Practice Project 3 - OpenCV based : People Counting -Object Detection

Table of Contents

1. Importing Libraries

In [ ]:

```
import cv2
```

2. Python implementation

Load the pre-trained Haar cascade for detecting faces

In [ ]:

```
face_cascade = cv2.CascadeClassifier(cv2.data.haarcascades + 'haarcascade_frontalface_default.xml')
```

Variables to keep track of people count

In [ ]:

```
total_people = 0
people_in = 0
people_out = 0
```

Capture video from a webcam or a video file

In [ ]:

```
cap = cv2.VideoCapture(0)  # Replace 0 with the video file path if using a file
```

In [ ]:

```
while True:
    # Read a frame from the video source
    ret, frame = cap.read()
    if not ret:
        break

    # Convert the frame to grayscale
    gray = cv2.cvtColor(frame, cv2.COLOR_BGR2GRAY)

    # Detect faces in the frame
    faces = face_cascade.detectMultiScale(gray, 1.1, 4)

    # Draw rectangles around the detected faces
    for (x, y, w, h) in faces:
        cv2.rectangle(frame, (x, y), (x+w, y+h), (0, 255, 0), 2)

    # Update people count based on face detection
    num_people = len(faces)
    if num_people > total_people:
        people_in += (num_people - total_people)
    elif num_people < total_people:
        people_out += (total_people - num_people)
    total_people = num_people

    # Display the current people count on the frame
    cv2.putText(frame, f'People In: {people_in}', (10, 30), cv2.FONT_HERSHEY_SIMPLEX, 0.8, (0, 255, 0), 2)
    cv2.putText(frame, f'People Out: {people_out}', (10, 60), cv2.FONT_HERSHEY_SIMPLEX, 0.8, (0, 255, 0), 2)

    # Display the frame
    cv2.imshow('People Counter', frame)

    # Exit loop if 'q' is pressed
    if cv2.waitKey(1) & 0xFF == ord('q'):
        break

cap.release()
cv2.destroyAllWindows()
```