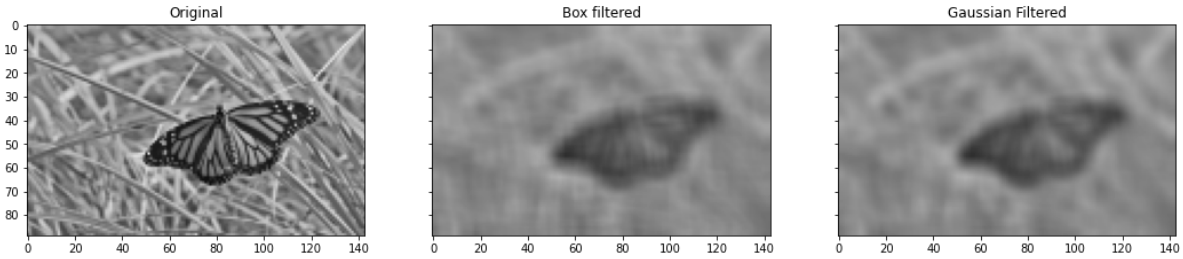
Name : Vakeesan.K

Index No. : 190643G

1)

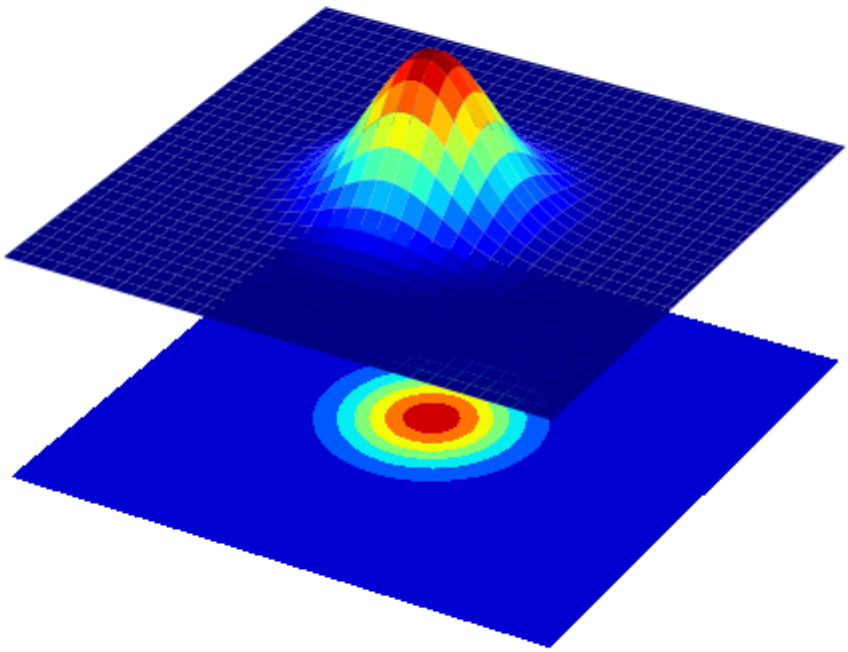
In [ ]:
```
%matplotlib inline
import cv2 as cv
import numpy as np
from matplotlib.pyplot as plt

img = cv.imread(r'C:\Python39\cv\exercices\lec 3\butterfly.jpg',cv.IMREAD_REDUCED_(
assert img is not None

k_size = 9
sigma = 4
kernel = np.ones((9,9),np.float32)/81

imgc = cv.filter2D(img,-1,kernel)

imgb = cv.GaussianBlur(img,(k_size,k_size),sigma)

fig,axes = plt.subplots(1,3,sharex='all', sharey='all',figsize=(18,6))
axes[0].imshow(img,cmap='gray',vmin=0,vmax=255)
axes[0].set_title('Original')
axes[1].imshow(imgc,cmap='gray',vmin=0,vmax=255)
axes[1].set_title('Box filtered')

axes[2].imshow(imgb,cmap='gray',vmin=0,vmax=255)
axes[2].set_title('Gaussian Filtered')
plt.show()
```



2)

In [ ]:
```
%matplotlib inline
import cv2 as cv
import numpy as np
import matplotlib.pyplot as plt
from mpl_toolkits.mplot3d import Axes3D
from matplotlib import cm


fig=plt.figure(figsize=(10,10))
ax = fig.add_subplot(111,projection='3d')

step=0.1
sigma=1

X = np.arange(-5,5 + step,step)
Y = np.arange(-5,5 + step,step)
XX, YY = np.meshgrid(X,Y)
```

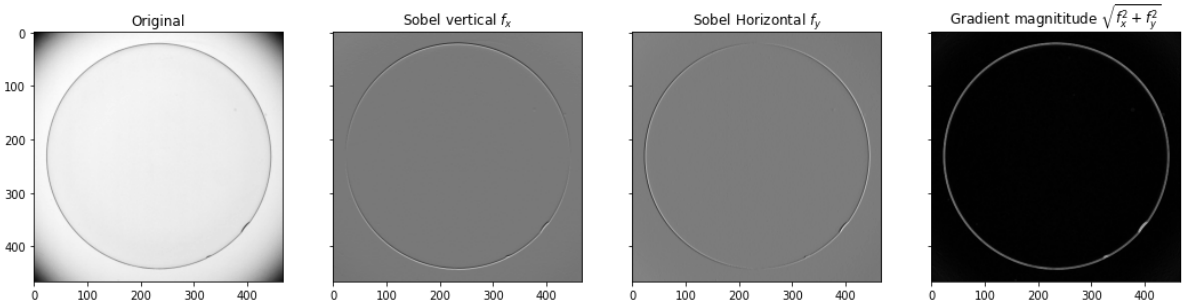
```
g = np.exp(-(XX**2 + YY**2)/(2*sigma**2))



# Plot the surface.

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


ax.zaxis.set_major_locator(LinearLocator(10))
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()
```



3)

In [ ]:
```
%matplotlib inline
import cv2 as cv
import numpy as np
from matplotlib import pyplot as plt
import matplotlib.image as mp_img

img = cv.imread(r'C:\Python39\cv\exercices\lec 3\contact_lens.tif',cv.IMREAD_REDUCE
assert img is not None

#sobel vertical
```

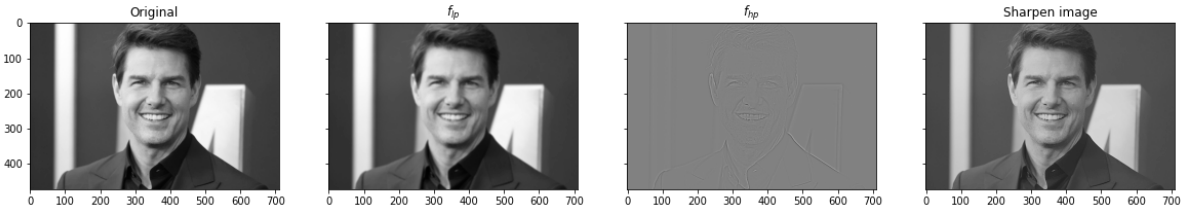
```
kernel_v=np.array([(-1,-2,-1),(0,0,0),(1,2,1)], dtype=np.float32)
imgv = cv.filter2D(img,-1,kernel_v)

#sobel horizontal
kernel_h=np.array([(-1,0,1),(-2,0,2),(-1,0,1)], dtype=np.float32)
imgh = cv.filter2D(img,-1,kernel_h)

grad_mag = np.sqrt(imgv**2+imgh**2)

fig,axes = plt.subplots(1,4,sharex='all', sharey='all',figsize=(18,6))
axes[0].imshow(img,cmap='gray')
axes[0].set_title('Original')
axes[1].imshow(imgv,cmap='gray')
axes[1].set_title('Sobel vertical $f_x$')
axes[2].imshow(imgh,cmap='gray')
axes[2].set_title('Sobel Horizontal $f_y$')
axes[3].imshow(grad_mag,cmap='gray')
axes[3].set_title('Gradient magnititude $\sqrt{f_x^2 + f_y^2}$')

plt.show()
```



4)

In [ ]:

```
%matplotlib inline
import cv2 as cv
import numpy as np
from matplotlib import pyplot as plt
import matplotlib.image as mp_img

img = cv.imread(r'C:\Python39\cv\exercices\lec 3\tom.jpg',cv.IMREAD_GRAYSCALE).asty
assert img is not None

#sharpening
kernel=np.array([(0,-1,0),(-1,5,-1),(0,-1,0)], dtype='float')
imgs = cv.filter2D(img,-1,kernel)

sigma =2
gaussian_1d = cv.getGaussianKernel(5,sigma)
f_lp= cv.sepFilter2D(img,-1,gaussian_1d,gaussian_1d)
f_hp= img - f_lp
sharpen= cv.addWeighted(img,1.0,f_hp,1.0,5)

fig,axes = plt.subplots(1,4,sharex='all', sharey='all',figsize=(20,5))
axes[0].imshow(img,cmap='gray')
axes[0].set_title('Original')
axes[1].imshow(f_lp,cmap='gray')
axes[1].set_title('$f_{lp}$')
axes[2].imshow(f_hp,cmap='gray')
axes[2].set_title('$f_{hp}$')
axes[3].imshow(sharpen,cmap='gray')
axes[3].set_title('Sharpen image')
plt.show()
```



In [ ]: