CNN_Convolution_n_Pooling

```
#import
import cv2
import numpy as np
from scipy import misc
i = misc.ascent()
#i = misc.face
```

```
<ipython-input-8-81001d7af789>:5: DeprecationWarning: scipy.misc.ascent has been deprecated in SciPy v1.10.0; and will be complet
  i = misc.ascent()
```

```
i_transformed = np.copy(i)
size_x = i_transformed.shape[0]
size_y = i_transformed.shape[1]
```

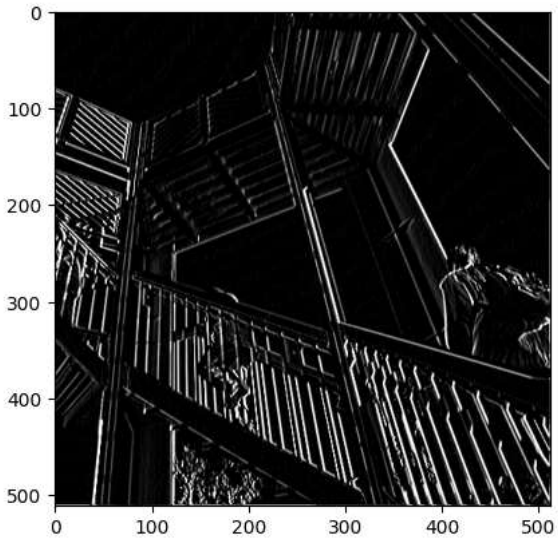
```
#plot
import matplotlib.pyplot as plt
plt.grid(False)
plt.gray()
plt.axis('off')
plt.imshow(i)
plt.show()
```

```
#filter1
#filter = [[0,1,0],[1,-4,1],[0,1,0]]

#filter2
filter = [[-1,-2,-1],[0,0,0],[1,2,1]]
#filter = [[-1,0,1],[-2,0,2],[-1,0,1]]
weight = 1
```

```
for x in range(1,size_x-1):
  for y in range(1,size_y-1):
    convolution = 0.0
    convolution = convolution + (i[x - 1,y-1]*filter[0][0])
    convolution = convolution + (i[x,y-1]*filter[0][1])
    convolution = convolution + (i[x + 1,y-1]*filter[0][2])
    convolution = convolution + (i[x - 1,y]*filter[1][0])
    convolution = convolution + (i[x,y]*filter[1][1])
    convolution = convolution + (i[x + 1,y]*filter[1][2])
    convolution = convolution + (i[x - 1,y+1]*filter[2][0])
    convolution = convolution + (i[x,y+1]*filter[2][1])
    convolution = convolution + (i[x + 1,y+1]*filter[2][2])
    convolution = convolution * weight
    if(convolution<0):
      convolution=0
    if(convolution>255):
      convolution=255
    i_transformed[x,y] = convolution
```

```
#let's draw transformed image
plt.gray()
plt.grid(False)
plt.imshow(i_transformed)
plt.show()
```




```
#pooling example
new_x = int(size_x/2)
new_y = int(size_y/2)
newImage = np.zeros((new_x, new_y))
for x in range(0, size_x,2):
  for y in range(0, size_y,2):
    pixels = []
    pixels.append(i_transformed[x,y])
    pixels.append(i_transformed[x+1,y])
    pixels.append(i_transformed[x,y+1])
    pixels.append(i_transformed[x+1,y+1])
    newImage[int(x/2),int(y/2)] = max(pixels)

print(newImage.shape)
#plot the image. Note the size of the axess --new 256 pixels instead of 512
plt.grid(False)
plt.gray()
#plt.axis('off')
plt.imshow(newImage)
plt.show()
```

```
(256, 256)
```