VOL. 23, NO. 3, 2024, 108-114
elektrika.utm.my
ISSN 0128-4428

ELEKTRIKA
Journal of Electrical Engineering

Physical Distance and Face Mask Wearing Surveillance System with Deep Learning

Abdulgader Ramadan Gadah[1*], **Lau Jing Yi**[1], **Suhaila Isaak**[1] and **Sarina Sulaiman**[2]

[1]Faculty of Electrical Engineering, Universiti Teknologi Malaysia, 81310 UTM Skudai, Johor, Malaysia.
[2]Faculty of Computing, Universiti Teknologi Malaysia, 81310 UTM Skudai, Johor, Malaysia.

*Corresponding author: gadah@graduate.utm.my

Abstract: The COVID-19 pandemic has resulted in the world's most critical global health catastrophe. To prevent the spread of COVID-19, people are encouraged to maintain 1 meter of physical distance and wear a face mask. However, many people refuse and forget to practice minimum physical distancing and wear their face masks. Besides, manual monitoring of physical distance and wearing face masks are impractical for a large population with insufficient manpower and resources. Hence, this project introduced a physical distance and face mask-wearing surveillance system utilizing deep learning at R&R Malaysia to ensure the safety of travelers during this COVID-19 pandemic. In this project, the system is implemented using the YOLOv4 algorithm to detect masked, non-masked, and incorrect mask-wearing faces and to calculate the physical distance between people. A total of 3,800 custom datasets were prepared to train the face mask detection model. As a result, this model achieved an average mAP of 95.86%, an F1-score of 0.93, and an average loss of 1.3972. The physical distancing detection model is employed on a pre-trained YOLOv4 algorithm to detect people. The Euclidean distance is calculated between the detected bounding boxes to compute the real distance between people.

Keywords: face mask, physical distancing, deep learning, YOLOv4, object detection

Article History: received 28 August 2024; accepted 7 November 2024; published 30 December 2024

1. INTRODUCTION

The most significant global health crisis in history has been brought on by the COVID-19 pandemic, and its effects are being felt not just in terms of economic costs but also in terms of social and human costs. The COVID-19 virus was initially discovered in Wuhan, China, towards the end of 2019. In January 2020, the outbreak was declared by the World Health Organization (WHO), then in March 2020, the pandemic was declared. Direct transmission of COVID-19 occurs when an individual comes into close contact with an infected person; indirect transmission occurs when an individual is in a contaminated environment. The COVID-19 virus can spread by human or airborne droplets that are 5–10 µm in size, which can lead to respiratory illnesses [1]. Droplet transmission often occurs when a person is within one meter of someone who is exhibiting symptoms of the virus and so has the potential of being infected through mouth, nose, and eyes [1]. It is important to control the spread of the COVID-19 virus due to the increasing number of infected cases and deaths.

The World Health Organization advises people to wear face masks and maintain a physical distance of at least one meter between each other to prevent the spread of disease from person to person [2]. Physical distancing is the process of creating a safe space between an individual and their environment. The amount of interaction between adults over 60 and youngsters under 20 has decreased by 95% and 85%, respectively, by keeping a physical distance [3]. This demonstrates that maintaining a suitable social distance lowers the chance of infection. As respiratory droplets are the primary means of COVID-19 transmission, there is a possibility that these droplets will end up in people's mouths and nostrils. Thus, it is essential to wear face masks to stop the infection from spreading. When worn across the nose and mouth, face masks in this instance function as a barrier to stop the droplets from spreading to other people.

To revive the economy of the nation, the Malaysian government has recently opened both domestic and international borders as well as all economic sectors. Many more individuals go around for employment and tourism due to opening borders and all sectors. Since most of their travels are by highway, this may cause many travelers to stop at the rest and relaxation (R&R) stop in Malaysia. However, because to resource limitations, manual verification of face mask wear and physical separation regulations takes time and might result in human error. It is critically necessary to grasp the optimal physical distance requirements that the public should adhere to in order to stop the spread of the virus. The public can be kept secure by using object detection algorithms to automatically and reliably detect social distance and face mask use. These algorithms check to see if face masks are being used and whether the distance is being broken.

It is crucial to understand and enforce the optimal physical distance requirements to curb the spread of the virus. By employing object detection algorithms, we can automatically and accurately monitor social distancing and face mask compliance. These algorithms can effectively check for mask usage and ensure that social distance is maintained, thereby promoting health and safety practices in the post-COVID-19 phase after 2022. Moreover, public health organizations may still recommend mask-wearing and distancing in certain situations, especially in crowded or indoor settings. Some individuals may also choose to wear masks and maintain distance as a personal comfort measure. By continuing these practices, we not only mitigate the spread of other illnesses, such as flu or RSV, but we also prepare for any potential future outbreaks.

One Dahua Technology product on the market can screen the temperature and identify single masked faces. It was approximately RM 3278 in price. In addition, there is a product called the Social Distancing Alarm Contact Tracing Wearable K59 that regulates the distance between individuals via Bluetooth and RFID. Nevertheless, only one detecting module is integrated into both models. Furthermore, they cannot be reconfigured, making them less adaptable to varying locations and specifications. The system will be far more effective at detecting big populations if it combines two detection modules into a single product. Therefore, deep learning has been suggested in this project to create a high accuracy detection system that aids in the real-time, low-effort monitoring of face mask wear and physical distance. Machine learning systems offer distinct advantages over RFID and Bluetooth technologies. For instance, implementing a machine learning-based solution can lead to significant cost reductions, as it eliminates the need for extensive hardware infrastructure typically associated with RFID or Bluetooth setups. Moreover, the convenience of using existing camera systems, such as CCTV or mobile phones helps the deployment this technology with minimal additional investment. Furthermore, machine learning systems provide greater flexibility and scalability, as the systems can be easily adapted to monitor not just face mask compliance and social distancing but also other behaviors and conditions, making them more versatile in various commercial contexts. This adaptability and cost-effectiveness position machine learning systems as a superior alternative to traditional proximity-based technologies for compliance monitoring and public safety applications.

Artificial intelligence (AI) and machine learning techniques such as deep learning mimic the functioning of the human brain to tackle typical learning problems. Representation learning, which is used in deep learning, enables a system to be fed raw data and determine what representations are needed for classification or detection [4]. The multidimensional and inherent relationship in data is generally found using many layers of representation in deep learning models, which are inspired by biological neurons. Multi-layered neural networks, which connect one or more hidden layers to create a network capable of learning complex structures with a high degree of abstraction, are the building blocks of deep learning [5].

Convolutional neural networks, or CNNs, are among the most widely used deep learning techniques, particularly in the fields of computer vision and image analysis. CNNs are feed-forward multilayered networks in which each layer is subjected to multiple transformations [6]. Typically, CNN structures are composed of alternating layers of convolution and pooling, with fully connected (FC) layers at the end. Convolutional layers and pooling layers perform feature extraction. The CNN's core module, the convolutional layer, assists in extracting high-level features from the input images; by sliding the input image over the filter, a feature map is created [5]. This is the result of the convolutional process, which offers details about the picture, such as its edges, curves, and straight lines. The output of each convolution process is fed into a non-linear processing unit, which replaces any negative values in the feature maps with zeros while preserving the positive values [5]. At the network end, the categorization process is managed by the full connected (FC) layer. This FC layer will perform the classification task by using the input from the feature extraction stages to compute the outcome of the previous layers [6].

Recent advancements in artificial intelligence (AI) and deep learning have greatly enhanced monitoring systems. Deep learning algorithms, especially convolutional neural networks (CNNs), excel in image analysis and object detection, utilizing multiple layers to extract high-level features from images through convolution and pooling. The YOLO (You Only Look Once) algorithm is notable for its speed and accuracy in real-time object detection, with YOLOv4 representing the latest iteration [7]. YOLOv4 improves upon YOLOv3 by employing CSPDarknet53 as its backbone, which enhances feature extraction and reduces computational complexity.

YOLO processes an entire image in a single evaluation, expediting object recognition. This research focuses on developing face mask detection and physical distancing systems using YOLOv4. The architecture of YOLOv4 consists of four distinct blocks: backbone, neck, dense prediction (one-stage), and sparse prediction (two-stage) in Figure 1. While YOLOv3 uses Darknet53, YOLOv4 employs CSPDarknet53, which divides the base layer of the feature map using a cross-stage hierarchy, reducing network characteristics and addressing the vanishing gradient problem [8, 9]. Integrating a YOLOv4-based surveillance system with public health recommendations can enhance safety and promote adherence to health guidelines, ensuring vigilance in protecting communities.

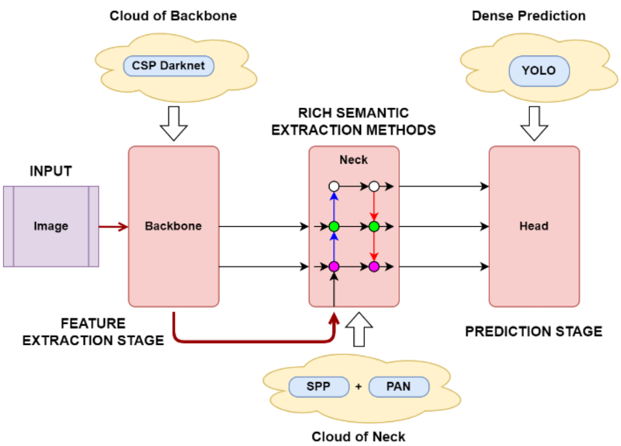

Figure 1. YOLOv4 architecture [5]

2. METHODOLOGY

In this study, we enhance the YOLOv4 algorithm for real-time face mask and social distancing detection. While YOLOv4 is a powerful object detection framework, we utilized several modifications to increase its effectiveness in public health monitoring. Firstly, the YOLOv4 architecture is optimized by adjusting the input layer to better differentiate between masked and unmasked faces. The transfer learning has been employed to fine-tune the model on a specialized dataset, improving accuracy in various environmental conditions.

Additionally, the contextual awareness was integrated into the detection process, allowing the system to adapt to different settings, such as indoors versus outdoors. Our approach includes the use of ensemble methods, combining outputs from multiple models to enhance detection reliability. To ensure efficient deployment on mobile devices, we had implemented model optimization techniques such as pruning and quantization, reducing computational load while maintaining accuracy.

The process of developing the physical distance and face mask-wearing surveillance system is divided into three parts: face mask detection model, physical distancing detection model, and implementation of both detection models on mobile devices as illustrated in Figure 2. The figure illustrating the real-time video streaming capture process outlines the workflow for monitoring face mask-wearing and social distancing using CCTV or mobile phone cameras. Initially, these devices capture continuous video streams of the monitored environment, providing essential real-time data for effective surveillance. Once the video is captured, it is processed using Python, which implements the YOLOv4 algorithm. The video feed is divided into individual frames, each representing a snapshot of the scene. This frame extraction is crucial for detailed analysis.

Each frame is analyzed by the YOLOv4 model, trained to detect objects such as faces and masks. The model assesses whether individuals are wearing masks and evaluates their proximity to determine compliance with social distancing guidelines. YOLOv4 assigns a confidence score to each detection, with only those surpassing a specified threshold being considered valid.

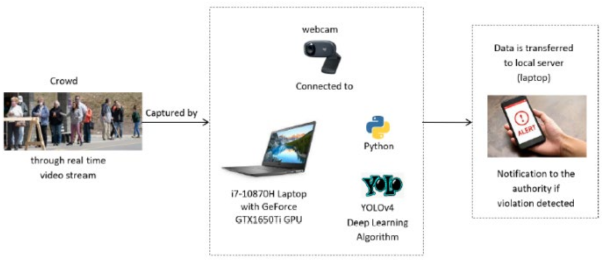

Figure 2. The block diagram of developing the physical distance and face mask-wearing surveillance system

When violations are identified, such as individuals not wearing masks or being too close together, then alerts are generated, detailing the location and time of the incident. These alerts are sent to a centralized surveillance center through a secure network connection, ensuring monitoring personnel receive timely information.

2.1 Face Mask Detection Model

The face mask detection model is developed in three stages: data preparation, custom object detection model training and performance evaluation. To train the face mask detection model, the images are collected from open-source websites, such as Joseph Nelson Roboflow [10], Prajnasb Github [11], X-zhangyang Github [12] and Kaggle [13]. A total of 3800 images of masked faces, non-masked faces, and incorrect mask-wearing faces are collected as custom datasets for training. The training dataset consists of 90% of the custom dataset while the testing dataset contains the rest of 10% to avoid overfitting of the model. Due to the laptop's limitation, the dataset's image size is limited to 416 x 416 pixels resolution to fulfil the minimum input criteria of Darknet architecture [14]. After the datasets are prepared, the faces with or without masks or incorrect mask-wearing in every image are annotated by drawing the bounding box on the desired area using an open-source labeling tool called LabelImg.

The output of the annotated file is a text file (.txt) that follows the annotation format of YOLOv4. This annotation format is used by the YOLOv4 object detection algorithm to provide the necessary information for training and inference. Each line in the text file contains the following information:

$$.txt =< \text{object\_class} >< \text{x} >< \text{y} >< \text{width} >< \text{height} > \quad (1)$$

where; the <object_class> represents the class or category of the detected object. The <x> and <y> are the coordinates of the center of the bounding box. The dimensions of the bounding box are the <width> and <height>.

The YOLOv4 model is trained using the raw photos and the accompanying annotated text file once the training datasets have been fully annotated. A 2.20GHz Intel Core i7 CPU and an Nvidia GeForce GTX 1650 Ti GPU with 4GB of RAM are used to train this customized YOLOv4 model. To train a custom detection model, the yolov4-custom, .cfg custom configuration file's hyperparameters must be adjusted to match the training of the custom model. According to Bochkovskiy, A. et al., there should be a minimum of 6000 iterations (max_batches = classes * 2000) and a maximum of the number of training pictures [15]. As a result, the training is set to 6000 iterations with a batch size of 64 and a subdivision size of 64 under the custom configuration file. For each cycle, 64 photos will be loaded if the batch size is 64. With 64 subdivisions, 1 image (64/64=1) will be transmitted to the GPU for processing for each mini-batch. In addition, the learning rate, momentum, and decay parameters are maintained at their respective default values. A breakdown of the model's hyperparameter setup can be seen in Table 1, and established through empirical testing, systematic tuning, and domain expertise to enhance model performance. The configuration file is trained alongside YOLOv4 pre-trained weights (yolov4.conv.137) when it is set.

The performance of the developed training model must be evaluated before exporting it as the detection model. If the bounding box is detected correctly on the masked faces, non-masked faces, and incorrect mask-wearing faces in the images or videos, the detection model can be used in the further stages. Determining hyperparameter values such as learning rate, filters, steps, and classes is essential for training effective deep learning models like YOLOv4 [16]. The learning rate, which influences how much the model weights are updated during training, is often found through experimentation. Techniques like grid search help identify the optimal learning rate, while learning rate schedulers adjust the rate dynamically during training, and cyclical learning rates vary it between minimum and maximum values to avoid local minima [17, 18].

The number of filters in convolutional layers is typically based on the task's complexity, starting with a baseline and adjusting according to validation performance. Generally, deeper layers have more filters to capture intricate features, as seen in YOLOv4. The number of training steps or epochs is determined using methods like early stopping, where training ceases when validation loss starts to worsen, thus preventing overfitting. Cross-validation techniques can also help identify the optimal number of steps for better generalization. The number of classes is decided by the application and dataset. For instance, in a face mask detection system, classes may include "masked," "unmasked," and "partially masked."

Table 1. Hyperparameter configuration of the model

Hyperparameters	Configuration
Batch	64
Subdivision	64
Network Size	416 x 416
Max_batches	6000
Steps	4800, 5400
Learning rate	0.001
Momentum	0.949
Decay	0.0005
Filters	24
Class	3

One of the performance evaluation approaches is the confidence score. If the confidence score is greater than the pre-defined threshold value, the bounding box will be created. To enhance the detection accuracy of the model, the threshold value should be modified experimentally to obtain the most appropriate value. Other than confidence score, intersection over union (IOU) is also a metric to evaluate the detection accuracy. A higher ratio of overlapping between the predicted bounding box and the ground truth box can be obtained with a higher IOU value to promise the prediction accuracy. The performance of the detection model can also be evaluated in terms of precision, recall, F1-score and mean Average Precision (mAP). Precision is the ability of the model to predict relevant object. It is a measure of efficiency. Recall is a measure of effectiveness, which means the model is able to find all the ground truth objects. F1-score is a measure that balances between precision and recall. A high value of F1-score shows that both precision and recall are high. Furthermore, average precision (AP) is the precision averages across all recall values between 0 and 1 at various IOU thresholds whereas the mAP is the average AP over multiple IOU. A high value of precision, recall, F1-score and mAP (greater than 0.9) can produce a good object detection model.

2.2 Physical Distancing Detection Model

The social distancing detection model utilized YOLOv4 algorithm to detect people in the image or frame. It also employed computer vision and Python to detect the distance between people to ensure their safety. The flow of the social distancing detection model is shown in Figure 3. Before feeding a video frame into the YOLOv4 pre-trained for physical distancing detection, the model first captures the first frame of the video to allow user selecting four points on the frame. These selected four points will then be transformed into a 2D bird's view by applying Perspective Transformation. After the transformation, the bird's view video is ready to feed into YOLOv4 pre-trained model frame by frame for detection.

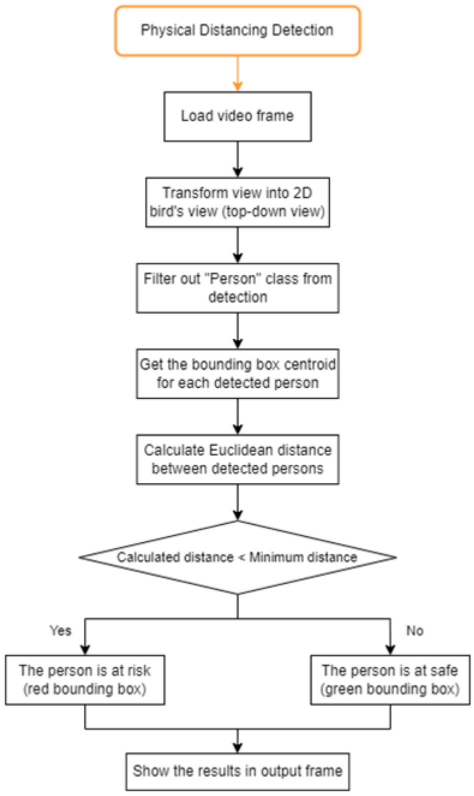

Figure 3. Flow chart of physical distancing detection model

The model starts the detection and only filters out the

"Person" class by discarding other classes since the model only requires person detection. After the person detection, it will return the bounding box coordinates with centroid value (x_center, y_center), width and height. The orientation in the bird's view transformation is determined based on the centroid of each person in the input frame. The centroid of bounding box for each person detection is then used to calculate the distance between two persons. The Euclidean distance is used to calculate the distance between the centroid of two detected bounding boxes as depicted in Equation (2)

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

where D is the distance between two bounding boxes, $(x_{1,}y_1)$ is the coordinate of bounding box 1, and (x_2, y_2) is the coordinate of bounding box 2. The distance, D is then compared with the pre-defined minimum threshold value to check whether the distance is violated or not. If the distance is less than the threshold value which means that two people violate the minimum social distance, the bounding box information will be stored in a violation list and the violated bounding box will be marked in red with a red line connecting it to the compared bounding boxes. On the other hand, if the distance is more than the threshold value, the detected bounding box will be marked in green, indicating they are performing physical distancing.

3. RESULTS AND DISCUSSIONS

This work introduces several novel contributions in comparison to existing face mask detection systems that utilize deep learning, particularly those using YOLOv4 and combined detection modules. A significant advancement in this work is the enhancement of the YOLOv4 architecture through customized input preprocessing techniques and modifications to the network structure. These improvements enable the model to more accurately differentiate between masked and unmasked faces in a variety of environments. Additionally, we uniquely integrated face mask detection with physical distancing assessment within a single framework, enabling simultaneous monitoring of both health compliance aspects. This dual approach streamlines the detection process, unlike many traditional systems that focus on one aspect at a time.

Our method also incorporates context-aware detection, adapting thresholds based on environmental conditions such as lighting and crowd density, which enhances reliability. Furthermore, an ensemble learning strategy has been employed that combines outputs from multiple YOLOv4 models, achieving greater accuracy and robustness against false positives. The system is also optimized for mobile deployment, allowing for efficient operation on mobile devices, and features a real-time notification system that alerts a centralized surveillance center to compliance violations, facilitating timely interventions.

3.1 Face Mask Detection Model Performances

Figure 4 shows the plotted graph of training loss and validation mean average precision (mAP). The model yielded an average loss of 1.3972 and an average mAP of 94.6%. The result that was obtained indicates that the model has trained well and can effectively conduct object detection, as illustrated in Figure 5. Intersection over Union (IoU) and detection threshold are two regulating elements that must be carefully chosen to guarantee optimal model performance. Tables 2 and 3 list the mAP values and F1-score of the detection model for various IoU and detection thresholds.

Based on Table 2, with a constant detection threshold, the increase of IoU threshold decreases the mAP value, indicating that the model has a lower ratio of correctly detected masked faces to the total number of detections. Therefore, lower IoU threshold is preferable. In addition, any increase in the threshold value will improve detection accuracy when the IoU threshold remains unchanged. For higher detection precision of the model, low IoU threshold and high detection threshold is required. Since F1-score is a measure that balances precision and recall, a higher F1-score will produce better detection results. Noticing that the F1-score is almost the same under different thresholds with constant IoU threshold, higher threshold values are preferable. Under the same threshold value, the F1-score decreases with increasing IoU threshold. Thus, a lower IoU threshold with a higher detection threshold is favorable.

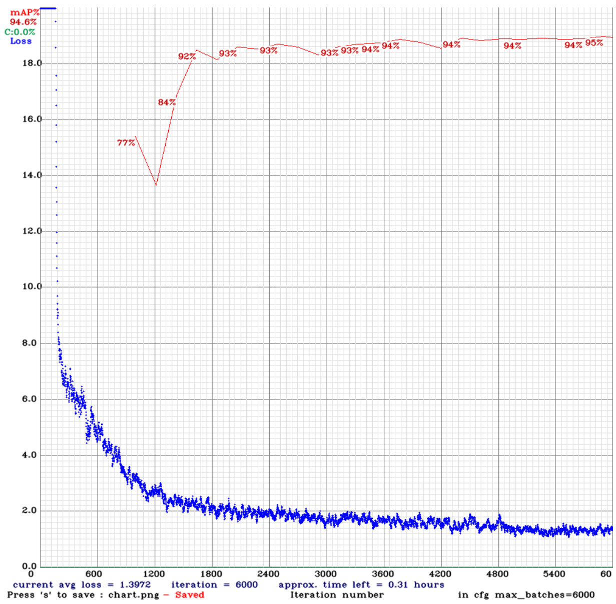

Figure 4. Training loss and validation mAP graph of the detection model

Table 2. mAP under different IoU and detection threshold

IoU	Detection Threshold	mAP (%)
0.20	0.20 - 0.35	95.87
0.25	0.20 - 0.35	95.86
0.30	0.20 - 0.35	95.72
0.35	0.20 - 0.35	95.64

0.40	0.20 - 0.35	95.25
0.50	0.20 - 0.35	94.83
0.75	0.20 - 0.35	73.62

Table 3. F1-Score under different IoU and detection threshold

IoU	Detection Threshold	mAP (%)
0.20	0.20 - 0.30	92
	0.35	93
0.25	0.20 - 0.30	92
	0.35	93
0.30	0.20 - 0.30	92
	0.35	93
0.35	0.20 - 0.35	92
0.4	0.20 - 0.35	92
0.5	0.20 - 0.35	91
0.75	0.20	69
	0.25-0.30	70
	0.35	71

Figure 5. Object detection on images

3.2 Physical Distancing Detection Model Performance

To test the social distancing detection model, pre-filmed videos captured are fed into the model as the input. Before feeding the model into YOLOv4 pre-trained model, the first frame of the input video is captured and four points that will be used for transforming the frame into 2D bird's view are selected. These four points are connected with red line forming a closed area, which is the interested region for person detection as shown in Figure 6. Meanwhile, Figure 7 illustrates the video frame that has been transformed into a two-dimensional (2D) bird's view. The bird's view video is fed into the YOLOv4 pre-trained model frame by frame for detection. Since the model only detects the "Person" class, thus the YOLOv4 pre-trained model only recognizes human-like objects in the frame. As illustrated in Figure 8, the model produces a good detection output that the person in the video frame is detected with green or red bounding box. The bounding boxes represented in green indicates the person is maintaining a physical distance as the distance is within the acceptable threshold value. The people with red bounding boxes indicate that they violate the pre-defined threshold value. On the upper right of the frame, the number of people violating the physical distance is shown. The threshold value might not be the same for every input, hence it needs to be adjusted accordingly to fit the social distance standard of 1 meter.

Figure 6. Frame with selected four points line

Figure 7. 2D bird's view

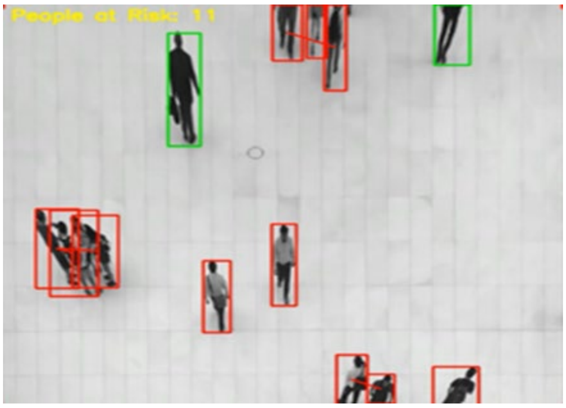

Figure 8. Output of physical distancing detection model using bird's view video frame

4. CONCLUSION

A face mask detection model and a social distancing detection model utilizing deep learning have been developed to identify masked faces, non-masked faces, incorrect mask-wearing faces and social distance between people. For face mask detection model utilizing YOLOv4, the model achieved average mAP of 95.86 %, F1-score of 0.93 and average loss of 1.3972. For social distancing detection model, it is tested on a pre-trained YOLOv4 model by calculating the Euclidean distance between the detected person bounding boxes on the bird's view video frame. In short, this system provides an efficient solution to monitor face mask wearing and physical distancing practices in public areas such as R&R Malaysia instead of monitoring manually.

ACKNOWLEDGMENT

Universiti Teknologi Malaysia supports this work under UTM-TDR25.4(T1): VOT Q.J130000.3551.06G93. The authors also would like to express sincere appreciation to Dr. Suhaila Isaak and Dr. Sarina Sulaiman for their guidance and assistance in realizing the project.

REFERENCES

[1] M. A. Kohanski, L. J. Lo, and M. S. Waring, "Review of indoor aerosol generation, transport, and control in the context of Covid-19," *International Forum of Allergy & Rhinology*, vol. 10, no. 10, pp. 1173–1179, 2020.

[2] World Health Organization, [Online] Available: *https://www.who.int/emergencies/diseases/novel-coronavirus-2019/*. [Accessed 20 November 2021].

[3] L. Matrajt and T. Leung, "Evaluating the Effectiveness of Social Distancing Interventions to Delay or Flatten the Epidemic Curve of Coronavirus Disease," *Emerging Infectious Diseases*, vol. 26, no. 8, pp. 1740-1748, 2020.

[4] Y. LeCun, Y. Bengio, and G. Hinton, "Deep Learning," *Nature*, vol. 521, no. 7553, pp. 436-444, 2015.

[5] E. Mbunge, S. Simelane, S. G. Fashoto, B. Akinnuwesi, and A. S. Metfula, "Application of deep learning and machine learning models to detect COVID-19 face masks - A Review," *Sustainable Operations and Computers*, vol. 2, pp. 235–245, 2021.

[6] A. Khan, A. Sohail, U. Zahoora, and A. S. Qureshi, "A survey of the recent architectures of deep convolutional Neural Networks," *Artificial Intelligence Review*, vol. 53, no. 8, pp. 5455–5516, 2020.

[7] 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)*, pp. 779-788, 2016.

[8] A. Bochkovskiy, C.-Y. Wang, and H.-Y. M. Liao, "Yolov4: Optimal Speed and accuracy of object detection," *Cornell University Library*, [Online] Available: *https://arxiv.org/abs/2004.10934* [Accessed: 10 December 2021].

[9] K. Andhy Panca Saputra, "Waste Object Detection and Classification using Deep Learning Algorithm: Yolov4 and Yolov4-Tiny," *Turkish Journal of Computer and Mathematics Education (TURCOMAT),* vol. 12, no. 6, pp. 5583-5595, 2021.

[10] J. Nelson, "Mask wearing object detection dataset," *Roboflow*, November 2020. [Online]. Available: https://public.roboflow.com/object-detection/mask-wearing. [Accessed: 20 November 2021].

[11] Prajnasb, "Observations/mask_classifier at master · prajnasb/observations," *GitHub*, [Online]. Available: https://github.com/prajnasb/observations/tree/master/mask_classifier. [Accessed: 20 November 2021].

[12] X-zhangyang, "X-zhangyang/real-world-masked-face-dataset: Real-world masked face dataset," *GitHub*, January 2021 [Online]. Available: https://github.com/X-zhangyang/Real-World-Masked-Face-Dataset. [Accessed: 20 November 2021].

[13] Larxel, "Face mask detection," *Kaggle*, [Online]. Available: https://www.kaggle.com/datasets/andrewmvd/face-mask-detection. [Accessed: 03 March 2022].

[14] Y. K. Yik, N. E. Alias, Y. Yusof, and S. Isaak, "A real-time pothole detection based on Deep Learning Approach," *Journal of Physics: Conference Series*, vol. 1828, no. 1, p. 012001, 2021.

[15] A. Bochkovskiy, "Alexeyab/Darknet: Yolov4 / scaled-yolov4 / yolo - neural networks for object detection (Windows and Linux version of Darknet)," *GitHub*. [Online].Available: https://github.com/AlexeyAB/darknet#how-to-train-to-detect-your-custom-objects. [Accessed: 5 November 2021].

[16] L. Yang and A. Shami, "On hyperparameter optimization of machine learning algorithms: Theory and practice," Neurocomputing, vol. 415, pp. 295–316, Nov. 2020.

[17] K. Leyton-Brown, H. Hoos, and F. Hutter, "An Efficient Approach for Assessing Hyperparameter Importance," in *Proceedings of the 31st International Conference on Machine Learning*, 2014, pp. 754–762.

[18] K. Greff et al., "LSTM: A Search Space Odyssey," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 28, no. 10, pp. 2222–2232, Oct. 2017.