

```
from google.colab import drive
```

```
# Mount google drive
drive.mount('/content/drive')
```

↗ Mounted at /content/drive

```
import numpy as np
import matplotlib.pyplot as plt
from PIL import Image
```

```
# Functions to perform Convolution
```

```
def convolve2d(image,kernel):
    kernel_height, kernel_width = kernel.shape
    image_height, image_width = image.shape

    output_height = image_height - kernel_height + 1
    output_width = image_width - kernel_width + 1

    output = np.zeros((output_height, output_width))

    for i in range(output_height):
        for j in range(output_width):
            patch = image[i:i+kernel_height, j:j+kernel_width]
            output[i,j] = np.sum(patch * kernel)

    return output
```

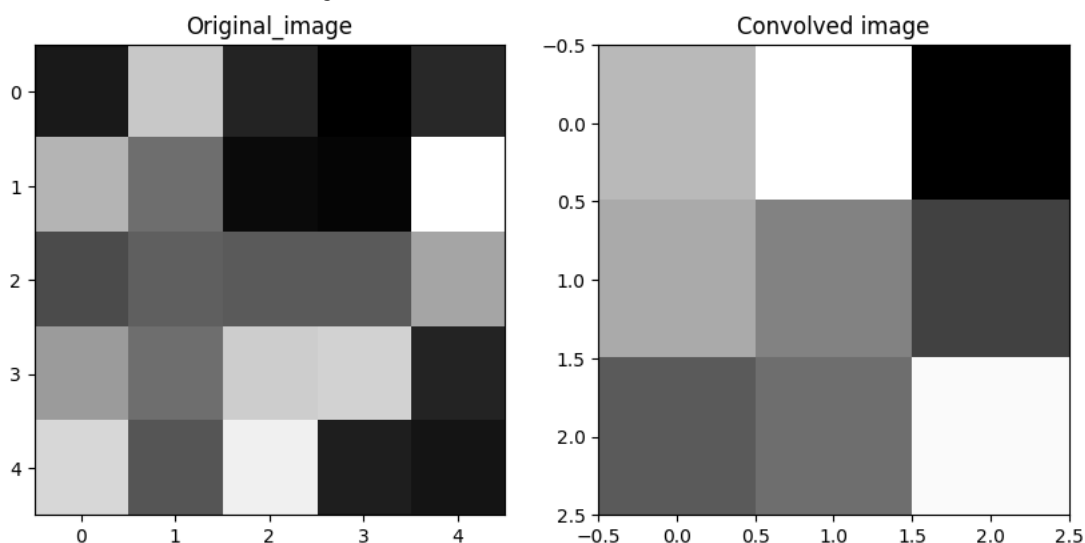
```
image = np.random.rand(5,5)
```

```
# Define a kernel (eg Edge detection)
kernel = np.array([[1,0,-1],
                   [1,0,-1],
                   [1,0,-1]])
```

```
# Apply Convolution
convolved_image = convolve2d(image, kernel)
```

```
# Plot the results
fig, axes = plt.subplots(1,2, figsize=(10,15))
axes[0].imshow(image,cmap='gray')
axes[0].set_title('Original_image')
axes[1].imshow(convolved_image,cmap='gray')
axes[1].set_title('Convolved image')
```

↗ Text(0.5, 1.0, 'Convolved image')



image

↗ array([[0.1309493 , 0.43041803, 0.91540936, 0.82419257, 0.52509212],
 [0.68948765, 0.18852429, 0.01044373, 0.91667327, 0.91336859],
 [0.01777695, 0.2715264 , 0.32433849, 0.2588693 , 0.37757316],
 [0.26018775, 0.72616846, 0.07894914, 0.83073501, 0.54901808],
 [0.87913607, 0.43225283, 0.88473155, 0.721969 , 0.06031682]])

```
image_path = '/content/drive/MyDrive/40_datascience_project/Day2 Dog Breed Prediction/7.jpg'
image = Image.open(image_path).convert('L') #Convert to grayscale
image = np.array(image)
```

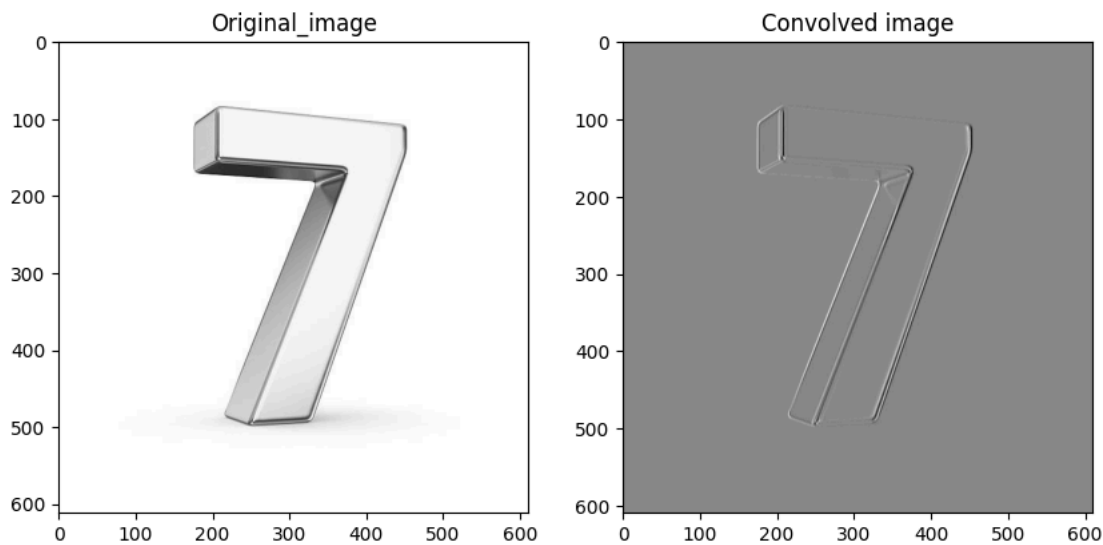
```
# image = np.random.rand(5,5)
```

```
# Define a kernel (eg Edge detection)
kernel = np.array([[1,0,-1],
                  [1,0,-1],
                  [1,0,-1]])
```

```
# Apply Convolution
convolved_image = convolve2d(image, kernel)
```

```
# Plot the results
fig, axes = plt.subplots(1,2, figsize=(10,15))
axes[0].imshow(image,cmap='gray')
axes[0].set_title('Original_image')
axes[1].imshow(convolved_image,cmap='gray')
axes[1].set_title('Convolved image')
```

```
Text(0.5, 1.0, 'Convolved image')
```



```
# image_path = '/content/drive/MyDrive/40_datascience_project/Day2 Dog Breed Prediction/biswash pp.jpg'
# image_path = '/content/drive/MyDrive/40_datascience_project/Day2 Dog Breed Prediction/20230121_190547.jpg'
image = Image.open(image_path).convert('L') #Convert to grayscale
image = np.array(image)
```

```
# image = np.random.rand(5,5)
```

```
# Define a kernel (eg Edge detection)
kernel = np.array([[1,0,-1],
                  [1,0,-1],
                  [1,0,-1]])
```

```
# Apply Convolution
convolved_image = convolve2d(image, kernel)
```

```
# Plot the results
fig, axes = plt.subplots(1,2, figsize=(10,15))
axes[0].imshow(image,cmap='gray')
axes[0].set_title('Original_image')
axes[1].imshow(convolved_image,cmap='gray')
axes[1].set_title('Convolved image')
```

