Topics in Numerical Analysis II
Computational Inverse Problems

Bangti Jin (b.jin@cuhk.edu.hk)

Chinese University of Hong Kong

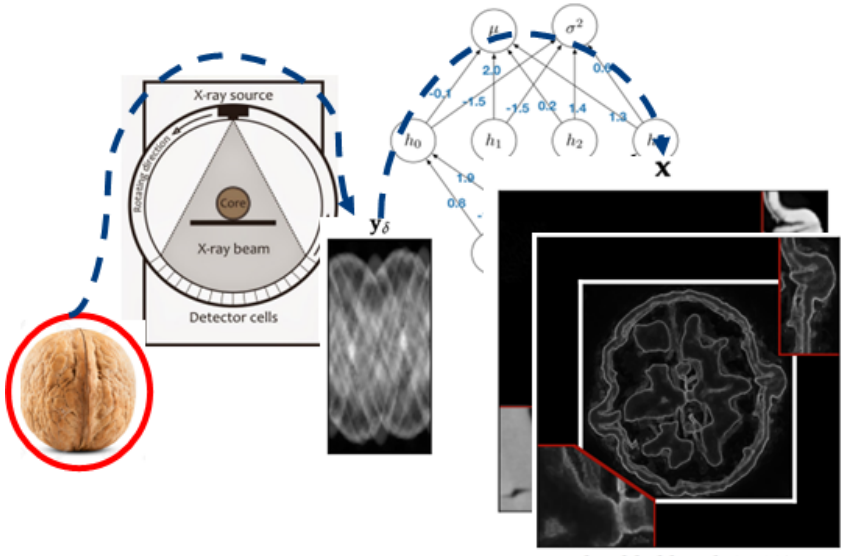
November 11, 2024

Outline

model setting: linear inverse problem

$$y^\delta = Ax + \eta$$

- x: unknown image / signal
- y^δ: noisy measurements
- η: data noise
- A: linear forward operator



The problem is often ill-posed

Traditionally, it is estimated through **regularized reconstruction**.
Can we design a **neural solver** with quantified uncertainty?

Deep learning for inverse problems

- there are many successful supervised approaches using deep learning for solving inverse problems.

$$\theta^* \in \arg\min \sum_i \|f_\theta(y^i) - x^i\|^2, \quad x^q = f_{\theta^*}(y^q)$$

  - unrolling (ISTA, ADMM, GD, PDHG, EM, ...)
  - learned prior / likelihood
  - ...

features

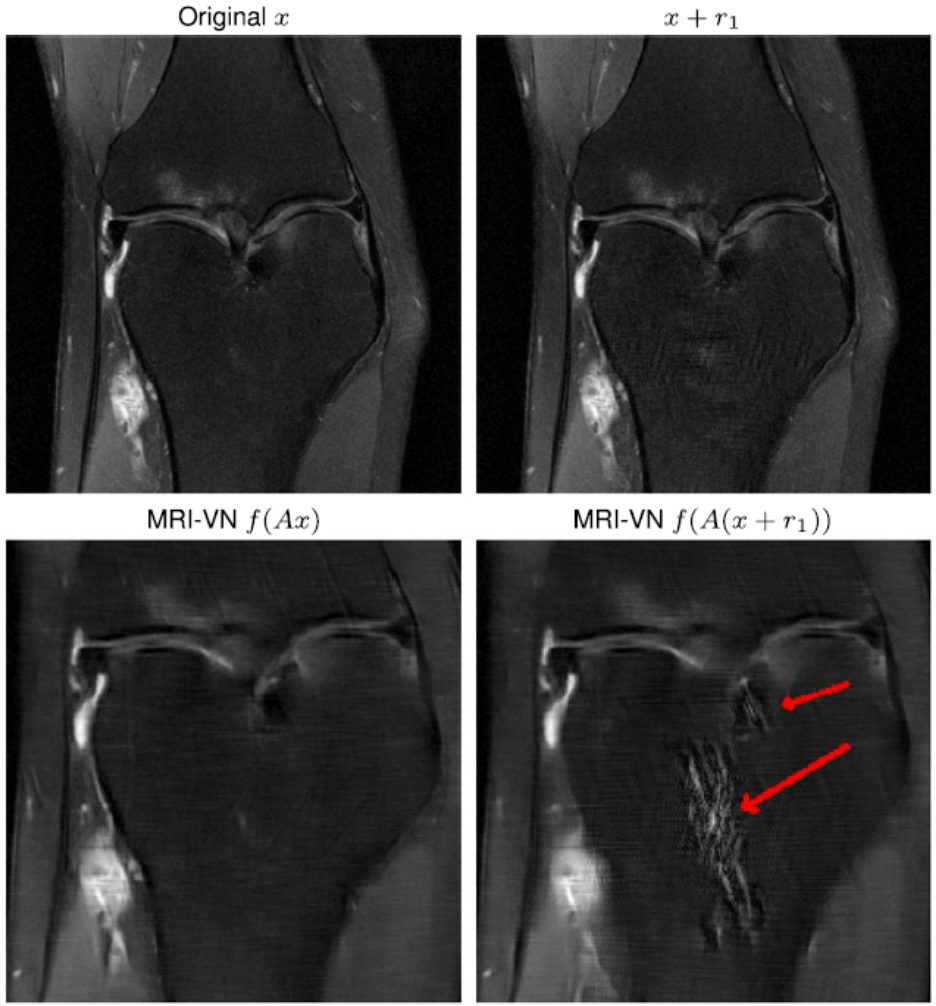
- cheap at deployment (passing through the network)
- require (**abundant**) paired training data $\{(x^i, y^i)\}_{i=1}^N$ (scarce in medical imaging)
- performance may **degrade significantly** when the test data distributes slightly differently from the training data! V Antun, F Renna, C Poon, B Adcock, A C Hansen. PNSA 2020

The peril of training data

sensitivity of classification neural networks


K. Eykholt et al CVPR 2018

The peril of training data V. Antun, F. Renna, C. Poon, B. Adcock, A. C. Hansen. PNAS 2019



Deep image prior

inversion tasks via energy minimization

$$x^* = \arg\min_x E(x; x_0) + R(x)$$

- x_0: corrupted image
- $E(x; x_0)$: task specific term (penalty)

deep image prior: $x = f_\theta(z)$ maps a random code vector z to an image x, and no explicit prior $R(x)$ Ulyanov-Vedaldi-Lempitsky 2018

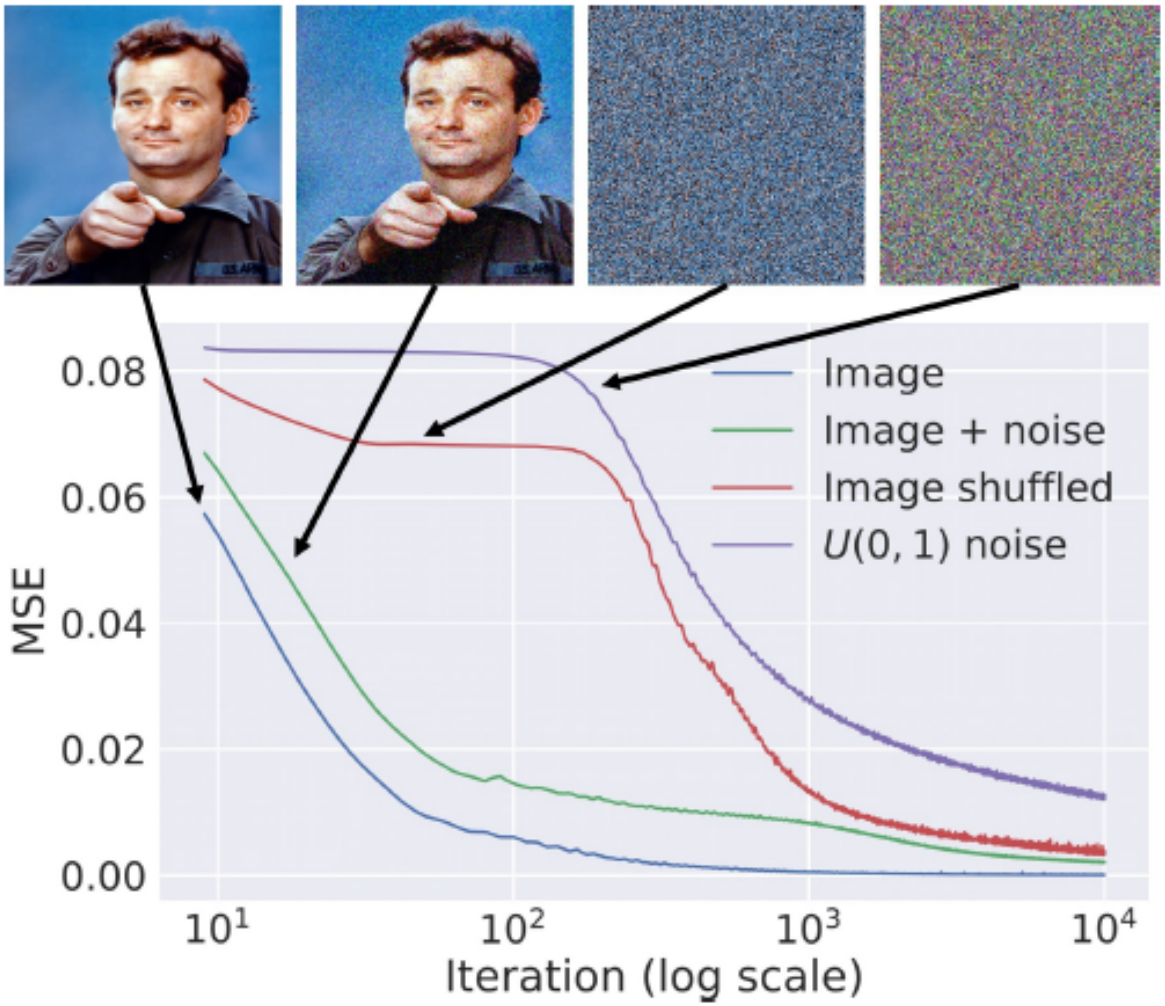
$$\theta^* = \arg\min_\theta E(f_\theta(z); x_0), \quad x^* = f_{\theta^*}(z).$$

procedure: θ^* computed by randomly initialized gradient descent, converges to a local minimizer

deep image prior: D Ulyanov, A Vedaldi, V Lempitsky CVPR 2018, 9446–9454

$$\theta^* = \arg\min_{\theta \in \mathbb{R}^{d_\theta}} \|A f_\theta(z) - y^\delta\|^2, \quad x^* = f_{\theta^*}(z)$$

z: random fixed input, θ: DNN parameters
insight: architecture alone can regularize inverse problems in imaging
...


0.08
0.06
0.04
0.02
0.00
MSE
10^1
10^2
10^3
10^4
Iteration (log scale)
Image
Image + noise
Image shuffled
$U(0, 1)$ noise

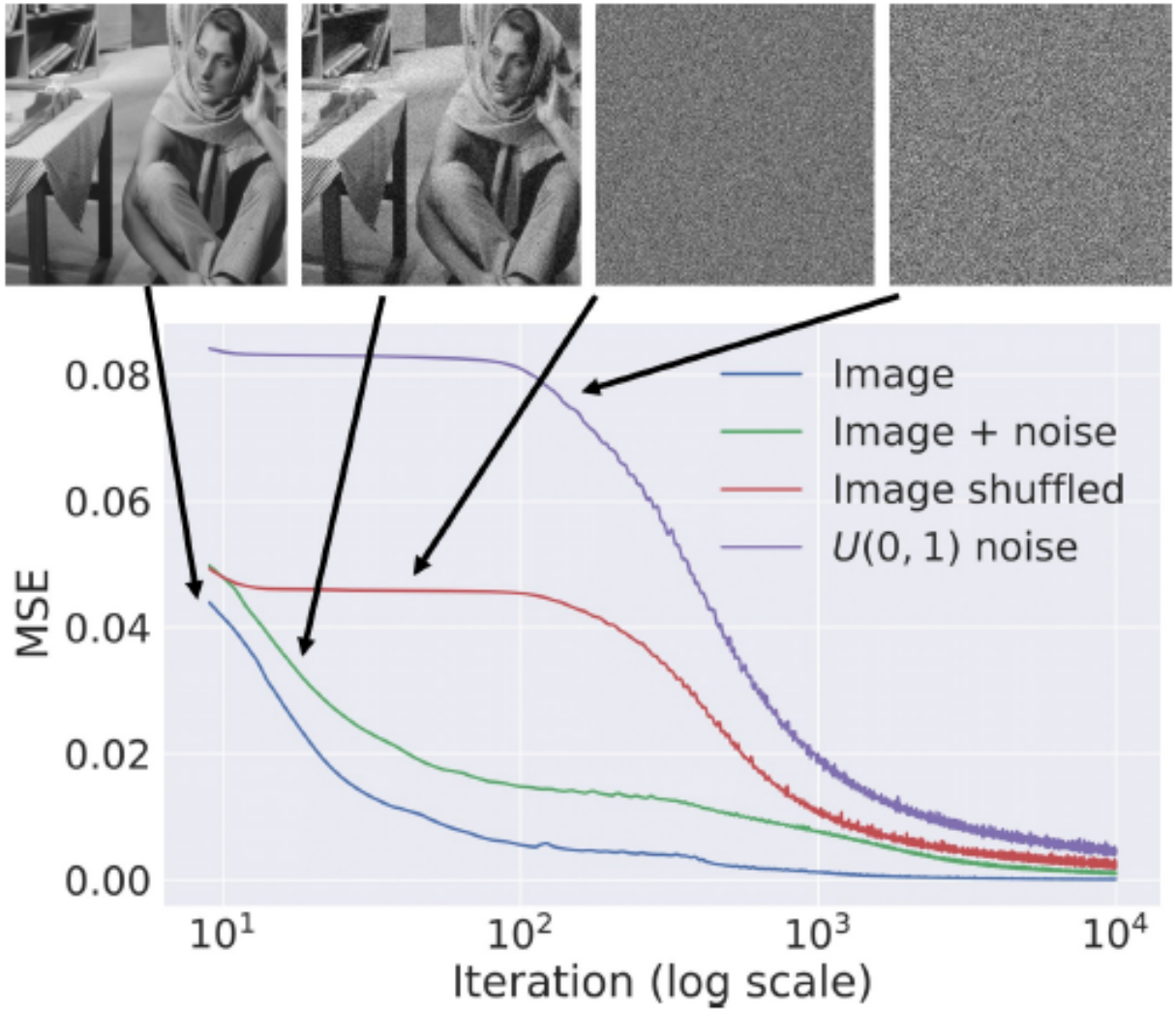
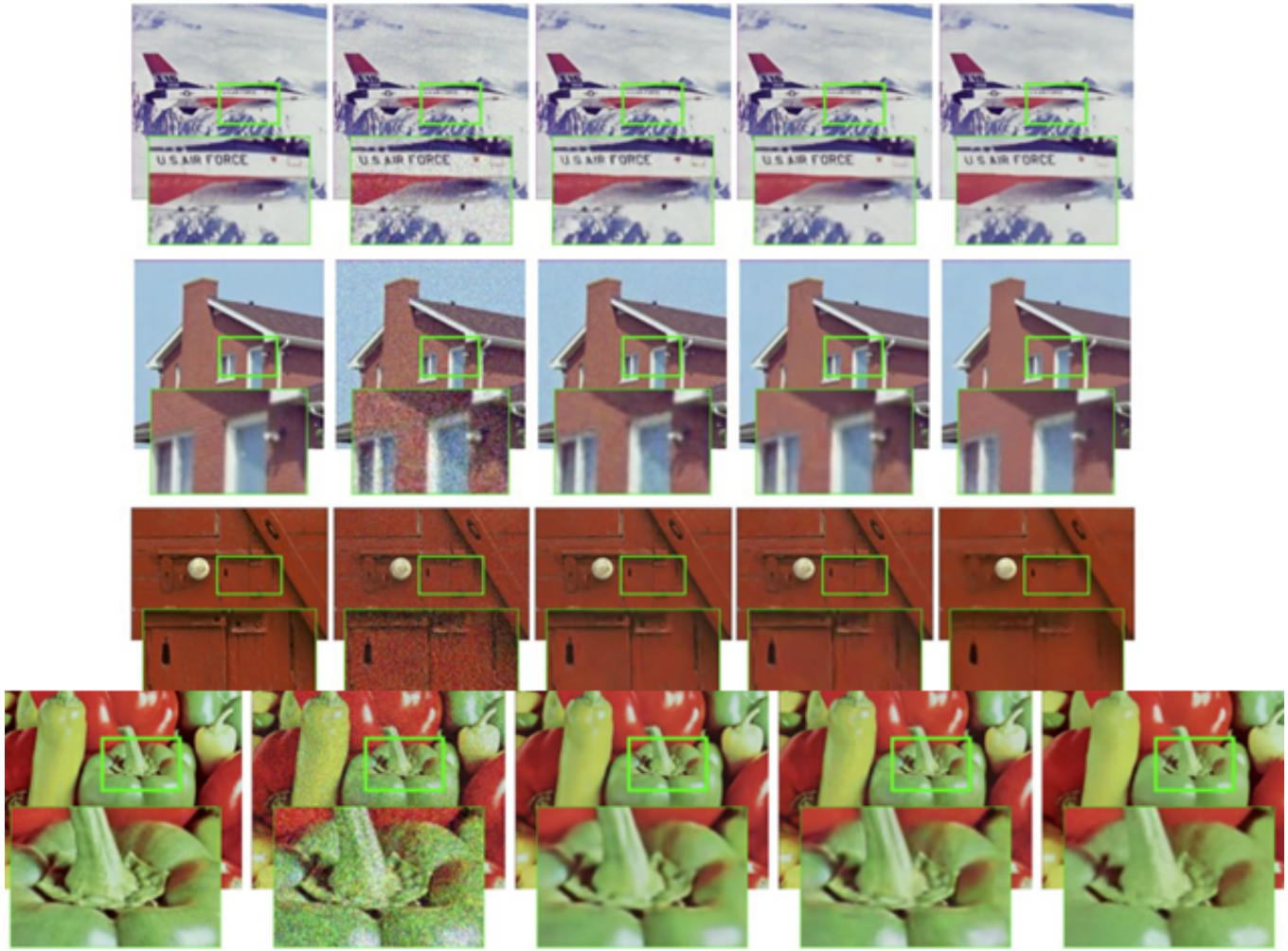
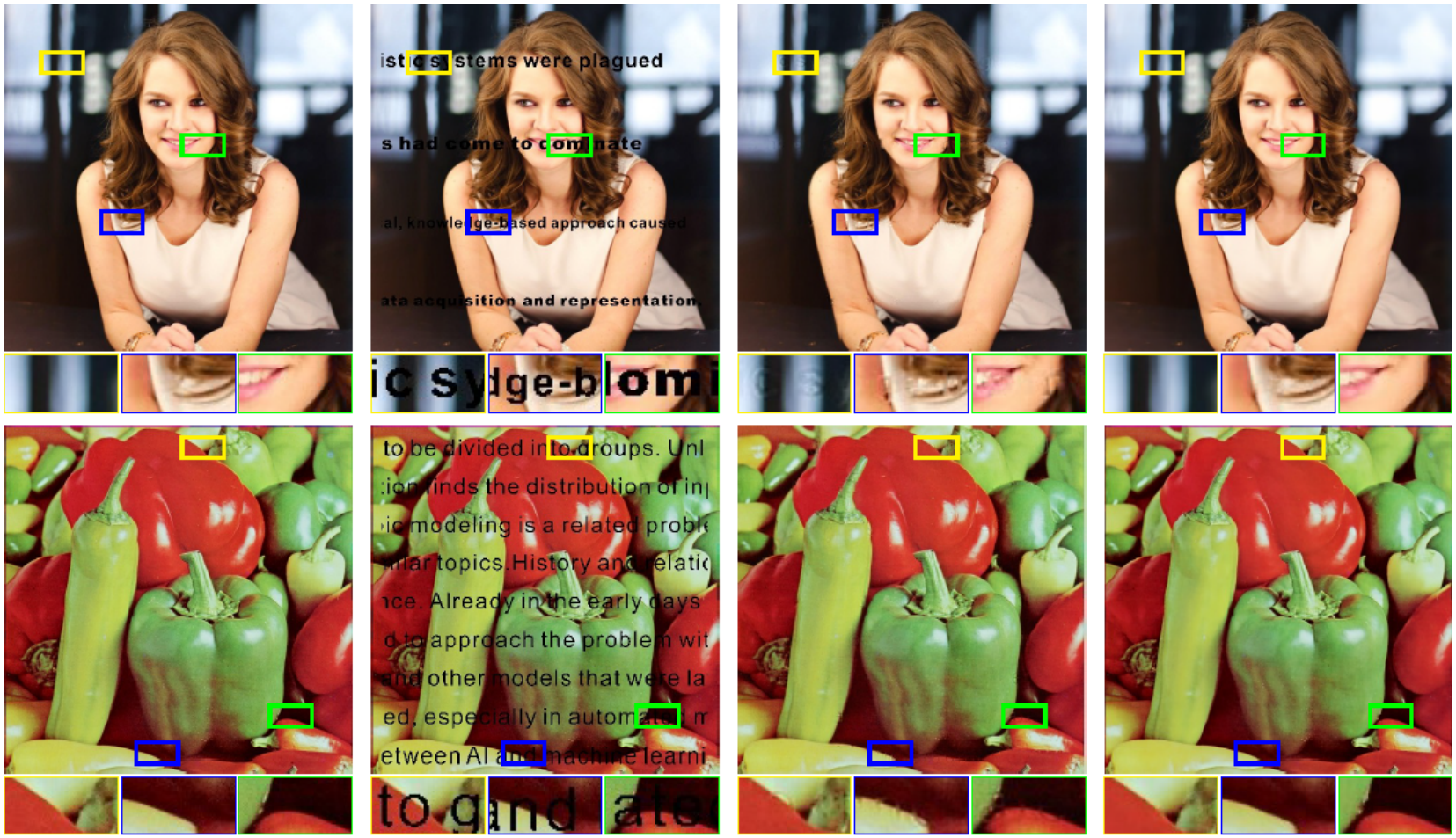

0.08
0.06
0.04
0.02
0.00
MSE
10^1
10^2
10^3
10^4
Iteration (log scale)
Image
Image + noise
Image shuffled
$U(0, 1)$ noise

Image denoising D Ulyanov, A Vedaldi, V Lempitsky CVPR 2018, 9446–9454



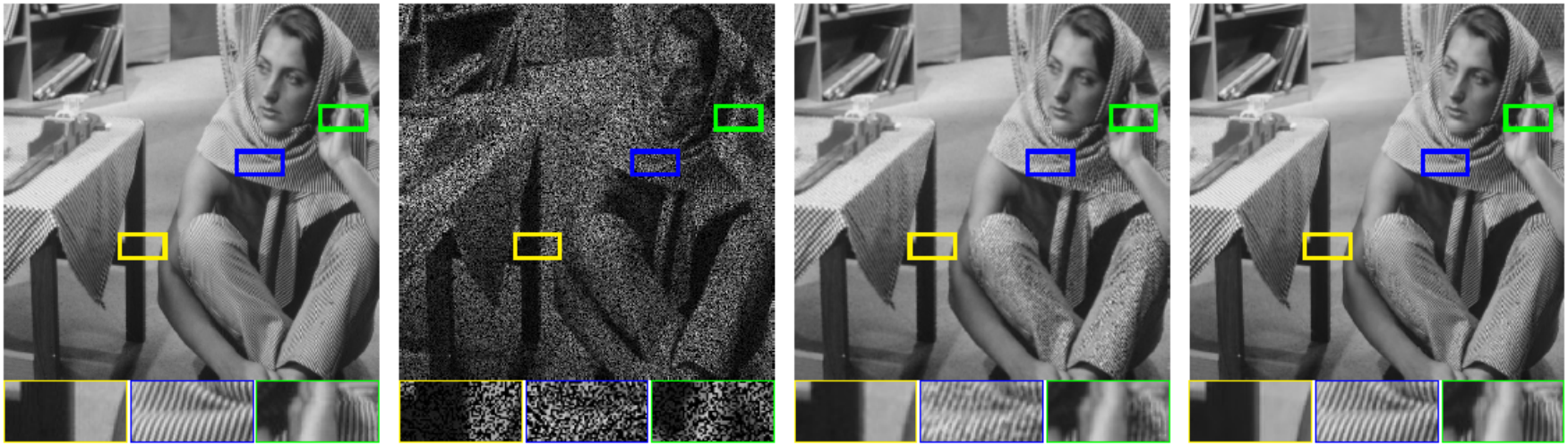
(a) GT (b) Input (c) Ours (d) CBM3D (e) NLM

Image inpainting D Ulyanov, A Vedaldi, V Lempitsky CVPR 2018, 9446–9454



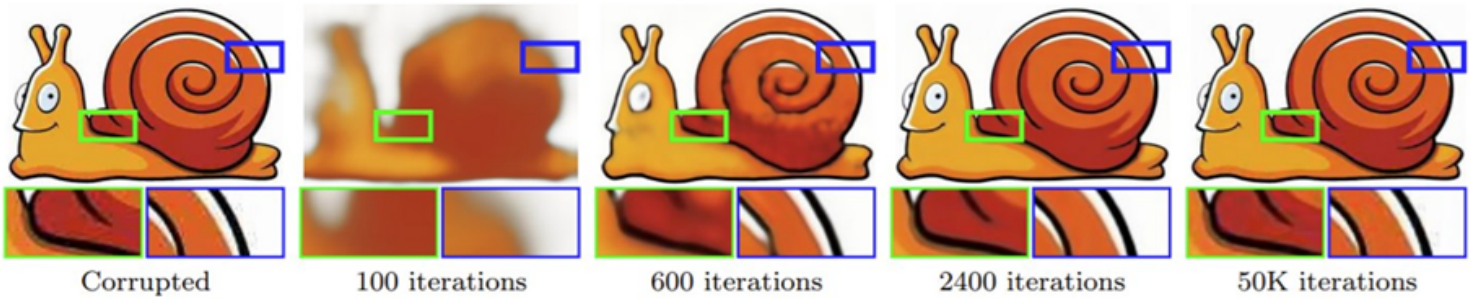
(a) Original image (b) Corrupted image (c) Shepard networks [47] (d) Deep Image Prior

Image inpainting D Ulyanov, A Vedaldi, V Lempitsky CVPR 2018, 9446–9454



(a) Original image (b) Corrupted image (c) CSC [44] (d) Deep image prior

overfitting without regularization D Ulyanov, A Vedaldi, V Lempitsky CVPR 2018



message: early stopping is needed !

effective strategy: regularize explicitly (via total variation) D. Otero Baguer, J. Leuschner, M. Schmidt. Inverse Problems 2020

electrical impedance tomography

$$\begin{cases} \nabla \cdot (\sigma \nabla u) = 0, & \text{in } \Omega \\ u = f, & \text{on } \partial\Omega \end{cases}$$

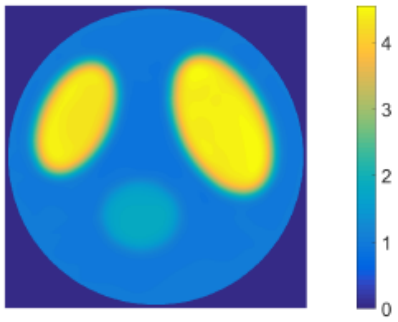
many Cauchy data pairs: $(f, \partial_\nu u(f)) \Rightarrow \sigma$

EIT reconstruction loss Bar-Sochen SIAM J. Imag. Sci. 2021

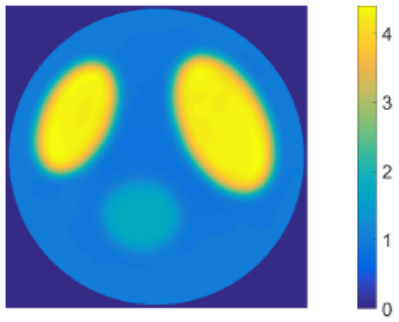
$$\mathcal{I}\Big(\sigma(\mathbf{x}; w_\sigma)\Big) = \frac{\lambda}{N_s}\sum_{i=1}^{N_s}|\mathcal{L}_i|^2 + \frac{\mu}{K}\sum_{k\in \mathrm{top}_K(|\mathcal{L}_i|)}|\mathcal{L}_k| + \frac{1}{N_b}\sum_{b=1}^{N_b}\Big|\sigma(x_b) - \sigma_0(x_b)\Big| + \eta\|w_\sigma\|_2^2 + \frac{\beta}{N_s}\sum_{i=1}^{N_s}|\nabla\sigma(x_i)|^p.$$

other approaches: physics informed neural networks, deep Galerkin method, deep Ritz method ...

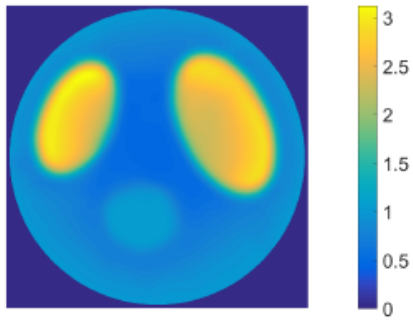
EIT reconstruction Bar-Sochen SIAM J. Imag. Sci. 2021



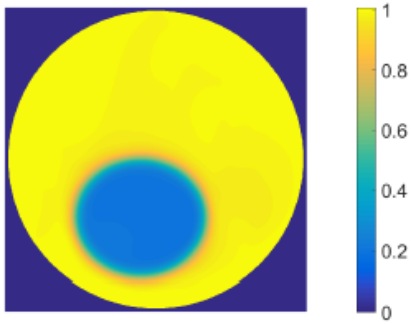
(a) no noise



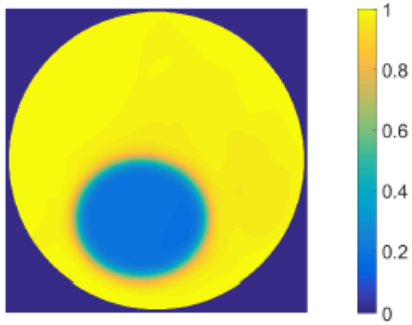
(b) $\sigma_{\text{noise}} = 1\text{e}{-6}$



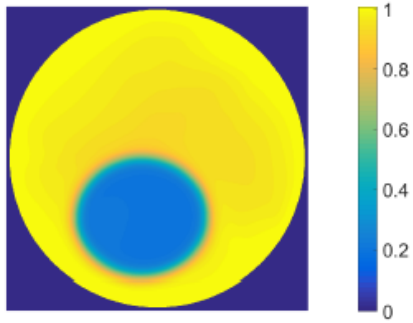
(c) $\sigma_{\text{noise}} = 5\text{e}{-6}$



(d) no noise

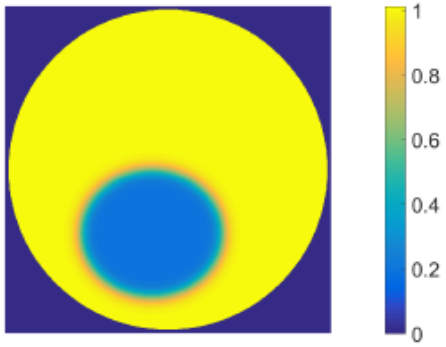


(e) $\sigma_{\text{noise}} = 1\text{e}{-6}$

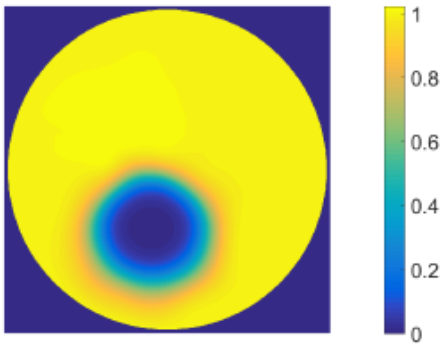


(f) $\sigma_{\text{noise}} = 5\text{e}{-6}$

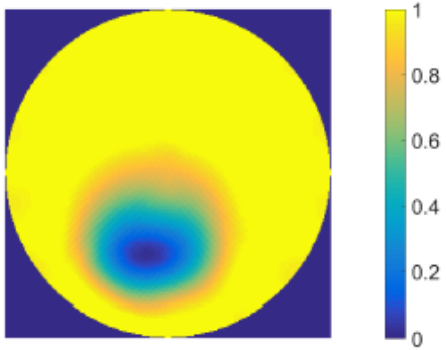
EIT reconstruction Bar-Sochen SIAM J. Imag. Sci. 2021



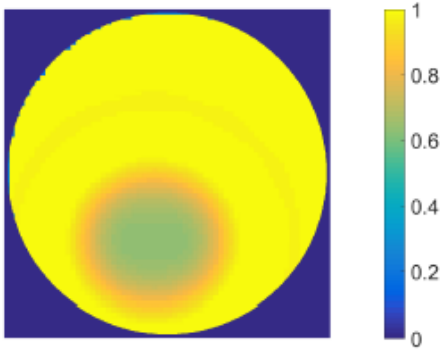
(a) Ground truth



(b) Proposed



(c) EIDORS [1]



(d) D-bar [24]

short history of the method

- first proposed by D Ulyanov, A Vedaldi, V Lempitsky CVPR 2018
  image denoising, super-resolution, inpainting, delurring
- deep decoder R Heckel, P Hand ICLR 2019
- theory: DIP for compressed sensing with convolution type architecture, smooth components converge faster than noise

  R Heckel, M Soltanolkotabi ICML 2020
- theory: connection with Tikhonov regularization (analytic DIP)

  S Dittmer, T Kluth, P Maass, D Otero Baguer JMIV 2020; Clemens Arndt 2022, Inverse Problems
- many applications: CT, PET, MRI, MPI, ...

compressed sensing (sparse recovery) Heckel et al 2020 ICML
observations

- Un-trained CNNs are empirically most effective when they are over-parameterized
- Early stopping can be critical, e.g., for denoising

goal:

- why for CS, gradient descent can reconstruct a good signal estimate without any regularization
- prove that this is possible with a minimal number of meas. prop. to signal dimen.

compressive sensing

$$y = Ax^* \in \mathbb{R}^m$$

recover $x^* \in \mathbb{R}^n$ from $m \ll n$ linear measurements

method: use an **over-parameterized** untrained CNN prior

$$G : \mathbb{R}^N \to \mathbb{R}^n$$

- $N \gg n$, parameter-vector $C \in \mathbb{R}^N$ (over-parameterized)
- G: deep decoder, simple *un-trained* convolutional network
- randomized initialized (without training)

reconstruction: (randomly initialized) gradient descent on the loss

$$\mathcal{L}(C) = \frac{1}{2}\|AG(C) - y\|^2$$

- two-layer convolutional generator $G : \mathbb{R}^{nk \times n} \to n$

$$G(C) = \mathrm{ReLU}(UC)v$$

- $v = [1, \ldots, 1, -1, \ldots, -1]/\sqrt{k}$ fixed weights
- $C \in \mathbb{R}^{n \times k}$: coeff. matrix of generator (network weights in layer 1)
- U: convolution with fixed kernel u, circulant matrix $U \in \mathbb{R}^{n \times n}$
- deep decoder (d layer)

$$G(C) = \mathrm{ReLU}(UB_dC_d)v, \quad B_{i+1} = \mathrm{cn}(\mathrm{ReLU}(U_iB_iC_i)), \quad i = 0, \ldots, d-1.$$

warmup: recovery with over-parameteried linear generator

$$\tilde{G}(c) = Jc,$$

with $J = \mathbb{R}^{n \times N}$, full rank

$$y = Ax^*$$

$A \in \mathbb{R}^{m \times n}$, Gaussian meas. matrix, iid $N(0, m^{-1})$ (approx. norm preserving, i.e. $\|z\| = \|Az\|$)

$$\mathcal{L}(c) = \frac{1}{2}\|AJc - y\|^2$$

gradient descent with small step size $\Rightarrow$ minimum norm sol.

$$\hat{c} = (AJ)^\dagger AJc^* = P_{J^T A^T} c^*$$

and

$$\hat{x} - x^* = J(\hat{c} - c^*) = J(I - P_{J^T A^T})c^*$$

Let $A \in \mathbb{R}^{m \times n}$ be random Gaussian matrix with $m \geq 12$, $w_1, \ldots, w_n$ be the left singular vectors of J associated with $\sigma_1 \geq \cdots \geq \sigma_n$. Then for any $x^* \in \mathbb{R}^n$, with prob. at least $1 - 3e^{-\frac{1}{2m}}$, the signal est. $\hat{x} = J\hat{c}$ based on $y = Ax^*$,

$$\|\hat{x} - x^*\|^2 \leq c\Big(\sum_{i=1}^{n} \sigma_i^{-2}(x^*, w_i)^2\Big) \sum_{i > \frac{2m}{3}} \sigma_i^2.$$

good approxim. if

- x^* lies in the span of leading $O(m)$ sing. vectors of J
- S.V. of J decay sufficiently fast

$$\mathcal{L}(C) = \frac{1}{2}\|AG(C) - y\|^2$$

$A \in \mathbb{R}^{m\times n}$: Gaussian r.v. with iid $N(0, m^{-1})$ entry

- gradient descent with constant step size
- random initialization, Gaussian $N(0, \omega^2)$
- iteration

$$C_{t+1} = C_t - \eta\nabla\mathcal{L}(C_t)$$

- key: replace J with $J(C) = \frac{\partial G}{\partial C}$

Let $A \in \mathbb{R}^{m\times n}$ be a random Gaussian matrix with $m \geq 12$ and suppose we are given a linear measurement $y = Ax^*$ of an arbitrary signal $x^* \in \mathbb{R}^n$. If $k \geq C_u \frac{m}{\xi^8}$ (for $\xi \leq 1$), $\omega \propto \|y\|/\sqrt{n}$, and the stepsize is sufficient small

$$\|G(C_\infty) - x^*\|^2 \leq C\Big(\sum_i \sigma_i^{-2}(w_i, x^*)^2\Big) \sum_{i>\frac{2m}{3}} \sigma_i^2 + \xi^2\|x^*\|^2.$$

key idea: for wide networks, in the neighborhood of random init., the Jacobian does not change much .... (similar to NTK)

$J(C_0)$ Jacobian at random initialization C_0

$$\mathbb{E}[J(C_0)J(C_0)^T] = \left(\frac{1}{2}\left(1 - \frac{1}{\pi}\cos^{-1}\frac{(u_i, u_j)}{\|u_i\|\|u_j\|}(u_i, u_j)\right)\right)_{i,j=1}^{n}$$
$$=: S(U) = JJ^T$$

J reference jacobian

i.e., Jacobian at initialization approximately only dependent on U

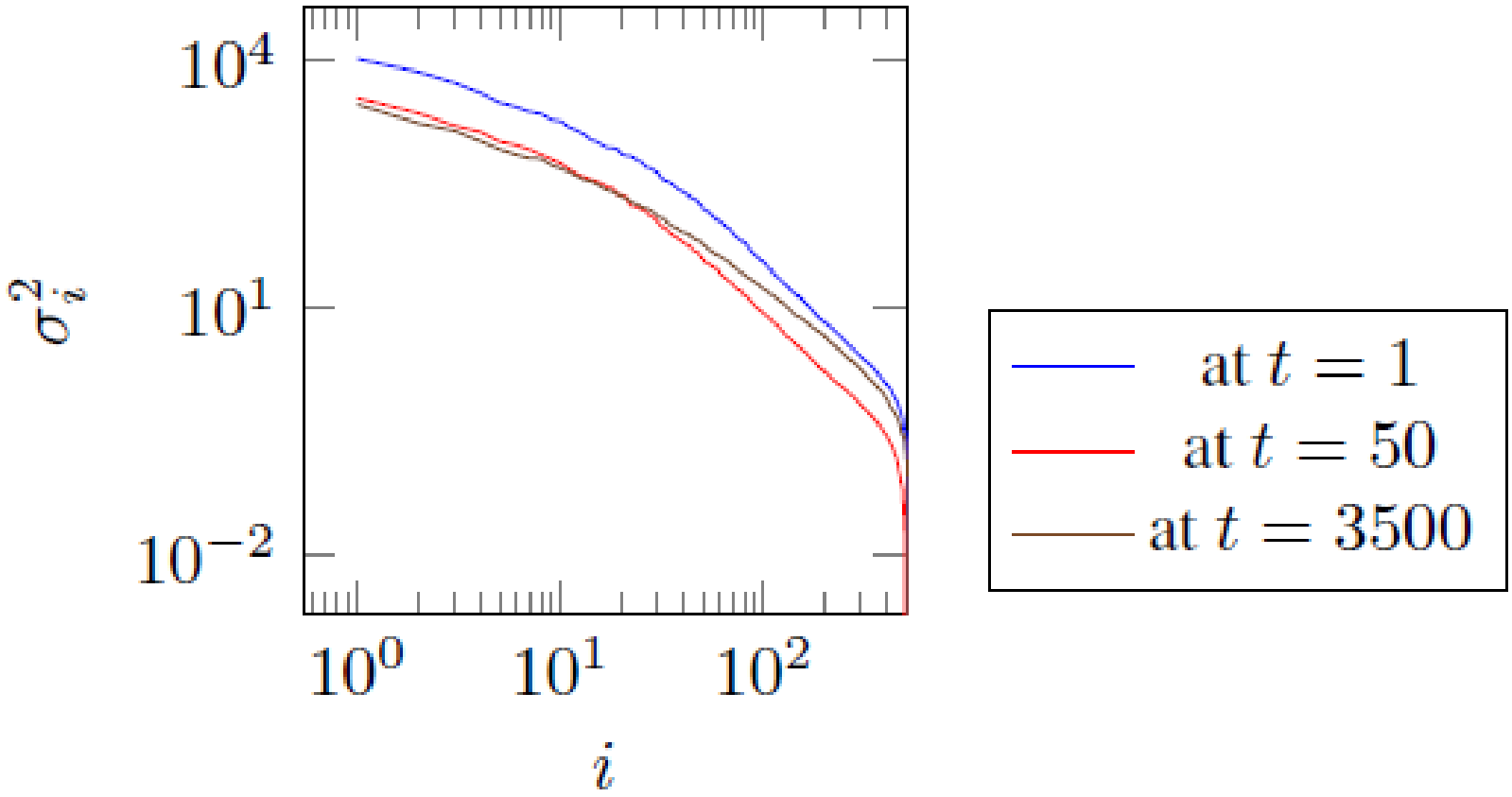
key facts: left singular vectors of J

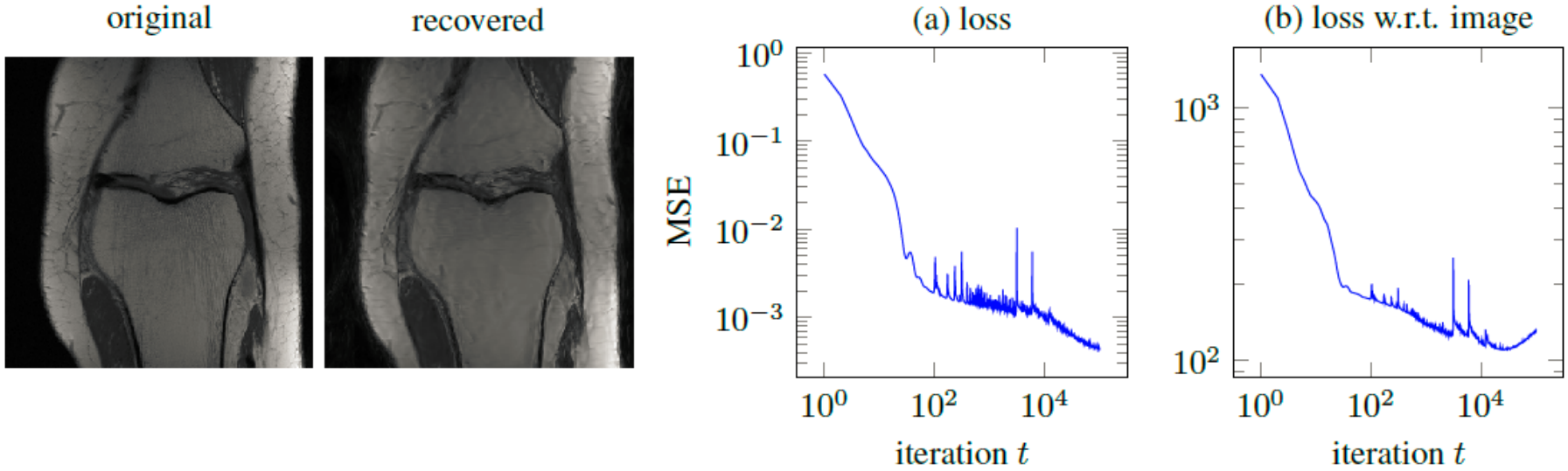
$$[w_i]_j = \frac{1}{\sqrt{n}} \begin{cases} 1, & i = 0, \\ \sqrt{2}\cos(2\pi jin), & i = 1, \ldots, n/2 - 1 \\ (-1)^j, & i = n/2 \\ \sqrt{2}\sin(2\pi ji/n), & i = n/2 + 1, \ldots, n - 1 \end{cases}$$

singular values

$$\sigma = \|u\| \sqrt{|Fg(\frac{u \otimes u}{\|u\|^2})|}$$

with $g(z) = \frac{1}{2}(1 - \frac{\cos^{-1} z}{\pi})z$


10^4
10^1
10^{-2}
c_i^2
10^0
10^1
10^2
i
at $t = 1$
at $t = 50$
at $t = 3500$


original
recovered
(a) loss
(b) loss w.r.t. image
MSE
10^0
10^{-1}
10^{-2}
10^{-3}
10^0
10^2
10^4
iteration t
10^3
10^2
10^0
10^2
10^4
iteration t

pros and cons of DIP

- free from any training data, robust w.r.t. distributional shift
- the need of early stopping
- fresh training for each new data $\Rightarrow$ expensive to deploy
- no uncertainty estimate

The case of general inverse problems

- apply gradient descent to

$$\mathcal{L}(C) = \tfrac{1}{2}\|AG(C) - y^{\delta}\|^2$$

  with random Gaussian initialization C_0

- discrepancy principle for stopping of the iteration (with $\tau > 1$

$$k_{\mathrm{dp}}^{\delta} := \min\{k \geq 0 : \|AG(C_k) - y^{\delta}\| \leq \tau\delta\}$$

benchmark setting

- source condition: $\mathcal{X}_{\nu,r} := \{x \in \mathbb{R}^n : x = (A^T A)^{\nu/2} w, \|w\| \leq r\}$
- worst-case-error rate: $e_{\mathrm{wc}}(\delta, r, \nu)$

Jahn, Jin, 2023

Assume that A and $S(U)$ have polynomially decaying singular values and aligned right singular vectors. Then, for $N = N(\delta, \epsilon)$ large enough and $\omega = \omega(\delta, \epsilon)$ small enough and a constant $C > 0$, there holds

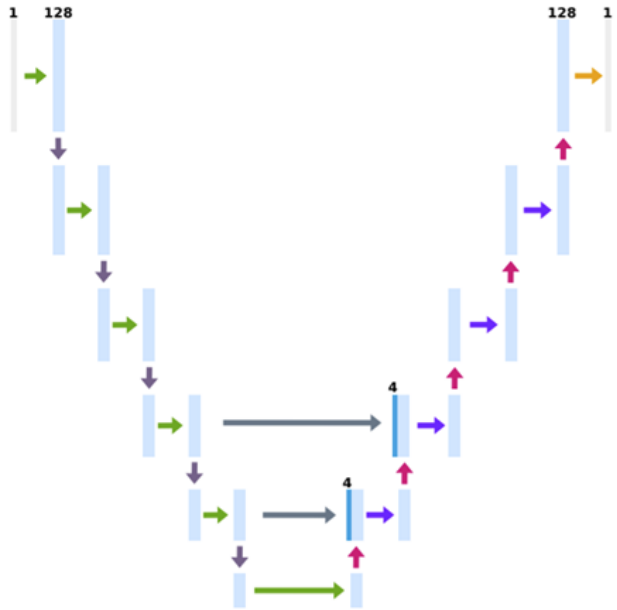
$$\inf_{x \in \mathcal{X}_{r,\nu}} \mathbb{P}\left(\|G(C_{k_{\mathrm{dp}}^{\delta}}) - x^{\dagger}\| \leq C e_{\mathrm{wc}}(\delta, r, \nu) \right) \geq 1 - \epsilon$$

message: **Optimal convergence for large enough network!**

deep image prior for inverse problems

$$\theta^* = \arg\min_{\theta \in \mathbb{R}^{d_\theta}} \{\ell(\theta) := \|Af_\theta(z) - y^\delta\|^2 + \lambda |f_\theta(z)|_{\mathrm{TV}}\}, \quad x^* = f_{\theta^*}(z)$$

- **Unsupervised** learning
- ~~Ordered pairs of ground truths images and real measurement data~~?
- Robust to out-of-distribution samples?
- Suitable for imaging applications with scarce data availability??



U-net architecture

Ronneberger et al MICCAI 2015

Accelerating DIP with pretraining

Can DIP benefit from **pretraining** for accelerating subseq. reconstructive tasks? If so, can we easily construct an informative dataset to warm–start DIP? How do inductive biases of pretraining impact the reconstructive task?

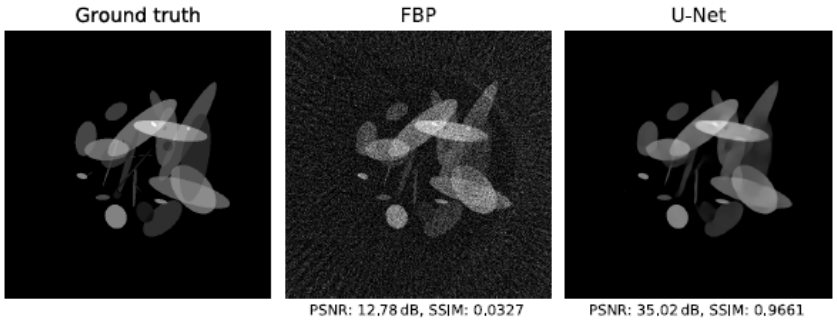
R Barbano, J Leuschner, M Schmidt, A Denker, A Hauptmann, P Maass, B Jin. 2022. IEEE Trans. Comput. Imag.

Educated DIP (EDIP)

- Step 1: Superv. pretraining on synthetic training data $(x^i, y_i^\delta)_{i=1}^N$

$$\theta_s^* = \arg\min_{\theta \in \mathbb{R}^{d_\theta}} \frac{1}{N} \sum_{i=1}^{N} \| f_\theta(A^\dagger y_i^\delta) - x^i \|^2$$

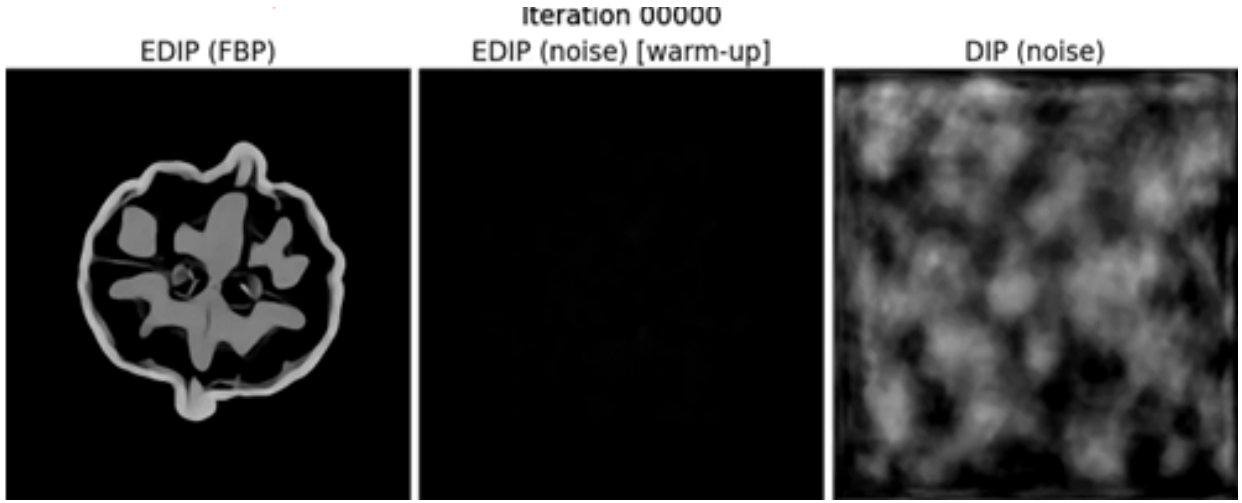
$A^\dagger$: approx. inverse, e.g., filtered backpropagation for CT



- Step 2: Unsupervised fine-tuning on real data

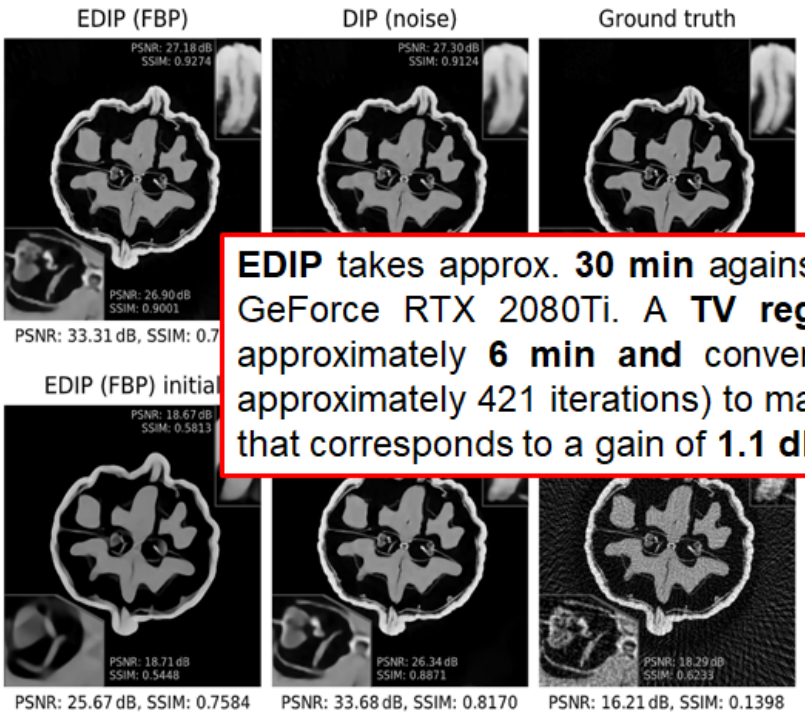
$$\theta_t^* \in \arg\min_{\theta \in \mathbb{R}^{d_\theta}} \|A f_\theta(z) - y^\delta\|^2 + \lambda |f_\theta(z)|_{\mathrm{TV}} \quad x^* = f_{\theta_t^*}(z)$$

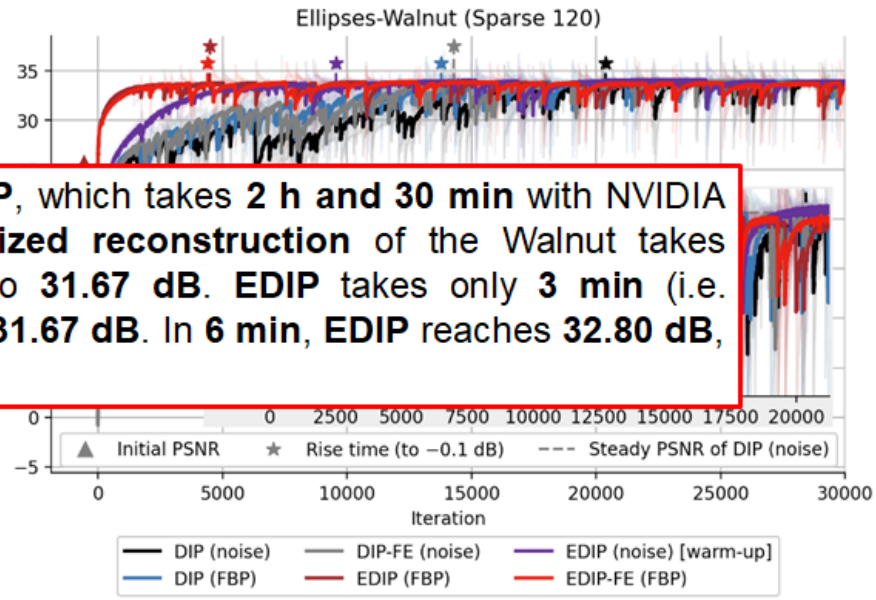
Barbano, R., Leuschner, J., Schmidt, M., Denker, A., Hauptmann, A., Maaß, P. and Jin, B., 2021. IEEE Trans.



https://educateddip.github.io/docs.educated_deep_image_prior/

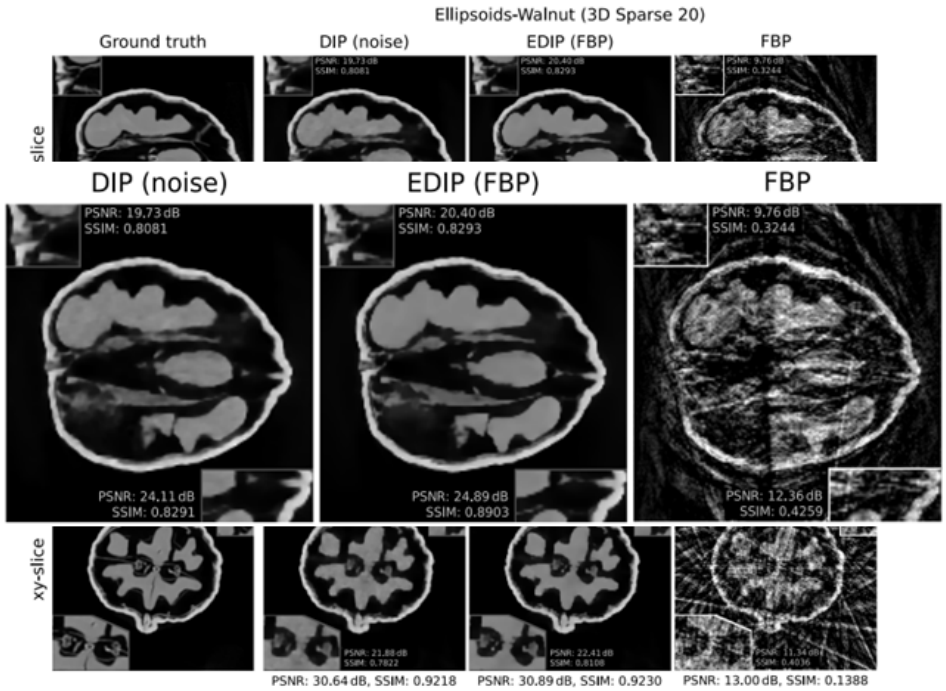
Reconstruction with walnut dataset

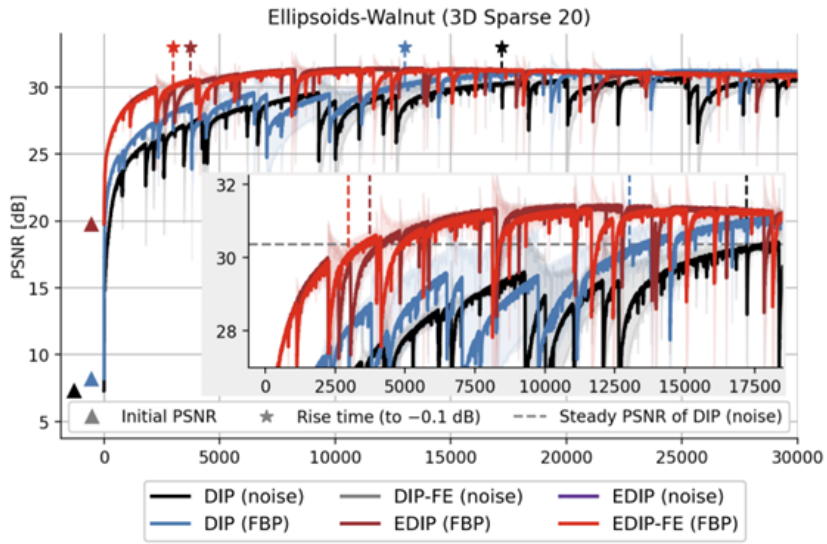




EDIP takes approx. **30 min** against **DIP**, which takes **2 h and 30 min** with NVIDIA GeForce RTX 2080Ti. A **TV regularized reconstruction** of the Walnut takes approximately **6 min and** converge to **31.67 dB**. **EDIP** takes only **3 min** (i.e. approximately 421 iterations) to match **31.67 dB**. In **6 min**, **EDIP** reaches **32.80 dB**, that corresponds to a gain of **1.1 dB**.

Reconstruction with **3D** walnut dataset (resolution: 167^3)





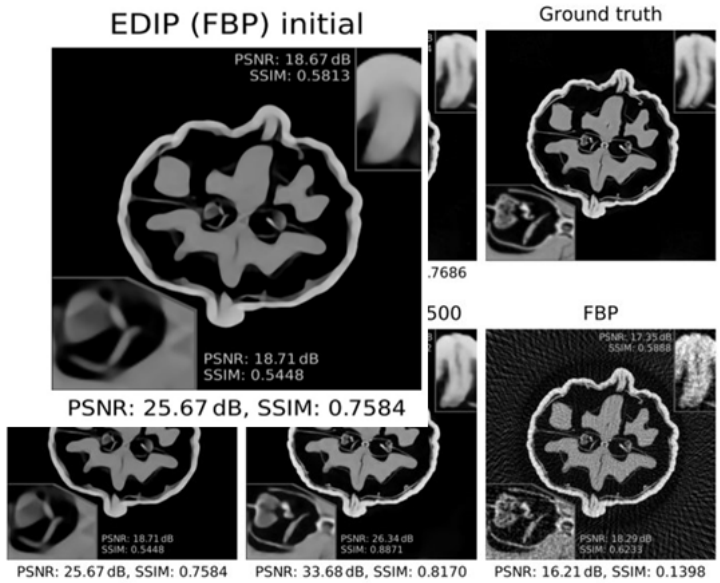
Shedding insights into pretraining

Feature reuse plays a very crucial role

- The deployment of the pretrained model on an out-of-distribution task shows high input-robustness

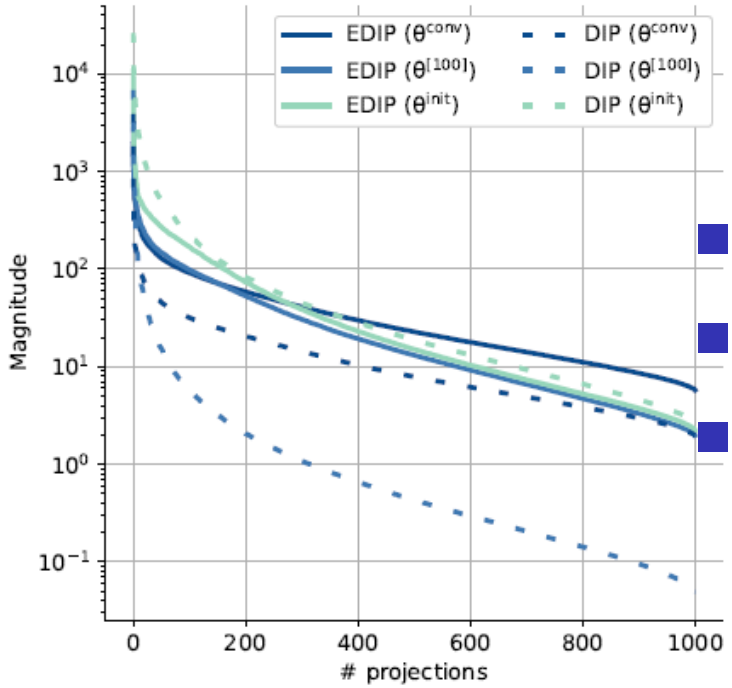
The feature reuse mechanism leads to hallucinatory behaviors

- Amending the knowledge acquired via pretraining protects from instabilities due to distributional shifts



spectral evaluation of pretraining

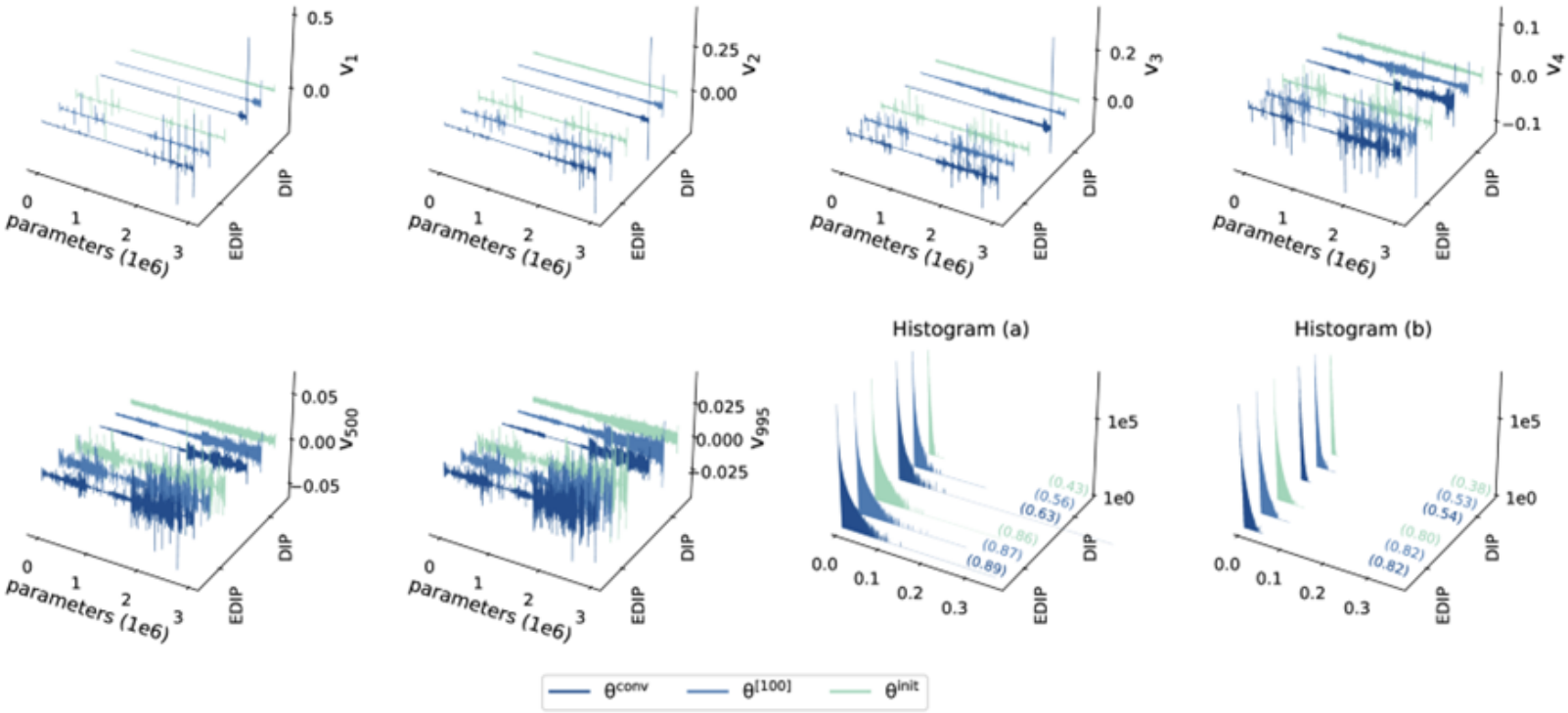
construct an approx SVD of the Jacobian via randomization



- SVs of DIP Jacobian shifts greatly
- SVs of EDIP Jacobian are stable
- $\Rightarrow$ dynamics of training process

spectral evaluation of pretraining

construct an approx SVD of the Jacobian via randomization



Uncertainty quantification?

Question: can we put an **error bar** on the DIP reconstruction, on realistic 3D dataset ?

J Antorán, R Barbano, J Leuschner, JM Hernández-Lobato, B Jin. Trans. Mach. Learn. (2023)

A Bayesian approach to quantify uncertainty

In the **Bayesian** framework, instead of finding a **single best** image, the **posterior** distribution

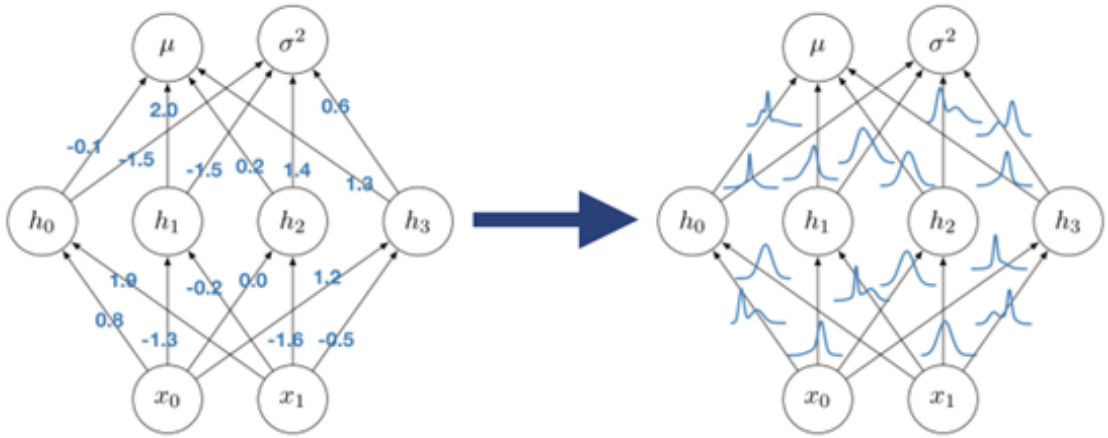
$$p(x|y^\delta) = p(y^\delta|x) \overbrace{p(x)}^{\text{prior belief}} / \underbrace{p(y^\delta)}_{\text{model evidence}}$$

ranks every x according to their agreement with y^δ and the prior belief

The normalization constant $p(y^\delta)$ provides an objective for hyperparameter selection without the need for validation data

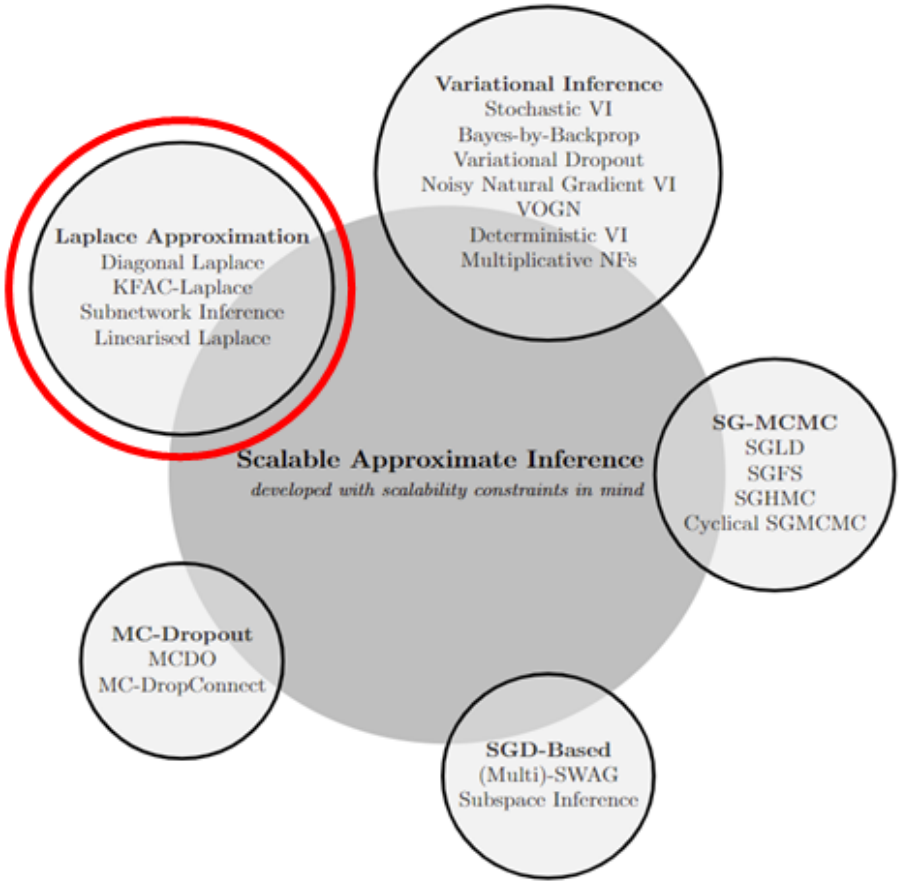
Tools for modelling uncertainty in deep learning

- Place a prior distribution $p(\theta)$ over NN parameters
- Define a likelihood function $p(y_\delta|\theta)$ to characterize the agreement of the NN function with the observations
- Update the weight distribution $p(\theta|y_\delta)$ using Bayes' rule



intractable posterior distribution over weights

towards scalable approximation ...



Barbano,..., 2022. Uncertainty quantification in medical image synthesis. In Biomedical Image Synthesis and Simulation (pp. 601-641). Academic Press.

Laplace approximation

- train the neural network (find a mode, standard training)

$$\theta^* = \arg\min_{\theta \in \mathbb{R}^{d_\theta}} \sum_{i=1}^{N} \underbrace{\ell(x^i, x^{i*}; \theta)}_{-\log p(x^i | f_\theta(A^\dagger y_i^\delta))} + \underbrace{R(\theta)}_{-\log p(\theta)}$$

- Approximate the posterior distribution over the NN parameters θ

  McKay 1992

$$p(\theta|\mathcal{D}) \approx q(\theta) := \mathcal{N}(\theta; \theta^*, \Sigma_\theta), \quad \text{with } \Sigma_\theta = -[\nabla_\theta^2 J(\mathcal{D}; \theta)|_{\theta=\theta^*}]^{-1}$$

- Further ... approximate (with input $A^\dagger y^\delta$)

$$q_{\text{ggn}}(\theta) := \mathcal{N}(\theta; \theta^*, (J^\top \Lambda(\theta) J + S_0^{-1})^{-1}|_{\theta=\theta^*}$$

A. Immer, M. Korzepa, M. Bauer. ICML 2021; E. Daxberger, E. Nalisnick, J. Allingham, J. Antorán J. M. Hernández-Lobato ICML 2021

linearized Laplace method

- A Gaussian is poor to the NN's posterior distribution ... yet exp. it is a very good posterior for a linear model
- linearized the underlying BNN

$$h(\theta) = f_{\theta^*}(A^\dagger y^\delta) + \underbrace{J_{\theta^*}(A^\dagger y^\delta)}_{\text{basis expansion!}} (\theta - \theta^*)$$

$$x \sim \mathcal{N}(h(\theta^*), J\Sigma_\theta(\ell, \sigma_\theta^2)J^\top)$$

linearization induces a **GP deep image prior**

- perform approx. inference in the GP model or solve it in closed form for regression $\Rightarrow$ conjugate Gaussian-linear model has:

Closed form **predictive posterior** and **marginal likelihood**

The revival of linearized neural networks

- David Mackay (1992) first introduced NN linearisation as an approximation to yield a closed form error-bars for Laplace approximated posteriors
- Lawrence (2000) finds the Laplace approximation to underperform without the linearisation step. This was also noted by Ritter (2018) in order to scale-up the Laplace approximation using Kronecker factorisation
- Khan (2019), Immer (2021) re-popularised the linearisation step by showing that it improves the quality of uncertainty estimates

Deep image prior + linearized Laplace $\Rightarrow$

linearised deep image prior for CT reconstruction?

- TV is not smooth enough ! $\Rightarrow$ linearized Laplace does not apply
- tractable hierarchical construction via predictive complexity prior

...

Method: compute a Gaussian-linear model type error-bars using a local linearization of the DIP around its optimal reconstruction

Designing a tractable Bayesian prior over images mimicking TV

- Bayesian construction

$$y^\delta|\theta \sim \mathcal{N}(y^\delta; Ah(\theta), \sigma^2 I),$$
$$\theta|\ell \sim \mathcal{N}(\theta; \theta^*, \Sigma_\theta(\ell, \sigma_\theta^2)), \quad \ell \sim p(\ell)$$

- Matern kernel

$$[\Sigma_\theta(\ell, \sigma_\theta^2)]_{kij,k'i'j'} = \sigma_d^2 e^{-\frac{-d(i-i',j-j')}{\ell_d}} \delta_{kk'}$$

with $k \sim$ convolutional filters, $i, j \sim$ spatial location within the filter

- connecting TV via predictive complexity prior

$$p(\ell) = \prod_d p(\ell_d) = \prod_d \mathrm{Exp}(\lambda\kappa_d)\left|\frac{\partial \kappa_d}{\partial \ell_d}\right|$$
$$\kappa_d := \mathbb{E}_{p(\theta_d|\ell_d,\sigma_d^2)\prod_{i\neq d}\delta_{\theta_i}}[|h(\theta)|_{\mathrm{TV}}]$$

optimizing $(\ell, \sigma_y^2, \sigma_d^2)$ via marginal likelihood (Type-II evidence)

$$\log p(\mathbf{y}_\delta|\boldsymbol{\ell};\sigma_y^2,\boldsymbol{\sigma}_{\boldsymbol{\theta}}^2) + \log p(\boldsymbol{\ell};\boldsymbol{\sigma}_{\boldsymbol{\theta}}^2) =$$
$$-\frac{1}{2}\sigma_y^{-2}||\mathbf{y}_\delta - \mathbf{A}\mathbf{f}(\boldsymbol{\theta}^\star)||_2^2 - \frac{1}{2}||\boldsymbol{\theta}^\star||^2_{\Sigma_{\boldsymbol{\theta}}^{-1}(\boldsymbol{\ell},\boldsymbol{\sigma}_{\boldsymbol{\theta}}^2)}$$
$$-\frac{1}{2}\log|\Sigma_{\mathbf{y}_\delta\mathbf{y}_\delta}(\sigma_y^2,\boldsymbol{\ell},\boldsymbol{\sigma}_{\boldsymbol{\theta}}^2)| - \lambda\sum_{d=1}^{D}\kappa_d + \log\left|\frac{\partial\kappa_d}{\partial\ell_d}\right|.$$

many computational tricks

- randomized estimators of determinants / gradients
- PCG with half precision
- low rank + diagonal preconditioner
- ...

public lib: `https://github.com/educating-dip/bayesdip`

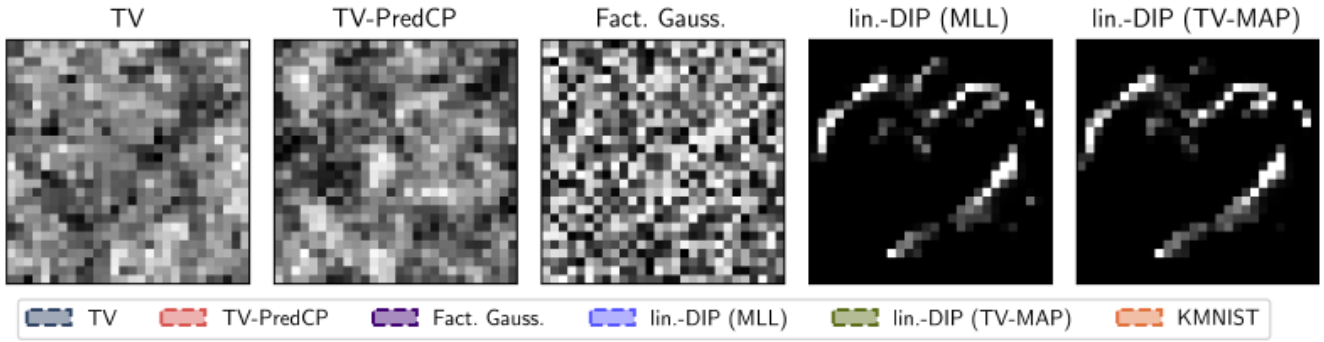
Five steps from regularized reconstruction to Bayesian inference

- Train (educated) U-net with “standard objective”
- Linearize around some acceptable parameter setting
- Build Bayesian hierarchical model: Build surrogate prior with a kernel that enforces TV smoothness
- Optimize hyperparameters with marginal likelihood
- Make prediction (UQ) through sampling

Outcome: computational pipeline providing **pixel-wise uncertainty** estimates and a **marginal likelihood** objective

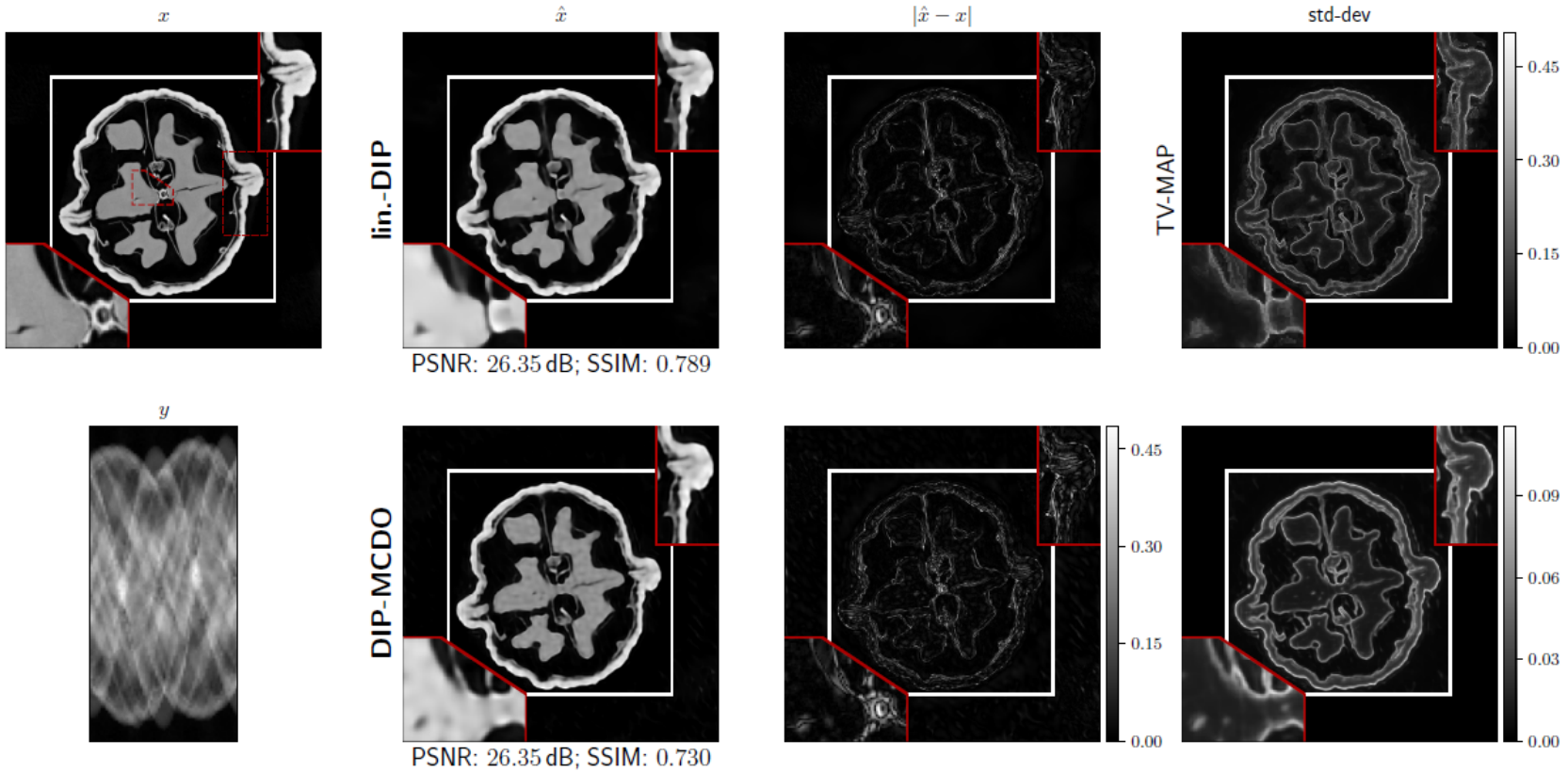
estimate the uncertainty assoc. with $\hat{x}$, instead of a full posteriori.

samples from different priors:



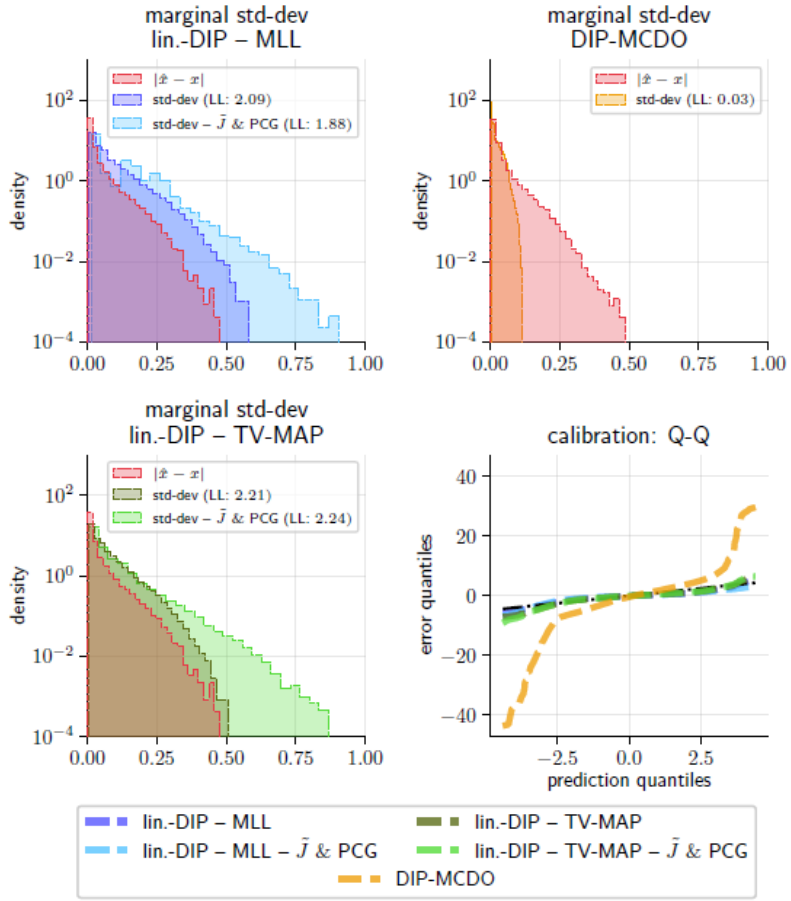
hierarchical construction: very powerful task-specific kernels

lin-DIP for high-resolution CT reconstruction (resolution: 501^2 px)



lin-DIP yields **more calibrated uncertainty estimates** than MC dropout

lin-DIP for high-resolution CT reconstruction



lin-DIP yields **more calibrated uncertainty estimates** than MC

Summary: Take-home message

- Pretraining is effective for accelerating DIP
- lin-DIP provides calibrated uncertainty estimates for DIP reconstructions
- Provide an efficient implementation of the method (as a library)
- the approach can be extended for Bayesian experimental design

  R Barbano, J Leuschner, J Antorán, B Jin, JM Hernández-Lobato arXiv:2207.05714
- other imaging modalities ...

(very incomplete) references

- **D Ulyanov, A Vedaldi, V Lempitsky. Deep image prior. CVPR 2018, pp. 9446-9454**
- R Heckel, M Soltanolkotabi. Compressive sensing with un-trained neural networks: Gradient descent finds the smoothest approximation. ICML 2020
- D Otero Baguer, J Leuschner, M Schmidt. Computed tomography reconstruction using deep image prior and learned reconstruction methods. Inverse Problems 2020; 36(9): 094004
- R Barbano, J Leuschner, M Schmidt, A Denker, A Hauptmann, P Maass. An educated warm start for deep image prior-based micro CT reconstruction? (2022) IEEE Trans. Comput. Imag.
- J Antorán, R Barbano, J Leuschner, JM Hernández-Lobato, B Jin. Uncertainty estimation for computed tomography with a linearised deep image prior (2023) Trans. Mach. Learn.
- R Barbano, J Leuschner, J Antorán, B Jin, JM Hernández-Lobato. Bayesian experimental design for computed tomography with the linearised deep image prior. (2022) arXiv:2207.05714
- ....