AUT-UofM Humanoid TeenSize Team Description Paper

RoboCup 2015 Humanoid TeenSize Robot League

Sepehr Ramezani[1], Mohammadhossein Heydari[1], Faraz Shamshirdar[1], Sepehr Behjou[1], Mehran Ahmadi[1], Mojtaba Karimi[1], Bita Azari[1], Sourosh Sadeghnejad[1], Amirhossein Hosseinmemar[2], John Anderson[2] and Jacky Baltes[2]

[1]*Humanoid Robotics Laboratory*
Amirkabir Robotic Institute,
Amirkabir University of Technology (Tehran Polytechnic),
No. 424, Hafez Ave., Tehran, IRAN., P. O. Box 15875-4413.
http://autman.aut.ac.ir

[2]Autonomous Agents Laboratory
University of Manitoba, Winnipeg, Canada, R3T 2N2
http://aalab.cs.umanitoba.ca

Abstract. This document introduces AUT-UofM humanoid joint team for participating in Humanoid Teen Size Robot League in RoboCup 2015 in Hefei, China. This team was founded in 2013 on collaboration between AUTMan humanoid team from Amirkabir University of Technology (Tehran Polytechnic), Iran and team Snobots from University of Manitoba, Canada. Our humanoid Teen Size research is mainly based on both universities' experiences provided from long time participation in RoboCup humanoid league. A brief history of Team AUT-UofM and its research interests and directions for future research will be described. Our main research interests within the scope of humanoid robots are robust walking, accurate localization, and strategic reasoning for soccer.

Keywords. RoboCup 2015, humanoid joint team, robust walking, localization, strategic reasoning.

1 Introduction

RoboCup is an important challenge problem in robotics. RoboCup is pursuing the goal which states "By the year 2050, develop a team of fully autonomous humanoid robots to win against the human world cup champion team". More importantly, RoboCup acts as an important benchmark in developing useful and practical humanoid robots for society. An ongoing concern for RoboCup is the fact that the costs of developing a team of humanoid robots is increasing rapidly as robots are getting ever more complex. As a result, it becomes difficult for teams to participate in the RoboCup competitions [1]. The Teen and Adult sized sub-leagues of the RoboCup humanoid league have some of the most complex and expensive robots and are thus greatly affected by these financial issues.

The introduction of a minimum height (45 cm) and the increase in the maximum height of (90cm) for the robots in the KidSize sub-league has clearly shown the strong direction of the humanoid league towards larger robots. However, larger robots clearly put a great strain on the limited resources available to most teams. The main issue is of course research funding, but also the number of students, lab space, and other resources needed to develop, run, and maintain the robots. And fielding a team of at least 11 fully capable adult sized humanoid robots, which is necessary to achieve our 2050 goal, will be impossible for a single team. Therefore, it is necessary to collaborate in other teams' research programs and make joint teams [2]. This is one of the RoboCup goals to encourage different groups from different universities and countries to have collaboration with each other [3]. Amirkabir University of Technology (Tehran Polytechnic), Tehran, Iran have been participating in Humanoid League of RoboCup competitions since 2011 and Prof. Baltes has participated with teams in the Humanoid League since its inception in 2002. Therefore, team AUTMan and Snobots decided to continue their successful collaboration and extend their joint team efforts with AUT-UofM in 2015. We expected and experienced initial problems as both teams tried to convince the other side to continue using their existing system. In spite of all these problems that need to be overcome, it was an extremely important and valuable lessons for all involved. We believe that such collaboration is vital for the future of RoboCup. Lessons learned from this and similar close collaboration will be crucially important for the future of RoboCup humanoid league, as it allows the organizers to fine tune rules for joint teams from actual experience rather than guesswork.

2 Recent Achievements

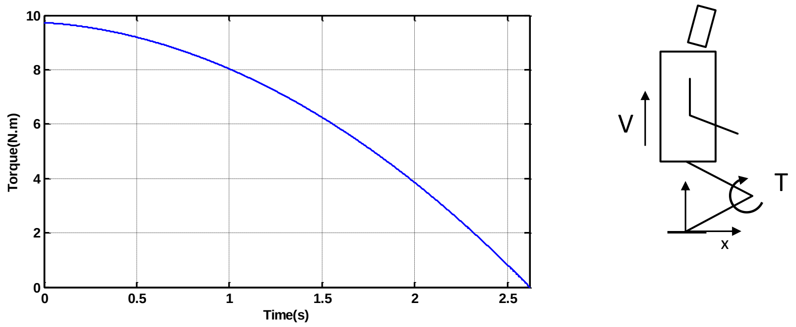
Team AUT-UofM Teen Size participated with great success in last years' RoboCup 2014 event in Joao Pessoa, Brazil. The team placed 1st in the Technical Challenge together with Team Nimbro from University of Bonn, Germany. Also the team won the 3rd Place in the Humanoid Teen Size Robot Sprint Challenge in International FIRA RoboWorld Cup 2014, Beijing, China. Both teams [4-5] have also published their research, extensively. In the last five years, team members have published several papers in the RoboCup symposium and are planning to submit papers to the 2015 RoboCup Symposium.

3 Development for 2015

In 2015, the goal of the collaboration between these teams and also participation in RoboCup 2015 is three-fold. First, the implementation of improvements for our teen sized humanoid robots framework and software by combining the extensive experiences from both teams. The main research thrust for the new robot design is to move away from ZMP based control to running robots. The other goal is to continue to demonstrate the feasibility of joint teams in RoboCup and the third one is to modify our previous Teen Size humanoid platform robot design called ARASH.

4 Hardware Design

ARASH (**A**nthropomorphic **R**obot **A**ugmented with **S**ense of **H**uman) is 100cm in height and 7.5Kg in weight. The new robot's kinematic structure has 20 degrees of freedom (DOF). The design uses 6 DOFs for each leg and 3 degrees of freedom for each arm. The camera of the robot is mounted on a 2 servo pan and tilt mechanism. MX-106 series in leg joints, MX-64 series in arms and MX-28 series in neck have been used [6]. To reach a higher performance two motors have used in the knee joints. During robot missions, worst case will occur when the robot goes to "stand up" condition. In this condition, large amount of torque will be applied on knee motors. To obtain motor torques in stand up mode, simplified model of robot dynamics (Figure 1) is analyzed. In this way, it is considered stand up velocity is constant (0.12 m/s) and mass of robot is 7.5kg. Simulation results show that knee motors experience maximum torque in beginning of movement. According to Figure 1 maximum of torque is about 10N.m. in full charge battery (14.8v), so stall torque of MX-106 is equal to maximum torque. By decrease in charge level, motors will not be able to pick up robot. So for motor failure risk, decreasing and increasing standup velocity, we use two synchronized parallel motor, simultaneously. Configuration of such actuators grants the robot wide range of motion and fast reaction. In this robot, AUT-UofM has focused on robustness, weight reduction and also nimble reaction.



A B

Fig. 1 a) torque of knee motors during stand up b) scheme of robot model

OpenCM9.04, an open-source controller that runs under 32bit ARM Cortex-M3 from ROBOTIS.Co [7], is adopted as a low level controller and Device Communication Manager (DCM) in our robot. It used for more compatibility with Dynamixel bus and easier sensory information fusion of inertial measurement unit. In other hand, CM9.04 controller used for sensor fusion and low level sensors filtering. It communicates with upper layer (PC) on serial interface @1Mbps. Controller combines internal gyro, accelerometer and manometer sensors data of GY-80 [8] IMU and it can provide 3D posture of robot COM in real-time.

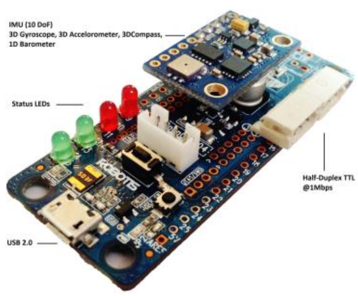


Fig. 2 CM9.04 low-level controller, main controller and peripheral connected device

Figure 2 shows low-level controller and peripheral connected device. We run Free RTOS (Real-Time Operating System) [9] on OpenCM9.04 controller for executing different tasks in ARM processor unit. Following diagram shows the whole structure of OpenCm9.04 process tasks as a DCM. Communication of controller and actuators is based on Half-Duplex TTL serial @1Mbps. In ARASH new version, MAXData QutePC-3001 [10] mini embedded board is used as a main controller. High performance and low power consumption are the main factor for using this kind of main boards as a main processor in humanoid robots.

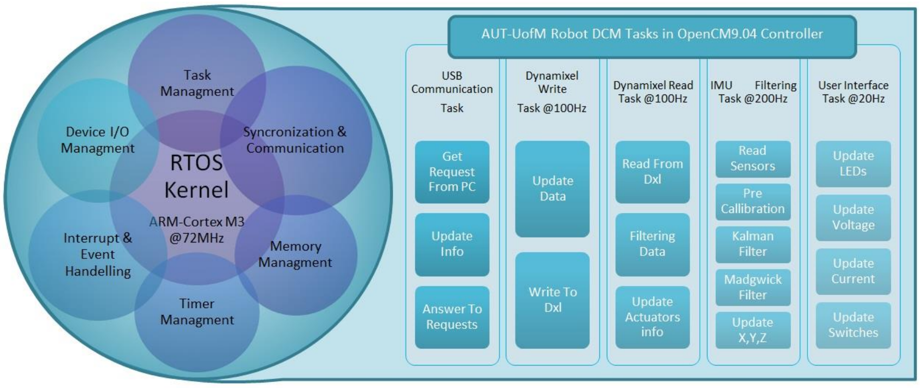


Fig. 3 OpenCm9.04 Controller internal tasks based on RTOS and filtering process diagram

5 Software Development

5.1 Software Architecture

The robotics community attachment to ROS [11], as a reliable software framework, prompted us to migrate our code base to it. This effort already started last year [2]. The results of this migration is a completely ported code to ROS (and Linux), ability of releasing portions of (or the whole) code as ROS packages (since the software licenses are more open) and use other people packages in the code base to speed up the development. In the software architecture section, we focused on improving the developing experience by automating some procedures that can also be done by the machine.

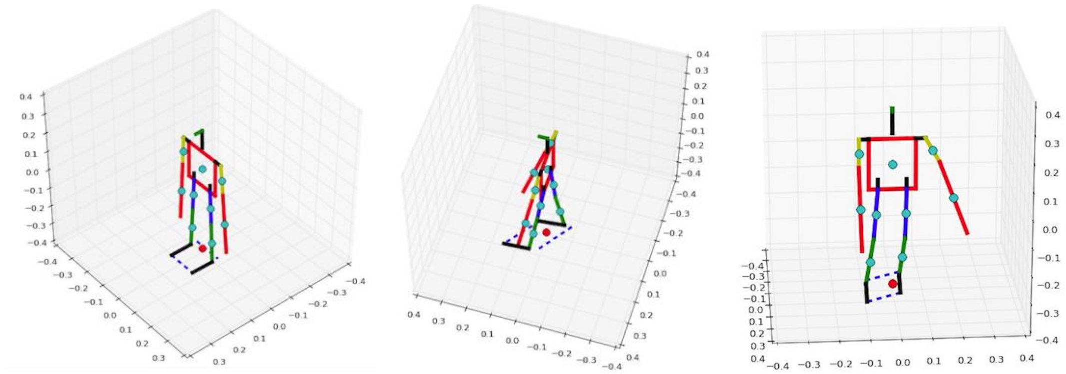
5.2 Simulation

It is obvious that a simulation is a must to ensure the functionality and reliability of a system. ROS standardized the simulation in a software package named rviz, it also helps to keep the shareability of code in a ROS platform. Considering this, there are good implementations freely available like SimRobot [12] specialized for humanoid soccer robotics that can be used as a reference for the design.

5.3 Motion

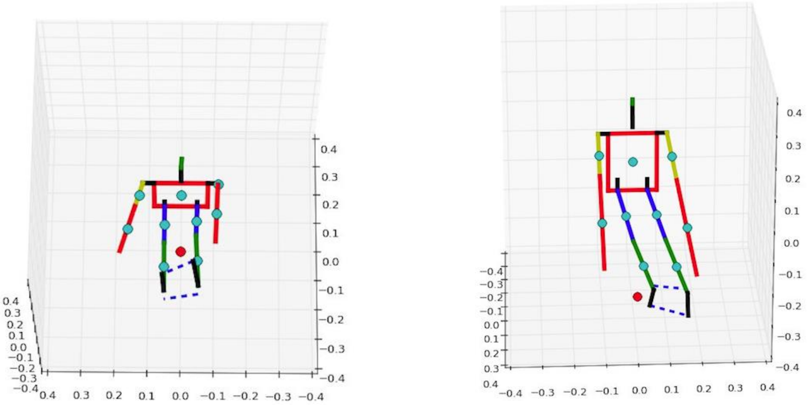
Last year we used a traditional ZMP controller, but for this year we have implemented four different dynamic walking engines (inspired by cyclical pattern generators (CPG))

which use different algorithms for interpolation. The four interpolation algorithms which we have tested are: Cubic Spline Interpolation, Bezier Curve Interpolation, Nearest Neighbour Interpolation and Linear Interpolation. We use these novel motion controller to improve the balance of our robot. One of the main issues that we faced by using the ZMP was the slow response of the robot. This means that for such a competition we believe that it is better not to use the ZMP and use a faster approach for our robot. Our walking engine is a dynamic walking and we use the real time with the nano precision to generate the interpolation. The default gait cycle time for our system is 650msec that means for one complete cycle which consists of left and right leg, it will take 650msec. This number can be changed and the lower the number is the faster the robot will finishes the walking gait cycle and the higher the number is the more time will take to finish the cycle. The current system is written in C++ programming language from scratch and we are planning to port our system into the ROS framework.
After testing ARASH with different algorithms mentioned above, our results show that the Bezier Curve Interpolation and Linear Interpolation work best. In practice, they are both fast and quick enough to accomplish the task which in here is getting to the ball very fast and quickly generate the trajectories on the fly and with almost no delay. On the other side the Bezier Curve Interpolation has it's own uniqueness and the curve will be smoother but more trajectories to go through. For now we concentrate on Bezier Curve interpolation and planning to have more tests on this approach and check the result with the linear interpolation. We have also developed a system for monitoring our degree of freedom as well as the center of mass of the robot and the support polygon. This system has been written in python and we can check the robot and forward and inverse kinematics visually in the system.



(a) (b) (c)

Fig. 4 (a) ARASH is in the standing mode and the red dot is in the support polygon which shows that the robot is stable, (b) ARASH during walking and still balanced, (c) ARASH during walking and still balanced



(a) (b)

Fig. 5 (a) Red spot outside of the support polygon which means the robot is not stable and balance anymore, (b) The red dot which showing the center of mass is outside of support polygon and robot is falling down

This will help us to tests our input first on this system and if it looked perfect then we can send our data to the servos. We also wrote this system from scratch. We show the center of mass for each individual link with the light blue spot and the center of mass for the whole body of the robot with the red spot which is on the ground. As long as the red dot is in the area of the support polygon, the robot is stable and when the red spot is outside of the support polygon it is not any more stable.

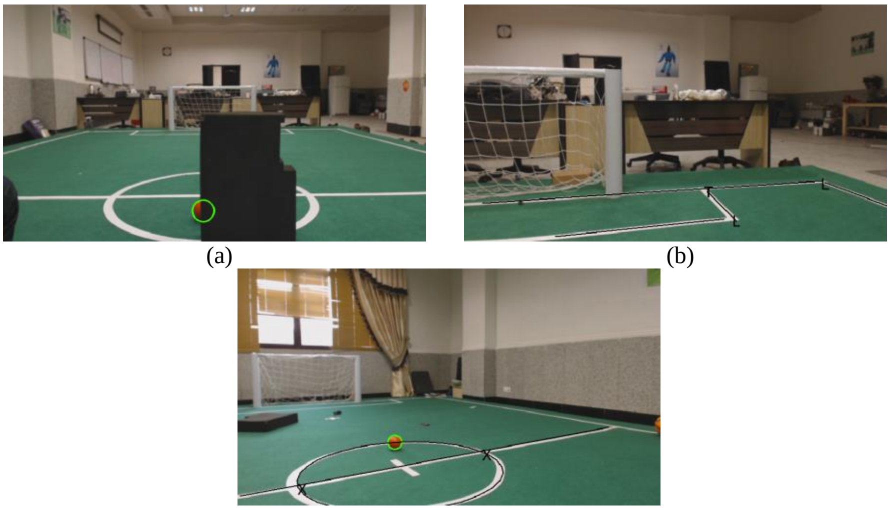
5.4 Cognition

Color classification: This part makes the further steps easier by making a knowledge base of classified colors. We process wide-angle YUV422 images from a Logitech C920. A set of random pixels are selected by human supervision in the image and a Lookup table is constructed based on it. This Lookup table is a mapping from YUV color space to a set of colors and assigns a class label to every pixel. For generating this table the k-d tree algorithm categorize similar pixels in labels that makes the known space grow fast.

Vertical Scan and Segmentation: The main purpose of segmentation layer is to find an approximate coordination of objects in the image by clipping color labeled ROIs. The ROIs, at first, are rectangles made by a vertical scan line algorithm, to improve the performance of this algorithm a convex hull of field and outside this hull will be ignored. Afterwards the adjacent segments are merged and make shapes. The identity of shape is guessed by heuristics which either a circle (the ball) or a line (the field lines and the goal posts) this strategy significantly improves the performance since the color labeling is not efficient anymore.

Feature Extraction and Object Recognition: In this step, each segment is processed by its corresponding feature extractor unit based on the color and the shape of the segment. These features extractors are explained in details in AUT-UofM last year TDP [2].

Modeling: Each object in the field is needed to be a model in the robot's mind. Modeling enable us to improve the overall object detections by using the past and teammates data. These models are more reliable since the objects detected in previous step can be a false or inaccurate detection. All these models are integrated in a major model named world model shared between teammates. By having the world model, each object position in the field, velocity (if has one) and their certainties, by the means of covariance matrices, are accessible.
Self-Localization: Using the world model Monte Carlo simulation accompanying Unscented Kalman Filter is utilized to approximate the robot position in the field. As it mentioned above that the world model stores position of objects relative to the field center, they were calculated by knowing the position of robot in the field. It also enables the robots to share the other teammate's objects positions and decide based on a consensus of opinions.



(a) (b) (c)

Fig. 7 (a) Detection of occluded ball, (b) Detection line intersections (L-corners and T-junctions), (c) Detection of balls, field center and line X-crossings

5.5 Robot Navigation

Each robot in dynamic environments needs path planning and navigation based on the planned path. Humanoid robots navigation has some more difficulties than other mobile robots due to their bipedal walking. We have used footstep planner package for navigating and finding the optimal path with the map providing the obstacles positions [14]. The map is provided by the localization module. The planner is based on SBPL and capable of dynamic re-planning. As the foot parameters differ for each robot, we changed the parameters to meet our new built robots' configuration. So, they plans path based on new parameters.

6 Conclusion and Acknowledgment

This report described the future technical plans and works done by the AUT-UofM Humanoid Teen Size Robot Joint Team for its entry in the RoboCup 2015 Humanoid Teen Size League. AUT-UofM team's focus, for the second year of RoboCup competition, has been on modifying our Teen Size Platform, localization, motion behavior, and vision module due to our past and relevant experience in various RoboCup leagues. We look forward to continuing and expanding our above researches with the new humanoid robots. For further information and to be familiar with our previous and new publications and recent activity done in the humanoid community and for seeing more pictures and videos, please see our official websites.

REFERENCES

1. Shangari, T.A., Shamshirdar, F., Heydari, M.H., Sadeghnejad, S., Baltes, J., Bahrami, M.: AUT-UofM Humanoid Teen Size Joint Team; A New Step Toward 2050's Humanoid League Long Term RoadMap. In: The 3rd International Conference on Robot Intelligence Technology and Applications (RiTA 2014), Beijing, China 2014
2. Members, AUT-UofM, AUT-UofM Humanoid Teen Size Team Description Paper, RoboCup 2014 Humanoid Teen Size Robot League. In: The 18th annual RoboCup International Symposium, Joao Pessoa, Brazil 2014
3. Baltes, J., Sadeghnejad, S., Seifert, D., Behnke, S.: RoboCup Humanoid League Rule Developments 2002-2014 and Future Perspectives. In: The 18th annual RoboCup International Symposium, Joao Pessoa, Brazil 2014
4. Members, S.T.: Snobot Humanoid Kid-Size Robot League Team Description Paper. In: The 18th annual RoboCup International Symposium, Joao Pessoa, Brazil 2014
5. Members, A.T.: AUTMan Humanoid Kid-Size Robot League Team Description Paper. In: The 18th annual RoboCup International Symposium, Joao Pessoa, Brazil 2014
6. ROBOTIS.: Dynamixel Actuator. Dynamixel Actuator, 2013.
7. ROBOTIS.: OpenCM9.04. Available. http://www.robotis.com/xe/darwin_en, 2013.
8. Arduino. GY-80 10DOF module, nine-axis attitude instrument, L3G4200D ADXL345 HMC5883L BMP085 module. Available: http://arduino.cc/en/, 2013.
9. R. Barry, "Real time engineering ltd: FreeRTOS main website, http://www.freertos.org."
10. MaxData Cooperation. http://www.maxdata.de.
11. Morgan, Q., Conley, K., Gerkey, B., Faust, J., Foote, T., Leibs, J., Wheeler, R., and Andrew Y. Ng. "ROS: an open-source Robot Operating System." In ICRA workshop on open source software, vol. 3, no. 3.2, p. 5. 2009
12. Laue, T., & Röfer, T. Simrobot-development and applications. In: International Conference on Simulation, Modeling and Programming for Autonomous Robots (SIMPAR), (2008).
13. Taubin, G., "Estimation of planar curves, surfaces, and nonplanar space curves defined by implicit equations with applications to edge and range image segmentation," IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 31, pp. .3993, 3311-3331
14. Hornung, A., Dornbush, A., Likhachev, M., and M. Bennewitz, "Anytime search-based footstep planning with suboptimality bounds," in Proc. of the IEEE-RAS Int. Conf. on Humanoid Robots, 2012