Highly Parallel Special-purpose Computer for Electroholography on System on a Chip

Yota Yamamoto[1], Nobuyuki Masuda[2], Hirotaka Nakayama[3], Tomoyoshi Shimobaba[1], Takashi Kakue[1], Tomoyoshi Ito[1]

[1] Graduate School of Science and Engineering, Chiba University, 1-33 Yayoi-cho, Inage-ku, Chiba 263-8522, Japan
[2] Faculty of Industrial Science and Technology, Tokyo University of Science, 6-3-1 Niijuku, Katsushika-ku, Tokyo 125-8585, Japan
[3] National Astronomical Observatory of Japan, 2-21-1 Mitaka-shi, Tokyo 181-8588, Japan
Keywords: Holography, 3D display, embedded system, SoC, FPGA.

ABSTRACT

For realizing electroholography, a compact and high-performance computer is required. In this study, we implemented highly parallel special-purpose computer for electroholography on system on a chip. As a result, we succeeded in speeding up calculation 200 times faster than a CPU and a GPU.

1 INTRODUCTION

Recently, commercial stereoscopic imaging systems that employ a head-mounted display (HMD) have become available. These systems are used in a wide range of fields, such as entertainment, education, and medical applications. However, HMDs have a problem known as vergence-accommodation conflict (VAC) [1]. In the binocular disparity method used for conventional HMDs, the difference between the accommodation distance and the vergence distance is inconsistent. As a result, nausea and fatigue occur.

Electroholography [2] is attracting attention as an ideal three-dimensional (3D) display technology because it can reproduce all the depth cues. It is worth noting that electroholography can solve the VAC problem with a natural stereoscopic viewing. In contrast, using electroholography in VR is difficult. Electroholography uses Computer-Generated Hologram (CGH) which is generated by calculation. The enormous computational power requirement for CGH has been a barrier to practical use [3].

Our team has been developed special-purpose computers for holography to reduce calculation time [4-6]. Especially we aim to mount on HMD; thus, we develop a small system. Our system used the Xilinx ZYNQ MPSoC, system-on-a-chip (SoC) in which an ARM CPU and FPGA are built in one chip.

In this paper, we introduced the detailed structure of highly parallel special-purpose computer using recurrence relation algorithm [7]. Moreover, we demonstrated its performance.

2 PRINCIPLE

2.1 Computer-generated Hologram

We used computer-generated hologram (CGH) based on point cloud 3D models. In M points cloud, the formula under the condition of $z_j >> x_j, y_j$ is following:

$$I(x_\alpha, y_\alpha) = \sum_{j=1}^{M} A_j \cos\left[2\pi\theta_{aj}\right], \quad (1)$$

$$\theta_{aj} = \left[\rho_j \cdot \{(x_a - x_j)^2 + (y_a - y_j)^2\}\right]. \quad (2)$$

We called Eq. (2) Fresnel approximation method. Here A_j (fixed at 1) is the amplitude of the point light source, ρ_j is $1/2\lambda|z_j|$, and λ is the reference light's wavelength. $I(x_\alpha, y_\alpha)$ are pixel values on CGH. Calculation complexity increases in proportion to the size of the hologram × the number of points. The enormous computational complexity and calculation time of CGH make it difficult to realize.

2.2 Recurrence Relation Algorithm

There is a simple calculation method named recurrence relation algorithm [7]. Here, we defined

$$\Gamma_j = \frac{1}{\lambda z_j} = 2\rho_j, \quad (3)$$

$$\Delta_{0j} = \rho_j\{2(x_0 - x_j) + 1\}. \quad (4)$$

In the recurrence relation algorithm, initially Eq. (2) as θ_{0j} is calculated. In the n-th in the x-axis direction, θ_{nj} is formulated with the recurrence relation algorithm follows:

$$\theta_{nj} = \theta_{(n-1)j} + \Delta_{(n-1)j}. \quad (5)$$

Also, update Δ_{0j} value using the following equation:

$$\Delta_{nj} = \Delta_{(n-1)j} + \Gamma_j. \quad (6)$$

We can calculate θ_{nj} by repeating simple Eq. (5) and Eq. (6).

3 IMPLEMENTATION

3.1 Architecture

Figure 1 shows the architecture of special-purpose pipeline for electroholography. We used recurrence relation algorithm to calculate CGH. The special-purpose pipeline is consisting of Basic Phase Unit (BPU), Additional Phase Unit (APU) and Quantization Unit (QU). BPU calculates basic phase in the recurrence relation

algorithm. APU calculates additional phase in recurrence relation algorithm. Each unit executes calculation in parallel one pixel on CGH. QU calculates light intensity and output “0” or “255” in pixel. The QU circuits are built in APU and BPU. MPU wraps 1 APU and 1,919 BPU’s. Our system has 3 MPU’s; thus, it can calculate 5,760 pixels on CGH at one time in parallel.

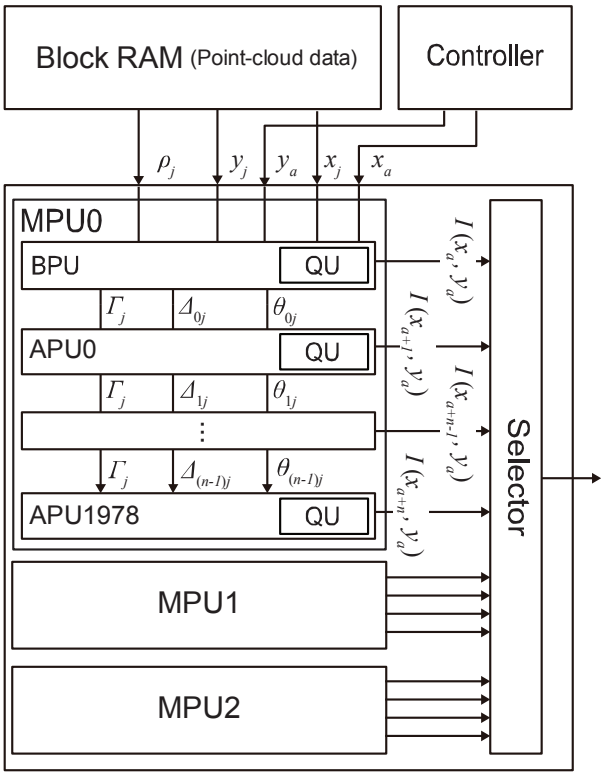


Fig. 1 Architecture of special-purpose pipeline for electroholography; Block RAM is the on-chip memory. It is used to store point cloud data.

3.2 Basic Phase Unit (BPU)

BPU calculates basic phase in recurrence relation algorithm. BPU calculates Eq. (2). For the next APU calculation, BPU also calculates Eq. (3) and Eq. (4).

Here, ρ_j in Eq. (4) was calculated by the CPU. In general speaking, division operation with FPGA cause increasing resource usage and decreasing operating frequency. In order to save resource usage and improve operating frequency, ρ_j was calculated by the CPU.

3.3 Additional Phase Unit (APU)

APU calculates additional phase in recurrence relation algorithm. APU calculates Eq. (5). For the next APU calculation, APU also updates Δ_{nj} using Eq. (6).

3.4 Quantization Unit (QU)

QU performs summing for each point cloud data which input from BPU and APU. Then, QU output “0” or “255” in pixel. Shown in Eq. (1), QU necessary to calculate cosine calculation. We used the LUT method, which was a fast method to use the table reference with the calculated cosine.

3.5 Optical Setup

The optical system (including the proposed system) is shown in Fig. 2. For a holographic display, a reflection type Spatial Light Modulator (SLM; 1,920 1,080 pixels) is used to display CGHs. Here, the pixel interval is 6.4 μm, the distance of the reproduced image from the hologram is 0.5 m, and the wavelength of the reference light is 532 nm.

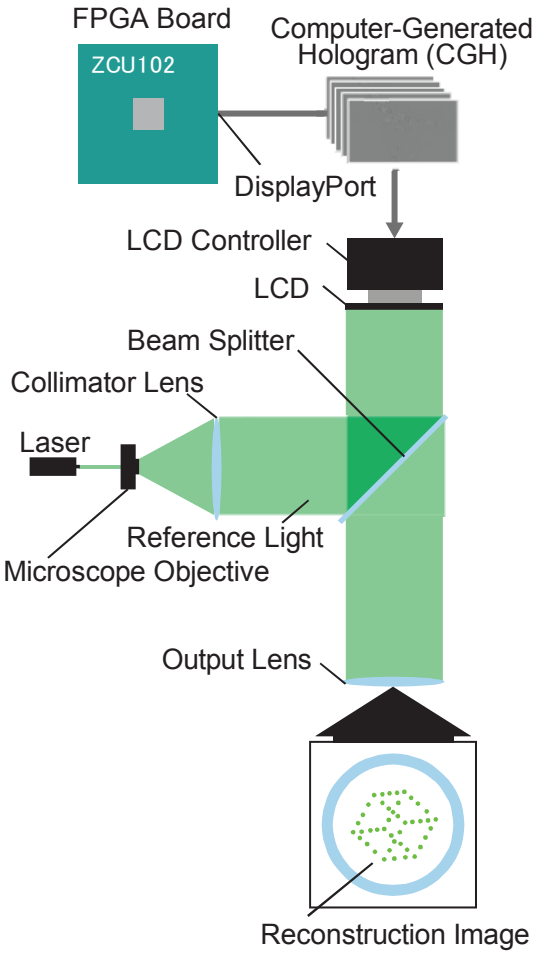


Fig. 2 Optical setup.

4 RESULTS

Table 1 shows the results of comparing time required for CGH calculation of 1,920 × 1,080 pixels from point cloud data with 95,949 points using the CPU, GPU, and FPGA.

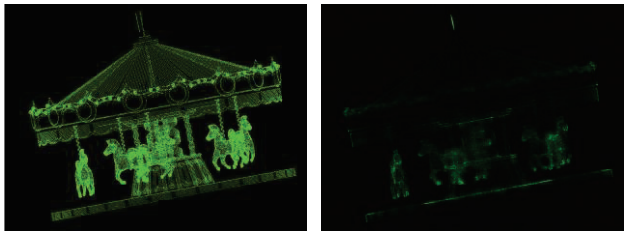
Table 1 The results of comparing time required for CGH calculation.

Hardware	Calculation time [s]	Acceleration ratio
FPGA	0.099	242
NVIDIA Jetson TX1	19.155	1.25
Intel Xeon	23.988	1.00

Here, we used an Intel Xeon CPU E52697 v2 2.70 GHz for CPU. We implemented CGH calculation program using a recurrence relation algorithm in float precision with Intel C compiler 16.0.1.150. Note that we used all cores and executed in parallel. We used an NVIDIA Jetson TX1 embedded system for GPU. In order to optimize the GPU's SIMD architecture, we implemented CGH calculation program using the Fresnel approximation method in floating precision with CUDA 8.0. It was also used all cores and executed in parallel.

As shown in Table 1, we succeeded in speeding up calculation 240 times faster compared to the CPU and 190 times faster compared to the GPU.

Figure 3 (a) shows 3D data which is used to calculate CGH. Figure 3 (b) shows a reconstruction image.



(a) 3D data image (b) Reconstruction image

Fig. 3 3D data image and reconstruction image.

5 DISCUSSION

We have been developed special-purpose computer for holography to reduce calculation time [4-6]. In this paper, we used recurrence relation algorithm which is developed for hardware implementation. As a result, the new system has 7 times as many parallel cores and 10 times faster in calculation time compared with last system [6].

Shown in Fig .3, reconstruction image didn't represent datil 3D images. It is not a problem at special-purpose pipeline, but SLM limitation. It is difficult for 1920 × 1080 pixel SLM to represent over 50,000-point cloud data. In the current study [8], we demonstrated 400,000-point cloud reconstruction image at 1,920 × 1,080 pixel SLM using spatiotemporal division method. For the future work, we will demonstrate special-purpose computer on system on a Chip using spatiotemporal division method. Or using more high resolution SLM, we will demonstrate special-purpose computer.

6 CONCLUSIONS

In this study, we implemented highly parallel special-purpose computer for electroholography on system on a chip. As a result, we succeeded in speeding up calculation 240 times faster than a CPU and 190 times faster than a GPU. Our system can calculate in 10 fps from 0.1-million-point cloud at CGH of 1,920 × 1,080 pixels.

ACKNOWLEDGEMENTS

This work was partially supported by JSPS KAKENHI Grant Numbers 19H01097.

REFERENCES

[1] G. Kramida, "Resolving the vergence-accommodation conflict in head-mounted displays," IEEE Transactions on Visualization and Computer Graphics, Vol. 22, No. 7, pp. 1912-1931 (2016).

[2] P. S. Hilaire, S. A. Benton, M. E. Lucente, M. L. Jepsen, J. S. Kollin, H. Yoshikawa, J. Underkoffler, "Electronic display system for computational holography," Proc. SPIE, Vol. 1212, pp. 174-182 (1990).

[3] M. Lucente, "Interactive three-dimensional holographic displays: seeing the future in depth," ACM SIGGRAPH Computer Graphics. Vol. 31, pp. 63-67 (1997)

[4] T. Sugie, T. Akamatsu, T. Nishitsuji, R. Hirayama, N. Masuda, H. Nakayama, Y. Ichihashi, A. Shiraki, M. Oikawa, N. Takada, Y. Endo, T. Kakue, T. Shimobaba, T. Ito, "High-performance parallel computing for next-generation holographic imaging", Nature Electronics, Vol. 1, pp. 254–259 (2018)

[5] T. Nishitsuji, Y. Yamamoto, T. Sugie, T. Akamatsu, R. Hirayama, H. Nakayama, T. Kakue, T. Shimobaba, T. Ito, "Special-purpose computer HORN-8 for a phase-type electro-holography," Optics Express, Vol. 26, Issue 20, pp. 26722-26733 (2018)

[6] Y. Yamamoto, N. Masuda, R. Hirayama, H. Nakayama, T. Kakue, T. Shimobaba, and T. Ito, "Special-purpose computer for electroholography in embedded systems," OSA Continuum, Vol. 2, No. 4, pp. 1166–1173 (2019).

[7] T. Shimobaba and T. Ito, "An efficient computational method suitable for hardware of computer-generated hologram with phase computation by addition," Computer Physics Communications, Vol. 138, No. 1, pp. 44-52 (2001).

[8] Y. Yamamoto, H. Nakayama, N. Takada, T. Nishitsuji, T. Sugie, T. Kakue, T. Shimobaba, and T. Ito, "Large-scale electroholography by HORN-8 from a point-cloud model with 400,000 points," Optics Express, Vol. 26, pp. 34259-34265 (2018).