

# **Chapter 1**

## **Introduction**

### **1. Introduction to IOT:**

In this paper, we present an Internet of Things sensor system to monitor the fill level of municipal trash cans. Smart cities have been identified as a promising potential application domain for the Internet of Things, with a wide range of possible services that can benefit city administration and citizens alike. One service that can be provided in a smart city is smart waste management. Public trash cans detract from the surrounding environment when they are full for long periods of time. On the other hand, it can be an expensive operation to send garbage trucks to every trash can in the city; if cans are empty, the journey accomplishes nothing. Cities develop rough algorithms for minimizing cost of various municipal services such as collecting trash, but Internet of Things sensors can improve these services by notifying relevant public works officials when particular trash cans are full.

### **1.1 Characteristics of IoT:**

#### **INTELLIGENCE**

Together algorithms and compute (i.e. software & hardware) provide the “intelligent spark” that makes a product experience smart. Consider Misfit Shine, a fitness tracker, compared to Nest’s intelligent thermostat. The Shine experience distributes compute tasks between a smartphone and the cloud. The Nest thermostat has more compute horsepower for the AI that make them smart.

#### **CONNECTIVITY**

Connectivity in the IoT is more than slapping on a Wi-Fi module and calling it a day. Connectivity enables network accessibility and compatibility. Accessibility is getting on a network while compatibility provides the common ability to consume and produce data. If this sounds familiar, that’s because it is Metcalfe’s Law and it rings true for IoT.

#### **SENSING**

We tend to take for granted our senses and ability to understand the physical world and people around us. Sensing technologies provide us with the means to create experiences that reflect a true awareness of the physical world and the people in it.

#### **EXPRESSING**

Expressing enables interactivity with people and the physical world. Whether it is a smart home or a farm with smart agriculture technology, expressing provides us with a means to create products that interact intelligently with the

real world. This means more than just rendering beautiful UIs to a screen. Expressing allows us to output into the real world and directly interact with people and environment.

## **ENERGY**

Without energy we can't bring our creations to life. The problem is we can't create billions of things that all run on batteries. Energy harvesting, power efficiency, and charging infrastructure are necessary parts a power intelligent ecosystem that we must design. Today, it is woefully inadequate and lacks the focus of many product teams.

## **SAFETY**

As we gain efficiencies, novel experiences, and other benefits from the IoT, we must not forget about safety. As both the creators and recipients of the IoT, we must design for safety. This includes safety of our personal data and the safety of our physical well-being. Securing the endpoints, the networks and the data moving across all of it means creating a security paradigm that will scale.

## **ARCHITECTURE**

The Internet of Things is considered as the third wave of the World Wide Web (WWW) after static web pages and social networking based web. The IoT is a worldwide network that connects different type of objects at anytime and anywhere via a popular internet protocol named Internet Protocol (IP) [13, 19]. According to most of the researcher's opinions about conventional IoT architecture, it is considered as three layers: - 1) Perception Layer 2) Network Layer 3) Application Layer In other aspects, some researchers analyzed one more layer which is also included in IoT's latest architecture that is a support layer that lies between the application layer and network layer. The support layer consists of fog computing and cloud computing. The cloud computing is also the hottest topic today in research. The perception layer is also called the recognition layer. The perception layer is the lowest layer of the conventional architecture of IoT. This layer's main responsibility is to collect useful information/data from things or the environment (such as WSN, heterogeneous devices, sensors type real world objects, humidity, and temperature etc.) and transform them in a digital setup. The main purpose of objects is unique address identification and communication between short-range technologies such as RFID, Bluetooth, Near-Field Communication (NFC), 6LoWPAN (Low Power Personal Area Network) This layer is the brain of conventional IoT architecture This layer's main responsibility is to help and secure data transmission between the application and perception layer of IoT architecture.

## **1.2 Problem Statement:**

Need to build a prototype for intruder alert system using a microcontroller Raspberry pi 3.

Intruder alert system should use the following function:

- The Smart home security system should be fitted inside the room if any obstacle is detected by the pirsensor, the bulb glows.
- It also simultaneously clicks the photo of the intruder.
- Whenever the light glows, an alert email notification is send to the owner of the house along with the photo attachment and the owner is able to get the image of the intruder, also the owner gets notification mobile.
- Build Java ME program to implement the relay.
- Python program to implement the pir sensor and activate the alert system and clicks the photo through USB Web camera.
- Python program to get the pushover notification.

## **1.3 Objectives:**

The device designed in this project can be installed at the main entrance of a house. It detects motion of any visitor with the help of PIR sensor and starts capturing the images with the help of a USB web cam. The images are temporarily stored on the Raspberry Pi and pushed to the Google Cloud from where they are sent as email alert to the house owner. So, the user gets the images of any visitor immediately on email which he can check from his smart phone. The Raspberry Pi connects with the Google Cloud over TCP-IP stack. The Raspberry Pi 3 is one of the IoT boards which comes equipped with on-board TCP/IP stack, so, it can be readily connected to an IoT network. The Pi uses OpenCV library to capture images from the Web Cam and send them over registered Email address of the user. The home security system designed in this project, though being simple, is a powerful an application. The user can keep surveillance of his house from anywhere, any time and always by just installing this small device at the main entrance. Many such devices can also be installed to further add security layers. The entrance of any intruder can be detected and alerted by the Email on the smart phone, then the user is free to take appropriate action like calling police, informing law enforcement etc.

## Chapter 2

### Literature survey

#### 2. Literature survey:

1) “ A Literature Survey on Smart Home Automation Security” published by Mr.Rohit Ragmahale, Dr. D. Y. Patil College of Engineering, Ambi, Pune University, Maharashtra. This paper presents a detailed description of different technologies and home automation systems from a security point of view. This paper highlights various security flaws in existing home automation systems and how the concept of security and the meaning of the word “intruder” have evolved over time.

2)Sanjana Prasad, P.Mahalakshmi, A.Jhon Clement Sunder, R.Swathi, “Smart Surveillance Monitoring System using Raspberry Pi and PIR Sensor”. This paper deals with the design and implementation of Smart surveillance monitoring system using Raspberry pi and PIR sensor for mobile devices. It increases the usage of mobile technology to provide essential security to our homes and for other control applications. The proposed home security system captures information and transmits it via a 3G Dongle to a Smart phone using web application. Raspberry pi operates and controls motion detectors and video cameras for remote sensing and surveillance, streams live video and records it for future playback.

3)G.Senthil kumar, V.Sathish Kumar, K.GopalaKrishnan,”Embedded Inage Capure System using Raspberry Pi System”. This paper proposes the Smart Surveillance System using Thing speak and Raspberry pi. This design is a small portable monitoring system for home and office security. The model uses hardware mechanism such as Raspberry pi (model B), Gyro sensor and Raspberry pi camera. This system will monitor when motion detected, the Raspberry Pi will control the Raspberry Pi camera to take a picture and sent out an alert email with the image to the user by using Wi-Fi adaptor according to the program written in python environment

## 2.1 Motivation

We are living in an age where tasks and systems are fusing together with the power of IOT to have a more efficient system of working and to execute jobs quickly! With all the power at our finger tips this is what we have come up with.

The Internet of Things (IoT) shall be able to incorporate transparently and seamlessly a large number of different systems, while providing data for millions of people to use and capitalize. Building a general architecture for the IoT is hence a very complex task, mainly because of the extremely large variety of devices, link layer technologies, and services that may be involved in such a system.

According to the survey of Mumbai police in the city of Mumbai which is one of the fastest growing cities in India, in the year 2015 alone, there had been 1200 cases of home burglaries in which only 240 cases were solved. Such burglaries amounted to loss property worth Rupees 15 crore and 72 lakhs out of which only valuables worth rupees 2 crore has been recovered. Frequency of such burglaries is found to be more in the suburban areas of Mumbai but even in the heart of the city such incidences are happening.

If we consider above cases, then the first question which comes to our mind 'Is our home really secure?' We start to think whether the security measures we have taken are enough. By placing an appropriate home security system, one can protect our home from criminals. Nowadays, safety depends on how well trained you are and how well secured you are. Of course, you can empower yourself and give yourself as well as your family peace of mind with monitored residential security systems. Smart Home security system is thus applicable and desirable for resident's safety and convenience. This will be achieved by turning your home into a smart home by intelligent remote monitoring. Smart home comes into picture for the purpose of controlling and monitoring the home. It will give you peace of mind, as you can have a close watch and stay connected anytime, anywhere.

## Chapter 3

### Proposed system

#### 3. Proposed system:

##### 3.1 Introduction of proposed system and architecture:

- The owner of the house would get an email on the registered email address with an image of the activity happening outside his house.
- As the owner of the house gets an image of the activity happening outside the house, if there is any suspicious activity happening then the image received will help the authorities to investigate into the matter.
- The owner also gets a push notification on cell phone.
- Whenever a motion is detected the lights get on outside the house.

##### 3.2 Architecture:

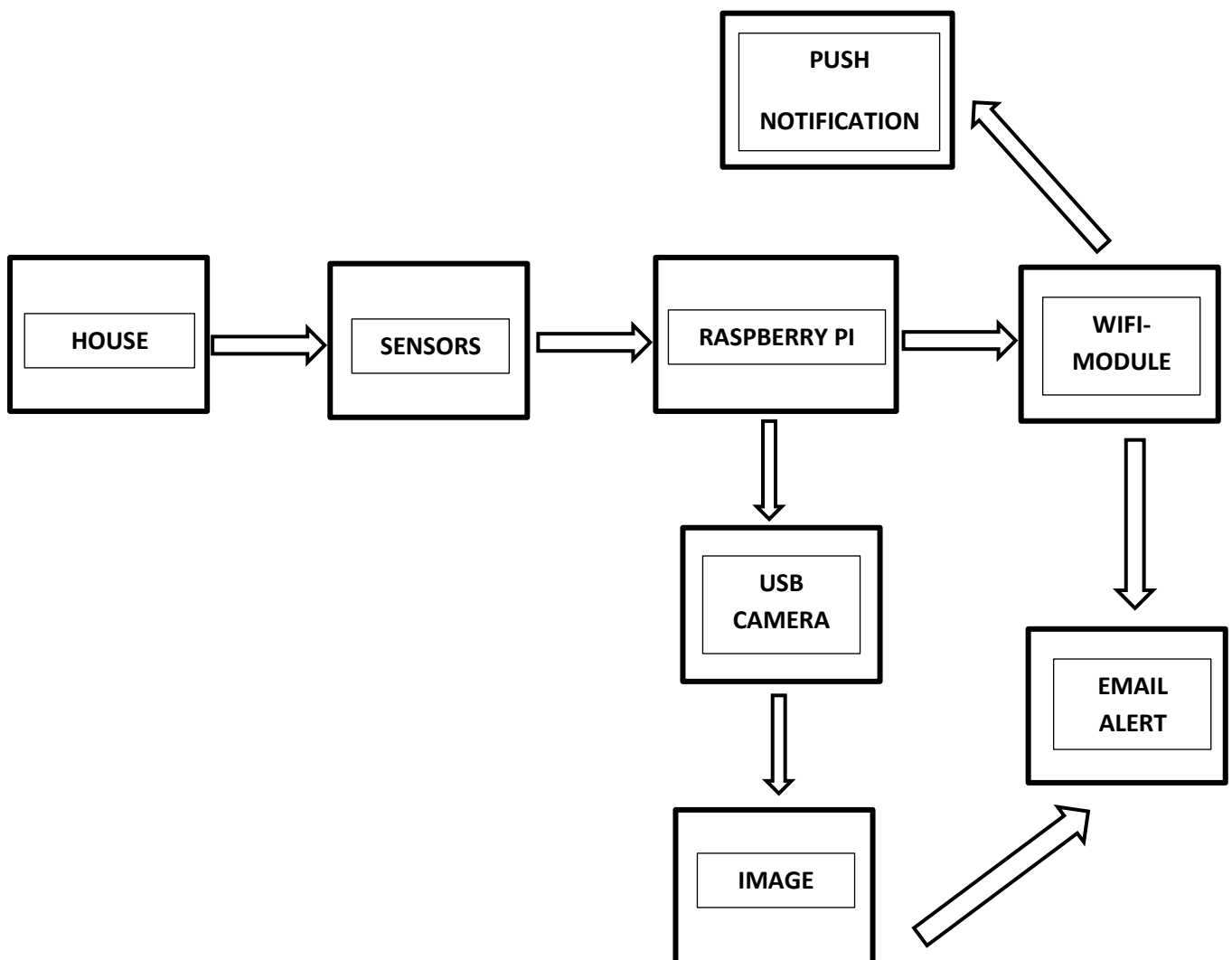


Fig 3.1 Architecture of Project

### 3.3 Hardware and software requirements :

- **Raspberry Pi 3B+ :**

The Raspberry Pi is a series of small single-board computers developed in the United Kingdom by the Raspberry Pi Foundation to promote teaching of basic computer science in schools and in developing countries.

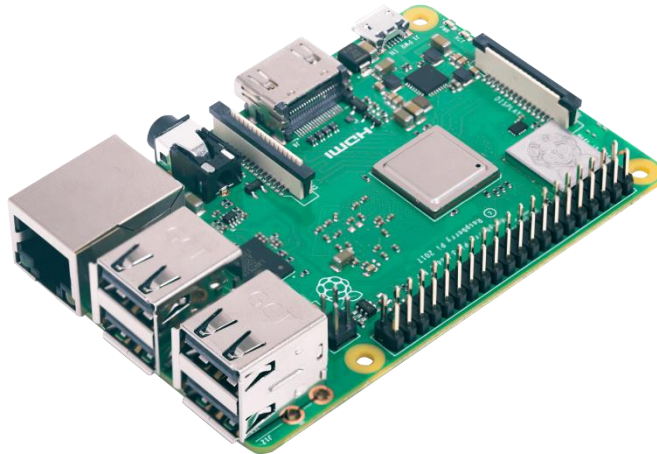


Fig 3.2.Raspberry Pi 3B+

- **PIR Sensor:**

A PIR or a Passive Infrared Sensor can be used to detect presence of human beings in its proximity. A PIR sensor detects the infrared light radiated by a warm object.

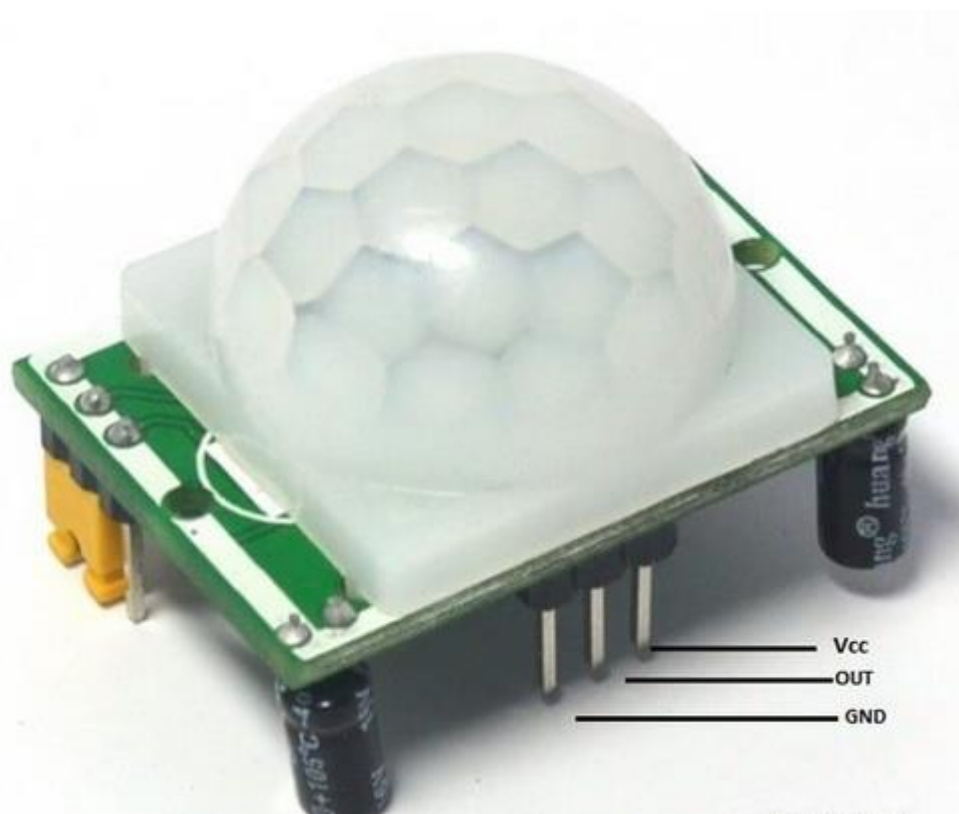


Fig 3.3.PIR Sensor

- **Single Channel Relay:**

A relay is an electrically operated device. It has a control system and (also called input circuit or input contactor) and controlled system (also called output circuit or output contactor). It is frequently used in automatic control circuit. To put it simply, it is an automatic switch to controlling a high-current circuit with a low-current signal.



Fig 3.4 Single Channel Relay

- **USB Web Camera** - A Web Camera module is interfaced with the Raspberry Pi through one of USB ports in Raspberry Pi 3. The OpenCV library is used to provide the functionality to work with this standard webcam. When "captured" by the computer, the video stream may be saved, viewed or sent on to other networks via systems such as the internet, and emailed as an attachment.



Fig. 3.5 USB Web Camera



## **Software:**

### **Raspbian OS:**

Raspbian is a free operating system based on Debian optimized for the Raspberry Pi hardware. An operating system is the set of basic programs and utilities that make your Raspberry Pi run.

### **Putty:**

PuTTY is a free and open-source terminal emulator, serial console and network file transfer application. It supports several network protocols, including SCP, SSH, Telnet, login, and raw socket connection. It can also connect to a serial port

### **VNC Viewer:**

In computing, Virtual Network Computing (VNC) is a graphical desktop sharing system that uses the Remote Frame Buffer protocol (RFB) to remotely control another computer. It transmits the keyboard and mouse events from one computer to another, relaying the graphical screen updates back in the other direction, over a network.

### **Advanced ip scanner:**

Advanced IP Scanner. Reliable and free network scanner to analyse LAN. The program shows all network devices, gives you access to shared folders, provides remote control of computers (via RDP and R admin), and can even remotely switch computers off. It is easy to use and runs as a portable edition.

### **Sd formatter:**

SD Card Formatter is a program that provides quick and easy access to all memory card formats like SD, SDHC and SDXC, and has been designed so that you can get rid of all the content stored on your SD card in one go.

### **Net Beans IDE 8.1:**

NetBeans IDE 8.1 provides out-of-the-box code analyzers and editors for working with the latest Java 8 technologies--Java SE 8, Java SE Embedded 8, and Java ME Embedded 8.

### **Java ME SDK 8.3:**

Java Platform, Micro Edition or Java ME is a computing platform for development and deployment of portable code for embedded and mobile devices (micro-controllers, sensors, gateways, mobile phones, personal digital assistants, TV set-top boxes, printers).[1]Java ME was formerly known as Java 2 Platform, Micro Edition or J2ME.

## Chapter 4

### Implementation

#### 4. Implementation

##### 4.1 Circuit Diagram

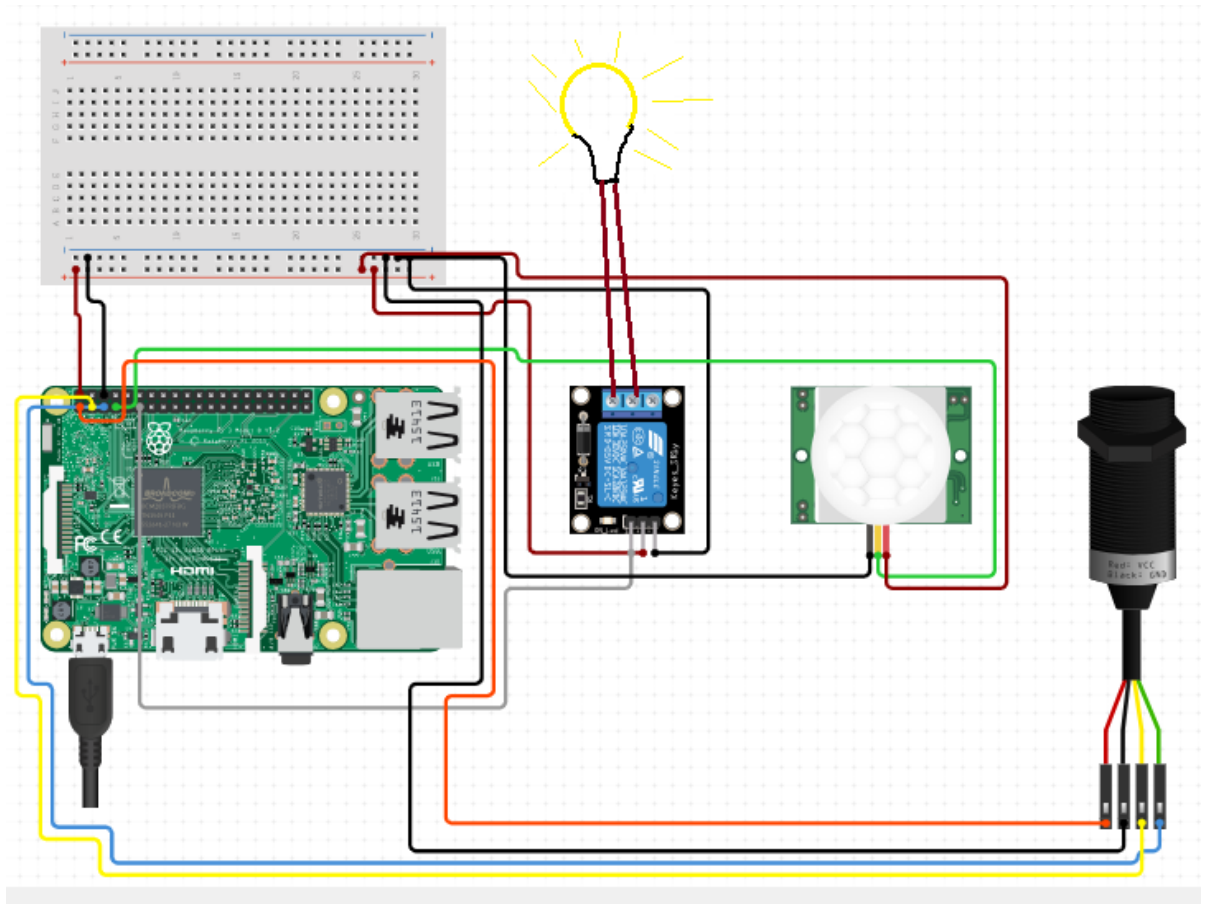


Fig 4.1 Connection of Raspberry pi with Relay,Sensors

#### 4.2 Setup:

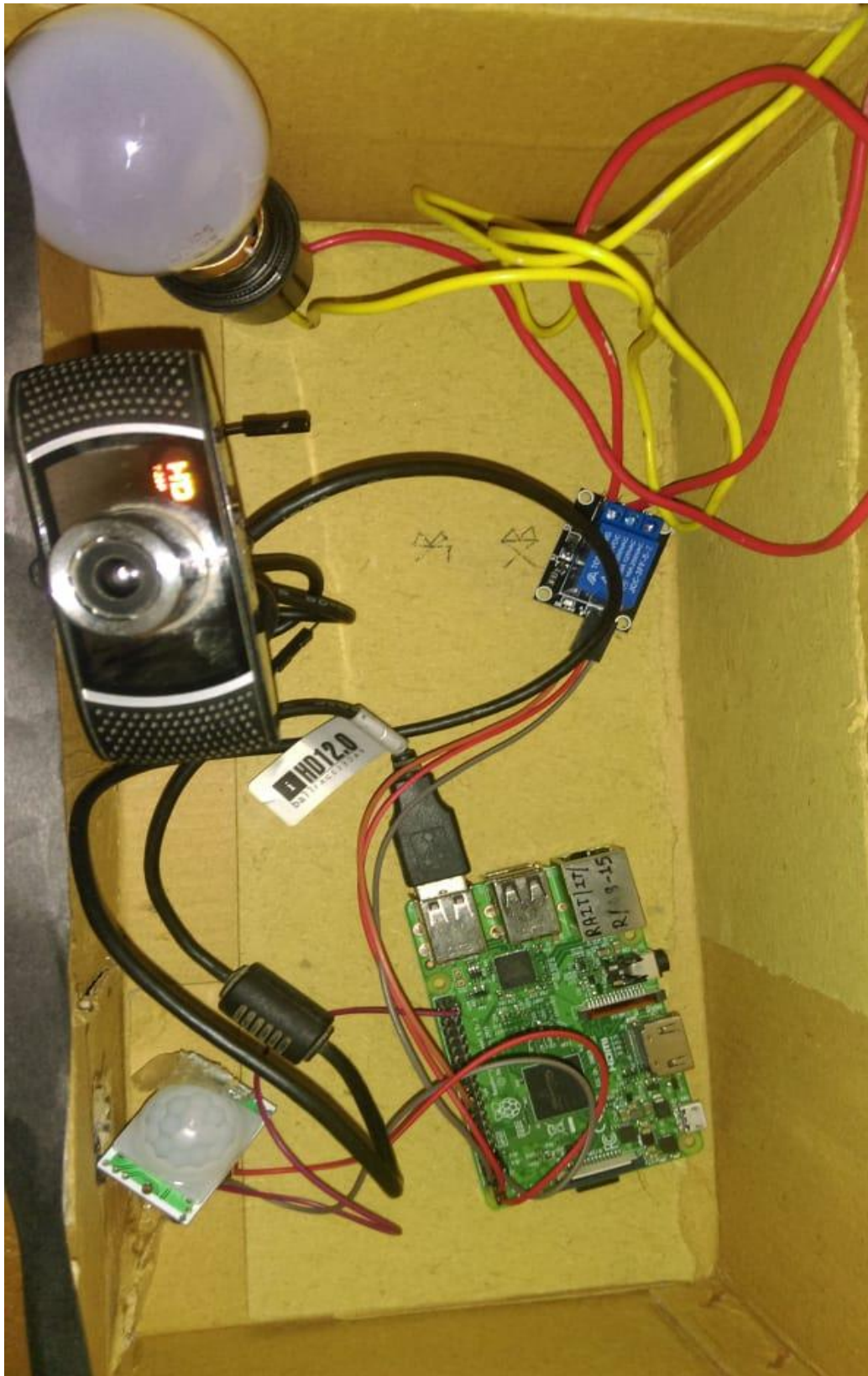


Fig 4.2 Setup

### 4.3 Steps to install essential packages on Raspberry pi:

```
$ sudo apt-get update
```

```
$ sudo apt-get upgrade
```

Now, we have to install OpenCV library. There are lots of methods available to install OpenCV. The simplest method to install the OpenCV on Linux is given in the OpenCV website or type git clone <https://github.com/opencv/opencv.git> Open the Linux terminal on the Raspbian and execute the following commands –

1. First install the compiler by running the following command –

```
$ sudo apt-get install build-essential
```

2. Install the required packages by running the following command –

```
$ sudo apt-get install cmake git libgtk2.0-dev pkg-config libavcodec-dev libavformat-dev libswscale-dev
```

3. Next install optional packages by running the following command –

```
$ sudo apt-get install python-dev python-numpy libtbb2 libtbb-dev libjpeg-dev libpng-dev libtiff-dev libjasper-dev libdc1394-22-dev
```

4. Install the OpenCV to any directory by running the following commands –

```
$ cd <home>
```

```
$ git clone https://github.com/opencv/opencv.git
```

5. Next, create a temporary directory (<c>) where the generated make files, project files, object files and output binaries should be saved. This can be done by running the following commands

```
$ cd opencv
```

```
$ mkdir build
```

```
$ cd build
```

```
$ cmake-
```

```
DCMAKE_BUILD_TYPE=RELEASEDCMAKE_INSTALL_PREFIX=/home/opencv ..
```

```
$ sudo apt-get install python-opencv
```

```
sudo reboot
```

The E-mail System is implemented on the Raspberry pi development board in Linux environment, which supports SMTP (Simple Mail Transfer Protocol), TCP/IP and HTTP. The web server Flash File System supports dynamically generated files that can include output data from hardware resources. This type of file is called an embedded server page (ESP). When the PIR sensor detects motion at the entrance, its digital output is set to HIGH. In the python script when the GPIO 4 goes high, the webcam connected to the raspberry pi takes the snap of the entrance and send the image attachments to the mail.

#### 4.4 Code:

# Python code:

```
import RPi.GPIO as GPIO
```

```
import time
```

```
import numpy as np
```

```
import cv2
```

```
from datetime import datetime
```

```
import os
```

```
import smtplib
```

```
from email.MIMEMultipart import MIMEMultipart
```

```
from email.MIMEBase import MIMEBase
```

```
from email.MIMEText import MIMEText
```

```
from email import Encoders
```

```
gmail_user = "teamiot3@gmail.com" #Sender email address
```

```
gmail_pwd = "team3@iot" #Sender email password
```

```
to = "dasrahulkumar89@gmail.com" #Receiver email address
```

```
subject = "Security Alert"
```

```
text = "Hello,Sorry to disturb you, here is some activity in your home. See the attached  
picture."
```

```
sensor = 4
```

```
GPIO.setmode(GPIO.BCM)
```

```
GPIO.setup(sensor, GPIO.IN, GPIO.PUD_DOWN)
```

```
previous_state = False
```

```
current_state = False
```

```
while True:
```

```
    previous_state = current_state
```

```
    current_state = GPIO.input(sensor)
```

```

if current_state != previous_state

    new_state = "HIGH" if current_state else "LOW"

    print("GPIO pin %s is %s" % (sensor, new_state))

    if current_state:

        cap = cv2.VideoCapture(0)

        ret, frame = cap.read()

        cap = cv2.VideoCapture(0)

        print "Saving Photo"

        picname = datetime.now().strftime("%y-%m-%d-%H-%M")

        picname = picname+'.jpg'

        cv2.imwrite(picname, frame)

        print "Sending email"

        attach = picname

        msg = MIMEMultipart()

        msg['From'] = gmail_user

        msg['To'] = to

        msg['Subject'] = subject

        msg.attach(MIMEText(text))

        part = MIMEBase('application', 'octet-stream')

        part.set_payload(open(attach, 'rb').read())

        Encoders.encode_base64(part)

        part.add_header('Content-Disposition', 'attachment; filename="%s"' %
            os.path.basename(attach))

        msg.attach(part)

        mailServer = smtplib.SMTP("smtp.gmail.com", 587)

        mailServer.ehlo()

        mailServer.starttls()

```

```
mailServer.ehlo()

mailServer.login(gmail_user, gmail_pwd)

mailServer.sendmail(gmail_user, to, msg.as_string())

mailServer.close()

print "Email Sent"

os.remove(picname)
```

# Java Code:

```
package pirrelay;

import javax.microedition.midlet.MIDlet;
import jdk.dio.DeviceManager;
import jdk.dio.gpio.GPIOPin;

public class Pirrelay extends MIDlet {

    private static final int sensor=1;

    private static final int relay=4;

    private GPIOPin sensor1;

    private GPIOPin relay1;

    @Override

    public void startApp() {

        try{

            sensor1=DeviceManager.open(sensor);

            relay1=DeviceManager.open(buzzer);

            while(true)

            {

                if(sensor1.getValue())

                {

                    System.out.println("Some intruder came into your house.");
```

```

relay1.setValue(false);

Thread.sleep(1000);

while(sensor1.getValue()){ }

relay1.setValue(true);

}}

}

catch(Exception ex){System.out.println(ex.getMessage());

}

}

```

@Override

```
public void destroyApp(boolean unconditional){ } }
```

Pushover app code:

```
import time
```

```
import RPi.GPIO as GPIO
```

```
import httplib, urllib
```

```
# setup GPIO using Broadcom SOC channel numbering
```

```
GPIO.setmode(GPIO.BCM)
```

```
# define the GPIO port you will use for the motion detector
```

```
PIR_SENSOR = 4
```

```
# number of seconds to delay between alarms
```

```
DELAY = 10
```

```
# set to pull-up (normally closed position for a PIR sensor dry contact)
```

```
GPIO.setup(PIR_SENSOR, GPIO.IN, pull_up_down=GPIO.PUD_UP)
```



```

# Pushover API setup

PUSH_TOKEN = "" # API Token/Key

PUSH_USER = "" # Your User Key

PUSH_MSG = "Motion detected!" # Push Message you want sent


# This function sends the push message using Pushover.

# Pass in the message that you want sent

def sendPush( msg ):

    conn = httplib.HTTPSConnection("api.pushover.net:443")

    conn.request("POST", "/1/messages.json",

        urllib.urlencode({

            "token": PUSH_TOKEN,

            "user": PUSH_USER,

            "message": msg,

        }), { "Content-type": "application/x-www-form-urlencoded" })


    conn.getresponse()

    return


try:

    # setup an indefinite loop that looks for the PIR sensor to be triggered

    while True:

        # motion is detected

        GPIO.wait_for_edge(PIR_SENSOR, GPIO.RISING)

        # print and push message

        print(PUSH_MSG)

        sendPush(PUSH_MSG)

```

```
time.sleep(DELAY)
```

```
except KeyboardInterrupt:
```

```
    # cleanup GPIOs on keyboard exit
```

```
    GPIO.cleanup()
```

```
# cleanup GPIOs when program exits
```

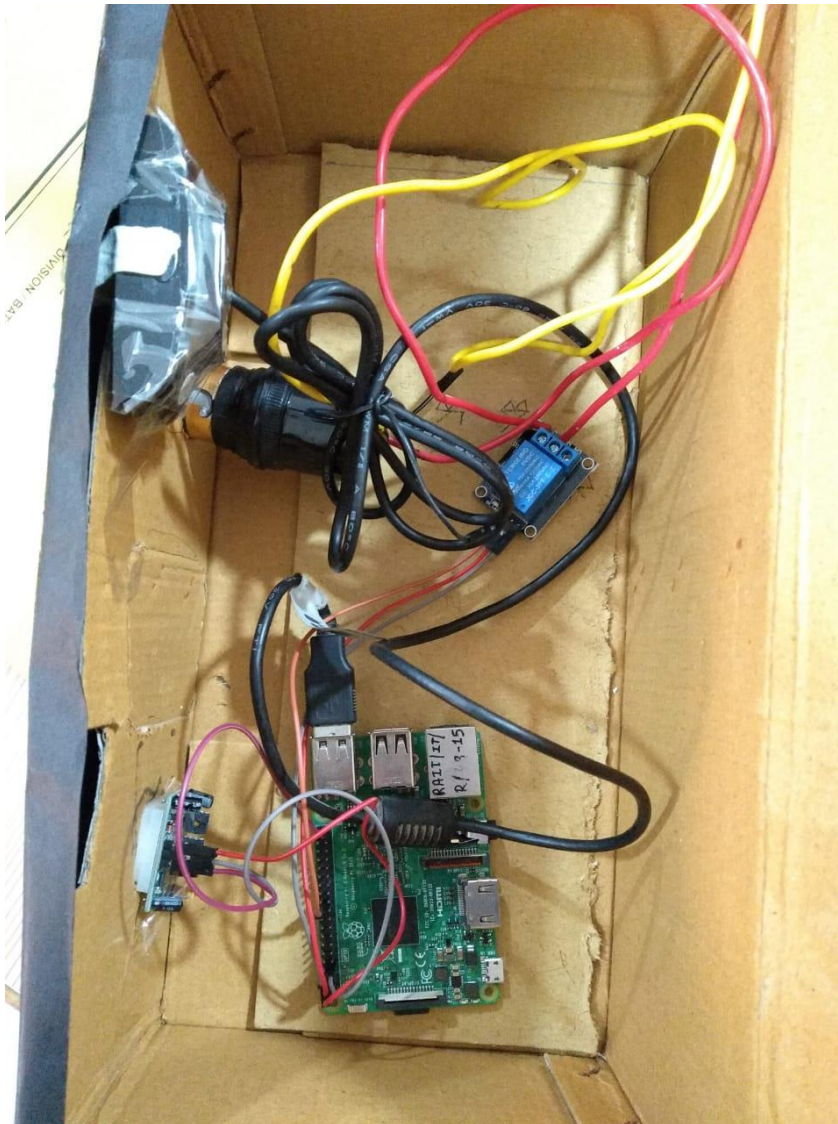
```
GPIO.cleanup()
```

## Chapter 5

### Result

#### 5.1 Images:

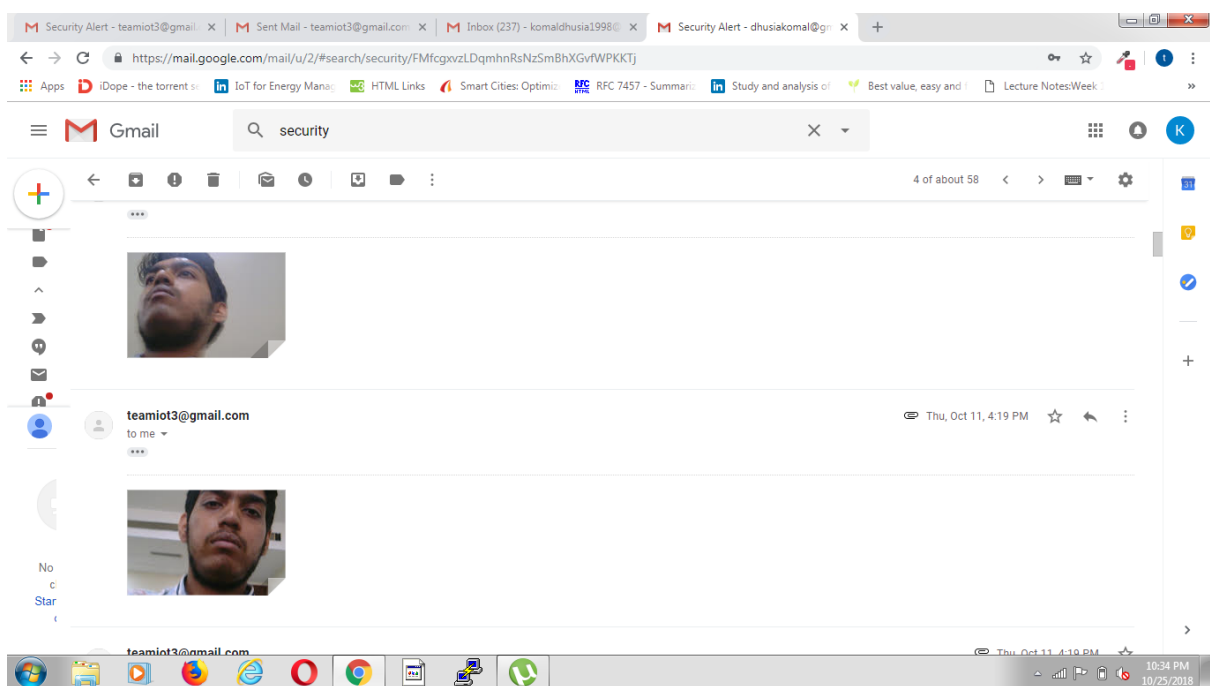
##### 5.1.1 Actual Circuit image:



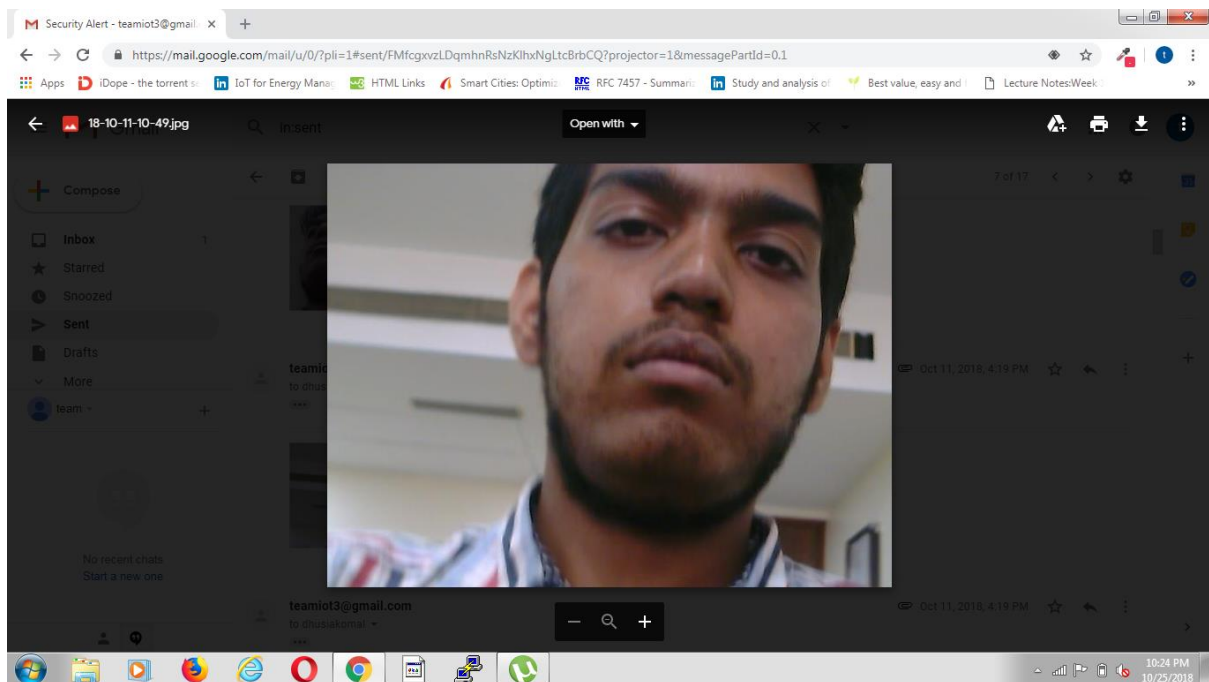
### 5.1.2 Model:



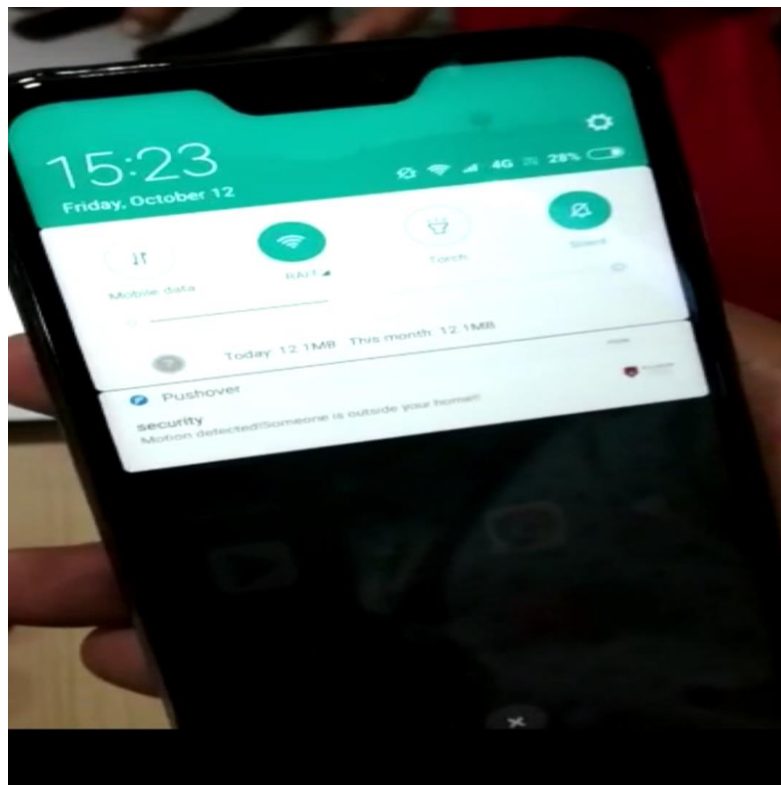
### 5.1.2 Email Notification:



### 5.1.3 Email Attachment:



### 5.1.4: Push Notification on Mobile Phone:



## **Chapter 6**

### **Conclusion and Future scope**

#### **6. Conclusion and Future scope**

##### **6.1 Conclusion:**

Suitability of IoT, in the context of the home security means integration of all devices and their monitoring, controlling and alerting in ways not possible before. In order to address the issues of flexibility, low cost home security and monitoring system using Raspberry Pi, in this project, we have proposed An Advanced Internet of Thing based security Alert System for Smart Home, to detect an intruder or any unusual event at home, when nobody is available there. With improving the security in smart home, the result of proposed approach has met our expectations, where attached sensors are properly working in real-time scenarios. Here system starts once without user input and further it automatically send email to owner on any intrusion detection and owner take necessity action. It can be concluded that the proposed system present the basic level of home security and remote monitoring while the required objectives of home security system have been achieved. This low-cost home security system has minimum delay during process of email alert. This project also confirms the advantage of Raspberry Pi as the flexibility in-terms of cost and broad probability of its usage. Preliminary analyses have shown encouraging results

##### **6.2 Future scope:**

. In future, we will apply fog computing in our proposed system in order to minimize the propagation delay and to enhance the security level of smart home. We can also include image processing to detect and recognize whether the intruder is a known person or someone unknown.

## Chapter 7

### References

#### 7. References:

- [1] A. Zanella, N. Bui, A. Castellani, L. Vangelista, and M. Zorzi, “Internet of Things for smart cities,” IEEE Internet of Things Journal, vol. 1, no. 1, pp. 22–32, Feb. 2014.
- [2]Mr.Krunal Solanki,Mr.Bharat Chaudhary,”Wireless Real time video surveillance system based on embedded web server and ARM9”,International Journal for Advance Research in Engineering and Technology,vol.2,Issue IV,ISSN 2320-6802,April 2014.
- [3]Sanjana Prasad, P.Mahalakshmi, A.Jhon Clement Sunder, R.Swathi, “Smart Surveillance Monitoring System using Raspberry Pi and PIR Sensor”, International Journal of Computer Science and Information Technilogies, vol.5 (6), ISSN: 0975-9646, 2014.
- [4]G.Senthil kumar, V.Sathish Kumar, K.GopalaKrishnan,”Embedded Inage Capure System using Raspberry Pi System”, International Journal of Emerging Trends & Technology and Engineering, Vol.3, issue 3, July 2014.
- [5] Rohit Ragmahale. A Literature Survey on Smart Home Automation Security, International Journal of Advance Research, Ideas and Innovations in Technology, INTERNATIONAL JOURNAL OF ADVANCE RESEARCH, IDEAS AND INNOVATIONS IN TECHNOLOGY (ISSN: 2454-132X)