MINI PROJECT 2 (19EI6PWMP2)
ON
"Smart Burglar Alarm"

A MINI PROJECT 2 REPORT

Submitted by

SUDARSHANA.S.RAO (1BM18EI053)
SUDAMSHU.S.RAO (1BM18EI052)
VIGNESH.V (1BM18EI062)

Submitted towards the fulfillment of requirements for completion of
Mini Project 2 (19EI6PWMP2)
IN
ELECTRONICS AND INSTRUMENTATION ENGINEERING

Under the Guidance of

Prof. Vani A
Assistant Professor
Department of Electronics and Instrumentation Engineering

DEPARTMENT OF ELECTRONICS AND INSTRUMENTATION ENGINEERING
B. M. S. COLLEGE OF ENGINEERING
(Autonomous Institution under VTU)
BENGALURU-560019

2020-2021

“Smart Burglar Alarm”

MINI PROJECT 2 REPORT

Submitted by,
SUDARSHANA.S.RAO (1BM18EI053)
SUDAMSHU.S.RAO (1BM18EI052)
VIGNESH.V (1BM18EI062)

Submitted towards the fulfillment of requirements for completion of

Mini Project 2 (19EI6PWMP2)

in

ELECTRONICS AND INSTRUMENTATION ENGINEERING

B.M.S. COLLEGE OF ENGINEERING

Under the guidance of

Prof. VANI A
Assistant Professor
B.M.S. College
of Engineering
Bangalore
560019

Department of Electronics and Instrumentation Engineering
B.M.S. COLLEGE OF ENGINEERING
Bengaluru - 19
Bull temple road,
Bangalore -560019

2020-2021

B.M.S. COLLEGE OF ENGINEERING

(Autonomous College Under VTU)

Department of Electronics and Instrumentation Engineering

BONAFIDE CERTIFICATE

This is to certify that the mini project 2 report titled, "**Smart Burglar Alarm"** is a bonafide work carried out by **Sudarshana.S.Rao (1BM18EI053), Sudamshu.S.Rao (1BM18EI052) and Vignesh.V (1BM18EI062),** in Submitted towards the fulfillment of requirements for completion of Mini Project 2 (19EI6PWMP2***)*** during the academic year 2020-2021.

Prof. Vani A
Project Guide
Assistant Professor
Department of EIE, BMSCE

Dr. Veena N Hegde
Professor & H.O.D
BMSCE

Dr. B V Ravishankar
Principal

Internal Examiner

External Examiner

B.M.S. COLLEGE OF ENGINEERING
BANGALORE -560019
<u>BONAFIDE CERTIFICATE FROM THE GUIDES</u>

Department: Electronic and Instrumentation Engineering
Candidates Degree Registered For: B.E
Name of the Guide: Prof. Vani A

CANDIDATE DETAILS:

SL. NO	Student Name	USN	Student Signature
1	SUDARSHANA.S.RAO	1BM18EI053	
2	SUDAMSHU.S.RAO	1BM18EI052	
3	VIGNESH.V	1BM18EI062	

Certified that this mini project 2 titled “Smart Burglar Alarm” the candidates have carried out the project to my satisfaction. This 6th semester academic year 2020-21 dissertation report was thoroughly scrutinized and corrected by me.
All corrections are incorporated by the students. The work is original and the mini project 2 report is the final one and of high standard. I duly certify the same.

SL NO	Guide Name	Dept/Organization/Designation	Signature
1	Prof. Vani A	Assistant Professor Department of Electronics and Instrumentation Engineering **B.M.S. COLLEGE OF ENGINEERING** Bull Temple Road, Bangalore 560019	

Date:

Seal:

<u>Guide/s Approval Certificate for Incorporating all corrections in the mini project 2 report</u>

We **Sudarshana.S.Rao, Sudamshu.S.Rao** and **Vignesh.V** students of 6th semester B.E in Electronics and Instrumentation, BMSCE, Bangalore, hereby declare that the mini project 2 work entitled **"Smart Burglar Alarm"** has been carried out under supervision of **Prof. Vani A** of Electronics and Instrumentation Department towards the fulfillment of requirements for the completion of mini project 2 (19EI6PWMP2).

SUDARSHANA.S.RAO

SUDAMSHU.S.RAO

VIGNESH.V

<u>Guide Signature:</u>

Prof. VANI A

<u>HOD Signature:</u>

Dr. VEENA N. HEGDE

ACKNOWLEDGEMENT

The completion of this undertaking could not have been possible without the assistance and participation of so many people. Their contributions are sincerely appreciated and greatly acknowledged. The team would like to express their deep appreciation particularly to the following people.

We extend our heartfelt thanks to the principal Dr. Ravishankar B. V. and the management and staff of BMSCE for providing us with necessary infrastructure that enables us to implement anything our engineering minds desire.

Words cannot describe our gratitude to Dr. Veena N Hegde, HOD, Department of EIE, BMSCE for her constant support throughout this journey.

We wish to express our sincere acknowledgement to our guide Prof. Vani A, Assistant Professor, Department of EIE, BMSCE for her valuable suggestions and guidance throughout the course of our project. We are grateful for her belief in us for the implementation of a project of this magnitude. We are truly grateful for her constant support and for allowing us to work flexibly which enabled us to set our own deadlines and objectives.

A large amount of material has been obtained from online journals, textbooks and numerous other scholarly and voluminous sources. We thank all those unseen faces for their invaluable contribution to our venture.

We thank our beloved parents on whose blessings we live and thrive. It is their prayers that have helped us translate our efforts into fruitful achievements.

Abstract

Burglary is an unwanted event that no one wants to happen in their premises. However, it is difficult for anyone to be alert at all the times to protect their home and with the present system available a home owner can react to the burglary post the occurrence. This simple device Smart Burglar Alarm helps in this situation by raising an Alarm when there is an unwanted intrusion in the secured premises and thus allowing the home owner to prevent the event of burglary.

The Smart Burglar Alarm is based on a passive infrared (PIR) sensor which is installed at the entrance of the house. When an intruder breaks in the house, the system activates a buzzer connected to the system and a message will be sent to the mobile phone of the home owner via a GSM module. By hearing the buzzer and by reading the text message in the mobile phone, the resident of the home can get alert about the intrusion and a possible burglary. The buzzer buzzes till the alarm system is turned OFF. Then the doors of the house will automatically lock so that the burglar doesn’t escape.

As another line of defense, a surveillance camera is used which is connected to a computer system where the live feed will be stored. In the computer system a machine learning algorithm detects the presence of any intruders using open cv2 module, histogram of oriented gradients and support vector machine modules which again sends a message to the mobile phone of the home owner. As a result, the home owner will be made aware of the burglary and therefore, the burglary can be stopped. In case of a burglar detection all the doors of the home will be locked so that the burglar cannot escape and thus the concerned authorities can apprehend the burglar.

Table of contents

Table of images

List of tables

Chapter 1-

Introduction

In this fast-moving digital world, everything is being automated. Home automation industry is growing rapidly and security should be given the utmost importance. As a result, the Smart Burglar Alarm is an imperative system in every house. The Smart Burglar Alarm system is a very simple system which makes use of an Arduino Uno board (R3 CH340G ATmega328p), Passive Infrared(PIR) Sensor (HC-SR501), GSM Module (SIM 900a), Camera (CCTV surveillance) and a 5V buzzer. The Smart Burglar Alarm can be considered as the first step towards home automation. [1]

The smart burglar alarm is a set of interconnected devices that is aimed to protect an object, usually a facility, against intruders and to notify the owner or/and the monitoring station/center of any violation of the protected zones. [2]

A major benefit of installing a burglar alarm to the property is that it can deter a break-in before it happens. By having a clear burglar alarm and or security system installed in a home, it makes it one step harder for an intruder to break-in and they are likely to move on to an easier target. [3]

Modern alarm security systems provide the ability to monitor the activities happening on the property when home owner is away. There are security systems that have an add-on feature of detecting fire or smoke. These systems will instantly notify home owner during emergencies like a fire breakout, gas leak or a flood emergency. [4]

The main and most essential reason to invest in a security system is to deter crime in your home. According to the Greenwich Study of Residential Security report, homes without a security system are 2.7 to 3.5 times more likely to be subject to a burglary. [5]

This Smart Burglar Alarm will alert the home owner of a burglary and thereby preventing the burglary from happening rather than reacting to the situation post the event of burglary. This is one of the crucial features of this mini project which several existing alarms currently are not addressing. [6]

The below figure illustrates the system architectural block diagram of the Smart Burglar Alarm:

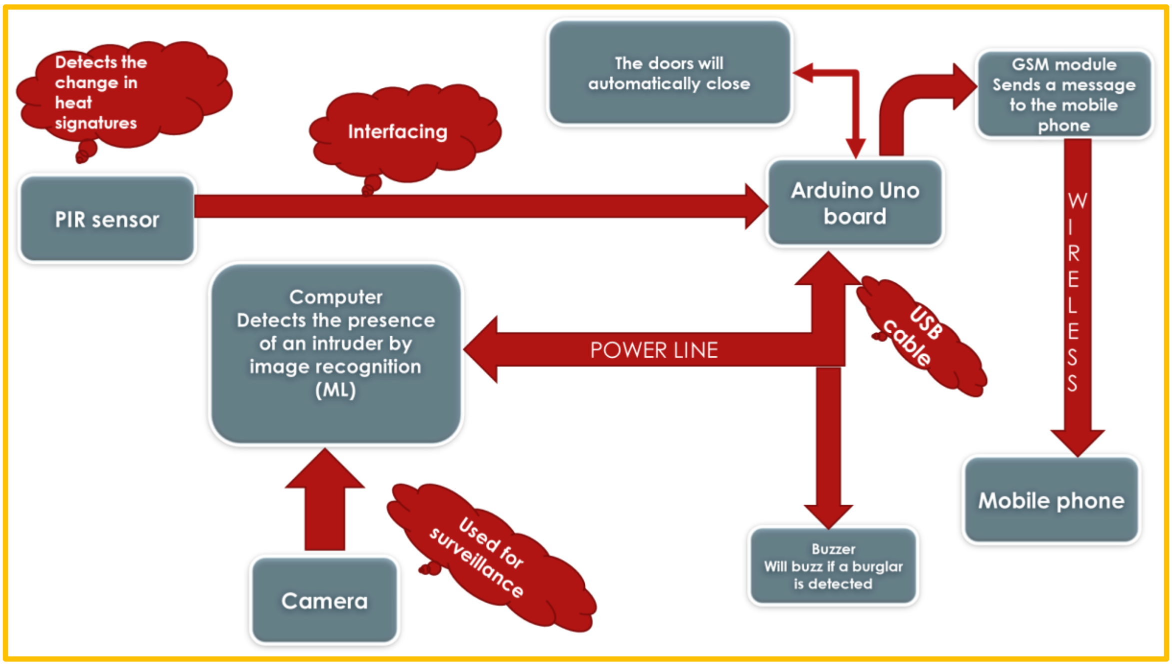


Figure 1.1: Overall block diagram

1.1 Problem Definition

1. Unauthorised intrusion into homes and attempt to burglary is one of the main issues faced by many home owners.
2. The home owner needs to be alerted and provide a mechanism to prevent the intrusion whereas, the existing system will allow the home owner to take necessary action post the occurrence of a burglary.
3. Availability of instant information in remote devices as the home owner will not be near their house all the time.
4. Avoidance of false triggering of burglar alarm.

Chapter 2 –

Literature Survey

1) ScienceDirect, Procedia Computer Science 135 (2018) 465–472, 3rd International Conference on Computer Science and Computational Intelligence 2018, “Design of Smart Home Security System using Object Recognition and PIR Sensor” by Nico Surantha and Wingky R. Wicaksono; Page numbers- 3 and 4.

This research aims to design and implement a home security system with the capability for human detection. The traditional home security system, i.e., closed circuit television (CCTV) can only capture and record the video without able to give warning feedback if there is any suspicious object. Therefore, an additional object detection and warning method is required if there is an intruder. The proposed design is implemented using Raspberry Pi 3 and Arduino, which are connected by USB cable. The PIR sensor is installed on the Arduino and the webcam is mounted on Raspberry Pi 3. The Raspberry Pi 3 is used to process inputs from received sensors and process images for human detection. The PIR sensor detects the movement around the sensor to activate the webcam to capture a picture. Then, the object recognition is performed using histogram of gradient (HoG) and support vector machine (SVM) to detect the suspicious object. If the suspicious object is detected, then alarm is activated to warn the house owner about the existence of intruder. The evaluation results show that it takes in average 2 seconds for proposed system to detect the intruder. It also shows that the system can successfully detects the intruder with accuracy of 89%.

2) “Human Detection Using Oriented Histograms of Flow and Appearance”, by Navneet Dalal, Bill Triggs, and Cordelia Schmid; GRAVIR-INRIA, 655 avenue de l’Europe, Montbonnot 38330, France; Page numbers- 3 to 12.

Detecting humans in films and videos is a challenging problem owing to the motion of the subjects, the camera and the background and to variations in pose, appearance, clothing, illumination and background clutter. We developed a detector for standing and moving people in videos with possibly moving cameras and backgrounds, testing several different motions coding schemes and showing empirically that orientated histograms of differential optical flow give the best overall performance. These motion-based descriptors are combined with our Histogram of Oriented Gradient appearance descriptors. The resulting detector is tested on several databases including a challenging test set taken from feature films and containing wide ranges of pose, motion and background variations, including moving cameras and backgrounds. We validate our results on two challenging test sets containing more than 4400 human examples. The combined detector reduces the false alarm rate by a factor of 10 relative to the best appearance-based detector, for example giving false alarm rates of 1 per 20,000 windows tested at 8% miss rate on our Test Set 1.

3) “Real time human detection in video streams”, by FATMA SAYADI*, YAHIA SAID, MOHAMED ATRI AND RACHED TOURKI Electronics and Microelectronics Laboratory Faculty of Sciences Monastir, 5000 Tunisia, in Recent Researches in Automatic Control, Systems Science and Communications; Page numbers- 1 to 4.

Detecting humans in films and videos is a challenging problem owing to the motion of the subjects, the camera and the background and to variations in pose, appearance, clothing, illumination and background clutter. We develop a detector for standing and moving people in videos, testing several different motions coding schemes and showing empirically that orientated histograms give the best overall performance. The resulting detector is tested on several databases including a challenging test set taken from feature films and containing wide ranges of pose, motion and background variations, including moving cameras and backgrounds. Experimental results from real time video are provided. These results are validated on a challenging test set containing 1000 human examples.

4) IJCSMC, Vol. 2, Issue. 5, May 2013, pg.417 – 424, RESEARCH ARTICLE, © 2013, IJCSMC All Rights Reserved 417, “Arduino Based Wireless Intrusion Detection Using IR Sensor and GSM”, by Prakash Kumar1, Pradeep Kumar2; Page numbers- 2 to 4.

Intrusion detection systems (IDS) strive to catch computer system intrusion & utilize by any garnering and analyzing data. Wireless IDSs garner all local wireless transmissions and generate alerts based either on predefined signatures or on anomalies in the traffic. These wireless IDS can monitor and analyze user and system activities of known attacks, identity abnormal network activity and detect policy violations. Intrusion detection systems (IDSs) should be designed to facilitate the detection of attempted and actual unauthorized entry into designated areas and should complement the security response by providing the security force with prompt notification of the detected activity from which an assessment can be made and a response initiated. We intended to avoid the access and keep track of the intruder's attempts and intensions. A clear and emerging new channel in the space of banking and payments is mobile. A key challenge with gaming user adopting of mobile banking and payment is the customer's lack of confidence in security of the

services. The economic growth in wireless network faults, vulnerabilities and attacks make the wireless local area network (WLAN) security management a challenging research area. Deficiencies of security methods like cryptography (WEP) and firewalls, causes the uses of more complex security systems.

5) International Research Journal of Engineering and Technology (IRJET), "Smart Home Automation and Security System using Arduino and IOT", by Siddharth Wadhwani1, Uday Singh2, Prakarsh Singh3, Shraddha Dwivedi4; Page numbers- 1 to 3

With the advancement of technology and more dependency of people on smart phone and increasing demands of easy and quick way of solving Daily life task, it has become very important to have a technology which can control over the domestic and industrial applications using IOT. Our paper 'Sensing and controlling the world around using Arduino and IOT 'deals with embedded technologies along with internet of things (IOT) using Arduino which employs the embedded block and script programming for Arduino and sensors like flex sensor, accelerometer, flame sensor, magnetic sensor, WI-FI module. In this paper we present a home automation and home security technique. The sensors will be interface with Arduino. The status of our home appliances will get uploaded to a cloud platform through wireless module. Our system and mobile should be connected over same wireless network. Our sensors will be able to enable or disable the sensors which will be in control of the user. The flex sensor will depend upon the gestures of our fingers to control the appliances. The magnetic sensor will enhance door breaking security. All these data can be seen by user on the cloud platform like THINKSPEAK. This paper will serve as an example of how IOT applications can make our life easier.

6) Appears in the 17th IAPR International Conference on Pattern Recognition, vol 2, pp 188–191, Cambridge, UK 2004, Research gate, "Detecting Human Motion with Support Vector Machines", by- Hedvig Sidenbladh; Page numbers- 2 and 3.

This paper presents a method for detection of humans in video sequences. The intended application of the method is outdoor surveillance. In such an uncontrolled environment, the appearance of humans varies hugely due to clothing, identity, weather and amount and direction of light. The idea is therefore to detect patterns of human motion, which to a large extent is independent of the differences in appearance. To this end, a Support Vector Machine is trained with dense optical flow patterns originating from humans. The subjects are moving in different angles to the camera plane, on different image scales. This trained SVM is the core of a human detection algorithm which searches optical flow images for human-like motion patterns.

Chapter 3 -

Design and Implementation

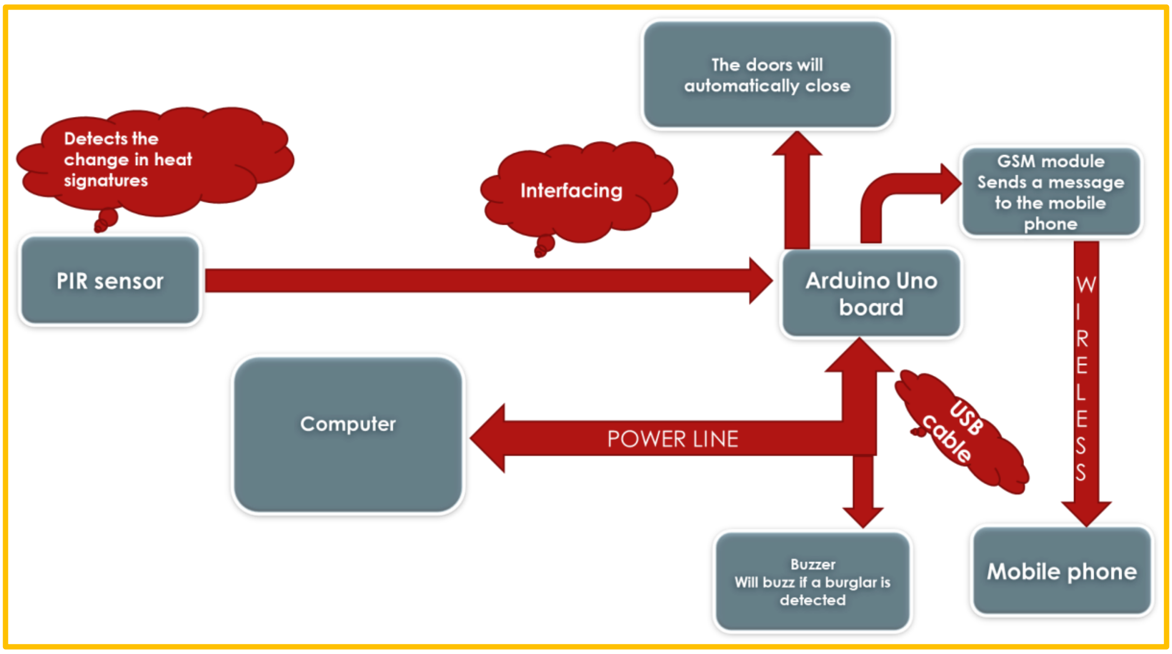


Figure 3.1: Block diagram of Burglar Alarm

3.1 Burglar Detection using PIR Sensor and Arduino Uno

The PIR sensor assembly is kept near the front entrance of the house for surveillance. The working principle of PIR sensor is:

A PIR sensor can detect changes in the amount of infrared radiation impinging upon it, which varies depending on the temperature and surface characteristics of the objects in front of the sensor. When an object, such as a person, passes in front of the background, such as a wall, the temperature at that point in the sensor's field of view will rise from room temperature to body temperature, and then back again. The sensor converts the resulting change in the incoming infrared radiation into a change in the output voltage, and this triggers the detection. Objects of similar temperature but different surface characteristics may also have a different infrared emission pattern, and thus moving them with respect to the background may trigger the detector as well.

Pairs of sensor elements may be wired as opposite inputs to a differential amplifier. In such a configuration, the PIR measurements cancel each other so that the average temperature of the field of view is removed from the electrical signal; an increase of IR energy across the entire sensor is self-cancelling and will not trigger the device. This allows the device to resist false indications of change in the event of being exposed to brief flashes of light or field-wide illumination. (Continuous high energy exposure may still be able to saturate the sensor materials and render the sensor unable to register further information.) At the same time, this differential arrangement minimizes common-mode interference, allowing the device to resist triggering due to nearby electric fields. However, a differential pair of sensors cannot measure temperature in this configuration, and therefore is only useful for motion detection. When the PIR sensor output is HIGH then the buzzer which connected to the Arduino Uno will start buzzing.

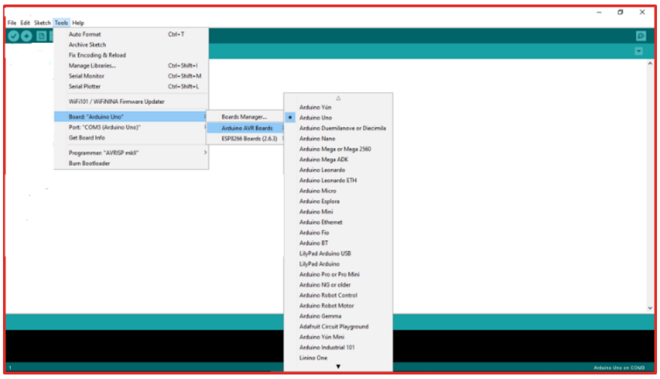


Figure 3.2: Configuration of Arduino Uno with the computer

The Figure 3.2 is elaborated as follows -

- The Arduino Uno board is connected to the computer via an USB cable and Arduino IDE is opened.
- In IDE, select the Tools menu and configure the COM3 serial port.

Table No.1- Specifications of PIR sensor

Voltage	4.8 V – 20 V
Current (idle)	<50 µA
Logic output	3.3 V / 0 V
Delay time	0.3 s – 200 s, custom up to 10 min
Lock time	2.5 s (default)
Trigger repeat	L = disable, H = enable
Sensing range	<120 °, within 7 m
Temperature	– 15 ~ +70 °C
Dimension	32 x 24 mm screw-screw 28 mm, M2 Lens diameter: 23 mm

From the above table it can be inferred that the PIR sensor is compact and can easily be mounted on almost any surfaces. PIR sensor also has a good range and so burglar detection will be effectively done.

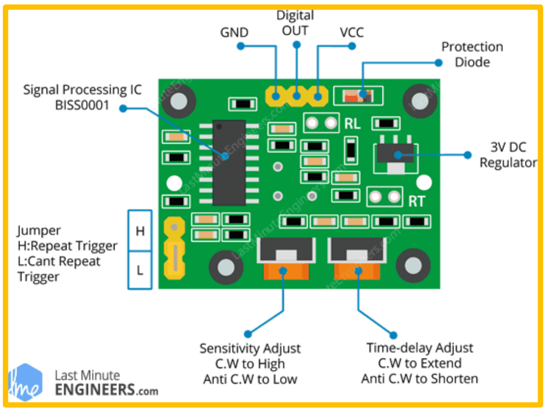


Figure 3.3: Parts of a PIR sensor

After the detection of burglar by the PIR sensor, the GSM module sends a text message “Intruder alert!” to the mobile phone of the home owner. It uses the following principle of operation:

SendMessage() – is the function created in the Arduino sketch to send an SMS. To send an SMS, the GSM module is set to Text mode first. This is achieved by sending an AT Command “AT+CMGF=1”. This command is set by writing this to SoftwareSerial port. To achieve this, mySerial.println() function is used. mySerial.println writes data to software serial port (the Tx pin of our Software Serial – that is pin 10) and this will be captured by GSM module (through its Rx pin). Further, the mobile number to which the SMS is to be sent has to be configured in the GSM module. This is achieved with AT command “AT+CMGS=\”+91xxxxxxxxxx\”\r” – where all x is replaced with the mobile number.

In next step, the actual content of SMS needs to be sent. The end of SMS content is identified with CTRL+Z symbol. The ASCII value of this CTRL+Z is 26. So, a char(26) is sent to the GSM module using the line mySerial.println((char)26); Each and every AT command may be followed by 1 second delay. A delay is given to the GSM module to respond properly. Once all of the above commands are sent to GSM module, a SMS in the set mobile number is sent.

When the home owner receives a text message on their phone, they can watch the real time security camera either by using an application like Microsoft team viewer or by accessing the CCTV footage using a mobile phone. As a result, the home owner will not be falsely alerted and can choose to ignore the message if the alarm was tripped by their pet or one of their family members as an accident.

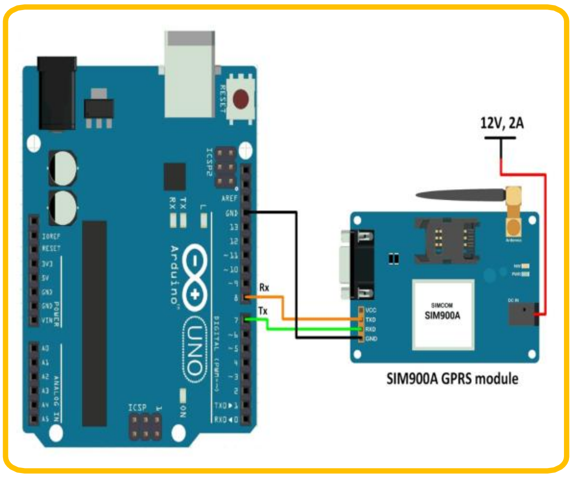


Figure 3.4: Circuit connections of GSM module with the Arduino Uno board

Interfacing of GSM module to Arduino UNO: -

- Tx pin of the GSM module is connected to the Rx pin of the Arduino Uno.
- Rx pin of the GSM module is connected to the Tx pin of the Arduino Uno.
- GND pin of the GSM module is connected to the ground pin of the Arduino Uno.

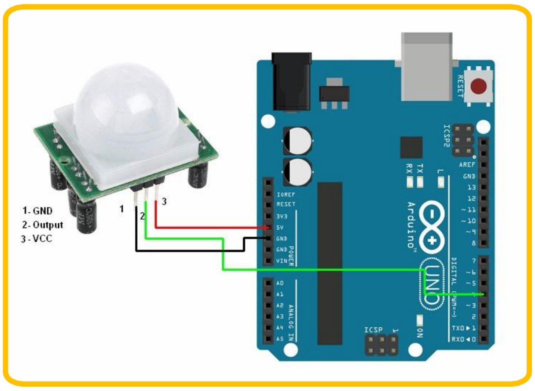


Figure 3.5: Circuit connections of PIR sensor with the Arduino Uno board

Interfacing of PIR sensor to Arduino UNO: -

- Vcc pin of the PIR is connected to the 5V pin of the Arduino Uno.
- GND pin of the PIR is connected to the ground pin of the Arduino Uno.
- Output pin of the PIR is connected to any one of the digital pins of the Arduino Uno.

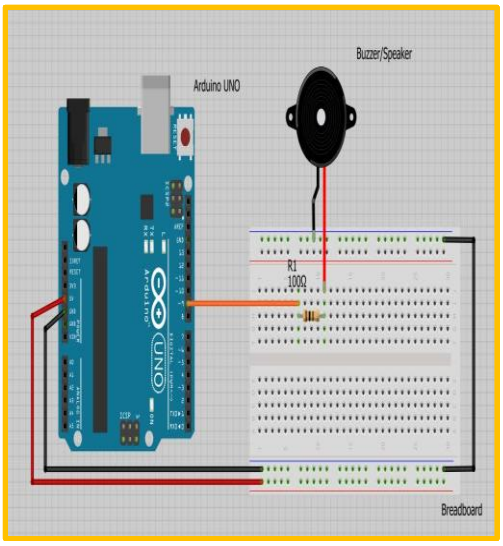


Figure 3.6: Circuit connections of 5V buzzer with the Arduino Uno board

Interfacing of buzzer to Arduino UNO: -

- Ground of the buzzer is connected to the GND of Arduino Uno.
- Vcc pin of the buzzer to 5V pin of the Arduino Uno.
- The longer lead of the buzzer is connected to the one of the analog pins of the Arduino uno via a 100-ohm resistor.

3.2 Arduino Uno Program Code

```
#include <SoftwareSerial.h>

int led_Pin = 8; // choose the pin for the LED
int PIR_sensor = 2;              // choose the input pin (for PIR sensor)
int buzzer_pin = 11;                // buzzer pin at 11
int val =0;
SoftwareSerial mySerial(10, 9);
/*
```

We declare and initialize the variables to be used.

Software serial: Create software serial object to communicate with SIM900 with the mentioned pin connections.

PIR_sensor: is an integer that contains the pin number of the Arduino to which the PIR sensor is connected.

buzzer_pin: is an integer that contains the pin number of the Arduino to which the buzzer is connected.

led_Pin: is an integer that contains the pin number of the Arduino to which the LED is connected

```
*/
        void setup()
        {
          mySerial.begin(9600);   // Setting the baud rate of GSM Module
          Serial.begin(9600);    // Setting the baud rate of Serial Monitor (Arduino)
          pinMode(led_Pin, OUTPUT);      // declare LED as output
          pinMode(PIR_sensor, INPUT);     // declare pir sensor as input
          pinMode(buzzer_pin, OUTPUT);        // declare buzzer as output
}
/*
```

Serial.begin(9600): Passes the value 9600 to the speed parameter. This tells the Arduino to get ready to exchange messages with the Serial Monitor at a data rate of 9600 bits per second. That's 9600 binary ones or zeros per second and is commonly called a baud rate.
mySerial.begin(9600): Begins the serial communication between the Arduino UNO and the SIM900 with the baud rate of 9600.
pinMode(PIR_sensor, INPUT): Sets the PIR sensor to input mode.
pinMode (buzzer_pin, OUTPUT): Sets the buzzer to output mode.

```
*/
        void loop()
        {
           val = digitalRead(PIR_sensor);  // read input value of the pir sensor
           if (val == HIGH)  // check if the input is HIGH
           {
              digitalWrite(led_Pin, HIGH);  // turn LED ON
              digitalWrite(buzzer_pin, HIGH); //ACTIVATE BUZZER
              Serial.println("Motion detected!\r\n\n");
              Serial.println("Alerting Admin/Owner \r\n\n");
              mySerial.println("AT+CMGF=1");    //To send SMS in Text Mode
              delay(300);
              mySerial.println("AT+CMGS=\"+919699449474\"\r"); // change to the phone
        number you using
              delay(300);
              mySerial.println("Someone is Trying to Break IN \r\n\n");//the content of the
        message
              delay(300);
```

```
        mySerial.println("Successfully Sent Message from GSM Module \r\n");//the
stopping character
        delay(500);
      }

    if (val == LOW)  // check if the input is LOW
    {
      digitalWrite(led_Pin, LOW); // turn LED OFF
      digitalWrite(buzzer_pin, LOW);  // turn off buzzer
      delay(1000);
      }

 }
/*
```

Val detects the output given by the PIR sensor (HIGH or LOW) and sends the signal to Arduino Uno.

If the PIR sensor detects a person it turns high and the Arduino receives information that a person is detected.

The buzzer starts buzzing.

An SMS is sent to the given number.

If the PIR sensor is Low it means nothing is detected and the buzzer is off.

mySerial.print("AT+CMGF=1\r"): This sends an SMS in text format which was stored in textForSMS to the given number.

mySerial.println("AT + CMGS = \"+923339218213\""): Contains the recipient's mobile number in international format.

If someone is trying to break into the house then a text message is sent to the home owner's phone.

*/

Connect D9 (Tx) and D10 (Rx) pin on shield to digital pin #9 and #10 on Arduino. As we'll be using software serial to talk to the shield, make sure the jumper cap is placed on the software serial port select. Power the shield using external power supply rated 5V, 2A. Also, make sure you select the external power source with the slide switch next to the DC jack. Connect all the ground in the circuit. Connect the antenna and insert fully activated SIM card in the socket.

3.3 Automatic locking of door after the burglar is detected

High-tech security technology allows users to open or close locks automatically, remotely and keyless. Home owners can program automatic door locks lock and unlock at pre-set specific times and no human intervention is necessary. The logic behind this is same as the logic behind the automatic garage door locking mechanism.

A NMOS is interfaced with the Arduino Uno and is connected to the front door. When the alarm system is turned ON and when the output of the passive infrared sensor is HIGH then the current is flow through the NMOS thus establishing contact with the door and will automatically lock the door. A diode is used to stop the reverse current and to prevent short circuiting of the system. The gate of the NMOS is connected to pin 2 of Arduino Uno, source to ground and drain to the DC motor. For the above application a relay can also be used. In the circuit diagram below, LED represents the automatic door locking mechanism.

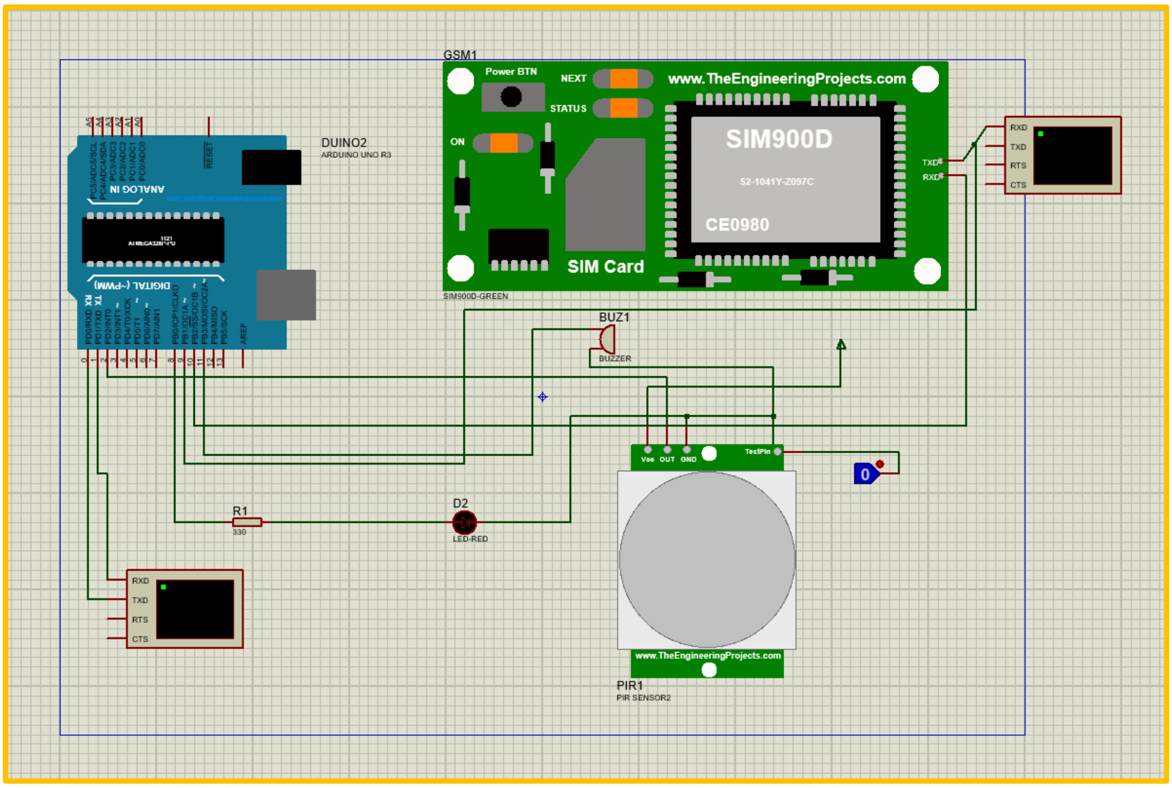


Figure 3.7: Circuit diagram of NMOS and DC motor for automatic door lock system

3.4 Human detection using machine learning

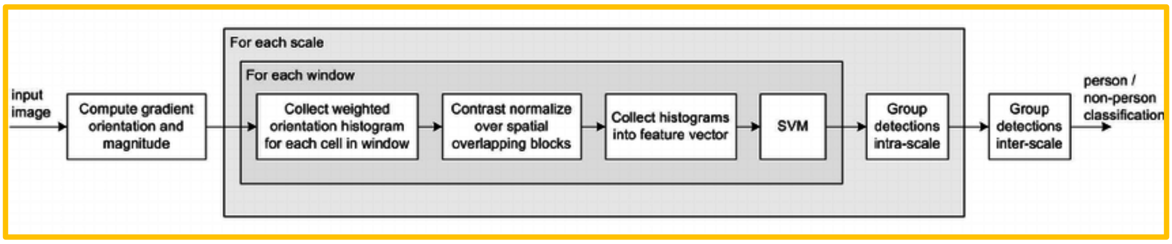


Figure 3.8: Block diagram of human detection

The surveillance cameras are placed near the front door of the house and an angle which covers most of the area in order to spot the burglar. The blind spots of the camera, if any, will be covered by the PIR sensor and vice versa.

An image gradient is a directional change in the intensity or colour in an image or the discontinuity in brightness in an image is called as the gradient of the image. In two dimensions (2D), in the rectangular cartesian (x, y) plane, a function of x and y is a surface giving the altitude at every point. The gradient is the generalization of the derivative: at a given (x, y) point, the gradient is oriented towards the direction of maximum slope, and its magnitude is the slope of the plane tangent to the surface at this point.

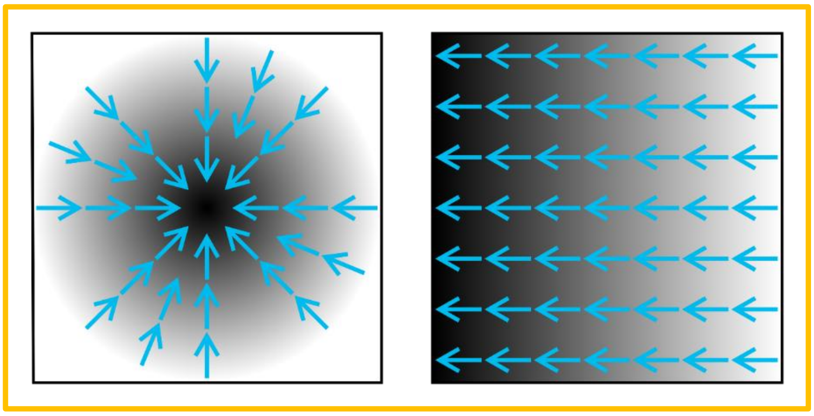


Figure 3.9: Gradient of two 2D functions

In this particular case, more black the image is the larger the function. In the centre of the left image, the gradient is not displayed, but it would be equal to 0. In a black and white picture, the greyscale level is analogous to the altitude, and the gradient is a measure of how fast the level changes and of the direction of the change. An edge in the picture (a black to white transition) leads to a large gradient perpendicular to the edge, from white to black. In a colour picture, one can compute a gradient for each colour level, and e.g., take the maximum gradient over the three colour levels.

A histogram is a data structure that is used to compress data and to represent its probability distribution. A histogram can have many dimensions but in practice, one dimensional (1D) and two dimensional (2D) histograms are most often used. In HOG, the gradients are stored in a 1D histogram.

In human detection using histogram of oriented gradients the following steps are used:

- The picture is scanned with a detection window of varying size.
- For each position and size of the detection window, the window is subdivided in cells. The cells are in practice relatively small: they typically contain only a small part of the person to be detected, maybe the side of an arm, or the top of the head.

- In each cell, a gradient is computed for each pixel, and the gradients are used to fill a histogram: the value is the angle of the gradient, and the weight is the magnitude of the gradient.
- The histograms of all cells are put together and fed to a machine learning discriminator to decide whether the cells of the current detection window correspond to a person or not.

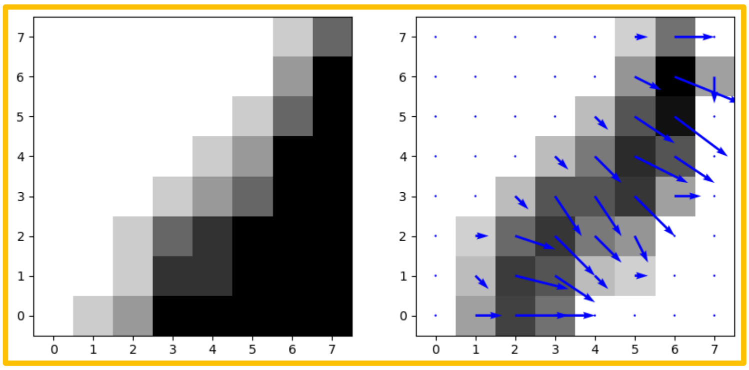


Figure 3.10: Left- the cell image data and right- the corresponding gradient

In the above image, the gradient is represented by an arrow and its magnitude is also shown as a greyscale colour map.

Support vector machine in addition to the histogram of oriented gradients is used to separate the human from other objects present in the image. Support vector machine is a supervised machine learning algorithm (The machine learns using labelled data. The model is trained on an existing data set before it starts making decisions with the new data) which is trained using datasets to identify human beings in an image amidst several other objects. First all the objects present in the image are masked using opencv2 then the support vector machine is trained to identify human beings in the image using labelled dataset.

- The machine learning python code was programmed in python 3 language on Jupyter notebook (anaconda navigator) compiler.
- OpenCV2 or Open-Source Computer Vision Library is a library designed for digital image processing.
- NumPy is a python library used for working with arrays and mathematical functions.
- Cv2.HOGDescriptor() function creates the HOG descriptor and detector with default parameters equal to HOGDescriptor(Size(64, 128), Size(16, 16), Size(8, 8), Size(8, 8), 9) and stores it in the variable ‘hog’.
- hog.setSVMDetector(cv2.HOGDescriptor_getDefaultPeopleDetector()) function calls the pre-trained support vector machine algorithm for human detection.

- cv2.startWindowThread() function sets up the window for the output video to be displayed on.
- cv2.VideoCapture(0) function starts the webcam of the computer system and stores it in the variable ‘capture’. The parameter 0 is passed inside the function to access the webcam.
- In order to save a video in the PWD (present working directory) of the compiler output, cv2.VideoWriter() function is used.
    - **Syntax**: cv2.VideoWriter('output_filename.extension', codec(*'filetype'), video’s FPS, frame size).
    - **Parameters: output-** The value of the --output is the path where the video file is stored in the computer. ‘.avi’ (Audio Video Interleave) files contain both audio and video data in a file container that allows synchronous audio-with-video playback.
    - **Codec-** cv2.VideoWriter_fourcc codec (coder and decoder) is used for recording the video where, fourCC is a 4-byte code used to specify the video codec. The codec is basically used to extract information from the video by converting the video into frames. ‘MJPG’ is the video extension which is used to detect motion in the video.
    - **FPS-** The rate at which the video is recorded.
    - **Frame size-** This parameter is the height and width of the video frame.
- Inside the infinite loop (while(True)) ret and frame are two variables in which the video is stored. The capture.read() function is used to read the data of the video onto the compiler.
- cv2.resize(frame, (640, 480)) function is used to resize the video frame in case the frame’s width and height is not (640, 480).
- cv2.cvtColor(frame, cv2.COLOR_RGB2GRAY) function is used to convert the frame from RGB (red-green-blue) to gray scale. The frame is converted to gray scale because in this application (human detection) colour is irrelevant and is considered as noise and therefore, edge detection is faster also the detection of humans. ‘Gray’ is a variable which stores the gray scaled image.
- hog.detectMultiScale(frame, winStride = (8, 8)) function detects objects of different sizes in the input image. The detected objects are returned as a list of rectangles. The list of rectangles is stored in ‘boxes’ and ‘weights’ variables.
    - **Syntax:** hog.detectMultiScale(img, winStride = (x, y))

- **Parameters: img-** Matrix of the type CV_8U or CV_8UC3 containing an image where objects are detected.
- **Winstride()-** It is also called Window stride. It must be a multiple of block stride. The winStride parameter is a 2-tuple that dictates the "step size" in both the x and y location of the sliding window. In the context of object detection, a sliding window is a rectangular region of fixed width and height that "slides" across an image. The below image illustrates the working of the winstride function.

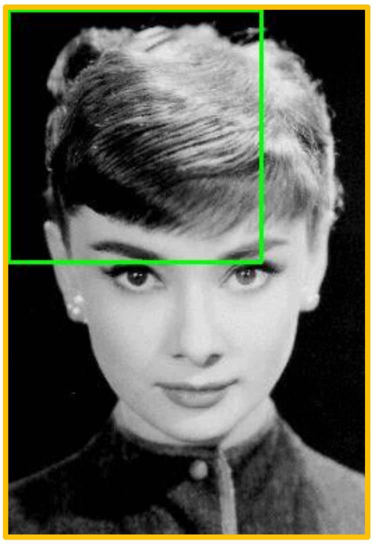


Figure 3.11: An example of applying a sliding window to an image for human detection

$$g = \sqrt{g_x^2 + g_y^2}$$

$$\theta = \arctan \frac{g_y}{g_x} \quad \text{..........} \quad \textbf{(3.1)}$$

Where g is the gradient of the image, g_x is the gradient in X-direction, g_y is the gradient in Y-direction and **θ** is the angle of the image.

For example, an 8×8 image patch contains 8x8x3 = 192-pixel values. The gradient of this patch contains 2 values (magnitude and direction) per pixel which adds up to 8x8x2 = 128 numbers. So, in a 640 X 480 image contains 9,21,600 pixels. Therefore, 6,14,400 equal HOG cells are formed in the image.

- At each stop of the sliding window, HOG features are extracted and are passed to Linear SVM for classification. The process of feature extraction and classifier decision is time consuming, so it is preferred to evaluate as little windows as possible.

- boxes = np.array([[x, y, x + w, y + h] for (x, y, w, h) in boxes]), in this command the features of humans (two eyes, hands, legs, ears, one mouth and nose) that were thus extracted are stored inside the array boxes.

- Inside the for loop (xA, yA, xB, yB) are the width and height of the human respectively. cv2.rectangle(frame, (xA, yA), (xB, yB), (0, 255, 0), 2) is used to draw the rectangular frame around the detected human.
    - **Syntax:** cv2.rectangle(image, start_point, end_point, colour, thickness)
    - **Parameters: image-** It is the image on which the rectangular frame is to be drawn.
    - **Start_point-** It is the starting coordinates of rectangle. The coordinates are represented as tuples of two values i.e. (X coordinate value, Y coordinate value).
    - **End_point-** It is the ending coordinates of rectangle. The coordinates are represented as tuples of two values i.e. (X coordinate value, Y coordinate value).
    - **colour**- It is the colour of border line of rectangle to be drawn. For BGR, we pass a tuple. e.g.: (0, 255, 0) for green colour.
    - **Return Value:** It returns an image.
- output.write(frame.astype('uint8')) function coverts the output image frame into an eight-bit integer value.
- cv2.imshow('frame', frame) function displays the output video frame window.
- if cv2.waitKey(1) & 0xFF == ord('c'), this if condition is used to close the output window by clicking on the ‘c’ key. The ord() method in Python converts a character into its Unicode code value. This method accepts a single character.
- capture.release() and output.release() functions are used to release the output video to be stored in the present working directory of the python compiler.
- cv2.destroyAllWindows() and cv2.waitKey(1) functions close the output window.

3.5 Human Detection Program Code

# Import the required modules.
import numpy as np # np is an arbitrary variable which is used as an abbreviation instead of typing numpy every time.

```
import cv2 # Open cv2 module.

hog = cv2.HOGDescriptor() # Initialize the Histogram Oriented Gradient descriptor for human detection.
hog.setSVMDetector(cv2.HOGDescriptor_getDefaultPeopleDetector()) # Support Vector Machine algorithm is used for supervised detection of human.

cv2.startWindowThread() # This function will access the video displayed on the respective window.

capture = cv2.VideoCapture(0) # Opens the webcam to start the video stream.

# The output will be written in the file named, "output.avi".
output = cv2.VideoWriter('output.avi', cv2.VideoWriter_fourcc(*'MJPG'), 15., (640, 480))
# This function is used to make annontations on a video.

while(True) : # Infinite loop.
    # Captures the video frame-by-frame.
    ret, frame = capture.read()

    # Resizing the frame for faster & more accurate detection.
    frame = cv2.resize(frame, (640, 480))
    # Converting the RGB colours to grayscale for faster detection.
    gray = cv2.cvtColor(frame, cv2.COLOR_RGB2GRAY)

    # The below two lines will detect people present in the frame, also returns the corresponding values of boxes & weights for the detected humans.
    boxes, weights = hog.detectMultiScale(frame, winStride = (8, 8))

    boxes = np.array([[x, y, x + w, y + h] for (x, y, w, h) in boxes]) # The sliding window parameters are stored in an array.

    for (xA, yA, xB, yB) in boxes :
        cv2.rectangle(frame, (xA, yA), (xB, yB), (0, 69, 255), 2) # This function draws a rectangular box around the detected human.
```

```
    output.write(frame.astype('uint8')) # Writes the boxes & weights on the output video.
    cv2.imshow('frame', frame) # Displays the output frame.

    # The user can close the webcam by pressing the key 'c'.
    if cv2.waitKey(1) & 0xFF == ord('c') :
        break

# The output video is stored in the PWD of the jupyter notebook.
capture.release()
output.release()
# Closes the output window.
cv2.destroyAllWindows()
cv2.waitKey(1)
```

3.6 Switching the computer system remotely using a smart phone

The home owner can switch on/off the system using his/her mobile phone, in order to shut down a PC that is connected to the smartphone, then an application called "Team Viewer" on the smartphone and its server on the PC has to be installed. The home owner is required to connect their phone with their PC by entering a partner ID and password (displayed on the pc) in their phone. By doing this, commands like shut down, sleep or hibernate can be sent to the PC from the smartphone. This feature is crucial because the home owner, family or pets will not falsely trigger the alarm.

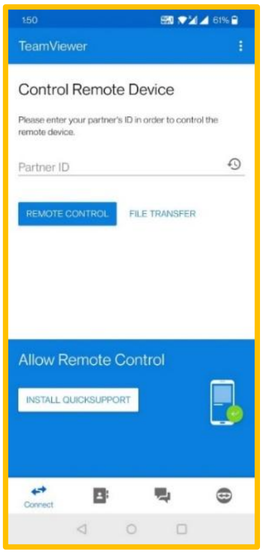


Figure 3.12: Image of the TeamViewer mobile phone application

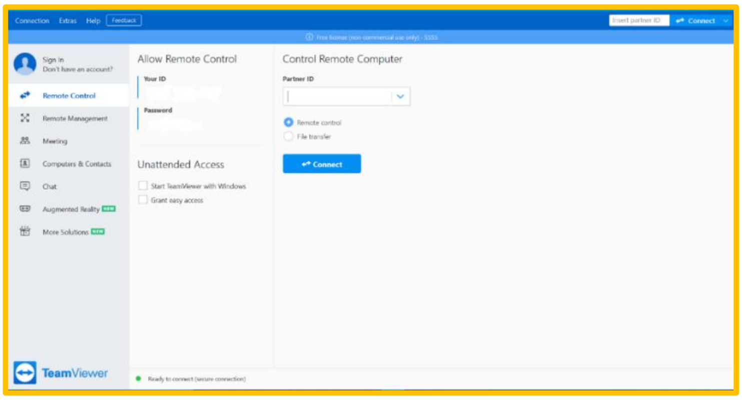


Figure 3.13: Image of the TeamViewer PC application

Table No.2- Cost analysis

Components	**Cost**
Arduino Uno with USB cable	₹425
PIR Sensor	₹150
Jumper cables	₹30
5V buzzer	₹35
GSM module	₹1190
Total:	₹1830

Chapter 4 -

Results & Conclusions

4.1 Results

I. The safety of the house is ensured with this prototype.
II. This can be considered as the first step towards home automation.
III. False detection of burglars is also avoided using machine learning effectively.
IV. This prototype is a very reliable cost-effective solution.
V. Burglar is detected successfully with high accuracy.
VI. By the usage of the PIR sensor and GSM module, the home owner is made aware of the intrusion and can prevent the burglary from happening.
VII. This mini project combines both ML & IoT domains.

The below images are the results of the ML model obtained while live testing. The webcam of the laptop was accessed by the python code and the results thus obtained were real time output.

Figure 4.1: Human detection

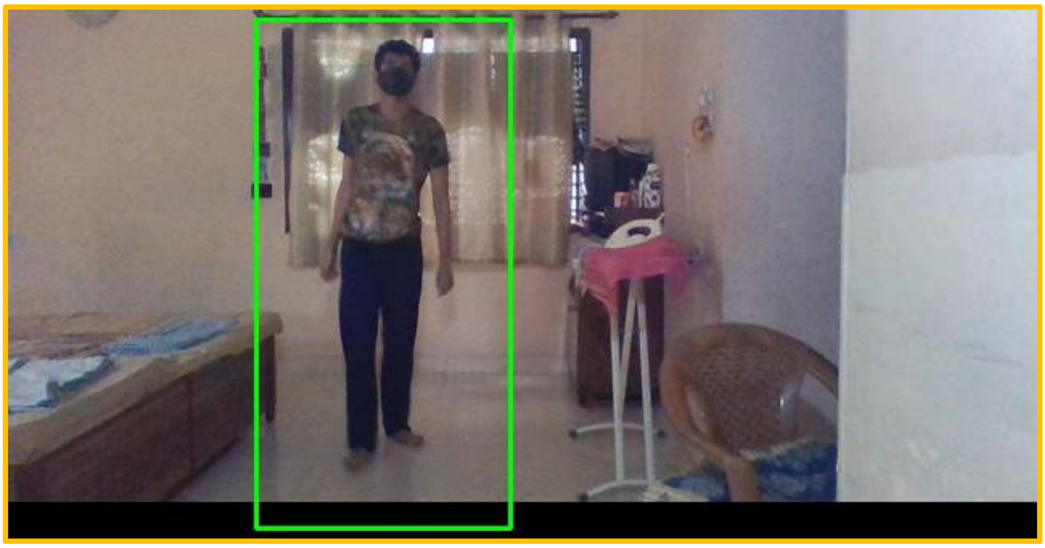


Figure 4.2: Human detection with face mask

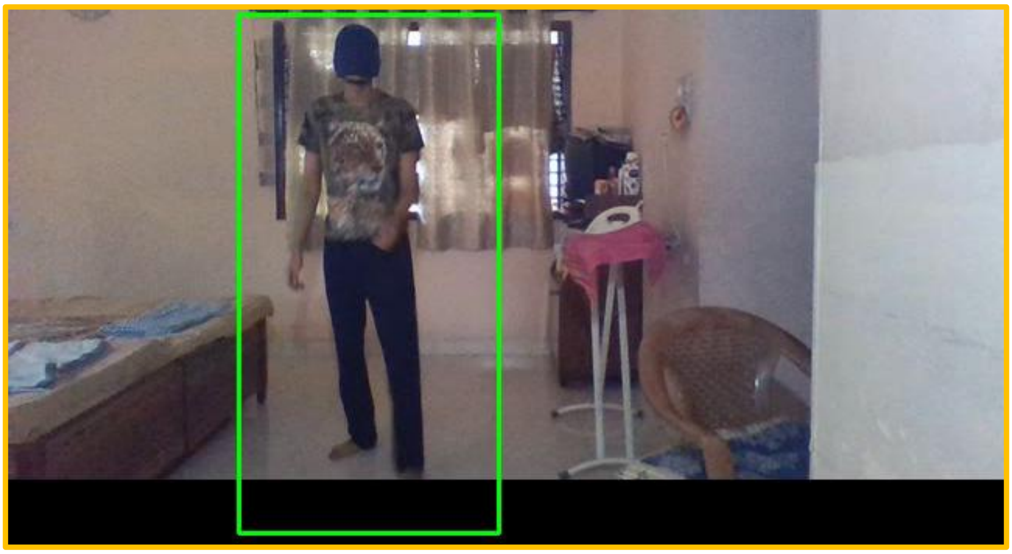


Figure 4.3: Human detection when the entire face is covered

Figure 4.4: Human detection while crouching

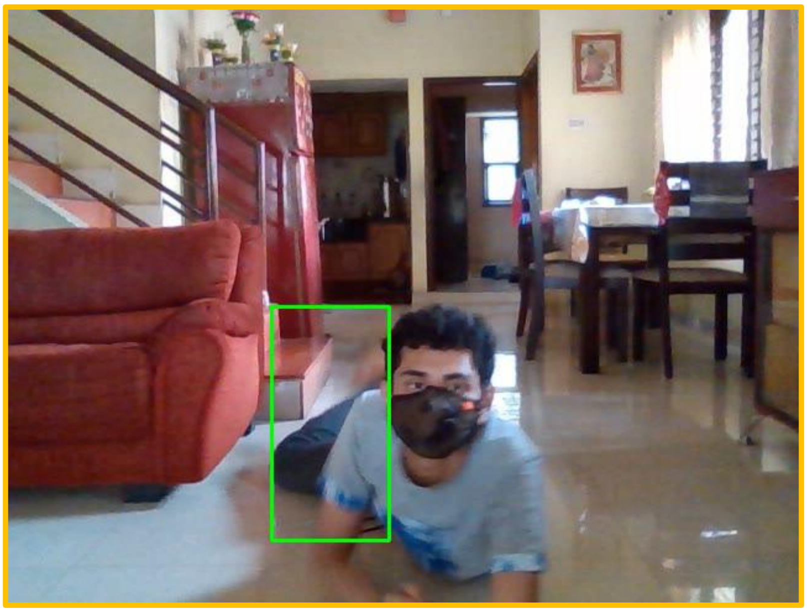


Figure 4.5: Human detection while laying down

Inference: - It is evident by the above images that the machine learning algorithm is not dependent on the visibility of the human's face as the algorithm is trained to detect humans and not facial recognition. The human was detected even when crouching and while laying down on the floor.

The below images are the output obtained by simulating the circuit diagram on Proteus simulation tool.

Case 1: - When the output of PIR sensor is low (no burglar). The buzzer doesn't buzz, GSM module will not send any message and the doors will remain in its position.

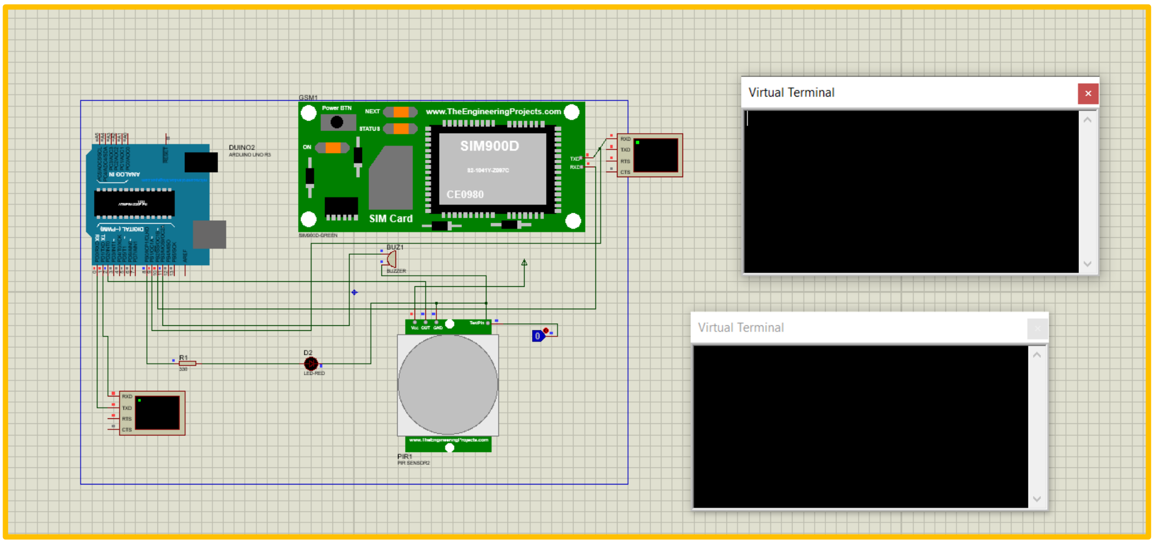


Figure 4.6: Simulation of the burglar alarm system in the absence of burglar

Case 2: - When the output of PIR sensor is HIGH (burglar detected). The buzzer starts to buzz, GSM module will send a text message the doors close automatically.

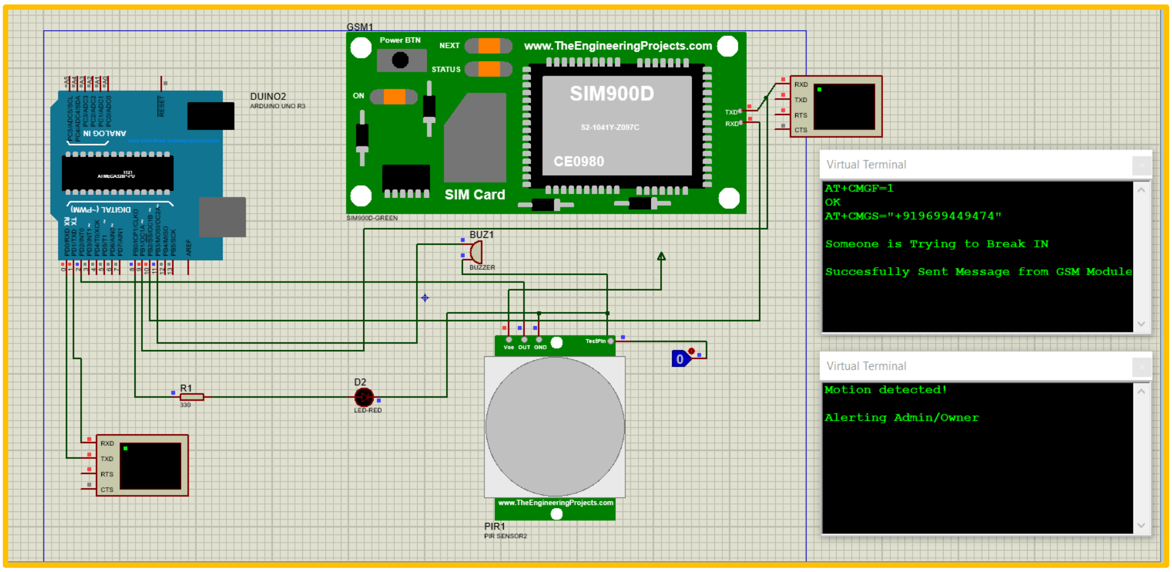


Figure 4.7: Simulation of the burglar alarm system in the presence of burglar

In the above two images, LED represents the automatic door locking mechanism.

Table No.3- Output Values

PIR sensor output	Buzzer output	LED output
HIGH	HIGH	HIGH
LOW	LOW	LOW

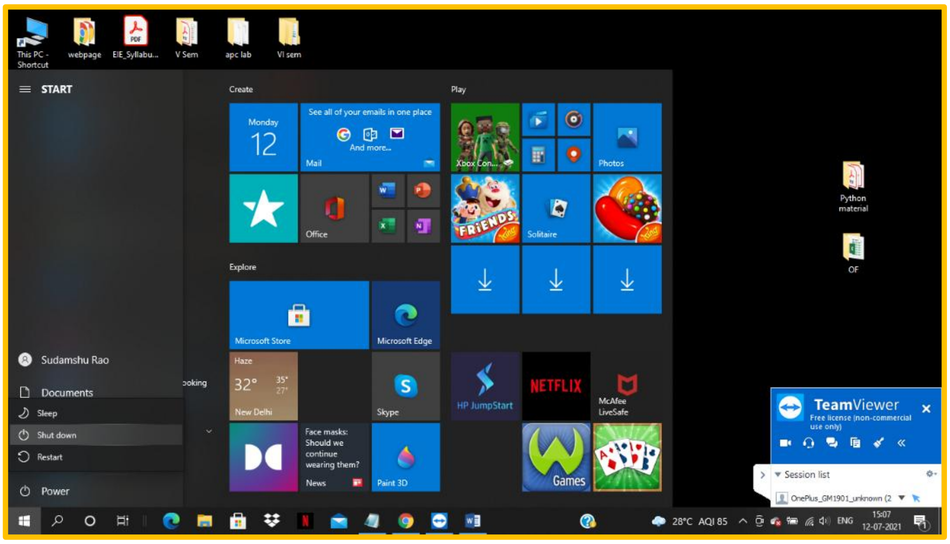


Figure 4.8: Sudamshu's PC view using TeamViewer

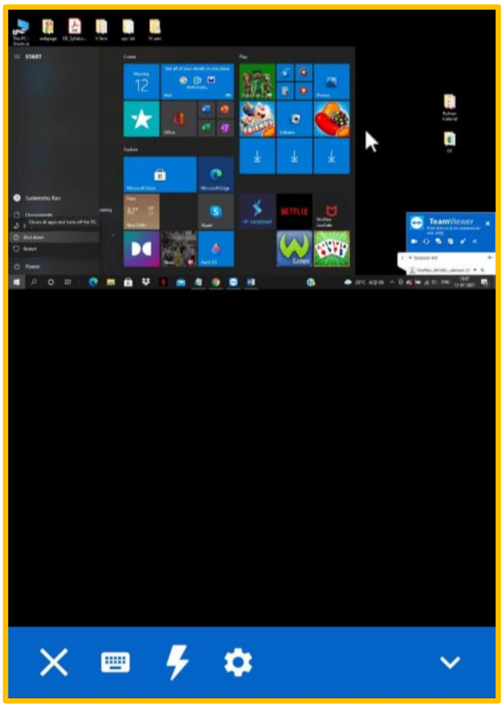


Figure 4.9: Sudamshu's phone view using TeamViewer

From the above two images it is evident that the PC can be shut down and power up using a smart phone via TeamViewer application.

4.2 Conclusions

- Scenario 1: The home owner before leaving the house can switch on the alarm using a mobile phone and before entering the house can switch it off so that the home owner is not detected as a burglar.
- Scenario 2: Suppose if a pet is detected by the PIR sensor then the ML algorithm will prevent false detection since the algorithm will detect for humans and the home owner can use TeamViewer app to see whether it was a human or not.
- Scenario 3: In case a particular case where the home owner is aware that a familiar person is entering or exiting his/her house, they can choose to ignore the intruder notifications or they can watch in real time (using team viewer app).
- Scenario 4: Suppose if the home owner is on a vacation where there is no network available then the control of this alarm can be given to a trusted person or their neighbour.
- Scenario 5: Suppose if the burglar is wearing a mask to cover the face, then the ML algorithm will still detect the burglar as it detects for humans and not faces.
- Scenario 6: Post man, milk man, delivery men, etc. will not be detected as burglars since this alarm is placed inside the house near the door. If they break in the house

(without any prior permission) then they will be considered as burglars and will be detected.

- ❖ Scenario 7: Suppose if a burglar, post man, delivery man together come near the door then the burglar will attempt burglary only after the post man, etc. leaves (as they will be witnesses to the burglary) and the alarm will detect the burglar and necessary action can be taken by the home owner.
- ❖ Scenario 8: If the burglar blasts the door open while intruding into the house and suppose the PIR sensor is damaged then the CCTV cameras and the ML algorithm will still be functional and hence the burglar will be detected. The neighbours will hear the explosion and alert the authorities.

4.3 Program Outcomes Addressed

A. PO1 Engineering knowledge: Apply the knowledge of mathematics, science, engineering fundamentals and an engineering specialization to the solution of complex engineering problems.

B. PO2 Problem analysis: Identify, formulate, research literature and analyse complex engineering problems reaching substantiated conclusions using first principles of mathematics, natural sciences and engineering sciences.

C. PO3 Design/development of solutions: Design solutions for complex engineering problems and design system components or processes that meet the specified needs with appropriate consideration for the public health and safety and the cultural, societal and environmental considerations.

D. PO4 Conduct investigations of complex problems: Use research-based knowledge and research methods including design of experiments, analysis and interpretation of data, and synthesis of the information to provide valid conclusions.

E. PO9 Individual and team work: Function effectively as an individual and as a member or leader in diverse teams and in multidisciplinary settings.

4.4 Further scope

- ✓ This project can also be interfaced with lights such that whenever the person enters a room the light can be switched on.
- ✓ This project can also be modified for banks to alert authorities before theft occurs.
- ✓ The machine learning model can be used in an automatic lift where the model will count the number of people entering the lift and when the threshold value is reached then the doors of the lift will automatically close.

- ✓ The machine learning model can be used in an automatic metro train where the model will count the number of people entering the train and when the threshold value is reached then the doors of the train will automatically close.
- ✓ This machine learning model can be used in shopping malls or shopping complexes where the number of customers can be counted daily.

References

1. ScienceDirect, Procedia Computer Science 135 (2018) 465–472, 3rd International Conference on Computer Science and Computational Intelligence 2018, "Design of Smart Home Security System using Object Recognition and PIR Sensor" by Nico Surantha and Wingky R. Wicaksono; Page numbers- 3 to 4.
2. "Human Detection Using Oriented Histograms of Flow and Appearance", by Navneet Dalal, Bill Triggs, and Cordelia Schmid; GRAVIR-INRIA, 655 avenue de l'Europe, Montbonnot 38330, France; Page numbers- 3 to 12.
3. "Real time human detection in video streams", by FATMA SAYADI*, YAHIA SAID, MOHAMED ATRI AND RACHED TOURKI Electronics and Microelectronics Laboratory Faculty of Sciences Monastir, 5000 Tunisia TUNISIA, in Recent Researches in Automatic Control, Systems Science and Communications; Page numbers- 1 to 4.
4. IJCSMC, Vol. 2, Issue. 5, May 2013, pg.417 – 424, RESEARCH ARTICLE, © 2013, IJCSMC All Rights Reserved 417, "Arduino Based Wireless Intrusion Detection Using IR Sensor and GSM", by Prakash Kumar1, Pradeep Kumar2; Page numbers- 2 to 4.
5. International Research Journal of Engineering and Technology (IRJET), "Smart Home Automation and Security System using Arduino and IOT", by Siddharth Wadhwani1, Uday Singh2, Prakarsh Singh3, Shraddha Dwivedi4; Page numbers- 1 to 3.
6. Appears in the 17th IAPR International Conference on Pattern Recognition, vol 2, pp 188–191, Cambridge, UK 2004, Research gate, "Detecting Human Motion with Support Vector Machines", by- Hedvig Sidenbladh; Page numbers- 2 and 3.

Appendix

Arduino Uno datasheet-

HK Shan Hai Group Limited
Room 620 , Yuhan building , songling road , Futian district , Shenzhen

Technical Specification

EAGLE files: arduino-duemilanove-uno-design.zip Schematic: arduino-uno-schematic.pdf

Summary

Microcontroller	ATmega328P-AU
Operating Voltage	5V
Input Voltage (recommended)	7-12V
Input Voltage (limits)	6-20V
Digital I/O Pins	14 (of which 6 provide PWM output)
Analog Input Pins	6
DC Current per I/O Pin	40 mA
DC Current for 3.3V Pin	50 mA
Flash Memory	32 KB of which 0.5 KB used by bootloader
SRAM	2 KB
EEPROM	1 KB
Clock Speed	16 MHz

the board

HK Shan Hai Group Limited
Room 620 , Yutian building , songling road , Futian district , Shenzhen

Power

The Uno can be powered via the USB connection or with an external power supply. The power source is selected automatically.

External (non-USB) power can come either from an AC-to-DC adapter (wall-wart) or battery. The adapter can be connected by plugging a 2.1mm center-positive plug into the board's power jack. Leads from a battery can be inserted in the Gnd and Vin pin headers of the POWER connector.

The board can operate on an external supply of 6 to 20 volts. If supplied with less than 7V, however, the 5V pin may supply less than five volts and the board may be unstable. If using more than 12V, the voltage regulator may overheat and damage the board. The recommended range is 7 to 12 volts.

The power pins are as follows:

- **VIN.** The input voltage to the Arduino board when it's using an external power source (as opposed to 5 volts from the USB connection or other regulated power source). You can supply voltage through this pin, or, if supplying voltage via the power jack, access it through this pin.
- **5V.** The regulated power supply used to power the microcontroller and other components on the board. This can come either from VIN via an on-board regulator, or be supplied by USB or another regulated 5V supply.
- **3V3.** A 3.3 volt supply generated by the on-board regulator. Maximum current draw is 50 mA.
- **GND.** Ground pins.

Memory

The Atmega328 has 32 KB of flash memory for storing code (of which 0,5 KB is used for the bootloader); It has also 2 KB of SRAM and 1 KB of EEPROM (which can be read and written with the EEPROM library).

Input and Output

Each of the 14 digital pins on the Uno can be used as an input or output, using pinMode(), digitalWrite(), and digitalRead() functions. They operate at 5 volts. Each pin can provide or receive a maximum of 40 mA and has an internal pull-up resistor (disconnected by default) of 20-50 kOhms. In addition, some pins have specialized functions:

- **Serial: 0 (RX) and 1 (TX).** Used to receive (RX) and transmit (TX) TTL serial data. TThese pins are connected to the corresponding pins of the ATmega8U2 USB-to-TTL Serial chip .
- **External Interrupts: 2 and 3.** These pins can be configured to trigger an interrupt on a low value, a rising or falling edge, or a change in value. See the attachInterrupt() function for details.
- **PWM: 3, 5, 6, 9, 10, and 11.** Provide 8-bit PWM output with the analogWrite() function.
- **SPI: 10 (SS), 11 (MOSI), 12 (MISO), 13 (SCK).** These pins support SPI communication, which, although provided by the underlying hardware, is not currently included in the Arduino language.

- **LED: 13.** There is a built-in LED connected to digital pin 13. When the pin is HIGH value, the LED is on, when the pin is LOW, it's off.

HK Shan Hai Group Limited
Room 620 , Yutian building , songling road , Futian district , Shenzhen

The Uno has 6 analog inputs, each of which provide 10 bits of resolution (i.e. 1024 different values). By default they measure from ground to 5 volts, though is it possible to change the upper end of their range using the AREF pin and the analogReference() function. Additionally, some pins have specialized functionality:

- **I^2C: 4 (SDA) and 5 (SCL).** Support I^2C (TWI) communication using the Wire library.

There are a couple of other pins on the board:

- **AREF.** Reference voltage for the analog inputs. Used with analogReference().
- **Reset.** Bring this line LOW to reset the microcontroller. Typically used to add a reset button to shields which block the one on the board.

See also the mapping between Arduino pins and Atmega328 ports.

PIR sensor datasheet-

HC-SR501 PIR MOTION DETECTOR

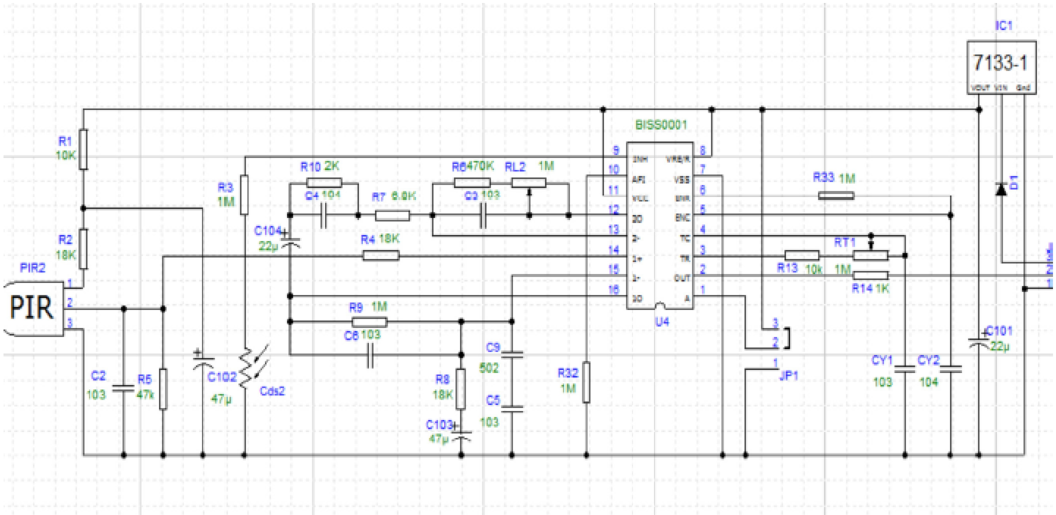
Product Discription

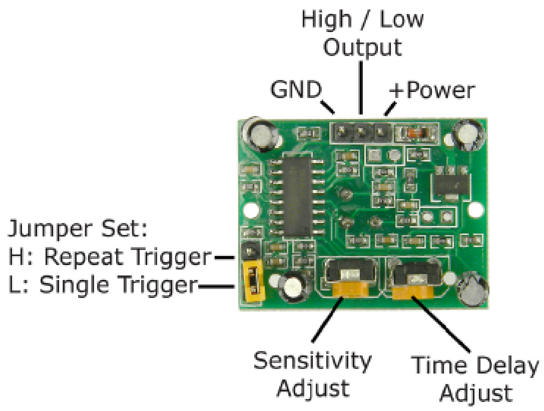
HC-SR501 is based on infrared technology, automatic control module, using Germany imported LHI778 probe design, high sensitivity, high reliability, ultra-low-voltage operating mode, widely used in various auto-sensing electrical equipment, especially for battery-powered automatic controlled products.

Specification:

- Voltage: 5V – 20V
- Power Consumption: 65mA
- TTL output: 3.3V, 0V
- Delay time: Adjustable (.3->5min)
- Lock time: 0.2 sec
- Trigger methods: L – disable repeat trigger, H enable repeat trigger
- Sensing range: less than 120 degree, within 7 meters
- Temperature: – 15 ~ +70
- Dimension: 32*24 mm, distance between screw 28mm, M2, Lens dimension in diameter: 23mm

HC-SR501 PIR MOTION DETECTOR





1 working voltage range :DC 4.5-20V

2 Quiescent Current :50uA

3 high output level 3.3 V / Low 0V

4. Trigger L trigger can not be repeated / H repeated trigger

5. circuit board dimensions :32 * 24 mm

6. maximum 110 ° angle sensor

7. 7 m maximum sensing distance

Product Type	HC--SR501 Body Sensor Module
Operating Voltage Range	5-20VDC
Quiescent Current	<50uA
Level output	High 3.3 V /Low 0V
Trigger	L can not be repeated trigger/H can be repeated trigger(Default repeated trigger)
Delay time	5-300S(adjustable) Range (approximately .3Sec -5Min)
Block time	2.5S(default)Can be made a range(0.xx to tens of seconds
Board Dimensions	32mm*24mm
Angle Sensor	<110 ° cone angle
Operation Temp.	-15-+70 degrees
Lens size sensor	Diameter:23mm(Default)

GSM module sim 900a datasheet-

DATASHEET GSM

Datasheet DW-SOHO-M-1GSM

Designed with DW-SOHO-M-1GSM for global market. DW-SOHO-M-1GSM compatible with DW-SOHO-H-2O, is a quad-band GSM engine that works on frequencies GSM 850MHz, EGSM 900 MHz, DCS 1800 MHz and PCS 1900MHz. The DW-SOHO-M-1GSM can search the 4 frequency bands automatically, and is compliant to GSM OpenBTS Phase 2/2+.

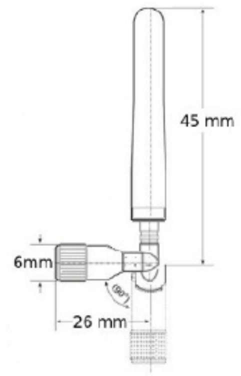
DW-SH-GSM FEATURES

Feature	Reference
Power supply	Single supply voltage 3.8V - 4.5V
External power supply required	NO
Power saving	Power consumption in SLEEP mode is 1.5mA
GSM class	SIM Card v3
Frequency Bands	850MHz, 900 MHz, 1800 MHz and 1900MHz.
Supported protocols	Voice (2G)
Temperature range	Normal operation: -25-+70 degrees Centigrade
External antenna	Connected via 50antenna connector
Audio features	Speech codec modes: Half Rate (ETS 06.20) Full Rate (ETS 06.10) Enhanced Full Rate (ETS 06.50 / 06.60 / 06.80) AMR Echo Cancellation Noise reduction
Certifications	CE FCC RoHS ISO 9001

www.denwaip.com

Denwa
UNIFIED COMMUNICATIONS

Antenna



Feature	Reference
Frecuenty	824-880/890-960/1850-1990/1710-2170/2400-2483
VSWR	2
Gain	3 dBi
Impedance ()	50
Polarization	Linear polarization
Connector	SMA plug
Length	5 mm
Color	Black
Temperature range	- 45° c a +75° c