Bremen Small Multi-Agent Robot Team (B-Smart) Team Description for RoboCup 2003

Andreas Hoppe, Jörg Kurlbaum, Björn Mohrmann, Martin Poloczek, Dschen Reinecke, Thomas Röfer, and Ubbo Visser

Bremer Institut für Sichere Systeme, TZI, FB 3 Mathematik/Informatik, Universität Bremen, Postfach 330440, 28334 Bremen, Germany
grp-ssl@informatik.uni-bremen.de

Abstract. The B-Smart Small Size Team is a student project at the University of Bremen. This article describes the progress of the development since the project starting point in this term. Last years hardware is used while the software was written from scratch. Due to the late start of our project, the software was developed in the last four weeks before the qualification deadline, and will be significantly enhanced in the next four months.

1 Introduction

The computer science department at the University of Bremen is involved in several projects that deal with autonomous robots (like the Bremen Autonomous Wheelchair Rolland). For the students there are projects that involve e.g. building maps for autonomous vehicles and of course the RoboCup. There are already two teams participating in the RoboCup, the Bremen Byters as a part of the

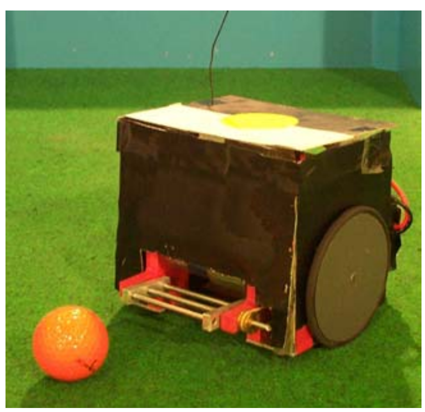

Fig. 1. A robot of B-Smart.

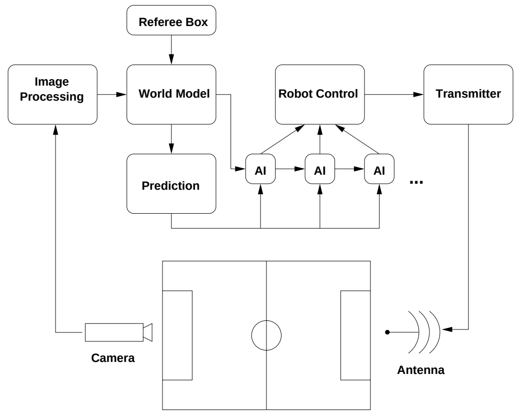

Fig. 2. Symbolic description of the architecture

GermanTeam [2] (Sony Four-Legged Robot League) and VirtualWerder [4] (Simulation League). Both Teams are already quite successful, so the goal of B-Smart is to also place a F-180 Team in the international research and competition field.

2 The Robots

The robots (cf. Fig. 1) were inherited from B-Smart 2002. They are equipped with a differential drive and a rotating kicking-mechanism, and they are completely remote controlled. The two DC motors with 14:1 gear reduction are triggered by a Fujitsu MB90F594A microcontroller that consists of a 16 MHz CPU, 256 KB flash memory, a serial RS-232 interface, and multiple analog and digital I/O-ports.

In addition to the controller, the microcontroller board contains a SE-200 transceiver module for communication working in the 433MHz band. The microcontroller board is powered by 6 NiMH rechargeable mignon batteries. The motors are powered by more powerful rechargeable batteries.

3 The Control Software

3.1 Architecture

The software is divided into small parts (cf. Fig. 2). The Vision System updates the World Model with every frame it processes. The prediction tries to determine the expected position of the robots (opponent and team members) and the ball. With this information and the information from the World Model the Robot AI tries to act reasonable for the given situation. The behavior is controlled by five reactive agents. Commands for each robot are generated and forwarded to Robot Control. The commands are interpreted and they are converted into control sequences usable for the robot. The Transmitter transports these sequences to the robots via the radio link.

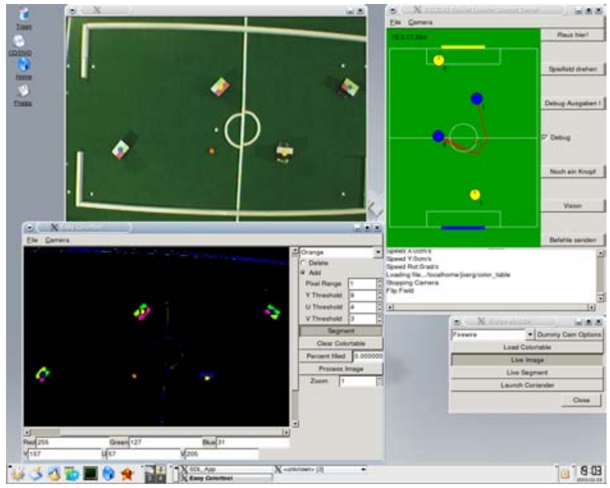

Fig. 3. A screenshot of the development tools.

3.2 Image Processing

For the image processing an overhead industrial camera on the fire-wire (IEEE 1394) bus is used. The camera provides 30 frames per second at a resolution of 640×480 pixels in YUV 4:1:1 format. For color segmentation a color table is used (i.e. a lookup table mapping from YUV space to color classes) that is created with an especially designed software tool (cf. the lower left window in Fig. 3). Most YUV colors are assigned to one of the color classes used by the processing software, e.g. the orange of the ball, yellow of the team markers, and so on. In every loop of the processing thread the system searches a grid of every fourth pixel in height and width for objects on the field. When detecting a pixel with reasonable color class, a flood fill algorithm searches for a contiguous area of that color class, and it decides whether an object was found or not. The image is only segmented in the neighborhood of the first pixel of an object. With that method it is possible to process 30 frames per second on the PC employed. The robot direction and identification is detected by a black and white pattern. Because the YUV 4:1:1 Format is improper in the color channels (U and V), the intensity (Y) is more reliable for marker detection. The markers are similar to the ones used by the 5DPO [1] F-180 team, and similar approaches are also used by other teams.

3.3 Remote Control

Commands are transmitted with a kind of Morse code on the analog radio link. It is a one-way communication at a rate of 30 to 40 commands per second and robot. Over this channel, a pulse width modulation is emulated by sending on/off instructions. Therefore one byte is sent as a command that consists of three bits for the robot id, two bits for the left motor (direction, on/off), two bits for the right motor, and one bit for the kicker (on/off).

3.4 User Interface

The complete software runs on a single PC with 2GHz CPU speed and the GNU/Linux operating system. The software is able to visualize the information provided by the world model, and it can display debugging information, e.g. planned paths (cf. the upper right window in Fig. 3). Information about the vision system is also provided (cf. the upper left window in Fig. 3). The world model is updated with every frame from image processing system. It is stack based and it is able to do predictions about the ball's and the robot's positions based on the information provided by the last records in the world model.

4 Future Work

As we just started the project our main intention is to develop a better framework to easily implement high-level control methods. B-Smart is part of a two-years student project at the University of Bremen that is also active in the Sony Four-Legged Robot League and the Simulation League. This student project is in close cooperation with a research project on plan recognition in RoboCup that is part of the priority program "Cooperating teams of mobile robots in dynamic environments" funded by the Deutsche Forschungsgemeinschaft (German Research Foundation).

The Bremen Byters in the Sony Four-Legged Robot League already use a robot simulator (SimRobot [3]) that will be integrated into the B-Smart software very soon.

In the near future, the balance of the robots must be improved, and a dribbling system as used by CMDragons and others shall be integrated.

In parallel to working with the robots using a differential drive, a new design with an omni-directional drive will be developed. In addition, the use of a local vision system is planned.

References

1. Paulo Costa *et al.*, 5dpo Team Description Paper. In: RoboCup 2002. Lecture Notes in Artificial Intelligence. Springer, to appear (already published in RoboCup 2002: Robot Soccer World Cup VI Pre-Proceedings)
2. GermanTeam, Sony Four-Legged Robot League, Humboldt-Universität zu Berlin, Universität Bremen, Technische Universität Darmstadt, Universität Dortmund, `http://www.robocup.de/germanteam`.
3. SimRobot – A 3-D Kinematic Robotics Simulator, `http://www.tzi.de/simrobot`.
4. Virtual Werder, RoboCup Simulation Team, Universität Bremen, `http://www.virtualwerder.de`.