AUTONOMOUS WHEELCHAIR WITH THE STORABLE MAP-GENERATION SYSTEM

A PROJECT REPORT

Submitted by

S.SILVAN (210916104031)
V.SURENDAR (210916104035)
S.THIRUNAVUKKARASU (210916104037)

in partial fulfillment for the award of the degree

of

BACHELOR OF ENGINEERING

IN

COMPUTER SCIENCE AND ENGINEERING

LOYOLA INSTITUTE OF TECHNOLOGY,

CHENNAI 600 123

ANNA UNIVERSITY: CHENNAI 600 025

MARCH 2020

ANNA UNIVERSITY::CHENNAI 600025

BONAFIDE CERTIFICATE

Certified that this project report "**AUTONOMOUS WHEELCHAIR WITH THE STORABLE MAP GENERATION SYSTEM**" is the bonafide work of "**S.SILVAN(210916104031)**", "**V.SURENDAR(210916104035)**", "**S.THIRUNAVUKKARASU(210916104037)**",who carried out the project work under my supervision.

SIGNATURE	**SIGNATURE**
DR.G.BHUVANESWARI.,Ph.D.,	**Ms.S.MADHAVI.,M.E.,**
HEAD OF THE DEPARTMENT,	**ASSISTANT PROFESSOR,**
Computer Science and Engineering,	Computer Science and Engineering,
Loyola Institute of Technology,	Loyola Institute of Technology
Chennai – 600 123.	Chennai – 600 123.

Submitted for the ANNA UNIVERSITY examination held on _________

INTERNAL EXAMINER **EXTERNAL EXAMINER**

ACKNOWLEDGEMENT

A project of this magnitude and nature requires kind cooperation and support from many for successful completion. We wish to express sincere thanks to all of those who were involved in the completion of the project.

It is our immense pleasure to express our deep sense of gratitude to our honourable chairman **Rev. Fr. Dr. J. E. ARULRAJ.,** for providing such an excellent lab facility, our correspondent **Rev. Sr. I. AROCKIYA MARY.**, for continuous motivation to complete the project.

We are extremely thankful to our beloved and dynamic principal **Dr. V. SUJATHA ANAND., Ph.D.**, for having given us an opportunity to demonstrate our innovative abilities.

We would like to thank our head, Department of Computer Science and Engineering, **DR. G. BHUVANESWARI., Ph.D.**, for the necessary help to complete the project successfully.

We take this moment as a great opportunity to thank our project coordinator **Mrs.P.RAMA,M.Tech.,(Ph.D).**, Assistant Professor, Department of Computer Science and Engineering and our guide **Ms. S. MADHAVI., M.E.,** Assistant Professor, Department of Computer Science and Engineering, for giving us their constant encouragement, support and valuable guidance to us during the project.

Last but not least, we are very grateful to our parents, friends and well-wishers for their constant encouragement during the project.

TABLE OF CONTENTS

ABSTRACT

Physically handicap people depend upon a wheelchair for navigation which restricts their mobility to a short-range mostly within indoors. The system must be manually controlled else map must be created and stored by the developers for navigation in an indoor location autonomously. All the user needed indoor map is difficult to create by a developer where he must be aware of the blueprint. We are proposing a solution where the system can generate an storable map based on which the user can navigate autonomously in the stored map location. The system perceives the environment through sensor data and constructs a 2D map and a path graph that consists of nodes to track walls, intersections, and rooms. Pathfinding algorithm is used to find the path between the current position and destination and the path is used to navigate to the destination autonomously.

LIST OF FIGURES

LIST OF ABBREVATIONS

ACRONYM	ABBREVATION
DC	DIRECT CURRENT
IC	INTEGRATED CIRCUIT
IDE	INTEGRATED DEVELOPMENT ENVIRONMENT
LED	LIGHT EMITTING DIODE
IR	INFRARED RADIATION

CHAPTER 1

INTRODUCTION

The most common form of assistance to provide support to the disabled is the wheelchair. Driving a wheelchair in domestic environments is a difficult task even for a normal person and becomes even more difficult for people with arms or hands impairments. Some disabled person who cannot manipulate the direction of the wheelchair with their arms due to a lack of force face major problems such as orientation, mobility, etc. The disabled person requires an aid for movement in both the indoor and outdoor environments. Therefore the Autonomous wheelchair is developed to overcome the above problems allowing the end-user to just perform safe movements and accomplish some daily life important tasks.

In an Autonomous wheelchair, the safety mechanism is used to detect the obstacles by ultrasonic sensors. Once it detect the obstacles it will stop the wheelchair and search for another way to move. The manual control one is a very reliable navigation control by using joystick or buttons.

An autonomous vehicle requires a map to navigate autonomously. The map is obtained by using GPS for outdoor navigation whereas for indoor navigation requires map stored by the developers. All the user needed indoor map is difficult to create by a developer where he must be aware of the locations blueprint. We are proposing a solution where the system can generate a storable map based on which the user can navigate autonomously in the map generated location.

CHAPTER 2

LITERATURE SURVEY

2.1 Sateesh Reddy Avutu, Dinesh Bhatia and B. Venkateswara Reddy (2016) ‘Design of low-cost manual cum Electric-powered wheelchair for Disabled person’s to use in indoor’.

A novel wheel chair named as “Design of low-cost manual cum electric-powered wheelchair for disabled person for indoor usage”. In this design, we employed Mechanical lever and gear box system. The lever is used to change mode of operation and the gear box system is used, to establish the contact between wheels of the wheel chair and two DC motors. The calculated result shows the robustness of the manual cum powered wheelchair design. Based on the result we ensure that the proposed wheelchair will improve the quality of the life of the elderly people and those with disabilities.

LIMITTIONS

It requires manual commands for operation. Through slight modifications we can make the system autonomous which will make the system more convenient to the user.

2.2 Khyati Meena ,Shubham Gupta and Vijay Khar (2017) ‘Voice Controlled Wheelchair’.

This paper presents an automatic wheel chair using voice recognition. A voice controlled wheelchair makes it easy for physically disabled person who cannot control their movements of hands. The powered wheel chair depends on motors for locomotion and voice recognition for command. The circuit comprises of an Arduino, HM2007 Voice recognition module and Motors. The voice recognition module recognizes the command by the user and provides the corresponding coded data stored in the memory to Arduino Microcontroller. Arduino Microcontroller controls the locomotion accordingly. The wheelchair also has provision for joystick for physically disabled people who can move their hands.

LIMITATIONS

One problem with voice control is that the voice’s limited bandwidth renders it impossible to make frequent small adjustments to the wheelchair’s velocity.

2.3 Harkishan Singh Grewal, Neha Thotappala Jayaprakash, Aaron Matthews, Chinmay Shrivastav, and Kiran George (2018) ‘Autonomous Wheelchair Navigation in Unmapped Indoor Environments’.

A novel approach to selecting a destination for an autonomous wheelchair in an unmapped indoor environment using a camera, ranging LIDAR, and computer vision is presented. The system scans the environment at startup and compiles a list of possible destinations for a user to easily make an selection. The proposed system was tested in a simulated shopping mall environment where destinations included various stores. The computer vision system was tested with images of store-fronts at various distances and angles.

LIMITATIONS

Using Wi-Fi Fingerprinting Localization can give better results than this method hence this method will take more time for path finding using computer vision.

2.4 Tarun Debnath, AFM Zainul Abadin & Md. Anwar Hossain (2018) 'Android Controlled Smart Wheelchair for Disabilities'.

This paper describes a control technology of wheelchair which may feel more flexible than traditional joystick controlled one. The main objective of our research is to develop new control architecture for a motorized wheelchair as well as an embedded system for monitoring critical patients. Such a smart wheelchair is designed for the disabled people in the developing countries as it will be very low-cost than existing others. Controlling is possible by android operated mobile or tab. In addition to button control, motion sensor controlling mechanism also has implemented. Moreover, bio-metric features have made wheelchair more suitable for critical patients. If the patient is in hostile condition, the wheelchair will produce an alert by raising the alarm with the measurement of the heartbeat at a particular interval.

LIMITATIONS

Inclusion of the storeable map generation system can make it more efficient to navigate independently in emergency situation to nearby hospitality.

2.5 Behnam Irani, Jingchuan Wang and Weidong Chen (2018) 'A Localizability Constraint-Based Path Planning Method for Autonomous Vehicles'.

A localizability constraint (LC)-based path planning method for autonomous vehicles which plans the navigation path according to LRF sensor model of the vehicle in an effort to maintain a satisfactory level of localizability throughout the path, as well as to reduce the overall localization error. This method is not limited to any specific algorithm in the optimization stage. Paths planned with and without LC are compared, and the influence of the LRF sensor model on planning outcomes is discussed through simulations.

LIMITATIONS

SLAM along with Kalman filtering and particle filtering algorithms based localization is more efficient than Constraint-Based planning method and considered to be with less localization errors.

2.6 Wenjing Li, Dan Hu and Zhiyong Lin (2018) 'Indoor Space Dimensional Model Supporting the Barrier-free Path-finding'.

A three-layer indoor space dimensional model, it contains geometric information, topological information and semantic information based on several traditional indoor space models. First geometric information, topological information and semantic information were extracted from the source data. Then the general semantic description of the indoor space and the topological relations of indoor space components were constructed. According to different scenes, multiple properties of user's behaviors and components' states were specified. The three layers of information were merged into a dimensional model, which was the basis for indoor path planning in different scenes. The geographic information software was used to plan indoor paths for users in ordinary scene and accessible scene respectively. It can adapt to the indoor routing requirements in different scenes and is easy to update.

LIMITATIONS

It requires more processing power due to generation of three layer of the indoor space dimensions. Hence, other major operation of the system can slow down.

2.7 Ananthakrishnan D.S, Jishnu Prakash K, Renjith R and Dr.Ansamma John (2019) 'Autonomous Indoor Navigation for Wheelchairs using Signboards'.

An autonomous indoor navigation system for wheelchairs based on sign board recognition. The system uses a deep learning model to detect signboards from surroundings and Azure Text Analytics API is used to extract the text from the signboard images. The system runs on a Raspberry Pi minicomputer and can be installed on any powered wheelchair.

LIMITATIONS

Development of such system are expensive.Hence it cannot be afforded by most of the people.

2.8 Rami Alkhatib, Afif Swaidan, Jana Marzouk, Maher Sabbah, Samir Berjaoui and Mohamad O.Diab (2019) 'Smart Autonomous Wheelchair'.

Smart wheelchairs are an assistive wheeled mobility device. Wheelchair made it easier for many to pursue their life activities including education, work and social life. A smart autonomous wheelchair is developed in this paper to enhance the maneuvering tasks. The wheelchair requires no human intervention during navigation and perception in addition to processing which is based on computer vision techniques.

LIMITATIONS

Can be only operated in stored map location. Including map generation and storing algorithm will make it more reliable and efficient.

2.9 Hassan M. Qassim and Heba Lakany (2019) ‘Virtual Environment Modelling using Simulated Laser Scanners’.

A semiautonomous electric wheelchair that has capabilities of navigating through various environments which include different types and sizes of obstacles. This paper describes a methodology to use a range laser scanner mounted on an electric wheelchair to map different environments. The electric wheelchair is simulated in a virtual environment. Mapping the environment is dependent on the information provided by the range laser. An algorithm was developed in MATLAB to record the data received by the range laser and use the data to map the environments and produce a 2D map. The suggested algorithm has been tested using two virtual environments representing rooms with different features. The results showed that range laser scanner can be used on an electric wheelchair platform for efficient mapping of the environment.

LIMITATIONS

Including our map storable algorithm can give make the system more efficient in autonomous navigation.

2.10 Jose Victorio Salazar Luces, Kengkij Promsutipong and Yasuhisa Hirata (2020) 'Indoor Way finding for an Electric Wheelchair Based on Wi-Fi Fingerprinting Localization'.

A method to enable the device to navigate toward the user upon request by using onboard sensors and the existing Wi-Fi infrastructure. Specifically, A map created with the Received Signal Strength Indicator (RSSI) of existing Wi-Fi access points (using a method called Fingerprinting). When a user requests an assistive device, the RSSI values at the user's position are sent to it, and the device determines the rough position of the user using a KNN algorithm. However, typical Fingerprinting methods are affected by infrastructural changes which alter the profile of RSSI values at each location. Therefore, that the map is constantly updated while the device moves in order to avoid errors due to changes in the infrastructure. The request originated with an error of 2.612 m, and it was able to navigate towards it.

LIMITATIONS

These systems are still quite expensive, and having this system implemented is difficult.

CHAPTER 3

SYSTEM ANALYSIS

3.1 EXISTING SYSTEM

The existing system consists of a standard powered wheelchair with an on-board computer, sensors and a graphical user interface. It utilizes voice control in combination with the navigation assistance which uses sensors to identify and avoid obstacles in the wheelchair's path.

Wheelchair navigation system based on a PDA equipped with wireless Internet access and GPS that can provide adaptive navigation support to wheelchair users in any geographic environment. Autonomous wheelchair requires map as input to navigate autonomously in unknown indoor. A map is created and stored by the developer for autonomous navigation in indoor.

3.1.1 LIMITATION OF EXISTING SYSTEM

- High cost
- Requires map to navigate autonomously
- High execution time
- Performance issues

3.2 PROPOSED SYSTEM

Our proposed system can generate an storable map based on which the user can navigate autonomously in the generated map location. The system perceives the environment through sensor data and constructs a 2D map and a path graph that consists of nodes to track walls, intersections, and rooms. Path finding algorithm can then be used to find the path between the current position and destination and the path is used to navigate to the destination autonomously without requirement to manual instruction.

3.2.1 ADVANTAGES OF PROPOSED SYSTEM

- User can create his own map without any prerequisite knowledge
- Requires less time
- Reliable
- High performance

CHAPTER 4

SYSTEM REQUIREMENTS

4.1 HARDWARE REQUIREMENTS

- Arduino Micro-controller
- Ultrasonic sensor
- Motor driver IC
- Bluetooth
- DC motor
- Switch
- Map generator

4.2 SOFTWARE REQUIREMENTS

- Arduino IDE
- Python IDE

CHAPTER 5

SYSTEM DESIGN

5.1 SYSTEM ARCHITECTURE

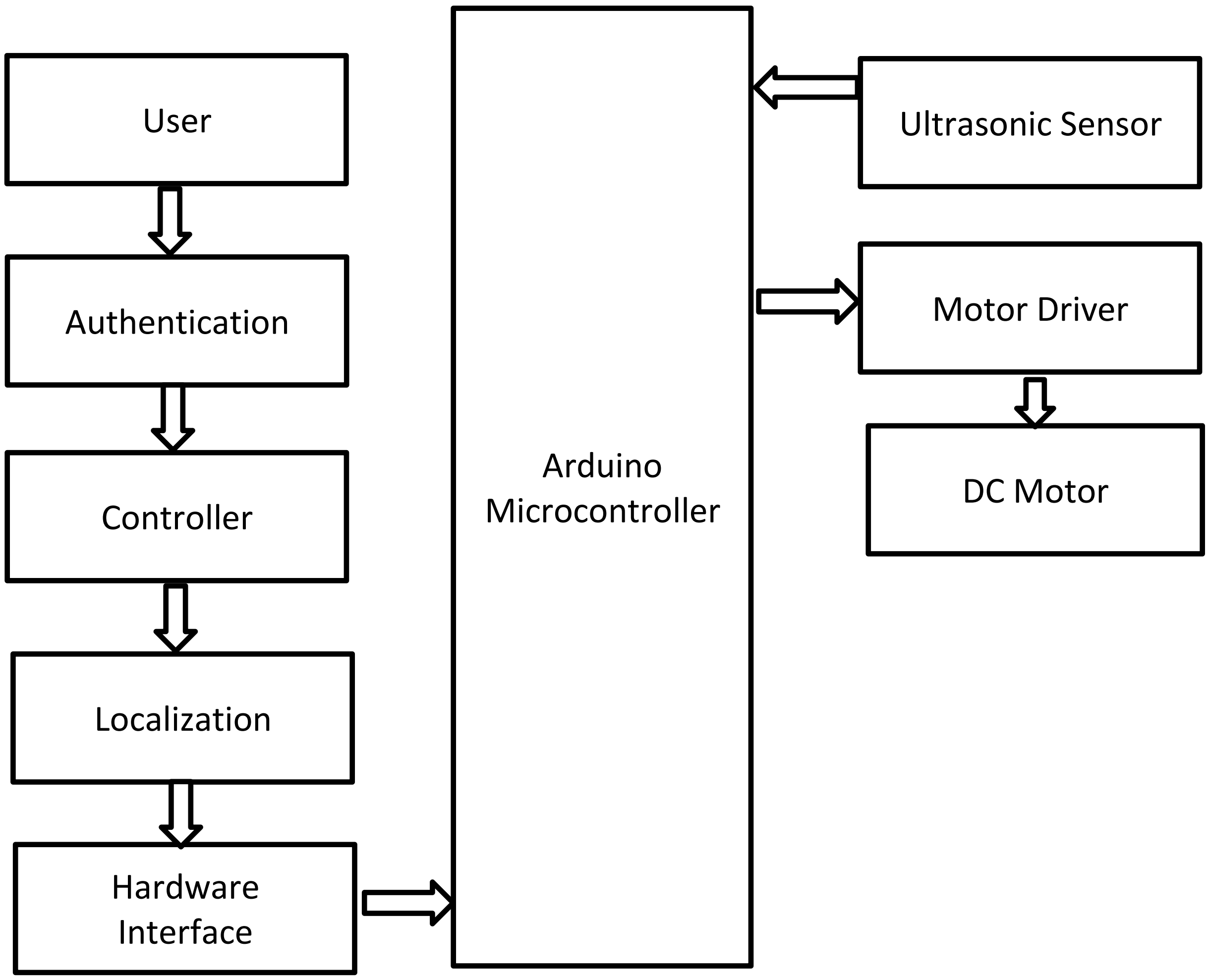


Fig 5.1.1 Architecture Diagram

5.2 WORKING PRINCIPLE

The Autonomous wheelchair gets input from the user through the keyboard and traverses through the path and detects obstacles through the ultrasonic sensor while navigating. While moving the sensor data is transferred to the map generation system via Bluetooth. The map generation systems use the data to create a 2D map with a graph structure. The generated map can be used to navigate autonomously without manual instruction once saved. When the user enters the destination node the system detects the efficient path by using the Path finding algorithm and uses the map to navigate by itself. The system interacts with the micro-controller and gives the command to the controller of the wheelchair to reach the destination from the current node updating the result in a viewing device.

After successful authentication of the user in the computer, the controller module gets initialized. The computer-user interaction is carried out in this module. The user has to choose whether to load or create a map based on which the system decides to perform map generation or to load the existing map and navigate.

In the map generation phase, the system operates in manual mode where the user has to control the system. While navigating in manual mode the system creates a map by interacting with the map generation module. The user can use the generated map temporarily or store it for future navigation.

If the user chooses to load the map the user is requested to enter the destination node from the displayed graph-structured map. The system will communicate with the localization module to determine its current position. Then it requests the map generation module for the path between the current position to the destination. After obtaining the path the controller calls the navigation methods to make the wheelchair navigate to the destination.

Obstacle avoidance is the task of satisfying some control objective subject to non-collision position constraints.Normally obstacle avoidance is considered to be distinct from path planning in that one is usually implemented as a reactive control law while the other involves the pre-computation of an obstacle-free path which a controller will then guide a robot along. With recent advanced in the autonomous vehicles sector, a good and dependable obstacle avoidance feature of a driver-less platform is also required to have a robust obstacle detection module..The obstacle detection system can detect both static and dynamic obstacles. Obstacles are detected through the data perceived from an ultrasonic sensor mounted at the front. While right and left turn the obstacle is detected using the side-mounted sensors.

CHAPTER 6

MODULES DESCRIPTION

6.1 AUTHENTICATION

Authentication is to verify the identity of the user to avoid access to the system by an unauthorized person. Authentication in the system is carried out in two phase as follows, Password based Authentication and OTP based Authentication.

Fig 6.1.1 Authentication

Password based Authentication

Simple password authentication offers an easy way of authenticating users. The user can gain permission to control the system by entering the user-name and password. If the user is already registered then the system will allow the user to control the vehicle else the system will deny the access. A new user can gain access over the system through registration with the help of the admin user.

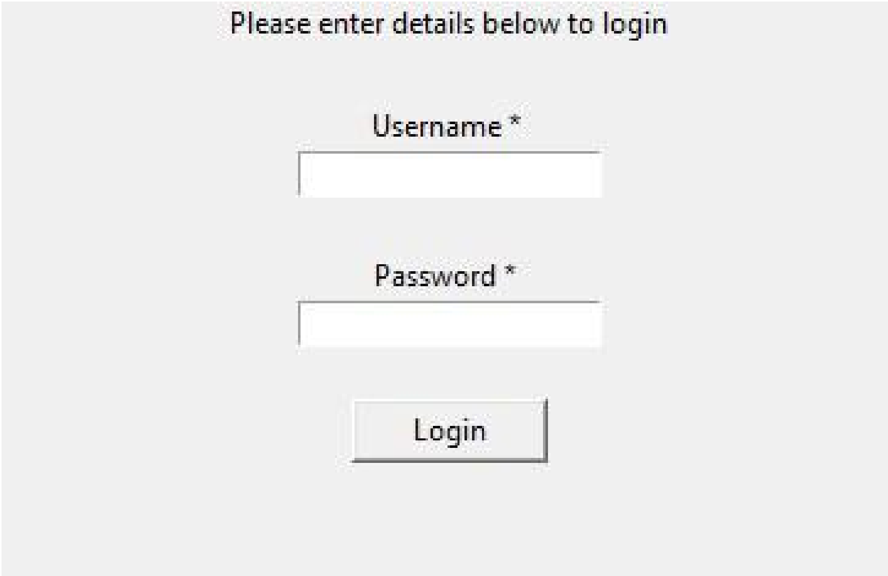


Fig 6.1.2 Login form

OTP based Authentication

Two-factor authentication is used to avoid registration of the unauthorized user. OTP will be sent to the admin's mobile number during the registration process. If the user enters the correct OTP as sent to the admin user mobile number the new user details get registered else the user will be asked to enter correct OTP.

Fig 6.2.3 Registration form

Encryption and Decryption

Encryption is the method of transforming plain text into cipher text to make unintelligible for asset protection. The password gets encrypted and decrypted using bcrpyt encryption and decryption algorithm which is currently the security standard for password hashing. Bcrypt is derived from the blowfish block cipher, to generate the hash, uses look-up tables which are initiated in memory.

6.2 CONTOLLER

A Centralized system needs to control all the operations of the system to make it more reliable and efficient. The controller acts as the centralized system communicates with all other modules to send instructions to wheelchair and receive perceived data from ultra-sonic sensors. After successful authentication, the controller module gets initialized. The computer-user interaction is carried out in this module. The user has to choose whether to load or create a map based on which the system decides to perform map generation or to load the existing map and navigate.

In the map generation phase, the system operates in manual mode where the user has to control the system. While navigating in manual mode the system creates a map by interacting with the map generation module. The user can use the generated map temporarily or store it for future navigation.

If the user chooses to load the map the user is requested to enter the destination node from the displayed graph-structured map. The system will communicate with the localization module to determine its current position. Then it requests the map generation module for the path between the current position to the destination. After obtaining the path the controller calls the navigation methods to make the wheelchair navigate to the destination.

Obstacle avoidance is the task of satisfying some control objective subject to non-collision position constraints.Normally obstacle avoidance is considered to be distinct from path planning in that one is usually implemented as a reactive control law while the other involves the pre-computation of an obstacle-free path which a controller will then guide a robot along. With recent advanced in the autonomous vehicles sector, a good and dependable obstacle avoidance feature of a driver-less platform is also required to have a robust obstacle detection module..The obstacle detection system can detect both static and dynamic obstacles. Obstacles are detected through the data perceived from an ultrasonic sensor mounted at the front. While right and left turn the obstacle is detected using the side-mounted sensors.

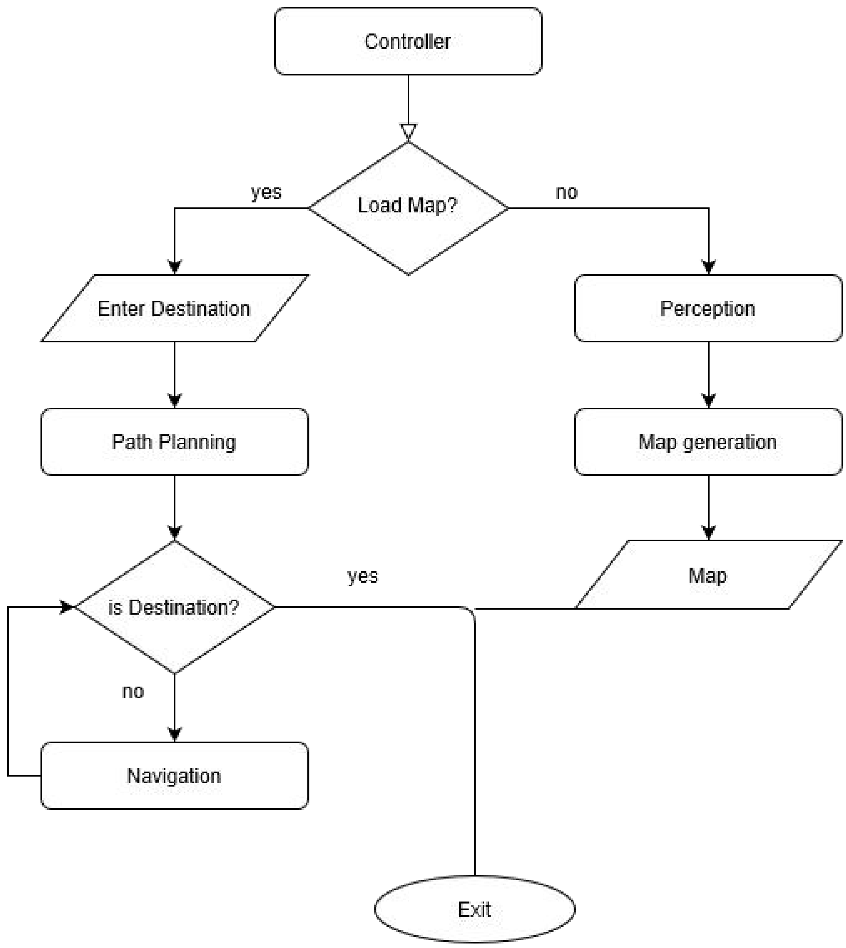


Fig 6.2.1 Flow diagram of Controller

6.3 HARDWARE INTERFACE

Hardware interface acts as the middleware between the hardware and software it handles Perception and Navigation of the wheelchair. Its role is to send navigation instructions and receive sensor data from the hardware which is required to generate a map.

Initially, Bluetooth connection is established between the Laptop and Bluetooth module in the wheelchair. The connection is used to transmit navigation instructions and receive back perceived sensor data from the ultrasonic sensor. Navigation of the wheelchair can be classified into two modes as follows,

Manual mode

The system works based on user commands. The user has to control the system via commands for each action. The user can control the system using the arrow keys in the keyboard. In manual mode, the perception is taken place while not in autopilot mode. If there is no stored map the user has to operate the system in manual mode.

Autopilot mode

In the presence of the map, the user need not control the system manually. The user has to specify the destination whereas the system plans the path from the current location to the destination. Using the path the system reaches the destination automatically.

To choose autopilot mode the system must consist map of the traversing location. It controls the navigation of the system by sending the following navigation commands.

Instruction	Description
0	Stop
1	Turn left with getting ultrasonic sensor reading
2	Move forward with getting ultrasonic sensor reading
3	Turn right with getting ultrasonic sensor reading
4	Turn left without getting ultrasonic sensor reading
5	Move forward without getting ultrasonic sensor reading
6	Turn right without getting ultrasonic sensor reading

Table 6.3.1 Navigation instructions

6.4 LOCALIZATION

Localization is a step implemented in the majority of robots and vehicles to locate with a really small margin of error. There are many different techniques to help an autonomous vehicle locate itself. A very popular technique if we also want to estimate the map exists. It is called SLAM (Simultaneous Localization And Mapping). In this technique, we estimate our position but also the position of landmark. Localization has been a major part of autonomous vehicles to locate itself in the environment. Its can be categorized into two major phases.

Global localization is to locate its position in the global environment. It is carried out using the sensor data perception phase. The environment is mapped through the sensor data received and the map can be used to localize itself.

The goal of **pose tracking** is to maintain accurate tracking of a vehicle's pose during the whole task. The pose of the system is to be updated during each movement of the wheelchair. The system uses an odometer to track the rotation and movement of the vehicle. The map in the system is a graph structure hence the localization can be represented as x and y coordinates.

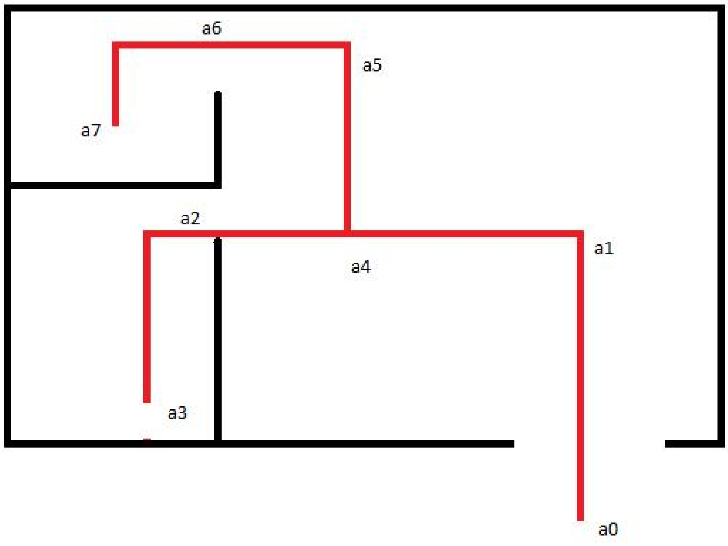


Fig 6.4.1 Localization

CHAPTER 7

CONCLUSION AND FUTURE ENHANCEMENT

The system generates a storable map based on the path tracked by the system, thus indoor navigation can be made autonomous and improved by using the map generated by the system at minimum cost and time.

Furthermore, navigation can be made more accurate by generating a 3D map and adding computer vision to the system whereas our system cannot detect doors, tables, and chairs which can detected by including computer vision.

APPENDIX 1

SAMPLE CODE

Arduino:

```
#include "SoftwareSerial.h"
SoftwareSerial serial(1,0);
int trigF=2;
int echoF=3;
int trigL=4;
int echoL=5;
int trigR=6;
int echoR=7;
int motor1F=9;
int motor1B=8;
int motor2F=11;
int motor2B=10;
float sensorReading;
int distance;
int sensorF;
int sensorR;
int sensorL;
int instruction;
bool obstacle;
int forwardDelay=900;
int rotationDelay=9*forwardDelay;
int forwardLimit=20;
int rotationLimit=30;
```

```
void setup() {
  serial.begin(9600);
  pinMode(trigF,OUTPUT);
  pinMode(echoF,INPUT);
  pinMode(trigL,OUTPUT);
  pinMode(echoL,INPUT);
  pinMode(trigR,OUTPUT);
  pinMode(echoR,INPUT);
  pinMode(motor1F,OUTPUT);
  pinMode(motor1B,OUTPUT);
  pinMode(motor2F,OUTPUT);
  pinMode(motor2B,OUTPUT);
}
void loop(){
  while(serial.available()==0){}
  instruction=serial.parseInt();
  switch(instruction){
    case 1:
    sensorR=readSensor(trigR,echoR);
    obstacle=obstacleDetector(sensorR,rotationLimit);
    if(not(obstacle)){
      motorDrive('l');
    }
    sensorF=readSensor(trigF,echoF);
    sensorL=readSensor(trigL,echoL);
    commandGenerator(obstacle,sensorF,sensorR,sensorL);
    break;
    case 2:
```

```
sensorF=readSensor(trigF,echoF);
obstacle=obstacleDetector(sensorF,forwardLimit);
if(not(obstacle)){
  motorDrive('f');
}
sensorR=readSensor(trigR,echoR);
sensorL=readSensor(trigL,echoL);
commandGenerator(obstacle,sensorF,sensorR,sensorL);
break;
case 3:
sensorL=readSensor(trigL,echoL);
obstacle=obstacleDetector(sensorL,rotationLimit);
if(not(obstacle)){
  motorDrive('r');
}
sensorF=readSensor(trigF,echoF);
sensorR=readSensor(trigR,echoR);
commandGenerator(obstacle,sensorF,sensorR,sensorL);
break;
case 4:
sensorR=readSensor(trigR,echoR);
obstacle=obstacleDetector(sensorL,rotationLimit);
if(not(obstacle)){
  motorDrive('l');
}
break;
case 5:
sensorF=readSensor(trigF,echoF);
```

```
    obstacle=obstacleDetector(sensorF,forwardLimit);
    if(not(obstacle)){
      motorDrive('f');
    }
    break;
    case 6:
    sensorL=readSensor(trigL,echoL);
    obstacle=obstacleDetector(sensorR,rotationLimit);
    if(not(obstacle)){
      motorDrive('r');
    }
    break;
  }
}
float readSensor(int trig,int echo){
  digitalWrite(trig,LOW);
  delayMicroseconds(20);
  digitalWrite(trig,HIGH);
  delayMicroseconds(20);
  digitalWrite(trig,LOW);
  sensorReading=pulseIn(echo,HIGH);
  distance=(0.0451*sensorReading)/2;
  return distance;
}
bool obstacleDetector(int distance,int limit){
    if(distance<=limit){
      return true;
    }
```

```
    else{
      return false;
    }
}
void motorDrive(char movement){
  switch(movement){
    case 'f':digitalWrite(motor1F,HIGH);
             digitalWrite(motor2F,HIGH);
             delay(forwardDelay);
             digitalWrite(motor1F,LOW);
             digitalWrite(motor2F,LOW);
             break;
    case 'r':digitalWrite(motor1B,HIGH);
             digitalWrite(motor2F,HIGH);
             delay(rotationDelay);
             digitalWrite(motor1B,LOW);
             digitalWrite(motor2F,LOW);
             break;
    case 'l':digitalWrite(motor1F,HIGH);
             digitalWrite(motor2B,HIGH);
             delay(rotationDelay);
             digitalWrite(motor1F,LOW);
             digitalWrite(motor2B,LOW);
             break;
  }
}
```

Python:

```
import serial
arduinoSerial=serial.Serial('com6',9600)
arduinoSerial.flushInput()
def readSerial(command):
    writeSerial(command)
    n=0
    while True:
        if(arduinoSerial.inWaiting()>0):
            break
    sensor=arduinoSerial.readline()
    return sensor
def writeSerial(command):
arduinoSerial.write(command)
sensorData=[]
states=[['q0',0,0]]
edges=[]
import turtle
import tracer
import localization
import graphGenerator
import pathFinder
import arduinoInterface
screen=turtle.Screen()
screen.screensize(800,500)
leftWall=turtle.Turtle()
leftWall.left(90)
```

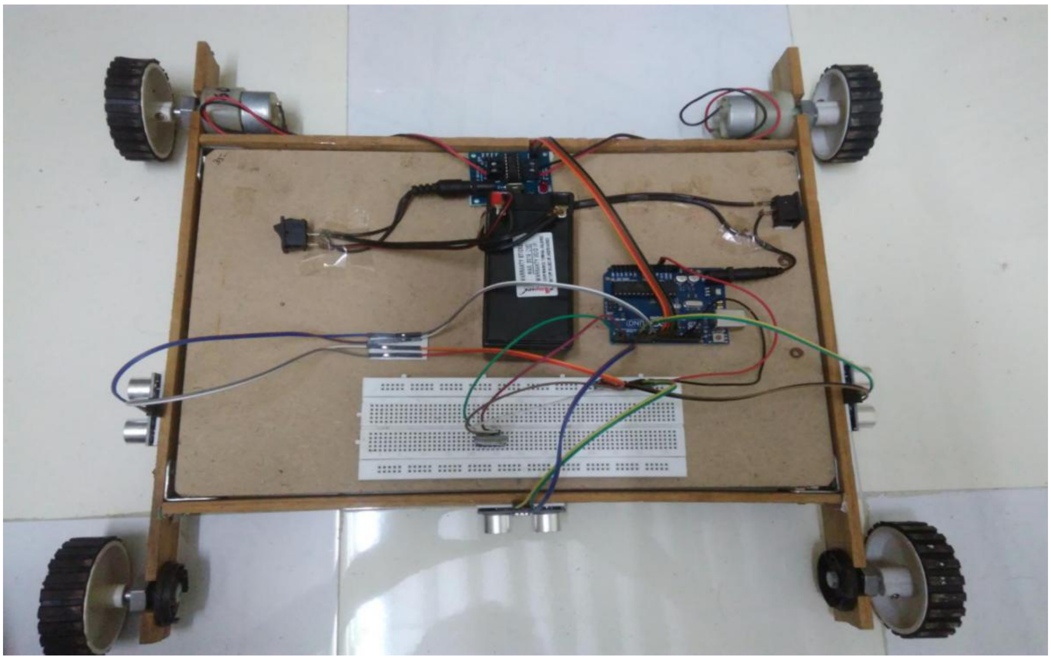
```
path=turtle.Turtle()
path.left(90)
path.write('q0')
rightWall=turtle.Turtle()
rightWall.left(90)
wallLimit=140
forwardLimit=20
rotationLimit=30
def drawWall(instruction,movement):
    head=path.heading()
    x=path.xcor()
    y=path.ycor()
    sensorDataLeft=instruction[0]
    sensorDataRight=instruction[2]
    if movement == 'L' or movement == 'R':
        leftWall.penup()
        rightWall.penup()
    else:
        checkLimit(instruction)
    if head == 0:
        leftWall.setposition(x-forwardLimit,y+sensorDataLeft)
        rightWall.setposition(x-forwardLimit,y-sensorDataRight)
        leftWall.setheading(head)
        rightWall.setheading(head)
    if head == 90:
        leftWall.setposition(x-sensorDataLeft,y-forwardLimit)
        rightWall.setposition(x+sensorDataRight,y-forwardLimit)
        leftWall.setheading(head)
```

```
        rightWall.setheading(head)
    if head ==180:
        leftWall.setposition(x+forwardLimit,y-sensorDataLeft)
        rightWall.setposition(x+forwardLimit,y+sensorDataRight)
        leftWall.setheading(head)
        rightWall.setheading(head)
    if head == 270:
        leftWall.setposition(x+sensorDataLeft,y+forwardLimit)
rightWall.setposition(x-sensorDataRight,y+forwardLimit)
        leftWall.setheading(head)
        rightWall.setheading(head)
def rotateLeft():
    data=tracer.trace(b'1')
    x,y=localization.getCoordinates()
    state=graphGenerator.setState(x,y)
    if state !=0 :
        path.write(state)
    if data[2]<rotationLimit:
        pass
    else:
        path.left(90)
        drawWall(data,'L')
def rotateRight():
    data=tracer.trace(b'3')
    x,y=localization.getCoordinates()
    state=graphGenerator.setState(x,y)
    if state !=0 :
        path.write(state)
```

```
    if data[2]<rotationLimit:
        pass
    else:
        path.right(90)
        drawWall(data,'R')
def moveForward():
    data=tracer.trace(b'2')
    localization.setCoordinates(int(path.heading()))
    x,y=localization.getCoordinates()
    globalState=graphGenerator.checkCurrentState(x,y)
    if data[1]<forwardLimit:
  pass
    else:
        path.forward(20)
        drawWall(data,'F')
turtle.listen()
turtle.onkey(rotateLeft,"Left")
turtle.onkeypress(moveForward,"Up")
turtle.onkey(rotateRight,"Right")
turtle.onkey(pathGenerator,"Tab")
turtle.mainloop()
```

APPENDIX 2

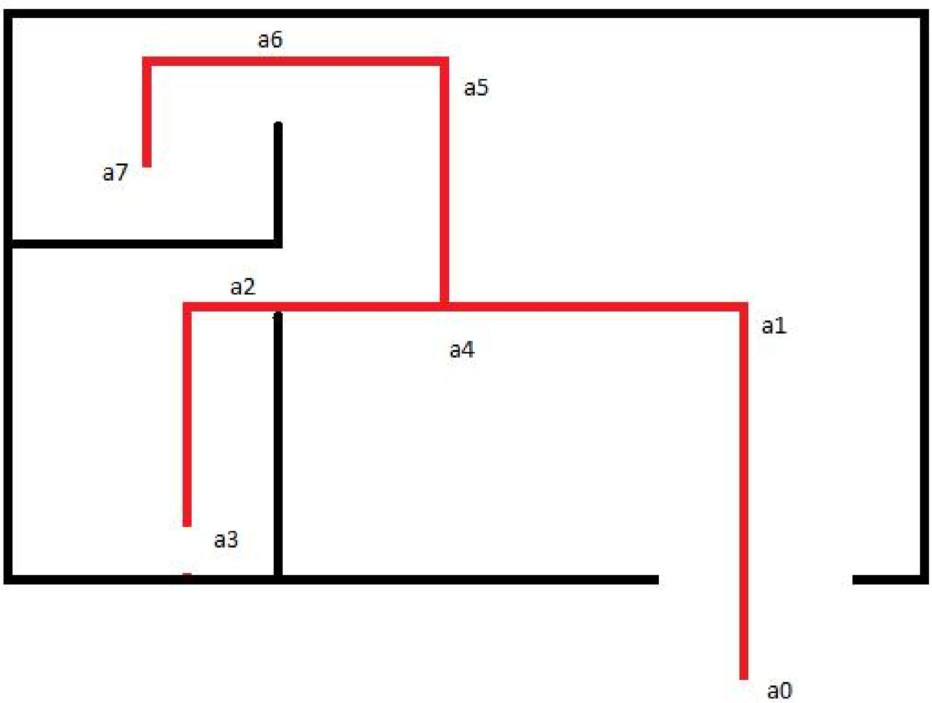
SCREENSHOTS



A.2.1: Wheelchair prototype

A.2.2: Log in page

A.2.3: Register page



A.2.4: Map generated

REFERENCES

1. Sateesh Reddy Avutu, Dinesh Bhatia and B. Venkateswara Reddy (2016) ‘Design of low-cost manual cum Electric-powered wheelchair for Disabled person’s to use in indoor’, International Conference on Next Generation Computing Technologies,pp. 178-184.
2. Khyati Meena ,Shubham Gupta and Vijay Khar (2017) ‘Voice Controlled Wheelchair’, 2017 2nd International Journal of Electronics, Electrical and Computational System, Volume 6,pp-23-27.
3. Harkishan Singh Grewal, Neha Thotappala Jayaprakash, Aaron Matthews, Chinmay Shrivastav, and Kiran George (2018) ‘Autonomous Wheelchair Navigation in Unmapped Indoor Environments’, National Science Foundation under grant no. ECCS-1150507.
4. Tarun Debnath, AFM Zainul Abadin & Md. Anwar Hossain (2018) ‘Android Controlled Smart Wheelchair for Disabilities’ ,Global Journal of Computer Science and Technology: G Interdisciplinary, Volume 18, Issue 1,Online ISSN: 0975-4172.
5. Behnam Irani, Jingchuan Wang and Weidong Chen (2018) ‘A Localizability Constraint-Based Path Planning Method for Autonomous Vehicles’, IEEE transactions on Intelligent Transportation Systems, ISSN:1524-9050.
6. Wenjing Li, Dan Hu and Zhiyong Lin (2018) ‘Indoor Space Dimensional Model Supporting the Barrier-free Path-finding’, National Science Foundation of China under Grant No. 41271449.
7. Ananthakrishnan D.S, Jishnu Prakash K, Renjith R and. Dr. Ansamma John (2019) ‘Autonomous Indoor Navigation for Wheelchairs using Signboards’, IEEE transactions on Intelligent Transportation Systems under Grant.

8. Rami Alkhatib, Afif Swaidan, Jana Marzouk, Maher Sabbah, Samir Berjaoui and Mohamad O.Diab (2019) ‘Smart Autonomous Wheelchair’, IEEE , Grant no:978-1-7281-3578-6/19.
9. Hassan M. Qassim and Heba Lakany (2019) ‘Virtual Environment Modelling using Simulated Laser Scanners’, 2 nd International Conference On Electrical, Communication, Computer, Power And Control Engineering ICECCPCE19, pp-143-147.
10. Jose Victorio Salazar Luces, Kengkij Promsutipong and Yasuhisa Hirata (2020) ‘Indoor Way finding for an Electric Wheelchair Based on Wi-Fi Fingerprinting Localization’, IEEE/SICE International Symposium on System Integration, pp-513-518.