



HAL
open science

Counting Number of People and Social Distance Detection Using Deep Learning

P. Kalaivaani, M. Abitha Lakshmi, V. Bhuvana Preetha, R. Darshana, M. K.
Dharani

► **To cite this version:**

P. Kalaivaani, M. Abitha Lakshmi, V. Bhuvana Preetha, R. Darshana, M. K. Dharani. Counting Number of People and Social Distance Detection Using Deep Learning. 5th International Conference on Computational Intelligence in Data Science (ICCIDS), Mar 2022, Virtual, India. pp.30-43, 10.1007/978-3-031-16364-7_3 . hal-04381295

HAL Id: hal-04381295

<https://inria.hal.science/hal-04381295v1>

Submitted on 9 Jan 2024

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



Distributed under a Creative Commons Attribution 4.0 International License



This document is the original author manuscript of a paper submitted to an IFIP conference proceedings or other IFIP publication by Springer Nature. As such, there may be some differences in the official published version of the paper. Such differences, if any, are usually due to reformatting during preparation for publication or minor corrections made by the author(s) during final proofreading of the publication manuscript.

COUNTING NUMBER OF PEOPLE AND SOCIAL DISTANCE DETECTION USING DEEP LEARNING

Dr PCD.Kalaivaani¹
Assistant Professor(SLG)
Department of CSE,
Kongu Engineering College
kalairupa@kongu.ac.in

M. Abitha Lakshmi² V.Bhuvana Preetha² R.Darshana²
Final CSE
Department of CSE,
Kongu Engineering College
abithamatheswaran2012@gmail.com
bhuvanapreetha.v@gmail.com
darshana26799@gmail.com

M.K.Dharani³
Assistant Professor
Department of CSE
Kongu Engineering College
dharani.cse@kongu.ac.in

Abstract . With its enormous spread, the continuing COVID-19 corona virus pandemic has become a worldwide tragedy. Population vulnerability rises as a result of fewer antiviral medicines and a scarcity of virus fighters. By minimizing physical contact between people, the danger of viral propagation can be reduced. Previously, the distance between two individuals, as well as the number of people breaching the distance, could be computed with alarm. In the proposed methodology, including the existing features in the previous methodology and additionally, the total number of people present in a given frame, as well as the number of people who were violated, non-violated are tallied. The most crucial step in improving social separation detection is to use proper camera calibration. It will produce better results and allow you to compute actual measurable units instead of pixels. It can also figure out how many people are in a certain location. As a result, the suggested system will be useful in identifying, counting, and alerting persons who are breaching the specified distance, as well as estimating the number of people in the frame to manage corona spread.

Keywords : COVID-19, Social distancing detection, Object detection: you only look once (YOLO), COCO, CCTV

1. Introduction

Since December 2019, the Corona Virus, which originated in China, has infected several countries around the world. Many healthcare providers and medical professionals are researching for proper vaccines to overcome the corona virus. Despite the fact that no development has been reported, it is not very efficient. The community is exploring for new ways to stop the infection from spreading. Liquid from such an infectious person's nostrils or throat leaks when they sneeze, cough, or talk. It spreads quickly and has an impact on others.

Social distancing refers to strategies for preventing the transmission of a virus by limiting people's physical contact, such as avoiding crowds in public areas and keeping a safe distance between them. If done early on, social distance can help stop the virus from spreading and the pandemic illness from reaching its peak. For the last few years, computer vision, machine learning, and deep learning has shown promise in a variety of real-world scenarios. We employed an overhead perspective to create a framework for social distance monitoring in this study.

The above viewpoint provides a larger field of vision and eliminates occlusion difficulties when computing the distance between people, making it a valuable tool for social distance monitoring. Object detection has become common place in recent years, thanks to the usage of video surveillance, object tracking, and pedestrian identification. As a result, the goal of the project is to create a deep learning domain that can identify social distance from above. YOLOv3 object recognition is used by the system to recognise people in a video sequence. Using the detected bounding box information, the pre-trained detection algorithm locates persons. The bounding box is established, and the Euclidean distance is used to calculate the centroid's pairwise distances of persons. A tracking algorithm is also utilized to identify persons and quantify the amount of people in video sequences, allowing for the tracking of those who cross the social distance threshold or beyond it.

2. Existing System

In the existing system, the people are detected using YOLOv3 object detection algorithm with coco dataset. The distance between the people is calculated using Euclidean distance, who are all in the frame and also the minimum distance is given to identify whether the computed distance is exceeding given distance. If the distance is lesser than the given value, then person is added to the violated set and the bounding box colour for that person will get changed to red. If the person is maintaining the exact distance (i.e. the distance is exact as given value or greater than the given value) then the person's bounding box will be green to indicate that they are maintaining the social distance.

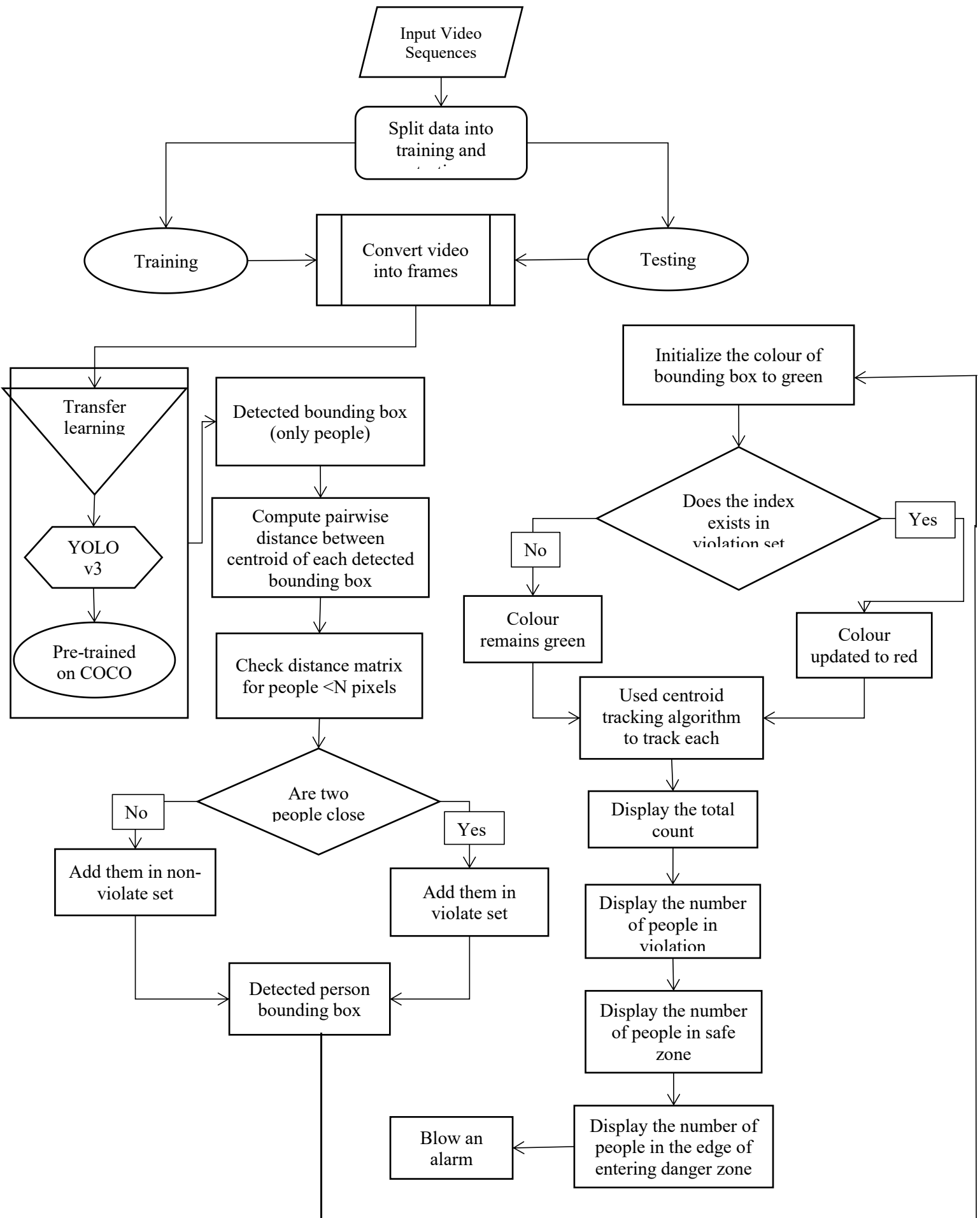


Fig.1. Architecture of proposed system

3. Proposed System

As a result of the pandemic crisis that has taken over the world and made certain things worse, vaccines designed for contagious diseases have not shown to be effective, and thus social distance has arisen as one of the most efficient ways to stop the corona virus from spreading. Physical separation from one another is referred to as social distancing. Around the world, cases are increasing at an alarming rate, prompting societal seclusion. We can compute and summaries the distances between individuals in the monitor who breaches the social distancing, utilizing CCTV cameras, and so keep track of human activities in public spaces. People who congregate in large numbers at religious sites might exacerbate the situation. Recently, all nations across the globe have been on lock-down, forcing residents to stay at home. However, as time passes, people will begin to attend more public and religious sites, thus in those conditions, the proposed system of monitoring social distance would be helpful all over the world. It is possible to track human beings and compute the distance between them in pixels, as well as set the standard-maintained distance to be followed, and get an overview of people who are all breaking the law and concerned authorities who can take action, using computer vision and deep learning, as well as the installed CCTV.

The proposed method uses the YOLO, COCO dataset to detect just persons in order to determine whether or not they are keeping a healthy distance. This keeps track of those who continue to break the social distance rules established by public health officials. An alert will sound if persons break the social distance inside the observation area. It also counts the number of persons who enter a restricted area and keeps track of the number. For each position, a distinct colour is allocated. A green colour box emerges when a person is in the safe zone. When he/she is about to enter a dangerous zone the colour of the box turns to orange to note that they are about to enter the danger zone. The box changes into red colour if the person enters into the danger zone.

4. YOLO Detector

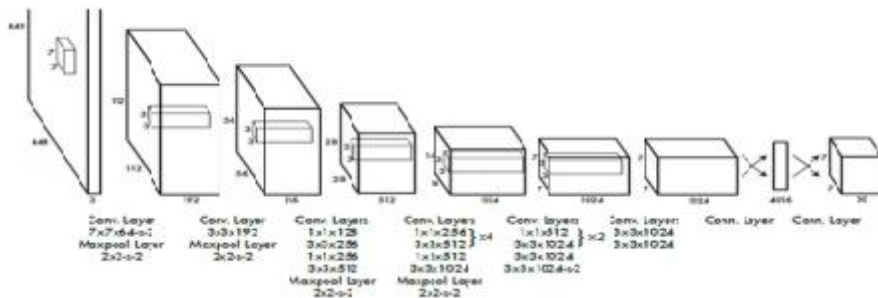


Fig.2. Architecture of YOLO

You Only Look Once (YOLO), a single convolutional neural network that differs from earlier detectors, frames objects as a regression issue to spatially distributed bounding boxes and related class probabilities directly from full images in a single evaluation. The YOLOv1 structure, which has two completely linked layers and 24 convolutional layers.

YOLOv3 features three distinct scales, each of which predicts three anchor boxes. For the four bounding box offsets, one objectless, and 80 class predictions in the COCO dataset, the tensors at each scale are $N \times N \times [3(4 + 1 + 80)]$. To determine the priors, YOLOv3 uses the k-means cluster (anchors). The width and height of the pre-selected 9 clusters are (1013), (1630), (3323), (3061), (6245), (59119). In order to detect objects, each group is given to a distinct feature map. Furthermore, YOLOv3 does feature extraction using a new network named Darknet-53. It still contains some shortcut connections, but it's a lot bigger today, with 53 convolutional layers compared to the YOLO 9000's 19 levels and the YOLO v1's 24 layers. This new network extracts features more precisely than Darknet-19, which was used in YOLOv2, or previous networks like ResNet-101 or ResNet-152.

5. COCO Dataset

COCO is a large-scale image dataset that may be used for segmentation, object detection, human key-point detection, stuff, and captioning. Its collection contains 1.5 million item instances and approximately 200K annotations, 330K images. All of the products are categorised into 80 groups. It contains Python, Matlab, and Lua APIs for loading, analysing, and displaying all picture annotations.

6. Centroid Tracking Algorithm

The bounding box is sent with (x, y) coordinates for every seen object in each and every frame in the centroid tracking technique. These bounding boxes can be generated by any object detector, such as Haar cascades, colour threshold + contour extraction, and so on. It is definitely necessary to compute bounding boxes for each frame in the video series. The centroid is computed and each bounding box is given a unique ID once the (x, y) coordinates of bounding boxes are allotted. For each continuous frame, the centroid of the object is calculated using the bounding box principle. Using a unique ID for each object detection, on the other hand,

can negate the purpose of object tracking. To fix this, we'll try to link the new object's centroid to that of an existing object. To do so, we'll use the formula,

$$D = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2} \quad (1)$$

to calculate the Euclidean distance between the two observed objects. We suppose that the observer will travel between the frames at first, but that the frame centroids will be the shortest, i.e. smaller than all other object distances. As a result, by comparing the centroids to the shortest distances between succeeding photos, the object tracker may be easily constructed. We add a large number of items to our list in order to detect them. We add a big number of objects to our identification list. It essentially ensures If more objects are recognized and monitored than previously detected and monitored objects, the newly discovered objects must be registered. Creating bounding boxes for newly discovered objects can be used to register the new things that have been detected. Following the identification of people, we must measure the centroid of the bounding box and store it in the list, as well as keep track of the object's motions by using Euclidean's method to calculate the minimum distance.

To improve the effectiveness of a tracking system, it must be able to cope with the case where an object moves out of the frame or out of the field of view, which we do by de-registering the object. De-registering an object entails removing the object's unique ID and other details once it has passed out of sight or scope, which is determined by comparing the object to other objects in the frame. We assume an item has vanished or migrated out of the field when it does not fit another current object after a given number of frames, and the old object is de-registered.

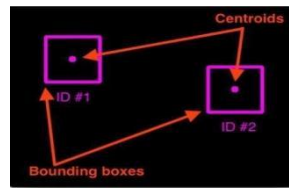


Fig.3. Centroid Tracking

7. Module Description

There are four modules in this module that involve and are completed by giving the pre-trained YOLO model. The modules are as follows:

- Module for detection
- Module for social distance
- Module for counting people
- Restriction module

7.1 Detection Module

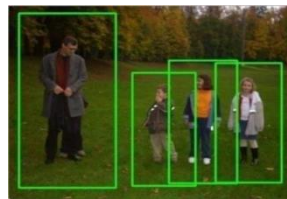


Fig.4. Detection of people

In the detection module, the frame or input video contains a pre-trained and pre-initialized yolo object identification module, as well as CNN output names and individual indexes to detect only the person in the video frame. The input frame's height and breadth are calculated using the frame measurements. Constructing the blob and then doing a frontal pass of YOLO object recognition using Open- CV. The class ID and confidence of the current object identification are retrieved after ensuring that the detected object is an individual and that the confidence rate (minimum) is met, and the filtering of identified objects begins. YOLO only gives the center (x, y) coordinates of the rectangular box after the width and height of the boxes have been scaled to the image's dimension. The top and left corners of the bounding box are obtained using the center of (x,y)- coordinates, and the values of bounding box coordinates, centroids, and confidences of a detected target are then adjusted. The centroid of a discovered individual is the centre of the bounding box, which is calculated by eliminating the centre of the bounding box.

To limit the amount of overlapping bounding boxes, Non-Maxima Suppression is utilized. Finally, check that the frame has at least one detection and that the box coordinates in the result list have been retrieved and changed. The Social distance module received the trust of each person identified, as well as the bounding box and centroid of each person observed, for further processing and to determine whether or not the individuals are maintaining a safe distance. To limit the amount of overlapping bounding boxes, Non-Maxima Suppression is utilized. Finally, double-check that the frame has at least one detection and that the box coordinates in the result list have been retrieved and changed. The Social distance module received the trust of each person

identified, as well as the bounding box and centroid of each person observed, for further processing and to determine whether or not the individuals are maintaining a safe distance.

7.2 Social Distance Module

The detection module is imported into the social distance module using the import command. For the whole input image, a single-stage network is built and employed in the structure to forecast the bounding box and class probability of identified items. Convolution layers are employed for feature extraction, while fully linked layers are used for class predictions in this design. The incoming frame is segmented into a region of $S \times S$, usually referred to as grid cells, during human identification. The bounding box prediction and class probabilities are linked to the grid cells, as well as predicting whether the person's bounding box's center is in the grid cell or not:

$$Conf(p) = Pr(p) \times IOU(pred, actual) \quad (2)$$

$Pr(p)$ in Eq. (2) denotes whether or not the person is within the bounding box that has been detected. $Pr(p)$ has a value of 1 for true and 0 for false. IOU stands for Intersection Over Union of the actual and anticipated bounding boxes ($pred, actual$). It is defined as follows:

$$IoU(pred, actual) = \frac{area_{BoxT \cap BoxP}}{area_{BoxT \cup BoxP}} \quad (3)$$

The anticipated bounding box is shown as BoxP, while the intersection region is shown as BoxT. Each recognized individual in the input frame is assigned to an appropriate region, which is forecasted and decided. To get the best bounding box, the confidence value is used to specify the bounding box following prediction. h, w, x, y are estimated for each anticipated bounding box of a detected person, with w, h determining width and height and x, y defining bounding box coordinates. As seen in Eq. 2, the model generates the expected bounding box values (4)

$$\begin{aligned} b_x &= \sigma(t_x) + c_x \\ b_y &= \sigma(t_y) + c_y \\ b_w &= p_w e_w^t \\ b_h &= p_w h_h^t \end{aligned} \quad (4)$$

In Eq. (4), b_x, b_y, b_w, b_h are expected coordinate bounding boxes, with x, y indicating the centre of the coordinates and w, h representing the width and height of the coordinates. The network output is defined by t_w, t_h, t_x, t_y , whereas the grid cell's top-left coordinates are defined by c_x, c_y . Anchor height and weight are represented by p_h and p_w . Using a threshold value, the high confidence data are processed, while the low confidence values are rejected. Non-maximal suppression is used to acquire the finalized position values for such observed bounding box. For each bounding box that is detected, a loss function is calculated. Confidence, classification, and regression are the three functions that make up the provided loss function. If an object is discovered in each grid cell, the classification loss is determined as the squared error of the conditional class probabilities for each detected object.

$$\ell_{cls} = \sum_{j=0}^{s^2} l_{ij}^{obj} \sum_{c \in class} l_i^{obj} (p_i(c) - p^*(c))^2 \quad (5)$$

In Eq. (5), if the person is discovered in grid cell I $l_{obj} ij = 1$, else equals 0. p_i denotes the conditional class probabilities for each class c in grid cell i . (c) . The errors in the predicted bounding box sizes and positions of identified items are estimated using the localization loss. The bounding box is expanded to include the detected item, which is a human. It is defined as follows:

$$\ell_{loc} = \lambda_{coord} l_{ij}^{obj} \sum_{i=0}^{s^2} \sum_{i=0}^n [(x_i - x_i^*)^2 + (y_i - y_i^*)^2 + (\sqrt{w_i} - \sqrt{w_i^*})^2 + (\sqrt{h_i} - \sqrt{h_i^*})^2] \quad (6)$$

$l_{obj} ij$ is equal to 1 in the preceding equation if the j th bounding box in grid cell I is used for object detection, else it is equal to 0. The model predicts the square root of the bounding box height and breadth, rather than width and height. The scaling parameter $coord$ is used in Eq. (6) to forecast bounding box coordinates. In the i th cell of the detected bounding box, the expected positions are represented by x_i, y_i, h_i, w_i , meanwhile the observed positions of the bounding box in the i th cell are given by $x_i^*, y_i^*, h_i^*, w_i^*$. The loss function of the anticipated bounding box with coordinates x, y is measured by Eq. (6). $l_{obj} ij$ is used to denote the potential of the detected person in the j th bounding box. When $(j = 0 \text{ to } B)$ is used as a predictor for each grid cell $I = 0 \text{ to } S^2$, the function in Eq.(6) calculates sum over each bounding box. Finally, the confidence loss, which is provided in Eq., is determined (7)

$$\ell_{conf} = \sum_{i=0}^{s^2} \sum_{j=0}^B l_{ij}^{obj} (C_i - C_i^*)^2 \quad (7)$$

$l_{noobj} ij$ is defined as the complement of $l_{obj} ij$ in Eq. (8). In cell I and $noobj$, the bounding box's confidence score C_i^* is being used to scale down the loss during background detection. In the great majority of cases, bounding boxes don't really include any objects that pose a class imbalance problem, hence the model is trained to identify background rather than identify objects. For the j th bounding box in grid cell I and $l_{obj} ij$, the loss is reduced by a factor $noobj$ (default: 0.5), where the confidence score is defined as C_i^* , and is equal to 1 if the j th bounding box in cell I is responsible for object identification; otherwise, it is equal to 0.

$$\ell_{conf} = \lambda_{noobj} \sum_{i=0}^{s^2} \sum_{j=0}^B l_{ij}^{noobj} (C_i - C_i^*)^2 \quad (8)$$

The centroid of each identified person bounding box, displayed as green boxes, is being used for distance calculation once individuals are spotted in sequence video frames. Using the identified bounding box coordinates, the center of the bounding box is determined (x, y). The Euclidean distance is then used to compute the distance among each detected centroid. Compute bounding box centroids for each consecutive frame in the video stream, then calculate the distance between each pair of identified bounding box centroids. A list is built to retain the centroid information of each identified individual, as well as a threshold value to determine whether two people are fewer than the specified N pixels apart. The bounding box of the detected individual is set to green. If the identified person crosses the given social distance, the information about that person is added to the violation set and checked in the violation set. The bounding box's colour is changed to red if the person's current index is in the violation set. At addition, in each frame of the video stream, the centroid tracking approach is used to track the recognised humans. People that are detected as being too close to a predefined social distance threshold can likewise be tracked using this technology. The output of the model is the total count of people who break the social separation rule.



Fig.5. Detection of people with bounding box

7.3 People Counting Module

The people counting module consists of four sections and they are,

- Total count of people
- Number of people in the green zone
- Number of people in the orange zone
- Number of people in the red zone

The Total count of people Section counts the length of and the total number of people information will get stored and displayed in the output frame. Initially, all the objects will get detected using YOLOv5 object detection algorithm, then using COCO dataset only the people are detected from all the detected objects and bounding box will be computed only for the person object also the colour of bounding box assigned as green for all the detected person. In the next step, the person's information who are violating the mentioned default minimum distance will get added to the violation set and then the bounding box colour. All updated to red to indicate they are in the danger zone. When two or more people are on the verge of violating the specific distance but haven't yet violated the distance, are considered to be in the orange zone and the count is calculated also the bounding box updated to orange. The remaining people in the frame are maintained the social distance is counted and they are considered to be in in the green zone and also the colour of the bounding box is updated to green.

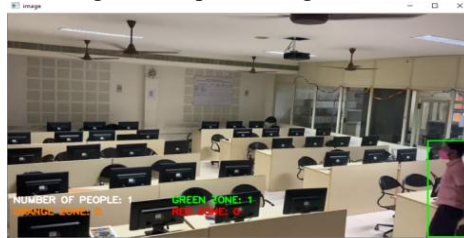


Fig.6. Count of people with zone

7.4 Restriction Module

In the restriction module, a hall is set up where a group of people enter with or without maintaining social distance. The number of people entering the hall is also been calculated using people counting algorithm. When one or two people enter the hall with appropriate distance they are considered to be in the green zone. i.e., two or more people with provided minimum distance between them, when two or more people are on the verge of violating the specific distance but haven't yet violated the distance considered to be in the orange zone. When two or more people completely violating the specified distance then they are considered to be in the red zone, they will also be alerted with an Alarm. The count which is calculated in the people counting module results in the total number of people in that frame and it also blows an alarm when the number of people exceeds than the allocated number of people in order to maintain the specific distance in the hall. When an alarm is blown, it doesn't alert just the person who is violating it but the entire team whose present in the particular hall which will be beneficiary to alert the environment. These are the basic functions that the restriction module performs which enhances the outlook on the proposed project, which will be helpful in the current pandemic situation to avoid the spread with the first and foremost procedure, which is to maintain social distance.



Fig.7. People altering with distance

8. Results and Conclusion

In social distance monitoring, indoor data sets with video sequences taken from an above perspective are used. Data collection is always divided into two parts: 70% training and 30% testing. People in a scenario can freely move around the area, with just radial distance and camera location influencing their visual appearance. The example frames show that a person's visual appearance varies depending on the dataset, and that people's heights, postures, and scales fluctuate. The implementation also makes use of Open-CV. The results of the pre-trained testing are discussed in the first paragraph, while the results of the specified detection model after transfer learning and training on the overhead data set are discussed in the second paragraph. In most cases, the model is confirmed by comparing identical video sequences. This section contains a model performance evaluation as well as a comparison to several deep learning models. This section includes a performance evaluation of the model as well as a comparison to different types of deep learning models. The Euclidean formula was used to calculate the distance between the persons in the frame, and the distance was calculated between the current object's centroid and the new object's centroid. The formula is provided in Eq 9 below.

$$D = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2} \quad (9)$$

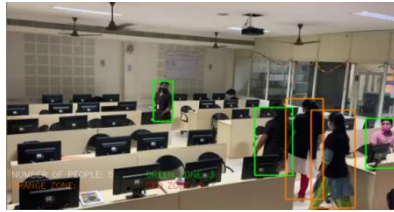


Fig.8. Sample output of the proposed framework

A variety of video sequences are used to analyse the test outcomes. The characters are free to wander around the scene in the video scenes. The size of the individual will change depending on where they are. Because the model only evaluates the human (person) class, a pre-trained model will only detect an object that has the appearance of a human. As illustrated by the green colour rectangles, the pre-trained model delivers results and also recognises varied sizes of person bounding boxes. People are designated as sustaining social distance thresholds along green rectangles in sample frames. The model is put to the test when numerous persons enter a scene. However, if the person's appearance changes, the model will produce false positives. This could be because when a pre-trained model is utilised, an individual's look from such an overhead view changes, that could be misleading for the present model. With over 500 sample frames, an overhead data set was employed to train the model. For training the given model, the epoch size 40 and each batch size 64 were chosen. After the model had been trained, a new layer was created and added to the pre-trained model. The results of the experiments show that transfer learning improves detection outcomes significantly. It may also be visualised accurately, with the model detecting people in various scenic locations.

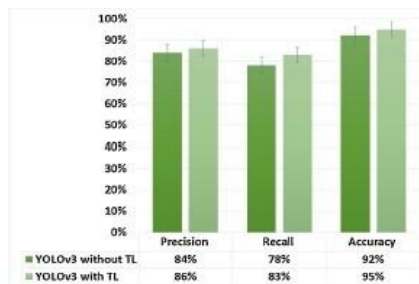


Fig.9. Model (YOLOv3) precision, recall, and accuracy with and without transfer learning.

The utility of a framework for socially distancing tracking that used a deep learning method and an overhead perspective was assessed using a number of quantitative indicators in this study. Precision, recall, and accuracy were used to evaluate the detection model's effectiveness. Transfer learning significantly enhanced the outcomes for the overhead view by data set, according to the findings. Several deep learning models have a low false detection rate, ranging from 0.7–0.4 percent without any training, indicating about deep learning's performance.



Fig.10. Accuracy of tracking with the YOLOv3 detection model, which has been pre-trained and trained.

9. Conclusion and Future work

The need for self-responsibility rises drastically when we examine the world following the COVID-19 outbreak. The scenario will mostly focus upon adopting and adhering to the WHO's set of precautions and restrictions, more specifically as the duty of someone who will entirely rely on themselves rather than the government. COVID-19 is disseminated through intimate contact with sick persons, therefore social distance is unquestionably the most important factor. To oversee wide regions, a distinct and successful method is required, and this survey article concentrates on that. Authorities will employ CCTV and drones to monitor human activity and regulate huge gatherings, averting law enforcement breaches. People will be signalled with a green light as long as they stay a safe distance. As the throng becomes larger, a red light will emerge on the CCTV, alerting the designated police to the location and allowing the issue to be addressed. Because supervising a large crowd is difficult, this survey can aid in the management of situations before they go out of control. As a consequence of this strategy, police on-the-ground operations will be reduced, allowing them to focus only on overseeing circumstances in locations where conditions are unfavourable, allowing them to utilise time effectively and conserve energy for equitable situations.

As a result, persons can be detected using the pre-trained YOLOv3 paradigm. The model has been trained on a separate data set and then attached to the prior model. Transfer learning has never been employed in a deep learning-based detecting approach for checking social distance from above, to our knowledge. The detection model provides bounding box details, including centroid coordinates.

The pairwise centroid distances between recognized bounding boxes are computed using the Euclidean distance. An estimation of physical separation to the pixel is performed to examine for social distance violations between persons, and a threshold is set. A violation threshold is used to evaluate if the distance result exceeds the set minimum social distance. In addition, to keep track of persons on the scene, a centroid tracking approach is used. The framework effectively detects people strolling too close together and violates social separation, according to experimental data; furthermore, the transfer learning methodology improves the detection model's entire efficiency and accuracy. Without transfer learning, a pre-trained model outperforms a detecting accuracy of 92 percent, but with transfer learning, the model generates a detecting accuracy of 95 percent. The tracking accuracy of this particular model is 95%.

In the future, the project could be improved for a variety of indoor and outdoor environments. The group of individuals who are breaking the social distance threshold will be tracked using a variety of detection and monitoring methods. We can build this model/system in real time with CCTV and required hardware pieces in future development.

References

- [1] J. Redmon, S. Divvala, R. Girshick, A. Farhadi, "You only look once: Unified, real-time object detection". Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2016.
- [2] J. Redmon and A. Farhadi, "YOLO9000: better, faster, stronger". Proceedings of the IEEE conference on computer vision and pattern recognition, arXiv:1612.08242 [cs.CV], 2017.
- [3] Dr. S. Syed Ameer Abbas, Dr. P. Oliver Jayaprakash, M. Anitha, X. Vinitha Jaini,
- [4] "Crowd Detection and Management using Cascade classifier on ARMv8 and OpenCV-Python", Mepco Schlenk Engineering College, Sivakasi, 2017 International Conference on Innovations in Information, Embedded and Communication systems (ICIIECS).
- [5] Joel Joseph Joy, Manali Bhat, Namrata Verma, Milind Jani, "Traffic Management Through Image Processing and Fuzzy Logic", D.J. Sanghvi College of Engineering.
- [6] Neel Bhavne, Aniket Dhagavkar, Kalpesh Dhande, Monis Bana, Jyoti Joshi, "Smart Signal-Adaptive Traffic Signal Control using Reinforcement Learning and Object Detection", Department of IT, RAIT, Nerul, Maharashtra.
- [7] Dr. S. Syed Ameer Abbas, M. Anitha, X. Vinitha Jaini, "Realization of Multiple Human Head Detection and Direction Movement Using Raspberry Pi", Electronics And Communication Engineering, Mepco Schlenk.
- [8] Article on Object detection with 10 lines of code by Moses Olafenwa on June 16, 2018 on <https://towardsdatascience.com>

- [9] World Health Organisation.WHO Corona- viruses Disease Dashboard (August 2020). Available at <https://covid19.who.int/table>.
- [10] Mahdi Rezaei, Mohsen Azarmi, "DeepSOCIAL: Social Distancing Monitoring and Infection Risk Assessment in COVID-19 Pandemic", "Institute for Transport Studies,The University of Leeds".
- [11] Rinkal Keniya,Ninad Mehendale,"Real- time social distancing detector using Net19 deep learning network".
- [12] Narinder Singh Punn, Sanjay Kumar Sonbhadra, Sonali Agarwal, "Monitoring COVID- 19 social distancing with person detection and tracking via fine-tuned YOLO v3 and Deep sort techniques",May06,2020.
- [13] J.Redmon and A. Farhari,"Yolov3:An Incremental improvement",2018.
- [14] N. Singh Punn, S.K. Sonbhadra, S. Agarwal, Monitoring covid-19 social distancing with person detection and tracking via fine-tuned yolo v3 and deepsort techniques (2020)
- [15] Figure 2 is taken from https://www.researchgate.net/figure/YOLO-architecture-YOLO-architecture-is-inspired-by-GooLeNet-model-for-image_fig2_329038564
- [16] A.J. Sathyamoorthy, U. Patel, Y.A. Savle, M. Paul, D. Manocha, COVID-Robot: Monitoring Social Distancing Constraints in Crowded Scenarios, <https://arxiv.org/abs/2008.06585v2>