

MRL Team Description Paper for Humanoid KidSize League of RoboCup 2018

Meisam Teimouri¹, Alireza Fatehi, Hamed Mahmoudi, Parham Sagharichi Ha, Amir Gholami, Mohammad Hossein Delavaran, Fateme Movafegh, Golnoush Rahmani and Erfan Fathi

Mechatronics Research Lab, Dept. of Computer and Electrical Engineering,
Qazvin Islamic Azad University, Qazvin, Iran
Email: m.teimouri@qiau.ac.ir
Web: <http://mrl.ir>

Abstract. This team description paper presents the specifications of the MRL KidSize humanoid robot system which contains different parts including system overview, robot vision, world modeling and motion control. MRL humanoid team is developed under the RoboCup 2018 rules to participate in the KidSize humanoid soccer league competition in Montreal, Canada and like the last years we will introduce a referee with sufficient knowledge of the rules available during the competitions. We use self-designed and self-constructed robots to participate in the competitions.

Keywords: RoboCup, KidSize Humanoid League, Bipedal Locomotion, World Model.

1 Introduction

RoboCup uses soccer as a research area to develop a team of humanoid robots that can win the human world champion soccer team in 2050. In the Humanoid league, human-like fully autonomous robots play soccer against each other and meanwhile handle stable walking, visual perception of the ball, players, and the field, modeling and kicking the ball, and also self-localization. The RoboCup soccer playing robots introduce challenges in design, control, stability, and behavior of autonomous humanoid robots.

The MRL project was started in 2003 in the Mechatronics Research Laboratory in Islamic Azad University, Qazvin branch looking onward to enhance the knowledge of robotics and the MRL humanoid KidSize soccer league is aimed to develop a humanoid platform for research and education. Our research center has the honor to hold the RoboCup IranOpen from 2003 to 2017. MRL has nine qualified teams and has had a successful history in RoboCup for many years. Our humanoid soccer playing team is one of the developing soccer-playing humanoid robots in the RoboCup Humanoid League and has participated in RoboCup and IranOpen Humanoid League since 2011. A brief highlights of our participation in the RoboCup and IranOpen competitions are as follows:

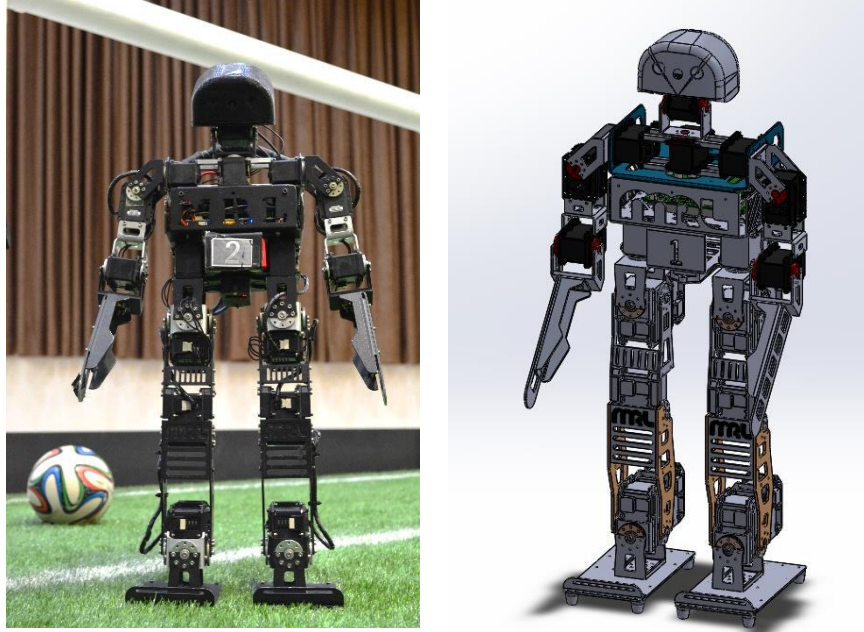


Fig. 1. Ashkan humanoid robot.

- RoboCup 2012, 2013, 2014, and 2017: reach quarter final in main competition. In 2017 we took first place in technical challenge competition.
- IranOpen 2013 and 2017: take first place and second place in 2013 and 2017 respectively.

This year we are planning to participate in the KidSize humanoid competition at RoboCup 2018 in Montreal, Canada. Our mission is to fulfill our study in motion control, vision, world modeling, and artificial intelligence.

MRL Humanoid Kid Size team consists of some researchers and students from software, hardware, electronics, and mechatronics.

2 Overview of the System

With the experiences gained by participating in RoboCup and IranOpen competitions, last year by eliminating all bending we designed a new KidSize robot that made manufacturing process easier and more accurate (Fig. 1). It has a well-known 20 degree of freedom structure with 58 cm tall and weight of 4.3 kg. All joints are equipped with Robotis Dynamixel MX series actuators. We have used six Dynamixel MX-64 for each leg, three Dynamixel MX-28 for each arm and two Dynamixel MX-28 in neck and head. The robot is powered by a 3-Cell, 2200mAh LiPo battery. Power management and actuators data streaming is done by our own made Controller Board (HSL-CB

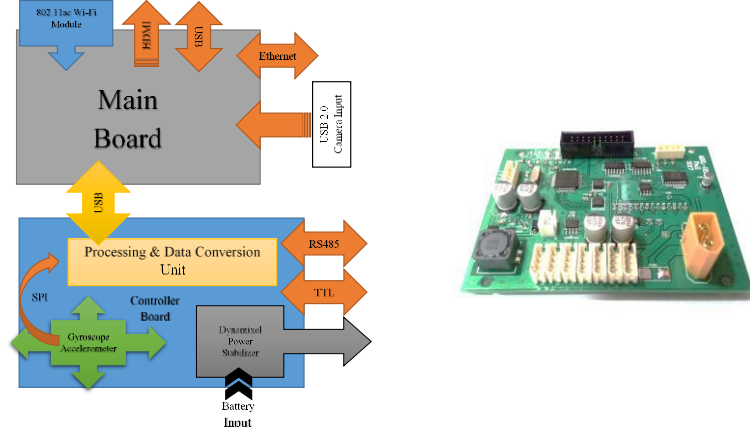


Fig. 2. Left: overview of our electronic system. Right: our self-constructed controller board.

shown in Fig. 2) and an Intel NUC is used as the main processing unit. Visual perception is done by a Logitech C920 normal webcam with 78° field of view. All mechanical parts of the robot are made of aluminum alloy 6061. We manufactured robot components by utilizing CNC Milling in order to increase accuracy.

The processing unit of controller board is based on an ARM STM32F405VE microcontroller which is designed to communicate with both T and R Dynamixel series at the same time. This board is equipped with 2 inertial sensors (gyroscope and accelerometer) and is in charge of three different tasks:

1. Data streaming according to each line protocol (RS485 or TTL).
2. Sampling inertial sensors, fusion them, and estimating trunk roll, pitch and yaw, according to [1].
3. Handling power distribution and stabilizing voltage for each chain of motors.

Our developments for the KidSize humanoid robot include the design and construction of modular software architecture based on the UPennalizers RoboCup released code [2]. The software contains visual perception, autonomous motion and walking controller, self-localization and communication. The project is still in progress and some developed methods are described in the current report.

3 Robot Vision

Vision is one of the most important interfaces for robot perception. The main vision sensor is a camera that is located in the robot's head. At the first step, we use V4L2 driver to grab the image in YUYV format. Then a color segmented image is constructed using a pre trained lookup table. Due to the recent changes in humanoid robot league

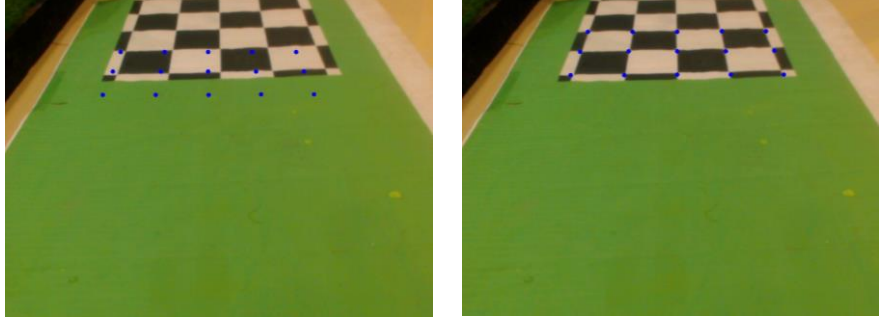


Fig. 3. Left: projected points before calibration. Right: projected points after calibration.

rules distinguishing objects relying only on the color segmented image is not feasible. So we are working on more efficient methods especially machine learning approaches.

3.1 Camera calibration

Camera is the main source of environment observations in a soccer playing humanoid robot. All interested objects in the field of play are detected using this sensor. To make an accurate world model a robot needs to estimate distance of the objects relative to an egocentric coordinate system. A typical method to calculate the distance is back projection that converts a 2D point from image coordinate frame to a 3D point on the robot coordinate system. In this method the relation between these two points is described usually by the Pinhole camera model [3]:

$$\mathbf{p}^I = \frac{1}{z_c} A \times B \times \mathbf{p}^R \quad A = \begin{bmatrix} \frac{f}{s_x} & 0 & u_0 & 0 \\ 0 & \frac{f}{s_y} & v_0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}, \quad B = \begin{bmatrix} R_{3 \times 3} & T_{3 \times 1} \\ 0 & 1 \end{bmatrix} \quad (1)$$

where A and B are intrinsic and extrinsic camera parameters respectively. In matrix A , focal length is specified by f in pixel unit and s_x , s_y are the horizontal and vertical scales of a pixel. The point (u_0, v_0) determines the center of image plane. B is a homogeneous transformation matrix that maps a point $\mathbf{p}^R$ described in robot coordinate system to a point $\mathbf{p}^C$ in camera coordinate frame. Also z_c is the z component of $\mathbf{p}^C$ that is expressed on the principal axis of the robot camera.

Considering a 2D point $\mathbf{p}^I$ in image plane, to get $\mathbf{p}^R$, there exists an infinite set of points on a ray connecting the camera center and $\mathbf{p}^I$ that projected to the same point $\mathbf{p}^I$. However we know that all desired objects are located on the field. So to get a unique point the intersection of the ray and field is calculated.

Manufacturing and aligning the camera lens, introduce some radial and tangential distortion that should be corrected. These imperfections are more critical on low cost cameras. To obtain radial, tangential and intrinsic parameters we have employed the Matlab camera calibration toolbox [3]. Also matrix B has significant affect on distance calculation. This matrix is derived from forward kinematics of our robot. Any bias in

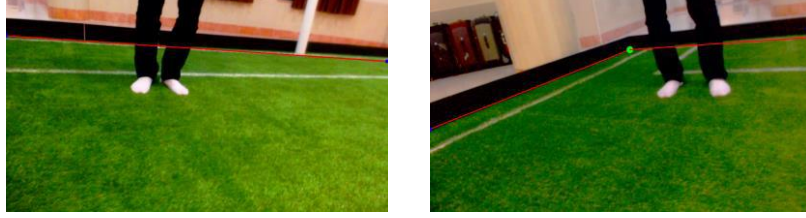


Fig. 4. Fitting line on the field boundary end points. Fitted lines marked with red. Left. Fitting by a single line. Right. Fitting by two lines.

joints, robot manufacturing and assembling can make this matrix far from the true matrix. To calibrate the parameters that contribute on matrix B , we have implemented an approach based on the particle swarm optimization (PSO). First we capture many corner points of a chessboard marker located in some different kwon positions relative to the robot. Then we try to discover a set of parameters that minimizes below function:

$$err(parm_i) = \sum_{j \leq n} (p_{observed,j}^l - p_{predicted,i}^l)^2 \quad (2)$$

where n is the number of captured points and $p_{predicted,i}^l$ is calculated using back projection incorporating $parm_i$. The result of our method is shown in **Error! Reference source not found.** To accelerate the calibration process we are developing a more automatic approach that optimizes all parameters simultaneously [4].

3.2 Line detection

Line based landmarks are distributed over the field. Since they are surrounded by green pixels they can be detected efficiently. In order to detect these landmarks first white segmented regions on the green area are skeletonized. Then a Hough Transform algorithm is applied to get a set of lines. These lines are projected to world coordinate system and the lines that can form a circle are classified as a circle landmark and are removed from the set. Remaining lines are merged to make longer lines if it is possible. Finally end points of lines are analyzed for detecting the intersections.

3.3 Field boundary detection

Field boundary detection is important, because our localization and further image processing algorithms are related on it. For detection of field boundary we have used the color segmented image. At the first step we scan the image column by column. For each vertical scan line we start from the bottommost pixel and go upward until there is a probable field color. A pixel is treated as a field color if it has been classified as green color or the significant of n adjacent pixels under it be green. The last field pixel on the column i must be green and is marked as the end point of that column. These end points form a histogram that describes the field boundary. There are other objects on the field

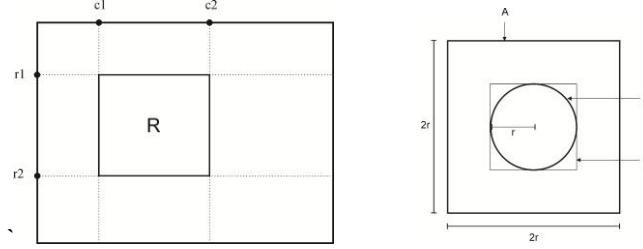


Fig. 5. Left: Calculation of the number of occurrences of $\mathbf{C}$ colors inside Region $\mathbf{R}$. Right: the template that are used to calculate the DOC of $\mathbf{p}_i$.

that can cause some gaps in the resulted histogram. To cope with this problem we have employed the Graham Convex Hull algorithm. This year we have used the field boundary as a valuable landmark for self-localization. The end points of the field boundary fitted by a maximum of two lines using a RANSAC method as suggested in [5]. So we incorporate them in localization like field lines. The result of our field boundary detection algorithm is illustrated in Fig. 4.

3.4 Ball detection

According to the new rule, ball has not a predefined color or pattern, so it seems that simple color segmented based approaches have been obsoleted. Last year we have proposed a new algorithm based on the particle filter and Adaboost classifier. The algorithm can be divided in two steps. At the first step some random particles are generated as suggested in [6] and at the second step the particles are validated. Each particle $\mathbf{p}_i$ is a circle with radius r located at a random pixel on the image. The radius of $\mathbf{p}_i$ can be derived by its location and the camera matrix. Then the probability that $\mathbf{p}_i$ is a bull must be evaluated. In [6] authors suggested the DOC to estimate this probability. Due to the manner in which they calculated the DOC, it can be a time consuming operation that will decrease the frame rate. To cope this problem we have used a more efficient approach. Assume $\mathbf{C}$ represents the set of all possible colors of an object. The integral image of $\mathbf{C}$ is:

$$II_C(x, y) = \sum_{x' \leq x, y' \leq y} F_C(x', y') \quad (3)$$

where $F_C(x', y')$ is a function that returns 1 if the classified color of the pixel (x', y') be a member of $\mathbf{C}$, otherwise it returns 0. Using the II_C we can efficiently calculate the number of occurrences of $\mathbf{C}$ colors in any rectangular region $\mathbf{R}$:

$$n_C(\mathbf{R}) = II_C(r2, c2) + II_C(r1, c1) - II_C(r1, c2) - II_C(r2, c1) \quad (4)$$

where $r1, r2, c1$ and $c2$ are some indexes that are specified in fig 3. Left. Now we can define DOC of $\mathbf{p}_i$ using a template (fig 3. Right) assigned to it:

$$DOC_{\mathbf{p}_i} = \frac{n_{BC(B)} - n_{NBC(B)}}{n_{AC(B)}} \times \frac{n_{NBC(A)} - n_{NBC(B)}}{n_{AC(A)} - n_{AC(B)}} \quad (5)$$

where BC , NBC and AC are the set of all ball colors, non-ball colors and all colors respectively.

Relying only on the Doc may result in detection of some false positives. To reject this false candidates we pass them among a strict filter. This filter is an Adaboost cascade classifier composed of 12 stages with a maximum of 30 decision trees in each stage. There are many features like SIFT, SURF, HOG and etc. that can be used to learning the classifier. Here we have employed HOG feature. To learn the classifier we have used some positive and negative samples prepared by an expert. Due to the sensitivity of HOG to the rotation, the positive samples are rotated to generate new samples. The results of our ball detection algorithm show that it can detect the ball robustly at a distance of about 2.5 meter.

4 World Modeling

World model is a key component in intelligent and autonomous robots. Modeling the system consist of a model for each static and dynamic object in the field of play. These models are formed by the incoming data from the sensors of the robot. Due to the noise and uncertainty of the observations and limitations in humanoid sensors, tracking the surrounding environment of the robot is an important challenge. This year we have enhanced self-localization model and we are working on modeling obstacles and ball.

4.1 Self-localization

Self-localization is the process of estimating the robot position employing noisy measurements. It is a key issue for a soccer playing robots. With respect to the limited field of view and limitation in robot sensors, tracking the position is a difficult problem. Last year we have implemented a hybrid method based on the MCL and the UKF as described in [7]. The key idea of this method was that kidnap and global localization problems can be handled by MCL as quickly as possible and the position tracking can be done with UKF models efficiently. When the uncertainty of the belief is high or a kidnap detected the belief is represented by MCL samples to discover the most probable positions. When the samples are converged in the small number of clusters, they estimated with UKF models efficiently. To calculate the parameters of the new generated UKF we have used Maximum Likelihood Estimation of the samples of corresponding cluster. The efficiency of our localization method is evaluated in some experiments [7].

5 Motion Control

Developing a stable walk engine for a humanoid soccer robot is one of the most challenging research area. We have employed a Linear Inverted Pendulum to model our robot in the single support phase. Using this model and with respect to a reference ZMP, the trajectory of the CoM is generated [8]. In order to walk on the artificial grass we have designed new feet with some cleats attached under it. The maximum speed of our

robots is about 28 cm/s. Small disturbances are detected using the accelerometer and gyroscope sensors. We have designed a simple PD controller that compensate hip, knee and ankle angles to keep projected center of mass on the support polygon and prevents a fall down. Also our robot can stand up from back and front autonomously.

6 Conclusion

In this paper we have presented the specifications of the hardware and software of MRL KidSize humanoid robot system developed under the RoboCup 2018 rules. MRL commits to participate in RoboCup 2018 in Montreal, Canada with further enhanced hardware and software based on the achievements of previous year and also commits to introduce a referee familiar with the rules of the Humanoid League.

We use our self-designed and self-constructed robots and we are working on this platform with some interested researchers and students modifying and optimizing the platform in vision, motion control, world modeling, behavior, and embedded control board.

References

1. P. Allgeuer and S. Behnke, "Fused angles: A representation of body orientation for balance," in *2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2015, pp. 366-373.
2. Upenn RoboCup: Code, available online: <https://fling.seas.upenn.edu/~robocup/wiki/index.php?n=Main.Code>, retrieved Jan 2013.
3. L. Deng, G. Lu, Y. Shao, M. Fei, and H. Hu, "A novel camera calibration technique based on differential evolution particle swarm optimization algorithm," *Neurocomputing*, vol. 174, pp. 456-465, 2016/01/22/ 2016.
4. D. Maier, S. Wrobel, and M. Bennewitz, "Whole-body self-calibration via graph-optimization and automatic configuration selection," in *2015 IEEE International Conference on Robotics and Automation (ICRA)*, 2015, pp. 5662-5668.
5. Y. Qian and D. D. Lee, "Adaptive Field Detection and Localization in Robot Soccer," in *RoboCup 2016: Robot World Cup XX*, S. Behnke, R. Sheh, S. Sarel, and D. D. Lee, Eds., ed Cham: Springer International Publishing, 2017, pp. 218-229.
6. Y. Hayashibara, H. Minakata, K. Irie, D. Maekawa, G. Tsukioka, Y. Suzuki, *et al.*, "CIT Brains KidSize Robot: RoboCup 2015 KidSize League Winner," in *RoboCup 2015: Robot World Cup XIX*, L. Almeida, J. Ji, G. Steinbauer, and S. Luke, Eds., ed Cham: Springer International Publishing, 2015, pp. 153-164.
7. M. Teimouri, M. E. Salehi, and M. R. Meybodi, "A hybrid localization method for a soccer playing robot," in *2016 Artificial Intelligence and Robotics (IRANOPEN)*, 2016, pp. 127-132.
8. S. Kajita, H. Hirukawa, K. Harada, and K. Yokoi, "Biped Walking," in *Introduction to Humanoid Robotics*, S. Kajita, H. Hirukawa, K. Harada, and K. Yokoi, Eds., ed Berlin, Heidelberg: Springer Berlin Heidelberg, 2014, pp. 105-158.