VOL 2 (2018) NO 4 - 2

e-ISSN : 2549-9904
ISSN : 2549-9610

INTERNATIONAL JOURNAL ON INFORMATICS VISUALIZATION

Smart Indoor Home Surveillance Monitoring System Using Raspberry Pi

Lee Han Keat#, Chuah Chai Wen#

#*Faculty of Computer Science and Information Technology, University Tun Hussein Onn Malaysia, Malaysia*
Email: hankeat95@gmail.com, cwchuah@uthm.edu.my

Abstract— **Internet of Things (IoTs) are internet computing devices which are connected to everyday objects that can receive and transmit data intelligently. IoTs allow human to interact and control everyday objects wirelessly to provide more convenience in their lifestyle. The Raspberry Pi is a small, lightweight and cheap single board computer that can fit on human's palm. Security plays a big role in a home. People concern about security by preventing any intruders to enter their home. This is to prevent loss of privacy and assets. The closed-circuit television (CCTV) is one of the device used to monitor the secured area for any intruders. The use of traditional CCTV to monitor the secured area have three limitations, which are requiring a huge volume of storage to store all the videos regardless there are intruders or not, does not notify the users immediately when there are motions detected, and users must always check the CCTV recorded videos regularly to identity any intruders. Therefore, a smart surveillance monitoring system is proposed to solve this problem by detecting intruders and capturing image of the intruder. Notifications will also be sent to the user immediately when motions are detected. This smart surveillance monitoring system only store the images of the intruders that triggered the motion sensor, making this system uses significantly less storage space. The proposed Raspberry Pi is connected with a passive infrared (PIR) motion sensor, a webcam and internet connection, the whole device can be configured to carry out the surveillance tasks. The objectives of this project are to design, implement and test the surveillance system using the Raspberry Pi. This proposed surveillance system provides the user with live stream of video feed for the user. Whenever a motion is detected by the PIR motion sensor, the web camera may capture an image of the intruder and alert the users (owners) through Short Message Service (SMS) and email notifications. The methodology used to develop this system is by using the object-oriented analysis and design (OOAD) model.**

Keywords— **Internet of Things, Raspberry Pi, Closed-Circuit Television, Smart Surveillance Monitoring System, Short Message Service, Email**

I. INTRODUCTION

The Raspberry Pi is a small, lightweight and cheap single board computer that fit on human's palm. The board comes with USB ports, a HDMI port, an Ethernet port, processor, RAM, storage (microSD card), GPIO (General Purpose Input Output) pins and an audio output jack [1][2]. The Raspberry Pi is a working computer or laptop, but very small in size. This device is suitable to apply and create Internet of Things (IoT) due to the sufficient capabilities and performance of it [3]. Therefore, this project will propose a smart surveillance monitoring system using the Raspberry Pi that will monitor an area. It can detect motion and sends notifications to the user through SMS and Email. The image of the intruder is captured as well when the motion sensor is triggered. The objectives of this project are:

- To design a surveillance system using the Raspberry Pi.
- To implement the surveillance system using the Raspberry Pi.
- To test the surveillance system using the Raspberry Pi.

The scope of this project is that this proposed surveillance system support for only a single user, which is the owner of the home. This proposed system is suitable to provide surveillance for a room, preferably the entrance door. The surveillance system using the Raspberry Pi is developed to fulfil all the required modules. There are five modules as shown in Table 1.

TABLE 1
MODULES OF THE PROPOSED SURVEILLANCE SYSTEM

Module	Description
Motion detection	Detect motions (by human) using the passive infrared (PIR) motion detector
SMS notification module	Sends a SMS notification to the user's phone when motion is detected
Email notification module	Sends an email to the user's email together with the captured image when motion is detected

Image capturing	Capture image of the intruder when the intruder triggers the motion detector
Live video feed	Provide live video feed of the webcam to the user
Storage	Store captured images for future usage such as viewing and verifying

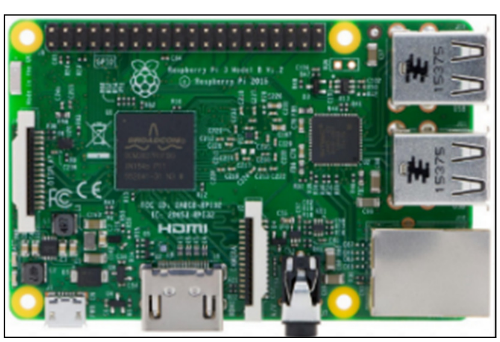
II. Literature Review

The literature review is discussed in this section. The literature reviews include the Raspberry Pi, Motion Detector Technology, Python Programming Language, Closed-Circuit Television (CCTV) and the comparison between existing Closed-Circuit Television (CCTV) system and proposed Raspberry Pi surveillance system.

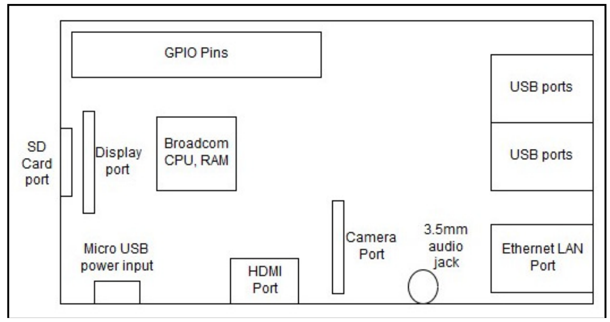
A. *Raspberry Pi*

The Raspberry Pi was created in the United Kingdom by Raspberry Pi Foundation [4]. The aim of creating the Raspberry Pi is to encourage education relating with computer science for the developing countries where computer exposure is very limited. Raspbian is one of the operating system for the Raspberry Pi. Raspbian which is one of the Linux distributions with Debian-based. The Raspberry Pi can be added with accessories to increase the functionalities. For example, a camera board, Gertboard and HAT (Hardware Attached on Top). The camera board has a design of a flexible flat cable (FFC) that is used to plug into the provided camera interface slot on the Raspberry Pi board [5]. The camera interface can be found in between the HDMI ports and the Ethernet port. The camera board can produce up to 1080p video. Gertboard is a device that expands the default GPIO pins on the Raspberry Pi. The Gertboard can be used to control LEDs, sensors, analog signals, and switches as well. The HAT expansion boards are also another powerful accessory for the Raspberry Pi. The most popular HAT is the Sense HAT. The Sense HAT attaches on the Raspberry Pi using the 40 GPIO pins. The Sense HAT have the orientation, pressure, humidity and temperature sensors [6]. Sense HAT has 8x8 LED Matrix that enables the user to display data from the sensor. The Raspberry Pi can support many programming languages, such as Scratch, HTML5, JavaScript, jQuery, Java, C, C++, Perl, and Erlang [7].

In this proposed smart indoor home surveillance system, the Raspberry Pi 3 Model B is used to develop the surveillance system. This generation of Raspberry Pi was released in February 2016. The Raspberry Pi 3 Model B and the digital layout is as shown in Figure 1.



(a)



(b)

Fig. 1 (a) Raspberry Pi 3 Model B [8]; (b) Digital layout for Raspberry Pi 3 Model B [9]

Based on Figure 1, the Raspberry Pi 3 Model B have the CPU speed between the range of 700 MHz to 1.2GHz. The on-board RAM is at 1 GB. The Raspberry Pi 3 Model B uses microSD card as the main storage for the operating system and programs. The Raspberry Pi 3 Model B also has 4 USB 2.0 slots for connecting keyboard or mouse, HDMI output to display the Raspberry Pi to a display monitor, an audio jack in 3.5mm for audio output, Wi-Fi 802.11n for wireless internet connectivity, built-in Bluetooth function for wireless hardware connectivity, and the 8P8C Ethernet port for wired internet connectivity. The presence of the 8P8C Ethernet port on the Raspberry Pi 3 Model B is to enable users to use wired internet connection if Wi-Fi is not preferable to the user. There are 40 populated GPIO pins on the Raspberry Pi 3 Model B. The Raspberry Pi 3 Model B consists of 26 digital inputs or outputs pins and the remaining 14 pins are used for supplying power to peripheral devices.

B. *Motion Detector Technology*

Motion detector or sensor spots moving objects, are used for detecting humans' motions. Motion detectors can be used widely for security purposes that monitors a specific location for unauthorized access. Motion detector also plays a big role in the security such as a burglar alarm. For example, when the motion detector is triggered, a security camera is activated to capture a video footage or image of the intruded area and notify the users. The proposed system uses a Passive Infrared (PIR) motion detector which works by reacting to the changes of infrared energy level, caused by objects such as human's and animal's movement [10].

C. *Python (Programming Language)*

This programming language also requires user to implement lower quantity of code lines to execute coding concepts compared to languages such as C++ and Java [11]. Python supports object-oriented, imperative, functional programming and procedural styles, together with automatic memory management as well. This makes Python is suitable to be used to develop the proposed Raspberry Pi Surveillance System. The developer of Python programming language is the Python Software Foundation and the license is also provided by the same foundation. The Python Software Foundation is an independent non-profit organization that holds the copyright on Python versions 2.1 and newer. The creation of Python is mainly influenced and inspired from other programming languages like C, C++, Java, Perl, and Lisp.

D. *Closed-Circuit Television*

The closed-circuit television or CCTV is considered as a video surveillance [12]. The CCTV transmit signals to a place such as a CCTV control room by using the video camera devices. CCTV's signal is not openly transmitted. The CCTV is widely used on various areas such as military secure areas, banks, casinos, hotels, airports, schools, hospitals, restaurants, convenience stores and many more. The other specialized usage of CCTV is at the industrial plants. This is where the CCTV is installed at locations where locations are not suitable for humans' presence such as radioactive exposed places. The human can monitor the location from the central CCTV control room. the basic mechanisms of CCTVs are discussed in Table 2.

TABLE 2
BASIC MECHANISMS OF CCTVS [13]

Mechanism	Description
Deterrence	This is where the criminals are conscious of the presence of CCTV. They may assess the risk of doing crime in this location
Efficient Deployment	To enable users monitoring a location to determine whether law officers support is needed
Self-Discipline	• Potential victims- Victims are reminded that potential crimes might happen on locations with CCTVs, making them be extra cautious • Potential criminals- create the feeling of fear to criminals as their crimes might be recorded
Capable guardian	The 'Routine Activity Theory' (Cohen and Felson,1979) said that crimes must have a motivated criminal, a fitting target, and absence of a capable guardian. Therefore, CCTV as a capable guardian may reduce criminal activities
Detection	To capture images/videos of criminals where crimes happen to use as a proof

The CCTV system requires a collection of hardware devices such as cameras, Digital Video Recorder (DVR) Unit, local monitor, network router, and remote devices. Figure 2 shows the connections between the devices in a CCTV system.

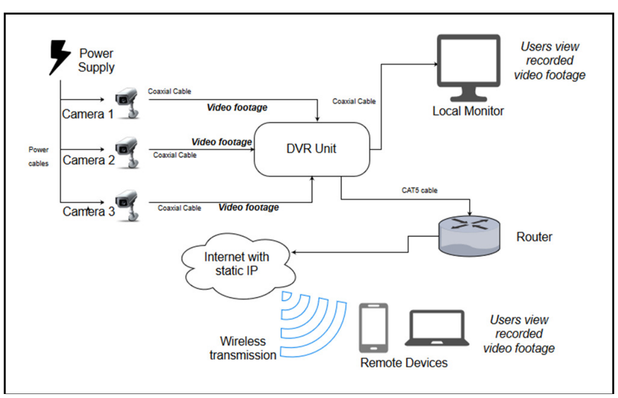


Fig. 2 Connectivity between devices in a CCTV system [14, 15]

From the DVR Unit, users can view the recorded the video footage from a local display monitor. The display monitor should be connected to the DVR Unit using coaxial cable as well. Users also able to view the recorded video footage remotely. This can be done by connecting the DVR Unit to a network router using the CAT5 cable. The router is then connected to the internet with a static IP address. The user can use the IP to view the video footage on their remote devices such as smartphones, tablets and laptops. This allow users to view the video footage at anywhere wirelessly if they are not physically at the DVR Unit.

E. *Comparison Between Existing CCTV System and Proposed Raspberry Pi Surveillance System*

There are differences and similarities between the existing CCTV systems and the proposed Raspberry Pi surveillance system. The comparison between these two are as shown in Table 3.

TABLE 3
COMPARISON BETWEEN EXISTING CCTV SYSTEMS AND THE PROPOSED RASPBERRY PI SURVEILLANCE SYSTEM

Properties	Existing CCTV Systems [16, 17, 18, 19, 20]	Proposed Raspberry Pi Surveillance System
Cloud storage	Unavailable	Available (Dropbox)
Storage usage	Inefficient, storing all video recordings	Efficient, storing only images when motion is triggered
Alerts / Notifications	Unavailable (passive)	Available (SMS and Email)
Power consumption	High	Low
Portability	Static	Can be moved/shifted easily according to current needs

III. METHODOLOGY

This project uses the object-oriented analysis and design (OOAD) model as the methodology. OOAD is a software engineering method that represent the system as a collection of interacting objects [21]. The OOAD is carried out by analyzing the requirements of the proposed system, then design the proposed system that can satisfy all the requirements, implement the design, and finally test the proposed system.

As for the requirements analysis, the requirements of the proposed system are determined by sorting and listing out all the objects needed to be included. The objects together with the respective processes and data input-output are as shown in Table 4.

TABLE 4
OBJECTS, INPUT, PROCESS AND OUTPUT OF THE PROPOSED SYSTEM

Objects	Input	Process	Output
Motion Detection	Infrared radiation	Interpret the detection	Trigger the SMS and Email notifications, and the webcam to

			capture image of intruders
SMS	Motion detection status	If motion is detected, sends a SMS notification to user	SMS notification to the user's phone
Image capturing	Motion detection status	Capture image immediately	Captured image of the intruders
Live video feed	None	Provide current video footage from webcam	Current video footage of the secured location
Storage	Captured image of the intruders	Save captured image to the user's Dropbox database account	A database of captured images of intruders
Email	Motion detection status	If motion is detected, sends an email notification to the user	Email with attached image captured of intruders

IV. System Design

The system design of the proposed smart indoor home surveillance monitoring system using Raspberry Pi is discussed in this section. The system design, database design, and the interface design are included in this section.

A. *System Design*

For the system design, the design can be represented using the Use-Case Diagram, Sequence Diagram, Activity Diagram, Class Diagram and Flow Chart Diagram. The diagrams are presented and discussed in the Figure 3, Figure 4, Figure 5, Figure 6, and Figure 7.

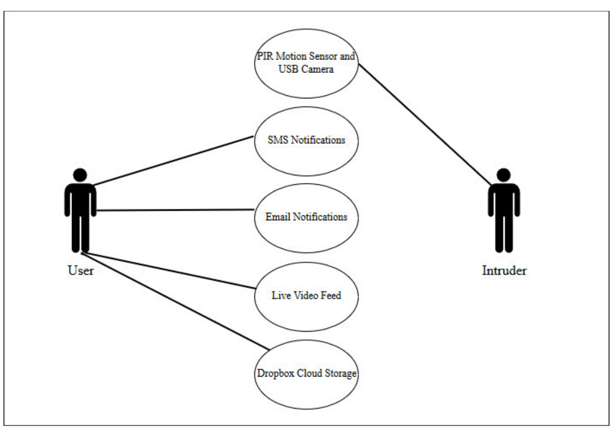


Fig. 3 Use-Case Diagram for the developed system

Based on Figure 3, the intruder triggers the PIR motion sensor and then followed up by the USB Camera to capture an image of the intruder. Meanwhile, the user can receive the SMS notifications and email notifications when intruder is detected by the system. Besides that, the user can view the live video feed on their mobile devices and access to the Dropbox cloud storage that contains all the captured images.

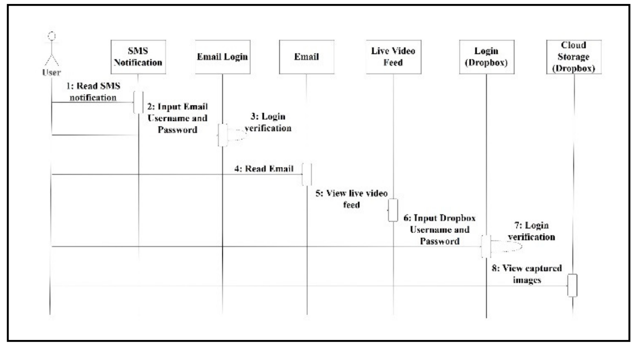


Fig. 4 Sequence diagram for the developed system

Based on Figure 4, the sequence of the system starts up when user read the SMS notification sent from the Raspberry Pi. Next, the user may view and read the email notification from the user's email account. Then, the user can view the live video feed provided by the Raspberry Pi surveillance system. The captured images stored in the user's Dropbox account can be viewed by the user by accessing his or her Dropbox folder.

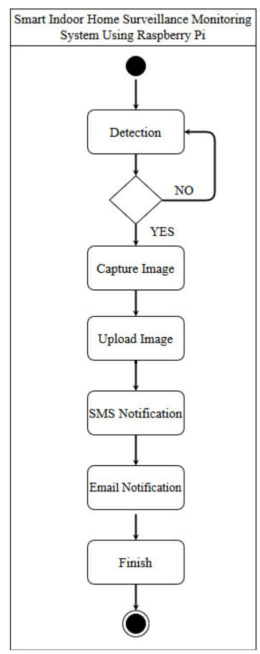


Fig. 5 Activity diagram for the developed system

Based on Figure 5, the activities done by the developed system started off with the detection of intruders in the secured area. The detection phase is repeated if no intruders are detected. If there is intruder detected by the system, the system captures an image of the intruder. Then, the system uploads the captured image to the user's Dropbox account. This is followed by the system sending a SMS notification and an email notification to the user. This finishes the activity done by the system. The whole activity phase is repeated after reaching the end to provide continuous intruders detection by the system.

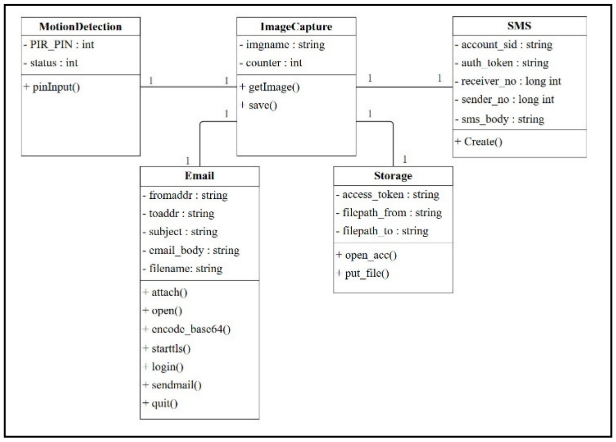


Fig. 6 Class diagram for the developed system

From Figure 6, the class diagram is made up of MotionDetection class, ImageCapture class, SMS class, Email class and a Storage class. All the classes have one-to-one relationship with another class as shown on the figure.

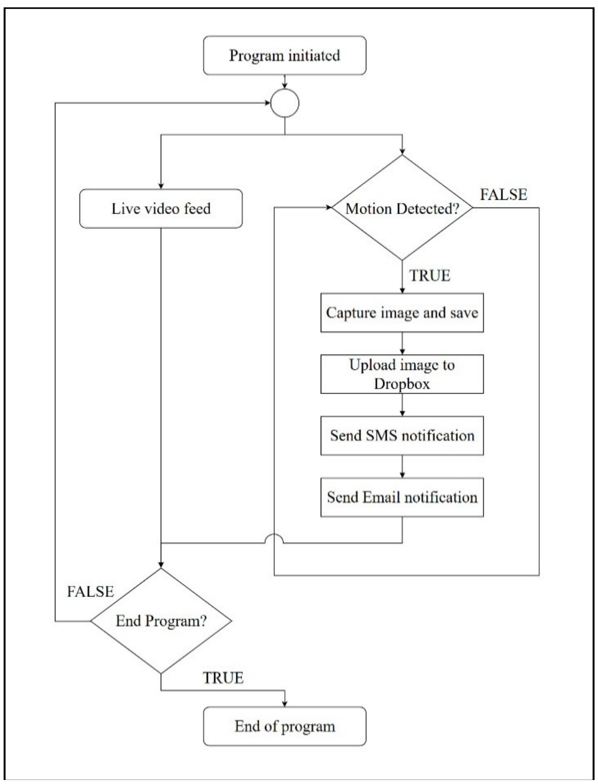


Fig. 7 Flow chart diagram for the developed system

The flow of the surveillance starts with the program being started by the user. This enables the Raspberry Pi surveillance system to be in the active state. During the active state, the live video feed function is enabled. The motion detection of the PIR motion detector is activated as well to spot any motions in the area. If there are motion detected, this triggers the webcam to capture the image immediately and save it. The system then uploads the captured image to the user's Dropbox cloud storage account. It followed by sending a SMS notification to the user's mobile phone about the intrusion. An email is send as well with the captured intruder's image as an attachment. If there are no motion detected, the system return to detecting motion state, making the Raspberry Pi surveillance system continuously detecting motion to avoid any overlook of the intrusions.

B. *Interface Design*

There are two interfaces exposed to the user. The interfaces are the SMS Interface and the Email Interface. The SMS interface for the developed smart indoor home surveillance monitoring system using Raspberry Pi is shown in Figure 8.

Fig. 8 SMS Interface for the developed system

Based on Figure 8, the interface for the SMS notification received by the user is as shown in the figure. The SMS content alerts the user that motion was detected at the secured area, the area where the developed smart indoor home surveillance monitoring system using Raspberry Pi is placed. This notify the user to check the user's email for the captured image of the motion. This is to verify the presence of intruders in the secured area by using the captured image attached in the email.

The email interface for the developed smart indoor home surveillance monitoring system using Raspberry Pi is shown in Figure 9.

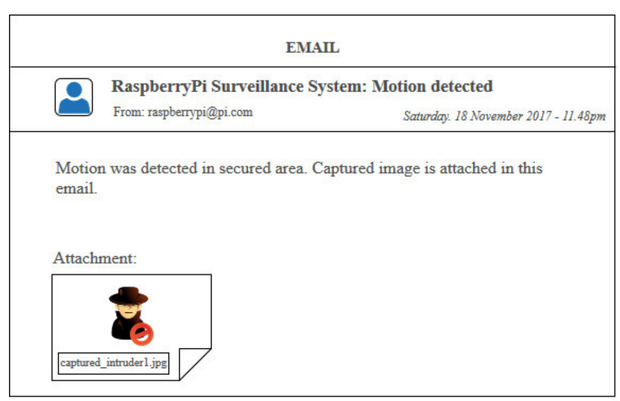


Fig. 9 Email Interface for the developed system

From Figure 9, the interface of the email includes the email subject 'Raspberry Pi Surveillance System: Motion detected' and the content of the email that aims to alert the user that motion was detected at the secured area. In the email, the captured image triggered by the motion is included in the email as an attachment, where the user can view the image to verify if there is really a legitimate intruder have intruded the secured area.

V. IMPLEMENTATION AND TESTING

There are six modules that are implemented in this project. The modules are motion detection, SMS notification, Email notification, image capturing, live video feed, and the Dropbox storage module. The modules are discussed as follows.

A. *Implementation of Motion Detection Module*

The code for the implementation of motion detection module is shown in Figure 10.

```
import RPi.GPIO as GPIO
import time

while RUNNING:
    if GPIO.input(PIR_PIN): # if motion sensor is triggered
        # Execution of SMS, Email, and Dropbox Storage Module codes

    else:
        time.sleep(0.3)      # halt program for 0.3 second
```

Fig. 10 Coding for the implementation of Motion Detection Module

Based on Figure 10, the codes are used to implement the motion detection module. This enables the motion sensor can sends signals to the program when motions are detected.

B. *Implementation of Motion Detection Module*

The code for the implementation of image capturing module is shown in Figure 11.

```
import time
from SimpleCV import Camera, Image

def capture_image(currenttime):
    cam = Camera()
    imgname = "cam_("+currenttime+").jpg"
    img = cam.getImage()
    img.save(imgname)
    print ("Image Captured")
    return img, imgname
```

Fig. 11 Coding for the implementation of Image Capturing Module

Based on Figure 11, the codes are used to implement the image capturing module. This allow the USB camera to capture an image after the motion detector is triggered by an intruder.

C. *Implementation of Email Notification Module*

The code for the implementation of email notification module is shown in Figure 12.

```
from email.mime.multipart import MIMEMultipart
from email.mime.text import MIMEText
from email.mime.base import MIMEBase
from email import encoders
def send_email_to_user(receiver_email, sender_email,
email_pwd, email_subject, body, filename):
    msg = MIMEMultipart()
    msg['From'] = receiver_email
    msg['To'] = sender_email
    msg['Subject'] = email_subject


    msg.attach(MIMEText(body, 'plain'))


    attachment = open("/home/pi/webapp/"+filename, "rb").read()
    part = MIMEBase('application', 'octet-stream')
    part.set_payload(attachment)
    encoders.encode base64(part)

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

    server = smtplib.SMTP('smtp.gmail.com', 587)
    server.starttls()
    server.login(sender_email, email_pwd)
    text = msg.as_string()
    server.sendmail(receiver_email, sender_email, text)
    server.quit()
    print("Email notification successfully sent to user")
```

Fig. 12 Coding for the implementation of Email Notification Module

Based on Figure 12, the codes are used to implement the email notification module. The codes allow the system to send an email notification with the captured picture as attachment to the user's email when there is motion detected.

D. *Implementation of SMS Notification Module*

The code for the implementation of SMS notification module is shown in Figure 13.

```
from twilio.rest import Client

def send_sms_to_user(account_sid, auth_token,
receiver_phone, sender_phone, body):
    client = Client(account_sid, auth_token)
    message = client.messages.create(
        to=receiver_phone,
        from_=sender_phone,
        body=body)
    print(message.sid)
    print("SMS sent successfully")
```

Fig. 13 Coding for the implementation of SMS Notification Module

Based on Figure 13, the codes are used to implement the SMS notification module. The system sends a SMS notification to the user when motion is detected by the system.

E. *Implementation of Dropbox Storage Module*

The code for the implementation of Dropbox storage module is shown in Figure 14.

```
import dropbox

def upload_dropbox(filename,access_token):
        file_from = '/home/pi/webapp/'+filename
        file_to = '/Public/'+filename
        dbx = dropbox.Dropbox(access_token)
        f = open(file_from, 'rb')
        dbx.files_upload(f.read(), file_to)
        print ("Image uploaded successfully")
```

Fig. 14 Coding for the implementation of Dropbox storage Module

Based on Figure 14, the codes are used to implement the Dropbox storage module. This allow the system to upload the captured image to the user's Dropbox folder, creating an image library for all the captured images by the system.

F. *Implementation of Live Video Feed Module*

The live video feed module is implemented by installing the *Motion* library to the Raspberry Pi. This can be done by using the command *sudo apt-get install motion* at the Raspbian's terminal.

After installing the *Motion* library, a configuration is made to ensure the *Motion* is always running when the Raspberry Pi boots up. This is done by editing the */etc/default/motion* file. This is to access and edit the file to make further configurations. After gaining access to the file, the line *start_motion_daemon=no* is changed to *start_motion_daemon=yes*. Figure 15 shows the content of the */etc/default/motion* file after being configured.

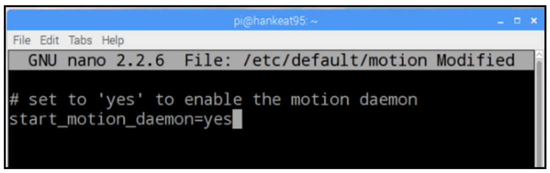


Fig. 15: The content of */etc/default/motion* file after changes

Next, the configuration file for the *Motion* is altered as well. To access the configuration file, the command *sudo nano /etc/motion/motion.conf* is used at the terminal.

There are a few configurations made in this file to make it suitable for this project. The changes are shown in Table 5 below.

TABLE 5
CHANGES DONE ON THE */ETC/MOTION/MOTION.CONF FILE*

Changes	Before	After
Video's width pixel	*width 352*	*width 320*
Video's height pixel	*height 288*	*height 240*
Video's auto brightness function	*auto_brightness off*	*auto_brightness on*
Video stream's framerate	*stream_maxrate 1*	*stream_maxrate 60*
Ensure stream not only limited to localhost connection	stream_localhost on	stream_localhost off

After making the changes, the command *sudo service motion restart* is entered to the terminal to restart the *Motion* service. The status of the *Motion* service can be verified by using the command *sudo service motion status*. Figure 15 shows the output of the command for checking the status of *Motion* service.

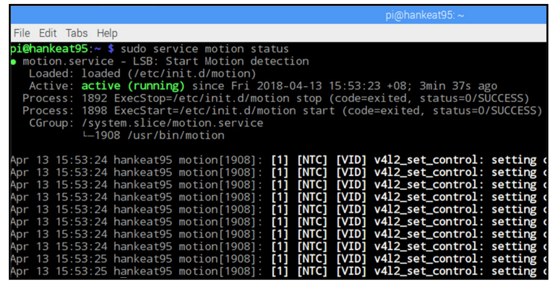


Fig.16: Output of the command *sudo service motion status* in the terminal

In Figure 16, the output status shows that the *Motion* service is active and running as it should be.

G. *Testing*

For testing this system, two testing methods are used. The testing methods are functionality testing and user acceptance testing.

1) Functionality Testing

The functionality testing for this system uses six test cases to verify the system's functionalities. The summary of the functionality testing is as shown in Table 6.

TABLE 6
RESULT FOR FUNCTIONALITY TESTING

No.	Test Scenarios	Expected Result	Actual Result	Pass/Fail
1	Motion detector able to trigger the web camera to capture image	Image of the intruder is captured	Image of the intruder is captured	Pass
2	SMS notifications have no significant delay	User is notified in less than 10 seconds after motion triggered	User is notified in 3 seconds after motion triggered	Pass
3	Email notifications have no significant delay	User is notified in less than 30 seconds after motion triggered	User is notified in less than 20 seconds after motion triggered	Pass
4	Cloud storage is storing every single image captured by the system	Yes	Yes	Pass
5	Live video feed has no significant delay	Yes	Yes	Pass
6	Motion detection and notifications are stopped on demand	Yes	Yes	Pass

Based on Table 6, six test scenarios are tested to justify the functionalities of the system. The actual results from the tests are compared with the respective expected results. If the actual results are equal or better than the expected results, then the test is considered as passed. From Table 6, all six tests have passed, meaning that the system's functionalities are performing as expected.

2) User Acceptance Testing

The user acceptance testing for this project involves five volunteer testers or respondents that test all the project's critical functions. The result from the five respondents obtained are summarized in Table 7.

TABLE 7
RESULT FOR USER ACCEPTANCE TESTING FROM FIVE RESPONDENTS

No.	Acceptance Requirements	Test Result (Number of people)	
		Accept	Reject
1	The system must execute to end of job.	5	0
2	The system is user friendly and not confusing.	5	0
3	The intruder is captured in the image.	5	0
4	SMS notifications are received.	5	0
5	Email notifications are received.	5	0
6	Live video feed is accessible.	5	0
7	Dropbox storage only stores captured images trigged by motion.	5	0
8	Live video feed is accessible on any browser.	5	0

Based on Table 7, there are eight acceptance requirements that are evaluated by the five respondents. The acceptance requirements are determined by the respondents when they use the system. The respondents can choose to accept or reject for each acceptance requirements based on their experience and observations when using the system. From Table 7, all the five respondents accepted every acceptance requirement. Therefore, the system is accepted by users with the acceptance requirements are fulfilled by the system.

VI. RESULTS

The results of the project are categorized according to the implemented module. The modules are SMS notification, Email notification and image capturing, live video feed, and image library in Dropbox.

A. *SMS Notification*

The SMS notification is sent to the user's mobile phone. The SMS notification is as shown in Figure 15.

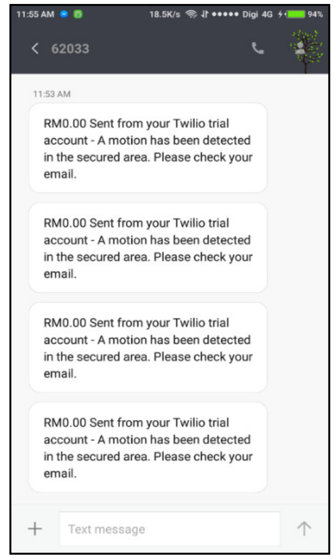


Fig. 17: SMS notification received on mobile phone

Based on Figure 17, the SMS notification alerts the user that motion was detected and request the user to check on the email for more information of the intrusion.

B. *Email Notification and Image Capturing*

The email notification is sent to the user's email. The email can be viewed from any electronic devices that supports the function of displaying email, such as a computer, tablet, smart phone or a smart TV. The image captured by the system's USB camera is attached to the email as well. Figure 17 shows the email that is viewed from a smart phone Gmail Application, while Figure 18 shows the email displayed from a computer's internet browser.

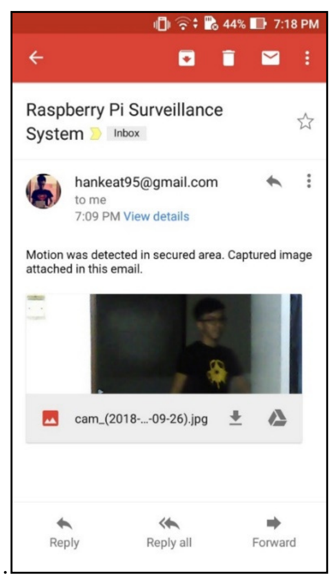


Fig. 18 Email notification viewed from a smart phone Gmail Application

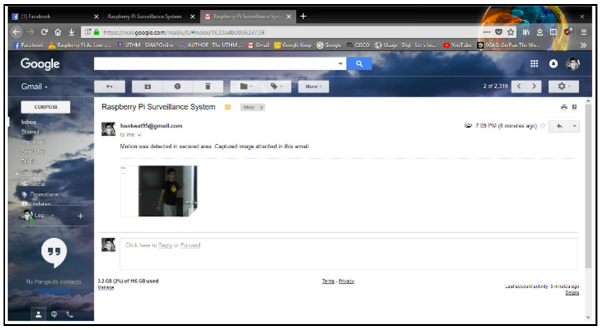


Fig. 19 Email notification viewed from a computer's internet browser

Based on Figure 18 and Figure 19, the email notification informs the user that motion was detected in the secured area monitored by the system. The captured image is attached to the email to allow users to view and verify the intrusion.

C. *Live Video Feed*

The live video feed can be viewed from any internet browser. Figure 20 shows the live video feed is viewed from a smart phone's internet browser, while Figure 20 shows the live video feed is displayed from a computer's internet browser.

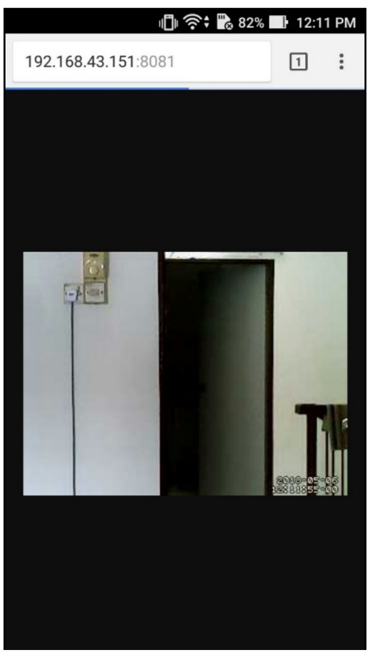


Fig. 20 Live video feed viewed from smart phone's internet browser

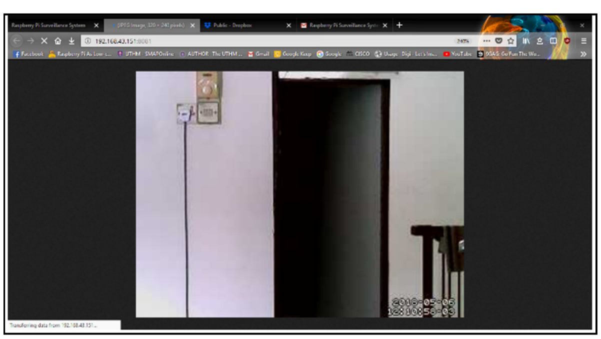


Fig. 21 Live video feed viewed from computer's internet browser

Based on Figure 19 and Figure 20, the live video feed allows the user to view the secured area monitored by the system at real time.

D. *Image Library in Dropbox*

The captured images by the system are stored in the user's Dropbox account. The images can be viewed anytime by the user. The image library can be accessed from any internet browser or the smart phone's Dropbox application. Figure 21 shows the image library accessed from the smart phone's Dropbox application, while Figure 22 shows the image library accessed from a computer's internet browser.

Fig. 22 Image library accessed from smart phone's Dropbox application

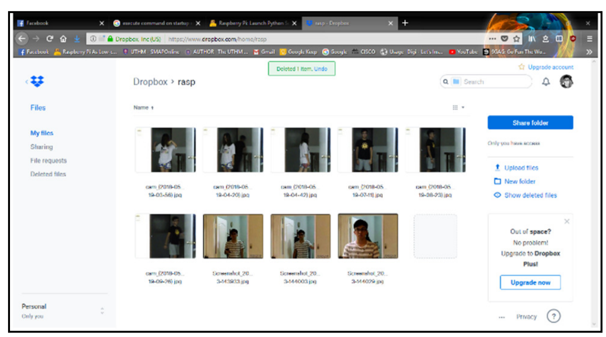


Fig. 23 Image library accessed from computer's internet browser

Based on Figure 22 and Figure 23, the image library provides the user the ability to access all the images captured by the system. Every pictures are named with the date and time of capture. This timestamp helps the user to sort the image files to ensure the images are organized properly.

E. *Sample Intrusion Detection*

An intruder detected by the system will send the user a SMS notification and an email notification. The captured image by the system is saved in the Dropbox cloud storage. The image is attached together with the email as well. Figure 24 shows the scene when the area is without intruders, while Figure 25 shows the image captured by the system when there is an intruder present in the area.

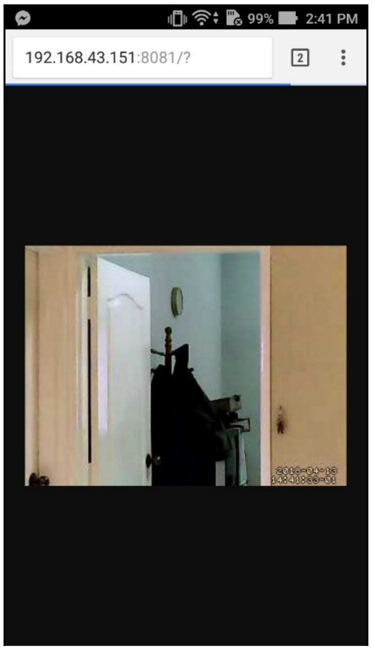


Fig. 24: Scene of the area without intrusion

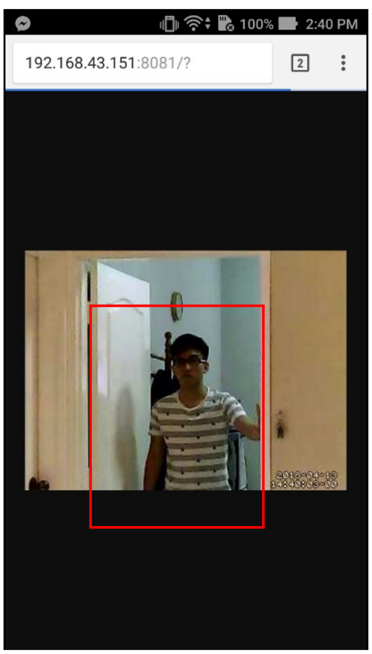


Fig. 25 Image captured by the system when intrusion is detected

Based on Figure 24, the figure shows an area of a door being monitor for intrusion, where intruder is not present. In Figure 25, the intruder (marked in red box) passed through the door and triggered the motion detector to capture an image.

VII. Conclusion

Throughout the whole process of developing the Smart Indoor Home Surveillance Monitoring System Using Raspberry Pi, the requirements and modules are determined and designed completely. This allow the development of the system to run smoothly and minimum errors or mistakes. This system has achieved the objectives while fulfilling the project's scope.

References

[1] Pi, R. (2013). *Raspberry pi. Raspberry Pi*, 1, 1.

[2] Richardson, M., & Wallace, S. (2012). *Getting started with raspberry PI*. " O'Reilly Media, Inc.".

[3] Upton, E., & Halfacree, G. (2014). *Raspberry Pi user guide*. John Wiley & Sons.

[4] Rao, P. B., & Uma, S. K. (2015). *Raspberry Pi home automation with wireless sensors using smart phone*. Int. J. Comput. Sci. Mob. Comput, 4, 797-803.

[5] Senthilkumar, G., Gopalakrishnan, K., & Kumar, V. S. (2014). *Embedded image capturing system using raspberry pi system*. International Journal of Emerging Trends & Technology in Computer Science, 3(2), 213-215.

[6] McManus, S., & Cook, M. (2017). *Raspberry Pi for dummies*. John Wiley & Sons.

[7] Maksimović, M., Vujović, V., Davidović, N., Milošević, V., & Perišić, B. (2014). *Raspberry Pi as Internet of things hardware: performances and constraints*. design issues, 3, 8.

[8] Ferdoush, S., & Li, X. (2014). *Wireless sensor network system design using Raspberry Pi and Arduino for environmental monitoring applications*. Procedia Computer Science, 34, 103-110.

[9] Dennis, A. K. (2013). *Raspberry Pi home automation with Arduino*. Packt Publishing Ltd.

[10] Ha, K. N., Lee, K. C., & Lee, S. (2006, October). *Development of PIR sensor based indoor location detection system for smart home*. In SICE-ICASE, 2006. International Joint Conference (pp. 2162-2167). IEEE.

[11] McConnell, S. (2004). *Code complete*. Pearson Education.

[12] Dempsey, John S. (2008). *Introduction to private security*. Belmont, CA: Thomson Wadsworth. p. 78. ISBN 9780534558734.

[13] Armitage, R. (2002). *To CCTV or not to CCTV. A review of current research into the effectiveness of CCTV systems in reducing crime*, 8.

[14] Kruegle, H. (2011). *CCTV Surveillance: Video practices and technology*. Butterworth-Heinemann.

[15] Park, J. O., & Kim, S. (2015). *Study on strengthening plan of safety network CCTV monitoring by steganography and user authentication*. Advances in Multimedia, 2015, 10.

[16] Keval, H. (2006, September). *Cctv control room collaboration and communication: Does it work?*. In Proceedings of human centred technology workshop (pp. 11-12)..

[17] Lee, J. V., Chuah, Y. D., & Chai, C. T. (2013). *A multilevel home security system (mhss)*. International Journal of Smart Home, 7(2).

[18] Poole, N. R., Zhou, Q., & Abatis, P. (2009). *Analysis of CCTV digital video recorder hard disk storage system*. digital investigation, 5(3), 85-92.

[19] Gerrard, G., Parkins, G., Cunningham, I., Jones, W., Hill, S., & Douglas, S. (2007). *National CCTV Strategy*. Home Office, London.

[20] Boghossian, B. A., & Velastin, S. A. (1999, September). *Motion-based machine vision techniques for the management of large crowds*. In Electronics, Circuits and Systems, 1999. Proceedings of ICECS'99. The 6th IEEE International Conference on (Vol. 2, pp. 961-964). IEEE.

[21] Design, O. O. (2009). *Object-Oriented Analysis and Design*.