# Computational Imaging with Event Cameras

Chris Metzler

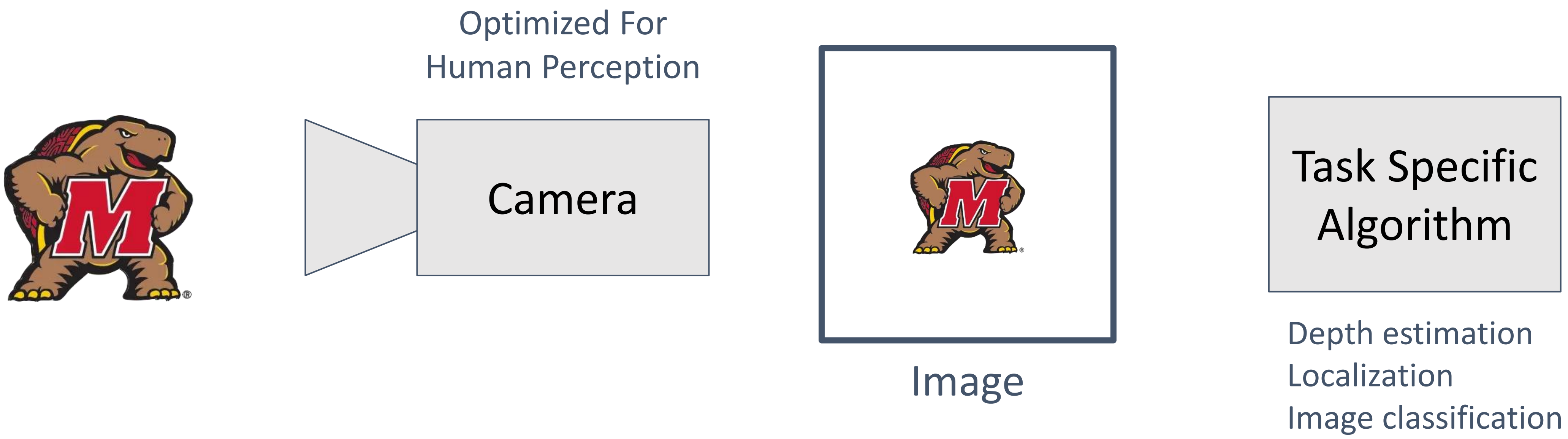
UNIVERSITY OF MARYLAND

PRG Perception & Robotics Group

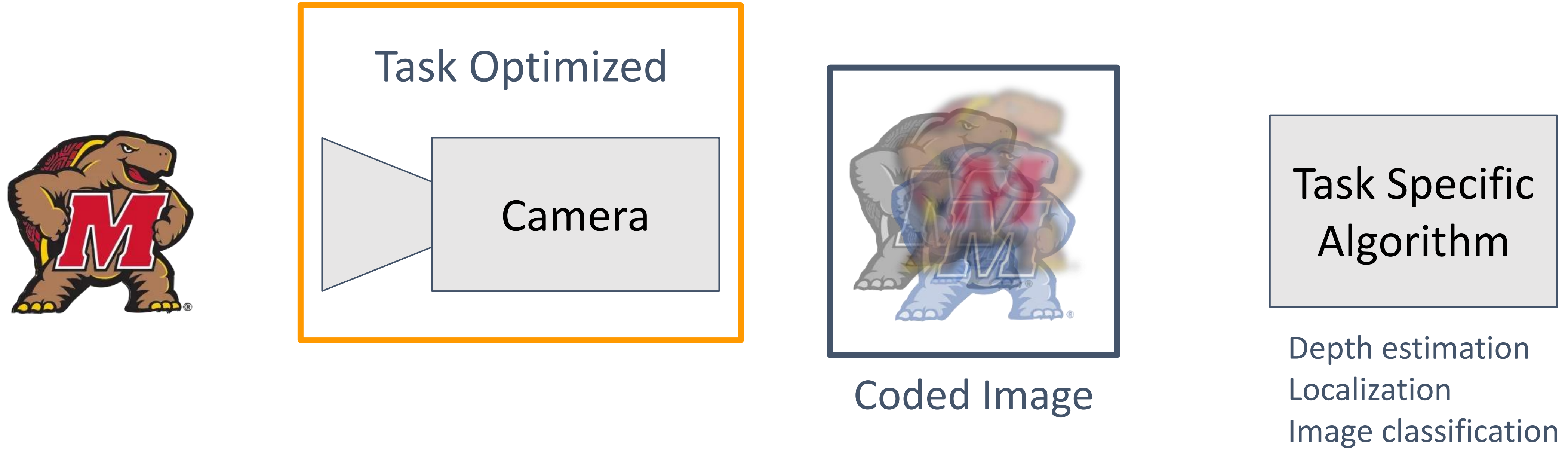
UMD Intelligent Sensing Lab

*Computational Imaging is the*
*Co-Design of Optics and Algorithms*

# Traditional Computer Vision



# Computational Imaging



# Computational Imaging in Nature

Polarization

Multi-focus

High dynamic range

Ultra-violet

[National Geographic]

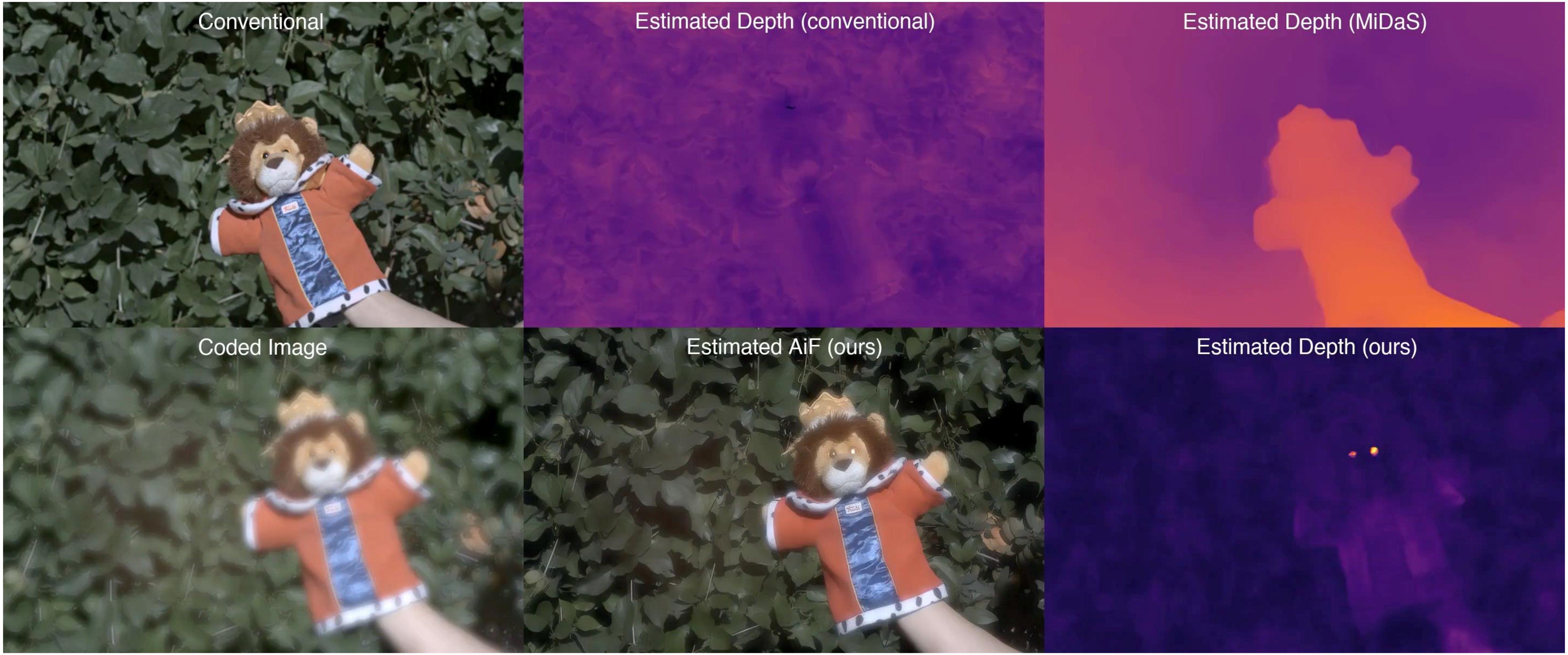
# Depth Estimation with Computational Imaging

Conventional

Coded Image


[Ikoma et al. 2021]

# Depth Estimation with Computational Imaging




[Ikoma et al. 2021]

# Computational Imaging for …

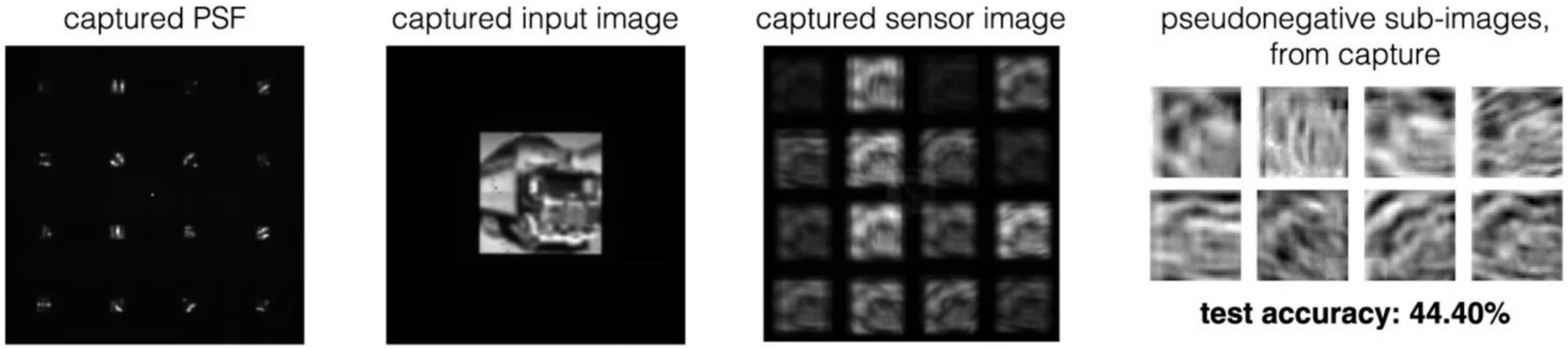
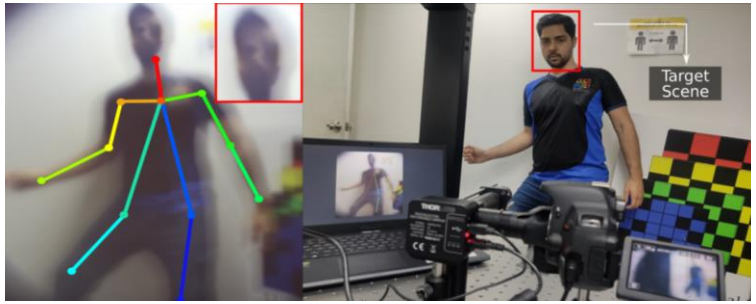


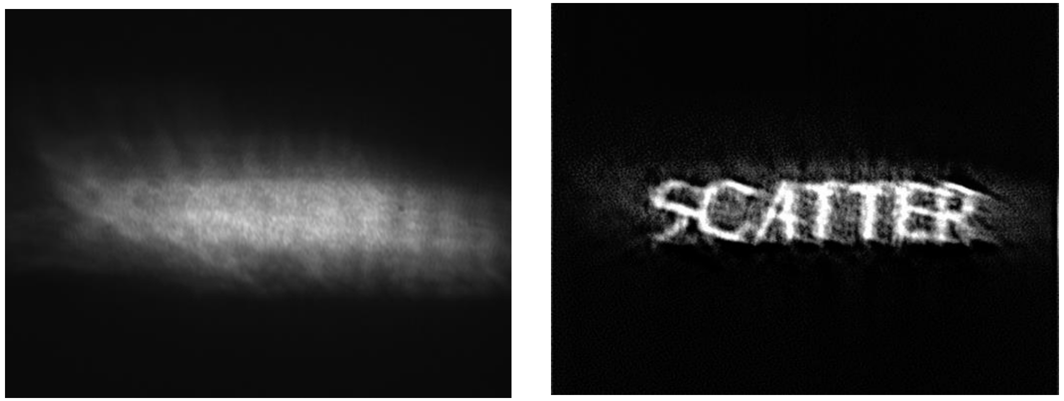

Image classification
(Chang et al. 2018)

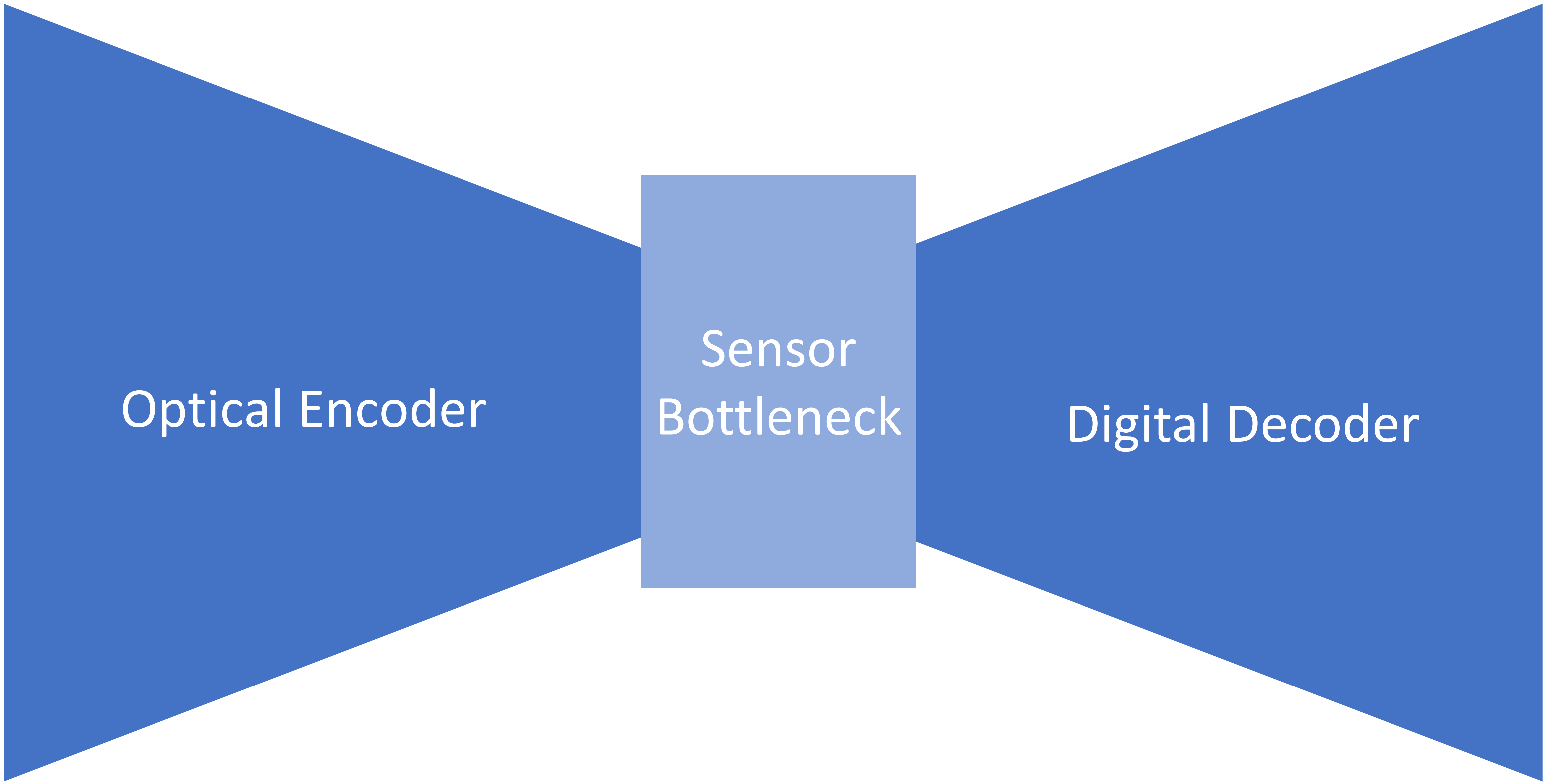
High dynamic range
(M. et al. 2020)



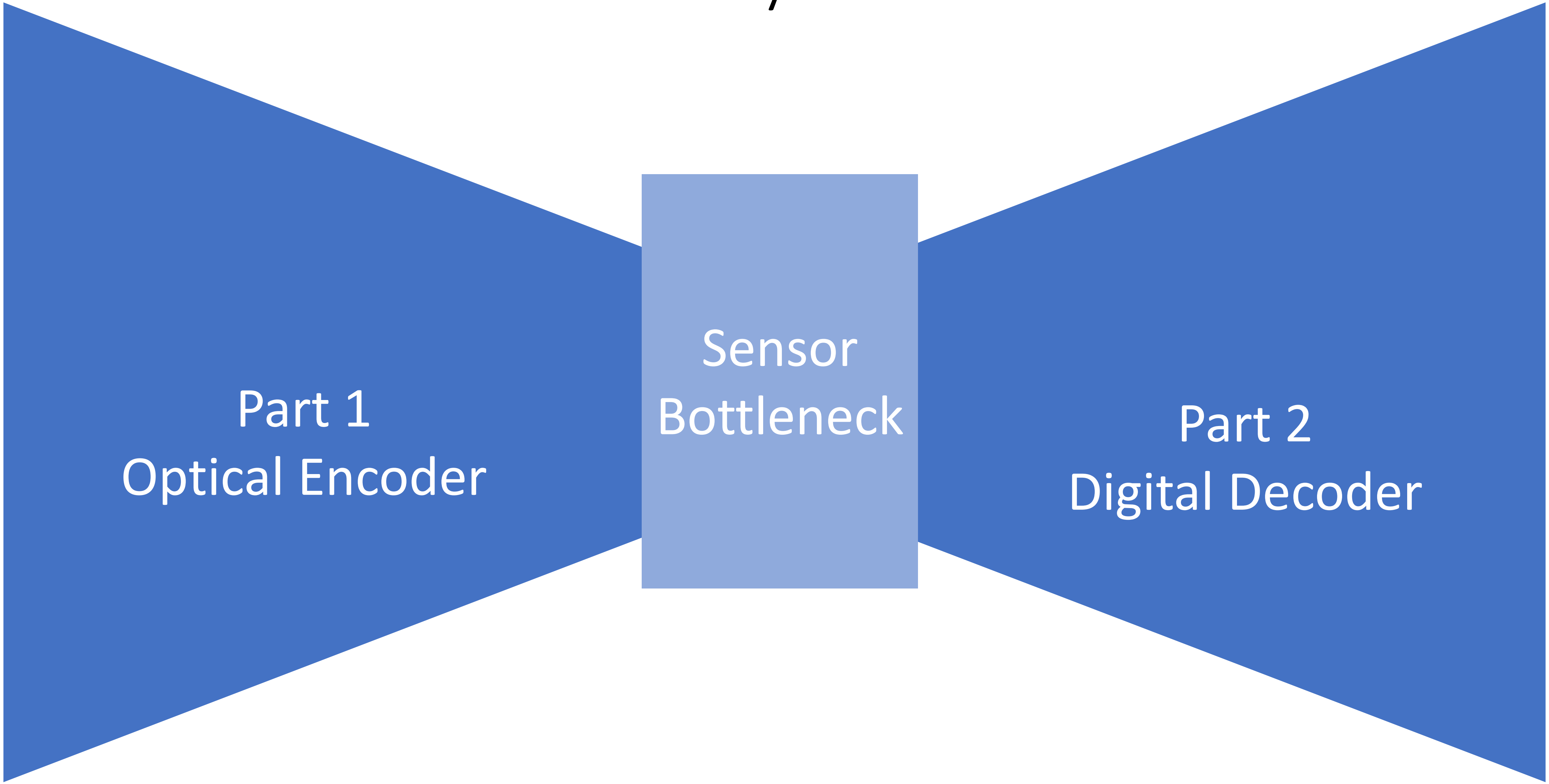

Privacy preservation
(Hinojosa et al. 2021)

Seeing through obstructions
(Shi et al. 2022, Xie et al. 2024)




Optical Encoder
Sensor
Bottleneck
Digital Decoder

# Todays Talk:


Part 1
Optical Encoder
Sensor
Bottleneck
Part 2
Digital Decoder

# Todays Talk:

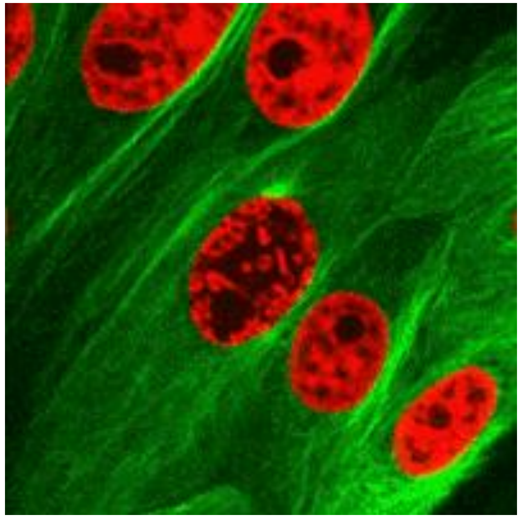
Part 1
Optical Encoder

Sensor
Bottleneck

Part 2
Digital Decoder

Sachin Shah

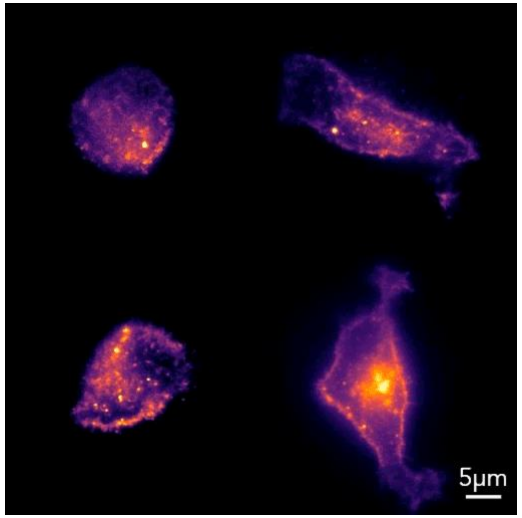
# Our Goal: Passive 3D Sensing at 1000+ fps


[Agarwal et al. 2018]


[National Geography]


[DJI]


[US Navy 2014]



[Knowles and Mahmood]

[PBS 2016]


[SpaceX 2019]

# One Option: Stereo Event Camera Systems

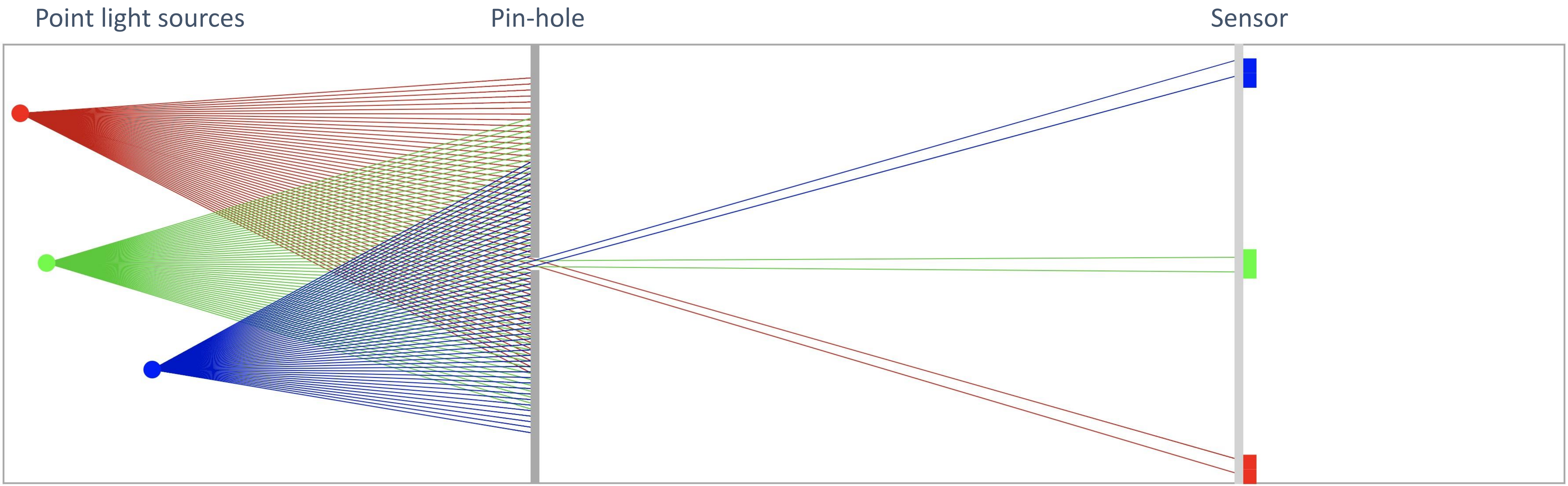
- Expensive
- Hard to synchronize
- Bulky

Image credit: Zhu et al. 2018

# Alterative: Event Camera with Coded Optics

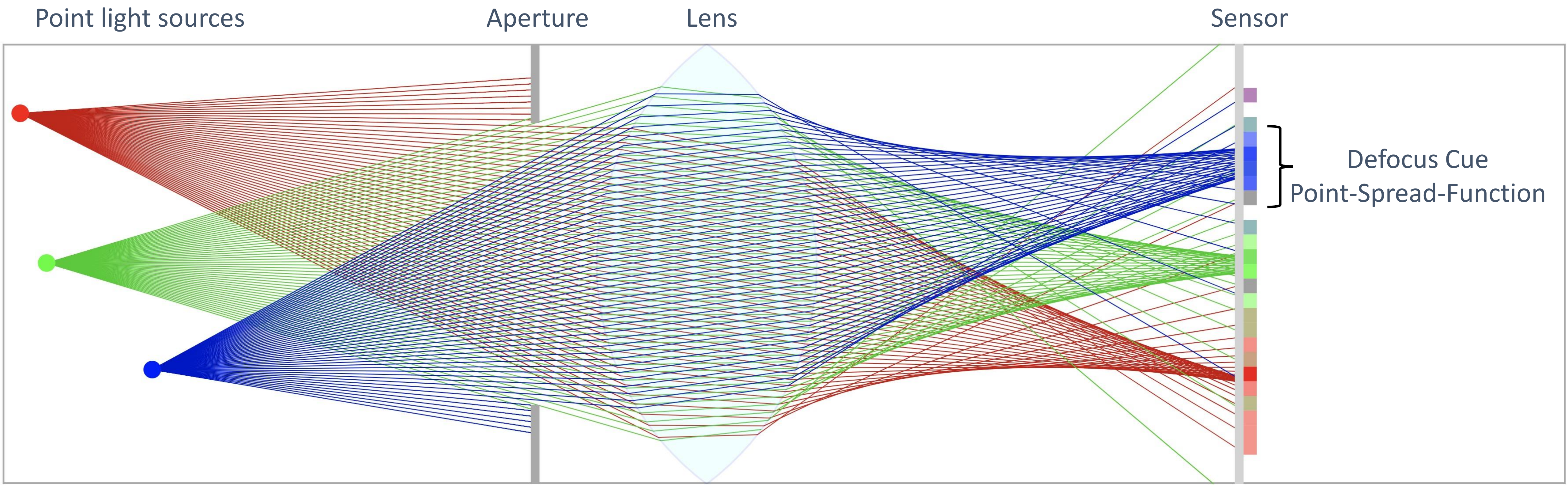

- ~~Expensive~~
- ~~Hard to synchronize~~
- ~~Bulky~~

# How a pinhole camera works




https://www.cs.umd.edu/~shah2022/optics/

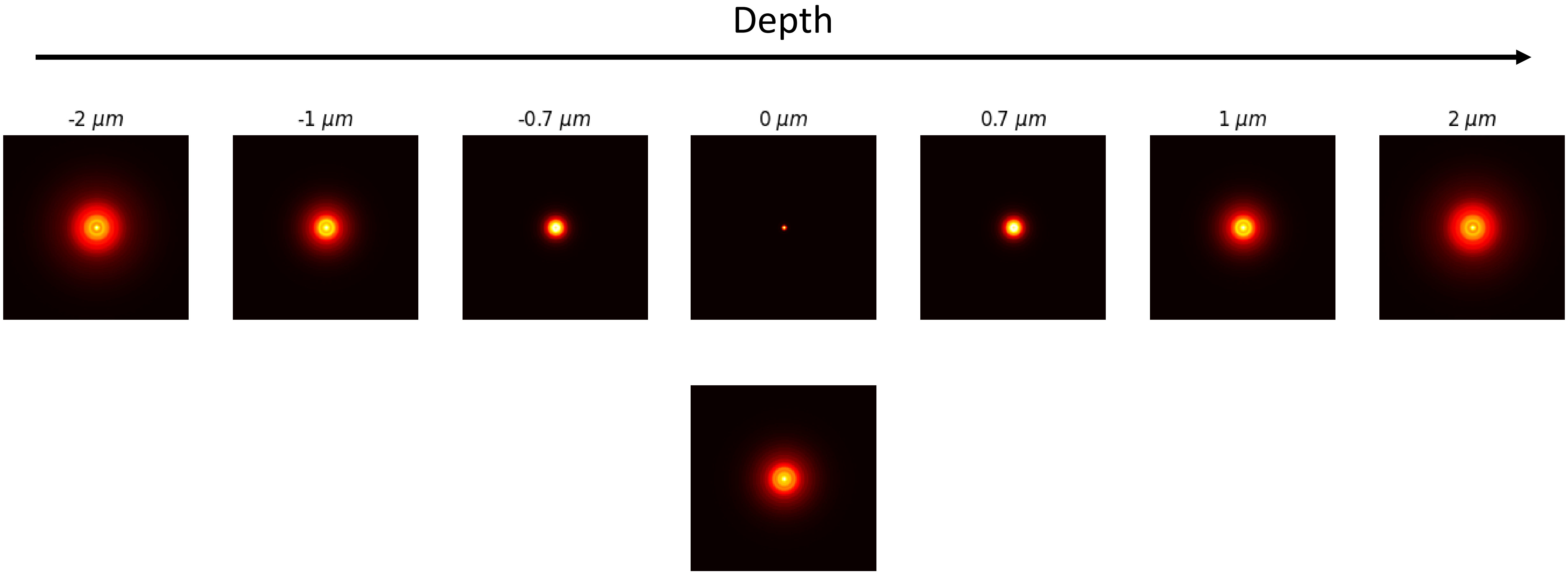
# How a real camera works




Cameras naturally encode some depth information

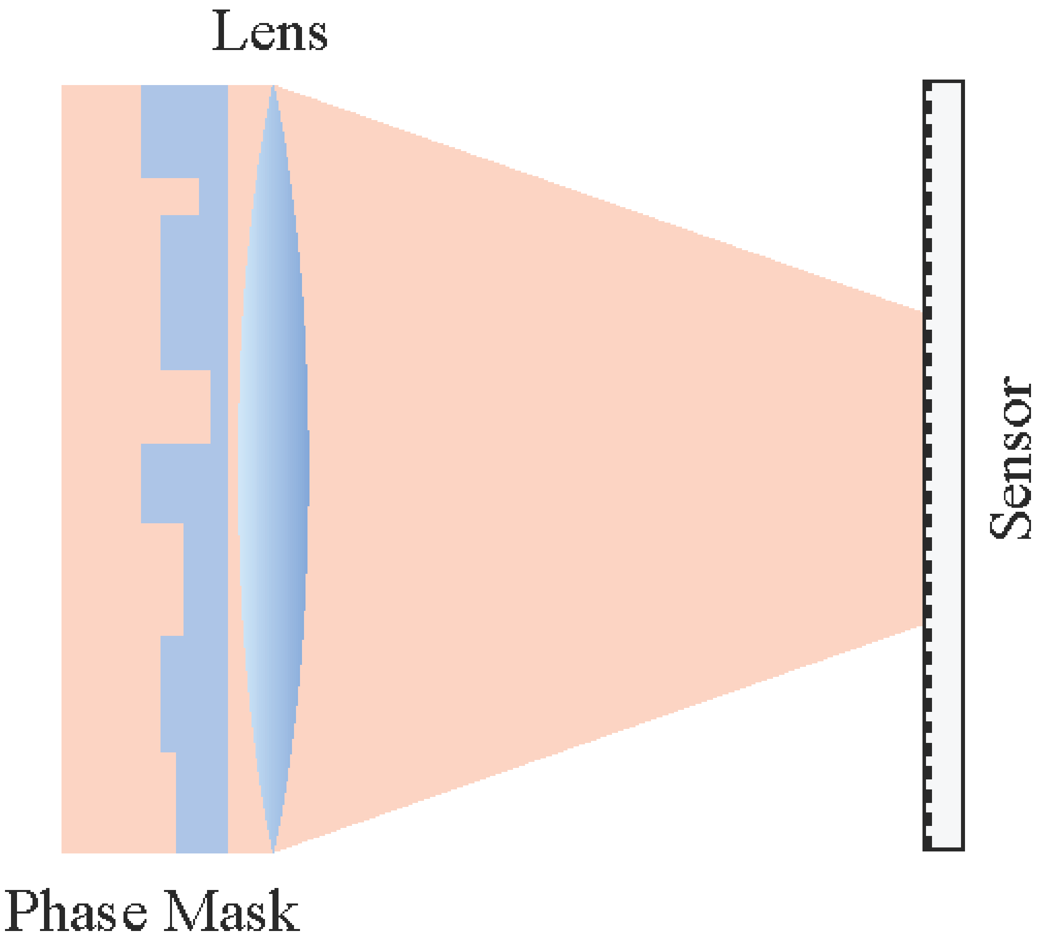
https://www.cs.umd.edu/~shah2022/optics/

# Estimating Depth From Defocus Cues



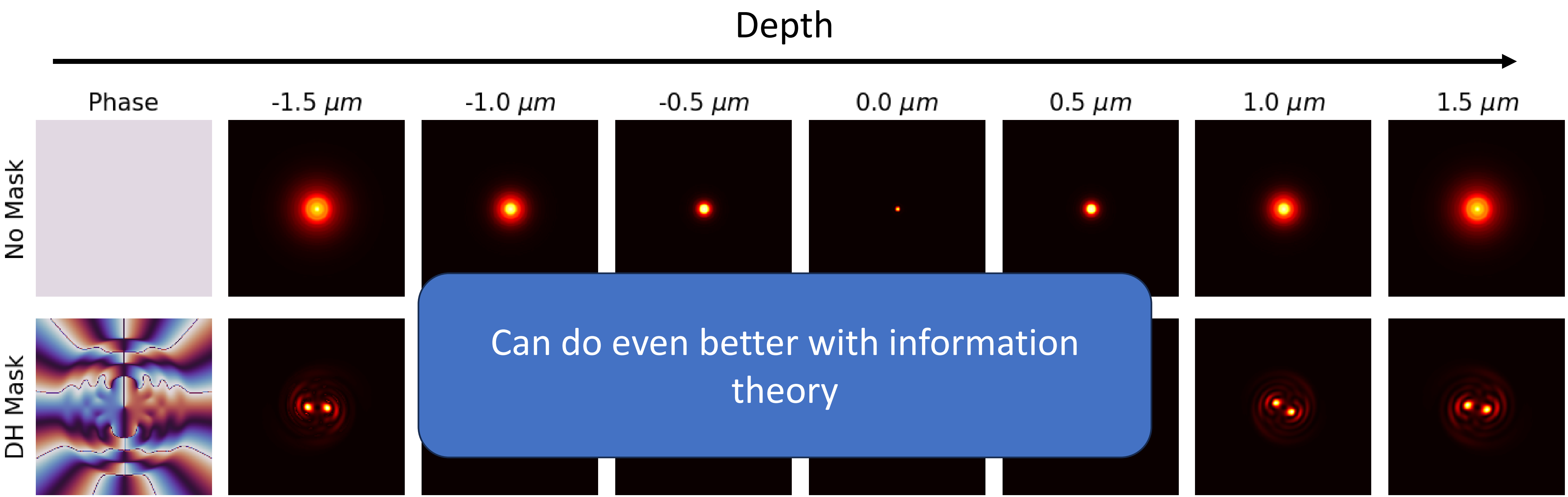

Depth is challenging to estimate with conventional optical systems

By introducing a phase mask into the optics, we can change the shape of the PSF




We can use elaborate defocus cues to “encode” depth into the images

# Double-Helix PSF




Depth is easy to estimate with DH optical systems

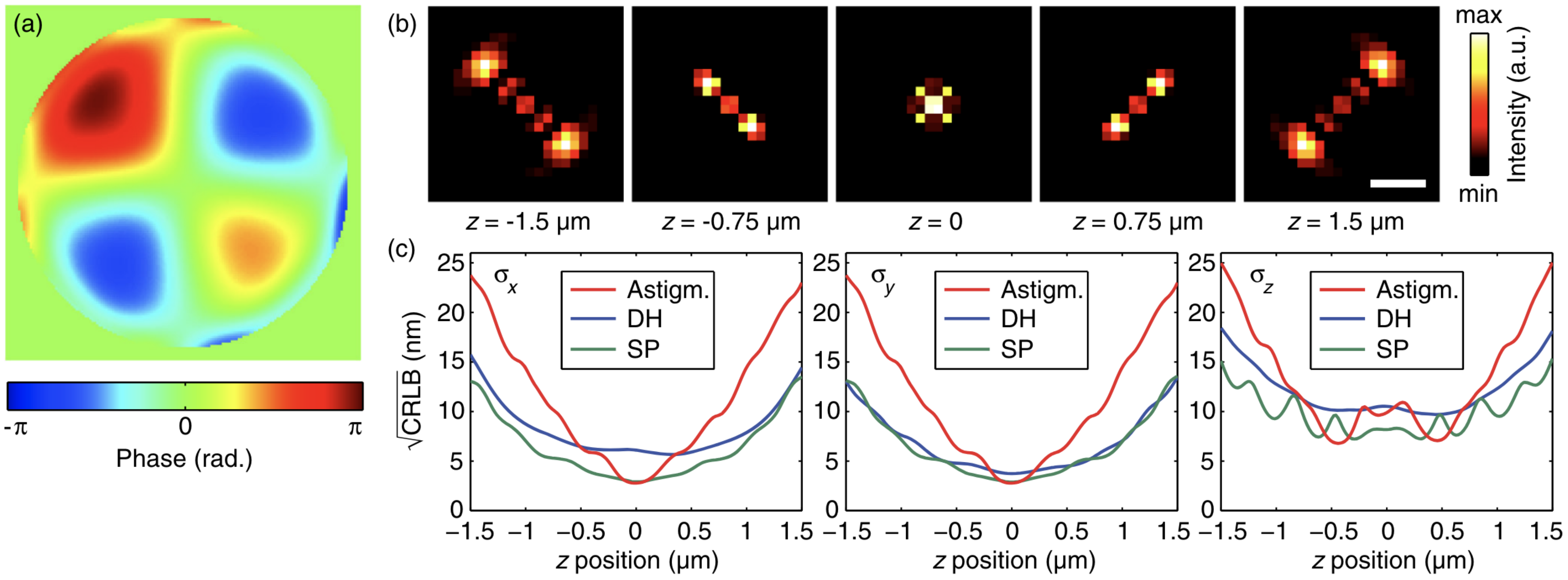
[Pavani et al. 2009]

# Designing a Phase Mask for a Conventional Camera

- Have a point source at location $\mathrm{x} = [x, y, z]^t$
- Observe $I = \textbf{Poisson}\left(h_\phi(\mathrm{x})\right)$, where the PSF $h$ is function of the phase mask $\phi$
- Construct the Fisher Information matrix associated with estimating $\mathrm{x}$
    - Error of maximum likelihood estimator of $\mathrm{x}$ is bounded by reciprocal of Fisher Information
- Design an optimal PSF by maximizing Fisher Information wrt $\phi$

[Shechtman et al. 2014]

# Designing a Phase Mask for a Conventional Camera




[Shechtman et al. 2014]

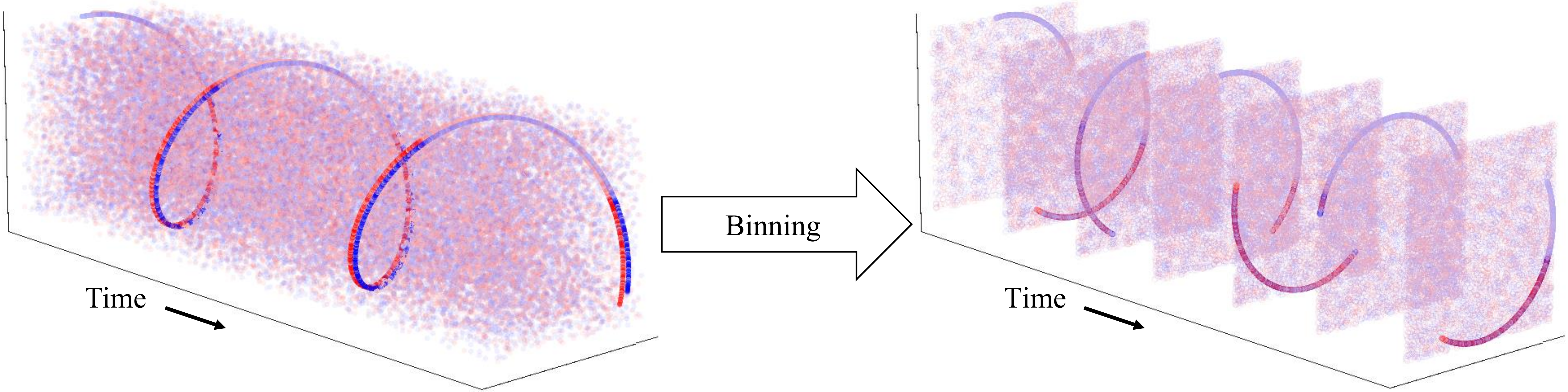
Can we extend this approach to event cameras?

# Designing a Phase Mask for an Event Camera

- Have a point source at location $\mathrm{x} = [x, y, z]^t$ moving with velocity $\Delta\mathrm{x} = [\Delta x, \Delta y, \Delta z]^t$
- Observe $\cancel{I = \mathrm{Poisson}\left(h_\phi(\mathrm{x})\right)}$, where the PSF $h$ is function of the phase mask $\phi$
- Construct the Fisher Information matrix associated with estimating $\mathrm{x}$
  - Error of maximum likelihood estimator of $\mathrm{x}$ is bounded by reciprocal of Fisher Information
- Design an optimal PSF by maximizing Fisher Information wrt $\phi$

[Shechtman et al. 2014]

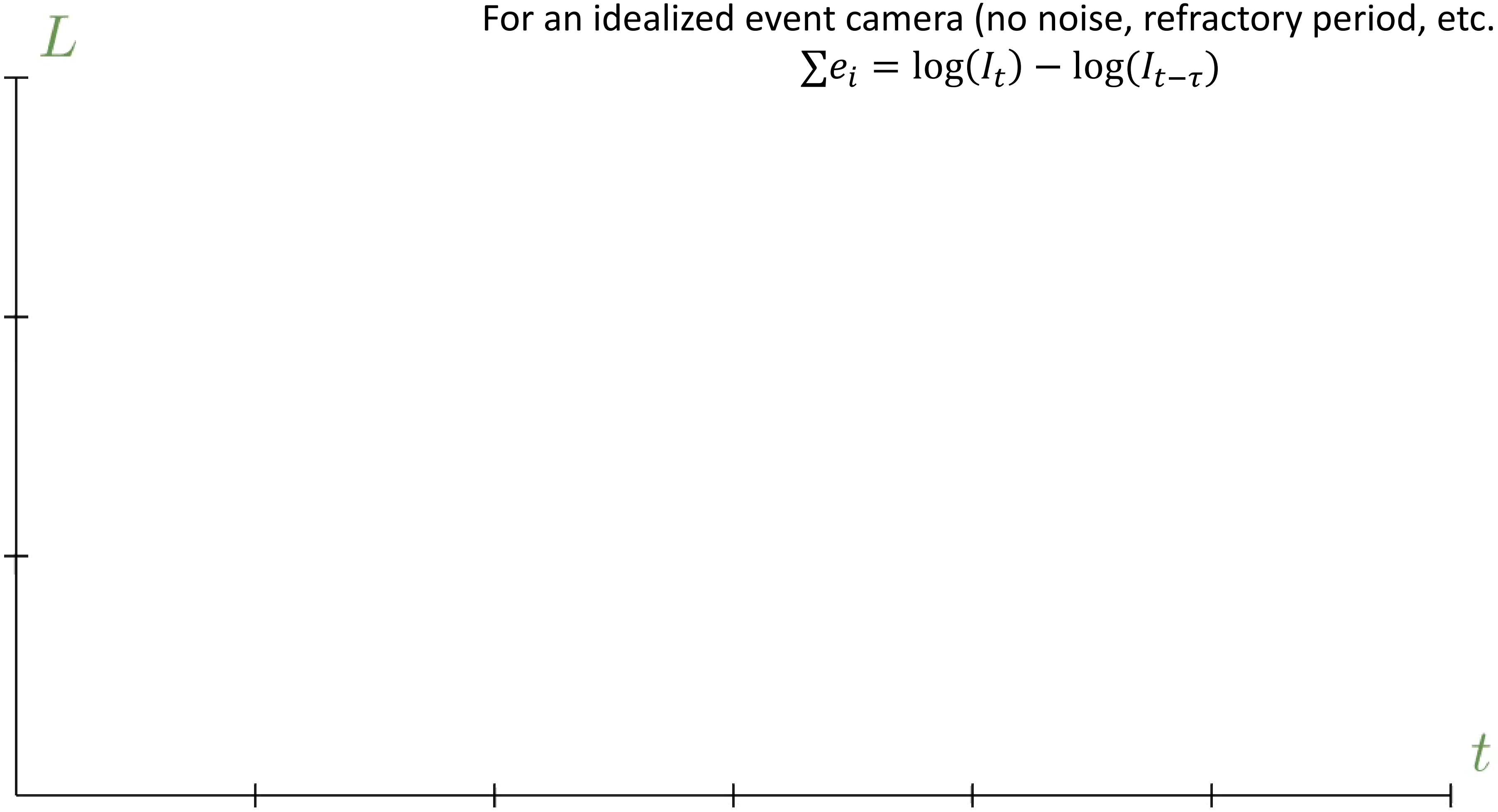
# Binning Events



# Binning Events

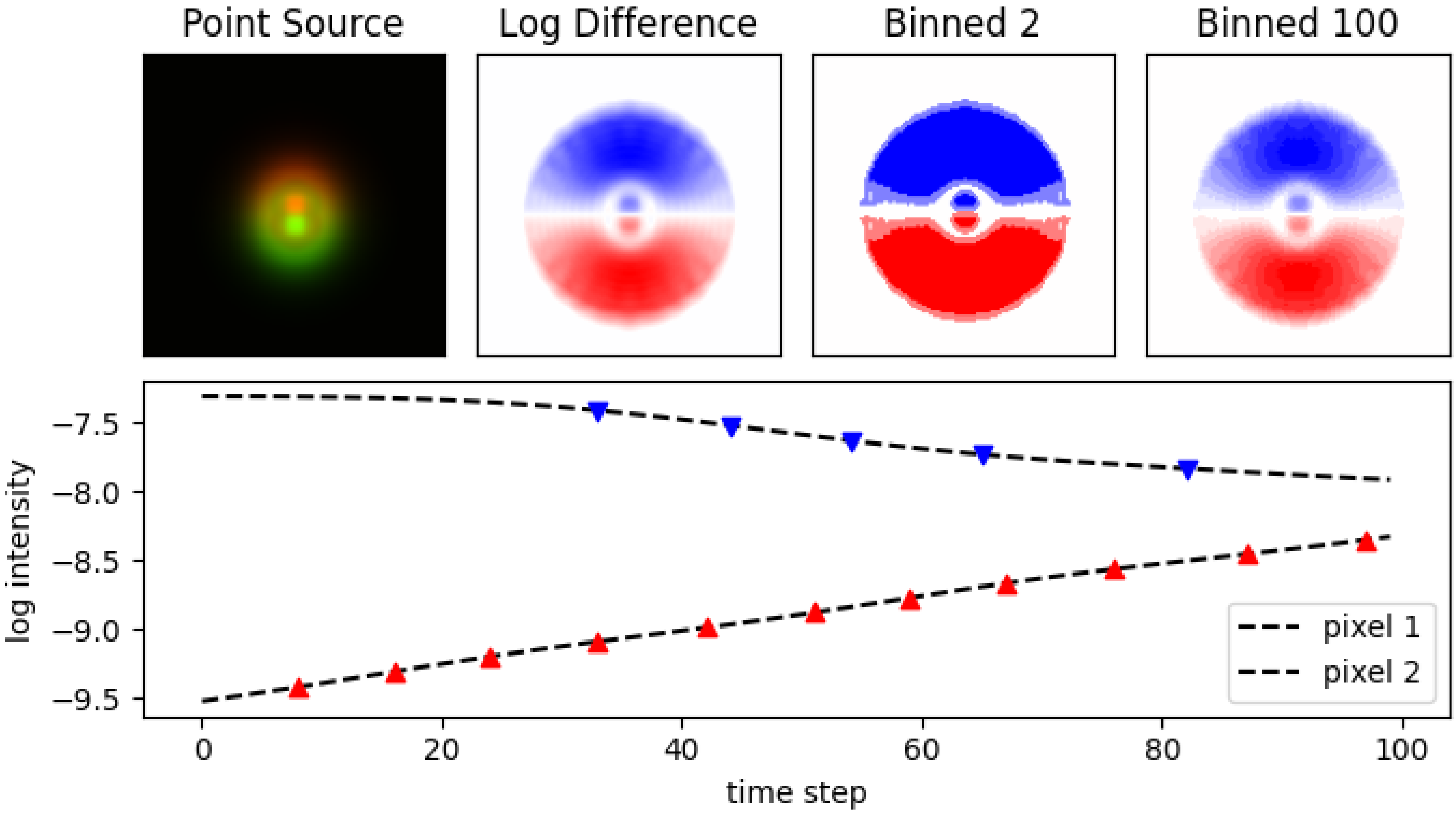
For an idealized event camera (no noise, refractory period, etc.):

$$\sum e_i = \log(I_t) - \log(I_{t-\tau})$$



# Binning Events

Binned events = Log difference between frames



# Theory: Stationary Flashing Point Source

$$\begin{aligned} M &= \log\left(I_t\right) - \log\left(I_{t-\tau}\right) \\ &= \log\left(I_t\right) \\ e^M &= I_t \sim \mathrm{Poisson}\left(\lambda = PSF\right) \end{aligned}$$

Key Finding: For blinking fluorescent molecules, the Fisher PSF is already optimal!

# Theory: Generalization

$$
\begin{aligned}
M &= \log\left(I_t\right) - \log\left(I_{t-\tau}\right) \\
&= \log\left(\frac{I_t}{I_{t-\tau}}\right) \\
e^M &= \frac{I_t}{I_{t-\tau}} \sim \frac{\mathrm{Poisson}\left(\lambda_t\right)}{\mathrm{Poisson}\left(\lambda_{t-\tau}\right)} \\
&\sim \mathcal{N}\left(\frac{\lambda_t}{\lambda_{t-\tau}}, \frac{\lambda_t}{\lambda_{t-\tau}^2} + \frac{\lambda_t^2}{\lambda_{t-\tau}^3}\right)
\end{aligned}
$$

$$\mathcal{I}(\theta) = \sum_{n}^{N} \frac{\mathcal{D}^T \mathcal{D}}{2\left(\mu+\nu\right)^2} \odot \begin{bmatrix} a & a & a & b & b & b \\ a & a & a & b & b & b \\ a & a & a & b & b & b \\ b & b & b & c & c & c \\ b & b & b & c & c & c \\ b & b & b & c & c & c \end{bmatrix}$$

Challenge #1: Highly non-convex wrt lens parameters

Challenge #2: Depends on particle position and motion

$$a = 2\mu^2\nu + 4\mu^2 + 2\mu\nu^2 + 12\mu\nu + 9\nu^2$$

$$b = -\left(2\mu^2\nu + 2\mu^2 + 2\mu\nu^2 + 7\mu\nu + 6\nu^2\right)$$

$$c = 2\mu^2\nu + \mu^2 + 2\mu\nu^2 + 4\mu\nu + 4\nu^2$$

$$\mathcal{I}(\theta) = \sum_{n}^{N} \frac{\mathcal{D}^T \mathcal{D}}{2\left(\mu+\nu\right)^2} \odot \begin{bmatrix} a & a & a & b & b & b \\ a & a & a & b & b & b \\ a & a & a & b & b & b \\ b & b & b & c & c & c \\ b & b & b & c & c & c \\ b & b & b & c & c & c \end{bmatrix}$$

Challenge #1: Highly non-convex wrt lens parameters
Solution #1: Regularize with INRs

Challenge #2: Depends on particle position and motion
Solution #2: Monte Carlo averaging

$$a = 2\mu^2\nu + 4\mu^2 + 2\mu\nu^2 + 12\mu\nu + 9\nu^2$$

$$b = -\left(2\mu^2\nu + 2\mu^2 + 2\mu\nu^2 + 7\mu\nu + 6\nu^2\right)$$

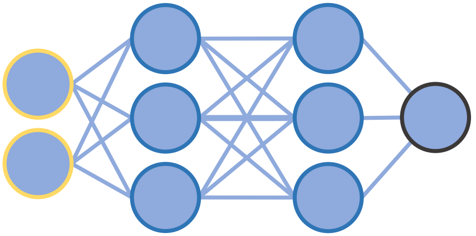
$$c = 2\mu^2\nu + \mu^2 + 2\mu\nu^2 + 4\mu\nu + 4\nu^2$$

# Implicit Neural Representations

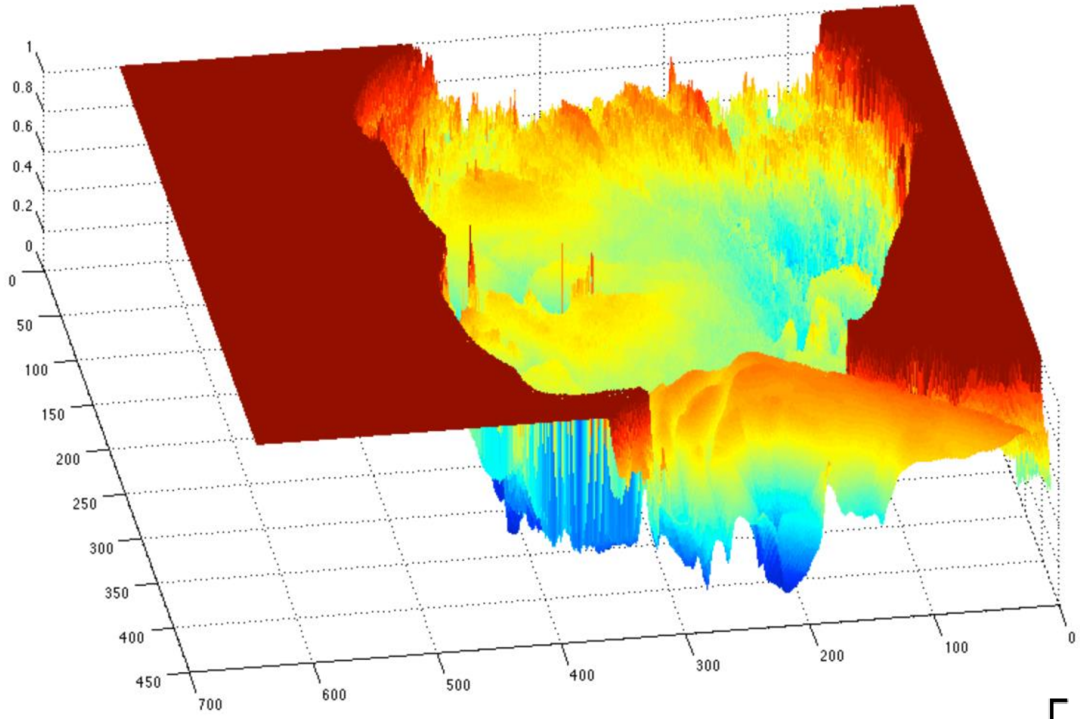
Form *functional* representations of images (or phase masks)

E.g., a (grayscale) image is a 2D function

grayscale image



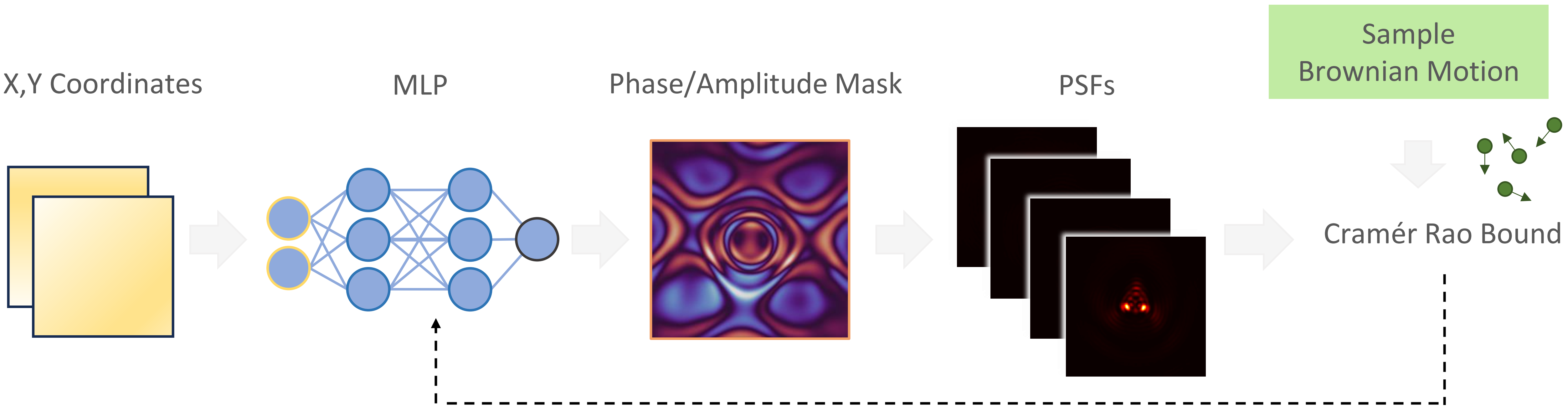
$f(\boldsymbol{x})$



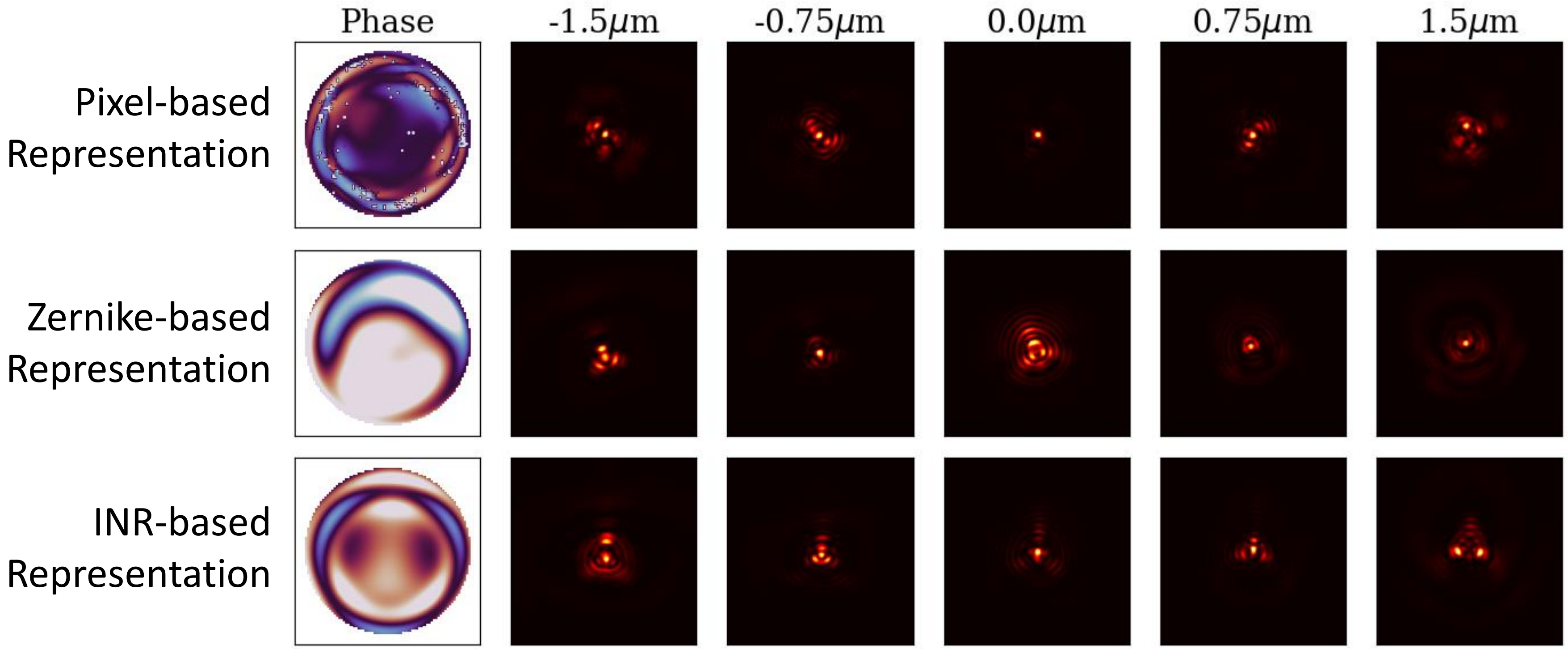

domain $\boldsymbol{x} = \begin{bmatrix} x \\ y \end{bmatrix}$

[Image Credit: Kris Kitani]

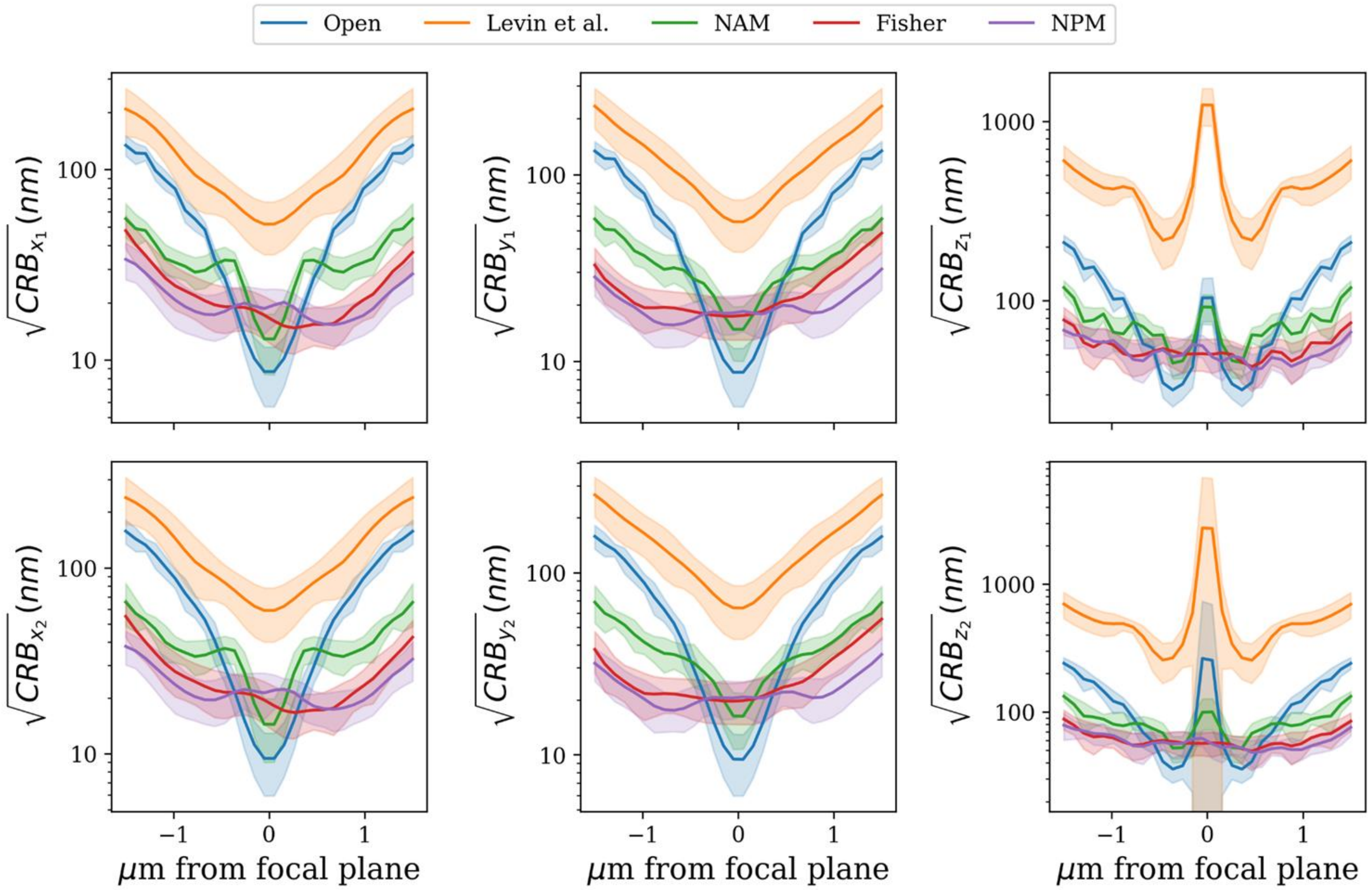
# Optical Design with Implicit Neural Representations and MC Sampling



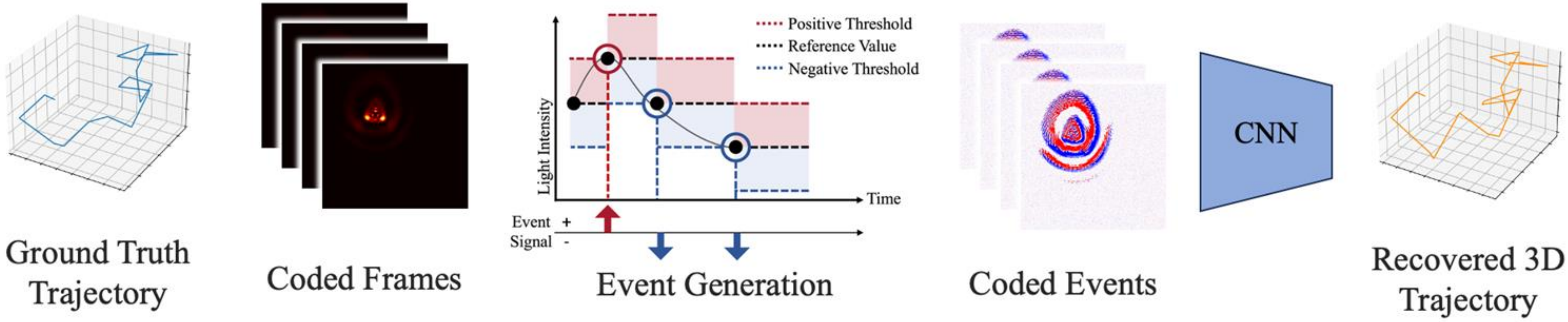
# Optical Design: Learned Masks



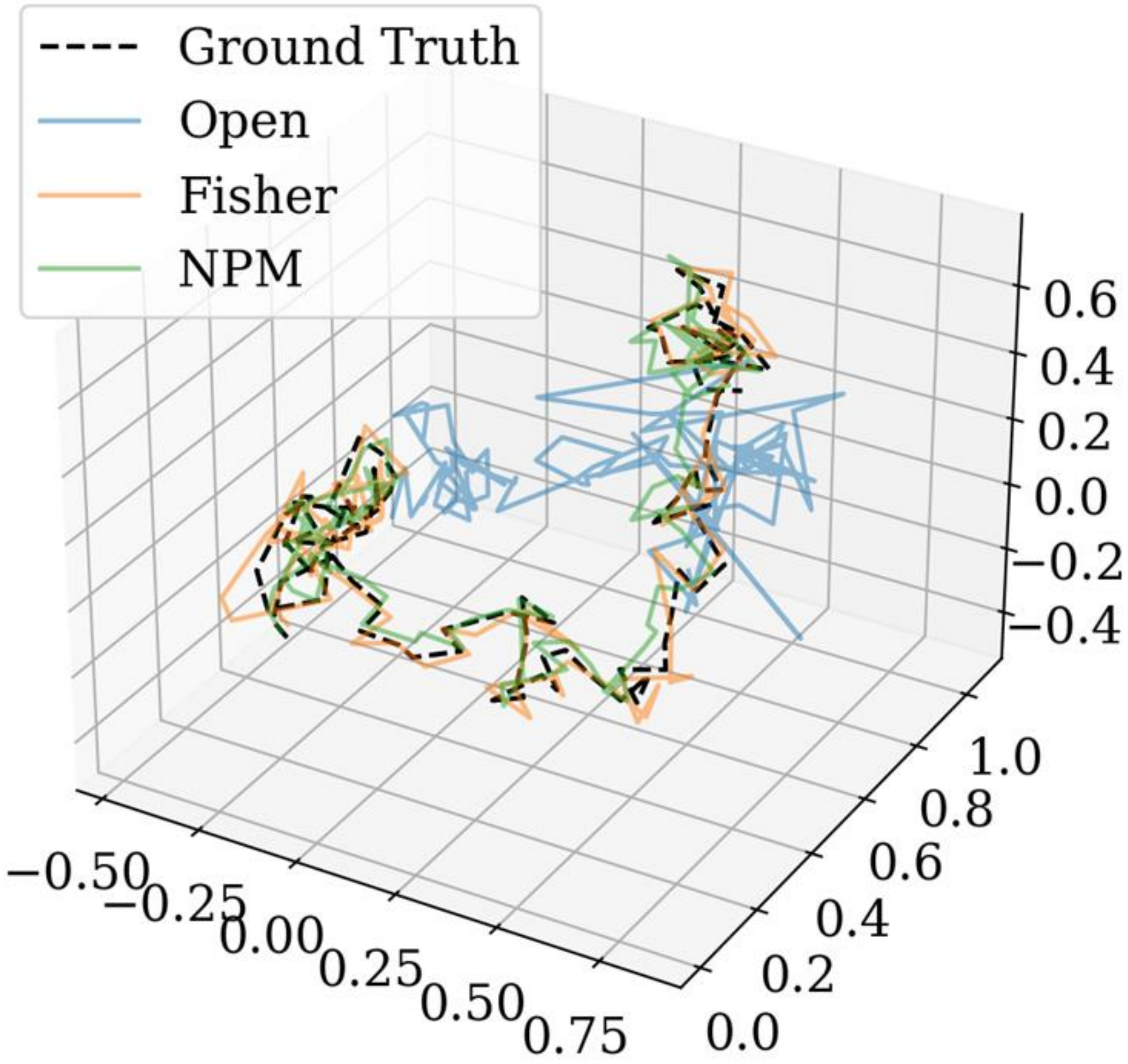
# Theoretical Results



# 3D Tracking: Simulation Training



# 3D Tracking: Results



# Lab Prototype: Setup

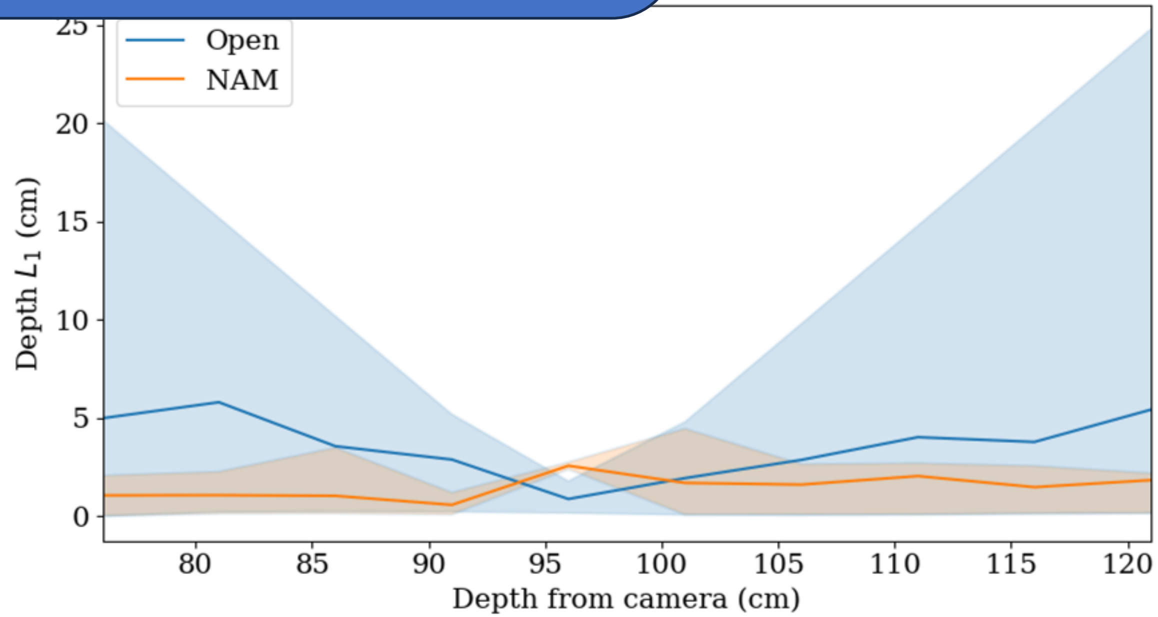
Prototype

Regular Event

Coded Event

Next Steps: More sophisticated optics

Optical Setup



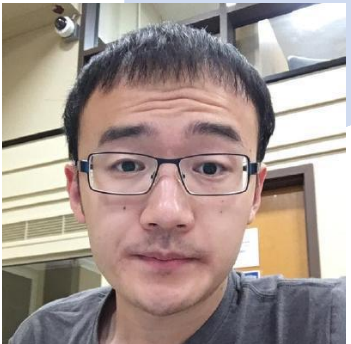
# Todays Talk:

Part 1
Optical Encoder

Sensor
Bottleneck

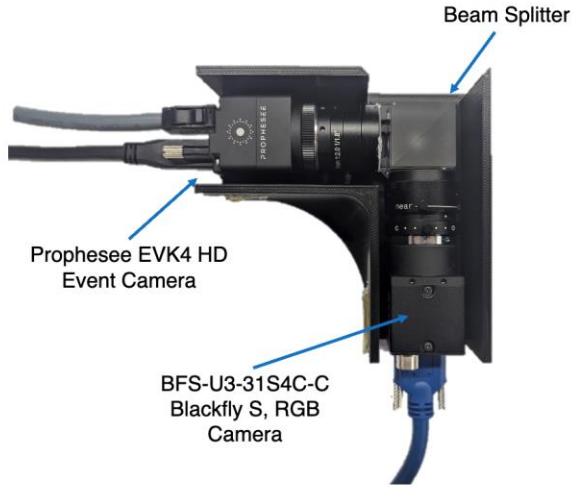
Part 2
Digital Decoder

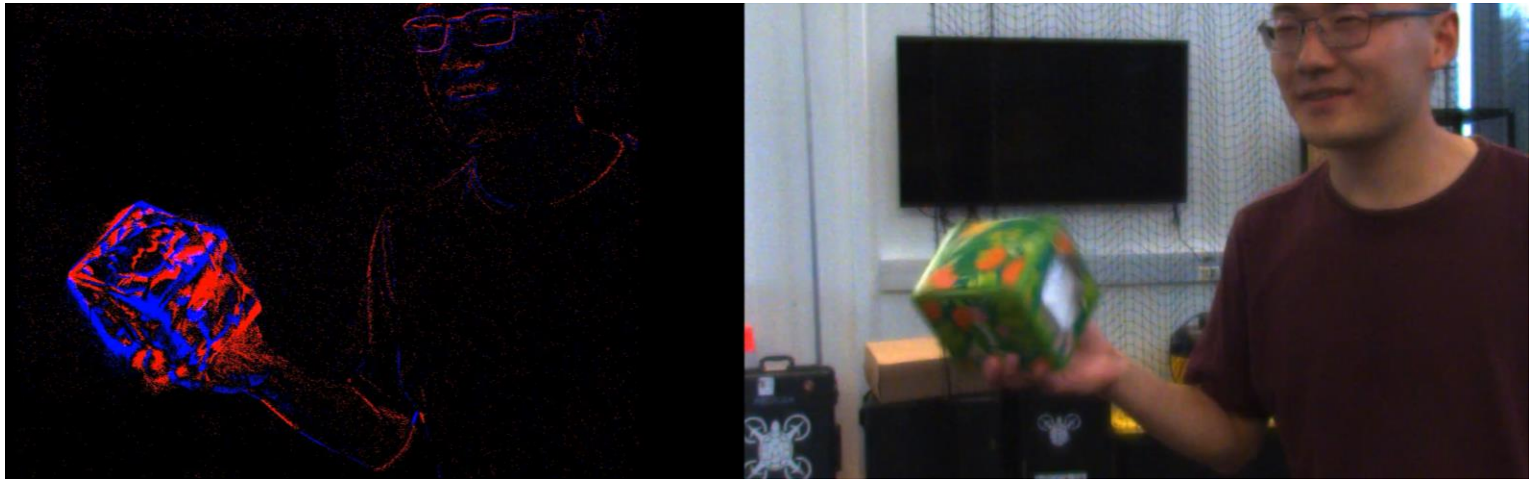
Jingxi Chen

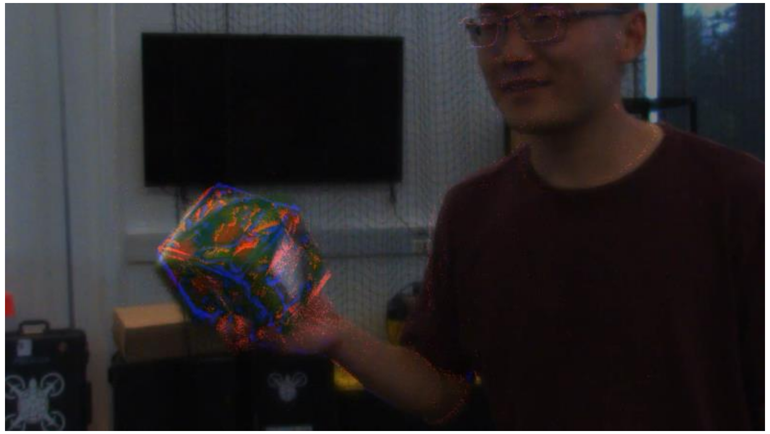


# Event-Guided Video Frame Interpolation

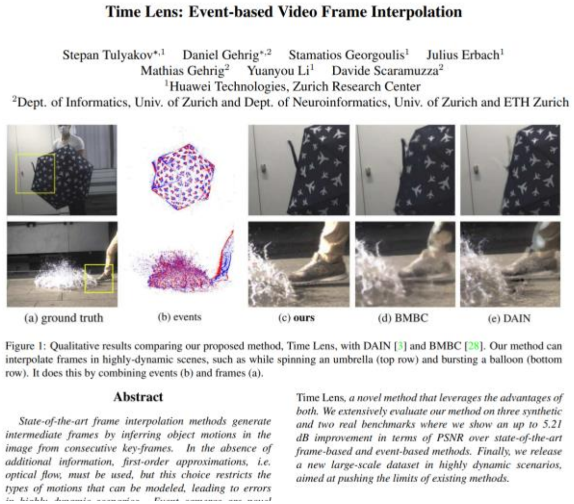
- Large motion between frames makes rgb-only video frame interpolation ill-posed
- Event-based Video Frame Interpolation (EVFI) addresses this challenge by using sparse, high-temporal-resolution event measurements as motion guidance.

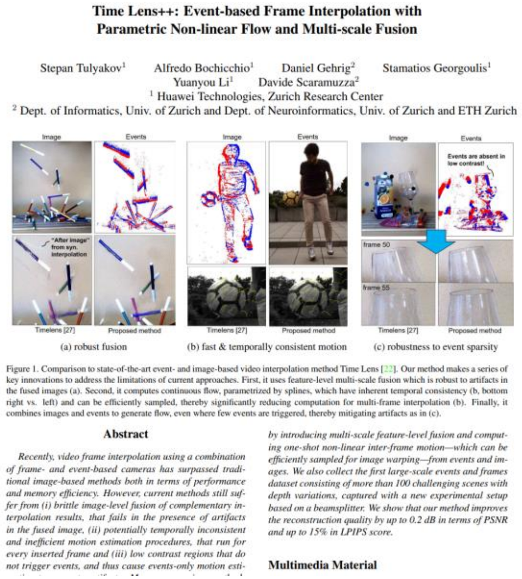


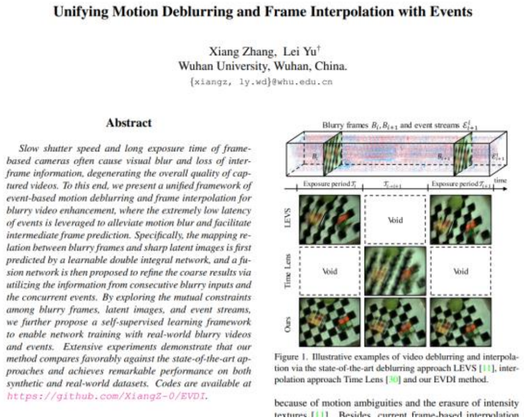



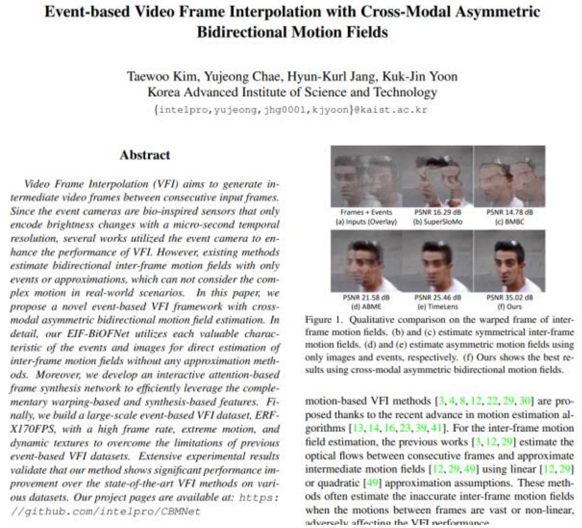


# Related Work




More paired training data, bigger and more expressive models, better performance

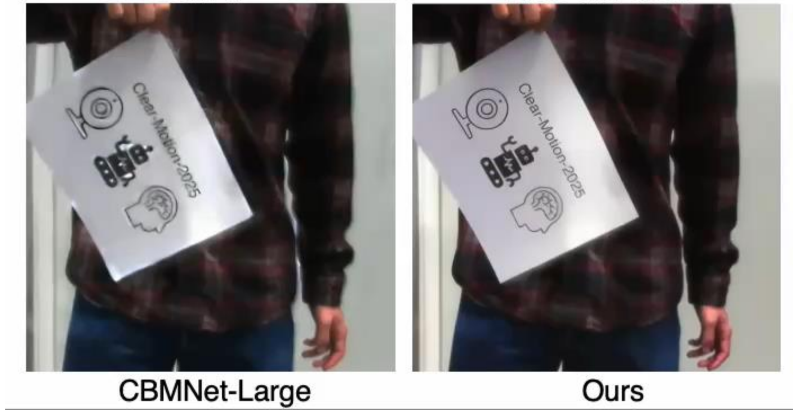
# Two Issues

More paired event + rgb data → more $$$$

Larger models → more overfitting to specific cameras and interpolation rates

In Domain:


Out of Domain:



Can we benefit from massive datasets and highly expressive models *without* having to generate or train these datasets/models ourselves?

# Our Contribution: Bring in the Big Guns!

*Pretrained* Video Diffusion Models

1. Internet-scale Data → Strong data prior (Generalization)

2. Video Diffusion → Denoising a video at once, temporal consistency

Image Credit: https://stability.ai/news/stable-video-diffusion-open-ai-video-model

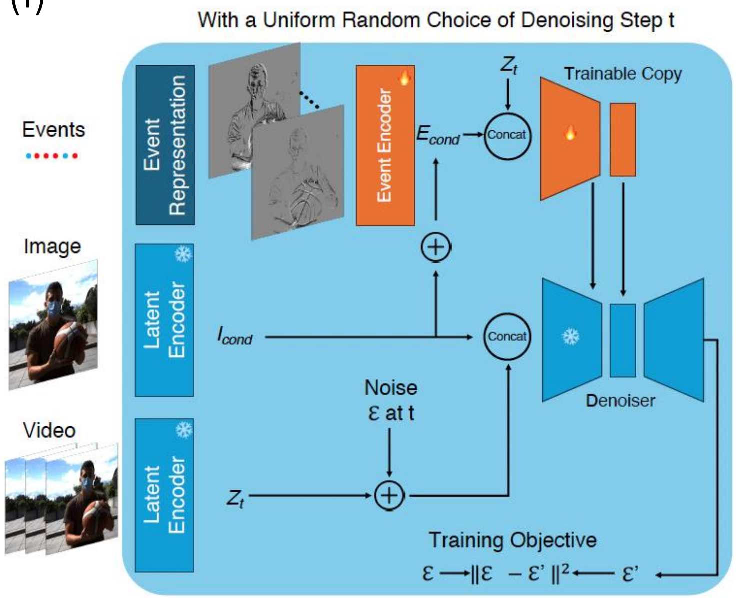
# Proposed Approach:

(i) Control a pretrained diffusion model with event guidance

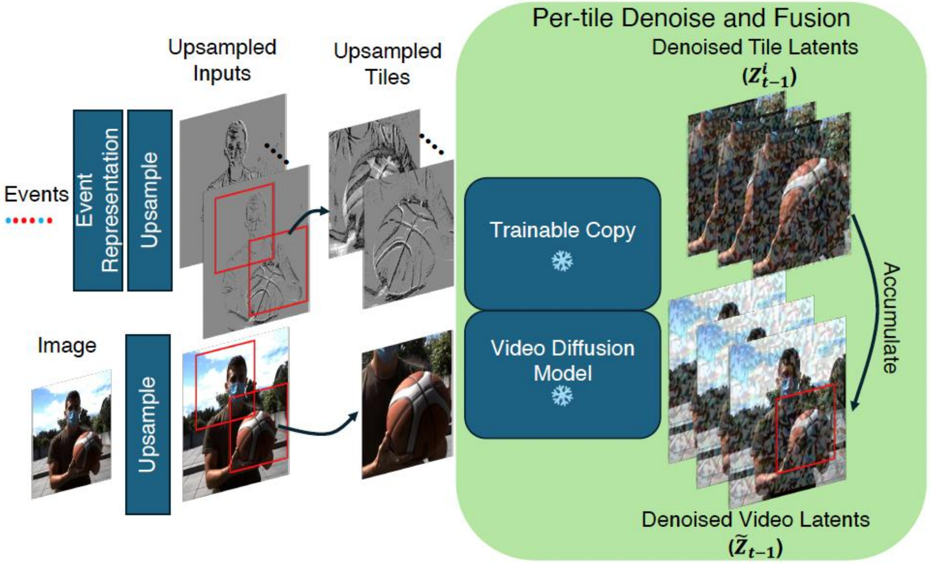
(ii) Preprocess to preserve spatial and temporal resolution

(iii) Use video *generation* network to perform video *interpolation*

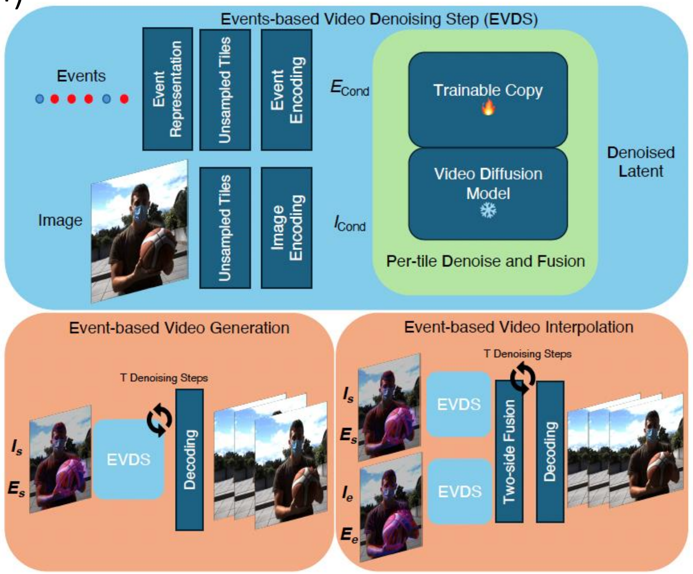
(i)




(ii)




(iii)



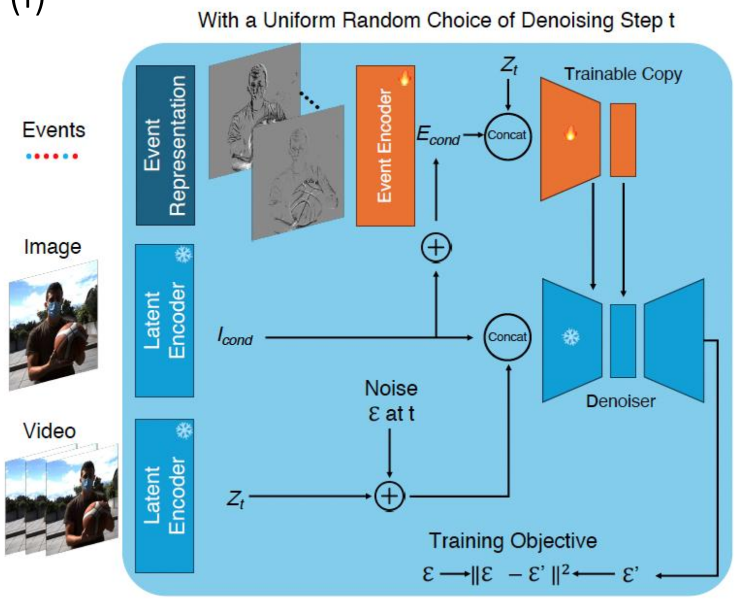
# Proposed Approach:

(i) Control a pretrained diffusion model with event guidance

(ii) Preprocess to preserve spatial and temporal resolution

(iii) Use video *generation* network to perform video *interpolation*

(i)




ControlNet-style approach adapts a small event-to-latent encoder while preserving the original video diffusion models weights.

Can learn to control diffusion model with only a limited amount of training data, without the risk of forgetting the original video priors

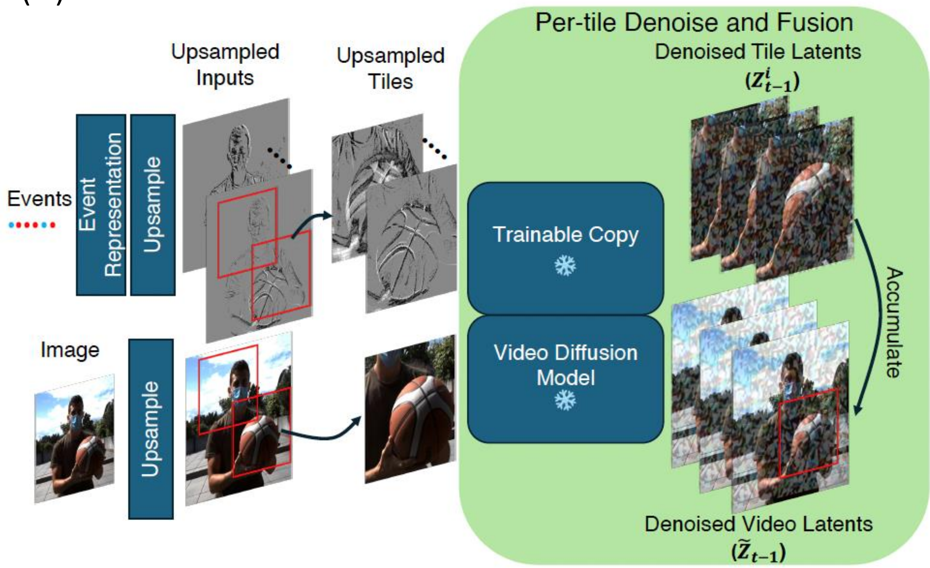
# Proposed Approach:

(i) Control a pretrained diffusion model with event guidance

(ii) Preprocess to preserve spatial and temporal resolution

(iii) Use video *generation* network to perform video *interpolation*

(ii)




The diffusion models encoding process is inherently lossy

Can preserve fine details by upsampling inputs before encoding

Without upsampling

With upsampling

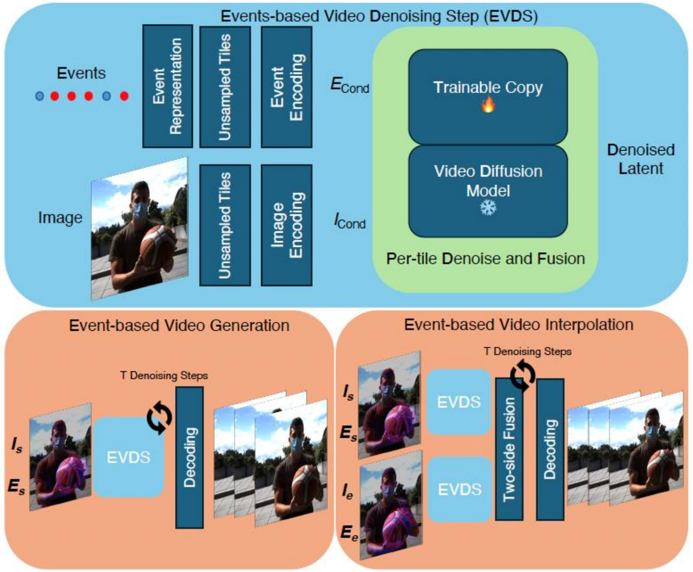
# Proposed Approach:

(i) Control a pretrained diffusion model with event guidance

(ii) Preprocess to preserve spatial and temporal resolution

(iii) Use video *generation* network to perform video *interpolation*

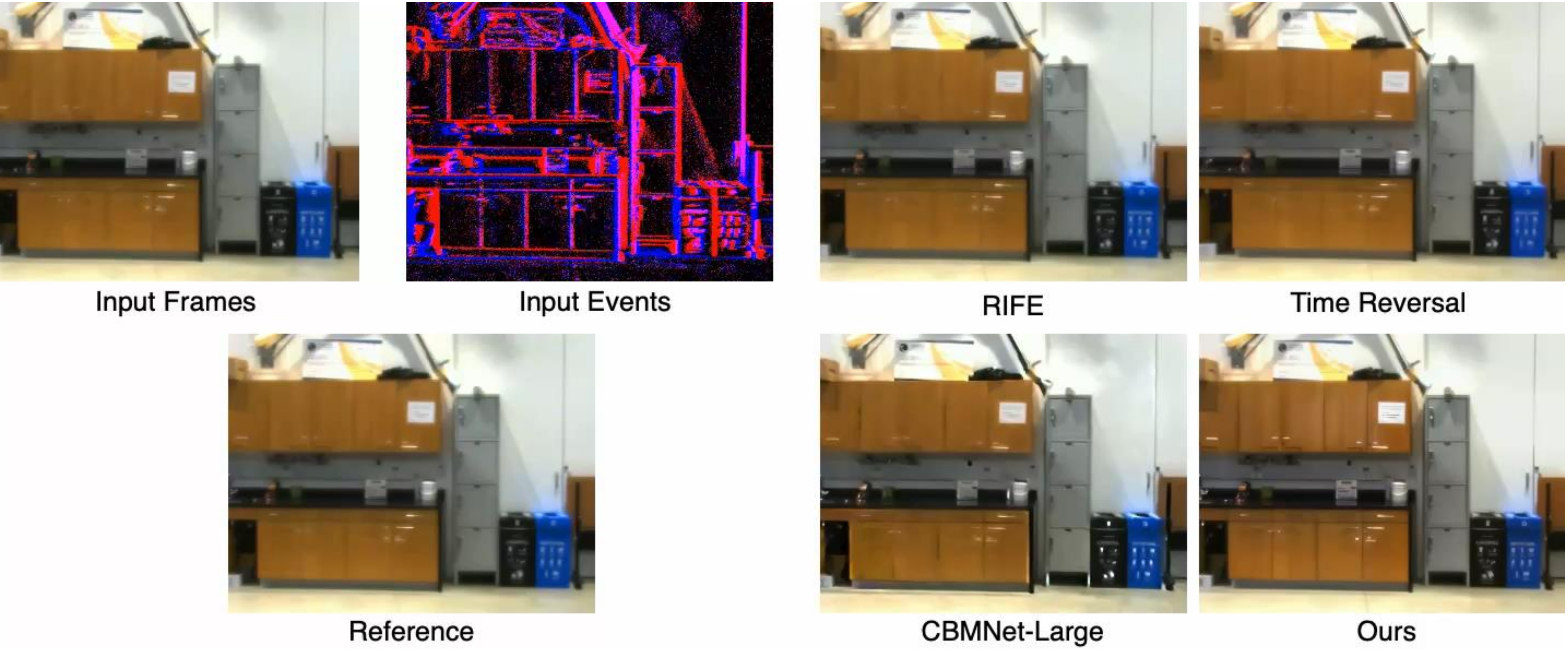
(iii)

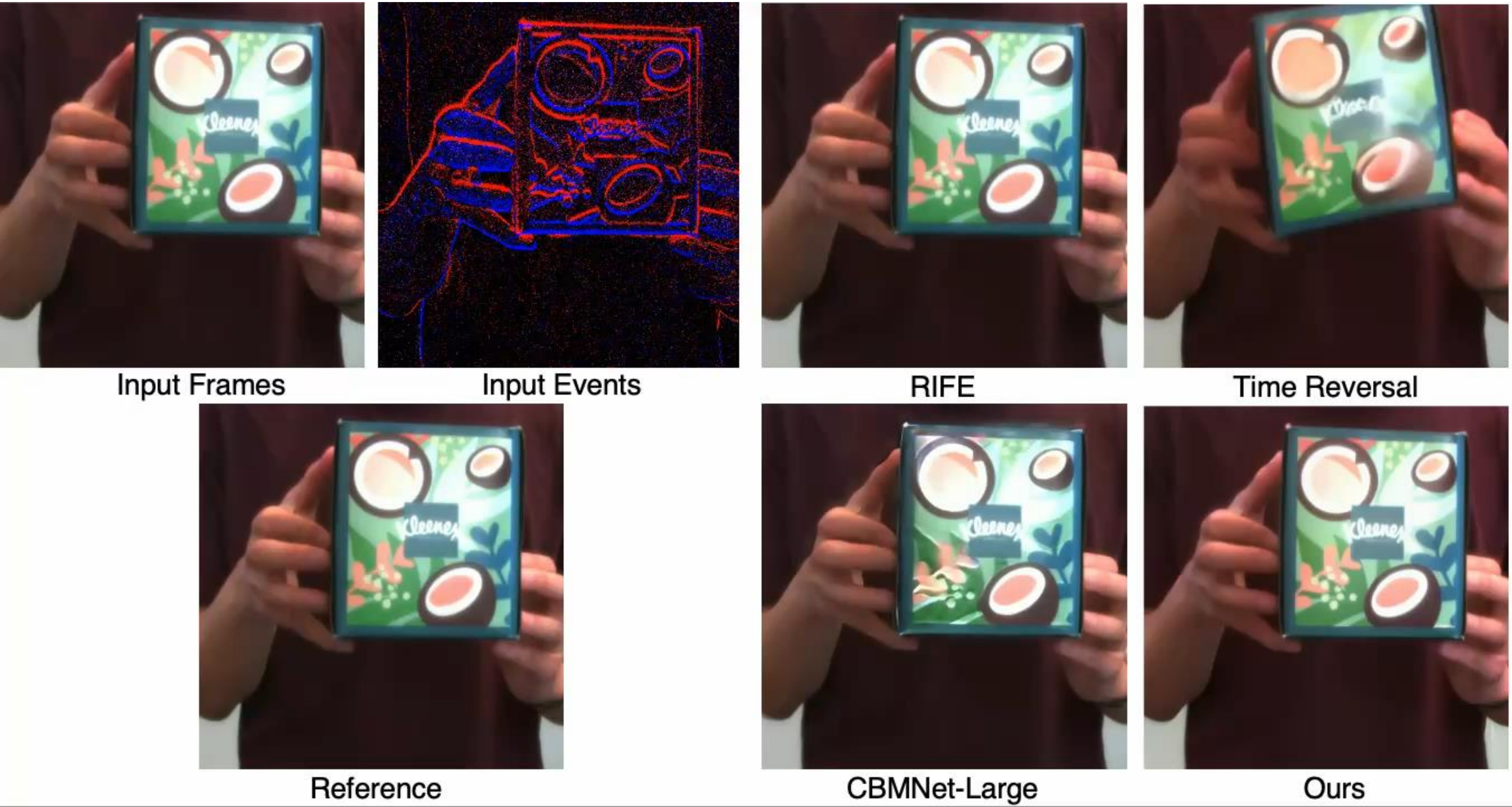



Video diffusion model performs generation, not interpolation

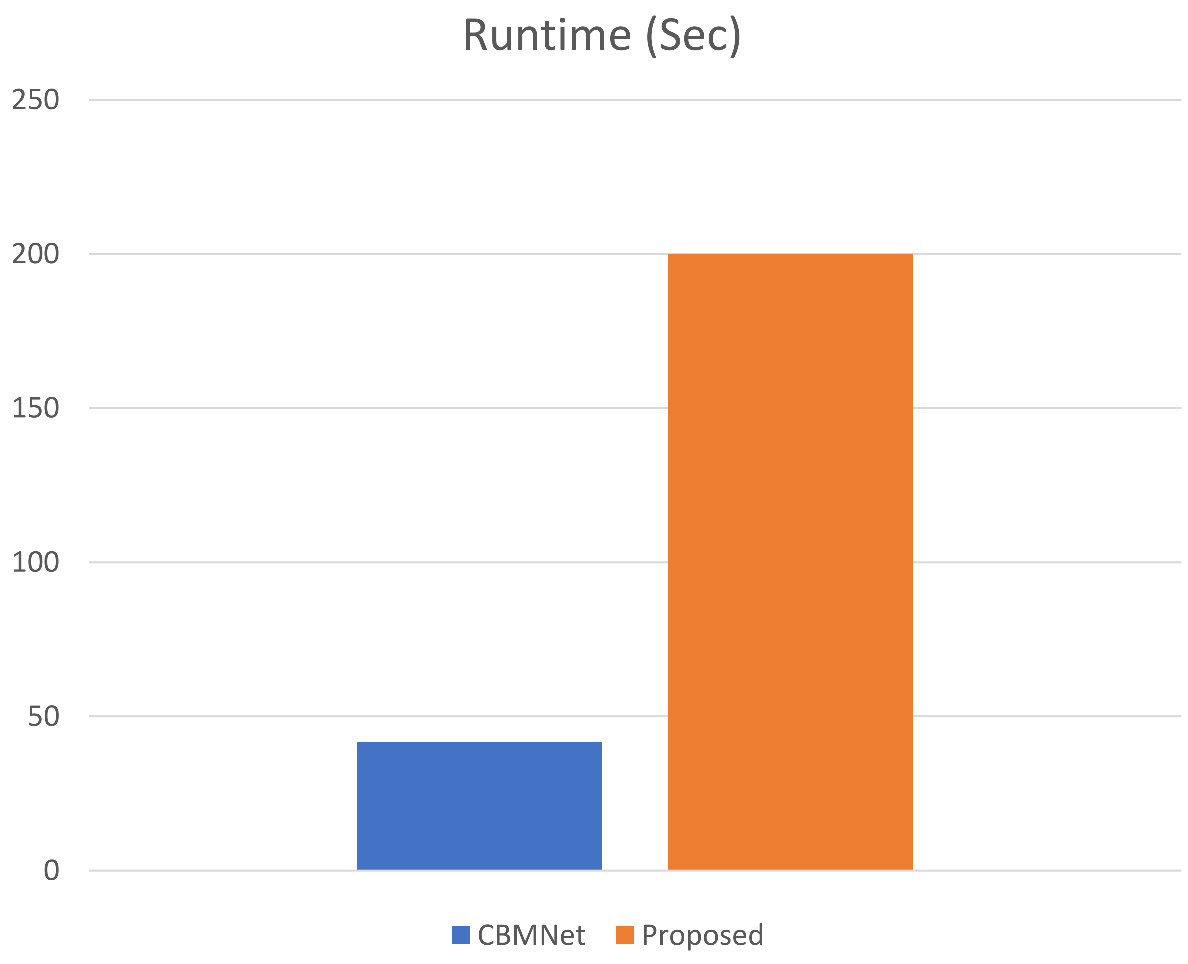
To interpolate, run video model forward (from first frame) and in reverse (from last frame). Blend latents.

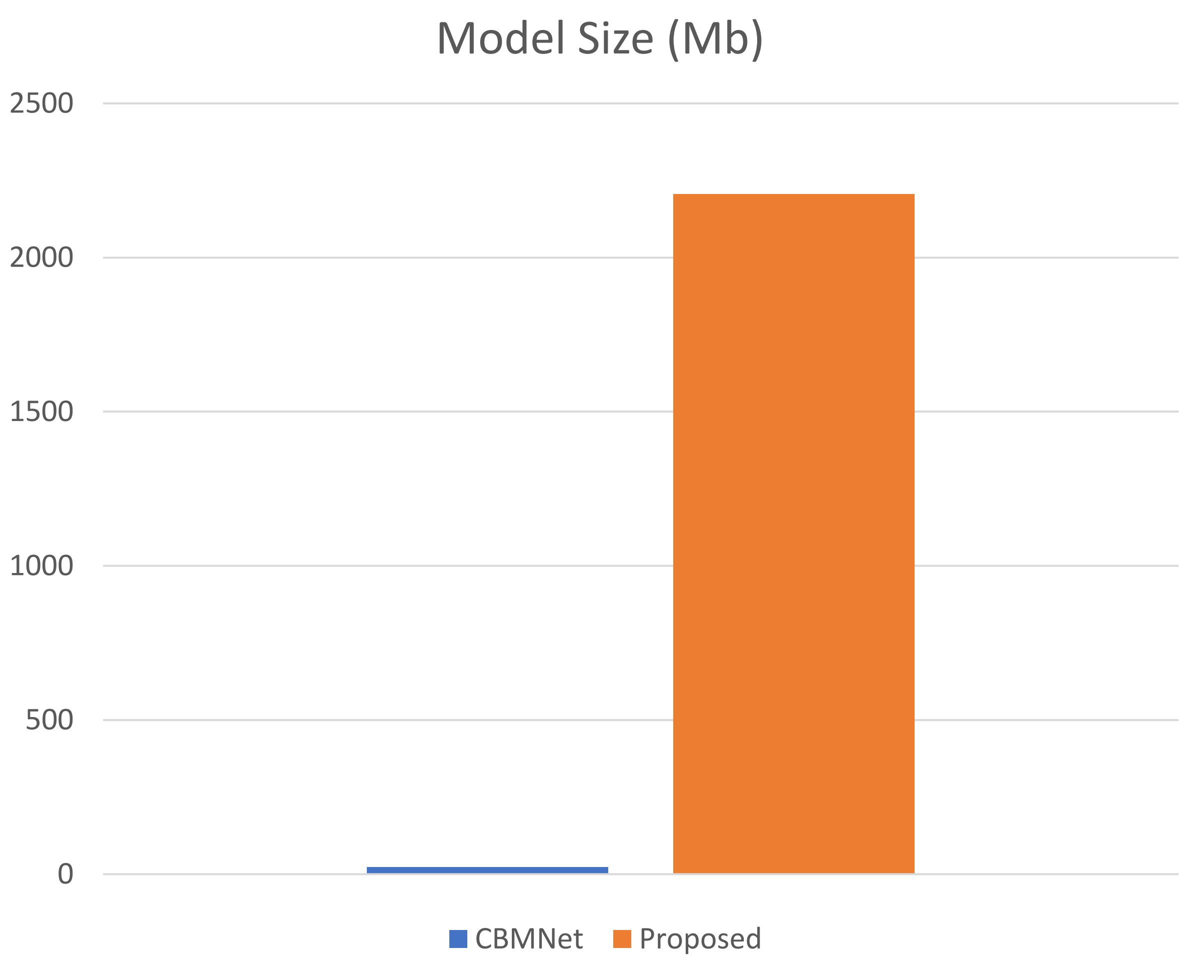
# Results: Generalizes to Extreme Interpolation without Fine Tuning



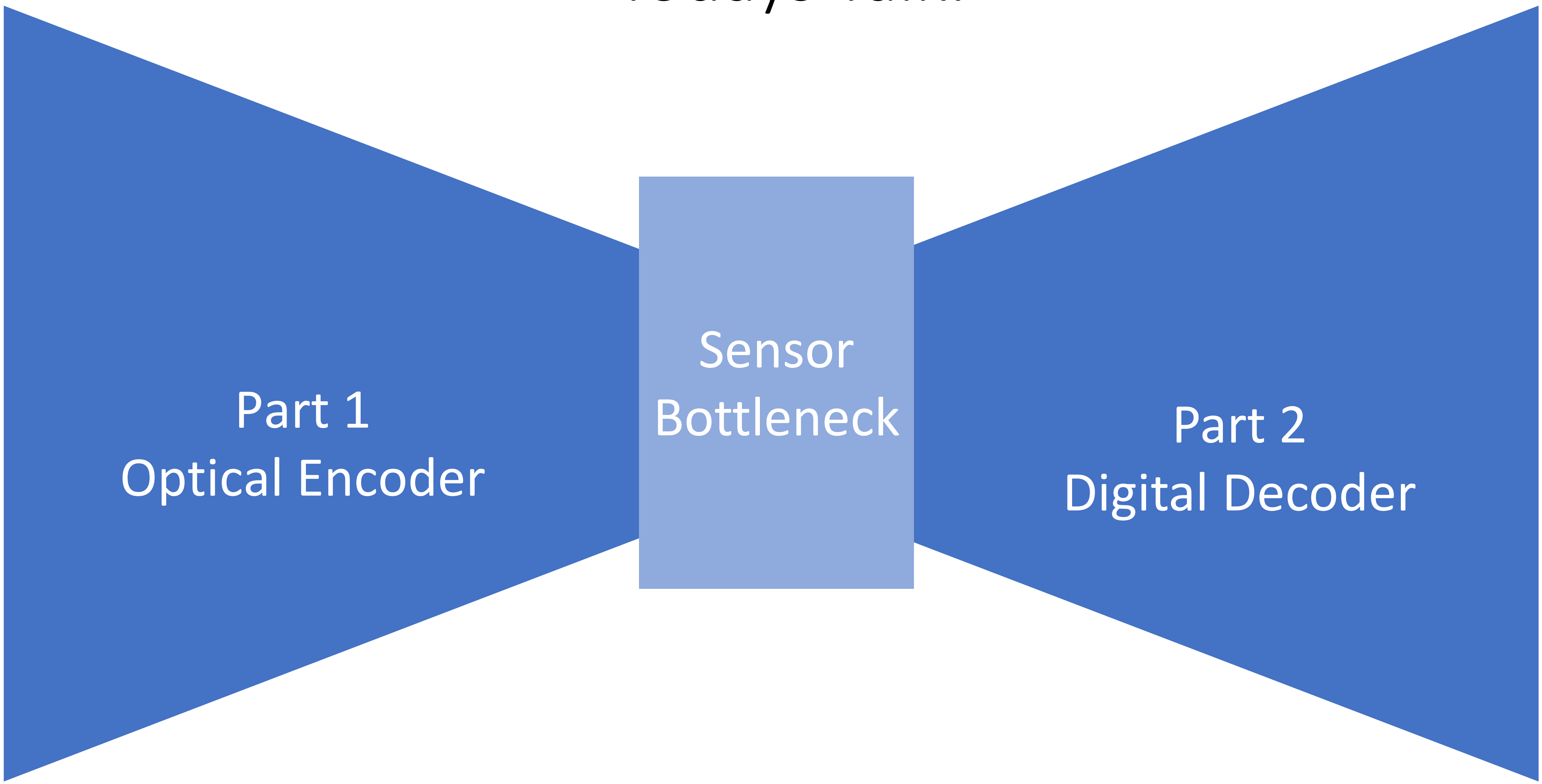

Input Frames
Input Events
RIFE
Time Reversal
Reference
CBMNet-Large
Ours

# Downsides?

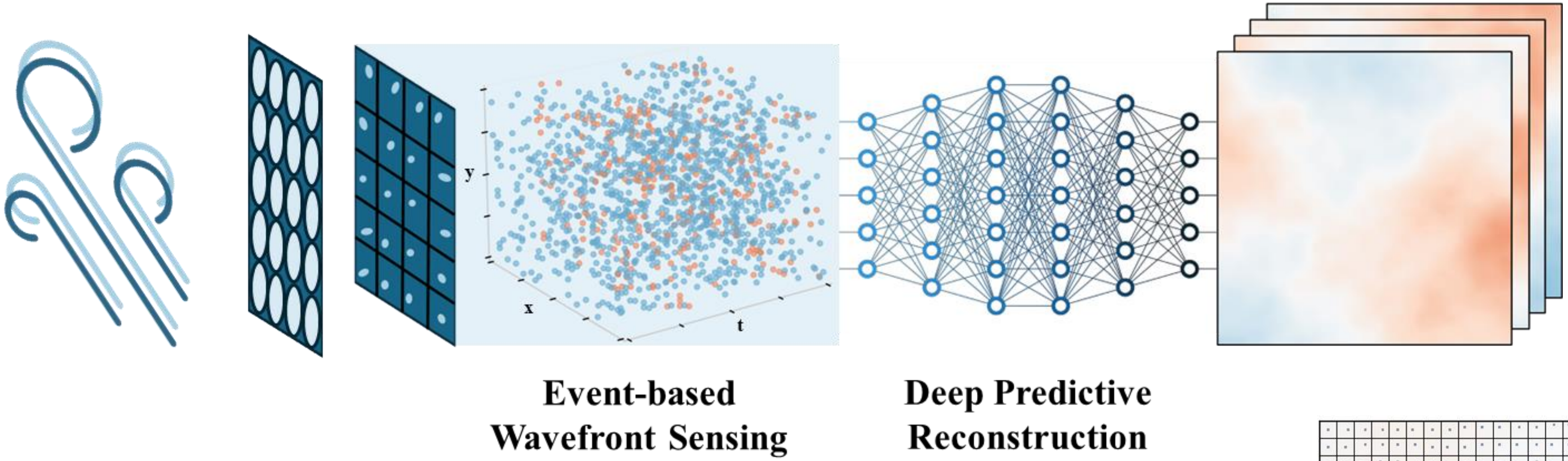

Runtime (Sec)
250
200
150
100
50
0
CBMNet
Proposed

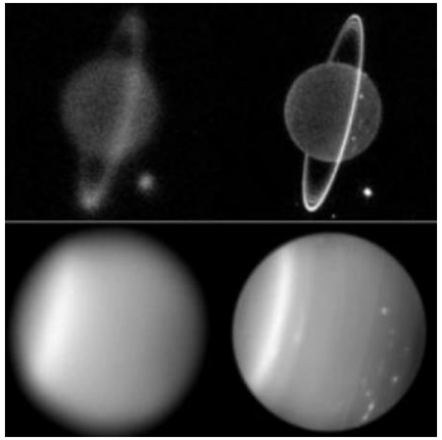

Model Size (Mb)
2500
2000
1500
1000
500
0
CBMNet
Proposed

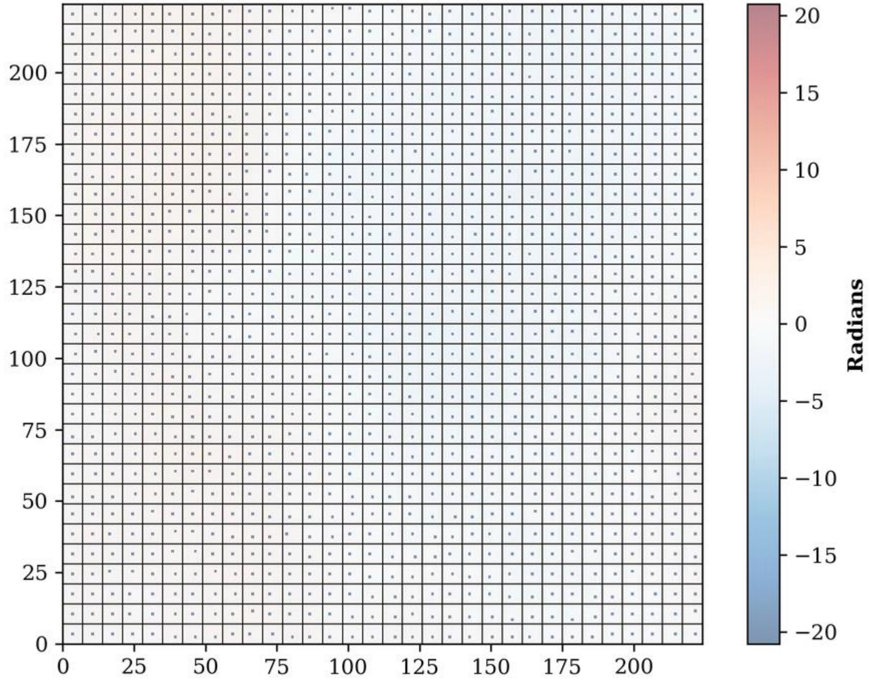
# Todays Talk:


Part 1
Optical Encoder
Sensor
Bottleneck
Part 2
Digital Decoder

# High-speed predictive wavefront sensing with event-cameras



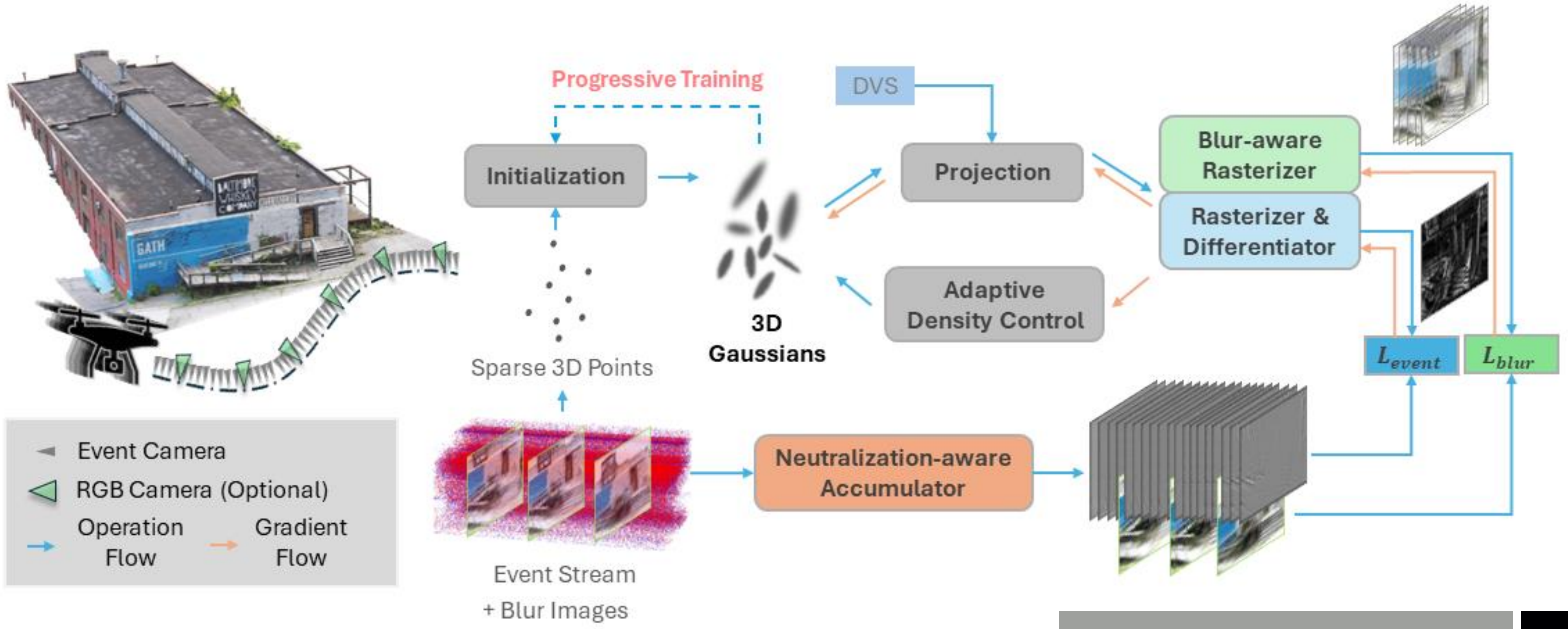



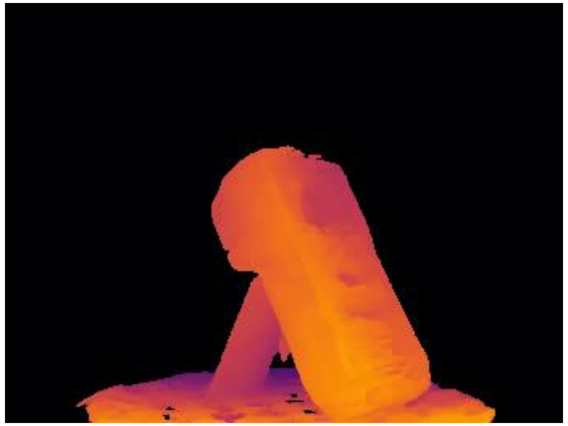



[Ziemann et al. 2024

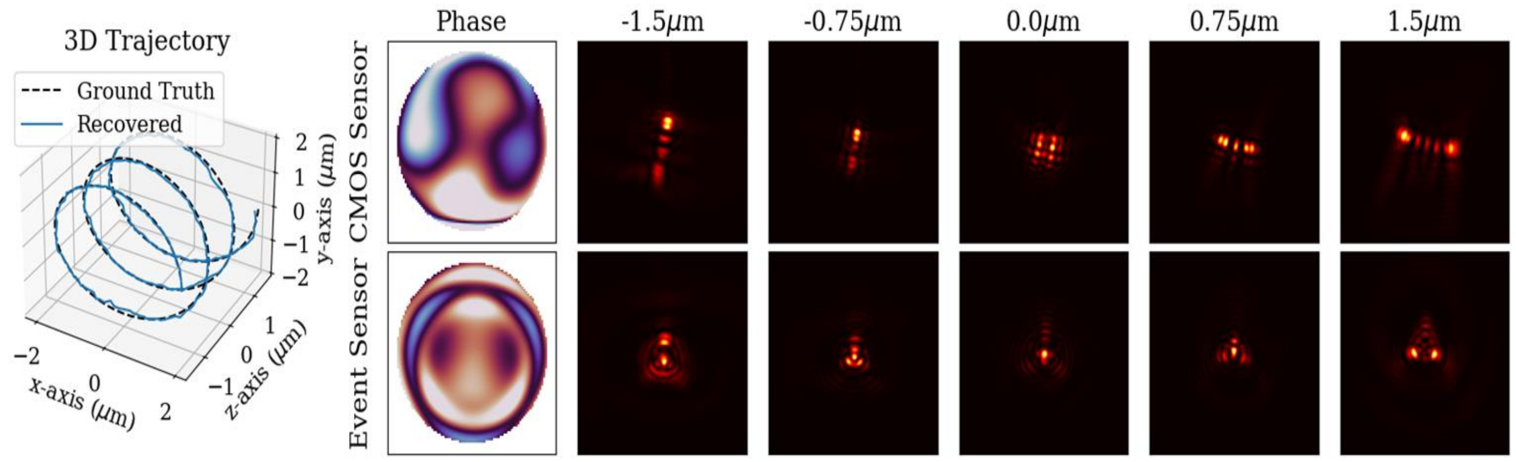
[Image Credit: *Sky and Telescope Magazine*]

# Dense 3D Reconstruction with Inverse Differentiable Rendering



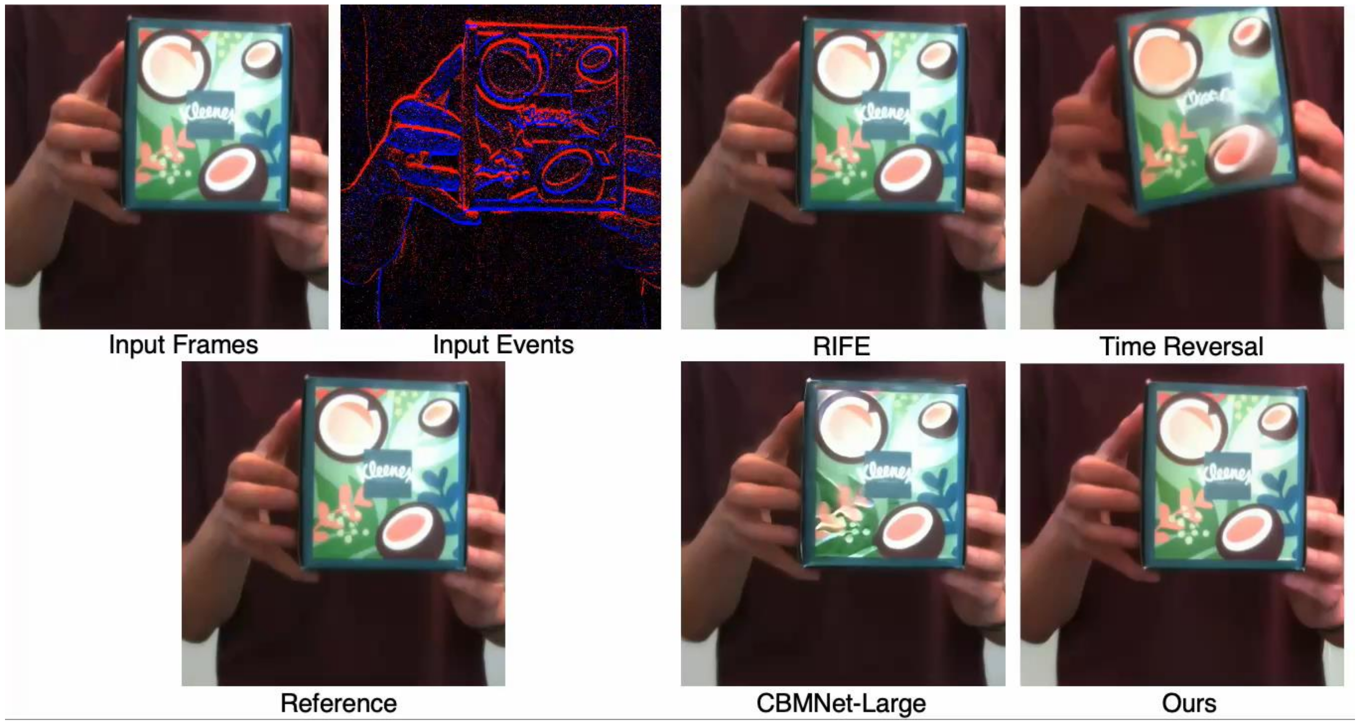



[Mahbub et al. 2023, Xiong et al. 2024]




Event cameras + coded optics

Event cameras + generative models




References:

CodedEvents: Optimal Point-Spread-Function Engineering for 3D-Tracking with Event Cameras *CVPR 2024*

Repurposing pre-trained video diffusion models for event-based video interpolation *CVPR 2025.* **Sat. morning poster session**

# Acknowledgements


Sachin Shah


Matthew Chan


Haoming Cai


Jingxi Chen


Sakshum Kulshrestha


Chahat Deep Singh


Brandon Feng

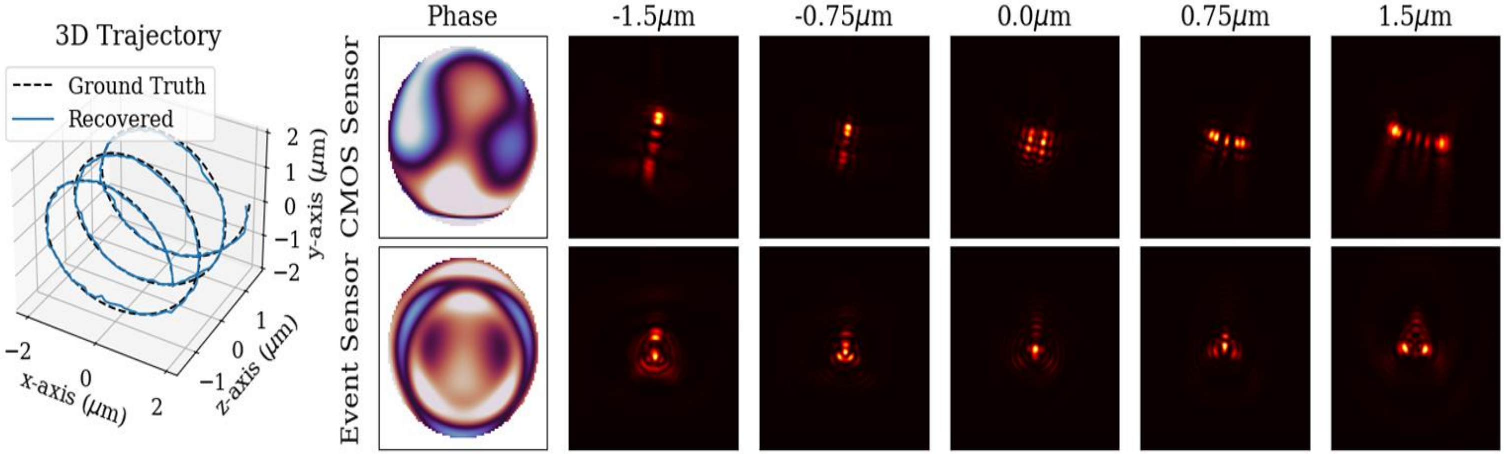

Tianfu Wang


Levi Burner


Dehao Yuan

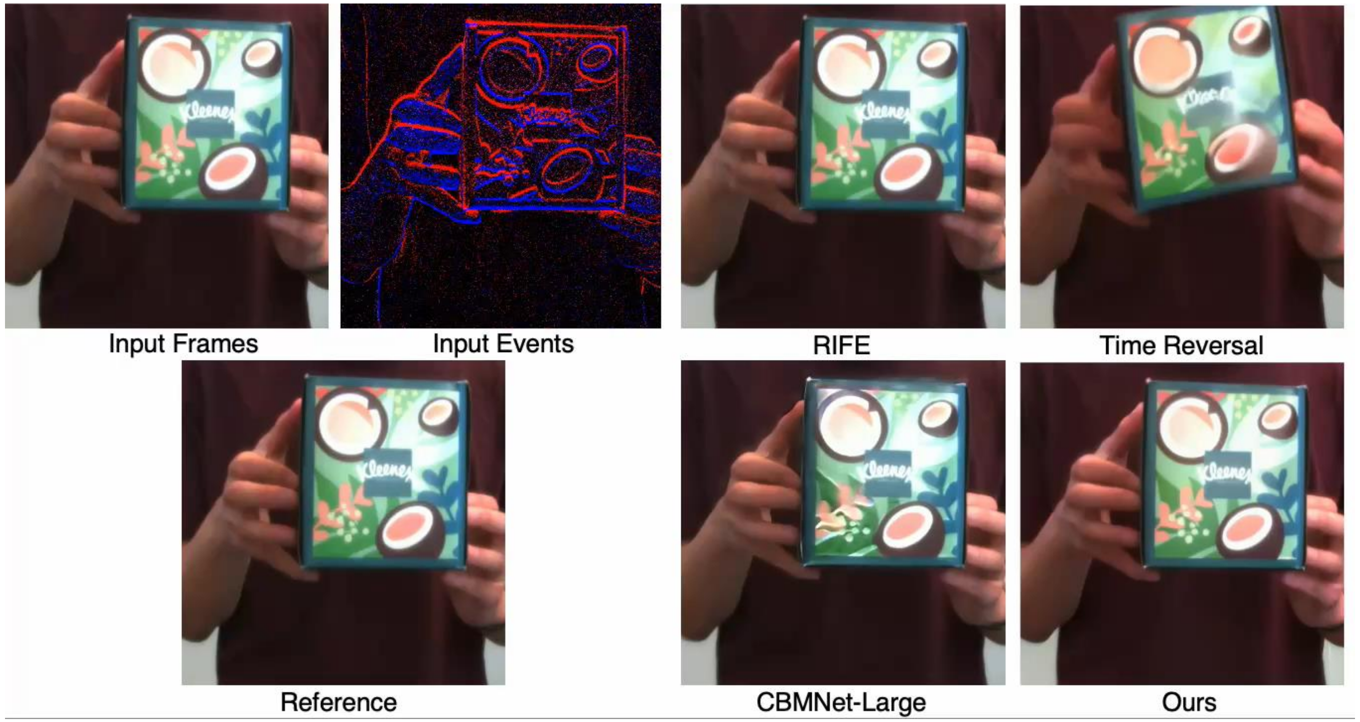

Cornelia Fermuller


Yiannis Aloimonos




Event cameras + coded optics

Event cameras + generative models




References:

CodedEvents: Optimal Point-Spread-Function Engineering for 3D-Tracking with Event Cameras *CVPR 2024*

Repurposing pre-trained video diffusion models for event-based video interpolation *CVPR 2025.* **Sat. morning poster session**