By using google colab code to access your webcam, capture three different facial expressions of yourself.

1. Apply both Canny Edge Detection and Sobel Edge Detection to only the face area of the image using a suitable threshold values.

To apply edge detection on the face, we must do face detection.

faces = face\_cascade.detectMultiScale(grayImage,1.1,4) print(f"{len(faces)} faces detected in the image.")

We use above line of code to detect faces and print how many face and it detect 1 only.

for x, y, w, h in faces:

    face\_roi = grayImage[y:y + h, x:x + w]

    blurred = cv2.GaussianBlur(face\_roi,(7,7),5)

    #sobel

    sobelx = cv2.Sobel(blurred, cv2.CV\_8U,1,0,ksize=3)

    sobely = cv2.Sobel(blurred, cv2.CV\_8U,0,1,ksize=3)

    sobel = sobelx + sobely

    grayImage[y:y+h, x:x+w] = sobel

Then, we take the face as region of interest and apply sobel and canny edge detection.

1. Perform blurring of only faces in those images.

  # detect all the faces in the image

  faces = face\_cascade.detectMultiScale(gray\_img,1.1,4)

  # print the number of faces detected

  print(f"{len(faces)} faces detected in the image.")

  # for every face, draw a green rectangle

  for x, y, w, h in faces:

    face\_roi = img[y:y + h, x:x + w]

    blurred = cv2.GaussianBlur(face\_roi,(7,7),5)

    img[y:y+h, x:x+w] = blurred

The steps to do burring is the same as the first question which take face as region of interest and apply blurring.

1. Calculating the number of coins in an image using contours.

The first step is apply gaussian blur and canny edge detection. This is because canny edge detection is better at drawing edge than sobel.

blur = cv2.GaussianBlur(gray,(17, 17), 0)

canny = cv2.Canny(blur, 30, 150)

Next, use opencv library to find contours which will also draw the contour on the image.

(cnts, \_) = cv2.findContours(canny, cv2.RETR\_EXTERNAL, cv2.CHAIN\_APPROX\_SIMPLE)

cv2.drawContours(img, cnts, -1, (0, 255, 0), 2)

From the above equation, we will get a return how many contour gain and we achieved 4 same to the image.