RASPBERRY PI MOTION DETECTION SURVEILLANCE SYSTEM

Opoku Mensah Stephanie[1], Danso Juliana Mantebea[1,2], Gyamfi Enoch Opanin[1,3], Michael Asante[1], Kwame Ofosuhene Peasah[1], Rodor Gershon[4]

[1] *Department of Computer Science, Kwame Nkrumah University of Science and Technology (KNUST), Kumasi, Ashanti Region, Ghana*

[2] *School of Computer Science and Engineering, University of Electronic Science and Technology of China, Chengdu, Sichuan Province, China*

[3] *School of Information and Software Engineering, University of Electronic Science and Technology of China, Chengdu, Sichuan Province, China*

[4] *College of Electronic and Information Engineering, Nanjing University of Aeronautics and Astronautics, Jiangsu Province, China*

ABSTRACT

As an essential constituent of many associations' security and safety priority, video surveillance has established its importance and benefits numerous times by providing immediate supervising of possessions, people, environment and property. This project deals with the design approach of an Embedded Real-Time Surveillance System Based Raspberry Pi Section Border Controller (SBC) for intruder detection that reinforces surveillance technology to provide essential security to our life and associated control and alert operations. The proposed security solution centers on our novel integration of cameras and motion detectors into web application. Raspberry Pi operates and controls motion detectors and video cameras for remote sensing and surveillance, streams live video and records it for future playback. This research is focused on developing a surveillance system that detects strangers and to respond speedily by capturing and relaying images to owner based wireless module. This Raspberry Pi based Smart Surveillance System presents the idea of monitoring a particular place in a remote area. The proposed solution offers a cost-effective ubiquitous surveillance solution, efficient and easy to implement. This project will also present the idea of motion detection and tracking using image processing. This type of technology is of great importance when it comes to surveillance and security. Live video streams will therefore be used to show how objects can be detected then tracked.

Keyword: - ***Embedded real-time surveillance intruder detection, Camera and motion detection, Facial recognition, Raspberry-pi SBC, PIR sensor, CCTV, Raspbian Operating System***.

---

1. INTRODUCTION

In our world today, security has become a key component since in our busy life we do not have sufficient time to monitor our properties and assets. Thus, the demands on video surveillance systems are rapidly increasing in the present day. One of the principal things people will want to know about their surveillance system is regardless of whether they have the ability to interface with it over the internet for remote viewing. In the past, security systems had to be supervised by a guard who was secured away a room throughout the day watching the screens to ensure that nothing would happen [1]. The. The other option was to come back and review the footage but damage could have happened. Along these lines, researchers and scientists needed to think of methods for overcoming that and accordingly enhancing security at large. Business spaces, colleges, healing facilities and distribution centers require video capturing systems that have the ability to alert and record beside live video streaming of the intruder. The progressions in video surveillance technology has made it possible to see your remote surveillance camera from any internet-enabled PC or smartphone from anywhere in the world. This includes the use of CCTV frameworks and IP cameras [2, 3].

This technology is awesome yet its cost of usage has ended up being an obstacle particularly for a small home application. Therefore, new innovative technology revolves around affordability of a product as far as its cost and simplicity of usage. The Raspberry Pi crosses both criteria in that it is a cheap, effective computer which can be interfaced with other modules to realize systems with immense functionality [4, 5]. A great deal should be possible on it ranging from motor speed control, automatic lighting, VPN server, security system and so on. The latter is of great interest in this project. The Raspberry Pi microcomputer is equipped for actualizing a cost-effective security system for various applications. This new emerging technology related to security provides a comfortable and safe environment for small homes. The various objectives of the system are to detect an intruder, take an image of the intruder and also convey an alert message to the facility owner. In doing so it thus allows for remote monitoring of homes from anywhere in the world. The system to be designed cannot wholly replace the role of CCTV and IP surveillance cameras especially in large commercial set ups but will make it easy for low income home owners to monitor their homes at a very affordable price [6].

In addition to the fact that the Raspberry Pi board is cheap, the camera to be used in this case is relatively cheap compared to the others. The whole security system circuitry is simple and easy to implement. Image processing is a term which indicates the processing on image or video frame which is taken as an input and the result set of processing may be a set of related parameters of an image. The purpose of image processing is visualization which is to observe the objects that are not visible. Analysis of human motion is one of the most recent and popular research topics in digital image processing [2, 3]. In which the movement of human is the important part of human detection and motion analysis, the aim is to detect the motions of human from the background image in a video sequence. It also includes detection and tracking. The process of object tracking is segmenting a region of interest from a video frames and keeping track of its motion and position [7]. The proposed home security system captures images and transmits it via a 3G Dongle (or wi-fi hotspot to the pi) to a Smart Phone as an email notification and allows user to stream live from anywhere with enough internet access using web application Raspberry pi. A log file consisting of accumulated daily information of what happened in the user's absence is recorded and sent via email as well as stored in a Github repository for daily analysis.

1.1 Problem Statement

Starting from small houses to huge industries, surveillance plays very vital role to fulfill our safety aspects as Burglary and theft have always been a problem. In big industries personal security means monitoring the people's changing information like activities, behavior for the purpose of protecting, managing and influencing confidential details [8]. Surveillance means watching over from a distance by means of electronic equipment such as CCTV cameras, but it is costly for normal residents to set up such kind of system and also it does not inform the user immediately when the burglary happens [9, 10]. The current study develops a fully automated machine to machine communication to solve tis prevalent cost problem of surveillance CCTV technologies. The need to develop a cost-effective surveillance system through innovative technology immensely influenced the development of this project. This project will design and implement a security system based on Raspberry Pi microcomputer. The system should be able to detect motion (intruder), activate a camera to take frames of video after motion is sensed and then send an alert to the facility owner through electronic mail plus an image attachment. The cost of installation of any security system depends on several factors [10].

First, the type of camera being used is of great consideration. Another aspect of this project is to present an idea of monitoring and tracking of an intruder through the use of a camera. Any object passing through the field of view of camera will be detected then tracked in case the object attempts to move anybody part. This research also seeks to contribute to understanding video surveillance and the surveilling parts assemblage. The orienting concept of the surveilling parts assemblage is used to theoretically ground the research. The surveilling parts assemblage works through a process of abstracting the parts of the surveillance system and categorizing them into discrete functional flows and then reassembled into a single large operable unit. Later, these single unit will be use to survey the main targets of intervention. This study assembles a surveillance CCTV unit using Raspberry pi and other technological components. In surveillance, CCTV camera is costly because of the use of computer. It reserves too much space for continues recording and also require manpower to detect the unauthorized Activity.

But compared to the existing system Raspberry pi system is much cheaper with better resolution and low power consumption feature. The proposed CCTV surveillance system has a very simple circuit structure that offers a lot of privacy. The operating system used here is Raspbian OS. Just because all of those weak points of the surveillance system, an energy efficient portable system is proposed, that can take pictures when the burglary happens and send out an alert signal at the same time enable live streaming of frame sequence of actions at precise time. This is much better than the current in use surveillance systems. It is simple to implement, small size portable stand-alone device with its own power source, energy capable with instantaneous alert, truly cheap for residential and personal use. Suppose owner place a camera at one particular area which is to be monitored for security purpose, if user wants to go to remote location and still wants to monitor that same room for security, then he can make use of remote surveillance system by use of mobile through internet facility. However, this can be used or can prove useful to a project targeting security setup which is limited to specific location to monitor properties and assets from a diverse location

1.2 Research Contributions

The main contribution of this paper is to design and develop a security system that includes features such as motion detection, image processing and emailing to facility owner. The system is to be based on Raspberry Pi 3 SBC. Specifically, the contributions of this paper can be broken down as follows:

- To study and describe how the Raspberry Pi can be interfaced with a motion detector (Passive Infra-red) and Pi camera.
- To study how a Raspberry Pi can be programmed so as to be able to send an email to a prescribed mail hub.
- To develop and build a prototype of the surveillance system based on the Raspberry Pi SBC.
- To design and implement a motion detecting and tracking system for real time video analysis

2 REVIEW OF RELATED LITERATURE

In Network video capture system using friendly ARM9 board support package (BSP) S3C2440 is presented. This application system captures video, shares among networked systems and also alerts the controlling person with short message service alarm as required by the client. This system works in a real time environment and is supported by embedded RT Linux. This system provides low cost and high effective intelligent monitoring system like in elevators, home security systems etc. with low power consumption [12]. Here RT Linux is used i.e. Real time operating system and also it can be wired or wireless internet access this could be its advantage. It alerts the person through the Short Message Service (SMS). Video capture is realized by the Video 4 Linux. This system has advantages such as higher intelligence, higher stability, and easy installation and disadvantage as it requires high cost and continues GSM network is required to send message, if the network is not available this system may not work properly [13]. The embedded Real-time video monitoring system based on ARM is designed, in which the embedded chip and the programming techniques are used. The central monitor which adopts Raspberry pi is the core of the whole system. Real time video transmission is widely used in surveillance, conferencing, media broadcasting and applications that include remote assistance. First, USB camera video data are collected by the embedded Linux system. All data processed compressed and transferred by the processing chip. Then, video data are sent to the monitor client by wireless network. This embedded monitoring system to overcome the weak points of the traditional video surveillance systems, such as complex structure, poor stability, and expensive cost. It can be widely used in many fields, and also used for long distance transmission [14].

Gantt [13], Deshmukh and Shinde [15] and Zhang, Srinivasan and Ganesan [16], all worked on implementation of a low-cost wireless remote surveillance system using Raspberry Pi. Conventional wireless CCTV cameras are widely used in surveillance systems at a low cost. Gantt [13] implemented a low cost and secure surveillance system using a camera with Raspberry Pi and the images acquired have to be transferred to the drop box using a 3G internet dongle. This was successfully implemented using Raspberry Pi and 3G dongle. Buenger [17] and Sanjana Prasad [18] worked on developing a smart surveillance system based on SBC of Raspberry Pi and motion detector sensor PIR. Their development boosts the practice of portable technology to offer vital safety to our daily life and home security and even control uses. The objective of their research is to develop a mobile smart phone home security system based on information capturing module combined with transmitting module based on 3G technology fused with web applications. The SBC will control the PIR sensor events and operates the video cameras for video streaming and recording tasks. Their system has the capability to count number of objects in the scene [19, 20]. Buenger [17] also evaluates the use of various sensors, wireless module, microcontroller unit and finger print module to formulate and implement a cost-effective surveillance system. He and his team [17] adopted an ARM core as a basis processor of the system. PIR sensor is used to detect motion in the vision area, while vibrating sensor is used to sense any vibration events such as sound of breaking. The intruder detection technique is proposed by using the PIR sensor that detect motion and trigger a system of alerting and sending short message service through GSM module for a specified phone number. Their work can be featured by adopting numerous diverse kinds of demanding database and thus it will be more secure and difficult to hack.

There are already several systems - both commercial solutions and hobbyist-developed – that include some of the functionalities the system herein described will implement. These have a wide range of applications, from security to sports and outdoor leisure. In this chapter, we analyze some of these systems, their pros and cons, and the feasibility of their use cases on the described system.

2.1 Security Video-Surveillance

Most video surveillance systems found in the market, sold both in retail and by specialized security companies, are typically composed of three components: a set of cameras, a central unit with Digital Video Recorder (DVR) capabilities and some sort of software interface for viewing and managing footage. The cameras are usually fixed (some might include a motor so that their orientation can be remotely controlled) and connected to a wall power outlet; each camera watches a determined physical area. The central unit includes a high-capacity hard disk. This unit stores the video taken from each camera in its hard-disk and keeps it for a set period of time before old recordings are deleted and replaced with new ones [6]. The central unit is connected to every single camera by cables and usually allows a connection to a TV or other type of screen for live video watching. These systems have a number of limitations, the first being lack of practicality. If it becomes necessary to install a new camera or relocate an existing one, the area to be covered needs to have a power source of some sort for the camera to work. Furthermore, cables also need to be relocated or, if the available ones are not long enough, new ones need to be ordered and installed [21]. The second limitation is image quality: since many of these systems are very low cost, imaging is often a sacrificed aspect in order to cut the price, and cheap camera sensors and lenses mean that image quality and definition may be poor in low-light areas, which can, for example, make the identification of a burglar impossible and thus make the system ineffective in its purpose. The last drawback is lack of flexibility. Since these systems are purely focused on security applications (rarely performing more than simple video capture and motion detection), and their software platforms are most times proprietary, it means the hardware - though technically capable - is limited to such tasks, not being adaptable to others like over-time data collection. The Swann 3425 Series, as shown in Figure 1, is an example of this type of system [22].

Fig -1: Swann DVR8-3425 - an example of a video surveillance system with a set of cameras and a central unit.

2.2 Sports Data Gathering

Over the last few years, a new field in sports has been gradually catching the interest of coaches, players and fans in order to analyze games in a deeper level. By applying image recognition methods to a game, it becomes possible to gather a whole set of data on player performance over a match. Pointing one or more cameras at a pitch and then making use of image recognition algorithms allows investigators to collect a wide amount of data on the game being played. The first example, as shown above, is the generation of the heat map of a player's performance. If the displayed map represented a midfield player and his team attacked from the left to the right, it can be concluded that he is too close to his own goal and not providing enough support to his attacking team mates. Individual heat maps can be generated for each player, and a general one for the whole team. Another example is the calculation of the distance a given player had to cover during a match. Such a calculation can potentially be made by pointing a camera at the pitch, identifying the player in question and then, knowing the dimensions of the pitch, calculate the covered distance using trigonometry. This can allow coaches to point out that a player is running too much and might be fatigued sooner than his team mates. Such examples may need to employ moderately complex algorithms.

In order to build a player's heat map, the system needs to be able to identify the area of the video pertaining to that player and automatically follow it around the pitch [23]. A potentially more complicated issue is player identification: how to know which area of the video feed corresponds to each player. Facial recognition algorithms need to be applied or, alternatively, the previous identification of each player, in which case the manual work of a human is necessary - possibly at the beginning of each match part, since the players leave the pitch (and consequently the video feed) during each break. Figure 2 depicts this scenario.

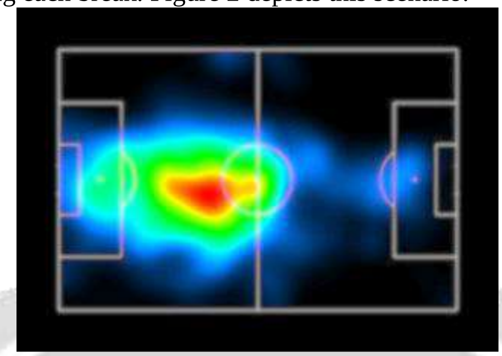


Fig -2: Heat map generated by a player in a football match, showing the areas of the pitch he spent the most time in.

2.3 Wilderness Cameras

A different application of a camera when connected to an embedded device are the so-called wilderness cameras like Bushnell^{TM} [24]. These are based on a weather-proof case, enclosing a camera, a motion sensor, a memory card for local storage and an embedded system. When installed outdoors, the motion sensor continuously looks for movement and, when it is detected, the camera takes a snapshot or a few seconds of video. The purpose of this behavior is to capture footage of wildlife that would be near impossible to obtain with human presence. Some models improve functionality with HD camera modules, GPS tagging of files, LEDs for lighting or infra-red night vision [24, 25]. They are powered by a battery. These systems are architecturally very simple and limited in their use, but they are also very well-tailored for it. The presence of a dedicated motion sensor [26] is one such indication: instead of having the camera permanently turned on and continuously analyzing its footage, the low-power sensor allows for the same effect to be achieved with a big increase in battery life. However, these systems are also closed platforms - there is no way to alter them in order to perform additional functions, like triggering a remote notification when movement is detected.

2.4 Human Monitoring Systems

Some systems are available whose goal is to collect data about the usage of public spaces, like stores or supermarkets. A set of cameras is installed around the space to properly cover the areas opened to the public and then run continuously while the space is being used. A set of techniques applied over the video feed can help gather data about the captured footage: heat maps can be calculated over a period of time, to assess the areas with the most, the less and no movement at all. Human recognition algorithms can recognize where a person is in the frame and treat that data accordingly, including the already mentioned calculation of heat maps [6]. The use of these techniques allows a wide range of data to be collected about the video feed. The calculation of the heat map of a particular area tells the zones of that area that saw the most movement - making it easy to identify the zones of the space that people visited or stopped most often, either to look at exposed products or marketing material. A camera looking at the checkout lanes can show if people prefer self-checkout machines or conventional ones with a human operator. Highway operators can discern whether people prefer to pay tolls with vending machines or human operators. Crossing this data with date and time shows how these affect all these factors (allowing, for example, for store operators to schedule employee shifts accordingly) and by integrating the system with the payment system it would allow for yet more conclusions to be drawn on average transaction value over time. Commercial solutions of this type already exist: BVI Network [27] as shown in Figure 3 and other like [28, 29, 30], are some of such examples.

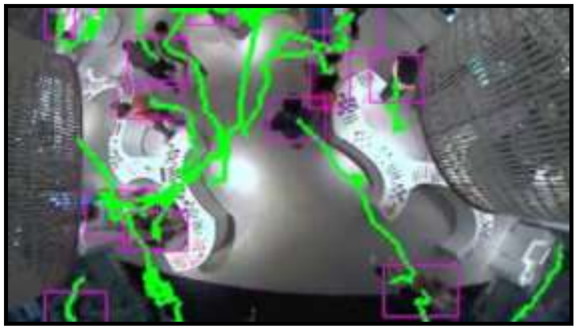


Fig -3: Human tracking by: BVI Networks. Note the humans enclosed in purple boxes and, in green, their pathway.

3 METHODOLOGY

The research approach used in this project is a methodological study of both specific techniques and techniques developed by others in the discipline of video surveillance. Rapid prototyping model was used to develop a prototype of the study to demonstrate the various architecture of the system. The various phases of prototyping were implemented in this project. Prototyping is intended to make a model of a system or part of a system that is developed to check the requirements and feasibility of the proposed solution. The model begins with establishing the prototype objectives and continues to defining prototype functionality, developing the prototype and evaluating the prototype. Prototyping is not a standalone, complete development methodology, but rather an approach to be used in the context of a full methodology.

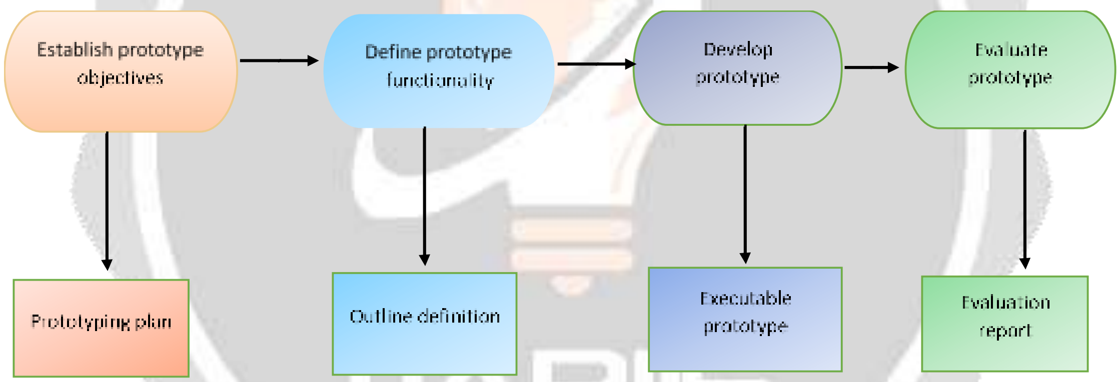


Fig -4: A Prototype Model

For this project, the prototype objectives are to develop a cos efficient system to be available to be used in homes and in big as well as small businesses for surveillance purposes. It also involves interfacing the raspberry pi SBC with passive infra-red motion detector and camera module. For the detailed design, it comprises of a wired on-board of all the components that includes the PIR sensor connected to the pi using jumper cables from the bread board to the General-Purpose input/output (GPIO) pins, two set of cameras (a USB camera and pi camera), one for capturing image frames and the other for live streaming. The following requirements are needed to set the pi running for surveillance. An SD card of at least 8GB is needed for better performance, an internet enabled environment and for remote viewing installation of VNC client and server software, the server on the pi and the client side on the laptop or any smart device capable of viewing the GUI of the raspberry pi. Figure 4 shows the prototyping model for this study

3.1 System Design

Over here, system design refers to the process of defining the architecture, components, modules, interfaces, and data for a system to satisfy specified requirements. This methodology section mainly focuses on the flowchart and user interface design. The research design used for this study is the prototyping which is a sample implementation of the system. It provides limited and main functional capabilities of the proposed system.

Algorithm 1: Proposed intruder motion detection

Input: Image camera sensors or feed, Passive Infrared (PIR) Sensor
Output: Email content (Zipped folder of captured images)
Begin

1. Power on the Raspberry Pi System
2. Continuously checking if motion is detected or not.
3. If no it goes back to step 2.
4. If yes it starts capturing the image frames.
5. Saves the image in a folder which is zipped and sent through email.
6. The zipped file is deleted to create space for the next capture
7. A notification is sent to the authenticate person for viewing
8. The authenticate person can also do a live video stream of the surveillance area
9. A log file of daily activities is stored and sent through email and also stored in GitHub repository

End

3.2 Flowchart

This following flowchart was used to design and thus document the security systems project. It illustrates the series of events starting from intrusion event up to the point when it sends out an alert. This algorithm was implemented using a Python script. Figure 5 below presents the basic flowchart of the entire system.

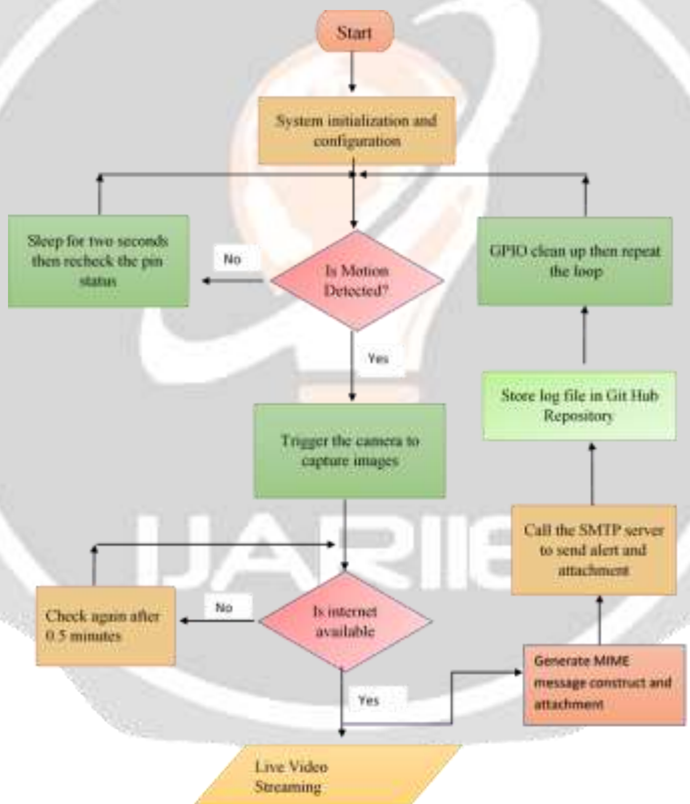


Fig -5: Flowchart of the Proposed System

3.3 Hardware Modules Set Up and Configuration

The entire system modules consist of seven hardware parts components which are 1) Raspberry Pi 3 controller, 2) PIR motion sensor, 3) RJ45 Ethernet connector/WI-FI connection, 4) Waveshare fish eye lens camera module, 5) MicroSD card, 6) LED and 220 Ohms resistor and 7) USB powered cable

3.4 Booting Up the Pi Model

The Raspberry pi model like every other system runs on the basis of a software known as the operating system. The operating system that was installed onto the SD card was downloaded from the Internet. It is much easier to download NOOBS (New Out of Box Software) operating system installation manager for the Raspberry Pi, since it combines the Raspbian OS with other utilities and libraries for easy installation. This was a zip file that was then extracted to an image file. Raspbian 'Wheezy' image was written into the 8GB Micro SD card. This was the operating system chosen to run on the Pi because the OS has been optimized and ported to the Raspberry Pi ARM architecture. This OS has very good integration with the hardware and comes preloaded with a GUI and development tools. After slotting in the Micro SD card and either connecting RJ45 Ethernet cable or wi-fi connection to the Pi and the personal computer with Putty software (Putty is a terminal client used to remotely access and control the Pi from computer running on Windows platform) the system was powered. Putty was then started and the default static IP address of the Pi was typed into the host name field. While doing this, windows pc was set to manual IP configuration. This was to allow it communicate with the Raspberry Pi. For the purpose of this project, we adopted Ubuntu which is compatible with the Raspbian OS of the pi hence the pi can be accessed using either Secure Shell (SSH) or Virtual Network Computing (VNC) remotely.

The following are stages involved in the installation and configuration of the Raspbian OS on the Raspberry pi:

Stage 1: Downloading the Raspbian OS using NOOBS

As shown in Figure 6, the operating system used for raspberry pi is a Raspbian OS, and it needed to be downloaded from the internet. For beginners and newbies, NOOBS is recommended for easy installation. It comes along with all the necessary libraries and utilities. This is available for free and can be downloaded from the Raspberry-pi website [4, 5, 31].

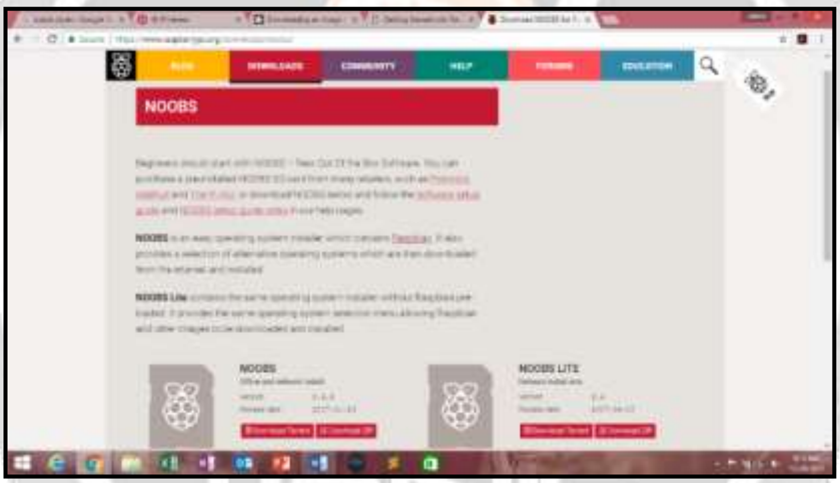


Fig -6: Downloading the New Out of Box Software

Stage 2: Format the SD Card

To install Raspbian, you will need an SD card that has 2 GB of space or more. If you have not already, format the drive to FAT-32. Figure 7 shows the interface that is presented when Raspbian OS prompts to format an SD card for the intended work.

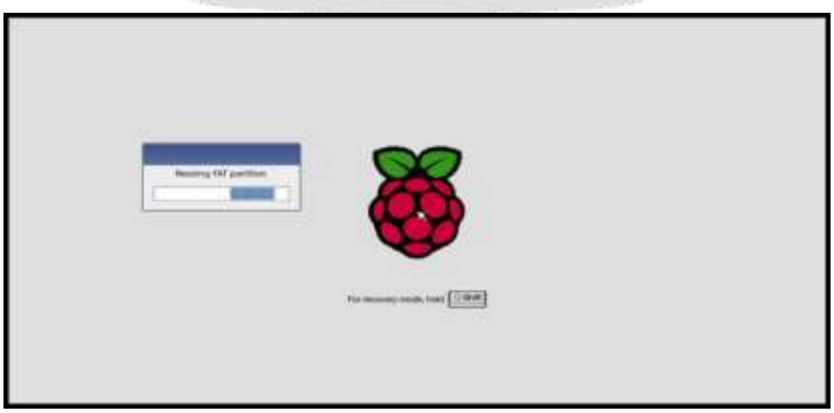


Fig -7: Formatting the SD Card

Stage 3: Setting up the Raspberry pi

Once you've flashed the disk image, place the SD card into your Raspberry Pi, plug in the HDMI monitor, any keyboards and mice, and then the power cable. Your Raspberry Pi should begin to boot and you should be able to see Raspbian on your screen. Figure 8 presents the screenshot interface for this process.

Fig -8: Setting up the Raspberry pi

Stage 4: Selection of the Raspbian OS

It will power up and will start running when you plug it in. A menu will appear, click on Raspbian and follow the instructions. Raspbian will be installed shortly and you will see desktop. This is as shown in Figure 9.

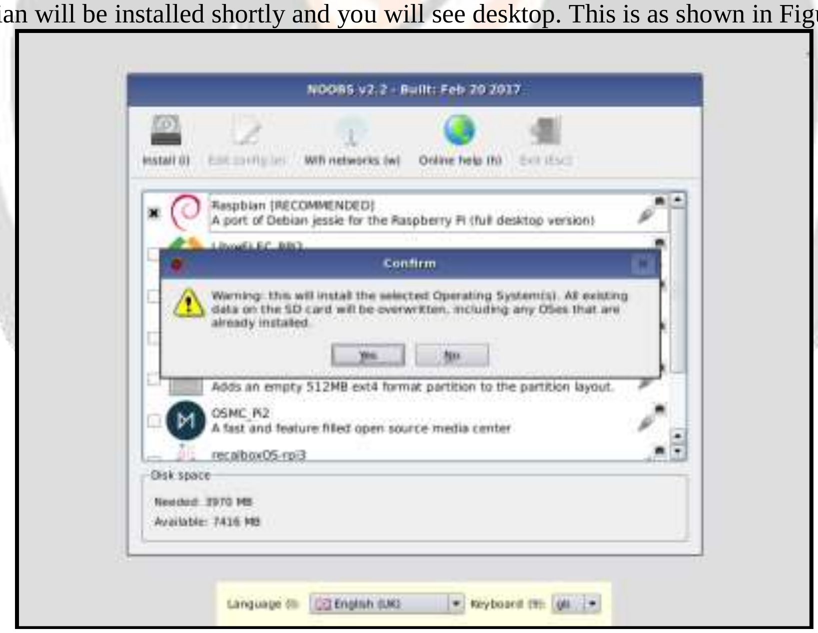


Fig -9: Selection of the Raspbian OS

Stage 5: Progress of installation

Wait for some few minutes whiles it continues installing. If it asks for username and password use these defaults or you can as well set your own for security reasons: 'Username: pi'. 'Password: raspberry'. Figure 10, shows a 12% progress of installation. This depicts the above steps have been successful so far.

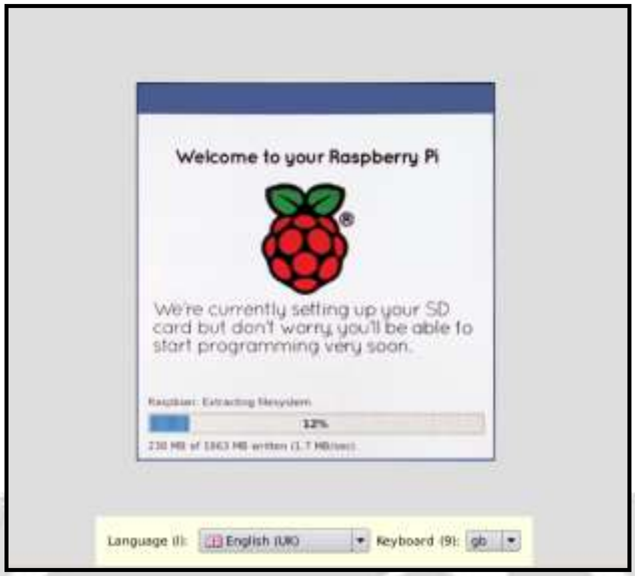


Fig -10: Progress of installation

Stage 6: Booting the Raspberry pi

You are now successfully running Raspbian!! In Figure 6, a screenshot that depicts a successful booting of the Raspbian OS is shown.

Fig -11: Booting the Raspberry pi

Stage 7: System Configuration

When the first boot is complete, you will be greeted with the configuration tool. There are a bunch of options here, some of these are useful others are useless for a beginner.

- Expand Filesystem: You should enable this (press enter). It will allocate all of the SD card's space to the OS.
- Change User Password: This is to change the default password (default password is 'raspberry'). You should change that in case you want to take up a project that might need stronger security (e.g. web server, home automation/security).
- Enable boot to: This lets you choose whether to boot into the desktop environment, scratch or the command line (default). Since most of the projects make use of the command line only you should leave it as is for now. You can go to the desktop environment from the command line by running the command startx.

- Internationalization Options: If you want to use a language other than English it can be changed in here along with keyboard layouts.
- Advanced options:
    - SSH: You should enable this even if you don’t know what it is right know. Most projects use it and it is a very useful tool in general since it lets you control all the aspects of the Pi without a dedicated display and keyboard and mouse.

Figure 12 show the screenshot of how the meu items of Raspbian OS is presented on the computer screen.

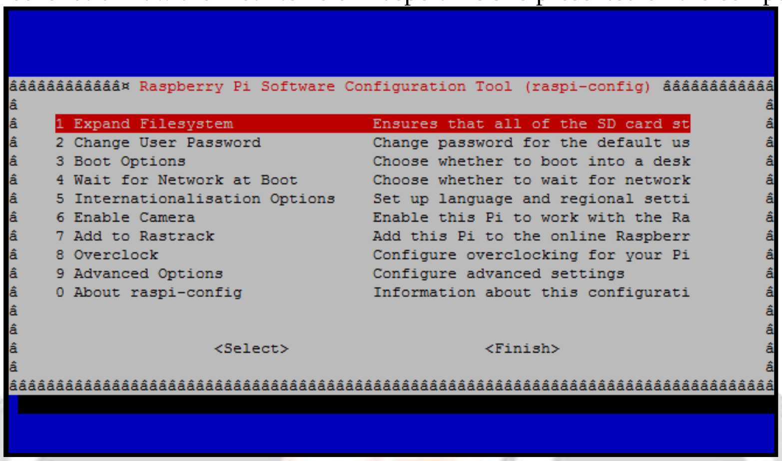


Fig -12: System Configuration

Stage 8: Setting Up internet connection on the pi

Internet was very necessary so that the pi can communicate over a set of internet protocols in other to allow the installation of necessary python packages. If you had plugged in a LAN cable then you should be now connected to the net but if you used a Wi-Fi dongle then it needs to be configured. This can be done from the command line itself but it is easier from the desktop environment. Type the following command to boot into the desktop environment.

```
startx <package>
```

- When you are successfully booted, click on menu in the upper left corner, then on preferences and select Wi-Fi Configuration.
- In the window that pops up select scan.
- A list of available networks should come up, double click the one you want to connect to.
- Enter your password in PSK textbox and click on the add button.

You should now be successfully connected to the World Wide Web. You now set-up and ready to go but a last step recommended to do before proceeding to update the software. You can do so by typing the following commands into the terminal or the command line (press enter after each line).

```
sudo apt-get update sudo apt-get upgrade
```

For live video streaming of the surveillance area remotely we used a web application which is known as remot3.it. This application allows you to host the services of the Raspberry pi over the internet for remote access. These services can be accessed using SSH, VNC or HTTP. Below is Figure 13 that shows the used architecture of the ‘rEmot∃.it’ configuration.

Fig -13: ‘rEmot∃.it’ configuration architecture

3.5 Enabling the PI Camera

This is the original camera that was made specifically for the raspberry pi. It was hooked on to the raspberry pi through the CSI-2 electrical port which is an extremely fast port. To configure and enable the camera, the following commands were executed at the CLI of the raspberry pi:

```
sudo apt_get update

sudo apt_get upgrade

sudo raspi_config
```

After the configuration settings, the system was rebooted to save all changes. This was done to make sure that the camera was allocated enough space in memory. The camera takes 5MP image and has a resolution of 1080 by 890. And to ensure that the camera was well configured and functional, the following command was executed.

```
sudo raspistill -o image.png
```

3.6 Setting up the Passive Infrared Sensor

This was used to control the entire system. The device used here was the Hc-SR501 passive infrared sensor. The detection range is 7 meters by 140 (degrees) coning angles. It provides an output of HIGH when a human body is detected within its range and an automatic delay LOW when the body leaves its range. The delay time is adjustable using the potentiometer on-board. The minimum delay time that can be set is 5 seconds and maximum of 200 seconds. The blocking time of the potentiometer is also provided on-board to set the blocking time during time during which the sensor does not respond to any change in motion. It was powered directly from the Pi through the 5v dc supply pin. Its output was connected as the input to the programmable GPIO pin.

4 HARWARE COMPONENT DETAILS

This section explains some of the hardware components that were used to develop the proposed.

4.1 Raspberry-pi

The raspberry pi is a low-cost single board, packing considerable computer power in a size of a credit card. The raspberry pi board contains many features like camera connector, Ethernet port, GPIO pins for interfacing sensors and switches, USB ports to connect to external devices like keyboard, mouse, Wi-Fi adapter etc., HDMI port to interface to monitors like LCD systems, projectors, TVs etc. and ab audio jack also available. By all these embedded on a single board. The Raspberry pi has no internal mass storage or built-in operating system and hence it requires an SD card preloaded with version of Linux Operating system. All models of Raspberry pi include an ARM compatible CPU and on-chip GPU. More detailed technical specifications of the Raspberry-pi are as follows [4, 5]. First, the hardware module was a 1.2GHz 64-bit quad-core ARMv8 CPU with an 802.11n Wireless LAN property. The capacity of the RAM of the module is 1GB (BLE). It has two USB ports of version 2.0 and 40 GPIO pins. The physical module was having a full HDMI, an Ethernet port, a combined 3.5mm audio jack and composite video camera interface (CSI), a micro SD card slot with a 10W, equivalent to a 2A power. Below is a figure of the raspberry pi 3 model B with 40 GPIO pins soldered to the board. A graphical representation of the description above is shown in Figure 14 (a).

4.2 Waveshare Fish Eye Lens Camera Module

The Camera Board on the Raspberry Pi is a small printed circuit board with a camera on it. The PCB is connected to a ribbon cable which connects to the Pi itself on its own port. The ribbon can be extendable. The camera on the board is very small (5MP camera). Since it uses 250mA, externally powering the Pi should be sufficient enough for the camera. Specific configuration settings are required to initialize the camera plus a Python script to enable it take pictures. This time it's a fish-eye variant of their camera, which gives a wide panoramic view - 222 degrees to be exact. It's available in normal and IR versions, making night-vision possible. If you need to capture more in your shots, for a project such as Pi CCTV or similar, this fish-eye lens could be just the job. This whole idea was deduced from Tutorials Point [32]. A graphical representation of the description above is shown in Figure 14 (b)

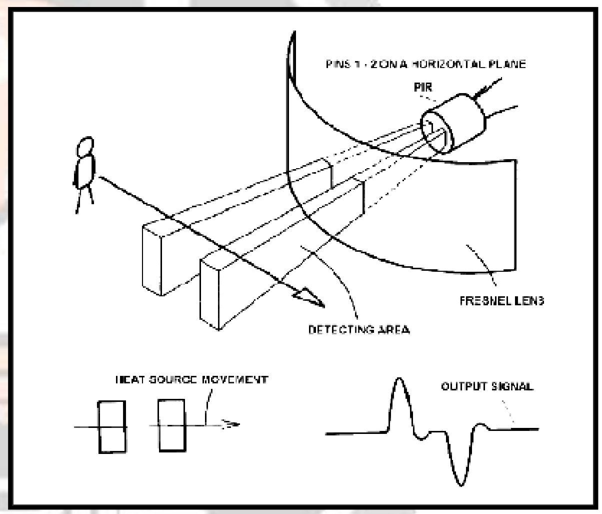
4.3 Passive Infrared (PIR) Sensor

This section explains some of the principle guiding the operation of a PIR Sensor. An individual PIR sensor detects 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 human, 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. The following Figure 14 (c) depicts how the PIR Sensor works

Connecting PIR sensors to a microcontroller is really simple. The PIR acts as a digital output so all you need to do is listen for the pin to flip high (detected) or low (not detected). It is likely that you'll want retriggering, so be sure to put the jumper in the H position! Power the PIR with 5V and connect ground to ground. Then connect the output to a digital pin, that is. the GPIO pin of the RPI device. A C/Python code can then be used to read a channel from the PIR sensor.

(a) Raspberry pi 3 model B

(b) RPI camera fish eye lens



(c) Operation of PIR Sensor

Fig -14: Hardware component (a) Raspberry pi 3 model B (b) RPI camera fish eye lens (c) Operation of PIR Sensor

5 SOFTWARE COMPONENT DETAILS

This section explains some of the software tools that were used to develop the proposed.

5.1 TightVNC

TightVNC is a cross platform free and open source remote desktop software application. It was developed using and extending the RFB protocol of virtual network computing to allow end users to control another computer’s screen remotely. TightVNC uses so called “tight encoding” of areas, which improve performance over low bandwidth connection. It is effectively a combination of the JPEG and zlib compression mechanisms. It is possible to watch videos and pay DirectX games through TightVNC over a broadband connection, albeit at a low frame rate. TightVNC includes many other common features of VNC derivatives, such as file transfer capability. Figure 15 shows a screenshot of the TightVNC interface.

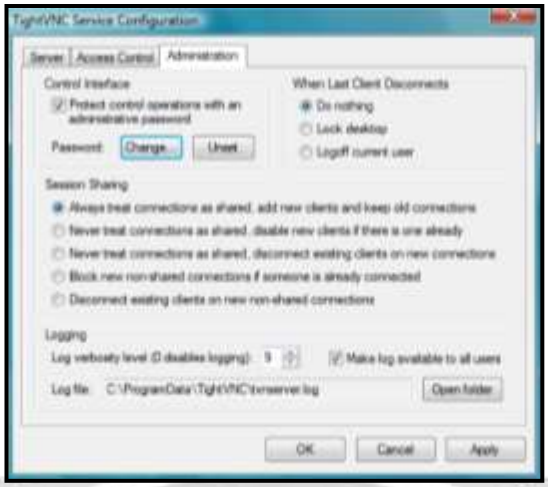


Fig -15: TightVNC Service Configuration Page

5.2 VNC Viewer

In computing, Virtual Network Computing (VNC) is a graphical desktop sharing system that uses the Remote Frame Buffer protocol (RFB) to remotely control another computer. It transmits the keyboard and mouse events from one computer to another, relaying the graphical screen updates back in the other direction, over a network. VNC is platform-independent – there are clients and servers for many GUI-based operating systems and for Java. Multiple clients may connect to a VNC server at the same time. Popular uses for this technology include remote technical support and accessing files on one's work computer from one's home computer, or vice versa. VNC was originally developed at the Olivetti & Oracle Research Lab in Cambridge, United Kingdom. The original VNC source code and many modern derivatives are open source under the GNU General Public License. There are a number of variants of VNC which offer their own particular functionality; e.g., some optimized for Microsoft Windows, or offering file transfer (not part of VNC proper), etc. Many are compatible (without their added features) with VNC proper in the sense that a viewer of one flavor can connect with a server of another; others are based on VNC code but not compatible with standard VNC. Figure 16 shows a screenshot of the GUI interface of Raspberry pi using the VNC viewer

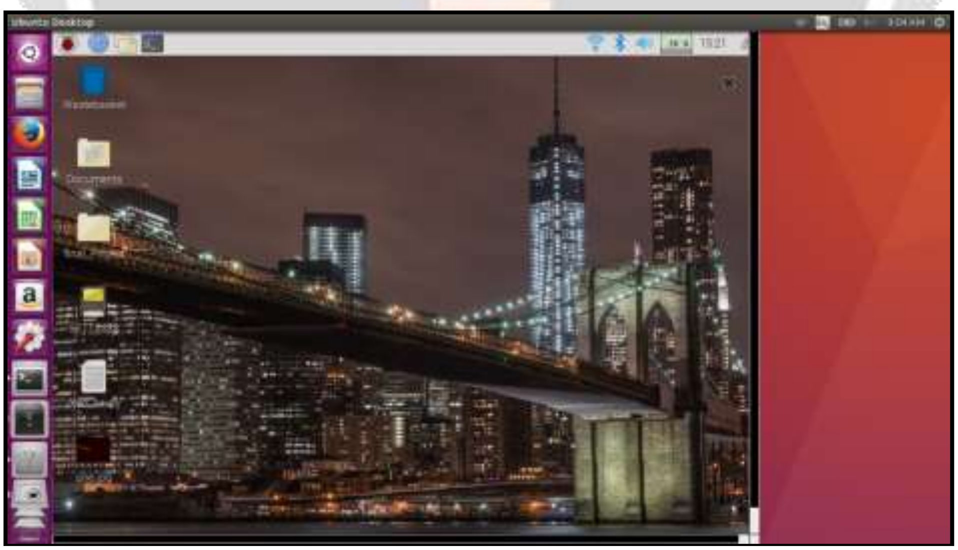


Fig -16: GUI interface of Raspberry pi for easy accessibility using VNC viewer.

5.3 Putty

Putty is a free and open-source terminal emulator, serial console and network file transfer application. It supports several network protocols, including SCP, SSH, Telnet, rlogin and raw socket connection. It can connect to serial port. Figure 17 shows a screenshot of the GUI interface Putty configuration dialog box.

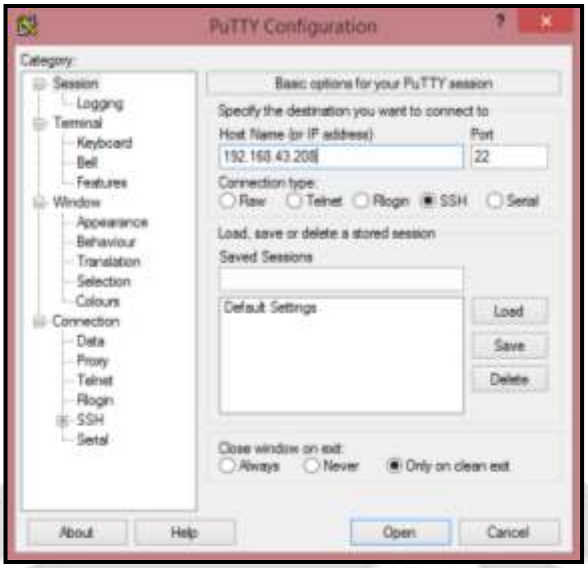


Fig -17: Putty Configuration Interface

5.4 Motion

Motion is an open-source Unix library written in C that connects to a camera and performs motion detection on the video feed. It has a set of parameters that can be configured to tune the motion detection algorithm. A number of features on what to do when motion is detected is also offered, including snapshot and video saving, inserting of rows into SQL and PostgreSQL databases, and others. The fact that it is open-source allows developers to inspect its workings and add new features [26].

6 PROGRAMMING DETAILS

6.1 Detecting Motion Using PIR sensor and Raspberry Pi

The programming logic aimed to detect motion after the PIR sensor was connected to the Raspberry pi. Many motion detectors use passive infrared (PIR) sensors. A PIR sensor permanently measures infrared light and notices whenever something in the infrared spectrum changes. This is all you need to detect motion, because nearly every object emits infrared light. That’s true for everything in front of your house, for example, the ground, a bicycle and so on. These things emit a constant portion of infrared light, and it does not change rapidly. But if a human being or an animal approaches the front door, the sensor will notice a big variation and sire a signal. The sensor has three pins that you need to connect to the Pi using female/male jumper wires. Figure 18 is a phone-captured image of how the developed system looks like.

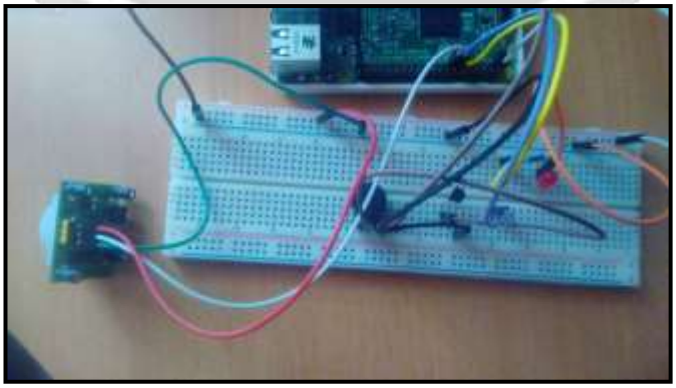


Fig -18: Raspberry pi motion sensor connection

6.2 Email Notification Program

In order to allow for email notifications to send, the OS needs a program that allows for emails to be sent. Simple Mail Transfer Protocol (SMTP) is a program that allows a system to deliver an email from a local computer to a mail host. It does not receive mail but can send out mail. SMTP is ideal for situations where alerts are needed to be sent, therefore it is useful when sending notifications. In this project, Mutt which is a command-line based Email client which incorporates SMTP is used. It’s a very useful and powerful tool to send and read mails from command line in Unix based systems. Mutt also supports POP and IMAP protocols for receiving mails. It opens with a colored interface to send Email which makes it user friendly to send emails from command line. It's fast, flexible and, best of all, surprisingly easy to use. A python script can be used to achieve this. It may just send a notification without image or can be modified to send an attachment along with the alarm message. For SMTP to support transmission of an attached file, Multipurpose Internet Mail Extension (MIME) is required.

7 RESULTS AND ANALYSIS

After importing the required packages and running the python code, the setup starts working. Whenever the motion is detected the output screen displays the message “intruder detected”. Simultaneously the camera module starts recording the events and E-mail notifications are sent to the user. Below are the results we obtained after running python code successfully. The first output is shown below. We can see an email with the text message displayed as “Intruder Alert”. Here we have our Gmail account named “stephanyopoku20@gmail.com”. All email notifications will be updated in this account. Whenever there is an intruder we get the following mail as shown in Figure 5.1

Fig -19: Mails to users whenever intruder is detected by the proposed system

We can see the above email is with an attachment. As soon as intruder will be detected, PIR sensor will sense the IR rays emitting from an intruder and will start giving high output. After that, the two cameras start working. Here the USB camera will begin to capture images as shown in Figure 18 and the pi camera takes the live video stream. Now an authenticate person will receive an email as shown in Figure 20 with an attachment. This attachment is in the form of .jpg file. Then an authenticate person will check mail, the following attachment of an intruder will be obtained as shown in Figure 5.2.

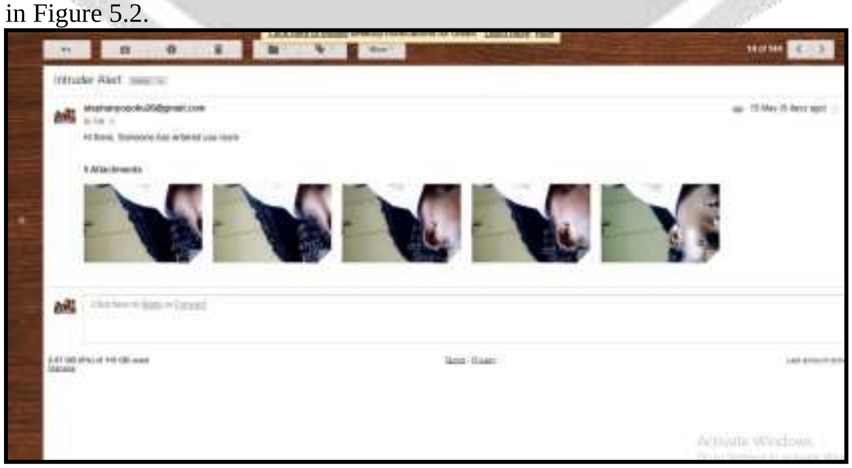


Fig -20: Motion detection images in email

Simultaneously the details of the captured image as well as the time and date are stored in a log file which is later sent and stored in a GitHub repository for future analysis. Figure 21 depicts such.

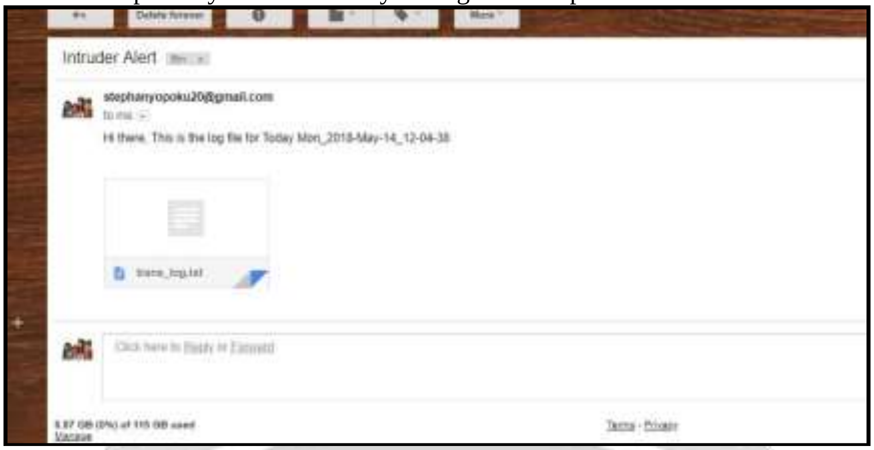


Fig -21: Log file of daily activities

Figure 22 shows a typical arrangement of the proposed system when incorporated in a real-life situation, whereby the proposed surveillance system will be built inside a house.

Fig -22: Model of a house with the surveillance system.

8 CONCLUSIONS AND RECOMMENDATIONS

In the end we conclude that every person wants to be in a better and secure world, this project has covered most of the existing systems so far and the work that has been done over this project. To provide better security and safety new designs are implemented and also that are cost effective. It will interest you to know that the proposed system can be made wireless for remote access and also better video quality which is easy to operate and, in a way, enable the security information to be carried and viewed anywhere. The smart surveillance system has been aimed to design in such a way that it can fulfil the needs of the user for surveillance area. It has countless applications and can be used in different environments and scenarios. For instance, at one scenario it can be used by any person working in an industry to be aware of the activity being happened at their own working places, in their absence, while at another instance it can be used for spy purposes at bank lockers, storage houses. Another application is to provide information to the user about what is happening in surveillance area by notification.

From improvement point of view, we can add new features to existing system such as, providing delay timer to the system so that owner can switch off the system. Major improvements on the system processor speed are much needed in order to process large files e.g. video for effective motion detection and tracking. The designed security system can be used in homes to monitor the facility at any given time. The future scope of this system can be extended further by adding additional infrared emitting system to detect people's face if they wore the mask on their face. Apart from this we can interface sensors like Gas sensors, Smoke sensors and fire sensors to give alerts respectively.

9 REFERENCES

[1]. Laganière, R. (2014). *OpenCV 2 Computer Vision Application Programming Cookbook: Over 50 Recipes to Master this Library of Programming Functions for Real-time Computer Vision*, 2nd Edition. Packt Publishing Ltd.

[2]. Arubas, E. (2013). Face Detection and Recognition (Theory and Practice). In *Eyal's Technical Blog*. Retrieved from http://eyalarubas.com/face-detection- and-recognition .html. on 05/05/2021

[3]. Datta, A. K., Datta, M., & Banerjee, P. K. (2015). Face detection and recognition: theory and practice. *CRC Press*.

[4]. Raspberry Pi. (2013), Raspberry Pi Web. Retrieved from: http://www.raspberrypi.org on 25/5/2021

[5]. Basha, S. N., Jilani, S. A. K., & Arun, M. S. (2016). An intelligent door system using Raspberry Pi and Amazon web services IOT. *International Journal of Engineering Trends and Technology (IJETT)*, *33*(2), 84-89.

[6]. Leu, J. S., Lin, W. H., & Tzeng, H. J. (2010). Design and implementation of a mobile home surveillance system. *Journal of the Chinese institute of engineers*, 33(5), 669-680.

[7]. Sharma, R. K., Mohammad, A., Kalita, H., & Kalita, D. (2014). Android interface-based GSM home security system. In *2014 IEEE International Conference on Issues and Challenges in Intelligent Computing Techniques (ICICT)*. 196-201.

[8]. Gu, Y., Kim, M., Cui, Y., Lee, H., Choi, O., Pyeon, M., & Kim, J. (2013). Design and implementation of UPnP-based surveillance camera system for home security. In *2013 IEEE International Conference on Information Science and Applications (ICISA)*. 1-4.

[9]. Willow G. (2014). OpenCV. Retrieved from https://www.willowgarage.com/pages/software/opencv on 06/03/2021.

[10]. Guennouni, S., Ahaitouf, A., & Mansouri, A. (2014). Multiple object detection using OpenCV on an embedded platform. In *2014 Third IEEE International Colloquium in Information Science and Technology (CIST)*. 374-377.

[11]. Trung, B. V. T., & Van Cuong, N. (2013). Monitoring and controlling devices system by GPRS on FPGA platform. In *2013 IEEE International Conference on Advanced Technologies for Communications (ATC)*. 713-717.

[12]. Karia, D., Baviskar, J., Makwana, R., & Panchal, N. (2013). Performance analysis of ZigBee based Load Control and power monitoring system. In *2013 IEEE International Conference on Advances in Computing, Communications and Informatics (ICACCI)*. 1479-1484.

[13]. Gantt, C (2013). Raspberry Pi Camera Module Review and Tutorial Guide. TweakTown News, Retrieved from http://www.tweaktown.com/guides/5617/raspberrypicameramodulereviewandtutorialguide/index4.html. on 22/2/2021

[14]. Ryu, Y., Yoo, J., & Kim, Y. (2012). Cloud services based Mobile monitoring for Photovoltaic Systems. In *4th IEEE International Conference on Cloud Computing Technology and Science Proceedings.* 578-580.

[15]. Deshmukh, A. D., & Shinde, U. B. (2016). A low-cost environment monitoring system using raspberry Pi and arduino with Zigbee. In *2016 IEEE International Conference on Inventive Computation Technologies (ICICT*. 3, 1-6.

[16]. Zhang, H., Srinivasan, R., & Ganesan, V. (2021). Low cost, multi-pollutant sensing system using raspberry pi for indoor air quality monitoring. *Sustainability*, *13*(1), 370.

[17]. Buenger, C, (2013). Raspberry Pi as Low-Cost HD Surveillance Camera. CodeProject, Retrieved from http://www.codeproject.com/Articles/6655RaspberryPiaslowcostHDsurveillance camera on 14/03/2021

[18]. Sanjana Prasad, P. A. (2014). Smart Surveillance Monitoring System Using Raspberry Pi and PIR Sensor. International Journal of Computer Science and Information Technologies (IJCSIT), 5(6), 7107-7109

[19]. Robson, C., Bohm, C., Plucinski, P., & Silverstein, S. (2012). High performance web applications for secure system monitoring and control. In *2012 IEEE Nuclear Science Symposium and Medical Imaging Conference Record (NSS/MIC).* 1109-1111.

[20]. Olaniyi, O. M., Bala, J. A., Ganiyu, S. O., & Wisdom, P. E. (2021). A Systematic Review of Background Subtraction Algorithms for Smart Surveillance System. *International Journal of Information Processing and Communication (IJIPC)* 8(1).35-54

[21]. Han, J., Yun, J., Jang, J., & Park, K. R. (2010). User-friendly home automation based on 3D virtual world. *IEEE Transactions on Consumer Electronics*, *56*(3), 1843-1847.

[22]. Swann Communications (2014), Swann 3425 Series DVR Manual, Retrieved from https://www.swann.com/au/downloads/dl/file/id/1507/product/662/3425_series_dvr_manual.pdf Retrieved on 18/02/2021.

[23]. Dhumane, A., Prasad, R., & Prasad, J. (2016). Routing issues in internet of things: a survey. In *Proceedings of the International Multiconference of Engineers and Computer Scientists (IMECS),* Hong Kong, 1, 16-18.

[24]. Bushnell Corporation (2016), Bushnell TrophyCam HD Instruction Manual, Manual Overland Park, KS: Bushnell Outdoor Products. Retrieved from https://www.bushnell.com/on/demandware.static/Sites-HuntShoot Accessories-Site/Library-Sites-HuntShootAccessoriesSharedLibrary/-/productPdfFiles/bushnellPdf/Product %20Manuals/Trail-Cameras/PDF/119676C_119677C_119678C_TrophyCamHD_1LIM_US-only_052014 _web.pdf on 21/02/2021.

[25]. Bushnell Trophy Cam HD Max Product Overview (2015) - Technical Specs, Retrieved from http://www. bushnell.com/all-products/trail-cameras/trophy-cam/trophy-cam-hd-max, on 21/02/2021.

[26]. MotionGuide (2020) Motion Guide for Motion Version 4.3.1. Retrieved from: https://motion-project.github.io/motion_guide.html on 25/02/2021

[27]. BVI Networks (2010), ShopperGauge in-store behavior monitoring system - Case study: Gauging the Impact of Display and Brand Messaging on the Cereal Category, Retrieved from http://ww1.prweb.com/prfiles/2011/01/05/2106834/ShopperGaugeCaseStudy.pdf on 18/06/2021

[28]. Caitlin S. (2016). Five Steps to Improve Shopper Marketing Impact. Retrieved from https://silo.tips/download/five-steps-to-improve-shopper-marketing-impact on 25/06/2021

[29]. Zalud, B. (2012). Video analytics: Big retail deal with big data. *Integrated Solution: Video*. Retrieved from https://www.securitymagazine.com/ext/resources/2012/September-2012/Retail-Video-Analytics---feat.doc 05/03/2021

[30]. Prism SkyLabs (2013), User Manual. Retrieved from https://docs.johnsoncontrols.com/exacq/v/u/exacqVision-Prism-Skylabs-Integration-Guide on 07/03/2021

[31]. Raspbian (2014). Retrieved from http://www.raspbian.org/RaspbianAbout on 19/05/2018

[32]. Tutorials Point (2013), Python Sending Email Using SMTP. Tutorials Point Simply Easy Learning. Retrieved from http://www.tutorialspoint.com/python/python_sending_email.htm on 02/04/2021

BIOGRAPHIES

	Opoku Stephanie Mensah had her bachelor's degree in computer science at KNUST in Kumasi Ghana from 2014-2018 with Second Class Upper Division. She continued as a research and teaching assistant in the same department from 2018-2019, when she published a paper on Repetitive Strain Injury during Computer Usage. She further joined Vodafone Ghana in 2019 as a graduate trainee in IT implementation, helping with the testing and implementation of digital systems. She has moved across various departments in the organization and now with the cybersecurity team as a cyber-complaint analyst, working mostly with 3rd parties, vendors as well as internal stakeholders, solving complaint with all Vodafone's approved security standards.
	Danso Juliana Mantebea (1994–) had her Bachelor's and Master's education at the Kwame Nkrumah University of Science and Technology (KNUST) Kumasi, in the Ashanti Region of Ghana. She currently possesses a Bachelor of Science (BSc) in Computer Science degree in 2018 with First Class Honors and successfully defended the thesis for her Master of Philosophy (MPhil) in Computer Science degree, hopefully to graduate in November, 2021, all from KNUST. She also possesses a Master in Engineering degree in Computer Science and Engineering in July, 2021, from the University of Electronic Science and Technology of China (UESTC), Chengdu, China, where she is currently admitted to pursue her 4-year PhD degree in Software Engineering between the years 2021 and 2025. From

<table>
<tr><td></td><td>2018 to 2019, she was a teaching and research assistant for KNUST. From 2019 to 2021, she is also a member of the eXtreme Interactive Experience (XIE) Lab, a subsidiary lab of the Future Media Research Center of UESTC, whereby her research area included the application of deep learning in autonomous driving and security strategies to curb its systems.</td></tr>
<tr><td></td><td>Gyamfi Enoch Opanin (1990–), in December 2013, had his Bachelor of Science (BSc) in Information Technology Education degree at the University of Education (UEW) and in July 2018, had his Master of Philosophy (MPhil) in Information Technology degree in the Kwame Nkrumah University of Science and Technology (KNUST), all located in Kumasi of the Ashanti Region of Ghana. He is a qualified teacher, possesses a degree in education and studied some major courses relating to teaching at the secondary school level of education. From 2015 to 2018, he was a part-time lecturer at the Christ Apostolic University College and at the Institute for Educational Development and Extension (IEDE) of the University of Education, Winneba. He currently pursuing a 4-year PhD in Software Engineering and is expected to complete in July 2022 where he is a member of the Key Laboratory of New Computer Application Technology and UESTC-IBM Technology Center. His research interest is centered on deep learning in computer vision, integrated with natural language and its applications in classification tasks.</td></tr>
<tr><td></td><td>Rodor Gershon possesses a bachelor of science in Information Technology Education from the University of Education, Winneba in Ghana. He is currently a final year graduate student at the College of Electronic and Information Engineering at Nanjing University of Aeronautics and Astronautics. His research interests include wireless communications and channel coding techniques for reliable communications.</td></tr>
</table>