A Deep Learning-Based Tool for Face Mask Detection and Body Temperature Measurement

Yuxi Zhang, Ali Al-Ataby and Fawzi Al-Naima

***Abstract*—Due to the COVID-19 pandemic outbreak, wearing a mask and ensuring normal body temperature in overcrowded areas such as workplaces has become obligatory. To save costs of manual supervision and reduce human contact for safety concerns, a deep learning -based tool for automatic mask detection and temperature measurement at the entrance of workplaces was developed in this paper. Using Python, image/video processing techniques related to face and object detection are used to process image input from a webcam. A deep learning algorithm called MobileNetV2 was used to build the face mask detector model. Moreover, a non-contact thermal sensor, which is the MLX90614, along with Arduino were employed to measure body temperature. The results of both mask detection and temperature measurement are displayed correctly on a Graphical User Interface (GUI). Besides, an additional function related to the Internet of Things (IoT) was implemented, which sends high temperature alerts to smartphones. It has been verified that the model can achieve an accuracy of about 98%. The developed system experiences a limitation when other objects are used to cover the mouth and nose in that they may still be classified as masks. However, compared to the mask detection systems available commercially, it can at least provide correct detection results when using the hand to pretend wearing a mask.**

***Index Terms*—COVID-19, face mask detection, temperature measurement, deep learning, IoT**

I. INTRODUCTION

COVID-19 is an infectious disease that infects the respiratory system caused by the SARS-CoV-2 virus [1]. Since the outbreak of the pandemic in December 2019, it has infected over 500 million people and caused around 6 million deaths globally [2]. The World Health Organization (WHO) suggested that wearing a mask in a public area is one of the most effective means to control the transmission of COVID-19 [3]. Authorities worldwide, especially the Chinese government, require masks to be worn indoors, such as in workplaces. Besides, identifying the disease at an early stage is also essential for controlling the spread of the pandemic [3]. High body temperature is a common symptom of COVID-19 patients. Thus, to search for any sign of infection, body temperature should be examined. In other words, to ensure that individuals are adhering to the norms prescribed by the health officials, supervision is required. However, at the entrance of companies, manual supervision of mask-wearing and body temperatures may result in high labor costs. Moreover, the safe social distance between the supervisory personnel and the employee is hard to maintain, increasing the probability of infecting by the virus. Therefore, a tool with a graphical user interface (GUI) that can automatically perform real-time mask detection and non-contact temperature measurement for staff in workplaces was built. To incorporate the concept of IoT, an additional notification-sending function was added to the original system.

II. SYSTEM FRAMEWORK

Figure 1 shows the framework of the proposed face mask detection and temperature measurement system. For mask detection, the real-time video stream with human faces collected by a webcam is fed into the system. Several image processing techniques are applied to each frame of the video stream. After preprocessing, the frame is fed into a prepared face detector to extract the Region of Interest (RoI) [4], which refers to the human face in this project. The actual mask detection is performed by a mask detector model that was developed based on a deep learning algorithm called MobileNetV2. The model detects the mask features on each RoI to classify whether there is a mask or not. Finally, a label with the probability of classification result is displayed on the GUI. For non-contact temperature measurement, the thermal sensor MLX90614 with Arduino UNO R3 board are used. To avoid interference with the continuous image data reading process, multithreading is required. Thus, the temperature data read by a serial port is fed into another thread for processing. The GUI shows temperature measurement, and if the temperature is higher than a predefined threshold (e.g. 37^{o} C), staff is informed that entering is not allowed. Besides, by using an external application called IFTTT [5], a high- temperature notification alert is sent to an administrative staff smartphone.

This paper is organized as follows: next section provides literature review of related work and methods. Section IV provides details about the methodology followed in this research and the proposed design. Section V provides system evaluation and results. Finally, sections VI and VII provide discussions and conclusions, respectively.

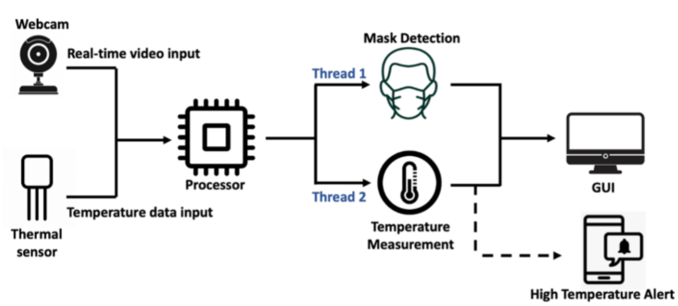

Fig. 1. The framework of the proposed system

III. LITERATURE REVIEW

A. Object Detection

Face mask detection refers to detecting whether a person is wearing a mask or not, including localizing the face first

[6]. The problem is a branch of general object detection in the computer vision field, aiming to detect the class of the object. With the development of deep learning, deep learning-based detectors have demonstrated robustness and high feature extraction capability. There are two categories of detectors; which are one-stage and two-stage detectors.

One-stage detectors use merely a single neural network for object detection. To achieve this, the ratio of width to height of objects should be predefined in anchor boxes. For example, YOLO divides an image into different cells and then attempts to match the anchor boxes to objects for each cell [7]. However, this approach shows undesirable performance for small objects. Thus, multi-scale detection such as Single Shot Multibox Detector (SSD) has been developed, which detects objects on several feature maps to allow detection in various sizes [8].

On the other hand, the two-stage detectors generate region proposals at the first stage and then fine-tune the proposals at the second stage. R. Girshick et al suggested a Region-based Convolutional Neural Network (R-CNN) in 2014 [9]. Later on, a fast version of R-CNN was proposed in 2015 to solve the problem of the original R-CNN [4]. It introduces a RoI pooling layer to feed all candidate regions at once. In this project, the developed mask detector model is a two-stage detector that realizes object classification over the RoI.

B. Convolutional Neural Networks

Convolutional Neural Network (CNN) is one of the most representative neural networks. It is employed for a wide range of applications related to pattern recognition, especially image classification [10]. However, as the network becomes deeper, it is found that excessive layers added into the network can cause the accuracy to degrade promptly after it becomes saturated. To solve this degradation problem, K. He et al proposed the Residual Network (ResNet) [11]. By using a residual block, several convolutional layers can be skipped, and inputs can be directly added to the outputs of the stacked layers. This residual configuration eases the learning process and simplifies the training of a deep network. Since problems such as object detection are usually deployed on mobile or embedded devices, where the computational resources are very limited, Mobile Network (MobileNet) [12] was proposed. It uses depth-wise convolution instead of the conventional convolution to extract features, which helps in reducing the computation cost to a great extent. It combines the merits of ResNet by incorporating the idea of residual learning [13].

C. Related Work on Face Mask Detection

In the traditional object detection, Deformable Part Model (DPM) has been adopted in building a face mask detector to help model the structure and orientations of faces first. A DPM-based face mask detector was proposed and used about 30,000 faces to achieve an exceptional accuracy of around 97% [14]. Even though it demonstrates majestic precisions, the computational cost can be intolerable due to the use of DPM. After the advent of the COVID-19 by the end of 2019, a large number of research work using pre-trained CNN models have been conducted on face mask detection. A face mask detection solution using transfer learning of InceptionNetV3 was proposed, showing a quite outstanding accuracy of 99% [15]. Considering the importance of applying mask detection to embedded devices, another effective model called SSDMNV2 was developed later [16]. It uses SSD as the face detector and MobileNetV2 as the framework for the classifier, which also presents a competitive accuracy of 93%. There is an example of using both the MobileNet and ResNet as the backbone to build a mask detector called RetinaFaceMask, which was proposed by M. Jiang et al [17]. Inspired by the above related work, in this project, the pre-trained CNN model MobileNetV2 was used as the backbone of the proposed mask detector model.

IV. METHODOLOGY AND DESIGN

Figure 2 demonstrates how the real-time mask detection function is achieved step-by-step. There are basically four stages in the procedure, which are preparing the dataset for training and testing, preprocessing the input data, building and training the proposed mask detector, and performing mask detection on each frame of the webcam video stream to implement real-time detection. The details of the real-time mask detection process are provided below.

A. Dataset Preparing and Preprocessing

The dataset consists of two categories of images, which are “with mask” and “without a mask”. Most of the input images come from the open-source platform available online [18], whereas a small number of them are generated by the Generative Adversarial Network (GAN) algorithm (Simply type “This person does not exist” in Google). To allow the proposed face mask detector also determine that covering the face by hand is to be considered as without a mask, several images with a hand covering the mouth were added to the “without a mask” images category. At a late stage of the project, a limitation was found, which is that other objects instead of masks can still be classified as masks. As a solution to overcome this limitation, 60 more confusing images were included. These are images with objects that are probably used to pretend to be a mask in reality such as a book or a mobile phone. To sum up, there are 710 images in the “with mask” category while 770 images are in the “without a mask” category.

The preprocessing of the input images is performed by TensorFlow/Keras. The primary image preprocessing includes converting the image into arrays and using the integral preprocessing technique that is particular for MobileNetV2. Besides, image data augmentation was also utilized to increase the diversity of the input data without collecting new data. By using this idea, sufficient data for training is going to be available, and, hence, the performance and accuracy of the deep learning model is improved.

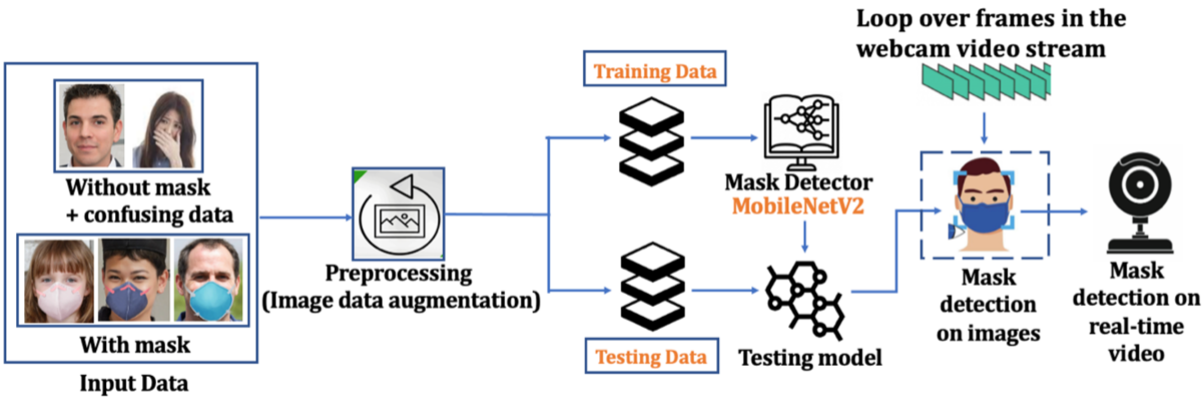

Fig. 2. Flow diagram of real-time mask detection

B. Face Mask Detector Development and Training

After preprocessing, the input image dataset was partitioned in such a way that 80% was used as the training data, and 20% was used as the testing data. For building the proposed mask detector model, MobileNetV2 was loaded as the base model while the head fully-connected layers were left off to be modified later. After reconstructing the head, the new head with fully-connected layers were added to the base model. This is the actual model that was trained. However, all the layers in the base model were set to be untrainable and merely the weights of the new head model were updated in the training process. The modification of the head of the model and the freezing of the base model layers are the concepts of fine-tuning. Details about fine-tuning are shown in a later section.

The Adaptive Moment Estimation (Adam) Optimizer with a learning rate of 0.0001 was used, which is the most common algorithm in machine learning. The Binary Cross-Entropy is used as the loss function, which is often adopted on binary classification problems such as this mask detection task. The loss and accuracy during training to evaluate the model are presented in the results section. After the training process has finished, the trained mask classifier model was saved and made ready to be loaded to perform the real-time mask detection task.

C. Real-time Implementation

The key to realizing a real-time problem is to conduct processing on each frame of the video stream and ensure the processing is finished on this frame before the next frame starts. Therefore, real-time mask detection requires constantly looping over every frame in the video stream. To achieve this, QTimer in PyQt is used to set a timer to decide when to start processing the next frame. To be more specific, the system calls a Detect And Predict Mask function every 0.002s, which is equivalent to the time required to capture the next frame.

The Detect And Predict Mask function consists of two phases. In the first phase, the prepared FaceNet is used to extract the RoI, which is the human face, and in the second phase, the developed mask detector model is used on each RoI to detect the face mask. The prepared FaceNet is an SSD face detector implemented by OpenCV. After a face is detected in the image, the Detect And Predict Mask function eventually returns a tuple with the face locations in the image and the predictions of the mask. According to the prediction result, a corresponding label of “mask” or “no mask” and a bounding box are displayed on the output frame in the GUI.

D. Fine-tuning MobileNetV2

There are multiple pre-trained CNN models with promising performance, such as ResNet and MobileNet. A common feature of these pre-trained models is that a relatively large dataset, for example, ImageNet, is used for training. Therefore, the advantage of pre-trained models is the ability to extract basic features from the shallow network and abstract features from the deep network. By fine-tuning these models, plenty of time to prepare a large amount of dataset and computational resources can be saved, because it avoids developing a new model from scratch. Besides, it also allows the proposed model to be more suitable for dealing with a particular task since a proper fine-tuning design can help in optimizing the parameters, increasing the generalization ability and accuracy of the model, and decreasing the risk of overfitting. To sum up, fine-tuning is a useful strategy to establish a baseline model while saving considerable time [19]. In the training process, the original base layers of MobileNetV2, which is the pre-trained model, are frozen to be untrainable, and merely the developed new head with fully-connected layers is trained. By freezing the base model layers, the weights of the base layers are not updated during the backpropagation process, while the head layer weights are tuned to allow the network to focus on learning the features of the data in subsequent deep layers.

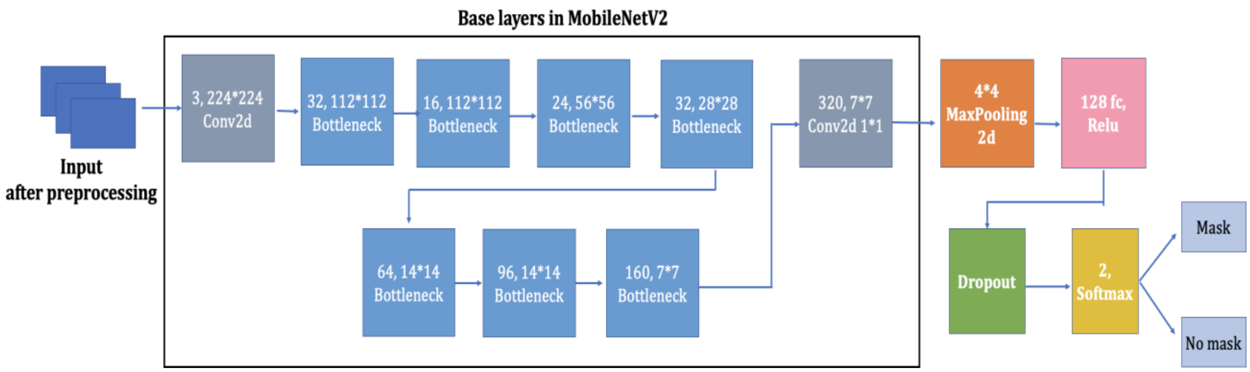

Fig. 3. CNN architecture of the mask detector model

In this project, when loading the pre-trained model of MobileNetV2 to build the proposed mask detector, the head fully-connected layers were excluded to be re-constructed later. In the developed new head model, max-pooling layers for 2D images were added to reduce the calculation cost while reserving the most important features. Then, the flattened high-level features were fed into the fully connected layer for classification. An activation function "Relu" was applied to increase non-linearities in the model, and 30% of neurons were randomly dropped to avoid the overfitting problem. Finally, the softmax function was chosen as a classifier at the output layer, which is able to give a probability set for each class label and choose the highest probability as the output. To allow the model to be more applicable for the binary classification problem to predict "mask" or "no mask", the output classes were specified to be two. Figure 3 demonstrates the complete CNN architecture of the proposed mask detector model.

E. Multithreading

In the process of incorporating the function of temperature measurement into the mask detection system, it was found that the temperature data reading and webcam image reading interfered with each other and in turn, a program termination was caused. Therefore, to allow the measured temperature data to be successfully shown in the GUI while the images are constantly being read from the webcam, another thread is used to read the temperature from a serial port where the thermal sensor MLX90614 is connected to the Arduino UNO R3 board.

F. IFTTT for IoT Notification

IFTTT is an external application, which provides a software platform that connects applications and services from different developers to trigger one or more automation involving those applications and services [5]. It uses "If this happens then execute that" logic in the application. The working principle of IFTTT is that it works by using an HTTP POST request to trigger an event. This is used to send a high-temperature notification alert to an administrative staff smartphone when the temperature is above a threshold value.

G. Brightness Checking

When testing the performance of the system, it was found that the ambient lighting condition may affect the mask detection results. To be more specific, if a hand is used to cover the mouth to pretend wearing a mask when the light is poor, the system will incorrectly determine that a mask is detected. However, if the light is sufficient, the system will always provide a correct detection result. Therefore, to avoid any mistakes in the system, brightness checking is necessary. It is found that the grayscale values of a dark image lie in the 0-30 range of the grayscale map (which is from 0 to 255), while for a bright image, the values lie above that range [20]. This suggests that the brightness of an image can be determined by analyzing the distribution of grayscale values. Therefore, a brightness checking function by applying the idea of a grayscale map is implemented. In this function, OpenCV is used to crop each frame in the video stream to allow it only to focus on the face and convert the face image into grayscale. By analyzing the proportion of dark pixels in the grayscale map, the brightness of the face can be checked.

V. SYSTEM EVALUATION AND RESULTS

A. Training Loss and Accuracy of Mask Detector

The loss and accuracy during the training process were used to evaluate the model performance. Figure 4 shows the loss and accuracy of the proposed mask detector model in each epoch. There are four different variables, which are training loss, training accuracy, validation loss, and validation accuracy.

Since the development of MobileNetV2 incorporates the core concept of ResNet by using residual block, ResNet should also be utilized as the backbone of the proposed mask detector model. Therefore, ResNet was attempted to be the baseline model. In the trial, the technique of fine-tuning the head layers and the hyperparameters remained the same. The loss and accuracy of the model in each epoch based on ResNet are given in Figure 5. The training speeds of the two models in each epoch are compared in Figure 6.

B. System Performance Testing

Figure 7 gives the testing results of the proposed system performance in different cases. From this figure, it is evident that the system is able to give the correct detection result when

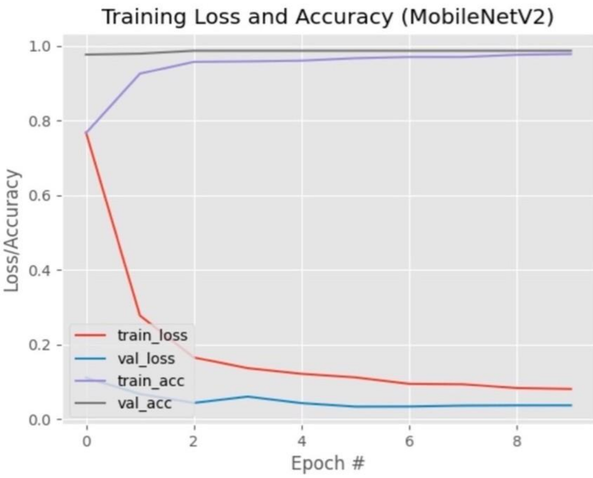

Fig. 4. Loss and accuracy of the proposed system

Fig. 5. Loss and accuracy when using ResNet

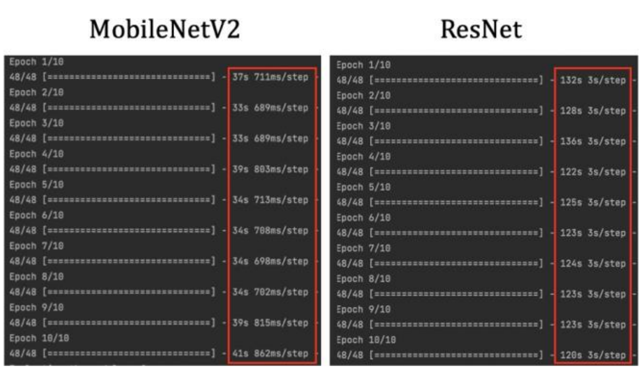

Fig. 6. Training speed comparison between MobileNetV2 and ResNet

a hand is used to pretend to be a mask. Also, entering is allowed when body temperature is normal. It is worth mentioning that the temperature is measured on the wrist for the convenience of use. To be more specific, for the hygiene concern, the wrist should be placed over the non-contact thermal sensor with a height of at least 1 cm above the sensor. Since the wrist can be exposed, the body temperature on the wrist may normally be lower than 36º C. By deliberately touching a cup of hot water for example, an abnormal temperature (i.e. above 37º C) can be obtained and entering is not allowed in this case. The notification sending event is triggered immediately when the system detects an abnormal temperature. The figure also shows an example of a notification received on a mobile phone screen saying "Alarm! Someone's temperature is very high! 41.90 degrees Celsius!". The insufficient light warning has also been tested. When the mask detection and temperature measurement on an employee are both completed, there will be a few moments when no face is detected on the scene, and the system is waiting for the next user to restart. Therefore, the system will be reset as the last sub-figure shows.

VI. DISCUSSIONS

The previous section demonstrates the results of the project. It is shown that the proposed system exhibits relatively complete and powerful functions and is suitable for practical applications. This section further discusses the results of the proposed system. Besides, several limitations and future work are also discussed.

A. *Result Reliability*

Referring to Figure 4, it can be seen that the mask detector with MobileNetV2 as the base model has an accuracy of about 98%. From the change of training loss, the trained model witnesses a decent convergence and fitting, because the smaller the loss,the more accurate the prediction of the model is. Besides, it is found that the validation accuracy is higher than the training accuracy. This might be because the number of samples for validation is generally much smaller than that for training (20% for testing, 80% for training). The second reason is that data augmentation, which increases data diversity and, hence, the learning difficulties, is generally not performed on validation data. Finally, since dropout is merely used on the training set, there will be a large loss of the training dataset.

In comparison, the mask detector based on ResNet experiences a relatively undesirable convergence, as shown in Figure 5. Despite the fact that it can also achieve a promising performance with an accuracy of about 94%, from Figure 6, it can be seen that the training speed is much slower than that of using MobileNetV2 as the backbone. To be more specific, for MobileNetV2, the training speed is about 35 seconds per epoch; while for ResNet, it is about 125 seconds per epoch. To sum up, after changing the hyperparameters in multiple trials, the proposed mask detector model has reached an outstanding performance with the most suitable configuration.

B. *Limitations*

In the process of conducting the project, it was found that if objects other than masks were used to cover the mouth,

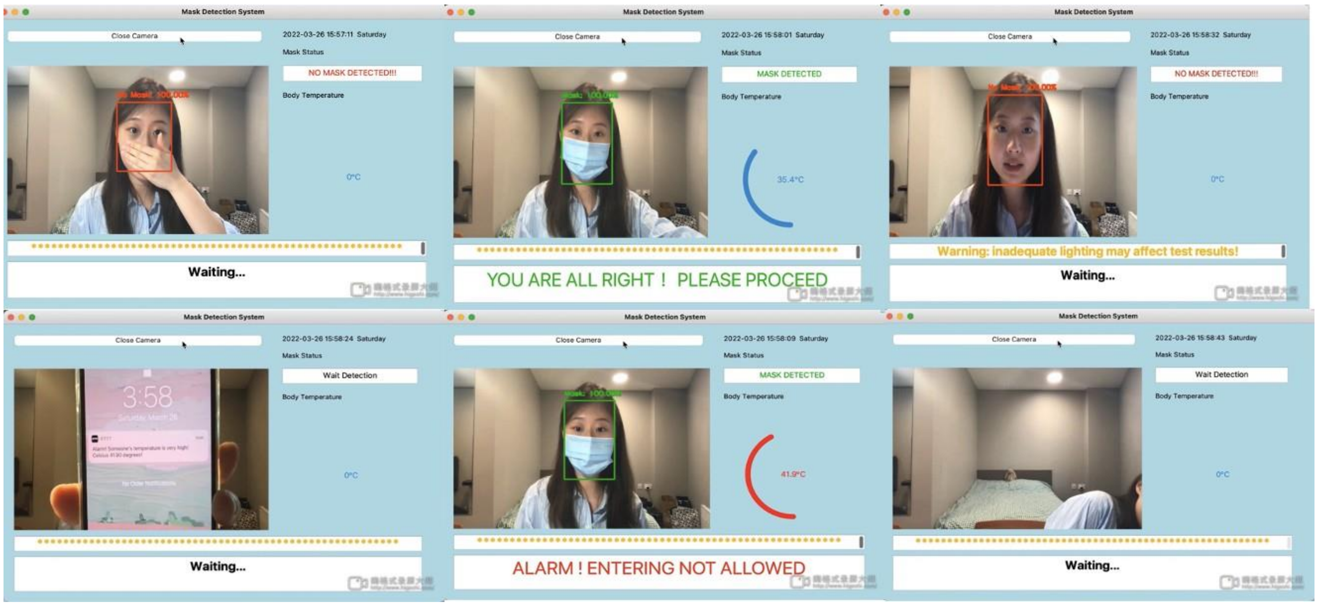

Fig. 7. Performance testing in different cases

the system would still consider that a mask was worn. After realizing this limitation, more confusing data were added to the input dataset to overcome it. Since finding these confusing data was time-consuming, about 60 pieces of confusing data were finally added to train the model. They include pictures of items that are likely to be used as masks in real life, such as books and mobile phones. However, after training, it was found that the proposed mask detection model still failed to provide the "mask is not detected" result when using some other objects. After communicating with the technical staff of a number of the commercially available mask detection systems on the market, it is confirmed that the current mask detection systems are almost all suffering from this limitation. Moreover, when covering the mouth by hand, these systems would still consider this as wearing a mask, while, fortunately, the proposed system can give correct detection result in this case. This is regarded as a contribution of the project to the industry world.

There may be two reasons for the above limitation. The first possible reason is that the features of masks are not obvious and not enough. Hence, other objects with these features can easily be viewed as masks. Secondly, it might be that the prediction is based on skin or non-skin. This implies that any area that is not skin may be identified as wearing a mask. Due to the complexity of the neural network, the suggested solution for this problem is to add more confusing data (at least 200 pieces) to the neural network model for training.

C. Future Work

The developed system can be used for multi-person mask detection simultaneously because it can detect all faces in the frame and then perform mask prediction on them at the same time. There is a considerable room for the proposed system to be expanded and developed in the future. First, the image quality of the system is not ideal, thus, cameras with higher resolution could be used to collect image data. Second, the current system is basically a software system developed on a laptop, so, in the future, dedicated hardware modules could be used to make it easily and quickly deployed to the market. For example, the mask detection algorithm can be applied to an independent hardware device, which can become an integrated mask detection and temperature measurement product. Moreover, more development can be achieved on the IoT side. For instance, the data on that product can be transmitted back to the background host computer through a Wi-Fi or Bluetooth modules for data storage and analysis. In addition, to enhance system's competitiveness and to reach a broader range of application scenarios, it is also possible to further explore how to perform face recognition while detecting a mask in the future.

VII. CONCLUSIONS

In conclusion, this paper demonstrates a system that can realize automatic real-time mask detection and body temperature measurement. The mask detection is achieved through a mask detector model that is constructed with a two-stage neural network algorithm. Specifically, the model was developed by fine-tuning a classic pre-trained model in CNN called MobilNetV2. The temperature measurement is achieved using a contactless thermal sensor with Arduino. The results of both functions are displayed through a well-designed GUI. In addition, this project also implements an additional function related to IoT. That is, when someone's temperature is found be too high, the system sends a high-temperature warning to the mobile phone of the Administration staff. This function is implemented with the aid of an external application called IFTTT. Since it is found that the brightness of the face may also affect the result of mask detection, the system also developed an ability to detect the brightness of the image and give a warning in case the brightness is too low. The test results show that the suggested mask classification model has an accuracy of about 98%. However, there exists several limitations in the project. For example, the system would still incorrectly identify other objects as masks. In the future, more confusing data should be added to train the model and try to solve the incorrectness of object detection in the system. A higher resolution camera and a more accurate thermal sensor should be used to improve system performance. To make the system more competitive, hardware modules should be employed to build the system as a commercial product.

REFERENCES

[1] U. Pranesh, S. Mukund, K. Sudarshan, Z. Kiran, and N.B Sai Shibu, “Development of Contactless Employee Management System with Mask Detection and Body Temperature Measurement using TensorFlow”, 2021 Sixth Inter- national Conference on Wireless Communications, Signal Processing and Networking (WiSPNET), 2021, pp. 235-240, doi: 10.1109/WiSP- NET51692.2021.9419418.

[2] NetEase News, “Real-time dynamic broadcast of the world’s new crown epidemic”, 2022. [online]. Available: https://wp.m.163.com/163/page/news/virus_world/index.html (Accessed: 14th April 2022)

[3] W. H. Organization, “Who disease outbreak news: novel coronavirus-republic of. korea (ex-china),” Jan. 2020.

[4] R. Girshick, “Fast R-CNN”, in 2015 IEEE International Conference on Computer Vision (ICCV), pp. 1440–1448, 07-13 December 2015.

[5] IFTTT, “What is IFTTT?”, 2022. [Online]. Available: https://ifttt.com/explore/new_to_ifttt. [Accessed: 15th April 2022].

[6] Z. Wang, G. Wang, B. Huang, Z. Xiong, Q. Hong, H. Wu, P. Yi, K. Jiang, N. Wang, Y. Pei et al., “Masked face recognition dataset and application”, arXiv preprint arXiv:2003.09093, 2020.

[7] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, “You only look once: Unified, real- time object detection”, in Proceedings of the IEEE conference on computer vision and pattern recognition, 2016, pp. 779–788.

[8] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C.-Y. Fu, and A. C. Berg, “SSD: Single shot MultiBox Detector,” in European Conference on Computer Vision, Springer, 2016, pp. 21–37.

[9] R. Girshick, J. Donahue, T. Darrell, et al. “Rich feature hierarchies for accurate. object detection and semantic segmentation,” in Proceedings of the IEEE conference on computer vision and pattern recognition, 2014, pp. 580– 587.

[10] J. Deng, W. Dong, R. Socher, et al. “ImageNet: A large-scale hierarchical image database”, in 2009 IEEE conference on computer vision andpattern recognition. 20-25 June 2009, pp. 248–255, DOI: 10.1109/CVPR.2009.5206848

[11] K. He, X. Zhang, S. Ren, and J. Sun, “Deep residual learning for image recognition”, in Proceedings of the IEEE conference on computer vision and pattern recognition, 2016, pp. 770–778.

[12] A. G. Howard, M. Zhu, B. Chen, et al. “MobileNets: Efficient convo- lutional neural networks for mobile vision applications”, arXiv preprint arXiv:1704.04861, April 2017.

[13] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L.C Chen, “MobileNetV2: Inverted residuals and linear bottlenecks”, in Proceedings of the IEEE conference on computer vision and pattern recognition, 2018, pp. 4510-4520.

[14] K. Zhang, Z. Zhang, Z. Li, and Y. Qiao.” Joint Face Detection and Alignment Using Multitask Cascaded Convolutional Networks”, IEEE Signal Processing Letters, vol. 23, no. 10, pp. 1499-1503, Available: 10.1109/lsp.2016.2603342.

[15] G. J. Chowdary, N. S. Punn, S. K. Sonbhadra, and S. Agarwal,” Face mask detection using transfer. learning of. inceptionv3”, International Conference on Big Data Analytics. Springer, Cham, 2020.

[16] P. Nagrath, R. Jain, A. Madan, et al. “SSDMNV2: A real time DNN-based face mask detection system using single shot multibox detector and MobileNetV2”, Sustainable cities and society, 66, 102692, 2021

[17] M. Jiang, X. Fan, H. Yan, “RetinaFaceMask: A face mask detector”, arXiv preprint. arXiv:2005.03950, 2020.

[18] A. Rosebrock, “COVID-19: Face Mask Detector with OpenCV, Keras/TensorFlow, and Deep Learning” [online]. Available: https://pyimagesearch.com/2020/05/04/covid-19-face-mask-detector-with-opencv-keras-tensorflow-and-deep-learning/?ga=2.182872101.2060180566.1634023030-775672767.1634023030 (accessed 10th Oct 2021)

[19] E. Cetinic, T. Lipic and S. Grgic, “Fine-tuning Convolutional Neural Networks for fine art classification”, Expert Systems with Applications, vol. 114, pp. 107-118, 2018. Available: 10.1016/j.eswa.2018.07.026.

[20] H. Kim, and S. Araújo, “Grayscale Template-Matching Invariant to Rotation, Scale, Translation, Brightness and Contrast”. In Advances in Image and Video Technology. PSIVT 2007. Lecture Notes in Computer Science, vol 4872. Springer, Berlin, Heidelberg.