

2019

Deep learning in computational microscopy

Thanh Nguyen, George Nehmetallah, Lei Tian. 2019. "Deep learning in computational microscopy." Computational Imaging IV. Computational Imaging IV. <https://doi.org/10.1117/12.2520089>
<https://hdl.handle.net/2144/40200>

"Downloaded from OpenBU. Boston University's institutional repository."

PROCEEDINGS OF SPIE

SPIDigitalLibrary.org/conference-proceedings-of-spie

Deep learning in computational microscopy

Nguyen, Thanh, Nehmetallah, George, Tian, Lei

Thanh Nguyen, George Nehmetallah, Lei Tian, "Deep learning in computational microscopy," Proc. SPIE 10990, Computational Imaging IV, 1099007 (13 May 2019); doi: 10.1117/12.2520089

SPIE.

Event: SPIE Defense + Commercial Sensing, 2019, Baltimore, Maryland, United States

Deep Learning In Computational Microscopy

Thanh Nguyen¹, George Nehmetallah¹, and Lei Tian²

¹*EECS Department, The Catholic University of America, 620 Michigan Av., N.E., Washington DC 20064*

²*Department of Electrical and Computer Engineering, Boston University, Boston, MA 02215*
nehmetallah@cua.edu

ABSTRACT

We propose to use deep convolutional neural networks (DCNNs) to perform 2D and 3D computational imaging. Specifically, we investigate three different applications. We first try to solve the 3D inverse scattering problem based on learning a huge number of training target and speckle pairs. We also demonstrate a new DCNN architecture to perform Fourier ptychographic Microscopy (FPM) reconstruction, which achieves high-resolution phase recovery with considerably less data than standard FPM. Finally, we employ DCNN models that can predict focused 2D fluorescent microscopic images from blurred images captured at overfocused or underfocused planes.

1. INTRODUCTION

Recently, deep convolutional neural networks (DCNNs) have been used to solve many imaging problems such as denoising, deconvolution, super-resolution, image classification, segmentation, phase imaging and imaging through scattering media, providing state-of-the-art performance with unmatched results [1-10].

In this work, we will first discuss the application of DCNN in 3D lensless computational imaging [1]. Specifically, we implement 3D deep convolutional neural networks (3D-DCNNs) to perform 3D computational optical phase reconstruction. For this end, we construct a database of synthetic 3D phantom datasets and demonstrate the ability of 3D-DCNN to reconstruct the 3D phase images which constitutes the 3D distributions of the index of refraction of the sample objects from their corresponding diffraction patterns. In the experimental optical setup, the phase objects are displayed on a spatial light modulator (SLM) and the diffracted intensity images are recorded on a CCD.

Secondly, we demonstrate a new DCNN architecture to perform Fourier ptychographic Microscopy (FPM) reconstruction, which achieves high-resolution phase recovery with considerably less data than standard FPM [11]. This novel deep learning framework can significantly reduce both the data requirement and reconstruction time in a Fourier ptychographic microscopy (FPM) system. In particular, the novel DCNN architecture combines a modified Unet structure and a conditional generative adversarial network (cGAN) network to perform high-speed FPM phase retrieval and with much reduced number of images required.

Thirdly, while automatic microscopy systems play an important role in acquiring large image datasets, these systems may introduce blurring effect when the sample is not in focus. Traditionally, automatic defocusing algorithms have been widely used in microscopic imaging systems for obtaining a clean sharp image. However, these defocusing algorithms often need a knowledge of the system's point spread function (PSF) or use iterative techniques which are time consuming [12]. A previous work used the same data set to predict the defocus level of images not about defocusing the images [13]. Here we present a method of using convolutional neural network as an autofocusing tools to enhance the sharpness of images as if taken at the focal plane.

2. EXPERIMENTAL SETUP AND DATA PREPARATION FOR LENSLESS 3D ODT

Optical diffraction tomography (ODT) is a promising technique that takes into account diffraction due to the internal refraction index distribution that is comparable in size to the wavelength [14]. Recently, researchers achieved considerable improvement in resolving the refraction index of sub-cellular structures using ODT with visible light [15]. However, ODT not only requires multiple capturing in different designed illuminations, which is necessary for tomographic reconstruction, but also phase retrieval techniques such as digital holographic microscopy [16], transport of intensity equation [17], and intensity diffraction tomography [18]. Iterative reconstruction techniques have become the dominant approach to solving

various inverse problems in imaging such as deconvolution [19] and denoising [20]. Since inverse scattering problem is an ill-posed inverse problem, other class of techniques which are based on compressed sensing and regularization to prevent overfitting have been proposed [21]. These techniques are generally based on the L_2 regularization used for smooth signals [22] or the L_1 regularization used for sparse signals [23]. Compressive sensing and regularization techniques have resulted in good image quality and less computational complexity which are two important objectives in the field of biomedical imaging such as MRI and CT. However, most of the current techniques still have certain limitations, and it is difficult to obtain a technique that is fast, high-resolution, portable, inexpensive, large field-of-view, and needs simple setup.

Specifically, we implement 3D deep convolutional neural networks (DCNNs) to perform 2D and 3D computational optical image reconstruction. We experimentally demonstrate by using a synthetic 3D phantom datasets as phase objects the ability of a DCNN to reconstruct 3D distributions of the index of refraction of the sample objects from their corresponding diffraction patterns. In the optical setup, the phase objects are displayed on a spatial light modulator (SLM) and the diffracted intensity images are recorded on a CCD. The inverse scattering problem is solved based on learning a huge number of training target and speckle pattern pairs. The proposed technique does not rely on a reference beam, thus employs a simpler optical setup than previous techniques based on the transfer matrix (TM) approach enabling model-free imaging without the need to know the underlying optical processes which is very important since many optimization techniques are very sensitive to errors caused by the inaccuracy of the forward model [2,24].

The synthetic phantoms created in this work, consist of 3D index of refraction distributions $\hat{n}_{x,y,z}$ mimicking 3D biological samples that are usually reconstructed using OPT/ODT setups. In an OPT/ODT setup, the illuminating beam exiting the test object is diffracted before reaching a CCD camera which captures a 2D raw intensity diffraction pattern. In a tomographic setup (either the sample is rotated or the illuminating beam is rotated keeping the sample stationary), an estimate of the internal structure of the test object $\hat{n}_{x,y,z}$ can be recovered from the recorded 3D angle-stacked set $I_{x,y,\theta}$ of the 2D raw diffraction patterns $I_{x,y}$ by solving the inverse scattering problem:

$$\hat{n}_{x,y,z} = H^{inv}(I_{x,y,\theta}) \quad (1)$$

with $H(n_{x,y,z}) = I_{x,y,\theta}$ denotes the forward operator through the optical system which relates the index of distribution to the captured intensity. In this work, 3D-DCNNs were chosen to solve this inverse problem instead of the traditional iterative optimization techniques [23].

A schematic diagram of the optical experimental setup is shown in Fig. 1, in which the light from a HeNe laser ($\lambda = 632.8\text{nm}$), passes through a spatial filter (SF), which contains a microscope objective (MO, 0.25 NA) and a pinhole ($10\ \mu\text{m}$), to be collimated by a collimating lens (CL, $f = 100\ \text{mm}$). The collimated beam passes through a linear polarizer (LP) to be modulated by a reflective phase only SLM (Holoeye, VIS-014), after being reflected by a beam splitter (BS). The phase-modulated beam is reflected from the SLM back to the BS, then to a linear polarizer analyzer (LPA), and a $4f$ relay system (in tomography configuration) before being recorded by a CCD (Lumenera L100, 1280×1024 , $5.2\ \mu\text{m}$ pixel) on an under-focused plane at a distance of 5 cm from the image plane.

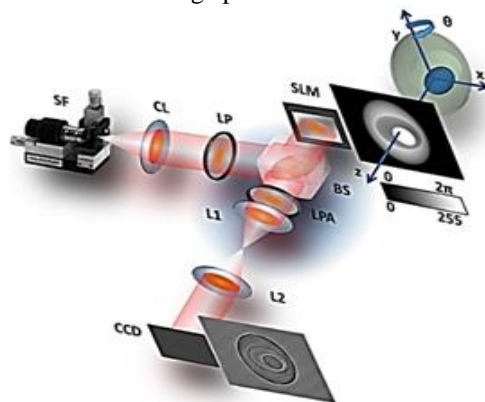


Fig. 1. Experimental optical imaging setup of a 3D tomographic configuration.

2.1 Training and validation of the 3D-DCNN computational optical tomography model

In order to create the 3D index of refraction distributions of test objects, 3D matrices were numerically simulated, in which each voxel represents a specific index of refraction. Each test object contains three large, medium, and small ellipsoids with random sizes and random locations, but with constant indices of refraction $n_L=1.365$, $n_M=1.340$, and $n_S=1.387$,

respectively. The index of refraction of the surrounding medium is: $n_{Med}=1.333$. To mimic the interaction of an optical beam with a test object, at each illuminating angle the 2D optical path lengths were computed as a 2D Radon transform of the 3D index of refraction distribution over the whole volume, as

$$OPL_{(\rho,\theta,y)} = \Gamma\{n(x,y,z)\} = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} n(x,y,z) \delta(x\cos\theta + z\sin\theta - \rho) dx dz \quad (2)$$

which will result in a set of 2D phase images where each phase image $\varphi_{\rho,\theta_i,y} = k_0 \cdot OPL_{\rho,\theta_i,y}$ corresponds to each of the projections orthogonal to θ_i , where k_0 is the wave number in free space. These phase images are displayed on the SLM. Due to the limited phase modulation of the SLM, the size and the indices of refraction of the object's ellipsoids are calibrated in the process so that the maxima of the 2D phase images don't exceed: $\phi_{max} = k_0 \cdot OPL_{max} = 2\pi$.

Initially, 240 test objects with sizes of 360x360x360 voxels each (voxel resolution = SLM's pixel size = 8 μ m) were simulated. Corresponding to a single test 3D object, a set of 360 phase images created using Eq. (2). Each of these phase images has a size of 360x360 pixels. Then, 360 diffractive images with angular resolution of 1 $^\circ$ were captured on a CCD for each of the phase images. The inverse Radon Transform (IRT) was implemented on each set consisting of 360 diffracted intensity images. At each height y_i , the IRT can be described as:

$$I_{(x,y_i,z)} = \int_0^\pi \left[\int_0^\infty h|\omega| M_{(\omega,y_i,\theta)} e^{j2\pi\omega\rho} d\omega \right]_{\rho=x\cos\theta+z\sin\theta} d\theta, \quad (3)$$

where $M_{\omega,y_i,\theta} = I(\omega\cos\theta, y_i, \omega\sin\theta)$ is the 2D Fourier transform of $I_{\omega,y_i,z}$, h is a hamming window. Since the dimensions of the SLM are 1920x1080 pixels, zero padding was used. The CCD used has 1280x1024 pixels. Due to the difference in pixel size between the CCD (pixel size = 5.2 μ m) and the SLM, image registration was performed to match the recorded diffracted images to the back projection images. The diffracted images were normalized to [0, 1] range based on the global maximum and minimum pixel value of the entire dataset.

2.2 Data pre-processing and visualization

Fig. 2(a) shows three orthogonal cross-sections at the center of the stack of the diffracted images captured by the CCD. For each synthetic phantom, the stack (sinogram) consists of 360 layers corresponding to the 360 projection angles. Using the sinogram as input to the DCNN model resulted in poor training. Fig. 2(b) is the inverse radon transform (IRT) of Fig. 2(a). When IRT is performed on the input data to the DCNN the training process was successful. Fig. 2(c) shows three orthogonal cross-sections at the center of the synthetic phantom used in the training process as ground truth.

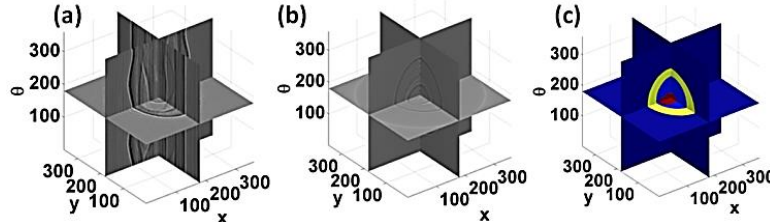


Fig. 2. Three orthogonal cross-sections along (x, y, θ) at the center of the stack of the: (a) diffracted images captured by the CCD, (b) same stack after using IRT, and (c) ground truth – 3D index of refraction distribution.

2.3 Methodology of 3D ODT

The proposed DCNN model, which is inspired from the U-net model [25], contains an input layer which is connected to the output layer by hidden layers containing nodes in which the data are convolved with the filters' weights, activated with rectified linear units (RELU), normalized with batch-normalization (BN), and sub-sampled with maxpooling or up-sampled with unpooling layers. Skip (concatenate) connections were implemented to transfer the information from the initial layers down the network by concatenating the layers before the maxpooling layers to the ones after the unpooling layers. Stochastic gradient descent (SGD) was used [26] with learning rate of 10^{-4} and decay step of 10^{-5} to update the weight parameters in the gradient of the loss function until it converges to a minimum. During the training, the mean squared error (MSE) (L_2 norm) was used as a loss function between the network output U and the 2D/3D ground truth of the training data set (G), which is presented as:

$$MSE = \frac{1}{xyz} \sum_{x,y,z} [U_{x,y,z} - G_{x,y,z}]^2. \quad (4)$$

The proposed 3D tomographic reconstruction based on 3D-DCNN model is implemented on a 3D convolutional architecture (see Fig. 3). An output neuron in this modified U-net model is computed through convolution operations (which we define as a convolution layer) with the preceding neurons connected to it such that these input neurons are

situated in a local spatial region of the input. Specifically, each output neuron value at position (x, y, z) in the j^{th} feature map in the i^{th} layer, is denoted as v_{ij}^{xyz} and is given by:

$$v_{ij}^{xyz} = \sum_m \sum_{p=0}^{P_i-1} \sum_{q=0}^{Q_i-1} \sum_{r=0}^{R_i-1} W_{ijm}^{pqr} v_{(i-1)m}^{(x+p)(y+q)(z+r)} + B_{ij}, \quad (5)$$

where W_{ijm}^{pqr} is the $(p, q, r)^{th}$ value of the kernel connected to the m^{th} feature map in the previous layer ($(i-1)^{th}$ layer), B_{ij} is the bias of the j^{th} feature map at the i^{th} layer, P_i, Q_i, R_i , are the height, width, and depth of the filter kernel. The maxpooling operation keeps the max value in a block of size $2 \times 2 \times 2$ and the unpooling operation repeats a value of a block of size $1 \times 1 \times 1$ to a block of size $2 \times 2 \times 2$.

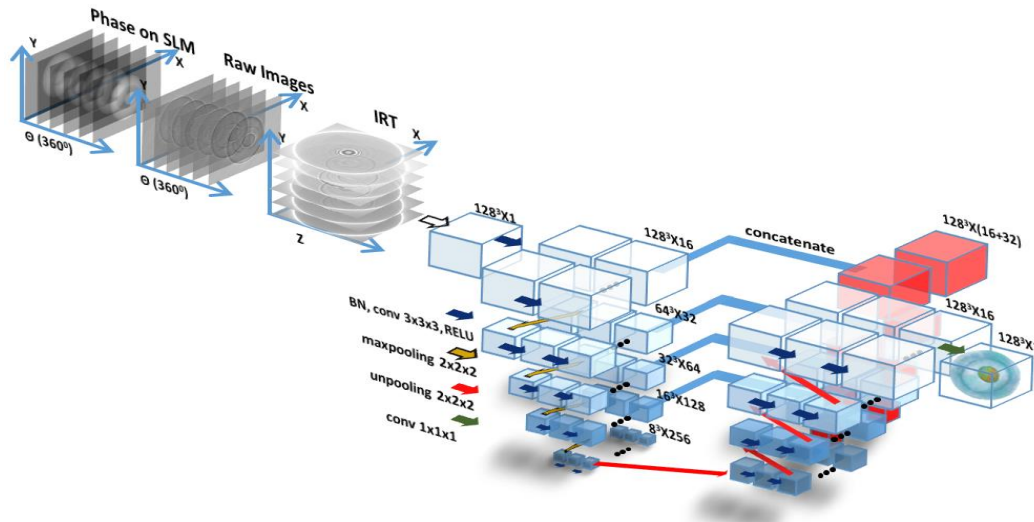


Fig. 3. Schematic of the 3D-DCNN model. The layers at the same level have the same dimensions.

The training was conducted in two different cases: (a) with and (b) without using the IRT (input to the DCNN is a 3D sinogram) as shown in Eq. (3). In both cases, the dataset which contains 240 objects was split into a training dataset of 200 objects, another validation dataset of 20 test objects, and the rest as a testing dataset. The model was trained with 100 epochs, and in each epoch a pair of test objects was randomly picked 100 times (batch size = 2) to feed into the model. The computation was performed on an Intel i7 CPU equipped with a GPU NVIDIA GeForce Titan Xp, and implemented using the Keras/Tensorflow backend framework.

2.4 ODT Results and network analysis

In order to evaluate the training process, we observe the loss values (MSE) of the training and validation across 100 epochs, as shown in Fig. 4(a). The plots show that without using IRT, the training loss converges in a stable manner. However, the validation loss did not converge, and kept oscillating around MSE ~ 0.55 starting from epoch 10 which leads to poor performance of the testing data. On the other hand, when IRT is used, both training loss and validation loss converge in a stable fashion without oscillation. The loss value in the proposed model decreases quickly and the model parameters kept changing until a minimum is achieved. In this study, we stopped at the 100th epoch because of the lack of improvement in the validation loss. Fig. 4(b) shows the loss value during the training process using IRT with and without using data augmentation. Due to the limited number of objects in a dataset, augmentation was used to increase the number of representative features of the dataset. In our case, data is augmented by rotating each 3D object by 90^0 in either $x, y,$ or z -axis before the training process. By doing that, the model is trained with more generalized features. As shown in Fig. 4(b), validation loss with augmentation is lower than without augmentation using the same number of epochs. Therefore, using augmentation or a larger real dataset will increase the performance of the model. Fig. 5 shows the 3D index of refraction distribution of two typical objects in the testing dataset. Fig. 5 (a) (left) shows the 3D transparent rendering using thresholds of 1.3335 and (right) the 3D contours before training with random initial weights. Fig. 5(b) (left) shows the transparent rendering using thresholds of 1.361 (above) and 1.362 (below) to avoid occlusion and (right) the 3D contours after training. Fig. 5(c) shows the 3D contours of the corresponding ground truth tomograms. Therefore, the medium is visualized as dark color to be able to see the small ellipsoids.

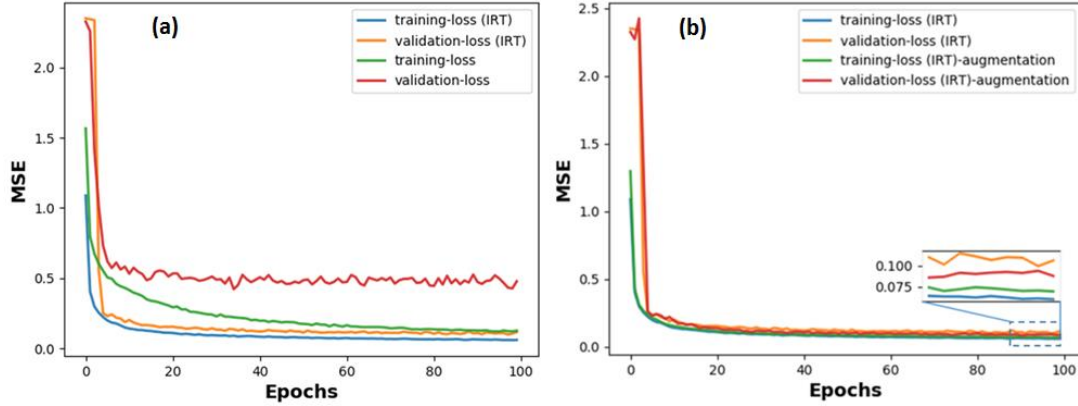


Fig. 4: Mean square error of the training and validation processes in 100 epochs (a) with and without using the IRT and (b) using (IRT) with and without augmentation.

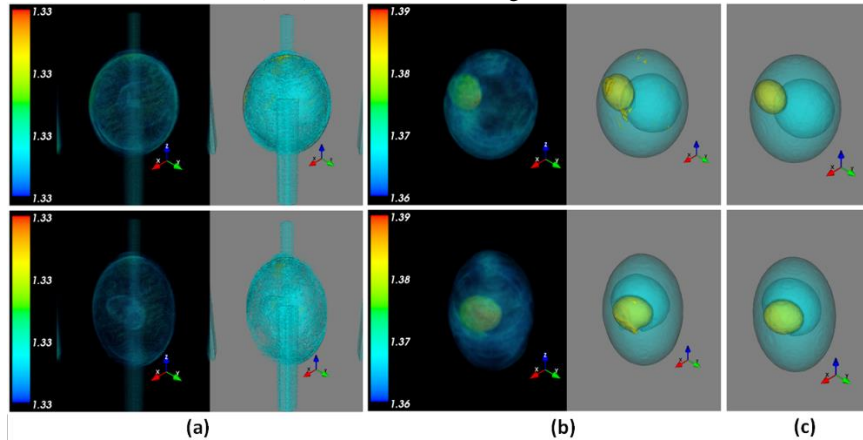


Fig. 5: Transparent 3D rendering and 3D contours of the index of refraction distribution of two typical objects from the testing dataset (a) before training with random initial weights and (b) after training. (c) 3D contours for ground truth.

3. EXPERIMENTAL SETUP AND DATA PREPARATION FOR FOURIER PTYCHOGRAPHY VIDEO RECONSTRUCTION

The proposed CNN based Fourier ptychographic microscopy (FPM) [27, 28] reconstruction algorithm takes a set of low-resolution intensity images I_α as the network input and output a single high-resolution phase image ϕ_G . The intensity images I_α are captured from illuminating the sample from α different illumination angles (LEDs) (Figure 6(a)), in which α_{BF} are brightfield (BF) and α_{DF} are darkfield (DF) (Figure 7). In the training stage, the ground truth phase image is fed into the CNN, which is obtained from the reconstructed high-resolution phase ϕ_{FPM} from the FPM algorithm [27] (Figure 6(b)). A key feature of the FPM is to reconstruct a high-resolution phase image using a set of low-resolution intensity images. The resolution enhancement factor is r in each dimension. To obtain the ground truth, it needed to capture the full FPM dataset, containing 173 images [27]. Since our DL scheme only requires training for the first ‘FPM frame’, the rest of the frame only requires α (< 173) images, which allows reducing the acquisition time, especially in a time-series experiment. We denote the set of α low-resolution images I_α as a tensor of dimension $W \times H \times \alpha$ and the corresponding ground truth ϕ_{FPM} a tensor of dimension $rW \times rH \times 1$ (Figure 6(b)).

The proposed CNN that performs FPM video reconstruction (Figure 6(c)) is based on the cGAN framework. It consists of two sub-networks, the generator G and the discriminator D (Figure 7). Here, the goal of the generator G is to be trained to predict a high-resolution phase $\phi_G = G(I_\alpha)$ from the low-resolution image set I_α input. To simplify the notation, we will drop the subscript α knowing that I will always contains α low-resolution intensity images. The generator network G consists of a set of parameters θ_G (weights and biases), which will be optimized through the training. The optimal θ_G is learned by minimizing a loss function l over N input-output training pairs:

$$\widehat{\theta}_G = \operatorname{argmin}_{\theta_G} \sum_{n=1}^N \frac{1}{N} l(G_{\theta_G}(I_n, \phi_n)). \quad (6)$$

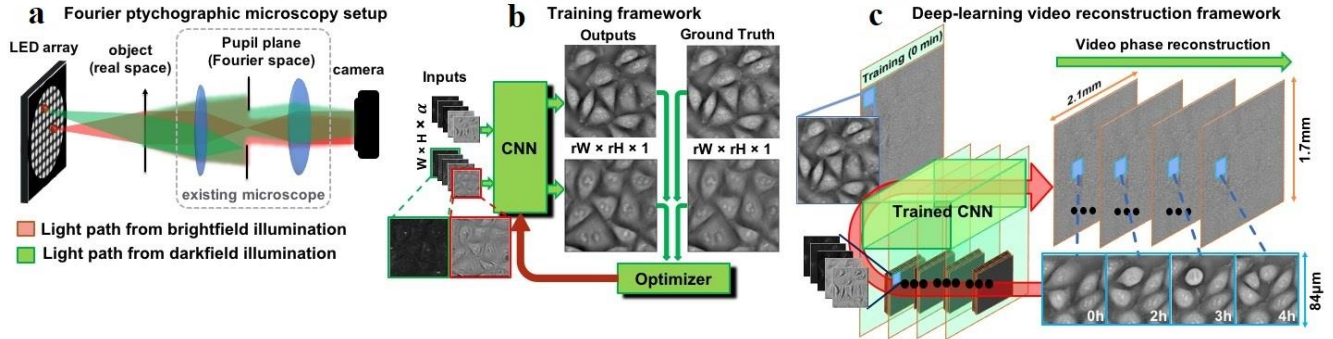


Fig. 6: The workflow of the proposed deep learning based Fourier ptychography video reconstruction. (a) The intensity data is captured by illuminating the sample from different angles with an LED array. (b) Training CNN to reconstruction high-resolution phase images. The input to the CNN are low-resolution intensity images; the output of the CNN is the ground truth phase image reconstructed using the traditional FPM algorithm in²⁷. The network is then trained by optimizing network's parameters that minimizes a loss function calculated based on the network's predicted output and the ground truth. (c) The network is fully trained using the first dataset at 0 min, then can be used to predict phase videos of dynamic cell samples over the course of 4 hours.

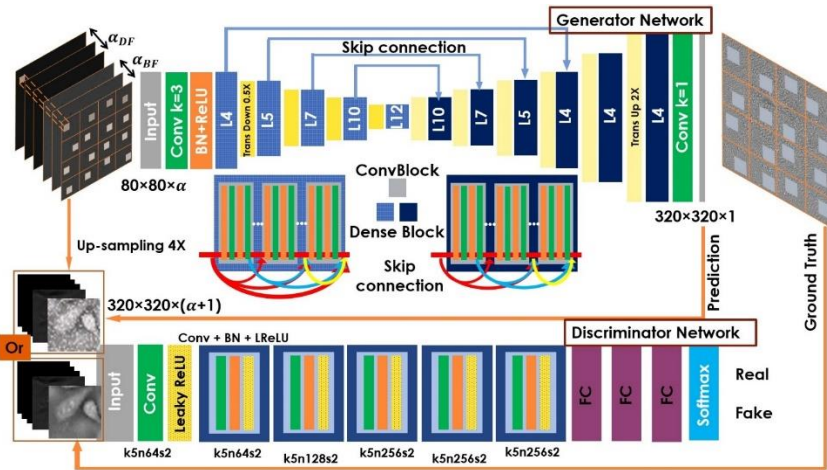


Fig. 7: The proposed condition generative adversarial network (cGAN) for FPM video reconstruction. The generator (top) and the discriminator (bottom) are constructed with the ConvBlock BN-ReLU-Conv(1 × 1)-BN-ReLU-Conv(3 × 3) and ConvBlock Conv-BN LeakyReLU, respectively. The generator output is the high-resolution phase. The discriminator tries to distinguish if that output phase is fake or real. The generator uses the UNet architecture. For the discriminator, the generator predicted phase or the ground truth phase is concatenated with the up-sampled intensity data as a conditional input to the discriminator network. The following color schemes are used: the two blocks ■ and ■ describe the dense concatenation inside the dense block in down-sampling and up-sampling path, respectively. ■ and ■ are transition layers interweaving with the dense blocks in the generator. ■ denotes the convolutional layer, ■ denotes the batch-normalization with a nonlinear ReLU layer in generator model, and ■ the batch-normalization with the leaky ReLU in the discriminator. In the last three layers of the discriminator, ■ denotes fully-connected layers for high-level feature reasoning. ■ is used at the end for binary classification. k#n#s# (# stands for some integer) denotes the stride size, number of channels, and stride of the convolution layer, respectively.

We emphasize that the choice of the loss function l significantly affects the quality of the training. We propose a mixed loss function that takes the weighted sum of multiple elementary loss functions. The generator G adopts the general "encoder-decoder" architecture used in UNet [25] to facilitate efficient learning of pixel-to-pixel information. UNet has shown to increase the network's performance by adapting to the high-complexity information in image dataset. To enhance the efficiency of the training process, batch-normalization (BN) is used to offset the internal covariate shift [29]. In addition, dropout regularization [30] is employed to constrain network's adaptation to the data during the training to avoid overfitting and increase the network's model accuracy. A known problem of training a CNN is that it can get saturated when the network's depth becomes too deep [31]. To mitigate this problem, the dense block (DB) proposed in the densely connected

network is used [32]. A DB connects each layer to its subsequent layers in a feed-forward fashion. The inputs to each layer are the feature-maps of all preceding layers; the output of the current layer's own feature-maps are inputs to all the subsequent layers (see Figure 7). The DB has several advantages, including (a) mitigation of the vanishing-gradient problem in the training; (b) reduction of the total number of parameters; (c) enhancement of feature propagation and reuse. A typical L -layer DB is shown in Figure 7 [33,34].

The discriminator network D aims to distinguish if the output from G is real or fake. Following Goodfellow [35] and Isola [36], we define a conditional Generative Adversarial Network (cGAN) to solve the following adversarial min-max problem:

$$\min_{\theta_G} \max_{\theta_G} E_{I,\phi} [\log D_{\theta_G}(I, \phi)] + E_I [\log(1 - D_{\theta_G}(I, G(I)))] \quad (7)$$

where ϕ is the phase map. The general idea behind this network is that it aims to train a generator G to 'fool' the discriminator D . Here, D is trained to distinguish whether the high-resolution phase image predicted by G represents a real phase image. It was observed that GAN in general is hard to train and it may fail when the generator collapses to a parameter setting where it always gives the same output. A successful strategy to avoid this failure is to allow the discriminator to perform minibatch discrimination [37]. In this case, the discriminator distinguishes if the reconstructed phase image is real or fake by evaluating multiple subregions of the G -predicted image instead of the whole.

3.1 Loss function and data preparation, training, evaluation, and testing

The motivation of the usage of the discriminator network D is that the commonly used pixel-wise loss functions, such as the mean absolute error (MAE), mean square error (MSE), and structural similarity index (SSIM), may not be the most appropriate figures of merit, in particular when assessing a CNN's performance in preserving high frequency content of reconstructed images. The minimization of these pixel-wise loss functions can lead to solutions that ignores the high-frequency details, while favors solutions that are smooth, albeit have less perceptual quality [38]. With cGAN approach, the generator G can learn to create a solution that resembles realistic high-resolution images with high-frequency details. For this purpose, we define the 'perceptual loss function' l as a weighted sum of multiple loss functions. This ensures that the model can learn the desired features containing both low-frequency and high-frequency information in the phase images. Specifically, our loss function consists of four components, including the pixel-wise spatial domain mean-absolute error (MAE) loss l_{MAE} , the pixel-wise Fourier domain mean-absolute error (FMAE) loss l_{FMAE} , the generator's adversarial loss l_G , and the weight regularization l_{θ_G} , written in the following form [11]:

$$l = \lambda_1(\beta_1 l_{\text{MAE}} + \beta_2 l_{\text{FMAE}}) + \lambda_2 l_G + \lambda_3 l_{\theta_G} \quad (8)$$

where $\lambda_1, \beta_1, \beta_2, \lambda_2, \lambda_3$ are hyper-parameters that controls the relative weights of each loss components. In practice, we found that the Fourier loss function is sensitive to pixel-wise corruption during the early stage of the training process. As a result, we use it only to refine the outputs by enforcing similarity in the frequency domain after initial training is done with the other three loss components. To test our CNN technique, we use FPM video data from [27]. The time-series data was taken on Hela cells at 2 min intervals over the course of ~ 4 hours that contains several cell cycles. Each FPM dataset contains 173 low-resolution intensity images, in which 37 are brightfield (BF), 136 are darkfield (DF). Each intensity image is 2560×2160 pixels in 16-bit grayscale. To generate the data for training, FPM phase reconstructions are used as the ground truth. Each FPM reconstructed phase image contains 12800×10800 pixels, which is 5×5 larger than the raw intensity image. To prepare the dataset for training, we use only the first FPM frame in the time-lapse as the training set. The input to the CNN are BF and DF image patches that are cropped from random locations. Each training input data is formed by stacking the BF and DF image patches. The same preprocessing steps are applied for training, validation, and testing. Once the CNN is trained, which only needs to be performed once using the first FPM frame taken at 0 min, the CNN is then applied to reconstruct high-SBP phase video frames (i.e. the testing step). To perform the reconstruction, similar data preprocessing steps are followed as the training phase. To reconstruct the video, we simply fed each FPM frame to the trained CNN to reconstruct the high-SBP dynamic information from the times-series data. The time for reconstructing each full-FOV, high-SBP image is $\sim 25 \pm 2$ seconds using our cGAN network with the added Fourier loss, which is $\sim 50 \times$ faster than the standard FPM algorithm (which took ~ 20 min for each frame).

3.2 FPM results and discussions

Several illumination patterns along with the corresponding networks were used for training and testing [11]. Several networks are applied to reconstruct the entire time series experiment. The dense block (DB) structure has shown to provide efficient presentation with a small number of parameters in the model [32]. Results from several illumination patterns [11]

with different angular coverage, all reconstructed with DenseNet (UNet with DB) and cGAN structure were performed. Heuristically, we found that the capacity of our CNN is that it can reliably utilize dark-field (DF) data up to 0.4 illumination NA. The reconstructions are further explored using two networks, D-B₉D₂₀-cGAN (D: discriminator network, 9 bright-field (BF), and 20 Dark-field (DF)) and D-B₉D₂₀-F-cGAN (F: Fourier domain loss significantly) [11]. The introduction of the Fourier domain loss significantly boosts the Fourier coverage up to the 0.4 illumination NA (<0.6 NA in the ground truth). We note that using the Fourier domain loss in the training process generally leads to enhancement of the sharpness of the results and the frequency measurement metric (FM) [39]; however, it may trade off image-space metrics, such as MAE, SSIM, and PSNR due to different metric weighting schemes involved. A unique feature of our technique is the ability to reconstruct high-SBP phase videos with training data only from the first time point of a long time-series experiment. To demonstrate the effectiveness of this strategy, we show our CNN predicted temporal frames over the course of over 4 hours. During this process, considerable amount of morphological (hence phase distribution) changes occur due to cell division over several cell cycles. Figure 8(b) shows several frames (reconstructed with D-B₉D₂₀-F-cGAN) of a zoom-in region, where one cell is growing and dividing into multiple cells, and another cell has its membrane rapidly fluctuating. A more quantitative evaluation of the ‘generalization error’ over time is presented in Figure 8(a), in which the MAE metrics of all the networks studied are plotted for every frame in the time series experiment. The error is low at the beginning of the experiment and grows slowly as the time progresses [11].

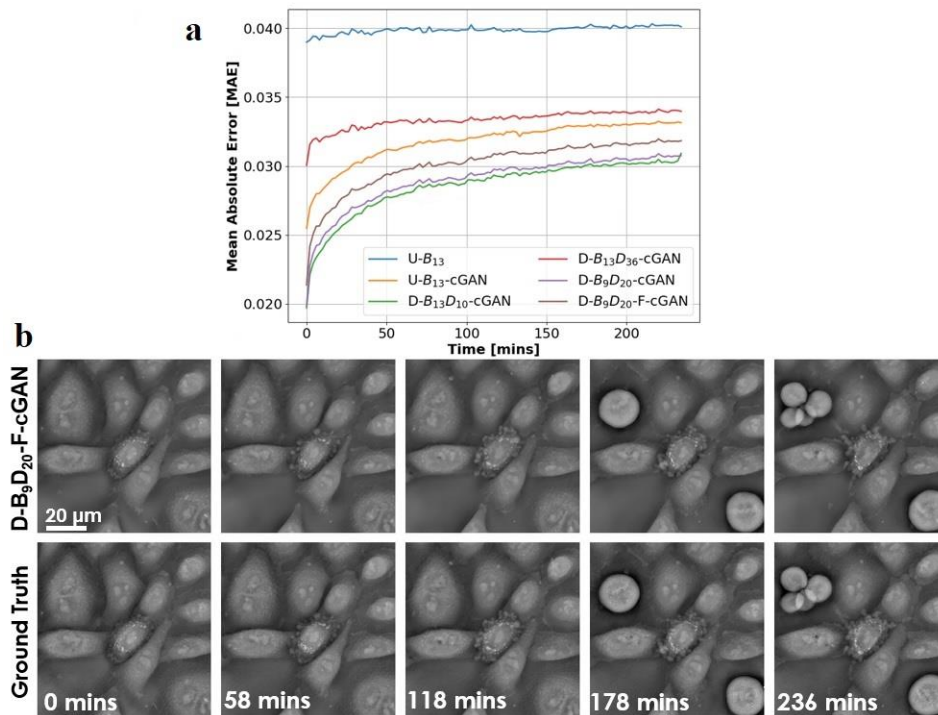


Fig. 8: Reconstructed temporal dynamic information using the proposed CNN. (a) The MAE metric is evaluated for every frame of the time-series experiment on all the CNN models. (b) Several frames of the reconstructed high-SBP phase video from a zoom-in region, where significant morphological changes are observed over the course of 4 hours [11].

4. SAMPLE AND DATA PREPARATION FOR 2D AUTOFOCUSING OF FLUORESCENT MICROSCOPIC IMAGES

In this section of this work, we aim at deblurring fluorescent images and predicting the focal plane of the fluorescent images using various CNN models. For this purpose, we used the image of Phalloidin (actin) dataset from the Broad Bioimage Benchmark Collection [40]. Images were acquired from one 384-well microplate containing U2OS cells stained with Hoechst 33342 markers (to label the nuclei) and with an exposure of 15ms and 1000ms for Hoechst and phalloidin respectively, at 20x magnification, 2x binning, and 2 sites per well. For each site, the optimal focus was found using laser auto-focusing to find the well bottom. The automated microscope was then programmed to collect a z-stack of 32 image sets with 2 μm between slices. Each image is 696 x 520 pixels in 16-bit TIF format, LZW compression [40]. We

have not used the entire dataset, we only used Phalloidin (actin) dataset with focused images as a ground truth and the overfocused plane at 10 μm distance from the focused plane as the input. In this experiment, we use the cGAN model (See Fig. 9) [15,36] with 128x128 image size, batch size of 16, and training with 400 epochs. We used 80% of training, 10% of validation, and 10% of testing. Our image-set consisted of focusing planes at slice numbers [0,4,8,12,16,20,24]. The input cropped images consisted of 128x128 pixels of the original 696x520 pixels. 80% of the dataset was used for training and the other 20% was used for testing and validation. A typical result generated by the modified UNET with cGAN is shown in Fig. 10. Using a focal length $f = 21$ (slice 21 is 10 μm from slice 16), we were able to generate a deblurred image very close to the ground truth focal plane at slice number $f=16$ using modified UNET with cGAN model.

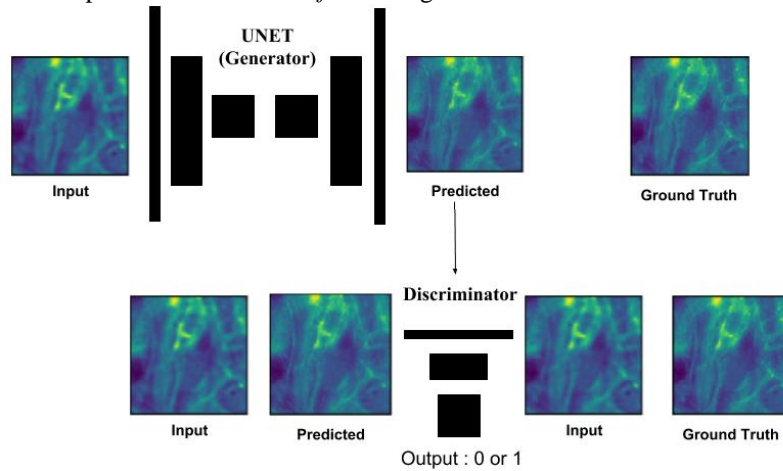


Fig. 9: Modified UNET with conditional generative adversarial network (cGAN) model.

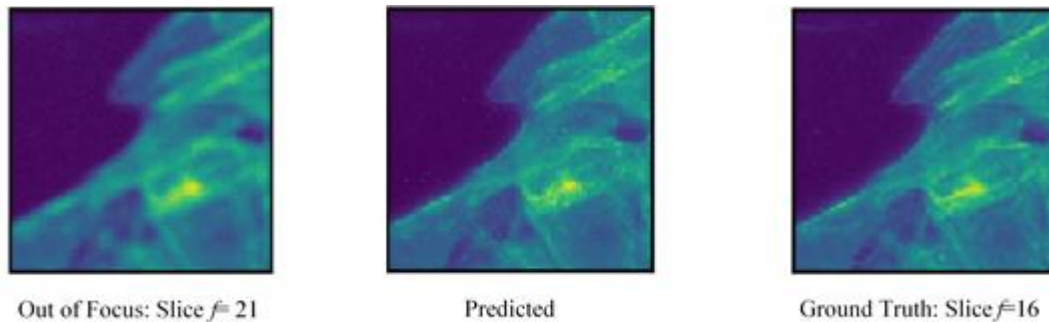


Fig. 10: (left) the out of focus image, (middle) the predicted image through the model, (right) the ground truth.

5. CONCLUSION

In the first part of this work, we demonstrated the feasibility to use 3D-DCNN based technique to reconstruct the 3D index of refraction distribution of synthetic phantoms. The experimental measurements were performed through a custom-built optical setup and computations were calculated using deep learning framework Keras/Tensorflow and accelerated with GPU GeForce NVIDIA Titan Xp. The 3D synthetic phantoms were constructed using simulation and 2D OPL phase images from 360 different angles were displayed on an SLM to mimic the light's phase change due to propagation through the sample. For each 3D phantom (ground truth), the CNN model was trained directly from a stacked series of 360 2D diffracted images that correspond to the phase images displayed on the SLM. The 3D-DCNN algorithm uses SGD as a back propagation method to tune the parameters of the model. As a result of this effort, we showed that the 3D-DCNN model can learn to reconstruct 3D tomographic index of refraction distributions.

In the second part of this work, we have demonstrated a deep learning framework for Fourier ptychography video reconstruction. The proposed CNN architecture fully exploits the unique high-SBP imaging capability of FPM so that it can be trained using a single frame and then be generalized to a full time-series experiment. In addition, the CNN requires reduced number of images for high-resolution phase recovery. The reconstruction of each high-SBP image takes less than

30 seconds. Overall, this technique significantly improves the imaging throughput of the FPM system by reducing both the acquisition and reconstruction time. The central idea of our technique is based on the observation that each FPM image contains a large cell ensemble covering all morphological information throughout the time-series experiment. By the principle of ergodicity, the statistical information learned from these large spatial ensembles in a single frame are shown to be sufficient to predict temporal dynamics with high fidelity. In practice, we showed that our trained CNN can successfully reconstruct a high-SBP phase video of dynamic live cell populations with reduced noise artifacts. Using the conditional generative adversarial network (cGAN) framework and a weighted Fourier loss function, the proposed CNN is able to more effectively learn the high-resolution information encoded in the darkfield data. The technique may find wide applications in *in vitro* live cell imaging and gather large-scale spatial and temporal information in a data and computation efficient manner.

In the third part of this work, we conclude that we have successfully deblurred and predicted the focal plane of fluorescent images using various CNN models. This technique can be applied to any set of images including quantitative phase images.

REFERENCES

- [1] C. Dong, C. C. Loy, K. He, and X. Tang, "Image super-resolution using deep convolutional networks," *IEEE Trans. on Pattern Analysis and Machine Intelligence* **38**(2), 295–307 (2016).
- [2] Yunzhe Li, Yujia Xue, and Lei Tian, "Deep speckle correlation: a deep learning approach toward scalable imaging through scattering media," *Optica* **5**, 1181-1190 (2018).
- [3] L. Xu, J. S. Ren, C. Liu, and J. Jia, "Deep convolutional neural network for image deconvolution," in *Advances in Neural Information Processing Systems* 1790–1798 (2014).
- [4] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," *Communications of the ACM* **60**(6), 84-90 (2012).
- [5] H. C. Burger, C. J. Schuler, and S. Harmeling, "Image denoising: Can plain neural networks compete with BM3D?" in *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*. IEEE, 2392–2399 (2012).
- [6] Xue, Yujia, Shiyi Cheng, Yunzhe Li, and Lei Tian, "Illumination coding meets uncertainty learning: toward reliable AI-augmented phase imaging," *arXiv preprint arXiv:1901.02038* (2019).
- [7] C. Dong, Y. Deng, C. Change Loy, and X. Tang, "Compression artifacts reduction by a deep convolutional network," in *Proc. of the IEEE Intern. Conf. on Computer Vision* 576–584 (2015).
- [8] K. Kulkarni, S. Lohit, P. Turaga, R. Kerviche, and A. Ashok, "Reconnet: Non-iterative reconstruction of images from compressively sensed measurements," in *Proc. of the IEEE Conf. on Computer Vision and Pattern Recognition* 449–458 (2016).
- [9] A. Sinha, J. Lee, S. Li, and G. Barbastathis, "Lensless computational imaging through deep learning," *Optica* **4**(9), 1117-1125 (2017).
- [10] K. H. Jin, M. T. McCann, E. Froustey, and M. Unser, "Deep convolutional neural network for inverse problems in imaging," *IEEE Trans. on Image Processing* **26.9**, 4509-4522 (2017).
- [11] T. Nguyen, Y. Xue, Y. Li, L. Tian, and G. Nehmetallah, "A deep learning approach for Fourier ptychography microscopy," *Optics Express*, **26**(20), 26470-26484 (2018).
- [12] S. Yadav, C. Jain, A. Chugh, "Evaluation of Image Deblurring Techniques," *International Journal of Computer Applications* (0975 – 8887) Volume 139 – No.12, April 2016.
- [13] S. J. Yang, M. Berndl, D. Michael Ando, M. Barch, A. Narayanaswamy, E. Christiansen, S. Hoyer, C. Roat, J. Hung, C. T. Rueden, A. Shankar, S. Finkbeiner and P. Nelson, "Assessing microscope image focus quality with deep learning," *BMC Bioinformatics* (2018) 19:77 (<https://doi.org/10.1186/s12859-018-2087-4>).
- [14] 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**, 266-277 (2009).
- [15] K. Kim, H. Yoon, M. Diez-Silva, M. Dao, R. R. Dasari, and Y. Park, "High resolution three-dimensional imaging of red blood cells parasitized by plasmodium falciparum and in situ hemozoin crystals using optical diffraction tomography," *J. Biomed. Opt.* **19**, 011005 (2014).
- [16] T. Nguyen, V. Bui, V. Lam, C. B. Raub, L. Chang, and G. Nehmetallah, "Automatic phase aberration compensation for digital holographic microscopy based on deep learning background detection," *Opt. Express* **25**, 15043-15057 (2017).
- [17] Laura Waller, Lei Tian, and George Barbastathis, "Transport of Intensity phase-amplitude imaging with higher order intensity derivatives," *Opt. Express* **18**, 12552-12561 (2010).
- [18] Ruilong Ling, Waleed Tahir, Hsing-Ying Lin, Hakho Lee, and Lei Tian, "High-throughput intensity diffraction tomography with a computational microscope," *Biomed. Opt. Express* **9**, 2130-2141 (2018)
- [19] T. F. Chan and C.-K. Wong, "Total variation blind deconvolution," *IEEE Trans. on Image Proc.* **7**(3), 370–375 (1998).
- [20] S. G. Chang, B. Yu, and M. Vetterli, "Adaptive wavelet thresholding for image denoising and compression," *IEEE trans. on image processing* **9**(9), 1532–1546 (2000).

- [21] L. Williams, G. Nehmetallah, and P. P. Banerjee, "Digital tomographic compressive holographic reconstruction of three-dimensional objects in transmissive and reflective geometries," *Appl. Opt.* **52**, 1702-1710 (2013).
- [22] B. Preece, T. Du Bosq, N. Namazi, G. Nehmetallah, and K. Kelly "A Noise Model for the Design of a Compressive Sensing Imaging System," *Proc. SPIE. 9452, Infrared Imaging Systems: Design, Analysis, Modeling, and Testing XXVI*, 94520L. (May 12, 2015).
- [23] A. Beck and M. Teboulle, "A fast iterative shrinkage-thresholding algorithm for linear inverse problems," *SIAM J. on imaging sciences* **2**(1) 183–202 (2009).
- [24] R. Horisaki, R. Takagi, and J. Tanida, "Learning-based imaging through scattering media," *Opt. Express* **24**(13), 13738-13743 (2016).
- [25] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," *Intern. Conf. on Medical Image Computing & Computer-Assisted Intervention*, Springer, Cham, (2015).
- [26] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *Nature* **521**.7553, 436-444 (2015).
- [27] L. Tian, Z. L., L.-H. Yeh, M. Chen, J. Zhong, and Laura Waller, "Computational illumination for high-speed in vitro Fourier ptychographic microscopy," *Optica* **2**(10), 904-911 (2015).
- [28] Lei Tian, Xiao Li, Kannan Ramchandran, and Laura Waller, "Multiplexed coded illumination for Fourier Ptychography with an LED array microscope," *Biomed. Opt. Express* **5**, 2376-2389 (2014)
- [29] S. Ioffe, C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," *ArXiv 1502.03167 Learning 2015*. [<https://arxiv.org/abs/1502.03167>].
- [30] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, R. Salakhutdinov, "Dropout: A simple way to prevent neural networks from overfitting," *The Journal of Machine Learning Research* **15**(1),1929-1958 (2014).
- [31] K. He, X. Zhang, S. Ren, & J. Sun, "Deep residual learning for image recognition," *CVPR* 770-778 (2016).
- [32] G. Huang, Z. Liu, KQ. Weinberger, & L. Van der Maaten, "Densely connected convolutional networks," *CVPR* **1**(2), 3 (2017).
- [33] F. Agostinelli, M. Hoffman, P. Sadowski, P. Baldi, "Learning activation functions to improve deep neural networks," *ArXiv 1412.6830 Neu & Evol Comp 2014*. [<https://arxiv.org/abs/1412.6830>].
- [34] V. Dumoulin and F. Visin, "A guide to convolution arithmetic for deep learning," *ArXiv 1603.07285 Machine Learning 2016*. [<https://arxiv.org/abs/1603.07285>].
- [35] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, Y. Bengio, "Generative adversarial nets," *Advances in neural information processing systems* 2672-2680 (2014).
- [36] P. Isola, JY. Zhu, T. Zhou, AA. Efros. "Image-to-image translation with conditional adversarial networks," *ArXiv 1611.07004 Comp Vis & Patt Recog 2017*. [<https://arxiv.org/abs/1611.07004>].
- [37] T. Salimans I. Goodfellow, W. Zaremba, V. Cheung, A. Radford, X. Chen, "Improved techniques for training gans," *Advances in Neural Information Processing Systems* 2234-2242 (2016).
- [38] Z. Wang, AC. Bovik, HR. Sheikh, EP. Simoncelli, "Image quality assessment: from error visibility to structural similarity," *IEEE Trans on Img Process* **13**(4), 600-612 (2004).
- [39] K. De and V. Masilamani, "Image sharpness measure for blurred images in frequency domain," *Procedia Engineering* **64**, 149-158 (2013).
- [40] https://data.broadinstitute.org/bbbc/image_sets.html.