https://doi.org/10.2352/EI.2024.36.5.MLSI-309


# What's Wrong with End-to-End Learning for Phase Retrieval?

***Wenjie Zhang***[1]***, Yuxiang Wan***[1]***, Zhong Zhuang***[2]***, Ju Sun***[1]
[1]***Department of Computer Science and Engineering, University of Minnesota, Email: {zhan7867,wan01530,jusun}@umn.edu***
[2]***Department of Physics and Astronomy, University of California, Los Angeles, Email: zhong@physics.ucla.edu***

## Abstract

*For nonlinear inverse problems that are prevalent in imaging science, symmetries in the forward model are common. When data-driven deep learning approaches are used to solve such problems, such intrinsic symmetries can cause substantial learning difficulties. In this paper, we explain how such difficulties arise and, more importantly, how to overcome them by preprocessing the training set before any learning, i.e., symmetry breaking. We take the far-field Fourier phase retrieval, which is central to many areas of scientific imaging, as an example and show that symmetric breaking can substantially improve data-driven learning performance. We also formulate the principle of symmetry breaking that can lead to efficient learning.*

## Introduction

Far-field phase retrieval (FFPR) is a nonlinear inverse problem (IP) that is central to scientific imaging [3, 28]. Given $\boldsymbol{Y} = |\mathscr{F}(\boldsymbol{X})|^2 \in \mathbb{R}_+^{M_1 \times M_2}$ ($\mathbb{R}_+$ means nonnegative reals, $\mathscr{F}$ is the oversampled Fourier operator, and $|\cdot|^2$ takes element-wise squared magnitudes and induces nonlinearity), FFPR is about recovering the complex-valued $\boldsymbol{X} \in \mathbb{C}^{N_1 \times N_2}$ from $\boldsymbol{Y}$. Fourier phases are lost in the measurement process because detectors in practical imaging systems cannot record complex phases. To ensure recoverability, $M_1 \geq 2N_1 - 1$ and $M_2 \geq 2N_2 - 1$ are necessary [9, 13]. Under such recoverability condition, FFPR is almost everywhere injective up to three intrinsic symmetries—operations on $\boldsymbol{X}$ that leave $\boldsymbol{Y}$ unchanged: translation, conjugate flipping, and global phase; see Fig. 1.

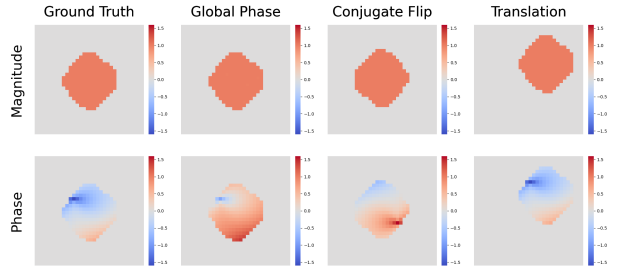


Figure 1: Illustration of the three intrinsic symmetries of FFPR: (i) global phase shift, (ii) 2D conjugate flipping and (iii) translation. Any composition of these three symmetries, when applied to a feasible solution $\boldsymbol{X}$, will lead to the **same measurement** $\boldsymbol{Y}$ in the Fourier domain.

In standard practice, FFPR is solved by meticulously designed iterative numerical methods, e.g., the famous hybrid input-output (HIO) method [7], in combination with the shrinkwrap heuristic [22]. The advent of data-driven deep learning (DL) has created numerous opportunities for novel methods for FFPR. For example, in the end-to-end DL approach, training sets of the form $\{(\boldsymbol{Y}_i, \boldsymbol{X}_i)\}$ are used to train deep neural networks (DNNs) that predict $\boldsymbol{X}$ given $\boldsymbol{Y}$ [4, 8, 14, 24, 33, 35]. Recent work [21, 37] has used pre-trained deep-generative models as plug-in priors to improve recovery performance.

In this paper, we highlight an overlooked learning difficulty associated with the end-to-end approach to FFPR. The difficulty is caused by the three intrinsic forward symmetries. As we illustrate later, these symmetries cause the inverse function that DNNs try to approximate in the end-to-end approach to be highly oscillatory. Such high irregularities cannot be effectively learned by existing DNNs learning techniques. To tame the difficulty, we propose a novel technique to process the training set, which we call *symmetry breaking*. We show that the proposed technique can substantially improve the end-to-end learning performance, regardless of the DNN models used. Preliminary versions of the work have appeared in the workshop paper [32] and a preprint [31].

## Methods

**Symmetry breaking for learning square root** To see why symmetries can cause learning difficulties, consider a simple IP: given $y = x^2$, recover $x$. In other words, we try to learn the square-root function, where we allow both positive and negative outputs. To implement the end-to-end approach, we draw random numbers $x_i$'s uniformly from $[-3, 3]$, and construct the training set $\{x_i^2, x_i\}$. We then train a 6-layer multilayer perceptron (MLP), with 100 hidden nodes in each layer to ensure sufficient capacity and with layer-wise batch normalization to ensure training stability. The functions learned by the MLP are shown in Fig. 2(i) and Fig. 2(ii), on ultra-dense and dense training sets, respectively. In neither case, we can learn a function with reasonable generalization. In particular, for the former case, we learn an almost trivial function. Why more is less here?

The culprit is the intrinsic sign symmetry $\{+x, -x\} \mapsto x^2$. Imagine that we had sampled every point in $[-3, 3]$ when constructing the training set, so the "function" induced by the training set is $x^2 \mapsto \{+x, -x\}$, where the output is an equivalence class induced by the sign symmetry. Practically, we only have a finite training set, with random sampling we will not see both $(x^2, +x)$ and $(x^2, -x)$ in the training set for any $x \in [-3, 3]$. But we can see $(x_1^2, x_1)$ and $(x_2^2, -x_2)$ in the training set where $x_1, x_2 \geq 0$, and $x_1, x_2$ are close. This implies that due to intrinsic sign symmetry, we can see, perhaps in many cases of the training set, that $y_1$ and $y_2$ are close, but their desired outputs differ in sign, and hence far away. Such a close-inputs-distant-outputs property implies that the target function determined by the training set is oscillatory, at least locally. Especially, the larger the training set, the more oscillatory the function can be. In the extreme case, any given DNN gives up fitting the data due to the high oscillations, i.e., what we

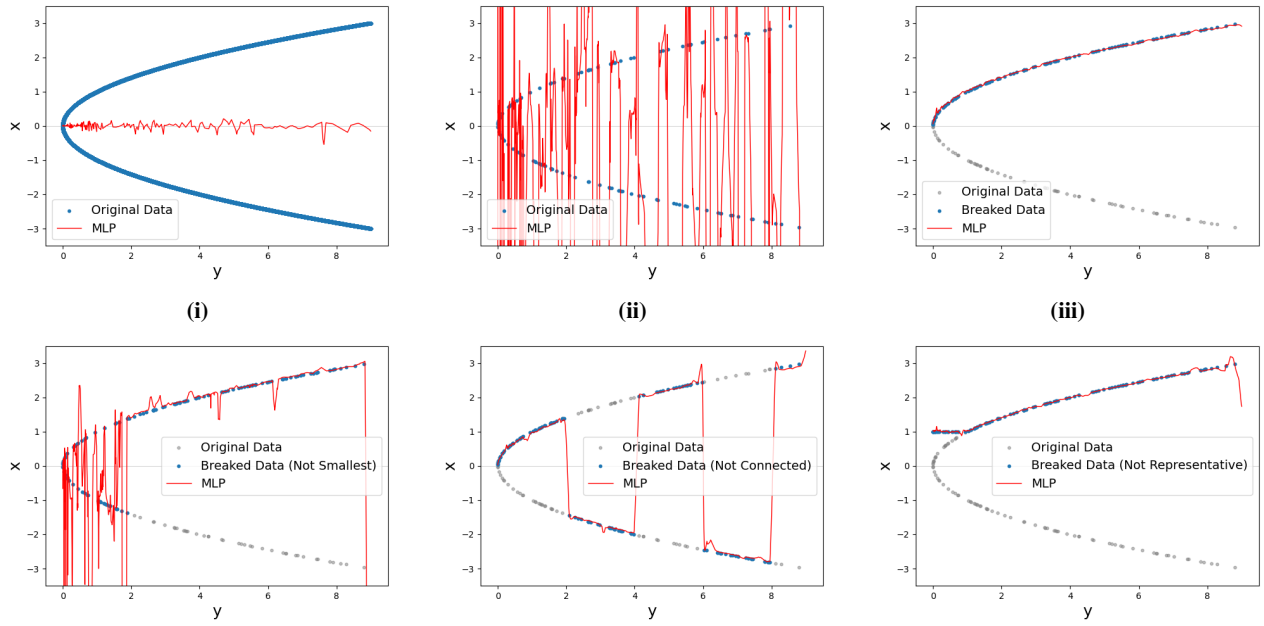


Figure 2: Our proposed *symmetry breaking* process substantially improves end-to-end learning on the square root problem. **Top row**: learning results on the raw training set with ultra-dense sampling (i) and dense sampling (ii), vs. on the training set after symmetry breaking; **Bottom row**: illustration of the three desired properties after symmetry breaking, without which the end-to-end learning can still suffer: (iv) non-smallest, (v) non- connected, and (vi) non-representative.

have seen in Fig. 2(i).

Taming the difficulty is easy: we can process the training set into $\{x_i^2, |x_i|\}$ (or $\{x_i^2, -|x_i|\}$), i.e., fixing the sign of the outputs. We call this a *symmetry breaking* process. Symmetry breaking leads to a much smoother target function and hence much easier for the DNN to learn (see Fig. 2(iii)).

Although the symmetry-breaking process seems natural, there is a principle behind: imagine the continuous case again where we sample every point in $[-3,3]$. The training set after the symmetry breaking is *representative*—every point in the output space is represented up to the sign symmetry, *smallest*—we do not have more than necessary data points in the processed training set, and *connected*—to minimize potential oscillation in the target function. The three principles are illustrated in Fig. 2(iv)-(vi).

**Symmetry breaking for phase retrieval** A central question now is how to derive an effective symmetry-breaking algorithm for FFPR, which has three symmetries. First of all, just to confirm these symmetries can cause similar oscillation issues like our square root example: consider Fig. 1(iii), i.e., the translated copy. The distance between its 2D conjugate flip version and itself is large, although they have exactly the same measurement. A perturbation argument like we did in the square-root example implies that we can easily face a similar close-inputs-distant-outputs issue.

Breaking the symmetries in the object domain seems hopeless: conjugate 2D flipping and non-zero content translation induce irregular equivalent sets in the object space that are hard to represent. Fortunately, the three symmetries can be equivalently represented in terms of the complex phase $e^{i\boldsymbol{\theta}}$ in the Fourier domain. Let $\mathscr{X}$ denote the oversampled Fourier transform of $\boldsymbol{X}$. Now 1) for 2D translation, any allowable[1] 2D translation $t_1, t_2 \in \mathbb{Z}$ induces the change $\mathscr{X}(k_1,k_2) \mapsto \exp[i2\pi(k_1t_1/M_1 + k_2t_2/M_2)]\mathscr{X}(k_1,k_2)$; 2) conjugate 2D flipping induces the change $\mathscr{X} \mapsto \overline{\mathscr{X}}$, i.e., change to the complex phase $e^{i\boldsymbol{\theta}} \mapsto e^{-i\boldsymbol{\theta}}$; and 3) global phase transfer induces the change $\mathscr{X} \mapsto e^{i\theta}\mathscr{X}$. The change due to 2) is a global sign flipping in the *angle* space of $\Theta$, and due to 3) is a line shifting in the *angle* space of $\Theta$. The equivalent sets are easy to represent in the angle space, but we take an equivalent representation in the complex phase space to avoid the tricky issue of dealing with the $2\pi$ periodicity in the angle space. However, finding a way to represent the equivalent sets of 1) is still tricky, whether considering it from the angle or the phase space.

So our overall strategy is a hybrid of rigorous symmetry breaking for global phase shift and 2D conjugate flip in the complex phase space, and *heuristic* "symmetry breaking" for translation in the original space–our later experiments show that this combination is effective. To break 2D translation, we propose simply centering the non-zero content as a heuristic. To break global phase shift and 2D conjugate flip, we perform the geometric construction in the angle space and then translate it back to the phase-space representation. To save space, we omit the intuition behind the construction and directly present the results as follows.

Consider the following set in the phase domain $\mathbb{S}^{M_1\times M_2}$ (the space of $e^{i\boldsymbol{\theta}}$), where $\mathbb{S}$ denotes the 1D complex circle:

$$\begin{aligned}\mathscr{H} \doteq \{\boldsymbol{\Omega} \in \mathbb{C}^{M_1\times M_2} \,:\, &\boldsymbol{\Omega}(1,1) = 1,\\ &\boldsymbol{\Omega}(1,2) \in \mathbb{S}_+, \boldsymbol{\Omega}(i,j) \in \mathbb{S}\ \forall \text{ other index } (i,j)\},\end{aligned} \tag{1}$$

Here, $\mathbb{S}_+$ the upper half circle. Formally, $\mathscr{H}$ can be understood

[1] The nonzero content cannot translate outside the boundaries.

as a set

$$\begin{bmatrix} \{1\} & \mathbb{S}_+ & \mathbb{S} & \cdots & \mathbb{S} \\ \mathbb{S} & \mathbb{S} & \mathbb{S} & \cdots & \mathbb{S} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \mathbb{S} & \mathbb{S} & \mathbb{S} & \cdots & \mathbb{S} \end{bmatrix}_{M_1 \times M_2}. \tag{2}$$

We can prove the following:

**Proposition 1** *Consider the conjugate flipping and global phase transfer symmetries only. The set $\mathscr{H}$ is a connected, smallest representative in the phase domain $\mathbb{S}$ with a negligible set $\mathscr{N} = \{1\} \times \{\omega \in \mathbb{S} : \mathrm{Im}(\omega) = 0\}^{M_1 \times M_2 - 1}$.*

To apply this, we work with end-to-end DNNs that directly predict the $N_1 \times N_2$ target in the object domain. We first center the nonzero content inside $\boldsymbol{X}_j$'s in the training set and then take the oversampled Fourier transform and perform the *symmetry breaking* as implied by Proposition 1 in the complex phase space. For any phase matrix $\boldsymbol{\Omega}$, the *symmetry breaking* goes naturally as follows: first a global phase transfer is performed to make $\boldsymbol{\Omega}(1,1) = 1$, and then a global angle negation is performed, i.e., $\theta \mapsto -\theta$ if the second angle is negative. Here, we assume the angle has been transferred to the range of $(-\pi, \pi]$. We summarize the whole pipeline in Algorithm 1.

**Algorithm 1** Procedure of symmetry breaking for FPR

**Input:** Forward mapping $\hat{f}(\boldsymbol{X}) \doteq |\mathscr{F}(\boldsymbol{X})|^2$ and randomly sampled input data points $D = \{\boldsymbol{X}_j\} \subset \mathbb{C}^{N_1 \times N_2}$
**Output:** Symmetry breaking training dataset

1: Centering the nonzero content inside $\boldsymbol{X}_j$'s in the original input space. This heuristically breaks 2D translation symmetry.
2: Taking the oversampled Fourier transform of $\boldsymbol{X}$ to get $\mathscr{X}$.
3: Decompose each element of $\mathscr{X}$ into polar form: $\mathscr{X}(k_1,k_2) = \boldsymbol{\rho}(k_1,k_2)e^{i\boldsymbol{\theta}(k_1,k_2)}$ with the phase $\boldsymbol{\Omega}(k_1,k_2) = e^{i\boldsymbol{\theta}(k_1,k_2)}$
4: Breaking symmetry in the phase domain with the simple algorithm $\boldsymbol{\Phi}$ described below.
5: **while** $\boldsymbol{X}_j \in D$ **do** $\boldsymbol{\Phi}$ as the following:
6: Performing global phase transfer to make $\boldsymbol{\Omega}(1,1) = 1$: $\boldsymbol{\Omega}(k_1,k_2) \leftarrow \overline{\boldsymbol{\Omega}(1,1)}\boldsymbol{\Omega}(k_1,k_2)$
7: **if** $\boldsymbol{\Omega}(1,2) \in \mathbb{S}_+$ **then**
8: $\boldsymbol{\Omega}(k_1,k_2) \leftarrow \boldsymbol{\Omega}(k_1,k_2)$
9: **else if** $\boldsymbol{\Omega}(1,2) \notin \mathbb{S}_+$ **then**
10: $\boldsymbol{\Omega}(k_1,k_2) \leftarrow \overline{\boldsymbol{\Omega}(k_1,k_2)}$
11: **end if**
12: **end while**
13: Applying forward mapping $\hat{f}$ on each point $\boldsymbol{X}_j$ and form a new training set $\{(\left|\mathscr{F}(\boldsymbol{X}_j)\right|^2, \boldsymbol{X}_j)\}$, which is a symmetry breaking set.

## Experiments

**Evaluation dataset** We use a simulated Bragg CDI (BCDI) crystal dataset as described in [38] to construct our training, validation, and test sets.

We first quickly review the data generation process: (1) Random convex and nonconvex rounded polygons are generated; (2) Magnitudes inside the polygons are set to 1, and those outside set to 0, to simulate the uniform magnitudes in practical crystal samples; (3) Complex phases inside the polygons are derived by first placing randomly distributed crystal defects and then projecting the resulting two-dimensional displacement fields onto the corresponding momentum transfer vectors. A variety of polygon shapes and defect densities are included in the dataset to ensure its diversity. The complex-valued images are placed in $128 \times 128$ black backgrounds; see the last column of Fig. 3 for samples.

In our experiment, we resize the complex-valued images into $32 \times 32$ to save computation, and perform 2x oversampled Fourier transforms to generate the phaseless measurements. We generate 4 training sets of varying sizes: 500, 1000, 1500 and 5000. Moreover, we independently generate a fixed 500-sample validation set and a 500-sample test set. The original training sets are designated as "before breaking", and those after preprocessing following Algorithm 1 are designated as "after breaking".

**Experiment setup** We choose two DNN backbones, UNet [4] and SiSPRNet [35] that have been used in end-to-end methods for FFPR. Since our argument about learning difficulty is about the training set, and also our symmetry breaking is only performed on the training set, we expect uniform improvement in performance due to the symmetry breaking, regardless of the DNN backbones used. To make these models compatible with our problem size, we adjust the number of decoder and encoder layers in both models. We train both models using standard MSE loss. For baseline, we pick HIO+ER+Shrinkwrap [22], a gold-standard method used in BCDI. To evaluate recovery quality, we use the symmetry-adjusted MSE (SA-MSE) as defined in [32], modulo the intrinsic symmetries.

**Results** We present the results from three settings: (1) SA-MSE evaluated on the training dataset using the final models (Table 1), (2) SA-MSE evaluated on the training dataset with the best validated model (Table 2), and (3) SA-MSE evaluated on the test set with the best validated model (Table 3).

Table 1: Summary of reconstruction quality in terms of the SA-MSE loss for FFPR on the **training set** using the **model obtained at the final training epoch**. For each of the experiments, the models are trained on the dataset with 500, 1000, 1500 and 5000 data points from the simulated 2D crystal dataset both before and after *symmetry breaking*. The SA-MSE loss for HIO+ER+Shrinkwrap is 0.0449.

| | UNet | | SiSPRNet | |
|---|---|---|---|---|
| **Sample** | **Before** | **After** | **Before** | **After** |
| 500 | 0.1806 | 0.0171 | 0.1868 | 0.0269 |
| 1000 | 0.0491 | 0.0115 | 0.1474 | 0.0210 |
| 1500 | 0.0609 | 0.0106 | 0.0697 | 0.0191 |
| 5000 | 0.0165 | 0.0060 | 0.0460 | 0.0120 |

First, we examine how symmetries affect training performance from Table 1. We observe that symmetry breaking uniformly improves the SA-MSE loss across the two backbone models, often by an order of magnitude. This is consistent with the symmetry-induced learning difficulty we expected. Moreover, in terms of SA-MSE loss, training on the raw training

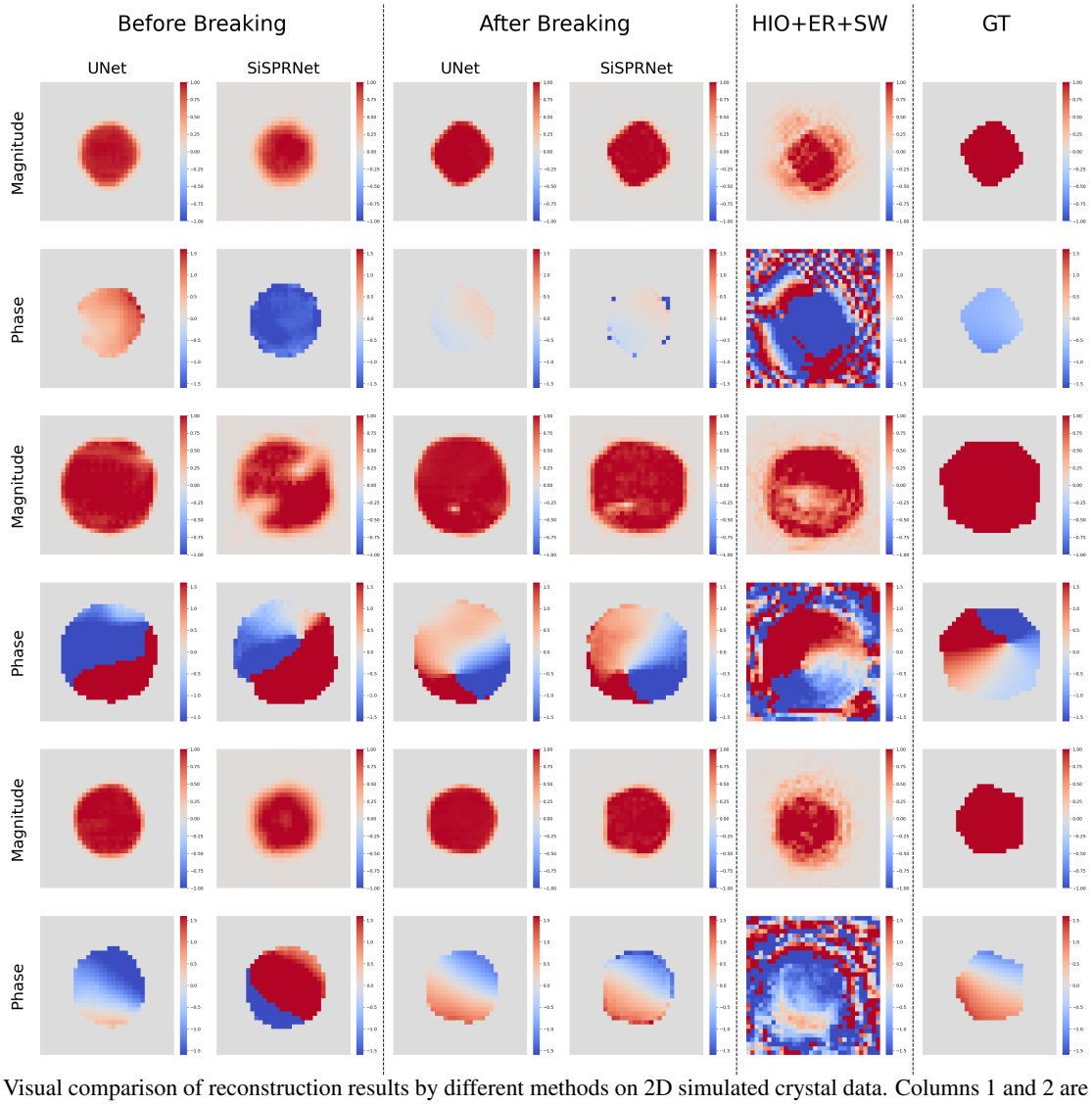


Figure 3: Visual comparison of reconstruction results by different methods on 2D simulated crystal data. Columns 1 and 2 are the results for UNet and SiSPRNet trained on the 5000-sample dataset before symmetry breaking, columns 3 and 4 are the results for these two networks trained on the 5000-sample dataset after symmetry breaking.

MSE loss for FFPR on the **training set** using the **model with the smallest validation loss**. For each of the experiments, the models are trained on the dataset with 500, 1000, 1500 and 5000 data points from the simulated 2D crystal dataset both before and after *symmetry breaking*. The SA-MSE loss for HIO+ER+Shrinkwrap is 0.0449.

| | UNet | | SiSPRNet | |
|---|---|---|---|---|
| **Sample** | **Before** | **After** | **Before** | **After** |
| 500 | 0.1341 | 0.0219 | 0.1525 | 0.0270 |
| 1000 | 0.0665 | 0.0236 | 0.1474 | 0.0212 |
| 1500 | 0.0672 | 0.0215 | 0.0701 | 0.0199 |
| 5000 | 0.0360 | 0.0187 | 0.0463 | 0.0147 |

Table 3: Summary of results in terms of SA-MSE loss for FFPR based on the **test dataset**. For each of the experiments, the models are trained on the dataset with 500, 1000, 1500 and 5000 data points from the simulated 2D crystal dataset both before and after *symmetry breaking*. The SA-MSE loss for HIO+ER+Shrinkwrap is 0.0436.

| | UNet | | SiSPRNet | |
|---|---|---|---|---|
| **Sample** | **Before** | **After** | **Before** | **After** |
| 500 | 0.1210 | 0.0448 | 0.1412 | 0.0426 |
| 1000 | 0.1057 | 0.0460 | 0.1389 | 0.0357 |
| 1500 | 0.1056 | 0.0385 | 0.0733 | 0.0336 |
| 5000 | 0.0580 | 0.0326 | 0.0568 | 0.0255 |

sets often performs even worse than the non-data-driven baseline HIO+ER+Shrinkwrap, despite the large capacity of the backbone models we choose. This suggests that symmetry breaking is a crucial step to take to maximize learning efficiency for data-driven FFPR.

Tables 2 and 3 suggest the performance boost due to symmetry breaking on the training sets carries on to their respective test sets, i.e., test performance is uniformly improved after symmetry breaking. Note that without symmetry breaking, the test performance is always worse than that of non-data-driven HIO+ER+Shrinkwrap baseline, whereas after symmetry breaking, we can start to see the advantage of data-driven methods: they outperform the baseline method.

Fig. 3 visually compares the recovery results before and after symmetry breaking on several randomly chosen test samples. Before symmetry breaking, both backbone models typically only recover roundish shapes, losing the boundary details. After symmetry breaking, for all of the visualized samples, we observe much improved magnitude recovery, especially of the boundary details. The estimated phases also reveal sharper details, again confirming the anticipated performance benefits of symmetry breaking.

## Related work

Recently, there have been intensive research efforts on solving IPs using deep learning [2, 19, 23, 36]. The end-to-end approach is attractive not only because of its simplicity but also because (i) we do not even need to know the forward models, so long as we can gather sufficiently many data samples and weak system properties such as symmetries—e.g., this is handy for complex imaging systems [10, 17]; (ii) or alternatives have rarely worked, and a good example is Fourier PR [7, 29].

Besides these, the end-to-end deep learning approach has been empirically applied to a number of problems with symmetries, e.g., blind image deblurring (i.e., blind deconvolution) [6, 30], real-valued Fourier phase retrieval [29], 3D surface tangents and normal prediction [5, 11], nonrigid structure-from-motion [16, 34], PDE IPs [15, 18, 27], blind source separation [20], single image depth estimation. The difficulty of learning highly oscillatory functions by DNNs is also found in DL for solving partial differential equations [1, 26].

For FFPR, recent work has also tried to integrate DL modules with the traditional regularized data-fitting framework [12, 25]. However, HIO is still needed to produce good initialization, and their methods mostly only perform local refinement.

## Acknowledgments

W. Zhang and Y. Wan are partially supported by a UMN CSE InterS&Ections Seed Grant. Dr. Kshitij Tayal, Dr. Jesse Lai and Raunak Manekar were involved in the early development of the ideas included here, as reported in the preliminary work [31, 32].

## References

[1] Ben Adcock and Nick Dexter. The gap between theory and practice in function approximation with deep neural networks, 2021.

[2] Simon Arridge, Peter Maass, Ozan Öktem, and Carola-Bibiane Schönlieb. Solving inverse problems using data-driven models. *Acta Numerica*, 28:1–174, may 2019.

[3] Tamir Bendory, Robert Beinert, and Yonina C. Eldar. Fourier phase retrieval: Uniqueness and algorithms. In *Compressed Sensing and its Applications*, pages 55–91. Springer International Publishing, 2017.

[4] Mathew J. Cherukara, Youssef S. G. Nashed, and Ross J. Harder. Real-time coherent diffraction inversion using deep generative networks. *Scientific Reports*, 8(1), nov 2018.

[5] Tien Do, Khiem Vuong, and Hyunjong Park. Egocentric scene understanding via multimodal spatial rectifier. *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2822–2831, 2022.

[6] Zhenxuan Fang, Fangfang Wu, Weisheng Dong, Xin Li, Jinjian Wu, and Guangming Shi. Self-supervised non-uniform kernel estimation with flow-based motion prior for blind image deblurring. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 18105–18114, June 2023.

[7] James R Fienup. Phase retrieval algorithms: a comparison. *Applied optics*, 21(15):2758–2769, 1982.

[8] Alexandre Goy, Kwabena Arthur, Shuai Li, and George Barbastathis. Low photon count phase retrieval using deep learning. *Physical Review Letters*, 121(24), dec 2018.

[9] M. Hayes. The reconstruction of a multidimensional sequence from the phase or magnitude of its fourier transform. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 30(2):140–154, apr 1982.

[10] Ryoichi Horisaki, Ryosuke Takagi, and Jun Tanida. Learning-based imaging through scattering media. *Optics Express*, 24(13):13738, jun 2016.

[11] Jingwei Huang, Yichao Zhou, Thomas Funkhouser, and Leonidas J Guibas. Framenet: Learning local canonical frames of 3d surfaces

from a single rgb image. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 8638–8647, 2019.

[12] Çağatay Işıl, Figen S. Oktem, and Aykut Koç. Deep iterative reconstruction for phase retrieval. *Applied Optics*, 58(20):5422, jul 2019.

[13] Kishore Jaganathan, Yonina C. Eldar, and Babak Hassibi. Phase retrieval: An overview of recent developments, 2015.

[14] Iksung Kang, Fucai Zhang, and George Barbastathis. Phase extraction neural network (phenn) with coherent modulation imaging (cmi) for phase retrieval at low photon counts. *Opt. Express*, 28(15):21578–21600, Jul 2020.

[15] Ehsan Kharazmi, Zhongqiang Zhang, and George E.M. Karniadakis. hp-vpinns: Variational physics-informed neural networks with domain decomposition. *Computer Methods in Applied Mechanics and Engineering*, 374:113547, 2021.

[16] Chen Kong and Simon Lucey. Deep non-rigid structure from motion. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1558–1567, 2019.

[17] Yunzhe Li, Yujia Xue, and Lei Tian. Deep speckle correlation: a deep learning approach toward scalable imaging through scattering media. *Optica*, 5(10):1181, sep 2018.

[18] Lu Lu, Raphael Pestourie, Wenjie Yao, Zhicheng Wang, Francesc Verdugo, and Steven G. Johnson. Physics-informed neural networks with hard constraints for inverse design, 2021.

[19] Alice Lucas, Michael Iliadis, Rafael Molina, and Aggelos K. Katsaggelos. Using deep neural networks for inverse problems in imaging: Beyond analytical methods. *IEEE Signal Processing Magazine*, 35(1):20–36, jan 2018.

[20] Weilin Luo, Hongbin Jin, Xiaobai Li, Hao Li, Kang Liu, and Ruijuan Yang. A novel complex-valued blind source separation and its applications in integrated reception. *Electronics*, 12(18), 2023.

[21] Liyuan Ma, Hongxia Wang, Ningyi Leng, and Ziyang Yuan. Admm based fourier phase retrieval with untrained generative prior, 2022.

[22] Stefano Marchesini, H He, Henry N Chapman, Stefan P Hau-Riege, Aleksandr Noy, Malcolm R Howells, Uwe Weierstall, and John CH Spence. X-ray image reconstruction from a diffraction pattern alone. *Physical Review B*, 68(14):140101, 2003.

[23] Michael T. McCann, Kyong Hwan Jin, and Michael Unser. Convolutional neural networks for inverse problems in imaging: A review. *IEEE Signal Processing Magazine*, 34(6):85–95, nov 2017.

[24] Christopher A. Metzler, Felix Heide, Prasana Rangarajan, Muralidhar Madabhushi Balaji, Aparna Viswanath, Ashok Veeraraghavan, and Richard G. Baraniuk. Deep-inverse correlography: towards real-time high-resolution non-line-of-sight imaging. *Optica*, 7(1):63, jan 2020.

[25] Christopher A Metzler, Philip Schniter, Ashok Veeraraghavan, and Richard G Baraniuk. prdeep: Robust phase retrieval with a flexible deep network. *arXiv preprint arXiv:1803.00212*, 2018.

[26] Vivek Oommen, Khemraj Shukla, Somdatta Goswami, Remi Dingreville, and George Em Karniadakis. Learning two-phase microstructure evolution using neural operators and autoencoder architectures, 2022.

[27] M. Raissi, P. Perdikaris, and G.E. Karniadakis. Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations. *Journal of Computational Physics*, 378:686–707, 2019.

[28] Yoav Shechtman, Yonina C. Eldar, Oren Cohen, Henry Nicholas Chapman, Jianwei Miao, and Mordechai Segev. Phase retrieval with application to optical imaging: A contemporary overview. *IEEE Signal Processing Magazine*, 32(3):87–109, may 2015.

[29] Ayan Sinha, Justin Lee, Shuai Li, and George Barbastathis. Lensless computational imaging through deep learning. *Optica*, 4(9):1117, sep 2017.

[30] Xin Tao, Hongyun Gao, Xiaoyong Shen, Jue Wang, and Jiaya Jia. Scale-recurrent network for deep image deblurring. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*. IEEE, jun 2018.

[31] Kshitij Tayal, Chieh-Hsin Lai, Vipin Kumar, and Ju Sun. Inverse problems, deep learning, and symmetry breaking. *arXiv preprint arXiv:2003.09077*, 2020.

[32] Kshitij Tayal, Chieh-Hsin Lai, Raunak Manekar, Zhong Zhuang, Vipin Kumar, and Ju Sun. Unlocking inverse problems using deep learning: Breaking symmetries in phase retrieval. In *NeurIPS 2020 Workshop on Deep Learning and Inverse Problems*, 2020.

[33] Tobias Uelwer, Alexander Oberstraß, and Stefan Harmeling. Phase retrieval using conditional generative adversarial networks. *arXiv:1912.04981*, 2019.

[34] Chaoyang Wang, Chen-Hsuan Lin, and Simon Lucey. Deep nrsfm++: Towards 3d reconstruction in the wild. *arXiv:2001.10090*, 2020.

[35] Qiuliang Ye, Li-Wen Wang, and Daniel P. K. Lun. Sisprnet: end-to-end learning for single-shot phase retrieval. *Optics Express*, 30(18):31937, August 2022.

[36] Yi Zhang, Dasong Li, Xiaoyu Shi, Dailan He, Kangning Song, Xiaogang Wang, Honwei Qin, and Hongsheng Li. Kbnet: Kernel basis network for image restoration. *arXiv preprint arXiv:2303.02881*, 2023.

[37] Yuhe Zhang, Mike Andreas Noack, Patrik Vagovic, Kamel Fezzaa, Francisco Garcia-Moreno, Tobias Ritschel, and Pablo Villanueva-Perez. Phasegan: a deep-learning phase-retrieval approach for unpaired datasets. *Opt. Express*, 29(13):19593–19604, Jun 2021.

[38] Zhong Zhuang, David Yang, Felix Hofmann, David Barmherzig, and Ju Sun. Practical phase retrieval using double deep image priors, 2022.

## Author Biography

*Wenjie Zhang is a first-year Ph.D. student in Computer Science, University of Minnesota. His research focuses mainly on computer vision and AI for science.*

*Yuxiang Wan is a research associate at the University of Minnesota. His research focuses mainly on scientific machine learning.*

*Zhong Zhuang is an AI for Science Postdoctoral Researcher at UCLA. His research interest is in developing foundational AI tools to advance scientific discovery. Before, he received his PhD degree from Electrical and Computer Engineering of University of Minnesota in 2023.*

*Ju Sun is an assistant professor in Computer Science & Engineering, University of Minnesota. His research interest lies at the theoretical and computational foundations of AI, and AI application in science, engineering, and medicine. Before, he received his PhD degree in electrical engineering from Columbia University (2011–2016) and worked as a postdoctoral scholar at Stanford University (2016–2019).*