

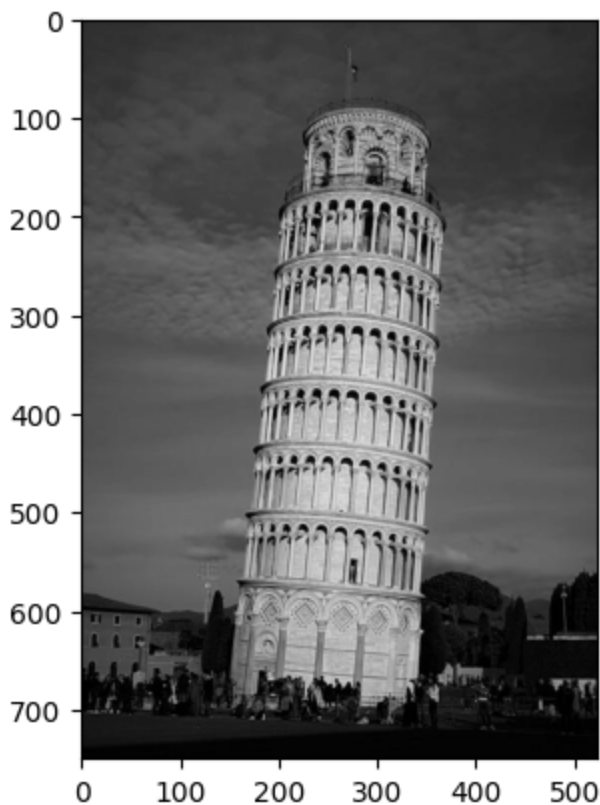
Lab 7 - Edge Detection

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

```
In [2]: # Read the input image

image=cv2.imread('../Images/pisa.jpeg', cv2.IMREAD_GRAYSCALE)
plt.imshow(image, cmap="gray")
```

Out[2]: <matplotlib.image.AxesImage at 0x17fc9985ee0>

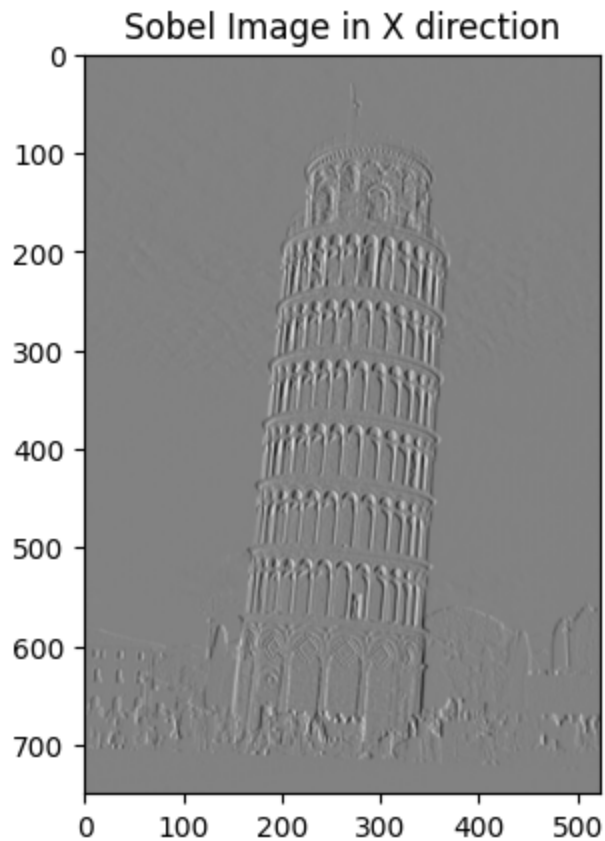


Sobel Filter

```
In [3]: # apply sobel filter

sobel_image_x = cv2.Sobel(image, cv2.CV_64F, 1, 0, ksize=5)
plt.title("Sobel Image in X direction")
plt.imshow(sobel_image_x, cmap='gray')
```

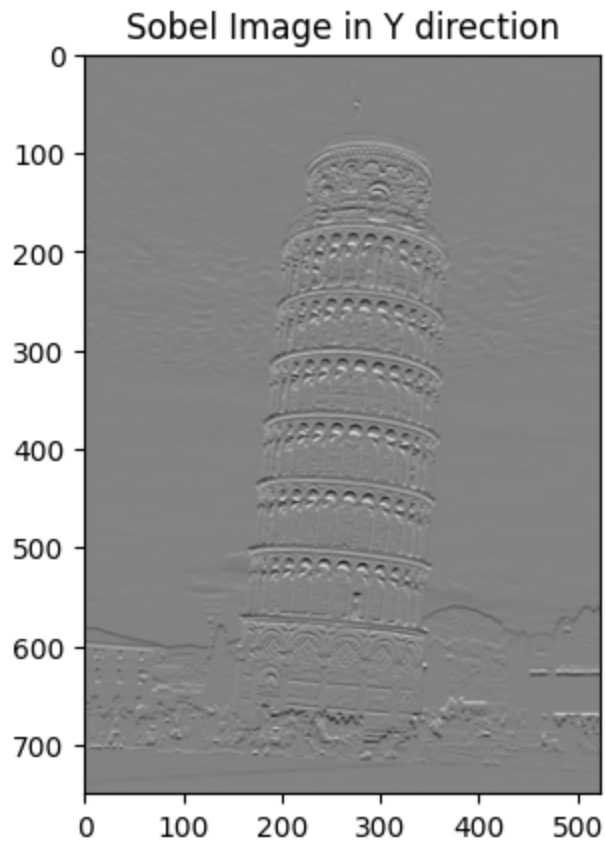
Out[3]: <matplotlib.image.AxesImage at 0x17fc9dc8fb0>



In [4]: *# apply sobel filter*

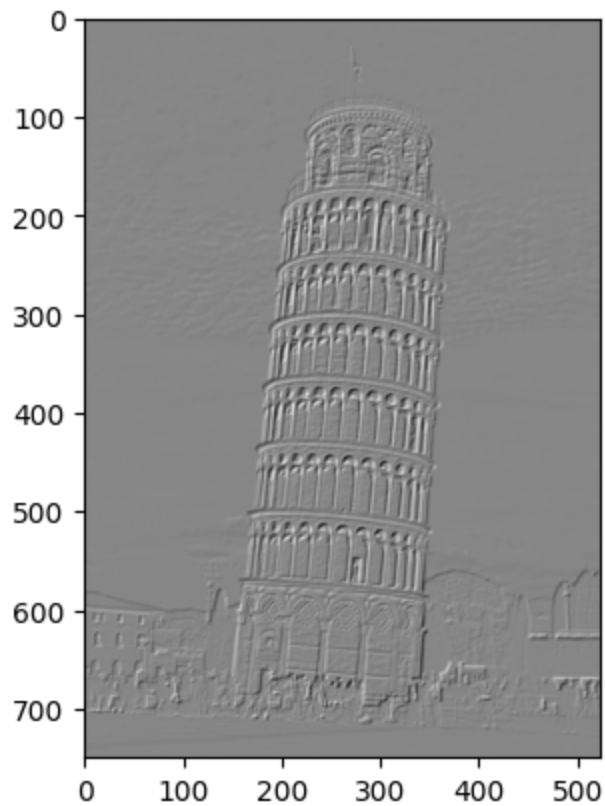
```
sobel_image_y = cv2.Sobel(image, cv2.CV_64F, 0, 1, ksize=5)
plt.title("Sobel Image in Y direction")
plt.imshow(sobel_image_y, cmap='gray')
```

Out[4]: <matplotlib.image.AxesImage at 0x17fc9e92ae0>



```
In [6]: sobel_image = cv2.add(sobel_image_x, sobel_image_y)
plt.imshow(sobel_image, cmap='gray')
```

Out[6]: <matplotlib.image.AxesImage at 0x17fc9de2630>



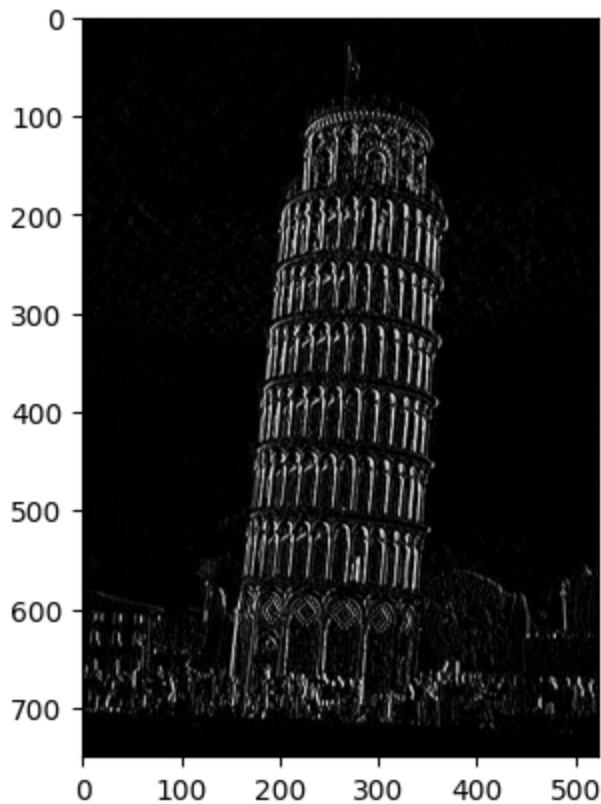
Prewit Filter

```
In [9]: # prewit filter
prewit_kernel_vertical = np.array([
    [-1,0,1],
    [-1,0,1],
    [-1,0,1]
])

prewit_kernel_horizontal = np.array([
    [-1,-1,-1],
    [0,0,0],
    [1,1,1]
])
```

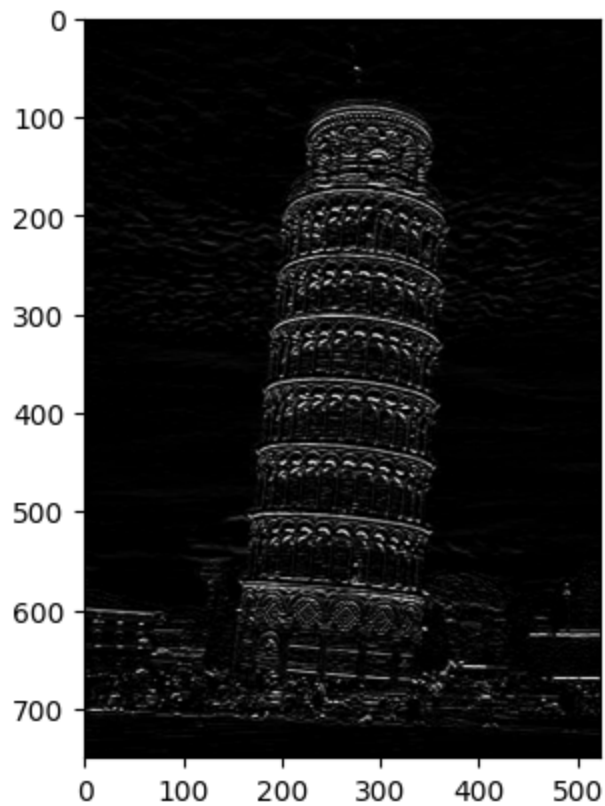
```
In [10]: prewit_image_vertical = cv2.filter2D(src=image, ddepth=-1, kernel=prewit_kernel_ver
plt.imshow(prewit_image_vertical, cmap='gray')
```

```
Out[10]: <matplotlib.image.AxesImage at 0x17fce6191f0>
```



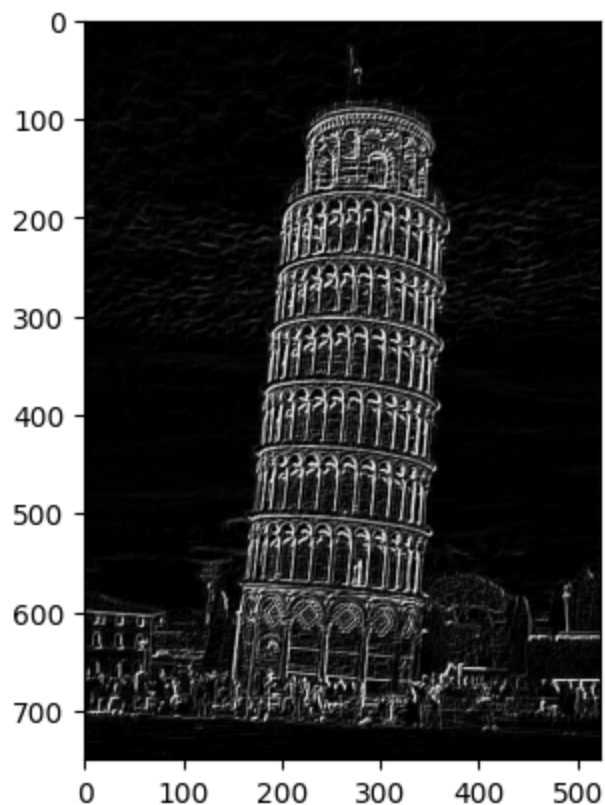
```
In [11]: prewit_image_horizontal = cv2.filter2D(src=image, ddepth=-1, kernel=prewit_kernel_h
plt.imshow(prewit_image_horizontal, cmap='gray')
```

```
Out[11]: <matplotlib.image.AxesImage at 0x17fcdf0d430>
```



```
In [12]: prewit_image = cv2.add(prewit_image_vertical, prewit_image_horizontal)
plt.imshow(prewit_image, cmap='gray')
```

Out[12]: <matplotlib.image.AxesImage at 0x17fce8382f0>



Robert Filter

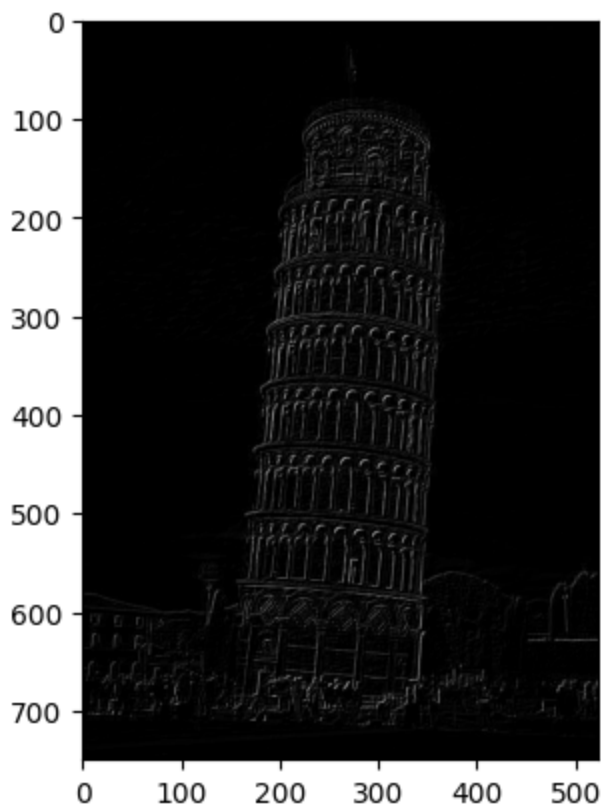
In [13]: *# roberts filter*

```
robert_kernel_x = np.array([
    [0,0,0],
    [0,1,0],
    [0,0,-1]
])

robert_kernel_y = np.array([
    [0,0,0],
    [0,0,1],
    [0,-1,0]
])
```

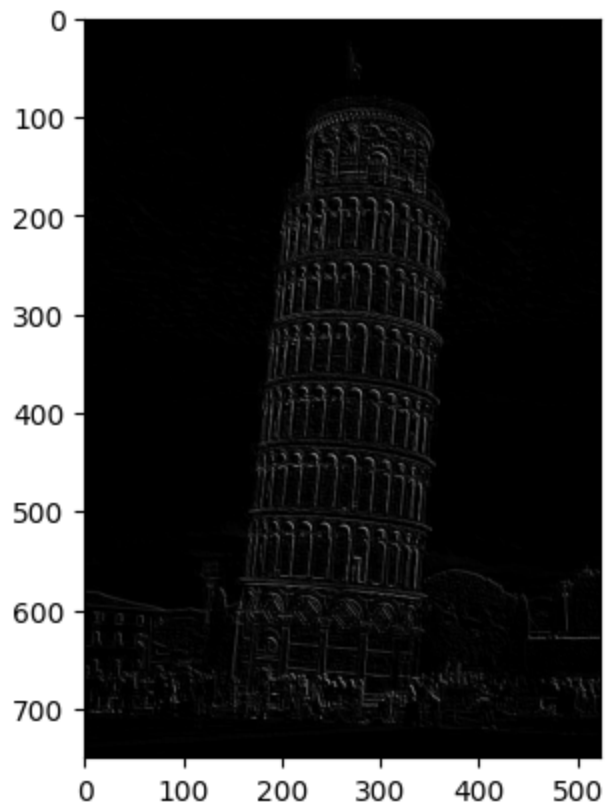
In [14]: `robert_image_x = cv2.filter2D(src=image, ddepth=-1, kernel=robert_kernel_x)`
`plt.imshow(robert_image_x, cmap='gray')`

Out[14]: `<matplotlib.image.AxesImage at 0x17fce8438f0>`



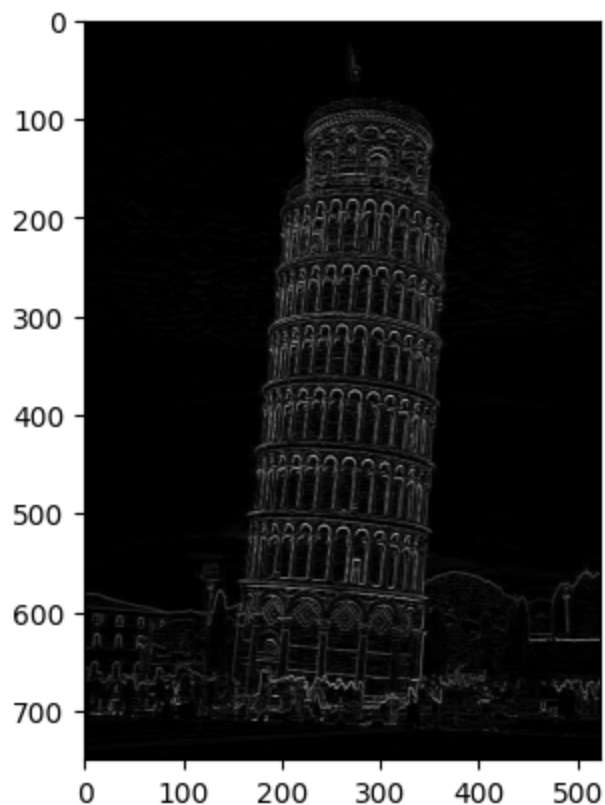
In [15]: `robert_image_y = cv2.filter2D(src=image, ddepth=-1, kernel=robert_kernel_y)`
`plt.imshow(robert_image_y, cmap='gray')`

Out[15]: `<matplotlib.image.AxesImage at 0x17fce83bec0>`



```
In [16]: robert_image = cv2.add(robert_image_x, robert_image_y)
plt.imshow(robert_image, cmap='gray')
```

Out[16]: <matplotlib.image.AxesImage at 0x17fce746f60>



```
In [25]: import matplotlib.image as mpimg

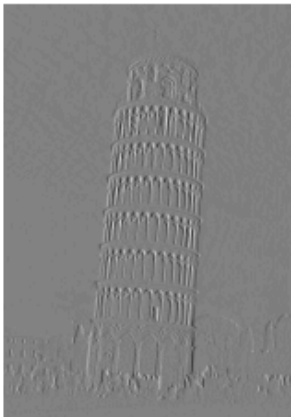
# Create a figure with a 3x3 grid
fig, axes = plt.subplots(3, 3, figsize=(8, 8))

image_paths = [sobel_image_x, sobel_image_y, sobel_image, prewit_image_horizontal,
titles = ['Sobel Filter X', 'Sobel Filter X', 'Sobel Image', 'Prewit Filter X', 'Pr

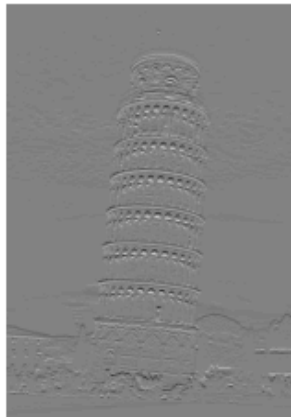
# Loop through each axis in the grid and each image path
t = 0
for i, ax in enumerate(axes.flat):
    # Load and display the image on the current axis
    ax.set_title(titles[i])
    ax.imshow(image_paths[i], cmap='gray')
    ax.axis('off') # Hide the axis labels and ticks

# Adjust the spacing between images
plt.tight_layout()
plt.show()
```

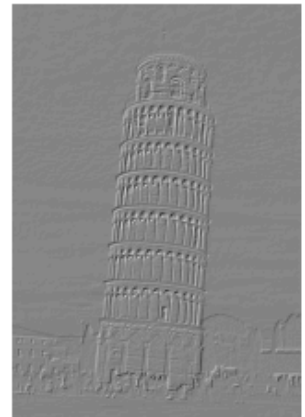

Sobel Filter X



Sobel Filter X



Sobel Image



Prewit Filter X



Prewit Filter Y



Prewit Image



Robert Filter X



Robert Filter Y



Robert Image



In []: