior can be used for imaging reconstructions, such that it can accelerate the iterative optimization process [119, 170, 99] and/or improve the reconstruction quality for ill-posed problems [125, 208, 81]. However, most methods discussed in this thesis (except in Chapter 6) do not involve a data prior for the following three reasons. Firstly, a large-scale dataset is not always available, especially for more specialized, non-standardized imaging systems. Secondly, even if we can obtain many raw measurements, obtaining high-quality groundtruth reconstruction is a challenge for inverse problems. Thirdly, the amount of data required for a good 2D/3D+time spatiotemporal data prior will be astronomical, considering it already takes millions of images to build a data prior for 2D images.

A future direction is to use a 2D/3D data prior combined with a motion model as the spatiotemporal data prior. The motion model would enhance the existing static data prior to support dynamic imaging reconstruction. With this data prior, the dynamic reconstruction (discussed in Chapter 5) can be significantly accelerated, as the scene can be directly inferred from the prior. Another possible direction is to build a robust motion prior from dynamic data, which can either depend on or be independent of the scene.

## Complex dynamics and time series reconstruction

In Chapter 5, we introduce the neural space-time model to handle deformable motion, enabling high-fidelity dynamic imaging reconstruction. In our demonstration, we estimate the motion within a single acquisition time, and thus the estimated motion is less complex because of its short duration.

While the recovery of fast, transient dynamics is valuable, especially for multi-shot imaging systems, biologists often want to observe some biological process over a long period of time. This observation typically involves collecting a time-series dataset with multiple acquisition timepoints and analyzing the dynamics over these reconstructions. This conventional approach assumes motion artifact-free reconstruction at each timepoint and also robust analysis tools for the dynamics, *e.g.*, particle or cell tracking. In contrast, the neural space-time

model offers an alternative solution that bypasses these two assumptions, performing image reconstruction and dynamic analysis simultaneously.

A logical next step is to extend the neural space-time model to handle longer, more complex motion using a time-series dataset. A potential challenge is that the modeling capacity of the neural space-time model might saturate when the dynamics get too long or complex. A data representation that does not rely on coordinate-based neural networks, such as those used in [55, 86], may be more scalable and better suited for handling long and complex dynamics. Another possible pitfall is that the current motion modeling assumes any frame of a dynamic scene can be mapped to a common reference frame. This assumption is likely to fail for longer dynamics, like embryonic development, and some tweaks on the neural space-time model might be necessary to enable the reconstruction of time-dependent scenes.

## Task-driven dynamic imaging

While this thesis advances unknown motion recovery and dynamic imaging formation, the downstream tasks involving recovered dynamics have not been fully explored. These tasks can have significant impact in real-world applications. *e.g.*, in Chapter 2, we demonstrate that it is possible to perform an axial scan of the sample, without relying an expensive, precision $z$-stage, by hand defocusing and the estimation of $z$-depth, which can be quite practical for low-resource settings. In Chapter 5, we can directly obtain the velocity and acceleration from the neural space-time model after the dynamic reconstruction, which can possibly be used to learn more insights and predict the state of intracellular components. Besides, event cameras are particularly suited for downstream tasks that interpret the dynamics, as the event recordings are sparse and only contain dynamic information. If tasks require contextual information from the static scene, the Noise2Image method in Chapter 6 can recover this information using the statistical properties of noise events.

Another opportunity lies in optimizing the design of imaging systems for specific tasks. In Chapter 3, we use an end-to-end optimization to design the illumination patterns for 3D refractive-index tomography. The optimization minimizes the voxel-level discrepancy between the groundtruth and the recovered refractive indices. If there is a certain downstream task we hope to achieve using the recovered refractive indices, the performance metric of the task can be directly set as the optimization objective, and thus the design can be optimized for the end application. This idea of using metrics other than the pixel intensity discrepancy can be further extended to general image reconstruction settings. When the pixel intensity of raw images is affected, *e.g.*, due to strong noise or scattering, task-based metrics may sometimes get more informative loss and gradient for the reconstruction.

## Reliable and accessible software

There is an urgent need of building reliable software tools, in order to make our computational approaches accessible to actual users who may not be software persons by training.

From researcher's perspective, however, making reliable and reusable software from a successful research paper still requires a significant amount of effort. This indicates a significant gap between research code and reliable production software, which could be bridged by developing more robust software infrastructure. The software infrastructure should include highly modular tools and functions shared by various computational imaging methods, such as data pre-processing, iterative optimization methods [21], and data visualization [161]. For example, during the development of the neural space-time model (in Chapter 5 and 6), I built a software tool to handle gradient descent-based image reconstructions with better performance and reliability [21]. With more investment in common infrastructure, researchers will not need to reinvent the wheel each time and can focus on the novel implementations.

To improve the robustness, it is critical to have comprehensive documentations during the code release. In addition to the information of software installation and usage, it is worth to also have a in-depth discussion of hyper-parameter tuning and the working boundary or failure modes of the software. The analysis of failure modes, as discussed in Section 5.5, can help users better understand the method, ultimately facilitating its adaption.

# Bibliography


[1] Eirikur Agustsson and Radu Timofte. "Ntire 2017 challenge on single image super-resolution: Dataset and study". In: *Proceedings of the IEEE conference on computer vision and pattern recognition workshops*. 2017, pp. 126–135.

[2] LJ Allen and MP Oxley. "Phase retrieval from series of images obtained by defocus variation". In: *Optics communications* 199.1-4 (2001), pp. 65–75.

[3] Anastasios N Angelopoulos et al. "Event based, near eye gaze tracking beyond 10,000 hz". In: *IEEE Transactionson Visualizations and Graphics* (2021).

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

[5] R. Wes Baldwin et al. "Event Probability Mask (EPM) and Event Denoising Convolutional Neural Network (EDnCNN) for Neuromorphic Cameras". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. June 2020.

[6] Štefan Bálint et al. "Correlative live-cell and superresolution microscopy reveals cargo transport dynamics at microtubule intersections". In: *Proceedings of the National Academy of Sciences* 110.9 (2013), pp. 3375–3380.

[7] Patrick Bardow, Andrew J Davison, and Stefan Leutenegger. "Simultaneous optical flow and intensity estimation from an event camera". In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2016, pp. 884–892.

[8] Juan Barrios-Avilés et al. "Less data same information for event-based sensors: A bioinspired filtering and data reduction algorithm". In: *Sensors* 18.12 (2018), p. 4122.

[9] Chinmay Belthangady and Loic A Royer. "Applications, promises, and pitfalls of deep learning for fluorescence image reconstruction". In: *Nature methods* 16.12 (2019), pp. 1215–1225.

[10] Ryad Benosman et al. "Event-based visual flow". In: *IEEE transactions on neural networks and learning systems* 25.2 (2013), pp. 407–417.

[11] Raphael Berner et al. "A 240× 180 10mw 12us latency sparse-output vision sensor for mobile applications". In: *2013 Symposium on VLSI Circuits*. IEEE. 2013, pp. C186–C187.

[12] Eric Betzig et al. "Imaging intracellular fluorescent proteins at nanometer resolution". In: *science* 313.5793 (2006), pp. 1642–1645.

[13] Basanta Bhaduri et al. "Diffraction phase microscopy: principles and applications in materials and life sciences". In: *Advances in Optics and Photonics* 6.1 (2014), pp. 57–119.

[14] Max Born and Emil Wolf. *Principles of optics: electromagnetic theory of propagation, interference and diffraction of light*. Elsevier, 2013.

[15] James Bradbury et al. "JAX: composable transformations of Python+ NumPy programs". In: *Version 0.1* 55 (2018).

[16] James Bradbury et al. *JAX: composable transformations of Python+NumPy programs*. Version 0.3.13. 2018. URL: `http://github.com/google/jax`.

[17] G. Bradski. "The OpenCV Library". In: *Dr. Dobb's Journal of Software Tools* (2000).

[18] Christian Brandli, Lorenz Muller, and Tobi Delbruck. "Real-time, high-speed video decompression using a frame-and event-based DAVIS sensor". In: *2014 IEEE International Symposium on Circuits and Systems (ISCAS)*. IEEE. 2014, pp. 686–689.

[19] Michal Byra et al. "Exploring the performance of implicit neural representations for brain image registration". In: *Scientific Reports* 13.1 (2023), p. 17334.

[20] Pablo Rodrigo Gantier Cadena et al. "SPADE-E2VID: Spatially-Adaptive Denormalization for Event-Based Video Reconstruction". In: *IEEE Transactions on Image Processing* 30 (2021), pp. 2488–2500.

[21] Ruiming Cao. *rmcao/CalCIL: v0.0.2: release*. 2024. DOI: `10.5281/ZENODO.13119649`. URL: `https://zenodo.org/doi/10.5281/zenodo.13119649`.

[22] Ruiming Cao, Guanghan Meng, and Laura Waller. "Speckle Structured Illumination of Dynamic Samples with a Neural Space-time Model". In: *Computational Optical Sensing and Imaging*. Optica Publishing Group. 2023, CW4D–2.

[23] Ruiming Cao et al. "3D Differential Phase Contrast Microscopy with Axial Motion Deblurring". In: *Computational Optical Sensing and Imaging*. Optica Publishing Group. 2020, CF4C–2.

[24] Ruiming Cao et al. "Algorithmic self-calibration for optimized 3D quantitative differential phase contrast microscopy". In: *Quantitative Phase Imaging VII*. Vol. 11653. SPIE. 2021, 116530R.

[25] Ruiming Cao et al. "Dynamic structured illumination microscopy with a neural space-time model". In: *2022 IEEE International Conference on Computational Photography (ICCP)*. IEEE. 2022, pp. 1–12.

[26] Ruiming Cao et al. "Neural space-time model for dynamic scene recovery in multi-shot computational imaging systems". In: *bioRxiv* (2024), pp. 2024–01.

[27] Ruiming Cao et al. "Noise2Image: Noise-Enabled Static Scene Recovery for Event Cameras". In: *arXiv preprint arXiv:2404.01298* (2024).

[28] Ruiming Cao et al. "Self-calibrated 3D differential phase contrast microscopy with optimized illumination". In: *Biomedical Optics Express* 13.3 (2022), pp. 1671–1684.

[29] Ruiming Cao et al. "Speckle flow structured illumination microscopy for dynamic super-resolution imaging". In: *High-Speed Biomedical Imaging and Spectroscopy VIII*. Vol. 12390. SPIE. 2023, p. 1239002.

[30] Lucas von Chamier et al. "Democratising deep learning for microscopy with ZeroCostDL4Mic". In: *Nature communications* 12.1 (2021), p. 2276.

[31] Michael Chen, Zachary F Phillips, and Laura Waller. "Quantitative differential phase contrast (DPC) microscopy with computational aberration correction". In: *Optics Express* 26.25 (2018), pp. 32888–32899.

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

[33] Wonshik Choi et al. "Extended depth of focus in tomographic phase microscopy using a propagation algorithm". In: *Optics letters* 33.2 (2008), pp. 171–173.

[34] Shwetadwip Chowdhury and Joseph Izatt. "Structured illumination quantitative phase microscopy for enhanced resolution amplitude and phase imaging". In: *Biomedical optics express* 4.10 (2013), pp. 1795–1805.

[35] Shwetadwip Chowdhury et al. "High-resolution 3D refractive index microscopy of multiple-scattering samples from intensity images". In: *Optica* 6.9 (2019), pp. 1211–1219.

[36] George E Cragg and Peter TC So. "Lateral resolution enhancement with standing evanescent waves". In: *Optics letters* 25.1 (2000), pp. 46–48.

[37] Daniel Czech and Garrick Orchard. "Evaluating noise filtering for event-based asynchronous change detection image sensors". In: *2016 6th IEEE International Conference on Biomedical Robotics and Biomechatronics (BioRob)*. IEEE. 2016, pp. 19–24.

[38] T Delbruck and CA Mead. "Analog VLSI phototransduction by continuous-time, adaptive, logarithmic photoreceptor circuits". In: *Technical report* (1995).

[39] Tobi Delbruck et al. "Frame-free dynamic digital vision". In: *Proceedings of Intl. Symp. on Secure-Life Electronics, Advanced Electronics for Quality Life and Society*. Vol. 1. Citeseer. 2008, pp. 21–26.

[40] Jia Deng et al. "Imagenet: A large-scale hierarchical image database". In: *2009 IEEE conference on computer vision and pattern recognition*. Ieee. 2009, pp. 248–255.

[41] Thomas Dertinger et al. "Fast, background-free, 3D super-resolution optical fluctuation imaging (SOFI)". In: *Proceedings of the National Academy of Sciences* 106.52 (2009), pp. 22287–22292.

[42] Saizhe Ding et al. "E-MLB: Multilevel Benchmark for Event-Based Camera Denoising". In: *IEEE Transactions on Multimedia* (2023).

[43] Alexey Dosovitskiy et al. "An image is worth 16x16 words: Transformers for image recognition at scale". In: *arXiv preprint arXiv:2010.11929* (2020).

[44] Regina Eckert, Zachary F Phillips, and Laura Waller. "Efficient illumination angle self-calibration in Fourier ptychography". In: *Applied optics* 57.19 (2018), pp. 5434–5442.

[45] Richard L Ehman and Joel P Felmlee. "Adaptive technique for high-definition MR imaging of moving structures." In: *Radiology* 173.1 (1989), pp. 255–263.

[46] Richard L Ehman et al. "Magnetic resonance imaging with respiratory gating: techniques and advantages". In: *American journal of Roentgenology* 143.6 (1984), pp. 1175–1182.

[47] Thomas Elsken, Jan Hendrik Metzen, and Frank Hutter. "Neural architecture search: A survey". In: *The Journal of Machine Learning Research* 20.1 (2019), pp. 1997–2017.

[48] Burak Ercan et al. "EVREAL: Towards a Comprehensive Benchmark and Analysis Suite for Event-based Video Reconstruction". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2023, pp. 3942–3951.

[49] Burak Ercan et al. "HyperE2VID: Improving Event-Based Video Reconstruction via Hypernetworks". In: *arXiv preprint arXiv:2305.06382* (2023).

[50] Richard [Dicklyon] F. Lyon. *Propellor with rolling-shutter artifact*. July 2021. URL: `https://commons.wikimedia.org/wiki/File:Propellor_with_rolling-shutter_artifact.jpg`.

[51] Yang Feng et al. "Event density based denoising method for dynamic vision sensor". In: *Applied Sciences* (2020).

[52] Matthias Feurer and Frank Hutter. "Hyperparameter optimization". In: *Automated machine learning*. Springer, Cham, 2019, pp. 3–33.

[53] Reto Fiolka et al. "Time-lapse two-color 3D imaging of live cells with doubled resolution using structured illumination". In: *Proceedings of the National Academy of Sciences* 109.14 (2012), pp. 5311–5315.

[54] Ronny Förster et al. "Motion artefact detection in structured illumination microscopy for live cell imaging". In: *Optics Express* 24.19 (2016), pp. 22121–22134.

[55] Sara Fridovich-Keil et al. "Plenoxels: Radiance fields without neural networks". In: *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 2022, pp. 5501–5510.

[56] H Fujii and T Asakura. "A contrast variation of image speckle intensity under illumination of partially coherent light". In: *Optics Communications* 12.1 (1974), pp. 32–38.

[57] Guillermo Gallego et al. "Event-based vision: A survey". In: *IEEE transactions on pattern analysis and machine intelligence* 44.1 (2020), pp. 154–180.

[58] Baoliang Ge et al. "Single-frame label-free cell tomography at speed of more than 10,000 volumes per second". In: *arXiv preprint arXiv:2202.03627* (2022).

[59] Daniel Gehrig et al. "Video to events: Recycling video datasets for event cameras". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2020, pp. 3586–3595.

[60] Antoine G Godin, Brahim Lounis, and Laurent Cognet. "Super-resolution microscopy approaches for live cell imaging". In: *Biophysical journal* 107.8 (2014), pp. 1777–1784.

[61] Joseph W Goodman. "Introduction to Fourier optics. 3rd". In: *Roberts and Company Publishers* (2005).

[62] Alexandre Goy et al. "High-resolution limited-angle phase tomography of dense layered objects using deep neural networks". In: *Proceedings of the National Academy of Sciences* 116.40 (2019), pp. 19848–19856.

[63] Rui Graca and Tobi Delbruck. "Unraveling the paradox of intensity-dependent DVS pixel noise". In: *arXiv preprint arXiv:2109.08640* (2021).

[64] Rui Graça, Brian McReynolds, and Tobi Delbruck. "Shining light on the DVS pixel: A tutorial and discussion about biasing and optimization". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2023, pp. 4044–4052.

[65] Shasha Guo and Tobi Delbruck. "Low cost and latency event camera background activity denoising". In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 45.1 (2022), pp. 785–795.

[66] Mats GL Gustafsson. "Nonlinear structured-illumination microscopy: wide-field fluorescence imaging with theoretically unlimited resolution". In: *Proceedings of the National Academy of Sciences* 102.37 (2005), pp. 13081–13086.

[67] Mats GL Gustafsson. "Surpassing the lateral resolution limit by a factor of two using structured illumination microscopy". In: *Journal of microscopy* 198.2 (2000), pp. 82–87.

[68] Mats GL Gustafsson et al. "Three-dimensional resolution doubling in wide-field fluorescence microscopy by structured illumination". In: *Biophysical journal* 94.12 (2008), pp. 4957–4970.

[69] Nils Gustafsson et al. "Fast live-cell conventional fluorophore nanoscopy with ImageJ through super-resolution radial fluctuations". In: *Nature communications* 7.1 (2016), p. 12471.

[70] Samuel W Hasinoff et al. "Burst photography for high dynamic range and low-light imaging on mobile cameras". In: *ACM Transactions on Graphics (ToG)* 35.6 (2016), pp. 1–12.

[71] Stefan W Hell and Jan Wichmann. "Breaking the diffraction resolution limit by stimulated emission: stimulated-emission-depletion fluorescence microscopy". In: *Optics letters* 19.11 (1994), pp. 780–782.

[72] Javier Hidalgo-Carrió, Guillermo Gallego, and Davide Scaramuzza. "Event-aided direct sparse odometry". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2022, pp. 5781–5790.

[73] Masahiko Hirano et al. "A highly photostable and bright green fluorescent protein". In: *Nature Biotechnology* 40.7 (2022), pp. 1132–1142.

[74] Jonathan Ho, Ajay Jain, and Pieter Abbeel. "Denoising diffusion probabilistic models". In: *Advances in neural information processing systems* 33 (2020), pp. 6840–6851.

[75] Kurt Hornik, Maxwell Stinchcombe, and Halbert White. "Multilayer feedforward networks are universal approximators". In: *Neural networks* 2.5 (1989), pp. 359–366.

[76] Roarke Horstmeyer et al. "Diffraction tomography with Fourier ptychography". In: *Optica* 3.8 (2016), pp. 827–835.

[77] Yuhuang Hu, Shih-Chii Liu, and Tobi Delbruck. "v2e: From video frames to realistic DVS events". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2021, pp. 1312–1321.

[78] Bo Huang, Hazen Babcock, and Xiaowei Zhuang. "Breaking the diffraction barrier: super-resolution imaging of cells". In: *Cell* 143.7 (2010), pp. 1047–1058.

[79] Bo Huang, Mark Bates, and Xiaowei Zhuang. "Super-resolution fluorescence microscopy". In: *Annual review of biochemistry* 78 (2009), pp. 993–1016.

[80] Herve Hugonnet, Moosung Lee, and YongKeun Park. "Optimizing illumination in three-dimensional deconvolution microscopy for accurate refractive index tomography". In: *Optics Express* 29.5 (2021), pp. 6293–6301.

[81] Ajil Jalal et al. "Robust compressed sensing mri with deep generative priors". In: *Advances in Neural Information Processing Systems* 34 (2021), pp. 14938–14954.

[82] Zhe Jiang et al. "Learning event-based motion deblurring". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2020, pp. 3320–3329.

[83] Michael Kellman et al. "Data-driven design for fourier ptychographic microscopy". In: *2019 IEEE International Conference on Computational Photography (ICCP)*. IEEE. 2019, pp. 1–8.

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

[85] Michael R Kellman et al. "Physics-based learned design: Optimized coded-illumination for quantitative phase imaging". In: *IEEE Transactions on Computational Imaging* 5.3 (2019), pp. 344–353.

[86] Bernhard Kerbl et al. "3D Gaussian Splatting for Real-Time Radiance Field Rendering." In: *ACM Trans. Graph.* 42.4 (2023), pp. 139–1.

[87] Junho Kim et al. "N-imagenet: Towards robust, fine-grained object recognition with event cameras". In: *Proceedings of the IEEE/CVF international conference on computer vision*. 2021, pp. 2146–2156.

[88] Myung K Kim. "Principles and techniques of digital holographic microscopy". In: *SPIE reviews* 1.1 (2010), p. 018005.

[89] Taewoo Kim et al. "White-light diffraction tomography of unlabelled live cells". In: *Nature Photonics* 8.3 (2014), pp. 256–263.

[90] Diederik P Kingma and Jimmy Ba. "Adam: A method for stochastic optimization". In: *arXiv preprint arXiv:1412.6980* (2014).

[91] Peter Kner et al. "Super-resolution video microscopy of live cells by structured illumination". In: *Nature methods* 6.5 (2009), pp. 339–342.

[92] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. "Imagenet classification with deep convolutional neural networks". In: *Advances in neural information processing systems* 25 (2012).

[93] Romain F Laine et al. "High-fidelity 3D live-cell nanoscopy through data-driven enhanced super-resolution radial fluctuation". In: *Nature Methods* (2023), pp. 1–8.

[94] Amit Lal, Chunyan Shan, and Peng Xi. "Structured illumination microscopy image reconstruction algorithm". In: *IEEE Journal of Selected Topics in Quantum Electronics* 22.4 (2016), pp. 50–63.

[95] Peter Lanzer et al. "ECG-synchronized cardiac MR imaging: method and evaluation." In: *Radiology* 155.3 (1985), pp. 681–686.

[96] Andrew C Larson et al. "Self-gated cardiac cine MRI". In: *Magnetic Resonance in Medicine: An Official Journal of the International Society for Magnetic Resonance in Medicine* 51.1 (2004), pp. 93–102.

[97] Hongmin Li et al. "Cifar10-dvs: an event-stream dataset for object classification". In: *Frontiers in neuroscience* 11 (2017), p. 309.

[98] Jiaji Li et al. "High-speed in vitro intensity diffraction tomography". In: *Advanced Photonics* 1.6 (2019), p. 066004.

[99] Dong Liang et al. "Deep magnetic resonance image reconstruction: Inverse problems meet neural networks". In: *IEEE Signal Processing Magazine* 37.1 (2020), pp. 141–151.

[100] Orly Liba et al. "Handheld mobile photography in very low light." In: *ACM Trans. Graph.* 38.6 (2019), pp. 164–1.

[101] Patrick Lichtsteiner, Christoph Posch, and Tobi Delbrück. "A 128x128 120 dB 15mus Latency Asynchronous Temporal Contrast Vision Sensor". In: *IEEE Journal of Solid-State Circuits* 43 (2008), pp. 566–576.

[102] Tsung-Yi Lin et al. "Microsoft coco: Common objects in context". In: *Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part V 13*. Springer. 2014, pp. 740–755.

[103] Hongjie Liu et al. "Design of a spatiotemporal correlation filter for event-based sensors". In: *2015 IEEE International Symposium on Circuits and Systems (ISCAS)*. IEEE. 2015, pp. 722–725.

[104] Renhao Liu et al. "Zero-Shot Learning of Continuous 3D Refractive Index Maps from Discrete Intensity-Only Measurements". In: *arXiv preprint arXiv:2112.00002* (2021).

[105] Ziji Liu et al. "Real-time brightfield, darkfield, and phase contrast imaging in a light-emitting diode array microscope". In: *Journal of biomedical optics* 19.10 (2014), p. 106002.

[106] Guolan Lu and Baowei Fei. "Medical hyperspectral imaging: a review". In: *Journal of biomedical optics* 19.1 (2014), pp. 010901–010901.

[107] Alice Lucas et al. "Using deep neural networks for inverse problems in imaging: beyond analytical methods". In: *IEEE Signal Processing Magazine* 35.1 (2018), pp. 20–36.

[108] Michael Lustig, David Donoho, and John M Pauly. "Sparse MRI: The application of compressed sensing for rapid MR imaging". In: *Magnetic Resonance in Medicine: An Official Journal of the International Society for Magnetic Resonance in Medicine* 58.6 (2007), pp. 1182–1195.

[109] Misha Mahowald and Misha Mahowald. "The silicon retina". In: *An Analog VLSI System for Stereoscopic Vision* (1994), pp. 4–65.

[110] Thomas Mangeat et al. "Super-resolved live-cell imaging using random illumination microscopy". In: *Cell Reports Methods* 1.1 (2021), p. 100009.

[111] Julien NP Martel et al. "Acorn: Adaptive coordinate networks for neural scene representation". In: *arXiv preprint arXiv:2105.02788* (2021).

[112] Alex Matlock and Lei Tian. "High-throughput, volumetric quantitative phase imaging with multiplexed intensity diffraction tomography". In: *Biomedical Optics Express* 10.12 (2019), p. 6432.

[113] Brian McReynolds et al. "Demystifying Event-based Sensor Biasing to Optimize Signal to Noise for Space Domain Awareness". In: *Advanced Maui Optical and Space Surveillance Technologies Conference (AMOS)*. University of Zurich. 2023.

[114] Shalin B Mehta and Colin JR Sheppard. "Quantitative phase-gradient imaging at high resolution with asymmetric illumination-based differential phase contrast". In: *Optics letters* 34.13 (2009), pp. 1924–1926.

[115] Ben Mildenhall et al. "Nerf: Representing scenes as neural radiance fields for view synthesis". In: *European conference on computer vision*. Springer. 2020, pp. 405–421.

[116] Ben Mildenhall et al. "Nerf: Representing scenes as neural radiance fields for view synthesis". In: *Communications of the ACM* 65.1 (2021), pp. 99–106.

[117] Michael J Mlodzianoski et al. "Sample drift correction in 3D fluorescence photoactivation localization microscopy". In: *Optics express* 19.16 (2011), pp. 15009–15019.

[118] Leonhard Möckl, Don C Lamb, and Christoph Bräuchle. "Super-resolved fluorescence microscopy: nobel prize in chemistry 2014 for eric betzig, stefan hell, and william e. moerner". In: *Angewandte Chemie International Edition* 53.51 (2014), pp. 13972–13977.

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

[120] Emeric Mudry et al. "Structured illumination microscopy using unknown speckle patterns". In: *Nature Photonics* 6.5 (2012), pp. 312–315.

[121] Marcel Müller et al. "Open-source image reconstruction of super-resolution structured illumination microscopy data in ImageJ". In: *Nature communications* 7.1 (2016), p. 10980.

[122] Thomas Müller et al. "Instant neural graphics primitives with a multiresolution hash encoding". In: *ACM Transactions on Graphics (ToG)* 41.4 (2022), pp. 1–15.

[123] Shree K Nayar and Tomoo Mitsunaga. "High dynamic range imaging: Spatially varying pixel exposures". In: *Proceedings IEEE Conference on Computer Vision and Pattern Recognition. CVPR 2000 (Cat. No. PR00662)*. Vol. 1. IEEE. 2000, pp. 472–479.

[124] Elias Nehme et al. "Deep-STORM: super-resolution single-molecule microscopy by deep learning". In: *Optica* 5.4 (2018), pp. 458–464.

[125] Anh Nguyen et al. "Plug & play generative networks: Conditional iterative generation of images in latent space". In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2017, pp. 4467–4477.

[126] Tan H Nguyen et al. "Halo-free phase contrast microscopy". In: *Scientific reports* 7 (2017), p. 44034.

[127] Tan H Nguyen et al. "Quantitative phase imaging with partially coherent illumination". In: *Optics letters* 39.19 (2014), pp. 5511–5514.

[128] Jonathon Nixon-Abell et al. "Increased spatiotemporal resolution reveals highly dynamic dense tubular matrices in the peripheral ER". In: *Science* 354.6311 (2016), aaf3928.

[129] Yuji Nozaki and Tobi Delbruck. "Temperature and parasitic photocurrent effects in dynamic vision sensors". In: *IEEE Transactions on Electron Devices* 64.8 (2017), pp. 3239–3245.

[130] Garrick Orchard et al. "Converting static image datasets to spiking neuromorphic datasets using saccades". In: *Frontiers in neuroscience* 9 (2015), p. 437.

[131] Vandana Padala, Arindam Basu, and Garrick Orchard. "A noise filtering algorithm for event-based asynchronous change detection image sensors on truenorth and its implementation on truenorth". In: *Frontiers in neuroscience* 12 (2018), p. 118.

[132] Liyuan Pan et al. "Bringing a blurry frame alive at high frame-rate with an event camera". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2019, pp. 6820–6829.

[133] Federico Paredes-Vallés and Guido CHE de Croon. "Back to event basics: Self-supervised learning of image reconstruction for event cameras via photometric constancy". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2021, pp. 3446–3455.

[134] Jeong Joon Park et al. "Deepsdf: Learning continuous signed distance functions for shape representation". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2019, pp. 165–174.

[135] Keunhong Park et al. "Nerfies: Deformable neural radiance fields". In: *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 2021, pp. 5865–5874.

[136] YongKeun Park, Christian Depeursinge, and Gabriel Popescu. "Quantitative phase imaging in biomedicine". In: *Nature photonics* 12.10 (2018), pp. 578–589.

[137] Adam Parslow, Albert Cardona, and Robert J Bryson-Richardson. "Sample drift correction following 4D confocal time-lapse imaging". In: *Journal of visualized experiments: JoVE* 86 (2014).

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

[139] Zachary F Phillips, Regina Eckert, and Laura Waller. "Quasi-dome: A self-calibrated high-na led illuminator for fourier ptychography". In: *Imaging Systems and Applications*. Optica Publishing Group. 2017, IW4E–5.

[140] Zachary F Phillips et al. "High-throughput fluorescence microscopy using multi-frame motion deblurring". In: *Biomedical Optics Express* 11.1 (2020), pp. 281–300.

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

[142] Gabriel Popescu. *Quantitative phase imaging of cells and tissues*. McGraw Hill Professional, 2011.

[143] Martin Priessner et al. "Content-aware frame interpolation (CAFI): Deep Learning-based temporal super-resolution for fast bioimaging". In: *Nature Methods* (2024), pp. 1–9.

[144] Albert Pumarola et al. "D-nerf: Neural radiance fields for dynamic scenes". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2021, pp. 10318–10327.

[145] Chang Qiao et al. "Evaluation and development of deep neural networks for image super-resolution in optical microscopy". In: *Nature Methods* 18.2 (2021), pp. 194–202.

[146] Maziar Raissi, Paris Perdikaris, and George E Karniadakis. "Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations". In: *Journal of Computational physics* 378 (2019), pp. 686–707.

[147] Henri Rebecq, Daniel Gehrig, and Davide Scaramuzza. "ESIM: an open event camera simulator". In: *Conference on robot learning*. PMLR. 2018, pp. 969–982.

[148] Henri Rebecq et al. "High speed and high dynamic range video with an event camera". In: *IEEE transactions on pattern analysis and machine intelligence* 43.6 (2019), pp. 1964–1980.

[149] John M Rodenburg. "Ptychography and related diffractive imaging methods". In: *Advances in imaging and electron physics* 150 (2008), pp. 87–184.

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

[151] Alon Saguy et al. "DBlink: Dynamic localization microscopy in super spatiotemporal resolution via deep learning". In: *Nature Methods* (2023), pp. 1–10.

[152] Cedric Scheerlinck, Nick Barnes, and Robert Mahony. "Continuous-time intensity estimation using event cameras". In: *Asian Conference on Computer Vision*. Springer. 2018, pp. 308–324.

[153] Cedric Scheerlinck et al. "Fast Image Reconstruction with an Event Camera". In: *IEEE Winter Conf. Appl. Comput. Vis. (WACV)*. 2020, pp. 156–163.

[154] Teresa Serrano-Gotarredona and Bernabé Linares-Barranco. "Poker-DVS and MNIST-DVS. Their history, how they were made, and other details". In: *Frontiers in neuroscience* 9 (2015), p. 481.

[155] Alexey Sharonov and Robin M Hochstrasser. "Wide-field subdiffraction imaging by accumulated binding of diffusing probes". In: *Proceedings of the National Academy of Sciences* 103.50 (2006), pp. 18911–18916.

[156] Colin JR Sheppard. "Defocused transfer function for a partially coherent microscope and application to phase retrieval". In: *JOSA A* 21.5 (2004), pp. 828–831.

[157] Sapna A Shroff, James R Fienup, and David R Williams. "Lateral superresolution using a posteriori phase shift estimation for a moving object: experimental results". In: *JOSA A* 27.8 (2010), pp. 1770–1782.

[158] Yaron M Sigal, Ruobo Zhou, and Xiaowei Zhuang. "Visualizing and discovering cellular structures with super-resolution microscopy". In: *Science* 361.6405 (2018), pp. 880–887.

[159] Vincent Sitzmann, Michael Zollhöfer, and Gordon Wetzstein. "Scene representation networks: Continuous 3d-structure-aware neural scene representations". In: *Advances in Neural Information Processing Systems* 32 (2019).

[160] Vincent Sitzmann et al. "Implicit neural representations with periodic activation functions". In: *Advances in neural information processing systems* 33 (2020), pp. 7462–7473.

[161] Nicholas Sofroniew et al. *napari: a multi-dimensional image viewer for Python*. 2024. DOI: 10.5281/ZENODO.3555620. URL: https://zenodo.org/doi/10.5281/zenodo.3555620.

[162] Juan M Soto, José A Rodrigo, and Tatiana Alieva. "Label-free quantitative 3D tomographic imaging for partially coherent light microscopy". In: *Optics express* 25.14 (2017), pp. 15699–15712.

[163] Artur Speiser et al. "Deep learning enables fast and dense single-molecule localization with high accuracy". In: *Nature methods* 18.9 (2021), pp. 1082–1090.

[164] Kenneth O Stanley. "Compositional pattern producing networks: A novel abstraction of development". In: *Genetic programming and evolvable machines* 8 (2007), pp. 131–162.

[165] Timo Stoffregen et al. "Reducing the sim-to-real gap for event cameras". In: *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XXVII 16*. Springer. 2020, pp. 534–549.

[166] N Streibl. "Depth transfer by an imaging system". In: *Optica Acta: International Journal of Optics* 31.11 (1984), pp. 1233–1241.

[167] Norbert Streibl. "Three-dimensional imaging by a microscope". In: *JOSA A* 2.2 (1985), pp. 121–127.

[168] Nina Sultanova, S Kasarova, and Ivan Nikolov. "Dispersion proper ties of optical polymers". In: *Acta Physica Polonica-Series A General Physics* 116.4 (2009), p. 585.

[169] Yu Sun et al. "Coil: Coordinate-based internal learning for tomographic imaging". In: *IEEE Transactions on Computational Imaging* 7 (2021), pp. 1400–1412.

[170] Yu Sun et al. "Scalable plug-and-play ADMM with convergence guarantees". In: *IEEE Transactions on Computational Imaging* 7 (2021), pp. 849–863.

[171] Yongjin Sung et al. "Optical diffraction tomography for high resolution live cell imaging". In: *Optics express* 17.1 (2009), pp. 266–277.

[172] Matthew Tancik et al. "Fourier features let networks learn high frequency functions in low dimensional domains". In: *Advances in Neural Information Processing Systems* 33 (2020), pp. 7537–7547.

[173] Matthew Tancik et al. "Learned initializations for optimizing coordinate-based neural representations". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2021, pp. 2846–2855.

[174] Lei Tian and Laura Waller. "3D intensity and phase imaging from light field measurements in an LED array microscope". In: *optica* 2.2 (2015), pp. 104–111.

[175] Lei Tian and Laura Waller. "Quantitative differential phase contrast imaging in an LED array microscope". In: *Optics express* 23.9 (2015), pp. 11394–11403.

[176] Lei Tian et al. "Computational illumination for high-speed in vitro Fourier ptychographic microscopy". In: *Optica* 2.10 (2015), pp. 904–911.

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

[178] Alex Trevithick et al. "Real-time radiance fields for single-image portrait view synthesis". In: *ACM Transactions on Graphics (TOG)* 42.4 (2023), pp. 1–15.

[179] Stepan Tulyakov et al. "Time lens: Event-based video frame interpolation". In: *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 2021, pp. 16155–16164.

[180] Stepan Tulyakov et al. "Time lens++: Event-based frame interpolation with parametric non-linear flow and multi-scale fusion". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2022, pp. 17755–17764.

[181] Raphaël Turcotte et al. "Dynamic super-resolution structured illumination imaging in the living brain". In: *Proceedings of the National Academy of Sciences* 116.19 (2019), pp. 9586–9591.

[182] Laura Waller. *iPhone propeller effect (aliasing)*. URL: `http://www.laurawaller.com/opticsfun/iPhoneAliasing.htm`.

[183] Laura Waller, Lei Tian, and George Barbastathis. "Transport of intensity phase-amplitude imaging with higher order intensity derivatives". In: *Optics express* 18.12 (2010), pp. 12552–12561.

[184] Laura Waller et al. "Phase from chromatic aberrations". In: *Optics express* 18.22 (2010), pp. 22817–22825.

[185] Hui-Wen Lu-Walther et al. "fastSIM: a practical implementation of fast structured illumination microscopy". In: *Methods and Applications in Fluorescence* 3.1 (2015), p. 014001.

[186] Bishan Wang et al. "Event enhanced high-quality image recovery". In: *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XIII 16*. Springer. 2020, pp. 155–171.

[187] Zhuo Wang et al. "Spatial light interference microscopy (SLIM)". In: *Optics express* 19.2 (2011), pp. 1016–1026.

[188] Zihao W Wang et al. "Event-driven video frame synthesis". In: *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV) Workshops*. Oct. 2019.

[189] Ziwei Wang et al. "An asynchronous kalman filter for hybrid event cameras". In: *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 2021, pp. 448–457.

[190] Ziyun Wang et al. "Event-based Continuous Color Video Decompression from Single Frames". In: *arXiv preprint arXiv:2312.00113* (2023).

[191] Markus Weiger et al. "Motion-adapted gating based on k-space weighting for reduction of respiratory motion artifacts". In: *Magnetic resonance in medicine* 38.2 (1997), pp. 322–333.

[192] Martin Weigert et al. "Content-aware image restoration: pushing the limits of fluorescence microscopy". In: *Nature methods* 15.12 (2018), pp. 1090–1097.

[193] Wenming Weng, Yueyi Zhang, and Zhiwei Xiong. "Event-based video reconstruction using transformer". In: *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 2021, pp. 2563–2572.

[194] Kai Wicker. "Non-iterative determination of pattern phase in structured illumination microscopy using auto-correlations in Fourier space". In: *Optics express* 21.21 (2013), pp. 24692–24701.

[195] Tony Wilson and Colin JR Sheppard. "The halo effect of image processing by spatial frequency filtering". In: *Optik* 59.1 (1981), pp. 19–23.

[196] Emil Wolf. "Three-dimensional structure determination of semi-transparent objects from holographic data". In: *Optics communications* 1.4 (1969), pp. 153–156.

[197] Jelmer M Wolterink, Jesse C Zwienenberg, and Christoph Brune. "Implicit neural representations for deformable image registration". In: *International Conference on Medical Imaging with Deep Learning*. PMLR. 2022, pp. 1349–1359.

[198] Yichen Wu et al. "Three-dimensional virtual refocusing of fluorescence microscopy images using deep learning". In: *Nature methods* 16.12 (2019), pp. 1323–1331.

[199] Yicong Wu and Hari Shroff. "Faster, sharper, and deeper: structured illumination microscopy for biological imaging". In: *Nature methods* 15.12 (2018), pp. 1011–1019.

[200] Zihui Wu et al. "Simba: Scalable inversion in optical tomography using deep denoising priors". In: *IEEE Journal of Selected Topics in Signal Processing* 14.6 (2020), pp. 1163–1175.

[201] Li-Hao Yeh, Shwetadwip Chowdhury, and Laura Waller. "Computational structured illumination for high-content fluorescence and phase microscopy". In: *Biomedical optics express* 10.4 (2019), pp. 1978–1998.

[202] Li-Hao Yeh, Lei Tian, and Laura Waller. "Structured illumination microscopy with unknown patterns and a statistical prior". In: *Biomedical optics express* 8.2 (2017), pp. 695–711.

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

[204] Andrew G York et al. "Instant super-resolution imaging in live cells and embryos via analog image processing". In: *Nature methods* 10.11 (2013), pp. 1122–1126.

[205] Alex Yu et al. "Plenoxels: Radiance Fields without Neural Networks". In: *arXiv preprint arXiv:2112.05131* (2021).

[206] Tong Yu and Hong Zhu. "Hyper-parameter optimization: A review of algorithms and applications". In: *arXiv preprint arXiv:2003.05689* (2020).

[207] Maxim Zaitsev, Julian Maclaren, and Michael Herbst. "Motion artifacts in MRI: A complex problem with many partial solutions". In: *Journal of Magnetic Resonance Imaging* 42.4 (2015), pp. 887–901.

[208] Kai Zhang et al. "Plug-and-play image restoration with deep denoiser prior". In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 44.10 (2021), pp. 6360–6376.

[209] Richard Zhang et al. "The Unreasonable Effectiveness of Deep Features as a Perceptual Metric". In: *CVPR*. 2018.

[210] Xiang Zhang and Lei Yu. "Unifying motion deblurring and frame interpolation with events". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2022, pp. 17765–17774.

[211] Guoan Zheng, Roarke Horstmeyer, and Changhuei Yang. "Wide-field, high-resolution Fourier ptychographic microscopy". In: *Nature photonics* 7.9 (2013), pp. 739–745.

[212] Guoan Zheng, Christopher Kolner, and Changhuei Yang. "Microscopy refocusing and dark-field imaging by using a simple LED array". In: *Optics letters* 36.20 (2011), pp. 3987–3989.

[213] Jingshan Zhong et al. "Nonlinear optimization algorithm for partially coherent phase retrieval and source recovery". In: *IEEE Transactions on Computational Imaging* 2.3 (2016), pp. 310–322.

[214] Alex Zihao Zhu et al. "Eventgan: Leveraging large scale image datasets for event cameras". In: *2021 IEEE International Conference on Computational Photography (ICCP)*. IEEE. 2021, pp. 1–11.

[215] Lin Zhu et al. "Neuspike-net: High speed video reconstruction via bio-inspired neuromorphic cameras". In: *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 2021, pp. 2400–2409.

[216] Yunhao Zou et al. "Learning to reconstruct high speed and high dynamic range videos from events". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2021, pp. 2024–2033.