



# Deep-learning projector for optical diffraction tomography

FANGSHU YANG,<sup>1,2</sup>  THANH-AN PHAM,<sup>2</sup>  HARSHIT GUPTA,<sup>2</sup>  
MICHAEL UNSER,<sup>2</sup> AND JIANWEI MA<sup>1,\*</sup> 

<sup>1</sup>*Department of Mathematics and Artificial Intelligence Laboratory, Harbin Institute of Technology, Harbin 150001, China*

<sup>2</sup>*Biomedical Imaging Group, École polytechnique fédérale de Lausanne (EPFL), CH-1015 Lausanne, Switzerland*

\*[jma@hit.edu.cn](mailto:jma@hit.edu.cn)

**Abstract:** Optical diffraction tomography is an effective tool to estimate the refractive indices of unknown objects. It proceeds by solving an ill-posed inverse problem for which the wave equation governs the scattering events. The solution has traditionally been derived by the minimization of an objective function in which the data-fidelity term encourages measurement consistency while the regularization term enforces prior constraints. In this work, we propose to train a convolutional neural network (CNN) as the projector in a projected-gradient-descent method. We iteratively produce high-quality estimates and ensure measurement consistency, thus keeping the best of CNN-based and regularization-based worlds. Our experiments on two-dimensional-simulated and real data show an improvement over other conventional or deep-learning-based methods. Furthermore, our trained CNN projector is general enough to accommodate various forward models for the handling of multiple-scattering events.

© 2020 Optical Society of America under the terms of the [OSA Open Access Publishing Agreement](#)

## 1. Introduction

In the last decade, optical diffraction tomography (ODT) has grown in interest [1]. ODT was first theoretically proposed in 1969 by Wolf [2] and later given a geometrical interpretation by Dändliker and Weiss [3]. This modality attempts to reconstruct the spatial distribution of the refractive index (RI) by sequentially illuminating the sample with tilted incident fields. The interaction between each incident field and the sample produces a complex scattered field recorded by the acquisition setup [4]. Importantly, in many practical applications, a full 360° probing of the object is not feasible. The limited view induces the so-called missing-cone problem and results in the underestimation of RI values and the elongation of the reconstructed shapes [5,6].

### 1.1. Classical reconstruction methods

In the early works [7–9], ODT took advantage of linear, direct-inversion algorithms that rely on single-scattering models such as Born [10] or Rytov [11] approximations. Under these models, the relationship between the measured wave and the Fourier transform of the sample enables the restoration of the RI distribution. In the Rytov approximation, which is preferred, the unwrapped phase of the measurement is utilized instead of the scattered field, so that the index-matching requirement is less stringent [12]. These algorithms are valid under the weak-scattering condition, but less accurate for samples with high RI contrast, large radii, or complex structures.

The restriction to weak scattering can be lifted by the use of nonlinear models [13] that account for multiple scattering, thereby improving the quality of reconstruction. Previous works have used the beam-propagation method (BPM) [14,15], the contrast source-inversion method [16], the conjugate-gradient method [17], the recursive Born approximation [18], and the Lippmann-Schwinger model [19,20]. Furthermore, the missing-cone problem is partially overcome by

including prior knowledge such as total variation (TV) [21], non-negativity constraints [22], other sparsity-enforcing penalties [15], or plug-and-play priors [23].

### 1.2. Learning-based reconstruction methods

In the past decade, several works that adopted deep neural networks (DNN) [24,25] have reached state-of-the-art performance in many challenging linear inverse problems such as image reconstruction [26], super-resolution [27], x-ray computed tomography [28,29], and compressive sensing [30]. Instead of retrieving the image from the measurements directly, the strategy is to train a convolutional neural network (CNN) [31] to map the initial reconstructions, which are usually obtained by classical methods, into the final desired results [32].

Another popular variant is to modify traditional iterative algorithms in such a way that the network plays the role of a regularizer. In [33], a model-based iterative algorithm for image reconstruction was proposed in which a deep CNN-based regularization prior was combined with numerical optimization modules such as conjugate-gradient algorithms to enforce data consistency and constrain the plausibility of the solutions. In [34], the authors proposed an iterative reconstruction method for nonlinear tomographic problems. Specifically, they devised a gradient-descent-like scheme where the gradient is learned via a CNN and showed that it works with any nonlinear operator. In [35], a CNN is trained as a projector on the training data and is then plugged in the alternating-direction method of multipliers method to solve linear inverse problems. The framework there is similar to plug-and-play priors for model-based reconstruction [36]. In [37], where the idea is similar to that in [35], a CNN is trained with multiple stages to approximate a projector for the consistent reconstruction of computed tomography (CT) images. The projector is then plugged into a relaxed projected-gradient-descent (PGD) method which is guaranteed to converge.

In diffraction tomography, learning-based RI tomography approaches are also being rapidly developed. In [14], the authors described a framework for the imaging of three-dimensional (3D) phase objects in a tomographic configuration implemented by training a DNN to reproduce the complex amplitudes of the measurements. The network was designed such that the voxel values of the RI of the 3D object are the inputs that are adapted during the training process. In [38], the authors improved upon [14] by using an  $\ell_1$  loss function and anisotropic TV. In [39], the scattering decoder framework (ScaDec) combines a back-propagation scheme with a U-Net [40]. The authors could recover strongly scattering objects in a purely data-driven fashion. Yet other CNN architectures were proposed in [41,42]. These works mainly concentrated on cases characterized by the absence of the missing-cone problem. Nevertheless, they suggest that CNN could be an appealing solution to the specific challenges of ODT.

### 1.3. Contribution

Inspired by [37], we develop a plug-and-play scheme with a CNN trained as a projector for ODT. Our approach contains a feedback mechanism that ensures consistency with the measurements while taking advantage of the complexities of CNN. By contrast to [37], we adapt the framework to a more challenging modality with an underlying nonlinear physical model and a missing-cone problem. We extensively validate the proposed method on simulated and real data in different scenarios. We compare the proposed method against the conventional direct inversion method (a.k.a. the filtered back-propagation (FBP) method), iterative inversion methods combined with TV regularization (*e.g.*, RytovTV), ScaDec, and the direct CNN. Our method outperforms traditional optimization-based methods in both simulated and real datasets with diverse configurations. Furthermore, our framework allows the use of more accurate models, for instance, BPM.

### 1.4. Organization

In Section 2, we present the physical model in ODT [10] and its classical approximations. In Section 3, we formulate the problem and explore an iterative scheme based on PGD. In Section 4, we describe the proposed deep-learning scheme; it involves a novel strategy to train the CNN as a projector onto a set of desirable solutions. In Section 5, we validate our method on two-dimensional (2D) simulated and experimental data against conventional and other CNN-based approaches. Finally, we summarize our study in Section 6.

## 2. Optical diffraction tomography

Let  $n : \Omega \mapsto \mathbb{R}$  denote the distribution of RI of the object immersed in a medium of RI  $n_b$ , where  $\Omega \subset \mathbb{R}^d$  includes the support of the sample and  $d$  is the dimension of the problem (e.g.,  $d = 2, 3$ ). In ODT, an incident plane wave  $u^{in} : \Omega \mapsto \mathbb{C}$  of wavelength  $\lambda$  illuminates the sample. The interaction between  $n$  and  $u^{in}$  produces a scattered field  $u^{sc} : \Omega \mapsto \mathbb{C}$ . The complex total field  $u = u^{in} + u^{sc}$  is measured at the location  $\Gamma$  of the detector plane (see Fig. 1). In the scalar-diffraction theory, the total field  $u$  is governed by the Helmholtz equation

$$\nabla^2 u(\mathbf{x}) + k_0^2 n^2(\mathbf{x})u(\mathbf{x}) = 0, \tag{1}$$

with the free-space wavenumber  $k_0 = \frac{2\pi}{\lambda}$ . Under suitable conditions, the integral form of (1) is the Lippmann-Schwinger equation

$$u(\mathbf{x}) = u^{in}(\mathbf{x}) + \int_{\Omega} g(\mathbf{x} - \mathbf{z})f(\mathbf{z})u(\mathbf{z})d\mathbf{z}, \quad \mathbf{x} \subseteq \Omega, \tag{2}$$

where  $f(\mathbf{x}) = k^2(\frac{n^2(\mathbf{x})}{n_b^2} - 1)$  denotes the scattering potential and  $k = k_0 n_b$  is the wavenumber in the medium. The convolution kernel  $g : \mathbb{R}^d \mapsto \mathbb{C}$  is the Green function of  $(\nabla^2 + k_0^2 n_b^2 \mathbf{I})$  given by

$$g(\mathbf{x}) = \frac{j}{4} H_0^{(1)}(k_0 n_b \|\mathbf{x}\|), \quad d = 2, \tag{3}$$

where  $H_0^{(1)}$  denotes the Hankel function of the first kind.

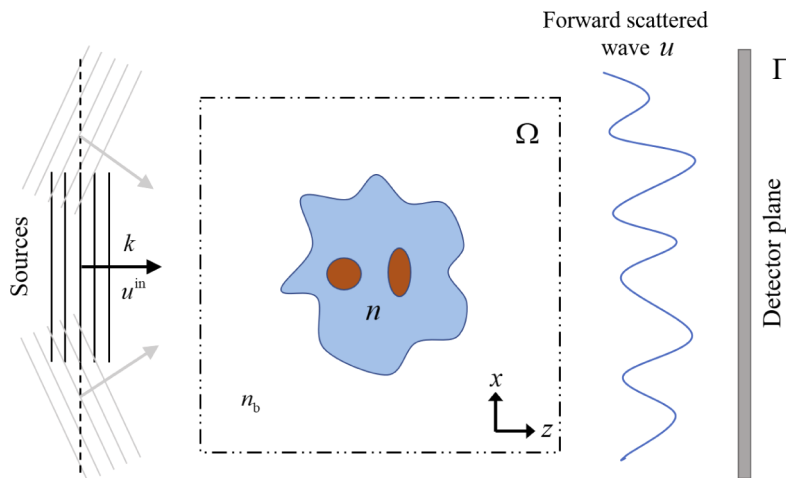


Fig. 1. Optical diffraction tomography.

Under specific hypotheses, we can linearize this problem by replacing the total field  $u$  with the incident field  $u^{\text{in}}$ . This yields the first-order Born approximation, written as

$$u_{\text{B}}^{\text{sc}}(\mathbf{x}) = \int_{\Omega} g(\mathbf{x} - \mathbf{z})f(\mathbf{z})u^{\text{in}}(\mathbf{z})d\mathbf{z}. \quad (4)$$

The fact that the Born model only allows one to observe optically thin samples is a serious drawback. The Rytov model [11] yields a similar equation with the left term of (4) being replaced by  $u^{\text{Ryt}} = u^{\text{in}}\log(\frac{u}{u^{\text{in}}})$ . In practice, it is necessary to unwrap the imaginary part of  $\log(\frac{u}{u^{\text{in}}})$ . In this work, we use the Rytov model as our forward operator, unless specified otherwise.

### 2.1. Formulation of the forward model

We discretize  $\Omega$  in  $N$  pixels, and we represent the sampled scattering potential  $f$  and the refractive index  $n$  by the vectors  $\mathbf{f} \in \mathbb{R}^N$  and  $\mathbf{n} \in \mathbb{R}^N$ , respectively. The sample is illuminated by a series of  $P$  incident plane waves  $\{u_p^{\text{in}}(\mathbf{x})\}_{p=1}^P = \{e^{j(\mathbf{k}_p \cdot \mathbf{x})}\}_{p=1}^P$  with the wavevector  $\mathbf{k}_p = k_b \mathbf{s}_p$ , where  $\mathbf{s}_p \in \mathbb{R}^d$  is the directional unit vector for the  $p$ th illumination. We acquire  $M$  complex measurements per illumination. According to the Rytov model [11], our discrete forward model reads as

$$\mathbf{y}_p^{\text{Ryt}} = \mathbf{S}_p \mathbf{F} \mathbf{f}, \quad p = 1, \dots, P, \quad (5)$$

where  $\mathbf{y}_p^{\text{Ryt}} \in \mathbb{C}^M$  are the measurements modified according to the Rytov model,  $\mathbf{S}_p \in \mathbb{R}^{M \times N}$  denotes a sampling operation that depends on the incident field  $p$ , and  $\mathbf{F} \in \mathbb{C}^{N \times N}$  applies a  $d$ -dimensional Fourier transform.

We also introduce the RI variation  $\delta \mathbf{n} \in \mathbb{R}^N$  whose  $m$ th component is given by

$$\delta n_m = n_b \left( \sqrt{\frac{f_m}{k^2} + 1} - 1 \right). \quad (6)$$

In this work, we utilized RI variations to describe the setup of diverse datasets.

## 3. Reconstruction of the distribution of RI

### 3.1. Problem formulation

We reconstruct our object by optimizing the constrained least-square problem

$$\hat{\mathbf{f}} = \arg \min_{\mathbf{f} \in \mathcal{S}} \frac{1}{2P} \sum_{p=1}^P \|\mathbf{y}_p^{\text{Ryt}} - \mathbf{H}_p^{\text{Ryt}}(\mathbf{f})\|_2^2, \quad (7)$$

where  $\mathbf{y}_p^{\text{Ryt}}$  denotes the measurement modified accordingly,  $\mathbf{H}_p^{\text{Ryt}} : \mathbb{R}^N \mapsto \mathbb{C}^M$  is the Rytov model defined in (5), and  $\mathcal{S}$  is a set of acceptable solutions that honor our prior information.

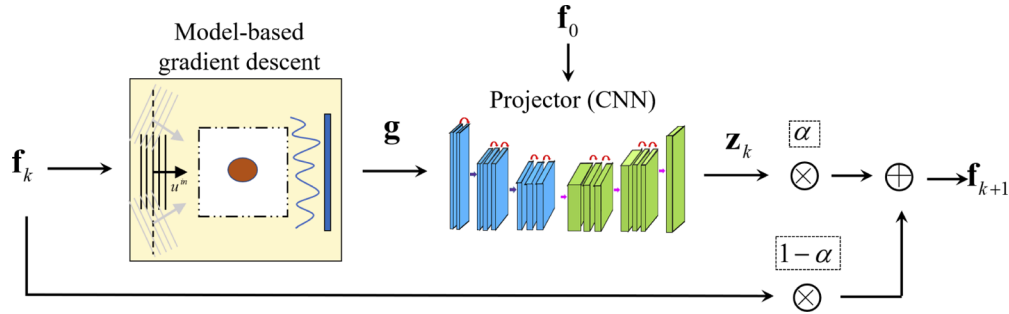
### 3.2. Iterative reconstruction scheme

The PGD [43] is a well-studied iterative method to solve (7). It alternates between taking a step in the direction of the negative gradient of the cost function and applying a projection operator  $P_{\mathcal{S}}$  that maps the current iteration onto  $\mathcal{S}$ . It can be written as

$$\mathbf{f}_{k+1} = P_{\mathcal{S}}(\mathbf{f}_k - \gamma \mathbf{H}^T \mathbf{H} \mathbf{f}_k + \gamma \mathbf{H}^T \mathbf{y}), \quad (8)$$

where  $\gamma$  denotes the step size subject to the condition  $\gamma < 2 / \|\mathbf{H}^T \mathbf{H}\|_2$ , and  $P_{\mathcal{S}} : \mathbb{R}^N \mapsto \mathcal{S}$  is a projection operator.

However, PGD is not ensured to converge when the operator  $P_S$  is not a valid projector, the condition of validity being  $P_S \circ P_S(\mathbf{x}) = P_S(\mathbf{x})$ ,  $\forall \mathbf{x} \in \mathbb{R}^N$ . We therefore consider the relaxed PGD (RPGD) of [37], which is ensured to converge (see Theorem 3 of [37]). The iterative reconstruction scheme is sketched in Algorithm 1. In Line 6, in particular, we need a projector onto  $\mathcal{S}$ . Inspired by recent works on learning approaches, our goal is to replace  $P_S$  with a CNN that is trained to act as a projector (see Fig. 2).



**Fig. 2.** One iteration of the RPGD for ODT. The forward model simulates the physical process from the current estimate  $\mathbf{f}_k$ . The vector  $\mathbf{g}$  is obtained by gradient descent from the data-fidelity term. Then,  $\mathbf{g}$  is fed into the (trained) CNN projector to obtain  $\mathbf{z}_k$ . The next estimate  $\mathbf{f}_{k+1}$  is the sum of  $\mathbf{z}_k$  and  $\mathbf{f}_k$  weighted by  $\alpha$  and  $(1 - \alpha)$ , respectively. Note that the initial guess  $\mathbf{f}_0$ , which is obtained with a direct inversion method (Rytov model), is directly fed into the (trained) CNN projector.

---

#### Algorithm 1 Iterative RPGD

**Input:**  $\mathbf{f}_0$ : initial guess of the scattering potential,  $\mathbf{H}$ : forward model,  $P_S$ : projector,  $\gamma$ : step size,  $c_k$ : positive sequence,  $\mathbf{y}$ : measurements,  $E$ : relative update,  $N$ : number of iterations

**Output:** final estimated scattering potential  $\hat{\mathbf{f}}$

**Initialize:**  $\alpha = 1$ ,  $e = 10^4$ ,  $\mathbf{g} = \mathbf{f}_0$

```

1:  $k = 0$ 
2: while  $e \geq E$  or  $k \leq N$  do
3:   if  $k \geq 1$  then
4:      $\mathbf{g} = \mathbf{f}_k - \gamma \mathbf{H}^T \mathbf{H} \mathbf{f}_k + \gamma \mathbf{H}^T \mathbf{y}$  ▷ Gradient descent
5:   end if
6:    $\mathbf{z}_k = P_S(\mathbf{g})$  ▷ Projection
7:   if  $k \geq 1$  then
8:     if  $\|\mathbf{z}_k - \mathbf{f}_k\|_2 > c_k \|\mathbf{z}_{k-1} - \mathbf{f}_{k-1}\|_2$  then
9:        $\alpha = c_k \|\mathbf{z}_{k-1} - \mathbf{f}_{k-1}\|_2 / \|\mathbf{z}_k - \mathbf{f}_k\|_2$  ▷ Update relaxed parameter
10:    end if
11:  end if
12:   $\mathbf{f}_{k+1} = (1 - \alpha)\mathbf{f}_k + \alpha\mathbf{z}_k$  ▷ Update reconstruction
13:   $e = \|\mathbf{f}_{k+1} - \mathbf{f}_k\|_2 / \|\mathbf{f}_k\|_2$  ▷ Calculate relative update
14:   $k \leftarrow k + 1$ 
15: end while

```

---

#### 4. Deep-learning-based projector for ODT

In this study, we replace Line 6 in Algorithm 1 by

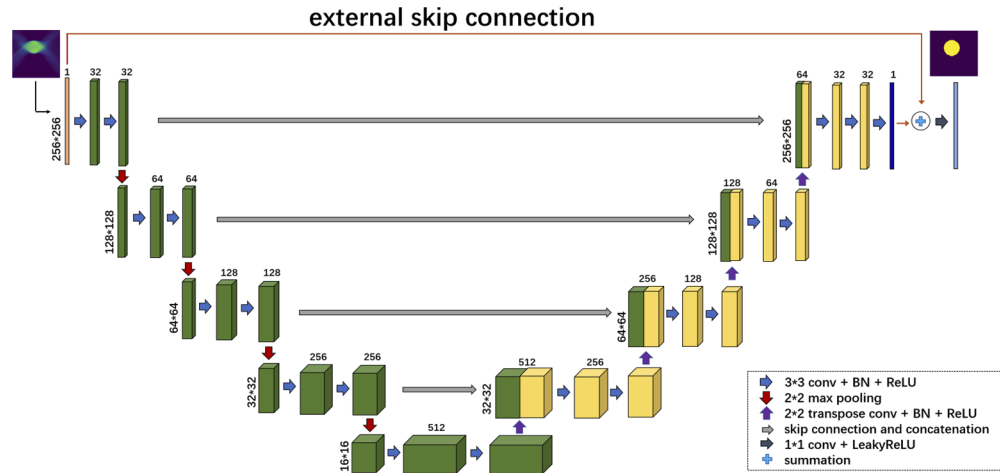
$$P_S(\mathbf{g}) = \text{CNN}(P_C(\mathbf{g})), \quad (9)$$

where CNN is trained to act as a projector and  $P_C$  denotes a projection onto the convex set  $C \subset \mathbb{R}^N$  that enforces physical constraints on the RI (e.g., non-positivity constraint).

The training data set consists of 2D objects placed randomly. We simulated the complex measurements by using an accurate forward model that relies on the Lippmann-Schwinger model [20]. The details of the simulation are described in Section 5.1.1.1.

##### 4.1. Network architecture

We design a CNN based on the U-Net architecture [28,40]. The setup includes an external skip connection between the input and the output so that the network acts as a residual net. Furthermore, we double the number of channels in the encoder (left path) each time the depth of the network increases (starting at 32, Fig. 3). We also replace the (3 \* 3) up-conv layer with a (2 \* 2) transposed convolutional layer, and add a Leaky rectified linear unit (LeakyReLU) [44] after the last convolutional layer (see Fig. 3).



**Fig. 3.** Architecture of the network used for RI reconstruction. Each dark and light cube corresponds to a multichannel feature map. The number of channels is shown at the top of the cube. The x-z size is provided at the lower-left edge of the cube (example shown for 16×16 in the coarsest resolution). The arrows denote the different operations, and the size of the corresponding parameter set is provided in the explanatory frame.

##### 4.2. Training strategy

In our experiments, we adopt a training strategy inspired by [37]. The input of the CNN during training time falls into three classes:

$$\tilde{\mathbf{f}}^{q,1} = \mathbf{B}(\mathbf{H}^{\text{Lip}}(\mathbf{f}^q)) \quad (10)$$

$$\tilde{\mathbf{f}}^{q,2} = \text{CNN}_{\theta_{t-1}}(\tilde{\mathbf{f}}^{q,1}) \quad (11)$$

$$\tilde{\mathbf{f}}^{q,3} = \mathbf{f}^q, \quad (12)$$

where  $\mathbf{H}^{\text{Lip}}$  denotes the Lippmann-Schwinger model, and  $\mathbf{B}$  is shorthand for FBP [11]. The learned parameters in the CNN after the  $t$ th epoch training by is denoted by  $\theta_t$ . The  $Q$  ground-truth

images are indicated by  $\{\mathbf{f}^q\}_{q \in [1 \dots Q]}$ . This strategy enhances the dataset as it introduces a variety of perturbations. On the contrary, there is no perturbation in (12) so that it encourages the network to mimic a projector.

The training process aims at minimizing the loss function

$$L(\theta) = \sum_{m=1}^M \sum_{q=1}^Q \|\mathbf{f}^q - \text{CNN}(\tilde{\mathbf{f}}^{q,m})\|_2^2. \quad (13)$$

The minimization of  $L$  proceeds using the Adam algorithm [45].

In practice, we train the network in three stages. In the first stage, we train it only with the input data set  $\{\tilde{\mathbf{f}}^{q,1}\}$  obtained by (10). Then, in the second stage, we concatenate the data set  $\{\tilde{\mathbf{f}}^{q,2}\}$  generated by (11) to  $\{\tilde{\mathbf{f}}^{q,1}\}$  and train the CNN with a total loss that integrates the two parts. Finally, in the third stage, the whole data set  $\{\tilde{\mathbf{f}}^{q,1}, \tilde{\mathbf{f}}^{q,2}, \tilde{\mathbf{f}}^{q,3}\}$  is fed into the optimization.

## 5. Numerical experiments

In this section, we present experiments that validate our proposed method on 2D simulated and experimental data. For the implementation, the training process is performed on a desktop workstation (Titan X GPU, Ubuntu operating system) and implemented on PyTorch [46], while the evaluation is implemented by MATLAB R2019a on another desktop computer (Intel Xeon E5-1650 CPU, 3.5 GHz, 32 GB of RAM) using the GlobalBioIm library [47].

**Parameter setting** For each of the three stages, the number of epochs was set as  $T_1 = 20$ ,  $T_2 = 30$ , and  $T_3 = 30$ . The initial learning rate was  $10^{-4}$ ; it was optimally reduced to one tenth of the current value at the 40th and 70th epoch. We set the batch size as 18 for the three stages and did draw an equal number of samples from each dataset  $\{\tilde{\mathbf{f}}^{q,m}\}_{m=1}^3$ .

The parameters for the iterative RPGD were set as follows: the relaxed parameter  $\alpha$  was initialized with 1, and all members of the sequence  $\{c_k\}_{k=1}^N$  were set to the constant value  $c$  which was tuned manually for different scenarios. Furthermore, either the relative update  $E = 10^{-4}$  or the number of iterations  $N = 10$  stopped the algorithm. We set  $C = \{\mathbf{f} \in \mathbb{R}^N : \mathbf{f} \leq 0\}$  (*i.e.*, non-positivity constraint), unless specified otherwise.

**Method of comparison** We quantitatively evaluated the quality of the reconstructed RI  $\hat{\mathbf{n}}$  with respect to the ground-truth  $\mathbf{n}$ . To that end, we adopted the relative error (ERROR) and structural similarity (SSIM) [48] defined as

$$\text{ERROR}(\mathbf{n}, \hat{\mathbf{n}}) = \frac{\|\mathbf{n} - \hat{\mathbf{n}}\|_2}{\|\mathbf{n}\|_2} \quad (14)$$

$$\text{SSIM}(\mathbf{n}, \hat{\mathbf{n}}) = \frac{(2\mu_{\mathbf{n}}\mu_{\hat{\mathbf{n}}} + c_1)(2\sigma_{\mathbf{n}}\sigma_{\hat{\mathbf{n}}} + c_2)}{(\mu_{\mathbf{n}}^2 + \mu_{\hat{\mathbf{n}}}^2 + c_1)(\sigma_{\mathbf{n}}^2 + \sigma_{\hat{\mathbf{n}}}^2 + c_2)}, \quad (15)$$

where  $\mu_{\mathbf{n}}$ ,  $\mu_{\hat{\mathbf{n}}}$ ,  $\sigma_{\mathbf{n}}$ ,  $\sigma_{\hat{\mathbf{n}}}$ , and  $\sigma_{\mathbf{n}\hat{\mathbf{n}}}$  are the local means, standard deviations, and cross-covariance for images  $\mathbf{n}$ ,  $\hat{\mathbf{n}}$ . Parameters  $c_1 = 10^{-4}$  and  $c_2 = 9 \times 10^{-4}$  are the default regularization constants. They avoid instabilities over image regions where the local mean or standard deviation is vanishing.

We compare the results of our method with those of FBP, a Rytov-based method with TV regularization (RytovTV), and ScaDec [39]. We also compare them with those of the direct CNN which shares the same architecture as the network of our method, but is only trained with the dataset in (10) (see Table 1). To solve the optimization-based method combined with TV regularization, we adopted the forward-backward splitting approach [49]. All the parameters are optimized to achieve the best relative error.

### 5.1. Validation on simulated data

#### 5.1.1. Training data composed of one disk

**Simulation setup** We generated 1,116 images composed of one disk arbitrarily positioned in the middle third of the horizontal direction and anywhere in the vertical direction. Each disk has a radius ranging from  $4\lambda$  to  $7.5\lambda$ . The data were split into 1,080 images for training, 18 images for validation, and 18 images for testing. The physical size of the image was set to  $(38\lambda \times 38\lambda)$ , and was discretized on a  $(256 \times 256)$  grid. We assumed that the background medium was oil (RI  $n_b = 1.525$ ). The value of the RI variation  $\delta n$  ranged arbitrarily from  $(-0.135)$  to  $(-0.055)$ , component-wise.

**Table 1. RI reconstructions.**

Method	Reference	Algorithm	Regularization	
FBP	[8,9]	direct	–	model-based
RytovTV	[6]	iterative	TV	
ScaDec	[39]	direct	–	CNN-based
DirectCNN	[28]	direct	–	
RytovCNN		iterative	CNN projector	

**Table 2. ERROR of RI reconstructions in the presence of noise, for one disk.**

	20dB SNR	15dB SNR	10dB SNR
FBP	$82.3 \cdot 10^{-4}$	$83.5 \cdot 10^{-4}$	$86.4 \cdot 10^{-4}$
RytovTV	$46.8 \cdot 10^{-4}$	$47.6 \cdot 10^{-4}$	$55.4 \cdot 10^{-4}$
ScaDec	$9.39 \cdot 10^{-4}$	$9.52 \cdot 10^{-4}$	<b><math>11.5 \cdot 10^{-4}</math></b>
DirectCNN	$7.83 \cdot 10^{-4}$	$8.82 \cdot 10^{-4}$	$15.4 \cdot 10^{-4}$
RytovCNN	<b><math>7.01 \cdot 10^{-4}</math></b>	<b><math>7.91 \cdot 10^{-4}</math></b>	$14.1 \cdot 10^{-4}$

The plane waves had incident angles uniformly distributed between  $(-45^\circ)$  to  $45^\circ$ , thus simulating a missing-cone problem. The wavelength of the illumination was set to  $\lambda = 450\text{nm}$ . We acquired  $P = 40$  views on a detector line of length  $97\lambda$ . In addition, to make the measurements more realistic, we added white Gaussian noise  $\mathbf{w}$  to the measurements  $\mathbf{y}$ . The noise was added such that the input SNR = 20dB, where  $\text{SNR}(\mathbf{y} + \mathbf{w}, \mathbf{y}) = 20\log_{10}(\|\mathbf{y}\|_2 / \|\mathbf{w}\|_2)$ . All simulations were implemented using the GlobalBioIm library [47].

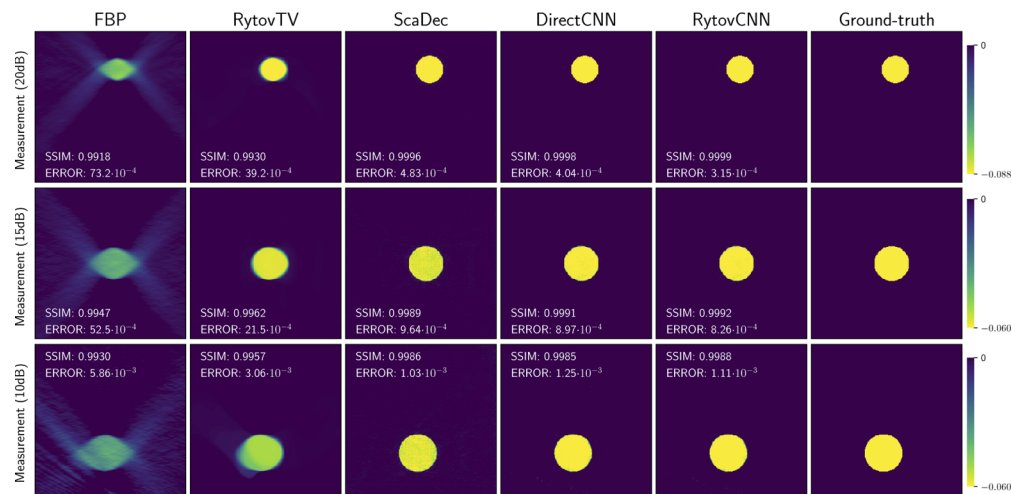
**Reconstruction of one circular disk** We evaluated the stability of the methods over some range of noise (10, 15, and 20 dB) in the context of limited-angle measurements. We summarize in Table 2 and Table 3 the average ERROR and SSIM values over the whole testing dataset. Notably, RytovTV dramatically underestimates the RI distribution and fails to reconstruct the shape of the objects. Our method obtains the highest-quality reconstructions in terms of the actual RI value and the shape of the objects. In most cases, it faithfully recovers the images with lower ERROR and higher SSIM values compared to the other methods. All the deep-learning-based methods did recover the samples successfully while the TV-regularized method did fail. Only for the 10 dB case, ScaDec performed slightly better for the relative error. This can be explained by the failure of the phase unwrapping on which the Rytov model relies, unlike ScaDec. The



proposed method takes 10 seconds for 10 iterations, whereas the RytovTV takes 50 seconds for 80 iterations. Three samples of the reconstructed images are shown in Fig. 4.

**Table 3. SSIM of RI reconstructions in the presence of noise, for one disk.**

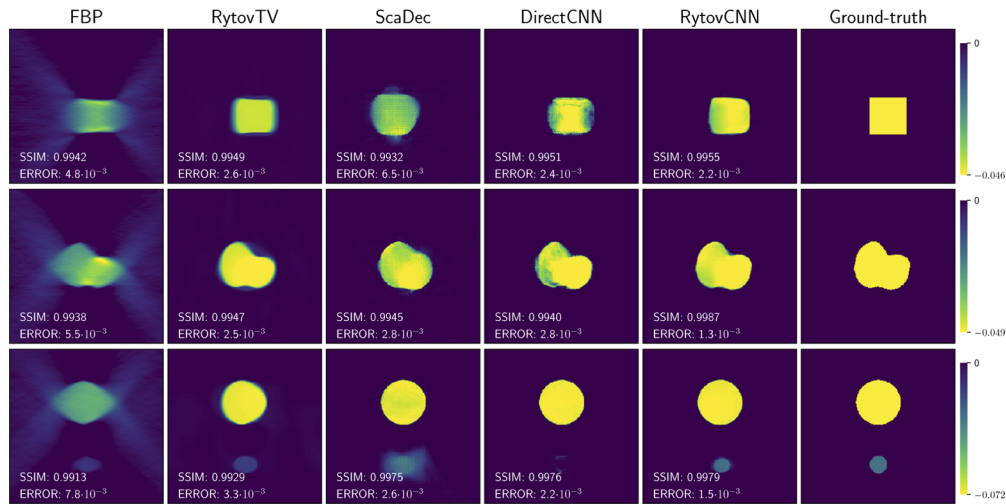
	20dB SNR	15dB SNR	10dB SNR
FBP	0.9915	0.9912	0.9890
RytovTV	0.9927	0.9926	0.9922
ScaDec	0.9991	0.9990	0.9987
DirectCNN	0.9993	0.9993	0.9984
RytovCNN	<b>0.9995</b>	<b>0.9994</b>	<b>0.9988</b>



**Fig. 4.** Reconstructions of simulated data for one disk. Our method is RytovCNN. From top to bottom, the SNR of the measurement is 20dB, 15dB, and 10dB. The corresponding RI variation is  $(-0.088)$ ,  $(-0.060)$ , and  $(-0.060)$ , respectively. The ground-truth images are presented in the last column.

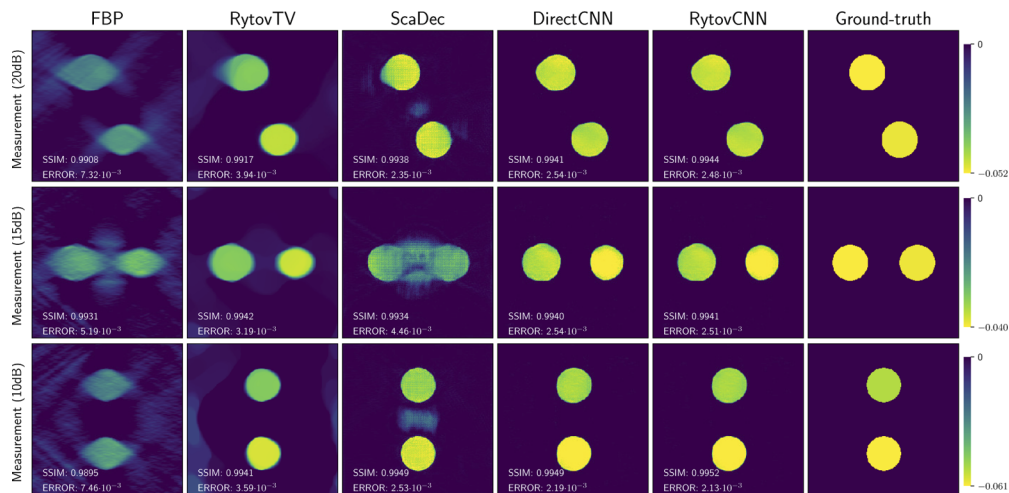
**Reconstruction of objects whose shape differ from the training data** We further assessed how the trained CNN performs when the testing set does not match the training set. We consider three examples with several shapes (see Fig. 5): a square, a cell-like sample, and two non-overlapping disks with independent radius and RI variation. In this case, the direct CNN and ScaDec both fail to recover the objects. However, our method leads to reconstructions with more accurate shapes and RI values because it incorporates a feedback mechanism that ensures consistency with the measurements. Furthermore, it performs slightly better than RytovTV. These illustrative examples suggest that our proposed method can perform well despite being trained on single disks. We emphasize that it may not perform well for all types of data, but it shows that the reconstructions may still be satisfactory for some testing data that does not match the training set.

In addition, we generated 18 other images containing two disks with arbitrary RI variation ranging from  $(-0.067)$  to  $(-0.033)$  (see Fig. 6). The rotation angle for this data was chosen between  $0^\circ$ ,  $22.5^\circ$ ,  $45^\circ$ , and  $90^\circ$ . In the  $0^\circ$  view, the two circles were placed vertically, while in the  $90^\circ$  view they were aligned horizontally. In this case, ScaDec performs less well, at the possible exception of the 10dB case (see Tables 4 and 5). Similarly, RytovTV leads to reconstructions with inaccurate shapes and visible artifacts in the background. On the contrary, both the direct



**Fig. 5.** Reconstructions of simulated data for several shapes. The SNR of the measurement is 20dB. From top to bottom, the object is a square, a cell-like sample, and two non-overlapping disks with independent radius and RI variation. The corresponding RI variation is  $(-0.046)$ ,  $(-0.049)$ , and  $(-0.072)$ , respectively. The ground-truth images are presented in the last column.

CNN and our method produce remarkably well recovered images. Moreover, our iterative scheme even offers a slight increase in the quality of the reconstruction. These results suggest that the proposed scheme is robust to the mismatch experienced in training on a single disk and testing on two.



**Fig. 6.** Reconstructions of simulated data for two disks. Our method is RytovCNN. From top to bottom, the SNR of the measurement is 20dB, 15dB, and 10dB. The corresponding RI variation is  $(-0.052)$ ,  $(-0.040)$ , and  $(-0.061)$ , respectively. The ground-truth images are presented in the last column.

**Table 4. ERROR of RI reconstructions in the presence of noise, for two disks.**

	20dB SNR	15dB SNR	10dB SNR
FBP	$6.23 \cdot 10^{-3}$	$6.57 \cdot 10^{-3}$	$6.85 \cdot 10^{-3}$
RytovTV	$3.58 \cdot 10^{-3}$	$3.93 \cdot 10^{-3}$	$4.36 \cdot 10^{-3}$
ScaDec <sup>a</sup>	$2.81 \cdot 10^{-3}$	$2.82 \cdot 10^{-3}$	<b><math>2.82 \cdot 10^{-3}</math></b>
DirectCNN <sup>a</sup>	$2.67 \cdot 10^{-3}$	$2.85 \cdot 10^{-3}$	$3.11 \cdot 10^{-3}$
RytovCNN <sup>a</sup>	<b><math>2.63 \cdot 10^{-3}</math></b>	<b><math>2.81 \cdot 10^{-3}</math></b>	$3.02 \cdot 10^{-3}$

<sup>a</sup>denotes the CNN is trained on data composed of one disk.

### 5.1.2. Training data composed of complex objects

**Simulation setup** We generated 1,116 images composed of 2D cell-like samples with two embedded smaller ellipses (see Fig. 7). The radius of the larger part ranged from  $10\lambda$  to  $12\lambda$ , while that of smaller ellipses ranged from  $1.5\lambda$  to  $2.5\lambda$ . The ellipses are arbitrarily located. The boundary of each large cell-like sample is randomly generated. Each object has its own RI variation ranging from  $(-0.1)$  to  $0.2$ . In this case, we set  $C = \{\mathbf{f} \in \mathbb{R}^N\}$ , and the SNR of the measurements is 20dB.



**Fig. 7.** Sixteen examples of the images randomly generated to train the CNN projector to reconstruct complex objects.

**Table 5. SSIM of RI reconstructions in the presence of noise, for two disks.**

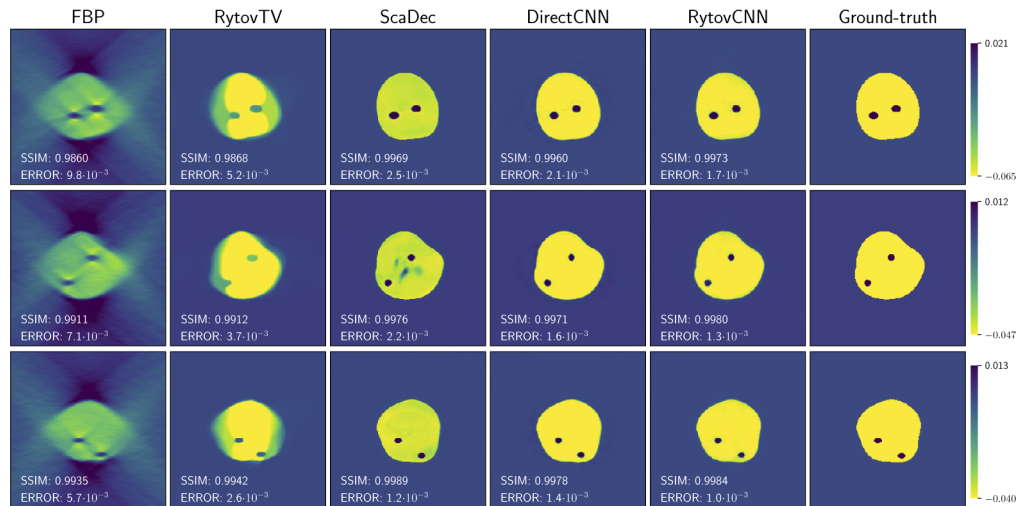
	20dB SNR	15dB SNR	10dB SNR
FBP	0.9917	0.9911	0.9896
RytovTV	0.9928	0.9928	<b>0.9927</b>
ScaDec <sup>a</sup>	0.9923	0.9923	0.9923
DirectCNN <sup>a</sup>	0.9932	0.9928	0.9921
RytovCNN <sup>a</sup>	<b>0.9933</b>	<b>0.9930</b>	0.9924

<sup>a</sup>denotes the CNN is trained on data composed of one disk.

**Reconstruction of complex objects** We give in Fig. 8 three examples of reconstructions. The average ERROR and SSIM over the whole testing data is also shown in Table 6. Notably, all the CNN-based approaches performed better than RytovTV. Moreover, our method (RytovCNN) still improves the quality of the reconstruction compared to ScaDec or direct CNN. Note that the columns BPMTV and BPMCNN will be discussed in Section 5.3.

### 5.2. Validation on experimental data

**Experimental setup** The experimental data were 2D cross sections of a 3D specimen with no variation along the y axis. The objects are two non-overlapping fibers placed in the medium



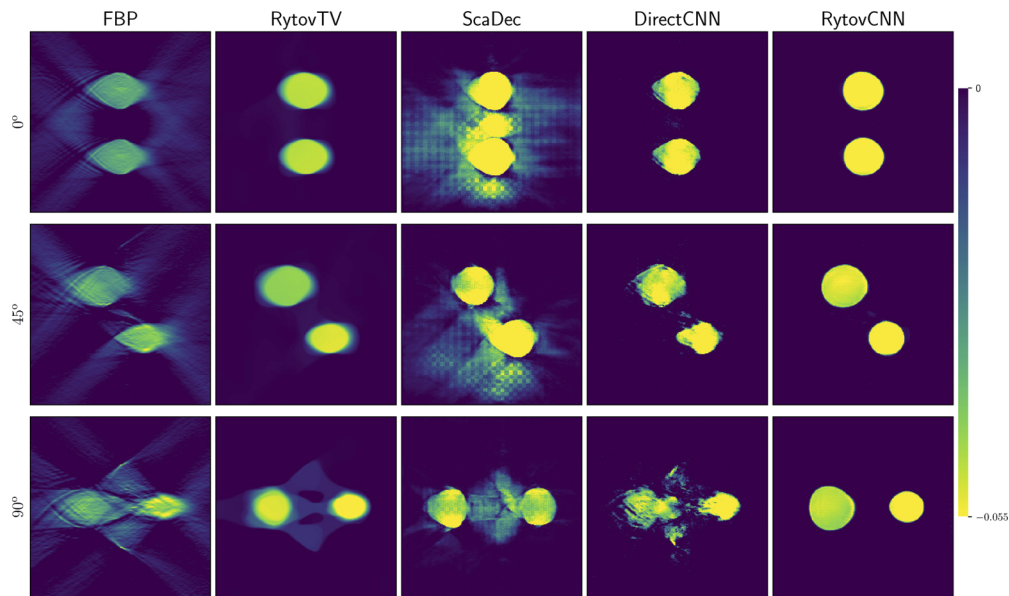
**Fig. 8.** Reconstructions of complex simulated data. The SNR of the measurement is 20dB. The corresponding RI variation is 0.086, 0.059, and 0.053, respectively. The ground-truth images are presented in the last column.

vertically, diagonally, and horizontally. The image size was  $(1024 \times 1024)$ . The reconstructed image size was cropped from the center to  $(512 \times 512)$ , which corresponds to  $(97\lambda \times 97\lambda)$ . The fibers had an expected RI variation of  $(-0.055)$  and a diameter equal to  $9 \mu\text{m}$ . In total,  $P = 160$  views were collected with the illumination of a laser diode (512 measurements per view). The other physical parameters replicated those of the simulation.

**Table 6. ERROR and SSIM of RI reconstructions in the presence of 20 dB noise, for complex objects. RytovCNN outperforms the other methods (the BPM-based methods are not considered).**

	FBP	RytovTV	ScaDec	DirectCNN	RytovCNN	BPMTV	BPMCNN
ERROR	$7.1 \cdot 10^{-3}$	$4.1 \cdot 10^{-3}$	$1.9 \cdot 10^{-3}$	$1.8 \cdot 10^{-3}$	<b><math>1.6 \cdot 10^{-3}</math></b>	$2.7 \cdot 10^{-3}$	<b><math>1.3 \cdot 10^{-3}</math></b>
SSIM	0.99	0.9906	0.9978	0.9969	<b>0.9983</b>	0.9944	<b>0.9987</b>

**Reconstructions** We trained the CNN over one disk. Since the experimental data have a size larger than the simulated data, we cropped the image before to feed it into the CNN and zero-padded the output correspondingly. The performance of the reconstruction with different methods is shown in Fig. 9. ScaDec produces blocky artifacts in all configurations. In addition, the direct CNN introduces deformation and underestimated RI. It fails spectacularly for the  $90^\circ$  case. RytovTV produces piecewise-constant structures, which suits well the dataset. Yet, artifacts exist and induce elongated shapes, which points to a weakness of this method toward the missing-cone problem. On the contrary, the proposed method generally obtains satisfying results in terms of RI and shape. Since the end-to-end approaches (ScaDec and DirectCNN) do not use the information from the measurement iteratively, they are more vulnerable to a potential mismatch between the synthetic training data and the real data. By contrast, our method and RytovTV are more robust because of the iterative information feedback from the measurement. Similar findings have been observed in [37]. Note that the most difficult case is  $90^\circ$ , because the missing-cone problem is acute for this configuration. Moreover, the considered forward model is least valid since multiple-scattering events occur in this setting.



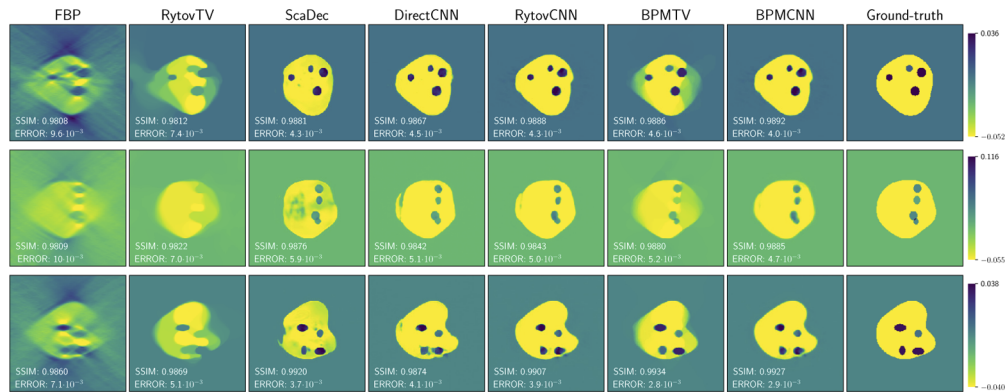
**Fig. 9.** Reconstructions of experimental data. From first to third row, the line that joins the center of the fibers is angled by  $0^\circ$ ,  $45^\circ$ , and  $90^\circ$  with respect to the vertical dimension, respectively. The variation of the fibers is  $(-0.055)$ .

### 5.3. Nonlinear forward model with deep-learning projector

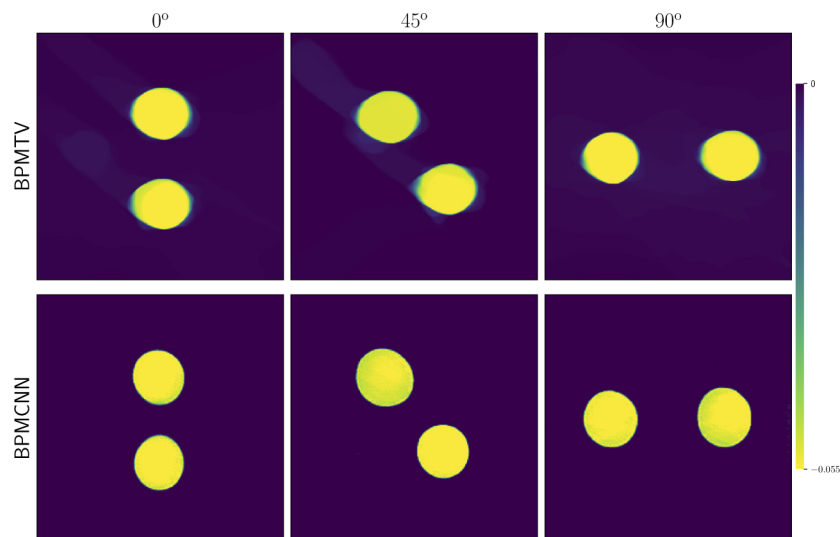
We combine the nonlinear forward model BPM [14,15] with the CNN based on the Rytov model and trained in the previous section. The idea is to assess the effect of our projector. Here, we present reconstructions of three synthetic data samples and one real data sample. The reconstructions of synthetic data were obtained with the CNN trained with complex objects in Section 5.1.2 with the Rytov model. Similarly, the reconstruction of real data was obtained with the CNN trained with single disks.

As shown in Fig. 10, RytovTV fails to restore the objects, while the TV-regularized BPM (BPM-TV) is able to recover the objects well. All the CNN-based methods perform better for recovering the shapes and RI values. The CNN projector combined with the BPM model (BPM-CNN) outperforms all the other methods for the whole testing of complex objects, as shown in Table 6.

Additionally, as shown in Fig. 11, the BPM model is able to reconstruct the samples with impressive improvement over the RytovCNN. These comparisons show that our framework is capable of reconstructing the samples even if the CNN projector was trained with the strategy described in Section 4.2 (*i.e.*, without knowledge of the BPM model).



**Fig. 10.** Reconstructions with a nonlinear forward model applied to synthetic data. From top to bottom, the value of the RI variation is 0.088, 0.171, and 0.078, respectively. The ground-truth images are presented in the last column. The training was done with simpler objects characterized by two embedded inclusions only.



**Fig. 11.** Reconstructions with a nonlinear forward model applied to experimental data. The value of the RI variation of the fibers is  $(-0.055)$ . The training was done with one disks.

## 6. Conclusion

To reconstruct data governed by the Helmholtz equation, we have proposed a general iterative framework that takes advantage of two worlds. On one hand, a deep-learning-based projector encodes priors learnt beforehand. On the other hand, the inversion method is based on a physical model and ensures consistency with the measurements. We have validated our approach in a challenging setting that combines the difficulties of the missing-cone problem with a nonlinear physical model. While our framework is trained on a simulated dataset that contains one object only, it remains applicable to experimental data featuring multiple objects in a variety of optical configurations, which is remarkable. In addition, we also validated the proposed approach on more complex object. Our numerical experiments have shown that the proposed method outperforms the conventional reconstruction of refractive indices in an optical diffraction tomography context, as well as recent deep-learning-based methods. Due to our combined approach (projector and physical model), we could mitigate the missing-cone problem. Furthermore, we have assessed the ability of our projector to work with a more advanced nonlinear forward model. The reconstructions were significantly improved upon those of the linear model.

## Funding

National Key Research and Development Program of China (2017YFB0202902); National Natural Science Foundation of China (41625017); European Research Council (692726).

## Acknowledgments

The authors would like to thank Joowon Lim and Prof. Demetri Psaltis for providing us with real data and the China Scholarship Council for supporting the visit of the first co-author.

## Disclosures

The authors declare no conflicts of interest.

## References

1. P. Liu, L. Chin, W. Ser, H. Chen, C.-M. Hsieh, C.-H. Lee, K.-B. Sung, T. Ayi, P. Yap, B. Liedberg, and K. Wang, "Cell refractive index for cell biology and disease diagnosis: Past, present and future," *Lab Chip* **16**(4), 634–644 (2016).
2. E. Wolf, "Three-dimensional structure determination of semi-transparent objects from holographic data," *Opt. Commun.* **1**(4), 153–156 (1969).
3. R. Dändliker and K. Weiss, "Reconstruction of the three-dimensional refractive index from scattered waves," *Opt. Commun.* **1**(7), 323–328 (1970).
4. M. Schürmann, J. Scholze, P. Müller, C. Chan, A. Ekpenyong, K. Chalut, and J. Guck, "Refractive index measurements of single spherical cells using digital holographic microscopy," *Methods Cell Biol.* **125**, 143–159 (2015).
5. Y. Sung and R. R. Dasari, "Deterministic regularization of three-dimensional optical diffraction tomography," *J. Opt. Soc. Am. A* **28**(8), 1554–1561 (2011).
6. J. Lim, K. Lee, K. H. Jin, S. Shin, S. Lee, Y. Park, and J. C. Ye, "Comparative study of iterative reconstruction algorithms for missing cone problems in optical diffraction tomography," *Opt. Express* **23**(13), 16933–16948 (2015).
7. B. Rappaz, P. Marquet, E. Cuhe, Y. Emery, C. Depeursinge, and P. J. Magistretti, "Measurement of the integral refractive index and dynamic cell morphometry of living cells with digital holographic microscopy," *Opt. Express* **13**(23), 9361–9373 (2005).
8. W. Choi, C. Fang-Yen, K. Badizadegan, S. Oh, N. Lue, R. R. Dasari, and M. S. Feld, "Tomographic phase microscopy," *Nat. Methods* **4**(9), 717–719 (2007).
9. Y. Sung, W. Choi, C. Fang-Yen, K. Badizadegan, R. R. Dasari, and M. S. Feld, "Optical diffraction tomography for high resolution live cell imaging," *Opt. Express* **17**(1), 266–277 (2009).
10. M. Born and E. Wolf, *Principles of Optics*, 7th (expanded) edition, vol. 461 (United Kingdom: Press Syndicate of the University of Cambridge, 1999).
11. A. Devaney, "Inverse-scattering theory within the Rytov approximation," *Opt. Lett.* **6**(8), 374–376 (1981).
12. T. C. Wedberg and J. J. Stamnes, "Experimental examination of the quantitative imaging properties of optical diffraction tomography," *J. Opt. Soc. Am. A* **12**(3), 493–500 (1995).
13. W. Strauss, "Nonlinear scattering theory," in *Scattering Theory in Mathematical Physics* (Springer, 1974), pp. 53–78.

14. U. S. Kamilov, I. N. Papadopoulos, M. H. Shoreh, A. Goy, C. Vonesch, M. Unser, and D. Psaltis, "Learning approach to optical tomography," *Optica* **2**(6), 517–522 (2015).
15. U. S. Kamilov, I. N. Papadopoulos, M. H. Shoreh, A. Goy, C. Vonesch, M. Unser, and D. Psaltis, "Optical tomographic image reconstruction based on beam propagation and sparse regularization," *IEEE Transactions on Comput. Imaging* **2**(1), 59–70 (2016).
16. A. Abubakar and P. M. van den Berg, "The contrast source inversion method for location and shape reconstructions," *Inverse Probl.* **18**(2), 495–510 (2002).
17. P. C. Chaumet and K. Belkebir, "Three-dimensional reconstruction from real data using a conjugate gradient-coupled dipole method," *Inverse Probl.* **25**(2), 024003 (2009).
18. U. S. Kamilov, D. Liu, H. Mansour, and P. T. Boufounos, "A recursive Born approach to nonlinear inverse scattering," *IEEE Signal Process. Lett.* **23**(8), 1052–1056 (2016).
19. H.-Y. Liu, D. Liu, H. Mansour, P. T. Boufounos, L. Waller, and U. S. Kamilov, "SEAGLE: Sparsity-driven image reconstruction under multiple scattering," *IEEE Transactions on Comput. Imaging* **4**(1), 73–86 (2018).
20. E. Soubies, T.-a. Pham, and M. Unser, "Efficient inversion of multiple-scattering model for optical diffraction tomography," *Opt. Express* **25**(18), 21786–21800 (2017).
21. L. I. Rudin, S. Osher, and E. Fatemi, "Nonlinear total variation based noise removal algorithms," *Phys. D* **60**(1–4), 259–268 (1992).
22. H. Lantéri, M. Roche, O. Cuevas, and C. Aime, "A general method to devise maximum-likelihood signal restoration multiplicative algorithms with non-negativity constraints," *Signal Process.* **81**(5), 945–974 (2001).
23. U. S. Kamilov, H. Mansour, and B. Wohlberg, "A plug-and-play priors approach for solving nonlinear imaging inverse problems," *IEEE Signal Process. Lett.* **24**(12), 1872–1876 (2017).
24. Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *Nature* **521**(7553), 436–444 (2015).
25. I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning* (MIT Press, 2016).
26. J. Schlemper, J. Caballero, J. V. Hajnal, A. Price, and D. Rueckert, "A deep cascade of convolutional neural networks for MR image reconstruction," in *International Conference on Information Processing in Medical Imaging* (Springer, 2017), pp. 647–658.
27. C. Dong, C. C. Loy, K. He, and X. Tang, "Image super-resolution using deep convolutional networks," *IEEE Transactions on Pattern Analysis Mach. Intell.* **38**(2), 295–307 (2016).
28. K. H. Jin, M. T. McCann, E. Froustey, and M. Unser, "Deep convolutional neural network for inverse problems in imaging," *IEEE Transactions on Image Process.* **26**(9), 4509–4522 (2017).
29. K. Hammernik, T. Würfl, T. Pock, and A. Maier, "A deep learning architecture for limited-angle computed tomography reconstruction," in *Bildverarbeitung für die Medizin 2017* (Springer, 2017), pp. 92–97.
30. A. Adler, D. Boubilil, and M. Zibulevsky, "Block-based compressed sensing of images via deep learning," in *IEEE 19th International Workshop on Multimedia Signal Process. (MMSP)* (IEEE, 2017), pp. 1–6.
31. Y. LeCun, K. Kavukcuoglu, and C. Farabet, "Convolutional networks and applications in vision," in *Proceedings of IEEE International Symposium on Circuits and Systems* (IEEE, 2010), pp. 253–256.
32. M. T. McCann, K. H. Jin, and M. Unser, "Convolutional neural networks for inverse problems in imaging: A review," *IEEE Signal Process. Mag.* **34**(6), 85–95 (2017).
33. H. K. Aggarwal, M. P. Mani, and M. Jacob, "MODL: Model-based deep learning architecture for inverse problems," *IEEE Transactions on Med. Imaging* **38**(2), 394–405 (2019).
34. J. Adler and O. Öktem, "Solving ill-posed inverse problems using iterative deep neural networks," *Inverse Probl.* **33**(12), 124007 (2017).
35. J. H. Rick Chang, C.-L. Li, B. Póczos, B. V. K. Vijaya Kumar, and A. C. Sankaranarayanan, "One network to solve them all – solving linear inverse problems using deep projection models," in *The IEEE International Conference on Computer Vision (ICCV)*, (2017), pp. 5888–5897.
36. S. V. Venkatakrishnan, C. A. Bouman, and B. Wohlberg, "Plug-and-play priors for model based reconstruction," in *IEEE Global Conference on Signal and Information Processing*, (IEEE, 2013), pp. 945–948.
37. H. Gupta, K. H. Jin, H. Q. Nguyen, M. T. McCann, and M. Unser, "CNN-based projected gradient descent for consistent CT image reconstruction," *IEEE Transactions on Med. Imaging* **37**(6), 1440–1453 (2018).
38. H. Qiao, J. Wu, X. Li, M. H. Shoreh, J. Fan, and Q. Dai, "GPU-based deep convolutional neural network for tomographic phase microscopy with  $l_1$  fitting and regularization," *J. Biomed. Opt.* **23**(06), 1 (2018).
39. Y. Sun, Z. Xia, and U. S. Kamilov, "Efficient and accurate inversion of multiple scattering with deep learning," *Opt. Express* **26**(11), 14678–14688 (2018).
40. O. Ronneberger, P. Fischer, and T. Brox, "U-Net: Convolutional networks for biomedical image segmentation," in *International Conference on Medical Image Computing and Computer-Assisted Intervention*, (Springer, 2015), pp. 234–241.
41. L. Li, L. G. Wang, F. L. Teixeira, C. Liu, A. Nehorai, and T. J. Cui, "Deep NIS: Deep neural network for nonlinear electromagnetic inverse scattering," *IEEE Trans. Antennas Propag.* **67**(3), 1819–1825 (2019).
42. T. Nguyen, V. Bui, and G. Nehmetallah, "3D optical diffraction tomography using deep learning," in *Digital Holography and Three-Dimensional Imaging* (Optical Society of America, 2018), pp. DW2F–4.
43. D. P. Bertsekas, *Convex optimization algorithms* (Athena Scientific, Belmont, Massachusetts, 2015).
44. A. L. Maas, A. Y. Hannun, and A. Y. Ng, "Rectifier nonlinearities improve neural network acoustic models," in *ICML Workshop on Deep Learning for Audio, Speech and Language Processing*, vol. 28 (Citeseer, 2013).



45. D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," arXiv preprint arXiv:1412.6980 (2014).
46. N. Ketkar, "Introduction to PyTorch," in *Deep Learning with Python*, (Springer, 2017), pp. 195–208.
47. E. Soubies, F. Soulez, M. McCann, T.-a. Pham, L. Donati, T. Debarre, D. Sage, and M. Unser, "Pocket guide to solve inverse problems with Global Bio Im," *Inverse Probl.* **35**(10), 104006 (2019).
48. A. Hore and D. Ziou, "Image quality metrics: PSNR vs. SSIM," in *International Conference on Pattern Recognition* (IEEE, 2010), pp. 2366–2369.
49. P. L. Combettes and V. R. Wajs, "Signal recovery by proximal forward-backward splitting," *Multiscale Model. Simul.* **4**(4), 1168–1200 (2005).