Team THORwIn

Team Description for Humanoid AdultSize League of RoboCup 2015

Stephen McGill[1,*], Seung-Joon Yi[1], Dennis Hong[2] and Daniel Lee[1]

[1] GRASP Lab, School of Engineering and Applied Science, University of Pennsylvania, USA
`{smcgill3,yiseung,ddlee}@seas.upenn.edu`
[2] RoMeLa, School of Engineering and Applied Science, University of California, Los Angeles, USA
`{dennishong}@ucla.edu`
`http://www.seas.upenn.edu/~robocup/`

Abstract. This paper details the hardware, software and electrical design of Team THORwIn's humanoid platform for the adult size league at RoboCup 2015 this year in Hefei, China. Team THORwIn uses a bipedal robot that competed in João Pessoa, Brazil for RoboCup 2014 that shares its lineage with previous RoboCup humanoid platforms including DARwIn-OP, CHARLI, and DARwIn-XOS and their accompanying robot soccer autonomy since RoboCup 2007 [1,2].

1 Introduction

Team THORwIn (Tactical hazardous Operations Robot with Intelligence) is competing with a fully autonomous, soccer-playing robot: THOR-OP (Tactical Hazardous Operations Robot, Open Source), shown in Figure 1. This is a modular, full-sized and low-cost humanoid robot developed and successfully used for a number of practical applications such as disaster response and fire suppression. The ultimate goal of humanoid robots is to work in environments designed for humans, and RoboCup aims to help achieve this goal by defeating the FIFA World Cup champions.

The THOR-OP is 1.47 m tall and weighing just 49 kg, with 31 degrees-of-freedom powered by Dynamixel PRO servo actuators that use cycloidal reduction gears in them to have higher impact tolerance compared to harmonic drives. The hardware consists of standardized and general purpose actuators and structural components, and the software is also fully modular with multiple control layers including a hybrid locomotion controller, a hierarchical arm controller and a platform independent operator interface.

RoboCup 2015 will be the second time competing with this humanoid platform for soccer in the adult size league. Additionally, we have used CHARLI, DARwIn-OP, and XOS in the adult, kid and teen size leagues respectively with very successful results. Our software architecture is based off of our three time champion DARwIn-OP robot's codebase but is much cleaner and faster now with some upgrades. This codebase was used successfully in 2014 to achieve first place in the AdultSize league with THOR-OP.

Team THORwIn would like to commit to participate in the RoboCup 2015 Humanoid League Competition – Adult Size League. We are also able to have members of our team volunteer as referees as they have sufficient knowledge of rules, even the updated ones for this year's competition at Hefei.

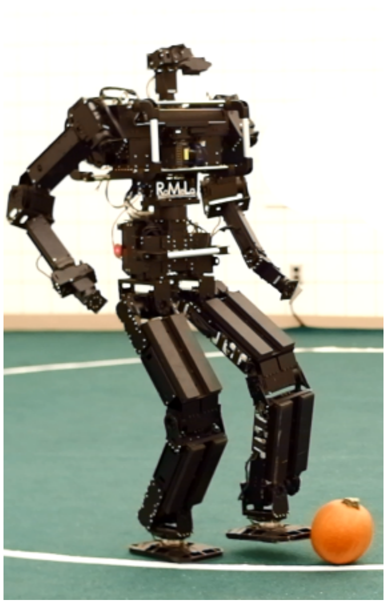

Fig. 1. THOR-OP practices its kicking maneuver

2 Research

Team THORwIn has ample research experience with successes in the humanoid robot arena, and more so with robot soccer. Our various collaborations over the past several years furthered humanoid research to very good levels not just at RoboCup but other projects from DARPA (Defense Advanced Research Projects Agency) and the ONR (Office of Naval Research).

Hybrid Walk Controller. We use a hybrid locomotion controller [3] which can dynamically switch between a standard ZMP preview controller using linear quadratic optimization and a reactive ZMP-based controller which uses closed solution of the linear inverted pendulum equation. The main benefit of this approach is that it provides a less computationally expensive, latency-free locomotion.

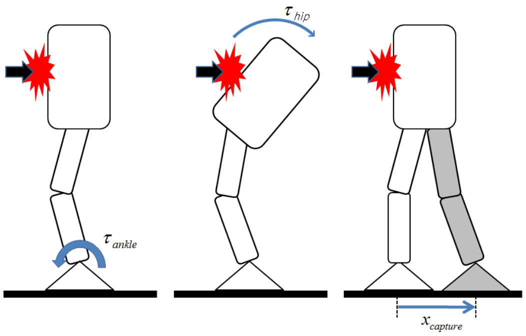

Fig. 2. Push recovery techniques help to avoid harmful effects of destabilization.

Push Recovery. To cope with unexpected perturbations, we have added two different types of push recovery control [4] that use the current state estimation from IMU and joint encoders. An ankle strategy is implemented by controlling the target joint angle of the position controlled ankle joint, which generates a control torque as a result, and the stop reflex strategy stops walking and lowers the center of mass to resist large perturbation. These strategies are shown in Figure 2.

Graphical User Interface. Command line scripts lack an ability to display rich sensor information, so we provide an HTML5 based graphical user interface (GUI). Browser support allowed for cross platform development and during the competition, we operated iOS, OSX, and Linux platforms simultaneously, with one master NodeJS [5] server to bridge raw UDP packets from the robot to WebSockets messages for the browser.

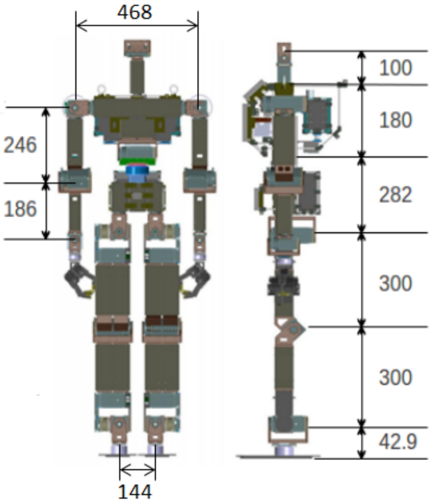

Fig. 3. THOR-OP Dimensions (in mm)

3 Hardware

The robot consists of 31 actuators, 7 in each pair of arms, 6 in each pair of legs, 2 in the torso, 2 for the head, and 1 in the chest. The robot stands 1.47m tall, weighs 49 kg and has a wingspan of approximately 1.95m. The arms and legs have shock absorbing padding to protect the robot in the event of a fall and there is a roll cage around the upper body both for protecting the sensors and computer and for ease of handling. Detailed link lengths are shown in Figure 3.

For actuators, the new series of Dynamixel PRO servomotors developed by Robotis, Co. Ltd are used at most joints. THOR-OP uses three different motor types: H42-20-S300-R, H54-100-S500-R and H54-200-S500-R. They are rated at 20W, 100W and 200W respectively and can be fitted with a number of different reduction gear boxes. For the THORwIn platform, two different gearboxes are used, one with in-line output axle and one with parallel output axle. Both gearboxes use cycloidal reduction gears that have higher impact tolerance than common harmonic drives. Communication with the servos is done via four RS485 buses.

4 Electronics and Power

The overall system structure and interfacing on THOR-OP includes one Intel NUC computer to perform high level computation. Communication from this computer to the actuators was divided into four independent chains of RS485s over one USB to RS485 interface board. The robot operates on a 24V input. The maximum current the robot has drawn at any time is 20A; for this reason, the two main computer reside on a separate supply to mitigate harmful current spike effects. With this separation, the required emergency stop switch kills only the servos when required, leaving the computer unaffected. The main sensors we use are a Logitech C920 HD head camera and MicroStrain 3DM-GX3-45 Inertial Measurement Unit. THOR-OP is capable of functioning completely under its own battery power. In fact, we use two 6-cell 11A LiPo batteries for the actuators and one 6 cell 3.25A LiPo battery for the computer. With these, the run time is close to 1 hour depending on the actions performed by the robot.

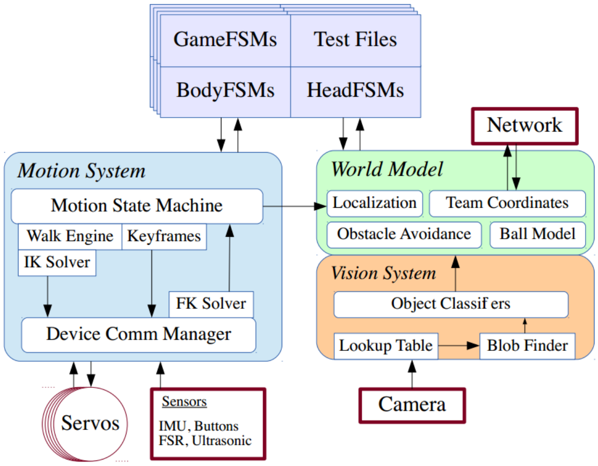

Fig. 4. The modular software architecture runs THOR-OP

5 Software Architecture

There are many advantages to modular software design. Each module performs logically discrete functionality, built separately from each other. When assembled together in proper hierarchy, they constitute the application program. This type of system is very reusable and extendable; in fact, we reused many modules from our own open-source code base for RoboCup [6] to save time and effort. An overview of the architecture is shown in Figure 4.

We modularized our software architecture in several ways. First, we set aside processes to interface with each hardware device, including cameras, laser scanners, motors, and human interface devices. Each process had a twin sub-process in our simulation environment driver, such that no code modifications were needed to run the robot in simulation or on real hardware. Furthermore, we made a module, with a clear application programming interface (API), that translated generic robot commands and queries into robot specific kinematics and motor packets, with the intention that a new robot platform needs only a similar module conforming to the API. Next, we constructed state machine modules that ran independently in a behavior process - ones for the overall body, the arms, the legs, the head, and the laser scanner.

Onboard the robot, communication channels included Boost shared memory segments [7] and ZeroMQ message queues [8]. While these interprocess channel assets would reside on one single computer, we leverage UDP messages to provide remote information at low bandwidth. All metadata is serialized via MessagePack [9].

Enhancements. For RoboCup 2015, the device communication module increases its usefulness by reading the current and positions of the motors at 100Hz, with writing at 100Hz. In previous years, we used selectively the readings from the servo motors, but this year hope to provide better stability by reading more often. Additionally, force torque sensors readings will be available from the ankle at the same sampling rate. This information will aid in touchdown detection and other dynamic feedback for walking and kicking.

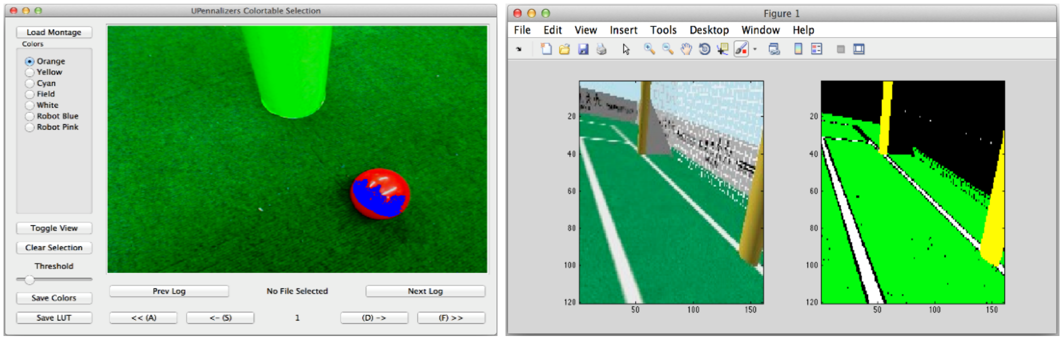

Fig. 6. Colortables can be made with a QT user interface for classifying ball colors (left). The generated lookup table can be monitored in MATLAB (right).

6 Vision

The eyes or eye of this adult size robot is a Logitech C920 HD camera. In our vision system, we use a look-up table to first categorize raw camera pixels into one of 8 color labels. The goal of the colortable generating utility is to provide a supervised learning approach to classify raw camera pixels into a set of 8 color class labels - possibly green for the field, orange for the ball, etc. Typically, we receive a YCbCr color formatted pixel array from the camera, and convert this to RGB for displaying to the user. Figure 5 shows the colortable generating tool for a real robot camera frame, and how the generated lookup table performs in simulation.

The user clicks on pixels that belong to each of these labels to generate both positive and negative examples of the color. We then apply a Gaussian mixture model to generate color segments. For each mixture, we perform a threshold to find pixels with a high probability of belonging to that color class mixture. With a high enough score, we add that pixel to a lookup table mapping to the color class. This lookup table from the mixture model to color class saves on per pixel computation, since the probability does not need to be computed for each pixel on every incoming frame; downsampling further increases speed. This labeled image is fed into high level blob detection routines and classifiers.

7 Locomotion Control

Overall, the locomotion of the robot can be controlled in three different modes. The *direct* mode, which uses the latency-free analytic ZMP controller, can control the velocity of the robot for every step. The next step position is calculated based on the current foot configuration, the target velocity and kinematic constraints. The *preview* mode uses the target pose of the robot to generate a number of optimal steps to reach the pose, and uses the preview controller for locomotion. Finally, the *special* mode is involved for special cases such as stepping over a block. For each mode, the start and end of locomotion is handled by a preview controller for a smooth transition between standing and walking.

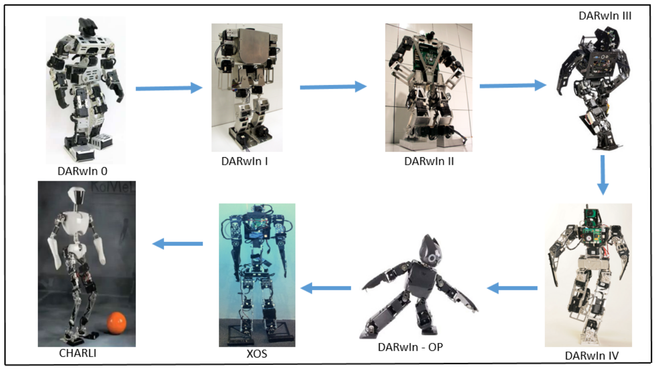

Fig. 7. The lineage of humanoid robots prior to THOR-OP is varied.

8 Prior Performance in RoboCup

The members of Team THORwIn have additional experience from Team DARwIn, Team DARwIn-XOS and Team CHARLI, with robots shown in Figure 7. Team THORwIn won the AdultSize championship in RoboCup 2015. In addition, Team DARwIn with the DARwIn-OPs (Dynamic Anthropomorphic Robot with Intelligence) won championships in the Kid-Size League for RoboCup 2011, 2012, and 2013. Team CHARLI is a two time Adult Size League champion for RoboCup 2011 and 2012 with CHARLI (Cognitive Humanoid Robot with Learning Intelligence) from Virginia Tech. This year, Team THORwIn is led by the University of Pennsylvania in collaboration with the University of California, Los Angeles (UCLA). With a history of using Dynamixel motors, Team THORwIn uses a slightly modified version of Robotis' THOR-MANG humanoid [10] that is THOR-OP.

9 Conclusion

Team THORwIn is excited at the possibility to compete in Hefei, China as part of the progress in humanoid robots that leads to the ultimate goal of RoboCup: for our humanoid robots to defeat human beings at the highest level of soccer. We are especially excited to continue building the community in the world's RoboCup participants, which is the backbone of the league.

References

1. D. Hong, Biologically inspired locomotion strategies: novel ground mobile robots at RoMeLa. *3rd International Conference on Ubiquitous Robots and Ambient Intelligence, Seoul, S.Korea, October 2006*
2. K. Muecke and D. Hong. Darwin's evolution: Development of a humanoid robot. *Proc. IEEE/RSJ Int. Conf. on Intelligent Robots and Systems, pages 2574 – 2575, October – November 2007.*
3. S.-J. Yi, D. Hong, and D. D. Lee, "A hybrid walk controller for resource-constrained humanoid robots," in *IEEE-RAS International Conference on Humanoid Robots, 2013.*
4. S.-J. Yi, B.-T. Zhang, D. Hong, and D. D. Lee, "Online learning of a full body push recovery controller for omnidirectional walking," in *IEEE-RAS International Conference on Humanoid Robots, 2011, pp.1–6.*
5. Node.js webserver. [Online]. Available: http://www.nodejs.org
6. S. G. McGill, J. Brindza, S.-J. Yi, and D. D. Lee, "Unified humanoid robotics software platform," in *The 5th Workshop on Humanoid Soccer Robots, 2010.*
7. Boost c++ libraries. [Online]. Available: http://www.boost.org
8. P. Hintjens. (2010) ZeroMQ: The Guide. [Online]. Available: http://zguide.zeromq.org/page:all
9. MessagePack serialization library. [Online]. Available: http://www.msgpack.org
10. THOR-MANG. "[RoboCup 2014] ROBOTIS disaster relief robot THORMANG wins the Adult League." http://en.robotis.com/BlueAD/board.php?bbs_id=news&mode=view&bbs_no=1140072