International Journal of Internet of Things 2022, 10(1): 1-21
DOI: 10.5923/j.ijit.20221001.01

The Development of an Integrated Wireless Security Surveillance System Based on Internet-of-Things Technologies

Vincent A. Akpan[1,*], Joshua B. Agbogun[2], Demehin A. Olatunji[3]

[1]Department of Biomedical Technology, The Federal University of Technology, Akure, Ondo State, Nigeria
[2]Department of Computer Science and Mathematics, Godfrey Okoye University, Enugu, Enugu State, Nigeria
[3]Department of Physics Electronics, The Federal University of Technology, Akure, Ondo State, Nigeria

Abstract Increase in the rate of criminal activities demands the development of wireless security surveillance systems for intruder detection with an alert unit. This paper presents the development, construction and analysis of low-cost integrated wireless security surveillance (IWSS) system based on Internet-of-things (IoT) technologies. The IWSS system comprise of: *(i).* An HC-SR04 ultrasonic sensor and its LCD to display the distance of an approaching or stationary object or person; *(ii).* Raspberry Pi Zero WH embedded system development board with a Raspberry Pi E305854 camera V2.1 module for capturing real-time streaming video of the environment under surveillance via 192.168.43.70 as the Internet Protocol (IP) address for viewing with laptop and/or android phone simultaneously from remote locations; *(iii).* Ardiuno Mega 2560 embedded system development board with a SIM800L GSM/GPRS S2-1065J module for wireless connectivity and bi-directional communication by sending short message service (SMS) notification the property owner to a specified mobile phone number; *(iv).* A signal conditioning, tone generator and a 25-Watt audio amplifier circuits that makes up an electronic alarm system which serves as the alerting system; and *(v).* An automatic Li-Po battery backup power supply system with automatic 13.5-V battery charger with battery level indicator that ensures uninterrupted functionality of the IWSS system. The two major parts of the IWSS system are built around the Raspberry Pi Zero WH and the Arduino Mega 2560 embedded system development boards. The developed and constructed IWSS system has been tested and its performances satisfied the desired objectives of the system. The software programs for the full implementation of the IWSS system are given in the Appendices. The developed IWSS system can be adapted for complete home automation.

Keywords Embedded systems, Home automation, Internet-of-Things (IoT), Security surveillance, Uninterruptible power supply, Wireless communication, Wireless security system

1. Introduction

Due to the increase in criminal activities, most especially petty theft and burglary, there is a great need to safeguard properties from criminals and intruders. Several intruder alarm systems make use of alarms which deter the criminal. The downside of this method is not subtle and sophisticated since criminals and/or intruders can disable the alarm and cart away properties of innocent citizens without fear of being caught [1,2]. On the other hand, the alarm system can serve as notification and alerting system.

Home security refers to crimes prevention such as burglaries and kidnapping. As a preventive measure, a resident may install an IP-based camera but this method is quite costly because the resident needs to buy a personal computer, router, and software. Internet protocol surveillance cameras based application have drastically increased in recent years by business owner and home residential in seeking to reduce crime [1]. Hence, manufacturers started incorporating wireless technology 802.11 to simplify surveillance cameras deployment and maintenance [1].

Criminals break into houses on a daily basis around the world carting with huge amount of money and precious items [3]. Sensitive and confidential documents, materials and equipment by corporation are constantly declared missing from where they are kept due to theft. So there is a need to provide a device that can detect unauthorized persons in an environment to alleviate crimes and it related offences. In fact, the annualised totals for burglary and related offences have been increasing continuously since 2009 [3]. Approximately €72 million worth of goods were stolen from Irish residential households in the period July 2008 to June

* Corresponding author:
vaakpan@futa.edu.ng (Vincent A. Akpan)
Received: Nov. 27, 2021; Accepted: Dec. 26, 2021; Published: Feb. 24, 2022
Published online at http://journal.sapub.org/ijit

2009; and this equates to an average value of €3,011 per home burgled [3].

Intrusion detection is typically one part of an overall protection system that is installed around a system or device and it is not a stand-alone protection measure. As mentioned earlier, intrusion is defined as “any set of actions that attempt to compromise the integrity, confidentiality, or availability of a resource” and intrusion prevention techniques (such as encryption, authentication, access control, secure routing, e.t.c.) are presented as the first line of defense against intrusions.

Wireless technology has been gaining rapid popularity over the years, and security is considered as one of the most critical parameter for the acceptance of any wireless networking technology [2]. Although implementation of technological solutions is the most common way to respond to threats of wireless security systems and susceptibility, wireless security is basically a management issue. Effective management planned after analyzing current threats can help to sort out issues much better.

The main goal of this paper is on the development of integrated wireless security surveillance (IWSS) system based on Internet-of-Things (IoT) technologies. The main goal will be accomplished according to the following specific procedures which includes to: *(i)* interface and configure a Raspberry Pi Zero WH embedded system development board with a camera module; *(ii)* interface and configure an Arduino Mega 2560 embedded system development board with an ultrasonic sensor, a liquid crystal display (LCD), and a SIM800L GSM/GPRS module; *(iii)* design an automatic battery backup power supply system with automatic battery charger with battery level indicator; *(iv)* an electronic alarm system which incorporates a 25-watt audio amplifier; *(ii)* configure the SIM800L S2-1065J module for seamless wireless connectivity and bi-directional communication with a remote laptop and android mobile phone for remote monitoring via a network service provider (NSP).

2. Theoretical Background

Home automation or smart home (also known as *domotic*) can be described as the introduction of technology within the home environment to provide convenience, comfort, security and energy efficiency to its occupants [4]. The security surveillance system or an intruder alarm system monitors and detects unauthorised entry to properties – homes and businesses alike - and alert the property owner or authorised third parties to an intrusion [5]. Obviously, wireless systems can be employed in this situation.

In the past few years, wireless technologies reached their breakthrough [6]. Wireless based systems, used every day and everywhere, ranges from wireless home networks and mobile phones to garage door openers [6]. As of today, little comparative research of wireless automation standards has been done, although such knowledge would provide valuable information to everyone looking for the most suitable system for given requirements.

Adding intelligence to home environment can provides increased quality of life for the elderly and disabled people who might otherwise require caregivers or institutional cares [7]. There has been a significant increase in home automation in recent years due to higher affordability and advancement in Smart phones and tablets which allows vast and countless connectivity [1–4,7]. With the introduction of the Internet-of-Things (IoT) technology the research and implementation of home automation are getting more popular.

IoT refers to uniquely identifiable objects and their virtual representations in an Internet-like structure. The term IoT was proposed by Kevin Ashton in 1999 although the concept had been initially discussed since at least in 1991 [8,9]. The Internet-of-Things has made it even easier than ever to set up a smart home in which door locks, lights, thermostats, vacuums, lawn mowers and even pet feeders can be remotely control led using smartphone with softwar application (App); and it has also made it simple (and relatively affordable) to monitor your home from pretty much anywhere [7].

The term IoT has been defined as the inter-networking of physical devices, vehicles (also referred to as “connected devices” and “smart devices”), buildings and other items-embedded with electronics, software, sensors, actuators and network connectivity that enable these objects to collect and exchange data [10]. In 2013 the global standards initiative on Internet-of-things (IoT-GSI) defined the IoT as “the infrastructure of the information society” [1]. The internet of things is defined as the network of physical objects that contain embedded technology to communicate and sense or interact with their internal states or the external environment.

Typically, IoT is expected to offer advanced connectivity of devices, systems, and services that goes beyond machine-to-machine (M2M) communications and covers a variety of protocols, domains, and applications [11]. The interconnections of these embedded devices (including smart objects) is expected to usher in automation in nearly all fields, while also enabling advanced applications like smart grid and expanding to areas such as smart cities [12,13].

IoTs can be viewed as connecting everyday objects like Internet enabled TVs, smart- phones, actuators and sensors to the Internet. The devices are smartly linked together allowing new forms of communication between people and things, and among things themselves. IoT technology has advanced significantly in the last few years since it has added a new dimension to the world of communication and information technologies. It's predicted that the number of devices connected to the Internet will increase from 100.4 million in 2011 to 2.1 billion by the year 2021; this growth is at a rate of 36% per year [3].

There is therefore a need to create a system which can discreetly alert appropriate authorities when an intruder is in the house thereby increasing the rate of arrests and ultimately reducing the rate of such crimes [14,15]. The security surveillance system based on IoT has always been considered as a second wall of defense from the security point of view. Security surveillance systems are cyberspace equivalent of the burglar alarms that are being used in physical security systems today [1,7].

3. Overview of the Proposed IWSS System

3.1. Block Diagram of the Proposed IWSS System

The block diagram of the proposed integrated wireless security surveillance (IWSS) system is shown in Figure 1. As evident in Figure 1, the block diagram consists of: *(i).* A HC-SR04 ultrasonic sensor and its LCD to display the distance of an approaching or stationary object or person; *(ii).* Raspberry Pi Zero WH embedded system development board with a Raspberry Pi E305854 camera V2.1 module; *(iii).* Ardiuno Mega 2560 embedded system development board with a SIM800L GSM/GPRS S2-1065J module for wireless connectivity and bi-directional communication; *(iv).* A signal conditioning, tone generator and a 25-Watt audio amplifier circuits that makes up an electronic alarm system; and *(v).* An automatic Li-Po battery backup power supply system with automatic 13.5-V battery charger with battery level indicator. In fact, the two major parts of the IWSS system are built around the Raspberry Pi Zero WH and the Arduino Mega 2560 embedded system development boards. The description of some key components and basic architecture of the proposed IWSS system are given in the next subsections.

3.2. The HC-SR04 Ultrasonic Proximity Sensor

The HC-SR04 ultrasonic sensor is a device that is capable of measuring the distance to an object by using sound waves [16–19]. It measures by sending a sound wave at a specific frequency and listening for that sound wave to bounce back. By recording the elapsed time between the sound wave being generated and sound wave bouncing back, it is possible to calculate the distance between the sonar sensor and the object. The HC-SR04 ultrasonic sensor used for obstacle detection is shown in Figure 2. The module pin description of the HC-SR04 ultrasonic sensor is shown in Table 1 while the electrical parameters are shown in Table 2.

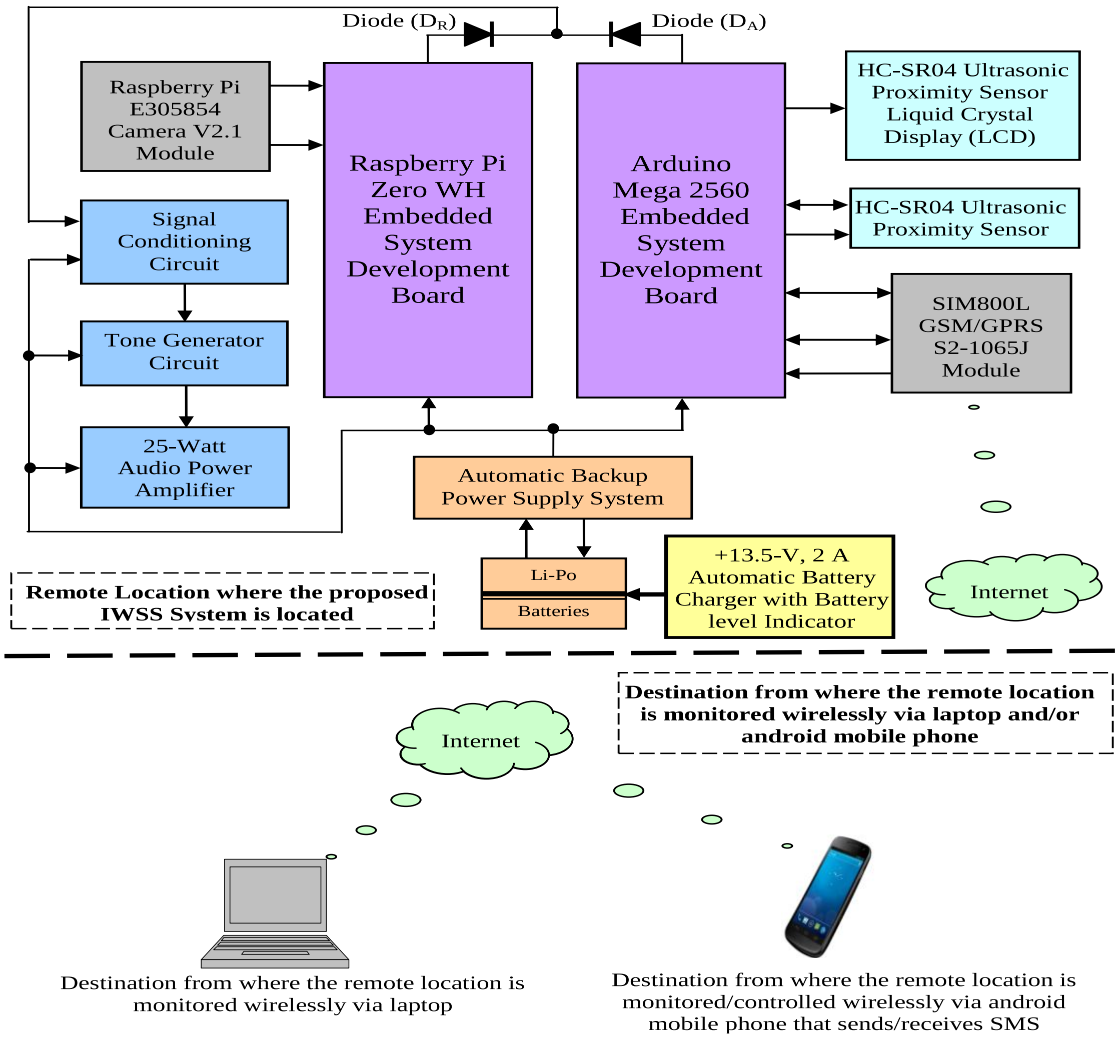

Figure 1. The block diagram of the proposed integrated wireless security surveillance (IWSS) system

Figure 2. The HC-SR04 ultrasonic proximity sensor

Table 1. Module pin definition for the HC-SR04 ultrasonic sensor [17–19]

Type	Pin Symbol	Pin Function Description
HC-SR04	Vcc	5 V power supply
	Trig	Trigger pin
	Echo	Receiver pin
	GND	Power ground

Table 2. Module pin definition for the HC-SR04 ultrasonic sensor [17–19]

S/N	Electrical Parameters	HC-SR04 Ultrasonic Module
1.	Operating Voltage	DC-5V
2.	Operating Current	15 Ma
3.	Operating Frequency	40 KHz
4.	Farthest Range	4 m
5.	Nearest Range	2 cm
6.	Measuring Angle	15°
7.	Input Trigger Signal	10 µs TTL Pulse
8.	Output Echo Signal	Output TTL level signal proportional with range
9.	Dimensions	45 x 20 x 15 mm

3.3. The YJD1602A-1 16-by-2 LCD Module

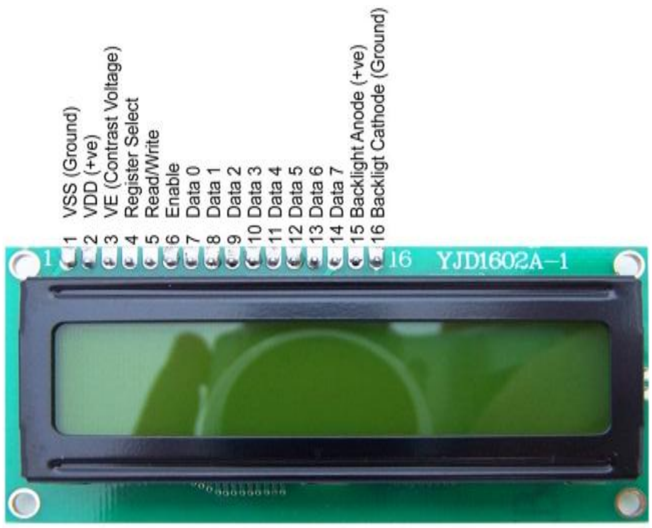

Figure 3. The physical picture and pin definition of the YJD1602A-1 16-by-2 LCD display module [17–20]

Liquid Crystal Display (LCD) screen is an electronic display module and find a wide range of applications in digital electronic systems [17–20]. The 16x2 LCD display is a very basic module and is very commonly used in various devices and circuits [17–20]. These modules are preferred over seven segments and other multi segment LEDs. The reasons being that LCDs are economical; easily programmable; have no limitation of displaying special and even custom character (unlike in seven segments), animations and so on. A 16x2 LCD means it can display 16 characters per line and there are two such lines. The YJD1602-A LCD module used in this work is shown in Figure 3.

3.4. The Arduino Mega 2560 Embedded System Development Board

The Arduino Mega 2560 embedded systems development board shown in Figure 4 is a microcontroller board based on the ATmega2560 [17–19,21]. It has 54 digital input/output pins (of which 15 can be used as PWM outputs), 16 analog inputs, 4 UARTs (hardware serial ports), a 16 MHz crystal oscillator, a USB connection, a power jack, an ICSP header, and a reset button. It contains everything needed to support the microcontroller; simply connect it to a computer with a USB cable or power it with an AC-to-DC adapter or battery to get started. The Mega is compatible with most shields designed for the Arduino Duemilanove or Decimals. The ATmega2560 on the Mega 2560 comes preprogrammed with a bootloader that allows you to upload new code to it without the use of an external hardware Programmer. It communicates using the original STK500 protocol (reference, C header files) [21].

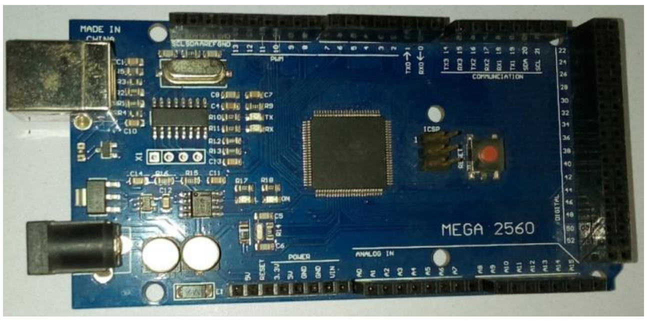

Figure 4. The top view of Arduino Mega 2560 embedded system development board

3.5. The Raspberry Pi Zero WH embedded System Development Board [22,23]

3.5.1. Architectural Description

Raspberry Pi Zero WH embedded system development board is a mini single-chip computer. The top view of the Raspberry Pi Zero WH embedded system development board is shown in Figure 5 [22–23]. Raspberry Pi Zero WH is the smallest chipset in the Raspberry Pi series and is 40% faster than the original Raspberry Pi but nearly half of its size. Raspberry pi Zero GPIO pin-out or Pin diagram, and programming methods are explained in detail in this post. The Raspberry Pi Zero supports mini connectors (like mini HDMI, mini USB power, and USB on-the-go port) to save more space. And the 40pin GPIO is unpopulated which provides the flexibility to use only the connections that the project requires. It consists of a **1-GHz BCM2835 single-core** processor, **512 MB RAM, mini-HDMI, USB On-The-Go ports**, and a camera connector.

One powerful feature of the Raspberry Pi is the row of GPIO (general-purpose input/output) pins along the extreme right edge of the board. Like every Raspberry Pi chipset, Zero consists of a 40-pin GPIO. The standard interface for connecting a single-board computer or microprocessor to other devices is through General-Purpose Input/Output (GPIO) pins. GPIO pins do not have a specific function and can be customized using the software.

The typical specification the Raspberry Pi Zero WH includes [22–23]: *1).* BCM2835 single-core processor: 1GHz ARM11 core; *2).* 512MB of LPDDR2 SDRAM; *3).* A micro-SD card slot; *4).* A mini-HDMI socket for 1080p60 video output; *5).* Micro-USB sockets for data and power; *6).* An unpopulated 40-pin GPIO header; *7).* An unpopulated composite video header; *8).* A dimension of about 65mm x 30mm x 5mm; *9).* Composite video and reset headers; and *10).* CSI camera connector.

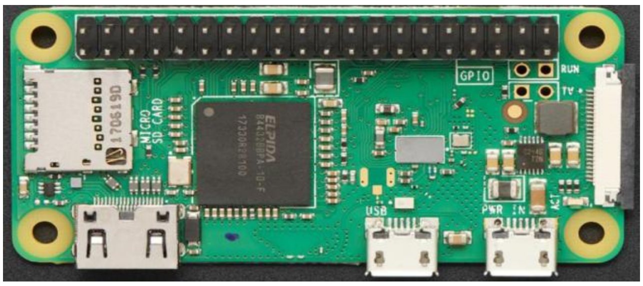

Figure 5. The top view of Raspberry Pi Zero WH embedded system development board

3.5.2. Raspberry PI Zero Power Pins

The Raspberry Pi board consists of two 5 V pins, two 3.3 V pins, and 9 ground pins (0 V), which are unconfigurable. The 5 V pins directly deliver the 5 V supply coming from the mains adaptor. This pin can be used to power up the Raspberry Pi zero, and it can also be used to power up other 5 V devices. The 3.3 V pin is there to offer a stable 3.3 V supply to power components and to test LEDs. The Ground (GND) pin is from where all voltages are measured and it also completes all the electrical circuits of the board.

3.5.3. Raspberry Pi Zero Input/Outputs Pins

A GPIO pin that is set as an **input** pin, receives the incoming voltage signal sent by the device connected to this pin. A voltage between 1.8 V and 3.3 V will be read by the Raspberry Pi as HIGH and if the voltage is lower than 1.8 V will be read as LOW. It is advisable not to supply more than 3.3 V to GPIO pins, or else it will destroy the Raspberry Pi zero board. A GPIO pin set as an **output** pin sends the voltage signal as high (3.3 V) or low (0 V). When this pin is set to HIGH, the voltage at the output is 3.3 V and when set to LOW, the output voltage is 0 V.

3.5.4. Other Important Pins on Raspberry Pi Zero

Along with the simple function of input and output pins, the GPIO pins can also perform a variety of alternative functions. Some specific pins are:

(a) PWM (pulse-width modulation) Pins: Software PWM available on all pins and Hardware PWM available on these pins: GPIO12, GPIO13, GPIO18, GPIO19.

(b) SPI PINS on R-Pi Zero: SPI (Serial Peripheral Interface) is another protocol used for master-slave communication. It is used by the Raspberry pi board to quickly communicate between one or more peripheral devices. Data is synchronized using a clock (**SCLK** at GPIO11) from the master (RPi) and the data is sent from the Pi to the SPI device using the **MOSI** (Master Out Slave In) pin. If the SPI device needs to communicate back to Raspberry Pi, the SPI device sends the data back using the **MISO** (Master In Slave Out) pin. **The SPI0 include** GPIO9 (MISO), GPIO10 (MOSI), GPIO11 (SCLK), GPIO8 (CE0), GPIO7 (CE1) while the **SPI1 consists of** GPIO19 (MISO), GPIO20 (MOSI), GPIO21 (SCLK), GPIO18 (CE0), GPIO17 (CE1), GPIO16 (CE2).

3.5.5. I2C Pins on R-Pi Zero

I2C is used by the Raspberry Pi board to communicate with devices that are compatible with Inter-Integrated Circuit (a low-speed two-wire serial communication protocol for communicating with low-speed peripheral). This communication standard requires master-slave roles between both the devices. I2C has two connections: **SDA (Serial Data)** and **SCL (Serial Clock)**. They work by sending data to and using the SDA connection and the speed of data transfer is controlled via the SCL pin. The Data pins are (GPIO2) and Clock (GPIO3) while the EEPROM Data ports are (GPIO0) and EEPROM Clock (GPIO1).

3.5.6. UART Pins on R-Pi Zero

Serial communication or the **UART** (Universal Asynchronous Receiver / Transmitter) pins provide a way to communicate between two microcontrollers or the computers. TX pin is used to transmit the serial data and RX pin is used to receive serial data coming from a different serial device. The transmitter pin is TX (GPIO14) while the receiver pin is RX (GPIO15).

3.6. The Raspberry Pi E305654 Rev 2.1 5-MP Camera Board Module [24–26]

The Raspberry Pi E305654 Rev 2.1 Camera Module, shown in Figure 6, is a 5 megapixel (5-MP) custom designed add-on for Raspberry Pi, featuring a fixed focus lens [24–26]. The camera module is capable of 2592 x 1944 pixel static images, and also supports 1080p30, 720p60 and 640x480p60/90 video. It attaches to Pi by way of one of the small sockets on the board upper surface and uses the dedicated Camera Serial Interface (CSI), designed especially for interfacing to cameras. In fact, The Raspberry Pi Camera

Board plugs directly into the CSI connector on the Raspberry Pi as shown in Figure 7. It is able to deliver a crystal clear 5MP resolution image or 1080p HD video recording at 30fps. The 5-MP native resolution sensor that is capable of 2592 x 1944 pixel static images with Omnivision 5647 sensor in a fixed focus module. Supports 1080p30, 720p60 and 640x480p60/90 video Camera is supported in the latest version of Raspbian, Raspberry Pi's preferred operating system. The camera board module is tiny with a dimension of about 25mm x 20mm x 9mm. It also weighs just over 3 g, making it perfect for mobile or other applications where size and weight are important. It connects to Raspberry Pi by way of a short ribbon cable. The camera sensor has a native resolution of 5-MP, and has a fixed focus lens on-board. In terms of still images, the camera is capable of 2592 x 1944 pixel static images, and also supports 1080p30, 720p60 and 640 x 480p60/90 video recording.

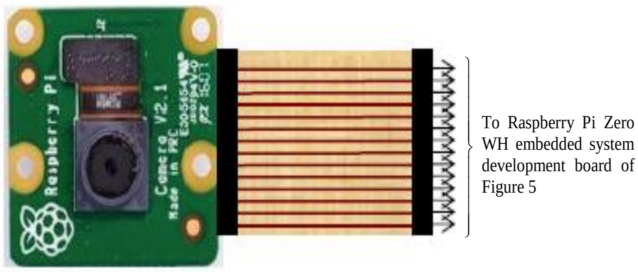

Figure 6. The Raspberry Pi E305654 Rev 2.1 5-MP camera board module

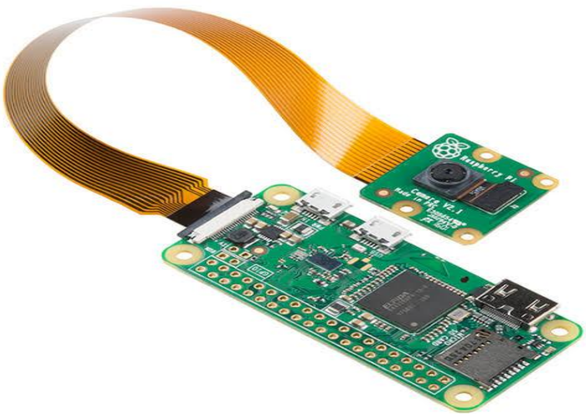

Figure 7. The Raspberry Pi E305654 Rev 2.1 camera module with the Raspberry Pi Zero WH board via the 15-pin MPIP CSI cable

The camera is supported in the latest version of Raspbian, Raspberry Pi's preferred operating system. The 1.4 µm x 1.4 µm pixel with OmniBSI technology for high performance (high sensitivity, low crosstalk, low noise) optical size of 1/4 automatic image control functions such as: *1).* automatic exposure control (AEC); *2).* automatic white balance (AWB); *3).* Automatic band filter (ABF); *4).* Automatic 50/60 Hz luminance detection; *5).* Automatic black level calibration (ABLC) programmable controls for frame rate; *6).* AEC/AGC 16-zone size/position/weight control, mirror and flip, cropping, windowing; and *7).* Panning digital video port (DVP) parallel output interface 32 bytes of embedded one-time programmable (OTP) memory. **The Raspberry Pi** E76181/E/94V-0 Rev1.3 **Camera Board is f**ully compatible with both the model A and model B Raspberry Pi embedded system development board. The **Raspberry Pi** E305654 Rev 2.1 **Camera Board** module attaches to Raspberry Pi Zero WH embedded System Development Board, by way of a 15 Pin Ribbon Cable, to the dedicated 15-pin MIPI (mobile industry processor interface) Camera Serial Interface (CSI), which was designed especially for interfacing to cameras as shown in Figure 7. The CSI bus is capable of extremely high data rates, and it exclusively carries pixel data to the BCM2835 processor [24–26].

3.7. SIM800L GSM/GPRS S2-1065J Module

SIM800L GSM/GPRS S2-1065J module is a miniature GSM modem, which can be integrated into a great number of IoT projects. The SIM800L GSM/GPRS module is shown in Figure 8. This module can be used to accomplish almost anything a normal cell phone can achieve which include but not limited to SMS text messages, make or receive phone calls, connecting to internet through GPRS, TCP/IP, e.t.c [27–29]. In fact, the module supports quad-band GSM/GPRS network, meaning it works pretty much anywhere in the world [27]. The typical specifications of the SIM800L GSM/GPRS module are listed in Table 3 [27–29].

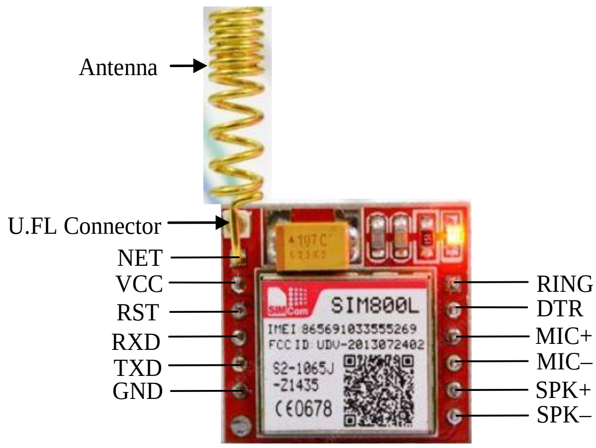

Figure 8. The SIM800L GSM/GPRS module

The SIM800L Quad-band Network Mini GPRS GSM Breakout Module is based on SIM800L module which supports quad-band GSM/GPRS network and is available for GPRS and SMS message data remote transmission [28]. The board features compact size and low current consumption. With power saving technique, the current consumption is as low as 1mA in sleep mode. It communicates with the microcontroller via UART port, supports command including 3GPP TS 27.007, 27.005 and SIMCOM enhanced AT Commands.

The SIM800L S2-1065J GSM/GPRS module is a QUAD BAND GSM/GPRS module which compatible with Arduino [21,29]. The module works to add both of GSM features (voice call or SMS) and GPRS features. The advantages of these modules are that the VCC and TTL serial have 5 V and can directly be connected to Arduino or other minimum systems with 5 V of voltage level. There are so many GPRS/GSM modules on the market which need to add 5 V regulator and level converter circuit, while SIM800L module already has a built-in regulator circuit and TTL level converter on the board [29].

It should be noted that the RX pin on module should not be connected directly to Arduino's digital pin as Arduino Mega 2560 uses 5V GPIO whereas the SIM800L module uses 3.3 V level and **is not 5 V tolerant**. This means the TX signal coming from the Arduino Mega 2560 must be stepped down to 3.3 V so as not to damage the SIM800L module. There are several ways to do this but the technique used here is a simple resistor divider. A 10 kΩ resistor is inserted between the SIM800L RX and Arduino Mega 2560, and then a 20 kΩ between SIM800L RX and GND [27]. Furthermore, absolute care must taken not to disconnect the GND before the positive power supply voltage rail (5 V or 3.3 V) rather the GND should be connected before the positive power rail. Otherwise, the SIM800L module can use the low voltage serial pin as the GND which would instantly damage the SIM800L module [27]. The descriptions of the SIM800L pinouts are listed in Table 4 [29].

Table 3. The typical specification of the SIM800L GSM/GPRS module [27–29]

S/N	Specifications
1.	TTL serial interface compatible with 3.3V and 5V Microcontrollers, compatible with Arduino.
2.	This SIM800L module has a set of TTL level serial interface, a set of power supply interface.
3.	Besides, there are a set of antenna interface on this module.
4.	Network support: Quad-Band 850 / 900/ 1800 / 1900 MHz, it can transmit Voice, SMS and data information with low power.
5.	VDD TTL UART interface The TTL UART serial interface, you can connect the MCU like 51MCU or ARM or MSP430 directly. The pin of VDD is used to match voltage of the TTL.
6.	Model: SIMCOM SIM800L
7.	Work voltage: 3.7 V to 5 V
8.	Size: 40mm x 28mm x 3mm
9.	GPRS multi-slot class 12/10
10.	GPRS mobile station class B
11.	Compliant to GSM phase 2/2+
12.	Class 4 (2 W @ 850/900MHz)
13.	Send and receive SMS messages
14.	Scan and receive FM radio broadcasts
15.	Lead out buzzer and vibrational motor control port
16.	Class 1 (1 W @ 1800/1900MHz)

Table 4. The pin description of the SIM800L GSM/GPRS module

Pin Number	Pin Name	Description
1	NET	This is a pin where the Helical Antenna provided along with the module can be soldered.
2	VCC	This supplies power for the module. This can be anywhere from 3.4 to 4.4 V. Note that connecting it to 5 V pin will likely destroy the module and It does not even run on 3.3 V. An external power source like Li-Po battery or DC-DC buck converters rated 3.7 V 2A is commended.
3	RST	This is a hard reset pin. This pin can be pulled low for 100ms to perform a hard reset.
4	RXD	This is the receiver pin and is used for serial communication.
5	TXD	This is the transmitter pin and is used for serial communication.
6	GND	This is the Ground pin and needs to be connected to GND pin on the Arduino.
7	SPK+	This is a differential speaker interface. The positive pin of a speaker can be tied directly to this pin.
8	SPK–	This is a differential speaker interface. The negative pin of a speaker can be tied directly to this pin.
9	MIC+	This is a differential microphone input. The positive of the microphone pin can be connected directly to this pin.
10	MIC–	This is a differential microphone input. The negative of the microphone pin can be connected directly to this pin.
11	DTR	This pin activates/deactivates sleep mode. Pulling it HIGH will put module in sleep mode, disabling serial communication. Pulling it LOW will wake the module up.
12	RING	This pin acts as a Ring Indicator. It is basically the "interrupt" out pin from the module. It is by default set high and will pulse low for 120ms when a call is received. It can also be configured to pulse when an SMS is received.

3.8. Technical Considerations on Raspberry Pi Zero WH and Arduino Mega2560 Development Tools

The raspberry Pi and the Arduino have become invaluable tools [21,22]. They are both extremely popular option and very budget-friendly as well as being suited in their particular applications. Both have their pros and cons as listed in Table 5. Choosing between an Arduino and raspberry Pi depends entirely on the project. Arduino is best suited as interface for tasks that requires reading data from sensors and execute corresponding actions in real-time. Also, Arduino have low power requirements and has very low maintenance requirements. Arduino is ideal for projects that need to be constantly running g with little or no interactions. Raspberry Pi on the other hand should be considered when the task is might need a personal computer to work. The Pi simplifies projects when lots of operations are required to manage. This could be connecting to the internet to read and write data. It could also include jugging media of any kind or connecting to an external display.

When Raspberry Pi was first introduced and released, it initially seemed that Arduino might become obsolete. However, this is not really the case as they perform different tasks. Whether the raspberry Pi or the Arduino is of choice, the fact remains that each has distinct advantages and disadvantages over each other as listed in Table 5. However, they can both be connected through a variety of options like the USB, local networks or through inputs and outputs ports in the Arduino to the raspberry Pi.

Table 5. Analysis of the difference and similarities between Arduino and Raspberry Pi

Remarks	Arduino	Raspbery Pi
Pros	Arduino is easier to get started with. Best used for real time applications of hardware, software and IDE is open source. Arduino does not need a lot of programming knowledge for basic applications. Arduino is very easy to extend and has a lot of contributed shield and libraries.	Raspberry Pi easily connects to the internet. The Pi has the entire Linux software stack available. The Raspberry can be programmed using a variety of languages.
Cons	Hardware access is not real time. If CPU is bugged down, hardware interfacing can be delayed. Arduino lacks enough power to drive inductive loads. Lacks an inbuilt analog to digital converter. Hardware is not open source.	Not as powerful as the Raspberry Pi. Can only be programmed using C/C++ Connection to the internet is trickier than the Pi but is possible. Data can be passed using YQL or JSON.

3.8.1. Raspberry Pi Zero HW Development Tools [22]

The raspberry Pi is effectively a mini computer on one board, it comes with a dedicated processor, memory, graphic driver, input and outputs like high definition multimedia interface (HDMI). Raspberry Pi Boards runs a specially designed version of the Linux operating system as well. Raspberry Pi is actually a system-on-chip (SoC). It owns a full version of Linux, such as Raspbian, and it is designed to help teach you as you go. Arduino on the other hand is more of a microcontroller than a computer that has a massive support community as well as hundreds of expansion options.

3.8.2. Arduino Mega2560 Development Tools [21]

Unlike the raspberry Pi development tools, Arduino boards and development tools are actually microcontrollers rather than a full computers. Arduino lacks a full operating system but it can execute programmes that are interpreted by its firmware. Because of this, you do have access to basic tools that operating system would provide but you gain the flexibility of executing code directly with no operating system overhead. Arduino has no API (application programming interface) as there is no operating system. The API is a set of routines, protocols and tools for building software applications. Arduino basically runs code on bare metal.

4. The Development of the Integrated Wireless Security Surveillance (IWSS) System Based on IoT Technologies

As mentioned in Section 3.1, the two major parts of the IWSS system are the Raspberry Pi Zero WH and the Arduino Mega 2560 embedded system development boards. The Raspberry Pi Zero WH board interface with the Raspberry Pi camera module while the Arduino Mega 2560 interfaces with the SIM800L GSM/GPRS S2-1065J module as shown in Figures 9 and 10 respectively as discussed in the next two sub-sections.

4.1. Raspberry Pi Zero WH Board and the Raspberry Pi Camera Module

Interfacing of the Raspberry Pi Zero WH embedded system development board with the Raspberry Pi E305654 camera board module described in Section 3.6 is shown in Figure 9. The camera module is configure in such a way that when the program is compiled, uploaded, and the serial monitor is opened; the camera sends out strings of numbers depending while waiting for moving object detection. When a moving object is detected in front of the camera, the string of numbers produced by the serial monitor stops reading and the object is displayed and the notification "*intruder detected, sending text notification*" is displayed on the serial monitor. At the instant when the moving object is detected, the blue and green LEDs connected to GPIO3 and GPIO13 respectively turns ON which triggers the electronic alarm system (discussed in Section 4.1.3) via Diodes D_{R1} and D_{R2}. The Raspberry Pi Zero WH has an IP address of 192.168.43.70.

Two options have implemented in this work to simultaneously view the streaming video from the camera in real-time, namely: HP Laptop and Android mobile phone. Firstly, the HP Laptop and Android mobile phone are connected to the Internet. Secondly, the browsers of the HP Laptop and Android mobile phone are opened and lunch with the IP address (192.168.43.70) of the Raspberry Pi.

4.1.1. SIM800L Wireless Connectivity and Obstacle Detection System with Arduino Mega 2560 Board

Interfacing of the Arduino Mega 2560 embedded system development board with the HC-SR04 ultrasonic proximity sensor, YJD1602A-1 16-by-2 LCD module and SIM800L GSM/GPRS S2-1065J module discussed respectively in Sections 3.2, 3.3 and 3.7 is shown in Figure 10. The HC-SR04 ultrasonic proximity sensor is employed here to sense the presence and distance of the approaching object. The Arduino Mega 2560 is programmed in such a way that whenever an object is 2 m closer to the HC-SR04 sensor, the electronic alarm system is triggered and the measured distance is displayed on the YJD1602A-1 16-by-2 LCD module.

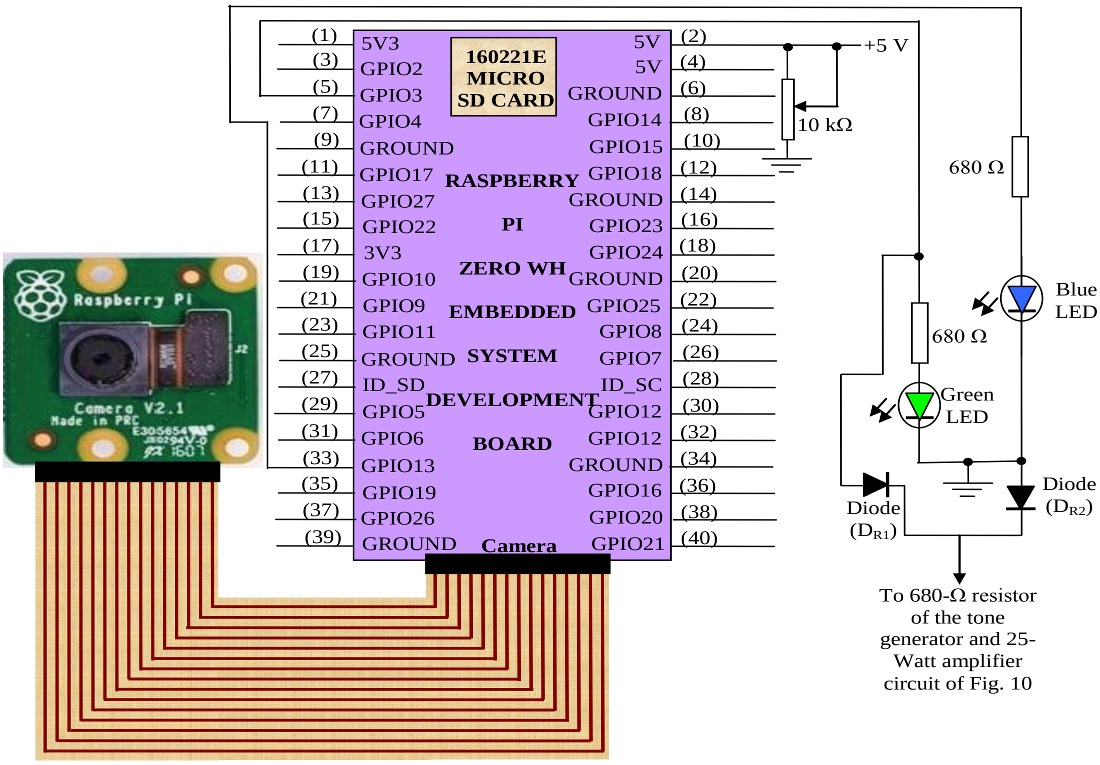

Figure 9. The Raspberry Pi Zero WH and the remote Raspberry Pi camera system for real-time video streaming

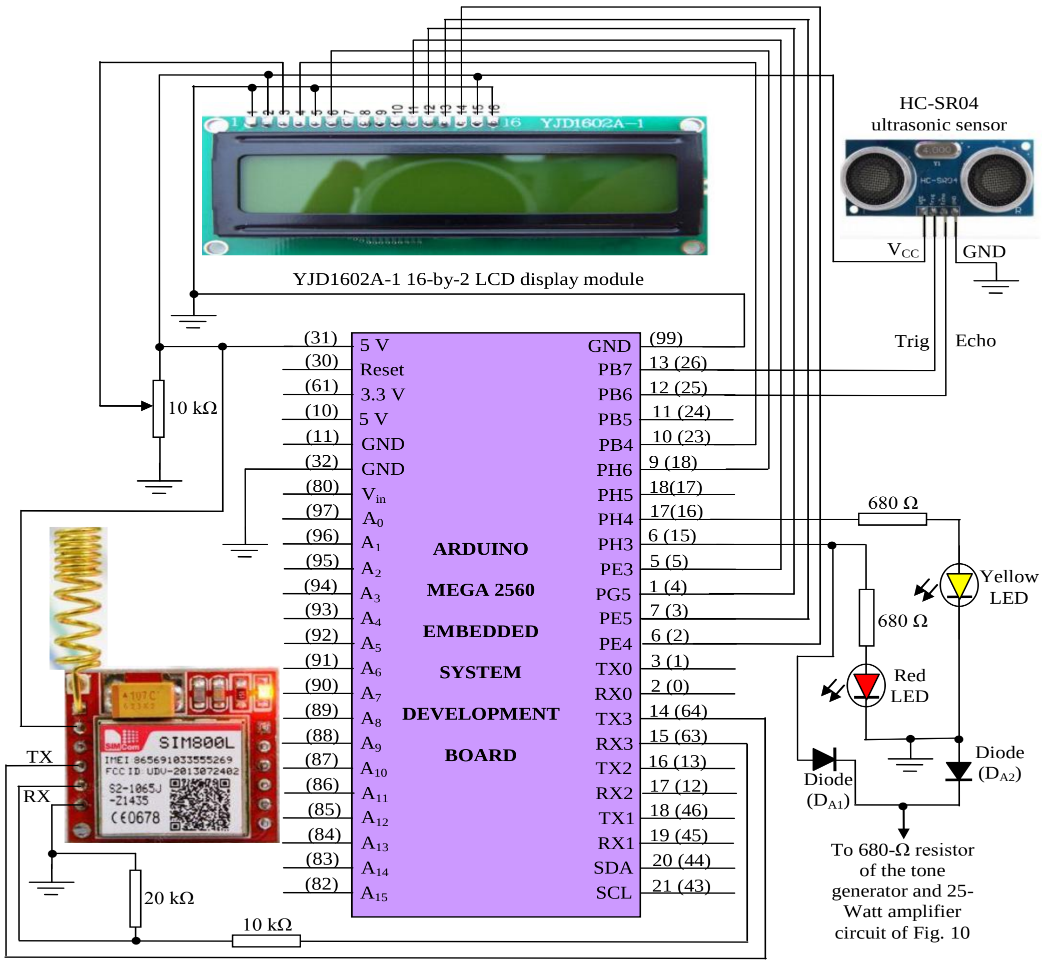

Figure 10. The SIM800L wireless connectivity and obstacle detection setup for the proposed integrated wireless security surveillance (IWSS) system

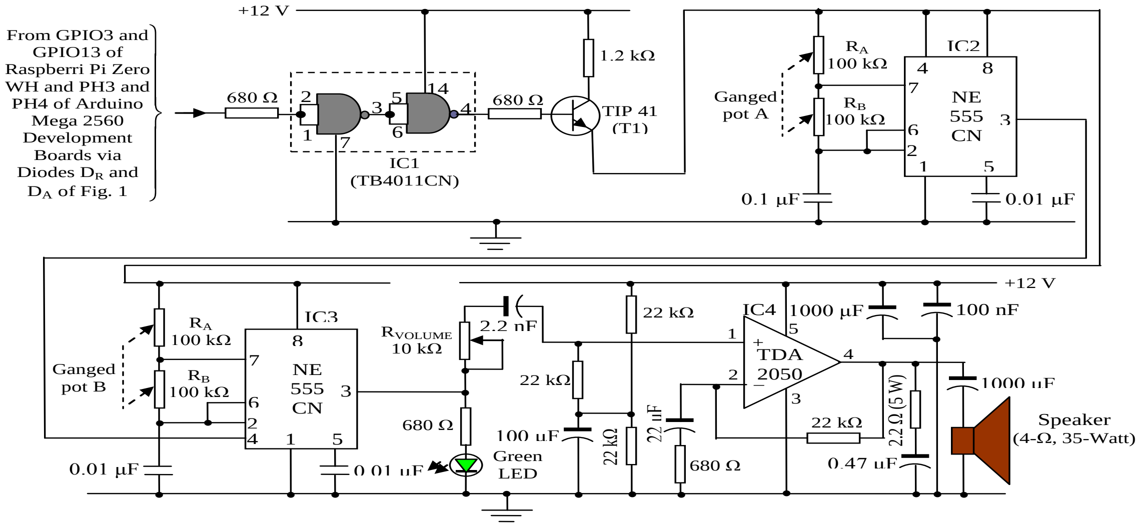

Figure 11. The complete circuit diagram of the electronic alarm unit with a 25-Watt audio amplifier system [17,19,30]

The wireless connectivity and bi-directional communication of the IWSS system is accomplished using the SIM800L GSM/GPRS S2-1065J module with a SIM (Subscriber Identification Number) with number 08114593935 from MTN Nigeria as the network service provider. The Arduino Mega 2560 has been configured and programmed in such a way that the SIM800L can send information (*"Status: Intrusion" "HUMAN IS HERE", "Day", "Date", Time*) via SMS. The SIM800L module sends test message to the specified phone number (07079535593) via the Internet whenever an object or human is detected at a range of 4 m and the electronic alarm system is activated. Whenever object is not detected, the yellow LED switched ON whereas when an object is detected the red LED is switched ON and the alarm system is triggered while the yellow LED turns OFF.

4.1.2. The Electronic Alarm System with a 25-Watt Audio Amplifier

The block diagram of the IWSS system was described in Section 1 with the introduction of the electronic alarm system incorporating a 25-Watt audio amplifier unit. Here, the complete circuit diagram electronic alarm system is shown in Figure 11 [17,19,30]. The electronic alarm system is built around two NE555CN timer ICs which forms the tone generator and a TDA2050 audio amplifier IC. Following the discussions in previous sections, whenever an obstacle or intruder is detected at a distance of 4 m from the ultrasonic sensor, the electronic alarm system is triggered.

In order to protect the Arduino Mega 2560 development board output ports, resistors are connected from the output ports to the CD4011BE to limit the current flowing through it as shown in Figure 11. Note that the output pin 4 of the CD4011BE NAND-gate configuration as shown in Figure 11 is used to bias transistor (TIP41 NPN transistor) via 680 Ω which delivers +12 V which serves as power supply voltages to the circuit of IC2. The circuit of IC2 is a NE555CN timer is wired as an astable multivibrator. The output taken from pin 3 of IC2 is high for one half of a cycle for one second and goes low for the next half cycle. When the output is low, IC3 is inhibited and the loud-speaker is off. During the next half cycle, the output is high. Thus, the output of IC2 oscillates at 1 kHz and this oscillating signal is amplified by a 25 Watts audio power amplifier built around IC4, TDA2050 and passed to the speaker situated on the far right bottom of Figure 11. When this electronic alarm is ON, it however signifies that an intruder has been detected at a distance of 4 m. The sound level (i.e. the volume) of the audio amplifier is regulated by the 10 kΩ potentiometer (R_{VOL}).

4.2. The Power System: Uninterruptible Power Supply with Backup Unit and Automatic Battery Charger

4.2.1. Block Diagram Description of the Power System

As discussed in Sections 4.1.1 and 4.1.2 using the block diagrams of Figures 9 and 10 respectively, for the proper operation of the IWSS system, the Raspberry Pi Zero WH and Arduino Mega 2560 embedded system development boards and associated components require +5 V while electronic alarm system with the audio power amplifier and associated components require 12-V for optimum performance.

The block diagram for the automatic two-way backup power system module supported with a two 12-V Li-Po rechargeable batteries connected in parallel as well as the battery-powered 5-V stabilized power supply unit and automatic battery charger with battery level indicator is shown in Figure 12 while the circuit diagrams of the stabilized dual power supply unit (SPSU) and battery-powered 5-V SPSU are shown in Figure 13 and Figure 14 respectively [18]. As it can be seen in Figure 5, the automation of the power supply module is controlled by two mechanical relays MR_1 and MR_2.

In the presence of public power supply, the SPSU of Figure 12 is activated and it delivers 12-V through diode D_P to MR_1 and 5-V through diode D_{W1} to MR_2 respectively to the automatic BMI machine for proper operation. Note that the 12-V from D_P: *1)* energizes MR_1 from normally-closed (NC) terminal to the normally-open (NO) terminal; and *2)* supplies stabilized 12-V that drives that BMI machine for proper operation.

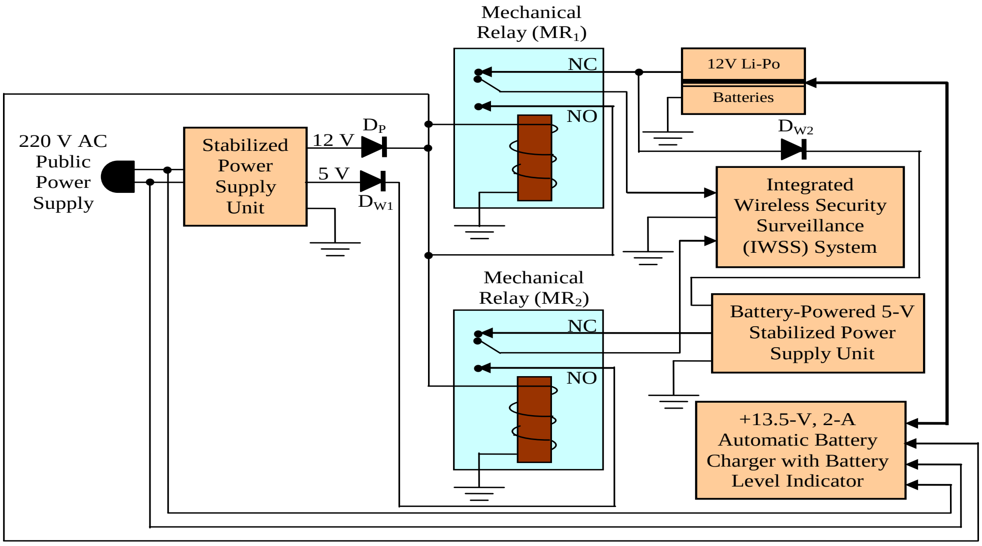

Figure 12. The block diagram of the proposed automatic two-way backup power supply module

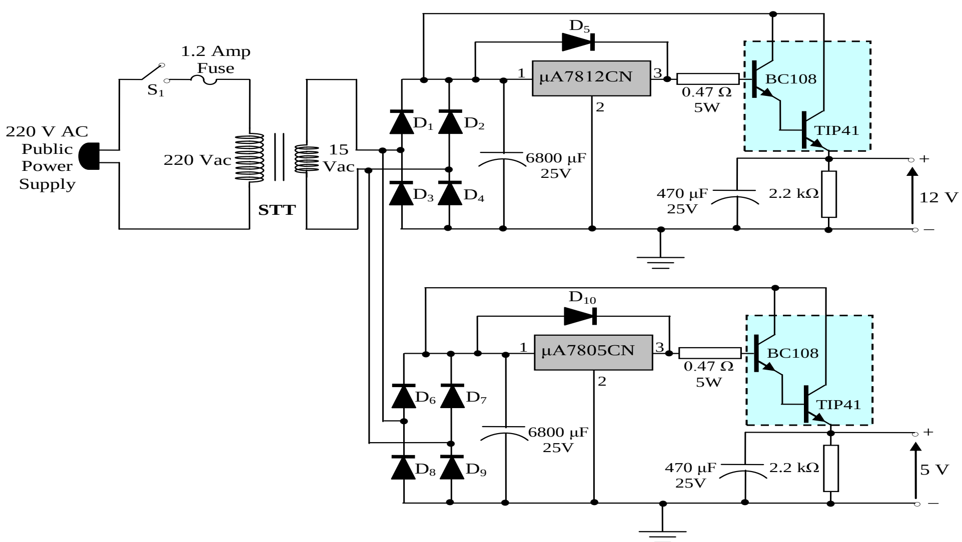

Figure 13. The circuit diagram of the stabilized dual (+12-V and +5-V) power supply unit based on the availability of public power supply [18]

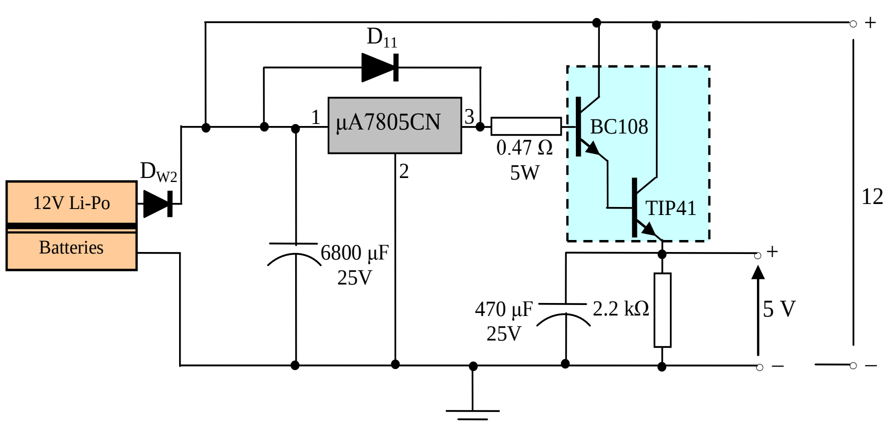

Figure 14. The circuit diagram of the dual (+12-V and +5-V) power supply units based on the charged two 12-V Li-Po batteries [18]

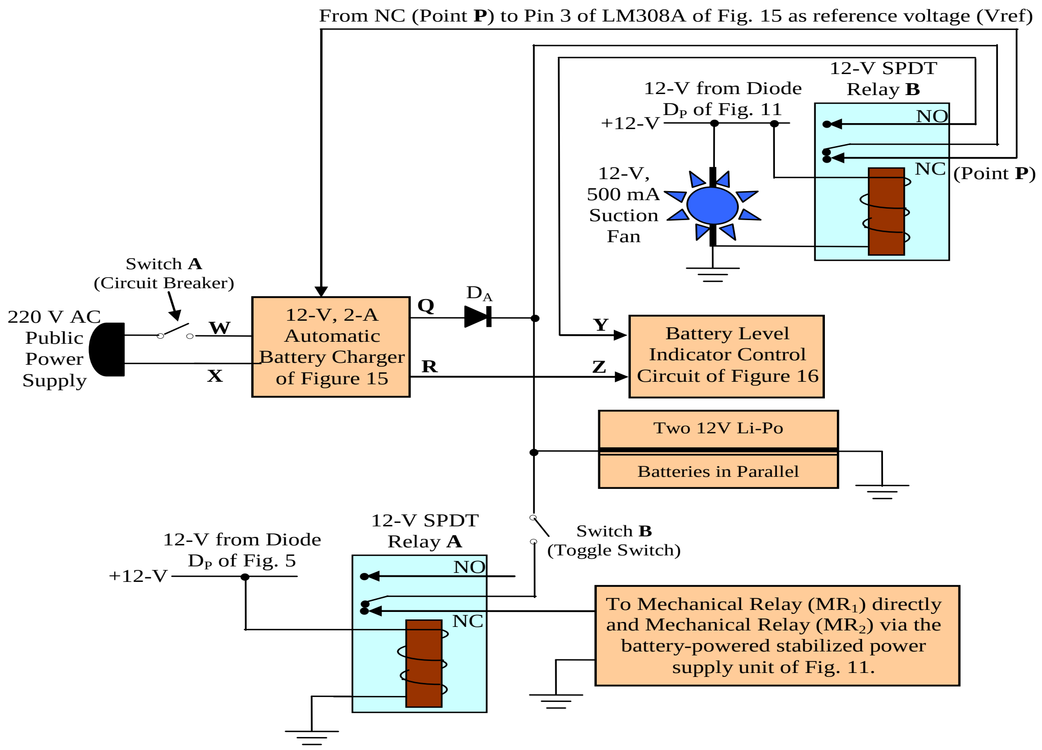

Figure 15. Block diagram of the automatic battery charger with battery level control unit and mechanical switching SPDT control relays

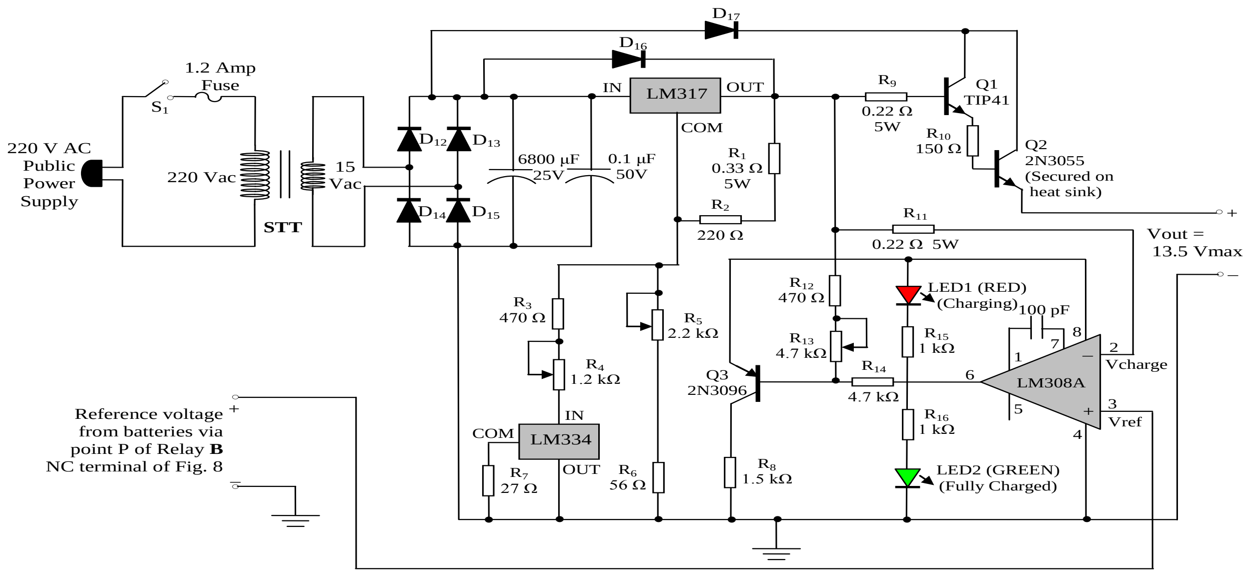

Figure 16. An automatic 12-Volt battery charger with automatic cut-off at full charge. Note that Vout = 13.5 Vmax is the output voltage to 12V Li-Po battery and Relay **B** shown in Figure 15 [31]

On the other hand, in the absence of public power supply, the output terminal of MR_1 and MR_2 automatically returns to the NC terminals; and the Li-Po battery supplies 12-V directly to the BMI machine and to the battery-powered 5-V stabilized power supply unit for the proper operation of the BMI machine. In this way, the BMI machine can be used both for in-door and out-door BMI measurements and monitoring.

To sustain the performance of the IWSS system in the absence of prolonged public power supply and to prolong the life span of the Li-Po batteries, an automatic battery charger with battery level indicator has been incorporated into the power system. The block diagram of the automatic battery charger with battery level indicator unit and electromechanical switching SPDT control relays is shown in Figure 15 and is discussed in the next sub-section 4.2.2.

4.2.2. Block Diagram Description of the Automatic Battery Charger with Battery Level Indicator Unit and Electromechanical Switching SPDT Control Relays

The block diagram of the automatic battery charger and all its sub-circuits are shown in Figure 15. Relay **A** is a 12V 10A

5-pin 2-output (one NC and one NO) electromechanical relay and it is driven by the +12-V biasing circuit of Figure 13 via diode D_P in Figure 12 only when PPS is available. Relay **B** is a 12V 10A 5-pin 2-output (one NC and one NO) electromechanical relay and it is driven by the same biasing circuit of Figure 13 via diode D_P in Figure 12 when the two Li-Po batteries is in used during the absence of PPS. The circuit of transformer STT, diodes D1-D4, voltage regulator µA7812CN and transistors BC108 and TIP41 is a biasing circuit that triggers Relay **A**. This circuit is driven only by the public power supply when available as shown in Figure 13. In a similar way, the circuit of Figure 14 is also a biasing circuit that is energized only in the absence of public power supply but provided that switch **B** is closed (i.e. the two Li-Po 12-V batteries are use in this operation). This circuit drives the 12-V 500 mA cooling suction fan and turns ON the battery voltage level indicator through Relay **B** in the NC position. The remaining two blocks: the automatic battery charger and the battery voltage level indicator are described in the next sub-sections 4.2.3.

Relay **A** plays a significant role in the overall switching of the load between the battery backup power system and the public power supply. In the presence of the public power supply with switch B ON, Relay **A** is triggered (switched to NO). At this instance, the IWSS system is power directly from the public power supply and the batteries are equally charged by the automatic battery charger. However, in the absence of the public power supply, the relay **A** and relay **B** are both switched OFF. Thus, input terminals of both relay are connected to their respective NC terminals. At this instant and with toggle switch **B** closed, the cooling fan and the battery voltage level indicator switches ON simultaneously and the IWSS system is powered by the battery backup system.

4.2.3. The Automatic Battery Charger and the Battery Voltage Level Indicator

The two circuits shown in Figure 16 and Figure 17 are the automatic 15 V-dc battery charger and the battery voltage level indicator respectively [31]. The battery charger designed for this inverter is a high performance charger with self regulating features. The main integrated circuit voltage regulator is the LM317 whose output voltage is fixed at 13.5 V using the resistors values defined by Equation (1) and shown in Figure 16 [32]:

$$V_{out} = 1.2 \times \left(1 + \frac{R_5 + R_6}{R_1 + R_2} \right) \quad (1)$$

The comparator sub-circuit is built around LM308A and transistor Q3. This circuit uses the batteries voltage from the NC terminal (Point P) of Relay **B** (Figure 15) as a reference voltage (V_{ref}) to control the charging of the batteries by comparing the V_{ref} against the charging voltage (V_{charge}). The batteries charging operation starts when the output of LM308A goes low (0 V), transistor Q3 turns OFF and transistor Q1 turns ON. When the batteries are fully charged, the output of LM308A goes high (+27 V), Q3 turns ON and Q1 turns OFF. The sensitivity of the comparison action is provided by the potentiometer (variable resistor R_{13}) which determines the ON-OFF switching of Q3.

As described in Coughlin and Driscoll [33], the comparator sub-circuit is configured in such a way that when V_{ref} is less than V_{charge}, the output of LM308A goes low (0 V), Q3 turns OFF and Q1 turns ON to charge the batteries with LED1 (RED) ON. When V_{ref} is equal to or greater than V_{charge}, the output of LM308A goes high (+12.5 V), Q3 turns ON and Q1 turns OFF to stop the charging process while LED2 (GREEN) turns ON indicating that the batteries are fully charged. The charging current is limited to 2 Amperes for smooth charging of the batteries. The charger switches ON and starts the charging process when the battery voltage drops below 10.5 V and switches OFF when the battery voltage reaches 12.5 V.

Also included within the battery charger circuit is a temperature compensation circuit built around LM334, which has been configured to switch OFF the charger when the internal temperature of the entire inverter system is greater than 45℃ (although it can traditionally withstand a temperature of 70℃) [31].

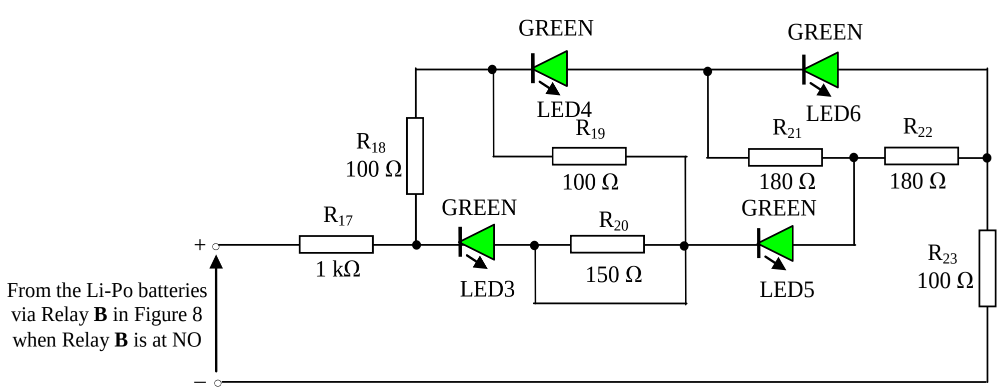

Figure 17. Battery voltage level indicator circuit [31]

Table 6. The processes, actions and the status of the IWSS system during testing

Processes	Actions	Status
Power is turned on in the circuit	The IWSS system is initialized and the camera as well as the proximity sensor enters standby mode.	The IWSS system is ready for surveillance
Motion is not sensed at distance greater than 4 m range	No intrusion/human detected	SMS not is sent to the specified phone number and the electronic alarm system is not activated
Motion is sensed at distance less than 4 m range	Intrusion/human is detected	SMS is sent to the specified phone number and the electronic alarm system activated
Motion is sensed at 2 m range	Intrusion/human is detected	SMS is sent to the specified phone number and the electronic alarm system activated
Motion is not sensed at 4 m range	No intrusion? Human detected?	SMS not is sent to the specified phone number and the electronic alarm system is not activated

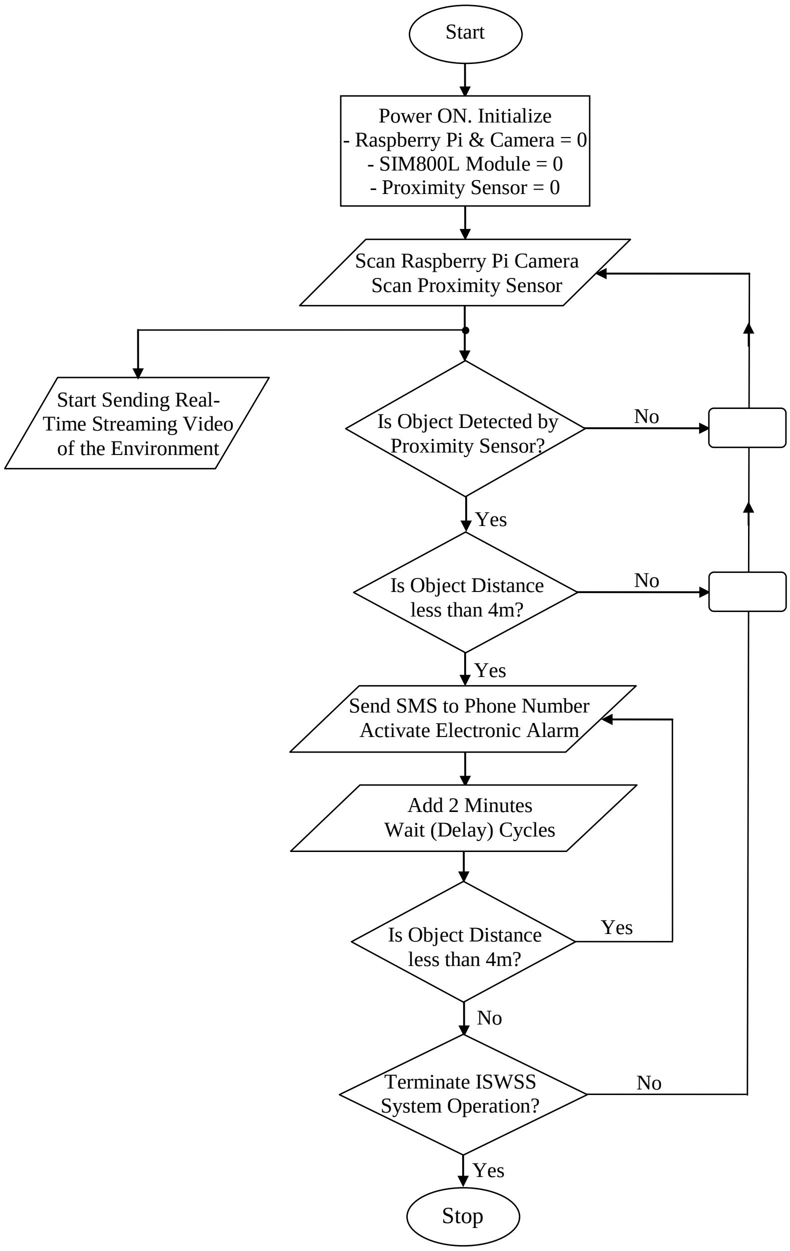

Figure 18. Flowchart for the operation of the IWSS system

The last major circuit is the battery voltage level indicator shown in Figure 17 and configured in such a way that all four LEDs turns ON when the two batteries are both fully charged but the LEDs begins to switch OFF one after the other starting from LED6 down to LED3 as the battery voltage drops from 12 V (typical) to 0V respectively.

4.3. IWSS System Integration, Deployment and Testing

The developed IWSS was deployed for security surveillance at the Department of Physics Electronics, The Federal University of Technology, Akure – Nigeria and its operation as well as its performance met the specific design objectives without human intervention.

The Arduino Mega 2560 controls the SIM800L GSM/GPRS S2-1065 module which sends only SMS to the specified mobile number based on the situation in the environment under surveillance should there be any form of intrusion while the Raspberry Pi Zero WH is used for streaming the real-time video of the environment under surveillance as viewed by the Raspberry Pi E305654 5-MP camera module.

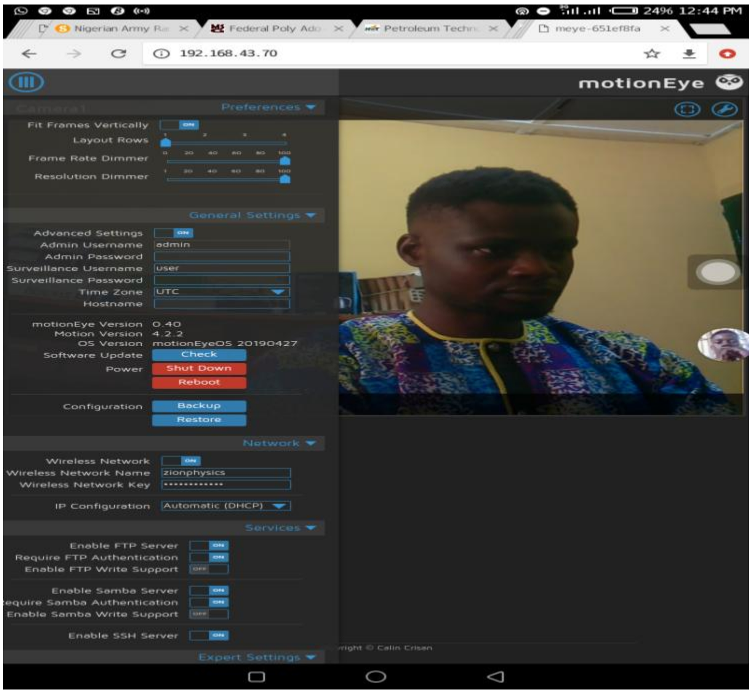

Figure 19. Motion detection in real-time video streaming using an android phone for Case 1

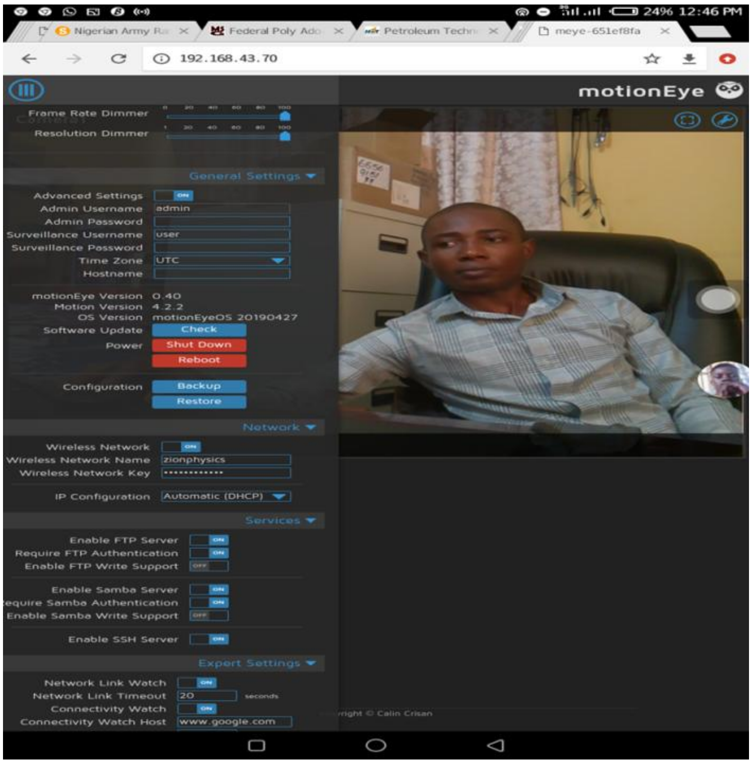

Figure 20. Motion detection in real-time video stream using an android phone for Case 2

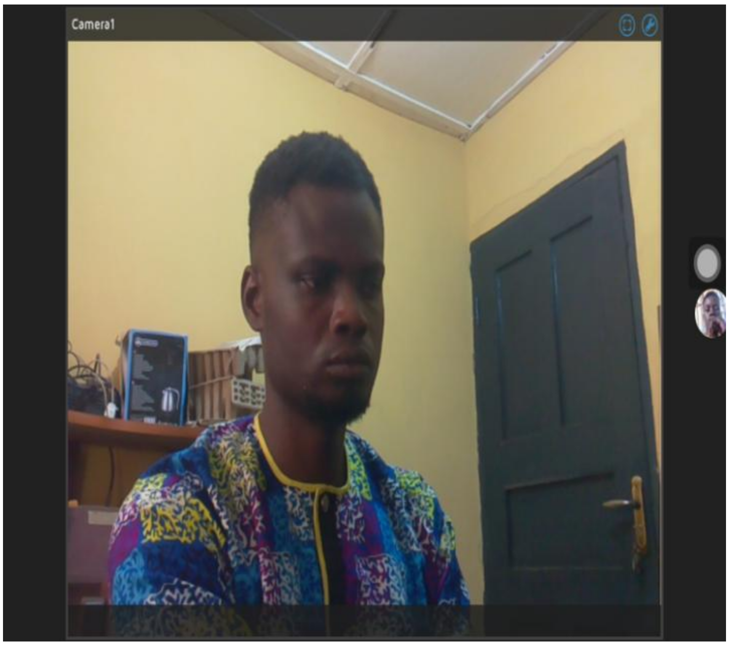

Figure 21. Motion detection in real-time video stream using an Hp Laptop for Case 3

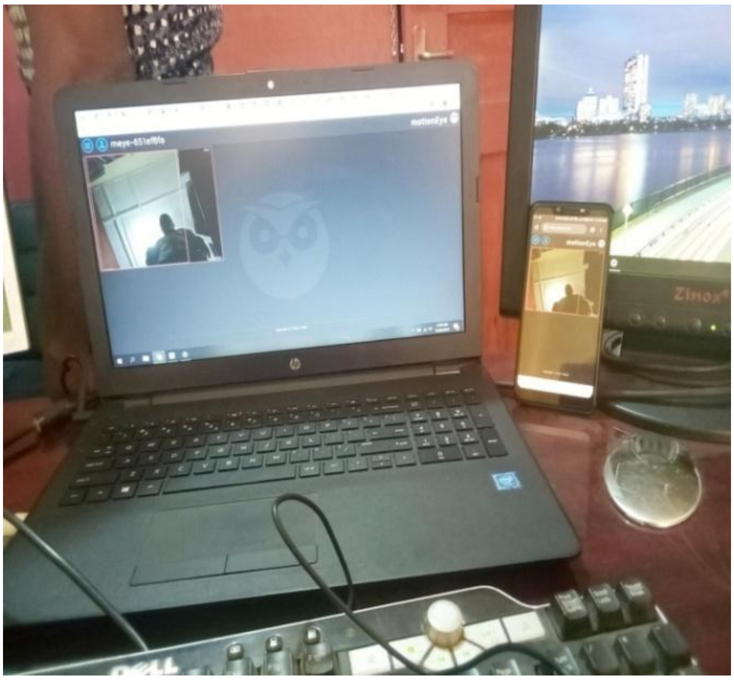

Figure 22. Motion detection in real-time video stream viewed simultaneously on android phone and Hp Laptop for Case 4

The process, action and the status of the IWSS system during testing is summarized in Table 6. The flowchart for the complete operation of the IWSS system is illustrated in Figure 18 in conjunction with Table 6. once the IWSS system is powered ON, the Raspberry Pi Zero WH embedded system development board, the raspberry Pi E305854 camera module, the Arduino Mega 2560 embedded system development board, the SIM800L GSM/GPRS S2-1065J module and the HC-SR04 proximity sensor are initialized. This is immediately followed by the scanning of the Raspberry Pi camera and the proximity sensor for object detection. The Raspberry Pi camera starts delivering real-time streaming videos to the destination IP address on both android mobile phone and the HP laptop (whether or not a moving object is detected by the camera). However, once any moving object is detected at a distance of less than 4 m, an SMS is sent to the specified destination phone number while the electronic alarm system is automatically activated as an indication of the presence of an intruder.

As long as the object is within a distance of less than 4 m from the proximity sensor, 2 minutes wait (delay) cycle is added while the electronic alarm system remains ON until the image distance is greater than 4 m from the proximity sensor. The operation of the IWSS system is automated in such a way that it remains in operation until it is switched off by the user as shown in the last decision block of Figure 18.

The automatic Li-Po battery backup power supply system with automatic 13.5-V battery charger with battery level indicator performed as expected and satisfied the design objectives. The software for the IWSS system are given in Appendix I (Software for implementing the obstacle avoidance HC-SR04 proximity sensor based on Ardiuno Mega 2560), Appendix II (Software for implementing the wireless connectivity based on Arduino Mega 2560 with SIM800L module) and Appendix III (Software for implementing the Raspberry Pi E305654 camera module based on Raspberry Pi Zero WH board).

During testing, the IWSS system yielded the expected results as shown in Figure 19 through Figure 21 for the four cases illustrated. The tests were carried out on HP laptop and Android mobile phone. Case 1 and Case 2 are based on HP laptop only as can be seen in Figure 19 and Figure 20 respectively; Case 3 is obtained using an android mobile phone only as shown in Figure 21; and Case 4 give the result using the HP laptop and Android mobile phone simultaneously as shown in Figure 22.

5. Conclusions

A novel integrated wireless security surveillance (IWSS) system has been designed, developed, constructed and deployed for security surveillance. The performance of the IWSS system has been investigated and operation of the system satisfies the design objectives. The results show that it can potentially be used for community based home security system by taking the advantages of wireless mesh network. Furthermore, it can eliminate blind spot, where the sensor and the camera can be placed at hidden places for enhanced detection and human face recognition. Multiple sensors and cameras deployment also can be placed at many locations. In addition, the system networking can be expanded throughout residential area by deploying additional open-mesh wireless mesh router, camera nodes, and sensor nodes. The low cost for the production and deployment of the IWSS system makes it an excellent product for the security surveillance of residential communities.

The developed IWSS system can be adapted for complete home and community as well as shopping mall automation for intrusion detection and notification. Rather than triggering the electronic alarm system for all human faces, facial image recognition algorithms can be incorporated to trigger the alarm for only unknown facial images [34–36]. The developed IWSS system also be adapted for much longer distances by the use of LiDAR sensors [17].

Appendix

Appendix 1. Software for implementing the obstacle avoidance HC-SR04 proximity sensor based on Ardiuno Mega2560

```
/*
HC-SR04 ultrasonic distance sensor]
VCC to arduino 5v GND to arduino GND
Echo to Arduino pin 13 Trig to Arduino pin 12
Red POS to Arduino pin 11
Green POS to Arduino pin 10
560 ohm resistor to both LED NEG and GRD power rail
*/
#include <LiquidCrystal.h>
LiquidCrystal lcd(10, 9, 5, 4, 3, 2);
// Initializes the LCD Ports
#define trigPin 13
#define echoPin 12
#define led 7
// Triggers Yellow LED
#define alarm 6
// Triggers Red LED and the Electronic Alarm System
void setup() {
  Serial.begin (9600);
  pinMode(trigPin, OUTPUT);
  pinMode(echoPin, INPUT);
  pinMode(led, OUTPUT);
  pinMode(alarm, OUTPUT);
}
void loop() {
  float duration, distance;
  digitalWrite(trigPin, LOW);
```

```
// This is where the LED On/Off happens
    digitalWrite(led,HIGH);
// When the Red condition is met,
// the yellow LED should turn off
  digitalWrite(alarm,LOW);
  lcd.begin(16, 2);
  lcd.print("within range   ");
  lcd.setCursor(0,1);
  lcd.print("limit   ");
    delay(500);
}
  else {
    digitalWrite(led,LOW);
    digitalWrite(alarm,LOW);
    lcd.print(distance);
    lcd.print(" cm ");
    delay(2000);
  }
  if (distance >= 400 || distance <= 0){
    digitalWrite(led,LOW);
// When the Red condition is met,
// the Green LED should turn off
  digitalWrite(alarm,HIGH);
  lcd.print(" obstacle alert ");
    delay(500);
  }
  else {
```

```
// Added this line
  delayMicroseconds(2);
// Added this line
  digitalWrite(trigPin, HIGH);
//   delayMicroseconds(1000); - Removed this line
  delayMicroseconds(10);
// Added this line
  digitalWrite(trigPin, LOW);
  duration = pulseIn(echoPin, HIGH);
  distance = (duration/2) / 29.1;
  if (distance < 400) {
      digitalWrite(led,HIGH);
  // When the Red condition is met,
  // the Green LED should turn off
    digitalWrite(alarm,LOW);
    lcd.print(distance);
      lcd.print(" cm");
      delay(500);
    }
    delay(500);
    lcd.clear();
  }
```

Appendix 2. Software for implementing the wireless connectivity based on Arduino Mega 2560 with SIM800L module

```
#include <SoftwareSerial.h>
#include <LiquidCrystal.h>
#include <SPI.h>
#include <SD.h>
#include <Wire.h>
#include <DS3231.h>
DS3231  rtc(SDA, SCL);
SoftwareSerial SMS(6, 7); //RX=6,TX=7
LiquidCrystal lcd(A0,A1,A2,A3,4,5);
int pir = 3;
int read;
int sense;
void setup() {
// put your setup code here, to run once:
pinMode(pir,INPUT);
   SMS.begin(9600);
// Setting the baud rate of GSM Module
   Serial.begin(9600);
   SPI.begin();
// Initialize the rtc object
  rtc.begin();

// The following lines can be uncommented to
// set the date and time
// rtc.setDOW(THURSDAY);
// Set Day-of-Week to SUNDAY
// rtc.setTime(11, 28, 50);
// Set the time to 12:00:00 (24hr format)
// rtc.setDate(14, 6, 2018);
// Set the date to January 1st, 2014
// pinMode(8,OUTPUT);
  lcd.begin(20, 4);
  lcd.setCursor(5, 0);
  lcd.print("--@Topmandreams-- ");
  lcd.setCursor(3, 1);
  lcd.print("ALERT SYSTEM");
  SMS.println("AT+CMGF=1");
  delay(2000);
  lcd.clear();
  lcd.display();
  lcd.setCursor(2, 0);
  lcd.print("SECURITY ALERT");
  lcd.setCursor(6, 1);
  lcd.print("SYSTEM");
  lcd.setCursor(4, 2);
void loop() {
 // lcd.display();
 // Send Day-of-Week
  Serial.print(rtc.getDOWStr());
  Serial.print(" ");
 // Send date
  Serial.print(rtc.getDateStr());
  Serial.print(" -- ");
 // Send time
   Serial.println(rtc.getTimeStr());
    Serial.print(" ");
    Serial.println();
    lcd.setCursor(0, 0);
    lcd.print(rtc.getDateStr());
    lcd.setCursor(10, 0);
    lcd.print("|");
    lcd.setCursor(11, 0);
    lcd.print(rtc.getTimeStr());
    lcd.setCursor(3, 1);
if (Serial.available()>0)
       Serial.read();
       delay(100);
  lcd.print("--SYSTEM ARMED--");
  /*
          lcd.setCursor(0, 2);
          lcd.print("Status:");
          lcd.setCursor(7, 2);
          lcd.print("Position:No Intrusion");
  */
read = digitalRead(pir);
if(read==HIGH){
   SMS.println("AT+CMGS=\"08114593935\"\r");
 // Replace x with mobile number
delay(100);
   SMS.println("Status:Intrusion");
   SMS.println("HUMAN IS HERE");
   SMS.print("Day:");
   SMS.println(rtc.getDOWStr());
            SMS.print("Date:");
            SMS.println(rtc.getDateStr());
            SMS.print("Time:");
            SMS.println(rtc.getTimeStr());
            delay(100);
            SMS.println((char)26);
// ASCII code of CTRL+Z
```

```
  lcd.print("WITH SMS...");
  lcd.setCursor(0, 3);
  lcd.print("Initializing");
  delay(500);
  lcd.setCursor(13, 1);
  lcd.print('!');
  delay(500);
  lcd.setCursor(14, 1);
  lcd.print('!');
  delay(500);
  lcd.setCursor(15, 1);
  lcd.print('!');
  delay(500);
  lcd.setCursor(16, 1);
  lcd.print('!');
  delay(500);
  lcd.setCursor(17, 1);
  lcd.print('!');
  delay(500);
  lcd.setCursor(18, 1);
  lcd.print('!');
  delay(500);
  lcd.clear();
```

```
            lcd.display();
            lcd.setCursor(0, 2);
            lcd.print("Status:");
            lcd.setCursor(7, 2);
            lcd.print("Intrusion");
            lcd.setCursor(0, 3);
            lcd.print("Position:");
            lcd.setCursor(9, 3);
            lcd.print("Inside");
            lcd.setCursor(15, 3);
            lcd.print("    ");
            lcd.setCursor(16, 2);
            lcd.print("    ");
            delay(2000);
            Serial.println("HUMAN IS HERE");
else{
  lcd.setCursor(0, 2);
  lcd.print("Status:");
  lcd.setCursor(7, 2);
  lcd.print("No Intrusion");
  lcd.setCursor(0, 3);
  lcd.print("Position:Outside");
  Serial.println("NO HUMAN");
}
```

Appendix 3. Software the Raspberry Pi E305654 camera module based on Raspberry Pi Zero WH board

```
 import pygame
  import pygame.camera
  # Captured image dimensions. It should be less than or equal to the
  # maximum dimensions acceptable by the camera.
 width = 320
 height = 240
  # Initializing PyGame and the camera.
 pygame.init()
 pygame.camera.init()
  # Specifying the camera to be used for capturing images.
  # If there is a single camera, then it have the index 0.
 cam =
pygame.camera.Camera("C:/Users/hp/Documents/raspberrypi/dev/vid
eo0",(width,height))
# Preparing a resizable window of the specified size for displaying
  # the captured images.
window =
pygame.display.set_mode((width,height),pygame.RESIZABLE)
  # Starting the camera for capturing images.
 cam.start()
  # Capturing an image.
 image = cam.get_image()
  # Stopping the camera.
 cam.stop()
  # Displaying the image on the window starting from
  # the top-left corner.
 window.blit(image,(0,0))
  # Refreshing the window.
 pygame.display.update()
  # Saving the captured image.
 pygame.image.save(window,'PyGame_image.jpg')
import pygame
```

```
# Preparing a resizable window of the specified size for displaying
# the captured images.
window = pygame.display.set_mode((width, height),
pygame.RESIZABLE)
# Starting the camera for capturing images.
cam.start()
for im_num in range(0, 2000):
    print("Image : ", im_num)
    # Capturing an image.
    image = cam.get_image()
    # Displaying the image on the window starting from
    # the top-left corner.
    window.blit(image, (0, 0))
    # Refreshing the window.
    pygame.display.update()
    # Saving the captured image.
    pygame.image.save(window, './pygame_images/image_' +
str(im_num) + '.jpg')
# Stopping the camera.
cam.stop()
import skimage.io
import os
import numpy
dir_files = os.listdir('./pygame_images/')
bg_image = skimage.io.imread(fname=dir_files[0], as_grey=True)
for k in range(1, len(dir_files)):
    fname = dir_files[k]
    im = skimage.io.imread(fname=fname, as_grey=True)
    bg_image = bg_image + im
bg_image = bg_image/(len(dir_files))
bg_image_bin = bg_image > 0.5
skimage.io.imsave(fname='bg_model.jpg', arr=bg_image)
```

```
import pygame.camera
# Captured image dimensions. It should be less than or equal
# to the maximum dimensions acceptable by the camera.
width = 320
height = 240
# Initializing PyGame and the camera.
pygame.init()
pygame.camera.init()
# Specifying the camera to be used for capturing images.
# If there is a single camera, then it has the index 0.
cam = pygame.camera.Camera("C:/Users/hp/Documents/raspberrypi/dev/video0", (width, height))
```

```
skimage.io.imsave(fname='bg_model_bin.jpg', arr=bg_image_bin*255)
bg_num_ones = numpy.sum(bg_image_bin)
test = skimage.io.imread(fname="./pygame_images/image_800.jpg", as_grey=True)
test_bin = test > 0.5
test_num_ones = numpy.sum(test_bin)
print("Num 1s in BG    :", bg_num_ones)
print("Num 1s in Test :", test_num_ones)
if(abs(test_num_ones-bg_num_ones) < 5000):
print("Change.")
```

REFERENCES

[1] M. N. M. Noor, "Community based home security system using wireless mesh network", *International Journal of Academic Research*, vol. 5, no. 5, pp. 73 – 79, 2013.

[2] A. Sen, L. Mahajan, P. Kuwor and A. Khatri, "Wireless security systems", *International Journal of Current Research*, vol. 8, no. 5, pp. 30389 – 30392, 2016.

[3] M. Channing, "House Alarms", *COUNSEL*, vol. 26, no. 1, pp. 141 – 143, 2010.

[4] Dickey, N. Banks, D. and Sukittanon, S. "Home Automation Using Cloud Network and mobile devices", *International Journal of Internet of Things*, vol. 2, no. 1, pp. 5 – 11, 2013.

[5] A. Bottoms, "*What works in Multi-Agency Partnerships: Lessons from the Five Towns*", 1991.

[6] C. Reinisch, "*Wireless Communication in Home and Building Automation*", Master Thesis, Viennia University of Technology, Feb 2007.

[7] J. R. Delaney and A. Colon, "*The best smart home security systems for 2021*", PCMAG Digital Group, Updated 10th September, 2021. Available [Online]: https://www.pcmag.com/picks/the-best-smart-home-security-systems.

[8] K. Ashton, "That 'Internet of Things' Thing, in the real world things matters more than ideas", *RFID Journal*, 22nd June, 2009.

[9] F. Mattern and C. Floerkemeier, "From the Internet of Computers to the Internet of Things". *Informatik- Spektrum*, vol. 33, no. 2, pp. 107 – 121, 2010.

[10] E. Brown, (13th September 2016), "*Who Needs the Internet of Things*?". Retrieved 12th June, 2021. Available [Online] www.Linux.com.

[11] J. Höller, V. Tsiatsis, C. Mulligan, S. Karnouskos, S. Avesand and D. Boyle, "*From Machine-to-Machine to the Internet of Things: Introduction to a New Age of Intelligence*", Elsevier, 2014.

[12] A. Zanella, N. Bui, A. Castellani, L. Vangelista and M. Zorzi, "Internet of Things for Smart Cities", *IEEE Internet of Things Journal*, vol. 1, no. 1, pp. 22 – 32, 2015.

[13] J. Hwang and Y. H. Choe, "*Smart Cities Seoul: A Case Study*", ITU-T Technology Watch. Retrieved 15th July, 2021.

[14] G. Honeywell, "How to Plan and install an Intruder Alarm", *HSC-Pro Electrician*, vol. 1, no. 1, pp. 1 – 8, 2009.

[15] L. Louis, "Working Principle of Arduino and Using It as a Tool For Study And Research", *International Journal of Control, Automation, Communication and System*, vol. 1, no. 2, pp. 21 – 24, 2016.

[16] Elec Freaks, HC-SR04 Ultrasonic Sensor User Guide. Retrieved on the 26th of May 2021. Available [Online]: www.elecfreaks.com.

[17] V. A. Akpan and S. A. Eyefia, "The Development of an Enhanced Obstacle Detection System with Alarm Based on the Internet-of-Things", *African Journal of Computing and ICT*, pp. 1 – 10, 2021. [Accepted]

[18] I. E. Owolabi, V. A. Akpan and O. P. Oludola, "A Low-Cost Automatic Body Mass Index Machine: The Design, Development, Calibration, Testing and Analysis", *International Journal of Biomedical and Clinical Sciences*, vol. 6, no. 3, pp. 100 – 119, 2021. Available [Online]: http://www.aiscience.org/journal/paperInfo/ijbcs?paperId=4688.

[19] V. A. Akpan and S. A. Eyefia, "Autonomous Vehicle with Machine Vision and Integrated Sensor Suit Based on Internet-of-Things Technologies", *International Journal of Electronics Engineering and Computer Science*, vol. 6, no. 3, pp. 18 – 30, 2021. Available [Online]: http://www.aiscience.org/journal/paperInfo/ijeecs?paperId=4702.

[20] 16x2 LCD Module (2016). Retrieved on the 16th of September 2017. Available [Online]: https://components101.com/16x2-lcd-pinout-datasheet.

[21] ArduinoMega Datasheet (Retrieved 17th September, 2017,). Available [Online]:: https://www.arduino.cc/en/Main/arduinoboardmega .

[22] Raspberry Pi Zero GPIO Pinout, Specifications and Programming language. Retrieved on 15th May, 2021. Available [Online]: https://www.etechnophiles.com/raspberry-pi-zero-gpio-pinout-specifications-programming-language/ .

[23] What is Raspberry Pi Zero? Pinout, Specs, Projects & Datasheet. Retrieved on 15th May, 2021. Available [Online]: https://www.theengineeringprojects.com/2021/03/what-is-raspberry-pi-zero-pinout-specs-projects-datasheet.html.

[24] Raspberry PI 5MP Camera Board Module, Retrieved 31st July,

2021. Available [Online]: https://www.amazon.com/Raspberry-5MP-Camera-Board-Module/dp/B00E1GGE40.

[25] Raspberry Pi Camera Board v1.3 (5MP, 1080), Retrieved 31st July, 2021. Available [Online]: https://uk.pi-supply.com/products/raspberry-pi-camera-board-v1-3-5mp-1080p.

[26] Getting Started With the Camera Module, Retrieved 31st July, 2021. Available [Online]: https://projects.raspberrypi.org/en/projects/getting-started-with-picamera/0.

[27] Send Receive SMS & Call with SIM800L GSM Module & Arduino. Retrieved on 25th June, 2021. Available [Online]: https://lastminuteengineers.com/sim800l-gsm-module-arduino-tutorial/.

[28] Sim800l EVB GSM/GPRS module. Retrieved on 25th June, 2021. Available [Online]: http://hub360.com.ng/product/standard-sim800l-module/.

[29] SMS Controller by Using SIM800L V2. Retrieved on 25th June, 2021. Available [Online]: https://www.instructables.com/SMS-Controller-by-Using-SIM800L-V2/.

[30] V. A. Akpan, S. A. Eyefia and G. M. Adewumi, "An IoT-Based Real-Time Embedded System for Gas-Smoke Detection with Automatic Electronic Alarm System", *International Journal of Electronics Engineering and Computer Science*, vol. 6, no. 2, pp. 7 – 17, 2021. Available [Online]: http://www.aiscience.org/journal/paperInfo/ijeecs?paperId=4686.

[31] V. A. Akpan and T. Ewetumo, "Design, Development and Construction of a Low Cost Automatic 2 KVA Inverter System", *Global Journal of Pure and Applied Sciences*, vol. 16, no. 1, pp. 141 – 149, 2010. Available [Online]: https://www.ajol.info/index.php/gjpas/article/view/62823/507297.

[32] S. Franco, "*Design with Operational Amplifier and Analog Integrated Circuits*", 2nd Ed., McGraw-Hill Books Company, Singapore, pp. 43 – 441, 1998.

[33] R. F. Coughlin and F. F. Driscoll, "*Operational Amplifiers and Linear Integrated Circuits*", 4th Ed., Prentice-Hall International Edition, New Delhi, India, pp. 215 – 226, 1989.

[34] V. A. Akpan and R. A. O. Osakwe, "Face image processing, analysis and recognition algorithms for enhanced optimal face recognition systems design: A comparative study", *African Journal of Computing and ICTs*, vol. 2, no. 2, pp. 21 – 40, 2009.

[35] V. A. Akpan, J. B. Agbogun, M. T. Bablola and B. A. Oluwade, "Radial basis function neuroscaling algorithm for efficient facial image recognition", *Machine Learning Research*, Vol. 2, no. 4, pp. 152 – 168, 2018.

[36] V. A. Akpan, "Radial Basis Function Neural Network-Based Adaptive SIFT-SURF Algorithms for Facial Image Features Extraction", *African Journal of Computing and ICT*, pp. 1 – 15, 2021. [Accepted]