

MOBILE ROBOTICS SOFTWARE REVIEWS

Dr. K.R. Prakash* Mr. Lakshmikantha K.S Mr. Rakesh M
Dept. of Mechanical Engineering Dept. of Mechanical Engineering Dept. of Mechanical Engineering
Ramya M.P Vinutha R. Nayaka
Student M.Tech Student M.Tech

Abstract--Mobile robot are being used to solve many complex problems like bomb disposal, automated home cleaning, Elder care, material handing etc. Novice users entering in the areas of mobile robotics spend substantial time in building hardware's like wheels, drives, chassis, sensors, and their interface. iRobot® create is one product which has many communication interfaces and ready to use built in sensors, wheels with encoders and rugged chassis, and can be controlled through codes from software's like real term, create.exe through communication interfaces. Standard script based coding makes the novice users directly get involved in solving real time problems. There are many software's through which iRobot create can be controlled, this paper explores few software's like Robot Operating System(ROS), Create.exe and realterm and lists their limitations and illustrate the steps involved in the control of the mobile robot. Feasibility of software's for navigation and / or artificial intelligence based application are reviewed and found that ROS is most suitable for deployment for such applications.

Keywords— Mobile Robots, navigation, iRobot® create, Artificial intelligence

I. INTRODUCTION

Robotics is the multi-disciplinary branch consists of mechanical engineering, electrical engineering, electronics engineering and computer science engineering that deals with the design, construction, operations, and application of robot, as well as computer systems for their control, sensory feedback, and information. Robotics are generally applied to solve dirt, dull, dangerous jobs where humans are prone to health and fatigue issues. Many of today's robots are inspired by nature contributing to the field of bio-inspired robotics.

The concept of creating machines that can operate autonomously dates back to classical times, but research into the functionality and potential uses of robots did not grow substantially until the 20th century. Throughout history, robotics has been often seen to mimic human behaviour, and often manage tasks in a similar fashion. Today, robotics is a rapidly growing field, as technological advances continue; researching, designing, and building new robots serve various practical purposes, whether domestically, commercially, or militarily. Many robots do jobs that are hazardous to people such as defusing bombs, mines and exploring shipwrecks.

A robot can be reprogrammable, multifunctional manipulator designed to move material, parts, tools or specialized devices through variable programmed motions for the performance of a variety of tasks.

A. Mobile Robotics

A mobile robot is an automatic machine that is capable of locomotion. Mobile robots can be "autonomous" (AMR - autonomous mobile robot) which means they are like teach pendent, or remote capable of navigating an uncontrolled environment without the need for physical or electro-mechanical guidance devices. Alternatively, mobile robots can rely on guidance devices that allow them to travel a pre-defined navigation route in relatively controlled space (AGV -Autonomous Guided Vehicle).

The development of accurate, robust and easy-to-use software tools for the simulation of robotic models, sensors, and control in virtual environments are illustrated in this paper. Fig.1 shows the mobile robot.



Fig. 1: Mobile robot

- Hospitals have been using autonomous mobile robots to move materials for many years.
- Mobile robots are also a major focus of current research and almost every major university has one or more labs that focus on mobile robot research.
- Mobile robots are also found in industrial, military and security settings.
- current research under study is utilizing iRobot® create.

B. *iRobot® Create*

iRobot® Create is a hobbyist robot manufactured by iRobot® that was introduced in 2007 and based on their Roomba vacuum cleaning platform. Fig. 2 shows external appearance of iRobot® create with its parts. The iRobot® Create is explicitly designed for robotics development and improves the experience beyond simply hacking the Roomba [2]. Create is an affordable mobile robot platform for educators, students and developers used to experiment and learn. The create consisting of serial communication, digital input & output, analog input & output, and an electric power supply, sensor data can be read and motor commands can be issued using the iRobot® Roomba Open Interface (ROI) protocol. Program behaviours, sounds and movements attach sensors, electronics, cameras, grippers and more. Advanced users can write custom software using a variety of methods that take advantage of the robot's "streaming sensor data" mode for more control of the robot. And highly advanced users can write programs for completely autonomous behaviour[1].

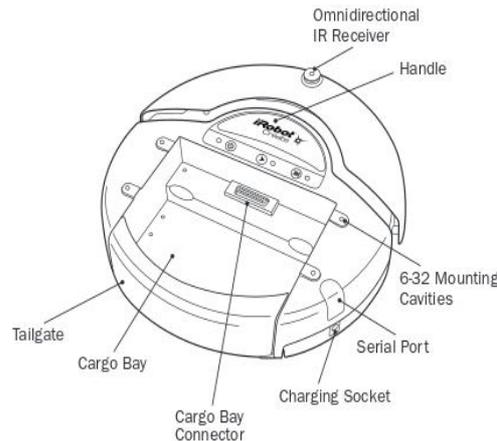


Fig. 2: iRobot® create

Some examples of iRobot® create projects:

- The iRobot® Create has been included in parts kits for the International Botball Competition since 2007.
- The iRobot® Create and a simulator developed in MATLAB are used in the Autonomous Mobile Robots course at Cornell University.
- Combining the iRobot® Create with an Xbox Kinect, student Philipp Robbel created a 3D mapping robot.

C. *Artificial Intelligence*

Artificial intelligence is the branch of computer science concerned with making computers behave like humans. The term was coined in 1956 by John McCarthy at the Massachusetts Institute of Technology [2]. Artificial intelligence includes the following areas of specialization:

- Games playing: programming computers to play games against human opponents.
- Expert systems: programming computers to make decisions in real-life situations (for example, some expert systems help doctors diagnose diseases based on symptoms).
- Natural language: programming computers to understand natural human languages.
- Neural networks: Systems that simulate intelligence by attempting to reproduce the types of physical connections that occur in animal brains.
- Robotics: programming computers to see and hear and react to other sensory stimuli.

Artificial intelligence and robotics are two closely related areas of study that fall under the field of computer science. Artificial intelligence and robotics specialists design computers and machines that act similar to humans with as little human involvement as possible.

Like the term "robot" itself, artificial intelligence is hard to define. Ultimate AI would be a recreation of the human thought process a man-made machine with our intellectual abilities. This would include the ability to learn just about anything, the ability to reason, the ability to use language and the ability to formulate original ideas. Roboticists are nowhere near achieving this level of artificial intelligence, but they have made a lot of progress with more limited AI. Today's AI machines can replicate some specific elements of intellectual ability of human.

Because of this, AI research is largely theoretical. Scientists hypothesize on how and why we learn and think, and they experiment with their ideas using robots. Brooks and his team focus on humanoid robots because they feel that being able to experience the world like a human is essential to developing human-like intelligence. It also makes it easier for people to interact with the robots, which potentially makes it easier for the robot to learn [2].

II. SOFTWARE TOOLS AVAILABLE

A. EMSS Framework

The EMSS (Environment Mapping Self Sustainable) iRobot® Create framework is a collection of interacting modules written in C++ designed to support iRobot® Create developers with a comprehensive set of controllers, Tasks, Schedulers, Movement Trackers, etc.. The framework provides not only an interface with the iRobot® Create hardware, but also a completely emulated interface which mimics the hardware in every sense. Along with core components for controlling the robot, the framework also features a set of GUI widgets for drawing maps and navigational paths as well as joysticks, control panels and more. The EMSS framework compiles on Linux, OS X, and Windows.

Fig. 3 illustrate how EMSS framework will works when it connected to iRobot® create ,it display an position data, sensor data and video recording of where the create can move. The EMSS iRobot® Create Framework is a great platform for anyone wishing to program the iRobot® Create hardware. Written in C++ and utilizing the Qt framework, EMSS compiles on Linux, OS X, and Windows. It is designed for 'high-power' robots which feature some sort of 32-bit computers as a controller, such as a top-mounted laptop. One of the most attractive features of the framework is the complete emulation of the iRobot® Create hardware a feature very valuable when writing and testing different algorithms. The framework is split into different modules which allow different implementations to be loaded. This design proves to flexible for different project needs and remains as a useful source base [3].

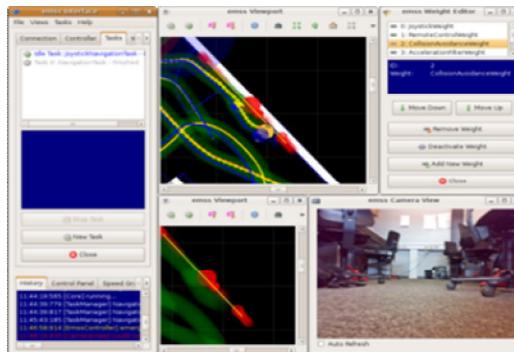


Fig. 3: The EMSS framework display screen when Create is running.

B. Pharos Testbed

Here is a quick description of the Pharos testbed. It also outlines the basic skills that potential users are recommended to have. The Pharos testbed is a "pervasive computing testbed that supports the study of complex cyber-physical systems." Put differently, it promotes research into the interactions of different types of hardware and software components and their interaction with the physical environment. A node within the Pharos testbed consists of a single Proteus robot, which contains motion, computation, and sensing capabilities.

The software stack of the Pharos Testbed consists of three layers. On the bottom is the Player/Stage middleware, which provides a software abstraction for movement and sensing capabilities. Above this resides The Pharos Middleware, which provides higher-level capabilities like GPS-based navigation services, and wireless communication. At the highest-level consists of the actual cyber-physical applications that are being evaluated using the testbed. It is recommended that potential users have a background in Linux, with knowledge of C, C++, and Java. Low level Player code tends to be written in C, while higher level algorithmic code is written in C++ and Java. Most of the coding for Player is straightforward, especially with many of the provided tools and code which handle many complex tasks, such as scheduling and controlling movement.

C. Create.exe

Create.exe is a stand-alone application used to control an iRobot® Create via a Serial or TCP/IP connection. This application will connect directly to the Create Serial Port via a COM Port or TCP/IP. It will poll and display sensor data, and allow you send individual commands or send entire scripts to control the Create. If you don't have a microcontroller on the Create and simply have a Bluetooth or other wireless serial connection, then this application will allow you to play with your robot right away. It also has a map that will use the distance and angle sensor data to display where you are and have been on a 2D grid[4].

Fig.4 illustrate the create.exe software contents .This application was written using Delphi 4[5].

To execute this application the following steps need to be carryout.

- download the create.exe software from this link <https://sites.google.com/site/irobotcreate2/home/files/Createv1.0.0.10.zip>.
- After download completes ,double click on the create.exe software for installation, it will take a few secs to install.

- When installation completes, the create.exe icon will appear on the desktop, double click on it to open the create.exe home page.
- To run this software ,the basic steps is as follows
 - Initialaize the COM PORT setting.
 - Select modes(passive,safe, full) on which robot works [preferabaly safe mode].
 - Click on the sensor data, to read the different sensor data values of an Robot.
 - This software can be run in both manual and autonomous mode.
 - For manual mode: the control buttons like forward, reverse, left, right, stop to control the movements.
 - For autonomous mode: to control the movements the script commands can be used.
 - The following data can be stored for future work, by using the Configuration file.
 - The configuration file consists of COM Port/TCP/IP settings, as well as the video URL.
 - You can then modify the settings directly in the config file or from the Configuration tab

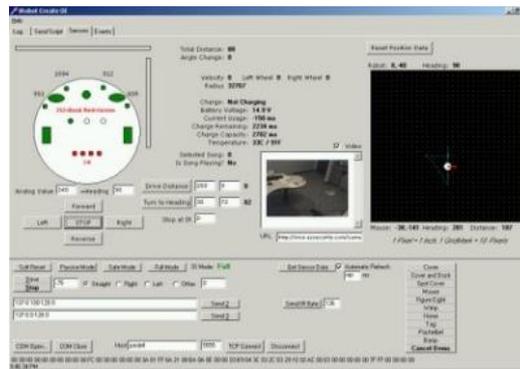


Fig. 4: The create.exe software shows sensor data ,position data, video.

Some of the Create.exe Features are:

- Remotely Control your iRobot® Create with any Windows PC.
- Displays all sensor data with visuals.
- Display a web page with embedded Video from your Robot's IP Cam.
- X,Y Coordinate Mapping.
- Analog Input on the Create can be used as a digital Compass input.
- Log tag shows the scripts for all commands sent to the Create so you can learn to write your own.
- Write scripts, save them, send them to the Create to execute them.

D. Realterm: serial terminal

Yet another open source software available to public. Fig. 5 shows realterm: serial terminal software when the create is connected to pc. Realterm is a terminal program specially designed for capturing, controlling and debugging binary and other difficult data streams. It is far better for debugging than HyperTerminal. It has no support for dialling modems, BBS etc that is the role of HyperTerminal does[6].

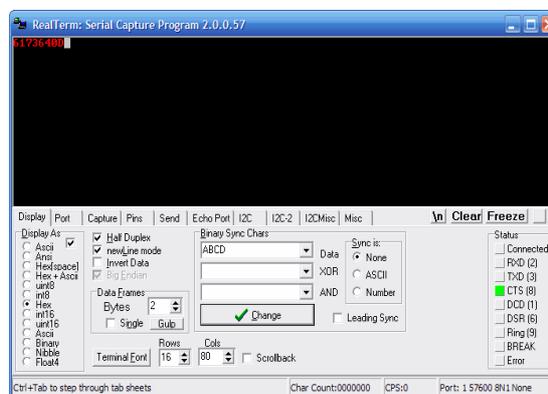


Fig. 5: The realterm: serial terminal shows that in display tab input type hex decimal has been selected.

E. Robot Operating System (ROS)

ROS is a framework that is widely used in robotics. The philosophy is to make a piece of software that could work in other robots by making little changes in the code. What we get with this idea is to create functionalities that can be shared and used in other robots without much effort so that we do not re-invent the wheel. ROS was originally developed in 2007 by the Stanford Artificial Intelligence Laboratory (SAIL) with the support of the Stanford AI Robot project. In the Fig. 6, you can see some fully supported platforms. Normally, these platforms are published with a lot of code, examples, and simulators to permit the developers to start their work easily.

The sensors and actuators used in robotics have also been adapted to be used with ROS. Every day an increasing number of devices are supported by this framework. ROS provides standard operating system facilities such as hardware abstraction, low-level device control, implementation of commonly used functionalities, message passing between processes, and package management. It is based on graph architecture with a centralized topology where processing takes place in nodes that may receive or post, such as multiplex sensor, control, state, planning, actuator, and so on. [7].



Fig.6: ROS supported robots

- Packages containing application-related code which uses one or more ROS client libraries, The Main ROS client libraries are C++, python, lisp other packages implement commonly used functionality and applications such as hardware drivers, robot models, planning, simultaneous localization and mapping, simulation tools, and other algorithms.
- ROS support the visualization software such as gazebo, RVIZ (ROS visualisation tool) etc.

1) ROS Package application areas will include:

- Perception
- Object Identification
- Segmentation and recognition
- Face recognition
- Gesture recognition
- Motion tracking
- Motion understanding

Among many softwares realterm and create.exe found suitable for early implementation and are thus compared for the application.

The table 1 indicates the comparison of Realterm software and Create.exe.

Features	Realterm	Create.exe
Sensor Data Reading	No	Yes
Programming Language	Script Commands	Script Commands
Video Recording	No	Yes
Colour Tracking To Detect The Objects	No	Yes
Manual Control	No	Yes
Position Data	No	Yes
Indication of Failure	No	Yes
Indicates Total Distance Travelled	No	Yes
Indicates Charging Capacity	No	Yes

Table 1: Difference between realterm and create.exe software.

III. METHODOLOGY

A. Create.exe

When the create is connected to pc, USB port is detected, now open the create.exe software .before going to send the script commands to create we need to do COM port setting.

1) Create in safe mode.

Fig. 7 show how the software looks when the create in safe mode. In safe mode, the create can run safely while sending commands. Otherwise create gets damages .

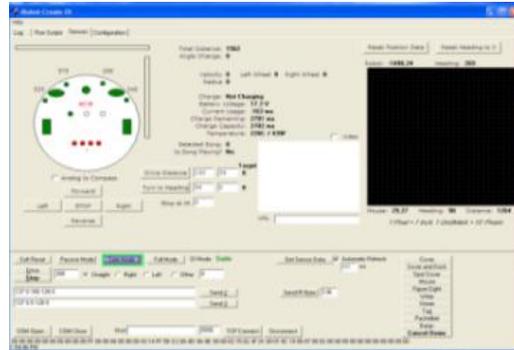


Fig.7: Create in safe mode

2) Reading of sensor data

While iRobot® create moving on the floor, if create sensors detects any obstacles through IR receivers relates sensors gets activated and shows sensor data's .Fig. 8 illustrates the sensor data in red colour. And the position data of create can be recorded as seen in right side.

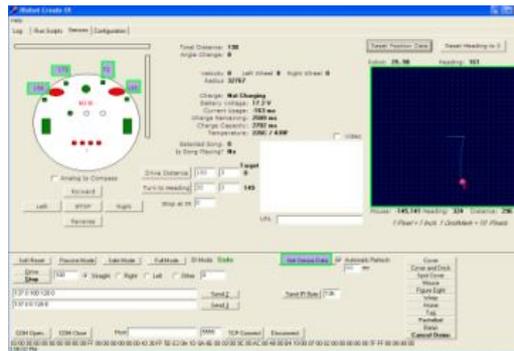


Fig. 8: Sensors data reading and position data recording.

3) Stuck Somewhere

Fig .9 shows, If the create is stuck somewhere on the floor the red blinking on circle (that circle shows a create).

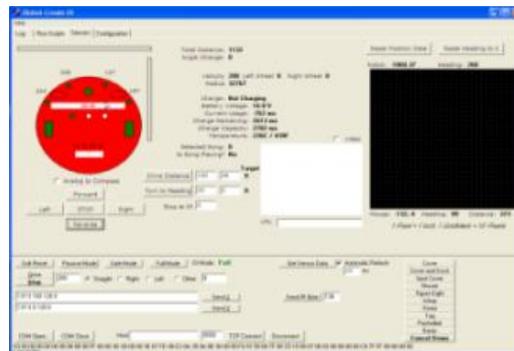


Fig. 9: when the create stuck

4) Video Recording

Fig. 10 shows ,using this software we can track the object through camera.

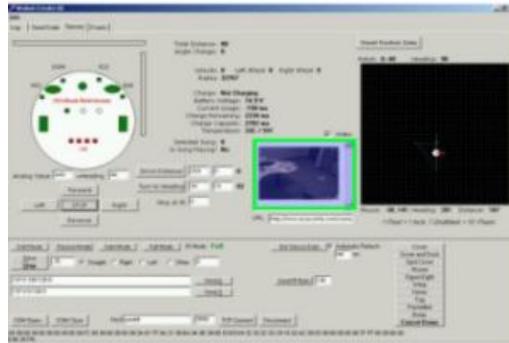


Fig. 10: Create is connected with camera to the create.exe software.

B. Realterm :serial terminal

The following guide provides information on how to create a connection to the Micro tops II sun photometer using realterm, and also how to retrieve collected

- Data Connect serial cable to COM port of PC
- Switch on sun photometer
- Run Realterm

The real terminal software screen contains the many tabs which are explained below.

1) *Display Tab:*

Fig. 11 shows Realterm displays data in meaningful forms

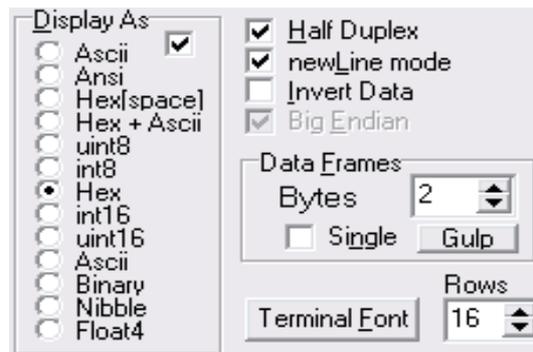


Fig. 11: Display tab setting to run the commands

2) *Port tab*

Fig. 12 shows the port setting of an create Realterm can connect to both SERIAL ports (real uarts, as well as USB, and network virtual uarts) or TCP/Telnet ports.

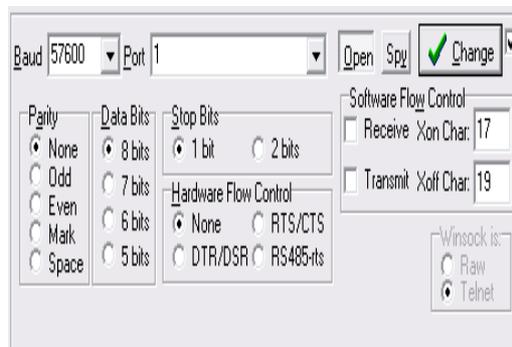


Fig. 12: Port setting

3) Send tab

Fig.13 shows the send tab through this we can send an scrip commands to create.

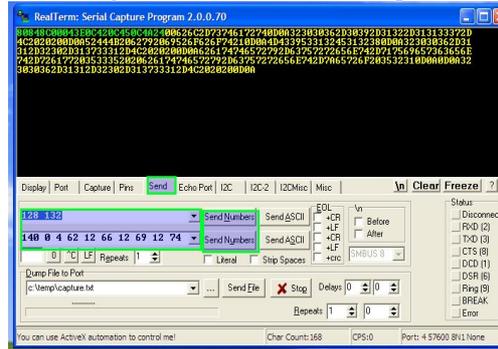


Fig. 13: Sending of command to create to drive

IV. CONCLUSIONS

Many mobile robotics application need tracking of path and visual aid through camera and this comparative study revealed create.exe is one of the easiest method to accomplish such application for novice users to kick start using the mobile robot like iRobot® create. ROS is capable of developing AI, navigation, 3D reconstructions and many more applications. Large number of dedicated ROS programmers support, online documentation and open software architecture make the software more suitable for novice and experienced Robot enthusiasts.

ACKNOWLEDGMENT

Special thanks for the financial support from KCTU/R&D Centered/2014-2015 Government order no CI 151 SPI 2013(P₁) to conduct research in the area of robotics. We thank department of mechanical engineering head of the department Dr. N. V. Ragavendra for his continuous motivation, support and infrastructure facilities for this research. We thank TEQTP-II for additional assistance in publishing this paper.

REFERENCES

- [1] Ankur Aggarwal, Rachana Jain "iRobot® create: programmed as provider" International Journal on Computer Science and engineering(IJCSE) ISSN: 0975-3397 Vol. 4 No. 04 April 2012 635.
- [2] study.com, Robotics and Artificial intelligence
http://study.com/directory/category/Computer_Sciences/Computer_and_Information_Sciences,_General/Robotics_and_Artificial_Intelligence.html.
- [3] Environment mapping Self-Sustainable Robot, David Grob and daniek Krusi,Advised by prof. Dr. Jochim Wirth, December,2008.
- [4] "Aaron Staranowicz,Gian Luca Mariottini,"A survey and Comparison of Commercial and Open-source Robotic Simulator Software".
- [5] [https://sites.google.com/site/iRobot®createe2/create.exesourcecode](https://sites.google.com/site/iRobot%20createe2/create.exesourcecode)
- [6] "Downloading data from the Microtops II Sunphotometer: Guide to RealTerm serial communication program". "learning ROS for robotics programming", Aaron martinez enrique fernande,ISBN 978-1-78216-144-8,First published : September 2013".