**Name:- Sanket Chandrashekhar Harvande (19)**

**Assignment No:- 04**

**Diving into Educational Robotics with Player/Stage**

Carlos Jaramillo

*Computer Engineering Department, city college of New York, CUUNY*

*New York, NY, 10031 USA*

[Cjarami00@cny.cunny.edu](mailto:Cjarami00@cny.cunny.edu)

**Abstract:**

This paper describes my first research experience in programming robots by using open-source software tools such as Player/Stage. Among the educational robots available at Brooklyn college`s Agents Lab, the Surveyor and the Scribbler were preferred. Eventually, the project focused itself on implementing a plug-in-driver for the Surveyor robot so that it could be controlled via Player/Stage. A Player/Stage-based GUI console-to remotely control the surveyor and the other robots with similar actuators and sensors – was also developed although for now it lacks some features that could allow simultaneous Localization and Mapping(SLAM) for Urban Search and Rescue(USAR) missions.

**Keywords-** Player/Stage, surveyor, SRV-1, Scribbler, driver, SRVjoy, educational robotics, open-source, USAR, Brooklyn College, Sklar.

I. Introduction

Diving into Educational Robotics with Player/Stage is the final report by the under graduate student under the guidance of mentor Elizabeth Sklar, Ph.D., from the Dept. of Computer Science in the New York University. The student adapt drivers from small educational robots available at the lab into Player/Stage. After his work the agent’s lab at Brooklyn College performed same research on the wide range of robots. To explore this robotics they used robot server & simulation, IPRE Scribbler, SRV-1. Among all these they used SRV-1 & Scribbler for their research. For their research they used C/C++ & existing Simulators.

II. Player/ Stage

For this projects they create free software that research in robot and sensors systems. In this research the player has used two simulations that is 2-D backend & 3-D environment. For backend they have widely used C/C++ also python programming. And for environment the player has used Gazebo. In is research they also have use AI fundamentals to develop reusable controllers for their and other robots.

III. Surveyor Plugin Driver for Players

They defined the key concepts in the process of writing drivers for other robots also. For example Roomba robot.

A. About the Surveyor SRV-1

It resembles a small military tank with treads. This kind of generation is available in SRV-1 lab, because they have ARM-7 processor on-board. It can communicate serially through XBee radio dongle. Low-resolution camera and four infra-red sensors are the capability SRV-1.

B. Inheriting from Player`s Driver Class

This driver acts a proxy between robot`s hardware interface and the function calls to the player server. A plugin driver is compiled as a shared-object that is dynamically linked to player`s driver.

$ *player surveyor.cfg*

This example configure file with various option which is shared-library for the plugin driver is named libsurveyor\_Driver.so

driver

(

name “surveyor”

plugin “libSurveyor\_Driver.so”

provides[“position2d:0” “camera:0”]

port“/dev/ttyUSB0”

image\_size“320 x 240”

)

The code has instantiated and its central thread has been started by Setup() function, and the remaining process is handled by Main() method.

C. Results

The Surveyor SRV-1 was initially tested by members of CLI (Command Line Interfece) and the utilities of this project is playerjoy and playercam. This job can done as long as the interfaces provide properly. Due to odometry robot is capable to move around in various direction, but now does not support. Due to this DC motors are use as servo-motors that can provide exact positioning data.

IV. SRV Joy: A Player GUI Console to the SRV-1

The use of GUI is important in this research because it allow a human operator to drive the robot and take snapshot images from the camera to present a live-video. Along with GUI there is another option which is SRVjoy which provides similar interface.

V. Conclusion

The research experience took into different way to interact with small robots. The main point in this research is Player/Stage was preferred to use API in this research. Due to my field I am very interested to connect hardware and software. In this project there has many things to do but I am concentrate on the functionality of SRV-1 driver and SRVjoy console. In all this research open source helps me lot.

Acknowledgement

We first thanks to John Cummins for his devoted guidance and work he has done. Also we thanks to our professors Sklar and Parsons for the C/C++ Libraries for the SRV-1 and Scribbler robots. Also, we thanks to Michael Jansen for the original implementation of the Surveyor driver for Player/Stage that can be informally found in the Surveyor forums.

Reference

[1] (2009) The Player Project website. [Online]. Available: <http://playerstage.sourceforge.net>

[2] Exploring Robotics (CC 30.03) course website [Online]. Available: <http://agents.sci.brooklyn.cuny.edu/cc30.03/>

[3] Bridges High School Workshop website. [Online]. Available: <http://bridges.brooklyn.cuny.edu/index.php/home>

[4] Lego Education website [Online]. Available: <http://www.legoeducation.com>

[5] Surveyor Corporation website. [Online]. Available: <http://www.surveyor.com/>

[6] Institute for Personal Robots in Education (IPRE) website. [Online]. Available: <http://roboteducation.org>

[7] Scribbler`s Myro [online]. Available: <http://wiki.roboteducation.org/Myro_Reference_Manual>

[8] John Cummins, M. Q. Azhar, and Elizabeth Sklar, “Using Surveyor SRV-1 Robots to Motivate CS1 Students,” AAAI.org, 2008

[9] Surveyor Robot Software from the Agents Lab at Brooklyn College. [Online]. Available: <http://agents.sci.brooklyn.cuny.edu/robotics.edu/bcsoftware.php>

[10] Brian P. Gerkey and Richard T. Vaughan, “Really Reusable Robot Code and the Player/Stage Project.” Software Engineering for Experimental Robotics Springer Tracts on Advanced Robotics, Springer, 2006.

[11] Introduction to Artificial Intelligence (CIS 32) course website. [Online]. Available: <http://www.sci.brooklyn.cuny.edu/~parsons/courses/32-spring-2009/>

[12] Player/Stage Drivers: Writing a Player Plugin from the Penn State Robotics Encyclopedia RoboWiki. [Online]. Available: <http://psurobotics.org/wiki/index.php?title=Player/Stage_Drivers>

[13] Jonas Fonseca and Bue Petersen, “Writing Player/Stage Drivers: a howto for ERSP Player Driver Source Package,” From the project: Player/Stage - Player driver implementation for ERSP, 2006