IJCRT.ORG ISSN : 2320-2882

INTERNATIONAL JOURNAL OF CREATIVE RESEARCH THOUGHTS (IJCRT)

An International Open Access, Peer-reviewed, Refereed Journal

SOCIAL DISTANCE VIOLATION AND FACE MASK DETECTION USING YOLOV3 DEEP LEARNING OBJECT DETECTION ALGORITHM

[1]Dr.K.Soumya, [2]Polisetti Srividya, [3]Potturi Keerthana, [4]Pudi Ikshitha, [5]R.H.S.V.Harshini, [6]S.S.V.Charishma

[1]Assistant Professor (C), [2,3,4,5,6] Student

Department of Computer Science and System Engineering,

Andhra University College of Engineering for Women, Visakhapatnam, Andhra Pradesh, India

Abstract: Since 2019 Covid caused a huge death rate among different age categories of the people. This virus makes people to realize about the importance of their health and pushed them towards to save their life at first instead of saving money in general. We are in an uncertain situation that we are in need of to do something to bring back our world to become normal stage. Henceforth, as per the world health organization (WHO) [1] reports, every nation's governments are insisting people to follow certain safety protocols such as wearing masks and maintain social distance from the surrounding people to avoid the infection from Coronavirus [2]. However, it is 80% of people do not follow the same and these habits make again welcoming corona viruses to increase the death rate. Hence, in order to save the lives of the people, a new novel framework for detecting the face mask of an individual from the crowd of people and also to identify people who do not maintain proper distance with others have been proposed. Hence, the results are useful for identifying the people who violate the rules of not wearing the mask and also not following the social distance. The following methods have been implemented using OPENCV and YOLOV3[3]. The results become more useful to the public welfare sectors of government for taking necessary actions for those who violate the government rule

Index Terms – Corona virus, WHO, Infection, Opencv, Yolov3.

I.INTRODUCTION

As the coronavirus (Covid19) pandemic broke out, the public was worried about the spread of the virus without an effective cure. The World Health Organization (WHO) [2] describes Covid 19 as a pandemic due to the increasing number of cases reported worldwide. Many countries impose blockades to limit the outbreak of illness, in which people are said to stay at home during this important time to minimize the risk of infection.

The public health organizations need to make it clear that avoiding close contact with others is the most effective strategy to slow the transmission of Covid19. During the quarantine period, group activities such as travel, meetings, gatherings, seminars, and prayers are prohibited to achieve social distancing. People are encouraged to organize and conduct events as much as possible by phone and email to reduce face-to-face contact. Coronavirus 2019 (COVID-19), the profoundly infectious viral ailment brought about by extreme intense respiratory condition Covid2, significantly affects the world's socioeconomics bringing about more than 3.8million passings around the world, arising as the worldwide wellbeing emergency since the time of the flu pandemic of 1918. After the main instances of this overwhelmingly respiratory viral ailment were first announced in Wuhan, Hubei Province, China, in late December 2019, SARS-CoV-2 quickly spread

across the world in a limited ability to focus time, convincing the World Health Organization (WHO) to pronounce it as a worldwide pandemic on March 11, 2020.

The COVID-19 pandemic has necessitated the development of innovative solutions to enforce safety protocols. This paper focuses on real-time automated monitoring for detecting face masks and social distancing violations in public places [4]. We propose a deep learning-based system that combines modern algorithms with geometric techniques to create a robust model for detection&reporting.

Face Mask Detection

Masks assume an urgent part in safeguarding the soundness of people against respiratory illnesses, as is one of a handful of the precautionary measures accessible for COVID-19 without a trace of vaccination. Our face mask detection system uses a deep learning model based on TensorFlow [7] and OpenCV to identify individuals wearing or not wearing masks in real-time. This system detects faces and evaluates facial points to determine whether a mask is being worn. If the person is wearing a mask, they will be displayed in a green rectangle-box with a safe alert. If not, they will be shown in a red rectangle-box with an alert message.

Social Distancing Detection

The WHO has guaranteed the Covid spreading as a worldwide pandemic in view of the augmentation in the development of Covid patients itemized over the world. In the battle against the Covid, social removal has shown to be a successful measure to hamper the spread of the illness. Our real time social distancing detection system [3] uses the Euclidean distance method to identify individuals maintaining social distancing with a minimum range of 0.80 meters between each other. If individuals maintain social distancing, they will be displayed in a green rectangular-box with a safe alert message. If not, the system will display an alert message in a red rectangular-box.

II.NEED OF THE STUDY

In recent years, the COVID-19 pandemic has led to the need for increased social distancing and the use of face masks to prevent the spread of virus. The consequence of this task shows that persistently catching recordings and distinguish a group who are not wearing facial covering as well as number of individuals who are not following social separating in each casing.

This project adopts YOLOv3 and TensorFlow, employing a novel approach to enhance the robustness of support boundaries and reputation accuracy. By leveraging these technologies, the aim is to achieve superior performance in detecting and maintaining safety protocols, such as social distancing and face mask adherence,may be utilized on real time video observation to screen public areas to find assuming people conveying facial covering and saving secure social separation. Then YOLOv3 designs to explore Real-Time Streaming Protocol (RTSP) video [3] transfers the use of OpenCV. The procedure of this algorithm has profound dominating and customary projective calculation methodologies which now presently not handiest works with to fulfill the genuine time necessities, but moreover proceeds with extreme forecast exactness.

If the man or lady is recognized as now following the Coronavirus insurance pointers, a purple body can be demonstrated so the man or lady the utilization of our product can hold social separation from the group and if they are appeared to be close and not maintaining social distance (threshold distance) then this system alerts the people in the public areas by showing them in a red rectangular box. These alerts can be seen in the output figures 3(a) &3(b) and 4(a) &4(b).

III.METHODOLOGY

In this literature review, we will focus on the application of YOLOv3 (You Only Look Once) for social distancing and face mask detection. YOLOv3 is a real-time object detection system [4] that has gained popularity due to its speed and accuracy. It has been used in various applications, including social distancing and face mask detection.

In the context of social distancing [2], YOLOv3 can be used to detect and count the number of people in a given area. By monitoring the number of people in a specific location, authorities can enforce social distancing guidelines and ensure public safety. For example, in a shopping mall or a public transportation hub, YOLOv3 can be used to monitor the number of people present and alert authorities if the number exceeds a predefined threshold. In the case of face mask detection, YOLOv3 can be used to identify individuals who are not wearing face masks in public spaces.
In the first place, we make a Cascade Classifier object to remove highlights of the face for making sense of before. The way for using XML document to denote face highlights is the boundary here. Subsequent stage is perused a picture with a face on it and convert it into a high contrast picture utilizing commands. After that, we'll be looking into the directions for the picture. This is finished for utilizing detect Multi Scale.

The directions for face represent square shape [7]. The scaling Factor is utilized for diminishing the shape esteem by 5% until the aspect is found. At long last, the face represented that imprinted arranged the window.

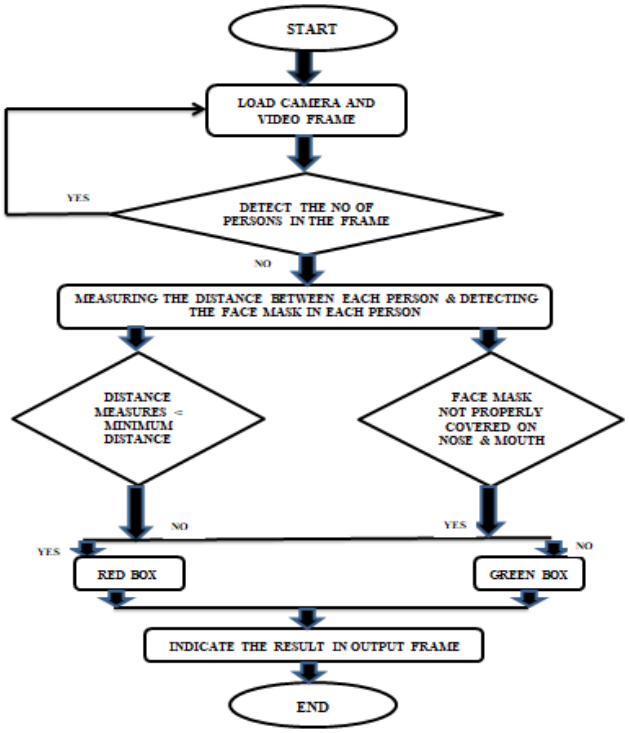

Fig 1: Proposed methodology workflow Process

IV.SYSTEM REQUIREMENTS

Software Requirements

1.Anaconda + Python 3.0
2.Editor : VSCODE
3.Environment : Tensor Flow
4.YOLOV3
5.SSD,MOBILENETV2

Hardware Requirements

1.Camera(CCTV,MOBILE,WEBCAM)
2.RAM:8GB RAM OR Higher
3.Processing Unit: Intel Core i7 or Higher
4.Storage: Fast & Reliable

V.IMPLEMENTATION

The Proposed Methodology work flow process mentioned in above figure has the two ways of descriptions. One is Social Distancing Detection process and the other one is Face Mask detection process. We shall see the detailed process as given below.

Social Distancing Detection Process

According to the stream, the underlying information video outlines are consistently recording the quantity of individuals are recognized as check from the beginning. In the event that any at least two individuals identified, just our framework can gauge a distance between those individuals and referenced about the standard course of social separating keep or not. Assume single individual just distinguished implies our framework doesn't ponder the infringement rule of social removing and furthermore this interaction can have the option to work out distance between two individuals and contrast it and least distance. The outcome demonstrates that number of individuals disregards social distance as displayed in red variety box. Assuming that no infringement happens, it displays in green variety box. [6]

Face Mask Detection Process

According to the stream, the underlying information video outlines are consistently recording the quantity of individuals are recognized as check from the beginning. In the event that a solitary group identified, just our framework can figure out the individual plainly covered nose and mouth by veil or not. In the event that an individual obviously covered implies it shows the outcome into the video outline as green box with message as veil. In the event that not implies it shown the outcome as red box with message as no cover.[7]

Combination Process

According to our technique, the result window shows that the resultant video outline as blended of both social removing and facial covering identification result with red or green variety boxes with messages. Additionally, this approach aims to enhance precision values particularly in the facial covering detection process

Sample code for face mask and social distance violation detection is shown below :-

```
        self.image_label = tk.Label(root)
        self.image_label.pack()

        self.predict_button = tk.Button(root, text="Predict", command=self.predict_image)
        self.predict_button.pack(pady=20)

        self.prediction_label = tk.Label(root, text="")
        self.prediction_label.pack()

        self.image_path = None

    def upload_image(self):
        file_path = filedialog.askopenfilename(filetypes=[("Image files", "*.jpg;*.png")])
        if file_path:
            self.image_path = file_path
            self.show_image(file_path)

    def show_image(self, file_path):
        img = Image.open(file_path)
        img.thumbnail((250, 250))
        img = ImageTk.PhotoImage(img)
        self.image_label.config(image=img)
        self.image_label.image = img

    def predict_image(self):
        if self.image_path:
            self.prediction_label.config(text="Predicting...")
            self.root.update()

            image = cv2.imread(self.image_path)
            predictions = self.predict(image)
            self.display_results(image, predictions)

    def predict(self, image):
        (h, w) = image.shape[:2]
```

TensorFlow execution

```
class YOLO:

    def __init__(self, config, model, labels, size=416, confidence=0.5, threshold=0.3):
        self.confidence = confidence
        self.threshold = threshold
        self.size = size

        self.labels = labels
        self.net = cv2.dnn.readNetFromDarknet(config, model)

    def inference_from_file(self, file):
        mat = cv2.imread(file)
        return self.inference(mat)

    def inference(self, image):
        ih, iw = image.shape[:2]

        ln = self.net.getLayerNames()
        ln = [ln[i - 1] for i in self.net.getUnconnectedOutLayers()]

        blob = cv2.dnn.blobFromImage(image, 1 / 255.0, (self.size, self.size), swapRB=True, crop=False)
        self.net.setInput(blob)
        start = time.time()
        layerOutputs = self.net.forward(ln)
        end = time.time()
        inference_time = end - start

        boxes = []
        confidences = []
        classIDs = []

        for output in layerOutputs:
            # loop over each of the detections
```

Loading yolo and accessing video frames

```
    def upload_image(self):
        file_path = filedialog.askopenfilename(filetypes=[("Image files", "*.jpg;*.png")])
        if file_path:
            self.image_path = file_path
            self.show_image(file_path)

    def show_image(self, file_path):
        img = Image.open(file_path)
        img.thumbnail((250, 250))
        img = ImageTk.PhotoImage(img)
        self.image_label.config(image=img)
        self.image_label.image = img

    def predict_image(self):
        if self.image_path:
            self.prediction_label.config(text="Predicting...")
            self.root.update()

            image = cv2.imread(self.image_path)
            predictions = self.predict(image)
            self.display_results(image, predictions)

    def predict(self, image):
        (h, w) = image.shape[:2]

        blob = cv2.dnn.blobFromImage(image, 1.0, (300, 300), (104.0, 177.0, 123.0))
        net.setInput(blob)
        detections = net.forward()

        predictions = []

        for i in range(0, detections.shape[2]):
            confidence = detections[0, 0, i, 2]
```

Object prediction per locations.

```
imagetest = PhotoImage(file="comimg/facereg.png")
imagetest1 = PhotoImage(file="comimg/social.png")
imagetest2 = PhotoImage(file="comimg/bothfi.png")
imagetest3 = PhotoImage(file="comimg/infrmm.png")
imagetest4 = PhotoImage(file="comimg/automatic.png")
imagetest5 = PhotoImage(file="comimg/quitt.png")

bt1 = Button(window, compound="top", text="SOCIAL DISTANCING FROM IMAGE", image=imagetest1, command=sociald)
bt1.place(x=40, y=280)
# bt1.grid(column=1, row=8)
# bt1.pack()

bt2 = Button(window, compound="top", text="FACE MASK FROM IMAGE", image=imagetest, command=facem)
bt2.place(x=40, y=50)
# bt2.grid(column=1, row=4)

bt3 = Button(window, compound="top", text="LIVE DETECTION", image=imagetest2, command=facesocial)
bt3.place(x=250, y=50)
#bt3.grid(column=4, row=4)

bt4 = Button(window, compound="top", text="ABOUT THE PROJECT", image=imagetest3, command=imform)
bt4.place(x=250, y=280)
#bt4.grid(column=2, row=4)

bt5 = Button(window, compound="top", text="VIDEO SURVEILLANCE", image=imagetest4, command=autom)
bt5.place(x=460, y=50)
# bt5.grid(column=2, row=8)

bt6 = Button(window, compound="top", text="QUIT", image=imagetest5, command=quit)
bt6.place(x=460, y=280)
# bt6.grid(column=3, row=8)

window.resizable(0, 0)
window.protocol("WM_DELETE_WINDOW", disable_event)
window.mainloop()
```

Image location frame continuously identification

VI.RESULT

The course of this task named as friendly removing and facial covering identification process has been effectively introduced and carried out in OpenCV and TensorFlow and YOLOv3 structure in Python Language. We coordinate the profound getting to know YOLOv3 module with the SSD[9] system for a speedy and green profound getting to know reply for genuine time human recognition in video transfers and utilize a three-sided closeness technique to degree distance among people recognized through Surveillance camera in genuine time in open areas and obliges specially crafted realities series to check the facial coverings location of the people worn through the overall population in real time by a switch of getting to be aware to a pre-talented SSD face finder.

This form coordinates among the people in public areas for social distance discovery and facial coverings recognition.

In the proposed device, three stages are followed:

1) Model improvement and preparing
2) Model testing
3) Model execution

Model improvement and preparing

Our system utilizes the exchange acquiring information on technique and will fine-follow the MobileNetV2 [9] model, that might be a profoundly unpracticed shape that should be possible to feature gadgets with compelled processing power The model is continually refined through training and improvement processes, adjusting parameters and optimizing architectures for better accuracy and efficiency. Data augmentation enhances the dataset's diversity, aiding the model's generalization. Regular evaluation identifies and addresses weaknesses, ensuring adaptability to evolving challenges in social distancing and face mask detection.[7,8]

Model testing

The device works in a robotized way and grants to consequently play out the social distance review process. When the model is trained with the custom reality and the pre knowledgeable datasets, we check the precision of the model on the test dataset with the helpful asset of the utilization of showing the jumping box with the choice of the tag and the self-thought score at the highest point of the crate. The proposed model initially recognizes all oldsters with inside the style of cameras and demonstrates a green jumping box round surely anyone extended way from each uncommon after that model directs a test on the ID of social distances kept up with in a public spot, on the off chance that oldsters penetrating social distance standards bounding box changes to red for those people and at the same time facial covering location is done with the use of jumping boxes on the distinguished oldsters face with veil or non-veil named and besides self-thought scores.. The contraption distinguishes the social separating and covers with an accuracy score of 90.9%. [6]

Model execution

The model execution process for social distancing and face mask detection using YOLOv3 involves several key steps. Initially, images or video frames are fed into the YOLOv3 network for analysis. Within this network, object detection algorithms identify individuals and detect the presence of face masks. Subsequently, distance measurement techniques are employed to evaluate the spacing between individuals, utilizing the bounding box coordinates obtained from the detection process. Simultaneously, mask classification algorithms are applied to determine whether individuals are wearing face masks. Through thresholding mechanisms, instances of inadequate social distancing or mask non-compliance are flagged for further scrutiny. Finally, the model outputs results highlighting detected violations and their precise locations, facilitating subsequent actions to ensure compliance with social distancing and mask-wearing protocols.[8,9]

VII.OUTPUT IMAGES

The GUI of social distancing and face mask detection system is shown in the following figure 2:

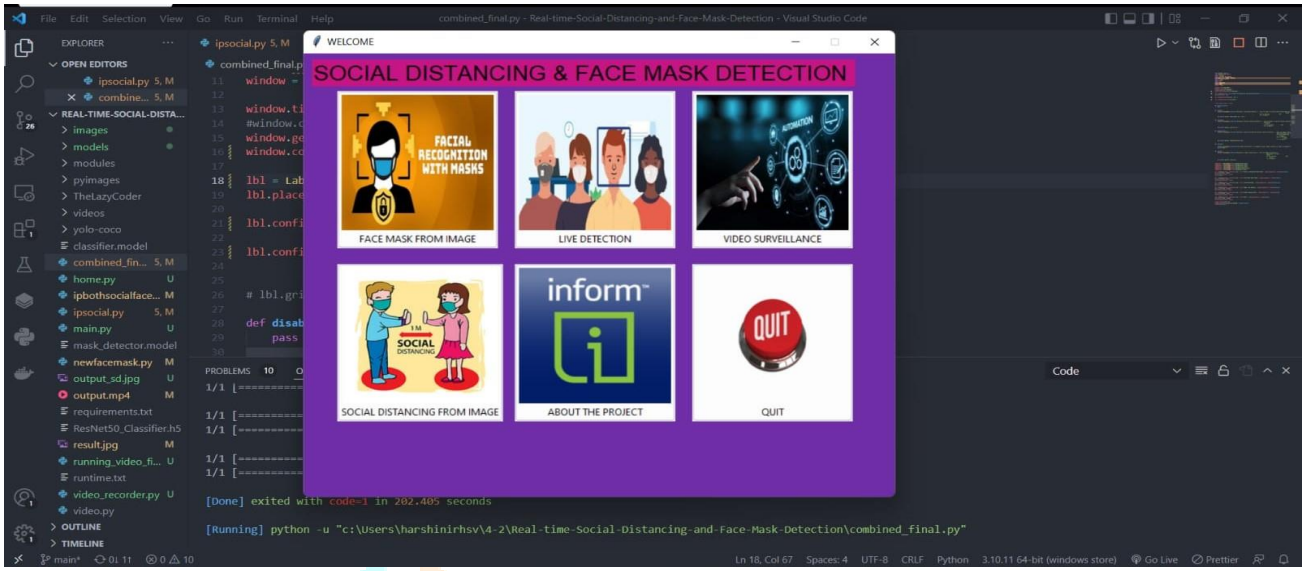

Fig 2: Dashboard UI

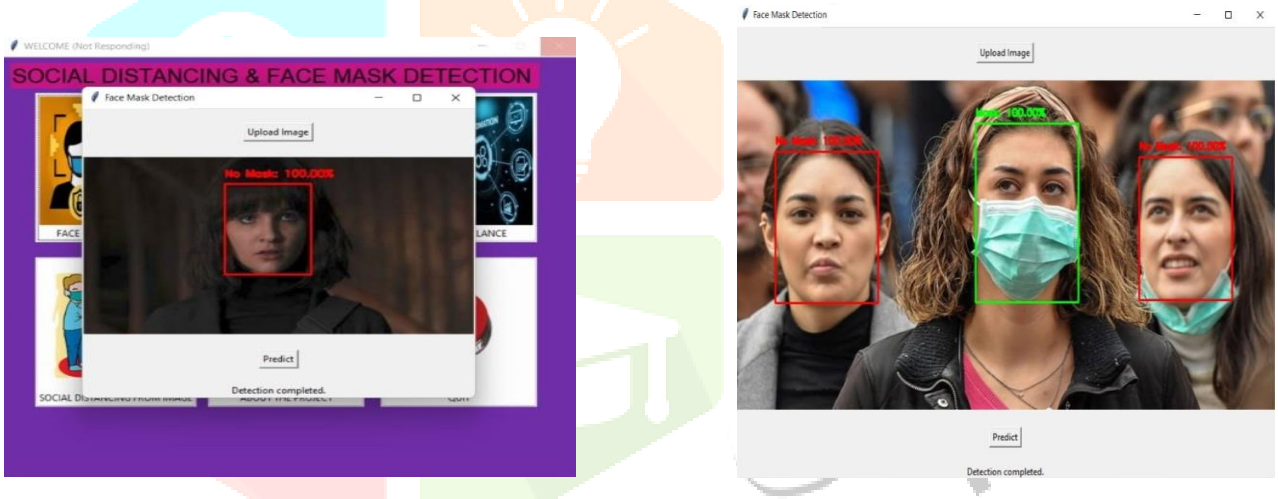

Fig 3(a) & 3(b): Face mask detection

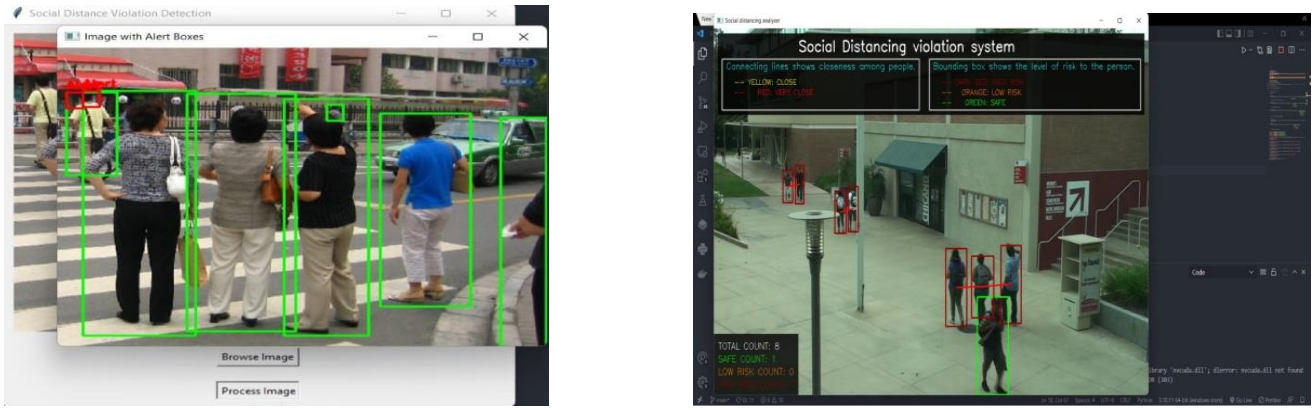

Fig 4(a) & 4(b): Social distance violation detection

VIII.CONCLUSION AND FUTURE SCOPE

In this undertaking, we broaden a shape that can stagger on proper time facial protecting and moreover assist with following the social keeping apart in this pandemic situation.. As indicated above we've utilized numerous libraries and endeavored numerous calculations. Modules like YOLOv3 and TensorFlow had been some of the greatest imperative libraries of our rendition. It will assist to observe and make precise people wellness via automatically following public regions to keep away from the unfurl of the COVID-19 contamination via automatic quality camera in actual time.

Chronic hacking and sniffling is one of the critical signs [2] and side effects of COVID-19 tainting as indicated by WHO ideas and furthermore one of the fundamental courses of illness unfurl to non-aggravated public. Elevated outline temperature is a some other key side effect of COVID-19 contamination, at gift circumstance warm screening is played out the utilization of hand held contactless IR thermometers wherein clinical inspector need to are accessible in close to vicinity with the man or lady need to be screened which makes the clinical analysts powerless to get excited and moreover its almost impractical to hold onto temperature for each and every individual in open areas, the proposed use-case might be prepared with warm cameras essentially based thoroughly screening to explore outline temperature of the people groups in open areas which could transfer some other helping hand to implementation organizations to really address the pandemic. [9]

IX.ACKNOWLEDGEMENT

We would really like to explicit our honest gratitude to all individuals who contributed to this project.. We would like to thank the developers of YOLOV3 and other open sources available used in this study for their invaluable contributions to the field of computer vision. We would like to extend our appreciation to our colleagues and mentors for their support and guidance throughout the project.

REFERENCES

1.Information on covid-19, the infectious disease caused by the most recently discovered coronavirus **https://www.who.int/health-topics/coronavirus**

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

3.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.

4.Punn, N. S., Sonbhadra, S. K., Agarwal, S., & Rai, G. (2020). Monitoring COVID-19 social distancing with person detection and tracking via fine-tuned YOLO v3 and Deepsort techniques. https://arxiv.org/pdf/2005.01385.pdf.

5.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.

6.Redmon, J., & Farhadi, A. (2018). YOLOv3: An Incremental Improvement. https://arxiv.org/pdf/1804.02767.pdf

7.GitHub - thtrieu/darkflow: Translate darknet to tensorflow. Load trained weights, retrain/fine-tune using tensorflow, export constant graph def to mobile devices. (n.d.). Retrieved May 11, 2022, from https://github.com/thtrieu/darkflow

8.Redmon, J., Divvala, S., Girshick, R., & Farhadi, A. (2015). You Only Look Once: Unified, Real-Time Object Detection. https://arxiv.org/pdf/1506.02640.pdf

9.Hou, Y. C., Baharuddin, M. Z., Yussof, S., & Dzulkifly, S. (2020). Social Distancing Detection with Deep Learning Model. 2020 8th International Conference on Information Technology and Multimedia (ICIMU), 334–338. https://doi.org/10.1109/ICIMU49871.2020.9243478