Pumas@Home 2013 Team Description Paper

Jesús Savage, Mauricio Matamoros, Marco Negrete, Israel Figueroa, Jesús Cruz, Luis Contreras, Abel Pacheco, and Jaime Márquez

Bio-Robotics Laboratory, UNAM, México DF 04510, MEX,
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-robot interaction interface based on Natural Language Processing, for understanding voice and gesture commands, Conceptual Dependence, Action Planning, based on the sensor's information and the state of the robot, and a set of Subtasks, performed by a Simple Task Planner which coordinates several modules that process information from the sensors and controls the hardware. For simplifying task planning along with real time awareness, all modules communicate with each other through a central module called Blackboard, which supports shared variables, with publisher/subscriber pattern, and message passing.

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 [1], 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 six years in the RoboCup competition at the categories @Home and will be used again in the competition in Eindhoven, The Netherlands in 2013, with robot Justina (see Figure 1).

Robot Justina integrates the work of several research areas, such as expert systems, computer vision, action, 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 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.

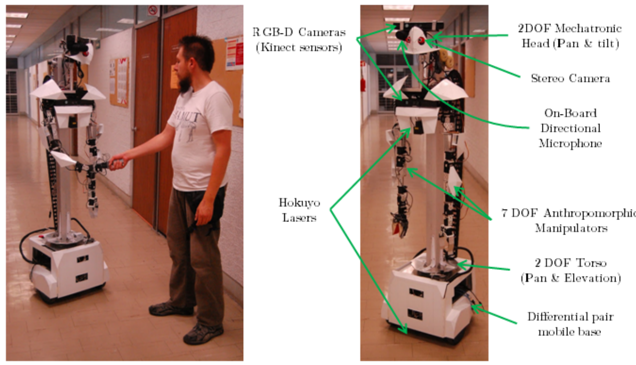


Fig. 1: Robot Justina

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.

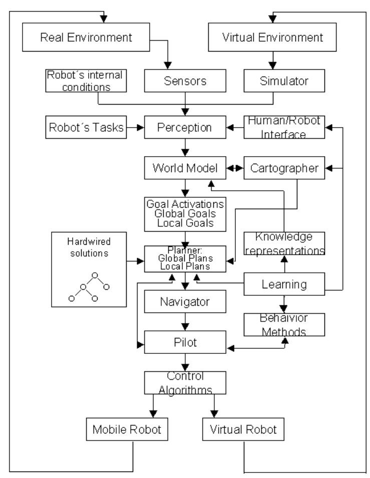


Fig. 2: ViRbot System Architecture

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.

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) [2] is used to represent meaning by finding the structure and meaning of a sentence in a single step. It allows rule based systems to make inferences from a natural language system in the same way humans do, using conceptual primitives that represent thoughts, actions, and the relationships between them.

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 [3].

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.*- Cluster sets are used to locate and represent obstacles and free space 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 This module controls the robot's movement to follow a path through the environment. This specified path is given by the Motion Planner module[4].

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 must communicate with each other, 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 to other modules. It also uses a producer/subscriber paradigm to enhance the Real Time Awareness of the system and reduce communication overloads. Also the Blackboard offers a flexible platform to implement ViRbot as shown in figure 3.

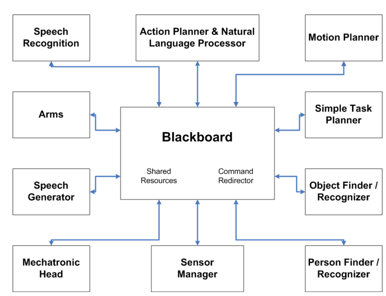


Fig. 3: Blackboard structure

Action Planner For the Action Planner, the rule-based expert system CLIPS, developed by NASA , 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. The Action Planner works in the highest level of abstraction, coordinating the tasks that the robot must perform and choosing the adequate behaviour in each situation.

Simple Task Planner In the ViRbot system, the Hardwired Solutions involve simple and repetitive tasks which requires little or no planning but several transformations and can be easily achieved by state machines (like look for a person/object or grasp an object). Those tasks are perfomed by the Simple Task Planner module. Also this module incorporates reactive behaviours if the Action Planner is not availiable.

Motion Planner The motion planner is responsible for finding the best sequence of movements to reach the final destination given by the Action Planner or the Simple Task Planner combining classic and modern techniques. It uses Cartographer's maps to calculate the path planning. In parallel to the geometrical representation, it generates a topological representation of the environment and, by Dijkstra algorithm [5], finds the optimal path between the current and goal positions. Obstacle avoidance is achieved using Potential Fields and Finite State Machine based behaviors.

Vision Subsystem The Vision Subsytem of the ViRbot architecture has been specialized in five modules: the Kinect module, Person Finder module, the Object Finder module, the Human Pose Detection module and the Visual Localization module.

Object Finder The Object Finder Module features a robust implementation of an object tracker where objects are segmented using receptive fields [6][7][8], and represented by feature points which are described in a multi-resolution framework, that gives a representation of the points in different scales.

(a) Visual identification of objects

(b) Objects are segmentated by depth to reduce compute time and enhance accuracy

Fig. 4: Object recognition and location using depth sensors

Detection and description of interest points are based on the SIFT (Scale-Invariant Feature Transform) algorithm [9] after a first segmentation by depth, see figure 4a. After an object recognition, the geometry of the object is computed from the depth map to fetch the object's centroid and orientation[10].

Kinect The Kinect module uses the Kinect Microsoft Xbox 360™[11], 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. The data acquired from the Kinect sensor is stored in the Blackboard as a RGB bitmap, depth bitmap and human skeleton array to be used by the other vision modules.

Person Finder The Person Finder Module uses VeriLook SDK for multiple face detection and recognition. The name asociated to the detected faces, if known, and it's confidence is stored in the Blackboard.

Human Pose Detection The Human Pose Detection Module uses data from the Kinect module to retrieve the detected human skeletons which are converted to Orthogonal Direction Change Chain Codes [12] which are used as input for a Hidden Markov Model , with a state grammar to detect, recognize and label human movements like stand-up, sit-down, walk-run, etc.

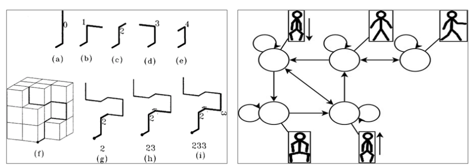
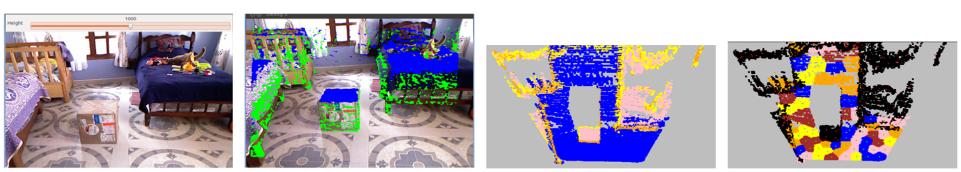


Fig. 5: Human skeletons are processed in a state grammar.

Cartographer The Cartographer module stores the several maps used by the motion planer and also generates topological maps with the 3D points obtained from the depth map of the Kinect module.



(a) Envoronment (b) Plane detection (c) Navigatable area (d) Topological map

Fig. 6: Topologic map generation from 3D points.

Speech Recognition This module is part of the Human/Robot Interface ViRbot subsystem and uses the MS SAPI 5.3. An array of hypothesized strings with its confidences is stored in the Blackboard to let the Action Planner to choose the most adequate candidate for the robot's current context.

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 SAPI 5.3 with the Loquendo Susan voice.

Sensor data processing The Sensors Data Processing module contorls the data acqusition from several sensors like lasers and sonars. The acquired data is stored in the Blackboard to be used by other modules.

3.2 Hardware

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 two 7 DOF articulated arm with anthropomorphic configuration built with Dynamixel servomotors and a microcontroller based embedded system for control and trajectory planning. The arm has an antropomorphic design 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 Torso The robot has a Mechatronic Torso with 2 DOF for control the elevation and pan of the arms and head. The torso increases the grasping range of the manipulators and allow the robot to look straight to both, small and tall humans.

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.

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 stereo 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 RoboCup competitions and in Atlanta 2007, the robot obtained the third place in this category. In these years, the full system has been improved having reliable performance and showing promising results. In the future, the structure of the robot will be redesigned to reduce it's weight and make its assembly and maintenance easier. Also, new algorithms for the recognition of human faces and objects, along with localization, 3D mapping and more complex behaviours arbitred by the Action Planner's expert system.

To improve navigation, SLAM technics are being developed using the visual relationship between two different views of the same scene. Also, to provide compatibility with the most used framework for robots, a bridge between the Blackboard and ROS is being developed.

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

References

1. Jesús Savage, Alfredo Weitzenfeld, Francisco Ayala, and Sergio Cuellar. The use of scripts based on conceptual dependency primitives for the operation of service mobile robots. RoboCup 2008 Symposium, Suzhou, China, 2008.
2. Roger C. Schank. *Conceptual Information Processing*. North-Holland Publishing Company, 1975.
3. 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.
4. Adalberto Llarena, Jesus Savage, Ángel Fernando Kuri Morales, and Boris Escalante-Ramírez. Odometry-based viterbi localization with artificial neural networks and laser range finders for mobile robots. *Journal of Intelligent and Robotic Systems*, 66(1-2):75–109, 2012.
5. S. M. LaValle. *Planning Algorithms*. Cambridge University Press, Cambridge, U.K., 2006. Available at http://planning.cs.uiuc.edu/.
6. 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.
7. Tse-Wei Chen, Yi-Ling Chen, and Shao-Yi 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.
8. Peng Chang and John Krumm. Object recognition with color cooccurrence histogram. In *Proceedings of CVPR '99*, 1999.
9. David G. Lowe. Distinctive image features from scale-invariant keypoints, 2003.
10. José Figueroa, Luis Contreras, Abel Pacheco, and Jesus Savage. Development of an object recognition and location system using the microsoft kinecttm sensor. In *RoboCup*, pages 440–449, 2011.
11. Microsoft. Kinect - xbox.com - xbox.com. `http://www.xbox.com/en-US/kinect`, 11 2010.
12. José Figueroa, Jesus Savage, Ernesto Bribiesca, and Enrique Succar. Recognition of static gestures using three-dimensional chain codes using dominant direction vectors. In *Intelligent Environments (Workshops)*, pages 231–241, 2012.