SMART DOOR LOCK SYSTEM(SENTRY)

ABHIMANYU SINGH
Computer Science
Binghamton University
asing134@binghamton.edu

ARNISH SAHAI
Computer Science
Binghamton University
asahai1@binghamton.edu

Abstract

The Internet of things is one of the fields that have been used to provide remote monitor and control for the home appliances. In this project we implemented a smart door lock system based on Raspberry pi technology where the camera has been utilized to provide secure door locking and unlocking mechanism.

Keywords

IOT, Raspberry Pi, Door Lock, Android App

1.Introduction

Security and safety are becoming more and more popular day by day and it is getting improved and used for ease in our life. Nowadays, technology has become an integrated part of people's lives therefore the security of one's home, office or organization must also not be left behind. Our Smart Door Lock System(SENTRY) is mainly designed and developed for the security purposes.In this system, the authorized individuals are only the ones who will get the permission to access the doors. This system also encompasses an android application through which we can remotely control the access to the door and provide live feed to the house owner, using which he/she can check who is outside the door.

2.Design

2.1 Hardware

2.1.1 Raspberry Pi

We have used Raspberry Module as our computation device which hosts our software, and collects reading from attached peripherals and performs actions based on set condition.Processor speed ranges from 700 MHz to 1.4 GHz for the Pi 3 Model B+ or 1.5 GHz for the Pi 4; on-board memory ranges from 256 MiB to 1 GiB random-access memory (RAM), with up to 4 GiB available on the Pi 4.

2.1.2 RPi Camera

The Raspberry Pi Camera v2 is a high quality 8 megapixel Sony IMX219 image sensor custom designed add-on board for Raspberry Pi, featuring a fixed focus lens. It's capable of 3280 x 2464 pixel static images, and also supports 1080p30, 720p60 and 640x480p60/90 video. It attaches to Pi by way of one of the small sockets on the board upper surface and uses the dedicated CSi interface, designed especially for interfacing to cameras. We have used this hardware to collect video data in order to detect faces and perform actions, stream video.

2.1.3 Servo

A servomotor is a rotary actuator or linear actuator that allows for precise control of angular or linear position, velocity and acceleration. It consists of a suitable motor coupled to a sensor for position

feedback. We have made use of the servo inorder to control our door lock.

2.2 Software and Libraries

2.2.1 OpenCV

OpenCV (*Open Source Computer Vision Library*) is a library of programming functions mainly aimed at real-time computer vision. We have used functions available in this library to implement our facial Recognition Part.

2.2.2 Python

Python is an interpreted, high-level, general-purpose programming language.Python's design philosophy emphasizes code readability with its notable use of significant whitespace. Its language constructs and object-oriented approach aim to help programmers write clear, logical code for small and large-scale projects. We have used this library to write our face recognition software

2.2.3 Android Studio

Android Studio is the official integrated development environment (IDE) for Google's Android operating system, built on JetBrains' IntelliJ IDEA software and designed specifically for Android development. We have made use of this to make our Android Application for the project.

2.2.4 Java

Java is a general-purpose programming language that is class-based, object-oriented, and designed to have as few implementation dependencies as possible. It is intended to let application developers *write once, run anywhere* (WORA), meaning that compiled Java code can run on all platforms that support Java without the need for recompilation. We have made use of this language to write our Application.

2.2.5 Motion

Using this software we implemented our live stream capability in our project.

2.2.6 Twilio

Twilio provides us with an API for sms notification, using this api we implemented the notification feature of our project.

3.Implementation

3.1 Face Recognition

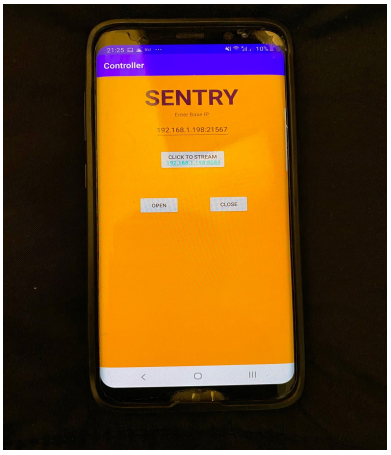
For face recognition, the first thing to do is to gather face data, a python script was written inorder to do this task. The training of our algorithm was done so that it can recognize the faces. Then from the video gathered from the Pi-Cam, using a pre-trained haarcascade_frontalface_default.xml file we detected faces in the video and then we gathered face embeddings and then compared them with our known faces.

3.2 Android Application

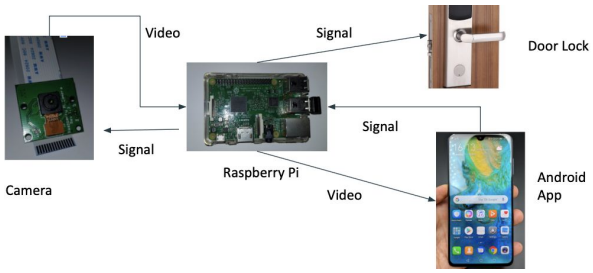
Three buttons are provided in our

1. Live Stream Button
2. Open Button
3. Close Button

A server was Set-upped in a raspberry-pi base station which received and processed the input sent by the app and performed the instructed task.



4.Data-Flow



This is the control flow as it can be clearly seen Raspberry pi is the main component, it sends out

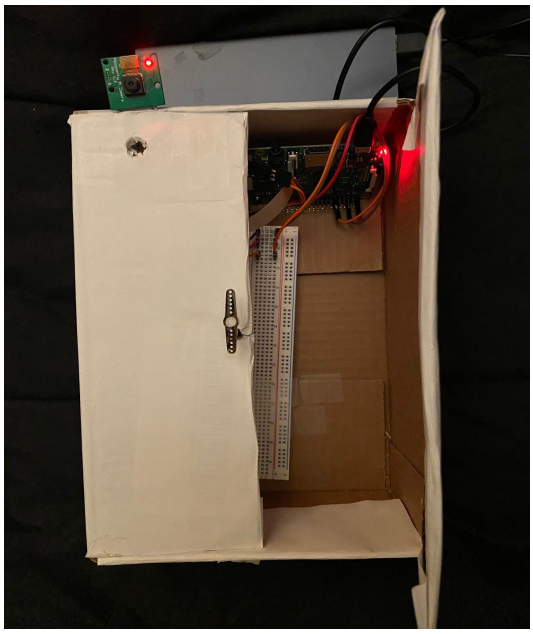
signal to Pi Camera to receive video from it, using video our face recognition module tries to find match, if match is found door is unlocked automatically if not notification is sent to the owner, notifying him about a stranger at the the door. Owner can check who is at the door using the live feed on the mobile app and then decide whether to unlock the door or not using the app.

5.Conclusion

We were able to achieve all the declared features,

1. Face Unlock
2. Remote Lock/Unlock Using App
3. Notification
4. Video Live stream

All the functions worked flawlessly while handling all the possible scenarios. Our door was unlocked as soon as it detected an authorised person. Notification was sent to the owner if a stranger was detected continuously for 20 seconds. We were able to unlock the door and see a live video feed in front of our door using our mobile app(demonstrated project video).



6.Refernces

[1] Hussein, Naser & Mansoori, Inas. (2017). Smart Door System for Home Security Using Raspberry pi3. 395-399. 10.1109/COMAPP.2017.8079785.

[2]Joseph Redmon , Santosh Divvala, Ross Girshick , Ali Farhadi, 2015. You Only Look Once: Unified, Real-Time Object Detection