Global Transitions Proceedings 2 (2021) 246–254

Contents lists available at ScienceDirect

Global Transitions Proceedings

journal homepage: http://www.keaipublishing.com/en/journals/global-transitions-proceedings/

IoT-Enabled smart doors for monitoring body temperature and face mask detection

B Varshini, HR Yogesh, Syed Danish Pasha, Maaz Suhail, V Madhumitha, Archana Sasi*

Dept. of CSE, Presidency University, Bengaluru, Karnataka, India

ARTICLE INFO

Keywords:
COVID-19
Face mask detection
Machine learning
Raspberry Pi
Sensors
Temperature detection
TensorFlow

ABSTRACT

COVID 19 pandemic is causing a global health epidemic. The most powerful safety tool is wearing a face mask in public places and everywhere else. The COVID 19 outbreak forced governments around the world to implement lockdowns to deter virus transmission. According to survey reports, wearing a face mask at public places reduces the risk of transmission significantly. In this paper, an IoT-enabled smart door that uses a machine learning model for monitoring body temperature and face mask detection. The proposed model can be used for any shopping mall, hotel, apartment entrance, etc. As an outcome a cost-effective and reliable method of using AI and sensors to build a healthy environment. Evaluation of the proposed framework is done by the Face Mask Detection algorithm using the TensorFlow software library. Besides, the body temperature of the individual is monitored using a non-contact temperature sensor. This proposed system can detect the users from COVID 19 by enabling the Internet of Things (IoT) technology.

1. Introduction

The coronavirus disease, or COVID-19, which originated primarily in Wuhan, China, has rapidly spread to several countries, including India, the world's second-most populous country with a population of more than 134 billion people [20-22]. With such a large population, India would have trouble preventing the spread of the coronavirus. Face masks and sanitizers are the most effective ways to minimize transmission. When it comes to reducing disease transmission, this has shown good results. Fever, sore throat, tiredness, loss of taste and smell, and nasal congestion are all common symptoms of coronavirus infection. The majority of the time, it is transmitted indirectly through surfaces. The incubation period can be very long, ranging from 10 to14 days in extreme cases, and the virus can attack directly (from one individual to other individuals) by respiratory droplets [2]. Governments implemented a variety of protection and safety initiatives to reduce disease transmission, including social distancing, mandatory indoor mask-wearing, quarantine, restricting citizens' traveling within state boundaries and abroad, self-isolation, and the exclusion and cancellation of big social occasions and meetings [10]. From work activities to social relationships, all kinds of sports activities, as well as off-screen and on-screen entertainment have all been affected due to this COVID-19 pandemic [4]. Individuals with high body temperature are not to be permitted to enter public places because they are at a high risk of infection and spreading the virus; wearing a mask is essential. At the entrances to any city, workplaces, malls, and hospital gates, temperature and mask checks are also necessary. As a result, a smart entry device that automatically monitors human body temperature and detects a mask at the door opening system is developed. An advanced idea is used in this system approach, which is a combination of all three including temperature detection, total people count, and mask detection.

The next part of this paper is structured as follows. Section 2 briefly describes the related works. Explanation about the proposed work is given in Section 3. Section 4 describes the working methodology and a detailed explanation of model implementation. The results and discussion about the working model is provided in Section 5. And the last section of the paper draws the conclusion and future work.

2. Related Work

The importance of body temperature assessment in clinic diagnosis and therapies cannot be overstated [23-25]. There are some drawbacks, including low measurement accuracy and a long measurement period. Traditional artificial measurement methods make it difficult to track patient body temperature in a timely manner automatically and accurately. To address the above problem, they presented a distributed monitor system that is used for measuring body temperature. Multi-temperature sensors, such as the DS18B20, were attached and are used to capture a person's body temperature signal, after which the SCM AT89C52 processed the signal. They use the nRF905 wireless transceiver chip to complete the signal wirelessly from the work station

* Corresponding author.
E-mail address: archana.sasi2k8@gmail.com (A. Sasi).

https://doi.org/10.1016/j.gltp.2021.08.071
Received 10 June 2021; Accepted 2 July 2021
Available online 12 August 2021

known as slave station to a central station, a USB adapter PDIUSBD12 to link the upper PC. Since the temperature calculated errors are less than plusmn0.1degC, this system showed that the device with wireless communication is much better, and it meets the clinic's medical requirements well. It can be transplanted into another sector, namely greenhouse environment intelligent monitor, with the help of the system's modularization design.

Real-time data collection is critical in the field of human health [11]. This paper outlines a method for tracking a human being's heartbeat rate per sec and detecting normal body temperature from a distance. The information was gathered from a group of volunteers, and the device was tested with sensors created by the research team. The Arduino microcontroller is designed to send data over the XBee wireless network to a remote PC station for display and storage. It reduces device power consumption by activating the sensor with a remote-control command from the receiving PC [26-28].

The performance of a Wireless Sensor Network in rising applications including weapon sensor ships, medical applications, habitat monitoring, and seismic surveillance was investigated [5, 17]. WSN has recently concentrated on domestic sequences and market applications. The efficiency of the PIC-created WSN models is demonstrated in this project. To establish sensing phenomena, the normal nodes of temperature sensors were used for networks. Here, findings show that the time setting has a significant impact on the sensor node's efficiency. The purpose of this document is to identify and briefly explain the critical factors and issues that affect WSN output.

Controlling laboratory measurements and clinical trials limit the realism and duration of various tests [3]. Tracking the influence of sleep deprivation on regular intervals known as circadian rhythms throughout the human body, for example, necessitates extremely precise profiling of skin temperature across the human body over several weeks, with real-time input from a remote clinician. They investigated the necessities for applications in the wearable sensors and emphasized the importance of personality behavior, like adaptive sampling to increase service energy-saving, adaptive strategy development, automatic atmospheric compensation, and automatic logging. They have developed and constructed a prototype of a wireless non-invasive monitor system that measures the body's precise temperature and provides real-time feedback to the doctor. They achieved an accuracy of 0.02°C by designing, parameterizing, and calibrating an active measuring subsystem that covers the average 16 - 42°C range of body temperature that has the consequence on skin temperature of circadian and mental rhythms based on two initial research. They found that their procedure has the potential for becoming a valuable medical research advantage.

Biometric individual reconnaissance systems are used to provide secure alternatives [14]. Although various biometric recognition methods and algorithms have been developed and published in the literature, no research into the correlations between biometrics has been conducted. In this study, they looked into whether biometric characteristics are linked to individuals attempting to extract a biometric feature from another biometric characteristic of the same individual. As a result, they developed and released a new smart frame that uses a new artificial neural network approach to generate fingerprint face masks with absolute percent-age errors ranging from 0.75 to 3.60. Experiments have shown that fingerprints can be used to create facial masks without prior awareness of the facets. Furthermore, fingerprints and faces have been shown to have a close relationship. Although the system is still in its early stages, the findings are very positive and hopeful.

By conducting tasks like real-time incident tracking and post-event analysis, video analytics improve video surveillance services [6]. Humans will save time and money, while the surveillance system's effectiveness will improve. One of the most common security standards for video analytics is to detect the presence of a person with a mask automatically. In this document, a four-part detection and eye detection method for masked face detection was suggested. The paper explains the concepts behind each of these procedures, as well as the use of commonly accessible people detection and face detection algorithms. This novel approach to the problem resulted in a less complicated solution that can be applied in real-time. The algorithm's success on test video sequences provides valuable information for improving masked face detection efficiency.

In the fields of facial recognition and computer vision, face mask detection has made significant progress [16]. A variety of techniques and algorithms were used to construct face detection models. The proposed approach in this project leverages profound intelligence, TensorFlow, Keras, and OpenCV to detect face masks. This method can be utilized in defense because it is low-cost to adopt. This technique does have a precision score of 0. 9264 and an F1 rating of 0.93.

The Haar Cascade algorithm to detect facets in the low-cost Internet of Things using the Raspberry Pi method was used [13]. It is a cutting-edge access control scheme. It shows a machine learning approach for facial recognition and detection that makes use of the OpenCV library's hair cascade to complete the task quickly and with a high detection rate. Face recognition is a way of recognizing and verifying an individual's identity by looking at their face. The Python programming language is used to make modifications to the framework. A grey and a colored picture of the faces are differentiated by the pro-positive style. The framework's effectiveness is calculated by measuring the face recognition rate for each individual in the database. The proposed system's findings can be used to accurately distinguish faces even from low-quality images.

[4] This paper offers a simple and low price IoT node, the mobile device, and fog-based machine learning (ML) instruments for statistical analyses and diagnostics. The IoT node analyzes the saturation of blood oxygen, respirator, toxicity, rate, and body temperature before updating the mobile app to show users' current health status. To prevent the virus from spreading, this app tells the user to maintain 2 meters' physical distance (or 6 meters). A Fuzzy Mamdani (running on a fog server) system also takes into account the potentially harmful environmental and user health when it calculates the risk of infection spread in real-time. The virtual zone's concept transmits environmental risk and offers up-to-date information for multiple locations. For different event scenarios, the energy consumption and required bandwidth (BW) are compared.

Face masks are becoming more popular in public due to the global outbreak of the coronavirus COVID-19 [19]. Before Covid-19, people wore masks as air pollution protective measures to protect their welfare. A few of them cover faces, and others are conscious of their look, to conceal their feelings from the public. According to scientists, wearing face masks slows COVID-19 transmission. The most recent influenza virus to strike human health in the 20th century is COVID19. The World Health Organization (WHO) proclaimed it a global pandemic in 2020 because of its rapid expansion. In under six months, COVID-19 infected over five million people in 188 countries. The coronavirus outbreak prompted unprecedented levels of international scientific collaboration. Artificial intelligence (AI) focused on machine learning and deep learning assists in the battle against Covid-19 in a variety of ways. Machine learning can help researchers and clinicians predict the spread of COVID-19, serve as an early warning system for pandemics, and identify vulnerable populations by evaluating vast amounts of data.

The role of data-driven mobile applications in combating the COVID-19 pandemic is examined [20]. Innovative case studies demonstrate two indoor safety monitoring and resource planning as evidence of practice during a serious pandemic. The corresponding multiplatform mobile applications were built using the App Sheet Framework, which automates the development of Google Sheets as a data source.

Unless the situation changes today, institutions such as the academy are at risk of closing down in light of the COVID-19 pandemic [15]. COVID 19 is a virus that causes serious respirational problems, also called Serous Acute Respiratory Syndrome. Corona virus-2 is a contagious disease that is transmitted through respiratory droplets from an individual who speaks, sneezes, or coughs. It is easy to spread, due to close contact with infected individuals and contact with infected objects or surfaces. Because COVID-19 vaccines are not widely available at the moment, the only way to protect ourselves is to avoid infection.

Implementation of facemask detection with alarm systems for physical distancing utilizing deep learning technique using CNN is discussed in [18, 12]. The researchers introduced a high accuracy strategy for detecting facial masks based on fully convolutional networks, gradient descent, and binomial cross-entropy by using semantic segmentation. The use of CNN to improve the accuracy and speed of the cultivar's recognition was devised to classify the numerous cultivars of Durio zibethinus (or commonly known as durian) based on the crop's visual structures [10]. Production of pulse oximetry kits using Internet of Things (IoT) [2, 9] technology as instruments for monitoring of covid-19 patients remotely via smartphones in terms of physical and social distancing protocols was utilized to track the body temperature of the individuals.

3. Proposed Approach

3.1. Software Requirements

3.1.1. TensorFlow

TensorFlow is a machine learning software library that is open source and free. It was created to perform large numerical computations without regard for deep learning. This TensorFlow can be used for a variety of activities, but it is primarily focused on deep neural network inference and training. TensorFlow also supports traditional machine learning. Google's TensorFlow is a Python library that allows for quick numerical computation. Deep learning models are either generated directly using TensorFlow, which is also a base library, or they are created to simplify the process by using wrapper libraries built on top of TensorFlow. TensorFlow enables the creation of dataflow graphs and structures to determine how the data flows through the graph by receiving inputs as a multi-dimensional tensor array. It allows building a flow chart for these inputs which is carried out on the one end and is performed on the other.

3.1.2. PuTTy

PuTTY is an open-source software that provides both serial console and software transfer for network files. . It supports a wide range of network protocols, such as SCP, Telnet, SSH, rlogin, and a crude socket connection which can also be connected to a serial port. PuTTy is a terminal emulator which enables users to access the Raspberry Pi command-line interface from any laptop or desktop device. SSH (secure shell) is used for this, which opens a terminal window on the laptop or device that can be used to send commands to the Raspberry Pi and retrieve data from it before sending it to the computer. Putty itself has the main Ikey file format, known as ppk. Raspberry Pis are commonly used as stand-alone, lightweight network computers. Raspberry Pi is wired to the same local network as the remote computer. On the Raspberry Pi, SSH is enabled which is supported by default in some Raspberry Pi distributions, but it can be configured again later using the Raspi-config tools.

3.1.3. VNC Viewer

Virtual Network Computing (VNC) is a graphical desktop sharing application that lets us monitor the desktop interface of one machine with another computer or mobile device remotely. The VNC viewer transmits to the VNC server with a mouse, keyboard, or touch case, receiving updates back on the display. Working directly on the Raspberry Pi is not always convenient. You may also want to include a remote control from another device to work on it. VNC uses Real VNC, which is used with the Raspberry OS. It comprises VNC Viewer, which allows users to remotely access a Raspberry with desktop, and a VNC server enables to monitor the Raspberry Pi remotely. It must be enabled first before using the VNC server. The VNC server provides the users with wireless monitoring to the Raspberry graphical desktop, which enables communication. However, the VNC server can be used to access the graphic remote if the Raspberry is headless and doesn't have a graphic screen.

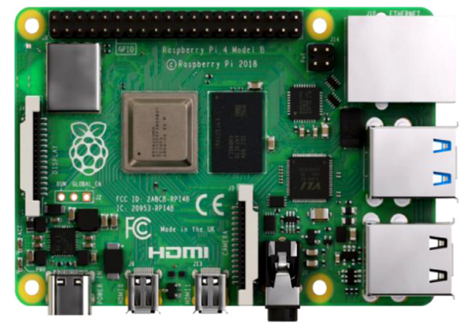

Fig 1. Raspberry Pi

3.2. Hardware Requirements

3.2.1. Raspberry Pi

The Raspberry Pi is a low-cost tiny computer that connects to a computer monitor or television and operates with a regular keyboard and mouse as shown in Fig 1 [8]. It is a handy little gadget that focuses on teaching people of all ages about scripting languages like Scratch and Python. It can perform all the functions of a desktop computer, such as internet surfing and viewing greater-definition clip, worksheets, and playing games. It has been used in several digital devices, including tweeting birdhouses, music machines, and detectors, as well as weather stations and infrared cameras since it is capable of interacting with the outside environment.

It has a 1.2-GHz quad-core chipset BCM2387 with a GPU support of a dual-core and a video core multimedia co-processor and the GPU, which includes dual core multimedia co-processor, including a Bluetooth 4,1 (Bluetooth and Bluetooth Classic). With Bluetooth Low Energy (BLE) and BCM43143 Wi-Fi, the Raspberry Pi 3 offers an up-grade towards a new main processor and improved networking. Furthermore, the power management of Raspberry 3 has been improved, with an upgraded power supply with 3.5 Amps that can handle more powerful external USB devices. The built-in USB ports of Raspberry Pi 3 provide sufficient connectivity to link the mouse or anything else to the RPi.

Most Raspberry Pi system chips can be overclocked to 800 MHz, and some can be overclocked to 1000 MHz. It is reported that the Raspberry Pi 2 can be similarly overclocked, even reaching 1500 MHz in extreme cases (without all safety features and overvoltage restrictions). On Linux distributions, you can use the program command to run "sudo raspiconfig" to perform boot overclocking without breaking the warranty. In certain instances, the Pi will automatically deactivate overclocking when the chip temperature reaches 85°C to cancel the automated settings on overclock and overclocking (that will cancel warranty). A cooler can be used to prevent overheating of Raspberry pi.

3.2.2. R pi's Cam (Raspberry Pi Camera)

An 8-megapixel sensor Pi Camera of Raspberry is used in this project. This camera module consists of 1080p30, 720p60, and 640 × 480p90 video support and support resolution of 3270 × 2444 pixels resolution. Fig 2 shows a Raspberry Pi camera module. It has a fixed lens and a Sony IMX219 image sensor that was specifically made for the R Pi as an add-on board. The Pi module is linked to the RPi through one of the board's little ports on the top part, and it also makes use of the specialized CSI gui, which is specifically made for camera connectivity.

3.2.3. IR Sensor

Infrared sensors are used to count and monitor the number of people who enter and leave the room. The IR sensor's operating voltage is 5VDC, and the I/O pins are 5V and 3.3V compatible. It comes with a variety of options. Fig 3 depicts an IR Sensor that features a built ambient light sensor and a mounting hole, as well as an adjustable sensing range of up to 20cm.

Fig 2. Raspberry Pi Camera

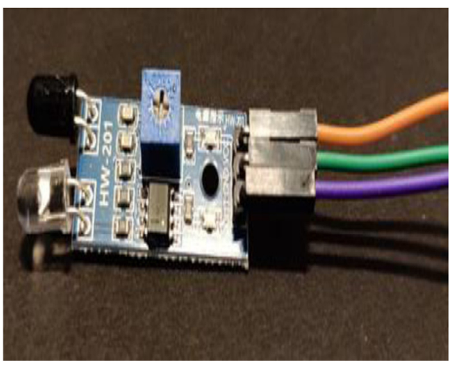

Fig 3. IR Sensor

Fig 4. Temperature Sensor

3.2.4. Temperature Sensor

The temperature sensor (MLX90614) acts as an infrared non-contact temperature reader that reads the temperature without contacting them. Fig 4 shows MLX90614 temperature sensor. Both the Signal ASSP and the IR Sentiment Detector Chip are in the same TO-39(is a type of 'metal can' (also known as 'metal header') package for semiconductor devices.). The thermometer's noise reducer amplifier, with a 17-bit ADC, and powerful DSP efficient unit is used which helps in achieving more correctness. The sensor does have a digital System Management Bus (SMBus) output, with PWN which has been factory calibrated and prepared. A 10-bit PWN is programmed to continuously broadcast the recorded temperature of approximately -19 to 130°C with an outcome resolved up to 0.15°C.

3.2.5. Servo Motor

A servo motor is used to demonstrate the opening and closing of the main door. Fig 5 shows a diagram of a Servo Motor [7], that produces velocity and torque based on the voltage and the amount of current supplied. It also works as a part of a closed-loop system providing velocity

Fig 5. Servo Motor

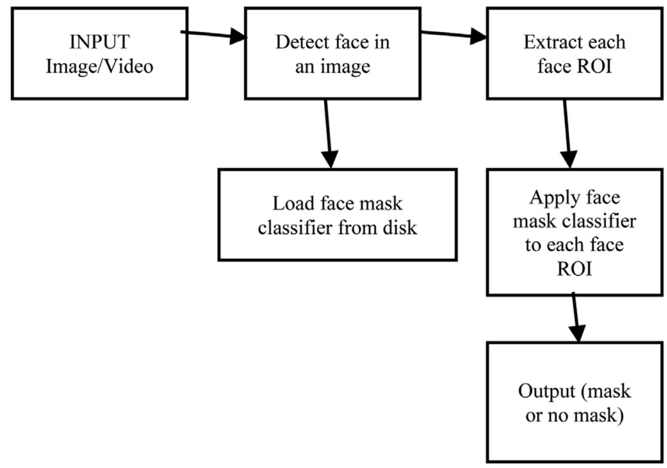

Fig 6. Face Mask Detection

and torque as commanded from the servo controller with a feedback device to close the loop.

4. Methodology

4.1. Face mask detection

4.1.1. Convolution Neural Networks (CNN) Algorithm

In this paper, a deep learning algorithm is used to identify face mask recognition and, Convolution Neural Networks (CNN) classification. A CNN is a form of artificial neural network that is specifically built to interpret pixel input and is mainly used for image recognition and analysis, in which each layer applies to a different set of filters. Around 100's to 1000's of filters is combined to give a final result and then the obtained output is sent to the next layer in this neural network. Evaluation of the proposed framework is done by the face mask detection algorithm using the TensorFlow software library as shown in Fig 6 . The Mask detector model is trained by using Keras and TensorFlow. The steps involved in the algorithm is given below

STEP 1: DATASET COLLECTION
STEP 2: PRE-PROCESSING
STEP 3: SPLITTING
STEP 4: TRAINING
STEP 5: TESTING/EVALUATION

According to the above-mentioned algorithm, all the required dataset and components for building the network is collected from various categories. Once the initial dataset is ready, the next step is to train and test the set. This test dataset is used only in evaluating the performance of the network. Next training should be done, so the neural network learns to identify different categories in the given labels. Finally, the dataset should be evaluated and compared with the ground-truth labels.

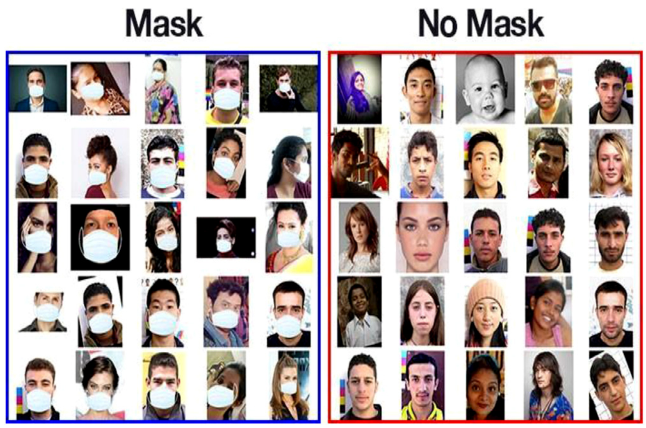

Fig. 7. People with and without mask

4.1.2. Dataset Collection

The images used for training and testing the model were obtained from the internet. The dataset used in this project was designed by Prajna Bhandary. This dataset contains 1,376 photos divided into two classes: 690 images with masks and 686 images without masks as shown in Fig 7. To create this dataset, they took regular photographs of people's faces and then used a custom-designed computer vision Python script to apply face masks to the pictures, yielding an artificial dataset. Facial landmarks allow the users to instantly infer the position of facial components such as the eyes, nose, eyebrows, mouth, and jawline. Then, using facial landmarks, the dataset of faces wearing masks can be created. To determine the bounding box region of a face in an image, start with an image of an individual who's not wearing a face mask and then apply face detection. It can capture the face Region of Interest (ROI) after determining where the face is now in the picture, and then utilize facial landmarks to detect the position of mouth, eyes, nose, and other features. Initially, an image of a mask is required, which will be put to the face automatically utilizing facial landmarks (particularly, the regions around the mouth and chin) to determine where the mask should be placed. After that, the mask is scaled and twisted before being fitted to the face, and the process is repeated for each of the input images, yielding an artificial face mask dataset as shown in Fig. 7.

The face is captured and the blob is constructed from the image that depicts people with and without wearing masks [1]. This blob is passed via network to achieve face detection from the extracted blob and the trust (i.e., probability) is also associated with extracted detection. The weak detection is filtered to ensure that the confidence (probability) is more than min degree of reliability so that face ROI (Region of Interest) is extracted and switched to RGB format from BGR format and it is reformatted to 4 × 224, and then pre-processing is done, Now extracted face is sent via the mask detection model to detect the face sent is wearing a proper face mask or without a face mask. So, the bounding box and text are drawn and probability is included in the label. Finally, a white box which is known as a rectangle bounding box appears with a label as a frame on the output screen.

4.2. Temperature detection

The MLX90614 temperature sensor is connected to the RPi's GPIO Pin, and appropriate code is written for the sensor. Output is Celsius, and if the temperature reaches the standard alarm is given as a warning. Fig 8 depicts the connection of temperature sensor, Pi cam, and IR sensors with Raspberry Pi.

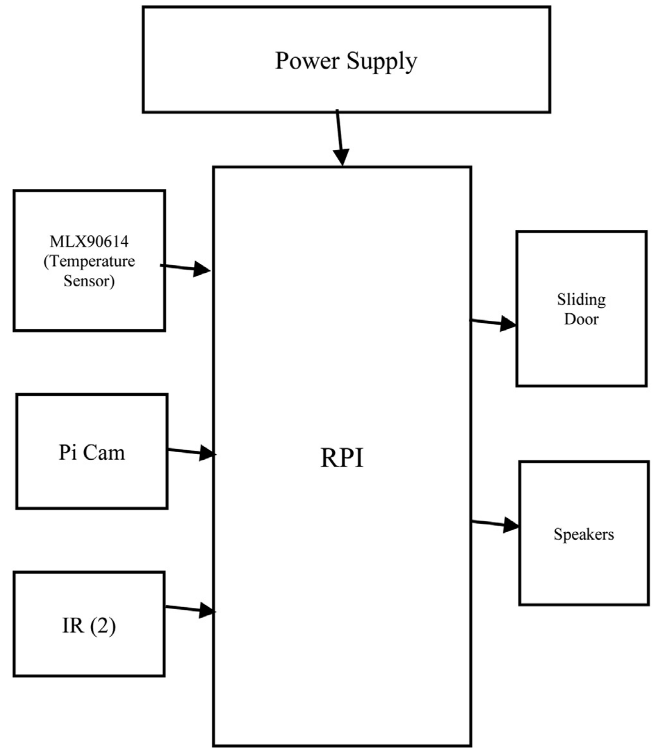

Fig 8. Connections

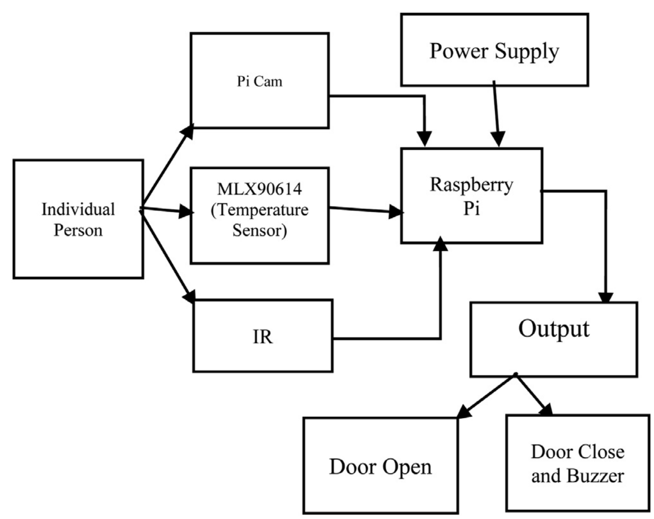

Fig 9. Overall Architecture Diagram

4.3. Number of people passing in and out of the room

Infrared sensors are used to monitor the number of people who enter and leave the room. The Internet of Things (IoT) technology is used to detect temperature and keep a count of the number of people, while mask detection is used to identify individuals near the camera are wearing a mask or not. Fig 9 shows the overall architecture diagram of the system.

4.4. System Overview

Fig 10 shows the overview of the connection structures that make up the solution. Any person attempting to enter the building should first pass through infrared sensors, which are used to track and manage the individual count of people entering the room and later exiting. Body temperature is tested only when the people's total count inside a room is less than the given limit. The MLX90614 body temperature sensor is used for this purpose. If the person's body temperature is too high, the door will not open; if the person's temperature is average, the

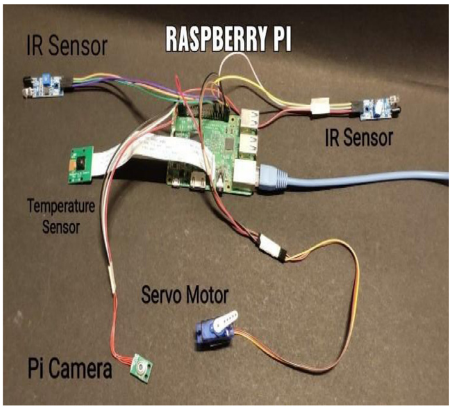

Fig. 10. System Overview

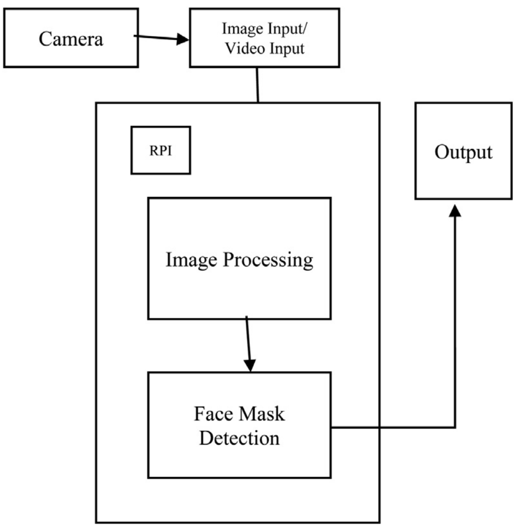

Fig 11. Face Mask Detection Caffe Model

door will open and proceed to the next level, i.e., mask detection. The Raspberry Pi single-board computer with Raspberry Pi Camera is used for this function. If an individual wearing a mask is detected, the door will be opened. If the individual is discovered without a mask, the door will not open. To ensure the guidelines and safety for indoor workers during this COVID-19, this IoT solution based comes into action.

4.5. Face Mask Detection

To implement the mask detection algorithm depending on the Tensor flow library, it contains two parts: the Face Detection Caffe model as shown in Fig 11 and Fig 12 depicts the various phases of the face mask detection model.

5. Results and Discussion

5.1. System requirement

The training was carried out on a computer running the 62-bit Windows10 operating system and equipped with an Intel ® CoreTM i5-8265U CPU running at 1.60GHz and 8 GB of RAM. Python 3.7 is being used as the application development language. The model was developed and trained using Keras as the backend and the Tensor-flow platform. To generate mask detector model input dataset and fine-tune MobileNetV2 is accepted using the training python script. A training history plot.png with accuracy/loss curves is also generated, as seen in

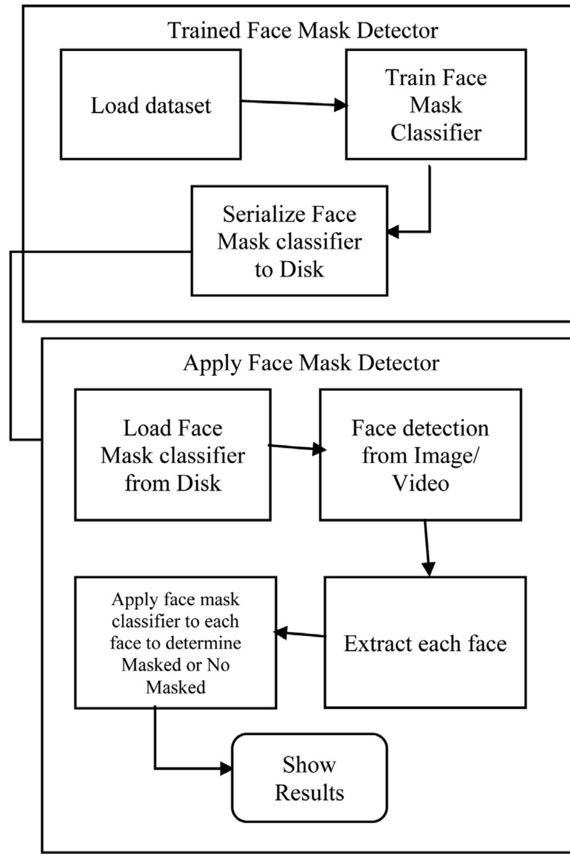

Fig 12. Phases of Face Mask Detection

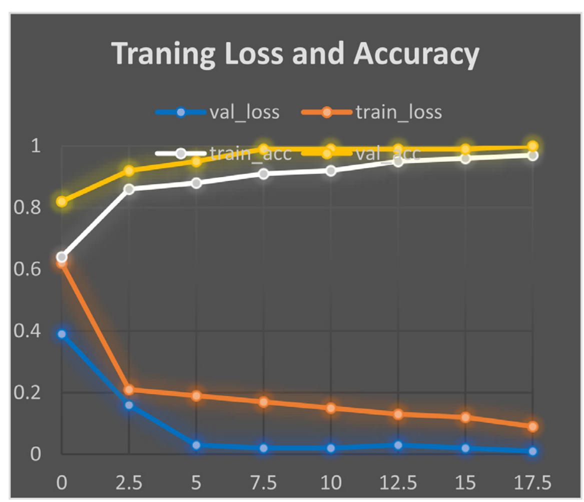

Fig 13. (Training loss and accuracy graph with number of epochs in X-axis and loss/accuracy in Y-axis)

Fig 13. Implementation of face mask detection in Raspberry Pi can be done using the mask detector model.

In this work, a Raspberry Pi 3 Model V is used. It is a low-cost, compact gadget that plugs into a computer. The Raspberry Pi is connected to the laptop via a LAN cable. The VCC of the first IR sensor has a connection with the 2nd pin on R pi, the GND pin has a connection with the 34th pin of the R pi, and the output is connected to the 40th pin of the R pi, which is a general-purpose I/O pin commonly known as GPIO pin. VCC of the second IR sensor has a connection with the 17th pin, GND to the 13th pin, and out pin to the R pi's 38th GPIO pin. The servo motor's

VCC has a connection with 5V input which is the 2nd pin, next GND pin is connected to pin number 39 and the signal pin to pin number 37 which is the GPIO pin on R pi's. The MLX90614 temperature sensor's VCC pin has a connection with 1st pin on R pi, its GND has a connection to pin number 16 on R pi, and the HCL and HDL pins of the sensor are connected to the 2nd and the 3rd pins of the R pi. The R pi camera is attached to the R pi's camera module port.

5.2. Quantitative Analysis with different test case

Optimization techniques are approaches for lowering training losses by changing the properties of neural networks such as weights and learning rate. As optimization elements were introduced in the analysis, Fig 13 depicts validity accuracy diagrams in relation to consistency and loss of validity as contrasted to loss of training. On the one hand, using more hidden layers provides a deeper analytical model, while on the other hand, each extra layer adds complexity to computing.

In addition, increasing the number of neurons in each layer will increase processing costs. To enhance the number of data samples, zoom, pre-processing, shear, and other image augmentation features are frequently used. When these parameters are used, images with these qualities are generated during deep learning model training. Image samples generated using image augmentation enhanced the rate of existing data samples by almost 3x to 4x. However, this cannot be employed because the model will be strongly biased and will fail to generalize properly.

The term epoch refers to the entire set of conceivable inputs. As in the case of calculating model weights after each epoch, the weights are re-adjusted and tested against the same dataset's subsequent cycle simulation (called next epoch). When this is run, the entire training data set is presumed to be in the main memory. Because it is not practical to retain the complete dataset in main memory at different periods for larger datasets. The epoch (dataset) is partitioned into batches, and each batch is loaded into the main memory and run in a sequential manner, with the findings totaled up and finally interpreted as an epoch output.

5.3. Qualitative analysis with different algorithms

The fundamental advantage of CNN over its predecessors would be that it automatically detects significant features without the need for human interference. As a result, CNN would be an excellent answer to computer vision and picture categorization challenges. To utilize another approach, first features from images should be created and then feed those features into a classification technique such as SVM, KNN, logistic regression, and so on. When compared to CNN, these algorithms learn less.

5.4. Comparative analysis with existing papers

A review of the literature finds that none of the previously published research attempted to incorporate all of the aforementioned criteria. Investigated mask detection and social distance recognition [2], but this system cannot be implemented on a Raspberry Pi due to the high processing capacity. Based on fully convolutional networks, the researchers developed a high-accuracy technique for detecting facial masks [13, 15]. However, it was not implemented on the Raspberry Pi.

During the training of the CNN model, very satisfactory validation accuracy was obtained through many experiments and has a recorded accuracy of 99 percent to batch sizes fixed to 32 and 20 iterations for epochs as shown in table 1. As indicated in Fig 13 the performance testing results from visual representation through accuracy and loss. Fig 14 shows the test results on the performance of the model in detecting a person wearing a facemask with a rate of 98.55%. The face will be bounded by the green-colored rectangular box if the mask is detected. Fig 15 displays the test results on the performance of the model in detecting a person without wearing a facemask with a rate of 100%. The face will be bounded by red colored rectangular box if the mask is not detected.

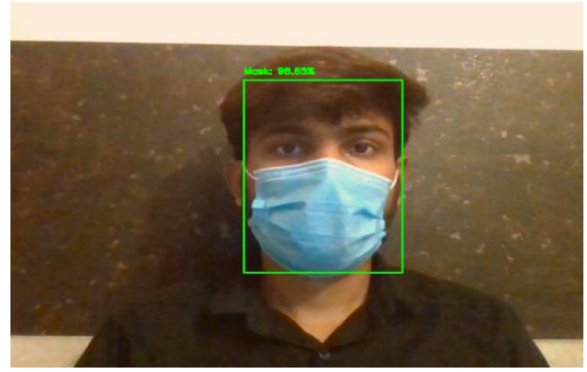

Fig 14. Result for image with mask in green bounding

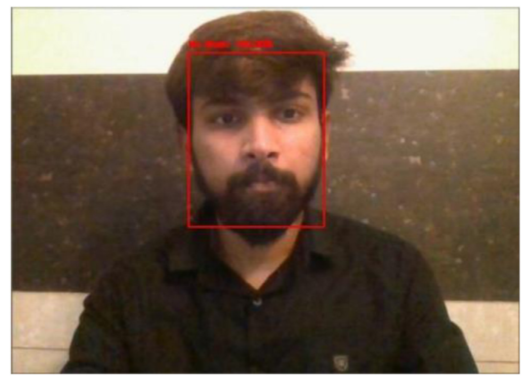

Fig 15. Result for image without mask in red bounding box

To check the temperature measurement system's accuracy and dependability, a mercurial thermometer is used to measure body temperature which is then compared to the temperature value of the system measure. Table 4 shows the results of the experiment. Experiment results show that the absolute measurement error is less than 0.1 C, which equals the medical body temperature monitor.

Equation

Train for 34 steps, validate on 276 samples

Epoch 1/20

Loss: 0.6431 - Accuracy: 0.6676 - Val-loss: 0.3696 – vale-Accuracy: 0.8242

Epoch 2/20

Loss: 0.3507 - Accuracy: 0.8567 - Val-loss: 0.1964 - vale-Accuracy: 0.9375

Epoch 3/20

Loss: 0.2792 - Accuracy: 0.8820 - Val-loss: 0.1383 - vale-Accuracy: 0.9531

Epoch 4/20

Loss: 0.2196 - Accuracy: 0.9148 - Val-loss: 0.1306 - vale-Accuracy: 0.9492

Epoch 5/20

Loss: 0.2006 - Accuracy: 0.9213 - Val-loss: 0.0863 - vale-Accuracy: 0.9688

Epoch 16/20

Loss: 0.0767 - Accuracy: 0.9766 - Val-loss: 0.0291 - vale-Accuracy: 0.9922

Epoch 17/20 loss: 0.1042 - Accuracy: 0.9616 - Val-loss: 0.0243 - vale-Accuracy: 1.0000

Epoch 18/20

Loss: 0.0804 - Accuracy: 0.9672 - Val-loss: 0.0244 - vale-Accuracy: 0.9961

Epoch 19/20

Table 1
Comparison of the article with previous articles

Literature	Aim	Method	Contribution
Adrian Rosebrock,2020	Face Mask Detector with OpenCV, Keras/TensorFlow, and Deep Learning	Concentrate on loading this disc mask detection data set, trained model on this dataset (using Keras/TensorFlow) and serialise your face m ask into your disc.	After the face mask detector is trained, the mask detector is loaded and faces are detected and each face is classified as with mask or without mask
Alfin Hidayat, Subono, Vivien Arief Wardhany, Ajie Setyo Nugroho, Sofyan Hakim, Mirtha Jhoswanda,2020	Designing IoT-Based Independent Pulse Oximetry Kit as an Early Detection Tool for Covid-19 Symptoms Development of portable pulse oximetry kit	Using the ESPDUINO-32 as a receiver, or a Bluetooth Low Energy receiver that will be linked to the Pulse Oximetry BLE.	Portable pulse oximetry kit products that are equipped with GPS and its integration with IoT technology
Cristina Stolojescu-Crisan, Bogdan- Petru Butunoi and Calin Crisan, 2020	System for avoiding touching various objects and surfaces in offices using IoT	QTOGGLE APP has been designed as a way of interconnecting sensors, actuators and other data sources with the purpose of multiple automations	Simple solution for building automation based on ESP8266/ESP8285 chips and Raspberry Pi boards.
Mrudula, Ananya Pandey, Kruthika Dinesh, Reethika P,2020	Face detection for Smart Door Unlocking System using Raspberry pi	Object detection using Haar feature-based cascade classifiers is a detection method proposed by. It is a Machine Learning based approach in which the cascade function is trained over a lot of positive and negative images and then used to detect objects in other images	Internet of Things (IoT) based Smart Door Unlocking System using Raspberry pi
Petrovic, Nenad & Radenković, Maša & Nejkovic, Valentina, 2020	Mobile application based on app sheet	Using IOT based computer vision system to monitor each Pearson inside a room by placing raspberry Pi at different location	AppSheet based covid-19 mobile application
Preeti Nagrath, Rachna Jain, Agam Madan, Rohan Arora, Piyush Kataria, and Jude Hemanth, 2020	A real time DNN-based face mask detection system using single shot multibox detector and MobileNetV2	The SSDMNV2 approach uses Single Shot Multibox Detector as a face detector and MobilenetV2 architecture as a framework for the classifier	The method used in this paper gives us an accuracy score of 0.9264 and an F1 score of 0.93
Meenpal, T, Balakrishnan, A., & Verma, A, 2019	Face mask detection using segmentation	For feature extraction, the VGG – 16 Architecture is employed. Fully Convolutional Networks are used for training to semantically segment out the faces in the image.	Method for creating precise face segmentation masks from any arbitrary size input image
Lim, M. G., & Chuah, J. H.,2018	Durian Types Recognition Using Deep Learning Techniques to increase the accuracy and speed of recognition, an efficient model based on durian characteristics using CNN.	Non-durian images are employed in the neural network training procedure.	The trained model's prediction accuracy on flawless bottom-view photos of Durio zibethinus was 82.50 percent.

Loss: 0.0836 - Accuracy: 0.9710 - Val-loss: 0.0440 - vale-Accuracy: 0.9883

Epoch 20/20

Loss: 0.0717 - Accuracy: 0.9710 - Val-loss: 0.0270 - vale-Accuracy: 0.9922

Loss/Accuracy

The main purpose of the developed system is to avoid the spread of COVID-19 in public places such as shopping malls, offices, and so on. The system can monitor an individual's body temperature and can perform face mask detection. The count of the people inside the room will be shown when the facemask detector model is loaded. When an individual passes through the IR sensor, it will proceed to the next level only if the people count inside the room is less than the defined limit. Then the temperature sensor detects their body temperature, and if it is less than the set limit, the Pi cam activates and checks if they are wearing a mask. The door automatically opens if the mask is detected and the count goes up by one; otherwise, the person is not allowed and the count stays the same. Similarly, if another person passes through the IR sensor, it tests their body temperature; if they meet all of the requirements, the count increases by one, and they will be allowed. The count increases until the maximum limit is reached; once the maximum limit is reached, the door will not open. When an individual moves through the exits, the 2nd IR sensor senses them, the door opens, and the count is reduced by one. An alert system is integrated into the system by use of the espeak library, a text-to-speech platform that generates a voice notification that informs a person who is not wearing a facemask if the body temperature is high and if the number of people exceeds the maximum set limit. Table 4 shows the results of the test case of different scenarios.

6. Conclusion

New developments and the availability of smart technologies force to the creation of new models, which will help meet the needs of developing countries. In this work, an IoT-enabled smart door is developed to monitor body temperature and detect face masks that can enhance public safety. This will help to reduce manpower while also providing an extra layer of protection against the spread of Covid-19 infection. The model uses a real-time deep learning system using Raspberry pi to detect face masks, and temperature detection as well as monitor the count of people present at any given time. The device performs excellently when it comes to temperature measurement and mask detection, the trained model was able to achieve a result of 97 percent. The test results demonstrate a high level of accuracy in detecting people wearing and not wearing facemasks, as well as it also generates alarms monitored and recorded. Furthermore, there are numerous techniques to enhance performance to improve results. Future development will include improving the accuracy of these steps, using a combination of various features, and improving performance, as well as producing a mobile app with a user friendly interface for monitoring. As a result, authorities will be able to take immediate action following pandemic safety standards. Table 2 and 3

Acknowledgment

First of all, we are indebted to GOD ALMIGHTY for allowing us to excel in our efforts to complete this project on time. We express our sincere thanks to our respected dean Dr. Abdul Sharief, Dean, School of Engineering, Presidency University for getting us permission to undergo the

Table 2
Temperature Measurements compared with mercurial thermometer

Temperature measurement in mercurial thermometer (°C)	Experiment temperature in this system (°C)	Temperature measurement error absolute value (°C)
36.8	36.03	0.05
37.2	37.16	0.04
38.0	37.92	0.08
38.5	38..52	0.02

Table 3
Validation/Accuracy

	precision	recall	f1-score	support
With mask	0.99	1.00	0.99	138
Without mask	1.00	0.99	0.99	138
Accuracy			0.99	276
Macro avg	0.99	0.99	0.99	276
Weighted avg	0.99	0.99	0.99	276

Table 4
Results

IR Count	Temperature Check	Mask Detection	Door Open/Close
Count < 5	Temp < 39·c	Mask detected	Open
Count > 5	Temp < 39·c	Mask detected	Close
Count < 5	Temp > 39·c	Mask detected	Close
Count < 5	Temp < 39·c	Mask not detected	Close
Count > 5	Temp > 39·c	Mask not detected	Close

project. We record our heartfelt gratitude to our beloved professor Dr. C. Kalaiarasan, University Project-II in charge, Associate Dean-Admin, Department of Computer Science and Engineering, Presidency University for rendering timely help for the successful completion of this project. We are greatly indebted to our guide Mrs. Archana Sasi, Assistant Professor, Department of Computer Science and Engineering, Presidency University for her inspirational guidance, valuable suggestions, and for providing us a chance to express our technical capabilities in every respect for the completion of the project work. We thank our friends for the strong support and inspiration they have provided us in bringing out this project.

References

[1] A. Rosebrock,"COVID-19: Face Mask Detector with OpenCV, Keras/TensorFlow, and Deep Learning", May 4, 2020, https://www.pyimagesearch.com/2020/05/04/covid-19-face-mask-detector-with-opencv-keras-tensorflow-and-deep-learning/.

[2] A. Hidayat, Subono, V.A. Wardhany, A.S. Nugroho, S. Hakim, M. Jhoswanda, "Designing IoT-Based Independent Pulse Oximetry Kit as an Early Detection Tool for Covid-19 Symptoms ", 2020 3rd International Conference on Computer and Informatics Engineering (IC2IE).

[3] Carlo Alberto Boano Matteo Lasagni Kay Romer Tanja Lange. "Accurate Temperature Measurements for Medical Research using Body Sensor Networks".

[4] Cristina S..........-.C......, B-P BUTUNOI and Calin C......," IoT Based Intelligent Building Applications in the Context of COVID-19 Pandemic", 2020, 10.1109/ISETC50328.2020.9301124

[5] F.H. Yahaya, Y.M. Yusoff, H.Z. Abidin, R.A. Rahman, Development of a PIC-based wireless Sensor node utilizing XBEE Technology, IEEE International Conference on Information Management and Engineering, 2010.

[6] Gayatri Deora, Ramakrishna Godhula and Dr. Vishwas Udpikar "Study of Masked Face Detection Approach in Video Analytics, IEEE Conference on Advances in Signal Processing, 2016.

[7] https://www.electronicwings.com/arduino/servo-motor-interfacing-with-arduino-uno.

[8] https://www.robocarstore.com/products/raspberry-pi-4-model-b-board-with-1gb-lpddr4-sdram.

[9] K Baskaran, Baskaran P., N. Kumaratharan, Rajaram V., "IoT Based COVID Preventive System for Work Environment", the Fourth International Conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud) (I-SMAC) IEEE Xplore Part Number: CFP20OSV-ART; ISBN: 978-1-7281-5464-0.

[10] Lim, M.G., & Chuah, J.H. (2018). Durian Types Recognition Using Deep Learning Techniques. 2018 9th IEEE Control and System Graduate Research Colloquium (ICSGRC). doi:10.1109/icsgrc.2018.8657535.

[11] L. Tang and H. Kiourmars, Wireless Network for Health Monitoring Heart Rate and Temperature Sensor", 10.1109/ICSensT.2011.6137000.

[12] T. Meenpal, A. Balakrishnan, A. Verma, Facial Mask Detection using Semantic Segmentation, 2019 4th International Conference on Computing, Communications and Security (ICCCS), 2019, doi:10.1109/cccs.2019.8888092.

[13] Ana.nya Pa.ndey Mrudula, Kruthi.ka Dinesh, P. Reethika, Smart Door Unlocking System, Int. Res. J. Eng. Technol. (IRJET) Volume: 07 (2020) Issue: 05 May.

[14] N. Ozkaya, S. Sagiroglu, Intelligent face Mask Prediction System, IEEE International Joint Conference on Neural Networks, 2008.

[15] Petrovic, N. & Radenković, M. & Nejkovic, V.. (2020). Data-Driven Mobile Applications Based on AppSheet as Support in COVID-19 Crisis. https://www.researchgate.net/publication/343678867_Data-Driven_Mobile_Applications_Based_on_AppSheet_as_Support_in_COVID-19_Crisis.

[16] 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 andMobileNetV2", Published online 2020 Dec 31.

[17] Rit.wik Biswas, Av.ijit Roy, Real-Time Temperature Graph using MATLAB and Arduino, Int. J. Eng. Res. Technol. (IJERT) 09 (05) (2020) (May 2020) https://www.ijert.org/real-time-temperature-graph-using-matlab-and-arduino .

[18] Sam.my V. Militante, Na.nette V. Dionisio, Deep Learning Implementation of Facemask and Physical Distancing Detection with Alarm Systems, in: Vocational Education and Electrical Engineering (ICVEE) 2020 Third International Conference on, 2020, pp. 1–5. https://ieeexplore.ieee.org/document/9232610.

[19] S.S.. Vedaei, A. Fotovvat, M.R. Mohebbian, G.M.E. Rahman, K.A. Wahid, P. Babyn, H.R. Marateb, M. Mansourian, And Ramin Sami, "An IoT-Based System for Automated Health Monitoring and Surveillance in Post-Pandemic Life", date of publication October 12, 2020.

[20] P. Subramani, G.B. Rajendran, J. Sengupta, R. Pérez de Prado, P.B. Divakarachari, A block bi-diagonalization-based pre-coding for indoor multiple-input-multiple-output-visible light communication system, Energies 13 (13) (2020) 3466.

[21] T. Nguyen, B.H. Liu, N. Nguyen, B. Dumba, J.T. Chou, Smart grid vulnerability and defense analysis under cascading failure attacks, *IEEE Trans. Power Delivery* (2021).

[22] V. Vinitha., V. Velantina., COVID-19 facemask detection with deep learning and computer vision, Int. Res. J. Eng. Technol. (IRJET) Volume: 07 (2020) Issue: 08 Aug.

[23] S. Rajendrakumar, V.K. Parvati, Automation of irrigation system through embedded computing technology, in: Proceedings of the 3rd International Conference on Cryptography, Security and Privacy, 2019, pp. 289–293.

[24] S. Vadivel, S. Konda, K.R. Balmuri, A. Stateczny, B.D. Parameshachari, Dynamic Route Discovery Using Modified Grasshopper Optimization Algorithm in Wireless Ad-Hoc Visible Light Communication Network, Electronics 10 (10) (2021) 1176.

[25] Yu. Chen, H. Zhang and NA. Wang, "Body Temperature and Alarm System Used in Hospital-Based on 1-wire and Wireless Communication Technology", 2008 International Workshop on Education and Training & 2008 International Workshop on Geoscience and Remote Sensing, pp. 401-404, 2008.

[26] M.A. Naeem, T.N. Nguyen, R. Ali, K. Cengiz, Y. Meng, T. Khurshaid, Hybrid Cache Management in IoT-based Named Data Networking, *IEEE Internet of Things J.* (2021).

[27] S.I. Chu, C.L. Wu, T.N. Nguyen, B.H. Liu, Polynomial Computation Using Unipolar Stochastic Logic and Correlation Technique, *IEEE Trans. Comput.* (2021).

[28] Y. Sun, J. Liu, K. Yu, M. Alazab, K. Lin, "PMRSS: Privacy-preserving Medical Record Searching Scheme for Intelligent Diagnosis in IoT Healthcare", IEEE Trans. Ind. Inf., doi: 10.1109/TII.2021.3070544.