

# Real-time gradation-expressible amplitude-modulation-type electroholography based on binary-weighted computer-generated hologram

Ren Noguchi<sup>1</sup>, Kohei Suzuki<sup>1</sup>, Yoshiki Moriguchi<sup>1</sup>, Minoru Oikawa<sup>2</sup>, Yuichiro Mori<sup>2</sup>, Takashi Kakue<sup>3</sup>, Tomoyoshi Shimobaba<sup>3</sup>, Tomoyoshi Ito<sup>3</sup>, and Naoki Takada<sup>2\*</sup>

<sup>1</sup> Graduate School of Integrated Arts and Sciences, Kochi University, Kochi 780-8520, Japan

<sup>2</sup> Research and Education Faculty, Kochi University, Kochi 780-8520, Japan

<sup>3</sup> Graduate School of Engineering, Chiba University, Chiba 263-8522, Japan

\*Corresponding author: [ntakada@is.kochi-u.ac.jp](mailto:ntakada@is.kochi-u.ac.jp)

Received May 7, 2021 | Accepted September 14, 2021 | Posted Online October 13, 2021

In amplitude-modulation-type electroholography, the binary-weighted computer-generated hologram (BW-CGH) facilitates the gradation-expressible reconstruction of three-dimensional (3D) objects. To realize real-time gradation-expressible electroholography, we propose an efficient and high-speed method for calculating bit planes consisting of BW-CGHs. The proposed method is implemented on a multiple graphics processing unit (GPU) cluster system comprising 13 GPUs. The proposed BW-CGH method realizes eight-gradation-expressible electroholography at approximately the same calculation speed as that of conventional electroholography based on binary computer-generated holograms. Consequently, we were able to successfully reconstruct a real-time electroholographic 3D video comprising approximately 180,000 points expressed in eight gradations at 30 frames per second.

**Keywords:** gradation-expressible electroholography; binary-weighted computer-generated hologram; digital micromirror device; computer-generated hologram.

**DOI:** [10.3788/COL202119.110501](https://doi.org/10.3788/COL202119.110501)

## 1. Introduction

Real-time electroholography based on a computer-generated hologram (CGH) is considered to realize the ultimate three-dimensional (3D) television experience<sup>[1-4]</sup>. However, the enormous amount of CGH computation associated with real-time electroholography prevents its practical realization.

The cost-effective graphics processing unit (GPU) provides high-performance computational power for facilitating the implementation of various numerical calculations. GPU-accelerated CGH calculations have been reported<sup>[5-15]</sup>, such as real-time electroholography using a multiple-GPU (multi-GPU) cluster system comprising many GPU boards<sup>[16-18]</sup>.

Amplitude-modulation-type CGH is applied to amplitude-modulation-type spatial light modulators (SLMs) such as the digital micromirror device (DMD). Although the light utilization efficiency of amplitude-modulation-type CGH is worse than that of the phase-only-type CGH, a DMD can display CGHs at higher frame rates than the phase-only-type SLM<sup>[19]</sup>. A color DMD LED holographic display has been reported<sup>[20]</sup>.

A DMD holographic image quality enhancement method using localized random down-sampling and adaptive intensity accumulation<sup>[21]</sup> and error diffusion<sup>[22,23]</sup> has been reported. By utilizing the advantage, wide-viewing-angle holographic displays have been reported<sup>[24-27]</sup>. The use of a DMD for gradation-expressible electroholography has also been reported<sup>[28,29]</sup>.

In our recent work, we proposed gradation-expressible electroholography using multiple bit planes consisting of binary-weighted CGHs (BW-CGHs)<sup>[30-32]</sup>. The proposed BW-CGHs can freely control the light intensity of reconstructed object points without affecting the brightness of the reconstructed light. However, real-time electroholographic reconstruction of a 3D object with various gradation values had not been achieved.

In this Letter, we propose an efficient and high-speed BW-CGH calculation method for real-time gradation-expressible electroholography using an amplitude-modulation-type SLM. In generating the BW-CGHs as bit planes, we avoid duplicating the calculation of the light intensities of object points with various gradation values.

## 2. Methods

In this study, we used the conventional binary CGH to reconstruct a bright 3D electroholographic image using an amplitude-modulation-type SLM (i.e., the DMD). A CGH is obtained using the following equation based on the Fresnel approximation:

$$I(x_h, y_h, 0) = \sum_{j=1}^{N_{obj}} A_j \cos \left\{ \frac{\pi}{\lambda z_j} [(x_h - x_j)^2 + (y_h - y_j)^2] \right\}, \quad (1)$$

where  $I(x_h, y_h, 0)$  is the light intensity of the point  $(x_h, y_h, 0)$  on a CGH.  $(x_j, y_j, z_j)$  and  $A_j$  are the coordinates and amplitude of the  $j$ th point of the 3D point cloud model, respectively. In this Letter, we set  $A_j$  to one.  $N_{obj}$  is the total number of points in the 3D model.  $\lambda$  is the wavelength of the reconstructing light. After calculating Eq. (1), a conventional binary CGH is generated by binarizing the light intensity  $I$  on a CGH. The corresponding pixel on the binary CGH is drawn in white when the light intensity  $I(x_h, y_h, 0)$  is greater than zero. Otherwise, it is drawn in black.

As shown in Figure 1, light intensities from a DMD can be controlled using binary pulse-width modulation (PWM) when the digital image with 8 bit depth (256 gradations) is input to a DMD module, as in a projector product<sup>[33]</sup>. A binary PWM sequence pattern is used to generate the light intensities from the DMD in proportion to the percentage of time the mirror is “on” during a single-frame-refresh period.

A BW-CGH is generated by changing the white of the conventional binary CGH to the gray with the constant gray-level value from 0 to 255. The BW-CGH can be displayed on a DMD using binary PWM. Figure 1 shows that the display time of the binary CGH can be controlled by the gray-level value of a BW-CGH. The intensity of the light diffracted by the BW-CGH is then weakened compared to that diffracted by the conventional binary CGH. That is, the BW-CGH acts as a neutral density filter for the conventional binary CGH. Figure 2 shows the reconstructions from BW-CGHs and the conventional binary CGH. The object points a, b, and c are reconstructed from two BW-CGHs (BW-CGH1, BW-CGH2) with different gray-level values and the conventional binary CGH, respectively. A higher gray-level value of a BW-CGH results in the reconstructed object point having a higher light intensity.

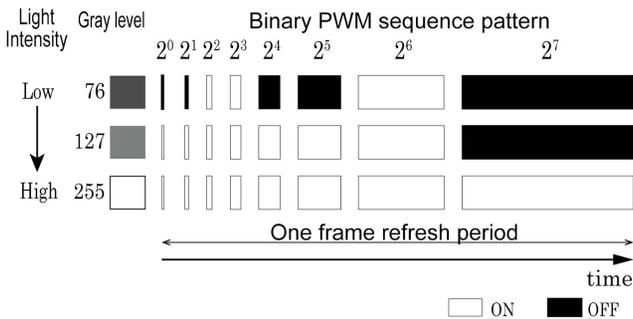


Fig. 1. Binary PWM sequence patterns corresponding to gray levels 76, 127, and 255.

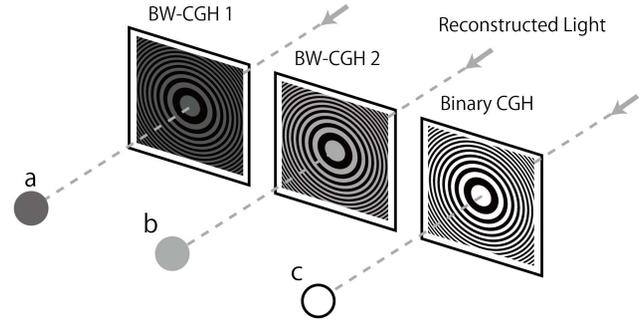


Fig. 2. Light intensities of object points reconstructed from BW-CGHs and conventional binary CGH.

Object points with various gradation values can be reconstructed when multiple BW-CGHs are used as multiple bit planes. Figure 3 shows an example of the object points expressed in eight gradations. As shown in Fig. 3, object points with gradation values ranging from one to seven are assigned to bit planes  $B^0$ ,  $B^1$ , and  $B^2$ , respectively. The bit planes  $B^0$ ,  $B^1$ , and  $B^2$  are obtained from the assigned object points using Eq. (1). The bit planes  $B^0$ ,  $B^1$ , and  $B^2$  are repeatedly displayed on an amplitude-modulation-type SLM, while the reconstructing light illuminates the SLM, thereby reconstructing a 3D-gradation image. Here, the reconstructed image from the  $m$ th bit plane  $B^m$  has a gradation value of  $2^m$ . For the BW-CGH corresponding to the  $m$ th bit plane  $B^m$ , we previously obtain the gradation value  $G^m$  of the gray area in an optical experiment. Therefore, the gradation values of the reconstructed 3D image from bit planes  $B^0$ ,  $B^1$ , and  $B^2$  become  $2^0$ ,  $2^1$ , and  $2^2$ , respectively.

Figure 4 shows a simple method for calculating the bit planes  $B^0$ ,  $B^1$ , and  $B^2$ . In this Letter, we call the method shown in Fig. 4 the duplicate calculation method. For example, the gradation values of the object points P1 to P7 are 1 to 7, respectively. As shown in Fig. 4, the object points P1 to P7 are assigned to the corresponding bit planes  $B^0$ ,  $B^1$ , and  $B^2$ , respectively. Lists  $B^0$ ,  $B^1$ , and  $B^2$  include the coordinate data of object points assigned to bit planes  $B^0$ ,  $B^1$ , and  $B^2$ , respectively. The bit planes  $B^0$ ,  $B^1$ , and  $B^2$  are calculated from Lists  $B^0$ ,  $B^1$ , and  $B^2$ , respectively. The object point P7 is assigned to the bit planes  $B^0$ ,  $B^1$ , and  $B^2$ . Therefore, using the method shown in Fig. 4, the light

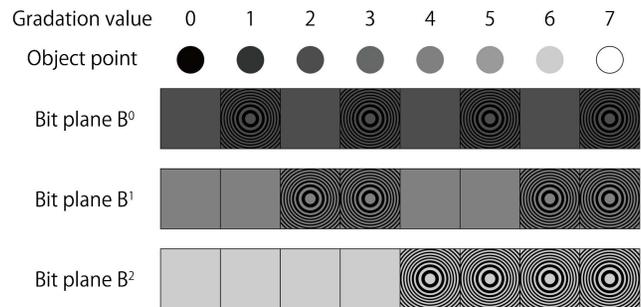


Fig. 3. Assignment of the object points of the 3D object to the bit planes.

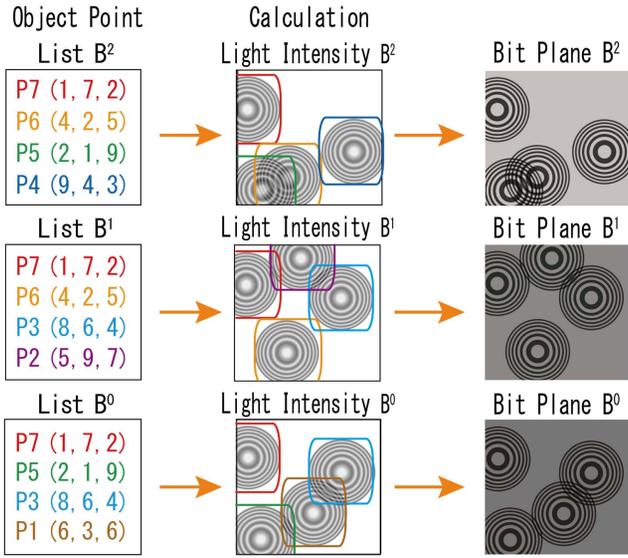


Fig. 4. Simple method for calculating the respective bit planes  $B^0$ ,  $B^1$ , and  $B^2$ .

intensity of the object point P7 is calculated three times to generate bit planes  $B^0$ ,  $B^1$ , and  $B^2$ . To improve efficiency, a way to avoid this duplicate calculation is desirable.

Figure 5 shows the proposed more efficient calculation method for avoiding duplicate calculation, whereby the object points of the 3D point cloud model are grouped by their gradation values. Object points with gradation values from 1 to 7 are assigned to Groups 1 to 7, respectively, with each object point in the 3D point cloud model belonging to only one group. Figure 6 shows the bit-plane flags of the seven groups, which indicate the bit planes to which the object points are assigned. Figure 5 shows how the program variables are prepared to store the light intensities  $I_{B0}$ ,  $I_{B1}$ , and  $I_{B2}$  of bit planes  $B^0$ ,  $B^1$ , and  $B^2$ . In the respective groups, the light intensities are calculated for the appropriate object points using Eq. (1). Here, the light intensity of each object point in the 3D model is calculated only once. The calculated values are then added to the light intensities of the corresponding bit planes based on the bit-plane flags shown in Fig. 6.

For example, in Group 7, the appropriate object points are assigned to bit planes  $B^0$ ,  $B^1$ , and  $B^2$ . As shown in Fig. 5, the light intensity of object point P7 is calculated using Eq. (1) just once. The value of the calculated light intensity is then added to light intensities  $I_{B0}$ ,  $I_{B1}$ , and  $I_{B2}$ .

After calculating the light intensities of all the groups in the corresponding bit planes, these light intensities are binarized, and then BW-CGHs are generated that correspond to bit planes  $B^0$ ,  $B^1$ , and  $B^2$  from the binarized data.

The total number of floating-point arithmetic operations in Eq. (1) is  $N_{fp}N_{obj} + (N_{obj} - 1)$ , where the number of floating-point arithmetic operations per object point is  $N_{fp}$ . We estimate the number of the respective floating-point arithmetic operations in the algorithms shown in Figs. 4 and 5. As shown in Fig. 6, the numbers of object points assigned to the bit planes  $B^0$ ,

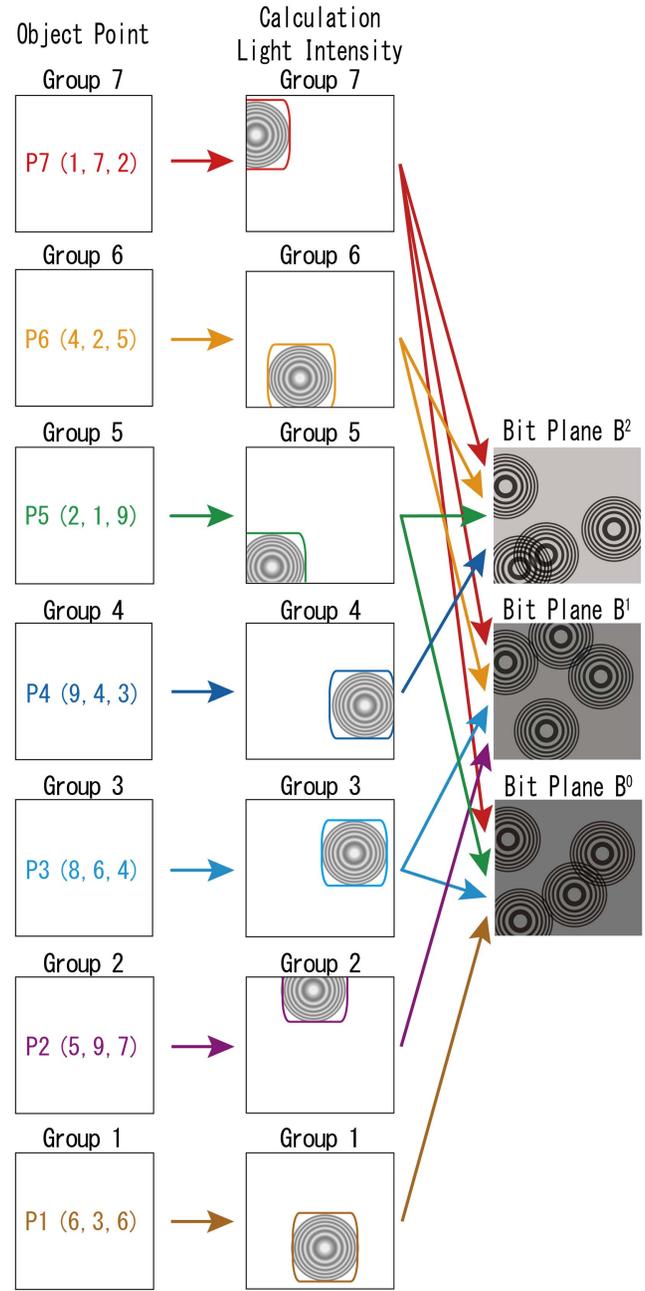


Fig. 5. Proposed method for calculating bit planes  $B^0$ ,  $B^1$ , and  $B^2$  by grouping the gradation values of the object points.

$B^1$ , and  $B^2$  are  $N_{GP7} + N_{GP5} + N_{GP3} + N_{GP1}$ ,  $N_{GP7} + N_{GP6} + N_{GP3} + N_{GP2}$ , and  $N_{GP7} + N_{GP6} + N_{GP5} + N_{GP4}$ , respectively, where the number of the object points belonging to Groups 1 to 7 is  $N_{GP1}$  to  $N_{GP7}$ , respectively. In the algorithm of Fig. 4, the number of the floating-point arithmetic operations to generate bit planes is  $B^0$ ,  $B^1$ , and  $B^2$  becomes  $N_{fp}[3N_{GP7} + 2(N_{GP6} + N_{GP5} + N_{GP3}) + N_{GP4} + N_{GP2} + N_{GP1}]$ . While, in the proposed algorithm of Fig. 5, the number of the floating-point arithmetic operations becomes  $N_{fp}(N_{GP7} + N_{GP6} + N_{GP5} + N_{GP4} + N_{GP3} + N_{GP2} + N_{GP1})$ . Therefore, the speed-up

	Gradation Value	Bit-plane Flag		
		$B^2$	$B^1$	$B^0$
Group 7	7	( 1 , 1 , 1 )		
Group 6	6	( 1 , 1 , 0 )		
Group 5	5	( 1 , 0 , 1 )		
Group 4	4	( 1 , 0 , 0 )		
Group 3	3	( 0 , 1 , 1 )		
Group 2	2	( 0 , 1 , 0 )		
Group 1	1	( 0 , 0 , 1 )		

Fig. 6. Bit-plane flags in the respective groups.

$S$  of the proposed algorithm compared with the algorithm shown in Fig. 4 theoretically becomes

$$S = \frac{3N_{GP7} + 2(N_{GP6} + N_{GP5} + N_{GP3}) + N_{GP4} + N_{GP2} + N_{GP1}}{N_{GP7} + N_{GP6} + N_{GP5} + N_{GP4} + N_{GP3} + N_{GP2} + N_{GP1}} \quad (2)$$

The computational complexity of Eq. (1) becomes  $O(N_{obj}N_{cgh})$ , where  $N_{cgh}$  is the total number of pixels on a CGH. The calculation time is enormous, as the BW-CGH is calculated as a bit plane by Eq. (1). To realize real-time gradation-expressible electroholography based on BW-CGHs, we accelerate the BW-CGH calculation using a multi-GPU cluster electroholography system, as shown in Fig. 7. This system consists of a CGH display node with a single SLM and CGH calculation nodes. These nodes are connected via a gigabit ethernet network. In the example shown in Fig. 7, each of the CGH calculation nodes in the system has three GPUs.

The CGH calculation nodes use pipeline processing to accelerate the BW-CGH calculations, as shown in Figure 8. Each of the GPUs from GPU 1 to GPU  $N$  generates three BW-CGHs, which become the three bit planes  $B^0$ ,  $B^1$ , and  $B^2$  at each frame of the eight-gradation 3D holographic video. In each frame, the respective symbols from Group 7 to 1 in Fig. 8 indicate the calculations of the light intensities of the object points assigned to

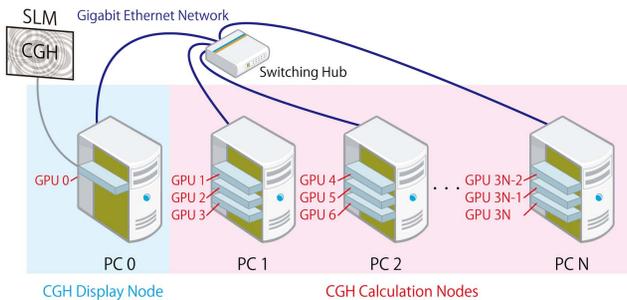


Fig. 7. Multi-GPU cluster electroholography system with a single SLM.

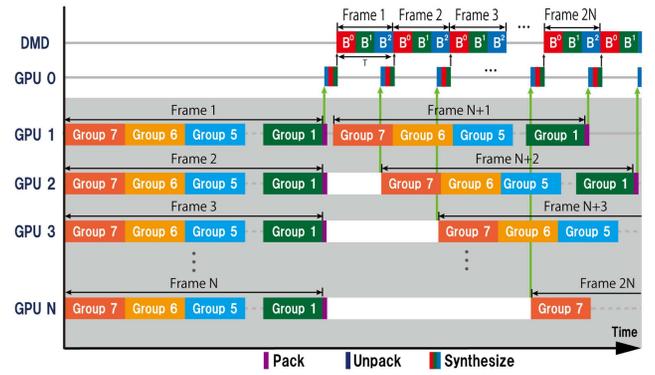


Fig. 8. Pipeline processing for real-time gradation-expressible electroholography based on BW-CGH.

the respective groups. In these groups, the calculated values are added to the corresponding light intensities  $I_{B0}$ ,  $I_{B1}$ , and  $I_{B2}$ .

In each frame, the calculated light intensities  $I_{B0}$ ,  $I_{B1}$ , and  $I_{B2}$  are binarized and packed to reduce the transfer of data to 1 bit per pixel<sup>[17]</sup>. The packed data are then sent to the CGH display node from each of the CGH calculation nodes. The CGH display node receives the packed data in the frame order of the holographic 3D video at the display time interval. GPU 0 on the CGH display node unpacks the packed data to generate three BW-CGHs, which become the bit planes  $B^0$ ,  $B^1$ , and  $B^2$ . The eight-gradation 3D holographic video can be reconstructed by the persistence of vision if the high-speed playback of the bit planes  $B^0$ ,  $B^1$ , and  $B^2$  is achieved.

The proposed method requires three BW-CGHs to be displayed on an SLM within a single-frame refresh period. For this reason, we adopted a DMD as the amplitude-modulation-type SLM. Red, green, and blue images can be sequentially displayed on a DMD in a time division manner within a single-frame refresh period when a red–green–blue (RGB) color image is input to a DMD module.

To achieve the high-speed BW-CGH playback, “synthesized RGB BW-CGH” shown in Figure 9 is used as an RGB color image that is input to a DMD module. Figure 9 shows how to make “synthesized RGB BW-CGH.” The respective BW-CGHs corresponding to  $B^2$ ,  $B^1$ , and  $B^0$  were converted into red, green, and blue BW-CGHs. The red, green, and blue colors have the same gradation values as the gradation values  $G^2$ ,  $G^1$ , and  $G^0$ , respectively. Here,  $G^2$ ,  $G^1$ , and  $G^0$  are the gradation values of the gray areas of the bit planes  $B^2$ ,  $B^1$ , and  $B^0$ , respectively. “Synthesized RGB BW-CGH” is synthesized from the red, green, and blue BW-CGHs. “Synthesized RGB CGH” is then input to the DMD module. As shown in the lower time chart of Figure 9, the DMD module automatically separates “synthesized RGB CGH” into the red, green, and blue BW-CGHs, and the respective bit planes  $B^2$ ,  $B^1$ , and  $B^0$  are displayed in order on the DMD panel within a single-frame refresh period. Here, a one-colored light is used as the reconstructing light. The color of the reconstructed 3D image is the same color as the reconstructing light.

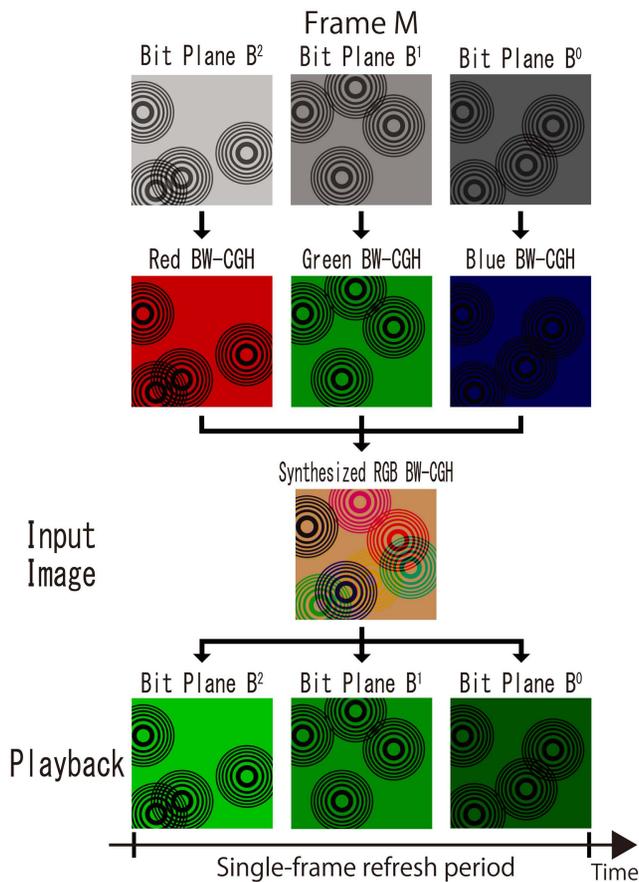


Fig. 9. Eight-gradation holographic 3D video reproduction using bit plans  $B^0$ ,  $B^1$ , and  $B^2$  on a DMD.

### 3. Results and Discussion

To evaluate the performance of the proposed method, we conducted an experiment using a multi-GPU cluster system consisting of a CGH display node with a single GPU board and four CGH calculation nodes. Each CGH calculation node has three GPU boards. Table 1 shows the specifications of the personal computers (PCs) that serve as the nodes in the multi-GPU cluster system.

Figure 10 shows the display-time interval against the number of object points assigned to the respective bit planes

Table 1. Specifications of the Personal Computers Comprising the Multi-GPU Cluster System.

CPU	Intel Core i7 7800X (Clock speed: 3.5 GHz)
Main memory	DDR4-2666 16 GB
OS	Linux (CentOS 7.6 x86_64)
Software	NVIDIA CUDA 10.1 SDK, OpenGL, MPICH 3.2
GPU board	NVIDIA GeForce GTX 1080 Ti

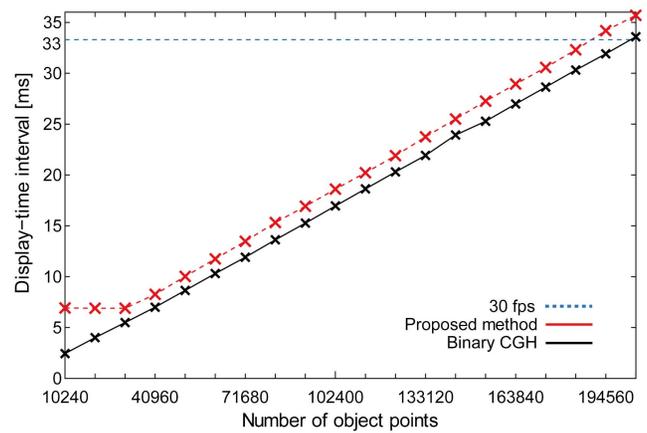


Fig. 10. Eight-gradation holographic 3D video reproduction using the bit planes  $B^0$ ,  $B^1$ , and  $B^2$  on a DMD.

$B^0$ ,  $B^1$ , and  $B^2$ . Here, the resolution of the calculated CGH is 1920 (pixels)  $\times$  1024 (pixels). The black and red lines show the single-frame display-time intervals of the conventional electroholography using a binary CGH<sup>[17]</sup> and the eight-gradation-expressible electroholography using the proposed method, respectively. In the proposed method, the three bit planes  $B^0$ ,  $B^1$ , and  $B^2$  are transferred at each frame. In Fig. 10, the transfer time per frame of the proposed method is approximately 6.0 ms, which results in a bottleneck, as compared with the calculation time of the three bit planes in the range of 30,720 points or less. In the range of 30,720 points or more, the single-frame display-time interval when using the proposed method falls within about 1.1 times that of using the conventional method<sup>[17]</sup> without the gradation expression, although the proposed method requires three BW-CGHs per frame. The proposed method realizes real-time eight-gradation-expressible holographic video reconstruction of a 3D object comprising approximately 180,000 points.

Figure 11 shows the optical setup, for which a 532 nm laser is used as a light source, with objective and collimator lenses used

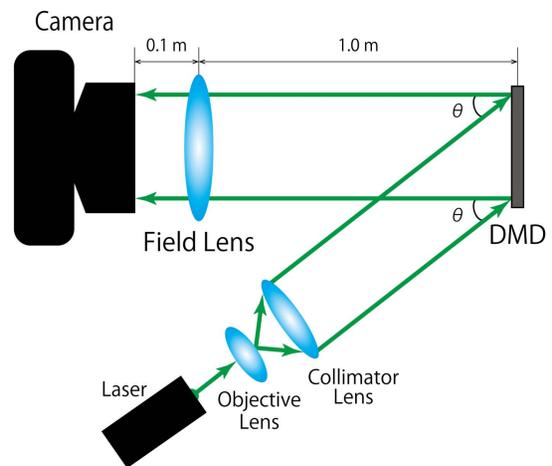


Fig. 11. Optical setup used in the evaluation experiment.

to generate parallel light beams from the light source. We used DLP LightCrafter 6500 EVM (Texas Instruments, micromirror pixel pitch: 7.6  $\mu\text{m}$ , micromirror array size: 1920  $\times$  1080) as a DMD module. The parallel light is obliquely incident to the DMD panel with an angle of incidence  $\theta$ , which serves as the reconstructed light. In this experiment, we set  $\theta$  to 24°. We placed the 3D point cloud models 1.0 m from the CGH and used a Canon EOS R6 digital camera.

We used “Jack-o’-lantern” and “Stanford bunny” as 3D videos. The 3D videos “Jack-o’-lantern” and “Stanford bunny” have 182,357 to 163,764 and 181,782 to 68,481 points, respectively. Figure 12 shows the gradation values of the 3D videos. The gradation values of the 3D models decrease in the z-axis direction from the front to the back. In every frame of the 3D videos, the respective areas with the respective gradation values remain the same, as shown in Fig. 12. The sizes of the reconstructed 3D videos “Jack-o’-lantern” and “Stanford bunny” are 4.5 cm  $\times$  4.5 cm  $\times$  5.0 cm and 3.0 cm  $\times$  3.5 cm  $\times$  4.0 cm, respectively.

Figures 13 and 14 show snapshots of the reconstructed holographic 3D videos “Jack-o’-lantern” (Data File 1) and “Stanford bunny” (Data File 2) using the proposed method. Here, in Data File 1, we set the gradation values  $G^2$ ,  $G^1$ , and  $G^0$  to

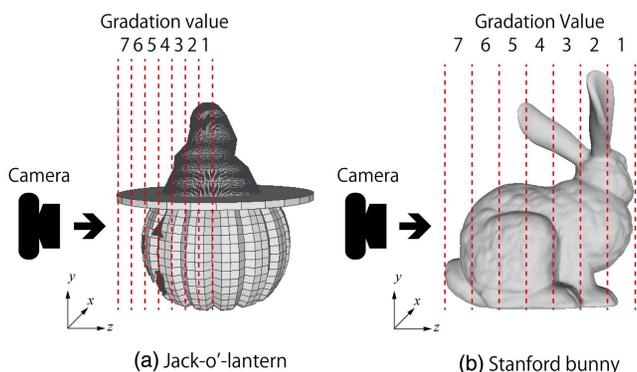


Fig. 12. Gradation values of 3D model “Jack-o’-lantern” and “Stanford bunny.”

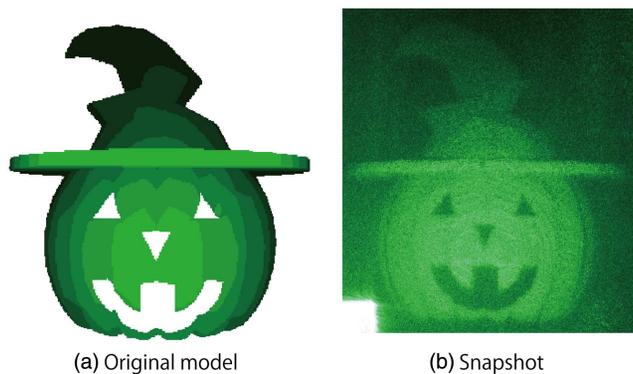


Fig. 13. Snapshot of a reconstructed 3D video “Jack-o’-lantern” (Data File 1).

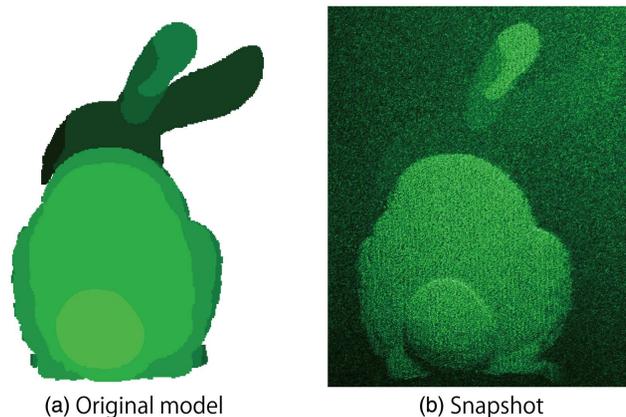


Fig. 14. Snapshot of a reconstructed 3D video “Stanford bunny” (Data File 2).

170, 110, and 110, respectively. In Data File 2, we set the gradation values  $G^2$ ,  $G^1$ , and  $G^0$  to 170, 40, and 25, respectively.

Table 2 shows the number of object points of 3D models “Jack-o’-lantern” in Fig. 13 and “Stanford bunny” in Fig. 14. In the duplicate calculation method shown in Fig. 4, Lists  $B^0$ ,  $B^1$ , and  $B^2$  assigned to bit planes  $B^0$ ,  $B^1$ , and  $B^2$  are used. From Table 2, the total number of the assigned object points of 3D models “Jack-o’-lantern” and “Stanford bunny” is 295,712 and 128,917, respectively. In the proposed method, the object points assigned Groups 1 to 7 were used. From Table 2, the total number of the assigned object points of 3D models “Jack-o’-lantern” and “Stanford bunny” is 165,600 and 68,489, respectively. From Eq. (2), the respective theoretical speed-up of the 3D models “Jack-o’-lantern” and “Stanford bunny” is 1.79 and 1.88, respectively.

Table 2. Number of the Object Points of 3D Models for the Duplicate Calculation and the Proposed Methods.

	Number of Object Points	
	Jack-o'-Lantern	Stanford Bunny
$B^0$	96,960	54,302
$B^1$	100,840	52,758
$B^2$	97,912	21,857
Group 7	28,488	7380
Group 6	26,560	31,969
Group 6	23,512	11,285
Group 5	19,352	3668
Group 4	23,064	2414
Group 2	22,728	10,995
Group 1	21,896	778

**Table 3.** Comparison of the Display Time Interval using a PC with a Single GPU.

	Display Time Interval [ms]		Speed-up
	Duplicate Calculation Method	Proposed Method	
Jack-o'-lantern	591.18	335.95	1.76
Stanford bunny	264.15	141.53	1.87

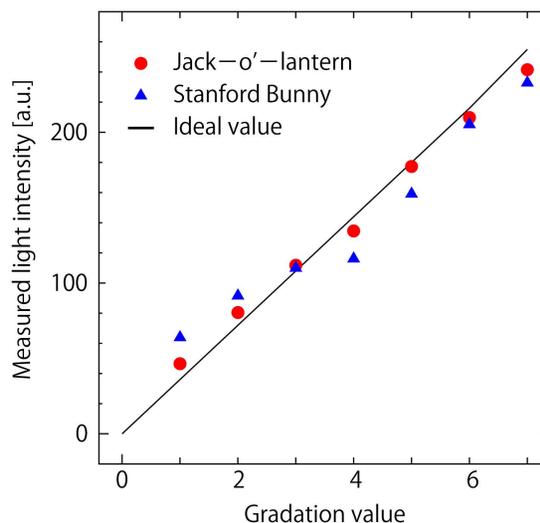
**Table 4.** Comparison of the Display Time Interval using the Multi-GPU Cluster System.

	Display Time Interval [ms]		Speed-up
	Duplicate Calculation Method	Proposed Method	
Jack-o'-lantern	57.21	29.89	1.91
Stanford bunny	29.87	13.49	2.21

In a PC with a single GPU (Table 1), we compared the performance of the proposed method with that of the duplicate calculation method. Table 3 shows the respective display time intervals of the duplicate calculation method and the proposed method using a PC with a single GPU. In two 3D models, the speed-up of the proposed method exceeds 98% of the theoretical speed-up.

In the multi-GPU cluster system consisting of a CGH display node with a single GPU board and four CGH calculation nodes with three GPUs (Table 1), we compared the performances of the proposed method with that of the duplicate calculation method. Here, in the duplicate calculation method, each frame of the 3D video was assigned to each CGH calculation node. At each frame, the bit planes  $B^0$ ,  $B^1$ , and  $B^2$  were assigned and calculated by the respective GPUs of each CGH calculation node. Table 4 shows the respective display time intervals of the duplicate calculation and the proposed methods using the multi-GPU cluster system. In two 3D models, the speed-up of the proposed method exceeds the theoretical speed-up. In the duplicate calculation method, at each CGH calculation node, all GPUs do not calculate the next frame until they have finished calculating all bit planes  $B^0$ ,  $B^1$ , and  $B^2$  at the present frame. Therefore, the performance of the duplicate calculation method using a multi-GPU cluster system is degraded. In the proposed method, Tables 3 and 4 show that the multi-GPU cluster system with 13 GPUs is about 11 times faster than a PC with a single GPU.

Figure 15 shows the measured light intensities obtained from the snapshots of the reconstructed 3D videos (Data File 1 and Data File 2). Using our proposed method, we realized a reconstructed 3D video expressed in eight gradations.

**Fig. 15.** Measured light intensities obtained from the snapshots of the reconstructed 3D videos (Data File 1 and Data File 2).

## 4. Conclusion

In our previous work, we used BW-CGHs as bit planes to realize gradation-expressible amplitude-modulation-type electroholography without controlling the intensity of the reconstructed light. In this Letter, we proposed the efficient and high-speed BW-CGH calculation of real-time gradation-expressible electroholography with an amplitude-modulation-type SLM. By generating BW-CGHs as bit planes, the proposed method avoids duplicate calculation of the light intensities of object points with various gradation values.

We implemented the proposed method on a multi-GPU cluster system comprising 13 GPUs and a DMD. Although eight-gradation-expressible electroholography requires three BW-CGHs per frame, using the proposed method, we realized eight-gradation-expressible electroholography at approximately the same calculation speed as conventional electroholography without gradation expression using a binary CGH. Consequently, we were able to successfully reconstruct a real-time electroholographic 3D video comprising approximately 180,000 points expressed in eight gradations at 30 frames per second.

## Acknowledgement

This work was partially supported by the Japan Society for the Promotion of Science (JSPS) KAKENHI (No. 21K11996) and I-O DATA Foundation.

## References

1. S. A. Benton and J. V. M. Bove, *Holographic Imaging* (Wiley, 2008).
2. N. Hashimoto, S. Morokawa, and K. Kitamura, "Real-time holography using the high-resolution LCTV-SLM," *Proc. SPIE* **1461**, 291 (1991).
3. K. Sato, K. Higuchi, and H. Katsuma, "Holographic television by liquid crystal devices," *Proc. SPIE* **1667**, 19 (1992).

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, and T. Ito, "High-performance parallel computing for next generation holographic imaging," *Nat. Electron.* **1**, 254 (2018).
5. Y. Pan, X. Xu, S. Solanki, X. Liang, R. B. A. Tanjung, C. Tan, and T.-C. Chong, "Fast CGH computation using S-LUT on GPU," *Opt. Express* **17**, 18543 (2009).
6. P. Tsang, W. K. Cheung, T.-C. Poon, and C. Zhou, "Holographic video at 40 frames per second for 4-million object points," *Opt. Express* **19**, 15205 (2011).
7. F. Yaraş, H. Kang, and L. Onural, "Real-time phase-only color holographic video display system using LED illumination," *Appl. Opt.* **48**, H48 (2009).
8. T. Sugawara, Y. Ogihara, and Y. Sakamoto, "Fast point-based method of a computer-generated hologram for a triangle-patch model by using a graphics processing unit," *Appl. Opt.* **55**, A160 (2016).
9. M.-W. Kwon, S.-C. Kim, and E.-S. Kim, "GPU-based implementation of one-dimensional novel-look-up-table for real-time computation of Fresnel hologram patterns of three-dimensional objects," *Opt. Eng.* **53**, 035103 (2014).
10. Y. Sando, K. Satoh, D. Barada, and T. Yatagai, "Real-time interactive holographic 3D display with a 360° horizontal viewing zone," *Appl. Opt.* **58**, G1 (2019).
11. N. Takada, T. Shimobaba, H. Nakayama, A. Shiraki, N. Okada, M. Oikawa, N. Masuda, and T. Ito, "Fast high-resolution computer-generated hologram computation using multiple graphics processing unit cluster system," *Appl. Opt.* **51**, 7303 (2012).
12. Y. Pan, X. Xu, and X. Liang, "Fast distributed large-pixel-count hologram computation using a GPU cluster," *Appl. Opt.* **52**, 6562 (2013).
13. B. J. Jackin, H. Miyata, T. Ohkawa, K. Ootsu, T. Yokota, Y. Hayasaki, T. Yatagai, and T. Baba, "Distributed calculation method for large pixel-number holograms by decomposition of object and hologram planes," *Opt. Lett.* **39**, 6867 (2014).
14. B. J. Jackin, S. Watanabe, K. Ootsu, T. Ohkawa, T. Yokota, Y. Hayasaki, T. Yatagai, and T. Baba, "Decomposition method for fast computation of gigapixel-sized Fresnel holograms on a graphics processing unit cluster," *Appl. Opt.* **57**, 3134 (2018).
15. T. Baba, S. Watanabe, B. J. Jackin, K. Ootsu, T. Ohkawa, T. Yokota, Y. Hayasaki, and T. Yatagai, "Fast computation with efficient object data distribution for large-scale hologram generation on a multi-GPU cluster," *IEICE Trans. Inf. Sys.* **E102-D**, 1310 (2019).
16. H. Niwase, N. Takada, H. Araki, Y. Maeda, M. Fujiwara, H. Nakayama, T. Kakue, T. Shimobaba, and T. Ito, "Real-time electroholography using a multiple-graphics processing unit cluster system with a single spatial light modulator and the InfiniBand network," *Opt. Eng.* **55**, 093108 (2016).
17. H. Sannomiya, N. Takada, T. Sakaguchi, H. Nakayama, M. Oikawa, Y. Mori, T. Kakue, T. Shimobaba, and T. Ito, "Real-time electroholography using a single spatial light modulator and a cluster of graphics-processing units connected by a gigabit Ethernet network," *Chin. Opt. Lett.* **18**, 020902 (2020).
18. H. Sannomiya, N. Takada, K. Suzuki, T. Sakaguchi, H. Nakayama, M. Oikawa, Y. Mori, T. Kakue, T. Shimobaba, and T. Ito, "Real-time spatiotemporal division multiplexing electroholography for 1,200,000 object points using multiple-graphics processing unit cluster," *Chin. Opt. Lett.* **18**, 070901 (2020).
19. M. L. Hueschman, B. Munjuluri, and H. R. Garner, "Dynamic holographic 3-D image projection," *Opt. Express* **11**, 437 (2003).
20. M. Chlipala and T. Kozacki, "Color LED DMD holographic display with high resolution across large depth," *Opt. Lett.* **44**, 4255 (2019).
21. J.-P. Liu, M.-H. Wu, and P. W. M. Tsang, "3D display by binary computer generated holograms with localized random down-sampling and adaptive intensity accumulation," *Opt. Express* **28**, 24526 (2020).
22. S. Jiao, D. Zhang, C. Zhang, Y. Gao, T. Lei, and X. Yuan, "Complex-amplitude holographic projection with a digital micromirror device (DMD) and error diffusion algorithm," *IEEE J. Sel. Top. Quantum Electron.* **26**, 2800108 (2020).
23. K. Min and J.-H. Park, "Quality enhancement of binary-encoded amplitude holograms by using error diffusion," *Opt. Express* **28**, 38140 (2020).
24. Y. Takaki and N. Okada, "Hologram generation by horizontal scanning of a high-speed spatial light modulator," *Appl. Opt.* **48**, 3255 (2009).
25. Y. Takaki and N. Okada, "Reduction of image blurring of horizontally scanning holographic display," *Opt. Express* **18**, 11327 (2010).
26. Y. Takaki and K. Fujii, "Viewing-zone scanning holographic display using a MEMS spatial light modulator," *Opt. Express* **22**, 24713 (2014).
27. Y. Takekawa, Y. Takashima, and Y. Takaki, "Holographic display having a wide viewing zone using a MEMS SLM without pixel pitch reduction," *Opt. Express* **28**, 7392 (2020).
28. Y. Takaki and M. Yokouchi, "Speckle-free and grayscale hologram reconstruction using time-multiplexing technique," *Opt. Express* **19**, 7567 (2011).
29. M.-C. Park, B.-R. Lee, J.-Y. Son, and O. Chernyshov, "Properties of DMDs for holographic displays," *J. Modern Opt.* **62**, 1600 (2015).
30. M. Fujiwara, N. Takada, H. Araki, S. Ikawa, H. Niwase, Y. Maeda, H. Nakayama, T. Kakue, T. Shimobaba, and T. Ito, "Gradation representation method using binary-weighted computer-generated hologram," *Opt. Eng.* **56**, 023105 (2017).
31. M. Fujiwara, N. Takada, H. Araki, C. W. Ooi, S. Ikawa, Y. Maeda, H. Niwase, T. Kakue, T. Shimobaba, and T. Ito, "Gradation representation method using binary-weighted computer-generated hologram based on pulse width modulation," *Chin. Opt. Lett.* **15**, 060901 (2017).
32. M. Fujiwara, N. Takada, H. Araki, S. Ikawa, Y. Maeda, H. Niwase, M. Oikawa, T. Kakue, T. Shimobaba, and T. Ito, "Color representation method using RGB color binary-weighted computer-generated holograms," *Chin. Opt. Lett.* **16**, 080901 (2018).
33. D. Dudley, W. M. Duncan, and J. Slaughter, "Emerging digital micromirror device (DMD) applications," *Proc. SPIE* **4985**, 14 (2003).