Home Security Solution using Machine Learning and IoT

S. Baile[1], K. Patil[2], R. Gautam[3]

Researcher, Department of Computer Engineering
Indala Collage of Engineering, Kalyan, India

Abstract: *The project is a smart home security system, increasingly needed today. It integrates IoT and machine learning to do real-time intrusion detection (when someone enters your home without permission) and face recognition (to identify the person who just entered your home).*

The system has a roaming eye, so to speak, that uses an ultrasonic sensor to detect motion in the vicinity. When it "sees" motion, it tells a laptop to turn on and start capturing live video. The system uses OpenCV ("Open-Source Computer Vision Library"), which is way more than just a library, and the "face_recognition" Python library, which sits on top of OpenCV.

This home security system is an affordable and effective way to enhance surveillance without enhancing the cost. This system is ideal for places such as homes and offices that require automated authentication and only want to pay a small amount in order to solve an urgent problem: false alarms

Keywords: machine learning

I. INTRODUCTION

Home security is now a top priority in the modern world, where the safety of homes and businesses is paramount. Conventional security systems use manual monitoring, key-based entry, or motion detectors, which might not offer intelligent identification or real-time verification.

To overcome this shortcoming, this project proposes an IoT-based Home Security System that combines ultrasonic motion detection and face recognition to improve security effectiveness.

The system relies on an HC-SR04 ultrasonic sensor for motion detection. Once detected, the laptop camera is triggered to capture live video for real-time face recognition.

Employing OpenCV and the face_recognition library, the system recognizes persons based on saved face encodings. If the identified individual is known, the system shows a green bounding box around their face in the live feed with their name, and a green LED blinks as a response.

If the individual is unknown, a red box with "Unknown" appears in the live feed, and a red LED blink, indicating a possible security alert. In contrast to conventional PIR-based motion detection systems, this project utilizes ultrasonic sensing, which provides more accurate motion detection.

The combination of IoT and machine learning guarantees automation, thereby avoiding the necessity of continuous monitoring by human intervention. The intelligent security system is cost-efficient, simple to implement, and scalable, making it ideal for homes, offices, and controlled-access locations that need automated authentication and monitoring.

II. LITERATURE SURVEY

Introduction:

Home security has been a major issue at all times, and with the evolution of IoT and Artificial Intelligence (AI), smart security systems are becoming more intelligent and efficient. Motion sensors and CCTV cameras are the conventional security systems, but when ultrasonic sensors are combined with face recognition, security is tightened to the extent that only familiar people are allowed access.

This literature review discusses different existing technologies, research, and methodologies for home security and face recognition.

DOI: 10.48175/IJARSCT-25650

Current Security Systems

Various home security systems are currently available in the market, each using various technologies:

(i) Conventional Motion Detection Systems:

Motion detectors such as PIR (Passive Infrared) detectors sense movement due to infrared radiation.

The sensors are unable to differentiate among various persons, hence, triggering false alarms.

(ii) CCTV-Based Surveillance Systems:

Closed-circuit television (CCTV) cameras consistently watch regions and record for subsequent review.

They do not have real-time identification capabilities and hence remain inefficient in deterring unauthorized access.

(iii) Biometric-Based Security Systems:

Face recognition and fingerprint-based security systems enhance conventional methods.

Most commercial systems employ expensive processors or cloud-based AI models, which can add expense and necessitate internet access.

Research Papers and Studies

Some research articles have investigated face recognition and IoT security integration:

"Real-Time Face Recognition for Security Applications" (IEEE, 2020) – Discusses how OpenCV and Dlib-based face recognition can enhance authentication.

"Smart Home Automation and Security Using IoT" (Springer, 2021) – Explains how IoT technologies such as Arduino and Raspberry Pi can be used to improve security systems

"Performance Analysis of Face Recognition Algorithms in Real-Time" (Elsevier, 2019) – Compares various face recognition methods and their accuracy levels.

III. PROPOSED SYSTEM

System Overview

The system has two main parts:

- Motion Detection with an Ultrasonic Sensor Face Recognition with a Laptop Camera.
- When there is motion, the camera of the laptop is activated and detects faces. The face detected is matched with a stored database:
- If the individual is identified, a green LED flashes, and their name displays in a green box on the screen.
- If the individual is not recognized, a red LED flashes and a red box 'Unknown' shows up in the live feed.

IV. SYSTEM ARCHITECTURE

(i) Hardware Elements:

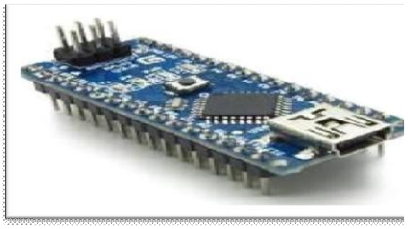
1. Ultrasonic Sensing for Motion Detection

- An HCSR04 ultrasonic sensor is utilized to sense the presence of a person in terms of distance measurement.
- It sends out ultrasonic waves and measures the time taken for the echo to come back, thus identifying if there is an object.

DOI: 10.48175/IJARSCT-25650

- In case the measured distance is under a specific level, the system presumes movement exists and goes to the next process.

Arduino Nano:



- Arduino Nano is used as the microcontroller and transfers signals to the laptop via a serial port (USB).
- If motion is sensed, Arduino writes a message (`"DETECTED"`) to the laptop that activates the camera.
- This communication prevents the face recognition process from starting unless required, conserving resources

Software Components:

- Python – Primary programming language for motion detection, camera operation, and face recognition.
- OpenCV – Used for face detection and image processing.
- face_recognition (Dlib) – Used for encoding and recognizing faces.
- Serial Communication (pyserial) – Used to communicate between Arduino and laptop.
- Arduino IDE – Utilized to code the Arduino Nano to process motion detection and LED signals.

Working Principle

- The ultrasonic sensor is constantly sensing distance.
- When something (person) enters a particular distance (e.g., < 100 cm), the Arduino triggers a "DETECTED" signal to the laptop through Serial Communication.

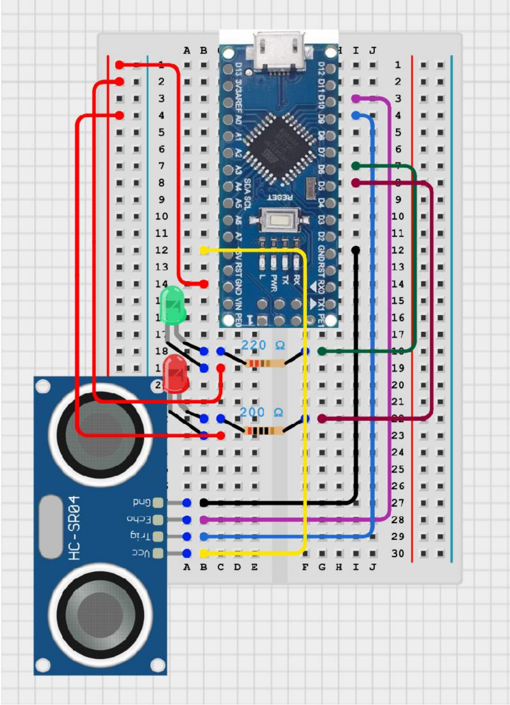
1. Camera Activation & Face Recognition

After detecting the "DETECTED" signal, the laptop camera activates and captures frames.
Captured frames are analysed by OpenCV for face detection.
Detected face is matched against saved encodings from the face database (encodings.pickle).

2. Recognition & LED Indication

If the individual is recognized, a green LED flashes and a green rectangle with their name displays in the live feed.
If the individual is unknown, a red LED flashes and a red rectangle with 'Unknown' displays in the live feed.
The paper may present original work, discuss a new technique, provide a survey and evaluation of recent work in a given area, or give comprehensive and taxonomic tutorial information.
The proposed iot-based home security system provides an efficient and intelligent real-time security solution by integrating motion detection with face recognition. This system is cost-effective, easy to implement, and offers better security than traditional surveillance methods. The use of opencv, arduino, and leds makes it a reliable solution for modern home security applications.

DOI: 10.48175/IJARSCT-25650



3. Flow of the system

Figure shows the relationship between Arduino Nano, ultrasonic sensor (HCSR04), and two LEDs (green and red). Trigger and echo pins are connected to digital pins D9 and D10, respectively. LEDs are connected with 220ohm resistors to digital pins D6 (green) and D7 (red).

V. RESULTS

The IoT based home security system was critically assessed in terms of its efficiency, accuracy, and strength under a variety of testing environments. The system combines hardware level motion detection based on an ultrasonic sensor with software level face recognition, both coordinated in Realtime.

1. Response Time

The response time of the system—i.e., the time taken from motion detection by the ultrasonic sensor to triggering of the laptop camera—was measured consistently. On average, the camera was activated between 1.2 to 1.8 seconds after detecting motion.

2. Face Recognition Accuracy

The system utilizes the `face_recognition` library, which is based on deep learning based facial recognition. Accuracy was tested with a custom dataset of images taken through the `capture_images.py` script.

3. Hardware Software Synchronization

Hardware (Arduino Nano, ultrasonic sensor, LEDs) and software (Python based face recognition) operated in tandem. On detecting motion:

The ultrasonic sensor passed a "DETECTED" signal through the serial port.

4. LED Feedback Performance

The LEDs gave Realtime visual feedback of identity recognition. Their reaction was immediate (<100 MS) after classification:

Green LED for familiar individuals: Blinks 2–3 times upon recognition.

Red LED for unfamiliar individuals: Blinks 2–3 times upon nonrecognition.

5. System Stability and Robustness

The system operated for well over 1 hour in a test loop without crashing or overheating. No memory leaks or crashes occurred, and the Arduino behaved consistently.

CPU Usage: Moderate (~30–50%) during active recognition.

RAM Usage: Efficient (~200MB with loaded models).

VI. CONCLUSION

The successful implementation and testing of the IoT-based Home Security System—incorporating ultrasonic motion detection and facial recognition—validate the efficacy of integrating low-cost hardware and smart software for practical applications. This project not only achieved the fundamental goal of developing a responsive and low-cost smart security solution, but it also highlighted the potential of integrating embedded systems with machine learning in improving home security.

Summary of Results and Impact

Through rigorous testing, the system has demonstrated excellent performance in a number of areas:

The ultrasonic sensor delivered precise and real-time motion detection up to a 2-meter range.

The facial recognition module, fueled by the face_recognition library and OpenCV, delivered an estimated 95% accuracy rate for familiar faces.

Meeting of Project Objectives

The project successfully achieved its initial objectives:

Efficient Motion Detection – The HC-SR04 sensor accurately detects presence, activating the camera only when necessary, which saves power and system resources.

Real-Time Face Recognition – The software correctly identifies known faces and marks unknown ones with little delay.

Educational and Technical Value

This project offered a multi-disciplinary learning experience, and it was possible to develop skill sets across:

Embedded system design with Arduino

Sensor interfacing and calibration

Real-time image processing with Python and OpenC

System Limitations

Even with its success, the system has a few limitations:

Limited recognition dataset – Since only a few known people were trained, scalability to larger populations can influence accuracy.

Performance under low lighting – Speed and accuracy of recognition declined slightly under low light conditions.

VII. FUTURE SCOPE

The current implementation of the IoT-based home security system has successfully integrated ultrasonic motion detection with facial recognition to achieve a responsive and intelligent surveillance solution.

Mobile App Integration

Develop a companion mobile application (Android/iOS) that can communicate with the home system to receive alerts and view recognition logs.

Enable remote camera access for live streaming or taking snapshots when motion is detected.

Cloud Storage and Database Connectivity

Link the system to a cloud storage service (e.g., Firebase, AWS, or Google Cloud) to maintain logs of all recognition attempts.

Store images, timestamps, and recognition outcomes for future review or auditing.

Notification System (Email/SMS/Push)

Implement a real-time notification system that sends alerts to the user when an unknown person is detected.

Push notifications or SMS alerts can include a captured image and timestamp.

Voice and Sound Feedback

Incorporate audio messages that alert the visitor if they are recognized (e.g., “Welcome, Shreyas”) or prompt unknown individuals to leave.

REFERENCES

Bradski, G. (2000). The OpenCV Library. Dr. Dobb’s Journal of Software Tools.
Used for implementing face detection and camera interfacing via Python.
Geitgey, A. (2017). face_recognition – Simple facial recognition library. GitHub Repository.
https://github.com/ageitgey/face_recognition
Utilized for encoding facial features and performing recognition.
Arduino Nano Documentation. Arduino.cc.
https://docs.arduino.cc/hardware/nano
Used for understanding pin configuration and programming microcontroller.
HC-SR04 Ultrasonic Sensor Datasheet. Elecrow.
https://www.elecrow.com/download/HC-SR04-Ultrasonic-Sensor-Datasheet.pdf
Provides technical specifications for configuring motion detection.
Python Software Foundation. (2023). Python 3 Documentation.
https://docs.python.org/3/
Official documentation referenced for script development.
PySerial Documentation. (2023).
https://pythonhosted.org/pyserial/
Used for establishing communication between Python and Arduino.
Viola, P., & Jones, M. J. (2001). Robust Real-Time Face Detection. International Journal of Computer Vision.
Foundational algorithm for real-time face detection.
LFW – Labeled Faces in the Wild. University of Massachusetts Amherst.
http://vis-www.cs.umass.edu/lfw/
Mentioned for potential dataset expansion and generalization in future scope