

DSPL Pumas@Home 2019 Team Description Paper

Jesus Savage, Edgar Vazquez, Edgar Silva, Jesus Hernandez, Adrian Sarmiento, and Angelica Nakayama

Bio-Robotics Laboratory, School of Engineering
National Autonomous University of Mexico
<http://biorobotics.fi-p.unam.mx>

Abstract. This paper describes the current research topics and main findings of Pumas Team as well as the efforts to implement all the developed software into the Toyota Human Support Robot (HSR). Pumas Team has participated in several national and international robotics competitions that have influenced our architecture for the development of better systems for our service robots. Current research topics include facial detection through multiple images using RGB cameras, creation of RGB-D representations of the environment and action planning using state space representations, among others. In our robotics architecture, the Virtual and Real roBOt sysTem (VIRBOT), the operation of service robots is divided into several subsystems, each of them has a specific functionality that contributes to the final operation of the robot. By combining symbolic AI with digital signal processing techniques a good performance of a service robot is obtained. We consider that our robotics architecture, VIRBOT, will be transferred successfully into the Toyota Human Support Robot (HSR).

1 Introduction

A service robot is a robot that can assist humans to perform common daily tasks in a common environment, such as houses, offices or hospitals. With this in mind, the final goal of service robot must be make the life of humans easier and more comfortable. Also, a robot can be an excellent companion, in example for elderly or lonely people, making their life better and happier.

To achieve this, a service robot must be capable of understanding spoken and visual commands in a natural way from humans, navigate in known and unknown environments avoiding static and dynamic obstacles, recognize and manipulating objects, detect and identify people, among several other tasks that a person might request.

The team Pumas has participated in national and international competitions. In this year, in the Robocup 2018, our team obtenied the second place in the category DSPL@Home with the robot "Takeshi" and we were finalist in World Robot Summit(WRS) 2018.

2. TAKESHI'S ROBOTICS ARCHITECTURE

The paper is organized as follows: section 2 enumerates the software components of our robot "Takeshi"; section 3 presents overview of the latest research developments in our laboratory; and finally, in section 5, the conclusions and future work are given.

2 Takeshi's Robotics Architecture

2.1 Hardware Configuration

Our service robot "Takeshi" is an HSR Toyota robot, has the following software configuration that it is based on the VIRBOT architecture [1], which provides a platform for the design and development of software for general purpose service robots, see figure 1. The VIRBOT architecture is implemented in our robots through several modules that perform well defined tasks [2], with a high level of interaction between them. The principal framework used for interaction is ROS, where a module is represented by one or several ROS's nodes. Also, for modules using the Microsoft operating system, we use our own middleware called Blackboard to link them with ROS nodes running on Linux. In the following sections are explained each of the layers of the VIRBOT system.



Fig. 1: Block diagram of the ViRBot architecture.

2.2 Inputs Layer

This layer process the data from the robot's internal and external sensors, they provide information of the internal state of the robot, as well as, the external

world where the robot interacts. Some of onboard sensors are a laser scanner, a RGB-D camera, an array of microphones, one stereo camera and two cameras with wide-angle lens. Digital signal processing techniques are applied to the data provided by the internal and external sensors to obtain a symbolic representation of the data, as well as, to recognize and to process voice and visual data. Pattern recognition techniques are used to create models of the objects and the persons that interact with the robot. With the symbolic representation this module generates a series of beliefs, that represent the state of the environment where the robot interacts.

2.3 Planning Layer

The beliefs generated by the perception module are validated by this layer, it uses the Knowledge Management layer to validate them, thus a situation recognition is created. Given a situation recognition, a set of goals are activated to solve it. Action planning finds a sequence of physical operations to achieve the activated goals.

2.4 Knowledge Management Layer

This layer has different types of maps for the representation of the environment, they are created using SLAM techniques. Also in this layer there is a localization system, that uses the Kalman filter, to estimate the robot's position and orientation. A rule based system, CLIPS, developed by NASA, is used to represent the robot's knowledge, in which each rule contains the encoded knowledge of an expert.

2.5 Execution Layer

This layer executes the actions and movements plans and it checks that they are executed accordingly. A set of hardwired procedures, represented by state machines, are used to partially solve specific problems, finding persons, object manipulation, etc. The action planner uses these bank of procedures and it joins some of them to generate a plan.

3 Current research

In this section is presented the current research developed in our laboratory to improve the performance of our service robots.

4 Implementing ViRBot into the Toyota HSR Simulator

The ViRBot architecture has been succesfully tested and implemented in Justina, a domestic service robot developed at the Biorobotics Laboratory of the National Autonomous University of Mexico. Implementation is made using the

4. IMPLEMENTING VIRBOT INTO THE TOYOTA HSR SIMULATOR

ROS platform running on Ubuntu 16.04. For speech recognition and synthesis, we use the Microsoft SAPI running on virtual machine inside the Ubuntu computer.

Currently we are working on the implementation of all the Justina's software on the Toyota HSR Simulator. Since we use the ROS code development guidelines, our programs are platform-independent enough and the only change should be the low-level sensor reading and actuator control. Figure 2 shows the robots Justina and HSR performing an object manipulation task, both running the same programs under the ViRBot architecture.

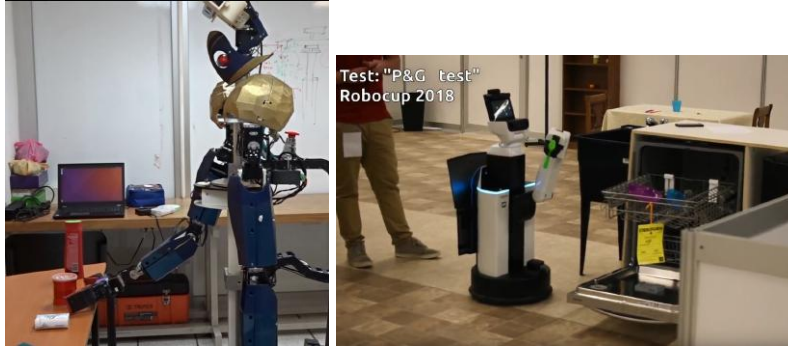


Fig.2: Justina and HSR object manipulation using the same programs both under the ViRBot architecture

Same as HSR, Justina has a simulator for testing algorithms before the real implementation. As described in section 4.1, simulated environment is accurate enough for testing high level plans. Figure 3 shows the Justina and HSR simulated environments.

4.1 High level task planning using conceptual dependency

This year, a new knowledge based system for high level task planning has been developed. This system is designed to interpretate a command in a natural language format and for generating a sequence of high level actions that have to be completed for accomplishing the command. With this system, the robot can generate behaviors to build his own world representation, similarly to a human being.

The system is composed by two principal modules:

Communication Module In this module, the grammar structure of the commands is defined. Performs a syntactical analysis and a semantic interpretation using inferences and Conceptual Dependency, in order to find a meaning for the command. Finally, this module returns a formal expression that describe a sequence of high level actions that the robot have to execute.

4. IMPLEMENTING VIRBOT INTO THE TOYOTA HSR SIMULATOR

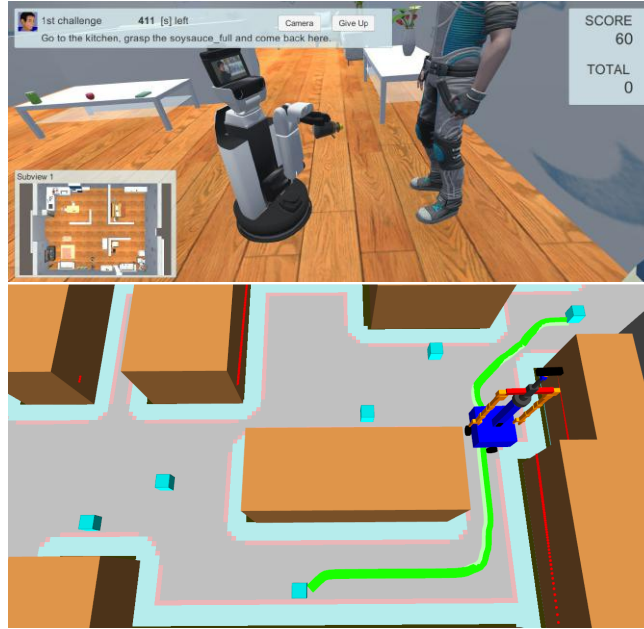


Fig. 3: Justina and HSR simulation environments for algorithm testing.

Execution Module Consist in a set of rules for infer and determinate the sequence of high level tasks as basic and atomic predefined actions that represents the plan that will allow the robot to accomplish the command that has been requested. This module was developed using CLIPS, an expert system designed to represent and model the human knowledge.

4.2 Facial detection through multiple images using RGB cameras

The facial detection is one of the most important tasks that a service robot must be able to perform. In the last RoboCup@Home, there was a test that involves stating the number and characteristics of several persons with different heights, gender and ages, aligned in different rows. The people can be standing, sitting or even lying down. This task can be difficult to solve with a single image capture, due the size of the crowd: if the robot is too close, it can not see all the people in the crowd. If the robot is too far, details of the picture can be lost.

We solved this test by moving the head of the robot in order to take several pictures with different angles, merging them into one single large picture, and then detecting faces in the large picture. An example of this is shown in figure 4.

For merging, we extract SURF [7] features for every picture. Then, we match the features using FLANN [8] in order to obtain the maximum and minimum distances of the paired points. We later found the holography matrix using



Fig. 4: Merged image and face recognition.

RANSAC [9] and perform a transformation to one of the pictures, and paste it together. All this process is done iteratively, were in every step we obtain a larger picture. After, this, a global color equalization is performed.

4.3 Low or null texture objects recognition using RGB-D cameras

Currently, several robust techniques based on feature extraction and description exist for object recognition. However, if the objects are low textured, only a few number of features can be extracted, making the matching process unreliable. For these cases, we developed a method that combine three characteristics of the objects: color, size and shape, after a 3D detection and segmentation in a plane for each object.

Color information is extracted from the HSV space of the object's RGB pixels and it is represented by the histogram of the Hue components.

The size and shape is estimated from the object's point cloud, which are obtained using an oriented bounding box (OBB) of the points cloud. The shape is characterized using the Hu Moments [5] of the convex hull calculated from the points projected over the plane below them. Thus, with these representations an histogram is obtained for each of the objects and for the recognition process we compare the object's histograms with the histogram of the object to be recognized.

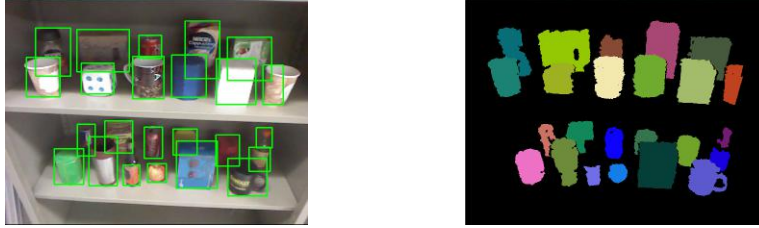
The histograms are compared using histogram intersections, [6]:

$$H(I, M) = \frac{\sum_{j=1}^n \min(I_j, M_j)}{\sum_{j=1}^n M_j} \quad (1)$$

Where I is the histogram of the object to be recognized, M represents one of the stored object's histograms and n is the number of bins in the histograms.

5. CONCLUSIONS AND FUTURE WORK

This method has been tested experimentally, showing fast and robust results for changes in light, scale, and rotation in a plane parallel to the plane below the object. Figure 5 shows an example of object's recognition on a shelf (multiple planes).



(a) Objects can be segmented even with (b) Point clouds corresponding to each segmented object.
occlusions.

Fig. 5: Example of object segmentation on several planes.

Table 1 shows the results obtained by comparing this histogram method with a SIFT algorithm for 25 objects without or almost no texture.

System	% Recognized	% No recognized
Proposed	91.333	7.666
SIFT	22.333	77.666

Table 1: Comparison of recognition of objects using color, size and shape histograms and SIFT.

As we can see for the results on table 1 this technique outperforms the SIFT one.

5 Conclusions and future work

It is clear, that during the 13 years in which our team Pumas has been participated in the RoboCup, 2 years in the Rockin [10] in the category @Home and World Robot Summit 2018, the performance and research developed, in the service robot area, in our laboratory has been improved considerably. Our service robot architecture, the VIRBOT, has been evolving according to the requirement that these robotics competitions asked each year. In terms of software, we have change the way of conceiving the tests of the competition: from static state machines to inferred action planning generated by a rule based system. As for future work, the computer vision algorithms will be improved by using Hidden Markov Models (HMM) to have a better recognition of objects and persons.

5. CONCLUSIONS AND FUTURE WORK

Also, it will be explored fault tolerant systems to help the robot to recover from failures.

References

1. *ViRbot: A System for the Operation of Mobile Robots*, Savage, Jesus and et al, RoboCup 2007: Robot Soccer World Cup XI, pp 512-519, Springer Berlin Heidelberg, 2007.
2. *The Design of Intelligent Agents: A Layered Approach*, Muller, Jorg P, Springer-Verlag New York, Inc.1997.
3. *Parallel implementation of roadmap construction for mobile robots using rgb-d cameras*, Marco Negrete, Jesús Savage, Jesús Cruz, and Jaime Márquez, OGRW2014, pages 184–187, 2014.
4. *Conceptual dependency and its descendants*, Steven L. Lytinen, Computers & Mathematics with Applications, 1992.
5. *Visual pattern recognition by moment invariants*, Ming-Kuei Hu, IRE Transactions on Information Theory, 8(2):179–187, 1962.
6. *Color indexing*, Michael J Swain and Dana H Ballard, International journal of computer vision, 7(1):11–32, 1991.
7. *Surf: Speeded up robust features*, Herbert Bay and Tinne Tuytelaars and Luc Van Gool, In ECCV, pages 404–417, 2006.
8. *Fast Approximate Nearest Neighbors with Automatic Algorithm Configuration*, Marius Muja and David G. Lowe, International Conference on Computer Vision Theory and Application VISSAPP'09, pages 331-340, 2009.
9. *Random Sample Consensus: A Paradigm for Model Fitting with Applications to Image Analysis and Automated Cartography*, Fischler, Martin A. and Bolles, Robert C., Commun. ACM, 24(6):381–395, 1981.
10. <http://rockinrobotchallenge.eu/home.php>