Multimedia Tools and Applications (2024) 83:30613–30639
https://doi.org/10.1007/s11042-023-16614-0

Real-time social distance monitoring and face mask detection based Social-Scaled-YOLOv4, DeepSORT and DSFD&MobileNetv2 for COVID-19

Mohammed Lakhdar Mokeddem[1] · **Mebarka Belahcene**[1] · **Salah Bourennane**[2]

Received: 1 August 2022 / Revised: 17 May 2023 / Accepted: 21 August 2023 /
Published online: 8 September 2023

Abstract
COVID-19 has spread rapidly worldwide, despite the availability of vaccines, the fear of the World Health Organization continues due to the mutation of the Coronavirus. This is what prompted us to propose this work of social distance and wearing a face mask to fight against this pandemic to save lives. In this work, we propose a real-time four-stage model with monocular camera and deep learning based framework for automating the task of monitoring social distancing and face mask detection using video sequences. This work based on Scaled-You Only Look Once (Scaled-YOLOv4) object detection model, Simple Online and Real-time Tracking with a deep association metric approach to tracking people. The perspective transformation is used to approximate the three-dimensional coordinates with Euclidean metric to compute distance between boxes. The Dual Shot Face Detector (DSFD) and MobileNetv2 face mask model used to detect faces of people who violate or cross the social distance. Accuracy of 56.2% and real-time performance of 32 frames per second are achieved by the Social-Scaled-YOLOv4 (Social-YOLOv4-P6) model trained on the MS COCO dataset and Google-Open-Image dataset. The results are compared with other popular state-of-the-art models in terms of Mean-Average-Precision, frame rate and loss of values. The DSFD&MobileNetv2 facemask detectors trained on Wider Face and Real Face mask dataset achieves an accuracy of 99.3%. The proposed approach is validated on indoor/outdoor public images and video sequences such as wider face dataset, Oxford Town Center dataset and open access sequences.

Keywords COVID-19 · Detection and Localization · Deep Learning · Social Distancing · Scaled-YOLOv4 · Tracking

✉ Mohammed Lakhdar Mokeddem
mohammedlakhdar.mokeddem@univ-biskra.dz

Mebarka Belahcene
mebarka.belahcene@univ-biskra.dz

Salah Bourennane
salah.bourennane@fresnel.fr

[1] RB_IAIM, LI3C, M. Khider University, Biskra, Algeria

[2] GSM, Fresnel Institut. Ecole Centrale, Marseille, France

1 Introduction

The report n.48 of the World Health Organization (WHO) noted that COVID-19 disease 2019 has globally infected over 58 million people and caused over 1.4 million deaths (9 April 2021). With this outbreak of COVID-19 coronavirus, many countries or we can say that all countries were obliged to commence new rules for social distance and face mask wearing. The governments have obliged hospitals and different organizations to use new infection interference measures to prevent the spreading of COVID-19 because its transmission rate is increasing. However, the transmission rate could vary per the measure and policies applied by the governments. As COVID-19 is transmitted through airdrops and shut contact, governments have started using new rules forcing individuals to prevent people from setting close to each other and wear a face mask to scale back the transmission and spreading rate. New mutated versions of the coronavirus took hold after the relaxation of many countries in adhering to safety rules (Indian, Nigerian …), which made The WHO recommend the use of personal protective equipment (PPE) among people in health care settings.

The spread of COVID-19 affected people's lives and disrupted the economy. It considered as major problem of public health and economy. The transmission of the COVID-19 virus spreads more easily in close contact and crowded environments. Countries need guidance and surveillance of people in crowded environments and public areas, incredibly packed to ensure that social distance and wearing face masks laws are applied. This could be achieved through video surveillance systems to obtain video sequences and deep learning models to detect human faces. However, most social distance applications and current research concerning social distance tasks solve the social distance problem but ignore wearing face masks. The lack of research will lead to the virus spreading by people who do not wear a face mask.

This study refers to the protection motivation theory, which is adaptable to both health-related and technology-related motivations. The concept of social distancing, risk persons and no mask or incorrectly face mask detections are added.

The aim of this study is to find an approach that can detect and track COVID-19 risk or contact people. The proposed approach is based on the unmasked or incorrectly masked faces detection using deep learning (DL) and social distancing. The use of this approach and the provision of health related data requested will increase our understanding of the concerns for the protection of people from the COVID-19 pandemic and play an important role in prevention. The objective is to find an automatic, efficient and rapid model, which could then be improved by strategies oriented towards the public of appropriate decisions.

Our contribution consists in associating several methods based on theSocial-Scaled-YOLOv4 model to create a detection, tracking and social distance system in order to prevent the spread of COVID-19. Detection technics for the couples of persons using DeepSORT tracker and a new face mask detection model namely DSFD&MobileNetv2 detection are proposed.

The proposed approach framework based on Social-Scaled- YOLOv4&DeepSort. The remainder of this paper is organized as follows. Section 2 introduces the most original recent works. The proposed method is described in Section 3 and the implementation platform and libraries are presented in Section 4. The experimental results and discussions are presented in Section 5. Section 6: approach limitations, Section 7: comparison with state of the art and Section 8: concludes this paper.

2 Related works

2.1 Recent methods of detection and localization based DL (Deep Learning)

In this paper, we relied on two surveys focus on DL approaches for detection and localization of objects, which can be found in [1, 2]. For face detection, we based on [3–5]. State of the art object detectors uses DL, which are divided in to two categories. The first one called two-stage detector models, of which the famous models are RCNN (Recurrent Convolutional Neural Network) [6], Fast RCNN [7], Faster RCNN [8], which starts with Region Proposal Network (RPN) to generate regions of interests and then performs the classification and bounding box regression. The second named one-stage detectors, of which the famous models are YOLO (You Only Look Once) [9], YOLOv2 [10], YOLOV3 [11], YOLOv4 [12], Scaled-YOLOv4 Scaling: Cross Stage Partial (CSP) [13], Single Shot Multibox Detector (SSD) [14], RetinaNet [15], and EfficientDet [16]. The most popular one stage model is YOLO; Fig. 1 shows the timeline and the comparison between members of the YOLO family models and their performance. The evaluation of models was usually based on two datasets, Pascal VOC [17] and MS-COCO (Microsoft Common Objects in Context) [18], the results are given in Table 1.

2.2 Recent methods COVID-19 social distancing based DL

An automated framework to monitor social distancing using surveillance video is presented in [18]. It uses YOLOv3 object detection model for detecting people and drawing the bounding boxes around them. It also compares the results with faster RCNN and SSD models through parameters like loss and FPS. Its advantages are that it presents deep

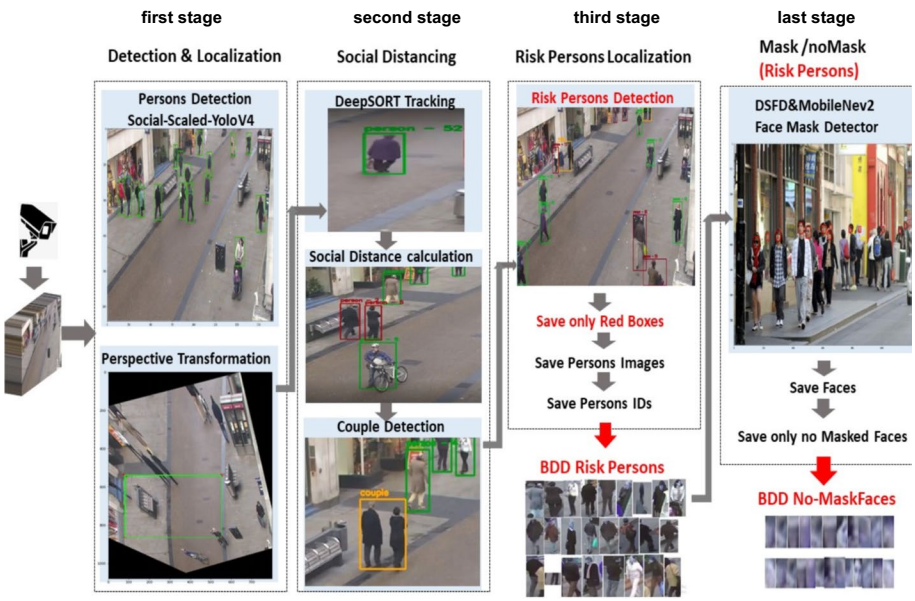

Fig. 1 Overall structure of proposed monitoring and face mask detection system

Table 1 Object detection models and their accuracies

Dataset-name	VOC [12]		MS-COCO [13]
Model-Name	m AP	FPS	m AP
RCNN [6]	0.53	0.5	-
FastRCNN [7]	0.68	7	0.19
FasterRCNN [8]	0.70	19	0.22
SSD [14]	0.75	45	0.27
YOLOv3 [11]	0.75	47	0.33
YOLOv4 [12]	0.79	62	0.43
RetinaNet [15]	-	-	0.415
YOLOv4-CSP [13]	-	-	0.462
EfficientDet-D0 [16]	-	-	0.346

comparative study between different models and it uses l2 norm to identify clusters of people not obeying social distancing.

M Rezaei and M Azarmi [20] proposed a work sous title “Deep-social: Social distancing monitoring and infection risk assessment in covid-19 pandemic”. In this paper the authors proposed a DL social distance system which utilizes a webcam as the source and Deep-social detector passed yolov4 detection model and euclidian distance to measure social distance between peoples. This paper supply also a magnificent visualization using many of tools such heat-maps, moving-trajectory …

Yang Yurtsever and Renganathan [21] proposed "a vision-based social distancing and critical density detection system for COVID19". This paper uses DL based real-time object detectors to measure social distancing. It uses: i) pre-trained models (YOLOv4 and Faster-RCNN) to detect persons with bounding boxes in each frame and ii) bird’s-eye view coordinates to transform the detected boxes in the image domain into real world domain.

Bharathi and Anandharaj [22] proposed Real-time DL framework to monitor social distancing using improved SSD based on overhead position. It is an efficient real-time monitoring of people to detect safe social distancing in public places. This model uses: i) improved SSD with Transfer Learning (TL) to detect persons with bounding boxes in each frame.

Meivel Sindhwani et all [23] proposed a work sous title “Mask Detection and Social Distance Identification Using Internet of Things and Faster R-CNN Algorithm”. This paper uses DL to monitor the social-distance between peoples to ensuring the safety in public-areas based Faster R-CNN detection model. This method is integrated in unmanned aircraft systems (drone) based Raspberry Pi4.

MD Elamin firoz et al. [46] proposed “Object Detection and Classification from a Real-Time Video Using SSD and YOLO Models” This research introduces an improved real-time object detection and recognition technique from web camera video. The technique detects and recognizes objects like people, vehicles, and animals. We use Single Shot Detector (SSD) and You Only Look Once (YOLO) models, which show promising results in object detection and recognition. This system can detect objects even in adverse and uncontrolled environments. We use convolutional neural network (CNN) for object classification. This technique provides real-time performance with satisfactory detection and classification results and better accuracy. This proposed model has an accuracy percentage of 63–90% in object detection and classification.

ML Mokeddem et all [47] proposed "COVID-19 risk reduce based YOLOv4-P6-Face-Mask detector and DeepSORT tracker" In this research the authors proposed a new high performance two stage facemask detector and tracker with a monocular camera and a deep learning based framework for automating the task of facemask detection based Scaled YoloV4 model (YOLOv4-P6-FaceMask) and tracking based DeepSORT tracker using video sequences. YOLOv4-P6-FaceMask is a model with high accuracy that achieves 93% mean average precision, 92% mean average recall and the real-time speed of 35 fps on single GPU Tesla-T4 graphic card.

Table 2 illustrated the difference between models.

2.3 Contributions

A new detection and social distance system is proposed in this paper to prevent the spread of COVID-19. Several methods based on the Scaled-YOLOv4 model are fully exploited. The main contributions are:

1. Detection of risk persons with bounding boxes in each frame.
   - Collection of indoor / outdoor sequences
   - Adaptation of a new version of the Scaled-YOLOv4 model (Social-Scaled-YOLOv4) for persons detection
   - Application to images and sequences in real time
2. Risk persons localization based perspective transformation
   - Use of perspective transformation technique (birds-eye view)
   - Extraction of 3D coordination using monocular camera based on perspective transformation
3. Social distance computation
4. Coupled people detection
   - Couple detection is passed on space and time
   - Distance between two people remains less than the permissible limit for social distancing (1.8 m) for an approximate period of 10 s.
5. DL detecting and tracking persons without face mask
   - Collection a dataset with face masks and without face masks
   - Propose a new face mask detector namely DSFD&MobileNetv2

Table 2 State-of-art social distance and facemask framework based deep learning (*SD;* Social Distancing *Dm: detection model Tr: tracking model S_ no_ MF: Masked faces detection)*

Model	*SD*	*Dm*	*Tr*	*MF*
M.Rezaei [20]	Yes	yolov4	Yes	no
Yang Yurtsever[21]	Yes	Faster-RCNN	no	no
Bharathi [22]	Yes	SSD	no	no
Meivel Sindhwani [23]	Yes	Faster-RCNN	no	no
MD Elamin [46]	Yes	YOLOv4	no	no
Ours	Yes	Scaled- Yolov4	Yes	Yes

6. Create risk persons database
7. DL detector to detect masked / no masked faces
8. Create masked / no masked faces database
9. Save persons breaching social distance norms (pedestrians and faces) for identification and tracking.

3 Proposed approach

The proposed method is depicted in Fig. 1. We propose a four-stage model including pedestrian detection, tracking, inter distance estimation as a solution for social-distance monitoring and face mask detection. The proposed system can be integrate on CCTV surveillance cameras with any resolution with an acceptable real-time performance. Social-Scaled-YOLOv4 (Social-YOLOv4-P6) is trained for pedestrian detection to identify human bodies in real-time video or online cameras, and then the extraction of 3D coordination is assured by perspective transformation method. The distance between centers of every box is computed using Euclidian distance and DeepSORT for persons tracking. Finally, persons faces are detected using DSFD model trained in Wider-Faces and MobileNetv2 classifier for mask classification. The main reason for using transfer-learning networks is that they provide excellent results in terms of accuracy and speed. In addition, the datasets used to train the model are huge MS-COCO dataset and Google-Open-Image dataset, which minimize the errors, the training time and prevent the models from over fitting. The dataset used for DSFD&MobileNetv2 is a collection of 5740 images belonging to two classes: “with mask” and “without mask” from the Real-World Masked Face dataset (RMFD) and Simulated Masked Face dataset (SMFD).

The majority of the sequences and images are datasets like Oxford Town Center (OTC) [24], Multiple Object Tracking (MOT) Dataset [25], or downloaded from the site: pixabay (https://pixabay.com/). The rest of used sequences are public and used already in references [22].

3.1 Detection and localization

The main objective of the first stage is to develop a robust pedestrian detection model. YOLO model belongs to the family of One-Stage Detectors, it is an object detection model used in DL use cases. In this paper, we will not talk about the history or background of previous versions of YOLO (v1, v2, and v3). Figure 2 shows all parties the overall structure of the one stage model YOLOv4. Which suggests a detection network with a backbone, a neck and heads. The CSPDarknet53 [26] is applied as a backbone and refers to a general feature extractor made up of CNN to extract informations in images to feature maps. Spatial Pyramid Pooling (SPP) [27] and Path Aggregation Network (PAN) [28] were applied as a neck. The SPP used to eliminate the requirement of fixed-size (e.g., 512×512) input image, the PAN used to collect multi-level features and connect with the spatial pyramid network and the YOLOv3 used as a head to predict the bounding box (calculate the coordinates, confidence threshold, non-maximum value suppression).

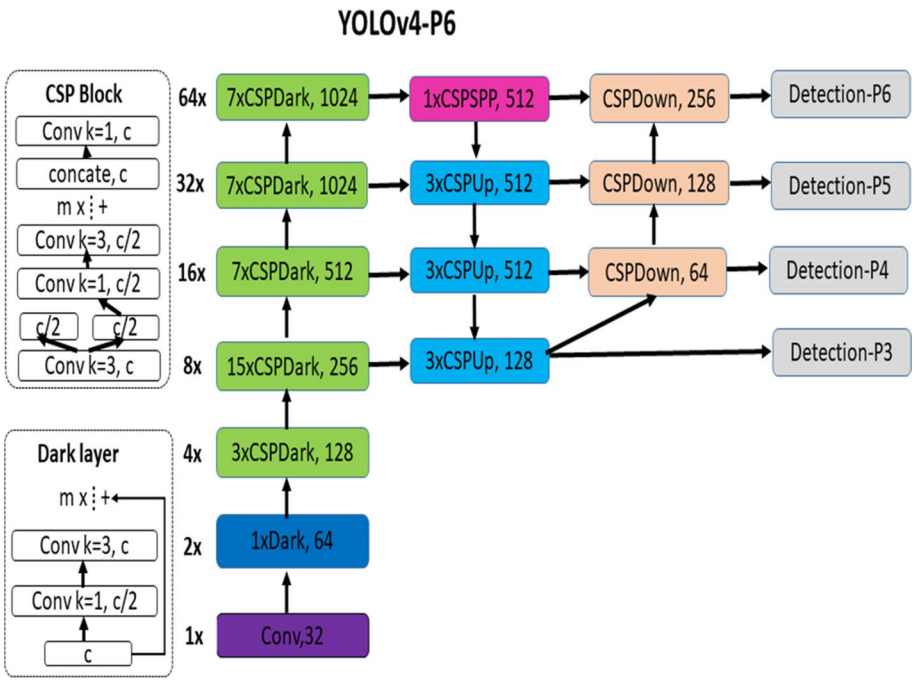

Fig. 2 Scaled-YOLOv4 detection Model

3.1.1 YOLOv4 scaling

In traditional models of detection, the scaling means modifies the depth of model by add more convolutional layers. For example, the VGGNet [29] scaled to VGG-11, VGG-13, VGG-16, and VGG-19 architectures. Now the scaling approach mean modifies the depth, width, resolution, and structure of the network, which finally forms scaled model for example Scaled YOLOv4. To prove the superiority of the selected model YOLOv4-P6 in terms of backbone, accuracy and real-time performance in this paper, we compare it with Fast-RCNN, Faster-RCNN, YOLOv3, YOLOv4, YOLOv4-CSP, SSD, RetinaNet, EfficientDet-D1, EfficientDet-D0 YOLOv4-P5 and YOLOv4-P7, which are the state-of-the-art pedestrian detection models. Table 3 shows the training parameters of Social-YOLOv4-P6 model on the MS- COCO and Google-Open-Image dataset. The comparison of testing results with state-off -art are given in Table 4.

3.1.2 Social-Scaled-YOLOv4

Training dataset To have a strong and robust pedestrian detector, we would need a group of training datasets that include different difficulties of image processing like blurring, distance between faces and camera, variety of gender or age, with annotation and labelling. We chose to used two datasets, MS-COCO and Google-Open-Image dataset.

Table 3 Training parameters of proposed Social-Scaled-YOLOv4 on the MS-COCO and Google-Open-Image dataset

Parameters	Value
Width	1280
Height	1280
Momentum	0.949
Learning rate	0.001
Batch_size	64
Subdivisions	8
Activation function	mish
Classes	80
Mini-batches	16,000
Weight decay	0.0005

Table 4 Comparison of the speed and accuracy Social-YOLOv4-P6 on the MS-COCO dataset

Model name	Backbone	Map	FPS
Fast RCNN	-	19	
Faster RCNN	-	22	3
SSD	VGG-16	27	10
YOLOv2	-	22	-
YOLOv3	Darknet-53	33	31
YOLOv4	CSPDarknet-53	43.5	62
RetinaNet	S96	44.3	42
YOLOv4-CSP	CSPDarknet-53 s	47.5	97
EfficientDet-D1	EfficientNet-B1	40.5	74
EfficientDet-D0	EfficientNet-B0	34.6	97
ASFF	Darknet-53	42.4	46
YOLOv3-SPP	Darknet-53	42.9	73
YOLOv4-P5	CSP-P5	51.8	43
YOLOv4-P6	CSP-P6	54.5	32
YOLOv4-P7	CSP-P7	55.5	17
Social-Scaled-YOLOv4(ours)	CSP-P6	56.2	32

Social-Scaled-YOLOv4 model parameters and results In our proposed approach, illustrated in Fig. 1, we use the Scaled YOLOv4 detection technique to detect persons in single pictures, real-time video, or online cameras (**first stage**). This paper will not discuss the history or background of previous versions of YOLO (YOLOv1, YOLOv2, and YOLOv3). We trained a custom YOLOv4-P6 model for persons detection and localization by using MS-COCO dataset. the network architecture of Scaled-YOLOv4 illustrated in Fig. 2.

Training parameters of proposed Social-Scaled-YOLOv4 shows in Table 3.

For the pedestrian detection task, fifteen different object detection models in the Tensorflow object model.

Zoo were trained and tested on the MS-COCO dataset to compare their accuracy with ours proposed model to assure the superiority of Social-Scaled-YOLOv4 model. Table 4 shows the models and their accuracies.

3.2 Persons tracking

The **second stage** after the people detection phase is track people and ID assignment for each box using the DeepSORT technique (Fig. 3).

DeepSORT [30] is an online algorithm for the track of people that considers both the information about the manifestation of the tracked objects and the bounding box parameters of the detection results to associate the detections in the frame at time t+1 with tracked objects at a time t. Therefore, DeepSORT needs to process the whole video at once, only considers information about the present and former frames to form predictions about the present frame.

At the first frame of the sequence the algorithm assigned to every bounding box that represents a poeple and has a confidence value above a set threshold. The Hungarian algorithm (combinatorial optimization algorithm) is used to assign the detections during a new frame to existing in order that the assignment cost function reaches the global minimum.

The cost-function involves the M-D (Mahalanobis-distance) (Eq. (1)) of the box that detected from the position predicted with the known position at time t of that object, and a visible distance (Eq. (2)) that considers the looks of the detected object and therefore the history of the looks of the tracked object.

The expression of M-D $d^{(1)}$ is given by:

$$d^{(1)}(i,j) = (d_j - y_i)^T S_i^{-1} (d_j - y_i) \tag{1}$$

where:

y_i the mean.

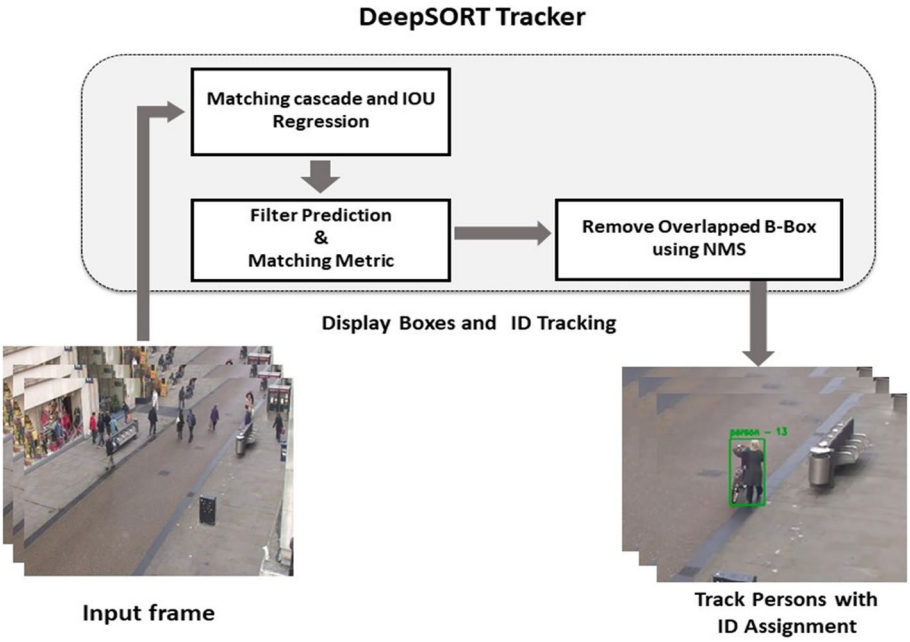

Fig. 3 DeepSORT Tracking technique

S_i the covariance matrix bounding box observations for the i-th track.
d_j the j-th detected bounding box.

The expression of visual distance $d^{(2)}$ that relies on appearance feature descriptors:

$$d^{(2)}(i,j) = \min\left\{ 1 - r_j^T r_k^{(i)} \middle| r_k^{(i)} \in \mathfrak{R} \right\} \tag{2}$$

where:

r_j the appearance descriptor extracted from the part of the image within the j-th detected bounding box.
$\mathscr{R}$ the set of last 100 appearance descriptors $r_k^{(i)}$ associated with the track i.

The cosine-distance used by $d^{(2)}$ measure between the j-th detection / i-th track in the current detection to select the track where visually the most similar detection was previously found.

The value function of assigning a detected object j to a track i is given by:

$$c_{i,j} = \lambda d^{(1)}(i,j) + (1-\lambda) d^{(2)}(i,j) \tag{3}$$

where:

λ parameter that can be set to determine the influence of the visual distance $d^{(2)}$ and the M-D $d^{(1)}$.

New track IDs are generated whenever (Fig. 3). There are more detections in a frame than already tracked persons.

A detection cannot be assigned to any track, because the detection is too far from any track, or not visually similar to any previous detection.

3.3 Distance computation

The third stage after the people tracking is the distance estimation between boxes. Such as in, the binocular stereo vision that uses two cameras of the same specification instead of human eyes is a popular technique for distance estimation but it is not appropriate for our application. We discussed here, the use of a single camera in the surveillance systems. This prompted us to search for a solution that enables us to calculate the distance between people, but by using a monocular camera, we were able to solve this problem using a technique called perspective transformation or birds-eye view.

3.3.1 Perspective transformation

The projection of a 3D scene world by employing a monocular camera into a 2D perspective image plane results in unrealistic pixel-distances between the objects, this method named Perspective Transformation (PT) or bird's eye technique, we will change the attitude of a given image or video for recuperating insights about the specified information. In PT, we would like to provide the points on the image from which we want to collect informations by changing the attitude (need a 3×3-transformation matrix). Straight lines will remain straight even after

the transformation. To seek out this transformation matrix four-point transformation method are used, where, the 4 points are within the order of top left, top right, bottom right, bottom left of the bounding box. PT and warp perspective methods from cv2 are used and the Euclidean distance criterion to evaluate inter-people distance is calculated. Figure 4 shows the original image captured from a perspective to the vertical view of bird's eye, where the dimensions in the picture have a linear relationship with real dimensions. The relationship between pixel (x, y) in the bird's eye picture and pixel (u, v) in the original picture is defined as:

$$\begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = M \begin{bmatrix} u \\ v \\ z \end{bmatrix} \tag{4}$$

$$\begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = \begin{bmatrix} m_{11} & m_{12} & m_{13} \\ m_{21} & m_{22} & m_{23} \\ m_{31} & m_{32} & m_{33} \end{bmatrix} \begin{bmatrix} u \\ v \\ z \end{bmatrix} \tag{5}$$

where **M** in Eq. (4) is the transformation matrix. Finally, the distance between each pair of people is measured by estimating the Euclidean distance (l_2-norm) between the center points of each boundary box in the bird's eye view.

3.3.2 Euclidean distance

L_2-norm distance (Eq. 6) is the shortest distance between two points (x_i, y_i) and (x_j, y_j) in a Euclidean space (two-dimensional space). It is used as a standard metric to measure the similarity between two data points and utilized in various fields.

$$\left(\left(x_i - x_j\right)^2 + \left(y_i - y_j\right)^2 \right)^{1/2} = d \tag{6}$$

The approximation of the number of pixels in an image that represents 1.8 m in real-world changes from dataset to other. Example: in the Oxford Town Center (OTC)

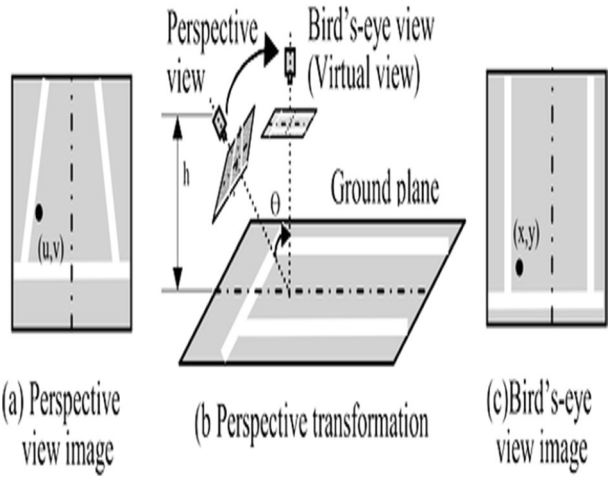

Fig. 4 Perspective Transformation

Fig. 5 Distance Risk (< 1.8 m)

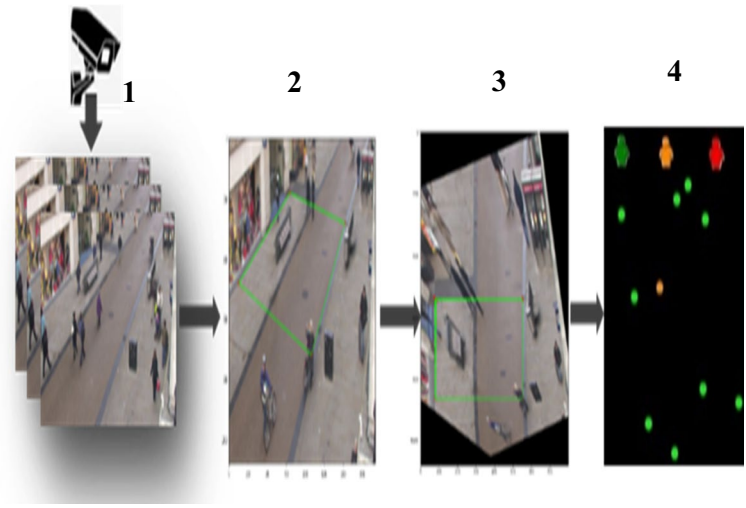

Fig. 6 Perspective Transformation (with couple detection). Oxford Town Centre dataset

dataset (Fig. 5) every 10 pixels, in the Bird Eye View space is equal to 0.98 m in the real-world. Therefore, 1.8 m in the real-world is equal to 19 pixels in the Bird Eye View space.

Figure 6 represents the bird eye blocs and steps:

1. Input real time video sequences
2. Four points representation (plot)
3. Output image after the application of perspective transformation
4. Bird eye view after detection of violation persons (red point: violated persons, orange point: coupled persons, green point: saved persons).

The output in Fig. 7 shows is the result of the proposed method without couple detection (no orange point) only red and green points (red points: violated persons, green points: saved persons).

3.4 Persons couple detection

How to deal with couples and family members when tracking social distance monitoring is one of the most important ideas offered by authorities. Some researchers advising the couples and family members to walk in a close proximity without being counted as a breach of social distancing countries, this is what encouraged some countries to establish new laws allowing family members to walk together, such as some countries in the European Union region and UK. We can notice that the current research of social distancing applies on every single individual but ignores couples and family members to walk together without considering it as a breach of social distancing.

The proposed technique for detecting couples of persons (Fig. 8) is based on ID number for each person, that we got by DeepSORT tracking for all frames of sequence. If the distance between two boxes is smaller than allowed distance 1.8mas shown in algorithm 1, the couple of two IDs (IDbox1,IDbox2) is saved in a list of tuples named cpID.

$$cpID = \left[\begin{array}{c} (ID_{box_1}, ID_{box_1}), ..., (ID_{box_i}, ID_{box_j}) \\ \left| i \in \left[0, lenght\_of\_risk\_box\right], j \in \left[i+1, lenght\_of\_risk\_box\right] \right. \end{array} \right] \tag{7}$$

In the next frame, we count the repetition of every couple of ID in *cpID*, the explanation is given in Algorithm 2.

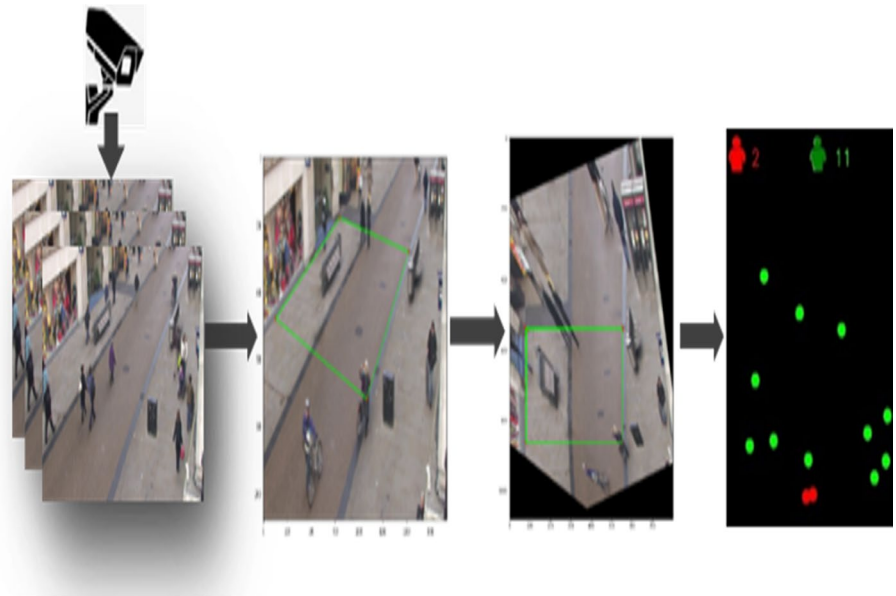

Fig. 7 Perspective Transformation (without couple detection)

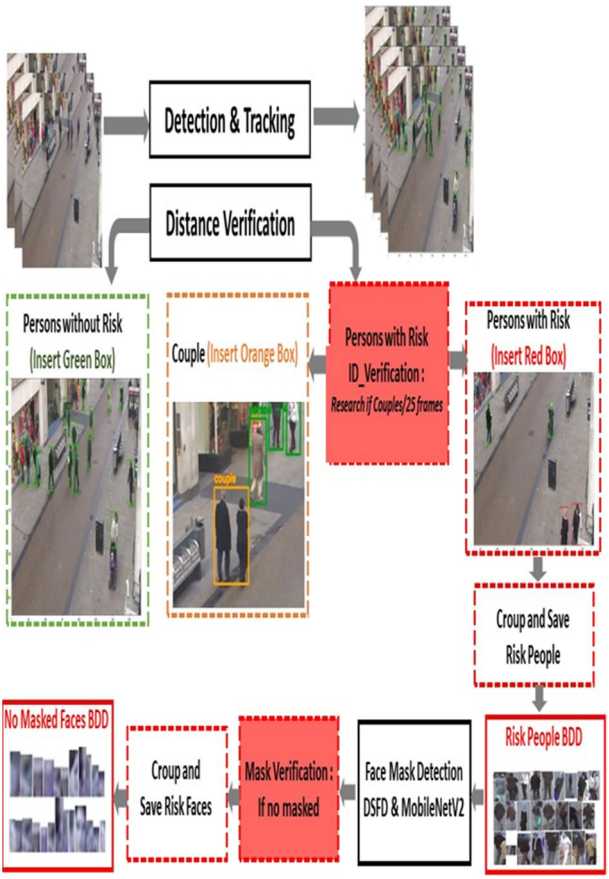

Fig. 8 Proposed technique of Detection and Tracking Risked/ Coupled Persons

3.5 Face mask detection

This section is divided into: i) Face detection with Dual Shot Face Detector DSFD [31], ii) Face Masked or unmasked with MobileNetv2 classifier. The structure of the proposed face mask detector is illustrated in Fig. 9.

3.5.1 Dual Shot Face Detector DSFD

In this section the first step for detect face in cropped person image DSFD model is used that inherits the architecture of SSD and introduces a Feature Enhance Module (FEM) for transferring the original feature maps to extend the single shot detector to dual shot detector. Specially, Progressive Anchor Loss (PAL), the model is trained on Wider Face dataset and its accuracy is equal to (easy: 0.966, medium: 0.957, hard: 0.904) and FDDB (discontinuous: 0.991, continuous: 0.862).

The Wider face contains 32,203 images and 393,703 faces with a high degree of variability in scale, pose and occlusion.

Algorithm 1: Green and Risk Boxes

input : $\boldsymbol{\beta}=\{b_1,\ldots,b_N\}$, ρ_t, zip_box=tuple(β)+tuple(ρ_t)
$\boldsymbol{\beta}$ is the list of all detected boxes
$\boldsymbol{\rho_t}$ is the list of birds eye points
output : *green_box, risk_box*

begin :

Initialization of variables : *allowed_distance , green_box={}, risk_box={}, cpID = {}*
for *variable* ***i*** *between 0 and length of* ***new box :***
  for *variable* ***j*** *between* ***i+1*** *and length of* ***new box :***
    distance *= the eucledianne distance between center of* ***new_box[i]*** *and* ***new_box[j]***
    if distance < allowed_distance :
      append ***new_box[i]*** *to* ***risk_box***
      append ***new_box[j]*** *to* ***risk_box***
      append (ID of ***new_box[j],*** *ID of* ***new_box[i])*** *to* ***cpID***
      append (ID of ***new_box[i],*** *ID of* ***new_box[j])*** *to* ***cpID***
    end if
  end for
end for
green_box *=* ***new_box*** *–* ***risk_box***
end

DSFD architecture uses the same extended VGG16 backbone as Pyramid Box [32] and S3FD [33] which is truncated before the classification layers and added with some auxiliary structures.

3.5.2 MobileNetv2 Classifier

The second step is face mask classification. For the face mask task, MobileNetv2 object classification models were trained and tested on a collected dataset, the face mask dataset named Simulated Masked Face Dataset [34] (SMFD) and Real Masked Face Dataset (RMFD) [35].

Data preprocessing and dataset Before the training of models, the step of image augmentation is done on collected dataset (SMFD [34] / RMFD [35]). This technique used to increase the size of dataset by artificially modifying. The images are augmented with distinct operations namely Shearing, Gaussian Blur, Average Blur, Motion Blur. The generated dataset is then rescaled to 224×224 pixels. An example is shown in Fig. 10.

Model training For training the model we load pre-trained MobileNetv2 model without last few layers and freeze all the layers, after we define the face mask classifier model by adding a few layers on top of MobileNetv2 pre-trained model, extract faces from the

Algorithm 2: Red and Orange boxes (coupled persons detection)

input : $\beta_r=\{b_1,\ldots,b_N\}$, *cpID*
$\boldsymbol{\beta_r}$ is the list of risk detected boxes
cpID is the list of tuples contain all couples of IDs of ***risk_box*** (output of algorithm 1)
output : *red_box, orange_box*

begin :

Initialization of variables : *cont=0,red_box={},*
not_red_box={}, ***orange_box={}***
for *variable* ***i*** *between 0 and length of* $\boldsymbol{\beta_r}$***:***
  for *variable* ***j*** *between* ***i+1*** *and length of* $\boldsymbol{\beta_r}$***:***
    A = $(\beta_r[i][0], \beta_r[j][0])$
    for *variable* ***w*** *between* ***0*** *and length of* ***cpID*** *:*
      if ***c_p_ID*** *[w] ==* ***A :***
        cont *= cont+1*
      end if
    end for
    if cont > 25
      Calculate coordinate of ***orange_box*** *ex****: x of orange box=min(****x of* $\beta_r[j]$ ***and*** $\beta_r[i]$ ***))***
      Append Calculate box to ***orange_box***
    end if
  end for
end for
red_box = β_r *– orange_box*
end

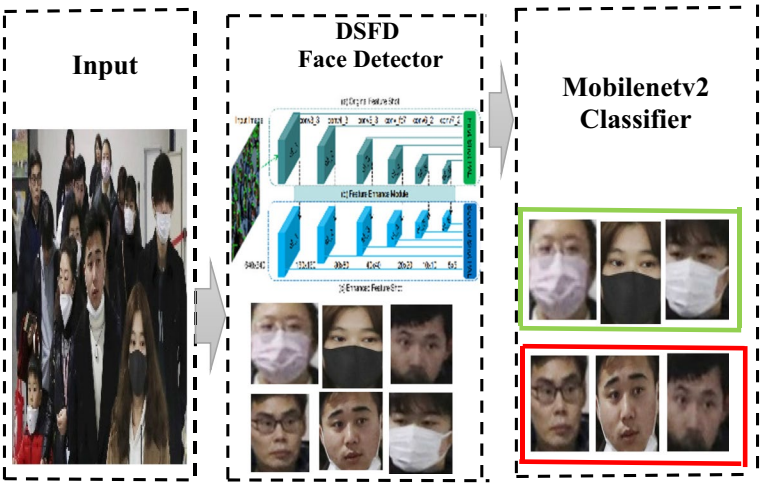

Fig. 9 Proposed DSFD&MobileNetv2 Face Mask Detector

Fig. 10 Images from SMFD dataset and RMFD dataset

dataset and save them in the specified directory, then should be two sub-directories corresponding to masked and no-masked faces. The parameters of model training are shown in Table 5 and the results of training in Fig. 11 which represent the progress in the training process and Fig. 12 which represent the test of DSFD&MobileNetv2 Mask Detector model with a set of public images.

Table 5 Training parameters of MobileNetv2 Face mask classifier

Parameters	Values
Width	224
Height	224
activation function	Relu
Learning Rate	0.0001
Epochs	20
Classes	2

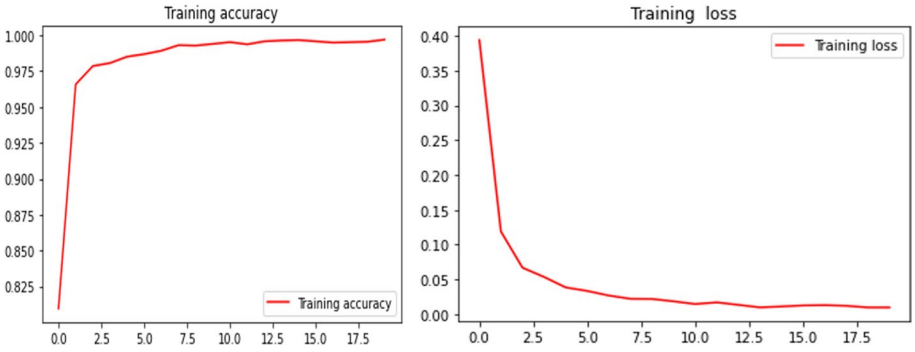

Fig. 11 MobileNetv2 classifier training results

Fig. 12 Output Results of DSFD&MobileNetv2 Mask Detector

3.5.3 Performance evaluation

For testing the performance of the DSFD&MobileNetv2 model, test part of collected dataset is used. We can see from Table 6 that the model achieves detection accuracy of **99.3%** and loss of **0.01%.** From the detection images obtained we can confirm that the DSFD&MobileNetv2 models detect correctly all indoor face images in front of the camera. They also detect all the faces of the outdoor images close to the camera as well as those, which are far from it with orientation of the head, bad resolution as well as the burring images.

4 Implementation platform and libraries

To implement the Social distance monitoring the Python language on Google Colab notebook and DESKTOP-DIPLV8E i5-3230 M, 2.60 GHz, and GeForce GTX 1080 Ti Graphics Cards—Nvidia are used. In the first stage, the weight of Social-Scaled-YOLOv4 converted from darknet format to TensorFlow format. The DSFD&MobileNetv2 face mask detection model trained and performed in a single Tesla T4 GPU of Google Colab. The

Table 6 Accuracy of trained DSFD&MobileNetv2 face mask detector

Model	Parameters	Accuracy %	Loss %
DSFD&VGG16	14,714,688	92.5	1.0
DSFD&VGG19	20,024,384	93.2	1.2
DSFD&Resnet50	23,587,712	94.4	0.1
DSFD&Mobilenet	3,228,864	95.9	0.2
DSFD&ResNet152	58,370,944	97.7	0.1
DSFD&MobileNetv2	2,257,984	99.3	0.01

libraries used in the implementation processes: Keras, Os, OpenCv, NumPy, MatPlotLib and pillow.

5 Results of proposed social distance and face mask monitor

Figure 13 provides a basic statistic about the number of persons every 100 frames from Oxford Town Center dataset, the number of people who break and don't break the rules of social distancing without taking into consideration coupled persons as violations.

The Fig. 13 is a 2D registration of the number of detected persons in 1000 frames from the Oxford Town Center Dataset, as well as the number of violations (Insafe) and number of safe persons.

5.1 Results: Social-Scaled-YOLOv4&DeepSORT on single images

Figure 14 shows the output of the proposed approach tested on single images. The big red boxes represent violate persons or persons break social distance rules.

If the big box is red, we pass to face mask detection. If the face is masked we insert a green face box, if the face is non-masked insert a red box faces. The results show the performance of the proposed approach for both cases indoor and outdoor.

The detection images obtained confirm that the proposed method for social distance and face mask is performing.

5.2 Results: Social-Scaled-YOLOv4&DeepSORT on video sequence

To evaluate the performance of this proposed solution in real time sequences; few tests are performed, in the series of experiments are shown in Fig. 15 and Fig. 16. The performance of the model is explored on outdoor and indoor sequences that contain difficulties and obstacles such as brightness, blurring, and proximity faces to camera,

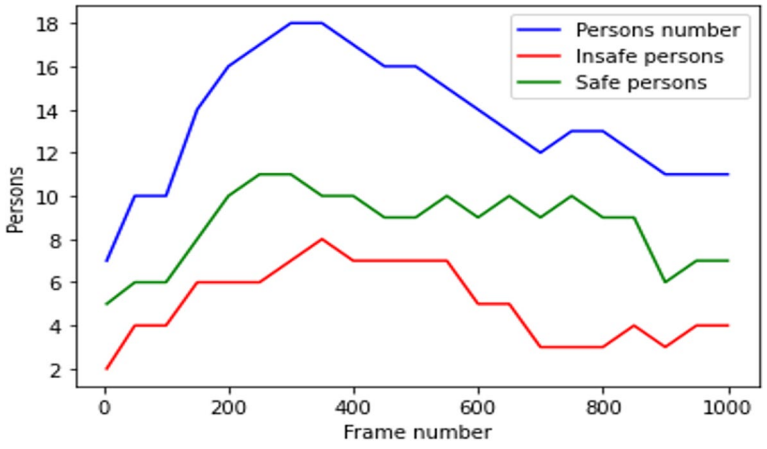

Fig. 13 Social-Scaled-YOLOV4&DeepSort Social Distance Results without Couples on Oxford Town Center dataset

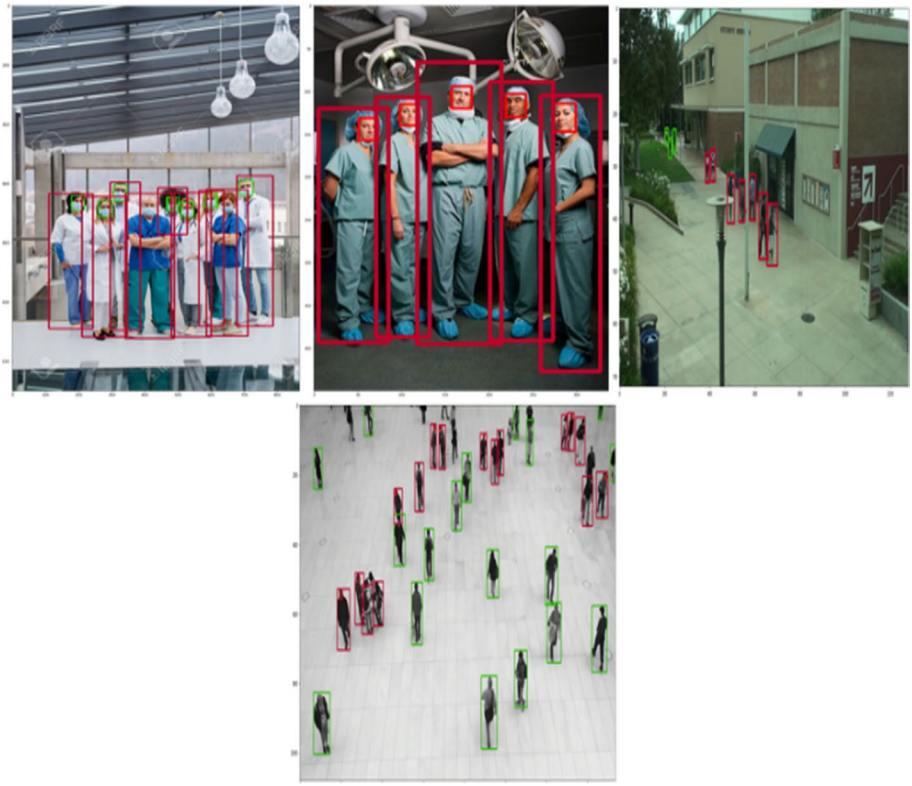

Fig. 14 The output of proposed Social-Scaled-YOLOv4&DeepSORT tested in single images

a) Oxford Town Center dataset

b) CCTV camera walk student

Fig. 15 Social-Scaled-YOLOv4&DeepSORT tested in outdoor sequences **a**) Oxford Town Center dataset **b**) CCTV camera walk student

a) Airport sequence without couple detection

b) Social distance detection MOT20 dataset and masked/no masked persons

c) Student in Chinese school from pixabay

Fig. 16 Social-Scaled-YOLOv4&DeepSORT tested in outdoor **a**) Airport sequence without couple detection **b**) Social distance detection MOT20 dataset and masked/no masked persons **c**) Student in Chinese school from pixabay

congestion (schools, airports, malls…) to show the effectiveness and accuracy of the study social distance monitor and face mask detector. The results are shown in Fig. 15 a), b) and Fig. 16a), b), c).

5.2.1 Social-Scaled-YOLOv4&DeepSORT in outdoor

Figure 15 shows the results of the proposed solution tested on outdoor sequences (Oxford Town Center Dataset, CCTV walk students, Persons walk in Bridge). Oxford Town Center camera calibration is available; this help us to extract the perspective transformation matrix. In the other sequence, 4-points are used to extract the perspective transformation matrix.

We notice that the model gives excellent results in outdoor environment that contain difficulties and obstacles such as brightness, blurring, and proximity faces to camera, and we can apply the detection of coupled persons.

Fig. 17 Results of the proposed Social-Scaled-YOLOv4 and DeepSORT save persons from Oxford Town dataset

Fig. 18 Results of the proposed Social-Scaled-YOLOv4&DeepSORT save faces from Oxford Town dataset

5.2.2 Social-Scaled-YOLOv4&DeepSORT indoor

Figure 16 shows the output of proposed solution tested on indoor sequences (MOT20 dataset, Student in Chinese school, airport sequences). In all sequence, 4-points technique are used to extract the perspective transformation matrix.

We notice that indoor crowded environment the detection application of coupled persons is very difficult and we apply social distance monitoring without couple detection. This helps us to conclude that the best option for indoor environments is to use social distance monitoring without coupled person detection.

5.3 Social-Scaled-YOLOv4&DeepSORT databases extraction

After the person detection, crop and save every person breaching social distance norms (red boxes) and the faces of persons that not wear a medical mask. The persons and faces are identified by the tracking ID number and number of frame in the sequence. For the faces, we save only faces of persons breaching social distance norms and not wearing face mask:

- Fig. 17 shows examples of persons cropped from Oxford Town Center dataset
- Fig. 18 shows examples of no-masked faces cropped from Oxford Town Center dataset
- Algorithm 3 illustrates the technique used to crop and save the picture of persons and faces only once using the tracking ID (identification number)

Algorithm 3: Save violate persons-unmasked face

Input : *red_boxes(id, x, y, w, h), i_p_s, i_f_s*
i_p_s is the list of all detected boxes
i_f_s is the list of birds eye points
Output *: file of violate persons, file of unmasked faces*
begin :
initialization of variables : *i_p_s={}, i_f_s={},*

```
for box in red_boxes :
    convert id to int
    if id not in i_p_s :
        append id to i_p_s
        crop box
        save box in persons output file
    end if
    if id not in i_f_s :
        append id to i_f_s
        crop box
        save box in faces output file
    end if
end for
End
```

6 Approach limitations

Bird eye view (Perspective Transformation) gives us a top view of a scene, this results in a close but inaccurate distance calculation. Figure 19 shows that the detection of coupled persons can be applied only in outdoor but in crowded indoor environment we cannot use this and we can use social distance monitoring without couple detection.

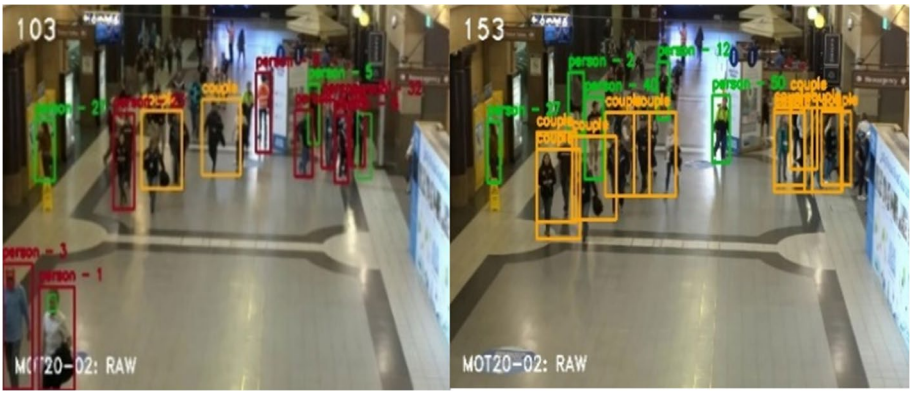

Fig. 19 Limits of Couple Detection (MOT20 Dataset)

Table 7 Options comparison of our approach social distance and face mask detection with state of the art

Authors	Models	Datasets	MAp
Kumar et al., 2021 [36]	Scaling YOLO	New dataset	71.69%
Loey et al., 2021 [37]	YOLO-V2 + ResNet50	MMD [38] + FMD [39]	81%
Mokeddem et al., 2021 [40]	YOLOV4	WiderFace [41] + FMD + RMFD	88.82%
Bala et al., 2021 [42]	MobileNetv2	RMFD	91.7%
Preeti Nagrath et al., 2021 [43]	SSD + MobileNetV2	RMFD + PyImageSearch	92.64%
Mingjie Jiang, 2021 [44]	RetinaFaceMask	Face Mask Dataset [45]	93.4%
DSFD&MobileNetv2(Ours)	DSFD + MobileNetv2	SMFD + RMFD	99.3%

SD Social Distance, *CD* Couple Distance, *MF* Masked Faces, *SRP* Save Risk Persons, *V* Visualization, *S_no_ MF* Save no Masked faces

Table 8 Comparison of our approach face mask detection with state of the art

Autors	Person Detector	Face Mask Detector	Options
Punn et al., 2020 [19]	YOLOV3	No	SD
Rezaei et al., 2020 [20]	DeepSocial(YOLOv4)	No	SD + CD + V
Yang et al., 2020 [21]	YOLOV4	No	SD + AA-V
Gopal et al., 2022 [22]	Improved SSD	No	SD
Meivel a et al., 2021 [23]	Faster Rcnn	Faster Rcnn	SD/ MF
Ours	Social-Scaled-YOLOv4	DSFD&MobileNetv2	SD + CD + MF + SRP + S_no_ MF

The use of three models (Social-Scaled-YOLOv4/DeepSORT/DSFD&Mobilenetv2) decreases the real time performance of the model.

7 Comparison with state of the art

In Tables 7 and 8, we present a comparison between the proposed approach and the state of the art. There is a comparison with the state of the art of published methods of social distance monitoring in Table 7, we can notice that the proposed approach offers a new option, that is, the use of DeepSORT to track persons and a technique to save risk pedestrians and no masked faces.

Table 8 provides the details of state-of-the-art model of face mask detection published in last two years. Notice that the proposed model:

- Provides an acceptable performance compared to the state of the art of face mask detection models;
- Improves the accuracy and real-time performance of DSFD&MobileNetv2; Uses the database extracted (violate persons) for person identification and risk person detection for a person's disease based on facial expression.

8 Conclusion

This paper developed a four stages real-time model based Deep Learning and monocular camera to monitor the social distance and face mask detection to avoid the spread of the corona virus and to assure persons safety in the COVID-19 pandemic. In the first stage, we used the Social-Scaled-YOLOv4 model for risk persons detection. In the second stage, DeepSORT is used for tracking people in the scene. Perspective transformation used to unrealistic pixel-distances between the persons. Then, the pairwise centroid distances between detected bounding boxes are measured with Euclidean distance. To check social distance violations between people, an approximation of physical distance to the pixel is used, and a threshold is defined. Distance less than threshold value and persons not coupled that means persons breaching social distance norms. Finally, face mask detection of these persons is realized by using the proposed DSFD&MobileNetv2 face mask detector. The result of this breaching is print red person box, save person boxes in a predefined folder, face mask detection is achieved by showing bounding boxes on the identified person face with mask (green box face), or no-mask (red box face). The Social-Scaled-YOLOv4 model achieves a detection accuracy of 56.2% on MS-COCO dataset and DSFD&MobileNetv2 model achieves a detection accuracy of 99.3%. Database of masked faces or no masked is created for identification.

In future work, we aspire to:

- To extract database to identify violate persons or infected persons based in facial expression.
- To combine tis monitoring system with another recent tracker
- to integrate this method in an embedded system (Arduino, raspberry Pi, ArduPilot …)

Data availability The datasets generated during and/or analyzed during the current study are available in the KAGGLE repository https://github.com/prajnasb/observations, https://pixabay.com/, MS-COCO (Microsoft Common Objects in Context) [18], MOT [25]

Declarations

Conflict of interest The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

References

1. Zou Z, Shi Z, Guo Y, Ye J (2019) Object detection in 20 years: A survey. arXiv preprint arXiv:1905.05055
2. Zhao ZQ, Zheng P, Xu ST, Wu X (2019) Object detection with deep learning: A review. IEEE Trans Neural Netw Learn Syst 30(11):3212–3232
3. Ameur B, Belahcene M, Masmoudi S, Hamida AB (2019) Efficient hybrid descriptor for face verification in the wild using the deep learning approach. https://doi.org/10.3103/S1060992X19030020
4. Belahcene M (2013) Biometric identification and authentification. Phd Thesis, Mohamed Khider University, Biskra
5. Elaggoune H, Belahcene M, Bourennane S (2020) Hybrid Descriptor Optimization for Face Recognition, Int. Conf. on Optimization and Learning, Cadiz, Spain, 17–19, OLA'2020

6. Girshick R, Donahue J, Darrell T, Malik J (2015) Region-based convolutional networks for accurate object detection and segmentation. IEEE Trans Pattern Anal Mach Intell 38(1):142–158. https://doi.org/10.1109/TPAMI.2015.2437384
7. Girshick R (2015) Fast r-cnn. In Proceedings of the IEEE international conference on computer vision. pp 1440–1448
8. Ren S, He K, Girshick R, Sun J (2016) Faster R-CNN: towards real-time object detection with region proposal networks. IEEE Trans Pattern Anal Mach Intell 39(6):1137–1149
9. Redmon J, Divvala S, Girshick R, Farhadi A (2016) You only look once: Unified, real-time object detection. In Proceedings of the IEEE conference on computer vision and pattern recognition. pp 779–788
10. Redmon J, Ali F (2017) YOLO9000: better, faster, stronger. Proceedings of the IEEE conference on computer vision and pattern recognition. p 7263–7271
11. Redmon J, Ali F (2018) YOLOv3: An incremental improvement. arXiv preprint arXiv:1804.02767
12. Bochkovskiy A, Wang C Y, Liao H Y M (2020) YOLOv4: Optimal speed and accuracy of object detection. arXiv preprint arXiv:2004.10934
13. Wang, Chien Y, Alexey B, Hong Y, Mark L (2021) Scaled-yolov4: Scaling cross stage partial network. Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition
14. Liu W, Anguelov D, Erhan D, Szegedy C, Reed S, Fu C Y, Berg A C (2016) Ssd: Single shot multibox detector. In European conference on computer vision. pp 21–37. Springer, Cham. https://doi.org/10.1007/978-3-319-46448-0_2
15. Lin TY, Goyal P, Girshick R, He K, Dollár P (2017) Focal loss for dense object detection. In Proceedings of the IEEE international conference on computer vision. pp 2980–2988. arXiv:1708.02002
16. Tan M, Pang R, Le Q V (2020) Efficientdet: Scalable and efficient object detection.In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp 10781–10790. arXiv:1911.09070
17. Everingham M, Van Gool L, Williams CK, Winn J, Zisserman A (2010) The pascal visual object classes (voc) challenge. Int J Comput Vision 88(2):303–338
18. Lin TY, Maire M, Belongie S, Hays J, Perona P, Ramanan D, Zitnick C L (2014) Microsoft coco: Common objects in context. In European conference on computer vision. pp 740–755. Springer, Cham. https://doi.org/10.1007/978-3-319-10602-1_48
19. Punn N S, Sonbhadra S K, Agarwal S, Rai G (2020) Monitoring COVID-19 social distancing with person detection and tracking via fine-tuned YOLO v3 and Deepsort techniques. arXiv preprint arXiv:2005.01385
20. Rezaei M, Azarmi M (2020) Deepsocial: Social distancing monitoring and infection risk assessment in covid-19 pandemic. Appl Sci 10(21):7514. https://doi.org/10.3390/app10217514
21. Yang D, Yurtsever E, Renganathan V, Redmill KA, Özgüner Ü (2021) A vision-based social distancing and critical density detection system for covid-19. Sensors 21(13):4608. https://doi.org/10.3390/s21134608
22. Gopal B, Ganesan A (2022) Real time deep learning framework to monitor social distancing using improved single shot detector based on overhead position. Earth Sci Inform, 1–18. https://doi.org/10.1007/s12145-021-00758-4
23. Meivel S et al (2022) Mask detection and social distance identification using internet of things and faster R-CNN algorithm. Comput Intell Neurosci 2022
24. Harvey A , LaPlace J (2019) Megapixels: origins, ethics, and privacy implications of publicly available face recognition image datasets. Megapixels 1(2):6
25. Agarwal A, Saurabh S (2017) Real-time* multiple object tracking (MOT) for autonomous navigation. Technical report. http://vision.stanford.edu/teaching/cs231n/reports/2017/pdfs/630.pdf
26. Wang C Y, Liao H Y M, Wu Y H, Chen P Y, Hsieh J W, Yeh I H (2020) CSPNet: A new backbone that can enhance learning capability of CNN. In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition workshops. pp 390–391
27. He K, Zhang X, Ren S, Sun J (2015) Spatial pyramid pooling in deep convolutional networks for visual recognition. IEEE Trans Pattern Anal Mach Intell 37(9):1904–1916
28. Girshick R, Donahue J, Darrell T, Malik J (2014) Rich feature hierarchies for accurate object detection and semantic segmentation. In Proceedings of the IEEE conference on computer vision and pattern recognition. pp 580–587
29. Simonyan K, Andrew Z (2014) Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv:1409.1556.
30. Wojke N, Bewley A, Paulus D (2017) Simple online and realtime tracking with a deep association metric. In 2017 IEEE international conference on image processing (ICIP). pp 3645–3649

31. Li J, Wang Y, Wang C, Tai Y, Qian J, Yang J, Huang F (2019) DSFD: dual shot face detector. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp 5060–5069
32. Tang X, Du D K, He Z, Liu J (2018) Pyramidbox: A context-assisted single shot face detector. In Proceedings of the European Conference on Computer Vision (ECCV). pp 797–813
33. Zhang S, Zhu X, Lei Z, Shi H, Wang X, Li S Z (2017) S3fd: Single shot scale-invariant face detector. In Proceedings of the IEEE international conference on computer vision. pp 192–201
34. SMFD (2020) github, [Online]. Available, https://github.com/prajnasb/observations Accessed 25 May 2020
35. Wang Z, Wang G, Huang B, Xiong Z, Hong Q, Wu H, Liang J (2020) Masked face recognition dataset and application. arXiv preprint arXiv:2003.09093
36. Kumar A, Kalia A, Verma K, Sharma A, Kaushal M (2021) Scaling up face masks detection with YOLO on a novel dataset. Optik 239:166744
37. Loey M, Manogaran G, Taha MHN, Khalifa NEM (2021) Fighting against COVID-19: A novel deep learning model based on YOLO-v2 with ResNet-50 for medical face mask detection. Sustain Cities Soc 65:102600
38. MMD (2020) Kaggle, [Online]. Available. https://www.kaggle.com/vtech6/medical-masks-dataset Accessed 20 Dec 2021
39. FMD (2020) Kaggle, [Online]. Available https://www.kaggle.com/andrewmvd/face-mask-detection Accessed 13 Jan 2022
40. Mokeddem M L, Belahcene M, Bourennane S (2021) YOLOv4Face mask: COVID-19 Mask Detector. International Conference on Cyber Management and Engineering (CyMaEn'21)
41. Yang S, Luo P, Loy C C, Tang X (2016) Wider face: A face detection benchmark. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 5525–5533).
42. Bala MMS (2021) A Deep Learning Technique To Predict Social Distance And Face Mask. Turk J Comput Math Educ (TURCOMAT) 12(12):1849–1853
43. Nagrath P, Jain R, Madan A, Arora R, Kataria P, Hemanth J (2021) SSDMNV2: A real time DNN-based face mask detection system using single shot multibox detector and MobileNetV2. Sustain Cities Soc 66:102692. https://doi.org/10.1016/j.scs.2020.102692
44. Jiang M, Fan X, Yan H (2020) Retinamask: A face mask detector. arXiv preprint arXiv:2005.03950
45. Chiang D (2020) Detect faces and determine whether people are wearing mask. Kaggle, 2020. [Online].
46. Feroz Md Alamin et al Object detection and classification from a real-time video using SSD and YOLO models. Computational Intelligence in Pattern Recognition: Proceedings of CIPR 2021. Springer Singapore, 2022
47. Mokeddem ML, Belahcene M, Bourennane S (2022) COVID-19 risk reduce based YOLOv4-P6-Face-Mask detector and DeepSORT tracker. Multimedia Tools and Applications, 1–25