Image Understanding and Processing (OpenCv-Python)

Lab Exercise – 08

Year 4 **Semester 1, 2024**

Goal

- Find image gradients, edges
- Apply cv2.Sobel() function
- Apply cv2.Laplacian() function

Edge Detection

Edge detection is considered as the most common approach for detecting meaningful discontinuities in the grey- level.

Write programs which explains the performance of edge detection on an image

1. Sobel Operator

Calculates first and second image derivatives using extended Sobel operator. Use the following function to apply the Sobel operator:

cv2.Sobel(src, type, xorder, yorder, ksize)

Parameters:

src – Source image

type – Destination image type

xorder – First Order derivative in x direction

yorder – First Order derivative in y direction

ksize – Size of the extended Sobel kernel, must be 1, 3, 5 or 7

The function is called with (xorder=1, yorder=0, ksize =3)

|-1 0 1|

|-2 0 2|

|-1 0 1|

and (xorder=0, yorder=1, ksize =3)

|-1 -2 -1|

| 0 0 0|

| 1 2 1|

2. Laplacian Operator

The function calculates the Laplacian of the source image by filtering the image with the following 3X3 aperture:

|-1 -1 -1|

| -1 8 -1|

|-1 -1 -1|

Use the following function to apply the Laplacian operator:

cv2.Laplacian(src, type)

Parameters:

src – Source image

dst – Destination image type

Modify the program to apply **Laplacian of Gaussian** to reduce the noisy edges. You may use the **cv2.GaussianBlur()** function to remove the Gaussian noise.