Smart Doorbell with Integrated Lock System

A PROJECT REPORT

submitted in partial fulfillment of the requirements

for the award of the degree of

BACHELOR OF TECHNOLOGY

in

COMPUTER SCIENCE AND ENGINEERING

by

Thanga Sai Naga Anirudh -21BAI1861

Uday Sri Vishnu – 21BAI1373

Shiva Sannidh – 21BAI1431

Sai Madhu – 21BAI1874

under the guidance of

Dr. soumya Ranjan

SCHOOL OF COMPUTER SCIENCE ENGINEERING

VIT CHENNAI

April 2024

ABSTRACT

This abstract introduces an innovative Smart door lock system incorporating state-of-theart face detection technology. This system merges advanced features to deliver both convenient and secure access control for residences or facilities. By harnessing the capabilities of face detection, users can effortlessly unlock their doors without the need for traditional keys. This integration ensures a heightened level of security, permitting only authorized individuals with registered faces to gain entry. Furthermore, the system includes features such as real-time monitoring and activity logs, allowing homeowners to monitor access and receive notifications of any unauthorized attempts. The integration of face detection technology into the smart door lock system marks a significant leap forward in home security, providing seamless integration and heightened protection for modern living spaces.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

ABBREVIATIONS AND NOMENCLATURE

The ESP32 originates from Espressif Systems, a Shanghai-based company, and is manufactured using TSMC's 40 nm fabrication technology. It builds on the foundation set by the ESP8266 microcontroller. The Arduino Integrated Development Environment (IDE) is a platform commonly used for programming and development..

CHAPTER I

1. INTRODUCTION

1.1 INTRODUCTION

This introduction introduces an advanced smart door lock system incorporating stateof-the-art facial recognition technology. Through the fusion of cutting-edge features, this system presents a secure and convenient solution for access control in both residential and commercial settings. By leveraging facial recognition, traditional keys are rendered obsolete, enabling users to unlock doors effortlessly. This technology guarantees enhanced security by permitting entry solely to authorized individuals with registered facial profiles. Additionally, the system offers real-time monitoring capabilities and maintains comprehensive activity logs, allowing homeowners to monitor access events and promptly receive alerts regarding any unauthorized attempts. The integration of facial recognition into the smart door lock system signifies a significant advancement in home security, providing seamless integration and advanced protection for contemporary living spaces.

1.1.1 Motivation

Traditional lock and key systems have long been plagued by issues like key misplacement, unauthorized duplication, and the inconvenience of carrying multiple keys. Moreover, concerns about security breaches and unauthorized access attempts persist among homeowners. Face detection technology addresses these challenges by ensuring that only individuals with registered facial data can unlock the door, greatly reducing the risk of unauthorized entry. The smart door lock system with integrated face detection is driven by the desire for convenient, secure, and efficient access control, coupled with enhanced security measures and real-time monitoring capabilities. By overcoming the limitations of traditional lock and key systems, this innovative solution aims to offer peace of mind and revolutionize the approach to safeguarding homes and establishments.

1.1.2 Objectives

The project's goal is to conceive and execute a smart door lock system featuring integrated face detection technology. Its objective is to devise a secure and userfriendly access control solution, eradicating the necessity for conventional keys. The project emphasizes the development of sturdy hardware and software elements, crafting an intuitive user interface, instating rigorous security protocols, conducting exhaustive testing and validation, contemplating scalability and compatibility, and furnishing comprehensive documentation and assistance. The ultimate aim is to furnish a dependable and user-centric smart door lock system, heightening security and providing seamless access control for both residential and commercial settings.

1.1.3 Scope of the Work

The Autonomous Slot-car system comprises a unique vehicle, track sensors, and computer control software. Its purpose is to enrich the single-player experience of Scalextric products. The system will offer a variety of features, allowing the car to competitively race against the customer at different difficulty settings.

1.2 ORGANIZATION OF THESIS

The thesis structure for the smart doorbell with integrated lock system comprises eight sections: introduction, literature review, methodology, system architecture, implementation and evaluation, results and analysis, discussion, and conclusion and future work. In the introduction, the project and its objectives are outlined, while the literature review delves into existing research on smart doorbells and integrated lock systems. The methodology section delineates the research methodology and design approach employed. Subsequently, the system architecture section illustrates the proposed architecture, and the implementation and evaluation section elaborates on the implementation process and evaluations conducted. Findings are presented in the results and analysis section, followed by a discussion, conclusion, and suggestions for future work.

Table 1: Design Specifications

S. No.	**Component name**	**Specification**	
1.	ESP32 Camera Module Microcontroller.	Overall Operating Voltage for controller: 5V	Current rating: 6mA – 310mA
2.	NPN transistor [TIP122]	Operating voltage: 5V	
3.	PN junction diode [1N4007]	Operating Reverse Voltage (Vr) : 1000V	
4.	7805 Regulator	Output Voltage: 5V	
5.	Solenoid lock	Operating Voltage: 12V	

6.	DC Source	Output Voltage: 12V
7.	Resistor	Resistance: 100 Ω, 1KΩ
8.	Capacitor	Capacitance: 20μF
9.	Push Button	

CHAPTER II

2. PROJECT DESCRIPTION

2.1 OVERVIEW OF PROJECT

This research aims to delve into the concept of a smart doorbell with an integrated lock system, focusing on exploring its potential benefits, challenges, and implications concerning privacy, security, and user experience. By analyzing existing research papers, the goal is to glean insights into the current landscape of smart door lock systems and pinpoint areas for further development and enhancement. The study investigates various aspects, including the integration of face detection technology into smart door lock systems, privacy concerns, security and reliability considerations, user experience and acceptance, as well as future directions and challenges. Ultimately, the research endeavors to contribute to the understanding and advancement of smart doorbells with integrated lock systems.

2.2. MODULES OF THE PROJECT

The smart door lock system incorporating face detection is structured around four main modules: camera-based access control, facial data collection and training, user interaction interface, and security and privacy protection.

2.1.1 Camera-Based Access Control Module
This module is responsible for capturing images and transmitting them to the main device.

2.1.2 Facial Data Collection and Training Module
Here, a collection of facial images is gathered to build a dataset for training the face detection algorithm.

2.1.3 User Interaction Interface Module
This module focuses on creating an intuitive interface for users to configure and operate the smart door lock system.

2.1.4 Security and Privacy Protection Module

This module is dedicated to implementing security measures to protect the facial data stored and to ensure user privacy concerns are addressed..

CHAPTER III

3. DESIGN Of Smart Doorbell with Integrated Lock System.

3.1 DESIGN APPROACH

To develop a smart doorbell with an integrated lock system, the design strategy involves several key steps. These include defining the system architecture, implementing face detection and recognition algorithms, integrating with the door lock system, designing a user-friendly interface, and ensuring security and privacy.

The system architecture can be structured around modules such as face detection, face recognition, and automatic door access control. Integrating these modules with the door lock system facilitates seamless automatic door access control based on facial recognition.

Designing a user-friendly interface is essential for easy management of the smart doorbell system. This interface should be intuitive and straightforward for users to configure and control the system effectively.

Ensuring security measures is paramount to protect stored facial data and address privacy concerns. Robust security protocols should be implemented to safeguard sensitive information and ensure user privacy.

By following this approach, a smart doorbell with an integrated lock system can be developed to provide efficient and secure access control for residential and commercial environments.

3.1.1 Codes and Standards

To uphold the quality and compliance of a smart doorbell with an integrated lock system, adherence to codes and standards is imperative.
Privacy standards must be observed to safeguard user data and uphold confidentiality. Similarly, security standards should be implemented to fortify protection against unauthorized access and cyber threats.
Interoperability standards are essential to ensure compatibility with existing smart home ecosystems. By adhering to these standards, the system can seamlessly integrate with other devices and platforms.
Accessibility standards play a vital role in promoting inclusivity and usability for individuals with disabilities. Upholding these standards ensures that the system can be accessed and operated by all users, regardless of their abilities.
Electrical and safety standards are critical to ensuring safe operation and compliance with regulations. Adhering to these standards mitigates potential risks and ensures the system's safe deployment and usage.

Installation and mounting standards provide guidelines for proper installation procedures, ensuring that the system is installed correctly and securely.
By adhering to these codes and standards, the smart doorbell with an integrated lock system can meet regulatory requirements, prioritize user privacy and security, and deliver a reliable and accessible product.

3.1.2 Realistic Constraints

The design of a smart doorbell with an integrated lock system necessitates consideration of various realistic constraints to ensure its effectiveness and compliance. Key factors include addressing privacy concerns associated with capturing and storing facial data and implementing robust security measures to thwart unauthorized access attempts.
Hardware limitations, such as size, power consumption, and thermal management capabilities, must be carefully optimized to maintain reliable performance.
Moreover, ensuring compatibility with existing smart home ecosystems and protocols is paramount, while prioritizing the development of a user-friendly interface that is intuitive and accessible to all users.
Compliance with electrical safety standards, certifications, and regulations is indispensable to guarantee safe operation and adherence to product compliance requirements.

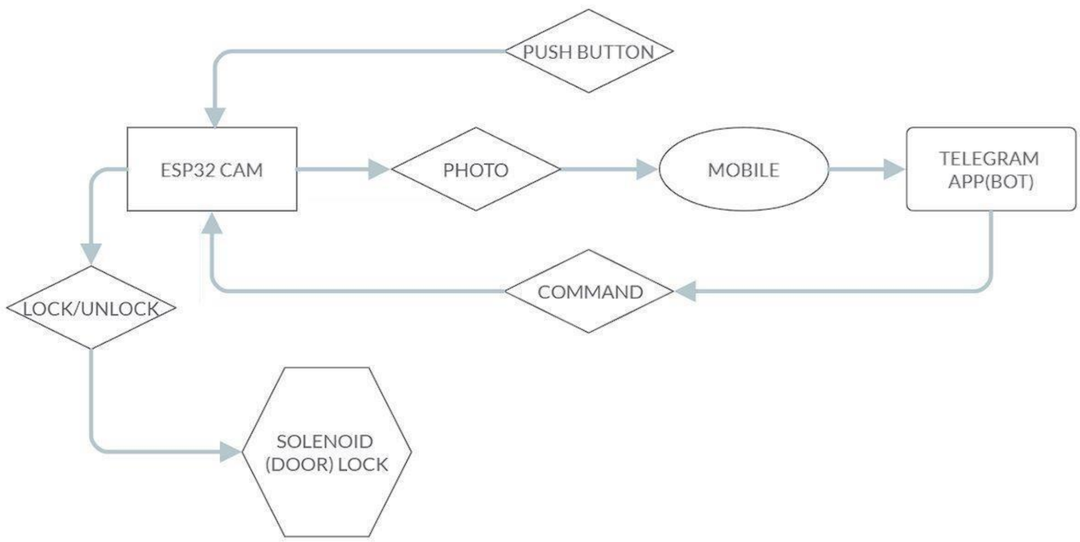
3.1.3 Alternatives and Tradeoffs

The design process of a smart doorbell with an integrated lock system involves careful consideration of various tradeoffs and alternatives. Factors such as wired versus wireless systems, facial recognition versus alternative authentication methods, integration with existing smart home ecosystems, cost versus features, privacy versus convenience, and security versus usability must all be weighed.

For instance, wired systems offer reliability but may necessitate professional installation, while facial recognition provides convenience but could potentially raise privacy concerns. Integrating with existing smart home ecosystems offers convenience but may limit customization options. Balancing robust security measures with userfriendly interfaces is essential.

By carefully considering these alternatives and tradeoffs, the smart doorbell with an integrated lock system can be tailored to meet the specific needs and preferences of users effectively.

DESIGN SPECIFICATIONS



CHAPTER IV

4. PROJECT DEMONSTRATION

4.1 INTRODUCTION

The smart doorbell with an integrated lock system serves as a connected device that enhances both home security and convenience. It enables homeowners to remotely monitor and manage access to their residences, providing them with peace of mind and effortless control.

The project aims to devise a smart doorbell with an integrated lock system that integrates face detection and recognition algorithms, ensuring secure and dependable access control. Throughout the design process, careful consideration is given to privacy concerns, security measures, hardware limitations, interoperability, usability, and regulatory compliance.
The thesis is structured into sections covering the introduction, literature review, methodology, system architecture, implementation and evaluation, results and analysis, discussion, and conclusion and future work, offering a comprehensive exploration of the project's objectives, processes, and outcomes.

4.2 ANALYTICAL RESULTS

Research papers and articles offer valuable insights into the design and implementation of smart doorbells with integrated lock systems. A smart doorbell, functioning as an internet-connected device, notifies homeowners when a visitor approaches the door. It activates either when the visitor presses the doorbell button or when motion sensors detect movement. Through a smartphone app, homeowners can view and communicate with visitors via the doorbell's integrated high-definition infrared camera and microphone. Additionally, certain smart doorbells provide the capability to remotely unlock the door using a smart lock.

The design process for smart doorbells with integrated lock systems must carefully address privacy concerns and implement robust security measures. This research underscores the necessity for comprehensive security management to safeguard data transmission between devices, as well as secure access within the product technology stack and other corporate systems.

CHAPTER V

5. CONCLUSION

5.1 COST ANALYSIS

5.2 SCOPE OF WORK

The scope of work for the smart doorbell with integrated lock system includes customizable security and privacy settings, innovative interaction designs, integration with IoT smart home systems, affordability, enhanced user experience, and seamless integration with smart lock systems.

5.3 SUMMARY

The smart doorbell with an integrated lock system serves as a connected device, elevating both home security and convenience. Its design approach conscientiously addresses privacy concerns, security measures, hardware limitations, interoperability, usability, and regulatory compliance.
The scope of work encompasses several critical aspects, including customizable security and privacy settings, innovative interaction designs, integration with IoT smart home systems, affordability, enhanced user experience, and seamless integration with smart lock systems.
The thesis is structured into sections covering the introduction, literature review, methodology, system architecture, implementation and evaluation, results and analysis, discussion, and conclusion and future work. The references cited offer valuable insights into the design and implementation of smart doorbells with integrated lock systems.

REFERENCES

1. CHALLENGING THE CONCEPT OF SMART DOORBELLS BY DESIGNING NEW INTERACTIONS BASED ON PRIVACY - Responsible Sensing Lab

2. CHALLENGING THE CONCEPT OF SMART DOORBELLS BY DESIGNING NEW INTERACTIONS BASED ON PRIVACY - TU Delft Repositories

3.A STUDY ON IOT SMART DOORBELLS - ResearchGate

4. Dashbell: A Low-cost Smart Doorbell System for Home Use – escholarship.

5. youtube links

APPENDICES

Appendix 1: CODING

```
#include <WiFi.h>
#include <WiFiClientSecure.h>
#include "soc/soc.h"
#include "soc/rtc_cntl_reg.h"
#include "esp_camera.h"
#include <UniversalTelegramBot.h>
#include <ArduinoJson.h>

// Replace with your network credentials const
char* ssid = "        "; //WiFi Name const char*
password = "          "; //WiFi Password

// Use @myidbot to find out the chat ID of an individual or a group
// You need to click "start" on a bot before it can message you
// Initialize Telegram BOT
String chatId = "XXXXXXXXXX";
String BOTtoken =
"XXXXXXXXXX:XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX";

bool sendPhoto = false;

WiFiClientSecure clientTCP;

UniversalTelegramBot bot(BOTtoken, clientTCP);

// Define GPIOs
#define BUTTON 13
#define LOCK 12
#define FLASH_LED 4

//CAMERA_MODEL_AI_THINKER
#define PWDN_GPIO_NUM     32
#define RESET_GPIO_NUM    -1
#define XCLK_GPIO_NUM      0
#define SIOD_GPIO_NUM     26
#define SIOC_GPIO_NUM     27

#define Y9_GPIO_NUM       35
#define Y8_GPIO_NUM       34
#define Y7_GPIO_NUM       39
#define Y6_GPIO_NUM       36
#define Y5_GPIO_NUM       21
#define Y4_GPIO_NUM       19
```

```
#define Y3_GPIO_NUM       18
#define Y2_GPIO_NUM        5
#define VSYNC_GPIO_NUM    25
#define HREF_GPIO_NUM  23 #define
PCLK_GPIO_NUM  22

        int  lockState  =  0;  String
 r_msg = "";


const unsigned long BOT_MTBS = 1000; // mean time between scan messages
unsigned long bot_lasttime; // last time messages' scan has been done

void handleNewMessages(int numNewMessages);
String sendPhotoTelegram();

String unlockDoor(){
if (lockState == 0) {
  digitalWrite(LOCK, HIGH);
  lockState = 1; delay(100);
  return "Door Unlocked.
  /lock";
} else{
 return "Door Already Unlocked. /lock"; }
}
String lockDoor(){
if (lockState == 1) {
  digitalWrite(LOCK, LOW);
  lockState = 0; delay(100);
  return "Door Locked.
  /unlock";
} else{
 return "Door Already Locked. /unlock"; }
}

String sendPhotoTelegram(){ const char*
  myDomain = "api.telegram.org";
  String getAll = "";
  String getBody = "";

  camera_fb_t * fb = NULL;
  fb = esp_camera_fb_get();
  if(!fb) {
    Serial.println("Camera capture
    failed"); delay(1000); ESP.restart();
    return "Camera capture failed";
  }

  Serial.println("Connect to " + String(myDomain));
```

```
if (clientTCP.connect(myDomain, 443)) { Serial.println("Connection
  successful");

 Serial.println("Connected to " + String(myDomain));

  String head = "--IotCircuitHub\r\nContent-Disposition: form-data;
name=\"chat_id\"; \r\n\r\n" + chatId + "\r\n--IotCircuitHub\r\nContent-Disposition:
form-data; name=\"photo\"; filename=\"esp32-cam.jpg\"\r\nContent-Type:
image/jpeg\r\n\r\n";
  String tail = "\r\n--IotCircuitHub--\r\n";

  uint16_t imageLen = fb->len; uint16_t extraLen
  = head.length() + tail.length(); uint16_t totalLen
  = imageLen + extraLen;

  clientTCP.println("POST /bot"+BOTtoken+"/sendPhoto HTTP/1.1");
  clientTCP.println("Host: " + String(myDomain)); clientTCP.println("Content-
  Length: " + String(totalLen));
  clientTCP.println("Content-Type: multipart/form-data; boundary=IotCircuitHub");
  clientTCP.println();
  clientTCP.print(head);

  uint8_t *fbBuf = fb->buf; size_t
  fbLen = fb->len; for (size_t
  n=0;n<fbLen;n=n+1024) {
   if (n+1024<fbLen) { clientTCP.write(fbBuf,
   1024);
    fbBuf += 1024;
   }
   else if (fbLen%1024>0) {
    size_t remainder = fbLen%1024; clientTCP.write(fbBuf,
    remainder);
   }
  }


  clientTCP.print(tail); esp_camera_fb_return(fb);

  int waitTime = 10000; // timeout 10 seconds
  long startTimer = millis();
  boolean state = false;

  while ((startTimer + waitTime) > millis()){
   Serial.print(".");
   delay(100); while
   (clientTCP.available()){
      char c = clientTCP.read();
```

```
      if (c == '\n'){ if
      (getAll.length()==0) state=true;
      getAll = ""; } else if (c != '\r'){
        getAll += String(c);
      } if
      (state==true){
        getBody += String(c);
      }
      startTimer = millis();
    } if (getBody.length()>0)
    break;
  }
  clientTCP.stop(); Serial.println(getBody);
 }
 else
 {
  getBody="Connected to api.telegram.org failed."; Serial.println("Connected
  to api.telegram.org failed.");
 } return
 getBody;
}


void handleNewMessages(int numNewMessages){
 Serial.print("Handle New Messages: ");
 Serial.println(numNewMessages);

 for (int i = 0; i < numNewMessages; i++){
  // Chat id of the requester
  String chat_id = String(bot.messages[i].chat_id); if
  (chat_id != chatId){
    bot.sendMessage(chat_id, "Unauthorized user", "");
    continue;
  }

  // Print the received message
  String text = bot.messages[i].text;
  Serial.println(text);

  String fromName = bot.messages[i].from_name;
  if (text == "/photo") {
    sendPhoto = true;
    Serial.println("New photo request");
  }
  if (text == "/lock"){ String r_msg =
    lockDoor();
    bot.sendMessage(chatId, r_msg, "");
  } if (text == "/unlock"){ String r_msg
  =                    unlockDoor();
  bot.sendMessage(chatId, r_msg, "");
```

```
  } if (text ==
  "/start"){
    String welcome = "Welcome to the ESP32-CAM Telegram Smart Lock.\n";
    welcome += "/photo : Takes a new photo\n"; welcome
    += "/unlock : Unlock the Door\n\n"; welcome +=
    "/lock : Lock the Door\n"; welcome += "To get the
    photo please tap on /photo.\n";
    bot.sendMessage(chatId, welcome, "Markdown"); }
 }
}


void setup(){
  WRITE_PERI_REG(RTC_CNTL_BROWN_OUT_REG, 0);
  Serial.begin(115200);
  delay(1000);

  pinMode(LOCK,OUTPUT);
  pinMode(FLASH_LED,OUTPUT);
  pinMode(BUTTON,INPUT_PULLUP);

  digitalWrite(LOCK, LOW);

  WiFi.mode(WIFI_STA);
  Serial.println();
  Serial.print("Connecting to ");
  Serial.println(ssid);
  WiFi.begin(ssid, password);
  clientTCP.setCACert(TELEGRAM_CERTIFICATE_ROOT);
  while (WiFi.status() != WL_CONNECTED) {
    Serial.print(".");
    delay(500);
  }
  Serial.println();
  Serial.print("ESP32-CAM IP Address: ");
  Serial.println(WiFi.localIP());

  camera_config_t config;
  config.ledc_channel = LEDC_CHANNEL_0;
  config.ledc_timer = LEDC_TIMER_0;
  config.pin_d0 = Y2_GPIO_NUM;
  config.pin_d1 = Y3_GPIO_NUM;
  config.pin_d2 = Y4_GPIO_NUM;
  config.pin_d3 = Y5_GPIO_NUM;
  config.pin_d4 = Y6_GPIO_NUM;
  config.pin_d5 = Y7_GPIO_NUM;
  config.pin_d6 = Y8_GPIO_NUM;
  config.pin_d7 = Y9_GPIO_NUM;
  config.pin_xclk = XCLK_GPIO_NUM;
```

```
  config.pin_pclk = PCLK_GPIO_NUM;
  config.pin_vsync = VSYNC_GPIO_NUM;
  config.pin_href = HREF_GPIO_NUM;
  config.pin_sscb_sda = SIOD_GPIO_NUM;
  config.pin_sscb_scl = SIOC_GPIO_NUM;
  config.pin_pwdn = PWDN_GPIO_NUM;
  config.pin_reset = RESET_GPIO_NUM;
  config.xclk_freq_hz = 20000000;
  config.pixel_format = PIXFORMAT_JPEG;

  //init with high specs to pre-allocate larger buffers if(psramFound()){
    config.frame_size = FRAMESIZE_UXGA;
    config.jpeg_quality = 10; //0-63 lower number means higher quality
    config.fb_count = 2;
  } else {
    config.frame_size = FRAMESIZE_SVGA;
    config.jpeg_quality = 12; //0-63 lower number means higher quality
    config.fb_count = 1;
  }

  // camera init
  esp_err_t err = esp_camera_init(&config);
  if (err != ESP_OK) {
    Serial.printf("Camera init failed with error 0x%x",
    err); delay(1000); ESP.restart();
  }

  // Drop down frame size for higher initial frame rate
  sensor_t * s = esp_camera_sensor_get();
  s->set_framesize(s, FRAMESIZE_CIF); //
UXGA|SXGA|XGA|SVGA|VGA|CIF|QVGA|HQVGA|QQVGA
}

void loop(){

  if (sendPhoto){
    Serial.println("Preparing photo"); digitalWrite(FLASH_LED,
    HIGH);
    delay(200); sendPhotoTelegram();
    digitalWrite(FLASH_LED,
    LOW);
    sendPhoto = false;
  }
  if(digitalRead(BUTTON) == LOW){
    Serial.println("Preparing photo");
    digitalWrite(FLASH_LED, HIGH);
    delay(200); sendPhotoTelegram();
    digitalWrite(FLASH_LED,
    LOW);
```

```
  sendPhoto = false;
}

if (millis() - bot_lasttime > BOT_MTBS)
{ int numNewMessages = bot.getUpdates(bot.last_message_received +
  1);

  while (numNewMessages)
  {
    Serial.println("got response");
    handleNewMessages(numNewMessages); numNewMessages =
    bot.getUpdates(bot.last_message_received + 1);
  }
  bot_lasttime = millis();
```

Appendix 2: Simulation Images