Region segmentation and parallel processing for creating large-scale CGHs in polygon source method

Kyoji Matsushima[a] and Sumio Nakahara[b]

[a]Department of Electrical and Electronic Engineering; [b]Department of Mechanical Engineering
Kansai University, Yamate-cho 3-3-35, Suita, Osaka 564-8680, Japan

ABSTRACT

A novel technique is presented for calculating large-scale CGHs by using polygon-based method. In the technique, wave fields are regionally segmented and propagated by using the shifted Fresnel method. As a result, large CGHs can be calculated even in the case that the whole frame buffer can not be stored in main memory. The produced full-parallax CGH has the size of four billion pixels and reconstructs the fine image of a smooth surface object accompanied with a strong depth sensation.

Keywords: Computer-generated hologram, Polygon, Surface model, Segmentation

1. INTRODUCTION

Computer-generated holograms (CGH), sometimes called digitally synthetic holograms, make it possible to create the hologram of virtual objects that exist only as the numerical model. In the process of making CGHs, the object fields emitted from the virtual object is, first, calculated from its numerical model. Secondly, the object field represented with distribution of complex-valued amplitude is encoded into either distribution of amplitude or phase. Finally, the spatial distribution obtained is printed by using some fringe-drawing device or displayed electronically on the screen by using some spatial light modulator such as LCD.

For the last decade, object fields in the first step is obtained by superimposing spherical waves on the hologram, which are emitted from point sources of light spreading over the surface of the virtual object. This method is very simple, but there is a big problem in its computation time. The computation time of this method is proportional to the product of the number of point sources and sampling points on the hologram. The product can be a tremendous number when one calculates a high-definition and large-scaled full-parallax CGH based on a surface model, and thus computation time is very long.

Recently one of the authors has proposed another technique[1] based on polygon sources of light instead of point sources. In this technique a virtual object is composed of some planar and arbitrarily tilted polygons. Object fields emitted from the polygons are individually calculated in a plane parallel to the hologram by using rotational transformation[2,3] and then propagated into the hologram. These computational processes are executed polygon-by-polygon, and then the individual polygon fields are totaled on the hologram or intermediate planes in a layered scene structure. Calculation by this technique is essentially faster than that using the point-based method, because of using FFT. In addition, this technique treats objects as a set of polygons and is closer to ordinary CG techniques than the point-based one. Thus, some researchers have proposed similar polygon-based methods[4,5] to improve our original idea mainly in computation time.

However, in the polygon-based method, region segmentation and parallel processing are difficult, because the algorithm requires calculation of numerical propagation/diffraction. Thus, producing large-scale CGHs are not easy. In this article it is intended to describe techniques to create high-definition and large-scaled CGHs with size of more than 10^9 pixels by using segmentation of frame buffers and parallel processing. The shifted Fresnel method[6] recently proposed is used for the numerically translational propagation in the segmented frame buffers.

Further author information: (Send correspondence to K. Matsushima)
K. Matsushima: E-mail: matsu@kansai-u.ac.jp
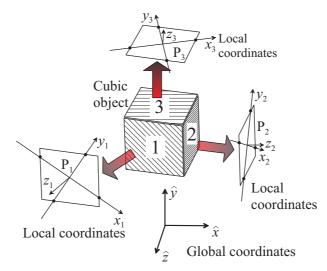S. Nakahara: E-mail: nakahara@ipcku.kansai-u.ac.jp

Figure 1. The object model and definitions of the global and local coordinates.

2. CALCULATION OF POLYGON FIELD

An example to calculate the object fields from a cubic object is shown in Fig. 1. The object model is given as a set of vertices and can be resolved into three polygons $P_{1,\cdots,3}$. Note that techniques of exact hidden surface removal[7-9] are out of the scope of this article. Only back-face culling is made by relation between the surface and hologram, and thus far side polygons are ignored.

Each polygon P_n has their own local coordinates (x_n, y_n, z_n). Any local coordinates can be adopted arbitrarily as long as a rotation matrix is well-defined. The polygon and the local coordinates is rotated by the rotation matrix so as to be parallel to the hologram or the global coordinates $(\hat{x}, \hat{y}, \hat{z})$
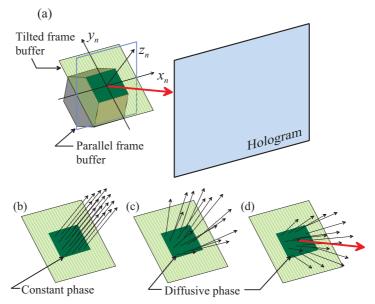

Figure 2. Configuration of surface functions.

2.1 Definition of Surface Function

A polygon source of light is represented by two dimensional distribution of complex-valued amplitude in the $(x_n, y_n, 0)$ plane. This distribution is called the surface function or material function. This function plays an important role to give appearance of the polygon in optical reconstruction.

The surface functions are generally given as the following form:

$$s_n(x_n, y_n) = a_n(x_n, y_n) \exp[i\phi_n(x_n, y_n)], \tag{1}$$

where the real-valued amplitude $a_n(x_n, y_n)$ gives the shape of the polygon as follows:

$$a_n(x_n, y_n) = \left\{ \begin{array}{lll} A_n & \cdots & \text{inside polygon} \\ 0 & \cdots & \text{outside polygon} \end{array} \right. \tag{2}$$

The amplitude A_n is different value for each polygon in order to shade the object appropriately as intended by the designer.[1] The texture of polygon can be also given as modulation of amplitude. In this case, A_n is not constant but a function of x_n and y_n. In this report, complex-valued buffers that store the surface functions in numerical calculation is called the tilted frame buffer as shown in Fig. 2 (a).

If the surface function is a simple polygon-shaped distribution of real-valued amplitude, the surface function behaves as if it was an aperture that shapes the polygon. As a result, light from the polygon does not spread over the hologram, as shown in Fig. 2 (b). Therefore, phase $\phi_n(x_n, y_n)$ of the surface function is necessary for giving diffusiveness to the wave field of the polygon source. Random functions are candidates for the diffusive phase $\phi_n(x_n, y_n)$. But full random functions are not appropriate, because the random phases are discontinuous and have a large spatial frequency that causes speckles in the reconstruction. Therefore, quasi-random phase distribution proposed as a digital diffuser[10] in Fourier holograms is used for the phase distribution.

The wave field of polygon is diffused by using the diffuser phase as shown in Fig. 2 (c). However, if the spatial carrier frequency of the surface function is zero, the wave field mainly propagates in the normal direction to the polygon, and thus light from polygon can not reach the hologram effectively. The center of spectrum of the surface function must be shifted so that light travels along optical axis and almost perpendicularly intersects the plane parallel to the hologram. This shift operation is a part of rotational transformation described in Sec. 2.2.

Examples of local coordinates and surface functions stored in the tilted frame buffer are shown in Fig. 3. These are of the case of the cubic object given in Fig. 1. The surface functions defined for polygons $P_1, \cdots, P_2$ in (a) are shown in (b), respectively. The amplitude of the surface function $s_n(x_n, y_n)$ is zero outside the polygon, while the amplitude of the inside is a constant value that is different for each polygon and determined so as to give appropriate shade to the object in optical reconstruction.[1]

2.2 Rotational Transformation

The polygons forming the object surfaces are usually not parallel to the hologram, and thus wave fields in the hologram plane can not be obtained by using ordinary formulas for diffraction or propagation of wave fields, such as Fresnel diffraction formula. Therefore, the surface functions must be transformed into wave fields given in a plane parallel to the hologram. This section briefly summarizes the procedure.

First, surface functions are Fourier transformed as follows:

$$S(u_n, v_n) = \mathbf{F}\{s_n(x_n, y_n)\}, \tag{3}$$

where $\mathbf{F}$ and $\mathbf{F}^{-1}$ stands for Fourier and inverse Fourier transform, respectively. The Fourier spectrum $S(u_n, v_n)$ of polygon n is numerically obtained by using fast Fourier transform (FFT). Here, u_n and v_n are Fourier frequencies.

Second, transformation matrix for changing the local coordinates into different coordinates parallel to the hologram is defined as,

$$\mathbf{T}_n^{-1} = \left[\begin{array}{ccc} a_1 & a_2 & a_3 \\ a_4 & a_5 & a_6 \\ a_7 & a_8 & a_9 \end{array} \right]. \tag{4}$$

Amplitude image   Phase image   Amplitude image

$S_1(x_1, y_1)$

$S_2(x_2, y_2)$

$S_3(x_3, y_3)$

(a) Local   (b) Tilted frame buffer   (c) Parallel frame
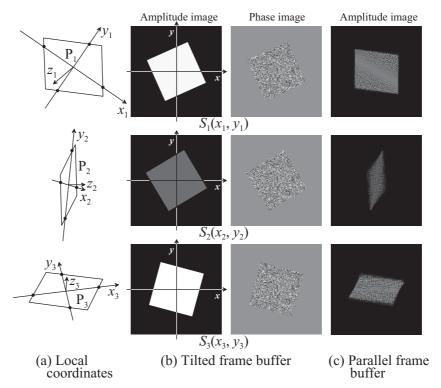coordinates   buffer

Figure 3. Examples of the surface functions.

Note that the matrix elements a_i depends on the polygon, but the subscript n is omitted for simplifying notation. Suppose that a local parallel coordinates is defined as $\hat{\mathbf{r}}_n = (\hat{x}_n, \hat{y}_n, \hat{z}_n)$. This coordinates share the origin with the local coordinates $\mathbf{r}_n = (x_n, y_n, z_n)$ but the $(\hat{x}_n, \hat{y}_n, 0)$ plane is parallel to the hologram. The coordinates rotation from a local parallel coordinates $\hat{\mathbf{r}}_n = (\hat{x}_n, \hat{y}_n, \hat{z}_n)$ to the local coordinates $\mathbf{r}_n = (x_n, y_n, z_n)$ is given as

$$\mathbf{r}_n = \mathbf{T}_n^{-1} \hat{\mathbf{r}}_n \tag{5}$$

There is somewhat of arbitrariness in the means for determining the matrix $\mathbf{T}^{-1}$. We used Rodrigues' rotation formula for obtaining the matrix in calculating the practical hologram of the following section.

Fourier frequencies $(\hat{u}, \hat{v})$ in the parallel coordinates is mapped into Fourier frequencies (u_n, v_n) in the local coordinates by

$$\begin{aligned} u_n &= \alpha(\hat{u}_n, \hat{v}_n) = a_1 \hat{u}_n + a_2 \hat{v}_n + a_3 \hat{w}(\hat{u}_n, \hat{v}_n), \\ v_n &= \beta(\hat{u}_n, \hat{v}_n) = a_4 \hat{u}_n + a_5 \hat{v}_n + a_6 \hat{w}(\hat{u}_n, \hat{v}_n), \end{aligned} \tag{6}$$
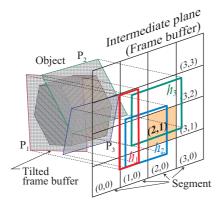
where $\hat{w}(\hat{u}_n, \hat{v}_n) = (\lambda^{-2} - \hat{u}_n^2 - \hat{v}_n^2)^{1/2}$ is the Fourier frequency in $\hat{z}_n$ axis. This mapping moves the origin of spectrum to a non-zero frequency. However, FFT algorithm can only work well around the zero frequency, therefore spectrum $\hat{S}(\hat{u}_n, \hat{v}_n)$ in the global coordinates should be obtained by remapping spectrum $S(u_n, v_n)$ onto the Fourier space $(\hat{u}_n, \hat{v}_n)$ as follows:

$$\hat{S}(\hat{u}_n, \hat{v}_n) \cong S(\alpha(\hat{u}_n, \hat{v}_n) - u_n^{(0)}, \beta(\hat{u}_n, \hat{v}_n) - v_n^{(0)}). \tag{7}$$

where $u_n^{(0)}$ and $v_n^{(0)}$ give shift values of the carrier frequencies, i.e. the center of spectrum. This remapping does not only rotate the coordinates but also change the direction in which polygon light travels. In most of polygons, the most appropriate shift values are

$$\begin{aligned} u_n^{(0)} &= \alpha(0,0) = a_3/\lambda, \tag{8} \\ v_n^{(0)} &= \beta(0,0) = a_6/\lambda, \tag{9} \end{aligned}$$
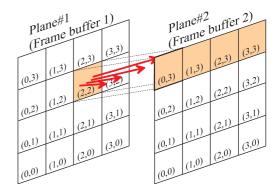
Figure 4. Segmentation in the object field.

Figure 5. Numerical propagation of segmented wave fields.

because the carrier frequency is changed to zero in parallel coordinates and thus the polygon fields forced to travel perpendicular to the hologram. However, there are a few exceptions. In some polygons of which normal vector are almost parallel to the hologram, fine adjustments should be made to the shift values.

Note that the nearly equal sign in eq.(7) states that an interpolation is required, because coordinates rotation does not only move the origin but also distorts the sampling grid.[3]

Finally, the wave field from the polygon is obtained in the parallel coordinates by using inverse Fourier transform as follows:

$$h_n(x_n, y_n) = \mathbf{F}^{-1}\{\hat{S}(\hat{u}_n, \hat{v}_n)\} \tag{10}$$

In numerical implement, the spectrum $\hat{S}(\hat{u}_n, \hat{v}_n)$ and wave field $h_n(x_n, y_n)$ are accumulated in a complex-valued buffer called parallel frame buffer as shown in Fig. 2 (a). Examples of $h_n(x_n, y_n)$ are shown as amplitude images in Fig. 3 (c).

3. FRAME BUFFER SEGMENTATION

The polygon-based techniques for synthesizing object field is the method based on wave optics or Fourier optics. In this branch of optics, wave fields are generally represented by distribution of complex amplitudes and treated as a sampled two-dimensional array of two floating-point values in computers. Therefore, when the array size is N^2 and single-precision floating point are used for calculation, the memory size needed for the array is $8N^2$ bytes. We created a CGH with pixels of $2^{16} \times 2^{16}$ in this investigation. Therefore, the frame buffer storing the whole CGH needs the memory size of 32 G Bytes. This estimation is only for a single frame buffer. Another memory spaces required for running the program, such as OS area, code area, temporary frame buffer, work space and I/O buffer etc., are not included at all. As a result, regional segmentation of CGH is essential to producing the large-scale CGH.

3.1 Segmentation in Calculating Object Fields

Both sizes of the tilted frame buffer and the parallel frame buffer in Fig. 2 depends on the polygon size. These are relatively small if the object have curved surfaces and is constructed of many polygons. The wave field of each polygon, stored in the parallel frame buffer, is numerically propagated into a intermediate plane. Then all polygon fields are gathered and summed up in a large frame buffer. This large frame buffer storing the whole object field in intermediate plane must be segmented because its size is same as the hologram one. Since the whole of the large frame buffer can not be in memory, the summing-up processes are made for each segment of the large buffer. When the summing-up process is finished for a segment, the segment is saved as a file on the disk, and then next segment is processed in memory. This process is parallelized for each segment in threads level. Therefore, the number of segments, simultaneously processed in memory, is same as the number of CPU cores. Furthermore, as shown in Fig. 4, polygons of which light reach to the currently processed segment are selected before calculating the polygon field. The selection is carried out by obtaining diffraction area of the polygon to the intermediate plane. For example, the wave fields of the polygon P_2 and P_3 are calculated and summed up but polygon P_1 is ignored when processing the segment (2, 1) of Fig. 4.

Table 1. Parameters used for calculating the object field and fabricating the hologram.

Total number of pixels	4.3×10^9 $(65,536 \times 65,536)$
Pixel pitches	1 μm $\times 1$ μm
Dimension of hologram	65.5×65.5 mm^2
Number of segments	8×4
Hologram coding	Binary amplitude
Reconstruction wavelength	632.8 nm
Total number of polygons of Venus	1396
Number of visible polygons of Venus	718
Dimension of Venus (W $\times$ H $\times$ D)	$26.7 \times 57.8 \times 21.8$ mm^3
Dimension of wallpaper (W $\times$ H)	65.5×65.5 mm^2
Distance of Venus from hologram	150 mm
Distance of wallpaper from hologram	300 mm

3.2 Segmentation in Translational Propagation

The object field obtained in intermediate plane is numerically propagated into a next plane or the hologram. This numerical propagation must be regionally segmented again. Shifted Fresnel method[6] for numerical propagation features the capability of calculating the target wave field from the source wave field whose center position is not same as the target. Therefore, the method allows us to calculate the wave field, for example, in the segment (0, 3) of plane #2 from the segment (2, 2) of plane #1, as shown in Fig. 5. This calculation can be also parallelized in threads level. As a result, memory is required only for one segment in plane #1 and same number of segments as CPU cores in plane #2.

4. CREATING THE HOLOGRAM "VENUS"

Fig. 6 shows the geometry for producing a large-scale CGH named "Venus" in this investigation. The Venus object is placed at 15 cm behind the hologram. In addition, a wallpaper object is placed at 15 cm behind the Venus. The Venus is a surface-modeled object constructed of 1396 polygons, while the wallpaper object is a simple planar image. Some parameters of the Venus hologram is summarized in Table 1.

4.1 Masking Wallpaper Field

If the field of the wallpaper was simply added to that of the Venus object, the Venus would be optically reconstructed as a phantom image, because simple superposition of these fields involves the Venus object transmitting wallpaper light. To prevent transmission of light, propagation process was divided into two steps. First, the wallpaper field is propagated into the object plane, as shown in Fig. 6. Then, the field is masked with the silhouette of the Venus.[7,8] Second, the field is propagated into the hologram after adding the Venus field.

4.2 Computation Time

The procedure described above is implemented by using Intel C++ Compiler 10 and Visual Studio 2008. Intel Math Kernel Library is also used for FFT. Numerical calculation was executed in a PC with 4 CPU of AMD Opteron 852 (2.6GHz) and 32 G Bytes RAM. Total computation time was 45 hours. Itemized computation times are shown in Fig. 7. Longest time was consumed in translational propagation. The amount is 37 hours and account for 82% of total computation time.
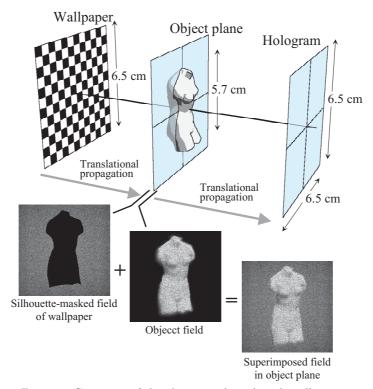
Figure 6. Geometry of the objects and masking by silhouette.

4.3 Optical Reconstruction of the Venus

The CGH "Venus" was fabricated by DWL 66 laser lithography system made by Heidelberg Instruments, GmbH. Optical reconstruction of the fabricated hologram is shown in Fig. 8. The pictures (a)–(d) were captured in various points of view. The frame of printed pictures in this figure is just same as the hologram frame. The shifts of the Venus and the wallpaper object in the picture frame are caused by parallax in the reconstructed field. This motion parallax gives a strong sensation of depth to observer's perception.

5. CONCLUSION

We numerically synthesized a large-scale CGH with 4 G pixels by using polygon-based method, which numerically generates object fields from polygon sources of light instead of point sources. In the technique, computation time is generally shorter as compared with the point-based methods. However, region segmentation and parallel processing was more difficult because of requirement of propagation of wave fields. We resolved this problem by
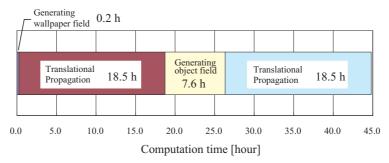

Figure 7. Computation time for producing the Venus.

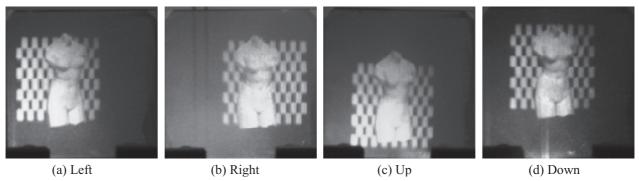(a) Left　　　　　　　(b) Right　　　　　　　(c) Up　　　　　　　(d) Down

Figure 8. Pictures of optical reconstruction of the fabricated hologram from various viewpoints.

using shifted Fresnel method. The created full-parallax CGH named Venus has good appearance and causes a strong depth sensation owing to a silhouette-masking technique.

ACKNOWLEDGMENTS

REFERENCES

[1] Matsushima, K., "Computer-generated holograms for three-dimensional surface objects with shade and texture," *Appl. Opt.* **44**, 4607–4614 (2005).

[2] Matsushima, K., Schimmel, H., and Wyrowski, F., "Fast calculation method for optical diffraction on tilted planes by use of the angular spectrum of plane waves," *J. Opt. Soc. Am.* **A20**, 1755–1762 (2003).

[3] Matsushima, K., "Formulation of the rotational transformation of wave fields and their application to digital holograph," *Appl. Opt.* **47**, D110–D116 (2008).

[4] Ahrenberg, L., Benzie, P., Magnor, M., and Watson, J., "Computer generated holograms from three dimensional meshes using an analytic light transport model," *Appl. Opt.* **47**, 1567–1574 (2008).

[5] Kim, H., Hahn, J., and Lee, B., "Mathematical modeling of triangle-mesh-modeled three-dimensional surface objects for digital holography," *Appl. Opt.* **47**, D117–D127 (2008).

[6] Muffoletto, R. P., Tyler, J. M., and Tohline, J. E., "Shifted fresnel diffraction for computational holography," *Opt. Express* **15**, 5631–5640 (2007).

[7] Matsushima, K. and Kondoh, A., "A wave optical algorithm for hidden-surface removal in digitally synthetic full-parallax holograms for three-dimensional objects," *SPIE Proc. Practical Holography XVIII* **#5290**, 90 (2004).

[8] Kondoh, A. and Matsushima, K., "Hidden surface removal in full-parallax CGHs by silhouette approximation," *Systems and Computers in Japan* **38**, 53–61 (2004).

[9] Matsushima, K., "Exact hidden-surface removal in digitally synthetic full-parallax holograms," *SPIE Proc. Practical Holography XIX and Holographic Materials XI* **#5742**, 25–32 (2005).

[10] Bräuer, R., Wyrowski, F., and Bryngdahl, O., "Diffusers in digital holography," *J. Opt. Soc. Am.* **A8**, 572–578 (1991).