CAMBADA@Home 2011
Team Description Paper

J. L. Azevedo, C. Cruz, J. Cunha, M. B. Cunha, N. Lau, C. Martins,
A. J. R. Neves, E. Pedrosa, A. Pereira, A. J. S. Teixeira, M. Antunes

Transverse Activity on Intelligent Robotics
IEETA/DETI – University of Aveiro
3810-193 Aveiro, Portugal

Abstract. This paper presents the newly created CAMBADA@Home team for the purpose of qualification to the RoboCup@Home competition in RoboCup'2011. The CAMBADA@Home team is built on the success of the CAMBADA robotic soccer team which competes in the Middle Size League.

1 Introduction

The CAMBADA[1]@Home is the University of Aveiro RoboCup@Home team. The project was created in January 2011 following the team past experience in the CAMBADA [1] robotic soccer team. The development of the CAMBADA soccer team started in 2003 and a steady progress was observed since then.

CAMBADA has participated in several national and international competitions, including RoboCup world championships (5^{th} place in 2007, 1^{st} in 2008 and 3^{rd} in 2009 and 2010), the European RoboLudens, German Open (2^{nd} in 2010) and the annual Portuguese Open Robotics Festival (3^{rd} place in 2006, 1^{st} in 2007, 2008, 2009 and 2010). Moreover, the CAMBADA team achieved excellent results in the mandatory technical challenge of the RoboCup MSL: 2nd place in 2008 and 1st place in 2009.

The CAMBADA@Home team is comprised of students from the Department of Electronics, Telecommunications and Informatics and researchers from the Institute of Electronics and Telematics Engineering of Aveiro.

This paper describes the current development stage of the team and is organized as follows: Section 2 presents the CAMBADA@Home robotic platform. Section 3 describes the general architecture of the robots. Section 4 presents the current version of the vision system. Section 5 addresses the indoor localization algorithm used by the robot. Section 6 describes the monitoring station application. Section 7 presents the Human-Robot interaction system and, finally, Section 8 concludes the paper.

---

1 CAMBADA is an acronym of Cooperative Autonomous Mobile roBots with Advanced Distributed Architecture.

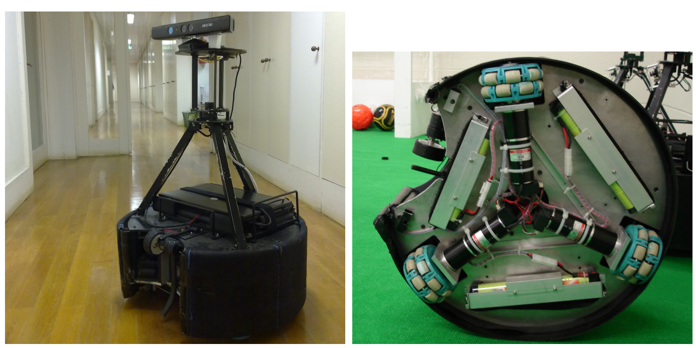
2 Robot Platform

The CAMBADA@Home robotic platform is based on the CAMBADA robotic soccer platform. The robot has a conical base with radius of 24 cm and height of 80 cm. The physical structure is built on a modular approach with three main modules or layers.

The top layer has the robot vision system. Currently the robot uses a low-cost sensor, the Microsoft Kinnect depth camera. This sensor provides rich information about the surrounding environment being a good alternative to more costly sensors such as laser range finders or time-of-flight cameras.

The middle layer houses the processing unit, currently a 13" laptop, which collects data from the sensors and computes the commands to the actuators. The laptop executes the vision software along with all high level and decision software and can be seen as the brain of the robot. Beneath the middle layer, a network of micro-controllers is placed to control the low-level sensing/actuation system, or the nervous system of the robot. The sensing and actuation system is highly distributed, using the CAN protocol, meaning the nodes in the network control different functions of the robot, such as motion, odometry and system monitoring.

Finally, the lowest layer is composed of the robot motion system. The robots move with the aid of a set of three omni-wheels, disposed at the perifery of the robot at angles that differ 120 degrees from each other, powered by three 24V/150W Maxon motors (Figure 1).



a) **b)**

Fig. 1. CAMBADA@Home hardware system: **a)** The robot platform. **b)** Detailed view of the motion system.

Being the project in its infancy, the team decided not to include a robotic arm in the robot platform for the moment. This allows the team to focus on the basic skills of a mobile domestic robot, such as indoor localization, safe navigation and human-robot interaction. It is our firm belief that with this focused approach the team will be able to tackle a subset of the competion challenges while obtaining a remarkable performance in its first participation.

3 Software Architecture

As with the hardware platform, the CAMBADA software architecture was successfully adapted to the RoboCup@Home environment.

Following the CAMBADA hardware approach, the software is also distributed. Therefore, five different processes are executed concurrently. All the processes run at the robot's processing unit in Linux.

All processes communicate by means of the RTDB[2] which is physically implemented in shared memory. The RTDB [2] is a data structure which contains the essential state variables to control the robot. The RTDB is divided in two regions, the local and shared regions. The local section holds the data local to the processes and is not broadcasted. The shared section is further divided to contain the data of the world state as perceived by a team of agents. One of the areas is written by the robot itself and broadcasted to the rest of the team while the other remaining areas store the information received from the other team-mates.

The processes composing the CAMBADA@Home software are (Figure 2):

Vision which is responsible for acquiring the visual data from the Kinect sensor.

Agent is the process that integrates the sensor information and constructs the robot's worldstate. The agent then decides the command to be applied, based on the perception of the worldstate.

Comm handles the inter-robot communication, receiving the information shared by the team-mates and transmitting the data from the shared section of the RTDB to the team-mates.

HWcomm or hardware communication process is responsible for transmitting the data to and from the low-level sensing and actuation system.

Monitor that checks the state of the remaining processes, relaunching them in case of abnormal termination.

Given the real-time constraints of the system, all process schedulling is handled by a library specifically developed for the task, the *Process Manager*.

4 Vision System

Humans rely heavily on vision or vision based abstractions to acknowledge the world, to think about the world and to manipulate the world. It is only logical to base the sensing of the world of an artificial agent in an artificial vision system with capabilities similar to the human vision system.

The vision subsystem of this robot is constituted by a single Kinect, accessed with the freenect [3] library, which provides a depth and colour view of the

[2] Real-Time DataBase

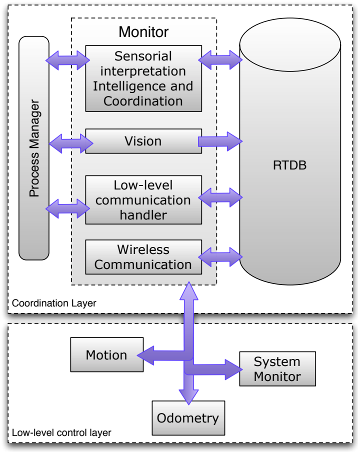


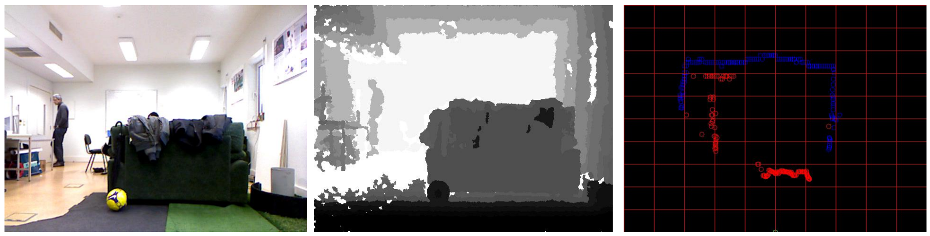
Fig. 2. CAMBADA@Home software architecture.

world. By using the depth sensor, it is possible to create an occupation map of the environment, which is later used to perform wall and obstacle detection. By creating a correspondence between the colour image and the depth image, it is also possible to add colour to the *point cloud* obtained from the depth sensor alone. All of the robot image processing is performed using the OpenCV [4] library when possible.

Currently, the vision subsystem is only being used to detect the nearest obstacles and the walls (Figure 3). This detection includes selection of likely points and later filtering of those points running only over a predefined set of interest points allowing the reduction of the needed processing time. The used methods are based on the analysis of the image by column, where the farthest point is considered as wall and the closest point is considered as obstacle. A point is effectively an obstacle if its height, relative to the floor, is less than the robot height.

Although the current wall detection is good enough to be fed into the localization algorithm, giving robust results, there is the opportunity to explore different localization methods suitable for indoor environments, one of them being the localization based on landmarks. Another related challenge is to be able to learn new objects and to later recognize them. To achieve this capabilities, the ability to learn new objects, preferably on-line, and to detect them when coming upon them in the world will be developed.

The vision subsystem objectives are not only to help the interaction with the world of the people but also with the people of the world. In order to do that, the capabilities of people detection and face detection and recognition will be developed. These capabilities will help the robot to know new people and recognize them later as well as to follow someone when requested. The depth sensor allows a segmentation method based in brightness discontinuity and



a) b) c)

Fig. 3. CAMBADA@Home vision system: **a)** The image captured by the Kinect rgb camera. **b)** The same image captured by the Kinect depth camera. **c)** The 2D vision of the extracted information of the depth camera, the blue points are walls and the red points are obstacles.

depth, which can be used to detect/extract individual objects from an image. The objects can be classified and matched with the proper category. The technique actually used to classify objects is SIFT [5], but based in time constraints a technique designed by SURF [6] can be more suitable.

5 Localization

For indoor localization, the team successfully adapted the localization algorithm used by the CAMBADA team to estimate a robot position in a robotic soccer field. The algorithm was initially proposed by the Middle Size League team Brainstormers Tribots [7].

The Tribots algorithm constructs a FieldLUT from the soccer field. A FieldLUT is a grid-like data structure where the value of each grid is the distance to the closest field line. The robot detects the relative position of the white lines points through vision and tries to match the seen points with the soccer field map, represented by the FieldLUT. Applying the gradient over the FieldLUT yields the direction to the closest line. The robot is able to localize itself in the field applying the weighted sum of gradients of the all line points extracted from the camera images. Using gradient descent the robot can iteratively improve the estimation of its pose in the field. After this visual optimization process the robot pose is integrated with the odometry data in a Kalman Filter for a refined estimation of the robot pose.

To apply the aforementioned algorithm in the RoboCup@Home environment the CAMBADA@Home team implemented an initial phase, that given a map of the environment, usually the building blueprints, determines which walls are seen from a given point. From the set of walls seen for every given point a FieldLUT is built. This results in a set of FieldLUTs that are dynamically loaded as the robot moves through the environment.

The need to use a set of FieldLUTs instead of a single FieldLUT for the entire map arises from the local minimum problem inherent to gradient descent algorithms. Since the walls in a domestic environment have an associated height which is naturally higher than the robot, from a given point in the environment there is usually a set of wall that are out of the line-of-sight of the robot. This scenario doesn't occur in a robotic soccer field where the lines are co-planar with the field. Therefore using a single FieldLUT in the RoboCup@Home environment could match the wall points extracted from the captured images to unseen walls, resulting in erroneous self-localization.

6 Monitoring station

The monitoring station, also known as basestation, has a determinant role both during the development of an autonomous assistant robot capability as well during its application. The basestation is an adapted version of the CAMBADA team basestation [8] taking in consideration a set of requirements that emerge from the develoment of a service and assistive robot (Figure 4).

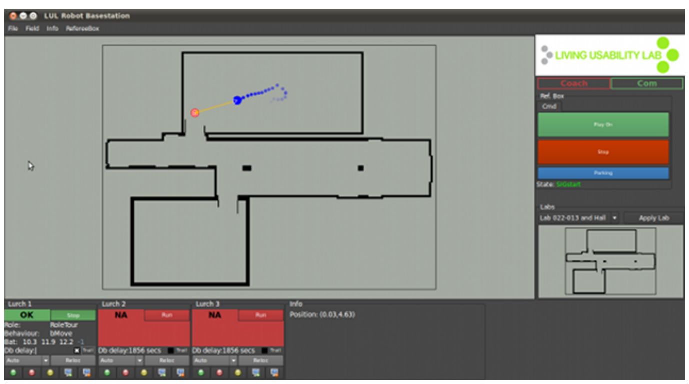


Fig. 4. CAMBADA@Home basestation GUI.

The basestation application provides a set of tools to perform the control and monitoring of the robot. Regarding the control activity, this application allows high level control of the robot by sending basic commands such as *run*, *stop* and *docking*. It also provides a high level monitoring of the robot internal state, namely its batteries status, current role and behavior, indoor self-localization, current destination point, breadcrumb trail, etc.

Furthermore, this application provides a mechanism that can be used to enforce a specific behavior of the robot, for debugging purposes.

7 Human-robot interaction

Spoken language is a natural way – possibly the most natural - to control and process human-robot interaction. It has some important advantages: eyes and hands free; communication from a distance, even without being in line of sight; no need for additional learning for humans.

Therefore, we will integrate in our mobile service robot some interaction facilities by means of three spoken and natural language processing components: an Automatic Speech Recognition (ASR) component to process the human requests (in form of command-like small sentences), a Text-to-Speech (TTS) component to generate more natural responses from the robot side, and a semantic framework (dialog manager) to control how these two components work together.

The requirements for this spoken and natural language interaction system result from the rulebook of the RoboCup@Home competition. An example of a use-case is the *Follow Me* task where the robot is asked to follow user. In this use-case two command-like sentences are needed: "[Robot's Name] follow me" and "[Robot's Name] stop follow me".

According the use-cases the following requirements for our speech-based interaction system are defined:

– The speech recognition component should be speaker independent, have a small vocabulary, and be context dependent and robust against stationary and non-stationary environmental noise.
– The speech output should be intelligible and sound natural.
– The dialog manager system should be mixed-initiative allowing both robot and user to start the action, provide or ask for help if no input is received or incorrect action is recognized, and ask for confirmation in case of irreversible actions.

In terms of hardware three types of input systems are being tested: a robot mounted microphone, an appropriate Bluetooth headset and a microphone array framework with noise reduction and echo cancellation. To deal with the high amount of non-stationary background noises and background speech usually present in these interaction environments, a close speech detection framework is applied in parallel to noise robust speech recognition techniques.

Speech recognition will be accomplished through the use of CMUSphinx, an Open Source Toolkit for Speech Recognition project by Carnegie Mellon University. Additionally, we are testing speech recognition results obtained by using the Microsoft Speech SDK. For this propose both speaker dependent and speaker independent profiles are being trained, and a specific grammar for command interaction defined, with each command-like sentence preceded by a predefined prefix (robot's name).

Finally, and to better control the overall communication process, a small and custom based dialog manager system is being defined.

For robot speak-back interaction and user feedback, external speech output devices (external speakers) will be used. The speech synthesis component will be implemented by means of a concatenative system for speech output. For that propose, we plan to use the Microsoft Speech SDK and the FESTIVAL Speech Synthesis system developed at the Edinburgh University. We are trying to implement some adaptation features like using the information on distance from robot to user to dynamically change the output volume, and changing the TTS rate from normal to slower according to user's age.

8 Conclusion

This paper presents the current stage of development of the CAMBADA@Home project. Although being the project at a very early stage the team has already obtained visible results. The CAMBADA platform was successfully adapted to the RoboCup@Home environment allowing the development of vision, self-localization and navigation algorithms that enable the robot to successfully move in a home scenario using pre-defined waypoints. Aditionally the CAMBADA basestation application was adapted to aid the development and monitoring of the robot. The team plans to build on these initial results in order to successfully complete the RoboCup@Home challenges at the RoboCup'2011 in Istanbul.

Acknowledgements

This work is part of the COMPETE - Programa Operacional Factores de Competitividade and the European Union (FEDER) under QREN Living Usability Lab for Next Generation Networks (http://www.livinglab.pt/).

References

1. Neves, A.J.R., Azevedo, J.L., Cunha, B., Lau, N., Silva, J., Santos, F., Corrente, G., Martins, D.A., Figueiredo, N., Pereira, A., Almeida, L., Lopes, L.S., Pinho, A.J., Rodrigues, J.M.O.S., Pedreiras, P.: CAMBADA soccer team: from robot architecture to multiagent coordination. In: Robot Soccer. INTECH (January 2010) 19–45
2. Almeida, L., Santos, F., Facchinetti, T., Pedreiras, P., Silva, V., Lopes, L.S.: Coordinating distributed autonomous agents with a real-time database: The cambada project. In Aykanat, C., Dayar, T., Korpeoglu, I., eds.: ISCIS. Volume 3280 of Lecture Notes in Computer Science., Springer (2004) 876–886
3. libfreenect homepage: Available `http://openkinect.org/wiki/Main_Page` Accessed in February 2011.
4. Bradski, G.: The OpenCV Library. Dr. Dobb's Journal of Software Tools (2000)
5. Lowe, D.G.: Object recognition from local scale-invariant features. In: ICCV. (1999) 1150–1157
6. Bay, H., Ess, A., Tuytelaars, T., Gool, L.J.V.: Speeded-up robust features (surf). Computer Vision and Image Understanding **110**(3) (2008) 346–359
7. Lauer, M., Lange, S., Riedmiller, M.: Calculating the perfect match: An efficient and accurate approach for robot self-localization. In Bredenfeld, A., Jacoff, A., Noda, I., Takahashi, Y., eds.: RoboCup. Volume 4020 of Lecture Notes in Computer Science., Springer (2005) 142–153
8. Figueiredo, N.M., Neves, A.J.R., Lau, N., Pereira, A., Corrente, G.A.: Control and monitoring of a robotic soccer team: The base station application. In Lopes, L.S., Lau, N., Mariano, P., Rocha, L.M., eds.: EPIA. Volume 5816 of Lecture Notes in Computer Science., Springer (2009) 299–309