Toward Practical Phase Retrieval with Deep Learning

Ju Sun
Computer Science & Engineering, UMN

IPAM WS: Diffractive Imaging with Phase Retrieval, Oct 14, 2022

UNIVERSITY OF MINNESOTA
Driven to Discover℠

Thanks to

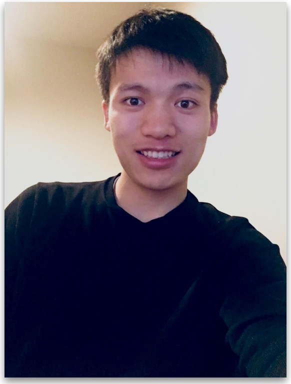
GLOVEX

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)

Thanks to

David Barmherzig

CCM, Flatiron Ins.

Felix Hofmann

DES, Oxford U.

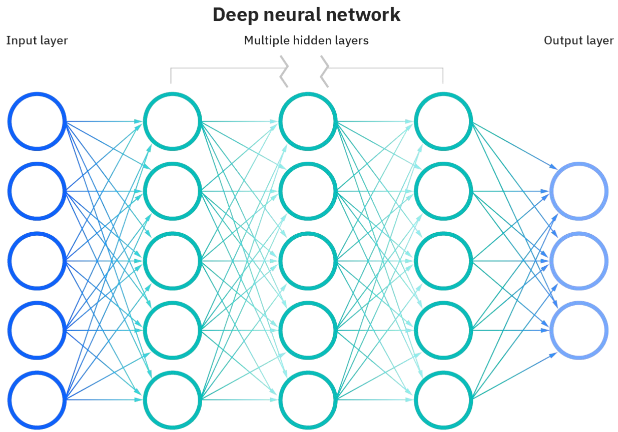
David Yang

DES, Oxford U.

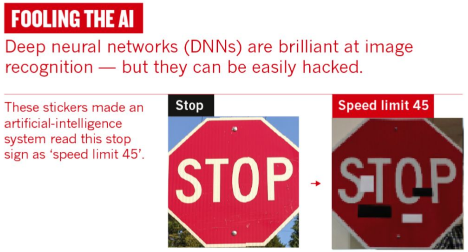
Research in the group

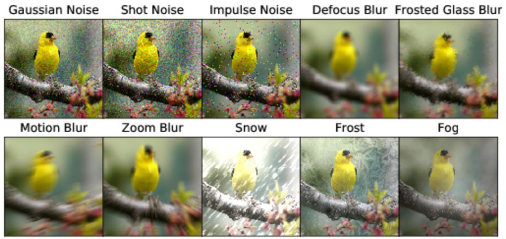
GLOVEX

(Machine) **Learning**, (Numerical) **Optimization**, (Computer) **Vision**, healthcar**E**, + **X**

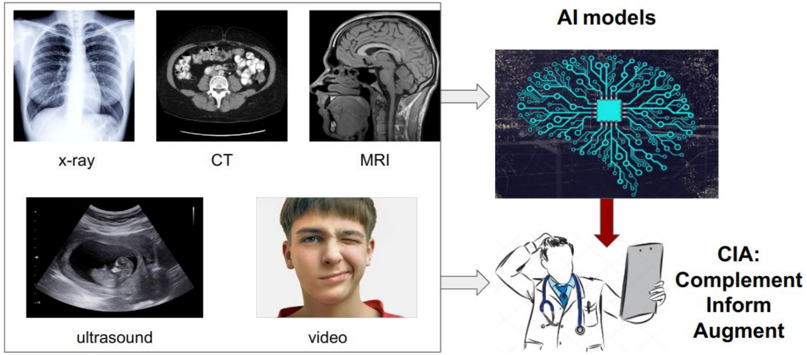


Our research themes

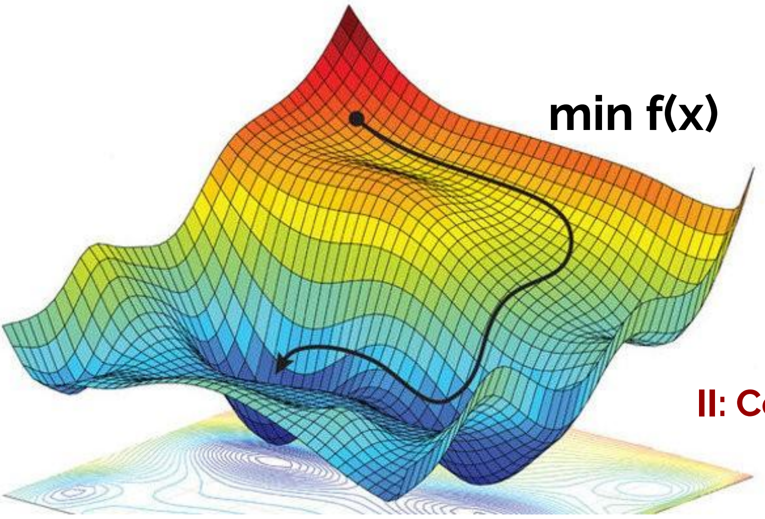




I: Trustworthy AI

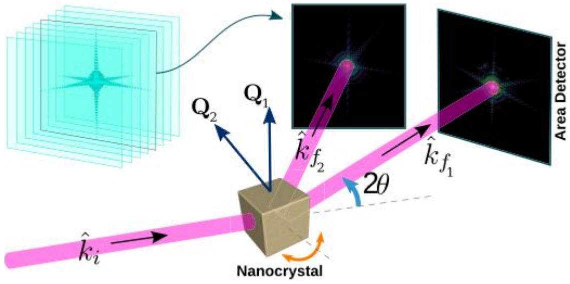


III: AI for Healthcare



$$\min f(x) \quad \text{s.t.} \quad g(x) <= 0$$

II: Computation for AI



IV: AI for Science and Engineering

Which phase retrieval (PR)?

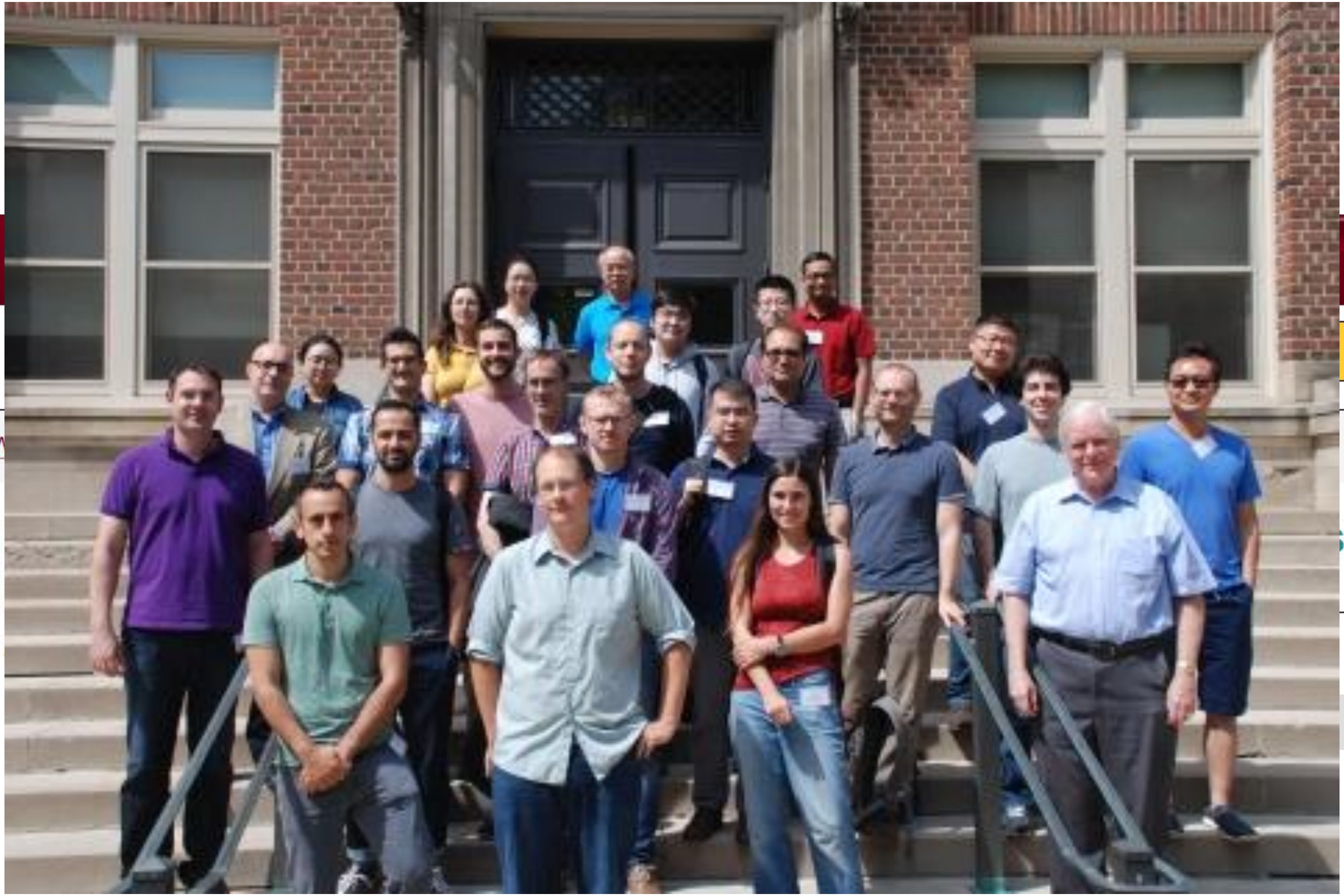
5 year ago…

University of Minnesota
Driven to Discover℠

IMA
INSTITUTE FOR MATHEMATICS AND ITS APPLICATIONS

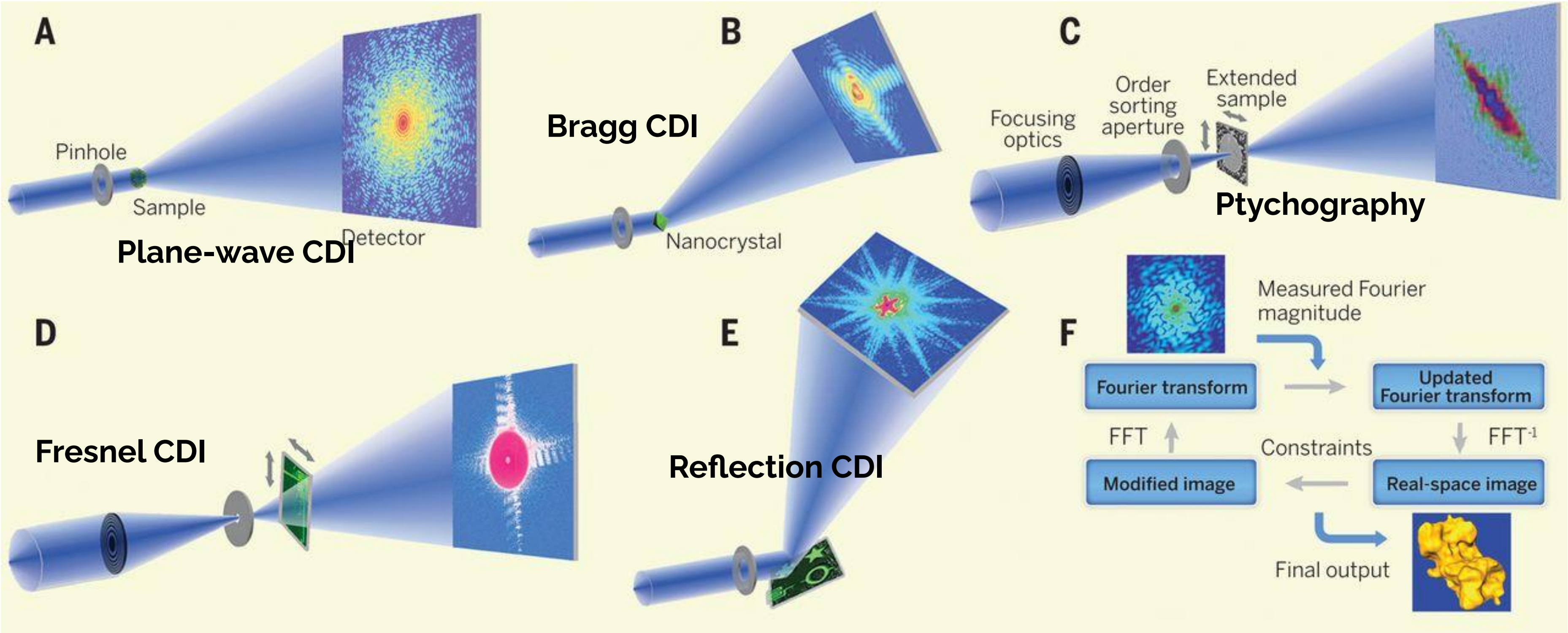
Home › Programs and Activities › Special W

- About
- Programs
  - Thematic Programs
  - Data Science
  - Hot Topics Workshops
  - Math-to-Industry Boot Camp
  - Public Lectures
  - Seminars
  - Special Workshops
  - Archived Programs
- Visiting



Mark Iwen	Michigan State University
Rayan Saab	University of California, San Diego
Aditya Viswanathan	Michigan State University

Which phase retrieval?



Miao et al. Beyond crystallography: Diffractive imaging using coherent x-ray light sources. Science, 2016 https://www.science.org/doi/10.1126/science.aaa1394

Are they created equal in difficulty levels?

$\boldsymbol{Y} = |\mathcal{A}(\boldsymbol{X})|^2$ **Consider** $\min_{\boldsymbol{X}} \ \|\boldsymbol{Y} - |\mathcal{A}(\boldsymbol{X})|^2\|_F^2$

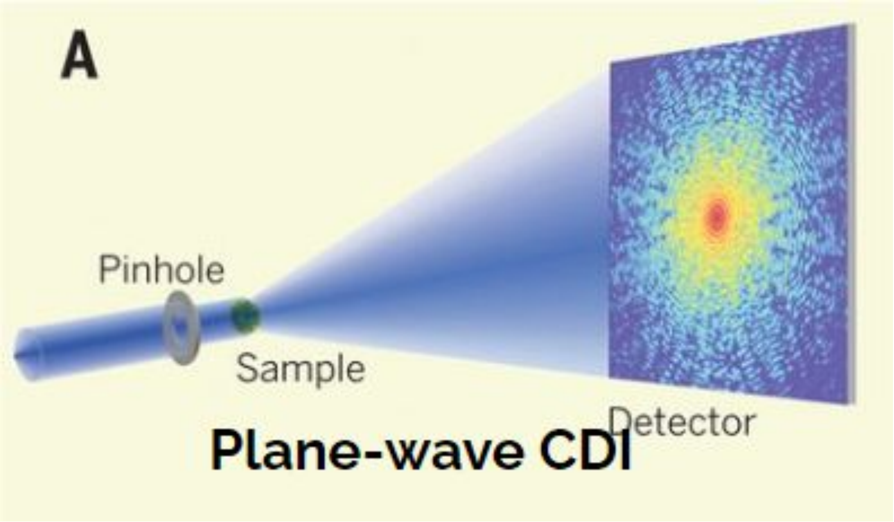
	Gaussian PR	Fresnel PR	Fraunhofer PR
$\mathcal{A}(\boldsymbol{X})$	$\{\langle \boldsymbol{G}_i, \boldsymbol{X}\rangle\}_{i=1}^m$	$\mathcal{F}(\boldsymbol{X} \odot [e^{i\pi C(m^2+n^2)}]_{m,n})$	$\mathcal{F}(\boldsymbol{X})$
Symmetries	Global phase	Global phase	Shift, flipping, global phase

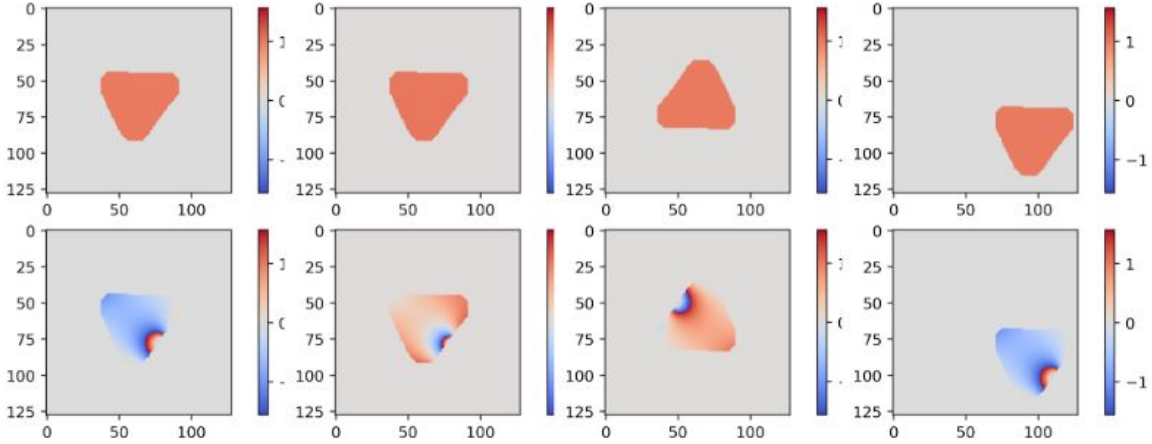
LS-solvable ⟶ **un LS-solvable (proliferation of local mins)**

Focus of this talk:

plane-wave CDI (Fraunhofer PR)

$$\boldsymbol{Y} = |\mathcal{F}(\boldsymbol{X})|^2$$





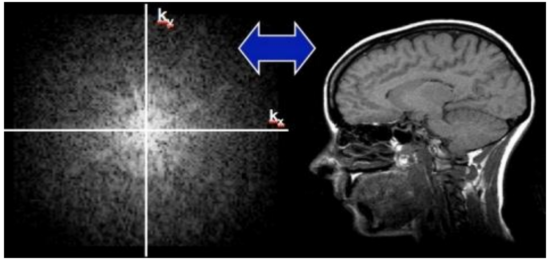
Three symmetries:
global phase
conjugate flipping
shift

When deep learning (DL) meets visual inverse problems (VIPs)

Visual inverse problems

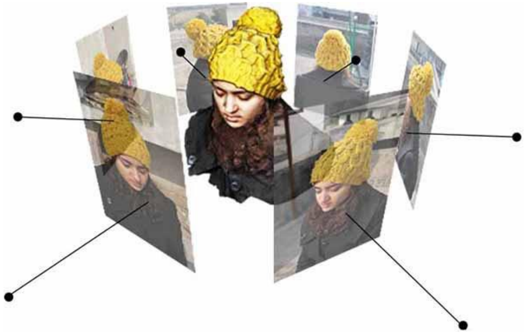
Image denoising

$$\mathbf{x} \underset{f^{-1}}{\overset{f}{\rightleftarrows}} \mathbf{y}$$

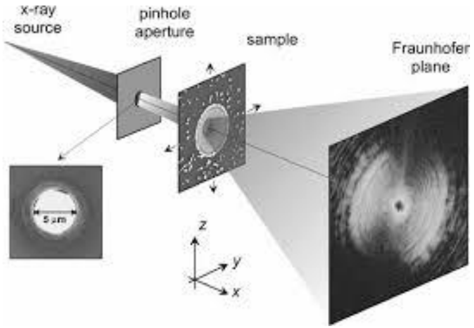


MRI reconstruction

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

Limitations:

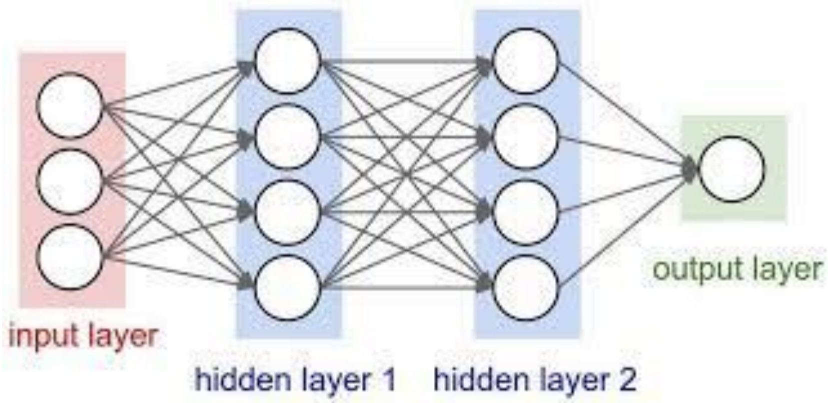
- Which ℓ ? (e.g., unknown/compound noise)
- Which R ? (e.g., structures not amenable to math description)
- Speed

DL methods: the radical way

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

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

$\mathbf{y}_i$



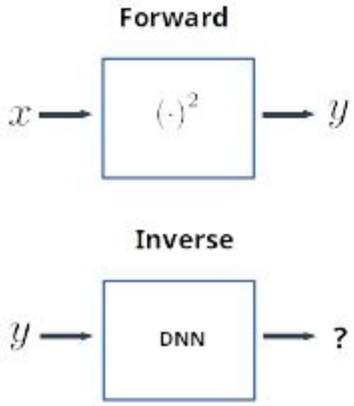
$\mathbf{x}_i$

Limitations:

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

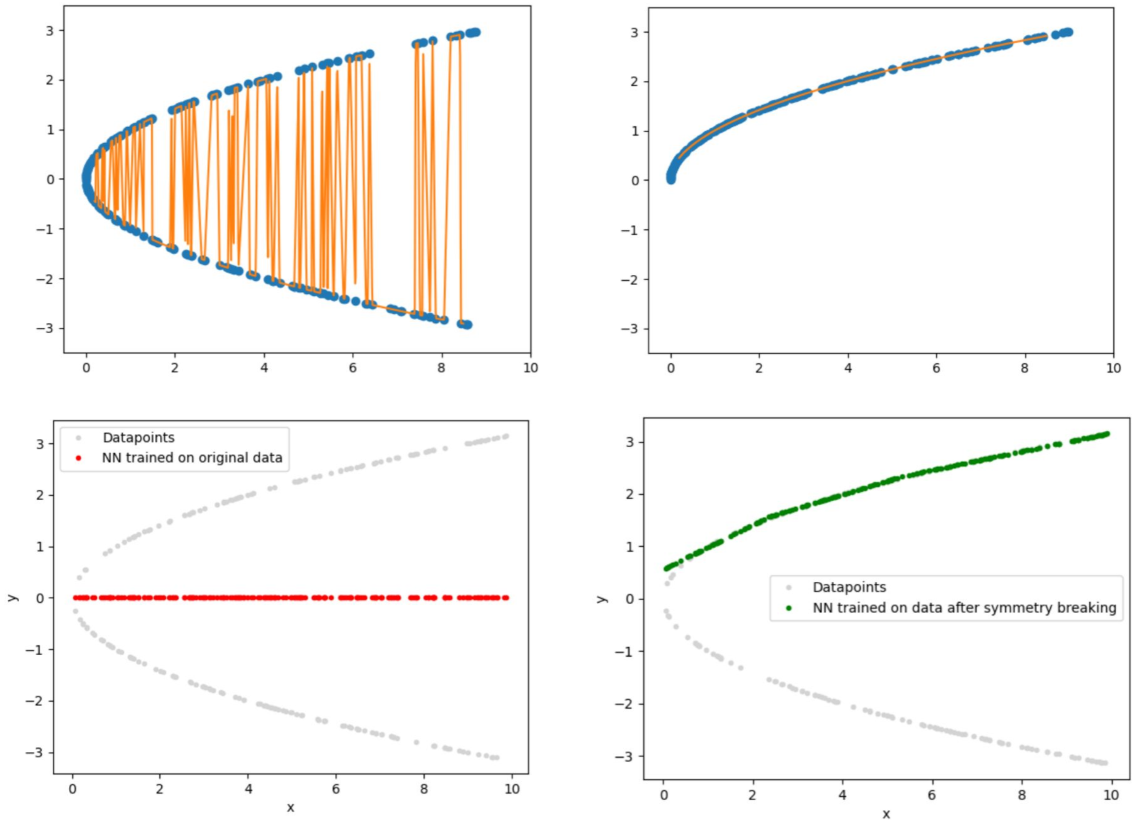
(see, e.g., Tayal et al. **Inverse Problems, Deep Learning, and Symmetry Breaking**. https://arxiv.org/abs/2003.09077)

Why symmetries hurt?



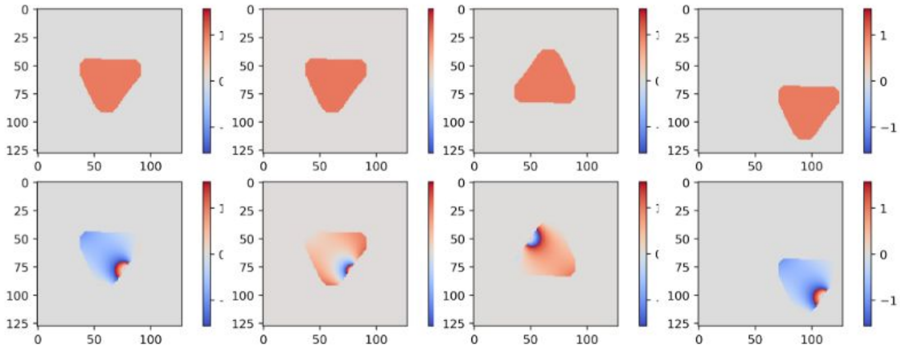
Original Dataset

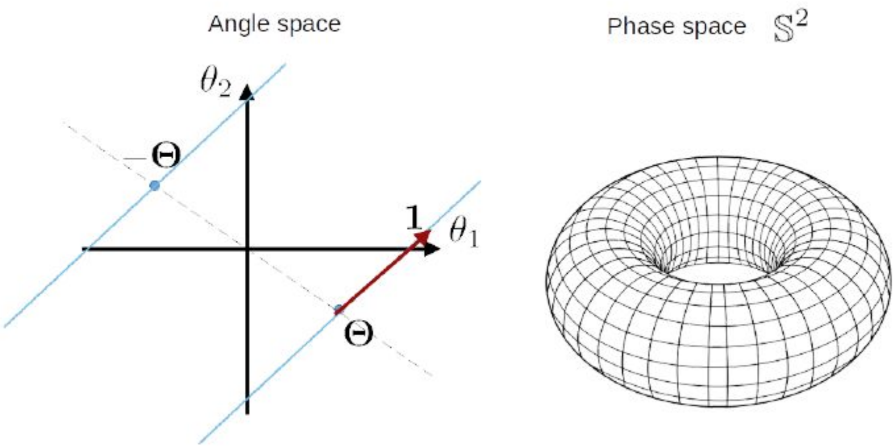
4	-2
9	3
16	-4
100	10
4	2
64	-8
36	6



Symmetry breaking

Idea: process the training set to **break the symmetries** and hence minimize oscillations





Three symmetries: **global phase, conjugate flipping, shift**

Tayal et al. **Unlocking Inverse Problems Using Deep Learning: Breaking Symmetries in Phase Retrieval**. 2020 https://sunju.org/pub/NIPS20-WS-DL4INV.pdf

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

<u>Recipe</u>: 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

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

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

E.g.,

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

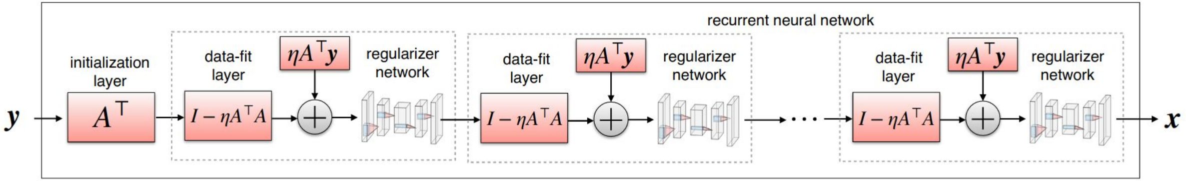


Fig credit: Deep Learning Techniques for Inverse Problems in Imaging https://arxiv.org/abs/2005.06001

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

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

$$\Downarrow$$

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

zero-training data!

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

Contrasting: Deep generative models

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

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

Successes of DIP

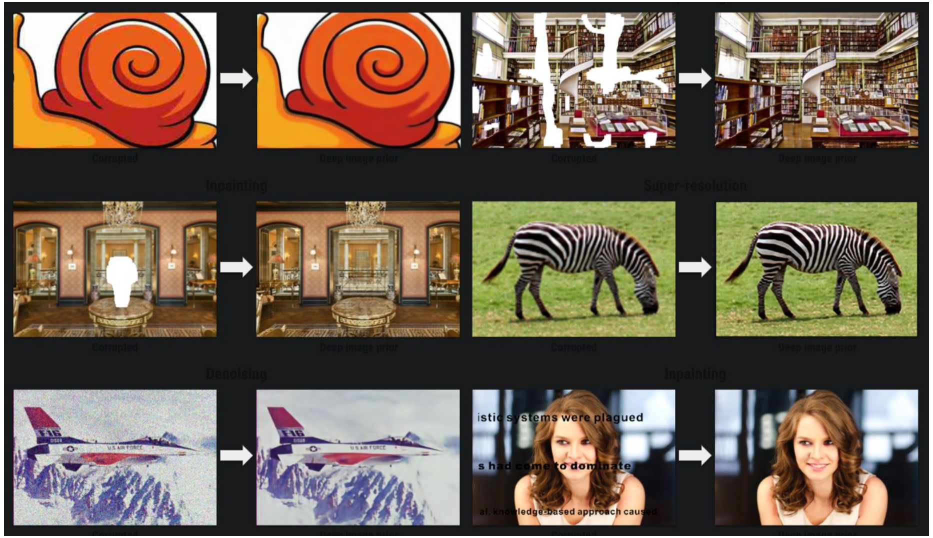
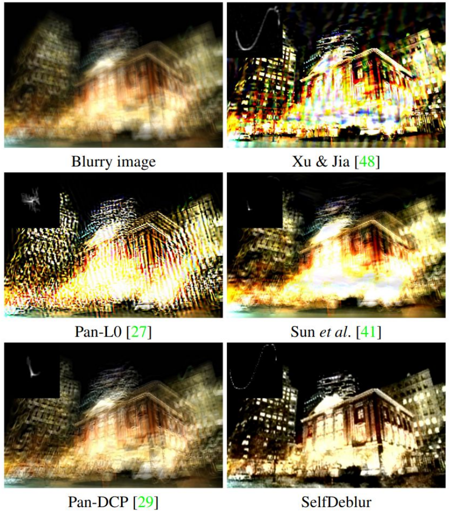


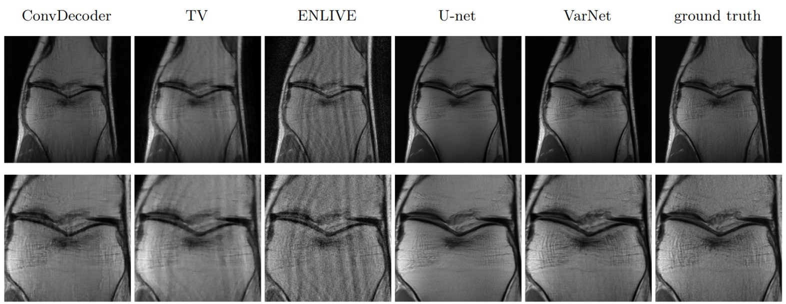
Image denoising/inpainting/super-resol/deJEPG/…



Blind image deblurring (blind deconvolution)

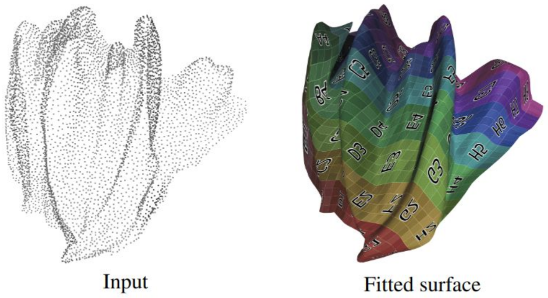
https://dmitryulyanov.github.io/deep_image_prior

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



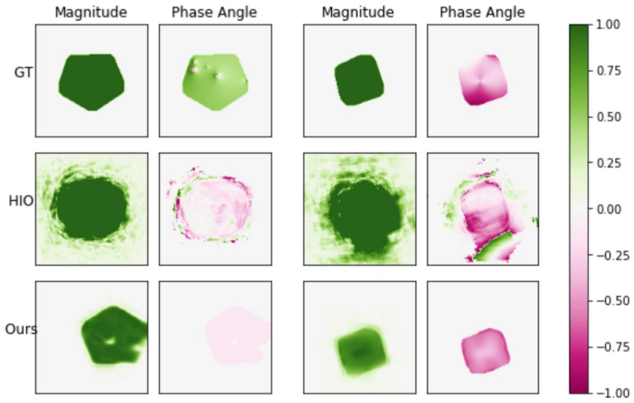
MRI reconstruction

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



Surface reconstruction

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



Phase retrieval

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

Many others:

- PET reconstruction
- Audio denoising
- Time series

See recent survey

Qayyum et al.

Untrained neural network priors for inverse imaging problems: A survey.

https://www.techrxiv.org/articles/preprint/Untrained_Neural_Network_Priors_for_Inverse_Imaging_Problems_A_Survey/14208215

DIP's cousin(s)

Deep image prior (DIP)

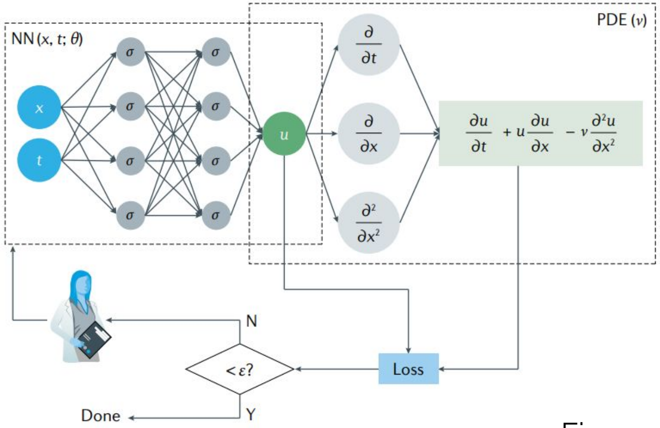
$\mathbf{x} \approx G_\theta(\mathbf{z})$ $\quad G_\theta$ (and $\mathbf{z}$) trainable

Idea: (visual) objects as continuous functions

Neural implicit representation (NIR)

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

Physics-informed neural networks (PINN)



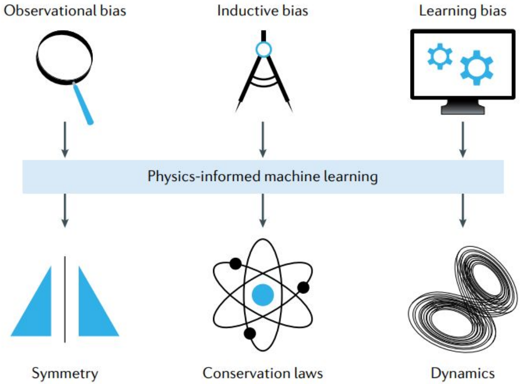
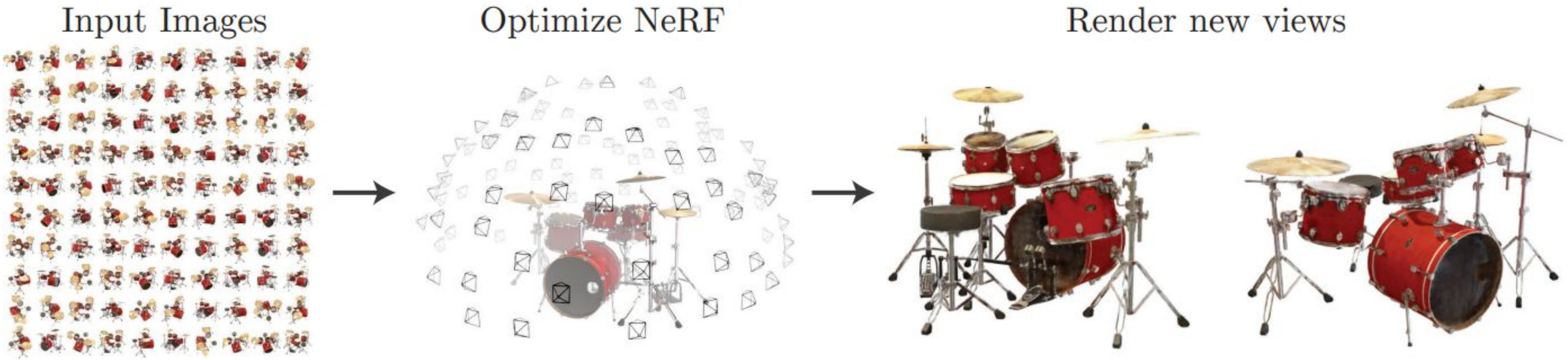
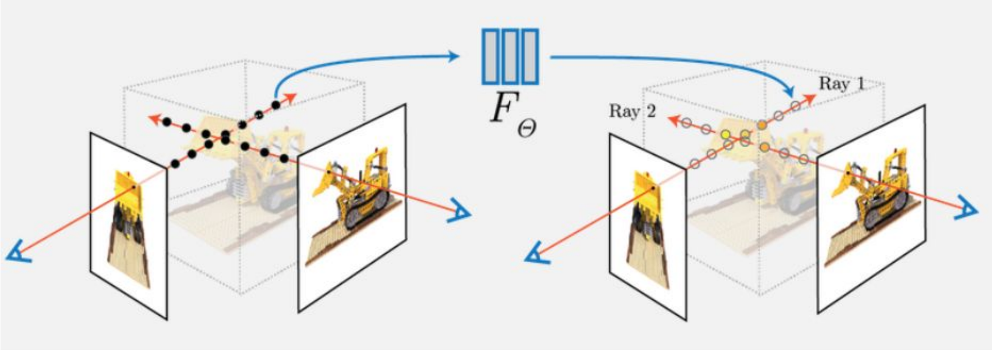
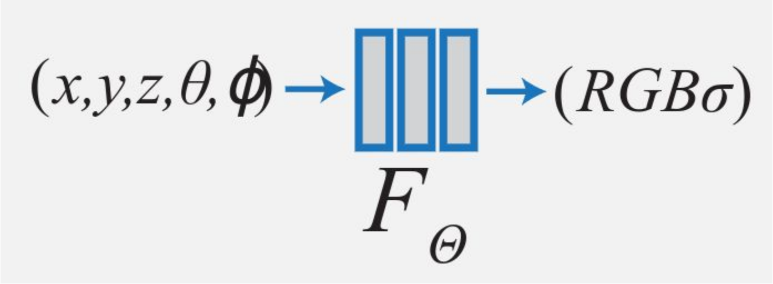


Figure credit: https://www.nature.com/articles/s42254-021-00314-5

NIR for 3D rendering and view synthesis







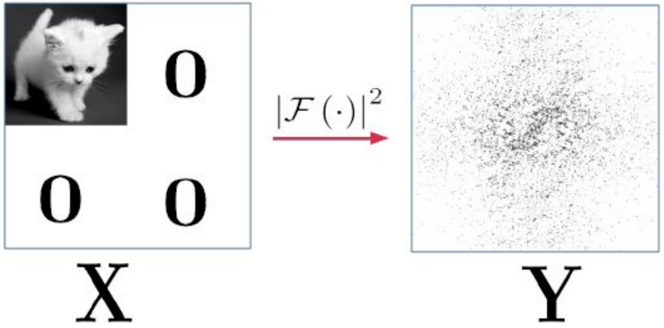
https://www.matthewtancik.com/nerf

DIP for Practical PR

Success of alternating projection for PR

$$\boldsymbol{X} \in \mathbb{C}^{n\times n}, \boldsymbol{Y} = |\mathcal{F}(\boldsymbol{X})|^2 \in \mathbb{R}^{m\times m},\ m \geq 2n-1$$

$$\min_{\boldsymbol{X}} \ \|\boldsymbol{Y} - |\mathcal{F}(\boldsymbol{X})|^2\|_F^2$$ **fails**

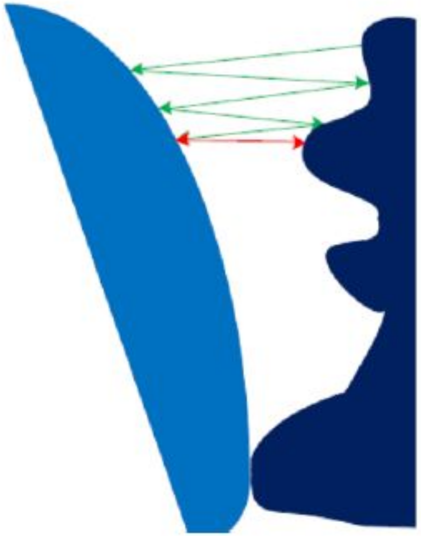


$\mathcal{F}$—oversampled Fourier transform

Find $\boldsymbol{X} \in \boxed{\mathbb{C}^{m\times m}}$

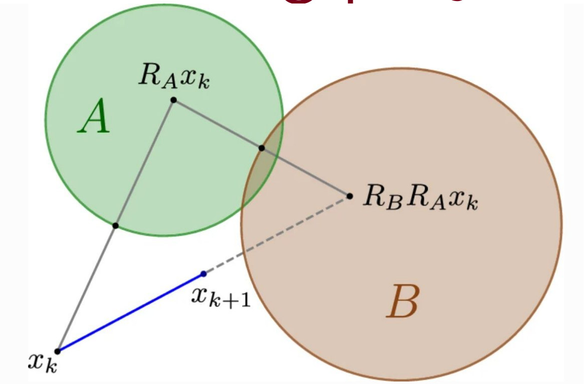
– $\mathcal{M}$ constraint: $|\mathcal{F}(\boldsymbol{X})|^2 = \boldsymbol{Y}$

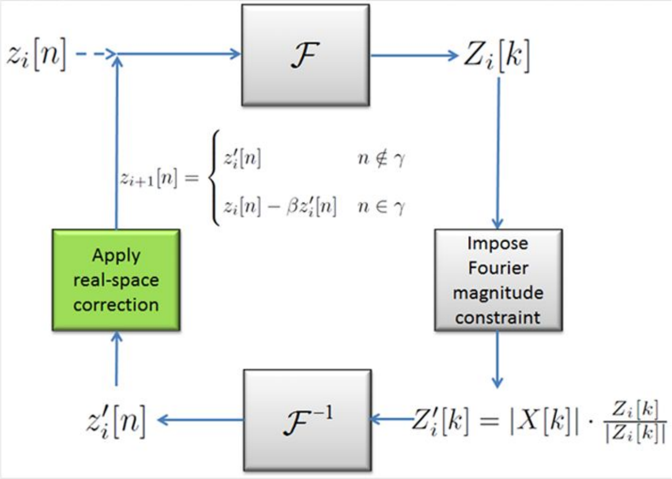
– $\mathcal{S}$ constraint: $\mathcal{A}(\boldsymbol{X}) = \boldsymbol{0}$



Alternating projection

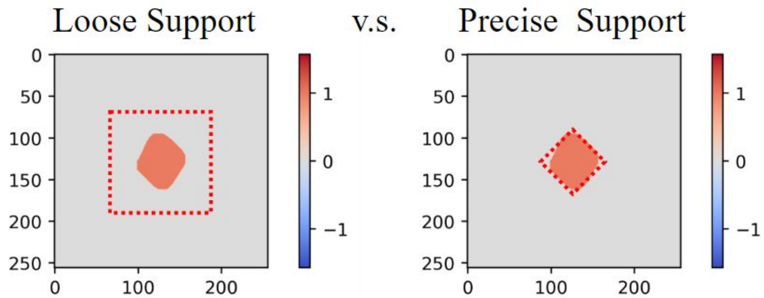
Alternating projection in action





Generalized alternating projection

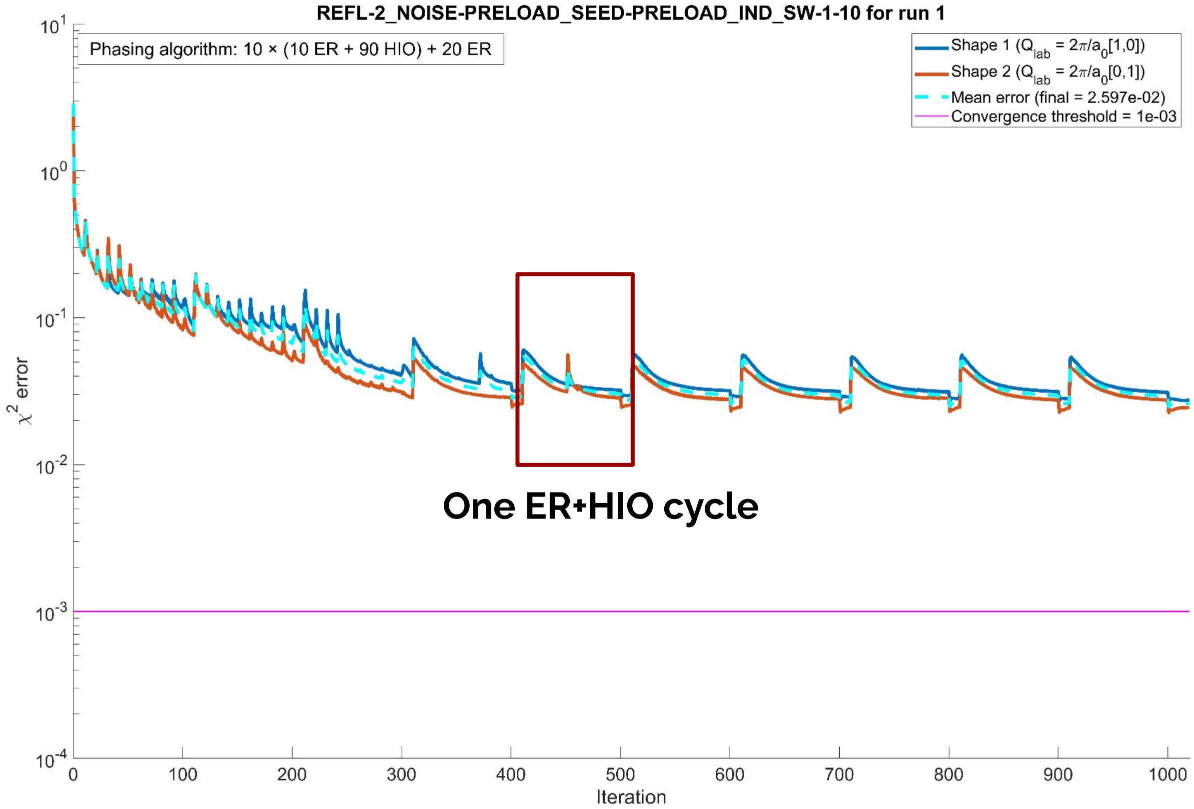
- Hybrid input-output (HIO)
- Relaxed averaged alternating reflections (RAAR)
- Difference map
- Alternating direction of method of multiplier (ADMM)



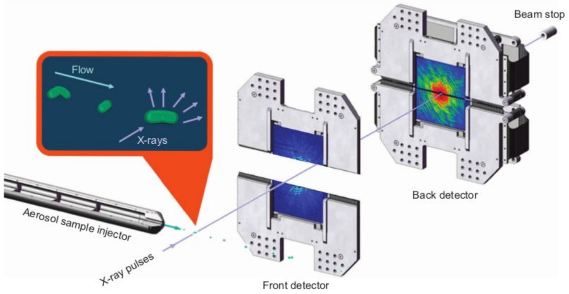
See: **Luke et al'19** https://doi.org/10.1137/18M1193025
Fannjiang & Strohmer'20 https://doi.org/10.1017/S0962492920000069

Loose initial support? **Shrinkwrap**

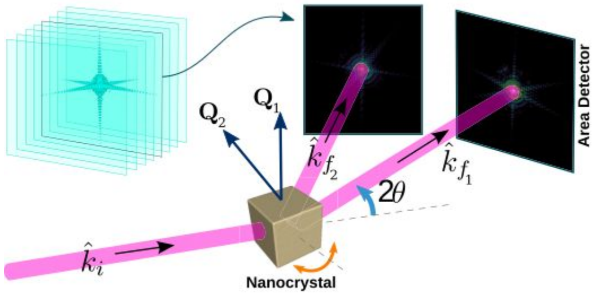
Alternating projection in action



One ER+HIO cycle



Coherent Diffraction Imaging (CDI)



Bragg Coherent Diffraction Imaging (BCDI)

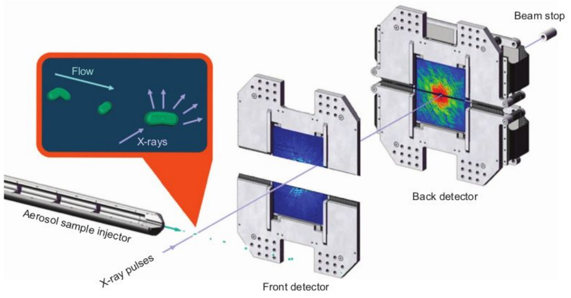
Limitations of SOTA methods on PR

Global issues

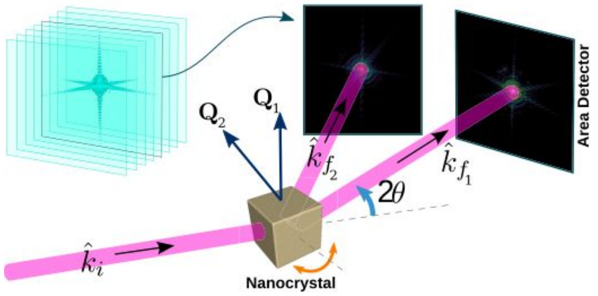
- Sensitivity to initial support estimation
- Sensitivity to **multiple hyperparameters** (e.g., HIO+ER+Shrikwrap)
- Low reconstruction quality (e.g., phases with singularities in BCDI)

Local issues

- Beamstop (i.e., missing data)
- Shot noise



Coherent Diffraction Imaging (CDI)



Bragg Coherent Diffraction Imaging (BCDI)

Back to the LS formulation

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

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

$$\Downarrow$$

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

zero-training data!

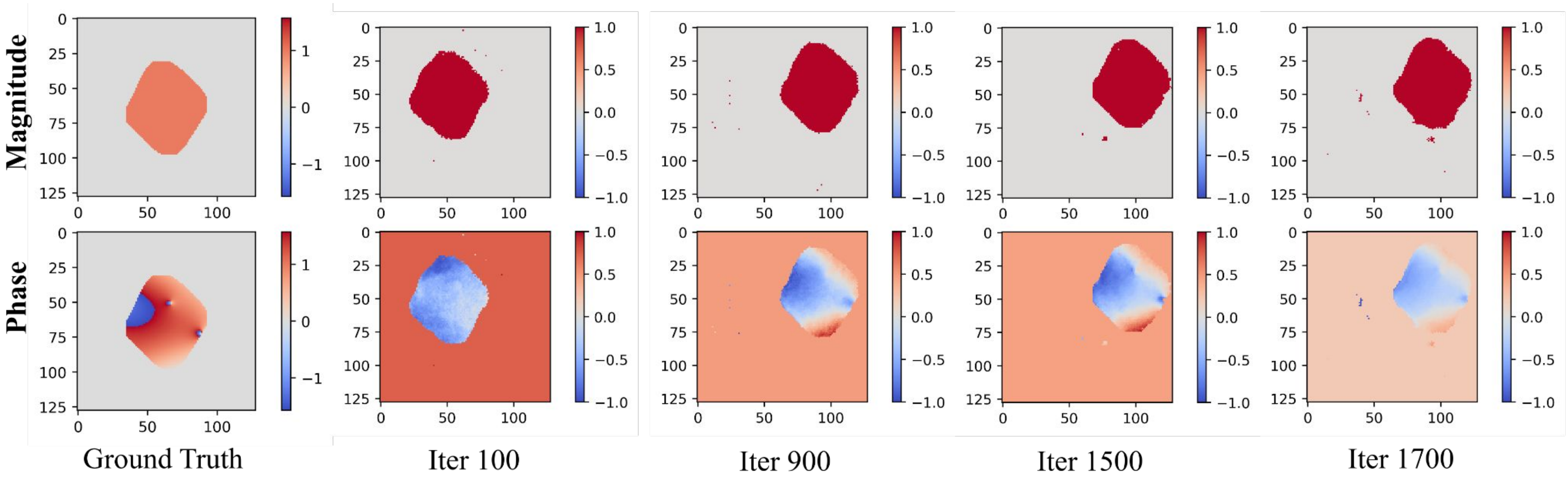
Ulyanov et al. **Deep image prior**. IJCV'20. https://arxiv.org/abs/1711.10925

$$\min_{\boldsymbol{X} \in \mathbb{C}^{n \times n}} \|\boldsymbol{Y} - |\mathcal{F}(\boldsymbol{X})|^2\|_F^2$$

$$\Downarrow$$

$$\min_{\boldsymbol{\theta}} \|\boldsymbol{Y} - |\mathcal{F} \circ G_{\boldsymbol{\theta}}(\boldsymbol{z})|^2\|_F^2$$

Single DIP doesn't work



Asymmetry between magnitude and complex phase

Double DIPs

$$\min_{\boldsymbol{X}\in\mathbb{C}^{n\times n}} \quad \|\boldsymbol{Y} - |\mathcal{F}(\boldsymbol{X})|^2\|_F^2$$

Reparameterizing X using two DIPs

$$\boldsymbol{X} = \boldsymbol{X}^{mag} e^{1j*\boldsymbol{X}^{phase}} = G_{\theta_1}^{mag}(z_1)\, e^{1j*G_{\theta_2}^{phase}(z_2)}$$

e.g., for BCDI on crystals, **magnitude known to be uniform**

OR

$$\boldsymbol{X} = \boldsymbol{X}^{real} + 1j * \boldsymbol{X}^{imag} = G_{\theta_1}^{real}(z_1) + 1j * G_{\theta_2}^{imag}(z_2)$$

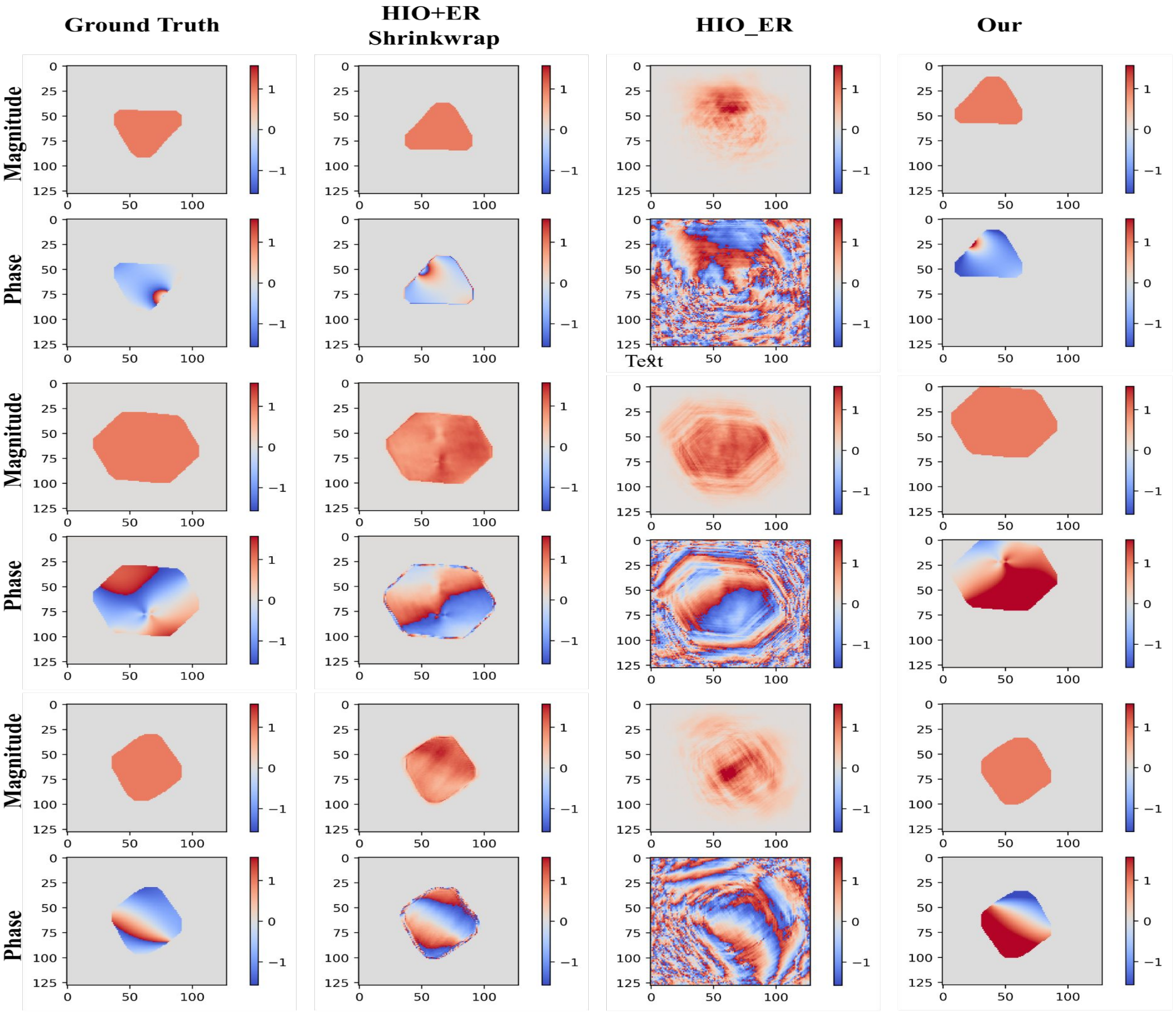
e.g., for CDI on certain bio-specimen, real part **known to be nonnegative**

X half of the size of Y in any dimension: no tight support needed

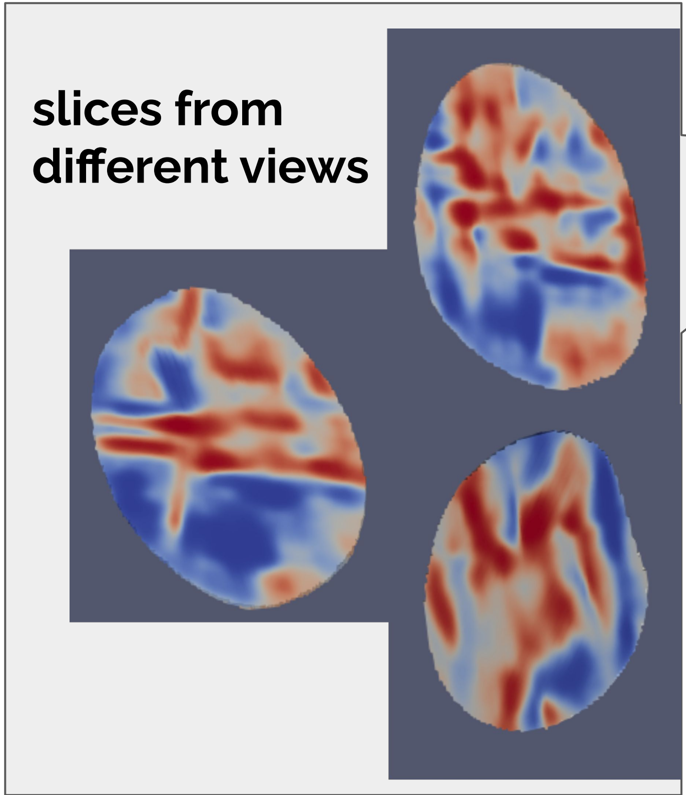
Zhuang et al. **Practical Phase Retrieval Using Double Deep Image Priors**. (Forthcoming)

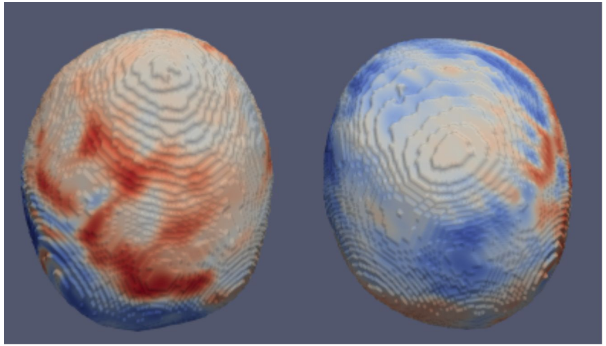
Results on 2D simulated data

zero-training data!

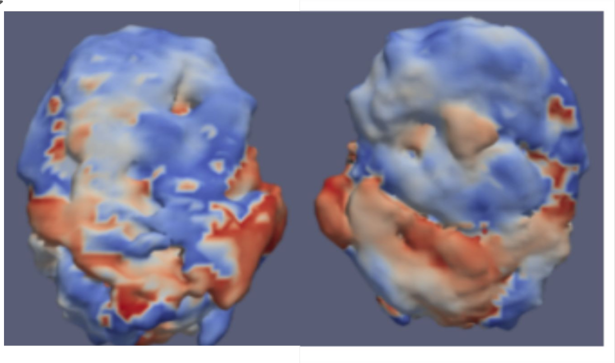


Realistic 3D results (single-reflection)





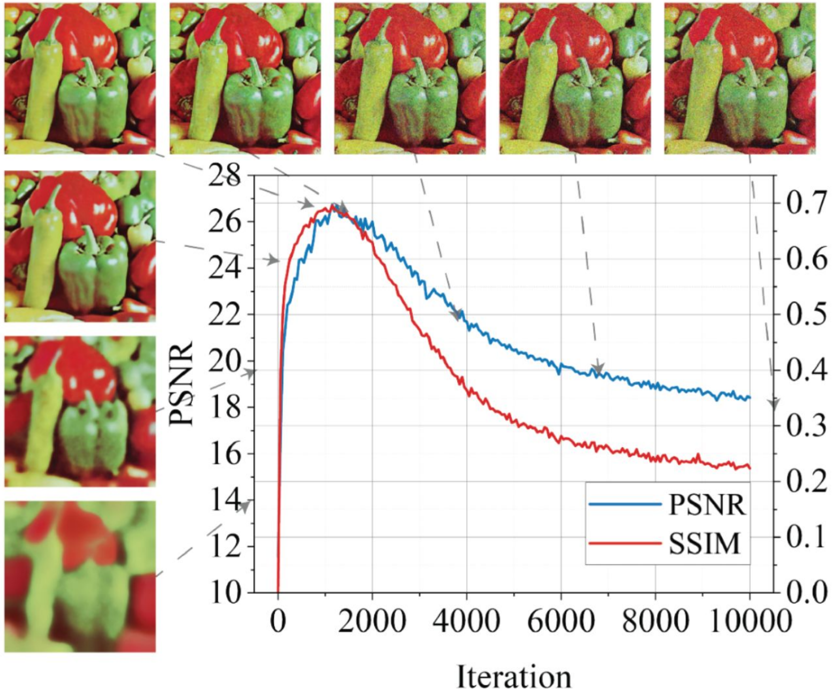
Our

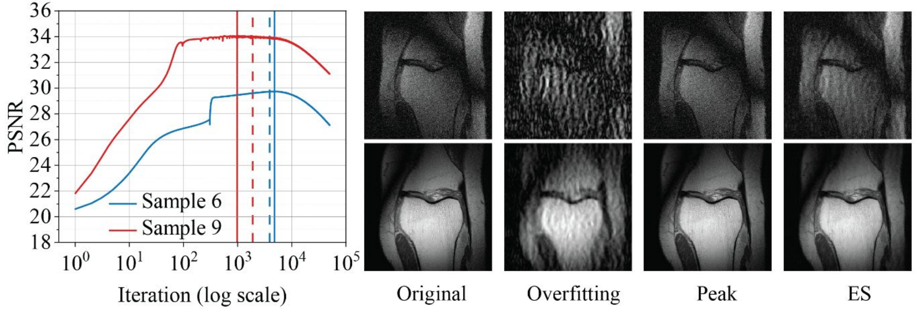


HIO+ER with Shrinkwrap

Yang et al. **Application of single-instance deep generative priors for reconstruction of highly strained gold microcrystals in Bragg coherent X-ray diffraction**. (Forthcoming)

Practical issues around DIP (and its cousin)



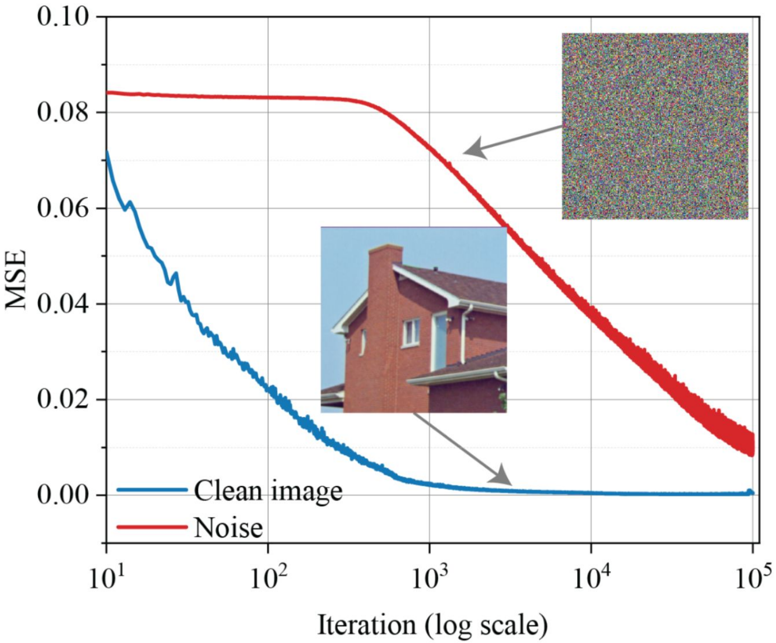


1) **Early learning then overfitting (ELTO)**
2) **Slow in convergence**
3) **Which G_θ?**

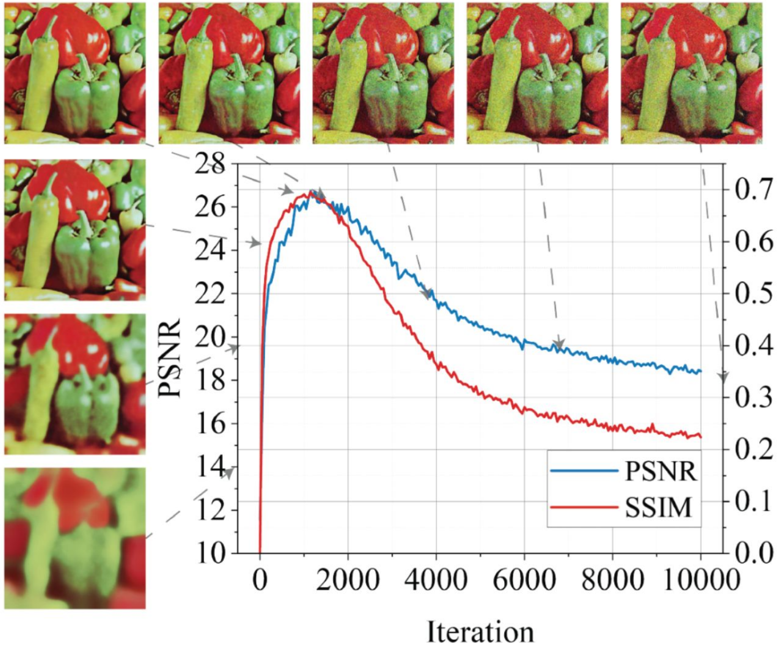
Why early-learning-then-overfitting (ELTO)?

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

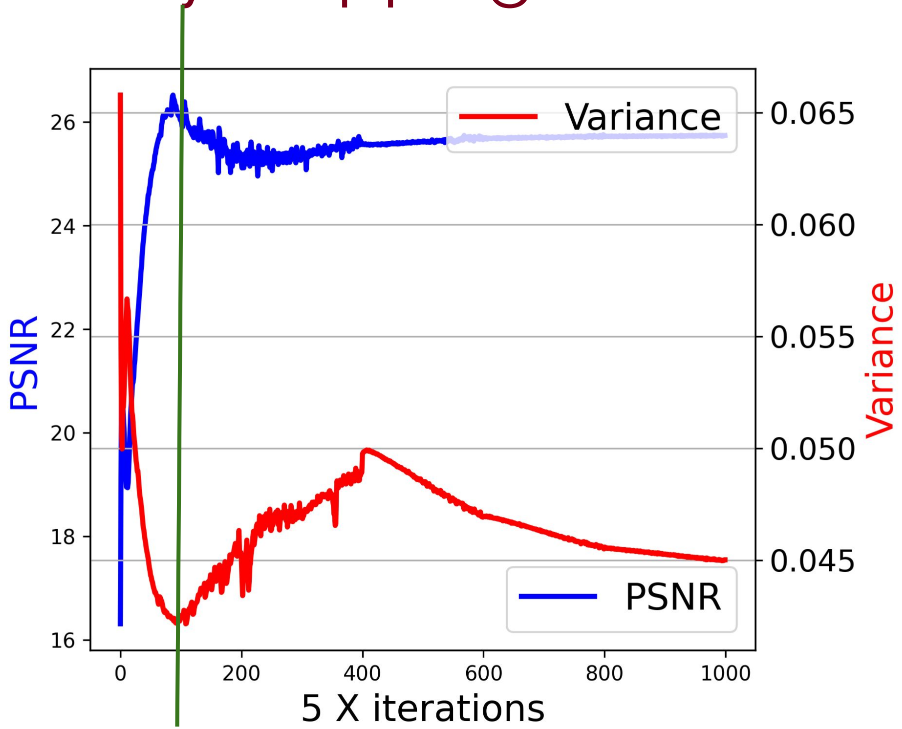
DIP learns signal **much faster than** learning noise



In practice, DIP heavily over-parameterized



Early stopping (ES)



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

Track running variance of estimates

Little over-head!

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)

- 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

Acceleration—random projector (RP)

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

Idea 1:

DIP: random z, trainable G
RP: random G, trainable z

Idea 2:

Reduce G, and put additional regularization

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

Li et al. **Random Projector: Toward Efficient Deep Image Prior**. (forthcoming)

Blind deconvolution

$$\mathcal{Y} = \mathcal{A}_0 \otimes \mathcal{X}_0 + \mathcal{Z},$$

Noise

Kernel/motif

pport /activation

m

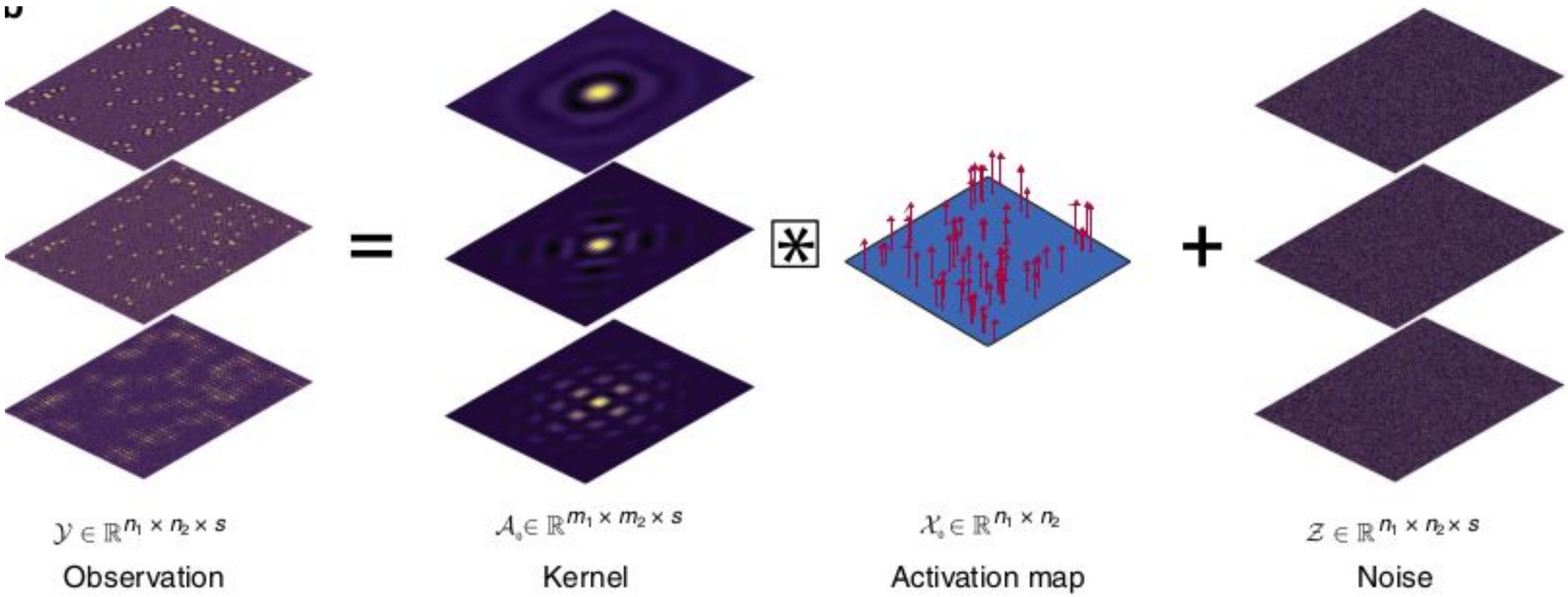
Sparse (containing many zeros)

Solve:

$$\arg \min_{\mathcal{A}} \min_{\mathcal{X}} \left[\psi_\lambda(\mathcal{A}, \mathcal{X}) \equiv \frac{1}{2} \|\mathcal{A} \otimes \mathcal{X} - \mathcal{Y}\|_F^2 + \lambda r(\mathcal{X}), \right]$$

Promoting sparsity

Scanning tunneling microscopy (STM)

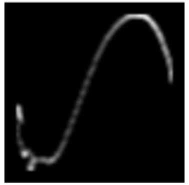


Cheung, Sky C., et al. "Dictionary learning in Fourier-transform scanning tunneling spectroscopy." *Nature communications* 11.1 (2020): 1-11.
https://www.nature.com/articles/s41467-020-14633-1

Blind image deconvolution (BID)

y

x



$$\nabla y = \nabla x \otimes k$$

Our recent work on BID

Tackle challenges about:

- **Unknown kernel size**
- Unknown noise type/level
- Both

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

$$\min_{\boldsymbol{\theta}_{\boldsymbol{k}},\boldsymbol{\theta}_{\boldsymbol{x}}} \ell(\boldsymbol{y}, G_{\boldsymbol{\theta}_{\boldsymbol{k}}}(\boldsymbol{z}_{\boldsymbol{k}}) * G_{\boldsymbol{\theta}_{\boldsymbol{x}}}(\boldsymbol{z}_{\boldsymbol{x}})) + \lambda_{\boldsymbol{k}} R_{\boldsymbol{k}} \circ G_{\boldsymbol{\theta}_{\boldsymbol{k}}}(\boldsymbol{z}_{\boldsymbol{k}}) + \lambda_{\boldsymbol{x}} R_{\boldsymbol{x}} \circ G_{\boldsymbol{\theta}_{\boldsymbol{x}}}(\boldsymbol{z}_{\boldsymbol{x}}),$$

arXiv > eess > arXiv:2208.09483

Search...

Help | Advance

Electrical Engineering and Systems Science > Image and Video Processing

[Submitted on 18 Aug 2022]

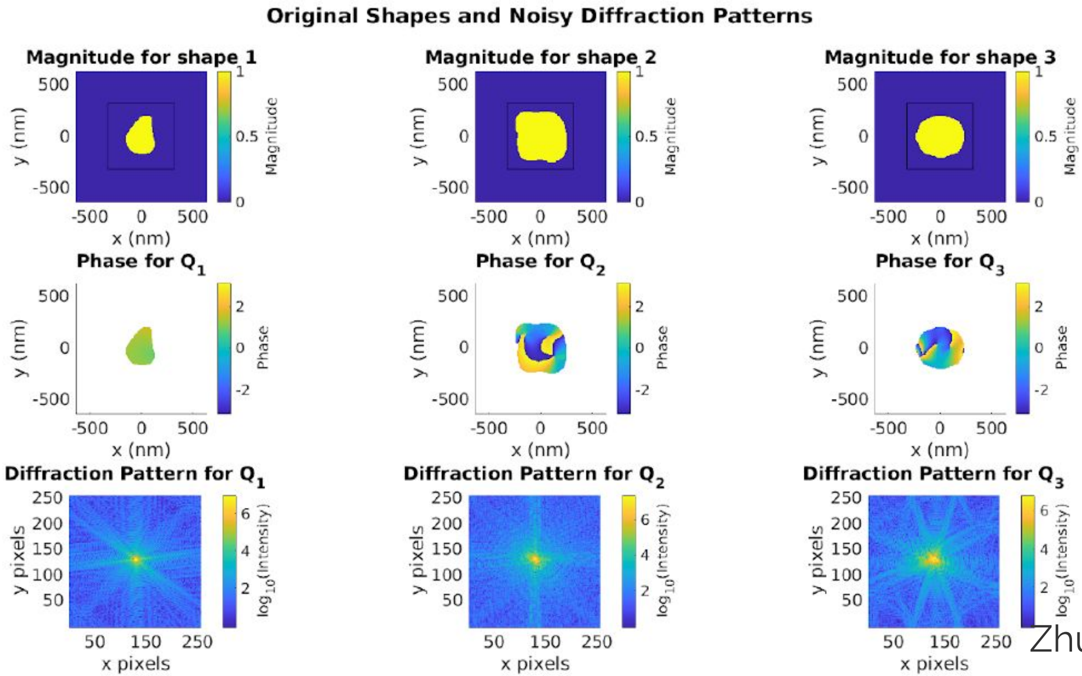
Blind Image Deblurring with Unknown Kernel Size and Substantial Noise

Zhong Zhuang, Taihui Li, Hengkang Wang, Ju Sun

Blind image deblurring (BID) has been extensively studied in computer vision and adjacent fields. Modern methods for BID can be grouped into two categories: single-instance methods that deal with individual instances using statistical inference and numerical optimization, and data-driven methods that train deep-learning models to deblur future instances directly. Data-driven methods can be free from the difficulty in deriving accurate blur models, but are fundamentally limited by the diversity and quality of the training data -- collecting sufficiently expressive and realistic training data is a standing challenge. In this paper, we focus on single-instance methods that remain competitive and indispensable. However, most such methods do not prescribe how to deal with unknown kernel size and substantial noise, precluding practical deployment. Indeed, we show that several state-of-the-art (SOTA) single-instance methods are unstable when the kernel size is overspecified, and/or the noise level is high. On the

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

"ImageNet" for PR?



Easy ⟹ **Difficult**

[//www.image-net.org/](http://www.image-net.org/)

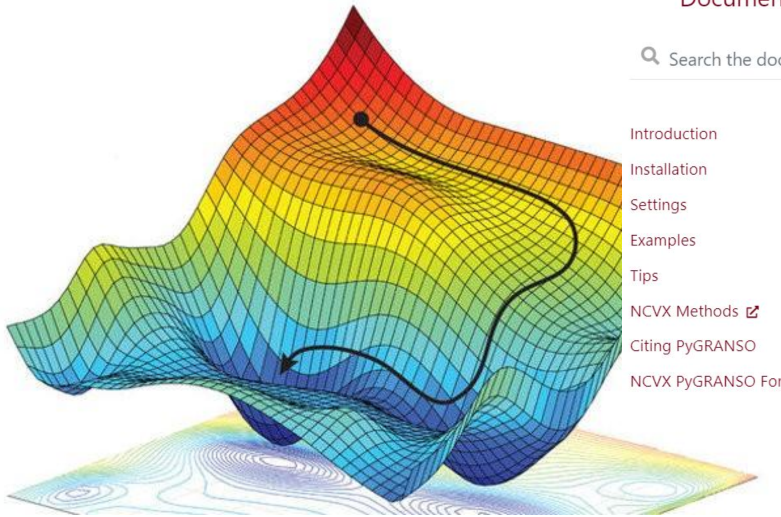
Shapes: polygon, convex, nonconvex

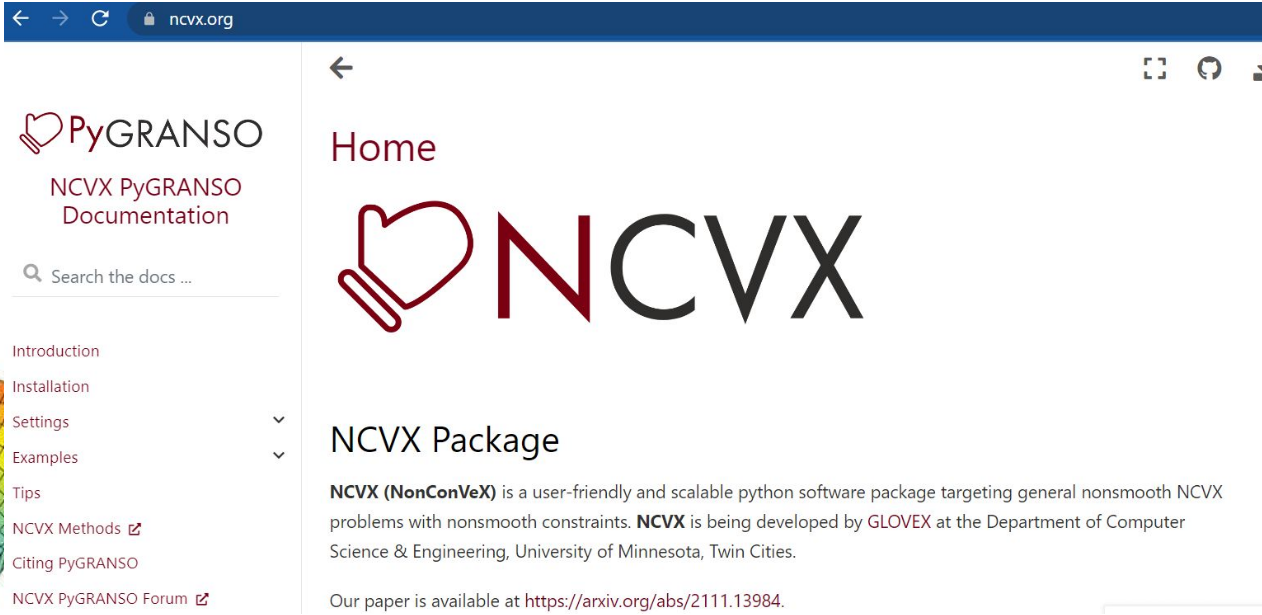
Defects: varying number and density

~ 100K samples!

Zhuang et al. **A realistic phase retrieval dataset for crystals**. (Forthcoming)

Constrained DL





min f(x) s.t. g(x) <= 0

- Auto-differentiation
- GPU support
- Tensor variable support

Both f and g can be nonconvex, non-differentiable, **potentially involving deep neural networks**

Liang et al. **NCVX: A General-Purpose Optimization Solver for Constrained Machine and Deep Learning**. https://arxiv.org/abs/2210.00973

Quick examples

$$\min_{\boldsymbol{w},b,\zeta} \frac{1}{2}\|\boldsymbol{w}\|^2 + C\sum_{i=1}^{n}\zeta_i$$

$$\text{s.t.} \quad y_i\left(\boldsymbol{w}^\intercal \boldsymbol{x}_i + b\right) \geq 1 - \zeta_i,\ \zeta_i \geq 0 \ \forall i = 1, \dots, n$$

```
def comb_fn(X_struct):
    # obtain optimization variables
    w = X_struct.w
    b = X_struct.b
    zeta = X_struct.zeta
    # objective function
    f = 0.5*w.T@w + C*torch.sum(zeta)
    # inequality constraints
    ci = pygransoStruct()
    ci.c1 = 1 - zeta - y*(x@w+b)
    ci.c2 = -zeta
    # equality constraint
    ce = None
    return [f,ci,ce]
# specify optimization variables
var_in = {"w": [d,1], "b": [1,1], "zeta": [n,1]}
# pygranso main algorithm
soln = pygranso(var_in,comb_fn)
```

$$\begin{aligned} \max_{\boldsymbol{x}'} \quad & \ell(\boldsymbol{y}, f_{\boldsymbol{\theta}}(\boldsymbol{x}')) \\ \text{s. t.} \quad & d(\boldsymbol{x}, \boldsymbol{x}') \leq \epsilon \\ & \boldsymbol{x}' \in [0,1]^n \end{aligned}$$

```
def comb_fn(X_struct):
    # obtain optimization variables
    x_prime = X_struct.x_prime
    # objective function
    f = loss_func(y, f_theta(x_prime))
    # inequality constraints
    ci = pygransoStruct()
    ci.c1 = d(x,x_prime) - epsilon
    ci.c2 = -x_prime
    ci.c3 = x_prime-1
    # equality constraint
    ce = None
    return [f,ci,ce]
# specify optimization variable (tensor)
var_in = {"x_prime": list(x.shape)}
# pygranso main algorithm
soln = pygranso(var_in,comb_fn)
```

Papers

- Zhuang et al. **Practical Phase Retrieval Using Double Deep Image Priors**. (Forthcoming)
- Yang et al. **Application of single-instance deep generative priors for reconstruction of highly strained gold microcrystals in Bragg coherent X-ray diffraction**. (Forthcoming)
- Zhuang et al. **A realistic phase retrieval dataset for crystals**. (Forthcoming)

DIP at large

- 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
- Zhuang et al. **Blind Image Deblurring with Unknown Kernel Size and Substantial Noise**. https://arxiv.org/abs/2208.09483 (Submitted to IJCV)
- Li et al. **Random Projector: Toward Efficient Deep Image Prior**. (forthcoming)