Parsian

(Amirkabir Univ. Of Technology Robocup Small Size Team)

Team Description for Robocup 2013

Seyed Mehdi Mohaimanian Pour, Vahid Mehrabi, Erfan Sheikhi, Masoud Kazemi, Alireza Saeidi, and Ali Pahlavani

Electrical Engineering Department
Amirkabir Univ. Of Technology (Tehran Polytechnic)
424 Hafez Ave. Tehran, Iran
small-size@parsianrobotic.ir

Abstract. This is the team description paper of the Robocup Small Size Soccer Robot team "Parsian" for entering the Robocup 2013 competitions in Netherlands. In this paper we will represent our robots' current hardware design, as well as the software architecture in detail with focus on new improvements that have been made since last year. Improvements and developments like new mechanical design, improvements on planinng structure and enhancements in predefined plays, a high speed positioning evaluator will be discussed in detail.

1 Introduction

''Parsian" small size soccer robots team, founded in 2005, is organized by electrical engineering department of Amirkabir University of Technology. The purpose of this team is to design and build small size soccer robots team compatible with International Robocup competition rules as a student based project.

"Parsian" team consists of six active members from electrical, mechanical and computer science backgrounds. We have been qualified for seven consequent years for RoboCup SSL. We participated in 2008, 2009, 2010, 2011 and 2012 RoboCup competitions. Our most notable achievements was Parsian's first place in RoboCup 2012 SSL's Passing and shooting technical challenges and forth place in RoboCup 2012 SSL competition.

In this paper we first introduce our robots' hardware (section 2). Our new mechanical design will be discussed In section 2.1 and our electrical design will be covered in section 2.2. Our vision system will be discussed briefly in section 3. Section 4 explains our software framework including high level planning algorithm, low level control algorithms and our new motion planner .

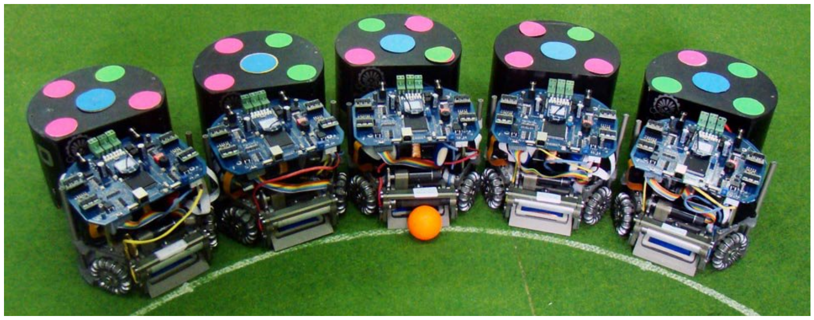

Fig. 1. Our Robots

2 The Robot's Hardware

2.1 The Robot's Mechanical Design

In this section we introduce our robot's *new* mechanical design which we have been working on since RoboCup 2010. Our current (2013) robots' mechanical design was described in detail in our 2012 extended team description paper [5].

The mechanical design of these robots was not significantly changed from the past year and Just some improvement and optimization is being applied to parts.

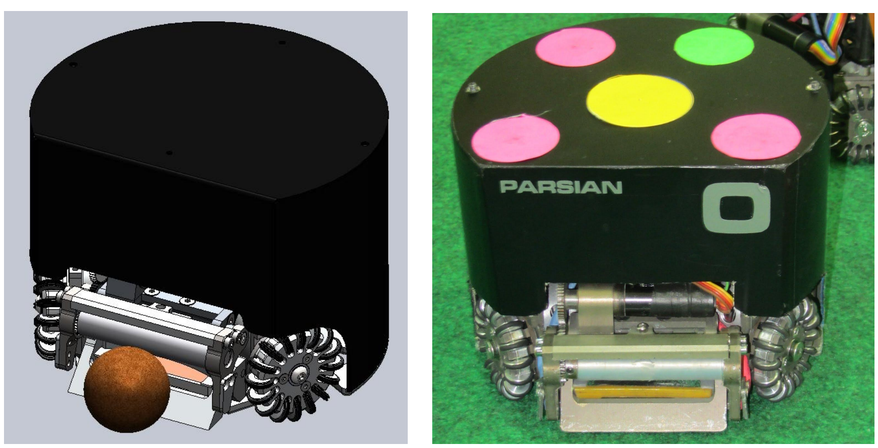

Fig. 2. Parsian's Robot's 2013 3D cad (left) and Real robot (right)

The current design's characteristics are as follows:

Robot Diameter	178 mm
Robot Height	138 mm
Ball Coverage	19 %
Max Linear Velocity	3.9 m/s
Weight	2.0 kg
Maximum kick speed	15 m/s
Limited kick speed	7.5 m/s
Maximum chip kick distance	7.0 m
Maximum ball speed catching	6.0 m/s

Main Structure and Driving System

In this section we express some information about mechanical component and devices.

The main structure consists of plates and columns that joins together with other components and fasteners such as screws. The most practical Material of the structure is 7075 Aluminum alloy and in some cases we have used steel, polyamide and etc. Our robots have 4 Omni-directional wheels and Each wheel is driven by a Maxon EC-45 30w brushless motor. So we can achieve to 3.9 m/s in max of linear velocity.

Fig. 3. Our Robot

Power transmission system of the robots is summarized to a pair of internal and external gears. By using this pair of gears the reduction ratio of gears is 3.6:1. Kicking system of robots is composed of a mechanical structure and a solenoid that each one is optimized with the simulation analysis and experimental tests. In this part, solenoid is a connection between the mechanical structure and electric charge board, So this optimization is a balancing between the hardware components.

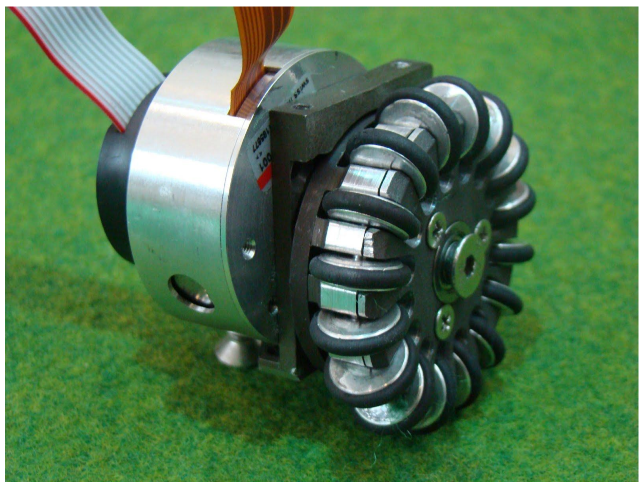

Fig. 4. Power transmission system

Direct/Chip Kick Our new robots use two solenoid systems in order to move plungers and kick the ball. For direct kicks, a cylindrical solenoid with length of 55mm is used with a 23AWG enameled wire. We optimized our direct kicking system to consume less space without losing efficiency. Kicker bar (plunger) is made of 3 parts with diameter of 13mm and total length of 130mm which are thread fastened to each other. The end part of this component is made of titanium alloy to endure high impact caused by kicker bar as shown in Figure 5(b).
The chip kick system is similar to direct kick, however its solenoid shape is flat. The size of the flat plunger in new design has been increased by 150% comparing to the old design. The mechanism which converts linear motion to angular motion is the same. Figure 5(a) shows our newly designed chip kick solenoid with the plunger.

Dribbling and Suspension System Dribbling system is also a practical ability in robot that with this the robot can absorbs and conveys the ball. According

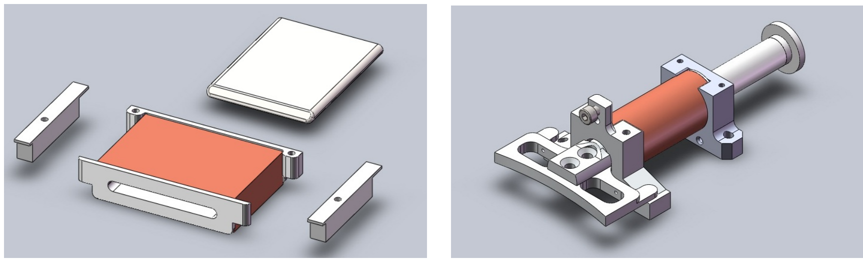

Fig. 5. (a) The new chip kick solenoid and plunger (b) The new linear kick solenoid and plunger

to the rules, geometrical design of this system is in a way that its not consist more than 20 percent of the ball. Furthermore the ball should stick entirely to the robot during the movement. The whole system is depicted in Figure 6.

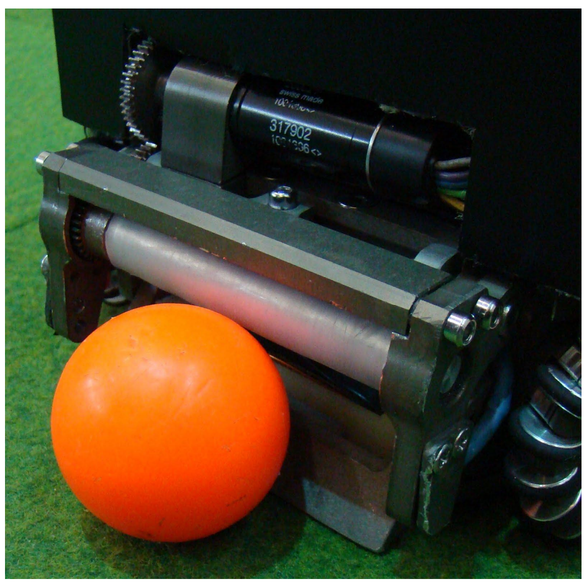

Fig. 6. The Dribbler system

Cover of the robot is also a important component that avoid damages to susceptible parts, specially electronics boards, batteries and motor components such as encoders and cables. Therefore, due to the severe impacts of the other robots and the ball, cover should be stable and resistant. In this case Fiber-cola and resin combination makes the Kevlar as the best material for achieving to these aims. So We used 3 layer of Kevlar and made up a 1.5mm thick cover.

2.2 Electrical Design

The electrical system hasn't been changed much since last year. We don't mean to make vast alterations to the current design. This year's modifications include reduction in size and weight of the electronic boards, removal of unnecessary elements and improvements in wireless communication system design. The main electronics design consists of a main board and a kicker board. Different parts of the main board are in charge of carrying out tasks such as driving BLDC1 motors, wireless communication, decoding sensors readout, execution of the control loop and sending control commands to the kicker board. The kicker board is designed to recharge the capacitors in the shortest time possible and release the energy stored into the solenoid over a controlled discharge time. Another feature that we have added to our hardware is gyro sensor. The robot uses the gyro data to move more efficient. A block diagram regarding the electronics systems design and behavior is shown in 7.

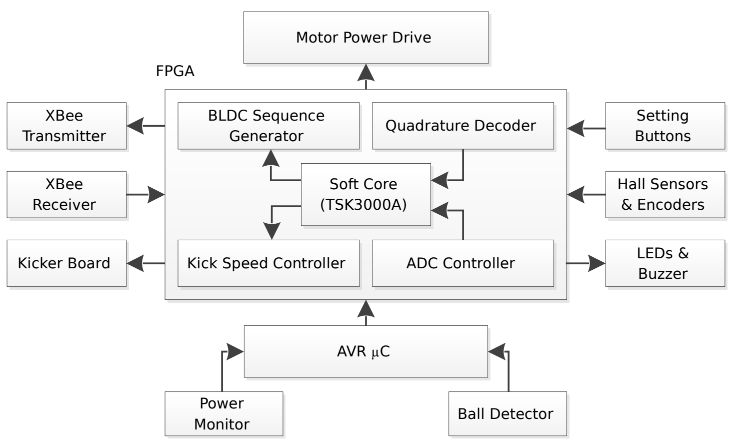

Fig. 7. The electrical system's diagram

Wireless Communication Wireless Communication Last years, there was one 2.4 GHz XBee PRO wireless module for receiving control commands from the remote host PC and sending monitoring data on each robot. The XBee is half-duplex so it cannot both send and receive data simultaneously. This results in loss of data. This year we use two XBee modules on each robot, one dedicated to receiving desired robot velocities, kick/chip desired speed and permissions and another XBee module to send battery levels, ball detection status and other monitoring data.

Main Processor A TSK3000A based soft processor, implemented on a Xilinx Spartan XC3S400 FPGA, operates as the main processor. This embedded processor receives control commands from wireless module and executes these commands using various components implemented inside the FPGA through a custom firmware developed in C language.

Kicker Board To decrease the size and weight of the kicker board, this year we have redesigned the kicker board by means of using new electrical components. The kicker board continuously charges two 2200 F 100V capacitors connected in parallels. The current design is based on DC to DC boost convertor circuit which utilizes a power MOSFET to discharge the stored energy into two solenoids. To increase the resolution of kick/chip speed we have designed a VHDL block which moderates the kick speed. With this new feature, kick/chip speed would be continues and this can be regulated with a high accuracy.

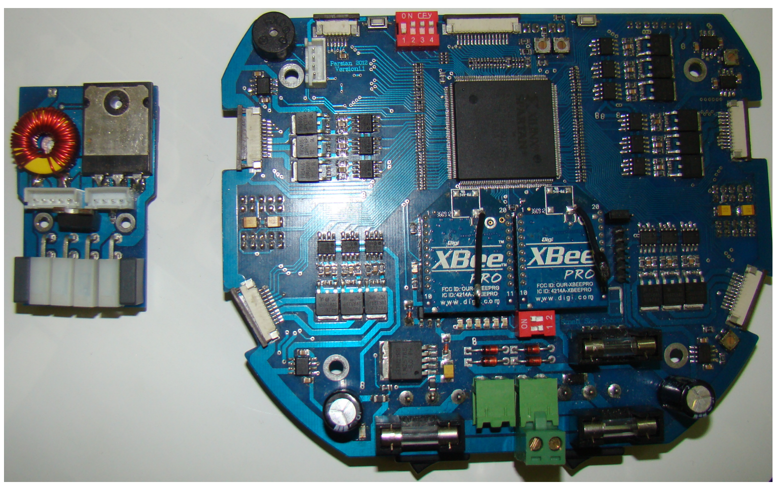

Fig. 8. The main and kicker board

3 Vision System

3.1 Filtering and Tracking

Processing of each camera's output is independent within the SSL shared vision system. The resulting package includes data of all detectable objects for each camera. In this manner there can be any number of different objects. i.e. the package may contain numerous ball positions inside it.

In order to have a unified view of the whole soccer field and to avoid mis recognition of noisy objects (e.g. the hands of referee which may be detected as ball) , the output of SSL-Vision's data should be merged and filtered.

There's also another problem that causes many problems specially in corners of field , since the fields lightening is not homogeneous all over the field (caused by shadows of referee and people standing around the field) , the SSL-Vision calibration wont detect all robots in every position so that robots will vanish occasionally during game . to remember the last place of robot and predict the possible position of robot we track every object in the field and do the prediction in every frame and use the SSL-Vision data packets to correct our estimated position .

For this purpose we divided the field to two *Half worlds* , each half world will update by receiving noisy data from it's own camera , after filtering these data and tracking objects in half world we will create our *World model* by merging two half worlds data.

In order to merge half worlds and track objects within them, the two half worlds data in each time step are passed through a *Bayesian filter*. The filtering is based on the euclidian distance between various objects within two consequent time steps. This process is done in a separate thread so the planning system can access the most reliable data at anytime. There is another level of filtering in this thread, which uses a Kalman filter in order to estimate velocities , accelerations and compensate the loop delay.

4 Planner

An overview of our planner system is demonstrated in figure 9. The data flow starts from vision part, in which SSL-Vision packets are received and processed. After this process the world model and its history are updated and the decision making loop is executed. The result of total processing cycle is the generated velocity commands for robots, which are sent to radio transmission module.

The planner framework is written in C++ using Qt Framework[6] under Ubuntu Linux OS.

4.1 High Level Planner

The Coach layer is the first step in the high level planning (decision making) loop. Choosing a formation for the team is done prior to any other decisions. According to policies, that are a mixture of manual configurations (and game-state dependant updated values, each cycle the coach layer decides the team's formation. Therefore, each agent takes part in one of the main plans: defense, midfield and offense. Defense plan consists of agents which are near the friendly penalty area, including goalie and some blocker agents. Middle plan agents intend to possess the ball owned by opponent and diminish their attacking opportunities with marking, blocking, ball interception and etc. Offense plan includes agents that are going to create attacking chances to score. One agent always takes the

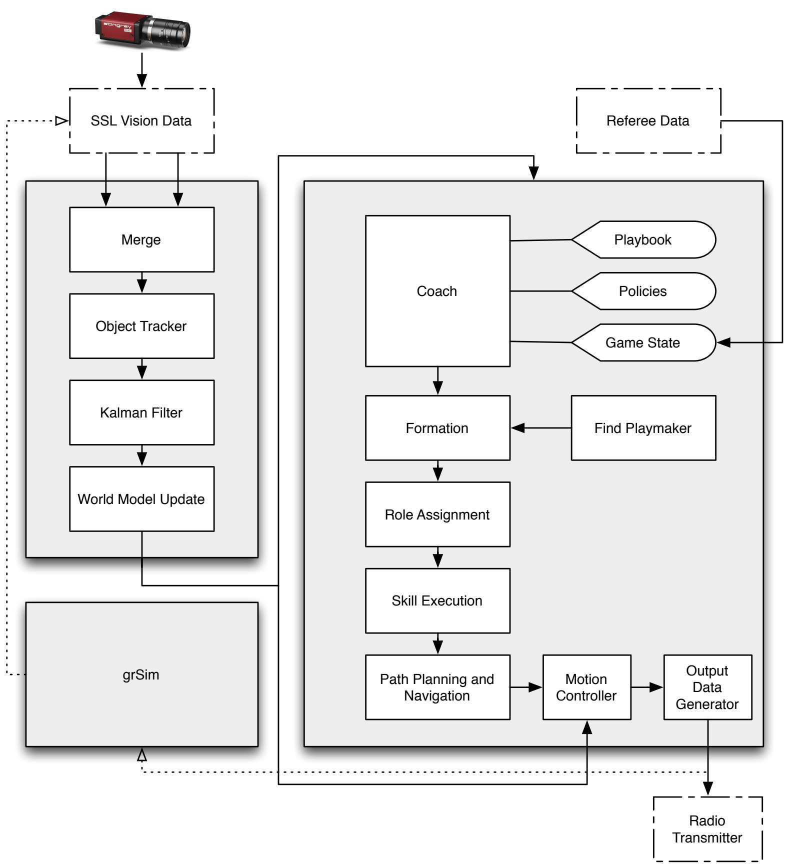

Fig. 9. Parsian's Software Architecture

role of the "playmaker" (the agent that possesses the ball), other offense agents should take suitable positions. After running the plans, a set of roles are assigned to agents in an optimized way, so that minimum movement is needed for agents to execute their roles. To perform a role, each agent may use a different set of basic skills. For example "marker" itself is a role but it uses the "gotopoint" skill to reach its target. The hierarchy of the coach structure is shown in figure 10.

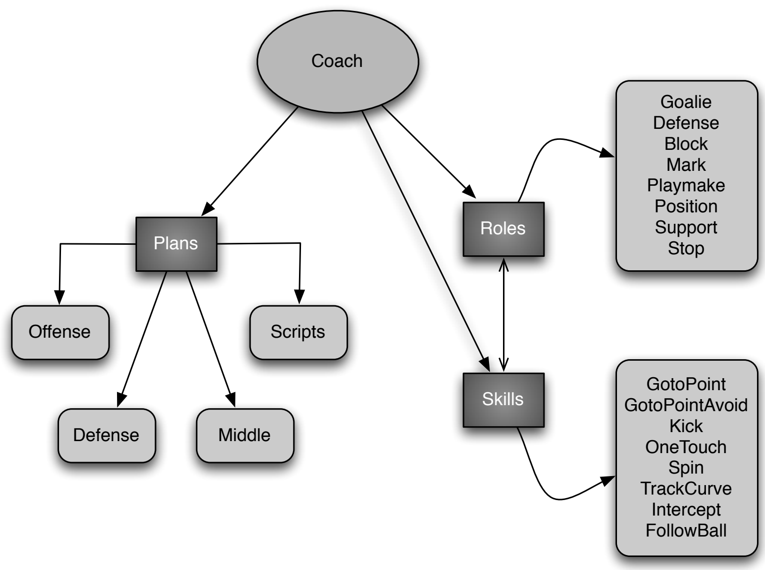

Fig. 10. The hierarchy of coach stucture

As a matter of fact, in a small-size game, most of the time the game is in stop mode (i.e. ball is moved out and the game should be started either by a direct or an indirect kick). Thus, having a knowledgeable game play when the game starts (direct or indirect kicks) may result in more scores. Kickoff, indirect kick, direct kick and penalty kick are the main "non-play-on" plays in a small-size robotic game. To have more diverse "non-play-on" game plans, we have implemented a script language. There is a simple kickoff plan written in our game script in figure 11.

Motion Planning In previous years we used ERRT algorithm for finding a path through obstacles in field , then trying to track that path and go to selected point using *Bang Bang Trajectory Generator* algorithm . Since the generated path by ERRT algorithm is not an straight path and Bang Bang trajectory planner is 1D trajectory planner for time optimization , we are currently working on a new motion planner that generates a safe path and motion profile on generated path

```
fav=1
$ourkickoff
>always
<start
<ballmoved
{
	>count(this,3)
	~kickoff3
	playmake(stop,wing)
	position(ourmidfieldtopwing,@onetouch)
	position(ourmidfieldbottomwing,@onetouch)
}
{
	>count(this,2)
	~kickoff2
	playmake(kickoff,stop)
	position(ourmidfieldtopwing,@onetouch)
}
{
	>count(this,1)
	~kickoff1
	playmake(stop,kickoff)
}
```

Fig. 11. A sample of OurKickOff script

considering robots dynamics and abilities . we hope that this motion planner will be finished soon and can briefly be described in our Extended Team Description Paper .

4.2 Future Plans

The list of our current research is given bellow. The main attitude of the mentioned researches is concentrated on improving the artificial intelligent methods utilized in the software architecture.

1. Designing and implementing a new path and trajectory planner.
2. Reimplementing new Kalman Filter for reducing the vision system delay and getting more accurate objects speed and acceleration .
3. Identification of robot's dynamics model to improve the navigation technique and path planning algorithms.

References

1. OpenGL - the industry standard for high performance graphics (2011), `http://www.opengl.org/`, [accessed February, 2011]
2. Browning, B., Bruce, J., Bowling, M., Veloso, M.: STP: Skills, tactics, and plays for multi-robot control in adversarial environments. Proceedings of the Institution of Mechanical Engineers, Part I: Journal of Systems and Control Engineering 219(1), 33–52 (2005)

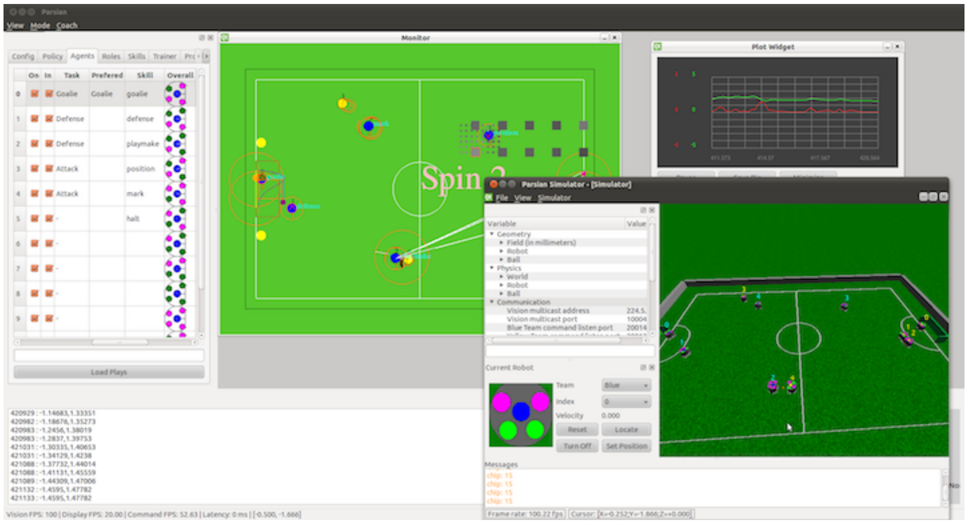

Fig. 12. The Software Environment

3. Bruce, J., Veloso, M.: Real-time randomized path planning for robot navigation. Lecture Notes in Computer Science pp. 288–295 (2003)
4. Bruce, J., Veloso, M.: Safe multirobot navigation within dynamics constraints. Proceedings-IEEE 94(7), 1398 (2006)
5. Mehrabi, V., Koochakzadeh, A., Poorjandaghi, S.S., MohaimanianPour, S.M., Sheikhi, E., Saeidi, A., Kaviani, P., Saharkhiz, S., Pahlavani, A.: Parsian - extended team description for robocup 2012. RoboCup 2012
6. Nokia Inc.: Qt - A cross-platform application and UI framework (2011), `http://qt.nokia.com/`, [accessed February, 2011]
7. Smith, R.: ODE - Open Dynamics Engine (2011), `http://www.ode.org/`, [accessed February, 2011]
8. Zickler, S., Laue, T., Birbach, O., Wongphati, M., Veloso, M.: SSL-vision: The shared vision system for the RoboCup Small Size League. RoboCup 2009: Robot Soccer World Cup XIII pp. 425–436 (2010)