# Grad Lab : Fraunhofer Computer generated hologram

Snehal Padhye

December 10, 2018

## 1 Abstract

The primary goal of this laboratory is to make holograms in the Fraunhofer diffraction region (far field), so that the diffracted amplitude is the Fourier transform of the object. The observation of diffracted light is considered at a plane yet more distant from the source than in the Fresnel approximation so that the expanding spherical wavefronts may be accurately modeled as planar. The mathematical form can be written as follows:

$$f[x, y; 0]exp^{(+i*\pi*(x^2+y^2))/\lambda_0*z1}$$

(1)

where the coordinates [x,y] specify location on observation plane. In gist, we are evaluating the Fourier transform of the rendered approximate Fourier transform, and thus producing an approximation of the original 2-D planar object.

## 2 Procedures

1. Create an N N 2-D complex-valued array filled with zeros. The dimension of the array should be a power of two, so I took N = 64.
2. Define a 2-D bitonal object over the N-D array.

3. Use a uniformly distributed random-number generator to assign a random phase  to each pixel in the array such that,

$$phase = (RAND - 0.5) * 2\pi \tag{2}$$

4. Compute either the discrete Fourier transform (DFT) or FFT of the 2-D centered complex-valued N -D array with the random phase. This produces a complex-valued array of two indices in the frequency domain.
5. Compute the magnitude and phase of the FFT of the array. 6. Normalize the magnitudes at each pixel by dividing by the maximum of all the magnitudes.
7. Select a cell size (8x8) for the hologram; this is the size of the bitonal cell that will be used to approximate the complex amplitude (magnitude and phase) of each pixel in the Fourier transform array.
8. Quantize the normalized magnitudes so that the largest value is the linear dimension of the cell array. For 8x8 cell, multiply the normalized magnitude by 8 and then round to the nearest whole number.
9. Quantize the phase at each pixel by dividing the calculated angle by $2\pi$ radians, multiplying by the linear dimension (8) of the cell, and rounding to the nearest whole number  this will produce phase numbers in the interval [4, +3].
10. Make bitonal apertures to make the 8  8 cell for each pixel in the 64  64 DFT array and display them as corresponding to the quantized magnitude and quantized phase. The vertical length of the aperture in each cell is the quantized magnitude of that pixel in the fourier transform array, while the horizontal position of the aperture is the quantized phase of the pixel of the fourier transform array.
11. Increase the width of the cells to three pixels, thus making a Lohmann Type-III hologram. The slot passing the light is replicated in the columns to either side. The resulting hologram passes more light to the reconstruction, but the average phase of that cell is still maintained. One of the two additional columns must wrap around to the other side of the cell.
12. After creating the bitonal CGH, we can check on its success by simulating the optical reconstruction in the Fraunhofer diffraction region, which requires evaluation of the squared magnitude of the 2-D Fourier transform of the (N = 512) bitonal CGH array. The display should show multiple replicas of the original object on either side of the bright DC term at the center of the reconstruction.

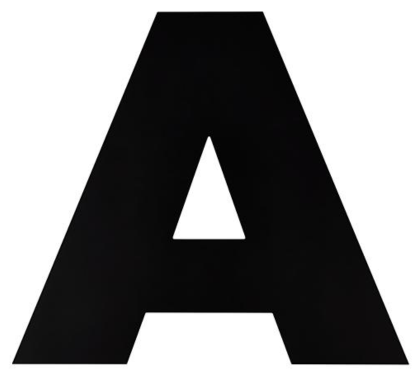

Figure 1: Original Image

13. Print out the bitonal array with an electrophotographic laser or inkjet printer on overhead transparency film at a small size.
14. Display the hologram by illuminating with a He:Ne laser and view in the Fraunhofer diffraction region.

## 3 Results

I took an image of aplhabet 'A' shown in firgure 1.

The computer generated hologram is shown in figure 2.

Simulation of reconstruction is shown in figure 3.

Reconstruction through laser is shown in figure 4 and 5.

## 4 Discussion

Explain the number of replicas in the reconstructed image of the hologram. The replica images in the reconstruction are due to the 88 cell. The inverted replicas are due to fact that the hologram is real valued, so its Fourier transform is symmetric (even).The reconstructed image is a scaled replica of the autocorrelation of the source function that has been scaled in both position and brightness.

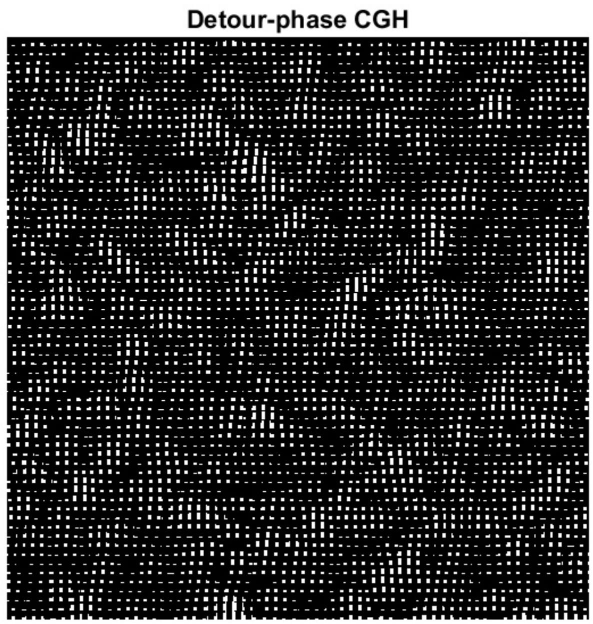


Figure 2: Computer Generated Hologram

2. Consider the limitations of the laser printer. The rectangular apertures of the hologram are approximated by spots of toner that are fused onto the transparency film by heat. As the scale of the rendered pattern is reduced, the spreading out of the toner spots ensures that the desired rectangular patterns will not be printed. What does this mean for the reconstruction of the hologram? A sketch of the effect of toner spread on the rendered pattern may be helpful.

which induces random variations in thickness of the transparency, which in turn produces random variations in the phase of the transmitted light. 3. Your reconstruction is probably pretty noisy. What are the possible mechanisms that generate the noise?The 2-D Lohmann hologram of an Archimedean spiral in a 32  32 array using an 8  8 cell and a random phase is shown in Figure 23.32, along with its optical and digital reconstructions. The cell size ensures that the digital reconstruction is nearly periodic with eight orders in the horizontal direction and eight replicas in the vertical direction. The reconstruction.jpg

Figure 3: Reconstructed Image

optical reconstruction was created by placing a laser print of the hologram on transparency material in an expanded beam from an He:Ne laser and recording the irradiance on a CCD sensor placed at the focus of the beam in the Fraunhofer diffraction region. The angular separation between the orders is determined by the pixel pitch in the rendered hologram via the scaling theorem of the Fourier transform; the smaller the pixel pitch, the larger the angle between orders. The dots in the vertical line in the center are reconstructions of undiffracted light through the hologram (the DC term); these will be significant during our discussion of the optical matched filter in the next section. Note that these reconstructions are not exactly Dirac delta functions, but also include other nearby frequencies due to the nonlinear quantization. The primary reconstructions of the object are located at the orders 1. The digital reconstruction in Figure 23.32d is rendered as the logarithm of the

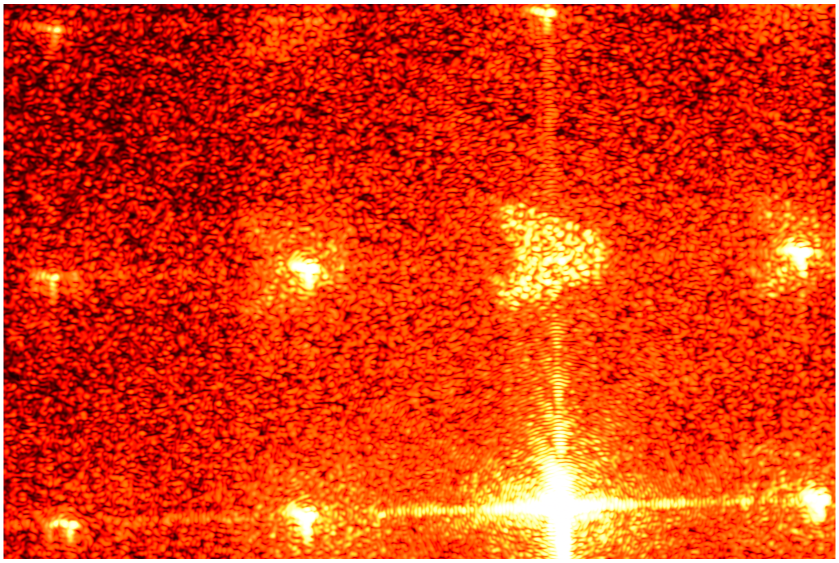

Figure 4: Reconstructed Image using laser

squared magnitude to simulate the visual appearance due to the logarithmic response of the human visual system; the similarity between the optical and digital reconstructions is evident. The random variations in brightness of spiral and of background, called speckle, are primarily due to the additive random phase. The eight replicas of the reconstructions along the vertical direction in the discrete reconstruction are due to cells 8 pixels tall in the CGH. Note that an approximate replica of the squared magnitude of the object —f [x, y]—2 is reconstructed at order 1, while the reconstruction at order +1All quantizations are nonlinear and produce errors; quantization error in CGHs produces error in the reconstructions. In a Fraunhofer hologram, the reconstruction by optical Fourier transformation spreads this quantization error over the entire space domain, where it appears as noise.

# 5 Code

```
I= imread('C:\Users\sneha\Documents\code\grad-lab-fourier\a.jpg');
I = imresize(I,[32 32]);
I = imbinarize(I);
I = imcomplement(I);
```

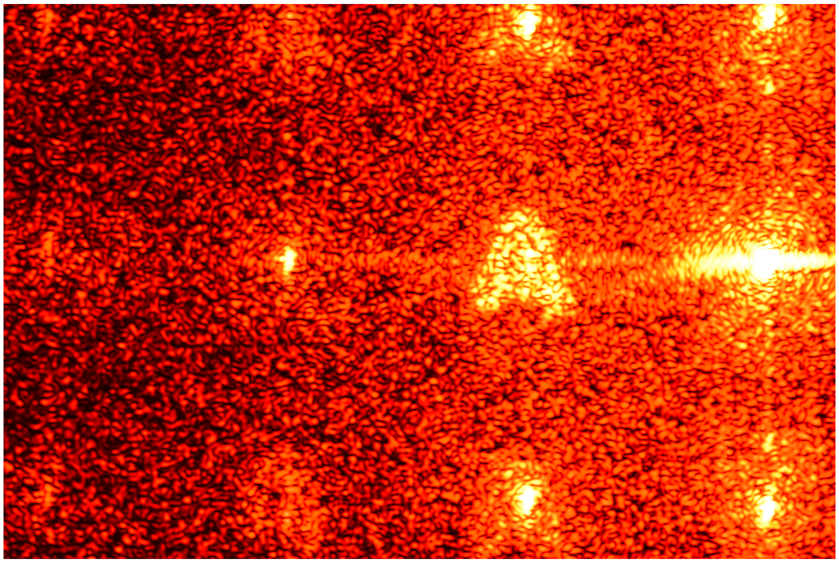

Figure 5: Reconstructed Image using laser

```
I = padarray(I,[16 16],0,'both');
I = double(I(:,:,2));
figure;
imshow(abs(I));
title('Original object')
PH = rand([64,64]);
%I = I.*exp(2i*pi*PH);% add a random phase to the object
I = I.*exp(2i*pi*(PH-(0.5)));
FTS= fftshift(fft2(ifftshift(I)));
A=abs(FTS);
figure;
imshow(mat2gray(A));
title('Object spectrum')
A=(A./max(max(A)))*8;
A=round(A);% The amplitude is divided into 8 levels
B=angle((FTS));
B=(((B./(2*pi))*8));
B=floor(B);% The amplitude is divided into 8 levels
H=zeros(16);
for m = 1:64
```

```
    for n = 1:64
        P =zeros(8);
        a = A(m,n);
        b = B(m,n);
        if b == -4
            P(9-a:8,8:8)=1;
            P(9-a:8,4+b+1:b+4+2)=1;
        elseif b == 3
            P(9-a:8,1:1)=1;
            P(9-a:8,4+b:b+4+1)=1;
        else
            P(9-a:8,4+b:b+4+2)=1;
            %P(9-a:a,4+b+1:b+4+1)=1;
        end
        if a == 0
            P =zeros(8);
        end
        %disp(P)
        H(8*(m-1)+1:8*(m-1)+8,8*(n-1)+1:8*(n-1)+8)=P;
        %disp(H)
    end
end
figure;
imshow(H)
title('Detour-phase CGH')
%Reconstruction (FFT)
R=fftshift(ifft2(ifftshift(H)));
figure;
imwrite(H,'hologram_snehal.tif');
%imshow(10.*mat2gray(abs(R));
imshow(log(abs(R)),[]);
title('Reconstructed image')
```