# MRL Middle Size Team: RoboCup 2014 Team Description Paper

M.Gholipour [1], H.Rasam Fard [2]
M.Montazeri, F.Fathalibeyglou, M.Rasam Fard, E.Saeedi Kamal, B.Tajshafaghi, M.Farsi, Sh.Kazemi, M.Hoshyari, S.Ziyadloo, Sh.Arash

Mechatronics Research Laboratory, Islamic Azad University of Qazvin - IRAN

[1]Gholipur@gmail.com [2] Hamed.Rasam@gmail.com

**Abstract**. This paper concisely describes the very main and new features of our soccer playing robots along with the improvements made since the previous years. Our major concerns for this year's competitions have been developing a new feed forward control system, designing a new Omni vision structure, designing a new ball handling system, also developing a new feature in AI software which is an accurate role assigner.

## 1 Introduction

The MRL middle size team has started its work at Mechatronic Research Laboratory of Azad University of Qazvin since Aug 2003. This team aims at establishing an intelligent control method for autonomous multi-robot systems in dynamic uncertain environment. MRL has begun the research and work in MSL since 2004. Our first official participation was during RoboCup 2005 competitions in Osaka and then Robocup 2006 in Bremen. We optimized hardware, control and software system for Robocup 2008 and designed robust system for Robocup 2009 in Graz, as a result we find ourselves between four top teams. In Robocup Singapore 2010 competitions we got the first place of technical challenge and again forth place of league competitions. Also, we got second place of free challenge and fifth place of league in Robocup Turkey 2011 competitions. After a year of hard work, we managed to get the second place of league in Robocup Mexico City 2012 competitions. We believe that, the Intelligent, cooperative and adaptive behavior of the robots is very important factor for a team success. With this regard our research is continuously focused on: reliability, sensor fusion, dealing with uncertainty of environment for the robots, world modeling and dealing with missing information. In the following sections we briefly explain current status and new achievements of our team.

## 2 Hardware and mechanical features

We designed a 4-wheel omnidirectional robot which is equipped with MAXON EC 200 watt brushless motors which provides more speed and easy to control [1]. Main processor is Lenovo X200 notebook PC and electronic equipment are developed with ARM Cortex M3® -LPC1768 microcontrollers with high speed CAN-bus. This year we designed a new ball handling system and new Omni vision structure which is explained in the following sections.

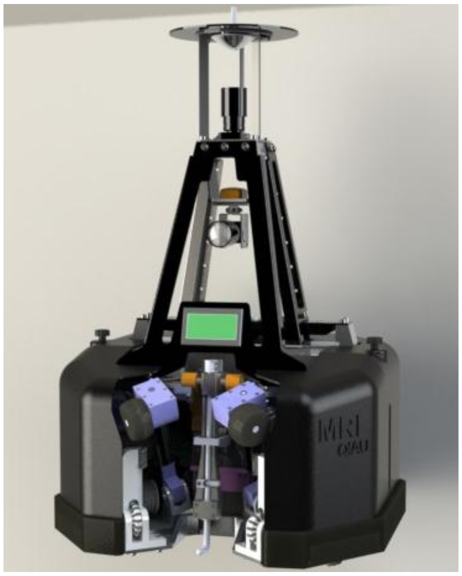

**Fig. 1.** MRL 4-Wheels robot

**Table 1.** Hardware specification of the robot

| Items | Description |
|---|---|
| Platform | 4 wheel Omnidirectional |
| Max speed | 3.5 m/s |
| Max acceleration | 4 m/s$^2$ |
| Kicker | Electromagnetic |
| Weight | 40 Kg |
| Laptop | Lenovo X200 |
| Camera | uEye UI-2210-C |
| Image processing | Omni directional mirror |
| Other sensor | IMU and IR |
| Controller | Neural Network PID |
| Spin back | Active<br>60watt 24V DC motor |

### 2.1 New Omni vision structure

The parabolic mirror is placed up and normal to the camera lens which provides 360-Degree view. The old three blades mirror holder which is shown in figure 2 didn't work out so well for full 360-Degree view. Thus, it has been replaced by a Poly carbon mirror holder as shown in figure 3. The new holder lets robot to see environment clearly.

**Fig. 2.** Three blade mirror holder

**Fig. 3.** Poly carbon mirror holder

### 2.2 New ball handling structure

This year, a new ball handling system has been developed as shown in figure 4 that enables the robot to control and hold the ball while it's driving in any direction even it's turning around. Also it would be able to catch the ball with higher factor of safety while passing or ball abduction by raising the height of wheels relative to the center of ball that allows us to use less energy and gain more speed or momentum.

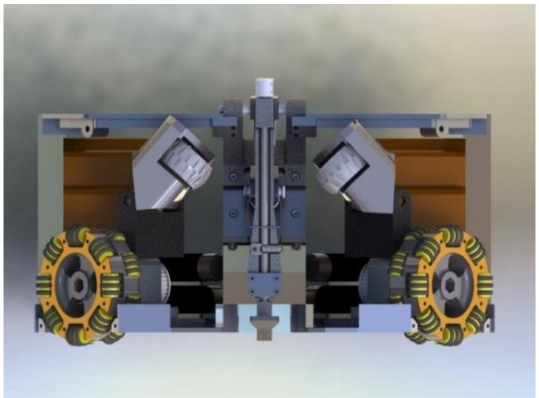

**Fig. 4.** The new ball handling mechanism

### 2.3 Electrical Design

The new electronic system of MRL team has been designed and developed continuously for more than 2 years. Our system consists of power, Kicker, monitoring, FDS, E2C and DMS boards; this approach simplifies the repairing process especially during the matches. Figure 5 shows the diagram of our Electric part.

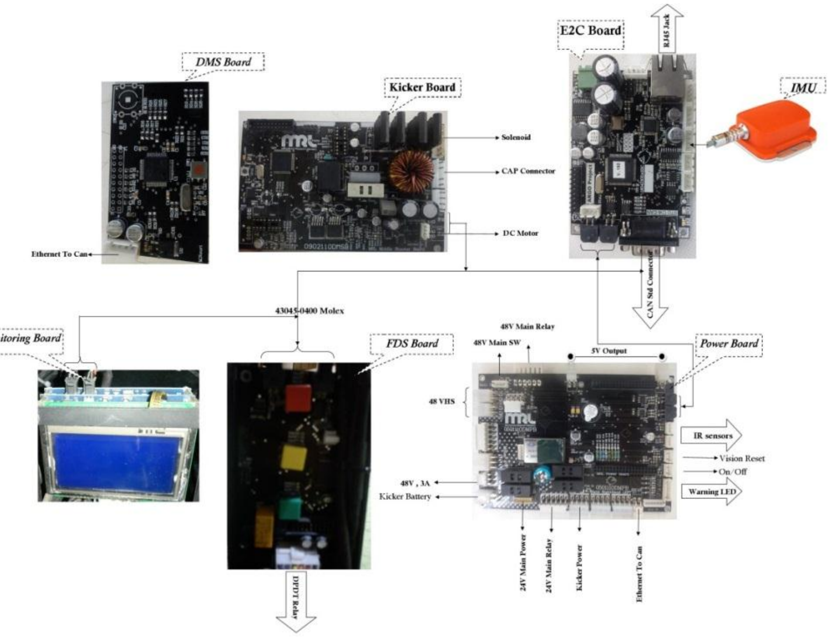

**Fig. 5.** Electronic Diagram

The major changes during the last year are Ethernet to can module (E2C), Fault Detection System (FDS), and Data rate Monitoring System (DMS). Since, the data transfer lost by disconnection of USB connector, we decided to change our protocol to Ethernet, where it is safer and more stable connection in compare with USB. Also by using E2C the usage of driver in high level has been eliminated and it causes standard Ethernet protocol which made it easier to use in AI system. Moreover we decide to use Ethernet protocol for all subsystem in early future where it causes more data transferring speed. In the other hand many of the problems came out from the fault and lost that happens in data transfer which is effective point in turning the system on and off. This fault has a destructive effect on the Epose where it causes the robot lose its control and move in unpredictable harmful way that makes hardware damage in both electrical and mechanical systems. For solving this problem we made change in our firmware and have decided to improve our communication by using a debugging and developing system. One of the other major changes that we used in our system is fault detection system and automatic supervision In order to detect and fix the errors. In this work, we present a new automatic supervision and fault detection product for our robot, based on the data analysis. This automatic supervision system includes data acquisition techniques to calculate main system's conditions and monitoring the data in real conditions of work. Also we changed the IMU connection from USB to the serial where it process completely in low level instead of direct connect to the AI system. Because the USB port was not stable enough we had many problems that came from the IMU connection. To avoid these problems this year we decided to change our system by using serial communication for IMU.

The same as previous years, we decided to utilize ARM7 microcontroller. It was selected for several reasons such as its powerful debugging capabilities and low-power design of ARM architecture. In addition, the ARM7 with TDMI-S core is one of the best choices for system control. Hence, only real-time tasks such as motor driving are executed in high level and all remained parts are implemented in ARM7 microcontroller. We used ARM7-TDMI core and developed the project in KEIL software.
In the following sections we briefly explain current electronic accessories.

### 2.3.1 Power Board

This module has a responsibility to distribute the power for all subsystems and turning the systems on and off. Also many parts same as IR sensors spin-back's motors and monitoring LEDs are controlled in this module.

### 2.3.2 E2C Module

This module changes all Can data format to Ethernet and vice versa. This is the only part that has a direct access to the high level (AI) system where all sections have a communication with this module by Can protocol and E2C module communicates with the AI by Ethernet protocol.

### 2.3.3 Battery

Power supply in our robot is departed in 2 sections, boards and motors power supply. Board power supply contains 6-cells and motors power supply contains 12-cells where robots can run for half game with these batteries. These batteries are lithium polymer (LiPo) battery, with capacity of 5000 mAH. When voltage of batteries reduces, the robot switches to sleep mode and stops working. Of course it was monitored by LEDs before reaching to low battery state and by a buzzer to show the critical state.

### 2.3.4 Kicker Board

The kicker board is designed to control the high voltage. It has one MOSFET for charging and four for kicking. An LPC2368 micro controller is used as controller. It creates pulse, limits the charger and communicates with the processor. The board also contains MOSFET driver to turn on and turn off the MOSFET in nanoseconds which prevents damaging them. The control board of the kicker circuit is a state machine with two states. The process begins by polling Up the Kick-Flag signal by main processor (Laptop) at state one. A signal that called Kick-flag is entered to the component to set the desired kick duration. When a high logic value is read by microcontroller at state two, the kicking sequence is initiated. In this state the microcontroller holds the kick signal high for the specified period of time. Keeping up signal (Kick-command), with high level logic at the different times can be creating different Kick powers. This process takes about 18 seconds. For improving the performance of this operation, we need to reduce this time to about 5 ~ 10 seconds.

### 2.3.5 Motor & Driver

There are four BLDC (Brushless DC) motors for each robot which are MAXON EC 200watt brushless motors and two DC 60watt motors for the dribbler. All motors are controlled by Epos driver that have an ability to manage low level control where directly receives speed from high level system.

### 2.3.6 Ball detection sensor

For recognizing the ball position in dribbler and distance of the ball from the dribbler, two IR transmitters and receiver sensors were used. This module is useful when robot tries to get accurately behind the ball.

### 2.3.7 FDS

We have a fault finder and monitoring system that give the capability of observing the required data from different parts of the robot without any connecting to the robot, also continuously reporting the available information on the CAN network and save them on SMD card. This module would be developed for new extensions like real-time system check or some manual commands. Also in this case we monitor many aspect of robot operation by the RGB LEDs that illustrate kicker motor's state, handling state and any other section which is needed to monitor during the match.

### 2.3.8 DMS

This new hardware has ability to illustrate the data rate in online format. This system shown the transfer as array LED where 50% means that system work properly, and if the data rate shown more that 60% continuously, it means that, fault happened in system that caused traffic in data transfer and if it was less than 40% continuously it means that, lost in transition happened.

### 2.3.9 Monitoring

For changing the states of robot we consider touch lcd with ability to monitor and change in system. In this case we could informed about all subsystem's state and debug them online in a very low period of time.

### 2.3.10 Current Sensor

For power management system, detecting over current, avoiding motor damage, battery current limitation, control usages in plunger and safety of kicker board, these sensors were utilized. Sometimes damaged happened in devices caused over current that destroy Electrical devices so we decided to use ACS712 as a current control to prevent these damages.

## 3 Software (AI and High level control):

### 3.1 High Level Control

The high level control system receives predicted data from vision software and destinations from AI planner module and make robot go to its destinations. So, the important part of control system is a path planning algorithm and we used the VORONOI as path planning algorithm [2].

The control system will find the path (VORONOI) by using start position, end position, initial velocity along x and y axis and direction of starting point and ending point. Using the result path usage time will be calculated in order to generate the velocity command which will be sent to the robot. The velocity command is generated separately for each point along the trajectory according to the frame rate. Each frame has its own velocity command. The velocity profile is generated by using Bang-Bang algorithm and makes a robot trapezoidal velocity as shown in figure 6.

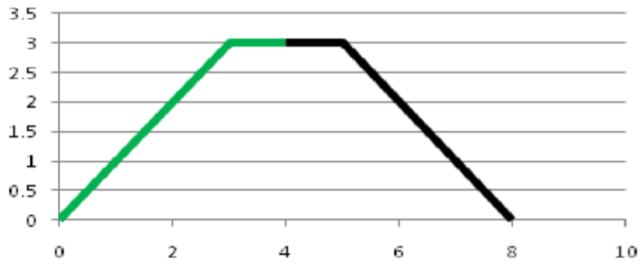

**Fig. 6.** Trapezoidal velocity profile

The robot can move with a faster velocity when it moves forward or backward and slower velocity when it moves to left or right direction. The binary search is used to find the different of velocity to time. The proper velocity is separated in x and y direction. After calculation along x and y direction, the maximum total time usage is selected in order to guarantee the robot motion. For example, if the total time usage along x- axis is greater than along y-axis, the trajectory time constrain is the total time along x- axis. The velocity that is generated for a robot in x direction will use a proper velocity along x-axis (maximum) and the velocity in y direction will use smaller value than the value that is calculated from previous step in order to make the robot moves smoothly. Also angular velocity time must be equal or lower than maximum time that calculated in the last step. This algorithm is calculated online at every frame and does not have any feedback from robot velocity because it's a feed forward algorithm.

### 3.2 Software architecture

The software architecture for our decision making system consists of three parts: Plays, Roles and Skills. Skills are any single tasks that robots can do. For example "go to point" is a skill. Each player in a real soccer game has a role like defender, forward, goalie and etc. Compared with real soccer game each robot can change its role at the right time. Switching time is an important approach to manage robots, for this reason we need an accurate role assigner. Hence, we have developed a new role assigner module that assigns role to the robot according to game state and cost of roles for each robot.

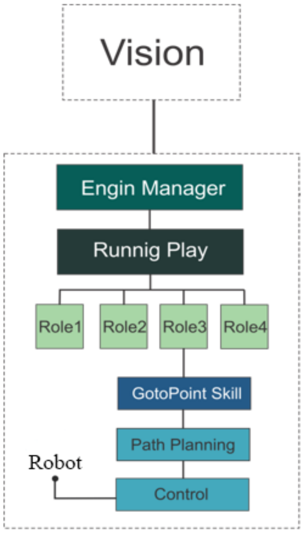

**Fig. 7.** Software architecture

## 4 Vision system and localization

Our vision system hardware is composed of a UEye camera that stands upwards with a hyperboloid mirror above it. This component provides an omni-directional vision. The output of this system is very reliable and accurate.

To process the gathered images, at first a median filter is applied in order to reduce image noises, and then the four standard color marks will be assigned to each pixel by the Color Lookup Table. The Color Lookup Table (CLT) is filled in another program, which classifies the HSL Color Space into four standard colors. This program takes some supervised samples from user to learn how to recognize the standard colors. In run time this CLT is used in an image processing algorithm to detect the ball, field, and obstacle areas in the image in real-time (50 frame/s) on the laptop computer.

Self localization is obtained through matching white lines in the camera pictures with the actual model. To recognize the lines in the pictures, we scan the radius of the picture from centre shown as figure 8. Then categorize the white and green connected spots, we register the center of white spot groups, which next and previous lines are green, as a part of the line and keep on this process for all radiuses, so a group of spots from the field lines are recognized shown as figure 9. At last the spots are converted from polar to Cartesian coordinate and from pixel mode to metric mode by using mirror equations shown as figure 10. Then, the position is again calculated by matching the spots with actual model of field lines shown as figure 11.

To recognize ball, first, the ball colors are segmented, circular shape segment is recognizes as the ball with designed algorithm. But now we are able to recognize any standard FIFA ball. We assigned a coefficient of error parameter to each recognized circle according to how much it is like to circle, and the circle with minimum coefficient of error is chosen. Also, we assume each black segment in the green area as an obstacle. All the above processes are done at once through entire $360^\circ$ scan of the Omni picture.

This year we use stereo vision system by another camera in front of the goalie to calculate the height of the kicked ball and precision enhancement of recognizing the ball far away from it. Also we are going to improve the ball detection algorithm when a small part of the ball appears in the images; this is implemented by circle fitting algorithms [3].

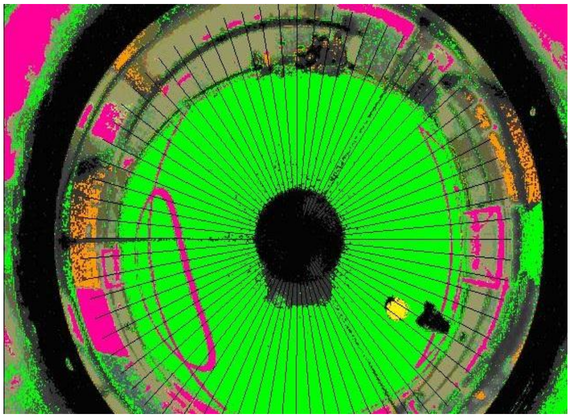

**Fig. 8.** Scan the radiuses from center

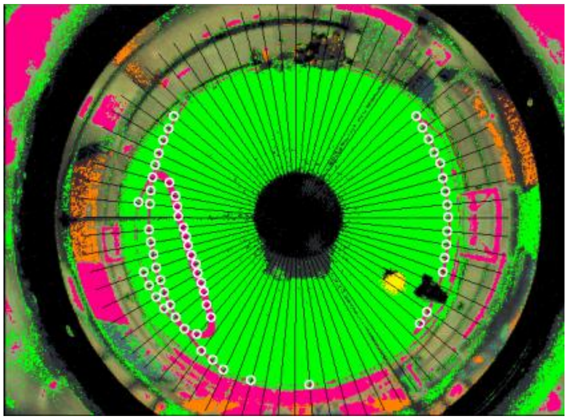

**Fig. 9.** Recognizing a group of spots

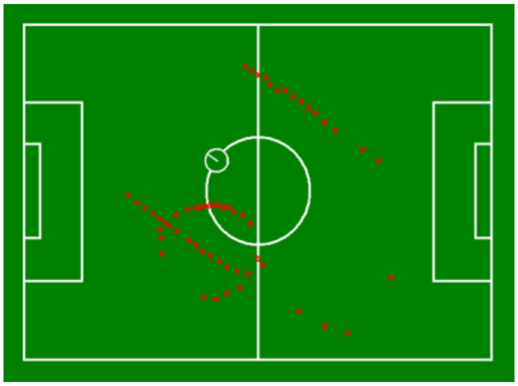

**Fig. 10.** Converting from pixel mode to metric mode

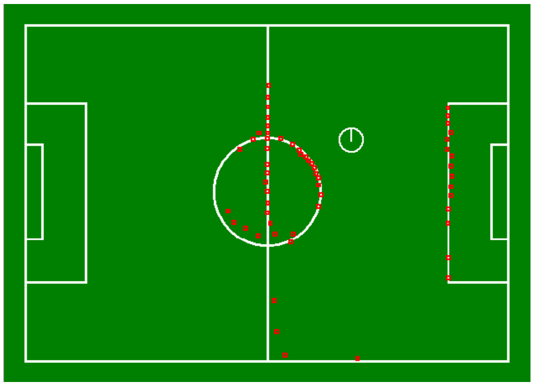

**Fig. 11.** Localizing by matching

Successful localization could be achieved by fusing data from various sensors to reduce the uncertainty. We use a gyroscope (IMU) sensor to calculate the angles of robot to accelerate matching process.

## References

1. V.Rostami, S.Ebrahimijam, P.Khajehpoor, P.Mirzaei, M. Yousefiazar "Cooperative Multi Agent Soccer Robot Team," International conference in enformatika system science and engineering, Volume 9, November 25-27, 2005, ISBN 975-98458-8-1, page 95-98.
2. J.W. Choi, R.E. Curry, G.H. Elkaim, Real-time obstacle avoiding path planning for mobile robots, in: Proceedings of the AIAA Guidance, Navigation and Control Conference, AIAA GNC 2010, Toronto, Ontario, Canada, 2010.
3. Nicolaj C. Stache, Henrik Zimmer, "Robust Circle Fitting in Industrial Vision for Process Control of Laser Welding", Proceedings of the 11th International Student Conference on Electrical Engineering, Prague MAY 2007.