Mini Project Report

on

SMART HOME DOORLOCK

Submitted by

Avinash pallapati : 24BEC036

Kesanapalli Madhava Naga Venkata Sai Ram : 24BEC019

JANPAREDDY SYAM SRIRAM

DURGESH: 24BEC016

Under the guidance of

SIBASANKAR PADHY

Assistant Professor

ELECTRONICS AND COMMUNICATION DEPARTMENT

DEPARTMENT OF ELECTRONICS AND COMMUNICATION ENGINEERING.

INDIAN INSTITUTE OF INFORMATION TECHNOLOGY DHARWAD

CERTIFICATE

This is to certify that the project proposal titled:

"AI-Based Smart Security and Door Locking System Using ESP32-CAM and Local Server"

has been successfully submitted by the following students:

- **J. Durgesh** – 24BEC016
- **K. Sairam** – 24BEC019
- **P. Avinash** – 24BEC036

of the **Department of Electronics and Communication Engineering**, in partial fulfillment of the academic requirements for the **Academic Year 2025–2026**.

SIBASANKAR PADHY
(Project Supervisor)

1. Title

AI-Based Smart Home Security and Door Locking System Using Local Machine Learning Server

2. Team Details

- **J. Durgesh – 24BEC016 (ECE)**
- **K. Sairam – 24BEC019 (ECE)**
- **P. Avinash – 24BEC036 (ECE)**

3. Abstract

This project presents an AI-powered smart security system designed for intelligent person detection and automated door access control. Although the original plan involved using an ESP32-CAM and cloud-based ML, hardware failures—specifically a non-functional ESP32-CAM and ESP32 Wi-Fi module—required major design modifications.

Due to these limitations, a **laptop camera** was used to capture images, and a **local machine learning server** running on the laptop replaced cloud integration. The ML model performs face recognition and classifies individuals as authorized or unauthorized. Based on the classification, the system triggers a buzzer and operates a solenoid door lock (where hardware is functional).

Even with hardware constraints, the project successfully demonstrates an AI-based recognition system integrated with access-control hardware.

4. Introduction / Problem Statement

Traditional security systems lack intelligence and cannot differentiate between authorized and unauthorized people. There is a growing need for smart systems that provide real-time analysis, remote control, and automated decision-making.

This project aims to build an AI-integrated security system capable of identifying individuals and controlling a door lock, offering a modern solution for homes, offices, and restricted areas.

5. Objectives

- To develop an AI-based face detection and recognition system.
- To integrate the ML output with buzzer and door lock mechanisms.
- To run the ML model on a **local server** due to ESP32 Wi-Fi failure.
- To design a cost-effective and modular security system.
- To allow future expansion toward cloud and mobile app integration.

6. Components Used

Hardware:

- Laptop Webcam (used instead of ESP32-CAM)
- ESP32 Module (used only for buzzer/lock control)
- Solenoid Door Lock
- Relay / MOSFET Driver
- Buzzer
- Power Supply
- Jumper Wires, Breadboard
- FTDI Programmer

Software:

- Python (OpenCV, ML Model)
- Flask Local Server
- ESP32 Arduino IDE Code

7. System Architecture (Actual Implementation)

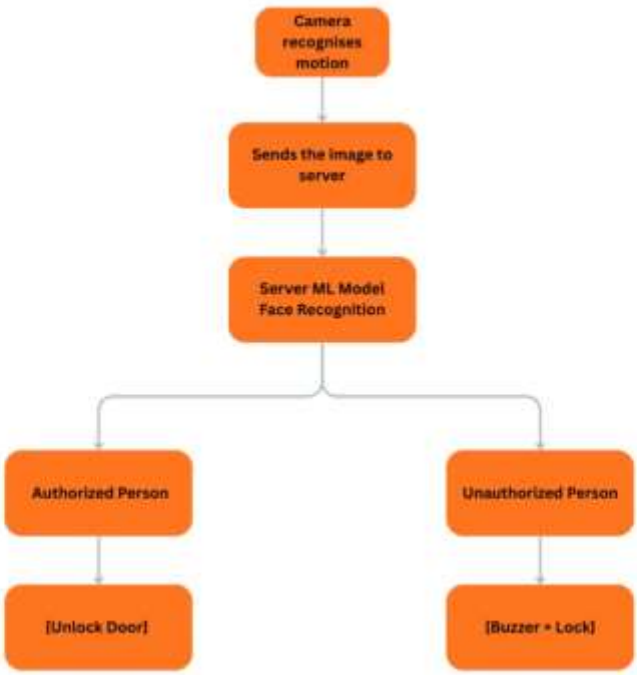
Hardware Workflow

1. Laptop webcam captures image of the person at the door.
2. Image is sent to a **local ML model** running on laptop.
3. ML model processes the image and identifies whether the person is authorized.
4. Based on result:
    - **Authorized** → Access allowed.
    - **Unauthorized** → Buzzer activates; lock remains engaged.

Software Workflow

- **Local ML Server:**
    - Receives image
    - Performs face recognition
    - Sends output to ESP32 (serial/local connection)
- **ESP32:**
    - Controls buzzer and solenoid lock
    - Works offline since Wi-Fi is not functional

Block Diagram



9. Working of the System

1. User stands in front of the camera.
2. Laptop captures the image and sends it to ML model.
3. The model compares the detected face with stored authorized images.
4. Classified as:
    - **Authorized:** No alarm; access granted.
    - **Unauthorized:** Buzzer activates; lock stays locked.
5. ESP32 receives the command and operates relay/MOSFET accordingly.

10. Advantages

- Low-cost and easily deployable.
- Local ML server ensures fast processing and no dependency on cloud.
- Hardware components are modular and expandable.
- Can be improved to support more users or more sensors.

11. *Drawbacks (Actual Project Limitations)*

1. ESP32-CAM Not Working

- Camera module failure prevented use of ESP32-CAM.
- Laptop webcam used instead, reducing system portability.

2. ESP32 Wi-Fi Not Working

Because Wi-Fi could not connect:

- Email alerts could not be implemented.
- Mobile app integration not possible.
- No cloud communication.
- ML had to run on a **local server**, not cloud.

3. Relay Module / Lock Issues

- Relay had hardware problems, so full solenoid lock operation could not be tested.
- Lock control was limited or simulated.

4. PIR Sensor Not Used

- Motion detection was unnecessary since the laptop continuously captures the frame.

12. Conclusion

Despite several hardware constraints, the project successfully implements the core intelligent security concept using AI-based face recognition and local server processing. The system can detect, recognize, and respond to unauthorized people and control hardware components such as buzzers and door locks.

Once ESP32-CAM, Wi-Fi, and relay hardware are fixed, the project can be fully upgraded to support:

- Cloud ML processing
- Email alerts
- Mobile app integration
- Real-time IoT dashboard

This makes the project scalable, modern, and suitable for future smart home automation.

13. Future Scope

- Integration of cloud ML for remote access
- Mobile app-based control
- Working ESP32-CAM for portable image capture
- Live video streaming
- Adding vibration, gas, or fire sensors for complete security system