

Article

Automatic Face Mask Detection System in Public Transportation in Smart Cities Using IoT and Deep Learning

Tamilarasan Ananth Kumar ¹, Rajendrane Rajmohan ¹, Muthu Pavithra ¹, Sunday Adeola Ajagbe ^{2,*}, Rania Hodhod ³ and Tarek Gaber ^{4,5,*}

- ¹ Computer Science and Engineering, IFET College of Engineering, Gangarampalaiyam 605108, Tamil Nadu, India; tananthkumar@ifet.ac.in (T.A.K.); rjmohan89@gmail.com (R.R.); pavimuthu27@gmail.com (M.P.)
- ² Computer Science and Engineering, Ladoke Akintola University of Technology, Ogbomoso 210101, Nigeria
- ³ TSYS School of Computer Science, Columbus State University, Columbus, GA 31907, USA; hodhod_rania@columbusstate.edu
- ⁴ Computer Science & Software Engineering, University of Salford, Manchester M5 4WT, UK
- ⁵ Faculty of Computers and Informatics, Suez Canal University, Ismailia 41522, Egypt
- * Correspondence: saajagbe@pgschool.lautech.edu.ng (S.A.A.); t.m.a.gaber@salford.ac.uk (T.G.)

Abstract: The World Health Organization (WHO) has stated that the spread of the coronavirus (COVID-19) is on a global scale and that wearing a face mask at work is the only effective way to avoid becoming infected with the virus. The pandemic made governments worldwide stay under lock-downs to prevent virus transmissions. Reports show that wearing face masks would reduce the risk of transmission. With the rise in population in cities, there is a greater need for efficient city management in today's world for reducing the impact of COVID-19 disease. For smart cities to prosper, significant improvements to occur in public transportation, roads, businesses, houses, city streets, and other facets of city life will have to be developed. The current public bus transportation system, such as it is, should be expanded with artificial intelligence. The autonomous mask detection and alert system are needed to find whether the person is wearing a face mask or not. This article presents a novel IoT-based face mask detection system in public transportation, especially buses. This system would collect real-time data via facial recognition. The main objective of the paper is to detect the presence of face masks in real-time video stream by utilizing deep learning, machine learning, and image processing techniques. To achieve this objective, a hybrid deep and machine learning model was designed and implemented. The model was evaluated using a new dataset in addition to public datasets. The results showed that the transformation of Convolution Neural Network (CNN) classifier has better performance over the Deep Neural Network (DNN) classifier; it has almost complete face-identification capabilities with respect to people's presence in the case where they are wearing masks, with an error rate of only 1.1%. Overall, compared with the standard models, AlexNet, Mobinet, and You Only Look Once (YOLO), the proposed model showed a better performance. Moreover, the experiments showed that the proposed model can detect faces and masks accurately with low inference time and memory, thus meeting the IoT limited resources.

Keywords: COVID-19; artificial intelligence; face mask system; deep learning; smart bus



check for updates

Citation: Kumar, T.A.; Rajmohan, R.; Pavithra, M.; Ajagbe, S.A.; Hodhod, R.; Gaber, T. Automatic Face Mask Detection System in Public Transportation in Smart Cities Using IoT and Deep Learning. *Electronics* **2022**, *11*, 904. <https://doi.org/10.3390/electronics11060904>

Academic Editor: Chiman Kwan

Received: 31 January 2022

Accepted: 10 March 2022

Published: 15 March 2022

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Copyright: © 2022 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

The pandemic caused by COVID-19 has kept the global community awake for the previous year. In recent times, a nation with a strong gross domestic product like India has reported more than 3.89 crores [1]. Considering these facts, experts are devoting considerable efforts towards creating novel remedies to the present catastrophe. Additionally, the WHO Records indicate that significant efforts have been made by many developed and developing countries with the provision of masks, respirators, operating rooms, facial masks, and other critical health equipment [2]. In this context, the WHO has developed

tight regulations for COVID-19 patients, curfews, social isolation, and screening. Moreover, many countries have incorporated digital tools that enable citizens to distinguish nearby COVID-19 patients via wireless position tracking technologies. However, in light of the COVID-19 effect and proliferation, researchers have devoted numerous resources to discovering and improving COVID-19 remedies; many novel solutions have been proposed for the existing disastrous situation, including automated sanitizing devices for disinfecting medical supplies and humans, infrared imaging devices, and others [3]. Significantly, several judicial governments have enacted stringent health preventive policies and made serious attempts to clean specific topographical regions with a variety of disinfectants, in addition to conducting various health-awareness efforts to promote society's health. In rare instances, governments have also penalized individuals for attempting to violate health and safety regulations. It is still debatable whether or not strict adherence to health and safety requirements such as wearing a mask and maintaining social distance is observed [4].

Public transportation brings the community people into direct connection with everyone, often for longer durations, and subjecting them to regularly touched areas, enhances a person's chance of contracting and transmitting COVID-19 [4]. Maintaining a six-foot separation from people is frequently challenging on public transit. Individuals may be unable to maintain the required distance of six feet from other passengers seated adjacent or from individuals strolling in or transiting through bus terminals. From business to social interactions, all types of off- and on-screening activities are needed in public transportation as well as public entrances to ensure social welfare [5–7]. As a result, an intelligent entry device is needed that constantly recognizes the presence of a mask at the door opening mechanism.

To the best of our knowledge, there is no comprehensive system for identifying and monitoring COVID-19 face mask detection in public transportations using IoT technologies. This article presents an IoT-based face mask detection system in public transportations, especially buses through the collection of real-time data via facial recognition devices. The major contributions of the paper are as follows:

- The use of IoT and deep learning techniques to classify images with and without face mask and detect the presence of face masks in real-time video streaming.
- The development of an efficient low-cost face mask detection system that can be used in public transportation. Creating a dataset of face and non-face image and using it for evaluating the proposed system.
- Comparing with benchmark models (Alexnet, Mobinet, and YOLO), the proposed system showed better results.

The following defines the paper's organization. Literature review of the related works is provided in Section 2. Section 3 shows a detailed explanation of the methodology of the proposed work. A detailed explanation of the system implementation is provided in Section 4. Section 5 presents the findings of the study as well as a discussion of the proposed system. Finally, a conclusion, as well as recommendations for future work, are included in the last section of the paper.

2. Related Works

COVID-19 has currently circulated the world without a 100% effective vaccine. Wearing a face mask has been found to be an effective method of anti-microbial barrier protection and numerous others, including washing the hands, practicing good hygiene, and frequent hand washing [8]. The idea of wearing a face mask in public places has now entered the realm of the public consciousness. This elevates the importance of AI and deep learning to allow for automatic face [9]. Researchers initially concentrated on using gray-scale images of faces to identify people. Some researchers were working on pattern identification models, such as AdaBoost [10], one of the most effective classifiers at the time. Viola–Jones detectors were developed later, allowing for real-time face detection. However, it had trouble working correctly in dull and dim light, which led to misclassifications under these conditions. In many fields, deep learning is used because of its popularity and unique

features, which include detection, identification, classification and recognition of objects. This led to the development of a robot that is capable of detecting the face of any human being and processing the data depending on the needs. As depicted in the process model, it begins by extracting the input image and its features using 3×3 matrices as Convolution (ConV) across a stride of 1 [11]. Featured maps are then created by taking the dot product of the layers that came before them in ConV and combining them into a single map as the result of this method. The efficiency of this method allowed the researchers and analysts to proceed with many other algorithms to achieve higher accuracy and better performance.

In the face detection technique used in [12,13], several attributes from the given input image can be used including face recognition, pose estimation, face expression, and pose estimation. It was a difficult task as every face had many changes in attributes like color, structures, etc. The most challenging aspect of this mission is to correctly identify the person's face in the image and then determine whether that person is hidden behind a mask or not. In order for the proposed system to be used for surveillance purposes, it is necessary for the system to detect the movement of a person's face and the mask.

Face mask detection using a smart city network was implemented for the whole city to ensure that every person in the society follows the rules [14]. The IoT (Internet of Things) concept was used along with the BlueDot and HealthMap services [15], while the use of automatic drones and cameras was proposed by [16] as a way to minimize the risk during COVID-19 spread. This allowed the government to easily manage and handle the crowds in all the public in a contactless manner.

The existing system has a camera that senses whether a person is wearing a mask or not and reports it to the person-in-charge to take an action. The system uses the Recurrent Neural Network (RNN) and Deep Neural Network (DNN) models, which compromised efficiency and accuracy [17]. The existing system had a very small training datasets, which caused the system to fail in meet the requirement needed by society. Moreover, this system was built using costlier materials, which makes it high-cost. The system has performed poorly with respect to accuracy, efficiency, and throughput.

Based on the literature analysis, the existing systems were able to automate the detection of masks and report to an in-charge on duty personnel with low accuracy and lots of work to be done.

3. Proposed Methodology

The methodology of the proposed system is comprised of two main steps: The first step is the creation of a face-matching model using deep learning and traditional machine learning techniques. The main challenge was to create a dataset that is composed of faces with and without face masks. A computer vision-based face detector was built using the created dataset, OpenCV, and Python with TensorFlow, withal in our custom machine learning framework. The computer vision and deep learning techniques were used to identify whether the person is wearing a face mask or not. This helps in expediting the proliferation of computer vision in the currently nascent areas such as digital signage, autonomous driving, video recognition, customer service, language translation, and mobile apps. The main element of deep learning is DNNs [18], which allows for object recognition segmentation. The proposed methodology utilize hybrid deep CNN classifier for segmenting the relevant features of face. DNNs are generally used in tasks related to computer vision as they act as an effective tool to increase the resolution of a classifier. Face recognition and classification models can be trained using CNN [19], advanced feature extraction, and classification methods to identify and classify facial images with minimal features and store fine details [19]. CNN is used to collect photos of people wearing face masks, rather than photos from a database, and then distinguish between the images of people wearing face masks and other people's photos based on facial expressions, content, and spatial information.

The primary purpose of using the Raspberry Pi [20] circuit board is to carry out critical tasks, such as the CPU, the GPU, input, and output. Raspberry Pi board features GPIO

pins are essential to using hardware programming to enable the Raspberry Pi to control electronic circuits and data processing devices on input and output data. It is possible to install and run the Raspberry Pi under the Raspbian OS and program it using the Python programming language. As a result, it can be a straightforward process to identify a person at the bus door or a station's entry point through an image/video stream using computer vision and deep learning techniques. If a person wearing a face mask enters the area, an automatic gate will open; if the person does not wear a mask, the gate will remain closed. The following subsections describe the face mask detection model and the operational technology in detail.

A. Face Mask Detection Model

Firstly, it is necessary to collect suitable examples of faces to feed the deep CNN classifier model so that it can determine if the individual in question is wearing a mask or not. Once the deep CNN classifier is trained, the face detection model needs to check for possible face-covering before classifying the individual since the Single Shot Multibox Detector and MobileNet (SSDMNV2) [21] evaluates whether the individual is wearing a mask or not. This research aims to improve the discrimination capability of masks without wasting significant computational resources; the DNN module from OpenCV uses the 'Single Shot MultiBox' (SSD) [22] object detection framework with ResNet-10 as its base. Our framework extends the features of the Raspberry Pi, such as live imaging, to occur in real-time. This deep CNN classifier expands on a trained model and independent network models to enable it to distinguish between a person who is wearing a mask or those who are not.

Several single-use, fixed image datasets are available for face detection only. Almost all the datasets can be considered fake in the absence of real-world information, and most of the existing ones suffer from the inclusion of incorrect information and noise. This required some effort to be done in order to identify the best possible dataset for the SSDMNV2 model. Kaggle and Witkowski's Medical Mask datasets [23] were utilized to expand the model's training datasets. In addition, data gathering was done using the masked dataset, which involved a blind application. The Kaggle dataset contains many individuals with faces blurred out to protect their privacy, as well as relevant XML files that describe their anonymity protection devices. The dataset holds a total of 678 photographs. PySearch for the expansion of the PyImage dataset in the Natural masking settings and return it as 'Prajary B'. The dataset consists of 1376 photos which are divided into two groups: those that have masks (686 images) and those that do not have masks (690). Few authors created a dataset by utilizing standard and identifying facial landmarks such as the eyes, brows, nose, mouth, and cheekbones, in addition to additional artificial points. Figure 1 shows the proposed face mask detection system.

Objects that are given as input are usually identified by their unique and specific features. There are many features and attributes in a human face. It can be used to recognize a face from any other objects around it in a given input. It identifies the faces by extracting given structural features like eyes, nose, mouth, ears, etc., and then uses these features to detect a face. Some classifiers would help to differentiate between facial and non-facial objects. Human faces will have specific features that can be used to find the differences between a face and other objects. In the next sub-section, we will be implementing a feature-based approach by using OpenCV, CNN (Convolution Neural Network), Keras, Tensor-flow. Overall, 96% validation accuracy has been attained during the CNN model training.

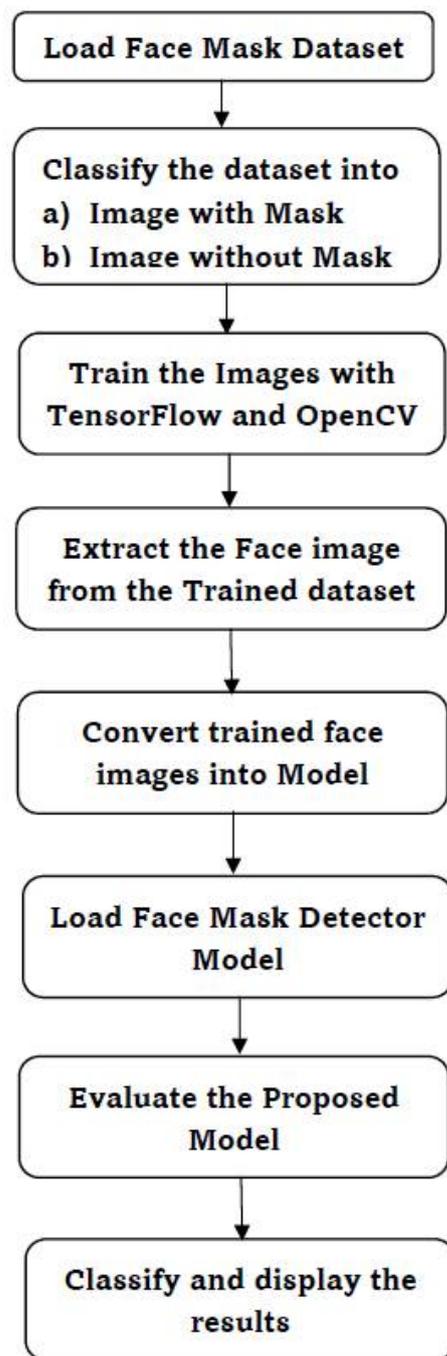


Figure 1. Proposed face mask detection system.

B. Operational Technology

The hardware module in Figure 2 recognizes whether a person wears a mask or not. People without masks can be detected and alerted using the app with any existing or IP cameras that have been connected to the system through network extenders. Additionally, users can add their faces and phone numbers to their expandable to be notified if they are missing a mask when they are out of the game. Administrators can send notifications to users if they believe that a user has not been adequately identified in the camera.

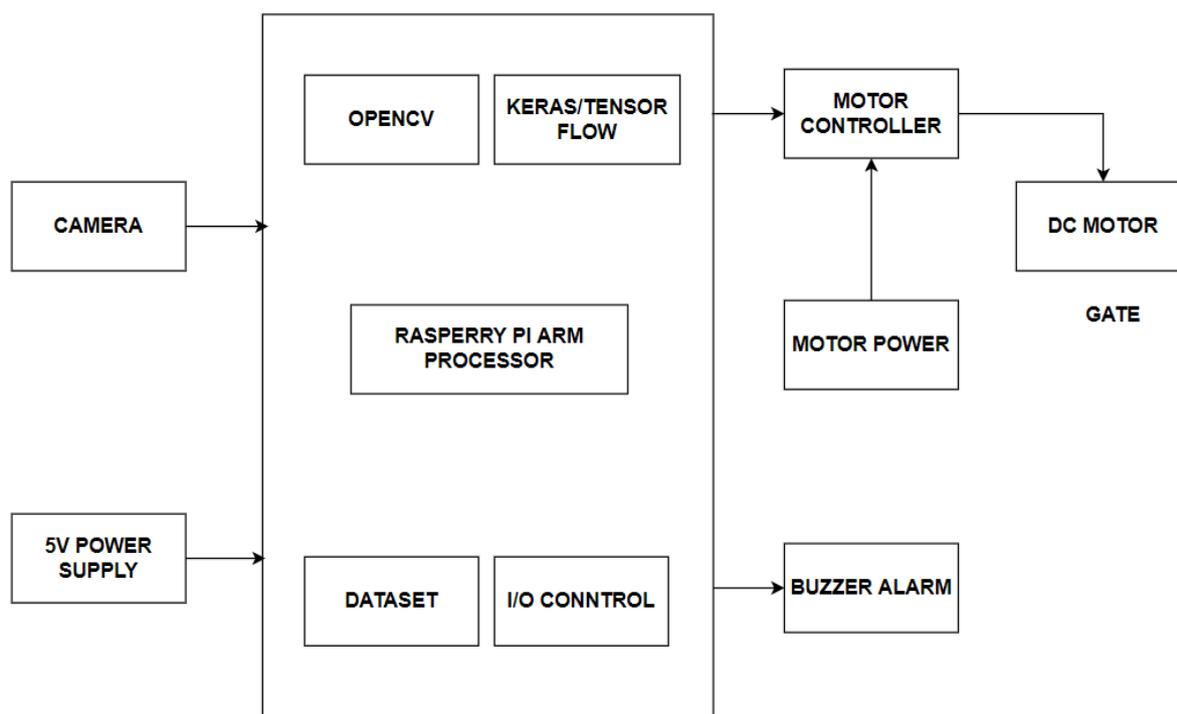


Figure 2. Hardware communication module for face mask detection.

With the help of a combination of deep learning and CNN [24] techniques, a real-time face mask detection system with an alert system has been developed. The image segmentation method produces efficient and accurate results for face detection. Face-reading showed that over half of the participants could accurately determine if the individual was wearing a mask or not. Algorithm 1 explains the methodology behind the face mask detection process.

Algorithm 1 Face Mask Detection Algorithm

- 1: *Input: Image Dataset with and without face mask*
 - 2: *Output: Classified Images with labels with and without mask*
 - 3: **for** each image in dataset **do**
 - 4: Create two categories for the image.
 - 5: Label each category according to mask with and without.
 - 6: Convert the RGB image to a grayscale image to a size 100×100 pixel.
 - 7: **if** face is detected **then**
 - 8: Contextually transform the image and integrate it to a four-dimensional array.
 - 9: Incorporate a Convolution layer with 200 filters to the mix.
 - 10: Incorporate the 2nd Convolution layer of 100 filters to the image.
 - 11: Add a Flatten surface to the deep CNN classifier to make it more accurate.
 - 12: Incorporate a dense layer of 64 neurons to the model.
 - 13: Incorporate the final Dense layer
 - 14: **if** mask is detected **then**
 - 15: Add the image to db Face with mask category
 - 16: **else**
 - 17: Add the image to db Face without mask category
 - 18: **end**
 - 19: **else if** face is not detected **then**
 - 20: Fall back to next image in dataset
 - 21: **end**
 - 22: **end**
-

4. System Implementation

Figure 3 shows the model of the proposed device for monitoring the people in the smart bus. The prototype model has the following kits: The model starts to work when connected to an AC supply, and it is developed considering its cost efficiency, size, and durability. The gate opens or closes depending on the person crossing it, with or without a mask, along with an alarm sound.

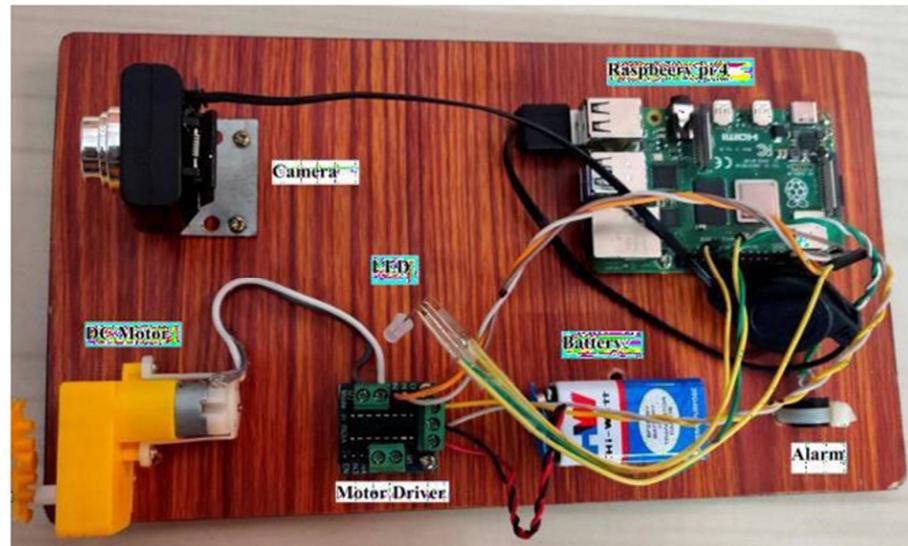


Figure 3. Implementation kit.

The proposed AI model has been incorporated into the Raspberry Pi kit by installing the necessary toolkits such as OpenCV, imutils and TensorFlow. Then, a new terminal is opened in Raspberry Pi for running the CNN with OpenCV toolkit. The preloaded images are then divided into two cross validation folds. The images will be classified into two groups, namely with and without a face mask, and are then stored in the Raspberry Pi. To pretrain the images, the sklearn and matplotlib packages in Raspberry Pi was used to execute the deep CNN classifier model. Finally, a web camera is connected to the Raspberry Pi for collecting real-time input of images with and without a face mask. These images can be dynamically trained in the Raspberry Pi kit using the Keras and Anaconda tools. The developed system would be installed in the entry spot of any public transportation. Whenever a passenger enters the transportation, the webcam would catch their facial image in real time using TensorFlow and OpenCV installed on the Raspberry Pi kit. These packages will detect whether a person is wearing a mask or not. If a person is wearing a mask in the right way, a green-colored box would appear around their face with a message saying “Thank you for wearing a mask”. Personnel who do not wear masks would have a red box around their face with a message stating “Alert !!! Wear a Mask”. The trained model in the Raspberry pi with the implementation of TenserFlow and the Visual Geometry Group (VGG16) Convolution system [25] showed accurate and efficient behavior. By analyzing regularly whether a person wears or does not wear a face mask to screen the Coronavirus, we can effectively assist with stopping the spread of the virus.

The proposed system supports real-time processing of the inputs and forces people to wear the face mask as per the guidelines. The proposed CNN model, which is extracted from the RNN model, helped us achieve a high accuracy of 97%. The rest of this section describes the steps taken by the system to recognize if a human being is wearing a mask or not.

Step 1: Data Visualization

As our training dataset contains many images, we would begin by plotting the images that fell into the most categories that we could find. There are approximately 686 images

with a face mask that have been marked as “yes” in the database and approximately 690 photographs of people without face masks are been marked “no,”.

Step 2: Data Augmentation

For data augmentation, more images were added to the dataset and each of the images was rotated as we continue through this step. Using the data augmentation method, we had 1376 images, with 686 images falling under the ‘yes’ category and 690 images falling under the ‘no’ category.

Step 3: Splitting the data

The dataset is split into two sets: 80% represent the training dataset for the Convolution layer and 20% represent the validation dataset for the proposed method. Images with facial mask in training dataset: 686.

- Images with facial mask in training dataset: 686;
- Images with facial masks in the validation dataset: 140;
- Images without facial mask in training set: 690;
- Images without facial masks in the validation dataset: 138.

Step 4: Modeling

Build our CNN model using layers like Conv2D, MaxPooling2D, Flatten, Dropout, and Dense. The ‘Softmax’ function outputs a vector with probabilities for each class in the final Dense layer. With only two categories, the Adam optimizer and the binary-cross entropy loss function were used.

Step 5: Validating the model

This step aims to fit the images from the training and validation datasets to the Sequential model with 30 epochs (iterations). Moreover, it is possible to train with more epochs to improve accuracy without over-fitting. The overfitting of data is avoided by utilizing cross-validation method. Here, the training data is divided into two folds and each fold is trained one at a time. In this manner, close correspondence of data is regularized for future prediction with new additional data.

```
Epochs = 30, validation data = validation generator, callbacks = [checkpoint]) >>
30/30, 220/220 [=====]-231 s 1 s/step-loss: 0.03680-acc: 0.98860.
```

The above code is the output obtained after 30 epochs. For executing 30 epochs, our model has taken 231 s with loss of 0.3%. Our model has 98.86% accuracy with the training dataset and 96.19% accuracy with the validation set. The above accuracy shows that this model is well-trained and does not cause overfitting.

Step 6: Categorizing Data

Once the model is developed, users label two different probabilities. «without facial mask» and «with facial mask» along with RGB values that are used to color the rectangle edges. [RED for no facial mask and GREEN for facial mask]

```
labels_dict = {0:'without_mask',1:'with_mask'}
color_dict = {0:(0,0,255),1:(0,255,0)}
```

The first code represents the labeling component of the output image. If the image is returned with classifier values as 0, then it is labeled under without mask component. If the image is returned with classifier value 1, then the image is labeled under with mask component. The second line of code represents the color value of the rectangular component, which is used to represent the face mask identifier in the input image.

Step 7: Face Detection Program Import

Here, the PC’s webcam is used to see if a face mask is worn or not. Initially, face detection program was implemented and the Haar feature-based sequence classifiers are used to detect facial characteristics. `face_cascade = cv2.CascadeClassifier('haarcascade_frontalface_default.xml')`

With the help of the OpenCV programming language, OpenCV generated this cascade deep CNN classifier that was used to diagnose the frontal face from thousands of images.

Step 8: Detecting Masked and Unmasked Faces

A Classification Algorithm detects a facial expression that uses the Software in the final step. `webcam = cv2.VideoCapture(0)` signifies webcam usage. The prototype would

then predict the likelihood of each class ([without a facial mask, with a facial mask]). The identifier would be selected and presented around our face images based on the probability.

The dataset created for the implementation is depicted in Figures 4 and 5. The training dataset has 1386 images and the validation dataset has 278 images. The predefined condition for entering a public transportation during this pandemic period is mandatory wearing of mask. Our DB system is constructed with images of people with mask, without mask and partially wearing mask. In case of images with masks, but not covered the nose, they will be considered as NO case. This is implemented for the sake of 100% accuracy in ethical face mask detection. Table 1 depicts the performance analysis of the proposed system with deep CNN classifier.



Figure 4. Dataset without mask.



Figure 5. Dataset with mask.

Table 1. Performance metrics.

	Precision	Recall	F1-Score	Support
Without face mask	0.960	1.000	0.980	69
With face mask	1.000	0.900	0.950	36
Accuracy	-	-	0.970	106

The proposed system produces the output as seen in Figure 6, which can investigate a person’s appearance whether he/she wore a mask on the bus. If the system detects that someone has not worn a mask on the entrance of the Smart Bus, the output is in Figure 7 will be of them.

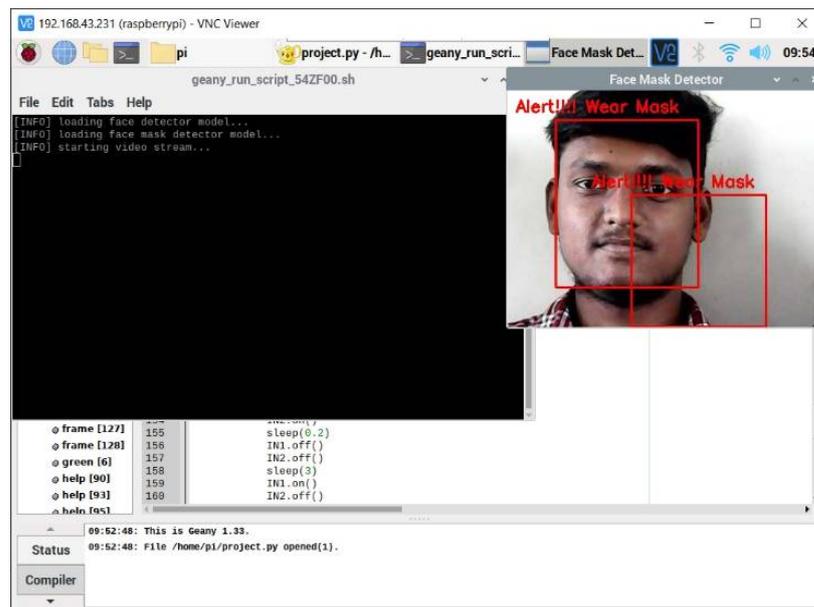


Figure 6. Person without mask.

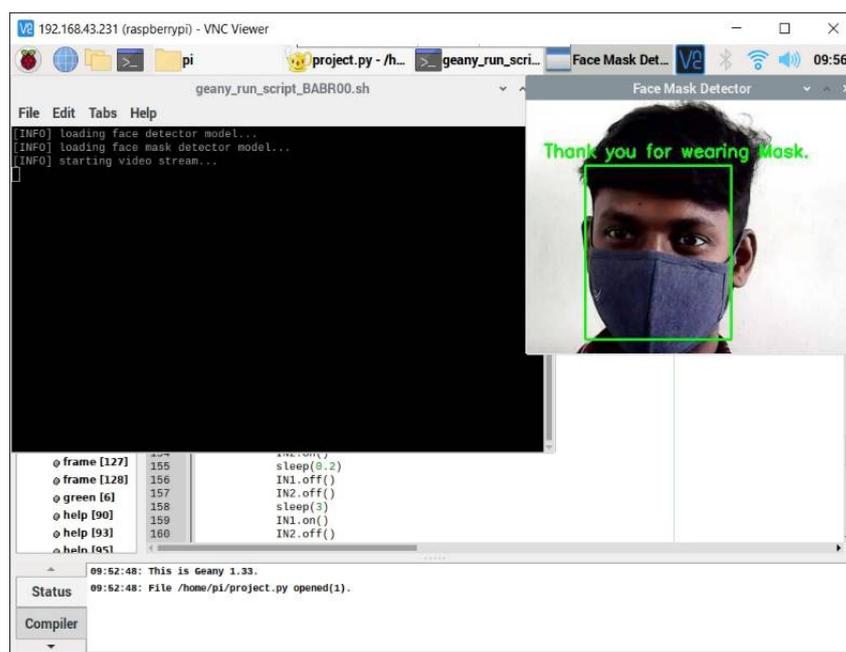


Figure 7. Person with mask.

5. Performance Discussion and Comparison

The performance of the proposed face mask detection system is measured and compared to other existing systems in terms of error rate, inference time, correlation coefficient, data over-fitting analysis, precision, and recall [26,27].

- Error rate: This type of error occurs most frequently when the most confidently predicted class does not match the actual class. Inference Time on CPU: The model takes time to figure out how to determine what type of image an input one is. Everything is covered, from reading the image to performing all intermediate transformations to arriving at the final class with a high degree of confidence.
- Correlation coefficient: The ratio between the ground truth visibility score and the predicted mask identification score is displayed in the correlation graph.
- Data over-fitting analysis: The relationship between the training and testing period in loss and accuracy is measured as the correlation coefficient.
- Precision: The ratio of the predicted number of accurate face mask identification to the total number of actual face mask cases.
- Recall: The ratio of predicted counts of face mask identified images to the total number of relevant face mask images.

Figure 8 depicts a graph that illustrates how accurate the different models are based on the number of mistakes in a given period. The graph demonstrates that the error rate in AlexNet [28] is extremely high, whereas the error rate in the proposed model is extremely low, as shown in the table.

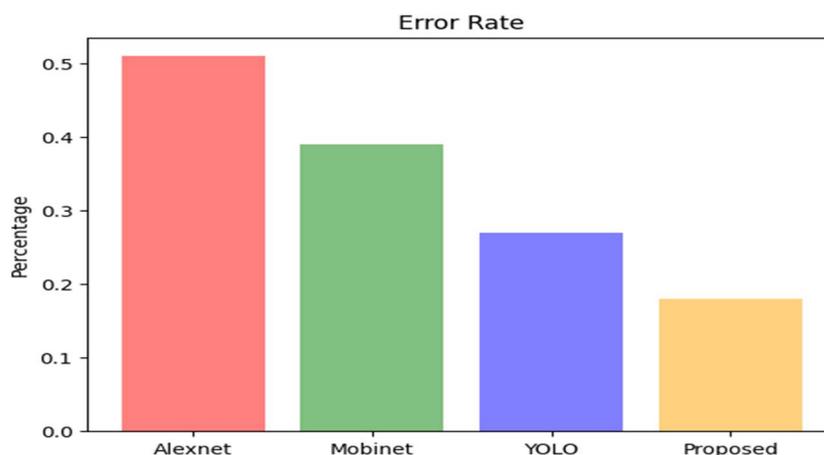


Figure 8. Error rate.

Following that, we compared the inference times of the different models. Iterations include the provision of test images to each model and the calculation of the average inference time over all iterations. As shown in Figure 9, the proposed system classifies the images in less time than MobiNet [29] and other models, which is a significant improvement.

In order to evaluate the relationship between the predicted image complexity score and ground truth visual difficulty score, we compute Spearman's rank correlation coefficients between the two scores. Our investigation makes use of Spearman's rank correlation coefficient, which is an appropriate measure because it is invariant across a wide range of scoring methods. It is possible to compute the Spearman's rank correlation coefficient in Python by utilizing the Spearman () SciPy function. Return the correlation coefficient after calculating it with two scores as input and returning the correlation coefficient. Our predictor has a Spearman's rank correlation coefficient ϵ of 0.851, which indicates that it performs exceptionally well when it comes to predicting the complexity of images. There is a significant correlation between the ground truth and predicted complexity scores, as illustrated in Figure 10.

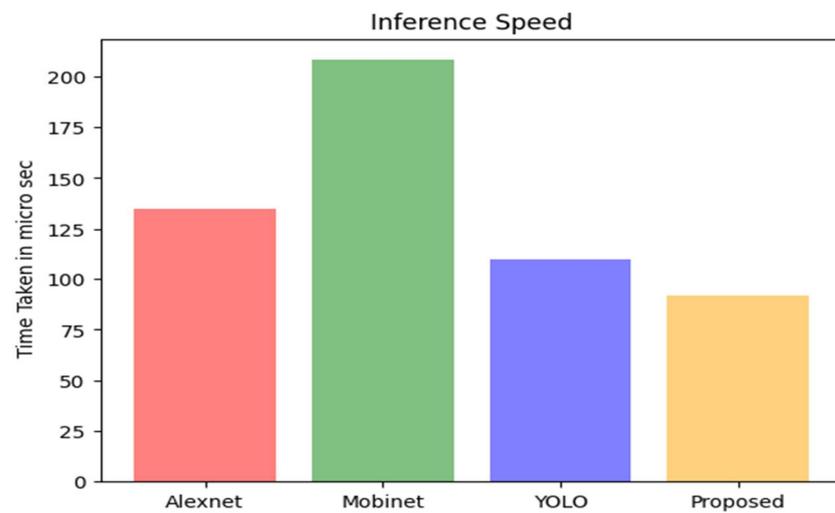


Figure 9. Inference speed.

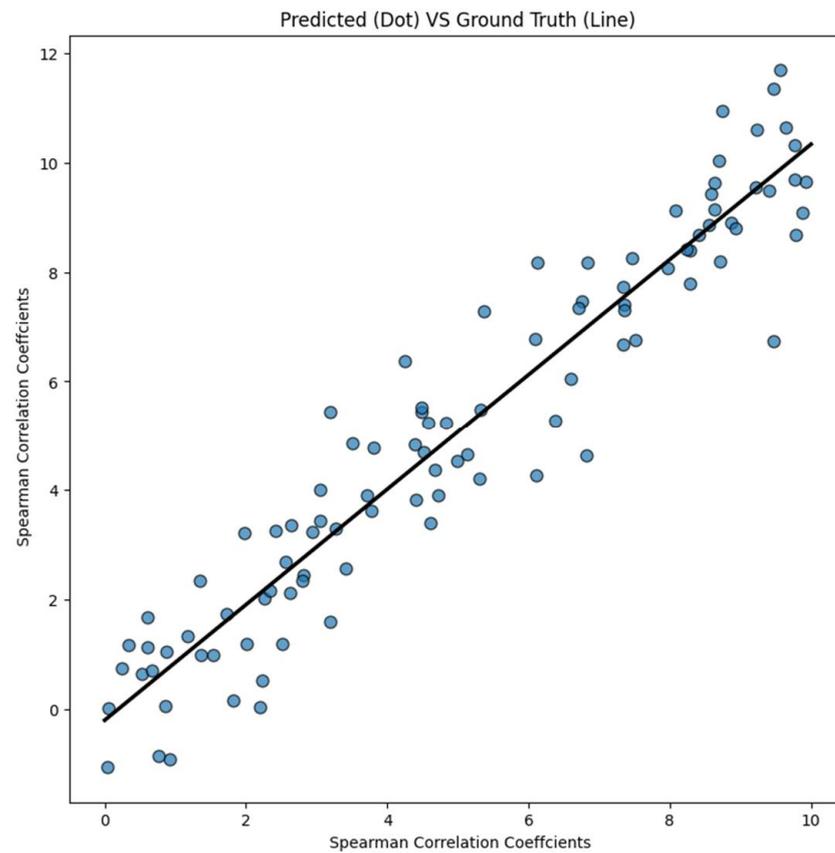


Figure 10. Ground truth vs. predicted in face mask.

The proposed model's data efficiency is determined by comparing the loss and precision of the training and validation runs. It is critical to monitor the model's accuracy and loss of precision during both the training and testing phases, which last a total of 20 epochs. Aside from that, as depicted graphically in Figure 11, the model's accuracy improves with time and becomes stable after epoch = 2.

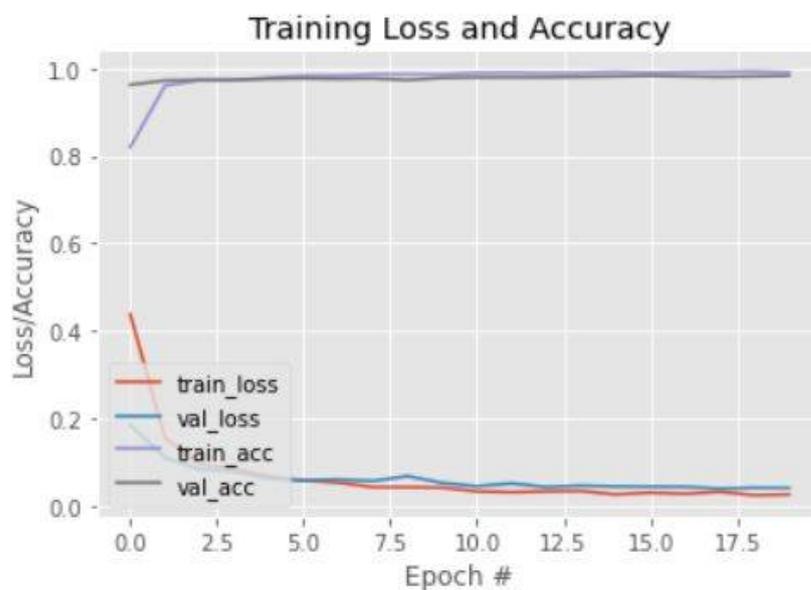


Figure 11. Training vs. validation.

To further prove the quality of the proposed model, we compared in terms of accuracy, precision and recall for detecting images for humans with and without a mask to be publicly available as baseline models similar to Alexnet, Mobinet, and YOLO. The results of this comparison are summarized in Table 2 showing that the proposed system outperforms other models in terms of accuracy, precision and recall.

Table 2. Models comparison.

Model	Without Mask			With Mask		
	Precision %	Recall %	Accuracy %	Precision %	Recall %	Accuracy %
Alexnet [28]	81.27	92.13	98.34	80.63	88.54	89.67
MobiNet [29]	90.08	94.61	98.17	93.45	93.67	94.56
YOLO [29]	96.61	97.26	99.68	97.27	97.36	98.34
Proposed	97.82	99.05	99.73	98.13	98.91	99.07

The experiments show that the proposed system detects faces and masks accurately while consuming less inference time and memory than previously developed techniques. To address the data imbalance problem that had been identified in the previously published dataset, efforts were made to create an entirely new, unbiased dataset that is particularly well-suited for COVID-19 related mask detection tasks, among other things. More accurate face detection, precise localization of the individual's identity, and avoidance of overfitting were all essential factors in developing an overall system that can be easily embedded in a device that can exist in public places to assist with the prevention of COVID-19 transmission.

6. Conclusions

COVID-19, the prevailing virus outbreak, has made us recognize the benefits of face masks. The use of face masks is essential when using public transportation. In this research, a face mask detection system based on deep CNN classifiers and the VGG16 model is demonstrated with the deployment in a public transportation system. The model is implemented with Raspberry Pi kit and open-source data analytics toolkits. A real-time dataset has been generated with a collection of 1386 images. This database comprises 686 images with facial masks and 690 images without a facial mask. The proposed model was compared with existing face mask detection frameworks like Alexnet, Mobinet and YOLO. The performance metrics such as error rate, inference speed, precision, recall,

accuracy, and overfitting analysis were performed. The results reveal that the proposed system has outperformed the existing models with an accuracy over 99% and an error rate less than 2%. In the future, our face mask detection system can be employed at air terminals, shopping malls, and other traffic places to detect and strengthen the importance of face mask-wearing.

Author Contributions: Conceptualization, Methodology, and Supervision, T.A.K.; Resources, Software, Project administration, R.R.; Methodology, Conceptualization, Resources, formal analysis and Writhing of original draft, M.P., S.A.A.; Validation and Data curation M.P.; Investigation, Editing and review S.A.A., Resources, Editing, and review, Supervision, R.H., T.G. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: Not applicable.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Panakaje, N.; Rahiman, H.U.; Rabbani, M.R.; Kulal, A.; Pandavarakallu, M.T.; Irfana, S. COVID-19 and its impact on educational environment in India. *Environ. Sci. Pollut. Res.* **2022**, 1–17. [[CrossRef](#)]
2. Al-Amer, R.; Maneze, D.; Everett, B.; Montayre, J.; Villarosa, A.R.; Dwekat, E.; Salamonson, Y. COVID-19 vaccination intention in the first year of the pandemic: A systematic review. *J. Clin. Nurs.* **2022**, *31*, 62–86. [[CrossRef](#)]
3. Coccia, M. Preparedness of countries to face COVID-19 pandemic crisis: Strategic positioning and factors supporting effective strategies of prevention of pandemic threats. *Environ. Res.* **2022**, *203*, 111678. [[CrossRef](#)] [[PubMed](#)]
4. Pan, L.; Wang, J.; Wang, X.; Ji, J.S.; Ye, D.; Shen, J.; Li, L.; Liu, H.; Zhang, L.; Shi, X.; et al. Prevention and control of coronavirus disease 2019 (COVID-19) in public places. *Environ. Pollut.* **2022**, *292*, 118273. [[CrossRef](#)] [[PubMed](#)]
5. Runde, D.P.; Harland, K.K.; Van Heukelom, P.; Faine, B.; O’Shaughnessy, P.; Mohr, N.M. The “double eights mask brace” improves the fit and protection of a basic surgical mask amidst COVID-19 pandemic. *J. Am. Coll. Emerg. Physicians* **2021**, *2*, e12335. [[CrossRef](#)] [[PubMed](#)]
6. Ajagbe, S.A.; Amuda, K.A.; Oladipupo, M.A.; Afe, O.F.; Okesola, K.I. Multi-classification of alzheimer disease on magnetic resonance images (MRI) using deep convolutional neural network (DCNN) approaches. *Int. J. Adv. Comput. Res.* **2021**, *11*, 51–60. [[CrossRef](#)]
7. Awotunde, J.B.; Ajagbe, S.A.; Oladipupo, M.A.; Awokola, J.A.; Afolabi, O.S.; Mathew, T.O.; Oguns, Y.J. An Improved Machine Learnings Diagnosis Technique for COVID-19 Pandemic Using Chest X-ray Images. In *Applied Informatics. ICAI 2021. Communications in Computer and Information Science*; Florez, H., Pollo-Cattaneo, M.F., Eds.; Springer: Cham, Switzerland, 2021; Volume 1455. [[CrossRef](#)]
8. Machingaidze, S.; Wiysonge, C.S. Understanding COVID-19 vaccine hesitancy. *Nat. Med.* **2021**, *27*, 1338–1339. [[CrossRef](#)] [[PubMed](#)]
9. Chen, J.; Li, K.; Zhang, Z.; Li, K.; Yu, P.S. A Survey on Applications of Artificial Intelligence in Fighting against COVID-19. *ACM Comput. Surv.* **2021**, *54*, 1–32. [[CrossRef](#)]
10. Raftarai, A.; Mahounaki, R.R.; Harouni, M.; Karimi, M.; Olghoran, S.K. Predictive Models of Hospital Readmission Rate Using the Improved AdaBoost in COVID-19. In *Intelligent Computing Applications for COVID-19*; CRC Press: Boca Raton, FL, USA, 2021; pp. 67–86.
11. Mahdi, M.S.; Abid, Y.M.; Omran, A.H.; Abdul-Majeed, G. A Novel Aided Diagnosis Schema for COVID 19 Using Convolution Neural Network. *IOP Conf. Ser. Mater. Sci. Eng.* **2021**, *1051*, 012007. [[CrossRef](#)]
12. Kumar, A.; Kaur, A.; Kumar, M. Face detection techniques: A review. *Artif. Intell. Rev.* **2019**, *52*, 927–948. [[CrossRef](#)]
13. Khodabakhsh, A.; Ramachandra, R.; Raja, K.; Wasnik, P.; Busch, C. Fake Face Detection Methods: Can They Be Generalized? In Proceedings of the 2018 International Conference of the Biometrics Special Interest Group (BIOSIG), Darmstadt, Germany, 26–28 September 2018; Institute of Electrical and Electronics Engineers: Piscataway, NJ, USA, 2018; pp. 1–6.
14. Rahman, M.M.; Manik, M.M.H.; Islam, M.M.; Mahmud, S.; Kim, J.-H. An Automated System to Limit COVID-19 Using Facial Mask Detection in Smart City Network. In Proceedings of the 2020 IEEE International IoT, Electronics and Mechatronics Conference (IEMTRONICS), Vancouver, BC, Canada, 9–12 September 2020; pp. 1–5.
15. Choubey, S.K.; Naman, H. A Review on Use of Data Science for Visualization and Prediction of the COVID-19 Pandemic and Early Diagnosis of COVID-19 Using Machine Learning Models. In *Internet of Medical Things for Smart Healthcare*; Springer: Singapore, 2020; pp. 241–265.
16. Jaiswal, A.; Gianchandani, N.; Singh, D.; Kumar, V.; Kaur, M. Classification of the COVID-19 infected patients using DenseNet201 based deep transfer learning. *J. Biomol. Struct. Dyn.* **2020**, *39*, 5682–5689. [[CrossRef](#)] [[PubMed](#)]

17. Li, X.; Wang, J.; Li, C.; Wang, Z.; Zhang, J. Predicting the Number of COVID-19 Cases Based on Deep Learning Methods. In Proceedings of the 2021 IEEE 4th International Conference on Big Data and Artificial Intelligence (BDAI), Qingdao, China, 2–4 July 2021; Institute of Electrical and Electronics Engineers: Piscataway, NJ, USA, 2021; pp. 37–42.
18. Vu, T.H.; Dang, A.; Wang, J.-C. A Deep Neural Network for Real-Time Driver Drowsiness Detection. *IEICE Trans. Inf. Syst.* **2019**, *E102.D*, 2637–2641. [[CrossRef](#)]
19. Bargshady, G.; Soar, J.; Zhou, X.; Deo, R.C.; Whittaker, F.; Wang, H. A Joint Deep Neural Network Model for Pain Recognition from Face. In Proceedings of the 2019 IEEE 4th International Conference on Computer and Communication Systems (ICCCS), Singapore, 23–25 February 2019; Institute of Electrical and Electronics Engineers: Piscataway, NJ, USA, 2019; pp. 52–56.
20. Wazwaz, A.A.; Herbawi, A.O.; Teeti, M.J.; Hmeed, S.Y. Raspberry Pi and Computers-Based Face Detection and Recognition System. In Proceedings of the 2018 4th International Conference on Computer and Technology Applications (ICCTA), Istanbul, Turkey, 3–5 May 2018; Institute of Electrical and Electronics Engineers: Piscataway, NJ, USA, 2018; pp. 171–174.
21. Nagrath, P.; Jain, R.; Madan, A.; Arora, R.; Kataria, P.; Hemanth, J. SSDMNV2: A real time DNN-based face mask detection system using single shot multibox detector and MobileNetV2. *Sustain. Cities Soc.* **2021**, *66*, 102692. [[CrossRef](#)] [[PubMed](#)]
22. Shyam, D.; Kot, A.; Athalye, C. Abandoned Object Detection Using Pixel-Based Finite State Machine and Single Shot Multibox Detector. In Proceedings of the 2018 IEEE International Conference on Multimedia and Expo (ICME), San Diego, CA, USA, 23–27 July 2018; Institute of Electrical and Electronics Engineers: Piscataway, NJ, USA, 2018; pp. 1–6.
23. Gorhekar, R.; Shah, C.; Shah, V.; Bide, P.J. A Survey on Covid-19 Face-Mask Detection Techniques. In Proceedings of the 2021 IEEE 4th International Conference on Computing, Power and Communication Technologies (GUCON), Kuala Lumpur, Malaysia, 24–26 September 2021; Institute of Electrical and Electronics Engineers: Piscataway, NJ, USA, 2021; pp. 1–6.
24. Sethi, S.; Kathuria, M.; Kaushik, T. Face mask detection using deep learning: An approach to reduce risk of Coronavirus spread. *J. Biomed. Inform.* **2021**, *120*, 103848. [[CrossRef](#)] [[PubMed](#)]
25. Tomás, J.; Rego, A.; Viciano-Tudela, S.; Lloret, J. Incorrect Facemask-Wearing Detection Using Convolutional Neural Networks with Transfer Learning. *Healthcare* **2021**, *9*, 1050. [[CrossRef](#)] [[PubMed](#)]
26. Chowdary, G.J.; Punn, N.S.; Sonbhadra, S.K.; Agarwal, S. Face Mask Detection Using Transfer Learning of Inceptionv3. In Proceedings of the International Conference on Big Data Analytics, Sonapat, India, 15–18 December 2020; Springer: Cham, Switzerland, 2020; pp. 81–90.
27. Shahin, M.K.; Tharwat, A.; Gaber, T.; Hassanien, A.E. A wheelchair control system using human-machine interaction: Single-modal and multimodal approaches. *J. Intell. Syst.* **2019**, *28*, 115–132. [[CrossRef](#)]
28. Song, Z.; Nguyen, K.; Nguyen, T.; Cho, C.; Gao, J. Camera-Based Security Check for Face Mask Detection Using Deep Learning. In Proceedings of the 2021 IEEE Seventh International Conference on Big Data Computing Service and Applications (BigDataService), Oxford, UK, 23–26 August 2021; Institute of Electrical and Electronics Engineers: Piscataway, NJ, USA, 2021; pp. 96–106.
29. Aadithya, V.; Balakumar, S.; Bavishprasath, M.; Raghul, M.; Malathi, P. Comparative Study between MobilNet Face-Mask Detector and YOLOv3 Face-Mask Detector. In *Sustainable Communication Networks and Application*; Springer: Singapore, 2022; pp. 801–809.