1. INTRODUCTION
   1. Brief Introduction

The IoT (Internet of Things) is the network of physical devices, vehicles, home appliances, and other items embedded with electronics, software, sensors, actuators and connectivity which enables these things to connect, collect and exchange data. IoT involves extending internet connectivity beyond standard devices, such as desktops, laptops, smartphones and tablets, to any range of traditionally dumb or non-internet-enabled physical devices and everyday object. Embedded with technology, these devices can communicate and interact over the Internet, and they can be remotely monitored and controlled [1].

Technology has brought a dynamic and tremendous change in robotics and automation field which ranges in all kinds of areas. Surveillance is the process of close systematic observation or supervision maintained over a person, group, etc. especially one in custody or under suspicion. The act of surveillance can be performed both indoor as well as in outdoor areas by humans or with the help of embedded system such as robots and other automation devices. A robot is nothing but an automatic electronic machine that is capable of performing programmed activities thus replacing human work, providing highly accurate results and easily overcoming the limitations of human beings. Thus replacing humans in the surveillance fields is the one of the great advancement in robotics [2].

Video Surveillance Robot is actually an IoT based technology which makes the deployment of surveillance in even the remotely inaccessible area where it is almost impossible for human to reach. Even if the reach is possible some areas are way too risky to be camp by humans. In such places it comes really handy to have some machine which can do the surveillance work instead of human. This is where Wireless Video Surveillance Robot comes into use.

The Robot uses Wireless connection to transmit and receive data rather than any other form of rays (used for data communication by other robots).

The Wireless Video Surveillance Robot is built with the help of Raspberry pi and has the wireless range up to 92m.

The camera is mounted on the robot to get better visibility of the objects nearby robot. The programming language we used in this project is python which will be interfaced with Raspberry Pi board.

The robot is web controlled through an interface depending upon the module on which the robot is built. The data is send via the Internet connection. The circuitry designed for the construction of robot can determine the range to robot and the speed at which the data can be transmitted between the interface and the robot.

1. **PROBLEM DEFINITION**

Video Surveillance is the process of monitoring a situation, an area or a person. This generally occurs in a military scenario where surveillance of borderlines and enemy territory is essential to a country’s safety. Human surveillance is achieved by deploying personnel near sensitive areas in order to constantly monitor for changes. But humans do have their limitations, and deployment in inaccessible places is not always possible.

There are also the added risks of losing personnel in the event of getting caught by the enemy. With advances in technology over the years, it is possible to remotely monitor areas of importance by using robots in place of humans. The deployment of robot is both risks effective and cost effective.

1. **OBJECTIVES**

Our objectives of this project are given below:

* To monitor/spy in real time.

**Sub-Objectives**

* To detect all the irregular movements within the area and select the suspicious ones conduct in-depth observation and identification of the targets.
* To explore the emerging technology as research.
* To study about raspberry Pi and its features.

1. **LITERATURE REVIEW**

The development & innovation in technological field is moving at the rocket speed. Every new day rises with some new innovation and the rate at which the technology is developing and has spread its wings wide it is almost near to impossible for a single person to keep the track of all the development been done till now. With development of technology in modern world gaining pace it is about time that some importance should be given to the department of surveillance technology which is still working on age old technology to get its work done. Consider an example of traffic signal. Since the establishment of embedded technology the prototype for traffic signal has remained same even though the technology surrounding it has been revised couple of times. The technology surrounding traffic signals is mostly vehicles which depend on it for its smooth functioning.

The earlier day’s vehicles were just installed with simple machinery which is required for working of the four or two wheels which are installed on it and some lights surrounding the vehicles. But with the advancement of technology the changes are made and now we have reached to the point where we don’t even need to drive our own cars.

This proves that how far we have moved with technology and how long the development in surveillance field is overdue. There are various work and researches carried on with the surveillance technology but none has been brought into real world; that too in abundance to be a noticeable change. Similarly various robotic technologies have been proposed and brought into implementation in the robotic field but none worked as fine as it should and has been rejected very soon than expected. The Wireless Video Surveillance Robot is used to overcome all these technological lacking and unavailability in the technological field to bring about some overdue changes in this field. We controlled overall mechanism of robot using web page. Thus this method is reliable over wide range. We can control this robot by sitting anywhere [3].

1. **REQURIMENT ANALYSIS**

The novel idea of this project is to develop efficient and cost effective automated real time Monitoring/ Spying Robot. The Prototype developed consists of three sections.

1. Input section consisting of Raspberry PI camera
2. Processing Unit consisting of Raspberry Pi
3. Output Unit consists of Web based Monitoring.

**5.1 Project Requirements**

1. **Block Diagram**

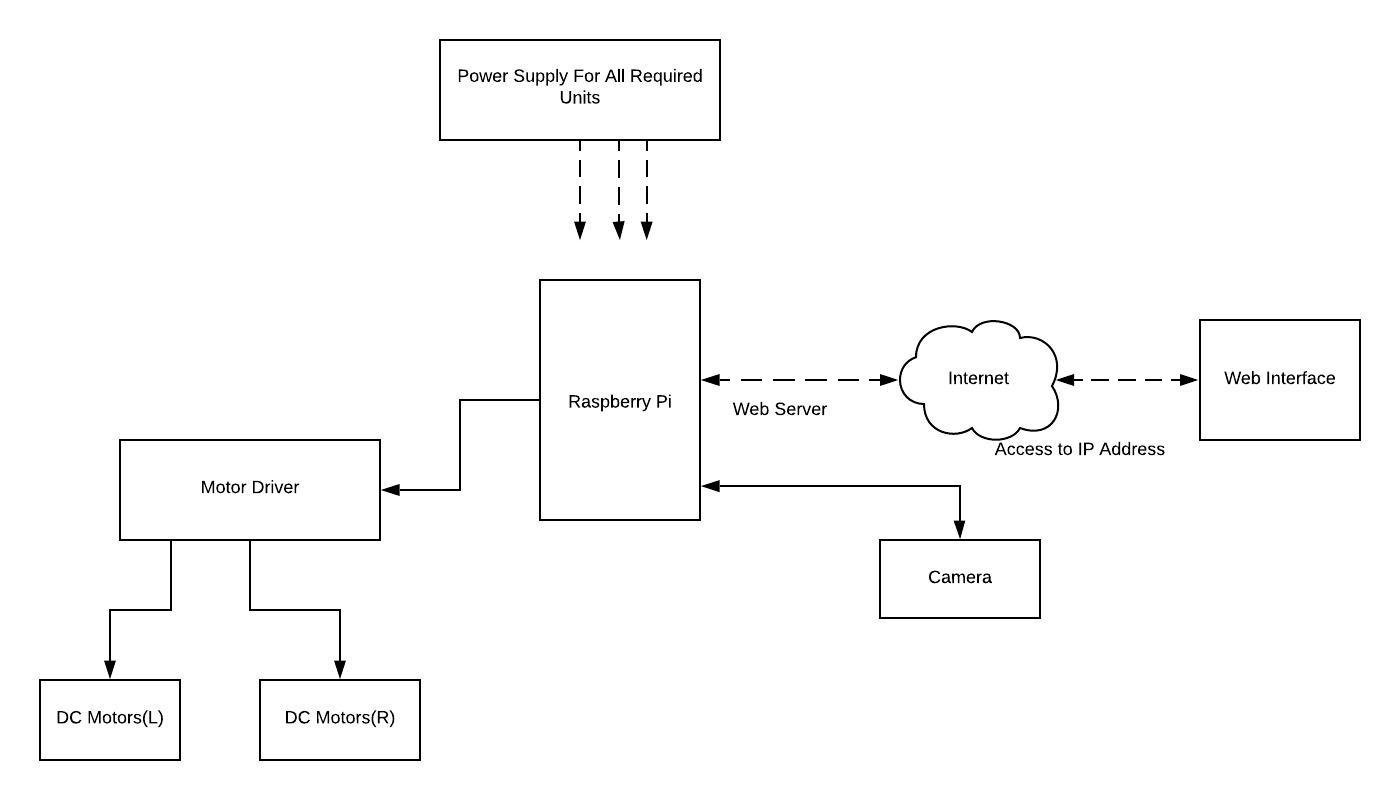


Figure: 1: Block Diagram

The hardware required for this projects are Raspberry Pi, DC Motors, Motor Driver and Pi Camera with some Power supply Units.

1. **Connection Diagram**

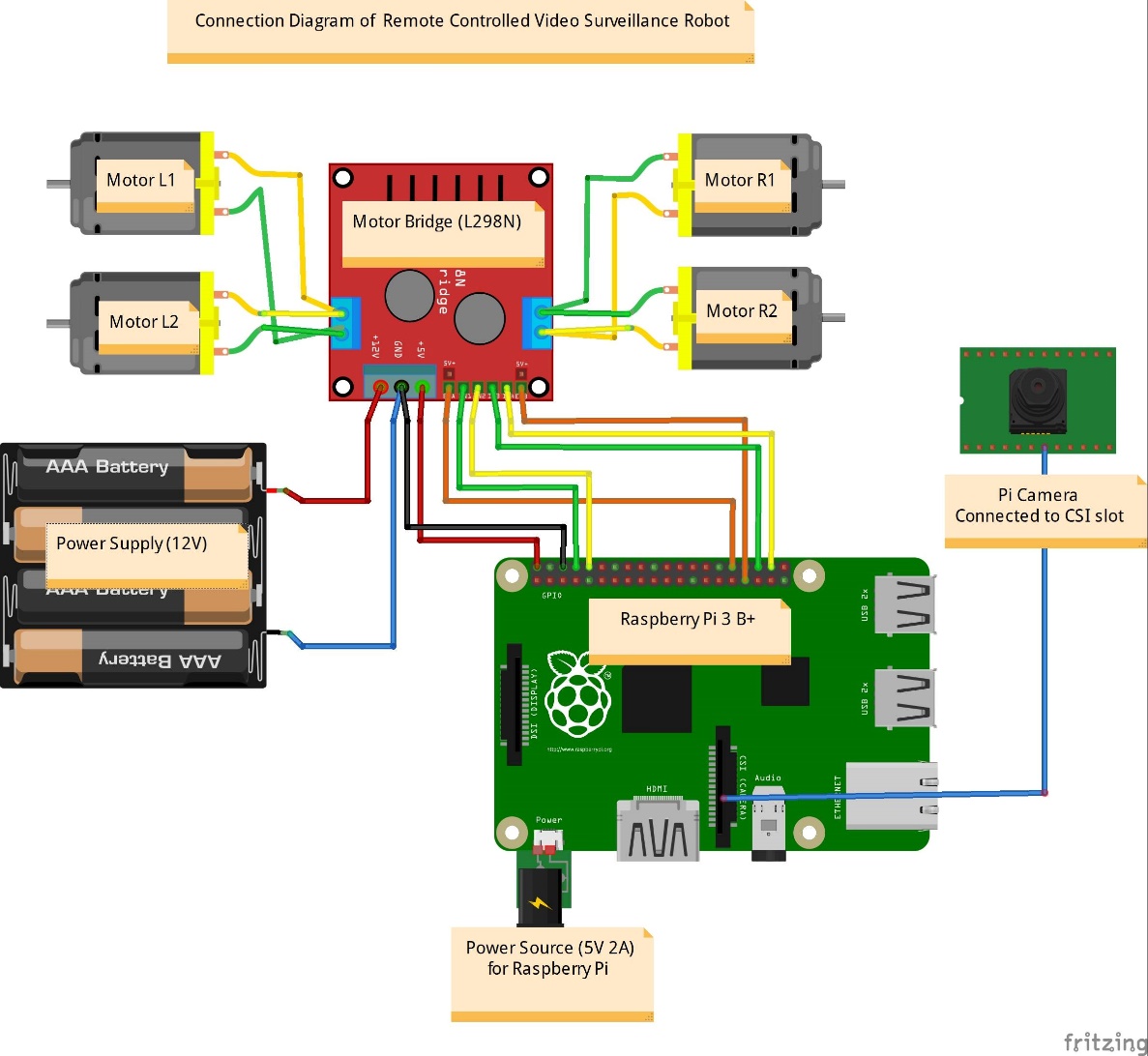


Figure.2 Connection Diagram of Remote Controlled Video Surveillance Robot

(*Source:* Fritzing)

**5.2 Hardware Requirement**

**1. Raspberry Pi**

A Raspberry Pi is a credit card-sized computer originally designed for education, inspired by the 1981 BBC Micro. Creator Eben Upton’s goal was to create a low-cost device that would improve programming skills and hardware understanding at the pre-university level. But thanks to its small size and accessible price. It was quickly adopted by tinkerers, makers, and electronics enthusiasts for projects that require more than a basic microcontroller (such as Arduino devices). The Raspberry Pi is slower than a modern laptop or desktop but is still a complete Linux computer and can provide all the expected abilities that implies at a low-power consumption level. The Raspberry pi is open hardware, with the exception of the primary chip on the Raspberry Pi, the Broadcom SOC (System on a Chip), which runs many of the main components of the board-CPU, graphics, memory, the USB controller, etc. Many of the projects made with a Raspberry Pi are open and well-documented as well. We are using raspberry Pi 3 b+ module for our project[4].

The Raspberry Pi 3 Model B+ is the latest product in the Raspberry Pi 3 range, boasting a 64-bit quad core processor running at 1.4GHz, dual-band 2.4GHz and 5GHz wireless LAN, Bluetooth 4.2/BLE, faster Ethernet, and PoE capability via a separate PoE HAT .

The dual-band wireless LAN comes with modular compliance certification, allowing the board to be designed into end products with significantly reduced wireless LAN compliance testing, improving both cost and time to market. The Raspberry Pi 3 Model B+ maintains the same mechanical footprint as both the Raspberry Pi 2 Model B and the Raspberry Pi 3 Model B.

**1.1 Specification:**

Processor : Broadcom BCM2837B0, Cortex-A53 64-bit SoC @ 1.4GHz

Memory : 1GB LPDDR2 SDRAM

Connectivity :

* + - * 2.4GHz and 5GHz IEEE 802.11.b/g/n/ac wireless LAN, Bluetooth 4.2, BLE
      * Gigabit Ethernet over USB 2.0 (maximum throughput 300Mbps)
      * 4 × USB 2.0 ports

Access : Extended 40-pin GPIO header

Video & sound:

* + - 1 × full size HDMI
    - MIPI DSI display port
      * MIPI CSI camera port
      * 4 pole stereo output and composite video port

Multimedia : H.264, MPEG-4 decode (1080p30); H.264 encode (1080p30); OpenGL ES 1.1, 2.0 graphics

SD card support: Micro SD format for loading operating system and data storage

Input power :

* + - * 5V/2.5A DC via micro USB connector
      * 5V DC via GPIO header
      * Power over Ethernet (PoE)–enabled (requires separate PoE HAT)

Environment : Operating temperature, 0–50°C

Production lifetime: The Raspberry Pi 3 Model B+ will remain in production until at least January 2023.

**1.2 Features**

**CPU (Central Processing Unit)**

The central processing unit is the brain of the raspberry pi board and that is responsible for carrying out the instruction of the computer through logical and mathematical operations. The raspberry pi uses ARM11 series processor, which has joined the ranks of the Samsung galaxy phone.

**GPU (Graphics Processing Unit)**

The GPU is a specialized chip in the raspberry pi board and that is designed to speed up the operation of image calculations. This board designed with a Broadcom video core IV and it supports OpenGL.

**GPIO Pins**

The general purpose input & output pins are used in the raspberry pi to associate with the other electronic boards. These pins can accept input & output commands based on programming raspberry Pi. The raspberry pi affords digital GPIO pins. These pins are used to connect other electronic components. For example, you can connect it to the temperature sensor to transmit digital data.

**Power Source Connector**

The power source cable is a small switch which is placed on side of the shield. The main purpose of the power source connector is to enable an external power source.

**UART**

The Universal Asynchronous Receiver/Transmitter is a serial input &Output ports. That can be used to transfer the serial data in the form of text and it is useful for converting the debugging code.

**Display**

The connection option of the raspberry pi board are two types such as HDMI and Composite. Many LCD and HD TV monitors can be attached using an HDMI male cable and with a low-cost adaptor. The version of HDMI are 1.3 and 1.4 are supported and 1.4 version cable is recommended. The O/Ps of the Raspberry Pi audio and video through HDMI, but does not support HDMI I/P. Older TVs can be connected using composite video. When using a composite video connection, audio is available from the 3.5mm jack socket and can be sent to your TV, you need a cable which adjusts from 3.5mm to double RCA connectors.

**GPIO Pinout Diagram**

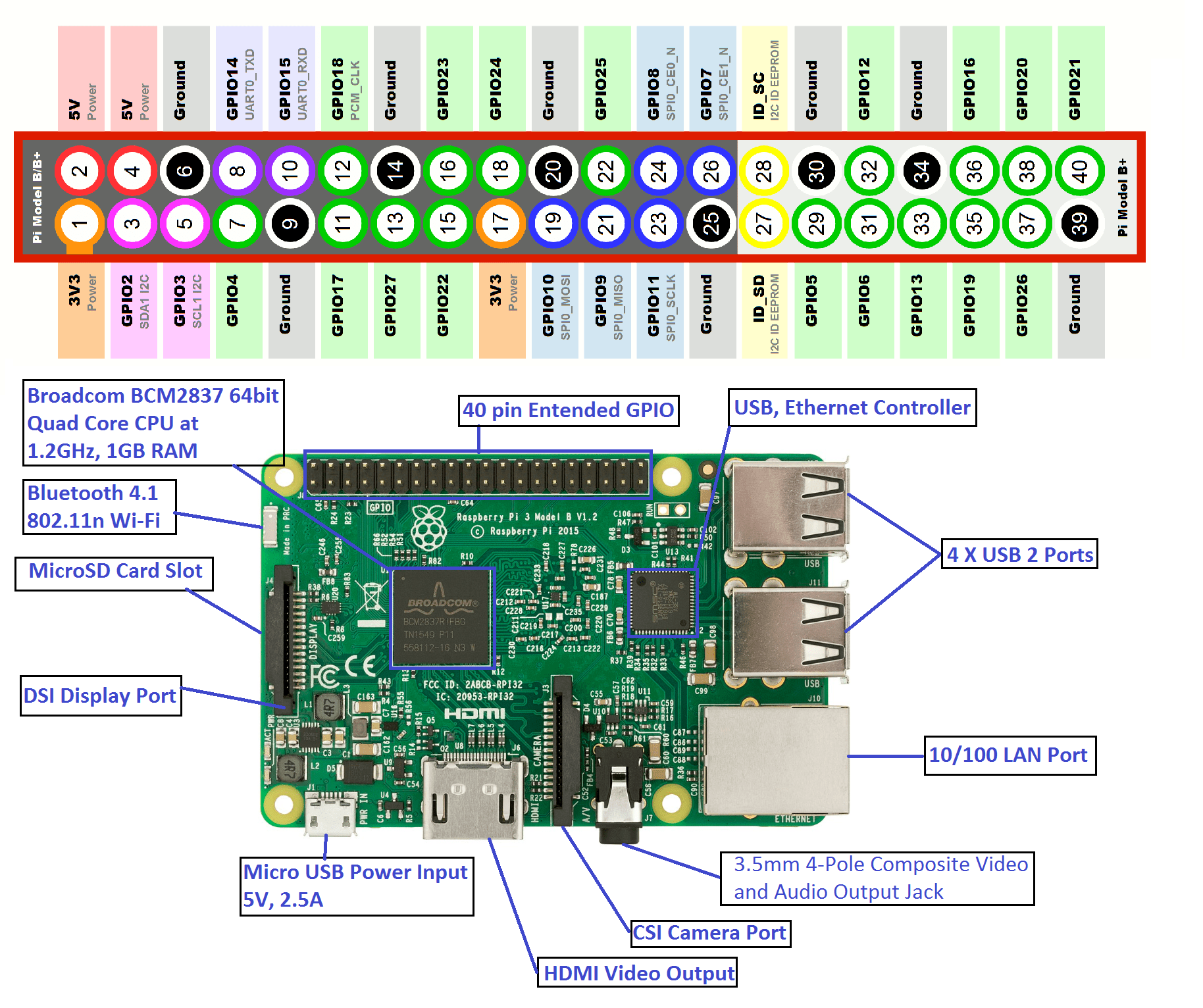
****

Figure: 3pin diagram and raspberry Pi (*Source*: https://electronicshobbyists.com)

**2 DC Motors**

A [DC motor is an electric motor](https://www.edgefxkits.com/speed-control-unit-designed-for-a-dc-motor) that runs on direct current power. In any electric motor, operation is dependent upon simple electromagnetism. A current carrying conductor generates a magnetic field, when this is then placed in an external magnetic field, it will encounter a force proportional to the current in the conductor and to the strength of the external magnetic field. It is a device which converts electrical energy to mechanical energy. It works on the fact that a current carrying conductor placed in a magnetic field experiences a force which causes it to rotate with respect to its original position.

Figure: 4 DC Motor (*Source:* <https://www.wikipedia.org/>*)*

Nearly all type DC motors have some internal mechanism, either electromechanical or electronic to periodically change the direction of current flow in part of the motor. Most types produce rotary motion; a linear motor directly produces force and motion in a straight line. DC motors were the first widely used, since they could be powered from existing direct-current lighting power distribution systems. A DC motor’s speed can be controlled over a wide range, using a variable supply voltage or by changing the strength of current in its field windings. Small DC motors are used in tools, toys, and appliances. The universal moto can operate on direct current but is a lightweight motors used for portable power tools and appliances. Larger DC motors are used in propulsion of rolling mills. The advent of power electronics has made replacement of DC motors with AC motors with AC motors possible in many applications.

A Simple DC motor has stationary set of magnets in the stator and an armature with one or more windings of insulated wire wrapped around a soft iron core that concentrates the magnetic field. The windings usually have multiple turns around the core, and in large motors there can be several parallel current paths. The end of the wire windings are connected to commutator. The commutator allows each armature coil to be energized in turn and connects the rotation coils with the external power supply through brushes. (Brushless DC motors have electronics that switch the DC current to each coil on and off and have no brushes.

The total amount of current sent to the coil, the coil’s size and what it’s wrapped around dictate the Strength of the electromagnetic field created.

The sequence of turning a particular coil on or off dictates what direction the effective electromagnetic fields are pointed. By turning on and off coils ion sequences a rotating magnetic field can be created. These rotating magnetic field interact with the magnetic fields of the magnets (Permanent or electromagnets) in the stationary part of the motor (stator) to create a force on the armature which causes it to rotate. In some DC moto design the stator fields use electromagnets to create their magnetic fields which allow the motor. At high power levels, DC motors are almost always cooled using forced air.

1. **Motor Driver L298N**

The L298N Motor Driver Module is a high voltage Dual H-Bridge manufactured by ST Company. It is designed to accept standard TTL voltage levels. H-bridge drivers are used to drive inductive loads that requires forward and reverse function with speed control such as DC Motors, and Stepper Motors. This Dual H-Bridge driver is capable of driving voltages up to 46V and continuous current up to 2A in each channels.

The L298N is a dual H-Bridge motor driver which allows speed and direction control of two DC motors at the same time.

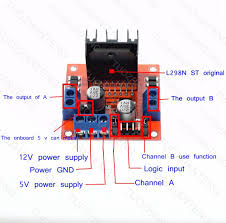


Figure: 5 L298N Motor Controller Board (*Source:* <https://www.teachmemicro.com/>*)*

The controller board, shown above, has +12V and +5V terminals. The +12V pin is where the motor power is attached. This pin can accept voltages from +7VDC to +35VDC

**Specification**

* Operating Voltage: 7 to 35V
* Peak current: 2A
* Maximum power consumption: 20W (when the temperature T = 75 °C)
* Driver Board Size: 55mm \* 49mm \* 33mm (with fixed copper pillar and the heat sink height)
* Driver Board Weight: 33

**Connection of DC Motors with L298N Driver Board**

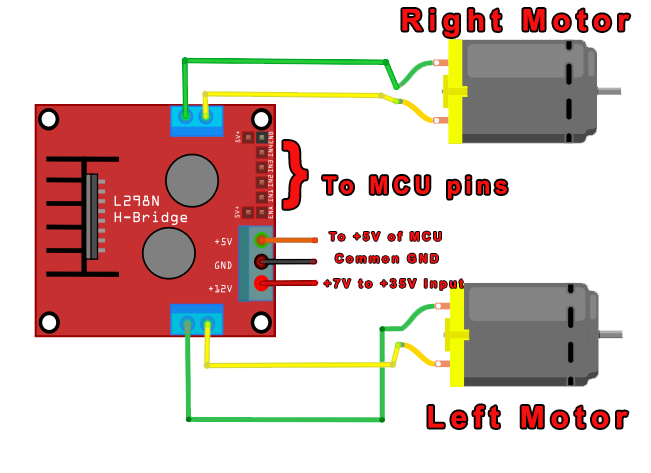
****

Figure 6: Connection of DC motors with L298N Driver board (*Source:* <https://www.teachmemicro.com/use-l298n-motor-driver/>)

Using the L298N if straightforward. If you want the left motor to rotate in one direction, apply a high pulse to IN1 and a low pulse to IN2. To reverse the direction, reverse the pulses to IN1 and IN2. The same applies to the right motor.

Here’s a table that summarizes the pins and corresponding motor direction.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| IN1 | IN2 | IN3 | IN4 | Direction |
| 0 | 0 | 0 | 0 | Stop |
| 1 | 0 | 1 | 0 | Forward |
| 0 | 1 | 0 | 1 | Reverse |
| 1 | 0 | 0 | 1 | Left |
| 0 | 1 | 1 | 0 | Right |

Figure: 7 L298N Logic Table (*Source:* <https://www.teachmemicro.com/use-l298n-motor-driver/>)

**4. Raspberry Pi Camera module**

The Raspberry pi camera module cam be used to take high definition video, as well as stills photographs. It’s easy to use for beginners, but has plenty to offer advanced users if you’re looking to expand your knowledge. There are lots of examples online of people using it for time-lapse, slow-motion and other video cleverness. You can also use the libraries we bundle with the camera to create effects.

If you’re interested in the nitty-gritty, you’ll want to know that the module has a five megapixel fixed-focused camera that supports 1080p30, 720p60 and VGA90 video modes, as well as stills capture. It attaches via a 15 cm ribbon cable to the CSI port on the Raspberry pi. It can be accessed through the MMAL and V4L APIs, and there are numerous third-party libraries built for it, including the pi camera Python library.

The camera module is very popular in home security applications, and in wildlife camera traps.

You can also use it to take snapshots.

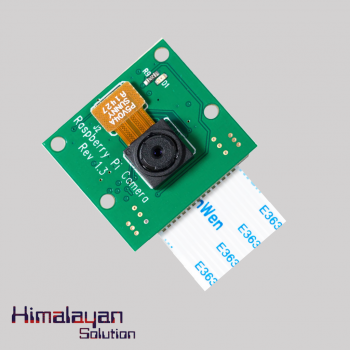


Figure: 8 Raspberry Pi camera

(*Source:* <https://himalayansolution.com/product/raspberry-pi-camera>)

**Features of Camera**

* **5**Mp sensor
* Wider image, capable of 2592x1944 stills, 1080p30 video
* 1080p video supported
* CSI Slot
* Size:25 x 20 x 9 mm

The camera consists of a small (25mm by 20mm by 9mm) circuit board, which connects to the Raspberry Pi’s camera serial Interface (CSI) bus connector via a flexible ribbon cable. The camera’s image sensor has a native resolution of five megapixels and has a fixed ribbon cable. The camera’s image sensor has a native resolution of five megapixels and has a fixed focus lens. The software for the camera supports full resolution still images up to 2592x1944 and video resolution of 1080p30, 720p60 and 640x480p60/90.

Installation involves connecting the ribbon cable to the CSI connector on the Raspberry Pi board. The flex cable inserts into the connector situated between the Ethernet and HDMI ports, with the silver connectors facing the HDMI port. The flex cable connector should be opened by pulling the tabs on the top of the connector upwards then towards the Ethernet port. The flex acute an angle. The top part of the connector should then be pushed towards the HDMI connector and down, while the flex cable is held in place.

**5. Power Supply**

# For the power supply purpose we have used Li-Po Battery 22000mAh which has the following specification:

* Capacity: 22000mAh
* Voltage: 11.1V
* Discharge Rate: 60C
* Configuration: 3S1P(3 Cell)

# Quickly recharged, safe and Portable

Figure: 9 Lipo Rechargeable battery

(Source: https://himalayansolution.com/product/lipo-rechargeable-battery/)

**5.3 Software Requirement/Programming**

Raspbian operating system is Linux based operating system and specially designed for raspberry pi. It has built in support for python programming language. So we are using python programming language for our project. The GPIO library of python programming language can be used to program GPIO pins of raspberry pi.

Raspbian is a [Debian](https://en.wikipedia.org/wiki/Debian)-based [computer operating system](https://en.wikipedia.org/wiki/Operating_system) for [Raspberry Pi](https://en.wikipedia.org/wiki/Raspberry_Pi). There are several versions of Raspbian including Raspbian Buster and Raspbian Stretch. Since 2015 it has been officially provided by the [Raspberry Pi Foundation](https://en.wikipedia.org/wiki/Raspberry_Pi_Foundation) as the primary operating system for the family of Raspberry Pi [single-board computers](https://en.wikipedia.org/wiki/Single-board_computers). Raspbian was created by Mike Thompson and Peter Green as an independent project. The initial build was completed in June 2012. The operating system is still under active development. Raspbian is highly optimized for the Raspberry Pi line's low-performance [ARM](https://en.wikipedia.org/wiki/ARM_architecture) CPUs.

Raspbian OS is official Operating System available for free to use. This OS is efficiently optimized to use with Raspberry Pi. Raspbian have GUI which includes tools for Browsing, Python programming, office, games, etc. Raspbian Os comes with different debian versions: Wheezy, Jessie, Stretch, and Buster. The one which we are using is Raspbian Stretch.

Beside python programming language we have use different software used to convert video taken by camera to its standard Mp4 format.

Below are the list of some software and tools that we have used in our projects.

**1. Python**

Python is a widely used high-level programming language for general-purpose programming, created by Guido van Rossum and first released in 1991. Python features a dynamic type system and automatic memory management and supports multiple programming paradigms, including object-oriented, imperative, functional programming, and procedural styles. It has a large and comprehensive standard library[5].

Two major versions of Python are currently in active use:

Python 3.x is the current version and is under active development.

Python 2.x is the legacy version and will receive only security updates until 2020. No new features will be implemented. Note that many projects still use Python 2, although migrating to Python 3 is getting easier.

It’s great as a first language because it is concise and easy to read, and it is also a good language to have in any programmer’s stack as it can be used for everything from web development to software development and scientific applications.

* Python works on different platforms (Windows, Mac, Linux, Raspberry Pi, etc).
* Python has a simple syntax similar to the English language.
* Python has syntax that allows developers to write programs with fewer lines than some other programming languages.
* Python runs on an interpreter system, meaning that code can be executed as soon as it is written. This means that prototyping can be very quick.
* Python can be treated in a procedural way, an object-orientated way or a functional way.

### **What can Python do?**

* Python can be used on a server to create web applications.
* Python can be used alongside software to create workflows.
* Python can connect to database systems. It can also read and modify files.
* Python can be used to handle big data and perform complex mathematics.
* Python can be used for rapid prototyping, or for production-ready software development.

**2. Flask:**

Flask is a micro web framework written in Python. It is classified as a micro framework because it does not require particular tools or libraries. It has no database abstraction layer, form validation, or any other components where pre-existing third-party libraries provide common functions. It is a lightweight [WSGI](https://wsgi.readthedocs.io/) (Web Server Gateway Interface) web application framework. It is designed to make getting started quick and easy, with the ability to scale up to complex applications. It began as a simple wrapper around [Werkzeug](https://www.palletsprojects.com/p/werkzeug) and [Jinja](https://www.palletsprojects.com/p/jinja) and has become one of the most popular Python web application frameworks.

These distributions will be installed automatically when installing Flask.

* [Werkzeug](https://palletsprojects.com/p/werkzeug/) implements WSGI, the standard Python interface between applications and servers.
* [Jinja](https://palletsprojects.com/p/jinja/) is a template language that renders the pages your application serves.
* [MarkupSafe](https://palletsprojects.com/p/markupsafe/) comes with Jinja. It escapes untrusted input when rendering templates to avoid injection attacks.
* [ItsDangerous](https://palletsprojects.com/p/itsdangerous/) securely signs data to ensure its integrity. This is used to protect Flask’s session cookie.
* [Click](https://palletsprojects.com/p/click/) is a framework for writing command line applications. It provides the Flask command and allows adding custom management commands.

**Flask Installation:**

Pip3 install Flask

Sample Flask Program:

from flask import Flask

app = Flask(\_\_name\_\_)

@app.route('/')

def hello\_world():

return 'Hello, World!'

if \_\_name\_\_ == '\_\_main\_\_':

app.run()

The above code shows "Hello, World!" on localhost port 5000 in a web browser when run with the Python app.py Command and the Flask library installed.

**3. jQuery**

jQuery is a [JavaScript library](https://en.wikipedia.org/wiki/JavaScript_library) designed to simplify [HTML](https://en.wikipedia.org/wiki/HTML) [DOM](https://en.wikipedia.org/wiki/Document_Object_Model) tree traversal and manipulation, as well as [event handling](https://en.wikipedia.org/wiki/Event_handling), [CSS animation](https://en.wikipedia.org/wiki/CSS_animation), and [Ajax](https://en.wikipedia.org/wiki/Ajax_(programming)). It is [free, open-source software](https://en.wikipedia.org/wiki/Free_and_open_source_software) using the permissive [MIT License](https://en.wikipedia.org/wiki/MIT_License).

JQuery is a cross-platform JavaScript library designed to simplify the client- side scripting of HTML. JQuery is the most popular JavaScript in use today.jQuery is a lightweight, "write less, do more", JavaScript library.The purpose of jQuery is to make it much easier to use JavaScript on your website.

jQuery takes a lot of common tasks that require many lines of JavaScript code to accomplish, and wraps them into methods that you can call with a single line of code. jQuery also simplifies a lot of the complicated things from JavaScript, like AJAX calls and DOM manipulation.

**4. MJPEG Streamer**

Mjpg-streamer is a command line application that copies JPEG frames from one or more input plugins to multiple output plugins. It can be used to stream JPEG files over an IP-based network from a webcam to various types of viewers such as Chrome, Firefox, Cambozola, VLC, mplayer, and other software capable of receiving MJPG streams.

It was originally written for embedded devices with very limited resources in terms of RAM and CPU. Its predecessor "uvc\_streamer" was created because Linux-UVC compatible cameras directly produce JPEG-data, allowing fast and perfomant M-JPEG streams even from an embedded device running OpenWRT. The input module "input\_raspicam.so" captures such JPG frames from a connected webcam. Mjpg-streamer now supports a variety of different input devices.

**5. FFMPEG**

FFMPEG is an open source multimedia project for working with audio and video. Based on the "libavcodec" A/V codec library and "libavformat" multiplexing framework, **FFmpeg** is a command-line utility that can encode and decode a variety of media formats.

FFmpeg is a set of open source audio and video tools for recording, converting, and streaming multimedia content. It supports nearly every digital format and codec known, from the old and obscure to the cutting edge. The toolset is highly portable available on most operating systems and platforms

## **FFmpeg Tools:**

The FFmpeg project distributes four major applications:

* FFmpeg — A command-line utility that can be used to process, convert, or manipulate media. This tool is the foundation for the other applications.
* FFplay — A simple media player.
* FFserver — A streaming media web server.
* FFprobe — A stream analysis tool.

FFmpeg is built with a number of self-contained libraries which provide discreet functionality that can be included into other applications. These features include codec encoding and decoding, compression, image scaling, resampling, and format conversion.

**6. HTTP**

HTTP (Hypertext Transfer Protocol) is the set of rules for transferring files, such as text, graphic images, sound, video, and other multimedia files, on the World Wide Web. As soon as a Web user opens their Web browser, the user is indirectly making use of HTTP. HTTP is an application protocol that runs on top of the TCP/IP suite of protocols (the foundation protocols for the Internet). The latest version of HTTP is HTTP/2, which was published in May 2015. It is an alternative to its predecessor, HTTP 1.1, but does not it make obsolete.

**How HTTP works**

As the Hypertext part of the name implies, HTTP concepts include the idea that files can contain references to other files whose selection will elicit additional transfer requests. In addition to the Web page files it can serve, any Web server machine contains an HTTP daemon, a program that is designed to wait for HTTP requests and handle them when they arrive. A Web browser is an HTTP client, sending requests to server machines. When the browser user enters file requests by either "opening" a Web file (typing in a URL) or clicking on a hypertext link, the browser builds an HTTP request and sends it to the Internet Protocol address (IP address) indicated by the URL. The HTTP daemon in the destination server machine receives the request and sends back the requested file or files associated with the request. As a note, a Web page often consists of more than one file.

**7. PUTTY**

Putty is an SSH and telnet client, developed originally by Simon Tat ham for the Windows platform. Putty is open source software that is available with source code and is developed and supported by a group of volunteers. IP address of Raspberry Pi have to enter and raspberry pi is ready for programming

**SSH Protocol**

The SSH protocol (also referred to as Secure Shell) is a method for secure remote login from one computer to another. It provides several alternative options for strong authentication, and it protects the communications security and integrity with strong encryption. It is a secure alternative to the non-protected login protocols (such as [telnet](https://www.ssh.com/ssh/telnet), rlogin) and insecure file transfer methods (such as [FTP](https://www.ssh.com/ssh/ftp/)).

## TYPICAL USES OF THE SSH PROTOCOL

The protocol is used in corporate networks for:

* providing secure access for users and automated processes
* interactive and automated file transfers
* issuing remote commands
* Managing network infrastructure and other mission-critical system components.

## HOW DOES THE SSH PROTOCOL WORK

The protocol works in the client-server model, which means that the connection is established by the SSH client connecting to the SSH server. The SSH client drives the connection setup process and uses public key cryptography to verify the identity of the SSH server. After the setup phase the SSH protocol uses strong symmetric encryption and hashing algorithms to ensure the privacy and integrity of the data that is exchanged between the client and server.

The figure below presents a simplified setup flow of a secure shell connection.

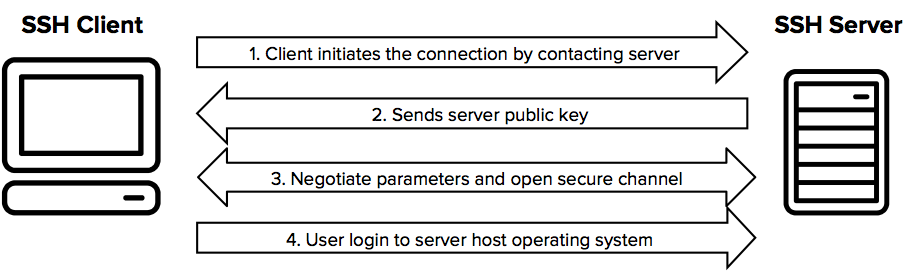


Figure: Secured Shell Connection (*Source:* <https://www.ssh.com/ssh/protocol/>*)*

1. **METHODOLOGY**

[The system consists of two major sections - one is the](https://www.ijert.org) [user section and other is the robot section. The user](https://www.ijert.org) [section may be a laptop or mobile for communicating](https://www.ijert.org) [with the robot end through the web browser. Thus by using a laptop or a mobile the](https://www.ijert.org) [user section can be a portable one compared to those that](https://www.ijert.org) [uses a typical stationary computer system. The](https://www.ijert.org) communication can be performed with IoT over Wi-Fi. The robot end consists of a four wheel chassis controlled over Raspberry Pi. The camera takes the frames of surrounding ahead which can be viewed in the web browser as M-JPEG video. The robot is controlled manually from the web browser based on the video transmitted.

**6.1 Flowchart**

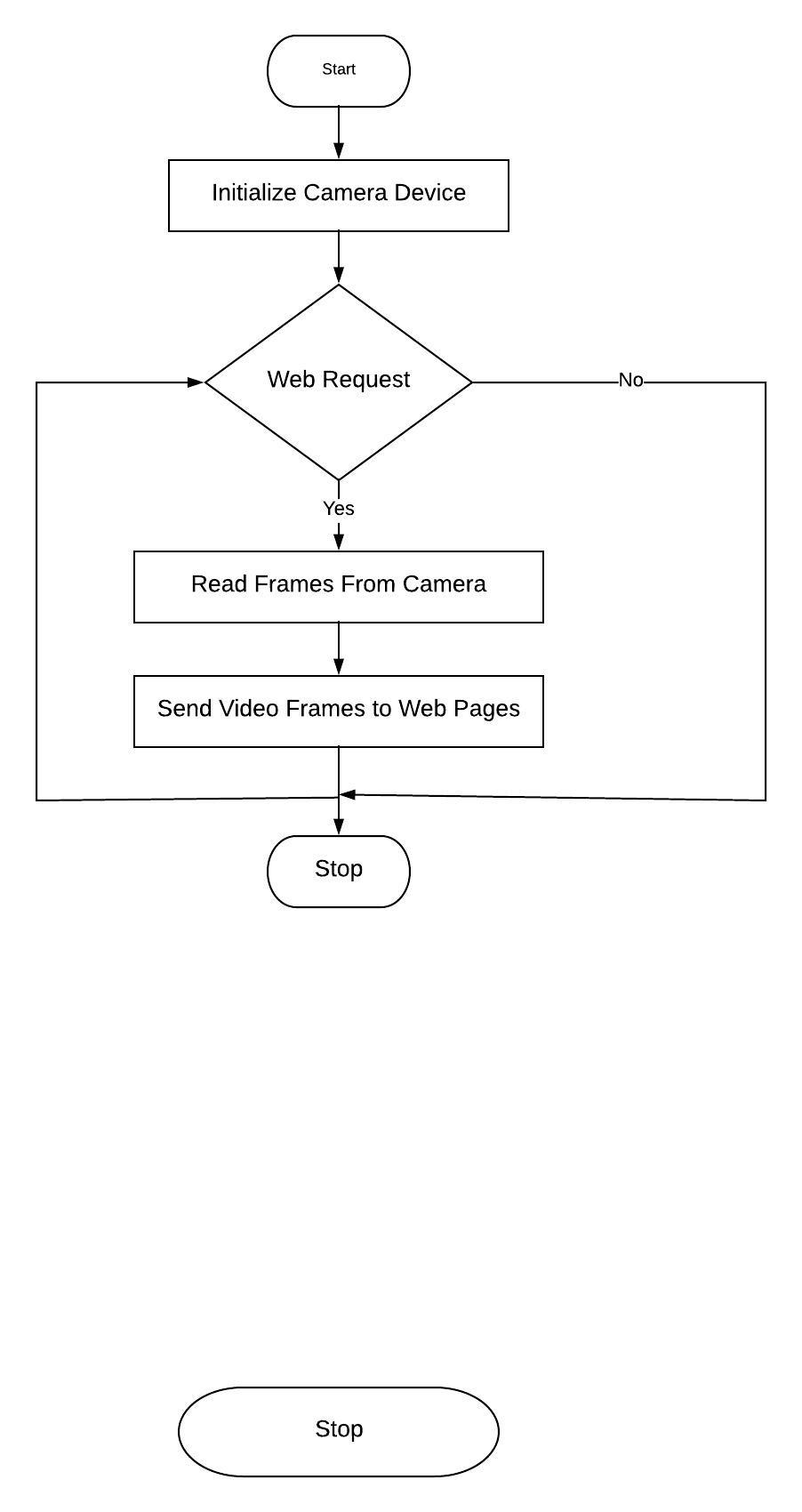
****

Fig.11 Flowchart for video surveillance

1. **IMPLEMENTATION DETAILS**

The raspbian operating system is burned into sixteen GB SD card using Computer. Then the SD card is inserted into raspberry Pi. The raspberry pi is connected to computer with Ethernet cable and is accessed from computer by using ssh protocol. We have install Python for python interpretrer in raspberry Pi. We used Python IDLE for python programming and store the program into raspberry pi.

The command list to install different software in raspberry pi are as:

* To Update raspberry Pi

sudo apt-get update

* To install python

sudo apt-get install python

* To access the raspberry pi from computer is

ssh pi@[ipaddress]

**RPi.GPIO:**

A powerful feature of the Raspberry Pi is the row of GPIO (general-purpose input/output) pins along the top edge of the board. Raspberry-gpio-python or RPi.GPIO, is a Python module to control the GPIO interface on the Raspberry Pi. It was developed by Ben Croston and released under an MIT free software license.

Raspbian OS comes with RPi.GPIO pre-installed.

To import the RPi.GPIO module:

import RPi.GPIO as GPIO

## Pin numbering

There are two ways of numbering the IO pins on a Raspberry Pi within RPi.GPIO. The first is using the BOARD numbering system. This refers to the pin numbers on the P1 header of the Raspberry Pi board. The advantage of using this numbering system is that your hardware will always work, regardless of the board revision of the RPi. You will not need to rewire your connector or change your code.

The second numbering system is the BCM numbers. This is a lower level way of working - it refers to the channel numbers on the Broadcom SOC.

To specify which you are using using (mandatory):

GPIO.setmode(GPIO.BOARD)

*or*

GPIO.setmode(GPIO.BCM)

**PinSetup:**

To setup the pin to input data from sensor.

**GPIO.setup(pin, GPIO.IN)**

To setup the pin to output mode.

**GPIO.setup(pin, GPIO.OUT)**

## **Input**

To read the value of a GPIO pin:

GPIO.input(channel)

(Where channel is the channel number based on the numbering system you have specified (BOARD or BCM)). This will return either 0 / GPIO.LOW / False or 1 / GPIO.HIGH / True.

## **Output**

To set the output state of a GPIO pin:

GPIO.output(channel, state)

State can be 0 / GPIO.LOW / False or 1 / GPIO.HIGH / True.

1. **APPLICATIONS**

* This robot can be used as spy or in similar military purpose.
* It can be used to explore the topography of remote place and also can be used to observe the behavior of strange animals.
* It is also applicable in rescue operation during earthquake.

1. **RESULTS AND OUTPUTS**

Finally on performing all the required procedure, we were able to operate the robot car remotely from the web browser and to give proper and efficient real time monitoring of the selected areas/things. The raspberry Pi successfully provided the automation by the help of camera module, motor driver and web interfacing.

Some Snapshot of our Project:

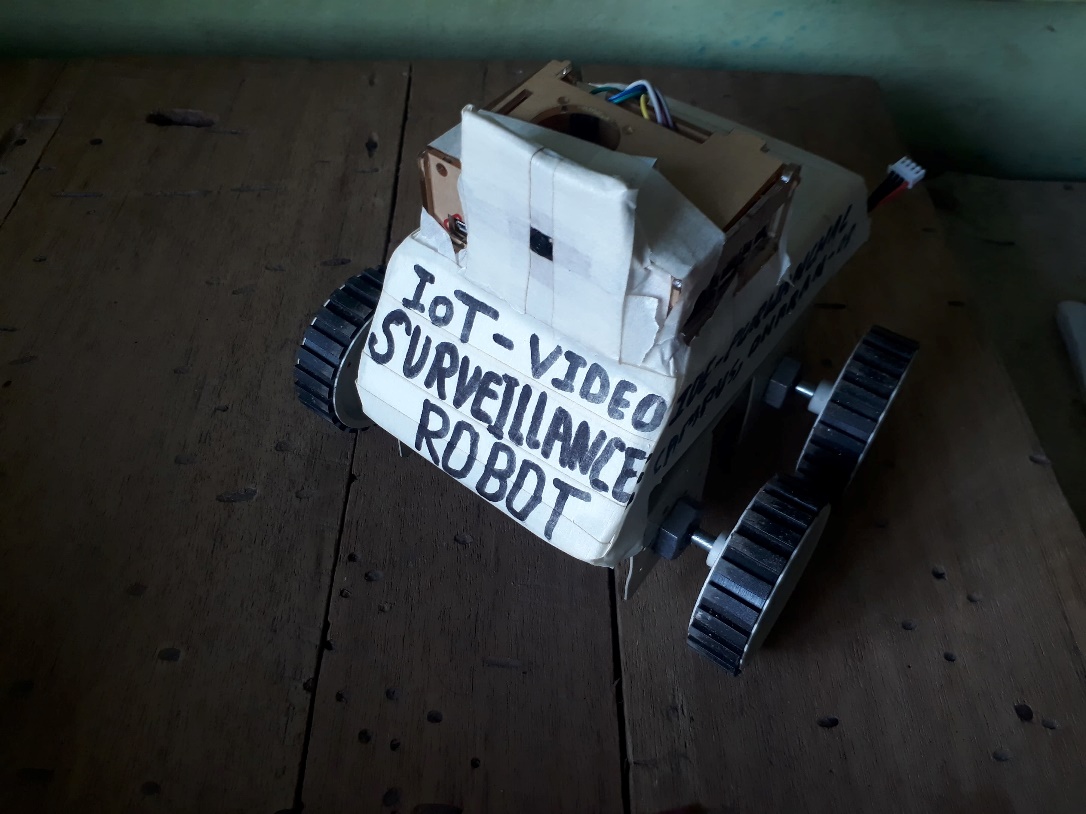


Figure: 12 Outlooks of Robot



Figure: 13 Robot with Web View

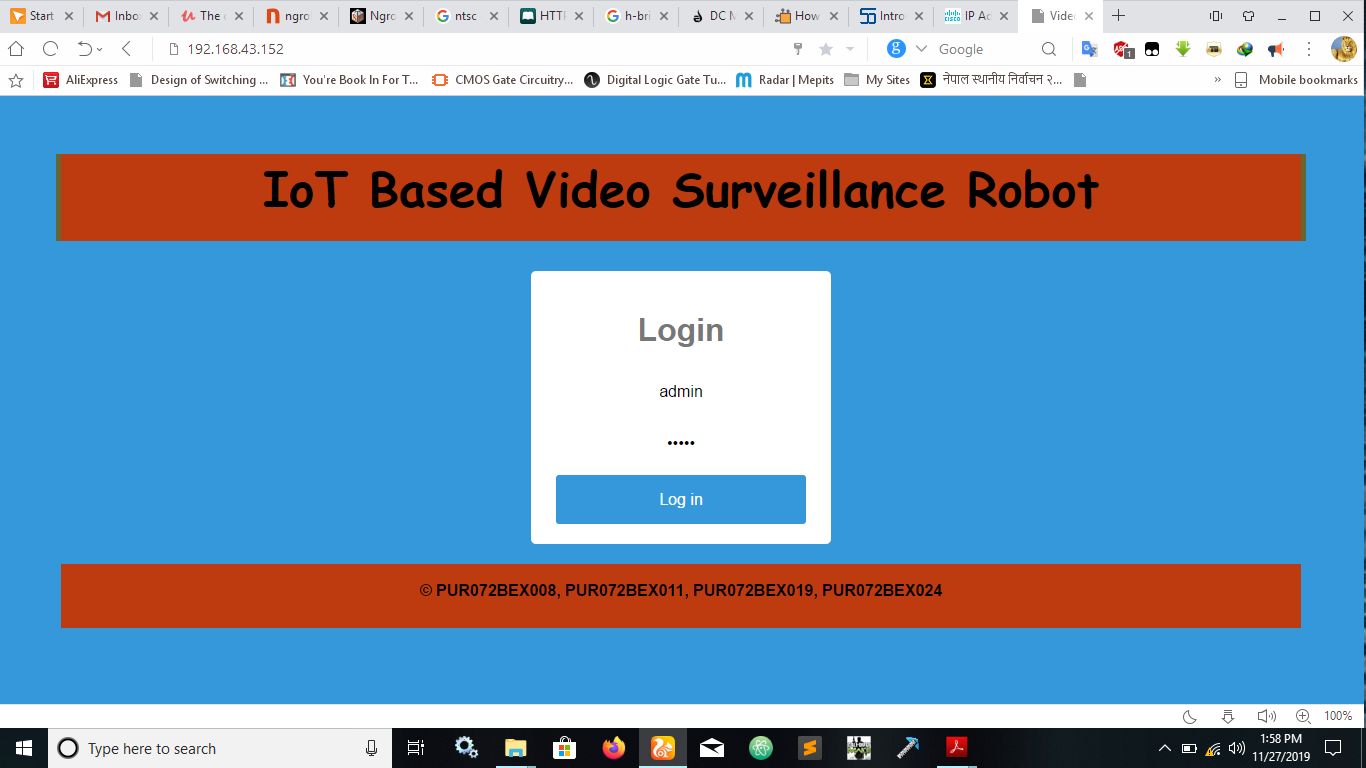


Figure: 14 Login Page

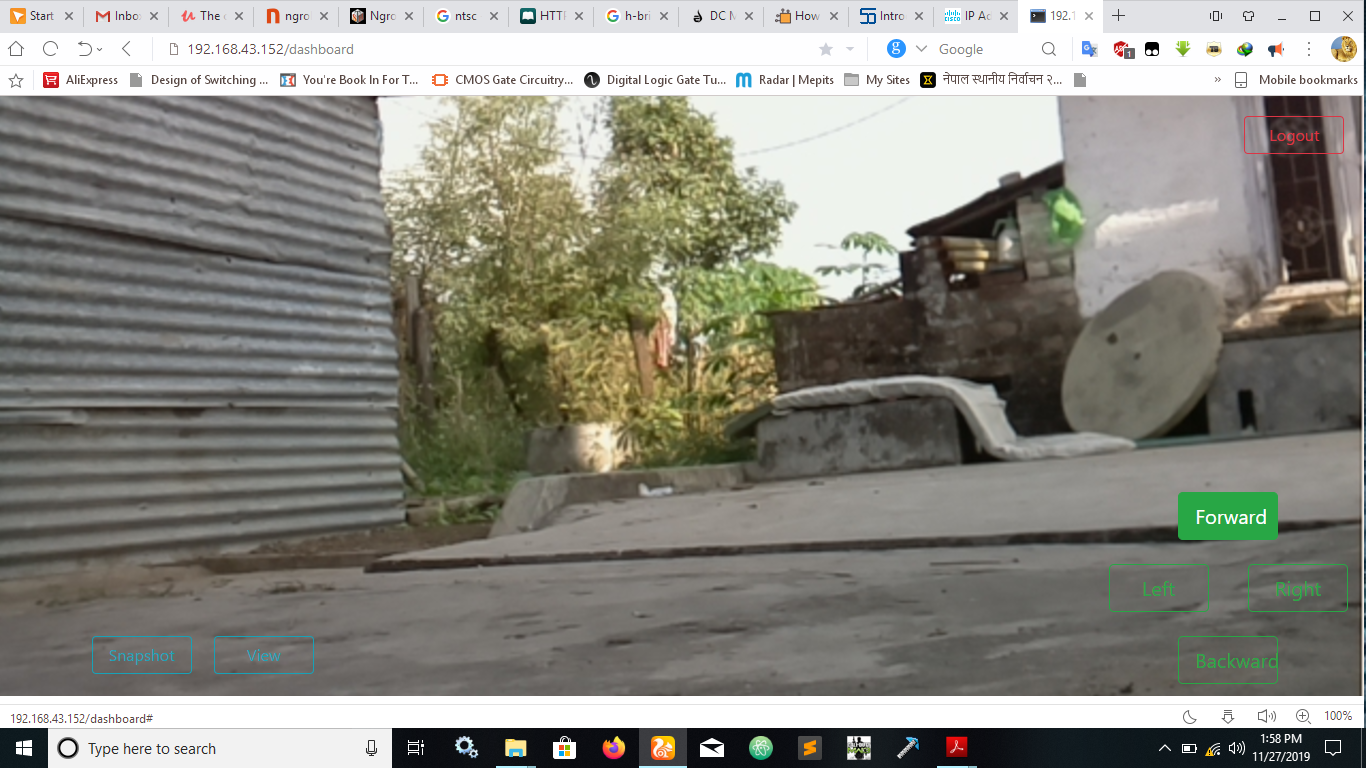
****

Figure: 15 Screenshot of Robot controller page

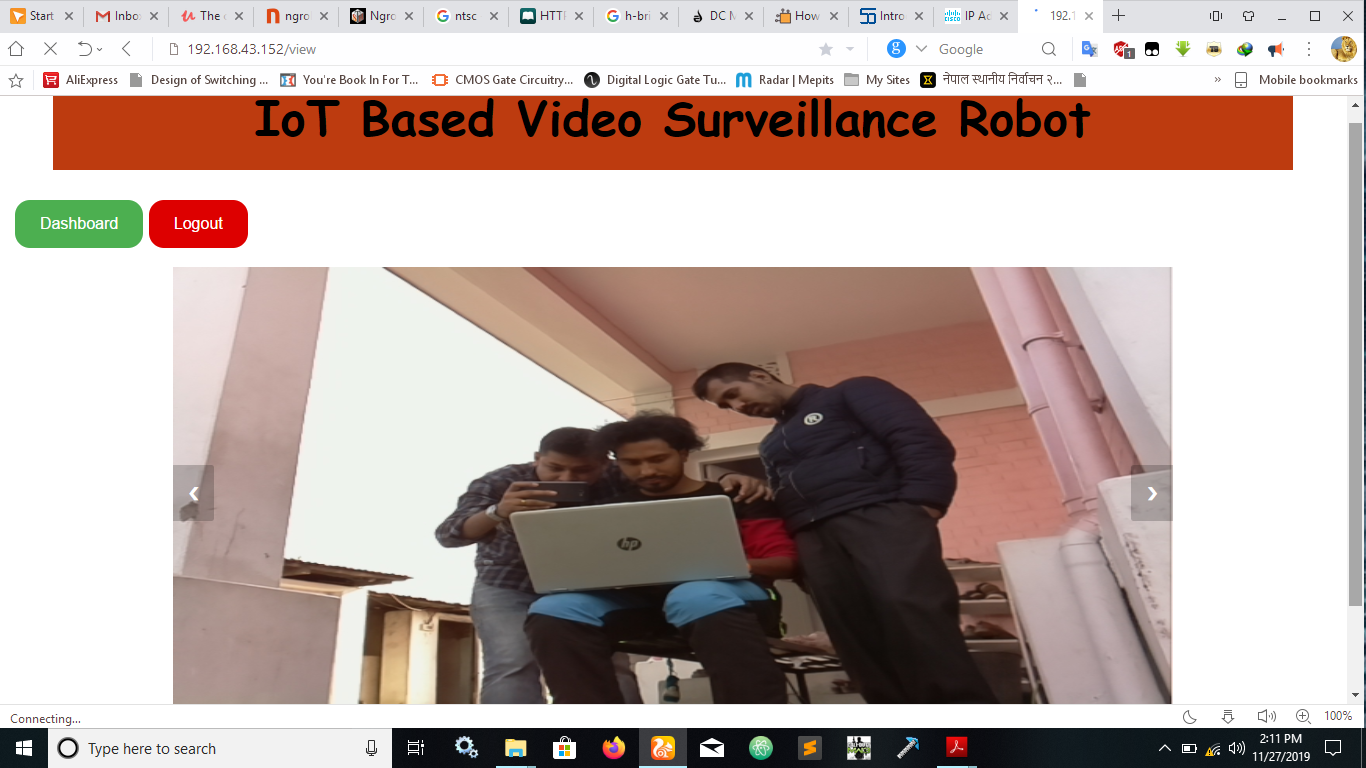


Figure: 16 Screenshot of snapshot taken by Robot

1. **FUTURE ENHANCEMENT**

With the completion of our project, there are certain enhancements that can be done. They are as follows.

* Facial recognition.
* Object Tracking.
* Adding different sensors to measure environmental parameter.
* GPS & GSM can be used for autonomous navigation.
* Ultrasonic Sensor can be added which help to secure the robot from external damage.
* Servo Motor can be attached with Camera which helps to Surveillance in any direction as possible.
* Efficiency can be improved by increasing the battery capacity.

1. **CONCLUSION**

By completing above mentioned work and following the process and steps we were able to accomplish our project. Further extension of present project could also be done by interfacing multiple sensors and High quality equipment thereby improving the performance of the Robot.

**APPENDIX A: COST ESTIMATION**

Table A.1 Expenditure

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| S.N | Component Names | Model No. | Quantity(pcs) | Price/piece  (RS) | Total  (RS) |
| 1 | Raspberry Pi | Pi 3 B+ | 1 | 6500 | 6500 |
| 2 | PI Camera | Pi  Rev 1.5 | 1 | 2000 | 2000 |
| 3 | Pi Display |  | 1 | 1500 | 1500 |
| 4 | Motor Driver (H-Bridge) | L298N | 1 | 600 | 600 |
| 5 | 4 Wheel chasis |  | 1 | 1000 | 1000 |
| 6 | Lipo Battery |  | 1 | 2000 | 2000 |
| 7 | SD Card | SanDisk | 1 | 400 | 400 |
| 8 | Jumper Wires |  | 50 | 10 | 500 |
| 9 | Miscellaneous |  | - | 700 | 700 |
|  | Total Cost |  | - | - | 15200 |

Table 1: Cost Estimation

**NOTE:**

The cost mentioned above is in Nepali Currency as per the present market price.

The cost mentioned above doesn’t include stationary costs.

The cost may vary with the market price as the prices may change.

1. **REFERENCES**
2. Jordan Teicher, "*The little-known story of the first IoT device*". February 7, 2018. [Online].Available:https://www.ibm.com/blogs/industries/little-known-story-first-iot-device
3. G. Anandravisekar, A. Anto Clinton, T. Mukesh Raj, L. Naveen, "*IoT Based Surveillance Robot*".International Journal of Engineering Research & Technology (IJERT). ISSN: 2278-0181, March 2018
4. S. Hafsah, S. Shadma, M. Gangat, “*Wireless Video Surveillance Robot*”. INTERNATIONAL RESEARCH JOURNAL OF ENGINEERING AND TECHNOLOGY (IRJET). ISSN: 2395-0072, April 2019
5. R. Grimmett, 2015. *Raspberry Pi Robotics Projects*. Birmingham B3 2PB, UK: Packt Publishing Ltd.
6. D. Paul, D. Harvey, 2019. *Python ® for Programmers*. Pearson Education, Inc.
7. V. Sriram, A. Sagar,S. Shanmugam, "*IoT Based Real Time Monitoring Mobile Robot*". School of Computer Science and Engineering, VIT University,Vellore, India. ISSN: 2321-9653, 2017
8. www.raspiberry.org