Biomedical Optics EXPRESS

Towards ultrafast quantitative phase imaging via differentiable microscopy [Invited]

UDITH HAPUTHANTHRI,[1,2] KITHMINI HERATH,[1,2] RAMITH HETTIARACHCHI,[1,2] HASINDU KARIYAWASAM,[1,2] AZEEM AHMAD,[3] BALPREET S. AHLUWALIA,[3] GANESH ACHARYA,[4] CHAMIRA U. S. EDUSSOORIYA,[2,5] AND DUSHAN N. WADDUWAGE[1,*]

[1] *Center for Advanced Imaging, Faculty of Arts and Sciences, Harvard University, Cambridge, MA 02138, USA*
[2] *Department of Electronic and Telecommunication Engineering, University of Moratuwa, Sri Lanka*
[3] *Department of Physics and Technology, UiT The Arctic University of Norway, Tromsø, 9037, Norway*
[4] *Division of Obstetrics and Gynecology, Department of Clinical Science, Intervention and Technology, Karolinska Institute, Stockholm, Sweden*
[5] *chamira@uom.lk*
[*] *wadduwage@fas.harvard.edu*

Abstract: With applications ranging from metabolomics to histopathology, quantitative phase microscopy (QPM) is a powerful label-free imaging modality. Despite significant advances in fast multiplexed imaging sensors and deep-learning-based inverse solvers, the throughput of QPM is currently limited by the pixel-rate of the image sensors. Complementarily, to improve throughput further, here we propose to acquire images in a compressed form so that more information can be transferred beyond the existing hardware bottleneck of the image sensor. To this end, we present a numerical simulation of a learnable optical compression-decompression framework that learns content-specific features. The proposed differentiable quantitative phase microscopy (∂-QPM) first uses learnable optical processors as image compressors. The intensity representations produced by these optical processors are then captured by the imaging sensor. Finally, a reconstruction network running on a computer decompresses the QPM images post aquisition. In numerical experiments, the proposed system achieves compression of × 64 while maintaining the SSIM of ~0.90 and PSNR of ~30 dB on cells. The results demonstrated by our experiments open up a new pathway to QPM systems that may provide unprecedented throughput improvements.

1. Introduction

Among the label-free imaging modalities, quantitative phase microscopy (QPM) is a simple but powerful approach, providing important biophysical information by quantifying optical phase differences [1,2]. From the phase map, one can further yield quantitative information about the morphology and dynamics of the examined specimens [3,4]. In addition to morphology, the measured phase maps can be converted to dry mass of the cells with accuracy that is of the order of femtograms per square microns [5,6]. QPM has found many important applications in biomedicine [7] including pathogen screening [8], cancer cell classification [9], and label-free analysis of histopathology specimens [10,11]. Moreover, recently quantitative phase imaging has even been extended to image the structures of thick biological systems such as zebrafish larval [12].

The first phase imaging mechanism was introduced by Zernike in his phase contrast microscopy [13]. Here, the phase shifts due to the refractive indices and depth differences in the specimen are converted into detectable intensity variations. Zernike's original design consisted of a phase filter which directly displays phase information by interfering scattered portion of light from an

#504954 https://doi.org/10.1364/BOE.504954
Journal © 2024 Received 6 Sep 2023; revised 15 Dec 2023; accepted 9 Feb 2024; published 22 Feb 2024

image, with its unscattered portion. Even though the work improved with several extensions [14,15], due to the non-linear dependency between phase and intensity, direct phase contrast techniques are incapable of quantitative phase measurements. QPM techniques overcome this problem by computational inverse reconstruction [7]. A typical quantitative phase microscope consists of an optical system (forward model) and a computational phase retrieval algorithm (inverse model) [16]. The forward optical system converts undetectable phase information into detectable interferometric fringe patterns; from the fringe patterns, the inverse reconstruction algorithm retrieves phase and intensity maps of the specimen. Recent developments in QPM have mostly been focused on improving the inverse reconstruction using GPU acceleration [17–19], deep-learning-based inverse solvers [20–25], and illumination patterns optimization [26,27].

These advancements have placed QPM in a unique position to measure large cell populations for applications in cytometry, a field currently dominated by flow cytometers. Most commercial flow cytometers can easily analyze hundreds of thousands of cells per second. But QPM-based image cytometers are currently orders of magnitude slower. The main bottleneck of QPM is the image acquisition speed, which is fundamentally governed by the pixel rate of image sensors. Currently, the pixel rate of a state-of-the-art camera is around 1×10^{10} pixels/sec. However, the pixel throughput of the front-end optics is virtually unlimited. An image passes through optics at the speed of light and has been the rationale for developing optical signal processors [28]. Here we propose to exploit this property to optically compress an image in order to measure the compressed form of the image using a high-speed light detector (such as a high-speed camera). Thus the pixel throughput of the original image would be increased at a rate proportional to the degree of compression.

Compressive imaging of biological specimens, using random sampling of the linearly projected image space, has been demonstrated before [29]. Better compression, however, may be achieved through learning dataset-specific features of images. To this end, here we propose to use differentiable microscopy ($\partial\mu$) [30,31] to identify important image features for compression, through machine learning. Our method consists of an optical processor, a camera sensor, and a deep neural network. The optical processor encodes phase information of an input light field onto the sensor. The sensor compressively measures the intensity of this output field. The measured intensity map is then used by the neural network to reconstruct the phase map of the original input light field. We use machine learning to co-design the optical processor and the decoding neural network end-to-end. We call this measurement scheme differentiable quantitative phase microscopy (∂-QPM). In numerical simulations, we show that our proposed approach can image phase information of in-vitro cells at $\times 64 - \times 256$ compression, accelerating image acquisition by the same amount. We thus demonstrate that orders of magnitude faster QPM is feasible through ∂-QPM. Of note, this work only presents a simulation of the optical processor, leaving the implementation to future work.

In the following sections, we first introduce the proposed ∂-QPM (section 2.1). Second, we assess the feasibility of using optical processors as image compressors, despite them being linear operators (section 2.2). Third, we demonstrate ∂-QPM (in simulations) for in-vitro cells at $\times 64 - \times 256$ compression. Last, we discuss multiple aspects of the proposed measurement paradigm including potential avenues to implement the optical processors.

2. Results

2.1. Differentiable quantitative phase microscopy (∂-QPM)

Figure 1 shows the schematic of the proposed ∂-QPM scheme that consists of an optical processor, a camera sensor, and a neural network. The optical processor maps an input light field (at the image plane of the microscope), to an output light field. We design the optical processor such that, the low-frequency intensity components of the output field, encode information about the phase of the input field. The output field is then imaged at low resolution using a camera sensor.

The sensor is smaller than what is required to measure the original input field at the Nyquist sampling rate, thereby performing a "compressive measurement". The measured intensity map is then "decompressed" and decoded using the neural network, to reconstruct the phase map of the original input field. Notably, in ∂-QPM, each sensor pixel codes for multiple pixels of the original input light field. We call this number the "compression". For any given camera, the compression is directly proportional to the improvement of imaging speed. Below, we write the mathematical model of the above process.

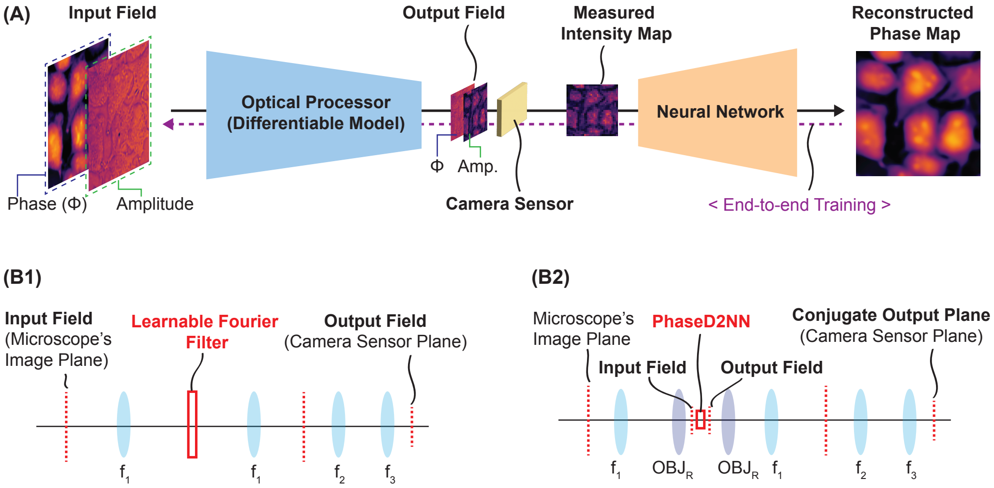

Fig. 1. **Overview of differentiable quantitative phase microscopy (∂-QPM)**: **(A)** End-to-end pipeline of ∂-QPM. The input light field propagates through an optical processor to produce the output light field. The output field is imaged using a smaller camera sensor at low resolution. The intensity map, imaged by the camera, is fed to the neural network to reconstruct a high-resolution phase map of the original input light field. **(B1)** A potential design of the optical processor using a Fourier filter with learnable transmission coefficients. All lenses (f_1, f_2, and f_3) are placed at 4f configurations. **(B2)** Another potential design of the optical processor using a diffractive neural network (PhaseD2NN). Here, f_1, f_2, and f_3 are lenses; two OBJ_R elements are objective lenses. The first OBJ_R forms a remote focus. The PhaseD2NN is placed with respect to the remote focal plane. The output plane of the PhaseD2NN is imaged to the camera sensor using the second OBJ_R and downstream lenses f_1, f_2, and f_3. All lenses are placed at 4f configurations.

Consider the electric field $x_{in} = A_{in}e^{j\phi_{in}}$ at the image plane of the microscope. x_{in} propagates through the optical processor $H_O(.)$ such that,

$$I = D(|H_O(x_{in})|^2) \tag{1}$$

$$\hat{\phi} = H_E(I) \tag{2}$$

Here $D(.)$ represents the low-resolution detection of the output light field. I represents the detected intensity map. $H_E(.)$ represents the neural network that reconstructs the phase map, $\hat{\phi}$, of x_{in} at its original resolution.

Parameters of both the optical processor and the neural network are optimized using machine learning methods. Specifically, we first parameterize the entire end-to-end model in a differentiable manner. The parameters are then optimized by reducing a loss function, that represents the reconstruction quality (see section 5.3). Equation (3) shows the simplified representation of the overall problem.

$$H_O^*, H_E^* = \underset{H_O,H_E}{\arg\min} \quad \mathcal{L}(\hat{\phi}, \phi_{in}) \tag{3}$$

Here $\mathcal{L}$ represents the composite loss function. Components of $\mathcal{L}$ are explained in the methods section (section 5.3). Motivated by our previous work on all-optical phase retrieval [30], for $H_O(.)$, we consider two types of optical processors, Learnable Fourier Filters (Fig. 1.B1) and PhaseD2NNs (Fig. 1.B2). These models are discussed in detail in the methods section 5.2.

The main limitation of our previous work [30] was the lack of *non-linearity* of the optical processor. Phase retrieval is a non-linear image translation problem, and our linear optical processor could only find an approximation. In contrast, here we use the optical processor as *a feature extractor*. The optical model only has to learn a faithful representation that contains sufficient information to computationally reconstruct the original phase map. However, when compressive measurements are used, the reconstruction problem is highly ill-posed. Therefore, next, we investigate the capacity of our linear optical processor to encode phase information, sufficient for inverse reconstruction.

2.2. Linear encoding does not degrade compressibility

With respect to the phase of the input field, our ∂-QPM scheme is similar to an autoencoder. The optical processor acts as the encoder; the neural network acts as the decoder; the output field of the optical processor acts as the bottleneck. Conventional autoencoders [32] have non-linear encoders that can learn compressed representations at their bottleneck. But here our encoder, i.e., the optical processor, is a linear system. We therefore first established the feasibility of linear compression in comparison to nonlinear compressor models.

Linear Encoding and Non-linear Decoding Allow Compression. First, we experimented on an autoencoder (AE) network with a *linear encoder* followed by a *non-linear decoder*. The reconstruction results obtained from this network were compared with a fully linear autoencoder and a fully nonlinear autoencoder. The qualitative results for the MNIST dataset [33] in Fig. 2 show that the autoencoder network with a *linear encoder* and *non-linear decoder* performs on par with the fully nonlinear autoencoder.

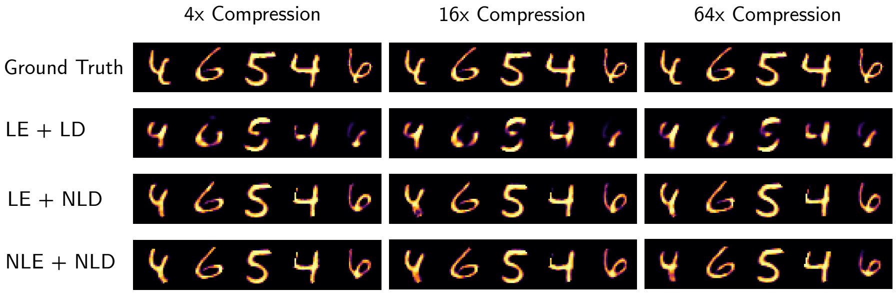

Fig. 2. Compressibility of MNIST images using autoencoders (AE) with linear (L) and nonlinear (NL) encoder (E)/ decoder (D). LE, LD, NLE, and NLD represent linear encoder, linear decoder, non-linear encoder, and non-linear decoder respectively.

Complex-valued Linear Encoding and Non-linear Decoding Allow Compression of Phase Information. While we assessed the feasibility of a linear encoder followed by a non-linear decoder, in QPM, another main hurdle is that information of interest is in the phase of the light field. Therefore, we further assessed the ability of an autoencoder network (complex-valued linear encoder + non-linear decoder, see section 5.1) to extract, compress, and reconstruct information encoded in the phase. Similar to previous results, for the PhaseMNIST dataset (see section 5.4), Fig. 3 shows that a complex-valued linear encoder with a nonlinear decoder achieves similar qualitative performance as the complex-valued nonlinear encoder and nonlinear decoder.

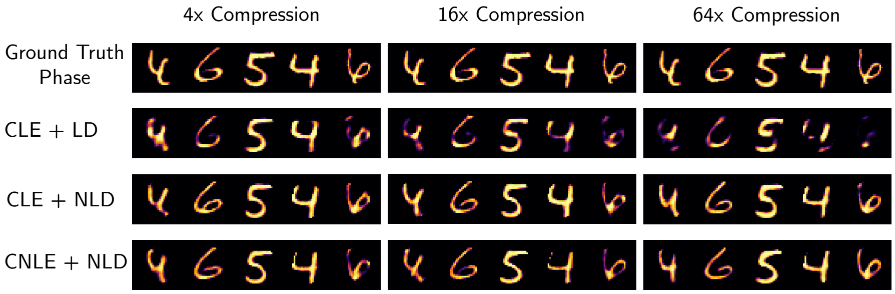

Fig. 3. Phase to intensity conversion and compressibility on PhaseMNIST dataset using linear (L) and nonlinear (NL) encoder (E)/ decoder (D). Both the encoders are complex-valued hence denoted as CLE and CNLE.

These results suggest the feasibility of a linear optical processor (encoder) followed by a nonlinear neural network (decoder) to compress and reconstruct information in the phase of the light field.

2.3. Optical encoding and electronic decoding enable compressed QPM

Our results in section 2.2 show that an autoencoder with a linear encoder and a non-linear decoder (*AE:LE+NLD*) can reconstruct images as well as a fully nonlinear model. In this section, we numerically test our ∂-QPM scheme with two types of optical processors ($H_O(.)$ in Eq. (3)) as encoders. For the decoder neural network ($H_E(.)$ in Eq. (3)), we use a state-of-the-art super-resolution model, SwinIR [34]. We evaluate ∂-QPM on an experimentally collected HeLa cell dataset (see section 5.4).

Learnable Fourier Filter (LFF) + SwinIR. Based on previous work [30], we first used a Learnable Fourier Filter (an LFF) as the optical processor. The LFF contained an optical 4-f system with a learnable circular Fourier filter. Similar to previous work [30], the transmission coefficients of the circular Fourier filter were treated to be learnable. The input and output fields were 256×256 squared aperture grids. The circular Fourier filter had a radius of 128 grid points. The coefficients of the filter were randomly initialized. We used SwinIR [34], a state-of-the-art super-resolution network, as the decoder neural network. We observed that directly training the end-to-end model (optical processor and SwinIR) was not ideal as the gradient flow between the the two models was weak. Therefore, we employed the 3-stage criteria for the optimization of the end-to-end model (as discussed in section 5.3). We tested compression levels ×64 and ×256 for the compressed optical output intensity field in our experiments.

Table 1 shows the performances at ×64, ×256 compression levels for the tested HeLa dataset (section 5.4). For each compression level, performances are reported with and without the fine-tuning step. The corresponding qualitative results are shown in Figs. 4 and 5. All proposed methods outperformed all-optical phase to intensity conversion baselines (B1, B2) [30] with a significant margin in terms of SSIM (structural similarity index) and PSNR (peak signal-to-noise ratio) [35]. Note that all-optical baselines use only the optical processor, and the output intensity measured by the camera sensor is considered the final output phase map. Also, here, the output is at the same resolution as the original input, thereby employing no compression. End-to-end fine-tuning showed a considerable improvement in the performance for all the cases. Our best method achieved PSNR= 29.76 dB, SSIM= 0.90 performance at ×64 compression, indicating that the proposed method is suitable for high-throughput QPM. Even at ×256 compression, the proposed method outperformed all-optical baselines by a considerable margin with PSNR= 27.61 dB and SSIM= 0.83.

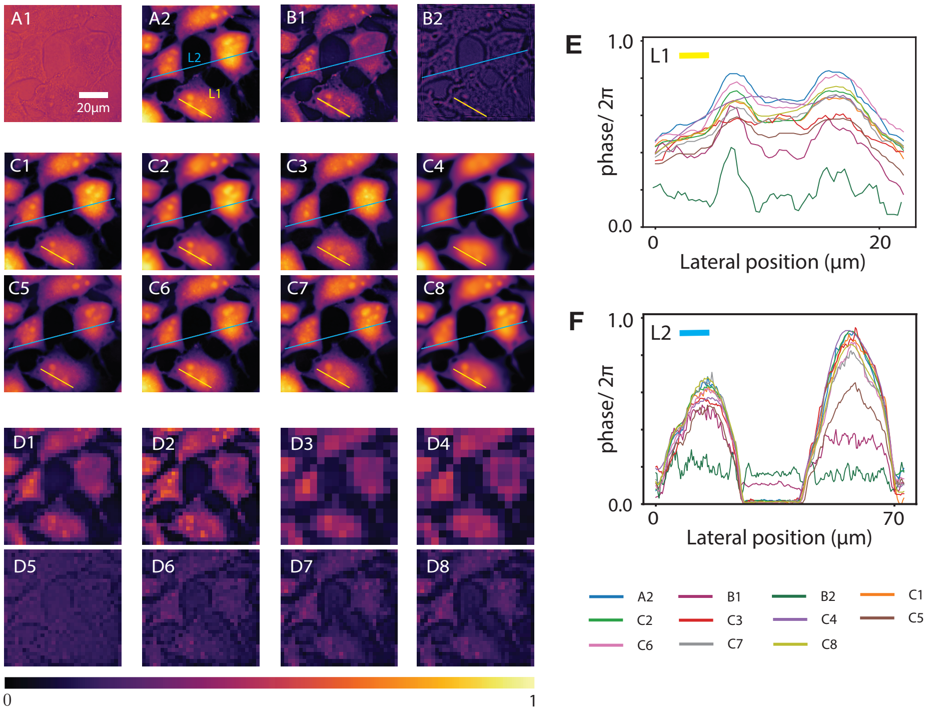

Fig. 4. Qualitative performance comparison of the proposed end-to-end ∂-QPM to all-optical methods: Amplitude of input field (A1), phase of the input field (A2) (i.e. patch FoV) from the test set, all-optical results using LFF (B1), PhaseD2NN (B2), Phase reconstructions from our approach 1: LFF + SwinIR with ×64 compression without fine-tuning (C1), with fine-tuning (C2), LFF with ×256 compression without fine-tuning (C3), with fine-tuning (C4), Phase reconstructions from our approach 2: PhaseD2NN + SwinIR with ×64 compression without fine-tuning + 1 optical layer (C5), without fine-tuning + 3 optical layers (C6), without fine-tuning + 5 optical layers (C7), with fine-tuning + 5 optical layers (C8), Corresponding compressed output intensity fields of optical feature extractor (D1-8). Phase values along the L1 and L2 lines show the local and global resolving power of the proposed methods (E, F).

We further tested our approach by including a noise model with Poisson noise and read noise [31]. We fine-tuned the best model (C2) with noise. A read noise with a standard deviation of 6.0 and a detector maximum photon count of 10000 were used. The proposed method with detector noise (E1) performed on par with the best model indicating that our LFF + SwinIR based ∂-QPM is robust to real-world noise conditions. We further discuss the effect of the detector noise in the discussion (see section 3.).

PhaseD2NN + SwinIR. Second, we tested a PhaseD2NN [30] as the optical processor in the proposed end-to-end framework. Similar to the previous section, the SwinIR super-resolution network was used for reconstruction. We selected the operating wavelength (λ = 632.8 nm) of the PhaseD2NN in the visible wavelengths, and it was evaluated on the same HeLa cell dataset (see section 5.4).

Since the pixel size matched the PhaseD2NN neuron size (316.4 nm × 316.4 nm), we could train the end-to-end network directly on the patch FoVs from the dataset. We followed the optimization criteria presented in section 5.3 for the end-to-end training. Notably, we observed that in step

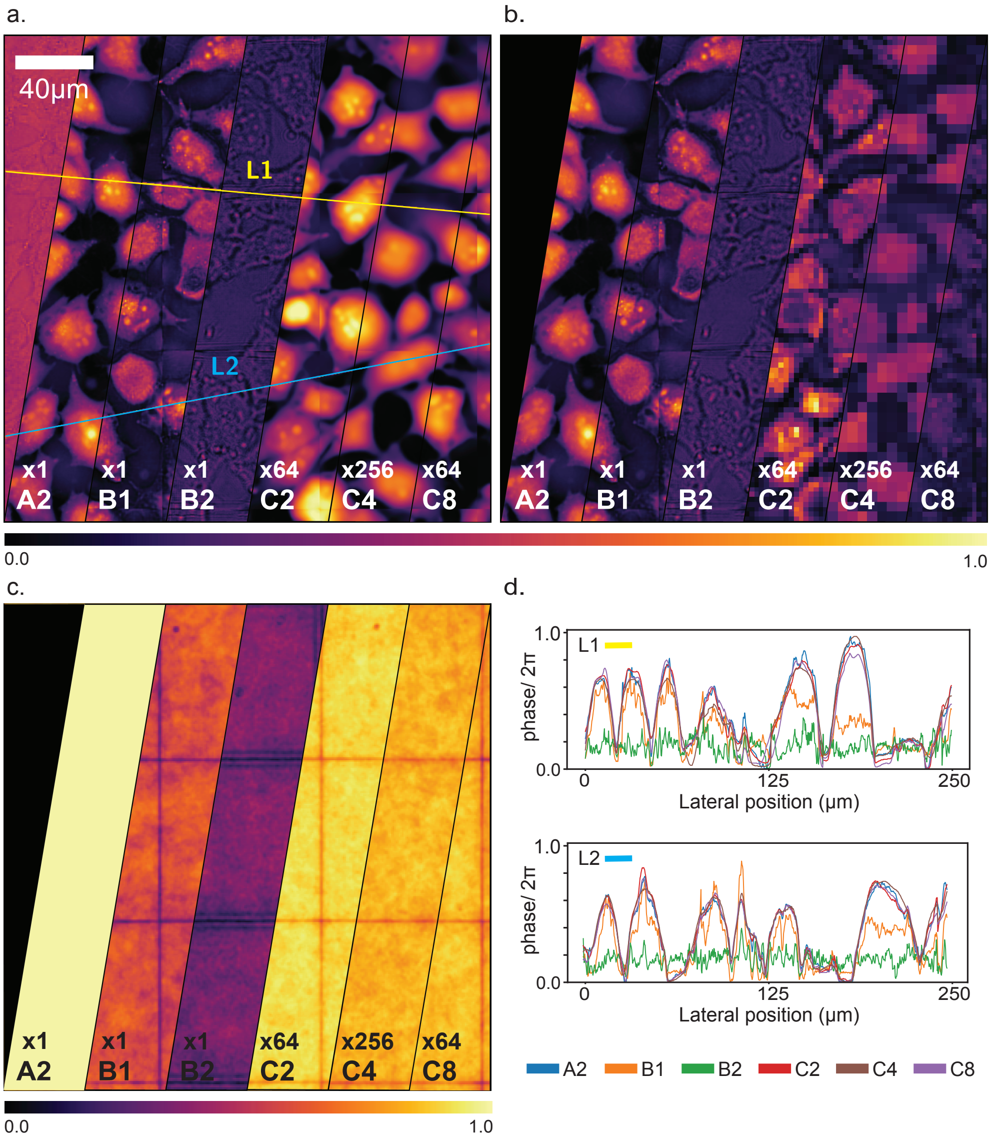

Fig. 5. Performance comparison of best methods using side-by-side comparisons of phase reconstructions (a), compressed intensity fields (b), and SSIM maps of reconstructions (c). Subfigure (d) shows the resolving power of the phase reconstructions. The compared segments: Groundtruth phase of the input field (A2) of a full FoV from the test set, all-optical results using LFF (B1) and PhaseD2NN (B2); Phase reconstructions from our approach 1 – LFF + SwinIR with ×64 compression with fine-tuning (C2), LFF with ×256 compression with fine-tuning (C4); and phase reconstructions from our approach 2 – PhaseD2NN + SwinIR with ×64 compression with fine-tuning + 5 optical layers (C8).

Table 1. Performance comparison: Best results for optical feature extraction networks LFF, PhaseD2NN are highlighted. These best models are further fine-tuned end-to-end with the detector noise simulation (noise specifications of the detector: read noise standard deviation= 6.0, maximum photon count= 10000) to improve realisticity. We calculate the patch and full FoV metrics on the test patch FoVs and full FoVs respectively. We reconstruct the full FoVs by tiling the reconstructed patch FoVs.

Experiment	Detector Noise	Optical Proc.	Compression	Finetune	# layers	full FoV Metrics		patch FoV Metrics	
						PSNR (dB)	**SSIM**	**PSNR (dB)**	**SSIM**
B1	All-optical LFF					16.1565	0.5880	16.9761	0.6008
B2	All-optical PhaseD2NN					12.3730	0.3163	12.6631	0.3320
C1	✗	LFF	64	✗	-	23.8267	0.8225	25.7840	0.8278
C2				✓	-	**27.2579**	**0.8967**	**29.7608**	**0.9031**
C3			256	✗	-	22.5457	0.7470	23.9536	0.7548
C4				✓	-	26.0003	0.8223	27.6129	0.8302
C5	✗	Phase D2NN	64	✗	1	22.6495	0.7808	23.8566	0.7889
C6					3	24.7560	0.8224	26.0716	0.8313
C7					5	24.8015	0.8107	26.0551	0.8185
C8				✓	5	**25.8617**	**0.8519**	**27.2449**	**0.8602**
E1	✓	LFF	64	✓	-	27.3794	0.8935	29.8110	0.8998
E2	✓	Phase D2NN	64	✓	5	25.7665	0.8477	27.0651	0.8558

1, PhaseD2NN training was not stable due to the large number of physical parameters with a larger grid size (e.g., 256 × 256). To increase the stability and gradient flow of this optimization step, we used a sub-optimization-schedule (shown in Supplement 1). We compressed the output intensity from the optical processor ×64 to obtain a higher throughput.

Table 1 shows the performances for ×64 compression level. We report the performances while selecting different layers of the PhaseD2NN as the output layer. The final model with 5 layers was fine-tuned according to the proposed optimization steps. Similar to section 2.3, fine-tuning improved the performance. We explored different numbers of diffractive layers for the PhaseD2NN without the fine-tuning step and the results are presented in Table 1.

We performed further experiments with the 5 layer PhaseD2NN (C8 and E2). Our method achieved the best performance of PSNR= 27.24 dB, SSIM= 0.86 with ×64 compression which was considerably higher than the all-optical baselines. Similar to the previous section, we tested our model for detector noise with similar specifications (of a maximum photon count of 10000 and detector read noise standard deviation of 6.0). The resultant performance with the detector noise (E2) was on par with the best model without the noise (C8). This indicates that our PhaseD2NN + SwinIR based ∂-QPM is robust to real-world noise conditions.

3. Discussion

Overall Comparison. Fig. 5 presents the qualitative results for best-performing models. Figure 5(d) shows that the proposed ∂-QPM systems have a higher resolving capability compared to the all-optical baselines. Figure 5(c) SSIM maps show how our methods perform for different regions of full field-of-view (FoV). Low SSIM in edges indicates that there is room to improve the proposed QPM just by refining the edges of generated patches. We also observed that the

LFF-based method outperformed the PhaseD2NN-based one. Further studies are needed to investigate the reason for this behavior.

Stability of PhaseD2NN Training. We observed that the optimization step 1, i.e., all-optical reconstruction (see section 5.3), was unstable for the PhaseD2NN. We suspect that the reason for this instability is the large FoV (of 256×256) resulting in a large number of learnable parameters. To overcome this, we used a sub-optimization-schedule for the PhaseD2NN training motivated by progressive growing learning principles [36] (see algorithm S1 in the Supplement 1). Instead of training the PhaseD2NN in an end-to-end fashion, here we optimize the PhaseD2NN layer by layer progressively with the phase reconstruction loss. With this schedule, we could efficiently train the optical processor. Even though one can argue that the proposed schedule leads to a sub-optimal solution, we achieved a sufficient performance for QPM [25] with this schedule. Nevertheless, an interesting future direction is to explore more efficient methods to train large D2NNs.

Effect of Photodetector Noise. To further evaluate the behavior of the proposed method with detector noise, we evaluated the method with maximum photon counts of 100 and 10000, and read noise standard deviations of 4.0 and 6.0. Changing the photon counts changes the Poisson noise in detection. Table 2 shows that ∂-QPM is robust to noise when the maximum photon count is 10000 (for most QPM applications such high light conditions can be used). We saw a significant reduction in performance when the maximum photon count was 100, i.e. at high Poisson noise conditions. Interestingly, the effect on the D2NN-based model was more severe than that of the LFF-based model. Thus an interesting future direction is to investigate better noise-aware training strategies for optical processors. Last, we did not see a significant effect from read noise.

Table 2. Performance of our method for different detector noise conditions: Our best models (C2, C8 in Table 1) are further fine-tuned with the corresponding noise specifications.

Optical Net	Noise Specifications		full FoV metrics		patch FoV metrics	
	max. photon count	σ_{read}	**PSNR (dB)**	**SSIM**	**PSNR (dB)**	**SSIM**
LFF	100	4	23.4928	0.7770	25.0266	0.7834
		6	22.5004	0.7606	24.1487	0.7663
	10000	4	**27.4122**	**0.8935**	**29.8110**	**0.8997**
		6	**27.3794**	**0.8935**	**29.8110**	**0.8998**
Phase D2NN	100	4	17.8942	0.6441	18.7526	0.6502
		6	16.8953	0.6094	17.7111	0.6158
	10000	4	**25.7193**	**0.8478**	**27.0456**	**0.8559**
		6	**25.7665**	**0.8477**	**27.0651**	**0.8558**

Compressibility limitations. Last, we tested our LFF-based approach on a QPM dataset of tissue with much more complex features (see section 5.4). The goal of this experiment was to investigate the limitations of our approach for complex features. We observed that our method failed to reconstruct high-resolution features at both ×64 and ×256 (see Supplement 1). There could be two potential reasons for the subpar performance. It could be the case that the optical processor cannot efficiently convert phase information to the latent intensity field at the detector. Alternatively, it could be the case that the reconstruction network is not capable of reconstructing highly compressed information from images with complex features. To investigate the latter we tested our reconstruction network on a simple resolution enhancement task on the same tissue dataset. As shown in Supplement 1, here too the reconstruction network failed. Thus we conclude that in our method, the compressibility is limited in the presence

Biomedical Optics EXPRESS

of complex features. Further studies are required to establish the compressibility bounds for data distributions of interest. Nevertheless, our successful demonstration on cell data opens doors to a number of applications in cell biology and medicine such as: pathogen screening [8]; stain-free quantification of chromosomal dry mass in living cells [37]; quantification of different growth phases of chondrocytes [38]; and identification of biophysical markers of sickle cell drug responses [39].

Realization of the Optical Processors. In this work, we only consider numerical simulations of optical processors. We now discuss the feasibility of their realization. The LFF setup can be implemented as an optical 4-f system with a transmissive spatial light modulator (SLM) (as shown in Fig. 1.B1), or with a reflective SLM (as experimentally demonstrated in our previous work [30]). The PhaseD2NN simulated in this work consists of submicron-sized "optical neurons" distributed in 3D in a micron-sized optical element. Fabricating such custom-designed 3D optics to a desired specification is extremely challenging. Nevertheless, D2NNs have been experimentally demonstrated at Terahertz wavelengths, and translating these models to visible wavelengths is an active area of research. For instance, two-photon lithography [40,41] is a promising avenue to fabricate D2NNs in 3D. Required fabrication precision may also be relaxed by incorporating the details about the fabrication imperfections during the design stage itself [42]. We leave the robustness improvement, realization, and experimental validation of the optical processors to future work.

4. Conclusion

Quantitative phase microscopy (QPM) is an emerging label-free imaging modality with a wide range of biological and clinical applications. Recent advances in QPM are focused on developing fast instruments through better detectors and fast deep-learning-based inverse solvers. However, currently, the QPM throughput is fundamentally limited by the pixel throughput of the imaging detectors. Orthogonal to current advances, to improve QPM throughput beyond the hardware bottleneck, here we propose to use content-aware compressive data acquisition. Specifically, we utilize learnable optical processors to extract compressed phase features. A state-of-the-art transformer deep network then decodes the captured information to quantitatively reconstruct the phase image. The proposed pipeline inherently improves the imaging speed while achieving high-quality reconstructions. Moreover, the advances presented in this work can lead to similar developments in a wide range of label-free coherent imaging modalities such as photothermal, coherent anti-Stokes Raman scattering (CARS), and stimulated Raman scattering (SRS).

5. Methods

5.1. Networks for linear compression feasibility studies

To establish the feasibility of using a linear system to compress an image/ optical field, we conducted the analysis presented in section 2.2. For the analysis, we implemented simple autoencoder networks with linear/ nonlinear decoders and linear/ nonlinear/ complex linear/ complex nonlinear encoders. All the autoencoders we discussed have the following general format.

$$h = Enc(y) \tag{4}$$

$$\hat{y} = Dec(h) \tag{5}$$

Here, y is the real/ complex input image (depending on the experiment), h is the latent code, $\hat{y}$ is the reconstructed image. $Enc(.)$ and $Dec(.)$ are the functions to encode the input and decode the latent code.

To train real-valued autoencoders, we considered the following objective function.

$$Enc^*, Dec^* = \underset{Enc,Dec}{\arg\min} \quad \mathbb{E}[\mathcal{L}(\hat{y}, y)] \tag{6}$$

Enc^* and Dec^* are the optimal encoders and decoders found through Adam optimization [43]. $\mathcal{L}$ shows the mean squared distance. $\mathbb{E}[.]$ denotes the expected value over the dataset.

Complex-valued autoencoders were trained with the following objective function.

$$Enc^*, Dec^* = \underset{Enc,Dec}{\arg\min} \quad \mathbb{E}[\mathcal{L}(\hat{y}, \angle y)] \tag{7}$$

Notably, here we consider $\angle y$ (input phase) as the ground truth. The goal is to extract information from the input phase and reconstruct it in the output (refer to section 2.2, 5.4).

The compression factor shown in Fig. 2 and 3 is defined as the ratio between the total number of pixels in y and h. We implemented the encoders using convolution layers, each having kernel size, padding, and stride to obtain ×2 downscale. Decoders are implemented by cascading transpose convolution, ReLU activations, and batch normalization layers. Complex-valued autoencoders allow complex values in the inputs, y.

5.2. Optical processors

We consider two types of optical processors based on previous work [30]: Learnable Fourier Filter and PhaseD2NN. This section gives a brief description of these optical processors and the mathematical modeling of light propagation through them.

Learnable Fourier Filter. The LFF is an optical 4-f system with a filter placed in the Fourier plane. The transmission coefficients of this filter are optimized during the optimization process. The overall system is modeled using the following equation.

$$x_{out} = \mathcal{F}^{-1}\left\{T \circ \mathcal{F}\left\{x_{in}\right\}\right\} \tag{8}$$

In this, x_{in} is the input light field coming to the LFF, x_{out} is the output light field, and T is the Fourier filter. $\mathcal{F}$, and $\mathcal{F}^{-1}$ denote the Fourier transform and the inverse Fourier transform where $\circ$ denote the Hadamard product.

PhaseD2NN. PhaseD2NN is a diffractive deep neural network [44] with only the phase of the transmission coefficients being optimized. The amplitude of the transmission coefficients is set to 1 in each layer. We modeled light propagation through D2NN using the Rayleigh-Sommerfeld diffraction theory [45, ch. 3.5]. After light propagates through a D2NN layer, the input field to the next layer is given by

$$x_{in}^{(n)} = \mathcal{RS}\left(x_{in}^{(n-1)} \circ T^{(n-1)}, \Delta z^{(n-1)}\right). \tag{9}$$

Here, $x_{in}^{(p)}$ denotes the input field to the p^{th} layer, $T^{(p)}$ is the complex transmission coefficient matrix of the p^{th} layer, and $\Delta z^{(p)}$ is the distance between the p^{th} and the $(p+1)^{\text{th}}$ layers. $\mathcal{RS}(.)$ denotes the Rayleigh-Sommerfeld diffraction operator. The output field of the PhaseD2NN is given by

$$x_{out} = x_{out}^{(M+1)} = \mathcal{RS}\left(x_{in}^{(M)} \circ T^{(M)}, \Delta z^{(M)}\right), \tag{10}$$

where M is the number of layers in the PhaseD2NN.

The PhaseD2NN simulated in this work consisted of 5 diffractive layers each having 256 × 256 optical neural grid. The size of each neuron was $\frac{\lambda}{2} \times \frac{\lambda}{2}$ (316.4 nm ×316.4 nm). Therefore, the size of the optical layer was 80.9984μm × 80.9984μm. Optical layers were separated with 3.373μm distance between each other. The distance between the input plane and the first optical layer was 3.373μm while the distance between the last optical layer and the detector plane was 5.904μm.

For both LFF and PhaseD2NN, the final intensity captured by the detector is given by

$$I = D\left(|x_{out}|^2\right) \tag{11}$$

where $D(.)$ denotes the low-resolution detection. We use the LFF and PhaseD2NN as all-optical baselines for result comparison. In that, they are detected at the original resolution of the input optical field following the Nyquist sampling theorem (i.e., without the $D(.)$ operator). All-optical baselines are optimized in such a way that $|x_{out}|^2$ gives an approximation of the input phase.

5.3. Optimization details

We follow a 3-stage optimization criteria for the improved stability of the end-to-end optimization; 1) optimize the optical processor; 2) optimize the decoder neural network; 3) end-to-end fine-tuning.

Optimize the optical processor. Here the optical processor is optimized to reconstruct the phase at its output intensity. For an input optical field $x_{in} = A_{in}e^{j\phi_{in}}$ we train an optical model H_O through which the input field is propagated to produce the output field $x_{out} = A_{out}e^{j\phi_{out}} = H_O(x_{in})$. The **phase reconstruction loss**, $\mathcal{L}_\phi$ introduced in previous work [30] is utilized here as,

$$\mathcal{L}_\phi = \mathbb{E}_{x_{in}\sim P_X}\left[L1(|A_{out}|^2, \phi_{in}/(2\pi))\right], \tag{12}$$

where, P_X and $L1(.)$ respectively represent the probability distribution of phase objects and the L1 loss.

Optimize the decoder neural network. At this stage, we consider the end-to-end network, however, only the weights of the neural network are optimized. The pretrained optical processor discussed in the previous step is utilized to encode the input phase. We demagnify the output field of the optical processor to compress the intensity representation. The super-resolution neural network reconstructs the input phase from the compressed intensity representation. The reconstructed phase information is given by $\hat{\phi} = H_E(D(|A_{out}|^2))$. Here $H_E(.)$ represents the decoder neural network. $D(.)$ is the optical demagnification layer, which is simulated through a stack of 2 × 2 average pooling operations [46]. Similar to previous work [34], we consider $\mathcal{L}_{swin}$, a combination of loss functions for this optimization,

$$\mathcal{L}_{swin} = \mathbb{E}_{x_{in}\sim P_X}\left[L1(\hat{\phi}, \phi_{in}/(2\pi)) + \mathcal{L}_{perceptual}(\hat{\phi}, \phi_{in}/(2\pi)) + \mathcal{L}_{adversarial}(\hat{\phi}, \phi_{in}/(2\pi))\right], \tag{13}$$

where, $\mathcal{L}_{perceptual}$ and $\mathcal{L}_{adversarial}$ represent the perceptual loss [34] and adversarial loss [34] respectively.

End-to-end fine-tuning. As the final stage, we finetune the end-to-end ∂-QPM pipeline to reconstruct the phase at the output of the network. To improve the reconstruction in terms of capturing fine cell structures, we incorporate the negative structural similarity index measure (SSIM) [47] as the loss function.

$$\mathcal{L}_{SSIM} = \mathbb{E}_{x_{in}\sim P_X} - \frac{1}{M}\sum_{j=1}^{M}\frac{(2\mu_{X_j}\mu_{Y_j} + C_1)(2\sigma_{X_jY_j} + C_2)}{\left(\mu_{X_j}^2 + \mu_{Y_j}^2 + C_1\right)\left(\sigma_{X_j}^2 + \sigma_{Y_j}^2 + C_2\right)}. \tag{14}$$

Here, X_j and Y_j represent equal-sized windows from a normalized input phase image ($\phi_{in}/2\pi$) and the corresponding reconstructed phase output ($\hat{\phi}$) respectively, with M number of windows for an image. P_X represents the probability distribution of input phase objects. $\mu_{X_j}, \mu_{Y_j}, \sigma_{X_j}, \sigma_{Y_j}, \sigma_{X_jY_j}$ are the means, variances, and the covariance of the X_j and Y_j windows respectively. $C_1 = (k_1 \times L)^2$ and $C_2 = (k_2 \times L)^2$ are regularization parameters with $L = 1.0$, $k_1 = 0.01$ and $k_2 = 0.03$.

5.4. Datasets

In our numerical experiments, we used three datasets.

PhaseMNIST Dataset. We developed *PhaseMNIST*, complex valued dataset for the evaluations in section 2.2. Each complex image of the dataset was obtained according to the Eq. (15).

$$I = 1.e^{j\pi\psi} \tag{15}$$

Here, I is the complex image, ψ is the images from the original MNIST dataset [33] scaled into $[0, 1]$.

HeLa Cell Dataset. We used a HeLa cell dataset [30] as the primary dataset for our experiments. We followed the sample preparation procedure explained in previous work [30]. Briefly, the data were acquired using a low spatially coherent quantitative phase microscopy system. The details of the experimental setup can be found in [48]. First, multiple phase shifted interferograms are recorded of both HeLa cells and tissue samples. The phase recovery is then performed by employing advanced iterative algorithm (AIA), which can retrieve phase maps using random phase-shifted interferograms. The details of the algorithm can be found in [49]. The initial dataset contained 501 complex-valued images (i.e. detected FoVs). Each detected FoV was obtained by a camera with a 2304×2304 pixel grid where the pixel size was 6.5 μm × 6.5 μm. The light field from the specimen was magnified ×60 before imaging onto the detector.

To pre-process the dataset, we first calculated the side length of the light fields before the magnification $\left(= \frac{2304\,\text{pixels} \times 6.5\,\mu\text{m/pixel}}{60} = 249.6\,\mu\text{m}\right)$. Second, we calculated the number of 316.4 nm ×316.4 nm sized pixels in these light fields $\left(= \text{round}\left(\frac{249.6\,\mu\text{m}}{316.4\,\text{nm/pixel}}\right) = 789\,\text{pixels}\right)$. Finally, we resized the detected FoVs (i.e. 2304×2304 pixel grids) into 789×789 pixel grids. This resulted in the light field before the magnification with a pixel size of 316.4 nm ×316.4 nm. We refer to these FoVs as full FoVs. We obtained train and test sets by dividing the full FoV dataset into 401 and 100 sets. For the training of the proposed networks, we used 256×256 cropped patches (i.e. patch FoVs) from the full FoVs. Phase values of the dataset were clipped into $[0, 2\pi)$.

Tissue Dataset. We also acquired a tissue dataset to investigate the limitations of our method further. We utilized a 4-micron thick tissue sample which was prepared on a reflecting substrate (si-wafer in our case). The sample was illuminated from above by a light beam, traverses through it, and is subsequently reflected off the Si substrate. We followed acquisition and preprocessing procedures similar to HeLa cells, with a magnification of ×20. There were 470 detected FoVs. Camera had 2304×2304 pixel grid where the pixel size was 6.5 μm × 6.5 μm. The side length of the light fields before the magnification was $\frac{2304\,\text{pixels} \times 6.5\,\mu\text{m/pixel}}{20} = 748.8\,\mu\text{m}$. Number of 316.4 nm ×316.4 nm sized pixels in these light field was round $\left(\frac{748.8\,\mu\text{m}}{316.4\text{nm/pixel}}\right) = 2367$pixels. We resized the detected FoVs (i.e. 2304×2304 pixel grids) into 2367×2367 pixel grids to match the pixel sizes of the light fields and the algorithm (full FoVs). The full FoV dataset was divided into 470 train FoVs and 117 test FoVs. Phase values of the dataset were clipped into $[0, 2\pi)$.

5.5. Simulation details

We numerically simulated and trained the proposed ∂-QPM pipeline using Python version 3.6.13. The simulation was done according to Eq. (8), (9), and (10). We used auto differentiation in PyTorch [50] framework (version 1.8.0) for the joint optimization/ training of the proposed pipeline. All experiments were conducted on a server with 12 Intel Xeon Platinum 8358 (2.60 GHz) CPU Cores and an NVIDIA A100 Graphics Processing Unit with 40 GB memory running on the CentOS operating system.

We used a batch size of 32, and learning rates of 0.1, 0.001 respectively for LFF and PhaseD2NN in the optimization stage 1. LFF was trained for 1500 epochs with multi-step learning rate scheduler [50] (milestones : [50, 400, 650, 1000, 1400], $\gamma = 0.1$). PhaseD2NN was trained for 1500 epochs after each optimizer initialization step in algorithm S1. For joint multi-layer optimizations in algorithm S1, a learning rate of 0.00005 was used for better stability. For optimization stage 2, we followed the same training configurations used in SwinIR section 4.1, 4.3 real-world image SR, with channel size of 1 and pixel-shuffle upsampling [34]. Lastly, for the final optimization stage (i.e. end-to-end fine-tuning), we fine-tuned the LFF + SwinIR and PhaseD2NN + SwinIR for 24000 and 3000 epochs respectively with a learning rate of 5×10^{-6}. We used Adam [43] as the optimizer for all optimizations.

Funding. National Institute of Mental Health (R21-MH130067); The Research Council of Norway.

Acknowledgments. This work was supported by the Center for Advanced Imaging at Harvard University (D.N.W., U.H., K.H., R.H., and H.K.), 1-R21-MH130067-01 (U.H., D.N.W). D.N.W. is also supported by the John Harvard Distinguished Science Fellowship Program within the FAS Division of Science of Harvard University.

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

Data availability. Data underlying the results presented in this paper can be obtained from the authors upon reasonable request. All codes developed in this work are available in the code repository at [51].

Supplemental document. See Supplement 1 for supporting content.

References

1. G. Popescu, T. Ikeda, R. R. Dasari, *et al.*, “Diffraction phase microscopy for quantifying cell structure and dynamics,” Opt. Lett. **31**(6), 775–777 (2006).
2. Y. Park, G. Popescu, K. Badizadegan, *et al.*, “Diffraction phase and fluorescence microscopy,” Opt. Express **14**(18), 8263–8268 (2006).
3. C. Fang-Yen, S. Oh, Y. Park, *et al.*, “Imaging voltage-dependent cell motions with heterodyne Mach-Zehnder phase microscopy,” Opt. Lett. **32**(11), 1572–1574 (2007).
4. M. S. Amin, Y. Park, N. Lue, *et al.*, “Microrheology of red blood cell membranes using dynamic scattering microscopy,” Opt. Express **15**(25), 17001–17009 (2007).
5. Y. Sung, N. Lue, B. Hamza, *et al.*, “Three-dimensional holographic refractive-index measurement of continuously flowing cells in a microfluidic channel,” Phys. Rev. Appl. **1**(1), 014002 (2014).
6. W. Choi, C. Fang-Yen, K. Badizadegan, *et al.*, “Tomographic phase microscopy,” Nat. Methods **4**(9), 717–719 (2007).
7. Y. K. Park, C. Depeursinge, and G. Popescu, “Quantitative phase imaging in biomedicine,” Nat. Photonics **12**(10), 578–589 (2018).
8. Y. Jo, S. Park, J. Jung, *et al.*, “Holographic deep learning for rapid optical screening of anthrax spores,” Sci. Adv. **3**(8), e1700606 (2017).
9. D. Roitshtain, L. Wolbromsky, E. Bal, *et al.*, “Quantitative phase microscopy spatial signatures of cancer cells,” Cytometry Pt A **91**(5), 482–493 (2017).
10. H. Majeed, A. Keikhosravi, M. Kandel, *et al.*, “Quantitative histopathology of stained tissues using color spatial light interference microscopy (cSLIM),” Sci. Rep. **9**(1), 14679 (2019).
11. Y. Rivenson, T. Liu, Z. Wei, *et al.*, “PhaseStain: The digital staining of label-free quantitative phase microscopy images using deep learning,” Light: Sci. Appl. **8**(1), 23 (2019).
12. M. Kandel, C. Hu, G. Naseri Kouzehgarani, *et al.*, “Epi-illumination gradient light interference microscopy for imaging opaque structures,” Nat. Commun. **10**(1), 4691 (2019).
13. F. Zernike, “Observation of transparent objects,” Physica pp. 974–986 (1942).
14. J. Glückstad, “Phase contrast image synthesis,” Opt. Commun. **130**(4-6), 225–230 (1996).
15. N. Shibata, S. D. Findlay, Y. Kohno, *et al.*, “Differential phase-contrast microscopy at atomic resolution,” Nat. Phys. **8**(8), 611–615 (2012).
16. Y. J. Jo, H. Cho, S. Y. Lee, *et al.*, “Quantitative phase imaging and artificial intelligence: A review,” IEEE J. Sel. Top. Quantum Electron. **25**(1), 1–14 (2019).
17. K. Kim, K. S. Kim, H. Park, *et al.*, “Real-time visualization of 3-d dynamic microscopic objects using optical diffraction tomography,” Opt. Express **21**(26), 32269–32278 (2013).
18. J. Lim, K. Lee, K. H. Jin, *et al.*, “Comparative study of iterative reconstruction algorithms for missing cone problems in optical diffraction tomography,” Opt. Express **23**(13), 16933–16948 (2015).
19. Y. Sung, W. Choi, C. Fang-Yen, *et al.*, “Optical diffraction tomography for high resolution live cell imaging,” Optics InfoBase Conference Papers 17, 1977–1979 (2009).
20. T. Nguyen, V. Bui, and G. Nehmetallah, “Computational optical tomography using 3d deep convolutional neural networks (3d-dcnns),” Opt. Eng. **57**(04), 1 (2018).
21. J. Di, K. Wang, Y. Li, *et al.*, “Deep learning-based holographic reconstruction in digital holography,” in *Imaging and Applied Optics Congress*, (Optica Publishing Group, 2020), p. HTu4B.2.
22. Y. Zhu, C. H. Yeung, and E. Y. Lam, “Digital holographic imaging and classification of microplastics using deep transfer learning,” Appl. Opt. **60**(4), A38–A47 (2021).

23. K. Wang, Q. Kemao, J. Di, *et al.*, “Y4-net: A deep learning solution to one-shot dual-wavelength digital holographic reconstruction,” Opt. Lett. **45**(15), 4220–4223 (2020).
24. H. Wang, M. Lyu, and G. Situ, “eholonet: A learning-based end-to-end approach for in-line digital holographic reconstruction,” Opt. Express **26**(18), 22603–22614 (2018).
25. K. Wang, J. Dou, Q. Kemao, *et al.*, “Y-net: A one-to-two deep learning framework for digital holographic reconstruction,” Opt. Lett. **44**(19), 4765–4768 (2019).
26. M. R. Kellman, E. Bostan, N. A. Repina, *et al.*, “Physics-based learned design: Optimized coded-illumination for quantitative phase imaging,” IEEE Trans. Comput. Imaging **5**(3), 344–353 (2019).
27. A. Matlock and L. Tian, “High-throughput, volumetric quantitative phase imaging with multiplexed intensity diffraction tomography,” Biomed. Opt. Express **10**(12), 6432–6448 (2019).
28. P. Yeh and C. Gu, *Landmark Papers on Photorefractive Nonlinear Optics* (World Scientific, 1995).
29. V. Studer, J. Bobin, M. Chahid, *et al.*, “Compressive fluorescence microscopy for biological and hyperspectral imaging,” Proc. Natl. Acad. Sci. **109**(26), E1679–E1687 (2012).
30. K. Herath, U. Haputhanthri, R. Hettiarachchi, *et al.*, “Differentiable microscopy designs an all optical quantitative phase microscope,” arXiv, arXiv:2203.14944 (2022).
31. U. Haputhanthri, A. Seeber, and D. Wadduwage, “Differentiable microscopy for content and task aware compressive fluorescence imaging,” arXiv, arXiv:2203.14945 (2022).
32. Y. Wang, H. Yao, and S. Zhao, “Auto-encoder based dimensionality reduction,” Neurocomputing **184**, 232–242 (2016). RoLoD: Robust Local Descriptors for Computer Vision 2014.
33. L. Deng, “The mnist database of handwritten digit images for machine learning research,” IEEE Signal Process. Mag. **29**(6), 141–142 (2012).
34. J. Liang, J. Cao, G. Sun, *et al.*, “Swinir: Image restoration using swin transformer,” *2021 IEEE/CVF International Conference on Computer Vision Workshops (ICCVW)* pp. 1833–1844 (2021).
35. A. Horé and D. Ziou, “Image quality metrics: PSNR vs. SSIM,” in *2010 20th International Conference on Pattern Recognition*, (2010), pp. 2366–2369.
36. T. Karras, T. Aila, S. Laine, *et al.*, “Progressive growing of gans for improved quality, stability, and variation,” ArXiv, arXiv:1710.10196 (2018).
37. Y. Sung, W. Choi, N. Lue, *et al.*, “Stain-free quantification of chromosomes in live cells using regularized tomographic phase microscopy,” PLoS One **7**(11), e49502 (2012).
38. Y. Sung, A. Tzur, S. Oh, *et al.*, “Size homeostasis in adherent cells studied by synthetic phase microscopy,” Proc. Natl. Acad. Sci. U. S. A. **110**(1), 1–2 (2013).
39. P. Hosseini, S. Z. Abidi, E. Du, *et al.*, “Cellular normoxic biophysical markers of hydroxyurea treatment in sickle cell disease,” Proc. Natl. Acad. Sci. U.S.A. **113**(34), 9527–9532 (2016).
40. D. Oran, S. G. Rodriques, R. Gao, *et al.*, “3d nanofabrication by volumetric deposition and controlled shrinkage of patterned scaffolds,” Science **362**(6420), 1281–1285 (2018).
41. G. Yang, Q. Yang, C. Zheng, *et al.*, “3D nanofabrication of multi-functional optical/multi-functional metamaterials,” in *Laser 3D Manufacturing X*, vol. PC12412 B. Gu and H. Chen, eds., International Society for Optics and Photonics (SPIE, 2023), p. PC124120N.
42. D. Mengu, Y. Zhao, N. T. Yardimci, *et al.*, “Misalignment resilient diffractive optical networks,” Nanophotonics **9**(13), 4207–4219 (2020).
43. D. Kingma and J. Ba, “Adam: A method for stochastic optimization,” *International Conference on Learning Representations* (2014).
44. X. Lin, Y. Rivenson, N. T. Yardimci, *et al.*, “All-optical machine learning using diffractive deep neural networks,” Science **361**(6406), 1004–1008 (2018).
45. J. W. Goodman and M. E. Cox, “Introduction to Fourier Optics,” Phys. Today **22**(4), 97–101 (1969).
46. H. Gholamalinezhad and H. Khosravi, “Pooling methods in deep neural networks, a review,” ArXiv, arXiv:2009.07485 (2020).
47. Z. Wang, A. Bovik, H. Sheikh, *et al.*, “Image quality assessment: From error visibility to structural similarity,” IEEE Trans. on Image Process. **13**(4), 600–612 (2004).
48. A. Ahmad, N. Jayakumar, and B. S. Ahluwalia, “Demystifying speckle field interference microscopy,” Sci. Rep. **12**(1), 10869 (2022).
49. Z. Wang and B. Han, “Advanced iterative algorithm for phase extraction of randomly phase-shifted interferograms,” Opt. Lett. **29**(14), 1671–1673 (2004).
50. A. Paszke, S. Gross, F. Massa, *et al.*, “Pytorch: An imperative style, high-performance deep learning library,” in *Advances in Neural Information Processing Systems 32*, H. Wallach, H. Larochelle, A. Beygelzimer, F. d’Alché-Buc, E. Fox, and R. Garnett, eds. (Curran Associates, Inc., 2019), pp. 8024–8035.
51. U. Haputhanthri, K. Herath, R. Hettiarachchi, *et al.*, “Towards ultrafast quantitative phase imaging via differentiable microscopy: code,” Github, 2024, https://github.com/wadduwagelab/OpticalElectronicQPI.