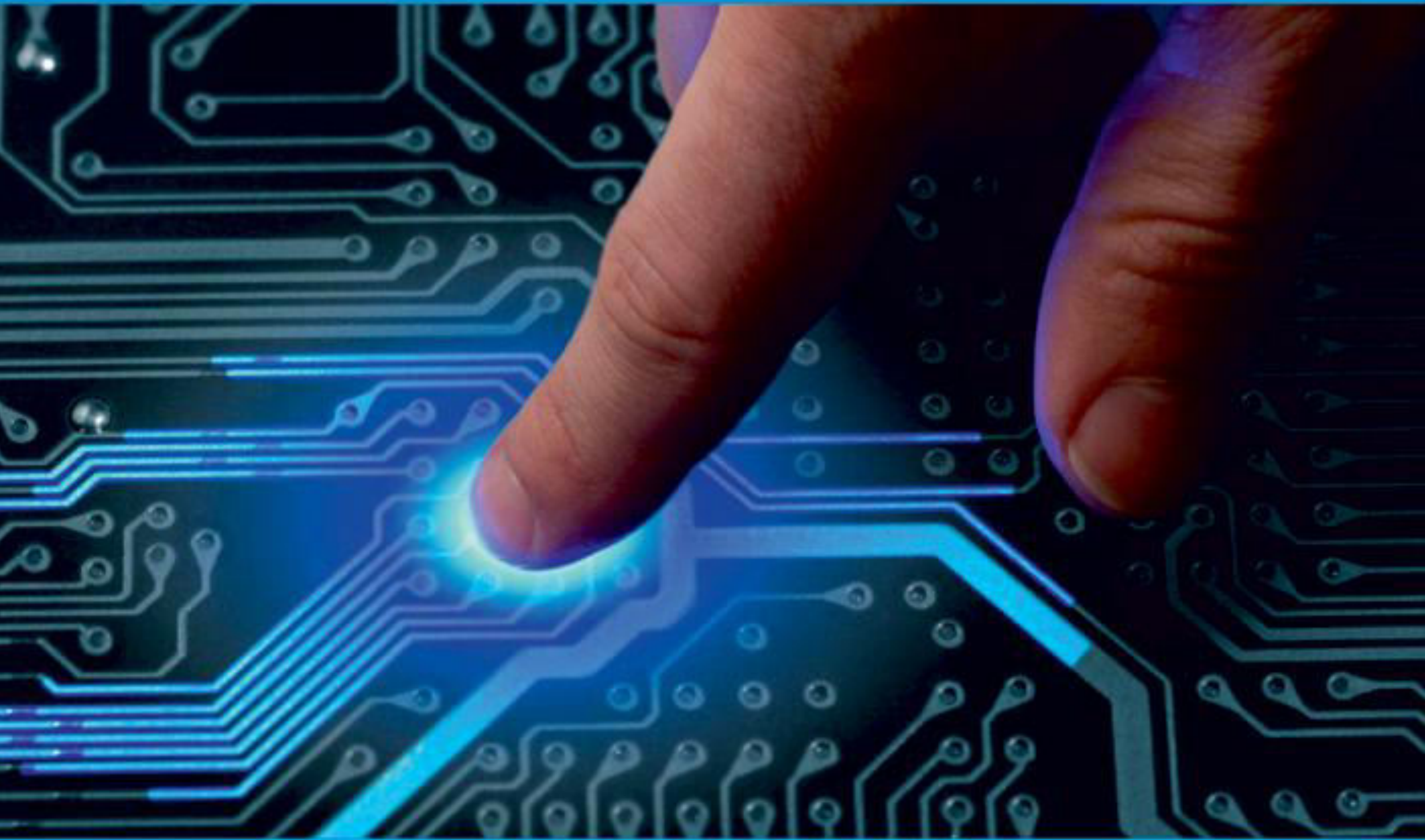
IJIRCCE

e-ISSN: 2320-9801 | p-ISSN: 2320-9798



INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH

IN COMPUTER & COMMUNICATION ENGINEERING

Volume 9, Issue 6, June 2021

INTERNATIONAL STANDARD SERIAL NUMBER INDIA

Impact Factor: 7.542

| e-ISSN: 2320-9801, p-ISSN: 2320-9798| www.ijircce.com | |Impact Factor: 7.542 |

|| Volume 9, Issue 6, June 2021 ||

| DOI: 10.15680/IJIRCCE.2021.0906277 |

Monitoring COVID-19 Social Distancing using Multiple Instance Learning

Asst. Prof Ch. Hari Prasad, N. Rakesh, P. Satya Siporah, M. Prathima, P.Sai Koushik

Asst. Professor, Department of Computer Science & Engineering, Vasireddy Venkatadri Institute of Technology, Guntur, India

Student, Department of Computer Science & Engineering, Vasireddy Venkatadri Institute of Technology, Guntur, India

Student, Department of Computer Science & Engineering, Vasireddy Venkatadri Institute of Technology, Guntur, India

Student, Department of Computer Science & Engineering, Vasireddy Venkatadri Institute of Technology, Guntur, India

Student, Department of Computer Science & Engineering, Vasireddy Venkatadri Institute of Technology, Guntur, India

ABSTRACT: In the absence of a vaccine or treatment, the most pragmatic strategies against an infectious disease pandemic are extensive early detection testing and social distancing. This study aimed to summarize public and workplace responses to Coronavirus Disease-19 and show how the system has operated during the COVID-19 pandemic.COVID-19 social distancing detector using OpenCV, Deep Learning, and Computer Vision.Detect weather the community is following social distancing or not, using live cameras and make a count who violated social distancing. Object tracking has been widely used in various intelligent systems, such as pedestrian tracking, autonomous vehicles. To solve the problem that appearance changes and occlusion may lead to poor tracking performance, we propose a multiple instance learning (MIL) based method for object tracking. In this we use YOLO v3 algorithm for object detection. This project plays an important role in an area where large numbers of people can be expected such as a shopping mall or movie theater or airport. So, with the help of this project we can ensure that people follow the process ofsocialization.

KEYWORDS: YOLO v3, Social Distancing, Pre- trained Model, Webcam, CNN.

1. INTRODUCTION

Social distancing is not a new concept. Social distancing is a method used to control the spread of contagious diseases. As the name suggests, social distancing implies that people should physically distance themselves from one another, reducing close contact, and thereby reducing the spread of a contagious disease (such as corona virus).
COVID-19 belongs to the family of corona virus caused diseases, initially reported at Wuhan, China, during late December 2019. Several health care organizations, medical experts and scientists are trying to develop proper medicines and vaccines for this deadly virus, but till date, no success is reported.
The rampant Corona virus disease has brought a global crisis with it's deadly spread to more than 180 countries. It is found that the lack of immunity against Covid-19 increases the vulnerability to the population. This is the reason that social distancing is being encouraged even after the development of vaccines, because it is the only feasible approach to stay completely safe. This situation forces the global community to look for alternate ways to stop the spread of this infectious virus. Social distancing is claimed as the best spread stopper in the present scenario, and all affected countries are locked-down to implement socialdistancing.
This research is aimed to support and mitigate the corona virus pandemic along with minimum loss of economic endeavors, and propose a solution to detect the social distancing among people gathered at any public place.
Social distancing aims at reducing the physical contact between possibly infected individuals and healthy persons. As per the WHO norms it is prescribed that people should maintain atleast
6 feet of distance among each other in order to follow social distancing. Proper social distancing is the best way to reduce infectious physical contact, hence reduces the infection rate. This reduced peak may surely match with the available health care infrastructure and help to offer better facilities to the patients battling against the corona viruspandemic.
Since the imposition of lockdowns and public restrictions in many parts of the world, the task of identifying whether a safe distance that complies with social distancing norms is presented as a new area of study in the field of computer vision. However, it is difficult for humans to estimate the distance between two people accurately from video footage

| e-ISSN: 2320-9801, p-ISSN: 2320-9798| www.ijircce.com | |Impact Factor: 7.542 |

|| Volume 9, Issue 6, June 2021 ||

| DOI: 10.15680/IJIRCCE.2021.0906277 |

alone, considering that cameras are usually angled in such a way as to maximise the field of vision. As a result, the proportions of objects in the images are distorted and make estimationinaccurate.
In this work, we propose a novel approach that allows a system to automatically detect when people are standing too close to each other, and to output an accurate estimate of the distance between them as well.

EXISTING SYSTEM

Detecting an object which is in motion, incorporates two stages: object detection and object Human detection using visual surveillance system is an established area of research which is relying upon manual methods of identifying unusual activities, however, it has limited capabilities . In this direction, recent advancements advocate the need for intelligent systems to detect and capture human activities. Although human detection is an ambitious goal, due to a variety of constraints such as low-resolution video, varying articulated pose, clothing, lighting and background complexities and limited machine vision capabilities, wherein prior knowledge on these challenges can improve the detection performance.

The primary stage of object detection could be achieved by using background subtraction, optical flow and spatio-temporal filtering techniques . In the background subtraction method , the difference between the current frame and a background frame (first frame), at pixel or block level is computed. Adaptive Gaussian mixture, temporal differencing, hierarchical background models, warping background and non-parametric background are the most popular approaches of background subtraction.

In optical flow-based object detection technique , flow vectors associated with the object's motion are characterized over a time span in order to identify regions in motion for a given sequence of images. Researchers reported that optical flow based techniques consist of computational overheads and are sensitive to various motion related outliers such as noise, colour and lighting, etc. In another method of motion detection Aslani et al. proposed spatio-temporal filter based approach in which the motion parameters are identified by using three-dimensional (3D) spatio-temporal features of the person in motion in the image sequence. These methods are advantageous due to its simplicity and less computational complexity, however shows limited performance because of noise and uncertainties on moving patterns .
Object detection problems have been efficiently addressed by recently developed advanced techniques. In the last decade, convolutional neural networks (CNN), region-based CNN and faster region-based CNN used region proposal techniques to generate the objectness score prior to its classification and later generates the bounding boxes around the object of interest for visualization and other statistical analysis .Although these methods are efficient but suffer in terms of larger training timerequirements.

PROPOSED SYSTEM

Since all those CNN based approaches utilize classification, another approach YOLO considers a regression based method to dimensionally separate the bounding boxes and interpret their class probabilities . In this method, the designed framework efficiently divides the image into several portions representing bounding boxes along with the class probability scores for each portion to consider as an object. This approach offers excellent improvements in terms of speed while trading the gained speed with the efficiency. The detector module exhibits powerful generalization capabilities of representing an entire image. For Detection we use YOLO-COCO Detector. In this we created 2 different folders one for keeping the detection and configuration and other consisting of the pre-trained model. YOLO-COCO file consists of coco.names, yolov3.cfg, yolov3.weights. We are using the pre-tarined models for object detection. Other than the two folders we have kept in the main folder, It contains we have the main python file which we are going to run in order to check if the program is running successfully or not. The main python file is interconnected internally with other two files which has configuration and the one which use the trained model to detect the people and lebel them. When we call the main file it access the webcam/input of the video path. Once the video is detected the trained model and configuration file start doing its work and the main main file then uses all the feature and detect the people in the frame. Here we also arrange a warning sound when it reaches the maximum violation count.

International Journal of Innovative Research in Computer and Communication Engineering

| e-ISSN: 2320-9801, p-ISSN: 2320-9798| www.ijircce.com | |Impact Factor: 7.542 |

|| Volume 9, Issue 6, June 2021 ||

| DOI: 10.15680/IJIRCCE.2021.0906277 |

II. IMPLEMENTATION

1.1 Flowchart

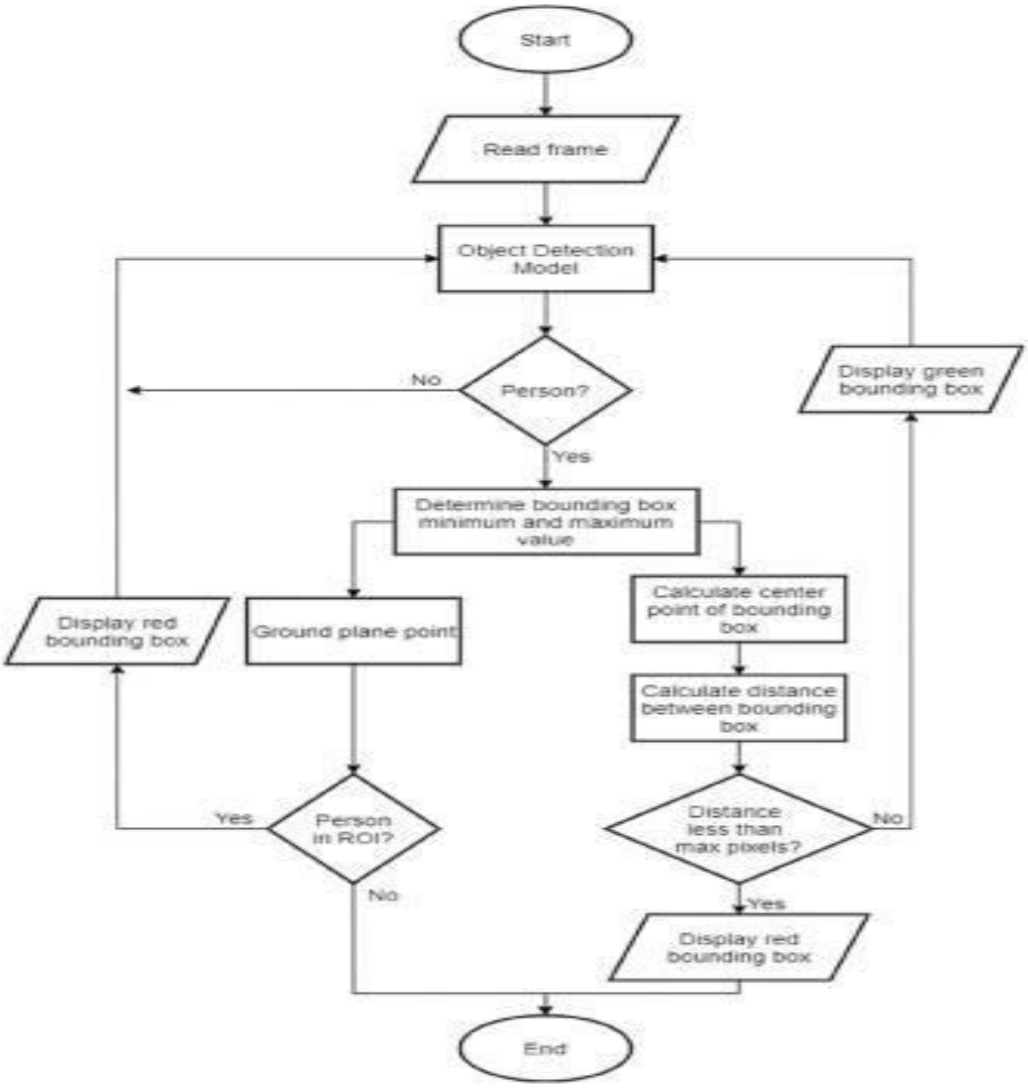


Fig 1: Flow chart diagram

1.2 Methodology

In this project, we are using a YOLO-COCO Detector algorithm. We have created 2 different folders one for keeping the detection and configuration and other consisting of the pre-trained model. YOLO-COCO file consists of coco.names, yolov3.cfg, yolov3.weights. We are using the pre-trained models for object detection. Other than the two folders we have kept in the main folder in the main folder we have the main python file which we are going to run in order to check if the program is running successfully or not. The main python file is interconnected internally with the other two files which has configuration and the one which use the trained model to detect the people and label them. When we call the main file it access the webcam of the laptop to take the input of the video file. Once the live video is detected the trained model and configuration file start doing its work and the main file then uses all the feature and detect the people in the frame not following and following social distancing. As per the program we keep an 6m distance between two persons. When the distance calculated, it checks with the constant value. Based on that it determines whether they are maintaining social distancing ornot.

| e-ISSN: 2320-9801, p-ISSN: 2320-9798| www.ijircce.com | |Impact Factor: 7.542 |

|| Volume 9, Issue 6, June 2021 ||

| DOI: 10.15680/IJIRCCE.2021.0906277 |

A schematic description of the YOLOv3 architecture is presentedbelow

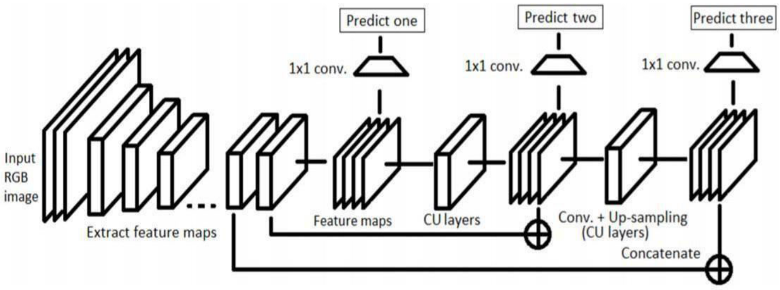


Fig1: YOLOv3architecture

Loss function: The overall loss function of YOLOv3 consists of localization loss (bounding box regressor), cross entropy and confidence loss for classification score, defined as follows:

$$\lambda_{coord} \sum_{i=0}^{S^2} \sum_{j=0}^{B} 1_{i,j}^{obj} ((t_x - \hat{t}_x)^2 + (t_y - \hat{t}_y)^2 + (t_w - \hat{t}_w)^2 + (t_h - \hat{t}_h)^2)$$
$$+ \sum_{i=0}^{S^2} \sum_{j=0}^{B} 1_{i,j}^{obj} (-\log(\sigma(t_o)) + \sum_{k=1}^{C} BCE(\hat{y}_k, \sigma(s_k)))$$
$$+ \lambda_{noobj} \sum_{i=0}^{S^2} \sum_{j=0}^{B} 1_{i,j}^{noobj} (-\log(1 - \sigma(t_o))$$

where λ_{coord} indicates the weight of the coordinate error, S^2 indicates the number of grids in the image, and B is the number of generated bounding boxes per grid.$1^{obj}{}_{i,j}$= 1 describes that object confines in the jth bounding box in grid i, otherwise it is 0.

1.3 workflow

In this section we are going to discuss the steps for represents the work flow for monitoring social distancing.

1. The trained object detection model to identify and track the person in a footage. For this object identification we use YOLOv3algorithm.
2. The trained model is feeded with the surveillance footage. The model generates a set of bounding boxes and an ID fro each identifiedperson.
3. Each individual person is associated with three-dimensional feature space (x,y,d). Here (x,y) are the centroid coordinates of the bounding box and d is the depth of the individual as observed from thecamera.

D can be calculated as:
D=((2*3.14*180)/(w+h*360)*1000+3)

Here ‘w’ is the width of the bounding box ‘h’ is the height of the bounding box

1. After getting centroid coordinates our next objective is to determine the person-to person distance between each pairwise-distinct set of people . This equates to nC2 number of distinct pairs. Where n is the no of tracked people. The following steps explain the calculation of the real distance between a pair of tracked people. For explanation purposes, we will denote two people in a pair as person 1 and person2.

First, we have to get the estimation of real distance which is calculated in previous step and denote them as Y1 and Y2. The, we calculate the absolute difference in the distance from the image plane between the two people and denote it asY12.

Y12=|Y2-Y1|

Secondly, we have to calculate the horizontal distance between the two people by finding the absolute difference in the horizontal coordinates, which are denoted as X1 and X2 respectively. It can be represented as X12.

| e-ISSN: 2320-9801, p-ISSN: 2320-9798| www.ijircce.com | |Impact Factor: 7.542 |

|| Volume 9, Issue 6, June 2021 ||

| DOI: 10.15680/IJIRCCE.2021.0906277 |

X12=|X2-X1|

Here mainly we have to remember that the distance Y12 is calculated in meters whereas X12 is calculated in pixels. So, in next step we have to convert unit of X12 inmeters.

2. After we determine the absolute difference between from the camera Y12 in meters and X12 is also in meters, we calculate the centroid-to-centroid distance in meters between person 1 and person 2. For this we use Pythagorean formula to obtain real distance between and it represented ind.

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

3. After getting distance between two persons it automatically checks that the distance between them is less than 6 m or not. If the distance between them is less than 6m it represents them in red color bounding box and else represent them in green color bounding box and it also automatically increase the violation count when they are violate socialdistance.

III. RESULTS

The final result of this project is new video file with violation count and it represents people both who are violating and who are not violating social distance. We can also save that final ouput in our pc and it also alerts a beep sound when it reaches the maximum violation count.

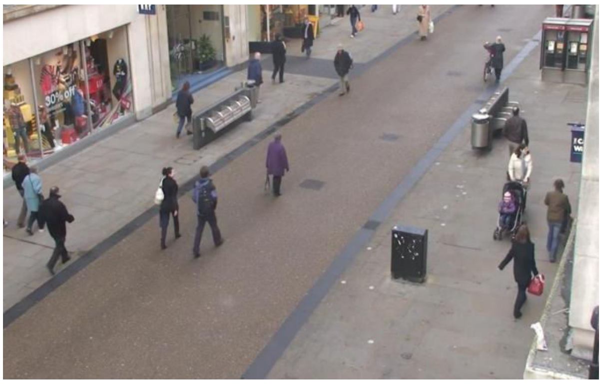


Fig 2: Inputvideofootage

Fig 3: Finaloutput

Here fig 2 represents input video file. When we given that file to our main code as input it starts detecting people by using YOLOv3 algorithm and calculating distances between them. See fig 3 this is how, final output will appear. The fig 3 represents violation count and also it shows who are violating social distance in red color bounding boxes. Here it also alert an warning sound when they reaches maximum violation count.

IV. CONCLUSION AND FUTUREWORKS

A tool for detecting social deviations we using an in-depth learning approach is proposed. Using YOLOv3 we can detect people in video footage. Using computer vision, we can calculate distance between people and also who can violate social distance represent them in red color bounding box. The proposed approched confirmed using several video footageslike

| e-ISSN: 2320-9801, p-ISSN: 2320-9798| www.ijircce.com | |Impact Factor: 7.542 |

|| Volume 9, Issue 6, June 2021 ||

| DOI: 10.15680/IJIRCCE.2021.0906277 |

walking pedestrians on the road, malls, railway station etc…,. The resulting approach is effective and was evaluated on a custom video footages dataset obtained for this research, as evidenced by the high accuracy value along with good precision and recall values. The low false alarm rate further validates the potency of the proposed approach. The approach can be fine-tuned for better performance according to the specific environment in consideration. In addition, large obstacles obstructing the field of view of the cameras may affect the tracking of people and in turn correct estimation of social distancing, which can be addressed in future work.

REFERENCES

1. World Health Organization,“WHO corona-viruses (COVID-19),” https://www. who.int/emergencies/diseases/novel-corona-virus-2019, 2020,[Online; accessed May 02,
2. 2020].
3. WHO, “Who director-general’s opening remarks at the media briefing on covid-19-11 march 2020.” https://www.who.int/dg/speeches/detail/, 2020, [Online; accessed March 12,2020].
4. L. Hensley, “Social distancing is out, physical distancing is in—here’s how to do it,” Global News–Canada (27 March 2020), 2020.
5. ECDPC, “Considerations relating to social distancing measures in response to COVID-19
   – second update,” https://www.ecdc.europa.eu/ en/publications-data/considerations, 2020, [Online; accessed March 23, 2020].
6. A.Rosebrock, “Opencv social distancingdetector,” Jun 2020. [Online]. Available:https://www.pyimagesearc h.com/2020/06/01/opencv-social-distancing-detector/
7. N. S. Punn, S. K. Sonbhadra, and S. Agarwal, “Monitoring covid-19 social distancing with person detection and tracking via fine-tuned yolo v3 and deepsort techniques,” 2020.
8. K.He, G.Gkioxari,P.Doll a´r, and R. Girshick, “Mask r-cnn,” in Proc. of IEEE International Conference on Computer Vision (ICCV), Oct 2017, pp.2980–2988.
9. https://www.pyimagesearch.com/2020/06/01/openc v-social-distancing-detector/
10. https://www.pyimagesearch.com/2018/11/12/yolo- object-detection-with-opencv/ https://www.pyimagesearch.com/2020/02/10/opencv-dnn-with-nvidia-gpus-1549-faster-
11. yolo-ssd-ad-mask-r- cnn/
12. https://www.youtube.com/watch?v=4eIBisqx9_g [12]https://machinelearningmastery.com/how-to-perform-object-detection-with-yolov3-in-ker-as/
13. https://www.analyticsvidhya.com/blog/2018/12/practical-guide-object-detection-yolo-fra mewor-python/
14. https://www.coursera.org/projects/real-time-object-detection-yolo