**DOI: 10.5281/zenodo.4288291**
**CZU 615.478.3:004**

# A SMART OMNIDIRECTIONAL CONTROLLED WHEELCHAIR

Rabiu Omeiza Isah[1*], ORCID ID:0000-0002-2784-8891,
O. M. Olaniyi[1], ORCID ID:0000-0002-2294-5545,
J. G. Kolo[2], ORCID ID:0000-0003-3687-5960,
D. Z. Babatunde[1], ORCID ID:0000-0003-2835-6028

[1]*Department of Computer Engineering, Federal University of Technology, Minna*
[2]*Department of Electrical and Electronic Engineering, Federal University of Technology, Minna*
*Corresponding author: Isah Omeiza Rabiu, *isah.rabiu@futminna.edu.ng*

Received: 10. 08. 2020
Accepted: 11. 22. 2020

**Abstract.** Wheelchair is used by individuals that find it challenging to walk. Various methods have been adopted in developing wheelchairs to suit the needs of the physically disabled using the available technologies. Problems associated with the already existing wheelchairs are difficulty in maneuvering them in a confined and constrained space and limitation in degree of movement they can achieve. The aim of this project is to develop a smart omnidirectional controlled wheelchair. The system is not only applicable for domestic use, it can also be used for sports and in the hospital. The movement of the wheelchair is controlled from a web application via Wireless Fidelity communication. The methodology employed includes designing the web application interface using Hypertext markup language and JavaScript programming language, the hardware part consists of the Raspberry Pi 3 Model B, programmed with python programming language. Then, the software and hardware part were integrated together to form a complete system. The main advantage of the system is that it allows the user of a wheelchair to maneuver through a confined and constrained space and control the wheelchair remotely. The performance measure considered were the accuracy of the obstacle detection unit in detecting brick wall, metal and wood, and the response time of the wheelchair to movement commands from the web application. The average detection accuracies for the brick wall, metal, and wood were 87.37%, 94.43%, and 83.57% respectively. The average response time of the wheelchair to movement commands from the web application was 1.04 seconds.

**Keywords:** *Omnidirectional, Wireless Fidelity, JavaScript, Raspberry Pi 3 Model B.*

## Introduction

Wheelchair is simply a chair that has wheels, used by individuals that find it challenging or impossible to walk due to injuries, disabilities and other health related problems [1]. There are different types of wheelchairs, which are categorized based on their specific features. These features may range from configuration of the wheels which can be four caster wheels, two motorized wheels with two caster wheels, two motorized wheels with four caster wheels or four motorized wheels, driving force of the wheels (powered or manual wheelchairs), to the type of additional technology used such as obstacle detection, seating adaptations, tilting forward or backward. Powered wheelchairs also vary in terms of their mode of controls which can be joystick-controlled, eye-controlled, voice-controlled, smartphone, tilt-controlled wheelchair [2]. The notable types of wheelchair that have been developed over the years are manual self-propelled wheelchairs, manual attendant-propelled wheelchairs, powered wheelchairs, single-arm drive wheelchairs, tilting and reclining wheelchairs, standing wheelchairs, sports wheelchairs, smart wheelchairs, in which each of them has their own specific features [3].

Recently, technological advancement has led to the improvement of manual wheelchair and powered wheelchair (powerchair), improvement mainly in methods of control of the wheelchair and the degree of mobility achievable by the wheelchair [4]. Some of the renowned improvements on wheelchair are construction of hand-control levers for Leveraged Freedom Chair manually propelled and designed to be low-cost for users in developing countries [5]. This allows the wheelchair to be used on an uneven terrain and to overcome obstacles and addition of two-geared wheels to manual wheelchair, which provides gearing leverages to the user. It offers an efficient speed ratio for the user (2:1), resulting in 100% efficiency in climbing hills [6].

Providing comfortability and ease of mobility to the physically disabled is a very crucial factor to be considered in the development of a wheelchair [7]. This factor is heavily dependent on the method of control adopted and the degree of mobility that is to be achieved. There are various problems that are encountered by the physically challenged persons while operating the wheelchair. The challenges include difficulty in manoeuvring the wheelchair in a confined and constrained space and limitations in degree of movement of a wheelchair [8]. In this paper, we present the development of a smart omnidirectional controlled wheelchair that can move without infringements using the developed web application. The proposed method of development as compared to other adopted methods has total control over the movement of the wheelchair, which in turn gives the user the feeling of freedom and liberty to move as they wish.

The remaining sections of the paper are divided into four. Section two presents the review of related baseline works. Section three presents design overview, Section four analyses and discusses results obtained while section five is the conclusion of the paper. Finally, section six presents future directions.

**Related works**

This work is about the design of a smart omnidirectional wheelchair to ease the life of the physically disabled. The use of four motorized wheel and web application control system is a technology that has not been previously considered in the construction of a wheelchair, some of the previous technologies used are two motorized wheels with two caster wheels or two motorized wheels with four caster wheels controlled with voice, joystick, brain impulse, or android based application (Autodesk, Inc., 2015). Thus, the review of related works alongside their limitations are presented:

The use of eye tracking as the method of control for wheelchair was implemented in [9]. This method of control provided mobility for physical disabilities such as monoplegia, paraplegia, hemiplegia and quadriplegia. And, the use of PIC 18FXXX microcontroller series that supports Multi Media Card (MMC) helped in creating a database for patients that could be used for studies and better calibrations. Its limitation was that users had to look extremely into the direction they intend to move in order to set the wheelchair in motion. And they also had to close their eyelids for three seconds to start and stop the motion alternately thereby focusing the user's eyes and mind on the control of the wheelchair rather than observing other important scenarios in his/her environment. Exposure of the user's eyes to infrared rays had a detrimental effect in a long run. The IR sensor obstructed the view of one of the user's eyes, thereby inhibiting the ability of the user to use his eyes effectively. The system was highly susceptible to wrong input such as user gazing at a specific direction. In another system designed in [10], voice commands was used to control the wheelchair that offered easy mobility to a wide range of physically disabled people. Its shortcoming was in the use of IR sensors for obstacle detection, which was ineffective as it worked on the principle of reflected light waves, which made the detection of darker surfaces very difficult and could not be used in sunlight due to interferences.

Another research team [11] described the development of a brainwave-controlled wheelchair using a Brain-computer Interface (BCI) that enables direct communication between the electrical wheelchair and the brain. The method used in recording the brain activity for controlling the wheelchair was electroencephalogram (EEG) through the use of Emotiv EPOC headset. The EEG signal recorded was transferred to a Personal Computer (PC) software through bluetooth, the PC in turn converted the EEG signal to electrical signal and sent it to the wheelchair for movement commands. The limitations of this project were in the adopted method of control, which required the user's brain to be focused on controlling the wheelchair, and the use of bluetooth for communication among the devices involved in the project. A wheelchair system that was controlled with the movement of the eye was developed by [12]. A camera placed in front of the user was used to achieve the eye method of control by capturing the image of the eye using Haar cascade algorithm and tracking the position of eye pupil using an image processing technique called coordinate system. Also, an ultrasonic sensor was used as the safety mechanism for the wheelchair to detect both static and dynamic obstacles and a central switch was used for emergency stop and call purposes. The limitation of this project was in the adopted method of control, which did not provide the user with the ability to wirelessly control the wheelchair.

Another methodology was introduced by [13]. The quality of the wheelchair designed was in its adopted method of control, which features a voice-controlled mechanism via an android phone and offered easy mobility for users that were suffering from a combination of a range of physical disabilities such as patients with quadriplegia and hearing impairment. It also had an obstacle detection unit achieved with the use of an ultrasonic sensor, which prevented the wheelchair from unnecessary collision due to user or system error. Its limitations were the use of voice-controlled system, which could not be used efficiently in a very noisy environment and, the susceptibility of the wheelchair system to wrong input in cases where the user says one of the common words for operating the wheelchair system, say "GO" in an ordinary statement that's not meant for the control.

According to the work done by [14], the system used a touchscreen-based android application for the control of the wheelchair. It offered the user precise control of the wheelchair and it also featured an obstacle detection sensor that provided easy navigation without colliding with objects in its way. Its shortcoming was in the direction of control achievable by the wheelchair system, which was limited to left, right, forward, backward and stop, directions such as diagonal, 90°, 180° movements were not achievable due to the type of wheel used for the wheelchair system. A joystick-controlled wheelchair was developed in [15]. It offered the physically challenged precise control of the wheelchair system with the use of a joystick. The use of joystick made the joystick-controlled wheelchair system less costly and complex in terms of control. The limitation was in the adopted method of control because the user could control the wheelchair only when in use. This implied that, it could not be controlled by a supporting party in cases where the physically challenged could not use the joystick. Also, the wheelchair could not be controlled at any point in time because the method of control was fixed on the wheelchair.

Other researchers [16] designed a wheelchair that could be controlled with an Android mobile application. The advantage of this project was that it allowed the wireless control of the wheelchair without any physical effort. The limitations of this project were in the choice of mode of wireless communication, which was Bluetooth, having a limited range of ten centimeters. Furthermore, the android mobile application mode of control necessitated the possession of an android phone to be able to control the wheelchair.

In the work of [17], the development of a four-wheel omnidirectional wheelchair and its control using hand gesture recognition-based algorithm was presented. Seven different dynamic hand gestures were mapped with different motions of the wheelchair and were classified by a Dendogram Support Vector Machine (DSVM) classifier using template-based features extracted from initial measurement unit (IMU) and EMG sensors. Dynamic gestures were identified autonomously using accelerometer response. Results obtained from the classifier performance showed an accuracy of 94% during training and 90.5% while controlling the wheelchair using gesture commands. However, the developed system could not be navigated in a constrained environment.

The proposed method in this work is different from the reviewed related works in terms of the method of control of the wheelchair, the degree of movement achievable and the obstacle detection unit adopted for the system. The method of control used for the system is a web application capable of controlling the wheelchair remotely through wireless fidelity communication technology (Wi-Fi) which would not have any detrimental effect as compared to Jain et al. (2015) who used eyeball motion detected by Infrared (IR) sensors as the method of control. Four mecanum wheel are used to achieve the complete degree of movement by controlling each of the wheels independently. The obstacle detection unit of the system comprises an ultrasonic sensor, which uses ultrasonic sound to detect and measure distance of the obstacles. The ultrasonic sensor is not limited in terms of the range of obstacles, it can detect as compared to Bhardwaj et al. (2016) who used IR sensors for obstacle detection unit, which cannot detect darker surfaces.

## Research methods

This system comprises two parts, which are the software and the hardware. The software consists of a web application with two different types of control mechanism (a touch control interface for smartphones and a mouse control interface for laptops and desktops) which are used for controlling the wheelchair and the hardware consists of various units that make up the wheelchair system.

### *System Hardware*

The system hardware was developed using a Raspberry Pi Model B single board computer with other components like Raspberry Pi 3 Model B Camera Module, ultrasonic sensor, Dual H-Bridge Motor Driver Relay interfaced with it.

***System Block Diagram***

The block diagram in Figure 1 shows the integration of different units to accomplish the development of the wheelchair system. The control unit is the Raspberry Pi 3 Model B, a System on Chip (SoC) based on BCM2837 64bit CPU Quad Core 1.2GHz. This is the brain of the system that interconnects all the units and helps in processing the various functionalities that are to be implemented in the system. The kinematics unit is where we have the Ilon wheel connected to a DC motor using a suitable hub. The HC-SR04 ultrasonic sensor is used for the obstacle detection unit. The control unit is powered by a different source from the kinematics unit. The Camera Serial Interface (CSI) on the Raspberry Pi is used for connecting the Raspberry Pi camera for surveillance. The BCM43438 wireless LAN on-board the Raspberry Pi is used for connecting to a router for communicating with the controller device on the same Local Area Network (LAN).

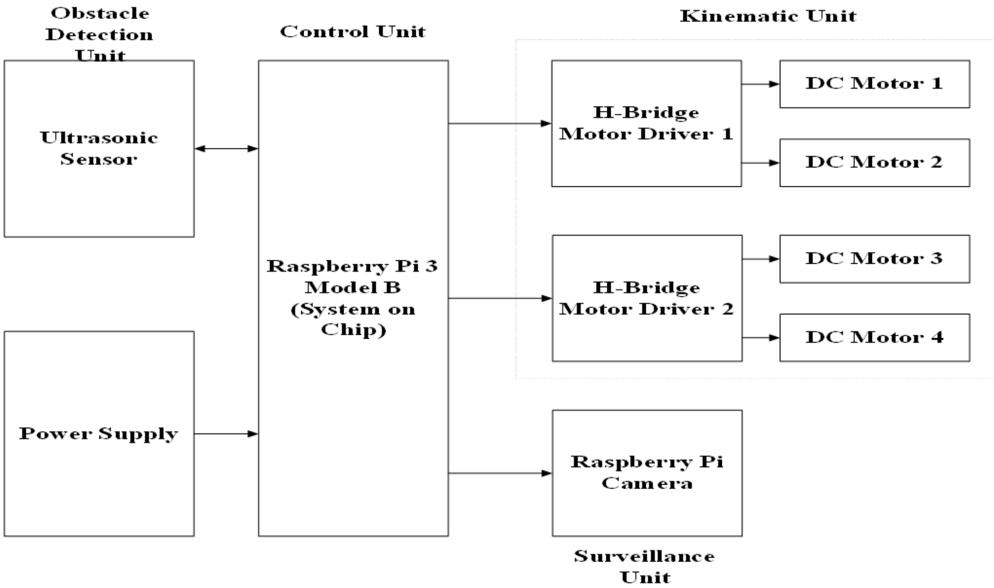



**Figure 1.** The Block Diagram of the System.

***Design Calculation***

The system was powered using 12V and 5V DC battery. The 12V battery was used to power the kinematics unit and the 5V battery was used to power the control unit. Since the control unit powers the obstacle detection, power supply, and surveillance unit, there's no need for separate calculation for them.

***Kinematic Unit Design Calculation***

The power rating, ampere-hour rating and how long the kinematics unit can be continuously operated before it needs to be recharged is calculated using equations 1 to 3:

$$Pmotor = Imotor * Vmotor \qquad (1)$$

Where,

*Pku* = total power for the kinematics unit
*Pmotor* = total power for one DC Motor
*Imotor* = operating current for one DC motor
*Vmotor* = operating voltage for one DC Motor
*Ahku* = Ampere Hour rating for the kinematics unit

*Tku* = time the kinematic unit will last for.

From the datasheet of the 12V DC Motor
*Imotor* = 1.42*A*
*Vmotor* = 12*V*
*Pmotor* = 1.42 * 12
*Pmotor* = 17.04*W*

Since we have four DC Motors, *Pku* will be,
*Pku* = 4 * 17.04
*Pku* = 68.16*W*
*Pku* ≈ 68*W*

Calculating the ampere hour rating for the kinematic unit

$$Ampere\ hour = \frac{power * time}{voltage} \tag{2}$$

$$Ahku = \frac{68 * Tku}{12}$$

$$Ahku = 5.7Tku\ Ah \tag{3}$$

A 2800mAh 12V battery will power the kinematic unit for *Tku* hours

2800*mAh* = 5.7*Tku Ah*
*Tku* = 0.49 *hrs*
*Tku* = 0*hrs* 29*mins*

**Control Unit Design Calculation**

The power rating, ampere hour rating and how long the control unit can be continuously operated before it needs to be recharged is calculated using equations 2, 4 and 5:

$$Pcu = Icu * Vcu \tag{4}$$

Where:
*Pcu* = total power for the control unit
*Vcu* = operating voltage of the control unit
*Icu* = operating current of the control unit
*Ahcu* = Ampere Hour rating for the control unit
*Tcu* = time the control unit will last for.

From the datasheet of Raspberry Pi 3 Module B (https://www.raspberrypi.org)
*Icu* = 2.1*A*
*Vcu* = 5*V*
*Pcu* = 2.1 * 5
*Pcu* = 10.5*W*

Calculating the ampere hour rating for the control unit from equation 2:

*Ahku= (10.5 * Tcu)/5*

*Ahku = 2.1Tcu Ah* (5)

*A 5000mAh 5V battery will power the kinematic unit for Tcu hours*

*5000mAh = 2.1Tcu Ah*
*Tcu = 2.38hrs*
*Tcu = 2hrs 23mins*

**The System Flowchart**

The system flowchart in Figure 2 shows the flow of the sequence of steps in operating the wheelchair system. The sequence of steps is stated below:

1. Start
2. Initialize a Wi-Fi connection using a router.
3. Execute the controller program on the Raspberry Pi single board computer.
4. Access the web application using an internet enabled device with a browser.
5. System executes stop() command and waits for further command.
6. Check if command is received.
7. If a command is received, then check if it is a forward () command Else go to step 6.
8. If the command received is forward (), then check if the distance to an obstacle is greater than 30cm. Else move the system according to the command received.
9. If distance to an obstacle is greater than 30cm move the system according to forward command. Else go to step 6.
10. Stop.

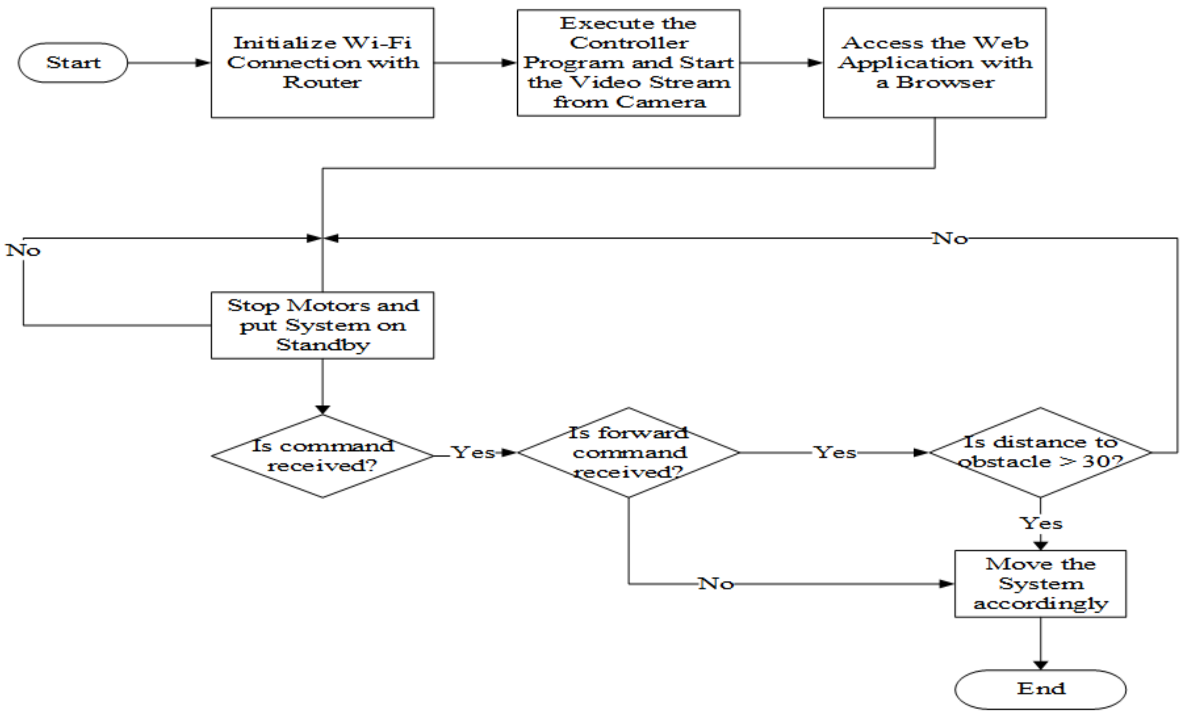



**Figure 2.** The System Flowchart.

**Hardware Design Consideration**

Hardware design includes the construction of the wheelchair chassis and several units, namely, Control Unit, Kinematics Unit, Obstacle Detection Unit, Power Supply Unit, Surveillance Unit and, Communication Unit.

### Control Unit

The Raspberry Pi 3 Model B is used as the brain of the system because of its high processing power as compared to other controllers of its size. The Lite Version of the Raspbian OS is operated using a command line interface (CLI) as shown in Figure 3 because it has no support for GUI, therefore, all the instructions carried out on the system were done through a Secure Shell network protocol (SSH). The RPi. GPIO python library was used in controlling general purpose input output (GPIO) pins of the Pi.

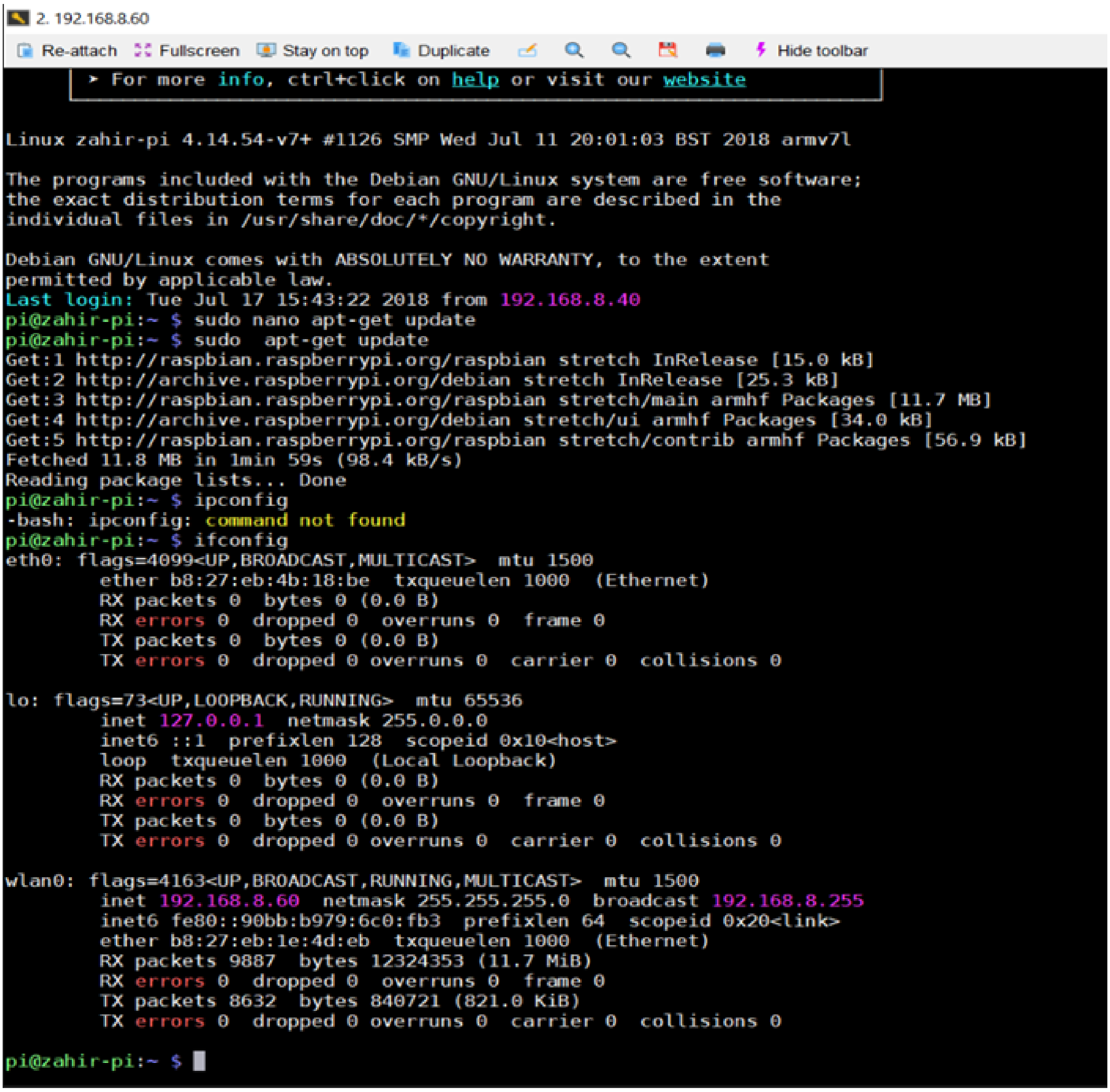



**Figure 3.** The screenshot of the Terminal Window.

A Full H-Bridge L298N motor driver module, which has two channels for controlling two DC motors simultaneously, drives the DC motors and it can deliver a maximum current of two (2) Amps to each of the channels. The four wheels rotate clockwise or anticlockwise based on the code written on the control unit to achieve the omnidirectional movement. Figure 4 shows the direction of rotation of each of the wheels to achieve an omnidirectional motion.

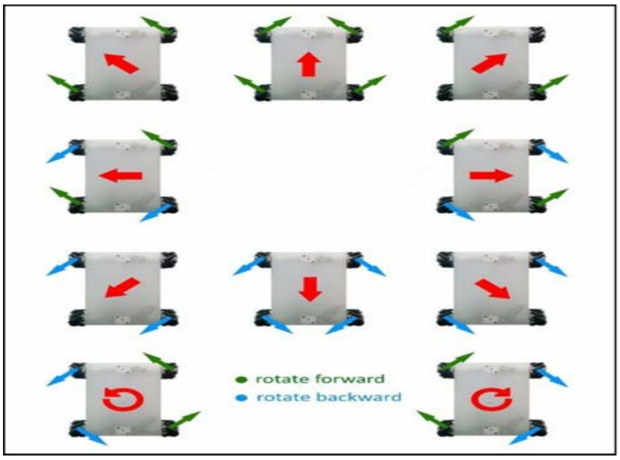



**Figure 4.** Achievable motions by the wheelchair [18].

**Kinematics Unit**

The kinematics unit was designed using four mecanum wheel motorized independently by four 12V DC motors. The DC motor has an operating current of 700mA and stall current of 1400mA.

**Obstacle Detection Unit**

The obstacle detection unit comprises of the HC-SR04 Ultrasonic Sensor shown in Figure 5. The ultrasonic sensor measures distance to a target by measuring the time of flight (TOF) of the emitted ultrasonic wave. The ultrasonic sensor (as used in this case) is configured to detect obstacle within a 30cm range along the wheelchair's path and it is mounted at the front of the wheelchair.


**Figure 5.** Ultrasonic Sensor (HC-SR04) [19].

The ultrasonic sensor continuously sends the distance measured to the control unit, if the control unit receives a signal that is within the specified detection range, the control unit temporarily takes control of the wheelchair by stopping all the DC motors until it receives a measured distance that is greater than 30cm [20].

**Power Supply Unit**

Two separate batteries were used in powering the wheelchair system. The 12V Li-Po battery was used to power the four DC motors through the L298N motor driver while the 5V battery was used to power the control unit.

The 12V battery was used for the DC motors because it can supply enough current to the motors as required. A common ground was established between the two power sources to prevent potential electrical hazards.

**Surveillance Unit**

The Raspberry Pi 5MP, v1.3 camera shown in Figure 6 was used in the surveillance unit. It was connected to the camera serial interface of the control unit via a 15 Pin Ribbon Cable to the dedicated 15-pin MIPI Camera Serial Interface (CSI) specifically designed for interfacing cameras to the Raspberry Pi. The feed from the Pi camera is displayed on web application to aid the control of the wheelchair as the users will not necessarily need to look away from the controller while operating the device. The live feed from the web application set at a spatial resolution of 640x480 and 15 frames per second (480p @ 15fps).

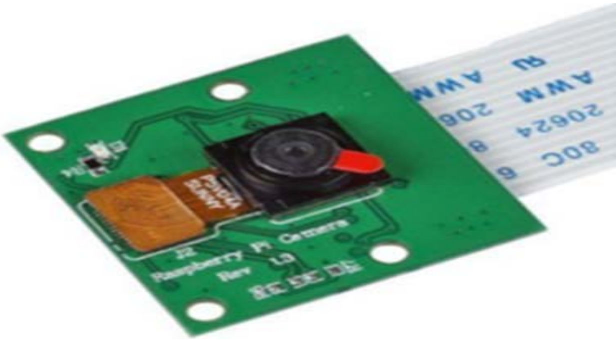


**Figure 6.** Raspberry Pi 5MP Camera.

**Communication Unit**

Communication between the wheelchair and the controlling device is through Wireless Fidelity Technology (Wi-Fi). The inbuilt Wi-Fi chip BCM43438 on the control unit was used to connect to a router (home network) and the Wi-Fi enabled controlling device is also connected to the same network as the control unit. The Raspberry Pi was configured to automatically connect to the router on boot up by setting a static IP address for the Pi. The procedure followed in setting up the Wi-Fi connection are as follows:

1. SSH into your Raspberry Pi with the correct credentials.
2. Input the command sudo nano /etc/dhcpd. conf at the CLI terminal
3. Move your cursor to the bottom of the script and add the following lines of code:
```
# for wired connection
interface eth0
static ip_address=192.168.8.60/24
static routers=192.168.8.1
static domain_name_servers=192.168.8.1
#for wireless connection
interface wlan0
static ip_address=192.168.8.60/24
static routers=192.168.8.1
static domain_name_servers=192.168.8.1
```
4. Save the file with Ctrl + o and exit the nano text editor with Ctrl + x.
5. Reboot the Raspberry Pi with sudo reboot.
6. SSH into the Pi again and input the command ip a to check for the assigned IP address. This shows the IP address that has been previously assigned to eth0 and wlan0. So anytime your Pi boots up to the same network, it always has the assigned static IP address.

**Software Design Consideration**

The software design is concerned with the development of a web application for controlling the wheelchair. The web application developed was hosted locally on a Lighthttpd web server installed on the Raspberry Pi. The Lighthttpd web server handles the request from the web application. The communication protocol between the web application and the web server is both HTTP and FastCGI (Fast Common Gateway Interface). The HTTP is used for getting the live feed from the Pi Camera while the Fast CGI is a binary protocol used for sending movement commands to the Raspberry Pi. The web application is created in such a way that it will be possible to control the wheelchair by using either a mobile phone or a laptop.

**Touch Control Interface**

The touch control designed for phone control where the user needs to press the desired button direction to move the wheelchair and then press the stop button to halt the wheelchair.

**Mouse Control Interface**

The mouse control was designed for laptop or desktop control where the user needs to press and hold on the desired button direction to move the wheelchair and then release the button to halt the wheelchair.

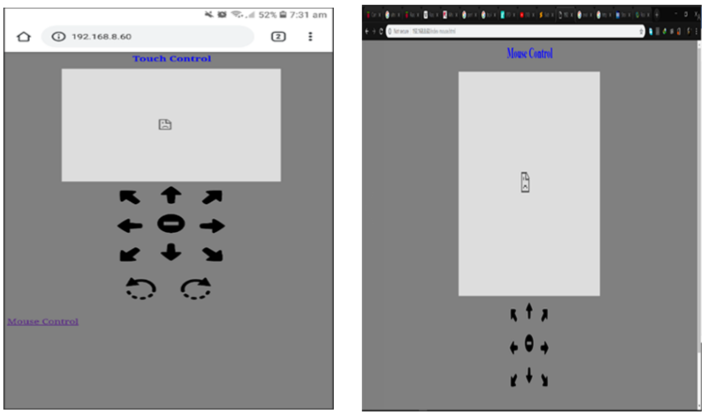
**Results and Discussions**

The developed prototype wheelchair system is shown in Figure 7.

**Figure 7.** The Developed System.

The chassis of the wheelchair system was constructed using aluminium and ABS Plastic. The control program for the system was developed using Python and Shell Script Code written for the Raspberry Pi to control the remaining modules. The touch control and mouse control version of the web application shown in Figures 8 was developed using HTML integrated with JavaScript.

The average detection accuracies for brick wall, metal, and wood are 87.37%, 94.43%, and 83.57% respectively as shown in Table 1. The accuracy when measured against metal is very high relative to wood and wall.




a) b)

**Figure 8.** Web application: (a) Touch control, and (b) Mouse Control.

*Table 1*

**Accuracy of the Obstacle Detection Unit with Different Materials**

| S/N | Wood (cm) | Metal (cm) | Wall (cm) |
|---|---|---|---|
| **1** | 23.5 | 29.1 | 24.5 |
| **2** | 24.7 | 28.2 | 25.1 |
| **3** | 23.3 | 29.5 | 26.5 |
| **4** | 24.5 | 28.1 | 25.7 |
| **5** | 25.1 | 27.9 | 24.9 |
| **6** | 24.8 | 28.2 | 26.0 |
| **7** | 23.9 | 27.8 | 27.9 |
| **8** | 25.4 | 29.6 | 27.1 |
| **9** | 28.0 | 27.7 | 26.7 |
| **10** | 27.5 | 27.2 | 27.7 |

The graph shown in Figure 9 implies that the obstacle detection unit of the system which is an ultrasonic sensor easily detects metal as compared to wood and brick wall. This means using the developed system in areas with more metallic object is safer.

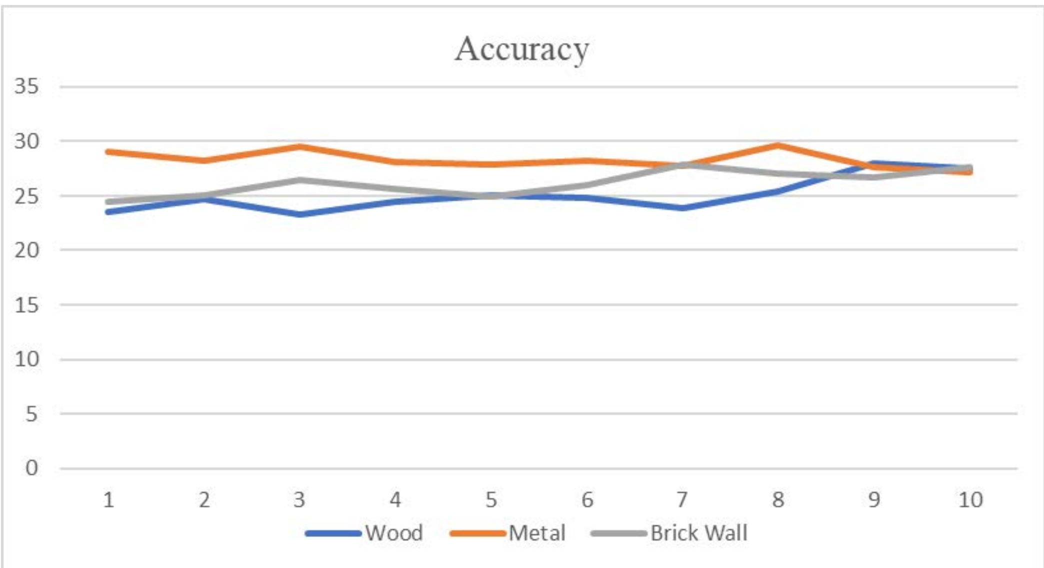



**Figure 9.** Graph comparing the accuracy of the obstacle detection unit with different materials.

The average response time of the wheelchair to movement commands from the web application was 1.04 seconds according to Table 2.

*Table 2*

**Response time of system to movement commands from web application**

| S/N | Response Time (sec) |
|---|---|
| 1 | 1.2 |
| 2 | 1.0 |
| 3 | 0.8 |
| 4 | 1.3 |
| 5 | 1.1 |
| 6 | 0.9 |
| 7 | 1.4 |
| 8 | 0.8 |
| 9 | 1.0 |
| 10 | 0.9 |

The graph shown in Figure 10 shows response time of the system to commands from the application. This implies that the developed system can be used in situations where quick movements is an important factor.

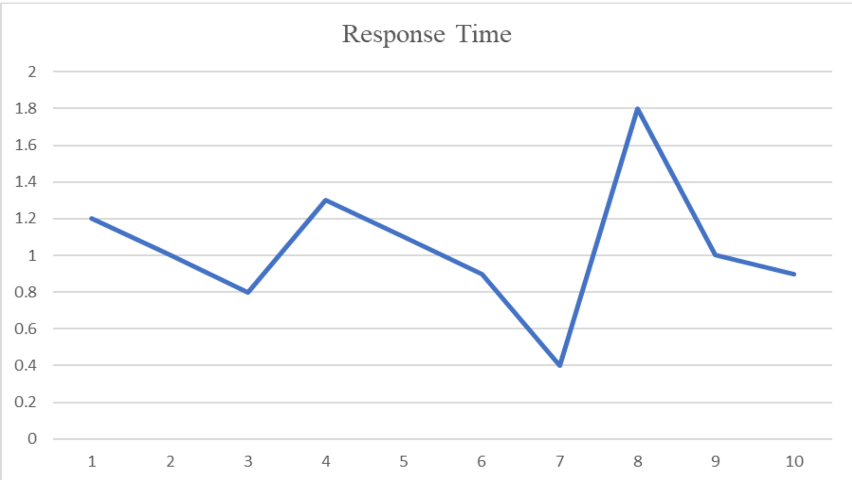



**Figure 10.** Graph showing the response time of the wheelchair system to movement commands.

The estimated cost in American dollar for the developed system is shown in Table 3. The total estimated cost for both the hardware and software is **$ 352.27.**

*Table 3*

**Bill of Engineering Measurement and Evaluation**

| S/N | Components | Cost ($) |
|---|---|---|
| 1 | Wheelchair Chassis Material | 28.2 |
| 2 | Raspberry Pi 3 Model B Full Kit | 38.7 |
| 3 | Raspberry Pi Camera Module | 7.69 |

*Continuation Table 3*

| | | |
|---|---|---|
| 4 | Ultrasonic Sensor (HC-SR04) | 1.28 |
| 5 | 12V Battery | 24.4 |
| 6 | Dual H-Bridge Motor Driver Relay (L298N) | 3.85 |
| 7 | 12V DC Motor | 69.7 |
| 8 | Mecanum Wheel with Hub | 74.87 |
| 9 | Motor Mounting Bracket | 17.95 |
| 10 | 16 GB Micro SD Card | 6.4 |
| 11 | Software Development | 50 |
| 12 | Miscellaneous | 19.23 |
| Total | | 352.27 |

**Conclusion**

This paper has successfully presented a smart omnidirectional wheelchair for the monoplegic, hemiplegic, and paraplegic. The system designed allows the user to easily manoeuvre in what would otherwise be an extremely complicated environment. The result of the use of the mecanum wheel results in the achievement of much higher driving accuracy and greatly improved overall experience for the user in terms of ease of use. The use of ultrasonic sensor helped in enhancing the safety of the wheelchair.

**Future directions**

There are lots of improvements that can be made on the current design and technology and lots of additional features can be added:

- Subsequent system should add shock absorbers to the wheels of the wheelchair to reduce vibrations while it is moving.
- Three additional ultrasonic sensors can be used so that the wheelchair can detect obstacle in all directions.
- New mechanism like using pan tilt with the camera's system will allow the user to have different angles of view when controlling the system.
- Addition of security to the web application by creating a login page to enhance safety and to restrict unauthorized users from controlling the wheelchair.

**Acknowledgements**

This work was supported by the Department of Computer Engineering, Federal University of Technology, Minna, Nigeria. The authors appreciate the management of the university for providing adequate research laboratory.

**References**


1. Sankardoss V., Geethanjali P. Design and low-cost implementation of an electric wheelchair control. In: *IETE Journal of Research, 2019*, 20, pp. 1-9.
2. Leaman, J, La H.M. A comprehensive review of smart wheelchairs: past, present, and future. *In: IEEE Transactions on Human-Machine Systems*, 2017 Jun 7;47(4), pp. 486-99.
3. Rathore D.K., Srivastava P., Pandey S., Jaiswal S. A novel multipurpose smart wheelchair. In: *2014 IEEE students' conference on Electrical, Electronics and Computer Science*, 2014 Mar 1 (pp. 1-4). IEEE.
4. Nirmal T.M. Wheelchair for physically and mentally disabled persons. In: *International Journal of Electrical and Electronics Research,* 2014, 2, pp. 112-118.
5. Lodhi D.K., Vats P., Varun A., Solanki P., Gupta R., Pandey M.K., Butola R. Smart electronic wheelchair using Arduino and Bluetooth module. In: *International Journal of Computer Science and Mobile Computing,* 2016, 5(5), pp.433-438.
6. K. D-Healthcare-Company. Affordable, foldable motorized wheelchair, http://kdsmartchair.com, Feb. 2015.
7. Swapna P., Sharmila B., Dharshan Y. Electric wheelchair for physically challenged. In: *International Research Journal of Engineering and Technology,* 2016, 3, pp. 1090-1094.
8. Nasri Y., Vauchey V., Khemmar R., Ragot N., Sirlantzis K., Ertaud J.Y. ROS-based autonomous navigation wheelchair using omnidirectional sensor. In: *International Journal of Computer Applications,* 2016, 133(6), pp.12-17.
9. Bhardwaj R., Gupta P., Jadhav P., Kadam B., Kedari A. Android based automated smart wheel. In: *International Journal of Innovative Research in Computer,* 2016, 4(3), pp. 3040-3047.
10. Jain M., Puri S., Unishree S. Eyeball motion-controlled wheelchair using IR sensors. World Acad. Sci. Eng. Technol. Int. J. Comput. Electr. Autom. Control Inf. Eng, 2015, 9(4), pp.906-909.
11. Swee S.K., Kiang K.D.T., You L.Z. EEG controlled wheelchair. In: *MATEC Web of Conferences, EDP Sciences,* 2016, 51, p. 02011.
12. Jafar F., Fatima S.F., Mushtaq H.R., Khan S., Rasheed A., Sadaf M. Eye controlled wheelchair using transfer learning. In*: 2019 International Symposium on Recent Advances in Electrical Engineering (RAEE), IEEE,* 2019, 4, pp. 1-5.
13. Dey P., Hasan M.M., Mostofa S., Rana A.I. Smart wheelchair integrating head gesture navigation. In: *2019 International Conference on Robotics, Electrical and Signal Processing Techniques (ICREST), IEEE,* 2019, pp. 329-334.
14. Singandhupe R.B., Sethi R.R. Automated wheelchair using android technology. In: *Imperial Journal of Interdisciplinary Research,* 2016, 2, pp. 654-657.
15. Saharia T., Bauri J., Bhagabati C. Joystick controlled wheelchair. In: *Int. Res. J. Eng. Technol*, 2017, 4, pp.235-237.
16. Patel M.D., Lakhapati S., Chavan R., Pathak A. Android controlled wheelchair. In: *International Journal for Research in Emerging Science and Technology,* 2017, 4(1), pp. 16-19.
17. Luo X., Amighetti A., Zhang D. A human-robot interaction for a mecanum wheeled mobile robot with real-time 3d two-hand gesture recognition. In: *Journal of Physics: Conference Series,* 2019, 1267(1), p. 012056.
18. Autodesk, Inc. (2015, April 9). Mecanum Wheel Robot - Bluetooth Controlled. Retrieved April 3, 2018, from Instructables: http://www.instructables.com
19. Yalung C.N., Adolfo C.M.S. Analysis of obstacle detection using ultrasonic sensor. In: *International Research Journal and Engineering Technology*, 2017, pp.1015-1019.
20. Adhikary P., Gupta N., Halder A., Deyasi A. Design of smart system for detecting proximity of obstacles at acute and obtuse angles. In: *2019 3rd International Conference on Electronics, Materials Engineering & Nano-Technology (IEMENTech), IEEE,* 2019, pp. 1-4.
21. How to Mechatronics (May, 2015), Arduino DC Motor Control Tutorial – L298N | PWM | H-Bridge. Retrieved from How to Mechatronics: https://howtomechatronics.com/tutorials/arduino/arduino-dc-motor-control-tutorial-l298n-pwm-h-bridge/, April, 2018.