Since January 2020 Elsevier has created a COVID-19 resource centre with free information in English and Mandarin on the novel coronavirus COVID-19. The COVID-19 resource centre is hosted on Elsevier Connect, the company's public news and information website.

Elsevier hereby grants permission to make all its COVID-19-related research that is available on the COVID-19 resource centre - including this research content - immediately available in PubMed Central and other publicly funded repositories, such as the WHO COVID database with rights for unrestricted research re-use and analyses in any form or by any means with acknowledgement of the original source. These permissions are granted for free by Elsevier for as long as the COVID-19 resource centre remains active.
A deep learning-based social distance monitoring framework for COVID-19

Imran Ahmed, Misbah Ahmad, Joel J.P.C. Rodrigues, Gwanggil Jeon, Sadia Din

1. Introduction

COVID-19 originated from Wuhan, China, has affected many countries worldwide since December 2019. On March 11, 2020, the World Health Organization (WHO) announced it a pandemic as the virus spread through 114 countries, caused 4000 deaths and 118,000 active cases (WHO; W.H. Organization, 2020). On October 7, 2020, they reported more than 35,537,491 confirmed COVID-19 cases, including 1,042,798 deaths. The latest number of infected people due to pandemic is reported to-date. To stop the virus spread, the global community is looking for alternate ways. The virus mainly spreads in those people; who are in close contact with each other (within 6 feet) for a long period. The virus spreads when an infected person sneezes, coughs, or talks, the droplets from their nose or mouth disperse through the air and affect nearby peoples. The droplets also transfer into the lungs through the respiratory system, where it starts killing lung cells. Recent studies show that individuals with no symptoms but are infected with the virus also play a part in the virus spread (W. C. D. C. Dashboard). Therefore, it is necessary to maintain at least 6 feet distance from others, even if people do not have any symptoms.

Social distancing associates with the measures that overcome the...
virus’ spread, by minimizing the physical contacts of humans, such as the masses at public places (e.g., shopping malls, parks, schools, universities, airports, workplaces), evading crowd gatherings, and maintaining an adequate distance between people (Adlhoch, 2020; Ferguson et al., 2005). Social distancing is essential, particularly for those people who are at higher risk of serious illness from COVID-19. By decreasing the risk of virus transmission from an infected person to a healthy, the virus’ spread and disease severity can be significantly reduced (Statista) Fig. 2. If social distancing is implemented at the initial stages, it can perform a pivotal role in overcoming the virus spread and preventing the pandemic disease’s peak, as illustrated in Fig. 3 (Harvard). It can be observed that social distancing can decrease the number of infected patients and reduce the burden on healthcare organizations. It also lowers the mortality rates by assuring that the number of infected cases (patients) does not surpass the public healthcare capability (Nguyen et al., 2020).

In the past decades, computer vision, machine learning, and deep learning have shown promising results in several daily life problems. Recent improvement in deep learning allows object detection tasks (Brunetti, Buongiorno, Trotta, & Bevilacqua, 2018) more effective. Researchers (Punn, Sonbhadra, & Agarwal, 2020b; Ramadass, Arunachalam, & Sagayasree, 2020; Yang, Yurtsever, Renganathan, Redmill, & Ozgüner, 2020), often utilize these methods to measure social distancing among people across the moving frames, as seen in Fig. 4. To determine the distancing between people, clustering and distance-based methods are utilized. From Fig. 4, it can be seen that most of the methods are developed using frontal or side view video sequences, which requires a proper camera calibration to map pixels to distance for real easily, measurable units (i.e., feet, meters, etc.). Secondly, if we assume a top-down approach, i.e., an overhead view approach, then the distance calculations from the overhead view will lead to a better distance approximation and wide coverage of the wide scene.

In this work, we used an overhead view to provide an effective framework for social distance monitoring. Some scholars, e.g. Ahmed
Ahmed, Din, Jeon, and Piccialli (2019), Ahmed, Ahmad, Aljuaid (2017), Ahmad, Khan, Qayum, and Adnan (2020), Ahmad, Din, Jeon, and Piccialli (2019), Ahmed, Ahmad, Adnan, Ahmad, and Khan (2019), Ahmed, Ahmad, Piccialli, Sangaiiah, and Jeon (2018), Ahmad, Ahmed, Ullah, Khan, and Adnan (2018), Choi, Moon, and Yoo (2015), and Migniot and Ababsa (2016) use an overhead perspective for human detection and tracking. The overhead perspective offers a better field of view and overcomes the issues of occlusion, thereby playing a key role in social distance monitoring to compute the distance between people. It might help overcome computation, communication load, energy consumption, human resource, and installation costs (Ahmad et al., 2019). This work aims to present a deep learning-based social distance monitoring framework for the public campus environment from an overhead perspective. A deep learning model, i.e., YOLOv3 (You Only Look Once) (Redmon & Farhadi, 2018), is applied for human detection. The current model (pre-trained on frontal or normal view data sets) is initially tested on the overhead data set. Transfer learning is also used to improve the efficiency of the detection model. The additional training is performed with overhead data set, and the newly trained layer is appended to the pre-trained model. In order to track the social distance between individuals, the Euclidean distance is used to approximate the distance between each pair of the centroid of the bounding box detected. In addition, a social distance violation threshold is specified using a pixel to distance estimation. Utilizing a centroid tracking algorithm to keep track of the person who violates the social distance threshold.

• To present a deep learning-based social distance monitoring framework using an overhead view perspective.
• To deploy pre-trained YOLOv3 for human detection and computing their bounding box centroid information. In addition, a transfer learning method is applied to enhance the performance of the model.
• In order to track the social distance between individuals, the Euclidean distance is used to approximate the distance between each pair of the centroid of the bounding box detected. In addition, a social distance violation threshold is specified using a pixel to distance estimation.
• To assess the performance of pre-trained YOLOv3 by evaluating it on an overhead data set. The output of the detection framework is assessed with and without the transfer learning. Furthermore, the...
model performance is also compared with other deep learning models.

The rest of the work discussed in the paper is structured as follows. The related work is presented in Section 2. A deep learning-based social distance monitoring framework has been presented in Section 3. The overhead view data set used for training and testing during experimentation is briefly discussed in Section 4. The detailed analysis of output results and performance evaluation of the model with and without transfer learning is also illustrated in this Section. The conclusion of the given work with potential future plans is provided in Section 5.

2. Literature review

After the rise of the COVID-19 pandemic since late December 2019, social distancing is deemed to be an utmost reliable practice to prevent the contagious virus transmission and opted as standard practice on January 23, 2020 (B. News, 2020). During one month, the number of cases rises exceptionally, with two thousand to four thousand new confirmed cases reported per day in the first week of February 2020. Later, there has been a sign of relief for the first time for five successive days up to March 23, 2020, with no new confirmed cases (N. H. C. of the Peoples Republic of China, 2020). This is because of the social distance practice initiated in China and, latterly, adopted by worldwide to control COVID-19. Ainslie et al. (2020) investigated the relationship between behavior of space between them. The authors used a mobile robot with an RGB-D sensor to detect individuals who do not obey a social distance restriction, i.e., 6 feet of space between them. The authors also compared results with faster-RCNN and SSD. Ramadass et al. (2020) developed an autonomous drone-based model for social distance monitoring. They trained the YOLOv3 model with the custom data set. The data set is composed of frontal and side view images of limited people. The work is also extended for the monitoring of facial masks. The drone camera and the YOLOv3 algorithm help identify the social distance and monitor people from the side or frontal view in public wearing masks. Pouw, Toschi, van Schadewijk, and Corbetta (2020) suggested an efficient graph-based monitoring framework for physical distancing and crowd management. Sathyamoorthy, Patel, Savle, Paul, and Manocha (2020) performed human detection in a crowded situation. The model is designed for individuals who do not obey a social distance restriction, i.e., 6 feet of space between them. The authors used a mobile robot with an RGB-D camera and a 2-D lidar to make collision-free navigation in mass gatherings.

Until now researchers have done considerable work for detection (Iqbal, Ahmad, Bin, Khan, & Rodrigues, 2020; Patrick et al., 2020; Yash Chaudhary & Mehta, 2020), some provides an smart healthcare system for pandemic using Internet of Medical Things (Chakraborty, 2021; Chakraborty et al., 2021). Prem et al. (2020) studied the social distancing impacts on the spread of the COVID-19 outbreak. The studies concluded that the early and immediate practice of social distancing could gradually reduce the peak of the virus attack. As we all know, that although social distancing is crucial for flattening the infection curve, it is an economically unpleasant step. In Adolph, Amano, Bang-Jensen, Fullman, and Wilkerson (2020), Adolph et al. highlighted the United States of America’s condition during the pandemic. Due to a lack of general support by decision-makers, it was not implemented at an initial stage, starting harm to public health. However, social distancing influenced economic productivity; even then, numerous scholars sought alternatives that overcame the loss.

Researchers provide effective solutions for social distance measuring using surveillance videos along with computer vision, machine learning, and deep learning-based approaches. Punn et al. (2020b) proposed a framework using the YOLOv3 model to detect humans and the Deepsort approach to track the detected people using bounding boxes and assigned IDs information. They used an open image data set (OID) repository, a frontal view data set. The authors also compared results with faster-RCNN and SSD. Ramadass et al. (2020) developed an autonomous drone-based model for social distance monitoring. They trained the YOLOv3 model with the custom data set. The data set is composed of frontal and side view images of limited people. The work is also extended for the monitoring of facial masks. The drone camera and the YOLOv3 algorithm help identify the social distance and monitor people from the side or frontal view in public wearing masks. Pouw, Toschi, van Schadewijk, and Corbetta (2020) suggested an efficient graph-based monitoring framework for physical distancing and crowd management.
Fig. 5. Flow diagram of overhead view social distance monitoring framework.

Fig. 6. General architecture of YOLOv3 utilized for overhead view human detection.
From the literature, we concluded that the researcher had done a considerable amount of work for monitoring of social distance in public environments. But, most of the work is focused on the frontal or side view camera perspective. Therefore, in this work, we presented an overhead view social distance monitoring framework that offers a better field of view and overcomes the issues of occlusion, thereby playing a key role in social distance monitoring to compute the distance between peoples.

3. Social distance monitoring

Researchers use a frontal or side perspective for social distance monitoring, as discussed in Section 2.2. In this work, a deep learning-based social distance monitoring framework using an overhead perspective has been introduced. The flow diagram of the framework is shown in Fig. 5. The recorded overhead data set are split into training and testing sets. A deep learning-based detection paradigm is used to detect individuals in sequences. There are a variety of object detection models available, such as Krizhevsky, Sutskever, and Hinton (2012), Simonyan and Zisserman (2014), Girshick, Donahue, Darrell, and Malik (2014), Szegedy et al. (2015), Girshick (2015) and Ren, He, Girshick, and Sun (2015). Due to the best performance results for generic object detection, in this work, YOLOv3 (Redmon & Farhadi, 2018) is used. The model used single-stage network architecture to estimate the bounding boxes and class probabilities. The model was originally trained on the COCO (Common objects in context) data set (Lin et al., 2014). For overhead view person detection, transfer learning is implemented to enhance the detection model’s efficiency, and a new layer of overhead training is added with the existing architecture.

After detection, the bounding box information, mainly centroid information, is used to compute each bounding box centroid distance. We used Euclidean distance and calculated the distance between each detected bounding box of peoples. Following computing centroid distance, a predefined threshold is used to check whether the distance among any two bounding box centroids is less than the configured number of pixels or not. If two people are close to each other and the distance value violates the minimum social distance threshold. The bounding box information is stored in a violation set, as seen in Fig. 1, and the color of the bounding box is updated/changed to red. A centroid tracking algorithm is adopted for tracking so that it helps in tracking of those people who violate/breach the social distancing threshold. At the output, the model displays the information about the total number of social distancing violations along with detected people bounding boxes and centroids.

In this work, YOLOv3 is used for human detection as it improves predictive accuracy, particularly for small-scale objects. The main advantage is that it has adjusted network structure for multi-scale object detection. Furthermore, for object classification, it uses various independent logistic rather than softmax. The model’s overall architecture is presented in Fig. 6; it can be seen that feature learning is performed using the convolutional layers, also called Residual Blocks. The blocks are made up of many convolutional layers and skip connections. The model’s unique characteristic is that it performs detection at three separate scales, as depicted in Fig. 6. The convolutional layers with a given stride are practiced to downsample the feature map and transfer invariant-sized features (Redmon & Farhadi, 2018). Three feature maps, as shown in Fig. 6, are utilized for object detection.

The architecture shown in Fig. 6 is trained using an overhead data set. For that purpose, a transfer learning approach is adopted, that enhance the efficiency of the model. With transfer learning, the model is additionally trained without dropping the valuable information of the existing model. Further, the additional overhead data set trained layer is appended with the existing architecture. In this way, the model takes advantage of the pre-trained and newly trained information, and both detection results are further deliver better and faster detection results.

The architecture shown in Fig. 6 used a single-stage network for the entire input image to predict the bounding box and class probability of detected objects. For feature extraction, the architecture utilizes convolution layers, and for class prediction, fully connected layers are used. During human identification, as seen in Fig. 6, the input frame is divided into a region of $S \times S$, also called grid cells. These cells are related to bounding box estimation and class probabilities. It predicts the probability of whether the center of the person bounding box is in the grid cell or not:

$$\text{Conf}(p) = \text{Pr}(p) \times \text{IOU}(\text{pred. actual})$$

In Eq. (1), Pr(p) indicates that whether the person present is in the detected bounding box or not. The value of Pr(p) is 1 for yes and 0 for
not. IoU(pred, actual) determines the Intersection Over Union of the actual and predicted bounding box. It is defined as (Redmon & Farhadi, 2018):

$$\text{IoU} = \frac{\text{area}(\text{BoxT} \cap \text{BoxP})}{\text{area}(\text{BoxT} \cup \text{BoxP})}$$  \hspace{1cm} (2)

where the ground truth box (actual) manually labeled in the training data set represented with BoxT, and the predicted bounding box is displayed as BoxP. area presents the area of intersection. An acceptable area is predicted and decided for each detected person in the input frame. The confidence value is applied after prediction to achieve the optimal prediction. For each predicted bounding box, \(w, h, x, y\) are estimated, where bounding box coordinates are defined by \(x, y\), and width and height are determined by \(w, h\). The model produces the following predicted bounding box values as seen in Fig. 7 and Eq. (3) (Redmon & Farhadi, 2018):

$$b_x = \sigma(t_x) + c_x$$  
$$b_y = \sigma(t_y) + c_y$$  
$$b_w = p_x t_{w}$$  
$$b_h = p_h t_{h}$$  \hspace{1cm} (3)

In Eq. (3), \(b_x, b_y, b_w, b_h\) are predicted coordinate bounding boxes, where the coordinates' center is represented as \(x, y\) and width and height with \(w, h\). \(t_w, t_h, t_x, t_y\) defined the network output and \(c_x, c_y\) are used to correspond the top-left coordinates of the grid cell as shown in Fig. 7, while the \(p_w\) and \(p_h\) are width and height of anchors.

A threshold value is defined that process the high confidence values and discards the low confidence values. Using non-maximal suppression, the final location parameters are derived for the detected bounding box. At last, loss function is calculated, for detected bounding box (Redmon & Farhadi, 2018). The given loss function is the sum of three functions, i.e., regression, classification, and confidence. At each grid cell, if the object is detected, then the classification loss is computed as the squared error of the conditional class probabilities and calculated as (Redmon & Farhadi, 2018):

$$\mathcal{L}_{cls} = \sum_{i=0}^{S^2} \sum_{j=0}^{S^2} \sum_{c} 1^{\text{obj}}_{i,j} (p_i(c) - p_{i,j}(c))^2$$  \hspace{1cm} (4)

In Eq. (4), in grid cell \(i\) if the person is detected then \(1^{\text{obj}}_{i,j} = 1\), otherwise equals to 0. The conditional class probabilities for class \(c\) in grid cell \(i\) are represented as \(p_{i,j}(c)\). The localization loss estimates the failures in the predicted bounding box sizes and locations. The bounding box containing the detected object, i.e., a person, is added. It is defined as (Redmon & Farhadi, 2018):

$$\mathcal{L}_{loc} = \lambda_{\text{coord}} \sum_{i=0}^{S^2} \sum_{j=0}^{S^2} \sum_{c} (x_i - x_{i,j})^2 + (y_i - y_{i,j})^2$$  
$$+ (\sqrt{w_i} - \sqrt{w_{i,j}})^2 + (\sqrt{h_i} - \sqrt{h_{i,j}})^2$$  \hspace{1cm} (5)

In above equation \(1^{\text{obj}}_{i,j}\) is equal to 1, in case if the \(j\)th bounding box in grid cell \(i\) is used for object detection, otherwise it is equal to 0. Instead of predicting simple height and width, the model predicts the square root of the bounding box width and height. In Eq. (5) the scale parameters \(\lambda_{\text{coord}}\) is used for predictions of bounding box coordinates and equals to 5 as (Redmon, Divvala, Girshick, & Farhadi, 2016). The predicted positions are represented with \(x, y, h, w\) in \(i\)th cell of detected bounding box, while the actual positions of bounding box in the \(i\)th cell is defined using \(x_i, y_i, h_i, w_i\). The Eq. (5) measures the loss function of predicted bounding box having coordinates value \(x, y\). To represent the possibility
of the detected person in the $j$th bounding box $1_{ij}^{(b)}$ is used. The value of $\lambda$ is constant, the function in Eq. (5) calculates sum over each bounding box, using ($j = 0$ to $B$) as predictor for each grid cell ($i = 0$ to $S^2$).

Finally the confidence loss is calculated that is given in Eq. (6) as (Redmon & Farhadi, 2018):

$$L_{\text{conf}} = \sum_{i=0}^{S^2} \sum_{j=0}^{B} 1_{ij}^{(b)} (C_i - C_i^*)^2$$

(6)

$$L_{\text{conf}} = \lambda_{\text{noobj}} \sum_{i=0}^{S^2} \sum_{j=0}^{B} 1_{ij}^{(b)} (C_i - C_i^*)^2$$

(7)

where the confidence score is defined as $C_i^*$, for $j$th bounding box in grid cell $i$ and $1_{ij}^{(b)}$ and is equal to 1 in case if in cell $i$ the $j$th bounding box is responsible for object detection; otherwise it is equal to 0. In case if the object is not detected, then the confidence loss is provided as (Redmon & Farhadi, 2018).

Fig. 9. Social distance monitoring from an overhead view using a pre-trained detection model. In sample frames, the people in green rectangles are those who maintain the social distancing. The people who violate the social distance threshold are shown red in rectangles. The manually labels yellow positive cross shows miss detections. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

Fig. 10. Training loss of YOLOv3 using overhead view data set.
In Eq. (7), $1_{\text{noobj}}$ is defined as the complement of $1_{\text{obj}}$. The bounding box’ confidence score $C^\ast$ in cell $i$ and $\lambda_{\text{noobj}}$ is used to weights down the loss during detecting background. As in most cases detected, bounding boxes do not contain any objects that cause a class imbalance problem; therefore, the model is more frequently trained to detect background rather than detect objects. To solve this, the loss is weight down by a factor $\lambda_{\text{obj}}$ (default: 0.5).

After detecting people in video frames, in the next step, the centroid of each detected person bounding boxes shown as green boxes are used for distance calculation, as shown in Fig. 8(b). The detected bounding box coordinates $(x, y)$ are used to compute the bounding box’s centroid. Fig. 8(c) demonstrates accepting a set of bounding box coordinates and computing the centroid. After computing, centroid, a unique ID is assigned to each detected bounding box. In the next step, we measure

![Accuracy vs Epoch](chart.png)

**Fig. 11.** Training Accuracy of YOLOv3 using overhead view data set.

![Sample Frames](images.png)

**Fig. 12.** Results of social distance monitoring, using transfer learning. It can be seen that the detection performance of the model is improved after transfer learning. In sample frames, the people in green rectangles maintain social distancing while in red rectangles are those who breach/violate the social distance. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)
the distance between each detected centroid using Euclidean distance.

For every subsequent frame in the video stream, we firstly compute bounding box centroids shown in Fig. 8(c); and then calculate the distance (highlighted with red lines) between each pair of detected bounding box centroids, Fig. 8(d). The information of each centroid is stored in the form of a list. Based on distance values, a threshold is defined to check if any two people are less than $N$ pixels apart or not. If the distance violates the minimum social distance set or two people are too close, then the information is added into the violation set. The bounding box color is initialized as green. The information is checked in the violation set; if the current index exists violation set, the color is updated to red. Furthermore, the centroid tracking algorithm is used to track the detected people in the video sequence. The tracking algorithm also helps to keep track of people who are violating the social distance threshold. At the output, the model displays information about the total number of social distancing violations.

4. Experiments, results, and discussion

The detailed descriptions of various experiments carried out in this work are presented in this section. For social distance monitoring, an indoor data set recorded at Institute of Management Sciences, Hayatabad, Peshawar Pakistan is used (Ahmed, Ahmad, Adnan, et al., 2019; 2019a), containing video sequences captured from the overhead view. The data collection is divided into 70% and 30% training and testing, respectively. There is no restriction on the mobility of persons throughout the scene. Peoples in the scene move freely; their visual appearance is affected by radial distance and camera position. From example frames, it can be observed that the human’s visual appearance is not identical, and peoples heights, poses, scales are varying in the data set. For implementation, we used OpenCV. The experimental results are divided into two subsections; first, the pre-trained model’s testing results are discussed, while in the second subsection, the results of the detection model after applying transfer learning and training on the overhead data set are explained. For comparison, the model is tested using the same video sequences. The performance evaluation of the model is also made in this section, along with a comparison with different deep learning models.

4.1. Results of social distance monitoring using pre-trained model

In Fig. 9, the testing results of the social distance framework using a pre-trained model (Redmon & Farhadi, 2018) has been visualized. The testing results are evaluated using different video sequences. The people in the video sequences are freely moving in the scenes; it can be seen from sample frames that the individual’s visual appearance is not identical to the frontal or side view (Fig. 9). The person’s size is varying at different locations, as shown in Fig. 9. Since the model only considers human (person) class; therefore, only an object having an appearance like a human is detected by a pre-trained model. The pre-trained model delivers good results and detects various size person bounding boxes, as shown with green rectangles in Fig. 9(a)–(c). From sample frames of Fig. 9, people are marked with green rectangles as they maintain a social distancing threshold. The model is also tested for multiple peoples, as depicted in Fig. 9(g)–(i), multiple people are entering in the scene. In sample images, it can be seen that after person detection, the distance between each detected bounding box is measured to check whether the person in the scene violates the social distance or not. In Fig. 9(e) and (h), two people at the center of the scene are marked with red bounding boxes as they violate or breaches the social distancing threshold. Some miss detections also occur that are manually labeled with a yellow cross in sample frames. From the sample frames, it can be seen that a person is effectively detected at several scene locations. However, in some cases, the person’s appearance is changing; therefore, the model gives miss detections. The reason for miss detection maybe, as the pre-trained model is applied, and an individual’s appearance from an overhead view is changing, which may be misleading for the model.

4.2. Results of social distance monitoring using transfer learning

The transfer learning methodology is applied to improve the accuracy of the detection model. Using an overhead data set, the model is additionally trained using 500 sample frames. The epoch size 40 and batch size 64 is set for training of the model. The training loss and accuracy curves are shown in Figs. 10 and 11. A new layer is obtained after training the model; that is further appended with a pre-trained model.

The model is now tested for the same test video sequences, as discussed in the above sub-section. The experimental findings reveal that transfer learning significantly increases the detection results, as seen in Fig. 12. From the sample images, it can be visualized that the model detects the individuals at various scene locations. People with various characteristics are effectively-identified, and the social distance between people is also computed, as shown in the sample frames. In sample frames of Fig. 12(a)–(c), there is no social distance violation found, since

| S. no. | Model                        | True detection rate | False detection rate |
|--------|------------------------------|---------------------|----------------------|
| 1.     | Fast-RCNN (pre-trained)      | 90%                 | 0.7%                 |
| 2.     | Faster-RCNN (pre-trained)    | 92%                 | 0.6%                 |
| 3.     | Mask-RCNN (pre-Trained)      | 93%                 | 0.5%                 |
| 4.     | YOLOv3 (pre-trained)         | 92%                 | 0.4%                 |
| 5.     | YOLOv3 (trained overhead data set) | 95%               | 0.3%                 |
all people are marked with green rectangle boxes by the automated framework. While in the sample frame Fig. 12(e), the violation is detected; however, the number of people present in the scene is small as compared to Fig. 12(b), where all people are maintaining social distance, and therefore not a single violation is observed. In Fig. 12(d)–(f), due to close interactions between people, violation is recorded by automated system. The same behavior can be found in Fig. 12(g)–(i) where people are around dozen in both (g) and (h) and violation in (h) is three times as compared to (g). In Fig. 12(d)–(f), multiple people are walking, and entering in the scene are detected and monitored. The framework effectively detected the breach of social distance between people and marked the bounding box as red rectangles if people are too close to each other.

4.3. Performance evaluation

Different quantitative metrics are used in this work to evaluate the performance of the framework for social distance monitoring using a deep learning model and an overhead perspective. To assess the efficiency of the detection model, Precision, Recall, and Accuracy is used. Furthermore, the findings are also compared with other deep learning models. For estimation of Precision, Recall and Accuracy, we used, tp true positive, fp false positives, tn true negative and fn false-negative. The Accuracy Recall and Precision results are shown in Fig. 13. It can be analyzed that when the model is additionally trained for overhead view data set, the overall performance of the detection model is improved. The tracking accuracy is also given in Fig. 14.

We also compared the newly trained YOLOv3 with other deep learning models. The True detection and False detection rate of different deep learning models are depicted in Table 1. From the results, it can be seen that transfer learning improved the results significantly for the overhead view data set. The false detection rate of different deep learning models are very small, about 0.7–0.4% without any training, which reveals the effectiveness of deep learning models. Different pre-trained object detection models are tested on the overhead data set. Although the models were trained on the different frontal data sets, they still show good results by achieving an accuracy of 90%. In Fig. 15, the comparison results of different state of the art detection are shown.

5. Conclusion and future works

In this work, a deep learning-based social distance monitoring framework is presented using an overhead perspective. The pre-trained YOLOv3 paradigm is used for human detection. As a person’s appearance, visibility, scale, size, shape, and pose vary significantly from an overhead view, the transfer learning method is adopted to improve the pre-trained model’s performance. The model is trained on an overhead data set, and the newly trained layer is appended with the existing model. To the best of our knowledge, this work is the first attempt that utilized transfer learning for a deep learning-based detection paradigm, used for overhead perspective social distance monitoring. The detection model gives bounding box information, containing centroid coordinates information. Using the Euclidean distance, the pairwise centroid distances between detected bounding boxes are measured. To check social distance violations between people, an approximation of physical distance to the pixel is used, and a threshold is defined. A violation threshold is used to check if the distance value violates the minimum social distance set or not. Furthermore, a centroid tracking algorithm is used for tracking peoples in the scene. Experimental results indicated that the framework efficiently identifies people walking too close and violates social distancing; also, the transfer learning methodology increases the detection model’s overall efficiency and accuracy. For a pre-trained model without transfer learning, the model achieves detection accuracy of 92% and 95% with transfer learning. The tracking accuracy of the model is 95%. The work may be improved in the future for different indoor and outdoor environments. Different detection and tracking algorithms might be used to help track the person or people who are violating or breaches the social distancing threshold.

Conflict of interest

None declared.

Declaration of Competing Interest

The authors report no declarations of interest.

Acknowledgments

This work is partially supported by FCT/MCTES through national funds and when applicable co-funded EU funds under the project UIDB/50008/2020; and by Brazilian National Council for Scientific and Technological Development (CNPq) via Grant No. 309335/2017-5.

References

Adolph, C., Amano, K., Bang-Jensen, B., Fullman, N., & Wilkerson, J. (2020). medRxiv. Adolph, C., Amano, K., Bang-Jensen, B., Fullman, N., & Wilkerson, J. (2020). medRxiv.
