

# Computer generated hologram from Multiview-plus-Depth data considering specular reflections

Antonin Gilles, Patrick Gioia, Rémi Cozot, Luce Morin

▶ **To cite this version:**

Antonin Gilles, Patrick Gioia, Rémi Cozot, Luce Morin. Computer generated hologram from Multiview-plus-Depth data considering specular reflections. IEEE International Conference on Multimedia Expo Workshops (ICMEW 2016), Jul 2016, Seattle, WA, United States. pp.1-6, 10.1109/ICMEW.2016.7574699 . hal-01415724

**HAL Id: hal-01415724**
**https://hal.science/hal-01415724**

Submitted on 13 Dec 2016

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

# Computer Generated Hologram from Multiview-plus-Depth data considering specular reflections

Antonin Gilles[1*] Patrick Gioia[1,2] Rémi Cozot[1,3] Luce Morin[1,4]

[1] IRT b<>com Cesson-Sévigné France
[2] Orange Labs Rennes France
[3] University of Rennes 1 Rennes France
[4] INSA Rennes Rennes France

## Abstract

A novel approach for hologram computation from Multiview-plus-Depth (MVD) data is studied in this paper. The proposed method consists of three steps. First, intensity views and depth maps pairs of the scene are taken from different perspective viewpoints. Then, the 3D scene geometry is reconstructed from the MVD data as a layered point-cloud. This 3D scene reconstruction step allows us to use only a few perspective projections of the scene without sacrificing any depth cue. Furthermore, in order to take into account specular reflections, each scene point is considered to emit light differently in all the directions. Finally, light scattered by the scene is numerically propagated towards the hologram plane in order to get the final CGH. Experimental results show that the proposed method is able to provide all the human depth cues and accurate shading of the scene without producing any visible artifact.

**Keywords :** Three-dimensional imaging, Holography, Computer-Generated Hologram

*This work has been achieved within the Institute of Research and Technology b<>com, dedicated to digital technologies. It has been funded by the French government through the National Research Agency (ANR) Investment referenced ANR-A0-AIRT-07. Authors can be reached at {antonin.gilles, patrick.gioia, remi.cozot, luce.morin}@b-com.com.

## 1 Introduction

Holography is often considered as the most promising 3D visualization technology, since it can provide the most realistic and natural depth illusion to the naked eye. Indeed, it provides all the human depth cues without the need for special viewing devices and without causing eye-strain [1]. In the last decades, several methods have been proposed to generate holograms by computer calculation. Using these methods, it is possible to obtain Computer-Generated Holograms (CGH) of synthetic or existing scenes by simulating the propagation of light scattered by the scene towards the hologram plane.

Thanks to its attractive visualization properties, CGH may replace the stereoscopic technique in the future. However, in order to be successful, CGH computation techniques should meet several requirements. First, CGH computation methods should provide all the human depth cues in order to produce a strong depth illusion. Second, the displayed 3D scene should be as realistic as possible, which means that occlusions as well as shading should be accurately reproduced. Last but not least, in order to have application in the field of videoconferencing or telepresence systems, computation techniques should be able to produce CGH of real scenes at interactive rates.

Meeting all these requirements is still very challenging for state-of-the-art methods. These methods can be classified depending on the type of input data they use to compute the hologram: synthetic 3D scenes or multiview data.

## 1.1 Hologram computation from synthetic 3D scenes

In order to compute a hologram from a synthetic 3D scene, CGH computation techniques usually sample it by a set of primitives and calculate the hologram as the sum of complex light waves scattered by each primitive. Commonly used primitives include points (point-source approach) and planar segments (polygon-based approach).

In the point-source approach, 3D scenes are populated by a collection of self luminous points, and the hologram is computed as the sum of light waves scattered by the points towards the hologram plane [2]. The complex wave scattered by each point is computed using the monochromatic spherical light wave equation. This approach is very flexible and does not impose any restriction on the scene geometry. Moreover, Computer Graphics (CG) shading techniques, such as Phong shading [3], can be integrated during CGH computation in order to produce photo-realistic effects [4, 5]. However, to produce shapes that appear solid and continuous, the scene needs to be sampled at very high densities, making the CGH computation prohibitively slow.

In the polygon-based approach, 3D scenes are populated by a collection of self luminous polygons, and the hologram is computed as the sum of light waves scattered by the polygons towards the hologram plane [6]. The complex wave scattered by each polygon is computed using propagation formulas such as the Angular Spectrum diffraction [7]. The computation of the Angular Spectrum diffraction involves the use of the Fast Fourier Transform (FFT) algorithm twice, and is therefore more time-consuming than the computation of the spherical light wave scattered by a single point. However, complex waves scattered by scene points located within a single planar segment are calculated all at once using the Angular Spectrum diffraction. As for the point-source approach, CG shading techniques can be integrated during CGH computation in order to produce photo-realistic effects. These include Blinn and Torrance Sparrow reflection models [8], Finite-difference time-domain (FDTD) method [9] and Phong reflection model [10].

All these methods present the common advantage of being able to produce CGHs of accurately shaded scenes with all the human depth cues. However, they require the input scene to be accurately modeled, with the position of light sources and materials composing each scene object fully described. As a consequence, these methods cannot be easily applied to real existing scenes.

## 1.2 Hologram computation from multiview data

In order to compute CGHs of real existing scenes, it is possible to use multiview data. Two approaches are commonly used for CGH computation from multiview data: the multiviewpoint projection based methods and the holographic stereogram approach.

Multi-viewpoint projection based methods generate holograms by first acquiring two-dimensional (2D) projections of a 3D scene using a 2D camera array or by shifting a single camera along the vertical and horizontal axis. The number of acquired images should be equal to the number of hologram pixels. Each hologram pixel is then computed by multiplying its corresponding projection image by a given complex point spread function (PSF) and by summation of the resulting inner product [11]. Occlusions as well as shading of the scene are naturally recorded in the acquired projections, and the resulting CGH is able to provide all the human depth cues. However, since each projection image yields a single hologram pixel, these methods require a huge number of images to be acquired, especially for high resolution CGHs. In order to reduce the number of required projections, several methods have been proposed. These include Horizontal Parallax Only (HPO) holograms [12], view synthesis algorithms [13], in which the computational complexity is increased, and integral imaging capture systems [14, 15], which make a trade-off between the number of projections and their resolution.

The holographic stereogram approach is another class of CGH computation methods which use multiview data as input. In this approach, the hologram is spatially partitioned into several holographic elements, called hogels. The hogels angularly multiplex the 2D projections of the scene towards the corre-

sponding viewing directions. This approach is computationally very efficient and does not require as many projection images as for multi-viewpoint projection based methods. Moreover, occlusions and shading of the scene are naturally recorded in the multiplexed 2D views. However, since each hogel only corresponds to a 2D parallax view, the accommodation cue is sacrificed, which strongly limits the naturalism of the perceived 3D effect. Several methods have been proposed to improve the depth illusion provided by the holographic stereogram, including phase-added stereograms [16, 17], diffraction-specific coherent panoramagram [18], and ray sampling plane techniques [19].

As discussed above, it is still very challenging to compute CGHs of real existing scenes which provide all the human depth cues and accurate shading of the scene with only a few projection images. In order to overcome these limitations, we propose a novel approach for CGH computation from Multiview-plus-Depth (MVD) data. A method for CGH computation from MVD data has been proposed in [20]. However, in this paper, authors did not take into account specular reflections, and the displayed scene was totally diffuse.

A preliminary version of the proposed method which can only render diffuse scenes is presented in Section 2, and the generalization of this method to scenes with specular reflections is described in Section 3. Finally, experimental results are analyzed in Section 4.

# 2 Hologram computation for a diffuse scene

In this section we limit ourselves to the rendering of diffuse scenes. The proposed method consists of three steps, which are detailed in the following. First, 2D plus depth projections of the scene are taken from different perspective viewpoints. Then, the 3D scene geometry is reconstructed from MVD data as a layered point-cloud. Finally, light scattered by the scene is numerically propagated towards the hologram plane and encoded into the final CGH.

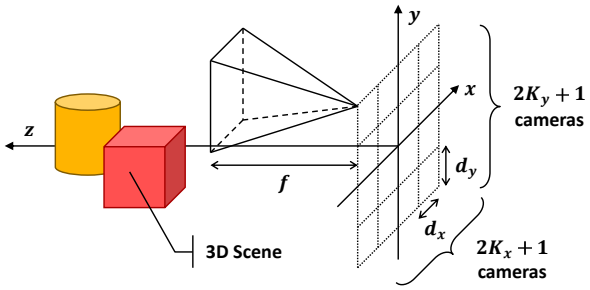



Figure 1: Multiview-plus-Depth capture geometry.

## 2.1 Step 1: Multiview-plus-Depth capture

The first step of the proposed method is to capture 2D plus depth projections of the scene from different perspective viewpoints. Although other camera layouts may also be used to acquire the MVD data, the proposed method uses $(2K_x+1)\times(2K_y+1)$ cameras aligned in parallel on a 2D grid, as shown in Figure 1.

The coordinate system is defined by $(x, y, z)$ so that the camera matrix lies on the $(x, y, 0)$ plane and the 3D scene is located between $z_{\min}$ and $z_{\max}$. We call $d_x$ and $d_y$ the distance between the cameras along the $x$ and $y$ axis, respectively, and $f$ the focal length of the cameras. The acquired intensity views and depth maps pairs have resolution $N_x \times N_y$ and pixel pitch $p$. Finally, the cameras are numbered by $(i, j)$, with $i \in \{-K_x, ..., K_x\}$ and $j \in \{-K_y, ..., K_y\}$.

The projection matrix of camera $(i, j)$ is therefore given by

$$P_{i,j} = \begin{bmatrix} \alpha & 0 & u_0 & \alpha x_i \\ 0 & \alpha & v_0 & \alpha y_j \\ 0 & 0 & 1 & 0 \end{bmatrix}, \qquad (1)$$

where $\alpha = f/p$ is the focal length expressed in pixels, $(x_i, y_j)$ is the camera location and $(u_0, v_0)$ corresponds to the principal point pixel's coordinates, given by

$$x_i = id_x \quad ; \quad y_j = jd_y \quad ; \quad u_0 = \tfrac{N_x}{2} \quad ; \quad v_0 = \tfrac{N_y}{2}. \qquad (2)$$

## 2.2 Step 2: Scene geometry reconstruction

The second step of the proposed method is to reconstruct the 3D scene geometry from the MVD data. We call $I_{i,j}$ and $D_{i,j}$ the intensity and depth images captured by camera $(i,j)$.

Since $D_{i,j}$ is encoded as an 8-bits gray level image, each pixel $(u,v)$ can be projected back to a 3D point whose coordinates are given by

$$\begin{pmatrix} x_{i,u} \\ y_{j,v} \\ z_d \end{pmatrix} = \begin{pmatrix} \frac{z(u-u_0)}{\alpha} - x_i \\ \frac{z(v-v_0)}{\alpha} - y_j \\ \frac{255-d}{255}(z_{\max} - z_{\min}) + z_{\min} \end{pmatrix}, \quad (3)$$

where $d = D_{i,j}(u,v)$.

The projected 3D points are thus naturally sliced into 256 depth layers separated by a distance $d_z = \frac{z_{\max}-z_{\min}}{255}$. Furthermore, in order to improve the visual quality, each projected 3D point is attributed a Gaussian distribution, as proposed in [21].

Each depth layer $d$ is therefore considered to operate as a surface source of light which emits a complex wave $o_d$ given by

$$o_d(x,y) = \sum_{\substack{i,j,u,v \\ D_{i,j}(u,v)=d}} \sqrt{I_{i,j}(u,v)} \exp\left[j\phi(x,y)\right] \times \exp\left[-\frac{(x-x_{i,u})^2 + (y-y_{j,v})^2}{2s^2}\right], \quad (4)$$

where $s^2$ is the variance of the Gaussian distribution, which controls the surface of each 3D point, and $\phi(x,y) \in [0; 2\pi[$ is the initial phase, set to a uniform random value in order to keep the diffuseness of the scene.

## 2.3 Step 3: Light propagation and hologram encoding

The third step of the proposed method is to compute the propagation of light scattered by the scene towards the hologram plane. The hologram lies in the $(x,y,0)$ plane, as shown in Figure 2. As stated in Section 2.2, the 3D scene is sliced into 256 depth layers numbered from 0 to 255, from the farthest to the nearest to the hologram plane.

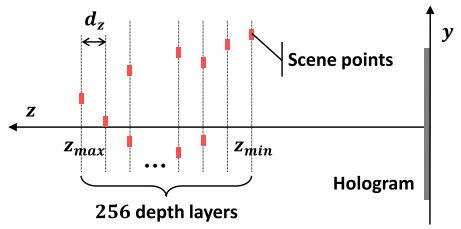



Figure 2: Numerical propagation of light scattered by the scene towards the hologram plane, side view.

Light scattered by the scene is numerically propagated and shielded from one layer to another using the recurrence formula

$$\begin{aligned} u_0(x,y) &= o_0(x,y), \\ u_d(x,y) &= o_d(x,y) + \mathcal{O}_d\left\{\mathcal{P}_{d_z}\left\{u_{d-1}\right\}\right\}(x,y), \end{aligned} \quad (5)$$

where $u_d$ is the total complex wave scattered by layer $d$, operator $\mathcal{O}_d$ stands for light shielding by layer $d$, and operator $\mathcal{P}_z$ stands for the numerical propagation of light between two parallel planes separated by a distance $z$.

The propagation of light is numerically computed using the Angular Spectrum diffraction [7], which expresses light propagation between two parallel planes separated by a distance $z$ as

$$\mathcal{P}_z\left\{u\right\} = \mathcal{F}^{-1}\left\{\mathcal{F}\left\{u\right\} \mathrm{e}^{-j2\pi\sqrt{\lambda^{-2}-f_x^2-f_y^2}z}\right\}, \quad (6)$$

where $\lambda$ is the wavelength of light, $f_x$ and $f_y$ are the spatial frequencies, and $\mathcal{F}$ and $\mathcal{F}^{-1}$ are respectively the forward and inverse Fourier Transform. These transforms can be computed using the Fast Fourier Transform algorithm (FFT).

Light shielding is performed using a binary cross-section mask function, as proposed in [22]. An occluding obstacle is located within layer $d$. Complex wave $u'_d$ incident on layer $d$ is given by

$$u'_d(x,y) = \mathcal{P}_{d_z}\left\{u_{d-1}\right\}(x,y), \quad (7)$$

where $u_{d-1}$ is the complex wave scattered by layer $d-1$. Part of $u'_d$ is shielded by the occluding scene points, and its amplitude vanishes in the area of the

obstacle. This is expressed by multiplying $u'_d$ by a binary mask function $m_d$ that has value zero on the obstacle and one elsewhere. $\mathcal{O}_d$ is thus given by

$$\mathcal{O}_d \{u'_d\}(x,y) = m_d(x,y)u'_d(x,y). \tag{8}$$

Finally, the complex wave scattered by layer $N_z - 1$ is numerically propagated to the hologram plane in order to obtain the object wave $O$, according to

$$O(x,y) = \mathcal{P}_{z_{\min}} \{u_{255}\}(x,y). \tag{9}$$

The final CGH is computed using the bipolar intensity formula [23], such that

$$H = \Re \{OR^*\}, \tag{10}$$

where $R$ is the reference plane wave, and $\Re\{C\}$ and $C^*$ are the real part and complex conjugate of $C$, respectively.

# 3 Rendering specular reflections

This section describes the generalization of the proposed method to the rendering of specular reflections. Indeed, MVD data do not only capture the geometry of the scene but also its shading, including specular reflections. These variations of brightness are totally lost when using the diffuse-only version of the method presented in Section 2.

## 3.1 Role of the initial phase distribution

As stated in Section 2.2, each depth layer $d$ is considered to operate as a surface source of light which emits a complex wave $o_d$ given by Eq. (4). In Eq. (4), the phase factor $h(x,y) = \exp[j\phi(x,y)]$ plays the role of a diffuser that spreads light in space. As a consequence, $\phi(x,y)$ is important for rendering specular reflections because it determines the diffuseness and direction of light emitted by each scene point.

We call $H$ the Fourier spectrum of $h$, given by

$$\begin{aligned} H(f_x,f_y) &= \mathcal{F}\{h\}(f_x,f_y) \\ &= \iint_{-\infty}^{+\infty} h(x,y)\exp[-j2\pi(f_x x + f_y y)]\mathrm{d}x\mathrm{d}y, \end{aligned} \tag{11}$$

where $f_x$ and $f_y$ are the spacial frequencies with respect to the $x$ and $y$ axis, respectively.

The Angular Spectrum diffraction [7] expresses the propagation of light scattered by $h$ over distance $-z$ as

$$\begin{aligned} h(x,y,z) &= \mathcal{F}^{-1}\left\{H\mathrm{e}^{j2\pi\omega z}\right\}(x,y) \\ &= \iint_{-\infty}^{+\infty} H(f_x,f_y)\mathrm{e}^{j2\pi(f_x x + f_y y + \omega z)}\mathrm{d}f_x\mathrm{d}f_y, \end{aligned} \tag{12}$$

where $\omega = \sqrt{\lambda^{-2} - f_x^2 - f_y^2}$.

In order to understand the behavior of $h$, we consider the time-independent complex form of a plane wave, given by

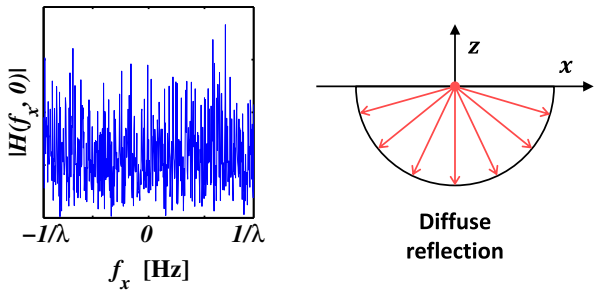
$$p(x,y,z) = A\exp(j\mathbf{k}\cdot\mathbf{r}), \tag{13}$$

where $A$ is the complex amplitude, $\mathbf{k}$ is the wave vector, of magnitude $|\mathbf{k}| = 2\pi/\lambda$, and $\mathbf{r} = \begin{pmatrix} x & y & z \end{pmatrix}^t$ is the position vector.

According to Eq. (12), $h$ can therefore be interpreted as a superposition of plane waves of complex amplitude $H(f_x,f_y)$ propagating with wave vector $\mathbf{k}$ given by

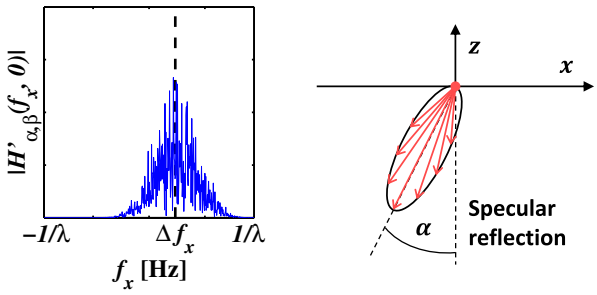
$$\mathbf{k} = 2\pi \begin{pmatrix} f_x \\ f_y \\ \sqrt{\lambda^{-2} - f_x^2 - f_y^2} \end{pmatrix}. \tag{14}$$

In the preliminary version of the proposed method, $\phi(x,y)$ is set to a uniform random value, and $h$ has therefore a broadband spectrum, as shown in Figure 3a. This means that $h$ is composed by an infinity of plane waves propagating in all the directions in space. As a consequence, each 3D scene point spreads light equally in all the directions, as shown in Figure 3b.




(a) (b)

Figure 3: (a) Fourier spectrum of the original diffuser, (b) illustration of diffuse reflection.




(a) (b)

Figure 4: (a) Fourier spectrum of the specular diffuser, (b) illustration of specular reflection.

### 3.2 Creation of a new phase distribution

In order to reproduce specular reflections, scene points should be initialized with a new phase distribution able to concentrate the energy of emitted light around a given direction, as shown in Figure 4b.

We consider the direction of emitted light given by angles $\alpha$ and $\beta$ around the vertical and horizontal axes, respectively. We call $h'_{\alpha,\beta}$ the diffuser which scatters light around this direction. Its Fourier spectrum should therefore be band-limited and shifted from the origin by $(\Delta f_x, \Delta f_y)$, as shown in Figure 4a. According to the grating equation [7], the shift in frequencies is given by

$$\Delta f_x = \frac{\sin(\alpha)}{\lambda} \quad ; \quad \Delta f_y = \frac{\sin(\beta)}{\lambda}. \tag{15}$$

We therefore multiply $H$ by a 2D Gaussian window in order to get a band-limited Fourier spectrum given by

$$H'_{\alpha,\beta}(f_x, f_y) = H(f_x, f_y) \exp\left[-\frac{(f_x - \Delta f_x)^2 + (f_y - \Delta f_y)^2}{2\sigma^2}\right] \tag{16}$$

where $\sigma$ is an adjustable parameter which controls the bandwidth of $H'_{\alpha,\beta}$ and therefore the diffuseness of each scene point. This parameter depends on the wavelength of light.

Since $\left|\mathcal{F}^{-1}\left\{H'_{\alpha,\beta}\right\}\right| \not\equiv 1$, we define the new phase distribution as

$$\phi'_{\alpha,\beta}(x,y) = arg\left\{\mathcal{F}^{-1}\left\{H'_{\alpha,\beta}\right\}(x,y)\right\}, \tag{17}$$

where $arg\{C\}$ is the phase of $C$. The new diffuser is therefore given by

$$h'_{\alpha,\beta}(x,y) = \exp\left[j\phi'_{\alpha,\beta}(x,y)\right]. \tag{18}$$

Since a new diffuser has to be computed for every possible values of $\alpha$ and $\beta$, this process is time consuming. In order to reduce the computational load, we first compute $h'_{0,0}$ using Eq. (18). Then, the other diffusers $h'_{\alpha,\beta}$ are approximated by simply multiplying $h'_{0,0}$ by a plane wave propagating in the direction given by angles $\alpha$ and $\beta$, such that

$$h'_{\alpha,\beta}(x,y) = h'_{0,0}(x,y) \exp\left[j2\pi(x\Delta f_x + y\Delta f_y)\right]. \tag{19}$$

### 3.3 Overall algorithm

In order to generalize it to the rendering of specular reflections, the proposed method is modified as follows.

First, 2D plus depth projections of the scene are taken from different perspective viewpoints, as described in Section 2.1.

Then, the 3D scene geometry is reconstructed from MVD data as a layered point cloud, as described in

Section 2.2. However, this time, the complex wave $o_d$ emitted by each depth layer $d$ is given by

$$o_d(x,y) = \sum_{\substack{i,j,u,v \\ D_{i,j}(u,v)=d}} \sqrt{I_{i,j}(u,v)} h'_{\theta_u,\theta_v}(x,y) \times \exp\left[-\frac{(x-x_{i,u})^2+(y-y_{j,v})^2}{2s^2}\right], \quad (20)$$

where

$$\theta_u = \arctan\left(\frac{(u-u_0)p}{f}\right) \quad ; \quad \theta_v = \arctan\left(\frac{(v-v_0)p}{f}\right) \quad (21)$$

define the direction from which the 3D point intensity has been captured by the camera.

Finally, light scattered by the scene is numerically propagated towards the hologram plane and encoded into the final CGH, as described in Section 2.3.

# 4 Experimental results

The proposed method was implemented in C++/CUDA on a PC system employing an Intel Core i7-4930K CPU operating at 3.40 GHz, a main memory of 16 GB and an operating system of Microsoft Windows 8 as well as three GPUs NVIDIA GeForce GTX 780Ti.

## 4.1 Hologram synthesis parameters

For the experiments, we used Unity3D software to capture the MVD projections of a highly reflective car. In this software, we set up $9 \times 9$ virtual cameras with resolution $1000 \times 1000$, focal length 2.0mm and pixel pitch $4.0\mu$m. The distance between the cameras was set to 0.3mm along the vertical and horizontal axes and the scene was located between $z_{\min} = 2.0$mm and $z_{\max} = 5.0$mm.

For the 3D scene geometry reconstruction step, we set the standard deviation $s$ of the 3D points' Gaussian distribution to be equal to $2.5\mu$m. The points' diffuseness parameter $\sigma$ was set to $94,224$Hz, $113,353$Hz and $127,492$Hz for the Red, Green and Blue channels, respectively. Finally, the hologram is sampled on a regular 2D grid of resolution $8192 \times 8192$ with sampling pitch $0.36\mu$m. The wavelengths are set to 640nm, 532nm and 473nm for the Red, Green and Blue channels, respectively.

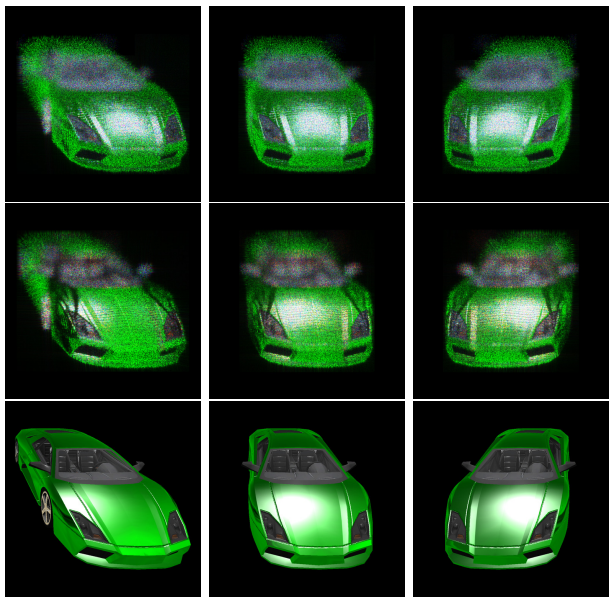


Figure 5: Scene images numerically reconstructed from the CGHs patterns generated by the diffuse-only version of the method (first row) and by the proposed method (second row), captured from different viewpoints. For the sake of comparison, Computer Graphics images synthesized from the same viewpoints are shown on the third row.

In the following, we compare the proposed method presented in Section 3.3 with its diffuse-only version, presented in Section 2.

## 4.2 Reconstructed scene images

Figure 5 shows the scene images numerically reconstructed from the CGHs patterns generated by the diffuse-only version of the method (first row) and by the proposed method (second row), captured from different viewpoints. For the sake of comparison, Computer Graphics images synthesized from the same viewpoints are shown on the third row. These experimental results show that both the proposed method and its diffuse-only version are able to produce correct motion parallax and occlusion effect,

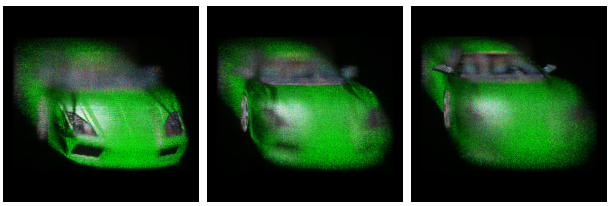


Figure 6: Scene images numerically reconstructed from the CGH generated by the proposed method, focused on different depths.

even if only a few MVD projections of the scene are used as input data.

In order to demonstrate the capability of reproducing specular reflections, we compare the numerically reconstructed scene images to their corresponding Computer Graphics images. As shown in the first row, the diffuse-only version of the method is unable to reproduce specular reflections. Indeed, the intensity of light reflected by the car hood remains the same in all the directions, whereas it should vary according to the viewpoint. On the other hand, specular reflections are reproduced without any visible artifact using the proposed method, as shown in the second row.

Finally, Figure 6 shows the scene images numerically reconstructed from the CGH generated by the proposed method, focused on different depths. As shown in this figure, when focusing on the headlights, the rear of the car is blurred, and vice versa. These results show that the proposed method can produce an accurate accommodation cue of the 3D scene with continuous depth change.

### 4.3 Hologram calculation time

In order to evaluate the performance of the proposed method, we generated six holograms of the reflective car scene with different number of MVD projections. Figure 7 shows the CGH calculation time depending on the number of MVD projections using the proposed method and its diffuse-only version.

The third step does not depend on the MVD projections and is exactly the same for the two methods. As a consequence, its calculation time remains

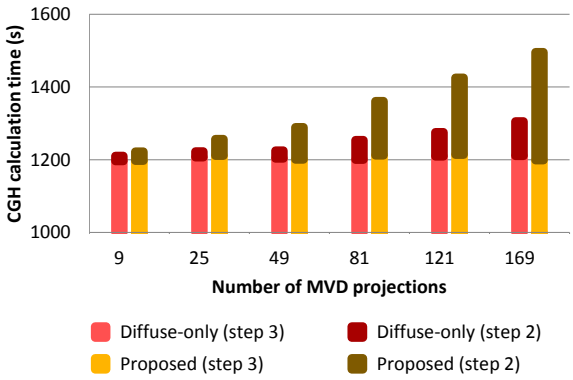



Figure 7: CGH calculation time depending on the number of MVD projections using the proposed method and its diffuse-only version.

constant for both of them. On the contrary, since the number of reconstructed scene points depends on the number of MVD projections, the calculation time of the second step increases linearly with it. Moreover, it is always more important for the proposed method than for its diffuse-only version. This is due to the fact that the computation of $h'_{\alpha,\beta}$ is more time-expensive than the computation of $h$. However, in all cases, most of the CGH calculation time is spent during the third step. As a consequence, the CGH calculation time of the proposed method has been increased only by 0.99%, 5.22%, 8.57% and 14.58% when the number of MVD projections is equal to 9, 49, 81 and 169, respectively, compared to the diffuse-only version.

## 5 Conclusion

In this paper, we proposed a CGH computation method from MVD data considering specular reflections. The proposed method consists of three steps. First, 2D plus depth projections of the scene are taken from different perspective viewpoints. Then, the 3D scene geometry is reconstructed from the MVD data as a layered point-cloud. In order to take into account specular reflections, each scene point is considered to emit light differently in all the direc-

tions. Finally, light scattered by the scene is numerically propagated towards the hologram plane and encoded into the final CGH.

Experimental results reveal that the proposed method is able to produce motion parallax with occlusion effect and accurate accommodation cue even if only a few MVD projections of the scene are used as input data. Furthermore, specular reflections are accurately reproduced with a small computational overhead. The validity of the proposed method was only verified with a fully synthetic scene, so in future study we plan to evaluate its sensitivity to errors in depth data when capturing a natural scene with real cameras.

The authors would like to thank Kenji Yamamoto's team from the National Institute of Information and Communications Technology for their warm welcome, for the very interesting discussions and for giving us access to their holographic display.

# References

[1] Ulf Schnars and Werner Jptner. *Digital Holography: Digital Hologram Recording, Numerical Reconstruction, and Related Techniques*. Springer Science & Business Media, December 2005.

[2] B. R. Brown and A. W. Lohmann. Complex Spatial Filtering with Binary Masks. *Applied Optics*, 5(6):967–969, June 1966.

[3] Bui Tuong Phong. Illumination for Computer Generated Pictures. *Commun. ACM*, 18(6):311–317, 1975.

[4] Tsubasa Ichikawa, Takuo Yoneyama, and Yuji Sakamoto. CGH calculation with the ray tracing method for the Fourier transform optical system. *Optics Express*, 21(26):32019–32031, December 2013.

[5] Takayuki Kurihara and Yasuhiro Takaki. Speckle-free, shaded 3d images produced by computer-generated holography. *Optics Express*, 21(4):4044–4054, 2013.

[6] Detlef Leseberg and Christian Frre. Computer-generated holograms of 3-D objects composed of tilted planar segments. *Applied Optics*, 27(14):3020–3024, July 1988.

[7] Joseph W. Goodman. *Introduction to Fourier Optics*. Roberts and Company Publishers, Englewood, Colo, 3rd edition, 2005.

[8] Kazuhiro Yamaguchi and Yuji Sakamoto. Computer-generated holograms considering background reflection on various object shapes with reflectance distributions. volume 7619, pages 761909–761909–11, 2010.

[9] Tsubasa Ichikawa, Yuji Sakamoto, Agus Subagyo, and Kazuhisa Sueoka. Calculation method of reflectance distributions for computer-generated holograms using the finite-difference time-domain method. *Applied Optics*, 50(34):H211–H219, 2011.

[10] Hirohito Nishi, Kyoji Matsushima, and Sumio Nakahara. Rendering of specular surfaces in polygon-based computer-generated holograms. *Applied Optics*, 50(34):H245, December 2011.

[11] Natan T. Shaked, Barak Katz, and Joseph Rosen. Review of three-dimensional holographic imaging by multiple-viewpoint-projection based methods. *Applied Optics*, 48(34):H120–H136, December 2009.

[12] Youzhi Li, David Abookasis, and Joseph Rosen. Computer-Generated Holograms of Three-Dimensional Realistic Objects Recorded Without Wave Interference. *Applied Optics*, 40(17):2864–2870, June 2001.

[13] Barak Katz, Natan T. Shaked, and Joseph Rosen. Synthesizing computer generated holograms with reduced number of perspective projections. *Optics Express*, 15(20):13250–13255, October 2007.

[14] Tomoyuki Mishina, Makoto Okui, and Fumio Okano. Calculation of holograms from elemental images captured by integral photography. *Applied Optics*, 45(17):4026–4036, June 2006.

[15] Natan T. Shaked, Joseph Rosen, and Adrian Stern. Integral holography: white-light single-shot hologram acquisition. *Optics Express*, 15(9):5754–5760, April 2007.

[16] Masahiro Yamaguchi, Hideshi Hoshino, Toshio Honda, and Nagaaki Ohyama. Phase-added stereogram: calculation of hologram using computer graphics technique. volume 1914, pages 25–31, January 1993.

[17] Hoonjong Kang, Takeshi Yamaguchi, and Hiroshi Yoshikawa. Accurate phase-added stereogram to improve the coherent stereogram. *Applied Optics*, 47(19):D44–D54, July 2008.

[18] James Barabas, Sundeep Jolly, Daniel E. Smalley, and Jr. Bove. Diffraction specific coherent panoramagrams of real scenes. volume 7957, pages 795702–795702–7, February 2011.

[19] Koki Wakunami, Hiroaki Yamashita, and Masahiro Yamaguchi. Occlusion culling for computer generated hologram based on ray-wavefront conversion. *Optics Express*, 21(19):21811–21822, September 2013.

[20] Takanori Senoh, Yasuyuki Ichihashi, Ryutaro Oi, Hisayuki Sasaki, and Kenji Yamamoto. Study of a holographic TV system based on multi-view images and depth maps. volume 8644, pages 86440A–86440A–15, 2013.

[21] Athanasia Symeonidou, David Blinder, Adrian Munteanu, and Peter Schelkens. Computer-generated holograms by multiple wavefront recording plane method with occlusion culling. *Optics Express*, 23(17):22149, August 2015.

[22] A. W. Lohmann. Three-dimensional properties of wave-fields. *Optik*, 51:105–107, 1978.

[23] Mark E. Lucente. Interactive computation of holograms using a look-up table. *Journal of Electronic Imaging*, 2(1):28–34, January 1993.