in Fellener,D., Hansen,Ch. (Ed.) EG2006 short papers conf.proceedings, pp.81-84, ISSN 1017-4656, 2006



# Digital HPO Hologram Rendering Pipeline

M. Janda[†1] and I. Hanák[‡1] and V. Skala[§1]

[1]CGDV, University of West Bohemia, Czech Republic

**Abstract**

*This paper describes a rendering pipeline for digital hologram synthesis. The pipeline is capable of handling triangle meshes, directional light sources, texture coordinates and advanced illumination models. Due to the huge computational requirements of hologram synthesis only the HPO holograms are considered.*

Categories and Subject Descriptors (according to ACM CCS): I.3.7 [Computer Graphics]: Raytracing; Hidden line/surface removal; Color, shading, shadowing, and texture

## 1. Introduction

As the computer and SLM technology, see [Har96] for details, advances the holographic displays are very close to their practical utilization. To have something to display on such displays there have to exist means of synthesizing the holograms. The synthesis is addressed in this paper.

Holograms are complex diffraction gratings with very large spatial frequencies comparable with the frequency of the diffracted light which is in a case of visible light approximately 5 MHz. The capturing of such frequency without aliasing needs a lots of samples. This makes synthesis very computationally intensive and must be therefore implemented as efficiently as possible.

The presented approach is fast because it avoids costly functions, e.g. square root, and performs pre-processing wherever it is possible to speed up the process. Although it is fast it is also capable of handling textures and advanced illumination models. It consumes exactly the same input as the classical rendering pipeline and therefore it can be seamlessly integrated into the existing rendering systems.

## 2. Digital Holography Essentials

At the beginning, there was a light wave, which can be described using the Equation 1.

$$\tilde{u}(\vec{x},t) = A(\vec{x})\exp[-i\varphi(\vec{x})]\exp(i\omega t) = \tilde{u}(r)\exp(i\omega t) \quad (1)$$

There are two terms in the equation, the spatial one – $\exp[-i\varphi(\vec{x})]$ – and the temporal one – $\exp(i\omega t) = \tilde{u}(r)\exp(i\omega t)$. Only the spatial one is the important one for synthesis. The temporal one can be omitted without consequences because the temporally coherent light is assumed.

The spatial term together with the amplitude $A(\vec{x})$ is called the complex amplitude. Each complex amplitude is, obviously, determined by the amplitude and phase and can be therefore written as a complex number, which is more convenient.

The whole holography stands on the phenomenon of interference. The interference occurs if two and more light waves are superposed. The resulting wave is obtained as simple summation of the complex amplitudes and the optical intensity of such wave is computed using the Equation 2. In digital holography, only the last two terms of the Equation 2 are computed [Luc94].

$$I = |\tilde{u}_R + \tilde{u}_S|^2 = |\tilde{u}_R|^2 + |\tilde{u}_S|^2 + \tilde{u}_R\tilde{u}_S^* + \tilde{u}_R^*\tilde{u}_S \quad (2)$$

The complex amplitude of the wavefront emerging from the scene have to be computed at each sampled point at the hologram frame. Once the complex amplitude is known, the final hologram is obtained from the Equation 3 where $\tilde{u}_S$ is the scene's complex amplitude and $\tilde{u}_R^*$ is the reference beam's complex amplitude.

† supported by MSMT Czech Rep. no. LC06008
‡ supported by 6FP NoE 3DTV project
§ suported by MSMT Czech Rep. no. 1P04LA240

© The Eurographics Association 2006.

in Fellener,D., Hansen,Ch. (Ed.) EG2006 short papers conf.proceedings, pp.81-84, ISSN 1017-4656, 2006



$$I_B = 2\Re\left\{\tilde{u}_R^* \tilde{u}_S\right\} = 2\left|\tilde{u}_R\right|\left|\tilde{u}_S\right|\cos\left(\varphi_R - \varphi_S\right) \qquad (3)$$

The last missing item to address is the computation of the complex amplitude of the wavefront emerging from a scene. Unfortunately, it is the hardest one as well because simple analytical description of a wavefront is possible only for a point light source. The spherical wavefront of a point light source is governed by the Equation 4.

$$\tilde{u}(x,y,z) = \frac{U}{r}\exp(i\varphi)\exp(ikr) \qquad (4)$$

$U$ relates to the energy radited by the point source
$k = 2\pi/\lambda$
$r = \sqrt{x^2+y^2+z^2}$

The whole scene has to be filled with lots of points and the final wavefront is then summation of their wavefront. The difficulty with this is that the amount of points has to be huge to obtain reasonably accurate result. Another problem is that the wavefront of some points can be blocked by occluding geometry so the summation has to be augmented with some visibility evaluator.

To reduce the amount of points involved in this summation the horizontal parallax only (HPO) holograms were introduced [Luc94]. Instead of considering every point of a scene only those having the same Y coordinate as the currently computed sample point are considered. Moreover the sampling in the vertical direction can be reduced.

For more details about holography see [Har96] and [Goo05]

## 3. Previous Work

The rendering method introduced in [JHS06] consists of several steps:

- Obtaining a slice of scene's geometry
- Removing backfacing parts
- Sampling the hemicircle above each hologram sample
- Choosing and accumulating the complex amplitude of the closest scene's point from each sampled direction
- Computing the interference pattern

The whole process of synthesis is depicted in the Figure 1. From each sampled point are cast rays into the scene and the closest points found are accumulated to obtain a guess of the final wavefront.

In this paper is presented a description of the enhancements of this method. The enhanced method is capable of handling textures and advanced illumination models so a complete rendering pipeline is obtained. All acceleration features are also included.

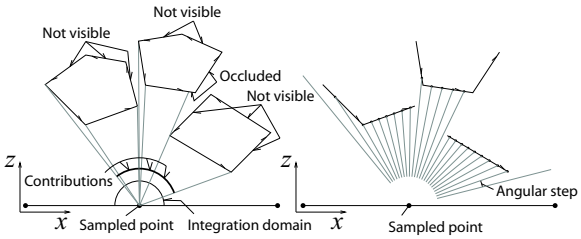


**Figure 1:** *Raycasting method for wavefront estimation.*

## 4. Rendering Pipeline

All relevant stages of the pipeline are described in this section . They are:

- Efficient edge oriented scanline slice retrieval
- Surface data interpolation
- Efficient angular sampling
- Advanced illumination model evaluation

### 4.1. Geometry Slicing

Because only HPO holograms are assumed the scene is reduced to its single planar slice consisting of polygons and polylines which are then utilized in the angular sampling stage. The method for obtaining the slice is almost the same as the scanline conversion of a triangle. The scanning is performed from up to down (up means Y axis).

The scanning is edge oriented. The list of active edges is maintained for each slice position. Each of the active edges has indices pointing at the opposing edge to the left and to the right. The orientation of the triangle is determined by the Y axis and the triangles' normal.

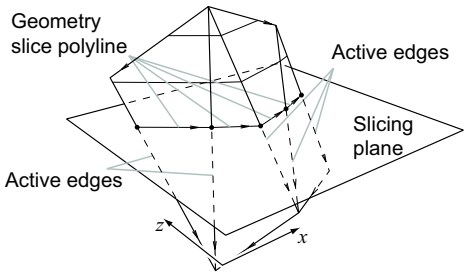


**Figure 2:** *Geometry slice retrieval.*

The intersection points for the new position of the slicing plane are computed from the previous ones by adding the vector, corresponding to the Y coordinate change. This change is constant for the whole edge because edges are linear objects.

For each scanline and for each active edge is created corresponding slice vertex. Then the slice edges are generated

© The Eurographics Association 2006.

in Fellener,D., Hansen,Ch. (Ed.) EG2006 short papers conf.proceedings, pp.81-84, ISSN 1017-4656, 2006



using the opposing edge information. Since the triangles are aware of their orientation the slice edges are all oriented from left to right. This orientation is utilized when visibility is evaluated, see [JHS06]. Vertices and edges from all objects are joined into one list and passed into the next stage of the pipeline.

### 4.2. Surface Data Interpolation

For the shading purposes some data has to be interpolated across the triangle surface, e.g. texture coordinates. The data needs to be interpolated form values stored in vertices.

The first phase of interpolation is performed during the geometry slicing where each scanned edge has data record associated. The data change corresponding to the change of Y position of the slicing plane is precomputed and the values at the next slice position are obtained by simply adding the difference to the previous ones.

Once the slice edges are obtained, each edge contains interpolated data values at their first vertex and then the data values difference between the second and the first vertex. This configuration reflects the angular sampling principle. The fraction of the difference corresponding to the sampled point is added to the base values to obtain the correct values. The fraction is determined as ratio $\mathbf{L}_S/\mathbf{L}$, see Figure 4.

The texture coordinates were mentioned as a example of data, but any other attributes can be added, e.g. vertex colours. Normals are also usually interpolated but the normals has to be normalized, which is quite costly operation. This can however be replaced by the normal map providing the texture fetch is faster than the normal normalization.

### 4.3. Angular Sampling

Next stage of the pipeline is the angular sampling. It evaluates the occlusion, illumination model and the phase at the hologram plane, see [JHS06].

The sampled angles are the same for the whole synthesis pass. During the sampling the vector representing a direction from the sampled point at every sampled angle is needed for illumination model evaluation. It is the viewer vector. These vectors are precomputed into a table.

The sine's of the sampling angles are also needed for the Z depth wavelength aligning. This aligning is not performed in the Z direction but it is rather performed in the sampled direction. This distance correction $d_c$ is computed from the triangle depicted in the Figure 3 as $z_c/\sin\alpha$.

The line sampling is performed using the sine law in the triangle. The triangle is depicted in the figure 4. The $\beta$ angle is gradually decreased and the $\alpha$ angle is gradually increased as the sampled angle advances counter clock wise. The $\alpha$ angle is used for recalculation of the triangle ratio. The $\gamma$ angle is used for evaluation of the distance d and $\beta$ angle

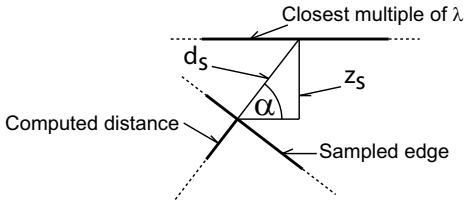


**Figure 3:** *Distance correction.*

is used for evaluation of the length used for finding out the parameter for data records interpolation.

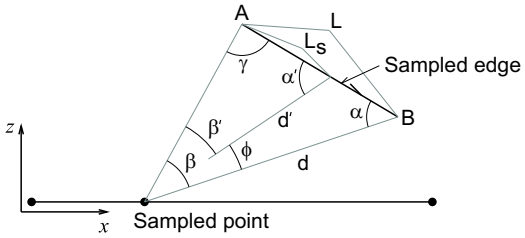


**Figure 4:** *Distance evaluation.*

This uniform change of the angles is advantage because the differential equation for cosine can be used. This differential equation is based on the well known reflection vector evaluation $\mathbf{R} = 2\mathbf{N}(\cos\alpha) - \mathbf{L}$. The cosine corresponds with the X coordinate of the vectors. The new cosine is obtained from the two previous values as is apparent form the Equation 5. The sine function can be computed similarly only the value of $\pi/2$ is subtracted form the initial angle.

$$\cos\beta_i = 2\cos\beta_{i-1}\underbrace{\cos\phi}_{const} - \cos\beta_{i-2} \tag{5}$$

### 4.4. Illumination Models

The data needed for evaluation of the illumination model are interpolated using the principle described in the Section 4.2. The view vector is precomputed in the table and since all light sources are assumed directional, the light vectors are also known. The one last missing component is the normal.

Normal can be interpolated in the same way as the texture coordinates but the interpolation does not preserve the required unity of the normal vector. The interpolated vector has to be normalized which requires square root operation. This can make the normal evaluation slow.

The alternative is to use normal map. The texture coordinates can be interpolated without problem and the corresponding normal can be obtained form the texture. On the other hand, the texture fetch is not trivial either. So several tests were performed and the results are in the Table 1.

© The Eurographics Association 2006.

in Fellener,D., Hansen,Ch. (Ed.) EG2006 short papers conf.proceedings, pp.81-84, ISSN 1017-4656, 2006



All necessary components are known so the illumination model can be evaluated. The simple Phong illumination model [Pho75] was used for the purposes of testing. The results proved great improvement in the look in comparison with the originally used constant shading in [JHS06]

## 5. Results

The Figure 5 contains the hologram computed by the proposed approach and the reconstruction of the scene. The details can be seen in the Figure 6.

The test scene contains two directional lights. Two objects have texture map applied. Geometry consists of 3492 faces and 100 kHz sampling frequency was used for rendering. The actual size of the hologram is $12 \times 6$ mm.

| Option | Time [s] |
|---|---|
| One light + normal map | 568 |
| One light + normal interp. | 608 |
| One light + normal interp. and 13968 faces | 652 |
| Two light + normal interp. | 748 |

**Table 1:** *Rendering times for various configurations*

## 6. Summary

The complete pipeline for rendering digital HPO holograms was proposed. It can be seamlessly integrated into existing rendering systems because it consumes the same scene content description as the standard 3D graphics pipeline and no other information is required.

There are several issues which will be addressed in the future work. The major one is that the presented holograms are quite small. The bigger holograms has to be rendered to obtain more useful viewing aperture. Additional acceleration should be achievable with specialized HW or standard GPU. Some progress in this area has already been achieved.

## 7. Acknowledgement

We would like to thank to the Faculty of Applied Science at the University of West Bohemia for its support and especially to our colleagues at the department for their support and useful comments.

Great thanks belong to the members of EU 3DTV NoE project for their support, namely Prof. Watson and his research group from University of Aberdeen, especially P. W. Benzie.

This work is supported by EU within FP6 under Grant 511568 with the acronym 3DTV.

## References

[Goo05] GOODMAN J. W.: *Introduction to Fourier Optics*, third ed. Roberts & Company Publishers, 2005.

[Har96] HARIHARAN P.: *Optical Holography: Principles, techniques, and applications*, second ed. Cambridge University Press, 1996.

[JHS06] JANDA M., HANÁK I., SKALA V.: Scanline rendering of digital hpo holograms and numerical reconstruction. In *Proc. SCCG 2006* (2006), vol. 1, pp. 66–73.

[Luc94] LUCENTE M.: *Diffraction-Specific Fringe Computation for Electro-Holography*. PhD thesis, MIT, 1994.

[Pho75] PHONG B. T.: Illumination for computer generated pictures. In *Communications of the ACM* (June 1975), vol. 18, pp. 311–317.

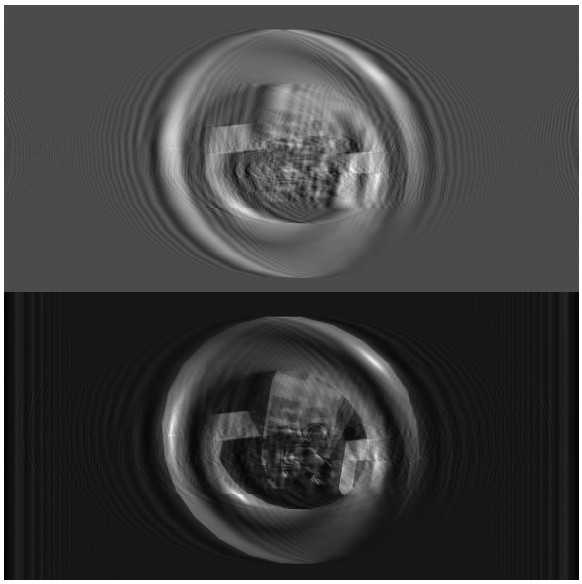

**Figure 5:** *Hologram in the top. Reconstruction in the bottom.*

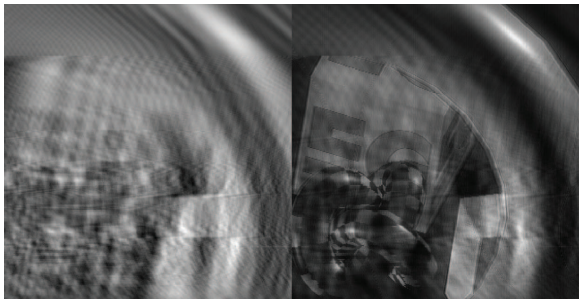

**Figure 6:** *Close up images. Hologram in the left. Reconstruction in the right.*

© The Eurographics Association 2006.