ER-Force
Team Description Paper for RoboCup 2008

Peter Blank, Michael Bleier, Sebastian Drexler, Jan Kallwies, Patrick Kugler, Philipp Nordhus, Jan Tully

Robotic Activities Erlangen e.V.
Department of Computer Science 5, University of Erlangen-Nuremberg
Martensstraße 3, 91058 Erlangen, Germany
er-force@robotics-erlangen.de,
team homepage: http://www.er-force.de

Abstract. This paper presents an overview description of ER-Force, a RoboCup Small Size League team. The current hard- and software designs of the robots, the vision system and AI software are described. Furthermore upcoming changes and improvements will be outlined.

1 Introduction

ER-Force is a robot soccer team from the University of Erlangen-Nuremberg. The team was formed in September 2006. After participating successfully at the German Open 2007 we founded "Robotic Activities Erlangen e.V.", a non-profit association. This year we are developing and improving many parts of our system for even better performance to join RoboCup 2008.

2 Hardware Architecture

Our team consists of 6 robots (one spare robot) that are identical in construction. Figure 1 shows our 2007 design which lacked motor encoders for accurate wheel travel. Moreover the solenoid kicker turned out not powerful and reliable enough. This is the reason why we decided to build completely new robots for RoboCup 2008.

2.1 Mechanical Design

Just like a lot of other Small Size teams ER-Force uses omni-directional drive, with three custom-build wheels driven by Maxon A-max 22 motors with integrated gears. Each motor has a quadrature encoder for wheel travel control and speed estimation. In contrast to our 2007 approach we will use a pneumatic kicker this year. In comparison to our solenoid kicker the components are by far lighter. Our current experiments show that speeds of roughly 5 m/s are possible with a pressure of less than 20 bar. Ball handling will be achieved by a rubber

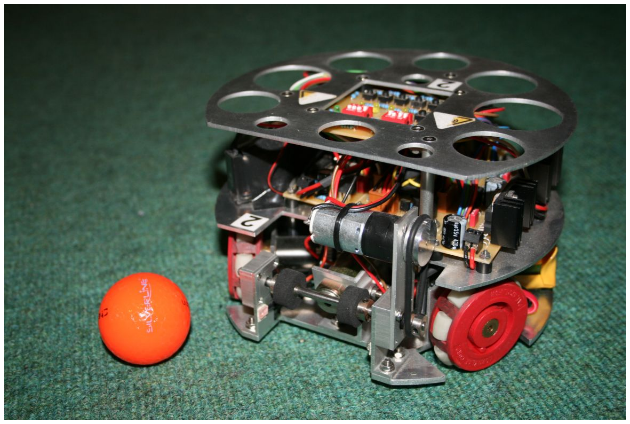

Fig. 1. ER-Force robot from 2007 without ID-plate

coated dribbling bar that exerts backspin on the ball (consistent with the Small Size League rules).

Figure 2 shows current CAD drawings of the new robot design. The drive and electronics will be single assembly units for easier repairs. The maximum diameter of our robots is 175 mm and the maximum height roughly 12 cm. For full rules compliance the dribbler will be adjusted that at all times less than 20% of the ball is covered.

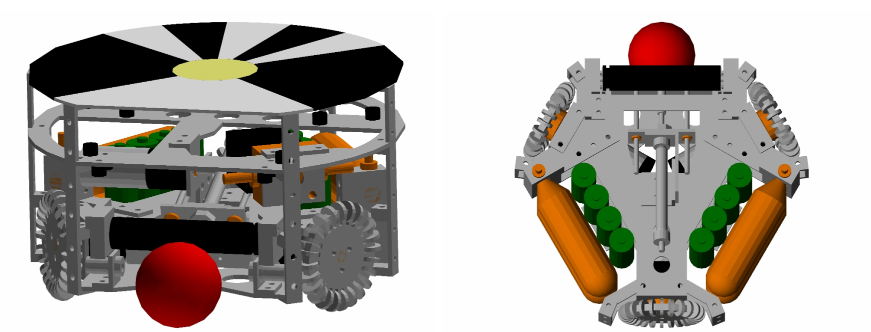

Fig. 2. CAD drawings of ER-Force's current design

2.2 Electronic Design

Our robot electronics consist of multiple microcontroller cores. One for odometer pulse generator analysis, one for drive control and PWM generation and one for communication and monitoring. The single microcontrollers are connected with each other using the SPI Bus. For radio communication we will use 433 MHz or 868 MHz transceivers. Various LEDs and a beeper operate as onboard debugging output. As power source we use a custom-made 9.6V NiMh rechargeable battery pack.

3 Vision Subsystem

The vision subsystem enables our team to localize and identify its robots as well as the enemy's robots in the soccer field. Images are captured at a frame rate of 40 Hz from two digital CCD-cameras (Allied VisionTech Guppy) mounted above the field. The images are then transferred via a FireWire IEEE 1394 interface card to the PC and processed by the vision software. The main parts of the software are calibration (field lines, camera distortion), segmentation of the image, localization of all robots and the ball, identification of the IDs and orientation of the friendly robots as well as tracking of eventually lost robots.

3.1 Camera Calibration

Field Lines - To correctly determine the position of the robots on the field, it is necessary to calibrate the cameras and correctly identify the field lines and field corners in both images. Calibration is done automatically or manually before every game and is kept constant during a game. For the automatic calibration five robots have to be placed in the corners and in the center of a half-field. By localizing all robots the calibration is performed fully automatically. If manual adjustments are necessary, the corner markers can be moved by hand.

Camera Distortion - Because of the wide-angle lenses used it is also necessary to compensate for camera distortion. This is done manually by using a test-pattern and changing the parameters. To gain performance only the pixel-coordinates are compensated for distortion, not the whole image.

Color Calibration - The colors of the field, the ball and the robot-markers have to be calibrated for every lighting condition. The calibration is done manually before every game by adjusting the thresholds for each color. To make calibration easier and less sensitive to different lighting, we use the YUV color space for the parameters and create an RGB color table after calibration is finished.

3.2 Segmentation and Localization

The segmentation is the most computationally intensive task of the vision system. To improve performance (especially on today's multi-core CPUs) all image processing tasks are done in multiple threads (typically one for each camera). The images are captured at a resolution of 780x582 in the native Bayer-Format of the cameras and are then transformed into the RGB color space by using a highly optimized transformation algorithm. After transformation every pixel is compared with a pre-calibrated color table to identify pixels with ball (orange) friend (i.e. blue) or enemy (i.e. yellow) color. The color table is generated in the color-calibration step before every game. After segmentation morphologic filters (open and close) are applied and connected regions are identified. For every connected region a global position is calculated by using the field calibration data. After this step three sets of possible friend robots, enemy robots and ball positions exist, where the most plausible ones are selected and sent to the AI module.

3.3 Identification and Orientation

So far only the positions of the objects on the field have been identified. This might be enough for the enemy robots and the ball, but the robots of our own team also need to be identified (assign a global ID to each robot) and checked for their orientation. To solve this task each of our robots carries a so-called ID-plate, where the direction and ID are color-coded with black and white circle segments. To read these plates the neighborhood of the friend positions found in the previous step is checked circular for black-white edges by using a Hough-Transformation. This transformation maps every possible edge-pixel to the so-called Hough-Space. In this space every possible edge in the image is represented by a single pixel. In this way every possible edge-pixel in the actual image is mapped onto all possible edges through this pixel, represented by a sine-wave in the Hough-Space. By searching maxima in this space the most probable edges are found and used to identify the robots and get their orientation.

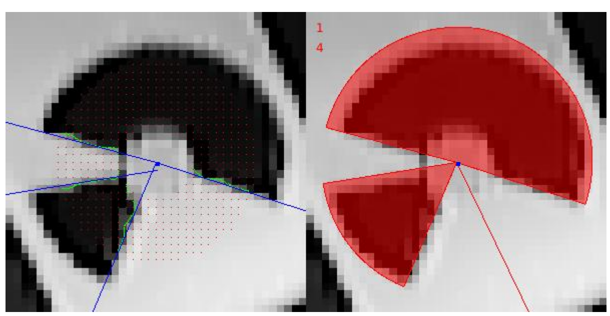

Fig. 3. Detected lines on ID-plate

3.4 Tracking

After all necessary information is extracted from the image, all positions are then used as input for the tracking module. In this module the positions in the current time step are compared to the previous positions. By doing so the velocity of the objects can be calculated and the position of possible lost robots can be predicted for a short time. A Kalman-filter for the tracker is currently under development.

4 Artificial Intelligence

The Artifical Inteligence in the RoboCup Small Size team ER-Force is located between the vision system and the motion control system. It receives the analysed vision data with UDP and sends commands to the motion control modul via UDP.

In the former system from 2007 the commands are given via some fuzzy logic. In different game situations was a different logic, and different roles. After each command a path finding algorithm prevented collisions. The current system consists of three layers. The first layer is a preprocessing of the vision data. The second layer is a decision layer. The third layer is the acting layer.

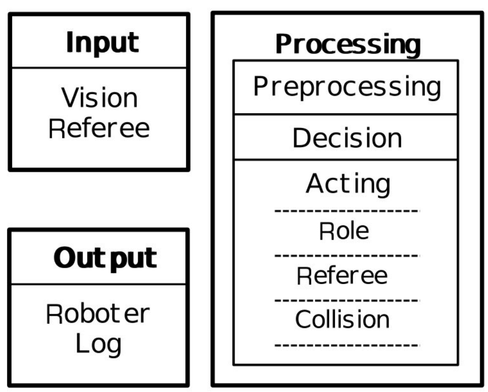

Fig. 4. Architecture of the AI module

4.1 Preprocessing

The preprocessing simplifies the data and extracts some characteristics, such as scoring chance or ball possesion chance for example. This simplifies the decision making process.

4.2 Decision

The decisons are made by taking the preprocessed input and choosing a role distribution for the robots by looking at a table. The table is trained via reinforcmente learning. This affords an evaluation of the decisions. This is done by events such as game won/lost, goal scored, or ball position changed and roles such as defender, passer, scorer, and seeker.

4.3 Acting

In the acting layer a subsumption architecture (Rodney A. Brooks) model is embedded. The layers in the subsumption architecture are parallel working with different priority to prevent conflicts.

The first layer with the lowest priority is the role layer. The role layer is only interested in playing soccer and winning the game. The second layer is the referee layer. The referee layer has a higher priority and controls the behavior of the robot so, that it follows the commands of the referee. The third layer is the collision layer. The collision layer prevents collisions by pathfinding algorithms. These algorithms are some kind of A* search.

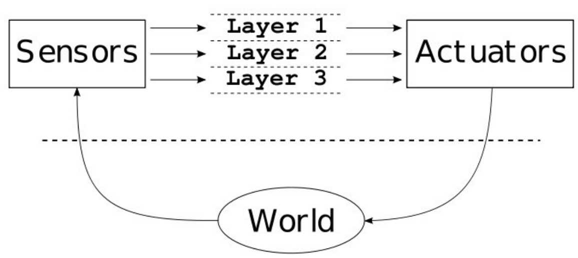

Fig. 5. Subsumption Architecture

References

[1] New Approaches to Robotics, Rodney A. Brooks, 1991, Cambridge, MA