Deep Denoising for Scientific Discovery

Carlos Fernandez-Granda

www.cims.nyu.edu/~cfgranda

12/1/2021

Acknowledgements

This work was supported by NSF grants OAC-1940097, OAC-2103936, and NRT-1922658

Acknowledgements

Sreyas Mohan (NYU, Flatiron Institute)

Zahra Kadkhodaie, Eero Simoncelli (NYU, Flatiron Institute)

Peter Crozier, Ramon Manzorro, Joshua Vincent (ASU)

Mitesh Khapra, Dev Sheth (IIT Madras)

David Matteson, Binh Tang (Cornell)

Motivation: Studying catalysis

90% of all manufactured goods involve catalytic processes somewhere in their production chain

Considerable impact in energy, healthcare (pharmaceuticals), new material (polymers), transport, and the environment (water, air-quality, renewable and bio-produced materials)

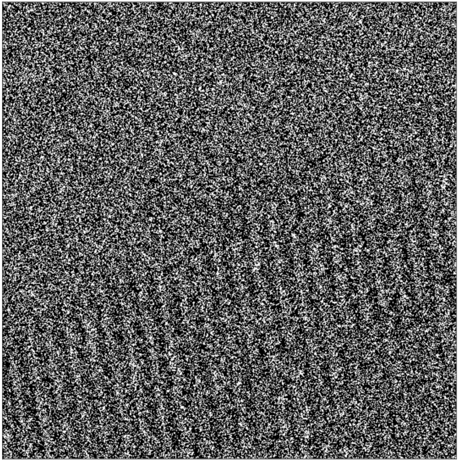
Motivation: Studying catalysis

90% of all manufactured goods involve catalytic processes somewhere in their production chain

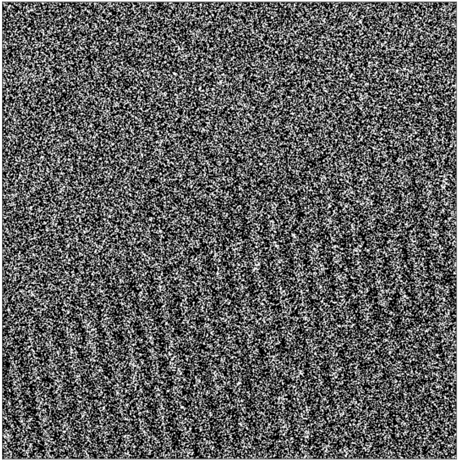
Considerable impact in energy, healthcare (pharmaceuticals), new material (polymers), transport, and the environment (water, air-quality, renewable and bio-produced materials)

To understand catalysis we need to see what is going on

Electron microscope image



Electron microscope image



We need to denoise!

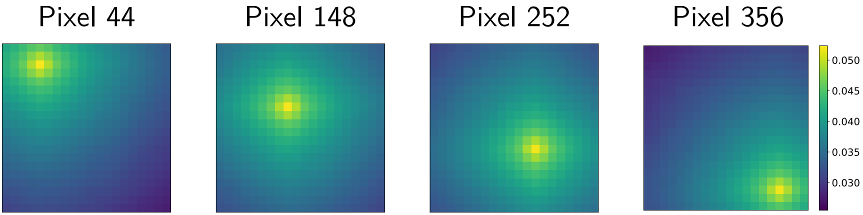
Traditional denoising

Linear regression from pixels to pixels is intractable ($10^4 \times 10^4$ matrix!)

Traditional denoising

Linear regression from pixels to pixels is intractable ($10^4 \times 10^4$ matrix!)

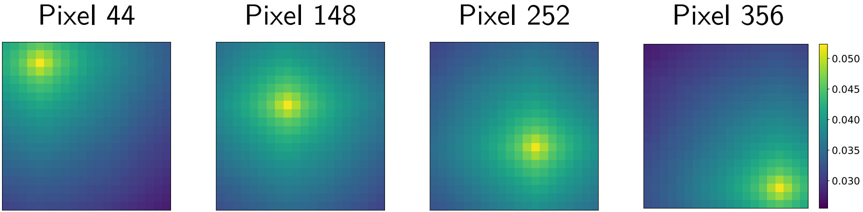
No need: Covariance between pixels is translation invariant



Traditional denoising

Linear regression from pixels to pixels is intractable ($10^4 \times 10^4$ matrix!)

No need: Covariance between pixels is translation invariant



Tractable alternative (Wiener 1950):

Optimize convolutional filter to minimize mean-squared error

Convolutional filter

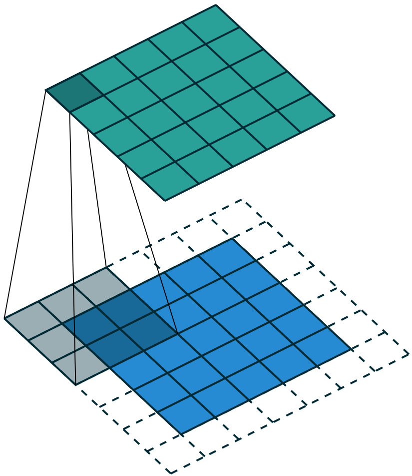


Image from *A guide to convolution arithmetic for deep learning*, Dumoulin & Visin, 2016.

Convolutional filter

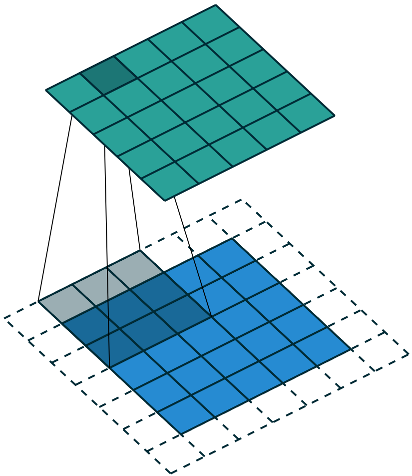


Image from *A guide to convolution arithmetic for deep learning*, Dumoulin & Visin, 2016.

Convolutional filter

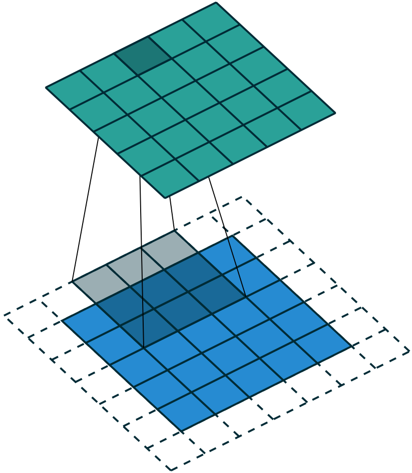


Image from *A guide to convolution arithmetic for deep learning*, Dumoulin & Visin, 2016.

Convolutional filter

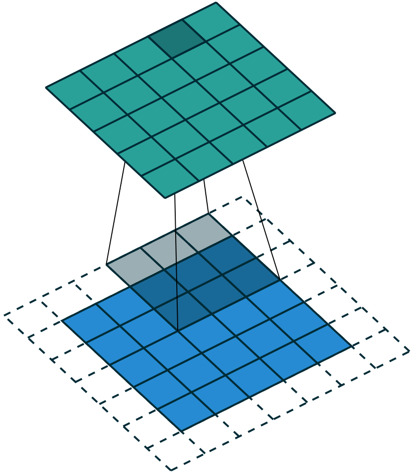
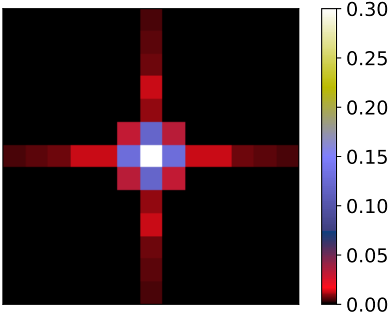


Image from *A guide to convolution arithmetic for deep learning*, Dumoulin & Visin, 2016.

Wiener filter (additive Gaussian noise. Low σ)

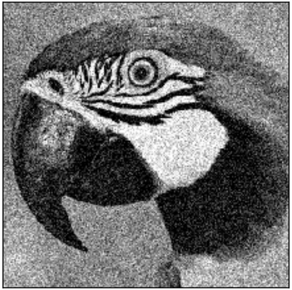
Example noisy image

Wiener filter

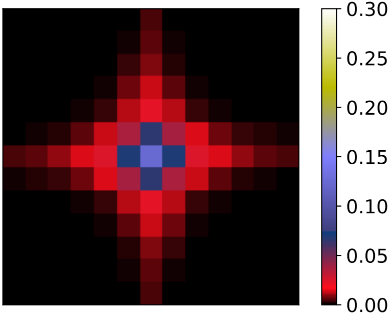


Wiener filter (additive Gaussian noise. Mid σ)

Example noisy image



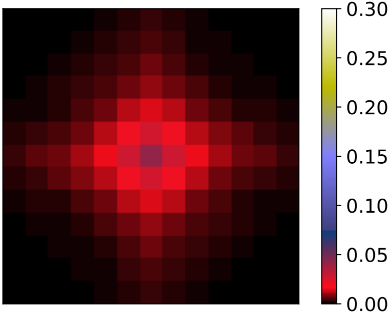
Wiener filter



Wiener filter (additive Gaussian noise. High σ)

Example noisy image

Wiener filter



Beyond the Wiener filter

Wiener filter: Weighted average of nearby pixels

Problem: Same average for each pixel

Blurs edges and other features

Beyond the Wiener filter

Wiener filter: Weighted average of nearby pixels

Problem: Same average for each pixel

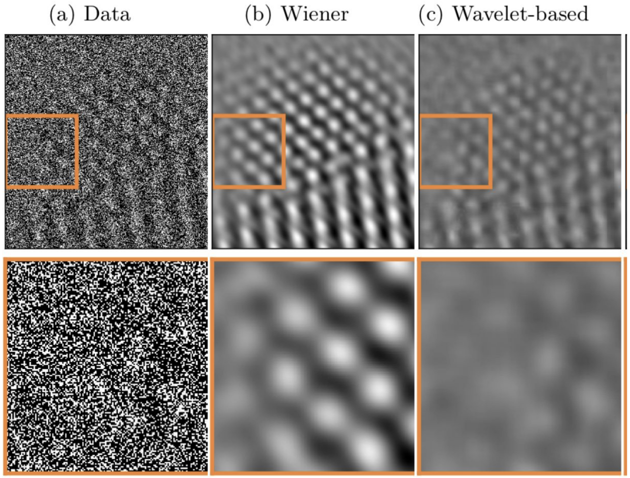
Blurs edges and other features

Pre-deep-learning solutions:

Adapt filter locally (e.g. bilateral filter [Tomasi and Manduchi 1998, Milanfar 2013])

Design/learn sparsifying transforms (wavelets, dictionary learning)

Results on electron microscopy

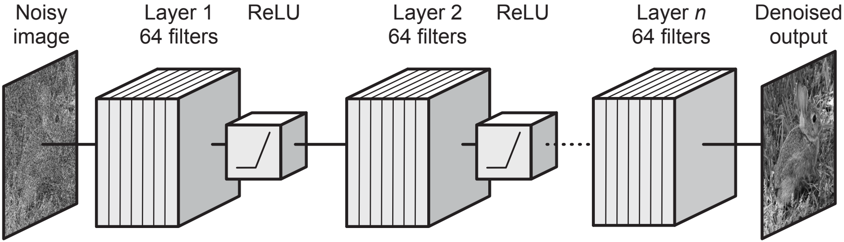


Deep-learning solution

Learn overparametrized nonlinear convolutional model

Deep learning for image denoising

Denoising Convolutional Neural Network (DnCNN)[1]



[1] *Beyond a Gaussian Denoiser: Residual Learning of Deep CNN for Image Denoising*. K. Zhang, W. Zuo, Y. Chen, D. Meng, L. Zhang. IEEE Transactions in Image Processing (2017)

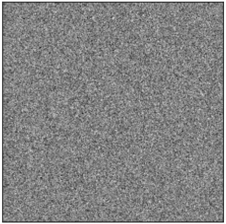
Deep learning for image denoising

- Gather dataset of natural images

Deep learning for image denoising

- Gather dataset of natural images
- Add synthetic Gaussian noise to generate noisy images

$+$



$=$

Deep learning for image denoising

- Gather dataset of natural images
- Add synthetic Gaussian noise to generate noisy images
- Train CNN to estimate clean image minimizing mean squared error

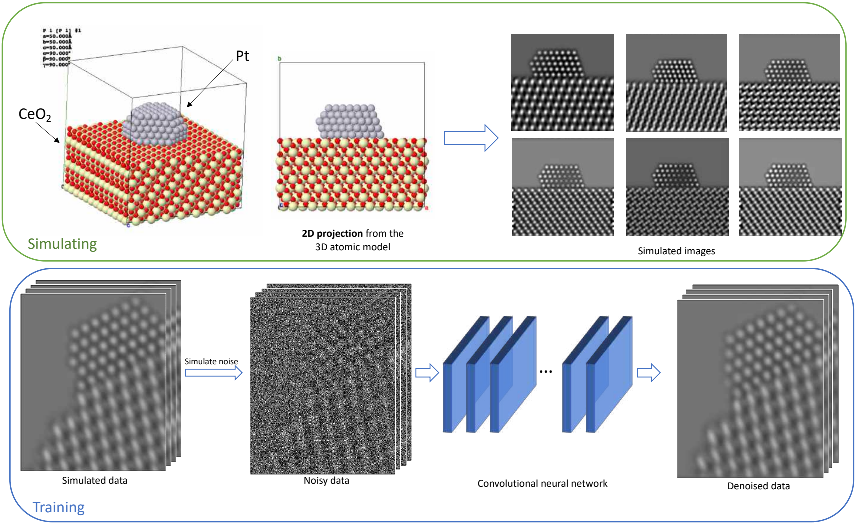
Works very well (state of the art)

Training data
(high noise)

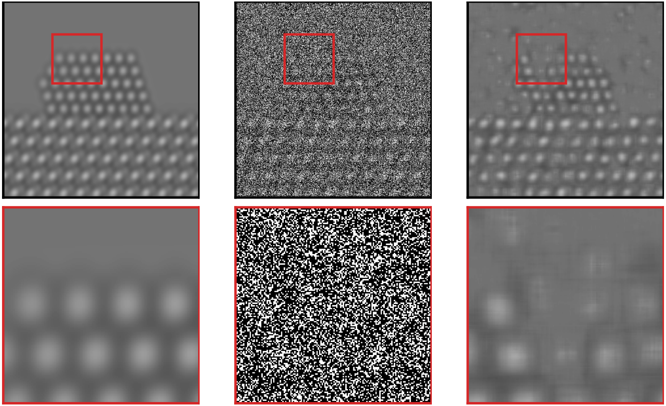
Test image
(high noise)

CNN

Application to electron microscopy



Application to electron microscopy



Simulated clean image

Noisy image

Denoised

Challenges in practice

Challenges in practice

- We need robustness to changes in imaging conditions

Challenges in practice

- We need robustness to changes in imaging conditions
- We need interpretability to understand how model works and adapt it

Challenges in practice

- We need robustness to changes in imaging conditions
- We need interpretability to understand how model works and adapt it
- We do not have ground-truth clean data to train the networks

Robustness

Interpretability

Unsupervised Denoising

Back to robustness

Generalization across noise levels

What if we test on noise level not seen during training?

Training data
(low noise)

Test image
(high noise)

Generalization across noise levels

What if we test on noise level not seen during training?

Training data
(low noise)

Test image
(high noise)

CNN

First-order Taylor expansion

Let f be the function learned by a CNN trained for denoising

First-order Taylor expansion

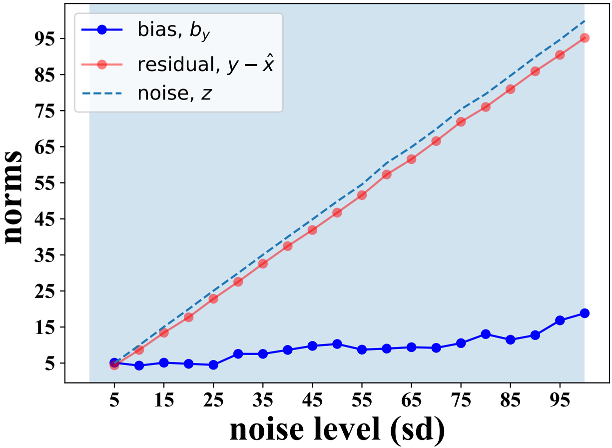
Let f be the function learned by a CNN trained for denoising

First-order Taylor expansion for fixed input y

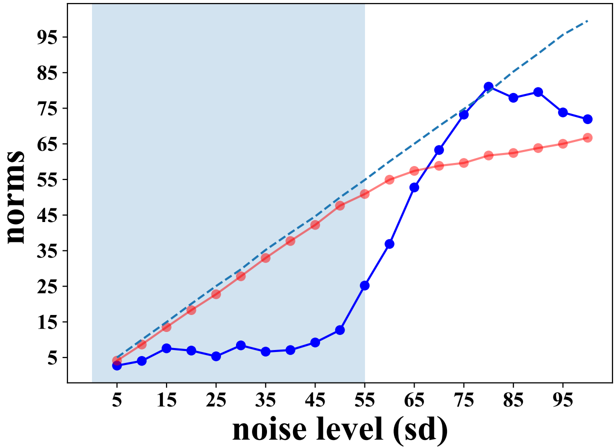
$$\begin{aligned}\hat{x} = f(y) &= W_L R(\ldots W_2 R(W_1 y + b_1) + b_2 \ldots) + b_L \\ &= A_y y + b_y\end{aligned}$$

W_1, W_2, ..., W_L are weight matrices
b_1, b_2, ..., b_L are bias vectors

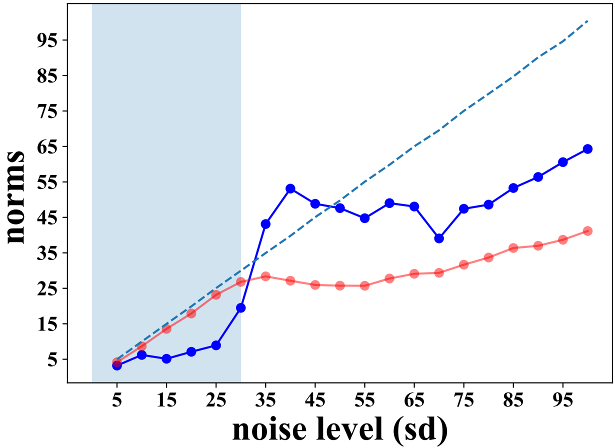
Residual and net bias



Residual and net bias



Residual and net bias



The bias overfits

Within training range, learned net bias is small

Out of the range, it explodes, coinciding with dramatic performance loss

Net bias seems to overfit trained noise levels

The bias overfits

Within training range, learned net bias is small

Out of the range, it explodes, coinciding with dramatic performance loss

Net bias seems to overfit trained noise levels

This motivates *removing all additive constants*

$$f(y) = W_L R(\ldots W_2 R(W_1 y + b_1) + b_2 \ldots) + b_L$$

The bias overfits

Within training range, learned net bias is small

Out of the range, it explodes, coinciding with dramatic performance loss

Net bias seems to overfit trained noise levels

This motivates *removing all additive constants*

$$f(y) = W_L R(\dots W_2 R(W_1 y + \cancel{b_1}) + \cancel{b_2} \dots) + \cancel{b_L}$$

It works

Training data
(low noise)

Test image
(high noise)

CNN

It works

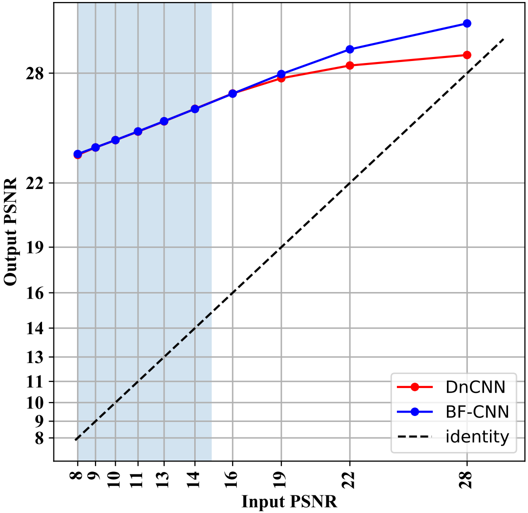
Training data (low noise)

Test image (high noise)

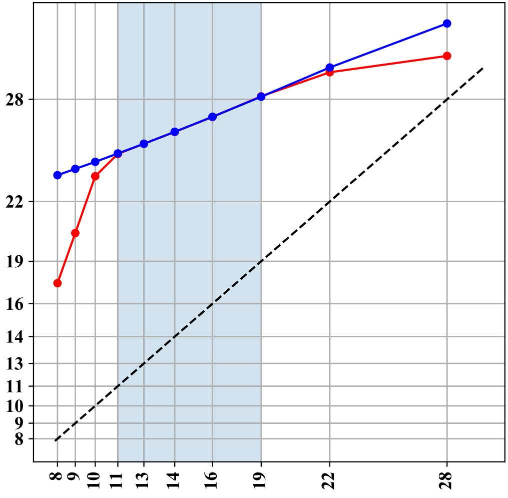
CNN

Bias-free CNN

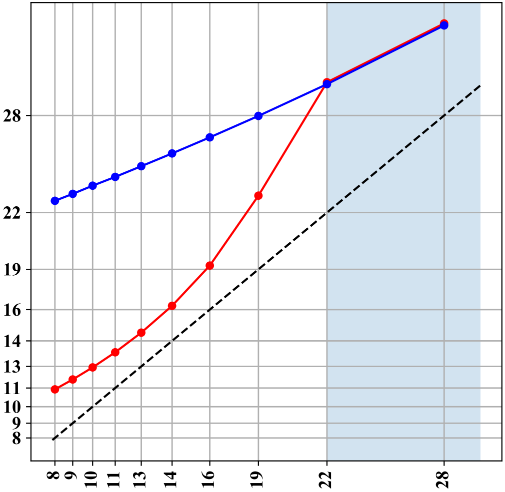
DnCNN vs bias-free DnCNN



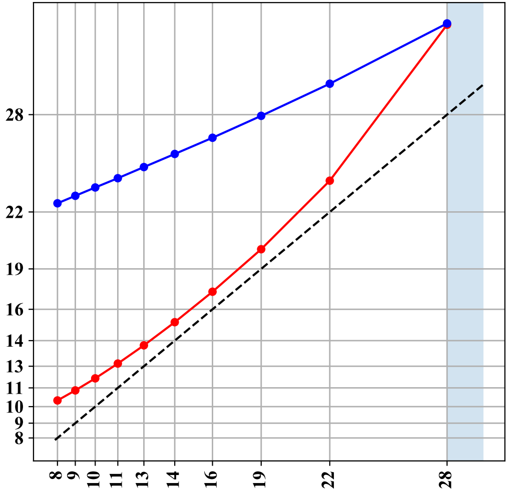
DnCNN vs bias-free DnCNN



DnCNN vs bias-free DnCNN



DnCNN vs bias-free DnCNN



Take away

Net bias overfits to noise level in training data

Bias-free networks generalize to new noise levels

Bias-free CNNs beyond denoising

- Deblurring, super-resolution and demosaicing using plug-and-play method. [Zhang et. al. IEEE PAMI 2021]
- Reflection removal. [Zheng et. al. CVPR 2021.]
- Tone mapping. [Le et. al. ICVRV 2021]
- Generative modelling. [Kadkhodaie et. al. NeurIPS 2021]
- Photometric stereo. [Honzatko et. al. 2021]

Robustness

Interpretability

Unsupervised Denoising

Back to robustness

Bias-free CNN is locally linear

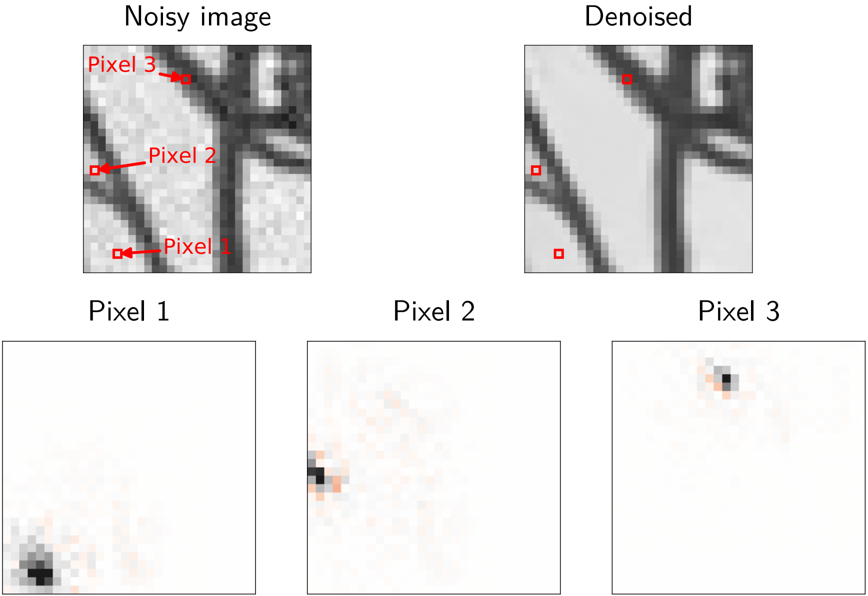
$$f(y) = W_L \, R \, W_{L-1} ... R \, W_1 y = A_y y$$

Rows interpreted as filters

Estimate at pixel i:

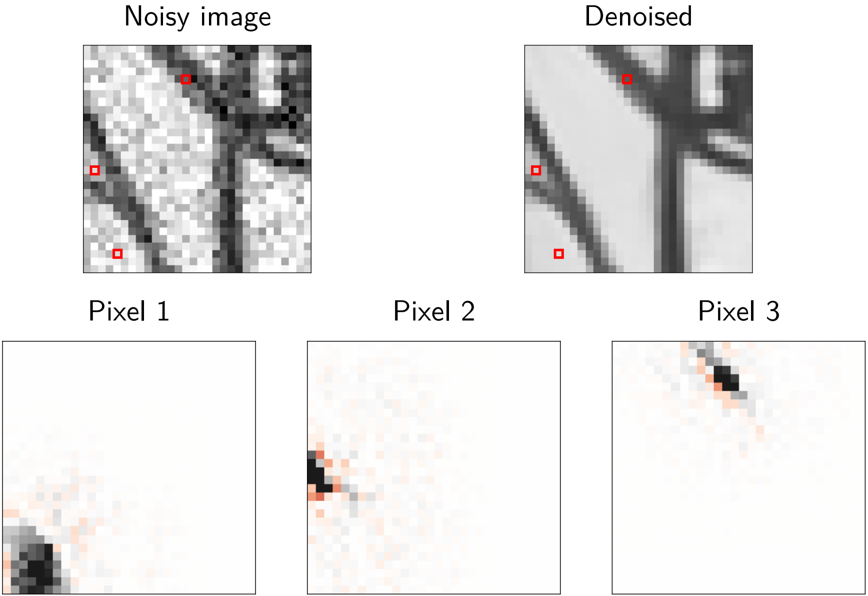
$$f_{\mathrm{BF}}(y)_i = (A_y y)_i = < i\text{th row of } A_y, y >$$

Low noise

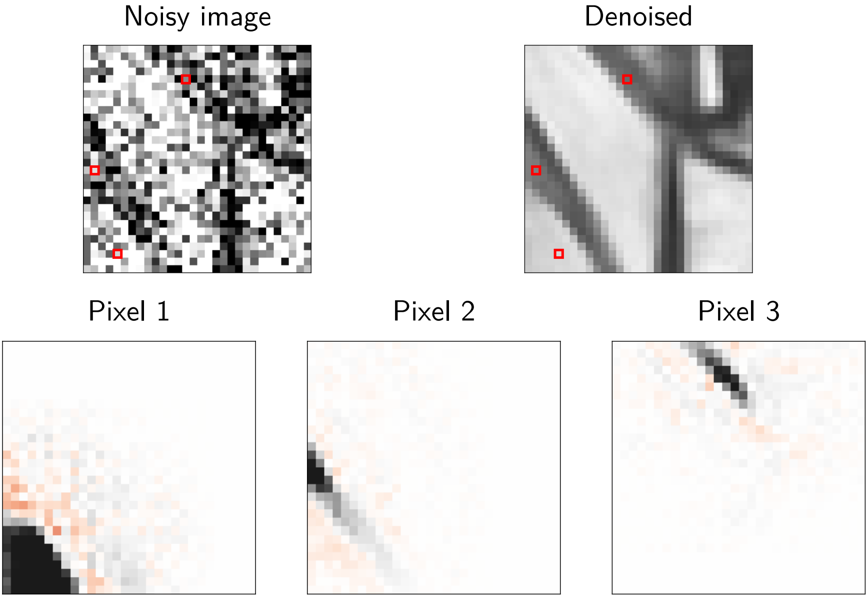


Noisy image
Pixel 3
Pixel 2
Pixel 1
Denoised
Pixel 1
Pixel 2
Pixel 3

Medium noise



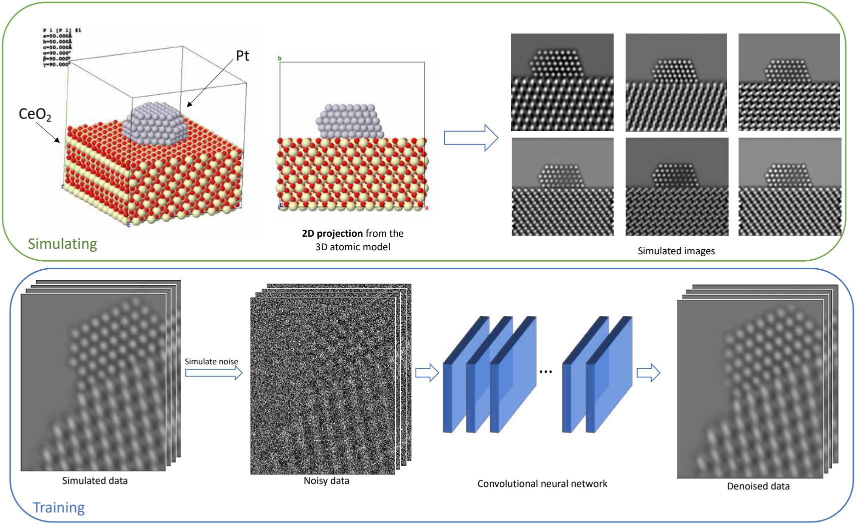
High noise



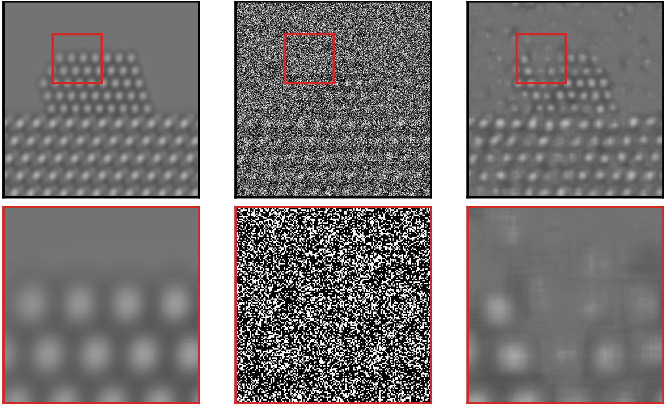
Take away

CNNs implicitly learns filters *adapted to image structure and noise*!

Application to electron microscopy



Application to electron microscopy

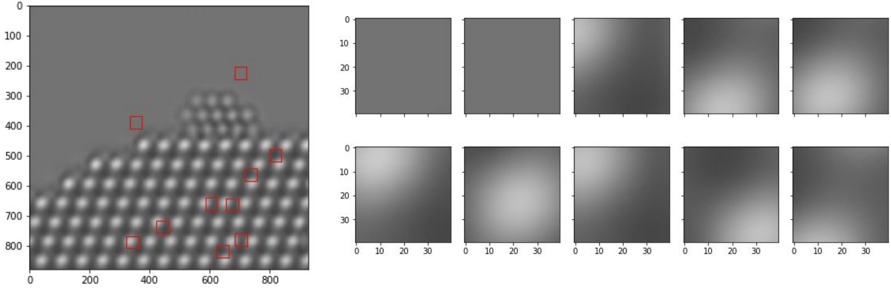


Simulated clean image

Noisy image

Denoised using DnCNN

Equivalent filters of DnCNN: small receptive field



Cannot exploit periodicity

Increasing field of view

Electron Microscopy

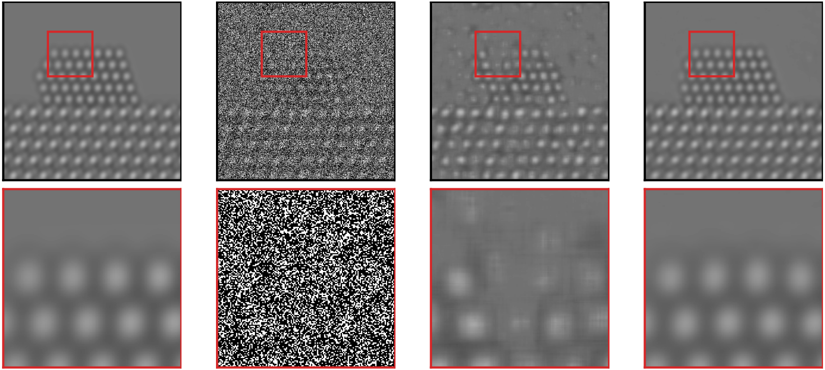
Model	Parameters	FoV	PSNR
SBD + DnCNN	668K	41×41	30.47 ± 0.64
SBD + Small UNet	233K	45×45	30.87 ± 0.56
SBD + UNet (32 base channels)	352K	221×221	36.39 ± 0.77
SBD + UNet (64 base channels)	1.41M	221×221	37.24 ± 0.76
SBD + UNet (128 base channels)	5.61M	221×221	38.05 ± 0.81
SBD + UNet (128 base channels)	70.15M	893×893	42.87 ± 1.45

Increasing field of view

Natural Images

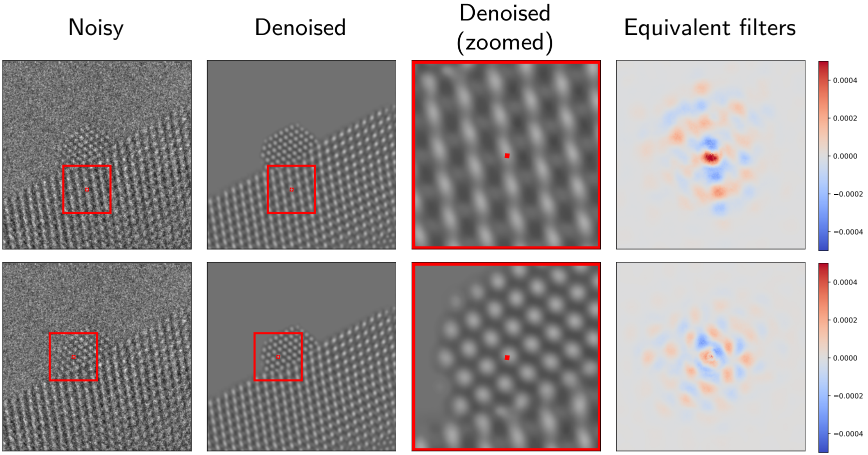
Model	Params	FoV	PSNR	
			$\sigma = 30$	$\sigma = 70$
UNet	102K	49 $\times$ 49	29.67 $\pm$ 2.84	26.16 $\pm$ 2.79
UNet	352K	221 $\times$ 221	29.65 $\pm$ 2.76	26.08 $\pm$ 2.68
UNet	4.4M	893 $\times$ 893	29.54 $\pm$ 2.82	26.07 $\pm$ 2.80

Results



Simulated clean image | Noisy image | DnCNN | Large receptive field

Equivalent filters



Robustness

Interpretability

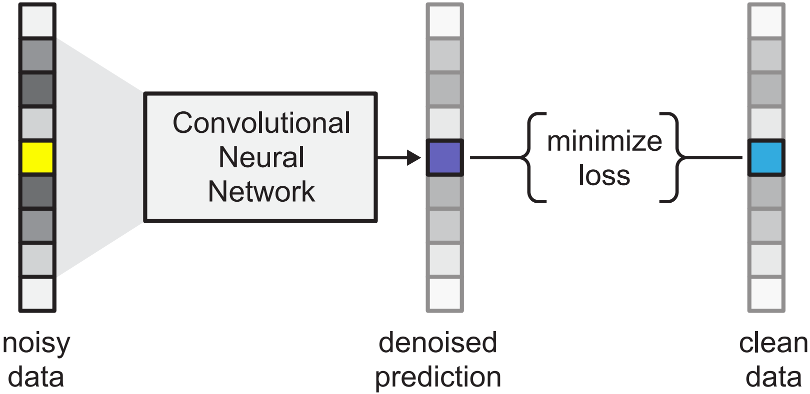
Unsupervised Denoising

Back to robustness

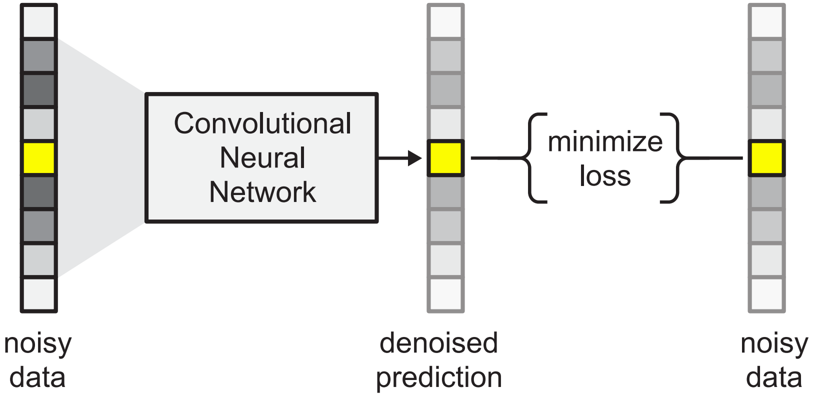
Challenge

What if we can't simulate ground truth (because we don't know it!)

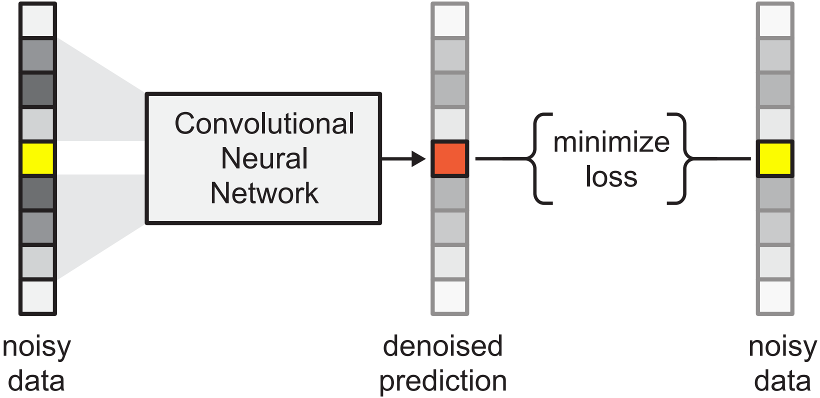
Recap: clean data is available



Only noisy data is available: thought experiment



Blind-spot denoising



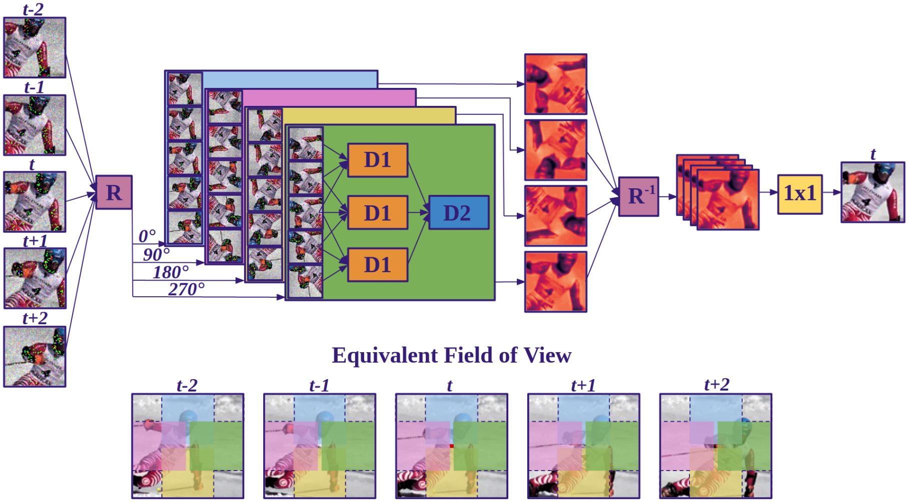
Unsupervised Denoising

- *Noise2noise: Learning image restoration without clean data*. Lehtinen, J., Munkberg, J., Hasselgren, J., Laine, S., Karras, T., Aittala, M., Aila, T. ICML 2018
- *Noise2void-learning denoising from single noisy images* A. Krull, T. Buchholz, F. Jug. CVPR 2019
- *Noise2self: Blind denoising by self-supervision*. J. Batson, L. Royer. ICML 2019
- *High-quality self-supervised deep image denoising* S. Laine, T. Karras, J. Lehtinen, T. Aila. Neurips 2019

Application to electron microscopy

We have videos, not single images

Unsupervised Deep Video Denoising

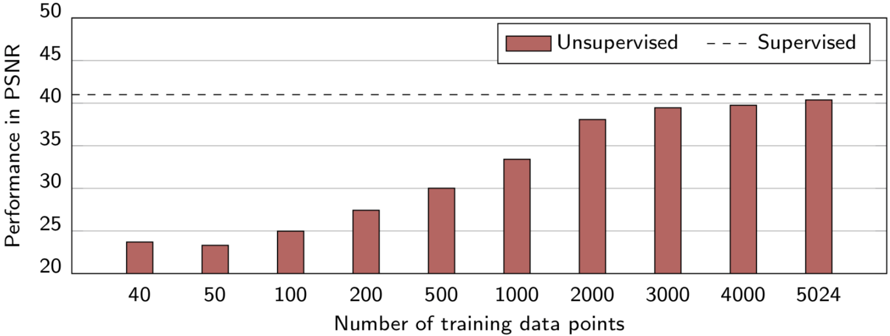


Architecture based on [Laine et. al. 2019], [Tassano et. al. 2019], and [Tassano et. al. 2020].

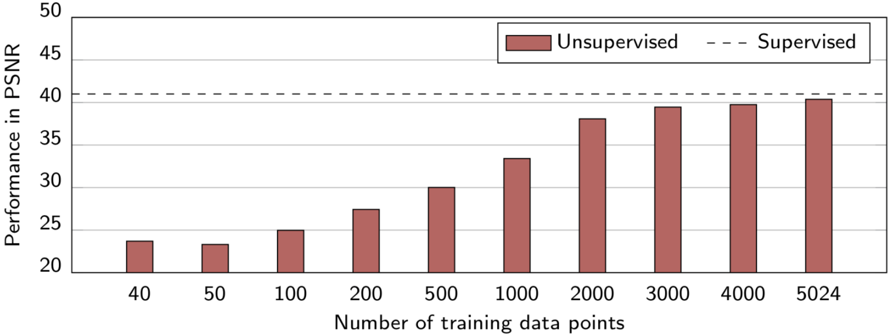
Performance comparable to supervised state of the art

		Traditional		Supervised CNN			Unsupervised CNN (UDVD)		
test set	σ	VNLB	VBM4D	VNLnet	DVDnet	FastDVDnet	1 frame	3 frames	5 frames
DAVIS	30	33.73	31.65	-	**34.08**	34.06	32.80	33.48	**33.92**
	40	32.32	30.05	32.32	**32.86**	32.80	31.48	32.20	**32.68**
	50	31.13	28.80	31.43	**31.85**	31.83	30.47	31.20	**31.70**
Set8	30	31.74	30.00	-	**31.79**	31.60	30.91	31.62	**32.01**
	40	30.39	28.48	**30.55**	**30.55**	30.37	29.63	30.42	**30.82**
	50	29.24	27.33	29.47	**29.56**	29.42	28.65	29.47	**29.89**

Problem: Requires a lot of data



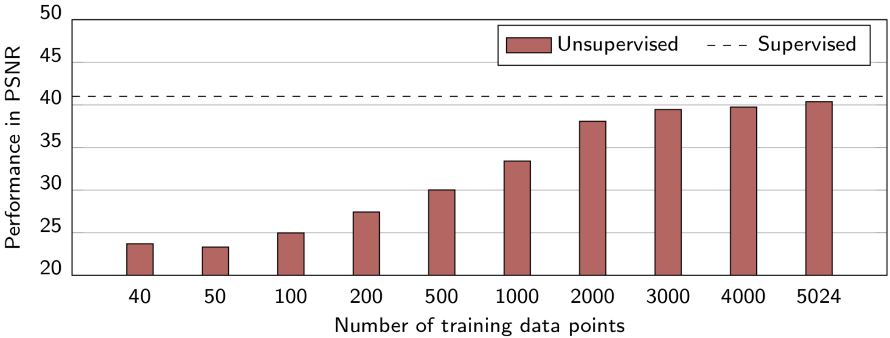
Problem: Requires a lot of data



Solution:

- Data augmentation

Problem: Requires a lot of data



Solution:

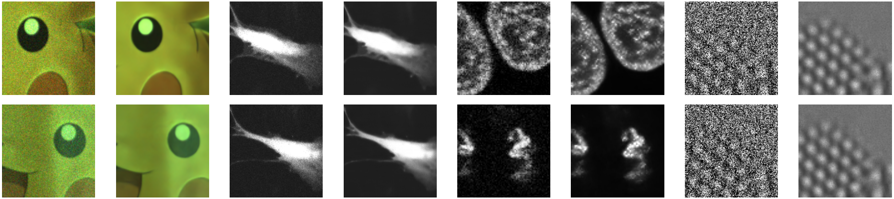
- Data augmentation
- Early stopping

Solution: Early stopping + data augmentation

	$\sigma = 90$									
	ten-v	snow	hyper	raft	motor	trac	sunf	touch	park	**mean**
No. of frames	75	59	37	29	32	85	85	85	85	-
No Aug (without ES)	24.13	22.89	22.04	20.99	20.06	24.84	25.98	25.67	23.35	23.33
No Aug (with ES)	30.15	25.49	27.48	26.05	23.79	28.18	31.91	29.87	25.46	27.60
F (without ES)	27.21	24.42	24.05	23.32	21.84	27.42	29.53	28.01	25.03	25.65
F (with ES)	30.35	**25.60**	27.72	*26.16*	23.89	**28.71**	32.17	29.93	25.59	27.79
F+TR (without ES)	27.11	24.77	24.25	23.55	21.98	27.80	30.22	28.56	25.44	25.96
F+TR (with ES)	**30.40**	25.59	**27.75**	*26.16*	**23.92**	28.63	**32.18**	**29.96**	**25.62**	**27.80**
UDVD*	28.78	25.16	26.78	25.81	23.57	26.42	29.04	28.71	24.23	26.50
FastDVDnet*	29.44	25.25	27.30	**26.35**	23.68	27.42	30.29	29.61	24.72	27.12

Real-world data

CNN \ ISO	1600	3200	6400	12800	25600	**mean**
UDVD	**48.04**	**46.24**	**44.70**	**42.19**	**42.11**	**44.69**
RViDeNet[2]	47.74	45.91	43.85	41.20	41.17	43.97



Raw video Fluorescence micr. Fluorescence micr. Electron microscopy

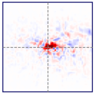
[2]Yue. et. al. CVPR 2020

Interpreting video denoisers

Most video denoisers compute optical flow, but UDVD does not

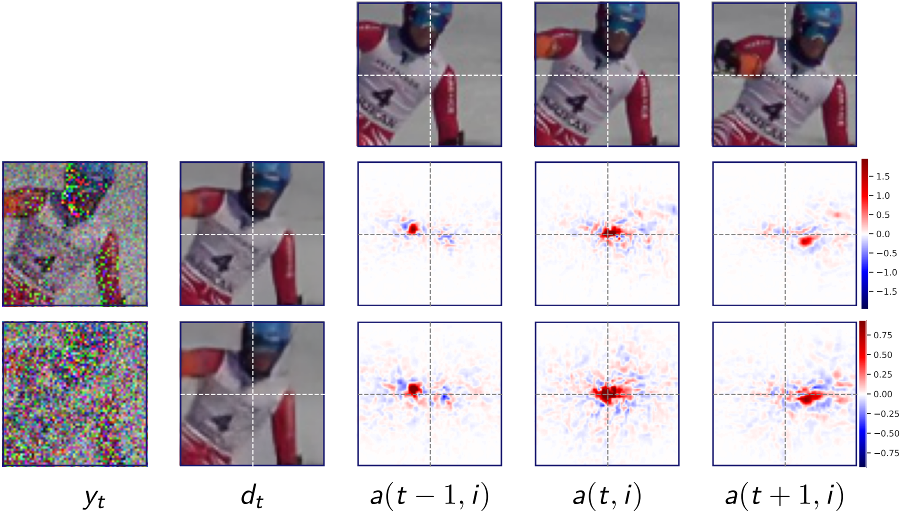
How does it achieve such good denoising?

Equivalent filters

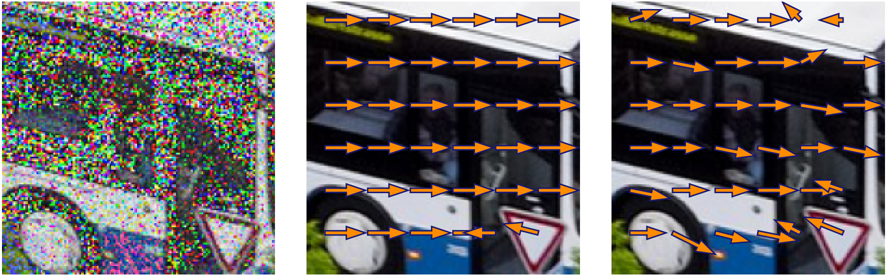


$$d_t(i) = \sum_{k=-2}^{2} \left\langle y_{t-k}, a(t-k,i) \right\rangle$$

UDVD learns adaptive spatio-temporal filtering



UDVD performs implicit motion compensation



(a) Noisy frame ($\sigma = 30$)

(b) Motion estimate from clean video

(c) Motion estimate from UDVD gradients

Take away

Networks trained for denoising learn to perform motion compensation!

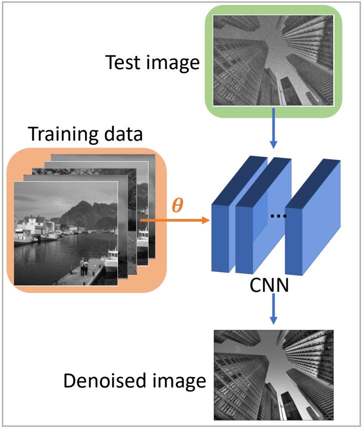
Robustness

Interpretability

Unsupervised Denoising

Back to robustness

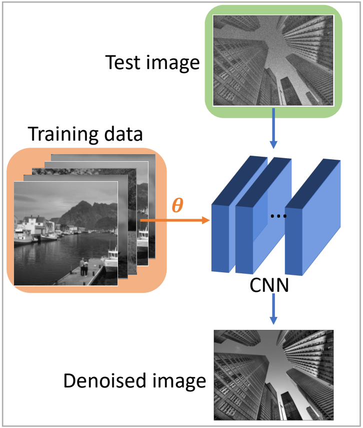
Standard supervised paradigm



Test and training data from

- same distribution
- different distributions

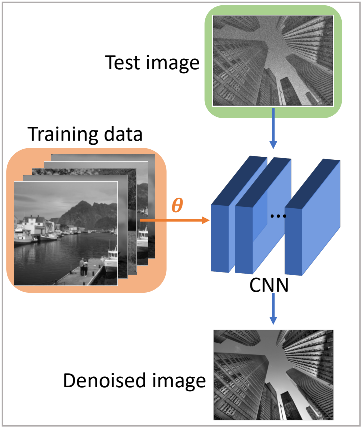
Standard supervised paradigm



Test and training data from

- same distribution 😀
- different distributions

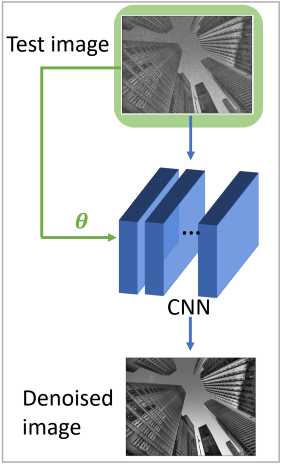
Standard supervised paradigm



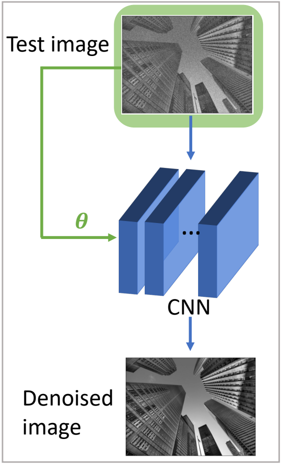
Test and training data from

- same distribution 😀
- different distributions ☹️

Training on test data (unsupervised paradigm)

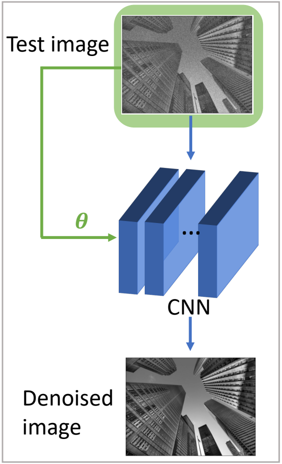


Training on test data (unsupervised paradigm)



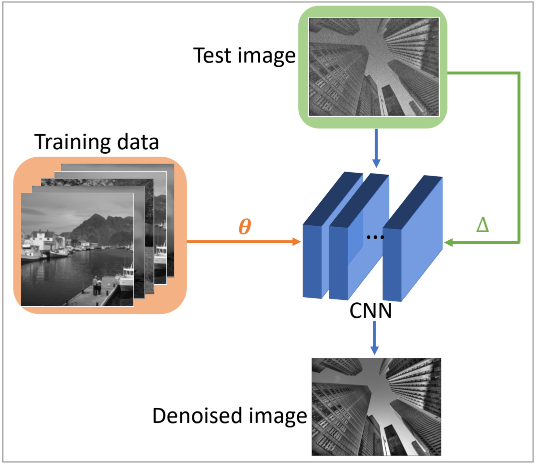
- Adaptation to test data 🙂

Training on test data (unsupervised paradigm)

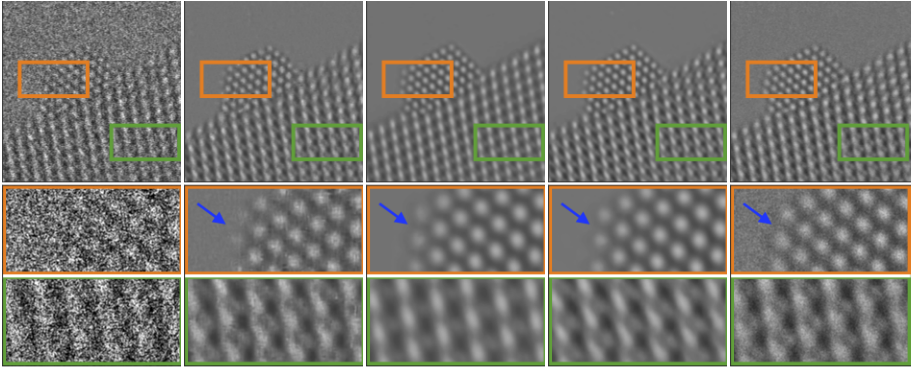


- Adaptation to test data 🙂
- Limited data ☹️

Proposed paradigm: Train, then adapt



It works!



Noisy image Unsupervised Supervised GainTuning Reference

Cost functions for test-time adaptation

1. Blind-spot technique
2. SURE: Stein's Unbiased Risk Estimator [Stein, 1981]
3. Noise resampling [Vaksman et. al. 2020]

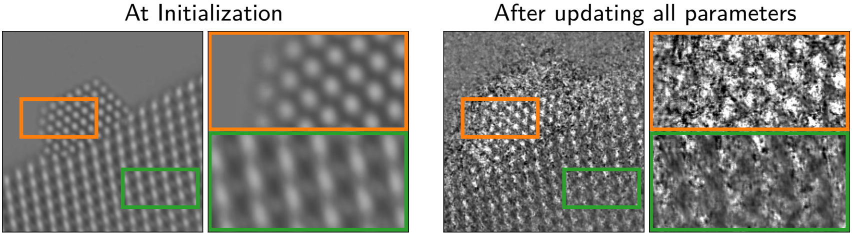
What parameters should we update

All of them [Soltanayev et. al. 2019, Vaksman et. al. 2020]?

What parameters should we update

All of them [Soltanayev et. al. 2019, Vaksman et. al. 2020]?

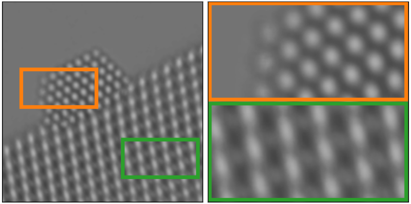
Problem: Severely overfits the noise



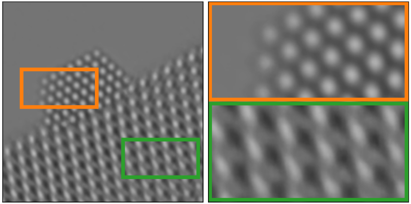
What parameters should we update

Proposed solution: Update only a single multiplicative *gain* per channel in each layer ($\approx 0.1\%$ of total)

At Initialization



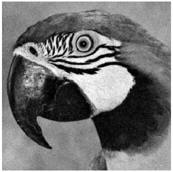
After updating only gains



Proof of concept

What if we test on noise level not seen during training?

Training data
(low noise)



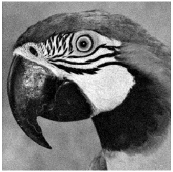
Test image
(high noise)

CNN
(pre-trained)

Proof of concept

What if we test on noise level not seen during training?

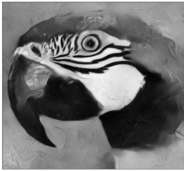
Training data (low noise)



Test image (high noise)

CNN (pre-trained)

CNN (GainTuning)

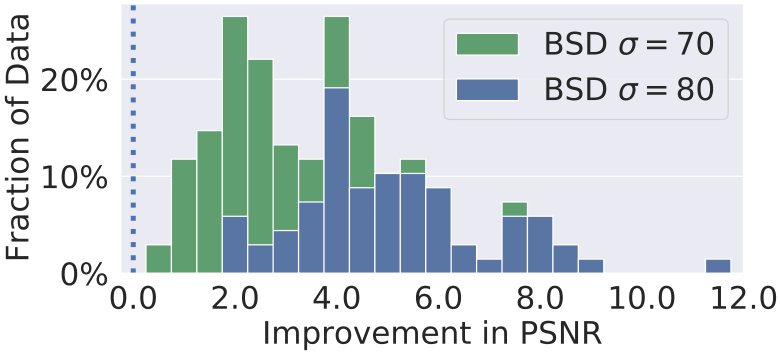


GainTuning for out-of-distribution noise

Test set	σ	Trained on $\sigma \in [0, 55]$		Bias Free Model	Trained on $\sigma \in [0, 100]$
		Pre-trained	Gaintuning		
Set12	70	22.45	25.54	25.59	25.50
	80	18.48	24.57	24.94	24.88
BSD68	70	22.15	24.89	24.87	24.88
	80	18.72	24.14	24.38	24.36

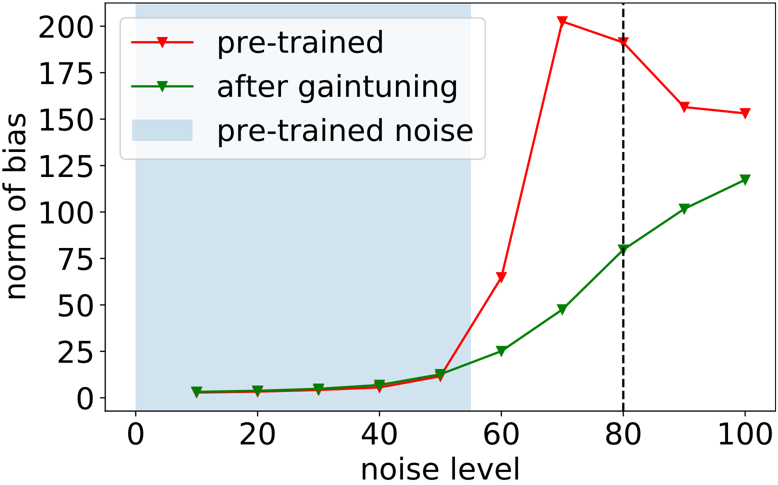
Can also adapt from Gaussian denoising to Poisson noise.

GainTuning for out-of-distribution noise



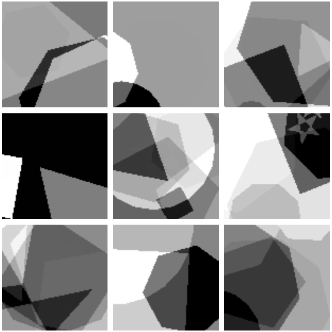
Can also adapt from Gaussian denoising to Poisson noise.

Reduces equivalent bias!



Another proof of concept

Training data



Piecewise constant images

Test data

Natural images

	Pre-trained	GainTuning
Average PSNR on test data	27.31	28.60

Another proof of concept

Training data

Natural images

Test data

Urban images

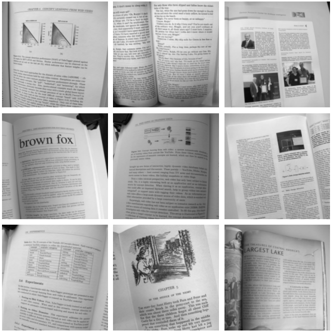
	Pre-trained	GainTuning
Average PSNR on test data	28.35	28.79

Another proof of concept

Training data

Natural images

Test data

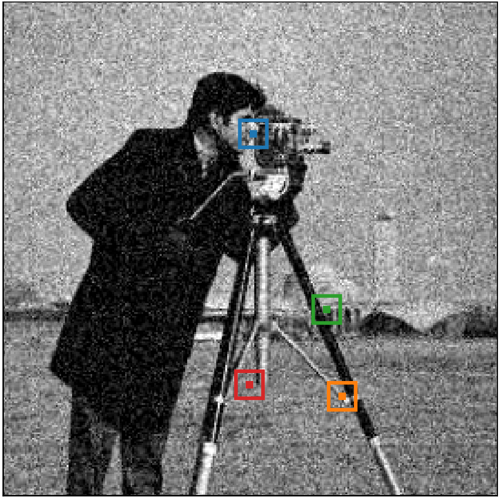


Scanned documents

	Pre-trained	GainTuning
Average PSNR on test data	30.02	30.73

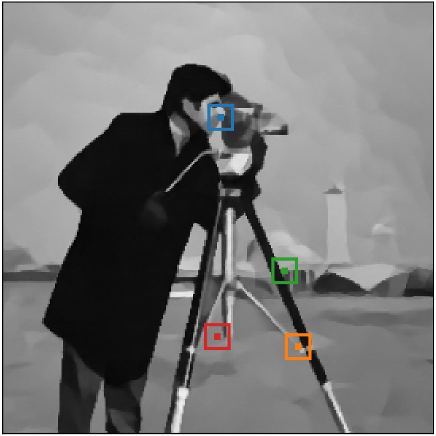
What is going on?

Noisy image

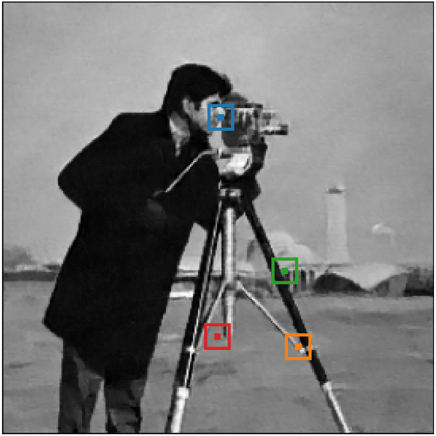


What is going on?

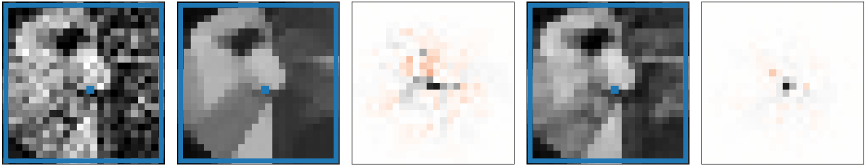
Trained on piecewise constant



After GainTuning

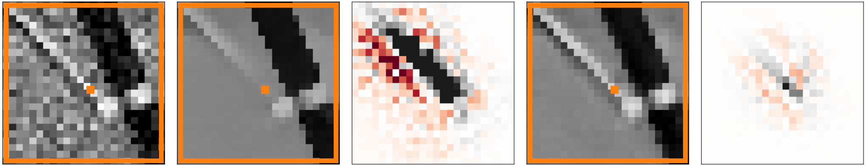


Equivalent filters



Noisy data Before GT Filter After GT Filter

Equivalent filters



Noisy data | Before GT | Filter | After GT | Filter

For more information

Robust and interpretable blind image denoising via bias-free convolutional neural networks
Mohan & Kadkhodaie et. al. ICLR 2020

Unsupervised deep video denoising
Sheth & Mohan et. al. ICCV 2021

Adaptive denoising via GainTuning
Mohan et. al. NeurIPS 2021

Deep denoising for scientific discovery: a case study in electron microscopy
Mohan et. al. 2021 (under review)

Developing and Evaluating Deep Neural Network-based denoising for Nanoparticle TEM Images with Ultra-low Signal-to-Noise
Vincent et. al. Microscopy & Microanalysis 2021