SindhUniv. Res. Jour. (Sci. Ser.) Vol. 51 (01) 119-128 (2019)

http://doi.org/10.26692/sujo/2019.01.22

SINDH UNIVERSITY RESEARCH JOURNAL (SCIENCE SERIES)

Fuzzy Logic Based Obstacle Avoidance Autonomous Robots

I. MALA, S. S. ZIA[++]*, M. NASEEM*, T. MOBEEN*, T. J. A. MUGHAL**

Electrical Engineering Department, Usman Institute of Technology, Karachi.

Received 07th September 2018 and Revised 11th February 2019

Abstract: The research shows benefits that fuzzy logic has brought to the performance of Autonomous mobile robot and the possible logic strategy for the performance improvement of motion control. The collision avoidance mobile robots have become a major area of research in robotics that involves various degrees of uncertainty. The developed fuzzy logic let the Autonomous mobile robot to have an ability to avoid the static obstacles coming in its path without the involvement of any human being. Fuzzy logic approach has been proved to be a simple and a more easy technique for such kind of robots.

Keywords: Fuzzy Logic, Robots, Rasberry pi, Fuzzy Inference System, Ultra Sonic Sensor

I. INTRODUCTION

Projects are all about learning and practicing the things which we have learned. It is the practical way of analyzing the things and reach to the conclusion. Now, when we talk about a final year project it is actually a chance for students to demonstrate all they have learned. Therefore we have decided to select a project based on **Robotics** as it has always been a part of research and as far as mobile robots are concerned, it is an important part of research around the late sixties. The use of mobile robots is slowly growing in number as it is used in many applications (Amruta *et al.,* 2017).

Mobile robots are the mechanical devices which has the ability to move in an environment without any human involvement, they are capable of taking all decisions by themselves (Hachour, 2008). Today's Robotics has made its most prominent progress to date in the realm of modern assembling. Robots have extraordinary criticalness in the present time and can understand a wide range of errands without the human mediation. Conversely with this, a self-governing robot would have the capacity to go in the controlled element condition and can bring choices without anyone else with the assistance of sensors (Gurjashan *et al.,* 2015).

There are numerous movement instruments accessible yet the wheel versatile robots have been by a wide margin the most well-known headway component in portable apply autonomy and in man-made vehicles. It can accomplish great efficiencies with a generally straightforward mechanical component.

The objective of our venture is to plan an autonomous robot which has a capability to move in an unpredictable and in dynamic environments, and for that reason robot has to sense the environment and carry out the required tasks regardless of any obstacles in their working area (Henry, 2016). Several different approaches are criticized to perform such operability but the Fuzzy logic found to be the best for this kind of robotics operations because it has the ability to respond effectively to the complex inputs. Fuzzy logic is well suitable for the low cost implementations based on cheap sensors (Sarmasti, 2010). Many researchers used Fuzzy control approach in sensor based navigational control of autonomous robots.

It will be operated without the human mediation and can take decision by itself using Fuzzy Logic and to maintain a strategic distance from obstruction that can be static. The client set the underlying position to the robot in this way, by overlooking the deterrents and take choice as indicated by its position and condition and made either forward, left, Right, Forward left, Forward right, Or Backward in the most pessimistic scenario.

The robot will be furnished with running sensors (ultrasonic HC-SR04) (Venkatashan and Sughanthi 2015) which will detect the scope of hindrances on the premise of how close or far the impediment is. For this quick and continuous preparing a solitary board PC will be utilized to be specific Raspberry Pi model2B which takes the readings from the sensors and maintain a strategic distance from the snags set in their way lastly proceeds onward.

[++]Corresponding Author: Syed Saood Zia, saood_zia@hotmail.com,
*Software Engineering Department, Sir Syed University of Engineering and Technology, Karachi
**Biomedical Engineering Department, HIIT, Hamdard University, Karachi.

At what parameter an autonomous robot take decision to keep away from barriers of their way? How it apprehend the barriers and what algorithms may be implemented. How are we able to provide self-sustaining robots the capability to solve issues in a way this is as "natural" for the robotic because it appears to be for us as people? and, why now not broaden these solutions by using a robots to real-global problems that can help to make our complex human lives a bit easier, this is a first rate motivation for undertaking in fuzzy logic and self-reliant robotics, and what makes this paintings very gratifying. Fuzzy good judgment is an exceptional tool to implement or to acquire the proposed requirement of mission which is to avoid impediment in their manner via taking self-decisions in a controlled surroundings (Carlo D'Ortenzio, 2012).

2. METHODOLOGY

Robotics is a vital place of studies that uses know-how throughout several disciplines together with mechanics, electronics, and laptop engineering, to move an autonomous robot in defined surroundings with a few diploma of autonomy, one of the important demanding situations of the self-sufficient navigation for autonomous robots is the detection and boundaries avoidance during the robot navigation assignment http://engineering.nyu.edu/mechatronics/smart/pdf/Intro 2Robotics.pdf This problem can be solved by way of referring to unique methods or algorithms to be able to reap excellent effects. Many research literatures used gentle computer algorithms to govern autonomous robots in academic area as well as inside the engineering area.

This Unmanned robotic vehicle is capable of performing the desired tasks in unstructured, uncertain and potentially hostile conditions with the principle of avoiding the obstacles (Hamami, *et al.,* 2016). The controller which is used in our task is Raspberry PI 2 model B has many advantages and flexibility. It works like a mini computer and has fastest overall performance associated with other controllers. Robot car is designed that allows you to operate without the human intervention and may take selections with which it will likely be programmed. The tool which we have used to assist robot is fuzzy logic. By this logic robot can take choices to avoid limitations coming of their course and can flow forward, left, right and reverse within the surroundings fuzzy logic might be carried out the use of a python language.

2.1. Project key features

1. Use of Raspberry pi2 model B
2. Implementation of Fuzzy logic
3. Python Programming
4. Matlab
5. Robot Car
6. Obstacle Detection (sensors)
7. Obstacle Avoidance Algorithms

2.2. System Design

Basically, the goal of our project consists of three phases:

- Intelligence
- Perception
- Action

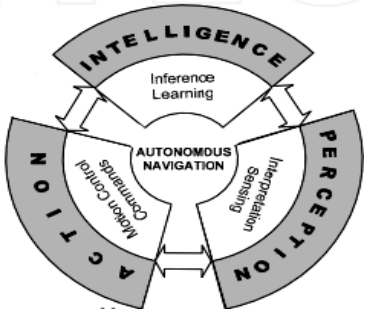


Fig. 2.1 Continuous Loop Diagram

2.2.1. Intelligence

In this segment robot have to be drafted intelligently in this sort of way that it takes choice with the aid of itself and have to understand the environment through using programming talents and equipment that are fed on this gadget to perform in a dynamic surrounding. Avoid limitations which had been located in its route by using its intelligence and revel in positive things then getting know-how approximately the barriers that where and in which role the obstacle be positioned.

2.2.2. Perception

At this stage sensors which are connected with robot go about as data sources takes perusing from the surrounding about the obstructions that how far, close, left or right be set in the way. Measure the separation from robot to the snag after then takes the deliberate perusing to the raspberry pi. Raspberry pi coordinates that perusing and takes choice as indicated by it and the sends the signs to robot.

2.2.3. Action

Here, at this level the intelligence and perception both phases makes up the activation section. With the assist of intelligence, sensing measurement and commands ship through the raspberry pi controls the entire robot automobile. Robot is ready to move on the course by itself without the human steering and remote devices. Now, the robotic is capable of heading off boundaries with the convenience of sensors and the fuzzy logic algorithms.

These three above stages whilst working collectively making up the autonomous navigation system and with assist of those our robot car navigate in the dynamic surroundings and keep away from the impediment coming in its course.

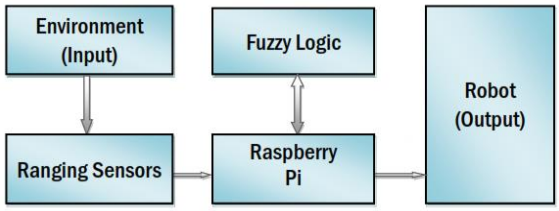


Fig. 2.2 Input to Output flow diagram

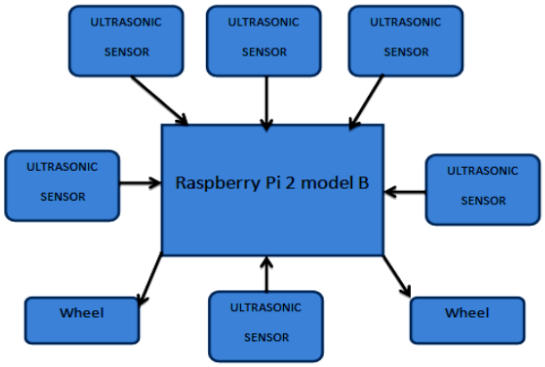


Fig. 2.3 Block diagram of Hardware architecture

2.3. Block Diagrams

In the previous figure, there are six ultrasonic sensors associated with the left, right, top and base of the robot and it additionally comprises of a scaled down PC referred to as raspberry pi which go about as an equipment in this venture likewise administers the whole robot by sending signals get by the sensors. It comprises of wheels at the corners to work the wheels there is IC L298N which go about as a controller for the motors.

2.3.1. Raspberry Pi

Will utilize mixes of GPIO pins to control four engines. The additional layer of multifaceted nature will be an additional H- bridge, which will permit us to use forward and turn around with our motors. With four motors fit for forward and invert, we can make an auto with settled wheels go ahead, switch, turn left, turn right, or rotate set up clockwise/counter clockwise. From that point onward, will acquire the HC-SR04 separate sensor, perusing values from that into our program. We can utilize this separation sensor to move our auto into self-rule. In the event that we had two, we could have one on the front, one at the back, and our auto would (nearly) never hit anything. Indeed, even only one on the front is genuinely effective, however we hit the back end off and on again.

2.3.2. HC-SR04

The distance sensor works by shooting ultrasonic waves, ascertaining the measure of time between sending the signals and getting it. We can utilize this time, and our insight into the speed of sound consistent to ascertain remove. It ought to noted here that the separation sensor functions admirably even on a few inclinations, if the protest that you're bobbing sound off of is more than a 33 degree point, your outcomes are probably going to be exceptionally erroneous.

2.3.3. Motors (Wheels)

IC L298 is utilized as a controller to work the engines of either side of the auto robot. Motors are utilized for the wheels with the goal that it can moves forward, backward, left or right. L298 IC fills in as a controller since it shields from the enough current streams over the wheels.

2.4. Fuzzy Inference System

A fuzzy inference system (FIS) is a system in which we uses the theory of fuzzy set to plot the given set of input to outputs which is the classes in the scenario of the fuzzy classification (Riza *et al.*, 2015) or in other words we can say that the Fuzzy inference is the procedure of formulating the mapping from a given input to an output by means of the fuzzy logic.

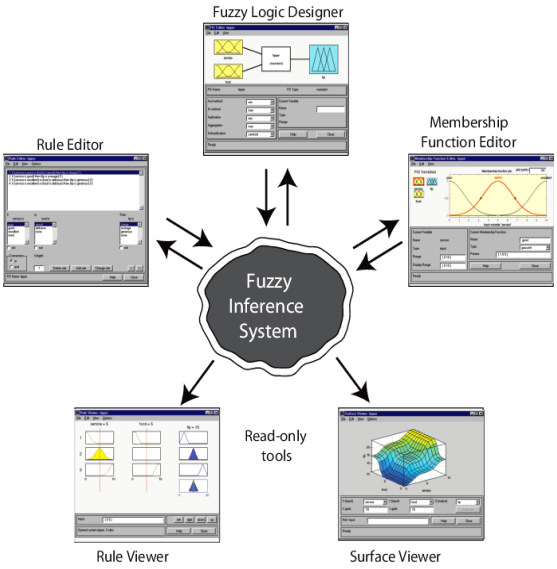


Fig. 2.4 Fuzzy Inference Systems (MATLAB)

The decisions can be made from the basis provided by mapping. The procedure of the fuzzy inference contains all of the bits that are described in the Membership Functions, Logical Operations, and some time If-Then Rules. The main types that we can implement on the Matlab are the Mamdani and the Sugeno.

The input variables are normally mapped in a fuzzy control system by the set of membership function namely "fuzzy sets" and “fuzzification” is termed as the process of transformation of a crisp value to a fuzzy value.

The steps that are involved in the process to compute the output of the given FIS inputs:
1. Determining a set of fuzzy rules.
2. Fuzzifying the inputs using the input membership functions.
3. The rules are evaluated by means of fuzzy reasoning.
4. Defuzzifying the output distribution when Crisp output is needed.

2.5. Flow Chart

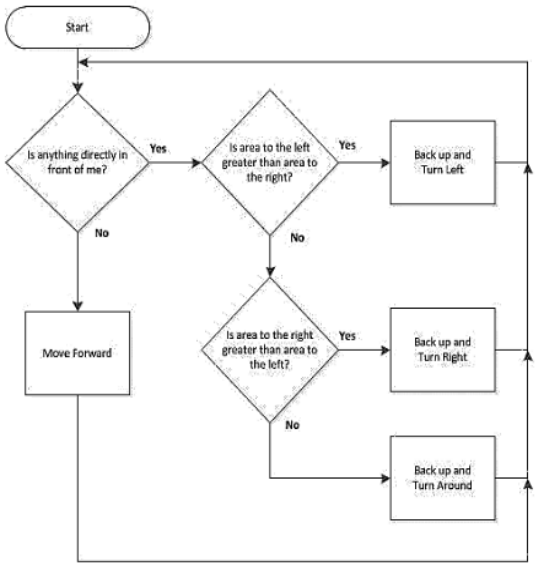


Fig. 2.5 Obstacle detection and avoidance flow chart

Firstly, the robot will begin to move straight on the way a short time later it gets the signs from the raspberry pi whether to push ahead, turn around, right or left. Raspberry pi gets the signs from the sensors then the sensors as indicated by their scope of detecting separation take choice and check is there any hindrance be put? At what distance? In which course it is be set? Sensors are in charge of measuring the separation effectively and the sends the deliberate readings to the raspberry pi. There are 24 GPIO sticks in raspberry pi from which 4 pins are associated with the auto robot. AT getting the signs from the raspberry pi these pins are actuated and the robot is prepared then it takes the choice by the assistance of calculations, summons and hardware's which are encouraged into it. On the off chance that any impediment is recognized on its route whether at a course then promptly it must be maintain a strategic distance from by the robot auto and it ought to be moves in an inverse heading without touching the deterrent and afterward subsequent to going for some time straight and stops.

3. IMPLEMENTATION

3.1. Hardware

- Raspberry Pi 2 model B.
- HC-SR04.
- Robot Car.
- L298N.
- Other peripherals.

3.1.1. Raspberry Pi 2, Model B

The Raspberry Pi is based on quad-core processor that runs at 900 MHz and has memory capacity of upto 1 GB. It is upgraded Broadcom BCM2836 processor, which is a powerful ARM Cortex-A7.

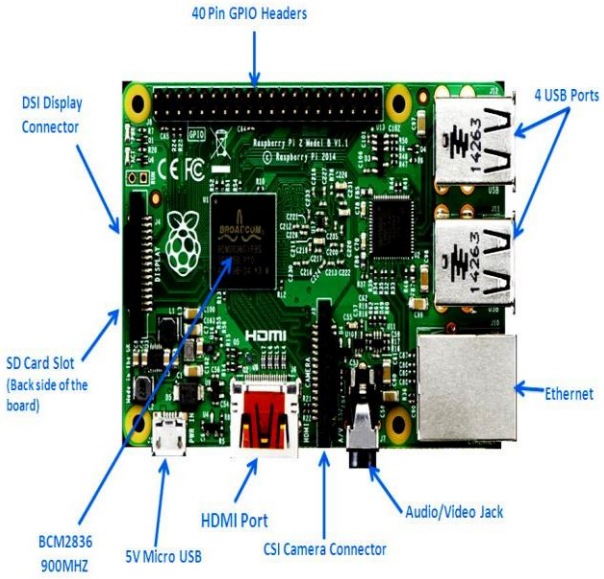


Fig. 2.6 Raspberry Pi 2, model B

Specifications:

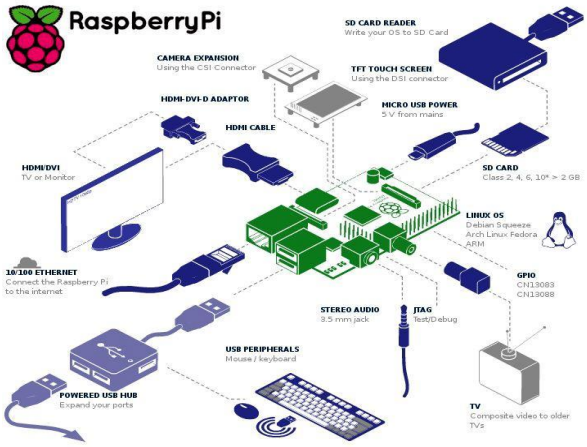


Fig. 3.2Raspberry pi connectors

Chip	Broadcom BCM 2836 SoC
Core Architecture	Quad-core ARM Cortex-A7
CPU	900 MHz
GPU	Dual Core VideoCore IV® Multimedia Co-Processor Provides Open GL ES 2.0, hardware-accelerated OpenVG, and 1080p30 H.264 high-profile decode Capable of 1Gpixel/s, 1.5Gtexel/s or 24GFLOPs with texture filtering and DMA infrastructure
Memory	1GB LPDDR2
Operating System	Boots from Micro SD card, running a version of the Linux operating system
Dimensions	85 x 56 x 17mm
Power	Micro USB socket 5V, 2A

Raspberry Pi Pin out Description:

- GPIO are your standard pins that simply be used to turn devices on and off, for example a LED.
- I2C (Inter-Integrated Circuit) pins allow you to connect and talk to hardware modules that support this protocol (I2C Protocol). This will typically take up 2 pins.
- SPI (Serial Peripheral Interface Bus) pins can be used to connect and talk to SPI devices. Pretty much the same as I2C but makes use of a different protocol.
- UART (Universal asynchronous receiver/transmitter) are the serial pins used to communicate with other devices.
- DNC stands for do not connect, this is pretty self-explanatory.
- The power pins pull power directly from the Raspberry Pi.
- GND are the pins you use to ground your devices. It doesn't matter which pin you use as they are all connected to the same line.

3.1.2. Ultra Sonic Sensor (HCSR-04)

This is the HC-SR04 ultrasonic ranging sensor. Each HC-SR04 module includes an ultrasonic transmitter, a receiver and a manipulate circuit. There are only four pins that you need to worry about on the HC-SR04: VCC (Power), Trig (Trigger), Echo (Receive), and GND (Ground).

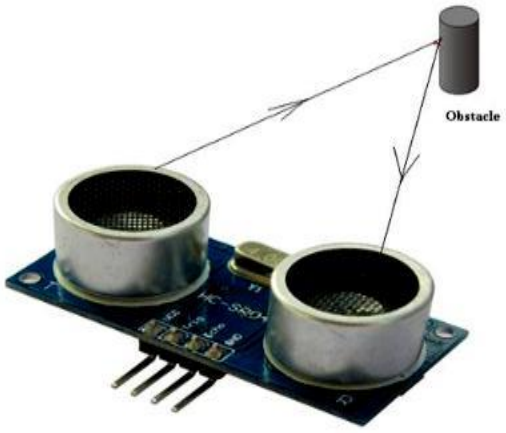


Fig. 3.4 Ultrasonic Sensor HCSR-04

How Ultra Sonic Sensor Works:

Ultrasonic sensors use sound to decide the distance between the sensor and the closest object in its path (Vidhya *et al.*, 2016). Ultrasonic sensors are basically sound sensors; however they operate at a frequency above human listening to. The sensor sends out a legitimate wave at a selected frequency. It then listens for that particular sound wave to bounce off of an object and come again. The sensor continues tune of the time among sending the sound wave and the sound wave returning.

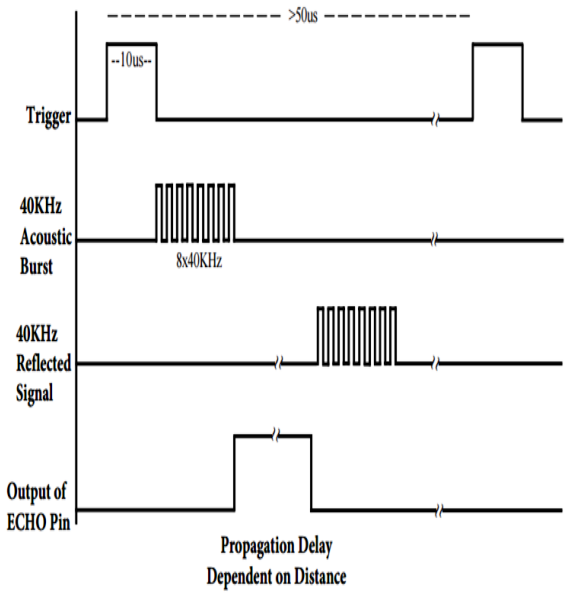


Fig. 3.5 Ultrasonic sensor HCSR-04 module timing diagram

The uses of Ultrasonic Sensor are in Security System, Level Detection, Robotics Barrier, Object distance measurement and Vehicle Detection Avoidance.

3.1.3. Robot Car

Robotic cars have a lot impacts on our society and to advancement in technology. This is an autonomous car robot. The car robot is multi-functional. The car would know exactly where to go and would be built lighter. The will be capable of moving independently in all dimensions.

3.1.4. L298N Motor Driver Module

It is a dual full-H driver with the ability to attach a large heat sink and pull 1.5A. It can handle more current and voltage.

- 46V.
- Low saturation voltage.
- 4A.
- Over temperature protection.

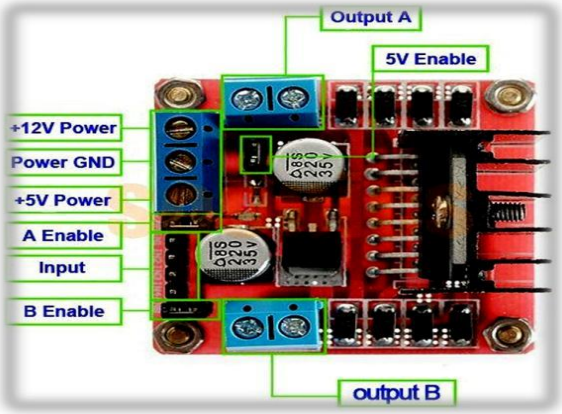


Fig. 3.6 L298N Motor Driver module

3.2. Software

- Raspbian (Operating System)
- Python (Programming Language)
- MATLAB (Fuzzy logic implementation)

3.2.1. Raspbian

Raspberry Pi supports different operating systems; following is the list of some:

- Noobs.
- Raspbian.
- Openelec & Xmbc.
- Risc OS.
- Retropie.
- Firefox OS.
- Ubantu Mate.
- Pidora.
- Linutop.
- Sarpi.
- Arch Linux ARM

The official operating system of the Raspberry Pi is Raspbian therefore most of the people start with it. Raspbian is a version of Linux which is specifically built for the Raspberry Pi. It is a free Debian-based operating system which is optimized for Raspberry Pi's hardware; it is packed with all the software that can be needed for every basic task with a computer.

Raspbian allows us to select a variety of images during the boot-up .The simplest way of installing the Raspbian on our Pi is by deploying its image ile into an SD card. This Operating system is supported for more than 35,000 packages and gives us the fast performance.

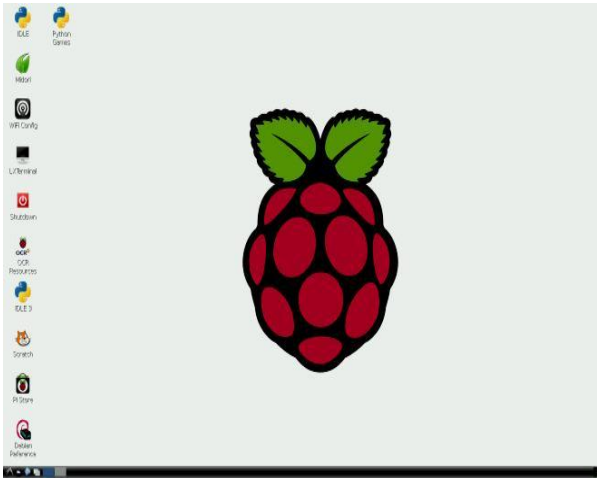


Fig. 3.7 Raspbian desktop environment

3.2.2. Python

Raspberry pi is supported by many different programming languages, but by the creator of language who wanted to support the PI by porting their creation or it depend on the user which programming language he likes to choose. Raspberry pi can be implemented by using the following languages:

- Python.
- Java.
- C, C++.
- Perl.
- Erlang.

But the programming language which we will implement is the Python.

Raspbian comes preloaded with **Python**, the official programming language of the Raspberry Pi and IDLE 3, a **Python Integrated Development Environment (**Ram and Gorisankar, (2016). Python is one of the primary programming languages hosted in Raspberry PI. Python is a wonderful Python is a

wonderful and powerful programming language that's easy to use (easy to read **and** write) and with Raspberry Pi lets you connect your project to the real world.

Advantages of Python:

Python is a very simple and easy programming language even a new developer can have its expertise within a short period of time. Its code is very simple to read and understand its logic and you can do a lot of things just by a quick overview of it. Also, a lot of complex functionalities can be executed easily.

- It supports multi systems and platforms.
- Object Oriented Programming is driven.
- Quick development of code by using less number of lines.
- A small team can handle Python effectively and can produce an accurate result.
- Can be used for complex application by giving quick and easy result.
- Ability of having a large number of resources.

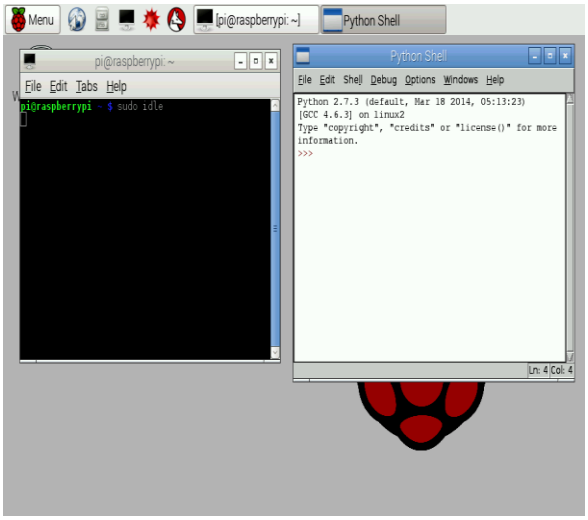


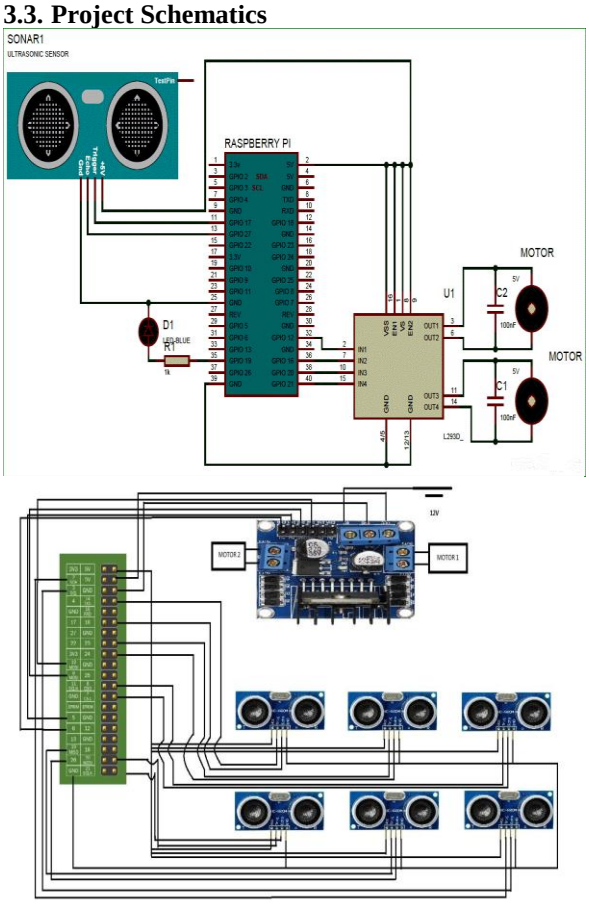
Fig. 3.8 Python Shell

3.2.3. Matlab

MATLAB can be used for different purposes but here in our project we are using MATLAB in order to simulate and analyze the Fuzzy logic systems. The toolbox lets us to see the complex system behaviors by implementing the easy logic rules, and then implements these rules in a fuzzy inference system.

MATLAB allows us to test the algorithms without any delay and without recompilation. The MATLAB Desktop environment, which allows us to work interactively with our data, helps us to keep track of files and variables, and simplifies common programming and debugging tasks.

3.3. Project Schematics



3.4. Problems

- Motor driver module selection
- Sensor Accuracy/Throughput
- Sensor Data interpretation
- Setting fuzzy membership functions and rules
- Auto-executable python script

3.5. Troubleshooting

i. While operating the Robot motors, initially motors were connected directly through the Raspberry Pi GPIO, therefore the GPIO of raspberry pi can't operate the motors as it draws just 3.3V which doesn't fulfill the requirement for driving motors. So, for that issue a motor driver module is required to drive the motors. L298N was observed as a best option for solving this problem. Raspberry Pi output was first connected to the L298N module through which the motors are connected and ultimately driven.

ii. Sensor precision and throughput is ending up noticeably less a risk to fruitful impact shirking. Be that as it may, this doesn't mean it is non-existent. Amid the way toward designing a crash maintaining a strategic distance from robot, exchange offs can be made that

could conceivably trade off the viability of the sensors in how the robot carries on.

iii. For sensor information understanding past coding can be trailed by changing the impediment separate (utilize scale) and note that readings for the information gathering.

iv. Firstly, create and draw membership functions and rules on a rough paper then implement it on a fuzzy interference system so that rules can't be blend with one another.

v. To make python script auto-executable, following steps were taken:

- Enter into Terminal mode.
- Type sudo nano /etc/profile.
- Edit the file by typing the path of the python script (sudo python "Enter the script path"). Then save the file.
- Go to the configuration mode of Raspberry Pi by typing sudo raspi-config.
- Select the boot option as Console Auto-login.

4. RESULTS AND DISCUSSION

4.1. Test Analysis and Design

Before amalgamating all the essential components into a furnished robot, following considerations were taken:

- While exploring Raspberry Pi ensures that Operating system correctly chosen, read the documentation of Raspberry Pi how to boot it and password logging procedure.
- Test the sensor's trigger, echo pins voltage before use
- Choose suitable motor driver module.
- While, constructing the circuit test all the sensors, Raspberry Pi Vcc and ground pins and recheck on errors in programming then safely adjust all the gadgets which was embedded in robot.
- Initially, create and draw membership functions and rules on a paper then implement it on a fuzzy interference system so that rules can't be blend with one another and also testing these rules on robot in real time before operate in dynamic environment.

4.2. Examining Sensors and Rules

4.2.1 Testing Sensors

- First set the raspberry pi board to work
- Install the OS in raspberry pi
- Build a circuit on board using ultra sonic sensor interfacing it with raspberry pi The ECHO output yields 5v. The input pin of Raspberry Pi GPIO is evaluated at 3.3v. So 5v can't be straightforwardly given to the unprotected 3.3v at the input pin. In this manner we utilize a voltage divider circuit utilizing suitable resistors to cut down the voltage to 3.3V.

Fig. 4.1 Test circuit (with 3 sensors) to examine sensor's working

Open the python IDE for running the following program, the following program consists of the sample code for a single sensor's formulation.

```
#Libraries
import RPi.GPIO as GPIO
import time
#GPIO Mode (BOARD / BCM)
GPIO.setmode(GPIO.BCM)
GPIO.setwarnings(False)

#set GPIO Pins
TRIG1=14
ECHO1=18

#set GPIO direction (IN / OUT)
GPIO.setup(TRIG1, GPIO.OUT)
GPIO.setup(ECHO1, GPIO.IN)
def distance1():
    # set Trigger to HIGH
    GPIO.output(TRIG1, True)

    # set Trigger after 10us to LOW
    time.sleep(0.00001)
    GPIO.output(TRIG1, False)

    StartTime1 = time.time()
    StopTime1 = time.time()

    # save StartTime
    while GPIO.input(ECHO1) == 0:
        StartTime1 = time.time()

    # save time of arrival
    while GPIO.input(ECHO1) == 1:
        StopTime1 = time.time()

    # time difference between start and arrival
    TimeElapsed1 = StopTime1 - StartTime1
    # multiply with the sonic speed (34300 cm/s)
    # and divide by 2, because there and back
    distance1 = (TimeElapsed1 * 34300) / 2

    return distance1
```

Fig. 4.2 Python script for a single sensor

- Notice the analysis by testing the sensor

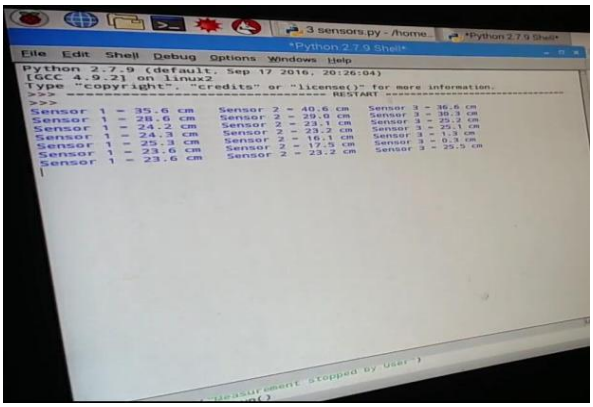


Fig.4.3 Analyzing 3 sensors readings

4.2.2 Testing fuzzy rules

The variables that are inputs from the sensors are normally mapped in a fuzzy control system through membership functions namely "fuzzy sets". We have set three rules far, near and very near based on the distance sensors getting from the obstacles through which the decision is made that in which direction the robot will move. Following are the test results for the fuzzy rules:

Fig. 4.4 Membership functions for fuzzification

5. CONCLUSION

This project shows the great benefits that fuzzy logic has brought to the performance of Autonomous mobile robot; this project shows the possible logic strategy for the performance improvement of motion control. The collision avoidance mobile robots have become a major area of research in robotics that involves various degrees of uncertainty. The developed fuzzy logic let the Autonomous mobile robot to have an ability to avoid the static obstacles coming in its path without the involvement of any human being. Fuzzy logic approach has been proved to be a simple and a more easy technique for such kind of robots. The Fuzzy logic control approach has proved to be a satisfactory control approach for the mobile robot to avoid the collision problem. It behaves intelligently in real time in order to prevent the robot to hit with any obstacle in the environment.

Overall, this project will serve as groundwork for much more meaningful investigation and development of more intelligent robots. This project can be further extended and can be used for the surveillance system by adding a camera. Furthermore, this project can also be used in lifting various things in mart or in an industry. This project can be used for implementing the idea of a driverless car. By modifying this robot this design can be used as a vacuum cleaner. It can also become a fire fighter robot by adding a camera.

REFERENCES:

Amruta N., A. Doddamani, D. Deshpande, S. Manjramkar, (2017) "Raspberry Pi Based Obstacle Avoiding Robot", International Research Journal of Engineering and Technology (IRJET) e-ISSN: 2395 - 0056 Vol.: 04 Issue: 02 . 67-72.

Carlo D'Ortenzio, (2012) "Understanding Change and Change Management Processes: A Case Study". Thesis degree of Doctor of Philosophy at the University of Canberra Canberra, AUSTRALIA.

Gurjashan S. P., M. D. Ansari, P. G. Design (2015) and Implementation of Autonomous Car using Raspberry Pi. International Journal of Computer Applications (0975 – 8887) Vol. 113 – No. 9 32Pp

Hachour, O. (2008) "Path planning of Autonomous Mobile robot", International Journal of Systems, Applications, Engineering & Development. Issue 4, Vol 2, 121-124.

Henry W., (2016) "Human inspired robotic path planning and heterogeneous robotic mapping ".A thesis submitted to the Victoria University of Wellington. Victoria University of Wellington.

http://engineering.nyu.edu/mechatronics/smart/pdf/Intro2Robotics.pdf

Hamami, G., A, M. Maznib, M. S. Amirc,F. Sukarmand and M. Katone. (2016) "A Review on Local Minimum and Multiple Minimum Avoidance Techniques in Local Path Planning". Journal of Advanced Research in Applied Sciences and Engineering Technology ISSN (online): 2462-1943 Vol. 4, No.1.58-71.

Riza L. S., C. Bergmeir, F. Herrera and J. M. Benitez (2015) "frbs: Fuzzy Rule-Based Systems for Classification and Regression in R". Journal of Statistical Software, Vol. 65, Issue 6.

Ram B. T., L. A. Gorisankar, (2016) "IoT for self-monitoring Analysis Remote Transducers". International Journal of Engineering and Technical Research (IJETR). Vol. 6, Issue 3, 44-50

Sarmasti E. M. R. (2010) "Fuzzy Logic Applications in Chemical Processes". The Journal of Mathematics and Computer Science Vol .1 No.4339-348.

Venkatashan P., and V. Sughanthi. (2015) "Space Detecting Robot using Ultra Sound Senor.". International Journal of Science, Engineering and Technology Research (IJSETR) Vol. 4, Issue 3.

Vidhya D. S., D. P. Reblo C. J., D'Silva L. W. Fernandas." (2016) Obstacle Detection using Ultrasonic Sensors". IJIRST –International Journal for Innovative Research in Science & Technology| Vol. 2 | Issue 11. 76-92.