The Journal of Open Source Software

pySLM2: A full-stack python package for holographic beam shaping

Chung-You Shih [1¶], **Jingwen Zhu** [1], **and Rajibul Islam** [1]

1 Institute for Quantum Computing and Department of Physics and Astronomy, University of Waterloo, 200 University Ave. West, Waterloo, Ontario N2L 3G1, Canada ¶ Corresponding author

DOI: 10.21105/joss.06315

Software

- Review
- Repository
- Archive

Editor: Rohit Goswami

Reviewers:

- @brandondube
- @sidihamady
- @wavefrontshaping

Submitted: 13 December 2023
Published: 11 December 2024

License
Authors of papers retain copyright and release the work under a Creative Commons Attribution 4.0 International License (CC BY 4.0).

Summary

Holographic beam shaping using spatial light modulators (SLMs) as a reprogrammable hologram offers a powerful tool for precise and flexible optical controls. It has been adopted for a wide range of researches, including atom trapping (Gaunt & Hadzibabic, 2012), optical addressing of individual quantum objects (Motlakunta et al., 2024), preparation of exotic quantum states (Islam et al., 2015), and multi-beam laser machining (Obata et al., 2010).

pySLM2 is a python package designed for holographic beam shaping applications, encompassing hologram generation, simulation, and hardware controls.

The package implements the hologram generation algorithms of the Lee hologram (Lee, 1978) and its improved alternatives (Shih et al., 2021; Zupancic et al., 2016), specifically targeting the digital micromirror device (DMD) based SLM with binary amplitude controls. It also implements the Gerchberg-Saxton algorithm (Gerhberg & Saxton, 1972) and its improved alternatives(Gaunt & Hadzibabic, 2012; Pasienski & DeMarco, 2008) suitable for liquid crystal on silicon (LCoS) based SLMs with pure phase controls.

At its core, the package uses `TensorFlow` (TensorFlow Developers, 2024) for numerical computations. By leveraging `TensorFlow`, the package harnesses the power of GPUs for faster computation without the need for code modification. This results in a significant speed-up for algorithms that are computationally expensive but benefit from parallelization, such as many hologram generation algorithms relying on iterative Fourier transformations.

In addition to hologram generation, the package provides functions to simulate beam profiles created by holograms, aiding users in evaluating algorithm performance. It also includes a variety of pre-defined optical profiles, such as Hermite-Gaussian, Laguerre-Gaussian, super-Gaussian, and Zernike polynomials, enabling users to construct target beam profiles with ease.

For hardware control, pySLM2 offers a unified application interface (API) compatible with various SLMs, ensuring seamless adaptation of code across different devices. Currently, the package supports DMD controllers from two commercial vendors: Visitech, INC and ViALUX GmbH.

Statement of need

High-quality optical controls are crucial for numerous scientific and engineering applications. For instance, in atom-based quantum information processors, quantum states of individual atoms are often manipulated by individually addressing laser beams. The quality of these addressing beams directly impacts the fidelity of quantum operations (Motlakunta et al., 2024).

Holographic beam shaping using SLMs provides a way for precise and adaptive optical controls. Compared to using conventional optical elements, holographic beam shaping has several advantages. Firstly, it can generate arbitrary beam profiles that are challenging to create with standard optical elements. For example, the Laguerre-Gaussian beam with a non-zero azimuthal index (often referred to as a doughnut beam), which can be used to trap atoms in a tube-like potential (Kuga et al., 1997), apply angular momentum to Bose-Einstein Condensate (Andersen et al., 2006), or achieve super-resolution imaging (Drechsler et al., 2021; Qian et al., 2021).

Secondly, holographic beam shaping can actively correct cumulative optical aberrations in the system arising from almost inevitable causes, such as surface irregularities, misalignment, and imperfect lens curvature, thereby achieving diffraction-limited performance at the target image plane. This enables the faithful production of target beam profiles with high accuracy, relieving the stringent requirements on optics quality and alignment precision. It has been shown that residual wavefront aberrations can be corrected to less than $\lambda/20$ root-mean-square (RMS) (Shih et al., 2021; Zupancic et al., 2016), which meets the Maréchal criteria (wavefront RMS error $< \lambda/14$) for optical performance.

At the time of writing, the `pySLM2` package, as detailed in this manuscript, has been used in the trapped ion quantum information processing researches (Kotibhaskar et al., 2023; Motlakunta et al., 2024; Shih et al., 2021). The authors believe that the package will benefit a broader community of researchers and engineers by offering turnkey solutions for applying holographic beam shaping to their work. Moreover, the primitives included in the package can assist researchers in rapidly prototyping new hologram generation algorithms.

There are existing open-source packages available that specialize in different levels of holographic beam shaping with SLMs. For example, `SLMLayout` is a package focusing on wavefront shaping with macropixel method (Matthès et al., 2019) with DMDs. `CGH-diff` offers `TensorFlow` scripts for an automatic differentiation-based algorithm (Yamamoto & Ochiai, 2021) for DMD hologram generation. `slmsuite` offers a comprehensive suite of hologram algorithms tailored for phase-only Spatial Light Modulators (SLMs), supported by GPU acceleration via `CuPy` (Okuta et al., 2017), and includes hardware control capabilities for Liquid Crystal on Silicon (LCoS) SLMs.

`pySLM2` implements hologram generation algorithms and primitives leveraging `TensorFlow` for GPU acceleration. Some algorithms, including the algorithms used in authors' prior works(Motlakunta et al., 2024; Shih et al., 2021) in the field of quantum information processing, are only available in `pySLM2` at the time of writing. In terms of hardware controls, `pySLM2` offers a unified API for programming DMDs from different vendors.

We hope that the addition of `pySLM2` to the community will facilitate researchers in applying Fourier holographic beam shaping ideas on various machines with fewer hardware hurdles.

Fourier Holography Basics

`pySLM2` is designed for holographic beam shaping using Fourier holography. The name "Fourier" comes from the fact that the electric fields of the beam at the hologram plane and the target plane are related by a Fourier transformation.

In a paraxial lens system, the lens act as a Fourier transform operator mapping the electric field in one focal plane to the electric field in the another focal plane. In the context of Fourier Holography, the two focal planes are referred as the image plane (IP) and the Fourier plane (FP). The electric fields at the two planes, $E_{\mathrm{IP}}(\mathbf{x}')$ and $E_{\mathrm{FP}}(\mathbf{x})$ respectively, are related by the following equation:

$$E_{\mathrm{FP}}(\mathbf{x})\mathrm{e}^{\mathrm{i}\Phi_{ab}} = \frac{\lambda f}{2\pi}\mathcal{F}\left[E_{\mathrm{IP}}(\mathbf{x}')\right](\mathbf{k}')\Big|_{\mathbf{k}'=\frac{2\pi}{\lambda f}\mathbf{x}}$$

In which, $\mathbf{x}'$ and $\mathbf{k}'$ denote the spatial coordinate and the wave vector at the image plane respectively, and $\mathcal{F}$ denotes Fourier transformation. The wave vector $\mathbf{k}'$ is related to the spatial coordinate $\mathbf{x}$ at FP by $\mathbf{x} = \frac{\lambda f}{2\pi}\mathbf{k}'$ where f is the effective focal length of lens and λ is the wavelength of the light.

The aberrations of the optical system can be modeled as a phase map Φ_{ab} in the Fourier plane. In pySLM2's convention, the plane SLM is placed is Fourier plane, and the image plane is where the targeted beam profile is desired. The SLM modulates the beam at Fourier plane to engineer the desired beam profiles at the image plane.

Hologram Generation Algorithm

Currently, pySLM2 supports two type of the spatial light modulator (SLM), liquid crystal on silicon (LCoS) SLM and digital micromirror device (DMD). The LCoS SLM modulates the phase profile purely without modifying the amplitude. As the time of writing, Gerchberg-Saxton (GS) (Gerhberg & Saxton, 1972) algorithm and the mixed-region amplitude freedom (MRAF) algorithm (Gaunt & Hadzibabic, 2012; Pasienski & DeMarco, 2008) are included.

On the other hand, DMDs use micromirrors to locally turn on and off the light by toggling the micromirrors between two directions. This allows binary amplitude control. By periodically turning on and off the micromirrors across the DMD to form grating patterns, diffracted beams with controllable phase and amplitude can be engineered to have the desired beam profiles. As the time of writing, a randomized algorithm (Zupancic et al., 2016) and an iterative Fourier transformation algorithm (Motlakunta et al., 2024; Shih et al., 2021) are provided for hologram generation.

Usages

pySLM2 offers commonly used optics profiles right out of the box, including Hermite Gaussian, Laguerre Gaussian, super Gaussian (also known as "flat top"), and Zernike polynomials. These profiles are implemented as functional objects, and pySLM2 automatically handles the profile sampling during hologram calculations.

For profiles that are not included by default, users have the option to either inherit from the base class and implement their custom profiles or generate the sampled profiles in an array format to pass them to the hologram calculation function. As illustrated in Figure 1, here's an example of creating a hologram to generate a Laguerre Gaussian beam with a mode of $l = 1$, $p = 0$, which often referred to as a "doughnut beam", from the fundamental Gaussian mode. Unless specified, the simulation shown in this paper is simulated with the following conditions: $\lambda = 369\ \text{nm}$ wavelength, $f = 200\ \text{mm}$ Fourier lens focal length, and with Texas Instrument DLP9500 as the SLM ($1\ \text{px} = 10\ \mu\text{m}$ micromirror size).

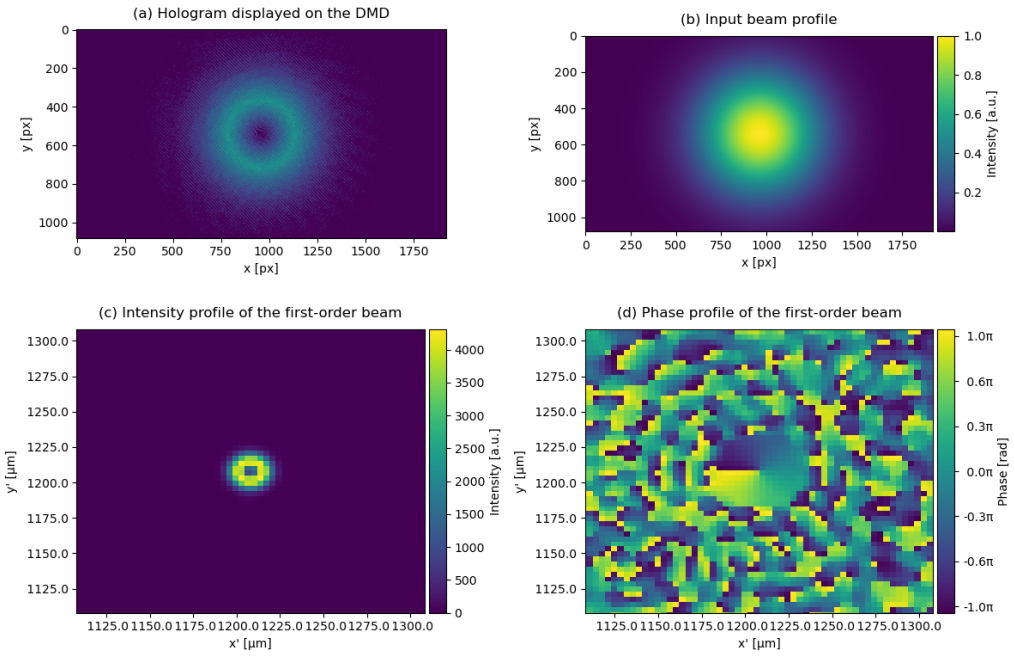

Figure 1: Hologram simulation for creating Laguerre Gaussian beam of $l = 1$, $p = 0$ mode from fundamental mode. (a) DMD mirror configuration. Bright pixels represent "on" and dark pixels represent "off". (b) Intensity profile of input fundamental Gaussian beam. (c) Intensity profile of the output Laguerre ($l = 1$, $p = 0$) Gaussian beam at the image plane. (d) Phase map of the output beam. An optical vortex can be observed at the center of the Laguerre $l = 1$, $p = 0$ mode (Source code: `examples/create_donut_beam.py`)

The arithmetic operations of the profiles are also overloaded, so one can easily combine different profiles through addition or rescale the profiles through multiplication. Shown in Figure 2, we create a hologram to generate two Gaussian beams. In the source code, it is written as adding two Gaussian profiles together at different positions.

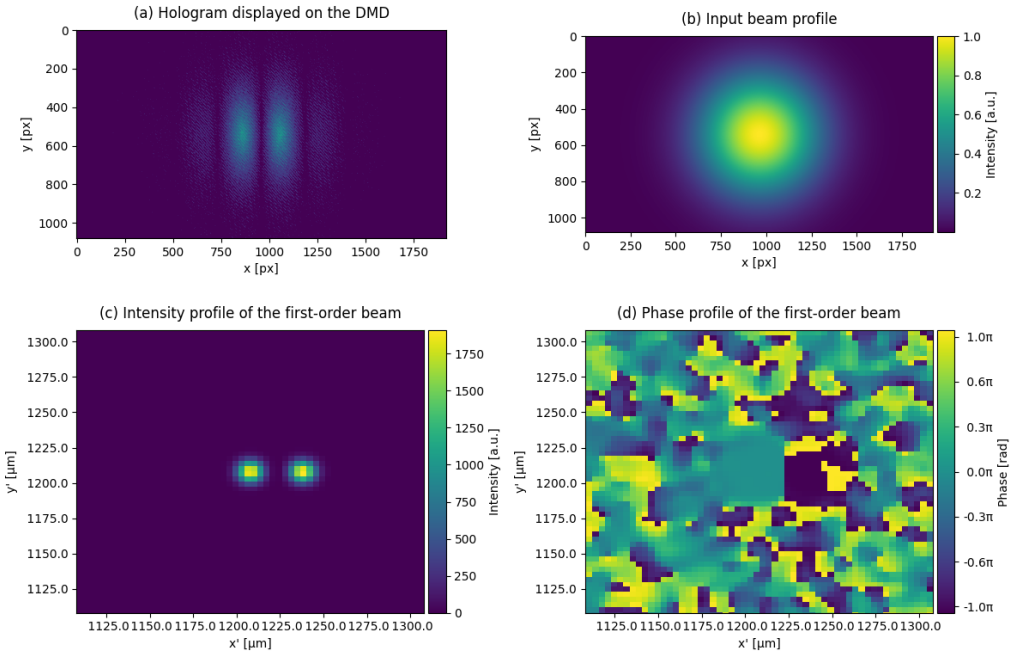

Figure 2: Hologram simulation for creating two Gaussian beams from one input Gaussian beam. (a) DMD mirror configuration. Bright pixels represent "on" and dark pixels represent "off". (b) Intensity profile of input single Gaussian beam. (c) Intensity profile of the two output Gaussian beams at the image plane. (d) Phase map of the output beam. An example of two Gaussian beams having opposite phases is shown. (Source code: `examples/create_donut_beam.py`)

Aberration Correction

One of the key advantages of holographic beam shaping is its capability to correct optical aberrations, and pySLM2 provides an easy method to achieve this correction. By supplying the aberration information during the hologram calculation, pySLM2 generates a hologram imprinted with a phase profile opposite to the aberration, effectively canceling the aberration out.

In the example depicted in Figure 3, we simulate the beam profile at the image plane both with and without aberration correction. Without aberration correction, the beam profile becomes distorted and broadened. In this particular simulation, spherical aberration is used, but pySLM2 is capable of correcting other types of aberrations as well.

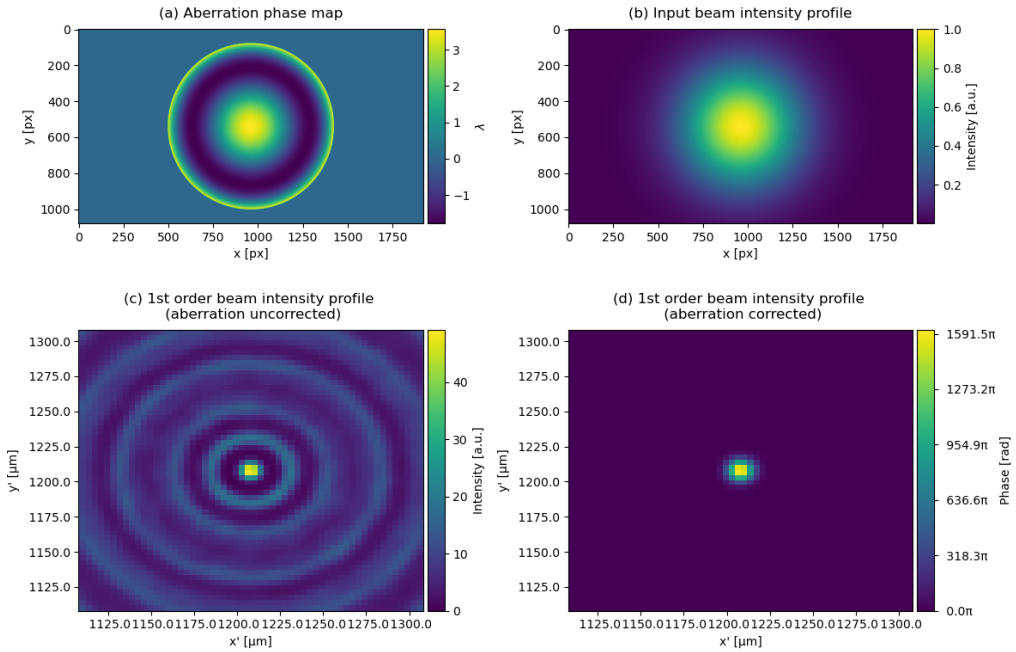

Figure 3: Simulation of the beam profiles at the image plane with and without aberration correction. (a) Phase map of the input beam with $\mathbb{Z}_4^0$ spherical aberration. (b) Intensity profile of the input beam. (c) Intensity profile of the first order beam without aberration correction. (d) Intensity profile of the first order beam with aberration correction. (Source code: `examples/aberration_correction.py`)

To obtain the phase map of the aberration, one can either use a wavefront sensor, such as a Shack–Hartmann sensor (Paschotta; Shack, 1971), to measure the wavefront, or one can allow light from different parts of the Fourier plane to interfere with each other to reconstruct the aberration phase profile from the resulting interference patterns. For a detailed description of the latter method, one can refer to Shih et al. (2021).

Hardware Controls

pySLM2 provides hardware controls for DMD controllers from both Visitech, INC and ViALUX GmbH. The controllers from these two companies use different communication protocols and architectures. The Visitech controller uses UDP over Ethernet, while the ViALUX controller uses USB3.

One of the goals of pySLM2 is to abstract the hardware details and offer a unified application interface for interacting with these devices. For instance, we have implemented the same `load_single` and `load_multiple` functions within the controller classes for both manufacturers' devices. These functions allow for the display of single holograms or the loading of multiple holograms that can be switched by triggers. Apart from the hardware-agnostic functions, it also exposed the lower-level access for advanced users to implement device specific controls.

As of the current writing, the package's hardware support is limited to DMD controllers. For users interested in using LCoS-SLM, open-source tools such as `slmsuite` and `slmPy` are available options.

Author Contributions

C.-Y.S. designed and implemented the package. J.Z. contributed to implementing the hardware controls and performance benchmarking. R.I. advised on the scientific aspects of the package.

All authors contributed to writing the manuscript and documenting the package.

Acknowledgements

The hardware controls for the DMDs from ViALUX GmbH in the package is built on top of the `AL4lib` (Popoff et al., 2022). We appreciate the work of the authors of `AL4lib`. We express gratitude to Kaleb Ruscitti for assisting with hardware testing and to Sainath Motlakunta and Nikhil Kotibhaskar for providing valuable feedback on the package.

We acknowledge financial support from the Natural Sciences and Engineering Research Council of Canada (NSERC) Discovery program (RGPIN-2018-05250) and the Institute for Quantum Computing for this work.

References

Andersen, M., Ryu, C., Cladé, P., Natarajan, V., Vaziri, A., Helmerson, K., & Phillips, W. D. (2006). Quantized rotation of atoms from photons with orbital angular momentum. *Physical Review Letters*, *97*(17), 170406. https://doi.org/10.1103/PhysRevLett.97.170406

Drechsler, M., Wolf, S., Schmiegelow, C. T., & Schmidt-Kaler, F. (2021). Optical superresolution sensing of a trapped ion's wave packet size. *Physical Review Letters*, *127*(14), 143602. https://doi.org/10.1103/PhysRevLett.127.143602

Gaunt, A. L., & Hadzibabic, Z. (2012). Robust digital holography for ultracold atom trapping. *Scientific Reports*, *2*(1), 721. https://doi.org/10.1038/srep00721

Gerhberg, R., & Saxton, W. (1972). A practical algorithm for the determination of phase from image and diffraction plane picture. *Optik*, *35*, 237–246. https://web.archive.org/web/20220505015731/http://www.u.arizona.edu/~ppoon/GerchbergandSaxton1972.pdf

Islam, R., Ma, R., Preiss, P. M., Eric Tai, M., Lukin, A., Rispoli, M., & Greiner, M. (2015). Measuring entanglement entropy in a quantum many-body system. *Nature*, *528*(7580), 77–83. https://doi.org/10.1038/nature15750

Kotibhaskar, N., Shih, C.-Y., Motlakunta, S., Vogliano, A., Hahn, L., Chen, Y.-T., & Islam, R. (2023). Programmable XY-type couplings through parallel spin-dependent forces on the same trapped ion motional modes. *arXiv Preprint arXiv:2307.04922*. https://doi.org/10.1103/PhysRevResearch.6.033038

Kuga, T., Torii, Y., Shiokawa, N., Hirano, T., Shimizu, Y., & Sasada, H. (1997). Novel optical trap of atoms with a doughnut beam. *Physical Review Letters*, *78*(25), 4713. https://doi.org/10.1103/PhysRevLett.78.4713

Lee, W.-H. (1978). III computer-generated holograms: Techniques and applications. In *Progress in optics* (Vol. 16, pp. 119–232). Elsevier. https://doi.org/10.1016/S0079-6638(08)70072-6

Matthès, M. W., Del Hougne, P., De Rosny, J., Lerosey, G., & Popoff, S. M. (2019). Optical complex media as universal reconfigurable linear operators. *Optica*, *6*(4), 465–472. https://doi.org/10.1364/OPTICA.6.000465

Motlakunta, S., Kotibhaskar, N., Shih, C.-Y., Vogliano, A., McLaren, D., Hahn, L., Zhu, J., Hablützel, R., & Islam, R. (2024). Preserving a qubit during state-destroying operations on an adjacent qubit at a few micrometers distance. *Nature Communications*, *15*(1), 6575. https://doi.org/10.1038/s41467-024-50864-2

Obata, K., Koch, J., Hinze, U., & Chichkov, B. N. (2010). Multi-focus two-photon polymerization technique based on individually controlled phase modulation. *Optics Express*, *18*(16), 17193–17200. https://doi.org/10.1364/OE.18.017193

Okuta, R., Unno, Y., Nishino, D., Hido, S., & Loomis, C. (2017). CuPy: A NumPy-compatible library for NVIDIA GPU calculations. *Proceedings of Workshop on Machine Learning Systems (LearningSys) in the Thirty-First Annual Conference on Neural Information Processing Systems (NIPS)*. http://learningsys.org/nips17/assets/papers/paper_16.pdf

Paschotta, R. *Shack–hartmann wavefront sensors*. RP Photonics Encyclopedia; RP Photonics AG. https://doi.org/10.61835/jcv

Pasienski, M., & DeMarco, B. (2008). A high-accuracy algorithm for designing arbitrary holographic atom traps. *Optics Express*, *16*(3), 2176–2190. https://doi.org/10.1364/OE.16.002176

Popoff, S. M., Shih, G., B., D., & GustavePariente. (2022). *Wavefrontshaping/ALP4lib: 1.0.1* (Version 1.0.1). Zenodo. https://doi.org/10.5281/zenodo.6121191

Qian, Z.-H., Cui, J.-M., Luo, X.-W., Zheng, Y.-X., Huang, Y.-F., Ai, M.-Z., He, R., Li, C.-F., & Guo, G.-C. (2021). Super-resolved imaging of a single cold atom on a nanosecond timescale. *Physical Review Letters*, *127*(26), 263603. https://doi.org/10.1103/PhysRevLett.127.263603

Shack, R. V. (1971). Production and use of a lenticular hartmann screen. *Spring Meeting of Optical Society of America, 1971*, *656*. https://doi.org/10.1364/JOSA.61.000648

Shih, C.-Y., Motlakunta, S., Kotibhaskar, N., Sajjan, M., Hablützel, R., & Islam, R. (2021). Reprogrammable and high-precision holographic optical addressing of trapped ions for scalable quantum control. *Npj Quantum Information*, *7*(1), 57. https://doi.org/10.1038/s41534-021-00396-0

TensorFlow Developers. (2024). *TensorFlow* (Version v2.18.0). Zenodo. https://doi.org/10.5281/zenodo.13989084

Yamamoto, K., & Ochiai, Y. (2021). Gradient-based optimization of time-multiplexed binary computer-generated holograms by digital mirror device. *Digital Holography and Three-Dimensional Imaging*, DTh7C–1. https://doi.org/10.1364/DH.2021.DTh7C.1

Zupancic, P., Preiss, P. M., Ma, R., Lukin, A., Tai, M. E., Rispoli, M., Islam, R., & Greiner, M. (2016). Ultra-precise holographic beam shaping for microscopic quantum control. *Optics Express*, *24*(13), 13881–13893. https://doi.org/10.1364/OE.24.013881