PUMAS 2011 Team Description Paper

Jesus Savage, Mauricio Matamoros, Israel Figueroa, Marco Negrete, Francisco Dorantes, Ismael Castillo, Luis Munguia, Luis Contreras, Abel Pacheco, David Esparza, Alejandra Sanchez, Marco Becerra

Bio-Robotics Laboratory, Department of Electrical Engineering
Universidad Nacional Autónoma de México, UNAM
Email: savage@servidor.unam.mx

Abstract. This paper describes the team PUMAS from the University of Mexico, UNAM, this team had participated in the RoboCup@Home since the first competition in Bremen in 2006, then in Atlanta in 2007, where our team obtained the third place, we also participated in China in 2008 and also in Graz, Austria in 2009, where our team qualified for the second stage of the competition, and in Singapore in 2010. Our robot uses a robotics architecture, the ViRbot, used to control the operation of service mobile robots. It accomplish the required commands using AI actions planning and reactive behaviors with a description of the working environment. In the ViRbot architecture the actions planner module uses Conceptual Dependency (CD) primitives as the base for representing the problem domain. After a command is spoken to the mobile robot a CD representation of it is generated, a rule based system takes this CD representation, and using the state of the environment generates other subtasks represented by CDs to accomplish the command. By using a good representation of the problem domain through CDs and a rule based system as an inference engine, the operation of the robot becomes a more tractable problem and easier to implement.

1 Introduction

There is a need for reliable Human-Robot interaction systems as experienced by the proliferation of new humanoid robots, particulary in Japan, with advanced interaction capabilities including spoken language. This can also be appreciated by new competitions to exalt the robots social interaction with humans such as the new league in the robots' competition Robocup: RoboCup@Home. The goal of this league is to promote the development of real-world applications and human-machine interaction with autonomous robots, or as the competition organizers put it: "The aim is to foster the development of useful robotic applications that can assist humans in everyday life" [11]. In this paper is presented a mobile robot architecture, the ViRbot system, whose goal is to operate autonomous robots that can carry out daily service jobs in houses, offices and factories. The ViRbot system has been tested in the last four Robocup@Home category in the Robocup competition; it will be used again in the same competition in Istanbul, 2011, with the robot PAC-ITO, see figure 1.

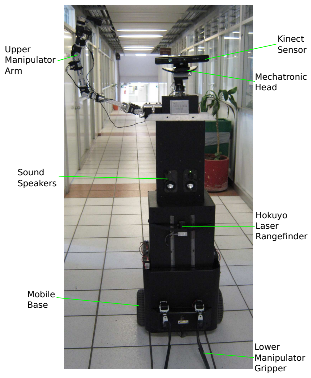

Fig. 1: Robot PAC-ITO, this robot has, for sensing the environment, a laser measurement system, a microphone and a vision system. Also it has two manipulators to pick objects.

The ViRbot system [12] divides the operation of a mobile robot in several subsystems, see figure 2. Each subsystem has a specific function that contributes to the final operation of the robot.

Each of the layers of figure 2 will be described in the following sections.

2 Hardware

2.1 Manipulators

The robot has two manipulators. The upper one is a 7 DOF articulated arm with anthropomorphic configuration built with Dynamixel servomotors. The arm was designed with 7 DOF in order to perform obstacle avoidance and natural human movements. For path tracking, a PDI plus gravity compensator control is used and a vision based control is implemented for handling objects. The lower manipulator has 2 DOF and is used for picking up objects from the floor.

2.2 Mechatronic Head

The Mechatronic head has similar movements as the human head with 3 degrees of freedom: pan and tilt in the head, and tilt in the neck. It carries two different sensing devices on it, a Kinect system and a camera. The mechatronic head has eyebrows that allows some facial expressions of the robot.

2.3 Internal Sensors

The robot used in this research have the following internal sensors that reflect its internal state: wheel encoders and battery level sensor. Each of the sensors' values can be read any time during the operation of the robot.

2.4 External Sensors

The robot has the following external sensors that sense the surrounding environment: infrared, laser, video-cameras and one microphone. Digital signal processing techniques are applied to the signals obtained from the video-cameras and microphones to be used in pattern recognition algorithms.

Vision Subsystem The vision subsystem hardware uses the Kinect sensor for Microsoft Xbox 360$^{\text{TM}}$[7], which features an RGB camera and a depth sensor, the latter enables the robot to see objects in three dimensions, which is essential for object manipulation and environment navigation.

The vision subsystem software features a robust implementation of an object tracker where objects are segmented using receptive fields [4][3][2], and represented by feature points which are described in a multi-resolution framework, that gives a representation of the points in different scales. The interest points detection and description was based in the SIFT (Scale-Invariant Feature Transform) algorithm [6], see figure 3.

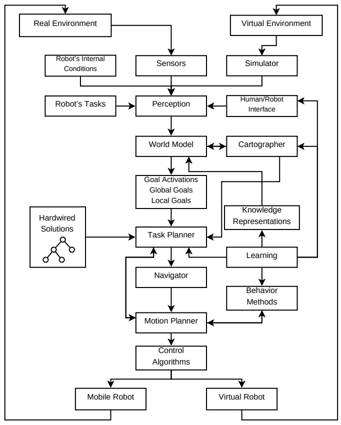

Fig. 2: The ViRbot System consists of several subsystems that control the operation of a mobile robot.

3 Software

3.1 Simulator

The ViRbot system contains a simulator that provides the values of the internal and external sensors that the robot has, each simulated sensor has a mathematical model. Also the simulation of new sensors can be incorporated easily.

3.2 Robot's Tasks

Set of tasks that the robot needs to accomplish according to the time when it was programmed.

3.3 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 of the robot's systems. Therefore, a

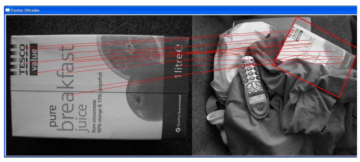

Fig. 3: The vision subsystem tracks and identifies objects, using both color and depth information.

Blackboard it is included: a flexible system to coordinate the pass of messages between the modules and it monitors the operation of each them, see Fig.4.

3.4 Human/Robot Interface

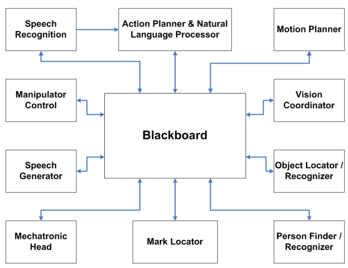

Fig. 4: Blackboard structure

The Human/Robot Interface subsystem in the ViRbot architecture has three modules: Natural Language Understanding, Speech Generation and Robot's Facial Expressions.

Natural Language Understanding The natural language understanding module finds a symbolic representation of spoken commands given to a robot.

It consists of a speech recognition system coupled with Conceptual Dependency techniques [13].

Speech Recognition For the speech recognition system it is used the Microsoft Speech SDK engine. It allows the use of grammars, that are specified using XML notation, which constrains the sentences that can be uttered and with that feature the number of recognition errors is reduced.

Conceptual Dependency Conceptual Dependency is a theory developed by Roger Schank for representing meaning. This technique finds the structure and meaning of a sentence in a single step. CDs are especially useful when there is not a strict sentence grammar.

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 beings do. CDs facilitate the use of inference rules because many inferences are already contained in the representation itself. The CD representation uses conceptual primitives and not the actual words contained in the sentence. These primitives represent thoughts, actions, and the relationships between them. Some of the more commonly used CD primitives are, as defined by Schank:

ATRANS: Transfer of ownership, possession, or control of an object (e.g. give.); PTRANS: Transfer of the physical location of an object (e.g. go.); ATTEND: Focus a sense organ (e.g. point.); MOVE: Movement of a body part by its owner (e.g. kick.); GRASP: Grasping of an object by an actor (e.g. take.); PROPEL: The application of a physical force to an object (e.g. push.); SPEAK: Production of sounds (e.g. say.).

Each action primitive represents several verbs which have similar meaning. For instance give, buy, and take have the same representation, i.e., the transference of an object from one entity to another.

For example, in the sentence “Robot, put the milk in the kitchen”, when the verb “put” is found, a PTRANS structure is issued. PTRANS encodes the transfer of the physical location of an object, and it has the following representation:

(PTRANS (ACTOR NIL) (OBJECT NIL) (FROM NIL) (TO NIL))

The empty (NIL) slots are filled by finding relevant elements in the sentence. So the actor is the robot, the object is the milk, and the robot will go from the front door to the kitchen (assuming the robot was initially in the front door). The final PTRANS representation is:

(PTRANS (ACTOR Robot) (OBJECT Milk) (FROM Front-Door) (TO Kitchen))

CD structures facilitate the inference process, by reducing the large number of possible inputs into a small number of actions. The final CDs encode the users commands to the robot. To carry out these commands an actions planning module must be used as described in the following section. CDs are suitable for the representation of commands and for asking simple questions to a robot, but they are not suitable for the representation of complex sentences.

Speech Generation and Robot’s Facial Expressions The text to speech generation system used is the Microsoft Speech API [8], that is freely available on every installation of Microsoft Windows XP.

In human to human communication, facial expression plays an important role, so we consider that the same thing applies to human to robot communication. Thus, our robot contains a mechatronic head that shows simple expressions through its movements , the opening of its mouth, the movement of its eyes and by modifying its eyebrows. The eyebrows are created using an array of LEDs that are turned on and off that creates different face expressions.

When the expression of the face changes the robot is ready for accepting commands, when it understood them and also when is executing them, etc.

3.5 Perception

The perception module obtains a symbolic representation of the data coming from the users and the robot’s sensors. The symbolic representation is generated by applying digital processing algorithms on the data coming from the sensors. With this symbolic representation, a belief is generated.

3.6 Cartographer

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

Raw maps are obtained by detecting the position of the obstacles using the robot's laser sensors; **Symbolic maps** contain each of the known obstacles defined as polygons, that consists of a clockwise ordered list of its vertexes. The Cartographer subsystem contains also topological and probabilistic (Hidden Markov Model) maps of the environment [14].

3.7 Knowledge Representation

A rule based system is used to represent the robot's knowledge, that is represented by rules, each one contains the encoded knowledge of an expert. In the ViRbot the rule based system CLIPS is used, an expert system shell, freely available and developed by NASA [10].

3.8 Navigation

The robot map captures the geometry of the environment in form of lines and rectangles, wich correspond to the existing objects and walls. Parallel to this geometrical representation, is built the topology of the environment, where a Graph is used to represent the conceptual structure of the inner division of regions accord to their functionalities. The nodes of the graph records the position in te metrical map of the individual regions in the environment, like kitchen, rooms, hall, etc., while the edges represent the connection between such regions. This representation is used to find a path between two different regions, by performing the Dijkstra algorithm [5] over the topological map betweeen the two corresponding nodes. The navigation task is performed by following the path founded by the Dijkstra algorithm. The path following module drives the robot over each sub-goal in reactive form using the Potential Fields method [1], where the robot is subject to repulsive and attractive forces, that is, each goal node should attract the robot to it's position, and obstacles should repulse it from them. We also use the Potential Fields in combination with robot behaviors to avoid robot traps created by non-convex obstacles.

3.9 World Model and Activations 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.

3.10 Hardwired Solutions

Set of hardwired procedures that solve, partially, specific problems. Procedures for movement, transference of objects, pick up, etc.

3.11 Task Planner

After receiving the CD representation from the Human/Robot interface the Perception subsystem perceives a new situation that needs to be validated by the World Model subsystem. The World Model validates the situation by the information provided by the Cartographer and the Knowledge Representation subsystem. The Planner subsystem takes as an input the output of the World Model subsystem and tries to take care of the situation presented. Planning is defined as the process of finding a procedure or guide for accomplishing an objective or task. In the ViRbot planner subsystem there are two planning layers, the upper is the actions planner, based on a rule based system, and the lower layer the movements planner, based on the Dijkstra algorithm.

3.12 Actions Planner

The Robot is able to perform operations like grasping an object, moving itself from on place to another, etc. Then the objective of action planning is to find a sequence of physical operations to achieve the desired goal.

3.13 Movements Planner

If the command asks the robot to go from one room to another the movements planner finds the best sequence of movements between rooms until it reaches the final destination. Inside of each room the movements planner finds also the best movement path considering the known obstacles, that represent some of the objects in the room. Thus, the movements planner uses this information to find the best path avoiding the obstacles that interfere with the goal.

3.14 Behavior Methods

After the movements planner finds the nodes where the robot needs to go, the Behavior subsystem tries to reach each of them, if it finds unexpected obstacles during this process it avoids them. The Behavior subsystem consists of behaviors based on potential fields methods and state machines, all these controls the final movement of the robot [9].

4 Learning

The following learning algorithms are used for our robot:

a. *Map building.*- Fuzzy Clustering Networks are used to locate and represent obstacles in the environment.
b. *Self Localization.*- Hidden Markov Models using Vector Quantization & Self Associative Neural Networks are used to find an estimation of the robot's position and orientation.
c. *Behaviors.*- Genetic Algorithms are used to learn new behaviors.

5 Conclusions

The ViRbot system was tested in the Robocup@Home category in the last four Robocup competitions, in Atlanta 2007, our robot TPR8 obtained the third place in this category. Some videos showing the operation of the Virbot system can be seen in `http://biorobotics.fi-p.unam.mx`.

References

1. R. C. Arkin. *Behaviour Based Robotics Intelligent Robots and Autonomous Agents*. Boston MA: MIT Press, 1998.
2. P. Chang and J. Krumm. Object recognition with color cooccurrence histogram. In *Proceedings of CVPR '99*, 1999.
3. 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.
4. 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.
5. S. M. LaValle. *Planning Algorithms*. Cambridge University Press, Cambridge, U.K., 2006. Available at http://planning.cs.uiuc.edu/.
6. D. G. Lowe. Distinctive image features from scale-invariant keypoints, 2003.
7. Microsoft. Kinect - xbox.com - xbox.com. `http://www.xbox.com/en-US/kinect`, 11 2010.
8. Microsoft. Speech server - microsoft corporation. `http://www.microsoft.com/speech/speech2007/default.mspx`, 01 2011.
9. J. P. Müller, M. Pischel, and M. Thiel. Modeling reactive behaviour in vertically layered agent architectures. In *Proceedings of the workshop on agent theories, architectures, and languages on Intelligent agents*, pages 261–276, New York, NY, USA, 1995. Springer-Verlag New York, Inc.
10. G. Riley. *CLIPS Reference Manual Version 6.0. Technical Report Number JSC-25012*. Software Technology Branch, Lyndon B. Johnson Space Center, Houston, TX, 1994.
11. Robocup@Home. Robocup@home. http://www.ai.rug.nl/robocupathome/, 2006.
12. 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.
13. R. C. Schank. *Conceptual Information Processing*. North-Holland Publishing Company, 1975.
14. 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.