A Responsive Vision System to Support Human-Robot Interaction

Bruce A. Maxwell, Brian M. Leighton, and Leah R. Perlmutter
Colby College
{*bmaxwell, bmleight, lrperlmu*}*@colby.edu*

Abstract

Humanoid robots are achieving mechanical capabilities that enable them to walk, run, and manipulate objects in their environment. To successfully interact with the world, they also need to be able to sense their environment. Human environments contain many visual cues, and visual feedback is a primary modality of human-human interaction, making it essential for human-humanoid robot interaction. Tasks for humanoid robots have several levels of complexity. Robot soccer is an example of a highly constrained task in an engineered visual environment. Tasks such as a tour guide require a more complex set of capabilities that focus on object recognition and identification of human characteristics such as faces and identities. The most complex tasks require fine manipulation of the environment or physical interaction with people. We present a vision system designed to meet the needs of tasks in the middle category. Simple games to motivate the visual sensing and interaction capabilities. The overall system is responsive to events in the environment and supports the required capabilities.

1. Introduction

The main research focus of humanoid robots to date has been the development of the mechanical and feedback control systems required for them to execute basic motions. This focus has led to significant advances in humanoid robot systems at all scales, and humanoid robot research platforms are becoming more widely available. The humanoid soccer league is one example of a growing community developing research grade humanoid robots.

In order for humanoid robots to function in human environments, they must have sensing mechanisms that enable them to respond to their environment. Some environments and some tasks permit these sensing mechanisms to be built into the environment, such as RFID tags or active localization systems. However, most human environments–which is where humanoid robots are most appropriately used–are engineered for people.

While human environments make use of multiple sensing modalities, particularly sound, the primary modality for sensing most human environments is vision. Signs provide labels or directions for navigation; visual gestures add context and clarification to conversations; and object detection and recognition permits us to identify and interact with individual items in the environment. In order to function in human environments, humanoid robots must have visual sensing appropriate for their tasks.

The specific visual sensing capabilities required by a humanoid robot will depend largely upon the role it is asked to play. Some robots require only basic visual sensing in engineered environments, such as robot soccer. At the other end of the spectrum, an in-home robotic assistant would require the ability to identify human identity, pose, and possibly emotions as well as detect and recognize most of the individual items in a house. The field of computer vision is making progress in all of these areas, but a general purpose vision system is not yet realistic. Developing a realistic vision system requires first identifying realistic tasks.

We separate humanoid robot roles into three categories, depending upon the type of sensing required. The first category consists of roles that require the robot to function in an engineered environment without direct human-robot interaction. The current humanoid robot soccer league is one example. During a soccer match, robots localize themselves and identify key game elements by locating specially colored landmarks or color material transitions. The use of well separated, saturated colors reduces the complexity of the visual sensing required to execute the task.

The second category consists of social roles where the robot is not interacting physically with a person and interactions with the environment are carefully prescribed and predefined. These roles require the ability to detect and identify people, detect relevant objects for the task, and possess basic localization and navigation skills in non-engineered environments. An example of such a task is playing Simon Says with a group of children, other simple games that involve taking turns, or acting as a tour guide in a museum.

The third category consists of tasks that involve both social roles and physical interaction with a person or the environment. The key differences between the second and third

categories are the need for more exact proprioception by the robot relative to the environment and the lack of significant structure to the interactions. Examples of such tasks include dishwashing, cooking, playing soccer against people, or assisting on a job site.

The focus of our work is on the second category of tasks. We have selected two games–a table-top game with blocks and Simon Says–to provide context for the development of a vision system appropriate for these humanoid robot roles.

2. Related Work

Many researchers have developed robot vision systems. Most of them have been single-purpose systems designed to meet the needs of a specific task. Some, however, have evolved into more general purpose vision systems that can be tailored to specific tasks more easily than building a new system.

One of the most common vision systems for category one tasks–engineered environments–is the CMUcam system. The first and second CMUcam systems were designed primarily as color blob or shape trackers and are heavily used in robot soccer tasks [7]. The most recent CMUcam3 system contains an ARM processor and supports a much wider variety of algorithms [6]. It is not a humanoid vision system, per se, but may provide the necessary components and hardware for building one.

A significant resource for building any computer vision system is the OpenCV software library, which implements many capabilities including: face detection, object recognition, feature calculations, and many other standard computer vision algorithms [1]. OpenCV is, like the CMUcam, a potentially significant piece of a vision system and provides optimized versions of many useful algorithms. Other than faces, however, users must build their own recognition systems with their own data using the algorithms provided.

One example of an actual robot vision system built for an object recognition task is the Curious George vision system built for the Semantic Robot Vision Challenge in 2007 [3]. The system was designed to learn the appearance of a set of objects using the World Wide Web and then recognize the objects in its environment.

An example of a vision system built for social robots is described in [4]. The system was designed to be able to run many different operators simultaneously with sufficient speed for social interactions. The system is flexible enough to permit one operator to track an object and control the camera orientation while allowing other operators to examine the image for objects, faces and other information.

Commercial vision systems are also available that provide more complete systems. Evolution Robotics provides a system that integrates visual navigation and object recognition, two essential tasks for humanoid robots [2]. Skilligent, Inc also provides an object recognition, tracking, and localization system [10].

Herein we describe a vision and decision-making system based on the vision module of Maxwell et. al [4]. It uses OpenCV to provide many of the basic vision algorithms and the Inter-Process Communication system developed by Simmons for communication between applications [9]. The system currently supports a suite of operators that provide information about the environment. Operators include face detection, color blob detection, text detection and simple OCR, motion detection, and a robot tracker. The system has a straightforward mechanism for adding the capability to detect specific objects using the OpenCV library.

The system permits any one operator to be used to track and control the pan-tilt orientation of the camera, and the decision-making application can turn operators on and off as necessary and weight their importance. The vision system runs a fixed number of operators on each frame to guarantee a high frame rate. Operators are selected stochastically, with higher weighted operators running more often. As social interactions are relatively slow, compared to camera frame rates, most operators do not need to run on every frame. Overall, the vision system provides a large suite of operators that are responsive to the environment in times appropriate for both social interaction and tracking tasks.

3. Experimental Setup

The experimental setup uses a Robonova platform, a 25cm tall humanoid robot with 16 degrees of freedom. The robot has an onboard microcontroller with a Basic interpreter that can execute simple programs. We have added a BlueSmirf Blue Tooth serial adapter that allows for data and commands to be sent to the robot from a host computer, a standard workstation running Linux. The Robonova provides sufficient complexity that we can model many of the actions we would expect a full-size humanoid robot to execute. With the Blue Tooth adapter we avoid the need for a tether while still enabling significant processing power for the perception and interaction systems.

Visual feedback for the robot is provided by a VC-C4 Canon PTZ camera placed 1m above the robot's work area. The work area is approximately 0.5m x 0.5m. The camera is attached to a host computer running a vision system that can detect the robot and objects in its work area. The host computer is also executing our interaction and reasoning system and building plans based on the world state detected by the vision system. The system comprises a complete feedback loop so the robot's actions are reflected in changes in the perceived world state. A diagram of the system is given in figure 1.

The robonova's workspace is a 50cm x 100cm rectangle with 12cm walls as shown in figure 2. A two-tiered rack above the workspace includes mounts for a downward facing camera to view the robot, as seen in figure 3, and a

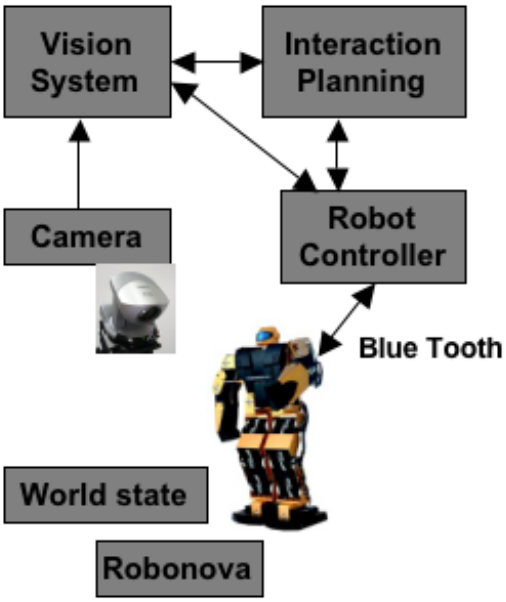


Figure 1. Diagram of the robot system and communication paths between modules.



Figure 2. Robot's workspace with the robot and the two cameras.

second camera at head-height to view someone interacting with the robot, as shown in figure 4. The entire setup sits on a table and provides a self-contained demonstration area where a person can easily interact with the robot and objects within its workspace.

The ultimate goal of this work is to move the vision system to two other platforms: HUBO and mini-HUBO. HUBO is an approximately 4' tall humanoid robot developed by KAIST, Korea. We will be working with Drexel University, which has a duplicate of HUBO, Jaemi HUBO. In addition, we will be working with Virginia Tech, which has developed a 17" tall humanoid robot with similar behavior to the full size HUBO. Our goal is to implement the vision system on both of these systems in the future.



Figure 3. Vision system detecting and tracking the robot.



Figure 4. Vision system detecting and tracking a face.

4. Games for Interaction Development

The motivation for developing vision capabilities is the task the humanoid robot must accomplish. Many games, particularly those played by children, fit within the category two set of tasks. The number of relevant objects in the environment tends to be small, the degree of physical interaction is minimal, and the interaction is circumscribed by the rules of the game. The one caveat is that even simple games have many implicit rules that must be built into the robot's programs in order for the robot to interact properly [8].

We are using two games to motivate the development of the vision system and to help us develop an abstraction of humanoid robot movement that enables connecting dialog and social interaction decision-making with physical actions. The first game is Simon Says, a children's game that requires the robot to move and detect motions in others. The second game is a tabletop game with blocks that requires the two players to propel one block between two others.

The rules for Simon Says are very simple. One actor plays the role of Simon and everyone else is a participant. The participants listen to Simon's instructions and take the appropriate action. If the actor playing Simon says to take an action, like waving your arms, and begins the description of the action with the words 'Simon Says', then the participants must execute the action immediately. On the

other hand, if the actor playing Simon does not begin the description with 'Simon Says', then the participants should not execute the action. Participants who improperly move, or are improperly still, are out of the game. The last participant left in the game is the winner. The actor playing Simon is responsible for identifying those who do not follow the instructions properly.

Simon Says does not require physical interaction with the robot, but does require the robot to sense appropriate motion in the participants. The robot must be able to detect motion, detect the location of the motion relative to a landmark on the participant's body, such as their face, and be able to identify which participants are out. The game permits the robot to exhibit a wide range of motions. It does not require the robot to plan extensively or make complex decisions, and the dialog is limited. If the robot does not have speech recognition, then the robot is limited to the role of Simon.

The table-top blocks game is a simple game played with two players, three blocks, and a stick or paddle for the person. Player one places three blocks on the table, one of which is the active block. Player two must attempt to send the active block in between the other two blocks. The robot kicks the active block, the person uses the stick or paddle to propel the block. Player two gets a point if the ball block goes between the other two blocks. Then the players switch roles. As an example of an implicit rule in the game, neither player should take too long to make their attempt.

4.1. Defining Capabilities

The two games require different types of vision capabilities, but are indicative of a range of category two tasks. Simon Says requires analysis of people, in particular groups of people. The following capabilities are required in order to play the game.

- Ability to identify the location of each participant, probably by detecting a face.
- Ability to identify if the person is moving significantly when they should be still, or still when they should be moving.
- Ability to identify a pointing direction to specify if a participant is out.

As shown in figure 4, the vision system can detect and track faces. Currently, the system can track up to eight faces, which is enough for prototype demonstrations. Figure 7 shows the system identifying boxes of motion and their extent. The vision system is also calibrated and can provide a 3D ray in space for each pixel in the image, providing sufficient information for the robot to point to a participant.

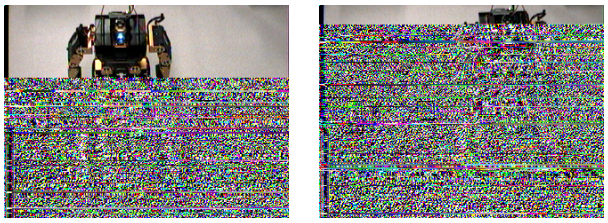


Figure 5. Vision system detecting the colored blocks in the robot environment.

In order to play the game in a more sophisticated manner, the robot would need additional capabilities. These are currently under development.

- Ability to identify the specific type of motion executed by each participant.
- Ability to recognize individuals and track them if they change position.
- Ability to determine if someone who is out is participating inappropriately.

The table top blocks game requires a different set of capabilities geared towards object recognition. The robot system must be able to do the following. As shown in figures 3 and 5, the system can track both the robot and the blocks.

- Ability to identify and locate the robot.
- Ability to identify and locate the game items such as the blocks and paddle.
- Ability to identify the location of other unknown objects in the game area such as the other player's body parts.

For the robot to play the game in a more sophisticated manner, such as knowing when a person is engaging it to play, the robot system needs additional capabilities. The first capability is enabled through face detection, the latter two are under development.

- Ability to identify that a person is in the appropriate location to play.
- Ability to identify the paddle and detect when it is in a person's possession.
- Ability to identify the blocks and detect when they are in a person's possession.

The latter capabilities would enable the robot to engage in interactions beyond the necessary physical actions required to play the game. It would also enable the robot to know when to begin a game and what actions the person is currently undertaking.

4.2. Defining Actions

In addition to defining visual capabilities, we are also attempting to develop an abstraction for the humanoid robot actions. Figure 6 shows a number of examples of gestures the robot may make during the course of a game. At the lowest level of abstraction, these gestures require joint angles. From the point of view of the decision engine, however, that level of abstraction is too detailed. We hope to use the same vision and interaction system on a number of different humanoid robots. As each robot has different hardware and different numbers of joints, however, we need a common language of motion.

Gestures are also more than just a set of joint angles. Gestures can have levels of intensity, and we want to be able to combine gestures, at different strengths, to achieve certain effects. The same gesture executed with different intensity can have significantly different semantic meaning in an interaction situation. A deep bow, for example, can have a significantly different meaning that a short bow, depending upon the context.

The facial animation field went through a similar process early in its development, with researchers using anthropological taxonomies to motivate a layer of abstraction that connected semantic expressions with the motion of vertices in the facial model [5]. As our two games motivate the visual sensing capabilities, so they also motivate the action capabilities and provide a specific set of motions required to complete the tasks. Simon Says is an especially good example because it incorporates the full range of motions of the humanoid robot. In the table-top blocks game the robot needs only a good walking and kicking engine.

We are currently developing a framework for identifying the vocabulary of humanoid robot actions.

- Identify individual poses or actions that fit within the task.
- Identify poses or actions that have similar semantic meanings.
- Generate a hierarchy of actions, where the actions at a lower level of the hierarchy are parameterized versions of the higher level.

The above process will generate a tree of gestures. Each level of the tree represents one level of abstraction and subdivision. The bottom level is a set of single fully defined poses or gestures. The level above the leaves will have some parameterization of the action. The topmost level will represent a large set of parameters such that by setting them appropriately the robot can achieve any single leaf node. The goal is to identify the level that balances the number of gestures with the number of parameters required for each gesture.

We will develop the abstraction layer in cooperation with our other partners at Drexel University, the University of Pennsylvania, Bryn Mawr College, and Virginia Tech who will be developing the low level systems for the HUBO, min-HUBO, and simulated HUBO robots. The combination of their low level control systems and our vision and interaction system will be an autonomous humanoid robot capable of complex tasks in human environments.

5. Summary

We have a testbed and basic infrastructure for developing and evaluating a vision and interaction system for humanoid robots. We are using simple games to motivate the development of useful capabilities for robot workspace manipulation and human-robot interaction. The vision system is built upon a software infrastructure that permits easy development of new operators and integration with other modules for decision-making and low level control.

We are also using the physical actions required by the games to guide the development of an abstraction layer for describing humanoid robot actions. In cooperation with our partners, we hope to integrate a complete humanoid robot system that is capable of autonomous interaction in human environments.

References

[1] G. Bradski and A. Kaehler. *Learning OpenCV: Computer Vision with the OpenCV Library*. O'Reilly, 2008. 2

[2] EvolutionRobotics. http://www.evolution.com/, 2009. 2

[3] S. Helmer, D. Meger, P.-E. Forssén, S. McCann, T. Southey, M. Baumann, K. Lai, B. Dow, J. J. Little, and D. G. Lowe. Curious george: The ubc semantic robot vision system. Technical Report AAAI-WS-08-XX, AAAI Technical Report Series, October 2007. 2

[4] B. A. Maxwell, N. Fairfield, N. Johnson, P. Malla, P. Dickson, S. Kim, S. Wojtkowski, and T. Stepleton. A real-time vision module for interactive perceptual agents. *Machine Vision and Applications*, 14:72–82, 2003. 2

[5] S. Platt and N. Badler. Animating facial expression. *Computer Graphics*, 15(3):245–252, 1981. 5

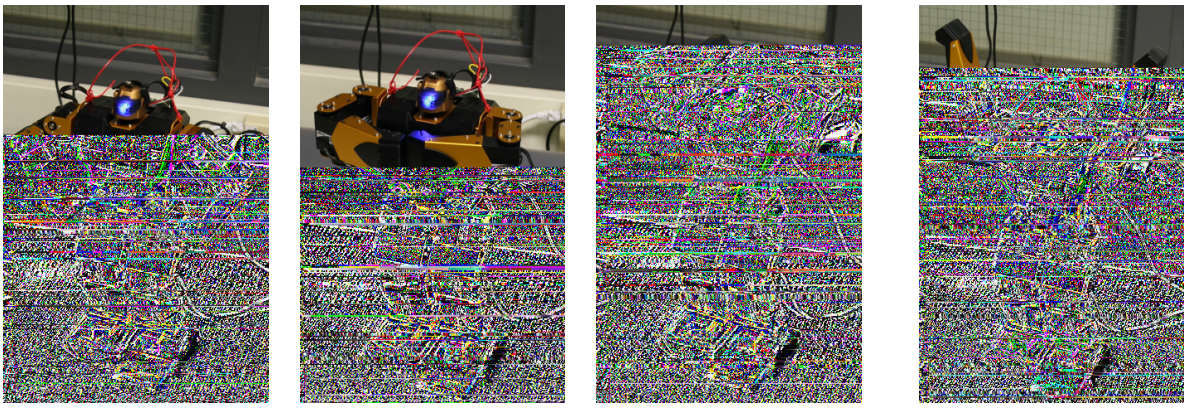
[6] A. Rowe, A. Goode, D. Goel, and I. Nourbakhsh. Cmucam3: An open programmable embedded vision sensor. Technical Report RI-TR-07-13, Carnegie Mellon Robotics Institute, May 2007. 2

[7] A. Rowe, C. Rosenberg, and I. Nourbakhsh. A second generation low cost embedded color vision system. In *Embedded Computer Vision Workshop*. IEEE, 2005. 2

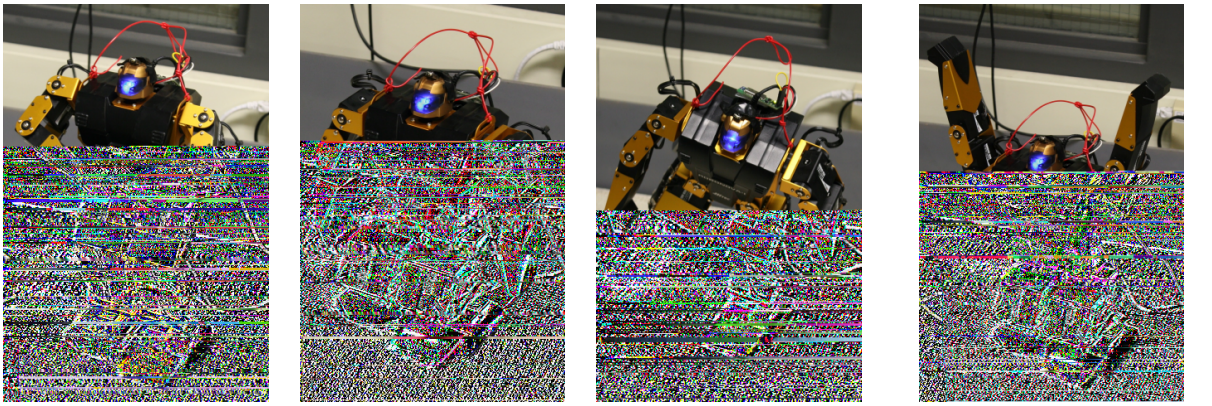
[8] K. Salen and E. Zimmerman. *Rules of Play: Game Design Fundamentals*. MIT Press, 2003. 3

[9] R. Simmons and D. James. *Inter-Process Communication: A Reference Manual*. Carnegie Mellon University, March 2001. 2

[10] Skilligent. http://www.skilligent.com/, 2009. 2

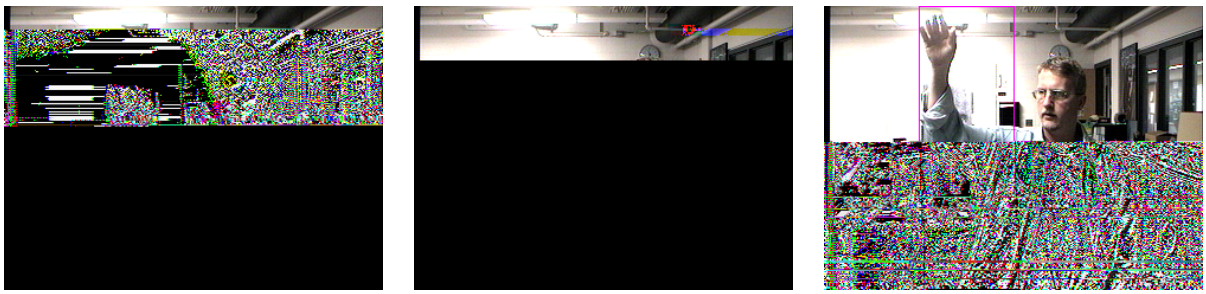


(a) Hands on hips (b) Hands on chest (c) Hands on head (d) Hands in air



(e) Hands on stomach (b) Sit down (c) Stand on one leg (d) Sit down with hands in air

Figure 6. Robonova demonstrating various positions for Simon Says.



(a) Arm low (b) Arm middle (c) Arm high

Figure 7. Vision system recognizing different kinds of motion. Note the pink box delineating the motion area.