

Lawrence Berkeley National Laboratory

LBL Publications

Title

Accelerating iterative ptychography with an integrated neural network

Permalink

<https://escholarship.org/uc/item/11s067m8>

Journal

Journal of Microscopy, 300(2)

ISSN

0022-2720

Authors

McCray, Arthur RC
Ribet, Stephanie M
Varnavides, Georgios
[et al.](#)

Publication Date

2025-11-01

DOI

10.1111/jmi.13407

Copyright Information

This work is made available under the terms of a Creative Commons Attribution License, available at <https://creativecommons.org/licenses/by/4.0/>

Peer reviewed

Accelerating iterative ptychography with an integrated neural network

Arthur R. C. McCray^{1,2} | Stephanie M. Ribet² | Georgios Varnavides^{2,3} | Colin Ophus^{1,2}

¹Department of Materials Science and Engineering, Stanford University, Stanford, CA, 94305, USA

²NCEM, Molecular Foundry, Lawrence Berkeley National Laboratory, Berkeley, CA, 94720

³Miller Institute for Basic Research in Science, University of California, Berkeley, CA, 94305, USA

Correspondence

Colin Ophus, Department of Materials Science and Engineering, Stanford University, Stanford, CA, 94305, USA
Email: cophus@gmail.com

Funding information

US DOE early career research award program, basic energy sciences contract DE-AC02-05CH11231, award BES-ERCAP0028035, and the Miller Institute for Basic Research in Science.

Electron ptychography is a powerful and versatile tool for high-resolution and dose-efficient imaging. Iterative reconstruction algorithms are powerful but also computationally expensive due to their relative complexity and the many hyperparameters that must be optimized. Gradient descent-based iterative ptychography is a popular method, but it may converge slowly when reconstructing low spatial frequencies. In this work, we present a method for accelerating a gradient descent-based iterative reconstruction algorithm by training a neural network (NN) that is applied in the reconstruction loop. The NN works in Fourier space and selectively boosts low spatial frequencies, thus enabling faster convergence in a manner similar to accelerated gradient descent algorithms. We discuss the difficulties that arise when incorporating a NN into an iterative reconstruction algorithm and show how they can be overcome with iterative training. We apply our method to simulated and experimental data of gold nanoparticles on amorphous carbon and show that we can significantly speed up ptychographic reconstruction of the nanoparticles.

KEYWORDS

ptychography, 4DSTEM, machine learning, gradient descent

1 | INTRODUCTION

Phase retrieval is a critical tool for modern transmission electron microscopy (TEM). Phase reconstruction enables dose-efficient imaging which is critical when working with beam sensitive or biological samples.¹ There are many STEM phase retrieval techniques including differential phase contrast,² parallax (or tilt-corrected bright field),³ and ptychography.⁴ The advent of high-speed pixelated electron detectors has increased both the accessibility and importance of 4D scanning TEM (4DSTEM) techniques that acquire a full diffraction pattern at each probe position.⁵ Ptychography is particularly relevant for a number of materials science challenges. It enables very high resolution imaging beyond the diffraction limit^{6,7}, and it can be used to directly measure magnetic fields in the sample⁸ as well as obtain out-of-plane structural information.^{9,10}

Electron ptychography typically involves defocusing the electron probe such that there is a significant overlap in the probe between adjacent scan positions.¹¹ The complex-valued object function can then be reconstructed by leveraging redundant information in the dataset. There are multiple methods for reconstructing electron ptychography datasets.^{12–14} Single-shot methods are popular due to their computational efficiency, and recent work has shown that machine learning models can also be used in this way.^{15–18} Iterative methods, including those based on gradient descent, have proven popular and effective because they allow for the simultaneous reconstruction of both the sample object as well as the electron probe. We can use these methods to solve for the aberrated probe and deconvolve it from the object, making it possible to perform “super-resolution” imaging.^{9,19} The downside to iterative reconstruction algorithms is that they are computationally expensive and can require long run times on powerful workstations. Additionally, the algorithms require the precise knowledge of many microscope parameters, and the best reconstructions often use a hyperparameter optimization routine.^{20,21} This makes it much easier to obtain high quality images, but once again at the cost of computational resources as the reconstruction must be performed many times. The computational expense of iterative ptychography is particularly noticeable when reconstructing datasets from samples that contain nanoparticles, step edges, or holes in the sample. In all of these cases, the object function has a strong low-frequency component that iterative ptychography and other phase reconstruction methods can struggle to reconstruct.²²

We therefore want to develop methods which accelerates gradient descent ptychographic reconstruction algorithms so that they can be more widely and easily applied. Momentum-based methods for accelerating gradient descent are well understood,^{23,24} but they can be difficult to apply to non-convex and complex-valued problems such as ptychography.^{25,26} Another potential avenue for acceleration would be through loop unrolling,^{27,28} but this is difficult to extend to a problem like ptychography for which the computational graph expands significantly.

In this work, we present a method for incorporating a neural network (NN) into an iterative ptychographic reconstruction algorithm. This “in the loop” NN is applied iteratively and works in Fourier space to accelerate the low-frequency convergence of the object. This is fundamentally different from other ptychographic regularization techniques, such as low-pass filtering or sparsification, [Moshtaghpour] that selectively remove signal from the reconstructed object. Our technique adds low-frequency signal and is conceptually most similar to accelerated gradient descent. We use a pre-trained NN to strengthen the gradient step that leverages known information about the ptychographic reconstruction algorithm. The model is trained first on a general set of brown noise objects, before transfer learning on a small set (3–4 scans) of quickly reconstructed experimental data. The result is a model that accelerates ptychographic reconstructions, which can be useful for many aspects of data processing and hyperparameter optimization. This work demonstrates our acceleration method for an electron ptychography algorithm, but the ideas presented here, as well as the difficulties and proposed methods for overcoming them, are broadly applicable to incorporating NNs into other iterative algorithms.

2 | RESULTS AND DISCUSSION

2.1 | Accelerating iterative ptychography with a neural network

Figure 1 demonstrates the difficulty of reconstructing low-frequency signals with iterative ptychography for a simulated dataset of a gold nanoparticle. The ground truth phase shift is shown in Figure 1(a), and a schematic showing the ptychography imaging geometry and probe overlaps are shown in Figure 1(b) and (c). The phase shifts obtained from a conventional gradient-descent reconstruction algorithm are shown in Figure 1(d) after performing 5-5,000 reconstruction iterations. High-frequency information from the sample, including lattice fringes and the edge of the nanoparticle, is clearly visible after 50 iterations, but the low-frequency shape function of the nanoparticle has not fully converged after 5,000 iterations. The slow convergence of the low-frequency information is particularly evident when looking at the differences between the true and reconstructed phase shifts, which are shown in Figure 1(e).

Here, we have modified a conventional, gradient descent based ptychographic reconstruction algorithm by integrating a NN into the iterative reconstruction loop. Our “in-the-loop” NN is applied to a single-slice ptychographic reconstruction algorithm that is implemented in the open-source py4DSTEM package and discussed elsewhere.^{14,29} The un-modified reconstruction algorithm is referred to as the conventional method, and the same hyperparameters are used when comparing the accelerated and conventional algorithms in all cases.

The results of using our accelerated algorithm are shown in Figure 1(f) and (g), and the details of our implementation will be presented in the following sections. The accelerated algorithm is used for iterations 2-5 of the reconstruction, and the conventional algorithm is used for the remaining iterations. The NN adds a significant low-frequency component to the reconstruction that helps to define the sample shape function. This can be seen in line traces of the phase shifts across the nanoparticle, which are shown in Figure 1(h) and (i) for the conventional and accelerated reconstructions respectively. The accelerated reconstruction clearly shows that the thickness signal of the nanoparticle is mostly recovered after only 50 iterations.

2.2 | Integrating a NN into an iterative reconstruction algorithm

Figure 2 (a) outlines the conventional iterative reconstruction algorithm that we use as the base of our method. The initial complex-valued object and probe, $\mathcal{O}(\mathbf{r})$ and $\mathcal{P}(\mathbf{r})$, are used to calculate diffraction amplitudes using a single-slice ptychography forward model. These intensities are compared with the input data, $I_0(\mathbf{r}, \mathbf{k})$, from which we calculate the error. The adjoint is then used to generate a modification to the object and probe, $\Delta\mathcal{O}(\mathbf{r})$ and $\Delta\mathcal{P}(\mathbf{r})$, which are added to the original object and probe to complete one iteration.

Our accelerated algorithm that incorporates a trained NN is outlined in Figure 2(b). In this case, the Fourier transform of the starting object and its update, $\mathcal{O}(\mathbf{k})$ and $\Delta\mathcal{O}(\mathbf{k})$, are used as inputs into a complex-valued neural network. The NN output is a complex-valued patch in Fourier space containing low-frequency components of the object, $\Delta\mathbf{k}_{small}$, that is added to the object in Fourier space. This gives the NN an opportunity to selectively accelerate low-frequency components of the object and which are carried into the following iteration.

As we discuss below, the accelerated algorithm is not used for each iteration when performing an accelerated reconstruction. Instead, the conventional reconstruction algorithm is run for 2 iterations, then the accelerated algorithm is run for 1–5 iterations, and then additional conventional iterations are run as needed. We have found that applying the model for multiple iterations improved the convergence rate of low spatial frequencies, but at the cost of potentially overshooting the true phase shift or hallucinating additional signals and artifacts.

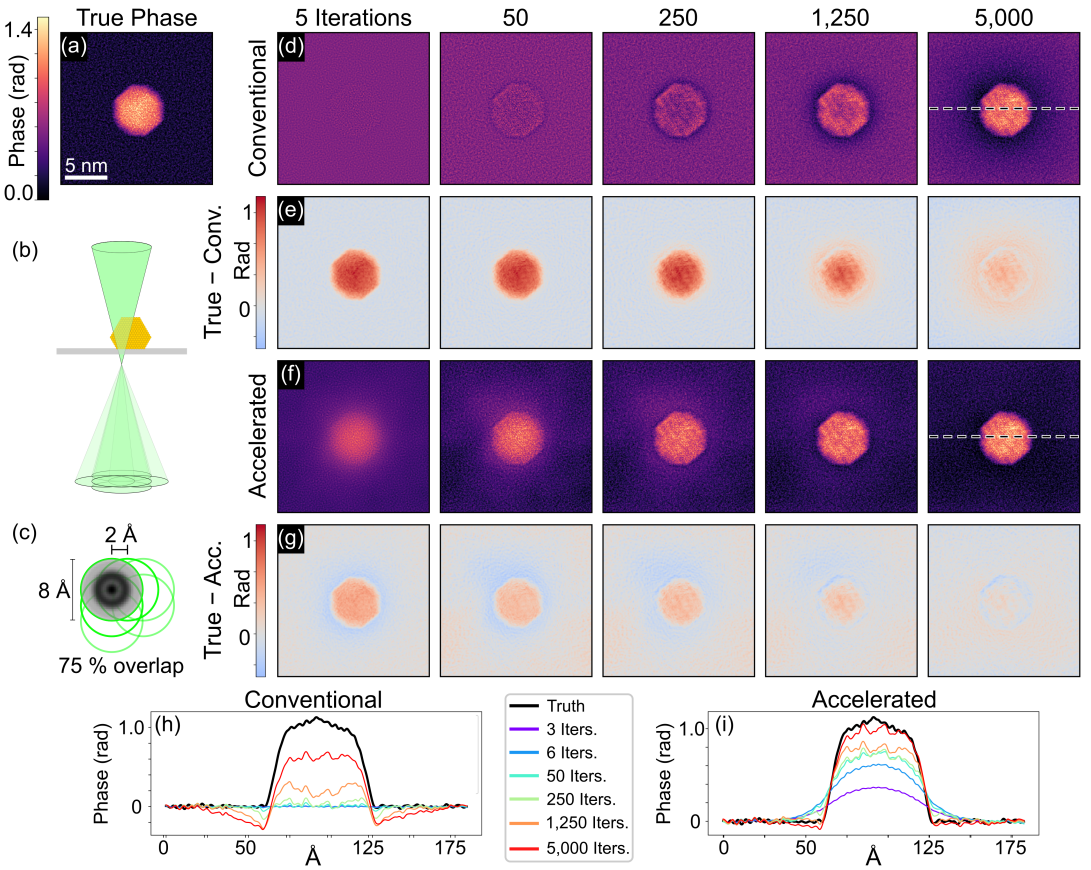


FIGURE 1 Accelerating ptychographic reconstruction for a simulated nanoparticle. (a) Ground truth phase shift for a simulated gold nanoparticle on a carbon substrate. (b) 4DSTEM schematic. (c) Diagram of overlapping probe positions. (d) Ptychographic reconstruction phase shifts of gold nanoparticle after 5-5,000 iterations. (e) Difference images between the true and reconstructed phase for each image in (d). (f) Reconstructed phase shifts after using the accelerated algorithm with a NN incorporated for three of the reconstruction iterations. (g) Difference images between the true phase shift and accelerated reconstructions. For both (e) and (g), a 2-pixel Gaussian filter was applied to both the ground truth and reconstructed images and their means were set to zero before calculating the difference. (h) and (i) Line plots across (d) and (f) for the conventional and accelerated algorithms respectively. Line plots follow the dashed lines shown in 5,000 iteration images. All phase images are plotted on the same color scale shown in (a), and all difference images are plotted using the same color scale shown in (e) and (g).

2.3 | Iterative training

The downside of implementing the NN iteratively is that it necessitates a much more complicated training process. Specifically, multiple training sets must be created in order for the model to be used in more than one iteration. For the first training set, i.e. when training a completely new model, the data can be generated in a conventional manner as follows: a large number of objects are generated, and for each object a 4DSTEM dataset is simulated using randomly chosen parameters as outlined in Table 1. Each simulated 4DSTEM dataset is reconstructed for 3 iterations using the conventional algorithm, with the object and update of the final iteration serving as the training input and the simulated object as the ground truth label. These training pairs are used to train a model that can be applied to a single iteration of the reconstruction algorithm. For this one iteration, the model

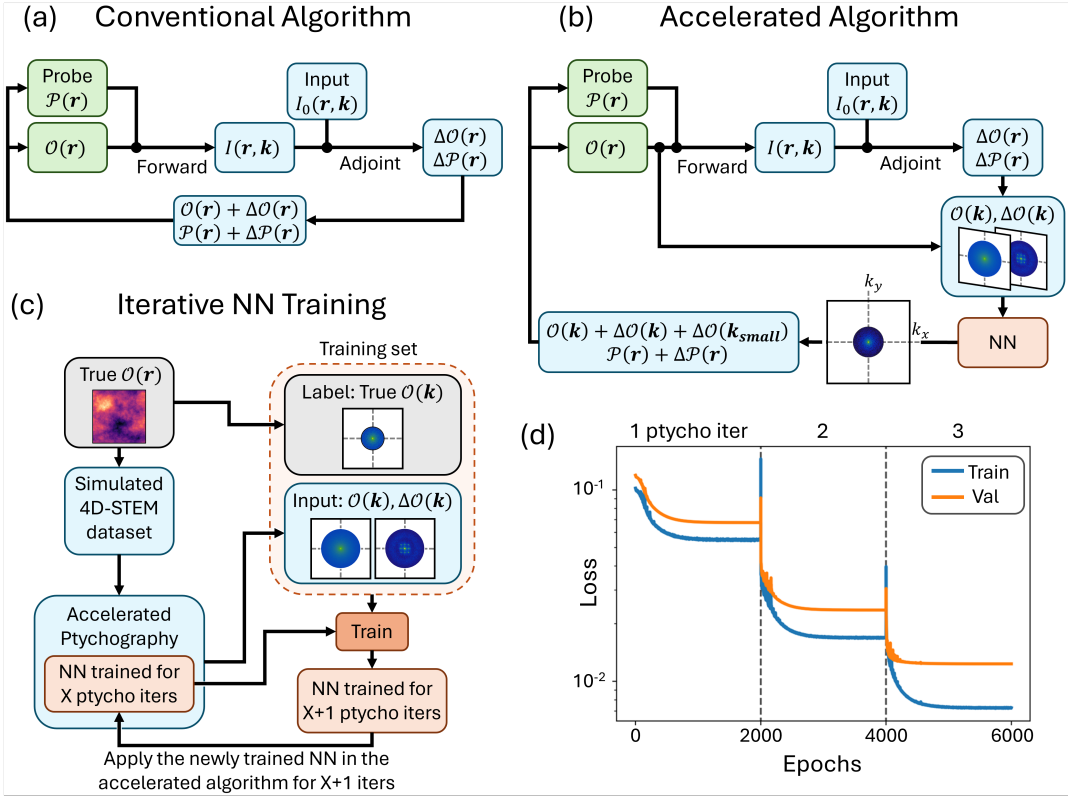


FIGURE 2 Integration of a NN into an iterative ptychographic reconstruction algorithm. (a) Overview of a conventional, gradient-descent based algorithm. (b) Overview of the accelerated algorithm that incorporates a NN into (a). (c) Diagram outlining how iterative training is performed. (d) Loss plot showing the training of a model on a set of 150,000 objects for 3 iterations of the phase reconstruction algorithm. The model is trained for 2,000 epochs for each additional iteration.

will behave quite well. It cannot be used for further iterations, however, and attempting to do so will cause the reconstruction algorithm to diverge. This is because the input object for the 4th iteration of the algorithm, the first iteration in which the object has been modified by the NN, will be unlike anything in the model's training set. The model will therefore give non-physical predictions that disrupt the reconstruction.

Applying the model iteratively requires training the model iteratively as well, and this is outlined in Figure 2(c). After the first iteration, training data for further iterations must be generated using the model that is currently being trained. The process for generating the second iteration of training data is as follows: 1) we simulate a 4DSTEM dataset for each ground truth object. 2) We reconstruct each dataset for four iterations. The third iteration uses the accelerated algorithm outlined in Figure 2(b) and the current version of the model. The inputs to the model are therefore the object and update that was previously used as training data. 3) The object and update of the third and fourth iterations are stored as the next set of training data. This training set now incorporates objects from the third iteration (that result from the conventional algorithm and that are the same as what was in the first training set) as well as objects from the fourth iteration that have been modified by the model in the third iteration. The second training set is therefore twice as big as first training set, though it still contains the same number of unique object labels. The model can be trained on this dataset and the resulting model can be used for two accelerated reconstruction iterations. This

Parameter	Minimum value	Maximum value
Scan sampling	1 Å	2.5 Å
# scan positions	30 × 30	150 × 150
Defocus	±200 Å	±500 Å
Convergence semiangle	15 mrad	25 mrad
Probe overlap	72 %	96 %
hline Accelerating voltage	300 kV	300 kV
Dose	$1 \times 10^5 \text{ e}/\text{\AA}^2$	∞
Detector shape	128 × 128	192 × 192

TABLE 1 Parameters used for simulating the training dataset.

can be repeated for further iterations, with the X 'th iteration requiring ptychographic reconstructions be run for $X + 2$ iterations for each item in the training set.

This method of training is computationally expensive. 4DSTEM datasets are quite big, and it was therefore necessary to generate them on-the-fly when training rather than writing them to disk. Our model was trained using 150,000 unique brown-noise objects. Their corresponding 4DSTEM datasets would have required approximately 135 TB of disk space, which was beyond our available resources. It took approximately 4,000 GPU hours to train our model for 3 iterations using a training set of 150,000 objects. Figure 2(d) shows the training and validation loss when training our model. The clear separation between the training and validation curves is an unfortunate result of the relatively small training set size, and it indicates over-training. We note that while the loss curves do flatten out, the model performance increases significantly on the validation data between 1,000 and 2,000 epochs of training even though the validation loss does not significantly decrease, see Supplementary Figure S1. The model was implemented in pyTorch and trained using a mean squared error loss function with the Adam optimizer and a learning rate of 1×10^{-3} .

2.4 | Complex-valued NN architecture

The model used in the accelerated algorithm is a complex-valued NN that receives, as input, two patches in Fourier space and produces, as output, a smaller patch in Fourier space that is added to the Fourier transform of the object in the reconstruction algorithm. Figure 3(a) and (b) show the Fourier transforms of the object and update respectively, $\mathcal{O}(\mathbf{k})$ and $\Delta\mathcal{O}(\mathbf{k})$, and the central patches of radii 32 pixels are shown in Figure 3(c) and (d). These patches are flattened and used as inputs into the NN. The NN, outlined in Figure 3(e), is a complex-valued dense network with separate branches for the object and update. In each hidden layer, the output from the update branch is added to the output from the object branch, thus encouraging the model to prioritize information coming from the update. The output of the model is reshaped into a patch of radius 16 pixels as shown in Figure 3(f), which is added to the object along with the update in order to produce new object for the next iteration. We used a 16 pixel radius output patch so that our model would only be able to affect the lowest spatial frequencies in the object, and the larger 32 pixel radius input patch was used so that the model had additional context and could better learn which spatial frequencies should be enhanced.

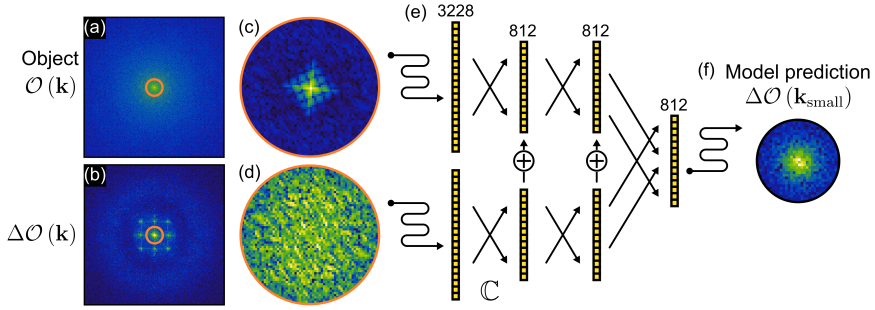


FIGURE 3 Inputs and architecture of the NN. (a) and (b) Fourier transforms are taken of the reconstructed object, $O(k)$, and the object update, $\Delta O(k)$. (c) and (d) Central patches of radius 32 pixels are taken from the object and update and used as the inputs to the NN. (e) The NN is a complex-valued dense network consisting of two hidden layers with the output from each update layer added to the object layer. (f) The NN prediction is a patch in Fourier space of radius 16 pixels that is added to the object.

2.5 | Training on noise

In order to generalize the model for as wide range of materials as possible, we simulated the training sets from brown noise objects. Brown noise was chosen so that the model would learn to accelerate the full spectrum of low frequencies within its band limit, and because the model only interacts with the low-frequency component of the object, high frequency information in the training data is less important. Training on brown noise therefore allowed the model to be widely applicable to most samples. Figure 4 shows examples of the noise objects used in the training set. The true phase object is shown in Figure 4(a), and a real-space representation of the label is shown in Figure 4(b). Note that (b) is simply a low-pass filtered version of (a). Figure 4(c) shows the reconstructed object phase after two iterations; the Fourier transform of this object (and the update) are used as inputs to the NN. There are gridding artifacts visible in the reconstructed object after two iterations that disappear as the reconstruction converges over additional iterations. These artifacts occur at high frequencies and therefore do not affect the NN training. The phase contributed by the model is shown in Figure 4(d), and the phase after the single accelerated reconstruction iteration is shown in (e). Figure 4(f)–(o) show the same items as (a)–(e) for two other objects from the training set.

The costly nature of training and the resulting small training set led to difficulties when generalizing our model. We found that we could successfully train and apply a model for a single set of imaging conditions over a small field of view, and models trained on purely brown noise worked well when applied to a very wide range of other objects, including for samples of nanoparticles. These experiments demonstrated the viability of our training method and also provided an opportunity to optimize the model architecture. However, generalizing to the full range of experimental conditions used for electron ptychography was unfeasible given the restricted training set size, and this led the model to be less stable for real-world application. We therefore applied transfer learning, with the intention that the base model would require a small amount of further training for the specific imaging conditions of a given experiment. Because the model only has access to low spatial frequencies, differential phase contrast (DPC) reconstructions were used to generate the training data. In contrast to ptychography, DPC allows for rapid reconstruction of low-spatial frequencies, which makes it possible to quickly generate a training set.¹⁴

2.6 | Spatial frequency dependent convergence

As demonstrated in Figure 1, a NN is able to significantly accelerate the reconstruction of low spatial frequencies. However, as discussed in Section 2.3, the accelerated algorithm cannot be used for every iteration as we were unable to train a model that would not eventually cause the reconstruction to diverge. We therefore want to evaluate the effect the model can have in the

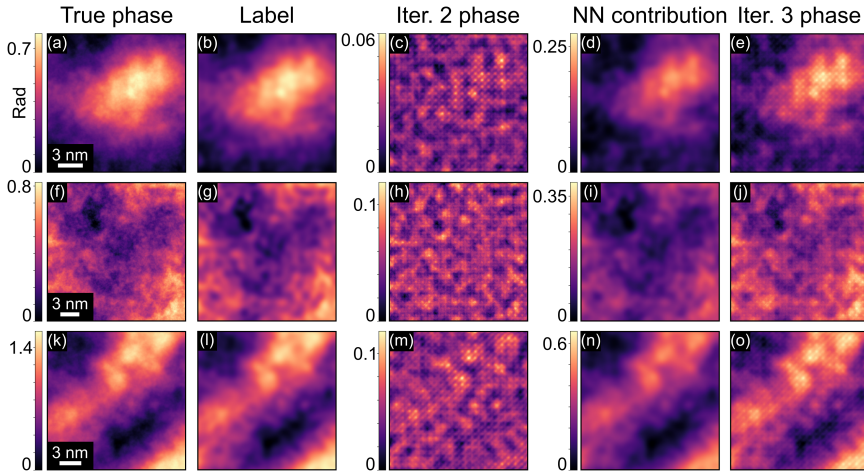


FIGURE 4 Examples of brown noise training data. (a) Ground truth phase shift of a simulated brown noise object. (b) Ground truth label shown in realspace, plotted on the same color scale as (a). (c) Object phase after 2 iterations of ptychographic reconstruction which is used as part of the input to the NN. (d) Contribution to the reconstructed phase made by the NN. (e) Object phase after the third iteration of the reconstruction algorithm that incorporated the NN, plotted on the same color scale as (d). (f)-(o) Same as (a)-(e) showing two more objects taken from the training set.

relatively few iterations for which it is applied. As a method of quantifying this, Figure 5 plots the spatial-frequency dependent error over the course of a reconstruction of a brown noise object, chosen to ensure that a broad range of spatial frequencies are well represented. We calculate this by computing the ℓ_1 error between the reconstructed and ground truth objects for annuli in Fourier space. This allows us to quantify the accuracy of the reconstruction at specific spatial frequencies.

Figure 5(a) shows for a single spatial frequency, $0.01 \text{ \AA}^{-1}$, the error from the conventional reconstruction is plotted with a dashed line, and the errors from accelerated reconstructions are plotted with solid lines. The accelerated algorithm can be used for one or more iterations during the reconstruction. Figure 5(b) plots the error from reconstructions that have used the accelerated algorithm for 1 to 5 iterations. These are plotted in different shades for each spatial frequency, with the darkest shade corresponding to the reconstruction that used the accelerated algorithm for 5 iterations, and the lightest shade corresponds to a reconstruction that used the accelerated algorithm for 1 iteration.

Beginning with $0.01 \text{ \AA}^{-1}$, plotted in blue, we see that there is a significant reduction in error for each iteration that the accelerated algorithm is used. This indicates that the model is unable to completely predict the low spatial frequencies from a single iteration, and that it is therefore important to use the accelerated algorithm iteratively. Once the model has been removed from the reconstruction loop, the error decreases at the same rate as for a conventional reconstruction until reaching the minimum value.

We observe similar trends for the higher spatial frequencies $0.1 \text{ \AA}^{-1}$ and $0.3 \text{ \AA}^{-1}$, plotted in orange and green respectively. In both cases, there is a significant improvement when using the accelerated algorithm for 5 iterations compared to a single iteration. The higher spatial frequencies have both a lower error and faster rate of convergence for both conventional and accelerated reconstructions.

For this dataset, the band limit for which the model could directly accelerate was $\leq 0.6 \text{ \AA}^{-1}$. We do see, however, that the convergence at higher spatial frequencies was indirectly accelerated as well. This can be seen in the separation between the error of the conventional and accelerated reconstructions at $1.5 \text{ \AA}^{-1}$ (plotted in red). The NN predictably has less of an impact at higher spatial frequencies, and the acceleration is also delayed until later in the reconstruction process. This is because the model

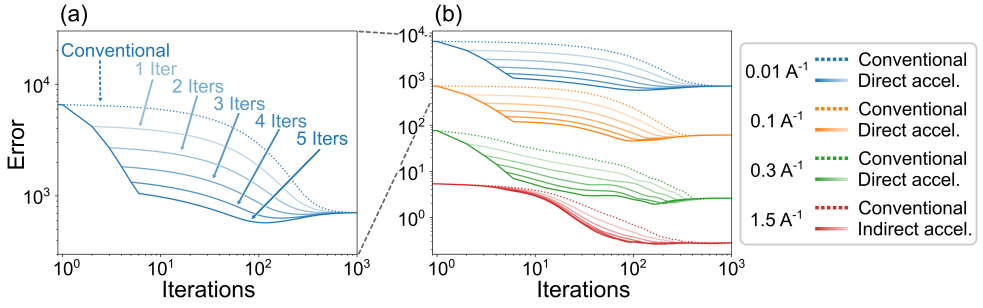


FIGURE 5 Error in the reconstructed phase of a brown noise object plotted for discrete spatial frequencies. (a) The L1 error when reconstructing without a NN is plotted with a dashed line, and the accelerated reconstructions are plotted with increasing color intensity corresponding to using the NN for 1-5 iterations of the reconstruction. (b) Plotting the error for 4 different spatial frequencies.

is not directly improving the high spatial frequencies, but the improved convergence of low spatial frequencies indirectly affects the convergence of high spatial frequencies as well.²²

2.7 | Application to experimental data

We apply our accelerated Ptychography algorithm to experimental data of gold nanoparticles on an amorphous carbon substrate. Figure 6(a)-(d) shows phase shifts after 10, 50, 250, and 1000 iterations of the conventional reconstruction algorithm. The high frequency atomic lattice information of the sample is visible after 10 iterations, as highlighted in Figure 6(e), but it takes hundreds of iterations for the shape function of the nanoparticle to reconstruct.

Applying our accelerated reconstruction algorithm to experimental datasets can be challenging due to the limited size dataset used to train the NN. While we found that brown noise objects generalized very well to arbitrary object functions including nanoparticles, the wide range of experimental imaging parameters and the associated variations in the reconstruction made it difficult to simulate a generally applicable model that could apply to any experimental dataset. Our solution was to perform a modest amount of transfer learning that would specify the base model for the particular experimental conditions used. The training set consisted of three different 4DSTEM datasets that were acquired during the same imaging session as the final dataset that we present here. These three scans were reconstructed using DPC as outlined in Supplementary Figure S2. We also verified the importance of training the base model. A NN trained on only the data used for transfer learning caused the accelerated reconstruction algorithm to diverge completely.

Figure 6(f)-(i) and (k)-(n) show the phase shifts when the accelerated algorithm is applied for 1 and 5 iterations respectively. The model is able to enhance the low-frequency component of the reconstruction and the shape functions of the nanoparticles are clearly visible after 10 iterations. While it does take many more iterations for even the accelerated algorithm to fully converge, having a significantly improved reconstruction at just 10 iterations can be very useful as it enables more efficient data processing. Using the accelerated algorithm also does not reduce the accuracy of high-frequency information in the reconstruction, as seen in the magnified images in Figure 6(j) and (o). Figure 6(p)-(s) show line plots, across the white line in (k), for all images. They show that all three reconstructions end with the same object after 1000 iterations. Figure 6 (s) validates our use of DPC data in the transfer learning training set, as it shows that the accelerated and conventional reconstructions converge to the same final phase. The accelerated algorithm does not affect the reconstruction of the probe, and this is shown in Supplementary Figure S3.

Unlike with the simulated data, here we see that there is less improvement to be gained by applying the model during multiple reconstruction iterations. Applying the model for 5 iterations does lead to a stronger low-frequency phase shifts compared to just

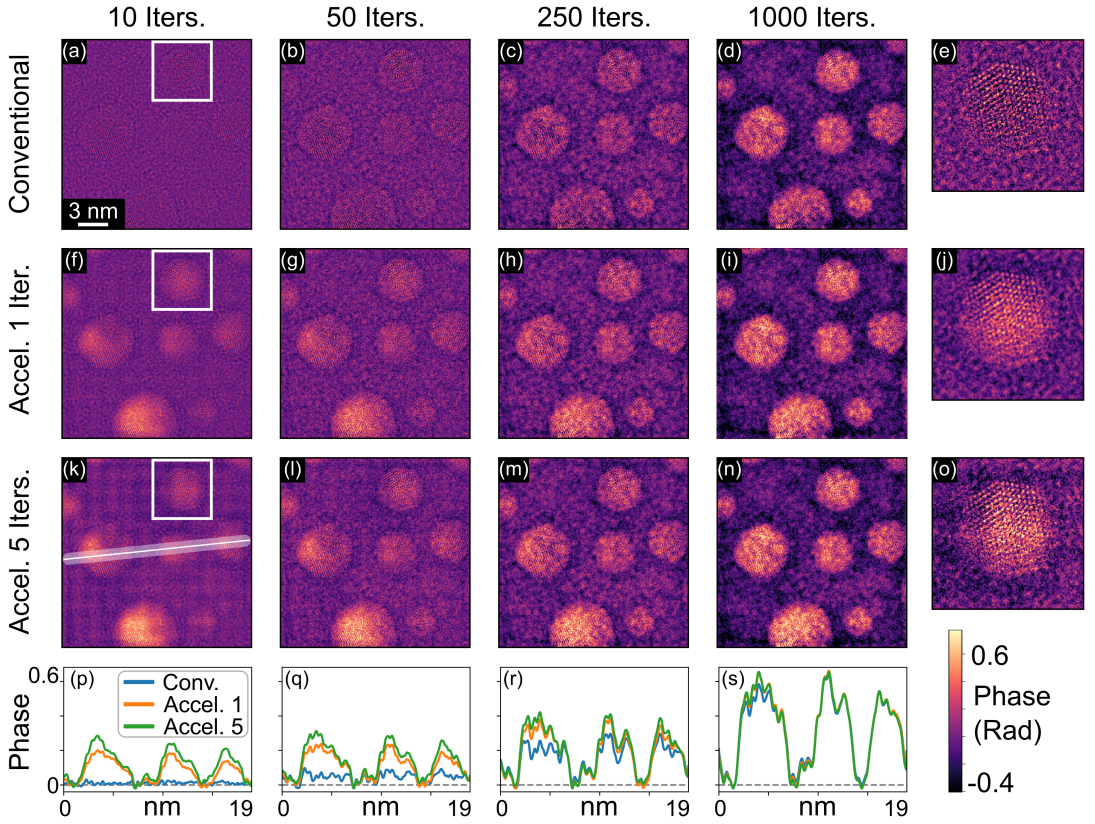


FIGURE 6 Application to experimental data of gold nanoparticles. (a)-(d) Conventional ptychographic reconstructed phases after 10, 50, 250, and 1,000 iterations. (e) Magnified image of the white box in (a). (f)-(j) The same as (a)-(e) but for the reconstruction performed with 1 iteration of the NN. (k)-(o) The same as (f)-(j) but with a NN applied for 5 iterations in the reconstruction. All images are shown on the same color scale. (p) Line plots of the 50 iteration phase reconstructions, (a), (f), and (k), across the white line shown in (k). (q)-(s) The same as (p) but for 50, 250, and 1,000 reconstruction iterations.

one iteration, but it also increases the magnitude of the square artifacts as seen in Figure 6(k). This appears slightly different from gridding artifacts that can occur during conventional ptychographic reconstructions,¹⁴ and they are quickly fixed by additional reconstruction iterations as seen in Figure 6(l).

We believe that the work presented here is interesting as it demonstrates a method for accelerating iterative ptychographic reconstructions algorithms, and also more broadly for the manner in which we have incorporated supervised learning into an iterative reconstruction algorithm. Beginning with the direct applications of accelerated ptychography, we anticipate that this would be of potential interest when processing many datasets and performing hyperparameter optimization. In cases of processing a large amount of data with similar acquisition parameters, it could be worthwhile to perform the additional transfer learning and apply the accelerated algorithm. However we also realize that there is significant room for improvement in its current form. We suspect that many of the faults of the model, namely the lack of generalizability, could be remedied by increasing the size of the training set. This was something we observed in early development when working within a restricted set of imaging conditions, but we were unable to fully scale the training set of the final model due to computational limitations. Having a generally applicable model that does not require transfer learning would greatly increase the usefulness, as it could then be

immediately applied towards live at-the-microscope data reconstructions. It is possible that altering the loss function could create a more conservative model that is more stable, though at the cost of less efficiently accelerating the reconstruction. It is also possible that other model architectures would be more easily generalized, though performing an extensive study of this would be difficult due to the expensive nature of the training.

It is also interesting to consider the more general implications of incorporating supervised machine learning into an iterative algorithm. We have demonstrated that a NN can be used to intelligently accelerate gradient descent, and that this can be applied to highly convex and complex-valued problems like ptychography. It is very difficult to apply methods like loop unrolling to ptychography algorithms, as unrolling just a few iterations causes the computational graph to become very large. The work presented here could potentially be generalized to achieve similar ends, and it presents an alternate path towards tasks like optimizing reconstruction parameters.

3 | CONCLUSION

In this work, we developed a method to accelerate gradient descent-based ptychographic reconstruction algorithms by integrating a pre-trained NN into the iterative loop. The NN operates on the low spatial frequencies in Fourier space, significantly improving convergence rates while preserving high-frequency accuracy. We demonstrated this approach on simulated datasets, where the NN effectively accelerated reconstruction of challenging low-frequency components. Although successful when applied to simulated data, the computational cost of the iterative training scheme made it difficult to generalize the NN to a wide range of experimental imaging parameters. We therefore applied transfer learning with differential phase contrast data, allowing our NN approach to then be applied to experimental data of gold nanoparticles on an amorphous carbon substrate. This method offers potential for more efficient data processing and hyperparameter optimization in ptychography and related iterative algorithms. Future work will focus on expanding the training set to improve generalizability, applying this method to more difficult datasets such as from low-dose experiments, and exploring broader applications for machine learning in reconstruction optimization.

4 | SUPPLEMENTAL

Additional supplementary figures are available in the attached pdf.

EXPERIMENTAL METHODS

Experimental data in Figure 6 was taken on a sample of gold nanoparticles on a carbon grid. The data was acquired on the TEAM I microscope at the National Center for Electron Microscopy, a modified FEI Titan double-aberration-corrected microscope. The TEAM I was operated at 300kV with a probe defined by a 22.6 mrad convergence angle. A step size of 2.4 Å was used, and the probe was defocused by about 40 nm. Data was acquired on the Dectris Arina camera.

ACKNOWLEDGMENTS

ARCM is supported by the US DOE program “4D Camera Distillery: From Massive Electron Microscopy Scattering Data to Useful Information with AI/ML. GV is support by the Miller Institute for Basic Research in Science. SR and CO acknowledge support from the US Department of Energy Early Career Research Award program. Work at the Molecular Foundry was supported by the Office of Science, Office of Basic Energy Sciences, of the U.S. Department of Energy under Contract No. DE-AC02-05CH11231. This research used resources of the National Energy Research Scientific Computing Center (NERSC), a

Department of Energy Office of Science User Facility using NERSC award BES-ERCAP0028035.

CONFLICT OF INTEREST

The authors have no conflicts of interest to declare.

CODE AVAILABILITY STATEMENT

The code and notebooks from this study will be published at [zenodo link] upon acceptance of the paper.

REFERENCES

1. Zhou L, Song J, Kim JS, et al. Low-dose phase retrieval of biological specimens using cryo-electron ptychography. *Nature Communications* 2020; 11(1): 2773. doi: 10.1038/s41467-020-16391-6
2. Shibata N, Findlay SD, Kohno Y, Sawada H, Kondo Y, Ikuhara Y. Differential phase-contrast microscopy at atomic resolution. *Nature Physics* 2012; 8(8): 611–615. doi: 10.1038/nphys2337
3. Varnavides G, Ribet SM, Ophus C. Tilt-Corrected BF-STEM. *Elemental Microscopy* 2024. doi: 10.69761/XUNR2166
4. Küçükoğlu B, Mohammed I, Guerrero-Ferreira RC, et al. Low-dose cryo-electron ptychography of proteins at sub-nanometer resolution. *Nature Communications* 2024; 15(1): 8062. doi: 10.1038/s41467-024-52403-5
5. Ophus C. Four-Dimensional Scanning Transmission Electron Microscopy (4D-STEM): From Scanning Nanodiffraction to Ptychography and Beyond. *Microscopy and Microanalysis* 2019; 25(3): 563–582. doi: 10.1017/S1431927619000497
6. Nellist PD, Rodenburg JM. Electron Ptychography. I. Experimental Demonstration Beyond the Conventional Resolution Limits. *Acta Crystallographica Section A Foundations of Crystallography* 1998; 54(1): 49–60. doi: 10.1107/S0108767397010490
7. Jiang Y, Chen Z, Han Y, et al. Electron ptychography of 2D materials to deep sub-ångström resolution. *Nature* 2018; 559(7714): 343–349. doi: 10.1038/s41586-018-0298-5
8. Chen Z, Turgut E, Jiang Y, et al. Lorentz electron ptychography for imaging magnetic textures beyond the diffraction limit. *Nature Nanotechnology* 2022; 17(11): 1165–1170. doi: 10.1038/s41565-022-01224-y
9. Chen Z, Jiang Y, Shao YT, et al. Electron ptychography achieves atomic-resolution limits set by lattice vibrations. *Science* 2021; 372(6544): 826–831. doi: 10.1126/science.abg2533
10. Ribet SM, Varnavides G, Pedroso CCS, et al. Uncovering the three-dimensional structure of upconverting core-shell nanoparticles with multislice electron ptychography. *Applied Physics Letters* 2024; 124(24): 240601. doi: 10.1063/5.0206814
11. Rodenburg J. Ptychography and Related Diffractive Imaging Methods. In: *Advances in Imaging and Electron Physics*. 150. Elsevier. 2008 (pp. 87–184)
12. Pennycook TJ, Lupini AR, Yang H, Murfitt MF, Jones L, Nellist PD. Efficient phase contrast imaging in STEM using a pixelated detector. Part 1: Experimental demonstration at atomic resolution. *Ultramicroscopy* 2015; 151: 160–167. doi: 10.1016/j.ultramic.2014.09.013
13. Rodenburg JM, Bates RHT. The theory of super-resolution electron microscopy via Wigner-distribution deconvolution. *Philosophical Transactions of the Royal Society of London. Series A: Physical and Engineering Sciences* 1992; 339(1655): 521–553. doi: 10.1098/rsta.1992.0050
14. Varnavides G, Ribet SM, Zeltmann SE, et al. Iterative Phase Retrieval Algorithms for Scanning Transmission Electron Microscopy. 2023. arXiv:2309.05250

15. Chang DJ, O’Leary CM, Su C, et al. Deep-Learning Electron Diffractive Imaging. *Physical Review Letters* 2023; 130(1): 016101. doi: 10.1103/PhysRevLett.130.016101
16. Babu AV, Zhou T, Kandel S, et al. Deep learning at the edge enables real-time streaming ptychographic imaging. *Nature Communications* 2023; 14(1): 7059. doi: 10.1038/s41467-023-41496-z
17. Seifert J, Bouchet D, Loetgering L, Mosk AP. Efficient and flexible approach to ptychography using an optimization framework based on automatic differentiation. *OSA Continuum* 2021; 4(1): 121. doi: 10.1364/OSAC.411174
18. Friedrich T, Yu CP, Verbeeck J, Van Aert S. Phase Object Reconstruction for 4D-STEM using Deep Learning. *Microscopy and Microanalysis* 2023; 29(1): 395–407. doi: 10.1093/micmic/ozac002
19. Ning S, Xu W, Loh L, et al. An integrated constrained gradient descent (iCGD) protocol to correct scan-positional errors for electron ptychography with high accuracy and precision. *Ultramicroscopy* 2023; 248: 113716. doi: 10.1016/j.ultramic.2023.113716
20. Pelz PM, Qiu WX, Bücke R, Kassier G, Miller RJD. Low-dose cryo electron ptychography via non-convex Bayesian optimization. *Scientific Reports* 2017; 7(1): 9883. doi: 10.1038/s41598-017-07488-y
21. Cao MC, Chen Z, Jiang Y, Han Y. Automatic parameter selection for electron ptychography via Bayesian optimization. *Scientific Reports* 2022; 12(1): 12284. doi: 10.1038/s41598-022-16041-5
22. Ophus C, Ewalds T. Guidelines for quantitative reconstruction of complex exit waves in HRTEM. *Ultramicroscopy* 2012; 113: 88–95. doi: 10.1016/j.ultramic.2011.10.016
23. Nesterov Y. *Introductory Lectures on Convex Optimization*. 87 of *Applied Optimization*. Boston, MA: Springer US . 2004
24. Kingma DP, Ba J. Adam: A Method for Stochastic Optimization. 2017. arXiv:1412.6980.
25. Xu R, Soltanolkotabi M, Haldar JP, et al. Accelerated Wirtinger Flow: A fast algorithm for ptychography. 2018. arXiv:1806.05546
26. Zhang B, Liu Y, Cao J, Wu S, Wang J. Fully complex conjugate gradient-based neural networks using Wirtinger calculus framework: Deterministic convergence and its application. *Neural Networks* 2019; 115: 50–64. doi: 10.1016/j.neunet.2019.02.011
27. Monga V, Li Y, Eldar YC. Algorithm Unrolling: Interpretable, Efficient Deep Learning for Signal and Image Processing. *IEEE Signal Processing Magazine* 2021; 38(2): 18–44. doi: 10.1109/MSP.2020.3016905
28. Adler J, Oktem O. Learned Primal-Dual Reconstruction. *IEEE Transactions on Medical Imaging* 2018; 37(6): 1322–1332. doi: 10.1109/TMI.2018.2799231
29. Savitzky BH, Zeltmann SE, Hughes LA, et al. py4DSTEM: A Software Package for Four-Dimensional Scanning Transmission Electron Microscopy Data Analysis. *Microscopy and Microanalysis* 2021; 27(4): 712–743. doi: 10.1017/S1431927621000477