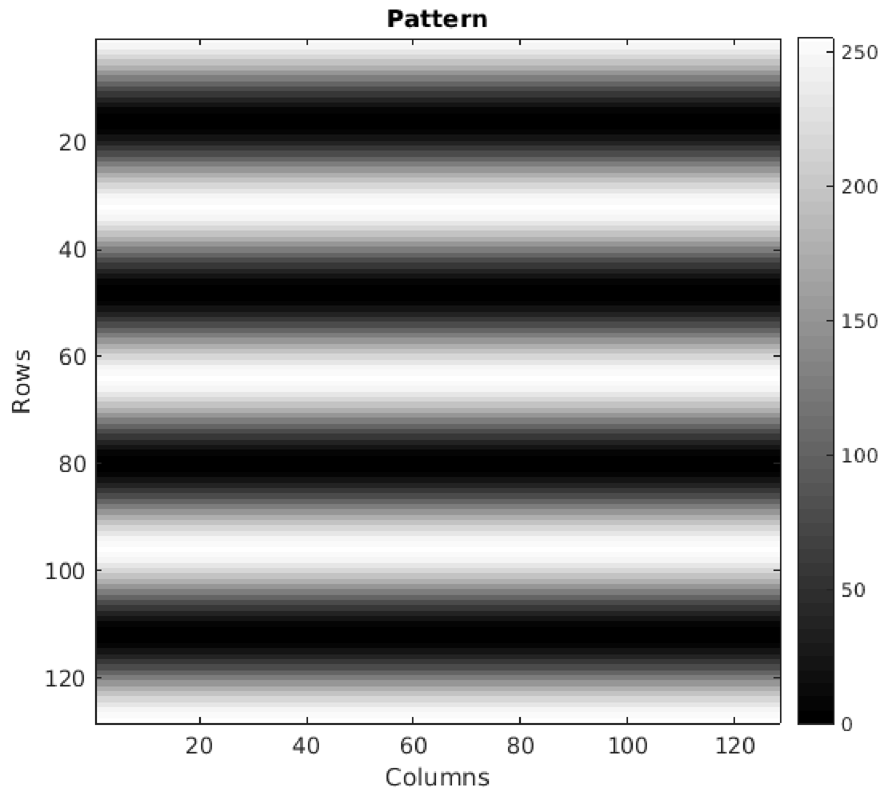
```
% BME671L Lab #7: 2D Fourier Transform, fft2, fftshift, linspace

% Your name: Ravitashaw Bathla

% DISPLAY ALL IMAGES USING COLORMAP gray(256). SET THE AXIS SO THE
% IMAGES ARE NOT DISTORTED (e.g. square or image)

close all, clear all
```

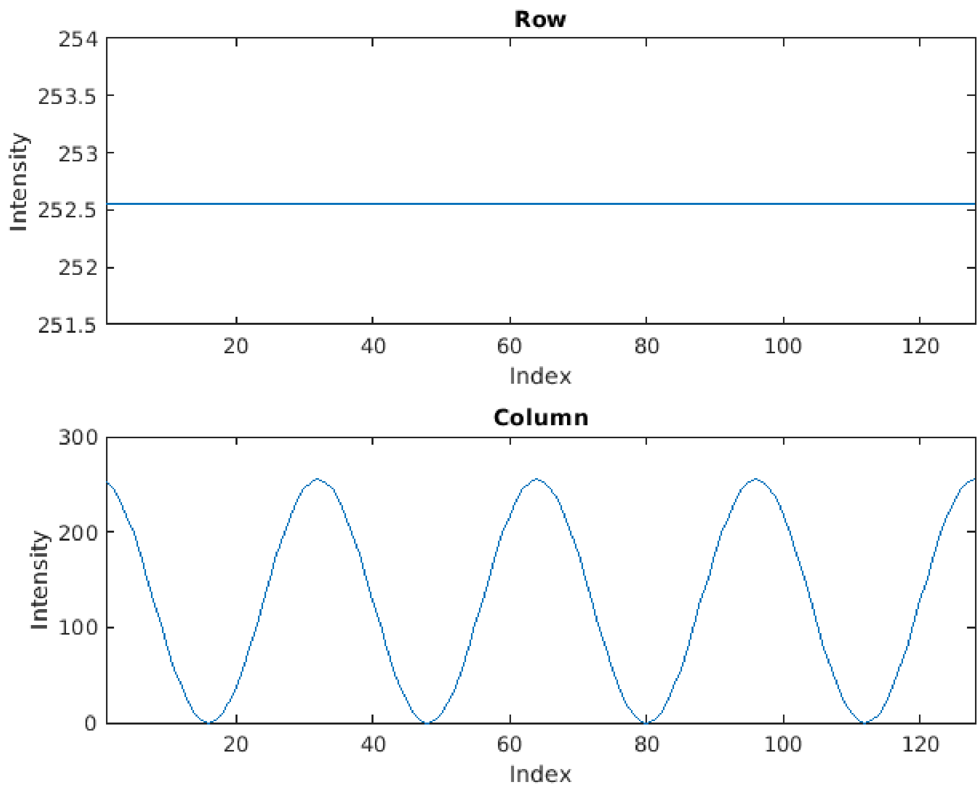
Q1: Load pattern.mat, display the image and add a color bar. What pattern do you observe? (e.g., sinusoid or square function? in what direction?) What is the frequency of this pattern in cycles per pixel?

```
load('pattern.mat');
figure(1), clf;
imagesc(pattern);
colormap gray(256);
colorbar;
ylabel('Rows');
xlabel('Columns');
axis image;
title('Pattern');

% YOUR ANSWER: There are four vertical sinusoids in the observed
 pattern.
% The frequency of each sinusoid will be 4/128 = 0.03125 cycles per
 pixel.
f = 4/128;
```



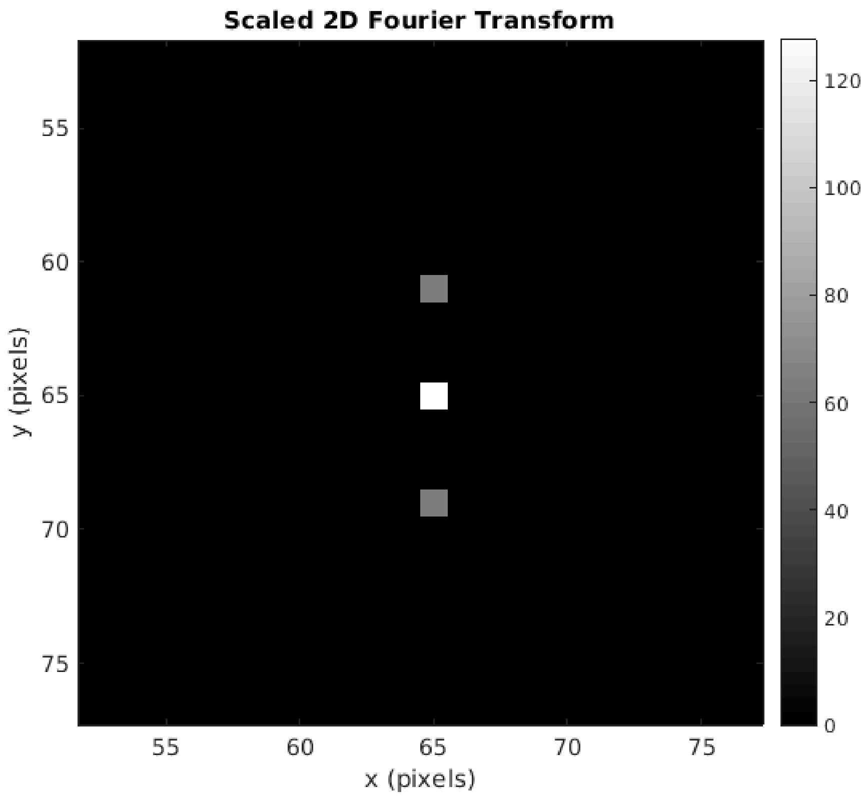
Q2: Plot one row and one column of pattern on a same figure using subplots. Add titles and axis labels. What type of function is represented in this image?

```
t = 1:128;

row = pattern(1,:);
col = pattern(:,1);

figure(2), clf;

subplot(2, 1, 1);
plot(t, row);
title('Row');
xlabel('Index');
ylabel('Intensity');
xlim([1 128]);

subplot(2, 1, 2);
plot(t, col);
title('Column');
xlabel('Index');
xlim([1 128]);
ylabel('Intensity');

% YOUR ANSWER: Its a sinusoid function (a cosine wave) on the column
 axis.
```



Q3: Display the 2D Fourier transform of this image using 'fft2'. Use the 'zoom'command to zoom in on the part of the image with signal. You must divide the matlab output by 128^2 for the scaling to be consistent.

```
ft = fftshift(fft2(pattern))/(128^2);
figure(3), clf;
imagesc(real(ft));
colormap gray(256);
axis image;
colorbar;
zoom(5);
title('Scaled 2D Fourier Transform');
xlabel('x (pixels)');
ylabel('y (pixels)');
%
 ***********************************************************************
%   SHOW YOUR IMAGE FOR Q3 TO THE TA TO RECEIVE CREDIT FOR THE LAB IF
 YOU
%   ARE NOT PRESENT AT THE DISCUSSION SESSION ON FRIDAY.
% ********************************************
 **************************
```

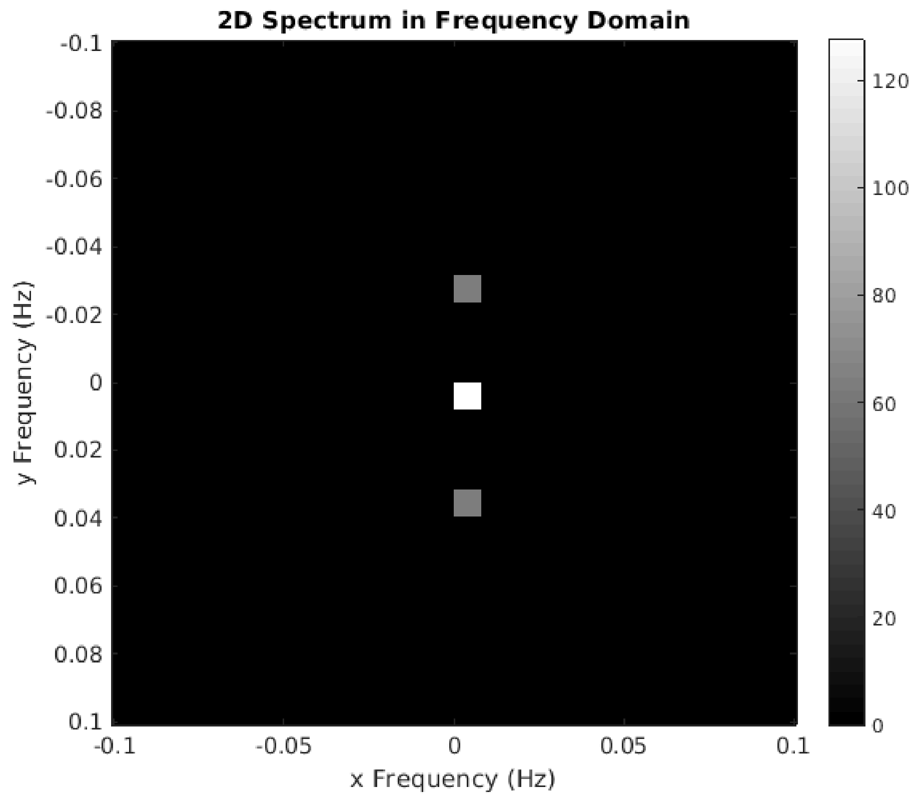


Q4: How does the 2D FT relate to the original image? Think about the original time domain and the frequency domain after FT.

```
% YOUR ANSWER:
% On observing the two figures, it can be inferred that there is a DC
% signal (the white patch) as well as a sinusoid (the positive and
 negative
% frequency). The DC component is bright white spot because it has
 higher
% amplitude and the gray spots are sinusoid components of positive and
% negative frequencies with smaller amplitude. The sinusoids are at
 peak
% +/- 4 from the centered DC component.
```

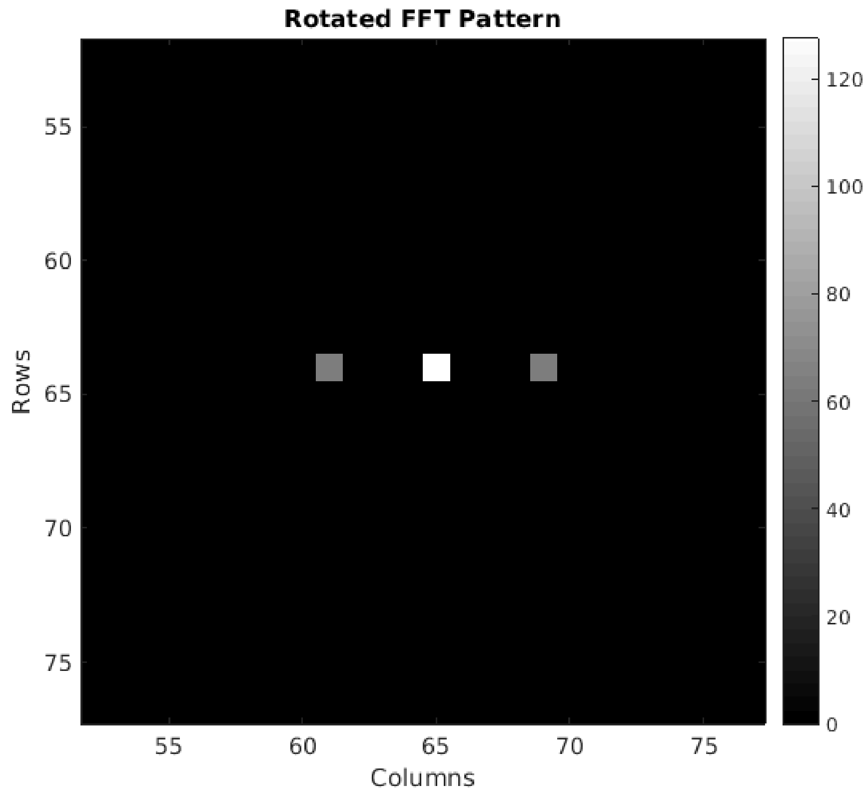
Q5: Re-plot the 2D signal, this time with respect to the proper frequency vectors. Again, zoom in.

```
figure(4), clf;
x_freq = linspace(-1/2, 1/2, 128);
y_freq = linspace(-1/2, 1/2, 128);
imagesc(x_freq, y_freq, real(ft));
colormap gray(256);
colorbar;
axis image;
zoom(5);
title('2D Spectrum in Frequency Domain');
xlabel('x Frequency (Hz)');
ylabel('y Frequency (Hz)');
```



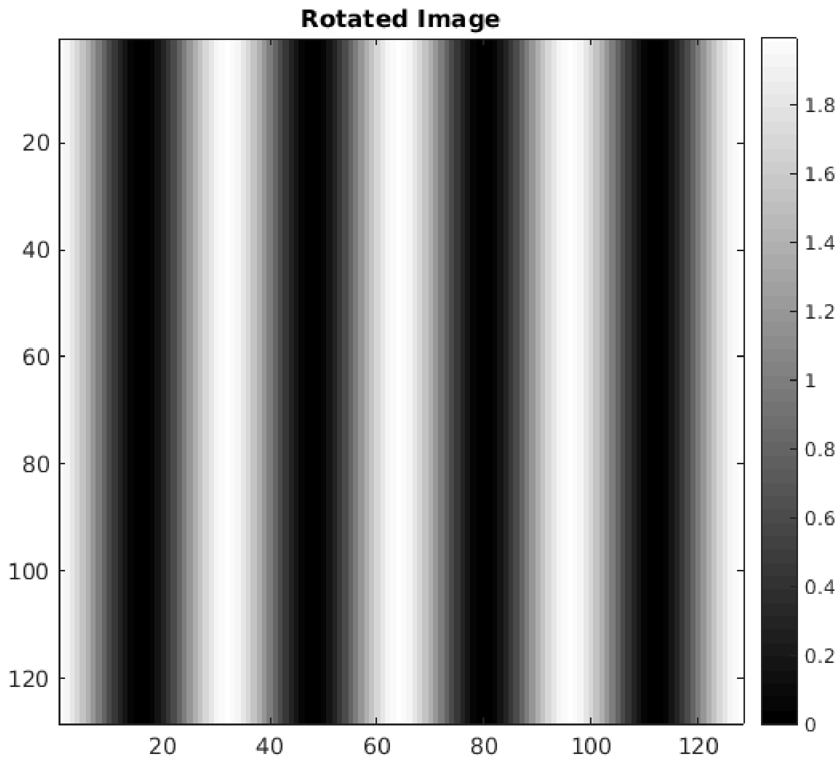
Q6: Time to work backwards... Design an image in FREQUENCY space that will return pattern, but rotated 90-degrees, using 'rot90()'. Display the image using 'imagesc()'(zoomed if necessary) below.

```
figure(5), clf;
ft_r = rot90(ft);
imagesc(real(ft_r));
x_freq = linspace(-1/2, 1/2, 128);
y_freq = linspace(-1/2, 1/2, 128);
title('Rotated FFT Pattern');
axis image;
colorbar;
zoom(5);
colormap gray(256);
ylabel('Rows');
xlabel('Columns');
```



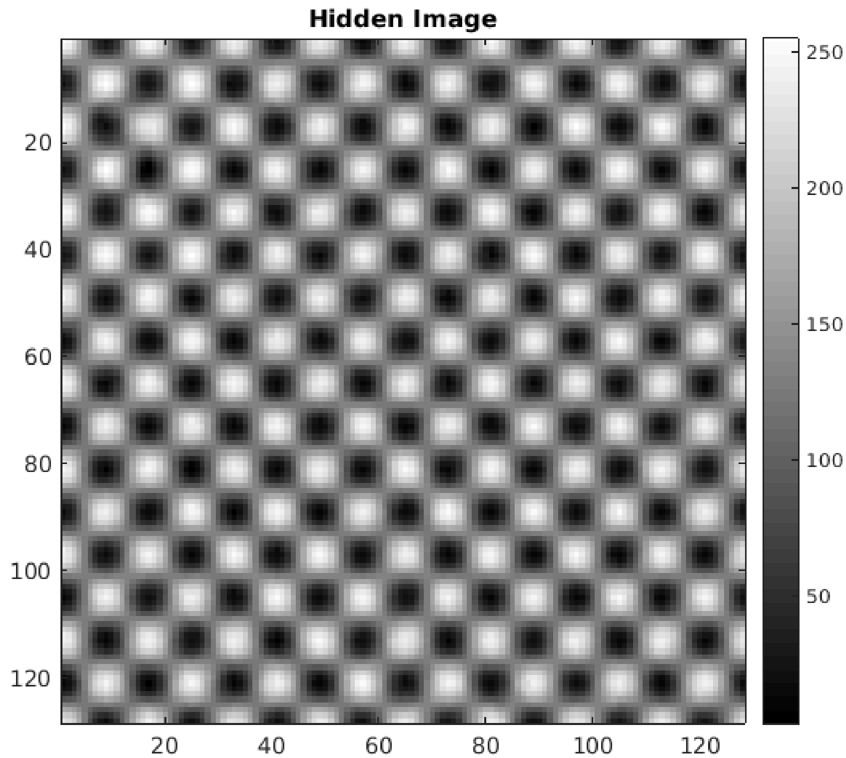
Q7: Take the inverse FT of your image using 'fftshift()' and then 'ifft2()', and prove that your design works. Display the image in the time domain.

```
figure(6), clf;
ift = 128*ifft2(fftshift(ft_r), 128, 128);
imagesc(abs(ift));
axis image;
colormap gray(256);
colorbar;
title('Rotated Image');

% The final image obtained from the rotated ffourier transform is also
% rotated.
```



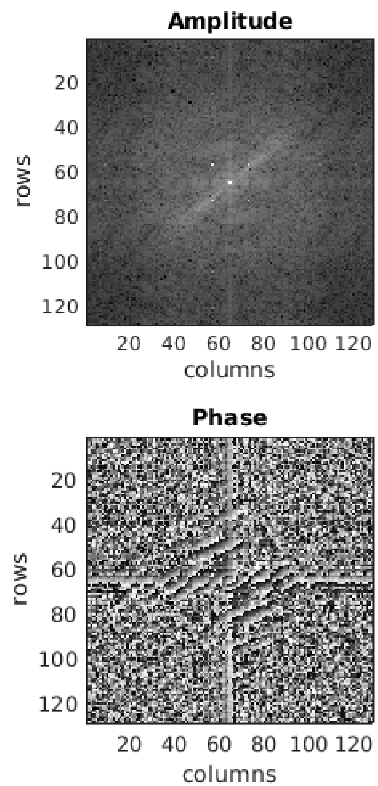
Q8: CLEANING AN IMAGE CONTAMINATED BY COSINUSOIDAL NOISE. Read and display an image from file 'hidden_image.jpg' using 'imread()' and 'imagesc()'.

```
figure(7), clf;
img = imread('hidden_image.jpg');
imagesc(img);
colormap gray(256);
colorbar;
axis image;
title('Hidden Image');
```



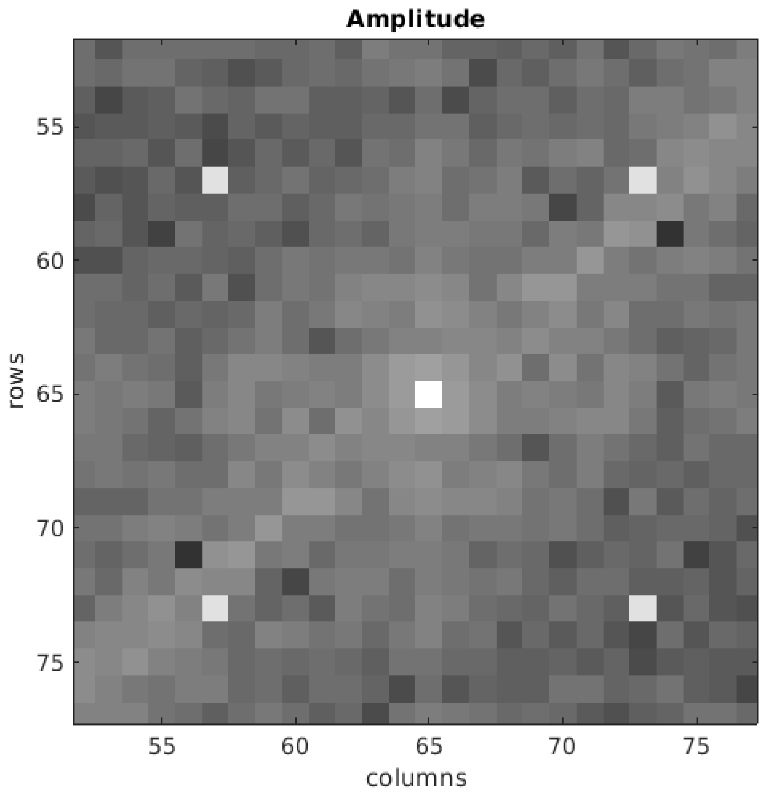
Q9: Take FT of this image and plot the amplitude and phase spectra in one figure using subplots. Hint: You need to plot the logarithm of the amplitude.

```
figure(8), clf;
ft = fftshift(fft2(img))/128^2;

subplot(2,1,1);
imagesc(log(abs(ft)));
title('Amplitude');
colormap gray(256);
xlabel('columns');
ylabel('rows');
axis image;

subplot(2,1,2);
imagesc(atan2(real(ft), imag(ft)));
title('Phase');
colormap gray(256);
xlabel('columns');
ylabel('rows');
axis image;
```



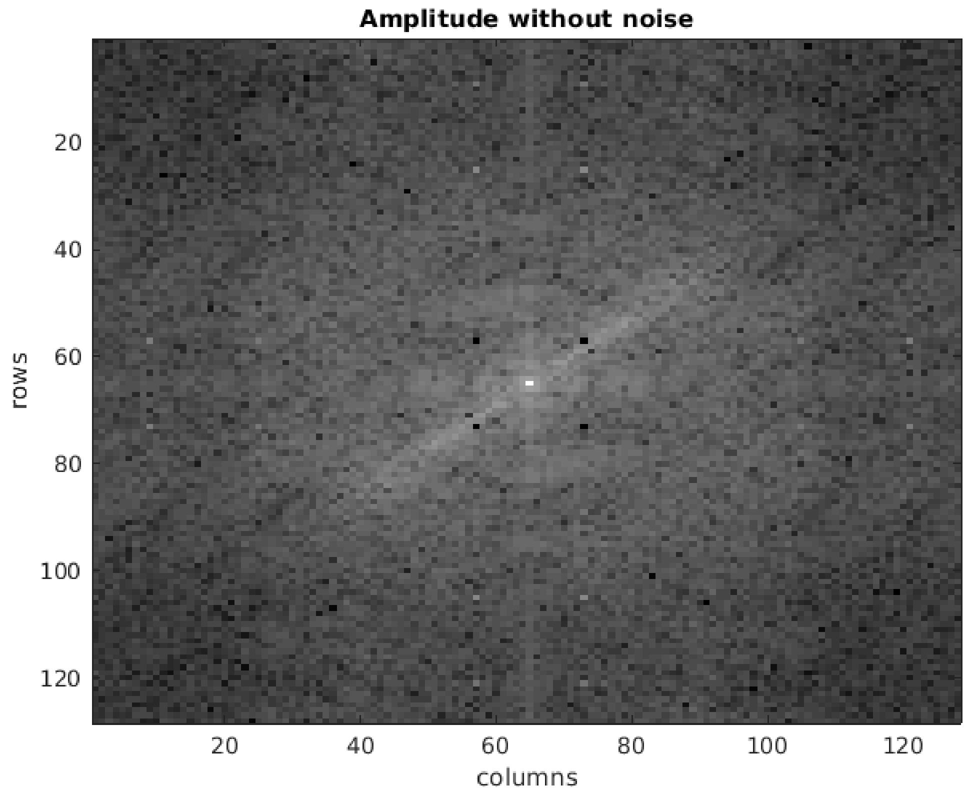
Q10: Zoom in on the appropriate part of the amplitude spectrum to find the offending sinusoids.

```
figure(9), clf;
ft = fftshift(fft2(img));
imagesc(log(abs(ft)));
title('Amplitude');
colormap gray(256);
xlabel('columns');
ylabel('rows');
axis image;
zoom(5);

% The offending signals are at (57, 57), (57, 73), (73, 57) and (73,
 73).
% The white patches visible in center seems to be the DC component in
 Figure 9.
```



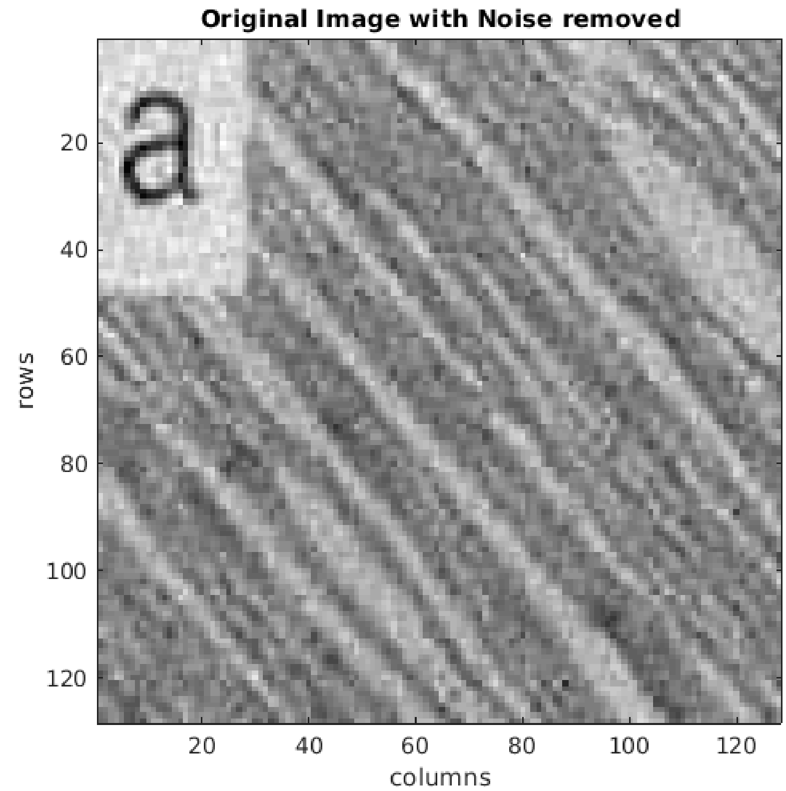
Q11: Make changes to the FT of this image to eliminate the sinusoidal noise. Hint: Create a new matrix, directly pick up the pixels that you need to clean and give them a value of 0. Plot the zoomed amplitude spectrum of the result.

```
new_image = repmat(ft, 1, 1);
new_image(57, 57) = 0;
new_image(73, 73) = 0;
new_image(73, 57) = 0;
new_image(57, 73) = 0;

figure(10), clf;
imagesc(log(abs(new_image)));
colormap gray(256);
title('Amplitude without noise');
xlabel('columns');
ylabel('rows');
```



Q12: Perform an inverse FT and examine the resulting image. Display the image that was hidden behind the noise. Hint: It should look like a biological sample labeled with an '(a)' at the upper left corner.

```
figure(11), clf;
colormap gray(256);
origin_image = ifft2(fftshift(new_image));
imagesc(abs(origin_image));
axis image;
title('Original Image with Noise removed');
xlabel('columns');
ylabel('rows');
```



When you are done:

```
% * Show up during the discussion session on Friday or show the
 indicated result/figure to the TA during the lab period
%       to receive credit
% * upload your script to Sakai
%   * upload a pdf containing your script and outputs.
```

Published with MATLAB® R2020b