EXAM III SOLUTIONS

1/ a) As we've seen previously, the intensity distribution produced by each array is:

$$I_{1,2} \sim \left(\frac{\sin N\delta/2}{\sin \delta/2}\right)^2, \quad \text{where } \delta = \frac{2\pi}{\lambda} a \sin\theta$$

For the resultant intensity of two coherent sources, have

$$I = I_1 + I_2 + 2\sqrt{I_1 I_2}\cos\phi \qquad (I_1 = I_2)$$

$$= 2I_1(1+\cos\phi)$$

$$= 4I_1 \cos^2(\phi/2)$$

$$\sim 4\left(\frac{\sin N\delta/2}{\sin \delta/2}\right)^2 \cos^2\left(\frac{\pi b \sin\theta}{\lambda}\right), \quad \text{where } \phi = \frac{2\pi}{\lambda} b \sin\theta$$

Have max intensity at $\frac{\delta}{2} = 0$, so $I_0 \sim 4N^2$

$$\Longrightarrow I = \frac{I_0}{N^2}\left(\frac{\sin \frac{N\delta}{2}}{\sin \delta/2}\right)^2 \cos^2\left(\frac{\pi b \sin\theta}{\lambda}\right)$$

1/

(b) Not just photons, but particles with mass, exhibit particle-wave duality. A consequence of this is that massive particles can be treated as "matter waves" having a characteristic wavelength first described by Louis de Broglie.

$$\Rightarrow p = m_0 v \qquad \Rightarrow \lambda = \frac{h}{p} = \frac{h}{m_0 v}$$

For an electron in an electric potential, V, it experiences a potential energy of $U = eV = \frac{1}{2} m_0 v^2$

$$\Rightarrow v = \sqrt{\frac{2eV}{m_0}} \qquad \Rightarrow \lambda_e = \frac{h}{\sqrt{2 m_0 eV}}$$

So, the e^- wavelength depends on it's own potential energy.

Compare diffraction limits:

$$\boxed{\theta_{min} = \frac{1.22\lambda}{D}}$$

γ/ $$\theta_{min}^{\gamma} = \frac{1.22\,(532\text{ nm})}{D} = \frac{(649.04\text{ nm})}{D}$$

e^-/ $$\theta_{min}^{e^-} = \frac{1.22}{D} \cdot \frac{h}{\sqrt{2 m_0 eV}} = \frac{1.22}{D} \cdot \frac{h}{\sqrt{2\,(0.511\text{ MeV}/c^2)(10\text{ keV})}}$$

$$= \frac{1.22}{D} \cdot \frac{1240\text{ eV}\cdot\text{nm}}{\sqrt{2\,(0.511\times10^6)^2(10\times10^3)}\,\text{eV}} = \frac{(14.96\text{ pm})}{D}$$

$\Rightarrow$ electrons offer $\sim 43372\times$ resolution enhancement over photons here!

P47 - Exam III - Problem 2

November 13, 2017

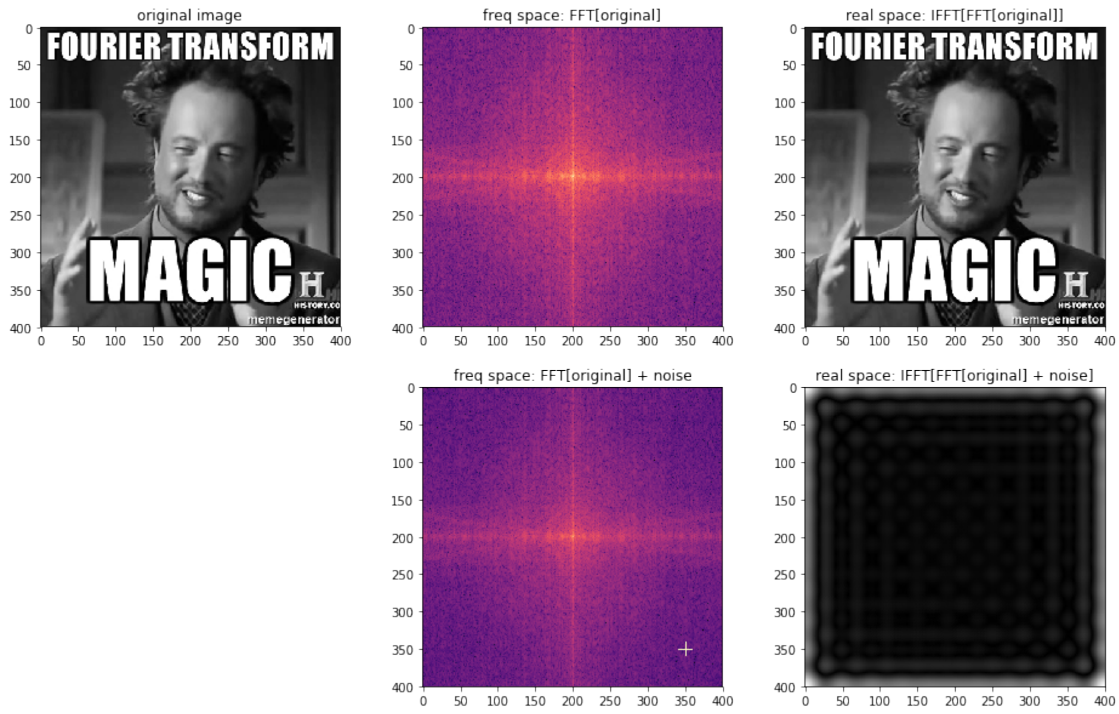
```
In [1]: # package imports
        from math import *
        import numpy as np
        import scipy.io as sio
        from skimage import io

        %matplotlib inline
        import matplotlib
        import matplotlib.pyplot as plt
```

1 Adding Frequency Noise

First, I'll load the original image I found, Fourier transform it, add some noise to the frequency spectrum, and then inverse Fourier transform it as the final file I'll give to you for the exam.

```
In [2]: # open original image into matrix
        file_loc = %pwd
        file_name = file_loc + '\\P47_ExamIII_2_img.jpg'
        image = io.imread(file_name, as_grey=True)

        # fft the image and create copy for adding noise
        fft_image = np.fft.fftshift(np.fft.fft2(image))
        fft_noise = np.copy(fft_image)

        # add frequency noise to copy
        noise_value = 10*np.abs(fft_image).max()
        fft_noise[350,340:360] = noise_value
        fft_noise[340:360,350] = noise_value

        # ifft each the original and noise-added freq spectra
        re_image = np.fft.ifft2(np.fft.ifftshift(fft_image))
        re_image_noise = np.fft.ifft2(np.fft.ifftshift(fft_noise))

        # save noisy field data to csv file
        np.savetxt('P47_ExamIII_2_noise.csv', re_image_noise, delimiter=',')

        # save again for matlab users
        sio.savemat('P47_ExamIII_2_noise.mat', {'field_i':re_image_noise})
```

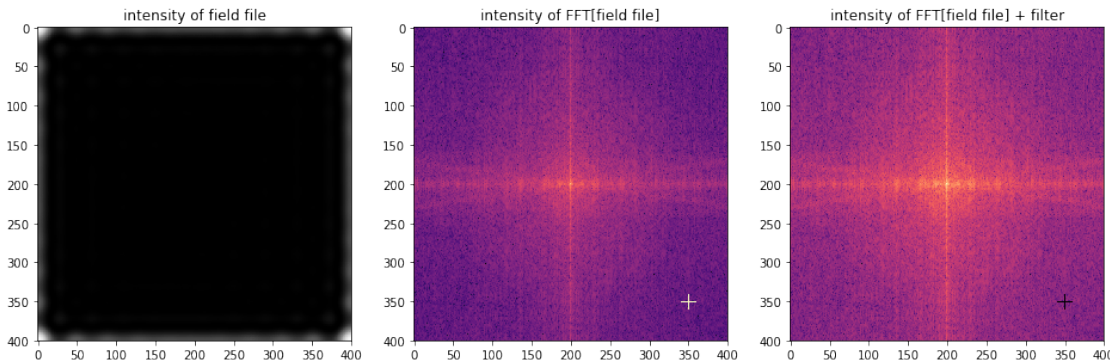
In [3]:
```python
# plotting
plt.figure(figsize=(16,10))

plt.subplot(2,3,1)
plt.imshow(image, cmap='gray')
plt.title('original image')

plt.subplot(2,3,2)
plt.imshow(np.log(np.abs(fft_image)), cmap='magma')
plt.title('freq space: FFT[original]')

plt.subplot(2,3,3)
plt.imshow(np.abs(re_image), cmap='gray')
plt.title('real space: IFFT[FFT[original]]')

plt.subplot(2,3,5)
plt.imshow(np.log(np.abs(fft_noise)), cmap='magma')
plt.title('freq space: FFT[original] + noise')

plt.subplot(2,3,6)
plt.imshow(np.abs(re_image_noise), cmap='gray')
plt.title('real space: IFFT[FFT[original] + noise]')

plt.show()
```



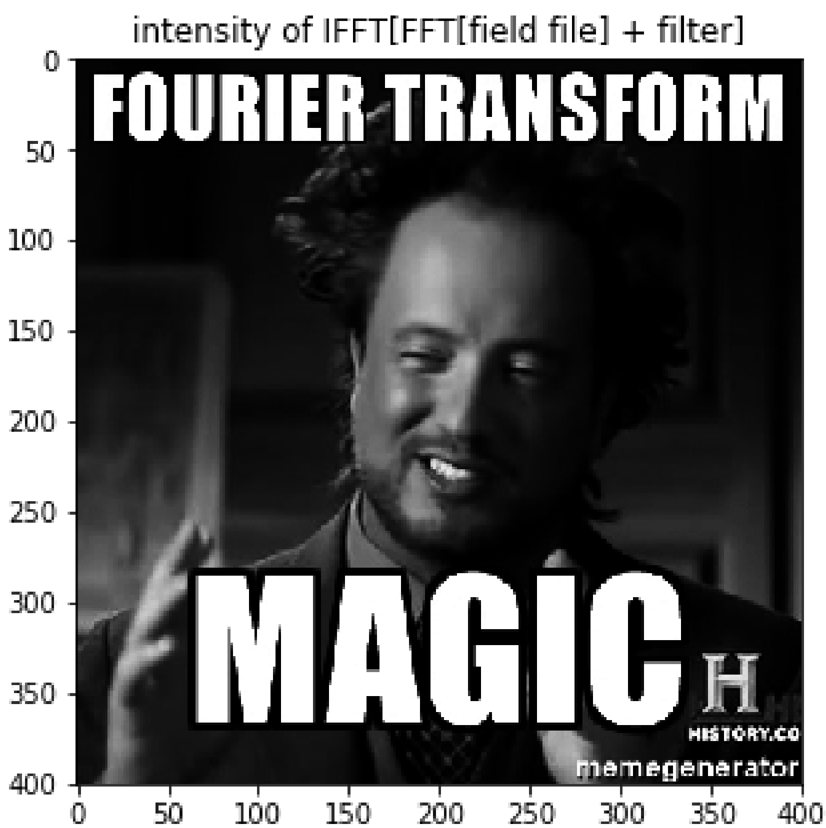
2 Solution

```
In [4]: # open noisy field data from disk to process
        field_i = np.loadtxt('P47_ExamIII_2_noise.csv', delimiter=',', dtype=np.complex128)

        # compute the irradiance/intensity distribution (I ~ EE*)
        # plotting needs floats, so take real of complex numbers (imaginaries = 0 anyway)
        irrad_i = (field_i*field_i.conjugate()).real

        # check field fft, shifting low freqs to center
        fft_field = np.fft.fftshift(np.fft.fft2(field_i))
        fft_irrad = (fft_field*fft_field.conjugate()).real

        # create a copy to edit
        fft_field_filt = np.copy(fft_field)

        # freq spectrum shows an anomolous bright cross
        # set those values to some small number
        fft_field_filt[fft_field_filt > 0.99*fft_field_filt.max()] = 0.01 + 0.01j
        fft_irrad_filt = (fft_field_filt*fft_field_filt.conjugate()).real

        # ifft the filtered freq spectum to get cleaned image
        field_f = np.fft.ifft2(np.fft.ifftshift(fft_field_filt))
        irrad_f = (field_f*field_f.conjugate()).real
```

```
In [5]: # plotting
        plt.figure(figsize=(16,16))

        # plot the initial irradiance distribution of file given
        plt.subplot(1,3,1)
        plt.imshow(irrad_i, cmap='Greys_r')
        plt.title('intensity of field file')

        # plot the far-field diffraction intensity
        plt.subplot(1,3,2)
        plt.imshow(np.log(fft_irrad), cmap='magma')
        plt.title('intensity of FFT[field file]')

        # plot the filtered diffraction pattern
        plt.subplot(1,3,3)
        plt.imshow(np.log(fft_irrad_filt), cmap='magma')
        plt.title('intensity of FFT[field file] + filter')

        plt.show()
```




```
In [8]: # plot final cleaned intensity distribution
        plt.figure(figsize=(5,5))
        plt.imshow(irrad_f, cmap='gray')
        plt.title('intensity of IFFT[FFT[field file] + filter]')
        plt.show()
```



3/

(a) Have allowed longitudinal modes given by

$$L = n\frac{\lambda}{2}$$, where n is some natural integer $(1, 2, 3, \dots)$

and have no index of refraction so photon velocity is $= c$

$$\Longrightarrow \nu\lambda = c \Longrightarrow \omega_L = \frac{2\pi c}{\lambda} = \frac{n\pi c}{L}$$

which means the modes are separated by

$$\delta\omega_L = \frac{\pi c}{L} = \frac{\pi(3\times10^8 \,m/s)}{(150\, cm)} = \underline{\underline{2\pi \times 10^8 \frac{rad}{sec}}}$$

(b) If the relative phases, ϕ_n, of the different modes are randomly distributed, then the modes are incoherent with each other. The total intensity, I, will just be the sum of the intensities, I_0, of each mode (all equally weighted by gain profile)

$$\Longrightarrow I = I_0\left(\frac{\Delta\omega}{\delta\omega_L}\right) = \underline{\underline{9549\, I_0}}$$

→

3/

(c) With mode-locking, the intensity must now be found by first adding-up the electric fields (not intensities!)

$$\Rightarrow I(t) = I_0 \left| e^{-i\omega_1 t} + \cdots + e^{-i\omega_n t} \right|^2 = I_0 \left(\frac{\sin \frac{n}{2} \delta\omega_L t}{\sin \frac{1}{2} \delta\omega_L t} \right)^2$$

where $\omega_n = n \delta\omega_L$ & $n = 9549$

the pulse duration is determined by the largest mode separation

$$\Rightarrow \tau = \frac{2\pi}{n\delta\omega_L} = \frac{2L}{nc} = \frac{(1 \text{ sec})}{(9549 \times 10^8)} = \underline{\underline{1.05 \times 10^{-12} \text{ sec.}}}$$

but the pulse separation is determined by the interval between modes

$$\Rightarrow \delta t = \frac{2\pi}{\delta\omega_L} = \underline{\underline{1 \times 10^{-8} \text{ sec}}}$$

with a peak intensity occurring at $t=0$

$$\Rightarrow I(t=0) = I_0 n^2 = \underline{\underline{9.12 \times 10^7 I_0}}$$

[<u>MUCH</u> larger than CW case]