https://doi.org/10.5392/IJoC.2018.14.3.001

Design and Implementation of High-Resolution Integral Imaging Display System using Expanded Depth Image

Min-Ho Song, Byung-Muk Lim, Ga-A Ryu
Department of Computer Science,
Chungbuk National University, Cheongju, South Korea

Jong-Sung Ha
Department of Computer Engineering,
Woosuk University, Chonbuk, South Korea

Kwan-Hee Yoo
Department of Computer Science,
Chungbuk National University, Cheongju, South Korea

ABSTRACT

For 3D display applications, auto-stereoscopic display methods that can provide 3D images without glasses have been actively developed. This paper is concerned with developing a display system for elemental images of real space using integral imaging. Unlike the conventional method, which reduces a color image to the level as much as a generated depth image does, we have minimized original color image data loss by generating an enlarged depth image with interpolation methods. Our method was efficiently implemented by applying a GPU parallel processing technique with OpenCL to rapidly generate a large amount of elemental image data. We also obtained experimental results for displaying higher quality integral imaging rather than one generated by previous methods.

Key words*: Integral Imaging, 3D Image Acquisition, Elemental Images, Depth Image, GPU Parallel Programming.*

1. INTRODUCTION

Autostereoscopic display systems have focused on various applications due to the convenience. Many research has been conducted, such as [1]-[3]. Generally, in integral imaging system, a 3D scene from many different perspectives is captured as individual images obtained by the array of microlenses, called elemental images (EIs). In most of the previous research related to integral imaging, integral images have been generated by using virtual lens arrays [4]-[6]. This method lacks a sense of reality and is limited for developing interaction-based contents since it produces 3D images from virtual objects in a virtual space instead of real space. To resolve these problems, integral imaging can be produced from images taken with depth cameras that can store real space. This approach can be utilized in more applications [7], [8].

* *Corresponding author, Email: khyoo@chungbuk.ac.kr*
Manuscript received May. 18, 2018; revised Jul. 31, 2018; accepted Jul. 31, 2018

An advantage of integral imaging is its simplicity in obtaining 3D information without costly devices and demanding conditions. Our research has made the best use of Kinect V1 and Kinect V2 [15], which are the main devices for extracting information from real spaces at present, generating 640x480 color images / 320x240 depth images, and 1920x1080 color images / 512x424 depth images, respectively. As the first stage work, Li et al. [7] created elemental images from depth images and color images extracted by a Kinect V1 device. And Jeong et al. [8] developed a real-time system to create elemental images based on GPU parallel processing using the same depth camera. Particularly, Jeong et al. [8] made use of 320x240 reduced resolution instead of the original 640x480 resolution for color images in order to fit into the resolution of depth images, which reduces the quality of the final elemental images.

There are a number of attempts to increase the image quality of elemental images for a real scene [9]-[11]. Specially, Zhang, et.al [9] tried to generate elemental images with a lot of voxels based on KinectFusion, and Kwon, et.al [10] tried to enlarge elemental images themselves by interpolation method from a generated elemental image. Because of the low quality of a given real scene, essentially, there is a limit to generate elemental images with good quality. To resolve the underlying problem, in this paper, we took a reverse approach to expand the resolution of depth images according to that of color images.

In section 2, we describe a method to generate elemental images for high-quality integral imaging. Section 3 presents experimental results that demonstrate that our method can improve the image quality in comparison to previous methods. Finally, we summarize our approach and suggest directions for further research in section 4.

2. INTEGRAL IMAGING DISPLAY SYSTEM USING EXPANDED DEPTH IMAGES

As illustrated in Fig. 1, high-resolution integral imaging display system using expanded depth images is composed of three stages. In the first stage, color images and depth images are captured from a real scene, and depth images are expanded by interpolation.

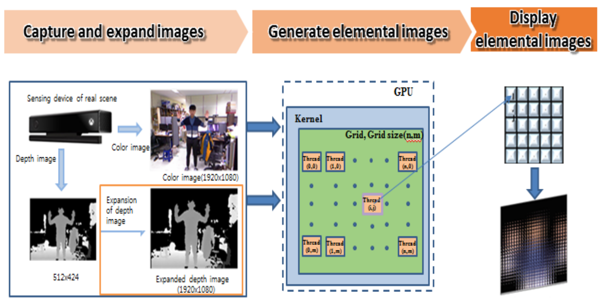


Fig. 1. Global flow of our proposed method

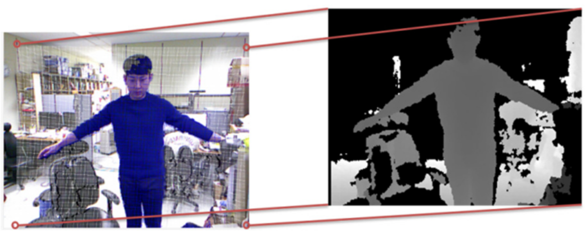


Fig. 2. Depth image mapped onto a color image

In the second stage, elemental images are generated from color images and expanded depth images. In the final stage, elemental images are displayed with the array of lenses in 3D autostereoscopic. In the first stage of capturing and expanding images, depth images and color images captured with a depth camera, Kinect V2, have the resolutions of 1920x1080 and 512x424, respectively. Since the depth resolution is lower than the color resolution, the elemental images generated then lose a lot of 3D information. To resolve this problem, we expand the depth images according to the size of the color images.

The simplest way to expand the depth images is to make empty depth images that are the same size as the color images, and then assign each of the depth images to them correspondingly based on the locations of the color images. This method cannot fully describe the real world because of many empty values in the expanded depth images.

Fig. 2 shows a depth image expanded from 512x424 resolution depth image being adapted to a 1920x1080 color image, in which there are a lot of unfilled pixels as shown in the right side of Fig. 2. To resolve the problem of unfilled depths, we consider interpolation methods [12], such as nearest neighbor, bilinear, and bi-cubic interpolation. The bi-cubic interpolation used in the proposed approach assigns weight values using cubic functions in 16 pixels neighboring a pixel to be recovered. We efficiently implemented bi-cubic interpolation with a GPU parallel programming technique [16], [17], which produces vertically interpolated pixels four times and then computes final interpolated pixel in the horizontal direction.

In the second stage, we generate the elemental images from the expanded depth images and color images based on the GPU parallel programming technique. In the method of Jeong et al. [8], a thread corresponding to each pixel of color and depth images is produced to compute the images projected onto a display screen through a whole lens array. An important point of their method is to determine the pixel pitch $P_I=(P_{IX},P_{IY})$ in object space as in Equation 1, where $Zcdp$ is the central depth plane of an object, g is the distance between the display panel and the lens array, $P_S=(P_{SX},P_{SY})$ is the pixel pitch of the LCD:

$$P_I = \frac{Z_{cdp}}{g} \cdot P_s \quad (1)$$

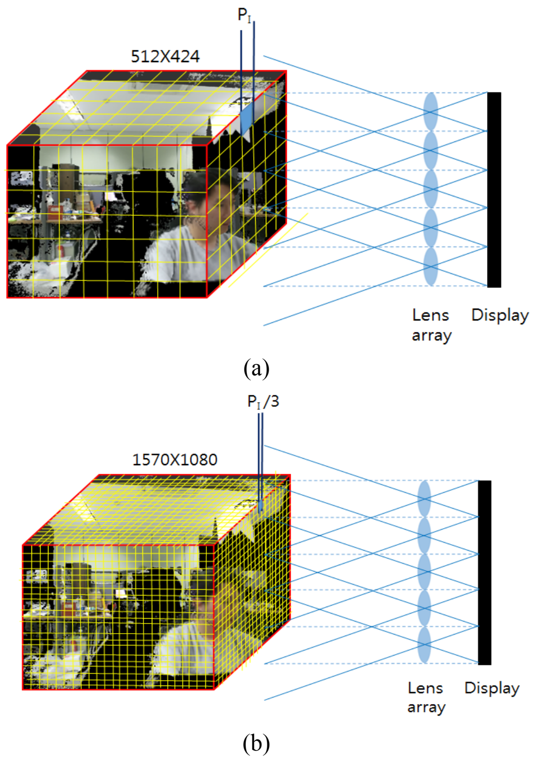


Fig. 3. Process of generating elemental images for (a) 512x424 color and depth images and (b) 1570x1080 color and depth images

Eq. 1 is modified to determine the pixel pitch as Eq. 2, where $RR = (RR_x, RR_y)$ is the resolution ratio. After that the size of color image is reduced to 512x424, Jeong et al. [8] gave the resolution ratio RR of the reduced color images to $RR =$

(1,1). Since the enlarged 1570x1080 depth image is used in this paper, we define that of 1570x1080 color images as RR $= (\frac{512}{1570}, \frac{424}{1080})$:

$$P_I = RR \cdot \frac{Z_{cdp}}{g} \cdot P_s \quad (2)$$

Fig. 3 (a) and (b) show elemental images generated for the color and depth images with different sizes 512x424 and 1570x1080 for the same object space, respectively. Since the resolution of the 1570x1080 images is $RR = (\frac{512}{1570}, \frac{424}{1080})$, their pixel pitch is about three times smaller than that of the 512x424 images as shown in Fig. 3.

The location of the $(u,v)^{th}$ pixel of the elemental images to be generated is determined by using the color and depth information of the $(i,j)^{th}$ pixel of the object space as in Eq. 3-5, which was presented by Jeong et al. [8]. In Eq. 3-5, P_L is the size of the elemental image and $d_c(i,j)$ is the depth of a transformed object, i_L and j_L are the index information of a lens array, respectively, and $\min|d_r|$ and $\max|d_r|$ are the minimum and maximum depths, respectively.

$$u = P_L \cdot i_L - (i \cdot P_I - P_L \cdot i_L) \cdot \frac{g}{d_c(i,j)} \quad (3)$$

$$v = P_L \cdot j_L - (j \cdot P_I - P_L \cdot j_L) \cdot \frac{g}{d_c(i,j)} \quad (4)$$

$$\mathrm{d_c(i,j)} = \frac{\max|\mathrm{d_r}| + \min|\mathrm{d_r}|}{2\mathrm{d_r(i,j)}} \cdot \mathrm{Z_{CDP}} = \frac{max|d_r| + min|d_r|}{2d_r(i,j)} \cdot \frac{f \cdot g}{g - f} \quad (5)$$

To generate elemental images based on GPU parallel programming techniques, a thread for each pixel of an object space is created to compute the color of the corresponding $(u,v)^{th}$ pixel of the elemental images by using Equations (1-5). When we directly adopt the parallel algorithm presented by Jeong et al. [8], it often occurs that two different pixels P_1 and P_2 of the object space may correspond to the same elemental pixel, *DP*, as illustrated in Fig. 4. In this case, the selection of the pixels P_1 and P_2 depends on the order of thread operations, even though the pixel P_2, which is near to the elemental pixel *DP*, should be chosen for the correction. We settled this nondeterministic phenomenon with an algorithm called Z-buffer [13]; the pixel near to the elemental images is selected by managing a buffer storing the depths from each pixel to the elemental images.

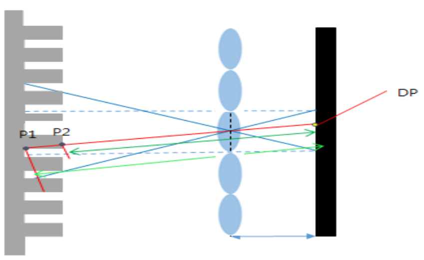


Fig.4. Instance of two different object pixels corresponding to the same elemental pixel

3. EXPERIMENTAL RESULTS

Table 1 describes the detailed environment for implementing and testing our integral imaging method using expanded depth images; software, hardware, depth camera, and lens array. We implemented the proposed method using parallel processing with OpenCL on a 64-bit computer composed of 2304 GPU cores, and depth camera Kinect V2. The 30x30 lens array has been used for testing.

Table1. Experimental environment

Devices & tools	Detailed information / specifications	
Software	Operation system	Microsoft Windows 7 (64-bit)
	Development tool	Microsoft Visual Studio 2010
	Graphics library	OpenCL
	GPU library	OpenCV, C#
Hardware	CPU	Intel(R) Core(TM) i7-4770 3.40GHz
	Memory	12 GB RAM
	GPU	NVIDIA GeForce GTX 780 (Core: 2304)
Depth Camera	Depth image	512x424
	Color image	1920x1080
	Maximum depth distance	4.5M
	Minimum depth distance	40cm
LCD monitor	Pixel pitch	0.16 mm
	Resolution	3840×2160 pixels
Lens array	Size	30x30
	Focal length	10 mm
	Lens pitch	19.25 mm

The sizes of the color and depth images captured with Kinect V2 are 1920x1080 and 512x424, respectively, as illustrated in Fig. 5 (a) and (b). Figure 5 (c) shows a 1570x1080 depth image expanded in such a way that the pixels of the color image are replaced by the pixels of the original depth image if their corresponding pixels exist, and filled with “0” otherwise. The depth image expanded with this simple method is significantly dark, because there are too many pixels that do not correspond to any pixel of the original depth image. On the other hand, the depth image expanded with the bi-cubic interpolation method has a shape that is similar to that of the original depth image, as illustrated in Fig. 5 (d). The figure shows a pixel of the color image is interpolated with a pixel of the original depth image if there is no corresponding pixel of the original depth image.

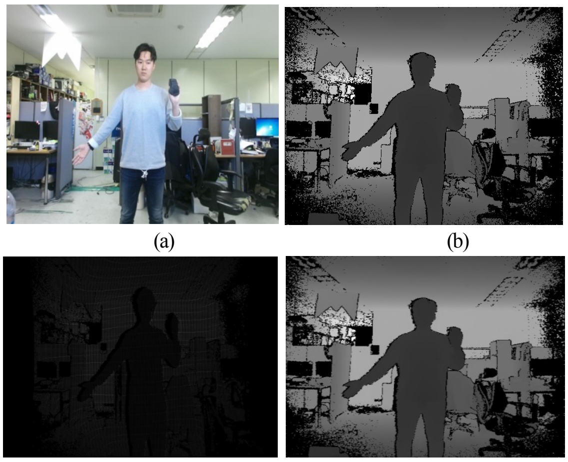

(a) (b) (c) (d)

Fig. 5. 1570x1080 original color image captured with Kinect V2 (a), 512x424 original depth image captured with Kinect V2(b), 1570x1080 depth image expanded in brute-force(c), and 1570x1080 depth image expanded with bi-cubic interpolation(d)

To evaluate elemental images generated by the method proposed in this paper, we define three types, T1, T2, and T3 of test data for a 515x424 original depth image and a 1570x1080 original color image captured with Kinect V2 as follows:

T1: Reducing color image with the method of Jeong et al. [8]: 515x424 original depth image and 515x424 reduced color image.

T2: Expanding depth image with a simple non-interpolation method: 1570x1080 expanded depth image and 1570x1080 original color image.

T3: Expanding depth image with a bi-cubic interpolation method: 1570x1080 expanded depth image and 1570x1080 original color image.

We generated elemental images by the three cases under the same conditions; the size of the lens array was 5 mm, the number of lenses in the array was 30x30, the focus distance was 10 mm, the pixel distance of the display panel was 0.277 mm, and the elemental images were captured from 2 meters away. A few evaluation methods for integral images are proposed [14]. We evaluated the results of the three elemental images and the generated parallax images in terms of whether the whole image, background, human body, head, face, and arms are clearly displayed. The overall evaluation results are described in Table 2, and the details are shown in Fig. 6, Fig. 7 and Fig. 8.

Fig. 6 shows examples of generated elemental images, in which the parallax image size is 20x20 since the size of the elemental images is 600x600, and the size of the lens array is 30x30. To identify the evaluation of image quality in Table 2, we generated parallax images based on the elemental images from the viewpoints of top left, center, and bottom right, as respectively illustrated in Fig. 7. Generally, we can recognize more easily the shapes in Fig. 7 (b) than in Fig. 7 (a). It is verified that Fig. 7 (c) shows the least loss and distortion of images.

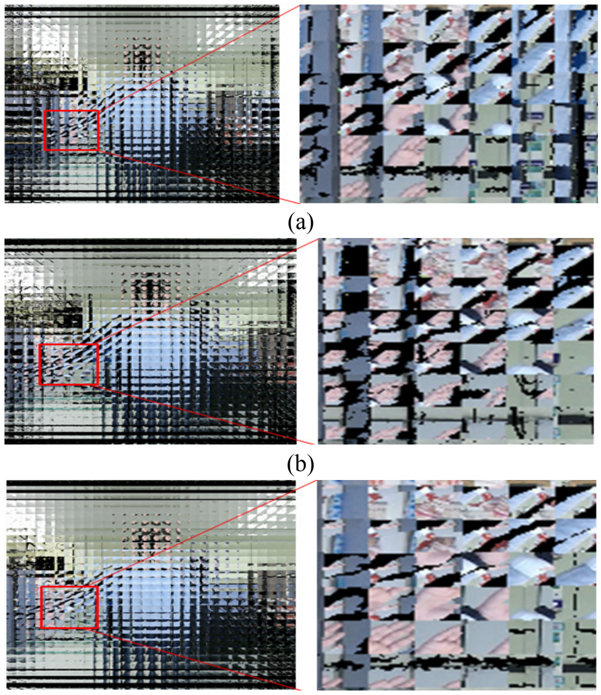

(a) (b) (c)

Fig. 6. Elemental images generated with the color and the depth images, respectively, for three cases, (a) T1, (b) T2 and (c) T3

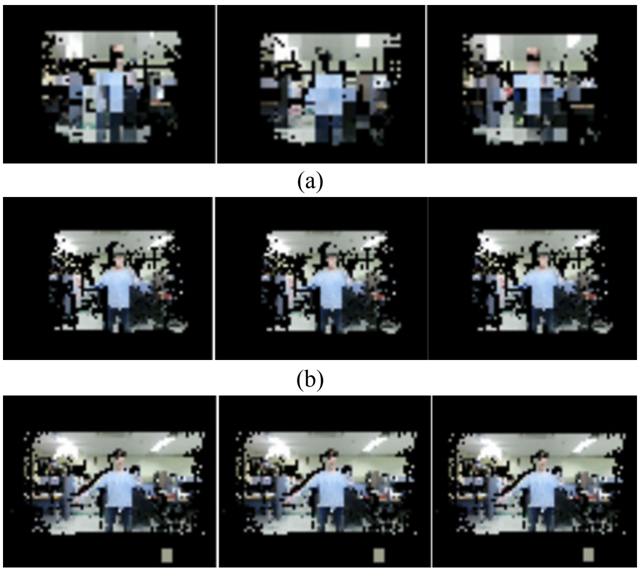
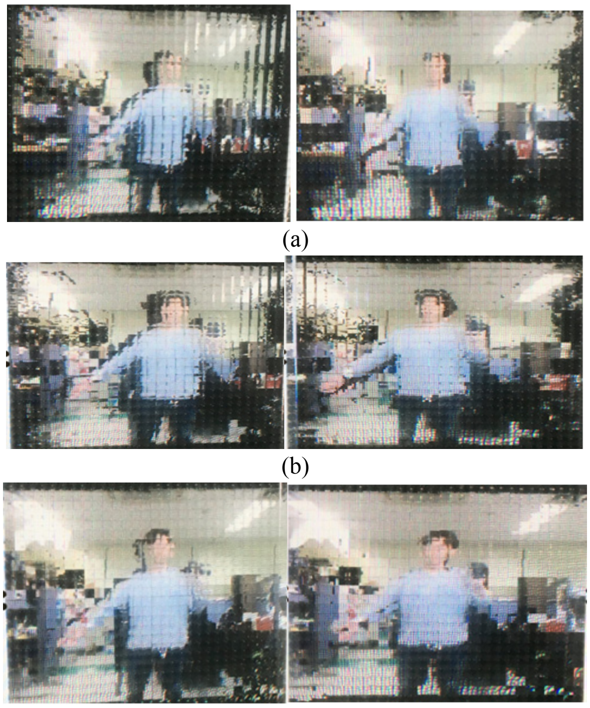

(a) (b) (c)

Fig. 7. Three parallax left, middle and right images based on elemental images from the viewpoints of top left, center, and bottom right, respectively, for three cases, (a) T1, (b) T2 and (c) T3

We tested the quality of auto-stereo image by visualizing elemental images with a 30x30 lens array without special glasses. Figure 8 shows examples of two photos produced with respect to the three cases, which were captured at 30 degrees to the left and from the front, respectively. The evaluation in Table 2 is supported by these results shown in Fig. 8. The images quality for T1 case as shown in Fig. 8(a) is low as a whole, and the head in particular is unrecognizable. Fig. 8(b) images for T2 case are cracked in the whole background and in part of the right hand since there are '0' values between depth pixels. As shown in Fig. 8(c) images for T3 case, we note that the overall shapes and the human can be more clearly recognized in comparison with the two other cases.



(a)

(b)

(c)

Fig. 8. 3D visualization captured at 30 degrees to the left and from the front, respectively, with each elemental image generated from a 30x30 lens array for three cases, (a) T1, (b) T2 and (c) T3

Table 2. Evaluation of image quality in three T1, T2 and T3 cases

Area	T1	T2	T3
Whole image	Considerable loss in both sides	Much loss in both sides	Inconsiderable loss in both sides
Background	Considerable loss in the parts with curvature or deep depth	Much loss in most parts	Inconsiderable loss except the mirror in the top middle left
Human body	Less loss than T2 in neck, armpits, and shoulders	The most loss in neck, armpits, and shoulders	The least loss in neck, armpits, and shoulders
Face	Distortion in hair and right side of face	Distortion in face and much loss around the face	The least distortion and loss
Left Arm	Not much difference	Not much difference	Not much difference
Right Arm (enlarged part)	Considerable loss around arms and distortion in palm	The most loss around arms and distortion in palm	The least loss around arms and distortion in palm

4. DISCUSSION

In this paper, we presented a method to increase the quality in integral imaging systems that provides auto-stereo images from the real world space. While previous methods reduce color images to fit into depth images to produce elemental images, we take a reverse approach of expanding the depth images to fit into the color images. The depth images are expanded by using an interpolation technique called the bi-cubic method, which results in an integral imaging system with higher image quality.

The computer technique of creating elemental images in real time for integral imaging by using depth cameras is still in the early stages, and more research is needed. For example, an effective method is needed to interpolate specific parts that are uncapturable in the left and right sides. In the future, we will extend our real-time integral imaging system to environments with high-density (HD) depth cameras.

ACKNOWLEDGEMENT

This research was supported by the MSIP (Ministry of Science, ICT and Future Planning), Korea, under the ITRC (Information Technology Research Center) support program (IITP-2016-R0992-16-1008) supervised by the IITP (Institute for Information & communications Technology Promotion).

REFERENCES

[1] G. Lippmann, "La photographie integrale," C. R. Acad. Sci., vol. 146, 1908, pp. 446-451.

[2] J. Hong, Y. Kim, H. J. Choi, J. Hahn, J. H. Park, H. Kim, S. W. Min, N. Chen, and B. Lee, "Three-dimensional display technologies of recent interest: principles, status, and issues," Applied Optics, vol. 50, 2011, pp. 87-115.

[3] X. Xiao, B. Javidi, M. Martinez-Corral, and "A. Stern, Advances in three-dimensional integral imaging: sensing, display, and applications," Applied Optics, vol. 52, 2013, pp. 546-560.

[4] S. W. Min, "Enhanced Image Mapping Algorithm for Computer-Generated Integral Imaging System," Japanese Journal of Applied Physics, vol. 45, no. 28, 2006, pp. 744-747.

[5] K. C. Kwon, C. Park, M. U. Erdenebat, J. S. Jeong, J. H. Choi, N. Kim, J. H. Park, Y. T. Lim, and K. H. Yoo, "High speed image space parallel processing for computer-generated integral imaging system," Optical Express, vol. 20, no. 2, 2012, pp. 732-740.

[6] D. H. Kim, M. U. Erdenebat, K. C. Kwon, J. S. Jeong, J. W. Lee, K. A. Kim, N. Kim, and K. H. Yoo, "Real-time 3D display system based on computer generated integral imaging technique using enhanced ISPP for hexagonal lens array," Applied Optics, vol. 52, 2013, pp. 8411-8418.

[7] G. Li, K. C. Kwon, G. H. Shin, J. S. Jeong, K. H. Yoo, and N. Kim, "Simplified integral imaging pickup method

for real objects using a depth camera," Journal of Applied Optics Korea, vol. 16, no. 4, 2012, pp. 381-385.

[8] J. S Jeong, K. C. Kwon, M. U. Erdenebat, N. Kim, and K. H. Yoo, "Development of a real-time integral imaging display system based on graphics processing unit parallel processing using a depth camera," Optical Engineering, vol. 53, no. 1, 2014, pp. 015103(1-9).

[9] Qiping Zhang, Yujiao Chen, Junhui Du, and Yiqun Liu, "Integral imaging display for natural scene based on KinectFusion," Optik, vol. 127, 2016, pp. 791-794.

[10] K. C. Kwon, J. S. Jeong, M. U. Erdenebat, Y. L. Piao, K. H. Yoo, and N. Kim, "Resolution-enhancement for an orthographic-view image display in an integral imaging microscope system," Biomedical Optical Express, vol. 6, no. 3, 2015, pp. 736-746.

[11] H. Navarro, J. C. Barreiro, G. Saavedra, M. Martinez-Corral, and B. Javidi, "High-resolution far-field integral-imaging camera by double snapshot," Optical Express, vol. 20, 2012, pp. 890-895.

[12] C. Tomasi and R. Manduchi, "Bilateral filtering for gray and color images," Sixth International Conference on Computer Vision, 1998, pp. 839-846.

[13] E. A. Karabassi, G. Papaioannou, and T. Theoharis, "A Fast Depth-Buffer-Based Voxelization Algorithm," Journal of Graphics Tools, ACM, vol. 4, no. 4, 1999, pp. 5-10.

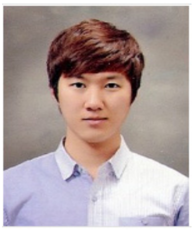
[14] X. Wang, L. He, and Q. Bu, "Performance characterization of integral imaging systems based on human vision," Applied Optics, vol. 48, 2009, pp. 183-188.

[15] Microsoft, *Kinect for Windows SDK Programming Guide*, 2016.

[16] nVidia, *CUDA C programming guide*, nVidia Corporation, 2014.

[17] nVidia, *OpenCL programming guide for the CUDA architecture*, nVidia Corporation, 2010.

Min-Ho Song
He received the B.E. in Computer Education at Chungbuk National University, Korea in 2014, and also received M.S. in Digital Informatics Convergence at Chungbuk National University in 2016. His research interests include computer graphics, and 3D digital contents.



Byung-Muk Lim
He received the B.E. in Computer Education at Chungbuk National University, Korea in 2015, and also received M.S. in Computer Science at Chungbuk National University in 2017. His research interests include computer graphics, and 3D digital contents.

Ga-A Ryu
She is a PhD student in Computer Science at Chungbuk National University. She received the B.E. in Cyber Security at Kwangju University, Korea in 2014, and also received M.S. in Digital Informatics Convergence at Chungbuk National University in 2016. His research interests include computer graphics, and 3D digital contents.

Jon-Sung Ha
He is a professor who is working for department Game and Contents at Woosuk University, Korea. He received the B.E. in computer of computer engineering at Seoul National University, Korea in 1984, and also received M.S., Ph.D. in computer science from KAIST (Korea Advanced Institute of Science and Technology), Korea in 1986 and 1996, respectively. His research interests include computational geometry, computer graphics, CAD/CAM, and 3D digital contents.

Kwan-Hee Yoo
He is a professor who is working for Department of Computer Science at Chungbuk National University, Korea. He received the B.S. in Computer Science from Chonbuk National University, Korea in 1985, and also received M.S., Ph.D. in computer science from KAIST (Korea Advanced Institute of Science and Technology), Korea in 1988 and 1995, respectively. His research interests include a u-Learning system, computer graphics, 3D character animation, dental/medical applications.