ROS for Humanoid Soccer Robots

Marc Bestmann
TAMS, Dept. Informatics
University Hamburg
Hamburg, Germany
bestmann@informatik.uni-hamburg.de

Norman Hendrich
TAMS, Dept. Informatics
University Hamburg
Hamburg, Germany
hendrich@informatik.uni-hamburg.de

Florens Wasserfall
TAMS, Dept. Informatics
University Hamburg
Hamburg, Germany
wasserfall@informatik.uni-hamburg.de

***Abstract*—The increase in software complexity and hardware costs forces robot soccer teams to collaborate. Instead of focusing on a specific architecture, we propose a set of common ROS messages as interface definitions to encourage software exchange. Furthermore, we implemented fundamental utility packages for collaborative play in mixed teams.**

***Index Terms*—ROS, humanoid, RoboCup, soccer, collaboration**

I. Introduction

Over the last years, the complexity of games and the size of participating robots increased in the RoboCup Humanoid League. This trend will continue, aiming at a human like soccer game. Therefore, it becomes increasingly necessary for participants to play together in a team with mixed robots and to exchange software, within and beyond the league. This introduces novel requirements for the software architectures of participating teams.

During the last four years, teams started to use the Robot Operating System (ROS) [1], which allows an easy exchange of software packages in principle. Still, the exchange remained difficult due to different message definitions [2] [3], architectures and a lack of knowledge about existing packages. To solve this, the Hamburg Bit-Bots, together with the WF Wolves and Rhoban FC, started an initiative to ease the use of ROS for soccer robots [5].

II. Approach

The fundamental idea is to specify a common set of ROS message definitions for the specific domain of robot soccer. Using these messages supports the exchange of ROS nodes between different architectures (see figure 1). The messages are accompanied by a set of visualization modules which display soccer specific information, e.g. an overview of the field with positions of the players. Furthermore, some standard modules for communication inside a team and with the game controller are provided, enabling collaborative play in mixed teams. To further simplify the exchange of packages, naming conventions are defined and a central source leading to all compatible packages is provided [4].

The proposed approach was tested by the Hamburg Bit-Bots and the WF Wolves in 2017. Multiple ROS packages were exchanged between both teams, even during competitions. In addition, both teams participated as a mixed team in the Iran Open with different types of robots, using the shared team communication package. Other teams are already planning to join this initiative, increasing the number of available compatible packages.

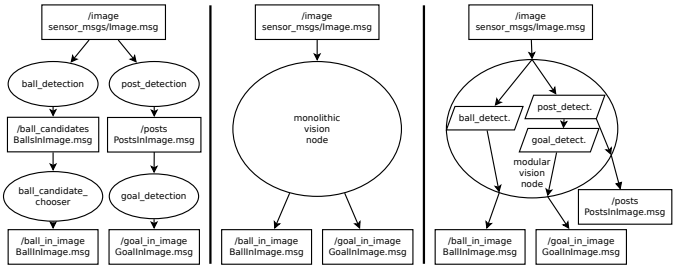

Fig. 1. The "messages first" concept applied to ball detection. In the first approach (**left**), the detection is implemented in single steps, represented by different nodes. This enables a high degree of interchangeability. In the second approach (**center**), everything is done in one node, which can not be divided, for example a neural network. In the third approach (**right**), the computation is done using a legacy system with an internal modular structure. An additional intermediate result is provided for visualization. All three presented implementations use the same set of messages and can therefore be easily exchanged and compared.

III. Conclusion and Further Work

By using the proposed "messages first" approach we ensure an increased compatibility between different teams without limitations on the architectural design. The common utility packages allow that different robots can play in a mixed team. Furthermore, new teams can utilize open source packages from different established teams to facilitate their entry into the league.

In the future, we would like to have more teams to join our initiative. This will help us to further generalize the messages and it will increase the base of usable packages. Furthermore, we plan to transfer messages which are generally applicable to humanoid robots to the ROS standards allowing their use outside of robot soccer.

References

[1] M. Quigley, et al. "ROS: an open-source Robot Operating System." ICRA workshop on open source software. Vol. 3. No. 3.2. 2009.
[2] L. Forero, J. Ynez, J. Ruiz-del-Solar "Integration of the ROS framework in soccer robotics: the NAO case." Robot Soccer World Cup. Springer, Berlin, Heidelberg, 2013.
[3] M. Schwarz, et al. "Humanoid teensize open platform nimbro-op." Robot Soccer World Cup. Springer, Berlin, Heidelberg, 2013.
[4] https://github.com/bit-bots/humanoid_league_msgs/wiki
[5] M. Bestmann "Towards Using ROS in the RoboCup Humanoid Soccer League." master's thesis, University Hamburg 2017.