

Fourier Optics and Computer Generated Hologram

¹Ferdous Ahmed and ²Md. Sultan Mahmud

¹*Electrical and mechanical Engineering Corps, Bangladesh Army, Bangladesh*
²*Professor and Head of the Department of Basic Sciences and Humanities, University*
of Asia Pacific, Dhanmondi R/A, Dhaka-1209, Bangladesh
*E-mail:*¹*rasselbd99@gmail.com,* ²*head.bsh@uap-bd.com*

Correspondence Author: Ferdous Ahmed, Electrical and mechanical Engineering
Corps, Bangladesh Army, Bangladesh, Cell: +8801769011945, E-mail:
rasselbd99@gmail.com

Abstract

Since its conception, holography has been put to a variety of uses including data storage, security, and interferometry. However, perhaps the most popular application of holography is using the technique to make three-dimensional images [1]. This paper describes an algorithm for computationally generating holograms along with methods to fabricate the hologram. Holography has since made a jump from optics tables to computers. The phrase, "digital holography", is common place in the optics community; it describes the methods used to reconstruct holographic images from physically recorded holograms as well as the methods used to construct holograms from virtual objects using a computer. In fact we have the ability to create digital holograms from imaginary objects and then recreate the image of those objects using the same digital holograms, the entire process performed on a single computer. Computer generated holograms printed on transparencies have also been used as optical filters.

Keywords: Holography, Interferometry, Digital holography, Computer generated hologram, Optical filters, etc.

Introduction

At the fundamental level, the difference between a hologram and a conventional photograph is that conventional photography records intensity of light while a hologram records both intensity and phase. Holography is done by recording the interference pattern between coherent light and the incident light reflecting off an object [2].

The wave theory of light is utilized in computer generated holography to represent both the object and reference waves mathematically [3, 7]. With this knowledge, the superposition of these waves at any point in space can be calculated to obtain the interference pattern required for the hologram. Computer generated holograms do not require actual objects to generate the hologram as long as the light scattered or diffracted off the object could be represented mathematically [3]. The light transmission and reflection properties of the object are no longer a problem since the ideal object wave can be computed mathematically, given its structure is properly described. Thus, a computer generated holographic image is computed by numerically simulating the physical phenomena of light diffraction and interference.

A hologram is a recording of the optical interference pattern that forms at the intersection of two coherent optical beams. A key property of this *interferometric* recording is that when it is illuminated by a readout beam, the signal beam is reproduced.

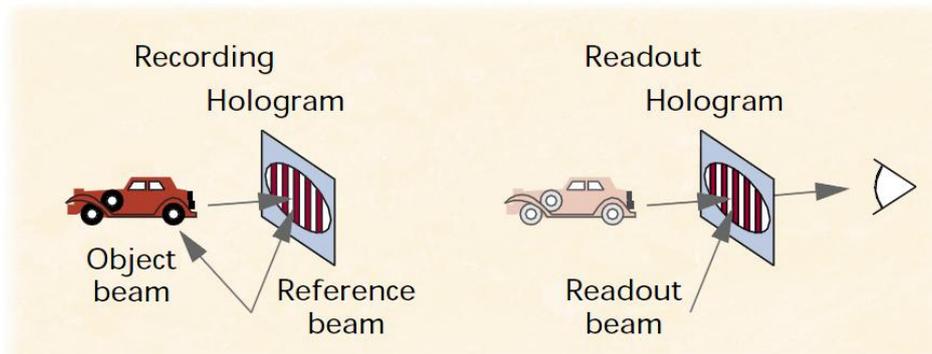


Figure 1: Hologram recording and reading process

Experimental

The optical holography technique demands the presence of the real object (for which the hologram is to be made) [4]. In optical holography recording, light from a coherent laser source scattered from the object is made to interfere with a reference light derived from the same source. The recorded fringe pattern when illuminated by the same reference light generates the object wave front (3D image). Nevertheless, in computer simulated holography, the entire recording and reconstruction processes are simulated using diffraction formula. The first computer generated hologram (CGH) was made by A. W. Lohmann and D.P. Paris in 1967. Thereafter this technique has undergone many improvements and changes in both procedure and computational techniques. Once the choice of the diffraction formula has been made, then the choice of suitable computer implementation procedures should be made. The computing environment should be chosen such that it improves speed, accuracy and also cost efficiency for that type of simulation method [5]. In our work, the Fraunhofer diffraction formula is used for simulation. The holograms thus generated using diffraction formula can be used for optical reconstruction.

Computer Generated Holography is an optical numerical technique, which avoid the traditional light interference recording process by computer numerical calculation and record the hologram directly. It does not require the actual light and the actual existence of the recording medium and reduces the difficulty of the realization of holography. Computer Generated Holograms has many applications in digital storage and display systems. Numerical generation and encryption of hologram are two important topics that have instigated abundant research and analysis works within the past twenty years. An important need for synthesizing computer generated holograms is to create optical wave fronts from objects that do not physically exist. The process of synthesizing a hologram usually consists of the following steps. The first step is to compute the propagation of the complex amplitude from the object to the hologram plane. The second step is to encode the complex amplitude as a real, nonnegative function from which the hologram can be generated on a graphic output device. The simulation of the interference fringes caused by interaction between the reference beam and object beam in conventional holography can be considered as an example for this. The next steps are to make the hologram and to reduce it to a reasonable size for diffracting light. The final photo reduction step could be eliminated by using special output devices such that hologram can be written directly in the desired size. An important property of hologram is that information about the object is spread over the entire image plane; therefore each and every portion of hologram contains the complete information of the object. This property can be utilized so as to make holograms effective for communication.

According to scalar diffraction theory, the emanating wavefronts are spherical. The scalar diffraction theory at the end of Fresnel approximation reshapes the spherical wavefronts to parabolic wavefronts. Since the parabolic approximation is valid only for small diffraction angles this approximation is also called as paraxial approximation. The far field approximation of Fresnel diffraction formula yields the Fraunhofer diffraction formula. The Fraunhofer diffraction formula is the Fourier transform of the aperture distribution. This formula greatly simplifies the calculations compared to Huygen – Fresnel diffraction formula. The simulation is employed using a simple algorithm in MATLAB language.

The far field amplitude calculation based on the Fraunhofer diffraction formula is the Fourier Transform operation. Numerically, Fourier Transform integral is solved using the Discrete Fourier Transform algorithm (DFT). But, DFT calculation increases time considerably proportional to N^2 . The speed of computation is increased by the Cooley – Turkey Algorithm called as the Fast Fourier Transform algorithm. This FFT computation time is proportional to $N \log N$. This is used to calculate the far field amplitude $U(x,y)$. If $G(x,y)$ is the Fourier transform of $F(\xi,\eta)$, then

$$G(x + pN, y + qN) = \frac{1}{N} \sum_{\xi=0}^{N-1} \sum_{\eta=0}^{N-1} F(\xi, \eta) \exp \left\{ \frac{-2\pi i}{N} (x\xi + y\eta) \right\} \times \exp \{-2\pi i(p\xi + q\eta)\} \quad (1)$$

for $x, y = 0, 1, \dots, N-1$

for all integer values of x, y, ξ, η, p and q , the second exponential term is 1. Thus

$$G(x+pN, y+qN) = G(x,y) \quad (2)$$

In the case of continuous signals, the origin of the transform domain is at its geometrical center. In Fraunhofer diffraction also, the zero frequency term is at the center of the display. In DFT, the origin can be shifted to the center, by reordering the transform coefficients. The reordering is performed by multiplying the matrix by $(-1)^{\xi+\eta}$. For $p=q=0.5$,

$$G(x + N/2, y + N/2) = G(x+pN, y+qN) \times (-1)^{\xi+\eta} \quad (3)$$

The Fourier Transforms of the object and reference waves have complex values and cannot be imaged. So, the square of the absolute values of the amplitudes (Intensity) is imaged. Fourier transform, when discretized with periodic sampling, is only the Fourier series representation of the 2D object. In our case, the matrix representing the object is real. The matrix contains N^2 elements. When Fourier transformed, the matrix becomes complex with $2N^2$ elements. It seems as if the dimension has increased but, in reality, it has not, as the Fourier transform exhibits the property of conjugate symmetry.

$$G^*(x, y) = G(-x, -y) \quad (4)$$

Due to this, almost 50% of the samples are not needed and can be generated from other samples. This is the property responsible for twin images during reconstruction.

Results and Discussions

A matrix is generated with $N \times N$ elements. Dirac delta function is generated in MATLAB by making only one element out of the total $N \times N$ elements as constant and rest of the elements are made zero. We chose the word "HOLOGRAM" as sample for analysis [4, 6]. The maximum value of the intensity is chosen as 100 and the minimum value is chosen as zero. In the case of loading, the opaque object plane may be generated using "MS Paint" which is a Windows built in program. The object is made transparent in the opaque background. The image is saved as a '*.jpg' file and it is loaded into MATLAB program using the 'imread' command. The object (HOLOGRAM) shown in the Figure-2 is used for converting into a matrix.

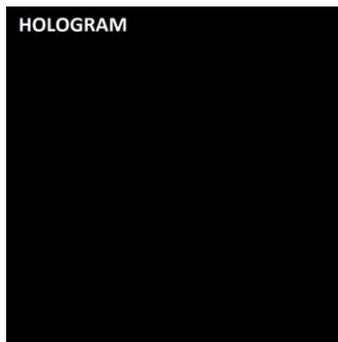


Figure 2: Image for Hologram

In MATLAB, even if the size of the object is not exactly $N \times N$ it can be used for manipulation using image resize tools (interpolation techniques). The command 'imresize' changes the size of the object to the desired size but this process creates unwanted noisy edges. Hence, in our entire work, images are created to the necessary sizes using MS Paint and not resized using MATLAB.

In MATLAB, 'fft' command is used to calculate Fourier Transform. In our case we need to use 'fft2' which is the command used to calculate 2D Fourier transform. This gives the complex amplitude distribution in the reconstruction plane. In order to make the diffraction patterns to be centered 'fftshift' command is used after all 'fft2' operations. The command 'fftshift' is equivalent to multiplying the matrix by $(-1)^{\xi+\eta}$.

The addition of the far field matrices of the object and reference is performed by matrix addition. The resulting matrix is the hologram matrix. The square of the matrix (every element multiplied on to itself and not usual matrix multiplication) gives the intensity values across the hologram plane. This kind of multiplication is done by using the (.*) command and not the (*) command. Filtering is done, using matrix subtraction. The matrix is imaged using 'imagesc' command instead of the usual 'image' command. This operation scales the pixel values of the matrix and displays the image. The reconstruction of hologram without and with filtering are shown in Figure - 3.

The matrix is squared again as explained earlier and the resulting matrix is imaged again where twin images are produced. This is in complete agreement with the theory [6]. The distribution of the frequency components in Fourier domain and energy distribution of light can be obtained from any standard Digital Signal Processing (DSP) and Quantum Electronics book [7]. The basic program followed is shown below,

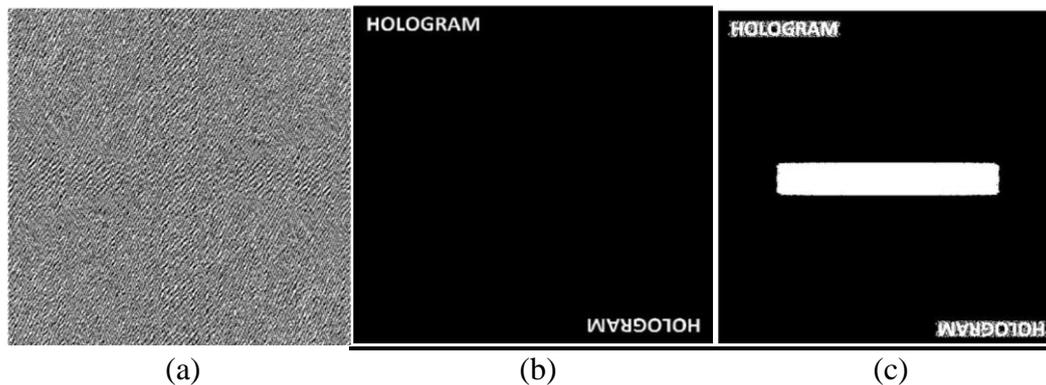


Fig-3 (a) Hologram (b) Reconstruction of hologram without filtering (c) Reconstruction of hologram with filtering

Basic Program

```
%Fraunhoffer Hologram
clc;
clear all;
N=1024;
```

%Loading the image

```
A=imread('Define location of the image');
A=A(1:N,1:N);
A=double(A);
```

%Dirac delta function

```
B=zeros(N,N);
B(513,513)=100;
```

% Generating the diffraction pattern for the object and reference

```
A1=fftshift(fft2(A));
I1=abs(A1).*abs(A1);
B1=fftshift(fft2(B));
I2=abs(B1).*abs(B1);
```

%Hologram construction and Filtering

```
D1=A1+B1;
I3=abs(D1).*abs(D1);
I4=(I3-I1);
figure, imshow(I4)
```

%Hologram reconstruction

```
D2=fftshift(fft2(I4));
I5=abs(D2).*abs(D2);
figure, imshow(I5)
```

%To see the object, reference, hologram and reconstruction

```
colormap(gray);
imagesc(I4);
imagesc(I5);
```

Conclusion

This paper discussed the basic principles and implementation of computer generated holograms. A simple algorithm for simulation of computer generated holograms was presented. Fourier transform operation was utilized to obtain the hologram of the object. The reconstruction of object image from the computer generated hologram was done. The image reconstructed from hologram contains the reproduction of the object image and its conjugate image. A method to rearrange the input image so as to identify both the object and conjugate in the reconstructed image was discussed in the paper. An interesting property of hologram is that every portion of the hologram contains information about the object.

References

- [1]. Holographic projection technologies of the future ‘Killer Applications’”
By Lance Winslow May 2007
- [2]. Tung H. Jeong “Basic Principles and Applications of Holography”, Lake Forest College, Lake Forest, Illinois; Fundamentals Of Photonics, Module 1.10, SPIE Digital Library
- [3]. Computer Generated Holography, James B Wendt
- [4]. J.W. Goodman,” Introduction to Fourier Optics”, 2nd ed., The McGraw-Hill Companies, Inc., 1996
- [5]. A. Vijaya kumar, B. J. Jackin and P.K. Palanisamy, “Computer generated Fourier holograms for undergraduate optics laboratory” www.physedu.in, Volume 28, No.4, Article Number: 4, Oct-Dec2012
- [6]. Divya P.S.¹, Sheeja M.K² “A study and simulation of computer generated Hologram”, International Journal of Advances in Engineering & Technology, July 2013. ©IJAET, ISSN: 22311963
- [7]. IEEE Journal of Selected Topics in Quantum Electronics, Vol. 4, No. 5, September/October 1998

