UTBot@Home 2019 - Using the Pioneer LX and Pionner AT robots to emulate emotions and manipulate objects

Felipe Pierre Conter[1], Piatan Sfair Palar[2], Marlon de Oliveira Vaz[3],
João Alberto Fabro[1] and André Schneider de Oliveira[2]

***Abstract*— This TDP describes the hardware and software implemented to allow two Pioneer robots available at UTFPR to execute domestic tasks related to the Robocup@Home initiative. The robots are a Pioneer LX research robot (also known as "Z") and a Pionner AT (a.k.a. "Apollo"). These robots have sensors, such as a laser scanner and a Kinect 3D sensor, that provide information necessary for navigation, object recognition, environment mapping, auto-localization. An innovative human-robot interaction interface, using different facial expressions based on simulated emotions, is also presented. The robots are programmed to execute some Robocup@Home tasks, such as voice recognition, object recognition and manipulation (using their embedded manipulator arms, and computer vision algorithms). All the programming uses ROS (Robot Operating System), so that the same control programs can be executed in both robots with minimum alterations.**

I. INTRODUCTION

Robocup@Home is a competition that aims to foster the development of robots that will help people in domestic environment, being the biggest competition for service robots on the planet [1][2]. In order to perform household tasks, the robotics fields of human-robot interaction, navigation, map construction on dynamic environments and computer vision are developed, among many others [1].

To accomplish these tasks, two Pioneer research robots are being used: a Pioneer LX (a.k.a. "Z") and a Pioneer AT (a.k.a. "Apollo"), presented in Figures 1 and 2. Both robots are being programmed with ROS (Robot Operating System), and a set of software libraries and tools that help to develop applications for robotics [3]. The control system of the robots is responsible for acquiring and interpreting various sensors used in both robots and, from these data, make decisions to fulfill its tasks.

In this document, the partial results and expectations of projects under development for the robots are presented. Section 2 presents a description of the robots hardware, including their specifications and characteristics. The software already developed, and what is still in development, are described in section 3. Conclusions and future perspectives are presented in section 4.

II. HARDWARE OF THE ROBOTS

A. Robots Specifications

One of the robots used as base for the project is the Pioneer LX ("Z"), which is a research platform based on the industrial robot Adept Lynx. This 60 kg robot can carry up to 60 kg of payload, and travel at a maximum speed of 1.8 m/s, and has a autonomy of 13 hours of continuous operation. The dimensions of the Pioneer LX robot are: 50 cm wide, 70 cm deep and 45 cm high. A computer with Dual Core 1.8 GHz processor, 2GB DDR3 RAM and Wireless Ethernet is included inside this robot. The final assembly includes the sensors detailed on the next topic, as well as a Genesis TAD model GT-7305 tablet, with a resolution of 1280x720, running Android, which is used to display faces simulating the robot's current emotions. Figure 1 presents this robot.

Fig. 1. Pioneer LX (a.k.a. "Z") with its articulated arm, Kinect sensor and the tablet that is used to present simulated emotions.

The other robot, based on a Pioneer P3-AT research robot, is a simpler and smaller research robot. It weights only 12

[1]J. A. Fabro is a professor with the Graduate Program on Applied Computing - PPGCA, at the Informatics Department - DAINF - at UTFPR - Federal University of Technology, Parana, Brazil - Campus Curitiba. F. P. Conter is a master student within this Graduate Program. fabro@utfpr.edu.br, felipeconter.cc@gmail.com

[2]M. O. Vaz is a professor with IFPR - Federal Institute of Parana, Campus Pinhais, Pinhais marlon.vaz@iffpr.edu.br

[3]A. S. de Oliveira is a professor with CPGEI - Graduate Program of Electrical Engineering and Industrial Informatics, UTFPR, Brazil - Campus Curitiba. P. S. Palar is a master student within this Graduate Program. andreoliveira@utfpr.edu.br, kpiatan@gmail.com

Kg, with 12 Kg of payload, and maximum speed of 0.7 m/s. Its autonomy is of 2 hours, with 3, 12 volts, standard batteries. Its dimensions are: 50 cm wide, 50 cm deep and 28 cm high. Inside the robot, a computer similar to the one found inside the Pionner LX also controls this robot. Figure 2 presents this robot.

B. *"Neck" Support*

On each robot's surface is a support constructed of aluminium (as seen in figures 1 and 2). These supports have 1,2 meters, allowing for the positioning of the tablet and the Kinect sensor close to a height comfortable to interact with people (1,5 meters approximately, in both robots).

Fig. 2. Pioneer AT (a.k.a. "Apollo") with its support for the Kinect sensor, and the tablet that is used to present simulated emotions.

C. *Sensors*

Some sensors are also embedded in the robots: "Z" has a 270-degree rangefinder laser sensor, with a high precision and big range (up to 25 meters) [1]. In addition to this sensor, that is used for localization and mapping, a Kinect depth sensor has also been incorporated. With this sensor it is possible to create a deep image, identify silhouette of persons, and objects, among other functions [4]. In order to establish the communication between the Kinect sensor and ROS the Openni Kinect open source project was used [5]. The data generated by Kinect and the laser sensor that are published in topics in ROS are key elements to operating most of the algorithms used in this project, since it is based on the interpretation of these images that the robot takes a large part of its decisions. "Appolo" does not have the same high quality laser sensor, but uses a low cost laser sensor [6] in its place.

III. SOFTWARE

A. *ROS - Robotic Operating System*

Each robot has a control notebook with the "kinect" version of ROS, on the operating system Ubuntu 16.04 LTS. ROS is a system that has several tools, libraries and conventions that simplify the development of complex and robust tasks for robots [3]. In this system, you need to run a core process, *roscore*, which function as a server that manages threads and nodes. Topics are data buses through which nodes exchange messages, and multiple nodes may subscribe to each topic (receiving its data) or publishing to it (sending data). Nodes are computational processes, that process data coming from some topics, and publish results to other topics.

Each robot has its kinematics calculated, and the integrated encoders provide velocity feedback by communicating through a serial-usb cable with the notebook, where it publishes and receives data through ROS. Each robot's notebook is configured to be accessed via secure shell (SSH) protocol from other computers so it can be monitored remotely over a wireless network. In addition, the ROS configuration allows the work with multiple computers [7], so parts of the processing can be done on another computers, making it possible to split the processing just by attaching other notebooks to the robot, if necessary.

B. *Emotion Simulation*

Emotions are simulated by the robot through faces, displayed according to the present situation, for a better visualization of internal states of the robot, and better interaction between the robot and people.

Fig. 3. Simulated emotion presented on the tablet (face) of the robots.

Emotions were based on Plutchik's wheel of emotions [8]. A tablet running the ROS Image Viewer application displays these faces by subscribing to a ROS topic. Figure 3 illustrates an emotion being displayed by the robot through the tablet.

Figure 4 depicts some of the faces developed to express different emotions during the interaction with people.

Fig. 4. Examples of faces developed.

C. Simultaneous Localization and Mapping

Using the encoders data and correcting the estimated current position of the robot using the laser sensor readings, it is possible to calculate an approximate relative position of the robot. The robot observes the environment in order to create a model (map). Measurements are made through these observations to identify where the robot is and landmarks are defined to help the robot to locate itself within the map. Any map error will propagate errors in the location, but fortunately, location estimation errors correlate. These correlation errors are corrected through the Extended Kalman Filter (EKF) system [9], which manages to store the correlation between all estimation errors and makes the location dependent only on the initial location uncertainty of the robot.

The use of SLAM (Simultaneous Mapping and Mapping) [10] has been implemented in the robot, a technique used to construct a map of an environment at the same time as it is updating the robot's position. To accomplish this, SLAM has a number of tasks: extraction of reference points, data association, state estimation, status update, and reference point update. For this, among other functions, it uses the EKF. This was done using ROS ready-made functions, of which the parameters were adjusted so that navigation is performed satisfactorily.

D. Voice recognition and synthesis

For speech synthesis, a ROS package called *espeak_ros* is being used. This package works by running a ROS node that subscribes to the topic *speak_line*, and performs voice synthesis based on the specified parameters. The configuration of these speech parameters is done through the *dynamic_reconfigure* package, which allows you to change parameters on ROS nodes without having to restart them. The configuration of the speech parameters encompasses several characteristics, such as speech speed, volume, pitch (lower or higher), the space between words, age and intonation variation. By altering these parameters, the speech characteristics can be defined in accordance with the emotion displayed on its "face" (which is a tablet). Voice recognition will be performed with the PocketSphinx package [11].

E. Object Recognition and Manipulation

Object recognition can be accomplished through different approaches. Simple techniques such as blob detection can help identify objects that have a well defined color and shape, with a strong difference in intensity from the background. In a simulated environment, with well defined objects such as plain colored cubes or spheres, segmentation of these objects was achieved through blob detection. However, since the spectrum of household objects varies in shape, colors, transparency, and many other characteristics, machine learning may perform better than traditional image processing techniques.

The latest experiment in the simulation environment V-REP [12] includes a table with a white cube on top of it, and a youBot robot, by automation company KUKA [13]. It was selected because of it's open platform, the presence of a manipulation arm, and a good model already available in V-REP. The scene in V-REP includes a 5x5 floor. The robot was modified by adding a laserscan Hokuyo URG-04LX-UG01 sensor [14] and V-REP sensor model called "Blob detection camera". The Hokuyo sensor was placed on the front side of youBot base, and the camera positioned on the tip of the arm, just before the grippers, as can be seen on figure 5.

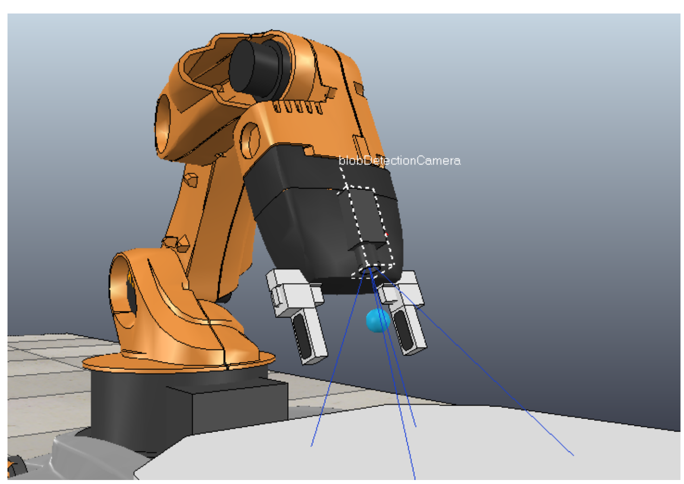

Fig. 5. Blob detection camera position on youBot arm.

The problem set in V-REP is the following: youBot has to position itself next to the table, with a predefined orientation (positioned towards the table). The table has a white cube on its top. Once arriving to it's destination, the robot has to position it's camera on top of the table and detect the cube, using the blob detection camera sequence of filters to segment the object, as displayed on figure 6. Robot movement and positioning was managed by ROS code running outside V-REP.

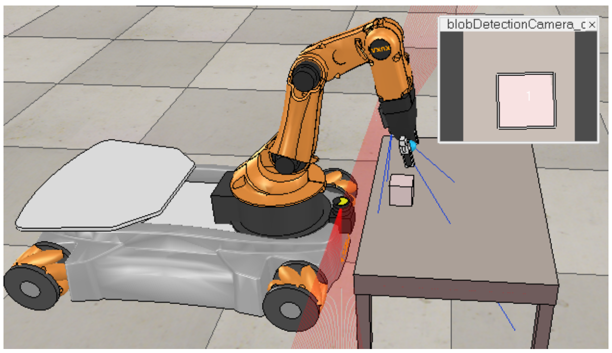

Fig. 6. Arm position and blob detection

Object detection in images is the first step towards object recognition (classification and definition of object's position in the image). The next step is to estimate the object's position relative to the robot, so that object gripping can be performed.

Both "Z" and "Apollo" robots are capable of tasks such as gripping and positioning of small objects, since both have manipulation arms. The integration of the object recognition, gripping and object manipulation in these robots is still in active development. It is expected that by the time of the competition, the performance of this task will have a significant advance.

IV. CONCLUSION AND FUTURE WORK

The team is capable of fulfilling several tasks of the Robocup@Home category successfully, and it is expected that by the competition date, even more results can be achieved. Among the tasks to be developed are the more accurate recognition of voice commands, and the integration of SLAM and manipulation tasks. Object recognition is expected to be the main advance of this year with our robots, using Kinect and plain RGB images. Also, initial research in applying machine learning techniques to object recognition may provide better performance in this task.

ACKNOWLEDGMENT

The authors would like to thank the financial support of UTFPR.

REFERENCES

[1] Official homepage: Robocup@Home. http://www.robocupathome.org/. Accessed June 2019.

[2] L. IOCCHII, D. HOLZ, J.R. del SOLAR, K. SUGIURA, T. van der ZANT, RoboCup@Home: Analysis and results of evolving competitions for domestic and service robots. Artificial Intelligence. 229, C (December 2015), 258-281. DOI: https://doi.org/10.1016/j.artint.2015.08.002

[3] About ROS Official Homepage. http://www.ros.org/about-ros/. Accessed June 2019.

[4] MICROSOFT, Develop Network: Kinect Sensor. http://msdn.microsoft.com/en-us/library/hh438998.aspx. Accessed June 2019.

[5] V. RABAUD, T. FOOTE, Openni_kinect ROS. http://wiki.ros.org/openni_kinect. Accessed June 2019.

[6] Lidar Scanner Ydlidar X4. https://acroname.com/sites/default/files/assets/ydlidar_g4_datasheet_0.pdf. Accessed June 2019.

[7] ROS Multiple Machines. http://wiki.ros.org/ROS/Tutorials/MultipleMachines. Accessed June 2019.

[8] R. PLUTCHIK, H. KELLERMAN (Ed.). Emotion: theory, research and experience. New York: Academic press, 1986.

[9] M. I. RIBEIRO, Kalman and Extended Kalman Filters: Concept, Derivation and Properties. http://users.isr.ist.utl.pt/~mir/pub/kalman.pdf. Accessed June 2019.

[10] S. RIISGAARD, M. R. BLAS, SLAM for Dummies. http://ocw.mit.edu/courses/aeronautics-and-astronautics/16-412j-cognitive-robotics-spring-2005/projects/1aslam_blas_repo.pdf. Accessed June 2019.

[11] PocketSphinks. http://wiki.ros.org/pocketsphinx/. Accessed June 2019.

[12] Coppelia Robotics V-REP: Create. Compose. Simulate. Any Robot. http://www.coppeliarobotics.com. Accessed June 2019.

[13] KUKA YouBot. Mobile manipulator for research and education. https://bit.ly/2r1uyql. Accessed June 2019.

[14] Scanning Rangefinder Distance Data Output/URG-04LX-UG01 Product Details — HOKUYO AUTOMATIC CO., LTD. https://www.hokuyo-aut.jp/search/single.php?serial=166. Accessed June 2019.