

Differentiable Microscopy Designs an All Optical Phase Retrieval Microscope

Kithmini Herath^{1,2}, Udit Haputhanthri^{†1,2}, Ramith Hettiarachchi^{†1,2}, Hasindu Kariyawasam^{†1,2}
 Raja Naeem Ahmad¹, Azeem Ahmad³, Balpreet S. Ahluwalia³
 Chamira U. S. Edussooriya², and Dushan N. Wadduwage^{1,*}

¹Center for Advanced Imaging, Faculty of Arts and Sciences, Harvard University, Cambridge, USA

²Department of Electronic and Telecommunication Engineering, University of Moratuwa, Sri Lanka

³UiT The Arctic University of Norway

*wadduwage@fas.harvard.edu

Abstract

Since the late 16th century, scientists have continuously innovated and developed new microscope types for various applications. Creating a new architecture from the ground up requires substantial scientific expertise and creativity, often spanning years or even decades. In this study, we propose an alternative approach called "Differentiable Microscopy," which introduces a top-down design paradigm for optical microscopes. Using all-optical phase retrieval as an illustrative example, we demonstrate the effectiveness of data-driven microscopy design through $\partial\mu$. Furthermore, we conduct comprehensive comparisons with competing methods, showcasing the consistent superiority of our learned designs across multiple datasets, including biological samples. To substantiate our ideas, we experimentally validate the functionality of one of the learned designs, providing a proof of concept. The proposed differentiable microscopy framework supplements the creative process of designing new optical systems and would perhaps lead to unconventional but better optical designs.

1. Introduction

Even though the invention of microscopes dates back to the late 16th century, it was during the late 19th to 20th century that the field saw a major renaissance [1]. This is especially due to the requirement of cross-domain scientific knowledge and creativity. For instance, as much as the knowledge of physics that is essential to build optical systems, fields such as chemistry (for sample preparation) and computer science (for image post-processing), play an important role in developing novel microscope architectures. Typical microscopy instrumentation involves understanding: the light-matter interaction at the specimen, how light propagates through optics, and how the final image is formed. The scientist's job is to design the optics such that the final image contains desired information about the measurands. This creative process may require enormous fuzzy calculations on the light propagation just to initialize a potential optical design. In this work, we present a data-driven top-down approach to supplement this creative design process.

Data-driven machine learning approaches have been increasingly used in various microscopy methods to perform tasks such as denoising, image reconstruction, and classification [2–4]. Despite the significant performance boost in such methods, the limitations inherited from the hardware of the microscope set the upper performance bound [5]. Therefore, over the past decade, researchers have focused on joint data-driven optimization of not only the reconstruction model but also the hardware of the microscope itself [6–11]. Nevertheless, most of these methods' focus was to optimize the parameters of an existing design. In contrast, the focus of this work is the optical design process itself, rather than data-driven optimization of existing designs. To this end, we introduce Differentiable Microscopy ($\partial\mu$) as a top-down design approach where new designs can be parameterized and

[†] These authors contributed equally.

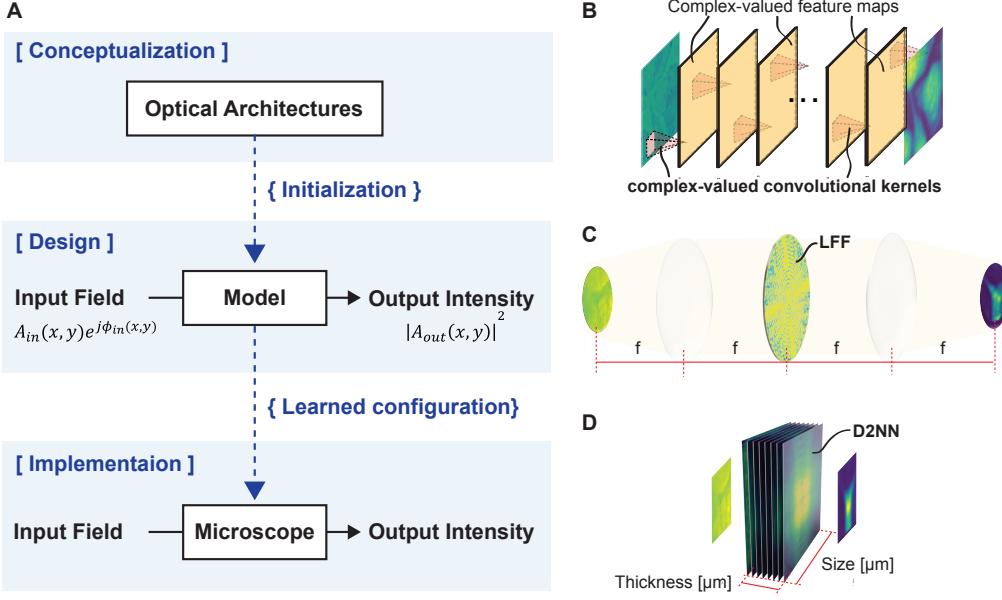


Figure 1. Differentiable microscopy ($\partial\mu$) based all-optical phase to intensity conversion. **(A)** Problem formulation as a learning task and model implementation pipeline in differentiable microscopy. **(B)** The complex-valued linear CNN architecture that was implemented mimicking optical constraints for linear phase retrieval. Note that there are no non-linear activations between convolutions. **(C)** Learnable Fourier filter (LFF) based design. Here, f denotes the focal length of the lens used to implement the 4- f system. **(D)** Diffractive deep neural network based design (D2NN). Note that D2NN is very compact and less than $50\mu\text{m}$ (i.e. 51.7λ , where λ is the operating wavelength) thick.

optimized first as black-box models and then as optical architectures. As a first step, in this paper, we propose new optical designs, differentiably model them and train them for the task of all-optical phase retrieval.

Measuring phase information of a light field is a longstanding problem in optics with valuable applications such as live cell imaging [12–16]. Objects like cells are thin and transparent. They are referred to as phase objects as they modulate the phase of the light field, not the intensity. Imaging devices however only measure the intensity of light. To measure the phase, interferometric systems are needed. This class of instruments are called quantitative phase microscopes (QPMs). QPMs interfere the phase object's light wave (object wave) with a second known light wave (called the reference wave) resulting in interference patterns (called interferograms). Phase images are then reconstructed from interferograms, by computationally solving an inverse problem. Here, to demonstrate $\partial\mu$, we selected a slightly different task, i.e. all-optical phase retrieval. All-optical phase retrieval does not require computational reconstruction. The interference pattern (i.e. the intensity recorded on the camera) itself should be the phase image. All-optical phase retrieval is a challenging design task with no general analytical solution using linear optics. The first approach, i.e. phase contrast, was introduced by Zernike [17, 18]. Following Zernike, Glückstad [19] introduced the generalized phase contrast (GPC) method. Both these were bottom-up designed, starting from design rules to generate desired interferences through mathematical treatment of the problem. More recently, in parallel to our work ([20]), Mengu *et al.* [21] demonstrated all-optical phase retrieval using a diffractive deep neural network (D2NN). This is a top-down design approach, and they trained and analyzed the D2NNs on a number of datasets. Our work differs from Mengu *et al.* [21], in the way we initialize the top-down design using inductive biases, interpretability of learned designs, the use of data with complex biological features, and experimental demonstration in the visible wavelengths. Importantly, we replicated the D2NN, and GPC methods, to compare and contrast all three (i.e. D2NN, GPC, and ours) in an unbiased fashion on the same datasets in numerical experiments. We also include a comprehensive discussion of more related work in the supplementary section A for the interested reader.

The rest of the paper is organized as follows. In the results section, we first introduce our proposed Differen-

tiable Microscopy ($\partial\mu$) approach. Subsequently, we present three differentiable optical architectures designed for all-optical phase retrieval. Next, we evaluate the performance of the Generalized Phase Contrast (GPC) method for the same purpose. To gauge the effectiveness, we numerically test multiple variants of all the approaches, on multiple datasets. We then experimentally demonstrate a proposed $\partial\mu$ design as a proof-of-concept. Last, we critically review all designs yielding intriguing insights into the top-down design of microscopes.

2. Results

2.1. Differentiable Microscopy: Designing optical microscopes top-down

Classical optical system design is a bottom-up approach where physicists assemble optical elements in a certain configuration to manipulate light passing through them. In our example of all-optical phase retrieval, we know that the signal is in the phase of the light field, and hence we need to assemble optics to measure the phase of light. The phase of light is measured by interfering the object-light-field with a reference field. But, when this phenomenon is not known *a priori* we cannot initiate an optical design bottom-up.

In Differentiable Microscopy, we present a top-down design approach to bypass this initialization problem. We start with the application, i.e., the desired input and output of the optical system. At this stage, the instrument design is thought of as a backbox that translates the input light field to an output intensity field (see the design stage in Fig. 1A). Our goal is to assume a reasonable functional form (for the design) with a set of parameters, assuming at least one configuration of the parameters will approximate a function that can translate inputs to the desired outputs. When the functional form is known we can “train” the parameters by some supervised machine learning approach given paired input and outputs. We will later discuss how the input-output pairs can be formulated. We first discuss how a “reasonable functional form” can be designed.

We know some constraints of the functional form. We know that it should be complex-valued, but linear (assuming no non-linear optical elements are used in the design). We know that it should be translational invariant laterally so that it is not sensitive to the position of objects in the sample. We first start with a function with these properties; in our example, we used a linear complex-valued convolutional neural network (see Fig. 1B and results in section 2.2) given the empirical ease of optimizing such a function. We then transformed this design into an optical design as described below. Our CNN is a fully convolutional function with only linear layers. Thus, the final form of the CNN should be equivalent to a single convolution that can be performed in the Fourier domain. In optics, this is a classical Fourier processor with one optical filter in the Fourier plane of a 4-f system (see Fig. 1C and results in section 2.3). The trainable parameters of the model are the transmission coefficients of the Fourier filter. For a given input field with an ‘ N ’ number of upstream diffraction-limited spots, the filter consists of ‘ $2N$ ’ trainable parameters. From a machine learning standpoint, we can call this a “differentiable optical architecture” with $2N$ parameters. Note that our architecture was derived with the help of desired “inductive biases” (linearity, complex weights, and translation invariance). We also note that one could further generalize this idea to other optical processors like diffractive-deep neural networks (D2NNs) by ignoring the inductive biases and having the model learn them through data at the expense of increased parameters (see Fig. 1.D and results in section 2.4). Nevertheless, both these models are examples of potential “optical architectures” and can be used to initialize an optical design top-down. In this work we discuss how they are used in our proposed framework; we leave the discovery of better architectures for future work.

The last piece of the puzzle is how to configure or “learn” the parameters of the optical architecture through data. If we have a dataset of paired inputs and outputs, one may start with a randomly initiated optical architecture and optimize its parameters given a loss function reflecting the desired image translation behavior of the system. To do this, we need a suitable loss function to optimize, and a dataset with paired inputs and outputs. Let’s discuss these two aspects with respect to our example. First, in our example of all-optical phase retrieval microscopy, we want to measure the phase information of the input light field on the camera of the optical system (i.e. the output). In other words, the output light field’s intensity should contain the phase information of the input field. So, the loss function should reflect this behavior. Thus, any loss that measures the distance between a scaled version of the output intensity and the input phase is suitable. In other words, the loss should optimize for the output intensity to be directly proportional to the input phase. We discuss the losses used in this work in the method section 5.1. Second, we need a dataset with input-output pairs to train the model. Such a dataset is easy to formulate with only the input light fields. This is because the desired output is readily available in the inputs, i.e. the phase component of the input light field. In this work, we synthetically generated input light fields for objects such as

MNIST digits by placing them in the phase components of the light field. We also utilized pre-acquired light fields of HeLa cells, and bacteria specimens. We discuss these datasets in the methods section 5.2.

Last, this process can be mathematically formulated as below.

$$A_{out}e^{j\phi_{out}} = G_{\{w_i\}}(A_{in}e^{j\phi_{in}}) \quad (1)$$

Here, $G_{\{w_i\}}(\cdot)$ is the optical model with a set of learnable parameters $\{w_i\}$. $A_{in}e^{j\phi_{in}}$ is the input light field with A_{in} and ϕ_{in} as the amplitude and phase components; similarly, $A_{out}e^{j\phi_{out}}$ is the output light field. We then utilize a loss function $\mathcal{L}(|A_{out}|^2/S, \phi_{in})$ that imposes the output intensity, $|A_{out}|^2$, to be directly proportional to the input phase ϕ_{in} . Here S is the constant of proportionality which is either pre-set or learned (see methods 5.1). Finally, we perform the following optimization over the training dataset.

$$\{w_{i,optimal}\} = \arg \min_{\{w_i\}} \left(\text{Average}_{\forall \phi_{in} \in \text{Trainingset}} \left(\mathcal{L} \left(\frac{|G_{\{w_i\}}(A_{in}e^{j\phi_{in}})|^2}{S}, \phi_{in} \right) \right) \right) \quad (2)$$

Please see section 5.1 for the exact details about the process.

In the next sections, we discuss this process for three top-down designs –i.e., the complex-valued CNN (section 2.2), the Fourier processor (section 2.3), and the D2NN (section 2.4)– we tested for our all-optical phase to intensity conversion example. We then present a bottom-up design for the same example for comparison (section 2.5). Last, we present an experimental validation of a Fourier processor design by implementing it in an optical system (section 2.6).

2.2. Complex-valued Linear CNN: Feasibility of Linear Phase Retrieval using a black-box model

All optical phase to intensity conversion is analytically unsolvable for an arbitrary input. We, therefore, first established the existence of approximate solutions to our data distributions. To do that, we trained a complex-valued linear CNN (C-CNN) imitating optical constraints. We experimented with different hyper-parameters (the number of layers, kernel size, and the number of channels) in the CNN architecture for each dataset (see supplementary section C.1). All kernels were complex-valued [22] with no non-linear activations. A bias term was added only to the last layer. The CNN architecture is shown in Fig. 1B.

Qualitative results of the CNN model are shown in Fig. 2B. Corresponding quantitative results are shown in row C-CNN in table 1. As seen, the model achieved high SSIM [23] values for all datasets. For MNIST[0, 2π] and HeLa[0, 2π] datasets the SSIM values were slightly lower than those for the others. This is consistent with previous work (chapter 2 of [24]) that implies that the linear approximation doesn't hold well at large phase variations. Qualitative results agree with this observation (see Fig. 2B). Thus our complex-valued CNN, established the existence of linear approximate solutions for given data distributions, while also setting empirical upper bounds for all-optical phase retrieval on each dataset.

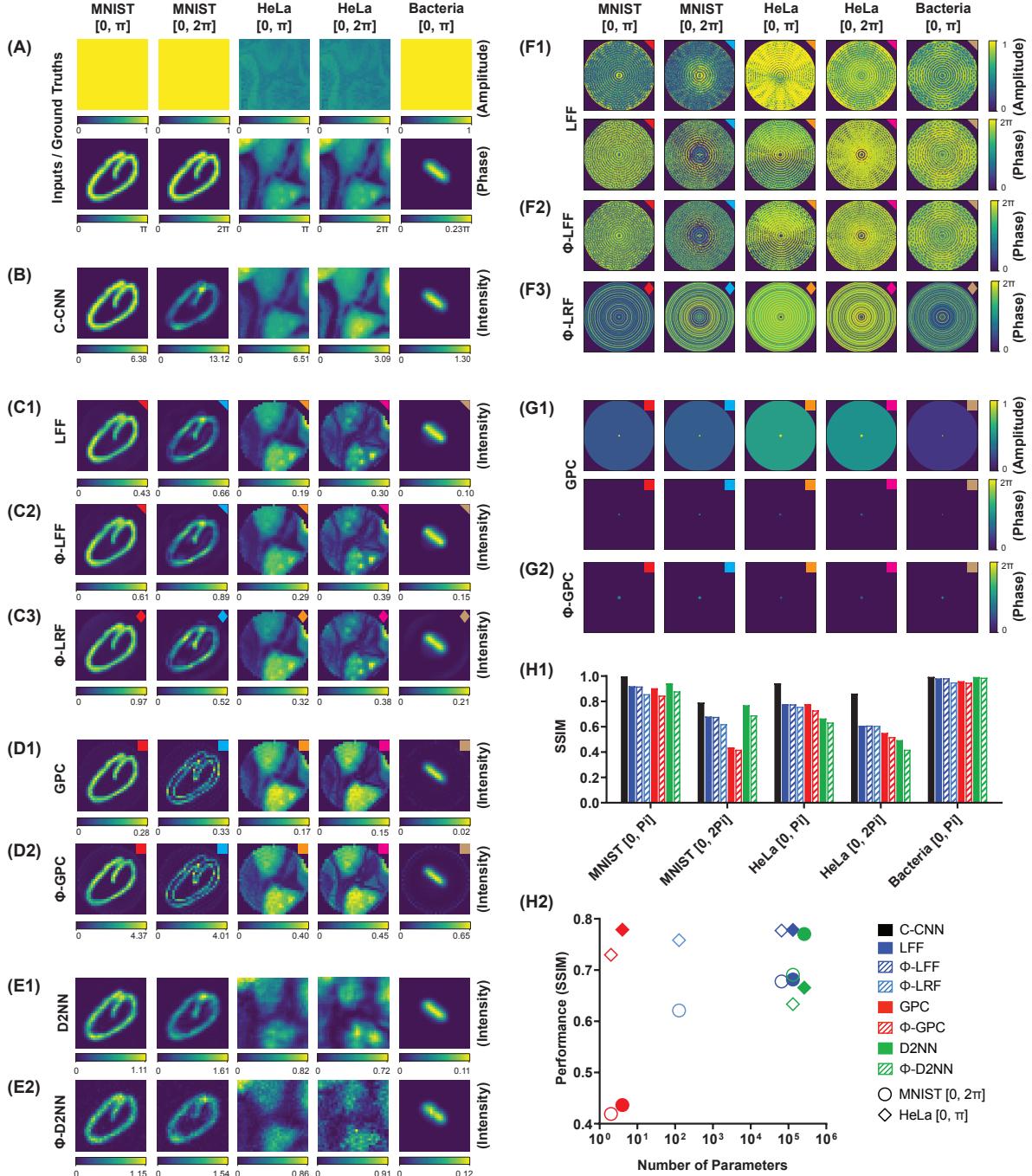


Figure 2. Overall qualitative results. **(A)** The amplitude (top row) and phase (bottom row) of the input field to the system for each dataset. **(B)** The results of the complex-valued linear CNN for the phase-to-intensity conversion task on each test dataset. The output intensity maps are visually similar to the input phase maps, suggesting a linear model can learn to convert phase to intensity for a given dataset. **(C1, C2, C3**) Show the reconstruction results by the best performing LFF, ϕ -LFF, and ϕ -LRF learned for each dataset, respectively. Similarly, **(D1, D2, E1, E2**) show the respective results for the GPC, ϕ -GPC, D2NN, and ϕ -D2NN. **(F1**) Shows the amplitude (top row) and phase (bottom row) of the learned LFFs for each dataset. **(F2, F3**) Shows the phase of the learned ϕ -LFF and ϕ -LRF for each dataset respectively. **(G1**) Shows the amplitude (top row) and phase (bottom row) of the learned GPC filters for each dataset. **(G2)** Shows the phase of the learned ϕ -GPC filter for each dataset. **(H1)** Shows a comparison of $\partial\mu$ architectures on different datasets. Our approach consistently outperforms the GPC [24] for all datasets except the HeLa[0, π] dataset, where it performs on-par with the GPC. The complex-valued CNN sets the empirical upper bounds for each dataset. **(H2)** Shows the performance of each $\partial\mu$ and GPC architecture with the trainable number of parameters. The ϕ -LRF achieves comparable/ superior performance for the MNIST [0, 2π] and HeLa [0, π] datasets with a significantly lower number of trainable parameters compared to the LFFs and ϕ -D2NN architectures. Even though the GPC shows comparable performance with the ϕ -LRF for the HeLa [0, π] dataset, the ϕ -LRF shows a significant improvement in performance for the MNIST [0, 2π] dataset.

Method	MNIST [0, π]		MNIST [0, 2π]		HeLa [0, π]		HeLa [0, 2π]		Bacteria [0, π]	
	SSIM \uparrow	L1 \downarrow	SSIM \uparrow	L1 \downarrow	SSIM \uparrow	L1 \downarrow	SSIM \uparrow	L1 \downarrow	SSIM \uparrow	L1 \downarrow
C-CNN	0.9982	0.0041	0.7913	0.0539	0.9417	0.0248	0.8619	0.0485	0.9938	0.0008
LFF	0.9205	0.0196	0.6814	0.0608	0.7783	0.0688	0.6078	0.0937	0.9823	0.0010
GPC*	0.9036	0.0320	0.4364	0.0951	0.7786	0.0748	0.5509	0.1091	0.9600	0.0029
D2NN	0.9433	0.0201	0.7703	0.0486	0.6655	0.0936	0.4942	0.1274	0.9926	0.0007
ϕ -LFF	0.9177	0.0196	0.6777	0.0608	0.7771	0.0688	0.6096	0.0937	0.9825	0.0010
ϕ -LRF	0.8573	0.0293	0.6211	0.0671	0.7583	0.0730	0.6078	0.0943	0.9508	0.0023
ϕ -GPC*	0.8466	0.0430	0.4191	0.0982	0.7297	0.0755	0.5184	0.1100	0.9479	0.0036
ϕ -D2NN	0.8796	0.0285	0.6906	0.0591	0.6334	0.0945	0.4191	0.1316	0.9867	0.0009

* Output is scaled for each dataset (see supplementary section B.1.2)

Table 1. **Overall quantitative results.** Performance of the Complex-valued CNN (C-CNN), GPC method, the learnable Fourier filter (LFF), and the D2NN are shown for each dataset. For the models learned with the \mathcal{L}_{LT} loss (see section 5.1), the best-performing cases in each performance metric for each dataset are in bold. Same values are compared side-by-side in Fig. 2-H1.

2.3. Learnable Fourier Filter: Black-box to an Optical Architecture

Our complex-valued CNN suggests that all-optical phase-to-intensity conversion is possible through linear convolutions. In the Learnable Fourier Filter (LFF) model, we optically implemented the convolution operation. Fig. 1.C shows the optical schematic of the model. The model consisted of an optical 4-*f* system (see supplementary section B.1.1) and a circular Fourier filter. The filter was on a 256×256 grid of transmission coefficients. The coefficients inside the filter were treated trainable; the ones outside were set to zero (see filters in Fig. 2.F1). The input/output of the model was of the same size as the filter. But the information of the phase object was placed within the small circular aperture on a 32×32 grid. The grid was padded $8\times$ to make up the 256×256 input to maintain a sufficient frequency sampling according to the Nyquist criterion. The first lens of the 4-*f* system, Fourier-transforms the input field; the filter modulates the field; and the second lens, inverse-Fourier-transforms the field back to the spatial domain. This physical system can be mathematically modeled as,

$$G_{\{H_t(x,y)\}} \equiv IDFT2(H_t(x,y) \circ DFT2(.)) \quad (3)$$

where H_t represents the learnable transmission coefficient matrix. $DFT2(.)$ and $IDFT2(.)$ are the two-dimensional Fourier and inverse Fourier transforms. The coefficients of the filter were initialized and optimized on each training data set, through the loss functions in eq. (4) or eq. (6). We conducted multiple experiments with different configurations and parameterizations of learnable Fourier filters on each dataset as explained below.

First, all transmission coefficients (both amplitude and phase) of the LFF were treated trainable. Results for this configuration are shown in row LFF in table 1 and Fig. 2.C1. The filters learned, for each dataset, are shown in Fig. 2.F1. Second, only the phase components of the transmission coefficients were treated trainable; the amplitudes were frozen to a unit amplitude. The results for this configuration are shown in row ϕ -LFF in table 1 and Fig. 2.C2. The filters learned, for each dataset, are shown in Fig. 2.F2. Third, we constrained the ϕ -LFF to be radial symmetric, to promote rotational invariance as an additional inductive bias. This configuration allowed us to reparameterize the filter design with two to three orders of magnitude lesser number of parameters compared to the ϕ -LFFs. Results for this configuration are shown in row ϕ -LRF in table 1 and Fig. 2.C3. The filters learned, for each dataset, are shown in Fig. 2.F3. As seen in our results, LFF performance decreased from the C-CNN baseline for all datasets and the degradation was more pronounced for the two Hela datasets. Interestingly ϕ -LFF performed similarly to the LFF models. Last, ϕ -LRF models performed slightly worse than the ϕ -LFF and LFF models.

Next, we tested an optical architecture that doesn't use any inductive bias for the same all-optical phase retrieval task.

2.4. D2NN: All-optical Phase Retrieval Through an Optical Architecture without Inductive Biases

Our third model is a Diffractive Deep Neural Network (see supplementary section A for a brief review of D2NNs). The model consists of a large number of parameters distributed in a partially-connected cascade of diffractive layers. A stand-alone layer is functionally similar to the learnable filter in the previous section. But their placement and combined function are not driven by any inductive biases. Thus the model should learn these constraints through data. Nevertheless, the D2NN is an interesting miniaturized design. (see supplementary section B.2 for details on mathematical modeling of D2NNs).

The proposed architecture is shown in Fig. 1.D. The network consisted of 8 diffractive layers which were separated by 5.3λ distance (λ is the operating wavelength of the input field). Each layer of the network consisted of a 128×128 neuron grid that contained trainable complex-valued transmission coefficients. The size of each neuron was $\lambda/2 \times \lambda/2$. The input to the network was a light field, i.e. a complex-valued image, where the information of interest can be found in its phase. This input was located in a 32×32 grid, 5.3λ before the first layer of the D2NN. The reconstruction plane (i.e. the image plane where a detector is placed) was at a 9.3λ distance from the last optical layer. The reconstruction was done on 32×32 grid on this reconstruction plane. The element size of the input grid and reconstruction grid was the same as the size of a neuron. Hence, this compact system has dimensions of $64\lambda \times 64\lambda \times 51.7\lambda$.

After mathematically modeling the diffractive layers as mentioned in supplementary section B.2, the learnable transmission coefficients (t_i) of each layer were optimized with the objective defined in eq. (4) and eq. (6). To maintain the passive nature of the layers, we constrained the amplitudes of the transmission coefficients to the range $[0, 1]$. In our study, we consider two types of D2NN designs; D2NNs composed of layers that modulate both the phase and the amplitude of the incoming waves (simply referred to as D2NN) and those that modulate only the phase (referred to as ϕ -D2NN). Quantitative results of D2NN and ϕ -D2NN designs are shown in Table 1. Qualitative results are shown in Figs. 2.E1, and Fig. 2.E2. As seen in the results, D2NNs performed slightly better than LFFs for MNIST and bacteria datasets but worse than LFFs for the more complex Hela datasets. Unlike in LFFs, ϕ -D2NNs performed slightly worse than D2NNs.

With no inductive biases and a large number of learnable parameters, D2NN is an optical architecture at one extreme end of the spectrum. In the next section, we tested the other end of the spectrum -i.e., a fully analytically designed optical architecture with only a few learnable parameters.

2.5. Generalized Phase Contrast: All-optical Phase Retrieval by an Analytically Designed Optical Architecture

Optical architectures presented in the previous sections are top-down approaches that use data to learn the optical design. In these models, a large number of parameters should be learned in general. In this section, we compare the other end of the spectrum, where the entire design is done bottom-up with domain knowledge. To this end, we tested an all-optical phase retrieval method called Generalized Phase Contrast (GPC) [19]. In GPC a two-part Fourier filter is designed to generate interference with the near-zero frequency components of the object light field, with its high-frequency components. The passband and transmission coefficients of the two parts of the filter were designed using the target data distributions (see supplementary section B.1.2 for detailed guidelines of the design process).

Similar to the LFF case, we tested both GPC and ϕ -GPC (phase-only filter) configurations. Results are shown in Table 1 and Fig. 2.D1 & D2. The filters learned, for each dataset, are shown in Figs. 2.G1 and G2. GPC and ϕ -GPC performed similarly to LFF and ϕ -LFF models on MNIST $[0, \pi]$, Hela $[0, \pi]$, and Bacteria $[0, \pi]$ datasets. But GPC models were considerably worse than LFF models for MNIST $[0, 2\pi]$ and Hela $[0, 2\pi]$ datasets.

Next, we present an experimental validation of the LFFs to show that our top-down designed optical models can be experimentally implemented.

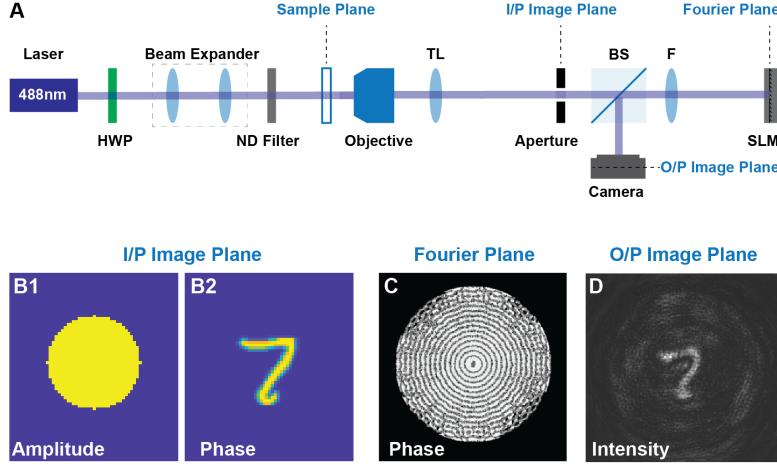


Figure 3. Experimental validation of the LFF using an SLM. (A) The optical schematic of the experimental setup (HWP - half-wave plate; ND Filter - neutral density filter; TL - tube lens; BS – 50/50 beam splitter; F – 180mm focal length lens; SLM spatial light modulator; I/P input; O/P – output). (B1,2) The ideal input image field of the fabricated phase object (B1 – Amplitude, and B2 – phase) generated at the first image plane of the microscope. (C) The phase filter configured on the SLM. (D) The image recorded by the camera (i.e., the intensity) at the output image plane of the 4-f system.

2.6. Experimental Validation of the Learnable Fourier Filter

We experimentally demonstrated our LFF architecture shown in Fig. 1.C. The goal was to validate that our data-driven top-down designs can be physically built with intended properties. To this end, we implemented the ϕ -LFF design for the MNIST dataset using a spatial light modulator (SLM). We fabricated a phase mask of digit 7 (from the test set) as the test object (using implosion fabrication [25]). The optical schematic of the setup is shown in Fig. 3.A (a photograph of the experimental setup is also shown in Fig. S13). An optically pumped semiconductor laser (Sapphire LPX 488, Coherent, Inc, CA, USA), was used as the light source. A half-wave plate ensured the correct polarization for the SLM. The output beam was expanded and collimated using a beam expansion system. A reflective neutral density (ND) filter (ND 2.0, Thorlabs, USA) was placed in the beam path to control the laser intensity. The phase object (mounted on a 3D stage) was illuminated using the collimated beam and the transmitted light was imaged by an objective (M Plan-50 \times /0.75 NA, Olympus) and a tube lens ($f=180$ mm, Achromatic, Thorlabs, USA). An aperture was placed at the image plane to match the field of view to that of the design. Next, a reflective 4-f system was implemented using a single lens ($f=180$ mm, Achromatic, Thorlabs, USA) and the SLM. The SLM (EXULUS-HD2, Thorlabs, USA) was placed at the Fourier plane of the 4-f (see Fig. 3.A). The reflected beam passing through the same lens was projected on the detector (Panda PCO 4.2, Excelitas PCO GmbH, Germany) using a beam splitter (BS 50/50, Thorlabs, USA). The respective LFF (see Fig. 3.C) was configured on the SLM after resampling to match the design (see supplementary section D for details). Fig. 3.D shows the image recorded on the camera, i.e., the output intensity. Fig. 3.B2 shows the expected phase image (from the fabricated digit) at the input image plane.

This demonstration was intended to be only a proof of concept. We did expect and observed some model mismatch from the design as discussed further in supplementary section D. While we leave solving this model mismatch to future work, the results presented here demonstrate that our top-down designs do in fact work as intended in experiments.

3. Discussion

In this work, we introduced differentiable microscopy ($\partial\mu$) as a top-down optical system design paradigm. Using the concept of "optical architectures", $\partial\mu$ bypasses the need to initialize a design. We demonstrated how an optical architecture can be derived with the help of inductive biases of the imaging task. To this end, we proposed an LFF architecture with different parameterizations. Using all-optical phase retrieval as an example,

we comprehensively compared it with an architecture derived with no inductive biases (i.e., D2NN) as well as an architecture that was fully analytically designed (i.e. GPC). Fig. 2.H1 and Table 1 summarizes the performance of all models on all datasets.

First, at-least one top-down design (i.e. from LFF, D2NN, ϕ -LFF, ϕ -LRF, ϕ -D2NN) outperformed their bottom-up GPC and ϕ -GPC counterparts interms of L1-loss on all datasets (SSIM trended the same except on HeLa $[0, \pi]$). This is because top-down design approaches could utilize the data distribution apriori while bottom-up design paradigms cannot. But this advantage comes at the cost of having to effectively optimize a large number of parameters. None of the architectures could reach the numerical upper bound (i.e., the C-CNN performances), perhaps due to sub-optimal convergence of parameters. Thus an interesting future direction is to explore better optimizers for these physics-constrained optical architectures. Nevertheless, this sub-par performance does not necessarily mean that the results are not quantitative. L1 loss measures the pixel-wise difference in phase values, and can be treated as a metric for quantitativeness. However, in the all-optical setting, there's a power depended constant of proportionality that should be measured to convert the intensity values to the exact phase values. Therefore another interesting future direction is to develop a robust strategy to measure the proportionality constant, perhaps through a second intensity measurement.

Second, all models performed similarly on datasets with simple features and low phase variations (i.e. the MNIST $[0, \pi]$ and bacteria $[0, \pi]$). For MNIST $[0, 2\pi]$, with simple features but high phase variation, GPC performed much worse than the others. For Hela $[0, \pi]$, with complex features but low phase variation, D2NN designs performed much worse than others most likely due to convergence issues. For Hela $[0, 2\pi]$, with complex features and high phase variation, LFFs performed considerably better than the rest. In general, we conclude that LFFs struck a balance between the number of parameters and performance in most cases (see Fig. 2.H2). This result suggests that a single diffractive layer, when placed right, can outperform a D2NN. This is because, though comprised of many layers, a D2NN is a cascade of linear operators that collapses to a single linear operation. The cascading is needed in the spatial domain to connect all pixels; but the same can be achieved by the Fourier operator. In fact, the Fourier operator fully connects the network while D2NN is partially connected. Furthermore, compared to D2NNs, it is easier to train LFFs due to the reduction of trainable parameters and the reduced computational complexity in wave propagation. For instance, while D2NNs are supposed to learn the inductive bias through data, they are poor in doing so in the presence of complex features. Moreover, our Phase LRF parameterization, in comparison to the other LFFs and D2NNs (Fig. 2.H2), can reduce the number of parameters by orders of magnitudes, at the cost of slight performance reduction. This may be a promising path toward efficient training.

Third, the top-down designed models will not generalize beyond their data distribution. It is important to understand the limits of the model performance when operated beyond the intended data distribution. To explore the generalizability limits of the LFF and D2NN, we conducted a series of validation experiments (see supplementary section C.2). Specifically, we trained the proposed ϕ -LFF, ϕ -LRF, and ϕ -D2NN on a specific dataset and tested them on the rest of the datasets. We observe that the models trained on the datasets with large phase variation tend to generalize on the same datasets with smaller phase variation.

Fourth, we experimentally validated that our learned LFF designs can be built with intended design properties. But our validation was strictly qualitative and only a proof of principle. Future work should experimentally replicate the numerical benchmarks presented in this work over multiple datasets. In our proof of concept experiment, we observed a model mismatch. Our model did not include exact discretization and quantization constraints imposed by the spatial light modulator (SLM) and the camera detector at the design stage. We introduced these constraints post-design, which lead to the model mismatch. A potential future direction is to build discretization- and quantization-aware differentiable models. One may also explore fabricating the LFF designs as phase and intensity masks.

Last, we observed that some LFF designs contained similar features to the GPC design. Specifically, most LFF, ϕ -LFF, and ϕ -LRF designs learned to delay near-zero frequencies with a distinct phase shift in the middle. This is not surprising given we are aware of the rules used to design GPCs. Nevertheless, it may be of interest to optical designers in terms of discovering new design rules that aren't immediately apparent. In a hypothetical world where GPC had not been discovered, one may be able to analyze our learned LFF designs and simplify them through logical ablation experiments until an analytical rule can be distilled out of the learned designs. While we do not claim differentiable microscopy ($\partial\mu$) can discover new interpretable design rules, we do not reject the possibility of a refined $\partial\mu$ framework that can.

4. Conclusion

The current paradigm within microscopy is to develop the system hardware for a specific microscope (e.g. quantitative phase microscope (QPM)) and then use that for different applications. Here, we propose to change the paradigm by introducing the concept of differentiable microscopy ($\partial\mu$) that can learn the optical imaging process required for a given technical specification (e.g., extracting phase information) and for different targeted applications. As a first demonstration of $\partial\mu$ we learned all-optical designs for phase retrieval microscopy.

In conventional QPM, the imaging pipeline is all-optical, but the imaging reconstruction requires computation resources. Our learned phase retrieval microscopes completely remove the need of post-imaging computation resources and have a huge potential of making the footprint of the microscope compact. We believe this work will motivate researchers to exploit our designs for applications where high-speed, high-throughput, and compact setups are needed. For instance, our designs may find use in point-of-care systems in micro-biology [26], histopathology [27], and material sciences [28]. Moreover, the proposed all-optical phase-to-intensity conversion can be exploited for applications beyond microscopy, such as for two-photon-polymerization based nano-fabrication [24] and optogenetics [29].

We also think, our work is transformative in physics-based machine learning. Notably, some of our learned designs are similar to the generalized phase (GPC) concept [24]. It would be interesting to see if $\partial\mu$ can invent well-known microscope designs from data for various other tasks (such as for depth-resolved imaging or for optical coherence tomography). We anticipate that differentiable microscopy will open doors to a new generation of interdisciplinary scientists, to design optical systems for challenging imaging problems.

5. Materials and Methods

5.1. All-optical Phase to Intensity Conversion as a Learning Problem

In a machine learning perspective, all-optical phase to intensity conversion is an image translation task. A computer model, subject to the physics of light propagation (*the rules*), should learn to translate complex-valued optical fields (inputs), to intensity maps proportional to the input phase (outputs). This task should be embedded in a loss function (*the questions*) and training data (*the correct answers*).

To this end, we first introduce **phase reconstruction loss**, \mathcal{L}_ϕ . An input optical field, $x_{in} = A_{in}e^{j\phi_{in}}$ (where A_{in} is the amplitude and ϕ_{in} is the phase of the input optical field) is propagated through the proposed model $G_{wi}(.)$ to produce the output field $x_{out} = A_{out}e^{j\phi_{out}} = G_{wi}(x_{in})$ (where A_{out} is the amplitude and ϕ_{out} is the phase of the output optical field). Then phase reconstruction loss is defined as,

$$\mathcal{L}_\phi = \mathbb{E}_{x_{in} \sim P_X} [Rh_\delta(|A_{out}|^2, \phi_{in}/k)], \quad (4)$$

where k , P_X , and $Rh_\delta(.)$ respectively represent the maximum phase of a phase object of the considered dataset, the probability distribution of phase objects, and the below-defined Reverse Huber Loss [30].

$$Rh_\delta(y, f(x)) = \begin{cases} |y - f(x)| & \text{for } |y - f(x)| \leq \delta \\ \frac{(y - f(x))^2 + \delta^2}{2\delta} & \text{otherwise.} \end{cases} \quad (5)$$

We chose the Reverse Huber Loss over L1 or MSE loss as it produces better reconstruction quality (based on the empirical results). We set the δ threshold to 0.95 of the standard deviation of the ground truth image following a similar procedure to [31]. In \mathcal{L}_ϕ however, the constant of proportionality of the objective (i.e. of $|A_{out}|^2 \propto \phi_{in}$) is fixed. We propose a second **Learned Transformation Loss**, \mathcal{L}_{LT} , which relaxes the model to learn a constant of proportionality (S).

$$\mathcal{L}_{LT} = \mathbb{E}_{x_{in} \sim P_X} \left[Rh_\delta \left(\frac{|A_{out}|^2}{S}, \frac{\phi_{in}}{k} \right) \right], \quad (6)$$

\mathcal{L}_{LT} is more consistent with the problem formulation in the literature [19]. Results of learning with these losses are summarized in Table S9.

5.2. Datasets:

For the experiments, we have considered the following datasets.

Phase MNIST Digits. To preserve consistency with existing studies [32] we consider two *Phase MNIST digit* datasets for evaluation. To construct these datasets, we convert the images from the MNIST digit dataset to phase images ($e^{j\phi_{in}}$) such that $\phi_{in} \in [0, \pi]$ for one dataset (MNIST[0, π]), and $\phi_{in} \in [0, 2\pi]$ for the other dataset (MNIST[0, 2π]). Each dataset contains 54000, 6000, 10000 train, validation, test images respectively.

We considered these datasets as *sparse datasets* because they contain relatively less complex structures (e.g. Contains only a single digit located at the center of the image grid).

HeLa Dataset. To evaluate the proposed method on microscopy, we acquire a dataset of *HeLa* cells where the samples are illuminated with 600 nm light. Dataset contains 10344 images where each image represents a complex field. 80%, 10%, 10% of the dataset are utilized for training, validation, test sets respectively.

The electric field amplitudes are normalized to [0, 1] based on the maximum amplitude obtained through the steps described in Section 5.4. To remove outliers in the phase, values are clipped to [0, 2π]. To study the effect of the distribution of phase values in a limited range, the phase information of the above dataset is re-scaled to [0, π] and considered as a separate dataset for reporting performance.

Bacteria Dataset. This is another acquired dataset for the purpose of evaluating the network performances for microscopy. The dataset has phase information $\phi_{in} \in [0, \pi]$. This can also be considered as a *sparse dataset* similar to the Phase MNIST dataset.

5.3. Sample Preparation:

HeLa cells. HeLa cells were grown in a standard humidified incubator at 37 °C with 5% CO₂ in minimum essential medium supplemented with 1% penicillin/streptomycin and 10% fetal bovine serum. Cells were seeded into the Polydimethylsiloxane chamber of 10 mm × 10 mm size with 150 μm thickness on a reflecting silicon substrate. The cells were left in the incubator for 1-2 days for their growth in a densely packed manner and fixed for ~ 20 min using 4% paraformaldehyde in phosphate buffered saline. The sample is then sealed with # 1.5 cover glass from the top which enabled to use water immersion objective lens for imaging and also avoided any air current in the specimen.

Bacteria. Bacteria samples has Brownian motion due to their small sizes, which introduce challenges in imaging them using a multi-shot QPM system. Multi-shot QPM acquires multiple interferometric frames for the phase recovery and therefore specimen should be immobile within the multi frame acquisition time. The bacteria samples were also prepared on a reflecting substrate (Si-wafer) due to the reflection geometry of our optical setup. First, the substrate was thoroughly washed with dist. H₂O and dried with N₂ gas. The 2-polydimethylsiloxane (PDMS) chamber of opening 10 × 10 mm was placed on top of Si-wafer and the opening area is filled with poly-L-lysine (PLL). The substrate is incubated with PLL for 15-20 minutes and then access amount of PLL is removed and gently washed using phosphate-buffered saline (PBS). The thin layer of PLL positively charges the Si-wafer surface and helps to adhere the negatively charge bacteria cells for imaging. The 20 - 25 μl volume of bacteria sample is pipetted in the PDMS chamber and left for 30 minutes of incubation. The bacteria cells were adhered to the surface of Si-wafer due to the electrostatic attraction between the negatively charged bacteria and positively charged wafer surface. The immobile bacteria cells were further gently washed off with PBS and covered with # 1.5 cover glass from the top to use high resolution water immersion objective lens (60 \times /1.2NA) for imaging.

5.4. Quantitative Phase Imaging:

For quantitative phase imaging, HeLa sample is placed under the Linnik interferometer based quantitative phase microscopy (QPM), which works on the reflection mode. The sample is illuminated with partially spatial and highly temporal coherent light source also called pseudo-thermal light source to generate high quality interferometric images. This light source illumination has unique advantages such as high spatial and temporal resolution, high spatial phase sensitivity, extended range of optical path difference adjustment between the interferometric arms etc over conventional light sources, e.g., lasers, white light, and light emitting diodes [27]. More details of the experimental setup can be found in previous work [27]. The imaging of HeLa cells is performed using high resolution water immersion objective lens (60 \times /1.2NA) at 660 nm wavelength. This provided the theoretical transverse resolution approximately equal to 275 nm, which is quite good for high resolution phase recovery of the specimens. The interferometric data is recorded with 5.3 Megapixels ORCA-Fusion digital CMOS camera (model # C14440-20UP). The camera sensor has 2304 × 2304 pixels with 6.5 μm pixel size and provided fairly big 225 $\mu\text{m} \times$ 225 μm FOV at 60 \times optical magnification in QPM system. We acquired approximately 500 such FOVs of different region of interests of the HeLa sample and each FOV contained roughly 50 – 60 cells. For bacteria samples, more than 50 FOVs of different regions of interests were found to be sufficient to generate large amount of interferometric data due to their small sizes compared to HeLa cells. Each FOVs contained more than 500 bacteria cells and thus provided approximately 25000 bacteria cells interferometric images for machine learning. The recorded interferometric images are further post processed with random phase-shifting algorithm based on principal component analysis [33] for the recovery of the complex fields related to HeLa cells.

5.5. Implementation Details:

We implemented the optical networks with Python version 3.6.13. We considered auto differentiation in PyTorch [34] framework version 1.10.0 for the training of the optical networks. We conducted all experiments on a server with 12 Intel(R) Xeon(R) Platinum 8358 (2.60 GHz) CPU Cores, an NVIDIA A100 Graphical Processing Unit with 40 GB memory running on the Rocky 8 Linux distribution. We considered a training batch size of 32 examples and a learning rate of 0.01 for the networks. We used Adam [35] as the optimizer for all optimizations.

Competing financial interests: The authors declare no competing financial interests.

Acknowledgements: The authors would like to acknowledge Dr. Quansan Yang, Dr. Gaojie Yang, and Prof. Edward S. Boyden of Massachusetts Institute of Technology, Cambridge, USA for the fabrication of the Phase MNIST samples used for the experimental validation.

References

- [1] John P. Wourms. Light Microscopy: Renaissance and Revolution. *BioScience*, 40(5):397–399, 5 1990. 1
- [2] Jongchan Park, David J. Brady, Guoan Zheng, Lei Tian, and Liang Gao. Review of bio-optical imaging systems with a high space-bandwidth product. *Advanced Photonics*, 3(04):1–18, 2021. 1
- [3] Yujia Xue, Shiyi Cheng, Yunzhe Li, and Lei Tian. Reliable deep-learning-based phase imaging with uncertainty quantification. 2019. 1
- [4] Hongda Wang, Yair Rivenson, Yiyin Jin, ZhenSong Wei, Ronald Gao, Harun Günaydin, Laurent A. Bentolila, Comert Kural, and Aydogan Ozcan. Deep learning enables cross-modality super-resolution in fluorescence microscopy. *Nature Methods*, 16(1), 2019. 1
- [5] Colin L. Cooke, Fanjie Kong, Amey Chaware, Kevin C. Zhou, Kanghyun Kim, Rong Xu, D. Michael Ando, Samuel J. Yang, Pavan Chandra Konda, and Roarke Horstmeyer. Physics-enhanced machine learning for virtual fluorescence microscopy. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 3803–3813, 2021. 1
- [6] Ayan Chakrabarti. Learning sensor multiplexing design through back-propagation. *Advances in Neural Information Processing Systems*, (Nips):3089–3097, 2016. 1
- [7] He Sun, Adrian V. Dalca, and Katherine L. Bouman. Learning a probabilistic strategy for computational imaging sensor selection. *IEEE International Conference on Computational Photography, ICCP 2020*, 2020. 1
- [8] Amey Chaware, Colin L. Cooke, Kanghyun Kim, and Roarke Horstmeyer. Towards an intelligent microscope: Adaptively learned illumination for optimal sample classification. *ICASSP, IEEE International Conference on Acoustics, Speech and Signal Processing - Proceedings*, 2020-May:9284–9288, 2020. 1
- [9] Alex Muthumbi, Amey Chaware, Kanghyun Kim, Kevin C. Zhou, Pavan Chandra Konda, Richard Chen, Benjamin Judkewitz, Andreas Erdmann, Barbara Kappes, and Roarke Horstmeyer. Learned sensing: jointly optimized microscope hardware for accurate image classification. *Biomedical Optics Express*, 10(12):6351, 2019. 1
- [10] Roarke Horstmeyer, Richard Y. Chen, Barbara Kappes, and Benjamin Judkewitz. Convolutional neural networks that teach microscopes how to image. 2017. 1
- [11] Kanghyun Kim, Pavan Chandra Konda, Colin L. Cooke, Ron Appel, and Roarke Horstmeyer. Multi-element microscope optimization by a learned sensing network with composite physical layers. *Optics Letters*, 45(20):5684, 2020. 1
- [12] Hassaan Majeed, Shamira Sridharan, Mustafa Mir, Lihong Ma, Eunjung Min, Woonggyu Jung, and Gabriel Popescu. Quantitative phase imaging for medical diagnosis. *Journal of Biophotonics*, 10(2):177–205, 2017. 2
- [13] Mustafa Mir, Huafeng Ding, Zhuo Wang, Jason Reedy, Krishnarao Tangella, and Gabriel Popescu. Blood screening using diffraction phase cytometry. *Journal of Biomedical Optics*, 15(2), 2010. 2
- [14] Darina Roitshtain, Lauren Wolbromsky, Evgeny Bal, Hayit Greenspan, Lisa L. Satterwhite, and Natan T. Shaked. Quantitative phase microscopy spatial signatures of cancer cells. *Cytometry Part A*, 91(5):482–493, 2017. 2
- [15] Pierre Marquet, Christian Depeursinge, and Pierre J. Magistretti. Review of quantitative phase-digital holographic microscopy: promising novel imaging technique to resolve neuronal network activity and identify cellular biomarkers of psychiatric disorders. *Neurophotonics*, 1(2), 2014. 2
- [16] Chenfei Hu and Gabriel Popescu. Quantitative phase imaging (QPI) in neuroscience. *IEEE Journal of Selected Topics in Quantum Electronics*, 25(1):1–9, 2019. 2
- [17] F. Zernike. Phase contrast, a new method for the microscopic observation of transparent objects. *Physica*, 9(7), 1942. 2, 1
- [18] F. Zernike. Phase contrast, a new method for the microscopic observation of transparent objects part II. *Physica*, 9(10), 1942. 2, 1
- [19] Jesper Glückstad. Phase contrast image synthesis. *Optics Communications*, 130(4-6):225–230, 1996. 2, 7, 11, 1
- [20] Kithmini Herath, Udit Haputhanthri, Ramith Hettiarachchi, Hasindu Kariyawasam, Azeem Ahmad, Balpreet S. Ahluwalia, Chamira U. S. Edussooriya, and Dushan Wadduwage. Differentiable microscopy designs an all optical quantitative phase microscope, 2022. doi: [10.48550/ARXIV.2203.14944](https://arxiv.org/abs/2203.14944). 2
- [21] Deniz Mengü and Aydogan Ozcan. All-optical phase recovery: Diffractive computing for quantitative phase imaging. *Advanced Optical Materials*, 10(15):2200281, 2022. 2, 1

- [22] Chiheb Trabelsi, Olexa Bilaniuk, Ying Zhang, Dmitriy Serdyuk, Sandeep Subramanian, Joao Felipe Santos, Soroush Mehri, Negar Rostamzadeh, Yoshua Bengio, and Christopher J Pal. Deep complex networks. In *International Conference on Learning Representations*, 2018. 4
- [23] Zhou Wang, A.C. Bovik, H.R. Sheikh, and E.P. Simoncelli. Image quality assessment: From error visibility to structural similarity. *IEEE Transactions on Image Processing*, 13(4):600–612, 2004. 4
- [24] Jesper Glückstad and Darwin Palima. Generalized phase contrast. *Springer Series in Optical Sciences*, 146:7–12, 2009. 4, 5, 10, 1, 2
- [25] Daniel Oran, Samuel G. Rodrigues, Ruixuan Gao, Shoh Asano, Mark A. Skylar-Scott, Fei Chen, Paul W Tillberg, Adam H. Marblestone, and Edward S. Boyden. 3D nanofabrication by volumetric deposition and controlled shrinkage of patterned scaffolds. *Science*, 362(6420):1281–1285, 12 2018. 8
- [26] Youngju Jo, Jaehwang Jung, Jee Woong Lee, Della Shin, Hyunjoo Park, Ki Tae Nam, Ji Ho Park, and Yongkeun Park. Angle-resolved light scattering of individual rod-shaped bacteria based on Fourier transform light scattering. *Scientific Reports*, 4:1–6, 2014. 10
- [27] Azeem Ahmad, Vishesh Dubey, Nikhil Jayakumar, Anowarul Habib, Ankit Butola, Mona Nystad, Ganesh Acharya, Purusotam Basnet, Dalip Singh Mehta, and Balpreet Singh Ahluwalia. High-throughput spatial sensitive quantitative phase microscopy using low spatial and high temporal coherent illumination. *Scientific Reports*, 11(1):15850, 2021. 10, 12
- [28] Renjie Zhou, Chris Edwards, Gabriel Popescu, and Lynford Goddard. Semiconductor defect metrology using laser-based quantitative phase imaging. *Quantitative Phase Imaging*, 9336(March 2015):93361I, 2015. 10
- [29] Jesper Glückstad, RenéL. Eriksen, and Andreas Gejl Madsen. GPC-modalities for neurophotonics and optogenetics. International Society for Optics and Photonics, SPIE, 2021. 10
- [30] Laurent Zwald and Sophie Lambert-Lacroix. The BerHu penalty and the grouped effect. 2012. 11
- [31] Deniz Mengü, Muhammed Veli, Yair Rivenson, and Aydogan Ozcan. Classification and reconstruction of spatially overlapping phase images using diffractive optical networks. pages 1–30, 8 2021. 11, 2
- [32] Xing Lin, Yair Rivenson, Nezih T. Yardimci, Muhammed Veli, Yi Luo, Mona Jarrahi, and Aydogan Ozcan. All-optical machine learning using diffractive deep neural networks. *Science*, 361(6406), 2018. 11, 2, 3
- [33] J. Vargas, J. Antonio Quiroga, and T. Belenguer. Phase-shifting interferometry based on principal component analysis. *Opt. Lett.*, 36(8):1326–1328, Apr 2011. 12
- [34] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Kopf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. Pytorch: An imperative style, high-performance deep learning library. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, editors, *Advances in Neural Information Processing Systems 32*, pages 8024–8035. Curran Associates, Inc., 2019. 12
- [35] Diederik Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *International Conference on Learning Representations*, 12 2014. 12
- [36] Yair Rivenson, Tairan Liu, Zhensong Wei, Yibo Zhang, Kevin de Haan, and Aydogan Ozcan. PhaseStain: the digital staining of label-free quantitative phase microscopy images using deep learning. *Light: Science and Applications*, 8(1), 2019. 1
- [37] Yoav N. Nygate, Mattan Levi, Simcha K. Mirsky, Nir A. Turko, Moran Rubin, Itay Barnea, Gili Dardikman-Yoffe, Miki Haifler, Alon Shalev, and Natan T. Shaked. Holographic virtual staining of individual biological cells. *Proceedings of the National Academy of Sciences of the United States of America*, 117(17), 2020. 1
- [38] Yair Rivenson, Yibo Zhang, Harun Günaydin, Da Teng, and Aydogan Ozcan. Phase recovery and holographic image reconstruction using deep learning in neural networks. *Light: Science and Applications*, 7(2), 2018. 1
- [39] Ayan Sinha, Justin Lee, Shuai Li, and George Barbastathis. Lensless computational imaging through deep learning. *Optica*, 4(9), 2017. 1
- [40] Michael Kellman, Emrah Bostan, Michael Chen, and Laura Waller. Data-Driven Design for Fourier Ptychographic Microscopy. *2019 IEEE International Conference on Computational Photography, ICCP 2019*, pages 2–9, 2019. 1
- [41] Michael R. Kellman, Emrah Bostan, Nicole A. Repina, and Laura Waller. Physics-Based Learned Design: Optimized Coded-Illumination for Quantitative Phase Imaging. *IEEE Transactions on Computational Imaging*, 5(3), 2019. 1
- [42] Julie Chang, Vincent Sitzmann, Xiong Dun, Wolfgang Heidrich, and Gordon Wetzstein. Hybrid optical-electronic convolutional neural networks with optimized diffractive optics for image classification. *Scientific Reports*, 8(1):1–10, 2018. 1

- [43] Carlos Mauricio Villegas Burgos, Tianqi Yang, Yuhao Zhu, and A. Nickolas Vamivakas. Design framework for metasurface optics-based convolutional neural networks. *Applied Optics*, 60(15):4356, 2021. [1](#)
- [44] Shane Colburn, Yi Chu, Eli Shilzerman, and Arka Majumdar. Optical frontend for a convolutional neural network. *Applied Optics*, 58(12):3179, 4 2019. [1](#)
- [45] Qiuhan Wu, Xiubao Sui, Yuhang Fei, Chen Xu, Jia Liu, Guohua Gu, and Qian Chen. Multi-layer optical Fourier neural network based on the convolution theorem. *AIP Advances*, 11(5):1–6, 2021. [1](#)
- [46] Linmei Li, Shiqiang Yan, Bingcheng Lin, Qihui Shi, and Yao Lu. *Single-Cell Proteomics for Cancer Immunotherapy*, volume 139. Elsevier Inc., 1 edition, 2018. [2](#)
- [47] Jingxi Li, Deniz Mengu, Yi Luo, Yair Rivenson, and Aydogan Ozcan. Class-specific differential detection in diffractive optical neural networks improves inference accuracy. *Advanced Photonics*, 1(04), 2019. [2](#)
- [48] Tao Yan, Jiamin Wu, Tiankuang Zhou, Hao Xie, Feng Xu, Jingtao Fan, Lu Fang, Xing Lin, and Qionghai Dai. Fourier-space Diffractive Deep Neural Network. *Physical Review Letters*, 123(2), 2019. [2](#)
- [49] Yi Luo, Deniz Mengu, Nezih T. Yardimci, Yair Rivenson, Muhammed Veli, Mona Jarrahi, and Aydogan Ozcan. Design of task-specific optical systems using broadband diffractive neural networks. *Light: Science and Applications*, 8(1):1–14, 2019. [2](#)
- [50] Md Sadman Sakib Rahman and Aydogan Ozcan. Computer-Free, All-Optical Reconstruction of Holograms Using Diffractive Networks. *ACS Photonics*, page acsphotonics.1c01365, 10 2021. [2](#)
- [51] J.W. Goodman. *Introduction to Fourier Optics*. Electrical Engineering Series. McGraw-Hill, 1996. [2](#), [3](#), [5](#), [6](#)
- [52] J. A. Ratcliffe. Some Aspects of Diffraction Theory and their Application to the Ionosphere. *Reports on Progress in Physics*, 19(1):306, 1 1956. [5](#)

Differentiable Microscopy Designs an All Optical Phase Retrieval Microscope

Supplementary Material

A. Related Work

Image Reconstruction through Post-processing. Commonly, images acquired from microscopy techniques need processing before they can be used for desired tasks. In the context of phase imaging, deep learning has shown promise in improving microscopy techniques through the post-processing of acquired images. These methods approximate the inverse function of imaging models and have applications such as virtual staining [36,37] and phase retrieval [38,39].

Optical Hardware Optimization. Rather than following the static optical setup of the microscope and post-processing its acquired images, recent methods have focused on optimizing certain parts of the optical hardware itself. Several approaches focus on optimizing the illumination patterns of the microscope [8–11]. This research direction of jointly optimizing the forward optics (by learning illumination patterns) with the inverse reconstruction model has been able to reduce the data requirement in QPM [40, 41]. However, since the focus is only on the *illumination aspect*, the extent to which the forward optics model can be optimized is constrained. In contrast, in differentiable microscopy ($\partial\mu$), here we focus towards learning new optical systems, and interpretable design rules.

Fourier Plane Filter Optimization. The 4-f optical system (Fig. 1(a)) produces the Fourier transform of the incident spatial light at the Fourier plane. This Fourier transform property of a lens can be leveraged to perform the convolution operation through light itself by placing filters at the Fourier plane. The effectiveness of this optical convolution has been studied in the literature to perform various classification tasks [42–45].

All-Optical Phase Retrieval Methods. The work of Zernike [17, 18] paved the way to clearly visualize transparent biological samples under a microscope without applying contrast dyes onto the specimen. Thus, the development of phase contrast microscopy is a significant milestone in biological imaging. This phase contrast method is implemented in a common-path interferometer where the signal and reference beams resulting from the illumination travel along the same optical axis and interfere at the output of the optical system generating an interferogram. The phase perturbations introduced by the optical characteristics (refractive index, thickness) of the object are converted to observable intensity variations on the interferogram, by the use of a Fourier plane filter in the optical path. This filter functions as a phase shifting filter which imparts a quarter-wave shift on the undiffracted light components (the reference beam). However, the linear relationship between the input phase and output intensity of this system is derived based on the “small-scale” phase approximation, where the largest input phase deviation is considered to be $\pi/3$. Glückstad [19] generalized Zernike’s phase contrast method and introduced the generalized phase contrast (GPC) method to provide an all-optical linear phase to intensity conversion approach, which overcomes the restriction of “small-scale” phase approximation. The descriptive mathematical analysis of the filter implementation in [24] shows the utilization of a phase shifting Fourier plane filter, where its parameters are carefully selected to appropriately generate the synthetic reference wave (SRW) to reveal the input phase perturbations in the output interference pattern. Through further mathematical analysis they have demonstrated how the system parameters can be optimized to produce accurate phase variation measurements and reduce the error in quantitative phase microscopy when a GPC-based method is utilized with post-processing. Even though the GPC method is capable of linearly mapping the input phase information to the output intensity all optically, it requires a careful mathematical analysis, a considerable amount of domain knowledge and relatively complex optical setups with post-processing to implement quantitative phase imaging. On the other hand, in our implementation the microscope is optimized for phase reconstruction as an end-to-end differentiable model. Furthermore, recently, Mengu *et al.* [21] trained diffractive deep neural networks (D2NN) on datasets such as MNIST, CIFAR10, TinyImageNet, FashionMNIST and demonstrated the all-optical phase to intensity conversion task. They tested a trained network on Pap-smear samples (monolayer of cells). It is also important to notice that the training strategy

(Ex: loss function) employed in this work is different from the phaseD2NN demonstrated in our work.

Diffractive Deep Neural Networks. D2NNs are a type of optical neural networks which is introduced by Lin *et al.* [32] as a physical mechanism to perform machine learning. They consist of a set of diffractive surfaces acting as the layers of the machine learning model. The transmission coefficients of each diffractive element act as the neurons of the network and they can be trained by modelling them with optical diffraction theory. The trained diffraction surfaces can be 3D printed and inference is done by propagating light through the D2NN. Since they work purely optically, the D2NNs has the capability of performing computations at the speed of light with no power consumption. D2NNs are utilized for various tasks such as image classification [46, 47], object detection and segmentation [48], designing task-specific optical systems [49], reconstruction of overlapping phase objects [31], and image reconstruction using holograms [50] and quantitative phase imaging [21].

B. Mathematical Modelling of Optical Systems

B.1. Learnable Fourier Filter Based Model

B.1.1 4-f system

Fig. S1 shows the system which is generally referred to as 4-f filtering architecture [51]. It is referred as 4-f because of the distance between the input plane (P_1) and the image plane (P_3) equals to $4 \times f$, where f is the focal length of the lenses. Here S is a point source, P_1, P_2, P_3 are the input plane, Fourier plane and the image plane, respectively, and L_1, L_2, L_3 are identical lenses each having a focal length f . The light from S is collimated with the lens L_1 . The input sample (P_1) which has $g(x_1, y_1)$ amplitude transmission coefficient is placed against L_1 to reduce the overall length of the system. The resultant light is then collimated with the Fourier transforming lens L_2 resulting $k_1 G\left(\frac{x_2}{\lambda f}, \frac{y_2}{\lambda f}\right)$ field at P_2 plane where G is the Fourier transform of g and k_1 is a constant. To manipulate the spectrum of input field $g(x_1, y_1)$, filter can be placed in the Fourier plane in P_2 . The transmission coefficient of the filter is given by

$$t_A(x_2, y_2) = k_2 H\left(\frac{x_2}{\lambda f}, \frac{y_2}{\lambda f}\right), \quad (1)$$

if the desired frequency domain transfer function is H and k_2 is a constant. Following lens L_3 applies the Fourier transform again on the modified spectrum to obtain the inverted modified field.

B.1.2 GPC Filter Search

The GPC introduced by Glückstad [24] is a circular filter in the Fourier plane consisting of 2 main regions, namely; the central region and the outer region. In this implementation, a phase shift is introduced by the central region of the filter which results in an interference pattern that produces a substantial contrastive image of the input phase object. This filter is characterized by four parameters $(A, B, \theta, \Delta f_r)$, and we followed two methods to select the best GPC filter parameters for a given input data distribution.

- **Method 1:** Obtained the best set of parameters through a grid search such that it achieves the maximum SSIM for a single batch of randomly selected ten images from the training set of a dataset.
- **Method 2:** Perform gradient descent using the Adam optimizer on the entire training set of a dataset to find the filter parameters.

When incorporating an SSIM loss, since the GPC method does not naturally reconstruct the output intensity ($out = |IDFT2\{DFT2\{E_{in}\} \circ G(A, B, \theta, \Delta f_r)\}|$ for an input field E_{in}) within the $[0, 1]$ range, it is required to select the appropriate scaling factor S for the filter reconstructions and incorporate it in the loss function. Therefore, we search the suitable S for the GPC reconstructions while searching for the optimal parameters that define the GPC filter, so that it can be scaled to obtain the best SSIM value as shown in eq. (2) for an intensity reconstruction within the $[0, 1]$ range. From the two methods, we select and report the GPC filter that obtains the highest SSIM on the test set of a dataset.

$$\mathcal{L}_{SSIM} = \arg \min_{A, B, \theta, \Delta f_r, S} -SSIM((S * out), gt_{angle}) \quad (2)$$

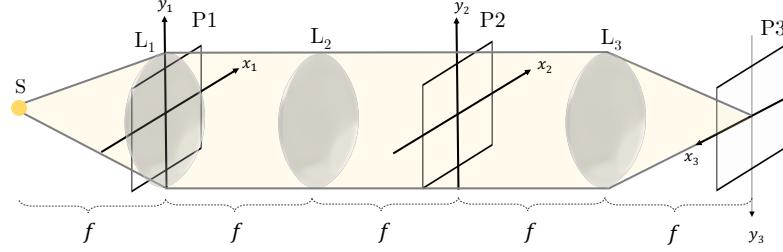


Figure S1. 4- f system

Given the size of a pixel $dx = \lambda/2$ in the spatial domain and the size of the input image $N = 256$, the frequency resolution is defined as $df = 1/(N * dx) = 1/(128\lambda)$. Table S2 demonstrates the best performing GPC and ϕ -GPC filter configurations for each dataset.

Dataset	A	B	θ	Δf_r	S
GPC Filter					
MNIST $[0, \pi]$	0.2640	0.9652	2.8833	$3df$	3.7200
MNIST $[0, 2\pi]$	0.1640	0.5360	3.2093	$3df$	8.2000
HeLa $[0, \pi]$	0.5548	0.9747	1.5568	$4.323df$	3.3567
HeLa $[0, 2\pi]$	0.5001	0.9843	1.8024	$4.323df$	3.6208
Bacteria $[0, \pi]$	0.1600	0.9600	3.0229	$2df$	8.4400
ϕ -GPC Filter					
MNIST $[0, \pi]$	1.0000	1.0000	2.6852	$6df$	0.2350
MNIST $[0, 2\pi]$	1.0000	1.0000	3.2507	$5df$	0.2050
HeLa $[0, \pi]$	1.0000	1.0000	1.5475	$4.166df$	1.6629
HeLa $[0, 2\pi]$	1.0000	1.0000	1.8291	$5df$	1.0000
Bacteria $[0, \pi]$	1.0000	1.0000	2.9502	$4.166df$	0.1934

Table S2. Best performing GPC and ϕ -GPC filter configurations

B.2. A Brief Review of Diffractive Deep Neural Networks

A D2NN comprises of a set of diffractive layers. Each diffractive element (neuron) in a diffractive layer can be considered as a secondary wave source according to the Huygens-Fresnel principle [32]. The diffraction at each neuron can be modelled using the Rayleigh-Sommerfeld diffraction formulation [51, ch. 3.5] as follows.

The Rayleigh-Sommerfeld diffraction theory describes the diffraction of light by an aperture in an infinite opaque planar screen. Consider such an aperture illuminated by a light wave which has a field $U(P_1)$ at a point $P_1 \equiv (x_1, y_1, z_1)$ on the aperture surface S as shown in Fig. S2.

The field at point $P_0 \equiv (x_0, y_0, z_0)$ after the diffraction can be given using the first Rayleigh-Sommerfeld solution to the diffraction problem, which is given as

$$U(P_0) = \iint_S U(P_1) \left(\frac{z_0 - z_1}{r^2} \right) \left(\frac{1}{2\pi r} + \frac{1}{j\lambda} \right) e^{j \frac{2\pi r}{\lambda}} dS. \quad (3)$$

Here, $r = \sqrt{(x_0 - x_1)^2 + (y_0 - y_1)^2 + (z_0 - z_1)^2}$ is the distance between the points P_1 and P_0 , λ is the wavelength of the light wave and $j = \sqrt{-1}$. This integral holds under the conditions that, U is a homogeneous scalar wave equation and it satisfies the *Sommerfeld radiation condition* which states that U must vanish at least as fast as a diverging spherical wave [51, ch. 3.4].

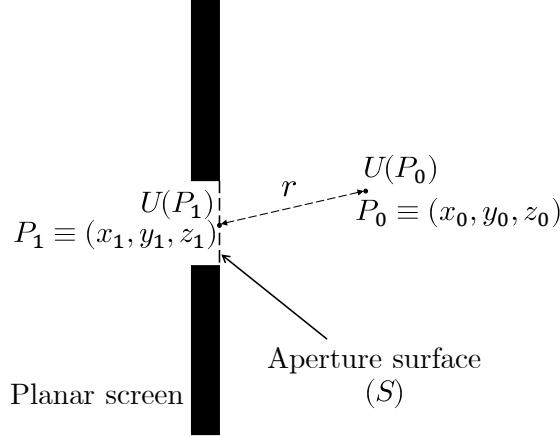


Figure S2. Diffraction by an aperture in a planar screen.

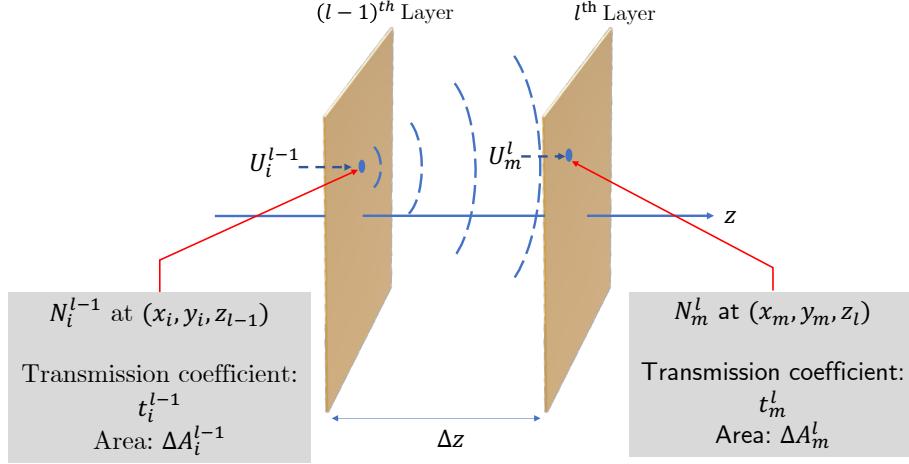


Figure S3. Diffraction of light waves by a neuron of a D2NN layer.

B.2.1 Mathematical Modelling of D2NNs

The integral in eq. (3) can be directly converted to a Riemann summation considering a discrete set of sampling points on the aperture plane. In this direct implementation, each neuron in a diffractive layer is considered as a sampling point.

As shown in Fig. S3, suppose that the field at the i^{th} neuron of the $(l-1)^{\text{th}}$ layer (N_i^{l-1}), which is located at (x_i, y_i, z_{l-1}) is given by U_i^{l-1} , and the diffraction caused at this neuron results in the field given by $U_{m,i}^l$ at N_m^l located at (x_m, y_m, z_l) . Then, $U_{m,i}^l$ is given by,

$$U_{m,i}^l = U_i^{l-1} t_i^{l-1} w_{m,i}^l \Delta A_i^{l-1}, \quad (4)$$

where

$$w_{m,i}^l = \left(\frac{\Delta z}{r_{m,i}^2} \right) \left(\frac{1}{2\pi r_{m,i}} + \frac{1}{j\lambda} \right) \exp \left\{ \left(\frac{j2\pi r_{m,i}}{\lambda} \right) \right\}. \quad (5)$$

Here, ΔA_i^{l-1} is the area of N_i^{l-1} , t_i^{l-1} is the transmission coefficient of the same neuron, λ is the wavelength of the optical wave, $\Delta z = z_l - z_{l-1}$ is the distance between the two adjacent layers along the direction of light propagation, $r_{m,i} = \sqrt{(x_m - x_i)^2 + (y_m - y_i)^2 + \Delta z^2}$, and $j = \sqrt{-1}$. The overall resulting field at N_m^l is the

superposition of all the fields resulted by the diffraction at each of the neurons at the $(l - 1)^{\text{th}}$ layer which is given by

$$U_m^l = \sum_i U_{m,i}^l. \quad (6)$$

However, direct implementation of these equations results in resource intensive computations in the simulations. A more computationally efficient modelling is done using the fact that the field resulted by an optical wave and its angular spectrum are related through the Fourier transform [51, 52]. This method is known as the angular spectrum (AS) method and it is used for simulating D2NNs in our work.

B.2.2 Angular Spectrum Method

A plane wave can be considered as a combination of a set of plane waves travelling in different directions. The complex amplitudes of these plane wave components form the angular spectrum of the given wave [52]. Consider the field of a light wave propagating in the z direction is given by,

$$U(x, y, z, t) = U(x, y, z)e^{-j2\pi f t}, \quad (7)$$

where f is the temporal frequency of the wave. Suppose that, at $z = 0$ plane, $U(x, y, z)|_{z=0} \equiv U(0)$. It can be shown that the field $U(0)$ and its angular spectrum $A(f_x, f_y, 0) \equiv A(0)$ are related by [52],

$$U(0) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} A(0) e^{j2\pi(f_x x + f_y y)} df_x df_y. \quad (8)$$

Here, $f_x = \frac{\alpha}{\lambda}$ and $f_y = \frac{\beta}{\lambda}$, where α and β are direction cosines of the plane wave components with respect to the x and y axes. Note that eq. (8) is of the form of 2-D inverse Fourier transform. Hence, the field and its angular spectrum can be considered as a Fourier transform pair. A generalized form of this result can be written as

$$U(z) \xleftrightarrow{\mathcal{F}} A(z), \quad (9)$$

where $U(z) \equiv U(x, y, z)$ is the field on $z = z$ plane and $A(z) \equiv A(f_x, f_y, z)$ is its angular spectrum. Suppose that the relationship between the angular spectrum at $z = 0$ and $z = z$ planes is given as

$$A(z) = A(0) G(z) \quad (10)$$

where, $G(z) \equiv G(f_x, f_y, z)$ is the propagation transfer function which characterises the propagation of the angular spectrum.

Electromagnetic waves satisfy the Helmholtz equation and therefore $U(z)$ should also satisfy it. This can be used to find an expression for $G(z)$. The Helmholtz equation is given as

$$(\nabla^2 + k^2) U(z) = 0, \quad (11)$$

where $k = \frac{2\pi}{\lambda}$ is the wavenumber which is the magnitude of the propagation vectors $\vec{k} = [k_x, k_y, k_z]$ of the wave components. Considering eq. (9), eq. (10), and eq. (11), it can be observed that $G(z)$ satisfies

$$\frac{d^2 G(z)}{dz^2} + k_z^2 G(z) = 0, \quad (12)$$

where

$$k_z = 2\pi \sqrt{\frac{1}{\lambda^2} - f_x^2 - f_y^2}. \quad (13)$$

An elementary solution to eq. (12) can be written as,

$$G(z) = \exp \left(j2\pi z \sqrt{\frac{1}{\lambda^2} - f_x^2 - f_y^2} \right). \quad (14)$$

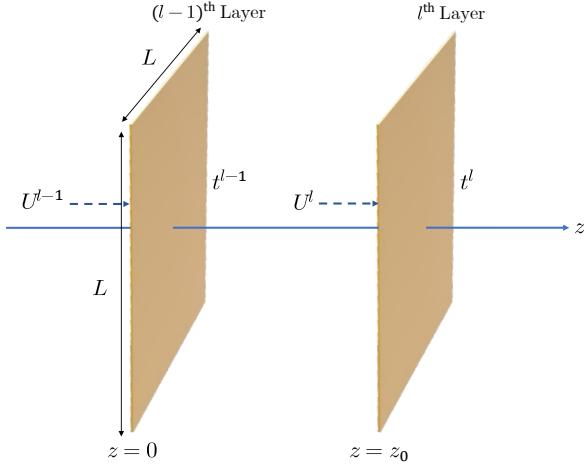


Figure S4. **Two adjacent layers of the D2NN.** The field incident on the $(l - 1)^{\text{th}}$ layer is U^{l-1} which is subjected to diffraction at the layer and results the field U^l just before the next layer.

Hence, eq. (10) can be re-written as

$$A(z) = A(0) \exp \left(j2\pi z \sqrt{\frac{1}{\lambda^2} - f_x^2 - f_y^2} \right). \quad (15)$$

This equation shows that when $f_x^2 + f_y^2 \leq \frac{1}{\lambda^2}$, the propagation of the angular spectrum along the z axis introduce a different set of phase shift to each of the components of the angular spectrum [51]. However, when $f_x^2 + f_y^2 > \frac{1}{\lambda^2}$, eq. (15) can be written as,

$$A(z) = A(0) \exp \left(-2\pi z \sqrt{f_x^2 + f_y^2 - \frac{1}{\lambda^2}} \right). \quad (16)$$

In this case, the angular spectrum is exponentially attenuated along the z axis. These wave components are known as *evanescent waves* and they do not propagate energy along the z axis.

B.2.3 Implementation of the Angular Spectrum Method

The Fourier relationship between the electric/magnetic field of a light wave and its AS can be obtained using the discrete Fourier transform (DFT) which can be computed efficiently using a fast Fourier transform (FFT) algorithm. Consider two layers of a D2NN at $z = 0$ plane ($(l - 1)^{\text{th}}$ layer) and at $z = z_0$ plane (l^{th} layer) as shown in Fig. S4. The input field at the $(l - 1)^{\text{th}}$ layer is given by $U(x, y, 0) \equiv U^{l-1}$ and the resulting field at the observation plane is given by $U(x, y, z_0) \equiv U^l$. The width and the height of both layers are L where t^{l-1} and t^l are the complex transmission coefficient matrices of the two layers.

Fig. S5 shows the computation pipeline of the output field using the AS method. Initially, the input field is sampled with a sampling interval of dx in the spatial domain. The sampled input has a size of $N \times N$ samples, where $N = L/dx$. The sampled input is then multiplied element-wise with t^{l-1} to obtain the effective field at the $(l - 1)^{\text{th}}$ layer which is given by

$$U_{\text{eff}}^{(l-1)} = U^{l-1} \circ t^{l-1}, \quad (17)$$

where \circ is the element-wise matrix multiplication.

Since the region of support of the input is bounded in the spatial domain, the region of support of its Fourier transform is unbounded and to get a more accurate AS, a higher number of samples of the Fourier transform need to be considered. Therefore, for the computation purposes a separate computational window of the size $wN \times wN$ samples is considered where w is the computation window factor. The oversampling in the Fourier domain is performed by zero padding U_{eff}^{l-1} at the boundary to match the size of the computation window. Then

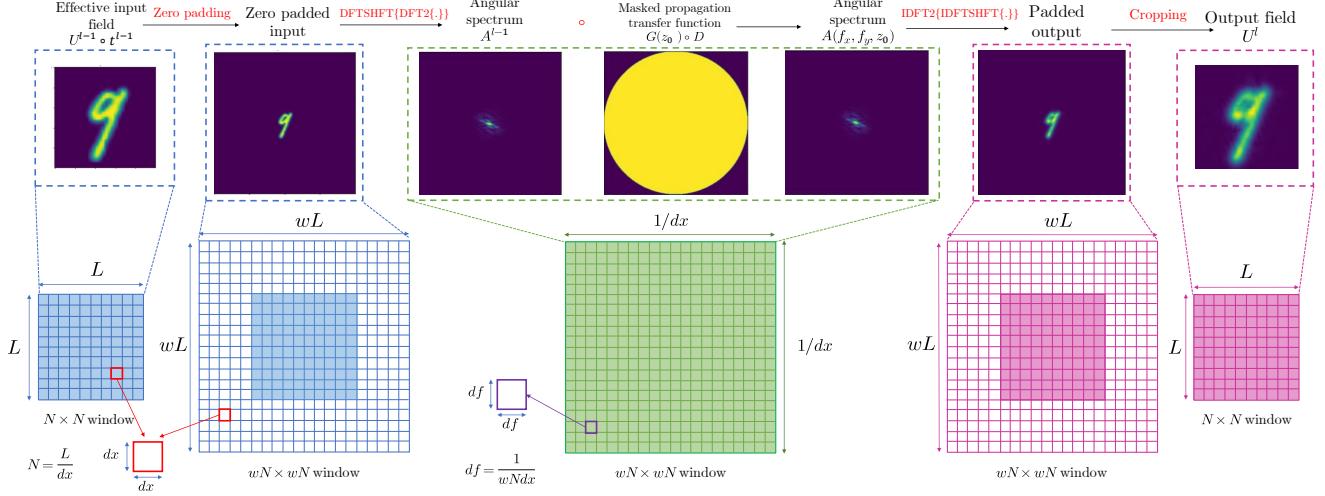


Figure S5. Computational pipeline of the AS method. In the figure $DFT2\{\cdot\}$ denotes the 2D-DFT operation, $DFTSHFT\{\cdot\}$ denotes the DFT shift operation, $IDFT2\{\cdot\}$ denotes the 2D-IDFT operation, $IDFTSHFT\{\cdot\}$ denotes the IDFT shift operation, and \circ denotes the element-wise matrix multiplication.

the 2D-DFT of the input field is taken followed by a DFT shift operation to make the center of the computation window $(0, 0)$ resulting the AS given by A^{l-1} .

A^{l-1} has spatial frequency components in the range of $\left[-\frac{f_s}{2}, \frac{f_s}{2}\right]$ in both f_x and f_y axes where $f_s = 1/dx$ is the sampling frequency in the spatial domain. Note that the gap between two samples in the angular spectrum is $df = 1/wNdx$. The propagation transfer function at $z = z_0$, $G(z_0)$ is also created in the computation window with a similar number of samples as A^{l-1} . Since the evanescent waves do not propagate energy along the z -axis, they are filtered out using a mask D . The masked propagation transfer function is given by

$$G(z_0) \circ D = \begin{cases} e^{j2\pi z_0 \sqrt{\frac{1}{\lambda^2} - f_x^2 - f_y^2}} & f_x^2 + f_y^2 \leq \frac{1}{\lambda} \\ 0 & f_x^2 + f_y^2 > \frac{1}{\lambda} \end{cases} \quad (18)$$

According to eq. (10), the AS of the field at $z = z_0$ is obtained by the element-wise multiplication of A^{l-1} and the masked propagation transfer function. Then an inverse DFT (IDFT) shift operation is performed followed by a 2D-IDFT operation to get the padded resulting field. Finally, the padding is removed to retrieve the resulting field U^l .

C. Further Experiments

C.1. Ablation Studies for CNN

We began experimenting with the number of linear convolutional layers for the CNN, where we increased the number of layers from 5-20 (results are shown in Table S3). Then we experimented with the kernel size and number of kernels per layer for the best performing CNNs selected from Table S3 for each dataset. We also experimented with the number of kernels per layer for the 5-layer CNN for each dataset to determine if a CNN with a lesser number of layers could achieve better performance. The results of this study for each dataset are depicted in Tables S4, S5, S6, S7, S8.

C.2. Generalizability

To discover how the learned optical models generalize across different datasets, we conducted a series of experiments on ϕ -LFF, ϕ -LRF, and ϕ -D2NN. In particular, we evaluate the generalizability of a given optical model by training it on a particular dataset and testing it on the rest of the datasets.

# layers	SSIM				
	MNIST [0, π]	MNIST [0, 2π]	HeLa [0, π]	HeLa [0, 2π]	Bacteria [0, π]
5	0.9980	0.7891	0.9148	0.7895	0.9796
6	0.9911	0.7348	0.9114	0.8247	0.9930
7	0.9814	0.7663	0.8995	0.6954	0.9928
8	0.9945	0.8010	0.9018	0.8537	0.9892
9	0.9847	0.7887	0.9199	0.7896	0.9874
10	0.9949	0.7630	0.9136	0.8063	0.9917
11	0.9823	0.7955	0.9295	0.8181	0.9870
12	0.9849	0.7549	0.9327	0.8297	0.9823
20	0.9867	0.7611	0.9065	0.6459	0.9854

Table S3. Effect of increasing the number of layers

# layers	Kernel size	# of kernels in each layer	SSIM
5	3	5	0.9984
5	3	10	0.9974
5	5	1	0.9949
5	5	5	0.9982

Table S4. Hyperparameter search - MNIST [0, π]

# layers	Kernel size	# of kernels in each layer	SSIM
5	3	5	0.7916
5	3	10	0.7756
8	5	1	0.7923
8	3	5	0.7681
8	5	5	0.7973

Table S5. Hyperparameter search - MNIST [0, 2π]

# layers	Kernel size	# of kernels in each layer	SSIM
5	3	5	0.9304
5	3	10	0.9329
12	5	1	0.8304
12	3	5	0.9170
12	5	5	0.9297

Table S6. Hyperparameter search - HeLa [0, π]

# layers	Kernel size	# of kernels in each layer	SSIM
5	3	5	0.8287
5	3	10	0.8051
8	5	1	0.8429
8	3	5	0.8404
8	5	5	0.7981

Table S7. Hyperparameter search - HeLa [0, 2π]

Generally, we observe that the models trained on datasets with larger phase ranges perform better for datasets with similar structural features and having phase values constrained to a smaller range. But for models trained on

# layers	Kernel size	# of kernels in each layer	SSIM
5	3	5	0.9898
5	3	10	0.9841
6	5	1	0.9890
6	3	5	0.9876
6	5	5	0.9919

Table S8. Hyperparameter search - Bacteria $[0, \pi]$

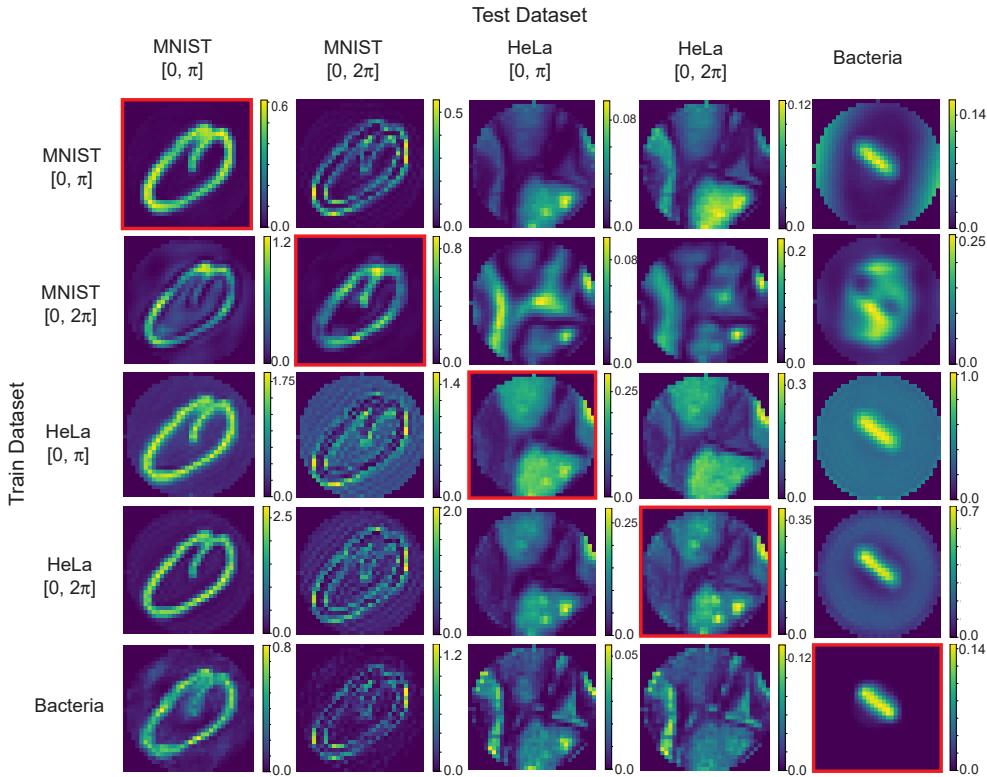


Figure S6. **Generalizability of ϕ -LFF.** The vertical axis shows the dataset used for training and the horizontal axis shows the dataset used for testing.

smaller phase ranges, this generalization does not hold.

C.2.1 ϕ -LFF Generalizability Experiments

The qualitative results in Figure S6 and the quantitative values reflected in the heatmaps S7 shows how generalizable the proposed ϕ -LFF is across different datasets.

When we train the model on MNIST $[0, 2\pi]$, it performs reasonably well on MNIST $[0, \pi]$ and performs better compared to the other three datasets. Similarly, when we train the model on HeLa $[0, 2\pi]$, it also has a good SSIM score (0.6138) on HeLa $[0, \pi]$ while having better performance compared to the other three test datasets. However, training on MNIST $[0, \pi]$ and HeLa $[0, \pi]$ datasets does not guarantee reasonable performance on MNIST $[0, 2\pi]$ and HeLa $[0, 2\pi]$ datasets respectively. This demonstrates that training on the datasets with higher phase range seems to perform better on datasets with similar structural features that are spread out in a much lower phase range. This suggests that data augmentations on the phase range could help improve performance of these models.

Furthermore, we observe that training with Bacteria dataset performs better on MNIST $[0, \pi]$ compared to other test datasets. This also can be explained with the above argument since Bacteria and MNIST both have relatively

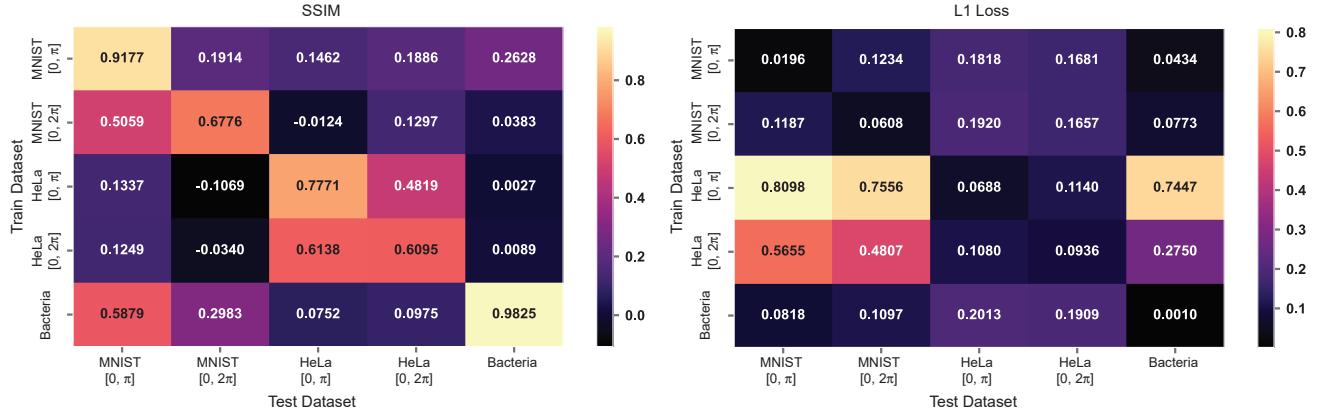


Figure S7. **Generalizability of ϕ -LFF.** The vertical axis shows the dataset used for training and the horizontal axis shows the dataset used for testing.

simple structural features that are located at the center of the field of view.

C.2.2 ϕ -LRF Generalizability Experiments

Similar to section C.2.1, we conduct experiments on the models with ϕ -LRFs. The results are shown in Fig. S8 and heatmaps S9.

Similar to previous experiments, here too, we observe similar generalization behavior. Models trained on MNIST[0, 2 π] and HeLa[0, 2 π] perform better on MNIST[0, π] and HeLa[0, π] compared to other test datasets. However, unlike in section C.2.1, we do not observe the generalizability between Bacteria and MNIST[0, π] datasets.

C.2.3 ϕ -D2NN Generalizability Experiments

Similar to the previous sections, we conduct generalizability experiments on ϕ -D2NN models. The results are shown in Fig. S10 and heatmaps S11.

Similar to previous generalizability experiments, models trained on MNIST[0, 2 π] and HeLa[0, 2 π] perform better on MNIST[0, π] and HeLa[0, π] respectively. Furthermore, the ϕ -D2NN model trained on MNIST [0, π] performs with an SSIM score of 0.6072 on the Bacteria dataset.

D. Mapping Design Configurations of the Trained Filter to the Experimental Setup

As explained in Section 2.3 our learned filters span a region of 256×256 pixels for an input phase image size of 32×32 . However, in the experimental setup the input would not necessarily span within a spatial dimension of 32×32 pixels. Therefore, in the experimental setup, based on the input size, the filter implemented in the SLM would have to be resized to properly follow the simulated conditions and match the experimental dimensions for reconstruction. The effect of upsampling the learned ϕ -LFF, ϕ -LRF, and ϕ -GPC filter to match the experimental setup is demonstrated in Fig. S12. The digit is clearly visible for the ϕ -LRF (Fig. S12-C1) and ϕ -LFF (Fig. S12-C2), however, the digits appear to have a low brightness and/or background artifacts compared to the reconstructions under the training conditions (Fig. S12-A1 and A2 respectively). This suggests that LFFs and LRFs may be sensitive to interpolation when resizing the filters. This can be alleviated by either training or finetuning the LFF/LRF in a simulated environment that matches the experimental setup, or improving the robustness by learning a filter on an augmented dataset that captures some of the experimental setup imperfections.

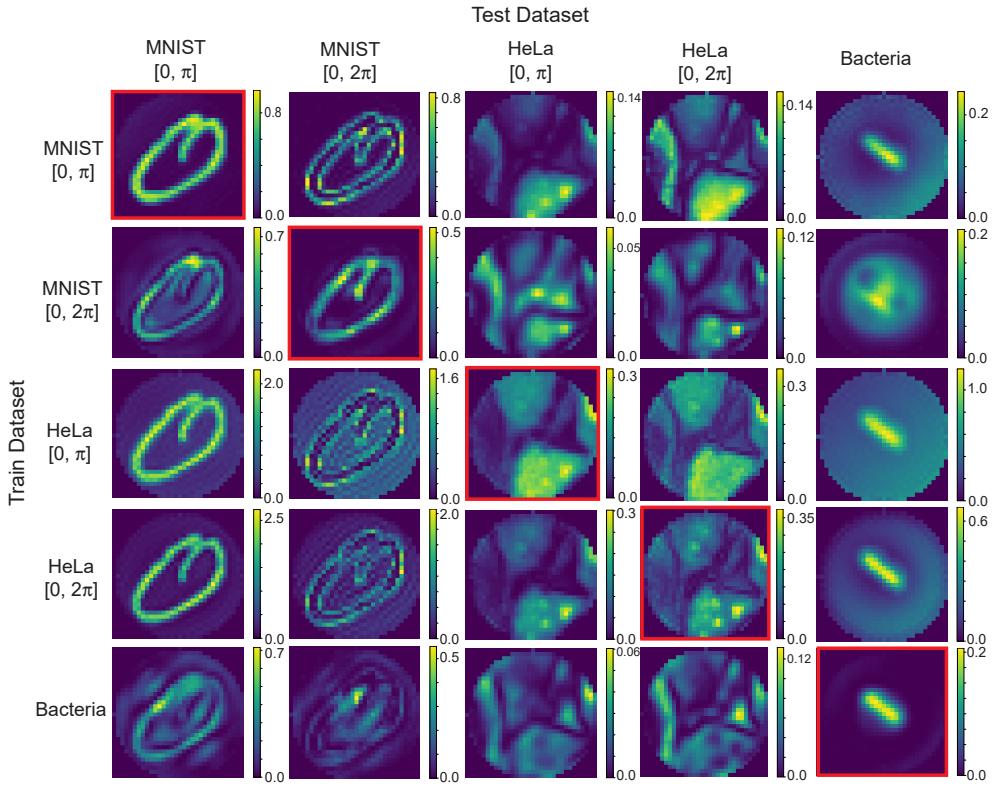


Figure S8. **Generalizability of ϕ -LRF.** The vertical axis shows the dataset used for training and the horizontal axis shows the dataset used for testing.

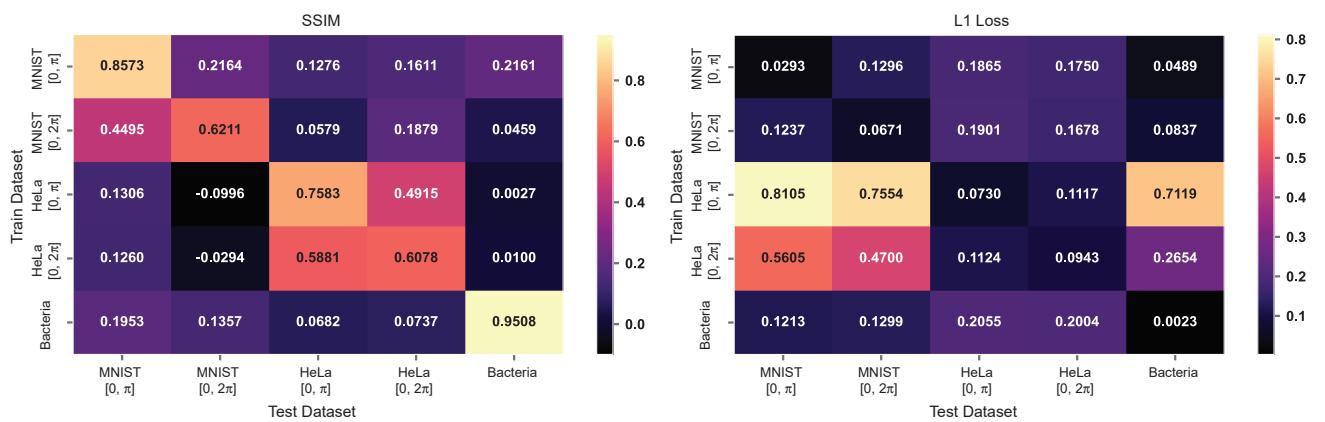


Figure S9. **Generalizability of ϕ -LRF.** The vertical axis shows the dataset used for training and the horizontal axis shows the dataset used for testing.

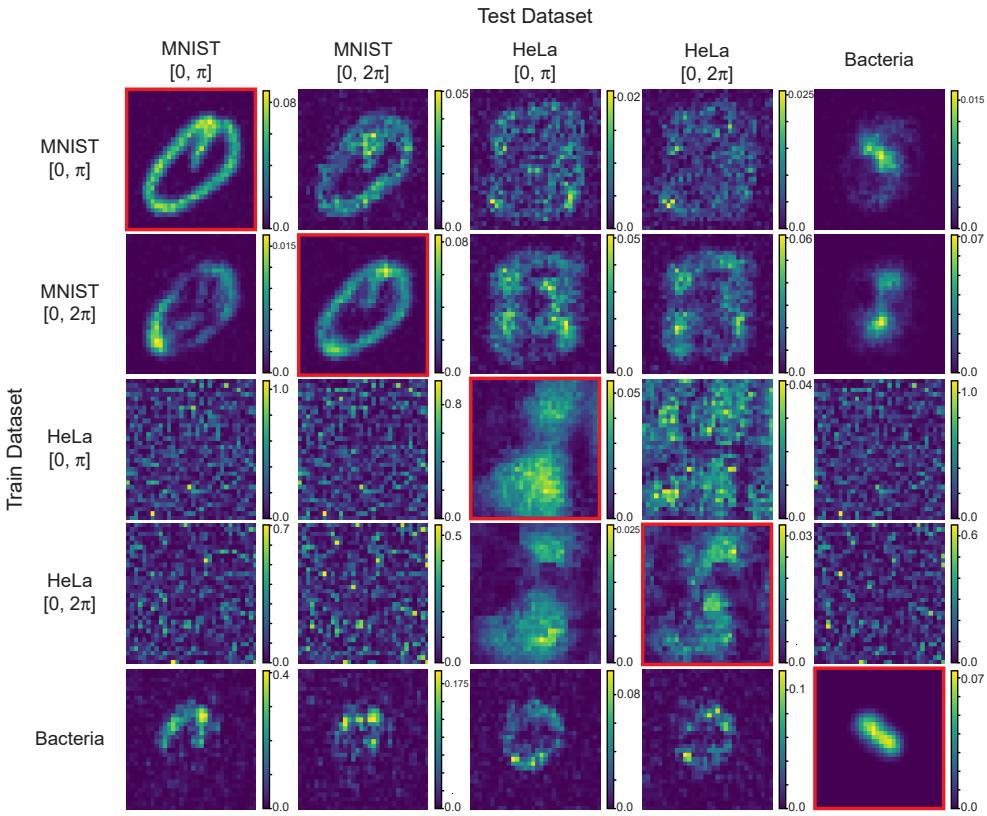


Figure S10. **Generalizability of ϕ -D2NN.** The vertical axis shows the dataset used for training and the horizontal axis shows the dataset used for testing.

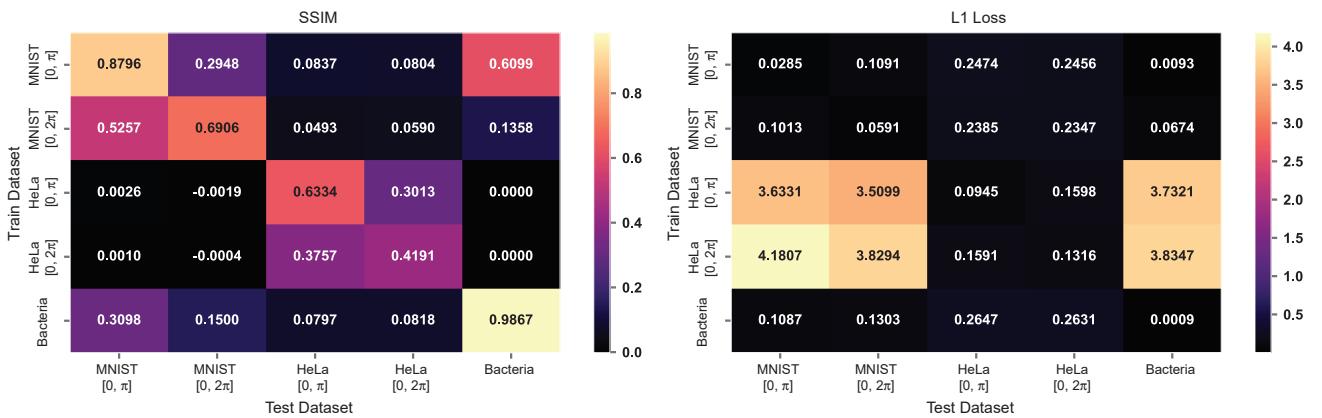


Figure S11. **Generalizability of ϕ -D2NN.** The vertical axis shows the dataset used for training and the horizontal axis shows the dataset used for testing.

Method	Loss	MNIST [0, π]		MNIST [0, 2 π]		HeLa [0, π]		HeLa [0, 2 π]		Bacteria [0, π]	
		SSIM \uparrow	L1 \downarrow	SSIM \uparrow	L1 \downarrow	SSIM \uparrow	L1 \downarrow	SSIM \uparrow	L1 \downarrow	SSIM \uparrow	L1 \downarrow
C-CNN	\mathcal{L}_{LT}	0.9982	0.0041	0.7913	0.0539	0.9417	0.0248	0.8619	0.0485	0.9938	0.0008
ϕ -LFF	\mathcal{L}_ϕ	0.9186	0.0201	0.6778	0.0609	0.7300	0.0914	0.6207	0.1034	0.9825	0.0007
	\mathcal{L}_{LT}	0.9177	0.0196	0.6777	0.0608	0.7771	0.0688	0.6096	0.0937	0.9825	0.0010
ϕ -LRF	\mathcal{L}_ϕ	0.8583	0.0292	0.6219	0.0671	0.7212	0.0932	0.6174	0.1039	0.9504	0.0023
	\mathcal{L}_{LT}	0.8573	0.0293	0.6211	0.0671	0.7583	0.0730	0.6078	0.0943	0.9508	0.0023
ϕ -GPC*	–	0.8466	0.0430	0.4191	0.0982	0.7297	0.0755	0.5184	0.1100	0.9479	0.0036
	\mathcal{L}_ϕ	0.8497	0.0337	0.5713	0.0736	0.4335	0.1371	0.2532	0.1711	0.9861	0.0009
ϕ -D2NN	\mathcal{L}_ϕ	0.8796	0.0285	0.6906	0.0591	0.6334	0.0945	0.4191	0.1316	0.9867	0.0009
	\mathcal{L}_{LT}	0.9225	0.0202	0.6801	0.0608	0.7301	0.0914	0.6202	0.1033	0.9818	0.0010
LFF	\mathcal{L}_ϕ	0.9205	0.0196	0.6814	0.0608	0.7783	0.0688	0.6078	0.0937	0.9823	0.0010
	\mathcal{L}_{LT}	—	0.9036	0.0320	0.4364	0.0951	0.7786	0.0748	0.5509	0.1091	0.9600
D2NN	\mathcal{L}_ϕ	0.8325	0.0357	0.5350	0.0776	0.4548	0.1302	0.5592	0.1321	0.9901	0.0007
	\mathcal{L}_{LT}	0.9433	0.0201	0.7703	0.0486	0.6655	0.0936	0.4942	0.1274	0.9926	0.0007

* Output is scaled for each dataset (see supplementary section B.1.2)

Table S9. **Overall quantitative results.** Performance of the Complex-valued CNN (C-CNN), GPC method, the learnable Fourier filter (LFF), and the D2NN are shown for each dataset. The best-performing cases in each performance metric for each dataset are in bold.

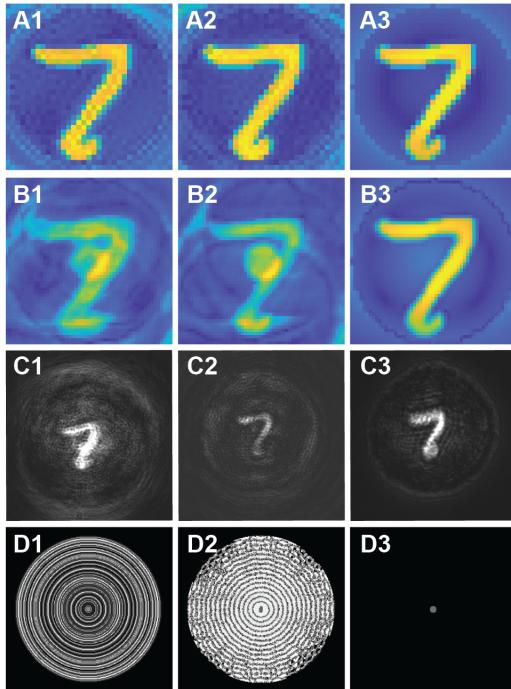


Figure S12. **Model mismatch due to up-sampling.** (A1-3) The numerical predictions under the training conditions (i.e., using 256x256 pixel filter) for ϕ -LRF, ϕ -LFF, and ϕ -GPC filters. (B1-3) The corresponding numerical predictions with the up-sampled filters matching the experimental setup. (C1-3) The corresponding experimental results with the upsampled filters. (D1-3) The corresponding up-sampled filters: D1 – ϕ -LRF, D2 – ϕ -LFF, and D3 – ϕ -GPC.

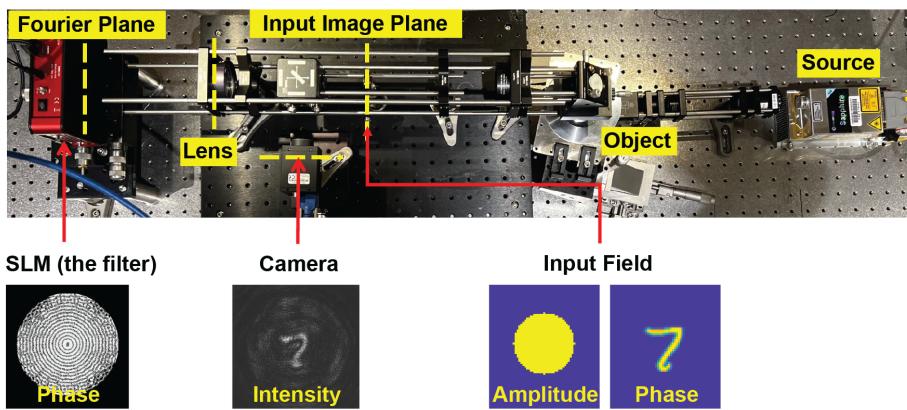


Figure S13. A photograph of the experimental setup.