Real-time Face Mask and Social Distancing Violation Detection System using YOLO

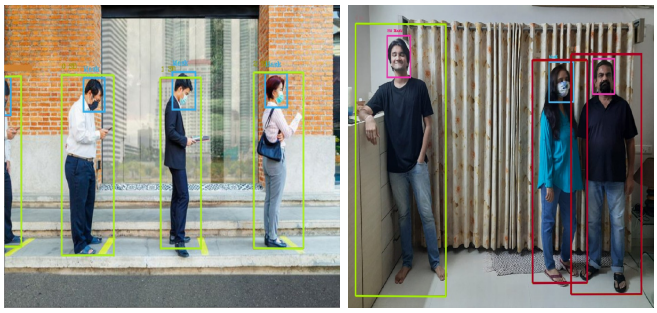
Krisha Bhambani
Dept. of Computer Engineering
Pune Institute of Computer Technology
Pune, India
krisha.bhambani@gmail.com

Tanmay Jain
Dept. of Computer Engineering
Pune Institute of Computer Technology
Pune, India
tanmayj000@gmail.com

Dr. Kavita A. Sultanpure
Dept. of Information Technology
Pune Institute of Computer Technology
Pune, India
kasultanpure@pict.edu

***Abstract*—With the recent outbreak and rapid transmission of the COVID-19 pandemic, the need for the public to follow social distancing norms and wear masks in public is only increasing. According to the World Health Organization, to follow proper social distancing, people in public places must maintain at least 3ft or 1m distance between each other. This paper focuses on a solution to help enforce proper social distancing and wearing masks in public using YOLO object detection on video footage and images in real time. The experimental results shown in this paper infer that the detection of masked faces and human subjects based on YOLO has stronger robustness and faster detection speed as compared to its competitors. Our proposed object detection model achieved a mean average precision score of 94.75% with an inference speed of 38 FPS on video. The network ensures inference speed capable of delivering real-time results without compromising on accuracy, even in complex setups. The social distancing method proposed also yields promising results in several variable scenarios.**

***Index Terms*—COVID-19, Social Distancing, Masks, YOLO, Real-time**



(a) Social distancing, face mask norms – followed by all the subjects.

(b) Social distancing, face mask norms – not followed by some subjects

Fig. 1: Images showing two different cases of violation detections.

I. Introduction

Since the COVID-19 pandemic took the world by storm, tough but necessary measures were taken by governments throughout the world to control its spread. This resulted in bringing normal day-to-day activities to a complete standstill. Months into lock down, when we see the curve flattening in several countries, the community grows restless. Relevant authorities like WHO have laid down certain guidelines to minimise people's exposure to the virus. Some safety measures people are encouraged to follow include wearing masks and maintaining a distance of 3 ft, which is approximately 1m, from another individual [1]. Fig 1 shows two test cases of our violation detector. Both figures include evaluations of people of varying heights, standing at different angles.

There are several countries in the world that have actually made mask wearing mandatory by law, and it has been observed that certain private organisations in the other countries have also been following in their footsteps. In vast establishments, it's hard to ensure that people are adhering to these crucial social distancing rules. To allow for easy tracking of such violators, an automated system is an absolute need of the hour.

We have recognized this need and have developed a model particularly suited to detect certain violations in real time. The first use of our model is to actually detect people's faces to determine whether or not they're wearing an acceptable mask. The second use is to determine whether or not social distancing is being maintained between 2 individuals, in the most efficient, accurate and simple manner, hence requiring overseeing authorities to take minimum effort.

To implement the above model, we have used object detection to detect exactly 3 classes: masked faces, unmasked faces, and people. While other models that have attempted to differentiate between masked and unmasked faces have favoured object detection networks like Single Shot Detector method [2], etc. to train on, we found that these were not efficient enough to help communities deal with potential risks in real-time.

The research we have done to add contributions to the analysis of the situation and come up with a solution to detect the violations includes:

- Data Collection from various data-sets, and self-annotations of images to test in difficult scenarios for mask detection, as well as creation of measured video test sets for social distancing.
- Self implementation of face detection using a custom-

built data-set comprising of a mixture of MAFA [3] and WIDER-FACE [4] data-sets to determine the accuracy with which masked faces can be detected.
- Thorough examination of methods to determine whether people are maintaining the recommended social distance or not, as well as development of an original method with minimum and user-friendly calibration.
- Study of several object detection methods that give maximum accuracy and FPS, so that the model has applications in real-time usage.

II. Related Work

A. Regression Based Object Detectors(YOLO)

Regression based object detectors like — You Only Look Once(YOLO) [5] and Single Shot Detector(SSD) [2] multibox have been proven to be significantly faster than region based object detectors [6]. Among the two, YOLO has long been the most popular choice among other similar object detectors. It takes as input the entire image at once, unlike region based detectors which deduce region proposals which are fed to the classifier. This makes it faster than other detectors by a wide margin. The model pipeline expects an RGB image which is divided into grid cells $S \times S$. Each grid cell is responsible for predicting B bounding boxes. For each bounding box 5 values are predicted x, y, w, h and c [5]. The coordinates of the centre point of a bounding box relative to a grid cell are x, y and the width and height of the bounding box are w and h. The confidence score of an object being present in a bounding box is c. For class probabilities C, the output of the object detector is a tensor of shape

$$(S \times S \times (B \times 5 + C)) \tag{1}$$

B. Detecting Masked Face in the Wild using LLE CNNs

In [3], the authors not only proposed the MAFA dataset, but also the use of Locally Linear Embedding(LLE) CNNs. The model categorizes faces as being covered by natural occlusions like hands and other occlusions like face masks of varying kinds. This was a breakthrough model as it beat all other models by achieving an Average Precision(AP) of 76.4% on the MAFA [3] test set for faces detection.

C. YOLOv3 and Deepsort to track individuals in Surveillance footage

Here, the authors implemented YOLOv3 [7] object detection and a Deepsort object tracking algorithm to track individuals in surveillance footage. Each individual at location (x, y) is hence mapped to a 3-Dimensional feature space (x, y, d), where d is the apparent depth of the person with reference to the camera. L2 norm is the computed for a pair of individuals. The closeness threshold for a pair of individuals is then updated dynamically based on the spatial location of person for a given range of pixels. The limitation observed here is that the threshold range is set in pixels between $(90, 170)$, which means there is no scope for calibration depending on positioning of camera.

D. Monitoring Face Masks and social distancing on surveillance footage

The study proposed by Khandelwal et al. [8] was focused on using Computer Vision based object detection models to monitor masked faces and social distancing violations using footage from surveillance cameras. This solution is specifically meant for factory setups. A two stage solution was implemented for detecting masked faces. Images are first run through a face detection model using a MobileNetV2 model [9]. The faces obtained are then classified as mask or no mask using a binary mask classifier. The model was trained on an original data set. For social distancing, the authors use SSD [2] for detection of person class. The authors have implemented a method of choosing 4 points that form a rectangle and have performed perspective transformations so that the given distances can be measured on a single plane. The comparison of these distances against the threshold requires the absolute distance between 2 points to be given. This is in fact feasible for a factory or a closed room, but for every public area, every road, this would be costly, and time consuming. It must be noted that these two models are separate entities and an integrated solution has not been presented.

III. Methodology

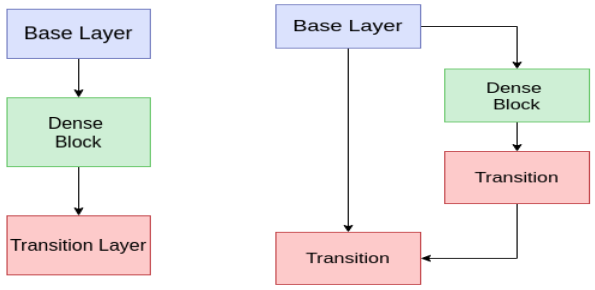
In our study, we propose a solution which performs real-time detection of individuals to track social distancing norms being followed and real-time face detection to track usage of face-masks, in several setups, including complex setups which are crowded and poorly lit. The techniques we used to formulate this solution have been described in this section.

A. Dataset

The dataset used comprises 7,959 images containing specific images from WIDER-FACE [4] and MAFA [3] datasets, with facial annotations belonging to two classes, masked faces and unmasked faces. We manually added a 3rd class by annotating individual people in every image. The bounding box coordinates and labels were then extracted from xml files for each image and normalized with respect to the height and width of the image. After verification of wrong annotations, 6,120 images were used for training and 1,839 images were set aside for validation. For social distancing, we have created our own dataset for testing the algorithm, and have also tested on several pictures from the internet where camera specifications are available(as in Fig1)

B. YOLOv4 Architecture and Functioning

Bochkovskiy et al [10] in 2020 proposed YOLOv4 with some major changes from its predecessor YOLOv3 [11], resulting in significant improvements in both speed and accuracy. YOLOv4 is extremely fast, easy to train, robust, stable and gives promising results even for tiny objects, hence, we selected it as our object detector of choice. For an input image/frame, it detects objects belonging to three classes — unmasked faces, masked faces and people. This effectively means that the same model is used for both person detection to



(a) DenseNet (b) CSPDenseNet

Fig. 2: Comparison in Structure between (a) Standard DenseNet and (b) CSPDenseNet

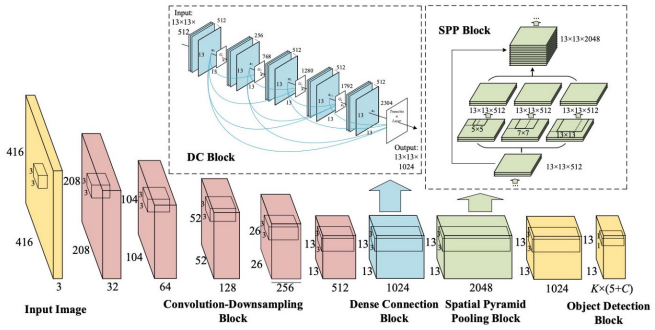


Fig. 3: Demonstration of SPP Integration with YOLOv4 Architecture

track social distancing and for masked-face detection for face-mask monitoring. This significantly boosts overall efficiency and simplicity significantly.

The pipeline has 3 parts, the backbone, the neck and the head. The network takes as input an RGB image or frame. The backbone is responsible for extracting features from the image. For this Cross-Stage-Partial-connections Darknet (CSPDarknet53) [12] proved to be the optimal choice [13]. In [12], the output from the base layer is divided into two segments. One goes to the Dense Block whereas the other goes directly to the next transition layer as show in fig. 2b. Dense blocks contain layers and each layer contains Batch Normalization and ReLU followed by a convolutional layer. Each layer of the Dense Block takes the feature maps of all previous layers as input. This expands the receptive field of the backbone and helps in isolating complex features of an image. Spatial pyramid pooling(SPP) [14] was used as the neck which contains blocks for increasing the receptive field and to aggregate parameters from different levels of the backbone. SPP's integration with the YOLO pipeline is described in fig. 3.

For this network the concept of Bag of Freebies (BoF) was adopted where freebies refer to training strategies which enhance the performance of the network without adding to its computational expense. There are multiple such freebies to choose from, out of which, Cutmix regularization [15], Mosaic data augmentation, DropBlock [16] regularization and Class label smoothing were chosen for the backbone. Strategies like Self-Adversarial Training, CIoU-loss [17], Cross mini-Batch Normalization (CmBN) [18], mosaic data augmentation, elimination of grid sensitivity and DropBlock regularization were adopted as some of the freebies for the detector. Bag of Specials (BoS) technique was also adopted wherein 'specials' refer to strategies that enhance the network's performance while increasing the inference cost by a small amount. Mish activation, Cross-stage partial connections (CSP) [12] and Multi-input weighted residual connections (MiWRC) were the specials chosen for the backbone, and Mish activation, SPP [14], SAM [19] and PAN [20] were chosen as specials for the detector.

C. Tracking Social Distancing

The solution proposed in our paper includes 2 steps: calibration and testing. In calibration, there are 2 steps:

- The user is required to input the focal length, and the sensor dimensions of the camera to be used.
- The user is then required to position these 2 individuals at the minimum social distance that is to be maintained, henceforth referred to as the reference social distance.

The advantage of this is that authorities can actually select the social distance they want maintained, according to the specific guidelines that they wish to follow. For eg. : WHO guidelines mention the recommended amount of social distance to be a minimum of 3 ft. [1], while CDC guidelines recommend the minimum to be 6 ft [21]. The social distance between 2 people is solely judged relative to the initial calibration, and the absolute distance need not be provided to the model.

This Social Distancing model uses the principle that a camera lens is essentially a convex lens, where the image is essentially captured on a screen. For this model, as aforementioned, we need the focal length and sensor dimensions.

The focal length of a lens is the distance from the optic center of the lens to its focus. In optics and photography, the focal length is measured in millimeters(mm). Longer the focal length, higher the magnification, but lower the angle of view. [22]

An image sensor is a device found in the hardware of a camera, that uses light to detect information to convert a view into an image. The sensor essentially functions as screen where all the pixels in an image are mapped. Greater the pixels mapped, greater the image quality.

Let field width, i.e., the observed width of a real life object be w and measured distance of an object from the camera be d.

Let the people between whom the distance is being measured be: Person 1 and Person 2, that stand at positions $(x1, y1)$ and $(x2, y2)$ in a given image. These coordinates are detected at the feet of the individuals. Through Fig 4, using similarity, we can show that [23]

$$\frac{\text{sensor dimension}}{\text{focal length}} = \frac{\text{field dimension}}{\text{distance to field}} \tag{2}$$

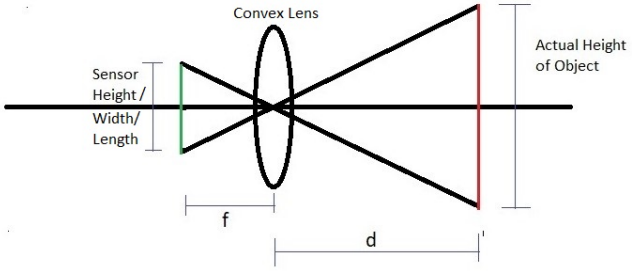


Fig. 4: Basic Visualisation of the Working of a Camera

To find the depth of an object in a photograph, the following formula can be obtained from equation 2 :

$$d = \frac{\text{actual ht of object(mm)} \times \text{focal length(mm)}}{\text{ht of object on sensor(mm)}} \quad (3)$$

where height is ht. Since we only have the height of the object in pixels(px.) in the image, we can use the following formula to obtain the height of the object in the image on the sensor in millimeters. Here, the height of an actual human is assumed to be 1.6m in this model, since the average height of a human is estimated to be that much. [24]

$$\text{object ht on sensor(mm)} = \text{object ht in image(px.)} \times \text{pixel size} \quad (4)$$

where px is measurement in number of pixels. Hence, the depth of a person would be equal to the distance a person stands from the camera. This distance from the camera for the 2 people will be represented as $d1$ and $d2$. To measure the approximate distance between these two people in the image, the difference of their x-coordinates are taken to be the social distance width.

$$\text{social distancing width(mm)} = (|x1 - x2|) \times \text{pixel size} \quad (5)$$

We can hence find out the actual field width using eqn. 2

$$w = \frac{\text{sensor width} \times \text{social distancing width(mm)}}{\text{focal length(mm)}} \quad (6)$$

If person 1 is assumed to be at $(0, d1)$ and person 2 at $(w, d2)$

$$\text{social distance} = \sqrt{(w-0)^2 + (d2-d1)^2} \quad (7)$$

To obtain the pixel size, the following formula must be used: [25]

$$\text{pixel size} = \frac{\frac{\text{sensor width(mm)}}{\text{width of image(px.)}} + \frac{\text{sensor height(mm)}}{\text{height of image(px.)}}}{2} \quad (8)$$

The social distance, which is first calculated in calibration mode, will be used as the reference social distance. In testing mode, the social distance between 2 individuals will be calculated using the equations shown above. If the calculated social distance is lesser than the reference social distance, the pair of individuals will be identified as violators.

IV. Experiments and Results

A. Metrics

To judge the performance of our solution, we evaluated certain metrics which have been discussed below.

1) Precision:

$$\frac{TP}{TP + FP} \quad (9)$$

2) Recall:

$$\frac{TP}{TP + FN} \quad (10)$$

where, TP = True Positives, FP = False Positives, FN = False Negatives.

3) F1 Score:

$$2 \times (\frac{Precision \times Recall}{Precision + Recall}) \quad (11)$$

4) Intersection over Union (IoU):

$$\frac{area(B_p \cap B_{gt})}{area(B_p \cup B_{gt})} \quad (12)$$

Where B_p is the region of Predicted Bounding Box and B_{gt} is the region of Ground Truth Bounding Box

5) Average Precision(AP): We followed PASCAL VOC 2010-2012 format for calculating the AP. For precision as a function of recall, $P(R)$:

$$P_{inter}(R) = \max_{R' \geq R} P(R') \quad (13)$$

$$AP_{0.5} = \sum_{i=1}^{n} (R_{i+1} - R_i) P_{inter}(R_{i+1}) \quad (14)$$

B. Experimental Setup

YOLOv4 was built on the Darknet Framework and was trained using NVIDIA Tesla P100 PCIE Graphics Processing Unit (GPU) with 16 gigabytes of memory, and 2.30GHz Intel Xeon CPU. For training, we set the hyper-parameters of the network as follows:

- Number of Steps: 8000
- Batch-Size: 64
- Mini Batch-Size: 64
- Momentum: 0.949
- Decay: 0.0005
- Initial Learning Rate: 0.001

We trained two models, one with all three classes — Unmasked face, masked face and person and the other with only two classes — Unmasked face and masked face. The models were trained on 6,120 images. The total number of iterations for which the model was scheduled to train for was 8000. After every 680 iterations, the current state of the model was evaluated by recording the AP for every class, precision, recall, F1 score and the mean AP (mAP). The weights of the model were saved for every 100 iterations completed, for future evaluation.

To test the social distancing model, we used a camera of:

- Focal length : 4.15mm
- Sensor dimensions : 4.80mm x 3.60m

Fig. 5: Inferences of Social Distance tracking on our customized set of pictures

- Camera height : 2.2 m

The calibration reference image involved two people standing at a distance of 1m, while the testing images consisted of people standing at varying positions with respect to each other.

C. Results

Table I and II show metrics recorded for both models where type A is the denotation of the YOLOv4 model which was trained to detect all 3 classes — Faces with no mask, faces with mask and entire people. On the other hand, type B is the denotation of the model which was trained to detect only faces without mask and faces with mask. AP_0, AP_1, AP_2 are the Average Precision for classes unmasked face, masked face and person respectively. The two models A and B delivered promising mAPs of 94.75% and 95.00% respectively, on the validation set for an IoU threshold of 50%. Fig. 6 is the plot of Precision v/s recall for all three classes. It can be inferred from the trajectory of the plot that the model got maximum positive classifications for all three classes right. The average FPS the model achieved was 38 FPS on the NVIDIA Tesla P100 GPU. As compared to the paper by Khandelwal et al. [8], that yields a mAP@ .50 IoU of 0.897 for object detection, our model has a significantly higher performance.

Through Tables III and IV, as well as Fig. 7 we can see that the social distancing model gives an extremely good estimation of whether social distancing is being violated or not, even in the most challenging cases, as shown in Fig. 5, without adding costly computations that reduce the overall FPS.

In Table IV, containing the results for test set shown in Fig. 5, "Maintained" means that the distance measured between a person with respect to everyone else is greater than or equal

TABLE I: Average Precision Metrics for IoU threshold = 50%

Type	AP_0	AP_1	AP_2	mAP
A	94.02%	95.53%	94.70%	94.75%
B	94.06%	95.93%	-	95.00%

TABLE II: Precision, Recall and F1 scores of both models

Type	Precision	Recall	F1 Score
A	0.89	0.93	0.91
B	0.90	0.91	0.90

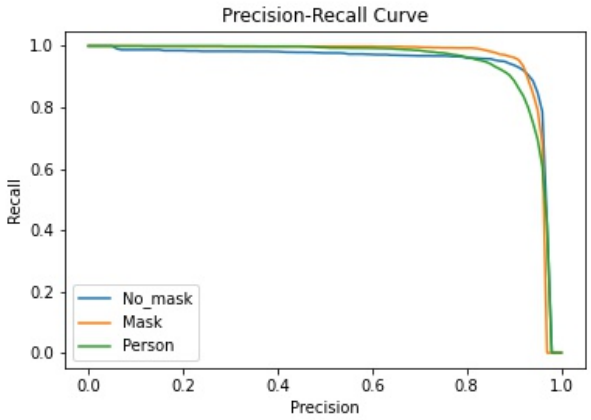


Fig. 6: Precision-Recall Curve for IoU Threshold at 50%

TABLE III: Distances at which violation of social distance is detected at various points from the position of the camera

Distance of closest subject from Camera(m)	Calibrated Social Distance(m)	Measured Social Distance(m)	Error(m)
2	1	1.13	0.13
4	1	1.044	0.044
6	1	1.031	0.031
8	1	0.952	0.048
10	1	1.336	0.336

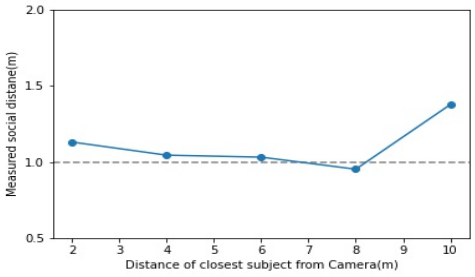


Fig. 7: Plot showing differences in measured social distance and actual social distance at varying distances of closest subjects from camera.

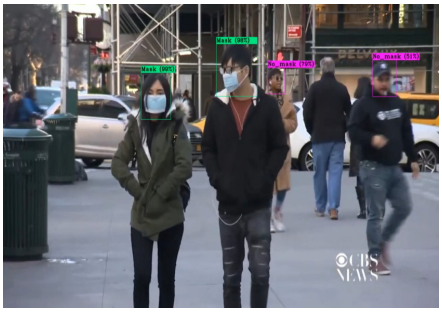


Fig. 8: Type B model inference for only masked and unmasked faces

TABLE IV: Result of social distancing tests

Label	Actual Distance between two people	Prediction made
a (Reference)	1m	-
b (person 1 - 2)	3m	Maintained
c (person 1 - 2)	2m	Maintained
d (person 1 - 2)	1m	Maintained
e (person 1 - 2)	1m	Maintained
f (person 1 - 2)	2m	Maintained
g (person 1 - 2)	3m	Maintained
h (person 1 - 2)	0.25m	Violated
i (person 1 - 2)	0.5m	Violated
i (person 1 - 3)	6m	Maintained
i (person 2 - 3)	7m	Maintained

to the minimum required social distance to be maintained. "Violated" means that the distance measured between a person with respect to everyone else is lesser than the minimum required social distance to be maintained. If an individual is detected violating the social distancing norms, he/she is bound by a red bounding box and if not, in a green bounding box. For face mask detection, if a person is found wearing a mask his face is bound by a blue bounding box, if not, is bound by a pink box as shown in Fig. 1, 5.

V. Conclusion

We have hence created a well integrated real time face mask and social distancing violation detection system, where object detection takes place using YOLO v4. The three classes that are simultaneously detected are masked and unmasked faces, as well as whole people. Using the coordinates given by the detection of the class person, the relative distance between 2 individuals is hence estimated using the principles of optics. After rigorous testing, we observe that the model yields fairly accurate results for a wide field of view, which is an essential criteria for usage in public places. Without any addition of time consuming computations or image warping, this light weight model is easy to calibrate and can be well used in real time due to high FPS and good accuracy.

References

[1] WHO, "Coronavirus disease (COVID-19) advice for the public", 2020, Available: https://www.who.int/emergencies/diseases/novel-coronavirus-2019/advice-for-public

[2] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, and S. Reed, "SSD: Single shot multibox detector", 2015, arXiv:1512.02325

[3] Ge, J. Li, Q. Ye and Z. Luo, "Detecting Masked Faces in the Wild with LLE-CNNs," 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, 2017, pp. 426-434, doi: 10.1109/CVPR.2017.53.

[4] Shuo Yang, Ping Luo, Chen Change Loy, Xiaoou Tang, "WIDER FACE: A Face Detection Benchmark", 2015, arXiv:1511.06523

[5] J. Redmon, S. Divvala, R. Girshick and A. Farhadi, "You Only Look Once: Unified, Real-Time Object Detection," 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, 2016, pp. 779-788, doi: 10.1109/CVPR.2016.91.

[6] R. Girshick, J. Donahue, T. Darrell and J. Malik, "Rich Feature Hierarchies for Accurate Object Detection and Semantic Segmentation," 2014 IEEE Conference on Computer Vision and Pattern Recognition, Columbus, OH, 2014, pp. 580-587, doi: 10.1109/CVPR.2014.81.

[7] Narinder Singh Punn, Sanjay Kumar Sonbhadra and Sonali Agarwal, "Monitoring COVID-19 social distancing with person detection and tracking via fine-tuned YOLO v3 and Deepsort techniques", 2020, arXiv:2005.01385.

[8] Prateek Khandelwal, Anuj Khandelwal, Snigdha Agarwal, Deep Thomas, Naveen Xavier and Arun Raghuraman, "Using Computer Vision to enhance Safety of Workforce in Manufacturing in a Post COVID World", 2020, arxiv:2005.05287.

[9] Andrew G. Howard, Menglong Zhu, Bo Chen, Dmitry Kalenichenko, Weijun Wang, Tobias Weyand, Marco Andreetto, Hartwig Adam, "MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications," arXiv:1704.04861.

[10] Alexey Bochkovskiy, Chien-Yao Wang, and Hong-Yuan Mark Liao, "Yolov4: Optimal speed and accuracy of object detection", 2020, arXiv:2004.10934.

[11] Joseph Redmon, Ali Farhadi, "YOLOv3: An Incremental Improvement", 2018, arXiv:1804.02767.

[12] Chien-Yao Wang, Hong-Yuan Mark Liao, Yueh-Hua Wu, Ping-Yang Chen, Jun-Wei Hsieh, and I-Hau Yeh, "CSPNet: A new backbone that can enhance learning capability of cnn", IEEE Conference on Computer Vision and Pattern Recognition Workshop (CVPR Workshop), 2020.

[13] Hui, "YOLOv4", Medium, 2020. [Online]. Available: https://medium.com/@jonathan_hui/yolov4-c9901eaa8e61

[14] Zhanchao Huang, Jianlin Wang, "DC-SPP-YOLO: Dense Connection and Spatial Pyramid Pooling Based YOLO for Object Detection", 2019, arXiv:1903.08589.

[15] Sangdoo Yun, Dongyoon Han, Seong Joon Oh, Sanghyuk Chun, Junsuk Choe and Youngjoon Yoo, "CutMix: Regularization Strategy to Train Strong Classifiers with Localizable Features", 2019, arXiv:1905.04899

[16] Golnaz Ghiasi, Tsung-Yi Lin, and Quoc V Le. DropBlock: "A regularization method for convolutional networks", Advances in Neural Information Processing Systems (NIPS), 2018, pages 10727–10737.

[17] Zhaohui Zheng, Ping Wang, Wei Liu, Jinze Li, Rongguang Ye, and Dongwei Ren, "Distance-IoU Loss: Faster and better learning for bounding box regression", Proceedings of the AAAI Conference on Artificial Intelligence (AAAI), 2020.

[18] Zhuliang Yao, Yue Cao, Shuxin Zheng, Gao Huang, and Stephen Lin, "Cross-iteration batch normalization", 2020, arXiv:2002.05712.

[19] Sanghyun Woo, Jongchan Park, Joon-Young Lee, and In So Kweon "CBAM: Convolutional block attention module", European Conference on Computer Vision (ECCV),2018, pages 3–19.

[20] Shu Liu, Lu Qi, Haifang Qin, Jianping Shi, and Jiaya Jia, "Path aggregation network for instance segmentation", IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2018, pages 8759–8768.

[21] "Coronavirus Disease 2019 (COVID-19)", Centers for Disease Control and Prevention, 2020, Available: https://www.cdc.gov/coronavirus/2019-ncov/prevent-getting-sick/social-distancing.html

[22] What is Focal Length? - Definition from Techopedia", Techopedia.com, 2020. [Online]. Available: https://www.techopedia.com/definition/2701/focal-length.

[23] Scantips.com, 2020, "The Math Of Camera Field Of View Calculations (FOV)", [online] Available at: https://www.scantips.com/lights/fieldofviewmath.html

[24] Max Roser, Cameron Appel, Hannah Ritchie , "Human Height". Published online at OurWorldInData.org, 2013, Available: https://ourworldindata.org/human-height

[25] "The Math of camera Field of View Calculations (FOV)", Scantips.com, 2020. [Online]. Available: https://www.scantips.com/lights/fieldofviewmath.html