Social Distance Monitoring

Team- **The Crew**

Coding:

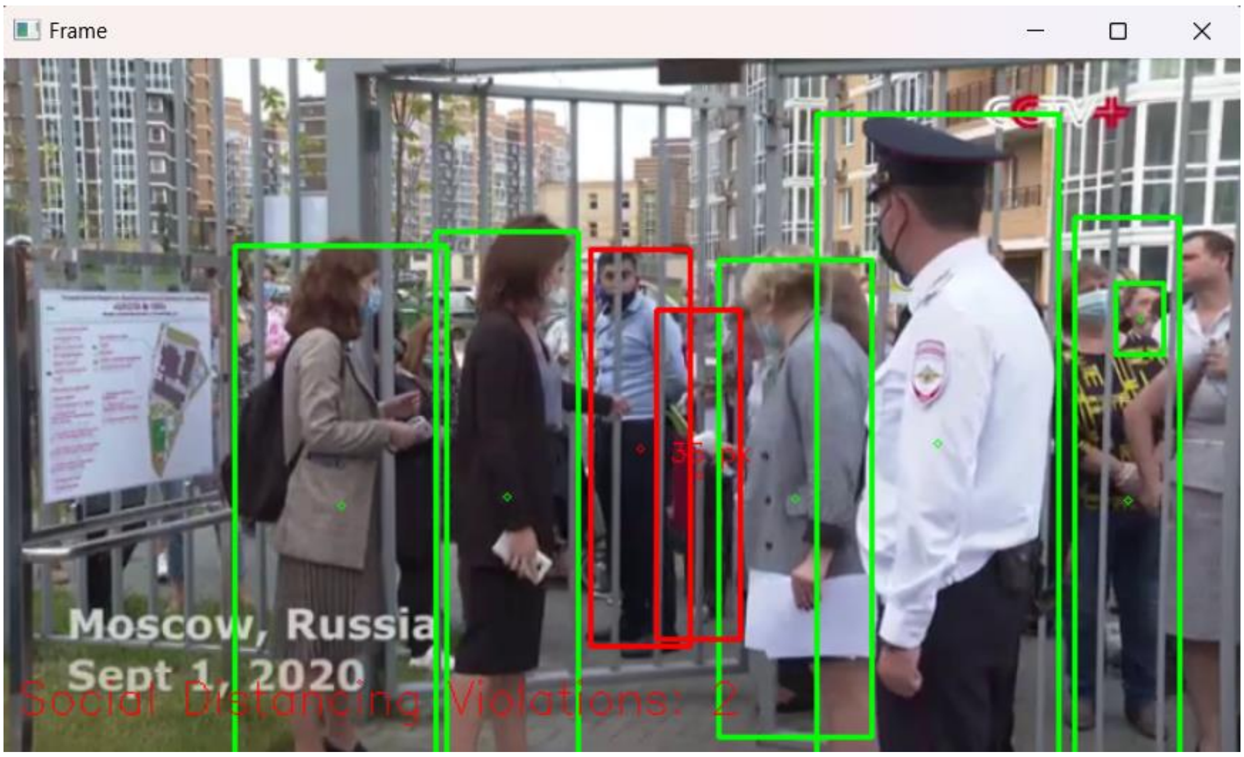
We have already stored a sample video in folder to recognize and detect a person and it shows distance between them.

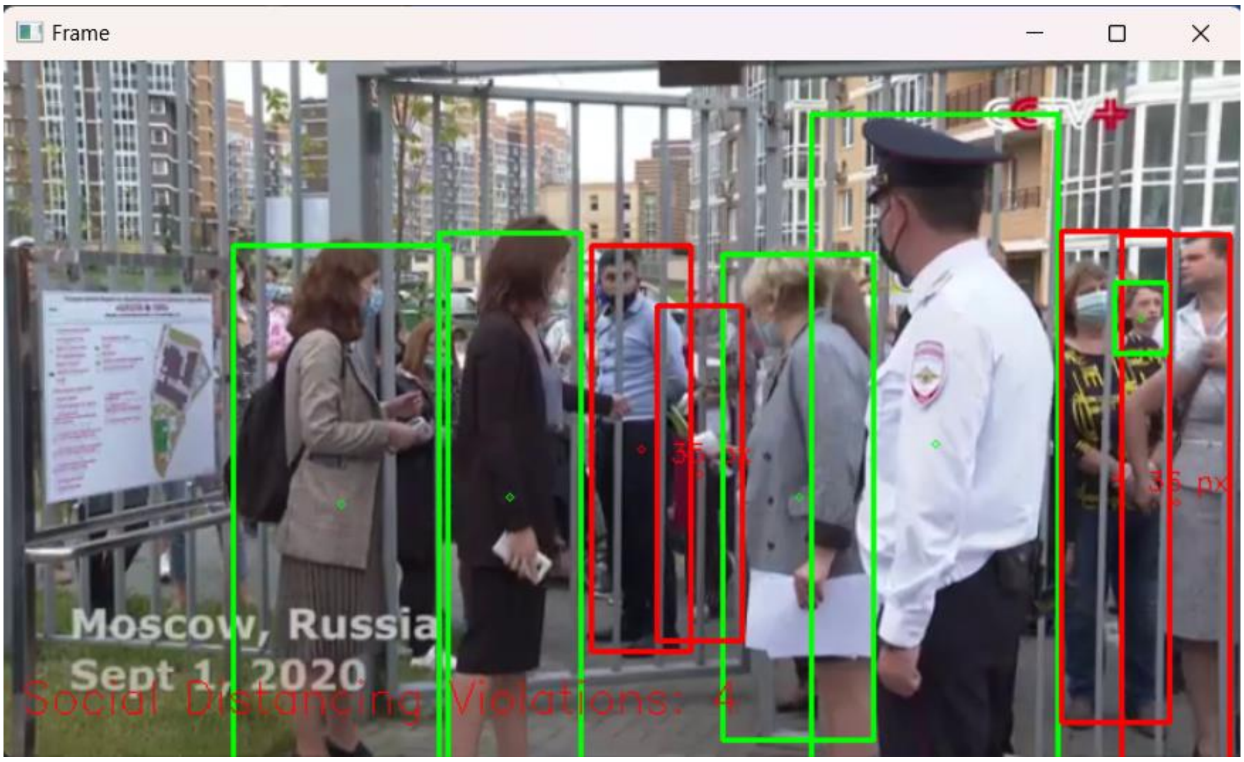
```
from scipy.spatial import distance as dist
import imutils
import numpy as np
import cv2

INPUT_FILE = "video2.mp4"
OUTPUT_FILE = "output.avi"
LABELS_PATH = "coco.names"
WEIGHTS_PATH = "yolov3.weights"
CONFIG_PATH = "yolov3.cfg"
MIN_CONF = 0.3
NMS_THRESH = 0.3
MIN_DISTANCE = 50

with open(LABELS_PATH) as f:
    labels = f.read().strip().split("\n")

yolo_net = cv2.dnn.readNetFromDarknet(CONFIG_PATH,
WEIGHTS_PATH)
layer_names = yolo_net.getLayerNames()
layer_names = [layer_names[i - 1] for i in
yolo_net.getUnconnectedOutLayers()]

video_stream = cv2.VideoCapture(INPUT_FILE)
writer = None

def detect_people(frame, net, ln, person_idx=0):
    height, width = frame.shape[:2]
    detections = []
    blob = cv2.dnn.blobFromImage(frame, 1 / 255.0, (416,
416), swapRB=True, crop=False)
    net.setInput(blob)
    layer_outputs = net.forward(ln)

    boxes = []
    centroids = []
    confidences = []

    for output in layer_outputs:
        for detection in output:
            scores = detection[5:]
            class_id = np.argmax(scores)
            confidence = scores[class_id]

            if class_id == person_idx and confidence >
MIN_CONF:
                box = detection[0:4] * np.array([width,
height, width, height])
                center_x, center_y, box_width, box_height =
box.astype("int")
                top_left_x = int(center_x - (box_width /
2))
                top_left_y = int(center_y - (box_height /
2))

                boxes.append([top_left_x, top_left_y,
int(box_width), int(box_height)])
                centroids.append((center_x, center_y))
                confidences.append(float(confidence))

    idxs = cv2.dnn.NMSBoxes(boxes, confidences, MIN_CONF,
NMS_THRESH)

    if len(idxs) > 0:
        for i in idxs.flatten():
            x, y = boxes[i][0], boxes[i][1]
            w, h = boxes[i][2], boxes[i][3]
            result = (confidences[i], (x, y, x + w, y + h),
centroids[i])
            detections.append(result)

    return detections

while True:
    grabbed, frame = video_stream.read()
    if not grabbed:
        break

    frame = imutils.resize(frame, width=700)
    people = detect_people(frame, yolo_net, layer_names,
person_idx=labels.index("person"))

    violations = set()

    if len(people) >= 2:
        centroids_array = np.array([r[2] for r in people])
        distance_matrix = dist.cdist(centroids_array,
centroids_array, metric="euclidean")

        for i in range(0, distance_matrix.shape[0]):
            for j in range(i + 1,
distance_matrix.shape[1]):
                if distance_matrix[i, j] < MIN_DISTANCE:
                    violations.add(i)
                    violations.add(j)
                    # Draw the distance between violating
pairs
                    distance_text =
f"{int(distance_matrix[i, j])} px"
                    cv2.putText(frame, distance_text,
                                (int((centroids_array[i][0]
+ centroids_array[j][0]) / 2),
                                 int((centroids_array[i][1]
+ centroids_array[j][1]) / 2)),
                                cv2.FONT_HERSHEY_SIMPLEX,
0.5, (0, 0, 255), 1)

    for i, (prob, bbox, centroid) in enumerate(people):
        start_x, start_y, end_x, end_y = bbox
        c_x, c_y = centroid
        color = (0, 255, 0)

        if i in violations:
            color = (0, 0, 255)

        cv2.rectangle(frame, (start_x, start_y), (end_x,
end_y), color, 2)
        cv2.circle(frame, (c_x, c_y), 2, color, 1)

    text = f"Social Distancing Violations:
{len(violations)}"
    cv2.putText(frame, text, (10, frame.shape[0] - 25),
cv2.FONT_HERSHEY_SIMPLEX, 0.85, (0, 0, 255), 1)

    cv2.imshow("Frame", frame)
    key = cv2.waitKey(1) & 0xFF

    if key == ord("q"):
        break

cv2.destroyAllWindows()
video_stream.release()
```

if you want to download the code:
https://codeshare.io/deDwMZ

download pretrained model like
“yolov.3”,”yolov.weights”,”coco.names”,
sample videos .

Output:


Frame
Moscow, Russia
Sept 1, 2020
Social Distancing Violations: 2
35 px


Frame
Moscow, Russia
Sept 1, 2020
Social Distancing Violations: 4
35 px
36 px