

```
In [80]: import numpy as np
import cv2 as cv
from matplotlib import pyplot as plt
from IPython import display

# functions
def out (str):
    return ("out/out_" + str + ".jpg")

# variable initialisation

# open image
original = cv.imread('opencv_logo.png')

# define kernels
# full 3x3
kernel3x3 = np.array([[1., 2., 1.], [2., 4., 2.], [1., 2., 1.]])/16
# moving average box filter kernel
kernelaverage = np.array([[1., 1., 1.], [1., 1., 1.], [1., 1., 1.]])/9
# 1D 3x1
kernel3x1 = np.array([[1.], [2.], [1.]])/4
# 1D 1x3
kernel1x3 = np.array([[1., 2., 1.]])/4

# question 3 perform gaussian and box filtering
gaussian2d = cv.filter2D(original, -1, kernel3x3)
box = cv.filter2D(original, -1, kernelaverage)

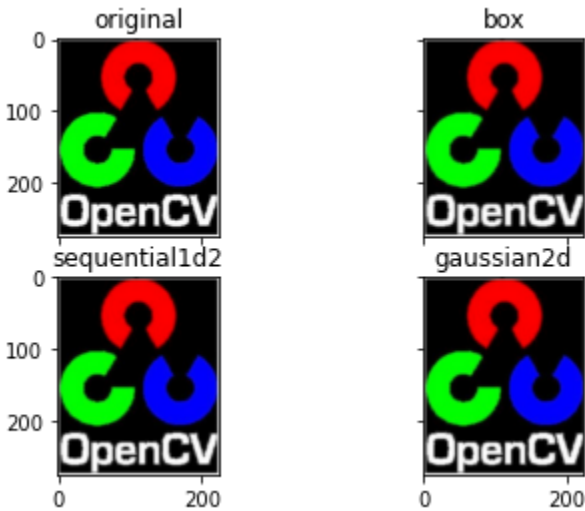
# question 4 perfrom sequential 1d guassian filtering and verify that it is the same
gaussian1d = cv.filter2D(original, -1, kernel1x3)
gaussian1d2 = cv.filter2D(gaussian1d, -1, kernel3x1)

# display in console
fig, axs = plt.subplots(2, 2)
axs[0, 0].imshow(original)
axs[0, 0].set_title('original')
axs[0, 1].imshow(box)
axs[0, 1].set_title('box')
axs[1, 0].imshow(gaussian1d2)
axs[1, 0].set_title('sequential1d2')
axs[1, 1].imshow(gaussian2d)
axs[1, 1].set_title('gaussian2d')

# Hide labels
for ax in axs.flat:
    ax.label_outer()

plt.show()

# output the filtered images
cv2.imwrite(out("original"), original)
cv2.imwrite(out("gaussian2d"), gaussian2d)
cv2.imwrite(out("box"), box)
cv2.imwrite(out("gaussian1d2"), gaussian1d2)
```



False

```
In [ ]:
```

```
In [ ]:
```

```
In [ ]:
```