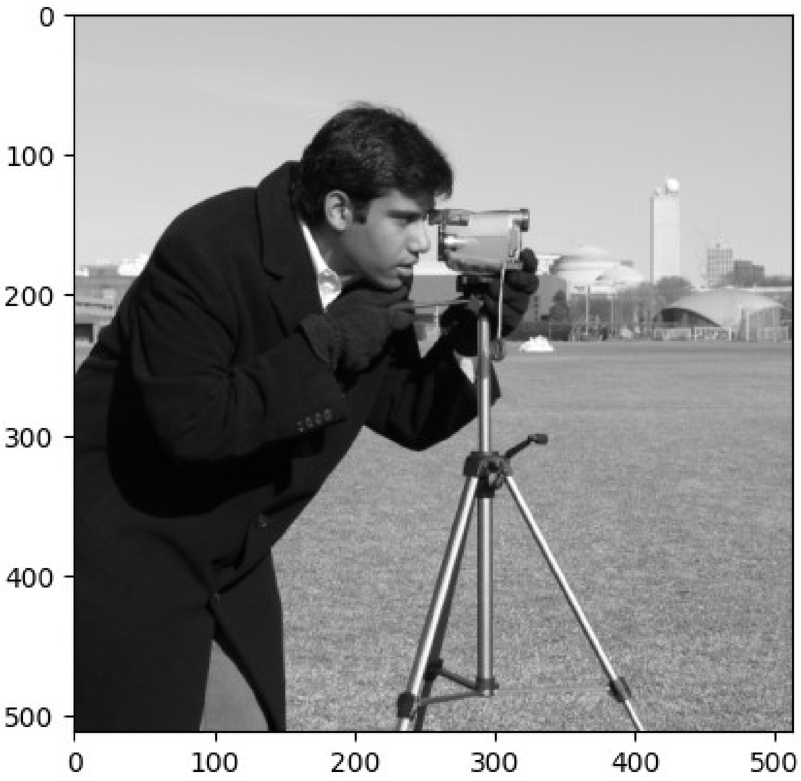
Name: Nisha Kini Roll No: C050

g(x) is large , edge is vertical and vice versa

Aim: To identify the edges of the given image using sobel filter

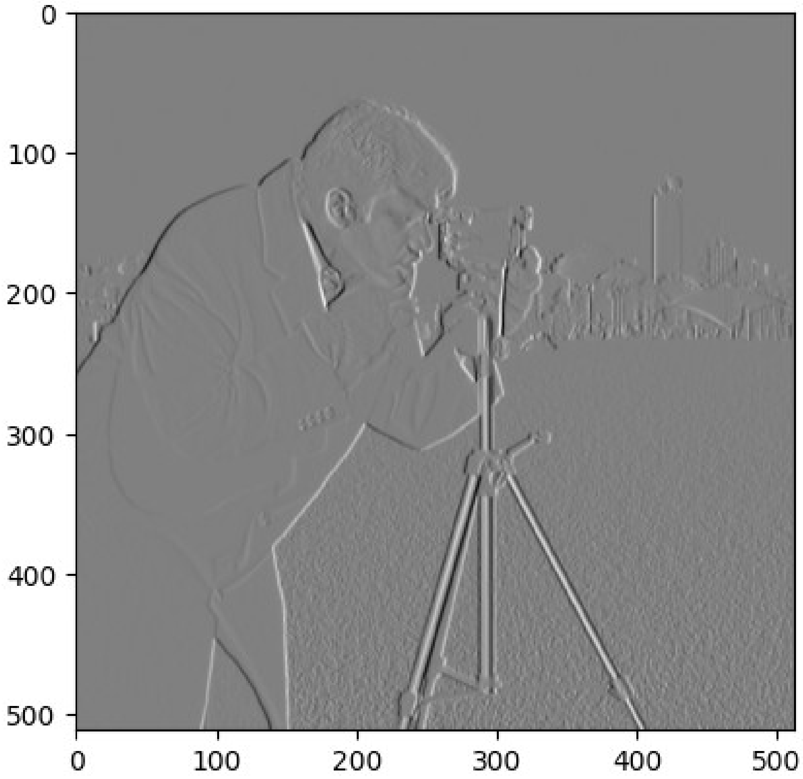
```
import numpy as np
import cv2
import matplotlib.pyplot as plt
from skimage import data

img = data.camera()
plt.imshow(img, cmap='gray')
```

```
<matplotlib.image.AxesImage at 0x7f415646d570>
```



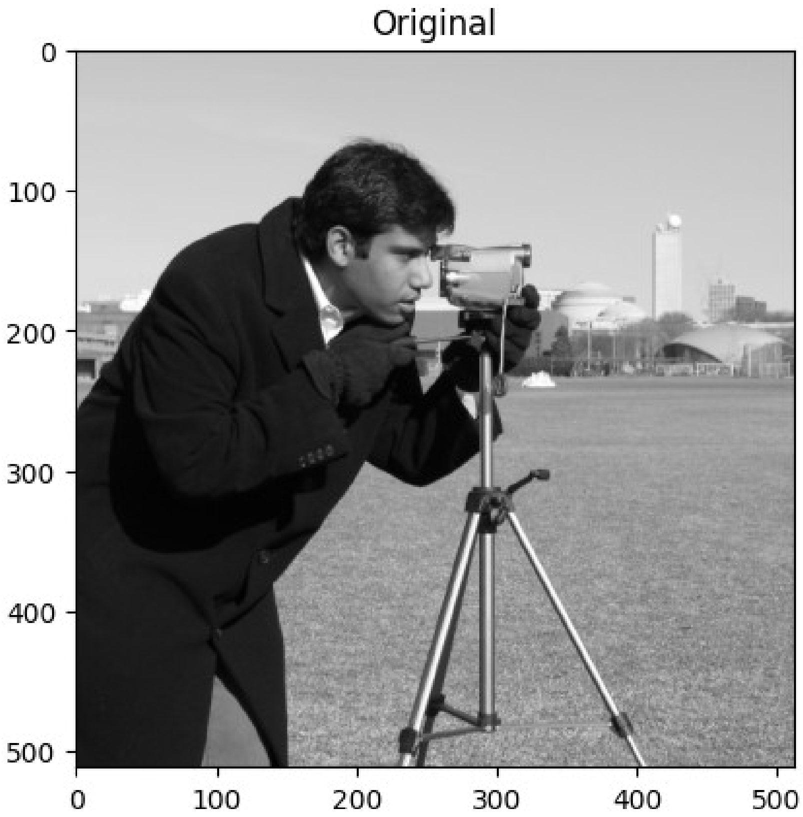
```
grad_x =cv2.Sobel(img, ddepth=cv2.CV_32F, dx=1, dy=0, ksize=3)
grad_y =cv2.Sobel(img, ddepth=cv2.CV_32F, dx=0, dy=1, ksize=3)

plt.imshow(grad_x, cmap ='gray')
```

```
<matplotlib.image.AxesImage at 0x7f4155ae63b0>
```

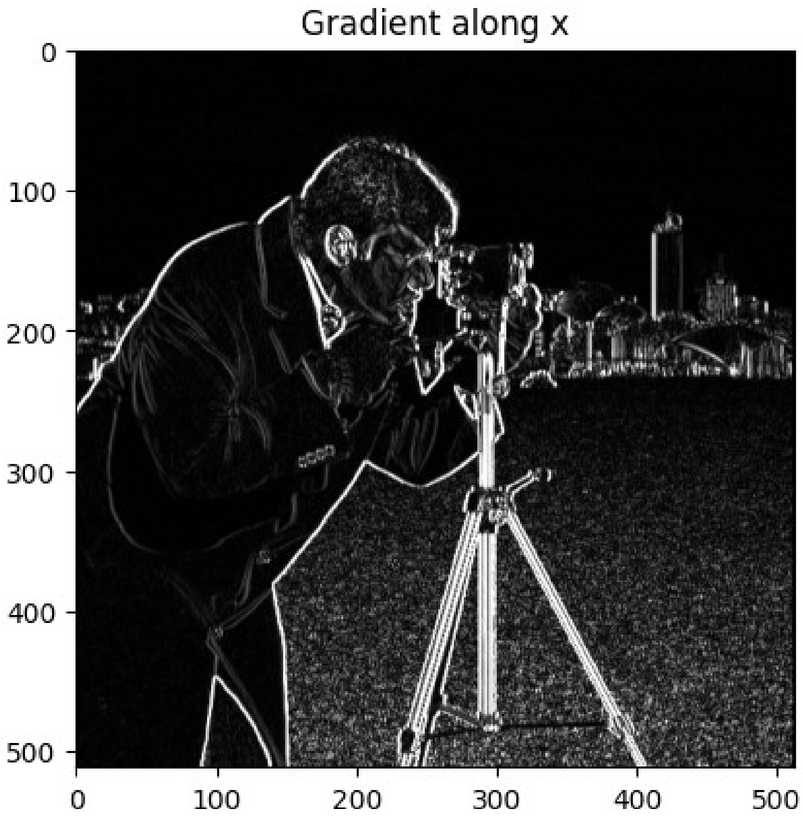


```
grad_x_abs=cv2.convertScaleAbs(grad_x)

grad_y_abs=cv2.convertScaleAbs(grad_y)

plt.imshow(img,cmap='gray')
plt.title('Original')
```

```
Text(0.5, 1.0, 'Original')
```

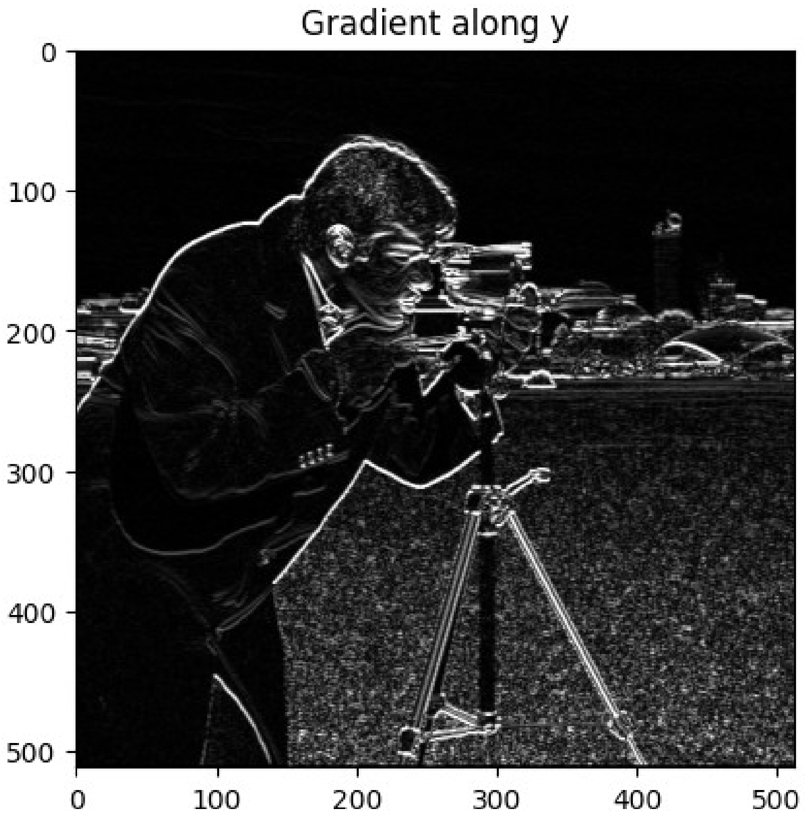


```
plt.imshow(grad_x_abs,cmap='gray')
plt.title('Gradient along x')

Text(0.5, 1.0, 'Gradient along x')
```



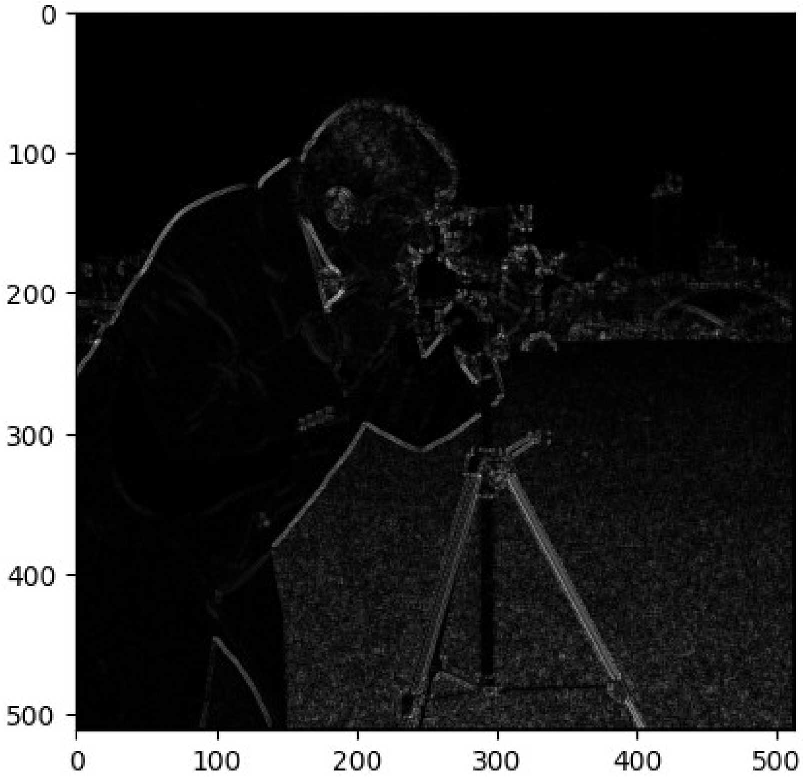
```
plt.imshow(grad_y_abs,cmap='gray')
plt.title('Gradient along y')
```

```
Text(0.5, 1.0, 'Gradient along y')
```



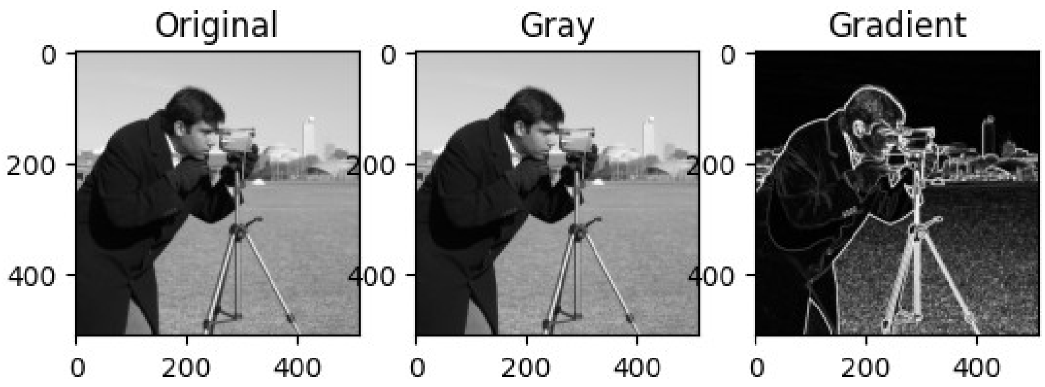
```
grad_xy =cv2.Sobel(img, ddepth=cv2.CV_32F, dx=1, dy=1, ksize=3)
grad_xy_abs=cv2.convertScaleAbs(grad_xy)
plt.imshow(grad_xy_abs,cmap='gray')
```

```
<matplotlib.image.AxesImage at 0x7f4155d71060>
```

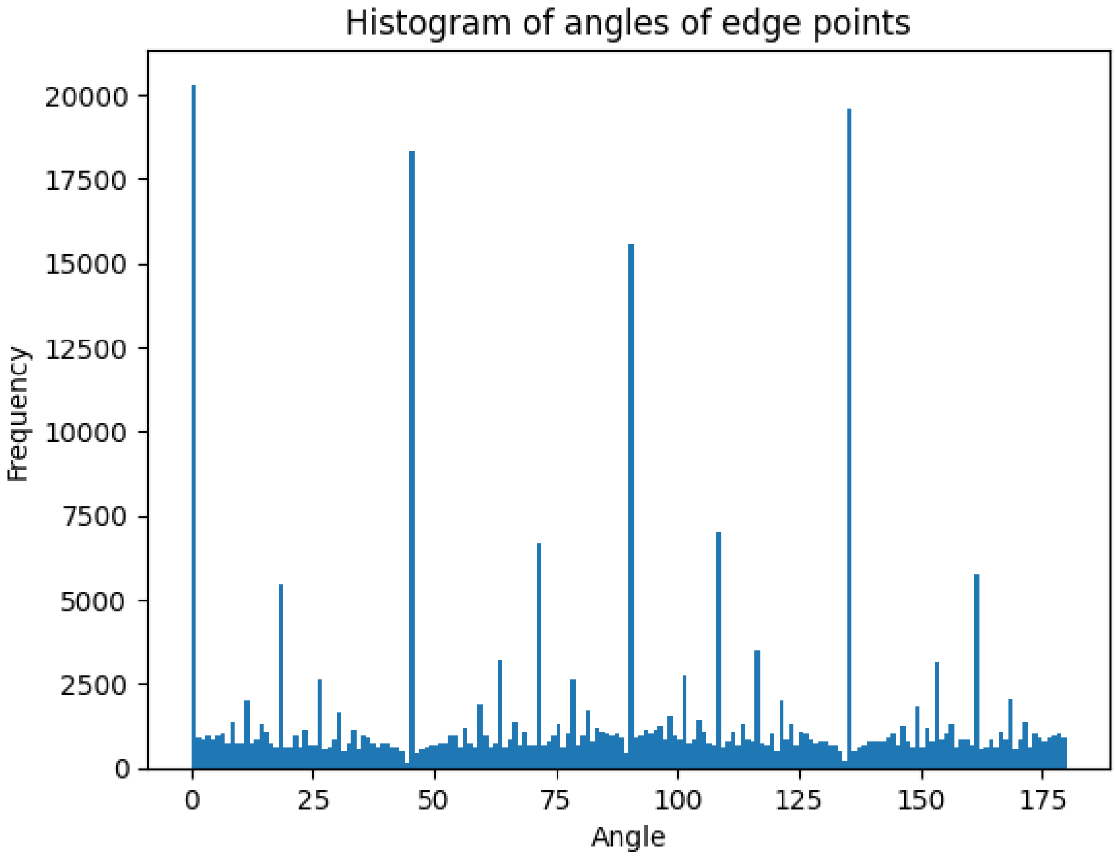


```
grad_magnitude=np.sqrt(grad_x**2+grad_y**2)
# grad_magnitude=grad_magnitude/np.max(grad_magnitude)
grad_mag_abs=cv2.convertScaleAbs(grad_magnitude)

plt.subplot(1,3,1)
plt.imshow(imag,cmap='gray')
plt.title('Original')
plt.subplot(1,3,2)
plt.imshow(img,cmap='gray')
plt.title('Gray')
plt.subplot(1,3,3)
plt.imshow(grad_mag_abs,cmap='gray')
plt.title('Gradient')
```

```
Text(0.5, 1.0, 'Gradient')
```



```
angle=(np.arctan2(grad_y,grad_x)*180/np.pi)%180
r,c=img.shape
angle1d=np.reshape(angle,(r*c,1))
plt.hist(angle1d,bins=180)
plt.title('Histogram of angles of edge points')
plt.xlabel('Angle')
plt.ylabel('Frequency')

Text(0, 0.5, 'Frequency')
```



Self used image

```
imag = cv2.imread('/content/iStock-1422593791.jpg')
img = cv2.cvtColor(imag, cv2.COLOR_BGR2GRAY)
plt.show('Grayscale', img)

plt.imshow(imag)
```

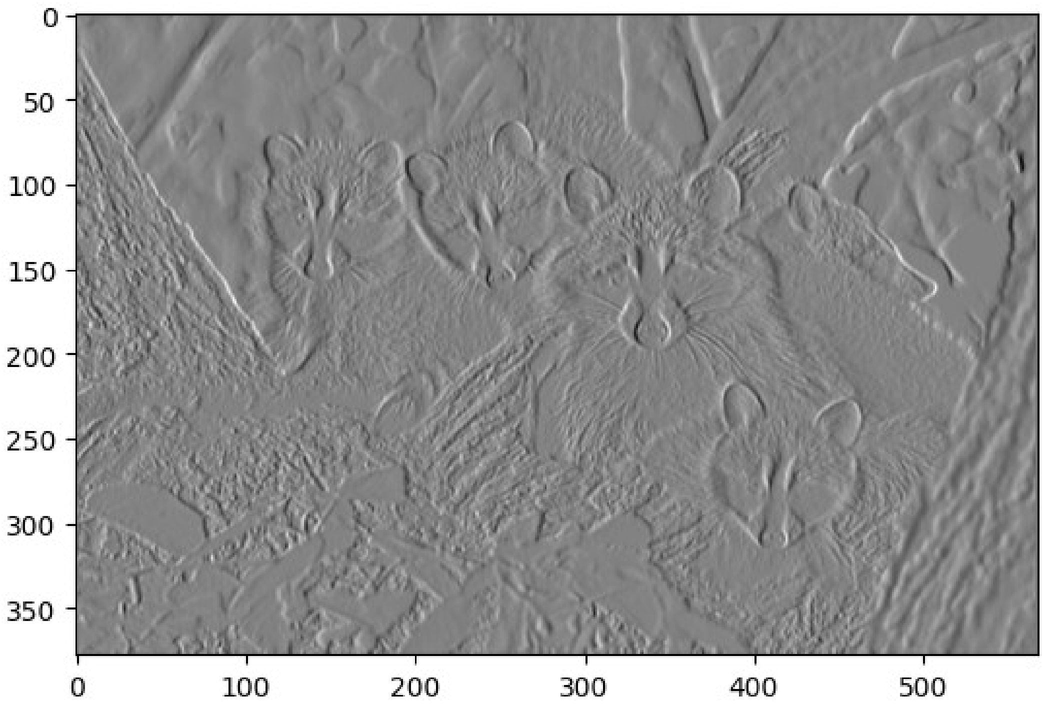
```
<matplotlib.image.AxesImage at 0x7f415046b850>
```

```
plt.imshow(img, cmap='gray')
```

```
<matplotlib.image.AxesImage at 0x7f4150a70040>
```

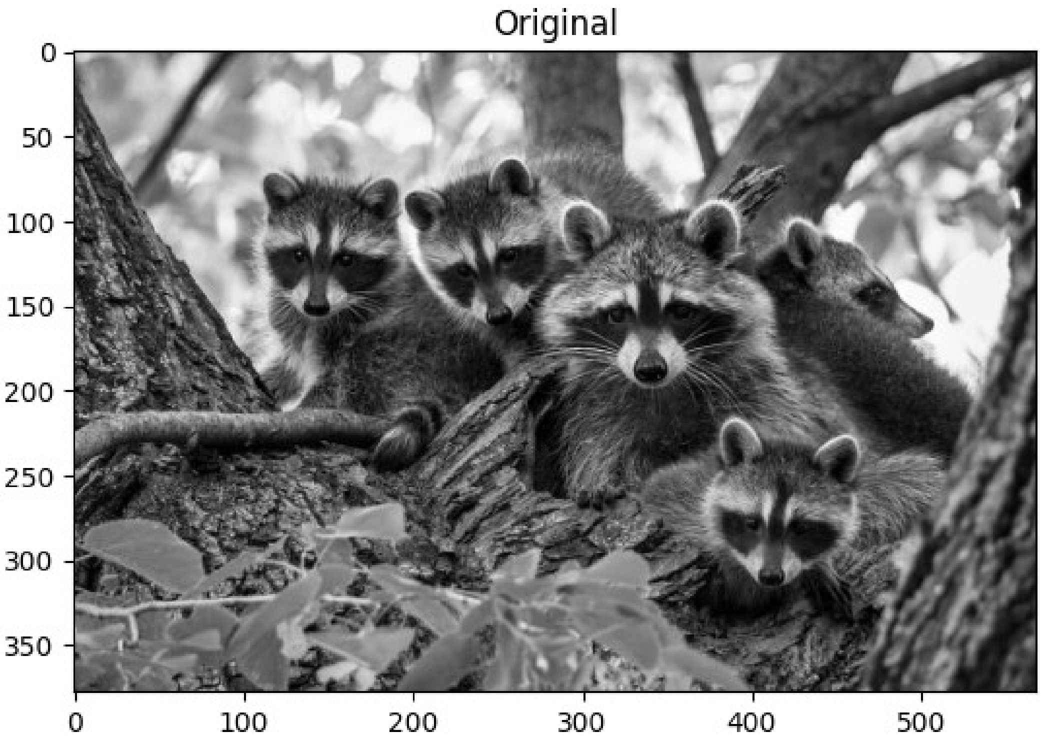
```
grad_x =cv2.Sobel(img, ddepth=cv2.CV_32F, dx=1, dy=0, ksize=3)
grad_y =cv2.Sobel(img, ddepth=cv2.CV_32F, dx=0, dy=1, ksize=3)

plt.imshow(grad_x, cmap ='gray')
```

```
<matplotlib.image.AxesImage at 0x7f41508df310>
```



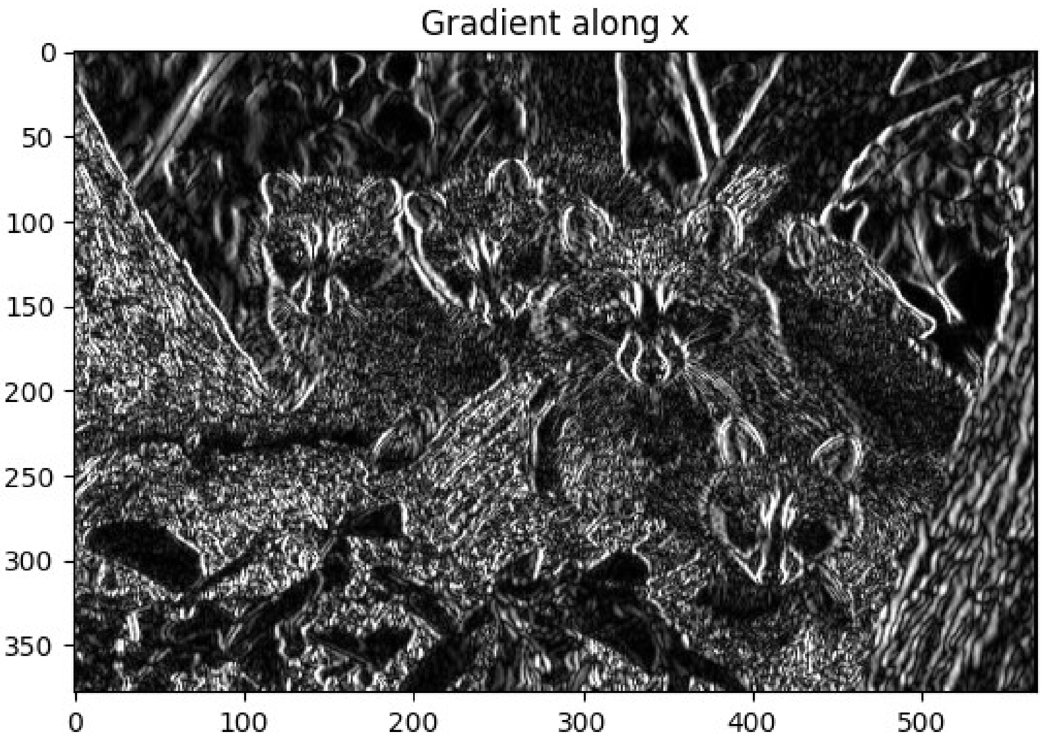
```
grad_x_abs=cv2.convertScaleAbs(grad_x)

grad_y_abs=cv2.convertScaleAbs(grad_y)

plt.imshow(img,cmap='gray')
plt.title('Original')
```

```
Text(0.5, 1.0, 'Original')
```

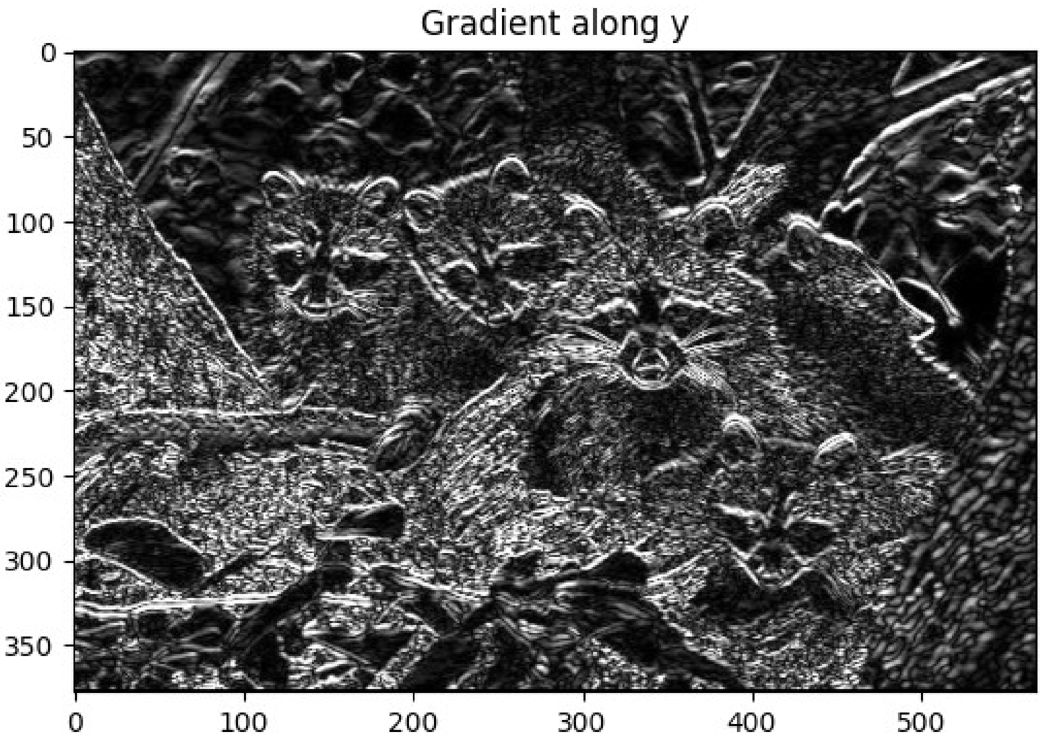


```
plt.imshow(grad_x_abs,cmap='gray')
plt.title('Gradient along x')
```

```
Text(0.5, 1.0, 'Gradient along x')
```



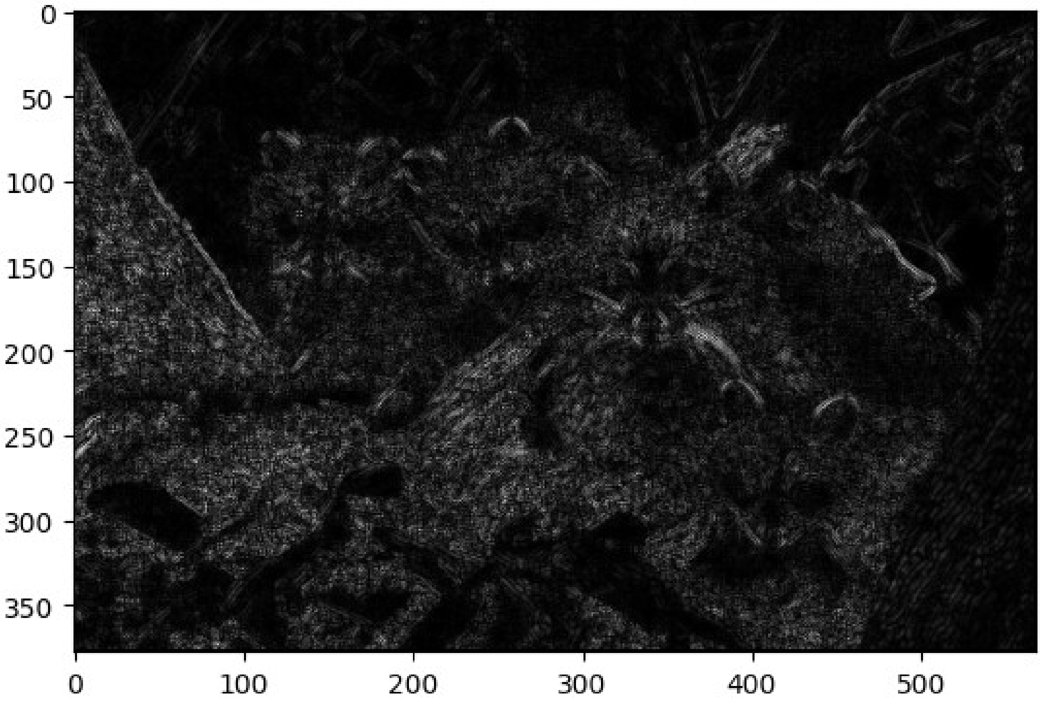
```
plt.imshow(grad_y_abs,cmap='gray')
plt.title('Gradient along y')

Text(0.5, 1.0, 'Gradient along y')
```



Gradient along y

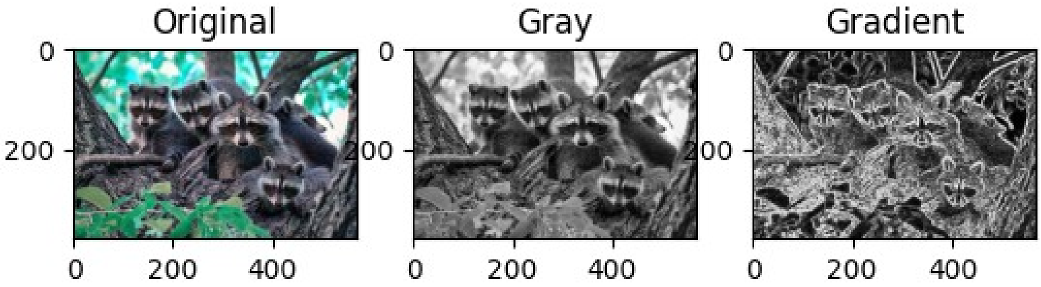
```
grad_xy =cv2.Sobel(img, ddepth=cv2.CV_32F, dx=1, dy=1, ksize=3)
grad_xy_abs=cv2.convertScaleAbs(grad_xy)
plt.imshow(grad_xy_abs,cmap='gray')
```

```
<matplotlib.image.AxesImage at 0x7f41506e41c0>
```



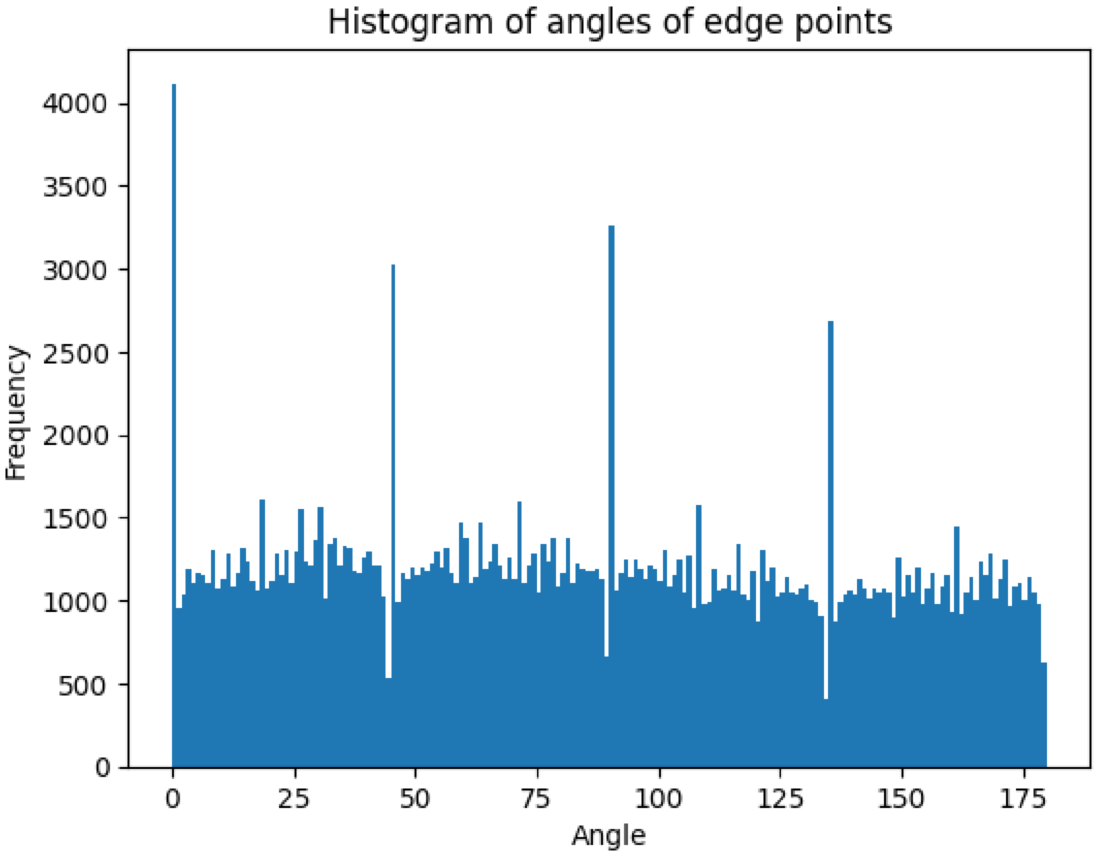
```
grad_magnitude=np.sqrt(grad_x**2+grad_y**2)
# grad_magnitude=grad_magnitude/np.max(grad_magnitude)
grad_mag_abs=cv2.convertScaleAbs(grad_magnitude)

plt.subplot(1,3,1)
plt.imshow(imag,cmap='gray')
plt.title('Original')
plt.subplot(1,3,2)
plt.imshow(img,cmap='gray')
plt.title('Gray')
plt.subplot(1,3,3)
plt.imshow(grad_mag_abs,cmap='gray')
plt.title('Gradient')
```

```
Text(0.5, 1.0, 'Gradient')
```



```
angle=(np.arctan2(grad_y,grad_x)*180/np.pi)%180
r,c=img.shape
angle1d=np.reshape(angle,(r*c,1))
plt.hist(angle1d,bins=180)
plt.title('Histogram of angles of edge points')
plt.xlabel('Angle')
plt.ylabel('Frequency')
```

```
Text(0, 0.5, 'Frequency')
```



```
img.shape
```

```
(378, 568)
```

Conclusion: Sobel filter is used to identify horizontal,vertical,diagonal and magnitude of gradients at each pixel of the given image. For each pixel, histogram of angle of each pixel is plotted, it is observed that most of the pixels are having gradients along 0,48,90, 135 degrees. If size of filter is increased from 3 * 3 to 9 * 9 and 31 * 31 then it is observed that pixels along small edges are not highighted with filter of large size however large edges with more options of angles can be identified by these filters. Sobel filter of large size can be used to reduce the effect of small edges considered as noisy objects.