Research Article

Physics-informed neural networks for diffraction tomography

Amirhossein Saba ⬤,* Carlo Gigli ⬤,† Ahmed B. Ayoub,† and Demetri Psaltis
École Polytechnique Fédérale de Lausanne, Optics Laboratory, Lausanne, Switzerland

Abstract. We propose a physics-informed neural network (PINN) as the forward model for tomographic reconstructions of biological samples. We demonstrate that by training this network with the Helmholtz equation as a physical loss, we can predict the scattered field accurately. It will be shown that a pretrained network can be fine-tuned for different samples and used for solving the scattering problem much faster than other numerical solutions. We evaluate our methodology with numerical and experimental results. Our PINNs can be generalized for any forward and inverse scattering problem.

Keywords: deep learning; physics-informed neural networks; scattering; three-dimensional imaging; optical diffraction tomography.

1 Introduction

Optical diffraction tomography (ODT) is an imaging technique for extracting the three-dimensional (3D) refractive index distribution of a sample, e.g., a biological cell using multiple two-dimensional (2D) images acquired at different illumination angles. The refractive index of the sample provides useful morphological information, making ODT an interesting approach for biological applications.[1–3] The conventional method to reconstruct the 3D refractive index from multiple projections was proposed by Wolf in 1969.[4] In Wolf's method, the 3D Fourier domain of the refractive index is filled with 2D Fourier transforms of the measured scattered fields. However, due to the finite number of projections, the limited numerical aperture of the optical system, and the single scattering approximation, missing frequencies are causing some distortion, elongation, and underestimation of the reconstructed refractive index.[5]

In the last several years, many different iterative methods have been proposed to reconstruct accurate refractive indices from ill-posed measurements.[5–9] The main idea behind these iterative approaches is to use a forward model that predicts the candidate projections for the current estimation of the refractive index in that iteration, compare this 2D prediction with the

experimental measurements of the projection as a loss function, and update the estimation of the refractive index upon the minimization of this loss function plus possibly any other prior knowledge, e.g., sparsity conditions. Importantly, such an iterative scheme requires an analytical/semi-analytical model to backpropagate the computed loss and update the estimation of the refractive index. This precludes the use of common mesh-based numerical solvers such as finite difference and finite element methods (FEMs). In Ref. 5, the authors use a linear (single-scattering) forward model, in the approach proposed in Refs. 6–8, referred to as learning tomography. The forward model is the beam propagation method (BPM) and, in Ref. 9, the authors resort to the Lippmann–Schwinger equation. The forward models used in these iterative solutions either have inaccuracies in the cases of multiple-scattering and high-contrast samples, or they are computationally demanding. As a result, presenting a fast, accurate, and differentiable forward model is necessary to be used in iterative ODT. Physics-informed neural networks (PINNs) can be a good candidate for solving forward scattering problems and for being used in iterative tomographic reconstruction.

PINNs have recently gotten intense research attention for solving different complex problems in physics.[10,11] These networks use physics laws as the loss function instead of the data-driven loss functions. In conventional supervised deep learning, a large dataset of labeled examples is used for the

---

training process: by comparing the known ground truth with the predictions from a deep multi-layer neural network, one can construct a loss function and tune the parameters of the network to solve complex physical problems. Different examples of these data-driven neural networks are proposed for optical applications such as resolution enhancement,[12] imaging through multi-mode fibers,[13,14] phase retrieval,[15] ODT,[16] and digital holography.[17,18] In these networks, the knowledge acquired by the network strongly depends on the statistical information provided in the dataset, and training such a network requires access to a large dataset. In contrast, PINNs directly minimize the physical residual from the corresponding partial differential equation (PDE) that governs the problem instead of extrapolating physical laws after going through a large amount of examples. In the pioneering approach proposed by Lagaris et al.,[19] the neural network maps independent variables, such as spatial and time coordinates, to a surrogate solution of a PDE. By applying the chain rule, e.g., through auto-differentiation integrated in many deep-learning packages, one can easily extract the derivatives of the output fields with respect to the input coordinates and consequently construct a physics-based loss.[20] The correct prediction can be therefore retrieved by minimizing the loss with respect to the network weights. This approach has been used to solve nonlinear differential equations,[21–24] to realize the forward model in the inverse design of optical components,[25] and to extract material parameters in near-field microscopy.[26]

Having the independent variables of PDE as the input of the neural network limits the use of PINNs when fast inference is required. For the example of optical scattering, the neural network should be trained for each refractive index distribution separately. A different idea was proposed recently in Ref. 27 to solve Maxwell's equations for micro lenses with different permittivity distributions. The calculation of physical loss, in this case, is based on the finite difference scheme, and in contrast to the previous approach that is trained for a single example, this model proved to be well-suited for cases in which fast inference is required. However, such a PINN was only demonstrated to work for homogeneous 2D samples.

In this paper, we extend this idea for inhomogeneous and 3D cases and present a MaxwellNet that is able to solve different forward scattering problems, such as light scattering from biological cells. In the first part of the work, we train MaxwellNet for 2D digital phantoms and show how this pretrained network can be fine-tuned to predict light scattering from more complex and experimentally relevant samples, in our case, HCT-116 cells. We benchmark the performance of MaxwellNet in solving scattering problems for 2D and 3D objects. Next, we demonstrate that such PINN can be efficiently used to invert the scattering problem through an iterative scheme and improve the results of conventional ODT. We first demonstrate the reconstruction of the refractive index distribution from synthetic data and then we validate the technique with experimental measurements of scattering from polystyrene microspheres.

2 Methodology

The main idea of our work, shown in Fig. 1, consists of two blocks. The first, MaxwellNet, is a neural network that takes as an input the refractive index distribution $n(r)$ and predicts the scattered field U^s. Its structure is based on the U-Net architecture,[28] and the training is performed on a large dataset of digital phantoms using a physics-defined loss function. Then, this network is used as a forward model in a second optimization task that compares the fields predicted by MaxwellNet for a candidate refractive index (RI) distribution with the ground truth projections, e.g., computed numerically or evaluated experimentally, and updates $n(r)$ up to convergence.

2.1 Forward Model: MaxwellNet

In this section, we describe the implementation of a PINN that predicts the scattered field for a known input RI distribution. For the sake of simplicity, we first describe the method for the 2D case, but we will show the extension to 3D in the following. In this case, MaxwellNet takes as an input the RI distribution as a discrete array of shape $N_x \times N_z \times 1$ and we do expect an output
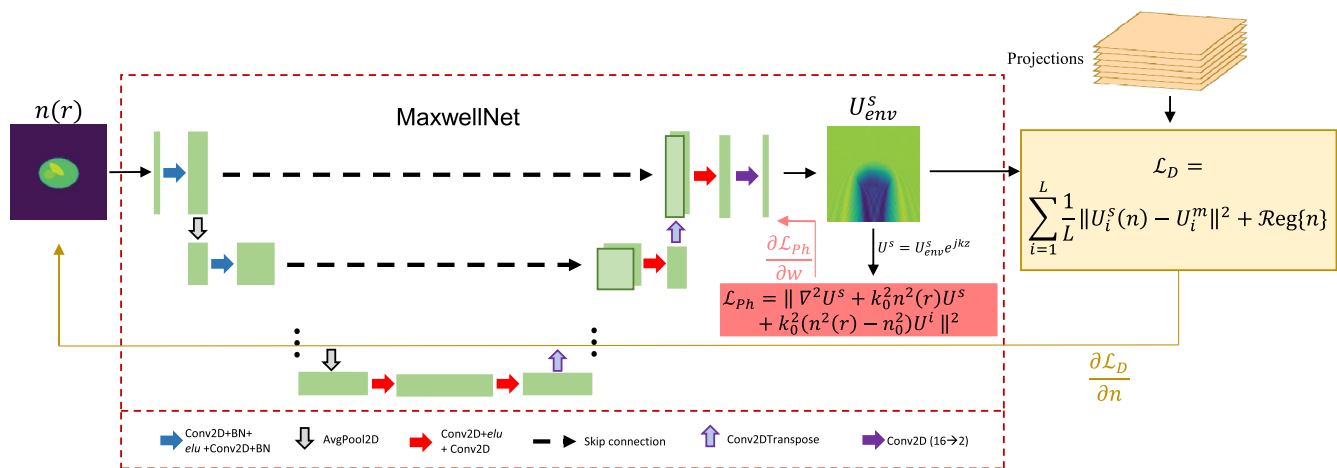

Fig. 1 Schematic description of MaxwellNet, with U-Net architecture, and its application for tomographic reconstruction. The input is a refractive index distribution and the output is the envelope of the scattered field. The output is modulated by the fast-oscillating term $e^{jk_0 n_0 z}$ to compute the physics-informed loss for tuning the weights. For tomographic reconstruction, we minimize a data-driven loss based on the difference between measured and calculated projections using MaxwellNet. A regularization term can be added to improve the reconstruction.

with size $N_x \times N_z \times 2$, where the two channels correspond to the real and imaginary parts of the complex field. Among all the available architectures, the choice of U-Net appears favorable as we do expect to embed the latent features of the RI distributions in a lower dimensional space through consecutive 2D convolutions and then retrieve the complex electromagnetic field in the same spatial points through the decoding step. A similar architecture was also proven to provide good accuracy for the computation of the scattered field from micro lenses.[27] We implement the present network in TensorFlow 2.6.0. For each step in the encoder, we use two Conv2D layers, each followed by batch-normalization and the *elu* activation function. A total number of five layers are adopted to encode the information and each one is terminated with average pooling to reduce the dimension. The maximum number of channels that we get in the latent space is 512. On the decoder side, we used transposed convolutional layers to the output with the size $N_x \times N_z \times 2$ (or $N_x \times N_y \times N_z \times 2$ in the 3D case). It should be noted that we also use residual skip connections from the encoder branch. In common data-driven training, we would tune the weights of this network by minimizing the difference between predictions and ground truth data computed with numerical solvers, in turn requiring a large database of simulations and consequently a massive computational cost. Here, we do not provide input-output pairs; instead we train the network by requiring that the Helmholtz equation is satisfied on the predicted field. To speed up the training and improve performances, we require the network to predict the slowly varying envelope of the scattered field U^s_{env} being the scattered field obtained after demodulated by the fast-oscillating component along the propagation direction $U^s = U^s_{env} e^{jk_0 n_0 z}$. We define a physics-informed loss function to be minimized by updating the weights of the network:

$$\mathcal{L}_{Ph} = \sum_r \frac{1}{N} \|[\nabla^2 + k_0^2 n^2(r)]U^s + k_0^2[n^2(r) - n_0^2]U^i\|^2, \qquad (1)$$

where k_0 is the wave number which is $k_0 = 2\pi/\lambda$ and $\lambda = 1.030 \ \mu$m is the wavelength. $n(r)$ is the RI distribution and n_0 is the RI of the background medium. The summation in Eq. (1) is done over the pixels of the computational domain and N is the number of pixels. To implement the Laplacian in Eq. (1), we follow the Yee grid finite difference scheme, computing the derivative of the variables by 2D convolutions with a 5×5 kernel.[29] Additionally, light scattering is by defining an open boundary problem. To satisfy the Sommerfeld radiation condition and confine the problem in a finite space we use a stretched-coordinate perfectly matched layer (PML)[30] at the edges of the simulation domain by introducing a complex coordinate transformation $[x \rightarrow x + if(x)]$ when calculating the derivatives inside the PML region. We use the gradient of the so-computed physical loss function to update the weights of the neural network, w through an Adam optimizer:

$$w \rightarrow w - \gamma_{Ph} \frac{\partial \mathcal{L}_{Ph}}{\partial w}. \qquad (2)$$

When we train MaxwellNet for a class of samples, it can accurately calculate the field for unseen samples from the same class. However, the key point to mention is that if we want to use MaxwellNet for a different set of RI distributions, we can fix some of the weights and adjust only a part of the network for the new dataset instead of re-starting the training from scratch. This process, referred to in the following as fine-tuning, is much faster than the original training of MaxwellNet. We will elaborate and discuss this interesting feature in Sec. 3.

It should be mentioned that we train MaxwellNet based on the Helmholtz equation with a scalar field approximation, as described in Eq. (1). The scalar approximation allows us to have a network with 2-channel output, representing the real and imaginary parts of the scalar field. We can also consider the full-vectorial Helmholtz equation where we need a larger network with 6-channel output to represent the real and imaginary parts of the three components of the field vector. However, the depolarization term can be neglected for samples with low refractive index gradients,[31,32] allowing us to have a MaxwellNet with fewer parameters and the scalar Helmholtz equation as the loss function.

2.2 Optical Diffraction Tomography Using MaxwellNet

Once MaxwellNet has been trained on a class of RI distributions, it can be used to rapidly backpropagate reconstruction errors with an approach similar to learning tomography.[6] Let us assume that we measure L projections U^m_i, with $i = 0, \ldots, L$, from an unknown RI distribution $\overline{n}(x, z)$ for different rotational angles. From these data, we can reconstruct a first inaccurate candidate $n(x, z)$ through the Wolf's transform using the Rytov approximation. Then, we need to calculate the projections by MaxwellNet for different illumination angles. To implement illumination angle rotation, we can geometrically rotate $n(x, z)$ based on the corresponding illumination angle and calculate the scattered field for the rotated refractive index. By feeding MaxwellNet with $n_i(x, z) = \mathcal{R}_i\{n(x, z)\}$, where $\mathcal{R}_i$ is the image rotation operator that corresponds to the i-th projection, we predict the complex scattered fields U^s_i for the same L angles. Consequently, we can construct a data-driven loss function $\mathcal{L}_D$ given by the difference $\|U^s_i - U^m_i\|^2$ plus any additional regularizer, compute its gradient through auto-differentiation, update $n(x, z)$, and iterate up to convergence:

$$\mathcal{L}_D = \sum_{i=1}^L \frac{1}{L} \|U^s(\mathcal{R}_i\{n\}) - U^m_i\|^2 + \mathcal{R}eg\{n, U^s(n)\}, \qquad (3)$$

$$n \rightarrow n - \gamma_D \frac{\partial \mathcal{L}_D}{\partial n}. \qquad (4)$$

Also, in this case, we use an Adam optimizer for updating the RI values. The regularizer in Eq. (3) consists of three parts: a total-variation (TV), a non-negativity, and physics-informed terms, $\mathcal{R}eg\{n, U^s_i(n)\} = \lambda_{TV}\mathcal{R}_{TV}(n) + \lambda_{NN}\mathcal{R}_{NN}(n) + \lambda_{Ph}\mathcal{L}_{Ph}(n, U^s)$. The TV regularizer helps smooth the RI reconstruction and the non-negativity regularizer adds the prior information that $n(x, z)$ should be larger than the background refractive index:

$$\mathcal{R}_{TV}(n) = \sum_r \sqrt{|\nabla_x n(r)|^2 + |\nabla_y n(r)|^2 + |\nabla_z n(r)|^2}, \qquad (5a)$$

$$\mathcal{R}_{NN}(n) = \sum_r \min[n(r) - n_0, 0]^2. \qquad (5b)$$

Importantly, we have to remark that MaxwellNet is trained for a specific dataset and accurately predicts the scattered field for RI distributions that are not too far from this set. To take into account this effect, we add the physics-informed loss to the regularizer. This further correction term helps to find RI values in a way that MaxwellNet can predict the scattered field for them correctly. In contrast to TV and non-negativity constraints that are used due to the ill-posedness of the ODT problem, the physics-informed regularizer is necessary in our methodology to ensure that the index distributions remain within the domain in which MaxwellNet has been trained.

The key advantages of using MaxwellNet with respect to other forward models are: differently from BPM, it can accurately calculate field scattering, considering reflection, multiple-scattering, or any other electromagnetic effects;[5–8] once trained, the field computations are performed in milliseconds, much faster than the Lipmann–Schwinger model; and finally, the data-driven error in Eq. (3) can be easily backpropagated differently from commercially available full-vectorial solvers. We discuss the reconstruction results and compare them with other methods in Sec. 3.2.

3 Results and Discussion

3.1 MaxwellNet Results

In this section, we evaluate the performance of MaxwellNet for the prediction of the scattered field from RI structures such as biological cells. First, we check the performance on a 2D sample assuming that the system is invariant along the y axis. The number of pixels for our model is $N_x = N_z = 256$ for both the x and z directions, and their size is $dx = 100$ nm. We also use PML with a thickness of 1.6 μm at the edges of our computational domain. We create a dataset of digital cell phantoms and divide it into the training and testing sets. MaxwellNet has $\sim 5.9 \times 10^6$ parameters to train and we use the Adam optimizer with a learning rate of 1×10^{-4} and batch training. The details about the dataset and training and validation losses are discussed in Appendix B. We train and test MaxwellNet in TensorFlow 2.6.0 on a desktop computer (Intel Core i7-9700K CPU, 3.6 GHz, 64 GB RAM, GPU GeForce RTX 2080Ti).

In Figs. 2(a) and 2(b), we choose two random examples of the digital phantoms in the test set (which is not seen by the network during the training). For each test case, in the second and third rows, we present the prediction of the envelope of the scattered field by the network, and we compare it with the result achieved by the FEM using COMSOL Multiphysics 5.4. We can see a very small difference between the results of MaxwellNet and COMSOL, which we attribute to discretization errors. There are different schemes of discretization in two methods that can cause such differences. To quantitatively evaluate the performance of MaxwellNet, we define the relative error of MaxwellNet with respect to COMSOL as

$$\text{Error} = \frac{\int \|U_{\text{MaxwellNet}}(r) - U_{\text{COMSOL}}(r)\|^2 dr}{\int \|U_{\text{COMSOL}}(r)\|^2 dr}, \quad (6)$$

where $U_{\text{MaxwellNet}}$ and U_{COMSOL} are the total fields calculated with MaxwellNet and COMSOL. The integration is done excluding the PML regions. The calculated relative errors for test case 1 and test case 2 in Fig. 2 are 4.1×10^{-2} and 4.6×10^{-2}, respectively.

It should be noted that once MaxwellNet is trained, the scattered field calculation is much faster than numerical techniques such as FEM. We present a time comparison in Table 1. For the test phantoms in Fig. 2, it took 17 ms for MaxwellNet in comparison with 13 s for COMSOL, meaning three orders of magnitude acceleration.

Furthermore, performing a physics-based instead of direct data-driven training holds promises for exploiting the advantages of transfer learning.[33] Maxwell equations are general but having a neural network that predicts the scattered field for any class of RI distribution in milliseconds with a negligible physical loss is usually unfeasible. Most of the previous PINN studies for solving partial differential equations are trained for one example, and they will work for that specific example. In our case, the U-Net architecture proved to be expressive enough to predict the field for a class of samples. However, if we use MaxwellNet for inference on an RI distribution completely uncorrelated with the training set, the accuracy drops. To evaluate the MaxwellNet extrapolation capability, we considered the model trained on phantoms samples in Fig. 2 and use it for inference on HCT-116 cancer cells. The comparison between MaxwellNet and COMSOL is shown in Fig. 2(c). The input of the network is a 2D slice of the experimentally measured HCT-116 cell in the plane of the best focus. The discrepancy between MaxwellNet and COMSOL is due to the fact that the former does not see examples of such RI distributions during the training. As a result, if we require accurate results for a new set of samples with different features, we have to re-train MaxwellNet for the new dataset, which would take a long time as shown in Table 1. However, it turns out that learning a physical law, as Maxwell equations, even though on a finite dataset, is better suited than data-driven training for transfer learning on new batches. Indeed, we can use the pretrained MaxwellNet on digital phantoms and fine-tune some parts of the network for HCT cells achieving good convergence in a few epochs. In this example, we create a dataset of 136 RI distributions of HCT-116 cancer cells and divide them into the training and validation sets. Some examples of the HCT-116 refractive index dataset are shown in Appendix B. A wide range of cells with different shapes are included in the dataset. We have single cancer cells, such as shown in Fig. 2(c), examples of cells in the mitosis process, or examples with multiple cells. In this case, we freeze the weights of the encoder part and fine-tune the decoder with the new dataset. We can see in Fig. 2(d) that after this correction step, the calculated field is much more accurate. As can be seen in Table 1, the fine-tuning process is two orders of magnitude faster than a complete training from scratch.

The 2D case is helpful for demonstrating the method and rapidly evaluating the performances. Nevertheless, full 3D fields are required for many practical applications. We can straightforwardly recast MaxwellNet in 3D using arrays of size $N_x \times N_y \times N_z \times 1$ as inputs of the network and requiring an $N_x \times N_y \times N_z \times 2$ output, with the two channels corresponding to the real and imaginary parts of the envelope of the scattered field. In this case, the network consists of Conv3D, AveragePool3D, and Conv3DTranspose layers instead of 2D counterparts. As a benchmark test, we created a dataset of 3D phantoms with 200 examples (180 for training and 20 for testing). The computational domain is defined with $N_x = N_y = N_z = 64$, $dx = 100$ nm, and PML thickness of 0.8 μm. To show the proof of concept of 3D MaxwellNet with limited computational resources, we used a lower number of pixels per dimension with
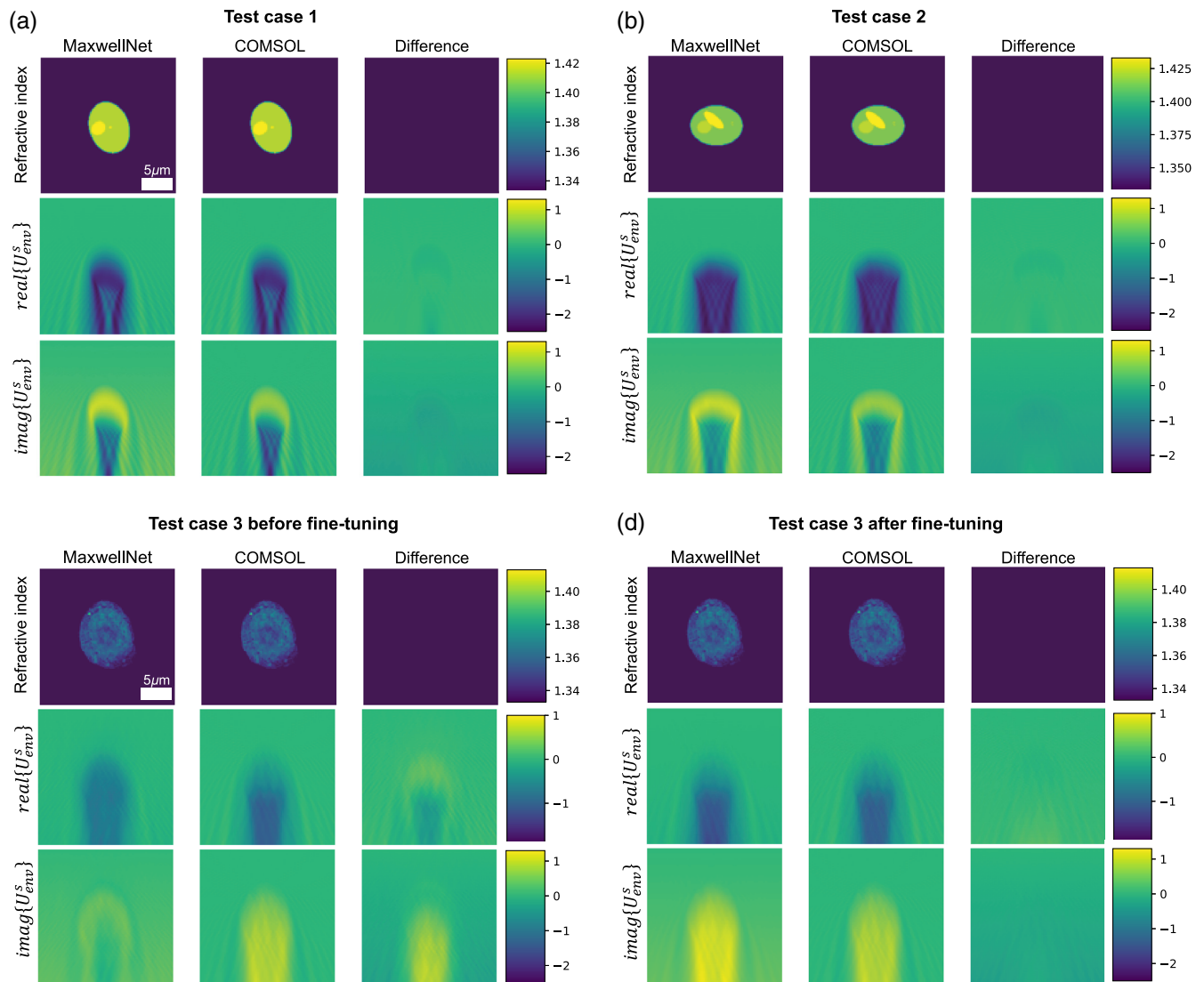
Fig. 2 Results of MaxwellNet and its comparison with COMSOL. (a) and (b) Two test cases from the digital phantom dataset and the prediction of the real and imaginary parts of the envelope of the scattered fields using MaxwellNet, COMSOL, and their difference. (c) Scattered field predictions from the network trained in (a) and (b) for the case of an experimentally measured RI of HCT-116 cancer cells and comparison with COMSOL. The difference between the two is no longer negligible. (d) Comparison between MaxwellNet and COMSOL after fine-tuning the former for a set of HCT-116 cells. MaxwellNet predictions reproduces much more accurate results after fine-tuning.

Table 1 Computation time comparison.

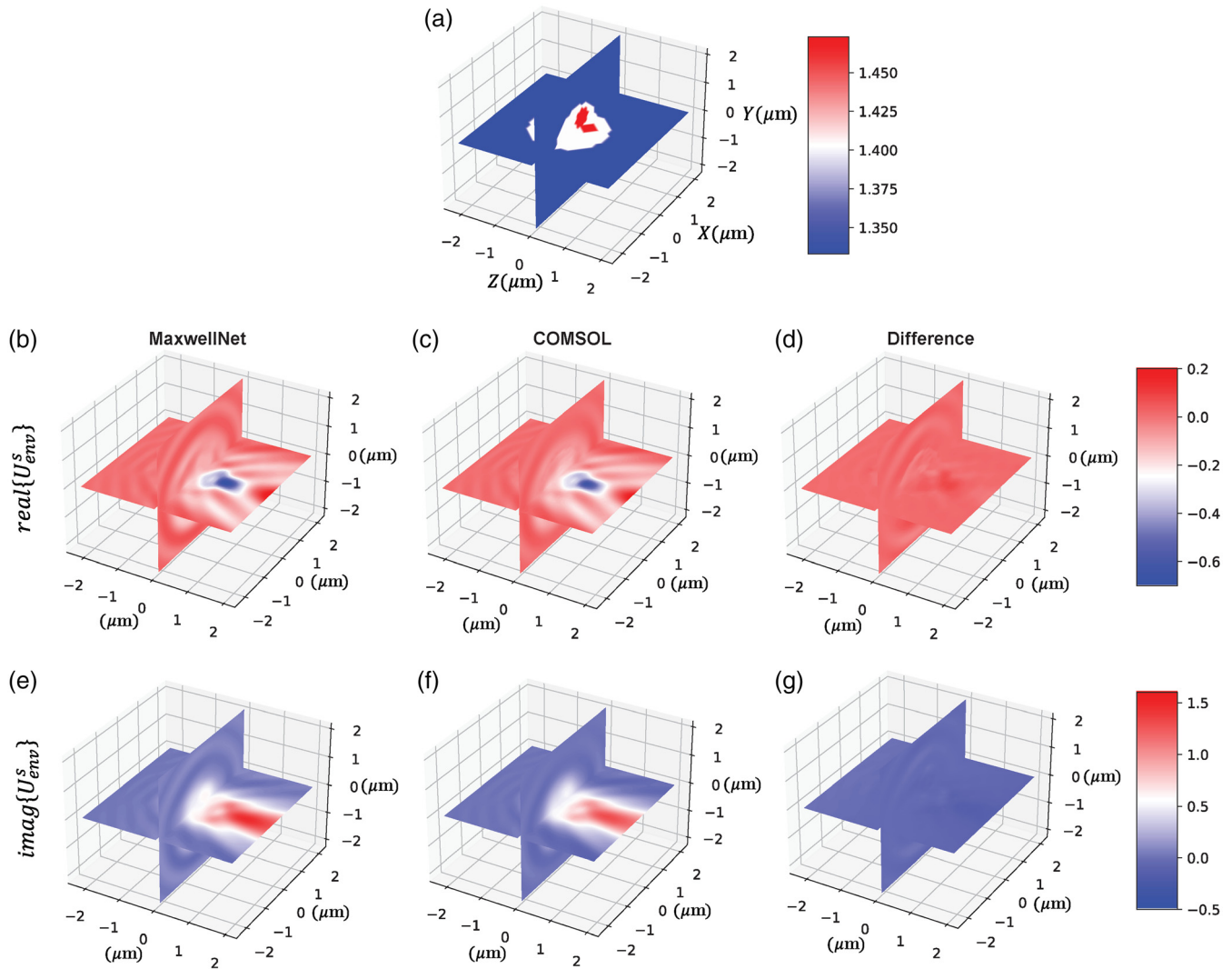| Dataset	2D phantoms (training)	2D HCT-116 (fine-tuning)	3D phantoms (training)
Training details	2700 samples 5000 epochs	122 samples 600 epochs	180 samples 5000 epochs
MaxwellNet training/fine-tuning	30.5 h	0.18 h	15.5 h
MaxwellNet inference	17 ms	17 ms	44.9 ms
COMSOL	13 s	13 s	2472 s

Fig. 3 Results of 3D MaxwellNet and its comparison with COMSOL. The RI distribution is shown in (a). The real part of the envelope of the scattered field calculated by 3D MaxwellNet is shown in (b), calculated by COMSOL in (c), and their difference in (d). The imaginary part of the envelope of the scattered field calculated by 3D MaxwellNet, COMSOL, and their difference are presented in (e)–(g), respectively.

respect to the 2D case, keeping the pixel size, dx, the same to have an accurate finite difference calculation. As a result, we have a limited computational domain size, which can be improved using more powerful resources.

The 3D version of MaxwellNet has $\sim 1.72 \times 10^7$ parameters. We use the Adam optimizer (learning rate $= 1 \times 10^{-4}$) and a batch size of 10. The results of the predicted field for an unseen example and its comparison with COMSOL are shown in Fig. 3. We can see that MaxwellNet performs as well as COMSOL in field calculation. The quantitative error described in Eq. (6) is 3.4×10^{-3} for the 3D example of Fig. 3. There are some marginal differences due to different discretization schemes. However, we can see in Table 1 that MaxwellNet is about 50,000 times faster than COMSOL in predicting fields (44.9 ms versus 41.2 min). This result and the significant efficiency in the computation time highlight the MaxwellNet potential for the calculation of the field in different applications. In the next

subsection, we demonstrate how this method can be applied for improving ODT reconstruction fidelity.

3.2 Tomographic Reconstruction Results

To show the ability of MaxwellNet to be used for different imaging applications, we implement an optimization task with MaxwellNet as the forward model for ODT, as explained in Sec. 2.2. In this example, we consider one of the digital phantoms in the test set of Fig. 2, and we use 2D MaxwellNet as the forward model to compute the 1D scattered field along the transverse direction x for $N = 81$ different rotation angles θ. We restrict ourselves to the range $\theta \in [-40 \text{ deg}, 40 \text{ deg}]$ as this is consistent with the typical conditions in common tomographic setups. As is shown in Fig. 4(a), the Rytov reconstruction obtained from these field projections is elongated along the z axis and underestimated due to missing frequencies.
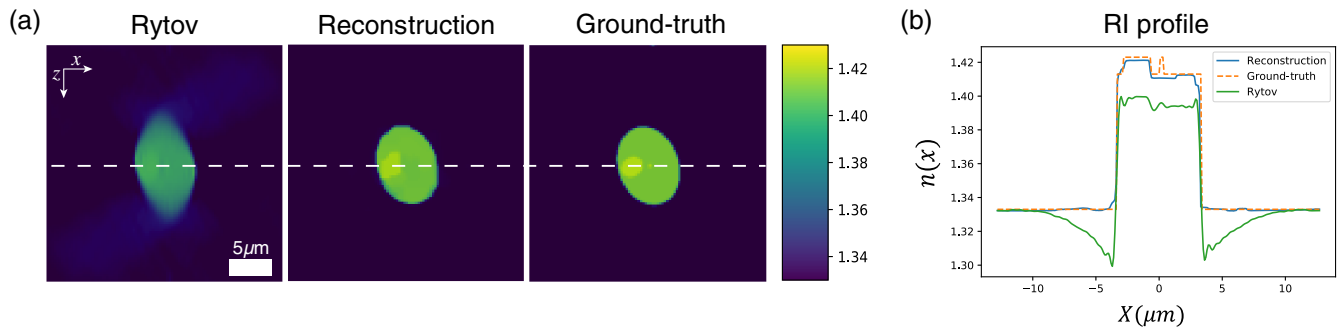
Fig. 4 Tomographic reconstruction of RI using MaxwellNet. (a) The RI reconstruction was achieved by Rytov, MaxwellNet, and the ground truth. (b) 1D RI profile at $z = 0$ (plane of best focus), for Rytov (green), MaxwellNet (blue), and the ground truth (orange).

We then minimize the loss function [Eq. (3)] to improve the RI reconstruction choosing as regularizer parameters $\lambda_{\mathrm{TV}} = 3.1 \times 10^{-7}$, $\lambda_{\mathrm{NN}} = 1 \times 10^{-1}$, $\lambda_{\mathrm{Ph}} = 5 \times 10^{-2}$, and the Adam optimizer with an initial learning rate of 3×10^{-4}. We also scheduled the learning rate, halving it every 1000 epochs to speed up convergence. The resulting RI distribution after 3000 epochs is shown in Fig. 4. It can be seen that the reconstructed RI is no longer underestimated nor elongated along the z axis. This is a significant improvement in comparison with the Rytov prediction. The missing details in the reconstructed RI, which can be better visible in the 1D cutline in Fig. 4(b), can be due to the missing information in the 1D fields that the optimization of RI could not retrieve.

Next, we try a 3D digital phantom from the test set and we use the 3D MaxwellNet as the forward model in our tomographic reconstruction method. Since generating synthetic data with COMSOL is time-consuming for multiple angles, we create synthetic scattered fields from the phantom with the Lippmann–Schwinger equation.[9] Later, we will show an experimental example where we illuminate the sample with a circular illumination pattern with an angle ≈ 10 deg. As a result, in this numerical example, we rotate the sample for 181 angles (including 1 normal incidence), equivalently to an illumination rotation with a fixed illumination angle of 10 deg. We keep the experimental conditions, $\lambda = 1.030$ μm and $n_0 = 1.33$. Then, we use these synthetic measurements for our optimization task along with TV, non-negativity, and physics-informed regularization. The reconstruction is achieved after 6000 epochs with $\lambda_{\mathrm{TV}} = 1.2 \times 10^{-8}$, $\lambda_{\mathrm{NN}} = 2 \times 10^1$, and λ_{Ph} started with 5×10^{-1} and divided by two in every 500 epochs. The reconstructions are shown in Fig. 5 in YX, YZ, and XZ planes. The first row shows the Rytov reconstruction where we can see a significant underestimation and elongation along the z axis that is due to the small illumination angle (10 deg). The details in the reconstruction achieved using MaxwellNet are slightly blurred in comparison with the ground truth as a result of low resolution with $\lambda = 1.030$ μm.

Additionally, we performed learning tomography[6] for the synthetic measurements using 181 projections. The 3D tomographic reconstruction using learning tomography is shown in the third row of Fig. 5. In comparison with MaxwellNet, learning tomography has some elongated artifacts, which can be due to the fact that reflection is neglected in its forward model. However, the reconstruction with learning tomography is smoother in comparison with the reconstruction of MaxwellNet, which is slightly pixelated. This issue happens because the beam propagation method, as the forward model of learning tomography, is a smooth forward model with respect to the voxels of the refractive index distribution, which is not the case for a deep neural network such as MaxwellNet. However, the reconstructions are quantitatively comparable. If we assume the reconstruction error of $\varepsilon(n_{\mathrm{recon}}, n_{\mathrm{truth}}) = \|n_{\mathrm{recon}} - n_{\mathrm{truth}}\|_2^2 / \|n_{\mathrm{truth}} - n_0\|_2^2$, we get an error of 0.613 for Rytov, an error of 0.146 for learning tomography, and an error of 0.116 for MaxwellNet reconstructions, as shown in Fig. 5. In terms of computation time with the desktop specifications we mentioned earlier, we used 3000 epochs for iterative optimization with MaxwellNet, each epoch taking 570 ms and 600 epochs for learning tomography, each epoch taking 710 ms, which means a four-fold factor for MaxwellNet in the computation time.

We also evaluated our methodology experimentally. We mentioned earlier that MaxwellNet takes care of reflection as a forward model, and therefore, our reconstruction technique can be used for samples with high contrast. In our experimental analysis, we try a polystyrene microsphere immersed in water, where we expect to have a ~ 0.25 refractive index contrast. Polystyrene microspheres (Polybead Polystyrene 2.0 Micron) are immersed in water and placed between two #1 glass coverslips. We have an off-axis holographic setup where we use a yttrium-doped fiber laser (Amplitude Laser Satsuma) with $\lambda = 1.030$ μm and we change the illumination angle with two Galvo mirrors. Using a delay path, the optical lengths of the reference and signal arms are matched. We measure holograms for 181 illumination angles and extract the phase and amplitude of the complex scattered fields using Fourier holography. More details about the experimental setup are discussed in Appendix C. Then, we use the extracted scattered fields for different projections for our optimization task to reconstruct the 3D RI distribution of the sample. The experimental projections are 2D complex fields that are imaged in the center of the sample using a microscope objective lens, and we can propagate them in the background medium to calculate the scattered field in any other plane, perpendicular to z axis, after the sample. This 2D field can be compared with the output of MaxwellNet in that plane, as described in Eq. (3). Additionally, the experimental projections are based on illumination rotation and we interpolate them to achieve the equivalent sample rotation projections. We iteratively optimize the loss function in Eq. (3) for 2000 epochs where we use the regularization parameters of
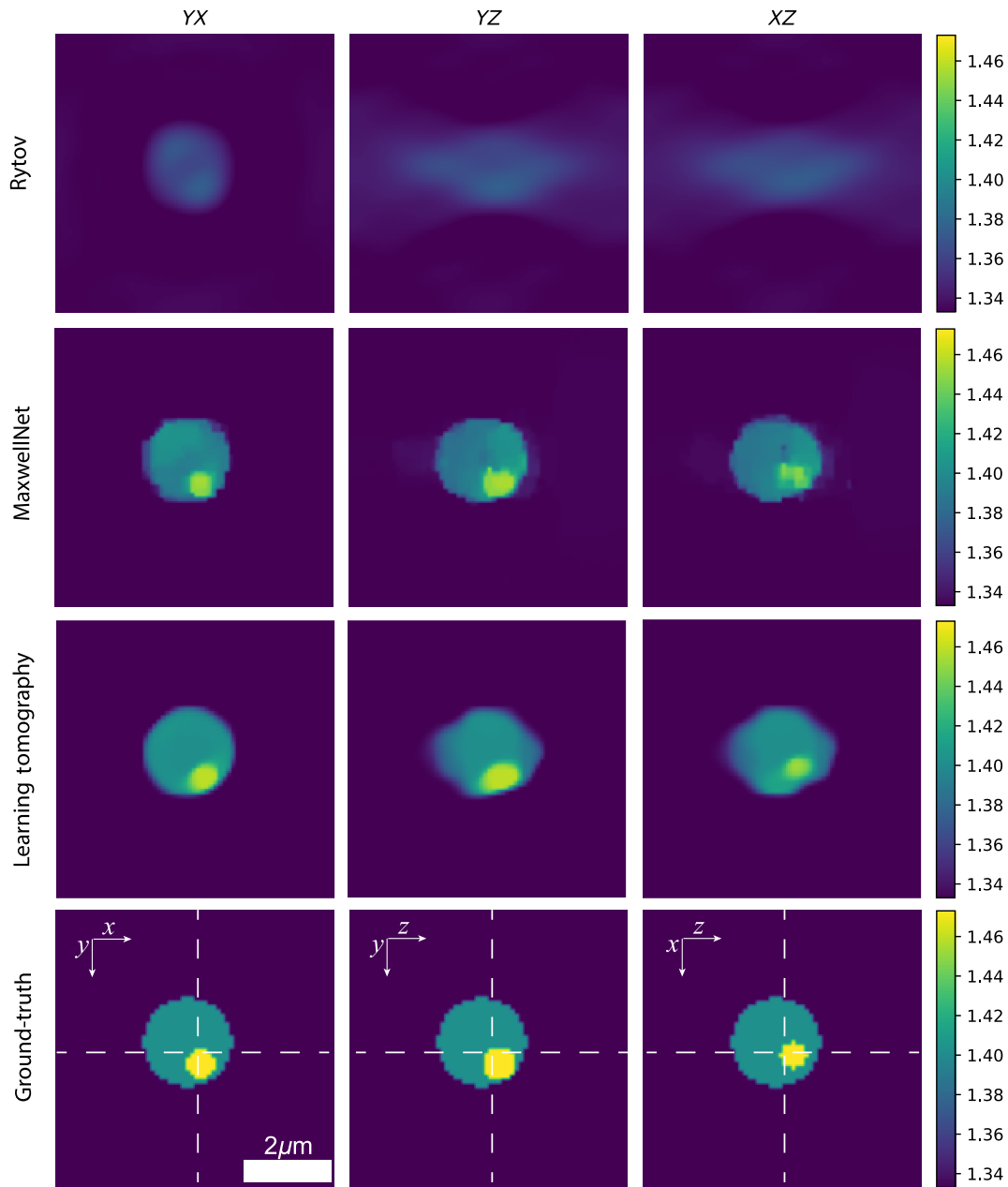
Fig. 5 Tomographic RI reconstruction of 3D sample using MaxwellNet. The RI reconstruction is achieved by Rytov, MaxwellNet, learning tomography, and the ground truth in different rows at the YX, YZ, and XZ planes in the center of the sample. A z-stack demonstration of the reconstruction is shown in Video 1 (Video 1, MP4, 1.4 MB [URL: https://doi.org/10.1117/1.AP.4.6.066001.s1]).

$\lambda_{\mathrm{TV}} = 3.8 \times 10^{-9}$, $\lambda_{\mathrm{NN}} = 5 \times 10^{1}$, and λ_{Ph} started with 1.5×10^{-1} and divided by two in every 500 epochs. The reconstruction is shown in Fig. 6 using Rytov, MaxwellNet, and learning tomography. It can be seen that the underestimation and z axis elongation in the Rytov reconstruction are remarkably improved. The reconstruction using learning tomography in Fig. 6 has artifacts due to the high refractive index contrast of the polystyrene bead and reflections that cannot be considered in the beam propagation method.

4 Conclusion

In summary, we proposed a PINN that rapidly calculates the scattered field from inhomogeneous RI distributions such as biological cells. Our network is trained by minimizing a loss function based on Maxwell equations. We showed that the network can be trained for a set of samples and could predict the scattered field for unseen examples that are in the same class. As our PINN is not a data-driven neural network, it can be trained
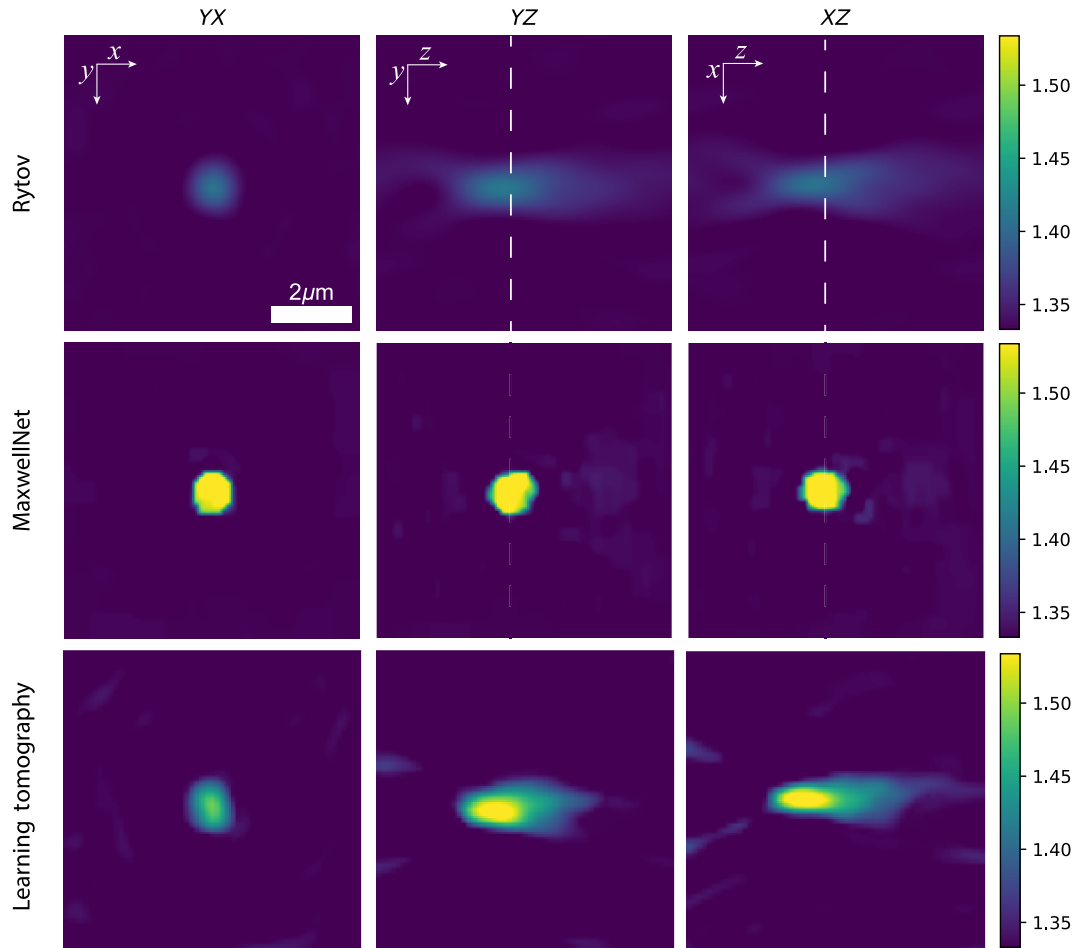
Fig. 6 Tomographic RI reconstruction of a polystyrene microsphere immersed in water. The projections are measured with off-axis holography for different angles. The RI reconstructions achieved by Rytov, MaxwellNet, and learning tomography are presented at the *YX*, *YZ*, and *XZ* planes in the center of the sample, respectively.

for different examples in different conditions. Even though the network is not efficiently extrapolating to classes that are statistically very different from the training dataset, we showed that by freezing the encoder weights and fine-tuning the decoder branch, one can get a new predictive model in a few minutes. We believe that this can be further used for changing the wavelength, boundary condition, or other physical parameters.

We used our PINN as a forward model in an optimization loop to retrieve the RI distribution from the scattered fields achieved by illuminating the sample from different directions, known as ODT. This example shows the ability of MaxwellNet to be used as an accurate forward model in optimization loops for inverse design or inverse scattering problems.

5 Appendix A: Calculation of Physics-Informed Loss

During the training of MaxwellNet, we calculate at each epoch the loss function in Eq. (1) for the network output. To evaluate the Helmholtz equation residual, we should numerically compute the term $\frac{\partial^2 U^s}{\partial x^2} + \frac{\partial^2 U^s}{\partial y^2} + \frac{\partial^2 U^s}{\partial z^2}$. In the previous PINN papers for solving PDEs,[19–21,23–26] the inputs of the network are the spatial coordinates x, y, z, and the derivatives with respect to these

variables can be calculated using the chain rule. In this implementation, the weights of the network can be trained to minimize the loss function for a single refractive index distribution, $n(r)$ in Eq. (1). In our approach, the PINN gets the refractive index, $n(r)$, on a uniform grid as the input and finds the field on the same grid which minimizes the loss function for that refractive index. The output of the network is the 3D array of the scattered field envelope, and we use a finite difference scheme to calculate the derivative of the field with respect to the coordinates:

$$\frac{\partial U^s}{\partial x} = \frac{U^s((i+1)\Delta x, j\Delta y, k\Delta z) - U^s((i-1)\Delta x, j\Delta y, k\Delta z)}{2\Delta x},$$

(7)

in which (i, j, k) are the pixel indices, and Δx, Δy, and Δz are the pixel sizes along the x-, y-, and z-axes. This way, we can calculate $\frac{\partial U^s}{\partial x}$ by convolving U^s with a kernel of $[-1/2, 0, 1/2]$ along the x axis. When computing electromagnetic fields, since the curl of the electric field gives the magnetic field and vice versa, a smart technique to improve accuracy is to use two staggered grids for discretizing fields, commonly referred to as the

Yee scheme.[29] In practice, this can be easily implemented through two shifted convolutional kernels for the two grids, $[-1/2, 1/2, 0]$ and $[0, -1/2, 1/2]$.

To minimize the discretization error, one can use a smaller pixel size, Δx, or higher-order approximations. Here, we use the fourth-order finite difference scheme[34] in which convolutional kernels of $[0, +1/24, -9/8, +9/8, -1/24]$ and $[+1/24, -9/8, +9/8, -1/24, 0]$ are used for the calculation of the derivatives in Eq. (1).

6 Appendix B: Training and Fine-tuning of MaxwellNet

As mentioned in Sec. 3, we create a dataset of digital cell phantoms to train and validate MaxwellNet. The dataset for 2D MaxwellNet includes 3000 phantoms with elliptical shapes oriented in different directions. The size of these phantoms is in the range from 5 to 10 μm, their refractive index varies in the range of (1.38, 1.45), and the background refractive index is $n_0 = 1.33$. Two examples of these phantoms are shown in Fig. 2. We divide this dataset into 2700 phantoms for training and 300 phantoms for testing. We use batch training with a batch size of 10 for 5000 epochs. This training took 30.5 h and after 5000 epochs, no significant decrease in the validation loss could be observed. The training and validation curves of the physical loss are shown in Fig. 7(a). This figure shows that MaxwellNet performs very well for out-of-sample cases.

We discussed in Sec. 3 using MaxwellNet that was trained for cell phantoms to predict the scattered field for real cells. A dataset of HCT-116 cancer cells is used for this purpose. The 3D refractive index of these cells is reconstructed using the Rytov approximation, with projections achieved with an experimental setup utilizing a spatial light modulator, as described in Ref. 8. Then, a 2D slice of the refractive index is chosen in the plane of best focus. A total number of 8 cells are used, and we rotated and shifted these cells to create a dataset of 136 inhomogeneous cells whose refractive index range is (1.33, 1.41). We use 122 of these images for training and 14 for validation. Some examples of the HCT-116 refractive index dataset are shown in Fig. 7(c). We freeze the encoder of MaxwellNet and fine-tune its decoder for this new dataset. The training and validation losses are shown in Fig. 7(b).

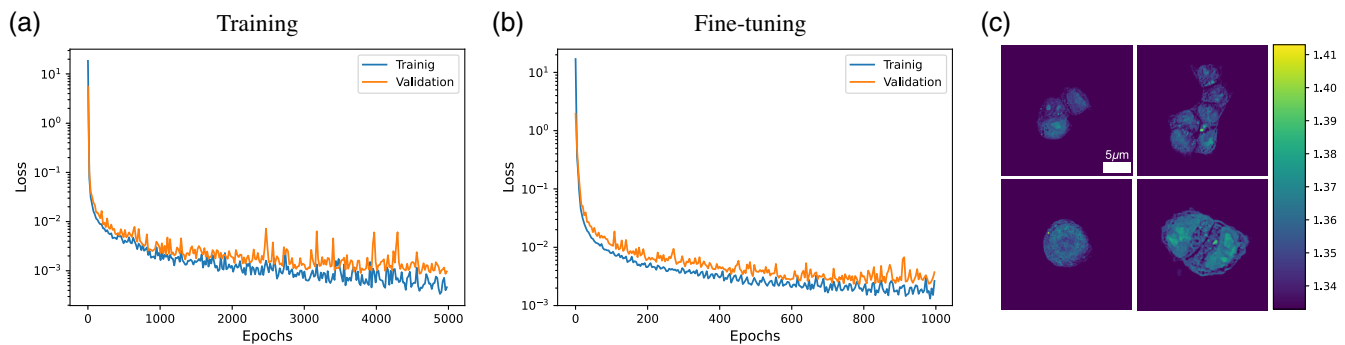For 3D MaxwellNet, a dataset of 200 phantoms is created. These 3D phantoms have a spherical shape with some details

Fig. 7 Training and fine-tuning of MaxwellNet. (a) Training (blue) and validation (orange) loss of MaxwellNet for the digital cell phantoms dataset. (b) Fine-tuning of the pretrained MaxwellNet for a dataset of HCT-116 cells for 1000 epochs. (c) Examples of the HCT-116 dataset.
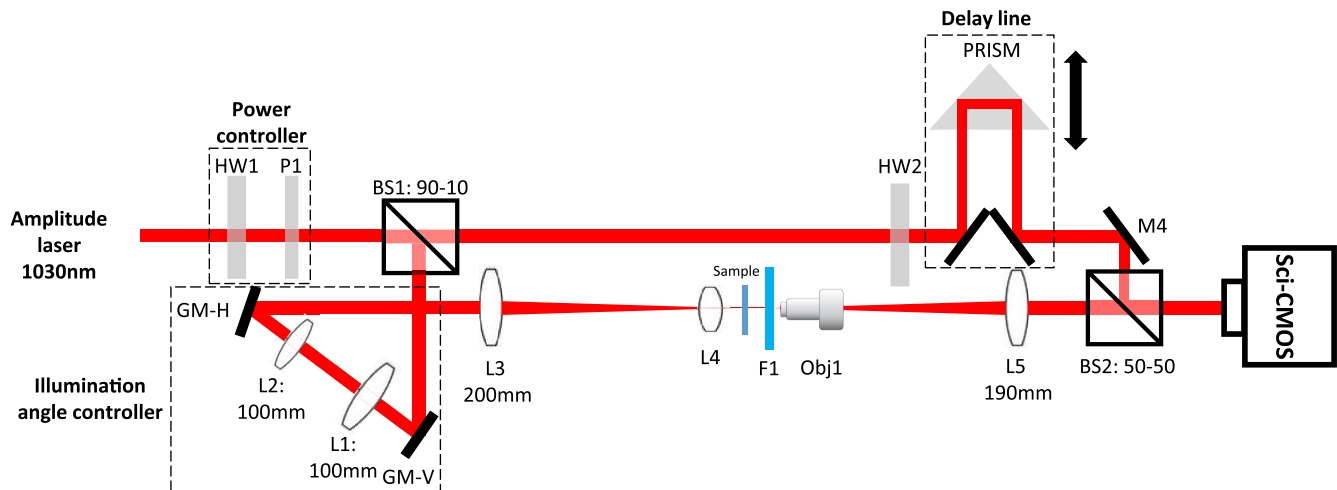

Fig. 8 Experimental setup for multiple illumination angle off-axis holography. HW: half-wave plate; P: polarizer; BS: beam splitter; L: lens; Obj: microscope objective; and M: mirror.

inside them and the range of their diameter is 1.8 to 2.4 μm. We randomly choose 180 phantoms for training and 20 phantoms for testing. We train 3D MaxwellNet with the training dataset with a batch size of 10. The example of Figs. 3 and 5 is one of the phantoms in the testing dataset.

7 Appendix C: Experimental Setup for ODT

For ODT, we require complex scattered fields from multiple illumination angles. The off-axis holographic setup to accomplish that is shown in Fig. 8. It relies on an ytterbium-doped fiber laser at $\lambda = 1.030$ μm whose power is controlled with a half-wave plate (HW) and a polarizing beam splitter. The optical beam is divided into the signal and reference arms using a beam splitter (BS1). In the signal arm, we use two galvo mirrors, GM-V and GM-H, to control the illumination angle in the vertical and horizontal directions. Using two 4F systems (L1–L4), we image these galvo mirrors on the sample plane, so the position of the beam remains fixed while changing the illumination angle. This way, we can illuminate the sample with a condensed plane wave. The sample is then imaged on the camera (Andor sCMOS Neo 5.5) using another 4F system consisting of a 60× water dipping objective (Obj1) and a tube lens L5. The signal and reference arms are then combined with another beam splitter, BS2, to create the off-axis hologram on the camera. A motorized delay line controls the optical path of the reference arm to match the optical path of the signal arm.

References

1. W. Choi et al., "Tomographic phase microscopy," *Nat. Methods* **4**(9), 717–719 (2007).
2. Y. Sung et al., "Optical diffraction tomography for high resolution live cell imaging," *Opt. Express* **17**(1), 266–277 (2009).
3. D. Jin et al., "Tomographic phase microscopy: principles and applications in bioimaging," *J. Opt. Soc. Am. B* **34**(5), B64–B77 (2017).
4. E. Wolf, "Three-dimensional structure determination of semi-transparent objects from holographic data," *Opt. Commun.* **1**(4), 153–156 (1969).
5. J. Lim et al., "Comparative study of iterative reconstruction algorithms for missing cone problems in optical diffraction tomography," *Opt. Express* **23**(13), 16933–16948 (2015).
6. U. S. Kamilov et al., "Learning approach to optical tomography," *Optica* **2**(6), 517–522 (2015).
7. S. Chowdhury et al., "High-resolution 3D refractive index microscopy of multiple-scattering samples from intensity images," *Optica* **6**(9), 1211–1219 (2019).
8. J. Lim et al., "High-fidelity optical diffraction tomography of multiple scattering samples," *Light Sci. Appl.* **8**(1), 1 (2019).
9. T.-A. Pham et al., "Three-dimensional optical diffraction tomography with Lippmann–Schwinger model," *IEEE Trans. Comput. Imaging* **6**, 727–738 (2020).
10. G. E. Karniadakis et al., "Physics-informed machine learning," *Nat. Rev. Phys.* **3**(6), 422–440 (2021).
11. S. Cai et al., "Physics-informed neural networks (PINNs) for fluid mechanics: a review," *Acta Mech. Sin.* **37**, 1727–1738 (2021).
12. Y. Rivenson et al., "Deep learning microscopy," *Optica* **4**(11), 1437–1443 (2017).
13. N. Borhani et al., "Learning to see through multimode fibers," *Optica* **5**(8), 960–966 (2018).
14. B. Rahmani et al., "Multimode optical fiber transmission with a deep learning network," *Light Sci. Appl.* **7**(1), 69 (2018).
15. Y. Rivenson et al., "Phase recovery and holographic image reconstruction using deep learning in neural networks," *Light Sci. Appl.* **7**(2), 17141 (2018).
16. J. Lim, A. B. Ayoub, and D. Psaltis, "Three-dimensional tomography of red blood cells using deep learning," *Adv. Photonics* **2**(2), 026001 (2020).
17. Z. Ren, Z. Xu, and E. Y. Lam, "End-to-end deep learning framework for digital holographic reconstruction," *Adv. Photonics* **1**(1), 016004 (2019).
18. D. Pirone et al., "Speeding up reconstruction of 3D tomograms in holographic flow cytometry via deep learning," *Lab Chip* **22**(4), 793–804 (2022).
19. I. E. Lagaris, A. Likas, and D. I. Fotiadis, "Artificial neural networks for solving ordinary and partial differential equations," *IEEE Trans. Neural Networks* **9**(5), 987–1000 (1998).
20. L. Lu et al., "DeepXDE: a deep learning library for solving differential equations," *SIAM Rev.* **63**(1), 208–228 (2021).
21. M. Raissi, P. Perdikaris, and G. E. Karniadakis, "Physics-informed neural networks: a deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations," *J. Comput. Phys.* **378**, 686–707 (2019).
22. S. M. H. Hashemi and D. Psaltis, "Deep-learning PDEs with unlabeled data and hardwiring physics laws," arXiv:1904.06578 (2019).
23. Z. Mao, A. D. Jagtap, and G. E. Karniadakis, "Physics-informed neural networks for high-speed flows," *Comput. Methods Appl. Mech. Eng.* **360**, 112789 (2020).
24. X. Jin et al., "NSFnets (Navier-Stokes flow nets): physics-informed neural networks for the incompressible Navier-Stokes equations," *J. Comput. Phys.* **426**, 109951 (2021).
25. Y. Chen et al., "Physics-informed neural networks for inverse problems in nano-optics and metamaterials," *Opt. Express* **28**(8), 11618–11633 (2020).
26. Y. Chen and L. Dal Negro, "Physics-informed neural networks for imaging and parameter retrieval of photonic nanostructures from near-field data," *APL Photonics* **7**(1), 010802 (2022).
27. J. Lim and D. Psaltis, "Maxwellnet: physics-driven deep neural network training based on maxwell's equations," *APL Photonics* **7**(1), 011301 (2022).
28. O. Ronneberger, P. Fischer, and T. Brox, "U-Net: convolutional networks for biomedical image segmentation," *Lect. Notes Comput. Sci.* **9351**, 234–241 (2015).
29. K. Yee, "Numerical solution of initial boundary value problems involving Maxwell's equations in isotropic media," *IEEE Trans. Antennas Propag.* **14**(3), 302–307 (1966).
30. W. C. Chew and W. H. Weedon, "A 3D perfectly matched medium from modified Maxwell's equations with stretched coordinates," *Microwave Opt. Technol. Lett.* **7**(13), 599–604 (1994).
31. A. Ishimaru, *Wave Propagation and Scattering in Random Media*, Vol. **2**, Academic Press, New York (1978).
32. A. Saba et al., "Polarization-sensitive optical diffraction tomography," *Optica* **8**(3), 402–408 (2021).
33. C. Tan et al., "A survey on deep transfer learning," *Lect. Notes Comput. Sci.* **11141**, 270–279 (2018).
34. A. Fathy et al., "A fourth order difference scheme for the Maxwell equations on Yee grid," *J. Hyperbol. Differ. Equ.* **5**(3), 613–642 (2008).

Amirhossein Saba is a PhD student in photonics at École Polytechnique Fédérale de Lausanne (EPFL), Lausanne, Switzerland. He received his BS degree in electrical engineering with a minor in physics from Sharif University of Technology, Tehran, Iran, in 2018. His current research

interests include computational imaging, polarization-sensitive, and non-linear imaging techniques.

Carlo Gigli received his MS degree in nanotechnologies for ICTs from Politecnico di Torino and Université Paris Diderot in 2017, and his PhD in physics at Université de Paris in 2021. During this period, his research interests included the design, fabrication, and characterization of dielectric resonators and metasurfaces for nonlinear optics. Since September 2021, he has been working as post-doc in the Optics Laboratory at EPFL, where his main activities focus on optical computing, physics informed neural network, and nonlinear tomography.

Ahmed B. Ayoub received his BS degree in electrical engineering from Alexandria University in Egypt in 2013. He received his MS degree in physics from the American University, Cairo, Egypt, in 2017. He is pursuing his PhD in electrical engineering at the École Polytechnique Fédérale de Lausanne, Lausanne, Switzerland. His current research interests include optical imaging and three-dimensional refractive index reconstructions for biological samples.

Demetri Psaltis received his BSc, MSc, and PhD degrees from Carnegie-Mellon University, Pittsburgh, Pennsylvania. He is a professor of optics and director of the Optics Laboratory at the École Polytechnique Fédérale de Lausanne (EPFL), Lausanne, Switzerland. In 1980, he joined the faculty at the California Institute of Technology, Pasadena, California. He moved to EPFL in 2006. His research interests include imaging, holography, biophotonics, nonlinear optics, and optofluidics. He has authored or coauthored over 400 publications in these areas. He is a fellow of the Optical Society of America, the European Optical Society, and SPIE. He was the recipient of the International Commission of Optics Prize, the Humboldt Award, the Leith Medal, and the Gabor Prize.