1. Perform basic Image Handling and processing operations on the image.

• Read an image in python and Convert an Image to Grayscale

**CODE:**

**import cv2**

**from google.colab.patches import cv2\_imshow**

**image\_path = 'opencv 1.PNG' # Make sure you've uploaded this to Colab**

**image = cv2.imread(image\_path)**

**if image is None:**

**print("Error: Cannot find the image.")**

**else:**

**gray\_image = cv2.cvtColor(image, cv2.COLOR\_BGR2GRAY)**

**print("Original Image:")**

**cv2\_imshow(image)**

**print("Grayscale Image:")**

**cv2\_imshow(gray\_image)**

**OUTPUT:**

****

2.2. Perform basic Image Handling and processing operations on the image

• Read an image in python and Convert an Image to Blur using GaussianBlur.

**CODE:**

**import cv2**

**from google.colab.patches import cv2\_imshow**

**image\_path = 'openCv 2.png'**

**image = cv2.imread(image\_path)**

**if image is None:**

**print("Error: Image not found.")**

**else:**

**blurred\_image = cv2.GaussianBlur(image, (15, 15), 0)**

**print("Original Image:")**

**cv2\_imshow(image)**

**print("Blurred Image (Gaussian Blur):")**

**cv2\_imshow(blurred\_image)**

**OUTPUT:**

****

**3.3. Perform basic Image Handling and processing operations on the image**

**• Read an image in python and Convert an Image to show outline using Canny function.**

**Code:**

**import cv2**

**from google.colab.patches import cv2\_imshow**

**image\_path = 'opencv 3.png'**

**image = cv2.imread(image\_path)**

**if image is None:**

**print("Error: Image not found.")**

**else:**

**gray\_image = cv2.cvtColor(image, cv2.COLOR\_BGR2GRAY)**

**edges = cv2.Canny(gray\_image, 100, 200)**

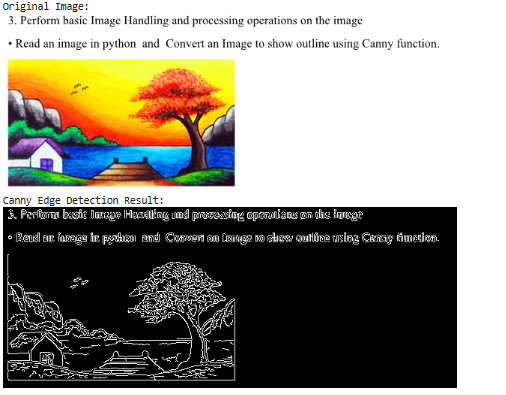
**print("Original Image:")**

**cv2\_imshow(image)**

**print("Canny Edge Detection Result:")**

**cv2\_imshow(edges)**

**Output:**

****

**4.4. Perform basic Image Handling and processing operations on the image**

**• Read an image in python and Dilate an Image using Dilate function.**

**Code:**

**import cv2**

**import numpy as np**

**from google.colab.patches import cv2\_imshow**

**image\_path = 'opencv 4.png'**

**image = cv2.imread(image\_path)**

**if image is None:**

**print("Error: Image not found.")**

**else:**

**gray = cv2.cvtColor(image, cv2.COLOR\_BGR2GRAY)**

**\_, binary = cv2.threshold(gray, 120, 255, cv2.THRESH\_BINARY)**

**kernel = np.ones((5, 5), np.uint8)**

**dilated\_image = cv2.dilate(binary, kernel, iterations=1)**

**print("Original Image:")**

**cv2\_imshow(image)**

**print("Dilated Image:")**

**cv2\_imshow(dilated\_image)**

**Output:**

****