Original Article

Abhishek Gudipalli*, Vaibhav Kejriwal, Vaishnavi Patel, Riddhi Gupta, and Utkarsh Dixit

COVID bell – A smart doorbell solution for prevention of COVID-19

https://doi.org/10.1515/pjbr-2022-0115
received January 11, 2023; accepted March 23, 2023

Abstract: The article introduces a novel strategy for efficiently mitigating COVID-19 distribution at the local level due to contact with any surfaces. Our project aims to be a critical safety shield for the general people in the fight against the epidemic. An ultrasonic sensor is integrated with the automated doorbell system to ring the doorbell with a hand motion. A temperature sensor Mlx90614 is also included in the system, which records the temperature of the person standing in front of the door. The device also includes a camera module that captures the image of the person standing at the front entrance. The captured image is processed through an ML model which runs at over 30 fps to detect whether or not the person is wearing a mask. The image and the temperature of the person standing outside are sent to the owner through the configured iOS application. If the person outside is wearing a mask, one can open the door through the app itself and permit the entry of the person standing outside thereby integrating the edge device with an app for a better user experience. The system helps in reducing physical contact, and the results obtained are at par with the already existing solutions and provide a few advantages over them.

Keywords: Arduino UNO, MLX90614, ESP 32 Cam, ESP8266 WiFi module, firebase, MATLAB, Swift, buzzer, IoS, XCode, CoreML, HM-10 Bluetooth BLE module, HC-SR04 ultrasonic sensor

1 Introduction

Several studies have looked into how long SARS-CoV-2 can persist on porous and non-porous surfaces. Viable viruses can be identified on non-porous surfaces for days to weeks. Since the beginning of the COVID-19 epidemic, there has been an upsurge in poisonings and injuries caused by the improper use of cleaners and disinfectants. Some disinfection methods, particularly those that include fogging or misting, are neither safe nor effective for inactivating the virus unless they are done appropriately. To cater to the needs of the pandemic, a lot of new technologies and innovations were introduced. Not only does identifying a person who has COVID pose a problem but also the challenge to stop the spread via surface contact became a major issue. The smart doorbell has been playing an important role in protecting our modern workspace since they were invented [1]. Large corporations have recently begun to offer a smart doorbell that integrates all available services, including face mask recognition at the door and opening it automatically at an industrial scale. While this proposed solution is best for offices and public entrances, it is not ideal for homes where more details of the person entering need to be given directly to the house owner. There are a lot of touch-free doorbell solutions in the market, and some are in the development phase but very few of those have a mobile application integrated with them, which gives various insights to the user about the guest. To solve this problem, we have integrated the concept of a smart doorbell with a user-friendly mobile application. Mobile applications give more control to the user in terms of monitoring and controlling the entrance of people into their homes. In terms of durability, energy efficiency, and cost-effectiveness, this contactless doorbell concept will

*** Corresponding author: Abhishek Gudipalli,** School of Electrical Engineering, Vellore Institute of Technology, Vellore, Tamil Nadu, India, e-mail: abhishek.g@vit.ac.in
Vaibhav Kejriwal: School of Electronics Engineering, Vellore Institute of Technology, Vellore, Tamil Nadu, India, e-mail: vaibhav.kejriwal2019@vitstudent.ac.in
Vaishnavi Patel: School of Electronics Engineering, Vellore Institute of Technology, Vellore, Tamil Nadu, India, e-mail: vaishnavi.patel2019@vitstudent.ac.in
Riddhi Gupta: School of Electronics Engineering, Vellore Institute of Technology, Vellore, Tamil Nadu, India, e-mail: riddhi.gupta2019@vitstudent.ac.in
Utkarsh Dixit: School of Electronics Engineering, Vellore Institute of Technology, Vellore, Tamil Nadu, India, e-mail: utkarsh.dixit2019@vitstudent.ac.in

outperform traditional ones put in our homes and offices. But most importantly, this will eliminate the need for a person to check the temperature manually with a temperature gun or thermometer. The person entering simply needs to hover their hands near the bell which has a temperature sensor that would record the temperature of the person. This along with a captured image from the camera module will be sent to the paired mobile application installed through firebase. The image of the person will be analysed by the Mask Detection model which can determine whether or not the person is wearing a mask with optimum accuracy. The data are fetched from firebase in the mobile application, and now, the user can choose to allow or deny the entry of the visitor with just the click of a button. Adding these features to the system will help the user in a lot more ways, and this will benefit the user as it is more convenient and safer to use than any other existing system.

2 Literature survey

2.1 Smart doorbells

Smart doorbells have become an essential and obvious part of the fully integrated implementation of smart homes. These smart doorbells help to increase the efficiency of security by avoiding unwanted access such as robbery and invasion as well as to keep a track of people entering the smart home [2–5]. The controller of these doorbells can easily answer the bell and decide whether to let the guest enter or not accordingly through adaptive learning and various other technologies [6]. As smart doorbells have been extensively used, newer and upcoming technologies have been introduced to improve the existing systems [5,7–9]. The issue with the current design lies in its complexity. This means that the working mechanism or the implementation of these integrated solutions is usually vague to the eye of the user. In case of failures or breakdowns, the user will need to contact a professional for repairs and maintenance. It is highly likely that the whole system may have to be replaced because of the failure of a component in the system, and such systems are not fully feasible for real-life implementation.

2.2 Dashbell

This smart doorbell system integrated with the already implied smart home systems goes a notch above the regular smart doorbells. The system uses a simple design based on Raspberry Pi, AWS (Amazon Web Services), and an Android app. The system like any smart doorbell sends a notification to the user's mobile through the application in which AWS serves as the link. The user can see who the visitor is and then decide whether to allow entry or not. What sets the dashbell apart is that it is more cost-efficient than the existing smart doorbells. It uses an Amazon Dash Button which is easily available in the market for 4.99 US Dollars or 390 Indian Rupees [10,11]. Also, as the system design is simple and easy to understand, anyone can understand the design and resolve the problem in case of a breakdown or malfunction. Further individual parts can be easily replaced instead of replacing the system as a whole. The issue with Dashbell is that even though it reduces the overall cost, it is still a bit costly to be used by the mass. This indeed can be made cheaper without affecting efficiency and functions. Another downside of the proposed implementation is that it does not provide a contact-free experience which may lead to contact spread of COVID-19 and other viruses and does not gather information about the guest's temperature and whether they are wearing a face mask or not. Also, the solution is only implemented for Android users and users with other operating software like iOS [12].

2.3 Survey on Internet of Things (IoT) and AI-based smart doorbell system

This survey was done to analyse the existing software technologies and suggest better alternatives for fast and accurate face detection. This survey points out the complexity of existing haar models to detect the face of a person. Haar models not only take time but also add extra steps to the procedure of face detection [13,14]. The major advantage of the model is that it is serverless, which makes it easier to add or delete functions as the pipeline grows. This article [15] proposes the use of Kairos face recognition, which is a service provided by Kairos company. This model does offer accuracy and speed but it cannot detect face masks, which is our added functionality to the smart doorbell system. Moreover, the Kairos software is a paid software that will cost more and more as the number of doorbell devices increases. One of the biggest disadvantages of this model is that it starts detecting the face when the doorbell is pressed [14]. This is not safe as COVID is a surface-spread virus. We capture the image as soon as the person hovers their hand in front of the doorbell. We have carefully done the literature survey to find the best possible open-source free-of-cost face mask detection model that proves to be highly accurate and efficient. We have chosen a very lightweight model that integrates easily with the app. The neural

layers of our model are arranged in such a way that increases the overall inference speed of face mask detection. Our model performs considerably well in dimly lit areas. It is also able to identify a face in case of backgrounds with high disturbance.

2.4 Smart doorbell design using federated deep learning

This article demonstrates a Doorbell Security System, which uses Federated Deep Learning to identify objects and people near the door and deploy it on a scalable serverless architecture [16]. Federated learning is a new and upcoming field in the world of Machine Learning. It works by the means of Model Aggregation over Data Aggregation providing faster and more accurate results as compared to other methods [17,18]. The article demonstrates an architecture consisting of a video doorbell that will stream video to the Federated Server and Cloud [19]. The Federated server will use the provided data to extract features and provide the user with valuable information in an iOS Application. It can detect people, objects, cars, animals, and more [17,20,22]. It also has a notification system to notify users whenever someone or something is detected by the doorbell. The article brings up an important issue which is the privacy of user data and aims to deploy Homomorphic Encryption and compute on Secure Multiparty Computation, which enables multiple parties to collaboratively work on an agreed-upon computation, thus preventing data leaks [21]. This type of model can incorporate the detection of Masks to identify people complying with the COVID protocols [23]. The current proposed model in the mentioned paper seems to be using a RESTFUL approach towards the communication between hardware and iOS application. It alerts users via Push Notification Services. This can be made more efficient with a real-time approach using a Web-socket implementation. This implementation combined with temperature detection, Mask detection, and the ability to remotely control the door would result in an almost perfect system that can help us curb the spread of viruses.

2.5 Development of COVID-19 detection smart doorbell based on IoT

As we know, Necessity is the mother of all innovations, and this system is born out of the need of monitoring visitors' temperature and reducing contact with possibly infected people as much as possible. The system proposes an Infrared system be installed on the bell switch, which senses the person's temperature upon contact using their fingers or thumb. The recorded temperature is displayed on an LCD screen for the residents to monitor. The temperature is also cross-referenced with standard fever temperatures to notify the user about the risks involved [24]. The proposed solution offers a good window into the development and integration of technology, mainly IoT in today's day and age where security risks are not necessarily limited to unwanted break-ins and uninvited guests. It treats and justifies COVID-19 spread as a security risk and rightfully so. The system comprises a Raspberry-pi along with a Fingerprint sensor, Infrared sensors, and a camera. It adds a layer of infrared temperature scanning to existing solutions as a means to detect COVID-19 using visitors' temperature readings. The major drawback of the proposed system is that it does not necessarily minimise contact as the visitor has to touch the doorbell for their temperature to be recorded. If an infected person touches the doorbell, it leads to a high risk of contact spread to the next visitors. It also does not take into account asymptomatic carriers of the virus as temperature for them remains within normal range. Although there is not any existing solution to detect the asymptomatic carriers apart from a sure-shot RTPCR test, the spread can still be minimised by following proper guidelines such as wearing masks. The system also does not offer a remote solution where the resident need not necessarily go to the door to allow a visitor to enter back in. A good example would be if a house has multiple residents and one of them is at the door. It is usually best to allow them to sanitise properly before coming in contact with other people. A remote solution could allow the outsider to get in without coming in contact with others and sanitise properly as per guidelines.

2.6 Development of COVID-19 detection smart doorbell based on IoT

This article successfully implements a smart doorbell that secures the home and office environment [25]. The major advantages of the proposed model include an HD camera and an intruder alarm system. The system turns on the alarm if anyone tries to steal the installed device itself. The video call and audio features allow the user to communicate with the person standing at the door. It is connected to smart artificial intelligence tools like Alexa and

Siri, which enhance the ease of use. They also suggest an automatic power supply system that ensures the device never turns off. This increases the battery life of the system as well. But looking at the current scenario of COVID, we decided to add more relevant features that are more advantageous like the face mask detection and temperature monitoring system. What makes our system more comprehensive is the integration of a mobile app that allows the user to check and control everything from the mobile app itself. Furthermore, we are using a real-time database instead of a primitive SD card to transfer and feed data into the device. Our model smartly switches between WiFi networks and Bluetooth based on the availability of the network and the proximity of the user from the bell. We propose to put the data collected to better use by analysing the temperature data for COVID spread situations in a particular region. The most important aspect of any device is its availability and cost-effectiveness. We have used commonly available hardware modules that ensure the wide availability and low cost of the system.

2.7 JustIoT IoT based on the Firebase real-time database

This article mainly consists of four components: back-end Google Firebase real-time database, front-end SPA (Single Page Application) web monitoring program (including mobile monitoring App), controller software hardware, and intelligence server that supports MQTT connection and condition control. The web portal receives data from microcontrollers like Arduino allowing users to control rules and monitor and control the system. The web portal uses a management-based web page on the Angular front-end, which is connected to the Firebase real-time database. MQTT server is used to establish a connection with Arduino. To further enhance the system in the future, the article states that a machine learning framework can be integrated to make it more intelligent [26]. Even the latest variant of CNN termed Capsule Network analysed and employed in previous studies [27–29] could also be used for increasing the intelligence in the proposed approach.

2.8 Impact of COVID-19 on education

The article puts forward an analysis of responses to a survey in which students of different organisations were asked about their opinions and feedback on the virtual/online classroom and education system. The study states that instead of utilising the benefit of e-teaching during the pandemic teachers rather conducted online classes in urgency and did not focus on building the students' skills and experiences. In its findings, the article also talks about how teachers and academians alike are not familiar and trained with online teaching platforms and tools, and hence, there is a gap in communication between the mentors and the students. The survey shows how the lack of attention and seriousness affected the education of students [30].

2.9 Role of IoT to avoid spreading of COVID-19

The article presents a solution to prevent the spread of COVID-19 using an IoT system which consists of an infrared (IR) thermometer, an IR sensor, a smartwatch, an optical camera, and an IP camera. The suspected zone is scanned using edge devices (the cameras and the sensors), and as soon as the high temperature is detected, the camera captures the image of the suspected person and sends the image and GPS location of the person to a health officer and action could be taken accordingly. For transferring and storing the data send to the officer, the system uses a secure cloud architecture within which an ML algorithm runs, which tries to identify the person and analyse the risk of spreading based on the data provided.

3 Design

3.1 Hardware

The hardware system design has been implemented in such a way that it is cheap, easy to comprehend for the user, and at the same time effective in comparison to existing smart doorbells. The system mainly uses an Arduino UNO as the microcontroller and different sensors and actuators like the HC-SR04 ultrasonic sensor, MLX90614 infrared temperature sensor, and various others. These elements were used keeping in mind their compatibility with the system, their price point, and most importantly their efficiency and simplicity. The hardware connects seamlessly to a Cloud Architecture Hosted on Firebase, which in turn communicates with the client-side app, developed on Swift using Xcode for iOS. Each element of the hardware and software side of the system and their respective function is explained and analysed on how to maximise their output, which are discussed in the following sections.

3.1.1 HC-SR04 ultrasonic sensor

This inexpensive sensor (Figure 1) has a range of 2–400 cm of non-contact measurement capability and a ranging accuracy of up to 3 mm. This sensor is widely used in mini-projects and has been proven to be effective in most situations. We are using the ultrasonic sensor to detect the presence of a person at the door as they wave their hand in front of the sensor. This sensor has a simple integration system and is inexpensive. In case of a breakdown, a person can easily buy it as it is widely available in the market, and the repair or replacement is also simple to implement.

3.1.2 MLX90614 infrared temperature sensor

It is a highly precise, non-contact IR temperature sensor designed by Melexis (Figure 1). The MLX90614 generates two temperature measurements: one the object and the other the ambient. The ambient temperature measures the temperature on the sensor's die, whereas the object temperature is a non-contact value "seen" from the sensor. The ambient temperature can be used to calibrate the data, but object temperature readings are what we truly need. This temperature sensor is designed to measure the temperature of a targeted object by absorbing the emitted IR rays. The object temperature measurement ranges from −70 to 382.2°C. The accuracy is nearly ±0.5°C (around room temperatures). The infrared technology used in the sensor helps to increase the efficiency of the measured temperature. This temperature sensor like the HC-SR04 is also easily available in the market and can be implemented with ease. As the guest waves their hand over the ultrasonic sensor, the MLX90614 activates and measures the guest's temperature and immediately sends the value to Arduino to be processed and transferred further to Firebase and then lastly to the mobile application. It is also one of the best inexpensive options for non-contact infrared temperature scanners.

3.1.3 ESP32-CAM module

The ESP32-CAM is a small-size camera module based on ESP32 (Figure 1). It is a low-power consumption camera

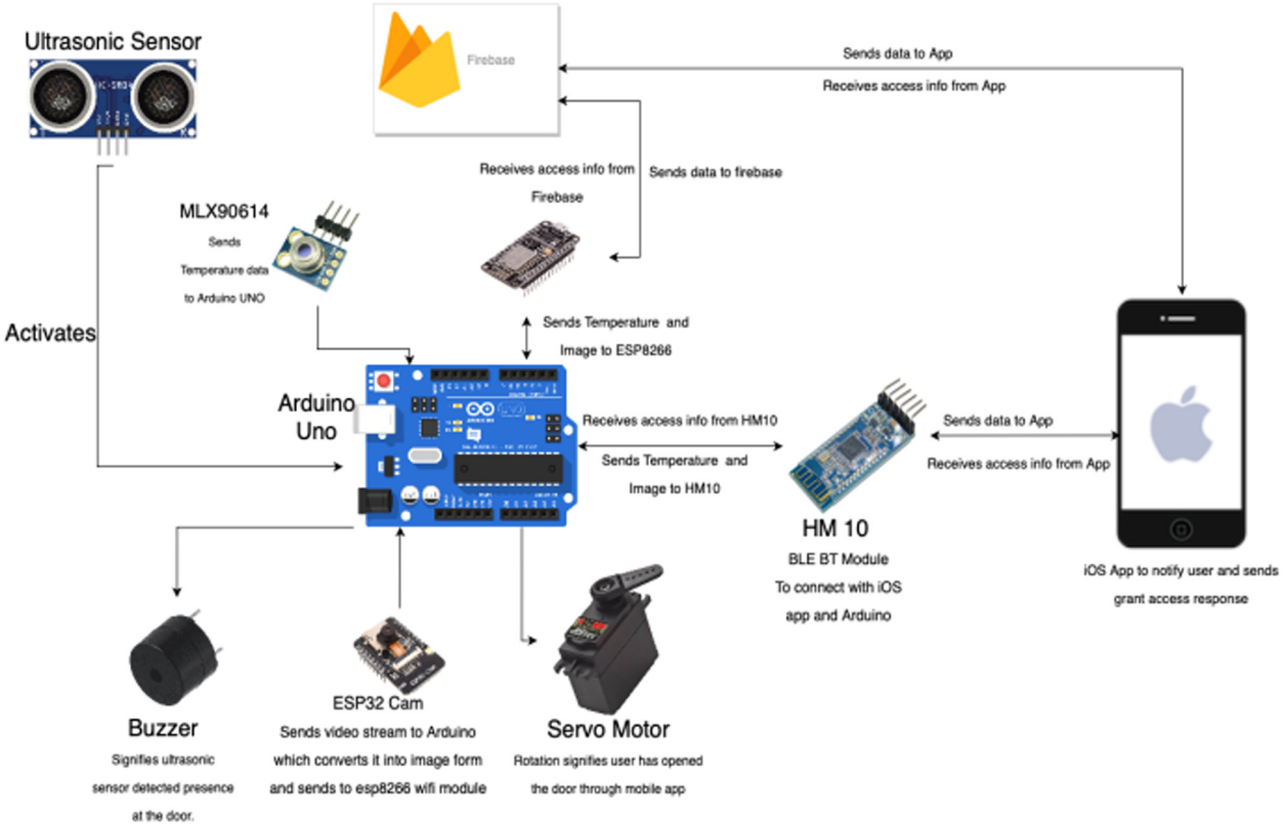

Figure 1: Flow chart of the integrated system.

module. There is an OV2640 camera included and it also consists of an onboard TF card slot. This camera module can be widely used in intelligent IoT applications such as WiFi image upload, wireless video monitoring, QR identification, and so on. As soon as the ultrasonic system activates the Arduino, it triggers the ESP32 camera and the camera starts functioning. The Arduino communicates with the ESP32 camera and commands it to send a frame with the specified quality and extension (.jpeg in our case). The camera will capture a frame and will send it to the Arduino board as per the configurations provided. The Arduino will convert the received .jpeg file into a base64 string. This allows us to pass all the collected data into native data types such as integers, floats, and strings, which makes the data easier to be processed by other devices. These data from the Arduino are passed to Firebase through the ESP8266 WiFi module, which further processes the data and sends it to the mobile application.

3.1.4 HM-10 Bluetooth BLE module

The HM-10 BLE module is based on TI CC2540 or CC2541 Bluetooth System on Chip (SOC). It is a BLE (Bluetooth Low Energy) module (System on Chip) that operates on a 3.3v SMD (Figure 1). BLE is designed to use less power and reduce the cost without compromising the communication range between devices. This module is supported by many mobile operating systems like iOS, and Android as well as macOS, Windows, and Linux. The HM-10 module is available in two versions in the market. The first being HM-10C and the second HM-10S. The module used with Arduino has gained popularity over the last few years. The reason is the standard UART connection, and this makes the connection with an Arduino simple and quick. The UART or the Universal Asynchronous Receiver/Transmitter layer serves a dual purpose. It simplifies functioning while hiding the BLE layer, which allows one to have almost zero control over what actually is happening on the BLE's side of things. The HM-10 only supports version 4.0, which limits it to not working with BT 2 or BT 2.1 modules; for example, the HC-06 and the HC-05.AT commands are sent through the serial UART connection to access control on the HM-10. This module allows us to connect to the client-side application via BLE. This acts as a Fall-back or the second layer of communication between devices. In case, the hardware is unable to connect to the Internet, and the device is well within the range of the user's phone, the data consisting of temperature and image as base64 string are transmitted to the application via Bluetooth. The command sent by the user's application to open the door is also sent back via Bluetooth and received by the hardware via the Bluetooth module.

3.1.5 Esp8266 WiFi module

The ESP8266 WiFi module is an SOC widely known as used for its IoT applications and implementations (Figure 1). It is an independent transmitter and collector, which is effectively accessible in the market at a low cost. It is mainly used to enable and establish an Internet connection to any microcontroller and applications of embedded systems. It is equipped for facilitating an application too; also, it can help in offloading all WiFi organising capabilities from another application processor. It has powerful onboard storage as well as processing capability that allows it to be easily integrated with various sensors and devices through its GPIOs or the General Input/Output pins with very less loading during runtime and a very minimal development upfront. Because of these specifications, the ESP8266 module fits perfectly in our system. As soon as the temperature data are sent to Arduino from the MLX90614 IR temperature sensor and the ESP32 camera module, respectively, the ESP8266 WiFi module helps the Arduino to transfer it to Firebase, which further processes the data and transfers it to the mobile application and *vice versa*, that is, sending data from Firebase (which was sent by the mobile application to let grant the access) to Arduino to be further processed.

3.2 Software

3.2.1 Firebase

Firebase (Figure 1) is a Google-acquired and maintained hosting platform that provides native software development kit. It has several features some of which include Authentication, Database, Server, Analytics, and much more. The features that we will be using include Authentication and Real-Time Database. Firebase Auth (Authentication) allows each user to be linked to the device of their choice and only access its data. It will also add a layer of security permitting only Reads and Writes from authorised devices. Firebase Realtime Database will allow us to send, store, and query data in JSON format in real-time. It is a NoSQL Web Socket-based implementation that allows us to update and notify the user in Real-Time.

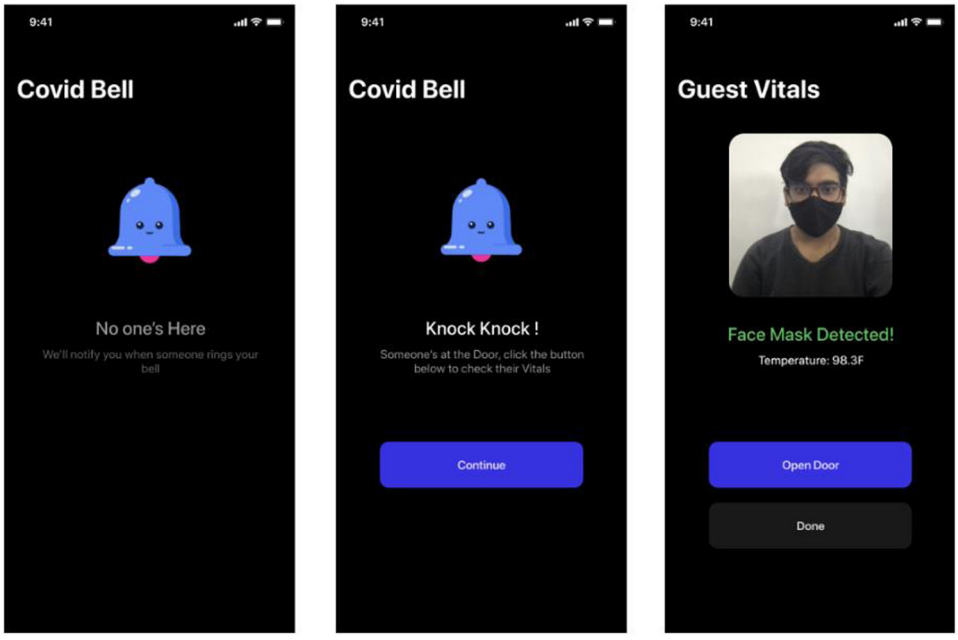

Figure 2: Mobile application working and interface.

3.2.2 Swift

Swift is an open-source, multipurpose, and multi-paradigm language based on Objective-C (Figures 1 and 2). It is developed and maintained by Apple Inc. and is also used to develop Native applications on Apple platforms like iOS and macOS. This is our language of choice for developing our client-side application as it provides robust performance leveraging Native APIs to give users a smooth experience. It offers multiple frameworks for developing applications but the one we use is UIKit [18]. It will not only allow us to develop the User Interface but also implement several client-side logics as per requirements.

3.2.3 CoreML

CoreML is a native framework provided by Apple to enable the use of machine learning in iOS mobile applications (Figure 1). It allows the user to build and train complex machine learning models in comparatively less time as it provides eGPU (external Graphical Processing Unit) support. This proves to be flexible for users as they can either write their models natively or convert the models written using other libraries like Keras, TensorFlow, or PyTorch using CoreML tools. Additionally, it also provides popular open-source models so users can either directly use them or fine-tune them according to their data set. One can easily train using a series of inputs received over time. It further allows the user to use specific libraries like Vision which is used for image analysis using Computer Vision, Natural Language for text-based applications, and Sound Analysis for recognising different sounds in audio.

4 Methodology and implementation

4.1 Hardware

The ultrasonic sensor sends a signal when a person's hand is found in proximity to it. This triggers the system as a whole, and the Arduino Uno gets activated. As soon as the activation takes place, the MLX90614 sensor notes the temperature of the visitor and sends it to Arduino, along with the ESP32 camera sending the captured image to Arduino. The Arduino collectively sends this information, *i.e.*, the temperature and captured image to Firebase through ESP8266 WiFi Module or HM10 BlueTooth module (as required).

4.2 Software

As the image is captured by (Figure 1), it is uploaded to Firebase Realtime Database as a Base-64 string along with the temperature and time stamp of the captured

image. Once the image is uploaded to Firebase RTDB, the client-side library automatically triggers a call to the database for querying data as it works on Web-Socket-based implementation, which is responsible for its real-time behaviour. This not only reduces the number of times an API is called but also allows us to have a working application that reacts in real time. The iOS app will query the data and notify the user about the visitor. The user can then check the vitals of the visitor such as temperature on the app and trigger the opening of the door by sending a confirmation to Firebase RTDB, which is then received by Arduino. Arduino triggers the servo motor to open the Door as per the command received. The vitals are analysed on the app itself, which in return alerts the user whether or not it is safe to let the visitor in. The Vitals are analysed in the following way:

- The temperature is analysed by checking whether the user's recorded temperature falls in the safe range or not.
- To analyse and detect face masks in the image, the app uses a swift written library that makes use of (AIZOOTech), which is trained with 7971 custom images in addition to the public data set available at WIDER Face and MAFA. The model resizes the image captured from the camera into 260 × 260. It comprises 24 layers in total while the backbone layer only consists of eight layers to make the model light. What makes the model robust is that the convolution layers are merged with the batch normalisation layers; therefore, the output is generated faster.

In order to make the above process more efficient in case of low network connection, a Bluetooth module on the hardware allows forming a connection with the iOS App using BLE. This way all the communication happens over Bluetooth as opposed to via the cloud. The data are captured by the hardware and sent over Bluetooth as encoded data. Upon receiving these data, it is cached by the App and processed as mentioned earlier. The final trigger for opening the door is also sent through this way or path. The cached data can then be saved or stored in a remote database for future work. This way of communication is integrated as a fall-back mechanism when both the modules are close-by and a network connection over the world wide web cannot be established. This makes the system more efficient and reliable for all types of scenarios.

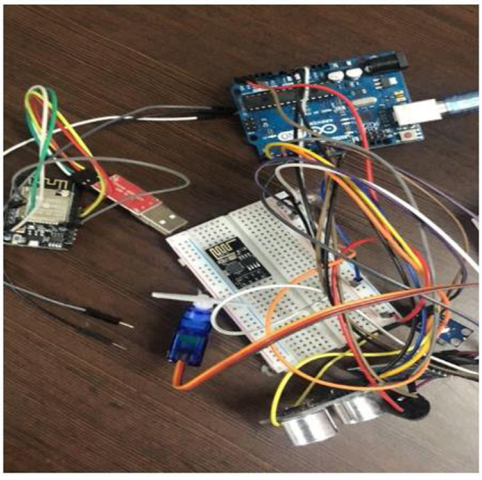

Figure 3: COVID bell hardware circuit with all the components.

5 Results and discussion

The pandemic has affected our lives drastically, which led to a lot of innovations and inventions using cutting-edge technology to help prevent this disease from prevailing. With the same intention, we have developed our system to help reduce the spread of COVID-19.

A user simply needs an Arduino Uno, an ESP 32 Cam3, Ultrasonic Sensor, MLX90614 Temperature Sensor, Servo Motor, ESP 8266 WiFi Module, HM10 Bluetooth Module, and a buzzer (Figures 1 and 3) to make an inexpensive smart doorbell system.

The difference between the COVID bell and other smart doorbells is manifold. First, the COVID bell is less expensive than other smart doorbells. Second, rather than being an integrated system, the COVID bell is a dispersed system, allowing for faster error identification and diagnosis. For example, instead of removing the entire bell system, if any component is not working, the user can easily identify and repair or replace the parts by checking each individual device. Third, because most smart doorbells are pricey equipment, they may be stolen. COVID bell, on the other hand, simply requires an interchangeable ultrasonic sensor to be placed outside the home, making it a far better security device alternative. Finally, unlike current smart doorbells, which are only available at selected retailers, the components of COVID bell are widely available.

While the COVID bell has some advantages over a traditional doorbell [5,7–9] in terms of functionality and security, it does have a few security and privacy concerns. Since the hardware components are connected to Arduino Uno, it is possible to use the IP address of Arduino Uno and hack the device to provide unauthorised visitor access or open the door without the owner's permission. We advise users to keep their networks password-protected.

Our technology also photographs visitors without their permission and stores the images on a private cloud server. On the other hand, the iOS app provides an additional layer of security as it only permits the owner to grant access to the visitor.

6 Future scope

We plan to use the system's data, which shows people's average temperatures and the frequency with which they wear masks, in the future. Firebase or the remote database used for Bluetooth data can be used to retrieve these data. These data can be analysed to determine how seriously people continue to take the pandemic and the risks of COVID spreading to various regions. Various machine learning and statistical models can be used to determine this. The government can use this information to determine which areas have a higher spread rate and require lockdowns. They can observe the seriousness with which individuals approach and adhere to COVID norms. In addition, we intend to integrate the same system with an Android application that can be easily implemented using Flutter and has the same design and specifications. The Android app can be used to implement the hardware and cloud system with minimal modifications.

Because their faces will be captured in the same manner as those of those who go through the authentication process, this system can also be customised to detect intruders who do not ring the doorbell. When looking over the house from a distance, this can be extremely helpful.

The brilliant doorbell can be associated with the web, permitting property holders to get to it through their iOS gadget from a distance. It is simple to adjust the temperature from a distance thanks to the temperature sensor, which can also be used to monitor the temperature inside and outside the house. Homeowners can create a more comprehensive smart home solution by integrating their smart doorbell with other smart home devices like security cameras, smart locks, and home automation systems using the Internet of Things. Data can be collected and analysed using IoT devices to learn about usage patterns and other factors that can help make products and user experiences better. The smart doorbell and temperature sensor can be programmed to send individual notifications to the homeowner's iOS device. The homeowner, for instance, can be notified when the temperature in the house reaches a certain level or when the doorbell rings while they are away.

7 Conclusion

The project uses mainly an Arduino, ESP32 Cam, WiFi and Bluetooth module, a few sensors, a Machine Learning algorithm, and an iOS application. The COVID bell is designed to notify the user whenever someone is standing at their door and to also capture and send the user their image. Users can stay connected to their homes while travelling, thanks to this system, ensuring their health and privacy. The project demonstrates a comprehensive, smart, and quick alternative for putting in place a smart doorbell. The project also contributes to the prevention of COVID and ensures the user's safety. Homeowners can remotely access the smart doorbell from their iOS device by connecting it to the Internet. The smart doorbell and temperature sensor can provide notifications and monitoring in real-time, allowing homeowners to remain aware of any potential changes or issues. Accessibility for people with disabilities can be improved with iOS-based smart doorbells. On their iOS device, a person who is hard of hearing or deaf can, for instance, receive visual alerts whenever the doorbell rings. After transferring the data from the hardware to the software and analysing it with Swift and CoreML, we inferred from the results that the desired outcomes were achieved, the user was shielded from potential dangers, and the solution made it simpler for the user to monitor the person standing on the door.

Funding information: The authors declare that there is no funding applicable regarding the publication of this article.

Author contributions: The authors confirm their contribution to the article as follows: Literature survey, Design and Implementation Vaibhav Kejriwal and Vaishnavi Patel; methodology by Riddhi Gupta and Utkarsh Dixit, supervision by Abhishek Gudipalli. All authors reviewed the final version of the manuscript.

Conflict of interest: The authors declare that there is no conflict of interest regarding the publication of this article.

Informed consent: Informed consent was obtained from all individuals included in this study.

Ethical approval: The conducted research is not related to either human or animals use.

Data availability statement: The datasets generated during and/or analysed during the current study are available from the corresponding author on reasonable request.

References

[1] R. V. S. Lalitha, K. Kavitha, N. V. Krishna Rao, G. Rama Mounika, and V. Sandhya, "Smart surveillance with smart doorbell," *Int. J. Innovative Technol. Explor. Eng.* (*IJITEE*), vol. 8, no. 8, p. 1841, June 2019. ISSN: 2278-3075.

[2] M. Senagala, "Rethinking smart architecture: Some strategic design frameworks," *Int. J. Archit. Comput.*, vol. 4, no. 3, pp. 33–46, 2006.

[3] J. C. Augusto and C. D. Nugent, "Smart homes can be smarter," In *Designing smart homes*, Springer, Heidelberg, 2006, pp. 1–15.

[4] D. Cook and S. Das, *Smart environments: technology, protocols and applications*, vol. 43, John Wiley & Sons, USA, 2004.

[5] S. Vig and S. Vig, "Smart doorbell security system and method to identify visitors," *US Patent App*, 2010. 12/861,427.

[6] O. Lehmann, M. Bauer, C. Becker, and D. Nicklas, "From home to world-supporting context-aware applications through world models," In *Pervasive Computing and Communications, 2004. PerCom 2004. Proceedings of the Second IEEE Annual Conference on*, IEEE, 2004, pp. 297–306.

[7] T. J. Huisking, "Method and apparatus for unlocking/locking a door and enabling two-way communications with a door security system via a smartphone," *US Patent App*, Sep. 7 2012. 13/606,690.

[8] C. -T. Yang and S. -H. Liu, "Smart doorbell with energy saving method," *US Patent 20*, May 21 2015, 150,138,353.

[9] A. M. Fadell, M. L. Rogers, Y. Matsuoka, D. Sloo, S. Honjo, S. A. McGaraghan, et al., "Visitor feedback to visitor interaction with a doorbell at a smart-home," *US Patent App*, Dec. 31 2014, 14/587,966.

[10] Amazon, "Amazon dash button," Nov. 22 2015. [Online]. amazon.com.

[11] P. Fernandez and M. Kesselman, "Thinking through the internet of things," *Lib. Hi Tech. N.*, vol. 32, no. 5, p. 4, 2015.

[12] "Dashbell: A Low-cost Smart Doorbell System for Home Use", 2017.[Online]. doi: 10.48550/arXiv.1706.09269.

[13] A. Ben Thabet and N. B. Amor, "Enhanced smart doorbell system based on face recognition," *2015 16th International Conference on Sciences and Techniques of Automatic Control and Computer Engineering* (*STA*), 2015, pp. 373–377. doi: 10.1109/STA. 2015.7505106.

[14] S. B. Prof, A. F. Sahu, K. K. Paswan, P. V. Tandi, Chunchawar, and R. D. Pooja, "IoT and AI based smart doorbell system," *Int. J. Creative Res. Thoughts* (*IJCRT*), vol. 6, no. 1, p. 1992, February 2018. ISSN: 2320-2882.

[15] H. H. Lwin, A. S. Khaing, and H. M. Tun, "Automatic door access system using face recognition," *Int. J. Sci. Technol. Res.*, vol. 4, no. 6, pp. 294–299, 2015.

[16] V. Patel, S. Kanani, T. Pathak, P. Patel, M. I. Ali, and J. Breslin, "A demonstration of smart doorbell design using federated deep learning," *CoRR arXiv:2010.09687v1* [*cs.DC*], 2017. doi: 10.48550/arXiv.2010.09687.

[17] Y. Liu, A. Huang, Y. Luo, H. Huang, Y. Liu, Y. Chen, et al., "Fedvision: An online visual object detection platform powered by federated learning," *arXiv:2001.06202* [*cs.LG*], 2020.

[18] Q. Yang, Y. Liu, T. Chen, and Y. Tong, "Federated machine learning: Concept and applications," *CoRR abs/1902.04885*, 2019. arXiv:1902.04885. doi: 10.48550/arXiv.1902.04885.

[19] J. Konečný, H. B. McMahan, F. X. Yu, P. Richtárik, A. T. Suresh, and D. Bacon, "Federated learning: Strate- gies for improving communication efficiency," *CoRR. abs/1610.05492*, p. arXiv:1610.05492, 2016, doi: 10.48550/arXiv.1610.05492.

[20] P. Liu, B. Qi, and S. Banerjee, "EdgeEye: An edge service framework for real-time intelligent video analytics," In *Proceedings of the 1st International Workshop on Edge Systems, Analytics and Networking* (*Munich, Germany*) (*EdgeSys'18*), New York, NY, USA, Association for Computing Machinery, 2018, pp. 1–6. doi: 10.1145/3213344.3213345.

[21] P. Bogetoft, D. L. Christensen, I. Damgård, M. Geisler, T. Jakobsen, M. Krøigaard, et al. "Secure multiparty computation goes live," vol. 5628, 2009, pp. 325–343. doi: 10.1007/978-3-642-03549-4_20.

[22] T. Elgamal, S. Shi, V. Gupta, R. Jana, and K. Nahrstedt, "SiEVE: Semantically encoded video analytics on edge and cloud," *arXiv:2006.01318* [*cs.DC*], 2020.

[23] A. Das, M. Wasif Ansari, and R. Basak, "Covid-19 face mask detection using TensorFlow, Keras and OpenCV," *2020 IEEE 17th India Council International Conference* (*INDICON*), 2020, pp. 1–5. doi: 10.1109/INDICON49873.2020.9342585.

[24] S. Kumar, S. Kaur, and S. Singh, "Development of COVID-19 detection smart doorbell based on IoT," *J. Adv. Res. Med. Sci. Technol.*, vol. 8, no. 4, p. 7, 2021.

[25] C. K. Gomathy and D. Satyaa, "Study on IOT smart doorbells," *Int. Res. J. Eng. Technol.* (*IRJET*) *E-ISSN: 2395-0056*, vol. 8, no. 9, 2021.

[26] W. -J. Li, C. Yen, Y. -S. Lin, S. -C. Tung, and S. Huang, "JustIoT Internet of Things based on the Firebase real-time database," *2018 IEEE International Conference on Smart Manufacturing, Industrial & Logistics Engineering* (*SMILE*), 2018, pp. 43–47. doi: 10.1109/SMILE. 2018.8353979.

[27] S. Pande and M. S. Chetty, "Analysis of capsule network (Capsnet) architectures and applications," *J. Adv. Res. Dyn. Control. Syst.*, vol. 10, no. 10, pp. 2765–2771, 2018.

[28] S. Pande and M. S. Chetty, "Bezier curve based medicinal leaf classification using capsule network," *Int. J. Adv. Trends Comput. Sci. Eng.*, vol. 8, no. 6, pp. 2735–2742, 2019.

[29] S. D. Pande and M. S. R. Chetty, "Fast medicinal leaf retrieval using CapsNet," In: *International Conference on Intelligent and Smart Computing in Data Analytics. Advances in Intelligent Systems and Computing*, S. Bhattacharyya, J. Nayak, K. B. Prakash, B. Naik, A. Abraham, (eds), vol. 1312, 2021.

[30] P. Sunitha, N. Ahmad, R. K. Barbhuiya, V. K. Gunjan, and M. D. Ansari, *4th International Conference on Communications and Cyber-Physical Engineering, ICCCE*, 828, Impact of Covid-19 on Education, 2021, pp. 1191–1197.