Elemental image array generation based on discrete viewpoint pickup and window interception in integral imaging

Min Guo,[1] Yujuan Si,[1,2,*] Yuanzhi Lyu,[3] Shigang Wang,[1] and Fushou Jin[4]

[1]College of Communication Engineering, Jilin University, No. 5372, Nanhu Road, Changchun 130012, China

[2]Zhuhai College of Jilin University, Zhuhai 519041, China

[3]Changchun Institute of Optics, Fine Mechanics and Physics, Photoelectric Technology Research and Development Center, No. 3888, East Nanhu Road, Changchun 130033, China

[4]College of Computer Science and Technology, Jilin University, No. 2699, Qianjin Street, Changchun 130012, China

*Corresponding author: siyj@jlu.edu.cn

Received 1 October 2014; revised 17 December 2014; accepted 18 December 2014;
posted 22 December 2014 (Doc. ID 223890); published 28 January 2015

We present a method to generate the elemental image array (EIA) in integral imaging in this paper. The discrete viewpoint image array is captured from a discrete viewpoint pickup platform and is treated by a window interception algorithm to obtain the subimage array (SIA). The EIA can be obtained from the SIA according to the transformation relationship between the EIA and SIA. We employ the EIA to display in the integral imaging system, indicating that the proposed method can truly represent the structure of the objects. © 2015 Optical Society of America

OCIS codes: (100.6890) Three-dimensional image processing; (110.6880) Three-dimensional image acquisition; (110.3010) Image reconstruction techniques.

http://dx.doi.org/10.1364/AO.54.000876

1. Introduction

As one of the promising methods in the area of three-dimensional (3D) sensing and display, integral imaging [1–5] offers a passive and relatively inexpensive way to capture 3D information and to visualize it optically or computationally. It has attracted much attention due to its various advantages, such as quasi-continuous, full parallax, full-color viewing angle, etc. [6–8]. As shown in Fig. 1, an integral imaging system mainly includes two parts: pickup and display [9–11]. In the pickup part shown in Fig. 1(a), rays of a 3D object, which passes through a lenslet array, are optically recorded as an elemental image array (EIA) on a recording medium, such as a charge-coupled device (CCD). And each elemental image (EI) in the array contains information of the object observed from a different position. In the display part shown in Fig. 1(b), the EIA is displayed on a high-resolution display, such as a liquid-crystal display. The rays emitted from the EIA are modulated by the lenslet array and propagated to reconstruct a 3D image in 3D space.

There are two kinds of methods to create content for integral imaging: computer graphics and live action. Computer graphics [12–14] deals with building computer models or representations and then displaying them by some method or theory to produce a high-quality image. This method can be used to generate an EIA of virtual things by using some computer graphical software packages, such as 3D studio max. However, live action [15–21] is preferable because it does not need the time-consuming modeling

1559-128X/15/040876-09$15.00/0

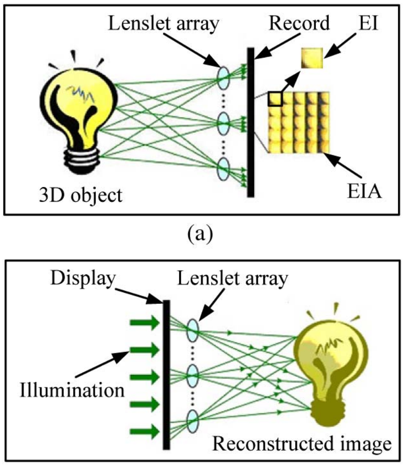

Fig. 1. Integral imaging system: (a) pickup part and (b) display part.

process. As the most direct and simplest way of live action, the lenslet-based pickup scheme [15,16] makes it possible to capture real objects directly using a lenslet array and a single recording medium. Based on this pickup scheme, Javidi *et al.* have proposed various approaches [6,7,10,11,17–20] to improve the quality of the reconstructed 3D image, but a trade-off relationship exists between the display resolution and the number of EIs for a given sensor device. Using a camera array [21] can solve the problem effectively, and each camera in the array records an EI of the object from a different location. However, the size of the camera array increases along with the increase of the display resolution, resulting in higher costs and more workload.

As we all know, a subimage array (SIA) is composed of a series of viewpoint images, and we can transform a SIA into a corresponding EIA [22]. Based on the above analyses, in this paper, we present a method to generate an EIA in integral imaging using a discrete viewpoint pickup and a window interception algorithm. And the major steps are as follows: we first capture the discrete viewpoint image array of an object, then obtain the SIA using the window interception algorithm, and finally transform the SIA into the corresponding EIA. The advantages of the proposed method are as follows:

(1) For a captured discrete viewpoint image array, the proposed method is flexible for display platforms with different lenslet arrays. To display a 3D image correctly, the number of the EIs in an EIA should be equal to that of the lenslets in a lenslet array used for display. In the proposed method, the number of the EIs is determined by the size of the interception window. For a given discrete viewpoint image array, we can easily obtain the EIAs for different lenslet arrays by employing windows of different sizes. Thus we can reduce unnecessary workforce of repeated collection and make full use of the captured discrete viewpoint image array.

(2) In the lenslet-based pickup scheme, we have to place a lenslet array close to the object owing to the low resolution of the lenslet array, so the size of the object is limited and must be less than that of the lenslet array. Nevertheless, we can generate the EIA of large-scale objects using the proposed method, because we can adopt a high-resolution camera lens and increase the distance between camera and object.

(3) Compared with the traditional camera array pickup method, the proposed method can significantly decrease costs and workload. That is because the image captured by every camera of the camera array is an EI, while the captured discrete viewpoint image in the proposed method is corresponding to a subimage (SI). If we need an EIA with a display resolution of 1024 × 768 pixels, and every EI includes 20 × 20 pixels, then we have to equip 1024 × 768 cameras when we employ the traditional camera array pickup method. Obviously, such a large-scale camera array is not only expensive but also difficult to be adjusted. However, we only need one camera and capture the object in 20 × 20 different positions if the proposed method is employed.

The structure of this paper is as follows: the discrete viewpoint pickup system is described in Section 2. Theoretical principle of the proposed method is discussed in Section 3. Section 4 covers the process of generating the EIA by the proposed method. All the experimental results are presented in Section 5, and conclusions are drawn in Section 6.

2. Discrete Viewpoint Pickup System

Figure 2 illustrates the pickup platform of the discrete viewpoint image array. And it mainly consists of real objects, a camera, a scale plate, and a photography track with a remote control. The scale plate and the two tripods are used to adjust the height of the photography track. When we operate the matching remote control, the camera can move from the right side to the left side at a uniform speed on the photography track. Meanwhile, the camera captures the discrete viewpoint images at regular intervals, and then we obtain one set of discrete viewpoint images. Afterward, we adjust the height of photography track and capture another set of discrete viewpoint images. Continue this process until all the needed discrete viewpoint images are captured. It is worth

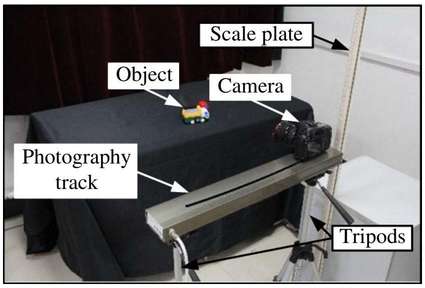

Fig. 2. Discrete viewpoint pickup.

noting that all the heights of the photography track are equally spaced. Finally, all the captured images are arranged to compose the discrete viewpoint image array.

To overcome the pseudoscopic problem, we have to rotate each EI 180 deg centro-symmetrically [16] in the lenslet-based pickup scheme. According to the relationship between the EIA and SIA [22], we can see that changes of the positions of the pixels in the EI are the same as changes of the positions of the SIs in the SIA. Therefore, we arrange the captured discrete viewpoint images according to the order of the SIs in the changed SIA, and then the orthoscopic (depth-corrected) virtual 3D image of the object can be formed behind the lenslet array during the display.

3. Theoretical Principle

Parameters related to the proposed method are from the following three parts: the lenslet-based pickup scheme, the discrete viewpoint pickup system, and the interception window. To describe their interrelation, we assume that parameters of the former two parts are known, and then we will calculate the parameters of the interception window according to the relationship between the proposed method and the lenslet-based pickup scheme. In this section, we only take one dimension into account for simplicity. The schematic diagrams of the lenslet-based pickup scheme and the proposed method are illustrated in Fig. 3, in which we use a dot to represent an imaging pixel.

As shown in Figs. 3(a) and 3(b), the three lenslets of the lenslet array correspond to the three EIs, respectively, and the pixel in the same location on each EI is corresponding to a parallel beam with a certain incident angle. Specifically speaking, there are three parallel beams which are parallel to each other in Fig. 3(a), the rays emitted from points $A_i (i = 1, 2, 3)$ denote the central axes of the parallel beams, and we use $A_i (i = 1, 2, 3)$ to describe the area between a_i and $a_{i+1} (i = 1, 2, 3)$. Then, as shown in Fig. 3(a), A_1, A_2, and A_3 are imaged in the three blue dots of the three EIs, respectively, and the three blue dots are all in the far right of the three EIs, respectively. Similarly, the areas B_1, B_2, and B_3 are imaged in the red dots shown in Fig. 3(b). The imaging pixels of A_i and $B_i (i = 1, 2, 3)$ in the ith EI are adjacent. According to the relationship between the EIA and SIA [22], the pixels at the same location in the EIs are collected to form the corresponding SI. Therefore, the imaging pixels of $A_i (i = 1, 2, 3)$ form the SI in blue dots and the imaging pixels of $B_i (i = 1, 2, 3)$ form the SI in red dots, as shown in Fig. 3(b).

We employ the proposed method to capture the same object and then generate the two SIs in Figs. 3(a) and 3(b). Figure 3(c) shows the schematic diagram of the proposed method while the camera is capturing the first discrete viewpoint image, and Fig. 3(d) illustrates the case while the camera is capturing the second discrete viewpoint image. As shown in Fig. 3(c), the area between a_i and $a_{i+1} (i = 1,$

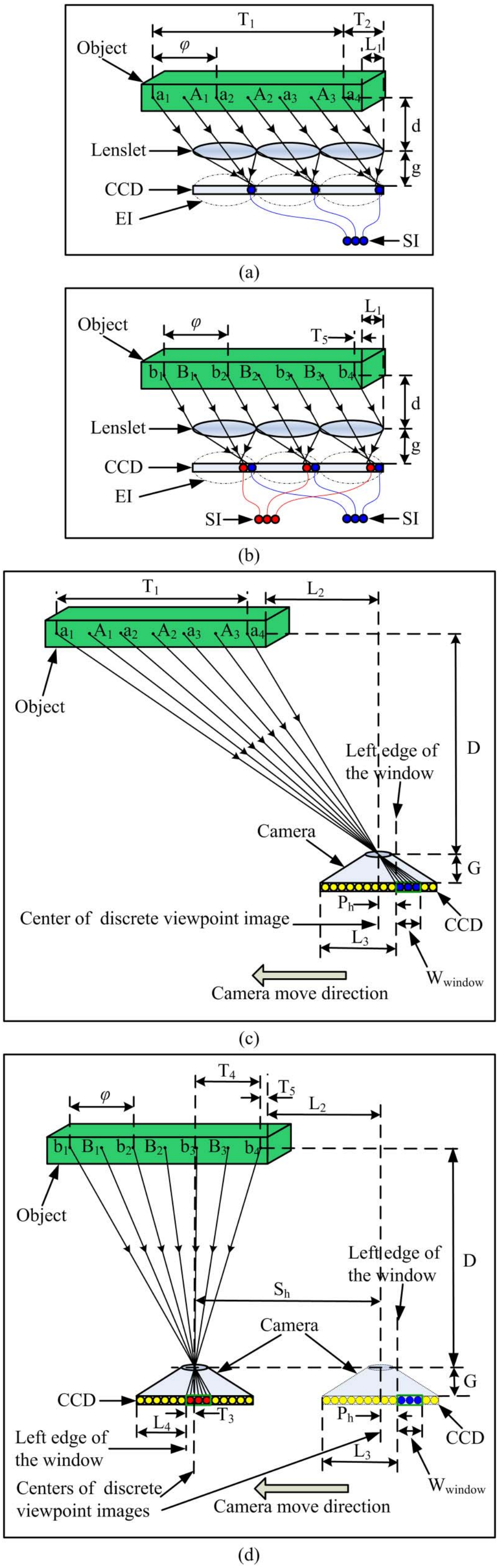

Fig. 3. Schematic diagrams of (a) lenslet-based pickup scheme for one pixel, (b) lenslet-based pickup scheme for the adjacent pixel, (c) the proposed method while the camera is capturing the first discrete viewpoint image, and (d) the proposed method while the camera is capturing the second discrete viewpoint image.

2, or 3) of the object is exactly imaged as one pixel in the CCD, so the SI (the blue dots) which corresponds to the areas $A_i (i = 1, 2, 3)$ shown in Fig. 3(a) is recorded in the first discrete viewpoint image. Similarly, as shown in Fig. 3(d), the area between b_i and $b_{i+1} (i = 1, 2, \text{ or } 3)$ of the object is exactly imaged as one pixel in the CCD, so the SI (the red dots) which corresponds to the areas $B_i (i = 1, 2, 3)$ in Fig. 3(b) is recorded in the second discrete viewpoint image. When we use a camera to capture the object from different locations, the obtained discrete viewpoint images include not only the two SIs (the blue dots and the red dots) but also other pixels (the yellow dots), as shown in Figs. 3(c) and 3(d). Afterward, we use a window to intercept every discrete viewpoint image and generate the two SIs. The EIA can be generated according to the relationship between the EIA and SIA [22].

The SIs in Figs. 3(a) and 3(b) are based on the orthographic projection while the discrete viewpoint images in Figs. 3(c) and 3(d) have perspective projection geometry. The two projections are different, but perspective projection can be approximated into orthographic projection when the focal length of the perspective projection is great and the object is far away from the projection center [23]. Therefore, we should ensure that the distance between the camera and the object is much larger than the object size, and then the SI generated by the proposed method can be used as the SI generated by the lenslet-based pickup scheme.

Next, we calculate the following parameters of the interception window: size and positions in all discrete viewpoint images. As mentioned in Section 2, all the capture positions are equidistant in horizontal direction, so the positions of the windows are also equidistant horizontally. Then, to get the positions of the windows in all the discrete viewpoint images, we only need to calculate the position of the window in the first discrete viewpoint image and the relative displacement of two horizontally adjacent windows. The detailed calculation process is as follows.

A. Size

As shown in Fig. 3(c), the camera's imaging distance is G, the distance between the camera focus and the object is D, and the width of the window can be calculated by the following equation:

$$W_{\text{window}} = T_1 \times G/D, \tag{1}$$

where W_{window} is the width of the window, T_1 is the distance between a_1 and a_4. From Fig. 3(a), we can see that T_1 is just the width of the lenslet array, and then Eq. (1) can be written as

$$W_{\text{window}} = W_{\text{lenslet}} \times G/D, \tag{2}$$

where W_{lenslet} is the width of the lenslet array, namely, 3 times of the width of one lenslet, as shown in Fig. 3(a).

To ensure that the area between a_i and $a_{i+1} (i = 1, 2, \text{ or } 3)$ of the object is exactly imaged as one pixel in the CCD, the camera's position must satisfy the following relations:

$$\frac{G}{D} = \frac{W_{\text{pix}}}{\varphi}, \tag{3}$$

where W_{pix} denotes the width of one pixel in CCD, φ is the width of the lenslet shown in Fig. 3(a).

B. Position of the Window in the First Discrete Viewpoint Image

In Fig. 3(c), the position of the shown camera is defined as the initial position of the movement of the camera in the horizontal direction, and the captured image is defined as the first discrete viewpoint image, then we can acquire the SI corresponding to the areas $A_i (i = 1, 2, 3)$ by intercepting the first discrete viewpoint image with the window. As shown in Fig. 3(c), we describe the distance between the left edge of the window and the center of the image as the horizontal position of the window in the first discrete viewpoint image. Then, from Figs. 3(a) and 3(c), we can get the following equation:

$$P_h = [(T_2 - L_1) + L_2] \times \frac{G}{D}, \tag{4}$$

where P_h is the horizontal position of the window in the first discrete viewpoint image; as shown in Fig. 3(a), T_2 is the distance between a_4 and the right edge of the lenslet array, L_1 is the distance between the right edge of the object and the right edge of the lenslet array, and then $(T_2 - L_1)$ is just the distance between a_4 and the right edge of the object; and L_2 is the distance between the right edge of the object and the center of the camera in Fig. 3(c). In addition, $(L_2 - L_1)$ denotes the distance between the right edge of the lenslet array and the center of the camera which is capturing the first discrete viewpoint image.

As shown in Fig. 3(a), T_2 can be expressed as follows:

$$T_2 = \left(\frac{\varphi}{2} - \frac{1}{2} \times \frac{\varphi}{k}\right) \times \frac{d}{g}, \tag{5}$$

where d is the distance between the lenslet array and the object shown in Fig. 3(a), g is the imaging distance of the lenslet array, and k represents the number of pixels each lenslet covers in one dimension, namely, the number of pixels each EI covers in one dimension.

To ensure that the area between a_i and $a_{i+1} (i = 1, 2, \text{ or } 3)$ of the object is exactly imaged as one pixel in the CCD, the ray emitted from a_4 must be imaged in the middle of two pixels, then we can conclude that P_h must be an exact multiple of W_{pix}, and that is:

$$P_h = N_1 \times W_{\text{pix}}, \tag{6}$$

where N_1 is an integer.

With Eqs. (4) and (6), we can see that the relative position of the lenslet array and the camera should satisfy the following requirement when the camera is capturing the first discrete viewpoint image:

$$L_2 - L_1 = \frac{D}{G} \times N_1 \times W_{\text{pix}} - T_2. \tag{7}$$

C. Relative Displacement

Since the imaging pixels of areas A_i and B_i in the ith $(i = 1, 2, 3)$ EI are adjacent in Fig. 3(b), the images captured by the camera in Figs. 3(c) and 3(d) are adjacent discrete viewpoint images. As shown in Fig. 3(d), the positions of the window in the two discrete viewpoint images are different.

Given the above analysis, we describe the relative displacement of the windows in two horizontally adjacent discrete viewpoint images with M_h. Then, as shown in Fig. 3(d), M_h can be written as

$$\begin{aligned} M_h &= L_3 - L_4 \\ &= P_h + T_3 \\ &= (L_2 - L_1 + T_2) \times \frac{G}{D} + T_4 \times \frac{G}{D} \\ &= (L_2 + T_4 - L_1 + T_2) \times \frac{G}{D}, \end{aligned} \tag{8}$$

where L_3 denotes the distance between the left edge of the discrete viewpoint image and the left edge of the inner window, as shown in Fig. 3(c), and L_4 is defined in a similar way, as shown in Fig. 3(d), T_3 is the distance between the left edge of the window and the center of the image in the second discrete viewpoint image, and T_4 represents the distance of b_4 from the center of the camera shown in Fig. 3(d).

Then, we can get the following equation from Fig. 3(d):

$$L_2 + T_4 = S_h - T_5, \tag{9}$$

where S_h is the distance between centers of the two cameras, T_5 is the distance of b_4 from the right edge of the object. Substituting Eq. (9) into Eq. (8), we get the following expression:

$$M_h = [S_h + T_2 - (L_1 + T_5)] \times \frac{G}{D}. \tag{10}$$

As shown in Fig. 3(b), $(L_1 + T_5)$ is the distance between b_4 and the right edge of the lenslet array. By a similar way to the calculation of T_2, we obtain equation as follows:

$$L_1 + T_5 = \left(\frac{\varphi}{2} - \frac{3}{2} \times \frac{\varphi}{k}\right) \times \frac{d}{g}. \tag{11}$$

Then, substituting Eqs. (5) and (11) into Eq. (10), we get the final expression for M_h:

$$M_h = \left(S_h + \frac{\varphi}{k} \times \frac{d}{g}\right) \times \frac{G}{D} = S_h \times \frac{G}{D} + \frac{\varphi}{k} \times \frac{d}{g} \times \frac{G}{D}. \tag{12}$$

For convenience, we define the expression of delta as follows:

$$\text{delta} = \frac{G}{D} \times \frac{\varphi}{k} \times \frac{d}{g}. \tag{13}$$

For the proposed method, the imaging distance G, the pickup distance D and the width of each pixel φ/k in the CCD are all known. Then we can conclude, from Eq. (13), that delta is directly proportional to the pickup distance of the lenslet array d when the imaging distance of the lenslet g is given a fixed value, so we call the intermediate variable delta the depth impact factor.

To ensure that the area between b_i and $b_{i+1}(i = 1, 2, \text{ or } 3)$ of the object is exactly imaged as one pixel in the CCD, M_h must be an exact multiple of W_{pix}, and that is:

$$M_h = N_2 \times W_{\text{pix}}, \tag{14}$$

where N_2 is an integer.

With Eqs. (12) and (14), we can know that the interval between positions of the camera, S_h, should satisfy the following requirement:

$$S_h = (N_2 \times W_{\text{pix}} - \text{delta}) \times \frac{D}{G}. \tag{15}$$

Equations (2), (4), (12), and (13) show the parameters' relationship among the following three: the interception window, the lenslet-based pickup scheme, and the discrete viewpoint pickup system. For a given EIA, which is captured in a lenslet-based pickup system, we can generate the same EIA using the proposed method by following the steps below. First, build the platform for discrete viewpoint pickup in light of the constraint Eqs. (3), (7), and (15), and collect the discrete viewpoint images; second, calculate the parameters of the interception window according to Eqs. (2), (4), and (12); and finally, intercept every discrete viewpoint image with the interception window, and generate the corresponding EIA. Obviously, the process is reversible. For a given platform for discrete viewpoint pickup, if the parameters of the interception window have been designed in light of the constraint Eqs. (3), (6), and (14), then we can build the corresponding lenslet-based pickup system according to Eqs. (2), (4), and (12). It is worth noting that the constraint Eqs. (3), (6), and (14) can be met as long as the parameters of the interception window are set at the integral multiple of the pixel size. All the above is the theoretical basis for the proposed method in this paper. In the next part, we focus on the process of generating the EIA with the proposed method.

4. EIA Generation with the Proposed Method

The process of generating the EIA by the proposed method is as follows. First, we capture the discrete viewpoint image array and recode the parameters of the pickup system. Then, we design the parameters of the interception window. At last, the SIA and the EIA can be generated. The process of the discrete viewpoint pickup is introduced in Section 2. Next, we will briefly describe the design of the parameters of the interception window and the process of generating the EIA.

From Fig. 3(d), we can see that the three parameters of the interception window should meet the following two requirements:

First, when we use the window to intercept the first discrete viewpoint image, the edges of the window should be within the image, the expressions are as follows:

$$-\frac{W_{\mathrm{CCD}}}{2} < P_h < \frac{W_{\mathrm{CCD}}}{2}, \tag{16}$$

$$0 < W_{\mathrm{window}} < W_{\mathrm{CCD}}/2 - P_h, \tag{17}$$

where W_{CCD} is the width of the CCD used in the camera.

Second, the left edge of the window in each discrete viewpoint image should be to the right of the left edge of the image, that is:

$$(k-1) \times M_h \le W_{\mathrm{CCD}}/2 + P_h. \tag{18}$$

Next, we will introduce the calculating process.

(1) Position of the window in the first discrete viewpoint image

We expand the one-dimensional case in Fig. 3(d) to the two-dimensional space. As shown in Fig. 2, when the photography track is at the minimum height and the camera is at the right-most of the photography track, the captured image is defined as the first discrete viewpoint image. In addition, we define the distance between the top edge of the window and the center of the image as the vertical position of the window in the first discrete viewpoint image. Then we can get its value span as follows in a similar way to Eq. (16):

$$-\frac{H_{\mathrm{CCD}}}{2} < P_v < \frac{H_{\mathrm{CCD}}}{2}, \tag{19}$$

where P_v is the vertical position of the window in the first discrete viewpoint image, and H_{CCD} is the height of the CCD used in the camera.

(2) Relative displacement

When the position of the window in the first discrete viewpoint image is fixed, the value span of delta can be calculated from Eqs. (12), (13), and (18):

$$0 \le \mathrm{delta} \le \frac{W_{\mathrm{CCD}}/2 + P_h}{k-1} - S_h \times \frac{G}{D}. \tag{20}$$

In a similar way to Eq. (12), we can get the following expression:

$$M_v = S_v \times G/D + \mathrm{delta}, \tag{21}$$

where M_v is the relative displacement of the window between every two vertically adjacent discrete viewpoint images. S_v is the distance between the vertically adjacent capture positions of the camera.

(3) Size

In a similar way to Eq. (17), the value span of the height of the window can be expressed as

$$0 < H_{\mathrm{window}} < H_{\mathrm{CCD}}/2 - P_v, \tag{22}$$

where H_{window} is the height of the window.

As mentioned in Section 1, the number of the EIs is determined by the size of the interception window, and we can obtain the EIAs for different lenslet arrays by employing windows of different sizes. The implementation steps are as follows: we first suppose that the lenslet array used for display includes $C \times R$ lenslets and each lenslet is quadrate, then we design the size of the window according to Eqs. (17) and (22), and let the ratio of width and height of the window be C/R, at last, we make the intercepted images' size be $C \times R$ pixels by magnifying or shrinking them. Then the EIA can be generated according to the relationship between the EIA and SIA [22].

5. Experimental Results

For the pickup platform shown in Fig. 2, two toy trucks with the same size of 105 mm(length) × 50 mm(width) × 60 mm(height) are used as the 3D objects, and the white truck, which is closer to the camera, blocks the red one. The camera model is Canon 7D, and the CCD size is 22.3 mm(width)× 14.9 mm(height). The pickup distance is set to be 600 mm, and the imaging distance is 17 mm. We capture the objects from 24 × 24 different positions, and distances between the adjacent capture positions of the camera are 12.5 mm (horizontal) and 6.7 mm (vertical), respectively. The resolution of the captured discrete viewpoint image is 2592 (horizontal) by 1728 (vertical) pixels. Figure 4 shows the captured discrete viewpoint image array, from which nine images are magnified and shown in Fig. 5; the nine magnified images are from rows 1, 12, and 24, columns 1, 12, and 24 of the array, respectively. Obviously, the discrete viewpoint image array presents the changing process of the red truck from visible to invisible,

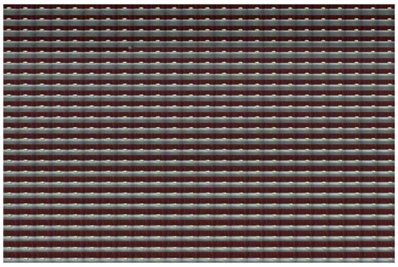

Fig. 4. Discrete viewpoint image array.

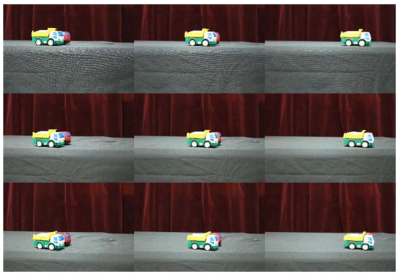

Fig. 5. Magnified discrete viewpoint images.

which is caused by the various positions of the camera.

Then, we process the discrete viewpoint image array using the designed window. For example, as shown in Fig. 6, the white frames denote the windows, with which we intercept the nine discrete viewpoint images shown in Fig. 5.

In the experiment, we introduce the “pixel” for the unit of the parameters of the interception window. And according to Eqs. (6), (16), and (19), the window in the first discrete viewpoint image is positioned as: $P_h = 0$ pixel and $P_v = 0$ pixel. Then the value scope for delta ranges from 0 pixel to 15 pixels, which can be calculated from Eq. (20).

We first choose delta $= 3$ pixels at random within the value span of delta mentioned above. Then from Eqs. (12), (14), and (21), the relative displacements of the window are 44 pixels (horizontal) and 25 pixels (vertical), respectively. From Eq. (17), the value scope for the width of the window ranges from 0 pixel to 1296 pixels, and the value scope for the height of the window ranges from 0 pixel to 864 pixels, which can be calculated from Eq. (22). For the display platform shown in Fig. 7, the lenslet array has 50 × 33 lenslets with the focal length of 5 mm, each lenslet element is square shaped and has a uniform base size of 1.08 mm × 1.08 mm. In addition, the resolution of the projector is 1280 (horizontal) by 800 (vertical) pixels. Then according to Eqs. (2) and (3), we set the size of the window to be 850 (horizontal) pixels and 561 (vertical) pixels, respectively, and then use it to intercept the discrete viewpoint images one by one. Afterward, we shrink every intercepted image by averaging all the pixels within every 17 × 17 pixels scope, and using the average as the corresponding pixel of the generated SI. The generated SIA when delta $= 3$ pixels is shown in Fig. 8, and the corresponding EIA is illustrated in Fig. 9. As we can see from Fig. 8, the SIA with 24 × 24 SIs has continuous parallaxes both horizontally and vertically. The EIA in Fig. 9 includes 50 × 33 EIs, and each EI includes 24 × 24 pixels.

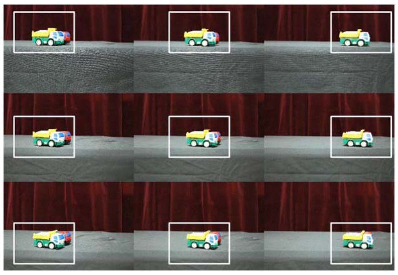

Fig. 6. Schematic diagram of the interception.

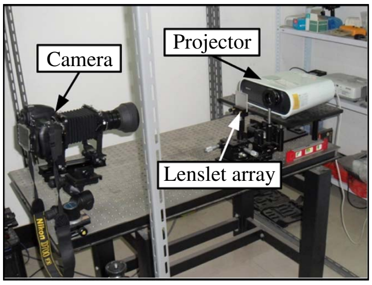

Fig. 7. Display platform of the integral imaging.

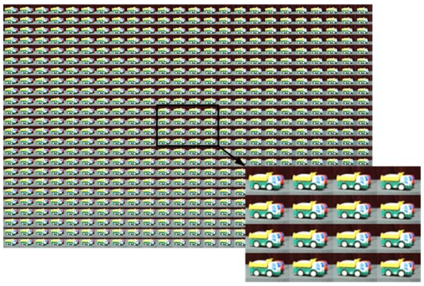

Fig. 8. SIA when delta $= 3$ pixels.

Now, parameters of the window and the discrete viewpoint pickup system are all know. Then we can calculate the parameters of the corresponding lenslet-based pickup system according to their correlation. The size of the lenslet array is 258.1 mm(width) × 170.3 mm(height) and the size of each lenslet is 5.16 mm × 5.16 mm, according to Eqs. (2) and (3). And from Eq. (13), the ratio of d and g is 4.24. Since the lenslet array plane is perpendicular to the optical axis of the camera, the horizontal and vertical distances between the bottom right corner of the lenslet array and the center of the camera shown in Fig. 3(c) are both −10.5 mm, which can be calculated from Eq. (4).

To verity the feasibility of the proposed method, the obtained EIA is displayed on the display platform shown in Fig. 7. As shown in Fig. 10, the two toy trucks can be observed from different viewing positions with the same height. Obviously, we can watch

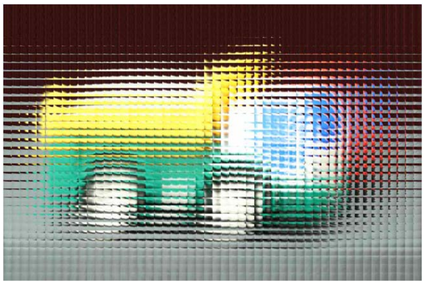

Fig. 9. EIA when delta $= 3$ pixels.

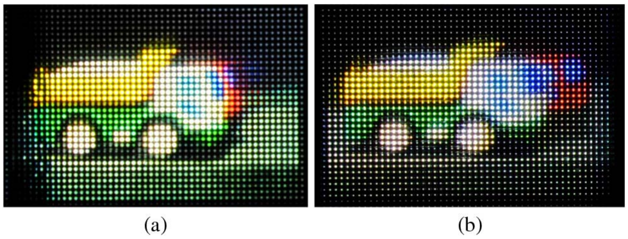

Fig. 10. Display status when delta = 3 pixels: (a) image observed from the left side, and (b) image observed from the right side.

the larger part of the red truck's head from the right side, and we can only watch the edge from the left side.

Due to the limitation of the display device we used, the resolution of images in Fig. 10 is low. However, with the development of a higher resolution display device and the improvement of the fabrication technology for lenslet arrays, high-resolution display will certainly be realized in integral imaging.

Based on the analysis above, we can conclude that there are several groups of corresponding relationships between the proposed method and the lenslet-based pickup scheme. From Eq. (2) we know that increasing the size of the window is equivalent to choosing a larger lenslet array in the lenslet-based pickup scheme. Equation (4) implies that the change of the window's position in the first discrete viewpoint image is equivalent to the translational movement of the lenslet array in its plane. Equations (12) and (13) indicate that the change of the relative displacement of the window is equivalent to the change of the pickup distance in the lenslet-based pickup scheme. Therefore, the parameters of the window can be designed according to the actual requirements.

For the sake of contrastive analysis, delta = 7 pixels is chosen, and the corresponding research is carried out. When $P_h = 0$ pixel, $P_v = 0$ pixel, and delta = 7 pixels, the width and height of the window are 950 pixels (horizontal) and 627 pixels (vertical), respectively. The generated SIA and EIA are as shown in Figs. 11 and 12, respectively. And the observed images from different positions of the EIA are shown in Fig. 13. Compare Figs. 8 and 10 with Figs. 11 and 13, we can see that the display depth when delta = 7 pixels is greater than the display depth when delta = 3 pixels.

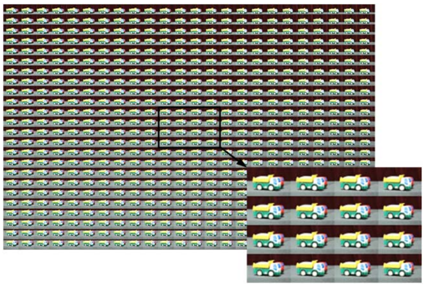

Fig. 11. SIA when delta = 7 pixels.

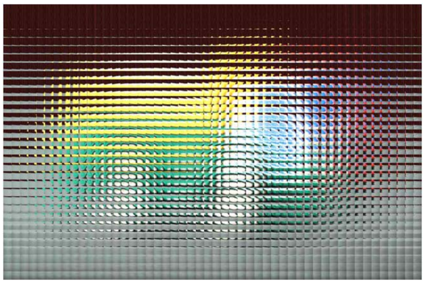

Fig. 12. EIA when delta = 7 pixels.

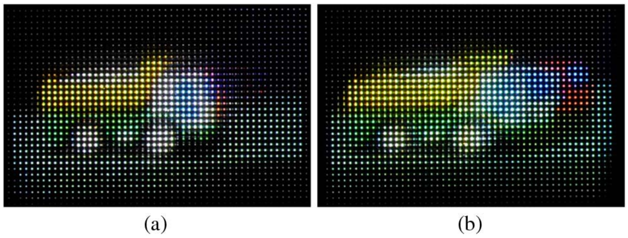

Fig. 13. Display status when delta = 7 pixels: (a) image observed from the left side, and (b) image observed from the right side.

As shown in Fig. 14(a), when the thickness of the object is 0 (in an ideal world), the captured images based on the perspective projection are the same as those of the orthographic projection. But in fact, as shown in Fig. 14(b), it is not possible that the thickness of one object is 0. For the perspective projection, the width of the object in the captured image

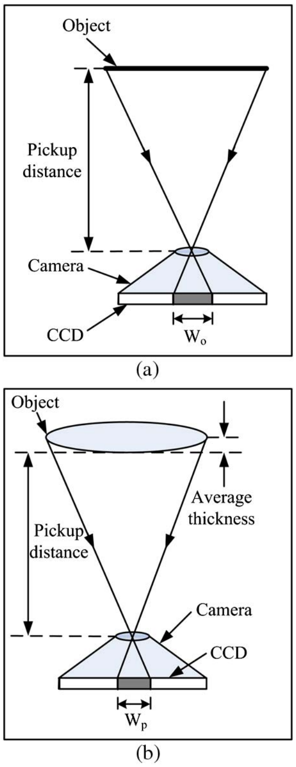

Fig. 14. Perspective projection: (a) when the thickness is 0, and (b) when the thickness is not 0.

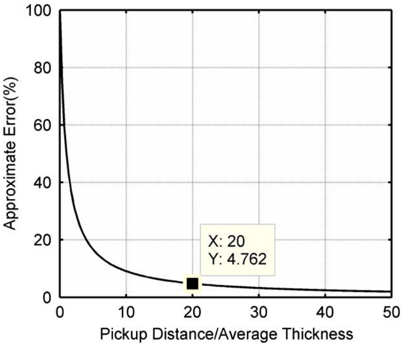

Fig. 15. Approximate error versus the ratio of pickup distance to average thickness.

is smaller than that of the orthographic projection, so there is an approximate error when we approximate the perspective projection into the orthographic projection. We define the approximate error as follows:

$$E = \frac{W_o - W_p}{W_o}, \qquad (23)$$

where E is the approximate error, W_o and W_p are the widths of the object in the captured images based on the orthographic projection and perspective projection, respectively.

Figure 15 shows the relationship of the approximate error versus the ratio of pickup distance to average thickness of the object, from which we can see that the effect of the approximate error diminishes along with the increase of the pickup distance for a given object. And the black square in Fig. 15 represents the data of our experiment, the pickup distance is 600 mm, the average thickness of the object is 30 mm, and the approximate error is 4.762%.

6. Conclusion

This paper presents a method of generating the EIA in integral imaging, which combines the discrete viewpoint pickup with the window interception algorithm. The experimental results indicate that the generated EIA can truly represent the parallax of the objects. By changing the size of the window, we can further obtain the EIAs for different display platforms without repeated pickup. Compared with the lenslet-based pickup scheme, the proposed method can generate the EIA of a large-scale object. In addition, it can significantly decrease costs and the workload by comparison to the traditional camera array pickup method. Since we approximate perspective projection into orthographic projection, there is an approximate error in the proposed method, but the approximate error can be decreased with the increase of the pickup distance.

This work is supported by the National High Technology Development Program (863 Program) under grant 2012AA011505.

References

1. M. G. Lippmann, “La photographic integrale,” C. R. Acad. Sci. **146**, 446–451 (1908).
2. H. E. Ives, “Optical properties of a Lippmann lenticulated sheet,” J. Opt. Soc. Am. **21**, 171–176 (1931).
3. C. B. Burckhardt, “Optimum parameters and resolution limitation of integral photography,” J. Opt. Soc. Am. **58**, 71–76 (1968).
4. N. Davies, M. McCormick, and M. Brewin, “Design and analysis of an image transfer system using microlenslet arrays,” Opt. Eng. **33**, 3624–3633 (1994).
5. B. Javidi and F. Okano, *Three-Dimensional Television, Video, and Display Technology* (Springer-Verlag, 2002).
6. A. Stern and B. Javidi, “Three-dimensional image sensing, visualization, and processing using integral imaging,” Proc. IEEE **94**, 591–607 (2006).
7. Y. S. Hwang, S.-H. Hong, and B. Javidi, “Free view 3-D visualization of occluded objects by using computational synthetic aperture integral imaging,” J. Disp. Technol. **3**, 64–70 (2007).
8. R. Martinez-Cuenca, G. Saavedra, M. Martinez-Corral, and B. Javidi, “Progress in 3-D multiperspective display by integral imaging,” Proc. IEEE **97**, 1067–1077 (2009).
9. J.-H. Park, K. Hong, and B. Lee, “Recent progress in three-dimensional information processing based on integral imaging,” Appl. Opt. **48**, H77–H94 (2009).
10. M. Cho, M. Daneshpanah, I. Moon, and B. Javidi, “Three-dimensional optical sensing and visualization using integral imaging,” Proc. IEEE **99**, 556–575 (2011).
11. J.-Y. Son, W.-H. Son, S.-K. Kim, K.-H. Lee, and B. Javidi, “Three-dimensional imaging for creating real-world-like environments,” Proc. IEEE **101**, 190–205 (2013).
12. T. Naemura, T. Yoshida, and H. Harashima, “3-D computer graphics based on the integral photography,” Opt. Express **8**, 255–262 (2001).
13. A. Rockwood and J. McAndless, “Through the looking glass: the synthesis of computer graphics and computer vision,” IEEE MultiMedia **6**, 8–11 (2002).
14. H. Liao, T. Dohi, and K. Nomura, “Autostereoscopic 3D display with long visualization depth using referential viewing area-based integral photography,” IEEE Trans. Visual. Comput. Graphics **17**, 1690–1701 (2011).
15. F. Okano, H. Hoshino, J. Arai, and I. Yuyama, “Realtime pickup method for a three-dimensional image based on integral photography,” Appl. Opt. **36**, 1598–1603 (1997).
16. J. Arai, F. Okano, H. Hoshino, and I. Yuyama, “Gradient-index lens-array method based on realtime integral photography for three-dimensional images,” Appl. Opt. **37**, 2034–2045 (1998).
17. J.-S. Jang and B. Javidi, “Improved viewing resolution of three-dimensional integral imaging by use of nonstationary micro-optics,” Opt. Lett. **27**, 324–326 (2002).
18. S.-H. Hong and B. Javidi, “Improved resolution 3D object reconstruction using computational integral imaging with time multiplexing,” Opt. Express **12**, 4579–4588 (2004).
19. J.-S. Jang and B. Javidi, “Three-dimensional synthetic aperture integral imaging,” Opt. Lett. **27**, 1144–1146 (2002).
20. A. Stern and B. Javidi, “3-D computational synthetic aperture integral imaging (COMPSAII),” Opt. Express **11**, 2446–2451 (2003).
21. M. Levoy, “Light fields and computational imaging,” IEEE Comput. Mag. **39**(8), 46–55 (2006).
22. J.-H. Park, J. Kim, and B. Lee, “Three-dimensional optical correlator using a sub-image array,” Opt. Express **13**, 5116–5126 (2005).
23. Y. Jia, *Machine Vision* (Science, 2000).