GPU Acceleration of Algorithm to Design Directional Volumetric Display for Real-time Processing

Daiki Matsumoto[1], Ryuji Hirayama[2,3], Naoto Hoshikawa[4], Hirotaka Nakayama[5], Tomoyoshi Shimobaba[1], Tomoyoshi Ito[1], Atsushi Shiraki[1]

[1] Chiba University, 1-33, Yayoi-cho, Inage-ku, Chiba 263-8522, Japan
[2] Research Fellow of the Japan Society for the Promotion of Science, 5-3-1 Kojimachi, Chiyoda-ku Tokyo 102-0083, Japan
[3] Tokyo University of Science, 6-3-1 Niijuku, Katsushika-ku, Tokyo 125-8585, Japan
[4] National Institute of Technology, Oyama College, 771 Nakakuki, Oyama, Tochigi 323-0806, Japan
[5] National Astronomical Observatory of Japan, 2-21-1, Osawa, Mitaka, Tokyo 181-8588, Japan
Keywords: GPU acceleration, Volumetric display, Directional image, Digital signage, Media art.

ABSTRACT

In this study, we attempted GPU acceleration of an algorithm to design a directional volumetric display. As a result, the GPU implementation was up to 45 times faster than the CPU implementation. We also confirmed that the GPU implementation could cooperate with a person tracking system in real-time.

1 INTRODUCTION

A volumetric display [1] can draw three-dimensional (3D) images in real space. In previous studies [2-4], we proposed an algorithm to design a directional volumetric display shown in Fig. 1. Figure 1(a) shows an outward appearance of the directional volumetric display using threads and a projector. The directional volumetric display can exhibit multiple directional images in different designated directions as shown in Fig. 1(b) and 1(c). However, we cannot observe these images from undesignated directions. By taking advantage of this characteristic, we aim to develop an information communication system which can track a person and exhibit a directional image only to the person in real-time as shown in Fig. 2.

(a) (b) (c)

Fig. 1 Directional volumetric display using threads and a projector [2]

(a) outward appearance, (b) observation from front, (c) observation from side.

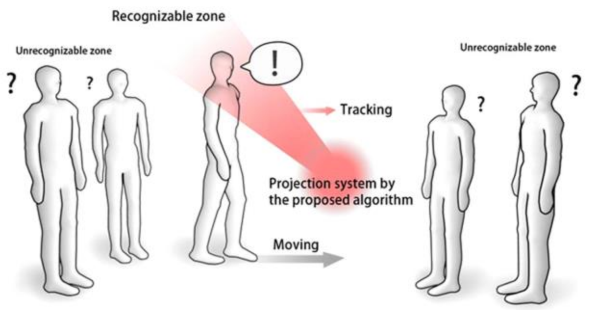


Fig. 2 Information communication system which we aim to develop [2]

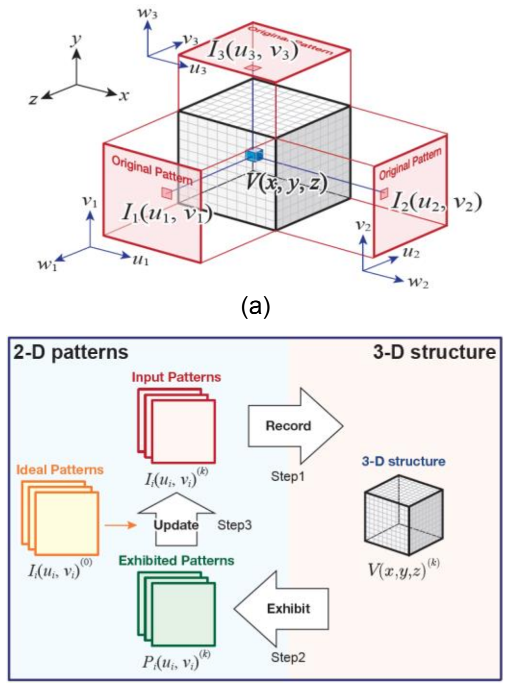
To develop this information communication system, we must compute each volume element (voxel) of the directional volumetric display in real-time. However, the algorithm to design the directional volumetric display is computational heavy. This is because the algorithm computes the 3D voxel values from pixel values of two-dimensional images. Besides, an iteration process of the algorithm to improve the image quality of exhibited images increases processing time. Thus, there are some limitations on image resolution, the number of iterations, and the number of images for real-time processing. Nevertheless, in the algorithm, each voxel value can be computed independently. Therefore, we can accelerate the algorithm by parallel computing using graphics processing unit (GPU) [5]. In this study, we implement the algorithm by using the GPU and accelerate the algorithm to relax the limitations for real-time processing. To confirm that the GPU implementation can process the algorithm in real-time, we also fabricate the directional volumetric display with a person tracking system.

2 METHODS

In this section, we describe the algorithm to design the directional volumetric display and the GPU implementation of the algorithm.

2.1 Algorithm

Figure 3 shows a scheme of the algorithm to design the directional volumetric display. Figure 3(a) shows a relationship between images and voxels. An iteration process to improve the image quality of exhibited images is shown in Fig. 3(b). Two algorithms, multiplication algorithm [2,3] and addition algorithm [4], to compute the iteration process was proposed. The multiplication



(a)

(b)

Fig. 3 Scheme of algorithm to design directional volumetric display [3]
(a) relationship between images and voxels,
(b) iteration process to improve the image quality of exhibited images.

algorithm has a limitation on ideal images, whereas the addition algorithm can record any ideal images. In this study, we use the addition algorithm from a perspective of practical use.

In the addition algorithm, the iteration process is computed by Eqs. (1)-(3).

$$V(x,y,z)^{(k)} = \sum_{i=1}^{N} I_i(u_i,v_i)^{(k)} , \quad (1)$$

$$P_i(u_i,v_i)^{(k)} = \sum_{w_i} V(x,y,z)^{(k)} , \quad (2)$$

$$I_i(u_i,v_i)^{(k+1)} = I_i(u_i,v_i)^{(k)} + \left(I_i(u_i,v_i)^{(0)} - P_i(u_i,v_i)^{(k)}\right) . \quad (3)$$

Here, k denotes the number of iterations, and N denotes the number of the ideal images. In Eq. (1), the voxel values $V(x,y,z)^{(k)}$ are determined by adding the pixel values of the input images $I_i(u_i,v_i)^{(k)}$. In Eq. (2), the pixel values of the exhibited images $P_i(u_i,v_i)^{(k)}$ are determined by adding the voxel values along the exhibit axis (w_i). In Eq. (3), the pixel values of the input images are updated by using the difference between the pixel values of the ideal images $I_i(u_i,v_i)^{(0)}$ and the exhibited images. In Eqs. (1) and (2), every voxel value is used for computation; therefore, the computational cost of the algorithm becomes high.

2.2 GPU implementation

We use NVIDIA GeForce 1050 and compute unified device architecture (CUDA) for the GPU implementation. The technical specifications of the NVIDIA GeForce 1050 are shown in Table 1. For parallel computing, each GPU thread is assigned to each voxel or pixel. We use Intel Core i5-6500 as a central processing unit (CPU). The technical specifications of the Intel Core i5-6500 are shown in Table 2.

To confirm the effectiveness of the GPU acceleration, we compare processing time of a projection system using the CPU implementation and the GPU implementation for fabricating the directional volumetric display using threads and a projector [6]. For the CPU implementation, Microsoft Visual C++ and OpenMP are used. For measuring the processing time, we use 24-bit color images. Additionally, to confirm that the GPU implementation can process the algorithm in real-time, we fabricate the directional volumetric display using threads and a projector with a person tracking system. The directional volumetric display with a person tracking system tracks a person and displays a directional image only to the person in real-time.

Table 1 The technical specifications of NVIDIA GeForce GTX 1050

GPU clack rate	1354 MHz
GPU cores	640 cores
Memory clock rate	3540 MHz
Global memory	2G GDDR5

Table 2 The technical specifications of Intel Core i5-6500

Base frequency	3.20 GHz
Cores / Threads	4 cores / 4 threads

3 RESULTS

3.1 Processing time of projection system

First, we present results of the GPU acceleration. In this study, we measured processing time with 1,5, and 10 iteration(s), because Structural Similarity (SSIM) [7] values of the exhibited images were converged with 5 to 10 iterations as shown in Fig. 4. The processing time comparison between the CPU implementation and the GPU implementation with two and three ideal images are shown in Table 3 and 4, respectively. As shown in Table 3 and 4, the GPU implementation was 24 times faster on average and up to 45 times faster than the CPU implementation. Figure 5 shows the processing time using ideal images which consist of 64 × 64 pixels. As shown in Fig. 5, we also confirmed that the GPU implementation could process the algorithm in 30 fps with up to five iterations and six ideal images which consist of 64 × 64 pixels.

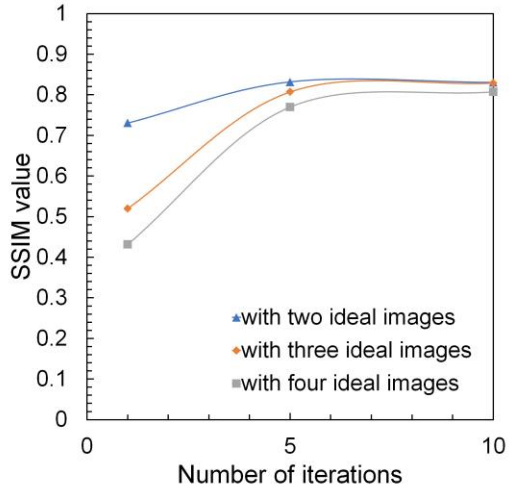


Fig. 4 Convergence of SSIM values of exhibited images accompanying the number of iterations

Table 3 Processing time comparison between the CPU and GPU implementation with two ideal images

Image resolution	Number of iterations	Processing time [ms] CPU	Processing time [ms] GPU	Speed-up
20×20	1	22.28	1.44	×15.45
	5	29.69	1.93	×15.42
	10	33.23	2.48	×13.41
64×64	1	39.78	1.45	×27.45
	5	241.48	9.47	×25.49
	10	499.52	16.89	×29.58
128×128	1	245.49	8.73	×28.13
	5	2297.23	92.88	×24.73
	10	4906.59	199.91	×24.54

Table 4 Processing time comparison between the CPU and GPU implementation with three ideal images

Image resolution	Number of iterations	Processing time [ms] CPU	Processing time [ms] GPU	Speed-up
20×20	1	30.24	1.52	×19.87
	5	37.13	2.16	×17.18
	10	41.93	2.72	×15.43
64×64	1	72.01	1.58	×45.68
	5	336.71	13.30	×25.32
	10	709.79	25.68	×27.64
128×128	1	306.59	10.08	×30.41
	5	2780.54	128.94	×21.56
	10	5818.23	286.76	×20.29

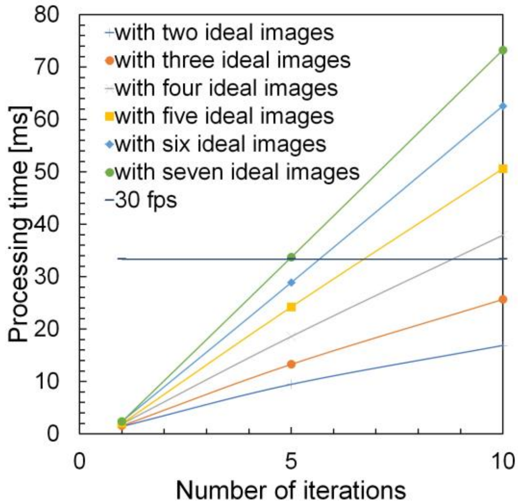


Fig. 5 Processing time using ideal images which consist of 64 × 64 pixels

3.2 Directional volumetric display with a person tracking system

We present observation results of the directional volumetric display using threads and a projector with a person tracking system. Figure 6 shows the observation results of the directional volumetric display. In this study, we used Fig. 6(a) and 6(b) as ideal image 1 and 2. Exhibited image 1 was displayed in a fixed direction. Exhibited image 2 was displayed in a direction of the moving person. As a result, we confirmed that the exhibited image 1 could be observed as shown in Fig. 6(c). At the same time, the exhibited image 2 could be observed from moving visual points as shown in Fig. 6(d), 6(e), and 6(f). However, we could not observe any image from undesignated angles as shown in Fig. 6(g).

4 CONCLUSIONS

In this study, we attempted the GPU acceleration of the algorithm to design the directional volumetric display. As a result, we confirmed that the GPU implementation becomes up to 45 times faster than the CPU implementation. We also confirmed that the GPU implementation could process the algorithm in 30 fps with up to five iterations and six ideal images which consist of 64 × 64 pixels. Based on the results, it is noted that the GPU acceleration relaxed the limitations on image resolution, the number of iterations, and the number of the ideal images for real-time processing

Also, we confirmed that the GPU implementation could process the algorithm in real-time and cooperate with a person tracking system. The directional volumetric display with a person tracking system can display a directional image only to the person. This characteristic can be utilized as digital signage and media art. For example, by displaying information in the language of each observer, the directional volumetric display could be utilized as multilingual signage. However, the

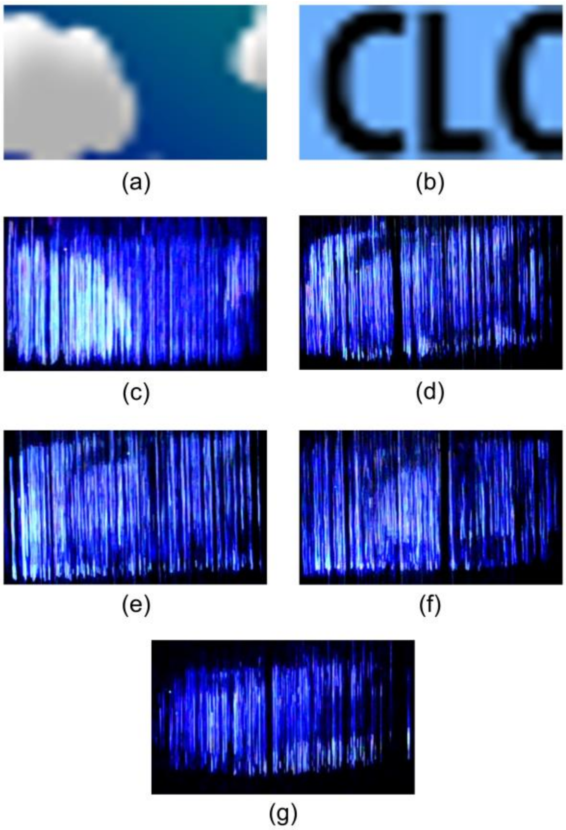


Fig. 6 Observation results of directional volumetric display using threads and a projector with a person tracking system

(a) ideal image 1, (b) ideal image 2,
(c) observation of exhibited image 1 from fixed angle,
(d) observation of exhibited image 2 from moving visual point 1,
(e) observation of exhibited image 2 from moving visual point 2,
(f) observation of exhibited image 2 from moving visual point 3,
(g) observation from undesignated angle.

directional volumetric display we fabricated in this study displayed images which consist of 20 × 20 pixels. To utilize the directional volumetric display as practical applications, we must develop a volumetric display which can display high-resolution images.

ACKNOWLEDGEMENT

This work was supported by JSPS KAKENHI Grant Number 18K11599.

REFERENCES

[1] D. MacFarlane, "Volumetric three-dimensional display," Applied Optics, Vol. 33, No. 31, pp. 7453-7457 (1994).

[2] H. Nakayama, A. Shiraki, R. Hirayama, N. Masuda, T. Shimobaba and T. Ito, "Three-dimensional volume containing multiple two-dimensional information patterns," Scientific Reports, Vol. 3, No. 1931, pp. 1-5 (2013).

[3] R. Hirayama, H. Nakayama, A. Shiraki, T. Kakue, T. Shimobaba and T. Ito, "Image quality improvement for a 3D structure exhibiting multiple 2D patterns and its implementation," Optics Express, Vol. 24, No. 7, pp. 7319-7327 (2016).

[4] A. Shiraki, D. Matsumoto, R. Hirayama, H. Nakayama, T. Kakue, T. Shimobaba, and T. Ito, "Improvement of an algorithm for displaying multiple images in one space," Applied Optics, Vol. 58, No. 5, pp. A1-A6 (2019).

[5] J. D. Owens, M. Houston, D. Luebke, S. Green, J. E. Stone and J. C. Phillips, "GPU Computing," Proceedings of the IEEE, Vol. 96, No. 5, pp. 879-899 (2008).

[6] A. Shiraki, M. Ikeda, H. Nakayama, R. Hirayama, T. Kakue, T. Shimobaba, and T. Ito, "Efficient method for fabricating a directional volumetric display using strings displaying multiple images, " Applied Optics, Vol. 57, No. 1, pp. A33-A38 (2018).

[7] Z. Wang, A. C. Bovik, H. R. Sheikh and E. P. Simoncelli, "Image quality assessment: from error visibility to structural similarity," IEEE Transactions on Image Processing, Vol. 13, No. 4, pp. 600-612 (2004).