The Attempto RoboCup Robot Team

Michael Plagge, Richard Günther, Dirk Neumann, and Andreas Zell

W.-Schickard-Institute for Computer Science, Dept. of Computer Architecture
Köstlinstr. 6, D-72074 Tübingen, Germany
{plagge,guenther,neumann,zell}@informatik.uni-tuebingen.de
http://www-ra.informatik.uni-tuebingen.de/forschung/robocup.html

Abstract. This paper describes the hardware and software architecture of the Attempto RoboCup team. The different parts of robot, sensor and computer hardware which are used are presented.

1 Introduction

For building a good team of agents that can take part in the RoboCup-2000 contest, ideas and results from different fields of research, e.g. artificial intelligence, robotics, image processing, engineering, multi agent systems can or even must be used and tested [1],[2].
Since we are developing a team for the mid-size contest our main objective is to build a robot system, which is able to recognize the environment in a suitable way and to build a fast and reliable control system, which is capable of solving the given task of playing football. This control system must cope with the dynamics and adverse aspects of RoboCup and with complex situations
The remainder of the paper is structured as follows: Section 2 describes the robot, sensor and computer hardware of the Attempto team robots. Section 3 gives an overview about the three different layers of our software architecture.

2 Hardware

2.1 Robot platform

As the basic robot platform for the field player we are using the Pioneer2 DX from ActivMedia Inc. These robots are equipped with a differential drive system with a free running caster wheel mounted at the back of the robot. The maximum achievable translation speed is about 1,5 m/s, the maximum rotational speed is $2\ \pi/s$. The robot can carry weights up to 20 kg and can be equipped with a maximum of three 7,2 Ah batteries, which allows an operating time with all additional hardware like PC and sensors of nearly three hours without recharging.
Via a serial device the robot can communicate with a remote computer. This device can operate at a maximum speed of 38400 bauds. The robot sends 20 times a second a status data packet to the remote computer. It also accepts commands

from the remote computer with the same rate. Therefore the minimal achievable response time for a closed loop controller is about 50 ms.
As the basic platform for the goalkeeper we are using a Pioneer AT. Each of the four wheels of this robot is driven by its own motor. The wheels on each side are coupled with a belt. The battery with a capacity of 12Ah allows an operating time with our additional hardware of 1.5 hours. A custom designed board with a MC68332 CPU replaces the standard MC68HC11 board and gives faster response time, higher precision of odometry and more flexible sonar firing patterns.

2.2 Sensors and actuators

As we are convinced that better sensors will result in a better situation assessment and, ultimately, in better playing capabilities, we try to employ a diversity of sensors on the robot. Therefore we are considering the use of the following sensors: Sonars, 2d laser scanner, color camera, 360^0 camera, digital compass.
Sonars: The Pioneer2 DX is equipped with eight, the Pioneer AT with seven Polaroid 6500 Ultrasonic transducers, which are mounted in front and at the front side of the robot.
Laser scanner: The employed laser scanner is a LMS200 from SICK AG. It has a 180^0 field of view and a angular resolution of $0{,}25^0$. It can measure distances up to 15 m with an accuracy of 10 mm. With a resolution of 1^0 and a total field of view of 180^0 and 500 kbps data transfer rate over a RS422 serial device the achievable scan rate is nearly 60 Hz. Its main drawbacks are its size (137*156*185 mm), weight (4,5 kg) and power comsumption (max. 17,5 W).
Color camera: For the task of object detection and classification we are using two vision systems. Both systems use a Siemens SICOLOR C810 CCD-DSP color camera, with a 1/3 inch CCD-chip and a resolution of 752x582 pixel. The output format is a regular CCIR-PAL signal with 625 rows and 50 half frames per second. One of the cameras is mounted at the front of the robot. This camera is equipped with a 2,8f wide angle lens. It is mainly for the detection of the ball and the objects, which lie in front of the moving robot.
360^0 camera: The second camera is mounted in an omnidirectional vision system, which is mounted at the top of the robot. A 4,2f lens is mounted at this camera, to achieve a large visual field. In contrast to most other omnidirectional vision systems this design has a paraboloid mirror instead of a conical mirror. This should give a better mapping of objects below the horizon ([3]).
Digital compass: This device is capable to determine the absolute orientation of the robot, where the error in measurement does not depend on the distance traveled or on other influences the odometry suffers from. It sends heading data with 5Hz, a resolution of 1^0 and an accuracy of 2^0.
Kicker: We developed two different kicking devices. A pneumatic one which consists of a pneumatic cylinder, an electric valve, and a tank for compressed air. The second kicking device is based on a spring mechanism wound up by a BMW car windshield wiper motor.

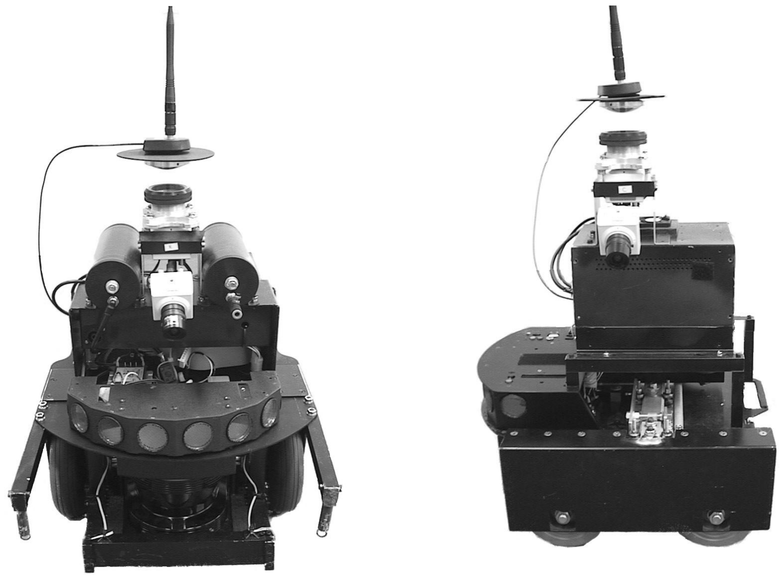

Fig. 1. The Attempto-Team robots

2.3 Onboard Computer

The onboard computer is a custom design based on standard PC parts with custom enclosure and is mounted at the rear top of the robot. Each PC has a 400 MHz AMD K6 CPU, 64 MB RAM and a 1,2 GB Hard Disk Drive. Additionally each computer is equipped with two PCI framegrabbers with a Booktree BT484 chip. These devices deliver images in YUV-format at 25 fps (PAL) and a maximum resolution of 768x576 pixels. For the connection to the laser scanner a high speed RS422 serial card was modified to achieve a data rate of 500 kbps, the highest data rate supported by the laser scanner. For the communication between the robots and to an external file server wireless PCMCIA Ethernet cards in a PCMCIA to ISA adaptor from ARtem Datentechnik, Ulm, with a data transfer rate of 2 Mb/s are used.

3 Software architecture

The software architecture of the Attempto team can be divided in three different layers: low level data processing, intermediate level layer, high level robot control. We now describe each layer in detail.

3.1 Low level data processing

In the bottom layer different server programs organize the communication with the sensor and robot hardware, and do the first steps of data processing.
The robot server receives status data from the robot, which contains position, wheel velocity, sonar data, and battery status and sends movement commands to the robot. This server also receives the orientation data from the compass. The aim in developing the robot server was to send commands as fast as possible to the robot, under the constraint that the robot is only capable to execute 20 commands per second, to achieve a minimum duration for one control loop cycle.
The laser scanner server configures the laser scanner device at startup time with a field of view of 180^0, an angular resolution of 1^0 and a distance resolution of 10mm. The laser scanner then starts to send whole 180^0 scans with a rate of 60 Hz.
The image processing grabs images from the front camera and the omnidirectional camera with a resolution of 384*288 pixels in YUV format. With this resolution it is possible to detect the ball with the front camera over a distance of 8 m and estimate the ball size and therefore the ball distance with an accuracy of 5 percent. The error in the angular position estimate is less than 1 degree.
To save processing time, the image processing does not search the whole image for the ball, but uses a history of ball positions in old images to predict the position of the ball in the next image. Only if the ball is not at the predicted position, the whole image is searched, starting the search at the predicted position.
Aside from the ball position the image processing provides an array data structure with 360 elements. Each of this elements represents a field of view of 1^0 and contains information about detected objects (ball, robots, goal, wall) and the determined distance and distance errors of these objects. In the field of view of the front camera the data structure additionally contains information about the type of the detected robots (own or opponent). Our high speed image processing needs only 3 ms per frame in the worst case (ball not at the predicted position). The average processing time for one frame is less than 1 ms. Therefore the image processing is capable of handling the 2 * 25 fps which the framegrabbers write to main memory in real time.

3.2 Intermediate layer

The intermediate layer consists of two different modules, the data fusion module, which fuses the data from the different low-level data processing servers and the motion control module, which is responsible for executing steering and control commands from the top layer.
The data fusion reduces the amount of information by extracting relevant object data from the raw sensor data. Objects fall in two different classes: dynamic objects like the ball and the other robots and static objects like the walls and the goals. The extracted information about an object includes opening angle in the field of view, distance and type of the object. The estimation error in the distance

measurement is provided [4]. Therefore for the upper layer it is not necessary to know from which sensor source a specific distance measurement comes, because the properties of the sensor device are modeled via the measurement error.

3.3 Top layer

The top layer realizes robot control consists of a planner and a world modeling component. The planner component is responsible for resolving situations by decomposing complex tasks into movement primitives which are executed by the motion control component in the intermediate layer.
The planner works on the data from the world model. It fuses the data from the internal sensors and the data coming from other teammates via the wireless Ethernet connection. It tries to keep track and identify all the objects in the environment, and tries to predict the trajectories of recently undetected objects. This component is also responsible for sending data of all objects detected by the internal sensors to all the other robots.

4 Summary and Discussion

This paper described the hardware and software architecture of our RoboCup robot team. Our approach so far has been hardware oriented: we tried to find the most capable robot platform within our budget and tried to maximize the number, diversity and the quality of our sensors. The this end we are using sonars, a wide angle color camera, an omnidirectional camera, a compass and a 2d scanning laser. Our underlying assumption is that at the current state of RoboCup play, improving the sensing capabilities will give a higher payoff than raising the speed of the robots or the onboard processing power or the "intelligence" of the robots. This is in contrast to the simulator or small size league, where all robots nearly have the same sensing capabilities. Our choice of sensors dictated the use of our pneumatic kicker and also the use of a larger PCI bus PC system with two frame grabbers. We use heavily specialized and optimized vision algorithms to keep the vision processing requirements low.

References

1. Kitano H., Asada M., Kuniyoshi Y., Noda I., Osawa E.: RoboCup: The Robot World Cup Initiative, Proc. of the first Int. Conf. on Autonomous Agents, Marina del Rey, CA, 1997, 340-347.
2. Asada M.(Ed): RoboCup-98: Robot Soccer World Cup II, Proceedings of the second RoboCup Workshop, 1998
3. Franz M.O., Schölkopf B., Mallot H.A., Bülthoff H.H., Where did I take that snapshot? Scene-based homing by image matching, In: Biol. Cybern. 79, 191-202
4. Mojaev A., Zell A.: Sonardaten-Integration für autonome mobile Roboter, in Levi P., Ahlers R.-J., May F., Schanz M. (Eds): Mustererkennung 1998, pp 556-565, Springer-Verlag 1998