Fall 2021

California State University, Northridge

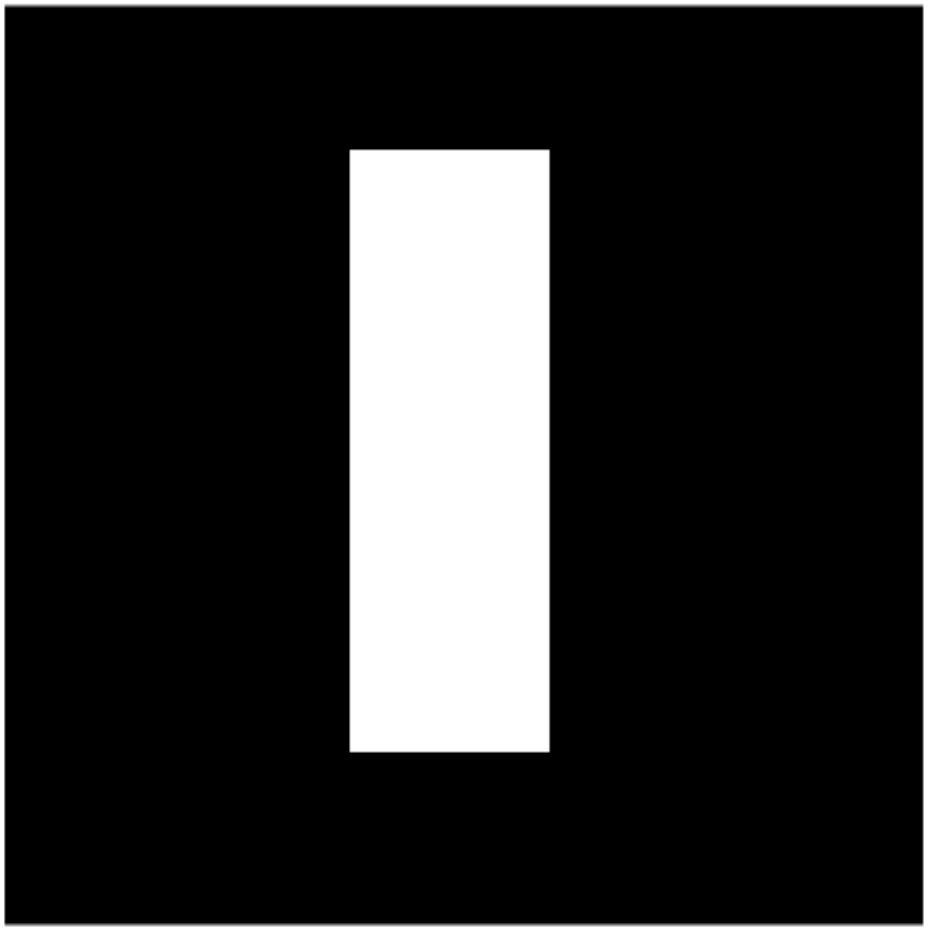
Department of Electrical and Computer Engineering

Computer Assignment 3
October 20, 2021

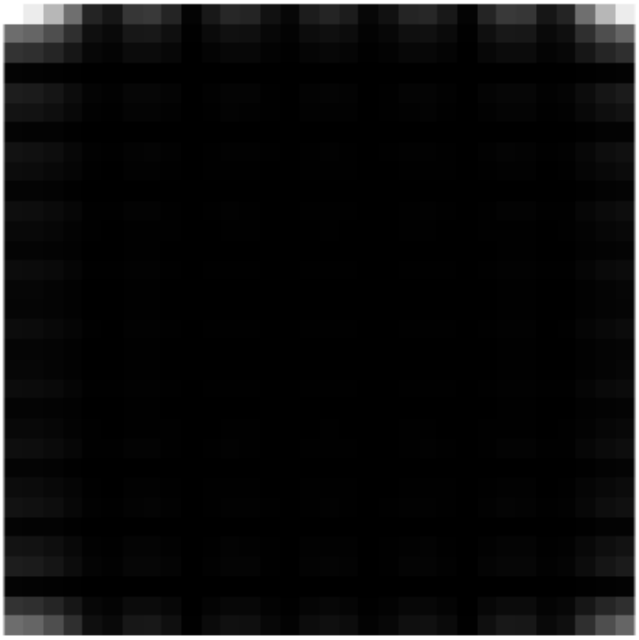
ECE 551
Professor: Shahnam Mirzaei

Written By: Morris Blaustein

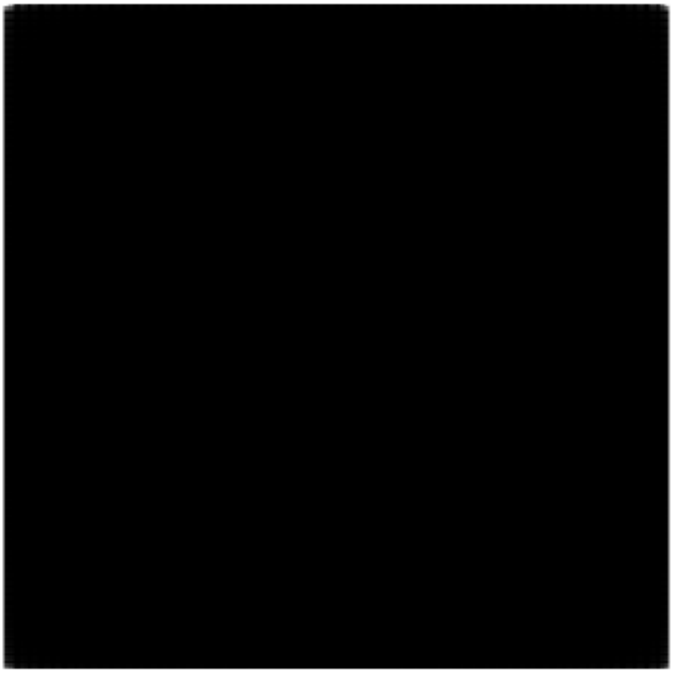
Experiment 1 Results



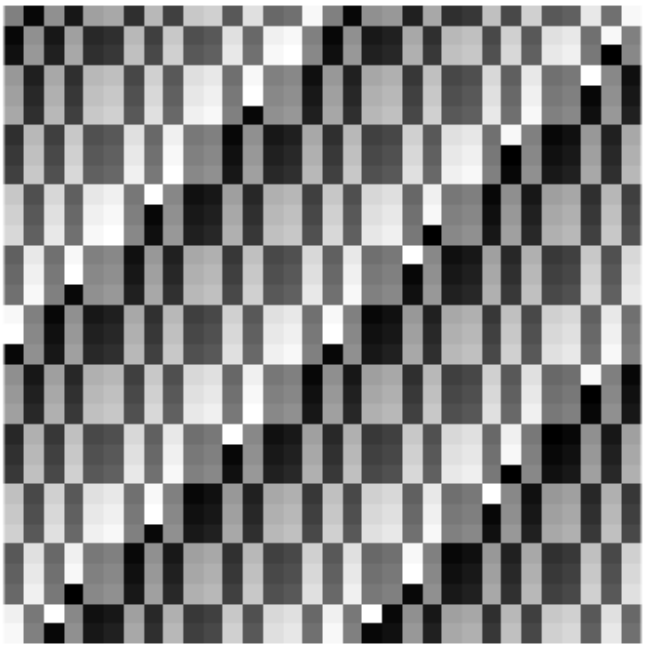
Rectangle



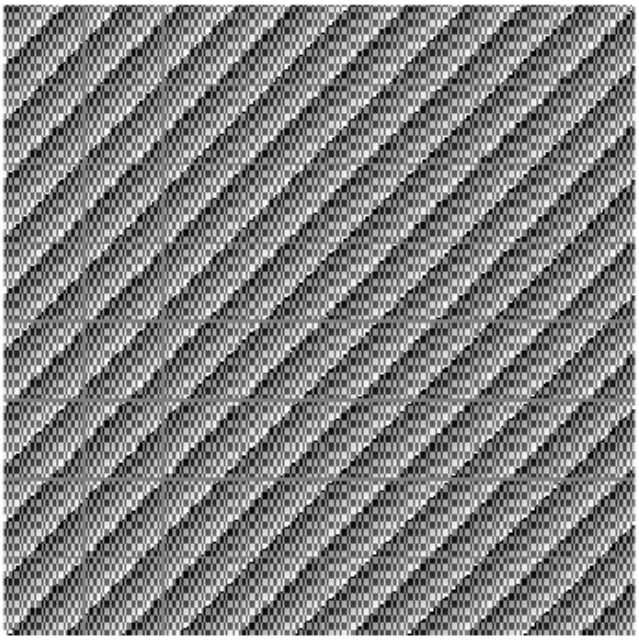
DFT Magnitude



DFT Magnitude, Enlarged Image



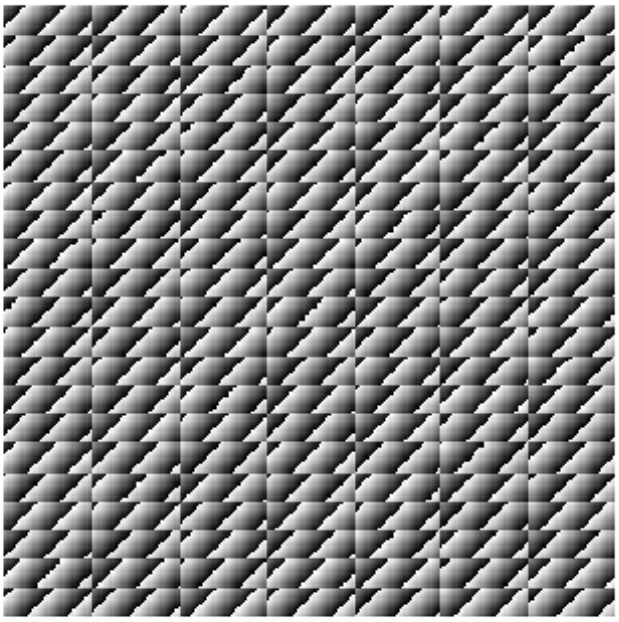
DFT Phase



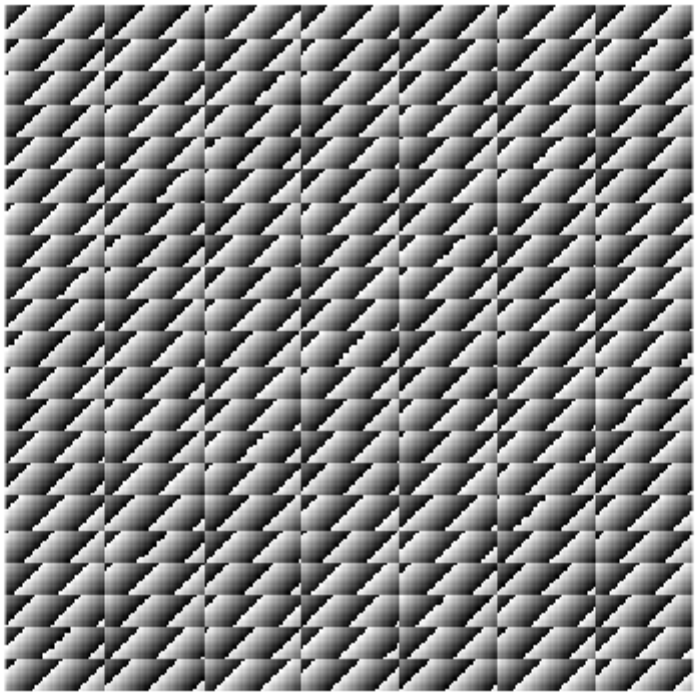
DFT Phase, Enlarged Image

DFT Magnitude with 256 Padding

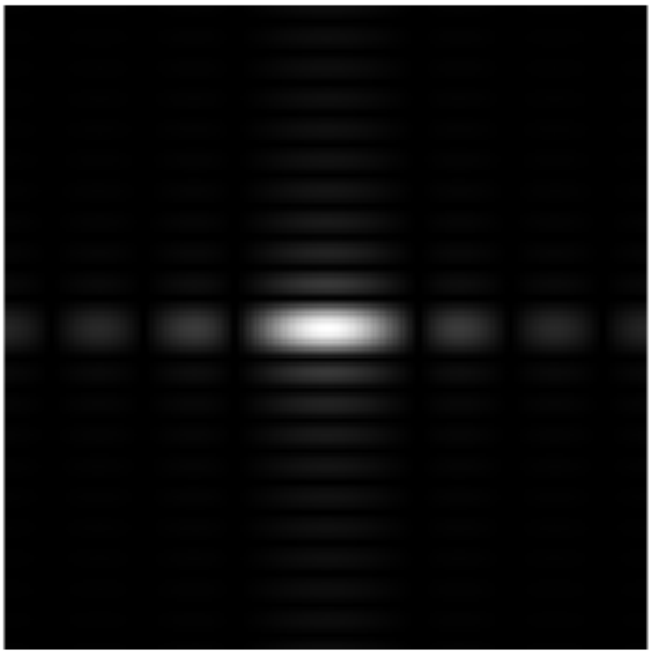
DFT Magnitude with 256 Padding, Enlarged Image



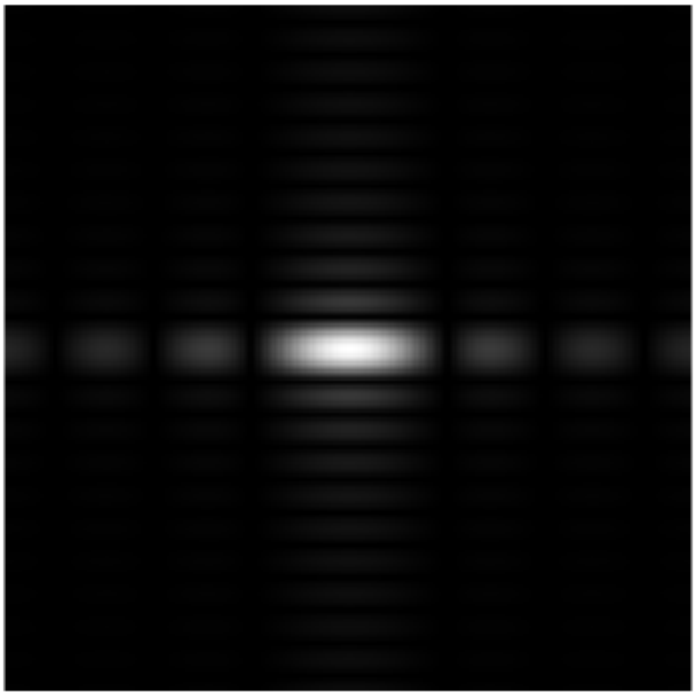
DFT Phase with 256 Padding



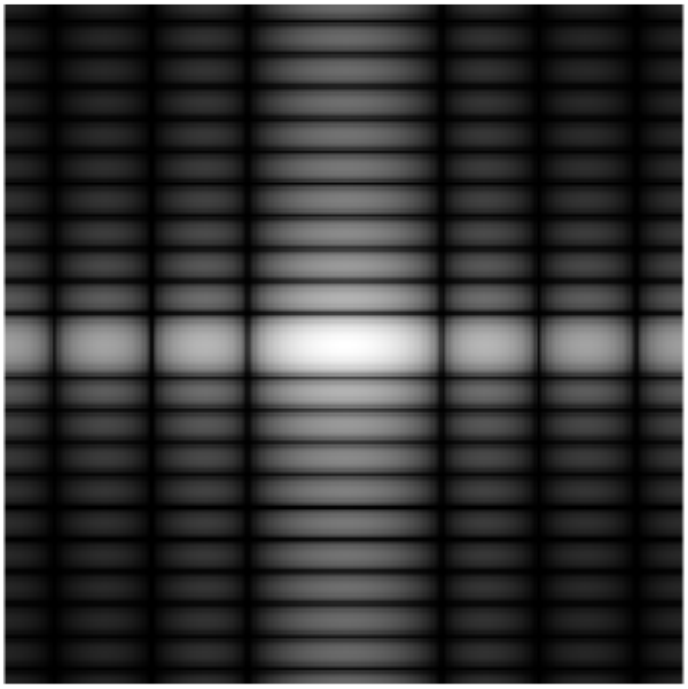
DFT Phase with 256 Padding, Enlarged Image



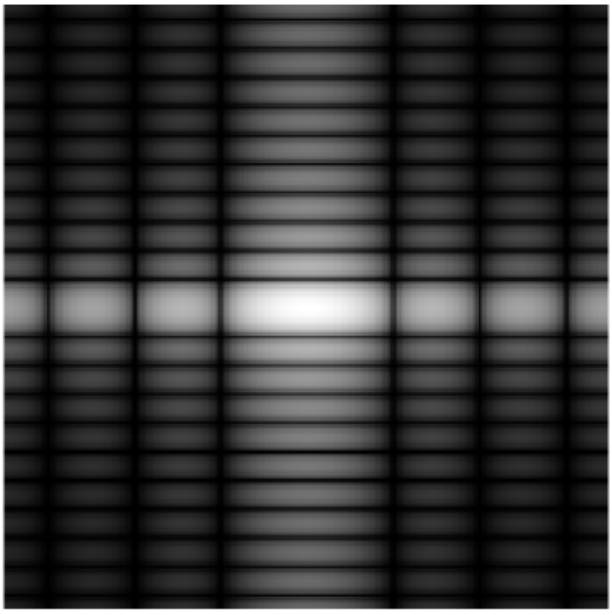
DFT Magnitude with 256 Padding and Log Transformation



DFT Magnitude with 256 Padding and Log Transformation, Enlarged Image

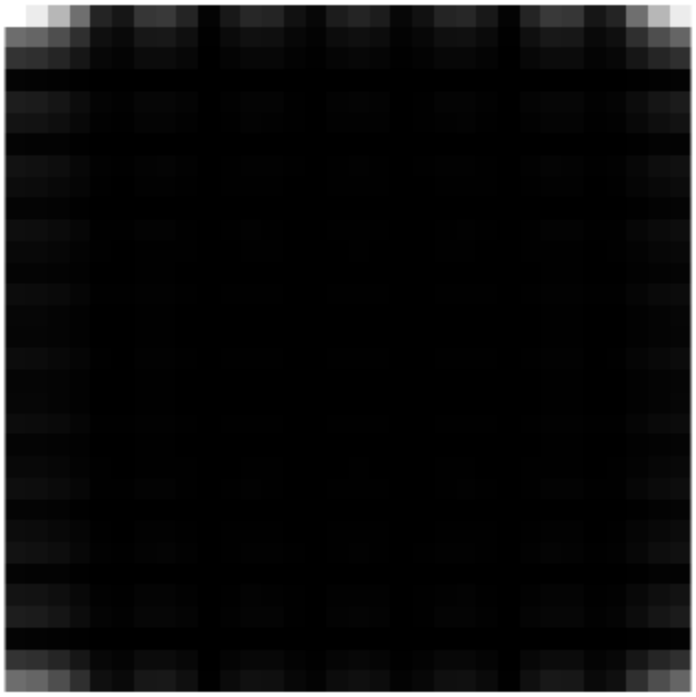


DFT Phase with 256 Padding and Log Transformation

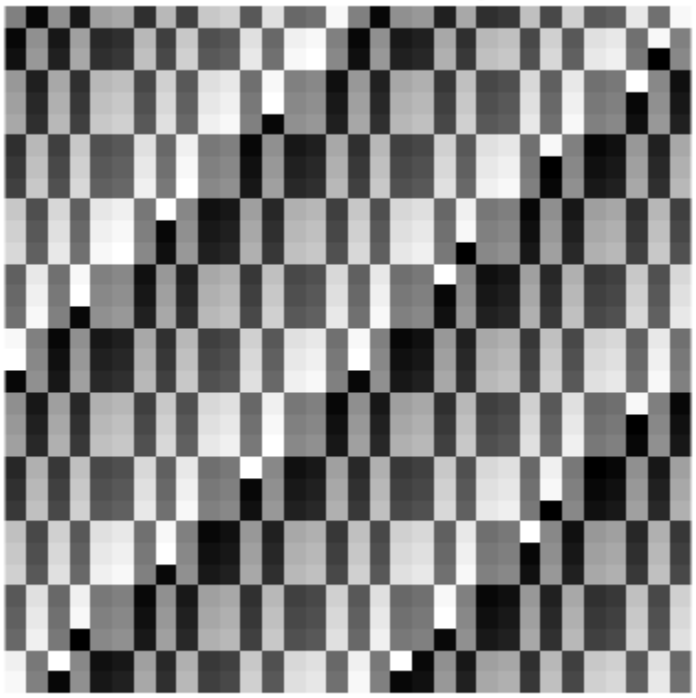


DFT Phase with 256 Padding and Log Transformation, Enlarged Image

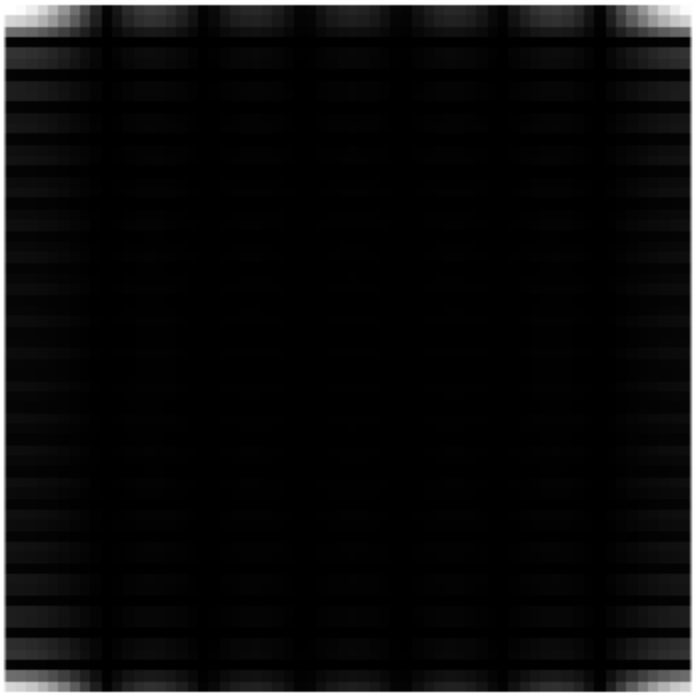
Magnitude of 32 padding



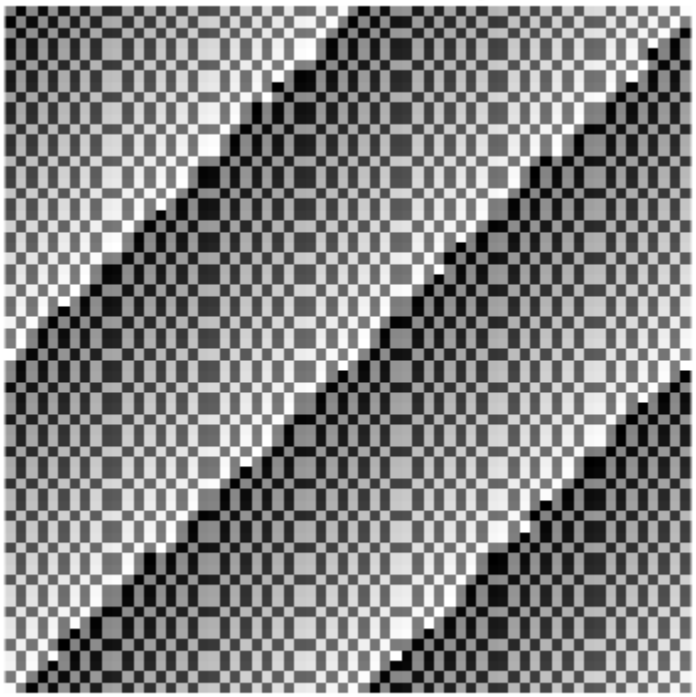
Phase of 32 padding



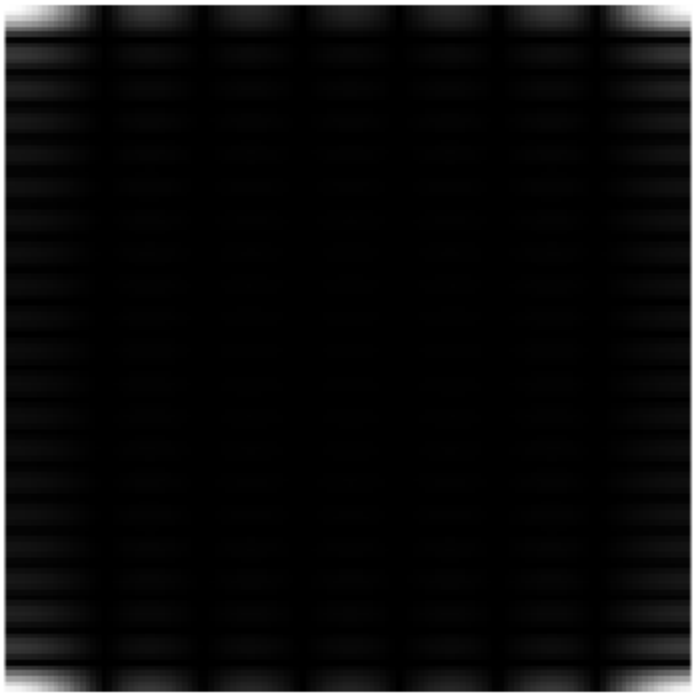
Magnitude of 64 padding



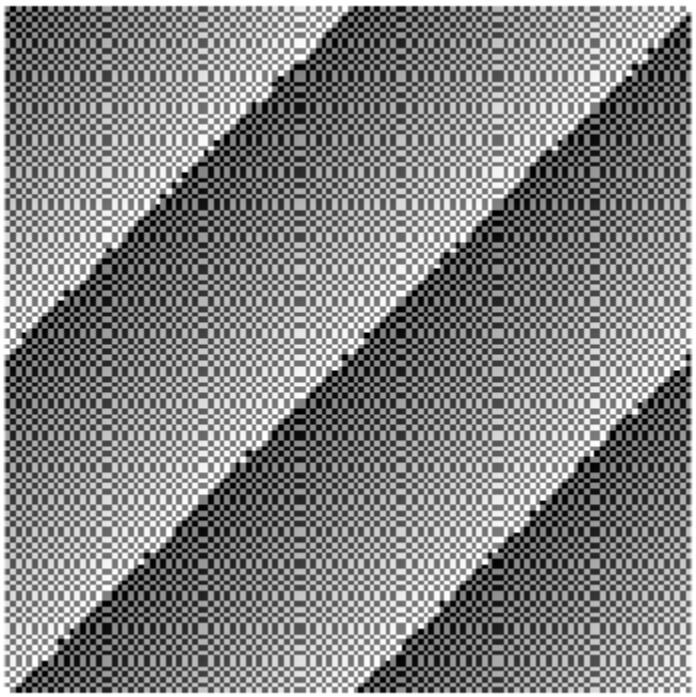
Phase of 64 padding



Magnitude of 128 padding

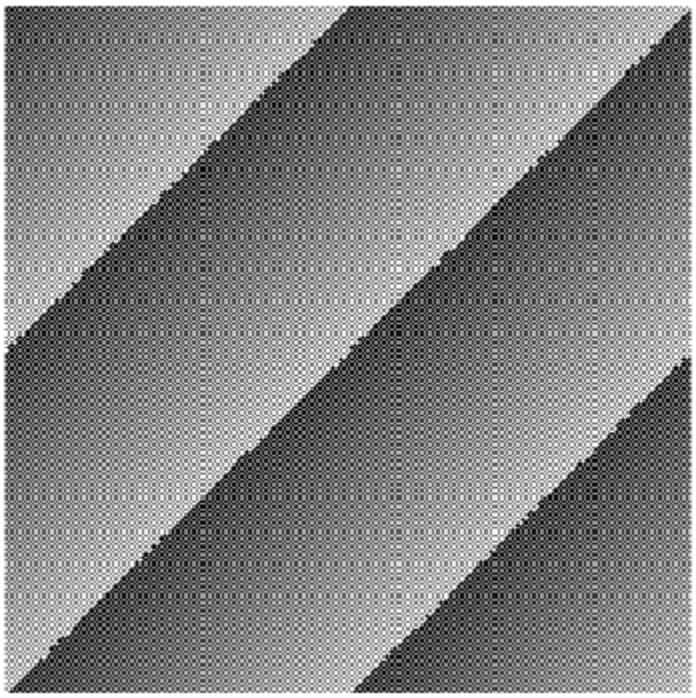


Phase of 128 padding



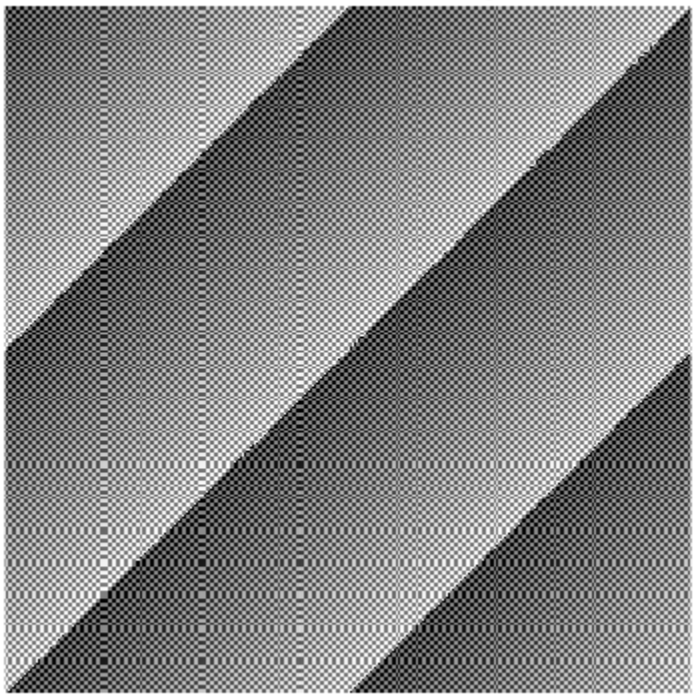
Magnitude of 256 padding

Phase of 256 padding



Magnitude of 512 padding

Phase of 512 padding



Experiment 1 Code

```
rect=zeros(32,32);
rect(6:26,13:19)=1;

%Question 11
rect_enl = zeros(256,256);
for i = 1:256
   for j = 1:256
      x = floor(i/8);
      y = floor(j/8);
      if x < 32
         x = x+1;
      end
      if y < 32
         y = y+1;
      end
      rect_enl(i,j) = rect(x,y);
   end
end

%Questions 1-8,11
imshow(rect_enl,'InitialMagnification', 'fit')
rect_fft = fft2(rect_enl);
rect_fft_mag = abs(rect_fft);
rect_fft_phase = angle(rect_fft);
figure
imshow(rect_fft_mag,[],'InitialMagnification', 'fit')
figure
 imshow(rect_fft_phase,[],'InitialMagnification', 'fit')
rect_fft_256 = fft2(rect,256,256);
rect_fft_mag_256 = abs(rect_fft_256);
rect_fft_phase_256 = angle(rect_fft_256);
figure
imshow(rect_fft_mag_256,[],'InitialMagnification', 'fit')
figure
imshow(rect_fft_phase_256,[],'InitialMagnification', 'fit')
rect_fft_mag_256_shift = fftshift(rect_fft_mag_256);
figure
imshow(rect_fft_mag_256_shift,[],'InitialMagnification', 'fit')
rect_fft_mag_256_shift_log = log(1+rect_fft_mag_256_shift);
figure
imshow(rect_fft_mag_256_shift_log,[],'InitialMagnification', 'fit')

%Question 10
```

```
for i= 5:9
   pad_zeros_fft(rect,i);
end

%Question 9
function padded_img = pad_zeros_fft(img,power)
size_img = size(img);
padded_img = zeros(2^power,2^power);
start_x = (2^power-size_img(1))/2;
start_y = (2^power-size_img(2))/2;
for i = 1:(2^power+size_img(1))/2
   for j = 1:(2^power+size_img(2))/2
      if ( i > start_x & j > start_y )
         padded_img(i,j) = img(i-start_x,j-start_y);
      end
   end
end
padded_img_fft = fft2(padded_img);
padded_img_fft_mag = abs(padded_img_fft);
padded_img_fft_phase = angle(padded_img_fft);
figure
imshow(padded_img_fft_mag,[],'InitialMagnification', 'fit')
title("Magnitude of " + 2^power + " padding")
figure
imshow(padded_img_fft_phase,[],'InitialMagnification', 'fit')
title("Phase of " + 2^power + " padding")
% imwrite(padded_img_fft_mag,[],"magnitude" + 2^power + ".png");
% imwrite(padded_img_fft_phase,[],"phase" + 2^power + ".png");
end
```

Experiment 1 Conclusion

I do not know if the Fourier spectrum I plotted in experiment 1 was correct. I showed the magnitude and phase of FFT directly after their calculation. I did not do any shifting. The magnitude and phase of the FFT of the 32x32 image is different than when it is enlarged to 256x256.The FFTs of the enlarged image 256x256 and the 256 zero padded image were the same. The magnitude of FFTs looks the same regardless of the amount of zero padding that is used. The phase of the FFTs looks more detailed when the amount of padding increases.

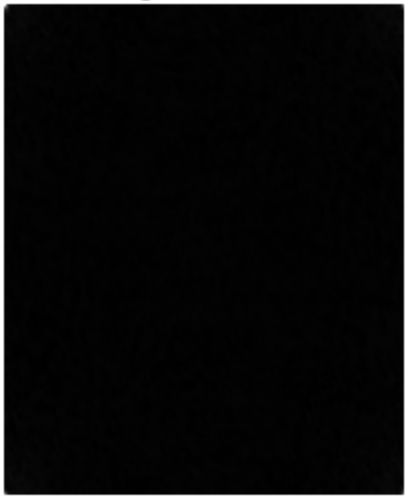
Experiment 2 Results

IFFT, Phase of Zebra

IFFT, Phase of Cheetah

IFFT, Magnitude of Zebra

IFFT, Magnitude of Cheetah



IFFT, Magnitude and Phase of Cheetah

IFFT, Magnitude and Phase of Zebra

IFFT, Magnitude of Z Cheetah

IFFT, Magnitude of Cheetah and Phase of Zebra

Experiment 2 Code

```
cheetah = rgb2gray(imread('cheetah.jpg'));
zebra = rgb2gray(imread('zebra.jpg'));
cheetah_fft = fft2(cheetah);
zebra_fft = fft2(zebra);
cheetah_phase = angle(cheetah_fft);
zebra_phase = angle(zebra_fft);
cheetah_mag = abs(cheetah_fft);
zebra_mag = abs(zebra_fft);

%Part a
zebra_ifft =ifft2(exp(i*zebra_phase));
figure
imshow(zebra_ifft,[])
title('IFFT, Phase of Zebra');

%Part b
cheetah_ifft =ifft2(exp(i*cheetah_phase));
figure
imshow(cheetah_ifft,[])
title('IFFT, Phase of Cheetah');

%Part c
zebra_ifft =ifft2(zebra_mag);
figure
imshow(zebra_ifft,[])
title('IFFT, Magnitude of Zebra');

%Part d
cheetah_ifft =ifft2(cheetah_mag);
figure
imshow(cheetah_ifft,[])
title('IFFT, Magnitude of Cheetah');

%Part e
cheetah_ifft =ifft2(cheetah_mag.*exp(i*cheetah_phase));
figure
imshow(cheetah_ifft,[])
title('IFFT, Magnitude and Phase of Cheetah');

%Part f
zebra_ifft =ifft2(zebra_mag.*exp(i*zebra_phase));
figure
imshow(zebra_ifft,[])
```

```
title('IFFT, Magnitude and Phase of Zebra');

%part g
mzebra_pcheetah_ifft =ifft2(zebra_mag.*exp(i*cheetah_phase));
figure
imshow(mzebra_pcheetah_ifft,[])
title('IFFT, Magnitude of Zebra and Phase of Cheetah');

%part h
mcheetah_pzebra_ifft =ifft2(cheetah_mag.*exp(i*zebra_phase));
figure
imshow(mcheetah_pzebra_ifft,[])
title('IFFT, Magnitude of Cheetah and Phase of Zebra');
```

Experiment 2 Conclusion

The phase of the FFT of the two images showed some detail of what the image looks like in the IFFT. The magnitude of the FFT does not have information when it is not associated with the phase. The IFFT produced the original image when both the magnitude and phase are used in the IFFT. When the magnitude and phase come from different images, the phase will mostly define what the image will look like. It retains important features from the image that it came from.