Visual Sensing Using Machine Learning

Vivek Boominathan

Email: vivek.booms@gmail.com

Webpage: vivekboominathan.com

Ashok Veeraraghavan

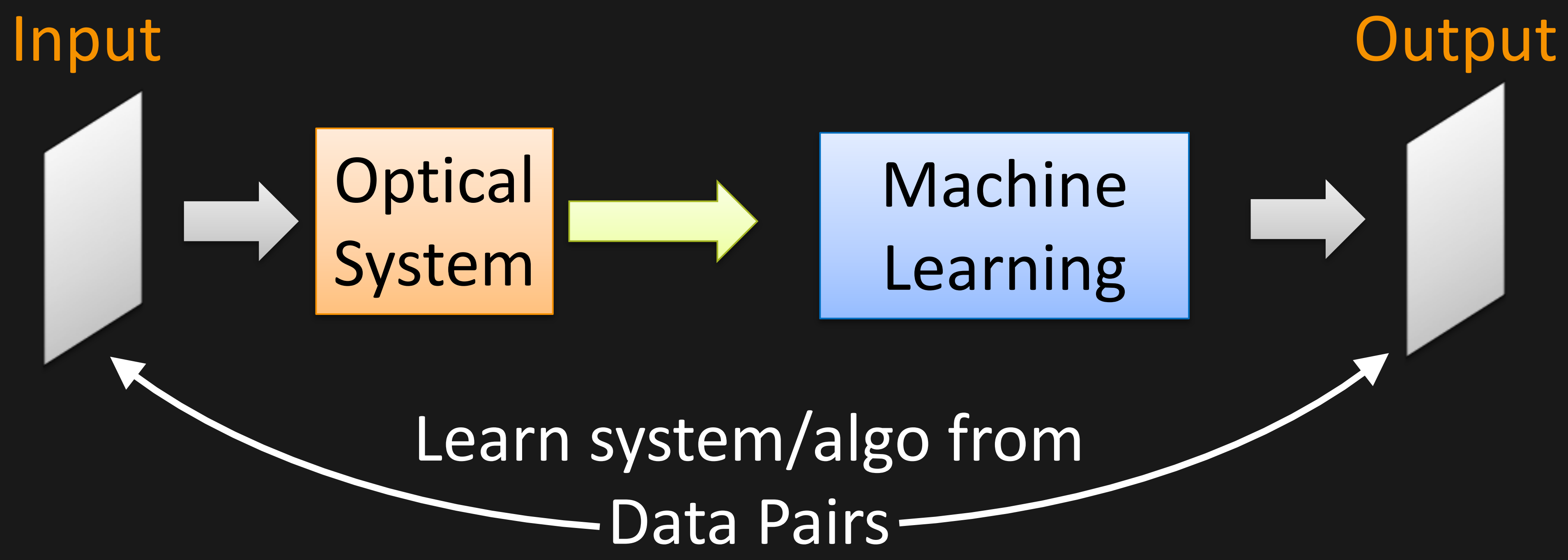
http://computationalimaging.rice.edu/

Rice University

Computational Visual Sensing

Tagging computational algorithms with optical systems has expanded visual sensing ability

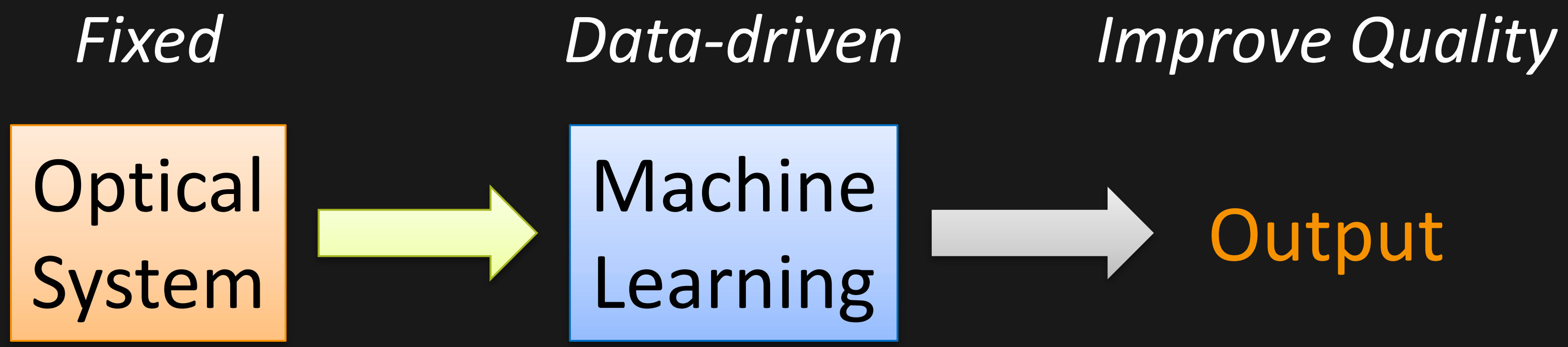
Data-driven Visual Sensing



- Visual data lie in a complex high-dimensional space that is hard to model analytically.
- Data-driven approach allows to “learn” a good approximate model from visual data.
- The “learned” model can improve the performance of visual sensing systems.

Visual Sensing using Machine Learning

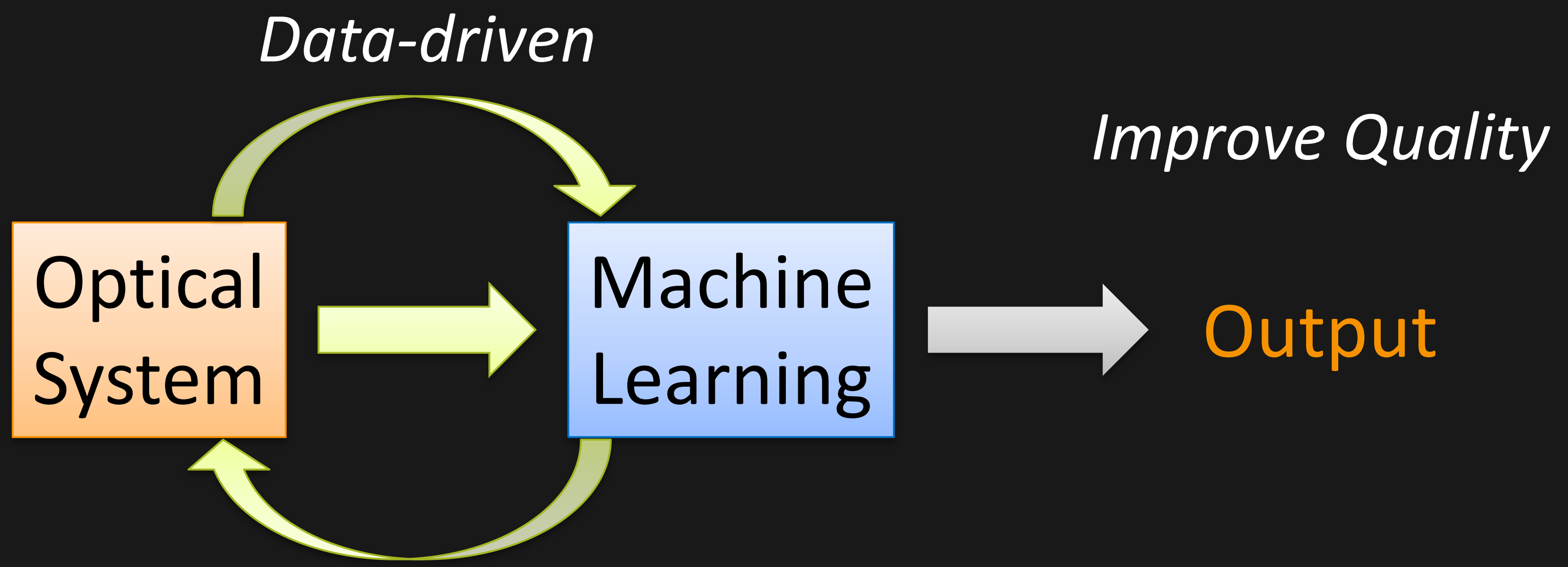
Part I: Backend ML



Data-driven "learned" models can improve the quality of visual sensing systems.

Visual Sensing using Machine Learning

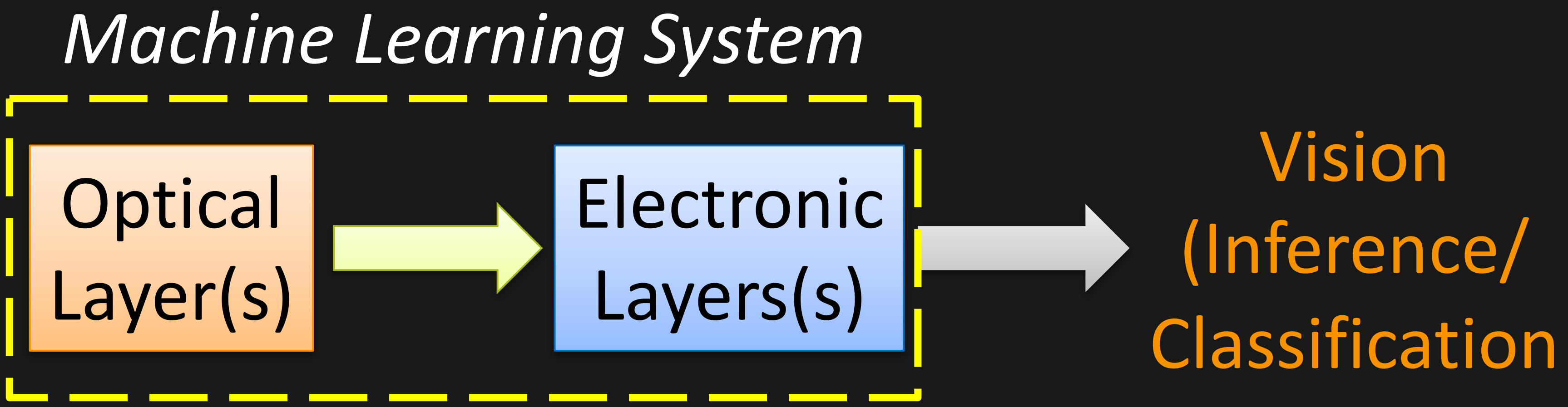
Part II: Joint design with ML



Joint design with data-driven techniques can bring out best in both systems

Visual Sensing using Machine Learning

Part III: ML with Optical System



Incorporating optical layer(s) into machine learning system can decrease latency and power

Visual Sensing using Machine Learning

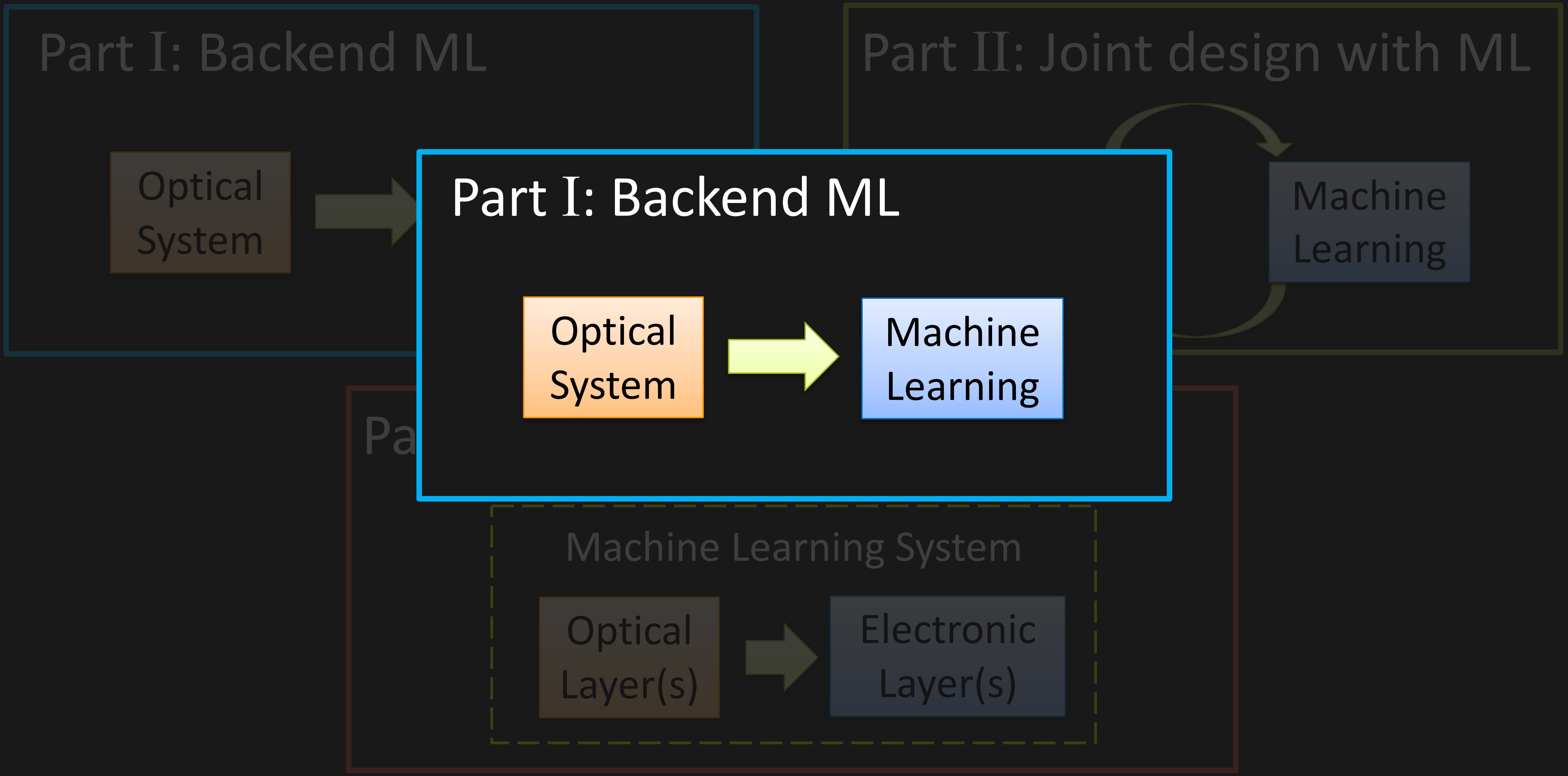


Image Priors

Hand-crafted priors

- Images as signals
- Signal statistics priors
- E.g. Sparse gradients, self-similarity

Data-driven priors

- Captures complex "visual" priors
- Not easily describable as signal statistics
- E.g. Perceptual feature maps, adversarial

Capturing good "Visual" priors lead to high-quality reconstructions

Example 1:

Solving Inverse Compressive Imaging Problems using Deep Pixel-level Prior

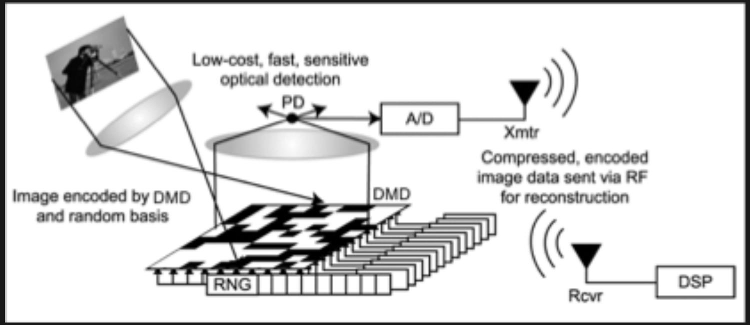
Akshat Dave, Anil Kumar Vadathya, Ramana Subramanyam, Rahul Baburajan, and Kaushik Mitra.
"Solving inverse computational imaging problems using deep pixel-level prior."
IEEE Transactions on Computational Imaging (TCI) (2018).

Compressive Optical System

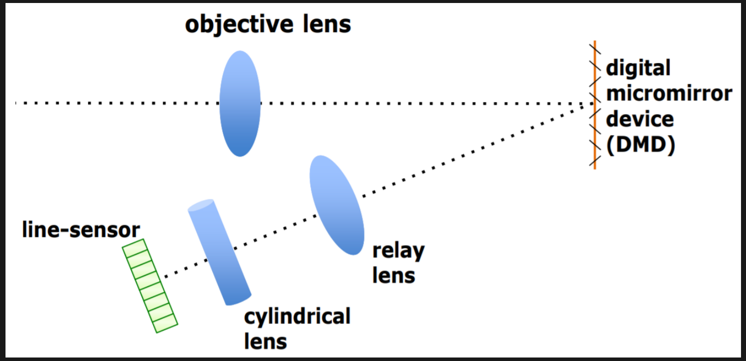
SPC

LiSens

Single Pixel Camera (SPC)



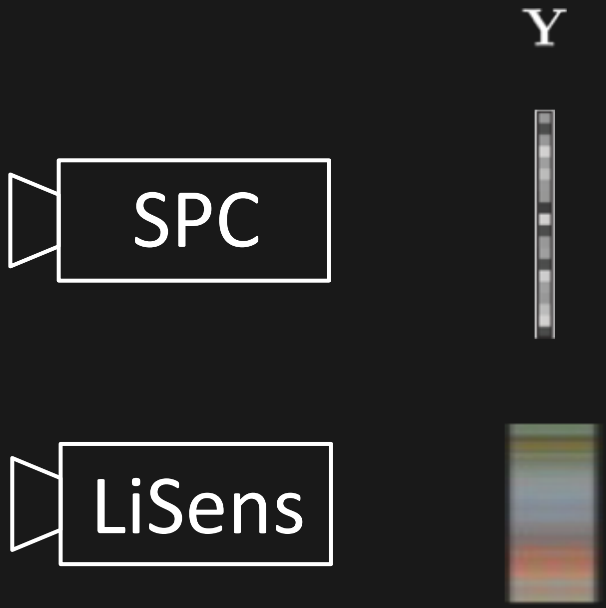
Line Sensor Camera (LiSens)



Spatially multiplexing to reduce sensor cost e.g. for non-visible wavelength

Baraniuk et al., 2006
Wang et al., 2015

Ill posed inverse problem



Measurements
<< Image size

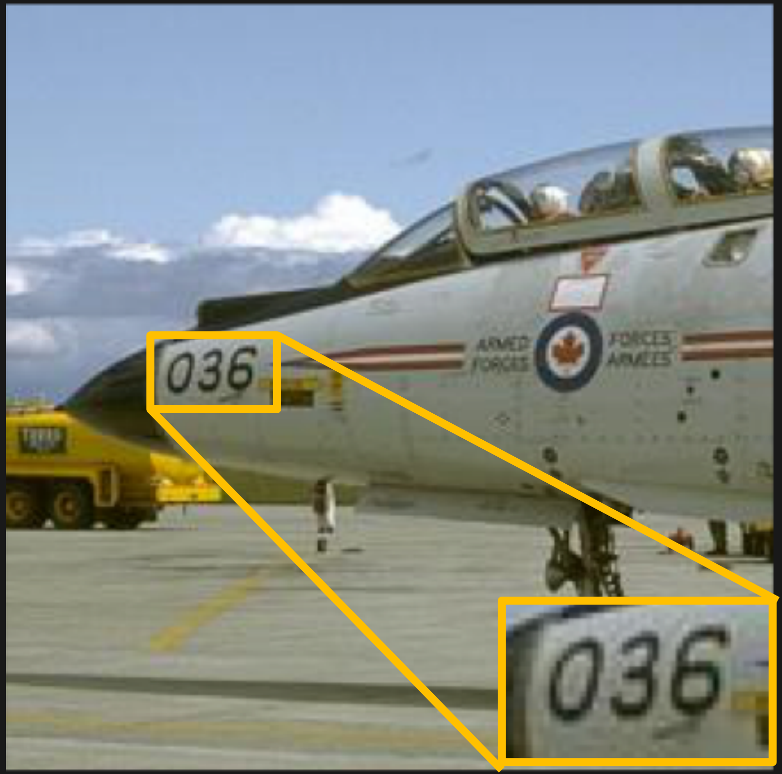
$$\boxed{\mathbf{Y} = A\mathbf{X}}$$

Forward model

Ill posed – A is is non invertible or poor condition number

Hand-crafted prior

Original Image



Reconstructed Image
(5% Measurements)

TVAL3

Learn the prior

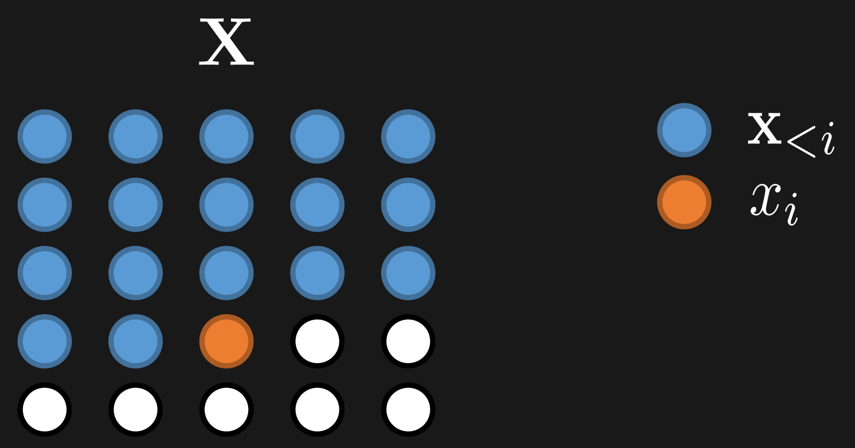
- Using a set of dataset of natural images

- Deep generative models: Quite successful in modelling natural image distribution

Dataset of natural images

$\longrightarrow$ Deep Generative Model $\longrightarrow$ $p_\theta(\mathbf{X})$

Autoregressive Models

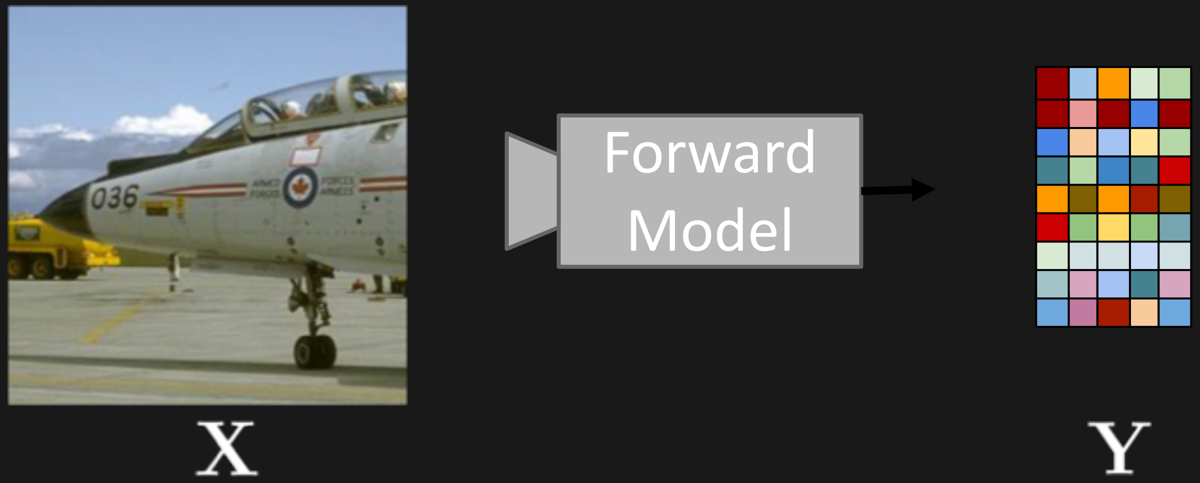


- Factorize the prior distribution as

$$p(\mathbf{X}) = p(x_1, x_2, \ldots, x_{n^2}) = \prod_{i=1}^{n^2} p(x_i | \mathbf{x}_{<i})$$

- Tractable expression for prior density
- Provides pixel-level consistencies in reconstructions
- PixelCNN++

Problem Formulation



- Given Y and the forward model, find X

- Maximum-A-Posteriori Inference

$$\hat{\mathbf{X}} = \underset{\mathbf{X}}{\operatorname{argmax}} \log p(\mathbf{X}|\mathbf{Y})$$

$$\hat{\mathbf{X}} = \underset{\mathbf{X}}{\operatorname{argmax}} \quad \underbrace{\log p(\mathbf{X})}_{\text{Learned Autoregressive Model}} \quad + \quad \underbrace{\log p(\mathbf{Y}|\mathbf{X})}_{\text{Forward Model}}$$

Iterative Approach

$$\hat{\mathbf{X}} = \underset{\mathbf{X}}{\operatorname{argmax}} \boxed{\log p(\mathbf{X})} + \boxed{\log p(\mathbf{Y}|\mathbf{X})}$$

Repeat until convergence

Prior Step

Gradient ascent over Autoregressive Prior

Likelihood Step

Satisfy forward model

Iterative Approach

$$\hat{\mathbf{X}} = \underset{\mathbf{X}}{\operatorname{argmax}} \boxed{\log p(\mathbf{X})} + \boxed{\log p(\mathbf{Y}|\mathbf{X})}$$

Repeat until convergence

Prior Step

Gradient ascent over Autoregressive Prior

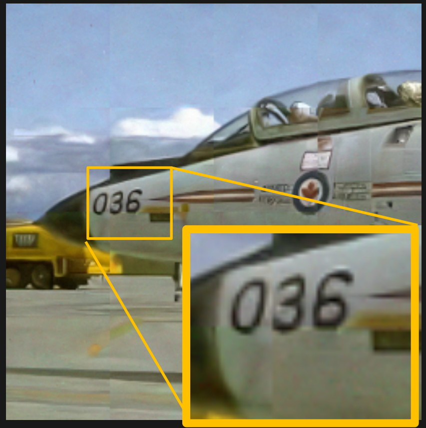
Likelihood Step

Satisfy forward model

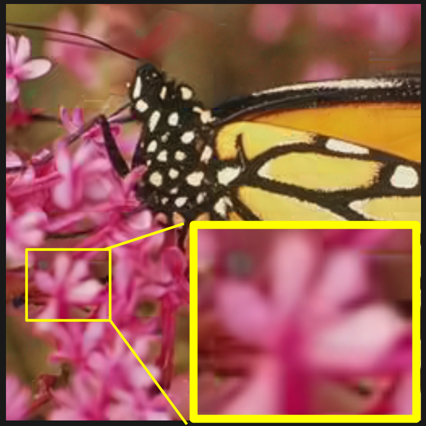
Results

Original Image	TVAL3	Ours

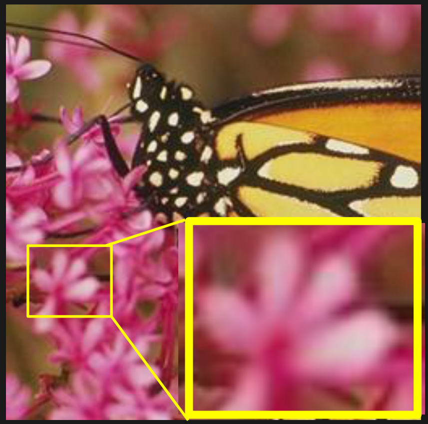
22.54 dB, 0.705



30.51 dB, **0.918**



23.78 dB, 0.917



32.14 dB, **0.987**

Better reconstructions in terms of metrics and pixel-level consistencies

Example 2:

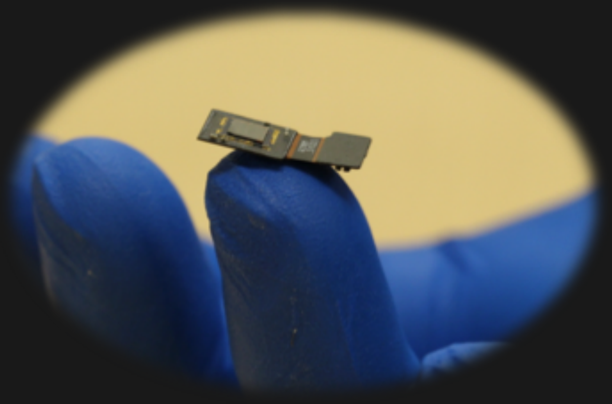
Deep Photorealistic Reconstruction of Lensless Images

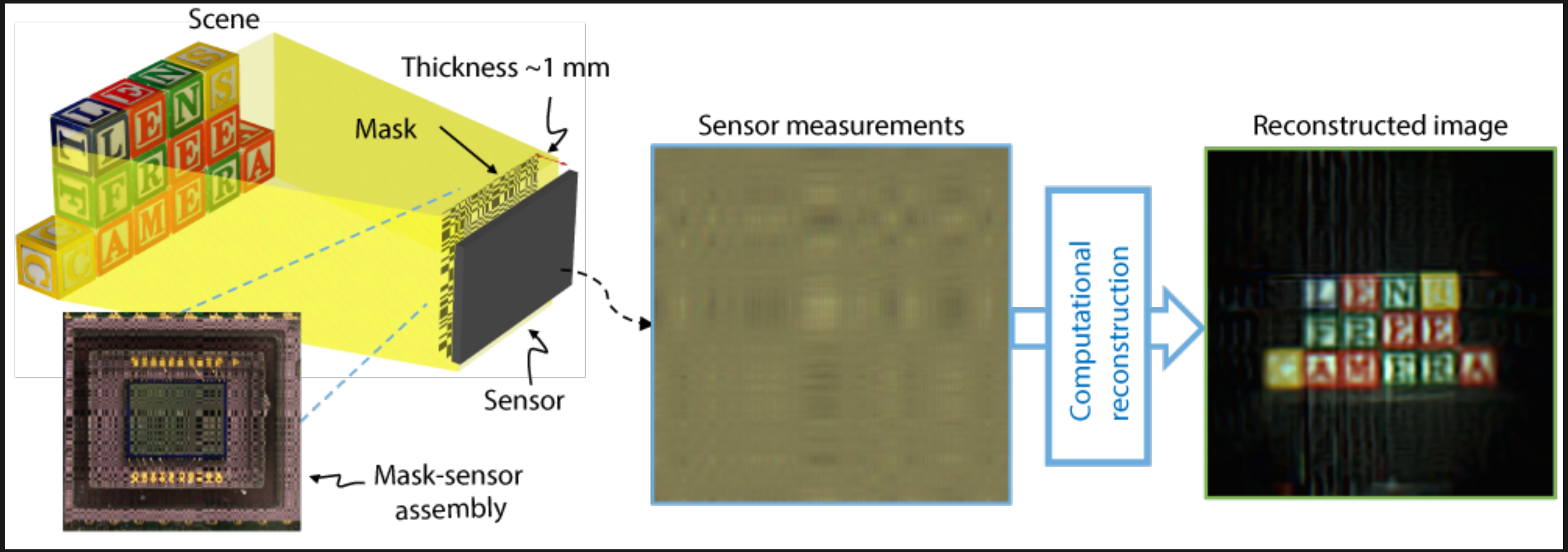
Salman Khan, Adarsh V.R., Vivek Boominathan, Jasper Tan, Ashok Veeraraghavan and Kaushik Mitra.

Under review

Thin Optical System

FlatCam





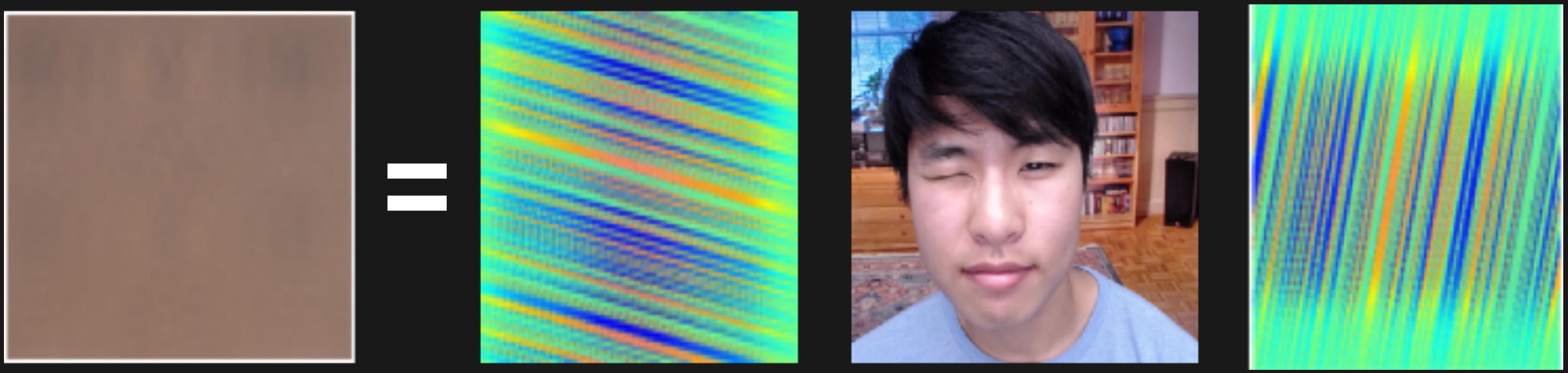
Drastically reducing camera thickness by replacing lens with thin mask

S. Asif et al., *IEEE Transactions on Computational Imaging* (2016)

Ill posed inverse problem

Forward Model:

Capture Scene



Y Φ_L X Φ_R

Ill posed – Φ_L and Φ_R are poorly conditioned

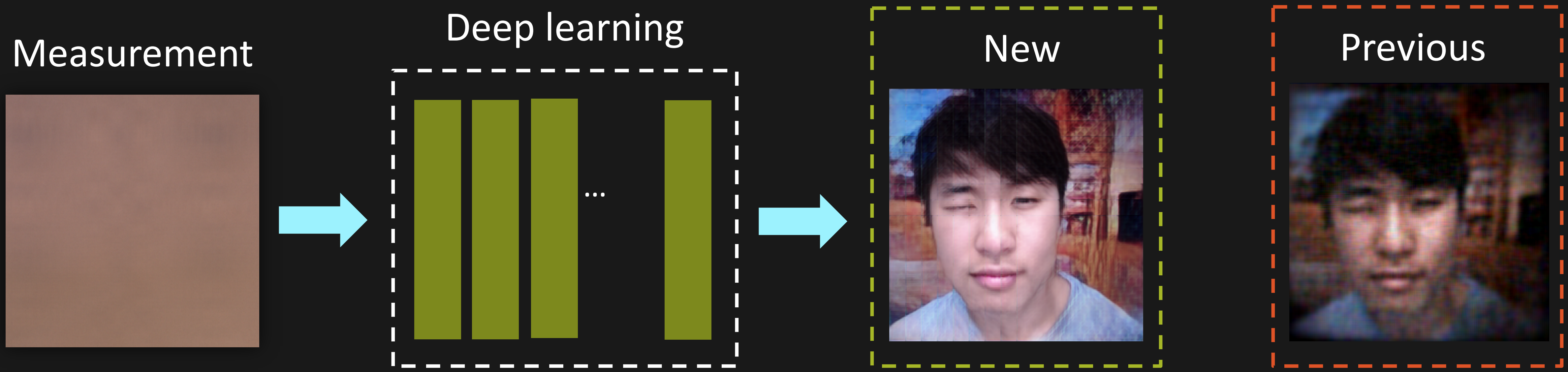
Regularized reconstruction

$$X = \arg\min_{X} ||\Phi_L X \Phi_R^T - Y||_F^2 + \lambda ||X||_F^2$$

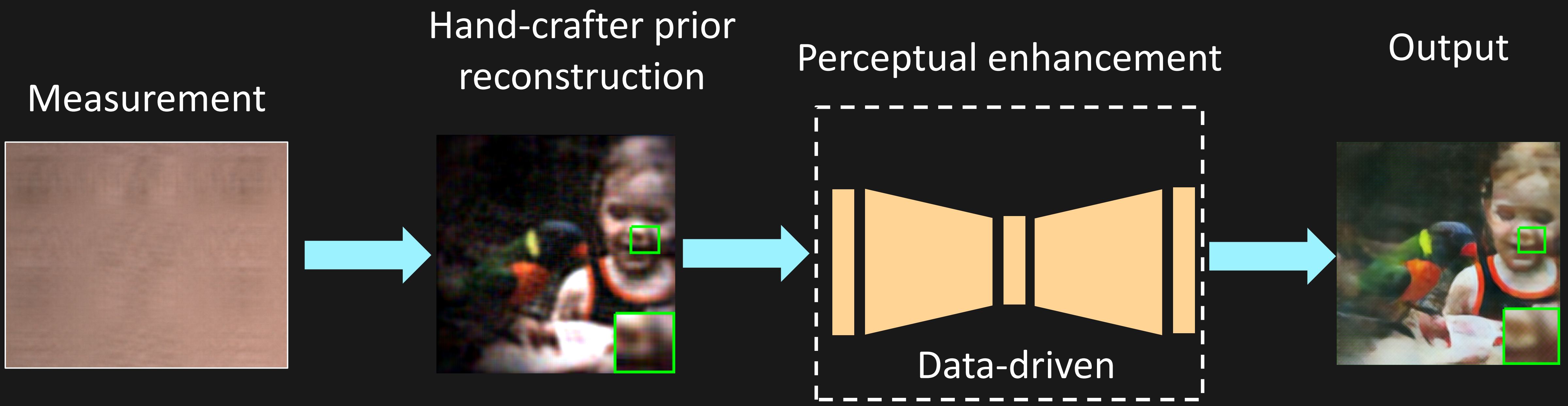
Reconstruction

Regularization

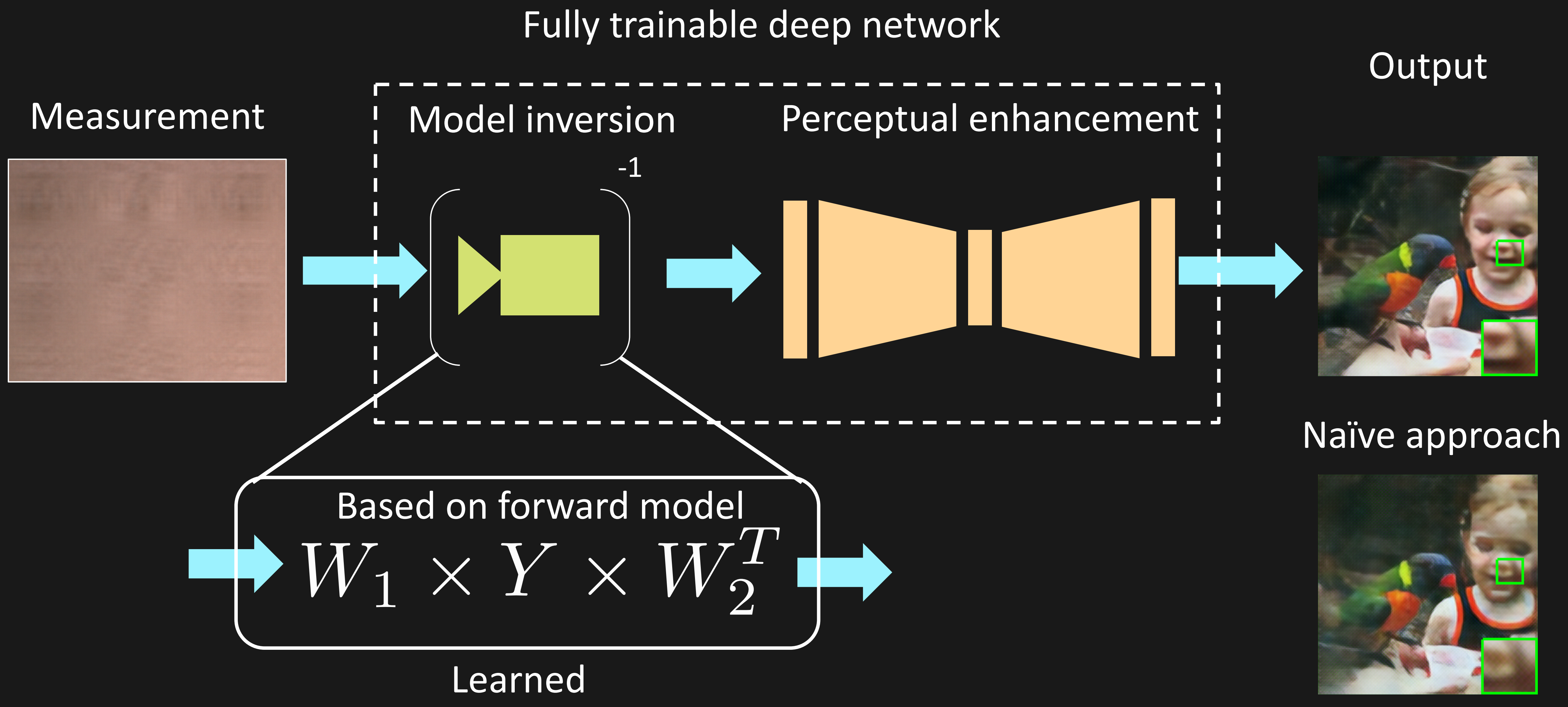
Data-driven reconstruction



Naïve approach

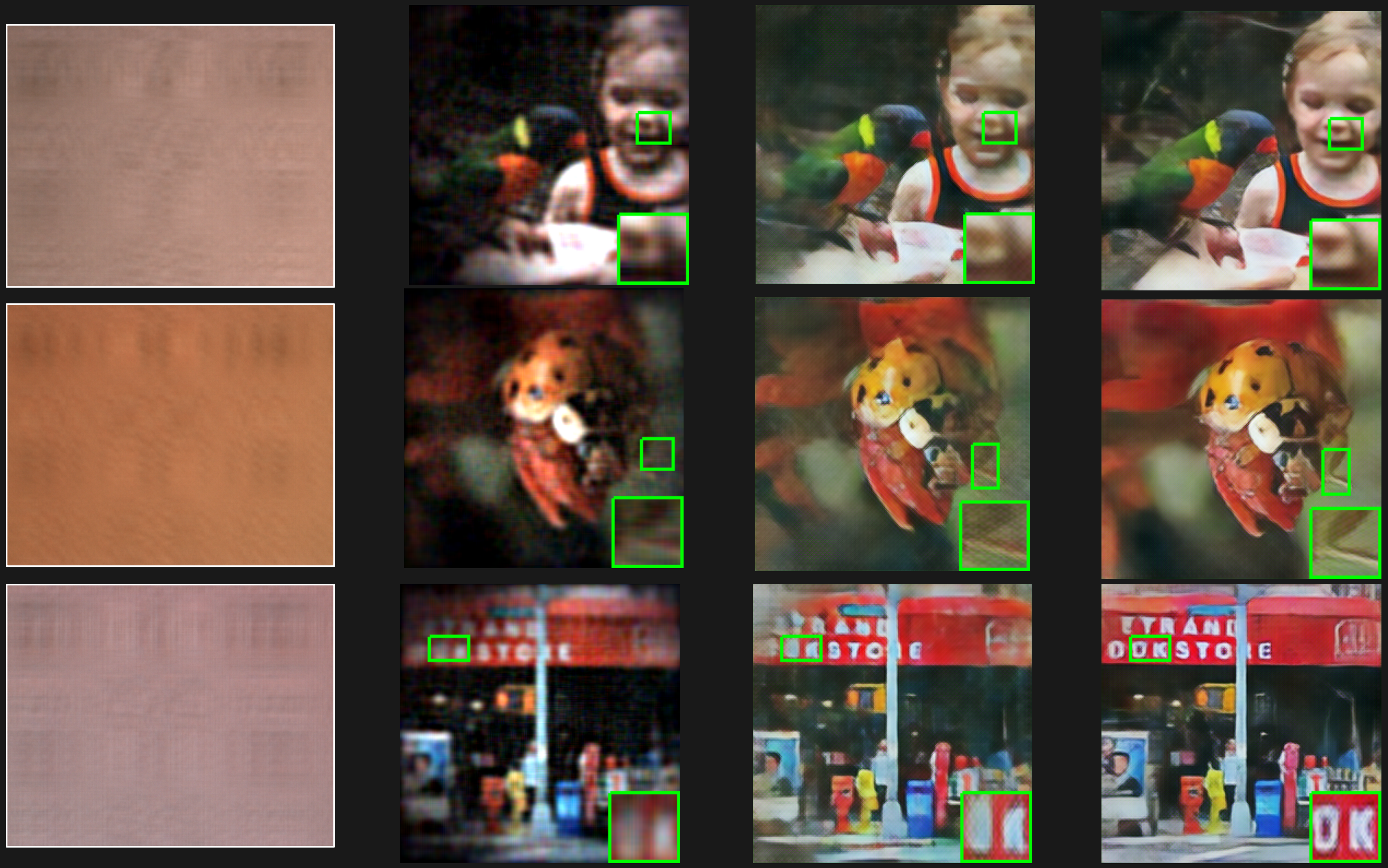


End-to-end approach



Results

Raw Captures | Tikhonov regularization | Naïve | End-to-End

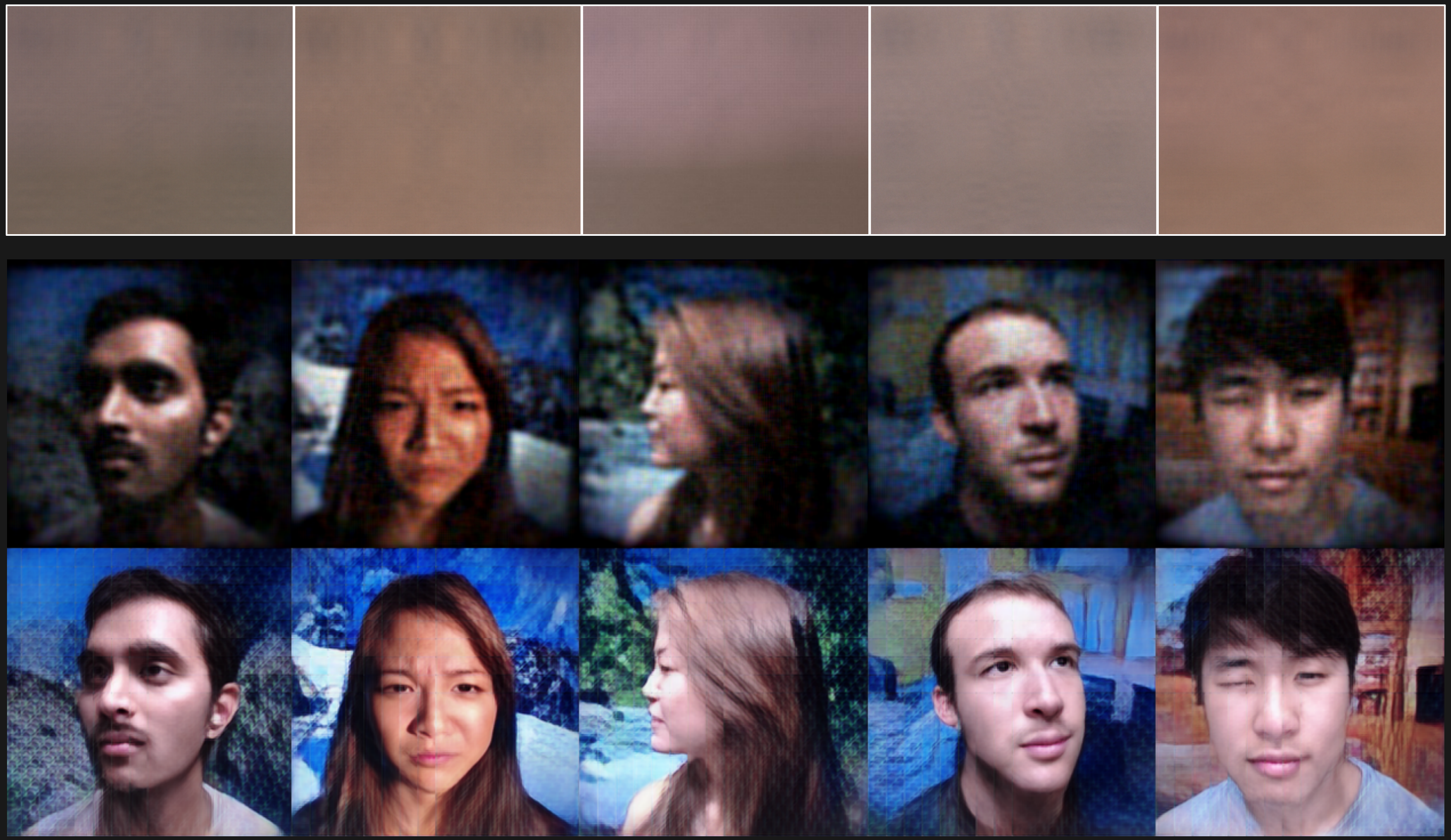


Results

Raw Captures

Tikhonov regularization

Data-driven End-to-End

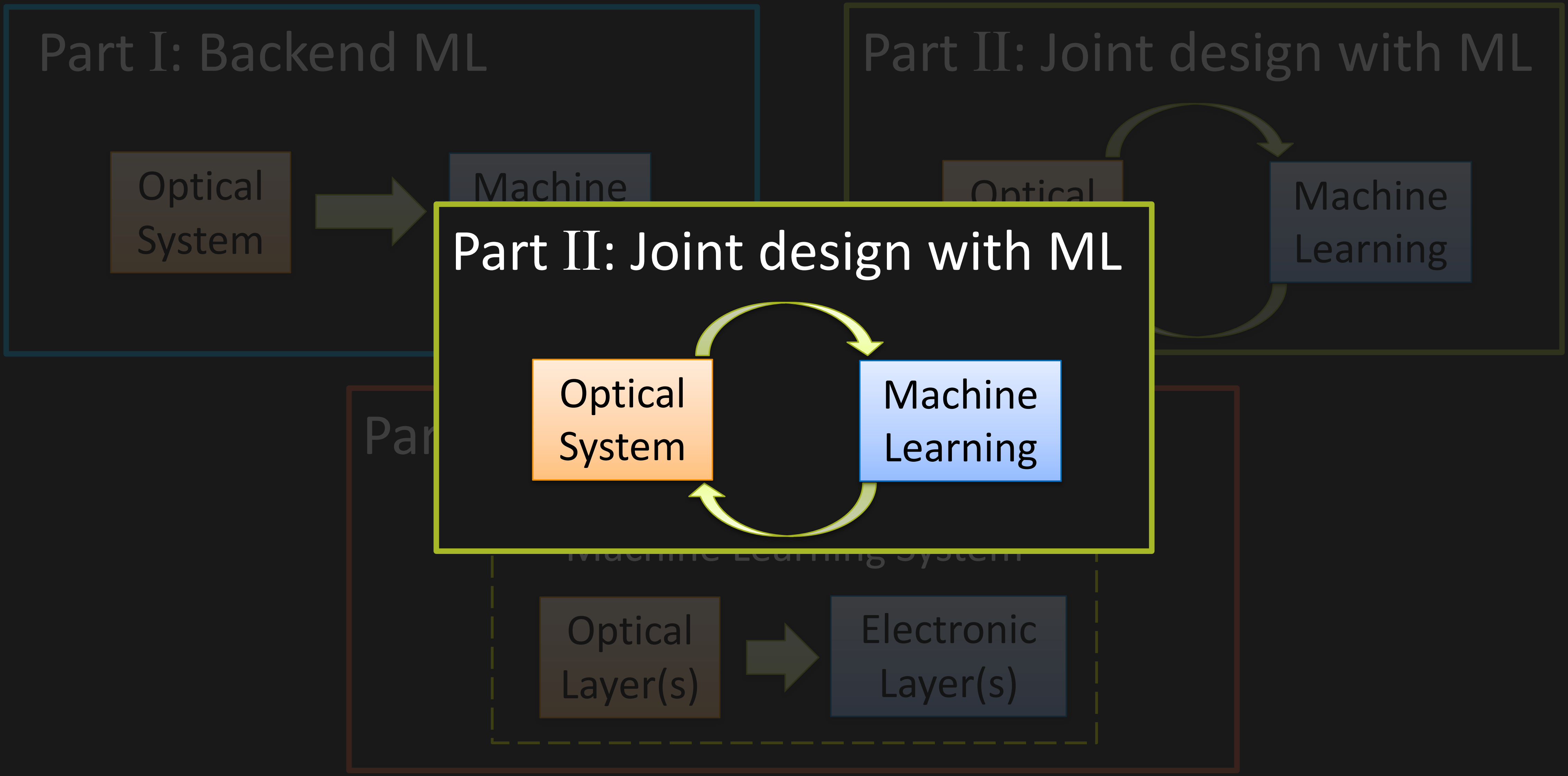


Part I: Summary

Fixed — Optical System → *Data-driven* — Machine Learning → *Improve Quality* — Output

Data-driven method captures good "Visual" priors leading to high-quality reconstructions

Visual Sensing using Machine Learning



Optical system design

Independent-crafted design

- Motivated by signal processing theory
- Reconstruction algorithm developed separately and may not compensate for drawbacks in design
- Doesn't achieve best optimal combination

Data-driven design

- Design and reconstruction algorithm tied tightly
- Design is optimized to bring the best out of reconstruction algorithm
- Jointly produce optimal system

Joint (optical + algorithm) design lead to high-quality reconstructions

Part II: Joint design with ML

Example 1:

PhaseCam3D - Learning Phase Masks For Passive Single-view Depth Estimation

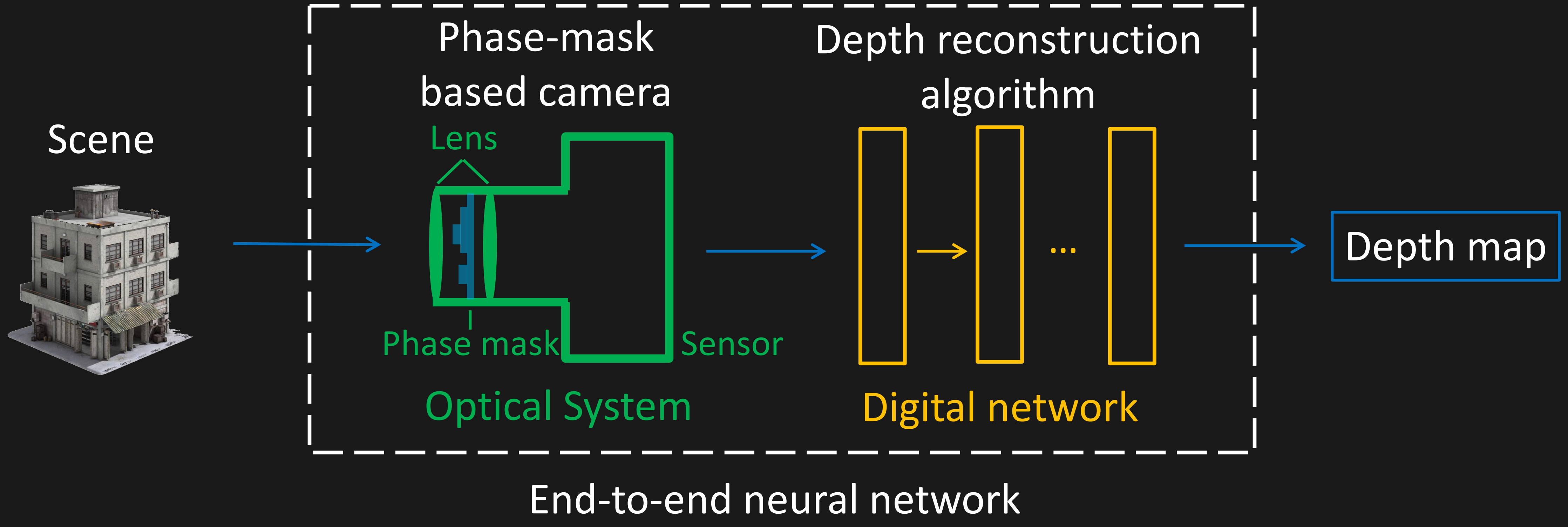
Yicheng Wu, Vivek Boominathan, Huaijin Chen, Aswin Sankaranarayanan, and Ashok Veeraraghavan.
“PhaseCam3D—Learning Phase Masks for Passive Single View Depth Estimation.”
IEEE International Conference on Computational Photography (ICCP), 2019

Carnegie Mellon University

RICE COMPUTATIONAL IMAGING

PhaseCam3D sensor

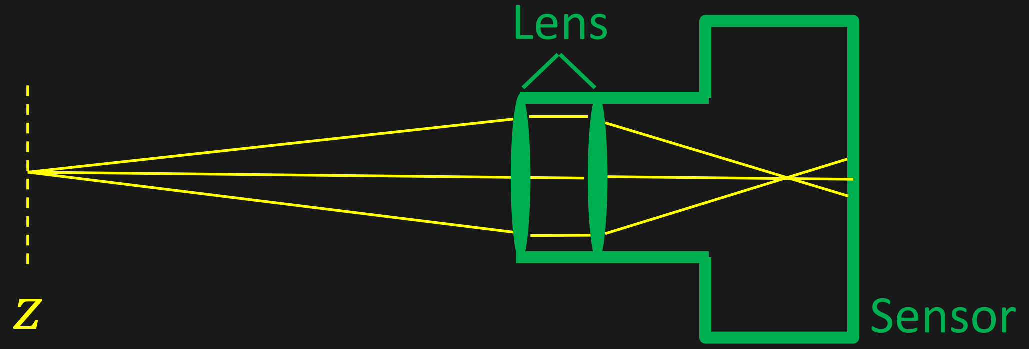
Part II: Joint design with ML



Defocus of general lens

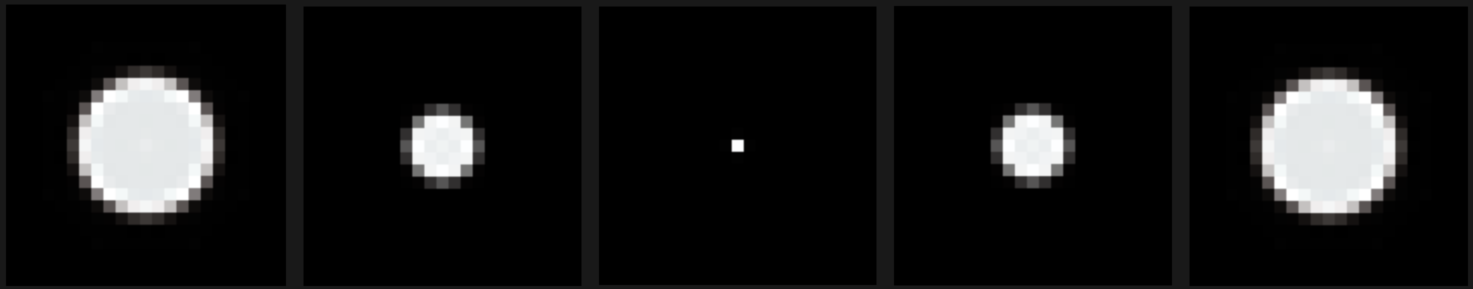
Defocused image

trentwoodsphoto.com



PSFs at different depths

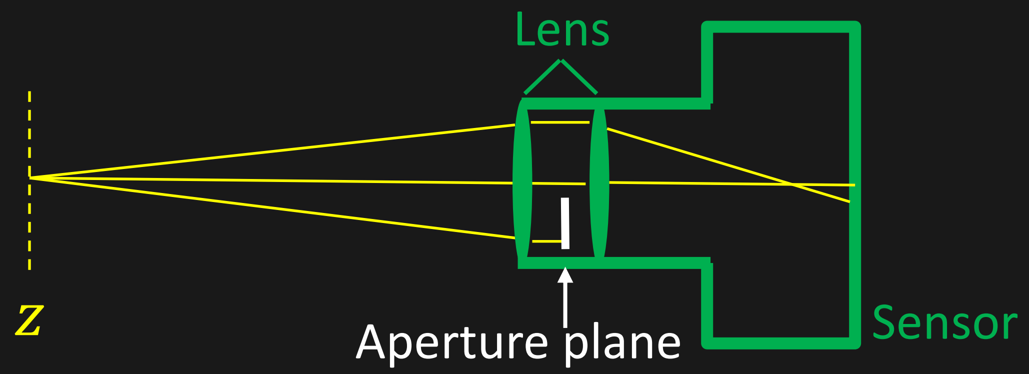
Far ⟵ Focus ⟶ Near

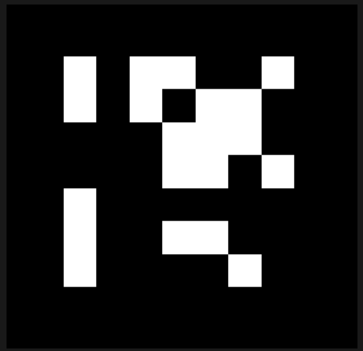


- × Identical PSF at both sides of the focal plane.
- × Impossible to tell the depth based on the blur size.

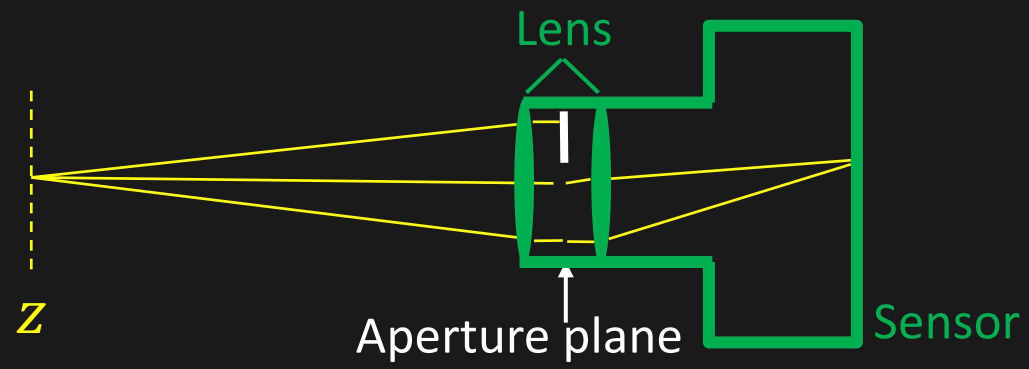
Independent-crafted designs

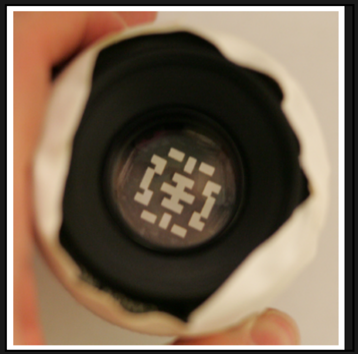
Veeraraghavan *et al.*, 2007

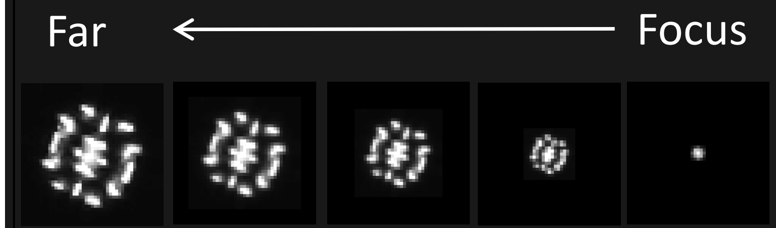




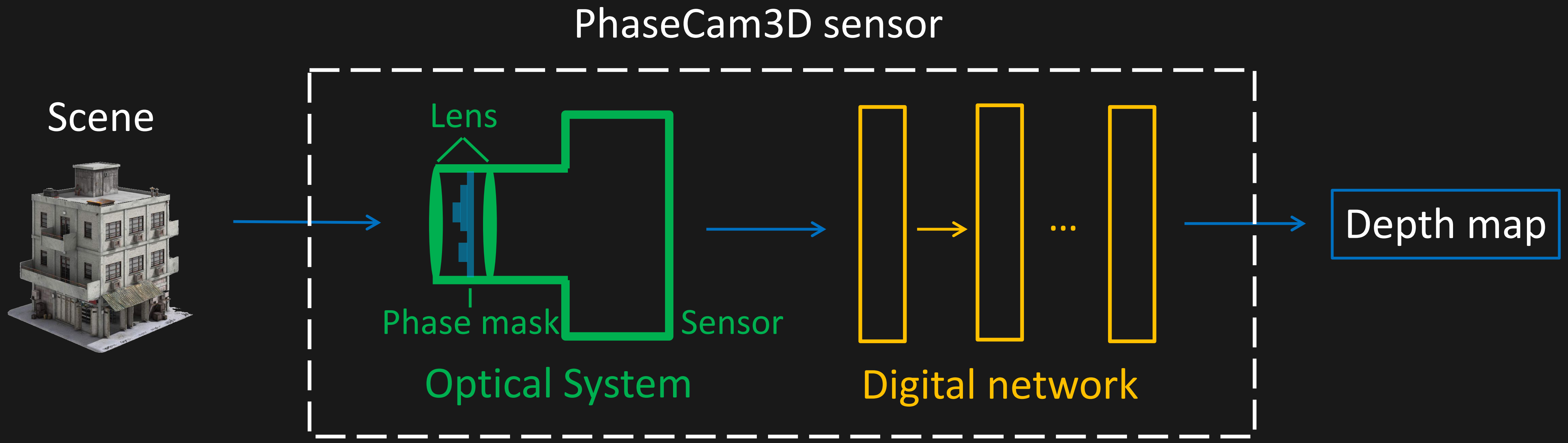
Levin *et al.*, 2007







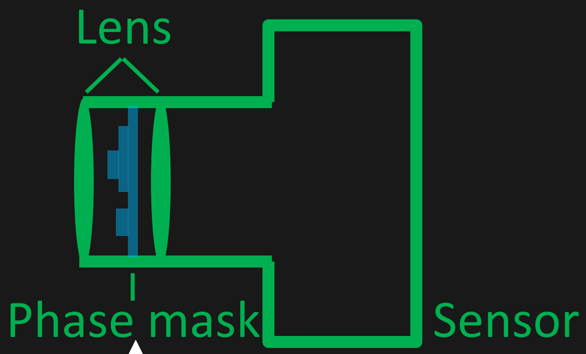
PhaseCam3D sensor



- ❑ Differential optical model
- ❑ Digital network
- ❑ End-to-end learning

Model for the optical system

Object



Pupil function P

PSF formulation

$$PSF = |F\{P\}|^2$$

$$P = \exp[j\phi_\lambda^M(h) + j\phi_\lambda^{DF}(z)]$$

Phase mask Defocus

Image formulation

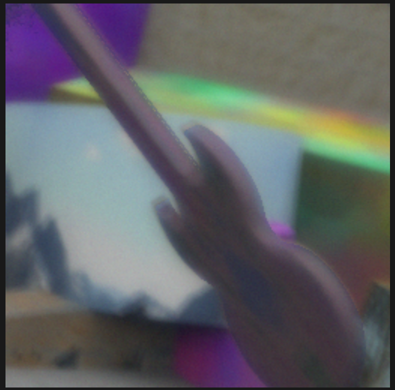
$$I_{sensor}(h) = \sum_z I_{obj}(z) \otimes PSF_\lambda(h, z) + N(0, \sigma^2)$$

Noise

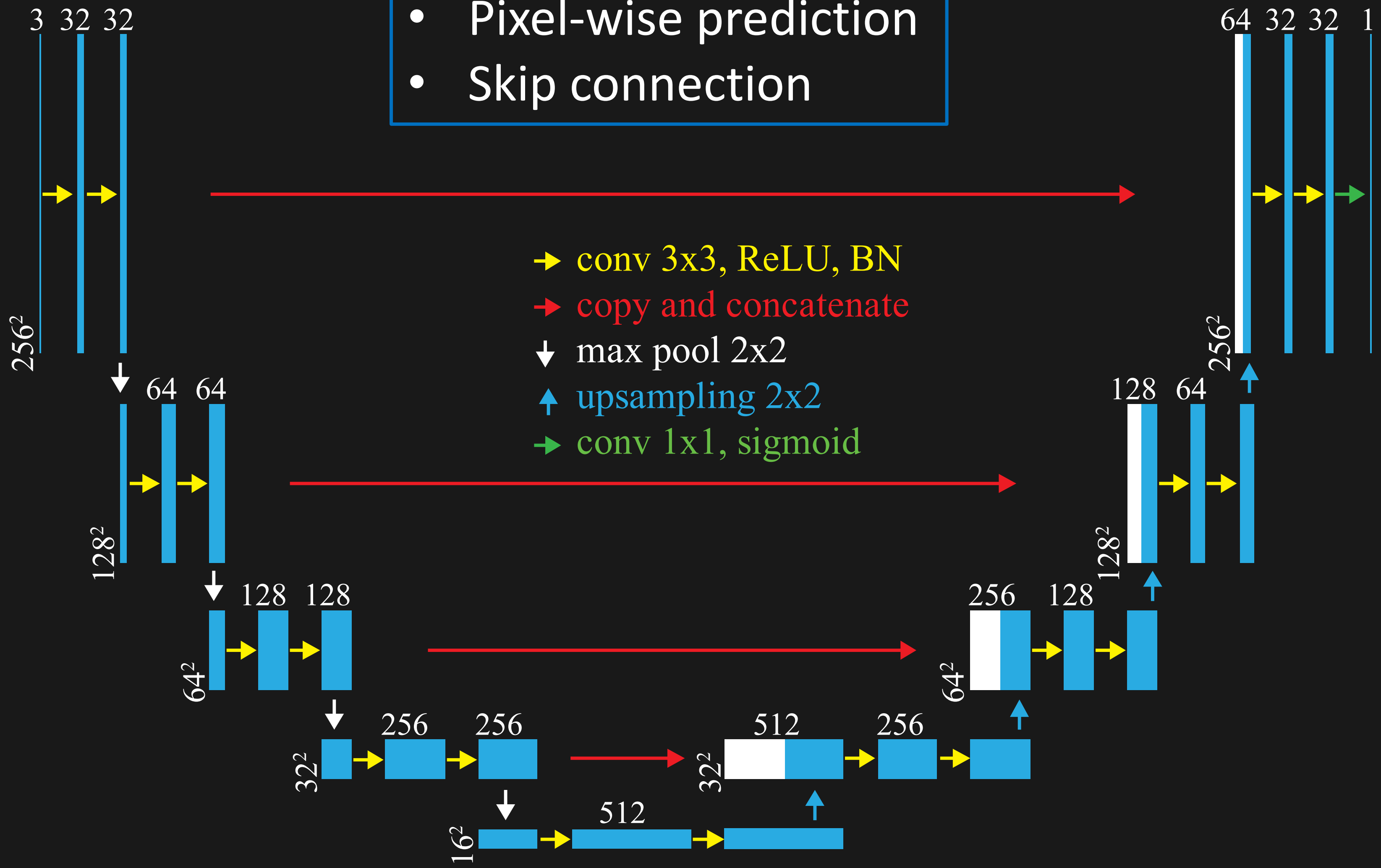
Differentiable model

Model for the digital network

Coded image

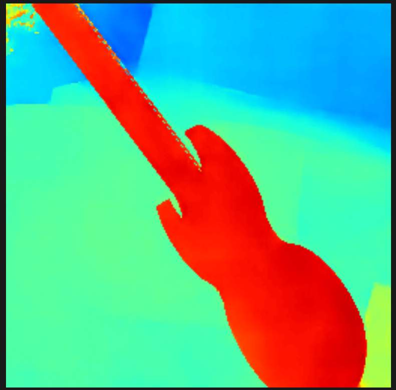


U-Net



- Pixel-wise prediction
- Skip connection

Estimated depth



Train the network

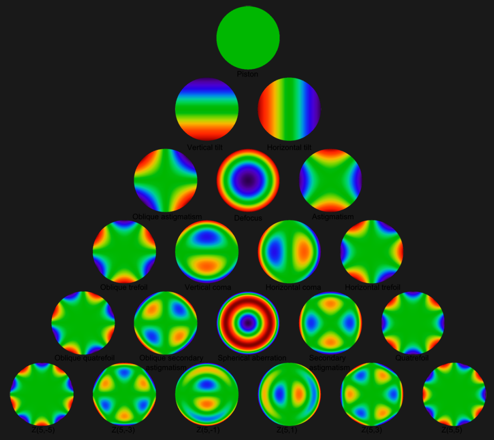
- RGBD dataset
  - Experimental: boundary mismatch, missing depth
  - Synthetic: precise texture and depth

- Loss functions
  - Root mean square (RMS) loss
  - Gradient loss

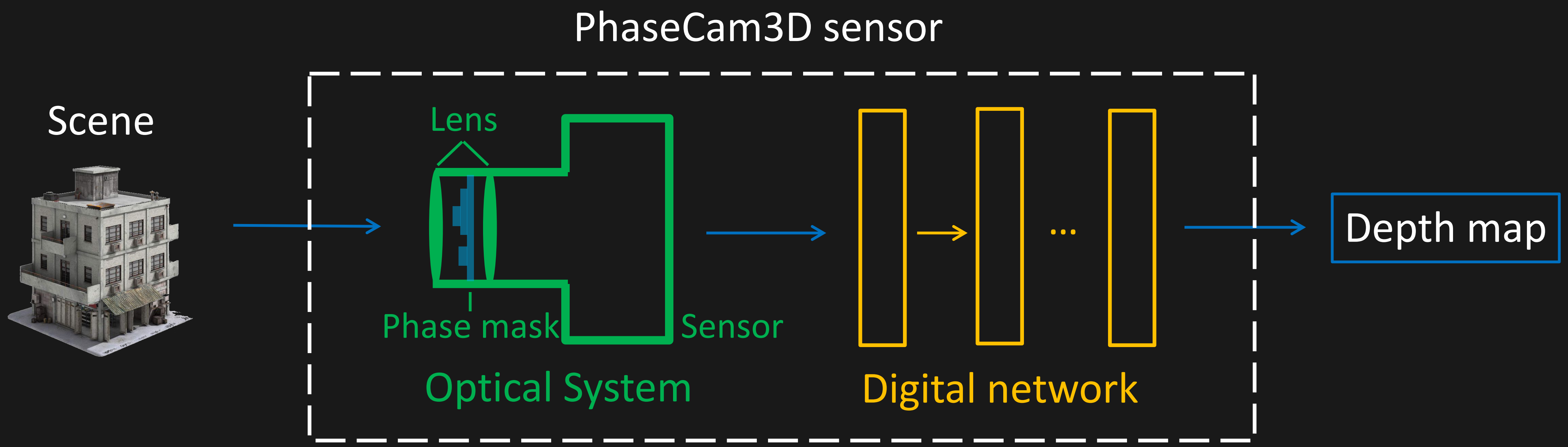
- Mask parameters
  - Coefficients of Zernike polynomial basis

FlyingThings3D

Zernike polynomials



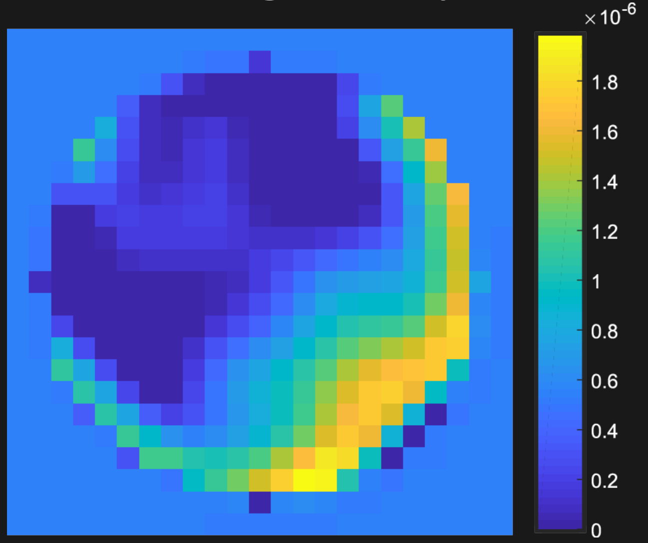
PhaseCam3D: an end-to-end learning approach



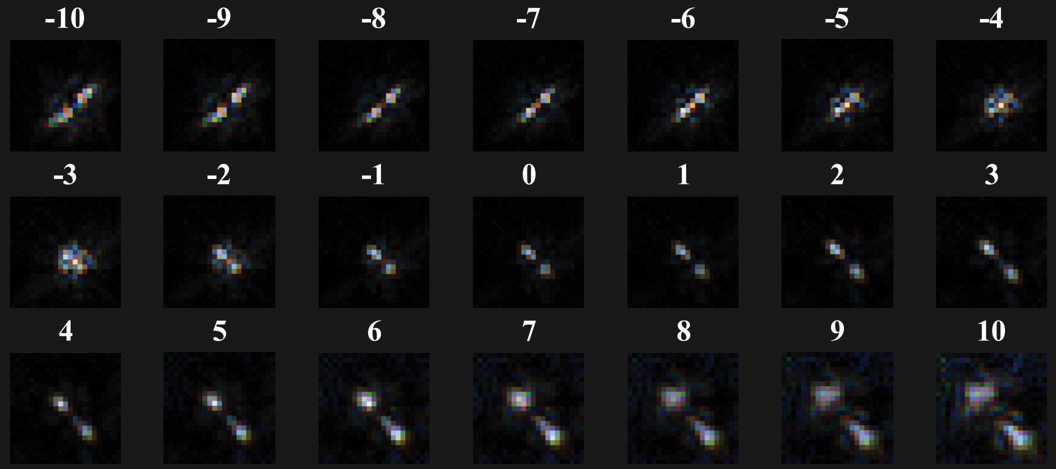
- ☑ Differential optical model
- ☑ Digital network
- ☑ End-to-end learning

Optimal simulation results

Height map



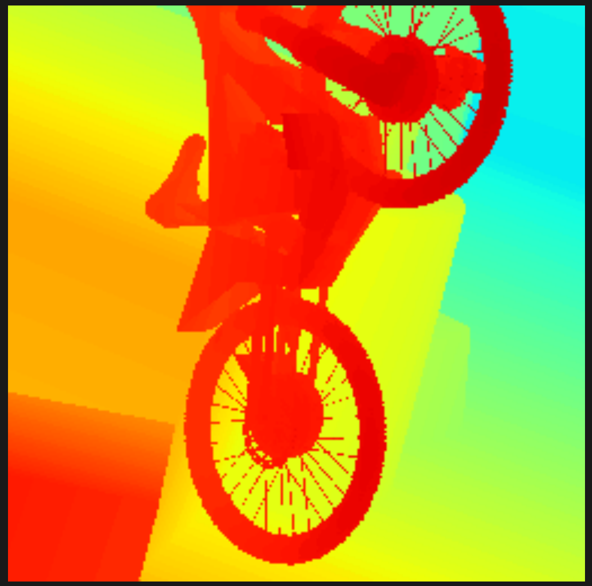
PSFs



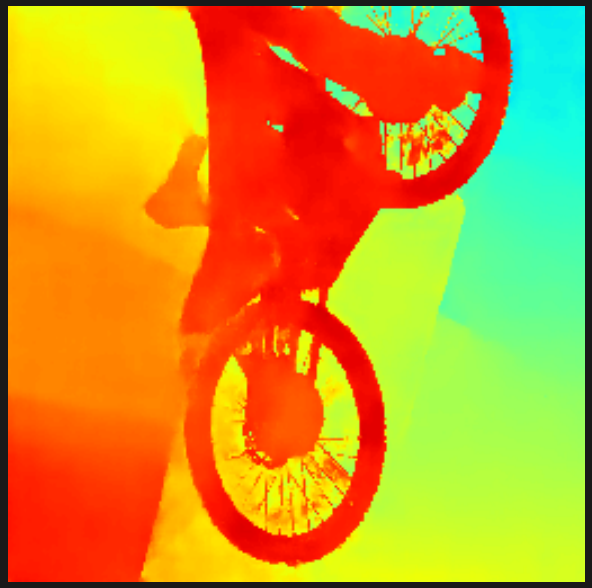
Sharp image

Coded image

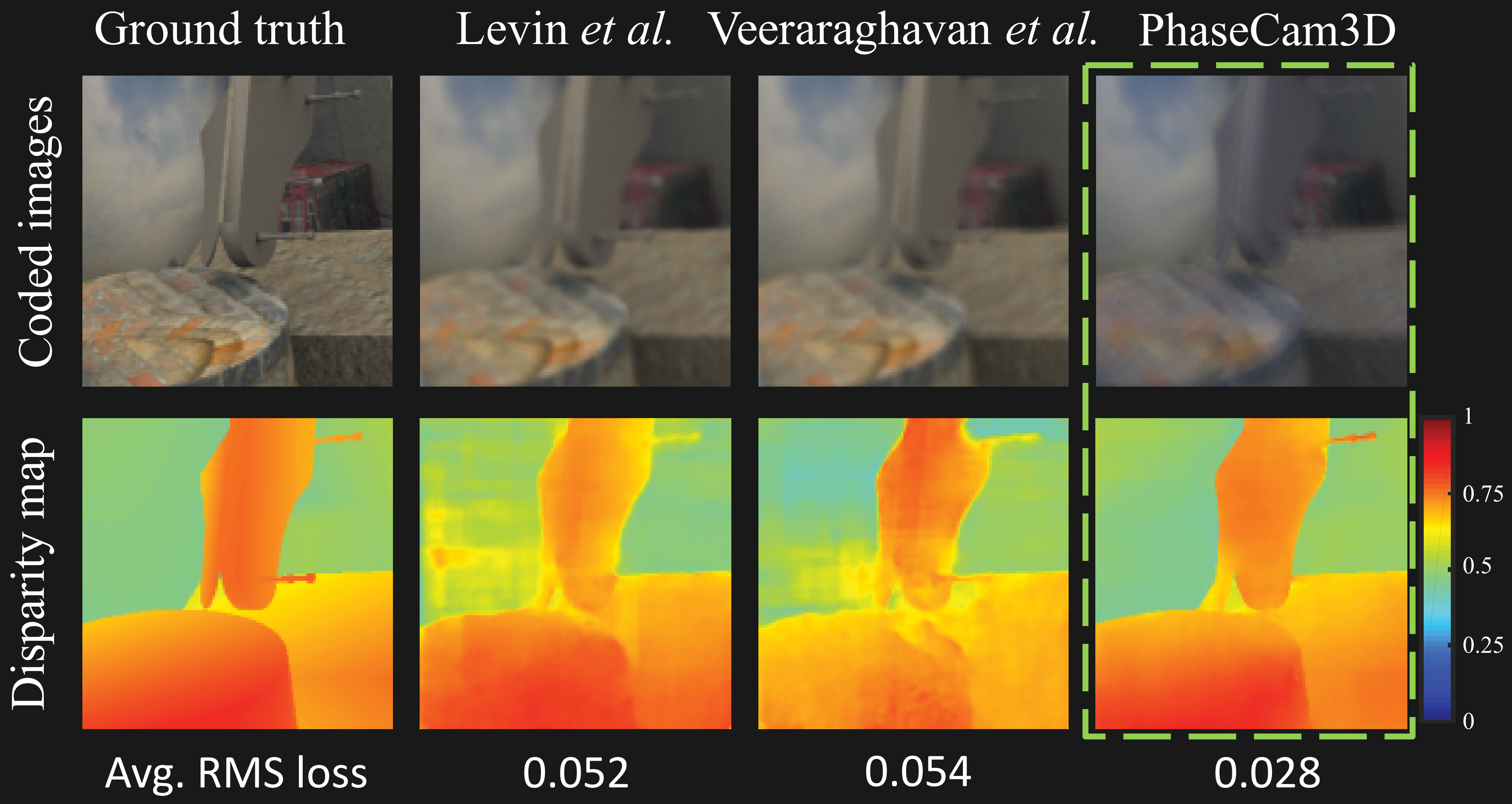
True disparity



Estimated disparity

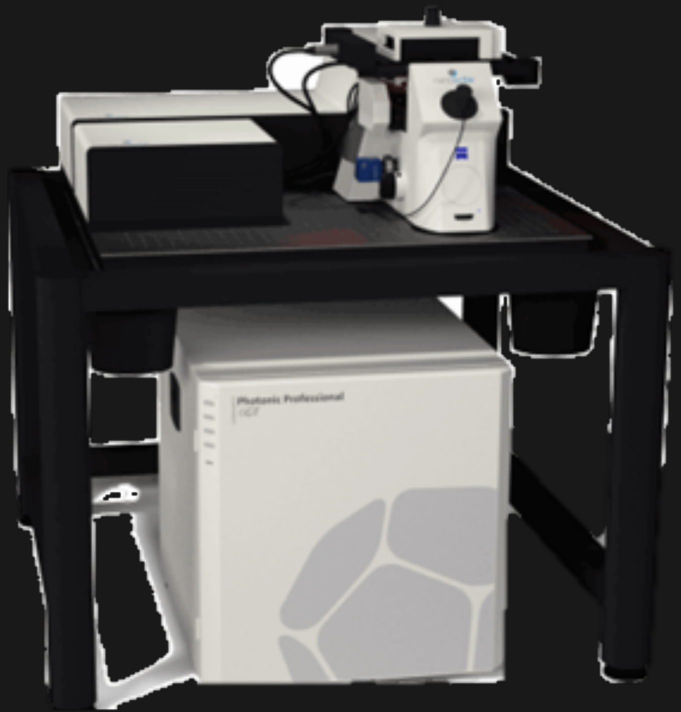


Comparison with independent-crafted designs



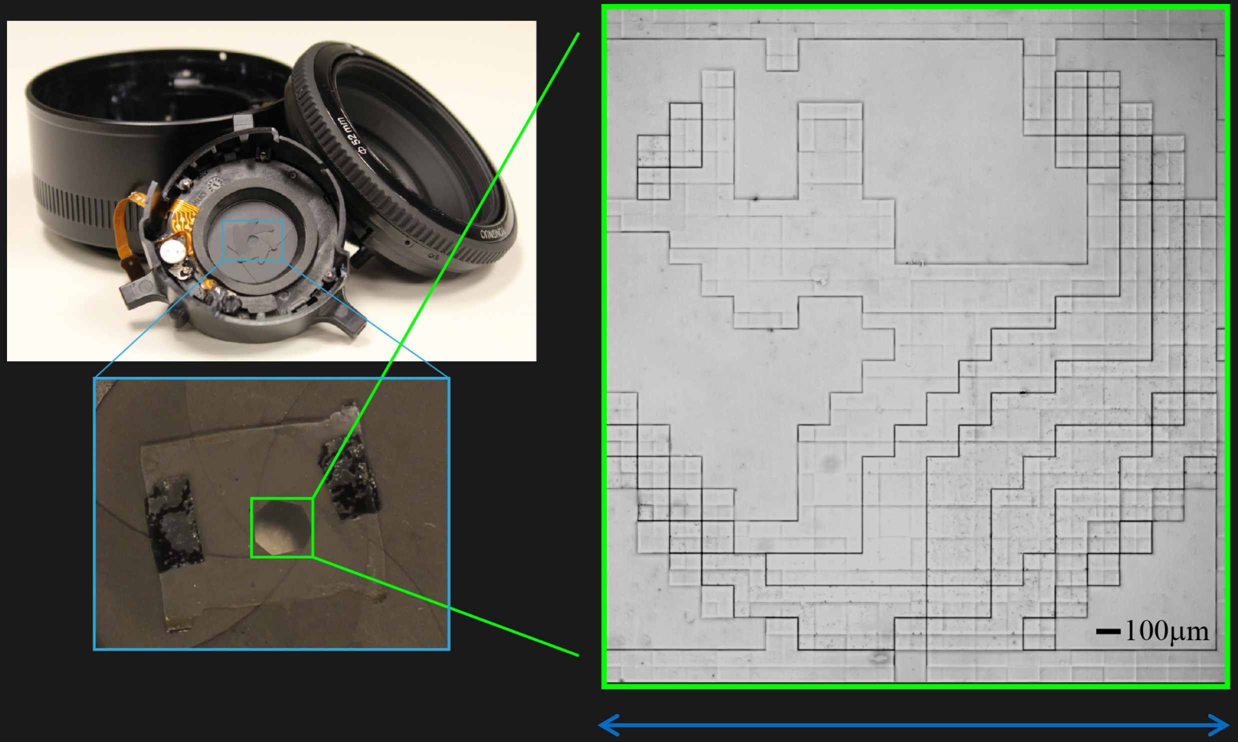
Fabricate the learned phase mask

Two-photon lithography 3D printer

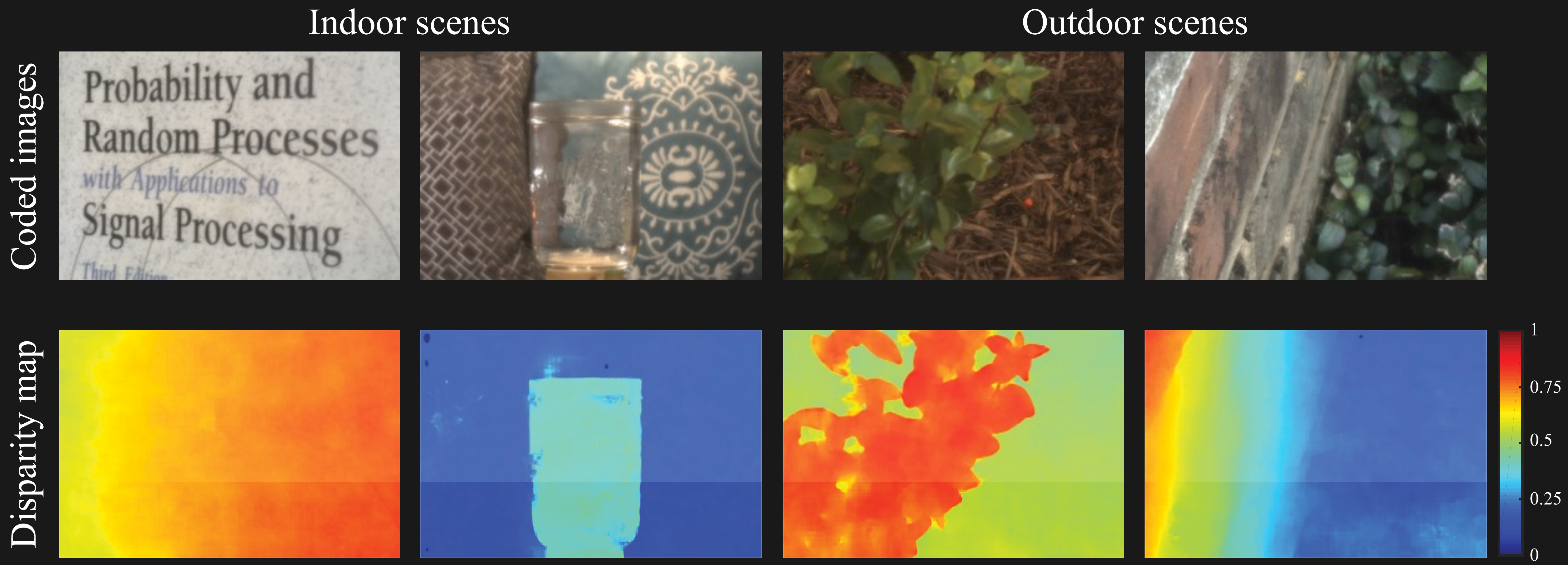


Photonic Professional *GT*, Nanoscribe GmbH

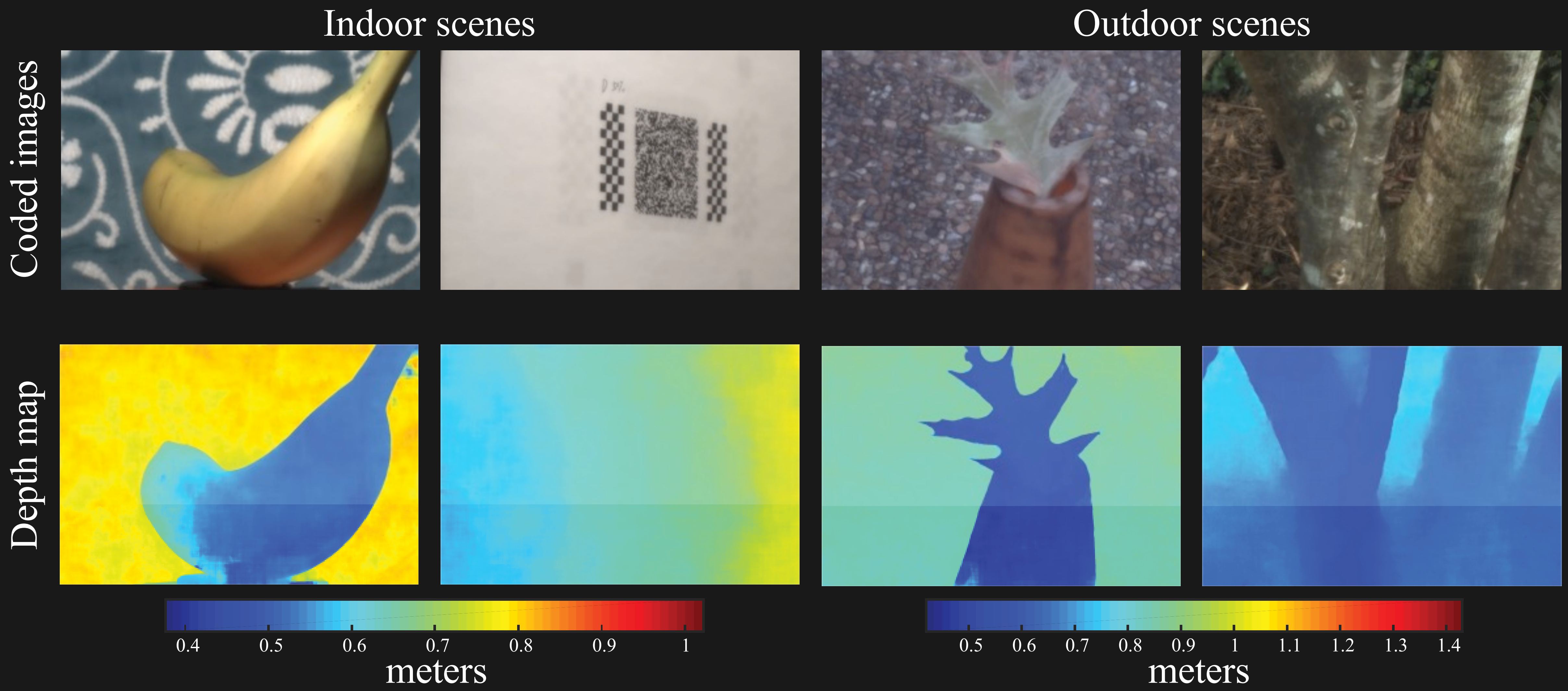
Fabricated phase mask



Experimental results



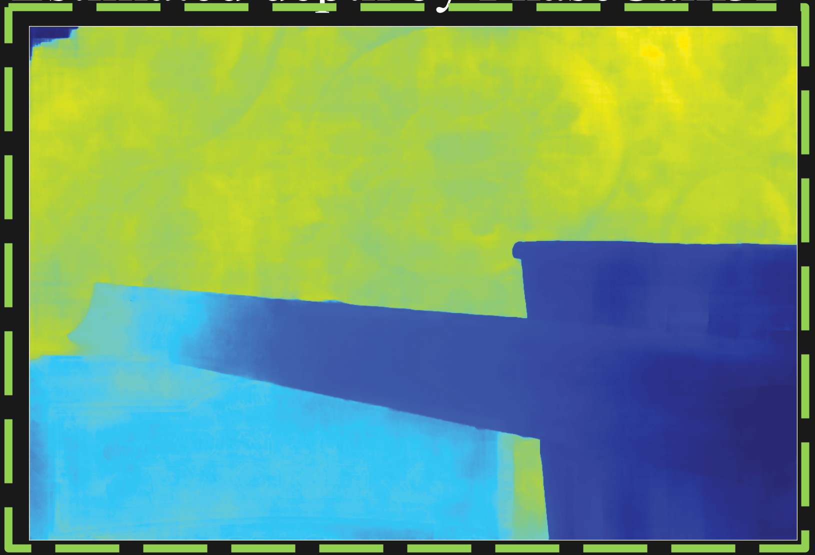
Experimental results



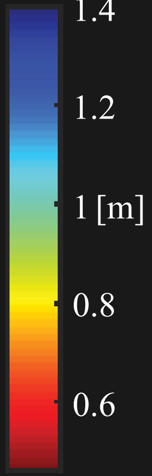
Accuracy evaluation: compare with Kinect

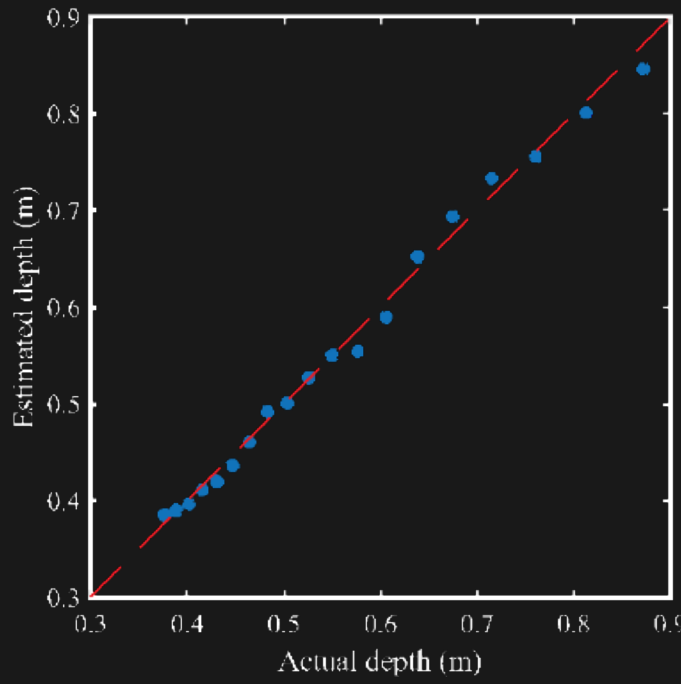
Coded Images

Estimated depth by PhaseCam3D



Estimated depth by Kinect





Error:

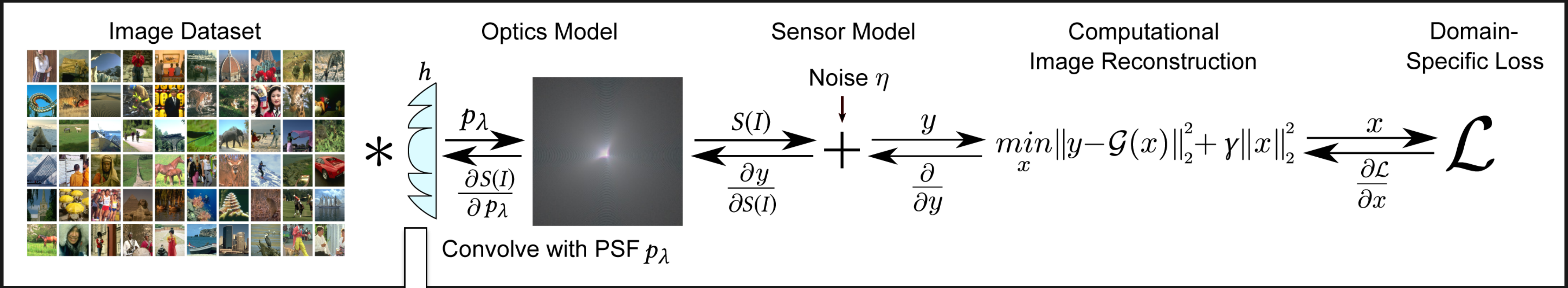
$\sigma_{PhaseCam3D} = 1.25\text{cm}$

Example 2:

End-to-end Optimization of Optics and Image Processing for Achromatic EDOF

Vincent Sitzmann, Steven Diamond, Yifan Peng, Xiong Dun, Stephen Boyd, Wolfgang Heidrich, Felix Heide, and Gordon Wetzstein. "End-to-end optimization of optics and image processing for achromatic extended depth of field and super-resolution imaging." *ACM Transactions on Graphics* (TOG), 2018.

Joint design of optics and Image reconstruction



Optical system:
Parameterized Lensing Element

Sitzmann *et al.*, TOG, 2018

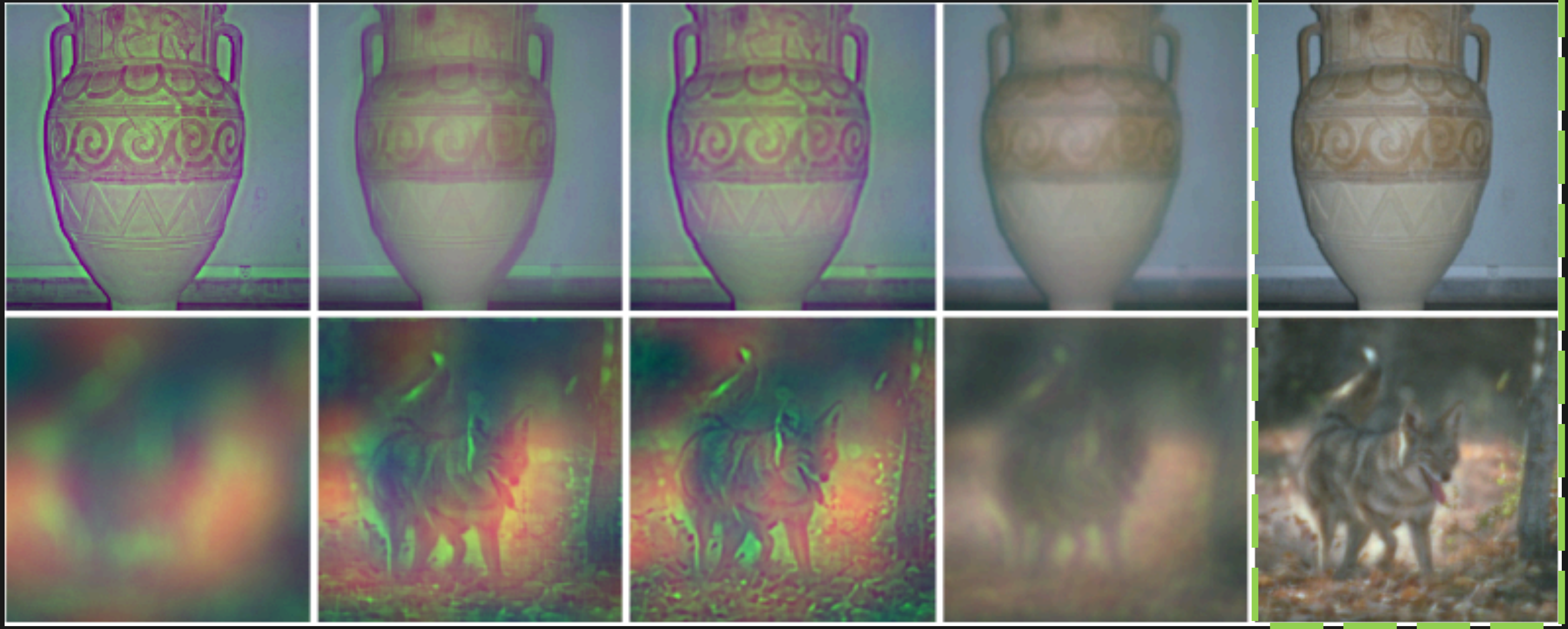
Results – Achromatic EDOF

Independent-crafted optics

Fresnel Lens | Cubic Phase Lens | Multi-focal Lens | Diffractive Achromat | End-to-end Optimized

Depth: 1.0m

Depth: 0.5m



Dowskey & Cathey, 95

Peng et. al., 2016

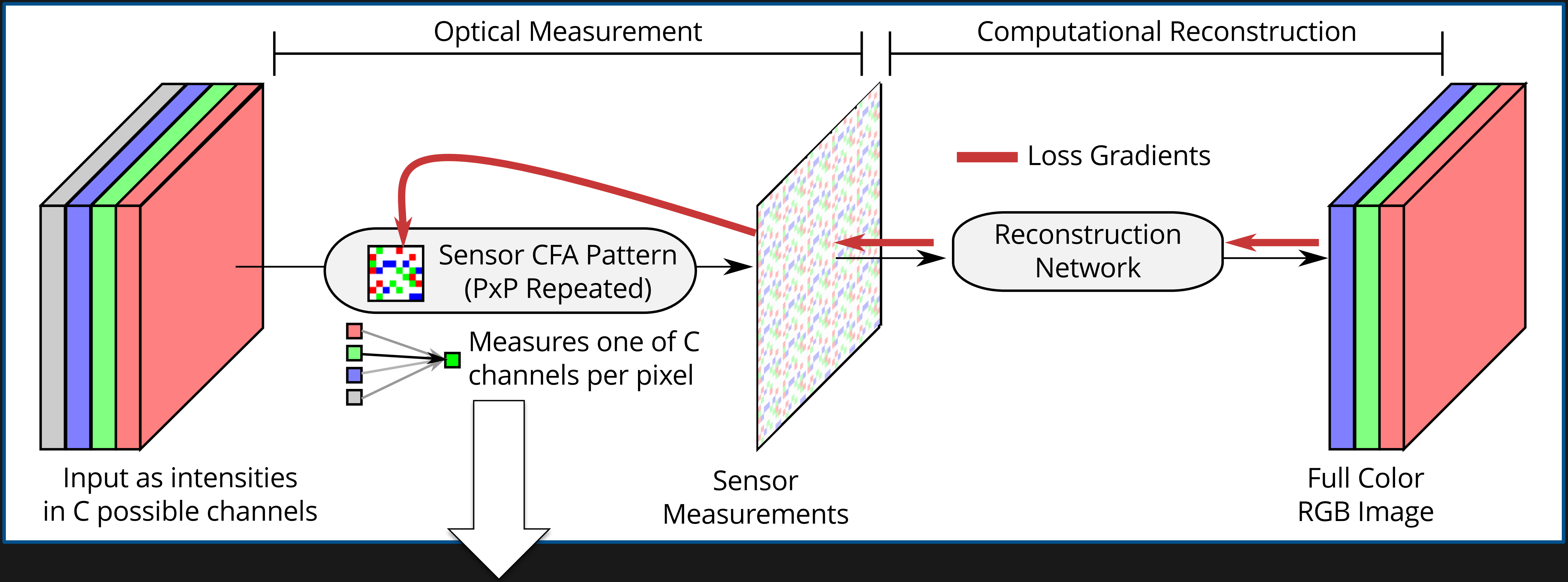
Sitzmann *et al.*, TOG, 2018

Example 3:

Learning Sensor Multiplexing Design through Back-propagation

Ayan Chakrabarti.
"Learning sensor multiplexing design through back-propagation."
In *Advances in Neural Information Processing Systems* (NeurIPS), 2016.

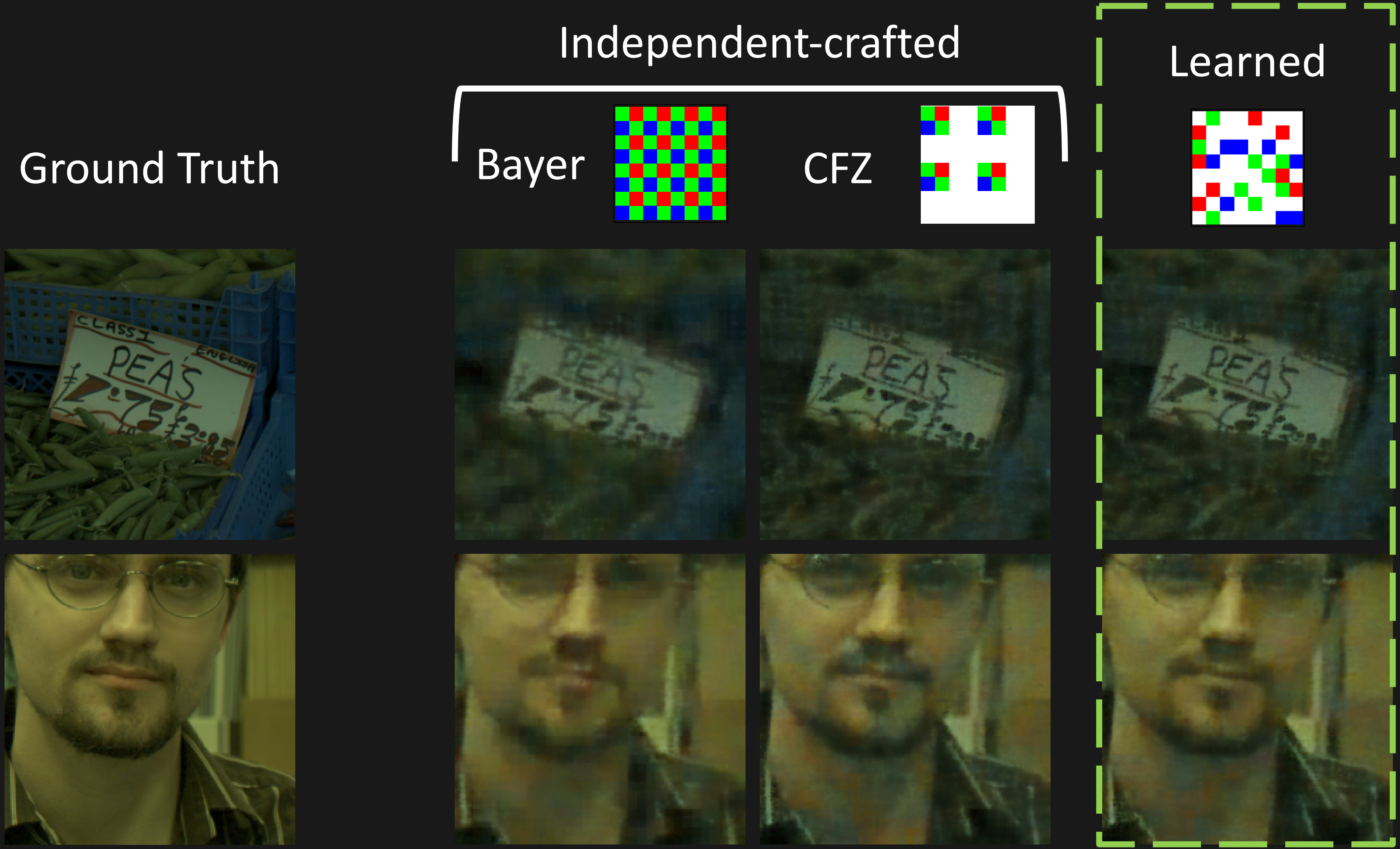
Joint design of color multiplexing and demosaicking



Optical system:
Sensor color filter array

Ayan Chakrabarti, NeurIPS, 2016

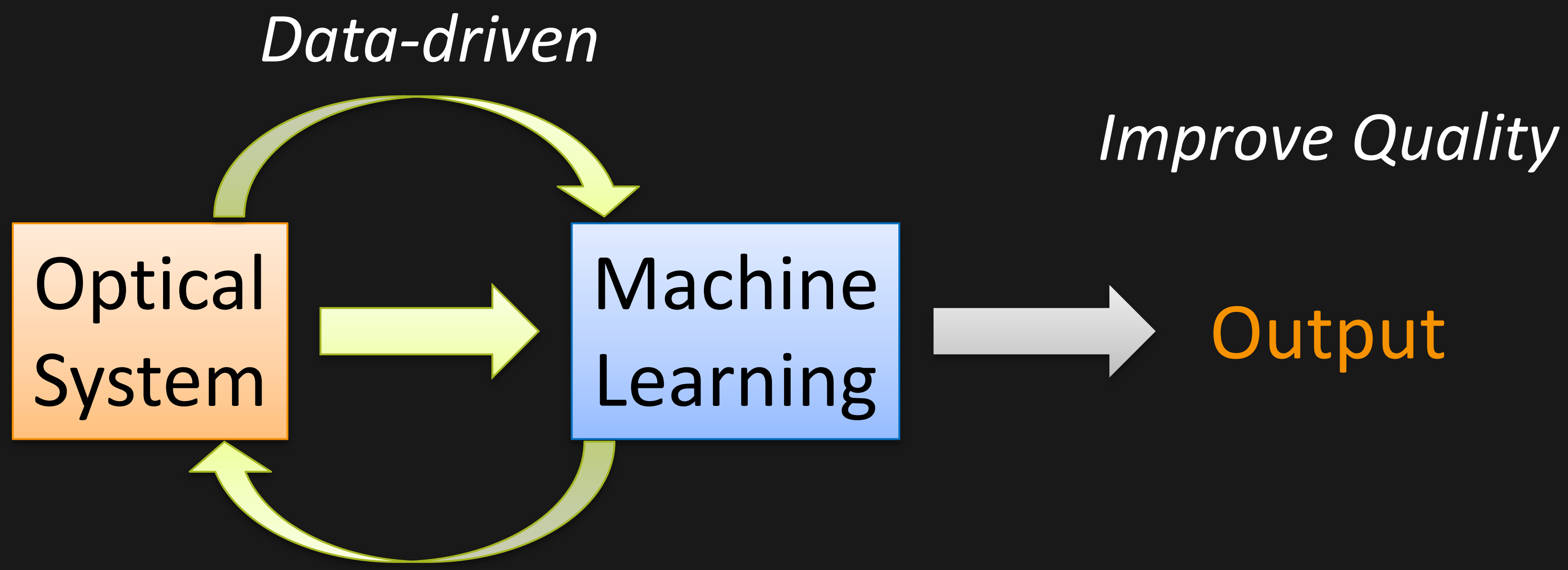
Results – Color demultiplexing



ICCP 14

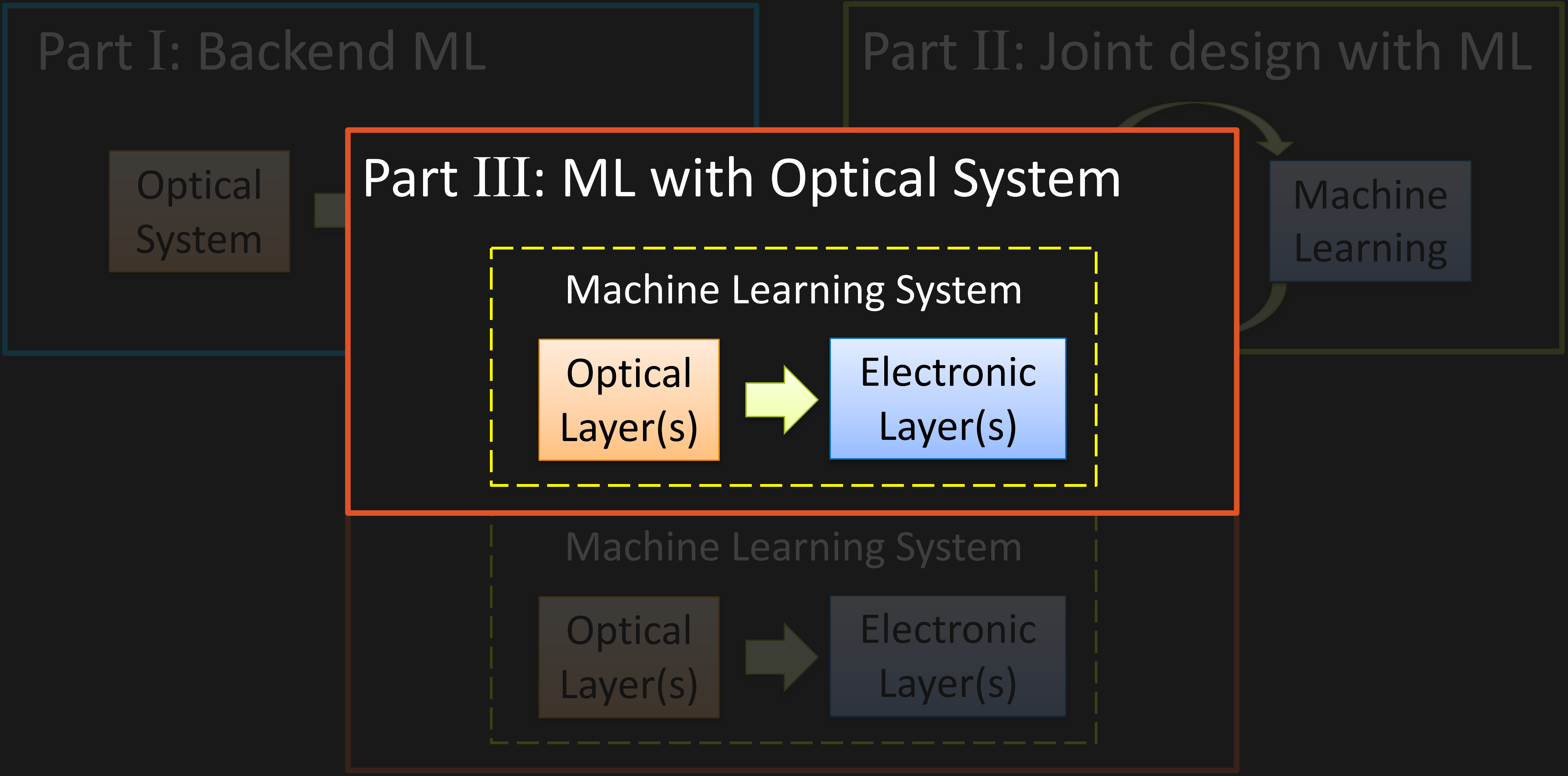
Ayan Chakrabarti, NeurIPS, 2016

Part II: Summary



Joint design with data-driven techniques can bring out best in both systems

Visual Sensing using Machine Learning



ML Optical layer

Traditional Vision

- Camera captures sharp image
- ML algorithm extracts features to perform inference
- All computation is electronic, consuming energy

ML with optical layer

- Optical layer directly extracts features
- Electronic layer uses the features for inference
- Reduced power consumption

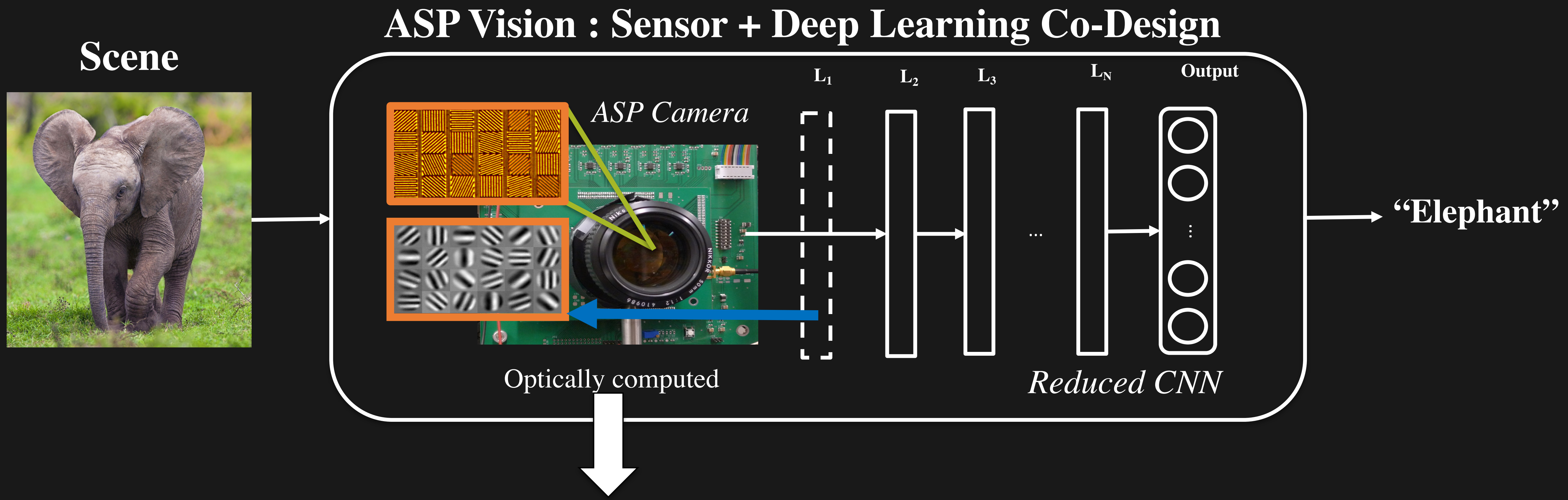
Optical layer can extract features directly that can be used for inference

Example 1:

ASP Vision: Optically Computing the First Layer of CNNs using Angle Sensitive Pixels

Huaijin G. Chen, Suren Jayasuriya, Jiyue Yang, Judy Stephen, Sriram Sivaramakrishnan, Ashok Veeraraghavan, and Alyosha Molnar.
“ASP vision: Optically computing the first layer of convolutional neural networks using angle sensitive pixels.”
Computer Vision and Pattern Recognition (CVPR), 2016.

ASP camera as first layer of DNN

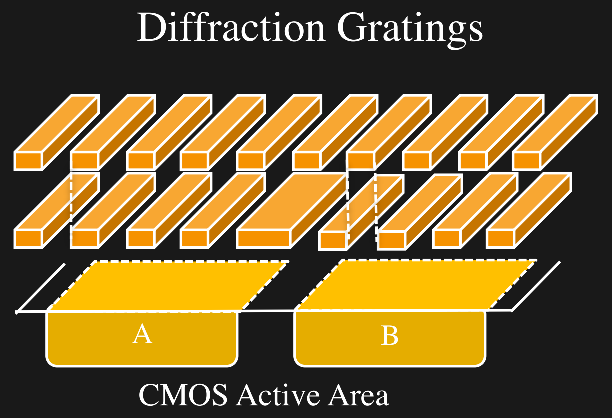


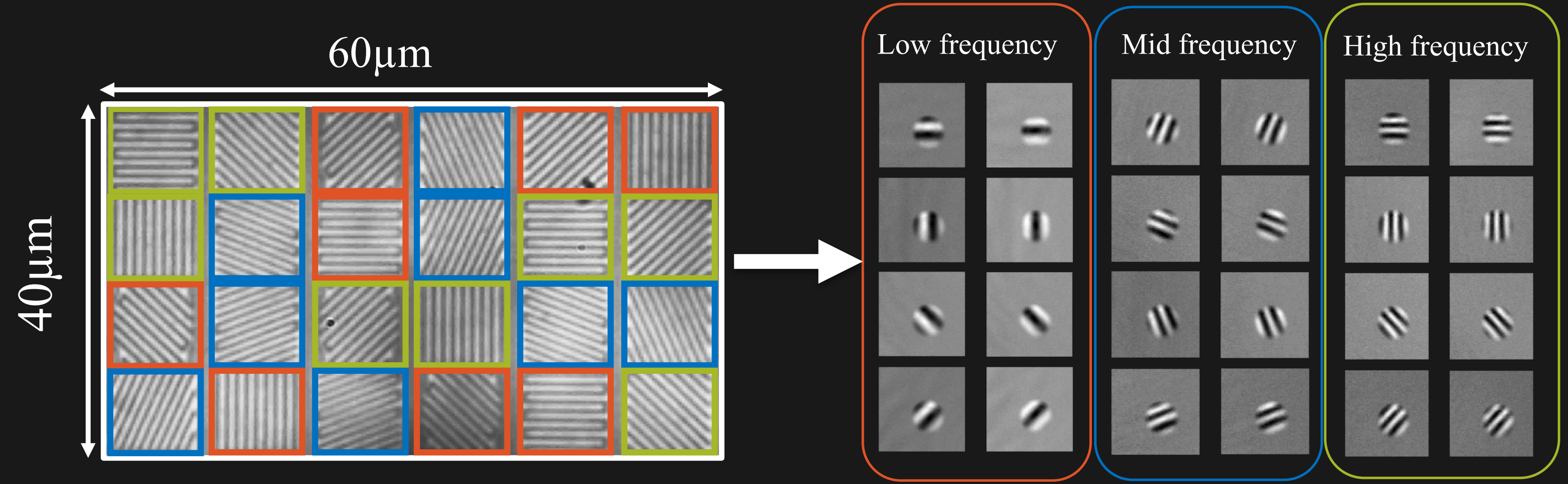
ASP camera has gabor like filters that show up as kernels in many CNNs, eg. AlexNet

Angle Sensitive Pixels (ASPs)

- ASPs are CMOS image sensors that have two diffraction gratings over photodiodes

- Different oriented/spaced gratings produce different oriented/frequency gabor-like point-spread-functions.





[Wang and Molnar, JSSC 2012]
[M. Hirsch et al., ICCP 2014]

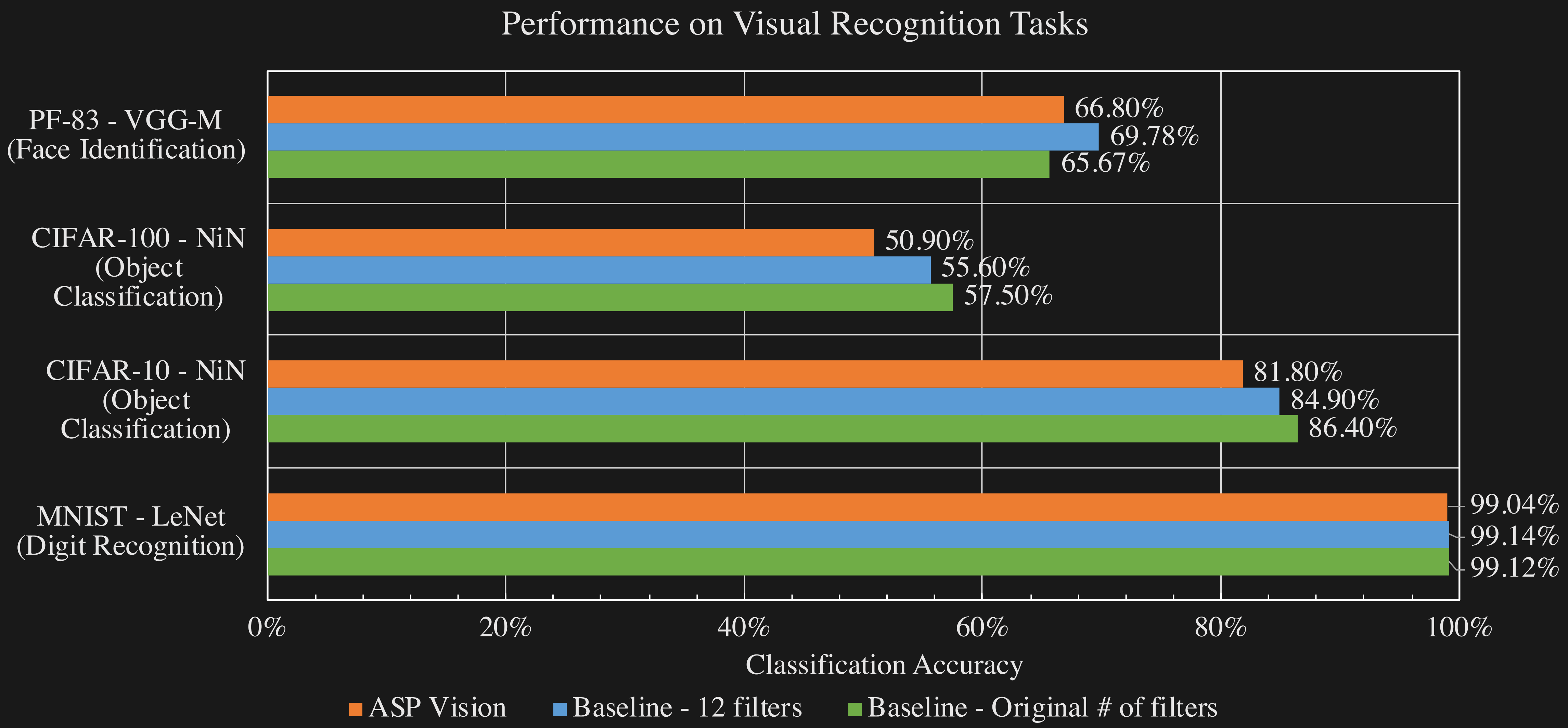
ASP sensors save energy with edge-only digitization

- 90% savings in image sensing
- 90% savings in bandwidth

[A. Wang, S. Sivaramakrishnan, and A. Molnar, CICC 2012]

	Sony (ISSCC 2015)	**ASP Image Sensor**
Resolution	5256 x 3934 (20M)	384 x 384 (effective ASP tile resolution: 96 x 64)
Energy consumption	Total power: 428 mW No breakdown of power reported	Total Power: 1.8 mW Pixel Array: 300 μW Amplifiers: 900 μW Timing/Addressing: 500 μW ADCs: 100 μW
	340 pJ/frame/pixel	**33 pJ/frame/pixel**
	90.2% Energy saving	
Transmission bandwidth	Transmitting the entire image	Transmitting only edges
	1.2 Mbits/frame @ 384 ×384×8bits	120 Kbits/frame @ 384×384×8bits
	10:1 Compression ratio	
Capabilities	2D image and video capture	2D images and video, edge filtered images, light field information

Results - ASP Vision performs comparably to CNNs on datasets



How many FLOPs can we save by skipping the first layer?

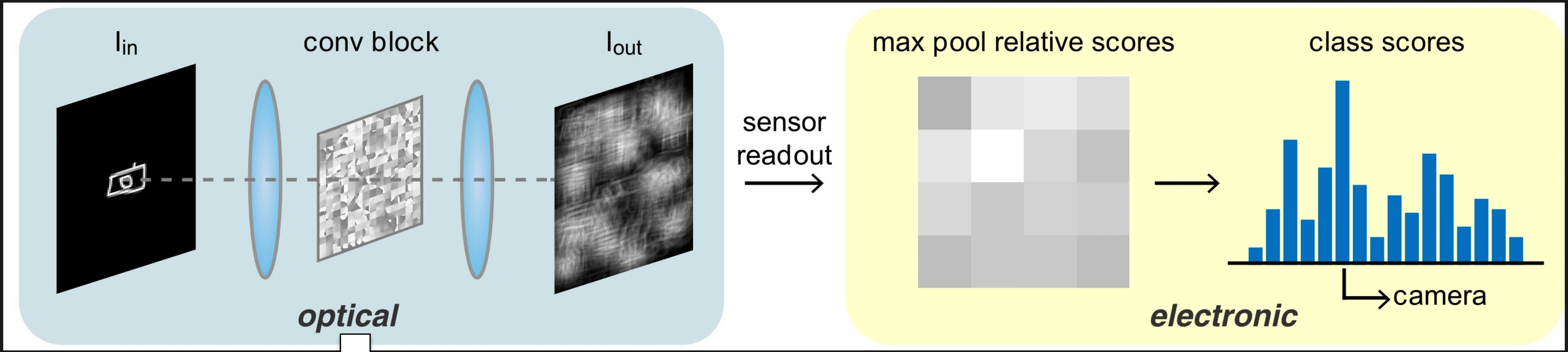
	VGG-M		NiN		LeNet	
# of Conv. Layers	8		9		4	
Input Image Size	$224 \times 224 \times 3$		$32 \times 32 \times 3$		$28 \times 28 \times 1$	
# of First Layer Filters	96 (Original)	12 (Prototype)	192 (Original)	12 (Prototype)	20 (Original)	12 (Prototype)
First Layer Conv. Kernel	$7 \times 7 \times 96$	$7 \times 7 \times 12$	$5 \times 5 \times 192$	$5 \times 5 \times 12$	$5 \times 5 \times 20$	$5 \times 5 \times 12$
FLOPS of Fist layer	708.0M	88.5 M	14.75M	921.6K	392 K	235 K
Total FLOPS	6.02G	3.83 G	200.3M	157 M	10.4 M	8.8 M
First Layer FLOPS Saving	11.76%	2.3%	7.36%	0.6%	3.77%	2.67%

Example 2:

Hybrid optical-electronic convolutional neural networks with optimized diffractive optics for image classification

Julie Chang, Vincent Sitzmann, Xiong Dun, Wolfgang Heidrich, and Gordon Wetzstein.
"Hybrid optical-electronic convolutional neural networks with optimized diffractive optics for image classification."
Scientific reports (2018).

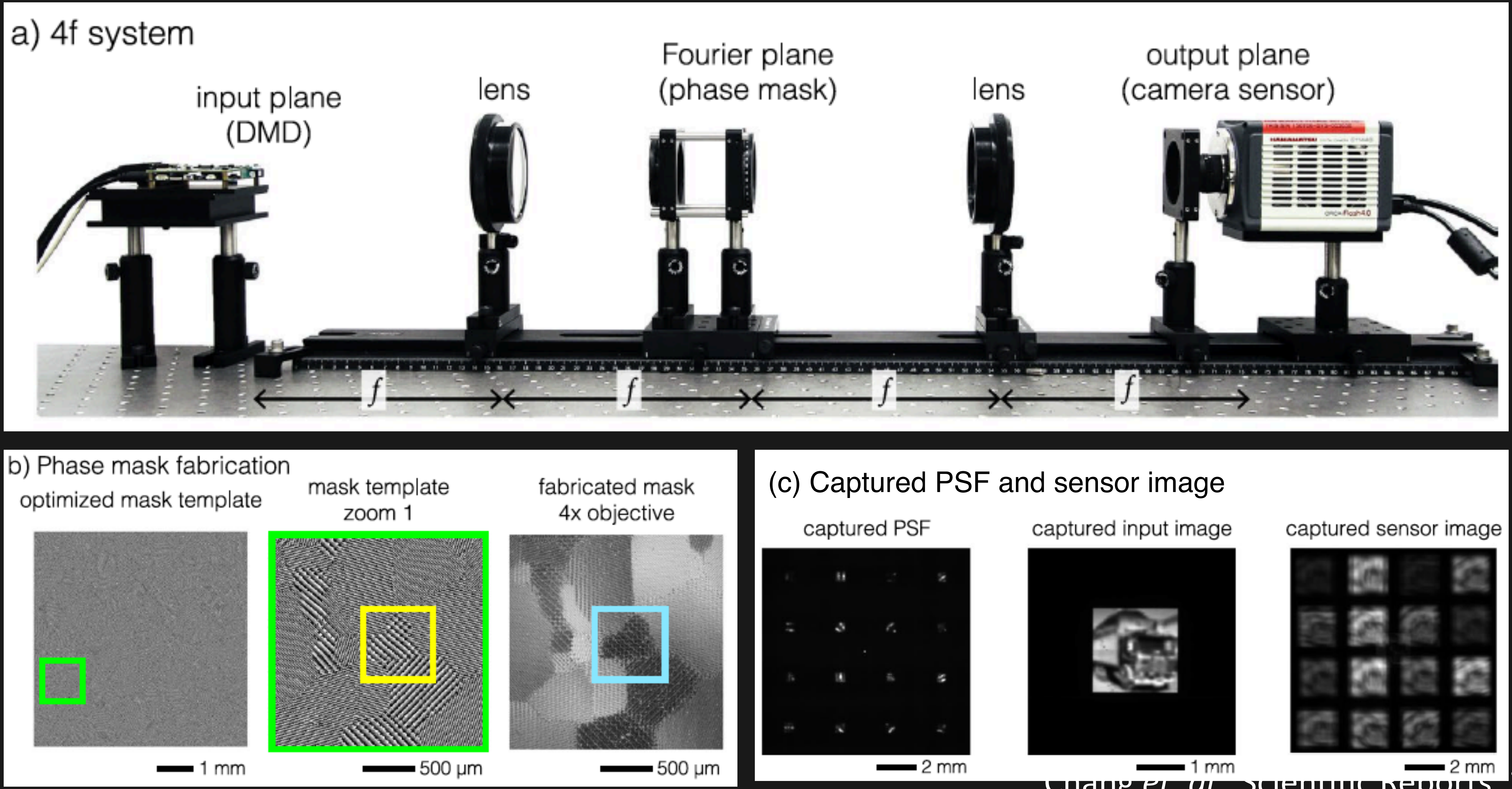
Hybrid optical-electronic CNN



Phase mask in Fourier plane to generate **learned** convolution filters

Chang *et. al.*, Scientific Reports, 2018

4F Optical system



Chang *et. al.*, Scientific Reports, 2018

Results – CIFAR 10 classification

Method	Test accuracy mean ± std. dev.	Learned parameters optical/electronic	FLOPs optical/electronic
Simulation:			
fully connected (FC) only	29.8 ± 0.5%	–/10,250	–/20,480
digital conv > ReLU > FC, unconstrained	51.9 ± 1.3%	–/82,586	–/1,490,954
digital conv > ReLU > FC, nonnegative	36.3 ± 0.5%	–/82,586	–/1,490,954
digital conv > ReLU > FC, pseudonegative	51.8 ± 0.6%	–/83,234	–/2,818,058
optical conv > ReLU > FC, pseudonegative	51.0 ± 1.4%	104,976/81,938	3,779,136/180,234
Physical experiment:			
optical conv > ReLU > FC, pseudonegative	44.4%	–/81,938	–/180,234

2x classification accuracy for same power

Only 12% of FLOPs for same accuracy

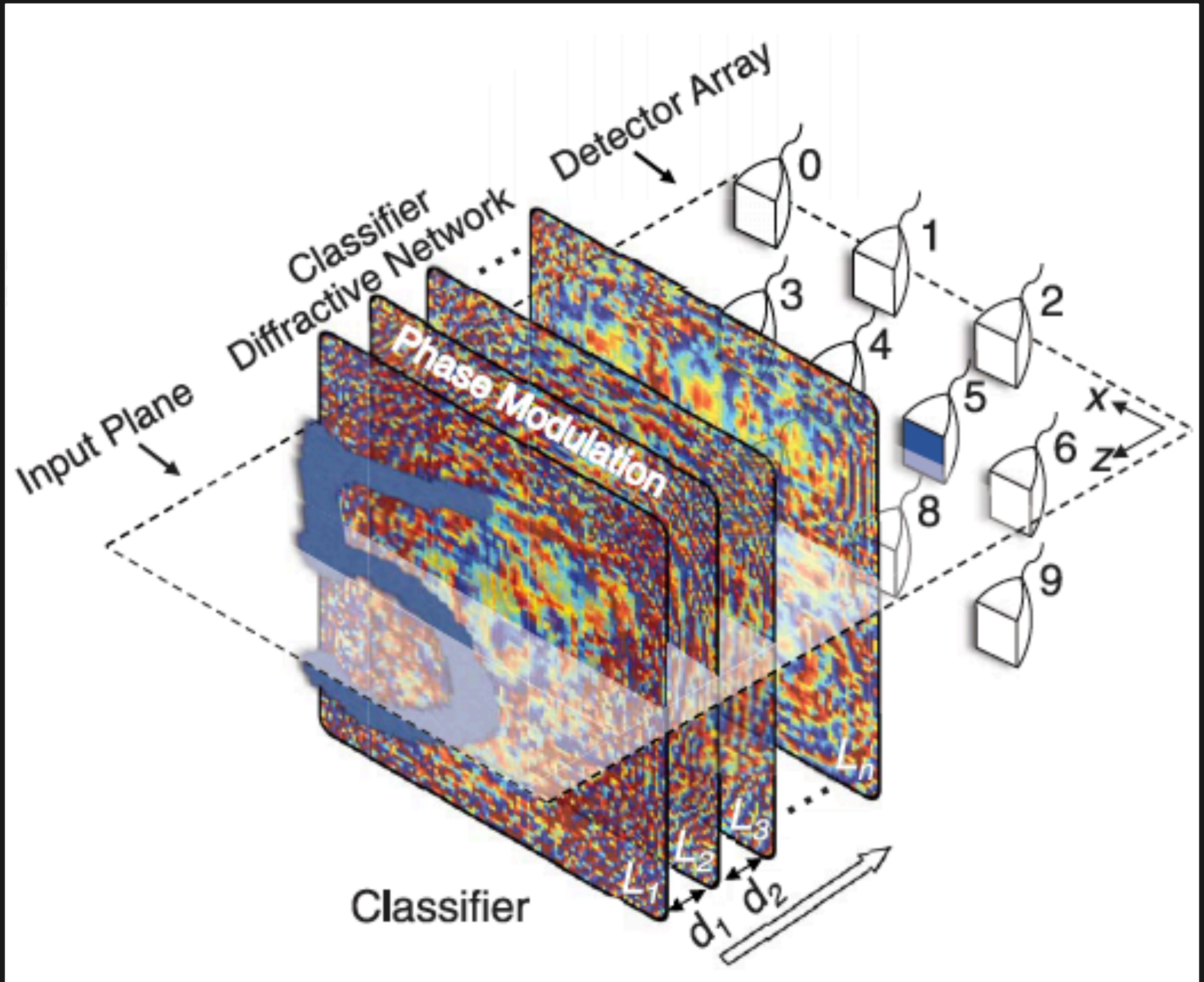
Chang *et. al.*, Scientific Reports, 2018

Example 3:

All-optical machine learning using diffractive deep neural networks

Xing Lin, Yair Rivenson, Nezih T. Yardimci, Muhammed Veli, Yi Luo, Mona Jarrahi, and Aydogan Ozcan.
"All-optical machine learning using diffractive deep neural networks."
Science (2018)

All Diffractive Optical System



Machine Learning System

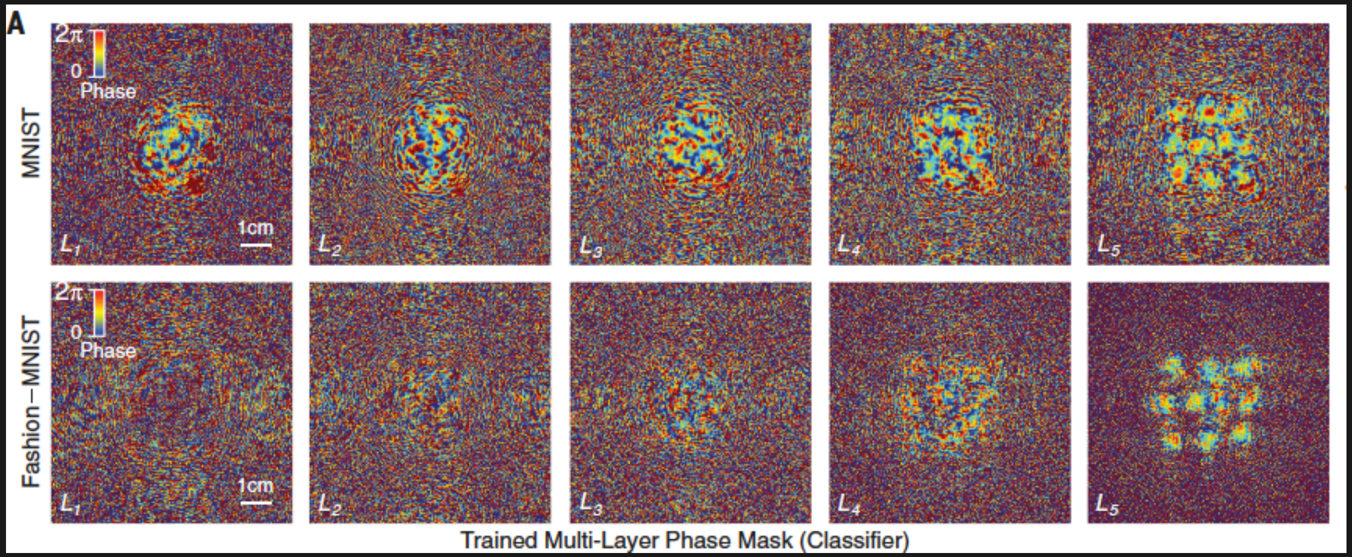
Diffractive Optical Layer

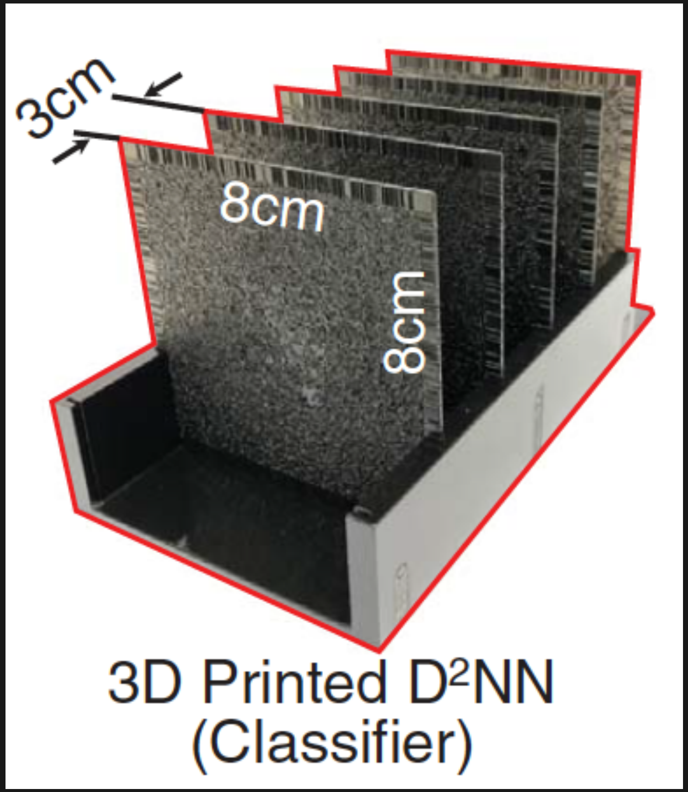
Diffractive layers interacting with input coherent field

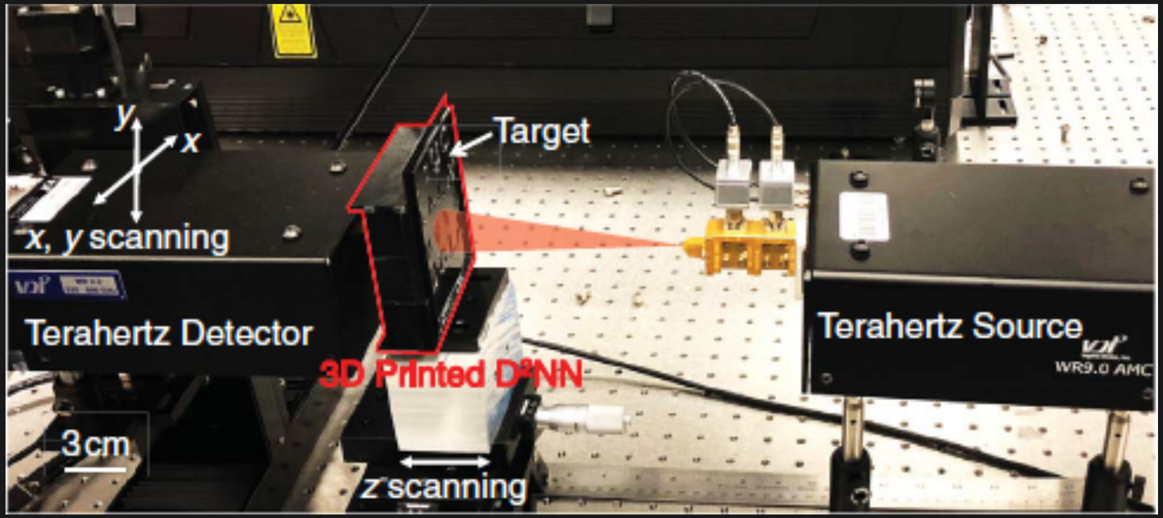
Lin *et. al.*, Science, 2018

Learned diffractive masks

Phase mask profiles



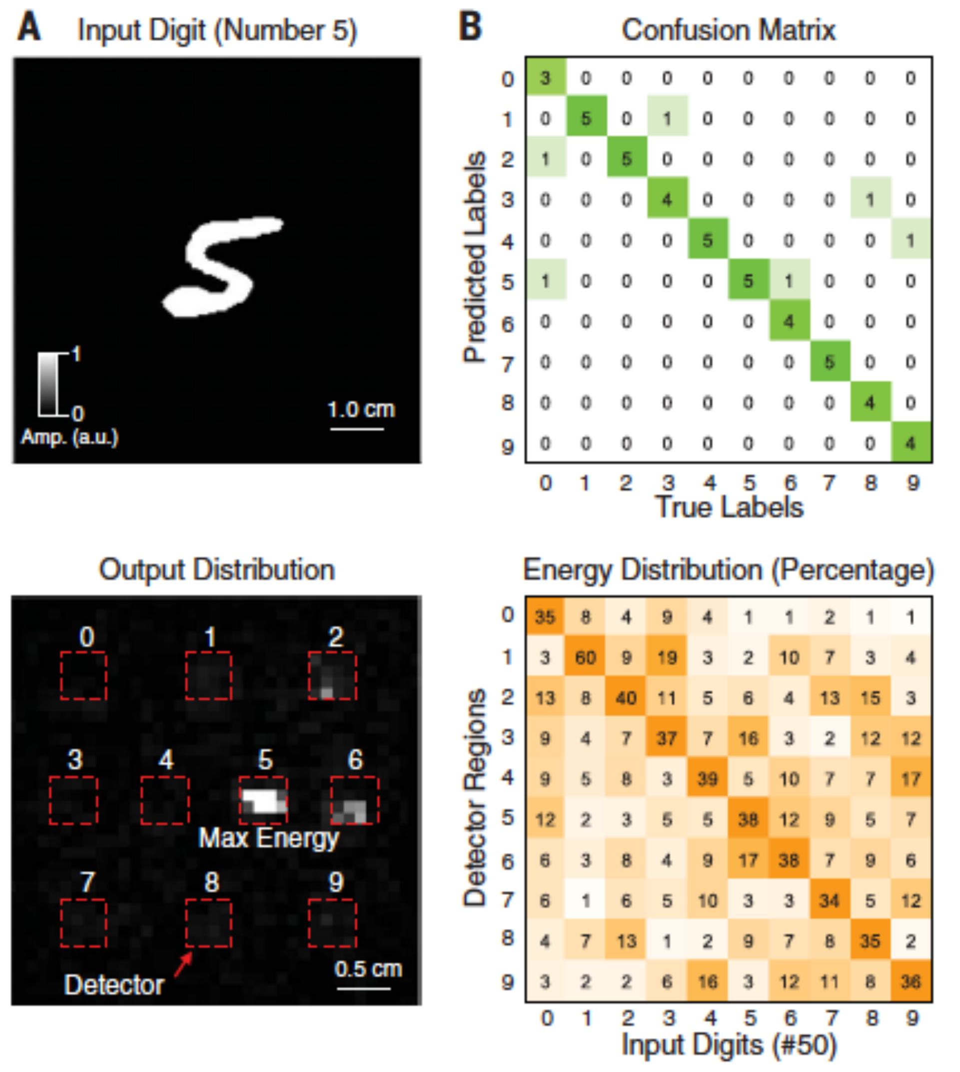




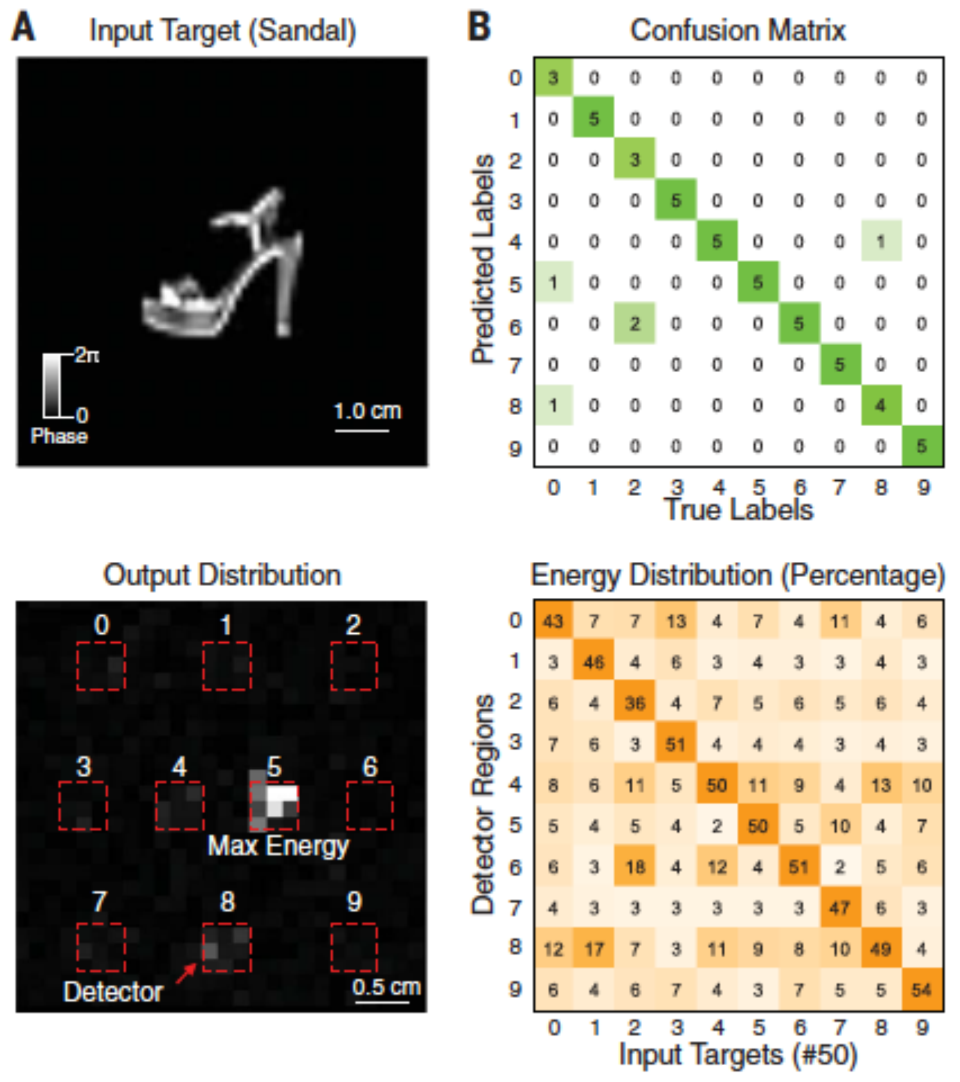
Lin *et. al.*, Science, 2018

Results – Classification accuracy

MNIST classification



Fashion-MNIST classification



Lin *et. al.*, Science, 2018

Part III: Summary

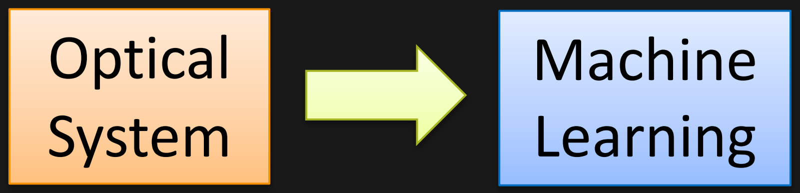
Machine Learning System

Optical Layer(s) → Electronic Layers(s) → Vision (Inference/ Classification)

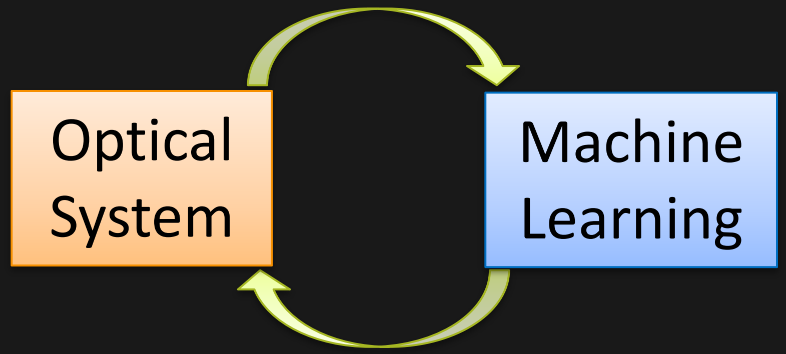
Incorporating optical layer(s) into machine learning system can decrease latency and power

Summary: *Visual Sensing using Machine Learning*

(I) Backend ML



(II) Joint design with ML



(III) ML with Optical System