Pumas@Home 2012 Team Description Paper

Jesus Savage, Marco Negrete, Mauricio Matamoros, Ismael Castillo, Luis Contreras, Abel Pacheco, David Esparza, Israel Figueroa, Alejandra Sánchez, Francisco Dorantes, Jesús Cruz, Ángel Molina, Víctor Barradas, Alejandro Macías, and Claudia Favela

Bio-Robotics Laboratory, UNAM, México DF 04510, MEX,
pumasathome@googlegroups.com,
http://biorobotics.fi-p.unam.mx

Abstract. This paper describes an autonomous service robot called Justina built by the team Pumas in the laboratory of Biorobotics at the National University of Mexico. The robot is based on the ViRbot architecture for autonomous mobile robots operation. ViRbot defines a human interaction interface based on Natural Language Processing for understanding voice and gesture commands. After this, a Conceptual Dependence (CD) representation is generated, and an action planner infers a set of rules to perform the commands based on the sensors information and the state of the robot. The aforementioned rules become subtasks that are performed by different submodules that control the hardware and share information by TCP/IP sockets through a Blackboard in order to achieve the given command. By using a good representation of the problem domain through CD's and a rule-based system, such as an inference engine, the operation of the robot becomes a more tractable problem and easier to implement.

1 Introduction

The service robots are hardware and software systems that can assist humans to perform daily tasks in complex environments. To achieve this, a service robot has to be capable of understanding commands from humans, avoiding static and dynamic obstacles while navigating in known and unknown environments, recognizing and manipulating objects and performing other several tasks that the human beings ask for.

The main objective of the ViRbot architecture [9], is to operate autonomous robots that can carry out daily service jobs in houses, offices and factories. This system has been tested in the last five years in the RobCup competition at the categories @Home and will be used again in the competition in Mexico City, 2012, with robot Justina, an autonomous mobile robot, see Figure 1.

Robot Justina integrates several research areas, such as expert systems, computer vision, path and motion planning, robot localization, arm control and place conceptualization.

Section 2 is an overview of the ViRbot Architecture; this system provides a platform for the design and development of robot Justina's software. Section 3 is

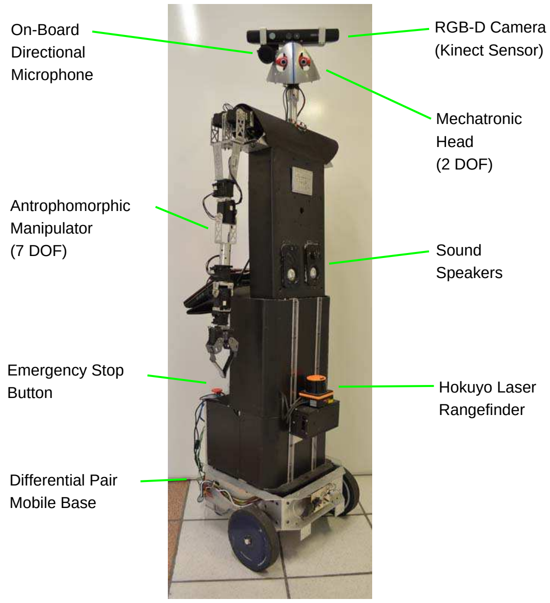

Fig. 1: Robot Justina

about the implementation of ViRbot Architecture as a set of software modules that perform the control of the hardware and a Blackboard for the common data and communication management. In Section 4 we present the conclusions and the future work.

2 ViRbot: A System for the Operation of Mobile Robots

In this system, the operation of a mobile robot is divided in several subsystems, as shown in figure 2. Each subsystem has a specific function that contributes to the final operation of the robot. Some of this layers will be described in this section.

Simulator This subsystem was built for testing new algorithms, training navigation through new scenarios and visualizing of results.

Perception. This module generates beliefs using the symbolic representation generated by the Human/Robot Interface submodule and the Robot's Task submodule configuration.

Robot's Tasks. The set of tasks and subtasks that the robot can accomplish.

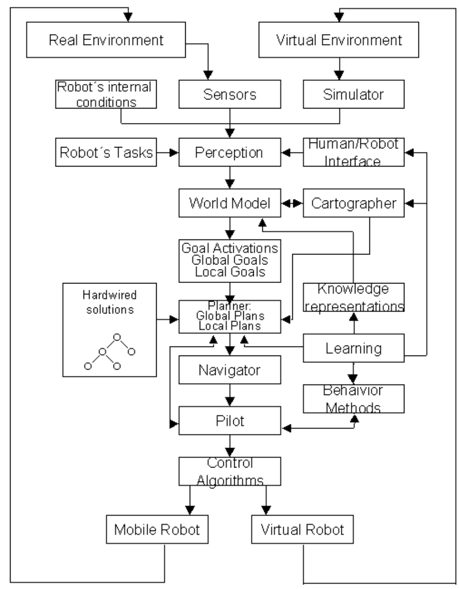

Fig. 2: Architecture of the ViRbot System

Human/Robot Interface. The purpose of this subsystem is to recognize and process the voice and gesture commands. It is divided in three modules:

- **Speech Recognition.** Using digital processing algorithms, it analyzes voice commands given to the robot.
- **Natural Language Understanding.** This module is used to find a symbolic representation of spoken commands given to the robot using the recognized sentences coupled with Conceptual Dependency techniques.
- **Conceptual Dependency.** The Conceptual Dependency (CD) theory was developed by Roger Schank [10] for representing meaning. This technique finds the structure and meaning of a sentence in a single step. One of the main advantages of CDs is that they allow rule based systems to make inferences from a natural language system in the same way humans do. CDs facilitate the use of inference rules because many inferences are already contained in the representation itself. The CD representation uses conceptual primitives, not the actual words contained in the sentence. These primitives represent thoughts, actions, and the relationships between them. CD structures facilitate the inference process by reducing a large number of possible inputs into a small number of actions. This makes CD suitable for the representation of commands or simple questions without a strict sentence grammar, but they are not suitable for the representation of complex sentences.

Cartographer This module has different types of maps for the representation of the environment like:

- **Raw maps.** These are obtained by detecting the position of the obstacles using the robot's laser sensors.
- **Symbolic maps.** These contain all the known obstacles as polygons defined by their vertexes.

The subsystem can also contain topological and probabilistic (Hidden Markov Model) maps of the environment [11].

World Model and Activation of Goals The belief generated by the perception module is validated by the cartographer and the knowledge representation modules, thus a situation recognition is created. Given a situation recognition, a set of goals are activated in order to solve it.

Knowledge Representation A rule based system is used to represent the robot's knowledge, in which each rule contains the encoded knowledge of an expert.

Learning The following learning algorithms are used for the robot:

1. *Map building.*- Fuzzy Clustering Networks are used to locate and represent obstacles in the environment.
2. *Self Localization.*- Hidden Markov Models using Vector Quantization & Self Associative Neural Networks are used to estimate the robot's position and orientation.
3. *Behaviors.*- Genetic Algorithms are used to learn new behaviors.

Task Planner The objective of the action planning is to find a sequence of physical operations to achieve the desired goal. This takes as input the output of the World Model subsystem.

Navigation Determines the robot's position, and controls the robot's movement in order to follow a specified path along the environment. This specified path is given by the Cartographer module.

Hardwired Solutions A set of hardwired procedures is used to partially solve specific problems, including movement, object manipulation, etc.

Behavior Methods A set of solutions encoded in State Machines, which are used in repetitive tasks. For example, in obstacle avoidance, the repetitive task consists of following a sequence of goals until reaching the final one.

Control Algorithms and Real and Virtual Actuators Control algorithms, like PID, are used to control the operation of the virtual and real motors. The virtual and the real robot receive the commands and execute them by interacting with the virtual or real environment and with the user.

3 ViRbot implementation on Robot Justina

ViRbot is implemented in robot Justina by means of a Blackboard architecture. This allows to use more than one platform and run software modules programmed in different languages in different computers. Robot Justina uses computers running both Linux and Microsoft Windows, and modules programmed in C#, C++ and CLIPS.

3.1 Software

Blackboard A general purpose service robot requires several highly specialized systems which in most cases must communicate with each other. Also, even though the Action Planner controls almost all the actions of the robot, it is not its task to manage connections and check the status of each robot subsystems. Therefore, a Blackboard is included: a flexible system to control the transmission of messages between modules, monitor their status and store the common data used by all the modules that integrate the system. All common data is stored in the Blackboard as shared variables to grant access at any time for the others subsystem that need the data to perform their own task. Also the Blackboard offers a flexible platform to implement ViRbot as shown in figure 3.

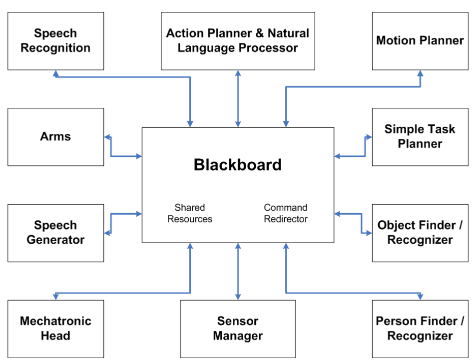

Fig. 3: Blackboard structure

Speech Recognition This module is part of the Human/Robot Interface ViRbot subsystem. It uses the Microsoft Speech Recognition SDK engine to convert voice commands to strings. This allows the use of different grammars using XML notation to define which sentences can be recognized, reducing the number of recognition errors.

Speech Generation This module is part of the ViRbot's Human/Robot Interface and allows the robot to interact with humans by synthesizing voice. It uses the Microsoft Speech API [7].

Action Planner For the Action Planner, the rule-based expert system CLIPS, developed by NASA [8], is used. With CLIPS it is easy to implement tasks of several subsystems of ViRbot Architecture such as the Robot's Tasks, Perception and Knowledge Representation subsystems.

Motion Planner The motion planner is responsible for finding the best sequence of movements in order to reach the final destination given by the Action Planner combining classic and modern techniques. It contains two maps: the Raw map and the Symbolic map, used to calculate the path planning. Parallel to the geometrical representation, it generates a topological representation of the environment and, by Dijkstra algorithm [4], finds the optimal path between the current and goal positions. Obstacle avoidance is achieved using Potential Field and Finite State Machine based behaviors.

Vision The vision subsystem hardware uses the Kinect Microsoft Xbox 360$^{\text{TM}}$[6], which features an RGB camera and a depth sensor. This sensor enables the robot to see objects in three dimensions, which is essential for object manipulation and environment navigation.

Fig. 4: Visual identification of objects

The vision subsystem software features a robust implementation of an object tracker where objects are segmented using receptive fields [3][2][1], and represented by feature points which are described in a multi-resolution framework, that gives a representation of the points in different scales. Detection and description of interest points are based on the SIFT (Scale-Invariant Feature Transform) algorithm [5] after a first segmentation by depth, see figure 4.

3.2 Hardware

Sensors The robot has several sensors for getting information on the surrounding environment: laser range finder for motion planning and obstacle avoidance, a kinect system for seeking humans and objects, a stereo VGA camera for pattern recognition and a directional microphone for natural-speech command interpretation. Digital signal processing techniques are applied to the obtained signals to interact with the dynamic environment.

Differential Pair Mobile Base The robot uses a differential pair mobile base to navigate. A microcontroller-based board sends and receives commands for motor control and encoder reading, using a serial communication interface.

Manipulator The robot has a 7 DOF articulated arm with anthropomorphic configuration built with Dynamixel servomotors and a microcontroller based embedded system for control and movement planning. The arm is designed with 7 DOF in order to perform a better obstacle avoidance and natural human-like movements. For path tracking, a PID plus gravity compensator control is used and a vision based control is implemented for handling objects.

Mechatronic Head The Mechatronic head design is based on the corresponding movements of the human head with 2 degrees of freedom: pan and tilt. This freedom of movement allows to point the sensors to obtain accurate readings of the environment and perform a systematic search in a particular zone of interest.

It carries three different sensing devices on it: a Kinect system, a directional microphone and a VGA camera. The sensors are integrated with a friendly plastic face providing confidence to humans that interact with the robot.

4 Conclusions

The ViRbot system was successfully tested in the Robocup@Home category in the last four RobCup competitions and in Atlanta 2007, the robot obtained the third place in this category. In this years, the full system has been improved having reliable performance and showing promising results.

In the future, a second arm and a mechatronic torso with 2 DOF will be integrated, to enhance object manipulation.

To improve navigation, SLAM technics are being developed using the visual relationship between two different views of the same scene.

Some videos showing the operation of the ViRbot system can be seen at `http://biorobotics.fi-p.unam.mx` .

References

1. P. Chang and J. Krumm. Object recognition with color cooccurrence histogram. In *Proceedings of CVPR '99*, 1999.
2. T.-W. Chen, Y.-L. Chen, and S.-Y. Chien. Fast image segmentation and texture feature extraction for image retrieval. In *Computer Vision Workshops (ICCV Workshops), 2009 IEEE 12th International Conference on*, pages 854 –861, 272009-oct.4 2009.
3. S. Ekvall and D. Kragic. Receptive field cooccurrence histograms for object detection. In *Intelligent Robots and Systems, 2005. (IROS 2005). 2005 IEEE/RSJ International Conference on*, pages 84 – 89, 2005.
4. S. M. LaValle. *Planning Algorithms*. Cambridge University Press, Cambridge, U.K., 2006. Available at http://planning.cs.uiuc.edu/.
5. D. G. Lowe. Distinctive image features from scale-invariant keypoints, 2003.
6. Microsoft. Kinect - xbox.com - xbox.com. `http://www.xbox.com/en-US/kinect`, 11 2010.
7. Microsoft. Speech server - microsoft corporation. `http://www.microsoft.com/speech/speech2007/default.mspx`, 01 2011.
8. G. Riley. *CLIPS Reference Manual Version 6.0. Technical Report Number JSC-25012*. Software Technology Branch, Lyndon B. Johnson Space Center, Houston, TX, 1994.
9. J. Savage, A. Weitzenfeld, F. Ayala, and S. Cuellar. The use of scripts based on conceptual dependency primitives for the operation of service mobile robots. RoboCup 2008 Symposium, Suzhou, China, 2008.
10. R. C. Schank. *Conceptual Information Processing*. North-Holland Publishing Company, 1975.
11. S. Thrun and A. Bücken. Integrating grid-based and topological maps for mobile robot navigation. In *Proceedings of the AAAI Thirteenth National Conference on Artificial Intelligence*, Portland, Oregon, 1996.