Raspberry Pi Project

Smart Robotic Vehicle

by

Aamir Ahmed

Muhammad Jami Raza

Muhammad Hamza Shafiq

Academic Preparation Course (APC)

Summer Semester 2019

Technical University Ilmenau

Abstract

Nowadays, remote controlled moving devices are widely used in a number of applications. Specially in the field of robotics, remote controlled robots are being used in may areas of life, like in the field of large scale production of commercial products, in the field of medical and also in the field of work where human lives are in danger and there is a huge risk to human life and in the presence of potential safety hazards for example in mine searching, disaster rescue for instance firefighting. Where the involvement of human can be dangerous for them but it is also necessary for the sake of rescue operation to continue, let's take the example of fire in a house, where it is required for the rescue teams to go inside the house to see or rescue the people inside the house. So, in that condition it is better to send a robot instead of human inside the house because human life is more important than materialistic robot.

So, it is the need of today's world to design or develop devices/machines for the advancement of traditional machines for example like robots. Controlling of robot through console or remote control is the conventional way of doing this work. As now a days science has developed itself to a great extent. So, we need to design a robot which can be controlled or operated with the simple human hand gestures unlike conventional consoles which are also difficult to operate and prior knowledge is also required to make it work. But if we change this way of controlling robot to simple way. So, it will ease the use of robot with more efficiency. In this way we can mitigate the safety hazard to human life more easily and we can also use this type of robot in other fields also like in medical.

Contents

1 Project Idea Based on the Task 4

1.1 **Project Idea** 4

1.2 Motivation 4

1.3 The Solution to a Real World Problem 4

1.4 Existing Example 4

1.5 Personal Skills 4

2 Goals Division 5

2.1 Mandatory Goals 5

2.2 Target Goals 5

2.3 Discretionary Goals 5

2.4 Goals Achieved 5

3 User Friendliness & Usability 5

3.1 Design and Characteristics 5

3.2 Problems in Long Time 6

3.3 Solution 6

4 Challenges 6

5 Project Structure Plan 7

5.1 Distribution of Project and Tasks 7

5.2 Sub-Areas Division of Hardware 7

5.2.1 Transmitter 7

5.2.2 Receiver 8

5.3 Block Diagram of Transmitting Part 9

5.4 Block Diagram of Receiver Part 9

6 Code 10

6.1 Main Processor 10

6.1.1 Code 10

6.2 Glove 12

6.2.1 Code 12

6.3 Video Streaming 16

6.3.1 Set Up Hardware and Configuration 16

7 Final Project 18

8 References 19

1 Project Idea Based on the Task

1.1 Project Idea

Computational innovation and Artificial Intelligence has made Human-computer interaction as most emphasizing field. The project we worked on for our Raspberry Pi Project was also based on HCI (Human controlled Interface), Smart robotic vehicle. The main objective of this project was to control the robotic vehicle using human gestures. The human gestures are sensed with the help of an accelerometer also known as inertial sensor and flex sensor. A micro-controller is used in the transmitter section and it will take input from the sensors and these sensed signals are then transmitted to the robotic vehicle via WiFi at the receiver section where these signals will be processed by microprocessor which will control the robot and the this robotic vehicle is also equipped with camera for real time video streaming. Thus, this model will be helpful to avoid danger for the people working in hazardous areas like firefighting, mines and surveillance.

1.2 Motivation

The real motivation for choosing this idea as a topic was to design the robot which can be controlled with hand gesture and also able to send real time video signal from the location where currently the robot is, to the operator so, he can see the scene in front of robot for efficient controlling because the robot will be operated in the remote location so it is necessary for the operator to see the current location of the robot and also the current scene in front of robot, and for this communication between robot and controller we use WiFi instead of antenna or Bluetooth which limits the functionality of robot. So, this robot can be used in the dangerous situation where human life is at risk without special training to operate it .

1.3 The Solution to a Real World Problem

In our daily life, we have seen countless examples of how machines are making our life simpler and easier. We have seen miners working in dangerous unstable mines. Their work also leads to exposure of different chemicals which cause health problems. Use of machines which can be operated by your gestures can save many lives. Other real life usage of such machines can be in medical surgeries, firefighting, underwater and outer space explorations.

1.4 Existing Example

There are some examples of this type of project already existing but we are using flex sensor to activate the robot, which is not commonly used and we have been using wireless communication over WiFi instead of Bluetooth and antenna.

1.5 Personal Skills

The main competencies required to bring this project to its full functionality involves both hardware and software fields. The hardware skills involve integration of sensors with microprocessor, coupling of motor driver with RPi along with camera installation.

Second competency which is the most important is in the software field. To accomplish the task of making the raspberry Pi to collect the UDP packets sent from transmitter and movement of vehicle, enhanced C #,python and Linux skills were required for writing code and using of right distribution of Raspberry pi. Apart from hardware and software competency team working, time management, problem solving skills and above of all patience was also required to full accomplish the project.

2 Goals Division

2.1 Mandatory Goals

To successfully complete our project, following were the mandatory goals
◦Activation of Vehicle by a hand gesture (using Flex sensor)
◦Movement of Vehicle and taking turns by different hand gestures

2.2 Target Goals

The goals at our discretion to add in our project were
◦Real time video streaming from the remote location.

2.3 Discretionary Goals

◦Calculating distance traveled by the vehicle.
◦Temperature and humidity sensing.

2.4 Goals Achieved

Following goals we have achieved in this project
◦Vehicle activation by using flex sensor
◦Movement of vehicle and changing directions by different hand gestures
◦Installed PiCam for real time video streaming

3 User Friendliness & Usability

3.1 Design and Characteristics

The main components of this project are accelerometer and flex sensor along with raspberry pi module. Accelerometer and Flex sensor based gesture controlled vehicle moves according to the movement of hand as we place the accelerometer and flex sensor on our hand. When we fold our palm, we'll be bending flex sensor, which is actually a variable resistor, this move will activate the vehicle and the activation LED will glow which is integrated on robotic vehicle and when we tilt our hand forward it will make robot to move forward. When we tilt hand in backward direction, then the robot changes its direction and state. Then it starts moving in backward direction until we change the hand direction.
Our hand movement is detected by the accelerometer which is connected to micro-controller on our hand. This transmitting device send signals via WiFi network to other micro-controller mounted on vehicle which acts as receiver. And for real time video streaming we can use any smart device to watch the streaming using the IP address of the RPI along with port no.

3.2 Problems in Long Time

The transmitter and receiver are connected via WiFi so in case of drop of signals or lack of network, there is a possibility of loss of connection. As it is powered by a battery, which can lead to limitation of its scope.

3.3 Solution

Lack of network or WiFi issues can be overcome by providing easy and open internet access throughout the region or area.

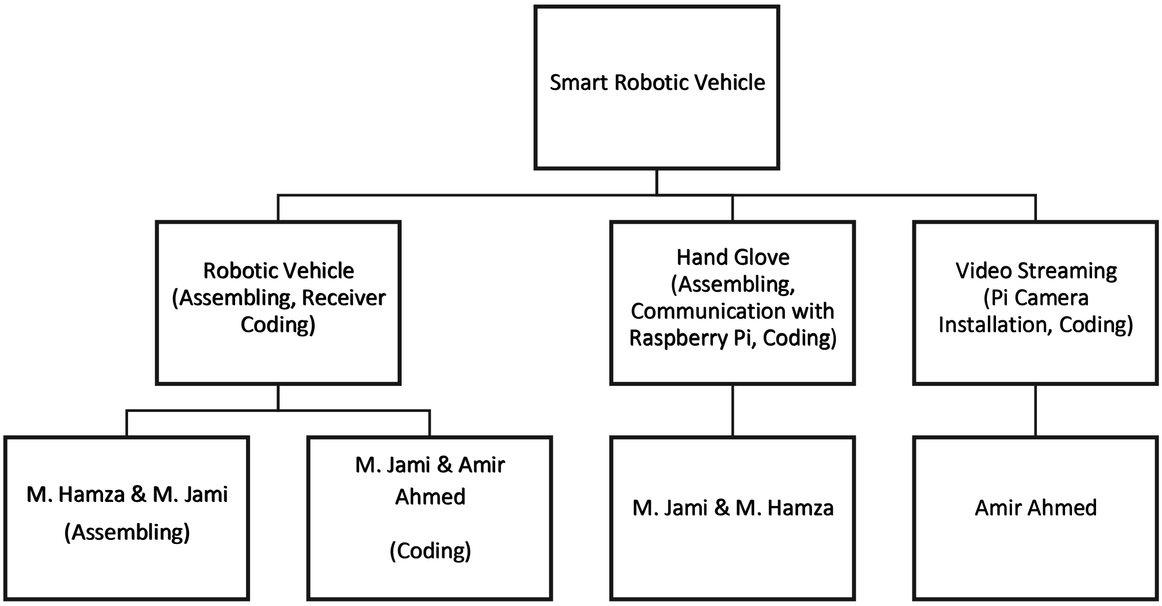
Long lasting battery and fast charging power banks can be used to increase its productivity.

4 Challenges

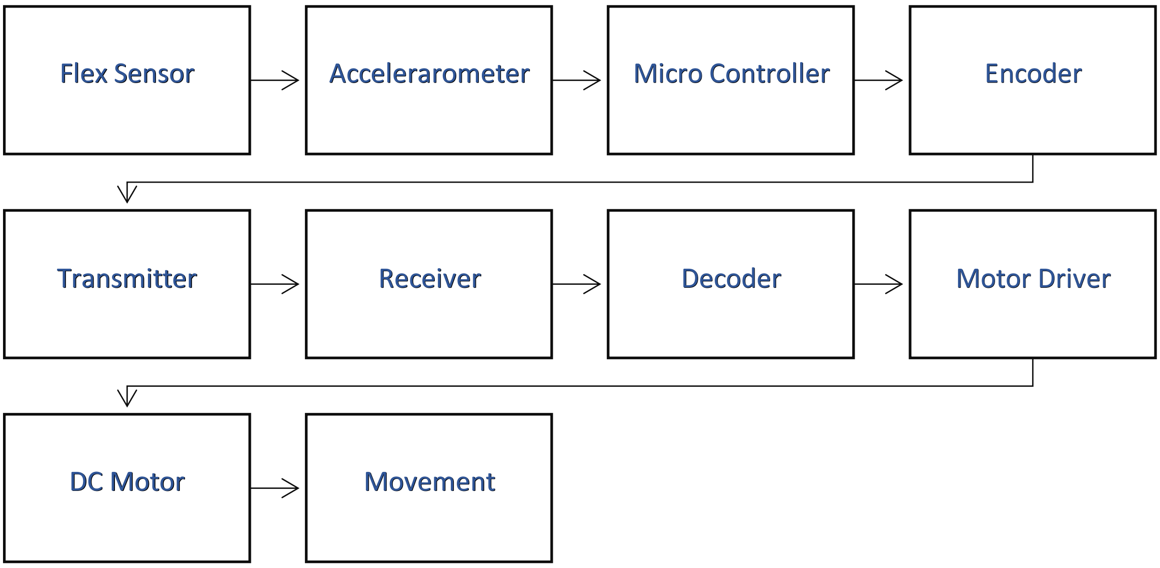
- The major challenges we faced during the project were:
- Learning python and C# was the main challenge for completion of this project.
- Integration of sensors
- To make communication between transmitter and receiver
- Knowledge of Raspberry pi and Arduino.
- Use of motor driver.
- Python troubleshooting techniques.
- Use of Camera for video streaming
- Finding good quality low cost components.
- Time Management.
- Team Working.
- Individual task assigning.

5 Project Structure Plan

5.1 Distribution of Project and Tasks



5.2 Sub-Areas Division of Hardware



5.2.1 Transmitter

The Transmitter on the glove has following components

5.2.1.1 Arduino with WiFi Shield

Arduino (MKR1000) will receive the output of the flex sensor and accelerometer, process it and send it to our Gesture Controlled Vehicle as udp packets.

5.2.1.2 Flex Sensor

Flex Sensor (2.2 inch) is a variable resistor which changes its electrical properties when it is bent. There are two output pins – one is for ground and the other is for the analog output. When the sensor is flat, it gives the normal resistance and when it bends towards level to 45 degree resistance increases and when it brand towards 90°, resistance increases. When it bends towards the membrane, resistance decreases.

5.2.1.3 Accelerometer

An accelerometer (MMA7361) is a sensor which can be used to sense the acceleration due to gravity on various axis acting on an object. Using this we can easily calculate the tilt of an object with respect to the Ground. We controlled our vehicle using the X and Y tilt analogue values.

5.2.2 Receiver

The Receiver on the vehicle has following components

5.2.2.1 Raspberry Pi

It is a low cost CPU which can be used as a general purpose PC for web surfing, video streaming etc. Due to its small size, it can be carried anywhere and can be powered using a mobile phone power bank. Raspberry pi has Several GPIO pins which will enable us to interact with outside world using various sensors which can be plugged into it. We used a motor driver to drive DC Motors in the our project because we cannot drive a motor directly using GPIO pin output of microcontroller as it doesn't have the power to run a motor.

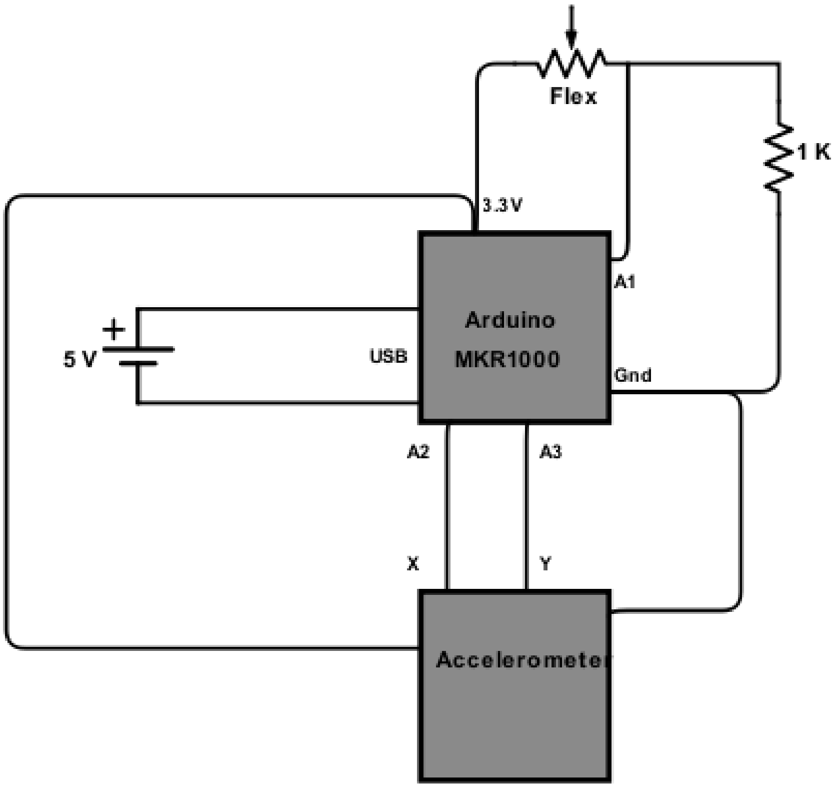
5.2.2.2 Motor Driver

Motor receives power from the Motor driver IC (L293d) which is being control by the RPi to rotate the motor in both directions.

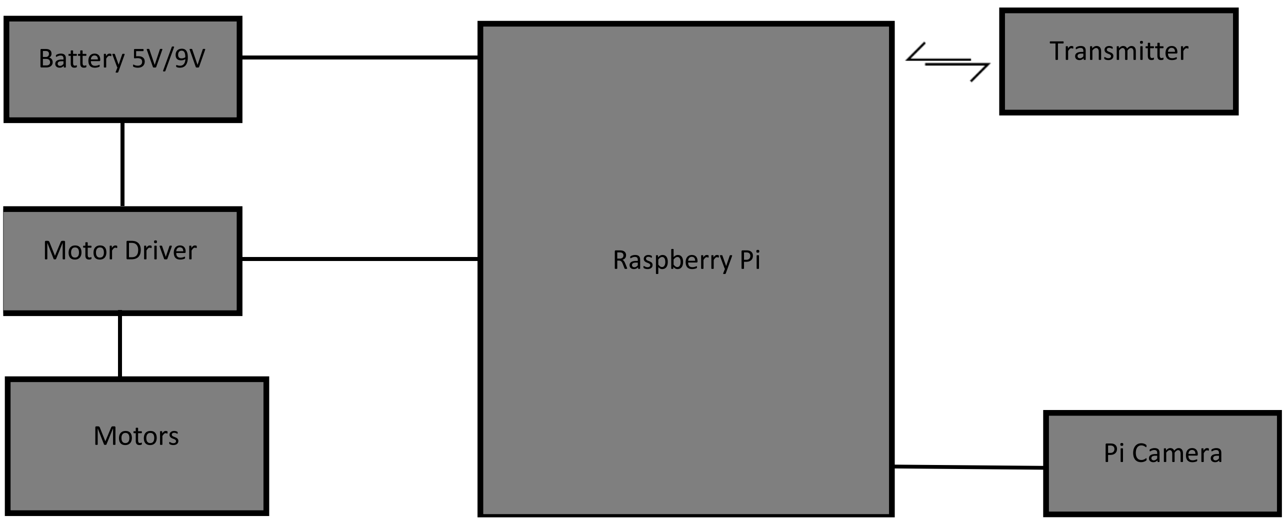
5.2.2.3 Motors

Two 9v DC motors are used controlled by a motor driver for the movement of vehicle. DC motors change its rotation depending on the polarity of its terminals. Motor Driver used to drive motors has H-bridge, so it can change its rotation. We have used L293D motor driver which have two H-bridges so it can control two motors.

5.3 Block Diagram of Transmitting Part



5.4 Block Diagram of Receiver Part



GPIO pins are connected as follow

- GPIO 11 to Motor A input 1
- GPIO 33 to Motor A input 2
- GPIO 15 to Motor B input 1
- GPIO 13 to Motor B input 2

6 Code

6.1 Main Processor

The code used for the movement of vehicle is given below. The Raspberry Pi module receives an input through glove and on the basis of input, it turns on the led showing the data is received. As per the movement of hand, it moves the vehicle in the assigned direction.

6.1.1 Code

```
import RPi.GPIO as GPIO

import socket

import csv

GPIO.setwarnings(False)

GPIO.setmode(GPIO.BOARD)

GPIO.setup(33,GPIO.OUT)

GPIO.setup(11,GPIO.OUT)

GPIO.setup(13,GPIO.OUT)

GPIO.setup(15,GPIO.OUT)

GPIO.setup(29,GPIO.OUT)

GPIO.setup(31,GPIO.OUT)

GPIO.output(31,True)

UDP_IP = "0.0.0.0"

UDP_PORT = 6661

sock = socket.socket(socket.AF_INET, # Internet

socket.SOCK_DGRAM) # UDP

sock.bind((UDP_IP, UDP_PORT))

while True:

data, addr = sock.recvfrom(1024)
```

```
raw=data
print "Received Data: ", raw
status, action = data.split(',')
print "Status: ", status
print "Action: ", action
if status=="1":
print "LED ON"
GPIO.output(31,True)
else:
print "LED OFF"
GPIO.output(31,False)
if action==" F":
GPIO.output(33,True)
GPIO.output(11,False)
GPIO.output(13,True)
GPIO.output(15,False)
print "Robot Move Forward"
elif action==" 0":
GPIO.output(33,False)
GPIO.output(11,False)
GPIO.output(13,False)
GPIO.output(15,False)
print "Robot Stop"
elif action==" B":
GPIO.output(33,False)
GPIO.output(11,True)
GPIO.output(13,False)
GPIO.output(15,True)
print "Robot Move Backward"
elif action==" L":
GPIO.output(33,False)
```

```
GPIO.output(11,True)
GPIO.output(13,True)
GPIO.output(15,False)
print "Robot Move Left"
elif action==" R":
GPIO.output(33,True)
GPIO.output(11,False)
GPIO.output(13,False)
GPIO.output(15,True)
print "Robot Move Right"
else:
print "STOP"
GPIO.output(33,False)
GPIO.output(11,False)
GPIO.output(13,False)
GPIO.output(15,False)
```

6.2 Glove

This code has been used to set up our Arduino mounted on the glove, it gets an operating voltage from a battery and input from flex sensor to get activated and then from accelerator to send an upd packet via WiFi to our Raspberry Pi module mounted on the vehicle.

6.2.1 Code

```
#include <SPI.h>
#include <WiFi101.h>
#include <WiFiUdp.h>
int ledPin = 6;
int s2,s3,s4;
int status = WL_IDLE_STATUS;
char ssid[] = "RootSaid"; //WiFi Network Name
char pass[] = "www.rootsaid.com"; //WiFi Network Password
int keyIndex = 0;
unsigned int localPort = 6661;
```

```
char packetBuffer[255];
char ReplyBuffer[] = "OFF";
WiFiUDP Udp;
void setup() {
pinMode (A1, INPUT);
pinMode (A2, INPUT);
pinMode (A3, INPUT);
Serial.begin(9600);
if (WiFi.status() == WL_NO_SHIELD)
{
Serial.println("WiFi shield not present");
while (true);
}
// Connect to WiFi
while ( status != WL_CONNECTED)
{
Serial.print("Attempting to connect to SSID: ");
Serial.println(ssid);
status = WiFi.begin(ssid, pass);
delay(10000);
}
Serial.println("WiFi Status ");
printWiFiStatus();
Serial.println("\n Starting Listener");
Udp.begin(localPort);
}
void loop() {
readAnalog();
process();
IPAddress ip(192, 168, 1, 3);
Udp.beginPacket(ip, localPort);
```

```
Udp.write(ReplyBuffer);
Udp.endPacket();
//Serial.println("Data Send");
delay(200);
}
void printWiFiStatus() {
Serial.print("SSID: ");
Serial.println(WiFi.SSID());
IPAddress ip = WiFi.localIP();
Serial.print("IP Address: ");
Serial.println(ip);
long rssi = WiFi.RSSI();
Serial.print("signal strength (RSSI):");
Serial.print(rssi);
Serial.println(" dBm");
}
void readAnalog()
{
s2 = analogRead(A1);
s3 = analogRead(A2);
s4 = analogRead(A3);
Serial.print(s2);
Serial.print(" -- ");
Serial.print(s3);
Serial.print(" -- ");
Serial.print(s4);
Serial.println(" ");
}
void process()
{
if ((s2<45))
```

```
{
if ((s3>600))
{
if((s4<600)&&(s4>400)){
Serial.println("1, F");
strcpy(ReplyBuffer, "1, F");
}
else if((s4>=600)){
Serial.println("1 FL");
strcpy(ReplyBuffer, "1, FL");
}
else if((s4<=400)){
Serial.println("1, FR");
strcpy(ReplyBuffer, "1, FR");
}
}
if ((s3<=600) && (s3>=450))
{
if((s4<600)&&(s4>400)){
Serial.println("1, 0");
strcpy(ReplyBuffer, "1, 0");
}
else if((s4>=600)){
Serial.println("1, L");
strcpy(ReplyBuffer, "1, L");
}
else if((s4<=400)){
Serial.println("1, R");
strcpy(ReplyBuffer, "1, R");
}
}
```

```
if ((s3<450))
{
if((s4<600)&&(s4>400)){
Serial.println("1, B");
strcpy(ReplyBuffer, "1, B");
}
else if((s4>=600)){
Serial.println("1, BL");
strcpy(ReplyBuffer, "1, BL");
}
else if((s4<=400)){
Serial.println("1, BR");
strcpy(ReplyBuffer, "1, BR");
}
}
}
else if((s2>=45))
{
strcpy(ReplyBuffer, "0, 0");
Serial.println("0, 0");
}
else
{
strcpy(ReplyBuffer, "0, 0");
Serial.println("0, 0");
}
 }
```

6.3 Video Streaming

This part explains the steps followed to add camera to our gesture controlled vehicle.

6.3.1 Set Up Hardware and Configuration

Connect camera module to Raspberry Pi and enables it

sudo raspi-config

To check whether Raspberry pi is detecting the camera module or not. Go to /dev/ and see whether ‘videox’ is present or not.

If it is not there, run

sudo raspi-update

Then restart the pi and run

sudo modprobe bcm2835-v4l2

Now to install motion by running the command

sudo apt install motion

Open up the configuration file in /etc/motion/motion.conf using a text editor

sudo nano /etc/motion/motion.conf

Edit the following lines.

daemon on

stream_localhost off

stream_maxrate 50

framerate 50

width 640

height 480

Now open the daemon file

sudo nano /etc/default/motion

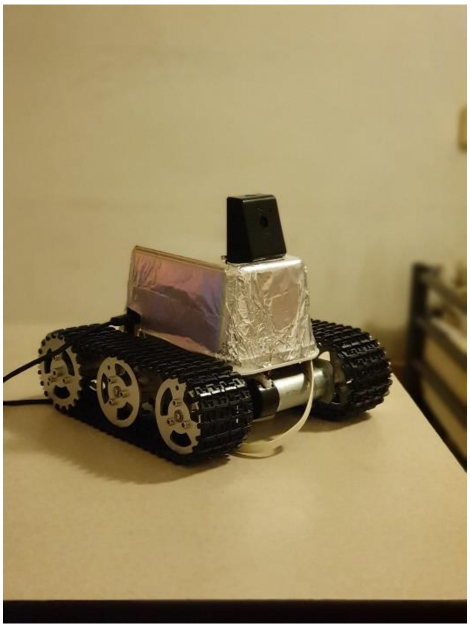
start_motion_daemon=yes

Now start the daemon by running the command

sudo service motion start

Now open up a web browser on any device on that network and enter the IP address of the pi with the port number 8081. Like http://192.168.1.58:8081

7 Final Project



8 References

- https://www.electroschematics.com/learn-use-mma7361/
- https://www.ardumotive.com/how-to-use-a-flex-sensor-en.html
- https://www.arduino.cc/en/Reference/EthernetUDPWrite
- https://www.instructables.com/id/How-to-use-the-L293D-Motor-Driver-Arduino-Tutorial/
- https://pythonprogramming.net/introduction-raspberry-pi-tutorials/