Design and Evaluation of the T-Team of the University of Tuebingen for RoboCup'98

Michael Plagge, Boris Diebold, Richard Günther, Jörn Ihlenburg, Dirk Jung,
Keyan Zahedi, and Andreas Zell

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

Abstract. In this paper we present the hard- and software architecture of the robots of the T-Team Tuebingen, which participated in the RoboCup'98. This paper describes how we try to accomplish the basic skills of our robot team capable of successfully playing robot soccer by designing our hard- and software and the experiences we made with our team at the RoboCup'98 in Paris.

1 Introduction

The RoboCup contest, which first took place in 1997 in Nagoya, Japan, is an interesting environment for teams of autonomous, mobile robots [1]. To accomplish the demands of this contest each individual robot player and the team as a whole must possess a set of basic skills. These skills can be separated in three different categories: sensorial skills, control skills and cooperative skills.

Detecting the ball and the goals, detecting the other robots and determining the own position are the basic sensorial skills. The RoboCup'98 showed that there are several different ways to fulfil these tasks [3], [4], [5]. Our approach to these three tasks relies mainly on vision processing. Only for the detection of the other robots and the walls we use additionally the available sonars. The skills of the robot control unit should be obstacle avoidance, the ability to drive to a singular position with respect to the ball position and the ability to move the ball along a specific trajectory. The assignment of the tactical role of each robot, e.g. defender or attacker and the cooperation of the robots in specific situations belong to the last category. For this last skill either direct or indirect communication is necessary.

Especially direct communication simplifies the process of team-play a lot [6]. It is possible, for example, to assign the tactic role of a robot in a dynamic manner with respect to the current situation. But the problem with direct, explicit communication is the reliability of the wireless connection. In RoboCup'98 several teams had problems with this reliability.

M. Asada and H. Kitano (Eds.): RoboCup-98, LNAI 1604, pp. 464-472, 1999.

The remainder of the paper is structured as follows: section 2 gives a short introduction to the RoboCup setup. Section 3 describes the hardware architecture of the T-Team. Chapter 4 gives an overview of the software structure. The next section 5 presents the "tactical" setup of the team, before we summarise our experience of the RoboCup'98 in section 6.

2 The RoboCup Setup

The rules for RoboCup [2], which were applied at Paris, define the following properties of the field and the involved objects in the Mid-Size-League: The field is 8,22 m long and 4,58 m wide. The surrounding wall is painted white and the height is 50 cm. A flat panel is placed on every corner of the field. It is located 20 cm from the corner for each axis. Green strips of width 30 mm are painted to identify the edge of the panel. The surface of the field is also painted green. One of the goals, whose width and height are 1,50 m and 0,50 m, is painted yellow and the other is painted light blue. The ball is a usual leather football with a diameter of 20 cm and is painted orange-red. The robots should be mainly black. Between 30 cm and 60 cm there should be a colour-marking of at least 10 cm in any dimension. These markings were not used in Paris, because nearly no team did rely on them.

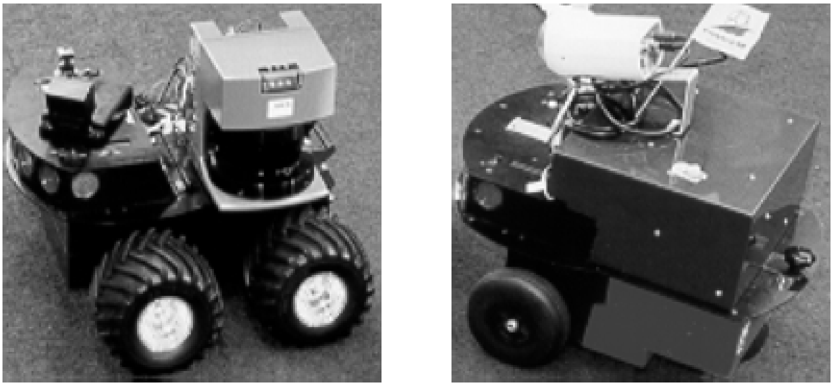
3 Hardware

To successfully operate in the RoboCup a robot has to sense players, ball, goals, and field borders, must have the ability to move around and manipulate the ball and must bridge the gap between sensing and acting. For these abilities the hardware of the robot can be separated in three different building blocks: sensors, control unit and actuators.

3.1 Actuators

Drive: The basis of our robot team are the commercially available robots Pioneer1 and Pioneer AT from Activmedia. These robots already existed at our robotics laboratory for the purpose of student education and research in the field of autonomous mobile robot systems. The four three-wheeled Pioneer1 robots are equipped with a differential drive and a free caster wheel at the back of the robot. On the Pioneer AT, which serves as goalkeeper in the team, each of the four wheels is driven by its own motor. The wheels on one side are coupled with a belt.

Kicker: As the maximum speed of the robots is about 60 cm/s, this is hardly sufficient for passing the ball or scoring a goal against a working goalkeeper without a kicking device. So we decided to develop a kicker for achieving higher ball speeds. In fact we developed two different kinds of devices. One consists of



a) b)

Fig. 1. a) goalkeeper robot with mounted SICK laser scanner. b) one of the field players. In the black box on the back of the robot is the PC. On top of the robot one can see the Sony camera.

a pneumatic cylinder, controlled via a solenoid valve. The compressed air tank with a volume of 500 cm^3 is located in the back of the robot. This volume of air is sufficient for nearly 30 kicks. The other device consists of a spring mechanism which is wound up by a strong motor. In Paris we used only the device that works with the compressed air, as the second device could not be tested well enough in time.

To prevent timing problems with releasing the kicking device and to use the kicker in an effective way, we developed a hard-wired release, which actuates the kicker in the moment the ball touches a microswitch at the front of the robot. To prevent the kicker from being actuated every time the ball touches the microswitch, the hard-wired release can be switched on and off by the robot control software.

3.2 Sensors

Cameras: Sony ECI 21 cameras, which produce a colour PAL signal, are mounted on the top of each field player. These cameras are equipped with a pan-tilt-unit and several other adjustable features like zoom and colour temperature. The view angle of the cameras is about 45^0. They are mounted on the robots with an angle of 15^0 downwards in such a way, that the upper edge of the surrounding wall still can be detected. Our on-board vision PC was designed large enough to hold a Matrox Meteor PCI bus frame grabber, which delivers 25 fps PAL images to main memory with a resolution up to 768*576. The goalkeeper uses a simple, colour based, commercial image processing system from NewtonLabs with a NTSC CCD camera. This system is capable of tracking coloured objects

with 60 Hz. It sends information, e.g. the size and centre of gravity, about these objects over a serial device.
Laser scanner: The employed laser scanner is a LMS200 from SICK AG. This device has a 180^0 field of view and a angular resolution of $0{,}5^0$. It can measure distances up to 15 m with an accuracy of 10 mm.
Ultrasonic Transducers: The Pioneer robots are equipped with seven Polaroid 6500 Ultrasonic transducers. These transducers are used for local obstacle avoidance and for self localisation of the goalkeeper.

3.3 Control Unit

Field players: A PC from standard components with Intel Pentium 166 MMX processor is mounted on each of the four Pioneer1 robots. At the time the system was designed the decision for this processor type was made because of the favourable relation between computing power and power consumption. Each PC has 64MB RAM, a 1,2 GB Hard Disk Drive and the before mentioned framegrabber. Additionally each PC has a wireless Ethernet card from ARtem Datenfunksysteme GmbH, Ulm, an ISA adaptor to a PCMCIA card, and is connected via this card and an AccessPoint with an external Computer. At Paris, we used external universal adaptors (small boxes placed on top of the robots), as we obtained the PCMCIA cards late for finishing the Linux device driver development. The connection to the microcontroller board, which controls the robot, is realized over a serial device. The standard Pioneer1 controller board is based on a Motorola MC68HC11. This microprocessor controls the motors of the robot and calculates the x,y position from the data it obtains from the wheel encoders, among other things.
Goalkeeper: The data of the onboard vision processing, based on a MC68332 CPU is evaluated by a microcontroller board based on a second Motorola MC68332 CPU, which was developed by us. This board is capable of realising a closed loop controller for the robot position in respect to the ball position, which works at 60 Hz.

4 Software Architecture

Our software architecture (Fig. 2) is build up from a set of independent modules. As the underlying operating system we use Linux, because of its simple hardware access mechanisms and the interprocess communication capabilities. The modules can be classified by the basic skills they are designed for to accomplish. The image processing, the laser data processing and the receiver part of the base server fulfil the basic sensorial skills described in section 1. The robot control and the sender part of the base server realise the control skills, and the global data processing is responsible for the team cooperation. Besides the software, which is involved in direct robot control, we developed two tools to handle the training of colours for the image processing system (Fig. 3) and to visualise the

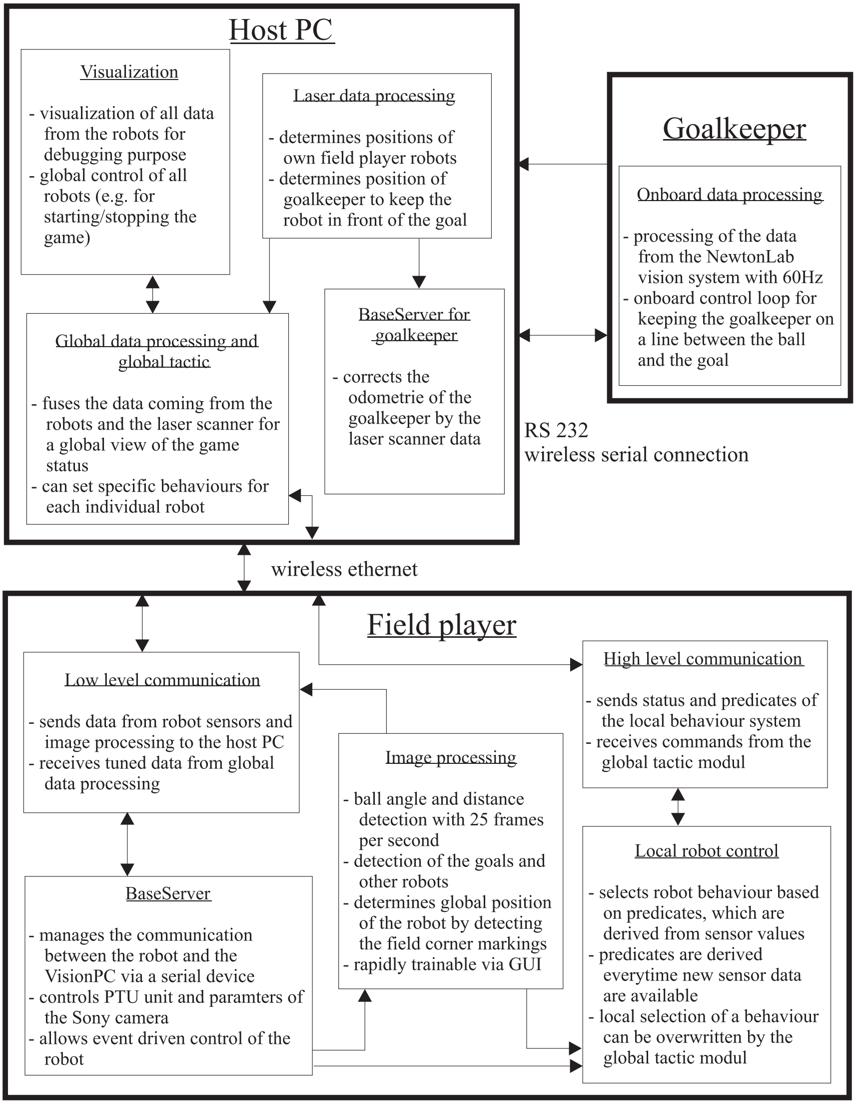


Fig. 2. Structure of the software architecture

robot sensor values and status for debugging purposes. The later tool also has the possibility to manage start and stop events during a RoboCup game in a comfortable way.

Image processing: The image processing system should handle the following three tasks in real time: Detection of the dynamic objects, detection of the statical environment (especially the goals), and estimating the position of the robot with the use of the markers in the corners of the field [7]. The image processing works with a resolution of 256*192 pixels in the YUV image format. With this resolution it is possible to detect the ball over a distance of 8 m and estimate the ball size and therefore the ball distance with an accuracy of 5 percent. The accuracy for the distance estimation of the other objects is worse because of the unknown size of these objects (only the maximum allowed size is specified and can be used for distance estimation). The error in the angular position estimate of all objects is about 1 degree. For the reason of saving time, the image processing does not search the whole image for an object, but uses a history of object positions in old images to predict the position of the object in the next new image. Only if the object is not at the predicted position, the whole image is searched for the object, starting the search at the predicted position.

The statical environment is detected by an algorithm, which predicts the lines which should be seen with respect to the current robot position. These lines are compared with the lines extracted from image processing. Even lines, which are partly hidden by an object, can be detected, if there are at least three scan points.

Experiments show, that the image processing needs 3 ms per frame in the worst case (no object 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 25 frames the framegrabber writes to main memory.

Base server: The base server is responsible for the communication with the microcontroller board on the robot. By using the signal handling scheme from Linux the module allows an asynchronous control of the clients. Data from the robots is put in a shared memory segment and can be accessed by the client via this segment.

Laser data processing: The data from the laser scanner, which is mounted on the goalkeeper, serves two different purposes. First, the global position of the goalkeeper is calculated and, if necessary, the position on the goalkeeper's microcontroller board is corrected. Second the laser data processing tries to localise all the other robots belonging to the team to provide data for them for an exact localisation.

Communication: In our architecture we use two different levels of communication, the so called "high level"- and "low level"-communication. The high level communication submits commands from the global data processing to each of the robot controls. The robot controls return their status back to the global data processing. The low level communication submits the fused data from the global data processing, for example the position of a robot evaluated by the laser data module. The robot itself sends back his sensor values and status data via the low level communication.

Robot Control: The architecture of our robot control is behaviour based [8]. But in contrast to the usual behaviour based control algorithms, which either use hard wired priorities or in which an arbiter controls the reaction of the system by merging the output of the behaviours, our system first evaluates a set of predicates (which can be observed as the output from a kind of virtual sensor) on which an appropriate behaviour is selected. That means that only one behaviour has to be computed at a time. The predicates are computed every time when there is new data from the vision system or the sensors of the robot. After a behaviour is selected it is accomplished until it gets a programmable timeout or a set of predicates for aborting the behaviour becomes true. Some of the behaviours can be selected directly by the global robot control.

Global data processing: The first task for the global data processing is to fuse the data it gets from each of the robots and from the laser data processing. It sends back this fused data to the robots. Additionally it provides the robots, which cannot see the ball, with information about the ball position obtained by robots, which see the ball. The second task is to select special behaviours on the robots for cooperation.

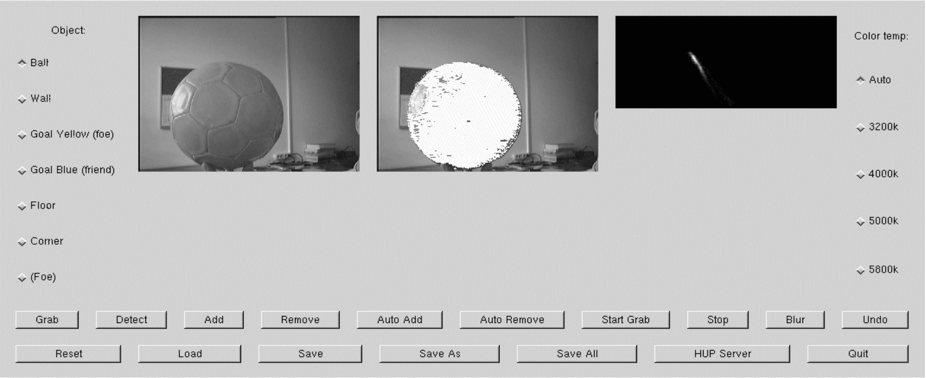


Fig. 3. GUI of our colour training centre. The leftmost picture shows the image, which was grabbed last. One can add a colour area to an object colour area by selecting the appropriate area in the grabbed picture with the mouse.

5 Team tactics

Because a team should be capable of playing soccer even in a situation where the direct communication does not work, it is safer to assign the tactical role to each team member statically, as long as there is no reliable indirect communication mechanism. So the following tactical roles are assigned to the robots:

Goalkeeper: Because of its limited field of view the CCD-Camera from the NewtonLab system is not able to sense the whole area around the goalkeeper. Therefore the camera is mounted in such a way, that the whole right part of the field with respect to the mounting point of the camera can be observed. Now the goalkeeper tries to stay on a line between the ball and the goal. Only in times, when the goalkeeper does not see the ball, it returns to its default position in front of the left side of the goal.

Defender: The two defenders are located at certain points right and left in front of the goal. To prevent the defenders from disturbing each other, each defender is responsible for its own half of the field. If the ball is farer away than 1,50 m the only task for the defenders is to track the ball. At the moment the ball comes nearer than this 1,50 m the defender should move to the ball and kick the ball into the opponent's half of the field. Afterwards it should return to its default position.

Attacker: The attacker's task is to push or kick the ball into the opponent's goal. After it detects the ball the attacker drives to the ball and turns with the ball into the direction of the opponent's goal. Because the attackers are not assigned to a specific area of the field they use a different mechanism to prevent disturbing each other. Only the attacker, which is nearest to the ball, goes for the ball. The other attacker just tracks the ball and does not move further. If the robot, which possesses the ball, detects the opponent goal, it enables the kicker release. Then the next time the ball touches the microswitch in front of the robot, it will be kicked in the direction of the opponent's goal.

6 Experience from RoboCup'98

The RoboCup'98 in Paris showed that there were a lot of different views about the way the task of playing soccer with robots could be accomplished. But it also showed that sometimes the realisation of these ideas failed due to problems on very low levels, e.g. mechanical failures. Especially problems, which arise from facts that can not easily be simulated or tested at the home laboratory were quite hard to work with. Our team in particular had problems with the reliability of the wireless communication between the host PC and the laser scanner. Therefore we could not use the laser scanner data for self localization of the robots. Additionally our cameras sensed the very dark green from the edge markings as black so that the image processing was not able to detect them. Due to these problems the robots were not able to relocalize when the odometry data deteriorated after a certain time. That led to several situations where the robots almost scored an own goal. But the rather successfull result of our team at the RoboCup showed that our architecture was robust enough to cope with problems like these. Besides to the robustness of our design there were some more points which were responsible for the successfull outcome. The first was the usage of a commercially available robot platform, which saves time to focus on more important aspects. The next was the ball handling and ball kicking mechanism we developed. Our PC based vision system with camera and framegrabber was more suitable and more flexible than commercial products, especially with respect to a faster and easier on-site colour retraining. The development of the microcontroller board for the goalkeeper allowed to realize a closed loop control of the goalkeeper's position with respect to the ball, working with 60 Hz. This was the key to success in penalty shooting (despite software problems in the preliminary round). In the future we will focus on a better cooperation between the robots. It should be self organising without the need for a centralised control. A second point of interest will be a probabilistic representation of the environment.

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. Rules http://www.er.ams.eng.osaka-u.ac.jp/robocup/middle98/
3. Gutmann J-S., Hatzack W., Herrmann I., Nebel B., Rittinger F., Topor A., Weigel T., and Welsch B. The CS Freiburg Team, Proc. of the second RoboCup Workshop, France, 1998, 451-459.
4. Kraetzschmar G. K., Enderle S., Sablatnög S., Bos T., Ritter M., Braxmayer H., Folkerts H., Mayer G., Müller M., Seidl H., Klinger M., Dettinger M., Wörz R. and Palm G. The Ulm Sparrows: Research into Sensorimotor Integration, Agency, Learning, and Multiagent Cooperation, Proc. of the second RoboCup Workshop, France, 1998, 459-464.
5. Kuth A., Bredenfeld A., Guenther H., Kobialka H.U., Klaassen B., Licht U., Paap K.L., Ploeger P.G., Streich H., Vollmer J., Wilberg J., Worst R. and Christaller T. Team Description of the GMD RoboCup-Team, Proc. of the second RoboCup Workshop, France, 1998, 439-449.
6. Yokota K., Ozaki K., Watanabe N., Matsumoto A., Koyama D., Ishikawa T., Kawabata K., Kaetsu H. and Asama H. Cooperative Team Play Based on Communication, Proc. of the second RoboCup Workshop, France, 1998, 491-496.
7. Shen W., Adibi J., Adobbati R., Lanksham S., Moradi H., Salemi B. and Tejada S. Integrated Reactive Soccer Agents, Proc. of the second RoboCup Workshop, France, 1998, 251-263.
8. Arkin R.C. Behavior-based robotics, MIT Press, 1998