hw8

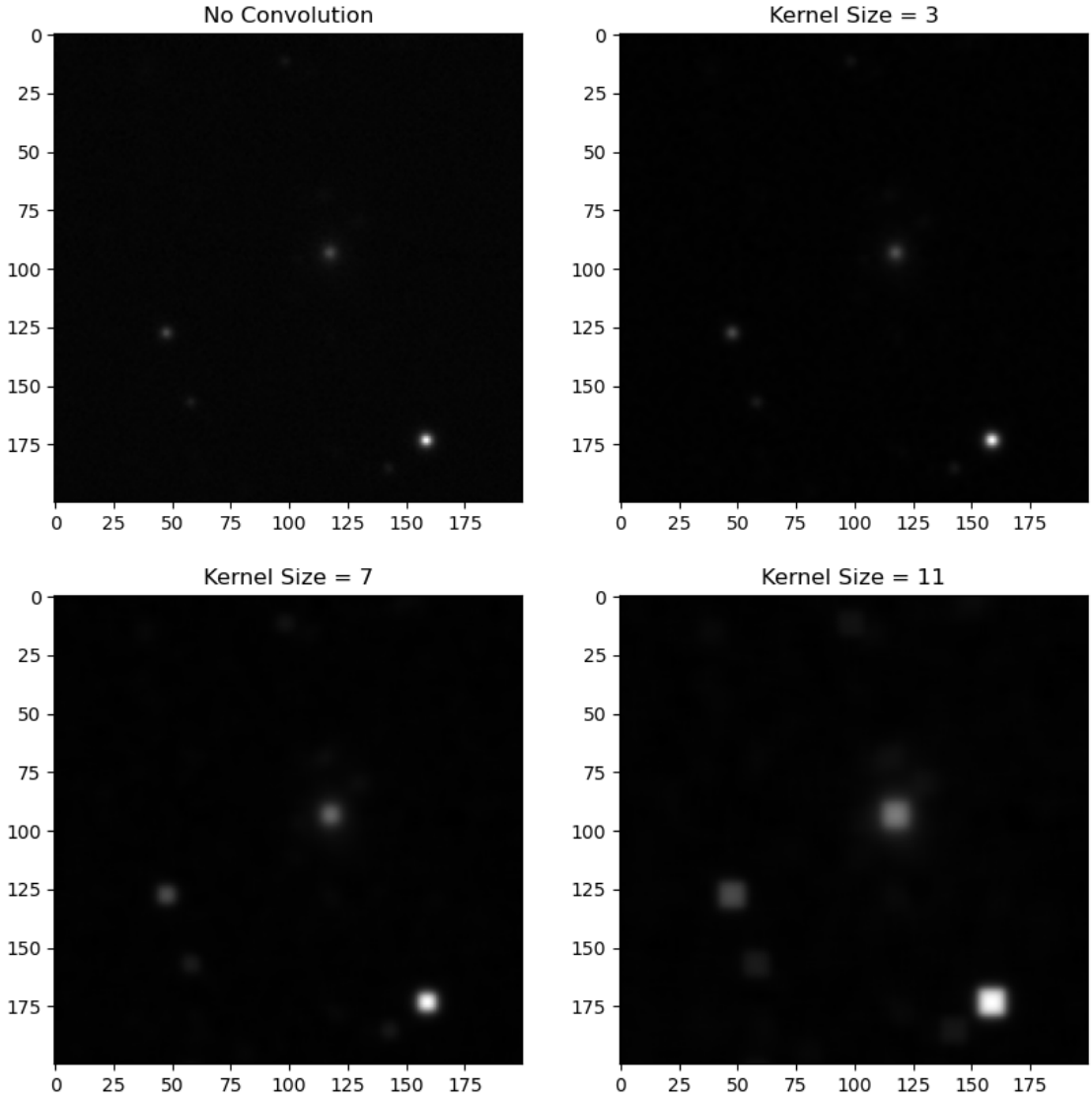
April 5, 2023

[ ]:
```
from astropy.io import fits
from astropy.convolution import Box2DKernel, convolve
import matplotlib.pyplot as plt
```

[ ]:
```
r = fits.open("DESJ053816.9-503050.8_r.fits")[0].data
boxed3 = convolve(r, Box2DKernel(3))
boxed7 = convolve(r, Box2DKernel(7))
boxed11 = convolve(r, Box2DKernel(11))

x, y = r.shape
r_center = r[x // 2 - 100 : x // 2 + 100, y // 2 - 100 : y // 2 + 100]
boxed3_center = boxed3[x // 2 - 100 : x // 2 + 100, y // 2 - 100 : y // 2 + 100]
boxed7_center = boxed7[x // 2 - 100 : x // 2 + 100, y // 2 - 100 : y // 2 + 100]
boxed11_center = boxed11[x // 2 - 100 : x // 2 + 100, y // 2 - 100 : y // 2 +␣
↪100]
```

[ ]:
```
fig, axes = plt.subplots(nrows = 2, ncols = 2, figsize = (10,10))
axes[0][0].imshow(r_center, cmap='gray')
axes[0][0].set_title("No Convolution")
axes[0][1].imshow(boxed3_center, cmap='gray')
axes[0][1].set_title("Kernel Size = 3")
axes[1][0].imshow(boxed7_center, cmap='gray')
axes[1][0].set_title("Kernel Size = 7")
axes[1][1].imshow(boxed11_center, cmap='gray')
axes[1][1].set_title("Kernel Size = 11")
plt.show()
```



2a

$\frac{1}{R} = \frac{\Delta\lambda}{\lambda} = \frac{V}{c}$

$V = \frac{c}{R} = \frac{3\times10^8 m\cdot s^{-1}}{2000}$

velocity resolution $= 1.5 \times 10^5 m \cdot s^{-1}$

2b

$\Delta\lambda = 3728.8 - 3726.1 = 2.7m$

$v = 3 \times 10^8 m \cdot s^{-1} \cdot \frac{2.7m}{3726.1m}$

$v = 217.385\ km \cdot s^{-1} = 2.17385 \times 10^5 m \cdot s^{-1}$

Yes, we can resolve it.