RoboPatriots: George Mason University 2008 RoboCup Team

Keith Sullivan, Brian Davidson, Christopher Vo,
Brian Hrolenok, and Sean Luke

Department of Computer Science, George Mason University
4400 University Drive MSN 4A5, Fairfax, VA 22030 USA
`{ksulliv2, bdavids1, cvo1, bhroleno, sean}@gmu.edu`

1 Introduction

The Autonomous Robotics Laboratory was established at George Mason University in 2006 with the goal of collaborative research in robotics, multiagent systems, computer vision, and sensor networks. While our previous work has focused on differential drive robots, RoboCup 2008 represents our first major effort with humanoid robots. The goal this year is modest: construct and program a working team for RoboCup. Our primary focus is developing a basic hardware and software platform to base future research initiatives on.

RoboPatriots has three robots: two attackers and one goalie. All three have the same hardware. The two attackers have one behavior mechanism, and the goalie has a different behavior mechanism. Presently, there is no communication between the robots, but that may change prior to the competition.

2 Hardware

We are primarily interested in software and embodied AI rather than in robot construction. For this reason, we have chosen the Kondo KHR-1HV as our robot base. Figure 1 shows the hardware architecture and information flow between components. Each robot has 20 degrees of freedom (DOF) controlled via servo motors. There are 6 DOF per leg, 3 DOF per arm, and 2 DOF in the neck. The sixteen KRS-788HV digital servos used in the arms and legs produce 10 kg-cm of torque at a speed of 0.14 sec. / 60 degrees. The two KRS-4024 servos in the shoulders produce 10.5 kg-cm of torque at a speed of 0.17 sec. / 60 degrees. These servos are controlled via the RCB-3 controller board. In addition, a KRG-3 single axis gyro and a RAS-3 dual axis accelerometer connect to the RCB-3.

The main processing is handled by a 600 MHz Verdex embedded computer attached to a Robostix microcontroller [1]. The Verdex communicates with the Robostix via I^2C and with the RCB-3 via serial at 115200 bps. A custom inverter board facilitates communication.

We replaced the stock head with a pan-tilt mount constructed from two standard micro-servos and a CMUCam3 [2]. The pan-tilt servos are controlled from the CMUCam3. The CMUCam3 communicates with the Verdex through

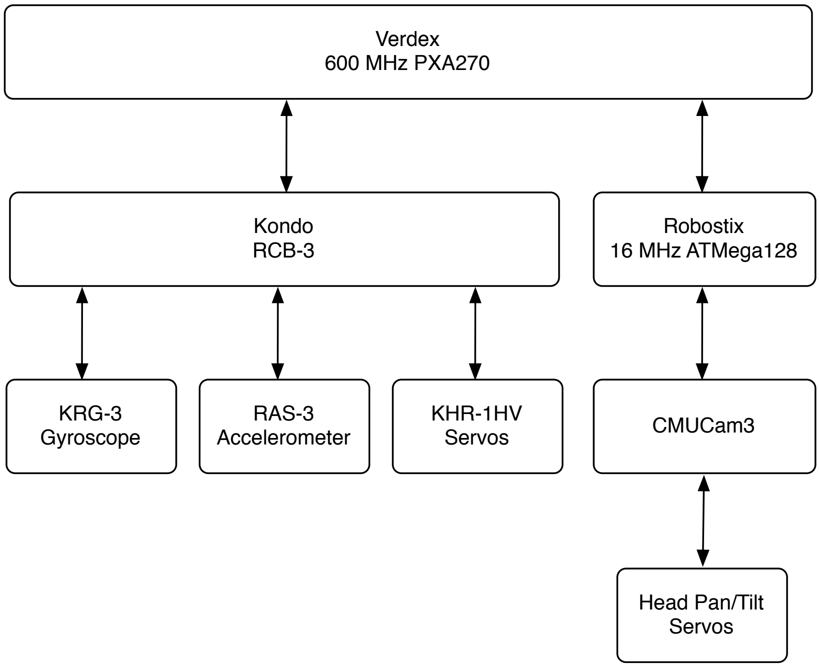

Fig. 1. Hardware architecture and information flow between components

a serial bridge on the Robostix: data travels via I^2C from the Verdex to the Robostix, and from the Robostix to the CMUCam3 via serial at 115200 bps.

The CMUCam3 is an embedded vision system based on the NXP LPC2106 60 MHz processor and an Omnivision CMOS sensor. The RGB CMOS sensor has CIF resolution (352x288), and operates at 26 frames per second. The CMUCam3 has an open-source API allowing customized software development.

The robots each have a 10.8 V, 800 mAh battery. A custom power regulator board ensures proper voltages to each component. At present there is no external communication (but this may change prior to the competition). The robots receive start and stop signals manually.

3 Software

Our robot architecture has two levels, each handled by a different microcontroller. At the lowest level, the RCB-3 itself handles getting up after a fall and gyro stabilization. Above it, we are constructing a simple software architecture which permits localization to ensure the robot remains on the playing field; tracking the ball; moving towards the ball; orientation for kicking; and kicking the ball towards the goal.

3.1 Architecture

We plan to base our software around a hierarchical finite state automata architecture we have developed, similar to [3] though developed independently. This architecture consists of a top-level finite-state automaton. Nodes in the FSA may take two forms: *atomic nodes* and *macro nodes*. Each edge from two nodes i and j is labeled with a *trigger*: a set of events which cause transition from i to j. The trigger may be "$*$", indicating that transition always occurs immediately regardless of the presence of an event. An automaton has a predefined start-node.

An *atomic node* represents a built-in behavior. When the automata has transitioned to an atomic node, the built-in behavior is repeatedly performed until transition out of the node occurs. One valid behavior may be to do nothing. A *macro node* represents a composed set of behaviors in the form of a *subsidiary finite-state automaton*. When this node is transitioned to, the top-level automaton is frozen and pushed onto a stack, and the subsidiary automaton is started at its start-node. Subsidiary automata may themselves have macro nodes, permitting arbitrary recursion among automata.

Exit from a subsidiary automaton may occur in one of two ways. First, if an event occurs which triggers a higher-level automaton (at any level) to exit its node via some transition, the subsidiary automaton is immediately quit and the higher-level automaton regains control. Second, the subsidiary automaton may define special exit nodes which quit the automaton and return control to its parent automaton.

3.2 Localization

RoboPatriots use an extended Kalman filter (EKF) approach for localization. Incorporating information about color tracking from the camera, neck servo position from the camera and a global map, the robot determines an approximate location. The dynamic nature of RoboCup and the imprecise nature of our robots makes highly precise localization unnecessary.

3.3 Motion

Using Kondo's Heart-to-Heart software, we programmed several basic moves into the RCB-3 on all three robots: walk forward, side step left, side step right, left and right kick, and stand-up from the front and back. In addition, the goalie has moves to drop into a blocking position and to swat the ball. The main behavior code then issues a command to the RCB-3 to execute one of these moves as appropriate. Other than gyro-based corrections, all the motions are hard coded, i.e., there is no ability to dynamically modify the motions.

Color tracking and neck servo information from the camera are used to orient the robot for kicking. Based on the camera servo position, the position of the ball's centroid, and the current robot orientation, the robot moves to position the ball in front of the appropriate foot.

4 Conclusion

RoboCup 2008 represents the first major research effort by the George Mason University Autonomous Robotics Laboratory. While our goals for this year are limited, we hope to develop a hardware and software platform for additional research in single and multi-robot systems.

References

1. (No Author): Gumstix, Inc. `www.gumstix.com` (2007)
2. Rowe, A., Goode, A.G., Goel, D., Nourbakhsh, I.: CMUcam3: an open programmable embedded vision sensor. See also `www.cmucam.org.` CMU-RI-TR-07-13, Robotics Institute, Carnegie Mellon University (2007)
3. Nicolescu, M., Matarić, M.: A hierarchical architecture for behavior-based robots. In: Proceedings of the First International Joint Conference on Autonomous Agents and Multi-Agent Systems. (2002)