Project Report

Smart Surveillance System

SONU SAHA (BT/IT/16-54)

1. INTRODUCTION

Internet of Things (IoT) is an interconnection of physical objects and other devices embedded with electronics, software, network connectivity and application-specific sensors and actuators that enable the objects to collect and exchange data. It is an important topic in the technology industry and its applications have gained popularity in recent years and by 2020, Internet-connected devices are expected to number between 26 billion and 50 billion[3].

A smart surveillance security camera system can have many benefits such as building security, remote monitoring or deter trespassers from attempting to gain access to the facility. The problem with the traditional surveillance system or CCTV camera is costly because of the use of many expensive components like a computer, cables, hard disk with higher capacity for the storage of video or images. It reserves too much space for continuous recording and requires manual verification for detection of an unauthorized person or activity. In comparison, Raspberry Pi is much cheaper with low power consumption feature which makes it able to solve many of the issues of cost that may discourage a person from investing in remote surveillance technology.

2. PROJECT GOAL

In this project, I am going to be using a Raspberry Pi 3 Model B+ (Fig. 1), which is a low cost, a credit-card sized computer that plugs into a computer monitor or can be accessed remotely from another PC using Wi-Fi. And the camera module rev 1.3 (Fig. 2) can be used to take high-quality video, as well as still photographs. It's able to deliver clear 5MP resolution image or 1080p HD video. Also, a Passive Infrared Sensor (PIR) (Fig. 3), which allow us to sense motion, almost always used to detect whether a human has moved in or out of the sensors range.

Fig. 1 : Raspberry Pi 3 B+ Board

Fig. 2 : Camera Module rev 1.3

Fig. 3 : PIR Sensor with 3-pin

The goal of the project is to develop a low-cost remote surveillance system using Raspberry Pi platform that provides the user with a much better way for monitoring. As the user should not be bothered by a constant alert in case of any motion detected, it additionally performs an image processing on the images in the cloud and then sends an alert using a text message to the user.

3. PROJECT OVERVIEW

In this prototype, the detection of the motion and capturing of the image is done on the Raspberry Pi and the image processing is done on the PC which acts as a local server. The basic working of the complete prototype can be described in the following steps:

The Raspberry Pi's functioning:

1. Sense motion using PIR sensor.
2. If motion is detected:
   I. Capture images using the camera module.
   II. Zip the folder that contain the images.
   III. Send the zipped file to cloud (in this case Dropbox)
3. If motion not detected, repeat Step 1 after some interval of time.

The next time it detects motion, the previously captured images and zipped file are deleted, thus clearing up the space on Raspberry Pi.

The program running on the server:

1. Check if new file has arrived in the Dropbox.
2. If new file is detected:
   I. Download the file.
   II. Unzip the file.
   III. Apply image processing on all the images.
   IV. Check if human detected.
      i. If yes, send a notification to the user.
      ii. Save a copy of the image.
   V. Delete the file from the Dropbox.
3. If new file is not detected, check the Dropbox again after some interval of time.

ADVANTAGE

Some of the advantages of this prototype are as follows:

1. It is a simple circuit which is cheap and easy to implement.
2. It does not need high memory storage for storing images, as old images are deleted automatically after the tasks are completed.
3. Complex and improved image processing algorithm can be implemented since it will run on the server, it wouldn't consume the Raspberry Pi resource.

3.1 The Flow Chart Diagram General System

The methodology followed in the project can be described by the following diagram:

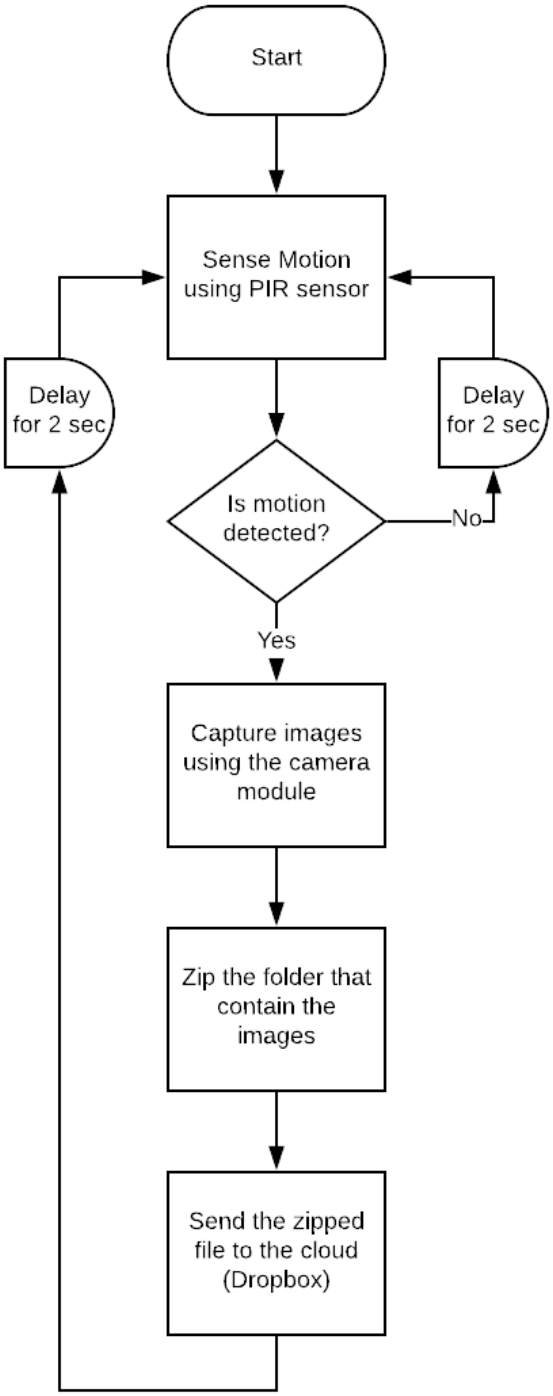

Fig. 4 : Flow on Raspberry Pi

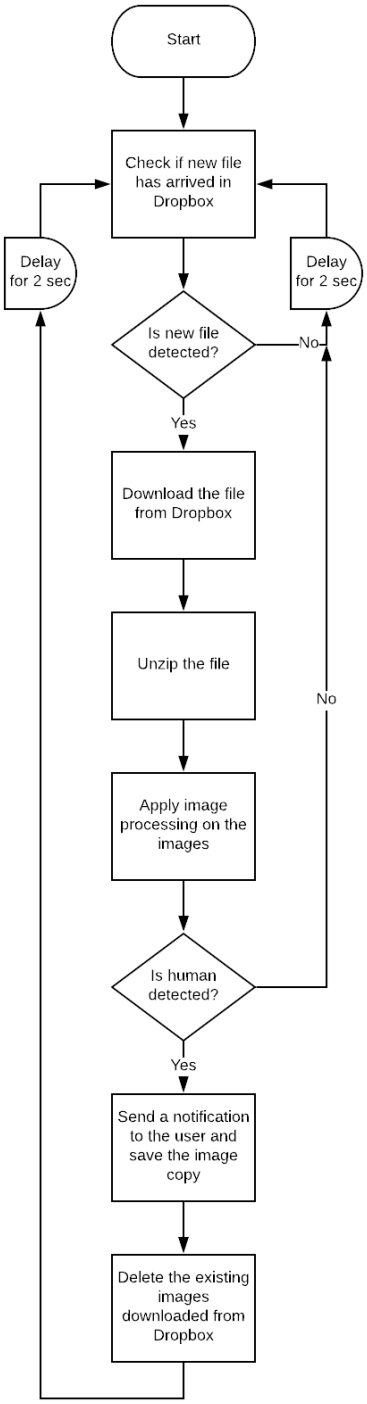

Fig. 5 : Flow on Server Machine

4. PROJECT IMPLEMENTATION

4.1 Hardware equipment list

To implement the prototype for this project I have used the following equipment :

1. The Raspberry Pi 3 Model B+ Board, 1 GB RAM.
2. The Raspberry Pi camera module rev 1.3
3. PIR Sensor
4. SD Card 32 GB.
5. Connector cables - female to female jumper wires.
6. Supply power adapter 5V and a USB cable.
7. A source of Wi-Fi (in my case, it's the mobile phone's hotspot).

Brief description of some of the Hardware equipment:

1. **The Raspberry Pi 3 Model B+ Board.**

In this surveillance system, the Pi board is the most important component, which acts as the brain of the project. It includes an Ethernet port, four USB 2.0 ports, audio input/output, HDMI output and general purpose input/output (GPIO) pins. The row of GPIO (Fig. 6) is one the powerful feature on Pi which are a physical interface between the Pi and the outside world. Of the 40 pins, 26 are GPIO pins and the others are power of ground pins plus two ID EEPROM pins[4].

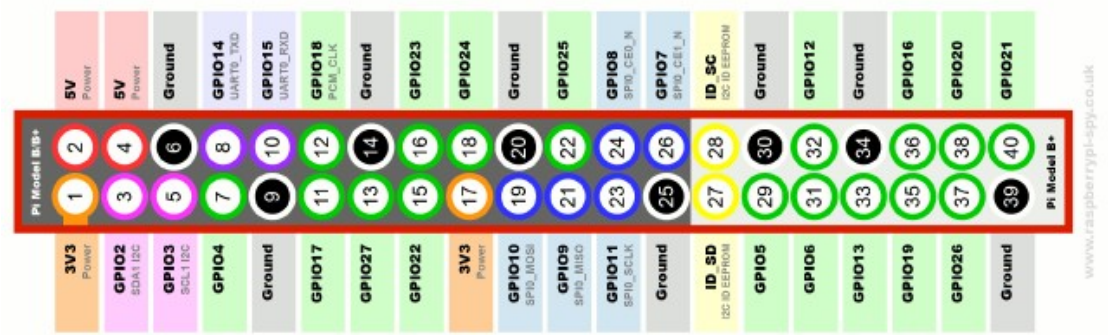

Fig. 6 : GPIO on Raspberry Pi 3 Model B+

2. **The Raspberry Pi Camera module rev 1.3.**

The camera connects to Raspberry Pi by way of a short flexible ribbon cable to the port available on the board. The camera connects to the BCM2837 processor on the Pi via the CSI bus, a higher bandwidth link which carries pixel data from the camera back to the processor. This bus travels along the ribbon cable that attaches the camera board to the Pi.

3. **PIR Sensor.**

The PIR Sensor consists of 3 pins, ground, signal, and power at the side or bottom. It is used to detect the movement of human being within a certain range(approximately have an average value of 10m) of the sensor[5].

The PIR sensor can be interfaced with the Pi's GPIO as show in the following diagram (Fig. 7):

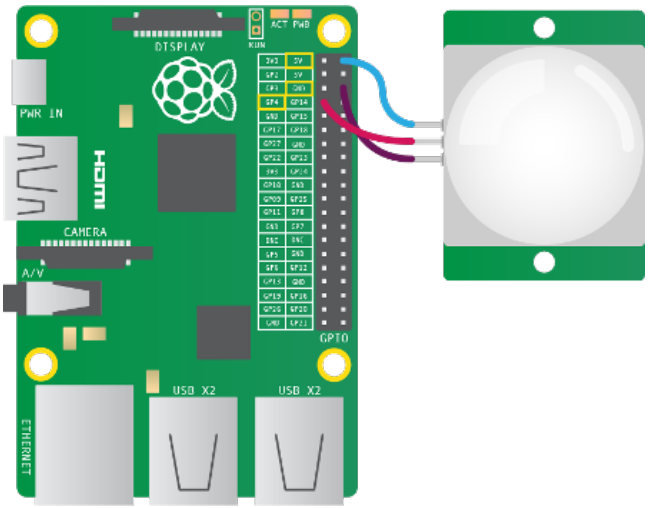

Fig. 7 : PIR Sensor interfaced with Raspberry Pi

i. Connect the one labeled VCC on the PIR sensor to the 5V pin on the Raspberry Pi's GPIO. This provides power to the PIR sensor.
ii. Connect the one labeled GND to any Ground pin on the Raspberry Pi. This completes the circuit.
iii. Connect the one labeled OUT to any of the GPIO pins. This pin will output a voltage when motion is detected, that can be received by the Raspberry Pi.

4. **SD Card.**

The external SD card is used to install the Raspbian OS (the OS on Raspberry Pi) on it and it acts as the storage device for the Raspberry Pi.

5. **Female to female jumper wires.**

The jumper wires (Fig. 8) are used as connecting cables for the interfacing of the PIR sensor with the Raspberry Pi.

Fig. 8: Female-to-female jumper wires

4.2 The Hardware Prototype

The setup for the prototype with all its components is shown in the following diagram (Fig. 9) :

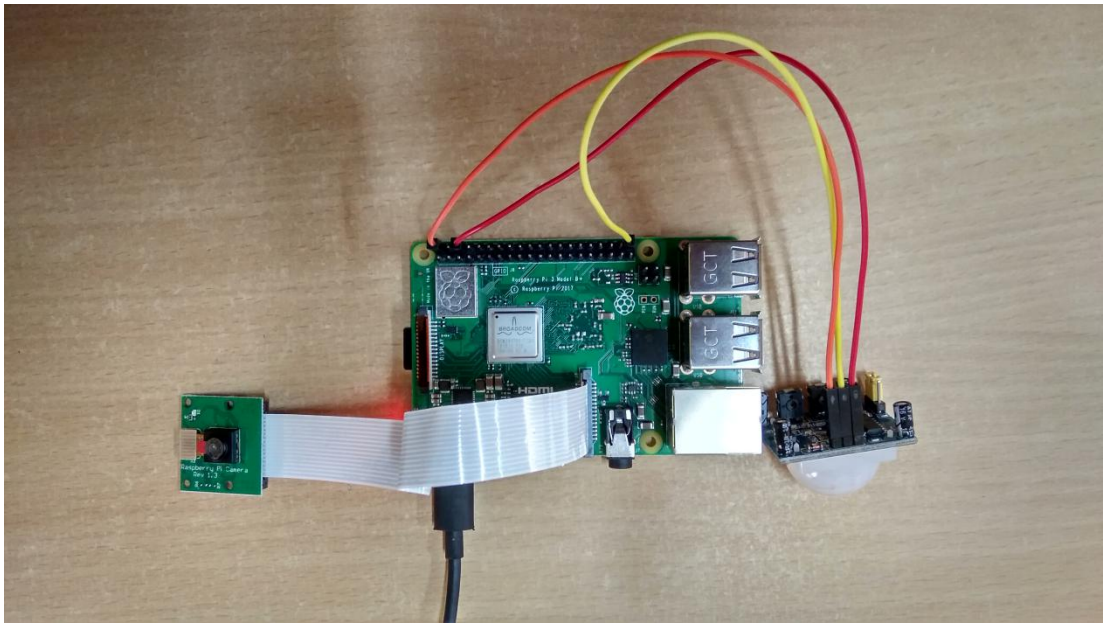

Fig. 9 : The Prototype : Raspberry Pi, Camera and PIR Sensor

4.3 Software Configuration

After the physical connection is properly setup, we must configure and implement the program on the Raspberry Pi and on the server machine.

Configuration on the Raspberry Pi:

1. Enable the SSH and Camera on the Raspberry Pi from the *raspi-config* menu. Secure Shell (SSH) is a feature of Linux that allows us to effectively open a terminal session on our Raspberry Pi from the command line of any host computer and enabling camera allows us to use the camera module.

2. Create a python file on your Raspbian which would perform the task to sense motion, capture the image, and then send the image to Dropbox.

The complete program can be broken down into 3 parts:

Part 1: Sense motion using PIR sensor and if motion is detected, capture images.

Part 2: Zip the folder that contains the captured images.

Part 3: Upload the zipped file to Dropbox using the API key of the app that is previously created on the Dropbox.

3. Execute the python file, and the Raspberry Pi is all set to go.

Configuration on the Server machine:

There are 3 files on the server machine. A Caffe prototxt file which is a deep learning framework, a pre-trained model and a python file which contains the program to perform all the required tasks.

The complete program can be broken down into 5 parts:

Part 1: Check for new file on Dropbox and if new file is found, download it using the API key of the same app where the file was uploaded from Raspberry Pi.

Part 2: Unzip the file that is downloaded from the dropbox.

Part 3: Apply image processing on the images using deep learning based object detection with OpenCV.

Part 4: If human is detected in the image processing, send a notification to the user and save a copy of the image.

Part 5: Delete the other files that were downloaded from Dropbox.

4.4 Experimental Result

The experiment for this project was carried out in Raspberry Pi and a laptop which has a Pentium Dual core processor with 4GB RAM. The experiment was checked thoroughly by repeating the motion of a person entering a room to see if each time a notification is sent or not, and the system worked in the intended way and flawlessly. To test the endurance of the hardware, the setup was left turned on for a couple of hours and tested afterwards. The processing of the Raspberry Pi was monitored for testing purpose and is shown in the Fig. 14, the images captured by the camera is shown in the Fig. 13, the processing of the program on server machine was monitored and is shown in the Fig. 15. The notification sent to the user is shown in Fig. 16.

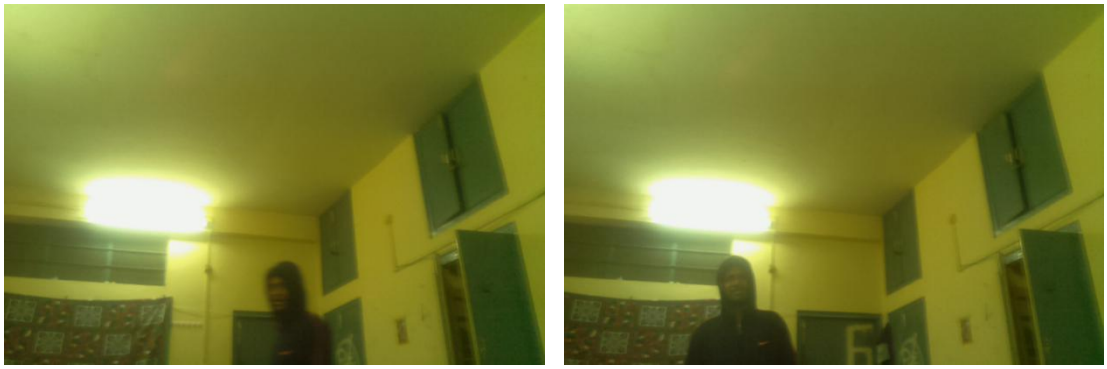

Fig. 13 : Images captured by the Camera when motion was detected

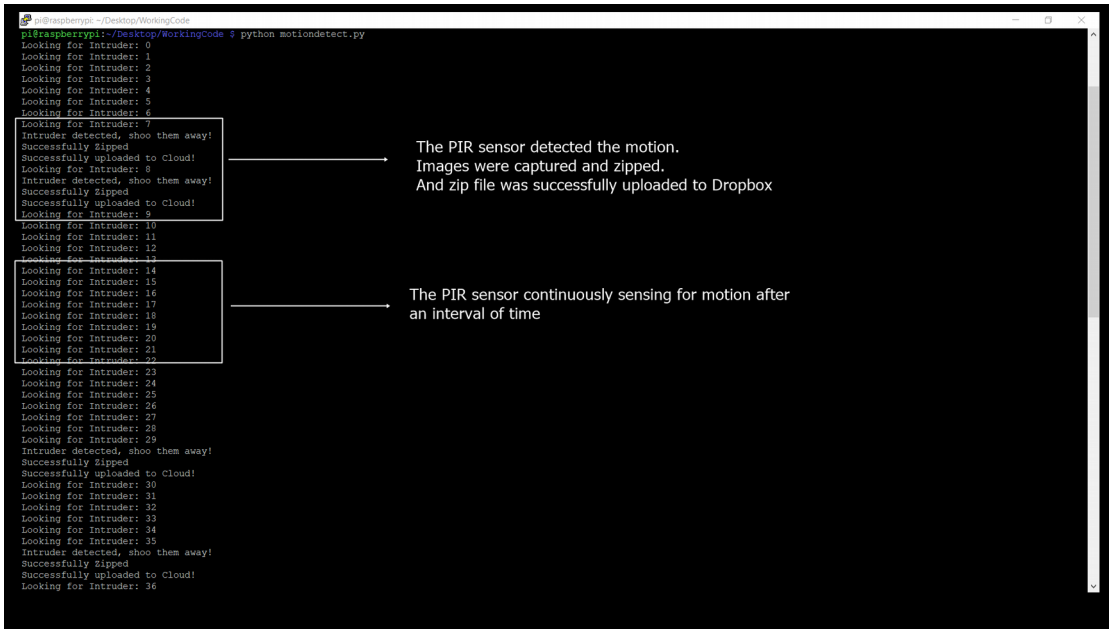

Fig. 14 : Program running on Raspberry Pi

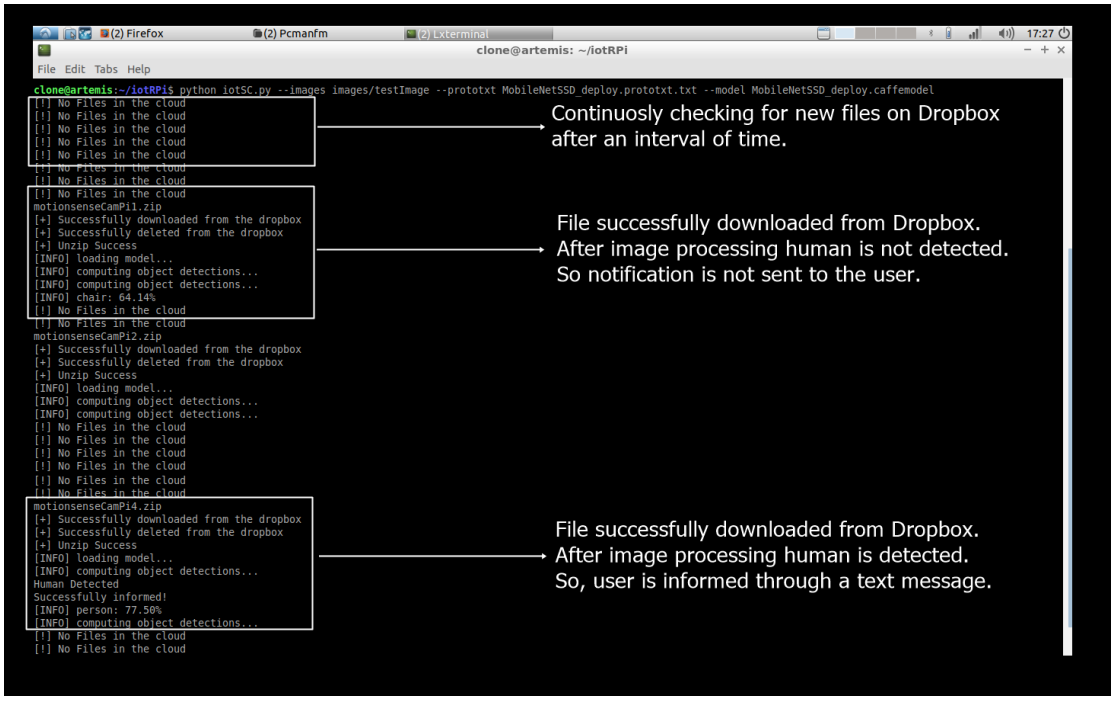

Fig. 15 : Program running on server machine

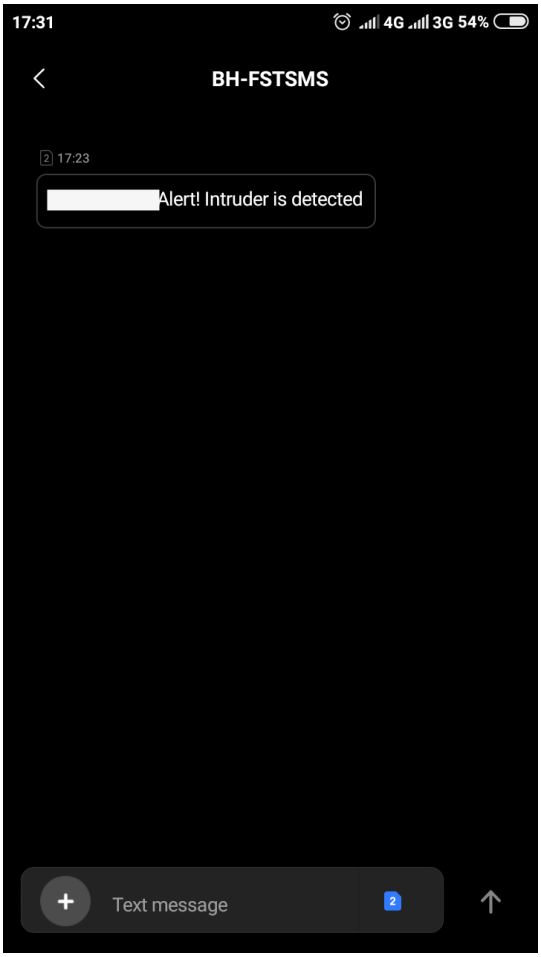

Fig. 16 : Notification on the user's phone through text message

5. CONCLUSION AND FUTURE SCOPE

The sensors placed at the entry of the room senses motion, then activates the camera to capture and the image is uploaded to the cloud. The images are verified for human presence and a notification is sent to the user irrespective of whether the phone has an active internet connection or is not a smart phone. This was the main objective of the project, which is the user should not be bother by constant notification on motion detection but also feels safe and not worry about any intrusion or break-ins when he is away from home. This setup can also be used in commercial offices where some areas are restricted for certain personnel trying to access such an area. Therefore, the extension and application of such a system is only limited only by the imagination.

The developed system can be easily upgraded to add extra safety features such as automated lock, alarm, phone call to the nearest police station etc. for increased safety. The system can also be further be developed by adding some security pin so that the authorized users can enter the pin, so that the system will be disabled for the required interval of time.

6. REFERENCES

[1] M. N. N. A. Asghar, M.H., “Principle application and vision in internet of things (iot),” in Communication Technologies (GCCT), 2015 Global Conference on, may 2015.

[2] Raj G Anvekar, Dr. Rajeshwari M Banakar “IoT Application Development: Home Security System” in 2017 IEEE International Conference on Technological Innovations in ICT For Agriculture and Rural Development

[3] Raymond James & Associates, “ The Internet of Things - A Study in Hype, Reality, Disruption, and Growth”

[4] Raspberry Pi Hardware - Raspberry Pi Documentation online at https://www.raspberrypi.org/documentation/hardware/

[5] PIR Sensor Circuit and Working with Applications - EIProCus, online at https://www.elprocus.com/pir-sensor-circuit-with-working/