import cv2

import numpy as np

# Load the pre-trained MobileNet-SSD model and the prototxt file

net = cv2.dnn.readNetFromCaffe('C:\\Users\\sudha\\Desktop\\deploy.prototxt', 'C:\\Users\\sudha\\Desktop\\MobileNet-SSD-master\\MobileNet-SSD-master\mobilenet\_iter\_73000.caffemodel')

# Initialize the video capture (0 for webcam or provide a video file path)

cap = cv2.VideoCapture(0)

# List of classes for MobileNet-SSD

CLASSES = ["background", "aeroplane", "bicycle", "bird", "boat", "bottle", "bus",

           "car", "cat", "chair", "cow", "diningtable", "dog", "horse",

           "motorbike", "person", "pottedplant", "sheep", "sofa", "train", "tvmonitor"]

# Colors for each class

COLORS = np.random.uniform(0, 255, size=(len(CLASSES), 3))

# Initialize a counter for people

people\_count = 0

while True:

    ret, frame = cap.read()

    if not ret:

        break

    # Prepare the frame for object detection

    (h, w) = frame.shape[:2]

    blob = cv2.dnn.blobFromImage(cv2.resize(frame, (300, 300)), 0.007843, (300, 300), 127.5)

    net.setInput(blob)

    detections = net.forward()

    current\_count = 0

    # Loop over the detections

    for i in np.arange(0, detections.shape[2]):

        confidence = detections[0, 0, i, 2]

        # Filter out weak detections

        if confidence > 0.2:

            idx = int(detections[0, 0, i, 1])

            # Only consider the "person" class

            if CLASSES[idx] == "person":

                current\_count += 1

                box = detections[0, 0, i, 3:7] \* np.array([w, h, w, h])

                (startX, startY, endX, endY) = box.astype("int")

                label = "{}: {:.2f}%".format(CLASSES[idx], confidence \* 100)

                cv2.rectangle(frame, (startX, startY), (endX, endY), COLORS[idx], 2)

                y = startY - 15 if startY - 15 > 15 else startY + 15

                cv2.putText(frame, label, (startX, y), cv2.FONT\_HERSHEY\_SIMPLEX, 0.5, COLORS[idx], 2)

    # Update the people count if it changes

    if current\_count != people\_count:

        people\_count = current\_count

    # Display the count on the frame

    cv2.putText(frame, f"People Count: {people\_count}", (10, 30), cv2.FONT\_HERSHEY\_SIMPLEX, 1, (0, 255, 0), 2)

    # Show the output frame

    cv2.imshow("Frame", frame)

    # Break the loop on 'q' key press

    if cv2.waitKey(1) & 0xFF == ord('q'):

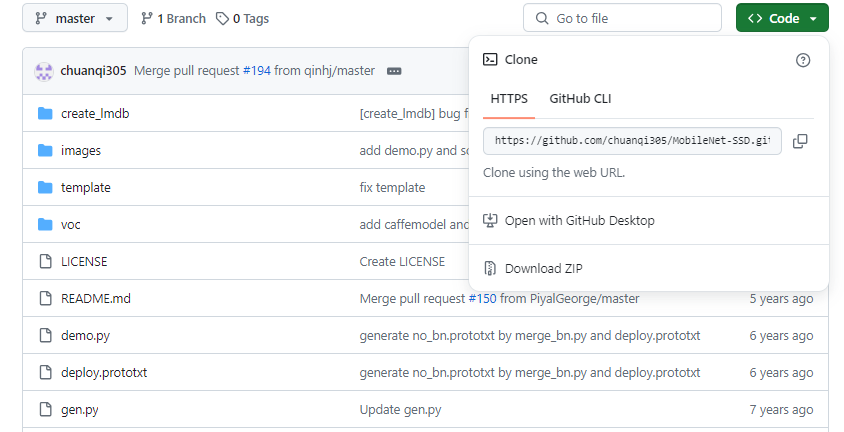
        break

# Clean up

cap.release()

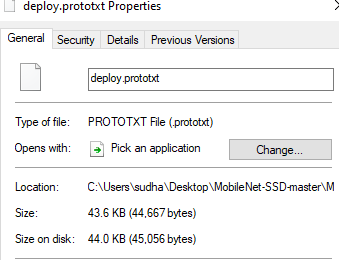
cv2.destroyAllWindows()

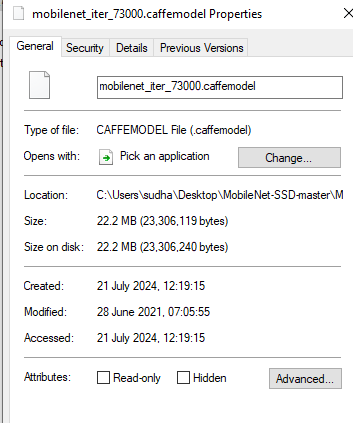
<https://github.com/chuanqi305/MobileNet-SSD>



Click Download ZIP

Extract this file desktop and





From Mobile-Net master folder copy the path of deploy.prototext file and caffemodel file and paste in cv2.dnn method.