Homework 7 Problem 2

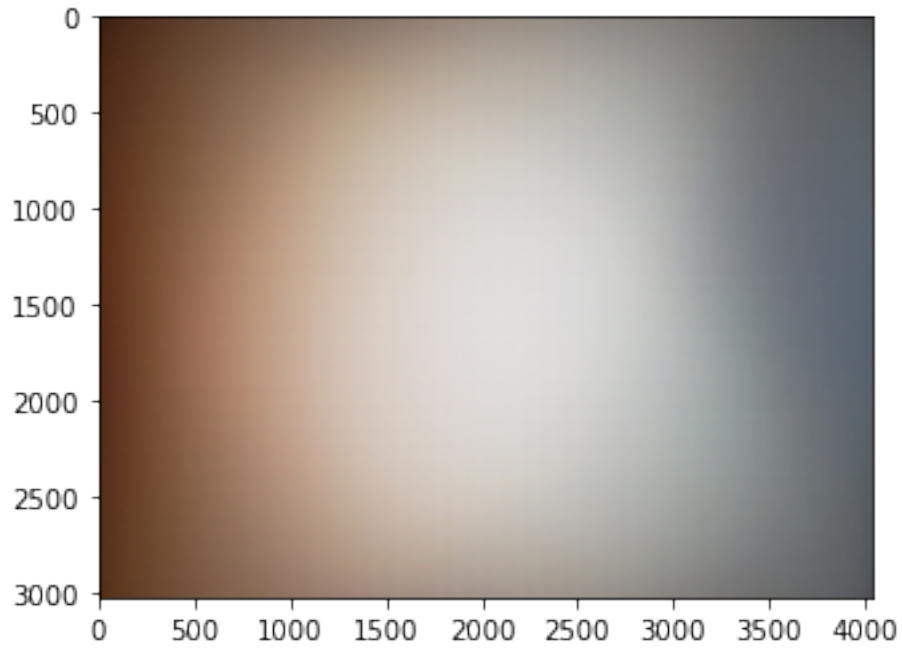
Austin Marga

First, our import statements.

```
import matplotlib.pyplot as plt
import matplotlib.image as mping
import numpy as np
from PIL import Image, ImageOps
```

Next, lets read in a `.jpg` picture that we want to investigate.

```
image = mping.imread("pattern.jpg")
imgplot = plt.imshow(image)
plt.show()
```

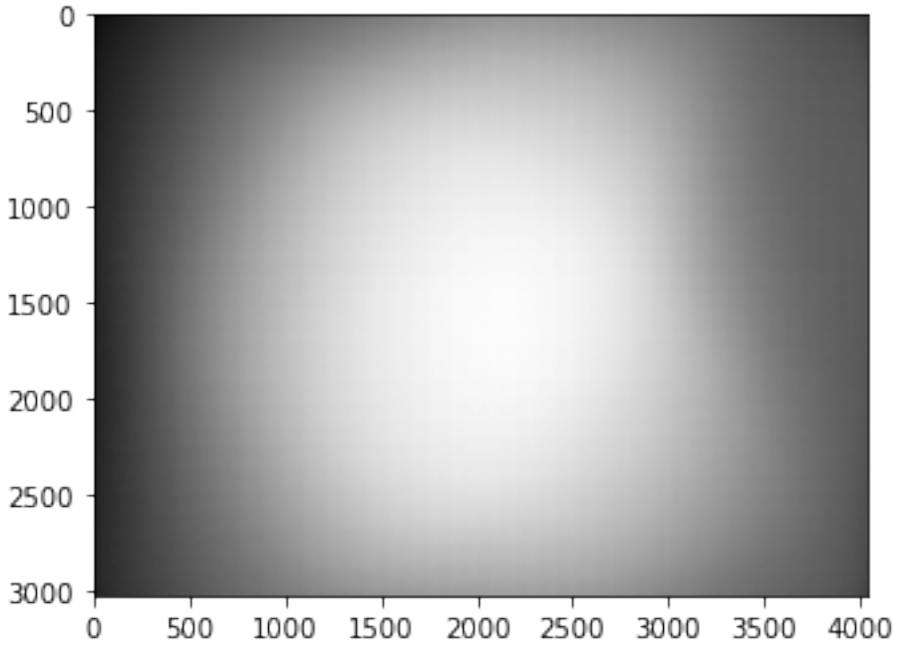


Converting the jpg to useable data

If you look closely, you can see a white grid pattern. Let's try to remove this. First, let's put it to a greyscale to we dont have to worry about the color data.

```
img = np.array(ImageOps.grayscale(Image.open('pattern.jpg')))
plt.imshow(img, "gray")
```

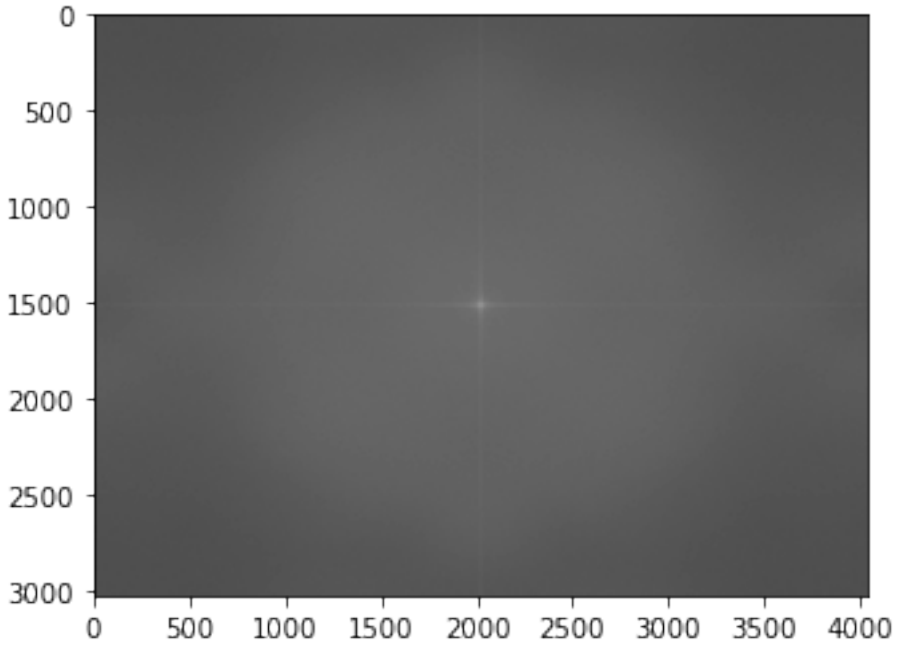
```
<matplotlib.image.AxesImage at 0x7febcfd2cac0>
```



Next, we can take the 2D FFT of the grid and center it.

```
fimg = np.fft.fft2(img)
fimgcenter = np.fft.fftshift(fimg)
plt.imshow(np.log(1+np.abs(fimgcenter)), "gray")
```

```
<matplotlib.image.AxesImage at 0x7febce4a2f70>
```



The center point seems to show a 2D fourier frequency peak that we want to remove. Since `fimgcenter` is a numpy array of numpy arrays, what my plan would be is to remove the arrays assigned to the pixels with x values close to 2000 and y values close to 1500 that form this cross shape. I didn't make much progress in trying to remove this section.

Padding the data

I've never dealt with tuples before, so this is me mucking around trying to figure out how to pad.

```
fimgcenter.shape
```

```
(3024, 4032)
```

```
pad = ()
```

```
pad_size = len(fimgcenter[1] - fimgcenter[0])
```

```
len(fimgcenter[0])
```

```
4032
```

```
for entry in range(0,pad_size):
    pad  = pad + (0,)
```

```
for entry in range(0,4032 - 3024):
    pad  = pad + (0,)
```

```
fimgcenter.shape
```

```
(3024, 4032)
```

```
type(fimgcenter[0][0])
a = np.complex128(0)
type(a)
```

```
numpy.complex128
```

```
def loop(pad):
    for entry in range(0,4032 - 3024):
        pad  = pad + np.complex128(0)
        return pad
```

```
a = loop(pad)
len(a)
```

```
5040
```

```
print(len(fimgcenter[1]))
```

```
4032
```

I thought I can concatenate tuples like this, but the structure of `fimgcenter` must be more complicated, like tuples with entries in the tuples being lists? Let's try to make a filter. This next line will give me an error if I run it.

```
# fimgcenter[0] = fimgcenter[0] + pad
```

Making the Filter

```
y_len = len(fimgcenter[0])
x_len = len(fimgcenter[1])
print(y_len)
print(x_len)
x = np.linspace(-10, 10, x_len)
y = np.linspace(-10, 10, y_len)
```

```
4032
4032
```

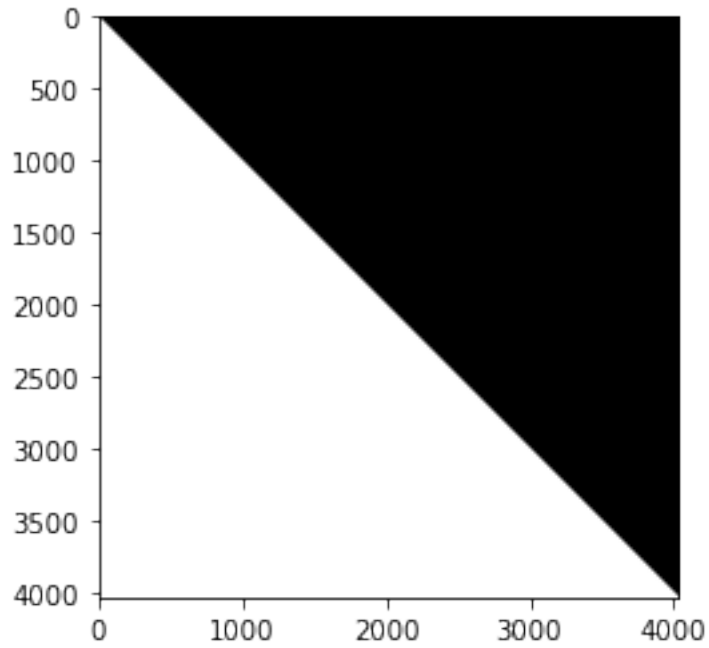
Let's try a filter that removes half of the white line we saw in the image. We can do that with a $y = -x$ type of 2d grid.

```
R = 0.00001
xx, yy = np.meshgrid(x, y, sparse=True)
lowpass = (np.sqrt(xx**2 + yy**2) >= R)*1

lowpass_rect = ((xx * yy) <= R)*1
lowpass_half = ((yy > xx))*1

plt.imshow(lowpass_half, "gray")
```

```
<matplotlib.image.AxesImage at 0x7febce41eca0>
```



Applying my filter

Because the dimensionality is wrong between x and y, I can't apply the filter. I Don't quite understand the composition of `fimgcenter`.

Let's walk through what I would have done. Next, I apply the filter to the FFT image.

```
# plt.imshow(np.log(1+np.abs(lowpass_half*fimgcenter_new)), "gray")
```

And then I would take the iFFT of the filtered image and display it here.

```
# filtered = np.fft.ifft2(lowpass_half*fimgcenter)
# plt.imshow(np.log(1+np.abs(filtered)), "gray")
```

If I had padded correctly, the result would be the original image with a reduction in the pattern.