Computer generated Fourier holograms for undergraduate optics laboratory

A. Vijayakumar[1], B. J. Jackin [2] and P. K. Palanisamy[3]

[1]Department of Electrical Engineering
Indian Institute of Technology Madras
Chennai 600036, India.

[2]Center for Optics Research and Education,
Utsunomiya University, Japan

[3]Department of Physics
Anna University
Chennai 600025, India.

(*Submitted* 23-08-2012)

Abstract

A computer simulation technique for generating Fourier holograms using the Fraunhofer diffraction formula is discussed. These generated Fourier holograms were verified for optical reconstruction after printing them on to films and the results were compared with the simulated reconstruction. A detailed analysis was done on the computation and optical reconstruction. Filtering techniques were employed and corresponding reconstructions were verified. MATLAB was used for computer simulation. The algorithm of the MATLAB program is discussed. Using this technique, computer generated holograms can be easily generated in a PC even at undergraduate level.

1. Introduction

Holography is popularly known as the art of recording and reproducing a 3D image or scene very close to its reality. But, from the scientific perspective, "it is the recording of both amplitude and phase information contained within a propagating wavefront over a certain region of space". The earliest methods of constructing optical holograms involves the use of highly coherent light sources, high resolution recording materials, optical elements and vibration isolation table.

The optical holography technique also demands the presence of the real object (for which the hologram is to be made). 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 wavefront (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[1]. Thereafter this technique has undergone many improvements and changes in both procedure and computational techniques [2].

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. In our work, the Fraunhofer diffraction formula is used for simulation. The holograms thus generated using diffraction formula can be used for optical reconstruction.

This paper consists of four parts. The first part describes the theory of our work. The second part gives the complete details of the simulation algorithm. The third part of the paper gives the details about the optical reconstruction. The final part contains the MATLAB code which can be used for reproducing our results.

2. Theory

According to scalar diffraction theory, the emanating wavefronts are spherical [3,4]. The scalar diffraction theory at the end of Fresnel approximation reshapes the spherical wavefronts to parabolic wavefronts [5]. 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. In the following theory, the notations followed in [5] are used.

Let the object be a planar transparent object in an opaque (ξ,η) plane called as the object plane. Let (x,y) be the hologram plane which is at a distance z from the object plane. The amplitude transmittance functions for the object and the reference waves are $T_o(\xi,\eta)$ and $T_r(\xi,\eta)$ respectively. The Fresnel diffraction formula is given by

$$U(x,y)=\frac{e^{jkz}e^{j\frac{k}{2z}(x^2+y^2)}}{j\lambda z}\int_{-\infty}^{+\infty}\int_{-\infty}^{+\infty}\left\{T(\xi,\eta)\exp\left[j\frac{k}{2z}(\xi^2+\eta^2)\right]\right\} \exp\left[-j\frac{2\pi}{\lambda z}(x\xi+y\eta)\right]d\xi d\eta \quad (1)$$

$T(\xi,\eta)$ in equation (1) must be replaced by $T_o(\xi,\eta)$ and $T_r(\xi,\eta)$ for calculation of the diffracted field amplitude profile for object and reference respectively. The equivalent experimental set up (for recording hologram) for the computer simulation discussed in this paper is the inline hologram recording set up as shown in figure 1.

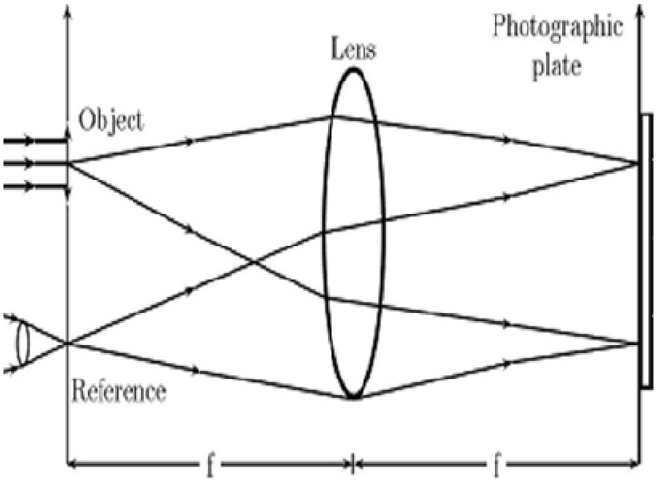


FIG.1 Hologram recording setup

The Fraunhofer diffraction formula relates the far field amplitude $U(x,y)$ (at the hologram plane) to the object plane aperture by the equation (2).

$$U(x,y)=\frac{e^{jkz}e^{j\frac{k}{2z}(x^2+y^2)}}{j\lambda z}\int_{-\infty}^{+\infty}\int_{-\infty}^{+\infty}T(\xi,\eta) \exp\left[-j\frac{2\pi}{\lambda z}(x\xi+y\eta)\right]d\xi d\eta \quad (2)$$

The above equation is evaluated at spatial frequencies $f_x = x/\lambda z$ [=cos (θ_x/λ)], $f_y = y/\lambda z$ [=cos (θ_y/λ)]. Neglecting the phase factors outside the integral, the kernel that transforms $T(\xi,\eta)$ to $U(x,y)$ is the kernel of 2D Fourier transform. Fraunhofer diffraction formula is obtained from Fresnel Integral of diffraction by the following condition

$$\frac{k}{2z}\left[\xi^2+\eta^2\right]_{\max} << 1 \quad (3)$$

i.e., for λ=632.8 nm and aperture radius of 0.01 m, z >> 1500 m. In the laboratory, the above condition is achieved by introducing a convex lens between the object plane and hologram plane such that the hologram plane coincides with the back focal plane of the lens.

The quadratic phase factor produced by the lens cancels out the phase $j\left(k/2z\right)\left[\xi^2+\eta^2\right]$. As a result,

the parabolic wavefronts reduce to plane wavefronts. The Fraunhofer diffraction equation reduces to the equation (2).

In holography, the interference pattern between the object and reference wave is recorded which is called as the hologram. In computation, the interference is obtained by the addition of the wave fields of the object and reference at the image plane. The wave fields of the object and reference at the image plane can be calculated by the formula given in [3].

By using the Linearity theorem of Fourier Transform, the two transfer functions can be added first and the sum of their wave fields at the image plane can be calculated. However, in the MATLAB code given, the fields are calculated separately and added finally. This helps to verify the program's outcome at each level.

$$U(x,y)=\frac{e^{jkf}e^{j\frac{k}{2f}(x^2+y^2)}}{j\lambda f}\int_{-\infty}^{+\infty}\int_{-\infty}^{+\infty}\left[T_o(\xi,\eta)+T_r(\xi,\eta)\right] \exp\left[-j\frac{2\pi}{\lambda f}(x\xi+y\eta)\right]d\xi d\eta \qquad (4)$$

In the real world, the measurable quantity is the intensity of light. So, when the intensity of the above expression is considered, the exponential terms outside the integral vanish.

To simplify the problem, the reference beam is considered as a plane wave in the image plane. As explained earlier, the set up for recording and reconstruction includes a convex lens such that the image plane is at the focus of the lens. Hence, a point source is chosen as the reference, which, upon Fourier transform, yields uniform illumination in the image plane. Mathematically, Dirac delta function, when Fourier transformed, produces a constant i.e., independent of the position in the image plane in spatial domain.

The intensity distribution is calculated in the image plane from equation (5).

$$I=UU^* \qquad (5)$$

The above equation gives the intensity distribution in the hologram. Hence, the transmission function *T*(*x,y*) is proportional to the intensity distribution. Details of the derivation are given in [6].

The reconstruction is done by evaluating the function *T*(*x,y*) using the Fraunhofer diffraction formula. This is given in equation (6)

$$U(u,v)=\frac{e^{jkf}e^{j\frac{k}{2f}(u^2+v^2)}}{j\lambda f}\int_{-\infty}^{+\infty}\int_{-\infty}^{+\infty}T(x,y) \exp\left[-j\frac{2\pi}{\lambda f}(xu+yv)\right]dxdy \qquad (6)$$

Again as explained earlier, the exponential term before the integral vanishes when the intensity is calculated. The result of the integration yields four terms. The first term corresponds to the Fourier transform of a constant. The second term is the autocorrelation function of $T_o(\xi,\eta)$ which produces a peak at the geometric center of the observation plane which will be removed by filtering. The third and fourth terms are of interest in holography and correspond to the interference between the object wave and reference wave from the point source. The third and fourth terms are

$$K1^*\int_{-\infty}^{+\infty}\int_{-\infty}^{+\infty}\left[\int_{-\infty}^{+\infty}\int_{-\infty}^{+\infty}\left[T_o(\xi,\eta)\right]\exp\left[-j\frac{2\pi}{\lambda f}(x\xi+y\eta)\right]d\xi d\eta\right] \exp\left[j\frac{2\pi}{\lambda f}(x\xi+y\eta)\right]\times\exp\left[-j\frac{2\pi}{\lambda f}(xu+yv)\right]dxdy$$

and

$$K1\int_{-\infty}^{+\infty}\int_{-\infty}^{+\infty}\left[\int_{-\infty}^{+\infty}\int_{-\infty}^{+\infty}\left[T_o(\xi,\eta)\right]\exp\left[-j\frac{2\pi}{\lambda f}(x\xi+y\eta)\right]d\xi d\eta\right]^* \exp\left[-j\frac{2\pi}{\lambda f}(x\xi+y\eta)\right]\times\exp\left[-j\frac{2\pi}{\lambda f}(xu+yv)\right]dxdy \qquad (7)$$

The evaluation of the two integrals yields two separate real images of the object at the (*u*,*v*) plane. In other words, two transmittance functions of the object are reproduced at the back focal plane of the lens such that one is the conjugate of the other. A detailed discussion is given in [6].

3. The simulation algorithm

The simulation Algorithm has two parts. The first part explains the simulation algorithm that is applicable to all computer programming languages. The second part gives the details of MATLAB code to perform the above simulation. Detailed information of sampling is given in [7].

Part – 1

The object and reference sizes must be chosen in accordance with the computation speed of the PC. Every computer language has its own merits and demerits. One of the well-known restrictions present in most of the languages is that when FFT of matrices are calculated, the matrix size is needed to be of the form 2^n. Let N x N be the size of the matrix.

The step by step procedure is given below

STEP – 1 Construction of Transfer function of Reference and Object

The reference as stated earlier, is the Dirac delta function. The transfer function of the matrix can be generated by making only one element of the square matrix as constant and the rest of the elements zero. There is no restriction in choosing the position. Nonetheless, if the center i.e $(((N/2)+1),((N/2)+1))$ is selected, it simplifies future hologram reconstruction. The object can be either generated or loaded. For simpler objects like an alphabet it is easier to generate and this method directly gives the matrix. In the case of other objects, like pictures, loading the image is necessary. The loaded image is then converted into a matrix for further manipulation. The resulting two matrices correspond to the transfer functions $T_o(\xi,\eta)$ and $T_r(\xi,\eta)$ in the object plane.

STEP – 2 Calculation of the Far field Amplitude *U*(*x*,*y*)

The far field amplitude calculation based on the Fraunhoffer 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

$$G(x+pN,y+qN)=\frac{1}{N}\sum_{\xi=0}^{N-1}\sum_{\eta=0}^{N-1}F(\xi,\eta)$$

$$\exp\left\{\frac{-2\pi i}{N}(x\xi+y\eta)\right\}\times\exp\left\{-2\pi i(p\xi+q\eta)\right\} \quad (8)$$

$$\text{for}\quad x,y=0,1,...N-1$$

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

$$G(x+pN,y+qN)=G(x,y) \qquad (9)$$

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 $p=q=0.5$,

$$G(x+\frac{N}{2},y+\frac{N}{2})=G(x+pN,y+qN)\times(-1)^{\xi+\eta} \quad (10)$$

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.

$$G^*(x,y) = G(-x, -y) \qquad (11)$$

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.

STEP – 3 Hologram construction and spatial filtering

The Hologram is the addition of far fields of the object and reference waves i.e interference. The Fourier transforms of the reference and object matrices are added. The square of the absolute values of the resulting matrix is imaged. The autocorrelation term derived above peaks at the center of the pattern during reconstruction. To reduce the intensity of the peak at the center, spatial filtering is done. The intensity of the Fourier transformed object matrix is subtracted from the hologram.

STEP – 4 Reconstruction of the hologram

The matrix corresponding to the filtered hologram is Fourier transformed. The intensity of the resulting matrix is calculated by squaring it. The matrix is imaged. The reconstructed image shows two images of the object out of which one is inverted according to the theory.

Part – 2

In many computer languages, the manipulation of FFT demands the matrix size to be 2^n. In Matlab this restriction is ruled out. However, in order to make this program compatible with other systems, matrix size of 256 x 256 (2^8) is chosen. The complete Matlab code for the simulation is given in the appendix of the paper.

STEP – 1

A matrix is generated with 256 x 256 elements. Dirac delta function is generated in Matlab by making only one element out of the total 256 x 256 elements as constant and rest of the elements are made zero. We chose the letters PKP (Pascal – Kirchoff - Poisson) as sample for analysis. 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 is generated using GIMP. GIMP (version 2.6) is free software compatible with Windows. The object is made transparent in the opaque background. The image is saved as a bitmap file and it is loaded into Matlab program using the 'imread' command. The object (PKP) shown in the figure 2. is used for all analysis converting into a matrix.

FIG.2 Object

In Matlab even if the size of the object is not exactly 256 x 256 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 GIMP and not resized using Matlab.

STEP – 2

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. 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 $(-1)^{\xi+\eta}$.

STEP – 3

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 hologram without and with filtering and their respective reconstructions are shown in the figure 3. The corresponding optical reconstruction is shown in figure 4.

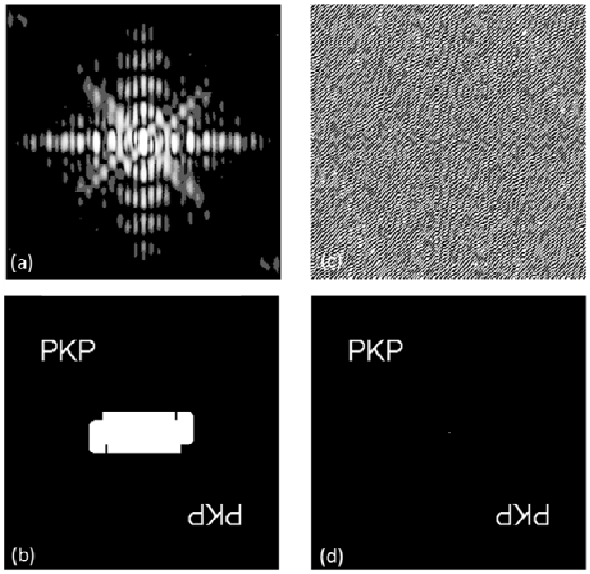


FIG.3 (a) Hologram without filtering (b) Reconstruction of hologram without filtering (c) Hologram with filtering (d) Reconstruction of hologram with filtering

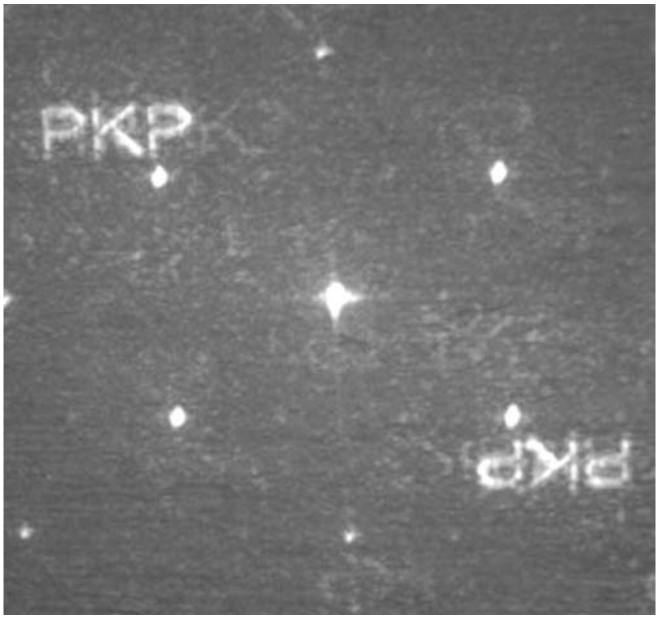
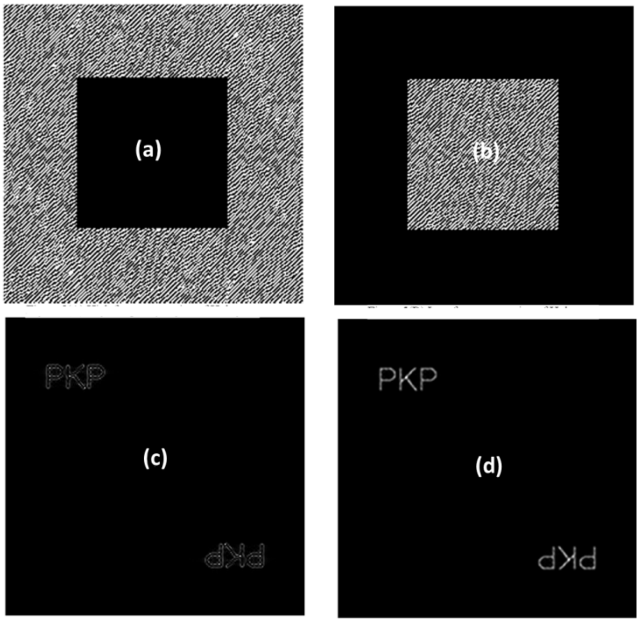


FIG.4 Result of optical reconstruction

STEP – 4

In this step, again '*fft2*' command is used. This gives the complex amplitude distribution in the reconstruction plane. The matrix is squared again as explained earlier and the resulting matrix is imaged again using '*imagesc*' command. Twin images are produced. This is in complete agreement with the theory.

To display the image, '*colormap (gray)'* is used. When the intensity ratio is varied, the reconstructed image contrast varies. In some cases, it is necessary to use '*mat2gray*' for saving the reconstructed image. For some intensity ratios, the image is fully opaque in Matlab display but is visible when equalized in GIMP. Nevertheless, for the first time Matlab user, '*colormap (prism)'* is suggested as it scales the image intensity values based on color and would not mislead the user. Instead of '*image*' command '*imagesc*' command is used to increase the visibility. The command '*subplot*' is used for viewing all results in the same page by partitioning it. Figure-5 shows the reconstructed image with high and low pass filtering. The distribution of the frequency components in Fourier domain can be obtained from any standard Digital Signal Processing (DSP) book. The high pass filtering is achieved by using only the borders of the hologram for reconstruction and vice versa for low pass filtering. The matlab simulation of the high pass and low pass filtering is shown in figure 5.



FI G.4 (a) Hologram with low frequency components removed (b) Hologram with higher frequency components (c) Reconstructed image for high pass filtering (d) Low pass filtering

When a photograph is torn into pieces, from one of the pieces, retrieval of full information is not possible. But, in a hologram, when broken into pieces, every single piece has the full information. This quality of the hologram reconstruction variation is demonstrated in computer generated Fourier holograms by reconstructing 64 x 64, 128 x 128, 192 x 192, sizes of the original 256 x 256 hologram. The convergence of the reconstructed image towards the original image with the increase in size of the hologram is visible from figure 5. The optical reconstruction is also in complete agreement with the simulation. The reconstruction images vary depending upon the position of the area in the hologram used which is evident from the filtering discussed above. In our case, the hologram areas from the corner are used for reconstruction.

The entire simulation is done using Matlab language - Version 6. The computer we used has a RAM of 512 MB and operating system is Windows XP. Hence, the computation speed is quite high.

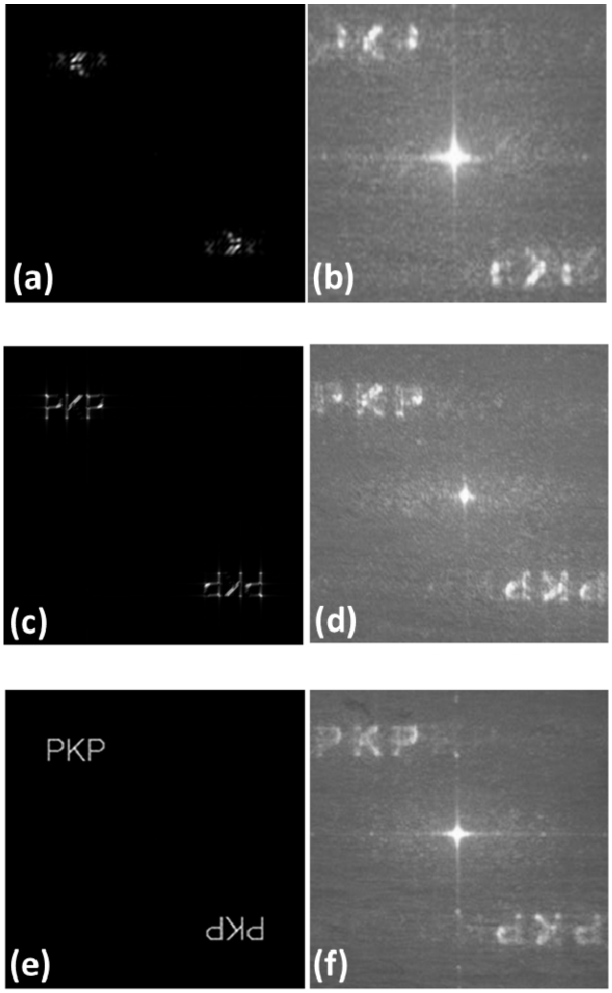


FIG.5 (a) (c) and (e) Computer reconstruction for 64 x 64, 128 x 128 and 192 x 192 hologram sizes (b) (d) and (f) Optical reconstruction for 64 x 64, 128 x 128 and 192 x 192 hologram sizes

4. Optical reconstruction

The image thus generated is transferred to the film with a resolution of 250 dpi and printed using offset style. This printing is very cheap and costs less than a rupee for making two holograms. This can be done even with a Epson printer (whose minimum resolution is 360 dpi). The holograms were reduced to sizes 8 x 8 mm, 10 x 10 mm, 12 x 12 mm respectively and printed. In our case, 10 x 10 mm produced good quality reconstruction. A He-Ne laser of λ = 632.8 nm and power 17 mW is used. The schematic of optical reconstruction set up is shown in figure 6. During reconstruction, the position of the lens is varied to fit the beam size to the hologram size. This position is crucial as it decides the size of the reconstructed image. In our case, if the hologram size is greater than 10 mm, a second lens is introduced between the first lens and the screen to expand the beam and at the same time rescale the reconstructed image to viewable size.

The photograph shows multiple images reconstructed. The central image is more intense than the other images. As explained earlier, the Fourier integral, when discretized, represents the Fourier series as it is evaluated at periodic intervals. When the Fourier transform is performed again, there is multiple image formation at regular intervals of N. The spacing between the images during optical reconstruction is a constant. The reason for the multiple image formation is explained by Macgregor [8].

In another method, the image is optically reduced using lens after printing the hologram - image on a pick sheet which improves the reconstruction quality. But, this method is expensive and costs Rs.150 per hologram. In an undergraduate physics laboratory the low cost offset printed hologram is sufficient for study.

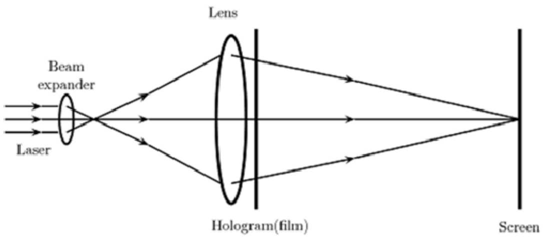


FIG.6 Schematic of optical reconstruction set up

A few other images after optical reconstruction is shown in figure 7.

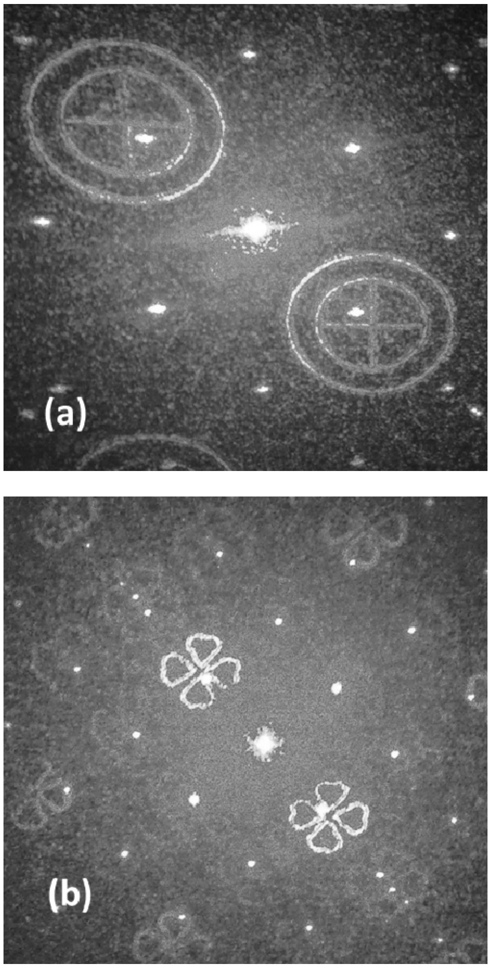


FIG. 7 Optical reconstruction of holograms made from other images

5. Conclusion

This paper suggests simple method for the computation of Fraunhofer Holograms. The simulated reconstruction and optical reconstructions were performed to verify the results. The significances of this work may be summarized as below.

The work presented in this paper can be reproduced free of cost by anyone who has access to a computer. Since this work is focused on the undergraduate level experimentation, it is carried out without any expensive charge coupled device (CCD) chip or spatial light modulator (SLM). For optical reconstruction alone you need a laser and beam expander. So this work also does not require a well-equipped applied optics laboratory. Hence this paper will be of interest to students, researchers and teachers who are interested in modern physics. Diffraction experiments, undergraduate and postgraduate level projects can be done using this method, free of cost and at minimum laboratory facility. So the work in this paper will also motivate academic level activities in modern optics.

We have discussed the fundamentals of computer generated holograms and demonstrated the generation and reconstruction of computer generated Fourier Holograms. We have done extensive research on computer generated holograms [9-10]. We are trying to implement the Fourier holograms generation using computer for our optics laboratory course. We believe that this paper will initiate active research in computer generated holograms in academic institutions.

The present work can also be used as a Ph.D tutorial for students interested to carry out research work in computer generated holograms. Multifunctional/ composite Fourier holograms can be designed by adding a lensing action, grating function etc to the standard hologram. The generation of composite holograms can be understood from the references [11].

Acknowledgements

The authors would like to thank 'Prof/Dr. Reviewer' for his suggestions, which improved the manuscript.

Appendix

```
%Fraunhoffer Hologram

%Defining the size of the object

clc;

clear all;

N=256;

%Loading the image

A=imread('D:\Holograms\pkp.bmp','bmp');
% should be suitably modified by the user

A=A(1:N,1:N);

A=double(A);

%Dirac delta function

B=zeros(N,N);

B(129,129)=100;

%To generate 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);

imwrite(I4,'D:\Holograms\pkp.bmp');

%Hologram reconstruction

D2=fftshift(fft2(I4));

I5=abs(D2).*abs(D2);

imwrite(I5, 'D:\Holograms\pkp.bmp');

%To see the object, reference, hologram and reconstruction

colormap(prism);
```

```
subplot(3,3,1);
image(A);
subplot(3,3,2);
image(B);
subplot(3,3,3);
imagesc(I1);
subplot(3,3,4);
imagesc(I2);
subplot(3,3,5);
imagesc(I4);
subplot(3,3,6);
imagesc(I5);
returnsubplot(3,3,4);
imagesc(I2);
```

References

[1] A.W. Lohmann and D.P. Paris**,** Applied Optics, 6:1739–1748, (1967).

[2] G. Tricoles, App. Opt, 26, 4351-4360 ,(1987).

[3] Wolf,E., and E.W.Marchand, J.Opt.Soc.Am., 54:587, (1964).

[4] Sommerfeld A, Optics, "Lectures on Theoretical Physics", vol IV, Academic Press Inc., NewYork, (1954).

[5] J.W. Goodman, "Introduction to Fourier Optics", San Francisco: McGraw Hill, (1968).

[6] R.J. Collier, C.B.Burckhardt, and L.H.Lin, 'Optical Holography' (Academic, New York, 1971) (p-206-217).

[7] F.Gascon, F.Salazar, Optik. 117 ,49–57 (2006).

[8] A.E.Macgregor, Am.J.Phys.60, 839-846 (1992).

[9] B.J. Jackin and T. Yatagai, Opt. Express, 18, 25546-25555, (2010).

[10] B.J.Jackin and T. Yatagai, Appl. Opt. 50, H147 – H152, (2011).

[11] D. P. Godwin, D. R. Selviah,and C.D. Carey and J. E. Midwinter, 1993, (IEEE: London, UK)147-152.