Optics EXPRESS

DeepCGH: 3D computer-generated holography using deep learning

M. HOSSEIN EYBPOSH,[1,2] NICHOLAS W. CAIRA,[1] MATHEW ATISA,[3] PRANEETH CHAKRAVARTHULA,[3] AND NICOLAS C. PÉGARD[1,2,4,*]

[1] *Department of Applied Physical Sciences, University of North Carolina at Chapel Hill, NC 27599, USA*
[2] *Department of Biomedical Engineering, University of North Carolina at Chapel Hill, NC 27599, USA*
[3] *Department of Computer Science, University of North Carolina at Chapel Hill, NC 27599, USA*
[4] *Neuroscience Center, University of North Carolina at Chapel Hill, NC 27599, USA*
[*] *pegard@unc.edu*

Abstract: The goal of computer-generated holography (CGH) is to synthesize custom illumination patterns by modulating a coherent light beam. CGH algorithms typically rely on iterative optimization with a built-in trade-off between computation speed and hologram accuracy that limits performance in advanced applications such as optogenetic photostimulation. We introduce a non-iterative algorithm, DeepCGH, that relies on a convolutional neural network with unsupervised learning to compute accurate holograms with fixed computational complexity. Simulations show that our method generates holograms orders of magnitude faster and with up to 41% greater accuracy than alternate CGH techniques. Experiments in a holographic multiphoton microscope show that DeepCGH substantially enhances two-photon absorption and improves performance in photostimulation tasks without requiring additional laser power.

1. Introduction

Computer generated holography (CGH) aims to synthesize custom light intensity distributions by modulating a coherent wavefront, typically by digitally encoding its phase with a spatial light modulator (SLM). CGH is the preferred method to create custom volumetric illumination in a broad range of applications including neural photostimulation [1–3], optical trapping [4,5], and 3D displays [6,7].

CGH algorithms identify the best possible wave modulation by solving a multidimensional, nonlinear, and non-convex inverse problem that is generally ill-posed. Holograms must be a free space solution of the wave propagation equation that conserve energy, and thus many target intensities are physically infeasible and cannot be rendered exactly. This issue is most notable in 3D CGH when the target intensities specified across several successive depth planes are mutually incompatible. Additional factors that prevent feasibility include the numerical aperture of the optical system, which sets the maximal allowable resolution, as well as hardware limitations such as finite SLM resolution. In practice, CGH solutions are always approximate and numerical methods are required to identify a feasible hologram that best matches the desired illumination pattern.

Aside from simple holographic superposition techniques [8], existing methods for CGH rely on iterative exploration. The most common approach is the Gercheberg-Saxton (GS) algorithm [9], which digitally propagates a complex wave back and forth between the image plane, where the intensity distribution is rendered, and the SLM plane, where the wavefront is modulated, while enforcing amplitude constraints at each step. This algorithm is simple and straightforward to implement, but yields sub-optimal solutions. More recently, advanced algorithms have been developed that compute holograms by solving an optimization problem with an explicit loss function. This includes non-convex optimization with a gradient descent [10], and methods based

#399624 https://doi.org/10.1364/OE.399624
Journal © 2020 Received 10 Jun 2020; revised 5 Aug 2020; accepted 17 Aug 2020; published 25 Aug 2020

on Wirtinger derivatives that redefine CGH as a quadratic problem which can be minimized with first-order optimization [11,12]. Both approaches yield significantly better solutions than GS, but at the cost of increased computation time.

Existing algorithms are unfit to meet a growing demand for synthesizing high resolution holograms in a short time window, for instance, in holographic optogenetic microscopes [1,13] where specific neuron ensembles must be stimulated in direct response to an online observation of animal behavior or neural activity [14]. Current strategies to accelerate hologram computation include optimizing hardware implementation [15] and compressed sensing approaches that reduce computation to smaller regions of interest [2] when the target intensity patterns are spatially sparse. All these approaches rely on time-consuming algorithms that must perform several iterations to identify feasible solutions.

Machine learning models such as convolutional neural networks (CNNs) are powerful tools to model and compute highly non-linear mappings in constant computation time [16,17] and therefore are excellent candidates for fast and efficient processing of optical information, including fast CGH. Neural networks have already been successfully implemented for applications in microscopy to estimate the refractive index map of 3D objects from indirect recordings using tomographic [18] coded illumination [19], or from intensity-only recordings of out-of-focus images [20–23].

CNNs have been implemented to directly map a target intensity pattern to a phase mask at the SLM and can synthesize low resolution (64×64 pixels) holograms [24]. The limitations of this direct phase inference method are twofold. First, the model is trained using a dataset made of random SLM phase masks (CNN outputs) paired with their simulated amplitude pattern (CNN input). This supervised approach restricts training to feasible target intensity distributions of random patterns and does not train the model to identify approximate solutions for infeasible patterns that routinely occur in real-world applications. Second, convolutional layers in CNNs operate across the spatial dimensions of the input data and are best suited to model and compute mappings that preserve some spatial correspondence between input and output. With direct phase inference, the CNN performs a cross-domain mapping of the illumination pattern, defined in the image plane, to the SLM phase mask, defined in the Fourier domain. Therefore the spatial correspondence is not preserved and the CNN capabilities are underutilized.

Since holograms are synthesized with coherent electromagnetic waves, the Huygens-Fresnel principle [25] determines how a wave at any given 2D plane along the optical axis propagates to the rest of the 3D volume. It is, therefore, possible to compute phase information at the SLM indirectly by estimating the phase and amplitude of the reconstructed field anywhere else along the optical axis. A natural approach for CGH with deep learning is to estimate the hologram in the image plane by inferring phase and amplitude at $z = 0$ which best leverages the abilities of CNNs for spatial feature learning [26–28].

We introduce DeepCGH, a new algorithm for hologram synthesis that addresses all these issues by employing a CNN to perform image plane holography with unsupervised training. A trained CNN first infers the complex field of a feasible hologram at the image plane that best matches the desired illumination pattern. The CGH solution, phase at the SLM plane, is then obtained indirectly by simulating the reverse propagation of the estimated wave to the SLM plane. In order to enable the unsupervised training of the model, our algorithm computes a virtual reconstruction of the hologram based on the estimated CGH solution and compares it to the target intensity pattern with an explicitly defined loss function. The computed loss across several training examples is used to optimize the parameters of the CNN.

In section 2, we introduce the DeepCGH algorithm. In section 3, we compare DeepCGH to existing algorithms in terms of speed and accuracy. Simulation results indicate that DeepCGH outperforms the current state of the art and yields holograms several orders of magnitude faster and with superior accuracy than existing algorithms. We present experimental results for a

CGH application in neural photostimulation with a holographic multiphoton microscope and demonstrate enhanced two-photon absorption in holograms generated with DeepCGH.

2. Methods

2.1. Optical setup

Our CGH setup is based on the standard Fourier holography configuration shown in Fig. 1. A collimated laser beam propagating along the optical axis, z, illuminates the active surface of an SLM with a static amplitude profile, $A_{Laser}(x, y)$. A computer controlling the SLM applies a custom phase pattern, $\phi_{SLM}(x, y)$, to the laser beam. The modulated complex wave, $P_{SLM}(x, y) = A_{Laser}(x, y)e^{i\phi_{SLM}(x,y)}$, propagates through the optical system that consists of a single lens and renders a volumetric image that is defined by its complex field, $P(x, y, z)$, in the image space. We define the center of this image, $z = 0$, at a distance f from the other side of the lens. In this f-f configuration, the task of a CGH algorithm is to identify the phase modulation, ϕ_{SLM}, for which $|P(x, y, z)|^2$ best matches the desired target intensity distribution.

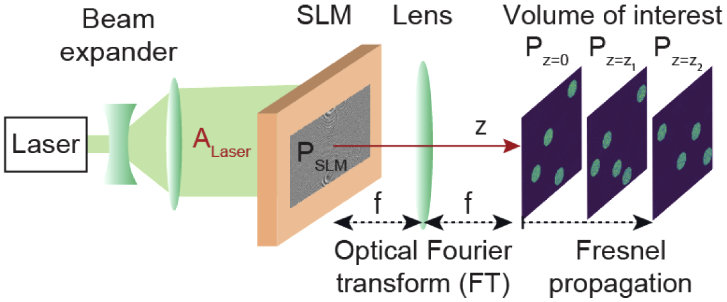

Fig. 1. Optical set up for Fourier holography. A SLM modulates a collimated coherent wave in the pupil plane of a thin lens to render a 3D intensity pattern discretized as a series of parallel planes along the optical axis.

The complex field at the center of the image plane, $P(x, y, z = 0)$, is a 2D “Optical” Fourier Transform (FT) of the field at the SLM plane, $P_{SLM}(x, y)$ (see Fig. 1), and is expressed using the Fraunhofer wave propagation model [25]:

$$P(x, y, z = 0) = \frac{1}{i\lambda f} \iint P_{SLM}(x', y') \exp\left[\frac{2i\pi(xx' + yy')}{\lambda f}\right] dx'dy'. \tag{1}$$

The phase and amplitude of the wave in the image plane, $P(x, y, z = 0)$, fully determines the intensity distribution in the entire 3D volume. The field at any location x, y, z is determined by the Fresnel wave propagation equation [25]:

$$P(x, y, z) = \iint \frac{P(x', y', 0)}{\sqrt{i\lambda z}} \exp\left[\frac{i\pi((x - x')^2 + (y - y')^2)}{\lambda z}\right] dx'dy'. \tag{2}$$

The volume of interest is decomposed into a series of adjacent planes at predetermined depths around $z = 0$ (e.g. $P_{z=0}$, $P_{z=z_1}$, and $P_{z=z_2}$ in Fig. 1) and discretized into $m \times m \times p$ voxels, where m is the edge length of the square hologram and p is the number of planes defining the volume. Naturally, propagation to planes located before $z = 0$ correspond to negative values for z in Eq. (2). We note that our method can be implemented with any holographic setup as long as it can be modeled with a differentiable transfer function from the SLM plane to the image plane.

The SLM pixel size, p_s, the wavelength λ, and the focal length of the lens, f, determine the span, $L = \lambda f/p_s$, of the addressable window in the (x, y) domain.

2.2. DeepCGH algorithm

Our algorithm is shown in Fig. 2(a). Input 3D target amplitudes, $A(x, y, z)$, consist of a multi-channel image with edges size of m and p channels that correspond to each depth plane. A *trained* CNN maps the target amplitude, $A(x, y, z)$, to a feasible approximation of the desired illumination pattern in the image plane by estimating the complex field at $z = 0$, $\hat{P}(x, y, z = 0)$. The complex field is then virtually propagated back to the SLM plane with an inverse 2D Fourier transform (Eq. (1)) to yield the solution to the CGH problem, i.e. SLM phase mask φ_{SLM}.

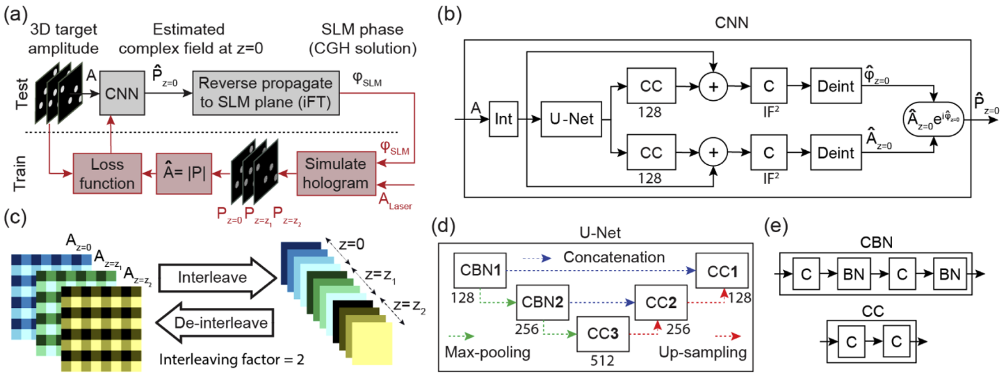

Fig. 2. DeepCGH algorithm. (a) A trained CNN *estimates* the complex field in the image plane at $z = 0$, $\hat{P}_{z=0}$, which is then reverse propagated to the SLM plane to yield the solution of the CGH problem, φ_{SLM}. During training of the CNN, the objective is to maximize the fidelity of the simulated illumination pattern, $\hat{A}$, that results from the estimated SLM phase. (b) CNN structure for DeepCGH. A plus sign, +, represents concatenation of the tensors along the channel dimension. *C* and *CC* indicate one and two convolutional layers respectively. The number of convolutional kernels is shown below each convolutional layer. Interleaving (Int) and De-interleaving (Deint) modules reshape the input and output of the CNN. Phase and amplitude are predicted by two separate paths. (c) The interleave module reduces the spatial dimensions of the network input without losing information by rearranging the pixels of the original input into multi-channel images of smaller size. The de-interleave module performs the inverse operation. (d) The U-Net model contains two types of convolutional blocks, CCn and CBNn, where n denotes the block number. The number of convolutional kernels is denoted below each convolutional block. (e) CCn and CBNn convolutional blocks, with convolutional, *C* and batch-normalization, *BN*, layers.

In this configuration, termed *image plane holography*, the input and output of the CNN share significant spatial correspondence, which simplifies the mapping that the CNN performs, thereby facilitating training [26,29–31]. During operation, this two-step sequence enables fast computation of the holograms in a fixed and predictable time window that only depends on the model size (number of convolutional layers, number of kernels, and kernel size), as well as on the number of depth planes and hologram resolution.

Our algorithm is unsupervised, i.e. trained without explicitly providing the optimal solution φ_{SLM}. For training, (see Fig. 2(a) in red), we first simulate the intensity pattern that results from the predicted SLM phase, $|\hat{A}(x, y, z)|^2$, and compare this simulated hologram with the target intensity pattern $|A(x, y, z)|^2$ via an explicitly defined loss function, $\mathcal{L}(A, \hat{A})$.

To this end, we simulate the complex field that will be synthesized at the SLM plane, $P_{SLM} = A_{Laser}e^{i\varphi_{SLM}}$. We virtually propagate the resulting wave to the image plane at $z = 0$ via a FT (Eq. (1)), and then to each plane in the discretized volume via the Fresnel propagation (Eq. (2)). The CNN is trained by minimizing the loss $\mathcal{L}(A, \hat{A})$ on a large pool of sample target intensity patterns. We employed the Adam optimizer [32] to find a suitable set of parameters for the CNN.

2.3. CNN structure and interleaving

Figure 2(b) shows the main components of the CNN structure. At the heart of our CNN is a fully convolutional network (U-Net model [33]) with five convolutional blocks that process the data at multiple scales (see Fig. 2(c)). Two convolutional blocks in the contracting path (denoted with CBNn, where n is the block number) are each followed by a max-pooling layer (green arrows) that halves the spatial dimensions of the data. Each CBNn module consists of two convolutional layers, with the same number of kernels, that are each followed by batch-normalization layers (see Fig. 2(d)) [34]. In the expanding path, three convolutional blocks (denoted with CCn, where n is the block number) are followed by up-sampling layers (red arrows) that doubles the spatial dimensions. The structure of CCn consists of two convolutional layers without the batch normalization (BN) layers (see Fig. 2(d)), since early simulation results indicated that they reduce convergence speed.

Two convolutional blocks after the U-Net (denoted with CC in Fig. 2(b)) create separate paths that will compute the phase, $\hat{\varphi}_{z=0}$, and amplitude, $\hat{A}_{z=0}$, of the complex field in the image plane at $z = 0$. Concatenation, denoted with + in Fig. 2(b), facilitates residual learning by the network [29–31,35].

The convolutional layers account for the majority of the computational cost of the DeepCGH algorithm, which is proportional to the size of their input along the (x, y) dimensions. To significantly reduce the spatial dimensions of the input to the U-Net and therefore decrease the computation time associated with the U-Net model, we implemented interleaving. The interleaving module [36,37], (see Fig. 2(e)) rearranges the raw target amplitude pattern $A(x, y, z)$ with size $m \times m \times p$ voxels into $IF^2 \times p$ channels with spatial dimension of $m/IF \times m/IF$ where IF is the interleaving factor. Each channel consists of a periodic sampling of pixels from the input that are separated by a specific distance IF along the (x, y) dimensions. In the illustrative example shown in Fig. 2(e), $m = 8$, $p = 3$, and $IF = 2$, thus the rearranged output has $3 \times 2^2 = 12$ channels. The color of each channel maps to the location of pixels in the original input data. Interleaving is a loss-free transformation that preserves the number of pixels.

As we increase the image size, we proportionally increase the IF to maintain a fixed spatial dimension in the input of the U-Net. The number of channels in the rearranged tensor increases quadratically with IF, which in turn slightly increases the computation time of the first convolutional layer but reduces the computation time of the rest of the network. Since interleaving compresses all the information in the first convolutional layer, a trade-off exists between speed gain and performance, and excessively increasing the IF will create an information bottleneck and result in degraded performance. The IF is also limited by the CNN structure and the number of pooling layers. In our model the U-Net consists of three max-pooling layers that each halve the spatial dimensions of the feature maps. The spatial dimension of the feature maps in the bottleneck of the U-Net is eight times smaller than the U-Net input, and should be considered while determining a suitable IF.

Interleaving increases the CNN's ability to generate high spatial frequency patterns [36], thus improving CNN's capability in modeling highly non-linear mappings. Interleaving also increases the CNN's receptive field, which is defined as the area along the spatial dimensions of the input image that affect each pixel in the output of the last convolutional layer. The receptive field of the first convolutional layer along the (x, y) dimensions with interleaving is IF^2 times the original receptive field. Therefore, interleaving will increase the receptive field of the entire network by a factor IF^2. A larger receptive field enhances the CNN's ability to perform CGH tasks by allowing each pixel in the input image to contribute to the output of the CNN.

DeepCGH can accommodate any orthogonal sampling of the 3D volume, simply by increasing the number of input channels or the input size. However, each new discretization of the 3D image domain and each SLM resolution requires a separate model that must be trained using a matched dataset. We kept the overall CNN structure (number of convolutional layers, number of convolutional kernels in each layer, and kernel size) fixed regardless of the edge size, m, and number of depth planes, p.

2.4. Loss function and feasibility criteria

To evaluate the performance of our algorithm in comparison with other CGH methods, we calculate the accuracy, AC, a measure of the similarity between the target intensity pattern, $I = |A(x, y, z)|^2$, and the simulated hologram, $\hat{I} = |\hat{A}(x, y, z)|^2$. The accuracy is based on the Euclidean norm in the volume of interest and is defined as:

$$AC(I, \hat{I}) = \frac{\sum_{x,y,z}\left(I.\hat{I}\right)}{\sqrt{\left[\sum_{x,y,z} I^2\right]\left[\sum_{x,y,z} \hat{I}^2\right]}}. \qquad (3)$$

We note that $AC = 1$ when the target and reconstruction are identical after normalization, i.e. when $I = K\hat{I}$, with $K{>}0$, and $0 \leq AC{<}1$ when the distributions are mismatched. Accuracy is differentiable with respect to the network parameters and therefore a suitable loss function for training DeepCGH.

A target intensity pattern is considered feasible when there exists an optimal value for φ_{SLM} for which the reconstructed pattern exactly matches the target, i.e. $AC = 1$. In practice, target intensity patterns for real-world applications are generally infeasible, and CGH algorithms can only compute a feasible approximation of the desired illumination pattern. When a hologram is not feasible, the approximation will always satisfy $AC{<}1$, even using the best CGH algorithm. The performance of CGH algorithms can be compared measuring the respective accuracy of the illumination patterns they yield, and the respective feasibility of any two patterns can be measured empirically by comparing their best achievable accuracy.

3. Results

We demonstrate DeepCGH's improved performance in comparison to existing techniques, both in simulation and experimentally, for applications in *3D cellular photostimulation* and *two-photon holographic optogenetics*. To evaluate our algorithm under a broad range of realistic experimental conditions, we trained and tested DeepCGH for various hologram resolutions and discretizations of the target illumination volume. We considered two SLM resolutions of square edge size, $m = 512$ and $m = 1024$ pixels, and three discretizations of the volume of interest into 3, 7, or 11 depth planes. For $m = 512$ and $m = 1024$ pixels we considered $IF = 16$ and $IF = 32$ respectively, setting the spatial dimensions of the input to the U-Net model at 32×32 pixels.

For each of the considered scenarios, we generated 30,000 samples for training each DeepCGH model with an additional 1,000 samples for testing. We used the same test samples again in the simulation of the other CGH techniques for a direct comparison with our model. Each sample, $A(x, y, z)$, consists of non-overlapping disks at random locations with fixed radius of 10 pixels. This particular data represents a suitable choice for applications in biology, where CGH is routinely implemented to illuminate custom groups of cell-sized targets. The amplitude within each disk is randomly assigned to values between 0.2 and 1 to account for the need to precisely place variable amounts of optical power in each target. We also normalize the input data to enforce conservation of energy across all depth planes, an essential feasibility criteria that target intensity distributions must meet. The total intensity is adjusted to keep voxel amplitudes between 0 and 1 and facilitate learning in the CNN.

We selected simulation parameters matching our experimental capabilities with a laser wavelength $\lambda = 1035$nm, a $f = 200$mm SLM lens, and a SLM with a pixel size of $p_s = 9.2$µm. In this configuration, the accessible window in the image plane is a square of size $X = f\lambda/p_s =$ 22.5mm. Adjacent depth planes are separated by 10mm, hence the total depth of the volume of interest is determined by the number of depth planes, p, and is equal to 20mm, 60mm, and 100mm for $p = 3$, $p = 7$, and $p = 11$, respectively.

We compared DeepCGH to two other existing CGH techniques. First, we considered Global Gerchberg-Saxton (GS), the simplest and most commonly used 3D variation of the Gerchberg-Saxton algorithm [9,38]. Second, we implemented Non-Convex Optimization for Volumetric CGH (NOVO-CGH) [10], one of the current state of the art CGH algorithms that yields holograms with very high accuracy by performing direct optimization of the phase in a forward model. We implemented DeepCGH in Python programming language using Tensorflow 2.0 deep learning framework [39] as we show in Code 1 [40] and trained our models with Nvidia GeForce RTX 2080Ti GPUs. The GS and NOVO-CGH algorithms were implemented with MATLAB and CUDA GPU libraries. All methods were tested on an Nvidia Titan RTX GPU.

3.1. Simulation results and computation speed

We compared the performance of DeepCGH with GS and NOVO-CGH by recording the computation time needed to yield the SLM phase mask and by measuring the accuracy of the simulated illumination patterns that different CGH techniques yield. Simulation results are shown in Fig. 3. An example target intensity pattern for $m = 512$ and $p = 7$ at depth $z = -30$mm is shown along with the simulated reconstructions obtained with all three algorithms (Fig. 3(a)). Supplementary Visualization 1 and Visualization 2 show 3D reconstructions of the simulated intensity pattern at different z values for the $512^{2*}7$ and $1024^{2*}11$ models respectively. Supplementary Visualization 3 shows projection views of a sample from the $1024^{2*}11$ DeepCGH model.

All CGH algorithms yield approximate solutions that place a significant amount of light in off-target locations. At each plane, we observe diffused patterns, in addition to the desired targets, that correspond to the footprint of photons focused on targets in adjacent planes. A qualitative comparison of the reconstructed images in Fig. 3(a) indicates that DeepCGH identifies solutions that spread the energy of these footprints more efficiently across space.

To quantitatively validate this observation, we measured the average accuracy of simulated reconstructions as well as the computation time of DeepCGH, $T_{DeepCGH}$, for 1,000 test samples. We compared those results to simulations of iterative CGH techniques on the same testing dataset for up to 3,500 iterations per sample.

Simulation results for the 512 pixels model with 7 planes and 1024 pixels model with 11 planes are shown in Fig. 3(b). Holograms generated with DeepCGH (in green) achieve significantly higher accuracy than existing techniques even after a large number of iterations.

Figure 3(c) shows the computation time for DeepCGH, $T_{DeepCGH}$, for six models with various resolutions and number of depth planes. We observed slight variations across samples, which can be attributed to the multi-threaded nature of operations by the operating system. The computation time increases with p and m, partly due to data transfer latency. Larger p and m values also increase the computation complexity of the first convolutional layer as well as the Fourier transform at the output of the model.

We also compared $T_{DeepCGH}$ to the computation time, T, required for iterative methods to reach 95% of their maximum accuracy. The speed gain factor, $T/T_{DeepCGH}$, (Fig. 3(d)) shows that DeepCGH is at least ten times faster than the iterative algorithms.

The accuracy (Fig. 3(e), in green), of DeepCGH is consistently higher than the maximum achievable accuracy of GS (red) and NOVO-CGH (blue) for all m and p, except for $m = 512$ and $p = 11$ where NOVO-CGH slightly outperforms DeepCGH. In this particular case, all

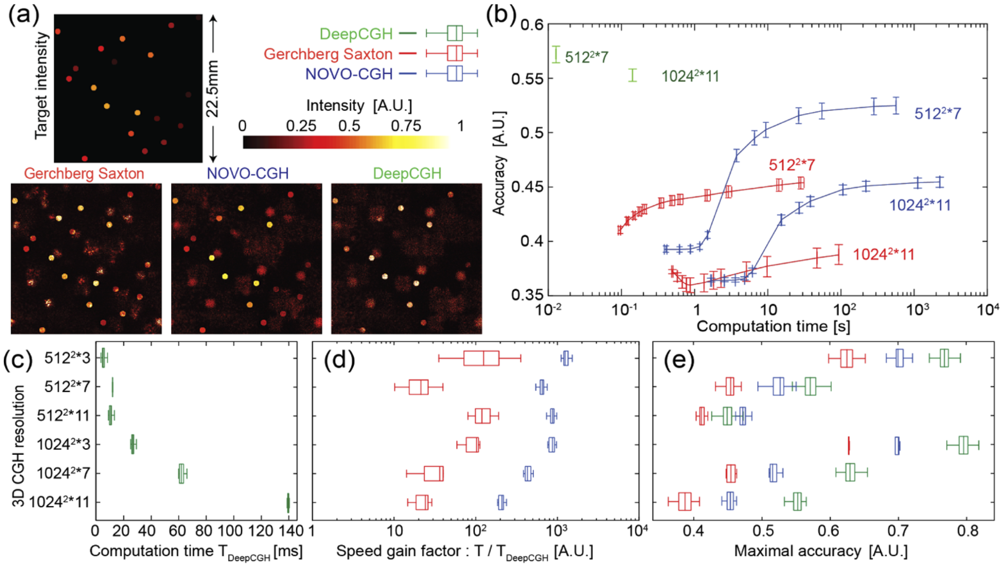

Fig. 3. Simulation results comparing DeepCGH to existing algorithms. (a) Example target distribution and simulated reconstructions of the intensity computed with GS, NOVO-CGH, and DeepCGH, at $z = -30$mm with SLM resolution $m = 512$. (b) Accuracy of the simulated 3D intensity reconstruction as a function of the computation time for holograms of size $m = 512$, with 7 depth planes, and $m = 1024$ with 11 depth planes. Computation speeds for GS and NOVO-CGH are shown for an increasing number of up to 3,500 iterations. Simulation results for six different 3D models show (c) the inference time for DeepCGH, (d) the ratio of the computation time, T, for iterative methods to reach 95% of the maximum accuracy they can reach, to the inference time of DeepCGH, $T_{DeepCGH}$, and (e) the maximum achievable accuracy of iterative methods in comparison to the accuracy of DeepCGH.

algorithms fail to identify a well-matched hologram. The average accuracy values (0.47 vs 0.45 for NOVO-CGH and DeepCGH, respectively) are overall low and the observed 2% difference is not substantial. In this case, the CGH problem is more difficult with 11 consecutive intensity constraints but with less degrees of freedom due to the small SLM resolution. This is evident in the case with $m = 1024$ and $p = 11$, where, for the same number of depth planes, DeepCGH yields solutions with higher accuracy than iterative methods.

With any CGH method, the accuracy of the holograms depends on the resolution (edge size m), number of depth planes (p), and on the feasibility of each individual target intensity pattern (see Fig. 3(e)). For the values of m and p we selected, DeepCGH was 9% to 43% more accurate than GS, and at least 10 times faster. Compared to NOVO-CGH and excluding the case where $m = 512$ and $p = 11$, DeepCGH was between 9% to 22% more accurate and at least 200 times faster. Overall, simulation results show that DeepCGH adequately computes high fidelity 3D holograms, significantly faster and with greater accuracy than the current state of the art.

3.2. Experimental results

Two-photon holographic photostimulation [41,42] is an immediate application that directly benefits from DeepCGH. 3D CGH is routinely used to activate custom groups of neurons expressing a photosensitive opsin [13,43]. Fast computation is necessary to elicit patterns of neural activity on demand in awake animals [1], and high accuracy is critical to maximize two-photon absorption by confining light into the desired targets while minimizing brain tissue heating [44,45]. In this set of experiments, we compared 3D measurements of two-photon absorption in a florescent calibration slide for holograms targeting identical distributions, but generated with different CGH techniques.

Our experimental setup is shown in Figure 4(a). We integrated a Fourier holography setup into the path of a femtosecond laser beam (Coherent Monaco 1035-80-60) providing coherent illumination, A_{Laser}, at wavelength $\lambda = 1035$nm and collimated on a reflective liquid crystal on silicon (LCoS) SLM (Meadowlark, 1920x1152) in the pupil plane of a $f = 200$mm lens. For simplicity, Fig. 4(a) shows the reflected beam on the other side of the SLM, as with a transmissive SLM design. The undiffracted light is reduced with a reverse pinhole made by grinding a <0.2mm hole in the center of an optical flat. The hologram is relayed into a microscope with a 16X objective (Nikon 16x /0.8W LWD) mounted on a precision mechanical stage (Sutter MP285).

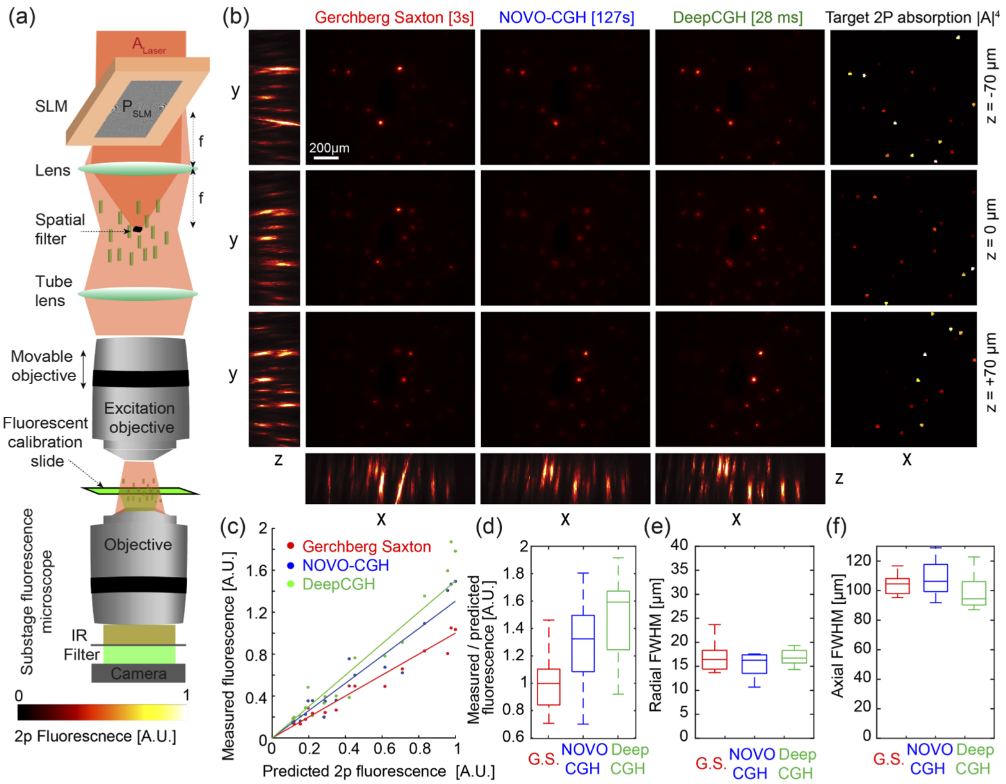

Fig. 4. (a) Experimental setup for multiphoton holography. The SLM and lens are placed in the optical path of a microscope to demagnify the hologram by a factor of 16. An ultra-fast laser provides a coherent light source, and a reverse pinhole blocks any undiffracted light. A calibration slide with a thin, uniform fluorescent film intercepts the hologram at a custom depth, z, and the corresponding two-photon fluorescence image is captured with sub-stage objective coupled to a camera. (b) Left, experimental measurements of fluorescence in the calibration slide observed at each of the $p = 3$ depth planes, and maximum intensity projection images along the (y) and (x) axes for holograms computed with GS, NOVO-CGH and DeepCGH and synthesized with the same amount of laser power. Right, uncompensated target two-photon absorption, $|A|^4$. (c) Two-photon fluorescence measured in $n = 20$ disks randomly picked in our experimental data as a function of the expected amount of two-photon absorption after power calibration to account for spatial dependency of the diffraction efficiency. (d) Same data as in (c) but showing the ratio of measured to predicted fluorescence. (e) Radial Full Width at Half Maximum (FWHM) for $n = 20$ randomly selected disks. (f)same as in (e) but showing axial FWHM.

A thin fluorescent calibration slide (Tamiya red TS36 on a glass slide) is placed under the objective, allowing visualizations of two-photon absorption at any depth to be captured by a camera (Thorlabs DCC1645M) with a substage microscope. To map multiphoton absorption in 3D, Z-stack images of the two-photon (2p) fluorescence are recorded by moving the excitation objective in 1µm increments along the z-axis and capturing one frame at each depth. Infrared filters (two Thorlabs FES0800) remove the remaining infrared light so that only fluorescence photons can be captured by the camera. The camera signal is only counted if it is twice above the standard dark noise level to eliminate background artifacts.

We considered samples similar to our simulation data with $m = 1024$. For each sample, we computed phase masks with GS and NOVO-CGH at maximal accuracy, and with DeepCGH. We placed each phase mask on the SLM to compare the resulting holograms under identical experimental conditions. Using the substage microscope, we recorded three 2D images of the fluorescence induced by two-photon absorption in the calibration slide. We captured two-photon fluorescence images in random order to account for possible photobleaching. We repeated the process while mechanically moving the excitation objective along the z axis by 1µm increments. The resulting volumetric images are quantitative measurements of two-photon absorption that compare the performance of the three CGH algorithms for photostimulation tasks with the same amount of laser power.

Figure 4(b) shows experimental recordings of two-photon fluorescence induced in the calibration slice at each of the $p = 3$ depth planes as well as maximum projection images along the (x) and (y) axis for a hologram with resolution $m = 1024$. The three fluorescence images are perfectly co-aligned and address the desired custom 3D locations. The fluorescence signal is brighter in the center of the image across all CGH methods. This is a well known property of Fourier holography systems that experience a gradual loss of diffraction efficiency when placing light at increasing radial distances from the optical axis. This issue is generally addressed with a power calibration [1] that digitally compensates for attenuation. Measurements of accuracy in simulations predicts that our method is able to focus more light onto the desired targets than iterative CGH methods. Our experiment confirms this prediction and shows that with the same amount of laser power, DeepCGH is able to yield stronger fluorescence in the calibration slide.

Two-photon absorption is a nonlinear process that is proportional to the square of the illumination intensity. Hence, any mismatch between the desired and actual illumination intensity in targets yields an even larger error in the resulting fluorescence image. This type of error is very common in holograms synthesized with the GS algorithm that does not explicitly optimize for hologram accuracy. Maximum fluorescence intensity projection images along the y and x axis (Fig. 4(b)) show that two targets near the center of the hologram seem to have their brightness overestimated by the GS algorithm, while both NOVO-CGH and DeepCGH seem to render more power-balanced distributions.

To compare CGH methods in their ability to address randomly distributed groups of cells with precise amounts of two-photon absorption, we considered $n = 20$ individual cell-sized disks randomly picked in the samples we recorded and measured the size and total amount of fluorescence in each spot from 3D recordings of fluorescence with all three CGH methods.

Results are shown in Fig. 4(c-f) and compare experimental measurements of the total fluorescence to the amount predicted by simulations (i.e. the square of the target intensity distribution, or $|A|^4$) adjusted for radial losses in diffraction efficiency. A linear interpolation of the experimental data (see Fig. 4(c)) as well as statistical analysis of the ratio between measured and predicted fluorescence (see Fig. 4(d)) show that DeepCGH is able to yield 16% more fluorescence than NOVO-CGH holograms and 48% more fluorescence than GS holograms.

We also compared the radial (see Fig. 4(e)) and axial (see Fig. 4(f)) dimensions of the fluorescent spots by measuring the Full-Width Half-Max (FWHM) from perfectly aligned high resolution volumetric recordings of two-photon fluorescence. The data indicates that the dimensions of the 3D fluorescent spots and therefore the achievable spatial resolution are not significantly affected by the choice of a different algorithm for hologram synthesis.

An example maximum projection image for our largest model ($m = 1024$ and $p = 11$) is shown in Fig. 5(a). Experimental results show that DeepCGH enables precise targeting of cell-sized objects randomly placed in a cube of 700µm edge size (see Supplementary Visualization 4 for co-aligned 3D tomographic reconstructions and animated projection views) and yields substantially more multiphoton absorption than iterative algorithms with the same amount of laser power. We measured the total optical power under the microscope objective for each SLM phase mask (Fig. 5(b), top) using a power meter (Thorlabs PM100). We observed small fluctuations of optical power (less than 1% on average) attributed to losses in the optical path and variable amounts of undiffracted light (blocked by the reverse pinhole) that are specific to each SLM phase mask. We computed the total amount of two-photon absorption in each 3D image by integrating the fluorescence signal across the volume of interest (Fig. 5(b), center).

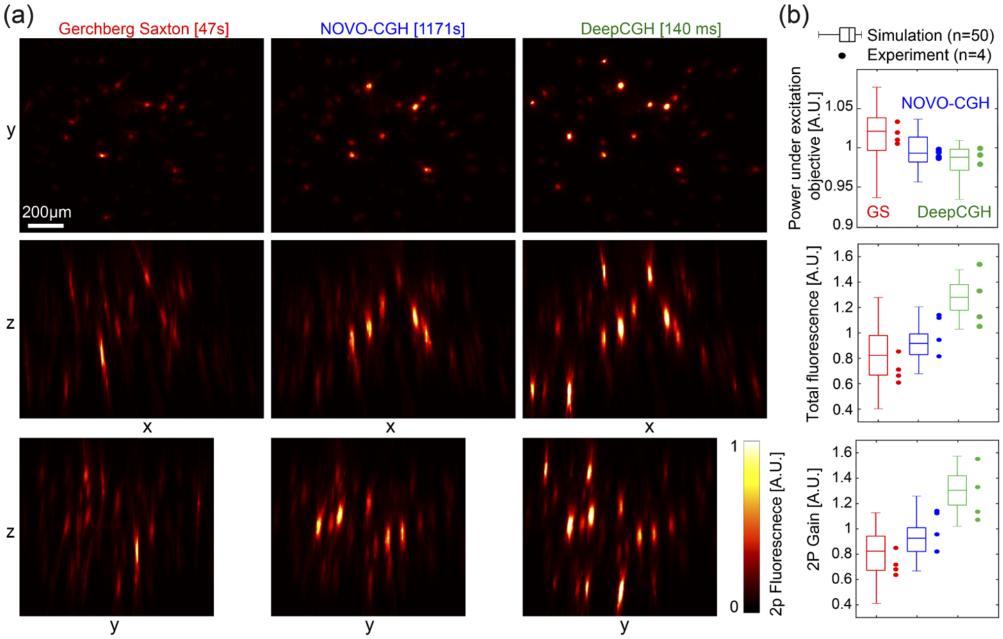

Fig. 5. (a) Maximum projection images from 3D recordings of two-photon fluorescence for holograms generated with GS, NOVO-CGH, and DeepCGH. The target intensity distribution and laser power is identical across all recordings. (b) Simulation (box-plots, n=50) and experimental (n=4) measurements of the total power under the objective (top), and total two-photon fluorescence signal in the 3D volume (center). Normalized two-photon gain defined as total fluorescence divided by the square of the intensity under the objective (bottom).

To quantify the bulk gains in multiphoton absorption at equal levels of energy under the objective (i.e. heat deposition), we measured the two-photon efficiency ratio (Fig. 5(b), bottom) by dividing the total amount of two-photon fluorescence by the square of the laser power under the objective. In addition to experimental results (data points), we also display estimated values of the same quantities from simulations on $n = 50$ holograms of the similar type (box plots).

Simulation results indicate that 3D holograms synthesized with DeepCGH have greater accuracy than existing methods. Hence, the resulting illumination patterns better concentrate photons into the targets that are more likely to undergo multiphoton absorption. Our experimental results confirm this prediction, showing that DeepCGH holograms can yield 50% more multiphoton absorption with the same amount of optical power under the objective. This substantial gain in performance directly benefits applications in biology and neuroscience by increasing two-photon photo-activation without additional tissue heating or photodamage.

3.3. Model generalizability

A CNN model is considered generalizable when it not overfitting to the dataset on which it was trained and is able to compute accurate results from previously unseen data. By this definition, DeepCGH is generalizable if a trained model can compute equally high-fidelity holograms of new unseen shapes that are not representative of the training image dataset.

Generalizability is a valuable feature for learning-based CGH algorithms, especially for experimental settings where some variability in target distributions is expected. For instance, in holographic optogenetics applications, the exact shape of neurons depends on the brain region being considered with slight variations specific to each animal. Training a separate model for each sample would be inefficient and time consuming.

To evaluate the generalizability of DeepCGH, we compared the accuracy of simulated DeepCGH holograms for models trained and tested on different types of data. We considered three datasets of 2D images with resolution $m = 512$ each with a random number of *disks*, *squares*, or *lines* randomly placed in the image field. We trained, and tested DeepCGH on all the nine possible combinations of training and testing datasets.

Accuracy measurements with 1,000 test samples of each data type are shown in Fig. 6(a). The accuracy of DeepCGH slightly decreases when the model is tested on a type of image data that differs from what the model was trained on. However, this drop in accuracy is insignificant (less than 0.5% in all cases). Simulation results also show that the *squares* model performs better than the *lines* model when tested with the *lines* data set. We believe that this could be due to the increased diversity of shapes in the *squares* data compared to the *lines* data. Image diversity helps the CNN explore a wider range of the manifold that represents the mapping from target intensities to complex field at $z = 0$. As a result, the CNN learns a mapping that is more generalized compared to models trained with the *lines* model. In some applications, one can also choose to purposefully overfit the model to a specific data type and tailor the cost function to favor a particular outcome with additional gains in performance.

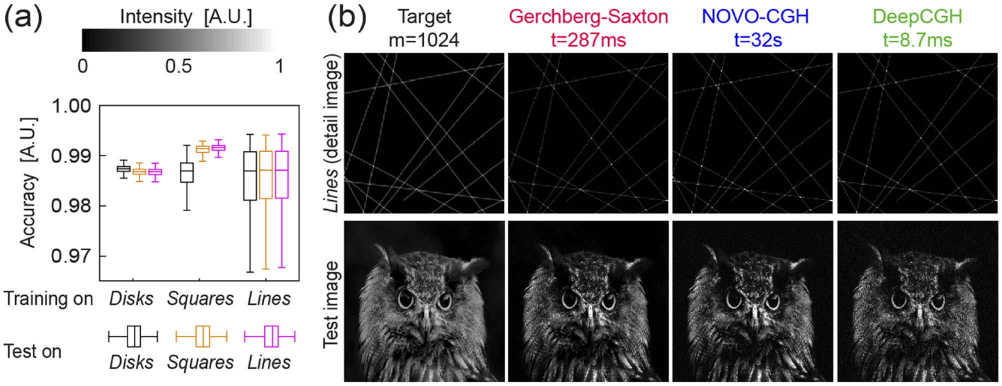

Fig. 6. (a) Comparison of the accuracy of simulated DeepCGH reconstructions when the model is trained (on 30,000 samples), then tested (on 1,000 samples) on three distinct image data sets made of either randomized *disks*, *squares*, or *lines*. (b) Example target intensities and reconstructions for GS, NOVO-CGH, and DeepCGH (trained on the *Lines* dataset) tested on line image (top) and a natural image (bottom). Line image is zoomed in to show detail.

We further illustrate the generalizabilty of DeepCGH with a model that is trained on the *lines* dataset and tested on a natural image. Simulation results in Fig. 6(b) show intensity reconstructions by each method with patterns picked from the *lines* test dataset and a natural image. DeepCGH yields image quality that compare to holograms obtained with iterative algorithms. This example perfectly illustrates the generalizability of the mapping that DeepCGH learns from a specific data type to synthesize high-fidelity holograms of completely different nature. Although DeepCGH holograms seem to contain more speckle than holograms synthesized with iterative techniques, we note that DeepCGH yields this solution 33 and 3,678 times faster than GS and NOVO-CGH, respectively, at speeds that would be compatible with online hologram synthesis at video rate.

4. Conclusion

We have developed DeepCGH, a new algorithm based on CNNs for fast and accurate 3D computer-generated holography. DeepCGH operates with a fixed computation time that is predetermined by the hologram size and model complexity. We found that DeepCGH not only synthesizes extremely large holograms (up to 11 Megavoxels) at record speeds, but also reliably identifies solutions with greater accuracy than existing techniques. We validated DeepCGH with experiments in multiphoton holography where the enhanced accuracy of our method yields substantially more two-photon absorption without hardware modifications. Image plane holography simplifies the mapping that the CNN performs and best utilizes the capabilities of CNNs compared to other CNN-based CGH approaches. DeepCGH can be easily customized to accommodate various spatial discretizations of the volume of interest and SLM resolutions by adjusting the number of input channels, the interleaving factor, or number of kernels in the CNN model. Finally, DeepCGH enables unsupervised training of the CNN, allowing the model to be tailored to custom applications of holography by selecting training datasets that best match the desired real-world experimental conditions, as there is no need to explicitly provide ground truth phase masks. Further tailoring can be achieved by customizing the loss function to optimize the model to best execute a user-defined task and directly optimize for the desired outcome instead of hologram accuracy.

Funding

Nvidia; Burroughs Wellcome Fund (5113244).

Acknowledgments

The authors thank Professors S. Chowdhury, H. Fuchs, and A. Giovannucci for providing valuable feedback on our work.

Disclosures

The authors declare no conflicts of interest.

References

1. A. R. Mardinly, I. A. Oldenburg, N. C. Pégard, S. Sridharan, E. H. Lyall, K. Chesnov, S. G. Brohawn, L. Waller, and H. Adesnik, "Precise multimodal optical control of neural ensemble activity," Nat. Neurosci. **21**(6), 881–893 (2018).
2. P. Pozzi, L. Maddalena, N. Ceffa, O. Soloviev, G. Vdovin, E. Carroll, and M. Verhaegen, "Fast calculation of computer generated holograms for 3d photostimulation through compressive-sensing gerchberg–saxton algorithm," Methods Protoc. **2**(1), 2–15 (2019).
3. W. Yang and R. Yuste, "Holographic imaging and photostimulation of neural activity," Curr. Opin. Neurobiol. **50**, 211–221 (2018).
4. K. C. Neuman and S. M. Block, "Optical trapping," Rev. Sci. Instrum. **75**(9), 2787–2809 (2004).
5. S. Koller, "Optical trapping: Techniques and applications," in Student Research Celebration, (Montana State University, 2017), pp. 1–2.

6. J.-H. Park, "Recent progress in computer-generated holography for three-dimensional scenes," J. Inf. Disp. **18**(1), 1–12 (2017).
7. R. Häussler, N. Leister, and H. Stolle, "Large holographic 3d display for real-time computer-generated holography," in *Digital Optical Technologies 2017*, vol. 10335 (International Society for Optics and Photonics, 2017), p. 103350X.
8. D. Leseberg, "Computer-generated three-dimensional image holograms," Appl. Opt. **31**(2), 223–229 (1992).
9. R. W. Gerchberg, "A practical algorithm for the determination of phase from image and diffraction plane pictures," Optik **35**, 237–246 (1972).
10. J. Zhang, N. Pégard, J. Zhong, H. Adesnik, and L. Waller, "3d computer-generated holography by non-convex optimization," Optica **4**(10), 1306–1313 (2017).
11. P. Chakravarthula, Y. Peng, J. Kollin, H. Fuchs, and F. Heide, "Wirtinger holography for near-eye displays," ACM Trans. Graph. **38**(6), 1–13 (2019).
12. P. Chakravarthula, Y. Peng, J. Kollin, F. Heide, and H. Fuchs, "Computing high quality phase-only holograms for holographic displays," in *Optical Architectures for Displays and Sensing in Augmented, Virtual, and Mixed Reality (AR, VR, MR)*, vol. 11310 (International Society for Optics and Photonics, 2020), pp. 1–16.
13. N. C. Pégard, A. R. Mardinly, I. A. Oldenburg, S. Sridharan, L. Waller, and H. Adesnik, "Three-dimensional scanless holographic optogenetics with temporal focusing (3d-shot)," Nat. Commun. **8**(1), 1228 (2017).
14. Z. Zhang, L. E. Russell, A. M. Packer, O. M. Gauld, and M. Häusser, "Closed-loop all-optical interrogation of neural circuits in vivo," Nat. Methods **15**(12), 1037–1040 (2018).
15. Y. Wang, D. Dong, P. J. Christopher, A. Kadis, R. Mouthaan, F. Yang, and T. D. Wilkinson, "Hardware implementations of computer-generated holography: a review," Opt. Eng. **59**(10), 1–15 (2020).
16. Y. Kiarashinejad, S. Abdollahramezani, and A. Adibi, "Deep learning approach based on dimensionality reduction for designing electromagnetic nanostructures," npj Comput. Mater. **6**(1), 12 (2020).
17. Y. Kiarashinejad, M. Zandehshahvar, S. Abdollahramezani, O. Hemmatyar, R. Pourabolghasem, and A. Adibi, "Knowledge discovery in nanophotonics using geometric deep learning," Adv. Int. Sys. **2**, 1900132 (2020).
18. 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).
19. Y. Xue, S. Cheng, Y. Li, and L. Tian, "Reliable deep-learning-based phase imaging with uncertainty quantification," Optica **6**(5), 618–629 (2019).
20. Y. Wu, Y. Rivenson, Y. Zhang, Z. Wei, H. Günaydin, X. Lin, and A. Ozcan, "Extended depth-of-field in holographic imaging using deep-learning-based autofocusing and phase recovery," Optica **5**(6), 704–710 (2018).
21. Y. Rivenson, Y. Zhang, H. Günaydın, D. Teng, and A. Ozcan, "Phase recovery and holographic image reconstruction using deep learning in neural networks," Light: Sci. Appl. **7**(2), 17141 (2018).
22. Y. Rivenson, Y. Wu, and A. Ozcan, "Deep learning in holography and coherent imaging," Light: Sci. Appl. **8**(1), 85 (2019).
23. F. Wang, Y. Bian, H. Wang, M. Lyu, G. Pedrini, W. Osten, G. Barbastathis, and G. Situ, "Phase imaging with an untrained neural network," Light: Sci. Appl. **9**(1), 77 (2020).
24. R. Horisaki, R. Takagi, and J. Tanida, "Deep-learning-generated holography," Appl. Opt. **57**(14), 3859–3863 (2018).
25. J. W. Goodman, *Introduction to Fourier optics* (Roberts and Company Publishers, 2005).
26. M. H. Eybposh, N. W. Caira, P. Chakravarthula, M. Atisa, and N. C. Pégard, "High-speed computer-generated holography using convolutional neural networks," in *Optics and the Brain*, (Optical Society of America, 2020), pp. BTu2C–2.
27. Y. Peng, S. Choi, N. Padmanaban, J. Kim, and G. Wetzstein, "Neural holography," in *SIGGRAPH2020, Seeing is Believing: XR Displays, Holograms, and Dyes*, (2020), pp. 3–4.
28. S. Rosen and S. Shoham, "Holographic display for optical retinal prosthesis: design and validation," in *Optogenetics and Optical Manipulation*, (2020), pp. 3–4.
29. I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning* (MIT, 2016).
30. F. Khoshnevisan and Z. Fan, "Rsm-gan: A convolutional recurrent gan for anomaly detection in contaminated seasonal multivariate time series," https://arxiv.org/abs/1911.07104.
31. M. H. Eybposh, M. H. Ebrahim-Abadi, M. Jalilpour-Monesi, and S. S. Saboksayr, "Segmentation and classification of cine-mr images using fully convolutional networks and handcrafted features," http://arxiv.org/abs/1709.02565.
32. D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," https://arxiv.org/abs/1412.6980.
33. 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.
34. S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," http://arxiv.org/abs/1502.03167.
35. K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," https://arxiv.org/abs/1412.6980.
36. W. Shi, J. Caballero, F. Huszár, J. Totz, A. P. Aitken, R. Bishop, D. Rueckert, and Z. Wang, "Real-time single image and video super-resolution using an efficient sub-pixel convolutional neural network," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, (2016), pp. 1874–1883.
37. L. Xiao, A. Kaplanyan, A. Fix, M. Chapman, and D. Lanman, "Deepfocus: Learned image synthesis for computational display," in *SIGGRAPH 2018 Talks*, (ACM, 2018), pp. 1–23.

38. R. Piestun, B. Spektor, and J. Shamir, “Wave fields in three dimensions: analysis and synthesis,” J. Opt. Soc. Am. A **13**(9), 1837–1848 (1996).
39. M. Abadi, A. Agarwal, P. Barham, E. Brevdo, Z. Chen, C. Citro, G. S. Corrado, A. Davis, J. Dean, M. Devin, S. Ghemawat, I. Goodfellow, A. Harp, G. Irving, M. Isard, Y. Jia, R. Jozefowicz, L. Kaiser, M. Kudlur, J. Levenberg, D. Mané, R. Monga, S. Moore, D. Murray, C. Olah, M. Schuster, J. Shlens, B. Steiner, I. Sutskever, K. Talwar, P. Tucker, V. Vanhoucke, V. Vasudevan, F. Viégas, O. Vinyals, P. Warden, M. Wattenberg, M. Wicke, Y. Yu, and X. Zheng, “TensorFlow: Large-scale machine learning on heterogeneous systems,” (2015). Software available from tensorflow.org.
40. M. H. Eybposh, “Python implementation of deepcgh,” Zenodo (2020). Https://doi.org/10.5281/zenodo.3988713.
41. V. Nikolenko, B. O. Watson, R. Araya, A. Woodruff, D. S. Peterka, and R. Yuste, “Slm microscopy: scanless two-photon imaging and photostimulation using spatial light modulators,” Front. Neural Circuits **2**, 5–19 (2008).
42. G. S. He, P. P. Markowicz, T.-C. Lin, and P. N. Prasad, “Observation of stimulated emission by direct three-photon excitation,” Nature **415**(6873), 767–770 (2002).
43. E. Papagiakoumou, V. De Sars, D. Oron, and V. Emiliani, “Patterned two-photon illumination by spatiotemporal shaping of ultrashort pulses,” Opt. Express **16**(26), 22039–22047 (2008).
44. A. Picot, S. Dominguez, C. Liu, I.-W. Chen, D. Tanese, E. Ronzitti, P. Berto, E. Papagiakoumou, D. Oron, G. Tessier, B. C. Forget, and V. Emiliani, “Temperature rise under two-photon optogenetic brain stimulation,” Cell Rep. **24**(5), 1243–1253.e5 (2018).
45. K. Podgorski and G. Ranganathan, “Brain heating induced by near-infrared lasers during multiphoton microscopy,” J. Neurophysiol. **116**(3), 1012–1023 (2016).