

SENSING AND MANIPULATING BUILT-FOR-HUMAN ENVIRONMENTS

Rodney Brooks, Lijin Aryananda, Aaron Edsinger, Paul Fitzpatrick, Charles Kemp, Una-May O'Reilly, Eduardo Torres-Jara, Paulina Varshavskaya, and Jeff Weber

*MIT Computer Science and Artificial Intelligence Laboratory
Cambridge, Massachusetts, 02139, USA*

{brooks, lijin, edsinger, paulfitz, cckemp, unamay, etorresj, paulina, jaweber}@csail.mit.edu

Received 8 November 2003

Revised 9 January 2004

We report on a dynamically balancing robot with a dexterous arm designed to operate in built-for-human environments. Our initial target task has been for the robot to navigate, identify doors, open them, and proceed through them.

Keywords: mobile manipulation, dynamic balance.

1. Introduction

The past decade has seen an explosion of research in humanoid robotics. The stated motivations for this work have varied widely. Many teams have concentrated on bipedal locomotion,^{1,2,3} some have been interested in human level social interactions,^{4,5,6} understanding human intelligence,^{7,8,9} modeling human learning capabilities^{4,10} and others have been more interested in entertainment.^{2,11} Some humanoid robots have had manipulation capabilities on static humanoid platforms¹² and some of that work is aimed at dexterity,¹³ plus there has been simple two armed grasping on mobile humanoid platforms.^{14,15} Overall there has been very little work combining dexterous manipulation with humanoid robots, static or mobile—much of that which has appeared, has been concerned with dynamic tasks like pole balancing and juggling¹⁶ rather than manipulation, or has used teleoperated manipulation.²

One of the usually unstated motivations for building humanoid robots is that ultimately we may want robots with human form to operate alongside people in human-centric environments.

The environment in which modern humans live has been built to accommodate the structure, and the physical and cognitive capabilities of humans. If we want robots to take over many of our daily tasks they need to be able to physically operate in these environments. This means that they should have a small footprint, be able to see the environment from a human viewpoint, and be able to dexterously manipulate objects at human waist level and above.

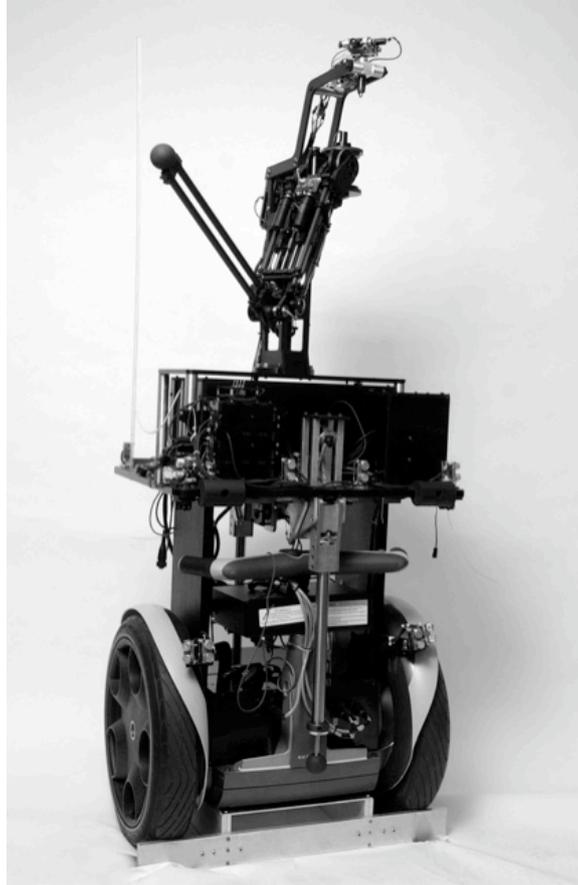


Fig. 1. The robot, named CARDEA, is built on a Segway Robotic Mobility Platform (RMP). The handlebars and their electronics have been moved down to just above the wheels. The box on the platform where a person would normally stand contains recharging electronics. Strapped to the top of that box is a laptop PC running Linux which handles all vision processing. A heavy plate that is part of the RMP supports a battery and electronics boxes, and on top of that is the force controlled arm. The robot has ten sonars, two whiskers and two cameras. The center front camera points slightly downwards and is used to navigate along corridors. A second camera mounted above the shoulder of the arm had a pan mechanism and was used to find doors in our demonstration. The mount shown has been upgraded to allow both pan and tilt.

We have built a prototype robot, named CARDEA, shown in Figure 1, to investigate three technical challenges that these requirements present:

- (1) locomotion with a large height to base-size ratio,
- (2) manipulation of unknown objects,
- (3) sensing a built-for-human environment well enough to manipulate it.

The first target task we have chosen for the robot is for it to navigate through

corridors and rooms of an office environment, to find and open doors, and to go through them.

The robot dynamically balances and has a force sensitive and force controlled arm. The arm design incorporates five degrees of freedom but at the time of writing the arm operates with only three. We are currently upgrading to six degrees of freedom plus a hand. With the current configuration CARDEA can push open a slightly ajar door and go through the door. It uses a vision system that is specialized for navigating in corridors and finding doors, but the principles of those vision tasks can be generalized to other behaviors.

The robot is behavior-based^{17,18,19,20,21} and its high-level behaviors emerge from the interactions of many low-level sensorimotor loops. Throughout this paper we refer to the behavior of the robot in terms of things like *following a corridor*, or *approaching a door*. These are convenient terms for describing the behavior of the robot but they often do not correspond to explicit states in the programs that control the robot.

This paper reports on the results of our work with CARDEA over a four month period in the middle of 2003, from initial design through to a completed demonstration. The paper is organized as follows: we start with a case for dynamic balance. That is followed by descriptions of the robot platform, its computational hardware architecture and the software systems responsible for vision, manipulation and navigation. The next section presents a visual sequence of Cardea performing a corridor following, door shoving and doorway passage demonstration. We end with discussion of possible future directions.

2. The Case for Dynamic Balance

Walking humanoid robots that match the degrees of freedom and structure of a typical human have the potential to be perfectly suited to built-for-human environments. Unfortunately, these highly complex systems are currently very costly and unlikely to be feasible for most applications until many years from now. In the near term, wheeled dynamically stable robots can provide many of the same benefits of a walking humanoid, but at a lower level of cost and complexity. General wheeled dynamically stable robots offer distinct advantages over traditional statically stable platforms, as well as statically stable platforms with articulated torsos that allow for sophisticated redistribution of weight.

General operation within built-for-human environments places special requirements on the capabilities of a robot. Both navigation and manipulation benefit from a servoed visual system located at around the height of the human head. For navigation, this configuration allows perception of free space, obstacles, and areas that require special locomotion. For manipulation, this facilitates the observation of surfaces, workspaces, and mechanisms of interest. Both navigation and manipulation also benefit from a body with a small footprint of human size or less, since this allows the robot to locomote through crowded spaces and to bring the body close

to human workspaces, such as tables. For manipulation, the robot should be able to reach out into the world, so that it can grab objects and interact with common items, such as doorknobs and cabinets. Practical reaching will often require reaching up high and reaching over surfaces, such as counters, tables, and shelves. Finally, built-for-human environments require stable robotic platforms that can deal with unexpected perturbations.

Wheeled dynamically stable robots address these requirements with four major design benefits. First, wheeled dynamically stable robots allow for the weight of the robot to be relatively high above the ground. Like humans, this type of robot functions in a way similar to an inverted pendulum, and consequently can handle weight relatively high above the point of ground contact. This is unlike traditional statically stable robots, which must be careful to keep their center of mass low to the ground in order to avoid tipping over. The ability to safely put weight up high is important because it means that the torso, head, and arms can be substantial and placed high above the ground in order to meet the sensing and manipulation requirements mentioned above. Second, small footprints are possible with wheeled balancing robots. Unlike a statically stable system, the base does not need to be large in order to be stable, since, like a human, it only needs one or two points of ground contact. Additionally, wheeled dynamically stable robots can have larger wheels while maintaining a small base platform, which allows them to better move across cluttered and rough terrain. Third, the axle near the floor can serve as another degree of freedom for tilting the body. Humans often tilt their body at the ankles in order to tilt the lower body and redistribute their weight for reaching out into the world. Wheeled dynamically stable robots get a similar degree of freedom without the cost or complexity of introducing another axis of tilt above the base. They can also tilt in order to traverse inclines. Fourth, wheeled dynamically stable robots are resistant to the unavoidable unexpected perturbations of day to day activity. The effective volume over which the center of mass can move is larger than with a statically stable system, because the dynamic system can catch itself.

3. The Robot Platform

3.1. *Mobility*

Rather than build a dynamic wheeled platform from scratch we chose one that is commercially available. We used a specially modified Segway HT platform, known as the *Segway Robot Mobility Platform* or *Segway RMP*.²² It is a little larger than we would have preferred—the wheel radius is 23.5cm so the front to back depth of the robot is more than 50cm—larger than a person. However these large wheels let us use pneumatic tires which make for a very smooth ride. Currently, the total height of the robot is roughly 165cm and the weight is about 70kg.

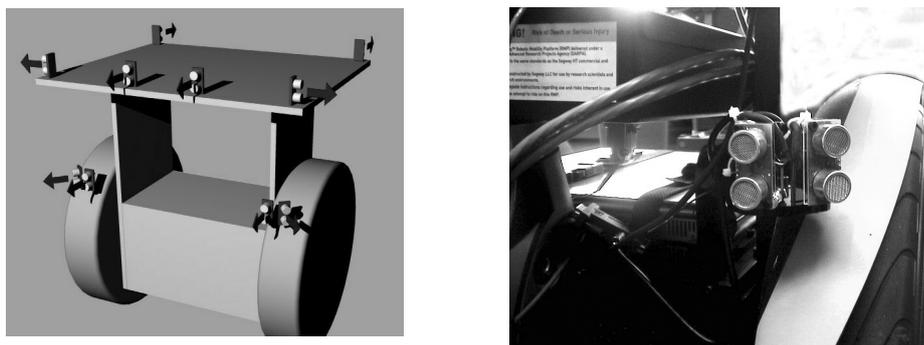


Fig. 2. Ten sonars are currently mounted on CARDEA. Four sonars are facing forward and placed at different heights. Two sonars are configured at a 45 degree angle. Two are pointed to each left and right side. The last two sonars are facing the back.

3.2. Sensors

CARDEA uses two main sensory media: ultrasound and vision. The ultrasound sensors are off-the-shelf sonars (Devantech SRF04), which operate at 40kHz and cover a range of 3cm to 3m. Their main cone angle is approximately 40 degrees (measured from the -3db point on one side of the axis to the -3dB point on the other side of the axis).²³ They were selected for their small minimum operating distance in order to allow the robot to safely interact within close distances to objects and people. Ten sonars are currently mounted in a configuration shown in Figure 2.

Visual input is received from two color digital cameras each with an IEEE-1394 (FireWire) interface, which were chosen for their superior image quality. They produce 640×480 24bit color images at the rate of 30 frames per second. Their placement is shown in Figure 3.

Supplementary information is obtained from a pair of on/off whiskers mounted on the front of the robot. Each whisker is simply a piece of flexible wire, attached to a contact switch. In support of these sensors various custom electronic boards and off-the-shelf devices such as the FireWire hub and the remote-control receiver are placed inside a number of boxes that are affixed on the RMP base top plate.

3.3. Manipulator: The Force Controlled Arm

The base is ultimately a mobile platform for a manipulator. Control of humanoid manipulators is a very active area of investigation, with notable work in dynam-

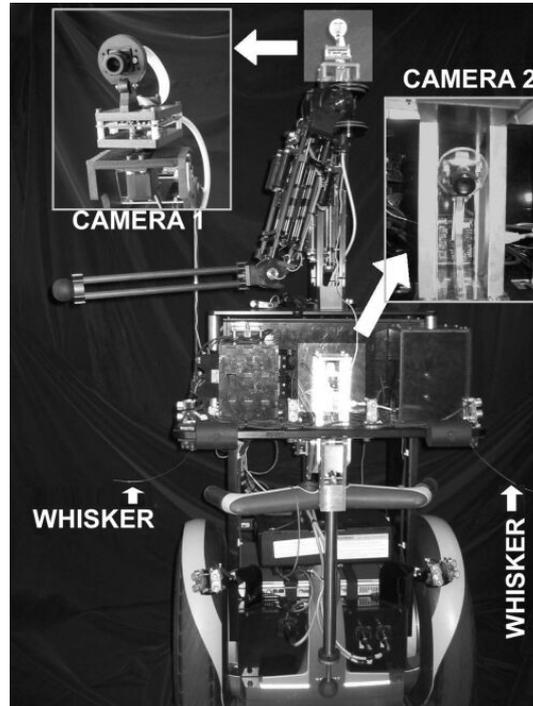


Fig. 3. Various actuators, sensors, and controller modules have been added onto the robot. The whiskers are placed toward the front side of the base.

ical representations for control²⁴ and in learning manipulator grasping policies.²⁵ CARDEA's first manipulating task was to push open a door. To this end, we designed a 5 degree-of-freedom (DOF) force controlled arm, pictured in Figure 4. This first arm serves as a prototype platform to explore the design space of dexterous humanoid arms and develop manipulator control strategies which support a behavior based approach.

3.3.1. Dexterous arm design

A central pillar of our design approach is that the manipulator must be passively and actively compliant, and it must be able to directly sense and command torques at each joint. Other elements of dexterous manipulator design meriting discussion include the mechatronic specification, modularity, mass distribution, and power consumption.

We chose a cable drive based design similar to that of the WAM arm.²⁶ The arm incorporates a 2 DOF pitch and yaw shoulder, a one DOF elbow, and a 2 DOF pitch and roll wrist. As pictured in Figure 1, only the shoulder and elbow are

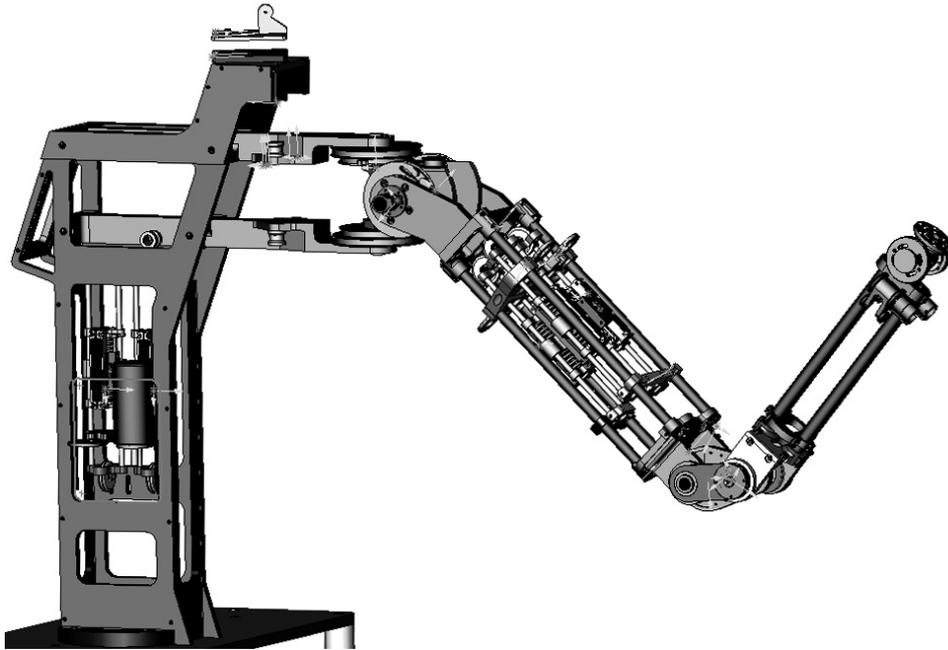


Fig. 4. The force controlled arm. Two SEA actuators in the torso drive the differential shoulder in pitch and yaw. Three SEA actuators in the upper arm drive the elbow and the differential wrist in pitch and roll. The elbow is designed with a virtual center so that drive cables routed to the wrist can pass through the exact center of rotation.

present. Each joint is driven by a Series Elastic Actuator (SEA).²⁷ We will limit our description of SEA actuators here as they are fully described in detail elsewhere.²⁷ In short, by placing a linear spring in series with the motor output, we gain a two-fold advantage. First, high impact shocks are low-pass filtered by the spring, protecting the geartrain, and, second, the output force is easily sensed by measuring the spring displacement. Each joint contains a brushless DC motor. Custom brushless motor amplifiers and sensory signal amplifiers are embedded throughout the arm. The physical distribution of the control electronics minimizes wiring run-length and simplifies cable routing. A custom DSP based controller is also embedded in the arm. It handles the high bandwidth control of up to 8 DOF in a 1Khz servo loop. A STACK running CREAL (see Section 5.2.1) interfaces directly with one or more of the DSP controllers.

CARDEA may eventually have two (or even three) kinematically asymmetric arms. A humanoid with asymmetric arms may gain a dexterity advantage not available to human arms. To investigate this potential advantage, we have pursued a design for the arm that lends itself to reconfiguration in terms of the link lengths and the

number of DOF. Additionally, we have pursued mechanically modular joints and actuators that allow a design economy through reuse of basic components. While modularity often comes at the expense of mechanical compactness, it allows us to easily try different arm morphologies. The shoulder and the wrist are based on a compact differential design. Aside from its modularity, the differential design is advantageous for the following reasons: it is kinematically simple because the axes of rotation intersect; the output torque in the pitch direction (against gravity) can effectively be greater than that of a single actuator due to the coupled nature of the joint; and it allows for cable routing through the center of the joint. Arm modularity is also gained by employing a cable drive system in conjunction with a carbon-fiber tube frame for each arm link. The length of each link can be easily modified by adapting the length of the tubes and cables. This allows for reconfigurability of the arm morphology without additional design or fabrication. Finally, the SEA actuators have been designed as modular units that can be easily mounted and positioned on the carbon-fiber frame.

Poor mass distribution in a manipulator will adversely impact its dynamics and consequently degrade the manipulator agility. It will also increase power consumption, limiting the mobility of a battery powered humanoid. Ideally the actuator mass is kept as close to the shoulder and torso as possible. Our design experience with the humanoid Cog²⁸ has shown that by placing many of the actuators distally along the arm, the actuators at the base of the kinematic chain tend to be overworked and prone to heat damage. A cable-drive system allows the mass of the actuators to be moved back onto the torso and to the upper-arm. CARDEA's two wrist actuators are mounted just below the shoulder and their drive cables are routed through the center of the elbow. This improves power consumption and agility while also lowering the torque requirement for the actuators and allowing greater physical dexterity in the joint ranges.

Power consumption is also tied to the backdrivability of the arm. High backdrivability is often desirable in humanoid arms. Unfortunately, it has an inverse relationship with the ability to hold a static posture against gravity with low power consumption. A manipulator with low backdrivability can hold itself against gravity indefinitely but is prone to geartrain damage and is cumbersome for human interaction. Our design is statically non-backdriveable. However, the SEA actuators employed allow it to be actively backdriveable and tolerant of impacts to the geartrain. Consequently, it can hold static postures with only a few watts of power consumption.

3.3.2. *Virtual spring control*

Spring-like compliance is important for behavior based manipulation. It allows for loosely controlled exploratory poking, pushing, and hitting of the environment. There is no need to directly model the environment or explicitly plan manipulator actions before manipulating. Instead, the manipulator can actively sense the

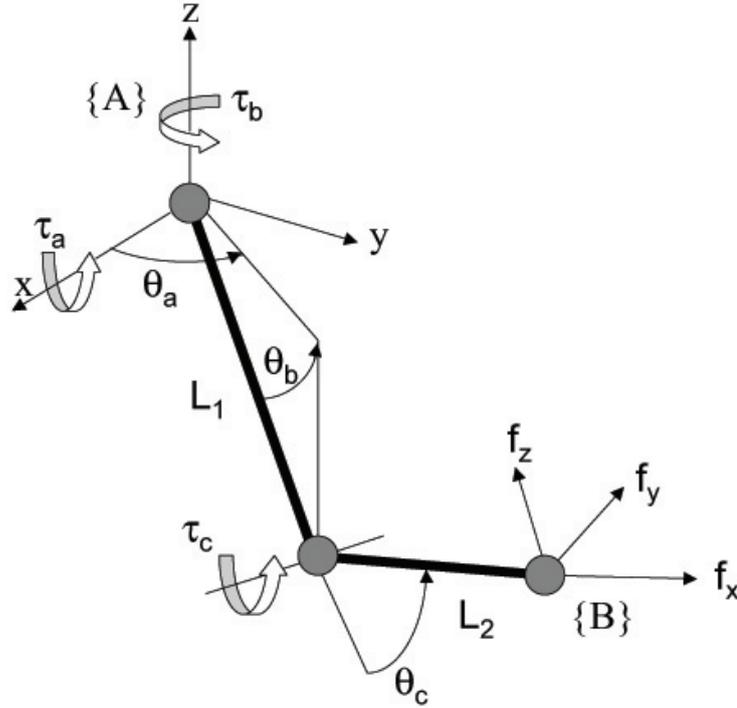


Fig. 5. The kinematics of the first three DOF of the arm. The shoulder yaw angle is θ_a , the shoulder pitch angle is θ_b , and the elbow pitch angle is θ_c . The upper arm length is 35cm and the forearm length is 38cm. The forward kinematics ${}^A_B \vec{X}$ give the location of the endpoint (frame $\{B\}$) with respect to frame $\{A\}$. The force produced at the endpoint with respect to frame A is ${}^A_B \vec{F}$. is related to the motor applied torques $\vec{\tau}$ by the Jacobian ${}^A_B J$ as: $\vec{\tau} = ({}^A_B J)^T ({}^A_B \vec{F})$.²⁹

environment during manipulation through exploratory actions. For example, when CARDEA reaches for a door it doesn't need to know the distance of the door *a priori*. Instead, by reaching forward blindly but compliantly, the distance of the door can be sensed during the manipulation act by monitoring the joint velocities.

Virtual spring control is a method by which a virtual spring and damper system is simulated at each joint or across multiple joints.³⁰ This approach to manipulator control affords a safe, human-like compliance to the manipulator while providing an intuitive control interface for doing behavior based manipulation.

We are able to directly sense and control the torque of each joint using SEA actuators. This allows us to do virtual spring control of the joints. We will limit our description of SEA actuators here as they are fully described in detail elsewhere.²⁷ In short, by placing a linear spring in series with the motor output, we gain a two-fold advantage. First, high impact shocks are low-pass filtered by the spring,

protecting the geartrain, and, second, the output force is easily sensed by measuring the spring displacement.

Simulated virtual springs can be attached between the arm end-effector and an external coordinate frame as well. This allows tasks such as reaching towards a target to be accomplished without solving the inverse-kinematics of the arm. Using the first 3 DOF of the arm as depicted in Figure 5, it is relatively simple to compute the forward kinematics and to relate the end-effector force to the joint torques.²⁹

A virtual spring attached to an external coordinate frame (say, to a door handle) can simulate virtual forces on the endpoint. These forces can be translated to joint torques which are then produced by the SEA actuators. In a behavior based approach to manipulation, low-level behaviors can actively attach and detach virtual springs to the sensed environment to guide the arm about the workspace. In addition, we can asymmetrically set the manipulator compliance by varying the virtual spring stiffness.

4. Cardea's Computational Hardware Architecture

In order to have a reliable and real-time platform, we designed a distributed computational architecture using custom made hardware.

Figure 6 shows the Cardea architecture for base, behavior, sensorimotor and power controllers plus the debugging system.

4.1. Behavior Control and Sensorimotor Controllers

All of CARDEA's higher-level behaviors are written in CREAL (see Section 5.2.1) and run on a custom embedded architecture called the STACK.³¹ The STACK is a small footprint, expandable architecture of up to 16 peripheral boards which plug into a bus to an 8-bit Rabbit 2000 embedded processor. The addressable peripheral boards are stacked under the the main processor and employ 8-bit PIC microcontrollers to read sensors or command actuators; and handle bus communication using a 9-bit RS-485 protocol.³² Each peripheral board is referenced in software as a buffer which can be written to or read from.

CARDEA's higher-level behaviors and some of its perceptual and motor systems are split between two STACKS communicating via an RS-232 token ring network. The Behavior STACK runs the robot's navigation behaviors which are described in detail in Section 5.2, and its peripheral boards interface to the ultrasound sensors, the contact switch whiskers and the R/C servo motor for camera pan. The Arm STACK controls the arm movements via a peripheral custom 16-bit DSP board, as explained in detail in Section 3.3. It also interfaces with the laptop running the vision system through an RS-232 standard. Such a low-bandwidth connection is sufficient in this case, since the transmitted packets are relatively small and the visual output is updated at 30 frames per second (see section 5.1 for details of CARDEA's vision).

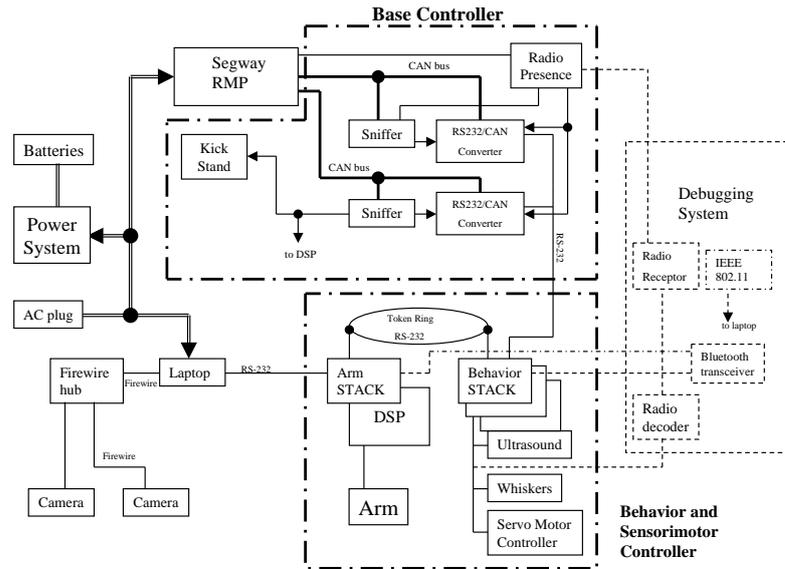


Fig. 6. CARDEA's computational hardware architecture, comprising modules that encapsulate the base control, the behavior and sensorimotor control and the power and debugging support systems. The base controller consists of CAN bus sniffers, RS-232/CAN converters, kickstand activation circuitry, and a radio presence check. The behavior and sensorimotor controller counts two STACKS which run high-level behavior for the RMP base and the arm respectively. They interface to all sensors including the vision system running on a Linux-based laptop.

4.2. Base controller

The base controller circuitry serves the dual purpose of implementing the actions and sensor collection of higher-level behaviors and running low-level safety checks on the base. Since CARDEA operates in built-for-humans environments, it must be prepared to fail gracefully, without compromising the safety of humans around it or damaging the environment.

The RS232/CAN converters shown in Figure 6 are the link between the high-level behaviors running on the Behavior STACK and the RMP base. They communicate with the STACK using the RS-232 standard, and with the RMP through a CAN bus. The RS-232/CAN converters buffer and reformat the data between these protocols. Their operation can be inhibited, in which case the RMP goes into its balancing mode. Two independent CAN busses are a safety feature of the RMP design²² and therefore require two converters.

CARDEA has a dynamically balancing platform, and so it is necessary to im-

plement safety measures to minimize the likelihood of the robot falling. We have implemented a number of mechanisms which operate on all control levels from the high-level robot behaviors to mechanical stops.

One such mechanism is the active kickstands, which are visible on the front of the robot on Figure 1. These are deployed whenever one of the sniffers placed on each of the CAN busses detects when the robot's pitch become greater than 18 degrees - a threshold angle obtained through experimentation. The sniffer also inhibits the RS-232/CAN converters so no commands come through to the base, alerts the DSP board on the Arm STACK, and reconfigures itself as a CAN bus controller to issue a shutdown command to the RMP. The arm automatically retracts into a safe position in this failure mode.

The sniffer also watches for commands being sent to the RMP that make it tilt at an angle between 15 and 18 degrees. In that case, all communication with the behavior controller is interrupted until the platform balances itself within 5 degrees of vertical. This ensures that the RMP can right itself from a precarious situation without failure and shutdown.

As a yet another safety measure, a radio remote control is used to select the operational mode of the robot out of three possibilities—the stop, manual and autonomous modes. In stop mode, the RS-232/CAN converters are inhibited, all communication between the high-level behavior controller and the RMP is interrupted and the robot comes to a complete stop. The manual mode allows a human operator to drive the robot using a joystick on the remote control. In autonomous mode, communication is normal but the joystick can override the commands sent from the behavior controller. In the current setup, the RMP will not start unless the Radio Presence check receives signal from the remote control. If during operation the signal from the remote control is lost, the Radio Presence subsystem will interrupt the communication line and it will stay in that state until the system is power cycled. To achieve that level of control, the Radio Presence subsystems is directly connected to the emergency stops of the RMP.²²

4.3. Power circuitry and debugging support

The robot, except the laptop and the RMP base, receives energy from lead acid batteries. Three independent groups of batteries are used - one for the safety systems, one for the arm and one for the rest of control electronics. The power system is fed by voltage from all three battery groups and has three functions. It supplies the voltages required by the different circuits, it switches the energy source between battery or external AC and it recharges the batteries. The AC connection is common to the power system, the laptop and the RMP recharger; therefore, only one connection is needed to recharge the robot.

In order to debug CREAL programs, a RS232/Bluetooth converter is used to program or monitor the STACKS remotely. For debugging vision algorithms, a LAN IEEE 802.11 card is connected to the laptop.

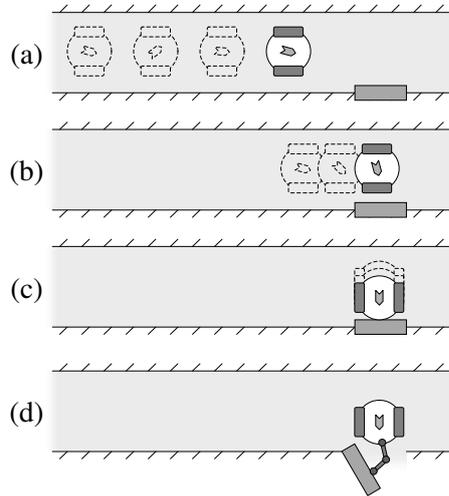


Fig. 7. Phases of corridor navigation. In (a), the robot is proceeding down a corridor. The robot keeps away from the corridor walls using visual information from the lower fixed camera, subsumed by sonar and whiskers if necessary. In this phase, the robot moves quite rapidly (approximately 1 m/s). The active camera scans the walls, looking for a door. When it detects a potential door, the robot slows down (b). As the robot continues to navigate as normal, the active camera fixates the potential door. If it is not in the end sufficiently “door-like” after this examination, the robot speeds off again. Otherwise it halts parallel to the door, turns to face it, and then approaches it (c). Alignment with the door is done using sonar and whisker information. Once alignment is complete, the robot arm is extended to push the door open (d).

5. Cardea’s High Level Software Systems

In this section we describe two high level software systems within the robot: vision and behavior based navigation and manipulation. The vision system, running on a Linux based laptop, interfaces to the behavior based system via messaging along a serial port to one STACK and through the token ring network connecting STACKS. The behavior based navigation and behavior based manipulation run on these same separate STACKS and communicate via the token ring.

5.1. The Vision System

CARDEA’s vision system exploits the common appearance and geometry of built-for-human environments to efficiently follow corridors and find doors. In built-for-human environments, doors, walls, and floors tend to have distinct appearances. Rooms with walls that have the same appearance as the floors are uncommon. CARDEA’s vision system makes use of appearance models to distinguish image regions that correspond to walls, doors, and floors. Doors, walls, and floors also tend to have similar geometric characteristics across human environments. CARDEA’s visual routines and the mounting positions of CARDEA’s two cameras take advantage

of some of the common geometric structure in built-for-human environments.

There is a long history of mobile robots that use vision for navigation in built-for-human environments.³³ In general, the common appearance and geometry of built-for-human environments has been used both implicitly and explicitly to help vision processing. The distinct functional requirements of doors, floors, and walls tend to lead to distinct materials and construction methods, which often result in distinct appearances. For example, floors must be able to handle the daily rubbing, scratching and pounding of feet carrying heavy bodies, which discourages the use of paint on floors and encourages the use of materials such as carpeting, tile, hard wood, and concrete. In addition, aesthetic and perceptual preferences often lead to doors that can be readily distinguished from the walls. Many geometric properties are common as well. For example, floors are usually flat and perpendicular to gravity, and walls are typically perpendicular to the floors. Doors are usually embedded in the walls, have their base against the floor, have a surrounding frame, and have a manipulable component in a narrow range of locations. Additionally, The dimensions of most human corridors and doors fall within a narrow range, and can be safely bounded below, since walking humans must fit comfortably. Many of these geometric properties are implicitly assumed by CARDEA's vision system. Although there will always be exceptions to these typical properties, methods that take advantage of them are usable in a significant percentage of human environments.

5.1.1. *The phases of door approach*

When in a corridor, the parallel execution of CARDEA's visual routines leads to four phases of behavior. In the first two phases, CARDEA traverses a corridor while seeking out doors. In the last two phases, CARDEA aligns with the first door it finds and enters the doorway. During these phases of behavior, two types of visual routines are active, floor following routines on the lower camera that help CARDEA navigate down the corridor, and door approaching routines on the upper camera that help CARDEA locate, investigate, and align with the first door CARDEA encounters (see Figure 7).

5.1.2. *Appearance based detection*

CARDEA's visual routines make use of color and texture based appearance models in order to detect floors and doors. CARDEA's FireWire cameras allow control of gain, exposure time, white balancing, and other parameters which make appearance models more effective. These models take the form of normalized histograms, which can be usefully thought of as an approximation of a probability mass function, $P(x)$.³⁴ These detectors are easy to debug, easy to train, straightforward to extend, effective over a wide variety of indoor human environments, and sufficient for the current building CARDEA inhabits.

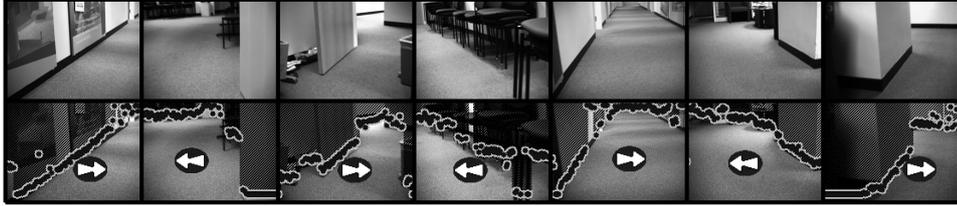


Fig. 8. CARDEA navigates using a combination of visual, sonar, and whisker information. This figure shows the visual estimate of free space in front of the robot. The current view from the fixed, downward-facing camera is examined to find the transition from floor to wall/obstacle by scanning along vertical columns of pixels and comparing their color and the local texture with models trained for the robot's environment. The overlaid blobs on the images in the lower row show the estimated floor boundaries for the images in the upper row. The arrow shows the direction the robot should move in (unless visual navigation is subsumed by sonar information).

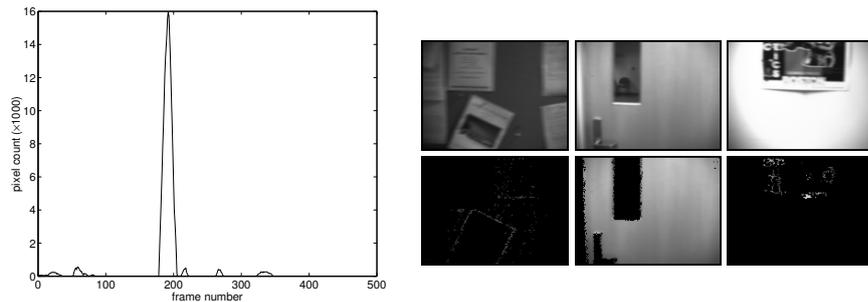


Fig. 9. Door detection experiment. The plot on the left shows the number of door-colored pixels detected by the upper camera as the robot traverses a corridor (the camera is fixed at 90° to the body for this experiment). The large peak corresponds to the view shown in the upper middle image. Two side peaks are also shown, responses for a notice board (left) and poster (right) respectively. During normal operation, the upper camera moves to maximize the number of door-colored pixels it sees, with periodic perturbations to disturb it from local optima.

5.1.3. Floor detection

CARDEA's corridor following system is modeled after the approaches of Horswill³⁵ and Lorigo.³⁶ An appearance based floor model is applied to video frames coming from CARDEA's lower camera. This model detects which pixels in each frame are likely to have been generated by the floor. A pixel with color, (r, g, b) , and texture, t , is marked as floor if the estimated probability, P_{color} , of the floor generating the color (r, g, b) is greater than the floor color threshold, T_{color} , and the estimated probability, $P_{texture}$, of the floor generating texture, t , is greater than the floor texture threshold, $T_{texture}$.

$$Detect_{floor}(r, g, b, t) = \begin{cases} 1 & \text{if } (P_{color}(r, g, b) > T_{color}) \wedge (P_{texture}(t) > T_{texture}) \\ 0 & \text{otherwise} \end{cases} \quad (1)$$

For each column of the image, I , the non-floor pixel that is closest to the bottom of the image is marked as the point at which the floor may end and an obstacle may begin. This leads to a function, D , that returns this pixel distance for each column x of the image I . In the following equation, F generates a feature image made up of (r, g, b, t) tuples from the input image I .

$$D(x, I) = \text{Min}\{d : Detect_{floor}(F(I)(x, d)) = 0\} \quad (2)$$

For indoor human environments, the floor is typically flat and CARDEA is sitting on the floor, so that the pixel distances for this floor/obstacle transition function, D , map directly to distances along the floor. Consequently, the 1st moment and average distance of D can be used to control CARDEA in order to follow the floor while avoiding obstacles, including the walls. The 1st moment, M_{floor} , of the function D suggests the direction to which CARDEA should turn in order to maximize the amount of clear floor ahead, while the average distance, A_{floor} , over the center half of the distance function D indicates the amount of clear floor currently in front of the robot.

$$\begin{aligned} M_{floor}(I) &= \frac{\sum_{x \in X} x * D(x, I)}{\sum_{x \in X} D(x, I)} \\ A_{floor}(I) &= \sum_{(0.25|X|) < x < (0.75|X|)} D(x, I) \end{aligned} \quad (3)$$

M_{floor} is used to control the turning rate and direction of the robot, while A_{floor} influences the robot's forward velocity. Within corridors with common structure and dimensions, these visually determined directions tend to move the robot down the center of the corridor (see Figure 8).

5.1.4. Door detection

Color histograms are commonly used to detect pixels of interest,^{37,34} and color is often used to detect regions to which a robot should orient.³⁸ CARDEA's door detection model uses 2D color histograms that represent the probability that a pixel with hue h and saturation s is generated by the surface of interest. Door detection uses a door appearance model, P_{door} , and a not-door appearance model, P_{-door} , trained on door images and video of door-free walls respectively. The door detector uses these two color models along with the user set parameter α , which can take prior probabilities and expected costs into account, to generate a maximum-likelihood door color categorization, $Detect_{door}$, over the (h, s) plane.

$$Detect_{door}(h, s) = \begin{cases} 1 & \text{if } (\alpha)P_{door}(h, s) > (1 - \alpha)P_{-door}(h, s) \\ 0 & \text{otherwise} \end{cases} \quad (4)$$

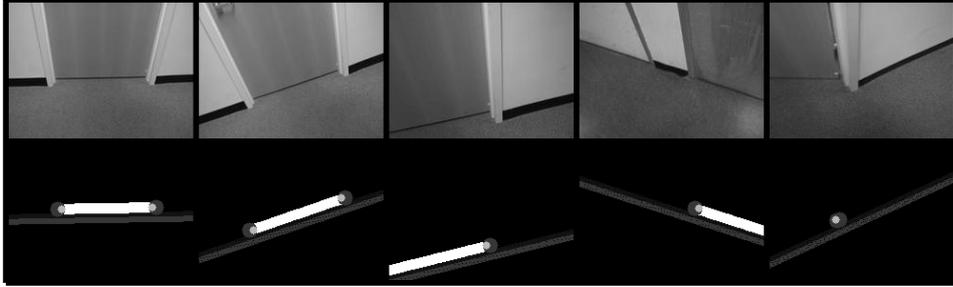


Fig. 10. CARDEA has a module for analyzing the door-floor interface, to detect exactly where the door frame is relative to its body, and whether the door has opened. This will be important when the robot has the ability to turn handles, so the robot can verify that the door is indeed opening. The Hough transform is used to find the strongest edge at $\pm 45^\circ$ to the horizontal, and a Viterbi search is performed above this edge to find the best match for the door location, based on color information. The upper row here shows typical images on the floor-door interface, and the lower row shows the output of analysis. The circles denote the left and right edges of the door, and are connected by a dark line. The long light line denotes the floor line. The image on the right shows that when the door is open, this can be detected.

This maximum-likelihood model, $Detect_{door}$, is in the form of a binary image, so it is easy to visualize and debug and leads to very efficient categorization of a pixel, since categorizing color only requires a look-up from a table with binary entries. While following a corridor using the lower camera, the upper camera looks at the walls and categorizes the pixels as coming from a door or not coming from a door (see Figure 9). The system makes a count, C_{door} , of these pixels in order to determine how close CARDEA is to the door, and calculates the 1st moment, M_{door} , in order to fixate and align with the door. In the following equation, $HS(I(x, y))$ returns the hue and saturation of the pixel at location (x, y) in image I .

$$\begin{aligned}
 C_{door}(I) &= \sum_{x \in X, y \in Y} Detect_{door}(HS(I(x, y))) \\
 M_{door}(I) &= \frac{1}{C_{door}} \sum_{x \in X, y \in Y} Detect_{door}(HS(I(x, y))) \begin{bmatrix} x \\ y \end{bmatrix}
 \end{aligned} \tag{5}$$

5.2. Behavior based Navigation and Manipulation

This section starts by describing CARDEA's behavior based software named CREAL. It then describes the implemented behaviors that collectively control hallway navigation and door passage.

5.2.1. CREAL: A CREATURE Language

The behavior system for CARDEA is programmed in a language known as CREAL³⁹ for *CREATURE Language*, which is similar in concept to the earlier *Behavior Language*⁴⁰, **L**⁴¹ and others⁴². All these languages were designed to make it easy to

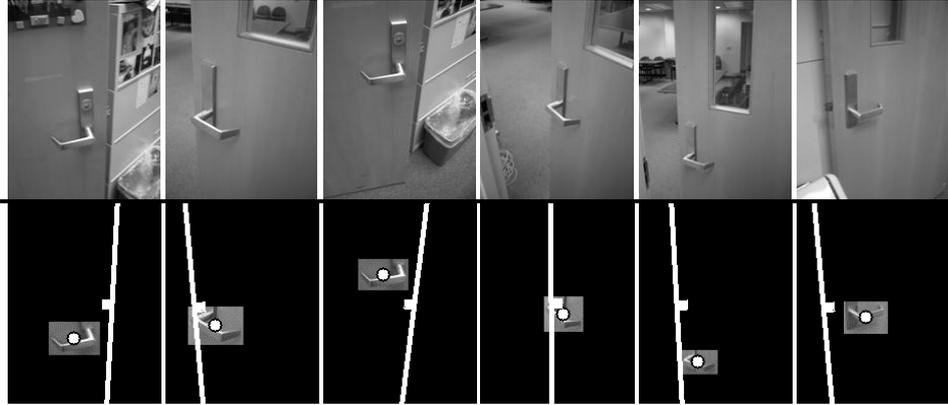


Fig. 11. While CARDEA cannot currently open a door, some work has been done to prepare the perceptual modules needed for this. A simple handle detector exists which finds the most plausible door edges in the current scene using a Hough transform and filtering by door color, and then searches for a perpendicular element near to this edge. In this figure, the white dots correspond to the estimated location of a handle, and the long edge shows the estimated door edge.

implement behavior based programs designed in the incremental and concurrent spirit of the subsumption architectures.⁴³

In CREAL programs the user specifies *modules* which are collections of threads sharing a local lexical environment. The threads may be scheduled to run periodically, when a condition is true, or on the occurrence of some combination of events. The scheduler for threads is not preemptive. Rather the semantics of CREAL are defined so that it is impossible to write a thread which will not reach a suspension point in a bounded amount of time. The threads are scheduled round robin, and should one be ready to run it is run until it suspends.

Each module can have both input and output *ports*. Modules are connected together with virtual wires connecting an output port of one to an input port of another. Messages, 8 or 16 bit numbers, can be sent along these wires. Wires can *suppress* and *inhibit* each other as defined in the original subsumption architecture.⁴³

The CREAL compiler is written in Emacs Lisp, and provides a macro environment at compile time which lets the user build higher level languages that expand into modules and wires.

For CARDEA the CREAL language supports both 8 and 16 bit arithmetic. The fundamental clock frequency is 64Hz, and no thread runs more often than that. Although our benchmark suggests that the Rabbit 2000 processors, which we use, could support close to 1,000 threads we have not used more than 50 threads on a single processor in any of our testing to date.

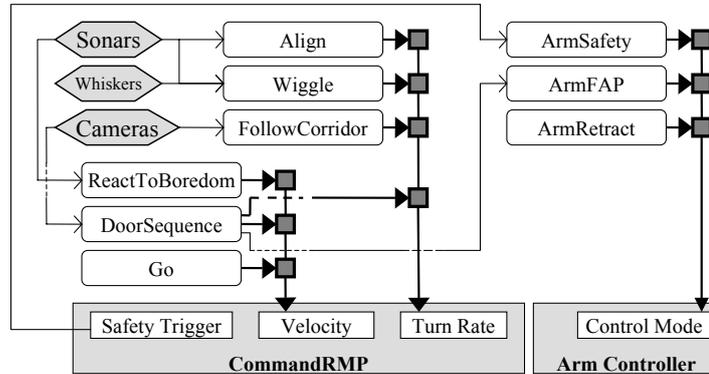


Fig. 12. The subsumption network of Cardea's key behaviors and their sensory inputs. A box at a wire junction indicates that the behavior sending messages to it subsumes messages of behaviors lower on the wire. To control the RMP there are three wires (each denoted by a black line): velocity, turn, and control mode. Following the velocity wire from its input back to its sources: *Go* continuously requests to go forward. Its message will be overridden when *ReactToBoredom* senses CARDEA has not moved in a while, or when *DoorSequence* is sensing door related percepts. Within the turn wire network no behavior continuously sends a turn message. Each behavior senses a unique set of environmental conditions that indicate to it that turning is required. *Align* senses a planar perpendicular surface in front of the robot. These are CARDEA's perceptual features of a door. *Wiggle* senses contact to one side of the robot via a whisker and, given sufficient space on the other side of the robot, requests a slight turn away from the point of contact. *FollowCorridor* uses camera information to head toward free space. *DoorSequence* handles door detection, turning toward the door, determining when to shove the door, and waiting for it to swing open. Within the arm wire network: *ArmRetract* continuously requests that the arm be tucked out of the robot workspace. It is subsumed by *ArmFAP* based on a trigger from *DoorSequence*. At any time, *ArmSafety* can take control of the wire to protect the arm in case of a fall.

5.2.2. Hallway Navigation and Door Opening

The behavior software for CARDEA's current capability is quite simple. Internally CARDEA has eleven low-level behaviors that are tight sensory motor loops. Figure 12 shows how CARDEA's nine key behaviors are interconnected using three wires for velocity, turn and arm control messages. The behaviors coordinate through mutually interacting with the environment and by one behavior sometimes subsuming the messages of others along a wire.

The arm's behaviors have access to several control modes implemented on the arm DSP controller. A control mode is a type of servo control loop and its associated gains. The primary control modes are: joint force, velocity, and angle; and end-point force, velocity, and position. All control modes utilize a low-level force control loop which ensures that the manipulator is compliant and safe to interact with. A behavior may employ different combinations of control modes and gains in the execution of its motor action. As behaviors are dynamically inhibited and subsumed during the manipulator's interaction with the environment, the manipulator can

exhibit different control properties. This ability to rapidly switch between different control modes is a key aspect of our behavior based approach to manipulation.

The **ArmRetract** behavior, when in control of the wire to the control mode, maintains a fixed virtual spring posture (see Section 3.3.2) to keep the arm withdrawn. **ArmSafety** is a retract reflex activated when CARDEA's **CommandRMP** module detects a dangerous situation. It subsumes control over all other arm behaviors and uses a force control mode to quickly withdraw the arm from the robot workspace.

There are four general behaviors (let us call them *macro-behaviors*) an observer sees when she watches CARDEA go down hallways, shove open an ajar door and pass through. During each macro-behavior, a number of CARDEA's behaviors operate simultaneously and switch rapidly.

- (1) *Macro behavior:* CARDEA traverses the corridor at walking pace, slightly weaving, yet avoiding the walls. *Internal Behaviors:* **Go** runs continuously to control CARDEA's speed. **FollowCorridor** runs when the vision system warns of a wall. **Wiggle** runs when CARDEA brushes a whisker on a wall.
- (2) *Macro behavior:* CARDEA encounters an obstacle in front of it, slows down and either turns away from it or aligns perpendicular to it. Once aligned, if CARDEA can not do anything, it appears bored and backs away to head off somewhere else. *Internal Behaviors:* **Go** directs CARDEA forward. **CheckMargins** slows CARDEA down as it detects, with sonar, obstacles in front. **Align** is active when it senses a planar, vertical surface perpendicular to the floor that is in front of the robot. It aligns CARDEA with the surface. After alignment, **ReactToBoredom** will back CARDEA away if CARDEA does not move for three seconds.
- (3) *Macro behavior:* CARDEA spots a door ahead, slows down, turns to line up with the doorway. After shoving open the door, it goes through. *Internal Behaviors:* **FollowCorridor** checks two messages from the lower camera that tell it where to head for maximal free space. If the upper camera detects a door ahead, it slows down and starts a short monostable. If the monostable expires before a door is confidently detected, it speeds up again. Otherwise, **DoorSequence** takes over and aims CARDEA towards the door. **Go** and **Align** align the robot closer to the door and the arm is signaled to push the door. During pushing, CARDEA is stopped. After the door swings open, **Go** directs CARDEA through.
- (4) *Macro behavior:* The door is pushed open. *Internal Behaviors:* This is accomplished with a fixed action pattern executing the following sequence of simple behaviors:
 - (i) **DoorReach** : This moves the end-effector to a fixed posture where the door handle is assumed to be. Because the arm workspace is constrained by the door, this behavior utilizes a joint angle control mode with high stiffness to precisely reach across the robot body without contacting the door.
 - (ii) **DoorPush**: Virtual springs (see Section 3.3.2) fixed to the opposite side of the door handle cause a forceful contact with the door.

- (iii) **DoorOpen**: The arm sweeps from left to right, maintaining contact with the door. Asymmetric compliance in the arm is employed with stiff virtual springs in the lateral direction and soft springs in the vertical direction.

The behaviors that ultimately navigate the RMP do so via the turn and velocity wires which terminate at the **CommandRMP** module. The **CommandRMP** module is responsible for the velocity and turn commands sent to the base controller. Its only action is to ensure that these commands will never put CARDEA into an unsafe or intractable situation.

6. Results

As proof of concept, the first demonstration CARDEA performs is the navigation of an office corridor, detection of a slightly ajar door, a push to open the door, and entry to the adjacent room. CARDEA's current layers of behavior for this functionality are intended to be situated below a real world application. The robot has no built-in knowledge that it is situated in a corridor. It is simply started up and left to its own devices to explore. By default it proceeds forward unless environmental stimuli influence its decision to change course. It is constantly on the lookout for a door while avoiding objects to its sides that have wall-like obstacle properties. Without visual confirmation of a door, if something like a wall or a door appears more or less in front of CARDEA it will square itself in front of the object. This way, should that be a door that it has not recognized, it will be able to proceed through the (clear) doorway if a human opens the door. After a while, if no one opens the 'door', Cardea backs up and heads somewhere else. When CARDEA sights a door and aligns to it, it triggers a push action with its arm to attempt to open the door. When the door passage is clear, it is ready to proceed through.

Figure 6 is a series of snapshots taken from one continuous span of operation. Cardea is started somewhat in the middle of the hallway simply for convenience. In this particular case, the door happens to be to CARDEA's right. CARDEA slows down as a poster board appears on its right. However, as it acquires more visual information on the poster board, the board is eliminated as a door candidate and CARDEA speeds up again. Upon sighting and confirming a door, it turns and aligns itself to the door, shoves the door open and proceeds through the passage.

7. Future Work: Cardea shoves open a door and beyond

Figure 14 is a cartoon representation of our goal for the ultimate version of CARDEA. In terms of hardware, the next generation of the arm will incorporate a shoulder roll DOF as well as a four DOF SEA hand with tactile sensing. We are working on other components including a head and multiple arms but do not have funding for integration into a single robot at this point, nor do we expect to ever build a robot which looks exactly like this cartoon.

We are also working to improve some of CARDEA's capabilities. Currently, in



Cardea starts down an office hallway. Timestamps in lower right corner show it travels at walking speed. A door can be seen to the robot's right just past the poster on the bulletin board.



Cardea slows as it first detects the door. Upon confirmation of door, it turns then approaches the door. It adjusts itself to be head on to the door.



Cardea's fixed action pattern for striking the door. The arm sweeps from its retracted position to the right and downwards then bends up and moves forward.



At its full arm extension, with door contact, Cardea's base moves backward slightly, then Cardea moves forward, through the door passage.

Fig. 13. A series of snapshots with timestamps in lower right corner demonstrating Cardea.



Fig. 14. This is a cartoon of a later generation CARDEA. It has an expressive face and active vision head, and three 6 DOF arms. As illustrated here it uses the same Segway “HT” base as our current robot, although we might ultimately prefer to use one based on Segway’s newer smaller model “p Series” platform as that is slightly narrower and lighter, and would navigate through narrower spaces.

terms of vision processing, CARDEA must be trained for each new environment. Future work may partially automate this training through the use of generic segmentation algorithms.⁴⁴ Additional appearance features, such as multi-scale texture elements, could be incorporated into the detectors we use without much difficulty. More advanced appearance based models that make use of BRDF (bidirectional reflectance distribution function) approximations or more complex surface properties, such as specularity and shininess, would require different training protocols and detection algorithms, but could still fit into CARDEA’s vision framework.⁴⁵

We have already developed additional routines for door analysis that should be useful for both locating and opening doors (see figures 10 and 11). Preliminary tests with captured data are promising, although these methods have yet to be integrated into Cardea.

Our vision is to have a robot that can freely operate in a built-for-human space interacting with people socially and manipulating a wide variety of objects in its world. We are not aiming our design at a particular application although there are a number of general areas of application which we hope that our work would help enable. These include:

- an assistant for in-home elder care providing basic nursing services,
- a mobile host robot that can interact with people visiting an office building or some other facility,
- a physical personal assistant for everyday office style work,
- and a reconnaissance robot that can enter undamaged buildings and explore them without causing any damage.

The experimental robot that we hope to morph our current CARDEA towards would have aspects of humanoid form though it would not try to mimic a human directly. Aspects of our vision for CARDEA include:

- an active vision head so that CARDEA can have stereo peripheral vision for navigation, and the ability to rapidly foveate stereo vision on people and objects for interaction,
- an expressive face so that CARDEA can interact socially with people as did our earlier robot Kismet,⁴
- three arms (humans look symmetric but in reality our left and right arms are differentially specialized) that are specialized to different grasping capabilities, and which can be used in pairs to provide different holding and manipulation specializations,
- the ability for the robot to recognize people from day to day and to develop unique and ongoing relationships with individuals,
- a vision system which is able to recognize objects¹⁰ including those that it has not been taught about but which it has discovered itself,
- the ability to grasp newly seen objects, preshaping its hand, and orienting its body to provide the appropriate grasp.

Our current cartoon design for CARDEA does leave out some important capabilities which we think will be important for a flexible humanoid form robot that can operate indoors. These include a much more dynamic balancing capability such as hips so that the robot can bend over and adjust its balance in order to reach towards distant objects, a swivel degree of freedom (such as that provided jointly by human hips and spine) so that the robot can reorient its torso without changing its stance, an active laser system to provide three-dimensional data for local navigation to augment passive vision, and a rich set of touch sensors to augment CARDEA's current sense of touch which is based on sensing forces.

We believe humanoid robots will have a future of applications. We believe that a rough humanoid form will mean that such robots will be well adapted to operate in built-for-human environments and will make it easier for people to interact with them. We do not, however, believe that such robots need to religiously adhere to exact human images in order to be successful and acceptable.

Acknowledgements

A number of other people worked on the hardware of the Cardea robot, including Jessica Banks, Alana Lafferty, and Alex Moore. They, along with Kathleen Richardson, also provided logistical support during testing of CARDEA. The Segway Robot Mobility Platform was provided by DARPA's Information Processing Technology Office through SPAWAR in San Diego. Funding for this work was provided by DARPA under contract number DABT 63-99-1-0012.

References

1. H. Lim, Y. Yamamoto, A. Takanishi, Control to realize human-like walking of a biped humanoid robot, in Proceedings of the IEEE International Conference on Systems, Man, and Cybernetics (2000), vol. 5 3271–3276.
2. Y. Sakagami, R. Watanabe, C. Aoyama, S. Matsunaga, N. Higaki, K. Fujimura, The intelligent ASIMO: system overview and integration, in Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (2002), vol. 3 2478–2483.
3. S. Kagami, M. Mochimaru, Y. Ehara, N. Miyata, K. Nishiwaki, T. Kanade, H. Inoue, Measurement and comparison of human and humanoid walking, in Proceedings of the IEEE International Symposium on Computational Intelligence in Robotics and Automation (2003), vol. 2 918–922.
4. C. Breazeal, *Designing Sociable Robots* (MIT Press, 2002).
5. T. Ogata, Y. Matsuyama, T. Komiya, M. Ida, K. Noda, S. Sugano, Development of emotional communication robot: WAMOEB-2R-experimental evaluation of the emotional communication between robots and humans, in Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS-00) (2000), vol. 1 175–180.
6. H. Kozima, Attention-sharing and behavior-sharing in human-robot communication, in Proceedings of the IEEE International Workshop on Robot and Human Communication (ROMAN-98) (1998) 9–14.
7. R. Brooks, L. A. Stein, Building brains for bodies, *Autonomous Robots* **1** (1994) (1) 7–25.
8. B. Scassellati, Foundations for a Theory of Mind for a Humanoid Robot, Ph.D. thesis, Massachusetts Institute of Technology, Department of Electrical Engineering Computer Science, Cambridge, MA, 2001.
9. G. Metta, G. Sandini, J. Konczak, A developmental approach to visually-guided reaching in artificial systems, *Neural Networks* **12** (1999) (10) 1413–1427.
10. P. Fitzpatrick, From First Contact to Close Encounters: A Developmentally Deep Perceptual System for a Humanoid Robot, Ph.D. thesis, Massachusetts Institute of Technology, Department of Electrical Engineering Computer Science, Cambridge, MA, 2003, available as Tech. Rep. AITR-2003-008.
11. M. Fujita, K. Kageyama, An open architecture for robot entertainment, in Proceedings of the First International Conference on Autonomous Agents (ACM Press, 1997) 435–442.
12. M. Williamson, Neural control of rhythmic arm movements, *Neural Networks* **11** (1998) (7-8) 1379–1394.
13. R. Platt, O. Brock, A. H. Fagg, D. Karupiah, M. Rosenstein, J. Coelho, M. Huber, J. Piater, D. Wheeler, R. Grupen, A framework for humanoid control and intelligence, in Proceedings of the IEEE International Conference on Humanoid Robots (HUMANOIDS-03) (Karlsruhe & Munich, Germany, 2003) .
14. S. Hashimoto, S. Narita, H. Kasahara, A. Takanishi, S. Sugano, K. Shirai, T. Kobayashi, H. Takanobu, T. Kurata, K. Fujiwara, T. Matsuno, T. Kawasaki, K. Hoashi, Humanoid robot-development of an information assistant robot Hadaly, in Proceedings of the 6th IEEE International Workshop on Robot and Human Communication (ROMAN-97) (1997) 106–111.
15. T. Kanda, H. Ishiguro, T. Ono, M. Imai, R. Nakatsu, Development and evaluation of an interactive humanoid robot "Robovie", in Proceedings of the IEEE International Conference on Robotics and Automation (ICRA-02) (2002), vol. 2 1848–1855.
16. C. G. Atkeson, J. Hale, F. Pollick, M. Riley, S. Kotosaka, S. Schaal, T. Shibata, G. Tevatia, S. Vijayakumar, A. Ude, M. Kawato, Using humanoid robots to study

- human behavior, *IEEE Intelligent Systems: Special Issue on Humanoid Robotics* **15** (2000) 46–56.
17. R. A. Brooks, Elephants don't play chess, *Robotics and Autonomous Systems* **6** (1990) (1&2) 3–15.
 18. R. A. Brooks, Intelligence without representation, *Artificial Intelligence Journal* **47** (1991) 139–160, originally appeared as MIT AI Memo 899 in May 1986.
 19. R. A. Brooks, New approaches to robotics, *Science* **253** (1991) 1227–1232.
 20. R. A. Brooks, Intelligence without reason, in Proceedings of the 1991 International Joint Conference on Artificial Intelligence (1991) 569–595.
 21. R. C. Arkin, Behavior-Based Robotics (MIT Press, 1998).
 22. Segway LLC, 286 Commercial Street, Manchester NH 03110, Segway Robotic Mobility Platform (RMP): Instructions for DARPA Users, Version 1.2.
 23. Srf04, ultra-sonic ranger technical specification, <http://www.robot-electronics.co.uk/htm/srf04tech.htm>.
 24. S. Schaal, Dynamic movement primitives - a framework for motor control in humans and humanoid robots, in Proceedings of the International Symposium on Adaptive Motion of Animals and Machines (2003) .
 25. R. Grupen, J. Coelho, Acquiring state from control dynamics to learn grasping policies for robot hands, *International Journal on Advanced Robotics* **16** (2002) (5) 427–444.
 26. W. T. Townsend, J. K. Salisbury, Mechanical design for whole-arm manipulation, in P. Dario, G. Sandini, P. Aebischer (eds.), Robots and biological systems: towards a new bionics? (Springer-Verlag, 1993) 153–164.
 27. G. Pratt, M. Williamson, Series elastic actuators, in Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS-95) (Pittsburgh, PA, 1995), vol. 1 399–406.
 28. R. A. Brooks, C. Breazeal, M. Marjanovic, B. Scassellati, M. M. Williamson, The Cog project: Building a humanoid robot **1562** (1999).
 29. J. Craig, Introduction to Robotics (Addison Wesley, 1989), 2 edn.
 30. J. E. Pratt, Virtual model control of a biped walking robot, Tech. Rep. AITR-1581, MIT Artificial Intelligence Laboratory, Cambridge, MA, USA, 1995.
 31. R. A. Brooks, et al., ALIVE STACK User Manual, February 2003, unpublished.
 32. R. A. Brooks, Proposed CREAL Communication Protocol - V1.0, July 2002.
 33. G. N. DeSouza, A. C. Kak, Vision for mobile robot navigation: A survey, *IEEE Transactions on Pattern Analysis and Machine Intelligence* **24** (2002) (2) 237–267.
 34. D. A. Forsyth, J. Ponce, Computer Vision: A Modern Approach (Prentice Hall, 2002).
 35. I. Horswill, Polly: A vision-based artificial agent, in Proceedings of the 11th National Conference on Artificial Intelligence (AAAI-93) (AAAI Press, Menlo Park, CA, USA, 1993) 824–829.
 36. L. Lorigo, R. Brooks, W. Grimson, Visually-guided obstacle-avoidance in unstructured environments, 1997.
 37. M. J. Jones, J. M. Rehg, Statistical color models with application to skin detection, *International Journal of Computer Vision* **46** (2002) (1) 81–96.
 38. J. Bruce, T. Balch, M. Veloso, Fast and inexpensive color image segmentation for interactive robots, in Proceedings of the 2000 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS '00) (2000), vol. 3 2061–2066.
 39. R. A. Brooks, Creature language, Sep. 2003, <http://www.ai.mit.edu/people/brooks/creal.pdf>.
 40. R. A. Brooks, The behavior language user's guide, AI Memo 1227, MIT Artificial Intelligence Lab, Cambridge, Massachusetts, Apr. 1990.
 41. R. A. Brooks, C. Rosenberg, L -a common lisp for embedded systems, in Association

- of Lisp Users Meeting and Workshop, LUV'95 (1995) .
42. I. Horswill, Functional programming of behavior-based systems, *Autonomous Robots* **9** (2000) (1) 83–93.
 43. R. A. Brooks, A robust layered control system for a mobile robot, *IEEE Journal of Robotics and Automation* **2** (1986) (1) 14–23.
 44. P. Fitzpatrick, C. C. Kemp, Shoes as a platform for vision, in Proceedings of the 7th IEEE International Symposium on Wearable Computers (White Plains, NY, USA, 2003) 231–234.
 45. R. O. Dror, E. H. Adelson, A. S. Willsky, Recognition of surface reflectance properties from a single image under unknown real-world illumination, in Proceedings of the Workshop on Identifying Objects Across Variations in Lighting at the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (Hawaii, 2001) .

Biography



Rodney Brooks is Fujitsu Professor of Computer Science and Engineering and Director of the MIT Computer Science and Artificial Intelligence Laboratory (CSAIL). He has a life-long obsession with robots and Lisp.



Lijin Aryananda received her B.S. and M.Eng. in Electrical Engineering and Computer Science from MIT. She is currently a Ph.D. candidate at MIT CSAIL.



Aaron Edsinger received his B.S. in Electrical Engineering and Computer Science from Stanford and his M.S. in Computer Science from MIT. He works in behavior based robot manipulation as a Ph.D. candidate at MIT CSAIL.



Paul Fitzpatrick is a Postdoctoral Lecturer at MIT CSAIL. He received his M.Eng. in Computer Engineering from the University of Limerick, Ireland, and a Ph.D. in Computer Science from MIT for work addressing developmental approaches to machine perception on a humanoid robot.



Charles C. Kemp is a Ph.D. candidate at MIT CSAIL. His current research focuses on wearable computing as a platform for A.I. and robotics. He received his B.S. and M.Eng. in Computer Science from MIT.



Una-May O'Reilly, Ph.D., is a research scientist at MIT CSAIL who also publishes in evolutionary computation.



Eduardo Torres-Jara received his B.S. degree in Electrical Engineering from Escuela Politecnica del Ejercito, Ecuador, and his M.S. from MIT in Electrical Engineering and Computer Science. He is currently a Ph.D. candidate at MIT CSAIL. He is a member of the IEEE society.



Paulina Varshavskaya received her B.Sc. in Computer Science with Cognitive Science from University College London and her M.S. in Electrical Engineering and Computer Science from MIT. She is currently a Ph.D. candidate at MIT CSAIL.



Jeff Weber is a Research Engineer specializing in mechanical design in the Humanoid Robotics Group at MIT CSAIL.