Holographic 3D particle imaging with model-based deep network
Ni Chen, Congli Wang, and Wolfgang Heidrich, Senior Member, IEEE

Abstract—Gabor holography is an amazingly simple and effective approach for three-dimensional (3D) imaging. However, it suffers from a DC term, twin-image entanglement, and defocus noise. The conventional approach for solving this problem is either using an off-axis setup, or compressive holography. The former sacrifices simplicity, and the latter is computationally demanding and time-consuming. To cope with this problem, we propose a model-based holographic network (MB-HoloNet) for three-dimensional particle imaging. The free-space point spread function (PSF), which is essential for hologram reconstruction, is used as a prior in the MB-HoloNet. All parameters are learned in an end-to-end fashion. The physical prior makes the network efficient and stable for both localization and 3D particle size reconstructions.

Index Terms—Holography, Neural networks, Inverse problems

I. INTRODUCTION

Holography is a powerful tool for 3D imaging and display due to its wavefront encoding ability. Since it has been proposed in 1948 [1], a large amount of work has emerged for solving the DC and twin-image problems in the hologram reconstruction. These problems can be improved by changing the optical system, such as off-axis holography [2], phase-shifting holography [3], optical scanning holography [4], and others. However, the simplicity and high space-bandwidth product of Gabor holography remains attractive and competitive. Apart from the DC term and twin-image, the main issue in 3D digital holographic imaging is the defocus noise. Compressive sensing approaches [5]–[7] are efficient in restoring 3D images from holograms. However, these methods are computationally intensive and usually require fine-tuned parameters, such as the regularization and relaxation ones [5], [8].

Recently, deep learning has emerged as a comprehensive tool in computational imaging [9], in areas of optical tomography [10], ghost imaging [11], digital holography [12], imaging through scattering media [13], and phase imaging [14]. However, compared to other fields, deep learning has been under-utilized in digital holography [15], especially for 3D holographic imaging. Most previous works for optical imaging rely on an end-to-end neural network, which requires large amounts of data for training. 3D holographic imaging is challenging for data-driven networks due to the limitations of acquiring large quantities of training data. There are mainly two popular ways for obtaining the training data-sets: (1) Display the target (labels) on spatial light modulators (SLM) or digital micromirror devices (DMD) for capturing the images (data) of the labels that passing through the optical system [11]; (2) Capture a few high-resolution optical images, calculate the labels with conventional imaging methods. More data and labels are obtained by cropping, rotating of the existing ones [16]. Both approaches are not practical for 3D images. Therefore, for 3D holographic imaging, only a few results have been reported [17]–[19]. The state-of-the-art technique can reconstruct particle locations in a volume [17], [18]. The authors obtained training data with a previously well-developed technique, for example, scanning microscopy, and a modified U-Net was applied for particle localization and suffering from a severe, time-consuming pre- and post-processing.

Besides holography, many optical imaging techniques rely on specific physical models. Most of the previous works rely on an end-to-end neural network while neglecting the knowledge in established physics models [20]. To integrate physical knowledge as network priors, several model-based schemes have been proposed to incorporate priors with the forward model in the learning process to ensure data consistency and prior suitability, known as the unrolled networks [21]–[23]. Such combined approaches have been successfully applied to computed tomography (CT) [24] and magnetic resonance imaging (MRI) reconstruction [25]. In particular, Kristina et al. [25] have unrolled a traditional model-based optimization algorithm based on the alternating direction method of multipliers (ADMM) [26] with a convolution neural network (CNN) for mask-based lens-less imaging.

Inspired by previous studies, we integrate the physical principles of holography with deep learning-based approaches for 3D holographic particle imaging. We consider 3D holographic imaging as a compressive sensing problem and adopting an iterative shrinkage-thresholding algorithm to perform 3D hologram reconstruction, named model-based holonet (MB-HoloNet). In particular, MB-HoloNet takes a single Gabor hologram along with the free-space PSF and outputs the 3D particle volume. It is composed of a fixed number of stages, each of which strictly corresponds to an iteration in the shrinkage-thresholding algorithm. Features of the objects and all the parameters involved are learned in an end-to-end fashion. The PSF fed into the network makes it more accurate and numerically stable. MB-HoloNet can work when the test

Compiled on September 17, 2020. This work was supported by baseline funding of the King Abdullah University of Science and Technology. N. Chen, C. Wang, and W. Heidrich are with the Visual Computing Center, King Abdullah University of Science and Technology, Thuwal 23955-8900, Kingdom of Saudi Arabia (e-mails: {ni.chen, congli.wang, wolfang.heidrich}@kaust.edu.sa).
holograms are captured under different environments, as in the training data. We further show that the network trained with synthesized data works for experimental captured hologram reconstruction.

II. PHYSICALLY BASED MB-HOLONET

The proposed method consists of two steps: (i) Linearization of Gabor hologram imaging; (ii) Solving the linear problem augmented with data-driven priors in MB-HoLoNet.

A. Linearization of Gabor holographic imaging

In a Gabor holographic imaging system, as in Fig. 1, a 3D object of amplitude transfer function \( o(r, z) \) is positioned at the origin of the coordinate system, with \( r = (x, y) \). A plane reference wave field \( \exp(jkz) \) of wave number \( k = 2\pi/\lambda \) illuminates the object, producing the object field \( o(r, z) \exp(jkz) \), with position \( z \) encoded in the phase delay.

Both the reference and the object field propagates along the \( z \)-axis, arriving at the sensor plane at \( z_0 \). According to scalar diffraction theory [27], with \( h(r, z) \) denoting free-space propagation of point \( (r, z) \) and \( \otimes \) for convolution, the impinging 2D fields are \( u_r(r) \approx 1 \) and

\[
u_o(r) = \int_{z_1}^{z_2} \left( o(r, z) \exp(jkz) \right) \otimes h(r_h - r, z_0 - z) \, dz.
\]

The two fields \( u_r \) and \( u_o \) interfere with each other and produce an interferometric intensity pattern \( I_h = |u_r + u_o|^2 \), known as the Gabor hologram, and is captured by the image sensor. Expanding \( I_h \) with \( \Re\{\cdot\} \) denoting the real part and \( \{\cdot\}^* \) denoting the complex conjugate:

\[
I_h(r) = 2\Re\{u_o\pi_r\} + |u_r|^2 + |u_o|^2, \tag{2}
\]

where \( |u_r|^2 \) is known from calibration, and \( |u_o|^2 \) is treated as a non-linear model error. With these, the hologram reconstruction is rephrased as a linear inverse problem:

\[
I_h \approx 2\Re\{Po\}, \tag{3}
\]

where bold fonts denote vectors in linear algebra, discretized from their continuous versions, and \( P \) is a linear operator that maps 3D complex fields to 2D complex fields.

B. Model-based HoLoNet

Due to the large solution space of \( o \), it is ill-posed for directly solving Eq. (4). By assuming that \( o \) is real valued with physically realistic sparsity properties [28], [29] in linear basis \( \Psi \), regularized least squares fitting Eq. (3) yields:

\[
\min_o \frac{1}{2} \|\Phi o - I_h\|_2^2 + \gamma \|\Psi o\|_1, \tag{4}
\]

where \( \Phi = 2\Re\{P\} \), \( \| \cdot \|_2^2 \) is the \( \ell_2 \)-norm squared, expressing fitting fidelity between measurement and the physical model, \( \| \cdot \|_1 \) is the \( \ell_1 \)-norm, and \( \gamma \) is a trade-off parameter. Though coupled, the two terms in Eq. (4) can be de-coupled and alternatively solved via the half-quadratic splitting method [30]. By introducing an additional variable \( v = o \), the two terms are separated:

\[
\min_{o,v} \frac{1}{2} \|\Phi o - I_h\|_2^2 + \gamma \|\Psi v\|_1 + \mu \|o - v\|_2^2; \tag{5}
\]

with \( \mu \to \infty \) being an algorithmic parameter. At the \( n \)-th iteration, solving \( o \) and \( v \) yields the following two update steps in sequence:

\[
o^{n+1} = \arg \min_o \|\Phi o - I_h\|_2^2 + \mu \|o - v^n\|_2^2, \tag{6}
\]

\[
v^{n+1} = \arg \min_v \frac{1}{2} \|v - o^n\|_2^2 + \frac{\gamma}{\mu} \|\Psi v\|_1. \tag{7}
\]

Equation (6) and Eq. (7) can be solved efficiently using proximal algorithms [26], [31]. However, these methods usually require hundreds of iterations to obtain a satisfactory result, which inevitably gives rise to high computational cost and is thus restricting for fast applications. In addition, all parameters must be pre-defined, and it can be quite challenging to tune them priori. Also, the selection for sparsity basis \( \Psi \) is usually hand-crafted, and the physical parameters of the setup need to be calibrated or chosen very carefully. To resolve these difficulties, we solve Eq. (6) and Eq. (7) using unrolled neuronal networks [32], seeking for a good balance between speed and accuracy. The idea is to turn the fixed, human-tuned algorithmic parameters, as well as the physical parameters of the imaging setup into differentiable system parameters that are automatically tuned via back-propagation through supervised learning.

We now discuss specific solutions (termed network modules) to these two steps, as illustrated in Fig. 2. To increase network flexibility, we allow both \( \gamma \) and \( \mu \) to vary across iterations, with subscript \( n \) denotes for the \( n \)-th iteration, i.e. the \( n \)-th stage in the network.

\( o^n \)-module corresponds to Eq. (6), which produces an immediate reconstruction \( o^{n+1} \) from the previously calculated \( v^n \). The solution to Eq. (6) is simply a least squares estimate:

\[
o^n = (\Phi^T \Phi + \mu_n I)^{-1} (\Phi^T v^n + \mu_n \Phi^T I_h), \tag{8}
\]

with \( I \) being the identity matrix. Fortunately, under the assumption of \( u_r \) being a plane-wave illumination, the linear operator \( P \) can be rephrased as a convolution followed by sum. We can further break down the above separately for each 2D plane at specific \( z \) as:

\[
o^n_z = [o^n_{z_1}, \ldots, o^n_{z_2}]^T, \tag{9}
\]

\[
o^n_z = F^{-1} \left( \frac{F(p_z)\Phi I_h + \mu_n F(v^n_z)}{\mu_n \|F(p_z)\|^2 + 1} \right), \tag{10}
\]

where \( p_z \) is the propagation kernel at depth \( z \), and \( F/F^{-1} \) denote for the forward/inverse Fourier transforms. We initialized \( v^0 \) with classical back-propagated solution, i.e. \( v^0 = \Phi^T I_h \).
\[ v^n \text{-module} \text{ aims to compute } v^n \text{ from } o^n \text{ according to Eq. (7), which is a special case of the so-called proximal mapping } \text{prox}_{\tau, \phi} (\cdot) \text{ where } \phi (v) = \| \Psi v \|_1. \text{ We now consider a general nonlinear transform } G (\cdot) \text{ that sparsifies the object. Replacing } \Psi \text{ with } G (\cdot), \text{ we can obtain a sparsity-inducing regulari-}
}

\[ G(v) = \text{ReLU} \left( \text{Conv} \left( \text{ReLU}(\text{Conv}(v)) \right) + v \right). \tag{11} \]

To obtain \( v \) from \( G(v) \), we introduce the inverse transformation function \( G^{-1}(\cdot) \), which can be written in the same structure as \( G(v) \). Incorporating \( G^{-1}(\cdot) \) and \( G^{-1}(\cdot) \) into Eq. (7), we obtain the optimization \( v^n = G^{-1} \left( \text{SoftThreshold} \left( G(o^n), \tau \right) \right) \), as in [23].

\section*{C. Network structure and loss design}

As stated previously, the parameters to be learned are \( \Theta = \{ \gamma_n, \mu_n, \tau_n, G_n, G_n^{-1} \}_{n=1}^{N_s} \). The whole framework of the MB-HoloNet consists of \( N_s \) stages, as shown in Fig. 2, with the input being a single 2D hologram image, and output being the desired 3D particle volume to be restored. As mentioned, \( o^n \) denotes the restored 3D object of the \( n \text{th} \) stage, and \( v^n \) denotes the auxiliary variable in the \( n \text{th} \) stage. In each stage, the \( o^n \) is updated according to Eq. (6), the estimated immediate reconstruction \( o^n \) is passed to the CNN for the next stage to compute \( v^n \) according to Eq. (7). The filter number of the CNN equals to the number of depth slices of the 3D object. A sigmoid activation layer is implemented before the output to map the object transfer function to the absorption function.

\textbf{Loss function:} \( o \) is the corresponding true value of the the estimated network output \( v^N \). MB-HoloNet seeks to optimize the following total loss function

\[ \mathcal{L} = \mathcal{L}_{\text{model}} + \tau \mathcal{L}_{\text{residual}} \tag{12} \]

given by the squared \( \ell_2 \)-norm of the 3D data and a penalty of the residual block, to guarantee both data and prior fidelity:

\[ \mathcal{L}_{\text{model}} = \frac{1}{N_b \times N_s} \sum_{n=1}^{N_s} \left\| v^{N_n} - o \right\|_2^2, \tag{13} \]

\[ \mathcal{L}_{\text{residual}} = \frac{1}{N_b \times N_s} \sum_{n=1}^{N_s} \sum_{m=1}^{N_s} \| G^{-1}_m (G(v)^n) - v^n \|_2^2, \tag{14} \]

where \( N_b \) is the block number, \( N_s \) is the stage number, and \( N \) is the object size.

To tackle the issues of limited computational memory and stagnation in local minima during optimization, every time only a small batch of the entire training data is fed into the network, instead of the entire set of holograms. The network is trained using the Adam optimizer [35], which is a form of gradient descent, where the initial learning rate is set empirically, decaying according to the loss descent rate dynamically as the training progresses. We implement the MB-HoloNet using Tensorflow 1.15 and Keras 2.2.5. All experiments were performed on a workstation with an Intel Core i7-6820 CPU and an NVIDIA GTX1080 GPU.

\section*{III. Verification}

\subsection*{A. Evaluation metrics}

We make use of three quantitative evaluation metrics for the assessment of the network performance. These are pairwise correlation coefficient (PCC), structural similarity index for measuring (SSIM) and mean absolute error (MAE). These are defined as:

\[ \text{SSIM}(x, y) = \frac{(2\mu_x \mu_y + C_1)(2\sigma_{xy} + C_2)}{\mu_x^2 + \mu_y^2 + C_1 \sigma_x^2 + C_2 \sigma_y^2} \]

\[ \text{MAE}(x, y) = \frac{1}{n} \sum_{i=1}^{n} |x_i - y_i| \]

where \( \mu_x, \mu_y \) are the means, \( \sigma_{xy} \) is the covariance, \( \sigma_x^2, \sigma_y^2 \) are variances, \( C_1 \) and \( C_2 \) are constants to avoid division by zero, and \( n \) is the number of pixels.
The PCC and SSIM are defined as follows:

\[
PCC(x, y) = \frac{\sum_{n=1}^{N} (x_n - \mu_x)(y_n - \mu_y)}{\sqrt{\sum_{n=1}^{N} (x_n - \mu_x)^2 \sum_{n=1}^{N} (y_n - \mu_y)^2}},
\]

\[
SSIM(x, y) = \frac{2\mu_x\mu_y + c_1(2\sigma_{xy} + c_2)\sqrt{c_2\sigma_y^2 + c_2}},
\]

\[
MAE(x, y) = \frac{1}{N} \sum_{n=1}^{N} |x_n - y_n|,
\]

where \(\mu_x\) and \(\mu_y\) are means of \(x\) and \(y\), \(\sigma_x^2\) and \(\sigma_y^2\) are the corresponding variances, and \(\sigma_{xy}\) is the covariance of \(x\) and \(y\). \(c_1 = (k_1L)^2\), \(c_2 = (k_2L)^2\), where \(k_1 = 0.01\), \(k_2 = 0.03\), \(L\) is the dynamic range of the pixel-values.

The MAE is an average measure of the absolute difference between two variables and its best possible score is 0.

### B. Numerical verification of particle localization

Particle localization is a special case of the more general particle volume reconstruction when the particle and voxel are of the same size. In this section, we verify the feasibility of the proposed MB-HoloNet and its robustness with single-pixel size particles.

![Fig. 3: Training and validation loss decrease along the training process. Only first 500 iterations are shown here.](Image)

In the first simulation, one thousand Gabor holograms of randomly distributed particle volumes were synthesized. The lateral and axial resolutions of the particle volumes are 20 μm and 100 μm, respectively, and the sensor pixel pitch is the same as the volume voxel size. The particle density is \(5 \times 10^{-3}\) particle per voxel (ppv). For each hologram, we have added 50 dB random Gaussian noise. The generated holograms, along with the 3D particle volumes, were fed into the proposed MB-HoloNet as the training data and label. We split 80% as training set and 20% for validation. Extra 100 holograms were synthesized with the same specifications for testing. Figure 3 presents one example of the training and validation loss. As seen, the MB-HoloNet converges gradually as the loss decreases along with the training. This agrees with our expectation that the network is continuously updating its parameters and learning representative features of the holograms.

Next, we verified the robustness of the MB-HoloNet by showing that the MB-HoloNet works for systems with varying specifications, and the trained network works for holograms captured in different scenarios as the training data. To verify that MB-HoloNet works for varying systems, objects with different dimensions (32 × 32 × 7, 64 × 64 × 32, and 128 × 128 × 30) were used to synthesize the holographic imaging system. In each configuration, the wavelength is fixed to \(\lambda = 660\) nm. Specifications of the training parameters and the test results are presented in Section III-A. Figure 5 plots the accuracy metric concerning varying noise levels, and Fig. 6 shows randomly selected scattering plot of the predictions and ground-truth. It shows that when the noise SNR is much lower than that in the training data (50 dB), we can still reach a reasonable image quality. This indicates that the network is numerically stable to noise.

![Fig. 4: Hologram localization reconstruction with three different dimensions, synthesized with various specifications. The red, blue circles are the predictions and ground-truth, and the green dots in red and blue circles are the unpaired ground-truth and the false predictions.](Image)

![Fig. 5: Test accuracy with respect to varying noise levels.](Image)
### Table I: Training details of different datasets.

<table>
<thead>
<tr>
<th>Cases</th>
<th>(N_x \times N_y \times N_z)</th>
<th>Training time</th>
<th>Test time</th>
<th>Test Score</th>
</tr>
</thead>
<tbody>
<tr>
<td>Train</td>
<td>(32 \times 32 \times 7)</td>
<td>20 min</td>
<td>12 ms</td>
<td>99.91</td>
</tr>
<tr>
<td></td>
<td>(64 \times 64 \times 32)</td>
<td>3 h30 min</td>
<td>17 ms</td>
<td>99.67</td>
</tr>
<tr>
<td></td>
<td>(128 \times 128 \times 30)</td>
<td>5 h31 min</td>
<td>41 ms</td>
<td>99.42</td>
</tr>
<tr>
<td>Results</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Data sets</td>
<td>1000</td>
<td>1000</td>
<td>1000</td>
</tr>
<tr>
<td></td>
<td>N_s</td>
<td>5</td>
<td>5</td>
<td>5</td>
</tr>
<tr>
<td></td>
<td>Batch size</td>
<td>32</td>
<td>32</td>
<td>32</td>
</tr>
<tr>
<td></td>
<td>Learning rate</td>
<td>(2 \times 10^{-3})</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td>(\tau)</td>
<td>(1 \times 10^{-4})</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td>Epochs</td>
<td>1000</td>
<td>1500</td>
<td>1000</td>
</tr>
</tbody>
</table>

![Fig. 6: MB-HoloNet particle reconstruction vs. ground-truth under varying noise.](image)

Fig. 6: MB-HoloNet particle reconstruction vs. ground-truth under varying noise. The red, blue circles are the predictions and ground-truth, and the green dots in red and blue circles are the unpaired ground-truth and the false predictions.

![Fig. 7: Test accuracy with respect to varying ppv.](image)

Fig. 7: Test accuracy with respect to varying ppv.

Figure 7 plots the accuracy metrics with respect to varying ppv and Fig. 8 shows several selected scatter plots of the predictions and ground-truth. They show that when the ppv is lower than in the training data, the trained network can always reconstruct the particles. However, the image quality degrades while the ppv increasing in the testing data, while the SSIM can still maintain a high degree even if the particle density of the test data is twice that of the training data.

One more simulation was performed for comparing the particle localization with the state-of-the-art work [17]. In the reference paper, volumes with particle per pixel (ppp) of \(1.9 \times 10^{-4} \sim 6.1 \times 10^{-2}\) were tested. This corresponds to a ppv of \(1.5 \times 10^{-6} \sim 4.8 \times 10^{-4}\). We have synthesized holograms for particles with the same ppv and dimension of \(64 \times 64 \times 64\), trained the network with 1500 data-sets for 2000 epochs. Figure 9 presents the comparison between the ground-truths and the network predictions, while Fig. 9(a) and (b) are with ppp of \(1.6 \times 10^{-2}\) and \(6.1 \times 10^{-2}\) respectively, the location extraction rates are 87.5% and 82.4%, which are lower than the 95% in the reference paper.

### C. Numerical verification of particle size reconstruction

We verify the particle shape reconstruction by setting the particle size larger than a single voxel. In this simulation, the particle radius is set to be 50 \(\mu\)m, and the pixel pitch of the camera is 7 \(\mu\)m. Therefore, one particle should cover about seven image pixels. The training process is the same as in previous sections. We have tested the trained network after 1500 epochs. Two randomly selected ground-truth and
The predictions are shown in Fig. 10. From the error plots, we can see that the particle shape and size have been well reconstructed.

**IV. EXPERIMENTAL VERIFICATION**

**A. Particle localization**

In the first experiment, particles with a diameter of 50 μm were randomly seeded in water. The particles were located at a distance ranging from 12 mm to 60 mm to the image sensor. A laser with a wavelength of 660 nm illuminated the particles and form the hologram captured by the CCD, as shown in Fig. 11(a). The pixel pitch and resolution of the CCD are 3.45 μm and 1024 × 1024 respectively. The captured hologram is subtracted by the background and scaled to a size of 128 × 128 pixels. Since the particles are very sparse, the depth interval of the view interest was set to 1 mm, which corresponding to a volume of 128 × 128 × 49 voxels. We generated one thousand and five hundred holograms with the same specifications as in the processed hologram and trained the network with the synthetic holograms. After 1000 epochs training (took 12 hours), we test the MB-HoloNet with the experimental captured hologram. Figure 11 shows the results, while Fig. 11(a) shows four processed holograms and Fig. 11(b) shows the reconstructed particle locations.

**B. Particle size reconstruction**

In the second experiment, we show that the proposed MB-HoloNet can reconstruct particle size and locations simultaneously. The particles with diameters ranging from 45 μm to 53 μm were adhered on a microscopy slide. The slide was put at a distance of 19.7 mm from the CCD camera. The pixel pitch and resolution of the CCD are 3.45 μm and 1024 × 1024 respectively. The holograms were cropped to 256 × 256 pixels.
and then resized to 128 × 128 pixels. Thus the diameter of the reconstructed particles from the processed holograms should be 7∼8 pixels. For the training data, volumes span a depth of 14 mm to 25 mm were chosen to synthesize the holograms. All other parameters matched the processed hologram. A total of 1500 data-sets were used to train the network for 1000 epochs, which took 4 h 32 min.

Figure 13 shows the results. Fig. 13(a) is the original hologram, Fig. 13(b) shows several sub holograms cropped from the original one, and Fig. 13(c) is the corresponding network predictions. The bottom row of Fig. 13(c) is the corresponding slice views of the above row. From the above row, we can see that the particles were reconstructed at the correct locations (z = 19.7 mm). Compare the bottom row with the corresponding holograms in Fig. 13(b), we see that most of the particles were reconstructed, the red rectangles shows the error predicated particles, while the error amount is at a low rate. The lateral pixels of each particle in Fig. 13(c) is seven, which matches the expectation.

(a) Processed original hologram.

(b) Cropped and resized sub-holograms.

(c) MB-HoloNet predictions.

Fig. 13: Captured hologram (a), processed holograms (b), hologram reconstruction with MB-HoloNet (c).

V. DISCUSSION AND CONCLUSIONS

We have presented MB-HoloNet for 3D particle imaging with Gabor holography. The MB-HoloNet takes a single 2D hologram as input and outputs the corresponding 3D particle volume. Compared to state-of-the-art technique [17], our proposed MB-HoloNet involves the prior of the underlying imaging model in network design and can reconstruct particle size and perform particle localization at the same time, even though the location extraction rate is slightly lower than the reference work. The physical prior employed makes it possible to train the network with a relatively small amount of training data (1000 ∼ 1500 in all of the presented cases in this paper). Besides, the single 2D image input and 3D volume output is simple and does not require time-consuming pre- and post-processing. Most importantly, the model trained with synthesized data can process experimentally captured holograms; this is superior to previously published works.

While the proposed MB-HoloNet has superior performance compared to the state-of-the-art 3D hologram reconstruction methods, there remains room for further improvements: (1) For the MB-HoloNet, training labels are 3D, which is GPU memory consuming and limits the total volume size that can be processed. The sparsity of the particles may be considered to alleviate the memory consumption, promote training speed, and more massive volume processing. (2) Due to the available small volumes mentioned in (1), the required input holograms have to be small. For this purpose, captured holograms (usually of a larger size) have to be cropped or resized. The cropped holograms cover a small imaging region of a few particles, also the resizing decrease hologram details, and usually leads to induced resolution. It is our next work to find an alternative 3D representation to overcome these limitations.

REFERENCES


