Neural Holography with Camera-in-the-loop Training

YIFAN PENG, SUYEON CHOI, NITISH PADMANABAN, and GORDON WETZSTEIN, Stanford University

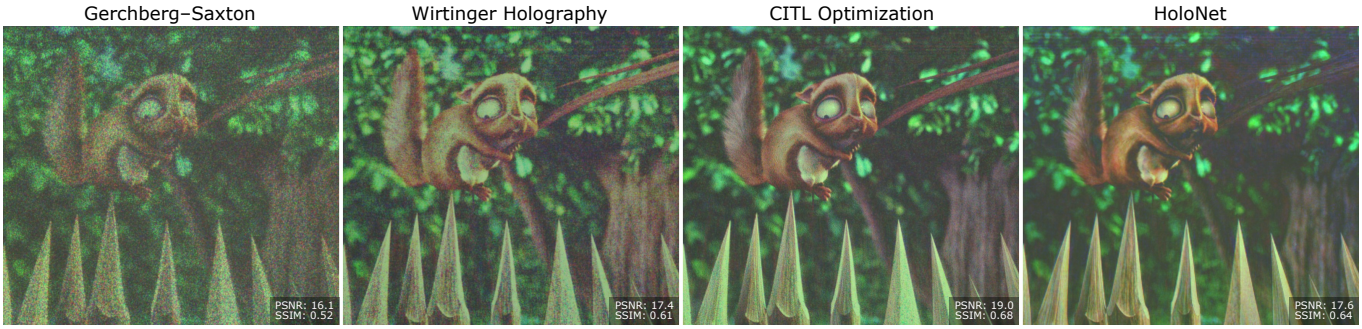

Fig. 1. Comparison of computer-generated holography (CGH) algorithms captured with a prototype holographic near-eye display. The classic Gerchberg–Saxton approach is intuitive but it suffers from speckle and other artifacts (left). Wirtinger Holography was recently introduced as an iterative CGH method that achieves better image quality (center left). We introduce camera-in-the-loop (CITL) optimization strategies that achieve unprecedented holographic image quality (center right). Moreover, we introduce a neural network architecture, HoloNet, that achieves a quality comparable to the best existing iterative approaches in real time for full-resolution 1080p images (right).

Holographic displays promise unprecedented capabilities for direct-view displays as well as virtual and augmented reality applications. However, one of the biggest challenges for computer-generated holography (CGH) is the fundamental tradeoff between algorithm runtime and achieved image quality, which has prevented high-quality holographic image synthesis at fast speeds. Moreover, the image quality achieved by most holographic displays is low, due to the mismatch between the optical wave propagation of the display and its simulated model. Here, we develop an algorithmic CGH framework that achieves unprecedented image fidelity and real-time framerates. Our framework comprises several parts, including a novel camera-in-the-loop optimization strategy that allows us to either optimize a hologram directly or train an interpretable model of the optical wave propagation and a neural network architecture that represents the first CGH algorithm capable of generating full-color high-quality holographic images at 1080p resolution in real time.

CCS Concepts: • **Hardware → Emerging technologies**; • **Computing methodologies → Computer graphics**.

Additional Key Words and Phrases: computational displays, holography, virtual reality, augmented reality

ACM Reference Format:
Yifan Peng, Suyeon Choi, Nitish Padmanaban, and Gordon Wetzstein. 2020. Neural Holography with Camera-in-the-loop Training. *ACM Trans. Graph.* 39, 6, Article 185 (December 2020), 14 pages. https://doi.org/10.1145/3414685.3417802

Authors' address: Yifan Peng, evanpeng@stanford.edu; Suyeon Choi, suyeon@stanford.edu; Nitish Padmanaban, nit@stanford.edu; Gordon Wetzstein, gordon.wetzstein@stanford.edu, Stanford University, 350 Jane Stanford Way, Stanford.

0730-0301/2020/12-ART185 $15.00
https://doi.org/10.1145/3414685.3417802

1 INTRODUCTION

Computer-generated holography has recently experienced a renaissance in the computer graphics and computational optics communities. For direct-view displays, holography enables glasses-free 3D display modes and in virtual and augmented reality systems, 2D or 3D holography has the potential to optimize some of the biggest remaining challenges, such as focus cues, vision correction, device form factors, image resolution, and brightness, as well as dynamic image and eyebox steering capabilities. However, the challenge of robustly and reliably achieving high image fidelity with experimental holographic displays while simultaneously achieving real-time performance remains unsolved. This challenge presents a major roadblock for making holographic displays a practical (near-eye) display technology.

Since the invention of the holographic principle by Dennis Gabor in the late 1940s, much progress has been made. The laser enabled the first optical holograms, and digital computers and spatial light modulators (SLMs) enabled holographic video based on computer-generated holography (CGH) [Benton and Bove 2008]. Over the last few decades, much effort has focused on advancing CGH algorithms (see Sec. 2). While these have become increasingly sophisticated, it is still challenging to robustly achieve an image quality approaching that of other display technologies. We argue that this discrepancy is not necessarily caused by the lack of good CGH algorithms, but by the challenge of adequately modeling a physical display system in simulation. It is easy to compute a phase pattern that should be displayed on an SLM to achieve a target image and make the result look good in simulation. However, it can be very challenging to achieve the same quality with an experimental holographic display.

To verify this claim, we ran a simple experiment shown in Figure 2 (B,C): using several different CGH algorithms, we optimize phase

patterns for a simulated holographic display (see details in Supplemental Section 1). We simulate the observed image assuming that the optical wave propagation of the display matches its simulated model and list the resulting peak signal-to-noise ratio (PSNR) and structural similarity (SSIM), averaged for 100 1080p images from the DIV2K dataset [Agustsson and Timofte 2017]. Surprisingly, a simple stochastic gradient descent approach (see Sec. 3) achieves the best results, although existing algorithms also achieve good image quality. We ran the same experiment again, assuming the same idealized wave propagation model when optimizing the SLM phase pattern, but this time we introduce a slight model mismatch between the model used in the optimization procedure and that used for simulating the image observed on the physical display. Specifically, we use calibrated laser intensity variation over the SLM, nonlinear phase distortions introduced by the SLM pixels, and optical aberrations of a prototype display for the mismatched model. The observed image quality is now significantly worse for all algorithms. We conclude that the choice of CGH algorithm is important to achieve good image quality, but that it may be even more important to calibrate and characterize a holographic display well.

Here, we develop an algorithmic CGH framework based on variants of stochastic gradient descent (SGD) to address these and other long-standing challenges of holographic displays. Using the proposed framework, we design a novel camera-in-the-loop (CITL) optimization strategy that allows us to iteratively optimize a hologram using a hybrid physical–digital wave propagation model. Specifically, this procedure uses a holographic display and a camera to show and capture intermediate results of an iterative CGH optimization method with the goal of directly optimizing the observed image rather than using a pure simulation approach. We show that this CGH approach achieves the best image fidelity to date, because it directly evaluates the error between target image and synthesized hologram using the physically observed holographic image. Our framework also allows us to automatically calibrate a differentiable wave propagation model of the physical display. This calibration procedure builds on automatic differentiation and is akin to the training phase of a neural network, where many example images are presented on a physical display and the error between captured result and target image is backpropagated into a differentiable proxy of the physical hardware system. This proxy models the intensity distribution of the laser source on the SLM, the nonlinear mapping from voltage to phase delay of each SLM pixel, and optical aberrations between the SLM and the target image plane. Unlike the global lookup tables used by commercial SLMs to model voltage-to-phase mapping, our differentiable model is capable of modeling a unique mapping function per SLM pixel and automatically calibrating them. Finally, we develop a neural network architecture, HoloNet, that is trained with our CITL-trained model, to enable full-color, high-quality holographic images at 1080p resolution in real time.

In summary, we make the following contributions:

- We derive a procedure to optimize an SLM phase pattern for a single target image with a camera in the loop. This procedure achieves unprecedented image quality—the best among all methods we evaluate.

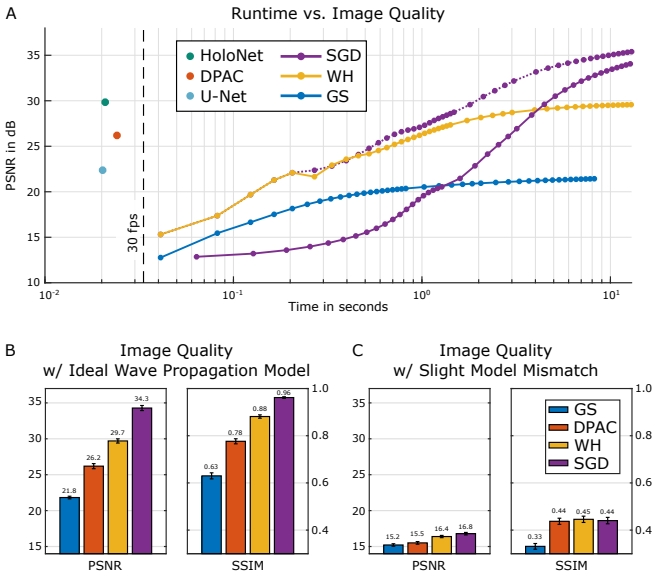

Fig. 2. Simulated results. All direct CGH algorithms, including double phase–amplitude coding (DPAC) and a U-Net neural network, achieve real-time rates of about 40 frames per second, but HoloNet is the only direct algorithm to also achieve a PSNR of ≈30 dB (A). Iterative algorithms typically achieve a better quality with more iterations. Gerchberg–Saxton (GS) converges to an average quality of just above 20 dB PSNR and Wirtinger Holography (WH) achieves a PSNR of ≈30 dB after a sufficient number of iterations. With ≈35 dB, our gradient descent approach (SGD) achieves the best quality among all CGH algorithms. We plot the convergence of SGD initialized with random phase (solid line) and initialized with the same 5 steps of the GS variant that WH uses for bootstrapping (dashed line). PSNR values of all methods are averaged over 100 test images. These results are computed assuming that the wave propagation model used for optimizing the SLM phase patterns and for simulating the final image match (B). In practice, SLM phase patterns are optimized with an ideal wave propagation model and a physical holographic display introduces a small amount of model mismatch due to optical aberrations, phase nonlinearity, and laser intensity variation on the SLM. Even a small amount of such a model mismatch causes all of these algorithms to fail (C). Error bars represent standard error.

- We propose a CITL learning strategy to train a differentiable proxy of the optical wave propagation of a specific holographic display. This model is interpretable and generalizes to unseen test images, removing the need for a camera during inference.
- We develop a network architecture, HoloNet, that incorporates the CITL-calibrated model and achieves high-quality 2D holographic images at 1080p resolution in real time.
- We explore extensions of the proposed system to varifocal and multiplane 3D holography applications.

Overview of Limitations. Most of the techniques we describe are developed for and evaluated with 2D holograms. Extending holograms to 3D has been a challenging and unsolved problem for decades. While conventional display technologies, such as liquid crystal or organic light emitting diode displays, can directly show a target image by setting their pixels' states to match those of the

image, holographic displays cannot. Holographic displays must generate a visible image indirectly through interference patterns of a reference wave at some distance in front of the SLM—and when using a phase-only SLM, there is yet another layer of indirection added to the computation. In addition to 2D CGH, we demonstrate first steps towards extending our CITL methods to 3D holography with the proposed varifocal and multiplane display modes. Finally, all of our techniques have different benefits and limitations, which we discuss in Section 8.

2 RELATED WORK

Holographic display technology and algorithms for computer-generated holography (CGH) have been active areas of research for decades. Here, we summarize the most relevant work.

Holographic Near-eye Displays. Dynamic, digital holographic displays have been enabled by phase-only spatial light modulators (SLMs) in conjunction with coherent light sources. While early efforts on technology development aimed for holographic television [Benton and Bove 2008], more recent work has focused on holographic near-eye displays for virtual and augmented reality applications. Some of these near-eye approaches aim at optimizing hardware aspects, such as diffractive optical elements [Li et al. 2016; Maimone and Wang 2020; Yeom et al. 2015], laser scanning or steering mechanisms [Jang et al. 2018, 2017], and operation with incoherent emitters [Moon et al. 2014] or amplitude-only SLMs [Gao et al. 2016]. Others focus on advancing algorithms for holographic image synthesis [Chakravarthula et al. 2019; Chen and Chu 2015; Maimone et al. 2017; Padmanaban et al. 2019; Shi et al. 2017]. Current AR/VR systems have limited resolution and lack focus cues and vision correction capability—the unique capabilities of holographic near-eye displays could address these challenges. Alternative technologies, such as near-eye light field displays [Hua and Javidi 2014; Huang et al. 2015; Lanman and Luebke 2013], also promise some of these benefits, but the resolution of these types of displays is fundamentally limited by diffraction. Holographic displays, on the other hand, utilize diffraction and interference to surpass 3D resolution limits of conventional displays.

Computer-generated Holography. Several algorithmic approaches have been explored to compute SLM phase patterns that optically produce a desired intensity distribution. Point-based methods are among the most popular algorithms. Here, the target scene is represented as a collection of points that are all propagated to the SLM plane and digitally interfered with a reference beam. Popular models for numerically propagating wave fields include the angular spectrum method and Kirchhoff or Fresnel diffraction [Goodman 2005]. To enforce the phase-only constraints imposed by current SLMs, direct methods use phase coding [Hsueh and Sawchuk 1978; Lee 1970; Maimone et al. 2017] to approximate the complex-valued wave field on the SLM with a phase-only field. To achieve the same goal, iterative methods [Chakravarthula et al. 2019; Dorsch et al. 1994; Fienup 1982; Gerchberg 1972; Peng et al. 2017] use optimization approaches based on phase retrieval [Shechtman et al. 2015]. Typically, direct methods are faster than iterative approaches but offer lower image quality or reduced brightness (Fig. 2, A).

Although most point-based methods do not model occlusions, depth discontinuities, or view-dependent lighting and shading effects of a scene, more sophisticated wave propagation approaches have addressed this problem using polygon [Chen and Wilkinson 2009; Matsushima and Nakahara 2009], light ray [Wakunami et al. 2013; Zhang et al. 2011], or layer [Zhang et al. 2017] primitives (see Park [2017] for a survey). Alternatively, holographic stereograms convert light fields into holograms and inherently encode depth- and view-dependent effects [Benton 1983; Kang et al. 2008; Lucente and Galyean 1995; Padmanaban et al. 2019; Yaras et al. 2010; Zhang and Levoy 2009; Ziegler et al. 2007].

Our approach is different from these methods in that it uses CITL optimization and wave propagation model calibration approaches that achieve unprecedented image quality for our experimental holographic display. The proposed model could potentially be used with many CGH algorithms; we demonstrate it in conjunction with a simple SGD solver. Gradient descent–type algorithms have been explored for phase retrieval [Chen et al. 2019], but to our knowledge we are the first to demonstrate that this simple algorithm achieves comparable or better image quality than other iterative computer-generated holographic display algorithms. This does not necessarily imply that SGD is the best or most elegant approach to holographic image synthesis, but that this trivial algorithm performs remarkably well and that it provides an intuitive and flexible platform to develop more advanced concepts on, such as the proposed CITL techniques.

Note that we are not the first to propose camera-based hardware calibration. Tseng et al. [2019], for example, recently described a technique that allows for non-differentiable hyperparameters of a camera's image processing pipeline to be estimated via a differentiable proxy using CITL training. However, their application, model, training procedure, and overall goals are all different from ours. Generally speaking, our CITL approaches closely follow a concept known as hardware-in-the-loop simulation, which is commonly applied to validate simulations of complex systems across engineering disciplines.

Holography and Deep Learning. Deep learning has recently become an active area of research in the computational optics community. For example, neural networks have been used in lensless holographic microscopy to help solve phase-retrieval problems [Rivenson et al. 2018; Sinha et al. 2017]. Recent surveys on the use of deep learning in holographic and computational imaging discusses these and other related techniques in detail [Barbastathis et al. 2019; Rivenson et al. 2019]. Many of these approaches learn wave propagation operators of an imaging system as black-box neural networks from a large number of training pairs comprising phase patterns and the corresponding intensity at some distance. Our CITL calibration technique is different in that it learns an interpretable model of the optical wave propagation of a holographic display. Some of the parameters we learn, such as phase nonlinearity of an SLM and source intensity of the laser, are unique to display applications. Moreover, our calibration approach is adaptive in that it uses a camera in the loop for the training phase. This is a fundamentally different optimization approach than the universal-function-approximator approach of fitting some black-box model to a training set of image pairs.

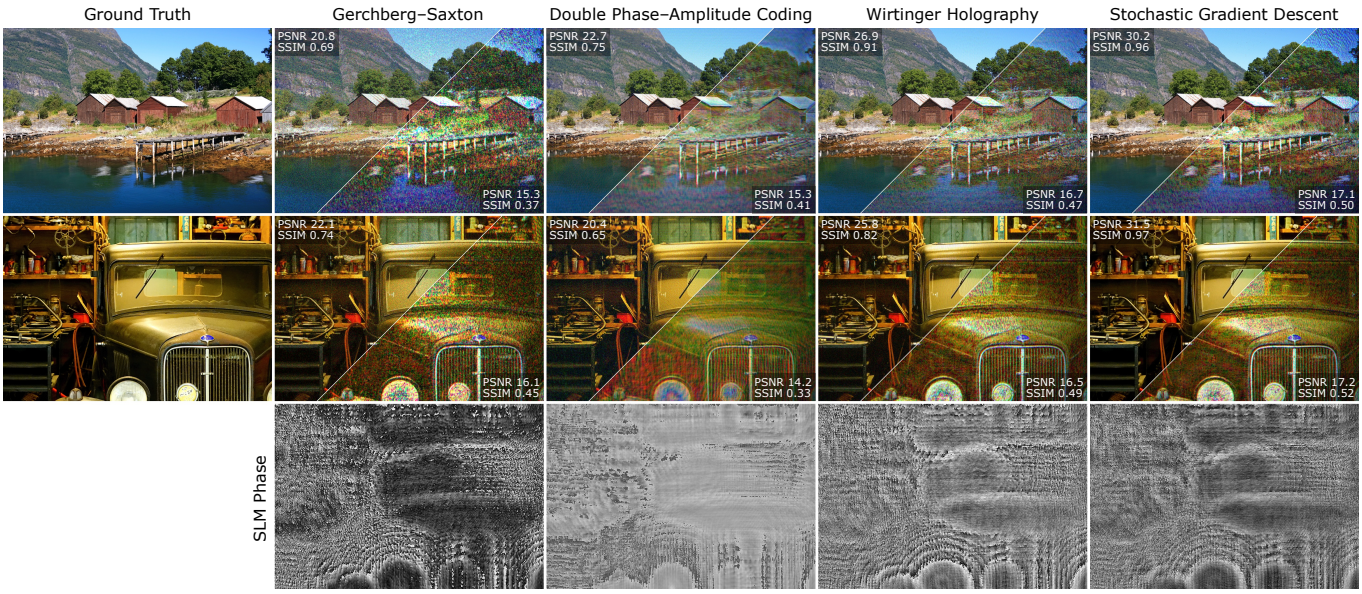

Fig. 3. Comparisons of several CGH algorithms in simulation. For both example scenes, we show the simulated result assuming a perfectly matched wave propagation model between phase optimization and image reconstruction in the upper left sub-images. For fair comparison, the mean amplitude of all results is scaled to match the mean amplitude of the target image. We see that the SGD approach significantly reduces noise artifacts, especially in large uniform areas. The lower right sub-images show simulated results with a small amount of model mismatch. Here, the SLM phase patterns are optimized with an ideal wave propagation model but the simulation introduces a small amount of optical aberration, phase nonlinearity, and source intensity variation, corresponding to our display prototype. All methods fail to produce high-quality results. We also show the optimized SLM phase patterns for the second scene in row 3.

Deep learning has also been proposed for computer-generated holography display applications. For example, Horisaki et al. [2018] recently proposed training a simple U-Net [Ronneberger et al. 2015] on phase–intensity image pairs and then predicting the SLM phase pattern from a target intensity during inference. This network architecture and training procedure are similar to the holographic imaging approaches discussed above. As demonstrated in Section 6, this universal-function-approximator approach does not work well for CGH applications. Our network generator, HoloNet, consistently achieves superior results to previous deep learning approaches for holographic display applications.

3 PROBLEM FORMULATION AND MOTIVATION

The holographic image synthesis problem we aim to solve is as follows. In a holographic display, a complex-valued wave field u_{src} generated by a coherent source is incident on a phase-only SLM. This source field could, for example, be modeled by a plane wave, a spherical wave, or a Gaussian beam. The phase of the source field is then delayed in a per-SLM-pixel manner by phase ϕ. The field continues to propagate in free space (or through some optical elements) to the target plane, where a user or a detector observe the intensity of the field. Here, the optical propagation is described by a function f. Although we do not know what f is exactly, we assume that we have a reasonably good model $\widehat{f}$ for it. For example, optical propagation from SLM to target plane can be modeled by a free-space wave propagation operator, such as the angular spectrum method [Goodman 2005; Matsushima and Shimobaba 2009],

$$\widehat{f}(\phi) = \iint \mathcal{F}\left(e^{i\phi(x,y)} u_{\text{src}}(x,y)\right) \mathcal{H}(f_x, f_y)\, e^{i2\pi(f_x x + f_y y)} df_x df_y,$$

$$\mathcal{H}(f_x, f_y) = \begin{cases} e^{i\frac{2\pi}{\lambda}\sqrt{1-(\lambda f_x)^2-(\lambda f_y)^2}\, z}, & \text{if } \sqrt{f_x^2 + f_y^2} < \frac{1}{\lambda} \\ 0 & \text{otherwise} \end{cases}, \quad (1)$$

where λ is the wavelength, f_x, f_y are spatial frequencies, z is the distance between SLM and target plane, and $\mathcal{F}(\cdot)$ denotes the Fourier transform.

In practice, the phase would be discretized as $\phi \in \mathbb{R}^{M\times N}$, where $M \times N$ is the SLM resolution, and $f, \widehat{f} : \mathbb{C}^{M\times N} \to \mathbb{C}^{M\times N}$. Given this model $\widehat{f}$, we can apply any phase-retrieval algorithm, such as Gerchberg–Saxton (GS) [Gerchberg 1972] or Fienup's method [Fienup 1982], to find the phase values ϕ that best approximate the target image. Generally speaking, this can be formulated as solving an optimization problem of the form

$$\underset{\phi}{\text{minimize}}\ \mathcal{L}\left(s \cdot |\widehat{f}(\phi)|, a_{\text{target}}\right), \quad (2)$$

where $\mathcal{L}$ is some loss function, $a_{\text{target}} \in \mathbb{R}^{M\times N}$ is the target amplitude, and s is a fixed or learnable scale factor that accounts for the fact that the implementation of the wave propagation operator can output values in a different range compared with the target. Please refer to Appendix A for a more detailed discussion on the relationship between amplitude, linear intensity, and gamma-corrected intensity and why it makes sense to use the amplitude of the target image in the loss function.

3.1 Computer-generated Holography via SGD

To solve Equation 2, many algorithms have been proposed. The iterative Gerchberg–Saxton algorithm (GS) is the classic approach, but many others find ϕ via *phase retrieval* [Shechtman et al. 2015]. The challenge with Equation 2 is that it is non-convex, so there may be infinitely many solutions that achieve a low residual or loss. Whereas the phase-retrieval community is concerned with finding the one phase function ϕ that matches some physical phase-delaying object, CGH is a much easier problem because we can pick any of the infinitely many possible solutions that achieve a small loss. After all, each one of them gives us the same intensity on the target plane.

Therefore, we argue that advanced mathematical concepts developed in the phase-retrieval community, such as Wirtinger Flow, may not be necessary for CGH to succeed. We implement the forward image formation (Eq. 1 and loss function evaluation) in PyTorch, let PyTorch's autodiff capabilities keep track of the gradients, and optimize the objective function using some variant of stochastic gradient descent (SGD), such as Adam [Kingma and Ba 2014]. Surprisingly, we find that this trivial approach achieves the best image quality compared with other iterative methods, such as Gerchberg–Saxton and Wirtinger Holography, and direct methods, such as double phase–amplitude coding (DPAC) [Maimone et al. 2017]. This experiment is shown in Figures 2 and 3; additional details are found in the Supplemental Material.

The insight that SGD can be used without modification for CGH optimization is valuable for multiple reasons. First, it is trivial to implement and execute on a graphics processing unit (GPU). Second, it is easy to use advanced loss functions $\mathcal{L}$ that apply perceptually motivated, scale-invariant, or other error metrics. Finally, as we will show in the following, this approach allows us to account for the model mismatch between the optical wave propagation of the display f and its simulated model $\widehat{f}$ (Eq. 1); moreover, it allows us to optimize an interpretable model of the optical wave propagation itself.

4 CAMERA-IN-THE-LOOP PHASE OPTIMIZATION WITH OPTICAL WAVE PROPAGATION

As shown in Figures 2 and 3, a model mismatch between the simulated wave propagation used for optimizing phase patterns and that observed with a physical display is one of the primary sources of image degradation in holographic displays. In this section, we introduce the idea of camera-in-the-loop (CITL) holographic image synthesis to mitigate this model mismatch. The experiments shown in this section motivate the promise of CITL optimization strategies to generate holographic images of very high quality.

Assume we use an autodiff approach to optimize SLM phase patterns, as discussed in Section 3. For this purpose, we start with some initial guess or previous estimate $\phi^{(k-1)}$, do a forward pass through the model and loss function $\mathcal{L}$, and then backpropagate the error using the gradients $\frac{\partial \mathcal{L}}{\partial \phi}$ to find the next step $\phi^{(k)}$ at iteration k. Typically, this procedure would use the simulated wave propagation model $\widehat{f}$ and one would hope that the physical display is calibrated well enough to match this model. In our CITL experiment, we would like to lift this assumption and aim at using the optical wave propagation f itself for both the forward pass and also its gradients in the error backpropagation. If this were possible, we could eliminate any possible model mismatch and optimize the observed image quality.

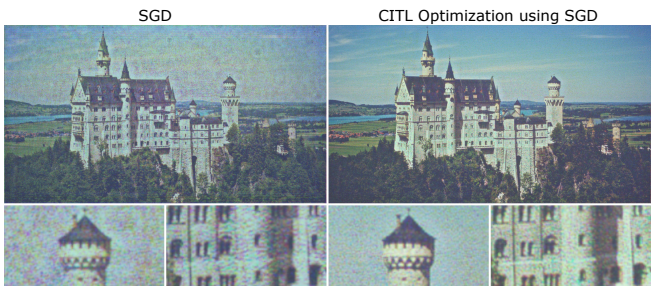

Fig. 4. Captured results of CITL optimization. *Left:* image quality achieved with a stochastic gradient descent solver assuming an idealized wave propagation model. Slight differences between the model used for optimizing the SLM phase pattern and the optical wave propagation of the display result in image degradation. *Right:* SGD-based CITL optimization can significantly reduce these artifacts by utilizing the optical wave propagation directly during the optimization.

The forward pass can be easily implemented with the physical display by displaying $\phi^{(k-1)}$ on the SLM and capturing the resulting intensity at the target plane with a camera. We then pass this captured image into the loss function. Unfortunately, we do not have access to the gradients of the physical model[1], so we cannot easily implement the backpropagation pass. Yet, we can approximate these gradients using the model proxy $\widehat{f}$ as

$$\underbrace{\frac{\partial \mathcal{L}}{\partial \phi} = \frac{\partial \mathcal{L}}{\partial f} \cdot \frac{\partial f}{\partial \phi}}_{\text{inaccessible}} \approx \underbrace{\frac{\partial \mathcal{L}}{\partial f} \cdot \frac{\partial \widehat{f}}{\partial \phi}}_{\text{accessible}}. \tag{3}$$

Note that $\frac{\partial \mathcal{L}}{\partial f}$ can be readily computed by passing the captured image into the loss function with autodiff enabled. To calculate the partial derivatives of the proxy model $\frac{\partial \widehat{f}}{\partial \phi}$, we need to computationally feed $\phi^{(k-1)}$ into $\widehat{f}$, and simultaneously show it on the SLM to capture $|f\left(\phi^{(k-1)}\right)|^2$, while autodiff keeps track of the gradients $\frac{\partial \widehat{f}}{\partial \phi}$. A gradient descent–type solver would then iterate as

$$\begin{aligned}\phi^{(k)} &\leftarrow \phi^{(k-1)} - \alpha \left(\frac{\partial \mathcal{L}}{\partial \phi}\right)^T \mathcal{L}\left(s \cdot \left|f\left(\phi^{(k-1)}\right)\right|, a_{\text{target}}\right) \\ &\approx \phi^{(k-1)} - \alpha \left(\frac{\partial \mathcal{L}}{\partial f} \cdot \frac{\partial \widehat{f}}{\partial \phi}\right)^T \mathcal{L}\left(s \cdot \left|f\left(\phi^{(k-1)}\right)\right|, a_{\text{target}}\right),\end{aligned} \tag{4}$$

where α is the learning rate. We call this procedure CITL hologram optimization. Conventional iterative phase-retrieval methods make the assumption that the optical wave propagation f and simulated proxy model $\widehat{f}$ closely match, and they optimize the phase pattern exclusively using the proxy. The CITL optimization approach uses the physical model for the forward pass and also for part of the backpropagation. The only approximation this approach makes is

[1]We assume that a finite difference approach to capturing the gradients of the physical model using $M \times N + 1$ images in every iteration is computationally infeasible.

in using the gradients of the proxy model to propagate the error back into the phase pattern. Therefore, fewer assumptions have to be made about the optical wave propagation. Indeed, our CITL optimization only makes the assumption that the gradients of the optical and the simulated wave propagation match reasonably well.

Figure 4 demonstrates this idea with experimental results captured using our prototype (see Appendix B and Supplement). On the left, we see the result of running SGD for 500 iterations and displaying the resulting phase pattern on the SLM. The slight model mismatch between the optical and simulated wave propagation models results in a noisy and otherwise degraded image. The artifacts are particularly visible in large uniform areas of the image, such as the sky. Using the described CITL optimization procedure, these artifacts can be mitigated and a significantly improved image quality is achieved with the same number of iterations. Additional captured results of this approach are shown in Figures 1, 6, and S3.

Although the approach discussed in this section achieves the best experimental image quality of any CGH algorithm to date, it has several limitations. Among these are the facts that this type of calibration is specific to a particular display, that a camera and hardware-in-the-loop optimization is required for each target image, and that the optimization takes several minutes for each target image. These shortcomings motivate the techniques introduced in the following sections. A more in-depth discussion of limitations can be found in Section 8.

5 HOLOGRAPHIC IMAGE SYNTHESIS WITH CAMERA-IN-THE-LOOP MODEL TRAINING

In this section, we explore another variant of CITL holography. Here, we split the optimization procedure into a training phase and an inference phase. A camera is only required for the training phase but not for inference. The benefit of this approach is that there is a one-time calibration procedure that requires the camera in the physical system, but once this stage is concluded, arbitrary holograms can be synthesized and displayed without requiring a camera. Moreover, rather than simply approximating the gradients of the physical model with those of a proxy, as done in the previous section, we now learn an interpretable parameterization of the optical wave propagation $\widehat{f}_\theta$ via a CITL training procedure. This CITL-calibrated model can be interpreted as a fully automatic calibration routine that allows a significantly more sophisticated wave propagation model to be estimated, compared to existing models using simple lookup tables.

5.1 A Parameterized Wave Propagation Model

The CITL training phase, outlined by Algorithm 1, is inspired by the training phase of neural networks. Given a dataset of J images, we calculate a loss of the result achieved with the current set of model parameters θ and backpropagate the error into the model parameters to improve them step-by-step over many iterations. This is a stochastic gradient descent (SGD) approach to solving the optimization problem

$$\underset{\{\phi_j,\theta\}}{\text{minimize}} \sum_{j=1}^{J} \mathcal{L}\left(s \cdot |\widehat{f}_\theta\left(\phi_j\right)|, a_{\text{target}_j}\right). \tag{5}$$

A naïve choice for the proxy model $\widehat{f}_\theta$ is a generic (convolutional) neural network. This may be feasible, because a network with sufficiently high capacity can approximate any function. However, such a universal-function-approximator approach is neither elegant nor efficient (i.e., it may require a lot of time to be trained) and it also does not allow for any insights on the actual physical display to be gained (see Secs. 6 and S1 for a direct comparison). Therefore, we use an interpretable, physically inspired model. Specifically, we parameterize four aspects of the wave propagation: content-independent source and target fields, optical aberrations, SLM phase nonlinearities, and content-dependent undiffracted light.

ALGORITHM 1: CITL Holography – Training

θ : model parameters
J : number of training images, ≈ 800
E : number of training epochs, ≈ 15
M : size of minibatch, ≈ 2
`camera_p(·)` : camera capture (raw mode) + homography
`loss_bp(·)` : backpropagation through loss function
`model_p(·; θ)` : propagation through parameterized model
`model_bp(·; θ)` : backpropagation through parameterized model
→ pre-compute all J phase maps via SGD to form a pool
foreach e **in** $1 \ldots E$ **do**
  foreach j **in** $1 \ldots J/M$ **do**
    $a_{\text{target}} \leftarrow$ load M target images, invert sRGB gamma, $\sqrt{\cdot}$
    $\phi \leftarrow$ load M corresponding phase maps from pool
    $a_{\text{model}} \leftarrow |\texttt{model\_p}(\phi; \theta)|$
    $\{\phi, s_1\} \leftarrow \texttt{model\_bp}(\texttt{loss\_bp}(\mathcal{L}(s_1 \cdot a_{\text{model}}, a_{\text{target}})); \theta)$
    $a_{\text{camera}} \leftarrow \sqrt{\texttt{camera\_p}(\phi)}$ ▷ capture M camera images
    $a_{\text{model}} \leftarrow |\texttt{model\_p}(\phi; \theta)|$
    $\{\theta, s_2\} \leftarrow \texttt{model\_bp}(\texttt{loss\_bp}(\mathcal{L}(s_2 \cdot a_{\text{model}}, a_{\text{camera}})); \theta)$
    → save updated phase maps ϕ to the pool
  end
end
return θ

5.1.1 Content-independent Source and Target Field Variations. In optics, a collimated source is typically modeled as a Gaussian beam, i.e., with a Gaussian describing its intensity variation. We adopt this approach to model possible laser source intensity variation over the SLM (i.e., the source plane) caused by the optical elements in the illumination path. For this purpose, we use a Gaussian mixture model $i_{src} = \sum_g w_g \mathcal{G}(\mu_g, \sigma_g)$. Here, w_g are the weights of each Gaussian, μ_g are their coordinate centers in x and y, and σ_g represents the standard deviation of Gaussian g. Typically, we use a total of $g = 3$ Gaussian functions. The source intensity i_{src} or amplitude $a_{src} = \sqrt{i_{src}}$ distribution is modeled for each channel separately. Additionally, we include a learnable complex-valued field u_t on the target image plane in our model. This accounts for content-independent contributions of undiffracted light or higher diffraction orders to the target.

5.1.2 Modeling Optical Propagation with Aberrations. Although the free-space propagation model (Eq. 1) is theoretically correct, in practice we observe optical aberrations in a physical holographic

display. These are caused by the many optical elements in the path, including beam splitters, lenses, and even the cover glass of the SLM. To model the phase distortions that these aberrations add to the wave field, we use Zernike polynomials to additively correct the phase of the propagation operator's transfer function. These polynomials are described by $\phi_z = \sum_k \beta_k Z_k$, where Z_k is the k^{th} Zernike basis function in Noll notation and β_k is the corresponding coefficient [Born and Wolf 1959]. The set of model parameters θ thus includes a finite set of Zernike coefficients β_k.

5.1.3 Modeling Phase Nonlinearities. With a phase-only SLM, we control the per-pixel phase delay induced to the source field by adjusting the voltages of each pixel. However, there may be a nonlinear mapping between voltages and phase, we can only control the phase delay for a limited range of values (typically $[-\pi, \pi)$) and, just like for any other display, these voltages are quantized to a few bits. The physical properties of the liquid crystal cells over the SLM along with nonuniformities of the backplane electronics may further result in a voltage-to-phase mapping that varies over the SLM. Note that it is standard practice for a phase-only SLM manufacturer to either provide a pre-calibrated voltage-to-phase mapping as a single lookup table per color channel or give users the means to calibrate their own lookup table. With our prototype, we followed the procedure outlined by the manufacturer to calibrate a lookup table for each color channel and upload these on the SLM driver. The models we describe in the following have the option to further refine the pre-calibrated global voltage-to-phase mapping or to calibrate a spatially varying mapping. To the best of our knowledge, we propose the first approach to modeling a spatially varying voltage-to-phase mapping in a differentiable manner and provide fully automatic calibration techniques.

First, assume that the nonlinear voltage-to-phase mapping is the same for each pixel on the SLM. In this case, we can model it using a single multilayer perceptron $\text{MLP}_\phi : \mathbb{R}_{[0,1]} \rightarrow \mathbb{R}_{[-\pi,\pi)}$, where we assume the range of feasibly voltage values to be normalized to the range $[0, 1]$. Given an image representing SLM pixel voltages, we apply MLP_ϕ to the input image using 1×1 convolutions. This is a differentiable variant of lookup tables used by most commercial SLMs.

To model a spatially varying mapping, we extend MLP_ϕ by concatenating an additional latent code vector $c\,(x, y)$ to its input. The resulting mapping function $\text{MLP}_{\phi_c} : \mathbb{R}_{[0,1]} \times \mathbb{R}^C \rightarrow \mathbb{R}_{[-\pi,\pi)}$ thus allows the same multilayer perceptron to provide a slightly different mapping function via a latent code vector that comprises C floating point numbers per pixel. We set C to 2. Such a conditioning-by-concatenation approach is standard practice in machine learning, but we are the first to propose it for modeling spatially varying voltage-to-phase mappings for holographic display applications. The weights, biases, and spatially varying code vectors are all trained end-to-end with the CITL procedure described in Section 5.2.

5.1.4 Content-dependent Undiffracted Light. With our SLM, we observe a non-negligible amount of content-dependent undiffracted light on the target plane. While undiffracted light is partly expected due to the imperfect diffraction efficiency of the SLM, one would typically expect this term to be largely independent of the displayed SLM pattern, which would be adequately modeled by source or target fields. To model the content-dependent nature of parts of this term, we connect the phase pattern of the SLM, ϕ, directly to the target plane with a small convolutional neural network $\text{CNN} : \mathbb{R}^{M \times N} \rightarrow \mathbb{C}^{M \times N}$.

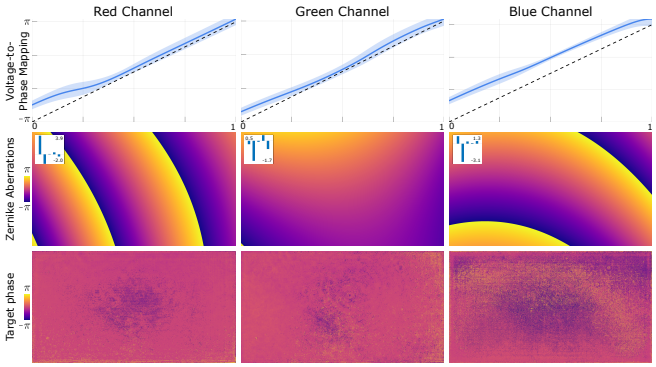

Fig. 5. Visualization of calibrated model parameters. *Row 1:* The spatially varying voltage-to-phase mapping is close to linear, although the red and blue channels show slightly different behavior in the lower and upper parts, respectively. We show the mean curves in bold and the variation over the SLM pixels in light blue. *Row 2:* The Zernike aberrations reveal a small amount of phase curvature in all channels as well as some tilt of the red and blue channels. *Row 3:* The content-independent phase distribution on the target plane contains high- and low-frequency variation.

ALGORITHM 2: CITL Holography – Inference

θ : model parameters, calibrated via Algorithm 1
foreach k **in** $1 \ldots K$ **do**
  $a_{\text{model}} \leftarrow |\texttt{model\_p}(\phi; \theta)|$
  $\{\phi, s\} \leftarrow \texttt{model\_bp}(\texttt{loss\_bp}(\mathcal{L}(s \cdot a_{\text{model}}, a_{\text{target}})); \theta)$
end
return ϕ

5.1.5 Parameterized Wave Propagation Model. The image formation defined by our parameterized wave propagation model of the holographic display is thus

$$\widehat{f_\theta}\,(\phi) = u_t(x, y) + \text{CNN}\,(\phi) + \iint \mathcal{F}\left(a_{src}\,(x, y)\, e^{i\text{MLP}_{\phi_c}(\phi, c(x,y))}\right) \cdot \mathcal{H}\left(f_x, f_y\right) e^{i\phi_z} e^{i2\pi(f_x x + f_y y)} df_x df_y, \quad (6)$$

where the model parameters θ include all weights and bias terms of MLP_{ϕ_c}, CNN, one latent code c for each SLM pixel that is concatenated with the input of MLP_{ϕ_c}, the means, centers, and standard deviations of the Gaussians modeling a_{src}, the target field u_t, and the Zernike coefficients β_k defining ϕ_z.

5.2 CITL Wave Propagation Model Training

To train the model parameters θ, we first pre-compute all phase maps for the entire training set using SGD with the idealized model (Sec. 3) to form a pool of phase maps. Our training set comprises the 800 1080p images of the DIV2K dataset [Agustsson and Timofte

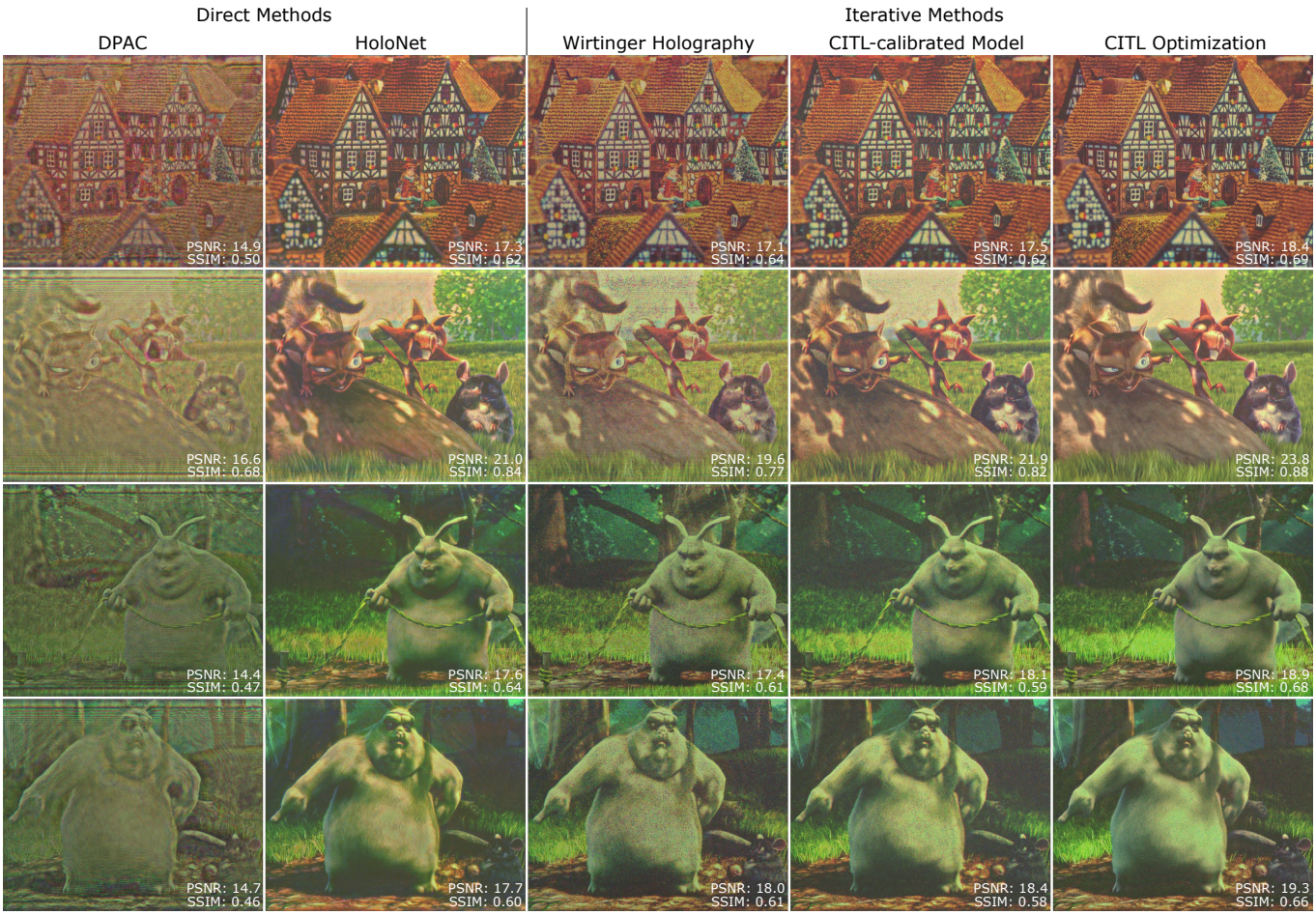

Fig. 6. Captured results. Among the real-time CGH algorithms (left), double phase–amplitude coding exhibits low contrast in addition to other artifacts; the proposed HoloNet approach shows significantly improved image quality. Iterative methods are slower (right), but can improve the image quality compared to real-time methods. Still, Wirtinger Holography is generally noisy and exhibits limited contrast. The proposed variants of SGD using a CITL-calibrated model and CITL optimization achieve the best results—especially the latter, which noticeably removes artifacts and speckle noise while preserving image contrast and detail.

2017]. We train the model using the hyperparameters and procedures outlined in Algorithm 1 with the Adam optimizer. Specifically, during each loop of the training procedure, we progressively update the initial phase maps using the current model and the model parameters using the camera images. The updated phase maps are saved to the pool after processing each minibatch. All of our source code is available on the project website at www.computationalimaging.org.

5.2.1 Ablation Study. We run an ablation study to highlight the contributions of individual model parts for the green channel, training all model variants for 1 epoch. Results are summarized in Table 1 and additional qualitative and quantitative results are shown in the Supplement. The model versions we evaluate include: the idealized model optimized via SGD (column 1); adding source amplitude (column 2); adding source amplitude and the Zernike polynomials (column 3); adding source amplitude, Zernike polynomials, and a global SLM lookup table (column 4); adding source amplitude, Zernike polynomials, and a spatially varying SLM lookup table (column 5); adding source amplitude, Zernike polynomials, the spatially varying SLM lookup table, a target field, and the CNN modeling the content-dependent undiffracted light (column 6). The average peak signal-to-noise-ratio (PSNR) for all 8 test images shows that each of the model parts improves the observed image quality and the final model achieves the best quality.

Table 1. Ablation study with captured data showing average PSNR and SSIM for 8 test images.

	SGD	$+a_{src}$	$+a_{src}$ $+\phi_z$	$+a_{src}+\phi_z$ $+\text{MLP}_{\phi}$	$+a_{src}+\phi_z$ $+\text{MLP}_{\phi_c}$	$+a_{src}+\phi_z+u_t$ $+\text{MLP}_{\phi_c}+\text{CNN}$
PSNR	18.1	18.2	18.2	19.1	19.2	**19.5**
SSIM	0.59	0.59	0.59	0.56	0.57	**0.60**

5.2.2 Interpreting the Parameters of the Trained Model. We visualize several of the calibrated model parameters in Figure 5. Interpreting these parameters reveals interesting insights about the physical display. For example, the voltage-to-phase mappings for all three color channels vary by a small amount over the SLM and the red and blue channels show some nonlinear behavior (row 1). The Zernike terms shows some field curvature, likely caused by optical aberrations, and the red and blue channels are tilted (row 2). This indicates that the laser sources are not perfectly aligned and propagate in slightly different directions. Finally, the phase of the target field shows speckle-like high-frequency structures (row 3). Additional model parameters are shown in the supplement.

5.3 Hologram Synthesis with the Trained Model

With a calibrated display model, the inference phase (i.e., synthesizing a hologram) is straightforward. For this purpose, variants of stochastic gradient descent can be employed to optimize an SLM phase for a given target image by implementing the forward pass of the image formation with the model-based wave propagation in a suitable language, e.g., PyTorch or TensorFlow, and having autodiff keep track of the gradients. As discussed in Section 3 and outlined in Algorithm 2, this simple procedure is an efficient way to synthesize holograms with either a vanilla free-space propagation operator or the proposed model-based wave propagation.

Figures 1, 4, 6, and S3 show detailed comparisons of captured results using the CGH algorithms we consider in this paper. Compared to other iterative methods, including Gerchberg–Saxton and Wirtinger Holography, the proposed CITL-calibrated model approach improves image contrast, detail, and speckle noise. Still, the direct CITL optimization approach introduced in Section 3.1 achieves the best quality of all methods we evaluate. This indicates that requirement of generalizing across images for the model comes at the cost of slight image degradation compared to overfitting it to a single image.

6 REAL-TIME HOLOGRAPHIC IMAGE SYNTHESIS

In this section, we introduce HoloNet, a neural network architecture that achieves real-time frame rates for high-quality 2D holographic image synthesis. As illustrated in Figure 7, HoloNet takes an sRGB image converted into amplitude at the target plane as input. Given the target amplitude, a *target-phase generator* subnetwork predicts a phase distribution at the target plane. This predicted phase distribution is combined with the target amplitude to create a complex-valued wave field. The content-independent target-plane variation is then subtracted from this wave field. The resulting field is propagated to the SLM plane via the Zernike-compensated wave propagation model, $\widehat{f}^{-1}$. The Zernike compensation is applied to the transfer function of the propagation operator, similarly to Section 5.1.2 for $\widehat{f}_\theta$, but by subtracting the Zernike polynomial to account for the reverse propagation direction. At the SLM plane, we compensate for the source intensity—also calibrated as part of $\widehat{f}_\theta$—by dividing it pointwise from the propagated field. We then concatenate the source-intensity-compensated amplitude and phase with the latent code vector, $c(x, y)$, before passing it all to the *phase encoder* subnetwork. This subnetwork converts the concatenated

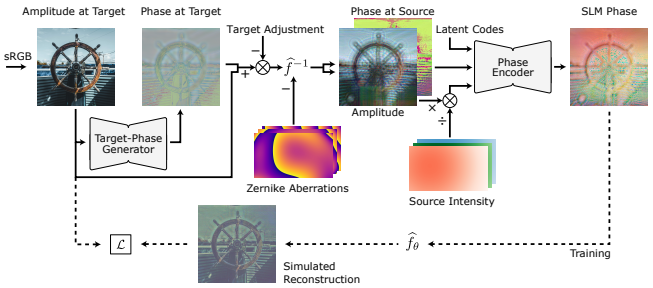

Fig. 7. Overview of HoloNet. A target sRGB image is converted first to amplitude values, then passed to a target-phase-generator subnetwork, which predicts a phase on the target plane. The predicted phase and input amplitudes are combined to make a complex field. The complex field is then adjusted by subtracting a content-independent target-field adjustment. The adjusted field is propagated via a Zernike-compensated propagation operator (as in Sec. 5.1.2, except the Zernike phases are subtracted) to the SLM plane. At the SLM plane, we divide by the source intensity to compensate for it. The resulting channels are concatenated with the latent codes and then go through the phase-encoder network. This network produces a final phase-only representation to be displayed on the SLM. During the training phase, the phase-only representation would be propagated back to the target plane with the proxy model $\widehat{f}_\theta$, where the loss can be calculated against the target amplitude. Note that while we show RGB inputs above, in reality, a separate network is trained for each color channel.

field into a phase-only representation. During inference, this phase pattern is shown on the SLM to produce the final image. During training, this SLM phase pattern is propagated to the target plane via the proxy model, $\widehat{f}_\theta$, and there compared to the target image.

The content-dependent undiffracted light subnetwork (Sec. 5.1.4) from $\widehat{f}_\theta$, while not explicitly included as part of HoloNet, is accounted for via the learning procedure. Since $\widehat{f}_\theta$ is used when calculating the target plane output for the loss function, HoloNet's target-phase-generator and phase-encoder subnetworks must learn to incorporate equivalent content-dependent terms to properly minimize the training loss.

Network Architecture. The target-phase-generator and phase-encoder subnetworks are both implemented using similar U-Net architectures [Ronneberger et al. 2015]. Both networks have four downsampling (and corresponding upsampling) blocks, with the initial feature count after the input layer set to 16 features. The down blocks have a leaky ReLU (0.2 negative slope) nonlinearity, and the up blocks have a ReLU nonlinearity. All blocks use batch normalization and have two convolutional layers each. Finally, the output nonlinearity is a hard tanh() function with limits of $\pm\pi$. The target-phase generator has 1 channel for the input (amplitude) and 1 for the output (phase). The phase encoder has 4 input channels (amplitude, phase, and latent code vector) and 1 for the output (phase).

Three separate networks are trained, one for each color channel, due to the need to specify different wavelengths in the wave propagation model, $\widehat{f}$.

Loss Function. We use a combination of ℓ_2 and perceptual losses [Johnson et al. 2016a] to train HoloNet:

$$\mathcal{L}_{\text{percep}}\left(s \cdot |\widehat{f_\theta}(\phi)|, a_{\text{target}}\right) = \|s \cdot |\widehat{f_\theta}(\phi)| - a_{\text{target}}\|_2^2 + \lambda_{\text{p}} \sum_{l=1}^{L} \|\text{P}_l(s \cdot |\widehat{f_\theta}(\phi)|) - \text{P}_l(a_{\text{target}})\|_2^2, \quad (7)$$

where $\text{P}(\cdot)$ represents a transform to a perceptual feature space and $\lambda_{\text{p}} = 0.025$ is the relative weight on the perceptual loss component. Our chosen feature space comprises the output of each layer l of VGG [Simonyan and Zisserman 2014] for the first $L = 5$ layers. The scale factor s is set to 0.95.

Training Details. The network is trained for 5 epochs using the Adam optimizer. The training images are from the DIV2K dataset [Agustsson and Timofte 2017], augmented with vertical and horizontal flipping (800 images × 4 per epoch). The images are preprocessed such that they occupy a 1,600 × 880 px region, padded with zeros out to 1080p. The loss is computed only on the unpadded region. We apply this transform to match the rectified region of our captured images.

Results. We evaluate HoloNet in simulation and experiment. Figure 2 (A) shows simulated quantitative results, highlighting the fast runtime of about 40 frames per second for full-color 1080p images as well as the high image quality of about 30 dB peak signal-to-noise ratio. Note that a comparable quality is only achieved by the previous best iterative method, Wirtinger Holography, after about 10 seconds. These simulated results all use a perfect wave propagation model (i.e., Eq. 1) without optical aberrations or phase nonlinearities.

We also show results captured with our prototype holographic display in Figures 1, 6, and S3. Here, HoloNet is trained with the procedure outlined above but including the CITL-calibrated model parameters discussed in Section 5. As seen on the left of Figure 6, HoloNet achieves a significantly better image quality than the best existing real-time method, DPAC. As expected from our simulations, the quality achieved by HoloNet and WH are comparable even for these captured experiments, although the artifacts observed for both methods are slightly different. WH seems noisier while HoloNet exhibits a slight amount of color and tone mismatch.

Comparison to Previous Work. Previous attempts at deep-learned CGH have used "universal function approximators" such as U-Nets [Horisaki et al. 2018]. To serve as a baseline representing these earlier methods, we train a U-Net with similar settings to our target-phase-generator and phase-encoder subnetworks, but with 10 down- and upsampling blocks. This architecture follows the spirit of Horisaki et al., but our U-Net variant has significantly higher capacity than their specific implementation to allow for a fair comparison with HoloNet and the other CGH algorithms. As seen in Figure 2 (A) and in the supplement, the U-Net approach is as fast as DPAC and HoloNet, but it cannot match the image quality of either. Note that all compared networks use the same idealized wave propagation model, are trained with the same loss function, and are trained to similar levels of relative convergence. While it is possible to add more capacity to the baseline U-Net, it already has 369,107,553 parameters—over 2 orders of magnitude more than HoloNet at 2,868,754. It is clear that the U-Net is not an efficient encoding method.

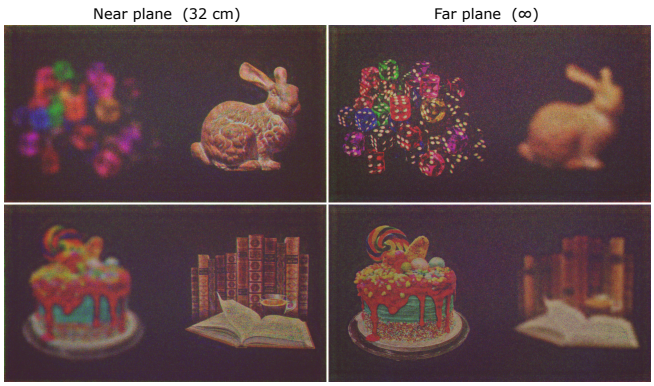

Fig. 8. Captured varifocal results, displayed at two different distances. The near plane corresponds to a 32 cm focusing distance for the camera, and the far plane corresponds to optical infinity.

7 TOWARDS 3D HOLOGRAPHIC DISPLAYS

In this section, we explore two approaches to extending the proposed algorithms to 3D holographic image presentation: a holographic varifocal and a multiplane display mode. Both of these display modes have the potential to mitigate the vergence–accommodation conflict as well as to optimize visual comfort and perceptual realism in VR/AR systems.

7.1 Holographic Varifocal Display Mode

Varifocal displays have been explored with conventional microdisplays for both VR and AR applications [Akşit et al. 2017; Chakravarthula et al. 2018; Dunn et al. 2017; Johnson et al. 2016b; Konrad et al. 2016; Laffont et al. 2018; Liu et al. 2008; Padmanaban et al. 2017; Shiwa et al. 1996]. Varifocal displays use eye tracking to determine the gaze depth of the user, i.e., the distance of the object the user fixates. The 2D focal plane of the near-eye display is then adjusted such that the magnified virtual image's distance matches the gaze depth. Adjusting the focal plane of a display can be achieved using either mechanical actuation or focus-tunable lenses.

As opposed to conventional microdisplay-based near-eye systems, a holographic display has inherent 3D display capabilities. Thus, holographic displays can achieve varifocal capabilities without the need for mechanical actuation or focus-tunable optics. And although running a holographic display in a varifocal mode only utilizes a subset of its capabilities, emulating a 3D display by adaptively shifting a 2D hologram is computationally more efficient than using a true 3D display mode [Maimone et al. 2017].

In Figure 8, we demonstrate our system's ability to operate at multiple distances using a varifocal display mode. Models for both distances (10.0 and 11.25 cm from the SLM, corresponding to infinity and around 32 cm from the camera, respectively) are trained and applied to generate the holograms. Defocus blur is computationally rendered in this example.

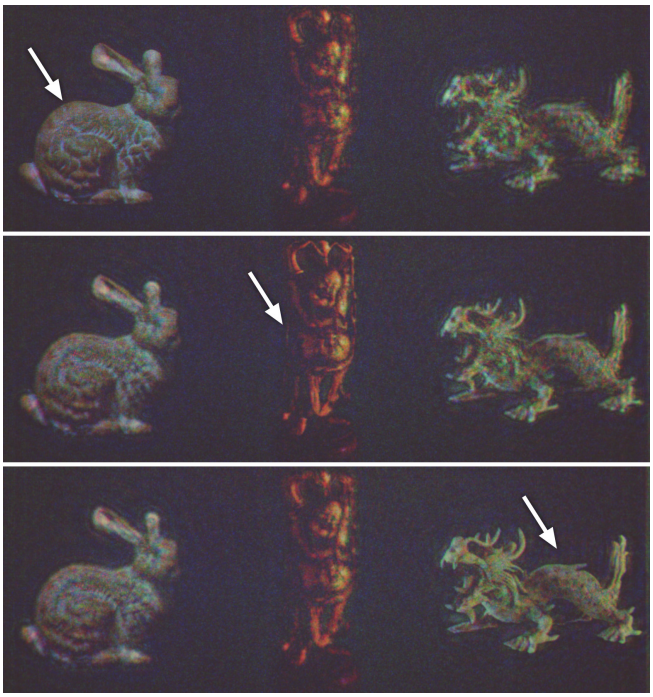

Fig. 9. Captured focal stack of a multiplane hologram. These results were optimized using the CITL-calibrated model where a single SLM phase pattern is optimized for all three distances simultaneously.

7.2 Holographic Multiplane Display Mode

Multiplane displays represent a true 3D display modality by presenting multiple additively overlaid image planes at different depths to the user. Significant research efforts have recently focused on hardware prototypes and layer decomposition algorithms for these types of displays [Akeley et al. 2004; Chang et al. 2018; Hu and Hua 2014; Liu et al. 2008; Llull et al. 2015; Love et al. 2009; Mercier et al. 2017; Narain et al. 2015; Rathinavel et al. 2018; Rolland et al. 2000]. Multiplane displays require either a single, high-speed SLM in conjunction with a focus-tunable lens or multiple optically overlaid displays. Again, holographic displays have the benefit of achieving the same display modality with a single SLM and without the need for focus-tunable optics.

In Figure 9, we show a captured focal stack of a holographic multiplane display. Similarly to the varifocal display mode, one may calibrate models for a few discrete distances, in this example three. Instead of optimizing a phase pattern for one selected depth, however, here we optimize a single phase pattern for all distances simultaneously. In this experiment, we place the far image (bunny) at 10.0 cm from the SLM and the near image (dragon) at 11.25 cm, corresponding to infinity and 32 cm of focus on the capturing camera, respectively. The middle image (Buddha) was placed halfway between the two at 10.75 cm.

Additional details and results for both varifocal and multiplane modes are included in the Supplement.

8 DISCUSSION

In summary, we make several important insights and contributions with our work. First, we show that a naïve gradient descent optimization algorithm enabled by automatic differentiation (SGD) achieves state-of-the-art results for computer-generated holography in simulation. Second, we show that it is extremely challenging for all existing CGH algorithms, including SGD, to achieve high image fidelity with an experimental display. To address this challenge, we introduce and explore the concept of camera-in-the-loop (CITL) holography. Specifically, we demonstrate that optimizing the SLM phase pattern for a single target image with a camera in the loop can achieve the best image quality of all methods for a physical display. To remove the need for using a camera for every target image, we also propose a CITL-calibrated, model-based approach. Here, we train the model with the camera in the loop on a training set of images but then do the inference on unseen test data without the camera. This approach still results in higher image quality than that achieved by existing methods, but it is not quite as good as the single-image overfitting approach. Both the CITL optimization and model-based approach are iterative methods that take minutes to converge. To address this challenge, we propose a neural network–based generator that achieves real-time frame rates at a higher quality than all previous direct and iterative CGH algorithms and that can incorporate the CITL-calibrated wave propagation model. Finally, we demonstrate several variants of the proposed SGD method that enable 3D holography via varifocal and multiplane display modes. Animated holographic video clips, demonstrating temporal coherence of our methods, are included in the supplemental video.

As shown in Figure 2 (A), CGH algorithms make a tradeoff between runtime and quality. Several direct methods, including double phase–amplitude coding (DPAC) and the U-Net approach, achieve real-time frame rates of about 40 frames per second for full-color 1080p images with our PyTorch implementation, when executed on an NVIDIA Quadro RTX 6000 graphics processing unit. The proposed neural network architecture, HOLONET, achieves similar frame rates but with a significantly higher quality. With iterative methods, e.g., Gerchberg–Saxton (GS), Wirtinger Holography (WH), and stochastic gradient descent (SGD), one makes an implicit tradeoff between the number of iterations, and therefore runtime, and the achieved quality. Overall, our SGD approach outperforms all other methods. Interestingly, HOLONET achieves a slightly higher quality in real time than the previous state-of-the-art iterative method (WH) does in about 10 s (after having converged).

Limitations. The proposed algorithms have several limitations. Although the CITL optimization procedure introduced in Section 3.1 achieves the best image quality, it requires a camera in the loop and several minutes of training time for each target image. This can be impractical in some applications. However, automotive head-up displays and other display types only use a limited number of display elements, such as arrows or menu items, which could all be precomputed using the proposed CITL optimization. The model-based approach proposed in Section 5 generalizes across images and does not require a camera for the inference phase, but the generalization capability comes at the cost of slightly reduced image quality compared with the single-image overfitting approach. The slight

difference in quality between CITL optimization and model-based approach is likely due to content-dependent undiffracted light, which is challenging to calibrate accurately. The CITL-calibrated model is also required to train HOLONET, which is therefore fundamentally limited by the model's accuracy.

HOLONET is successful in generating high-quality RGB images both in simulation and with the CITL-calibrated model. However, we only demonstrate its capability of generating 2D images. There are several options to extend this approach to 3D holography. First, one could train one network for each target distance and then pick the network trained for the specific distance that the user fixates. This is similar to the varifocal display mode discussed in Section 7. Alternatively, the HOLONET architecture could be adapted to enable a neural-network-type multiplane display mode. For this purpose, multiple target-phase generators, i.e., one for each target distance, could be coupled with the same phase encoder on the SLM plane. In practice, such a network would require a lot of memory, which prevented us from implementing it on our GPUs.

Our holographic display prototype has several limitations. First, the display is a benchtop setup and not miniaturized for wearable applications. However, recent work has demonstrated miniaturized holographic near-eye display prototypes [Maimone et al. 2017], which could directly benefit from our algorithms. Second, the display is monocular and currently does not support stereo display. A second SLM and additional optical elements would be required to enable a stereoscopic holographic display mode.

Future Work. There are several interesting directions for future research. For example, developing neural-network architectures for real-time, true-3D holography is an exciting direction. For this purpose, either an RGB-D video stream could be directly used to generate holograms with continuous depth or light fields could be converted to phase patterns to enable 3D holograms with depth and view-dependent effects. Another possible direction could be to explore augmented reality applications, where the proposed CITL calibration scheme could be used with different types of optical beam combiners, such as diffractive waveguides or curved freeform combiners.

9 CONCLUSION

At the intersection of graphics and computational optics, advanced computer-generated holography algorithms are a key enabling technology for 3D virtual and augmented reality applications. With our work, we take first steps to combine classical CGH algorithms and optical systems with modern machine-learning techniques to address several long-standing challenges, such as speed and image quality. We believe that our work paves the way for a new era of neural holographic displays.

ACKNOWLEDGMENTS

We would like to thank Julien Martel for help with the camera calibration and Jonghyun Kim for help with the opto-electronic system. Suyeon Choi was supported by a Kwanjeong Scholarship and a Korea Government Scholarship. This project was further supported by Ford (Alliance Project), NSF (awards 1553333 and 1839974), a Sloan Fellowship, an Okawa Research Grant, and a PECASE by the ARO.

REFERENCES

Eirikur Agustsson and Radu Timofte. 2017. NTIRE 2017 Challenge on Single Image Super-Resolution: Dataset and Study. In *CVPR*.

Kaan Akşit, Ward Lopes, Jonghyun Kim, Peter Shirley, and David Luebke. 2017. Near-eye Varifocal Augmented Reality Display Using See-through Screens. *ACM Trans. Graph. (SIGGRAPH Asia)* 36, 6 (2017), 189:1–189:13.

Kurt Akeley, Simon J. Watt, Ahna Reza Girshick, and Martin S. Banks. 2004. A stereo display prototype with multiple focal distances. *ACM Trans. Graph. (SIGGRAPH)* 23, 3 (2004), 804–813.

George Barbastathis, Aydogan Ozcan, and Guohai Situ. 2019. On the use of deep learning for computational imaging. *Optica* 6, 8 (2019), 921–943.

Stephen A. Benton. 1983. Survey Of Holographic Stereograms. In *Proc. SPIE*, Vol. 0367.

Stephen A. Benton and V. Michael Bove. 2008. *Holographic Imaging*. Wiley-Interscience.

Max Born and Emil Wolf. 1959. *Principles of Optics*. Cambridge University Press.

Praneeth Chakravarthula, David Dunn, Kaan Akşit, and Henry Fuchs. 2018. Focusar: Auto-focus augmented reality eyeglasses for both real world and virtual imagery. *IEEE TVCG (ISMAR)* 24, 11 (2018), 2906–2916.

Praneeth Chakravarthula, Yifan Peng, Joel Kollin, Henry Fuchs, and Felix Heide. 2019. Wirtinger Holography for Near-eye Displays. *ACM Trans. Graph. (SIGGRAPH Asia)* 38, 6 (2019).

Jen-Hao Rick Chang, B. V. K. Vijaya Kumar, and Aswin C. Sankaranarayanan. 2018. Towards Multifocal Displays with Dense Focal Stacks. *ACM Trans. Graph. (SIGGRAPH Asia)* 37, 6 (2018), 198:1–198:13.

Jhen-Si Chen and Daping Chu. 2015. Improved layer-based method for rapid hologram generation and real-time interactive holographic display applications. *OSA Opt. Express* 23, 14 (2015), 18143–18155.

Rick H-Y Chen and Timothy D Wilkinson. 2009. Computer generated hologram with geometric occlusion using GPU-accelerated depth buffer rasterization for three-dimensional display. *Applied optics* 48, 21 (2009), 4246–4255.

Yuxin Chen, Yuejie Chi, Jianqing Fan, and Cong Ma. 2019. Gradient descent with random initialization: fast global convergence for nonconvex phase retrieval. *Mathematical Programming* 176 (2019), 1436–4646.

Rainer G Dorsch, Adolf W Lohmann, and Stefan Sinzinger. 1994. Fresnel ping-pong algorithm for two-plane computer-generated hologram display. *OSA Applied optics* 33, 5 (1994), 869–875.

D. Dunn, C. Tippets, K. Torell, P. Kellnhofer, K. Akşit, P. Didyk, K. Myszkowski, D. Luebke, and H. Fuchs. 2017. Wide Field Of View Varifocal Near-Eye Display Using See-Through Deformable Membrane Mirrors. *IEEE TVCG (VR)* 23, 4 (2017).

James R Fienup. 1982. Phase retrieval algorithms: a comparison. *Applied optics* 21, 15 (1982), 2758–2769.

Qiankun Gao, Juan Liu, Jian Han, and Xin Li. 2016. Monocular 3D see-through head-mounted display via complex amplitude modulation. *OSA Opt. Express* 24, 15 (2016).

Ralph W Gerchberg. 1972. A practical algorithm for the determination of phase from image and diffraction plane pictures. *Optik* 35 (1972), 237–246.

Joseph W Goodman. 2005. *Introduction to Fourier optics*. Roberts and Company.

Ryoichi Horisaki, Ryosuke Takagi, and Jun Tanida. 2018. Deep-learning-generated holography. *Appl. Opt.* 57, 14 (2018), 3859–3863.

CK Hsueh and AA Sawchuk. 1978. Computer-generated double-phase holograms. *Applied optics* 17, 24 (1978), 3874–3883.

Xinda Hu and Hong Hua. 2014. Design and assessment of a depth-fused multi-focal-plane display prototype. *J. Disp. Technol.* 10, 4 (2014), 308–316.

Hong Hua and Bahram Javidi. 2014. A 3D integral imaging optical see-through head-mounted display. *Optics express* 22, 11 (2014), 13484–13491.

Fu-Chung Huang, Kevin Chen, and Gordon Wetzstein. 2015. The light field stereoscope: immersive computer graphics via factored near-eye light field displays with focus cues. *ACM Trans. Graph. (SIGGRAPH)* 34, 4 (2015), 60.

Changwon Jang, Kiseung Bang, Gang Li, and Byoungho Lee. 2018. Holographic Near-eye Display with Expanded Eye-box. *ACM Trans. Graph. (SIGGRAPH Asia)* 37, 6 (2018).

Changwon Jang, Kiseung Bang, Seokil Moon, Jonghyun Kim, Seungjae Lee, and Byoungho Lee. 2017. Retinal 3D: augmented reality near-eye display via pupil-tracked light field projection on retina. *ACM Trans. Graph. (SIGGRAPH Asia)* 36, 6 (2017).

Justin Johnson, Alexandre Alahi, and Li Fei-Fei. 2016a. Perceptual losses for real-time style transfer and super-resolution. In *ECCV*. 694–711.

Paul V. Johnson, Jared AQ. Parnell, Joohwan Kim, Christopher D. Saunter, Gordon D. Love, and Martin S. Banks. 2016b. Dynamic lens and monovision 3D displays to improve viewer comfort. *OSA Opt. Express* 24, 11 (2016), 11808–11827.

Hoonjong Kang, Takeshi Yamaguchi, and Hiroshi Yoshikawa. 2008. Accurate phase-added stereogram to improve the coherent stereogram. *OSA Appl. Opt.* 47, 19 (2008).

Diederik P Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980* (2014).

Robert Konrad, Emily A. Cooper, and Gordon Wetzstein. 2016. Novel Optical Configurations for Virtual Reality: Evaluating User Preference and Performance with Focus-tunable and Monovision Near-eye Displays. In *Proc. ACM SIGCHI*. 1211–1220.
Pierre-Yves Laffont, Ali Hasnain, Pierre-Yves Guillemet, Samuel Wirajaya, Joe Khoo, Deng Teng, and Jean-Charles Bazin. 2018. Verifocal: A Platform for Vision Correction and Accommodation in Head-mounted Displays. In *ACM SIGGRAPH 2018 Emerging Technologies*. 21:1–21:2.
Douglas Lanman and David Luebke. 2013. Near-eye light field displays. *ACM Trans. Graph. (SIGGRAPH Asia)* 32, 6 (2013), 220.
Wai Hon Lee. 1970. Sampled Fourier transform hologram generated by computer. *Applied Optics* 9, 3 (1970), 639–643.
Gang Li, Dukho Lee, Youngmo Jeong, Jaebum Cho, and Byoungho Lee. 2016. Holographic display for see-through augmented reality using mirror-lens holographic optical element. *OSA Opt. Lett.* 41, 11 (2016), 2486–2489.
Sheng Liu, Dewen Cheng, and Hong Hua. 2008. An optical see-through head mounted display with addressable focal planes. In *Proc. IEEE ISMAR*. 33–42.
Patrick Llull, Noah Bedard, Wanmin Wu, Ivana Tosic, Kathrin Berkner, and Nikhil Balram. 2015. Design and optimization of a near-eye multifocal display system for augmented reality. In *OSA Imaging Appl. Opt.*
Gordon D. Love, David M. Hoffman, Philip J. W. Hands, James Gao, Andrew K. Kirby, and Martin S. Banks. 2009. High-speed switchable lens enables the development of a volumetric stereoscopic display. *Opt. Express* 17, 18 (2009), 15716–25.
Mark Lucente and Tinsley A Galyean. 1995. Rendering interactive holographic images. In *ACM SIGGRAPH*. 387–394.
Andrew Maimone, Andreas Georgiou, and Joel S Kollin. 2017. Holographic near-eye displays for virtual and augmented reality. *ACM Trans. Graph. (SIGGRAPH)* 36, 4 (2017), 85.
Andrew Maimone and Junren Wang. 2020. Holographic Optics for Thin and Lightweight Virtual Reality. *ACM Trans. Graph. (SIGGRAPH)* 39, 4 (2020).
Kyoji Matsushima and Sumio Nakahara. 2009. Extremely high-definition full-parallax computer-generated hologram created by the polygon-based method. *Applied optics* 48, 34 (2009), H54–H63.
Kyoji Matsushima and Tomoyoshi Shimobaba. 2009. Band-limited angular spectrum method for numerical simulation of free-space propagation in far and near fields. *Optics express* 17, 22 (2009), 19662–19673.
Olivier Mercier, Yusufu Sulai, Kevin Mackenzie, Marina Zannoli, James Hillis, Derek Nowrouzezahrai, and Douglas Lanman. 2017. Fast Gaze-contingent Optimal Decompositions for Multifocal Displays. *ACM Trans. Graph. (SIGGRAPH Asia)* 36, 6 (2017), 237:1–237:15.
Eunkyong Moon, Myeongjae Kim, Jinyoung Roh, Hwi Kim, and Joonku Hahn. 2014. Holographic head-mounted display with RGB light emitting diode light source. *OSA Opt. Express* 22, 6 (2014), 6526–6534.
Rahul Narain, Rachel A. Albert, Abdullah Bulbul, Gregory J. Ward, Martin S. Banks, and James F. O'Brien. 2015. Optimal Presentation of Imagery with Focus Cues on Multi-plane Displays. *ACM Trans. Graph. (SIGGRAPH)* 34, 4 (2015), 59:1–59:12.
Nitish Padmanaban, Robert Konrad, Tal Stramer, Emily A. Cooper, and Gordon Wetzstein. 2017. Optimizing virtual reality for all users through gaze-contingent and adaptive focus displays. *PNAS* 114 (2017), 2183–2188. Issue 9.
Nitish Padmanaban, Yifan Peng, and Gordon Wetzstein. 2019. Holographic Near-eye Displays Based on Overlap-add Stereograms. *ACM Trans. Graph. (SIGGRAPH Asia)* 38, 6 (2019).
Jae-Hyeung Park. 2017. Recent progress in computer-generated holography for three-dimensional scenes. *Journal of Information Display* 18, 1 (2017), 1–12.
Yifan Peng, Xiong Dun, Qilin Sun, and Wolfgang Heidrich. 2017. Mix-and-match holography. *ACM Trans. Graph.* 36, 6 (2017), 191.
K. Rathinavel, H. Wang, A. Blate, and H. Fuchs. 2018. An Extended Depth-at-Field Volumetric Near-Eye Augmented Reality Display. *IEEE TVCG* 24, 11 (2018).
Y. Rivenson, Y. Wu, and A. Ozcan. 2019. Deep learning in holography and coherent imaging. *Light: Science & Applications* 8, 85 (2019).
Yair Rivenson, Yibo Zhang, Harun Günaydın, Da Teng, and Aydogan Ozcan. 2018. Phase recovery and holographic image reconstruction using deep learning in neural networks. *Light: Science & Applications* 7, 2 (2018), 17141.
Jannick P. Rolland, Myron W. Krueger, and Alexei Goon. 2000. Multifocal planes head-mounted displays. *Appl. Opt.* 39, 19 (2000), 3209–3215.
Olaf Ronneberger, Philipp Fischer, and Thomas Brox. 2015. U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical image computing and computer-assisted intervention*. Springer, 234–241.
Y. Shechtman, Y. C. Eldar, O. Cohen, H. N. Chapman, J. Miao, and M. Segev. 2015. Phase Retrieval with Application to Optical Imaging: A contemporary overview. *IEEE Signal Processing Magazine* 32, 3 (2015), 87–109.
Liang Shi, Fu-Chung Huang, Ward Lopes, Wojciech Matusik, and David Luebke. 2017. Near-eye Light Field Holographic Rendering with Spherical Waves for Wide Field of View Interactive 3D Computer Graphics. *ACM Trans. Graph. (SIGGRAPH Asia)* 36, 6, Article 236 (2017), 236:1–236:17 pages.
Shinichi Shiwa, Katsuyuki Omura, and Fumio Kishino. 1996. Proposal for a 3-D display with accommodative compensation: 3DDAC. *Journal of the Society for Information Display* 4, 4 (1996), 255–261.
Karen Simonyan and Andrew Zisserman. 2014. Very deep convolutional networks for large-scale image recognition. In *CVPR*.
Ayan Sinha, Justin Lee, Shuai Li, and George Barbastathis. 2017. Lensless computational imaging through deep learning. *Optica* 4, 9 (2017), 1117–1125.
MJ Townson, OJD Farley, G Orban de Xivry, J Osborn, and AP Reeves. 2019. AOtools: a Python package for adaptive optics modelling and analysis. *Optics express* 27, 22 (2019), 31316–31329.
Ethan Tseng, Felix Yu, Yuting Yang, Fahim Mannan, Karl ST Arnaud, Derek Nowrouzezahrai, Jean-François Lalonde, and Felix Heide. 2019. Hyperparameter optimization in black-box image processing using differentiable proxies. *ACM Trans. Graph. (SIGGRAPH)* 38, 4 (2019), 1–14.
Koki Wakunami, Hiroaki Yamashita, and Masahiro Yamaguchi. 2013. Occlusion culling for computer generated hologram based on ray-wavefront conversion. *Optics express* 21, 19 (2013), 21811–21822.
Fahri Yaras, Hoonjong Kang, and Levent Onural. 2010. State of the Art in Holographic Displays: A Survey. *Journal of Display Technology* 6, 10 (2010), 443–454.
Han-Ju Yeom, Hee-Jae Kim, Seong-Bok Kim, HuiJun Zhang, BoNi Li, Yeong-Min Ji, Sang-Hoo Kim, and Jae-Hyeung Park. 2015. 3D holographic head mounted display using holographic optical elements with astigmatism aberration compensation. *OSA Opt. Express* 23, 25 (2015), 32025–32034.
Hao Zhang, Liangcai Cao, and Guofan Jin. 2017. Computer-generated hologram with occlusion effect using layer-based processing. *Applied optics* 56, 13 (2017).
Hao Zhang, Neil Collings, Jing Chen, Bill A Crossland, Daping Chu, and Jinghui Xie. 2011. Full parallax three-dimensional display with occlusion effect using computer generated hologram. *Optical Engineering* 50, 7 (2011), 074003.
Zhengyun Zhang and M. Levoy. 2009. Wigner distributions and how they relate to the light field. In *Proc. ICCP*. 1–10.
Remo Ziegler, Simon Bucheli, Lukas Ahrenberg, Marcus Magnor, and Markus Gross. 2007. A Bidirectional Light Field-Hologram Transform. In *Computer Graphics Forum (Eurographics)*, Vol. 26. 435–446.

APPENDIX A

In this appendix, we briefly review the relationship between amplitude of a wave field, linear intensity, and gamma-corrected intensity because it can be confusing. A complex-valued field u is typically modeled as $u(x,y) = a(x,y)\,exp(i\phi)$, where a is the spatially varying amplitude and ϕ is the phase. The phase of wave fields in the visible spectrum cannot be directly measured, but we can record the linear intensity of this field as $i_{lin} = |u(x,y)|^2 = a(x,y)^2$. For example, a photograph captured with a camera in RAW mode is close to linear in intensity. Most cameras, however, internally apply a gamma correction to the linear intensity before saving them out as an 8-bit image (e.g., as a .jpg file): $i = \gamma(i_{lin})$. This nonlinear gamma curve γ is typically defined by the sRGB standard as

$$\gamma(i_{lin}) = \begin{cases} 12.92\, i_{lin} & i_{lin} \leq 0.0031308 \\ 1.055\, i_{lin}^{0.41\overline{6}} - 0.055 & \text{otherwise} \end{cases} \tag{8}$$

With these definitions in hand, we can specify a routine to process a target image, given in sRGB space, with any CGH algorithm. For this purpose, we load an 8-bit image where intensities i are specified in gamma-corrected space for each color channel. We invert the gamma to compute linear intensity as $i_{lin} = \gamma^{-1}(i)$ and finally compute the amplitude as $a = \sqrt{i_{lin}}$. The sRGB gamma function closely resembles the shape of the function $(\cdot)^{2.2}$, so $a \approx \sqrt{i^{2.2}} \approx i$. Due to the fact that the sensitivity of the human visual system is closely approximated by the sRGB gamma function, using either the gamma-corrected intensity i or the target amplitude a for CGH optimization, i.e., optimizing the difference between target amplitude and estimated amplitude of some computed phase pattern on the target plane, is usually a good idea.

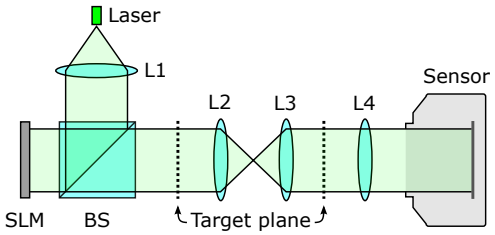

Fig. 10. Holographic near-eye display setup. A laser emits a coherent wave field that is collimated by a lens (L1). Using a beam splitter (BS) cube, the field is directed to the reflective spatial light modulator (SLM) where it is delayed in a per-SLM-pixel manner. The resulting diffracted field forms an image at the target plane. An optional 4f system (L2, L3) can be used to filter out the DC component of the field and higher diffraction orders. The resulting image is re-imaged on a sensor using another lens (L4) or directly observed by a user.

APPENDIX B

In this appendix, we describe implementations details of our holographic display system, which is also illustrated in Figure 10.

Hardware. The SLM in our prototype is a HOLOEYE LETO phase-only LCoS with a resolution of 1,920 × 1,080 and a pixel pitch of 6.4 μm. This device provides a refresh rate of 60 Hz (monochrome) with a bit depth of 8 bits and a diffraction efficiency of over 80%. The laser is a FISBA RGBeam fiber-coupled module with three optically aligned laser diodes with wavelengths of 638, 520, and 450 nm. Note that in our implementation, color images are captured as separate exposures for each wavelength and then combined in post-processing. The eyepiece is a Nikon AF-S 50 mm lens. Other components include a polarizer, collimating lenses, and a beam splitter. We further use a 4f system to provide options of filtering out the zeroth and higher order diffraction artifacts. All images are captured with a FLIR Grasshopper3 2.3 MP color USB3 vision sensor through a Nikon AF-S Nikkor 35mm f/1.8G lens.

Software. We implemented all CGH algorithms, except Wirtinger Holography, in PyTorch. Matlab code for WH was provided by the authors of that paper. All of our source code, calibrated model parameters, and the pre-trained HoloNet are available on the project website at www.computationalimaging.org. Zernike polynomials were calculated using AOtools [Townson et al. 2019]. Additional implementation details, including the calibration procedure and the training/inference runtimes, are included in the Supplementary Material Section S5.