



e-ISSN: 2319-8753 | p-ISSN: 2347-6710

IJRSET

International Journal of Innovative Research in
SCIENCE | ENGINEERING | TECHNOLOGY

INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH

IN SCIENCE | ENGINEERING | TECHNOLOGY

Volume 12, Issue 3, March 2023

ISSN INTERNATIONAL
STANDARD
SERIAL
NUMBER
INDIA

Impact Factor: 8.423

9940 572 462

6381 907 438

ijirset@gmail.com

www.ijirset.com

Virus Protocol Detection System Using Deep Learning

Suraj Sutar, Adesh Thombre, Jayesh Munot

Dept. of Computer Engineering, KJCOEMR, Pune, Maharashtra, India

Dept. of Computer Engineering, KJCOEMR, Pune, Maharashtra, India

Dept. of Computer Engineering, KJCOEMR, Pune, Maharashtra, India

ABSTRACT: The Virus had a huge impact on different sectors in many countries and such impact caused problems to many people around the world. Alleviation in such pandemic can be solved by social distancing as well as putting on its face mask. This small step of wearing a face mask as well as following social distancing would save lots of lives as the spread of the virus could be mitigated.

This project is implemented by considering these facts of Virus protocols. The project aims to detect face mask, social distancing on video feed and also reminder for vaccination by using deep learning models. SSD algorithm is used for face mask detection and if mask is not wear then recognition of face will be done by using face recognition algorithm. YOLO object detection is used to detect the people in frame and distance is measured by using Euclidean formula. This project also has one more feature that is detection for vaccination, so next dose will be calculated and reminder will be sent to user.

Also developed a desktop application where admin will get notified and can monitor the violations. If violation happens, data will get stored into firebase database and it will also notify admin. The desktop application has features of viewing history of violated data, notification functionality, user registration, etc.

KEYWORDS: YOLO Object Detection, Firebase, SSD, face recognition

I.INTRODUCTION

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 favored object detection networks like Single Shot Detector method 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 and WIDER-FACE 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.

Social Distance:

Face Mask Detection:

For detection of face mask required modules like 'Face Recognition' and 'SSD' are loaded into the system. 'SSD Module' plays an important role to check whether mask is wear properly or not. If not, the module will detect user and notify to that user. SSD stands for 'Single Shot Multibox Detector'. It is a technique that is used to detect objects in images using a single deep neural network. Basically it is used for object detection in an image. We are using SSD for detecting mask as an object in our system. For this purpose we are labelling our image dataset (with-mask and without-mask). By labelling it means we are plotting co-ordinates for the position of mask over face. As these co-ordinates in turn will be used to detect whether person has wear mask properly or not.



Face Recognition:

For detection of user, firstly User dataset will be loaded which is needed to train the model. Then we take the frame as input from the face mask detection module. After getting the frame, detection of face is done using SSD, extracting features of detected face is done using Face recognition. Finally classifying the extracted feature using KNN and then get the details of the person who has not wear the mask properly. Once we get the details of that person we will notify that user. There is a possibility that the user may avoid our notification, so we calculate how often user is breaking the rules. Once the count for breaking the rule goes beyond standard count, system will notify to the organization.

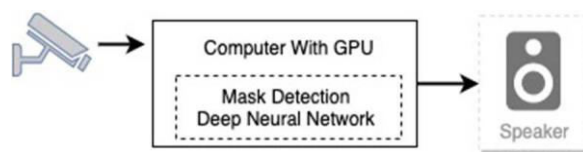
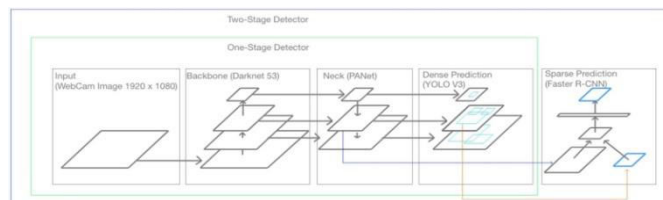


Fig. The block diagram of the face mask detection hardware.



The Hardware Elements Of face Mask Detection:

This section will explain about the hardware architecture of our system. The hardware diagram system can be seen on Fig. 1. The whole hardware system on Fig. 1 consists of a digital webcam camera, PC, and a speaker. In order to do the face mask detection, a deep neural network has been chosen as a detection method. All the computation of the face detection is done in a computer which is mounted to the GPU to enhance the graphical calculation of the image. This system is running in real-time application when the camera detected the user who wear or un-wear the face mask. When the system detected the user who un-wear or wear face mask from the camera, it will order the speaker

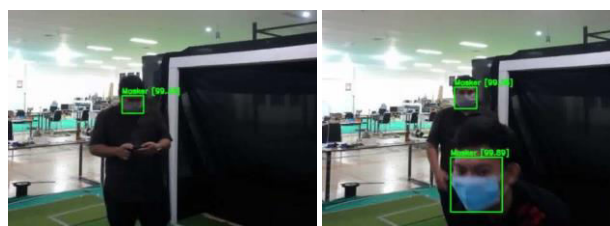
to alert him/her to wear the face mask and will be going on until the user put their face mask properly. The prototype of the face mask detector is presented on Fig. 2. This figure shows the position of each parts of the face mask detector when it is operated in real-time application. The webcam camera, as seen on Fig. 2, was mounted on the top of the monitor, which allowed the detection of the user at around 5 meters. The PC and speaker on the other hand, was put on the back of the monitor for the safety reason and to make it more compact so that it can be moved everywhere freely. The prototype, illustrated on Fig. 2, is equipped by MiniPC embedded with GTX 1060 Nvidia GPU for the object detection method calculation.

The face Mask Detection:

As for the face mask detector method, this work implemented the deep neural network known as YOLO V4. According to, the YOLO V4 is able to run twice faster than the other deep neural network method which is used to detect the object. The performance of this version is able to improve the YOLOv3' AP by 10% and FPS by around 12%. By reflecting these results, it is suitable to implement the method into the real-time face mask detector where high detection accuracy is needed. In this work, the object which needs to be detected is the face mask wearer. The YOLO V4 which has been implemented in this work consists of two-stages detector. As seen on Fig. 3, the first-stage detector consists of input, backbone, neck, and dense prediction. Moreover, the second- stage of detector has sparse prediction to predict the object by understanding the bounding boxes and the classes on the object. The input image of this work needs to be handled in the resolution of around 1920x1080 pixel in real-time application to ensure the detection of the moving object who wear the face-mask properly.

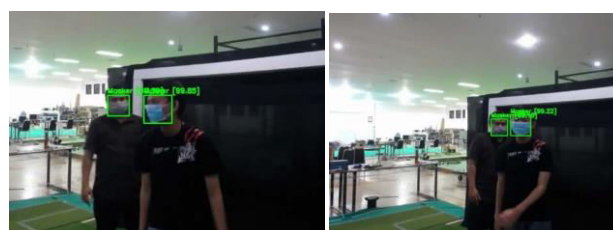
II. EXPERIMENTAL RESULTS

This section will present the experiment results of the face mask detection in real-time application and that has already been installed at Politeknik Negeri Batam. The first experiment which is depicted on Fig. 4 has been done as the first trial before it is implemented for the moving person. Fig. 4(a) illustrates the face detector that detected the single user wearing a face mask accurately even it has some disturbance in the area. As for Fig. 4(b), the user was added slowly from below the camera and the detector was able to detect the mask properly. When the users are standing close to each other as seen on Fig. 4 (c)-(f), this system was also able to detect the face mask even if the user was surrounded by many objects with similar color. After did the trial with no error, we are ready to verify this device with more user. As seen on Fig. 5(a)-(c), we added the user into three people with different types of face mask such as surgical and fabric face mask. Each person was standing in different position to verify the performance of face mask detection. From the picture it is verify that the face detected remains steady in detecting face mask of the users even the lighting was in different brightness. To make different in brightness, we turned off and on the lamp at our lab as to test the feature of this system. On the other hand, the experiment of detecting a non- wearing mask is presented on Fig. 6(a)-(c). On Fig. 6(a) the first user who wear a white T-shirt attempted to pull off his mask, and the mask detector was able to distinguish the non- wearing mask and mask-wearing user. Fig. 6(b) also presented the mask detection precisely. Moreover, on Fig. 6(c) the first and third users tried to take off their mask, and the mask detector detected the face mask condition steadily.



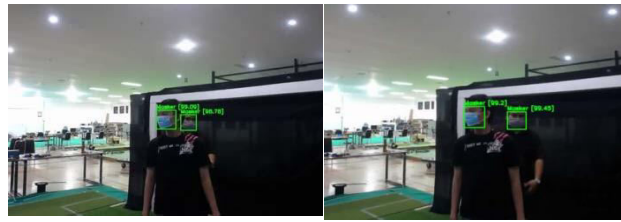
(a)

(b)



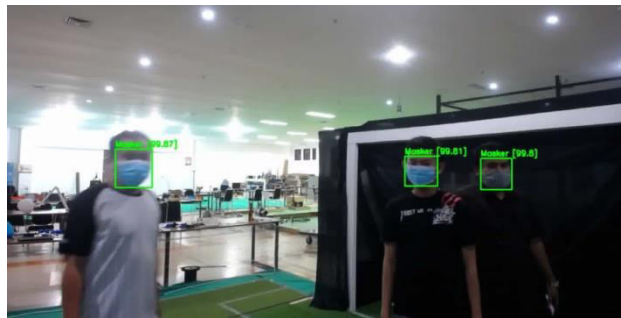
(c)

(d)

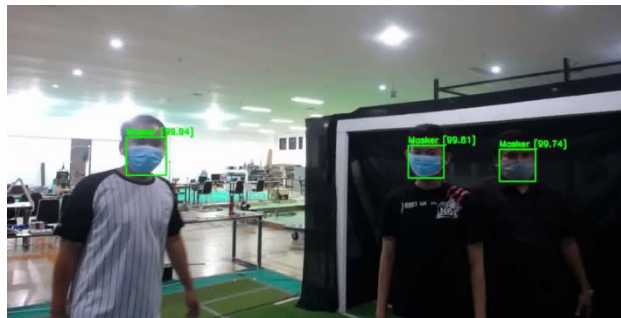


(e)

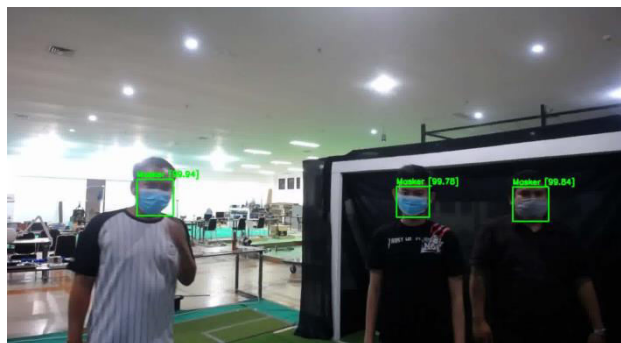
(f)



(a)



(b)



(c)

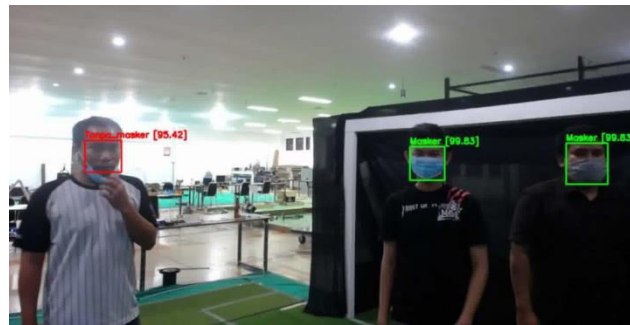
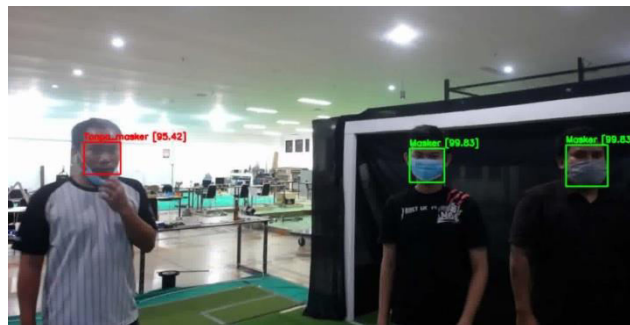
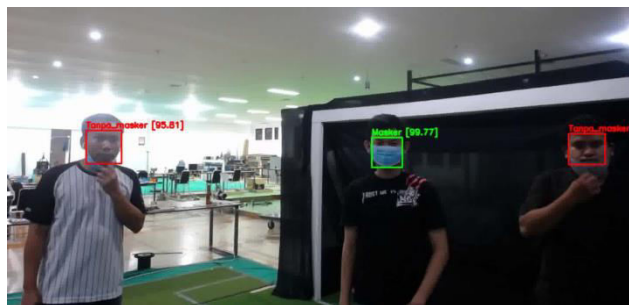


Fig. The detect or detected multiple people who are wearing a face mask with different position from each other.



(b)



(c)

Fig. The face mask detection detected the non-wearing user with different angle of pose



When the user attempted not to wear the face mask properly, represented on Fig. 8, this device will announce that the user needs to wear the mask by the speaker plugged into the device. Fig. 8 (a) illustrates the non-wearing user of face

mask and Fig. 8 (b) for wearers of face mask. From all these experiments, the average FPS generated by the face detection is about 11,1 FPS which is presented on Fig. 9. From all these experiments, the face detector which is built by YOLO v4 algorithm is able to detect and distinguish a non-wearing and a wearing-mask user properly in every different situation such as lighting, mess up area, and clean area.

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, while CDC guidelines recommend the minimum to be 6 ft . 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. 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 sensor dimension focal length = field dimension distance to field.

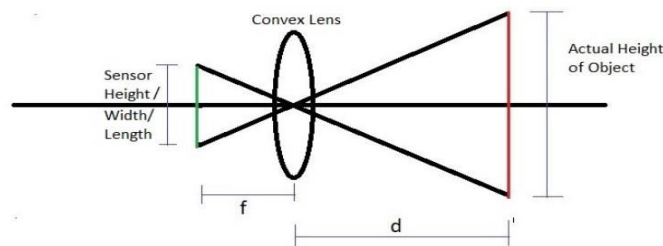


Fig. 4: Basic Visualisation of the Working of a Camera

To find the depth of an object in a photograph, the following formula can be obtained from equation 2 : $d = \text{actual ht of object(mm)} \times \text{focal length(mm)} / \text{ht of object on sensor(mm)}$ (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. $\text{object ht on sensor(mm)} = \text{object ht in image(px.)} \times \text{pixel size}$ (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 distancing width. $\text{social distancing width(mm)} = (|x1 - x2|) \times \text{pixel size}$ (5) We can hence find out the actual field width using eqn. 2 $w = \text{sensor width} \times \text{social distancing width(mm)} / \text{focal length(mm)}$ (6) If person 1 is assumed to be at $(0, d1)$ and person 2 at $(w, d2)$ $\text{social distance} = \sqrt{(w - 0)^2 + (d2 - d1)^2}$ (7) To obtain the pixel size, the following formula must be used: $\text{pixel size} = \text{sensor width(mm)} / \text{width of image(px.)} + \text{sensor height(mm)} / \text{height of image(px.)}$ (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.

II. RELATED WORK

This section highlights some of the related works about human detection using deep learning. A bulk of recent works on object classification and detection involve deep learning are also discussed. The state-of-the-art review mainly focuses on the current research works on object detection using machine learning. Human detection can be considered as an object detection in the computer vision task for classification and localization of its shape in video imagery. Deep learning has shown a research trend in multi-class object recognition and detection in artificial intelligence and has achieved outstanding performance on challenging datasets. Nguyen et al. presented a comprehensive analysis of state-of-the-art on recent development and challenges of human detection. The survey mainly focuses on human descriptors, machine learning algorithms, occlusion, and real-time detection. For visual recognition, techniques using deep convolutional neural network (CNN) have been shown to achieve superior performance on many image recognition benchmarks. Deep CNN is a deep learning algorithm with multilayer perceptron neural networks which contain several convolutional layers, sub-sampling layers, and fully connected layers. Later, the weight in the whole layers in the networks are trained for each object classification based on its dataset. For object detection in image, the CNN model was one of the categories in deep learning which are supervised feature learning methods robust in detecting the object in different scenarios. CNN has achieved great success in large-scale image classification tasks due to the recent high-performance computing system and large dataset such as ImageNet. Different CNN models for object detection with its object localization had been proposed in terms of network architecture, algorithms, and new ideas. In recent years, CNN models such as Alex Net, VGG16, InceptionV3, and ResNet-50 are trained to achieve outstanding results in object recognition. The success of deep learning in object recognition is due to its neural network structure that is capable of self-constructing the object descriptor and learning the high-level features which are not directly provided in the dataset. The current state-of-the-art object detectors with deep learning had their pros and cons in terms of accuracy and speed. The object might have different spatial locations and aspect ratios within the image. Hence, the real-time algorithms of object detection using the CNN model such as R-CNN and YOLO had further developed to detect multi-classes in a different region in images had been developed. YOLO (You Only Look Once) is the prominent technique for deep CNN-based object detection in terms of both speed and accuracy. The illustration for the YOLO model is shown in Figure 1. Adapting the idea from the work, we present a computer vision technique for detecting people via a camera installed at the roadside or workspace. The camera field-of-view covers the people walking in a specified space. The number of people in an image and video with bounding boxes can be detected via these existing deep CNN methods where the YOLO method was employed to detect the video stream taken by the camera. By measuring the Euclidean distance between people, the application will highlight whether there is sufficient social distance between people in the video.

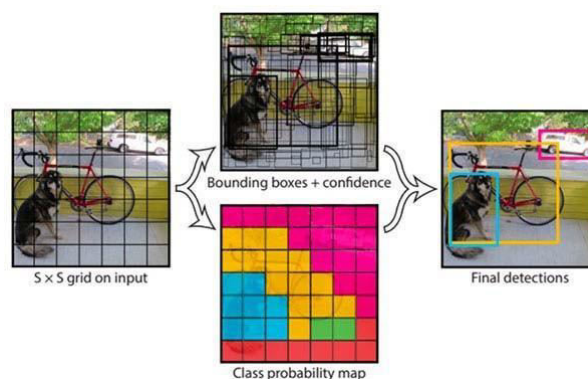


Figure 1 The illustration of YOLO model for object detector pipeline.

III. METHODOLOGY

This social distancing detection tool was developed to detect the safety distance between people in public spaces. The deep CNN method and computer vision techniques are employed in this work. Initially, an open-source object detection network based on the YOLOv3 algorithm was used to detect the pedestrian in the video frame. From the detection result, only pedestrian class was used and other object classes are ignored in this application. Hence, the bounding box best fits for each detected pedestrian can be drawn in the image, and these data of detected pedestrians will be used for

the distance measurement. For camera setup, the camera is captured at fixed angle as the video frame, and the video frame was treated as perspective view are transformed into a two-dimensional top- down view for more accurate estimation of distance measurement. In this methodology, it is assumed that the pedestrians in the video frame are walking on the same flat plane. Four filmed plane points are selected from frame and then transformed into the top-down view. The location for each pedestrian can be estimated based on the top-down view. The distance between pedestrians can be measured and scaled. Depending on the preset minimum distance, any distance less than the acceptable distance between any two individuals will be indicated with red lines that serve as precautionary warnings. The work was implemented using the Python programming language. The pipeline of the methodology for the social distancing detection tool is shown in Figure 2.

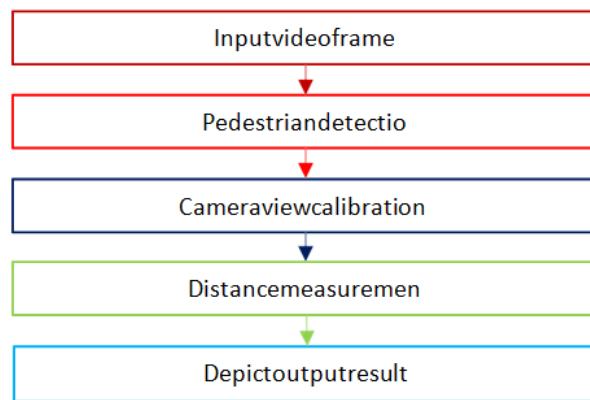


Figure 2 Pipeline for social distancing detection.

IV.CONCLUSIONS

The proposed and implemented system aims to help people in private sectors to follow the rules. We have provided the best possible solution to reduce the problem of violation of Virus rules. The major concern that private sector facing is, proper follow of Virus protocols. Through this system, we can monitor the people in the organization that leads to help follow the rules and also reduce the new Virus cases. System also have functionality of alert, so that people gets the notification if they unfollow the rules.

REFERENCES

1. <https://www.scitepress.org/Papers/2021/104831/104831.pdf>
2. <https://ieeexplore.ieee.org/document/9342585>
3. <https://viso.ai/deep-learning/object-detection/>
4. <https://www.pyimagesearch.com/2020/05/04/Virus-face-mask-detector-with-opencv-keras-tensorflow-and-deep-learning/>



INNO SPACE
SJIF Scientific Journal Impact Factor
Impact Factor: 8.118



ISSN INTERNATIONAL
STANDARD
SERIAL
NUMBER
INDIA



INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH

IN SCIENCE | ENGINEERING | TECHNOLOGY

 **9940 572 462**  **6381 907 438**  **ijirset@gmail.com**



www.ijirset.com

Scan to save the contact details