Intrusion Alert System with Raspberry Pi

Jadhav Neha Kishor, 2001EE21
Electrical Department
Indian Institute of Technology, Patna
Patna, India

***Abstract*—In this paper focusing on home security, we intend to propose an intrusion alert system. Stopping thefts and crimes while they are in progress is the main motivation of our paper. The proposed system consists of means for sensing an unauthorized entry, location, capturing image of the unauthorized object/person, an integrated alarm unit: sound alarm, LED lights and a written warning on LCD display to deter intruders. It also features a means for communication interface to transmit the relevant information to the user.**

Keywords—Raspberry Pi, Intrusion Alert System, Home security system

I. Introduction

Internet of Things incorporates the internet and physical objects involving various domains such as home automation, human health and environmental monitoring industrial process. In this paper, we intend to propose an intrusion alert system. Stopping thefts and crimes while they are in progress is the main motivation of our paper. On average, every 3 minutes a burglary or robbery happens in India amounting to several thousand crores of loss annually according to National Crime Records Bureau (NCRB).Yet, nearly 64% of Indians are not equipped to handle home intrusions [1]. Research has shown that burglars are 3 times more likely to target properties that don't use a security system with a camera [2]

A smart intruder alert system provides a means for sensing an unauthorized entry, capturing image of the unauthorized object, an integrated alarm unit: sound alarm through Buzzer, LED lights as well as LCD Display giving written warning to deter intrusion and a means for communication interface to transmit the related information to the user.

II. RELATED WORKS

Govinda et al. (2014) first discussed the implementation of smart home-based security system [3]. Two methods were suggested for implementation of home security over IoT. One of which uses web cameras that detects motion then sounds an alarm and informs the owner through email. This method is noble for detecting motion and finding a revolutionary method of notifying the user through mail. Jayashri and Arvind (2013) implemented access authentication system through fingerprint to unlock a door [4]. Setting up fingerprint sensors is a complex task, as integration to IoT set up requires sensors with high resolution. Some scholars are of the opinion that using fingerprint sensor is not entirely safe because it is easy replicate fingerprints by lifting them.

In 2016, intruder detection using LDR sensors and Laser ray was proposed. In implementing this, a laser was directed towards a LDR sensor which sounds an alarm connected to the sensor when there is a detachment between laser and LDR sensor thereby alerting the neighbors and sends an alert to the homeowner [5]. This solves the challenge of covering out of range places by fixed cameras.

III. Requirements

Raspberry Pi 3B acts as a mini computer which makes the functioning of the system possible. A PIR (Passive Infrared) Sensor uses a pair of pyroelectric sensors to detect heat energy in the surrounding environment thus detecting the presence of an intruder. Pi Camera is attached to capture images of the intruder. A 16x2 LCD display is used along with an i2c module to send out warnings. The display is accompanied by buzzer and LED lights to send warning signs to the intruder. All of these components, except the display and Raspberry Pi are mounted on a breadboard using jumper wires. The following is a circuit diagram of the proposed system.

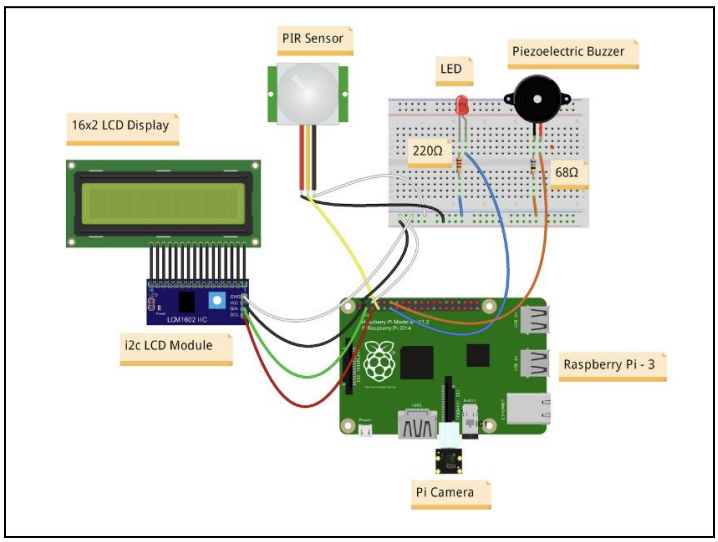

Fig.1: Circuit Diagram of Intrusion Alert System

IV. Methodology

After assembling the circuit, the Raspberry Pi 3 was powered and Wi-Fi was enabled for connectivity to the internet. After activating Python, a script to import libraries was written.

A. Importing Libraries

The code is written in python and makes use of the drivers module cloned from lcd [6].

```
import RPi.GPIO as gpio
import picamera
import time

# make sure to have the drivers folder on the same parent folder as this python file
import drivers
```

```
import smtplib
from email.MIMEMultipart import MIMEMultipart
from email.MIMEText import MIMEText
from email.MIMEBase import MIMEBase
from email import encoders
from email.mime.image import MIMEImage
```

B. *Configuring GPIO pins*

Here we configure the global variables by setting the GPIO pin ids (corresponding to the circuit diagram). We configure the LED and buzzer pins as output and PIR sensor as input as data is received from it. They are initially turned off to ensure that the program starts correctly.

```
# Set the GPIO pins ids
led=17
buzzer=18
pir=4

HIGH=1
LOW=0

gpio.setwarnings(False)
gpio.setmode(gpio.BCM)
```

```
# Output GPIO pins
gpio.setup(led, gpio.OUT)
gpio.setup(buzzer, gpio.OUT)

# Input GPIO pins
gpio.setup(pir, gpio.IN)
data=""

# Setup LED and buzzer
gpio.output(led, LOW)
gpio.output(buzzer, LOW)
```

C. *Mailer Object*

We create the MIMEMultipart mailer object and set its sender address/password, receiver address, Subject, and the body. We will attach the image to this configuration while sending it.

```
# Configure the to and from email address
from_email = "anshulthakkar25@gmail.com"
from_email_pwd = "Raspberry@4"

to_email = "anshulthakkar25@gmail.com"

mail = MIMEMultipart()
```

```
mail['From'] = from_email
mail['To'] = to_email
mail['Subject'] = "Someone is at Your Doorstep! "
body = "Hey Anshul,\nSomeone tried to open your door. See the below attachment
```

D. *Camera Object*

The camera object is created from the PiCamera library and is formatted according to the position (by rotating it 180 degrees) and setting the brightness and auto white balance mode.

```
# Setup camera
camera = picamera.PiCamera()
camera.rotation=180
camera.awb_mode= 'auto'
camera.brightness=55
```

E. *LCD Display Object*

Using the drivers library, the LCD display object is created as follows.

```
# Setup LCD display from the driver package
display = drivers.Lcd()
```

F. *Create Mail with image attachment and send it to user*

The following function creates the mail by attaching the locally stored image specified by the name data. It then establishes the server connection and TLS (transport layer security). Using the credentials specified in the global variable, it logs in to the Gmail account and sends the mail through the Simple Mail Transfer Protocol (SMTP) with port id 587. Finally, it updates the LCD display and console with the message “Mail Sent” and quits the connection to the server.

```
# Function to send the mail to the owner with images attached
def sendMail(data):

    # Create mail object with image attachment and text
    mail.attach(MIMEText(body, 'plain'))
    dat='%s.jpg'%data
    attachment = open(dat, 'rb')
    image=MIMEImage(attachment.read())
    attachment.close()
    mail.attach(image)
```

```
# Estabilish connection and send the mail
server = smtplib.SMTP('smtp.gmail.com', 587)
server.starttls()
server.login(from_email, from_email_pwd)
text = mail.as_string()
server.sendmail(from_email, to_email, text)
```

```
# Notify that mail has been sent in LCD and console
print('Mail Sent')
display.lcd_clear()
display.lcd_display_string("Mail Sent", 1)

server.quit()
```

G. *Capturing and Saving the Image*

The following function captures the image from the pi-camera with a timer delay and saves it locally under the same directory as the python script. It names the image with the date and time so as to provide more information to the owner

about the time of intrusion. It updates the LCD and console when the capture is over and triggers the sendMail() function with the name of the image file to be attached to the mail.

```
def capture_image():
    # Capture the image and save the name as the time
    data = time.strftime("%d_%b_%Y|%H:%M:%S")
    camera.start_preview()
    time.sleep(3)
    camera.capture('%s.jpg'%data)
    camera.stop_preview()
```

```
# Notify that image has been captured in LCD and console
print('Image Captured')
display.lcd_clear()
display.lcd_display_string("Image", 1)
display.lcd_display_string("Captured", 2)
time.sleep(1)

# Send the mail
sendMail(data)
```

H. Main Loop

This is the part of the program that creates the forever loop. During each iteration, the program checks if the PIR sensor detects any motion. If so, it turns on the LED, buzzer and displays a message on the LCD monitor. Finally, it triggers the capture_image() method which captures and saves an image of the intruder locally. It further changes the message in the LCD and triggers the sendMail() method. Inside the send mail, we create a mailer object with text and image, establish a connection with the Gmail server, log in to the raspberry pi Gmail account created, and send the data. We finally update the LCD display to show the appropriate message. We now disconnect from the server and return back to the main infinite loop. Inside the loop, we again detect if there is any motion. If so, we do the above-mentioned process again, or otherwise, we clear the LCD display and turn off the buzzer and LED, and wait till there is any motion again.

```
# Run a forever loop
while 1:
    # PIR sensor detects any motion
    if gpio.input(pir)==1:
        # Trigger the led and buzzer
        gpio.output(led, HIGH)
        gpio.output(buzzer, HIGH)
```

```
# Display that intrusion has been detected in the LCD and console
print('Intrusion Detected')
display.lcd_clear()
display.lcd_display_string("Intrusion", 1)
display.lcd_display_string("Detected", 2)
```

```
# Capture the image and send the mail
capture_image()

# Wait for the PIR sensor to not detect any motion
while(gpio.input(pir)==1):
    time.sleep(1)
```

```
# PIR sensor doesn't detect any motion
else:
    # Turn off LED and buzzer and clear the LCD display
    gpio.output(led, LOW)
    gpio.output(buzzer, LOW)
    display.lcd_clear()
    time.sleep(0.01)
```

V. Remote Controlling

The system works independently, without the requirement of human input. The system is always active by default and can be de activated by a line of command. The system sends updates at every instance of an intruder activating the sensors. <insert photo of email>

VI. Future Scope

The scope of the project could be extended to include face recognition. Right now, the intrusion is triggered if any motion occurs to near the doorstep or any disturbance is caused to the door. However, it is helpful if it is only triggered and sends a mail if a person is in the door step. Further, it is unnecessary for it to alert the owner or the people they have whitelisted if it identifies their entry. To tackle this, facial recognition could help in selectively identifying the intruder. To extend it even further, microphone could be added to record the sound/voice of the intruder. To provide more context, instead of a picture, a video with the audio could be recorded and sent to the owner. An all-in-one system of this sort would typically cost thousands of rupees in the market but it could be added with a few additions to the code and hardware.

VII. Conclusion

This project provides an all-in-one system to alert about an intrusion, help in catching the intruder and possibly deter the intrusion from taking place. Since the project is built using raspberry pi, it could be extended further to add more functionalities mentioned previously.

Acknowledgments

“Acknowledgment(s)” is spelled without an “e” after the “g” in American English.

As you can see, the formatting ensures that the text ends in two equal-sized columns rather than only displaying one column on the last page.

This template was adapted from those provided by the IEEE on their own website.

REFERENCES

[1] https://www.indiatoday.in/impact-feature/story/every-3-minutes-a-burglary-robbery-or-a-break-in-is-taking-place-in-india-and-its-time-we-reconsidered-the-safety-levels-of-our-homes-1621113-2019-11-21

[2] https://safeatlast.co/blog/burglary-statistics/#gref

[3] Govinda K and Sai Krishna Prasad K and Sai ram susheel 2014 Intrusion detection system for smart home using laser rays International Journal for Scientific Research & Development (IJSRD) 2 176-78

[4] Jayashri B and Arvind S 2013 Design and Implementation of Security for Smart Home based on GSM technology International Journal of Smart Home 7 201-08

[5] Cristian C, Ursache A, Popa D O and Florin Pop 2016 Energy efficiency and robustness for IoT: building a smart home security system Faculty of Automatic Control and Computers University Politehnica of Bucharest, Bucharest, Romania 43

[6] https://github.com/the-raspberry-pi-guy/lcd