Optics EXPRESS

Randomized probe imaging through deep k-learning

ZHEN GUO,[1,5] ABRAHAM LEVITAN,[1,6] GEORGE BARBASTATHIS,[2,3,7] AND RICCARDO COMIN[4,8]

[1] *Department of Electrical Engineering and Computer Science, Massachusetts Institute of Technology, Cambridge, Massachusetts 02139, USA*
[2] *Department of Mechanical Engineering, Massachusetts Institute of Technology, Cambridge, Massachusetts 02139, USA*
[3] *Singapore-MIT Alliance for Research and Technology (SMART) Centre, 138602, Singapore*
[4] *Department of Physics, Massachusetts Institute of Technology, Cambridge, Massachusetts 02139, USA*
[5] *zguo0525@mit.edu*
[6] *alevitan@mit.edu*
[7] *gbarb@mit.edu*
[8] *rcomin@mit.edu*

Abstract: Randomized probe imaging (RPI) is a single-frame diffractive imaging method that uses highly randomized light to reconstruct the spatial features of a scattering object. The reconstruction process, known as phase retrieval, aims to recover a unique solution for the object without measuring the far-field phase information. Typically, reconstruction is done via time-consuming iterative algorithms. In this work, we propose a fast and efficient deep learning based method to reconstruct phase objects from RPI data. The method, which we call deep k-learning, applies the physical propagation operator to generate an approximation of the object as an input to the neural network. This way, the network no longer needs to parametrize the far-field diffraction physics, dramatically improving the results. Deep k-learning is shown to be computationally efficient and robust to Poisson noise. The advantages provided by our method may enable the analysis of far larger datasets in photon starved conditions, with important applications to the study of dynamic phenomena in physical science and biological engineering.

1. Introduction

Diffractive imaging is a set of lensless imaging techniques that are used for the reconstruction of non-periodic objects [1,2], such as integrated circuits [3], biological proteins [4], bone tissue [5], and more. In single-frame diffractive imaging, an incident beam illuminates an isolated unknown sample. Object features that are comparable in size to the illumination wavelength cause diffraction and the resulting intensity pattern is subsequently measured on a camera. The phase retrieval algorithm then recovers the lost phase information and reconstructs a discrete representation of the object [6–8]. For extended objects, multi-frame measurements can be made by scanning a localized illumination across a sample, a method known as ptychography [2,9]. The uniqueness of the reconstruction is guaranteed by illumination overlap between the multiple measurements, improving the reliability of the reconstruction [10,11].

The trade-off between single-frame and multi-frame diffractive imaging is that more measurements provide more stringent constraints on object reconstruction at the expense of longer time to acquire the data. Efforts have been made to implement ptychography with single-shot measurements, though they come at the cost of high hardware complexity and low information acquisition efficiency [11–13]. The search of a single-frame imaging method that retains the reliability and flexibility of multi-frame approach continues.

#445498
Journal © 2022

https://doi.org/10.1364/OE.445498
Received 18 Oct 2021; revised 7 Dec 2021; accepted 22 Dec 2021; published 10 Jan 2022

Randomized Probe Imaging (RPI) is a single-frame diffractive imaging method that uses randomized light, rather than a finite support constraint, to generate a unique solution to the phase retrieval problem [14]. The combination of randomized illumination and a band-limiting condition on the object provides enough information in the single-frame diffraction intensity to guarantee a unique solution up to a global additive phase factor. RPI is promising, for example, for time-dependent nanoscale X-ray imaging, since it does not introduce any optics behind the sample, or require any alternations to the sample. It has been shown that RPI can produce high-fidelity reconstructions using gradient descent based iterative algorithms [14]. However, conventional iterative algorithms are computationally expensive and typically do not exploit regularizing priors based on the statistical properties of scattering objects. As a result, it can be challenging to process large volumes of data with these algorithms, and they can have limited performance under low-light conditions.

Here, we propose a deep learning framework – deep k-learning – which is specifically designed to address the issues of computational load and low-light performance for far-field RPI reconstructions. Recently, many deep learning based algorithms have been proposed to solve phase retrieval problems, including reconstructions in tomography [15–21], ptychography [22–26], and holography [27–30]. Compared with conventional iterative approaches, deep learning algorithms can produce moderate quality reconstructions with low data redundancy, high computational efficiency, and low latency [15,22,27]. Deep learning methods have been particularly successful under noisy, low-light conditions [31,32].

In most previous works, a deep neural network (DNN), typically a convolutional neural network (CNN), is trained with examples of objects and their corresponding diffraction patterns. The goal is to minimize the loss between the generated objects output by the network and the ground truth. After training, the network will have learned the direct transformation from measurement to scattering object, implicitly incorporating the physics of light propagation. This is known as End-to-End training, and it relies on the idea that a learnable transfer function exists which maps the intensity measurements onto the object domain. In contrast, deep-k-learning uses an approximated version of the object – the output from one iteration of an iterative algorithm – as the input to the neural network. This follows a recent thread of research that leverages approximate physical operators to generate an input image, also referred to as the "Approximant", which is already in the object domain [31,33–36], generally finding vastly improved results even with simpler neural network architectures.

The use of an approximate physical operator has three main advantages over an End-to-End approach. First, the network no longer needs to learn the diffraction physics, which allows for leaner and simpler network architectures. Second, weight-sharing convolutional layers are not well suited to learning maps between the far-field and object domains. This is because the inductive bias in a convolutional layer assumes that relationships between input and output features are local and translationally equivariant. When mapping between far-field and object domains, these assumptions are emphatically not true. Third, pre-trained models and transfer learning can be applied when the network's inputs and outputs follow a natural image distribution, allowing for major speedups when training domain specific models.

The rest of this paper is structured as follows: First, we describe the principle of RPI and its experimental design in Section 2.. Next, we formulate the End-to-End phase retrieval method and discuss its pitfalls in Section 3.. We explain the proposed deep k-learning framework in Section 4.. Finally, we share numerical and experimental results in Section 5. and 6.. Concluding remarks are in Section 7..

2. Principle of RPI

The experimental geometry of RPI is outlined in Figure 1. A randomized zone plate first focuses coherent illumination at a wavelength λ to a focal spot. An order selecting aperture blocks unwanted higher order diffraction from the zone plate, producing an aperture filled with a band-limited random field at the sample plane. The randomized probe $P(x, y)$ then interacts with a thin sample described by a complex object function $O(x, y)$. In our work, we consider phase-only objects $O(x, y) = \exp(i\phi(x, y))$ for simplicity. The resulting exit wave $E(x, y) = P(x, y)O(x, y)$ propagates to the Fraunhofer plane where its intensity is measured by a charge-coupled device (CCD) camera. The noiseless intensity measurement $I_0(k_x, k_y)$ thus can be written as

$$I_0(k_x, k_y) = |\mathscr{F}\{P(x, y)O(x, y)\}|^2, \tag{1}$$

where $\mathscr{F}$ denotes the Fourier transform operator when the exiting wave propagates to the far-field. In practice, measurements are also subject to various sources of corrupting noise such as Poisson statistics and additive noise due to the CCD circuitry and detection process. We express the noisy measurement $I(k_x, k_y)$ in the far-field as

$$I(k_x, k_y) = \mathscr{P}\{I_0(k_x, k_y)\} + \mathscr{N}, \tag{2}$$

where $\mathscr{P}$ denotes Poisson sampling with parameter λ and $\mathscr{N}$ is the additive Gaussian noise.

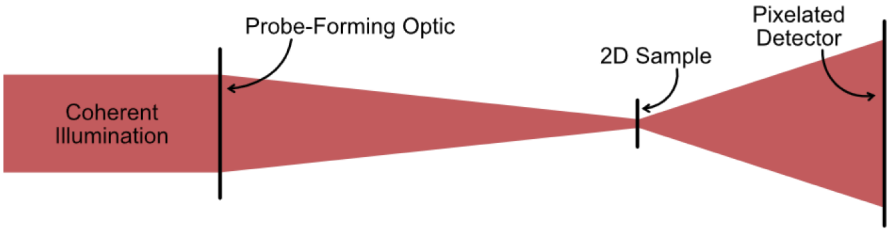


Fig. 1. A conceptual diagram of the layout used in an RPI experiment.

In the RPI reconstruction process, the measured single-frame diffraction intensity $I(k_x, k_y)$ and prior knowledge of the probe wavefield $P(x, y)$ are used to reconstruct a discrete representation of the object $O(x, y)$. Note that the presence of randomized illumination $P(x, y)$ breaks the spatial shift and conjugate inversion degeneracy of the classic two dimensional Coherent Diffractive Imaging (CDI) problem [37]. Rather than resorting to a finite support constraint as in traditional CDI, the reconstruction process in RPI uses a band-limiting constraint on the object to restrict the number of free parameters and achieve sufficient data redundancy.

Importantly, this reconstruction process is only well-posed when the diffraction pattern contains sufficiently more measurements than the number of independent degrees of freedom in the object. Without additional information about the object, this leads to an expectation that a stable reconstructions can be achieved when the highest frequency k_p at which the probe has nonzero power remains larger than the frequency k_o to which the object is band-limited. Based on this analysis, it is useful to define the resolution ratio $R = \frac{k_o}{k_p}$ [14]. As the resolution ratio decreases, the sampling redundancy increases, producing more stable (but lower-resolution) reconstruction. In this work, we consider the role of machine learning approaches at various values of R, ranging from low values (~ 0.25) where the reconstruction is extremely tightly constrained to high values (~ 2) where, without additional information, the problem is almost certainly ill-posed.

3. End-to-end phase retrieval

Convolutional neural networks are an indispensable tool for many modern computer vision applications, such as image classification [38], objection detection [39], and neural style transfer [40]. Many recent works have also shown that convolutional networks perform well in solving phase retrieval problems [31–33,41–43].

The most basic way to apply a convolutional neural network to the phase retrieval problem, which remains the basic standard, is known as the End-to-End approach. In this design, one trains a network using the raw diffraction patterns as an input, producing as output an estimate of the retrieved object. In our case, this output would be an estimate of the phase of a thin, phase-only object. This works well, or at least passably, for many variants of diffractive imaging based on Fresnel propagation [31–33,44].

Considering the design of a standard convolutional network, outlined in Fig. 2, can help us understand why these networks are a natural fit to Fresnel-based phase retrieval problems.

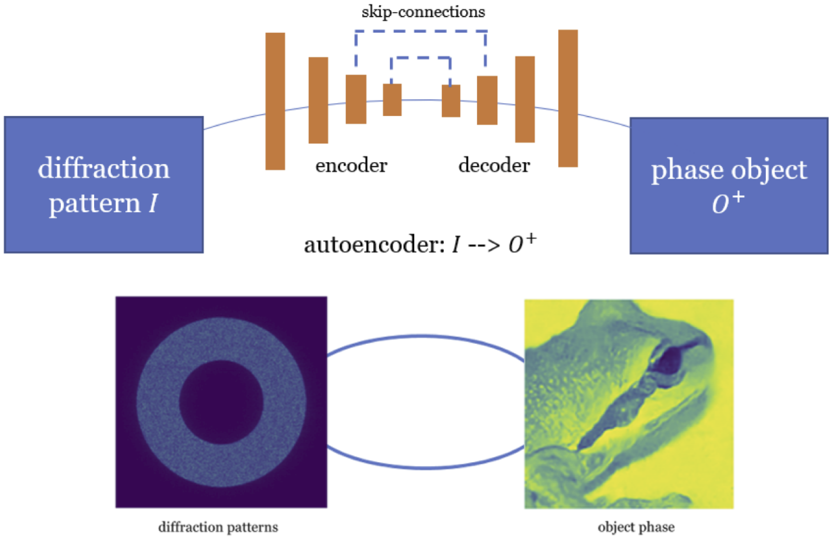


Fig. 2. Architecture of conventional encoder-decoder

Typical networks are divided into an encoding arm and a decoding arm. The encoding arm learns to predict a representation of the scattering object in a low-dimensional latent space based on an input diffraction pattern. The decoding arm learns to map from the embedding manifold back to the discrete representation of the scattering object - the desired final result. Often, skip connections are used to bypass the feature maps from the encoder arm to the the corresponding layers in the decoders arm. This allows local information to be transferred directly from the input to output domains, which helps preserve high frequency structures in the reconstruction [44].

Convolutional neural networks often work well when the relationship between the input and output domains is fundamentally local. This is because weight-sharing convolutional layers preserve translation equivariance [45], such that a shifted input to a layer produces a shifted output. Because Fresnel diffraction patterns do preserve the location of features in the scattering object, the map that must be learned to perform phase retrieval naturally shares the same translation equivariance as the convolutional layers.

However, convolutional networks are not ideal when the input diffraction patterns are in the far-field regime (as is the case for RPI), for two major reasons. First, the real-space to Fourier space mapping is global. In a far-field phase retrieval such as RPI, every pixel in the diffraction pattern includes a contribution from every pixel in the real-space object domain. Second, the real-space to Fourier space mapping does not respect translation equivariance. A shifted input diffraction pattern should be mapped to a version of the output object with linear phase ramp, rather than a translated version of the corresponding output object. This is formalized with the following inequality:

$$g_0(x+\delta x, y+\delta y) \neq |\mathscr{F}\{P(x,y)O(x+\delta x, y+\delta u)\}|^2. \tag{3}$$

Although the presence of boundaries and downsampling layers means that typical convolutional network architectures are not strictly translation-equivariant in a formal sense, their bias toward retrieving local, translation-equivariant maps makes the direct application of convolutional neural network to end-to-end far-field phase retrieval problematic.

4. Our solution: deep k-learning

4.1. Physical operator and autoencoder

The workaround we used for applying convolutional neural networks to phase retrieval in the far-field (in this case, RPI) is to apply an approximate map from the diffraction pattern domain to the object domain *before* using the neural network for the final reconstruction. This framework is depicted in Fig. 3. Although the approximate map cannot produce an accurate reconstruction on it's own, it creates inputs for training and inference which already live in the same image space as the final reconstructed objects. We call this approach deep-k-learning, because it is designed to compensate for the issues created by having input data which is organized in k-space.

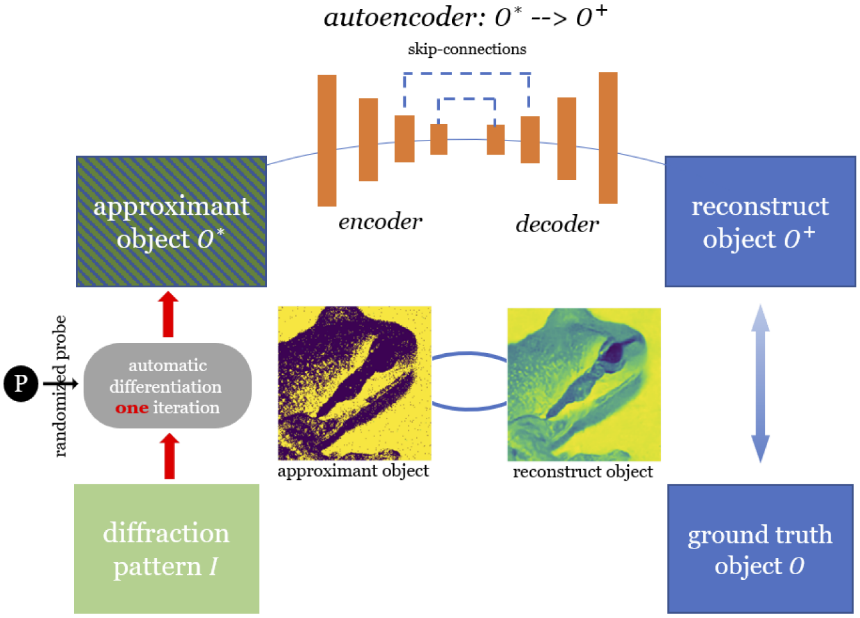


Fig. 3. Our deep k-learning framework

The choice of approximate mapping is clearly of crucial importance. In this work, we use a single iteration of a gradient-descent based iterative algorithm solving the following optimization problem for a diffraction pattern I_i:

$$\mathbf{O}_i^+ = \underset{\mathbf{O}_i'}{\operatorname{argmin}}\, \mathscr{L}\left\{I_i, |\mathscr{F}\{P \times O_i'\}|^2\right\} \tag{4}$$

Here, O_i' is a low-fidelity estimate of the band-limited object and P is the known probe state. The probe P is either known *a priori* (as for simulation), or retrieved via ptychography measurement (as for experiment). The output of a single step of the iterative algorithm when initialized with a uniform object is called the Approximant and is denoted by O_i^*. When more steps of the optimization is taken (with lower learning rate), we regard the output as iterative reconstruction. Adam optimizer is chosen to generate Approximant and iterative reconstruction for fair comparisons. Approximant is then fed into a CNN based autoencoder $G_{\mathbf{w}}$ with parameters $\mathbf{w}$. The training process learns a map from Approximants to ground-truth objects, formally written as:

$$\hat{\mathbf{w}} = \underset{\mathbf{W}}{\operatorname{argmin}} \sum_i \mathscr{L}\left\{O_i, G_{\mathbf{w}}(O_i^*)\right\} \tag{5}$$

where the optimal weights after gradient descent are $\hat{\mathbf{w}}$, and O_i is the ground truth object.

The network we used is an autoencoder architecture [46], where the encoder arm shares similar architecture as EfficientNetB7 [47] to enable efficient feature extraction. The feature pooling is built on inverted residual blocks (or MBConv), where the input and output of the residual block are thin bottleneck layers as opposed to traditional residual blocks to achieve efficient feature extraction [47,48]. In each inverted residual block, convolutional layers are being deployed to extract local features, and squeeze and excitation (SE) blocks are being used to extract global features [49]. Note that the convolutional layers in our implementation are combined with depth-wise and point-wise convolutions to reduce the computation cost [50]. Residual connections within each block are employed to avoid the problem of vanishing gradients [51], batch normalization is adopted to stabilize the learning process [52], and dropout layers are used to prevent over-fitting. Down-sampling in the encoder arm is achieved via average pooling block by block, with a pooling size of (32, 32) in total. Therefore, the final embedded output from the encoder has a dimension of (H/32, W/32, C), where H and W are the height and width of the input object, and C is the channel size in the last inverted residual block. In our implementation, C is 2560.

The decoder arm is comprised of five residual up-sampling blocks with up-sampling. The up-sampling is achieved by transposed convolution. Each up-sampling transposed convolution layer is followed by two convolution layers with same filter and kernel sizes. The scaling factor of all the up-sampling blocks is (32, 32) in total, producing an output with the shape (H, W, 1). Skip connections are used between encoder and decoder arms to preserve high-frequency information [44]. The detailed network architecture can be found in our github page.

4.2. Supervised representation and adversarial loss

Three main choices exist for the loss function $\mathscr{L}$ needed to train the deep k-learning framework: supervised loss, representation loss, and adversarial loss. In this work, we constructed a loss function consisting of a mix of all three types. When the network is trained with a mix of all three types, we call it generative deep k-learning. When the network is trained with supervised loss only, we call it non-generative. The supervised loss, which directly compares predicted ground truth objects, is the main component. In our implementation, supervised loss was implemented as the negative Pearson correlation coefficient (NPCC) between the reconstructed objects and the ground truth, defined as

$$\text{NPCC} = -r_{X,Y} = -\frac{\text{cov}(X, Y)}{\sigma_X \sigma_Y}, \tag{6}$$

where cov is the covariance and σ_X, σ_Y are the standard deviations of X and Y, respectively. Previous works have shown that NPCC is more effective in recovering fine features than pixel-wise loss functions [32,33,44,53]. In the context of our network, NPCC is written as:

$$\mathscr{L}_{\text{npcc}}(G_{\mathbf{w}}) = \mathbb{E}_{O,O^*}[-r_{O,G_{\mathbf{w}}(O^*)}] \tag{7}$$

To define the representation loss, we use an ImageNet pretrained EfficientNetB0. This representation loss is a perceptually-motivated loss which measures the mean absolute error between the latent space representation of the reconstructed object $H(O^+)$ and the embedding of the ground-truth object $H(O)$. Here, H refers to the pretrained EfficientNetB0 encoder. It may improve the reconstruction quality without changing the network architecture [54], helping the generative model to synthesize features closer to the ground truth distribution. In our implementation, we choose L1 or mean absolute error to measure the distance between the two distributions:

$$\mathscr{L}_{\text{mae}}(G_{\mathbf{w}}) = \mathbb{E}_{O,O^*}[\|H(O) - H(G_{\mathbf{w}}(O^*))\|_1] \tag{8}$$

The adversarial loss is computed with a CNN based discriminator. Our implementation of adversarial loss is inspired by conditional generative adversarial networks (cGANs), a particular training strategy that uses a discriminator to compete with the autoencoder/generator [55–57]. The objective of cGAN for our RPI problem can be written as follows:

$$\mathscr{L}_{\text{adv}}(G_{\mathbf{w}}, D'_{\mathbf{w}}) = \left(\mathbb{E}_{\mathbf{o}\sim\mathbf{p}_{\mathbf{o}}(\mathbf{o})}\left[\log D_{\mathbf{w}'}(\mathbf{o})\right] + \mathbb{E}_{\mathbf{o}^*\sim\mathbf{p}_{\mathbf{o}^*}(\mathbf{o}^*)}\left[\log(1 - D_{\mathbf{w}'}(G_{\mathbf{w}}(\mathbf{o}^*)))\right]\right) \tag{9}$$

Now, our autoencoder becomes a generative model G that tries to generate objects with the highest possible value of $D(G(\mathbf{o}^*)))$ to fool the discriminator D, as shown in the second term of Eq. (9). Simultaneously, the discriminator D tries to maximize its ability to recognize ground truth objects as real and generated objects as fake, i.e. $\hat{G_{\mathbf{w}}} = \arg\min_{G_{\mathbf{w}}} \max_{D_{\mathbf{w}'}} \mathscr{L}_{\text{adv}}(G_{\mathbf{w}}, D'_{\mathbf{w}})$. This component of the loss updates the weights in the discriminator. During training, the generator and discriminator are simultaneously updated based on their respective losses. The adversarial loss generally is thought to help the autoencoder/generator learn the transformation of the noise within the object Approximant to plausible features in the final reconstructed object, given the prior of ground truth distribution O.

Finally, the total loss for the generator of our deep k-learning framework is defined as:

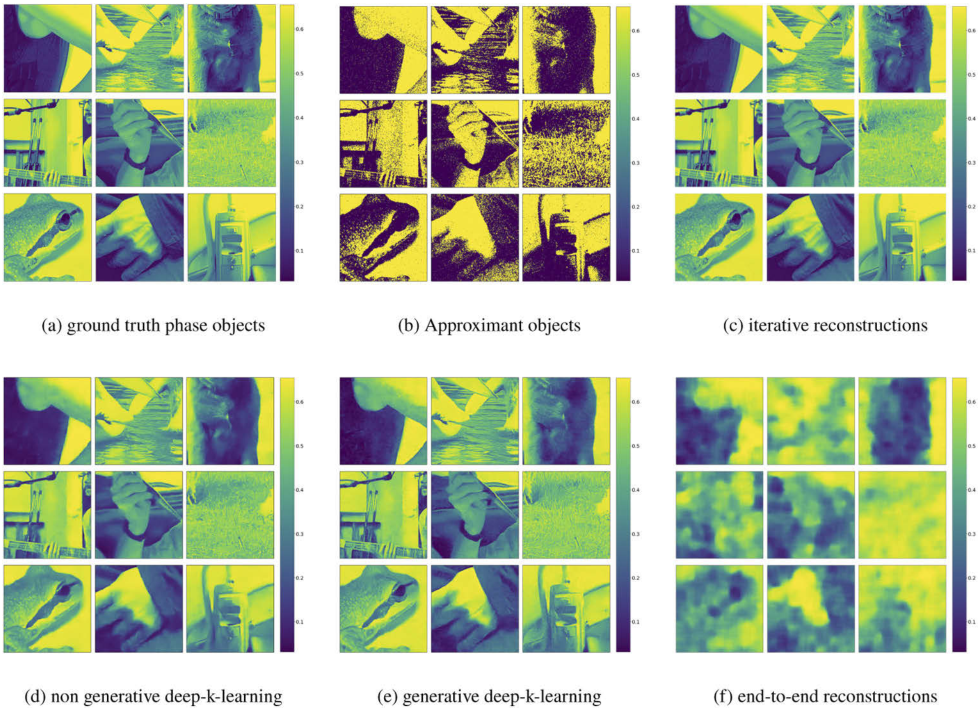
$$\mathscr{L}_{\text{total}} = \mathscr{L}_{\text{npcc}}(G_{\mathbf{w}}) + \alpha \times \mathscr{L}_{\text{mae}}(G_{\mathbf{w}}) + \beta \times \arg\min_{G_{\mathbf{w}}}\max_{D_{\mathbf{w}'}} \mathscr{L}_{\text{adv}}(G_{\mathbf{w}}, D'_{\mathbf{w}}) \tag{10}$$

Here, α and β are hyper-parameters that determine the relative weights between the three types of learning loss. For the non generative framework, α and β are set to zero.

5. Numerical results

We conducted a set of numerical simulations to demonstrate the effectiveness of the deep k-learning method on the RPI phase retrieval problem. We focused on the role of the resolution ratio R and the noise level. High resolution ratios R and low signal regimes are particularly interesting to study because these conditions are the most challenging scenarios for iterative algorithms, and therefore are most likely to benefit from the added information about the object distribution that deep-k-learning can introduce.

In our first experiment, we studied the performance of the various proposed methods under ideal illumination conditions. We simulated an RPI experiment using 256×256 pixel objects defined with uniform amplitudes and phases drawn from randomly cropped ImageNet images, scaled to a range of up to 1 radian. 4,000 training examples and 100 testing examples were simulated, at $R = 0.5$ with 10^4 photons per pixel in the 256×256 pixel object. Fig. 4 shows a visual comparison between the phase images reconstructed with each method. Fig. 4(a) shows a set of ground truth objects selected from the testing dataset. In Fig. 4(b), the corresponding Approximants are shown. We can see that the approximate map successfully retrieves the general structures of the object, albeit at an incorrect overall scale. Additionally, noise and artifacts are readily apparent and, when just considered as images, the Approximants are of low quality. In contrast, Fig. 4(c) shows the converged results from iterative reconstructions after 100 iterations. Visually, they look identical to the ground truth phase objects, as expected based on the ideal imaging conditions [14].



(a) ground truth phase objects

(b) Approximant objects

(c) iterative reconstructions

(d) non generative deep-k-learning

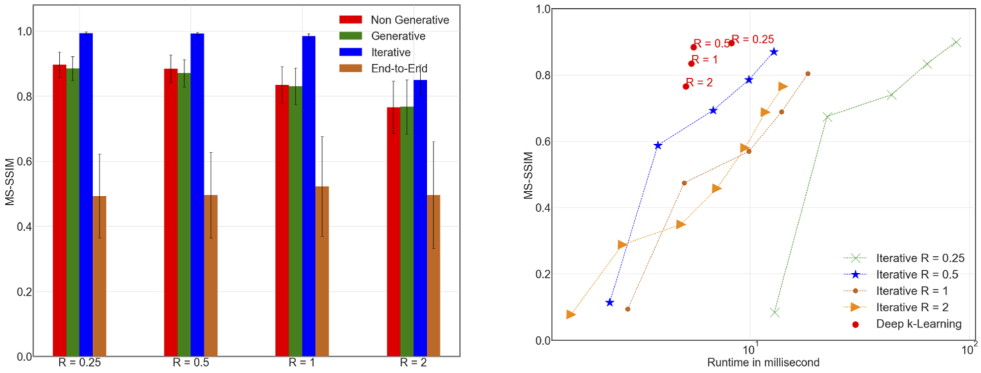
(e) generative deep-k-learning

(f) end-to-end reconstructions

Fig. 4. Visual comparison for the phase-only object reconstruction at $R = 0.5$ with 10^4 photons per pixel. The color bar is set to the range of the ground truth images. (a) contains the ground truth phase-only objects, (b) contains the input Approximant with one iteration, (c) contains the iterative reconstructions, (d) contains the non generative deep-k-learning reconstructions, (e) contains the generative reconstructions, (f) contains the end-to-end reconstructions.

Moving to the neural network outputs, Fig. 4(d) shows the non generative deep k-learning reconstructions. Drastic improvements are obvious when compared with the input Approximants. Reconstruction are now smoother and contain fine details that were washed out by noise in the input Approximants. However, although the results have high visual quality, there are noticeably missing fine features when compared with ground truth and iterative reconstructions. Fig. 4(e) has the generative deep-k-learning reconstructions, although visually the difference between non-generative and generative reconstructions under these illumination conditions is not obvious. Finally, Fig. 4(f) contains the output of the end-to-end network reconstructions. These results only contain low frequency information about the phase objects. This is not entirely unexpected, given the previous arguments about the mismatch between convolutional neural networks and mappings between k-space and real-space.

After confirming that deep-k-learning is capable of producing moderate quality images under ideal conditions, we studied how its performance depends on the relationship between the highest frequencies in the object and those in the probe. In Fig. 5(a) we show a quantitative comparison of reconstruction quality at values of R ranging from 0.25 to 2 at 10^4. The x-axis represents the resolution ratio R, and the y-axis reports the MS-SSIM (Multi-scale Structural Similarity) metric for the reconstruction quality. The reported value is the mean MS-SSIM result over the test reconstruction set, and the error bars show the standard deviation within the test dataset. Recall that larger values of R describe more challenging conditions where the features in the object are smaller when compared to the speckle size in the probe.



(a) simulation results for R from 0.25, 0.5, 1, to 2

(b) runtime comparison at different R values

Fig. 5. Quantitative comparison between different training frameworks at different R

A total of four reconstruction methods are reported in the figure: non generative deep k-learning, generative deep k-learning, an iterative algorithm (100 iterations), and the End-to-End training method. Details about the data processing and network training can be found in the Appendix. Iterative reconstructions have the best performance over the full range of R. At $R \leq 1$, the MS-SSIM evaluations of iterative reconstructions approach 1, as expected [14]. As R increases beyond 1, the iterative reconstructions start to degrade. These observations agree with previous work, as larger values of R lower the data redundancy in the diffraction patterns. Both variants of deep k-learning methods outperformed end-to-end networks, although the results still underperformed the iterative reconstructions. Reconstruction quality also degraded with R across methods, as expected. However, the End-to-End reconstructions' quality plateaus at a lower value of R, regardless of the oversampling ratio, in agreement with the visual observations.

Although the deep-k-learning reconstructions do not produce the same level of precision as the iterative results, they have a major advantage in runtime. Fig. 5(b) compares the per-pattern runtime of iterative algorithms and deep k-learning method across R. The intermediate results from the iterative reconstructions are shown at 1, 5, and 10 iterations, and every 10 iterations thereafter until they surpass the comparable deep-k-learning result. The strong dependence of per-iteration runtime on R arises because smaller values of R require more highly textured probes, which are stored in larger arrays. These results reveal that the deep k-learning results have comparable quality with iterative reconstructions at around 40 to 50 iterations. However, the computational speedup provided by deep k-learning ranges from 3x to 10x, depending on the value of R.

Finally, we investigated the performance of deep-k-learning for RPI under noisy conditions, where knowledge of the object's prior statistics is the most valuable. Fig. 6 shows a visual comparison for the phase images reconstructed at $R = 0.5$, with illumination levels ranging from 10^3 to 1 photon per object pixel. As the photon incidence rate decreases, reconstruction quality inevitably decreases as well. As expected, the iterative reconstruction quality is strongly dependent on the photon shot noise level, with the signal quickly fading under growing background. However, the deep-k-learning results generally retain their visual quality even at photon rates low enough to cause the iterative method significant degradation. In the single photon case, the iterative reconstruction becomes nearly unrecognizable, while both deep k-learning methods produce reconstructions that, while visibly degraded, maintain many of the general features of the object.

Fig. 7(a) shows the quantitative comparison from the same sweep over low photon imaging conditions, following the same format as Figure 5. These quantitative results confirm the analysis

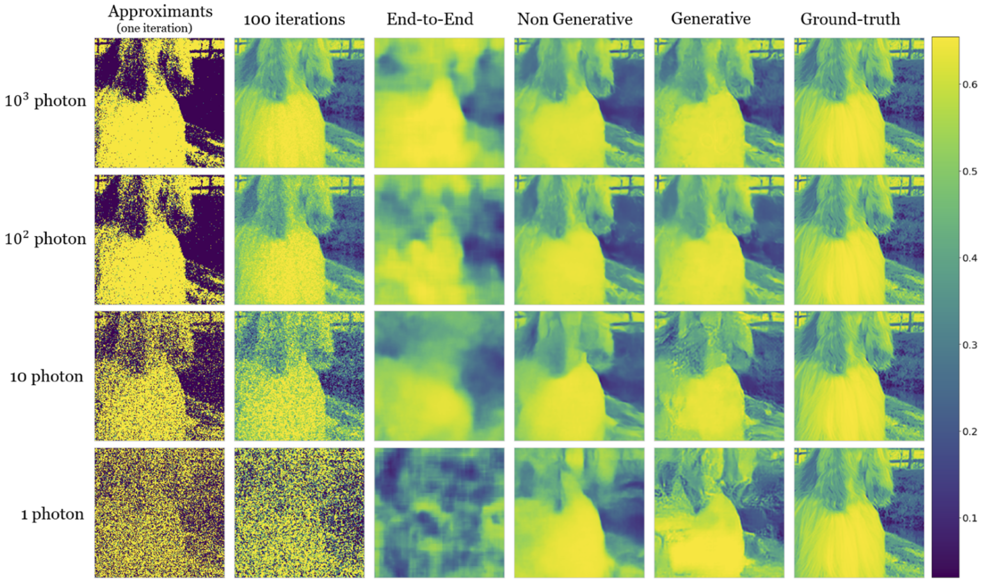
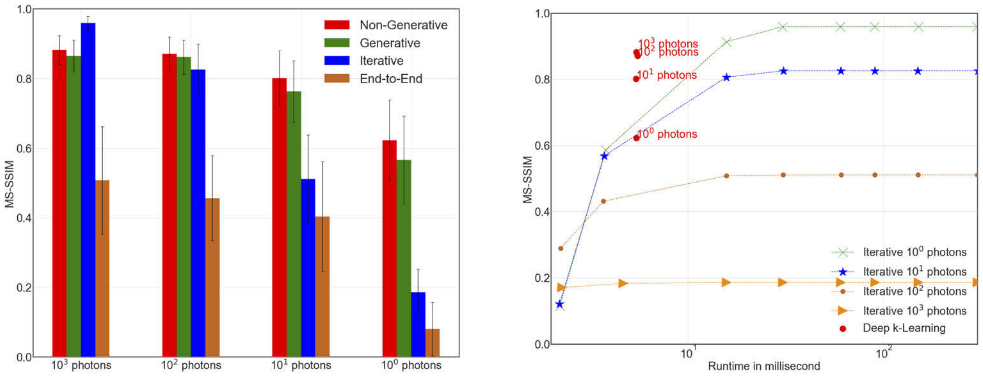


Fig. 6. Visual comparison for the phase-only object reconstruction for R=0.5 at low photon imaging conditions. The colorbar is set to the range of the ground truth images.

from our visual inspection of Fig. 6. Both deep k-learning methods are significantly more robust to Poisson noise than the iterative methods, producing reconstructions with superior quality starting at 10^2 photons. As the photon number decreases further, the gap between deep k-learning and iterative reconstruction quality grows. This shows the effectiveness of the strong object prior embedded in the deep k-learning methods through the training process.



(a) results for R = 0.5 at low photon conditions

(b) runtime comparison at low photon conditions

Fig. 7. Quantitative comparison between different training frameworks at low photon imaging conditions

Finally, in Fig. 7(b) we consider the runtime speedup available under high noise conditions, comparing the iterative algorithm with the best variant of deep-k-learning method at each imaging condition. Due to the feed-forward nature of deep learning, deep k-learning takes under 10 milliseconds to produce each result, while the iterative algorithm require around 100 milliseconds to converge, suggesting that the 10x speedup under ideal illumination is preserved, or even improved upon, under adverse, noisy conditions.

Overall, our simulation results show that deep k-learning is both faster and more robust to Poisson corruption than the iterative algorithm. Particularly when photon levels reach 10^2 photons per object pixel or lower, deep k-learning outperforms iterative algorithms in terms of reconstruction quality with much faster computational speed.

6. Experimental results

To demonstrate that the deep k-learning approach can successfully be translated from simulation to experiment, we performed phase retrieval with deep k-learning on a large dataset of RPI diffraction patterns collected from an optical table-top apparatus. To draw test images from a well understood distribution, we used a Spatial Light Modulator (SLM) to display 256x256 phase-only images randomly drawn and cropped from the ImageNet dataset. The experimental design is diagrammed in Figure 8.

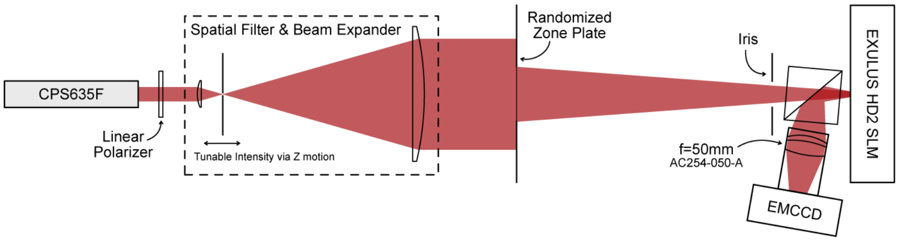


Fig. 8. A diagram of the experimental design for our tabletop demonstration.

Polarized light was generated by passing a 635 nm laser diode source (Thorlabs CPS635F) through a film polarizer aligned to the optic axis of the SLM. This light was then spatially filtered by a 5 μm pinhole at the focus of a beam expander to enforce spatial coherence across the beam diameter. A randomized pattern was then imprinted on the wavefield using a randomized zone plate with a 2 cm diameter and a 50 cm focal length, producing a focal spot with an overall diameter of 2 mm. An adjustable iris acted as an order selecting aperture for this diffractive optic.

The focus of the randomized zone plate was aligned to the plane of a reflective SLM (Thorlabs EXULUS HD2) at normal incidence. The phase-only SLM consisted of pixels arranged with an 8 μm pitch, each of which imprints a variable phase delay between 0 and 2π on the light field. The reflection was then separated with a non-polarizing 50/50 beamsplitter cube placed approximately 5 degrees from normal to prevent higher order reflections from overlapping with the primary beam on the detector. The Fourier plane was finally imaged on a EM-CCD camera (QImaging Rolera EM-C2) with 8 μm pixels, placed at the focus of an achromatic doublet with a 50 mm focal length (Thorlabs AC254-050-A). A 992×992 pixel region was cropped from the detector, such that the real-space grid corresponding to the measured slice of reciprocal space consists of 8 μm pixels, aligned with the pitch of the SLM.

We collected four datasets under different imaging conditions, targeting photon fluxes of 1, 10, 100, and 1000 photons per pixel in the 256x256 object. The CCD is calibrated to allow conversion between analog-digital units (ADUs) and photon counts. A detailed summary of the experimental measurements, including more information on the noise properties of the detector, can be found in the Appendix. For each imaging condition, we initially collected a ptychography dataset on a standard test image (cameraman) to calibrate our knowledge of the probe state. Once calibrated, we collected a set of $4,000$ training and 100 test diffraction patterns from the cropped ImageNet objects. In each case, the images were first converted to 8-bit greyscale images, and finally displayed on the SLM such that the full 8-bit range corresponded to a sweep from 0 to 2π radians. Details about experimental measurements can be found in the Appendix.

In Fig. 9 we show the visual comparisons of the test set between different reconstruction algorithms under low photon conditions. These reconstructions are produced with the same set of algorithms we studied in section 5.. The most visible difference between simulation and experiment is that, while in our simulations we assumed the randomized probe focal spot covers the entire field-of-view, in the experiments the objects are illuminated by a finite circular probe. Thus, the edges of the object window are not illuminated by the probe and thus do not contain any object features.

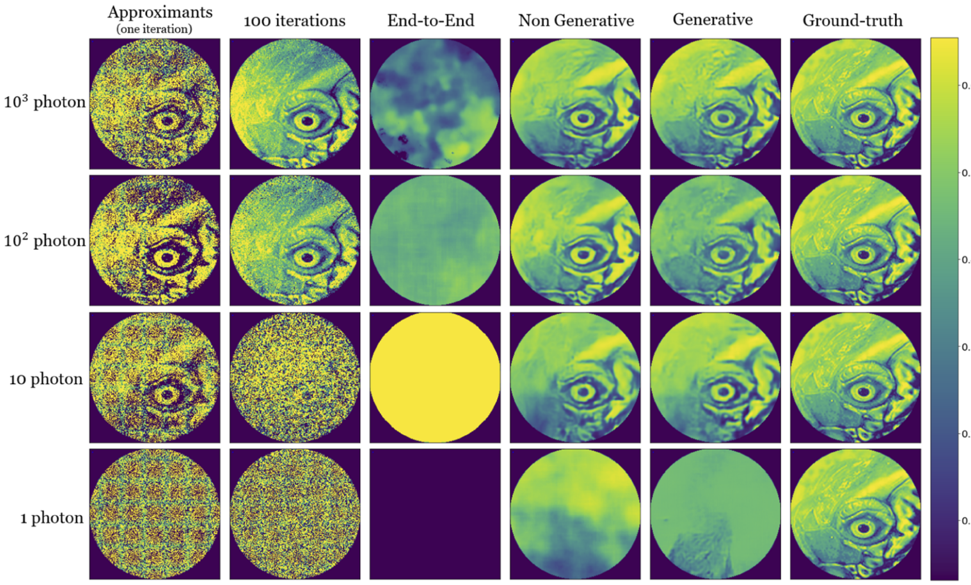


Fig. 9. Experimental reconstruction comparison between different methods under low photon conditions. The colorbar is set to the range of the ground truth images.

Near 10^3 photons/pixel, iterative reconstructions show good results in the central region, getting more and more noisy toward the weaker edge of the probe. This is due to the spatial variation of the illumination intensity profile. Both the non generative and generative deep k-learning produce visually high quality reconstructions over the entire field of view of the probe, although the networks minor artifacts are indeed introduced, especially at the lower end of the photon incidence rates.

As we decrease the photon budget down through 100 to roughly 10 photons per object pixel, the quality of the generative reconstructions slowly decreases while the noise rapidly takes over and dominates the iterative results. The End-to-End model begins to diverge at 10 photons per object pixel. As we lower the signal rate further, to 1 photon per object pixel, the reconstructions from all methods fail. To account for the disparity, it is important to recognize that due to the presence of readout noise and other non-Poisson sources of noise, the signal to noise ratio of these images is far lower than that of our simulated dataset at 1 photon per pixel.

Fig. 10 shows a numerical comparison that confirms our observations. Note that we only include the illuminated region (the region in the center images of Fig. 9) when computing the MS-SSIM values for each method. For iterative reconstruction, we also shift the output pixel to compensate for a slight misalignment in our optical system. Compared with simulation results, deep k-learning methods maintain a moderate quality level in the range of 0.8 under the second-to-lowest illumination conditions, while the quality of the iterative results deteriorates as the photon number decreases. End-to-end MS-SSIM drops to near zero starting at 10 photon per object pixel, as the pixel values of the outputs are outside the range of the ground truth. These results suggest that the iterative algorithm is much more prone to noise degradation than the deep-k-learning approach. Thus, deep-k-learning emerges as a valuable alternative particularly under noisy conditions. This is because under such noisy experimental conditions the deep k-learning algorithm is far more effective at incorporating strong object priors to regularize the reconstructions, mitigating noise effects.

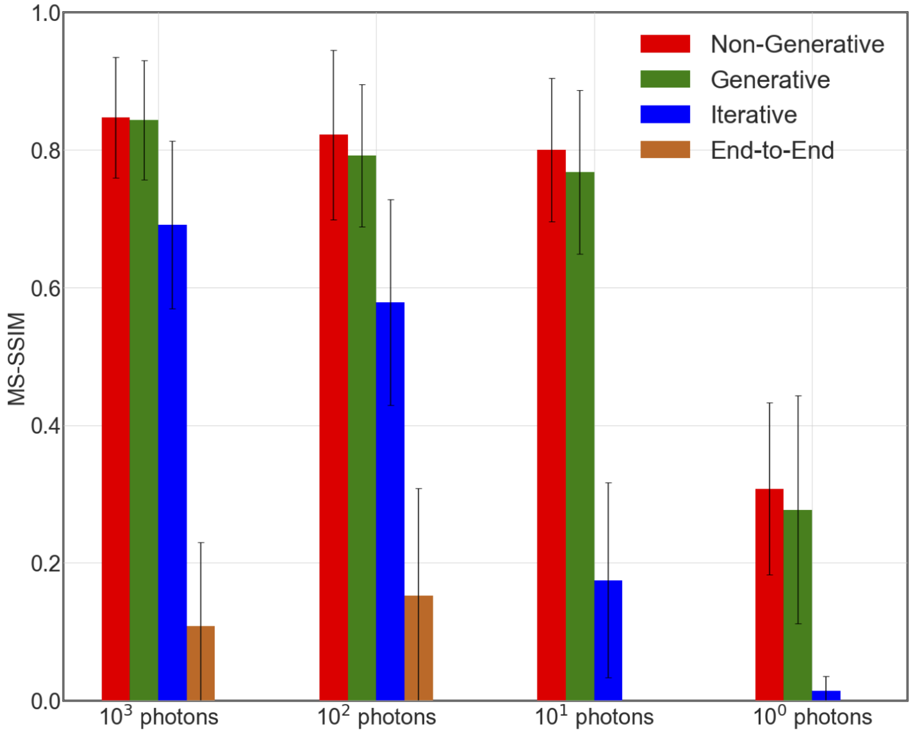


Fig. 10. MS-SSIM comparison between deep-k-learning and iterative algorithm on different Poisson noise corrupted imaging conditions

7. Conclusion

We have demonstrated a reliable machine learning-based computational imaging method, deep k-learning, that works well for Randomized Probe Imaging with phase-only objects. Our deep k-learning framework outperforms End-to-End machine learning design in all simulations and experiments, but underperforms traditional iterative reconstruction when illumination is ample and noise is low. On the other hand, deep k-learning is more robust under Poisson statistics-dominated low photon incidence conditions, with reconstructions degrading gracefully even when the strong noise drives the iterative method to complete failure. In all cases, the fully trained deep k-learning method is, as expected, more computationally efficient. The improved resilience to noise makes deep k-learning endowed RPI attractive in situations where illumination power is limited or the samples are sensitive to excessive radiation exposure. Thus, we expect it to find application to many dynamic phenomena in the physical, material and biological sciences and engineering.

Optics EXPRESS

Appendices

A. Discussion of end-to-end RPI phase retrieval

Let $\Gamma \doteq \{O_i, I_i\}$ be the paired training dataset, P the known randomized probe, and let $G_{\mathbf{w}}$ be a set of parameters for the deep neural network that can be trained with. The parameters $\mathbf{w}$ are also commonly referred to as the “connection weights” or simply “weights” in traditional neural network architectures. Then the end-to-end phase retrieval problem becomes of finding the optimal weights $\hat{\mathbf{w}}$ such that given any intensity pattern within the dataset distribution Γ, along with the known probe P, the network can produce a generated object O_i^+ that is an equivalent class to the O_i, or formally

$$\hat{\mathbf{w}} = \underset{\mathbf{W}}{\operatorname{argmin}} \sum_i \mathscr{L}\left\{O_i, G_{\mathbf{w}}(I_i, P)\right\} \tag{11}$$

where $\mathscr{L}$ is the loss function that measures the discrepancy between the generated object O_i^+ and ground truth O_i. For RPI, the equivalence class is defined to be the set of all objects which may be derived from the $E_i(x, y)$ by changing in the global phase. The commonly encountered spatial shift and time-reversal symmetries in diffractive imaging systems are not symmetries of the RPI system, due to the presence of the randomized probe [37]. For global phase degeneracy, any complex rotation of O_i^+ in degree ϕ would result in identical far-field intensity pattern I_i, and therefore, the output O_i^+ of the formulation above also needs to take those degenerate solutions into account. An alternative formulation inspired by [58] would be

$$\hat{\mathbf{w}} = \underset{\mathbf{W}}{\operatorname{argmin}} \sum_i \mathscr{L}\left\{I_i, |\mathscr{F}\{G_{\mathbf{w}}(I_i, P)\}|^2\right\} \tag{12}$$

Here, the problem of phase retrieval becomes equivalent to that of minimizing the loss in the far-field domain. Thus, the inverse problem is indirectly solved, with the optimization forcing the network to generate the amplitude and phase of the exiting wave E, rather than the object O. Since the applied constraint is in the far-field domain, the formulation would preserve the global phase degeneracy in its solution. However, in this case, the network would learn priors based on the training distribution E, and it would be challenging to continuously sample this distribution and capture its statistics for testing as E is the product of the object and randomized probe. It is easier to guarantee that the training distributions O follow the same statistics of the testing distribution O, as long as training and testing dataset are both constrained to natural images with geometric features.

B. Network training procedure

Our proposed deep k-learning networks were implemented in Python 3.7.9 using TensorFlow 2.3.1, and trained with NVidia V100 tensor core graphics processing unit on MIT Supercloud [59]. The object training set was from 4,000 natural images in ImageNet, where phases were set to be the images and amplitudes were set to be one. The $(256 \times 256 \times 3)$ ImageNet images were converted to gray-scale from the original RGB format. Therefore, the total training object dataset is a complex matrix with dimension of $(4000, 256, 256, 1)$. The randomized probe P was generated based on the method in [60] given the sampling ratio R. The far-field diffraction patterns were then numerically simulated based on the optical setup in Figure 1. The approximate objects were subsequently generated via automatic differentiation with one iterations with steepest gradient descent for each diffraction pattern, and the loss function $\mathscr{L}$ here is the mean square error (MSE) on the amplitude. The iterative results are from 100 iterations with 0.5 learning rate. After numerical simulation, we normalized all of the paired training data in the $\Gamma \doteq \{O_i, O_i^*, I_i\}$ dataset between $[0, 255]$. This would improve network training stability later on. For training, Adam optimizer [61] was used with parameters $\beta_1 = 0.9$ and $\beta_2 = 0.999$, the initial learning rate was 2×10^{-4}. The validation split was 0.1 to provide an unbiased evaluation of a model fit on the training dataset.. The learning rate would be reduced by half when the validation loss stops improving for 10 epochs. We set the maximum epoch to be 200, and the training would stop early when either the validation loss plateaus for 20 epochs, or the minimal learning rate 10^{-8} is reached. This early-stop technique would prevent the model from over-fitting. We keep the same training parameters for all the networks, the variations of different training were i), the training strategy (either end-to-end or deep-k-learning), ii), generative or non generative, iii), network weights initialization (with random initial weights or ImageNet pre-trained weights in the encoder arm), and iv) hyper-parameter β for generative deep k-learning (α is fixed as 1/8 to reduce the complexity of hyper-parameter grid search) in the total loss function of the autoencoder/generator. When the network is initialized with pre-trained weights in the encoder, the 200 epochs would be completed in two steps: in the first step, we only train the decoder of the network while the encoder would be frozen with pre-trained weights; in the second step, we unfreeze and train the entire network. This can accelerate the training for models with pre-trained weights.

For end-to-end training, we divided each far-field diffraction pattern into multiple patches with dimension of $(256, 256, C_R)$, where C_R is the number of channels that depends on the dimension of the diffraction pattern with the given oversampling ratio R. The inputs to the end-to-end network are the multi-patch representation of diffraction pattern concatenated with the randomized probe that is also in multi-patch representation. This way, we can keep the number of parameters in the end-to-end network to be roughly the same as the deep-k-learning framework (around 76.5 million in total parameters in both cases, not counting the discriminator network and pre-trained EfficientNetB0), and makes fair performance comparison later on. Also, in the end-to-end neural network, we removed the skip connections between encoder and decoder because of the large domain transfer in-between.

C. Experimental procedure for measurements

Table 1. Summary of the four sets of experimental measurements

Target photon/pixel	Measured photon/pixel	Averaged SNR
1000	996	6.09
100	127	2.07
10	11.9	0.375
1	1.77	0.0525

Data were collected under four different experimental imaging conditions individually (Table 1). For the 10, 100, and 1000 photon per object pixel collections, the total image intensity was modulated by extending the exposure time, using an EM gain of 54 (corresponding to EM level of 3800 in the camera software) and an offset level of 0. To implement the necessary range of attenuations, we chose a pinhole size of 5μm, significantly smaller than the waist of the beam emerging from the collimating objective; and moved the pinhole away from the center to further lower photon fluxes. The offset level for this measurement was set to 500, due to the extremely weak signal level. We also collected 10 background images per signal level under a reproduction of the imaging conditions, with the laser turned off.

The number of photons per object pixel was calculated empirically by summing over the captured diffraction signal in each image, with the mean background signal for that imaging condition subtracted off. After multiplying by a previously calibrated conversion factor to convert between ADUs and photon counts [62], we were able to calculate the mean number of photons measured on the detector under the respective imaging condition.

To calculate the reported signal to noise ratios, we separated the noise contribution into signal-dependent and signal-independent contributions. The signal-independent portion, which included readout noise, dark current, and shot noise from background photons, was calibrated empirically using the statistics of the dark images. Specifically, the standard deviation of the background images was calculated in binned 8 by 8 pixel regions to produce a low-resolution map of the empirical signal-independent noise level. We estimated the signal-dependent contribution by assuming it is dominated by Poisson noise. Under this assumption, the standard deviation of the signal-dependent noise can be estimated by the square root of the measured signal (minus the mean background) at each pixel. The total variance at each pixel is thus determined by the sum of the squares of the standard deviations of the two contributions. The reported signal to noise ratios are defined as the ratio of the sum of the signal image (the total power in the signal channel across the entire image) to the sum of the calculated standard deviations due to noise (the total power in the noise channel across the image).

At each photon incidence rate condition, we first took a 31×31 step ptychography dataset with 75μm steps in order to retrieve the probe and background states. Scanning for the ptychography dataset was implemented by shifting a displayed image digitally across the SLM. Ptychographic reconstructions were performed via automatic differentiation ptychography using the Adam algorithm, with a single probe mode and a quadratic background correction. A learning rate scheduler was used to lower the learning rate by a factor of 0.2 at plateaus to ensure good convergence. After performing the reconstruction we displayed the ImageNet images, upsampled so that each pixel in the image covered a 2 by 2 pixel region on the SLM, in series to collect the RPI datasets.

Funding. Southern University of Science and Technology, China (028644-00030); Office of Science (DE-SC0021939); National Research Foundation Singapore.

Acknowledgments. We would like to acknowledge Mo Deng and Iksung Kang for helpful discussions. The work performed by Z.G. and G.B. was supported by the Southern University of Science and Technology, China, under Award Number 028644-00030. G.B. was also supported by Singapore NRF's Intra-Create thematic grant on Retinal Analytics through Machine learning aiding Physics (RAMP), contract NRF2019-THE002-0006. The work performed by A.L. and R.C. was supported by the Department of Energy, Office of Science, Office of Basic Energy Sciences, under Award Number DE-SC0021939. The authors state that the opinions expressed herein are our own and do not represent the opinions of the funding agencies. The authors acknowledge the MIT SuperCloud and Lincoln Laboratory Supercomputing Center for providing HPC resources that have contributed to the research results reported within this paper. We would also like to thank the NVIDIA Corporation for their donation of GPU hardware.

Disclosures. The authors declare that there are no conflicts of interest related to this article.

Data availability. The simulated and experimental datasets and analysis results presented in this paper are publicly available at Ref. [63]. The code used for reconstructions and analysis is publicly available at Ref. [64].

References

1. I. Peterson, B. Abbey, C. Putkunz, D. Vine, G. van Riessen, G. Cadenazzi, E. Balaur, R. Ryan, H. Quiney, I. McNulty, A. Peele, and K. Nugent, "Nanoscale fresnel coherent diffraction imaging tomography using ptychography," Opt. Express **20**(22), 24678–24685 (2012).
2. H. N. Chapman and K. A. Nugent, "Coherent lensless x-ray imaging," Nat. Photonics **4**(12), 833–839 (2010).
3. M. Holler, M. Odstrcil, M. Guizar-Sicairos, M. Lebugle, E. Müller, S. Finizio, G. Tinti, C. David, J. Zusman, W. Unglaub, O. Bunk, J. Raabe, A. F. J. Levi, and G. Aeppli, "Three-dimensional imaging of integrated circuits with macro- to nanoscale zoom," Nat. Electron. **2**(10), 464–470 (2019).
4. C. Y. Hémonnot and S. Köster, "Imaging of biological materials and cells by X-ray scattering and diffraction," ACS Nano **11**(9), 8542–8559 (2017).
5. C. Muehleman, J. Li, D. Connor, C. Parham, E. Pisano, and Z. Zhong, "Diffraction-enhanced imaging of musculoskeletal tissues using a conventional X-ray tube," Academic Radiol. **16**(8), 918–923 (2009).
6. J. R. Fienup, "Phase retrieval algorithms: a comparison," Appl. Opt. **21**(15), 2758–2769 (1982).
7. E. J. Candes, X. Li, and M. Soltanolkotabi, "Phase retrieval via wirtinger flow: Theory and algorithms," IEEE Trans. Inform. Theory **61**, 1985–2007 (2015).
8. Y. Shechtman, Y. C. Eldar, O. Cohen, H. N. Chapman, J. Miao, and M. Segev, "Phase retrieval with application to optical imaging: a contemporary overview," IEEE Signal Process. Mag. **32**, 87–109 (2015).

9. R. L. Sandberg, A. Paul, D. A. Raymondson, S. Hädrich, D. M. Gaudiosi, J. Holtsnider, R. I. Tobey, O. Cohen, M. M. Murnane, H. C. Kapteyn, C. Song, J. Miao, Y. Liu, and F. Salmassi, "Lensless diffractive imaging using tabletop coherent high-harmonic soft-x-ray beams," Phys. Rev. Lett. **99**(9), 098103 (2007).
10. A. Tripathi, I. McNulty, and O. G. Shpyrko, "Ptychographic overlap constraint errors and the limits of their numerical recovery using conjugate gradient descent methods," Opt. Express **22**(2), 1452–1466 (2014).
11. P. Sidorenko and O. Cohen, "Single-shot ptychography," Optica **3**(1), 9–14 (2016).
12. B. Lee, J.-Y. Hong, D. Yoo, J. Cho, Y. Jeong, S. Moon, and B. Lee, "Single-shot phase retrieval via fourier ptychographic microscopy," Optica **5**(8), 976–983 (2018).
13. D. Goldberger, J. Barolak, C. G. Durfee, and D. E. Adams, "Three-dimensional single-shot ptychography," Opt. Express **28**(13), 18887–18898 (2020).
14. A. L. Levitan, K. Keskinbora, U. T. Sanli, M. Weigand, and R. Comin, "Single-frame far-field diffractive imaging with randomized illumination," Opt. Express **28**(25), 37103–37117 (2020).
15. Z. Liu, T. Bicer, R. Kettimuthu, D. Gursoy, F. De Carlo, and I. Foster, "Tomogan: low-dose synchrotron X-ray tomography with generative adversarial networks: discussion," J. Opt. Soc. Am. A **37**(3), 422–434 (2020).
16. M. Araya-Polo, J. Jennings, A. Adler, and T. Dahlke, "Deep-learning tomography," The Leading Edge **37**(1), 58–66 (2018).
17. T. Würfl, F. C. Ghesu, V. Christlein, and A. Maier, "Deep learning computed tomography," in *International conference on medical image computing and computer-assisted intervention*, (Springer, 2016), pp. 432–440.
18. D. Ardila, A. P. Kiraly, S. Bharadwaj, B. Choi, J. J. Reicher, L. Peng, D. Tse, M. Etemadi, W. Ye, G. Corrado, D. P. Naidich, and S. Shetty, "End-to-end lung cancer screening with three-dimensional deep learning on low-dose chest computed tomography," Nat. Med. **25**(6), 954–961 (2019).
19. H. Zhang, L. Li, K. Qiao, L. Wang, B. Yan, L. Li, and G. Hu, "Image prediction for limited-angle tomography via deep learning with convolutional neural network," arXiv preprint arXiv:1607.08707 (2016).
20. I. Kang, A. Goy, and G. Barbastathis, "Dynamical machine learning volumetric reconstruction of objects' interiors from limited angular views," Light: Sci. Appl. **10**(1), 74 (2021).
21. A. Goy, G. Rughoobur, S. Li, K. Arthur, A. I. Akinwande, and G. Barbastathis, "High-resolution limited-angle phase tomography of dense layered objects using deep neural networks," Proc. Natl. Acad. Sci. U. S. A. **116**, 19848–19856 (2019).
22. Z. Guan and E. H. Tsai, "Ptychonet: Fast and high quality phase retrieval for ptychography," Tech. rep., Brookhaven National Lab.(BNL), Upton, NY (United States) (2019).
23. T. Nguyen, Y. Xue, Y. Li, L. Tian, and G. Nehmetallah, "Deep learning approach for Fourier ptychography microscopy," Opt. Express **26**(20), 26470–26484 (2018).
24. Y. Chen, Z. Luo, X. Wu, H. Yang, and B. Huang, "U-net CNN based Fourier ptychography," arXiv preprint arXiv:2003.07460 (2020).
25. L. Boominathan, M. Maniparambil, H. Gupta, R. Baburajan, and K. Mitra, "Phase retrieval for Fourier ptychography under varying amount of measurements," arXiv preprint arXiv:1805.03593 (2018).
26. J. Zhang, T. Xu, Z. Shen, Y. Qiao, and Y. Zhang, "Fourier ptychographic microscopy reconstruction with multiscale deep residual network," Opt. Express **27**(6), 8612–8625 (2019).
27. Y. Rivenson, Y. Wu, and A. Ozcan, "Deep learning in holography and coherent imaging," Light: Science & Applications **8**(1), 1–8 (2019).
28. R. Horisaki, R. Takagi, and J. Tanida, "Deep-learning-generated holography," Appl. Opt. **57**(14), 3859–3863 (2018).
29. M. H. Eybposh, N. W. Caira, M. Atisa, P. Chakravarthula, and N. C. Pégard, "DeepCGH: 3D computer-generated holography using deep learning," Opt. Express **28**(18), 26636–26650 (2020).
30. Z. Ren, H. K.-H. So, and E. Y. Lam, "Fringe pattern improvement and super-resolution using deep learning in digital holography," IEEE Trans. Ind. Inf. **15**(11), 6179–6186 (2019).
31. A. Goy, K. Arthur, S. Li, and G. Barbastathis, "Low photon count phase retrieval using deep learning," Phys. Rev. Lett. **121**(24), 243902 (2018).
32. M. Deng, S. Li, A. Goy, I. Kang, and G. Barbastathis, "Learning to synthesize: Robust phase retrieval at low photon counts," Light: Sci. Appl. **9**(1), 1–16 (2020).
33. I. Kang, F. Zhang, and G. Barbastathis, "Phase extraction neural network (PhENN) with coherent modulation imaging (CMI) for phase retrieval at low photon counts," Opt. Express **28**(15), 21578–21600 (2020).
34. Y. Rivenson, Y. Zhang, H. Günaydın, D. Teng, and A. Ozcan, "Phase recovery and holographic image reconstruction using deep learning in neural networks," Light: Sci. Appl. **7**(2), 17141 (2018).
35. Y. Xue, S. Cheng, Y. Li, and L. Tian, "Reliable deep-learning-based phase imaging with uncertainty quantification," Optica **6**(5), 618–629 (2019).
36. A. Matlock and L. Tian, "Physical model simulator-trained neural network for computational 3d phase imaging of multiple-scattering samples," arXiv preprint arXiv:2103.15795 (2021).
37. A. Fannjiang and W. Liao, "Phase retrieval with random phase illumination," J. Opt. Soc. Am. A **29**(9), 1847–1859 (2012).
38. A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," Commun. ACM **60**(6), 84–90 (2017).
39. S. Ren, K. He, R. Girshick, and J. Sun, "Faster r-cnn: Towards real-time object detection with region proposal networks," arXiv preprint arXiv:1506.01497 (2015).

40. Y. Jing, Y. Yang, Z. Feng, J. Ye, Y. Yu, and M. Song, "Neural style transfer: A review," IEEE Trans. Visual. Comput. Graphics **26**(11), 3365–3385 (2020).
41. C. Metzler, P. Schniter, A. Veeraraghavan, and R. Baraniuk, "prDeep: Robust phase retrieval with a flexible deep network," in *Proceedings of the 35th International Conference on Machine Learning*, vol. 80 of *Proceedings of Machine Learning Research* J. Dy and A. Krause, eds. (PMLR, 2018), pp. 3501–3510.
42. Y. Zhang, M. A. Noack, P. Vagovic, K. Fezzaa, F. Garcia-Moreno, T. Ritschel, and P. Villanueva-Perez, "Phasegan: A deep-learning phase-retrieval approach for unpaired datasets," arXiv preprint arXiv:2011.08660 (2020).
43. A. Sinha, J. Lee, S. Li, and G. Barbastathis, "Lensless computational imaging through deep learning," Optica **4**(9), 1117–1125 (2017).
44. M. Deng, S. Li, Z. Zhang, I. Kang, N. X. Fang, and G. Barbastathis, "On the interplay between physical and content priors in deep learning for computational imaging," Opt. Express **28**(16), 24152–24170 (2020).
45. G. Barbastathis, A. Ozcan, and G. Situ, "On the use of deep learning for computational imaging," Optica **6**(8), 921–943 (2019).
46. A. Krizhevsky and G. E. Hinton, "Using very deep autoencoders for content-based image retrieval," in *ESANN*, vol. 1 (Citeseer, 2011), p. 2.
47. M. Tan and Q. Le, "Efficientnet: Rethinking model scaling for convolutional neural networks," in *International Conference on Machine Learning*, (PMLR, 2019), pp. 6105–6114.
48. M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L.-C. Chen, "Mobilenetv2: Inverted residuals and linear bottlenecks," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, (2018), pp. 4510–4520.
49. J. Hu, L. Shen, and G. Sun, "Squeeze-and-excitation networks," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, (2018), pp. 7132–7141.
50. F. Chollet, "Xception: Deep learning with depthwise separable convolutions," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, (2017), pp. 1251–1258.
51. C. Szegedy, S. Ioffe, V. Vanhoucke, and A. Alemi, "Inception-v4, inception-resnet and the impact of residual connections on learning," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 31 (2017).
52. S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," in *International conference on machine learning*, (PMLR, 2015), pp. 448–456.
53. J. Johnson, A. Alahi, and L. Fei-Fei, "Perceptual losses for real-time style transfer and super-resolution," in *European conference on computer vision*, (Springer, 2016), pp. 694–711.
54. H. Zhao, O. Gallo, I. Frosio, and J. Kautz, "Loss functions for image restoration with neural networks," IEEE Trans. Comput. Imaging **3**(1), 47–57 (2017).
55. M. Mirza and S. Osindero, "Conditional generative adversarial nets," arXiv preprint arXiv:1411.1784 (2014).
56. A. Creswell, T. White, V. Dumoulin, K. Arulkumaran, B. Sengupta, and A. A. Bharath, "Generative adversarial networks: An overview," IEEE Signal Process. Mag. **35**(1), 53–65 (2018).
57. M. Arjovsky, S. Chintala, and L. Bottou, "Wasserstein generative adversarial networks," in *International conference on machine learning*, (PMLR, 2017), pp. 214–223.
58. R. Manekar, K. Tayal, V. Kumar, and J. Sun, "End-to-end learning for phase retrieval," in *ICML workshop on ML Interpretability for Scientific Discovery*, (2020).
59. A. Reuther, J. Kepner, C. Byun, S. Samsi, W. Arcand, D. Bestor, B. Bergeron, V. Gadepally, M. Houle, M. Hubbell, M. Jones, A. Klein, L. Milechin, J. Mullen, A. Prout, A. Rosa, C. Yee, and P. Michaleas, "Interactive supercomputing on 40, 000 cores for machine learning and data analysis," in *2018 IEEE High Performance extreme Computing Conference (HPEC)*, (IEEE, 2018), pp. 1–6.
60. S. Marchesini and A. Sakdinawat, "Shaping coherent X-rays with binary optics," Opt. Express **27**(2), 907–917 (2019).
61. D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," arXiv preprint arXiv:1412.6980 (2014).
62. I. Kang, "High-fidelity inversion at low-photon counts using deep learning and random phase modulation," Master's thesis, Massachusetts Institute of Technology (2020).
63. Z. Guo and A. Levitan, "Replication Data for: Randomized probe imaging through deep k-learning," Harvard Dataverse (2021), https://doi.org/10.7910/DVN/NZPFYK.
64. Z. Guo, A. Levitan, G. Barbastathis, and R. Comin, "Code for Randomized probe imaging through deep k-learning," Github (2021), https://github.com/zguo0525/Randomized-probe-imaging-through-deep-k-learning.