TEAM: 92-93 **SUBJECT:** Computer Vision **FACULTY:** Dr. Tilottama Goswami

COMPUTER VISION ASSIGNMENT – 1

AIM: The Aim of this code is to demonstrate the application of Edge Detection Gaussian based operators specifically the Canny, Laplacian, Gaussian Blur on an image.

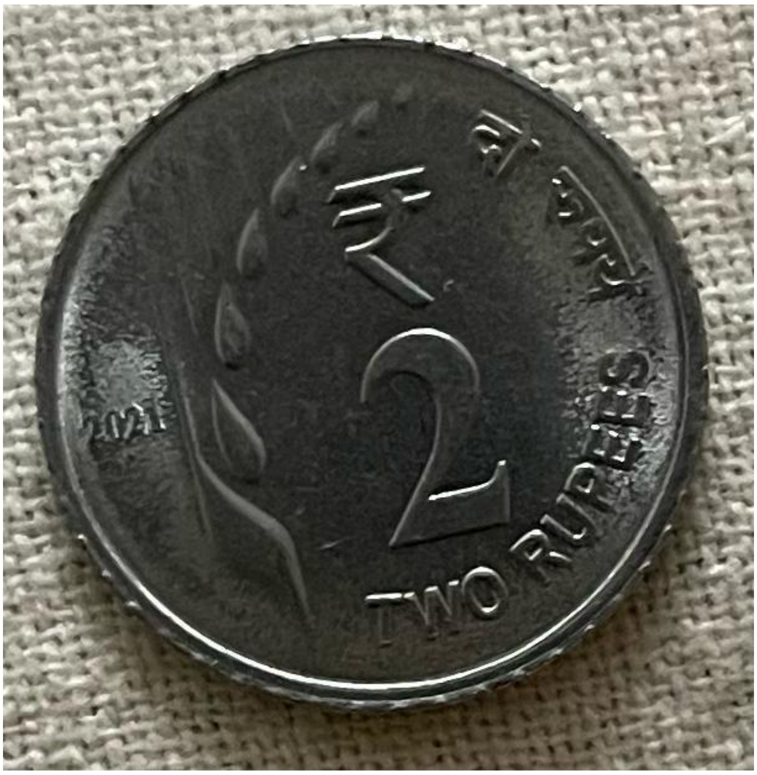
- **Gaussian Blur**: It is a pre-processing step that helps in reducing noise and creating a smoother image. It is often used before applying edge detection to improve the performance of the edge detection algorithms.
- **Canny Edge Detection**: It is widely used and robust. It provides good localization, low error rate, and minimal response to noise. The Canny edge detector involves multiple steps, including gradient computation, non-maximum suppression, and edge tracking by hysteresis.
- **Laplacian Edge Detection**: It emphasizes high-frequency components and is sensitive to noise. The absolute value of the Laplacian is often taken to highlight edges. It may result in thicker edges compared to Canny.

CODE:

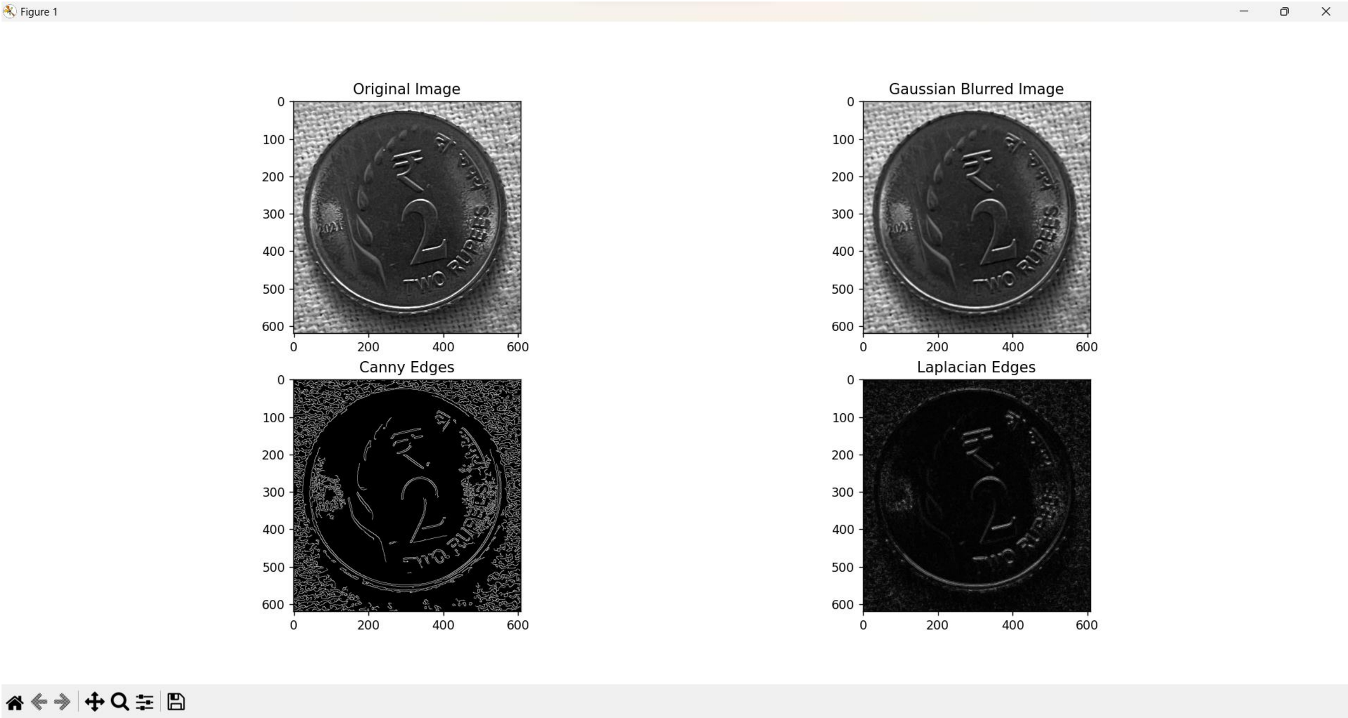
```
import cv2

import numpy as np

import matplotlib.pyplot as plt

# Load an image

image = cv2.imread('coin.jpeg', cv2.IMREAD_GRAYSCALE)

# Check if the image is loaded successfully

if image is None or image.size == 0:

    print("Error: Could not open the image or the image is empty.")

    exit()

# Apply Gaussian blur

blurred_image = cv2.GaussianBlur(image, (5, 5), 0)

# Apply Canny edge detection

canny_edges = cv2.Canny(blurred_image, 50, 150)

# Apply Laplacian edge detection

laplacian_edges = cv2.Laplacian(blurred_image, cv2.CV_64F)
```

```
# Display the original image and edges

plt.figure(figsize=(12, 8))

plt.subplot(2, 2, 1), plt.imshow(image, cmap='gray'), plt.title('Original Image')

plt.subplot(2, 2, 2), plt.imshow(blurred_image, cmap='gray'), plt.title('Gaussian Blurred Image')

plt.subplot(2, 2, 3), plt.imshow(canny_edges, cmap='gray'), plt.title('Canny Edges')

plt.subplot(2, 2, 4), plt.imshow(np.uint8(np.abs(laplacian_edges)), cmap='gray'), plt.title('Laplacian Edges')

plt.show()
```

INPUT:



OUTPUT:



CONCLUSION:

- The Canny filter is a more sophisticated and widely used filter for edge detection. It consists of four main steps: smoothing, gradient calculation, non-maximum suppression, and hysteresis thresholding. Smoothing is done by applying a Gaussian filter to the image to reduce noise.
- Laplacian is very sensitive to noise, as it amplifies any small variation in pixel intensity, and that it does not provide information about the edge direction or orientation.