COMPUTER GENERATED HOLOGRAPHY

Arnab Ghosh, 1st Year PhD student
Rochester Institute of Technology
Chester Carlson Center for Imaging Science

ABSTRACT

Hologram is the two-dimensional record of the interference pattern that is generated when a known light source (the reference) interacts with the object. It is based on Huygens' principle that assumes light as wavelets. An interference pattern is generated when different wavelets have different phases created by a diffraction grating which implies that the light can interact constructively or destructively. Therefore, a coded grating can recreate the complex valued wavefront using the principle of diffraction. This paper is going to present the idea of Computer-Generated Holography often abbreviated as CGH where a simple 2D image is used and the Fraunhofer approximation is assumed so a coded grating pattern can be plotted on a film which when illuminated with a laser reconstructs the image.

Keywords: Hologram, Fraunhofer approximation, CGH, Detour-Phase Cell Oriented Holograms

INTRODUCTION

Hologram is a complete recording of the light wave generated as it encapsulates the frequency amplitude and the phase of the complex wavefront. Computer generated holograms try to encode the complex generated holograms onto a real, nonnegative function which can be reproduced using a plotting device and appropriately illuminating with a monochromatic wavelength.

There are different ways in which the holograms can be encoded into a real valued function. The one which is used is called the Lohmann-type cell-oriented hologram [1]. The Fourier transform of the object image is encoded using rectangular apertures on an opaque cell. The final hologram is then the combination of all the apertures for each complex value in the Fourier domain. The resultant hologram is bitonal in nature since it can have only '0' for the cell or '1' for the aperture which allows the monochromatic light to pass through.

The Lohmann-type hologram used here encodes the magnitude of the Fourier transform using the height of the aperture and the phase information is encoded by shifting the aperture laterally from the center of the aperture.

The Lohmann hologram then can be optically reconstructed by assuming the Fraunhofer diffraction region. A monochromatic light from a laser illuminates the hologram and assuming the observation plane to be at a large distance from the object we generate the Fourier transform of the pattern printed onto a transparent film.

PROCEDURE

The procedure to create the detour-phase CGH was adapted from Dr Williams J. Dallas of the University of Arizona [8].

A complex valued array of N x N size is generated, and it is centered so that the origin lies in the middle of the array. A simple character (in this case “$” sign) is drawn over the array such that the symbol is white whereas the background is black. The binary image will act as our object image which will be optically reconstructed using the hologram to regenerate the image.

Fig 1: Simple 2D Bitonal object

A uniform random phase is generated and the 2D complex valued function called the object function is generated.

$$\Phi = (\text{RAND} - 0.5)\,.2.\pi$$

(where RAND stands a random number between 0 and 1)

The resultant function is described as below

$$g\,[n, m] = |f\,[n, m]\,|e^{i\phi}$$

The Fast Fourier Transform routine from Scipy.fft module is applied on the function. The diagonal quadrants are swapped before and after the FFT since the module is written for single sided arrays and the point of symmetry of the image is kept at the principal focus line through the center of the image. To make it two sided the quadrant are swapped such that 1st quadrant is

swapped with the 3rd quadrant and vice versa. Similarly, 2nd quadrant is swapped with the 4th quadrant and vice versa.
The magnitude and the phase of the resultant Fourier transform are normalized by dividing the magnitude value with the maximum magnitude and the phase by 2π respectively.

The normalized magnitude and phase are then quantized into 8 levels such that a super-pixel for the Lohmann type I and III holograms consists of an 8 x 8 array. The magnitude has quantized levels from 0 to 8 and the phase has quantization levels ranging from [$-\pi$ to $3\pi/4$].

An error-diffused quantization is also introduced during the quantization process where the quantized complex value of a recent pixel is subtracted from the next pixel before quantization.

Now the bitonal apertures for each discrete value in the complex valued Fourier Transform function array of 64 X 64 size is created. The length of the aperture encodes the magnitude of the complex value at the pixel number and the phase is encoded by shifting the aperture from the cell center with possible values ranging from [-4 to +3] which corresponds to the phase quantized values of [$-\pi$ to $+3\pi/4$].

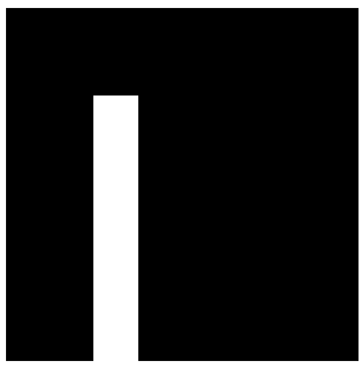


Fig 2: Lohmann Type I aperture

Lohmann Type III hologram was also generated by increasing the aperture width and splitting the aperture and wrapping the phase in a circular way around the bitonal cell.

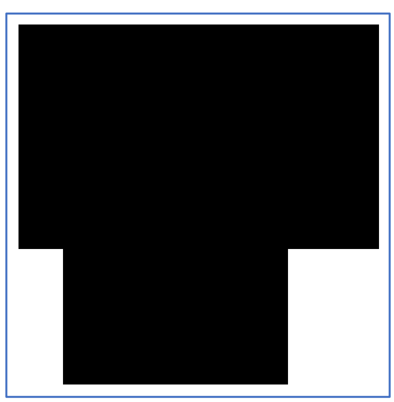


Fig 3: Wrapping of phase and splitting of the aperture for Lohmann Type III

ERROR DIFFUSION

Error diffusion quantization is a technique where the fidelity of the reconstruction is improved by subtracting quantization error of a recent pixel from the adjacent complex value before quantization. The quantization is then reapplied on the pixel and the process is repeated.

The algorithm is given as follows

Q[x] denotes quantization to the nearest step level.
q_F' denotes the quantized complex valued function of the Fourier transform signal
F' [n, m] is the error corrected Fourier Transform complex value at the particular pixel and F [n, m] is the normal Fourier transform.

```
Initiate e [ n, m] with 0 + i0

for each pixel [n, m] in array:
     F' [n, m] = F [n, m] - e [n', m']

     q_F' [n, m] = Q [F' [n, m]]

     e [n, m] = F' [n, m] – q_F' [ n, m]

where n' and m' are the pixel coordinates of the previous
iteration.
```

OPTICAL RECONSTRUCTION

A monochromatic laser light is intercepted by a converging lens which then illuminates the CGH pattern. The square of the amplitude of the resultant is given by the square of the Fourier transform of the encoded pattern.

The pattern generated because of light emerging from a converging lens creates a converging spherical wavefront which can be modeled as light with wavelength λ_o converging at distance z_1.

The final pattern on the observation plate is given by the equation

$$g[\,x;\lambda_o, z_1\,] = \left(f[x].\, exp\left[-i\,.\,\pi\frac{x^2}{\lambda o.z1}\right]\right) * exp\left[+i\,.\,\pi\frac{x^2}{\lambda o.z1}\right] = \mathrm{F}\left[\frac{x}{\lambda o.z1}\right]\ exp\left[+i\,.\,\pi\frac{x^2}{\lambda o.z1}\right]$$

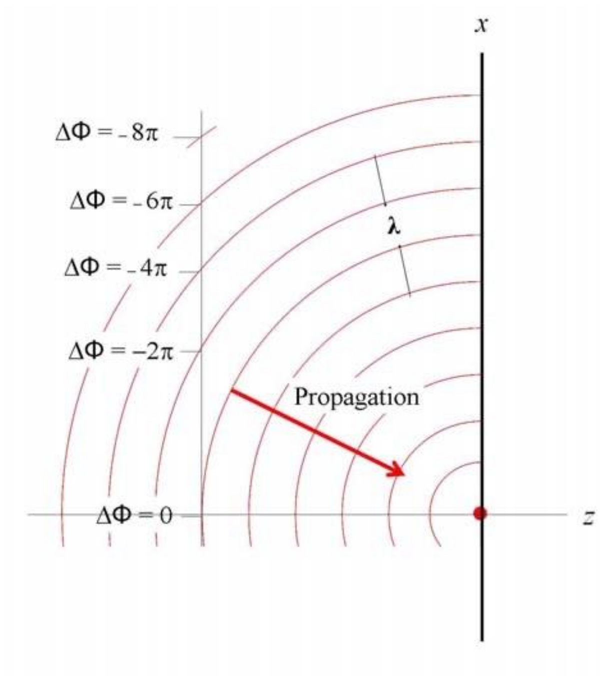


Fig 4 : Off axis phase is smaller than the on axis phase in a converging wave. Adapted from [2]

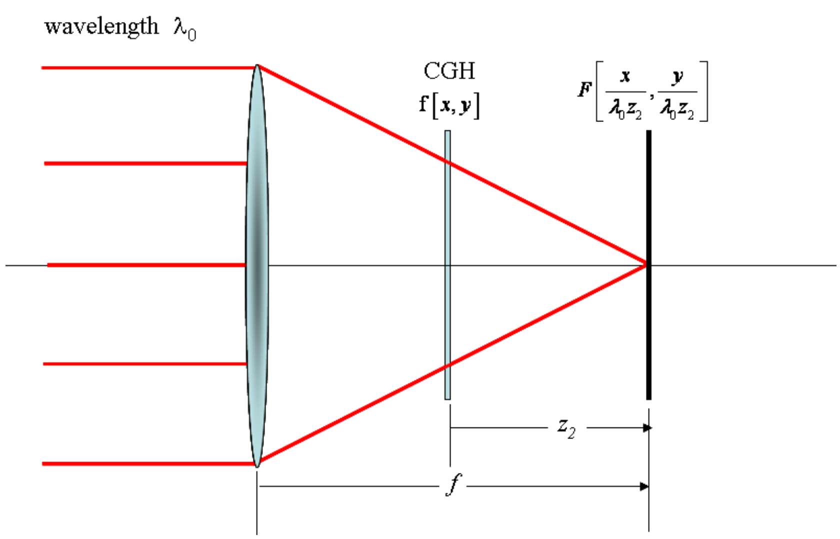
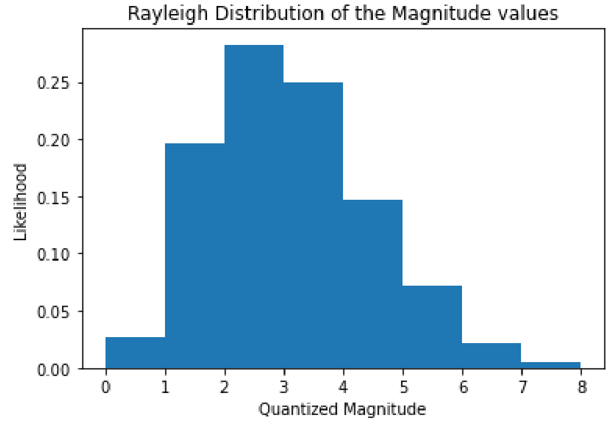
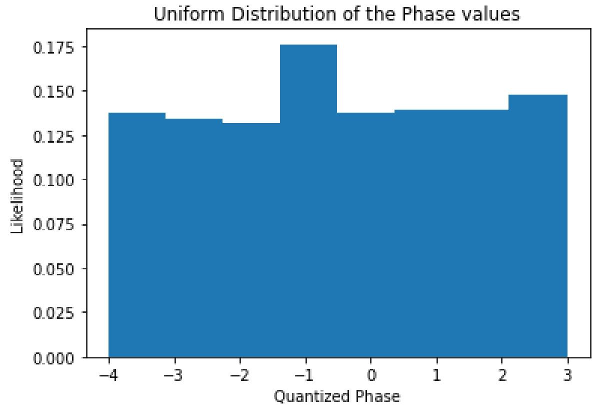


Fig 5: Reconstruction of CGH. Adapted from [2]

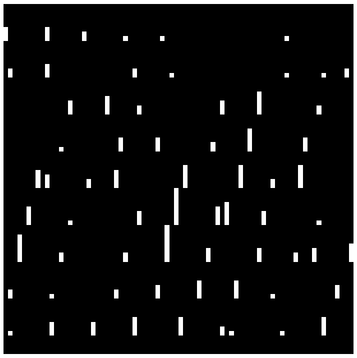
OBSERVATIONS



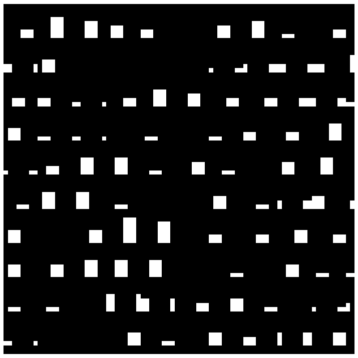


The quantized magnitude of the Complex valued Fourier transform of the image is flattened and a histogram is generated. It is clear from the distribution that it follows Rayleigh distribution since the phase generated is from a random uniform distribution.
The probability of level G is directly proportional to $|G|\exp[-|G|^2]$.

Also, the phase is randomly generated from a uniform distribution and that is reason behind the uniform distribution characteristics of the phase quantized histogram.

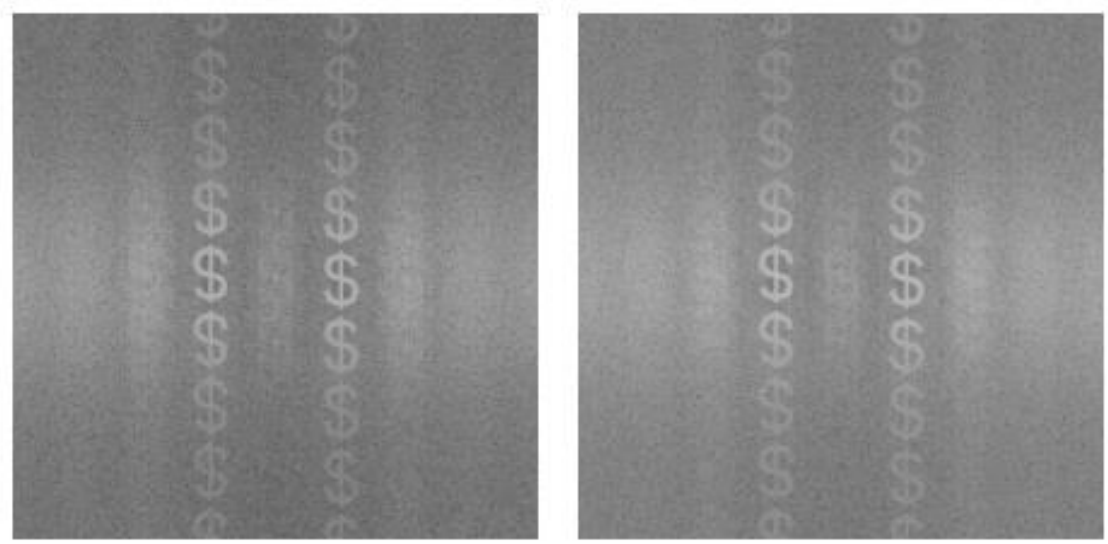


Portion of Lohmann Type I Hologram.



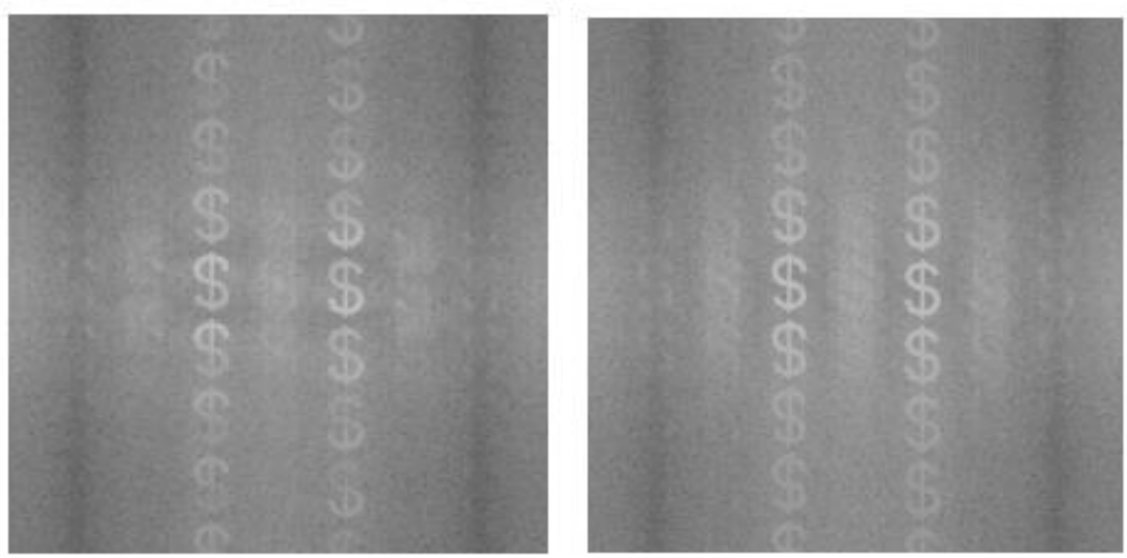
Portion of Lohmann Type III Hologram

A zoomed in region of both the hologram types are shown above for comparison. The Lohmann Type 1 has narrower slits and therefore allows less amount of incident light to pass through which decreases the brightness of the image and reduces the SNR ratio. Lohmann Type III hologram allows more signal to pass through thereby improving the SNR ratio. The difference is not apparent in the computer-generated reconstruction as physical ramifications of using lossy optical systems cannot be recreated with a mathematical operation like the Fourier Transform on the hologram image.



(a) (b)

Fig 6: Computer generated Reconstruction of Lohmann Type I Hologram by taking the Fourier Transform of the grating
and taking the log of the absolute value of the resultant transform image. a. (Error diffused) b. (error undiffused)



(a) (b)

Fig 7: Computer generated reconstruction Of Lohmann Type III Hologram by taking the Fourier Transform of the grating
and taking the log of the absolute value of the resultant transform image. a. (Error diffused) b. (Error undiffused)

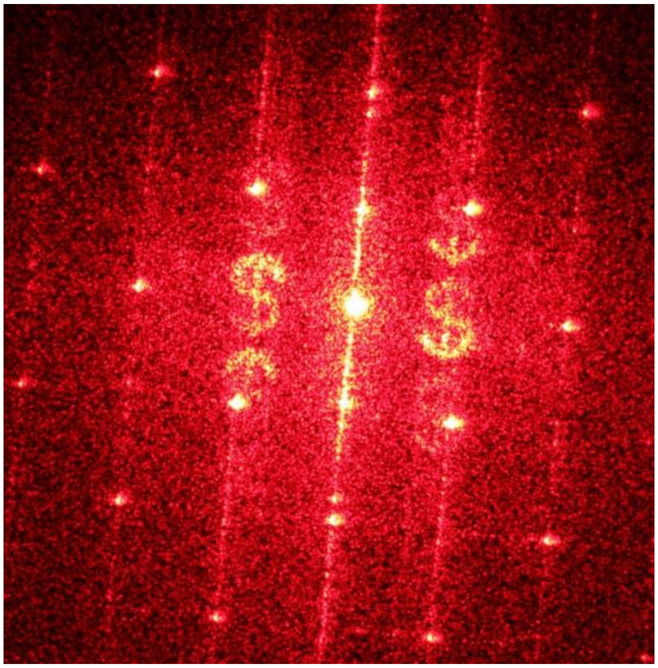


Fig 8: Optical Reconstruction of **Lohmann Type III** hologram recreating the original “$’ symbol. Error diffused and font size of symbol at 70.

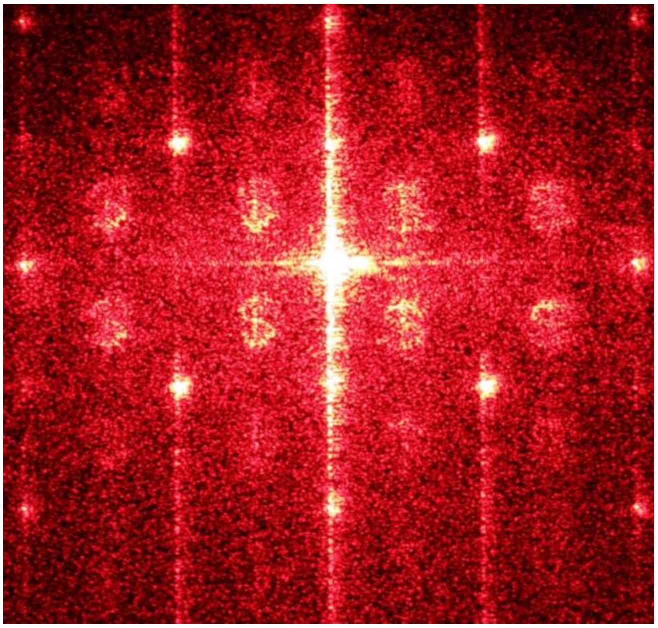


Fig 9: Reconstruction of **Lohmann Type III** hologram recreating the original ‘$’ symbol. Error diffusion not done and font size at 32

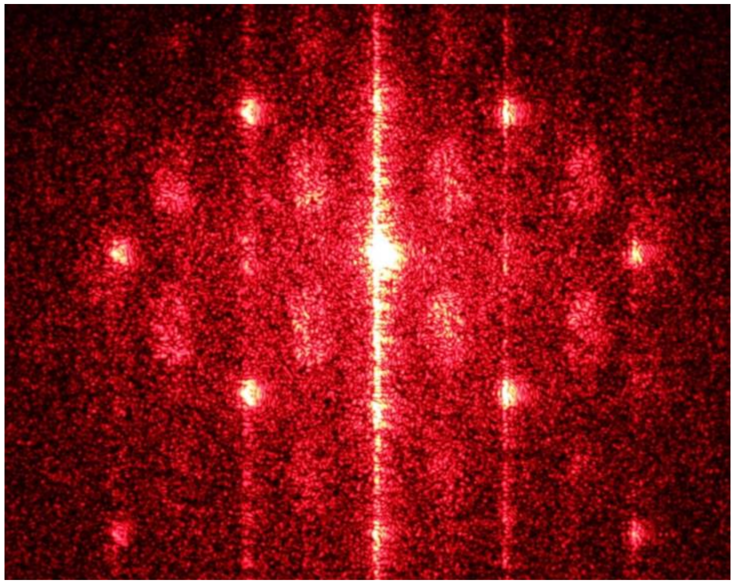


Fig 10: Reconstruction of **Lohmann Type I** hologram recreating the original '$' symbol. Error diffusion not done and font size at 32

It is clear from the actual experimental data of the reconstruction of the hologram that the Lohmann Type I has more blurred image as the signal to noise ratio is low. Also increasing the size of the symbol image allowed for more signal to pass through thereby giving a brighter reconstruction. Lohmann Type III hologram has a better SNR and better image quality demonstrated by brightness as well as sharpness. Error diffusion also helped in increasing the clarity of the reconstructed image.

DISCUSSION

One of the interesting observations from the optical reconstruction of the hologram is that there are multiple copies of the image formed in the observation plane. The reason behind is when we are taking the Discrete Fourier Transform or Fast Fourier Transform, we are effectively sampling the image at fixed intervals as well sampling the frequency spectrum. The resultant DFT or FFT is sampled and periodic. So, 8 X 8 image DFT would have $\Delta x = 1$ unit, N = 8X8 samples and $\Delta\xi = k/N$, $\Delta\eta = l/N$. The optical reconstruction would regenerate the scaled version of continuous Fourier transform of the function in the spatial domain. The final transform would effectively generate a periodic version of the sampled image. In the computer regeneration we did an inverse DFT which generates 512/64=8 images on each axis a and y. Therefore, the number of images is given as 8x8=64.

The plotting device that generates the mathematically computed variable into a physical cell-aperture print is limited by physical considerations such as smallest printable spot size, reproduction accuracy, toner bleeding etc. The minimum printable block size of electrophotographic system is 3x3 cell. Anything smaller than this will cause the adjacent toner dots to bleed into the adjacent pixel.

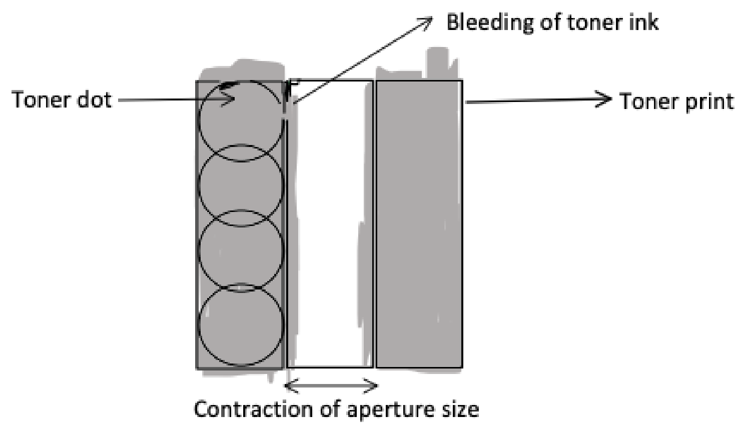


Fig 11: A diagram which show how toner bleeds into adjacent areas.

The contraction of aperture size would cause signal loss as more of the aperture area becomes opaque to incident light. This also leads to error in reproduction of complex magnitude of the Fourier transform function as the rectangular region can be filled from the top or the bottom which leads to magnitude error since the height of the aperture encodes magnitude.

There are multiple sources of error that can creep up in the computer-generated hologram. The application of Discrete Fourier Transform or Fast Fourier Transform generates aliasing errors. Also, the hologram is of limited size, therefore all the frequencies cannot be encoded and therefore leads to frequency truncation errors. One of the most obvious reasons behind reconstruction error is sampling of the complex valued function and quantization of the wavefronts. Also, the Lohmann type holograms use bitonal cells to encode the complex value information which can also lead to binarization error. As mentioned above plotter resolution also limits the accuracy with which the bitonal arrays can be printed which creates noise.
One of the factors that can also lead to errors is human handling of the holograms can lead to smearing of the hologram. Also, the optical system carries with it errors which can creep up to the result.

In a cell oriented CGH we also find overlap error. It happens when the aperture of a cell overlaps with the adjacent cell. A circular wrap around as shown in Lohmann Type III can mitigate the error but that leads to the problem of reproduction of areas lower than the minimum printable size which can generate error. There are various approximations involved in the creation of a reconstructed image of a hologram in the Fraunhofer diffraction region. Yet it was possible to reconstruct the dollar sign and therefore the project can be termed successful.

APPENDIX

Please find the code implementation in Python using SciPy module and NumPy.

```
#!/usr/bin/env python3
# -*- coding: utf-8 -*-
"""
Created on Sun Nov 14 21:21:42 2021

@author: arnabghosh
"""
import numpy as np
import matplotlib.pyplot as plt
from PIL import Image, ImageDraw,ImageFont
from scipy.fft import fft2,fftshift,fft,ifft
from scipy import stats
img_width = 64
img_height = 64
font_size = 70

'''Creating the image which will
be used for creating the hologram'''
def create_image(text,img_width,img_height,font_size):
    #Load the image and index over N/2 to N/2-1
    img = Image.new('1', (img_width, img_height), color='black')
    draw = ImageDraw.Draw(img)
    font                                                          =
ImageFont.truetype('/System/Library/Fonts/Supplemental/Arial.ttf
',font_size)
    w,h = draw.textsize(text,font=font)
    #Draw text
    draw.text(((img_width-w)/2,(img_height-
h)/2),text,font=font,fill=1)
    return(img)
#DFT or FFT of the complex array

'''Create the complex valued
array with random phase to
imitate a  diffused plane '''
string = '$'
```

```
img = create_image(string,img_width,img_height,font_size)


# img = Image.open('konoha.jpeg')
# img = img.convert('1')
# img = img.resize((64,64))
#Show Image
plt.figure(0)
plt.axis('off')
plt.imshow(img)
img = img.transpose(Image.TRANSPOSE)
img = img.rotate(0)
random_phase = np.random.rand(64,64)
random_phase = (random_phase-0.5)
random_phase = (random_phase)*2*np.pi
diffused_real = np.multiply(img,np.cos(random_phase))
diffused_im = np.multiply (img,np.sin(random_phase))
g = diffused_real + 1j*diffused_im

'''FFT of the 2d-image'''
F_g = fftshift(fft2((fftshift(g))))
max_mag = np.max(np.abs(F_g))
#print(max_mag)
F_real = np.real(F_g)
F_imag = np.imag(F_g)
'''Normalizing the maginitude and phase and Quantization'''
#print(F_imag)
#g_image = ifftshift(ifft2(g_fourier))
#print(np.max(np.abs(F_g)))'''
absF_g_norm = np.abs(F_g)/np.max(np.abs(F_g))
phaseF_g_norm = np.angle(F_g)/(2*np.pi)
#Fg_norm= np.multiply(absF_g_norm,np.exp(1j*np.angle(F_g)))
#print(Fg_norm)
'''Using a 8X8 cell and Quantizing the maginitude and phase'''
q_F_mag = np.floor(8*absF_g_norm+0.5)
q_F_phase = np.floor(8*phaseF_g_norm)
q_F = np.multiply(q_F_mag,np.exp(1j*q_F_phase*np.pi/4))
q_mag = np.zeros((img_width,img_height),dtype=float)
q_phase = np.zeros((img_width,img_height),dtype=float)
#error_real = absF_g_norm - q_F_mag
'''Error diffusion'''
e = 0+1j*0
for x in range(img_width):
    for y in range(img_height):
        real_new = F_real[x,y] - np.real(e)
```

```
        imag_new = F_imag[x,y] - np.imag(e)
        f = real_new + 1j*imag_new
        #quantize f
        q_mag[x,y] = np.floor(8*(np.abs(f)/max_mag)+0.5)
        q_phase[x,y] = np.floor((8*(np.angle(f))/(2*np.pi)))
        q_real            =((q_mag[x,y]/8)*max_mag)             *
np.cos(q_phase[x,y]*np.pi*0.25)
        q_imag            =((q_mag[x,y]/8)*max_mag)             *
np.sin(q_phase[x,y]*np.pi*0.25)
        q_f = q_real + 1j*q_imag
        # error diffusion
        e = f - q_f
        print(e)
print(np.max(q_phase))


'''Bitonal aperture'''

#aperture = np.zeros((8,8))

#final_image = np.zeros((512,512))
final_image = Image.new('1',(512,512),color='black')


for k in range(img_width):
    for l in range(img_height):
        aperture = Image.new('1',(8,8),color='black')

        pixmap = aperture.load()
        for y in range(int(q_F_mag[k,l])):
            #Lohmann Type 1
            pixmap[int(q_F_phase[k,l])+4,7-y] = 1

            #Lohmann Type 3

            # if(int(q_phase[k,l])==-4):
            #     pixmap[7,7-y]=1
            #     pixmap[1,7-y]=1
            #     pixmap[0,7-y]=1
            # elif(int(q_phase[k,l])==3):
            #     pixmap[0,7-y]=1
            #     pixmap[6,7-y]=1
            #     pixmap[7,7-y]=1
            # else:
            #     pixmap[int(q_phase[k,l])+4,(7-y)] = 1
            #     pixmap[int(q_phase[k,l])+3,(7-y)] = 1
            #     pixmap[int(q_phase[k,l])+5,(7-y)] = 1
```

```
        # plt.axis("off")
        # plt.imshow(aperture)
        # plt.show()
        # break
        final_image.paste(aperture,(8*k,8*l))


'''Finding out the hologram works or not'''
inverse = np.log(np.abs((fftshift(fft2(fftshift(final_image))))))


"Plotting"

plt.imsave('hologram_dollar1.png',final_image,cmap='gray',dpi   =
600)
#Plot the image
plt.figure(1)
plt.axis('off')
#plt.hist(q_F_mag)
plt.imshow(inverse,cmap='gray')

'''Plotting the Magnitude distribution'''
plt.figure(2)

plt.hist(q_F_mag.flatten(),density = True , bins = 8)
plt.xlabel("Quantized Magnitude")
plt.ylabel("Likelihood")
plt.title("Rayleigh Distribution of the Magnitude values")
plt.show()

'''Plotting the Uniform Phase distribution'''
plt.figure(3)

plt.hist(q_F_phase.flatten(),density = True , bins = 8)
plt.xlabel("Quantized Phase")
plt.ylabel("Likelihood")
plt.title("Uniform Distribution of the Phase values")
plt.show()
```

REFERENCES

1. A.W. Lohmann and D.P. Paris, Appl.Opt., 6, 1739, 1967.
2. R.L Easton, Jr., Computer Generated Holography lab write up.
3. R.L. Easton, Jr., R. Eschbach, and R. Nagarajan, Error diffusion in cell-oriented holograms to compensate for printing constraints, Journal of Modern Optics 43(6), 1219-1236, 1996.

4. R.L. Easton, Jr., §23 in Fourier Methods in Imaging, John Wiley & Sons, 2010.
5. Reynolds, G.O., J.B. DeVelis, G.B. Parrent, and B.J. Thompson, The New Physical Optics
6. Notebook, SPIE, 1989.
7. Iizuka, K. Engineering Optics, Springer-Verlag, 1984
8. Dallas, W.J., “Computer-Generated Holograms”, §6 in The Computer in Optical Research, B.R. Frieden, Springer-Verlag, 1980.
9. Collier, Robert J., C.B. Burckhardt, and L.H. Lin, Optical Holography, Academic Press, 1971.