

Supplemental Material: Simultaneous Color Computer Generated Holography

ERIC MARKLEY, Reality Labs Research, Meta, USA

NATHAN MATSUDA, Reality Labs Research, Meta, USA

FLORIAN SCHIFFERS, Reality Labs Research, Meta, USA

OLIVER COISSART, Reality Labs Research, Meta, USA

GRACE KUO, Reality Labs Research, Meta, USA

S1 ADDITIONAL IMPLEMENTATION DETAILS

Spatial Light Modulator. For all simulations, a spatial light modulator (SLM) with 1920×1080 pixels and a pixel size of $8 \mu\text{m} \times 8 \mu\text{m}$ is used. The phase ranges of the red, green, and blue channels are 2.4π , 5.9π , and 7.4π , respectively, unless otherwise noted. These values were experimentally calibrated for the Holoeye Pluto-2.1-Vis-016 SLM. The propagation distance of all simulated holograms is 100 mm unless otherwise noted.

Modified High Order Angular Spectrum Method (HOASM). We implement a modified version of the High Order Angular Spectrum Method (HOASM) as described by Gopakumar et al. [2021]. Instead of propagating the zero- and first-order together, we propagate them separately. The zero-order is propagated by performing the traditional angular spectrum method (ASM). To propagate the first-order, we pattern the zero-padded Fourier transform of the complex field to be propagated into a 3×3 grid. The center Fourier transform of the grid is then zeroed out. The Fourier representation of the first-order is then weighted with a sinc function and propagated to the sensor plane using ASM. The field is then down-sampled and cropped. The complex fields of the zero- and first-orders are then split into real and imaginary parts and stacked before being fed into a U-Net. The U-Net consists of 4 downsampling layers, the number of channels increases from 4 to 32 during the first downsampling layer and doubles in each of the next 3 downsampling layers until there are 256 channels. Four upsampling layers are then applied, producing a single-channel output representing the intensity of the propagated wavefront.

Camera Space to SLM Space Homography. To perform either offline or active camera-in-the-loop optimization, the captured wavefront and SLM must be in the same space. This requires a transform and downsampling of the captured image to place it in the same coordinate system as the SLM pattern used to generate it. We opt to use an affine transform to perform this mapping. The affine transform is calculated as follows: first, an SLM pattern is calculated that produces a grid of dots. The dots are then detected on the sensor, and their centers are estimated in camera space coordinates. The centers of the dots are known in SLM space since the target image containing the dots is in SLM space for optimization. Finally, Python’s OpenCV package is used to produce the affine transform matrix that maps the captured dots to the SLM coordinate space. A unique homography is calculated for each depth location and color channel.

Source Power Optimization. Correctly setting the power of each color channel of the SLED for a given hologram is an important step to achieving good color fidelity. To achieve this, we use an active camera-in-the-loop based approach to optimize the power of the color channels. First, the power of the source is set to an arbitrary value less than 100% across all three color channels. A baseline reference image is captured, debayered, and mapped to the SLM space. Three learnable weighting parameters, one for each color channel, are initialized to unity and applied

Authors’ addresses: Eric Markley, emarkley@berkeley.edu, Reality Labs Research, Meta, USA; Nathan Matsuda, nathan.matsuda@meta.com, Reality Labs Research, Meta, USA; Florian Schiffers, fschiffers@meta.com, Reality Labs Research, Meta, USA; Oliver Coissart, ocoissart@meta.com, Reality Labs Research, Meta, USA; Grace Kuo, gracekuo@meta.com, Reality Labs Research, Meta, USA.

to the captured reference image. These weighting parameters serve as a proxy to optimizing the source power. An iterative process is then undertaken, where an image is captured on the camera, debayered, and mapped to the SLM space. The loss between this image and the target image is calculated and then backpropagated using the computational graph of the weighting parameters applied to the reference image. The initial source power is then multiplied by the updated weighting parameters, and a new image is captured, restarting the iterative loop. This is done until the color weighting parameters have converged, usually taking between 15-30 iterations. If the process fails to converge or the initial source power multiplied by the weighting function becomes greater than 100%, the exposure time is increased, and the source power optimization is restarted.

Although we use camera feedback in this process, we note that the information needed for source power optimization is contained in the color balance of the image itself. We believe this step could be replaced with a precomputed source power that's dependent on the image content.

Simultaneous Color Focal Stack Optimization. In this approach, we optimize the SLM pattern for a focal stack using simultaneous color holography. First, we load a learned model for wave propagation for our unfiltered holography system. Then, the SLM pattern is initialized and target intensities defined for each plane in the multiplane hologram. Gradient descent is used to optimize the SLM pattern. For each plane, the field generated from the current SLM pattern is calculated and propagated to the current plane of interest using the learned propagation model. The loss is computed using our custom perceptual loss function and back-propagated to calculate the gradient of the loss with respect to the SLM pattern. The calculated gradients are then used to update the SLM pattern. This process is performed iteratively for each depth plane and continued until the SLM pattern converges. Finally, we capture a simultaneous color focal stack by displaying the optimized SLM pattern, moving the camera to each plane in the multiplane hologram and capturing. Pseudocode for this method is provided in Algorithm 1.

Algorithm 1 Simultaneous Color Focal Stack Optimization

- 1: **Initialize:** Load learned model for wave propagation. Initialize SLM pattern P_{SLM} and target intensity I_{target} for each depth plane in the focal stack.
 - 2: **Optimize:**
 - 3: **while** SLM pattern P_{SLM} not converged **do**
 - 4: **for** each depth plane, d , in the focal stack hologram **do**
 - 5: Clear previous gradients
 - 6: Generate field from current SLM pattern and propagate to depth plane d : $Field_{out} = Holo(P_{SLM}, depth = d)$
 - 7: Compute loss with custom perceptual loss function: $Loss = LossFunc(Field_{out}, I_{target}[d])$
 - 8: Backpropagate the gradient of $Loss$ with respect to P_{SLM}
 - 9: Update the SLM pattern P_{SLM}
 - 10: **Capture:** Experimentally capture the simultaneous color focal stack
 - 11: **for** each depth plane, d , in the focal stack hologram **do**
 - 12: Move camera to depth plane d
 - 13: Capture hologram: $Hologram = CaptureImage(camera)$
 - 14: Add captured hologram to the focal stack: $FocalStack.append(Hologram)$
 - 15: **Output:** The optimized SLM pattern, P_{SLM} , and the captured simultaneous color focal stack, $FocalStack$
-

S2 EFFECT OF EYEPIECE ON NEAR-EYE HOLOGRAPHIC DISPLAYS

Our experimental setup creates an image directly on the bare sensor, but for a human-viewable system, we would use an eyepiece between the image plane and the user's eye. For a near-eye display, the eyepiece is needed to make the image plane appear further away so the eye can focus on it. Together with the lens of the user's eye, the eyepiece creates an optical relay system that generates a copy of the image plane on the user's retina.

There are three major differences between this more realistic setup with an eyepiece and our experimental setup without an eyepiece: (1) The relay system generally does not have unit magnification, which causes the image plane and corresponding color depth replicas to appear at new axial locations. (2) The finite aperture of the eyepiece can block some of the wavefront, creating artifacts and vignetting at the edges of the field of view. (3) Optical aberrations in the eyepiece can reduce image quality, adding blur or undesirable speckle if not compensated for.

We now explore each of these differences in more detail.

S2.1 Effect of the Eyepiece on Depth Replica Locations

After being viewed through the eyepiece, the image plane of the holographic display appears at a new depth in the world. This is governed by the thin lens equation

$$z_{\text{world}} = \frac{f z_{\text{slm}}}{f - z_{\text{slm}}}, \quad (1)$$

where f is the focal length of the eyepiece, z_{slm} is the actual distance from the eyepiece to the image plane, and z_{world} is the apparent distance to the image when viewed through the eyepiece.

The apparent positions of the replica planes will also be shifted in the world based on the eyepiece, and in general, replica planes are spread further apart. We illustrate this effect with a concrete example.

Consider the case where we have an eyepiece of focal length $f = 30$ mm and the SLM is co-located with the eyepiece for a thin form factor. If we want the image plane to appear at 1 m in the world, we can use Eq. 1 to calculate that the propagation distance from the SLM should be $z_0 = 29.13$ mm. If the image is created in green (512 nm), the replicas in red (636 nm) and blue (453 nm) will appear at 36.18 mm and 25.77 mm, respectively, relative to the SLM. For both of these replica planes, we can apply Eq. 1 to calculate the position of these planes in the world after being viewed through the eye piece.

For the red replica at 36.18 mm, z_{world} is negative, indicating that the apparent image is *behind* the viewer (in other words, the illumination is converging, rather than diverging)—this is not representative of real, physical objects and our eyes generally cannot focus on this plane, meaning that this replica is no longer visible.

For the replica in blue at 25.77 mm relative to the SLM, we calculate that this corresponds to a depth 18.28 cm once viewed through the eyepiece. Recall that the original (non-replica) plane is a 1 m from the eye piece, demonstrating that the eyepiece causes the replicas to be significantly more spread out. In fact, the eyes of many adults cannot accommodate at such a close distance, meaning that such individuals would not be able to focus on the replica plane.

In general, as the eyepiece focal length gets shorter, the small distances between the replicas are exaggerated. This makes them less visible since the replicas appear in regions where it's more difficult for the viewer to accommodate.

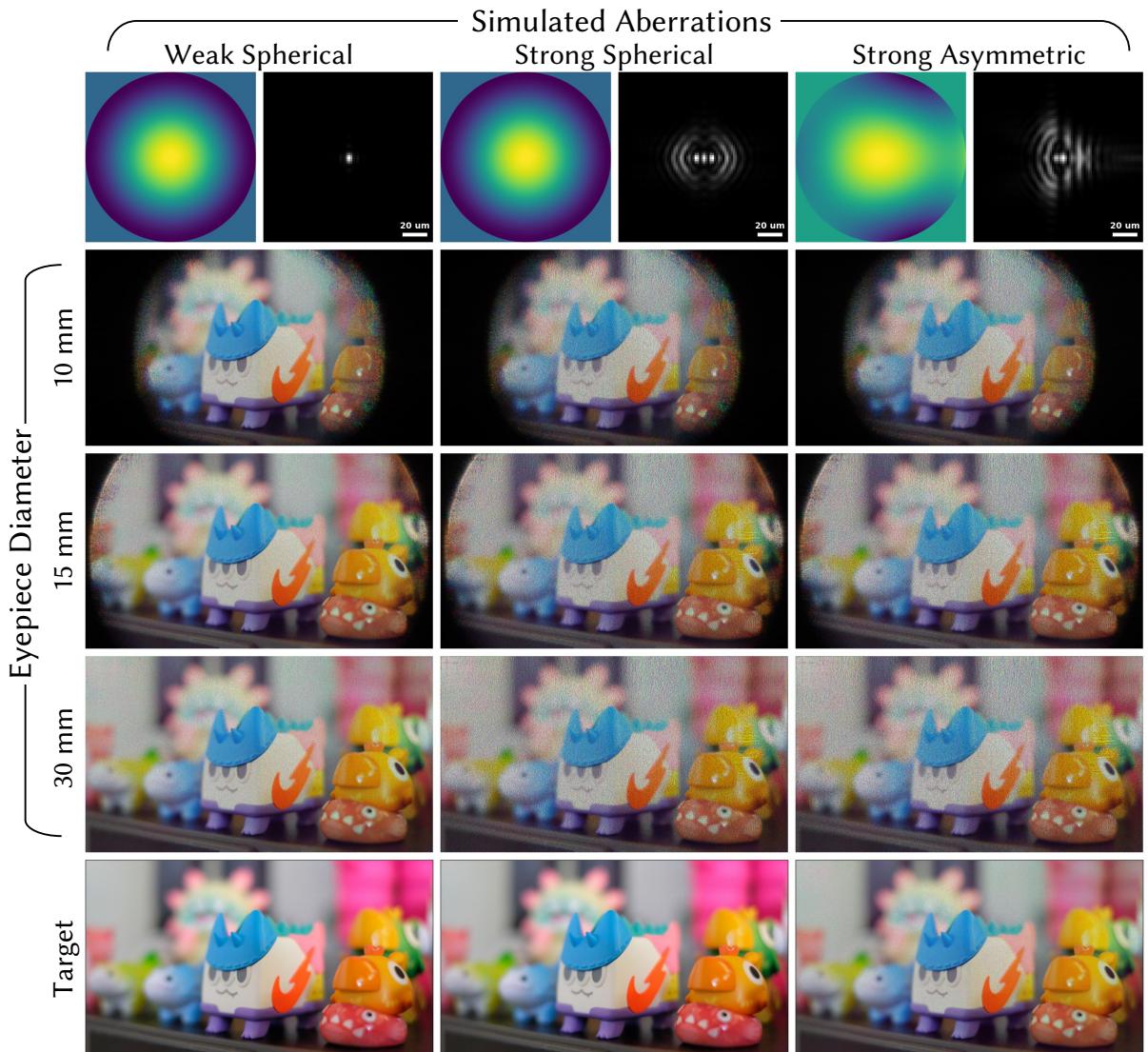


Fig. S1. Effect of eyepiece on image quality (simulation). Each column shows different simulated eyepiece aberrations, with the Zernike phase error and PSF shown at the top. Each row shows a different eyepiece diameter. With small aberrations where the PSF is contained to a single SLM pixel (left column), there is minimal impact on image quality. Stronger aberrations, both symmetric (center column) and asymmetric (right column) add additional speckle to the image. This could be compensated for computationally if the aberrations are known. When the eyepiece diameter is smaller than the SLM size (10 mm, top) only the center of the image is visible and there is additional speckle at the edges. When the eyepiece is approximately the same size as the SLM (15 mm, center), there is vignetting in the corners and reduced quality at the edges. When the eyepiece is sufficiently large (30 mm, bottom), there are no artifacts from the eyepiece aperture size.

S2.2 Effect of the Eyepiece on Image Quality

Since the eyepiece creates a relay system between the image plane and the user's retina, with an ideal eyepiece, the image seen by the user is a magnified version of the image directly in front of the SLM. However, the finite size of the eyepiece aperture can cause artifacts at the edges of the field of view, and aberrations in the eyepiece can further reduce image quality.

Figure S1 simulates the effect of these non-idealities on image quality. We assume an $f = 30$ mm focal length eyepiece and an SLM with $8 \mu\text{m}$ pixels and 1920×1080 resolution, which matches our experimental results. We simulate an eyepiece diameter ranging from 10 mm to 30 mm. Note that the SLM is about 16 mm across. When the eyepiece diameter is larger than the SLM size (30 mm, bottom), there are no visible edge effects in the image. When the eyepiece diameter is close to the SLM size (15 mm, middle), some of the image is lost in the corners and there's additional speckle around the edges. When the eyepiece has a significantly smaller diameter (10 mm, top), only a fraction of the image is viewable due to the finite size of the eyepiece. This suggests that the eyepiece diameter must be larger than the size of the image plane in order to cover the full field of view.

We also simulate the effect of aberrations on image quality. The left column show a simulated lens with a small amount of spherical aberration, which is representative of a well-corrected eyepiece. Since the point spread function (PSF) of this simulated lens is smaller than $8 \mu\text{m}$, the SLM pixel size, we see good image quality with minimal speckle. Once the spot size starts to exceed the pixel size of the SLM, additional speckle becomes visible in the image. The image quality is qualitatively similar for symmetric aberrations like spherical (center column) and asymmetric aberrations (right column). We note that holographic displays can compensate for aberrations [Maimone et al. 2017] if the aberrations are known, so even poorly corrected eyepieces can yield high quality images. However, here we simulate the case where aberrations are not corrected for computationally.

In conclusion, without further computational correction, an eyepiece will not reduce image quality if the PSF is smaller than the SLM pixel size and the eyepiece diameter is larger than the total SLM size.

S3 ACTIVE CAMERA-IN-THE-LOOP (CiTL)

Active CiTL [Peng et al. 2020] is a special case of camera-calibrated models in which an image is displayed on the SLM, and camera captures are used to improve that particular image using the difference between the experimental capture and target image. While active CiTL is incompatible for real time displays, it does provide a useful proof of achievable hologram quality. Consequently, we implemented active CiTL for our system as follows.

First, an SLM pattern is optimized using our learned simulation model and the computational graph is retained. This SLM pattern is then displayed and the resulting hologram is captured. A homography is applied to the captured hologram for each color channel to map it from camera space to simulation space. Our perceptual loss function is applied to the remapped captured hologram and target image. Backpropagation is performed using a computational graph saved during the forward pass, but the experimentally captured hologram is used in the loss function (instead of the simulated model output). This is the first time to our knowledge that active CiTL has been combined with a deep component to the forward model. Figure S2 shows reduced noise and improved color fidelity for holograms generated with active CiTL. Since active CiTL uses the difference between the experimental capture and the target, the alignment between the two must be precise. We find that improved alignment using a piecewise affine homography, rather than a global affine homography, dramatically improves color fidelity. A comparison of this case is shown in Figure S3.



Fig. S2. **Active camera-in-the-Loop (CiTL) reduces noise and improves color fidelity.** The first column of this image depicts experimentally captured color holograms. The second columns shows images that were iteratively improved with a camera in the system using the active CiTL algorithm of Peng et al. [2020].

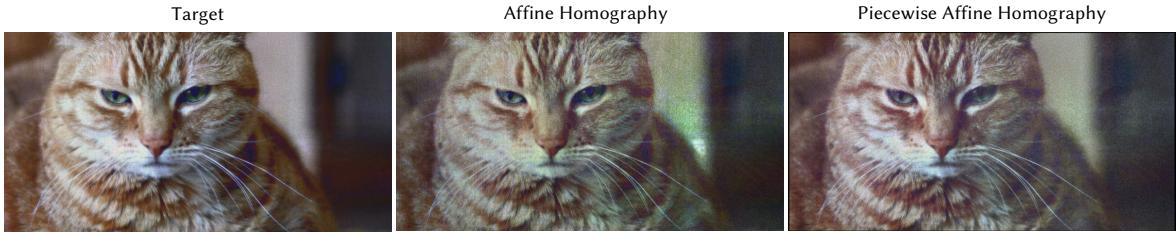


Fig. S3. **Piecewise affine homography improves color fidelity for active CiTL.** The first column shows the target image. The second column shows the experimentally captured hologram optimized using active CiTL with a global affine homography. The third column depicts active CiTL with a piecewise affine homography, which reduces color artifacts and noise due to better alignment during optimization. Cat source image by Chris Erwin (CC-BY-2.0).

S4 ADDITIONAL EXPERIMENTAL RESULTS AND FAILURE CASES

Figure S4 depicts additional captured results, which are intended to showcase a wider variety of scenes and include failure cases of our method. Our method has the most difficulty when the target has large, flat areas (i.e. textureless) of saturated color. Textureless targets lack high frequency information that can be leveraged by our loss function, leading to substantial artifacts such as color non-uniformity and ringing. These artifacts are particularly apparent in the image of colored bars in Fig. S4. Highly saturated images or “unnatural” images (like the colored bars) often fail due to disparate color channels, resulting in a single SLM pattern having to produce three holograms at the same plane with substantially different structures. In contrast, natural images typically have similarly structured color channels. We provide further simulation results of this failure case in Figure S5.

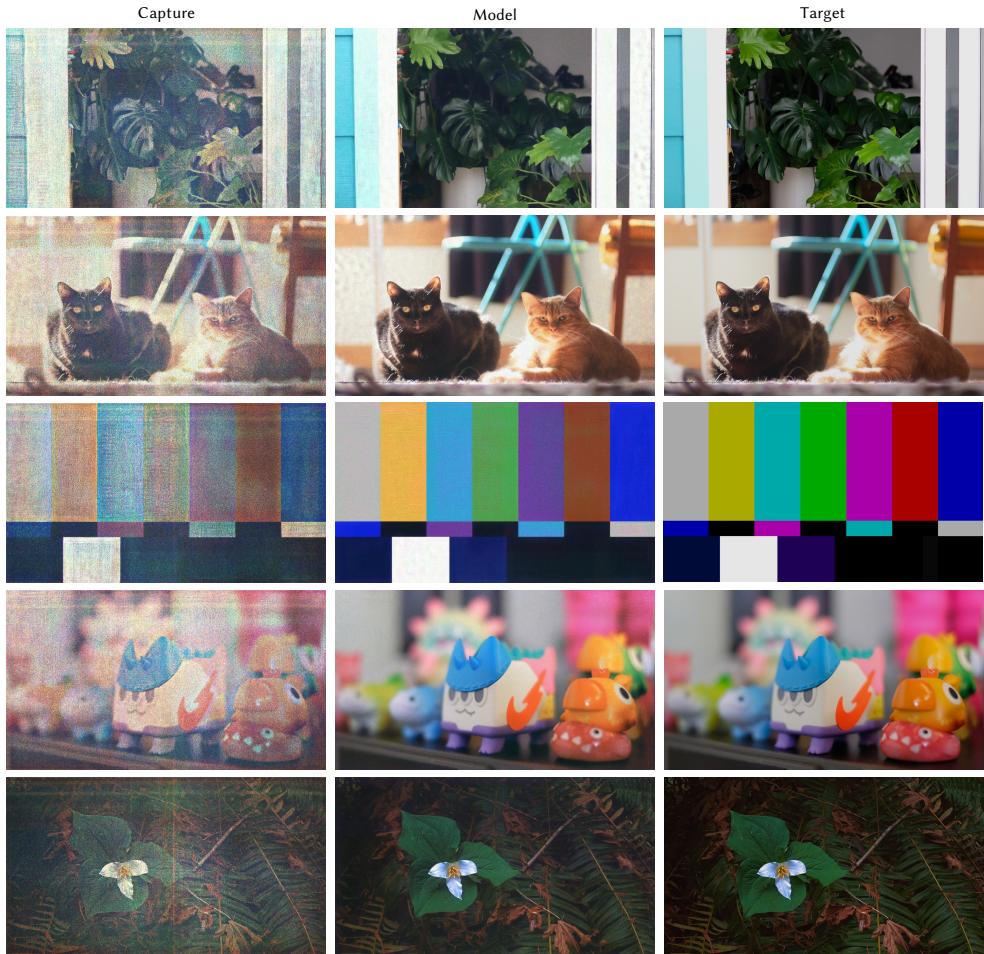


Fig. S4. Additional simultaneous color holograms captured in experiment. The first column depicts holograms captured in experiment. The second column shows the simulation output. The third column depicts the target image. Although our system performs well on most natural scenes, unnatural images such as the bars in the center row are more challenging for our algorithm.

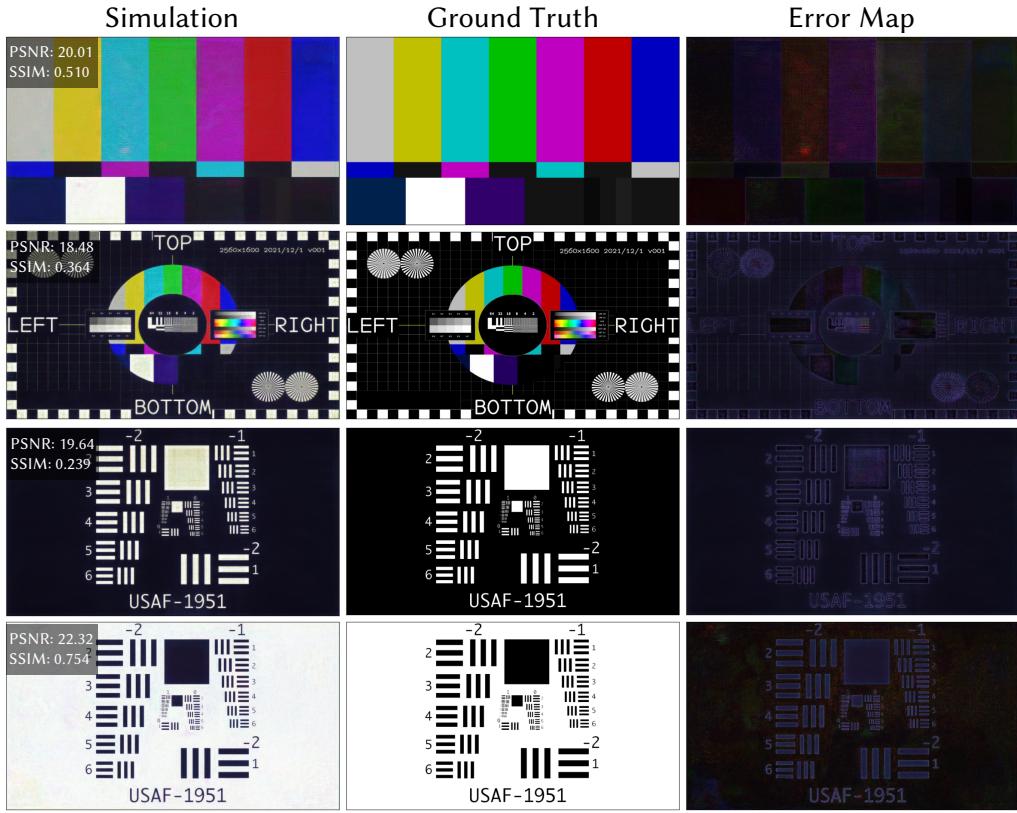


Fig. S5. Failure case simulation results. The first column depicts holograms simulated using our proposed method with no U-Net component. The second column shows the target image. The an error map between the simulation and the ground truth. Our proposed method struggles to produce holograms that contain large patches of highly saturated colors. Our method also struggles to produces white content on a black background as demonstrated in row three.

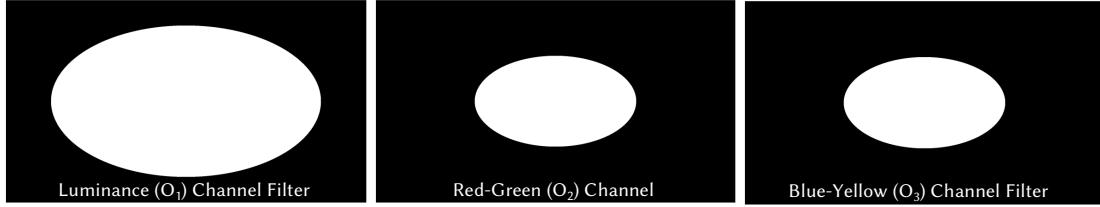


Fig. S6. Perceptual loss function filters in Fourier opponent color space. The white areas of the filters pictured represents the pass band of the filter. The luminance channel has a filter width of 75% of Fourier space. Both chrominance channels (Red-Green, Blue-Yellow) have filter widths of 45% of Fourier space.

S5 ADDITIONAL PERCEPTUAL LOSS FUNCTION DETAILS

Figure S6 shows a visualization of the perceptual loss function filters used in our algorithm. These filter sizes were kept constant regardless of the scene being optimized. To test the effectiveness of our perceptual loss function, we applied it to a personally captured dataset of 294 images. For each target image, an SLM pattern was optimized using both the traditional RGB loss function and our perceptual loss function. The resulting hologram was then captured, and the perceptual filter was applied. The PSNR, SSIM, and NMSE were calculated for the filtered simulated holograms and the perceptually filtered target image. The average metrics over the entire dataset are provided in Table S1.

To make our system more general, we defined the perceptual loss filters relative to the maximum spatial frequency of the SLM instead of in physical units. However, by choosing the focal length of the eye piece, we can relate the filter sizes to physical quantities through the following relationship:

$$\text{cutoff (cycles/deg)} = \frac{f}{2p} \cdot \frac{\pi}{180} \cdot \gamma \quad (2)$$

where f is the focal length of the eye piece, p is the SLM pixel size, and γ is the filter width defined as a fraction of the Fourier space extent. In our simulations, $\gamma = 0.75$ for luminance and 0.45 for chrominance as shown in Fig. S6. Therefore, with an 8 μm pixel pitch and eye piece focal length of 36.6 mm, the cutoffs correspond to 30 cycles/deg and 18 cycles/deg, for luminance and chrominance respectively. This is similar to perceptual measurements of human vision: Mullen [1985] measure approximately 30 cycles/deg in luminance and 11-12 cycles/deg in chrominance, which is actual slightly less accuity in color than we target in our images.

	PSNR	SSIM	NMSE
RGB Loss Function	20.11	0.603	0.010
Perceptual Loss Function	26.58	0.869	0.003

Table S1. A comparison of the average PSNR, SSIM, and NMSE for holograms optimized with the traditional RGB loss function and perceptual loss function. The metrics were calculated between the perceptually filtered simulated holograms and the perceptually filtered target. The data set used was a personally captured set of 294 images of natural scenes.

S6 THE EFFECT OF BIT DEPTH ON HOLOGRAM QUALITY

The effect of quantization on hologram quality is an important consideration when choosing an extended phase SLM. We define the effective bit depth as the number of bits contained in a 2π interval of the extended range. For example, the effective bit depth of an 8-bit SLM with a phase range of 8π is 6 bits as each 2π interval contains 64 discrete samples i.e. 6 bits. To determine the minimum bit depth required for adequate image quality, we simulated holograms using an SLM with a 2π phase range and bit depths from 2 bits to 8-bits. Simulations are done by optimizing the hologram with gradient descent, then quantizing to the target bit depth. A significant drop off in both PSNR and SSIM was observed between 5 and 6 bits, as depicted in Fig. S7. This suggests that the minimum effective bit depth required for an extended phase SLM is 6 bits. Since most commercially available SLMs are 8 bits, this suggests that the maximum phase range in any channel should be 8π , which aligns well with the SLM used in our experiments (maximum phase range of 7.4π in the blue channel). Additionally, we find that no image quality improvement is achieved for simultaneous color holograms once each color channel has a bit depth of at least 6 bits across a 2π phase range. The results of this simulation are displayed in Figure S8.

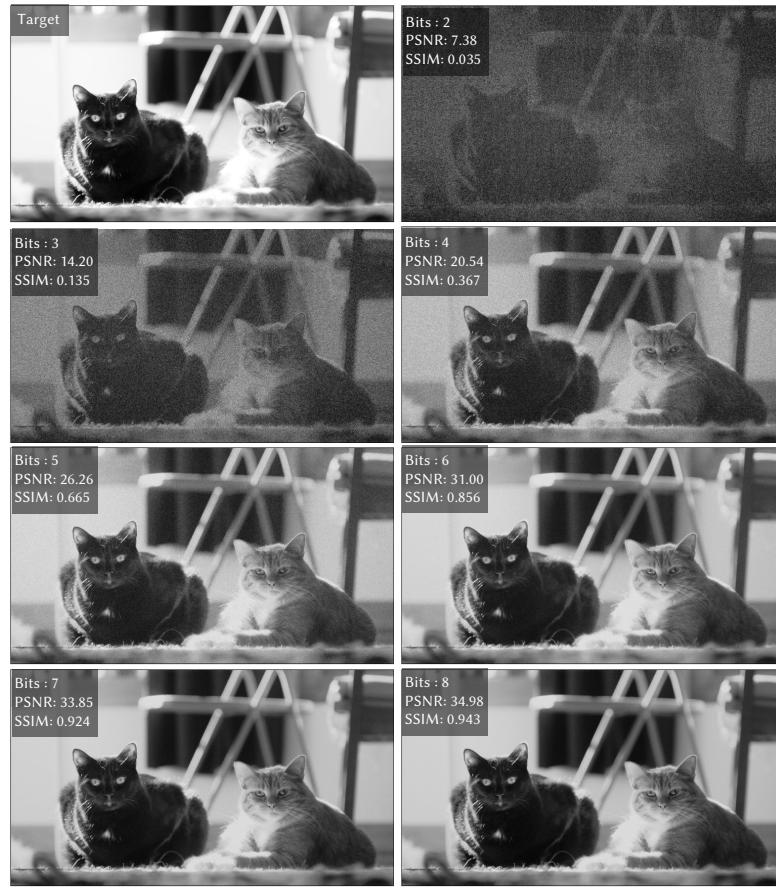


Fig. S7. An analysis of SLM bit depth on hologram quality in simulation We simulate holograms using SLMs of 2 to 8 bits. The target image is pictured in the top left of the figure. One should note the rapidly increasing drop off in both PSNR and SSIM between 5 and 6 bits.

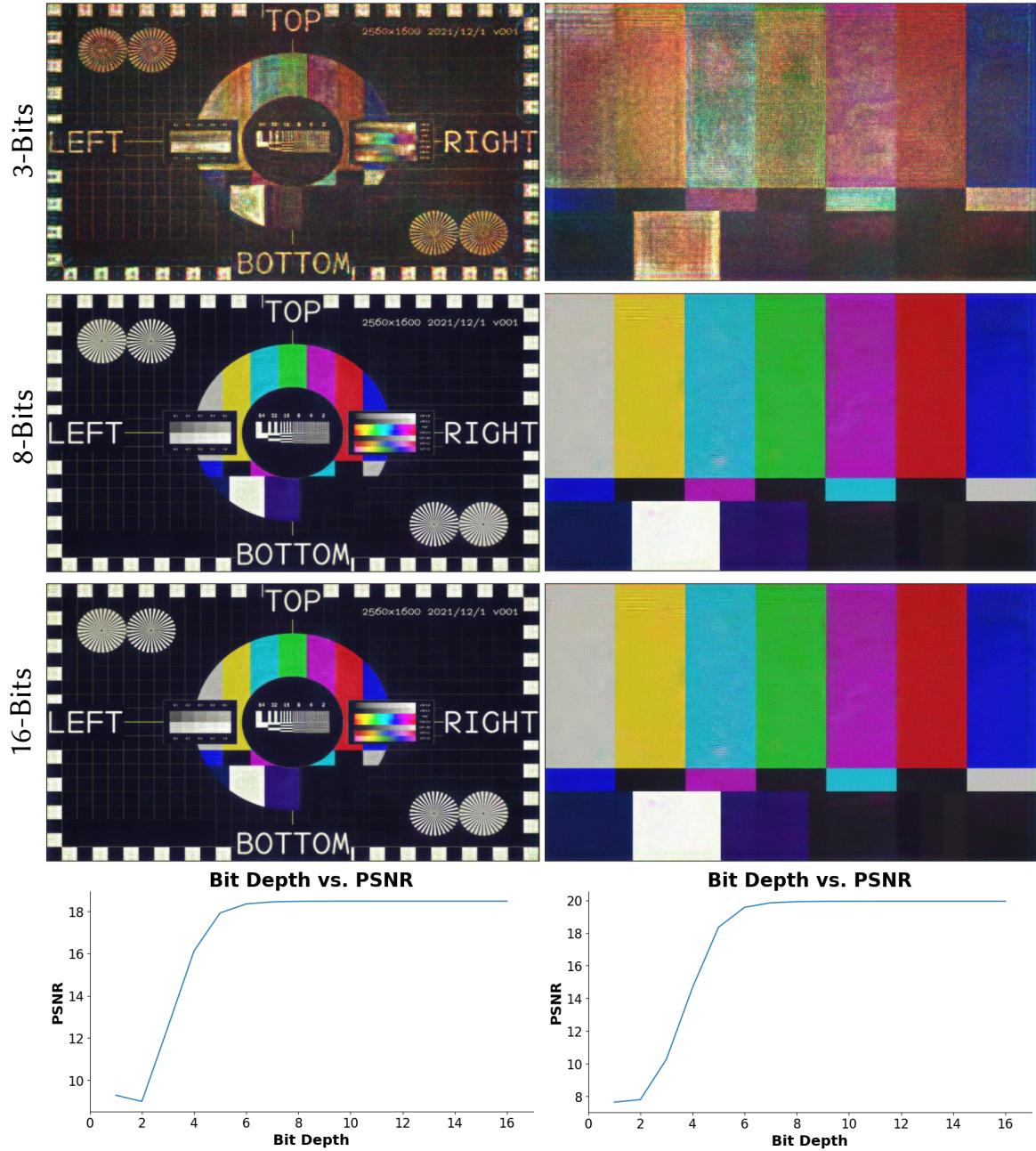


Fig. S8. An analysis of SLM bit depth on simultaneous color hologram quality in simulation We simulate holograms using SLMs of 1 to 16 bits. Rows 1-3 depict the simulated simultaneous color holograms with an SLM of bit depth 3, 8, and 16 respectively. Row 4 depicts the bit depth vs. PSNR. PSNR is calculated with the perceptually filtered ground truth and perceptually filtered simulated hologram.

S7 THE EFFECT OF PHASE RANGE ON SIMULTANEOUS COLOR HOLOGRAM QUALITY

The phase range of an SLM plays a critical role in the arena of simultaneous color holography. In this context, we must ensure that the phase range for each of the three color channels is sufficiently different. This differentiation is crucial as it allows for the production of unique holograms in each color channel, overcoming the inherent depth-wavelength ambiguity associated with the ASM kernel. However, there's a trade-off. An SLM with a larger phase range can potentially degrade the image quality by reducing the effective bit depth, as detailed in Section S6. Additionally, SLMs with larger phase ranges tend to have slower refresh rates, counterbalancing some of the advantages gained from simultaneous illumination.

Thus, when choosing an SLM for this purpose, the aim should be to select one that offers the minimal phase range necessary to achieve the desired image quality. In our study depicted in Figure S9, we conducted a series of experiments with various phase values to find a range that delivers adequate quality simultaneous color holograms for both natural and unnatural images. We utilized gradient descent with a naive ASM forward model for each SLM pattern. The L2 norm between the target image and the simulated hologram served as the loss function.

Our experiment involved fixing the red channel phase range at 2π and incrementally increasing the blue channel phase range from 2π to 8π . We set the green channel's phase range as the average of the red and blue channels. Our findings reveal that for unnatural images, a minimum phase range of 5π in the blue channel is sufficient, and for natural images, a phase difference of 4π is adequate. We find this result unsurprising as unnatural images have largely unique color channels while natural images have fairly similar color channels. This suggests the task of optimizing an SLM pattern for natural images is easier and consequently requires less phase.

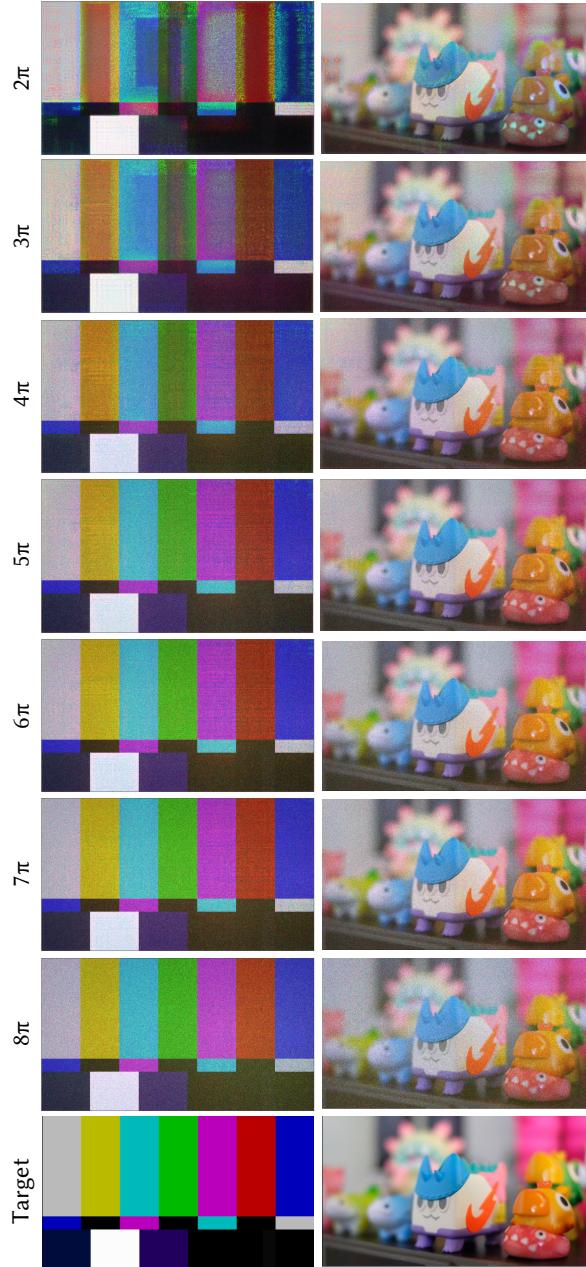


Fig. S9. A demonstration of SLM phase range on hologram quality in simulation] The images in the first column represent unnatural images, while the second column shows natural images. Each row corresponds to a specific maximum phase range in the blue channel, denoted by the labels of π . The phase range in the red channel is fixed at 2π , and the green channel's phase range is the average of the red and blue channels. Our findings demonstrate that a minimum phase range of 5π for unnatural images and 4π for natural images in the blue channel produces satisfactory holographic reconstructions. This is anticipated given that unnatural images typically have more unique color channels, thus necessitating a larger phase range, while natural images with their similar color channels require less phase variation.

S8 BIT AND DEPTH DIVISION IMPLEMENTATION DETAILS AND ANALYSIS

In this section we provide our implementation details of bit and depth division holography. Additionally, we analyze the methods for SLMs of various phase ranges. We implement bit division largely as laid out by Jesacher et al. [2014]. First we calculated the three color channels SLM patterns using a modified Gerchberg-Saxton approach assuming a 2π phase range in each color channel. Instead of using the Fourier transform for propagation as in Jesacher et al. [2014], we use ASM match our other results. This is run until convergence, and 3 unique SLM patterns are produced. These SLM patterns are then combined via an optimization problem as described by [Jesacher et al. 2014]. We then used the combined SLM pattern to simulate a color hologram at the sensor plane. Pseudocode for this algorithm is provided by Algorithm 2.

Algorithm 2 Bit Division Holography

```

1: Step 1: Individual color channel SLM pattern calculation
2: for each color channel (Red, Green, Blue) do
3:   Get the target intensity for the current color channel:  $I_{target}$ 
4:   Initialize the hologram phase pattern for the current color channel:  $H_0$ 
5:    $A = Amplitude(Source) * e^{iH_0}$ 
6:   while not converged do
7:     Perform the modified Gerchberg-Saxton algorithm for the current color channel:
8:      $B = angularSpectrumPropagation(A, d_{color}, \lambda_{color})$ 
9:      $C = Amplitude(I_{target}) * e^{iPhase(B)}$ 
10:     $D = angularSpectrumPropagation(C, d_{color}, \lambda_{color})$ 
11:     $A = Amplitude(Source) * e^{iPhase(D)}$ 
12:   Store the optimized hologram for the current color channel:  $H_{color}$ 
13: Step 2: Hologram combination
14: Combine the three optimized holograms by solving the optimization problem described by Jesacher et al. [2014]
15: Step 3: Hologram simulation
16: Simulate a color hologram at the sensor plane using the combined hologram
17: Output the final optimized holograms

```

We choose to implement the depth division method using gradient descent-based optimization rather than a modified Gerchberg-Saxton (GS) algorithm for multiplane holograms originally proposed by Makowski et al. [2010, 2008] for depth division holography. Since we use gradient descent in our approach, we determined this was a more fair comparison. In our implementation the SLM pattern is first converted to a complex field. The complex field is then propagated to $depth = 68$ mm, 80 mm, 100 mm using the ASM kernel for the red color channel. These correspond to the replica planes. The intensity of the fields are then calculated at each target plane and compared to the blue, green, and red channels, respectively, using an L2 loss function. Backpropagation is then used to calculate the gradients of the loss function with respect the SLM voltage values and then update these voltages. Pseudocode for this algorithm is provided by Algorithm 3.

We implement both the bit and depth division holography methods for 3 simulated SLMs. The first SLM has a uniform 2π phase range in each color channel. This phase range is optimal for depth division, but performs the worst of the simulated SLMs for bit division, demonstrating how bit division relies on extended SLM phase range. The next simulated SLM is an arbitrary standard SLM i.e. not extended phase. We model this SLM to have 2π phase range in red, 2.7π in green, and 3.4π in blue. The simulated holograms increase in quality from the 2π SLM for the bit division method, but decrease in quality for depth division. Finally we simulate the Holoeye Pluto SLM

Algorithm 3 Depth Division Holography

```

1: Initialization:
2: - Choose a depth to display the multicolor hologram:  $d_{target}$ 
3: Step 1: Calculate depth planes
4: Set the red channel propagation distance,  $d_{red}$ , equal to the target depth:  $d_{red} = d_{target}$ 
5: Determine the ratio  $r$  of the propagation distance to the wavelength for the red color:  $r = d_{red}/\lambda_{red}$ 
6: Apply this ratio to the blue and green wavelengths to calculate the propagation distances (depth planes)
   where their angular spectrum kernel equals that of the red color:  $d_{blue} = r * \lambda_{blue}$ ,  $d_{green} = r * \lambda_{green}$ 
7: Step 2: Initialize multiplane hologram optimization
8: Initialize parameters for gradient descent optimization, including learning rate, number of iterations, and
   initial hologram
9: Step 3: Gradient descent optimization
10: for iteration from 1 to numIterations do
11:   Initialize target intensities for red, green, and blue
12:   for d in  $[d_{red}, d_{green}, d_{blue}]$  do
13:     Propagate the field using angularSpectrumPropagation:  $propagatedField = angularSpectrumPropagation(currentHologram, d, \lambda_{red})$ 
14:     Compute targetIntensity from propagatedField
15:     Append targetIntensity to respective color targetIntensities
16:   Compute the loss:  $loss = computeLoss(target, targetIntensities)$ 
17:   print("Iteration:", iteration, "Loss:", loss)
18:   Compute the gradient:  $gradient = computeGradient(target, targetIntensities, currentHologram)$ 
19:   Update currentHologram:  $currentHologram = currentHologram - learningRate * gradient$ 
20: Step 4: Return the optimized multiplane hologram
21: Output the final optimized hologram: currentHologram

```

used in our experimental setup. This SLM has a 2.4π phase range in red, 5.9π phase range in green, and 7.4π phase range in blue. The results for depth division continue to degrade with this SLM, since the depth division algorithm does not take into account the wavelength-dependent response of the SLM. The results improve for bit division with the additional extended phase. This suggests that that phase diversity across channels provides the best performance for bit division holography, while phase uniformity across channels provides the best performance for depth division holography. The results of the outlined experiment can be found in Figs. S11 and S12.

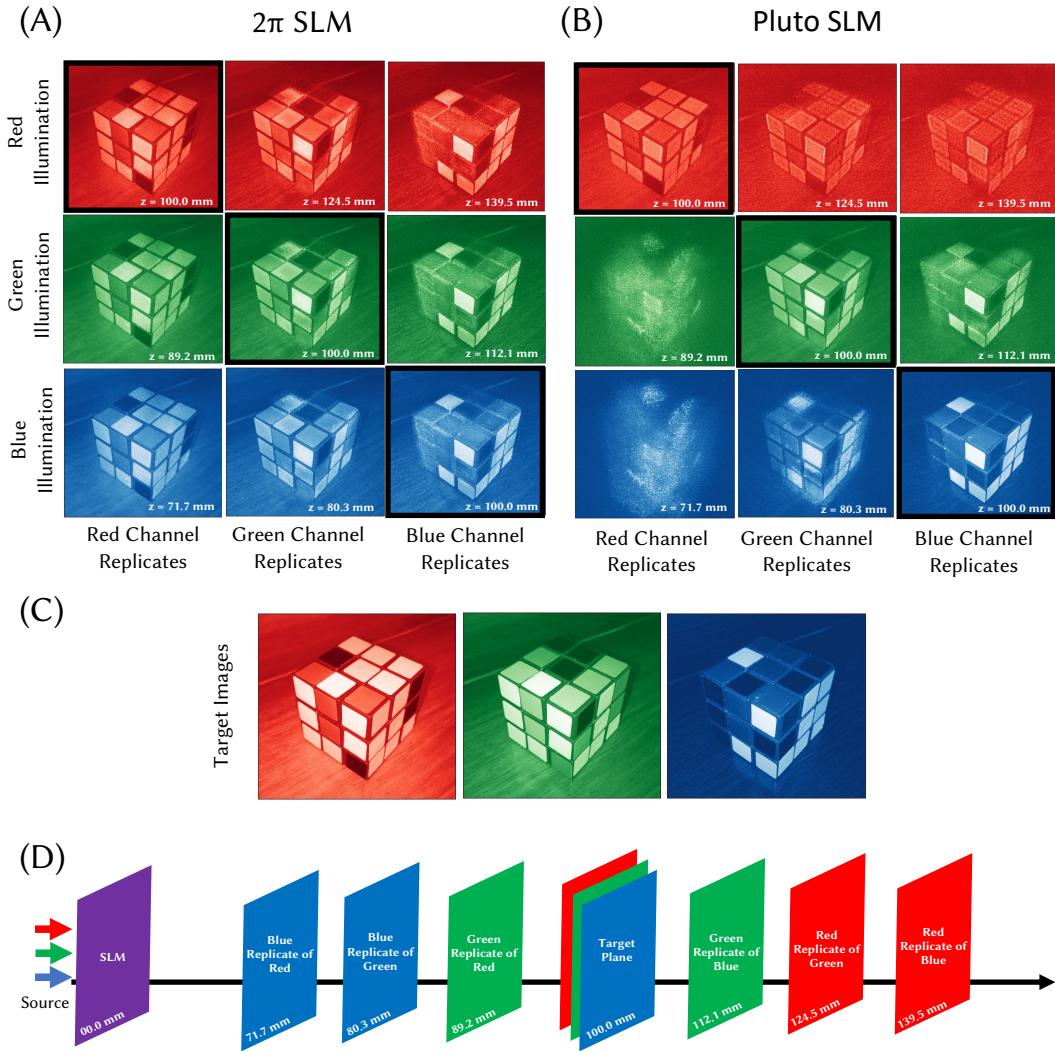


Fig. S10. Extended phase range reduces depth replicas in simulation. (A) Using an SLM with a uniform 2π phase range across all channels leads to strong depth replicas, which reduce image quality at the target plane (diagonal) compared to the targets (C) and add in-focus content at depths that should be defocused. By using the extended phase Holoeye Pluto-2.1-Vis-016 SLM (with Red: 2.4π , Green: 5.9π , Blue: 7.4π phase ranges), depth replicas are significantly reduced (B), improving the quality of target plane holograms and creating defocused content at other depths. (D) Schematic illustrating the positions of the replicate planes and target plane. Rubik's cube source image by Iwan Gabovitch (CC BY 2.0).

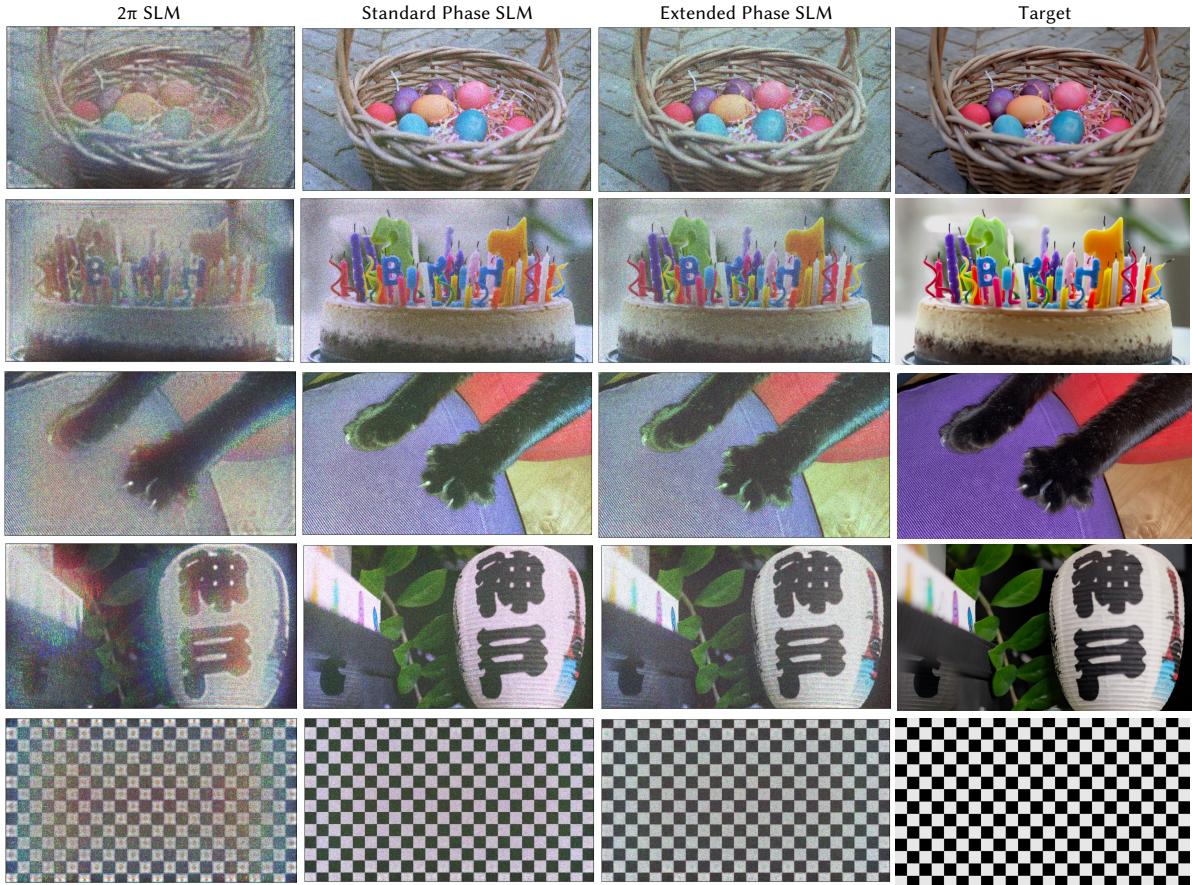


Fig. S11. SLM phase range affects hologram quality for bit division holography. Bit division takes advantage of the extended phase range of the SLM, so does not perform well with an SLM with only 2π phase range per channel (left column). With a “standard” SLM with realistic wavelength dependence to the phase, bit division performs better. It works best with the extended phase range of the simulated Holoeye Pluto that we use for our experiments.

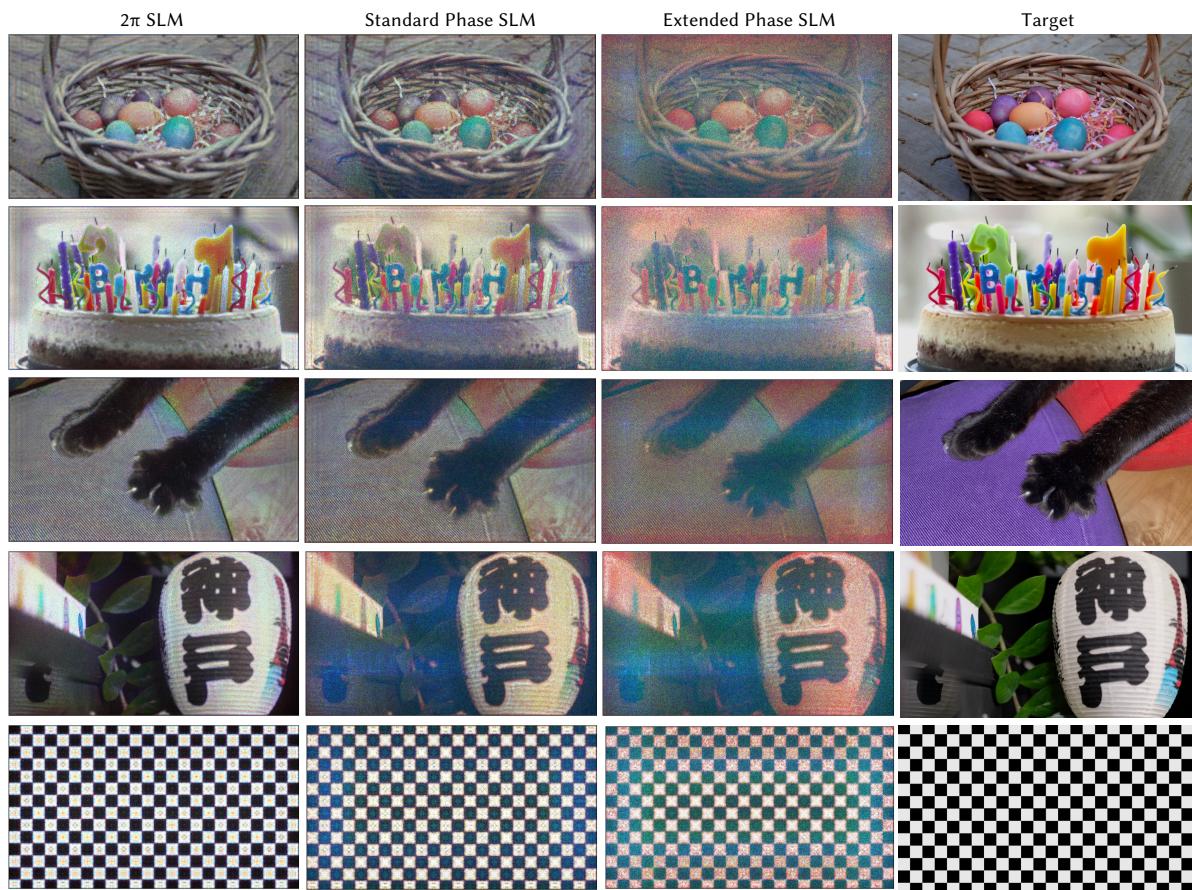


Fig. S12. SLM phase range affects hologram quality for depth division holography. The depth division approach assumes no wavelength dependence of the SLM, which is simulated in the first column. With a standard SLM with 2π phase in red and realistic wavelength dependence (second column) the results are slightly degraded due to the violation of the no wavelength dependence assumption. Finally, with the extended phase range of the simulated Holoeye Pluto SLM, the results show significant color artifacts and noise.

REFERENCES

- Manu Gopakumar, Jonghyun Kim, Suyeon Choi, Yifan Peng, and Gordon Wetzstein. 2021. Unfiltered holography: optimizing high diffraction orders without optical filtering for compact holographic displays. *Optics Letters* 46, 23 (2021), 5822. <https://doi.org/10.1364/ol.442851>
- Alexander Jesacher, Stefan Bernet, and Monika Ritsch-Marte. 2014. Colour hologram projection with an SLM by exploiting its full phase modulation range. *Optics Express* 22, 17 (8 2014), 20530. <https://doi.org/10.1364/oe.22.020530>
- Andrew Maimone, Andreas Georgiou, and Joel S Kollin. 2017. Holographic near-eye displays for virtual and augmented reality. *ACM Transactions on Graphics (TOG)* 36, 4 (2017), 85.
- Michał Makowski, Izabela Ducin, Maciej Sypek, Agnieszka Siemion, Andrzej Siemion, Jarosław Suszek, and Andrzej Kolodziejczyk. 2010. *Color image projection based on Fourier holograms*. Technical Report 8. 1227 pages.
- Michał Makowski, Maciej Sypek, and Andrzej Kolodziejczyk. 2008. *Colorful reconstructions from a thin multi-plane phase hologram*. Technical Report.
- Kathy T Mullen. 1985. The contrast sensitivity of human colour vision to red-green and blue-yellow chromatic gratings. *The Journal of physiology* 359, 1 (1985), 381–400.
- Yifan Peng, Suyeon Choi, Nitish Padmanaban, and Gordon Wetzstein. 2020. Neural holography with camera-in-the-loop training. *ACM Transactions on Graphics* 39, 6 (11 2020). <https://doi.org/10.1145/3414685.3417802>