Multimode Control of Wheeled Bot Using Python and Virtual Network Computing

Rajesh Kannan Megalingam, Santosh Tantravahi, Hemanth Sai Surya Kumar, Nagasai Thokala, Hari Sudarshan Rahul
Department of Electronics and Communication Engineering,
Amrita Vishwa Vidyapeetham , Amritapuri Campus,
Kollam, India
rajeshkannan@ieee.org, tantravahi007@gmail.com, suryahemanth9@gmail.com, nagasaichowdary@gmail.com,
puramrahul5@gmail.com

***Abstract*—In this paper the design and development of multiple control interfaces of a mobile wheeled bot is presented. The controller is a Raspberry Pi Zero and the software platform is developed using Python. The bot has been programmed using Python and Virtual Network Computing(VNC) viewer. The multiple control interfaces include keyboard, Graphical User Interface (GUI) in desktop computer and VNC in mobile phone. The communication between the bot and the controller is wireless communication through a local network. One of the multiple control interfaces can pass the commands to move the robot. Experimental setup included five different paths in which the robot is made to move using each of the control methods. The response timings measured are noted and evaluated.**

Keywords— mobile robot, Python, virtual network computing, multi-mode control

I. INTRODUCTION

For several years humans have made use of mobile robots to manipulate many things and to reach places where humans physically cannot reach. Mobile robotics is an emerging field and a lot of research is being carried out in many parts of the world. Mobile robots can be made completely autonomous or partially controllable or fully controllable depending on the need. Mobile robots are also used in many space stations, research laboratories etc. There can be several types of mobile robots and they can be controlled using many different methods. They are designed depending upon the need and the conditions under which they have to work. These kind of robots can be made to withstand very harsh environments. Mobile robots can have different size based on the need for the hour and function which they perform. These robots have come in very handy and have opened up new possibilities.

The mobile robot developed and the multiple control interfaces designed, on which this paper is written is a simple movable device which is made of very low cost materials. The low cost single board computer chosen was Raspberry pi zero which can support wireless communication. The programming environment is PYHTON IDLE on which the controller software and the keyboard and GUI based software are built. This project also uses VNC viewer for mobile phone applications.

We use three modes of control: laptop keys, mobile phone, and GUI. This research works determines, out of the chosen three modes of control which is the best. This is because there are situations which require speed and efficiency of the bots. Keeping in mind that the time required to catch up the maximum speed of the bot is almost a constant. The test conducted give the speed with which commands are passed and executed by the single board computer. The experiments have been carried on fixed path lengths and the time taken by the bot to cover the distance has been measured. The results give the best method of usage.

II. MOTIVATION

Mobile robotics has been very handy over the years and it continues to be up to the present time. They reach places which humans cannot and explore harsh terrains. Their uses can be found from common household to big space research labs. They can be used for a variety of applications, for example, can be used to carry research/surgery equipment, goods in industry, sensors in areas where humans can't reach, in mines etc. Their control is very essential and have several applications. Some situations require speed and others, efficiency. A combination of different sensors when integrated with the bot can increase the productivity and functionalities of the bot. The bot can be made to follow a fixed path or move in an environment guided by the sensory inputs. Generally, a terminal is required to control the bot, unless it is fully automated.

III. RELATED WORKS

There are several aspects to mobile robots and they differ in the implementation based on applications: indoor, outdoor, surveillance, sensing, identifying etc. Simulation of a three stage torque controlled wheeled mobile robot based on trajectory tracking and non-holonomic concept is proposed in paper [1]. IoT based, remote monitored spy robot platform for surveillance is proposed in paper [2]. The authors claim that this Raspberry Pi based platform with night vision camera can be customized depending on the application. A mobile, omni wheeled robot, using simultaneous localization and mapping (SLAM) algorithms is presented in paper [3]. The authors explain that this kind of robot is suitable for indoor applications. Design and development of mobile robot for transportation, gripping and identification based on machine vision is discussed in [4]. The researchers use Raspberry Pi for image processing. But the control interface mechanism in not clear.

Single control interface based, mobile robot with Swedish wheels considering the wheel's slipping is presented in the research article [5]. A model predictive controller using nonlinear time-varying systems is used in the robot for tracking control and stabilization of a wheeled mobile robot based on non-holonomic concepts [6]. Only simulation results are available and the control interface is not discussed. A single control interface, remote control based mobile robots in unknown environment using wireless sensor technology is explained in research paper [7].

In this research article [8], the authors concentrate on how to avoid singularity issue in over-actuated, pseudo omnidirectional, wheeled mobile robot rather than the control interface, where as our intention is to figure out which control interface would be better in such mobile robots. In yet another paper [9], the authors use Robot Operating System (ROS) based motion planning for omni-directional wheeled mobile robot rather than the control interface. Design and implementation of two wheeled robot to tilt up and down using autonomous balance mechanism is presented in [10]. An orthotic arm controlled by a GUI is presented in [11].The control of a wheel chair based on Raspberry Pi platform is discussed in [12]. As mentioned earlier mobile robot design includes several aspects of mechanical design, algorithms used, technologies implemented etc. based on applications.

IV. System Architecture

The system architecture comprises of three major blocks: the command block, controller block, and the mobility block. The command block comprises of the three interface units: keyboard, mobile phone and GUI. Keyboard of any computer in which our software platform is installed, can be used. We use the VNC viewer installed in the mobile phone as the second control interface. The third control interface is the GUI installed in any computer or laptop. We use the WiFi communication interface for the commands from the command block to control the bot. The controller block consists of the single board computer and interfaces to WiFi, motor driver etc. The mobility block comprises of the motors, motor drivers and the wheels. In all of the control methods, we use Python as the programming language to send commands from micro-controller to motor driver. Fig. 1 shows the system architecture diagram.

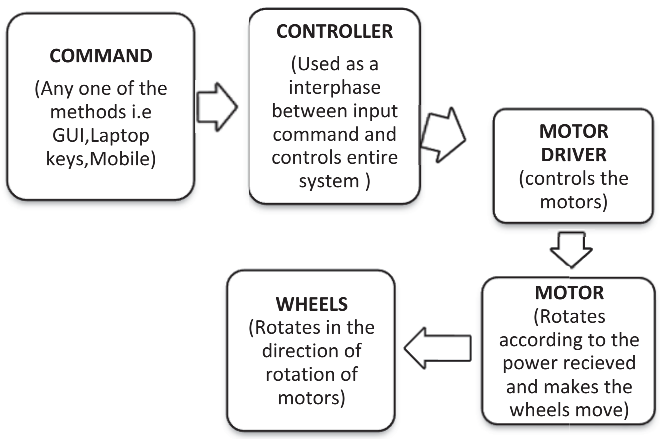

Fig. 1. Multi-Mode Robot Control System Architecture

A. COMMAND BLOCK

GUI (Graphical User Interphase) is a type of interface that allows the user to interact with other devices through graphical icons or buttons and send the required command through them. We have developed this GUI in Python which consists of five buttons for the five commands: up, down, left, right and stop. Up and down buttons are used to command the robot to move forward and backward respectively. Left button is used to control the robot to take left turn and right button to take right turn. Stop button can be used anytime to stop the robot.

In case of mobile control VNC viewer app is used. VNC viewer app is one of the ways to control the wheeled bot using mobile phone with WiFi. In order to start a remote-control session, run VNC Viewer and identify VNC Server on the host computer (Raspberry pi) one wants to control. Once authenticated, VNC Viewer displays the host computer's desktop in a new window, and can take control using the device touch. This is in a way sharing the PC desktop and controlling the robot. In this case, the host computer or laptop should be running the VNC server on the host computer.

Keyboard of a desktop PC or laptop can be used to send command to the bot through which we can control the direction of the bot. Keys i, j, k, l and s are used for controlling the robot to move in different directions. Keys i, j, k, and l moves the robot in forward, left, backward and right directions and key s is used to stop the robot.

B. MOBILITY BLOCK

Mobility block consists of the motors, motor drivers and wheels. DC motors are always preferred over stepper motors because of their speed, weight, size, cost. depending on the applications. In our case where our main object is to experiment with various modes of control, we want the robot to be as compact and light weight as possible. In addition, a DC motor’s speed can be changed by using a variable supply voltage or PWD of the controller.

C. CONTROLLER BLOCK

The controller block interfaces with the WiFi, decodes the commands and issues control signals to the respective modules to make the robot move. It also keeps up the timing and sequence in which the control signals are generated. It takes decisions based on the commands to control the robot movement in any direction. It can take commands from any of the three interfaces.

V. DESIGN AND IMPLEMENTATION

The design and the implementation of the robot is shown in the Fig. The figure shows the GUI, a mobile phone with VNC and a keyboard. The bot consists of the chassis, single board computer, motor driver, motors, wheels, control circuitry and power supply unit. The multimode control interfaces include GUI, a mobile phone and a keyboard.

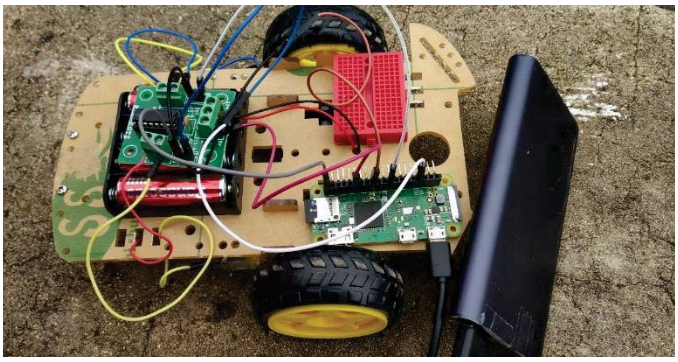

Fig. 2. The robot with the laser cut chassis and control circuitry

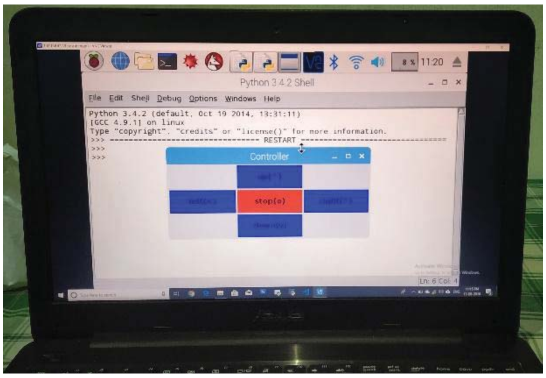

Fig. 3. Python based GUI on a laptop

Fig. 4. VNC base Mobile Phone Control

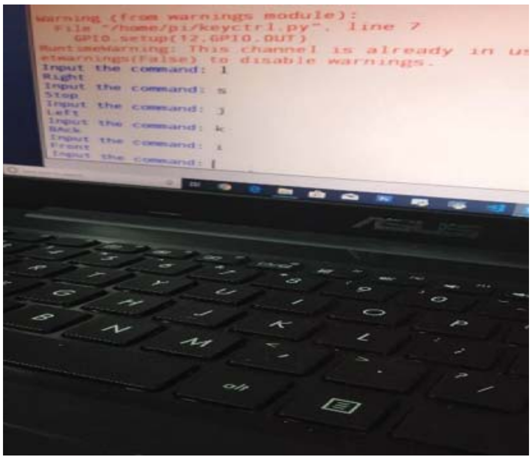

Fig. 5. Keyboard based Control

A. ROBOT CHASSIS:

The robot chassis is made of laser cut cardboard of dimensions 8 inches by 5 inches which have provisions to fix the motors and castor wheel along with providing space for the circuitry, single board computer and power supply unit. We can fit two DC motors (BO -battery operation) to the body. The speed of the body depends on the load it can carry. If we use huge battery pack for the power supply, then the speed of the bot reduces. The robot chassis along with the circuitry is shown in Fig. 2.

B. SINGLE BOARD COMPUTER:

We use the single board computer, Raspberry pi Zero micro-controller which has multicore processor, GPU, ROM and I/O peripherals inside it in the form of System-on-Chip (SoC). It is a low cost basic computer with 1GHz single-core CPU, 512MB RAM, Mini HDMI port, Micro USB OTG port, Micro USB power, GPIO(General purpose Input / Output pins) as its major components. It can run on many Operating systems but “Raspbian” is the official OS used.

C. MOTOR DRIVER:

Motor driver act as an interface between Micro-controller and the motors. The most commonly used motor driver IC’s are from the L293 series such as L293D, L293NE, etc. These ICs are designed to control 2 DC motors simultaneously. With the use of motor driver, we can control the speed of motor; we can control the direction of rotation. The L293D is designed to provide bidirectional drive currents up to 600-mA at voltages from 4.5 V to 36 V.

D. KEYBOARD, GUI AND MOBILE PHONE:

Any standard keyboard with PC or a laptop key board can be used as one of the control mechanisms. This is shown in Fig. 5. Similarly, any desktop computer or laptop in which the GUI is installed can be used to control the bot. All Android based phones that can support VNC can be used for the third control mode. The GUI we developed for controlling the bot using Python is shown in Fig. 3. It is a simple graphical interface which can be used to control the bot in any direction with much ease. VNC viewer app is used with the mobile phone to remote control the robot which is shown in Fig.4. This requires another PC as the VNC server in which the VNC viewer is run to display the host computer’s desktop on the mobile phone so that not only screen sharing takes place but also the control of the device. This way, the bot is controlled via the same GUI used as one of the control modes, but through the mobile phone.

Working of the bot developed in this project is based on single board computer, Raspberry pi Zero board, which is the main controller of the bot. Wheels connected to the motors are controlled using motor driver, which controls the power of wheels depending on the command received from the single board computer. Commands are issued by one of the modes: GUI or mobile phone or keyboard. Wheels are powered by external source connected to this motor driver. Similarly, another external power source is used for Raspberry pi. Depending on the commands sent through the terminal, single board computer notifies the motor driver to regulate the power supplied to wheels and make the bot move accordingly.

VI. EXPERIMENTS AND RESULTS

The experiment is done on a smooth plane surface. The surface is a rectangular plane with length 2.04 meters and breadth 1.2 meters. The tests were carried out in a room and it was ensured that the plane was smooth. This place was chosen so that the bot works in a close to one of many practical environments and not an ideal one. All the tests were carried out on the same plane and the path followed by the bot was highlighted using a tape. We created five different paths for the bot to traverse: straight, straight-right, straight-left, U-turn and rectangle. In all of these paths 10 trails were conducted, using each of the three modes: GUI, Mobile phone and Keyboard. This gives us a total of 30 trails for each of the different paths. The taken during each of these trails are tabulated and compared to find out which of the three modes is the fastest.

TABLE I. STRAIGHT LINE PATH

Trail no	GUI (sec)	Mobile (sec)	Laptop keys (sec)
1	5.1	5.1	5.0
2	5.3	5.2	5.2
3	5.3	5.1	5.0
4	5.4	5.2	5.3
5	5.1	5.4	5.5
Mean	5.34 sec	5.23 sec	5.13 sec
Mean RPM:	1.98 RPSEC	2.02 RPSEC	2.06 RPSEC

Table 1 shows the timings list for the robot to cover a straight line path of 2 m in each of the three control modes. The revolutions per second (RPSEC) is almost constant, but the slight different arises due to the passing of commands from the terminal. Table 2 shows the straight-right path timings of the bot using each of the three control modes. In this case, the keyboard control timings are better compared to the GUI and mobile phone control modes. The distance covered is 2.04 m in straight line and 1.2 m in right direction. In case of straight-left path the distance covered is 2 m in straight line and 1.2 m in left direction. In this case mobile phone mode is the best compared to the other two operating modes. Table 4 shows the timings of the U turn path. Here GUI mode seems to offer best timings. Distance covered is 2 m straight up and down and breadth is 1.02 m. In case of rectangular path timings shown in Table 5, keyboard control mode offers better timings compared to the GUI and the mobile phone control modes.

TABLE II. STRAIGHT-RIGHT PATH

Trail no	GUI (sec)	Mobile (sec)	Laptop keys (sec)
1	8.4	8.2	8.2
2	8.6	8.5	8.51
3	8.7	8.2	8.65
4	9.0	8.5	8.2
5	8.8	8.4	8.2
Mean	8.7 sec	8.3 sec	8.24 sec

TABLE III. STRAIGHT-LEFT PATH

Trail no	GUI (sec)	Mobile (sec)	Laptop keys (sec)
1	9.4	8.6	9.0
2	9.5	8.7	8.9
3	9.5	8.2	9.1
4	9.0	8.7	9.2
5	9.5	8.5	9.0
Mean	9.38 sec	8.5 sec	9.04 sec

TABLE IV. 'U' TURN PATH

Trail no	GUI (sec)	Mobile (sec)	Laptop keys (sec)
1	13.4	14.8	13.3
2	13.3	14.9	13.3
3	13.0	14.9	13.3
4	13.4	14.4	13.4
5	13.3	14.6	13.4
mean	13.26 sec	14.72 sec	13.34 sec

TABLE V. RECTANGULAR PATH

Trail no	GUI (sec)	Mobile (sec)	Laptop keys (sec)
1	17.2	17.6	16.4
2	17.6	17.5	16.2
3	17.2	17.3	16.5
4	17.4	17.7	16.4
5	17.6	17.6	16.3
mean	17.4 sec	17.54 sec	16.36 sec

From the experiments, it is observed that all the methods are equally efficient, but as the complexity of the path increases, the number of commands from the terminal increases and therefore laptop keys turn out to be more efficient compared to others.

VII. FUTURE WORKS

The basic bot developed in this project has a huge scope when scaled up or used as is for such applications where the multimode control is required. The bot can be equipped with cameras and sound sensors can be used for surveillance and vigilance purposes. A small robotic arm can be fixed on the bot to make it hold something and can be used to shift things from one place to another. Using a stronger material for body can improve the rigidity of the bot. Using powerful motors can improve the speed and working of the bot. The bot can be instilled with self-learning algorithms so that it can work in harsh and tough environments without any human aid.

VIII. CONCLUSIONS

The mobile bot has been successfully built with the aid of Python and VNC viewer and it is operational. Three (3) different modes of control have been used to control the bot: keyboard, GUI, mobile phone. Based on the tests conducted and results obtained it has been found that out of the three modes of control, operation using laptop keys is found as the most efficient.A negligible tolerance arises due to slight wobbling of wheels.

ACKNOWLEDGMENT

We are indebted to the HuT labs and Electronics and Communication Engineering department at Amrita Vishwa Vidyapeetham, Amritapuri Campus for their constant support and guidance for this work.

REFERENCES

[1] V. Gupta, N. Bendapudi, I. N. Kar and S. K. Saha, "Three-stage computed-torque controller for trajectory tracking in non-holonomic wheeled mobile robot," 2018 IEEE 15th International Workshop on Advanced Motion Control (AMC), Tokyo, 2018, pp. 144-149. doi: 10.1109/AMC.2019.8371077

[2] G. O. E. Abdalla and T. Veeramanikandasamy, "Implementation of spy robot for a surveillance system using Internet protocol of Raspberry Pi," 2017 2nd IEEE International Conference on Recent Trends in Electronics, Information & Communication Technology (RTEICT), Bangalore, 2017, pp. 86-89. doi: 10.1109/RTEICT.2017.8256563

[3] K. Krinkin, E. Stotskaya and Y. Stotskiy, "Design and implementation Raspberry Pi-based omni-wheel mobile robot," 2015 Artificial Intelligence and Natural Language and Information Extraction, Social Media and Web Search FRUCT Conference (AINL-ISMW FRUCT), St. Petersburg, 2015, pp. 39-45.doi: 10.1109/AINL-ISMW-FRUCT.2015.7382967

[4] J. D. Lee, Y. H. Wu, Y. J. Jhao, L. Y. Chen and H. I. Chen, "Development of mobile robot with vision inspection system and three-axis robot," 2018 3rd International Conference on Control and Robotics Engineering (ICCRE), Nagoya, 2018, pp. 6-10. doi: 10.1109/ICCRE.2018.8376424

[5] L. Labakhua, I. Martins and M. Igor, "Control of a mobile robot with Swedish wheels," 2017 IEEE International Conference on Power, Control, Signals and Instrumentation Engineering (ICPCSI), Chennai, 2017, pp. 267-272. doi: 10.1109/ICPCSI.2017.8392225

[6] M. M. Ma, S. Li and X. J. Liu, "Tracking control and stabilization of wheeled mobile robots by nonlinear model predictive control," Proceedings of the 31st Chinese Control Conference, Hefei, 2012, pp. 4056-4061.

[7] G. Mester, "Wireless sensor-based control of mobile robots motion," 2009 7th International Symposium on Intelligent Systems and Informatics, Subotica, 2009, pp. 81-84. doi: 10.1109/SISY.2009.5291190

[8] C. P. Connette, C. Parlitz, M. Hagele and A. Verl, "Singularity avoidance for over-actuated, pseudo-omnidirectional, wheeled mobile robots," 2009 IEEE International Conference on Robotics and Automation, Kobe, 2009, pp. 4124-4130. doi: 10.1109/ROBOT.2009.5152450

[9] J. Yin, G. Yang, F. Zhao and H. Qiu, "Motion planning implemented in ROS for Omni-directional Wheeled Mobile Robot," 2015 IEEE International Conference on Information and Automation, Lijiang, 2015, pp. 2695-2700. doi: 10.1109/ICInfA.2015.7279741

[10] T. S. Cleatus, S. Hedge, P. Harshitha and S. Jaswanth, "Unmanned two wheeled bot using feedback linearization," 2017 International Conference on Intelligent Computing and Control Systems (ICICCS), Madurai, 2017, pp. 1059-1062. doi: 10.1109/ICCONS.2017.8250629

[11] Rajesh Kannan Megalingam ; Vineetha Nandakumar ; A Athira ; G S Gopika ; Anjali Krishna "Orthotic arm control using EOG signals and GUI" 2016 International Conference on Robotics and Automation for Humanitarian Applications (RAHA) Year: 2016 **DOI:** 10.1109/RAHA.2016.7931871

[12] RajeshKannanMegalingam;Jeeba M.Varghese ; "Study and analysis of embedded system based indoor navigation on m ultiple platforms" 2016 International Conference on Communication and Signal Processing (ICCSP) Year: 2016 DOI: 10.1109/ICCSP.2016.7754343