Reduction Of Storage Space In CCTV And IP Cameras using Internet Of Things

A PROJECT REPORT

Submitted by

SARAVANAN M	**(211416104253)**
SHARATH RA	**(211416104259)**
SURYA J	**(211416104278)**

in partial fulfillment for the award of the degree

of

BACHELOR OF ENGINEERING

IN

COMPUTER SCIENCE AND ENGINEERING

PANIMALAR ENGINEERING COLLEGE, CHENNAI-600123.

ANNA UNIVERSITY: CHENNAI 600 025

APRIL 2020

ANNA UNIVERSITY: CHENNAI 600 025

BONAFIDE CERTIFICATE

Certified that this project report **“REDUCTION OF STORAGE SPACE IN CCTV AND IP CAMERAS USING INTERNET OF THINGS”** is the bonafide work of “**SARAVANAN M (211416104253), SHARATH RA (211416104259)**, **SURYA J (211416104278)**” who carried out the project work under my supervision

SIGNATURE

Dr. S. MURUGAVALLI, M.E., Ph.D.,
HEAD OF THE DEPARTMENT

Computer Science and Engineering,
Panimalar Engineering College,
Varadharajapuram, Nasarathpettai,
Poonamallee,
Chennai-600 123.

SIGNATURE

Mrs. D.ANURADHA, M.E.,
SUPERVISOR
ASSOCIATE PROFESSOR

Computer Science and Engineering,
Panimalar Engineering College,
Varadharajapuram, Nasarathpettai,
Poonamallee,
Chennai-600 123.

Certified that the above-mentioned students were examined in Anna University project viva-voce held on--------------------

INTERNAL EXAMINER **EXTERNAL EXAMINER**

ACKNOWLEDGEMENT

We express our deep gratitude to our respected Secretary and Correspondent **Dr.P.CHINNADURAI, M.A., Ph.D.** for his kind words and enthusiastic motivation, which inspired us a lot in completing this project.

We would like to extend our heartfelt and sincere thanks to our Directors **Tmt. C.VIJAYARAJESWARI**, **Thiru. C.SAKTHIKUMAR, M.E.,** and **Tmt. SARANYASREE SAKTHIKUMAR B.E., M.B.A.,** for providing us with the necessary facilities for completion of this project.

We also express our gratitude to our Principal **Dr.K. Mani, M.E., Ph.D.** for his timely concern and encouragement provided to us throughout the course.

We thank the HOD of CSE Department, **Dr. S. MURUGAVALLI, M.E., Ph.D.,** for the support extended throughout the project.

We would like to thank my **Project Guide Mrs. D.Anuradha M.E., Associate Professor** and all the faculty members of the Department of CSE for their advice and suggestions for the successful completion of the project.

SARAVANAN M

SHARATH RA

SURYA J

ABSTRACT

Smart camera has been used for surveillance system for many places. Internet of things is the communication of anything with any other thing, the communication mainly transferring of useable data, for example a sensor in a room to monitor and control the temperature. This paper aims to describe a security alarm system using Internet of things which helps to monitor and get alarms when motion is detected and sends photos and videos to a cloud server. The photos and videos are sent directly to a cloud server, when the cloud is not available then the data is stored locally on the Raspberry Pi and sent when the connection resumes. Therefore, advantages like these make this application ideal for monitoring homes in absence. The purpose of the project is to make a system, which would detect and take snapshots and videos of the motion when detected and upload to an external server. The major use of the 'Motion Detection' is at homes, buildings and also for surveillance for security for example of server rooms.

TABLE OF CONTENT

LIST OF FIGURES

LIST OF ABBREVATIONS

IoT	Internet of Things
GSM	Global System for Mobile Communication
SIM	Subscriber Identity Module
IP	Internet Protocol
WSN	Wireless Sensor Networks
ADT	Android Development Tool

CHAPTER 1

INTRODUCTION

CHAPTER 1

INTRODUCTION

1.1 AIM:

The main aim of this project is smart home Surveillance based on Internet of Things.

1.2 SYNOPSIS:

During the past few years, in the area of wireless communications and networking, a novel paradigm named the Internet of Things (IoT) which was first introduced by Kevin Ashton in the year 1998, has gained increasingly more attention in the academia and industry. By embedding shortrange mobile transceivers into a wide array of additional gadgets and everyday items, enabling new forms of communication between people and things, and between things themselves, IoT would add a new dimension to the world of information and communication. Unquestionably, the main strength of the IoT vision is the high impact it will have on several aspects of every-day life and behavior of potential users. From the point of view of a private user, the most obvious effects of the IoT will be visible in both industry and domestic fields. In this context, assisted living, smart homes and offices, e-health, enhanced learning is only a few examples of possible application scenarios in which the new paradigm will play a leading role in the near future. Similarly, from the perspective of business users, the most apparent consequences will be equally visible in fields such as automation and industrial manufacturing, logistics, business process management, intelligent transportation of people and goods. Machine learning and understanding of human actions is a challenging area that has received much attention within the past years. Video Surveillance is one of the active research topics in Image Processing. Video Surveillance started with analogue CCTV systems, to gather information and to monitor people, events and activities. Existing digital video surveillance systems provide the infrastructure only to capture, store and distribute video, while leaving the task of threat detection exclusively to human operators.

Human monitoring of surveillance video is a very labor-intensive task. Detecting multiple activities in real-time video is difficult in manual analysis. Thus the Intelligent video surveillance system is emerged. The analytics software processes video flow images to automatically detect objects (peoples, equipments, vehicles) and event of interest for security purposes. Observing or analyzing a particular site for safety and business purposes is known as video surveillance. Security and crime control concerns are the motivating factors for the deployment of video surveillance cameras. Video surveillance cameras are used in shopping centres, public places, banking institutions, companies and ATM machines.

Nowadays, researches experience continuous growth in network surveillance. The reason being is the instability incidents that are happening all around the world. Therefore, there is a need of a smart surveillance system for intelligent monitoring that captures data in real time, transmits, processes and understands the information related to those monitored. The video data can be used as a forensic tool for after- crime inspection. Hence, these systems ensure high level of security at public places which is usually an IoT Based Smart Security Surveillance System As video cameras are available at good price in the market, hence video surveillance systems have become more popular. Video surveillance systems have wide range of applications like traffic monitoring and human activity understanding. In video surveillance system we demonstrate a system which analyses activity in the monitored space in real time, and makes the events available for generating real time alerts and content based searching in real time. Advantages of video surveillance: Availability- There was a time when the•

surveillance techniques were utilized only in shopping centers and malls. Now-a-days, you can notice closed-circuit televisions almost at any place you visit, from a small store to homes and holy places. As a result, they guarantee greater public security at a fraction of the cost. Real-time monitoring- Traditionally big . organizations have always had the benefits of video surveillance manned by security professionals. In the past times, the events captured on video were used to expose important information and work as proof after the event happened. But, modern technologies let users to check and reply to alarms immediately.

CHAPTER 2

LITERATURE SURVEY

CHAPTER 2

LITERATURE SURVEY

The past decade has seen significant advancement in the field of consumer electronics. Various ‘intelligent’ appliances such as cellular phones, air-conditioners, home security devices, home theatres, etc. are set to realize the concept of a smart home. They have given rise to a Personal Area Network in home environment, where all these appliances can be interconnected and monitored using a single controller. Busy families and individuals with physical limitation represent an attractive market for home automation and networking. A wireless home network that does not incur additional costs of wiring would be desirable. Bluetooth technology, which has emerged in late 1990s, is an ideal solution for this purpose. This paper describes an application of Bluetooth technology in home automation and networking environment. It proposes a network, which contains a remote, mobile host controller and several client modules (home appliances). The client modules communicate with the host controller through Bluetooth devices.[1]

Technology is a never ending process. To be able to design a product using the current technology that will be beneficial to the lives of others is a huge contribution to the community. This paper presents the design and implementation of a low cost but yet flexible and secure cell phone based home automation system. The design is based on a standalone Arduino BT board and the home appliances are connected to the input/ output ports of this board via relays. The communication between the cell phone and the Arduino BT board is wireless. This system is designed to be low cost and scalable allowing variety of devices to be controlled with minimum changes to its core. Password protection is being used to only allow authorized users from accessing the appliances at home.[2]

Now a days, the technology has in a grown at high speed. This paper proposes the design of IOT based home automation system using Wi-Fi. This research work presents the design and implementation of arduino based IOT based system. This is Wi-Fi based system and uses wireless technology (Wi-Fi). The system has three components: An arduino, a Wi-Fi module for signal transfer and smart phone for controlling web page server. The design is based on standalone arduino Wi-FI board and appliances are connected to this board using relays, the smart phone interact with the arduino via Wi-Fi. The aim of project is controlling the home appliances when the user is away from the place.[3]

The Internet of Things (IoT) is the network of physical objects or "things "embedded with electronics, software, sensors, and network connectivity, which enables these objects to collect and exchange data. The Internet of Things allows objects to be sensed and controlled remotely across existing network infrastructure, creating opportunities for more direct integration between the physical world and computer-based systems, and resulting in improved efficiency, accuracy and economic benefit. Each thing is uniquely identifiable through its embedded computing system but is able to interoperate within the existing Internet infrastructure. Experts estimate that the IoT will consist of almost 50 billion objects by 2020. The IoT is enabled by the latest developments in RFID, smart sensors, communication technologies, and Internet protocols. The basic premise is to have smart sensors collaborate directly without human involvement to deliver a new class, of applications. The current revolution in Internet, mobile, and machine-to-machine (M2M) technologies can be seen as the first phase of the IoT. In the coming years, the IoT is expected to bridge diverse technologies to enable new applications by connecting physical objects together in support of intelligent decision making. IOT mainly contains six elements to manage its operations identification, sensing, communication, computation, services and semantics. Single Board Computers (SBCs) integrated with sensors and built-in TCP/IP and security functionalities are typically used to realize IoT products (e.g., Arduino Yun, Raspberry PI, Beagle Bone Black, etc.). Such devices typically connect to a central management portal to provide the required data by customers. IOT uses IPv6 addresses to exchange the data between physical objects. IoT protocols are divided into four broad categories, namely: application protocols, service discovery protocols, infrastructure protocols and other influential protocols. Common Operating Systems that are used in IoT environments are Tiny OS, Contiki, Lite OS, Riot OS and Android. Smart buildings, Smart home, smart Grids, smart City and smart healthcare are different services that are provided by IoT.[4]

With the trend going on in ubiquitous computing, everything is going to be connected to the Internet and its data will be used for various progressive purposes, creating not only information from it, but also knowledge and even wisdom. In this paper, we have reported an effective implementation for Internet of Things used for monitoring and controlling regular domestic conditions by means of low cost ubiquitous sensing system. The description about the integrated network architecture and the interconnecting mechanisms for reliable measurement of parameters by smart sensors and transmission of data via Internet is being presented. The environmental sensing system is also able to corresponding loads based on the decisions taken by remote user or server. The longitudinal learning system is able to provide self-control mechanism for better operations of the devices in monitoring stage. Along with the environmental conditions system is also able to provide home security based on alarms input. The paper proposes a framework for effective monitoring and control with MQTT as application layer protocol between server and users as well as server and Gateway. The system incorporates ZigBee for communication between Wireless Sensing Nodes and Gateway.[6]

CHAPTER 3

SYSTEM ANALYSIS

CHAPTER 3
SYSTEM ANALYSIS

3.1 EXISTING SYSTEM

Bluetooth based home automation system using cell phones: In Bluetooth based home automation system the home appliances are connected to the Arduino BT board at input output ports using relay. The program of Arduino BT board is based on high level interactive C language of microcontrollers; the connection is made via Bluetooth. The password protection is provided so only authorized user is allowed to access the appliances. The Bluetooth connection is established between Arduino BT board and phone for wireless communication. In this system the python script is used and it can install on any of the Symbian OS environment, it is portable. One circuit is designed and implemented for receiving the feedback from the phone, which indicate the status of the device.

3.1.1 PROBLEM DEFINITION:

It needs to improve the Smart home automation surveillance techniques. its needs more secure and efficient .

3.2 PROPOSED SYSTEM

The main aim of this project is the utilization of Raspberry Pi to depict a security alert framework utilizing low processing power chips utilizing Internet of things which screens and get alerts when movement is identified and sends photographs and recordings to a cloud server. Moreover, Internet of things (IoT) based application can be utilized remotely to see the movement and get warnings when movement is identified. The photographs and recordings are sent straight forward to a cloud server, sent as Gmail Notifications with snapshots and SMS alerts and when the cloud is not accessible then the information is put away locally on the Raspberry Pi and sent when the association resumes. Accordingly, points of interest such as these make this applications perfect for smart security surveillance monitoring where ever the security is a big concern and required security alert system with instant notifications such as in Industries, Banks, IT Offices and in Homes, this system can be best utilized. The whole report is centered on the field of embedded system, Internet of Things and the use of Linux based OS to run applications on them. Existing system In the existing method the Raspberry Pi is connected to web camera to detect and capture the snapshots, videos of the motion when detected and the RJ45 LAN cable to connect to the Internet for sending and receiving data.

CHAPTER 4

REQUIREMENTS SPECIFICATION

CHAPTER 4

REQUIREMENTS SPECIFICATION

4.1 HARDWARE REQUIREMENT

- RASPBERRY PI
- PIR SENSOR
- PI CAMERA

4.2 SOTWARE REQUIREMENT

- PYTHON

4.3 HARDWARE SPECIFICATION

4.3.1 RASPBERRY PI

The Raspberry Pi is a low cost, credit-card sized computer that plugs into a computer monitor or TV, and uses a standard keyboard and mouse. It is a capable little device that enables people of all ages to explore computing, and to learn how to program in languages like Scratch and Python.

FIG 4.1 RASPBERRY PI 3

4.3.2 PIR SENSORS

Passive infrared sensor (PIR sensor) is an electronic sensor that measures infrared (IR) light radiating from objects in its field of view. They are most often used in PIR-based motion detectors. PIR sensors are commonly used in security alarms and automatic lighting applications. PIR sensors detect general movement, but do not give information on who or what moved. For that purpose, an imaging IR sensor is required. PIR sensors are commonly called simply "PIR", or sometimes "PID", for "passive infrared detector". The term passive refers to the fact that PIR devices do not radiate energy for detection purposes. They work entirely by detecting infrared radiation (radiant heat) emitted by or reflected from objects.

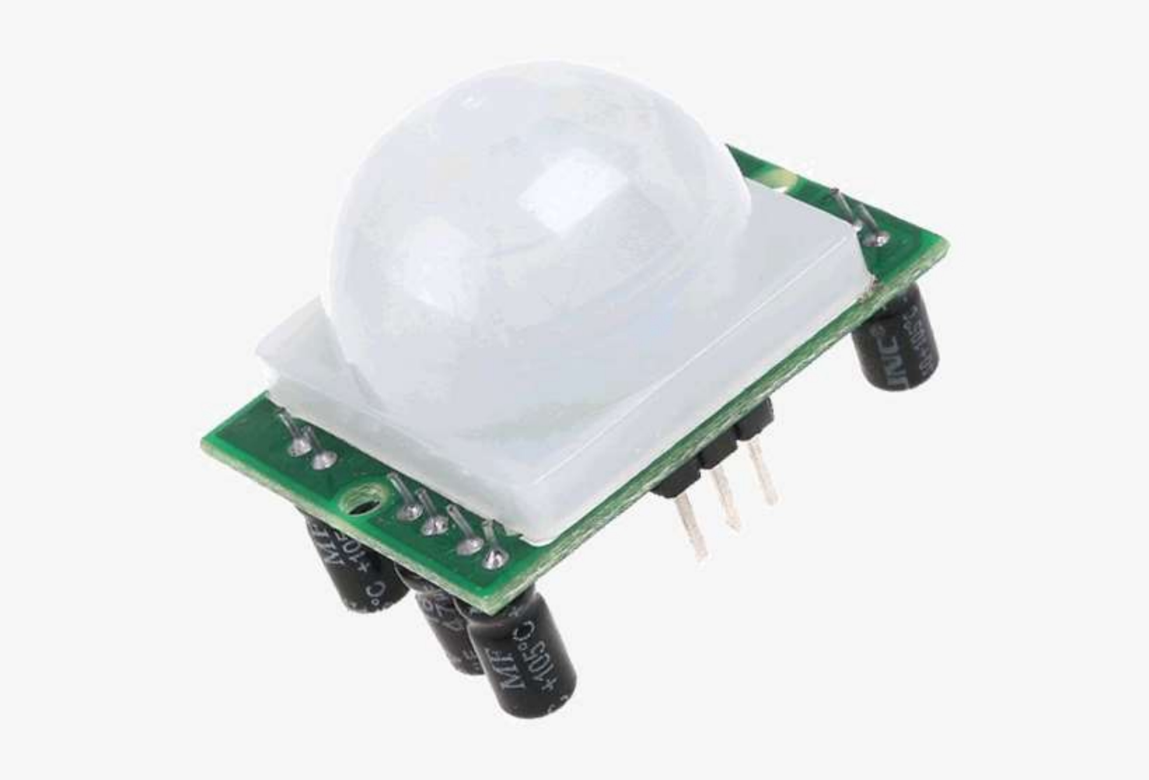


FIG 4.2 PIR SENSOR

4.3.2.1 OPERATION

A PIR sensor can detect changes in the amount of infrared radiation impinging upon it, which varies depending on the temperature and surface characteristics of the objects in front of the sensor. When an object, such as a person, passes in front of the background, such as a wall, the temperature at that point in the sensor's field of view will rise from room temperature to body temperature, and then back again. The sensor converts the resulting change in the incoming infrared radiation into a change in the output voltage, and this triggers the detection. Objects of similar temperature but different surface characteristics may also have a different infrared emission pattern, and thus moving them with respect to the background may trigger the detector as well. PIRs come in many configurations for a wide variety of applications. The most common models have numerous Fresnel lenses or mirror segments, an effective range of about 10 meters (30 feet), and a field of view less than 180. Models with wider fields of view, including 360°, are available, typically designed to mount on a ceiling. Some larger PIRs are made with single segment mirrors and can sense changes in infrared energy over 30 meters (100 feet) from the PIR. There are also PIRs designed with reversible orientation mirrors which allow either broad coverage (110° wide) or very narrow "curtain" coverage, or with individually selectable segments to "shape" the coverage.

4.3.3 PI CAMERA

The Pi camera module is a portable light weight camera that supports Raspberry Pi. It communicates with Pi using the MIPI camera serial interface protocol. It is normally used in image processing, machine learning or in surveillance projects.

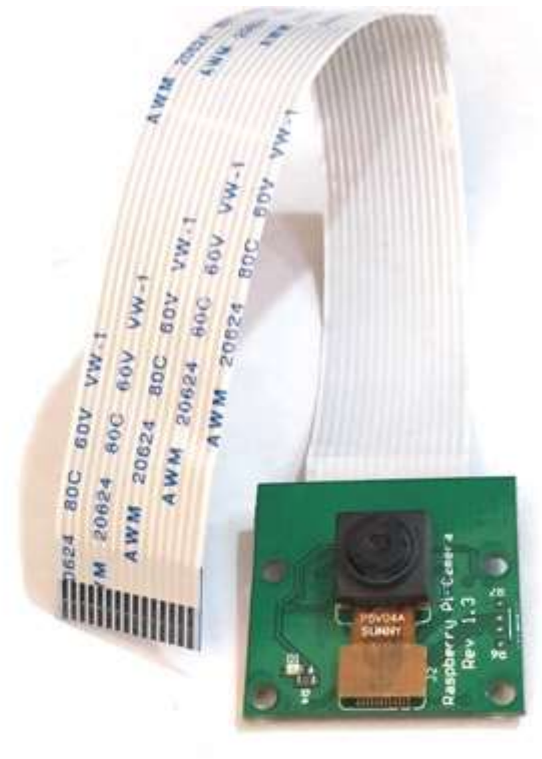


FIG 4.3 PI CAMERA

4.4 SOFTWARE SPECIFICATION

4.4.1 PYTHON

Python is an interpreted, high-level and general-purpose programming language. Created by Guido van Rossum and first released in 1991, 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. Python is dynamically typed and garbage-collected. It supports multiple programmingparadigms,including structured (particularly, procedural), ob ject-oriented, and functional programming. Python is often described as a "batteries included" language due to its comprehensive standard library.

Python was created in the late 1980s as a successor to the ABC language. Python 2.0, released in 2000, introduced features like list comprehensions and a garbage collection system with reference counting.

Python 3.0, released in 2008, was a major revision of the language that is not completely backward-compatible, and much Python 2 code does not run unmodified on Python 3.

The Python 2 language was officially discontinued in 2020 (first planned for 2015), and "Python 2.7.18 is the last Python 2.7 release and therefore the last Python 2 release. No more security patches or other improvements will be released for it. With Python 2's end-of-life, only Python 3.5.x and later are supported.

Python interpreters are available for many operating systems. A global community of programmers develops and maintains C Python, a free and open-source reference implementation. A non-profit organization, the Python Software Foundation, manages and directs resources for Python and C Python development.

CHAPTER 5

SYSTEM DESIGN

CHAPTER 5

SYSTEM DESIGN

5.1 SYSTEM ARCHITECTURE

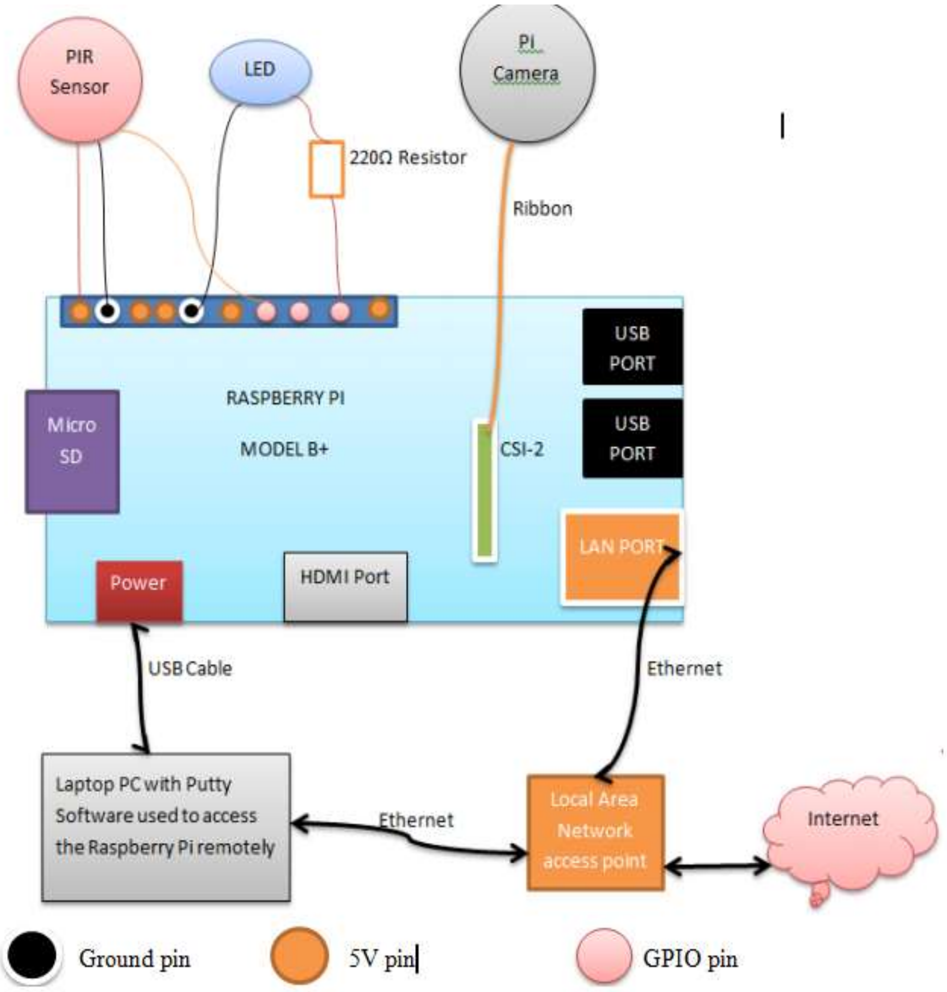


FIG 5.1 SYSTEM ARCHITECTURE

5.2 FLOW CHART DIAGRAM

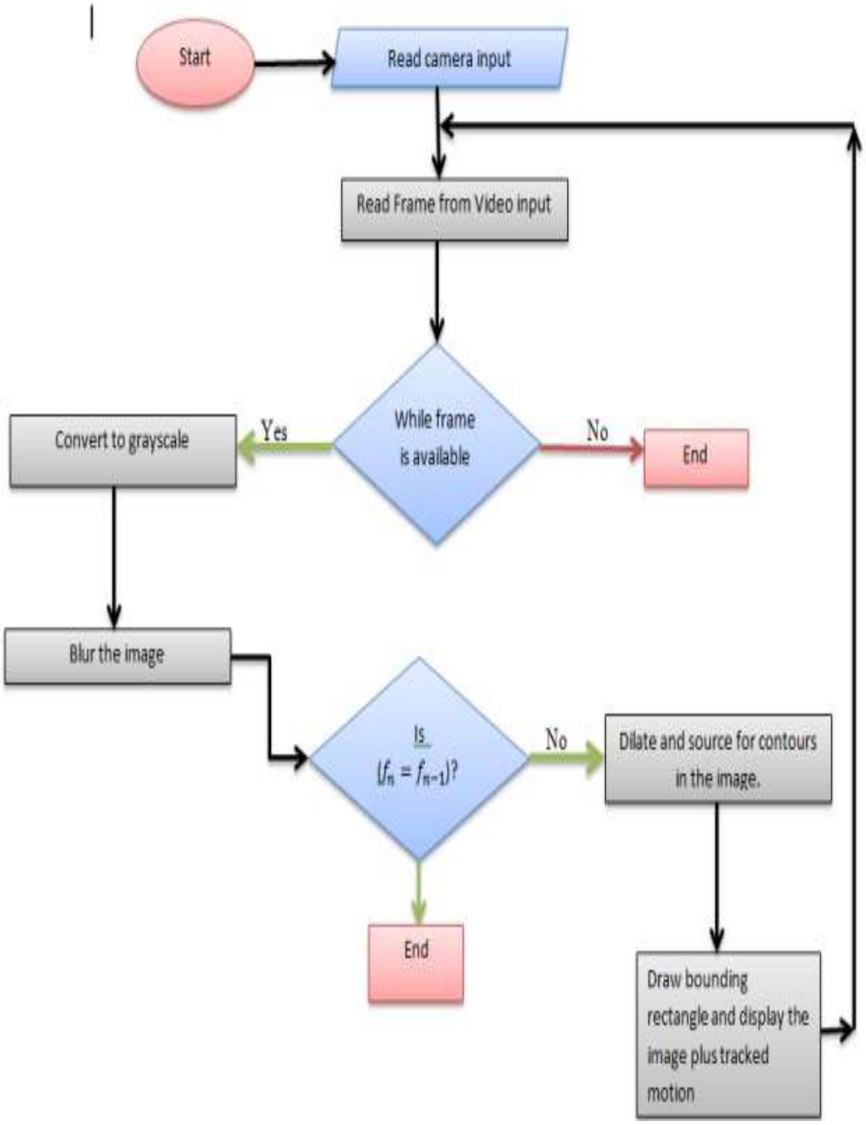

Start
Read camera input
Read Frame from Video input
While frame is available
Yes
No
End
Convert to grayscale
Blur the image
Is $(f_n = f_{n-1})$?
No
Dilate and source for contours in the image.
End
Draw bounding rectangle and display the image plus tracked motion

FIG 5.2 FLOW CHART DIAGRAM

5.3 USECASE DIAGRAM

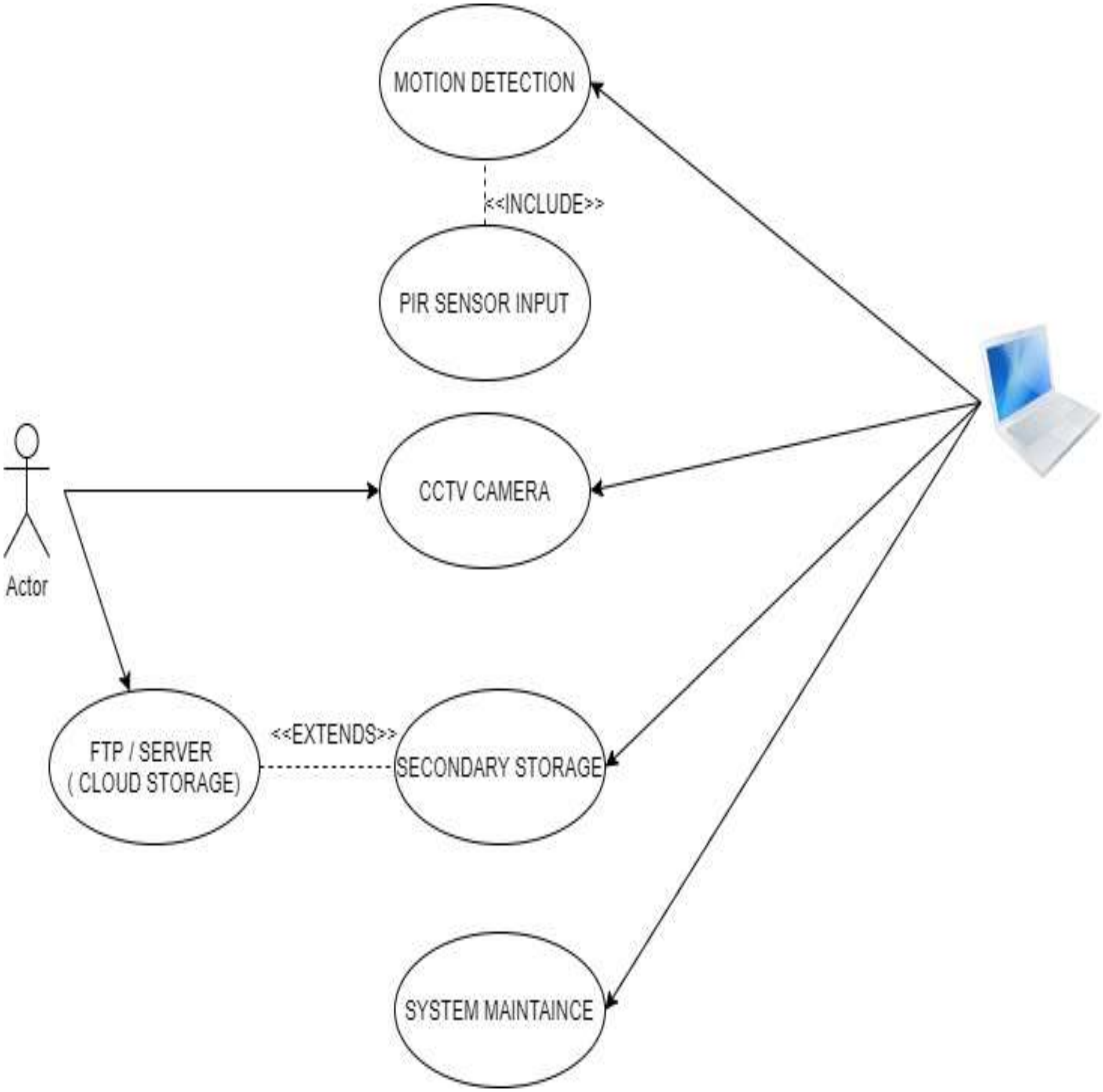


FIG 5.3 USECASE DIAGRAM

5.4 ACTIVITY DIAGRAM

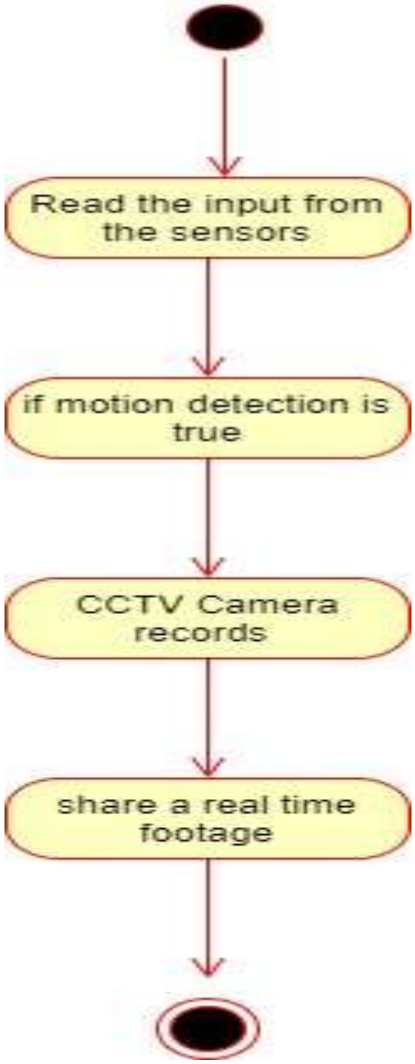

Read the input from the sensors
if motion detection is true
CCTV Camera records
share a real time footage

FIG 5.4 ACTIVITY DIAGRAM

5.5 SEQUENCE DIAGRAM

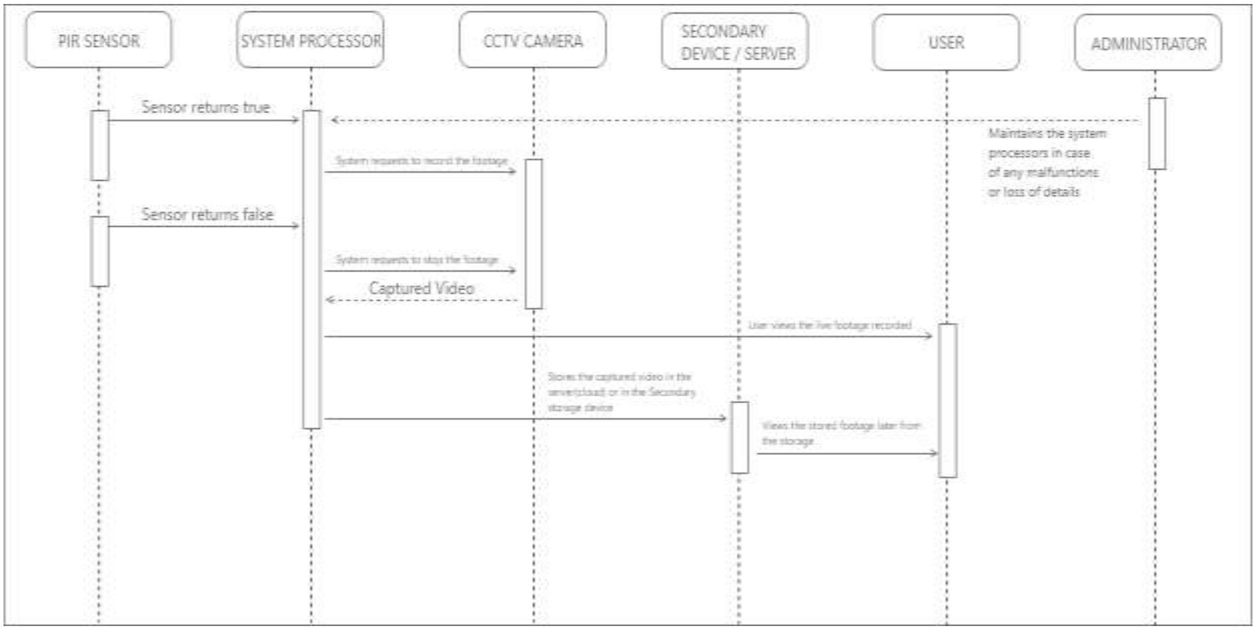

PIR SENSOR
SYSTEM PROCESSOR
CCTV CAMERA
SECONDARY DEVICE / SERVER
USER
ADMINISTRATOR
Sensor returns true
Maintains the system processors in case of any malfunctions or loss of details
System requests to record the footage
Sensor returns false
System requests to stop the footage
Captured Video
User views the live footage recorded
Stores the captured video in the server(cloud) or in the Secondary storage device
Views the stored footage later from the storage

FIG 5.5 SEQUENCE DIAGRAM

CHAPTER 6

SYSTEM ARCHITECTURE

CHAPTER 6
SYSTEN ARCHITECTURE

6.1 PROJECT DESCRIPTION AND MODULE

The main objective of the project is to reduce the size of the stored data/ video in the secondary memory in home surveillance / also in industrial surveillance. To obtain that we propose a system that only records the information which has the motion, that is whenever there is a motion detected the system records and saves only those information. The architecture for our system consists of 3 major components they are Raspberry Pi 3 Processor, PI Camera of 2Mp and last but not the least the PIR sensor. The PIR sensor is used as the motion detection sensor; it uses the Passive Infrared radiation to detect the motion in the desired range. And the input from the PIR sensor is used as a trigger to activate the camera recording. Whenever the PIR sensor returns true then the Raspberry Pi process the input and accesses the camera module to record the video using the PI camera. Here the language we use to interact with raspberry pi processor is Python which is again one of the easiest and also developing languages. Using this language we code and interact with the system and also make the system working process easier. In python there are many predefined libraries we can use and using those if we want to capture the video it doesn’t take the direct video it takes the sequence of frames and then process it to the final video. But coding for the individual camera integrating with a single motion detection sensor seems easy but as we can see Raspberry pi is also used as a mini computer so it can process multiple integrated camera and PIR sensors in the Home surveillance / industrial surveillance. But still though we take the storage of only the motion captured video the clarity of the video is not compressed or reduced so there may be varying in the size of the captured video in accordance to the clarity of the video taken by the camera. As the system is said each of the main modules and the steps of installation is mentioned below.

6.1.1 PIR Module

The PIR (Passive Infra-Red) Sensor is a Pyro electric device that detects human body motion by measuring changes in the infrared levels emitted by surrounding objects [8]. This motion can be detected by checking for a high signal on a single I/O pin. Incorporating a Fresnel lens and motion detection circuit. High sensitivity and low noise. The output is a standard 5V active low output signal. The module provides an optimized circuit that will detect motion up to 6 meters away inexpensive and easy to use, The Output can be connected to GPIO pins of Raspberry Pi directly to monitor signal.

6.1.2 PI camera Module

The Raspberry Pi Camera Module is a custom designed add on for Raspberry Pi . It attaches to Raspberry Pi by way of one of the two small sockets on the board upper surface. This interface uses the dedicated CSI interface, which was designed especially for interfacing to cameras. The Raspberry Pi Camera Module is a 5MP CMOS camera with a fixed focus lens that is capable of capturing still images as well as high definition video. Stills are captured at a resolution of 2592 x 1944, while video is supported at 1080p at 30 FPS, 720p at 60 FPS and 640x480 at 60 or 90 FPS . The camera is supported in the latest version of Raspbian, Raspberry Pi's preferred operating system.

6.1.3 Raspberry pi Module

Raspberry pi is a small credit-card sized computer capable of performing various functionalities such as in surveillance systems, military applications, etc. It is the main element in the field of the internet of things . It provides access to the internet and hence the connection of automation system with remote location controlling device becomes possible. Raspberry Pi is available in various versions. The other major advantage is that it is a simple circuit and the operating system used here is Raspbian OS. The processor at the heart of the Raspberry Pi system is a Broadcom BCM2835 system-on-chip (SoC) multimedia processor.

6.1.4 THE PUTTY CONFIGURATION

1. Obtain a copy of PUTTY pre-configured for use at Columbia from the PUTTY download page.

2. Save the installer file to your download directory or desktop.

Important: Exit all applications before you begin the installation process.

3. Double-click on the file PuTTY-install.exe to begin the installation.

4. At the Choose Destination screen, Click Next to accept the recommended default destination location

for installing PUTTY.

5. Click Next on the Select Program Folder screen to select PUTTY as the recommended Program Folder

name (PUTTY should already be displayed in the Program Folders text box).

6. Click Finish on the final screen to complete the installation.

6.1.5 INSTALLATION OF VNC VIEWER

Once you have an operating system running here are the required steps to install a VNC server:

1.Open LX terminal

2. In LX terminal type in "sudo apt-get update" to update the operating system to the latest version

3. When prompted to update type in "y" and press enter.

4. After updating is complete type in "sudo install tightvncserver". Press "y" and hit enter when prompted

5. Once tight VNC server has completed installation you can start it by typing in "vncserver:1"

6. It will prompt you to create a password. Keep in mind passwords can be at MOST 8 characters long.

7. Enter a password .The VNC server is now running in the background of your Raspberry Pi's operating system. We use any computer on your network with a VNC client to remotely access the Raspberry Pi

6.1.6 STEPS TO INSTALL RASPBIAN OS

In order to install Raspbian OS,first next out of box software(NOOBS) has to be installed.

1.First step is to allocate the drive for installing OS

2.SD adaptor can also be used for this purpose

3.Download WINDISK 32 utility from source forge project which is a zip file

4.Extract and run the zip file

5.Select the file and click run as administrator

6.Select the image file which was extracted above

7.Select the drive letter of the SD card in the device box

8.Click write and wait for write process to complete

9.Exit the image and eject the SD card

CHAPTER 7

SYSTEM IMPLEMENTATION

CHAPTER 7
SYSTEM IMPLEMENTATION

7.1 ALGORITHM

```
# capture frames from the camera
for f in camera.capture_continuous(rawCapture, format="bgr",
use_video_port=True):
        # grab the raw NumPy array representing the image and initialize
        # the timestamp and occupied/unoccupied text
        frame = f.array
        timestamp = datetime.datetime.now()
        text = "Unoccupied"

        ##############################################################
#########
        # COMPUTER VISION
        ##############################################################
#########
        # resize the frame, convert it to grayscale, and blur it
        gray = cv2.cvtColor(frame, cv2.COLOR_BGR2GRAY)
        gray = cv2.GaussianBlur(gray, tuple(conf['blur_size']), 0)

        # if the average frame is None, initialize it
        if avg is None:
                print "[INFO] starting background model..."
                avg = gray.copy().astype("float")
                rawCapture.truncate(0)
                continue
        # accumulate the weighted average between the current frame and
        # previous frames, then compute the difference between the current
        # frame and running average
        frameDelta = cv2.absdiff(gray, cv2.convertScaleAbs(avg))
        cv2.accumulateWeighted(gray, avg, 0.5)
```

7.2 SOURCE CODE

```
from picamera import PiCamera
import RPi.GPIO as GPIO
import time
GPIO.setwarnings(False)
GPIO.setmode(GPIO.BOARD)
#Read output from PIR motion sensor
GPIO.setup(11, GPIO.IN)
#LED output pin
GPIO.setup(3, GPIO.OUT)
while True:
i=GPIO.input(11)
#When output from motion sensor is LOW
if i==0:
print "No intruders",i
camera.stop_recording()
for f in camera.capture_continuous(rawCapture, format="bgr",
use_video_port=True):
# grab the raw NumPy array representing the image and initialize
# the timestamp and occupied/unoccupied text
frame = f.array
timestamp = datetime.datetime.now()
# resize the frame, convert it to grayscale, and blur it
gray = cv2.cvtColor(frame, cv2.COLOR_BGR2GRAY)
gray = cv2.GaussianBlur(gray, tuple(conf['blur_size']), 0)
if avg is None:
print "[INFO] starting background model..."
# accumulate the weighted average between the current frame and
# previous frames, then compute the difference between the current
# frame and running average
frameDelta = cv2.absdiff(gray, cv2.convertScaleAbs(avg))
cv2.accumulateWeighted(gray, avg, 0.5)
avg = gray.copy().astype("float")
rawCapture.truncate(0)
#Turn OFF LED
GPIO.output(3, 0)
time.sleep(0.1)
#When output from motion sensor is HIGH
elif i==1:
print "Intruder detected",i
camera.start_recording('/home/pi/Desktop/video.h264')
#Turn ON LED
GPIO.output(3, 1)
time.sleep(0.1)
else:
 camera.start_preview()
```

CHAPTER 8

TESTING

CHAPTER 8

TESTING

8.1 TESTING TECHNIQUES / TESTING STRATERGIES

8.1.1 TESTING

Testing is a process of executing a program with the intent of finding an error. A good test case is one that has a high probability of finding an as-yet – undiscovered error. A successful test is one that uncovers an as-yet- undiscovered error. System testing is the stage of implementation, which is aimed at ensuring that the system works accurately and efficiently as expected before live operation commences. It verifies that the whole set of programs hang together. System testing requires a test consists of several key activities and steps for run program, string, system and is important in adopting a successful new system. This is the last chance to detect and correct errors before the system is installed for user acceptance testing.

The software testing process commences once the program is created and the documentation and related data structures are designed. Software testing is essential for correcting errors. Otherwise the program or the project is not said to be complete. Software testing is the critical element of software quality assurance and represents the ultimate the review of specification design and coding. Testing is the process of executing the program with the intent of finding the error. A good test case design is one that as a probability of finding an yet undiscovered error. A successful test is one that uncovers an yet undiscovered error. Any engineering product can be tested in one of the two ways:

8.1.1.1 WHITE BOX TESTING

This testing is also called as Glass box testing. In this testing, by knowing the specific functions that a product has been design to perform test can be conducted that demonstrate each function is fully operational at the same time searching for errors in each function. It is a test case design method that uses the control structure of the procedural design to derive test cases. Basis path testing is a white box testing.

Basis path testing:

- Flow graph notation
- Cyclometric complexity
- Deriving test cases
- Graph matrices Control

8.1.1.2 BLACK BOX TESTING

In this testing by knowing the internal operation of a product, test can be conducted to ensure that "all gears mesh", that is the internal operation performs according to specification and all internal components have been adequately exercised. It fundamentally focuses on the functional requirements of the software.

The steps involved in black box test case design are:

- Graph based testing methods
- Equivalence partitioning
- Boundary value analysis
- Comparison testing

8.1.2 SOFTWARE TESTING STRATEGIES:

A software testing strategy provides a road map for the software developer. Testing is a set activity that can be planned in advance and conducted systematically. For this reason a template for software testing a set of steps into which we can place specific test case design methods should be strategy should have the following characteristics:

- Testing begins at the module level and works “outward” toward the integration of the entire computer based system.
- Different testing techniques are appropriate at different points in time.
- The developer of the software and an independent test group conducts testing.
- Testing and Debugging are different activities but debugging must be accommodated in any testing strategy.

8.1.2.1 INTEGRATION TESTING:

Integration testing is a systematic technique for constructing the program structure while at the same time conducting tests to uncover errors associated with. Individual modules, which are highly prone to interface errors, should not be assumed to work instantly when we put them together. The problem of course, is “putting them together”- interfacing. There may be the chances of data lost across on another’s sub functions, when combined may not produce the desired major function; individually acceptable impression may be magnified to unacceptable levels; global data structures can present problems.

8.1.2.2 PROGRAM TESTING:

The logical and syntax errors have been pointed out by program testing. A syntax error is an error in a program statement that in violates one or more rules of the language in which it is written. An improperly defined field dimension or omitted keywords are common syntax error. These errors are shown through error messages generated by the computer. A logic error on the other hand deals with the incorrect data fields, out-off-range items and invalid combinations. Since the compiler s will not deduct logical error, the programmer must examine the output.

Condition testing exercises the logical conditions contained in a module. The possible types of elements in a condition include a Boolean operator, Boolean variable, a pair of Boolean parentheses A relational operator or on arithmetic expression. Condition testing method focuses on testing each condition in the program the purpose of condition test is to deduct not only errors in the condition of a program but also other a errors in the program.

8.1.2.3 SECURITY TESTING:

Security testing attempts to verify the protection mechanisms built in to a system well, in fact, protect it from improper penetration. The system security must be tested for invulnerability from frontal attack must also be tested for invulnerability from rear attack. During security, the tester places the role of individual who desires to penetrate system.

8.1.2.4 VALIDATION TESTING

At the culmination of integration testing, software is completely assembled as a package. Interfacing errors have been uncovered and corrected and a final series of software test-validation testing begins. Validation testing can be defined in many ways, but a simple definition is that validation succeeds when the software functions in manner that is reasonably expected by the customer. Software validation is achieved through a series of black box tests that demonstrate conformity with requirement. After validation test has been conducted, one of two conditions exists.

* The function or performance characteristics confirm to specifications and are accepted.

* A validation from specification is uncovered and a deficiency created.

Deviation or errors discovered at this step in this project is corrected prior to completion of the project with the help of the user by negotiating to establish a method for resolving deficiencies. Thus the proposed system under consideration has been tested by using validation testing and found to be working satisfactorily. Though there were deficiencies in the system they were not catastrophic

8.1.2.5 USER ACCEPTANCE TESTING

User acceptance of the system is key factor for the success of any system. The system under consideration is tested for user acceptance by constantly keeping in touch with prospective system and user at the time of developing and making changes whenever required. This is done in regarding to the following points.

- Input screen design.
- Output screen design.

8.2 TESTING SUMMARY

S.NO	ACTION	INPUT	EXPECTED OUTPUT	ACTUAL OUTPUT	TEST RESULTS	TEST COMMENTS
1	PIR INPUT	TRUE / FALSE	TRUE / FALSE	TRUE / FALSE	PASS	PIR SENSOR TESTED AND WORKING PERFECTLY
2	CAMERA MODULE	PIR SENSOR RETURNS TRUE	STARTS RECORDING	STARTS RECORDIN G	PASS	WHEN MOTION DETECTED CAMERA STARTS RECORDING
3	CAMERA MODULE	PIR SENSOR RETURNS FALSE	STOPS RECORDING	STOPS RECORDIN G	PASS	WHEN MOTION NOT DETECTED CAMERA

						STARTS RECORDING
4	CAMERA MODULE	PIR RETURNS NONE	NO RECORDING BUT LIVE STREAM	NO RECORDING BUT LIVE STREAM	PASS	NOT RECORDED WHEN NO INPUT FROM THE PIR SENSOR GOES LIVE STREAMING
5	LOCAL STORAGE	RECORDED VIDEO	FILE SAVED WITH DATE AND TIME	FILE SAVED WITHOUT DATE AND TIME	FAIL	RECORDED VIDEO MUST HAVE THE DATE AND TIME MENSIONED
6	LIVE FEED	ANY INPUT (LIKE PAUSE PLAY STOP)	ACTION MUST BE PERMITED	NO ACTION TAKEN PLACE	FAIL	DURING THE FEED WE CANNOT PERFORM ANY ACTIONS, BUT CAN VIEW FROM THE STORED DATA

CHAPTER 9

CONCLUSION

CHAPTER 9

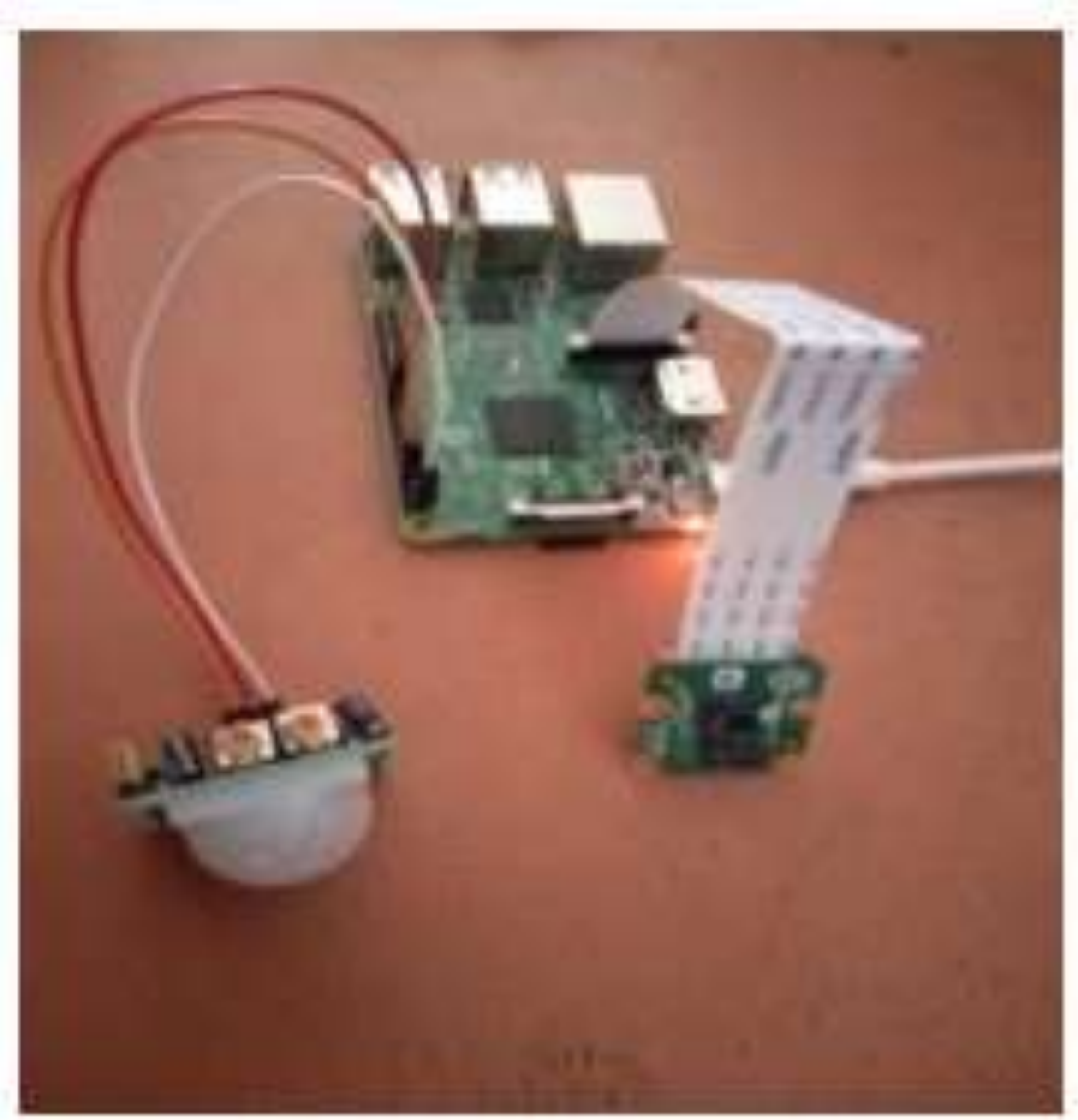
CONCLUSION

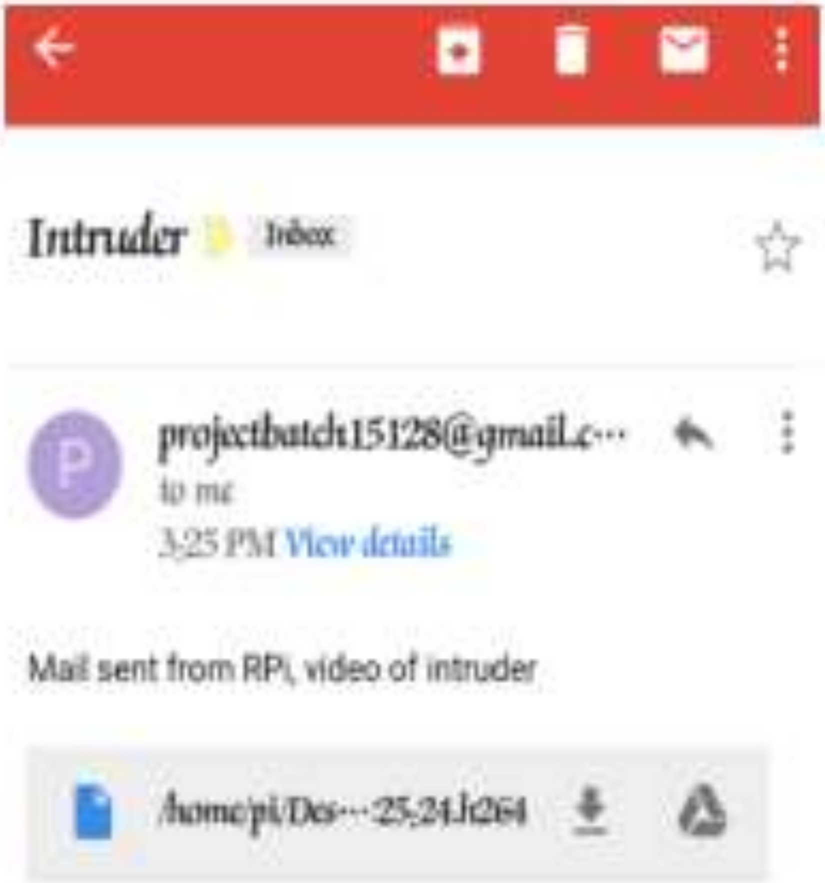
Hence IoT based smart security surveillance system using Raspberry pi have been successfully designed and implemented which is capable of recording the videos and capturing the images and the same has been uploading to cloud service Dropbox if it is not available then stores locally on Raspberry Pi storage. At the same time SMS notifications and Gmail notifications with captured snapshots will send to user. Live video streaming also provided to monitor continuously. It is advantageous as it offers reliability and privacy on both sides. It is authenticated and encrypted on the receiver side; hence it offers only the person concerned to view the details. Necessary action can be taken in short span of time in the case of emergency conditions such as industries, offices, military areas, smart homes, elderly person falling sick etc.

APPENDICES

APPENDICES

A.1 Sample Screen





Intruder Inbox

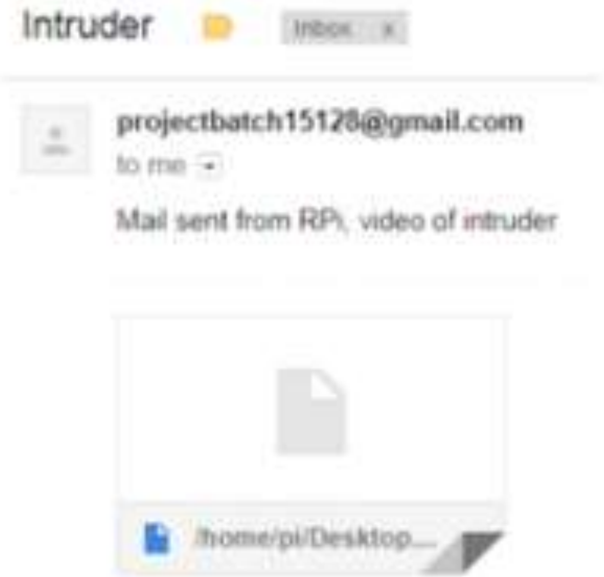
projectbatch15128@gmail.c···

to me

3:25 PM View details

Mail sent from RPi, video of intruder

/home/pi/Des···25;24.h264



Intruder Inbox x

projectbatch15128@gmail.com

to me

Mail sent from RPi, video of intruder

/home/pi/Desktop...

REFERENCE

REFERENCE

1. Shrikanthan N. Tan F Karande A., “Bluetooth Based Home Automation” , Microprocessors Micro sysytems, Elsevier Science, B.V. vol.26,no.6, pp281-289,(2009)

2. Piyare R., Tazil M., “Bluetooth based Home automation system using Cell Phone”. Consumer Electronics,IEEE 15th International Symposium on, vol.45, no.3, pp.192-195(2011).

3. EASAMBATTU, Thejaswini; REDDY, P. Ajay Kumar; RAMAIAH, G.N. Kodanda. Controlling home appliances through GSM modem and Internet. International journal of Electronics Engineering Research, [S.I.], p. 1-7, oct 2013.

4. P. Magrassi, T. Berg, “A World of Smart Objects”, Gartner research report R-17-2243,12 August 2002.

5. White Paper: “Internet of Things Strategic Research Roadmap”, Antoine de Saint-Exupery, 15 sep 2009.

6. Souza, Alberto M.C. Amazonas, Jose R.A. “A Novel Smart Home Application Using an Internet ofThings Middleware”, Proceedings of 2013 European Conference on Smart Objects, Systems and Technologies (SmartSysTech), pp. 1 – 7, June 2013.

7. Perumal, T, Ramli, A.R, Chui Yew Leong, “Design and implementation of SOAP based residential management for smart home systems”, IEEE Transactions on

8. Ming Wang, Guiqing Zhang, Chenghui Zhang, Jianbin Zhang, Chengdong Li. "An IoT-based appliance control system for smart homes", Fourth International

9. Conference on Intelligent Control and Information Processing (ICICIP), pp. 744 – 747, June 2013.

10. Yepeng Ni, Fang Miao, Jianbo Liu, Jianping Chai. "Implementation of Wireless Gateway for Smart Home", Communications and Network, pp. 16-20,2013.

11. Kelly, S.D.T., Suryadevara, N.K.; Mukhopadhyay, S.C. "Towards the Implementation of IoT for Environmental Condition Monitoring in Homes", Sensors Journal, IEEE, Volume: 13, pp- 3846 –3853, May 2013.

[1] Z. Sundas, "Motion Detecting Camera Security System with Email Notifications and Live Streaming Using Raspberry Pi." .

[2] M. Peter and H. David, "Learn Raspberry Pi with Linux," Apress, 2012.

[3] P. S. Dhake and B. Sumedha S., "Embedded Surveillance System Using PIR Sensor.," vol.No. 02, no. 3, 2014.

[4] J. D., "Real Time Embedded Network Video Capture And SMS Alerting system," Jun.2014.

[5] S. Sneha, "IP Camera Video Surveillance using Raspberry Pi.," Feb. 2015.

[6] F. C. Mahima and A. Prof. Gharge, "Design and Develop Real Time Video Surveillance System Based on Embedded Web Server Raspberry PI B+ Board. International Journal of

Advance Engineering and Research Development (Ijaerd), NCRRET.," pp. 1–4, 2015.

[7] J. G. J, "Design and Implementation of Advanced ARM Based Surveillance System Using Wireless Communication.," 2014.

[8] P. Sanjana, J. S. Clement, and S. R., "Smart Surveillance Monitoring System Using Raspberry PI and PIR Sensor.," 2014.

[9] U. Kumar, R. Manda, S. Sai, and A. Pammi, "Implementation Of Low Cost Wireless Image Acquisition And Transfer To Web Client Using Raspberry Pi For Remote Monitoring.

International Journal of Computer Networking, Wireless and Mobile Communications (IJCNWMC).," vol. No. 4, no. 3, pp. 17–20, 2014.

[10] "The History of Security _ PerspecSys.com.htm." .

[11] A.-D. Osama, "Cisco IP Video Surveillance Introduction," Cisco Expo, 2009.

[12] "What is a security system and how does it work _ SafeWise.htm." .

[13] T.K. Hareendran, "GSM Home Security Alarm System With Arduino," Library Security System, 2014. .

[14] R. Verman, "Distance Education In Technological Age," Anmol Publ. Pvt Ltd, p. 166,2005.

[15] "Television Rides Wires," Pop. Sci., no. February, p. 179, 1949. [16] "Introduction to Closed Circuit Television," Jan-2013. .

[17] B. Messauod, Access Control Systems: Secuiry, Management and Trust Models., 1st ed.Austin, TX, USA: Springer, 2006.

[18] “IP Surveillance,” IT Encyclopedia.

[19] “ijcsit2014050648.pdf.” .

[20] B. E. Reddy, M. Veeresha, and N. Rao, “Image Processing: A Survey.”

[21] A. Ambrosetti and P. H. Rabinowitz, “Dual variational methods in critical point theory and

applications,” J. Funct. Anal., vol. 14, no. 4, pp. 349–381, 1973.