Face mask detection using deep learning on NVIDIA Jetson Nano

Noor Faleh Abdul Hassan, Ali A. Abed, Turki Y. Abdalla
Department of Computer Engineering, College of Engineering, University of Basrah, Basrah, Iraq

Article Info

Article history:

Received Aug 14, 2021
Revised Mar 23, 2022
Accepted Apr 17, 2022

Keywords:

Computer vision
COVID-19
MobileNetV2
NVIDIA jetson nano
OpenCV

ABSTRACT

In December 2019, the coronavirus pandemic started. Coronavirus desease-19 (COVID-19) is transmitted directly from contaminated surfaces via direct touch. To combat the virus, a multitude of equipment is needed. Masks are a vital element of personal protection in crowded places. As a result, determining if a person is wearing a face mask is critical to assimilating to contemporary society. To accomplish the objective, the model presented in this paper used deep learning libraries and OpenCV. This approach was chosen for safety concerns due to its high resource efficiency during deployment. The classifier was built using the MobileNetV2 structure, which was designed to be lightweight and capable of being utilized in embedded devices such as the NVIDIA Jetson Nano to do real-time mask recognition. The stages of model construction were collecting, pre-processing, splitting data, creating the model, training the model, and applying the model. This system utilized image processing techniques and deep learning to process a live video feed. When someone is not wearing a mask, the output eventually produces an alarm sound through a built-in buzzer. Experimental results and testing were used to verify the suggested system's performance. Including both training and testing, the achieved recognition rate was 99%.

Corresponding Author:

Ali A. Abed
Department of Computer Engineering, College of Engineering, University of Basrah
Basrah, Iraq
Email: ali.abed@uobasrah.edu.iq

1. INTRODUCTION

Computer vision science uses various imaging technologies as input devices, rather than visual organs, with computers processing and interpreting visual information in place of the brain. Computer vision technology is continuously developing, and computers are now capable of recognizing and responding to a wide variety of facial expressions [1]. At the moment, the coronavirus desease-19 (COVID-19) epidemic is sweeping the globe. Coronavirus is discharged into the air when someone coughs, talks, or sneezes and may infect others in close proximity [2]. COVID-19 infected approximately 5 million individuals in 188 countries in less than six months. The virus spreads via intimate contact and in densely populated regions. Its expansion has resulted in unprecedented levels of scientific collaboration on a global scale [3].

According to the World Health Organization (WHO), the COVID-19 virus is primarily transmitted via breathing fluids and social integration. To control the spread of this infection, certain preventive measures, such as isolation and the usage of masks [4], [5]. Face mask identification has established itself as an enthralling area of study in computer vision applications. Face detection has a variety of uses, ranging from facial recognition to capturing facial movements, the latter of which needs a very precise display of the face [6]. Because of the expansion of the coronavirus illness, it has become necessary to wear a face mask to avoid viral infection. Only those wearing masks are permitted to enter the office or any other organization. Without the face mask, entry into the organization is restricted [7]. COVID-19 has made us understand the need to know the severe consequences of not wearing one now more than ever. However, it is crucial to implement face mask detectors at bus stations, crowded residential areas, market places, educational institutions, and treatment centers to ensure public safety [8].

Mehmood *et al*. [9] introduced a light-weight deep learning method with energy devices. Using the Viola-Jones technique in combination with several classifiers, they were able to identify and then remove certain portions of the photograph's subject that included just the face. Additionally, the method utilized was a low-cost pre-trained convolutional neural network (CNN) containing features for representing faces. The repository's large data set was then indexed to improve the speed of retrieval for real-time searching. In the final analysis, Euclidean distance was used to assess the degree of similarity between the query and repository pictures. Triantafyllidou *et al*. [10] presented a light-weight deep CNN with a state-of-the-art recall rate that outperformed the competition on the demanding face detection dataset and benchmark (FDDB) dataset, which had just 113.864 features that were not restricted. Lin *et al*. [11] presented encouraging outcomes on FDDB, annotated facial landmarks in the wild (AFLW), and WIDER FACE evaluations using the G-Mask technique. RoIAlign was able to capture spatial position, and feature extraction was conducted using ResNet-101. Generalized Intersection over Union (GIoU) was used as a bounding box error function to raise the number of identifications. Botella-Campos *et al*. [12] presented a method for identifying individuals nearby. Face recognition, which noticed individuals taking photos with the phone's camera while in a conversation. Ravidas and Ansari *et al*. [13] proposed a deep CNN (DCNN) for multiple-view faces. Oumina *et al*. [14] investigated deep convolution networks to extract intricate characteristics from photos of faces. Support vector machine (SVM) and k-nearest neighbors (K-NN) were used to assess the collected characteristics. Combining the MobileNetV2 model with the SVM yielded the best accuracy with a 97.1% success rate.

Sikandar *et al*. [15] suggested a technique to detect masked faces from automated teller machine (ATM) monitoring security cameras correctly, and the detection rate was 96.48%. Rao *et al*. [16] proposed CNN model for facial detection that had an accuracy of 91.21% while scanning the public without a facemask. The information from the identity database was linked with a phone number and address database to get specifics on that individual, and an appropriate amount was sent to his mobile number and address. Qin and Li [17] introduced a novel technique for detecting the facemask-wearing nation called super resolution and classification network (SRCNet) to acquire an accuracy rate of 98.70%. Sheikh and Qidwai [18] proposed a new technique using a light mobile system and evaluated the execution of their classifier developed using MobileNetV2 with a remarkable degree of accuracy of 91.68%. Sethi *et al*. [19] proposed an approach that integrated a single-stage and two-stage detector. ResNet50 was utilized as a baseline for fusing high-level semantic information. The transfer learning approach combined the various feature maps into a single map that integrated more information. The researchers also developed a bounding box modification to help enhance the localization of a mask's position. Using three famous base architectures, ResNet50, AlexNet, and MobileNet succeeded, resulting in almost excellent outcomes of 98.2% with little computation time.

This paper presents machine vision and learning techniques with OpenCV, Keras, and TensorFlow to do the work efficiently. COVID-19 transmission is inhibited by a security camera that detects people using face masks while identifying others who wear face masks but do not conceal the face. The best results are achieved by using the least amount of time and resources. Train the model by using images of individuals using and not applying face masks. This research suggests a method that produces boundary boxes (red or green) that indicate whether individuals are wearing masks. This application will keep track of how many individuals use face masks each day. The critical contribution of the mask detection process is provided by using a new MacBook with fast central processing unit (CPU) and graphics processing unit (GPU) specifications to training a model to acquire high accuracy reach of 99%, which is extremely high than every other researcher when training and testing on another personal computer (PC). Furthermore, after integrating the trained model on a low-cost developer kit, tiny and effective officially named the NVIDIA Jetson Nano. This tiny equipment can be used to inhibit the infection from spreading in crowded institutions like schools. A Logitech USB camera C920e is utilized with a Nano kit to capture real-time video. TensorFlow and Keras python libraries are utilized in the process. When the model identifies those who are not employing a face mask, a buzzer sounds an alert. Instead of the typical system, which detects just one face mask, the work enables a multiple face mask detection method.

The paper's part is structured as follows: here it is. Section 2 covers the scheme as well as the training and testing database. Section 3 contains the complete findings and commentary. Section 4 focuses on the facts and conclusions that need further investigation.

2. RESEARCH METHOD

CNN is a deep neural network class that aims to replicate image analysis via the brain's visual cortex (cerebral cortex). Previously, most computer vision researchers extracted characteristics manually to achieve better classification results. In the training stage, CNNs are capable of automatic feature extraction since they use the pool and convolutional layers. Many kinds of filters have been learned to accomplish a particular classification goal, resulting in a convolutional layer. At the same time, the pooling layers shrink the dimension of feature extraction while retaining an image's size and appearance. Several CNN models are widely used [20].

2.1. The proposed model

The learning algorithm used for face mask identification in this work is MobileNetV2, whereas the visual classifier is used for face mask identification. This model uses Google's convolutional neural network (CNN) combined with improved computational speed and efficiency [21]. It is suitable for both high-and low-computing scenarios. MobileNetV2 expands on the concepts of MobileNetV1 [22]. The MobileNetV1 network architecture includes two levels. Started, a layer is also known as a Depthwise convolution and uses one convolution filter for each input port to perform light filtering. Finally, a layer is a convolution of 1×1, known as a pointwise convolution, which uses linear combinations of input channels to generate extra features. In this instance, the ReLU6 is used as a reference point. ReLU6 is often employed because of its strong statistical characteristics when used with low-precision computation [23].

Blocks are classed into two types in MobileNetV2 [24]. The first block contains a stride of 1 and is the residual block. To be effective, a decreasing block must have a stride of two. The structure is made up of both kinds of blocks. The first layer is a ReLU activation followed by a convolution with a pool size of 1×1. In other words, Depthwise convolution is indeed the second layer. In this third layer, 1×1 convolution is performed again, but no non-linearity is added. To paraphrase, when ReLU is employed, deeper networks have the capability of classifiers based on non-zero output produced. One convolution layer and 19 bottleneck layers are part of the MobileNetV2 network design. An illustration of MobileNet architecture is shown in Figure 1.

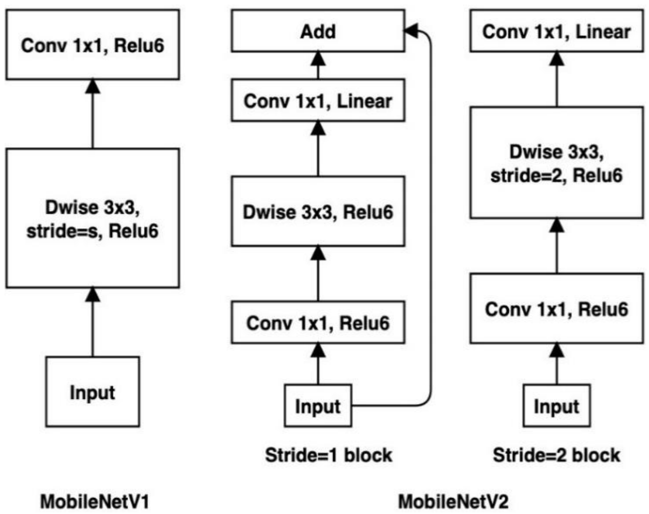

Figure 1. MobileNet architecture

The model begins by loading the dataset for mask detection. Deep learning libraries are used for data preparation. TensorFlow, Keras, and OpenCV are all used to train the MobileNetV2 classifier. CNN algorithm is used in the proposed system of this paper. The mask detection technique is adopted by utilizing a MacBook M1 with fast CPU and GPU capabilities to train and test a model to achieve high accuracy of 99% for training and 100% for testing. After that, the trained model is being integrated on a low-cost development kit named the Jetson Nano. To record real-time video, a Logitech USB camera C920e is used in combination with a Nano kit. As required, faces are retrieved from pictures and video streams after building a face mask classification model. Mask detection identifies humans who are utilized masks or who are not utilized masks at all, depending on the situation. When no mask is detected, the built-in buzzer sounds a warning to wear a mask. Figure 2 shows the suggested system's flowchart.

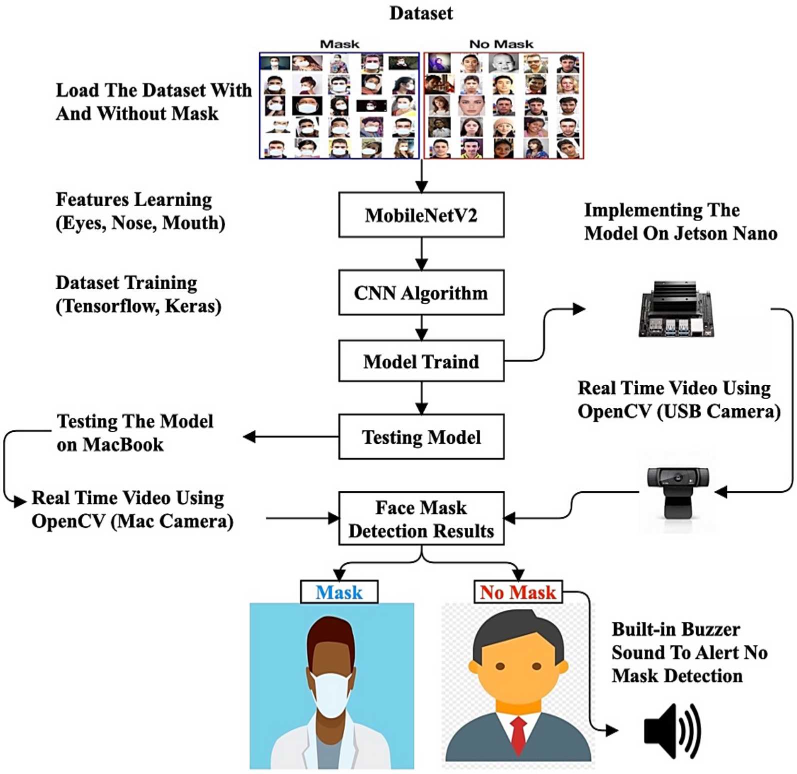

Figure 2. Flowchart of the proposed model

2.2. Dataset

The initial stage in developing a face-mask recognition model is the gathering of data. The dataset includes both mask wearers and non-mask wearers is collected from another mask dataset. This article utilizes 2,165 images of data that were masked together with 1,930 images of data that were not masked to generate this dataset. It has been cropped so that just the item's face remains visible. Once the data has been tagged, it is next divided into two groups: those that have a mask and those that do not. When the data has been tagged, it is then split into two groups, which are already known.

2.3. Pre-processing

Before data training and testing, the pre-processing takes place. The four steps of pre-processing are as follows: downscaling the picture, converting the image into an array, utilizing MobileNetV2 to preprocess the input, and lastly, performing hot encoding on the labels. Because training models are so effective, pre-processing such as picture scaling is essential to computer vision. More often than not, models perform better when images are reduced in size. Images are discovered to be 224×224 pixels in size. Next, the data is converted into arrays for the loop function to use. The pre-processing will utilize the pre-trained MobileNet model. To conclude this phase, labeled data must be executed via hot encoding since learning algorithms cannot process labeled data directly. In other words, instead of numeric input and output, any variable must be able to understand and evaluate the tagged data, the method is given a numerical label as well.

2.4. Dividing the data and building the model

Of the total data, 75% is made up of training data, while the remaining 25% is comprised of testing data. All the masks in this collection have been included. However, some masks are not. The proposed model is built in six steps: training picture generator, basic model using MobileNetV2, model parameter addition, compilation, model training with MacBook M1 chip, and model storing for future predictions on Jetson Nano kit are all included in this work.

2.5. Testing the model

Experiments are performed on a new mac Apple M1, which has 16 GB of RAM and an 8-core CPU and GPU. Several experimental trials are created and implemented using the Python 3.8 kernel. The metrics used to assess the MobileNetV2 model are shown in (1)-(4), which are based on [25].

$$Accuracy = \frac{[TP+TN]}{[TP+FP+TN+FN]} \tag{1}$$

$$Precision = \frac{[TP]}{[TP+FP]} \tag{2}$$

$$Recall = \frac{[TP]}{[TP+FN]} \tag{3}$$

$$F1 - Score = 2 \times \frac{[Precision \times Recall]}{[Precision + Recall]} \tag{4}$$

The abbreviations TP signifies true positive, TN signifies true negative, FP signifies false positive, and FN signifies false negative [26]. True positive values in the previous equations refer to pictures that have been labeled as true and produced a true result as predicted by the model. Similarly, true negative pictures are those that have been categorized as true but generated an incorrect outcome because of prediction. False-positive images are ones that have been categorized as false yet produced false positives because of prediction. False-negative images are those that are categorized as false yet turn out to be accurate, resulting in false negatives. Due to the balanced nature of the courses, accuracy is a good starting point. Precision is a metric that indicates the number of expected positive values. The recall statistic quantified a classifier's ability to identify all positive cases, while the F1-score quantified test accuracy. These evaluation measures have been selected because they provide the most accurate findings through a balanced dataset. Model testing is divided into stages to verify that it is capable of making accurate predictions. Predictions are made about the testing set's first stage.

2.6. Implementing the model

The model is invoked using the NVIDIA Jetson Nano shown in Figure 3. The face detection method is activated once the USB camera scans the images or real-time video. If a face is spotted, the procedure moves on to the next step. On detected frames identifiers, reprocessing is done, which includes decreasing the picture size, transforming it to a matrix, then the input is pre-processed using MobileNetV2. After that, the stored model is utilized to predict the input data. Enhance the model for predicting the processed input image that has been previously created. Also, the video frame includes the person's image in a mask and the percentage they believe is likely. A buzzer sounds or beeps when no mask has been seen.

Figure 3. NVIDIA Jetson Nano developer kit

3. RESULTS AND DISCUSSION

Table 1 summarizes the results of twenty rounds of evaluating the model's loss and accuracy while training on the MacBook M1 chip. According to Table 1, accuracy improves as the second period begins, whereas loss decreases. If the accuracy remains steady, no further iterations are needed to improve the model's accuracy.

The model is evaluated in the next step to get the results shown in Table 2. The average macro function computes F1 for every label and gives the average without considering the percentage for every label in the dataset. The weighted average function computes F1 for every label and generates an average while accounting for the proportion for every label in the dataset. The simulation outcome of putting the principles into action on the MacBook M1 chip is shown in Figure 4. A possible benefit of the proposed method is that it might easily snap a portrait of several people in one photo. The result of real-time multiple face mask identification on the NVIDIA Jetson Nano GPU development board is shown in Figure 5. It also demonstrated multiple face mask detection with beeps when no mask is present.

Table 1. Iterations of evaluating the model's accuracy

Epoch	Loss	Accuracy	Val_Loss	Val_Accurcy
1	0.3994	0.8578	0.1517	0.9853
2	0.1533	0.9620	0.0857	0.9902
3	0.1068	0.9707	0.0640	0.9915
4	0.0846	0.9719	0.0549	0.9915
5	0.0694	0.9796	0.0504	0.9927
6	0.0621	0.9824	0.0457	0.9927
7	0.0563	0.9812	0.0434	0.9915
8	0.0574	0.9821	0.0390	0.9915
9	0.0507	0.9861	0.0386	0.9902
10	0.0487	0.9846	0.0393	0.9902
11	0.0381	0.9889	0.0376	0.9915
12	0.0417	0.9883	0.0370	0.9902
13	0.0423	0.9867	0.0398	0.9902
14	0.0442	0.9864	0.0352	0.9915
15	0.0445	0.9870	0.0371	0.9902
16	0.0363	0.9877	0.0373	0.9902
17	0.0437	0.9833	0.0344	0.9915
18	0.0346	0.9898	0.0333	0.9915
19	0.0343	0.9883	0.0348	0.9902
20	0.0320	0.9895	0.0309	0.9915

Table 2. Evaluation of the model

	Support	Recall	F1-score	Precision
Mask	433	100%	99%	99%
No mask	386	98%	99%	100%
Accuracy	819		99%	
Average macro	819	99%	99%	99%
Average weighted	819	99%	99%	99%

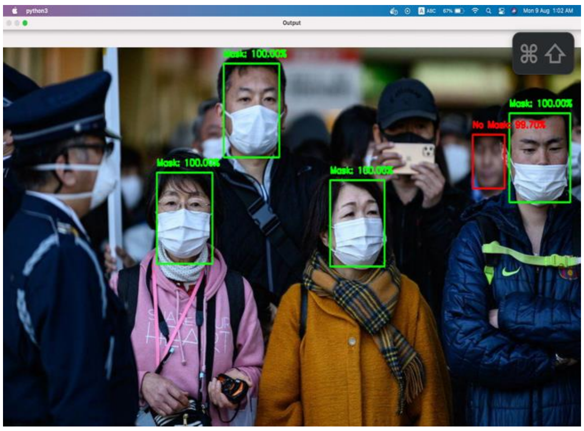

Figure 4. Predicting input data on MacBook

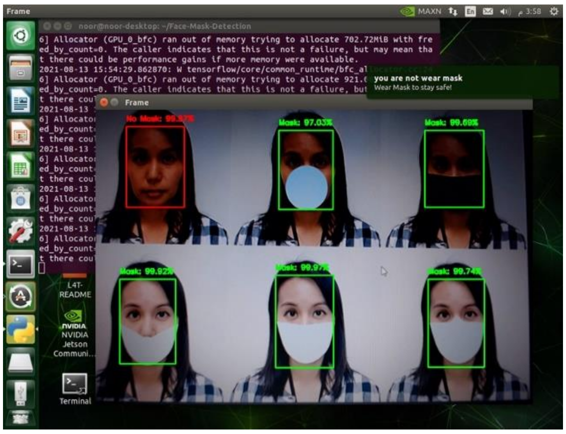

Figure 5. Real-time on NVIDIA Jetson Nano

4. CONCLUSION AND FUTURE DEVELOPMENT

The suggested scheme employs computer vision, the MobileNetV2 architecture, and the NVIDIA Jetson Nano board to safeguard society and protect individuals by inhibiting the COVID-19 virus from propagating if there are too many people in one location. The notion of detecting whether people are wearing face masks is best applied in high-traffic areas, such as markets, offices, and train stations, because transmission possibilities are most significant in those regions. Anywhere may be installed the system to capture a video input with the live stream or recorded video feed, or a combination. Therefore, a detection model that can function in real-time is precise enough to detect small things like masked faces and buzzer sounds when no face mask is identified might be highly beneficial in these edge applications in surveillance systems. The experimental results proved a high accuracy (recognition) rate of about 99% while training the model using MacBook M1 chip. In the future, an integrated system with a temperature sensor and an external warning device can be used in the placement of outside to eliminate the spreading of the virus.

REFERENCES

[1] G. Yang *et al.*, "Face mask recognition system with YOLOV5 based on image recognition," in *2020 IEEE 6th International Conference on Computer and Communications (ICCC)*, Dec. 2020, pp. 1398–1404, doi: 10.1109/ICCC51575.2020.9345042.

[2] S. Singh, U. Ahuja, M. Kumar, K. Kumar, and M. Sachdeva, "Face mask detection using YOLOv3 and faster R-CNN models: COVID-19 environment," *Multimedia Tools and Applications*, vol. 80, no. 13, pp. 19753–19768, May 2021, doi: 10.1007/s11042-021-10711-8.

[3] V. Vinitha and V. Velantina, "Covid-19 facemask detection with deep learning and computer vision," *International Research Journal of Engineering and Technology (IRJET)*, vol. 7, no. 8, pp. 3127–3132, 2020.

[4] S. Sen and K. Sawant, "Face mask detection for covid-19 pandemic using pytorch in deep learning," *IOP Conference Series: Materials Science and Engineering*, vol. 1070, no. 1, Feb. 2021, doi: 10.1088/1757-899X/1070/1/012061.

[5] S. Yadav, "Deep learning based safe social distancing and face mask detection in public areas for COVID-19 safety guidelines adherence," *International Journal for Research in Applied Science and Engineering Technology*, vol. 8, no. 7, pp. 1368–1375, Jul. 2020, doi: 10.22214/ijraset.2020.30560.

[6] P. Nagrath, R. Jain, A. Madan, R. Arora, P. Kataria, and J. Hemanth, "SSDMNV2: A real time DNN-based face mask detection system using single shot multibox detector and MobileNetV2," *Sustainable Cities and Society*, vol. 66, Mar. 2021, doi: 10.1016/j.scs.2020.102692.

[7] Y. Ayyappa, P. Neelakanteswara, A. Bekkanti, Y. Tondeti, and C. Z. Basha, "Automatic face mask recognition system with FCM AND BPNN," in *2021 5th International Conference on Computing Methodologies and Communication (ICCMC)*, Apr. 2021, pp. 1134–1137, doi: 10.1109/ICCMC51019.2021.9418243.

[8] S. Sakshi, A. K. Gupta, S. Singh Yadav, and U. Kumar, "Face mask detection system using CNN," in *2021 International Conference on Advance Computing and Innovative Technologies in Engineering (ICACITE)*, Mar. 2021, pp. 212–216, doi: 10.1109/ICACITE51222.2021.9404731.

[9] I. Mehmood *et al.*, "Efficient image recognition and retrieval on IoT-assisted energy-constrained platforms from big data repositories," *IEEE Internet of Things Journal*, vol. 6, no. 6, pp. 9246–9255, Dec. 2019, doi: 10.1109/JIOT.2019.2896151.

[10] D. Triantafyllidou and A. Tefas, "Face detection based on deep convolutional neural networks exploiting incremental facial part learning," in *2016 23rd International Conference on Pattern Recognition (ICPR)*, Dec. 2016, pp. 3560–3565, doi: 10.1109/ICPR.2016.7900186.

[11] K. Lin *et al.*, "Face detection and segmentation based on improved mask R-CNN," *Discrete Dynamics in Nature and Society*, vol. 2020, pp. 1–11, May 2020, doi: 10.1155/2020/9242917.

[12] M. Botella-Campos, J. L. Gacía-Navas, A. Rego, S. Sendra, and J. Lloret, "A new system to detect coronavirus social distance violation," *International Journal of Electrical and Computer Engineering (IJECE)*, vol. 11, no. 6, pp. 5034–5048, Dec. 2021, doi: 10.11591/ijece.v11i6.pp5034-5048.

[13] S. Ravidas and M. A. Ansari, "Deep learning for pose-invariant face detection in unconstrained environment," *International Journal of Electrical and Computer Engineering (IJECE)*, vol. 9, no. 1, pp. 577–584, Feb. 2019, doi: 10.11591/ijece.v9i1.pp577-584.

[14] A. Oumina, N. El Makhfi, and M. Hamdi, "Control the COVID-19 pandemic: face mask detection using transfer learning," in *2020 IEEE 2nd International Conference on Electronics, Control, Optimization and Computer Science (ICECOCS)*, Dec. 2020, pp. 1–5, doi: 10.1109/ICECOCS50124.2020.9314511.

[15] T. Sikandar, W. N. A. W. Samsudin, M. F. Rabbi, and K. H. Ghazali, "An efficient method for detecting covered face scenarios in ATM surveillance camera," *SN Computer Science*, vol. 1, no. 3, May 2020, doi: 10.1007/s42979-020-00163-6.

[16] T. S. Rao, S. A. Devi, P. Dileep, and M. S. Ram, "A novel approach to detect face mask to control covid using deep learning," *European Journal of Molecular and Clinical Medicine*, vol. 7, no. 6, 2020.

[17] B. Qin and D. Li, "Identifying facemask-wearing condition using image super-resolution with classification network to prevent COVID-19," *Sensors*, vol. 20, no. 18, pp. 5236–5259, Sep. 2020, doi: 10.3390/s20185236.

[18] S. Sheikh and U. Qidwai, "Using MobileNetV2 to classify the severity of diabetic retinopathy," *International Journal of Simulation Systems Science & Technology*, Mar. 2020, doi: 10.5013/IJSSST.a.21.02.16.

[19] S. Sethi, M. Kathuria, and T. Kaushik, "Face mask detection using deep learning: An approach to reduce risk of Coronavirus spread," *Journal of Biomedical Informatics*, vol. 120, Aug. 2021, doi: 10.1016/j.jbi.2021.103848.

[20] M. Y. Kamil, "A deep learning framework to detect Covid-19 disease via chest X-ray and CT scan images," *International Journal of Electrical and Computer Engineering (IJECE)*, vol. 11, no. 1, pp. 844–850, Feb. 2021, doi: 10.11591/ijece.v11i1.pp844-850.

[21] S. A. Sanjaya and S. Adi Rakhmawan, "Face mask detection using MobileNetV2 in the era of COVID-19 pandemic," in *2020 International Conference on Data Analytics for Business and Industry: Way Towards a Sustainable Economy (ICDABI)*, Oct. 2020, pp. 1–5, doi: 10.1109/ICDABI51230.2020.9325631.

[22] A. G. Howard *et al.*, "MobileNets: Efficient convolutional neural networks for mobile vision applications," *CoRR*, Apr. 2017.

[23] T. Ghosh *et al.*, "Bangla handwritten character recognition using MobileNet V1 architecture," *Bulletin of Electrical Engineering and Informatics (BEEI)*, vol. 9, no. 6, pp. 2547–2554, Dec. 2020, doi: 10.11591/eei.v9i6.2234.

[24] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L.-C. Chen, "MobileNetV2: inverted residuals and linear bottlenecks," in *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, Jun. 2018, pp. 4510–4520, doi: 10.1109/CVPR.2018.00474.

[25] S. Saypadith and S. Aramvith, "Real-time multiple face recognition using deep learning on embedded GPU system," in *2018 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA ASC)*, Nov. 2018, pp. 1318–1324, doi: 10.23919/APSIPA.2018.8659751.

[26] A. Al Mamun, P. P. Em, and J. Hossen, "Lane marking detection using simple encode decode deep learning technique: SegNet," *International Journal of Electrical and Computer Engineering (IJECE)*, vol. 11, no. 4, pp. 3032–3039, 2021, doi: 10.11591/ijece.v11i4.pp3032-3039.

BIOGRAPHIES OF AUTHORS

Noor Faleh Abdul Hassan M.Sc. student at the Department of Computer Engineering-University of Basrah-Iraq. Received the B.Sc. degree in Computer Engineering in 2015. Her field of interest is Computer Vision, Embedded systems, and Robotics. She is currently a group researcher working in developing deep learning models and algorithms in Python. She can be contacted at email: pgs2341@uobasrah.edu.iq.

Ali A. Abed Assistant Professor at University of Basrah Department of Computer Engineering in Iraq. Received the B.Sc. & M.Sc. degrees in Electrical Engineering in 1996 & 1998 respectively. He received a Ph.D. in Computer & Control Engineering in 2012. His field of interest is Robotics, Computer Vision, and IoT. He is IEEE senior member, IEEE member in Robotics & Automation Society, IEEE member in IoT Community, member ACM. He is currently supervising a group of researchers working with developing deep learning models for computer vision applications. He can be contacted at email: ali.abed@uobasrah.edu.iq.

Turki Y. Abdalla is a Professor at the University of Basrah, College of Engineering. Received a Ph.D. in Electrical Engineering from the College of Engineering at the University of Basrah, Iraq. Robot control, multiple mobile robots, intelligent control, soft computing, and image processing are his research interests. He is an IEEE Senior Member. He can be contacted at email: turki.abdalla@uobasrah.edu.iq.