

High-performance parallel computing for next-generation holographic imaging

Takashige Sugie¹, Takanori Akamatsu¹, Takashi Nishitsuji¹, Ryuji Hirayama¹, Nobuyuki Masuda², Hirotaka Nakayama³, Yasuyuki Ichihashi⁴, Atsushi Shiraki¹, Minoru Oikawa⁵, Naoki Takada⁵, Yutaka Endo¹, Takashi Kakue¹, Tomoyoshi Shimobaba¹ & Tomoyoshi Ito¹

¹ Chiba University

² Tokyo University of Science

³ National Astronomical Observatory of Japan

⁴ National Institute of Information and Communications Technology

⁵ Kochi University

*itot@faculty.chiba-u.jp

Holography is a leading method of recording and reproducing 3D images. The increasingly widespread availability of computers has encouraged the development of holographic 3D screens (electroholography). Although electroholography was first proposed a half-century ago, it has not been used in practical applications. A fundamental problem is the enormous volume of data that a hologram requires. Even modern computational power is inadequate to

process this volume of data in real time. The area from which the reconstructed image can be viewed is determined by the way in which the light diffracted by the hologram is spread, and this in turn depends on the pixel pitch of the hologram. A smaller pitch creates a wider viewing angle. At a pixel pitch of 1 μm , the viewing angle extends to 30°, thus making it practical for everyday applications. The pixel pitch of a typical current liquid crystal display is approaching this limit. However, a high-definition display device requires a large number of pixels, thus making processing challenging. At a 1 μm pixel pitch, even a display device that is 1 cm \times 1 cm in size would require 100 million (10^8) pixels. Our group has been pursuing a five-year project with the goal of realising an arithmetic circuit that is able to drive a video-rate 1 cm \times 1 cm computer-generated hologram of 10^8 pixels at a pixel pitch of 1 μm . In the course of this research, we have developed a special-purpose holography computing board by using eight large-scale field-programmable gate arrays (FPGAs). This computing board is far beyond the scope of current commercial offerings. We have also succeeded in achieving a parallel operation of 4,480 hologram calculation circuits on a single board. By clustering eight of these boards, we succeeded in increasing the number of parallel calculations to 35,840, thus allowing computations to be performed 1,000 times faster than those of a personal computer. By using a 3D image comprising 7,877 points, we succeeded in updating 10^8 -pixel holograms at a video rate, thus allowing 3D movies to be projected. We further demonstrated that the system speed scales up in a linear manner as the number of parallel circuits is increased. The system operates at 0.25 GHz with an effective speed equivalent to 0.5 Pflops (10^{15} floating-point operations per second), matching that of a high-performance computer. These results suggest that a holographic

3D image system can be constructed using currently available technology. In a further step, we will upgrade the system to a large-scale integration (LSI) circuit that is 1 cm × 1 cm in size by using existing technology. Coupling this LSI to a 10⁸-pixel display would create a chip dedicated to holography. Given that the computation of a hologram treats each pixel independently, a suitable arrangement of these dedicated chips could create a 3D video system of arbitrary size and shape (hemispherical, spherical, cylindrical, etc.). As one of immediate goals, we can create a wide 3D projection space by incorporating our dedicated chip into a head-mounted display.

Video holography (electroholography) was first demonstrated in 1990 ¹. As the field developed, it became clear that the main constraints on electroholography were the lack of high-definition display devices and the need for high-speed computation ². In the intervening period, the definition achievable by display devices has increased tenfold, from approximately 10 μm to nearly 1 μm, and is now approaching levels that make routine applications practical ³. However, as the precision of the holographic display increases, the computational load increases. For example, a 1 m × 1 m hologram with a pixel pitch of 1 μm would require 10¹² pixels compared with 10⁶ pixels of a typical 2D display (a 10⁶ times increase). When the cost of converting a 3D image to a hologram is taken into account, an increase of 10⁶ in computational power is required. Research on the development of holographic 3D image systems for practical uses has focused mainly on speeding up the processing time.

A number of algorithms have been proposed on the basis of techniques such as table lookup ⁴ or the difference method ⁵, and great progress has been made ⁶⁻⁷. However, it

is difficult to develop practical technologies simply by increasing the operating speed of the software. For the real-time processing of the enormous amount of information required, massively parallel and distributed computing systems are required. GPU computing has been the topic of active research in various fields since the early 2000s. Holographic computations are a good fit for GPU acceleration ⁸. In addition, multi-GPU systems using multiple GPU boards have been studied for a real-time reconstruction of electroholography ⁹⁻¹⁰. Although a multi-GPU system accelerates holographic calculations, Song et al. have also discussed that it is difficult to speed up in proportion to the number of GPUs ¹⁰.

As for dedicated hardware, a group at the MIT Media Lab developed a special-purpose computational board as a part of their holographic video system in the 1990s ¹¹⁻¹². Buckley has developed an error reduction system for a holographic projector ¹³. Seo et al. have studied their dedicated hardware design based on an architecture similar to our HORN (HOlographic ReconstructionN) ¹⁴. For digital holography, rather than electroholography, we have also developed another type of special-purpose computers named FFT-HORN ¹⁵⁻¹⁷. Designing a dedicated computer for digital holography is difficult because a fast Fourier transform (FFT) has to be implemented as a large-scale circuit. Cheng et al. have also developed an efficient FPGA-based digital holography system ¹⁸.

Since 1992, our group has been studying dedicated holography systems, with a focus on hardware. We developed a series of prototype machines named HORN-1 to HORN-4 by using hand wiring ¹⁹⁻²². In 2004, we introduced the large-scale FPGA board for HORN-5 using printed circuit boards ²³. HORN-6 was a cluster system of 16 HORN-5 boards and was able to run 20,000 circuits of hologram calculation in

parallel²⁴. HORN-7 had a reduced communication load because the hologram data was sent directly to the display device²⁵. For the realisation of a commercial image system capable of handling holography, a dedicated chip that integrates a calculation circuit with the display device represents a significant forward step. Hardware systems integrating the computing circuit and the display device at the board level have also been developed²⁶⁻²⁷. In 2012, we launched the HORN-8 project, with the goal of developing practical circuits for use in electroholography.

Features of HORN-8 system

HORN-8 system is a prototype for large-scale holographic calculation circuits, which was developed to show that the calculation speed can be boosted to a practical level in hardware. The large difference is observed when HORN-8 system is compared with the previous HORN-5 system that was developed using similar large-scale FPGA technology. After determining the basic specifications, we outsourced the HORN-5 board design to a vendor. We also used a commercial driver for communication between the host PC and the HORN-5 board. In contrast, the HORN-8 system's board was fabricated entirely within our research group so that we were able to obtain a system configuration better optimised for holography calculations.

Packing the circuits (calculation pipelines) more densely improves the computational speed and simplifies the system. In addition, the wires in the circuits become shorter, which facilitates high-speed calculations. However, it is difficult to determine the best possible wiring arrangement: using a long wire to bypass other wires limits or lowers the calculation speed. For the HORN-8 project, a board that adheres to the PCI-

Express standard adopted by recent PCs was developed. The board is 22 cm × 13 cm in size, smaller than the PCI-standard HORN-5 board (33 cm × 15 cm). We were able to mount eight FPGAs (twice as many as in HORN-5) on the smaller HORN-8 board by eliminating unnecessary elements and wiring. We adopted a ring bus for inter-FPGA communication, i.e. the communication signal lines pass through all the FPGAs. This ring bus architecture shortens the wire lengths and reduces the number of electronic parts; however, the board fails to operate even if one FPGA malfunctions.

The HORN-8 system is a great improvement over the previous systems. Compared with HORN-5, the HORN-8 board can execute five times as many operations per clock, and its communication speed is 15 times faster. These improvements have enabled us to evaluate a large-scale parallelisation of dedicated holographic circuits and discover the following features.

- (1) The effective performance of one board can give 99% or more of the theoretical performance, indicating that the transfer time can be hidden in the calculation time.
- (2) The speed of a cluster is proportional to the number of boards, with an effective performance that is 90% or more than its theoretical performance.
- (3) The HORN-8 system can calculate by dispersing holograms.

The above results show that hardware, such as HORN-8, can speed up holographic calculation almost in proportion to the number of parallel processes. With current technology, it would be possible to build the HORN-8 system on a chip that could be mass-produced. The HORN-8 system demonstrates that holographic calculations can be greatly accelerated by dedicated hardware, as shown in this study.

Hardware design and development

Figure 1 shows the HORN-8 board. Eight large-scale FPGA chips are mounted on a standard PCI-Express (Gen 1) board. Seven FPGAs (Xilinx Virtex5-XC5VLX 110) are used for calculation and one (Xilinx Virtex5-XC5VLX30T) for communication. A board that can house more than four large-scale FPGAs is not currently commercially available; thus, to support massively parallel and distributed processing, we designed and developed a custom board. This HORN-8 board ran at an operating frequency of 0.25 GHz and had a communication speed of 1.2 GB/s.

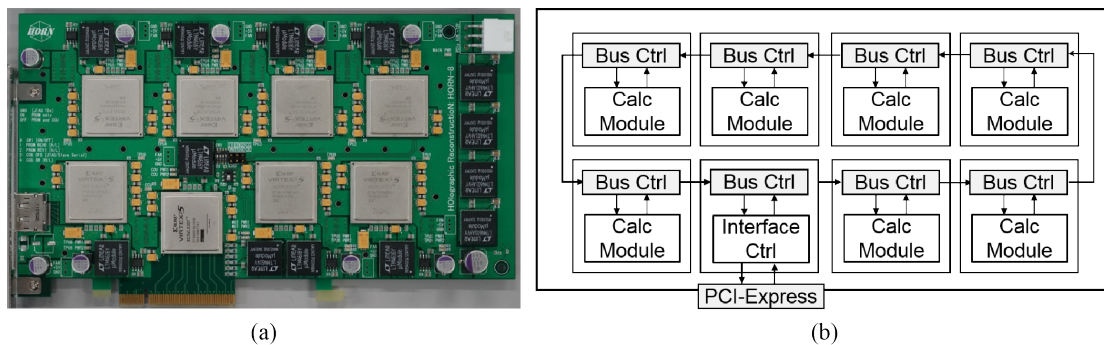


Figure 1. HORN-8 board: (a) Top view of the board and (b) the basic system. The board closely integrated seven computational FPGAs and one communication FPGA. For on-board communication, a ring bus was used. For communication between the board and the host computer, the PCI-Express standard was followed, and a direct memory access transfer circuit was used to achieve a speed equivalent to 75% of the 1.6 GB/s theoretical limit.

In this system, the 3D image is represented by a point cloud model. The hologram is amplitude type. The calculation complexity is $O(N_{\text{obj}} N_{\text{hol}})$, where N_{obj} and N_{hol} are

the numbers of object points and hologram pixels, respectively. The reconstructed image is see-through type. Although hidden surface processing is commonly used in 2D displays, it is being developed for electroholography²⁸. Hidden surface processing increases the computational cost; however, as the processing also decreases the number of object points, the cost is lower, depending on the model's shape. Hidden surface processing was thus not considered in this research.

Although several effective software algorithms have been proposed for the calculations, we adopted the point-cloud based method. For example, with the split lookup tables (S-LUT) method²⁹, the complex amplitude data in the horizontal (x) and vertical (y) directions are independently calculated beforehand and stored in tables. This reduces the computational complexity to $O(N_{\text{obj}} N_y + N_x N_y)$, where $N_x \times N_y = N_{\text{hol}}$, but the tables occupy a large amount of memory, thereby making this method unsuitable for hardware implementation.

The polygon method³⁰ generates a hologram from polygons using FFTs and has a computational complexity of $O(N_{\text{poly}} N_{\text{hol}} \log N_{\text{hol}})$, where N_{poly} is the number of polygons. The computational cost thus depends on the 3D model, and this method is effective for 3D models comprising large, flat polygons. However, for more complex models comprising small polygons, the calculation speed may be slower than the point-cloud based method³¹. In addition, it is difficult to implement large-scale FFT circuits in hardware in parallel.

The layer method³²⁻³³ uses depth information to divide the 3D model space into layers based on depth and uses FFTs to calculate diffraction patterns. Its computational complexity for generating a hologram is $O(N_{\text{layer}} N_{\text{hol}} \log N_{\text{hol}})$, where N_{layer} is the number of layers. The layer method can be effective for generating

holograms from complete spaces, such as real scenes that include the background. In contrast, the point-cloud based method is more effective for localised models, such as visualising numerical simulation or medical imaging data. Similar to polygon method, the layer method is accelerated using FFTs, thereby making it unsuitable for hardware implementation.

Hardware systems have the potential to achieve very high acceleration via mass production (high parallelisation). Therefore, we adopted the point-cloud based algorithm for HORN-8 system, as it is suitable for hardware implement.

In the point-cloud method, we apply the Fresnel approximation formula with the reference light as the plane wave:

$$I(x_\alpha, y_\alpha) = \sum_{j=1}^{N_{\text{obj}}} A_j \cos \left[2\pi \left(\frac{(x_\alpha - x_j)^2 + (y_\alpha - y_j)^2}{2\lambda z_j} \right) \right], \quad (1)$$

where index j represents an object point, A_j is the intensity of each point and λ is the wavelength of the reference light. By calculating the light propagating from object point one to N_{obj} , the value of one pixel α of the hologram is obtained. By applying Equation (1) to the entire hologram plane, a single hologram is generated. The HORN-8 system implemented Equation (2), which was derived from Equation (1). In this equation, the coordinate variables are normalised by the pixel pitch p of the hologram. All values of A_j are set to one and excluded from the equation.

$$I(X_\alpha, Y_\alpha) = \sum_{j=1}^{N_{\text{obj}}} \cos \left[2\pi \left(\frac{p}{2\lambda Z_j} (X_{\alpha j}^2 + Y_{\alpha j}^2) \right) \right] = \sum_{j=1}^{N_{\text{obj}}} \cos [2\pi \theta_{\alpha j}] \quad (2)$$

Equation (2) was designed to be appropriate for our hardware design. The coordinate variables became integers, and the phase $\theta_{\alpha j}$ of the next pixel on the hologram could be obtained simply by summing the difference ³⁴.

Figure 2 shows a block diagram of the implemented circuit. For the first pixel, Equation (2) derives the basic processing unit (BPU). Thereafter, the difference ($\Gamma = 2\Delta X_{\alpha j}$; $\Delta = p/2\lambda Z_j$) is summed to create the additional processing unit (APU). We implemented 1 BPU and 639 APUs in each FPGA and constructed a pipeline of 640 parallel operations. Therefore, the number of parallel operations on a single board was $640 \times 7 = 4,480$. Previous HORN computers had used a table (memory reference) for cosine operation. Given that the goal was to develop an LSI chip, we developed a new cosine calculator for HORN-8 to allow the pipeline to be memoryless. We devised a novel approximate algorithm for hologram generation and realised cosine operation by using two adder-subtractor gates ³⁵. All FPGA memories could be used both to input and output data, and the maximum number of input object points was 65,536 (2^{16}). When a larger number of object points needed to be addressed, the object was divided for calculation. If the refresh rate was sufficiently fast, it was recognised as a single object ³⁶.

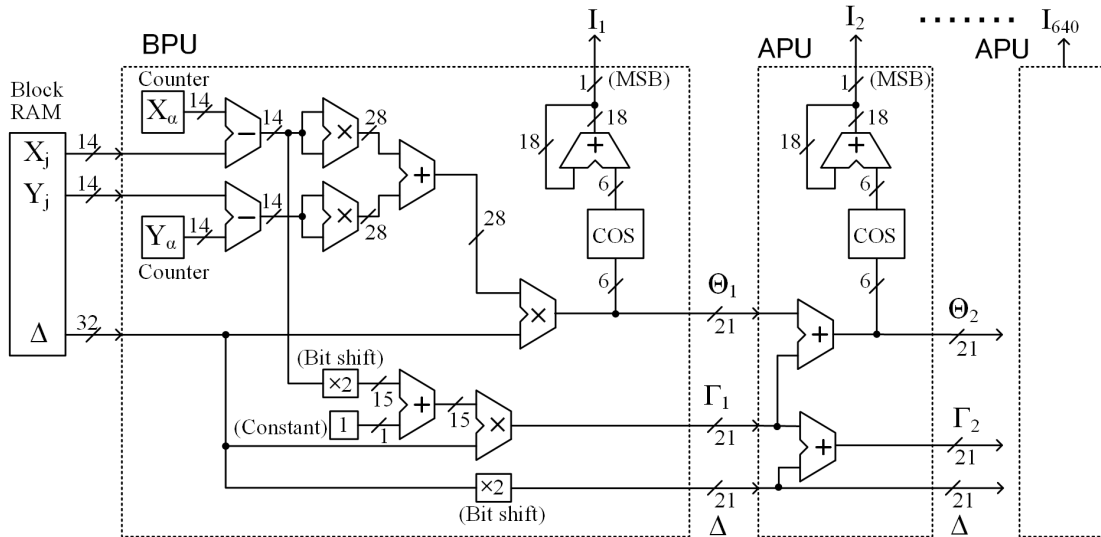


Figure 2. HORN-8 pipeline (one FPGA chip): One BPU and 639 APUs constitute the pipeline, which performs parallel computation. The BPU calculates the initial phase and the first pixel value. The APU calculates the subsequent pixel value on the basis of difference data. The development environment was Xilinx ISE 14.7. The usage rate of the slice (logic cell) was 98% and that of the on-chip memory (Block RAM) was 94%. The pipeline operated stably at 0.25 GHz.

Performance

Table 1 compares the HORN-8 board's performance with that of a CPU (Central Processing Unit) and a GPU. Considering that each pixel is calculated independently in holographic applications, the calculation time is proportional to the number of pixels. Here we compare the time required to generate a hologram of $1,920 \times 1,080$ (2 million pixels), which is the size of a standard liquid crystal display (LCD), from a 3D image containing 10,000 points. The speed of holographic calculation is proportional to the number of cores. A multi (or many) -core GPU has been shown to work

effectively at a rate 30 times that of a CPU. Despite its large number of cores, the dedicated circuit of the HORN-8 board functioned efficiently, achieving speeds 160 times those of a CPU and 6 times those of a GPU. Smooth image movement requires a frame rate of 10 fps (frames per second) or better.

Table 1. Performance comparison of the HORN-8 board, CPU, and GPU. This is the time taken to generate a hologram of $1,920 \times 1,080$ pixels from a 3D image of 10,000 points.

System	Time/CGH (sec)	Speed ratio	Frame rate (fps)
CPU: Core i7-6700K, 4cores, 4 GHz	2.951	1	0.34
GPU: GTX TITAN X, 3,072cores, 1 GHz	0.106	28	9.4
HORN-8board: 4,480 parallel (cores), 0.25 GHz	0.019	155	53

Table 2 shows the measurement results of the HORN-8 system. The total calculation time T_{total} can be expressed as

$$T_{total} = T_{in} + T_{horn} + T_{out} , \quad (3)$$

where T_{in} is the time taken to transfer the object data from the host PC to the HORN-8, T_{horn} is the calculation time for the HORN-8's FPGAs and T_{out} is the time taken to transfer the hologram data generated by the HORN-8 system.

The system was configured so that the calculation results were transferred during the hologram calculation, so we measured the two terms together as follows:

$$T_{\text{calc}} = T_{\text{horn}} + T_{\text{out}} . \quad (4)$$

Since the system was designed to output the hologram data in parallel with hologram calculation, the hologram transfer time T_{out} was concealed, ideally $T_{\text{calc}} = T_{\text{horn}}$.

We can theoretically obtain T_{horn} from the hardware design as

$$T_{\text{horn}} = \frac{N_{\text{obj}} \times N_{\text{hol}}}{f \times C} , \quad (5)$$

where f is the system's operating frequency (0.25 GHz) and C is the number of parallel holographic calculation circuits (cores). Here, $C = 4,480 \times (\text{number of boards})$.

The value of T_{horn} determines the upper limit of HORN-8's performance, i.e. its theoretical performance. Table 2(a) shows the theoretical T_{horn} values alongside T_{calc} .

The system's effective performance P_{effect} can be expressed as

$$P_{\text{effect}} = \frac{T_{\text{horn}}}{T_{\text{total}}} . \quad (6)$$

Table 2(a) presents the results for one board, showing an extremely high effective performance of 99%. The ratio of the transfer time to the total time T_{total} , shown in brackets besides T_{in} , is very small.

Table 2. Performance of the HORN-8 system with (a) one board and (b) a cluster of boards, showing the time taken to generate a $1,920 \times 1,080$ -pixel hologram.

(a) One HORN-8 board

N_{obj}	T_{in} [msec]	$T_{calc} (T_{horn})$ [msec]	T_{total} [msec]	P_{effect}
10,000	0.17 (0.9%)	18.54 (18.51)	18.71	98.4%
20,000	0.21 (0.6%)	37.10 (37.03)	37.31	99.2%
30,000	0.28 (0.5%)	55.66 (55.54)	55.94	99.2%
40,000	0.34 (0.5%)	74.24 (74.06)	74.58	99.3%
50,000	0.40 (0.4%)	92.80 (92.57)	93.20	99.3%
60,000	0.47 (0.4%)	111.37 (111.09)	111.84	99.3%
70,000	0.62 (0.5%)	129.90 (129.60)	130.52	99.3%
80,000	0.68 (0.5%)	148.47 (148.11)	149.15	99.3%
90,000	0.75 (0.4%)	167.03 (166.63)	167.78	99.3%
100,000	0.80 (0.4%)	185.60 (185.14)	186.40	99.3%
1,000,000	7.62 (0.4%)	1856.17 (1851.43)	1863.79	99.3%
10,000,000	76.77 (0.4%)	18561.90 (18514.29)	18638.67	99.3%

(b) Cluster of HORN-8 boards

N_{obj}	2 boards T_{calc} [msec] (P_{effect})	4 boards T_{calc} [msec] (P_{effect})	6 boards T_{calc} [msec] (P_{effect})	8 boards T_{calc} [msec] (P_{effect})
10,000	9.93 (93.3%)	5.19 (89.2%)	3.52 (87.6%)	2.70 (85.9%)
20,000	19.54 (94.8%)	9.93 (93.2%)	6.61 (93.4%)	5.02 (92.2%)
30,000	29.22 (95.0%)	14.70 (94.5%)	9.91 (93.4%)	7.41 (93.7%)
40,000	38.94 (95.1%)	19.58 (94.6%)	13.11 (94.2%)	9.82 (94.2%)
50,000	48.59 (95.2%)	24.50 (94.5%)	16.35 (94.4%)	12.28 (94.2%)
60,000	58.26 (95.3%)	29.31 (94.8%)	19.61 (94.2%)	14.70 (94.5%)
70,000	68.17 (95.1%)	34.88 (92.9%)	22.94 (94.5%)	17.21 (94.1%)
80,000	77.86 (95.1%)	39.74 (93.2%)	26.13 (94.5%)	19.62 (94.4%)
90,000	87.64 (95.1%)	44.62 (93.4%)	29.40 (94.5%)	22.05 (94.5%)
100,000	97.41 (95.0%)	49.51 (93.5%)	32.67 (94.5%)	24.49 (94.5%)
1,000,000	972.18 (95.2%)	494.78 (93.5%)	326.43 (94.5%)	244.99 (94.5%)
10,000,000	9714.58 (95.3%)	4935.02 (93.8%)	3263.16 (94.6%)	2448.10 (94.5%)

Table 2(b) shows the results for the cluster systems. Even the eight-board system achieved an effective speed as high as 94% for over 40,000 object points. This is approximately 5% lower than for a single board but remained almost constant as the number of boards increased. The communication cost was also well hidden in the

calculation cost for the cluster systems, showing that HORN-8's architecture avoids communication bottlenecks. In addition, it shows that, given sufficient computational power, a large memory is not required. These results demonstrate that high parallelisation can greatly accelerate holographic calculations, suggesting that dedicated hardware can be effective.

Figure 3 shows the performance of the cluster system constructed from multiple HORN-8 boards. The calculation speed increased in approximate proportion to the number of boards used. The time required to generate a hologram of $1,920 \times 1,080$ pixels from a 3D image comprising 100,000 points was 186 ms when using a single board and 24.5 ms when using eight boards. The eight-board cluster system was 7.6 times faster than the single board.

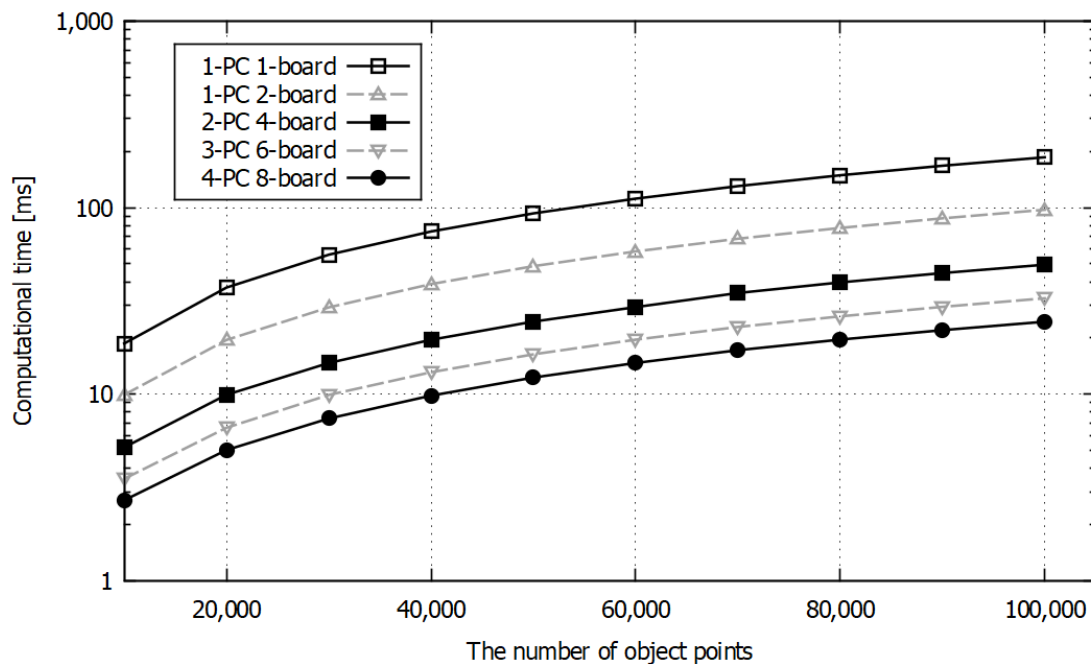


Figure 3. Performance of the HORN-8 cluster system. One or two HORN-8 boards were connected to each PC, and the PCs were clustered. The vertical axis expressed in

logarithm shows the time taken to generate a hologram of $1,920 \times 1,080$ (2 million) pixels.

Figure 4 shows the 3D images reconstructed from the holograms produced by the eight-board HORN-8 cluster. We created 7,877-point 3D CGs for a planetary exploration satellite and then made holograms from them. The left side of Fig. 4 shows the 3D image by the $1,920 \times 1,080$ hologram at a pixel pitch of $6.5 \mu\text{m}$; the hologram was displayed on an LCD of the same size and projected using an optical system (see also Supplementary Video S1). The right of the figure shows $9,600 \times 10,800$ (100 million) holograms at a pixel pitch of $1 \mu\text{m}$, as reconstructed using a computer simulation. This simulation was used because no current electronic display device is capable of generating 100 million pixels (see also Supplementary Video S2). The simulation was run using the open source CWO library, which was also developed by our group³⁷. Considering that the viewing angle of a 100-million pixel hologram at $1 \mu\text{m}$ pixel pitch is approximately 30° , three images were reconstructed from different angles.

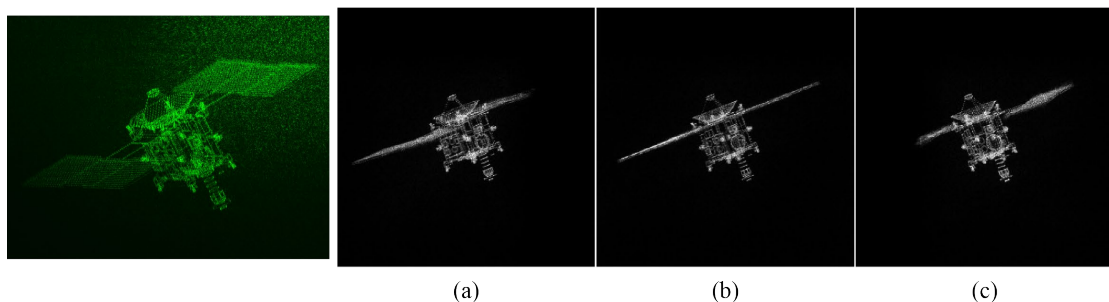


Figure 4. Electroholography from the eight-board HORN-8 cluster system: These images were optically reconstructed from a 2-million pixel hologram with $6.5\text{-}\mu\text{m}$ pixel pitch (left) and computationally reconstructed images from a 100-million pixel

hologram with 1- μm pixel pitch (right a-c). Figure (a) shows the projection as viewed from the left at 11° , (b) from the front at 0° and (c) from the right at 11° .

The goal of this research was to demonstrate the real-time generation of $10,000 \times 10,000$ (100 million) pixel holograms. The generation time when using the eight-board HORN-8 cluster system was 100 ms, which is the frame rate of 10 fps. Although the volume of information captured in a hologram is huge, parallel computation was demonstrated to work well, and dedicated circuits were shown to be effective. Given that data communication is not performed between pixels and output data are not reused, the pixels can be calculated separately. This approach does away with the potential communication bottleneck that may arise in conventional numerical calculation, thus allowing the video system to operate at speeds proportional to the number of physical calculation circuits used. This research demonstrated that next-generation holographic 3D imaging systems can be realised using currently available technology.

Finally, we also present an example of large-scale electroholography using the HORN-8 system, though not in real time. We generated a 100-million-pixel hologram from a 10-million-point object. The 3D model was the same planetary exploration satellite as shown in Fig. 4. We used simulation to reconstruct the image obtained from this large-scale hologram (Fig. 5). Since the HORN-8 system can only handle 65,536 (2^{16}) object points at a time, we divided the object data into 160 blocks of 65,000 points each and generated holograms from each block data. We reconstructed each of these in turn by the time-division method to obtain a single still image. The process required 125 s on the eight-board HORN-8 cluster system, but this result

shows that it will be possible to achieve high-quality electroholographic images with a wide field of view by scaling up the HORN-8 system. It also suggests that large-scale electroholography does not require ultra-high-speed processors or large amounts of memory and can be realised by mass-producing a dedicated calculation system (dedicated chip) for holography.



Figure 5. Example of a large-scale electroholographic image obtained using the eight-board HORN-8 cluster system. A 100-million-pixel hologram was generated from a 10-million-point object and reconstructed via simulation. The object data was divided into 160 blocks, and a separate hologram was prepared for each. These were then reconstructed by the time-division method to obtain a single still image. A total of 125 s were required to generate the hologram. The progress of the time-division reconstruction is also shown in Supplementary Video S3, where the 160 reconstructed images are displayed sequentially at 60 fps.

References

- [1] Hilaire, P. S. *et al.* Electronic display system for computational holography. *Proc. SPIE* **1212-20**, 174-182 (1990).
- [2] Lucente, M. Interactive three-dimensional holographic displays: seeing the future in depth. *Comp. Graphics* **31**, 63-67 (1997).
- [3] Aoshima, K. *et al.* Submicron magneto-optical spatial light modulation device for holographic displays driven by spin-polarized electrons. *J. Disp. Technol.* **6**, 374-380 (2010).
- [4] Lucente, M. Interactive Computation of Holograms Using a Look-Up Table. *J. Electron. Imaging* **2**, 28-34 (1993).
- [5] Yoshikawa, H., Iwase, S. & Oneda, T. Fast computation of Fresnel holograms employing difference. *Proc. SPIE* **3956**, 48-55 (2000).
- [6] Nishitsuji, T., Shimobaba, T., Kakue, T. & Ito, T. Review of fast calculation techniques for computer-generated holograms with the point light source-based model. *IEEE Trans. Ind. Inform.* **13**, 2447-2454 (2017).
- [7] Shimobaba, T., Kakue, T. & Ito, T. Review of fast algorithms and hardware implementations on computer holography. *IEEE Trans. Ind. Inform.* **12**, 1611-1622 (2016).
- [8] Masuda, N., Ito, T., Tanaka, T., Shiraki, A. & Sugie, T. Computer generated holography using a graphics processing unit. *Opt. Express* **14**, 603-608 (2006).
- [9] Takada, N. *et al.* Fast high-resolution computer-generated hologram computation using multiple graphics processing unit cluster system. *Appl. Opt.* **51**, 7303-7307 (2012).
- [10] Song, J., Park, J., Park, H., & Park, J.-II. Real-time generation of high-definition resolution digital holograms by using multiple graphic processing units, *Opt. Eng.*

- 52**, 015803 (2013).
- [11] Watlington, J. A., Lucente, M., Sparrell, C. J., Bove Jr., V. M., & Tamitani, I. A hardware architecture for rapid generation of electro-holographic fringe patterns, *Proc. SPIE* **2406-23**, 172-183 (1995).
- [12] Lucente, M. & Galyean, T. A. Rendering interactive holographic images, *Proc. ACM SIGGRAPH 95*, 387-394 (1995).
- [13] Buckley, E. Real-Time Error Diffusion for Signal-to-Noise Ratio Improvement in a Holographic Projection System. *J. Disp. Technol.* **7**, 70-76 (2011).
- [14] Seo, Y. H., Choi, H. J., Yoo, J. S. & Kim, D. W. An architecture of a high-speed digital hologram generator based on FPGA. *J. Syst. Architect.* **56**, 27-37 (2010).
- [15] Masuda, N, *et al.* Special purpose computer for digital holographic particle tracking velocimetry. *Opt. Express* **14**, 587-592 (2006).
- [16] Abe, Y. *et al.* Special purpose computer system for flow visualization using holography technology. *Opt. Express*, **16**, 7686-7692 (2008).
- [17] Masuda, N. *et al.* Special purpose computer system with highly parallel pipelines for flow visualization using holography technology. *Comput. Phys. Commun.* **181**, 1986-1989 (2010).
- [18] Cheng, C. J., Hwang, W. J., Chen, C. T. & Lai, X. J. Efficient FPGA-based Fresnel transform architecture for digital holography. *J. Disp. Technol.* **10**, 272-281 (2014).
- [19] Ito, T., Yabe, T., Okazaki, M. & Yanagi, M. Special-purpose computer HORN-1 for reconstruction of virtual image in three dimensions. *Comp. Phys. Commun.* **82**, 104-110 (1994).
- [20] Ito, T. *et al.* Special-purpose computer for holography HORN-2. *Comp. Phys. Commun.* **93**, 13-20 (1996).

- [21] Shimobaba, T. *et al.* Special-purpose computer for holography HORN-3 with PLD technology. *Comp. Phys. Commun.* **130**, 75-82 (2000).
- [22] Shimobaba, T. & Ito, T. Special-purpose computer for holography HORN-4 with recurrence algorithm. *Comp. Phys. Commun.* **148**, 160-170 (2002).
- [23] Ito, T. *et al.* A special-purpose computer HORN-5 for a real-time electroholography. *Opt. Express* **13**, 1923-1932 (2005).
- [24] Ichihashi, Y. *et al.* HORN-6 special-purpose clustered computing system for electroholography. *Opt. Express* **17**, 13895-13903 (2009).
- [25] Okada, N. *et al.* Special-Purpose Computer HORN-7 with FPGA Technology for Phase Modulation Type Electroholography. *Proc. The 19th International Display Workshops in conjunction with Asia Display 2012 (IDW/AD' 12)*, 3Dp-26 (2012).
- [26] Ito, T. & Shimobaba, T. One-unit system for electroholography by use of a special-purpose computational chip with a high-resolution liquid-crystal display toward a three-dimensional television. *Opt. Express* **12**, 1788-1793 (2004).
- [27] Shimobaba, T., Shiraki, T., Masuda, N. & Ito, T. Electroholographic display unit for three-dimensional display by use of special-purpose computational chip for holography and reflective LCD panel. *Opt. Express* **13**, 4196-4201 (2005).
- [28] Ichikawa, T., Yamaguchi, K. & Sakamoto, Y. Realistic expression for full-parallax computer-generated holograms with the ray-tracing method. *Appl. Opt.* **52**, A201-A209 (2013).
- [29] Pan, Y. *et al.* Fast CGH computation using S-LUT on GPU. *Opt. Express* **17**, 18543-18555 (2009).
- [30] Matsushima, K. & Nakahara, S. Extremely high-definition full-parallax computer-generated hologram created by the polygon-based method. *Appl. Opt.* **48**, H54-H63 (2009).

- [31] Ogihara, Y., Ichikawa, T. & Sakamoto, Y. Fast calculation with point based method to make CGHs of the polygon model. *Proc. SPIE* **9006**, 90060T-1 (2014).
- [32] Chen, J.-S., Chu, D. & Smithwick, Q. Rapid hologram generation utilizing layer-based approach and graphic rendering for realistic three-dimensional image reconstruction by angular tiling. *J. Electron. Imaging* **23**, 023016 (2014).
- [33] Chen, J.-S. & Chu, D. P. Improved layer-based method for rapid hologram generation and real-time interactive holographic display applications. *Opt. Express* **23**, 18143-18155 (2015).
- [34] Shimobaba, T. & Ito, T. An efficient computational method suitable for hardware of computer-generated hologram with phase computation by addition. *Comp. Phys. Commun.* **138**, 44-52 (2001).
- [35] Nishitsuji, T., Shimobaba, T., Kakue, T., Arai, D. & Ito, T. Simple and fast cosine approximation method for computer-generated hologram calculation. *Opt. Express* **23**, 32465-32470 (2015).
- [36] Niwase, H. *et al.* Real-time spatiotemporal division multiplexing electroholography with a single graphics processing unit utilizing movie features. *Opt. Express* **22**, 28052-28057 (2014).
- [37] Shimobaba, T. *et al.* Computational wave optics library for C++: CWO++ library. *Comput. Phys. Commun.* **183**, 1124-1138 (2012).

Acknowledgements

This work was supported by the Japan Society for the Promotion of Science (Grant-in-Aid No. 25240015).

Author contributions

T.I. planned the project. Takashige S. and T.I. designed the HORN-8 board, and Takashige S. and M.O. developed it. T.A., Takashige S. and T.I. designed the circuit of the HORN-8 system, and T.A. implemented it to FPGA. R.H., H.N., Takashige S., T.A, Tomoyoshi S. and T.I. evaluated the HORN-8 system. H.N. and R.H. created 3D models for holography. T.N., N.T., Y.E., N.M. and Tomoyoshi S. developed the supported algorithms for the HORN-8 system. Y.I., A.S. and T.K. built the optical system. All authors contributed the discussions and reviewed the manuscript.

Author Information

The authors declare no competing financial interests.