AI Smart Doorbell

Tanvir Md Raiyan
Department of CSE
Ahsanullah University of Science and Technology
Dhaka, Bangladesh
tanvirraiyan456@gmail.com

Musaddique Ali Erfan
Department of CSE
Ahsanullah University of Science and Technology
Dhaka, Bangladesh
musaddique.cse.20200204049@aust.edu

Adibul Haque
Department of CSE
Ahsanullah University of Science and Technology
Dhaka, Bangladesh
adibul.cse.20200204029@aust.edu

Yousuf Ali Shafin
Department of CSE
Ahsanullah University of Science and Technology
Dhaka, Bangladesh
yousuf.cse.20200204037@aust.edu

***Abstract*—The proliferation of Internet of Things (IoT) devices has transformed home security, enabling affordable and intelligent solutions for everyday use. This paper presents the design and implementation of a smart doorbell system that integrates real-time video streaming, face recognition, and mobile app interaction. Built using a Raspberry Pi 4 as the central processing unit, an ESP32 for peripheral control, and an Android application for user interaction, the system offers features such as remote door unlocking, face registration, and instant notifications. The face recognition model, deployed on the Raspberry Pi using OpenCV, allows dynamic retraining to accommodate new users. Communication between components is facilitated by the MQTT protocol, ensuring efficient and secure data exchange, while Firebase Cloud Messaging (FCM) delivers real-time alerts to users. This low-cost, scalable solution demonstrates the potential of combining open-source hardware and software to enhance residential security. We discuss the system architecture, hardware components, methodology, and performance outcomes, highlighting its practicality and adaptability for smart home applications.**

***Index Terms*—Internet of Things (IoT), Smart Doorbell, Face Recognition,Raspberry Pi, ESP32, MQTT Protocol, Firebase Cloud Messaging (FCM), OpenCV, Home Security, Android Application**

I. Introduction

Home security systems have evolved significantly with the advent of IoT technologies, shifting from traditional passive devices to interactive, intelligent systems. Conventional doorbells, limited to auditory alerts, are being replaced by smart alternatives that offer video surveillance, remote access, and personalized recognition capabilities. However, many commercial smart doorbells, such as Ring or Nest, come with high costs and subscription fees, making them inaccessible to budget-conscious consumers. This project addresses these challenges by developing an open-source, cost-effective smart doorbell using widely available hardware and software tools.

The proposed system leverages the computational power of the Raspberry Pi 4, the versatility of the ESP32 microcontroller, and an Android application to create a cohesive smart doorbell ecosystem. Key features include live video streaming via MJPEG, a face recognition model for visitor identification, remote door unlocking, and real-time notifications through Firebase Cloud Messaging (FCM). The system employs the MQTT protocol for efficient communication between the Raspberry Pi and ESP32, ensuring low latency and reliability. Additionally, the face recognition model, implemented with OpenCV, supports dynamic retraining, allowing users to register new faces directly from the mobile app.

This paper outlines the hardware components, system architecture, and development methodology of the smart doorbell. The solution aims to provide an accessible, customizable alternative to commercial products, contributing to the growing field of DIY IoT applications for smart homes. The following sections detail the literature review, hardware setup, implementation process, and conclusions drawn from the project.

II. Literature Review

Various innovative approaches to smart doorbell technology, each addressing unique aspects of security, health, and user experience. Traditional home security systems are often easily compromised and require costly installations. To address these limitations, researchers have explored smart home security techniques, with face recognition emerging as a promising avenue.

Jain et al. [4] propose an IoT-based smart doorbell using a Raspberry Pi that employs face recognition for secure home access. Their system alerts users via Google Firebase on Android devices and stores visitor images for authentication. It offers a cost-effective solution, triggering a buzzer upon detecting unauthorized intrusion and notifying users.

Dadhich et al. [1] introduce a contactless IoT doorbell designed to mitigate the spread of COVID-19 through non-contact temperature measurement. Their system utilizes a NodeMCU and an MLX90614 infrared temperature sensor to screen visitors for fever, logging readings in a Firebase database and providing a companion mobile app. The system notifies homeowners of visitors potentially having a fever.

Fig. 1. Photo of AI Smart Doorbell

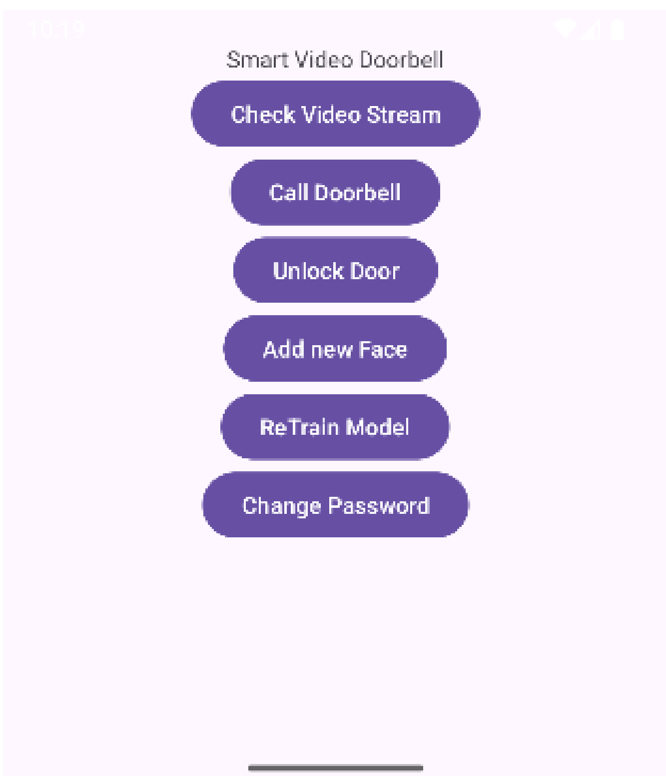

Fig. 2. Photo of Android App

Gudipalli et al. [3] present the ”COVID bell,” a smart doorbell solution focused on minimizing surface contact to prevent COVID-19 spread. It integrates an ultrasonic sensor for touchless activation, a temperature sensor, and a camera for mask detection. Data is transmitted to a mobile app, enabling users to remotely grant or deny entry based on health and safety parameters.

Pawar et al. [5] propose a smart home security system combining IoT and face recognition. Their system uses a camera connected to a Raspberry Pi, along with PIR and ultrasonic sensors, to detect motion and capture images. Face recognition is performed using the Local Binary Pattern (LBP) algorithm. The door unlocks if a face matches a known home member; otherwise, the doorbell rings.

Giorgi et al. [2] describe a smart doorbell system employing both iris and voice recognition for user identification. Their system uses machine learning algorithms for local computations, implementing edge computing analytics to determine user identity by combining voice and iris biometrics. Implemented on reconfigurable hardware, it accelerates intensive recognition functions while maintaining reasonable power consumption.

These diverse approaches collectively advance smart doorbell technology, offering enhanced security, health monitoring, and user convenience.

III. Hardware Components

The smart doorbell system relies on a combination of affordable, off-the-shelf hardware components, carefully selected to balance cost, performance, and functionality. The key components are:

A. *Raspberry Pi 4*

Serves as the central processing unit, handling video processing, face recognition, and communication with the mobile app. It is powered by a 5V/3A power supply and uses a MicroSD card (16GB or higher) for storing the operating system and local data.

B. *Camera Module*

A high-definition camera compatible with the Raspberry Pi, providing video input for MJPEG streaming and face recognition.

C. *ESP32*

A low-power microcontroller that interfaces with the doorbell push button and communicates with the Raspberry Pi via the MQTT protocol. It ensures efficient signal handling and peripheral control.

D. *Doorbell Push Button*

A simple mechanical button that triggers the system when pressed, sending an input signal to the ESP32.

E. *Keypad*

Connected to the ESP32, this component allows users to input a password to unlock the door, providing an alternative to app-based or face-based authentication.

F. Power Supply

A 5V/3A adapter powers the Raspberry Pi and other components, ensuring stable operation.

G. MicroSD Card

Stores the Raspberry Pi's operating system (Raspbian), application code, and trained face recognition models.

H. Jumper Wires and Connectors

Facilitate electrical connections between the Raspberry Pi, ESP32, camera module, and doorbell button.

These components were chosen for their compatibility, affordability, and widespread availability, making the system replicable by hobbyists and researchers. The total cost is significantly lower than that of commercial smart doorbells, aligning with the project's goal of accessibility.

IV. Methodology

The development of the smart doorbell involved a multi-layered approach, integrating hardware setup, software implementation, and system testing. The methodology is divided into six key phases:

A. Hardware Assembly

The Raspberry Pi 4 was configured with Raspbian OS on a 16GB MicroSD card and connected to the HD camera module via the CSI port. The ESP32 was wired to the doorbell push button and a 4x4 keypad using jumper wires, with both devices powered by a 5V/3A supply. Connections between the Raspberry Pi and ESP32 were established for MQTT communication.

B. Software Architecture

A Python Flask API was deployed on the Raspberry Pi to manage requests from the Android app, including video streaming, door unlocking, and face registration. The ESP32 was programmed to publish doorbell press events and keypad inputs to an MQTT broker on the Raspberry Pi. FCM was integrated to send notifications to the mobile app.

C. Face Recognition Implementation

OpenCV was installed on the Raspberry Pi to process video streams from the camera module. A pre-trained face detection model (e.g., Haar Cascade) was used to identify faces, followed by a custom-trained recognition model (Resnet) to classify them. The model was designed to support retraining with new faces added via the Android app.

D. Keypad Authentication

The ESP32 was programmed to read password inputs from the keypad. Upon entry, the password is sent to the Raspberry Pi via MQTT, where it is compared against a stored hash (e.g., using SHA-256 for security). If the password matches, the Raspberry Pi triggers the door unlock mechanism (assumed to be a relay or servo, not specified in the provided data).

E. Android Application Development

The Android app was developed with features including MJPEG video streaming, face registration, remote door unlocking, and real-time notifications. It communicates with the Flask API and receives FCM alerts for doorbell presses or unrecognized visitors. The app does not directly handle keypad inputs, as this is managed locally by the ESP32 and Raspberry Pi.

F. System Integration and Testing

The system was tested for end-to-end functionality. When the doorbell button is pressed, the ESP32 notifies the Raspberry Pi via MQTT, triggering video capture and face recognition. If a face is recognized, the app receives a notification with the visitor's identity; if not, the user can register the face or unlock the door remotely. Alternatively, entering the correct password on the keypad unlocks the door without app intervention. Latency, recognition accuracy, and password validation were assessed, with adjustments made to optimize performance.

The MQTT protocol ensured low-latency communication, while Flask provided a lightweight API for app interaction. The face recognition model achieved satisfactory accuracy after initial training, with retraining capabilities enhancing its adaptability.

V. Conclusion

The smart doorbell system presented in this paper demonstrates a successful fusion of IoT technologies into a practical, multi-modal home security solution. By integrating a Raspberry Pi 4, ESP32, camera module, keypad, and Android app, the system delivers advanced features—video streaming, face recognition, password-based unlocking, and real-time notifications—at a significantly lower cost than commercial alternatives. The use of MQTT for device communication, FCM for notifications, and OpenCV for face recognition ensures efficiency and adaptability, while the keypad adds a robust alternative authentication mechanism.

This project underscores the power of open-source hardware and software in creating scalable, user-friendly IoT devices. The combination of biometric (face recognition) and knowledge-based (password) authentication caters to diverse use cases, from identifying frequent visitors to providing access during app unavailability. Future enhancements could include improving face recognition accuracy with deep learning models (e.g., FaceNet), adding cloud-based video storage, or incorporating additional sensors like motion detectors. The system serves as a compelling proof of concept for affordable smart home solutions, inviting further exploration and refinement by the research and DIY communities.

References

[1] Simran Dadhich, Pritsam Dabre, Ruben Dabreo, and Prachi Raut. Contactless iot doorbell for covid-safe household. In *2021 IEEE 9th Region 10 Humanitarian Technology Conference (R10-HTC)*, pages 01–04, 2021.

[2] Roberto Giorgi, Nicola Bettin, Sara Ermini, Francesco Montefoschi, and Antonio Rizzo. An iris+voice recognition system for a smart doorbell. In *2019 8th Mediterranean Conference on Embedded Computing (MECO)*, pages 1–4, 2019.

[3] Abhishek Gudipalli, Vaibhav Kejriwal, Vaishnavi Patel, Riddhi Gupta, and Utkarsh Dixit. COVID bell – A smart doorbell solution for prevention of COVID-19. *Paladyn, Journal of Behavioral Robotics*, 14(1):20220115, 2023.

[4] Abhishek Jain, Surendra Lalwani, Suyash Jain, and Varun Karandikar. Iot-based smart doorbell using raspberry pi. In Raj Kamal, Michael Henshaw, and Pramod S. Nair, editors, *International Conference on Advanced Computing Networking and Informatics*, pages 175–181, Singapore, 2019. Springer Singapore.

[5] Suraj Pawar, Vipul Kithani, Sagar Ahuja, and Sunita Sahu. Smart home security using iot and face recognition. In *2018 Fourth International Conference on Computing Communication Control and Automation (ICCUBEA)*, pages 1–6, 2018.