A Minor Project report on

# Real-Time Vehicle Security with face recognition and E-mail alerts

Submitted for the partial fulfilment of the academic requirements for the Award of the Degree of

**BACHELOR OF TECHNOLOGY**

**IN**

**ELECTRONICS AND COMMUNICATION ENGINEERING**

Submitted By

**K. LAKSHMI NITYA 21911A04G9**

|  |  |
| --- | --- |
| **M. SRIVANI** | **21911A04H8** |
| **J. AKHILA** | **21911A04G1** |
| **K. GAYATHRI** | **21911A04G6** |
| **M. ANIL** | **21911A04J1** |

## Under the Esteemed Guidance of Dr. A. Laxman, M.Tech., Ph.D., Assistant Professor



## Department of Electronics and Communication Engineering VIDYA JYOTHI INSTITUTE OF TECHNOLOGY (An Autonomous Institute)

**Accredited by NAAC & NBA, Approved by A.I.C.T.E., New Delhi, Permanently Affiliated to JNTU, Hyderabad)(Aziz Nagar, C.B. Post, Hyderabad -500075) 2021-2025**



**(Accredited by NAAC, Approved by AICTE New Delhi & Permanently Affiliated to JNTUH)**

### Aziz Nagar Gate, C.B. Post, Hyderabad-500 075

## Department of Electronics and Communication Engineering

**(Accredited by NBA)**

# CERTIFICATE

This is to certify that the project report titled **“REAL TIME VEHICLE SECURITY WITH FACE RECOGNITION AND E-MAIL ALERTS**” is being submitted by **K. LAKSHMI NITYA (21911A04G9), M. SRIVANI (21911A04H8), J. AKHILA (21911A04G1), K. GAYATHRI (21911A04G6), M. ANIL (21911A04J1),** of IV B. Tech I Semester of ***Electronics & Communication Engineering*** is a record Bonafide work carried out by them. The results embodied in this report have not been submitted to any other University for the award of any degree.

## INTERNAL GUIDE HEAD OF THE DEPARTMENT

Dr. A. Laxman, M.Tech., Ph.D., Dr. M. Rajendra Prasad M.E., Ph.D.,

Assistant Professor Professor & HOD

Department of ECE Department of ECE

**EXTERNAL EXAMINER**

## DECLARATION

This is to certify that the work reported in present project entitled “**Real Time Vehicle Security with Face Recognition and E-Mail Alerts”** is a record of work done by us in the Department of Electronics and Communication Engineering, Vidya Jyothi Institute of Technology, Jawaharlal Nehru Technological University, Hyderabad. The reports are based on the project work done entirely by us and not copied from any other source.

**PROJECT ASSOCIATES**

**K. LAKSHMI NITYA 21911A04G9**

**M. SRIVANI 21911A04H8**

1. **AKHILA 21911A04G1**
2. **GAYATHRI 21911A04G6**

**M. ANIL 21911A04J1**

## ACKNOWLEDGEMENT

We would like to express our sincere gratitude **to Dr. A. Laxman, M.Tech., Ph.D.,** Project guide who has guided and supported us through every stage in the project**.**

We are really grateful to **Dr. M. RAJENDRA PRASAD M.E., Ph.D.,** Professor & HOD ECE Dept, Vidya Jyothi Institute of Technology for his time to time, much needed valuable guidance throughout my study.

We express our hearted gratitude to is being submitted by **Dr. A. PADMAJA**

**M.Tech., PhD., Dean (Accreditation and ranking)**, Vidya Jyothi Institute of Technology for giving us spontaneous encouragement for completing theproject.

We thank **Dr. E. SAIBABA REDDY, M.E., Ph.D.,** Principal, Vidya Jyothi Institute of Technology for encouraging us in the completion of our project**.**

It is our privilege to express our gratitude and indebtedness to **Dr. PALLA RAJESHWAR REDDY M.Sc., Ph.D.,** Secretary & Correspondent of Vidya Jyothi Institute of Technology for his moral support**.**

We express our heartful thanks to **Staff of Electronics and Communication Department**, Vidya Jyothi Institute of Technology for helping us in carrying out our project successfully**.**

K. LAKSHMI NITYA **21911A04G9**

M. SRIVANI  **21911A04H8**

1. AKHILA  **21911A04G1**
2. GAYATHRI  **21911A04G6**

M. ANIL  **21911A04J1**

## ABSTRACT

Now-a-days vehicle thefts are getting increased day-by-day. So, to reduce these kinds of thefts we have come up with a real time application. The proposed system provides security and better theft control by using facial recognition and giving shock treatment when the unauthorised person tries to start the ignition. It will detect the face and sends the image of the person to the authorised user mail-id and asks for the consent of the user to access or not if the user doesn’t give access, then a shock will be given to the unauthorised person and the engine doesn’t start. The system uses Arduino UNO along with a pie cam and a wi-fi controller installed in the vehicle the implemented system is very simple with greater security for vehicle anti-theft protection and low cost compared to others

The system provides an efficient, user-friendly solution for monitoring and controlling unauthorized access to vehicles. It employs a camera module to capture images of individuals attempting to access the vehicle, with face recognition software verifying the identity of the user against a pre-stored database.

When an unauthorized attempt is detected, the system instantly triggers an alert mechanism. An email notification containing the intruder's image and the timestamp is sent to the registered vehicle owner's email address via an IoT module such as ESP8266 or GSM. The Arduino Uno acts as the central controller, managing peripheral sensors (e.g., motion or door sensors) and facilitating communication between the hardware components.

This low-cost and scalable solution combines the computational power of external face recognition systems (like Raspberry Pi or cloud-based APIs) with the simplicity and reliability of Arduino. Applications include personal vehicle security, fleet management, and rental services, offering peace of mind and enhanced protection against theft or unauthorized use.

### TABLE OF CONTENTS

|  |  |  |
| --- | --- | --- |
| **CHAPTER** | **TITLE** | **PAGE NO** |
| **1** | **INTRODUCTION** | 1 |
| **2** | **RELATED WORK** | 2 |
|  | 2.1 Access towebcam | 2 |
|  | 2.2 Face identification | 3 |
|  | 2.3 Data collection | 4 |
|  | 2.4 Training | 5 |
|  | 2.5 Face Recognition | 6 |
|  | 2.6 Programming Arduino | 6 |
| **3** | **REAL TIME VEHICLE SECURITY** | 7 |
| **4** | **TECHNICAL SPECIFICATIONS** | 8 |
|  | 4.1 Hardware Requirements | 8 |
|  | 4 .2 Software Requirements | 9 |
| **5** | **BLOCK DIAGRAM** | 10 |
| **6** | **POWER SUPPLY** | 11 |
| **7** | **METHODOLOGY** | 12 |
| **8** | **EXISTING METHOD** | 13 |
|  | 8.1 Disadvantages of existing method | 14 |
|  | 8.2 methodology flow chart | 15 |
|  | 8.3 Hardware Design | 16 |
| **9** | **CONSTRUCTION AND WORKING** | 17 |
|  | 9.1 Arduino Uno | 18 |
|  | 9.2 Shock Sensor | 19 |
|  | 9.3 Relay Module | 20 |
| **10** | **HISTORY OF ARDUINO** | 21 |
| **11** | **SHOCK SENSOR** | 25 |
| **12** | **RELAY MODULE** | 28 |
| **13** | **CODE IN THIS PROJECT** | 30 |
|  | 13.1 Environmental Advantages | 36 |
|  | 13.2 Disadvantages | 37 |
| **14** | **FUTURE SCOPE AND CONCLUSION** | 38 |
| **15** | **APPLICATIONS** | 39 |
|  | 15.1 Vehicle Access Control | 39 |
|  | 15.2 Theft Prevention | 39 |
|  | 15.3 Safety Features | 39 |
| **16** | **ADDITIONAL FEATURES** | 40 |
|  | 16.1 Benefits | 40 |
|  | 16.2 Results | 41 |
|  | 16.3 Conclusion | 41 |
| **17** | **REFERENCES** | 41 |

## CHAPTER 1

### INTRODUCTION

With the increasing number of vehicle thefts and unauthorized access, there is a growing need for more sophisticated security solutions. Traditional vehicle security systems often rely on alarm systems and GPS tracking, which, while effective to some extent, are limited in their capabilities to prevent unauthorized entry in real time. Recent advances in embedded systems and artificial intelligence, however, offer a promising solution to these limitations. Real-time vehicle security systems utilizing face recognition and email alerting capabilities represent a new frontier in vehicle protection, leveraging the power of microcontroller technology such as Arduino, combined with IoT and facial recognition techniques.

Vehicle security is a critical concern in modern times, and face recognition technology offers an advanced, reliable solution to enhance safety measures. By integrating face recognition systems into vehicles, unauthorized access can be effectively prevented. This technology uses cameras and sophisticated algorithms to authenticate the identity of the driver by analyzing their facial features, ensuring that only authorized individuals can operate the vehicle. Unlike traditional security methods such as keys or passwords, face recognition is harder to replicate or bypass, providing an additional layer of security. Furthermore, the system can log entries and exits, alert owners of suspicious attempts, and even link with connected devices to notify authorities in case of theft. With its ability to offer seamless, touchless authentication, face recognition is paving the way for safer and smarter vehicle access systems in the era of intelligent transportation.

## CHAPTER 2 Related work on Face Recognition module and vehicle security

**2.1 Face Recognition Module:**

A face recognition module is a compact system that uses advanced algorithms to identify and verify individuals based on their facial features. Equipped with a camera and a microcontroller or processor, it captures and analyzes facial data, matching it with stored templates for authentication. These modules are widely used in security systems, access control, and smart devices due to their high accuracy and touchless operation. Their integration into applications like vehicle security, attendance systems, and surveillance enhances safety and convenience, making them a key component in modern technology solutions.

The steps involved in the face recognition module are as follows: Step 1: Access to webcam step 2: Face identification.

step 3: Data collection Step 4: Training step 5: Face recognition step 6: Programming Arduino

## Step 1: Access to Webcam

The 1st step for facial recognition was to have access to a camera or a computer vision. Since India is under lockdown the cheapest solution which I found was to use my computers webcam to which I had access with a python program using OpenCV module.

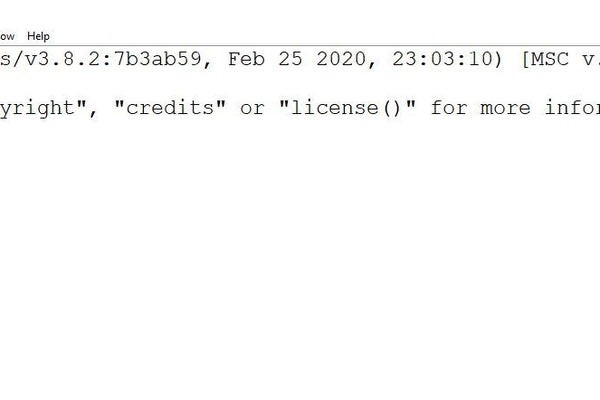


Fig 2.1 IDLE python to access OpenCV module

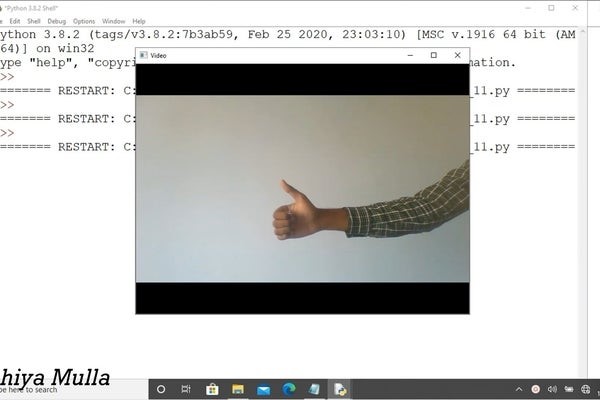


Fig:2.2 Web Cam access

OpenCV (Open-Source Computer Vision Library) is an open-source computer vision and machine learning software library. OpenCV was built to provide a common infrastructure for computer vision applications and to accelerate the use of machine perception in commercial products.

If OpenCV is installed on your computer then you are good to go. If not then follow this step. open command prompt and type "pip install OpenCV"..

Once OpenCV is installed we are good to go... To check if its properly installed open your Python interpreter and import the library. See the image above that should be your output.

Download the python file "AccessTo\_webcam.py" and run it. I have provided all the necessary comments there.

There you go, now you have access to the webcam. Well done. let us proceed to step 2.

**Step 2: Face Identification.**

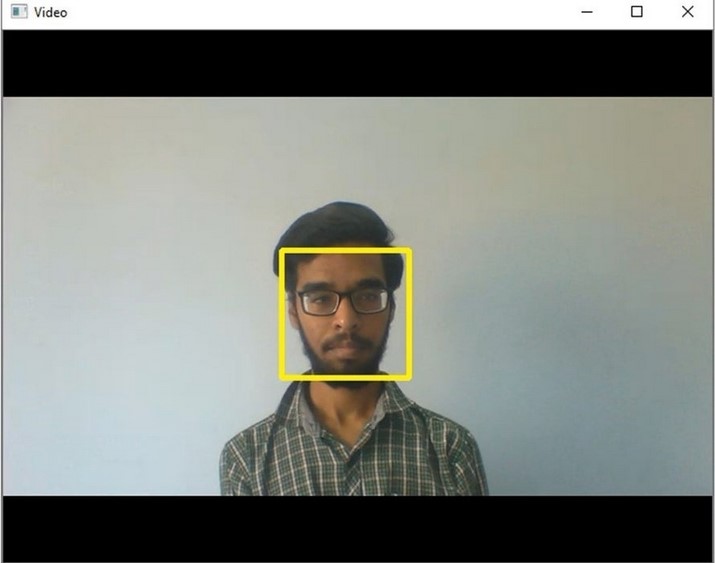


Fig 2.3 Face Identification

As shown in Fig 2.3 with the help of the same OpenCV module, we have to identify whether there is a face on the video stream or not.

OpenCV provides a training method or pre-trained models called as Cascade Classifier. The pre-trained models are located in the data folder in the OpenCV installation. I'm providing that file just download it and place it in your project folder. The folder where the "AccessTo\_webcam.py" file is stored. If you have not created one then do it.

Download "haarcascade\_frontalface\_default" and place it in the main project folder.

Download "Face\_identification.py" and place it in the main project folder. All the explanation is provided in it.

Now you can identify the faces in a video stream. So, let's proceed to step 3.

Attachments

## Step 3: Data Collection



Fig 2.3 Collection of data

To recognize the faces, we need to train our python program. For which we need some data as shown in fig 2.3.

Data collection is rather the easiest step in this project. create a folder named "image data" in your main project folder. Inside the "image data" folder create some additional folders with the person's name, where we will store the data. for example:

In the "image data" folder I have created two more folders named "HRK" and "Yahiya". as shown in the above image.

Now go ahead create your own folders and name them.

Once the folders are created then start collecting images of that specific person. I recommend collecting nearly about 20 images per person. You can also add more images but see to it that data collected for all the persons contains the same number of images. It helps to provide accuracy.

that's it now let's move on to step 4.

## Step 4: Training

In brief, we will go through all the folders and images which are present in the "image data" folder and create a dictionary that will contain the label ID and the corresponding name. Simultaneously we will load the image to detect the face in each and every image which we call it "Region of Interest" and create a “. yml" file which contains that information.

Assuming that you have data collected for person X and Y.

we will label person X as 1 which will be his label ID and name will be X itself. We load the image to find his face i.e Region of interest and append the data to a list. similar steps will be followed for person Y. And finally, we will create a “. yml" file.

Download the "face\_trainer.py" file and place it in the main project folder. All the necessary explanation is provided in that file itself.

When you run this program, it will go through all the images and create two files named "labels. Pickle" and "trainer. Yml". Now you have trained your own model. so let's proceed to step 5.

## Step 5: Face Recognition

If you have gone through all the steps properly then you may have created your own trained data. Now we will use that data for face recognition.

Basically, we will load our trained models into the python file, access our webcam, and identify Faces in the video stream and do a comparison or prediction between the current face which is identified in the video stream, and the model which was trained. if the data is matched then we say that the person is recognized it is just that simple...

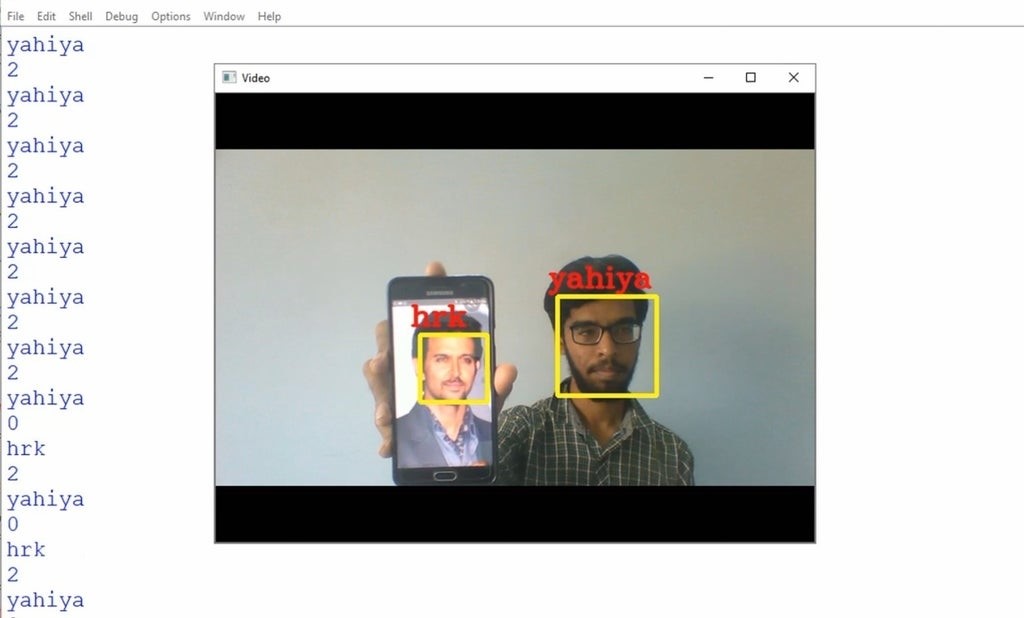


Fig 2.4 Recognition of face

Download "face\_recognise.py" and run it. All the necessary information is provided in it. Now your face may have been recognized. if the accuracy is not good then try updating the data. if you are all good to go then let’s proceed to step 6

### Step 6: Programming Arduino

The last and the final step is programming Arduino, and to provide a mode of communication between python and Arduino. For communication, I used "Serial Communication". Go through the video which I have linked above to find how Serial Communication works and to establish one. You will find all the required files in the video description.

If you have gone through the video then let me explain to you what I did. When my face is recognized then the label ID provided is 2. Once label ID is 2 I will send '1' as the serial data to my Arduino. Which will turn on my LED chaser circuit. If label ID is other than 2 then i will send '0' as the serial data, which will turn off my LED chaser Circuit.

Download the "ard\_chaser.ino" file. It is a simple LED chaser program that uses serial communication.

Similarly download "face\_recogniser1.py" that will establish the serial communication between Arduino and the python program.

**2.2 Real time vehicle Security:**

A Car theft detection system is a comprehensive security solution designed to prevent unauthorized access and theft of vehicles by combining hardware, software, and connectivity. These systems typically include various sensors, such as motion, tilt, door, and glass break detectors, which monitor for unusual activity around the vehicle. If an unauthorized attempt is detected, alarms are triggered to deter thieves and alert nearby individuals. Many systems also feature GPS tracking, allowing vehicle owners or authorities to monitor the car’s realtime location and set up geo-fences to receive alerts if the car leaves a designated area.

Advanced systems integrate biometric authentication, such as fingerprint or facial recognition, ensuring only authorized users can start or access the vehicle. Engine immobilizers add another layer of protection by preventing the car from starting without proper authentication. Smart connectivity through mobile apps enables remote control of vehicle functions like locking/unlocking doors, disabling the engine, or receiving instant alerts about suspicious activities.

CCTV or camera integration provides visual evidence of unauthorized access attempts, while artificial intelligence (AI) and machine learning analyse patterns to detect anomalies, such as unusual driving behaviours or repeated tampering. These features make car theft detection systems highly effective for personal vehicles, fleet management, and public transportation security.

The benefits of these systems include real-time alerts, effective deterrence, and assistance in recovering stolen vehicles. However, challenges like false alarms, installation costs, and ensuring privacy compliance need to be addressed for optimal performance. By leveraging modern technology, car theft detection systems provide peace of mind and a robust safeguard against vehicle theft.

**2.3 Technical Specifications:**

For integrating the circuit, we have some hardware and software requirements they are mentioned below:

## Hardware Requirements

* Camera
* Arduino
* Relay
* LCD
* LED

##  Shock Sensor  Motor Software Requirements

* Python IDLE
* Arduino IDE  Embedded C

* 1. **CIRCUIT IMPLEMENTATION:**

This image showcases a prototype circuit setup involving an Arduino microcontroller, a breadboard, and several connected components, including a servo motor, relay module, and sensors. The breadboard is used for temporary connections, enabling the testing of the circuit without soldering. The servo motor is likely being used for motion control, while the relay module allows the Arduino to manage higher-power components. This setup appears to be part of a project, potentially for an automation or security system, demonstrating the integration of various electronic modules for functionality testing**.**

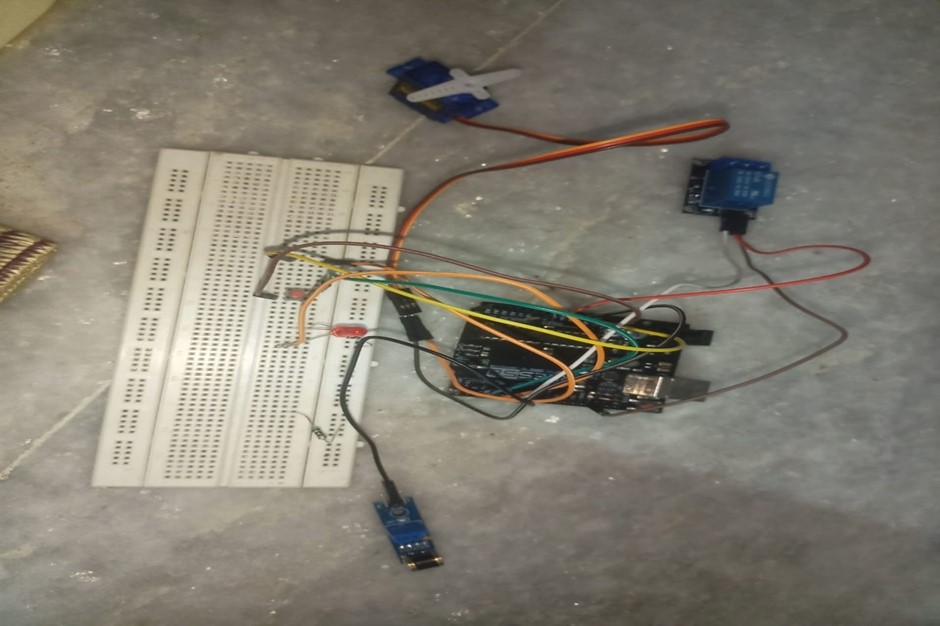


Fig 2.5: Circuit Implementation

* 1. **Block Diagram:**

This block diagram represents a system integrating various components for vehicle security. The central unit is an Arduino microcontroller powered by a power supply. It interfaces with a camera and laptop for face recognition, serving as an access control mechanism. Relays control both a shock mechanism and a motor, ensuring safety and operational functionality. An LCD provides a user interface for displaying messages, while a buzzer serves as an alert mechanism for unauthorized access or system notifications. This configuration combines hardware and software for efficient security management.

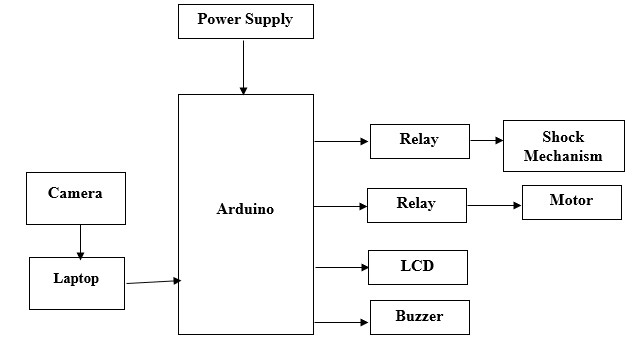


Fig 2.6 Block diagram

## 2.6 POWER SUPPLY

We require a power supply to provide power to the components i.e., a reliable power source to ensure continuity in system This block diagram illustrates a system centred around an Arduino UNO, powered by a dedicated power supply. The Arduino interfaces with multiple components for functionality. Inputs include a GPS module for location tracking and a GSM module for communication, while outputs are connected to a camera, LCD for display, and storage for data logging. Additionally, a relay controls higher-power devices, a motor provides mechanical action, and a buzzer acts as an alert system. This setup integrates various hardware elements, making it suitable for applications like vehicle tracking, security systems, or automation projects.

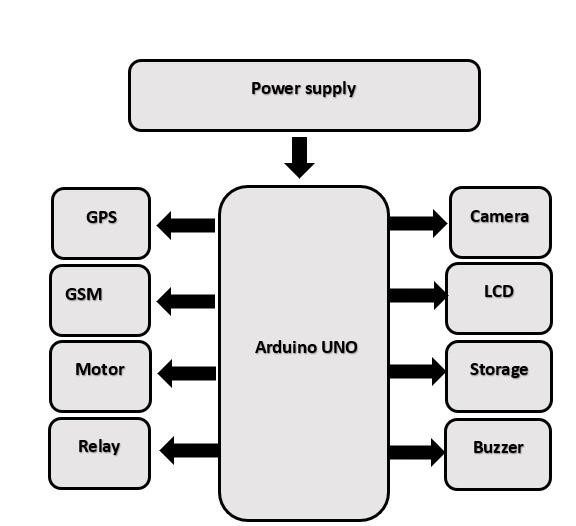


Fig 2.7 power supply diagram

## CHAPTER 3 METHODOLOGY

This project suggests a sophisticated facial recognition authentication mechanism. Robust and effective apps utilize a real-time verification system. A sophisticated algorithm is used for authentication in face detection and recognition systems. The image of the occupant of the vehicle, positioned correctly in front of the driver's seat, is captured by the camera. After obtaining the person's photograph, the algorithm attempts to identify the face. It gathers a diverse dataset of facial images for training the recognition model. Includes various lighting conditions, angles, and facial expressions. The normalize and resize images to ensure consistency for betterment Apply face detection to locate and extract faces from images. Implement liveness detection mechanisms to differentiate between live faces and still images, preventing spoofing attacks. Techniques may include analysing facial movements, checking for blinking, or employing 3D face reconstruction. Develop an intuitive and user-friendly interface for system configuration and monitoring, providing feedback to users during the recognition process. Implementing a secure and efficient database results in storing facial templates of authorized users in any amount of number Regularly updating the database allows us to include new users or remove revoked access. Implementation of a backup mechanism, alternative methods like lock so that there occurs no problems or delay in case of face recognition failure. Provide detailed documentation for system setup, maintenance, and troubleshooting. Provide detailed documentation for system setup, maintenance. Remember to tailor these processes to the specific needs and limits of the car security system you're creating... Implement liveness detection mechanisms to differentiate between live faces and still images, preventing spoofing attacks. Techniques may include analysing facial movements, checking for blinking, or employing 3D face reconstruction**.**

## 3.1 Existing Method

Car alarm techniques are used to prevent the car theft with the help of different type of sensors like pressure, tilt and shock & door sensors. These systems however bear some limitations such as high cost, high false alarm rate, and easy to be disabled. In order to solve these problem recent advancements in computer hardware and software have enabled automobile industry to develop affordable automated biometrics-based identification and verification systems. Many biometrics, including face detection, facial features, hand geometry, handwriting and voice have been used for the identification and verification of individuals. But biometric has its own disadvantages such as the systems are not 100% accurate, they require integration and/or additional hardware and cannot be reset once compromised, you can always change your password if somebody learns it, but there’s no way to modify your iris, retina or fingerprint. Once somebody has a working copy of these, there’s not much you can do to stay safe, other than switching to passwords or using another finger.

## Disadvantages Of Existing Method

* Due to longer distance (range), siren cannot be heard
* Most of the cars have similar sounds, and physically
* Alarms can be disabled on theft attempts
* Alarm sound can be mitigated in crowded areas.
* Cost is high

## 3.2. Methodology Flowchart

The system captures live video from the webcam and loads sample images of authorized individuals. It then encodes the faces in the sample images for comparison with detected faces in the live video stream. Detected faces are compared with the known face encodings to determine if they match any authorized individuals. If a match is found, the individual’s name is displayed. If no match is found, an alert email is sent and printed in the console. Additionally, the system sends signals to an Arduino board to control security mechanisms such as engine activation or immobilization. The code continuously loops to capture video frames, detect faces, perform recognition, and take appropriate actions based on the recognition results. Pressing ’q’ on the keyboard terminates the program. This methodology ensures vehicle security by actively monitoring and identifying individuals attempting to access the vehicle, thereby preventing unauthorized entry or theft.

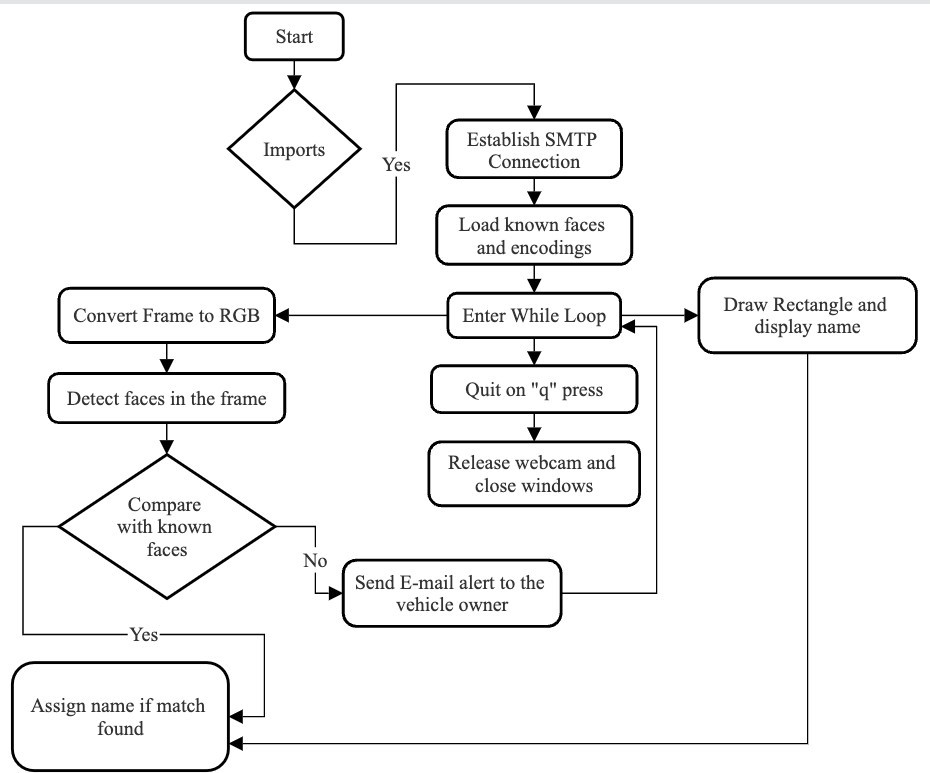


Fig 3.1 flowchart of methodology

## 3.3. Hardware Design

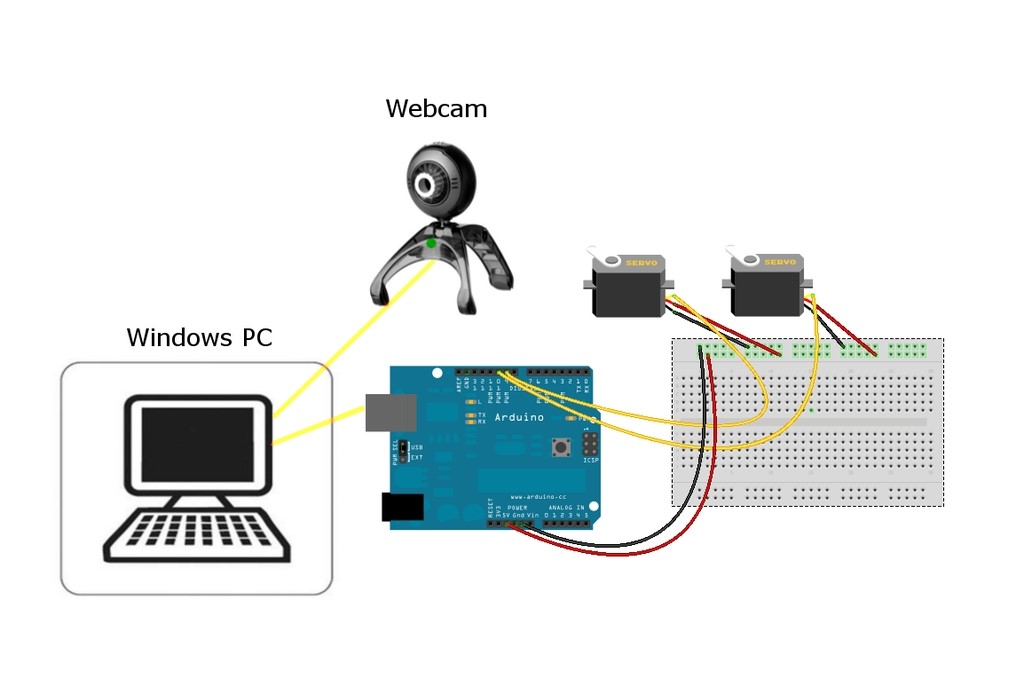


Fig 3.2 design of hardware

This system employs advanced face recognition technology to enhance security and automate access control. When a user approaches, the face recognition module scans and analyses the facial features. If the detected face matches the authorized user database, the system activates a servo motor to grant access or perform a designated action. However, if the face is not recognized as authorized, the system triggers an alert mechanism. In this case, an LED begins blinking to indicate a security breach, and the shock sensor module actively monitors for any unusual physical tampering or impact on the system. This dual-layered approach ensures that access is granted only to authorized individuals while providing immediate alerts in the event of unauthorized attempts.

## CHAPTER 4 CONSTRUCTION AND WORKING MAIN COMPONENTS

1. ARDUINO UNO
2. SHOCK SENSOR

c)RELAY MODULE

## INTRODUCTION OF ARDUINO ASSEMBLY

WHAT IS ARDUINO

Arduino is an open-source hardware and software company, project and user community that designs and manufactures single-board microcontrollers and microcontroller kits for building digital devices and interactive objects that can sense and control both physically and digitally. Its products are licensed under the GNU Lesser General Public License (LGPL) or the GNU General Public License (GPL), permitting the manufacture of Arduino boards and software distribution by anyone. Arduino boards are available commercially pre-assembled form or as do-it-yourself (DIY) kits . Arduino board designs use a variety of microprocessors and controllers. The boards are equipped with sets of digital and Analog input/output (I/O) pins that may be interfaced to various expansion boards or breadboards (shields) and other circuits. The boards feature serial communications interfaces, including Universal Serial Bus (USB) on some models, which are also used for loading programs from personal computers. The microcontrollers are typically programmed using a dialect of features from the programming languages C and C++. In addition to using traditional compiler tool chains, the Arduino project provides an integrated development environment (IDE) based onthe Processing language project. The Arduino project started in 2003 as a program for students at the Interaction Design Institute Ivrea in Ivrea, Italy, aiming to provide a low-cost and easy way for novices and professionals to create devices that interact with their environment using sensors and actuators. Common examples of such devices intended for beginner hobbyists include simple robots, thermostats and motion detectors. The name Arduino comes from a bar in Ivrea, Italy, where some of the founders of the project us to meet. The bar was named after Arduino of Ivrea, who was the margrave of the March of Ivrea and King of Italy from 1002 to 1014.

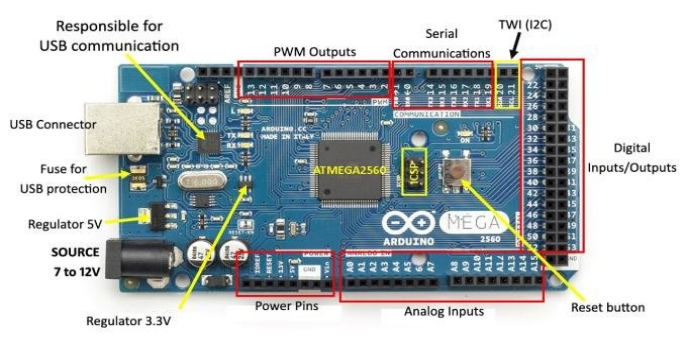


Fig 4.1 Arduino Mega 2560

## HISTORY OF ARDUINO

The Arduino project was started at the Interaction Design Institute Ivrea (IDII) in Ivrea, Italy. At that time, the students used a BASIC Stamp microcontroller at a cost of about 3780 INR, a considerable expense for many students. In 2003 Hernando Barragan created the development platform Wiring as a Master's thesis project at IDII, under the supervision of Massimo Banzi and Casey Reas Case is known for co-creating, with Ben Fry, the Processing development platform. The project goal was to create simple, low-cost tools for creating digital projects by non-engineers The Wiring platform consisted of a printed circuit board (PCB) with an ATMEL328P microcontroller an IDE based on Processing and library functions to easily program the microcontroller .In 2003, Massimo Banzi, with David Mellis , another IDII student, and David Cuartielles, added support for the cheaper ATmega8 microcontroller to Wiring. But instead of continuing the work on Wiring, they forked the project and renamed it Arduino.

The initial Arduino core team consisted of Massimo Banzi, David Cuartielles, Tom Igoe, Gianluca Martino, and David Mellis, but Barragan was not invited to participate. Following the completion of the Wiring platform, lighter and less expensive versions were distributed in the open-source community. It was estimated in mid-2011 that over 300,000 official Arduinos had been commercially produced, and in 2013 that 700,000 official boards were in users' hands In October 2016, Federico Musto, Arduino's former CEO, secured a 50% ownership of the company In April 2017, Wired reported that Musto had "fabricated his academic record.... On his company's website, personal LinkedIn accounts, and even on Italian business documents, Musto was until recently listed as holding a PhD from the Massachusetts Institute of Technology. In some cases, his biography also claimed an MBA from New York University." Wired reported that neither University had any record of Musto's attendance, and Musto later admitted in an interview with Wired that he had never earned those degrees.

The controversy surrounding Musto continued when, in July 2017, he reportedly pulled many Open-source licenses, schematics, and code from the Arduino website, prompting scrutiny and outcry In October 2017, Arduino announced its partnership with ARMHoldings (ARM). The announcement said, in part, "ARM recognized independence as a core value of Arduino ... without any lock vendors and architectures. In early 2008, the five co-founders of the Arduino project created a company, Arduino LLC, to hold the trademarks associated with Arduino. The manufacture and sale of the boards was to be done by external companies, and Arduino LLC would get a royalty from them. The founding bylaws of Arduino LLC specified that each of the five founders transfer ownership of the Arduino brand to the newly formed company. At the end of 2008, Gianluca Martino's company Smart Projects, registered the Arduino trademark in Italy and kept this a secret from the other cofounders for about two years. This was revealed when the Arduino company tried to register the trademark in other areas of the world (they originally registered only in the US), and discovered that it was already registered in Italy. Negotiations with Gianluca and his firm to bring the trademark under control of the original Arduino company failed. In 2014, Smart Projects began refusing to pay royalties. They then appointed a new CEO, Federico Musto, who renamed the company Arduino SRL and created the website arduino.org, copying the graphics and layout of the original arduino.cc. This resulted in a rift in the Arduino development team In January 2015, Arduino LLC filed a lawsuit against Arduino SRL In May 2015, Arduino LLC created the worldwide trademark Genuino used as brand name outside the United States At the World Maker Faire in New York on October 1, 2016, Arduino LLC co-founder and CEO Massimo

Banzi and Arduino SRL CEO Federico Musto announced the merger of the two companies By 2017 Arduino AG owned many Arduino trademarks. In July 2017 BCMI, founded by Massimo Banzi David Cuartielles David Mellis and Tom Igoe acquired Arduino AG and all the Arduino trademarks. Fabio Voilante is the new CEO replacing Federico Musto, who no longer works for Arduino AG.

### SOFTWARE

A program for Arduino hardware may be written in Embedded C and C++ with compilers that produce binary machine code for the target processor. Atmel provides a development environment for their 8-bit AVR and 32-bit ARM Cortex-M based microcontrollers: AVR

Studio(older) and Atmel Studio (newer )

### a) IDE

The Arduino integrated development environment (IDE) is a cross-platform application (for Windows, macOS, Linux) that is written in the programming language Java. It originated from the IDE for the languages Processing and Wiring. It includes a code editor with features such as text cutting and pasting, searching and replacing text, automatic indenting, brace matching, and syntax highlighting, and provides simple one-click mechanisms to compile and upload programs to an Arduino board. It also contains a message area, a text console, a toolbar with buttons for common functions and a hierarchy of operation menus. The source code for the IDE is released under the GNU General Public License, version 2.The Arduino IDE supports the languages C and C++ using special rules of code structuring The Arduino IDE supplies a software library from the Wiring project, which provides many common input and output procedures. User-written code only requires two basic functions, for starting the sketch and the main program loop, that are compiled and linked with a program stub main () into an executable cyclic executive program with the GNU toolchain, also included with the IDE distribution The Arduino IDE employs the program avrdude to convert the executable code into a text file in hexadecimal encoding that is loaded into the Arduino board by a loader program in the board's firmware. **b) Sketch**

A sketch is a program written with the Arduino IDE. Sketches are saved on the development computer as text files with the file extension .into Arduino Software (IDE) pre-1.0 saved sketches with the extension .pde. A minimal Arduino C/C++ program consist of only two functions :

Setup ():

This function is called once when a sketch starts after power-up or reset. It is used to initialize variables, input and output pin modes, and other libraries needed in the sketch.

Loop ():

After setup () function exits (ends), the loop () function is executed repeatedly in the main program. It controls the board until the board is powered off or is reset

### SHOCK SENSOR

What is a Shock Sensor?

A shock sensor is a device that detects and measures impacts, shocks, or vibrations. These sensors convert mechanical energy into electrical signals, which can be used to trigger alarms, record data, or monitor equipment health.



Fig 4.2 Shock Sensor module Types of Shock Sensors:

1. Accelerometer-based sensors
2. Piezoelectric sensors
3. Capacitive sensors
4. Inductive sensors
5. Optical sensors
6. Strain gauge sensors
7. MEMS (Micro-Electro-Mechanical Systems) sensors

How Shock Sensors Work:

1. Accelerometer-based sensors measure acceleration changes.
2. Piezoelectric sensors convert mechanical stress into electrical signals.
3. Capacitive sensors detect changes in capacitance due to impact.
4. Inductive sensors measure changes in inductance.
5. Optical sensors detect changes in light reflection or transmission.

Applications:

1. Industrial monitoring (machinery, equipment)
2. Transportation (vehicle impact detection)
3. Sports (helmets, protective gear)
4. Consumer electronics (drop detection)
5. Aerospace and defense (impact detection)
6. Medical devices (patient monitoring)
7. Package tracking and monitoring
8. Vibration analysis
9. Structural health monitoring
10. Impact testing

Key Specifications:

1. Sensitivity
2. Frequency range
3. Resolution
4. Accuracy
5. Operating temperature
6. Durability
7. Power consumption
8. Response time
9. Measurement range
10. Interface (e.g., analog, digital, wireless)

Advantages:

1. Real-time impact detection
2. Enhanced safety
3. Reduced equipment damage
4. Improved product design
5. Increased efficiency
6. Cost-effective
7. Compact size
8. Low power consumption

Factors to Consider When Selecting Shock Sensors:

1. Application requirements
2. Sensor type and technology
3. Sensitivity and accuracy
4. Environmental conditions
5. Power consumption
6. Integration complexity
7. Cost
8. Size and weight
9. Durability
10. Certification and compliance

### RELAY MODULE

A Relay module is an essential electronic tool used in various devices to act as a switch between low-powered digital electronics and high-powered devices. It allows digital circuits and microcontrollers like Arduino to control motors or lighting circuits.



Fig 4.3 Relay Module

Act as a switch that opens or closes electrical circuits when activated by a signal.

It consists of two internal metal contacts that do not connect with each other. However, an internal switch connects these contacts to complete an electrical circuit, allowing current flow. To activate the relay, a voltage power current is applied to an electromagnetic coil. This pulls the metal contacts together and allows current to flow on the other side of the relay.

Can be used in various applications, such as mains switching, automating electrical appliances and lights, isolated power delivery, and high current switching. They are often integrated with microcontrollers like Arduino,ESP32 and Raspberry Pi, enabling projects like motion sensor lamps, smartwatch car remotes, touchless doorbells, and more. It comes in different types, including electromechanical relays and solid-state relays. Relay modules can be different types and price range, and have different channel types, such as single-channel, dual-channel, four-channel, or eight-channel relays.

Each channel of the relay can control a different device independently. To control 4 devicess 4-channel relay will be needed.

Acts as a switch between low-powered digital electronics and high-powered devices.It allows the control of motors, lighting circuits, and other high-powered devices using microcontrollers.

They are used in various applications, such as mains switching, automation, and high current switching.

**How Does a Relay Module Work?**

Consists of two internal metal contacts that are not connected to each other, but an internal switch connects these contacts to complete an electrical circuit, allowing current flow. This functionality makes relays an essential component in enabling the control of high-powered devices using low-powered digital electronics.

When a voltage is applied to the electromagnetic coil of the relay, it creates an electromagnetic field that pulls the metal contacts together. This action completes the circuit, allowing current to flow on the other side of the relay. The relay module essentially acts as a switch, controlled by the digital circuit or microcontroller.

The ability to control higher power loads through the module is particularly useful in various applications. For example, can be used for mains switching, automating electrical appliances and lights, isolated power delivery, and high current switching. By integrating them with microcontrollers like Arduino or Raspberry Pi, innovative projects such as motion sensor lamps, smartwatch car remotes, and touchless doorbells can be created.

They come in different types, including electromechanical relays and solid-state relays. Additionally, they can vary in terms of the number of channels they offer, such as singlechannel, dual-channel, four-channel, or eight-channel relays.

### 4.1 ADVANTAGES

Advantages of Vehicle Security using Face Recognition: **Security Advantages:**

1. Enhanced Biometric Authentication: Face recognition provides a secure and uniqueidentification method.
2. Reduced Theft: Prevents unauthorized access and theft.
3. Immobilizer: Disables vehicle operation for unauthorized individuals.
4. Alert System: Notifies owner of suspicious activity.

**Convenience Advantages**:

1. Keyless Entry: Eliminates need for physical keys.
2. Personalized Settings: Adjusts vehicle settings (e.g., seat, mirror positions).
3. Streamlined Rental Process: Simplifies vehicle sharing and rental.
4. Automated Driver Identification: Eliminates manual driver verification.

**Safety Advantages:**

1. Driver Fatigue Monitoring: Detects drowsiness and distraction.
2. Driver Attention Monitoring: Ensures driver focus on the road.
3. Emergency Response: Sends alerts in case of accidents.
4. Improved Road Safety: Reduces risk of accidents.**Cost-Effective Advantages**:
5. Reduced Insurance Premiums: Enhanced security reduces risk.
6. Lower Maintenance Costs: Fewer mechanical issues due to unauthorized access.
7. Increased Vehicle Value: Face recognition technology adds value.
8. Reduced Administrative Costs: Simplified driver management. **Operational Advantages**:
9. Real-Time Monitoring: Tracks vehicle usage and driver behavior.
10. Data Analytics: Provides insights on driving habits and vehicle performance.
11. Automated Reporting: Generates reports for vehicle usage and incidents.
12. Centralized Management: Monitors multiple vehicles and drivers.

**Environmental Advantages**:

1. Reduced Carbon Footprint: Encourages responsible driving habits.
2. Improved Fuel Efficiency: Optimizes vehicle performance.**Other Advantages**:
3. Enhanced User Experience: Personalized settings and convenience features.
4. Increased Driver Accountability: Promotes responsible driving.
5. Compliance with Regulations: Meets emerging regulatory requirements.
6. Scalability: Easily integrates with existing systems.

By implementing face recognition technology in vehicle security systems, manufacturers and owners can enjoy numerous benefits that enhance security, convenience, safety, and costeffectiveness.

### 4.2 DISADVANTAGES

Disadvantages of Vehicle Security using Face Recognition: **Technical Limitations**:

1. Accuracy Issues: False positives/negatives due to lighting, angles, or facial changes.
2. Sensor Malfunctions: Camera or sensor failures affect system reliability.
3. Processing Power: Requires significant computational resources.
4. Database Management: Secure storage and management of facial data.**Security Concerns:**
5. Data Breaches: Unauthorized access to facial data.
6. Cyber Attacks: Vulnerability to hacking and system compromise.
7. Spoofing Attacks: Manipulation of facial recognition using masks or images.
8. Identity Theft: Potential for stolen facial data.

**Privacy Issues**:

1. Biometric Data Collection: Raises concerns about personal data ownership.
2. Surveillance: Continuous monitoring of drivers and passengers.
3. Data Sharing: Potential for sharing facial data with third parties.
4. Lack of Anonymity: Facial recognition eliminates anonymity.**Practical Limitations:**
5. Variability in Lighting: Difficulties with facial recognition in low/high light conditions.
6. Facial Changes: Difficulty recognizing faces with glasses, masks, or makeup.
7. Multiple Drivers: System complexity with multiple authorized drivers.
8. System Updates: Requires regular updates to maintain accuracy.**Cost and Complexity**:
9. High Implementation Costs: Expensive hardware and software.
10. Maintenance Costs: Regular updates and system maintenance.
11. Complexity: Requires expertise for installation and troubleshooting.

**User Acceptance**:

1. Public Perception: Concerns about privacy and surveillance.
2. User Resistance: Resistance to biometric authentication.
3. Cultural Differences: Variability in acceptance across cultures.

**Regulatory Challenges**:

1. Compliance: Meets varying regulatory requirements globally.
2. Data Protection: Adherence to data protection laws.
3. Liability: Unclear liability in case of system failures.

**Other Disadvantages**:

1. Dependence on Technology: System failure affects vehicle operation.
2. Limited Compatibility: Potential issues with older vehicles or systems.
3. Environmental Factors: Extreme temperatures, humidity, or vibrations affect systemperformance.

While face recognition technology enhances vehicle security, it's essential to address these disadvantages to ensure a reliable, secure, and user-friendly system.

### 4.3 APPLICATIONS

Applications of Vehicle Security using Face Recognition: **Vehicle Access Control**:

1. Biometric ignition: Face recognition replaces traditional keys or keyless entry systems.
2. Driver authentication: Ensures only authorized drivers can operate the vehicle.**Theft Prevention**:
3. Face-based immobilizer: Prevents hotwiring or key duplication.
4. Alert system: Notifies owner of unauthorized access attempts.**Safety Features:**
5. Driver fatigue monitoring: Detects drowsiness or distraction.
6. Driver attention monitoring: Ensures driver focus on the road.

**Vehicle Sharing and Rental**:

1. Secure rental management: Face recognition streamlines rental processes.
2. Vehicle sharing platforms: Ensures authorized users.

**Law Enforcement and Surveillance:**

1. Automatic license plate recognition (ALPR) integration.
2. Facial recognition for suspect identification.

**Intelligent Transportation Systems (ITS):**

1. Electronic toll collection (ETC) systems.
2. Traffic monitoring and management.**Additional Features:**
3. Personalized settings: Adjusts vehicle settings (e.g., seat, mirror positions).
4. Driver behavior analysis: Provides insights on driving habits.
5. Emergency response: Sends alerts in case of accidents.
6. IoT (Internet of Things) integration.**Benefits:**
7. Enhanced security.
8. Convenience.
9. Improved safety.
10. Reduced theft.
11. Personalized experience.

**Real-World Implementations**:

1. Cadillac's Face Recognition Technology.
2. BMW's Driver Recognition System.
3. Toyota's Face Recognition-based Start System.
4. Hyundai's Biometric Authentication System.

The integration of face recognition technology in vehicle security systems offers a robust and convenient solution for ensuring vehicle safety, security, and personalized experiences.

### CHAPTER 5 RESULTS AND CONCLUSION

The future trajectory of vehicle security systems utilizing facial and voice recognition technology entails ongoing enhancements in recognition accuracy and speed, ensuring seamless authentication processes. Integration with anti-theft systems will bolster security measures, deterring theft attempts effectively. Moreover, robust data privacy protocols will safeguard user information, while exploration of cloud-based solutions will facilitate scalable data management. These advancements will collectively shape the landscape of vehicle security, fostering heightened protection and user confidence in the evolving automotive landscape.

### RESULT

Here we implemented an camera and microcomputer base processing unit for detection and recognition of authorized and unauthorized persons using Open CV and Anaconda. The face is detected by a cascade face detector in the acquired images. The result proves that the computer vision-based algorithm is reliable for protecting the vehicle from the modern improve theft techniques.

## CONCLUSION

From these we implement theft control techniques that will provide the important functions required by advanced intelligent car security, to avoid theft and protect the usage of unauthenticated users. Secured and safe environment system for automobile users and also key points for the investigators can easily determine the hijacked image

## REFERENCES

1. "Face Recognition Technology for Vehicle Security" by S. K. Singh and R.K. Sharma (2020)
2. "Real-Time Vehicle Security System Using Face Recognition and IoT" by A.K. Singh et al. (2019)
3. "Vehicle Security System with Face Recognition and Email Alerts" by M. A.

Al-Garadi et al. (2018)

1. "Face Recognition for Vehicle Access Control" by Y. Liu et al. (2017)
2. "Real-Time Face Recognition for Vehicle Security" by H. K. Lee et al.(2016)

### APPENDIX CODE USED IN THIS PROJECT REAL TIME VEHICLE SECURITY

#include <Servo.h>

// Pin Definitions const int ledPin = 2; // LED pin const int relayPin = 3; // Relay module pin (not used in this version) const int shockSensorPin = 4; // Shock sensor module pin (not used in this version) const int pinControl = 5; // Pin to control the system (used as the pin for HIGH/LOW check) const int servoPin = 6; // Servo motor pin

// Variables

Servo servoMotor;

bool authorized = false; // Initial state

void setup() {

pinMode(ledPin, OUTPUT);

pinMode(relayPin, OUTPUT);

pinMode(shockSensorPin, INPUT); // Keeping it for future expansion (not used here) pinMode(pinControl, INPUT); // Pin connected to HIGH/LOW for control

servoMotor.attach(servoPin);

Serial.begin(9600);

Serial.println("System Status: Unauthorized");

}

void loop() {

// Check the state of pinControl to decide if the system is authorized

if (digitalRead(pinControl) == HIGH) { // Pin is powered (HIGH, button pressed) authorized = true;

// Update system status to Authorized

Serial.println("System Status: Authorized");

digitalWrite(ledPin, LOW); // Turn off the LED to indicate authorized state

// Run the servo continuously

servoMotor.write(90); // Rotate servo to 90 degrees

delay(1000); // Keep at 90 degrees for 1 second

servoMotor.write(0); // Rotate servo back to 0 degrees

delay(1000); // Keep at 0 degrees for 1 second

} else { // Pin is LOW (connected to ground, button not pressed)

authorized = false;

// Update system status to Unauthorized

Serial.println("System Status: Unauthorized");

digitalWrite(ledPin, HIGH); // Blink the LED to indicate unauthorized state

// Blink the LED for unauthorized state

delay(500);

digitalWrite(ledPin, LOW);

delay(500);

}

// Relay remains off unless additional logic is implemented

digitalWrite(relayPin, LOW);

}

### FACE RECOGNITION CODE

import cv2 import face\_recognition import numpy as np

# Load a sample image and encode it (for recognition) known\_image = face\_recognition.load\_image\_file("owner.jpg") # Replace with the path to your image known\_encoding = face\_recognition.face\_encodings(known\_image)[0]

# Create a list of known face encodings and their names known\_face\_encodings = [known\_encoding] known\_face\_names = ["Owner"]

# Initialize the camera video\_capture = cv2.VideoCapture(0)

while True:

# Capture frame-by-frame ret, frame = video\_capture.read() small\_frame = cv2.resize(frame, (0, 0), fx=0.25, fy=0.25) # Resize for faster processing rgb\_small\_frame = small\_frame[:, :, ::-1] # Convert BGR to RGB

# Find all face locations and encodings in the frame face\_locations = face\_recognition.face\_locations(rgb\_small\_frame) face\_encodings = face\_recognition.face\_encodings(rgb\_small\_frame, face\_locations)

for (top, right, bottom, left), face\_encoding in zip(face\_locations, face\_encodings):

# Check if the detected face matches any known face matches = face\_recognition.compare\_faces(known\_face\_encodings, face\_encoding) name = "Unknown"

# Use the known face with the smallest distance if a match is found

face\_distances = face\_recognition.face\_distance(known\_face\_encodings, face\_encoding) best\_match\_index = np.argmin(face\_distances) if matches[best\_match\_index]: name = known\_face\_names[best\_match\_index]

# Scale back up face locations top, right, bottom, left = top \* 4, right \* 4, bottom \* 4, left \* 4

# Draw a rectangle around the face cv2.rectangle(frame, (left, top), (right, bottom), (0, 255, 0), 2)

# Label the face cv2.rectangle(frame, (left, bottom - 35), (right, bottom), (0, 255, 0), cv2.FILLED) font = cv2.FONT\_HERSHEY\_DUPLEX

cv2.putText(frame, name, (left + 6, bottom - 6), font, 1.0, (255, 255, 255), 1)

# Display the results cv2.imshow('Video', frame)

# Break loop on 'q' key press if cv2.waitKey(1) & 0xFF == ord('q'): break

# Release the camera and close windows video\_capture.release() cv2.destroyAllWindows()