Principe - Physics 265 PS8

January 6, 2024

1 Physics 265 Problem Set 8

Rene L. Principe Jr.
PhD Physics
2015-04622

[1]:
```
import numpy as np
import matplotlib.pyplot as plt
from scipy.special import jv
from tqdm import tqdm
```

2 Problem 8.1

2.1 Fraunhofer diffraction pattern by a circular aperture

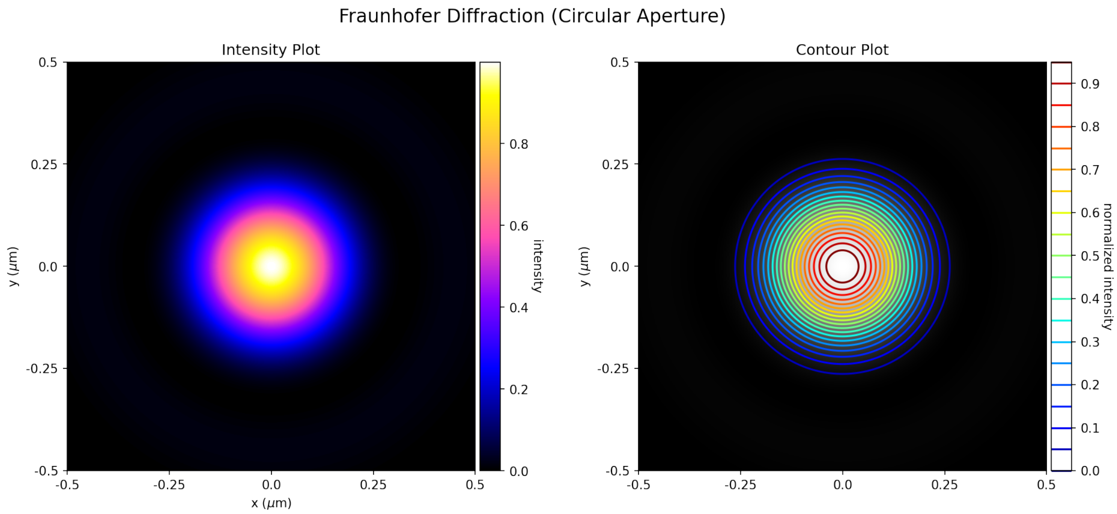
(Section 8.5.1, Born & Wolf). Generate the normalized (I0 = 1) intensity contour plot I(x, y) that is produced by a point light source (located at infinity, = 0.550 micron) via a uniformly-illuminated lens of radius r = 1 cm, focal length f = 3 cm, and refractive index n = 1.53.

The origin of the x−y plane coincides with the lens focus. Intensity resolution of contour plot: 0.05

[2]:
```
def I_circ(lambda_, x, y, a, I0):
    w = np.sqrt(np.power(x, 2) + np.power(y,2))
    k = 2 * np.pi / lambda_
    num = 2 * jv(1, k * a * w)
    denom = k * a * w
    return I0*(num/denom)**2

I0 = 1
r = 1
f = 3
n = 1.53
lambda_ = 0.550e-4
n_points = 1000

x = np.linspace(-0.5, 0.5, n_points) * 1e-4
X, Y = np.meshgrid(x, x)
```

```
I_circle = I_circ(lambda_, X, Y, r, I0)
```

[3]:
```
plt.figure(dpi = 150, figsize = (15,6))

plt.subplot(121)
plt.title('Intensity Plot')
img1 = plt.imshow(I_circle, cmap='gnuplot2', origin='lower')
cbar1 = plt.colorbar(img1, fraction=0.045, pad=0.01)
cbar1.set_label('intensity', rotation=270, labelpad=10)
plt.xlabel('x ($\mu$m)')
plt.ylabel('y ($\mu$m)')
ticks = np.linspace (0, n_points, 5)
ticklabels = [str(tick/n_points-0.5) for tick in ticks]
plt.xticks(ticks, ticklabels)
plt.yticks(ticks, ticklabels)

plt.subplot(122)
plt.title('Contour Plot')
plt.imshow(I_circle, cmap='gray', origin='lower')
img2 = plt.contour(I_circle, levels=np.arange(0, 1, 0.05), cmap='jet')
cbar2 = plt.colorbar(img2, fraction=0.045, pad=0.01)
cbar2.set_label('normalized intensity', rotation=270, labelpad=10)
plt.ylabel('y ($\mu$m)')
plt.xticks(ticks, ticklabels)
plt.yticks(ticks, ticklabels)

plt.suptitle('Fraunhofer Diffraction (Circular Aperture)', fontsize = 15)
plt.show()
```

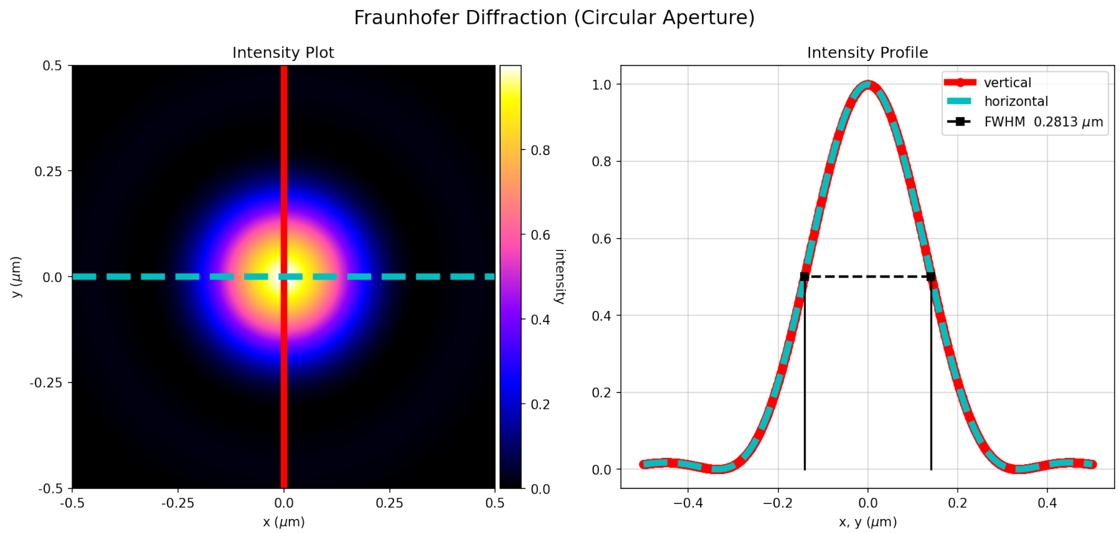


3 Problem 1.2

For f = 3 cm and n = 1.153, plot the full width at half maximum (FWHM) value of the normalized central spot as a function of circular aperture radius r where: 0.1 r (cm) 10 (512 data points) and = 0.550 micron.

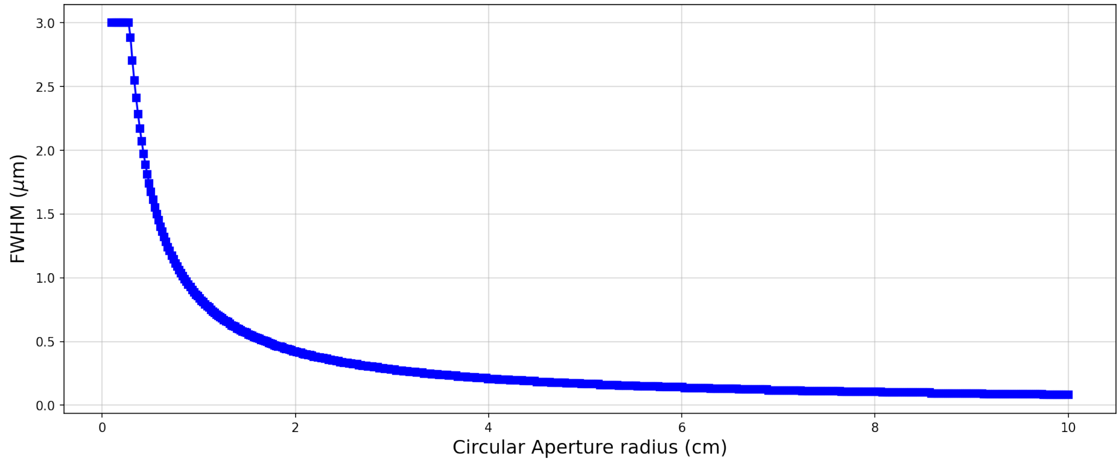
Discuss briefly the salient features and implications of your results.

[4]:
```
i_half = int(n_points/2)

plt.figure(dpi = 150, figsize = (15,6))

plt.subplot(121)
plt.title('Intensity Plot')
img1 = plt.imshow(I_circle, cmap='gnuplot2', origin='lower')
cbar1 = plt.colorbar(img1, fraction=0.045, pad=0.01)
cbar1.set_label('intensity', rotation=270, labelpad=10)
plt.xlabel('x ($\mu$m)')
plt.ylabel('y ($\mu$m)')
plt.xticks(ticks, ticklabels)
plt.yticks(ticks, ticklabels)
plt.axvline(i_half, color = 'r', ls = '-', lw = 5)
plt.axhline(i_half, color = 'c', ls = '--', lw = 5)

plt.subplot(122)
plt.title('Intensity Profile')
plt.grid(alpha = 0.5)
plt.plot(x/1e-4, I_circle[:, i_half], 'ro-', lw = 5, label = 'vertical')
plt.plot(x/1e-4, I_circle[i_half, :], 'c--', lw = 5, label = 'horizontal')
plt.xlabel('x, y ($\mu$m)')


x_c_cross = I_circle[:, i_half]
x_half = x[x_c_cross > 1/2]
x1 = np.max(x_half)/1e-4
x2 = np.min(x_half)/1e-4
FWHM_circle = x1-x2
plt.plot([x1,x1], [0,0.5], 'k')
plt.plot([x2,x2], [0,0.5], 'k')
plt.plot([x1,x2], [0.5,0.5], 'ks--', lw = 2, label = 'FWHM % .4f $\mu$m' %␣
 ↪FWHM_circle)

plt.legend()
plt.suptitle('Fraunhofer Diffraction (Circular Aperture)', fontsize = 15)
plt.show()
```



[5]:
```
f = 3
r_values = np.linspace(0.1, 10, 512)
FWHM = np.zeros(len(r_values))

for i, r in tqdm(enumerate(r_values)):
    x_c_cross = I_circ(lambda_, x, 0, r, I0)
    x_half = f*(x[x_c_cross > 1/2])/1e-4
    FWHM[i] = np.max(x_half) - np.min(x_half)

plt.figure(dpi = 150, figsize = (15,6))
plt.grid(alpha = 0.5)
plt.plot(r_values, FWHM,'bs-')
plt.xlabel('Circular Aperture radius (cm)', size=15)
plt.ylabel('FWHM ($\mu$m)', size=15)
plt.show()
```

```
512it [00:00, 1148.86it/s]
```



The Full Width at Half Maximum (FWHM) demonstrates a decreasing trend with an increase in the circular aperture radius. In this scenario, Fraunhofer diffraction is dictated by a Bessel function, resulting in an Airy pattern. Smaller apertures tend to generate more pronounced Airy patterns, contributing to larger FWHM values. As the circular aperture widens, the influence of the Bessel function diminishes, leading to a reduction in FWHM values.

4 Problem 1.3

4.1 Fraunhofer diffraction pattern by a rectangular aperture

(Section 8.5.1, Born & Wolf).

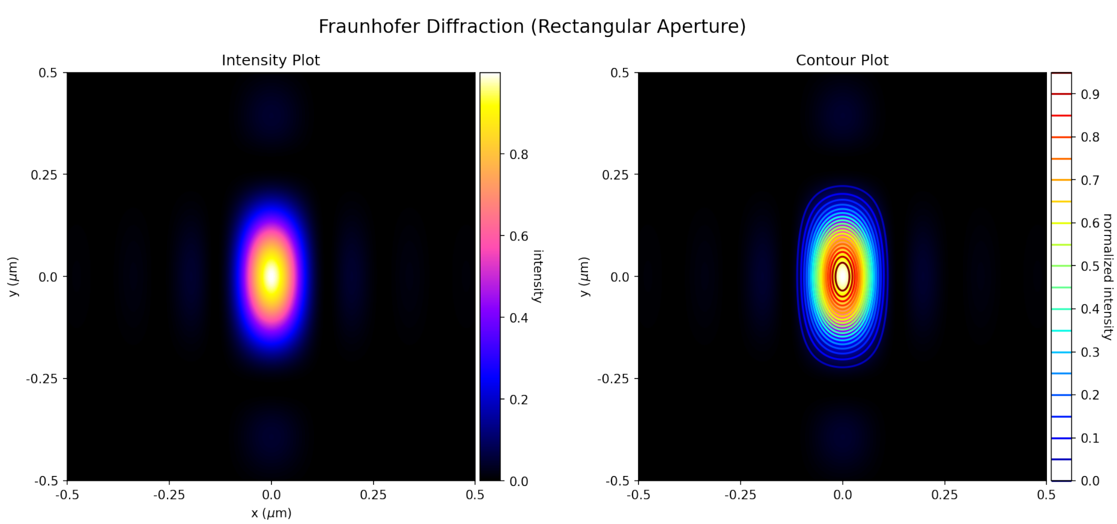
Generate the normalized (I0 = 1) intensity contour plot I(x, y) that is produced by a point light source (located at infinity, = 0.550 micron) via a uniformly-illuminated cylindrical lens of dimensions a = 2b = 2 cm, f = 3 cm, n = 1.53.

The origin of the $x - y$ plane coincides with the lens focus. The longer side of the lens is parallel with the y-axis. Intensity resolution of contour plot: 0.05

[6]:
```
def I_rec(lambda_, x, y, a, b, I0):
    k = 2 * np.pi / lambda_
    factor1 = np.sin(k * x * a) / (k * x * a)
    factor2 = np.sin(k * y * b) / (k * y * b)
    return np.power(factor1, 2) * np.power(factor2, 2) * I0

I0 = 1
r = 0.5
a = 2
b = a/2
f = 3
n = 1.53
lambda_ = 0.550e-4
```

```python
n_points = 1000

x = np.linspace(-0.5, 0.5, n_points) * 1e-4
X, Y = np.meshgrid(x, x)

I_rectangle = I_rec(lambda_, X, Y, a, b, I0)
```

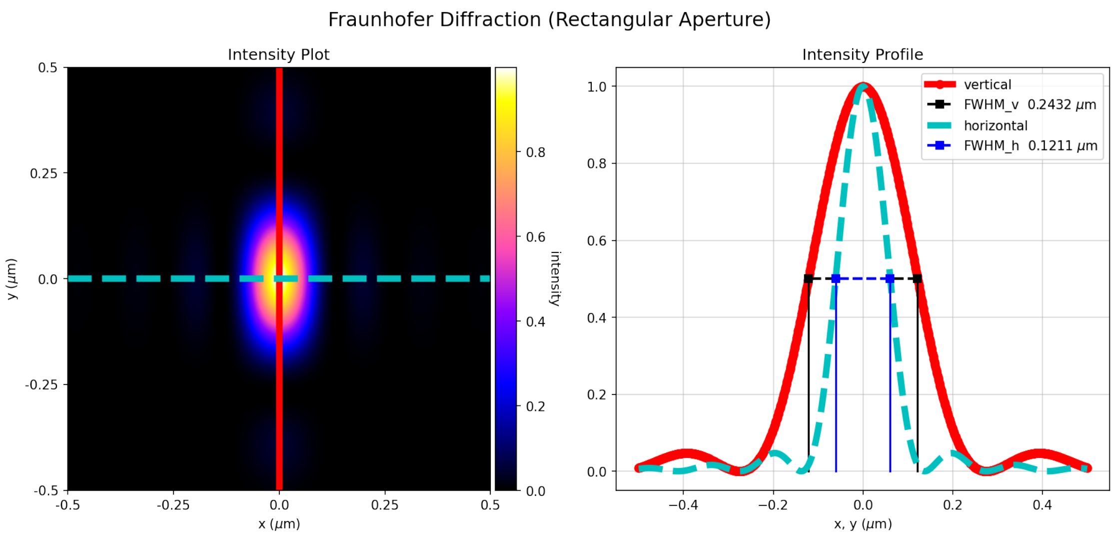
[7]:
```python
plt.figure(dpi = 150, figsize = (15,6))

plt.subplot(121)
plt.title('Intensity Plot')
# img1 = plt.imshow(np.log(I_rectangle), cmap='gnuplot2', origin='lower')
img1 = plt.imshow(I_rectangle, cmap='gnuplot2', origin='lower')
cbar1 = plt.colorbar(img1, fraction=0.045, pad=0.01)
cbar1.set_label('intensity', rotation=270, labelpad=10)
plt.xlabel('x ($\mu$m)')
plt.ylabel('y ($\mu$m)')
plt.xticks(ticks, ticklabels)
plt.yticks(ticks, ticklabels)

plt.subplot(122)
plt.title('Contour Plot')
plt.imshow(I_rectangle, cmap='gnuplot2', origin='lower')
img2 = plt.contour(I_rectangle, levels=np.arange(0, 1, 0.05), cmap='jet')
cbar2 = plt.colorbar(img2, fraction=0.045, pad=0.01)
cbar2.set_label('normalized intensity', rotation=270, labelpad=10)
plt.ylabel('y ($\mu$m)')
plt.xticks(ticks, ticklabels)
plt.yticks(ticks, ticklabels)

plt.suptitle('Fraunhofer Diffraction (Rectangular Aperture)', fontsize = 15)
plt.show()
```



[8]:

```
i_half = int(n_points/2)

plt.figure(dpi = 150, figsize = (15,6))

plt.subplot(121)
plt.title('Intensity Plot')
img1 = plt.imshow(I_rectangle, cmap='gnuplot2', origin='lower')
cbar1 = plt.colorbar(img1, fraction=0.045, pad=0.01)
cbar1.set_label('intensity', rotation=270, labelpad=10)
plt.xlabel('x ($\mu$m)')
plt.ylabel('y ($\mu$m)')
plt.xticks(ticks, ticklabels)
plt.yticks(ticks, ticklabels)
plt.axvline(i_half, color = 'r', ls = '-', lw = 5)
plt.axhline(i_half, color = 'c', ls = '--', lw = 5)

plt.subplot(122)
plt.title('Intensity Profile')
plt.grid(alpha = 0.5)
plt.plot(x/1e-4, I_rectangle[:,500], 'ro-', lw = 5, label = 'vertical')

x_r_cross = I_rectangle[:,500]
x_half = x[x_r_cross > 1/2]
x1 = np.max(x_half)/1e-4
x2 = np.min(x_half)/1e-4
FWHM_rec = x1-x2
plt.plot([x1,x1], [0,0.5], 'k')
plt.plot([x2,x2], [0,0.5], 'k')
plt.plot([x1,x2], [0.5,0.5], 'ks--', lw = 2, label = 'FWHM_v % .4f $\mu$m' %␣
↪FWHM_rec)

plt.plot(x/1e-4, I_rectangle[500, :], 'c--', lw = 5, label = 'horizontal')

x_r_cross = I_rectangle[500,:]
x_half = x[x_r_cross > 1/2]
x1 = np.max(x_half)/1e-4
x2 = np.min(x_half)/1e-4
FWHM_rec = x1-x2
plt.plot([x1,x1], [0,0.5], 'b')
plt.plot([x2,x2], [0,0.5], 'b')
plt.plot([x1,x2], [0.5,0.5], 'bs--', lw = 2, label = 'FWHM_h % .4f $\mu$m' %␣
↪FWHM_rec)

plt.xlabel('x, y ($\mu$m)')
```

```python
plt.legend()
plt.suptitle('Fraunhofer Diffraction (Rectangular Aperture)', fontsize = 15)

plt.show()
```



5 Problem 1.4.

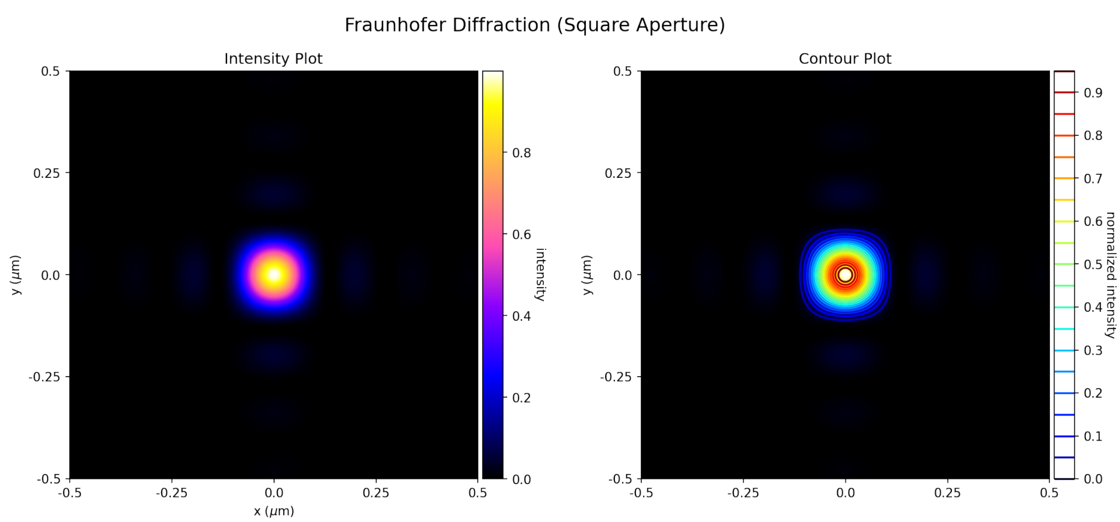
5.1 Fraunhofer diffraction pattern by a square aperture

(Section 8.5.1, Born & Wolf). Generate the normalized (I0 = 1) intensity contour plot I(x, y) that is produced by a point light source (located at infinity, = 0.550 micron) via a uniformly-illuminated cylindrical lens of dimensions a = b = 2 cm, f = 3 cm, n = 1.53. The origin of the x−y plane coincides with the lens focus. The longer side of the aperture is parallel with the y-axis. Intensity resolution of contour plot: 0.05

[9]:
```python
I0 = 1
r = 0.5
a = 2
b = 2
f = 3
n = 1.53
lambda_ = 0.550e-4
n_points = 1000

x = np.linspace(-0.5, 0.5, n_points) * 1e-4
X, Y = np.meshgrid(x, x)
```

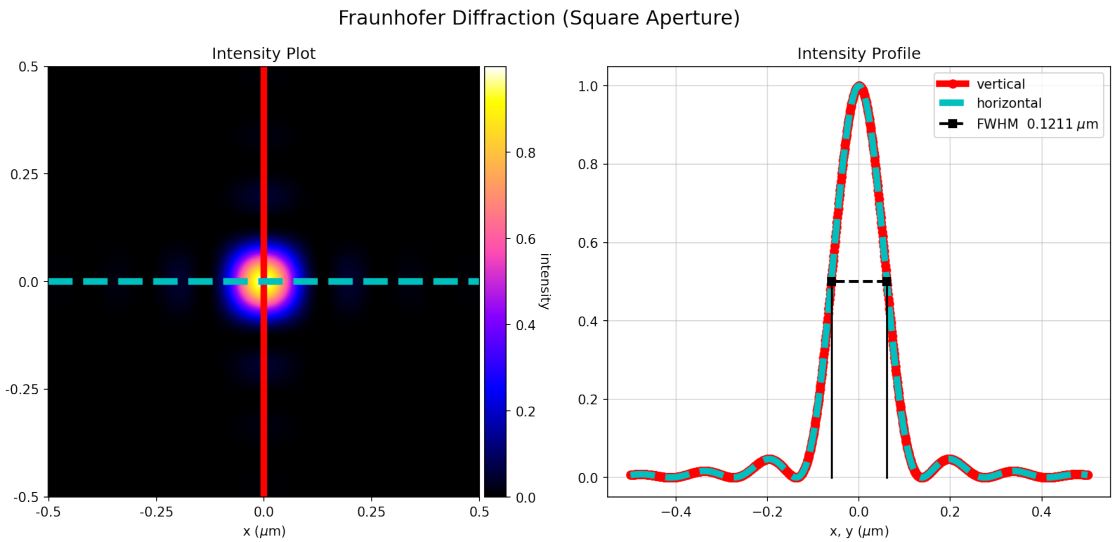
```
I_square = I_rec(lambda_, X, Y, a, b, I0)
```

[10]:
```
plt.figure(dpi = 150, figsize = (15,6))

plt.subplot(121)
plt.title('Intensity Plot')
img1 = plt.imshow(I_square, cmap='gnuplot2', origin='lower')
cbar1 = plt.colorbar(img1, fraction=0.045, pad=0.01)
cbar1.set_label('intensity', rotation=270, labelpad=10)
plt.xlabel('x ($\mu$m)')
plt.ylabel('y ($\mu$m)')
plt.xticks(ticks, ticklabels)
plt.yticks(ticks, ticklabels)

plt.subplot(122)
plt.title('Contour Plot')
plt.imshow(I_square, cmap='gnuplot2', origin='lower')
img2 = plt.contour(I_square, levels=np.arange(0, 1, 0.05), cmap='jet')
cbar2 = plt.colorbar(img2, fraction=0.045, pad=0.01)
cbar2.set_label('normalized intensity', rotation=270, labelpad=10)
plt.ylabel('y ($\mu$m)')
plt.xticks(ticks, ticklabels)
plt.yticks(ticks, ticklabels)

plt.suptitle('Fraunhofer Diffraction (Square Aperture)', fontsize = 15)
plt.show()
```



[11]:
```
i_half = int(n_points/2)
```

```python
plt.figure(dpi = 150, figsize = (15,6))

plt.subplot(121)
plt.title('Intensity Plot')
img1 = plt.imshow(I_square, cmap='gnuplot2', origin='lower')
cbar1 = plt.colorbar(img1, fraction=0.045, pad=0.01)
cbar1.set_label('intensity', rotation=270, labelpad=10)
plt.xlabel('x ($\mu$m)')
plt.xticks(ticks, ticklabels)
plt.yticks(ticks, ticklabels)
plt.axvline(i_half, color = 'r', ls = '-', lw = 5)
plt.axhline(i_half, color = 'c', ls = '--', lw = 5)

plt.subplot(122)
plt.title('Intensity Profile')
plt.grid(alpha = 0.5)
plt.plot(x/1e-4, I_square[:,500], 'ro-', lw = 5, label = 'vertical')
plt.plot(x/1e-4, I_square[500, :], 'c--', lw = 5, label = 'horizontal')

x_r_cross = I_square[:,500]
x_half = x[x_r_cross > 1/2]
x1 = np.max(x_half)/1e-4
x2 = np.min(x_half)/1e-4
FWHM_square = x1-x2
plt.plot([x1,x1], [0,0.5], 'k')
plt.plot([x2,x2], [0,0.5], 'k')
plt.plot([x1,x2], [0.5,0.5], 'ks--', lw = 2, label = 'FWHM % .4f $\mu$m' %␣
↪FWHM_square)


plt.xlabel('x, y ($\mu$m)')
plt.legend()
plt.suptitle('Fraunhofer Diffraction (Square Aperture)', fontsize = 15)

plt.show()
```



6 Problem 8.5

For the same aperture size, plot the diagonal FWHM of the central spot as a function of the ratio a/b where: 0.1 a/b 10 (512 data points) and = 0.550 micron. Discuss briefly the salient features and implications of your results.