UURT Humanoid Robot Team Description Paper for Humanoid KidSize League of RoboCup 2012

R. Shabani, H. Yazdanjouei, F. Tofigh*, S. Rabiei, and R. Ghasemzadeh

Robotic Lab.
Faculty of Engineering
Urmia University - Urmia 57153, Iran
uurt@urmia.ac.ir
http://www.urmia.ac.ir/englishsite/Engineering/robotic/

Abstract. This paper presents current version of humanoid robot systems of Urmia University Robotic Team (UURT) for the upcoming KidSize competition RoboCup 2012, Mexico City, Mexico. First version of our robots was developed for RoboCup IranOpen 2011 and RoboCup 2011 in Istanbul, Turkey. This year our robots' system has been modified based on our experiences of the last year. The mechanical structure is a new design using Bioloid Kit links based on Dynamixel MX-28 and AX-12 servos. Central brain of the robots is a Vortex 86DX, 32 bit x86 CPU running at 1000MHz with 256 MB RAM. This processor manages self-localization based on vision and object recognition, role engine based on fuzzy logic, Motion Controller and Network. Each robot is able to walk, pass, kick and dribble when it catches the ball.

1 Introduction

Because of containing different research fields like machine vision techniques, stability problems, machine intelligence, etc. humanoid robots are one of the challenging and interesting fields in robotic world.

UURT is a student robotic team of Urmia University, established in 2005 working on Deminer robots, Soccer 2D simulation, Humanoid and Aerospace researches and has awarded some successes in national robotic competitions in Iran. We started our works on December 2010 by working on Robotis' "Bioloid Premium Kit"[1]. But very soon we found out that we need changes to be able to control this kit so our mechanical design team, designed a new structure using connection links of Bioloid kit.

This paper provides a brief overview of our current works on humanoid KidSize robots that is going to be used during the next RoboCup competitions.

* f.tofigh@gmail.com

[1] http://www.robotis.com/

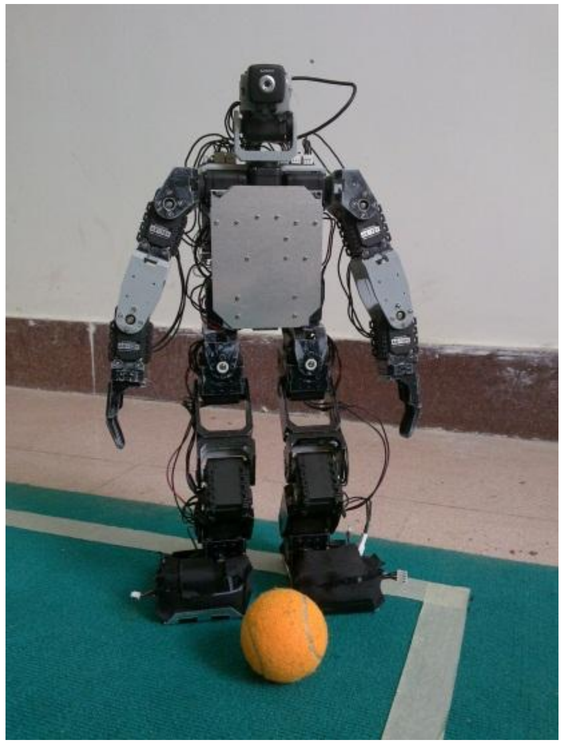

Fig. 1. UURT Humanoid KidSize Robot

2 Team members

Assistant Prof: Rasul Shabani (Team leader)
graduate Students: Farzad Tofigh[2] (contact person), Hossein Yazdanjouei, Mojtaba Mir Mobini
undergraduate Students: Sourush Rabiei, Roya Ghasemzadeh, Hamed Rostamzadeh, Mahdi Mahmudi

3 Mechanical and Electrical Designs

The basic set of the "Bioloid Premium Kit" was enhanced by changing the complete mechanical structure and using Robotis Dynamixel MX-28 servos and a 6 degree of freedom Inertial Measurement Unit (IMU) sensor. The Robot has 20 actuated degrees of freedom, 6 per leg, 3 per arm and 2 degrees of freedom for neck. The robot is controlled by RoBoard RB-110 based on the Vortex86DX,

[2] Referee candidate

32 bit x86 1000MHz CPU with 256MB RAM[3]. It is equipped with VIA VT6655 Chipset wireless adapter. RB-110 is powered by a Li-Polymer battery providing 2 Ah, which gives us 40 minutes of run time. A combination of Gyroscope and acceleration sensors has been used to facilitate the control of robot to have a stable motion. All sensors are monitored using 10 bit A/D embedded on RoBoard RB-110.

The actuators are Dynamixel MX-28 and AX-12 and the camera is Cam Serial JPEG Camera Module connected to RB-110 using full duplex TTL port and equipped with a pan/tilt. Actuators are powered by an additional Li-Polymer battery and communicate with RB-110 through TTL port of communication. Physical specifications of robots are 47cm height and 1.9 kg weight which is classified in KidSize according to RoboCup rules.

4 Software Design

The designed software is a single process with 3 to 5 threads. The main thread runs decision making algorithm which is a simple finite-state machine. Input of this machine is provided by role engine in form of high-level text commands. Other threads run image processing algorithms, motion algorithms and control decisions according sensors' data.

Considering this fact that instruction packets must be run in actuators, sequentially, sending motion packets and running them is a time-consuming process so a real-time system is needed to synchronize threads. The operating system running on robot is an optimized compilation of LFS 7.0 for i586 architecture. The kernel of this Linux system is patched with the Xenomai real-time subsystem. So scheduling mechanism is pretty different from regular kernels in major distributions of Linux systems. Goal of this kernel is to maintain hard-real-time features in user space. Of course using this scheduling algorithms in a normal program, requires a special API for multi-threading and concurrency. There are four different APIs provided by Xenomai team. We used native POSIX-based API in order to maintain compatibility with normal kernels.

4.1 Image Processing Software

To have an environmental perception a vision system is essential. The main goal of vision system is to detect objects of interest, such as ball, goals and robots. The input of this system is the image frames acquired by camera and are fed to image processing algorithm.

Image processing algorithm is divided into two main parts: a common preprocessing stage and an object recognition algorithm.

In the preprocessing stage, some filters like mean filter and median filter is used to reduce noise. In the second stage, at first RGB color space is converted to HSV color space. Then special colors have been searched filtering special ranges

[3] http://www.roboard.com/

in Hue and Value matrices. These ranges themselves, are obtained by a small GUI tool **"Vision Control"**. Output is an XML1-compatible file containing HSV ranges of colors and information about ball, camera and field. Then this file is used by vision algorithms as an input reference. This makes program more flexible in different lighting conditions. After that some features like the distances between the points on the boundary of objects and center of them are extracted from the images and artificial neural networks have been used to recognize ball, goals, opponent and team-mate robots.

An additional detection algorithm using Hough transform is used to identify the lines on the field[1]. By knowing the lines we are able to detect specific marks on the field. Based on the found objects and an internal model, the robot will try to localize itself on the field[2,3].

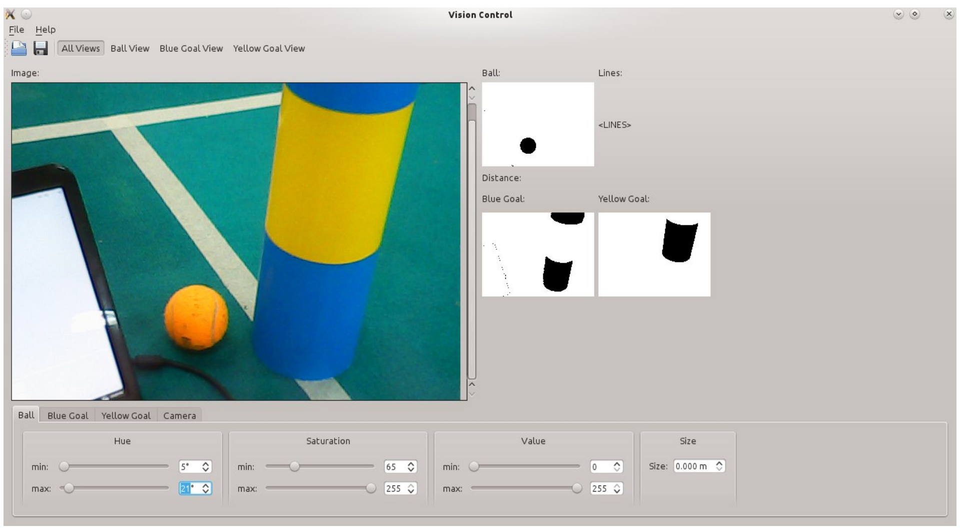

Fig. 2. Running Vision Control program - All segmentations

4.2 Role Engine

Once the position of objects of interest in the field was detected by image processing software, the role of each robot will be determined using a fuzzy inference system[4,5].

These roles are some linguistic commands made by fuzzy inference engine using Takagi-Sugeno model. The membership functions used in fuzzy inference system are determined by experience and have triangular and trapezoidal shapes. Then these high-level linguistic commands are submitted to behaviour engine.

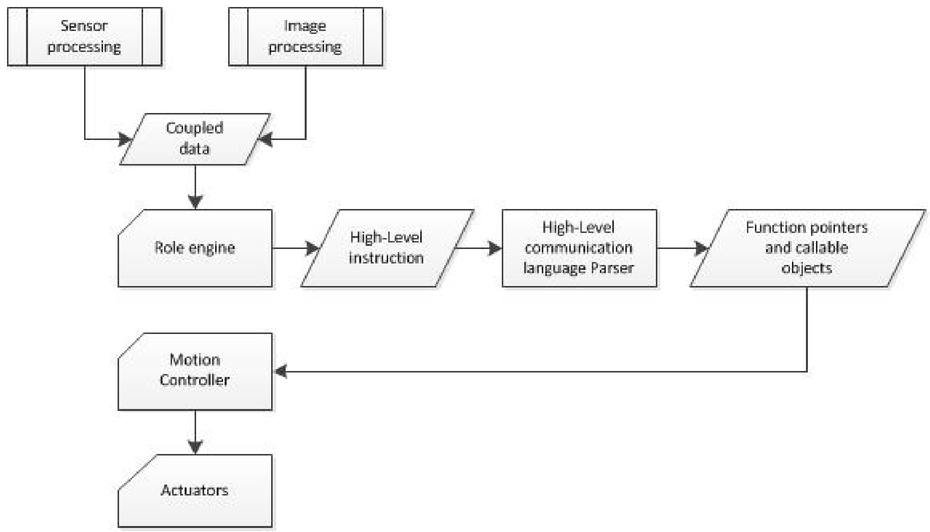

Fig. 3. Schematic of control architectures

4.3 Behaviour Engine

The main duty of behaviour engine is to covert High-level linguistic commands received from role engine to low-level commands to be performed by each robot.

Behaviour engine consists of two parts: High-level communication language parser and a static collection of external data structures. Language parser is a simple deterministic finite automata (dfa) which constructs data structures. Data structures are callable objects and function pointers wrapped by thread objects. These threads run concurrently or synchronized to each other due to pre-defined and constant conditions.

4.4 Motion Control

Motion control is the part that converts low-level commands issued by behaviour engine to a sequence of predefined packets for motors. These packets have been modified using Robomotion software developed by Robotis company. To use the mtn files produced by Robomotion directly, a text parser have been developed.

5 Conclusion and Future work

As explained in previous sections of this paper lots of hardware and software tools were developed by UURT to reach good and successful achievements in RoboCup 2012 competitions. to reach this goal our robots are able to find the ball, turn to ball,walk, kick and also dive to catch the ball, autonomously. Currently, we are focused on implementing dynamic real-time control on our robots and also communication between robots to control the game in contact with each other. Team progress reports is available on our homepage.

References

1. Gonzalez, R.: Digital image processing. Prentice Hall, Upper Saddle River, N.J (2008)
2. Rath, C.: Self-localization of a biped robot in the robocup domain. Master's thesis, Institute for Software Technology, Graz University of Technology (2010)
3. Chang, S.H., Chang, W.H., Hsia, C.H., Ye, F., Chiang, J.S.: Efficient neural network approach of self-localization for humanoid robot. In: Pervasive Computing (JCPC), 2009 Joint Conferences on. (dec. 2009) 149 –154
4. Kasaei, S., Kasaei, S., Kasaei, S., Taheri, M.: design and implementation a robust structure and fuzzy rule-based motion controller for a humanoid soccer robot. BRAIN. Broad Research in Artificial Intelligence and Neuroscience **1** (July 2010)
5. Kasaei, S., Kasaei, S., Kasaei, S., Taheri, M.: Dynamic role engine and formation control for cooperating agents. In: the International Conference on Man-Machine Systems (ICoMMS), Batu Ferringhi, Penang, MALAYSIA (October 2009)