SPL Portuguese Team: Team Description Paper for RoboCup 2012

A. J. R. Neves[1], N. Lau[1], Luís Paulo Reis[2], António P. Moreira[3], João Silva[1], Alina Trifan[1], Bruno Pimentel[1], Carlos Sobrinho[1], Edgar Domingues[1], Nima Shafii[2], Seraphin Miranda[2], Luis Cruz[2]

[1]IEETA/DETI – University of Aveiro, 3810-193 Aveiro, Portugal
[2]LIACC/FEUP – Faculty of Engineering of the University of Porto, Portugal
[3]INESC-P – University of Porto, Portugal

Abstract. Portuguese Team intends to participate in the Standard Platform League at the RoboCup 2012 for the second time, after a first participation in 2011. The team is composed of researchers from two universities, joining previous experience in 3 teams in RoboCup: FC Portugal (Simulation 2D, Simulation 3D, Coach Competition, Rescue Simulation, Rescue Infrastructure and Physical Visualization/Mixed Reality), CAMBADA (Middle Size League and RoboCup@HOME) and 5DPO (Small-Size and Middle-Size leagues). Concerning scientific results achieved by the three teams, its members have a combined publishing rate of more than 100 papers regarding Robotic Soccer and related issues in international journals and conferences. This paper describes the Portuguese Team for the purpose of the qualification for RoboCup 2012.

1 Introduction

Portuguese Team is a RoboCup Standard Platform League (SPL) team composed by members of the three Portuguese teams that achieved best results in RoboCup European and World championships: FC Portugal, CAMBADA and 5DPO.

Based on previous experience, Portuguese Team developed an agent similar to the one developed for FCPortugal 3D and uses a similar distributed architecture as the one used in the CAMBADA robots. This distributed architecture is based on several processes, namely Communications, Vision and Agent, centered in a Real-time database. The agent is based on several modules, each one with a specific purpose: WorldState, AgentModel, Geometry, Optimization, Skills, Utils, Strategy and DeviceManager.

This paper describes the current development stage of the team and is organized as follows: Section 2 describes the general architecture of the software running on the robot. Section 3 describes the vision system developed for the NAO robot. Section 4 describes general implementation lines of the agent, adapted from the simulation agent. Finally, Section 5 presents some conclusions, followed by a few of the main related publications.

Statement of Commitment: Portuguese Team hereby commits to participate in the 2012 RoboCup competition if it is qualified for participation with a team formed with mixed members from two Portuguese Universities, namely University of Porto and University of Aveiro and three Portuguese RoboCup teams: FC Portugal, CAMBADA and 5DPO.

Team Constitution: The Portuguese Team is currently formed by the following members: João Silva (Team Leader, PhD student-CAMBADA), Luis Paulo Reis (PhD-FC Portugal), Nuno Lau (PhD-FC Portugal/CAMBADA), António Paulo Moreira (PhD-5DPO) and António J. R. Neves (PhD-CAMBADA), Nima Shafii (PhD Student-FC Portugal), Alina Trifan (PhD student-CAMBADA).

Most of the team members are PhD or PhD students with strong background on cooperative robotics and a large number of RoboCup participations in the context of FC Portugal, CAMBADA and/or 5DPO teams.

2 Software architecture

The software developed for the Portuguese Team robots uses a similar distributed architecture as the one used in the CAMBADA robots [1]. This distributed architecture is based on several processes, namely Communications, Vision and Agent, centered in a Real-time database (RtDB).

Following the CAMBADA software approach, the software used in the robot is also distributed. Therefore, three different processes are executed concurrently. All the processes run on the robot's processing unit in Linux. Furthermore, a NAO qi module is also implemented, to handle the communication between the agent and the robot DCM (Fig. 1).

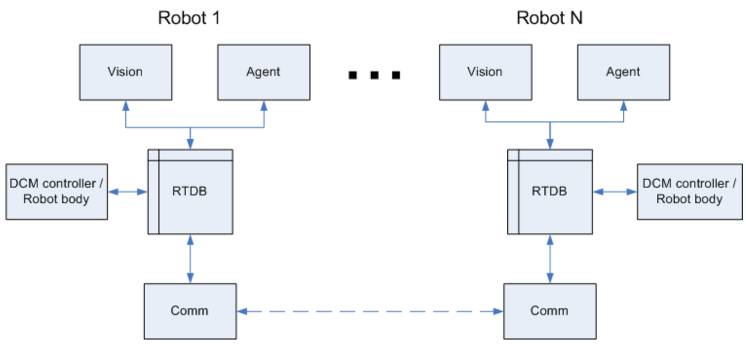


Fig. 1: Diagram of the software running on the robots.

Inter-process communication is handled by means of a RtDB, implemented as a block of shared memory. The RtDB is divided into two regions, the local and the shared one. The local region allows communication between processes running on the robot. The shared region implements a Blackboard communication paradigm and allows communication between processes running on different robots. All shared sections in the RtDB are kept updated by an adaptive broadcasting mechanism that minimizes delay and packet collisions.

The processes composing the Portuguese Team robot software are:

- **Vision**: is responsible for acquiring the visual data from the robot cameras.
- **Agent**: is the process that integrates the sensor information and constructs the robot's worldstate, taking the decisions based on this information.
- **Communications**: handles the inter-robot communications, receiving the information shared by other robots and transmitting the data from the shared section of the RtDB.

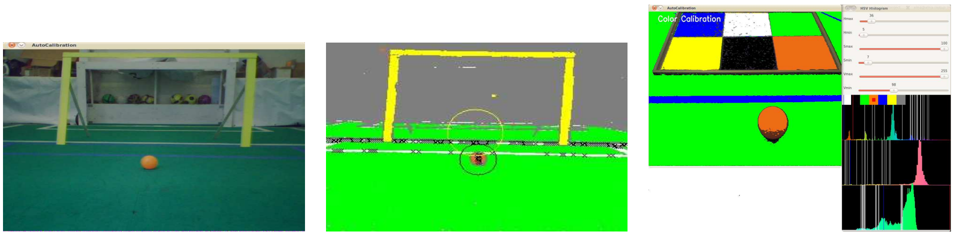
3 Vision

The architecture of the vision system can be divided into three main parts: access to the device and image acquisition, calibration of the camera parameters and object detection and classification [2]. Moreover, apart from these modules, two applications have also been developed either for calibrating the colors of interest (NaoCalib) or for debugging purposes (NaoViewer). These two applications run on an external computer and communicate with the robot through a TCP module of the client-server type that we have developed. The current version of the vision system represents the best trade-off that the team was able to accomplish between processing requirements and the hardware available in order to attain reliable results in real time.

Having the possibility of running the vision module as a server, the two applications that we have developed, NaoCalib and NaoViewer can act as clients that can receive, display and manipulate the data coming from the robot. Thus, NaoViewer is a graphical application that allows the display both of the original image as well as the corresponding index image containing the validation marks for each object of interest that was found. This application was essential in terms of understanding what the robot "sees" since NAO does not have any graphical interface that allows the display and manipulation of images. Also considering the limited resources of the robot the choice of building a graphical interface on the robot was out of the question. NaoCalib is a very helpful application that we developed for the calibration of the colors of interest (see Fig. 2).

4 Agent

The architecture of the agent is modular and similar to the one used in the simulation league. It starts by integrating the robot body information into an agent model and then uses it to build a world model. The creation of the world model is based on work developed for the CAMBADA team [3]. Afterwards, the agent will evaluate its conditions and the ones from the world (including



(a) (b) (c)

Fig. 2: In (a), an image captured by the NAO camera. in (b), the same image with the colors of interest classified. The marks over the color blobs represent the detected objects. In (c), an example of the classification of the colors of interest, by means of the NaoCalib application.

information shared by its team mates) and decide how to act, filling a shared memory with all the data necessary for that action to occur.

The main modifications from the simulated agent were in the low level communications. In simulation, the agent communicates with the server to send the actuator values and receive the sensor values. In the real robot, instead of communicating with a server, the agent sends the actuator values and reads the sensor values from the shared memory.

Behavior models The high level behaviors can be ported from the simulated robot to the real robot without changes, as long as the low level behaviors are developed for both the simulated and the real robot. When adapting the behaviors from simulation to a real robot we tried to keep the architecture of the agent similar on both [4–6].

There are currently several behavior models on the team software structure, being the main ones the Slot and the CPG behavior models [7].

Since walking is one of the most important basic behaviors, it has its own implementations. Several walking behaviors developed and optimized for the FCPortugal 3D simulation team [8] were adapted. On the last competition, the existing behavioral structure was as depicted in Fig. 3.

OmnidirectionalWalk Behavior This behavior, based on the Sven Behnke's walk, produces the desired motion calculating the trajectories of three parameters for each leg, and then uses the Leg Interface to convert these parameters into joint angles. To adapt this behavior from the simulation to the real robot we modified the Leg Interface according to the dimensions of the real robot.

TFSWalk Behavior This behavior has an interface based on speed and angle, which allow us to make the robot move in straight or curved lines, forward or backwards. The main problem in the adaptation of this behavior was the friction

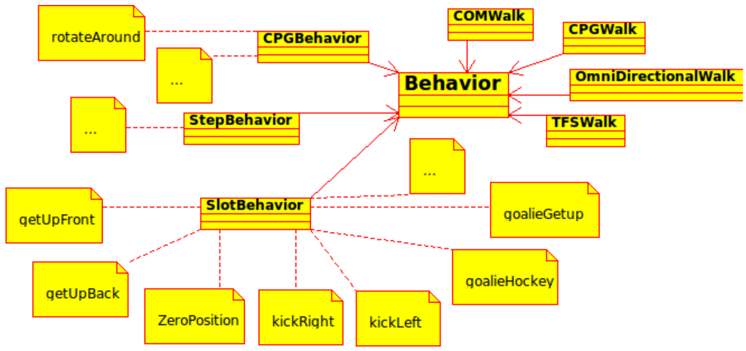


Fig. 3: Behavior structure diagram.

of the ground, since originally in the simulation, the robot kept its feet too close to the ground. The problem was tackled with two approaches, one that adds coronal movement and one that lifts the feet higher in its trajectory [9].

The TFSWalk is the main walking gait currently used, since it was verified that, with the current implementation, it is faster and more stable than the omnidirectional one.

5 Conclusions

This paper describes the current development stage of the SPL Portuguese Team for the purpose of the qualification for RoboCup 2012. Most of the work was done in the development of an efficient vision system and on the adaptation of some modules from the software used on the CAMBADA robots. Moreover, the agent from FCPortugal 3D was adapted, mainly the low level behaviors and walking.

The team wants to thanks the support of the institutions involved on the project, IEETA, Universitity of Aveiro, FEUP, INESC and University of Porto.

Main project related publications

1. Neves, A.J.R., et al.: CAMBADA soccer team: from robot architecture to multiagent coordination. In Papić, V., ed.: Robot Soccer. InTech (2010) 19–45
2. Trifan, A.L., Neve, A.J., Cunha, B., Lau, N.: A modular real-time vision system for humanoid robots. In: Proc. of IS&T/SPIE Electronic Imaging 2012. (2012 (in press))
3. Silva, J., Lau, N., Neves, A.J.R., Rodrigues, J., Azevedo, J.L.: World modeling on an MSL robotic soccer team. Mechatronics **21** (2011) 411–422
4. Domingues, E., Lau, N., Pimentel, B., Shafii, N., Reis, L.P., Neves, A.J.R.: Humanoid behaviors: From simulation to a real robot. In Antunes, L., Pinto, H.S., eds.: Progress in Artificial Intelligence. Volume 7026 of Lecture Notes in Artificial Intelligence (LNAI). Springer (2011) 352–364
5. Cruz, L., Reis, L.P., Rei, L.: Generic optimization of humanoid robots' behaviours. In Antunes, L., Pinto, H.S., Prada, R., Trigo, P., eds.: Proceedings of the 15th Portuguese Conference on Artificial Intelligence. (2011) 385–397
6. Rei, L., Reis, L.P., Lau, N.: Optimizing a humanoid robot skill. In Lima, P., Cardeira, C., eds.: Proceedings of the 11th International Conference on Mobile Robots and Competitions,. (2011) 78–83
7. Picado, H., Gestal, M., Lau, N., Reis, L., Tomé, A.: Automatic generation of biped walk behavior using genetic algorithms. In Cabestany, J., Sandoval, F., Prieto, A., Corchado, J., eds.: Bio-Inspired Systems: Computational and Ambient Intelligence. Volume 5517 of Lecture Notes in Computer Science. Springer Berlin / Heidelberg (2009) 805–812
8. Shafii, N., Passos, L.S., Reis, L.P.: Humanoid soccer robot motion planning using graphplan. In Lima, P., Cardeira, C., eds.: Proceedings of the 11th International Conference on Mobile Robots and Competitions,. (2011) 84–89
9. Shafii, N., Reis, L., Lau, N.: Biped walking using coronal and sagittal movements based on truncated fourier series. In Ruiz-del Solar, J., Chown, E., Plöger, P., eds.: RoboCup 2010: Robot Soccer World Cup XIV. Volume 6556 of Lecture Notes in Computer Science. Springer Berlin / Heidelberg (2011) 324–335