Name: C. J. Kurukulasuriya

Index Number: 190337X

In [ ]:

```
%matplotlib inline
```

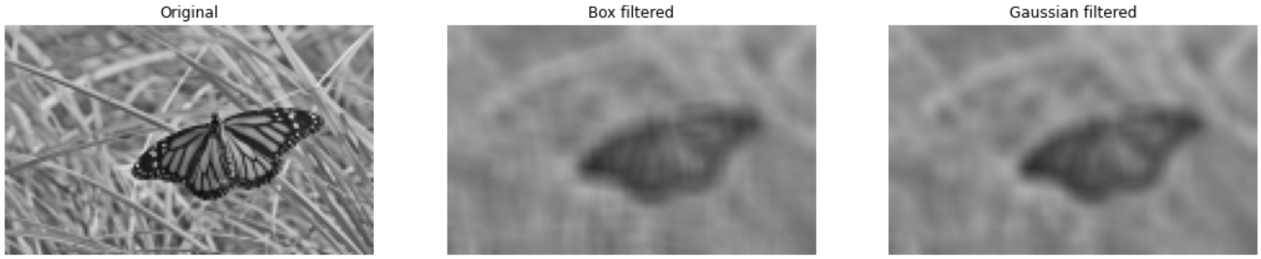
In [ ]:

```
import numpy as np
import cv2 as cv
import matplotlib.pyplot as plt
```

Q1

In [ ]:

```
im = cv.imread(r'Images/butterfly.jpg', cv.IMREAD_REDUCED_GRAYSCALE_8).astype(np.float3
assert im is not None

k_size = 9
sigma = 4
box_kernel = 1.0/81 * np.ones((k_size, k_size))
im_avg = cv.filter2D(im, -1, box_kernel)
im_gauss = cv.GaussianBlur(im,(k_size, k_size),sigma)

fig, ax = plt.subplots(1,3, figsize=(18,6))
ax[0].imshow(im, cmap = 'gray', vmin = 0, vmax = 255)
ax[0].set_title('Original')
ax[1].imshow(im_avg, cmap = 'gray', vmin = 0, vmax = 255)
ax[1].set_title('Box filtered')
ax[2].imshow(im_gauss, cmap = 'gray', vmin = 0, vmax = 255)
ax[2].set_title('Gaussian filtered')

for i in range(3):
    ax[i].axis('off')
plt.show()
```



Q2

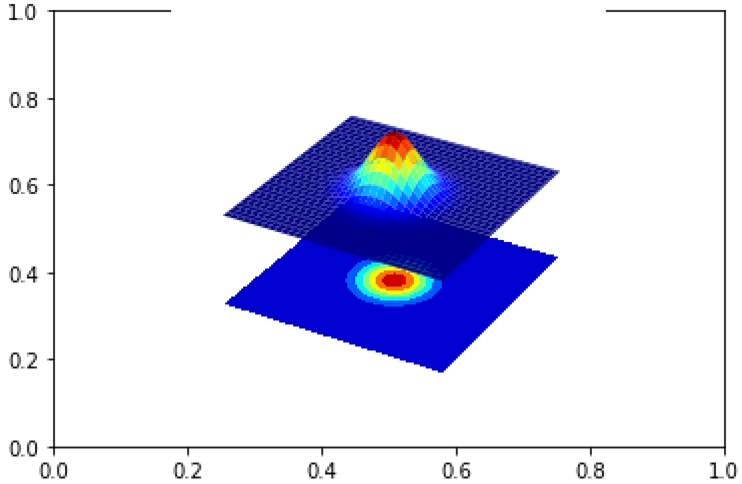
In [ ]:

```
from matplotlib import cm
```

In [ ]:

```
fig, ax = plt.subplots()
ax = fig.add_subplot(111, projection = '3d')

step = 0.1
X = np.arange(-5, 5+step, step)
```

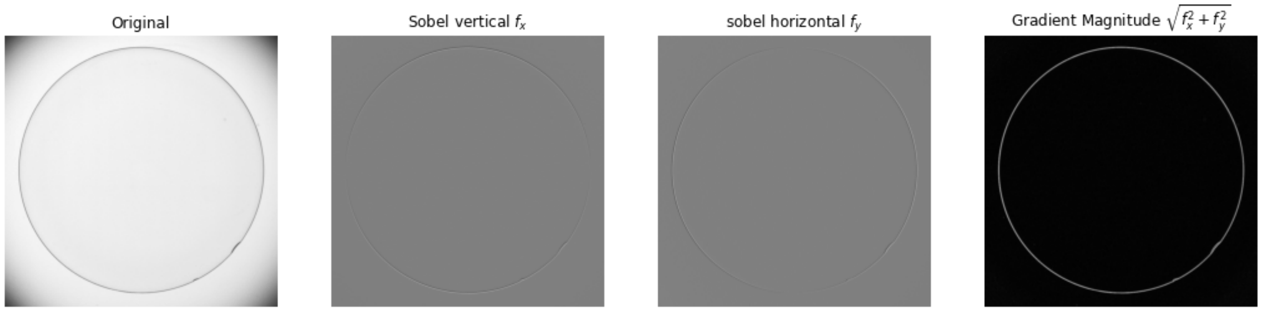
```
Y = np.arange(-5, 5+step, step)
XX, YY =np.meshgrid(X, Y)
sigma = 1
g = np.exp(-(XX**2+YY**2)/2*sigma**2)

surf = ax.plot_surface(XX, YY, g, cmap = cm.jet)

cset = ax.contourf(XX,YY, g, zdir='z', offset = np.min(g) - 1.5, cmap = cm.jet)
ax.set_zlim(np.min(g)-2, np.max(g))
plt.axis("off")
plt.show()
```



Q3

In [ ]:

```
im = cv.imread(r'Images/contact_lens.tif', cv.IMREAD_GRAYSCALE).astype(np.float32)
assert im is not None

sobel_v = np.array([[-1, -2, -1] , [0, 0, 0] , [1, 2, 1]] , dtype = np.float32 )
im_sobelv = cv.filter2D(im, -1, sobel_v )

sobel_h = np.array([[-1, 0, 1] , [-2, 0, 2] , [-1, 0, 1]] , dtype = np.float32 )
im_sobelh = cv.filter2D(im, -1, sobel_h )
grad_mag = np.sqrt(im_sobelv**2+im_sobelh**2)

fig, ax = plt.subplots(1,4, figsize=(18,6))

ax[0].imshow(im, cmap = 'gray', vmin = 0, vmax = 255)
ax[0].set_title('Original')

ax[1].imshow(im_sobelv, cmap = 'gray', vmin = -1020, vmax = 1020)
ax[1].set_title('Sobel vertical $f_x$')

ax[2].imshow(im_sobelh, cmap = 'gray', vmin = -1020, vmax = 1020)
ax[2].set_title('sobel horizontal $f_y$')

ax[3].imshow(grad_mag, cmap = 'gray')
ax[3].set_title("Gradient Magnitude $\sqrt{f_x^2 + f_y^2}$")

for i in range(4):
    ax[i].axis('off')

plt.show()
```



Q4

In [ ]:

```
f = cv.imread(r'Images/tom.jpg', cv.IMREAD_GRAYSCALE).astype(np.float32)
assert im is not None

sigma = 2
gaussian_1d = cv.getGaussianKernel(5, sigma=sigma)
f_lp = cv.sepFilter2D(f, -1, gaussian_1d, gaussian_1d)
f_hp = f - f_lp
f_sharp = cv.addWeighted(f, 1.0, f_hp, 2.0, 0)

fig, ax = plt.subplots(1,4, figsize=(18,6))

ax[0].imshow(f, cmap = 'gray', vmin = 0, vmax = 255)
ax[0].set_title('Original')

ax[1].imshow(f_lp, cmap = 'gray')
ax[1].set_title('$f_{lp}$')

ax[2].imshow(f_hp, cmap = 'gray', vmin = -1020, vmax = 1020)
ax[2].set_title('$f_{hp}$')

ax[3].imshow(f_sharp, cmap = 'gray')
ax[3].set_title("Sharpened")

for i in range(4):
    ax[i].axis('off')
plt.show()
```