Deep Image Prior (and Its Cousin) for Inverse Problems: the Untold Stories

Ju Sun
Computer Science & Engineering, UMN

Mid-West Machine Learning Symposium (MMLS), 2023
On the occasion of NeurIPS'23 deadline

University of Minnesota
Driven to Discover℠

Thanks to

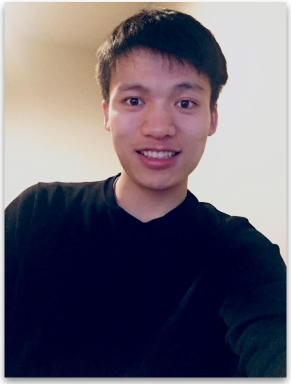
GLOVEX

https://glovex.umn.edu/

Taihui Li (CS&E)

Zhong Zhuang (ECE)

Hengkang Wang (CS&E)



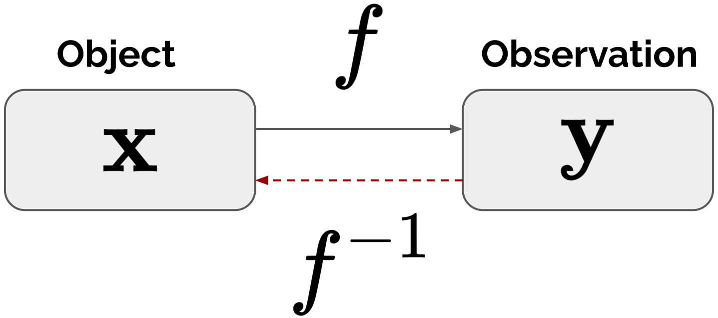
Le Peng (CS&E)

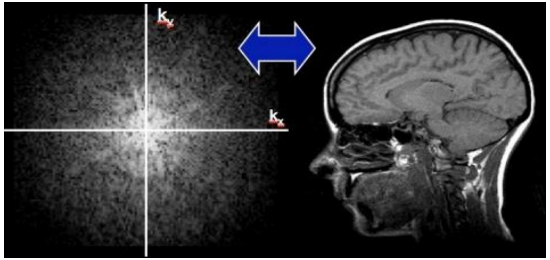
Hengyue Liang (ECE)

Tiancong Chen (CS&E)

Visual inverse problems

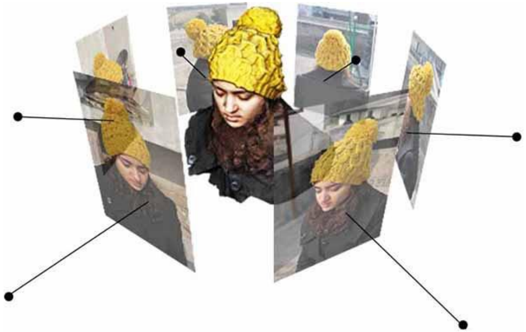
Image denoising



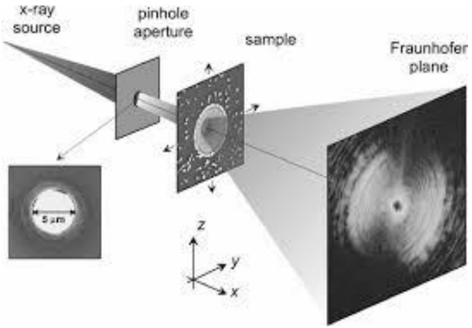


MRI reconstruction

Image super-resolution



3D reconstruction



Coherent diffraction imaging (CDI)

Traditional methods

Inverse problem: given $\mathbf{y} = f(\mathbf{x})$, recover $\mathbf{x}$

$$\min_{\mathbf{x}} \underbrace{\ell(\mathbf{y},\ f(\mathbf{x}))}_{\text{data fitting}} + \lambda \underbrace{\mathrm{R}(\mathbf{x})}_{\text{regularizer}}$$

RegFit

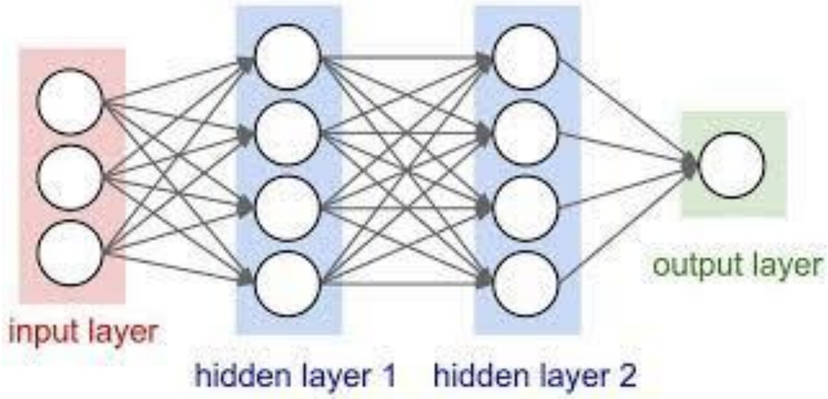
How has deep learning (DL) changed the story?

DL methods: the radical way

Inverse problem: given $\mathbf{y} = f(\mathbf{x})$, recover $\mathbf{x}$

Learn the f^{-1} with a training set $\{(\mathbf{y}_i, \mathbf{x}_i)\}$

$\mathbf{y}_i$



$\mathbf{x}_i$

Limitations:

- Wasteful: not using f
- Representative data?
- Not always straightforward

(see, e.g., Tayal et al. **Inverse Problems, Deep Learning, and Symmetry Breaking**. https://arxiv.org/abs/2003.09077)

DL methods: the middle way

Inverse problem: given $\mathbf{y} = f(\mathbf{x})$, recover $\mathbf{x}$

$$\min_{\mathbf{x}} \underbrace{\ell(\mathbf{y}, f(\mathbf{x}))}_{\text{data fitting}} + \lambda \underbrace{\mathrm{R}(\mathbf{x})}_{\text{regularizer}}$$

RegFit

Recipe: revamp numerical methods for RegFit with **pretrained/trainable DNNs**

DL methods: the middle way

Algorithm unrolling

$$\min_{\mathbf{x}} \underbrace{\ell(\mathbf{y},\, f(\mathbf{x}))}_{\text{data fitting}} + \lambda \underbrace{\mathrm{R}(\mathbf{x})}_{\text{regularizer}}$$

If R proximal friendly

$$\mathbf{x}^{k+1} = \mathcal{P}_R\left(\mathbf{x}^k - \eta \nabla^\top f(\mathbf{x}^k)\, \ell'(\mathbf{y},\, f(\mathbf{x}^k))\right)$$

Idea: make $\mathcal{P}_R$ trainable, using $\{(\mathbf{x}_i\,,\,\mathbf{y}_i)\}$

E.g.,

$$\ell(\mathbf{y},\, f(\mathbf{x})) = \|\mathbf{y} - \mathbf{A}\,\mathbf{x}\|_2^2$$

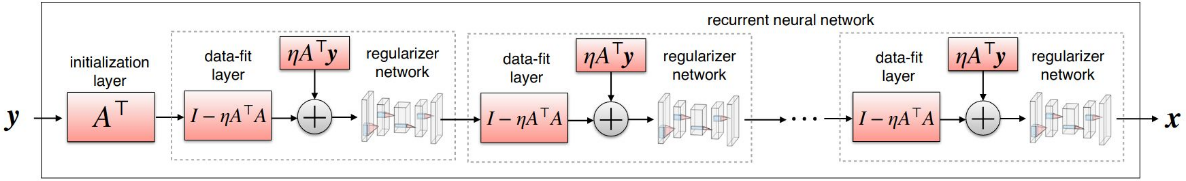


Fig credit: Deep Learning Techniques for Inverse Problems in Imaging https://arxiv.org/abs/2005.06001

DL methods: the middle way

$$\min_{\mathbf{x}} \underbrace{\ell(\mathbf{y},\, f(\mathbf{x}))}_{\text{data fitting}} + \lambda \underbrace{\mathrm{R}(\mathbf{x})}_{\text{regularizer}}$$

Using $\{\mathbf{x}_i\}$ only

Plug-and-Play

$$\mathbf{x}^{k+1} = \mathcal{P}_R\left(\mathbf{x}^k - \eta \nabla^\top f(\mathbf{x}^k)\ell'(\mathbf{y},\, f(\mathbf{x}^k))\right)$$

E.g. replace $\mathcal{P}_R$ with pretrained denoiser

Deep generative models

Pretraining: $\mathbf{x}_i \approx G_\theta(\mathbf{z}_i) \;\; \forall i$

Deployment: $\min_{\mathbf{z}} \; \ell(\mathbf{y},\, f \circ G_\theta(\mathbf{z})) + \lambda R \circ G_\theta(\mathbf{z})$

DL methods: a survey

Deep Learning Techniques for Inverse Problems in Imaging

Gregory Ongie[*], Ajil Jalal[†], Christopher A. Metzler[‡]
Richard G. Baraniuk[§], Alexandros G. Dimakis[¶], Rebecca Willett[||]

April 2020

Abstract

Recent work in machine learning shows that deep neural networks can be used to solve a wide variety of inverse problems arising in computational imaging. We explore the central prevailing themes of this emerging area and present a taxonomy that can be used to categorize different problems and reconstruction methods. Our taxonomy is organized along two central axes: (1) whether or not a forward model is known and to what extent it is used in training and testing, and (2) whether or not the learning is supervised or unsupervised, i.e., whether or not the training relies on access to matched ground truth image and measurement pairs. We also discuss the tradeoffs associated with these different reconstruction approaches, caveats and common failure modes, plus open problems and avenues for future work.

Focuses on **linear** inverse problems, i.e., f linear

https://arxiv.org/abs/2005.06001

Limitations of middle ways:

- Representative data?
- Algorithm-sensitive
- Good initialization? (e.g., Manekar et al. **Deep Learning Initialized Phase Retrieval.** https://sunju.org/pub/NIPS20-WS-DL4FPR.pdf)

DL methods: the **economic (radical)** way

Deep image prior (DIP) $\mathbf{x} \approx G_\theta(\mathbf{z})$ G_θ (and $\mathbf{z}$) trainable

$$\min_{\mathbf{x}} \underbrace{\ell(\mathbf{y}, f(\mathbf{x}))}_{\text{data fitting}} + \lambda \underbrace{\mathrm{R}(\mathbf{x})}_{\text{regularizer}}$$

$$\Downarrow$$

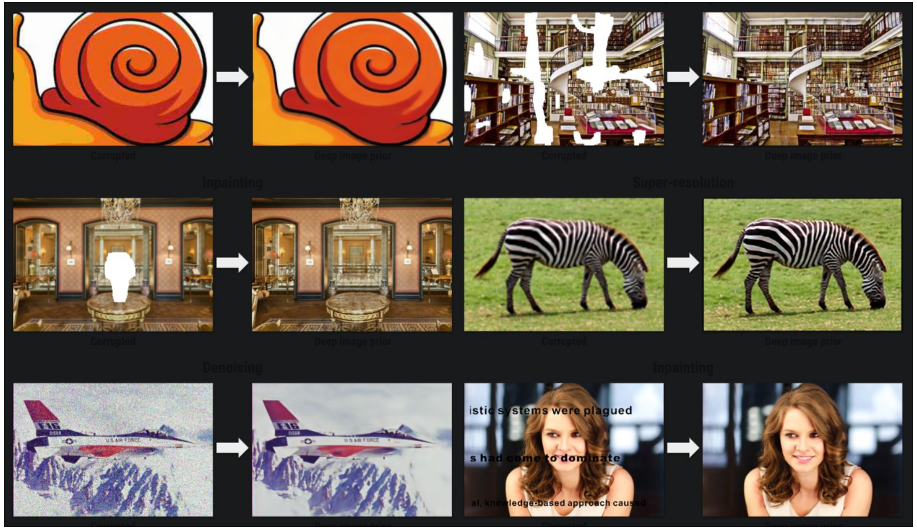
$$\min_{\theta} \ell(\mathbf{y}, f \circ G_\theta(\mathbf{z})) + \lambda R \circ G_\theta(\mathbf{z})$$

No extra training data!

Ulyanov et al. **Deep image prior**. IJCV'20. https://arxiv.org/abs/1711.10925

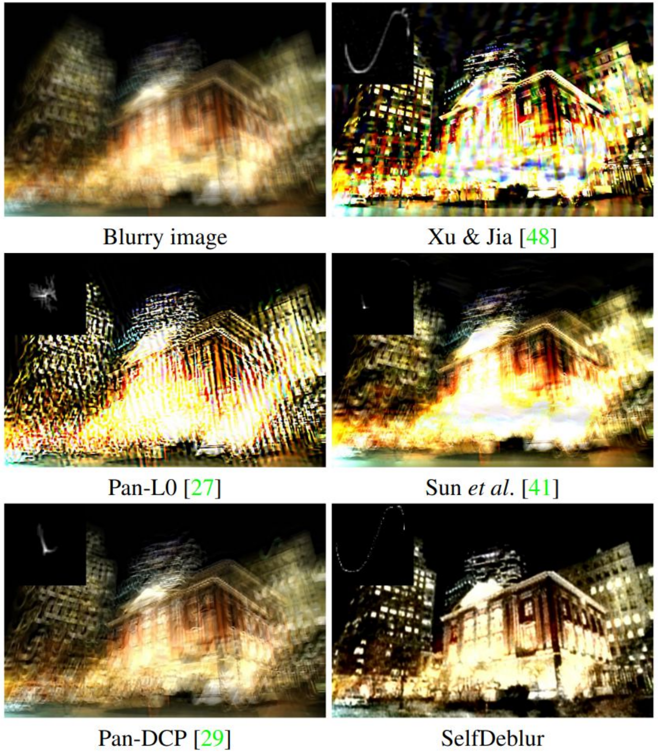
In other words, deep reparametrization

Successes of DIP



denoising/inpainting/super-resol/deJEPG/…

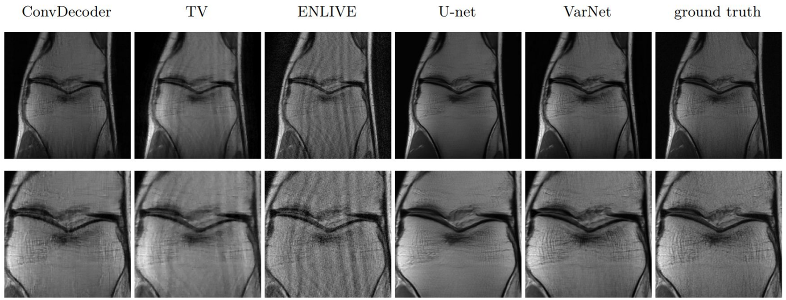
https://dmitryulyanov.github.io/deep_image_prior



Blind image deblurring (blind deconvolution)

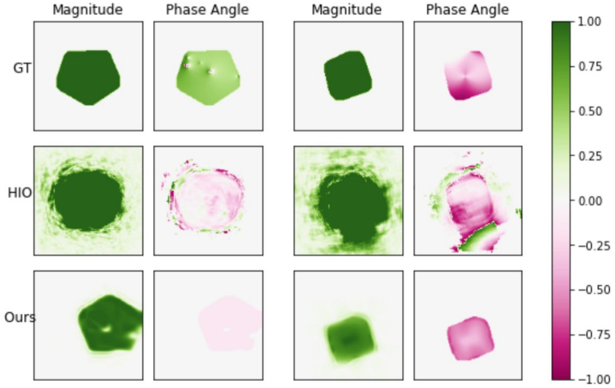
Ren et al. **Neural Blind Deconvolution Using Deep Priors**. CVPR'20. https://arxiv.org/abs/1908.02197

Zhuang et al. **Blind Image Deblurring with Unknown Kernel Size and Substantial Noise**. https://arxiv.org/abs/2208.09483



MRI reconstruction

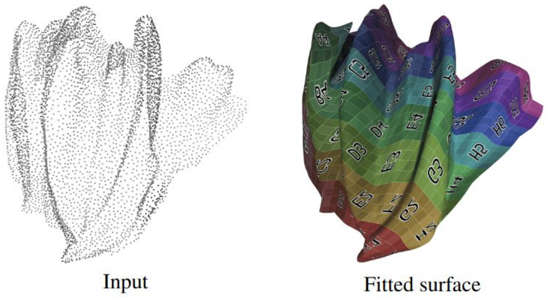
Darestani and Heckel. **Accelerated MRI with Un-trained Neural Networks.** https://arxiv.org/abs/2007.02471 (ConvDecoder is a variant of DIP)



Phase retrieval

Tayal et al. **Phase Retrieval using Single-Instance Deep Generative Prior.** https://arxiv.org/abs/2106.04812

Zhuang et al. **Practical Phase Retrieval Using Double Deep Image Priors.** https://arxiv.org/abs/2211.00799



Surface reconstruction

Williams et al. **Deep Geometric Prior for Surface Reconstruction.** CVPR'19. https://arxiv.org/abs/1811.10943

Many others:

- PET reconstruction
- Audio denoising
- Time series

See recent survey

Oayyum et al. **Untrained neural network priors for inverse imaging problems: A survey.** T-PAMI'22. https://ieeexplore.ieee.org/document/9878048

DIP's cousin(s)

Deep image prior (DIP)

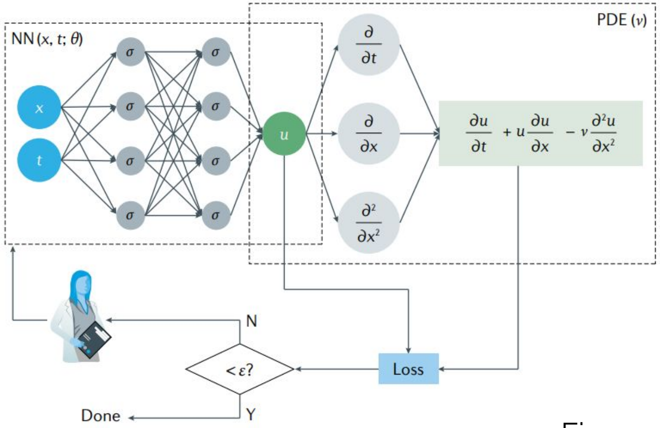
$\mathbf{x} \approx G_\theta(\mathbf{z})$ G_θ (and $\mathbf{z}$) trainable

Idea: (visual) objects as continuous functions

Neural implicit representation (NIR)

$\mathbf{x} \approx \mathcal{D} \circ \overline{\mathbf{x}}$ $\mathcal{D}$: discretization $\overline{\mathbf{x}}$: continuous function

Physics-informed neural networks (PINN)



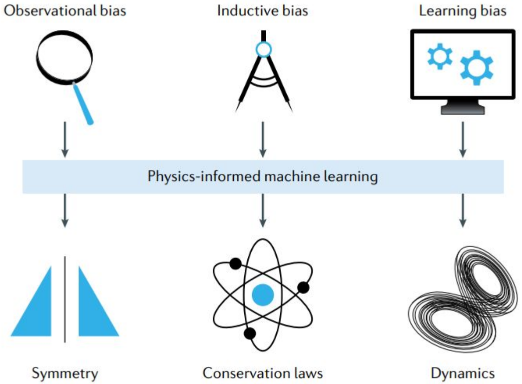
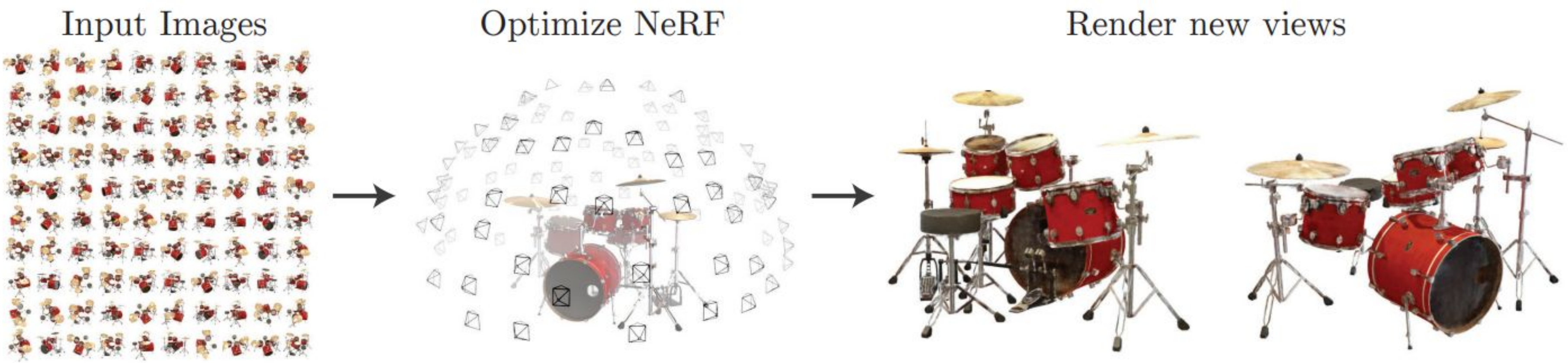
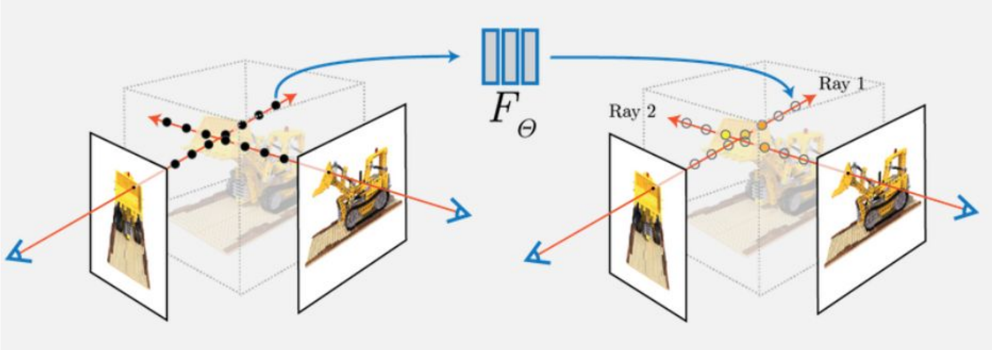
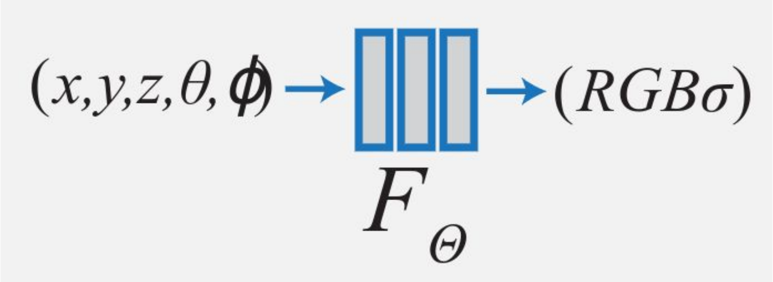


Figure credit: https://www.nature.com/articles/s42254-021-00314-5

NIR for 3D rendering and view synthesis

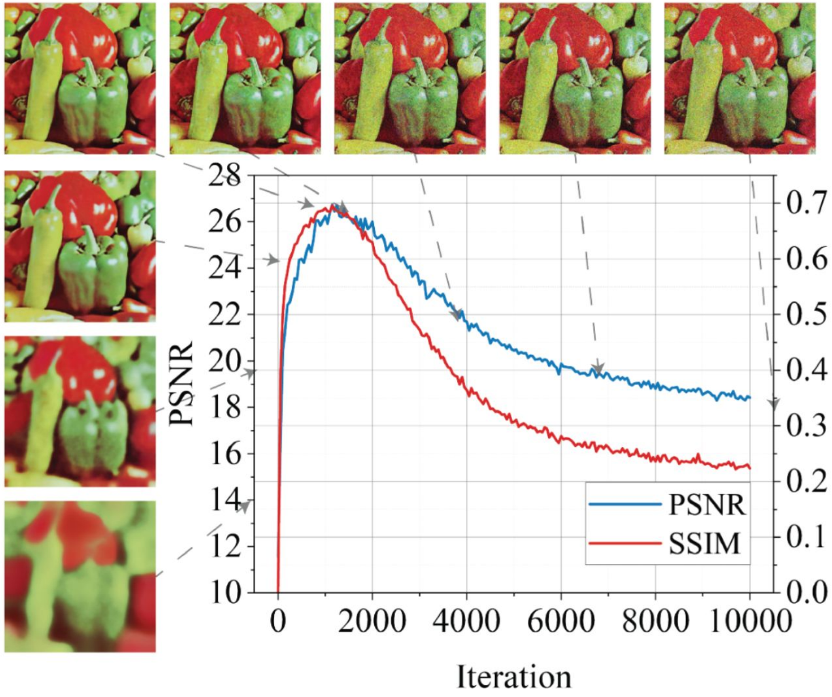


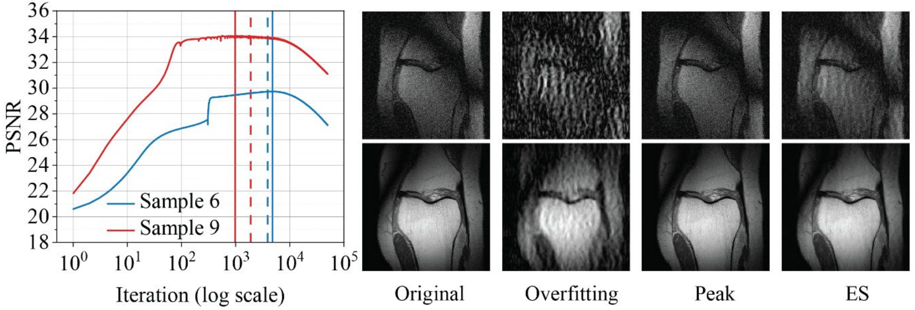




https://www.matthewtancik.com/nerf

Practical issues around DIP (and its cousin)





1) **Early learning then overfitting (ELTO)**
2) **Slow in convergence**
3) **Which G_θ?**
4) **Their niches?**

Our work

- Tackle early-learning-then-overfitting (ELTO) by **early stopping**
    - Li et al. **Self-Validation: Early Stopping for Single-Instance Deep Generative Priors** (BMVC'21) https://arxiv.org/abs/2110.12271
    - Wang et al. **Early Stopping for Deep Image Prior** https://arxiv.org/abs/2112.06074
- **Practical** blind image deblurring (BID) /**Practical** phase retrieval (PR)
    - Zhuang et al. **Blind Image Deblurring with Unknown Kernel Size and Substantial Noise**. https://arxiv.org/abs/2208.09483
    - Zhuang et al. **Practical Phase Retrieval Using Double Deep Image Priors.** https://arxiv.org/abs/2211.00799
- Toward **fast** computation for DIP
    - Li et al. **Deep Random Projector: Accelerated Deep Image Prior**. CVPR'23.

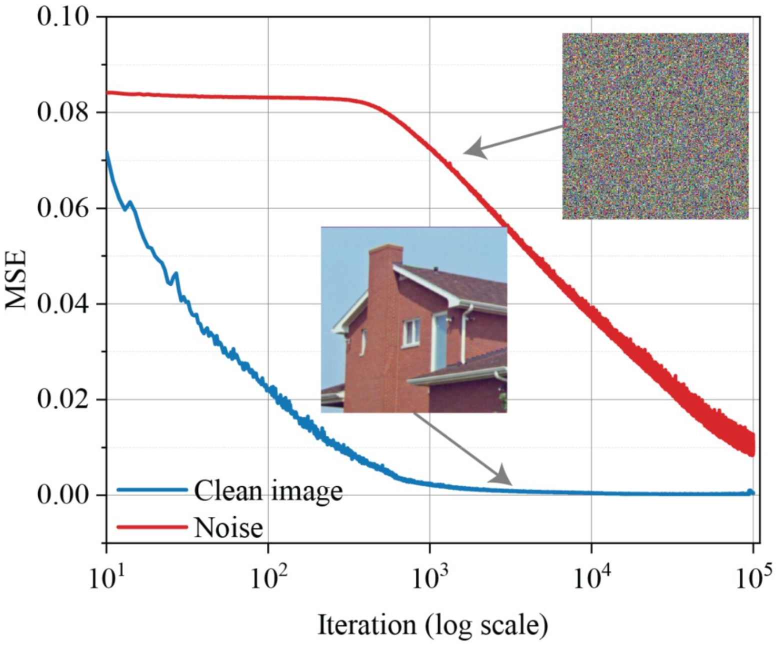
Early stopping for ELTO

- Li et al. **Self-Validation: Early Stopping for Single-Instance Deep Generative Priors** (BMVC'21) https://arxiv.org/abs/2110.12271
- Wang et al. **Early Stopping for Deep Image Prior** https://arxiv.org/abs/2112.06074

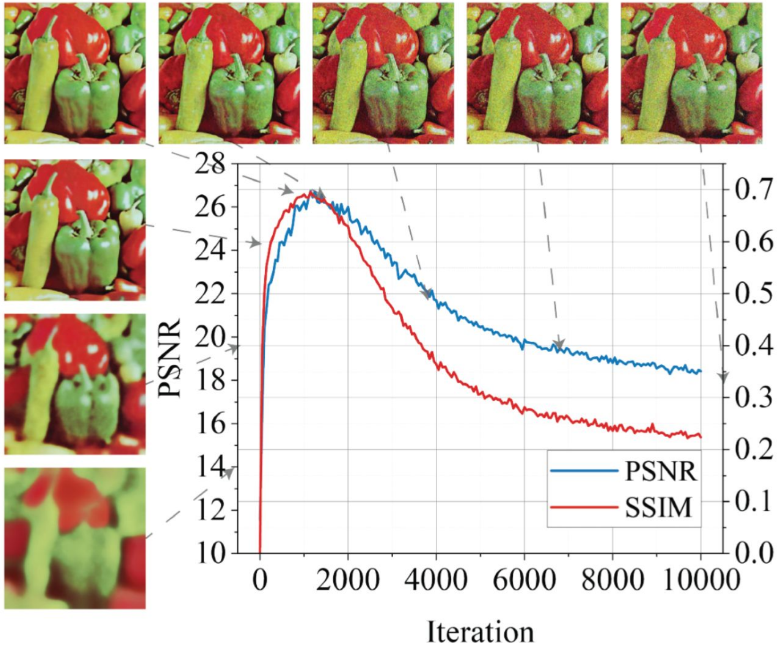
Why early-learning-then-overfitting (ELTO)?

$$\min_{\theta} \ell(\mathbf{y},\ f \circ G_{\theta}(\mathbf{z}))$$

DIP learns signal **much faster than** learning noise



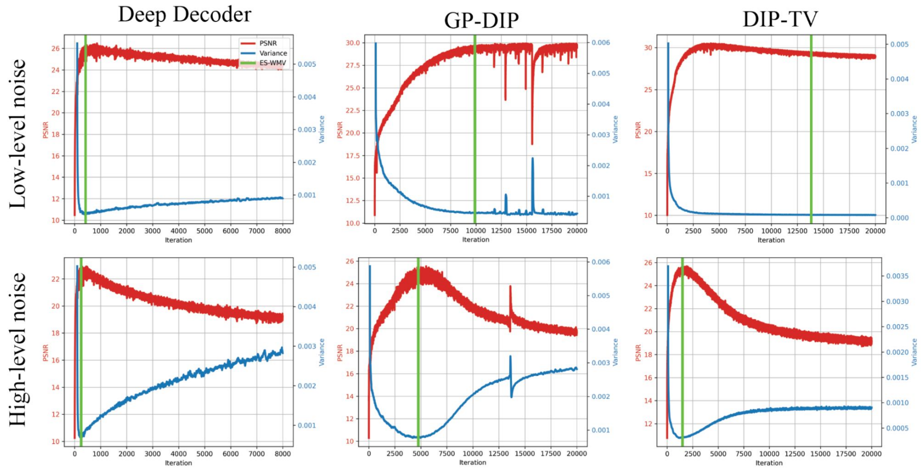
In practice, DIP heavily **over-parameterized**



Tackling ELTO via regularization

$$\min_{\theta} \ell(\mathbf{y}, f \circ G_\theta(\mathbf{z}))$$

- Regularize the network G_θ
- Regularize the estimation $G_\theta(\mathbf{z})$, i.e., bringing back $R \circ G_\theta(\mathbf{z})$



[Keckel & Hand'18] [Cheng et al'19] [Liu et al''18]

Cons: right regularization levels?

Detailed references: https://arxiv.org/abs/2112.06074

Tackling ELTO via noise modeling

- Noise modeling
  - Noise-specific regularizer
  - Explicit noise term

Double Over-parameterization:

$$\min_{\theta, \mathbf{g}, \mathbf{h}} \|\mathbf{y} - \phi(\theta) - (\mathbf{g} \circ \mathbf{g} - \mathbf{h} \circ \mathbf{h})\|_F^2$$

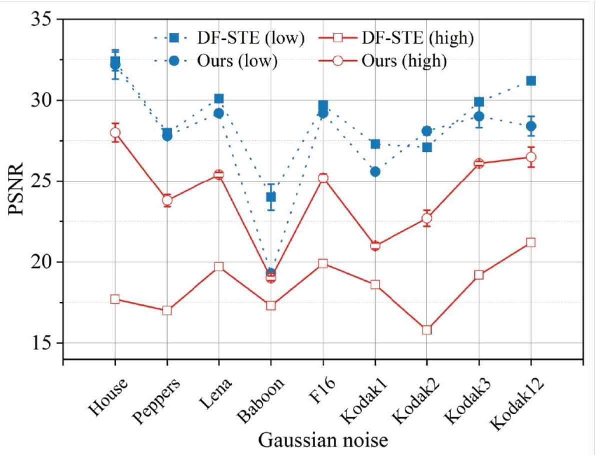
[You et al'20]

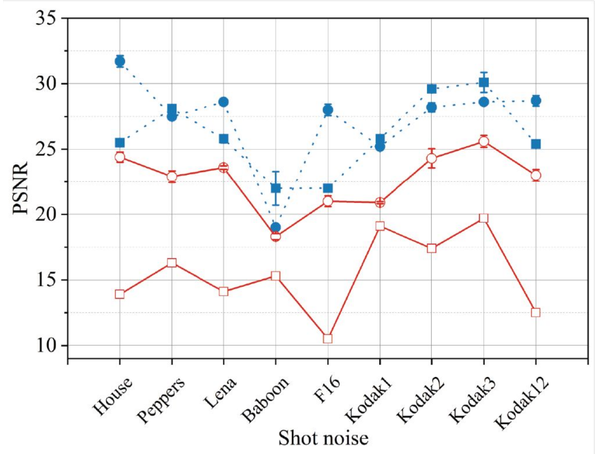
Rethinking DIP for denoising:

$$\eta(\mathbf{h}(\mathbf{y}), \mathbf{y}) = \mathcal{L}(\mathbf{y}, \mathbf{h}(\mathbf{y})) + \underbrace{\frac{2\sigma^2}{N} \sum_{i=1}^{N} \frac{\partial \mathbf{h}_i(\mathbf{y})}{\partial (\mathbf{y})_i}}_{\text{divergence term}} - \sigma^2$$

[Jo et al'21]

Cons: need detailed noise info



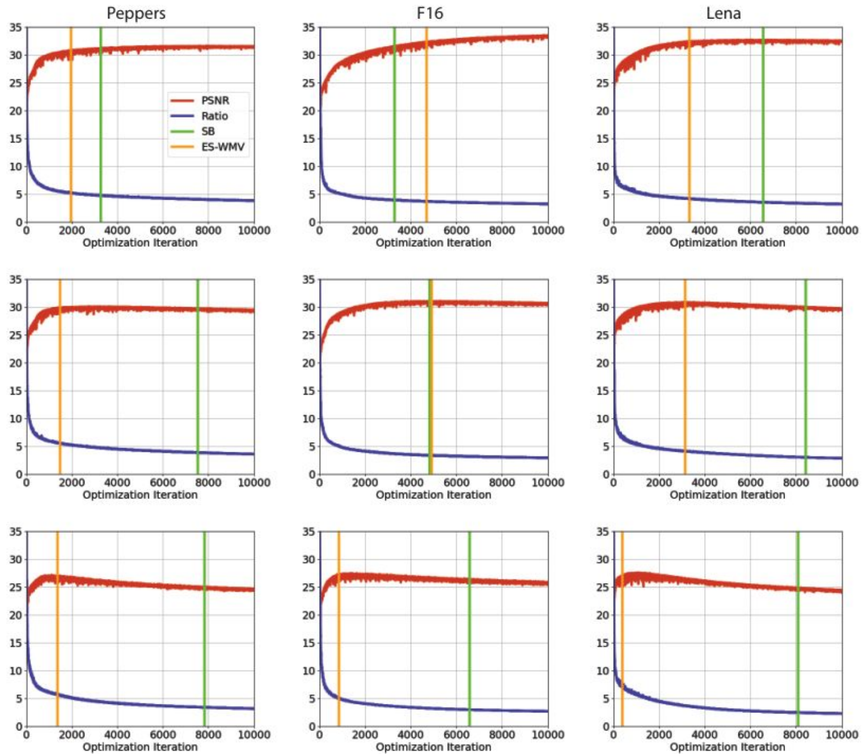


Detailed references: https://arxiv.org/abs/2112.06074

Tackling ELTO via early stopping

Cons: model- or noise-specific

[Shi et al'21]



Low-level noise

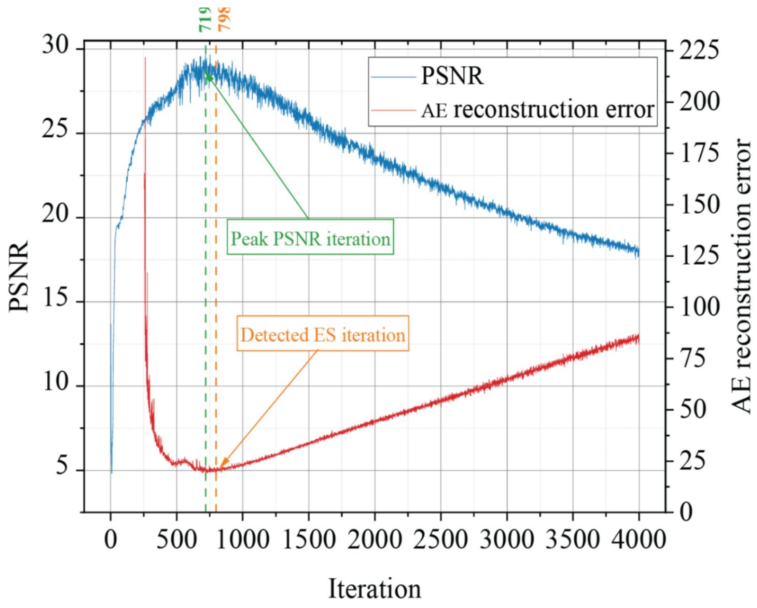
Medium-level noise

High-level noise

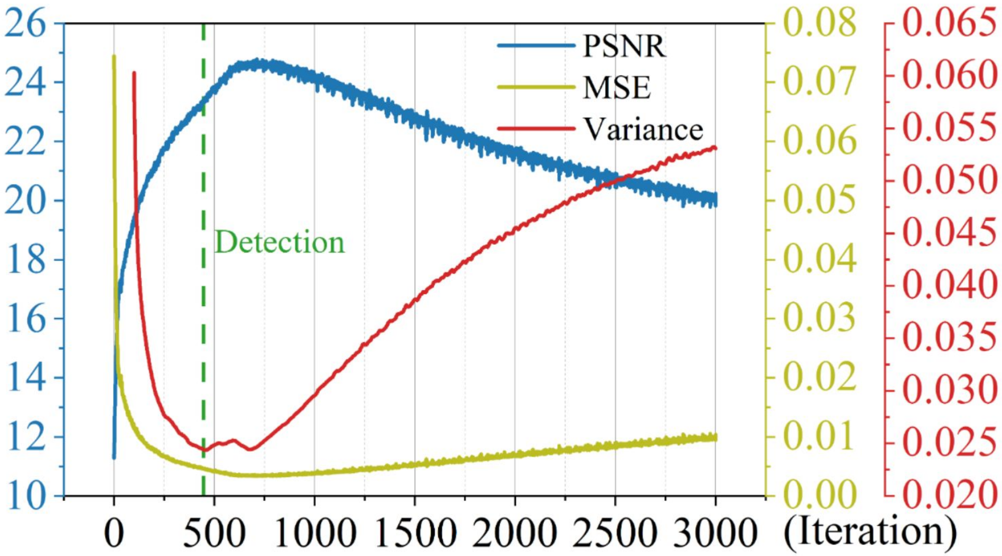
On Measuring and Controlling the Spectral Bias of the Deep Image Prior. https://link.springer.com/article/10.1007/s11263-021-01572-7

Detailed references: https://arxiv.org/abs/2112.06074

An interesting observation



ES Ver 1.0: based on autoencoder Rec Err



ES Ver 2.0: based on running variance

ES base on moving variance (MV)

Algorithm 1 DIP with ES–WMV

Input: random seed $\boldsymbol{z}$, randomly-initialized $G_{\boldsymbol{\theta}}$, window size W, patience number P, empty queue $\mathcal{Q}$, iteration counter $k = 0$

Output: reconstruction $\boldsymbol{x}^*$

while not stopped **do**
  update $\boldsymbol{\theta}$ via Eq. (2) to obtain $\boldsymbol{\theta}^{k+1}$ and $\boldsymbol{x}^{k+1}$
  push $\boldsymbol{x}^{k+1}$ to $\mathcal{Q}$, pop queue front if $|\mathcal{Q}| > W$
  if $|\mathcal{Q}| = W$ **then**
    calculate VAR of elements in $\mathcal{Q}$
    update $\mathrm{VAR}_{\min}$ and the corresponding $\boldsymbol{x}^*$
    if no decrease of $\mathrm{VAR}_{\min}$ in P consecutive iterations **then**
      stop and return $\boldsymbol{x}^k$
    end if
  end if
  $k = k + 1$
end while

Algorithm 2 DIP with ES–EMV

Input: random seed $\boldsymbol{z}$, randomly-initialized $G_{\boldsymbol{\theta}}$, forgetting factor $\alpha \in (0, 1)$, patience number P, iteration counter $k = 0$, $\mathrm{EMA}^0 = 0$, $\mathrm{EMV}^0 = 0$,

Output: reconstruction $\boldsymbol{x}^*$

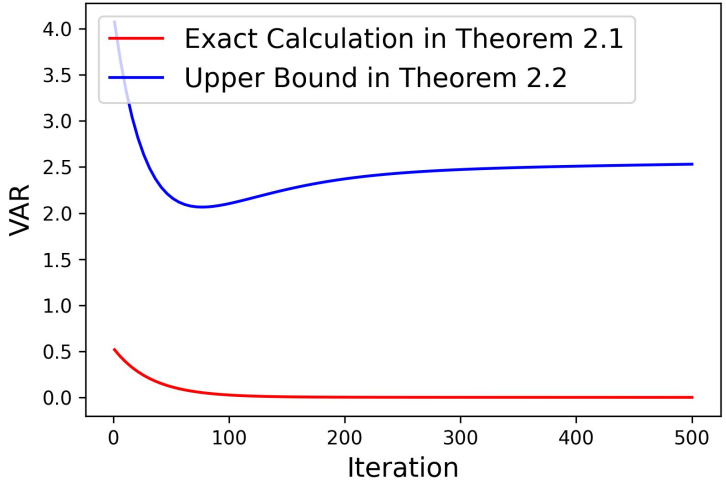
while not stopped **do**
  update $\boldsymbol{\theta}$ via Eq. (2) to obtain $\boldsymbol{\theta}^{k+1}$ and $\boldsymbol{x}^{k+1}$
  $\mathrm{EMA}^{k+1} = (1 - \alpha)\mathrm{EMA}^k + \alpha \boldsymbol{x}^{k+1}$
  $\mathrm{EMV}^{k+1} = (1 - \alpha)\mathrm{EMV}^k + \alpha(1 - \alpha)\|\boldsymbol{x}^{k+1} - \mathrm{EMA}^k\|_2^2$
  update $\mathrm{EMV}_{\min}$ and the corresponding $\boldsymbol{x}^*$
  if no decrease of $\mathrm{EMV}_{\min}$ in P consecutive iterations **then**
    stop and return $\boldsymbol{x}^k$
  end if
  $k = k + 1$
end while

Table 5. Wall-clock time of DIP, SV-ES, ES-WMV and ES-EMV per epoch on *NVIDIA Tesla K40 GPU*: mean and (std).

	DIP	SV-ES	ES-WMV	ES-EMV
Time(secs)	0.448 (0.030)	13.027 (3.872)	0.301 (0.016)	0.003 (0.003)

Very little overhead

A bit of justification



Theorem 2.1. *Let σ_i's and $\boldsymbol{w}_i$'s be the singular values and left singular vectors of $\boldsymbol{J}_G(\boldsymbol{\theta}^0)$, and suppose we run gradient descent with step size η on the linearized objective $\hat{f}(\boldsymbol{\theta})$ to obtain $\{\boldsymbol{\theta}^t\}$ and $\{\boldsymbol{x}^t\}$ with $\boldsymbol{x}^t \doteq G_{\boldsymbol{\theta}^0}(\boldsymbol{z}) + \boldsymbol{J}_G(\boldsymbol{\theta}^0)(\boldsymbol{\theta}^t - \boldsymbol{\theta}^0)$. Then provided that $\eta \le 1/\max_i(\sigma_i^2)$, the running variance of $\{\boldsymbol{x}^t\}$ is*

$$\mathrm{DISP}_2^2(t) = \sum_i C_{m,\eta,\sigma_i} \langle \boldsymbol{w}_i, \widehat{\boldsymbol{y}} \rangle^2 \left(1 - \eta\sigma_i^2\right)^{2t}, \qquad (7)$$

where $\widehat{\boldsymbol{y}} = \boldsymbol{y} - G_{\boldsymbol{\theta}^0}(\boldsymbol{z})$, and $C_{W,\eta,\sigma_i} \ge 0$ only depends on W, η, and σ_i for all i.

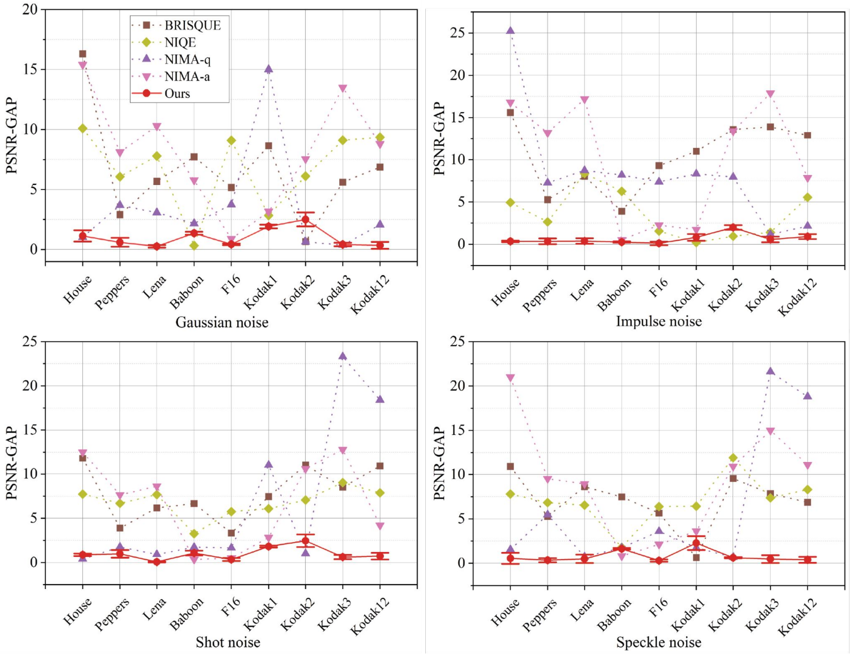
Theorem 2.2. *Assume the same setting as Theorem 2 of [16]. Our WMV is upper bounded by*

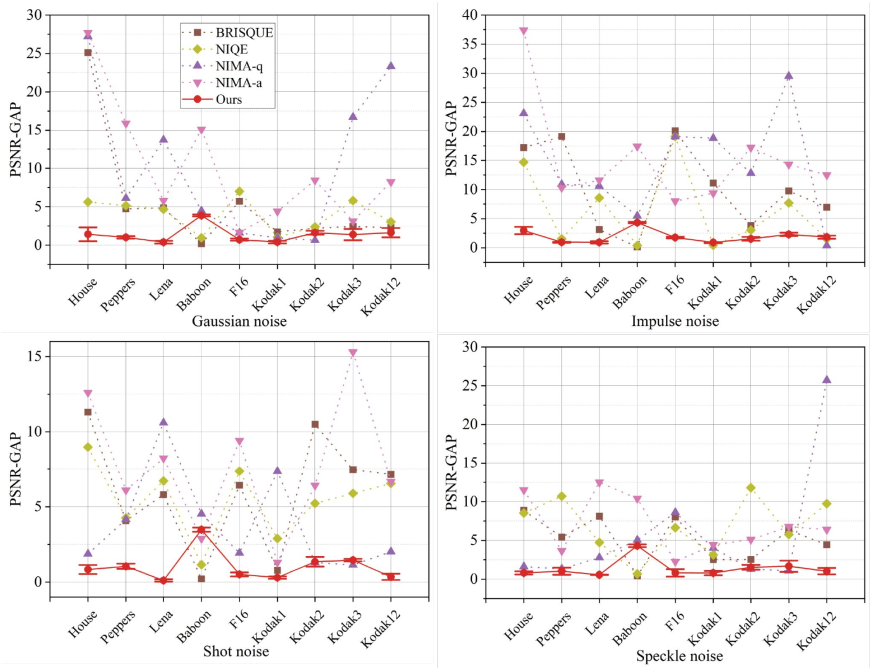
$$\frac{12}{W}\|\boldsymbol{x}\|_2^2 \frac{\left(1-\eta\sigma_p^2\right)^{2t}}{1-(1-\eta\sigma_p^2)^2} +$$

$$12\sum_{i=1}^{n}\left(\left(1-\eta\sigma_i^2\right)^{t+W-1} - 1\right)^2 (\boldsymbol{w}_i^\intercal \boldsymbol{n})^2 + 12\varepsilon^2\|\boldsymbol{y}\|_2^2.$$

with high probability.

Effective across types\levels of noise





High-Level

Low-Level

Typical detection gap: around 1 PSNR point

Effective on real-world denoising

NTIRE 2020 Real Image Denoising Challenge (RGB track) for **1024** Images

- Unknown noise types and levels

Table 7. ES-WMV on real image denoising: mean and (std).

	Detected PSNR	PSNR Gap	Detected SSIM	SSIM Gap
DIP (MSE)	34.04 (3.68)	0.92 (0.83)	0.92 (0.07)	0.02 (0.04)
DIP (ℓ_1)	33.92 (4.34)	0.92 (0.59)	0.93 (0.05)	0.02 (0.02)
DIP (Huber)	33.72 (3.86)	0.95 (0.73)	0.92 (0.06)	0.02 (0.03)

Effective on advanced tasks

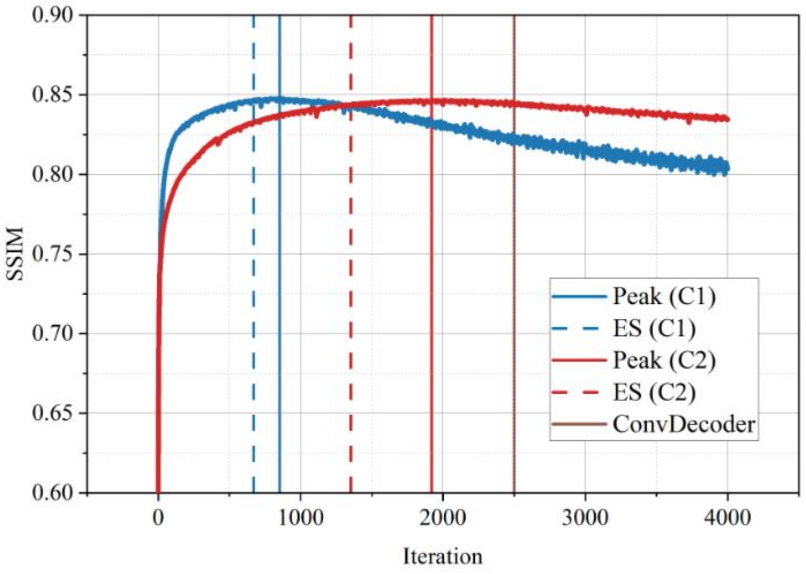
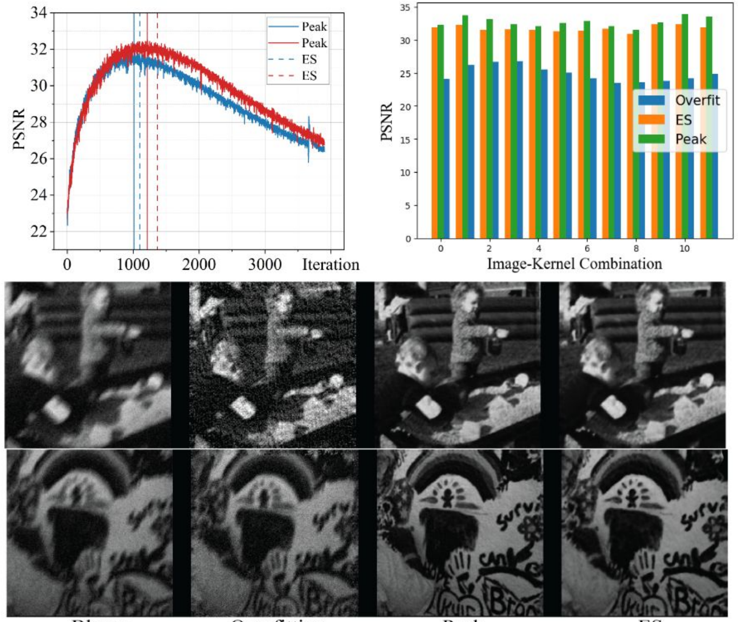


Figure 5. Detection performance on MRI reconstruction



Code available at: https://github.com/sun-umn/Early_Stopping_for_DIP

Toward practical blind image deblurring

- Zhuang et al. **Blind Image Deblurring with Unknown Kernel Size and Substantial Noise**. https://arxiv.org/abs/2208.09483

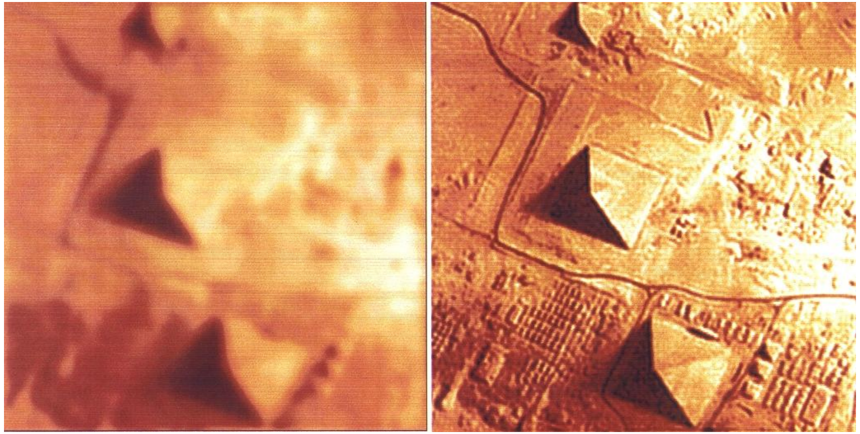
Blind image deblurring (BID)

$$\underbrace{\mathbf{y}}_{\textbf{blurry and noisy image}} = \overbrace{\mathbf{k}}^{\textbf{blur kernel}} * \underbrace{\mathbf{x}}_{\textbf{clean image}} + \overbrace{\mathbf{n}}^{\textbf{noise}}$$

Mostly due to optical deficiencies (e.g., defocus) and motions

Given $\mathbf{y}$,
recover $\mathbf{x}$ (and/or $\mathbf{k}$)

Also **Blind Deconvolution**



Landmark surveys

- 1996: Kundur and Hatzinakos. **Blind image deconvolution**. https://doi.org/10.1109/79.489268
- 2011: Levin et al. **Understanding blind deconvolution algorithms**. https://doi.org/10.1109/TPAMI.2011.148
- 2012: Kohler et al. **Recording and playback of camera shake: Benchmarking blind deconvolution with a real-world database**. https://doi.org/10.1007/978-3-642-33786-4_3
- 2016: Lai et al. **A comparative study for single image blind deblurring**. https://doi.org/10.1109/CVPR.2016.188
- 2021: Koh et al. **Single image deblurring with neural networks: A comparative survey** https://doi.org/10.1016/j.cviu.2020.103134
- 2022: Zhang et al. **Deep image blurring: A survey** https://doi.org/10.1007/s11263-022-01633-5

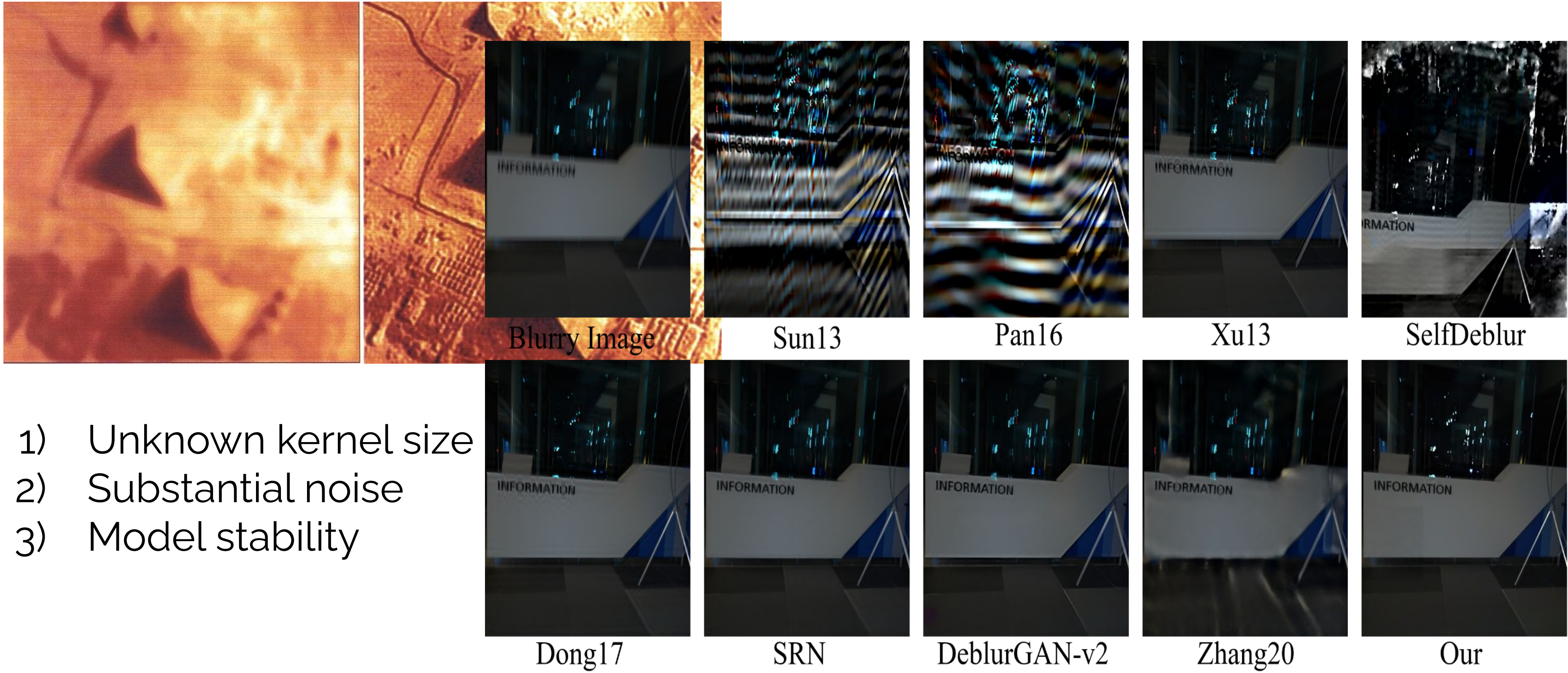
See also: **Awesome Deblurring**
https://github.com/subeeshvasu/Awesome-Deblurring

Key challenge of data-driven approach:

obtaining sufficiently expressive data (Koh et al'21. Zhang et al'22)

Practicality challenges

$$\underbrace{\mathbf{y}}_{\textbf{blurry and noisy image}} = \overbrace{\mathbf{k}}^{\textbf{blur kernel}} * \underbrace{\mathbf{x}}_{\textbf{clean image}} + \overbrace{\mathbf{n}}^{\textbf{noise}}$$



1) Unknown kernel size
2) Substantial noise
3) Model stability

Double DIPs

$$\underbrace{\mathbf{y}}_{\text{blurry and noisy image}} = \overbrace{\mathbf{k}}^{\text{blur kernel}} * \underbrace{\mathbf{x}}_{\text{clean image}} + \overbrace{\mathbf{n}}^{\text{noise}}$$

$$\min_{\boldsymbol{k},\boldsymbol{x}} \underbrace{\ell(\boldsymbol{y}, \boldsymbol{k} * \boldsymbol{x})}_{\text{data fitting}} + \lambda_{\boldsymbol{k}} \underbrace{R_{\boldsymbol{k}}(\boldsymbol{k})}_{\text{regularizing } \boldsymbol{k}} + \lambda_{\boldsymbol{x}} \underbrace{R_{\boldsymbol{x}}(\boldsymbol{x})}_{\text{regularizing } \boldsymbol{x}}$$

Idea: parameterize both $\mathbf{k}$ and $\mathbf{x}$ as DIPs

- CNN + CNN (Wang et al'19, https://doi.ieeecomputersociety.org/10.1109/ICCVW.2019.00127; Tran et al'21, https://arxiv.org/abs/2104.00317)
- MLP + CNN (SelfDeblur; Ren et al'20, https://arxiv.org/abs/1908.02197)

Still problematic with

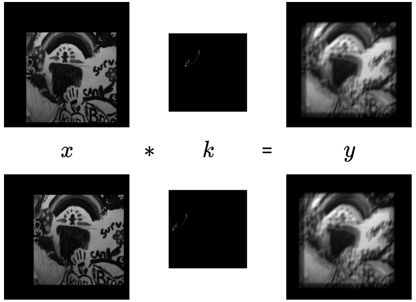
1) kernel size over-specification 2) substantial noise

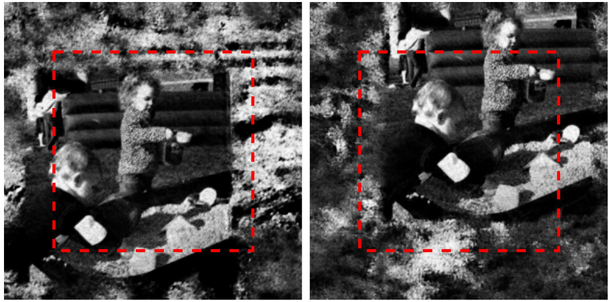
A glance of our modifications

Over-specify $\mathbf{k}$
Over-specify $\mathbf{x}$

$\mathbf{k}$ **~half of the image sizes**

Handle bounded shift





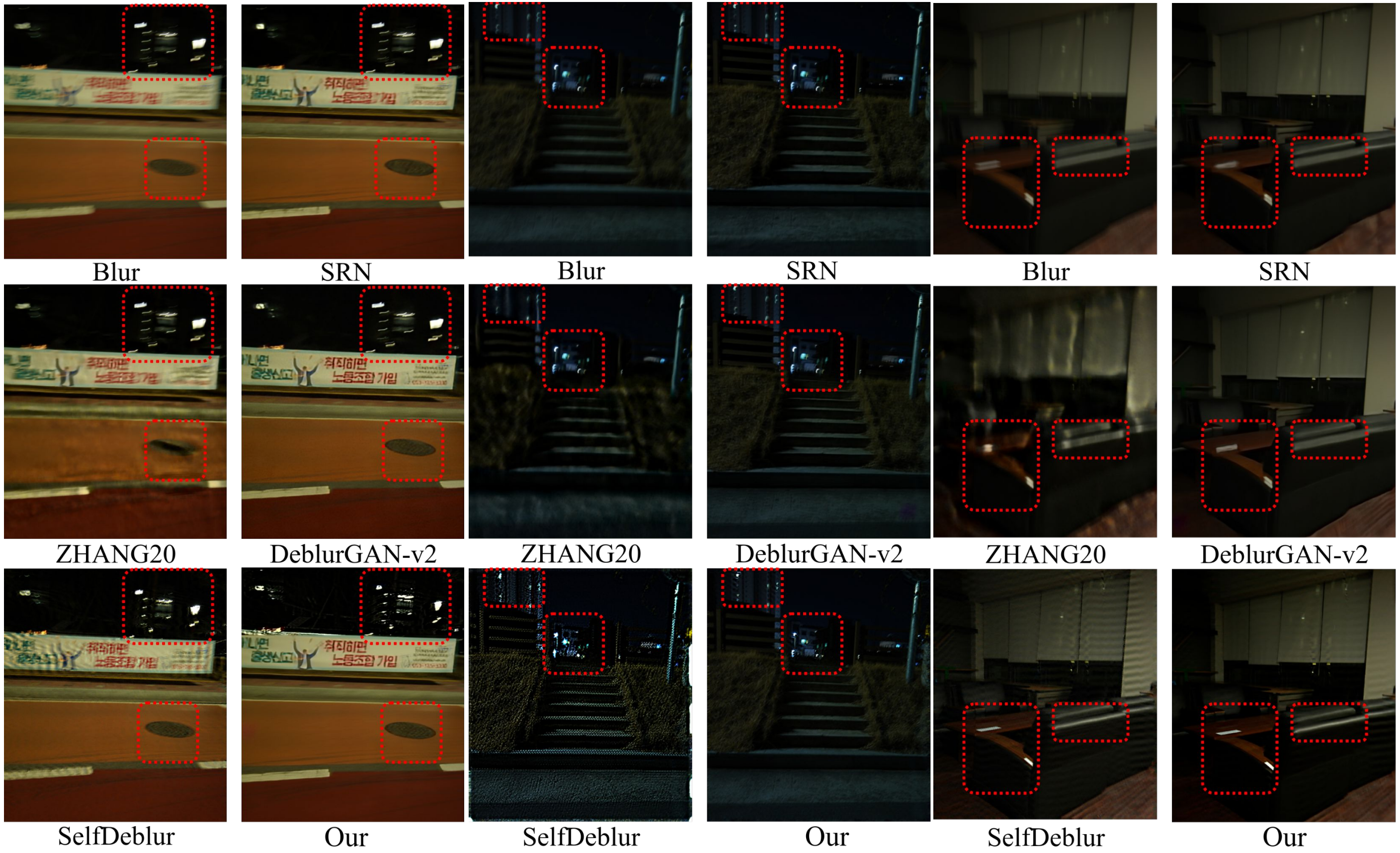
$$\min_{\boldsymbol{\theta}_k, \boldsymbol{\theta}_x} \|\boldsymbol{y} - G_{\boldsymbol{\theta}_k}(\boldsymbol{z}_k) * G_{\boldsymbol{\theta}_x}(\boldsymbol{z}_x)\|_2^2 + \lambda \frac{\|\nabla G_{\boldsymbol{\theta}_x}(\boldsymbol{z}_x)\|_1}{\|\nabla G_{\boldsymbol{\theta}_x}(\boldsymbol{z}_x)\|_2}$$

ℓ_1/ℓ_2 vs ℓ_1

Table 1: ℓ_1/ℓ_2 vs TV for noise: mean and (std).

	Low Level		High Level	
	PSNR	λ	PSNR	λ
$\frac{L1}{L2}$	32.64 (0.69)	0.0001 (0.018)	27.74 (0.23)	0.0002 (0.0019)
TV	31.12 (0.52)	0.002 (0.07)	24.34 (0.78)	0.02 (0.10)

Real world results



Difficult cases

1) High depth contrast
2) High brightness contrast

Outperform SOTA data-driven methods!

Closing

$$\min_{\theta} \ell(\mathbf{y}, f \circ G_{\theta}(\mathbf{z})) + \lambda R \circ G_{\theta}(\mathbf{z})$$

Addressing practicality issues around DIP

- **Early stopping** to tackle early-learning-then-overfitting (ELTO)
- Careful customization makes **blind image denoising** and **phase retrieval** work in unprecedented regimes
- (brief) **Deep random projector**—toward efficient DIP

Papers

- Li et al. **Self-Validation: Early Stopping for Single-Instance Deep Generative Priors** (BMVC'21) https://arxiv.org/abs/2110.12271
- Wang et al. **Early Stopping for Deep Image Prior** https://arxiv.org/abs/2112.06074 (Under review for ICLR'23)
- Zhuang et al. **Blind Image Deblurring with Unknown Kernel Size and Substantial Noise**. https://arxiv.org/abs/2208.09483 (Under review for IJCV)
- Zhuang et al. **Practical Phase Retrieval Using Double Deep Image Priors**. https://arxiv.org/abs/2211.00799 (Electronic Imaging'23)
- Li et al. **Deep Random Projector: Toward Efficient Deep Image Prior**. (CVPR'23)