

```
In [1]: import matplotlib.pyplot as plt
import numpy as np
import cv2 as cv
```

Reading Image

```
In [2]: def read_image():
    path = "/media/rifat/STUDY/4-1/LAB/Image_Processing/image/tiger.jpeg"
    img = plt.imread(path)
    plt.title("Original image")
    plt.imshow(img)
    grayscale = cv.cvtColor(img,cv.COLOR_RGB2GRAY)

    return img, grayscale
```

Building Kernel

```
In [18]: def kernel_build():
    kernel1 = np.array([[0, 1, 0], [1, 1, 1], [0, 1, 0]])
    kernel2 = np.array([[1, 0, 0], [0, 1, 0], [0, 0, 1]])
    kernel3 = np.array([[2, 1, 0], [1, 0, -1], [0, 1, -2]])
    kernel4 = np.array([[ -1, -1, -1], [ -1, 8, -1], [ -1, -1, -1]])
    kernel5 = np.array([[1, 2, 1], [0, 0, 0], [ -1, -2, -1]])
    kernel6 = np.array([[ -1, -1, 2], [ -1, 2, -1], [ -1, -1, 2]])
    kernel = [kernel1, kernel2, kernel3, kernel4, kernel5, kernel6]
    return kernel
```

Applying Kernel Over image

```
In [19]: def image_build(grayscale, kernel):
    image1 = cv.filter2D(grayscale, -1, kernel[0])
    image2 = cv.filter2D(grayscale, -1, kernel[1])
    image3 = cv.filter2D(grayscale, -1, kernel[2])
    image4 = cv.filter2D(grayscale, -1, kernel[3])
    image5 = cv.filter2D(grayscale, -1, kernel[4])
    image6 = cv.filter2D(grayscale, -1, kernel[5])
    image = [image1, image2, image3, image4, image5, image6]
    return image
```

Plotting New Images

```
In [22]: def plot_img(img_set):
n = len(img_set)
plt.figure(figsize = (16, 10))
for i in range(n):
img = img_set[i]
ch = len(img.shape)
plt.subplot(2, 3, i + 1)
plt.imshow(img_set[i], cmap = 'gray')
plt.show()
```

```
In [23]: if __name__ == "__main__":
img, grayscale = read_image()
kernel = kernel_build()
new_image = image_build(grayscale, kernel)
plot_img(new_image)
```

