

Name: Vakeesan

Index No. 190643G

1)

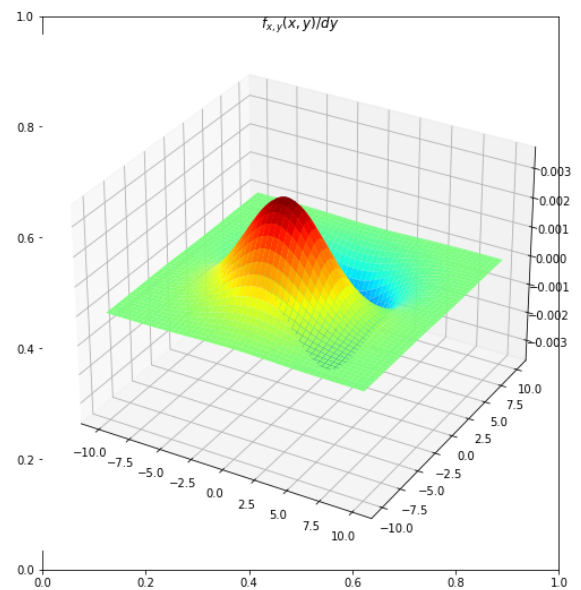
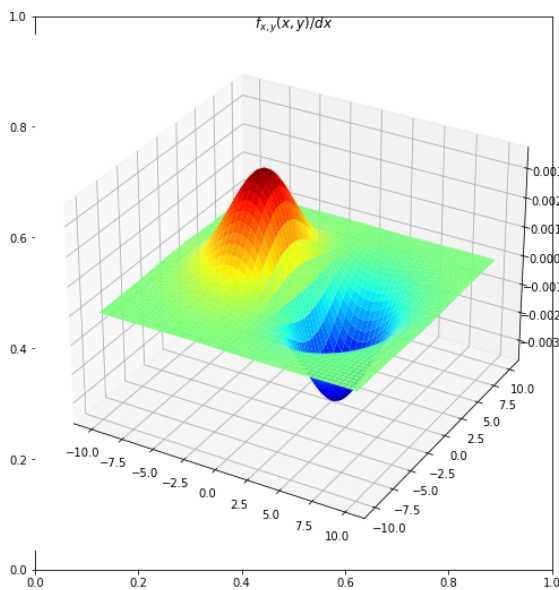
```
In [ ]: from matplotlib import cm
import numpy as np
import matplotlib.pyplot as plt

sigma=3
x=np.linspace(-10,10, num=100)
y=np.linspace(-10,10, num=100)

x, y = np.meshgrid(x, y)

zx = -(x/(2*np.pi*sigma**4))*np.exp(-(x**2+y**2)/(2*sigma**2))
zy=-(y/(2*np.pi*sigma**4))*np.exp(-(x**2+y**2)/(2*sigma**2))
fig,ax = plt.subplots(1,2,figsize=(18,9))
ax = fig.add_subplot(121, projection='3d')
ax.plot_surface(x,y,zx, cmap=cm.jet)
ax.set_title("$f_{x,y}(x,y)/dx$")

ax2 = fig.add_subplot(122, projection='3d')
ax2.plot_surface(x,y,zy, cmap=cm.jet)
ax2.set_title("$f_{x,y}(x,y)/dy$")
plt.show()
```



2)

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

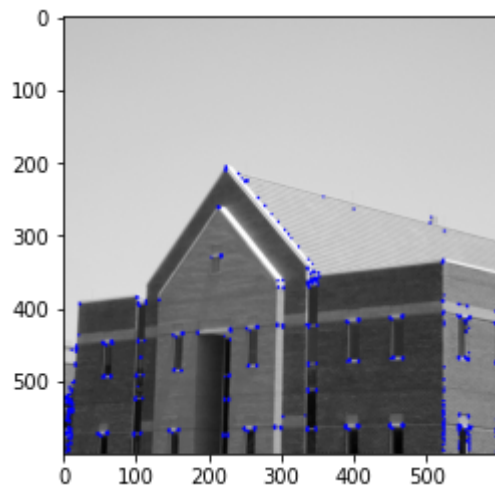
img = cv.imread(r'C:\Python39\cv\exercices\lec 4\building.tif',cv.IMREAD_COLOR)
gray=cv.cvtColor(img,cv.COLOR_BGR2GRAY)
gray=np.float32(gray)
assert img is not None
dst = cv.cornerHarris(gray,2,3,0.04)
#result is dilated for marking the corners, not important
```

```

dst = cv.dilate(dst, None)
# Threshold for an optimal value, it may vary depending on the image.
img[dst > 0.01*dst.max()]=[0,0,255]

plt.imshow(img, cmap='gray')
plt.show()

```



```

In [ ]: %matplotlib inline
from matplotlib import cm
import numpy as np
import cv2 as cv
import matplotlib.pyplot as plt
from skimage.feature import peak_local_max

img = cv.imread(r'C:\Python39\cv\exercices\lec 4\building.tif', cv.IMREAD_COLOR)
assert img is not None
I=cv.cvtColor(img, cv.COLOR_BGR2GRAY)
I=np.float32(I)

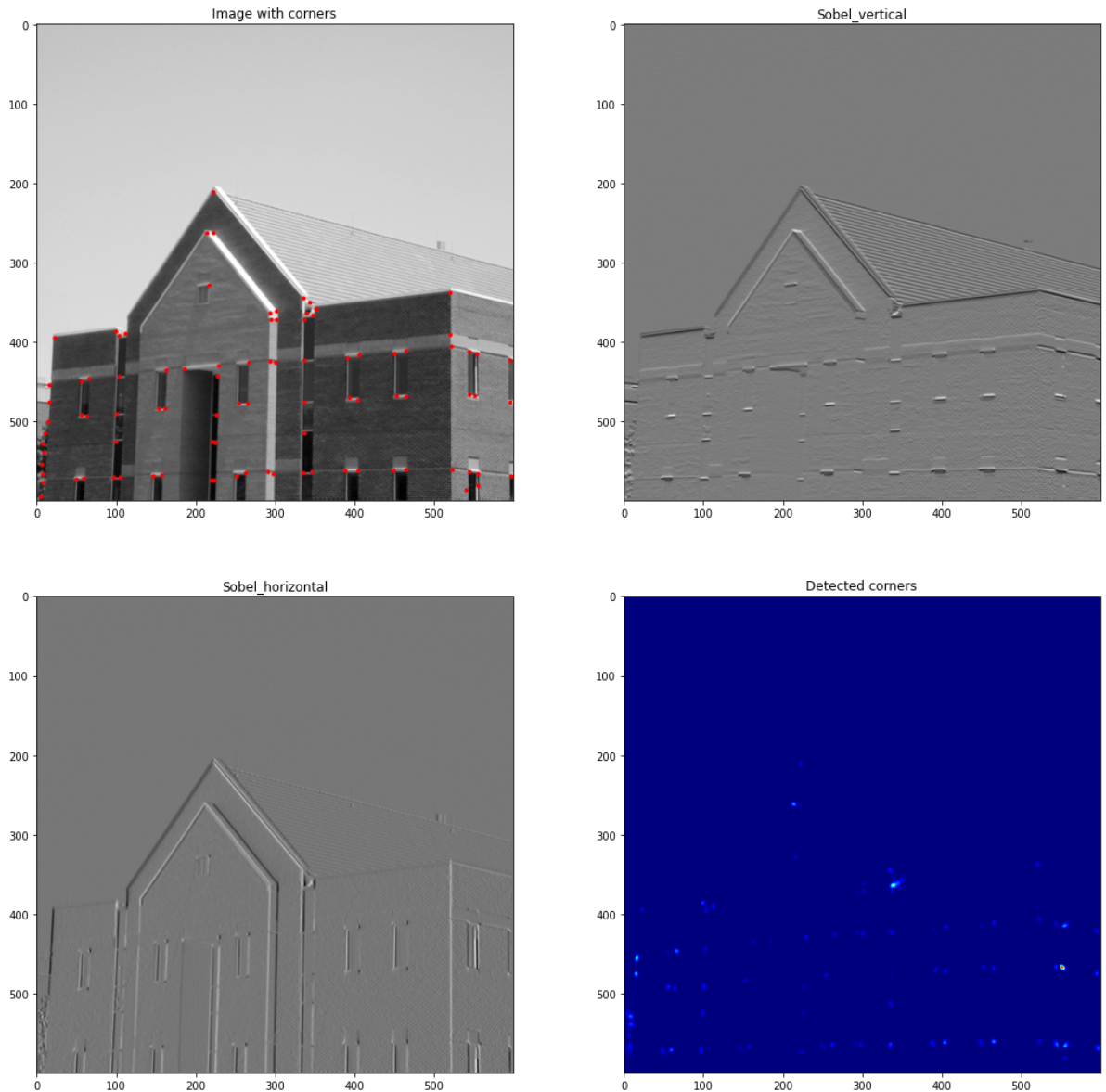
sobel_v=np.array([[ -1, -2, -1],[0,0,0],[1,2,1]])
sobel_h=np.array([[ -1, 0, 1],[-2,0,2],[ -1,0,1]])
Ix =cv.filter2D(I, -1, sobel_v)
Iy =cv.filter2D(I, -1, sobel_h)

sigma=3
ksize=7
m11 =cv.GaussianBlur(Ix**2, (ksize,ksize), sigma)
m12=cv.GaussianBlur(Ix*Iy, (ksize,ksize), sigma)
m21=m12
m22=cv.GaussianBlur(Iy**2, (ksize,ksize), sigma)

det = m11*m22-m12*m21
trace=m11 + m22
alpha=0.04
R=det-alpha*trace**2
R[R< 1e8]=0
coordinate=peak_local_max(R, min_distance=2)
fig,ax=plt.subplots(2,2,figsize=(18,18))
ax[0][0].imshow(img, cmap='gray')
ax[0][0].set_title("Image with corners")
ax[0][0].plot(coordinate[:,1], coordinate[:,0], 'r.')
ax[0][1].imshow(Ix+127, cmap='gray')
ax[0][1].set_title("Sobel_vertical")
ax[1][0].imshow(Iy+127, cmap='gray')
ax[1][0].set_title("Sobel_horizontal")
ax[1][1].imshow(R+127, cmap=cm.jet)

```

```
ax[1][1].set_title("Detected corners")
plt.show()
```

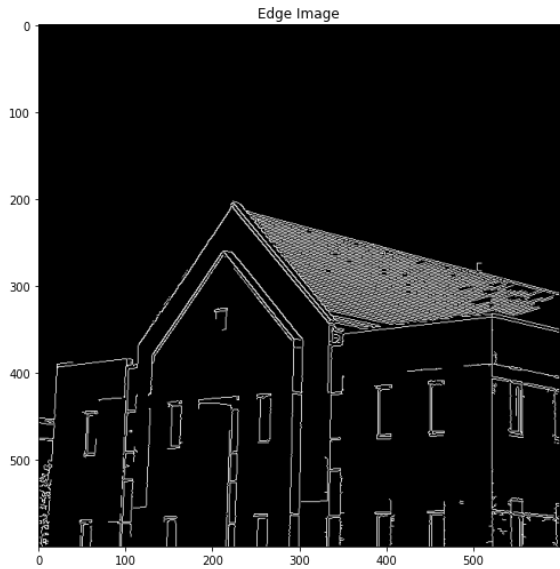
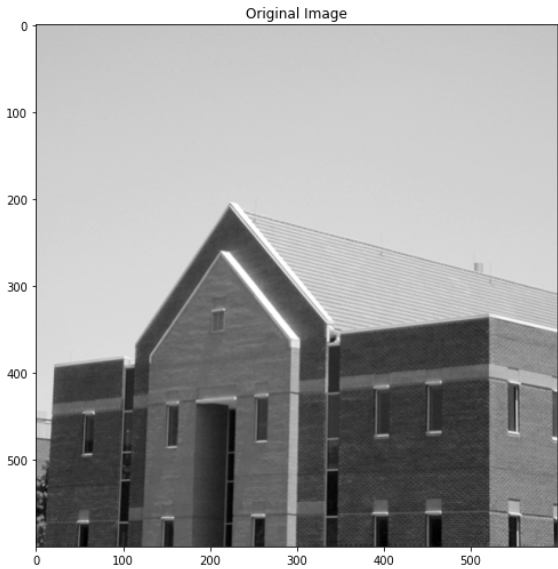


4)

```
In [ ]: import numpy as np
import cv2 as cv
from matplotlib import pyplot as plt
img = cv.imread(r'C:\Python39\cv\exercices\lec 4\building.tif',0)
edges = cv.Canny(img,100,200)
fig,ax=plt.subplots(1,2,figsize=(18,9))
ax[0].imshow(img,cmap = 'gray')
ax[0].set_title('Original Image')

ax[1].imshow(edges,cmap = 'gray')
ax[1].set_title('Edge Image')

plt.show()
```



```
In [ ]:
```