WE ARE HIRING! WE ARE HIRING!
WE ARE HIRING! WE ARE HIRING! WE ARE HIRING!
WE ARE HIRING! WE ARE HIRING! WE ARE HIRING!

BIC @ CSBD

CSBDeep

CSBDeep - a deep learning toolbox for microscopists.

What can I do with my data?

Available tools

Questions

Contact | Impressum

CSBDeep - a deep learning toolbox for microscopy image restoration and analysis

Fluorescence microscopy is a key driver of discoveries in life-sciences, and the CSBDeep toolbox is offering a collection of state-of-the-art methods for content-aware image restoration and segmentation. On this website we showcase a number of real-world scenarios that might be useful in the context of your own work. Additionally, we will describe the respective methods and tools we provide and link to further resources wherever we can.

The entire CSBDeep toolbox is fully open source and intended to be used from either Python or Fiji.

http://csbdeep.bioimagecomputing.com

Microscopy Image Restoration and Downstream Analysis

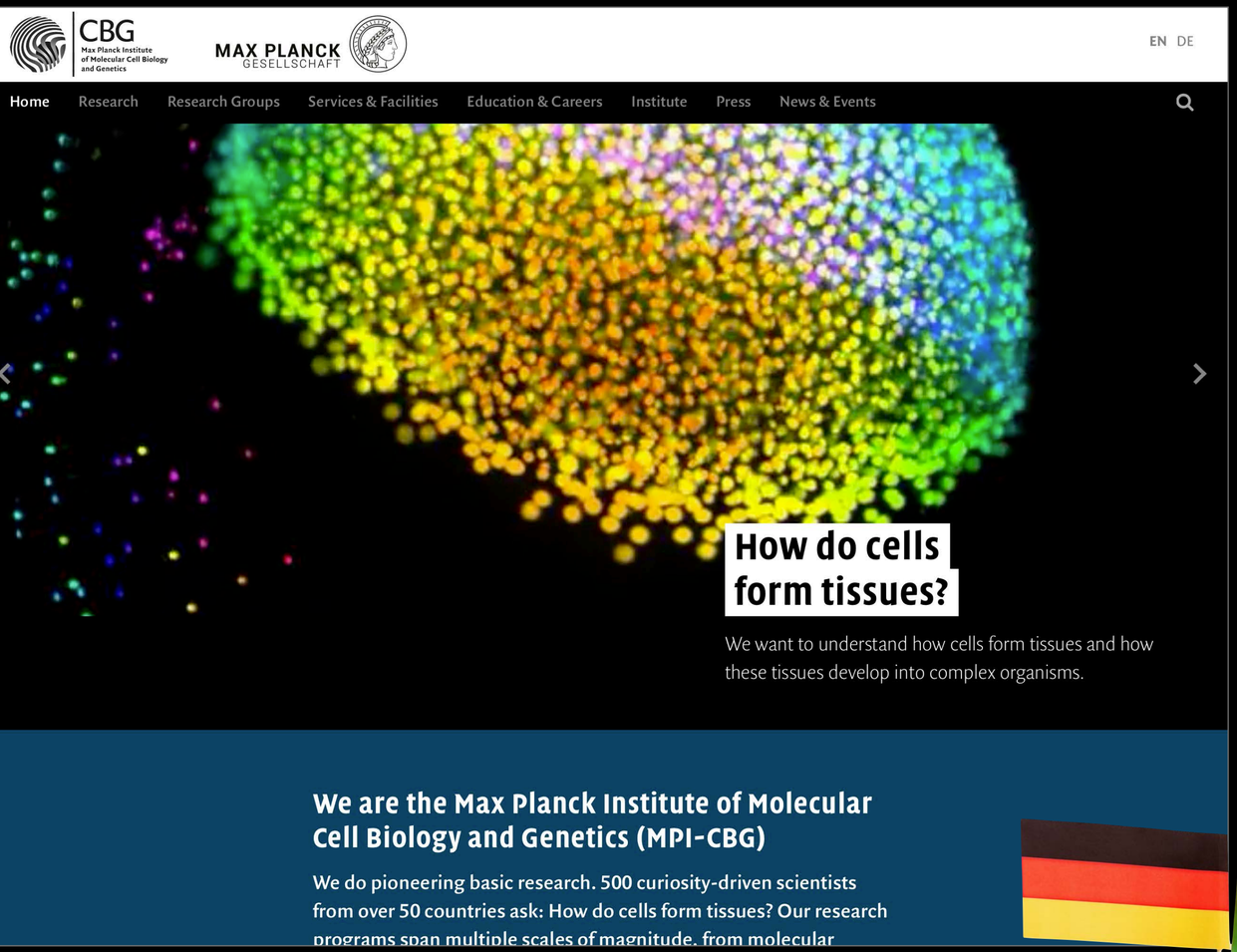
recent improvements and (my) hopes for the future

0. THE JUG LAB IS MOVING!

1. CONTENT-AWARE RESTORATION
2. DIVERSITY DENOISING
3. SHOULD YOU USE DL METHODS?
4. WHAT WILL THE FUTURE BRING?

MAX PLANCK GESELLSCHAFT

HUMAN TECHNOPOLE



HUMAN TECHNOPOLE

RESEARCH GROUP		IMAGE ANALYSIS FACILITY
METHODS RESEARCH (ML, OPTIMIZATION, ...)	RESEARCH SOFTWARE ENGINEERING	BIOIMAGE ANALYSIS (SERVICE + TEACHING)

INNOVATION "the brain"

- AI and CV research
- publication of new methods/ ideas

IMPLEMENTATION "the glue"

- making useful and usable tools out of the latest methods out there...

IMPACT/UTILITY "the muscle"

- analysis as a service
- building workflows that solve existing problem
- teaching

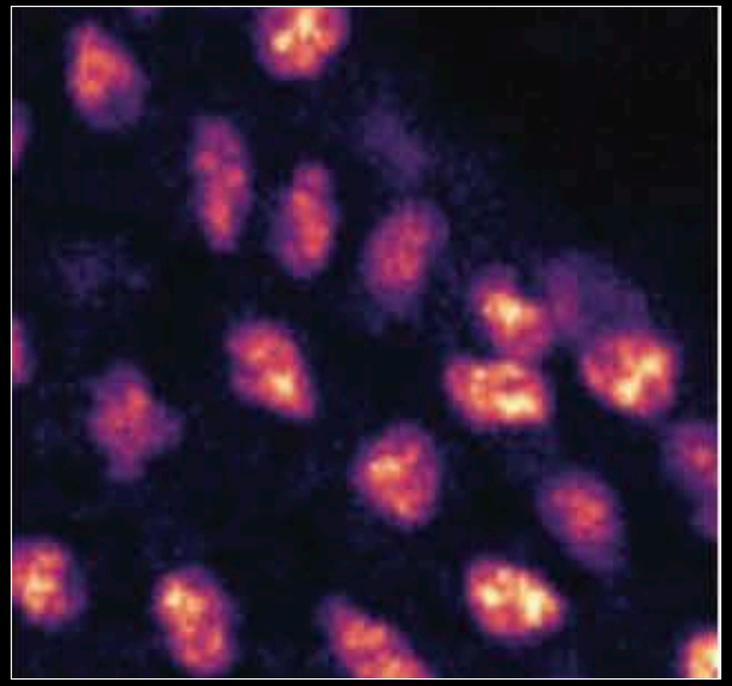
1. CONTENT-AWARE RESTORATION
2. DIVERSITY DENOISING
3. SHOULD YOU USE DL METHODS?
4. WHAT WILL THE FUTURE BRING?

CARE TRAINING IN A NUTSHELL

>> **Weigert et al. 2018**
>> data, **Rink Lab**, MPI-CBG

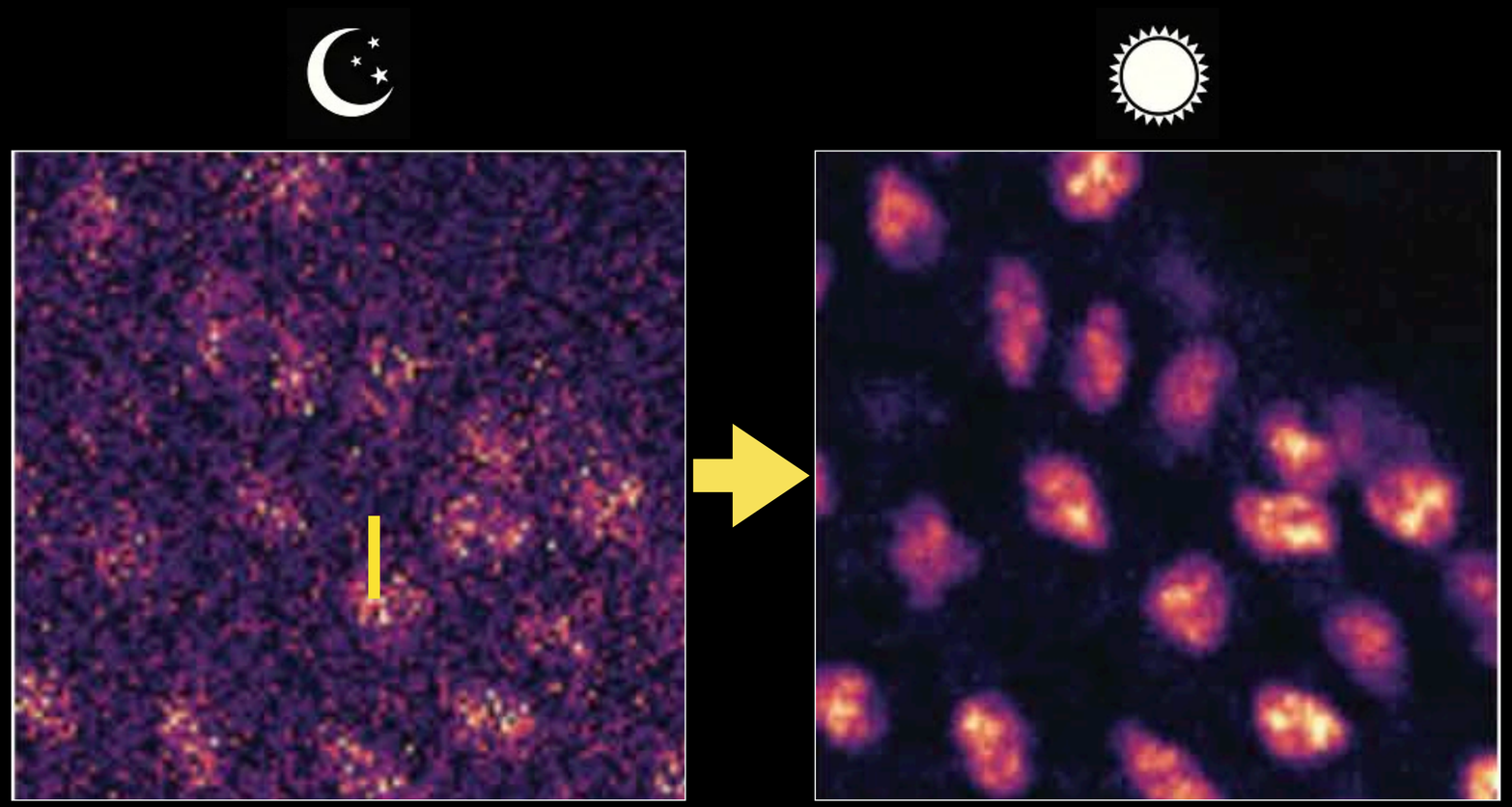
Martin Weigert

Uwe Schmidt



CARE TRAINING IN A NUTSHELL

>> **Weigert et al. 2018**
>> data, **Rink Lab**, MPI-CBG

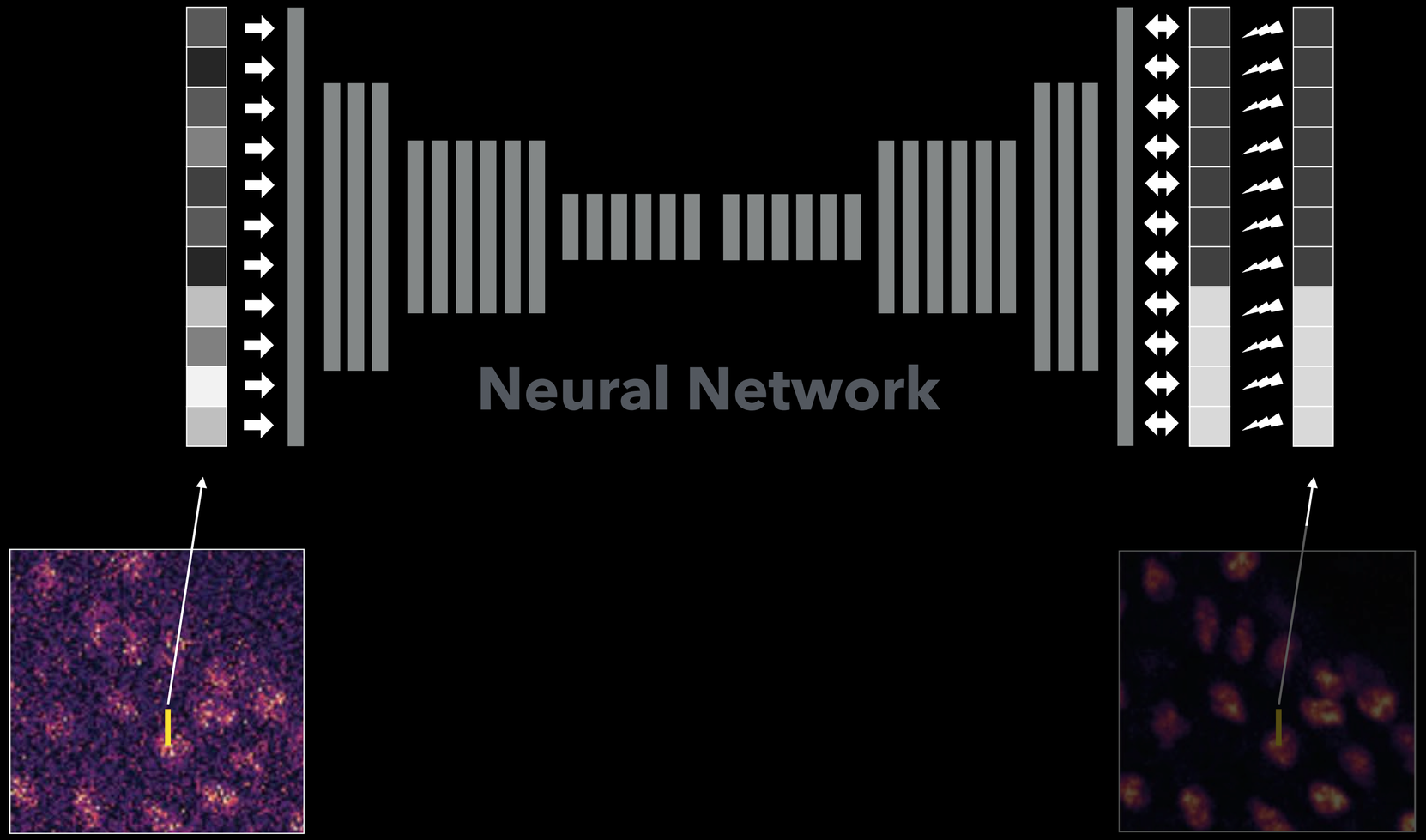


Martin Weigert

Uwe Schmidt

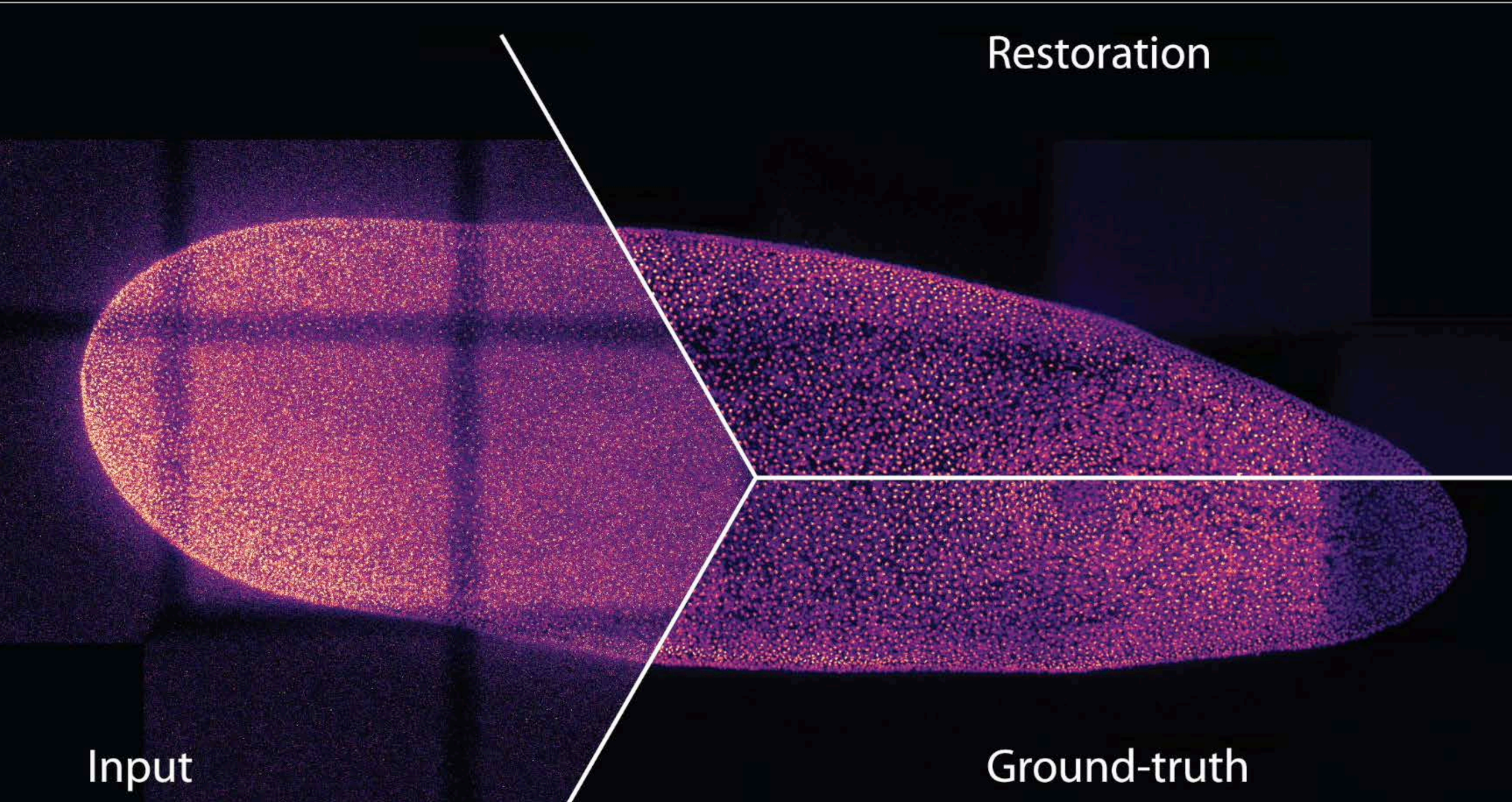
TRADITIONAL CARE TRAINING IN A NUTSHELL

>> **Weigert et al. 2018**
>> data, **Rink Lab**, MPI-CBG



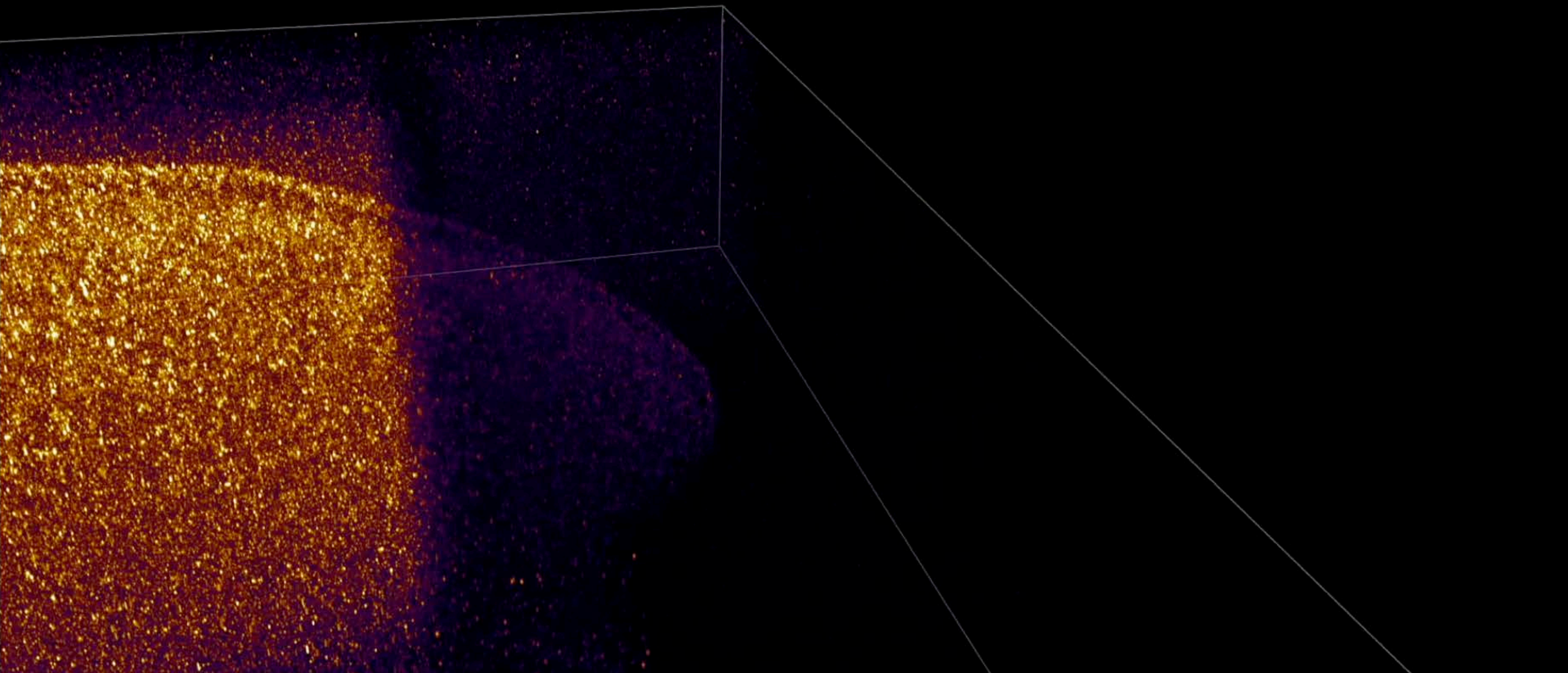
Martin Weigert

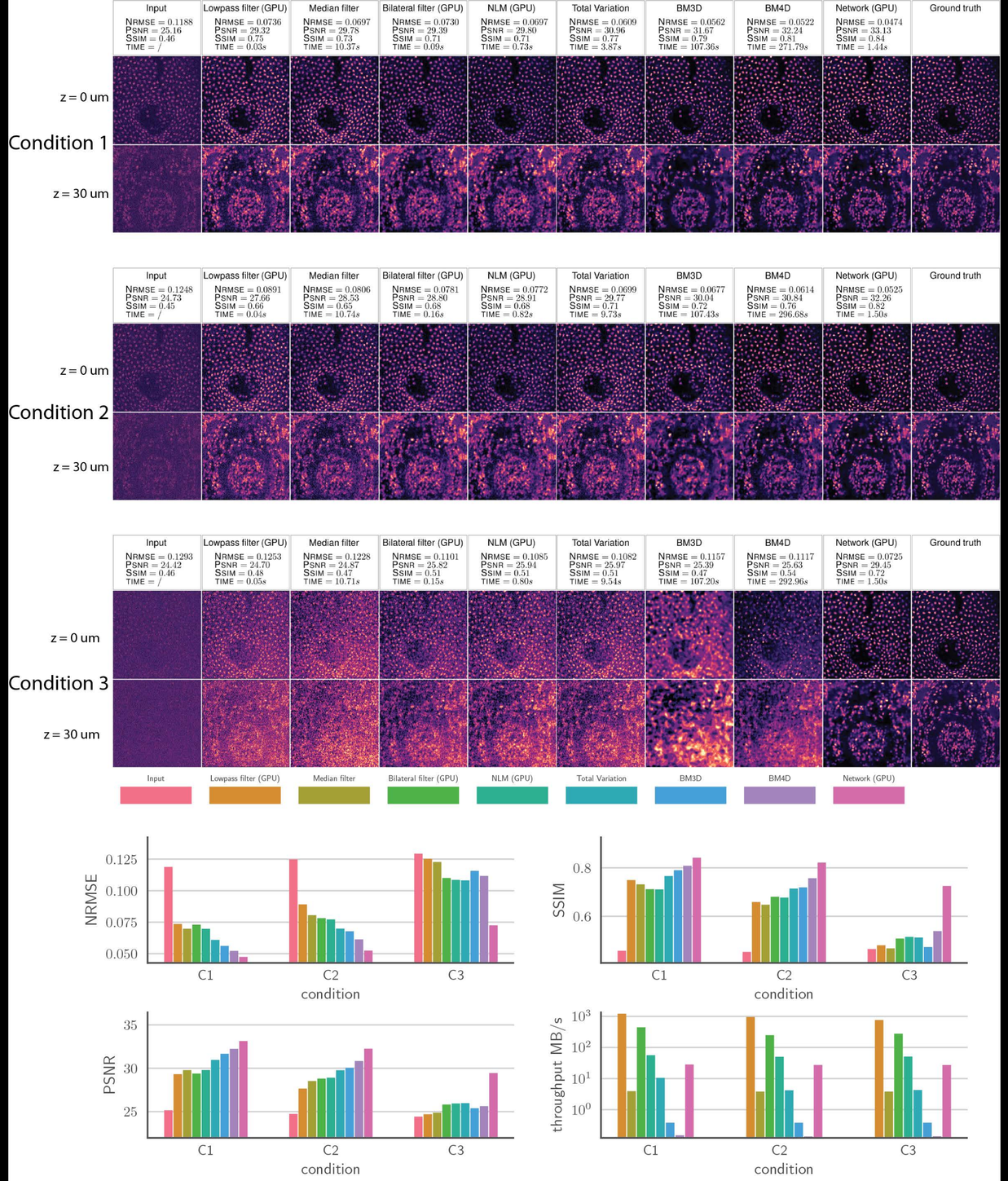
Uwe Schmidt

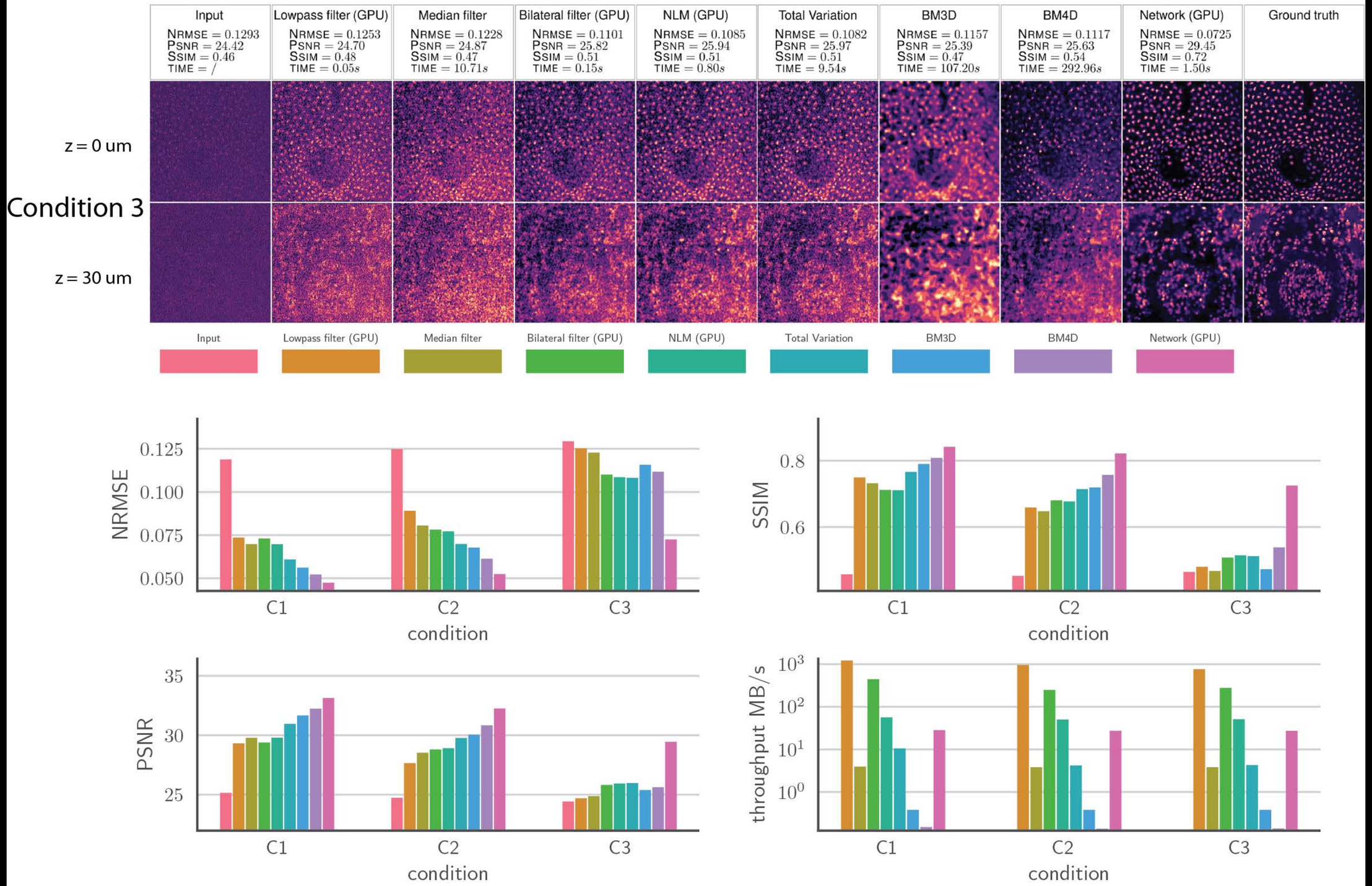

Restoration
Input
Ground-truth

CARE – TRAINED ON MICROSCOPY DATA

>>data, **Rink Lab**, MPI-CBG

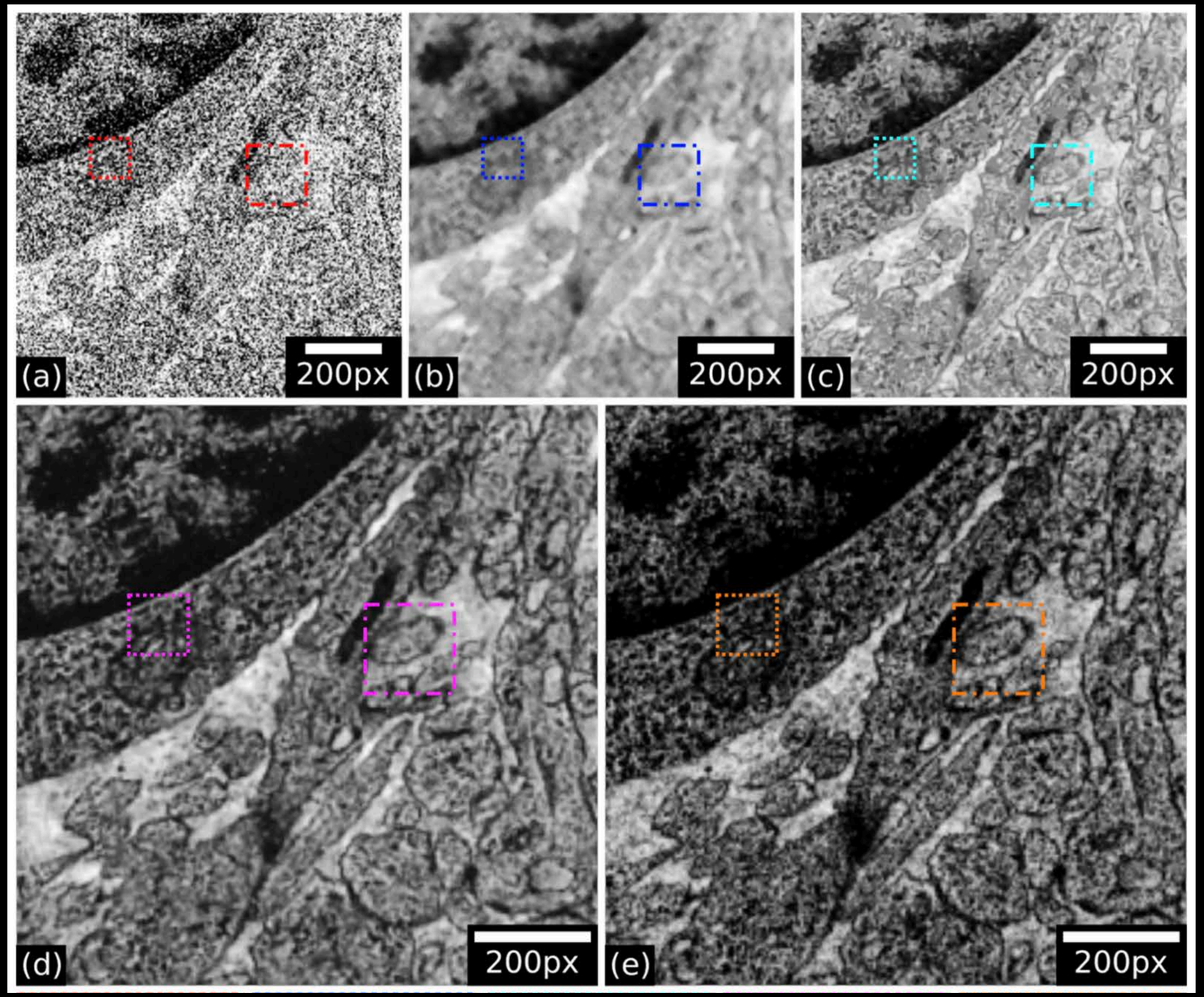



Input
NRMSE = 0.1188
PSNR = 25.16
SSIM = 0.46
TIME = /
Lowpass filter (GPU)
NRMSE = 0.0736
PSNR = 29.32
SSIM = 0.75
TIME = 0.03s
Median filter
NRMSE = 0.0697
PSNR = 29.78
SSIM = 0.73
TIME = 10.37s
Bilateral filter (GPU)
NRMSE = 0.0730
PSNR = 29.39
SSIM = 0.71
TIME = 0.09s
NLM (GPU)
NRMSE = 0.0697
PSNR = 29.80
SSIM = 0.71
TIME = 0.73s
Total Variation
NRMSE = 0.0609
PSNR = 30.96
SSIM = 0.77
TIME = 3.87s
BM3D
NRMSE = 0.0562
PSNR = 31.67
SSIM = 0.79
TIME = 107.36s
BM4D
NRMSE = 0.0522
PSNR = 32.24
SSIM = 0.81
TIME = 271.79s
Network (GPU)
NRMSE = 0.0474
PSNR = 33.13
SSIM = 0.84
TIME = 1.44s
Ground truth
z = 0 um
Condition 1
z = 30 um
Input
NRMSE = 0.1248
PSNR = 24.73
SSIM = 0.45
TIME = /
Lowpass filter (GPU)
NRMSE = 0.0891
PSNR = 27.66
SSIM = 0.66
TIME = 0.04s
Median filter
NRMSE = 0.0806
PSNR = 28.53
SSIM = 0.65
TIME = 10.74s
Bilateral filter (GPU)
NRMSE = 0.0781
PSNR = 28.80
SSIM = 0.68
TIME = 0.16s
NLM (GPU)
NRMSE = 0.0772
PSNR = 28.91
SSIM = 0.68
TIME = 0.82s
Total Variation
NRMSE = 0.0699
PSNR = 29.77
SSIM = 0.71
TIME = 9.73s
BM3D
NRMSE = 0.0677
PSNR = 30.04
SSIM = 0.72
TIME = 107.43s
BM4D
NRMSE = 0.0614
PSNR = 30.84
SSIM = 0.76
TIME = 296.68s
Network (GPU)
NRMSE = 0.0525
PSNR = 32.26
SSIM = 0.82
TIME = 1.50s
Ground truth
z = 0 um
Condition 2
z = 30 um
Input
NRMSE = 0.1293
PSNR = 24.42
SSIM = 0.46
TIME = /
Lowpass filter (GPU)
NRMSE = 0.1253
PSNR = 24.70
SSIM = 0.48
TIME = 0.05s
Median filter
NRMSE = 0.1228
PSNR = 24.87
SSIM = 0.47
TIME = 10.71s
Bilateral filter (GPU)
NRMSE = 0.1101
PSNR = 25.82
SSIM = 0.51
TIME = 0.15s
NLM (GPU)
NRMSE = 0.1085
PSNR = 25.94
SSIM = 0.51
TIME = 0.80s
Total Variation
NRMSE = 0.1082
PSNR = 25.97
SSIM = 0.51
TIME = 9.54s
BM3D
NRMSE = 0.1157
PSNR = 25.39
SSIM = 0.47
TIME = 107.20s
BM4D
NRMSE = 0.1117
PSNR = 25.63
SSIM = 0.54
TIME = 292.96s
Network (GPU)
NRMSE = 0.0725
PSNR = 29.45
SSIM = 0.72
TIME = 1.50s
Ground truth
z = 0 um
Condition 3
z = 30 um
Input
Lowpass filter (GPU)
Median filter
Bilateral filter (GPU)
NLM (GPU)
Total Variation
BM3D
BM4D
Network (GPU)
NRMSE
0.125
0.100
0.075
0.050
C1
C2
C3
condition
SSIM
0.8
0.6
C1
C2
C3
condition
PSNR
35
30
25
C1
C2
C3
condition
throughput MB/s
10^3
10^2
10^1
10^0
C1
C2
C3
condition


Input
NRMSE = 0.1293
PSNR = 24.42
SSIM = 0.46
TIME = /
Lowpass filter (GPU)
NRMSE = 0.1253
PSNR = 24.70
SSIM = 0.48
TIME = 0.05s
Median filter
NRMSE = 0.1228
PSNR = 24.87
SSIM = 0.47
TIME = 10.71s
Bilateral filter (GPU)
NRMSE = 0.1101
PSNR = 25.82
SSIM = 0.51
TIME = 0.15s
NLM (GPU)
NRMSE = 0.1085
PSNR = 25.94
SSIM = 0.51
TIME = 0.80s
Total Variation
NRMSE = 0.1082
PSNR = 25.97
SSIM = 0.51
TIME = 9.54s
BM3D
NRMSE = 0.1157
PSNR = 25.39
SSIM = 0.47
TIME = 107.20s
BM4D
NRMSE = 0.1117
PSNR = 25.63
SSIM = 0.54
TIME = 292.96s
Network (GPU)
NRMSE = 0.0725
PSNR = 29.45
SSIM = 0.72
TIME = 1.50s
Ground truth
z = 0 um
Condition 3
z = 30 um
Input
Lowpass filter (GPU)
Median filter
Bilateral filter (GPU)
NLM (GPU)
Total Variation
BM3D
BM4D
Network (GPU)
NRMSE
0.125
0.100
0.075
0.050
C1
C2
C3
condition
SSIM
0.8
0.6
C1
C2
C3
condition
PSNR
35
30
25
C1
C2
C3
condition
throughput MB/s
10^3
10^2
10^1
10^0
C1
C2
C3
condition

CARE ON SCANNING ELECTRON MICROSCOPY DATA

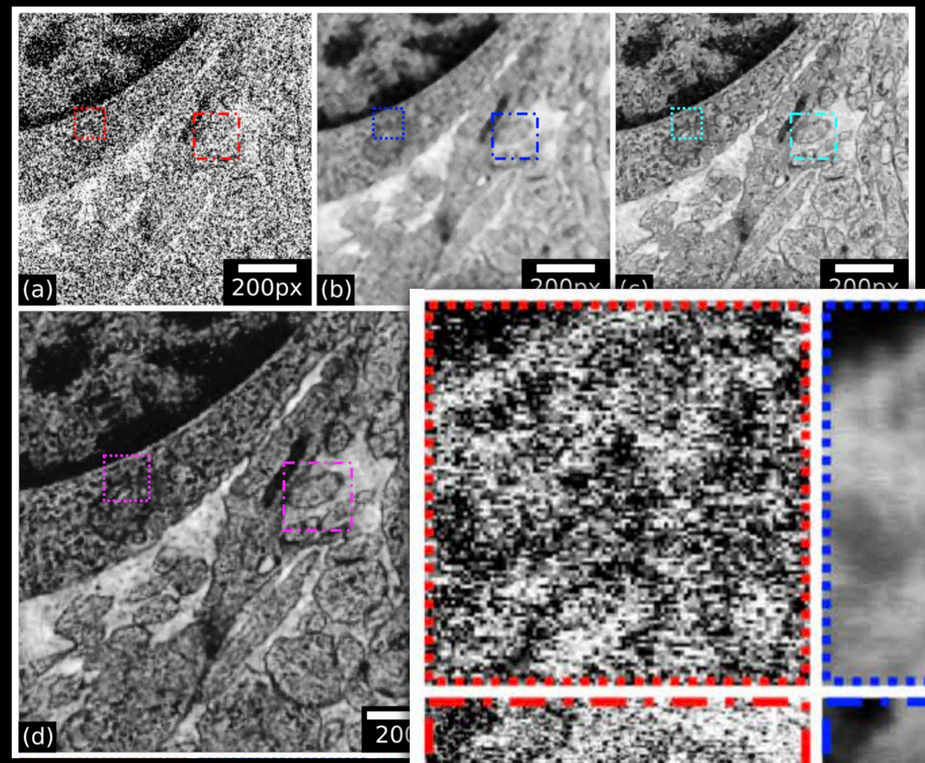
>> data, **Jékely Lab** (University of Exeter)



Tim-Oliver Buchholz

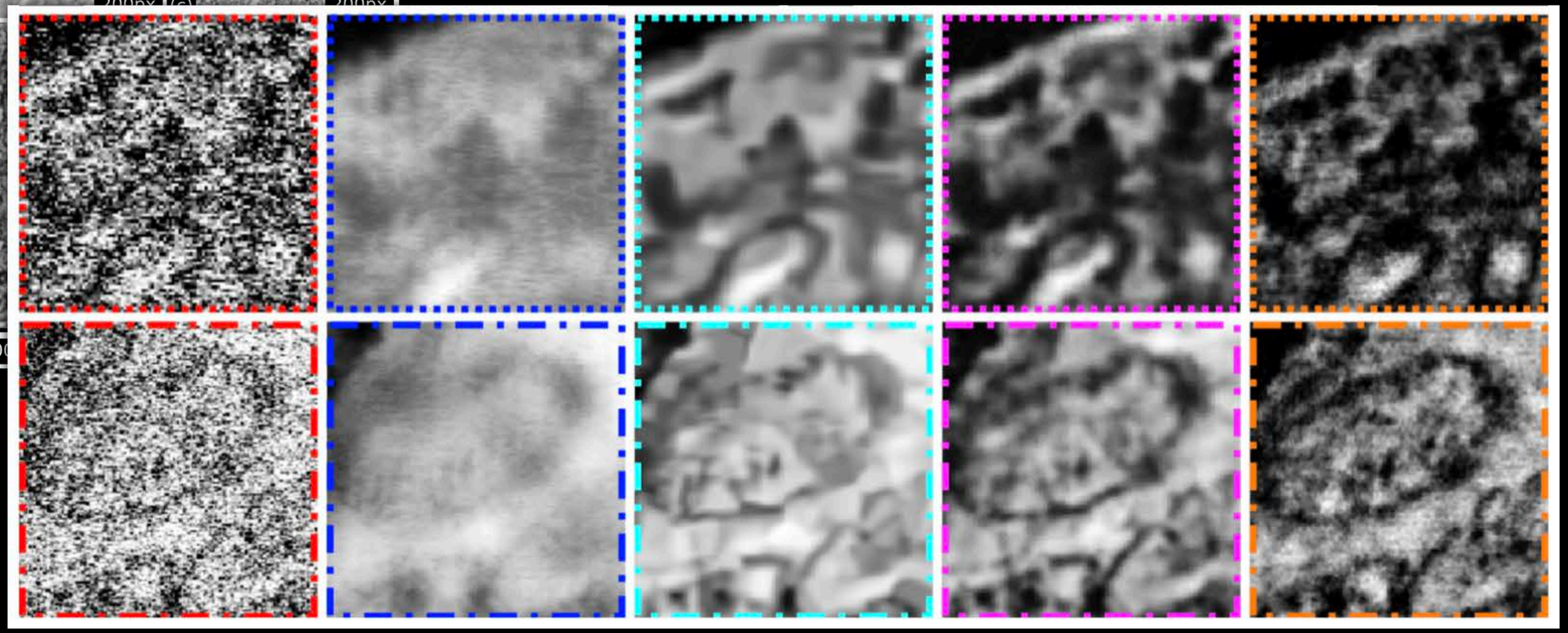
CARE ON SCANNING ELECTRON MICROSCOPY DATA

>> data, **Jékely Lab** (University of Exeter)



	PSNR	SSIM
Input (5 MHz)	6.62	0.09
NLM (Buades et al., 2005)	9.25	0.16
BM3D (Dabov et al., 2009)	9.41	0.37
SEM-CARE (*our*)	**16.56**	**0.47**

BASELINES



RAW (**0.2usec**; 5MHz) | NLM | BM3D | **SEM-CARE** | **GT** (**5usec**; 4*0.2MHz)

CARE on scanning electron microscopy data

CSBDeep

CSBDeep - a deep learning toolbox for microscopists.

What can I do with my data?

Available tools

Questions

Contact | Impressum

Image restoration

Image restoration with pairs of noisy and clean images:

- Deconvolution (Microtubules)
- Denoising in 2D (human U2OS data)
- Denoising in 3D (Planaria nuclei)
- Denoising in 3D (Tribolium nuclei)
- Isotropic Reconstruction (Drosophila nuclei)
- Isotropic Reconstruction (Zebrafish retina)
- Surface Projection (Drosophila wing, e-cadherin)

Search or jump to... / Pull requests Issues Marketplace Explore

HenriquesLab / ZeroCostDL4Mic

Watch 27 Star

Code | Issues 11 | Pull requests 10 | Actions | Projects | Wiki | Security | Insights

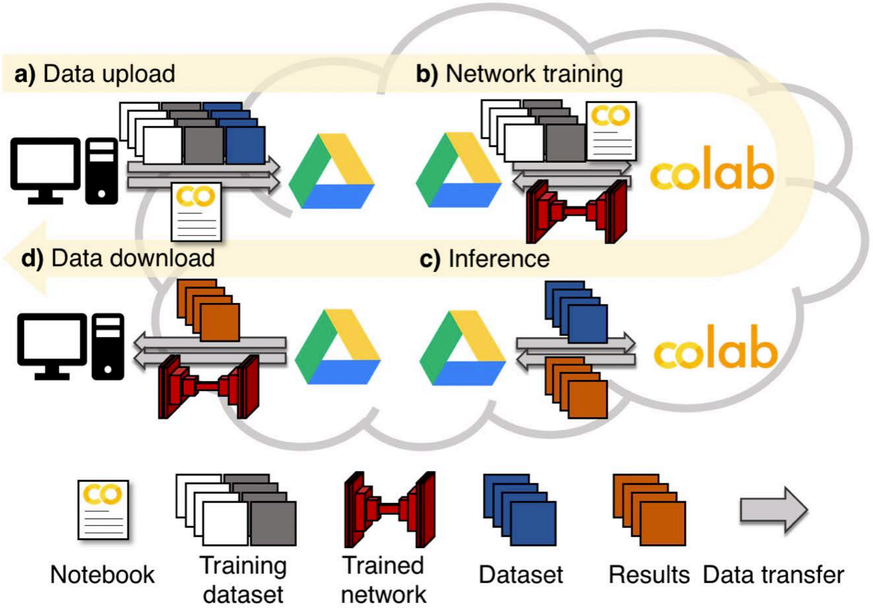
Home

Romain F. Laine edited this page 20 days ago · 158 revisions

ZeroCostDL4Mic - What is it?

Overview

ZeroCostDL4Mic is a toolbox for the training and implementation of common Deep Learning approaches to microscopy imaging. It exploits the ease-of-use and access to GPU provided by Google Colab.



Training data can be uploaded to the Google Drive from where it can be used to train models using the provided Colab notebooks in a web-browser. Inference (predictions) on unseen data can then also be performed within the same notebook, therefore not requiring any local hardware or software set-up.

Pages

Main:

- Home
- Step
- How
- Tips,
- Data
- Quali
- Runni
- Runni Floyd

Fully sup

- U-Ne
- StarD
- Noise
- CARE
- Label
- Objec
- pix2p
- Cycle
- Deep

Other res

- Gloss
- Usefu
- Exam

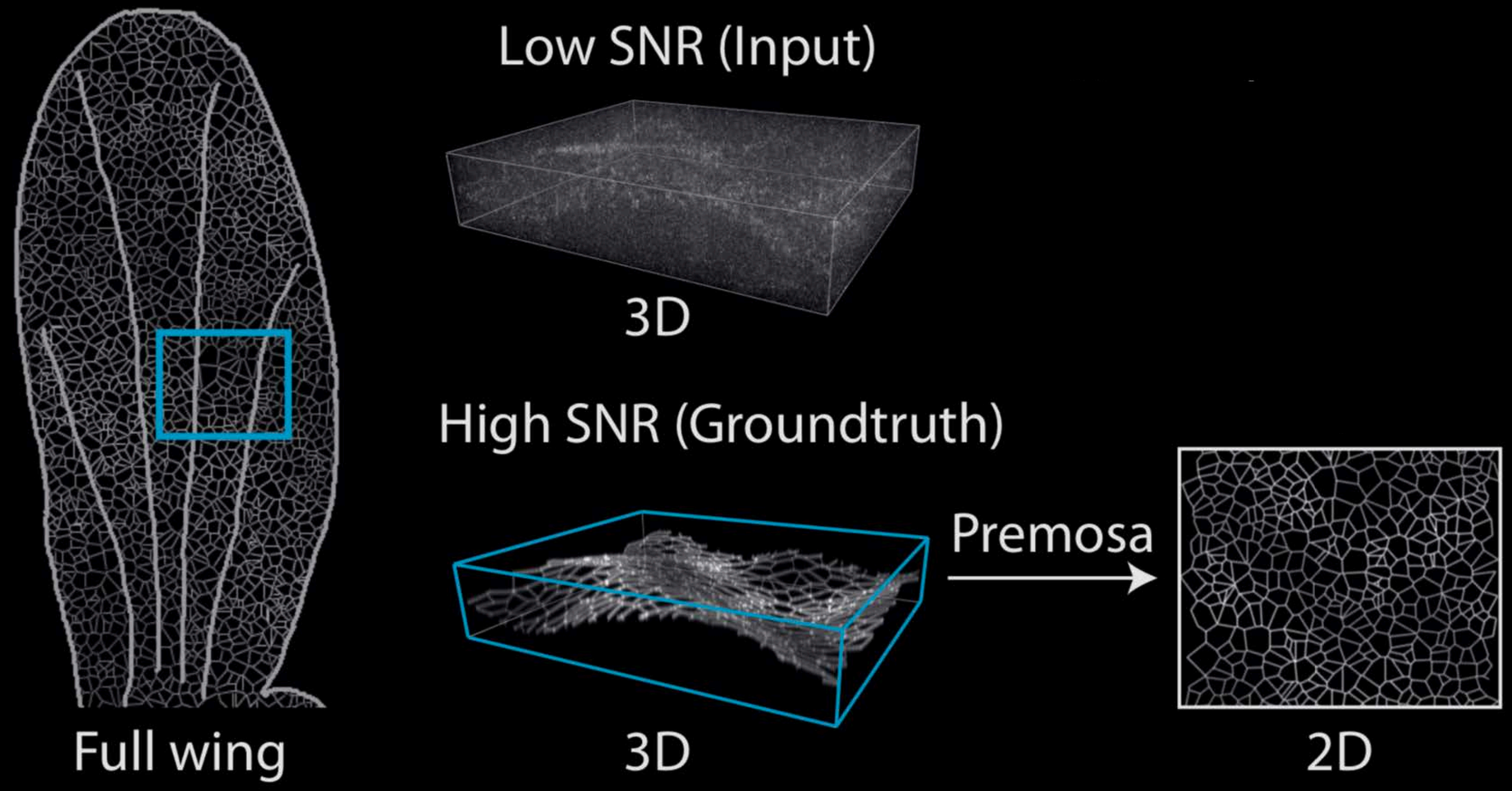
Clone this

https://g

LIMITATIONS OF CARE?

1. GT MUST BE OBTAINABLE!
2. TRAINING DATA MUST SAMPLE ALL VISUAL FEATURES OF INTEREST!

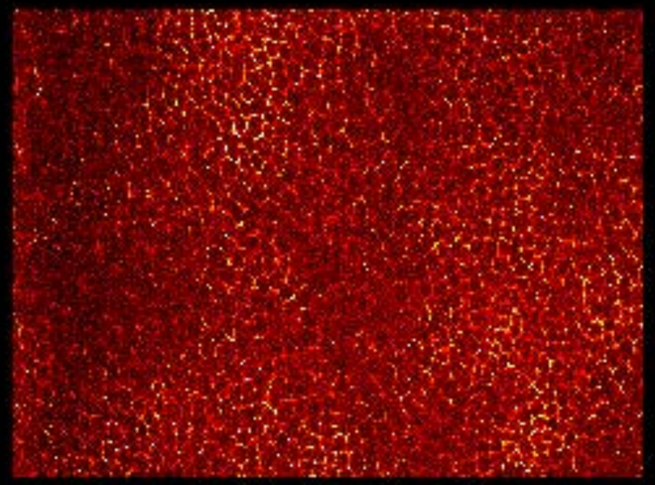
CARE – A (SLIGHTLY MORE) ADVANCED EXAMPLE



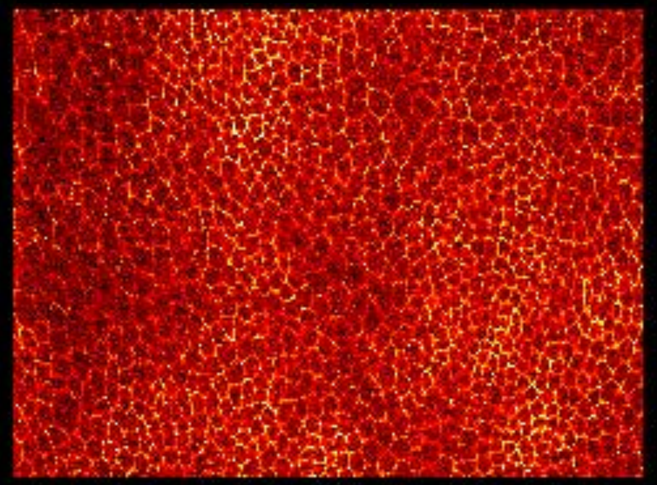
CARE – A (SLIGHTLY MORE) ADVANCED EXAMPLE

>> fly wing, **Eaton Lab**, MPI-CBG

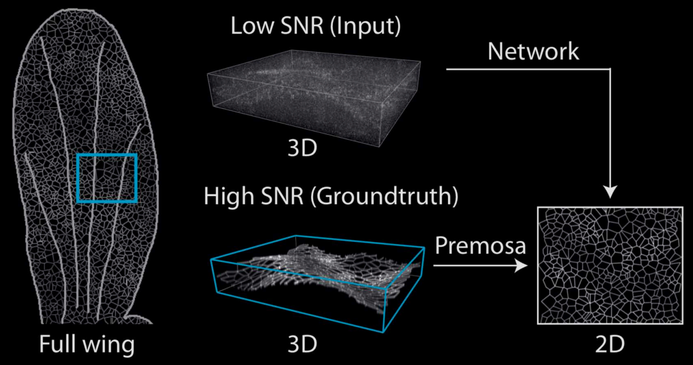
INPUT (MAX PROJ.)



INPUT (PREMOSA)



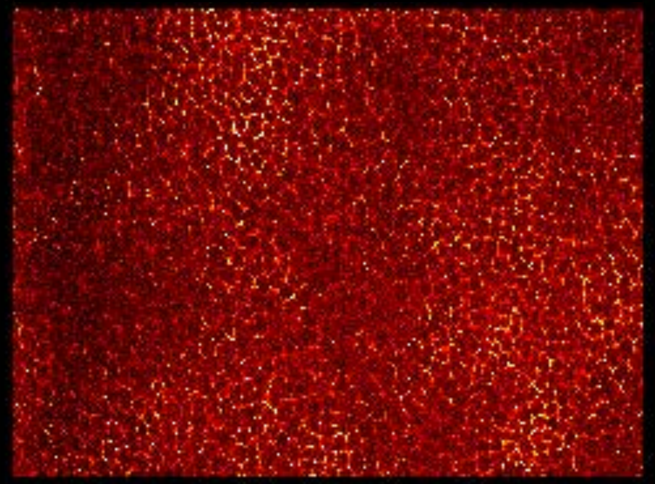
5% laser power
120ms exposure



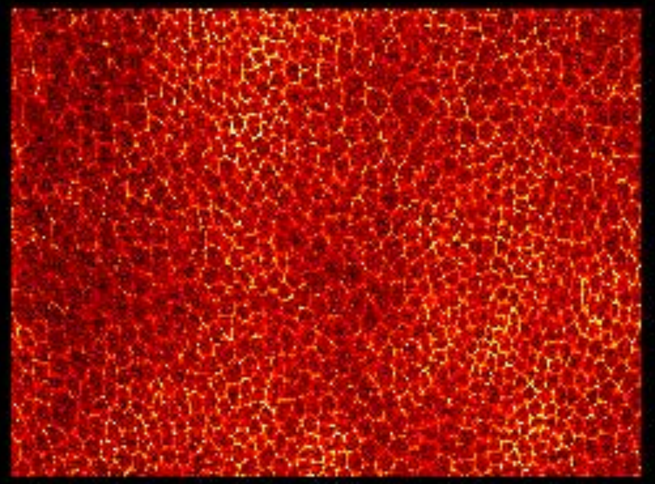
CARE – A (SLIGHTLY MORE) ADVANCED EXAMPLE

>> fly wing, **Eaton Lab**, MPI-CBG

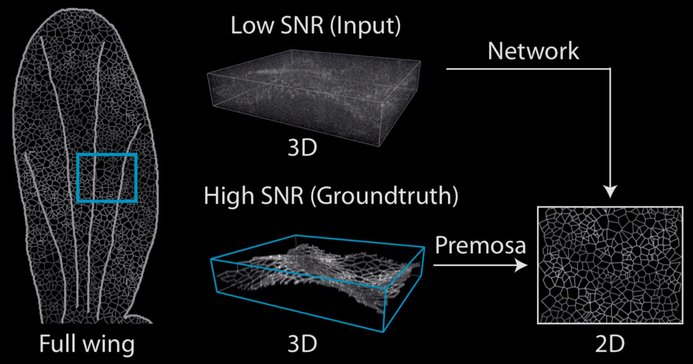
INPUT (MAX PROJ.)



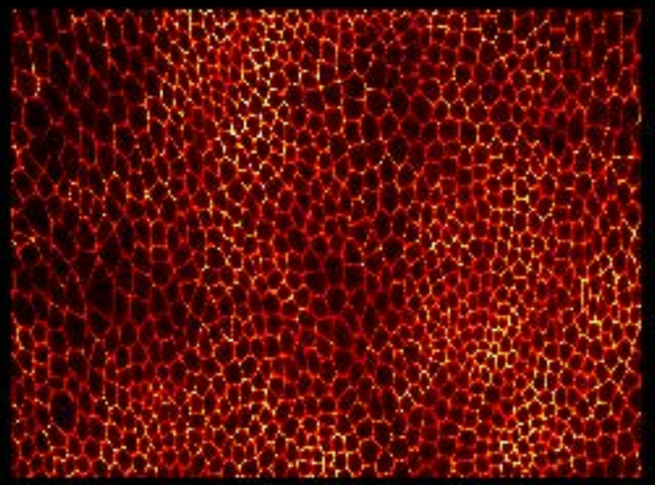
INPUT (PREMOSA)



5% laser power
120ms exposure



GT (PREMOSA)

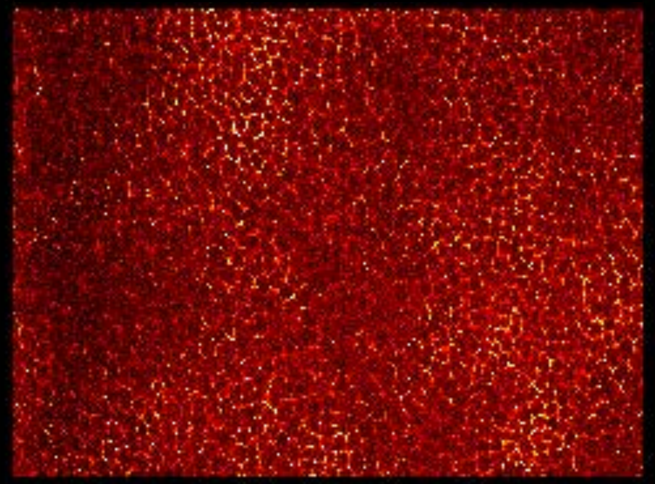


50% laser; 240ms exp.

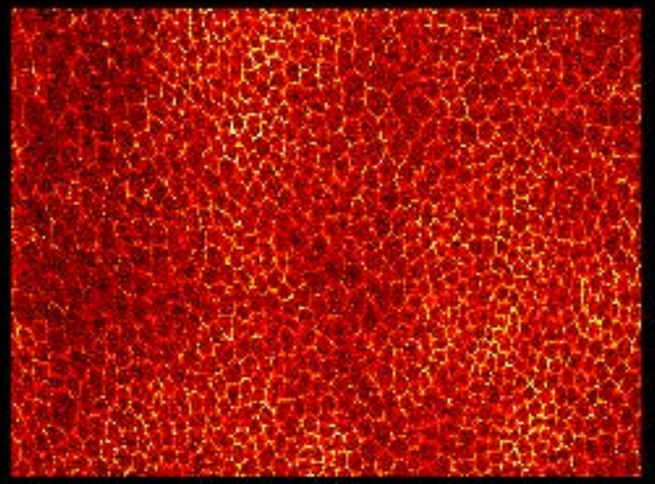
CARE – A (SLIGHTLY MORE) ADVANCED EXAMPLE

>> fly wing, **Eaton Lab**, MPI-CBG

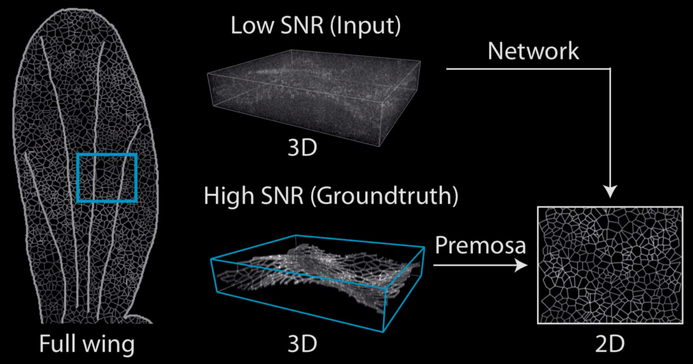
INPUT (MAX PROJ.)



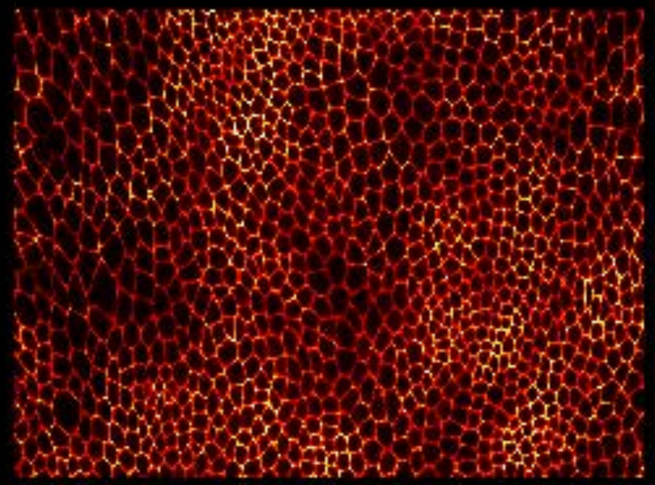
INPUT (PREMOSA)



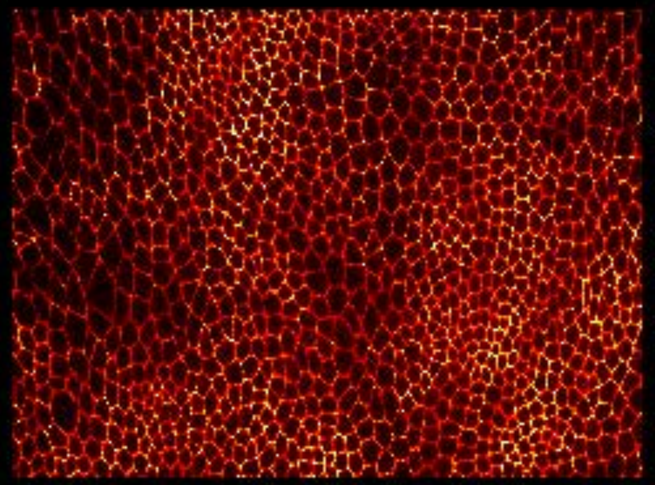
2% laser power
120ms exposure



CARE+

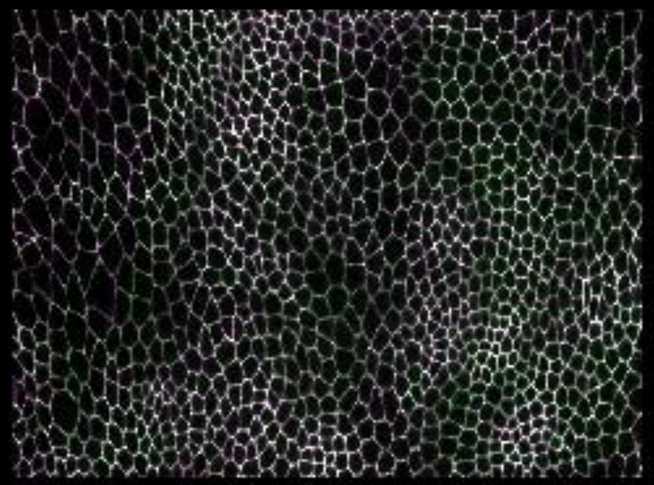


GT (PREMOSA)



50% laser; 240ms exp.

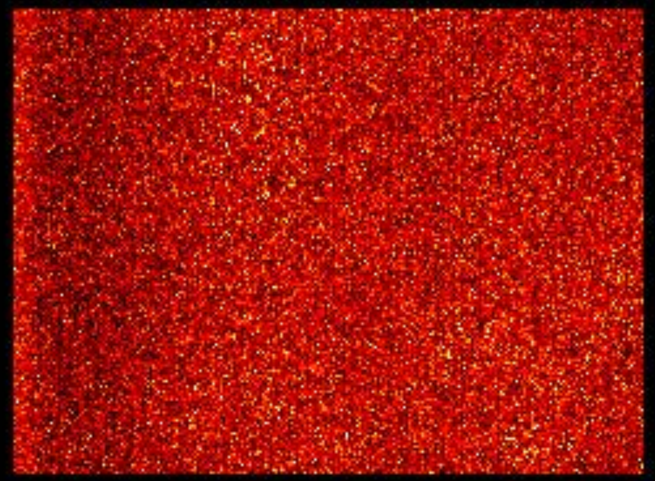
CARE vs. GT



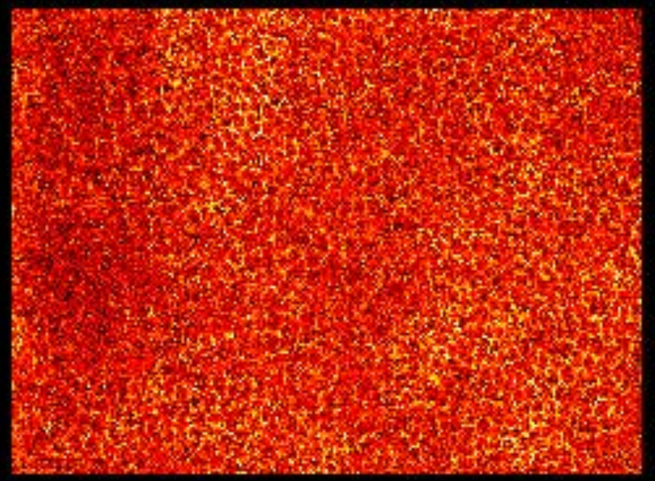
CARE – A (SLIGHTLY MORE) ADVANCED EXAMPLE

>> fly wing, **Eaton Lab**, MPI-CBG

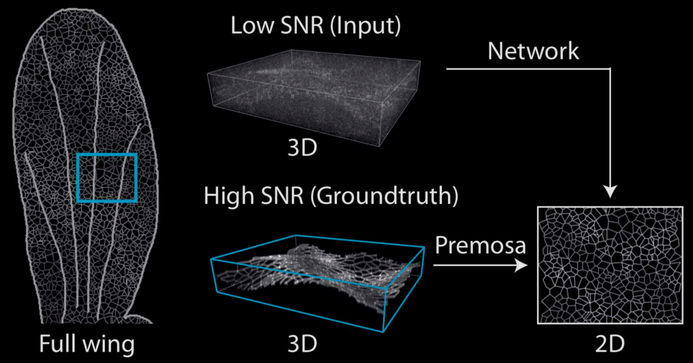
INPUT (MAX PROJ.)



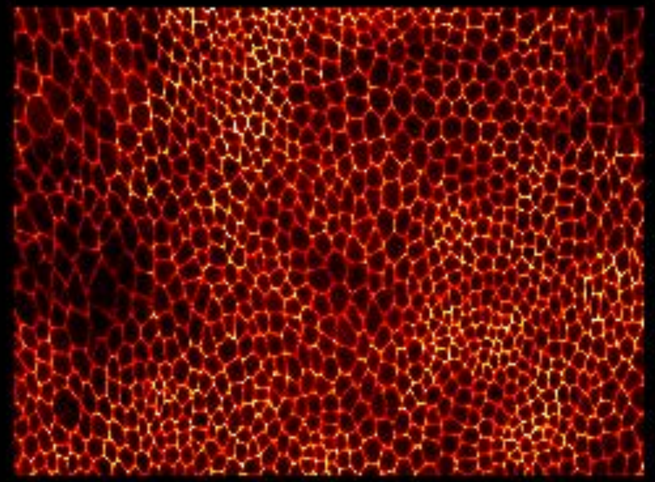
INPUT (PREMOSA)



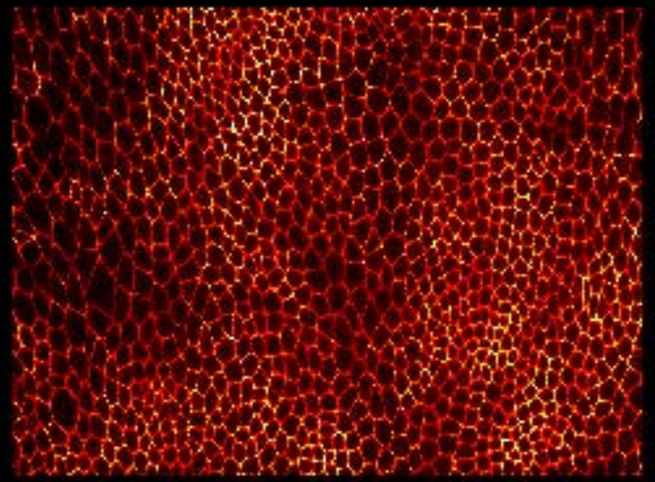
2% laser power
120ms exposure



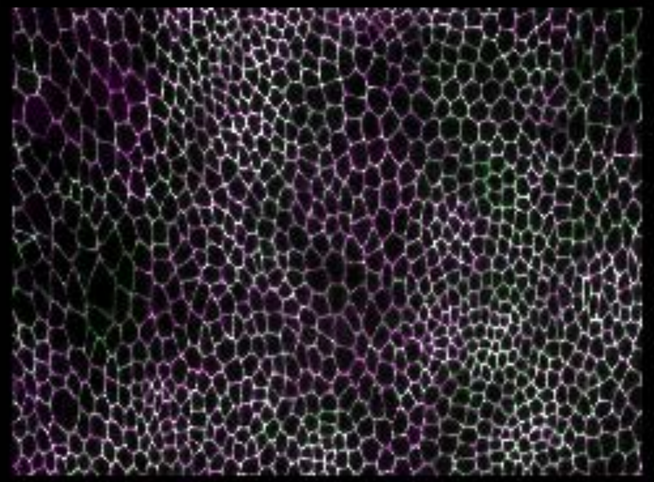
CARE+



GT (PREMOSA)



CARE vs. GT

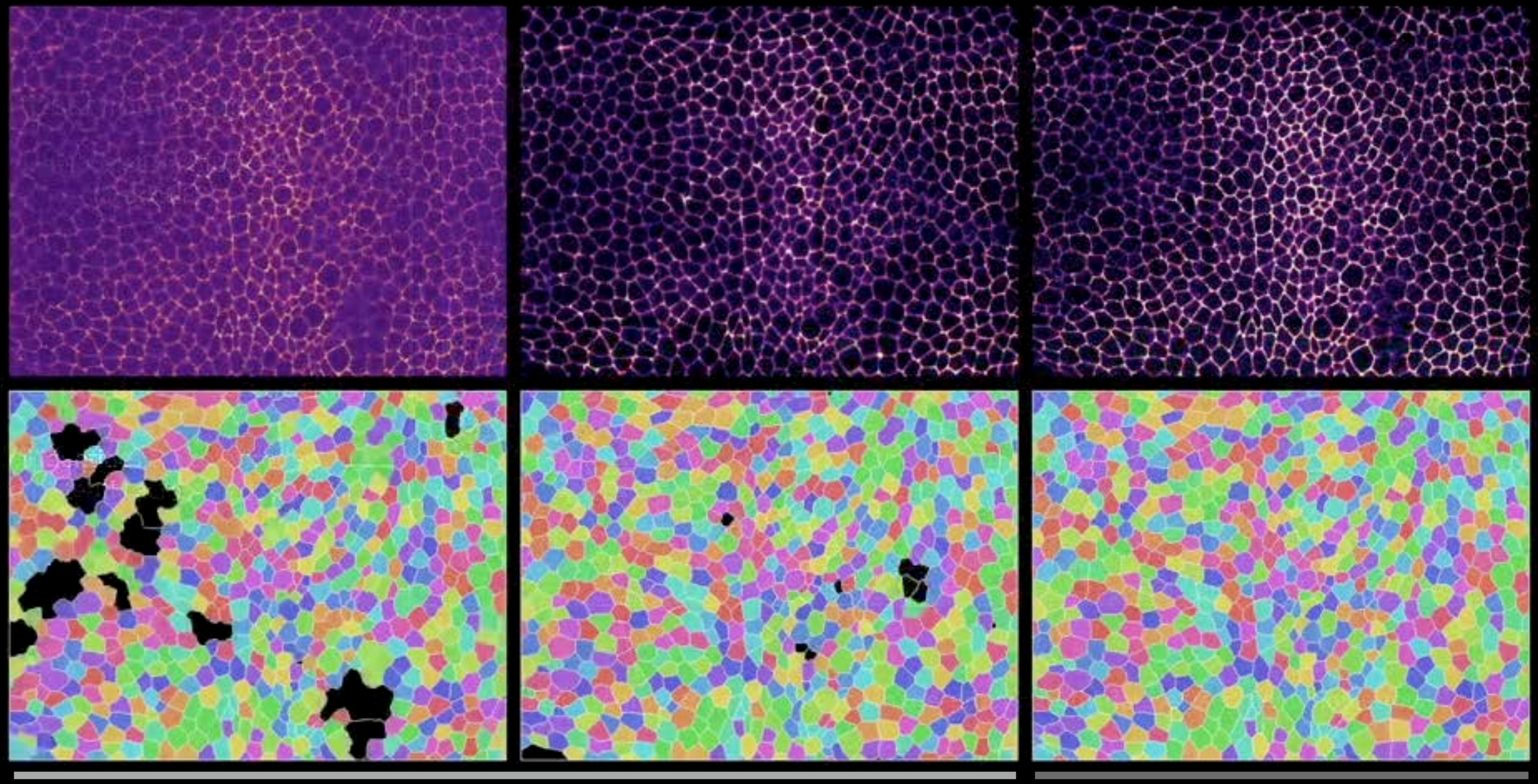


50% laser; 240ms exp.

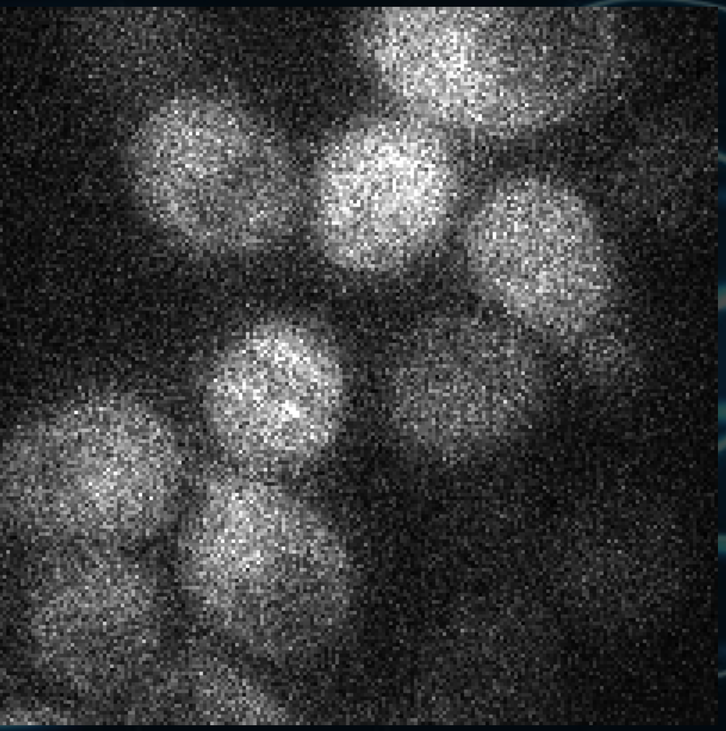
CARE – WHY BOTHER?

>> fly wing, **Eaton Lab**, MPI-CBG

RAW + PREMOSA | CARE+ | GROUND TRUTH

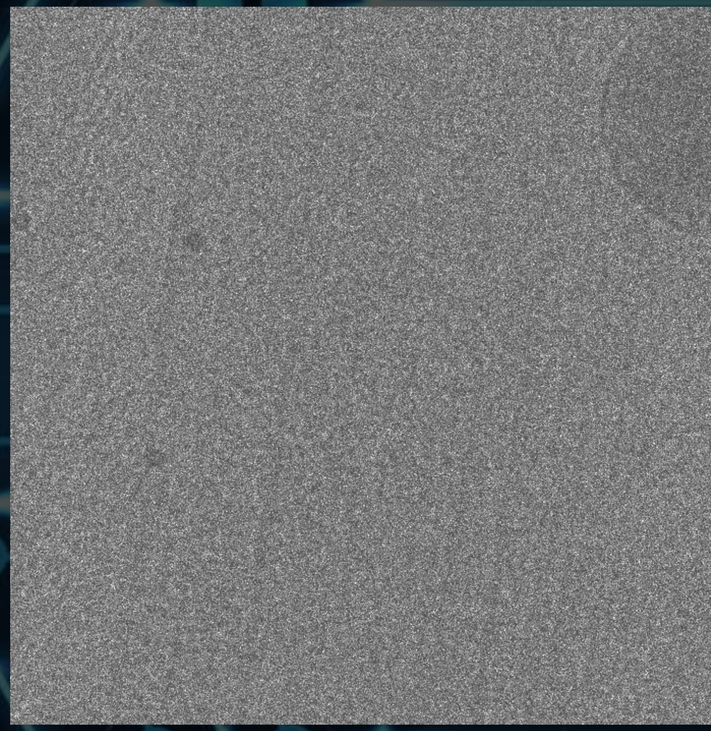


PACKING ANALYZER | + MANUAL CURATION



FAST PROCESSES

WHAT IF SUFFICIENT QUALITY TRAINING DATA CAN'T BE IMAGED?



CRYO-TOMOGRAPHY

Alex Krull

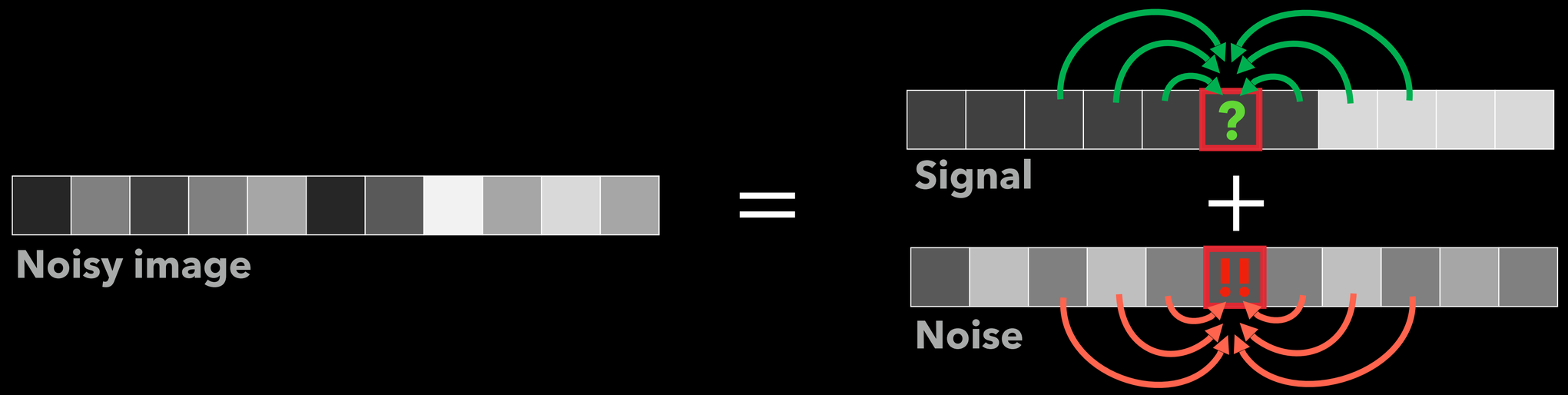
Tim-Oliver Buchholz

Another new idea:

Noise2Void

Krull et al., Noise2Void - Learning Denoising from Single Noisy Images, CVPR'19
Batson and Royer, Noise2Self - Blind Denoising by Self-Supervision, arXiv:1901.11365

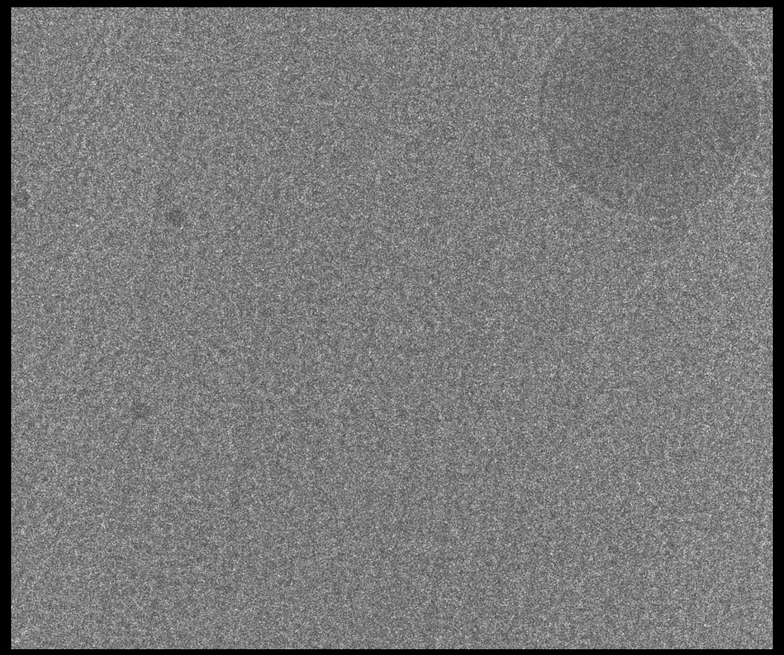
What if you'd only have **one** set of noisy images?



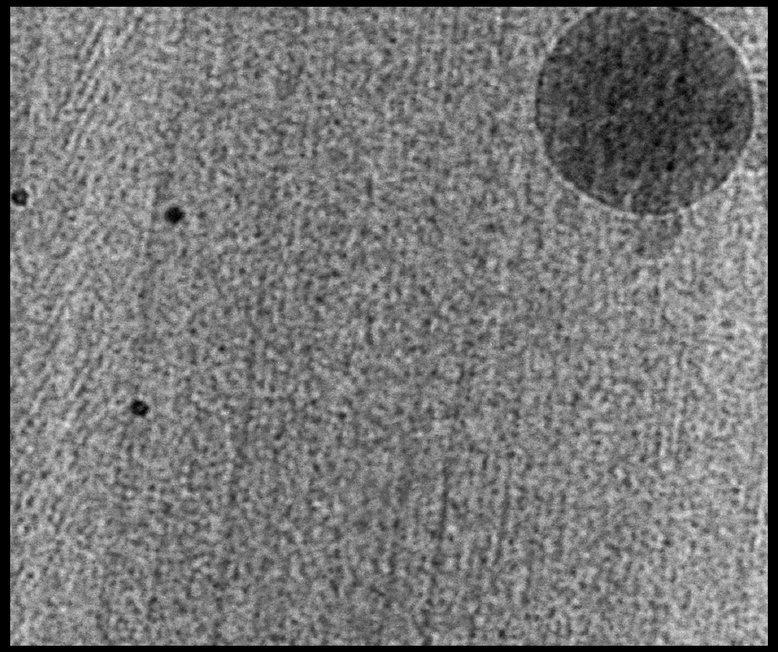
Another new idea:

Noise2Void

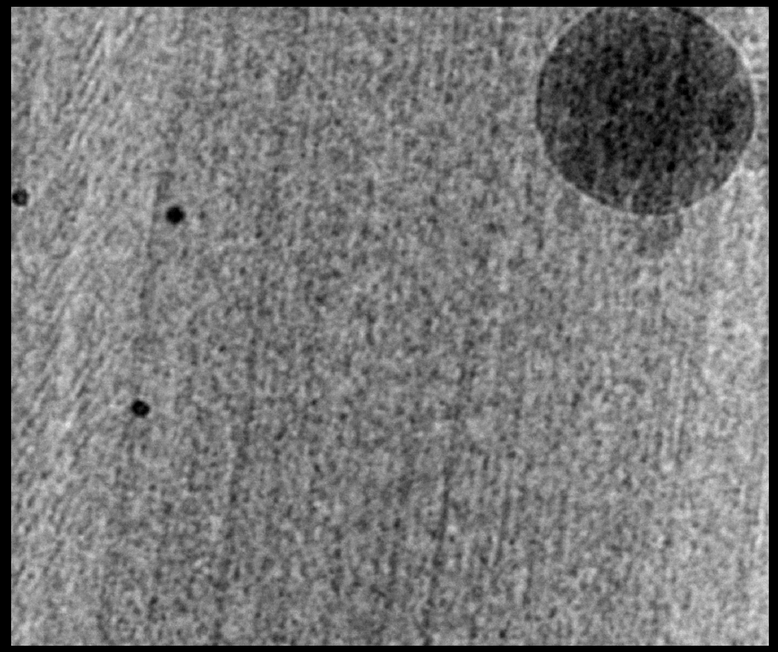
Krull et al., Noise2Void - Learning Denoising from Single Noisy Images, CVPR'19
Batson and Royer, Noise2Self - Blind Denoising by Self-Supervision, arXiv:1901.11365



INPUT

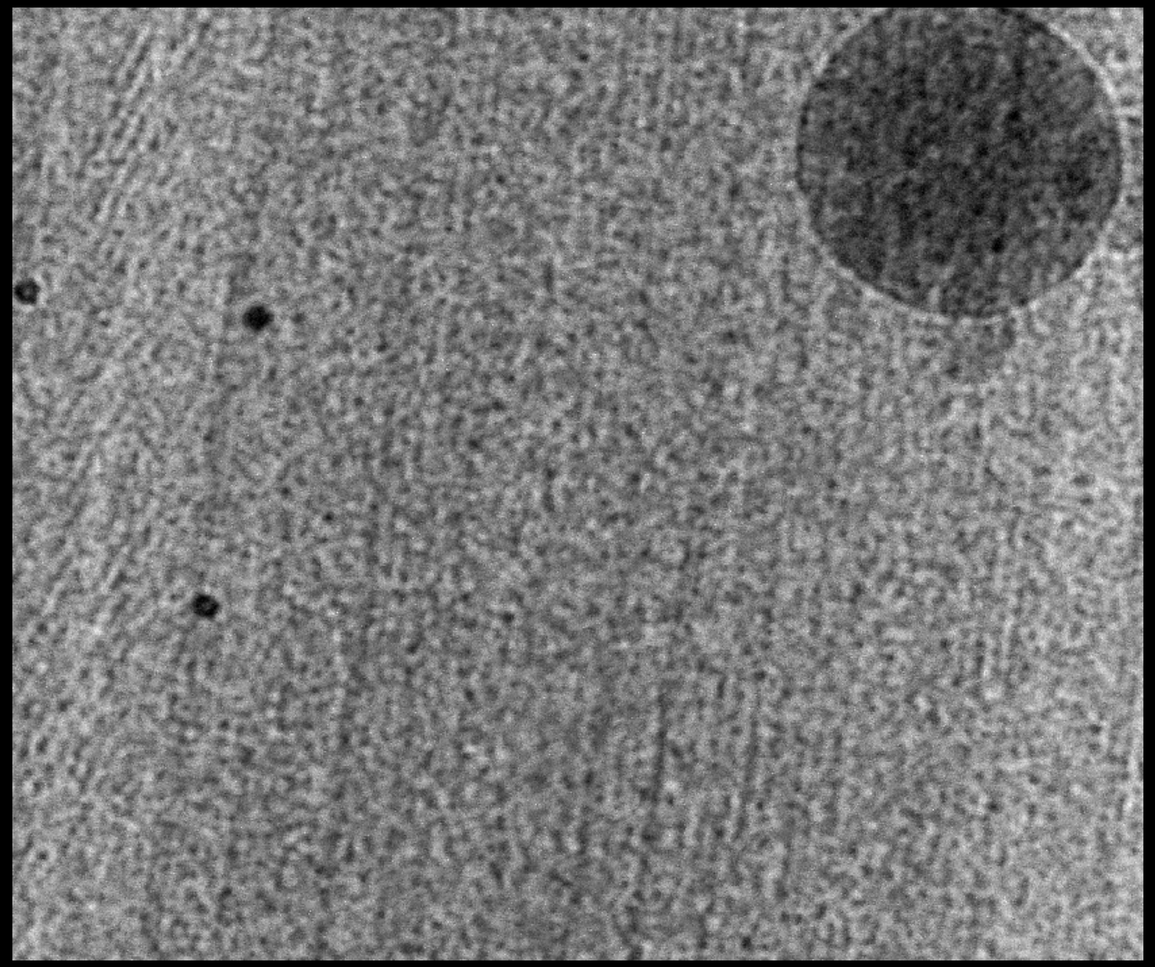


NOISE2VOID

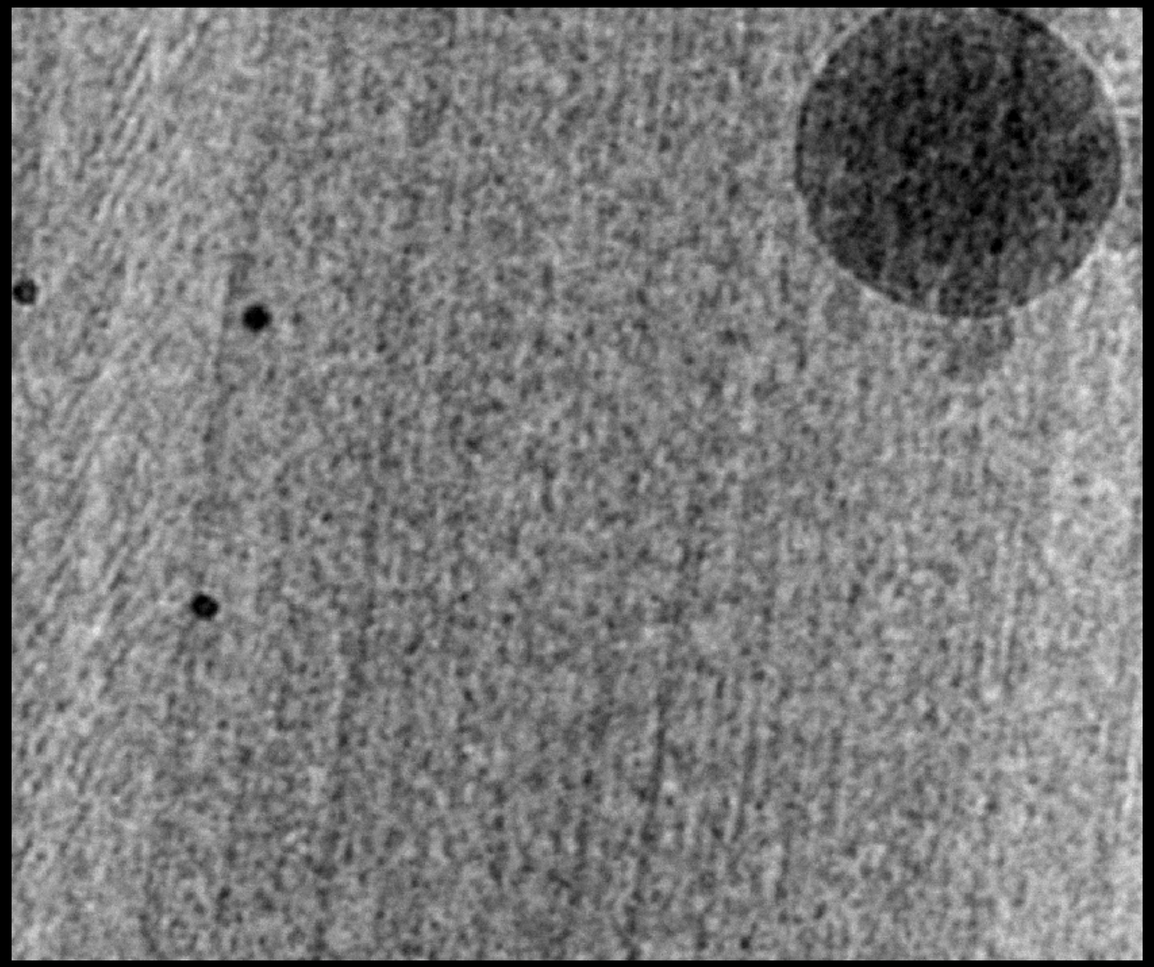


CARE (NOISE2NOISE)

Krull et al., Noise2Void - Learning Denoising from Single Noisy Images, CVPR'19



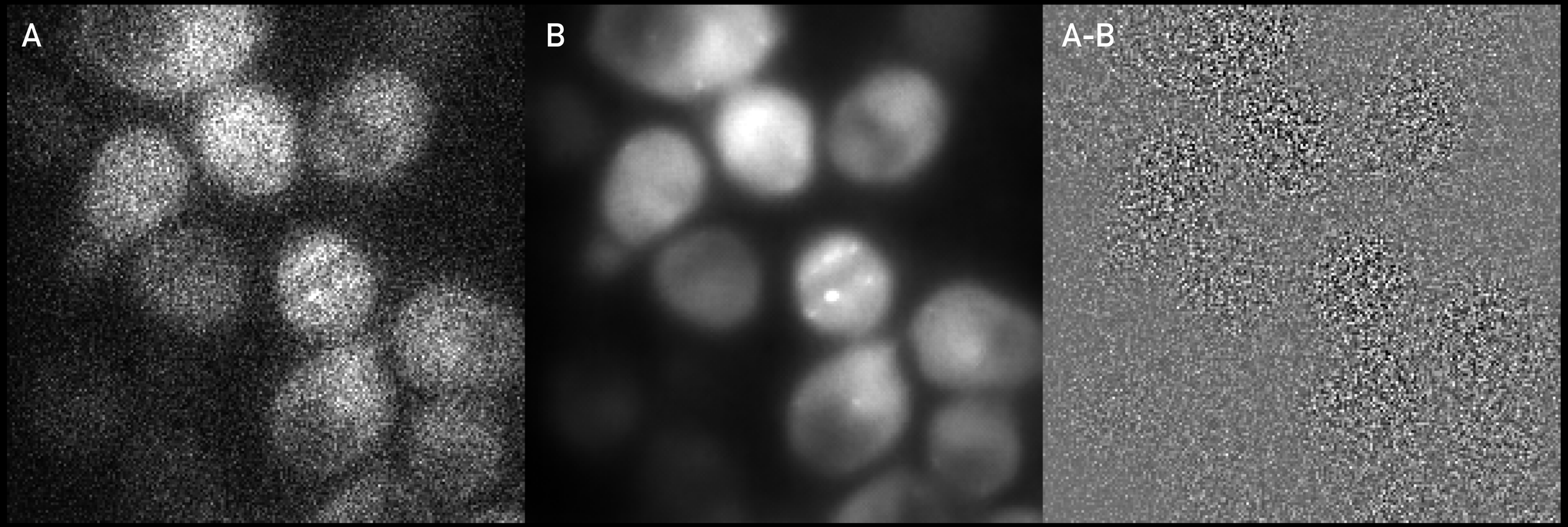
NOISE2VOID



CARE (NOISE2NOISE)

Krull et al., Noise2Void - Learning Denoising from Single Noisy Images, CVPR'19

>> Data: **Stephanie Heinrich**, ETH Zurich



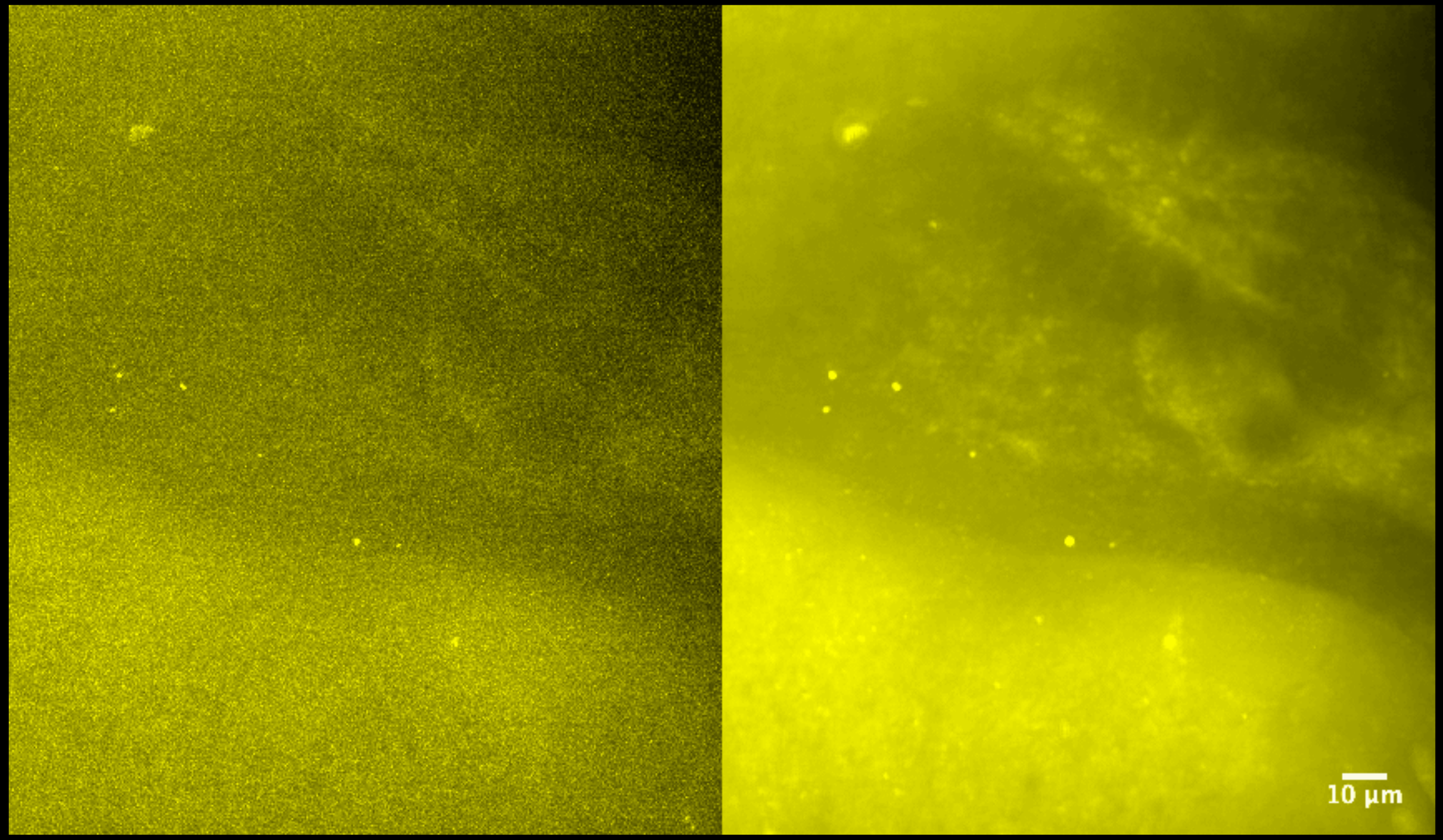
NOISE2VOID

OF COURSE ALSO USEFUL FOR LM DATA!

Krull et al., Noise2Void - Learning Denoising from Single Noisy Images, CVPR'19

NOISE2VOID - ONE MORE EXAMPLE...

>> Data: **Bettencourt-Dias**, at Physiology Course 2019



CSBDeep

CSBDeep - a deep learning toolbox for microscopists.

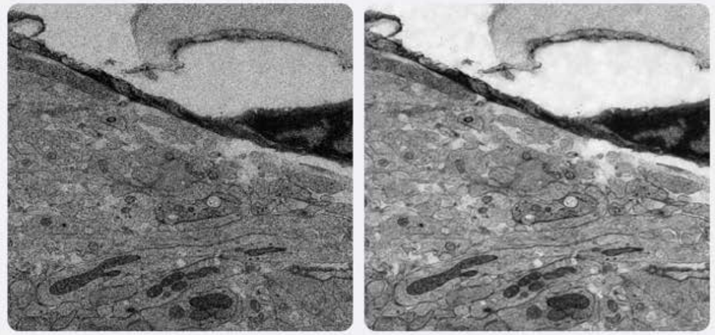
What can I do with my data?

Available tools

Questions

Contact Impressum

Denoising Noise2Void without clean data (2D SEM data)



Authorship

The data was acquired by Réza Shahidi and Gáspár Jékely at University of Exeter and originates from this publication restoration for electron microscopy by Tim-Oliver Buchholz *et al.*

Details

The model was trained and the output predicted by using the `train + predict` command of Noise2Void in Fiji.

Demo

For demonstration purposes, the trained model was uploaded to the BioImage Model Zoo, you can download it here.

The downloaded model can be used for prediction via the `predict` command of Noise2Void in Fiji.

Videos

Search or jump to... / Pull requests Issues Marketplace Explore

HenriquesLab / ZeroCostDL4Mic

Watch 27 Star 213 Fork 47

Code Issues 11 Pull requests 10 Actions Projects Wiki Security Insights

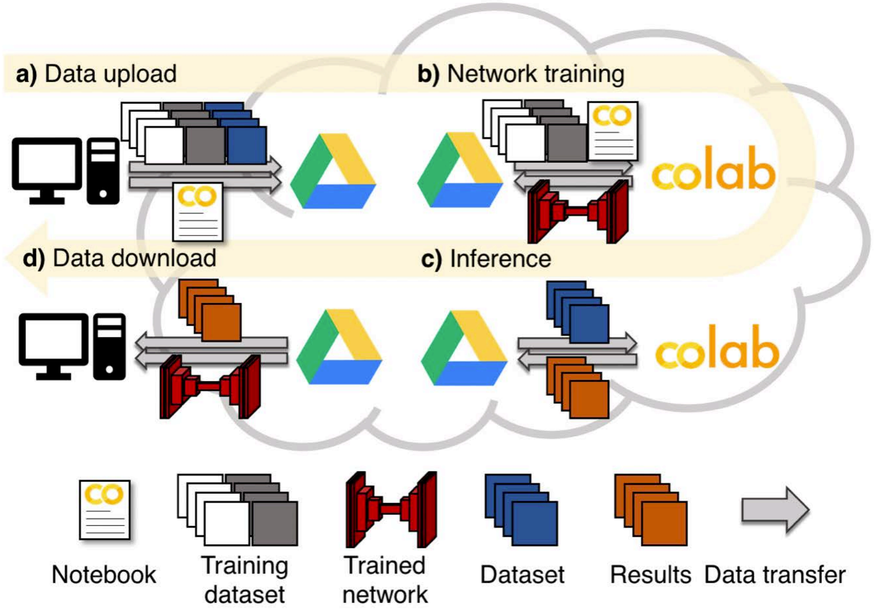
Home

Romain F. Laine edited this page 20 days ago · 158 revisions

ZeroCostDL4Mic - What is it?

Overview

ZeroCostDL4Mic is a toolbox for the training and implementation of common Deep Learning approaches to microscopy imaging. It exploits the ease-of-use and access to GPU provided by **Google Colab**.



Training data can be uploaded to the Google Drive from where it can be used to train models using the provided Colab notebooks in a web-browser. Inference (predictions) on unseen data can then also be performed within the same notebook, therefore not requiring any local hardware or software set-up.

Pages 21

Main:

- Home
- Step by step "How to" guide
- How to contribute
- Tips, tricks and FAQs
- Data augmentation
- Quality control
- Running notebooks locally
- Running notebooks on FloydHub

Fully supported networks:

- U-Net
- StarDist
- Noise2Void
- CARE
- Label free prediction (fnet)
- Object Detection (YOLOv2)
- pix2pix
- CycleGAN
- Deep-STORM

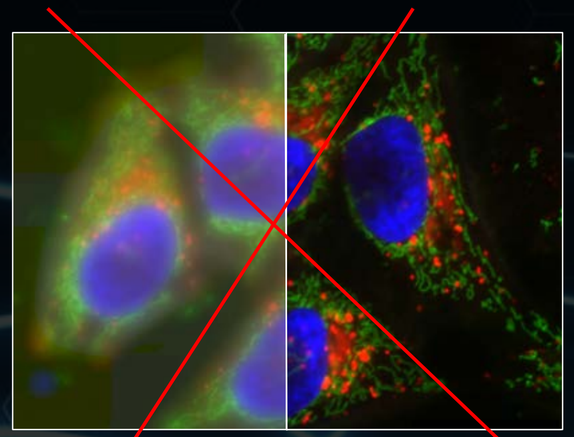
Other resources:

- Glossary
- Useful resources
- Example gallery

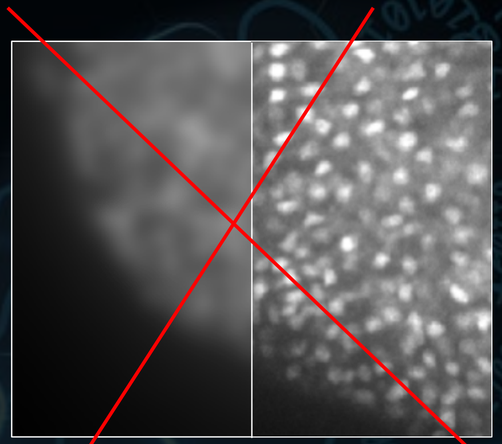
Clone this wiki locally

https://github.com/Henric

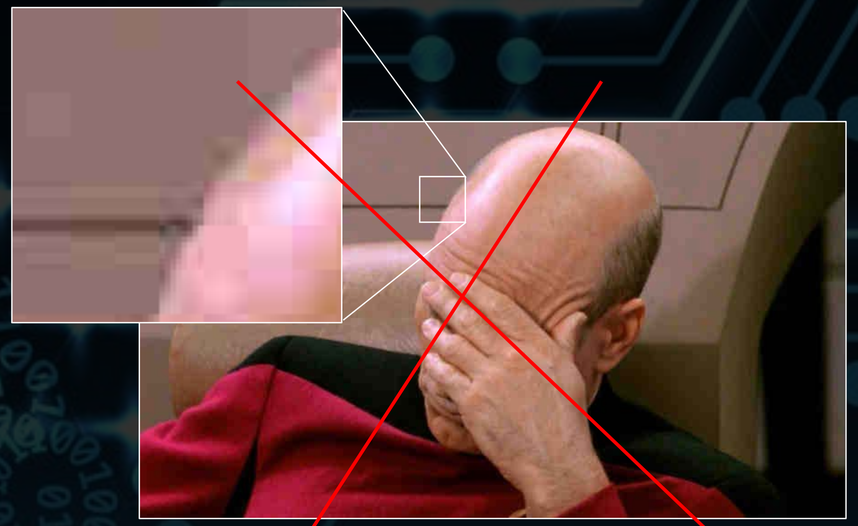
LIMITATIONS OF NOISE2VOID?



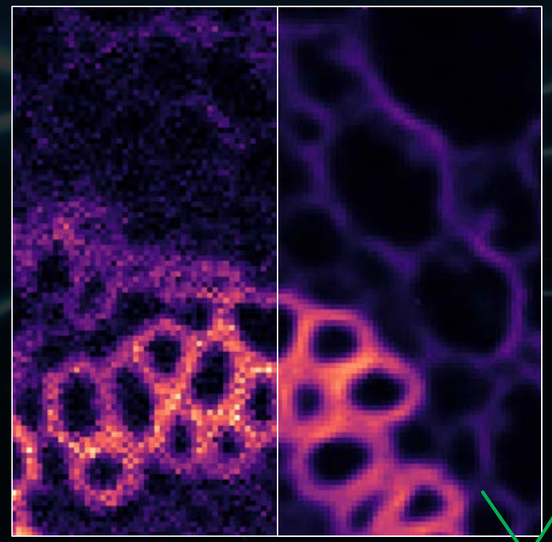
Computational clearing



Deconvolution



JPEG compression artifacts



per-pixel noise

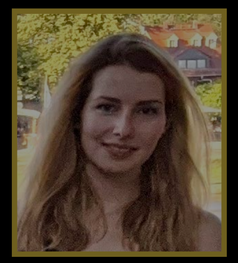
- Low light images
- Shot noise
- Readout noise
- Non-structured noise

Take home message:

Noise2Void is as easy as it gets if reducing per-pixel noise is the goal!

a variation on N2V:

DecoNoising



Anna Goncharova

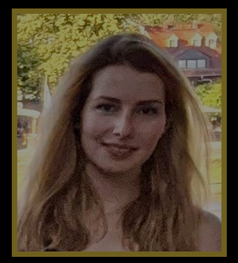
Alf Honigmann

Alex Krull

Goncharova et al., Improving Blind Spot Denoising for Microscopy, BIC@ECCV 2020

a variation on N2V:

DecoNoising



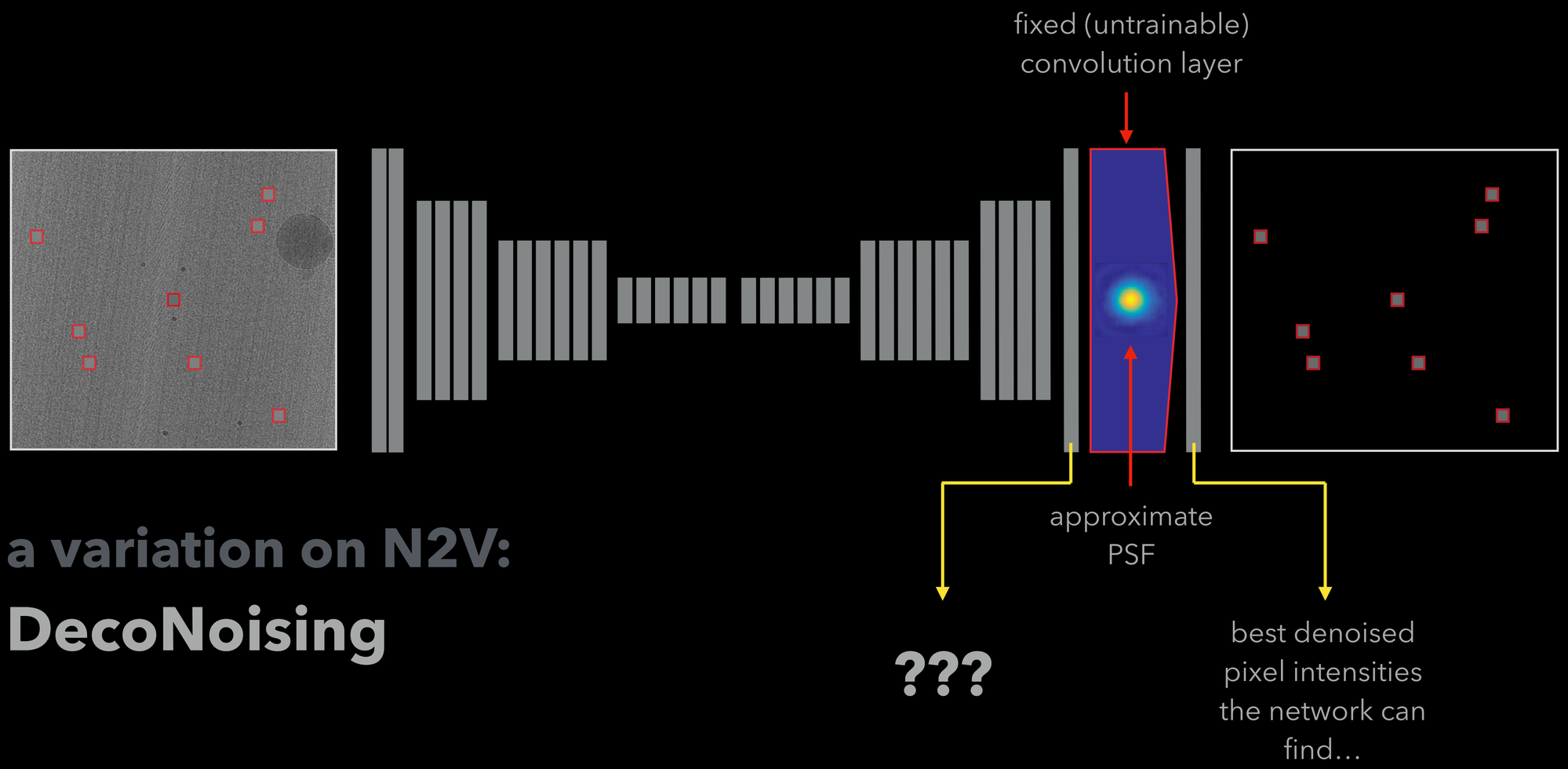
Anna Goncharova

Alf Honigmann

Alex Krull

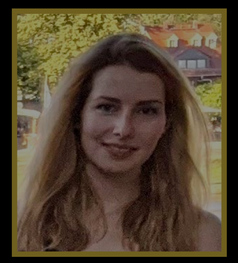
Goncharova et al., Improving Blind Spot Denoising for Microscopy, BIC@ECCV 2020

DecoNoising — a noise2void extension



a variation on N2V:

DecoNoising



Anna Goncharova

Alf Honigmann

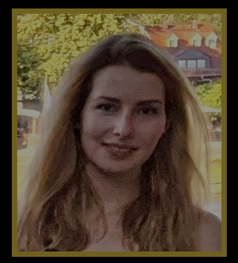
Alex Krull

Goncharova et al., Improving Blind Spot Denoising for Microscopy, BIC@ECCV 2020

DecoNoising — a noise2void extension

a variation on N2V:

DecoNoising



Anna
Goncharova

Alf
Honigmann

Alex
Krull

Goncharova et al., Improving Blind Spot Denoising for Microscopy, BIC@ECCV 2020

Search or jump to... / Pull requests Issues Marketplace Explore

juglab / DecoNoising

Unwatch 7 | Star 3 | Fork 0

Code | Issues | Pull requests | Actions | Projects | Wiki | Security | Insights | Settings

master | 1 branch | 0 tags | Go to file | Add file | Code

alex-krull initial commit 780acec on 22 Aug 2020 1 commit

deconoising	initial commit	10 months ago
unet	initial commit	10 months ago
.gitignore	initial commit	10 months ago
Convallaria-Prediction.ipynb	initial commit	10 months ago
Convallaria-Training.ipynb	initial commit	10 months ago
LICENSE.txt	initial commit	10 months ago
README.md	initial commit	10 months ago
script_training.py	initial commit	10 months ago

README.md

DecoNoising: Improving Blind Spot Denoising for Microscopy

Anna S. Goncharova, Alf Honigmann, Florian Jug, and Alexander Krull
Max Planck Institute of Molecular Cell Biology and Genetics (MPI-CBG)
Center for Systems Biology (CSBD) in Dresden, Germany

You can find the paper here.

Abstract

Many microscopy applications are limited by the total amount of usable light and are consequently challenged by the resulting levels of noise in the acquired images. This problem is often addressed via (supervised) deep learning based denoising. Recently, by making assumptions about the noise statistics, self-supervised methods have emerged. Such methods are trained directly on the images that are to be denoised and do not require additional paired training data. While achieving remarkable results, self-supervised methods can produce high-frequency artifacts and achieve inferior results compared to supervised approaches. Here we

About

No description, website, or topics provided.

Readme

View license

Releases

No releases published
Create a new release

Packages

No packages published
Publish your first package

Contributors 3

anngoncharova
alex-krull Alexander Krull
fjug Florian Jug

Languages

Jupyter Notebook 99.4%
Python 0.6%

Search or jump to... / Pull requests Issues Marketplace Explore

HenriquesLab / ZeroCostDL4Mic

Watch 27 | Star 213 | Fork 47

Code | Issues 11 | Pull requests 10 | Actions | Projects | Wiki | Security | Insights

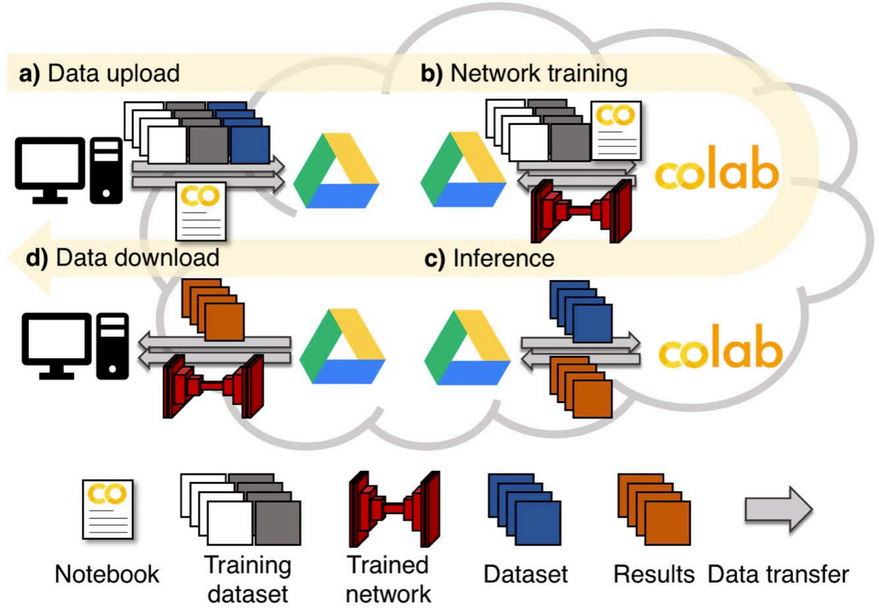
Home

Romain F. Laine edited this page 20 days ago · 158 revisions

ZeroCostDL4Mic - What is it?

Overview

ZeroCostDL4Mic is a toolbox for the training and implementation of common Deep Learning approaches to microscopy imaging. It exploits the ease-of-use and access to GPU provided by Google Colab.



Training data can be uploaded to the Google Drive from where it can be used to train models using the provided Colab notebooks in a web-browser. Inference (predictions) on unseen data can then also be performed within the same notebook, therefore not requiring any local hardware or software set-up.

Pages 21

Main:

- Home
- Step by step "How to" guide
- How to contribute
- Tips, tricks and FAQs
- Data augmentation
- Quality control
- Running notebooks locally
- Running notebooks on FloydHub

Fully supported networks:

- U-Net
- StarDist
- Noise2Void
- CARE
- Label free prediction (fnet)
- Object Detection (YOLOv2)
- pix2pix
- CycleGAN
- Deep-STORM

Other resources:

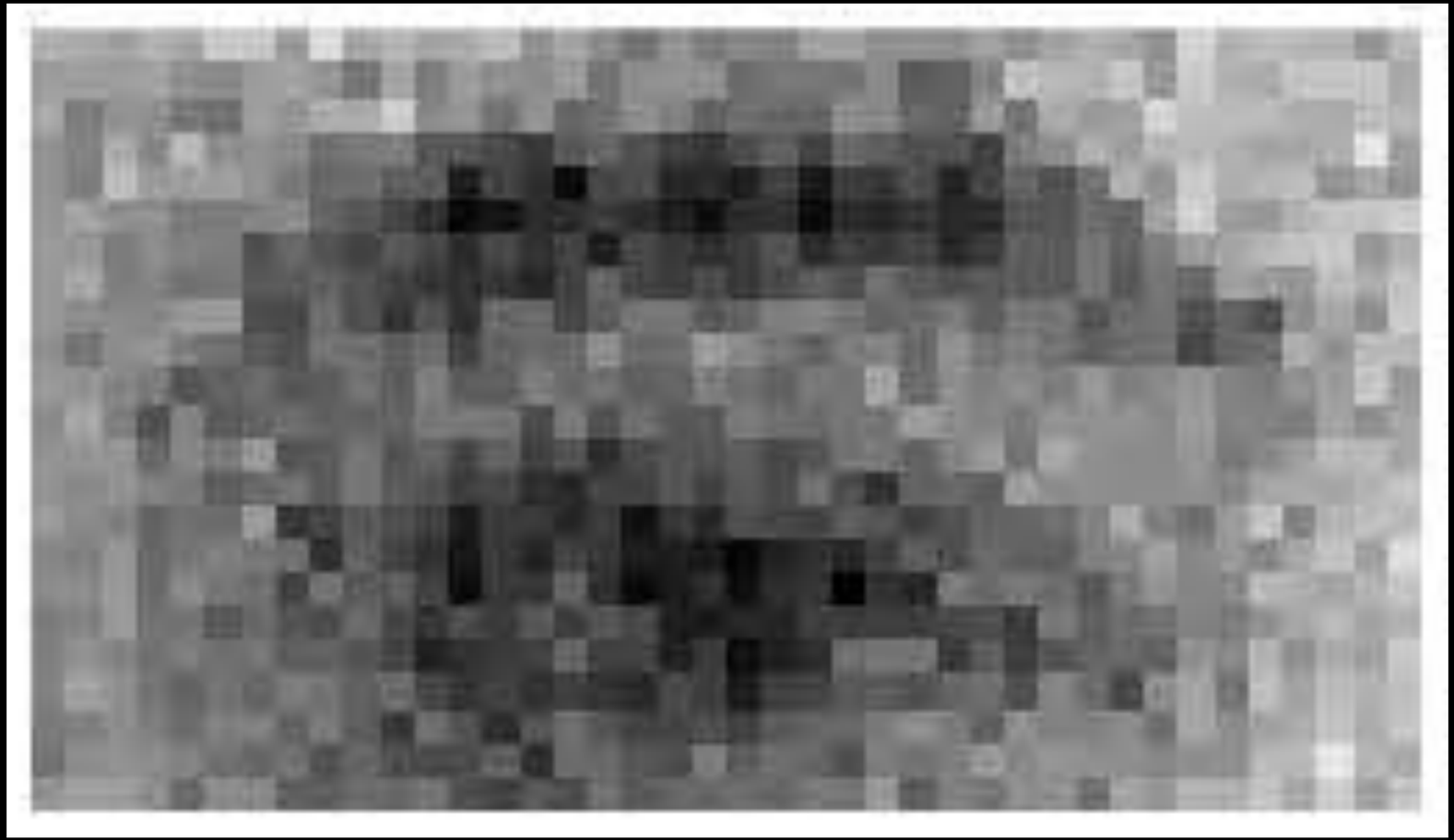
- Glossary
- Useful resources
- Example gallery

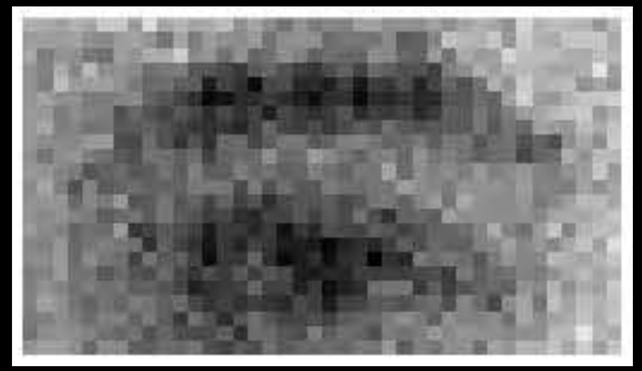
Clone this wiki locally

https://github.com/Henric

1. CONTENT-AWARE RESTORATION
2. DIVERSITY DENOISING
3. SHOULD YOU USE DL METHODS?
4. WHAT WILL THE FUTURE BRING?

CAN THE TRUE SIGNAL IN A NOISY INPUT BE RESTORED WITH CERTAINTY?





INPUT (crop)

Ground Truth

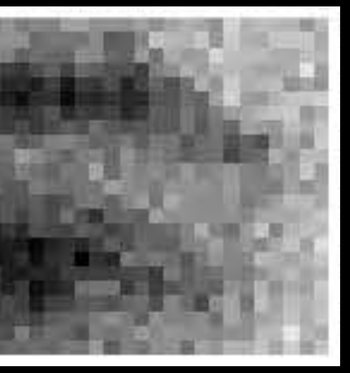
DivNoising (sample 1)

DivNoising (sample 2)

Mangal Prakash

Alex Krull

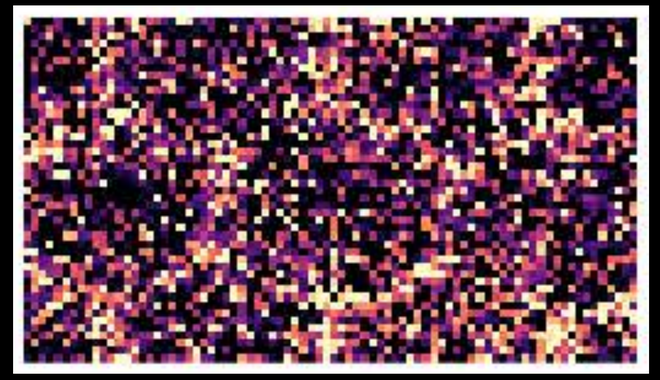
Prakash et al., DivNoising: Diversity Denoising with Fully Convolutional Variational Autoencoders, ICLR'21



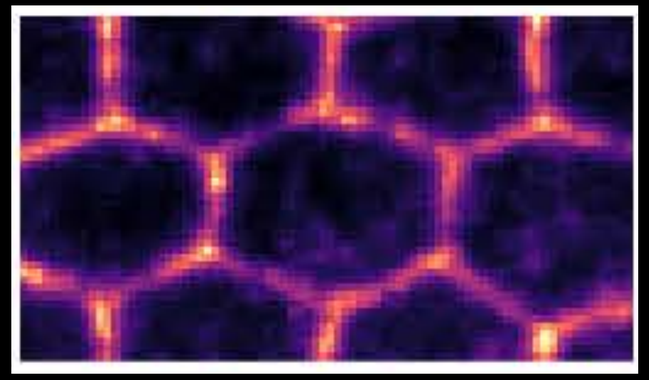
Ground Truth

1)

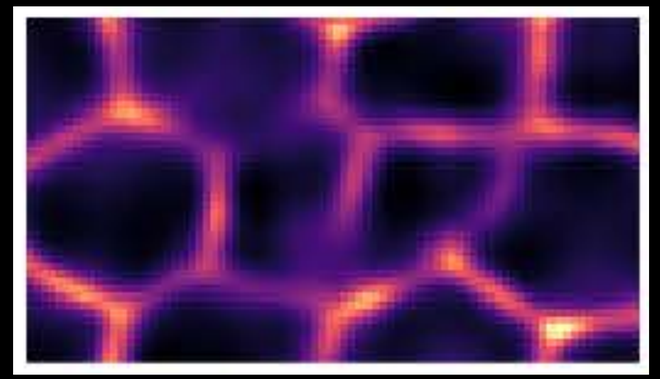
DivNoising (sample 2)



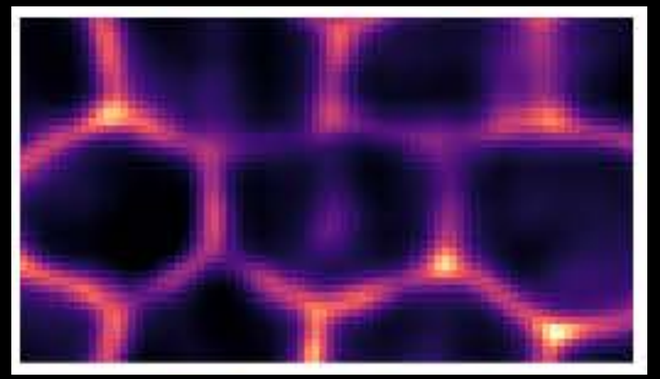
INPUT (crop)



Ground Truth

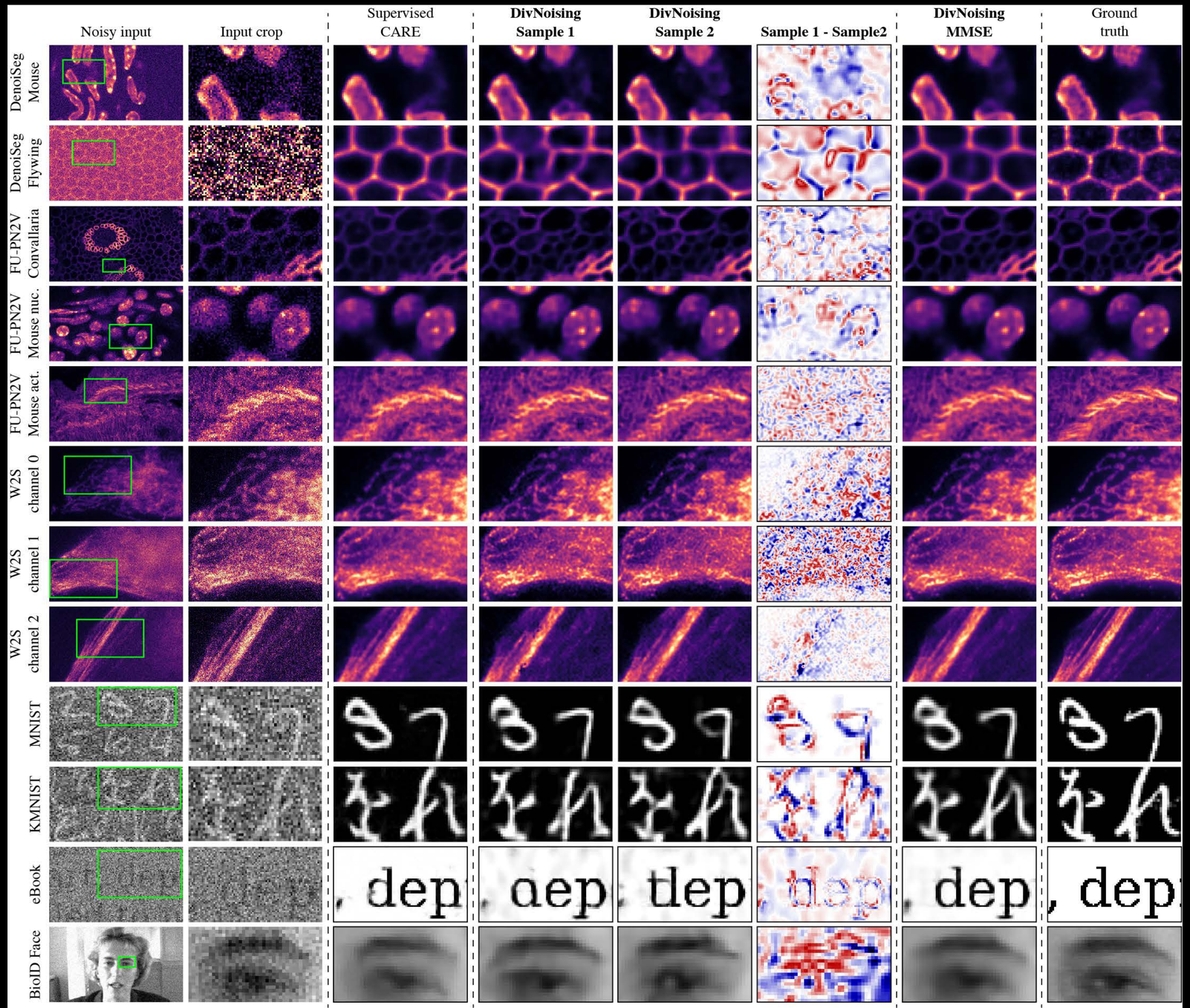


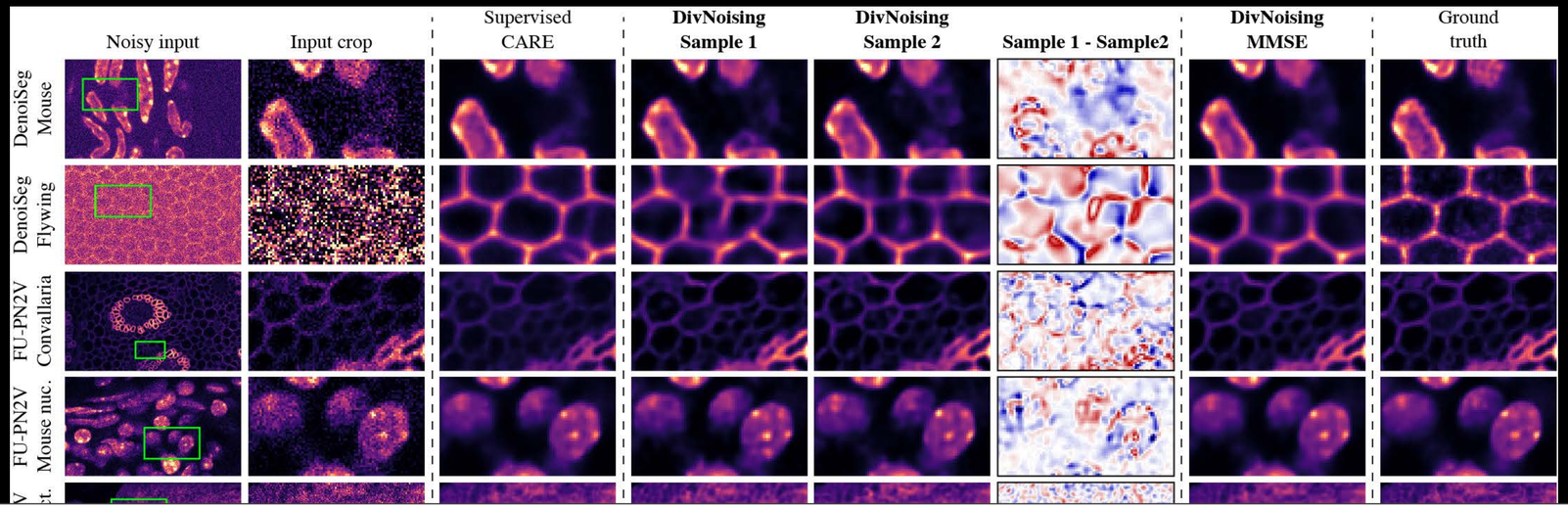
DivNoising (sample 1)



DivNoising (sample 2)

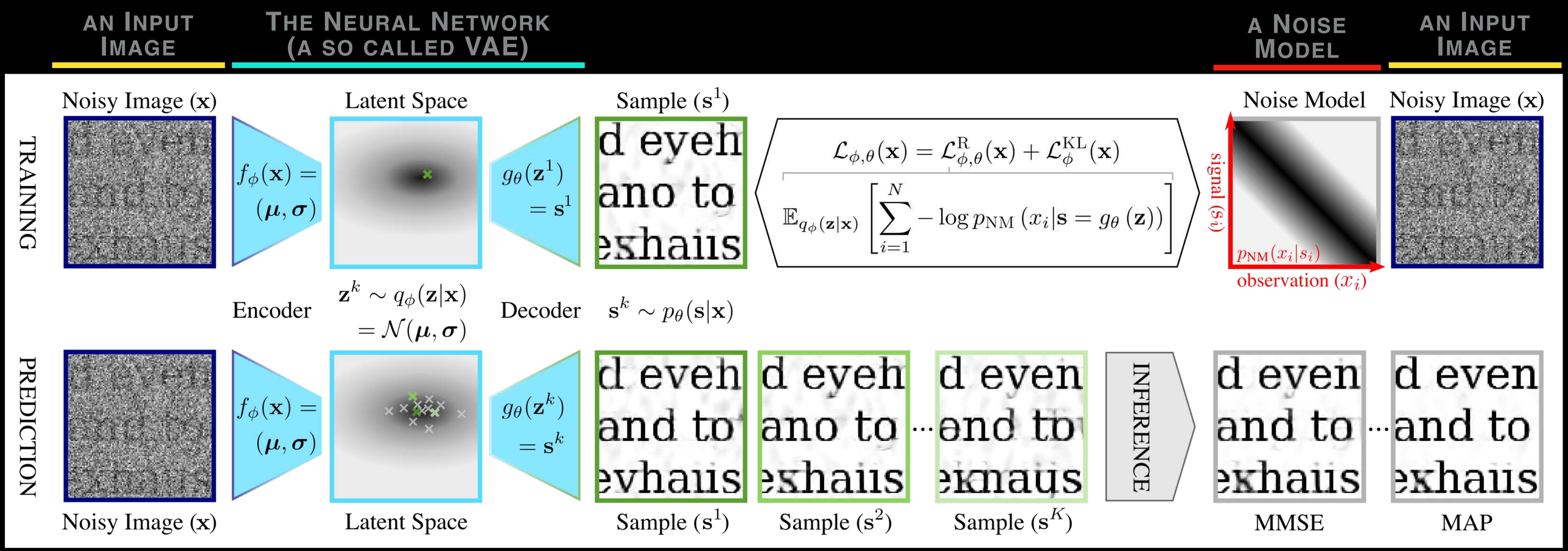
Prakash et al., DivNoising: Diversity Denoising with Fully Convolutional Variational Autoencoders, ICLR'21


Noisy input
Input crop
Supervised CARE
DivNoising Sample 1
DivNoising Sample 2
Sample 1 - Sample2
DivNoising MMSE
Ground truth
DenoiSeg Mouse
DenoiSeg Flywing
FU-PN2V Convallaria
FU-PN2V Mouse nuc.
FU-PN2V Mouse act.
W2S channel 0
W2S channel 1
W2S channel 2
MNIST
KMNIST
eBook
BioID Face



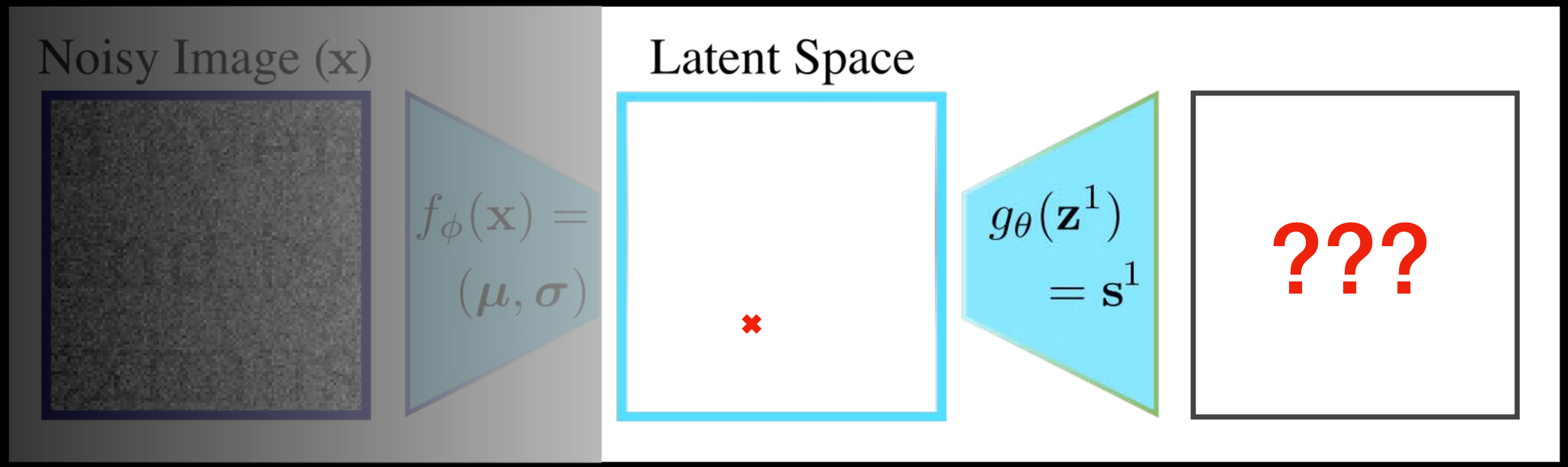
	Dataset	Unsupervised N2V	Unsupervised PN2V	Unsupervised DivNoising (MMSE)	Supervised CARE
DenoiseSeg	Mouse	33.55	**34.19**	34.03	34.63
	Flywing	22.85	24.85	**25.10**	25.60
FU-PN2V	Convallaria	35.73	36.47	**36.94**	36.71
	Convallaria (bootstr.)	35.73	**36.70**	36.63	36.71
	Mouse Actin	33.39	33.86	**33.98**	34.20
	Mouse Nuclei	35.84	**36.35**	36.31	36.58
W2S	Ch. 0 (avg1)	**34.36**	-	34.13	34.30
	Ch. 1 (avg1)	31.96	-	**32.28**	32.11
	Ch. 2 (avg1)	34.78	32.48	**35.18**	34.73
	Ch. 0 (avg16)	38.83	39.19	**39.62**	41.94
	Ch. 1 (avg16)	37.80	38.24	**38.37**	39.09
	Ch. 2 (avg16)	40.17	40.49	**40.52**	40.88

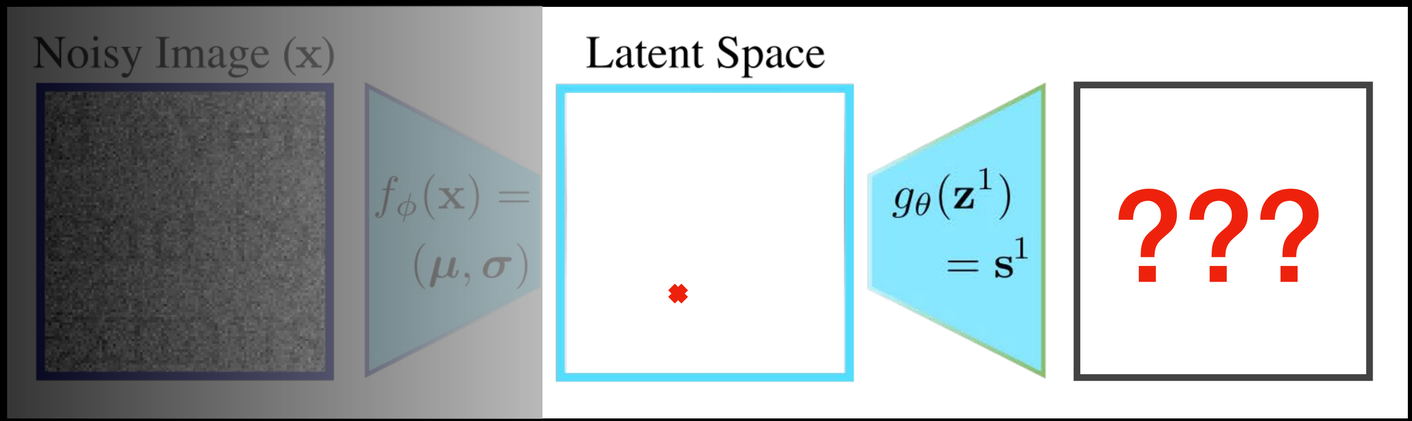
QUICK GLANCE UNDER THE HOOD…

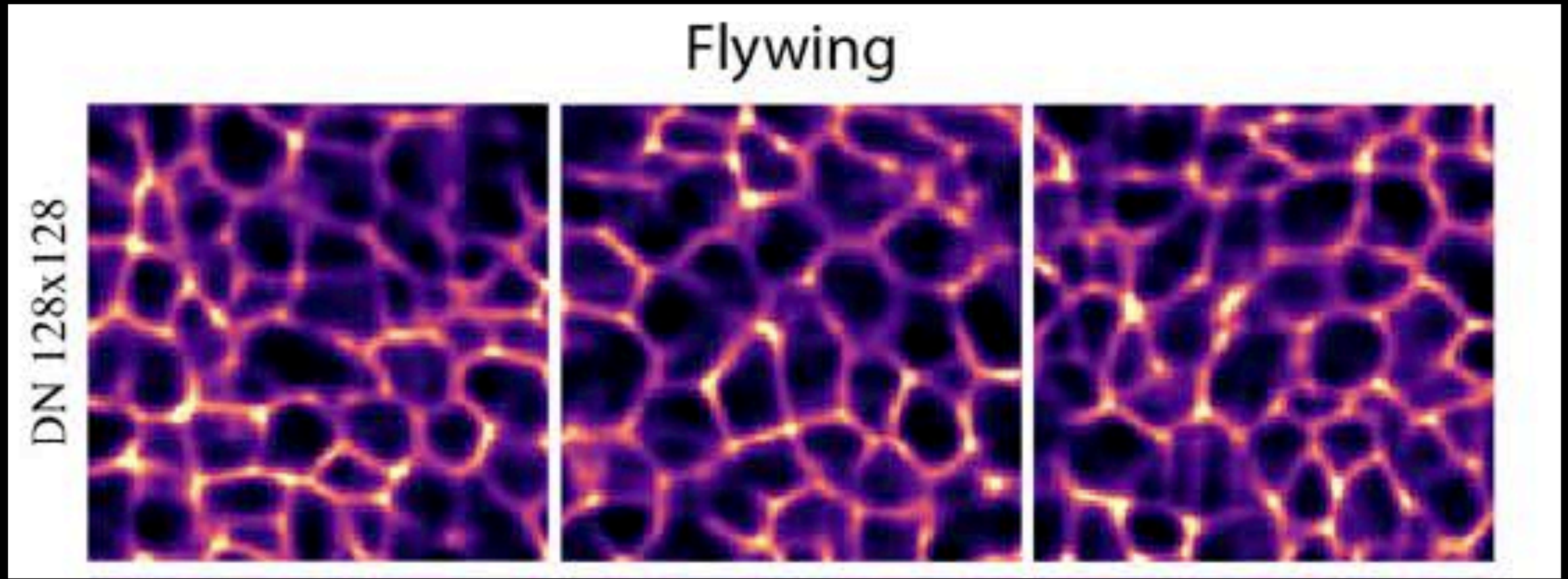


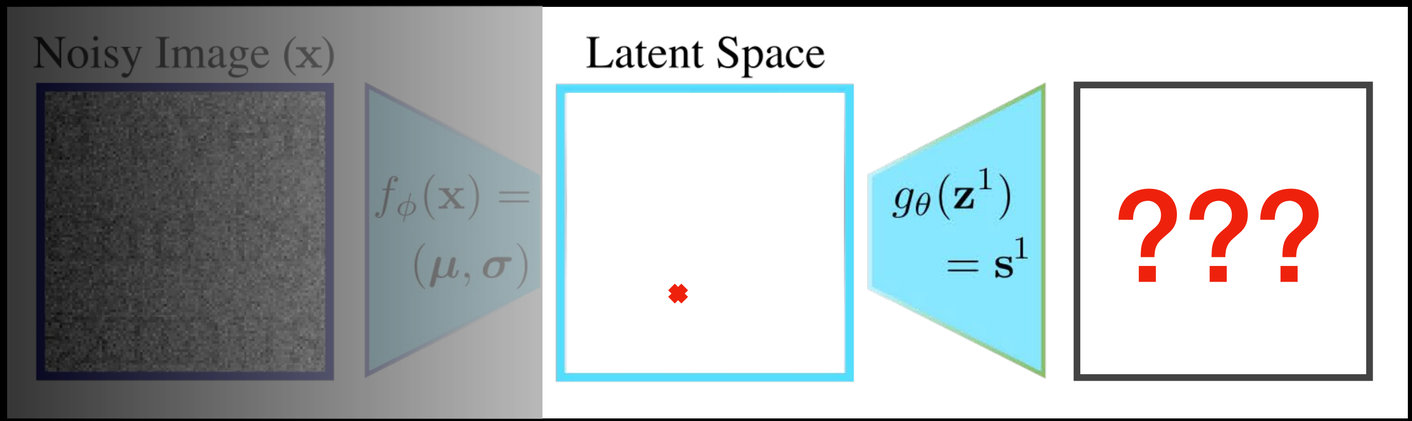
"THE MAGIC" HAPPENS HERE… ;)

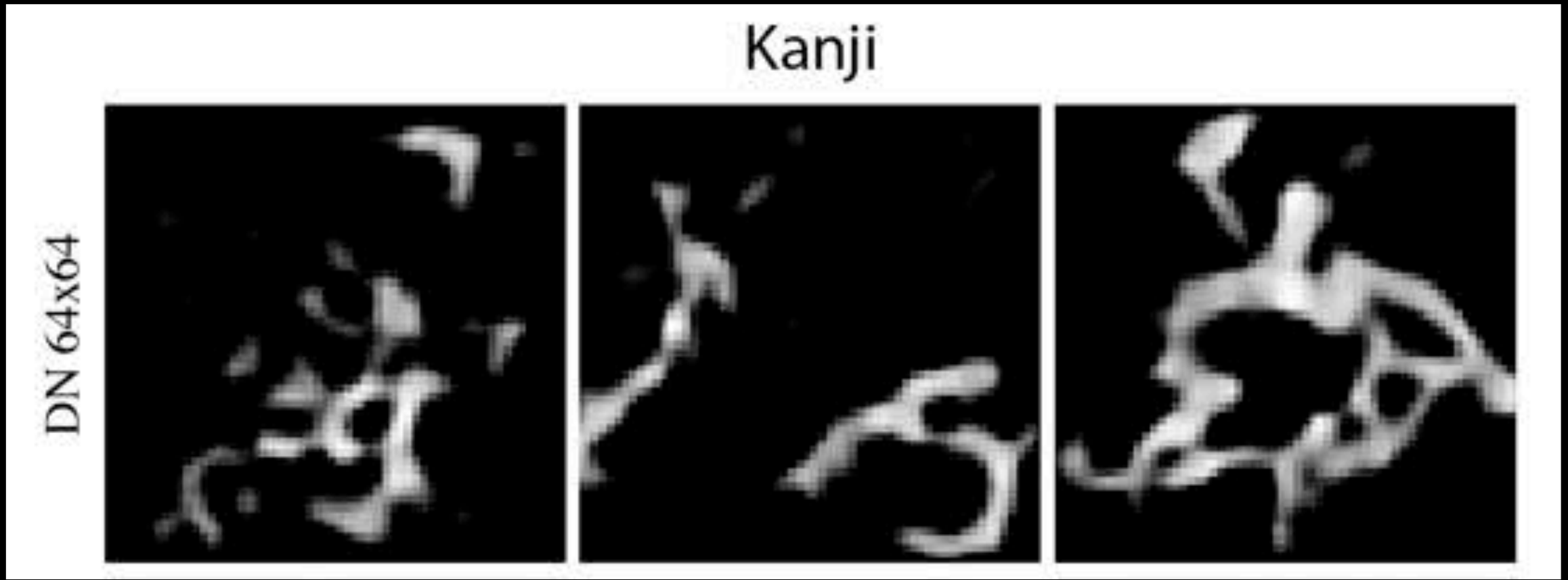
Prakash et al., DivNoising: Diversity Denoising with Fully Convolutional Variational Autoencoders, ICLR'21

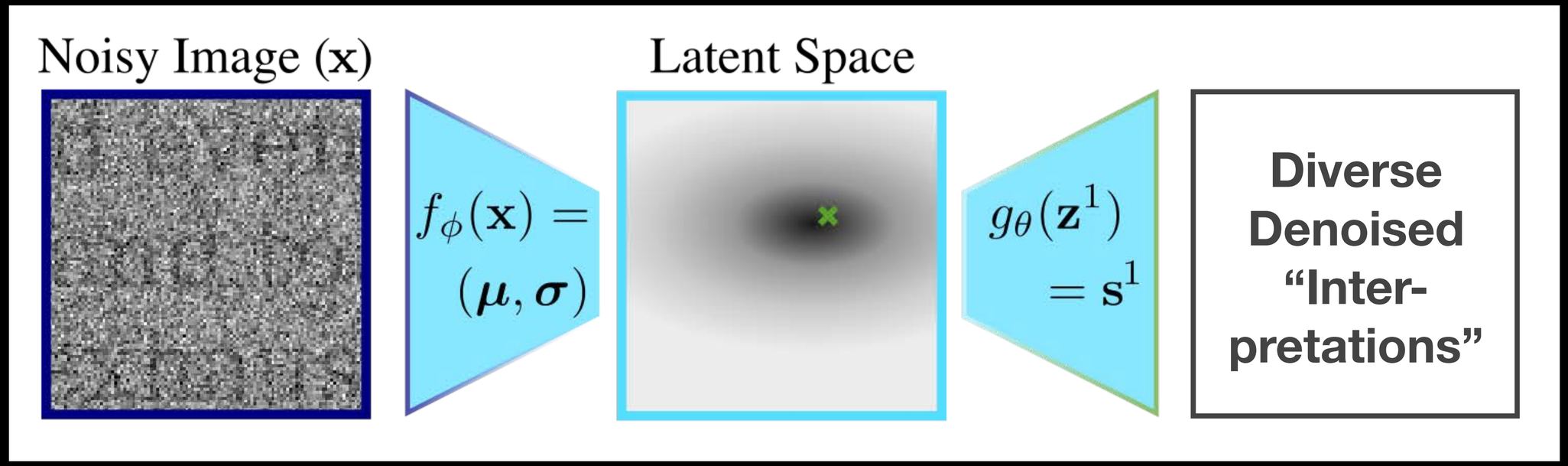

Noisy Image (x)
$f_\phi(\mathbf{x}) = (\boldsymbol{\mu}, \boldsymbol{\sigma})$
Latent Space
$g_\theta(\mathbf{z}^1) = \mathbf{s}^1$
???


Noisy Image (x)
$f_\phi(\mathbf{x}) = (\boldsymbol{\mu}, \boldsymbol{\sigma})$
Latent Space
$g_\theta(\mathbf{z}^1) = \mathbf{s}^1$
???


Flywing
DN 128x128


Noisy Image (x)
$f_\phi(\mathbf{x}) = (\boldsymbol{\mu}, \boldsymbol{\sigma})$
Latent Space
$g_\theta(\mathbf{z}^1) = \mathbf{s}^1$
???


Kanji
DN 64x64

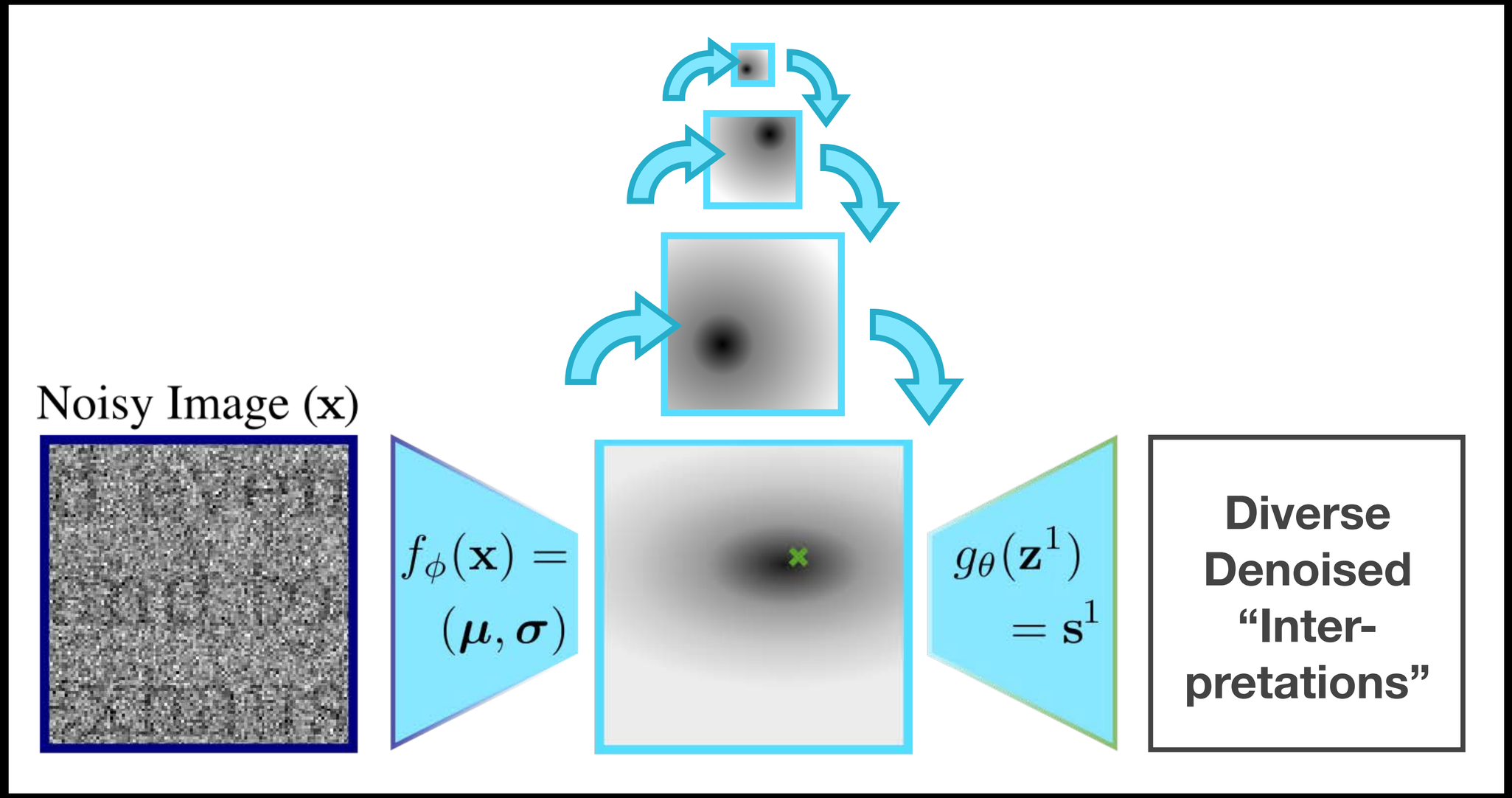


Mangal Prakash

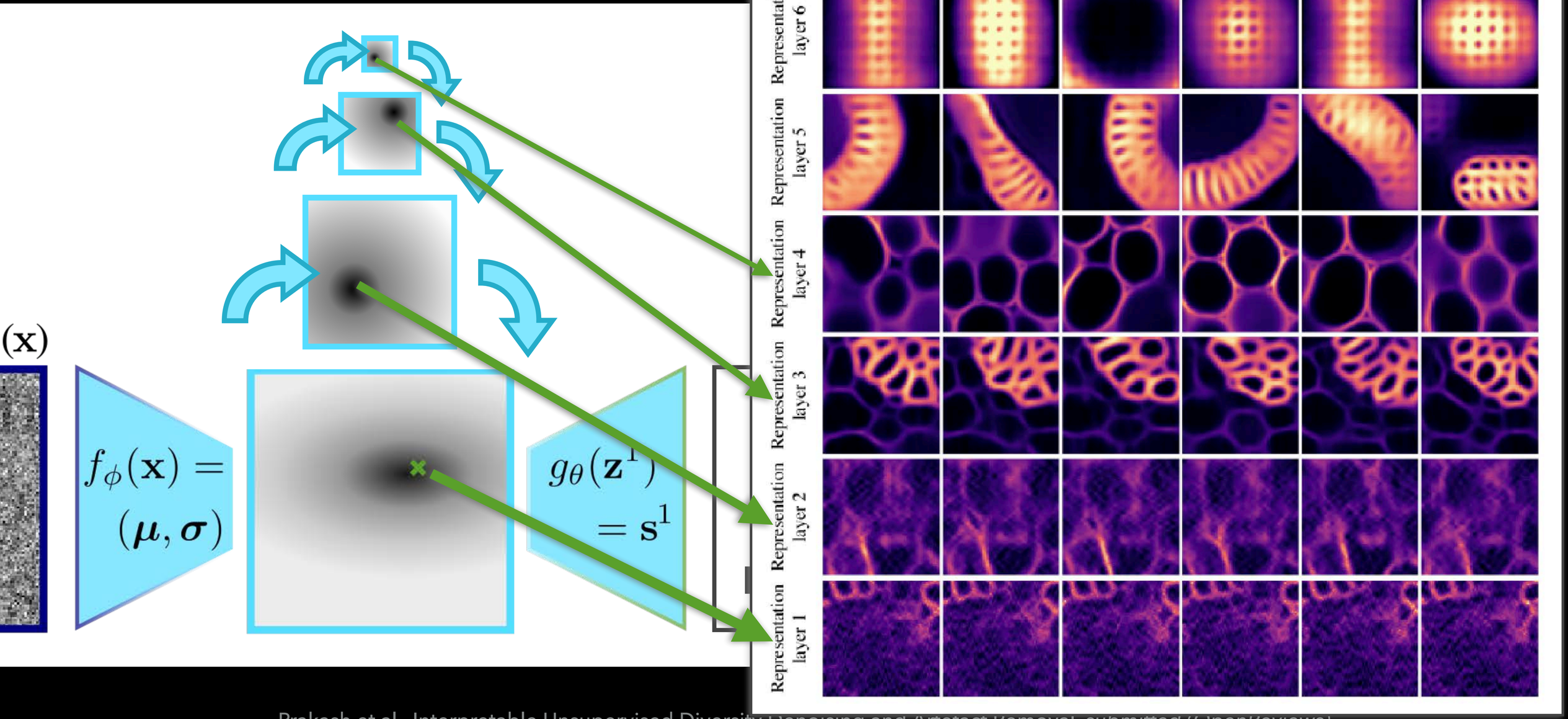
Mauricio Delbracio

Peyman Milanfar

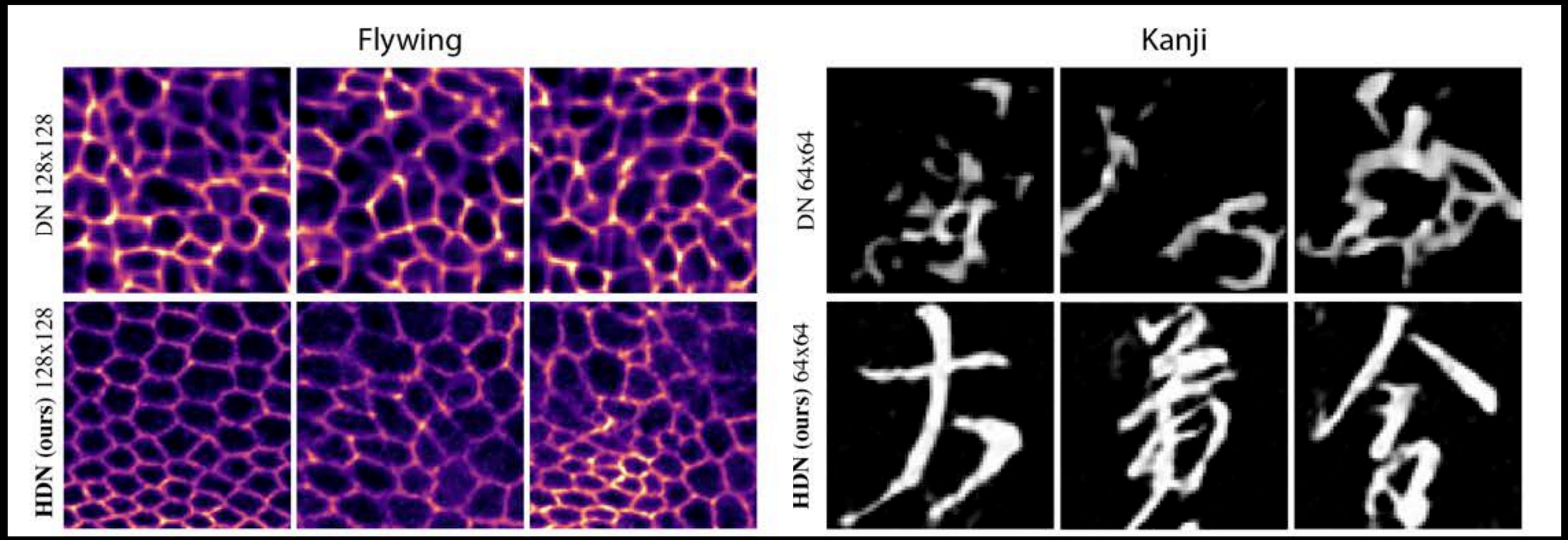
Prakash et al., Interpretable Unsupervised Diversity Denoising and Artefact Removal, *submitted (OpenReviews)*



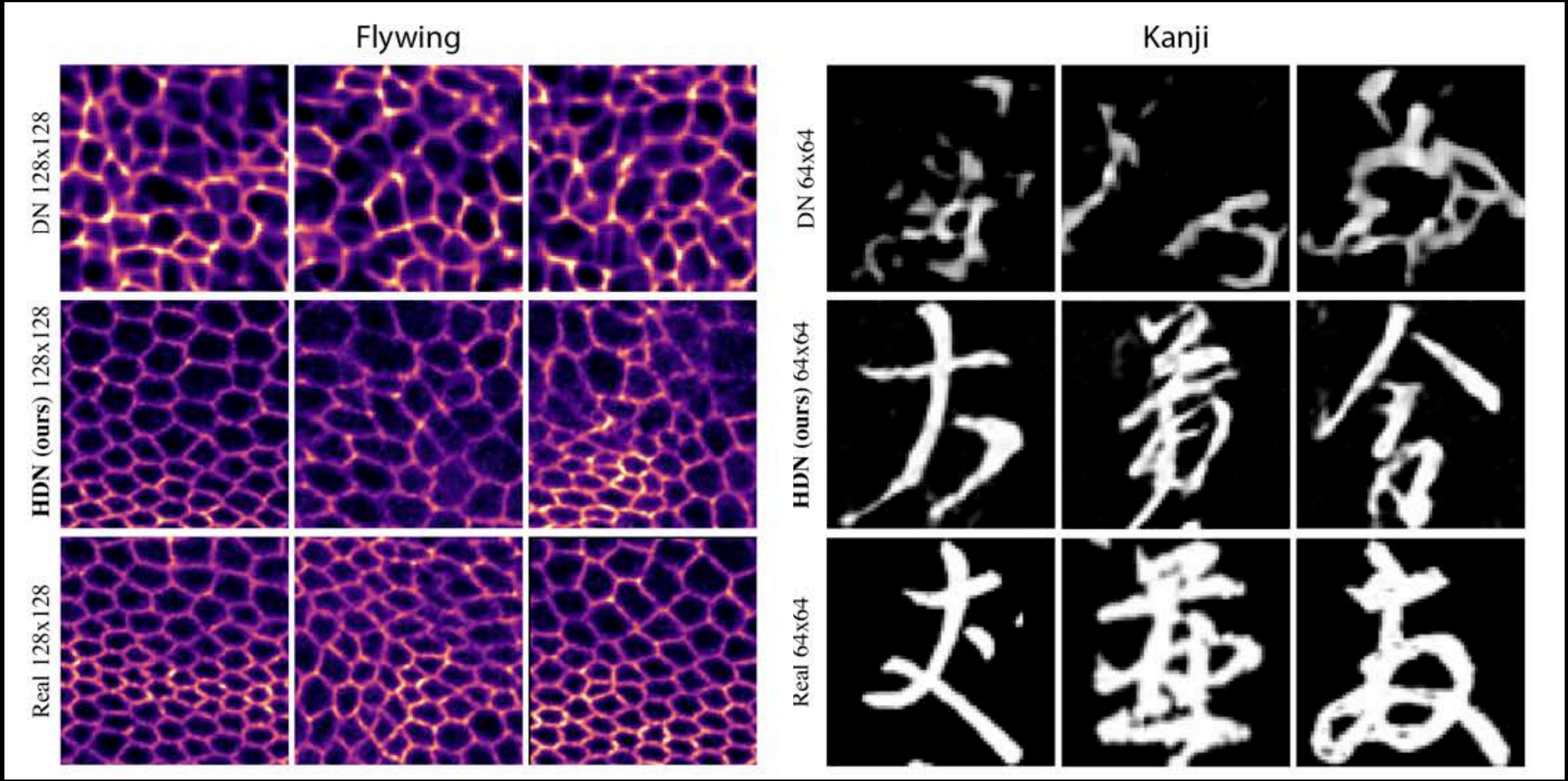
Prakash et al., Interpretable Unsupervised Diversity Denoising and Artefact Removal, *submitted (OpenReviews)*



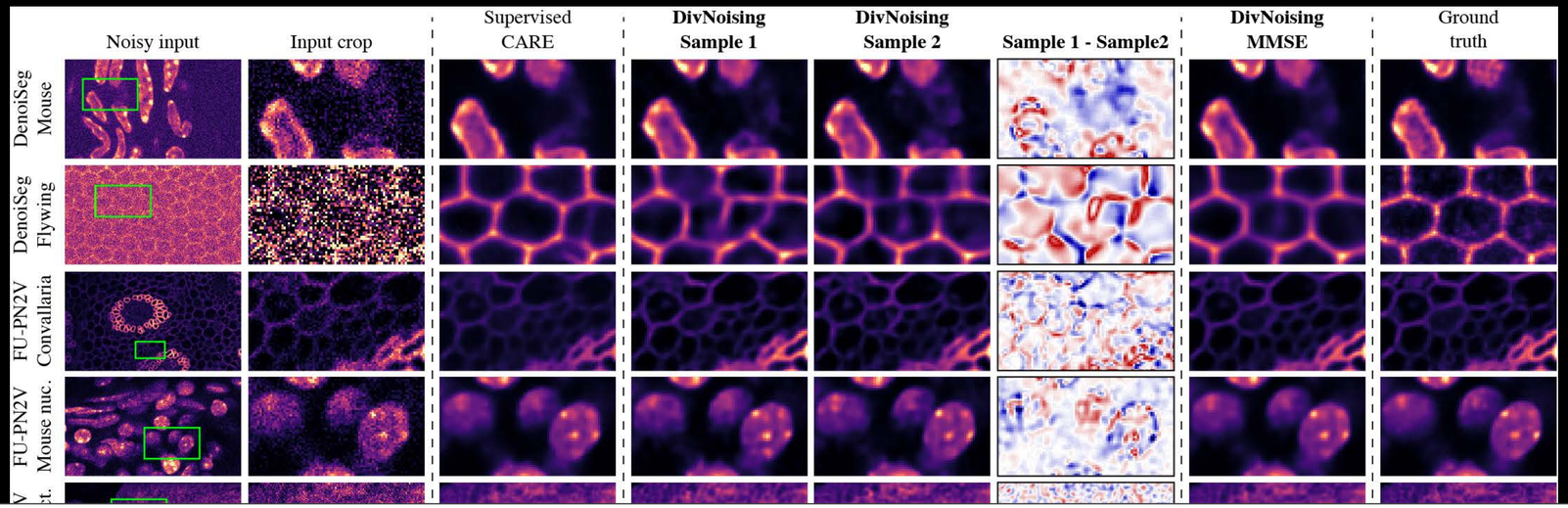
Prakash et al., Interpretable Unsupervised Diversity Denoising and Artefact Removal, *submitted (OpenReviews)*



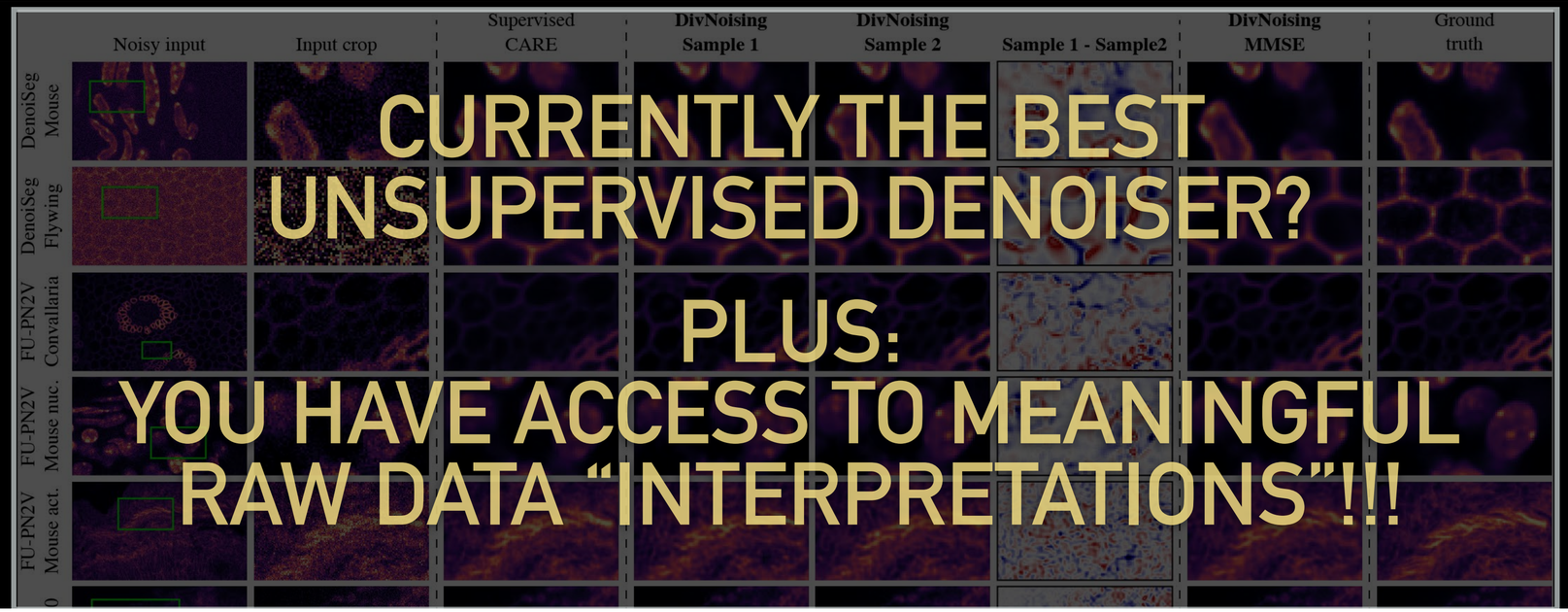
Prakash et al., Interpretable Unsupervised Diversity Denoising and Artefact Removal, *submitted (OpenReviews)*



Prakash et al., Interpretable Unsupervised Diversity Denoising and Artefact Removal, *submitted (OpenReviews)*



	Dataset	Unsupervised N2V	Unsupervised PN2V	Unsupervised DivNoising (MMSE)	Supervised CARE
DenoiseSeg	Mouse	33.55	**34.19**	34.03	34.63
	Flywing	22.85	24.85	**25.10**	25.60
FU-PN2V	Convallaria	35.73	36.47	**36.94**	36.71
	Convallaria (bootstr.)	35.73	**36.70**	36.63	36.71
	Mouse Actin	33.39	33.86	**33.98**	34.20
	Mouse Nuclei	35.84	**36.35**	36.31	36.58
W2S	Ch. 0 (avg1)	**34.36**	-	34.13	34.30
	Ch. 1 (avg1)	31.96	-	**32.28**	32.11
	Ch. 2 (avg1)	34.78	32.48	**35.18**	34.73
	Ch. 0 (avg16)	38.83	39.19	**39.62**	41.94
	Ch. 1 (avg16)	37.80	38.24	**38.37**	39.09
	Ch. 2 (avg16)	40.17	40.49	**40.52**	40.88



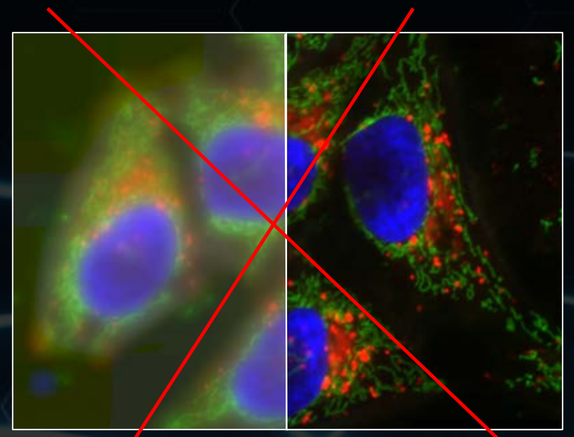
CURRENTLY THE BEST UNSUPERVISED DENOISER?

PLUS: YOU HAVE ACCESS TO MEANINGFUL RAW DATA "INTERPRETATIONS"!!!

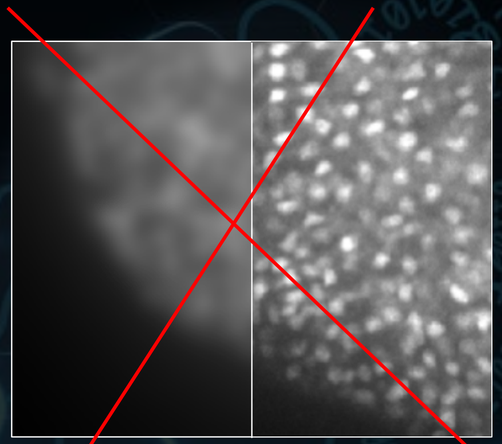
	Non DL	Single-image DL		Multi-image DL			Diversity DL			Supervised	
Dataset	**BM3D**	**DIP**	**S2S**	**N2V**	**N2Same**	**PN2V**	**HVAE**	**DN**	**HDN**	**N2N**	**CARE**
Convallaria	35.45	-	-	35.73	36.46	36.47	34.11	36.90	**37.39**	36.85	36.71
Confocal Mice	37.95	-	-	37.56	37.96	38.17	31.62	37.82	**38.28**	38.19	38.39
2 Photon Mice	33.81	-	-	33.42	33.13	33.67	25.96	33.61	**34.00**	34.33	34.35
Mouse Actin	33.64	-	-	33.39	32.55	33.86	27.24	33.99	**34.12**	34.60	34.20
Mouse Nuclei	36.20	-	-	35.84	36.20	36.35	32.62	36.26	**36.87**	37.33	36.58
Flywing	23.45	24.67	-	24.79	22.81	24.85	25.33	25.02	**25.59**	25.67	25.79
BSD68	28.56	27.96	28.61	27.70	27.95	28.46	27.20	27.42	**28.82**	28.86	29.07
Set12	29.94	28.60	29.51	28.92	29.35	29.61	27.81	28.24	**29.95**	30.04	30.36
BioID Faces	33.91	-	-	32.34	34.05	33.76	31.65	33.12	**34.59**	35.04	35.06
CelebA HQ	33.28	-	-	30.80	31.82	33.01	31.45	31.41	**33.54**	33.39	33.57
Kanji	20.45	-	-	19.95	20.28	19.40	20.44	19.47	**20.72**	20.56	20.64
MNIST	15.82	-	-	19.04	18.79	13.87	20.52	19.06	**20.87**	20.29	20.43

One more thing...

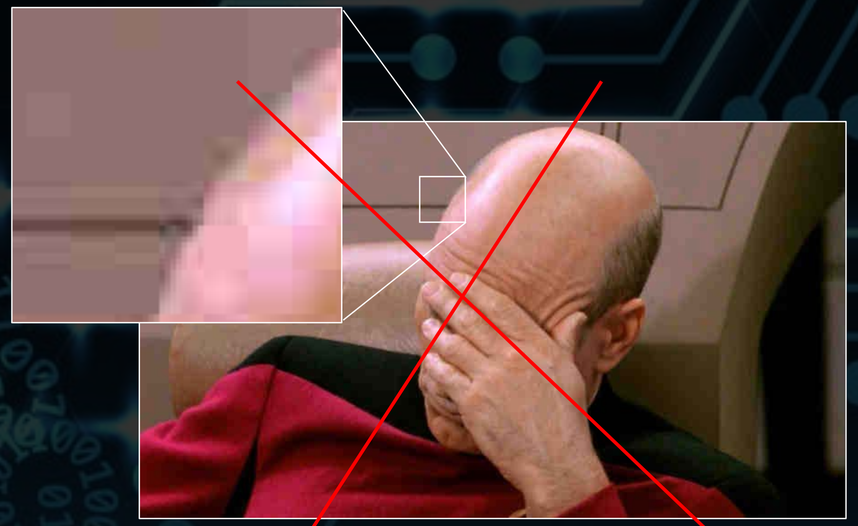
LIMITATIONS OF NOISE2VOID?



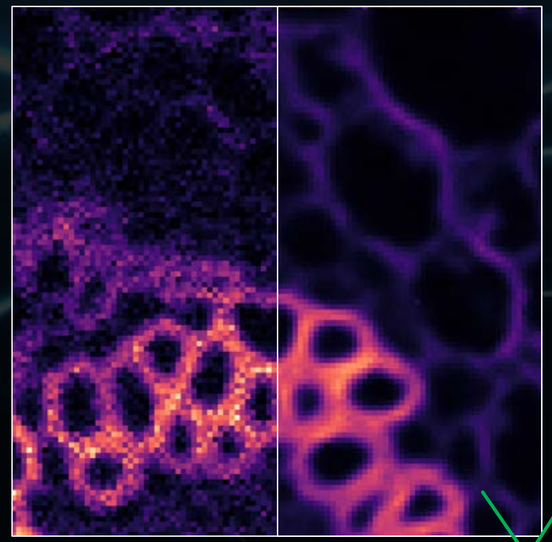
Computational clearing



Deconvolution



JPEG compression artifacts



per-pixel noise

- Low light images
- Shot noise
- Readout noise
- Non-structured noise

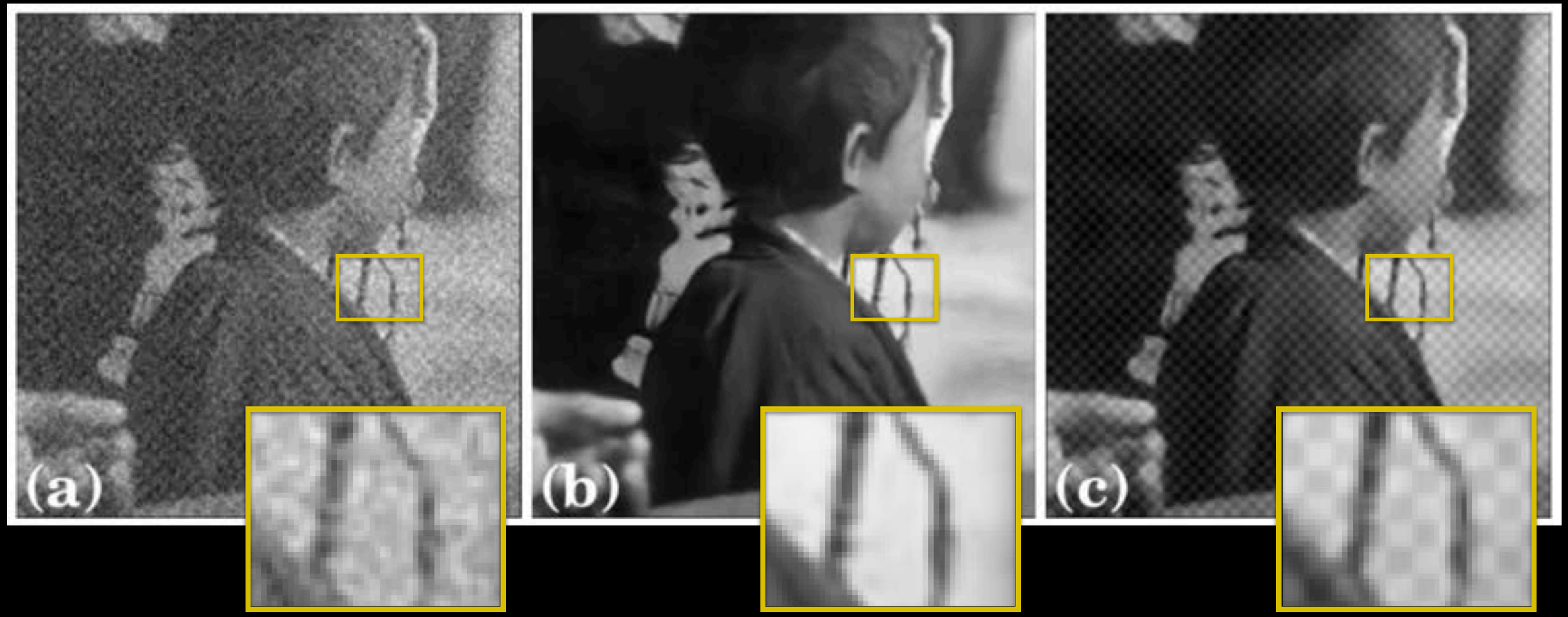
Take home message:

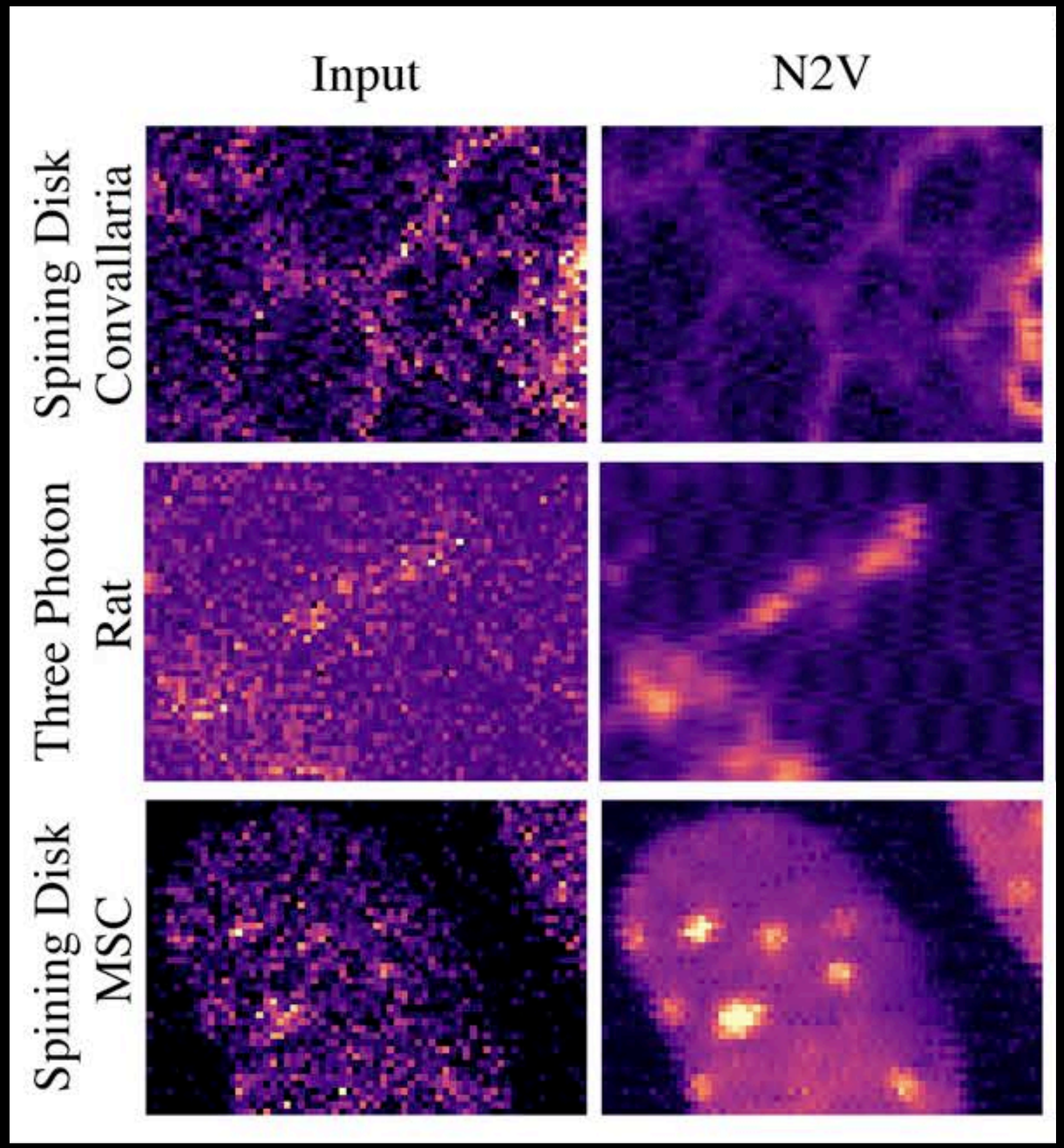
Noise2Void is as easy as it gets if reducing per-pixel noise is the goal!

INPUT

TRADITIONAL CARE

NOISE2VOID

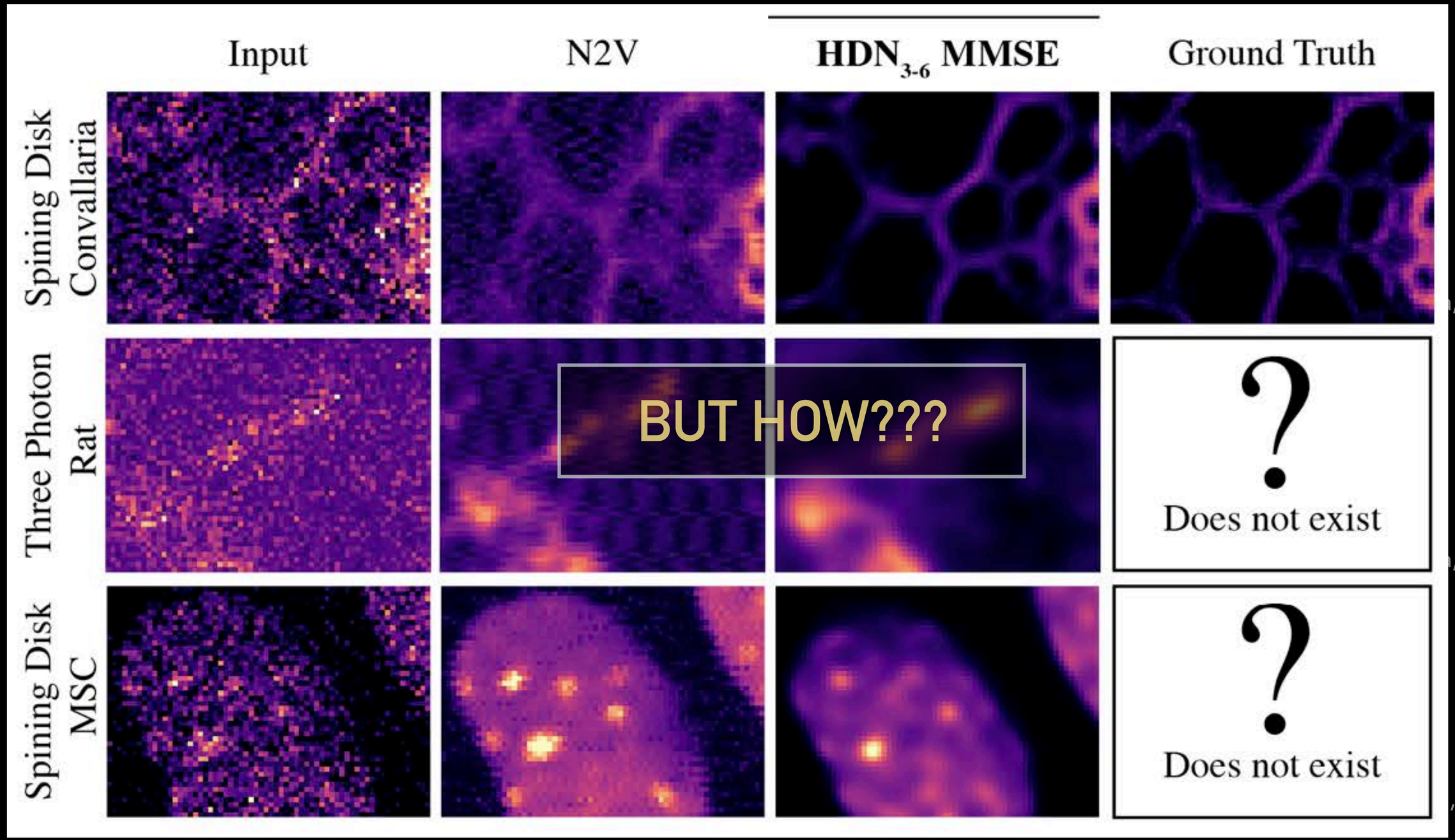




>> data, **Myers Lab**, MPI-CBG

>> data, **Prevedel Lab**, EMBL-HD

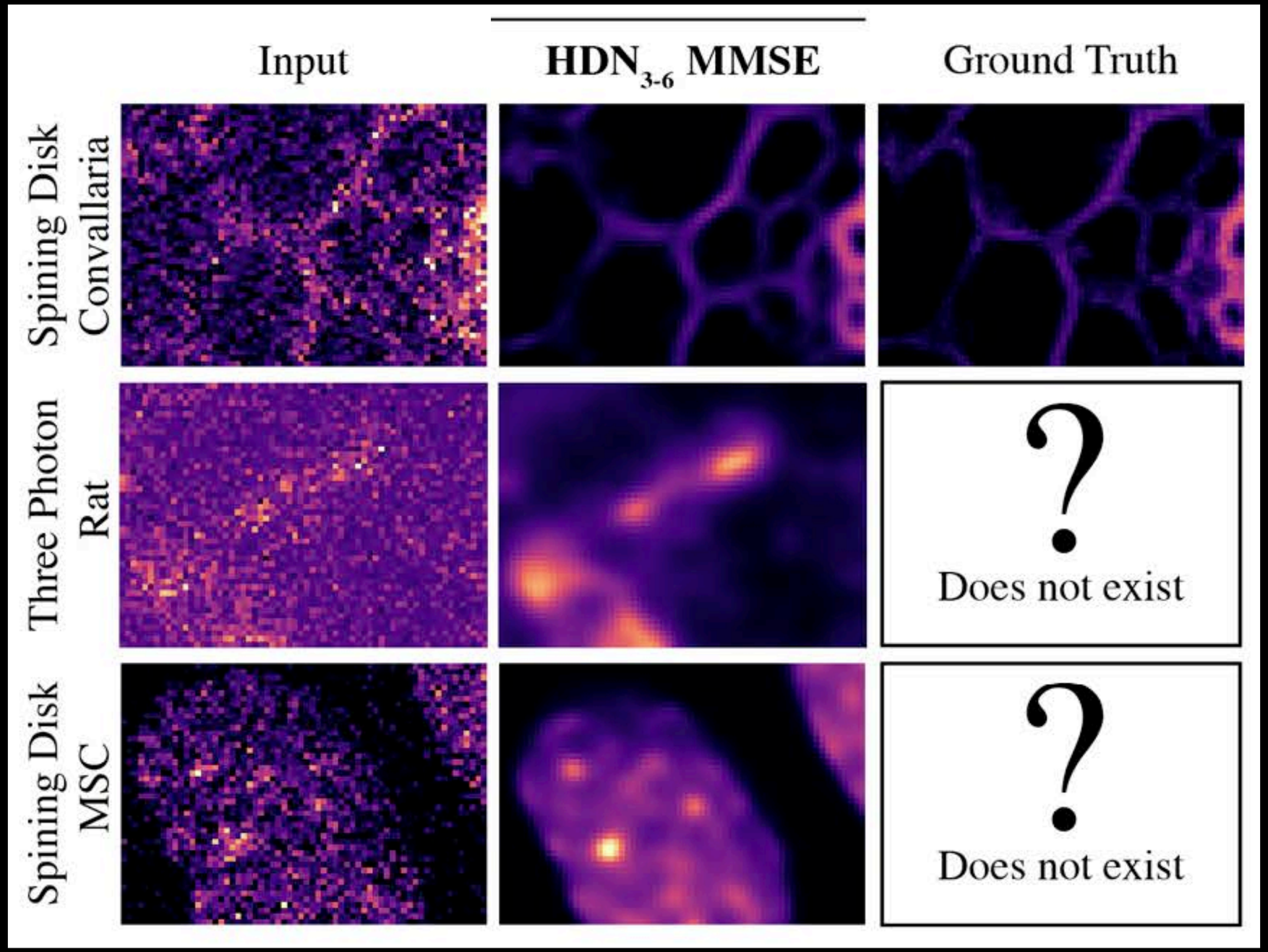
>> data, **Tabler Lab**, MPI-CBG

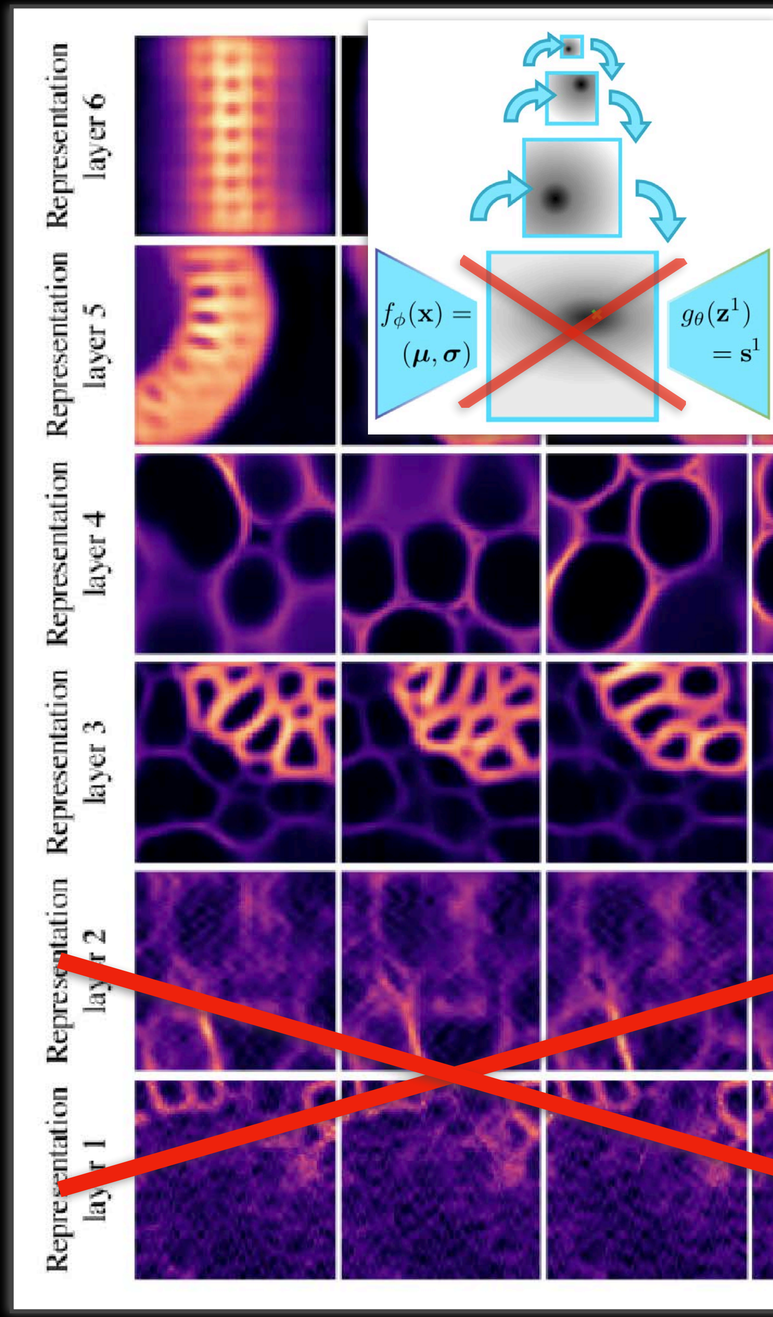


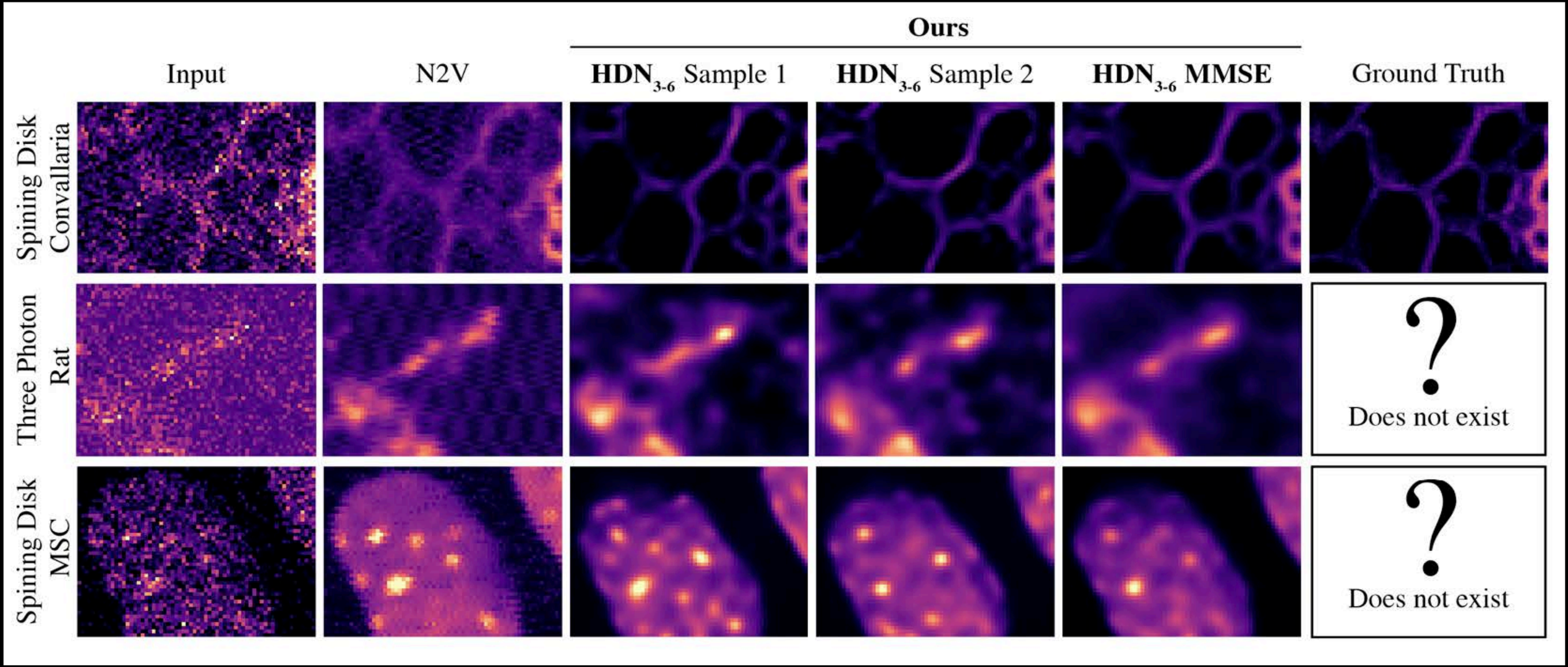
, **Jug Lab**, belie

, **Prevedel Lab**

, **Tabler Lab**, M


Input
HDN3-6 MMSE
Ground Truth
Spining Disk Convallaria
Three Photon Rat
Spining Disk MSC
?
Does not exist
?
Does not exist


Representation layer 6
Representation layer 5
Representation layer 4
Representation layer 3
Representation layer 2
Representation layer 1
$f_\phi(\mathbf{x}) = (\boldsymbol{\mu}, \boldsymbol{\sigma})$
$g_\theta(\mathbf{z}^1) = \mathbf{s}^1$



Prakash et al., Interpretable Unsupervised Diversity Denoising and Artefact Removal, *submitted (OpenReviews)*

WHEN CAN YOU USE DIVNOISING?

1. FOR PIXEL-NOISE REMOVAL: ALWAYS
2. FOR STRUCTURED-NOISE REMOVAL:

   WHENEVER THE PIXEL RESOLUTION IS HIGHER THAN THE OPTICAL RESOLUTION AND REMOVING LOWEST LATENT LAYERS DOES NOT REMOVE REAL STRUCTURES…

1. CONTENT-AWARE RESTORATION
2. DIVERSITY DENOISING
3. SHOULD YOU USE DL METHODS?
4. WHAT WILL THE FUTURE BRING?

I WILL CUT A LONG STORY SHORT…

1. IF YOU DON'T NEED IT, DON'T USE IT!
2. IF YOU NEED IT, READ THE PAPER BY LAINE ET AL.!
3. BE SURE TO ENABLE OTHERS TO REPRODUCE YOUR RESULTS BY…
   - PROVIDE (SOME) OF YOUR DATA
   - OPEN YOU ANALYSIS PIPELINE!

COMMENT | FOCUS

Check for updates

Avoiding a replication crisis in deep-learning-based bioimage analysis

Deep learning algorithms are powerful tools for analyzing, restoring and transforming bioimaging data. One promise of deep learning is parameter-free one-click image analysis with expert-level performance in a fraction of the time previously required. However, as with most emerging technologies, the potential for inappropriate use is raising concerns among the research community. In this Comment, we discuss key concepts that we believe are important for researchers to consider when using deep learning for their microscopy studies. We describe how results obtained using deep learning can be validated and propose what should, in our view, be considered when choosing a suitable tool. We also suggest what aspects of a deep learning analysis should be reported in publications to ensure reproducibility. We hope this perspective will foster further discussion among developers, image analysis specialists, users and journal editors to define adequate guidelines and ensure the appropriate use of this transformative technology.

Romain F. Laine, Ignacio Arganda-Carreras, Ricardo Henriques and Guillaume Jacquemet

Microscopy is a leading technology used to gain fundamental insight for biological research. Today, a typical microscopy session may generate hundreds to thousands of images, generally requiring computational analysis to extract meaningful results. Over the last few years, deep learning (DL) has increasingly become one of the gold standards for high-performance microscopy image analysis[1,2]. DL has shown the capacity to

sequences of operations, it becomes difficult to understand what features of the original images are used. For that reason, they are often thought of as 'black boxes' as, for most users, only the input images and output predictions are readily available.

The training data provided to the ANN commonly consist of a large set of representative input images and their expected results. For instance, in denoising, the training dataset comprises matched

analysis and the level of computational skills required to run them.

In this Comment, we suggest a set of best practices for implementing and reporting on the use and development of DL image analysis tools. These suggestions are primarily built from our own experiences in developing and using DL tools. Here we cover traditional image analysis tasks, such as cell segmentation, as well as more recent approaches for image

sequential mathematical operations on the input data. As we inspect deeper into these

depends on what tasks or combination of tasks need to be performed, the scale of the

inadequate for the desired task, the resulting model will produce unwanted results

Cite this article

Laine, R.F., Arganda-Carreras, I., Henriques, R. *et al.* Avoiding a replication crisis in deep-learning-based bioimage analysis. *Nat Methods* **18,** 1136–1144 (2021). https://doi.org/10.1038/s41592-021-01284-3

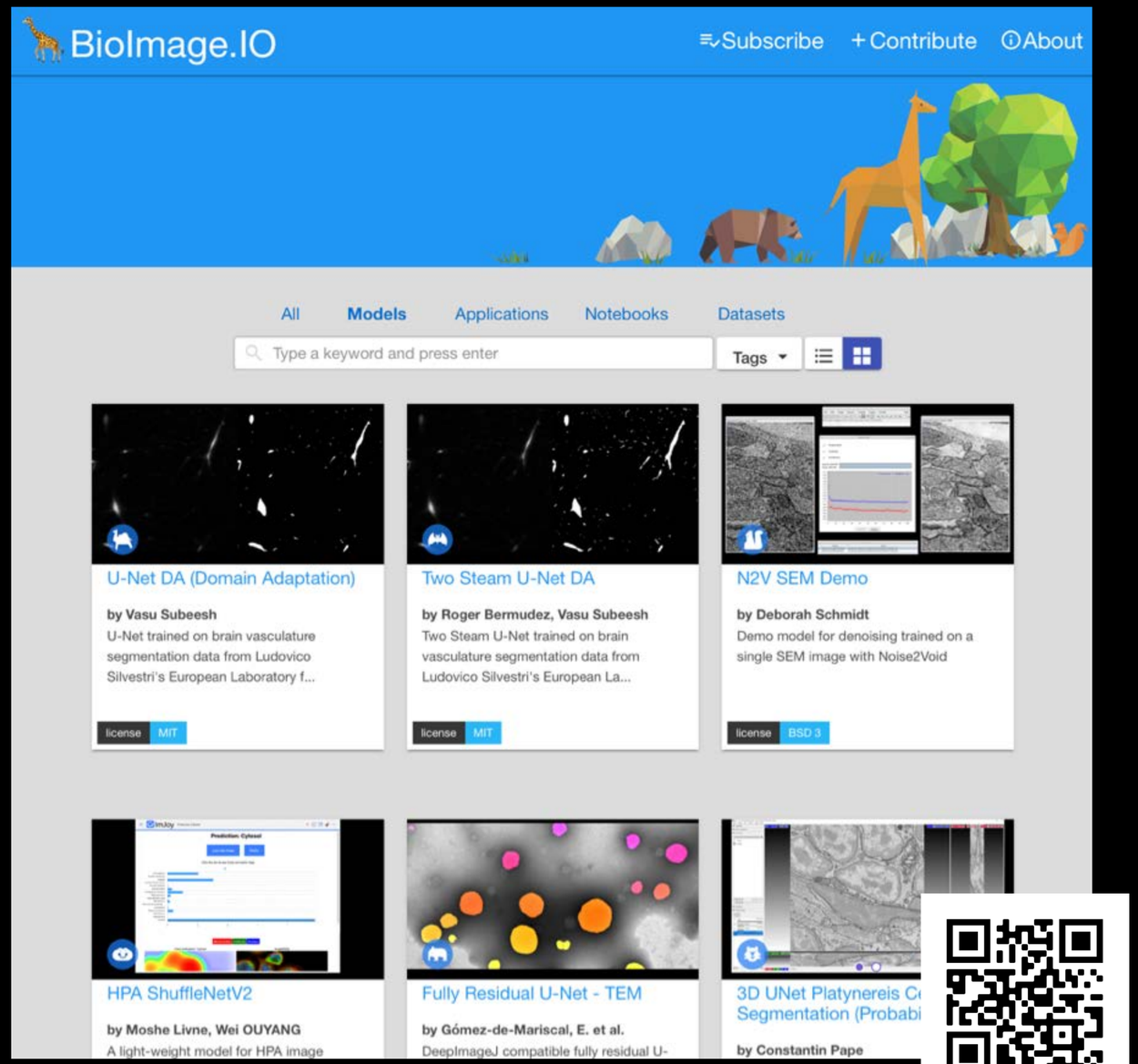
Download citation

Published	Issue Date
04 October 2021	October 2021

1. CONTENT-AWARE RESTORATION
2. DIVERSITY DENOISING
3. SHOULD YOU USE DL METHODS?
4. WHAT WILL THE FUTURE BRING?

WHO KNOWS… I GUESS ONLY TIME WILL TELL…

…BUT THERE IS ONE THING I KNOW FOR SURE…


BioImage.IO
Subscribe
+ Contribute
About
All
Models
Applications
Notebooks
Datasets
Type a keyword and press enter
Tags
U-Net DA (Domain Adaptation)
by Vasu Subeesh
U-Net trained on brain vasculature segmentation data from Ludovico Silvestri's European Laboratory f...
license MIT
Two Steam U-Net DA
by Roger Bermudez, Vasu Subeesh
Two Steam U-Net trained on brain vasculature segmentation data from Ludovico Silvestri's European La...
license MIT
N2V SEM Demo
by Deborah Schmidt
Demo model for denoising trained on a single SEM image with Noise2Void
license BSD 3
HPA ShuffleNetV2
by Moshe Livne, Wei OUYANG
A light-weight model for HPA image
Fully Residual U-Net - TEM
by Gómez-de-Mariscal, E. et al.
DeepImageJ compatible fully residual U-
3D UNet Platynereis C
Segmentation (Probabi
by Constantin Pape

Standards; Easy of use; Reproducibility; FAIR; etc.

Everyone is Welcome! Join us!!!

Alex Krull
Alumnus Postdoc

Tim-Oliver Buchholz
Alumnus PhD st.

THANKS – DANKE – GRAZIE – MERCI – GRACIAS – DĚKUJI – ASANTE – HVALA

Mangal Prakash
Alumnus PhD st.

Tobias Pietzsch
Superhero

Anna Goncharova
PhD Student

Deborah Schmidt
Alumnus RSE

Christopher Schmied
Bioimage Analyst

Tom Burke
RSE

Matthias Arzt
RSE

Uwe Schmidt
Freigeist

Joran Deschamps
RSE

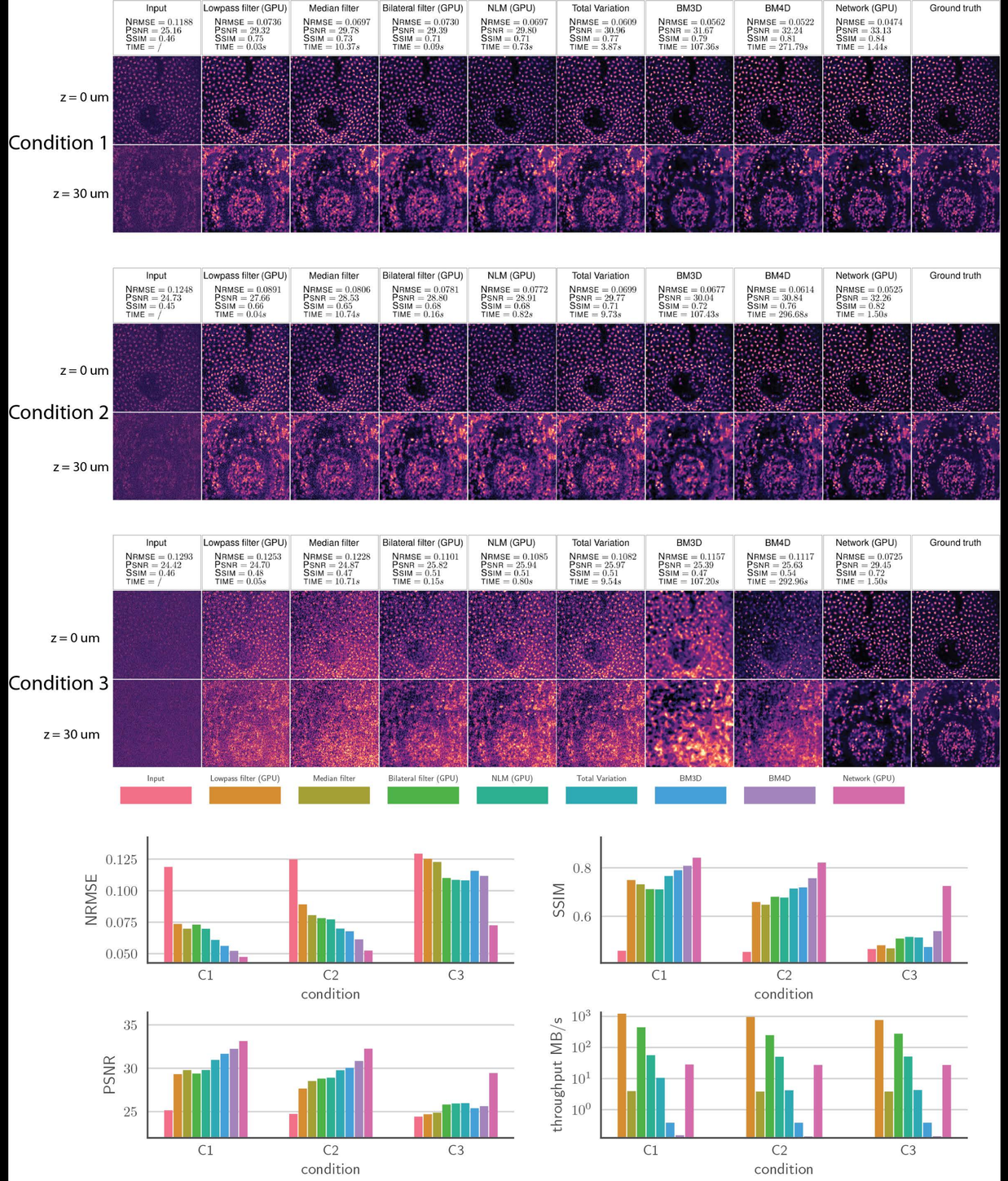
Nuno Pimpao M.
PhD Student

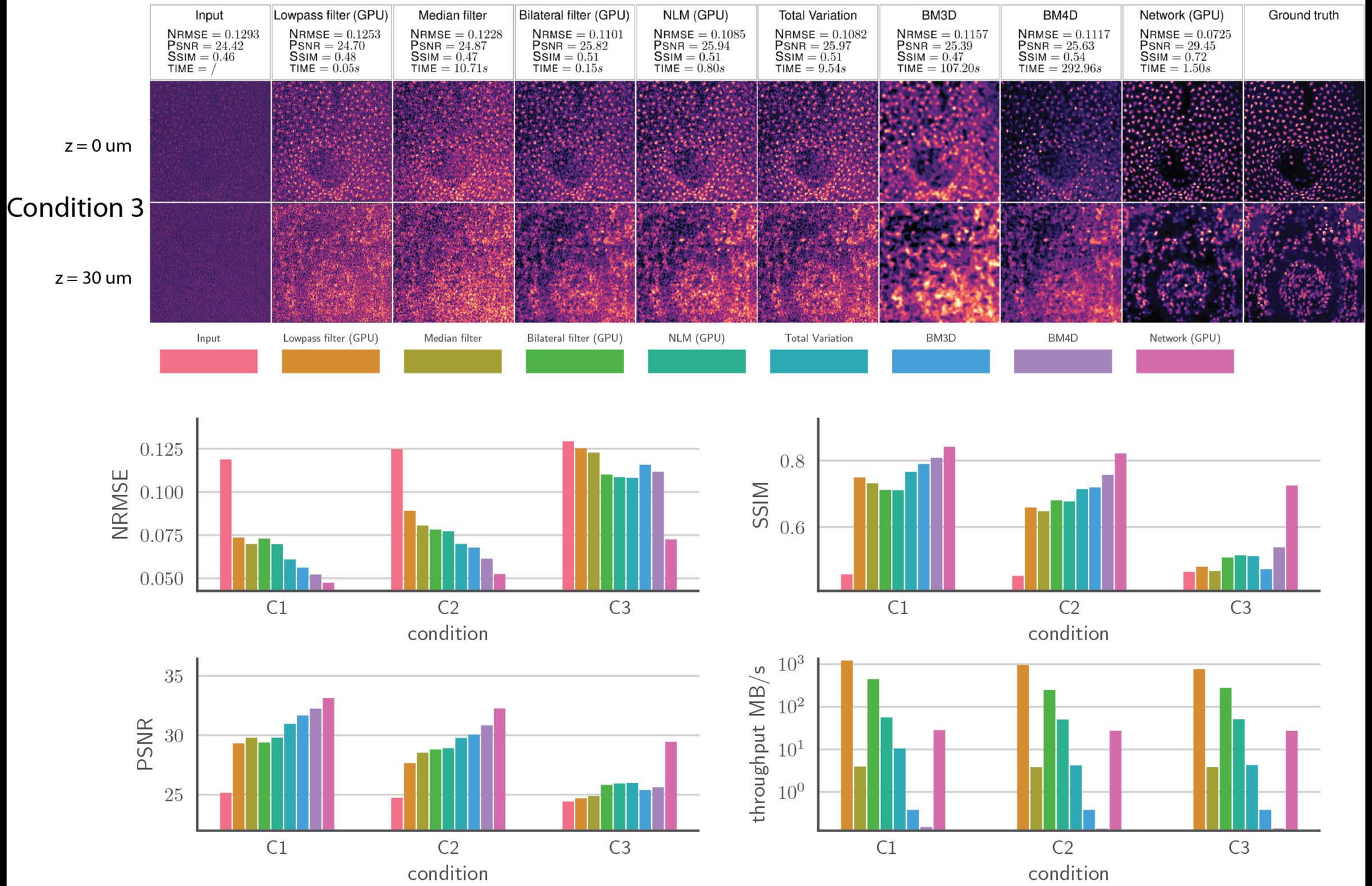
Manan Lalit
PhD Student

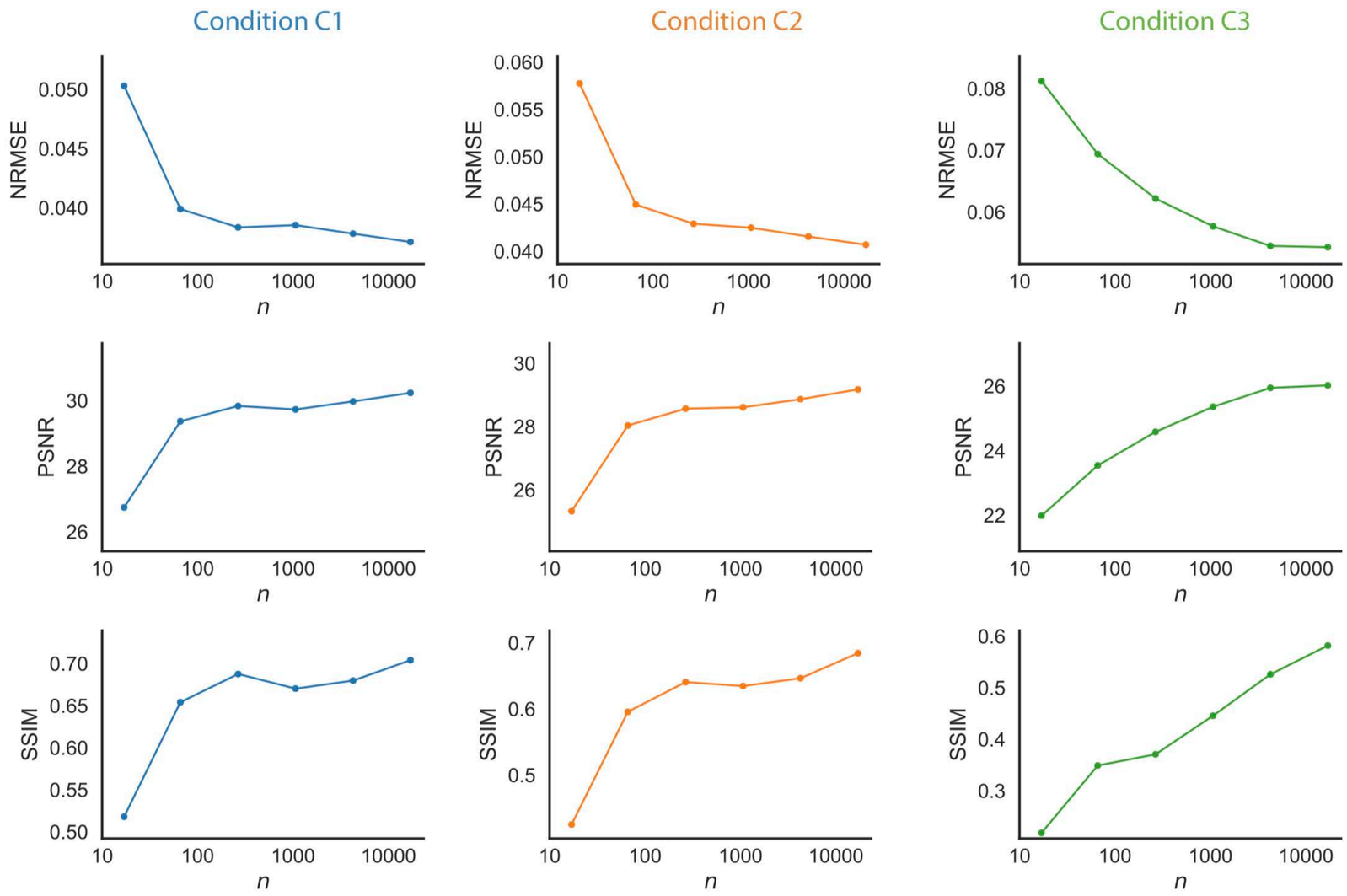
Martin Weigert
Wunderkind

Model Zoo Collab-orators

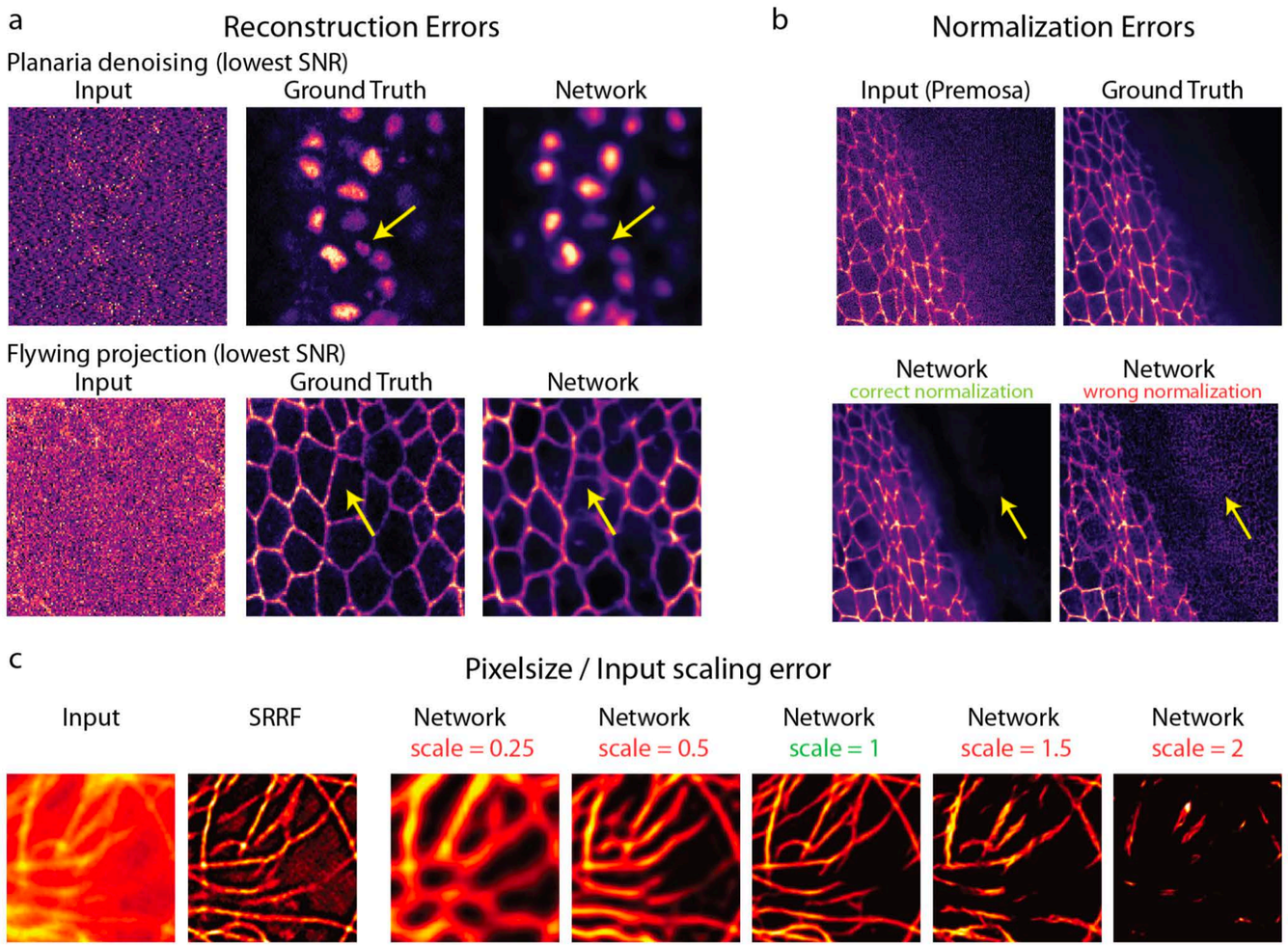
Euro-BioImaging ERIC
Henriques Lab (IGC)
Jug Lab (CSBD/HT)
Kreshuk Lab (EBML-HD)
Lundberg Lab (KTH)
deepImageJ Team
BioImage Archive (EMBL EBI)
Partner LS Ris


Input
NRMSE = 0.1188
PSNR = 25.16
SSIM = 0.46
TIME = /
Lowpass filter (GPU)
NRMSE = 0.0736
PSNR = 29.32
SSIM = 0.75
TIME = 0.03s
Median filter
NRMSE = 0.0697
PSNR = 29.78
SSIM = 0.73
TIME = 10.37s
Bilateral filter (GPU)
NRMSE = 0.0730
PSNR = 29.39
SSIM = 0.71
TIME = 0.09s
NLM (GPU)
NRMSE = 0.0697
PSNR = 29.80
SSIM = 0.71
TIME = 0.73s
Total Variation
NRMSE = 0.0609
PSNR = 30.96
SSIM = 0.77
TIME = 3.87s
BM3D
NRMSE = 0.0562
PSNR = 31.67
SSIM = 0.79
TIME = 107.36s
BM4D
NRMSE = 0.0522
PSNR = 32.24
SSIM = 0.81
TIME = 271.79s
Network (GPU)
NRMSE = 0.0474
PSNR = 33.13
SSIM = 0.84
TIME = 1.44s
Ground truth
z = 0 um
Condition 1
z = 30 um
Input
NRMSE = 0.1248
PSNR = 24.73
SSIM = 0.45
TIME = /
Lowpass filter (GPU)
NRMSE = 0.0891
PSNR = 27.66
SSIM = 0.66
TIME = 0.04s
Median filter
NRMSE = 0.0806
PSNR = 28.53
SSIM = 0.65
TIME = 10.74s
Bilateral filter (GPU)
NRMSE = 0.0781
PSNR = 28.80
SSIM = 0.68
TIME = 0.16s
NLM (GPU)
NRMSE = 0.0772
PSNR = 28.91
SSIM = 0.68
TIME = 0.82s
Total Variation
NRMSE = 0.0699
PSNR = 29.77
SSIM = 0.71
TIME = 9.73s
BM3D
NRMSE = 0.0677
PSNR = 30.04
SSIM = 0.72
TIME = 107.43s
BM4D
NRMSE = 0.0614
PSNR = 30.84
SSIM = 0.76
TIME = 296.68s
Network (GPU)
NRMSE = 0.0525
PSNR = 32.26
SSIM = 0.82
TIME = 1.50s
Ground truth
z = 0 um
Condition 2
z = 30 um
Input
NRMSE = 0.1293
PSNR = 24.42
SSIM = 0.46
TIME = /
Lowpass filter (GPU)
NRMSE = 0.1253
PSNR = 24.70
SSIM = 0.48
TIME = 0.05s
Median filter
NRMSE = 0.1228
PSNR = 24.87
SSIM = 0.47
TIME = 10.71s
Bilateral filter (GPU)
NRMSE = 0.1101
PSNR = 25.82
SSIM = 0.51
TIME = 0.15s
NLM (GPU)
NRMSE = 0.1085
PSNR = 25.94
SSIM = 0.51
TIME = 0.80s
Total Variation
NRMSE = 0.1082
PSNR = 25.97
SSIM = 0.51
TIME = 9.54s
BM3D
NRMSE = 0.1157
PSNR = 25.39
SSIM = 0.47
TIME = 107.20s
BM4D
NRMSE = 0.1117
PSNR = 25.63
SSIM = 0.54
TIME = 292.96s
Network (GPU)
NRMSE = 0.0725
PSNR = 29.45
SSIM = 0.72
TIME = 1.50s
Ground truth
z = 0 um
Condition 3
z = 30 um
Input
Lowpass filter (GPU)
Median filter
Bilateral filter (GPU)
NLM (GPU)
Total Variation
BM3D
BM4D
Network (GPU)
NRMSE
0.125
0.100
0.075
0.050
C1
C2
C3
condition
SSIM
0.8
0.6
C1
C2
C3
condition
PSNR
35
30
25
C1
C2
C3
condition
throughput MB/s
10^3
10^2
10^1
10^0
C1
C2
C3
condition


Input
NRMSE = 0.1293
PSNR = 24.42
SSIM = 0.46
TIME = /
Lowpass filter (GPU)
NRMSE = 0.1253
PSNR = 24.70
SSIM = 0.48
TIME = 0.05s
Median filter
NRMSE = 0.1228
PSNR = 24.87
SSIM = 0.47
TIME = 10.71s
Bilateral filter (GPU)
NRMSE = 0.1101
PSNR = 25.82
SSIM = 0.51
TIME = 0.15s
NLM (GPU)
NRMSE = 0.1085
PSNR = 25.94
SSIM = 0.51
TIME = 0.80s
Total Variation
NRMSE = 0.1082
PSNR = 25.97
SSIM = 0.51
TIME = 9.54s
BM3D
NRMSE = 0.1157
PSNR = 25.39
SSIM = 0.47
TIME = 107.20s
BM4D
NRMSE = 0.1117
PSNR = 25.63
SSIM = 0.54
TIME = 292.96s
Network (GPU)
NRMSE = 0.0725
PSNR = 29.45
SSIM = 0.72
TIME = 1.50s
Ground truth
z = 0 um
Condition 3
z = 30 um
Input
Lowpass filter (GPU)
Median filter
Bilateral filter (GPU)
NLM (GPU)
Total Variation
BM3D
BM4D
Network (GPU)
NRMSE
0.125
0.100
0.075
0.050
C1
C2
C3
condition
SSIM
0.8
0.6
C1
C2
C3
condition
PSNR
35
30
25
C1
C2
C3
condition
throughput MB/s
10^3
10^2
10^1
10^0
C1
C2
C3
condition



Additional Supplementary Figure 4: Dependency of restoration quality on amount of training data for denoising of Planaria (*Schmidtea mediterranea*) volumes. Shown are the test error (NRSME, SSIM, PSNR) of networks trained with $n \in (17, 66, 266, 1063, 4251, 17005)$ subvolumes of size $64 \times 64 \times 16$.



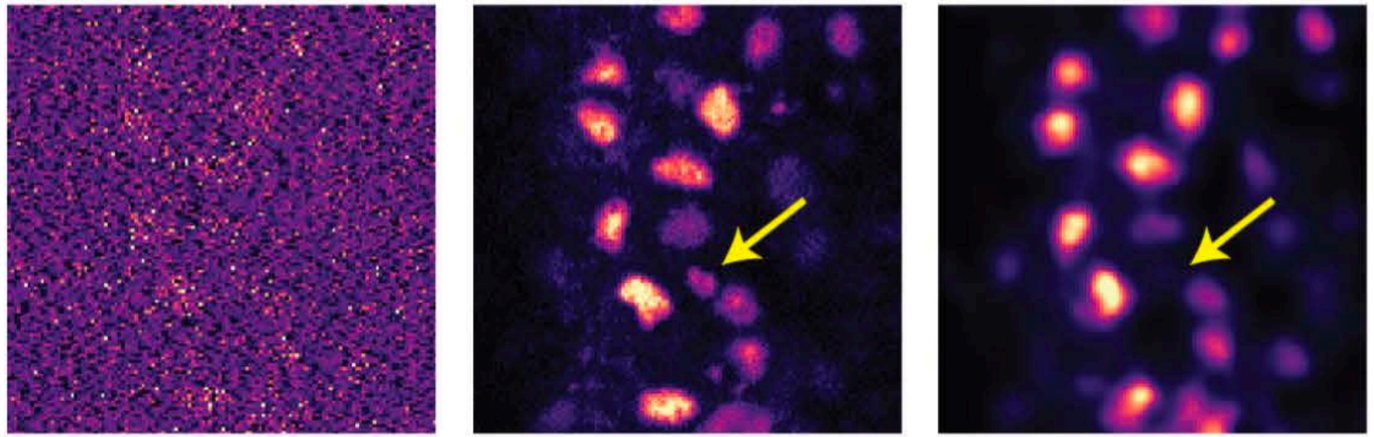
Additional Supplementary Figure 6: Failure cases of network reconstructions. a) Missing (top) or erroneously added (lower) structures for restorations of extremely challenging and noisy inputs. b) Restoration errors due to wrong normalization of the input at test time. Here, the network was trained with inputs being normalized with percentiles $(p_{min}, p_{max}) = (2\%/99.7\%)$, but evaluated with percentile normalization of $(p_{min}, p_{max}) = (1\%/30\%)$. c) Influence of wrong pixelsizes/scaling for tubulin restoration networks. Here, the network was trained with a specific assumed pixelsize and PSF width. Applying the network to wrongly scaled inputs results in erroneous results.

a

Reconstruction Errors

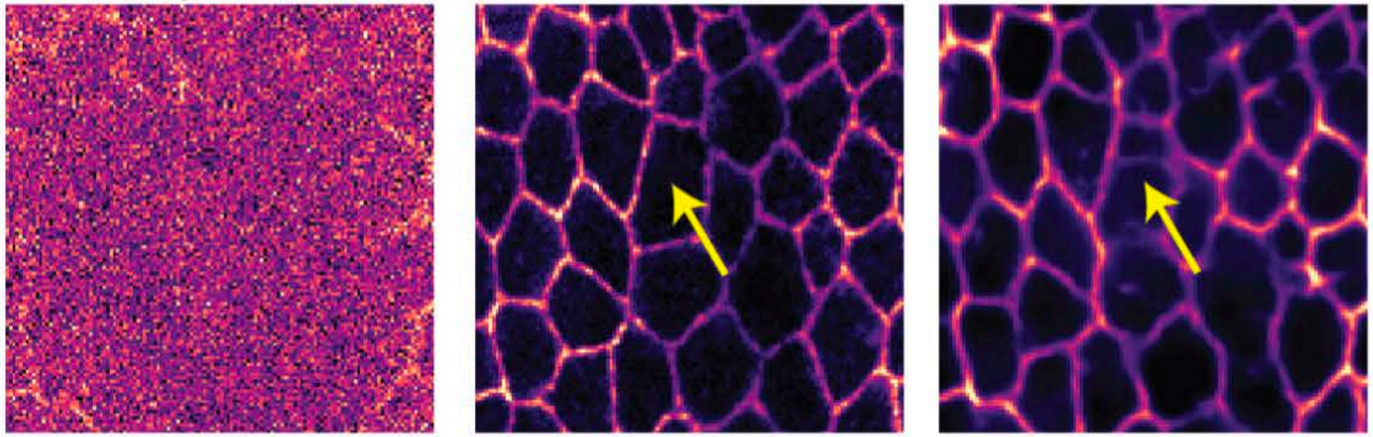
Planaria denoising (lowest SNR)

Input | Ground Truth | Network



Flywing projection (lowest SNR)

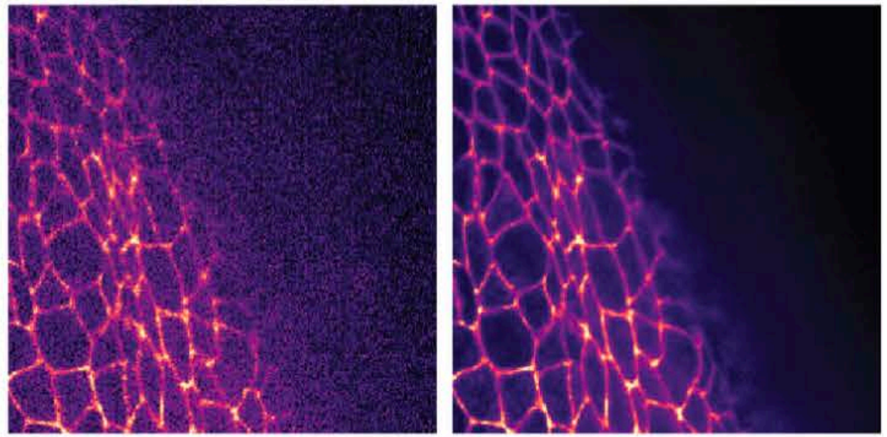
Input | Ground Truth | Network



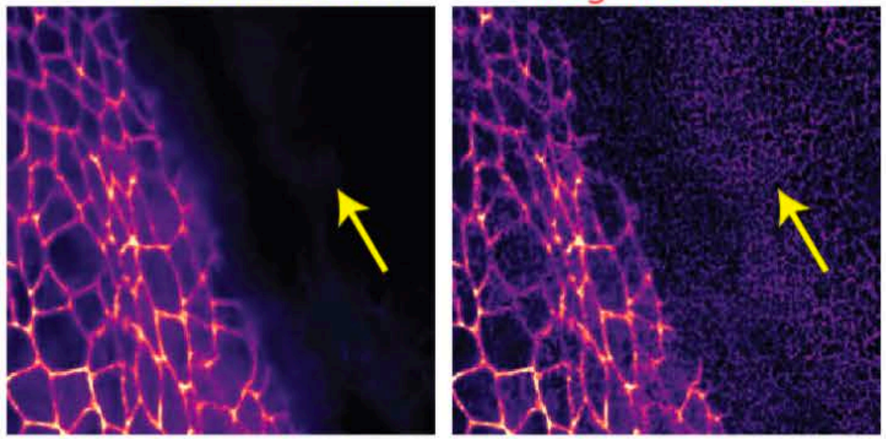
b

Normalization Errors

Input (Premosa) | Ground Truth

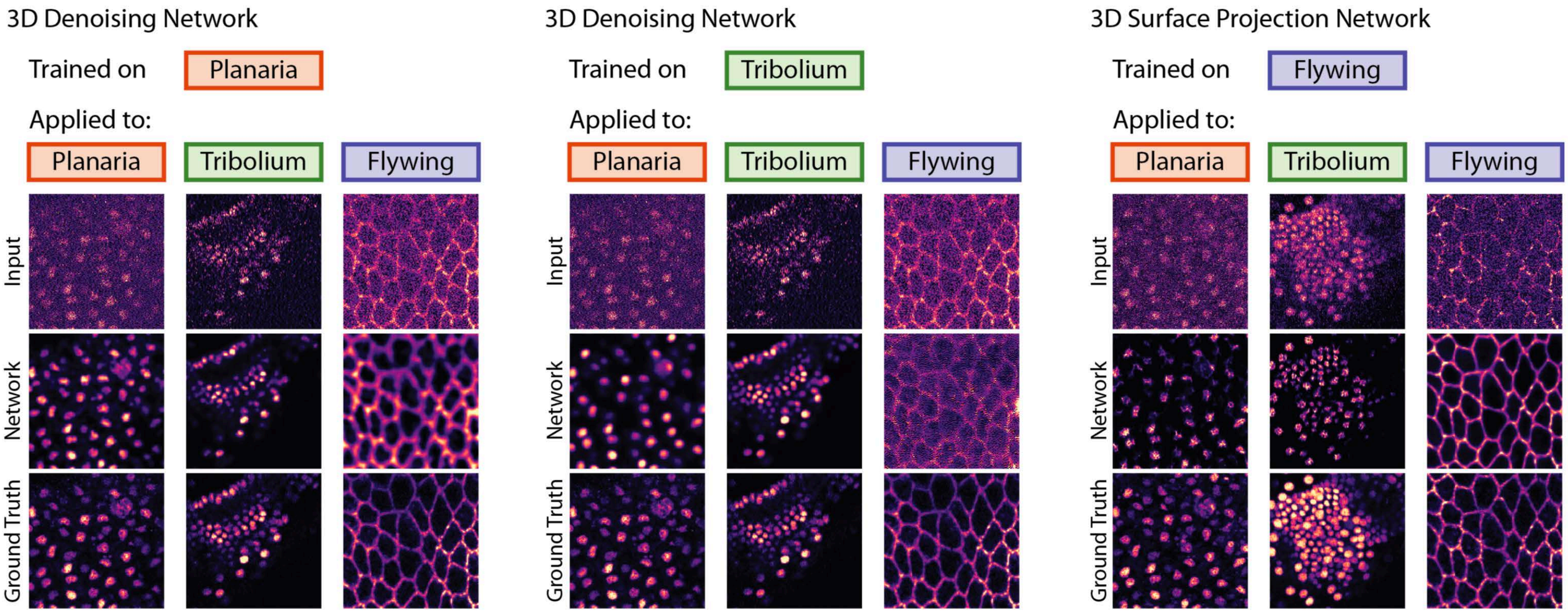


Network correct normalization | Network wrong normalization



c

Pixelsize / Input scaling error

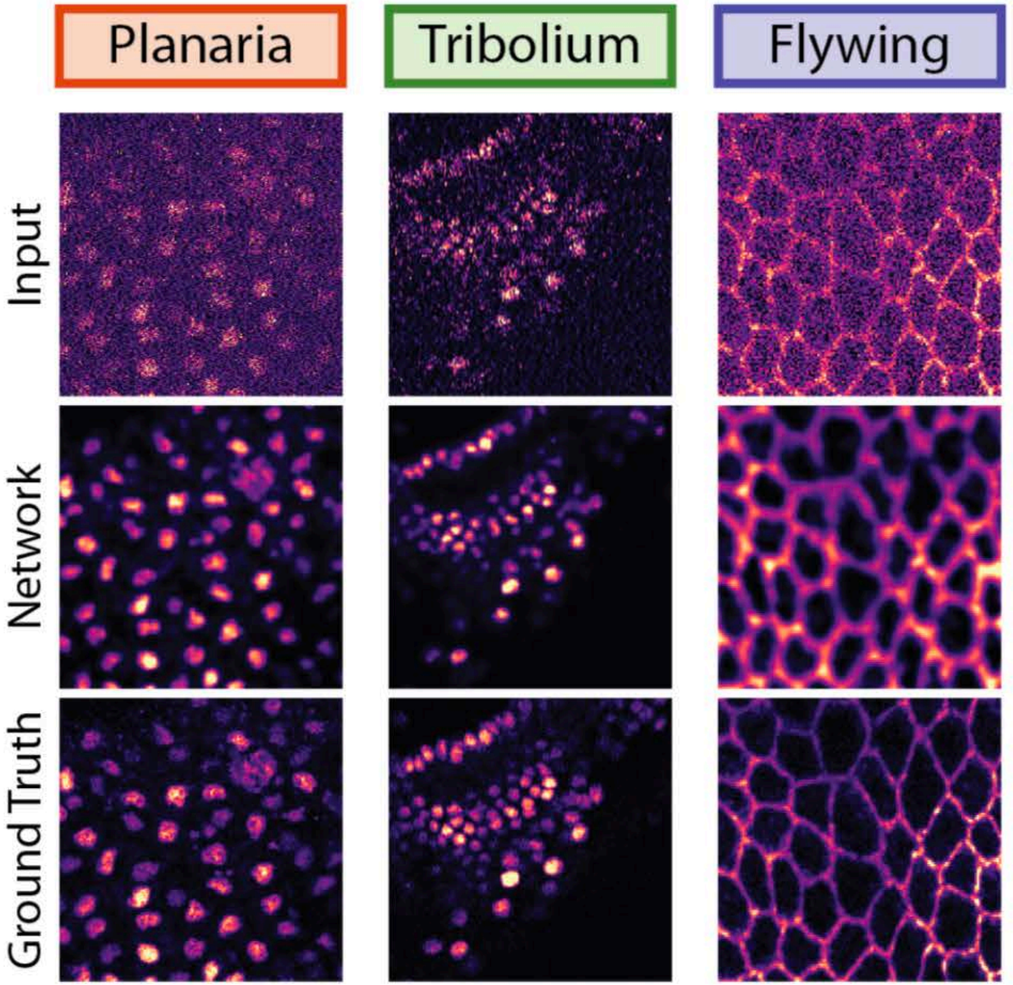


Additional Supplementary Figure 5: Cross-application of networks trained on a specific data set to dissimilar images of a different data set.

3D Denoising Network

Trained on Planaria

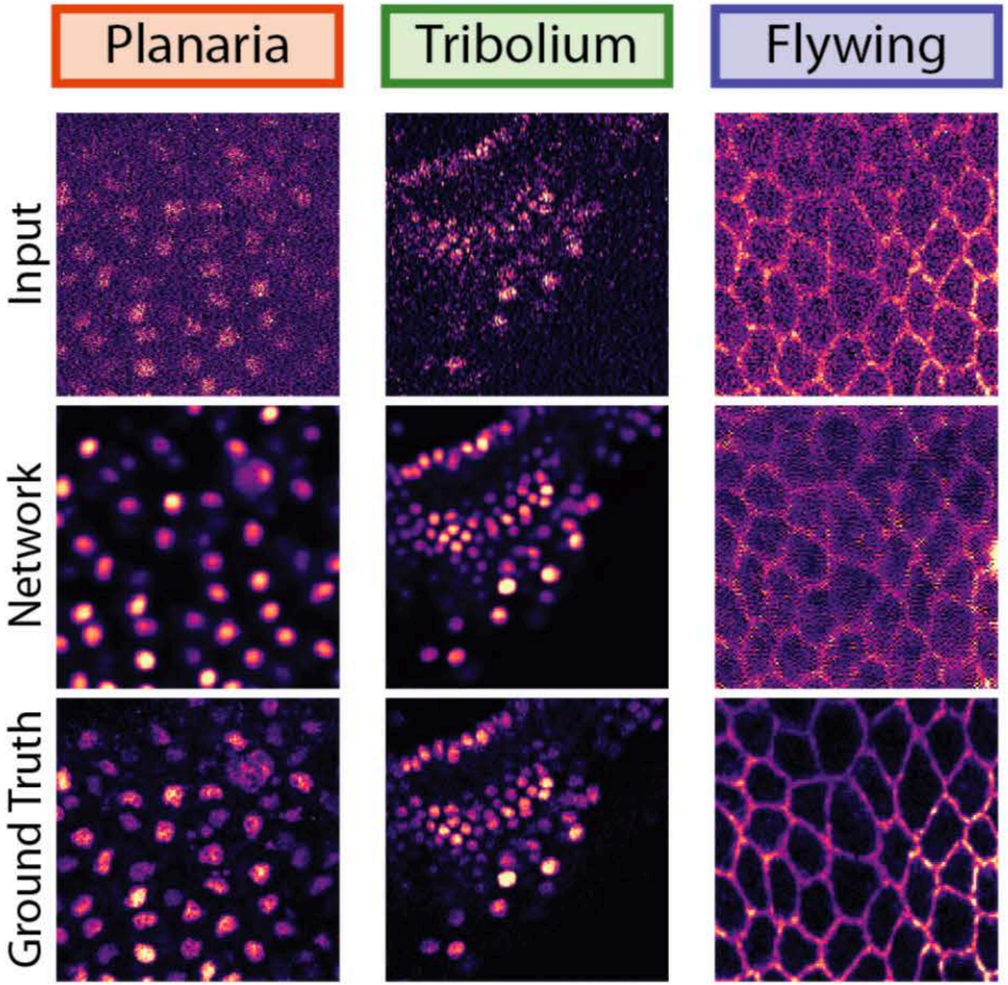
Applied to:



3D Denoising Network

Trained on Tribolium

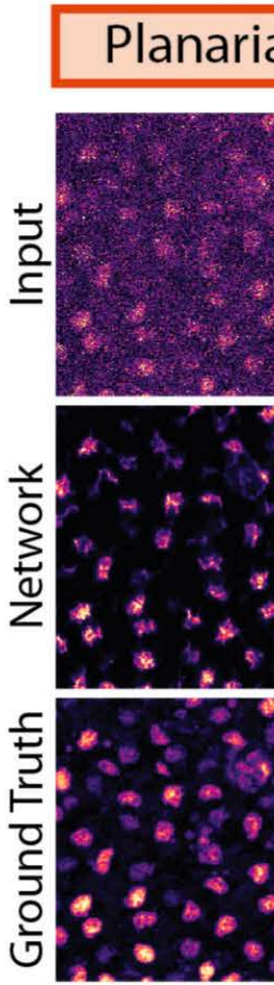
Applied to:

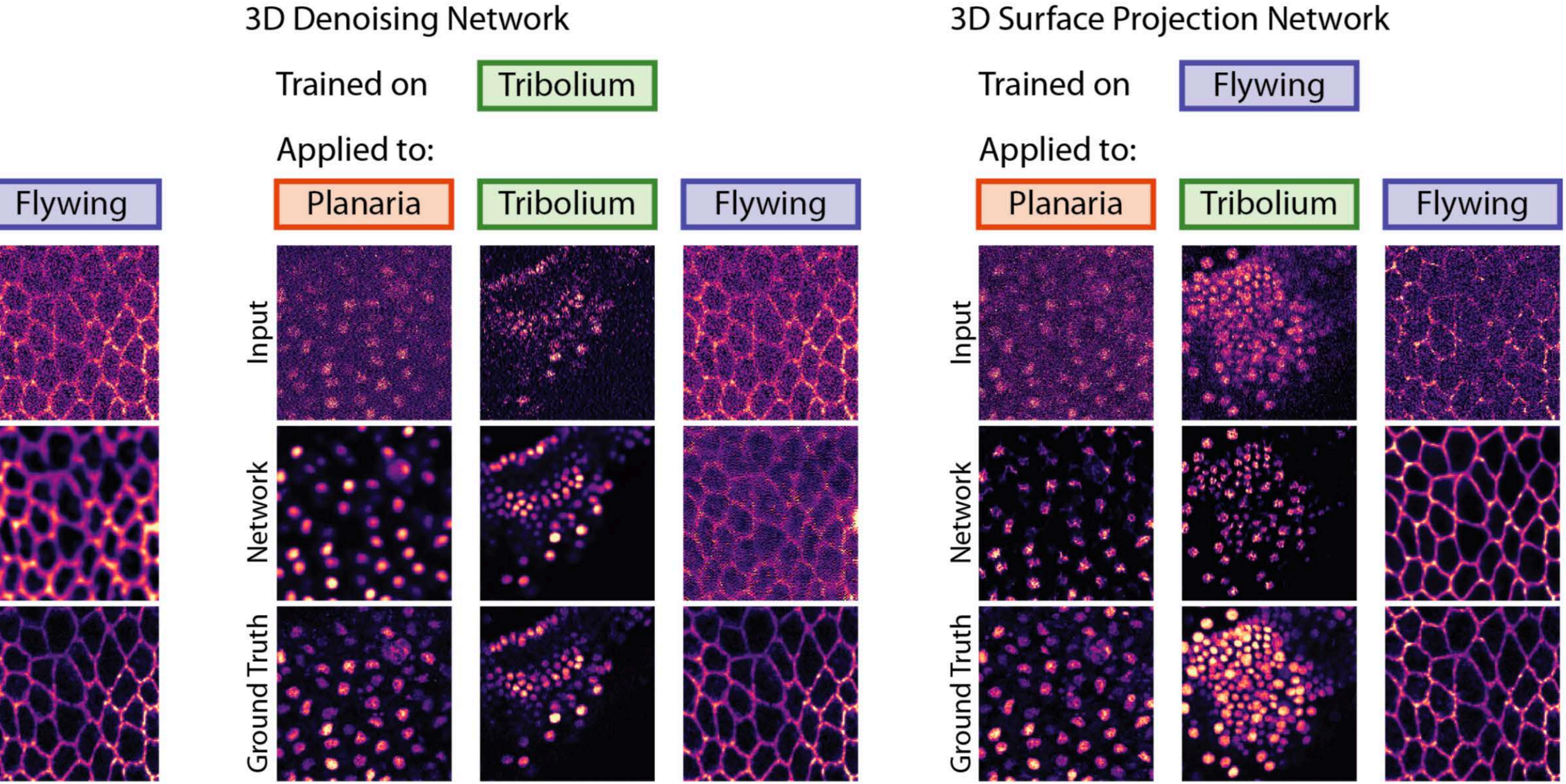


3D Surface

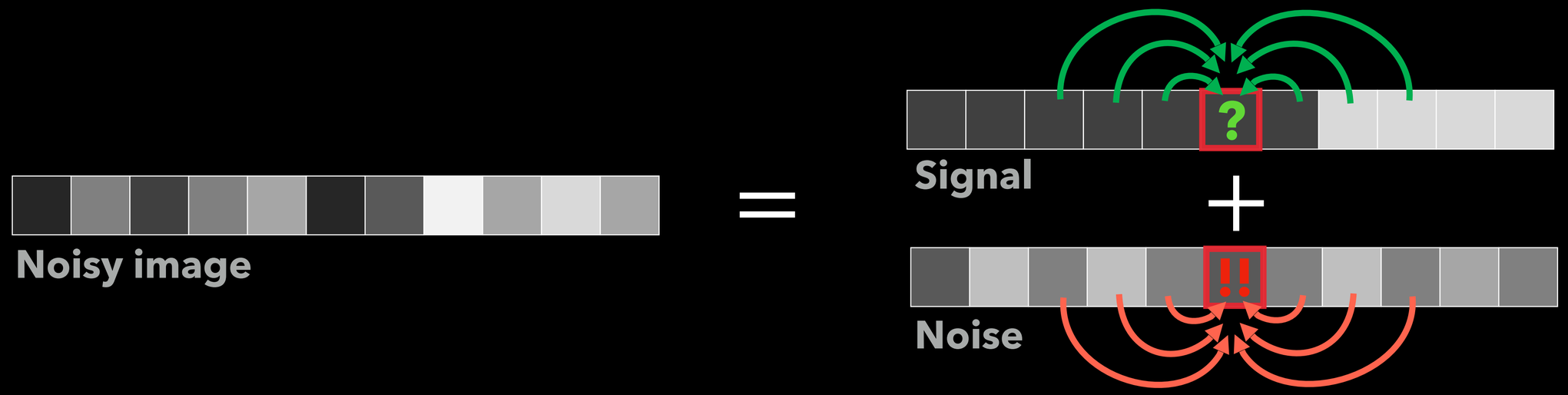
Trained o

Applied t




3D Denoising Network
Trained on
Tribolium
Applied to:
Flywing
Planaria
Tribolium
Flywing
Input
Network
Ground Truth
3D Surface Projection Network
Trained on
Flywing
Applied to:
Planaria
Tribolium
Flywing
Input
Network
Ground Truth

What if you'd only have **one** set of noisy images?

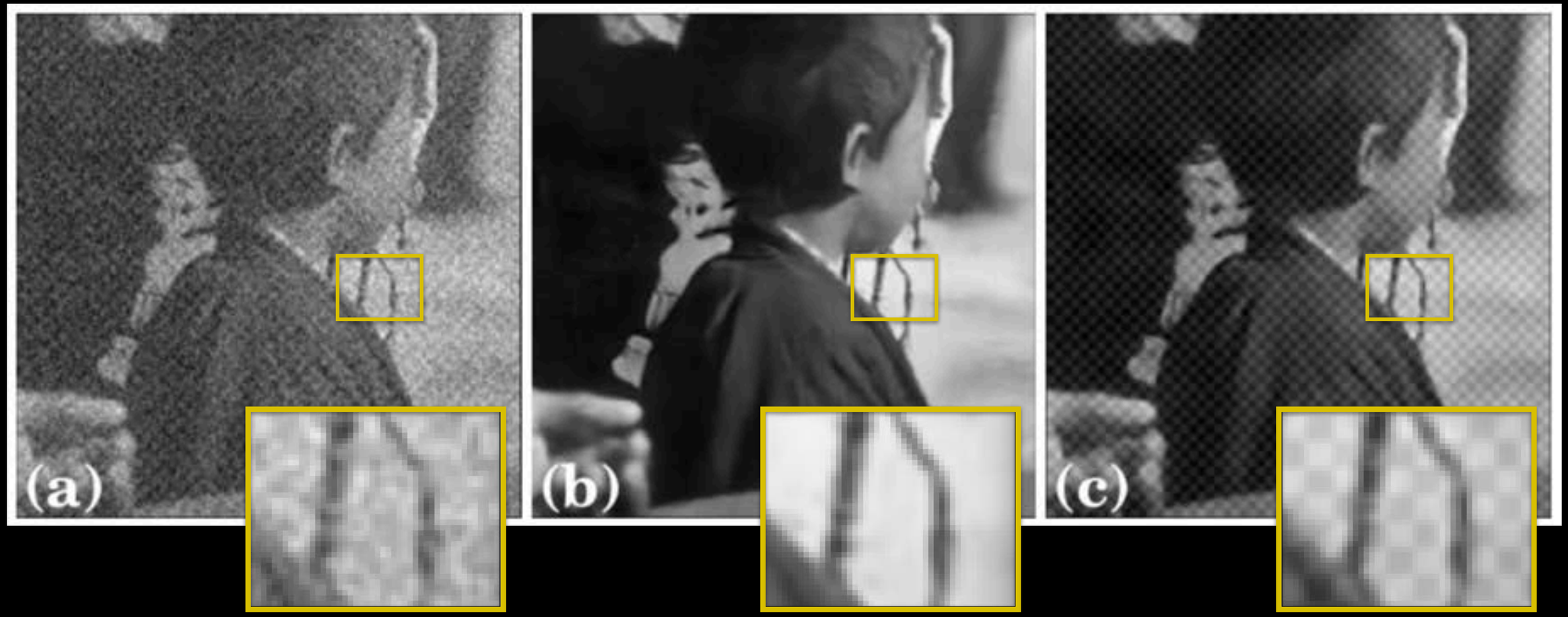


Another new idea:

Noise2Void

Krull et al., Noise2Void - Learning Denoising from Single Noisy Images, CVPR'19
Batson and Royer, Noise2Self - Blind Denoising by Self-Supervision, arXiv:1901.11365

INPUT

TRADITIONAL CARE

NOISE2VOID