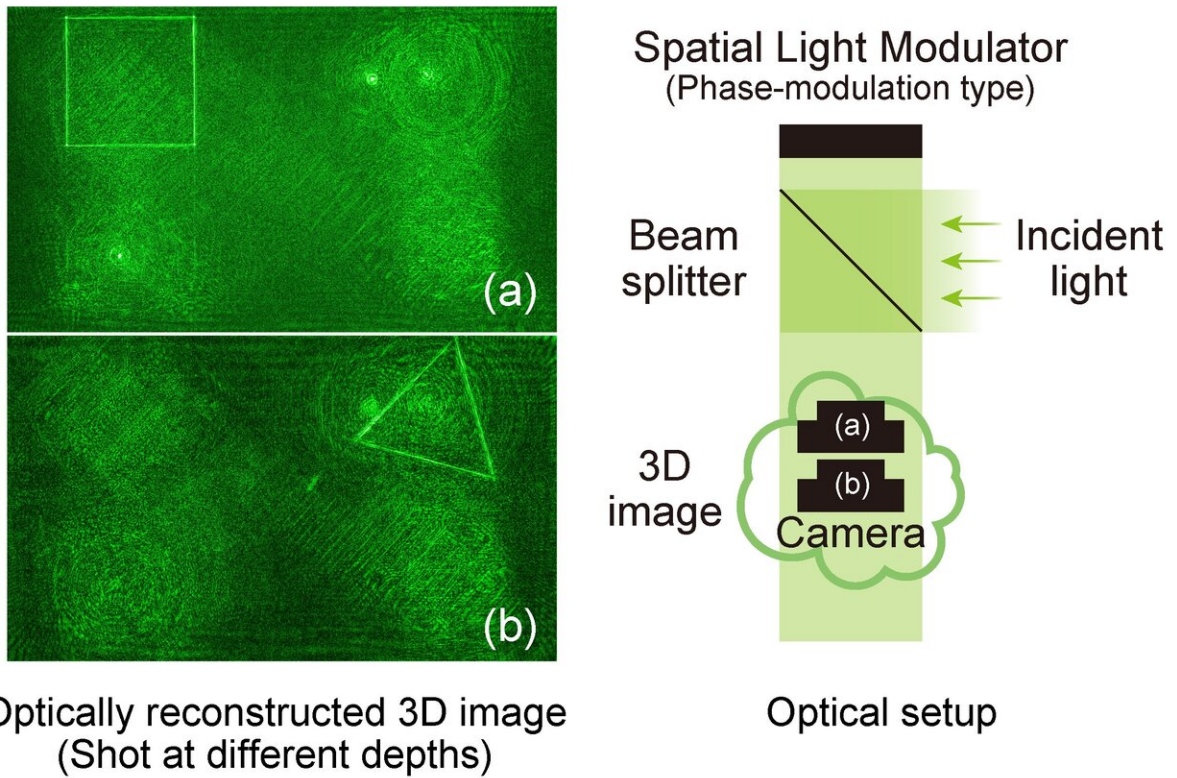


# Lightning-fast algorithms can lighten the load of 3-D hologram generation

June 8 2020



(left) Different images at depths (a) and (b) (see right) show how the distribution of light over space forms a truly 3D image. (right) Schematic of holography setup. The calculated hologram is displayed on a spatial light modulator while laser light is directed to reflect off its surface, interfere with the original beam and form a 3D image at the camera. Credit: Tokyo Metropolitan University

Researchers from Tokyo Metropolitan University have developed a new way of calculating simple holograms for heads-up displays (HUDs) and near-eye displays (NEDs). The method is up to 56 times faster than conventional algorithms and does not require power-hungry graphics processing units (GPUs), instead running on normal PC computing cores. This opens the way to developing compact, power-efficient augmented reality devices, including 3-D navigation on car windshields and eyewear.

The term hologram may still have a sci-fi ring to it, but holography, the science of making 3-D records of light, is used everywhere, from microscopy, fraud prevention on banknotes to state-of-the-art data storage. Everywhere, that is, except for its most obvious potential application: truly 3-D displays. The deployment of 3-D displays that don't require special glasses has yet to become widespread. Recent advances include [virtual reality](#) (VR) technologies, but the vast majority rely on optical tricks that convince the human eye to see things in 3-D. This is not always feasible and limits its scope.

One of the reasons is that generating a hologram of arbitrary 3-D objects is a computationally heavy exercise. This makes every calculation slow and power-hungry, a serious limitation when you want to [display](#) large 3-D images that change in real time. The vast majority require specialized hardware like GPUs, the energy-guzzling chips that power modern gaming. This severely limits where 3-D displays can be deployed.

Thus, a team led by Assistant Professor Takashi Nishitsuji looked at how holograms were calculated. They realized that not all applications needed a full rendering of 3-D polygons. By solely focusing on drawing the edge around 3-D objects, they succeeded in significantly reducing the computational load of [hologram](#) calculations. In particular, they could avoid using fast-Fourier transforms (FFTs), the intensive math routines

powering holograms with full polygons.

The team combined simulation data with real experiments by displaying their holograms on a [spatial light modulator](#) (SLM) and illuminating them with [laser light](#) to produce a real 3-D image. At high resolution, they found that their method could calculate holograms up to 56 times faster, and that the images compared favorably to those made using slower, conventional methods. Importantly, the team only used a normal PC computing core with no standalone graphics processing unit, making the whole process significantly less resource hungry.

Faster calculations on simpler cores means lighter, more compact power-efficient devices that can be used in a wider range of settings. The team is pursuing the development of heads-up displays (HUDs) on car windshields for navigation, and even augmented reality eyewear to relay instructions on hands-on technical procedures, both exciting prospects for the near future.

**More information:** Takashi Nishitsuji et al, Fast calculation of computer-generated hologram of line-drawn objects without FFT, *Optics Express* (2020). [DOI: 10.1364/OE.389778](https://doi.org/10.1364/OE.389778)

Provided by Tokyo Metropolitan University

Citation: Lightning-fast algorithms can lighten the load of 3-D hologram generation (2020, June 8) retrieved 24 April 2024 from

<https://phys.org/news/2020-06-lightning-fast-algorithms-d-hologram.html>

<p>This document is subject to copyright. Apart from any fair dealing for the purpose of private study or research, no part may be reproduced without the written permission. The content is provided for information purposes only.</p>
--