Advanced Robotics, Vol. 19, No. 3, pp. 221–248 (2005)

Also available online - www.vsppub.com

Full paper

Design and control of a mobile hyper-redundant urban search and rescue robot

ALON WOLF [1,*], HOWARD H. CHOSET [2], H. BENJAMIN BROWN, JR. [1] and RANDALL W. CASCIOLA [3]

[1] *The Robotics Institute, Carnegie Mellon University, Pittsburgh, PA 15213, USA*
[2] *The Mechanical Engineering Department, Carnegie Mellon University, Pittsburgh, PA 15213, USA*
[3] *AugmenTech, Inc., 5001 Baum Boulevard, Suite 480 Pittsburgh, PA 15212, USA*

Received 1 June 2004; accepted 3 September 2004

Abstract—In this work we introduce a new concept of a search and rescue robotic system that is composed of a hyper-redundant robot (HRR) which is mounted on a mobile base. This system is capable not only of inspecting areas reachable by the mobile base, but also of inspecting unreachable areas such as small cracks and pipes, using the camera and sensors mounted on its end-effector. The HRR is composed of 14 actuated, serially chained, d.o.f. which have been designed for compactness, yet each is strong enough to support the entire robot structure. The low-level control system of the robot incorporates 14 embedded mini-circuit boards implementing a low number of wires and high-speed, I^2C data bus technology. The high-level control algorithms implement kinematic methods to perform on-line motion control both in a global coordinate frame and in the camera coordinate frame for which a camera is mounted on the robot end-effector. The user interface allows the user full and easy control and interpretation of the data received from the camera. All the components of the system are hose-able as required by search and rescue regulations. Finally, we describe the integration of all the above-mentioned components in a complete operational mobile hyper-redundant search and rescue robotic system. This system has been deployed and tested in several search and rescue training scenarios.

Keywords: Search and rescue robot; hyper-redundant mechanism; snake robots; mechatronics; user interface.

1. INTRODUCTION

Urban search and rescue (USAR) robots are no longer just an academic curiosity. Unfortunate past events such as the Mexico City earthquake in 1985, the Hanshin-Awaji (Kobe) earthquake in 1995, the 1995 Oklahoma City bombing, and the September 11 attack on the World Trade Center (WTC) stress the need for these

*To whom correspondence should be addressed. E-mail: alon.wolf@cmu.edu

robots in assisting rescue workers in unreachable or unsafe places. For example, in the Mexico City earthquake, of the 135 rescue workers killed, 65 died while going through confined spaces which were flooded, trapping the them inside [1, 2].

The September 11 terror attack on the WTC was the first known event where robots were actually used in an USAR effort. In the WTC, robots were used to explore unreachable spaces by delivering real-time photographic information from environments unreachable and hostile both to the rescue workers and their trained dogs. This unfortunate event provided an opportunity to test the existing USAR robotic systems in real rubble piles, to explore the robot–human interaction, and to define new requirements and mechanical needs for these systems. However, USAR research efforts had started even before these events. Some of these works confront the challenge of platform development [3–9], and some address the software and human interface development issues [3, 10–14]. In Ref. [3], the researchers introduce the idea of marsupial and shape-shifting robots for USAR, where shape-shifting qualities are used to overcome obstacles or get a better view point. This idea was further investigated in Ref. [4], where a team of marsupial robots was deployed to cover a large search area. Another robotic structure was studied in Ref. [7], where the researchers introduced a fire-fighting robot, capable of climbing rails such as those found on high-rise apartment complex balconies. Software and human interface development address control issues, multi-agent collaboration in the field [11], data flow from a multi-sensor system [12], and a human interface which enables the operator to focus on driving the robot and less on understanding the incoming information [4].

The present work introduces a new concept for USAR robotic systems which addresses the requirements defined following the WTC experience. The present system is composed of a hyper-redundant robot (HRR) mounted on a mobile base (Fig. 1). Both the mobile base and the HRR are equipped with cameras and audio sensors that provide the operators with real-time images and sounds from the environment. The main goal of the HRR (which resembles an elephant trunk) is to extend the sensing ability of the mobile base by enabling the operator to take advantage of the high maneuverability embedded in the HRR, and to steer the camera (or any other sensors) around obstacles into small cracks and to places not accessible to the mobile base.

A redundant robot has at least an extra 1 d.o.f. than required, in order to compensate for simple constraints, i.e. using an elbow-up versus an elbow-down configuration, to reach a target position. HRRs have many more d.o.f. than required, which enables them to handle more constraints, such as those presented in highly convoluted volumes, while at the same time enabling them to perform a variety of tasks. It is no surprise that HRRs are versatile — look at their biological counterparts, e.g. snakes, elephant trunks and worms, all of which can poke through and crawl through crevices as well as manipulate objects. Starting in 1972 with Hirose's pioneering work in HRR design [14], and following with the work of Chirikjian and Burdick [13], there has been considerable attention given

Figure 1. Our HRR mounted on a mobile base in a search and rescue scenario (NASA Aims).

to HRR design. Most of the mechanism designs, however, have been limited to the plane. A few three-dimensional (3D) robots were developed, but they were somewhat bulky or had limited strength. Our design seeks to provide both compactness and strength. When designing a hyper-redundant mechanism, we consider several critical design constraints, such as maximum torque-to-weight ratio to allow the robot to support itself, minimum envelope diameter to allow itself to fit through small cracks and confined spaces, minimum achievable radius of curvature resulting from short links with maximum angular travel between links (for high maneuverability), and rugged construction. Secondary design goals include minimum backlash and compliance in the structure, and 'reasonable' speed of motion. In this work we introduce a novel design of a 3D motion HRR which is composed of seven actuated, serially chained, 2-d.o.f. joints that have a compact structure and are strong enough to allow the robot to support itself.

The mechanism by itself is not enough; one must also consider mechatronic integration and control. Mechatronic requirements and low-level control, however, become non-trivial in HRRs because of the relatively many d.o.f. Even something as seemingly simple as wire complexity becomes an additional challenge for robot design because all of the wires must fit inside a small volume inside the mechanism. For the present robot we have developed custom electronics sitting on a novel bus structure to regulate actuation motion and have sensory input for future expansion.

Finally, high-level control is required because the user cannot independently control all of the internal d.o.f. at once to produce purposeful motion. This becomes very challenging due to the system's redundancy. We have also developed a novel

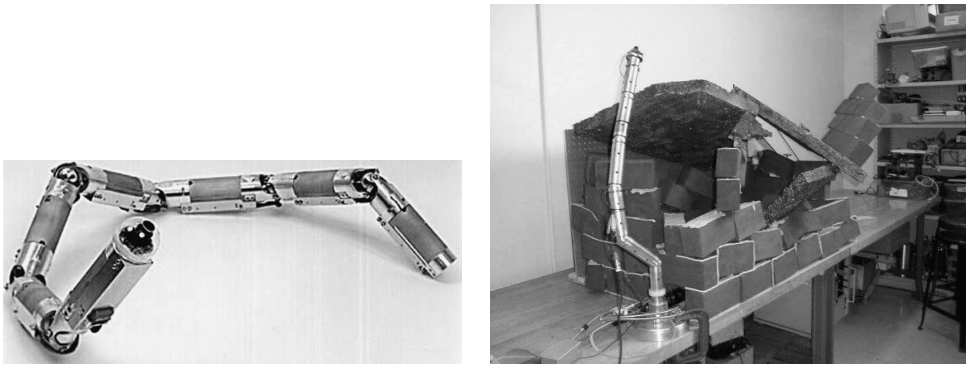
joystick input, image display and user interface to 'drive' the snake robot around using the motion control algorithms.

In the following sections we address all the above-mentioned challenges by introducing our hyper-redundant system's hardware and software, design, and integration into a complete and operational HRR mobile system for search and rescue applications.

2. PRIOR WORK

Snake robots and elephant-like robots are the main subject of several robotic researchers who often address them as HRRs [13]. The maneuverability inherent in these types of mechanical structures and their compliance, i.e. their ability to conform to environmental constraints, allow them to overcome obstacles of significant complexity compared to conventional robots, hence they are an important challenge for robotic mechanism designers [15, 16]. One of the pioneer works in this area was introduced by Hirose [14]. In this work he developed a device that mimicked the locomotion of real snakes on the ground. HRR research continued in the early 1990s at Caltech with the planar hyper-redundant manipulator by Chirikjian and Burdick. Their contribution focused on novel end-effector placement algorithms for these robots [13, 17].

Recently, other researchers, such as Yim [18] at PARC, Miller [19, 20] on his own and Haith at NASA Ames [21], have extended Hirose's pioneering work on snake locomotion, where Yim and Haith used Yim's Polybot modules to form a modular HRR. We are interested in building a HRR which is capable of maneuvering in three dimensions. The Pacific Northwest Labs developed a 3D mechanism which was incredibly strong, yet moved relatively slow and was quite large. This robot moved slowly because it was intended to be used for surgical bomb disarming, so that a technician could tele-operate this robot to probe the internals of a bomb without accidentally detonating it. Kinematically, the mechanism is a sequence of linearly actuated universal joints stacked on top of each other. Takanashi, at NEC, developed [22, 23] a new 2-d.o.f. joint for snake robots that allowed a more compact design (Fig. 2a). This joint uses a passive universal joint to prevent adjacent bays from twisting, while at the same time allowing 2 d.o.f.: bending and orienting. This universal joint enveloped an angular swivel joint, which provided the 2 d.o.f. Being installed on the outside, the universal joint rendered the joint relativelly bulky. Researchers at JPL [24] 'inverted' Takanashi's design by placing a small universal joint in the interior of the robot (Fig. 2b). This allowed for a more compact design, but came at the cost of strength and stiffness (backlash). Other known designs use cable/tendon actuation systems for driving the robot, yet these designs are somewhat cumbersome and require quite a large external driving system [14–25]. Finally, Ma *et al*. have also presented the mechanical design of a HRR and its control algorithm for inspection of confined spaces [26].



(a) (b)

Figure 2. (a) NEC snake robot. (b) JPL serpentine robot.

3. MECHANICAL STRUCTURE OF THE SYSTEM

3.1. Our hyper-redundant mechanism

The challenge for a HRR mechanism is to be strong enough to lift itself in three dimensions, but be small and light enough to be useful to demonstrate even basic planning. Also, the robot should have a small achievable radius of curvature resulting from short links with maximum angular travel between links (for high maneuverability). Finally, each of the joints should have a reasonable speed of motion. Integrating these requirements to a single working mechanical system is a great challenge. In this section we present our new hyper-redundant mechanism and its mechanical properties.

The HRR is composed of seven stages connected by seven actuated universal joints which are stacked in a serial kinematic chain (Fig. 3). A descriptive schematic of one stage of the HRR is given in Fig 4. An actuated universal-joint (U-joint) design was selected for its simplicity and ruggedness. In this design, U-joint 'crosses' are connected to one link with a pitch pivot joint and to the next with a yaw pivot joint. The pitch and yaw joints are always orthogonal, and intersect along the link centerlines; this leads to a relatively simple kinematic system. The pitch and yaw joints are actuated by linear actuators placed within the link's envelope. The links are configured such that the axes at each end of any link are parallel; thus, one link has pitch joints at both ends actuated by its two linear actuators and the next link has two yaw joints. This arrangement facilitates packaging of the two linear actuators side-by-side within the link.

Ball screws were chosen for the linear actuators because of their high efficiency (compared to lead screws) and effective speed reduction. The screws are fixed in bearings mounted to the links, while the nuts drive clevises connected to the crosses of the U-joints. The screws are driven by brush-type, permanent-magnet, DC motors which can be operated with simple, pulse-width-modulated (PWM) control. For compactness, the gear motor and ball screw are placed side-by-side with a small

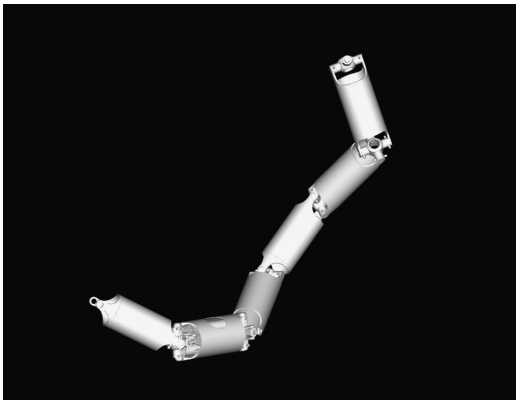


Figure 3. Seven actuated universal joints connected in a serial kinematic chain.

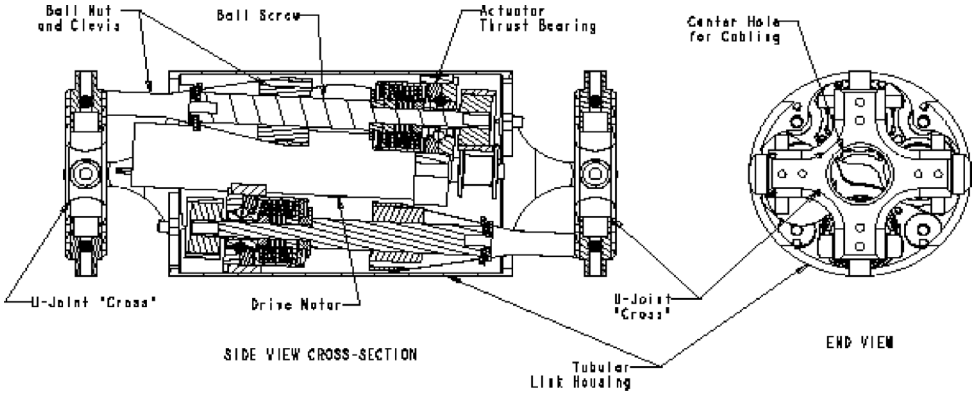


Figure 4. Descriptive schematic of one stage of the hyper-redundant robot.

toothed-belt drive connecting them. Each actuator is mounted to the link through a steel flexure that accommodates the slight lateral movement of the screw as the joint angle changes. A novel feature of this design is the overload mechanism or 'snubber' (Fig. 5). It is designed to absorb the kinetic energy of the links and motors when the mechanical stops are reached, and to accommodate imposed loads on the snake without damage to the actuators or structure. Belleville spring washers (four series sets of three parallel-stacked washers) are mounted in the 'snubber housing' such that the ball screw can move axially by 1 mm if the preload value is exceeded. Belleville washers are hardened steel washers with a slight conical shape. They can be compressed flat to produce an axial spring force. They can be combined in parallel or series, by stacking the cone sections parallel or opposing, to adjust the force and deflection characteristics. The thrust load of the screw is taken by a custom-made, four-point-contact bearing integrated into the 'snubber' housing.

The primary means of motion delivery to our HRR joints is through a ball screw mechanism. The ball screws are 6 mm diameter with 1 mm lead, rated at 700 N and are connected to the crosses at 14.7 mm from the pivot. The motors used are

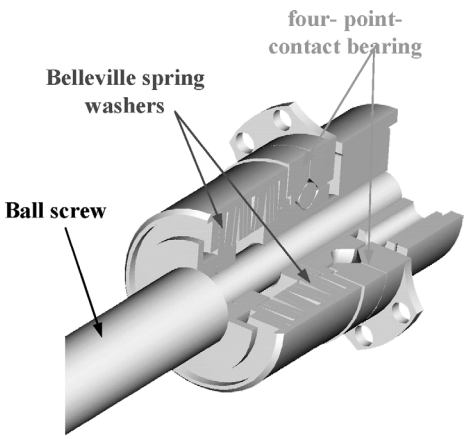


Figure 5. 'Snubber'.

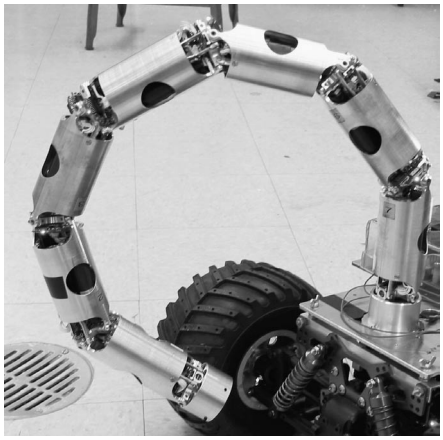


Figure 6. The ±55° of joint angular travel resulting in a highly maneuverability HRR.

Maxon RE-13 (13 mm diameter) gear motors with 16.58 : 1 planetary gear reducers and 16-count encoders (64 counts per revolution with quadruature encoding). The motors develop about 38 mN m of continuous torque; this translates to 380 N of force at the ball screw (well below the rated load), assuming there is a 2 : 1 belt drive and transmission efficiencies. The snubber mechanisms are preloaded to about 600 N to protect the ball screws and bearings from overload; no displacement occurs until this load value is reached, so the normal stiffness of the structure is not compromised. The motor no-load speed at the nominal 12 V input is 8900 r.p.m., which corresponds to 5 s time to travel the full 22.4 mm of screw travel resulting in ±55° of joint angular travel, which results in a high maneuverability (Fig. 6). Tests of the joints indicate that the actuators can produce 4.5 N m of torque at 12 V DC (0.40 A), i.e. each ball screw produces 307 N at 14.7 mm radius on the U-joint cross. Based on the expected 5.08 mN m at 0.40 A, theoretical output would be 1060 N

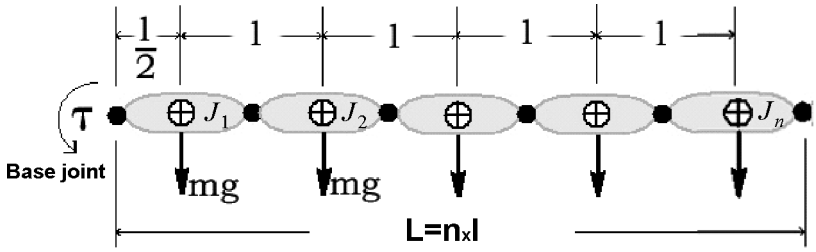


Figure 7. N-link serial chain.

with 100% transmission efficiency. This indicates that the overall drive efficiency is only (307 N/1060 N) 29%. This is much lower than the predicted efficiency of 48%, based on 75% efficiency for the motor gearing, 75% for the toothed belt and 85% for the ball screw. This point will need further investigation to see if significant increases in efficiency and output torque are possible.

In order to determine the number of links we can stack together to form the HRR, a simple calculation had to be done: the total torque about a joint needed to 'cantilever lift' (lift when extended horizontally) a single link, assuming its center-of-mass to be at its geometric center, is 0.113 N m. For N joints connected in a serial connection the total torque, τ, in the base joint is given by:

$$\sum M_{\text{base}} = \tau - mg\left(\frac{l}{2} + \frac{3l}{2} + \frac{5l}{2} + \cdots + \frac{(N+4)l}{2}\right) = 0, \tag{1}$$

hence $\tau = (mgl/2)N^2$ (Fig. 7). As one can see, the torque to lift N joints, in series, is N^2 times 0.113 N m.

Given 4.5 N m available joint torque and according to these calculations, our snake should be composed of seven links in order to be able to cantilever lift itself. Tests on the complete snake robot confirm this capability. This ability is important to allow the snake to achieve arbitrary configurations working against gravity.

3.2. Mechanical accuracy of the HRR

In this section we present an accuracy estimation of our 14-d.o.f. HRR. The robot accuracy is affected by errors related to motor accuracy and manufacturing accuracy (in this analysis we neglect control accuracy). These errors, which are unknown *a priori*, can be expressed by the four link kinematics parameters that were introduced by Denavit and Hartenberg [27]. The method that is implemented here for the analysis of our mechanical structure was first presented by Wu [28].

First, we determine the relationship between two successive local coordinate frames, i and $i + 1$. This can be derived using Denavit and Hartenberg kinematic parameters at the ith joint [27]. These parameters are θ_i, r_i, l_i, α_i. For physical interpretation of the parameters, see Fig. 8. Using these parameters, it is possible to define the transformation matrix between coordinate frames i and $i + 1$. This

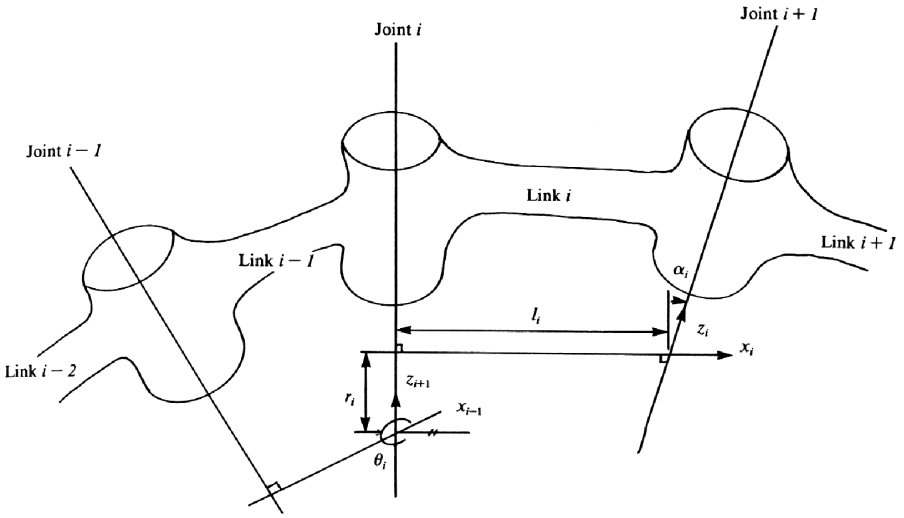


Figure 8. Kinematic parameters of a joint [27].

matrix, denoted by A_i, is given as:

$$A_i = \begin{bmatrix} C\theta_i & -S\theta_i C\alpha_i & S\theta_i S\alpha & l_i C\theta_i \\ S\theta_i & C\theta_i C\alpha_i & -C\theta_i S\alpha_i & l_i S\theta_i \\ 0 & S\alpha_i & C\alpha_i & r_i \\ 0 & 0 & 0 & 1 \end{bmatrix}. \tag{2}$$

After defining $A_1, \ldots, A_{14}$ for a 14 joint mechanism, one can define the transformation matrix, T_{14}, relating the position (location and orientation) of the 14-d.o.f. system in the world coordinate frame as:

$$T_{14} = A_1 * A_2 * \cdots * A_{14}. \tag{3}$$

As one can observe from (1) and (2) T_{14} depends on the 56 (4×14) kinematic parameters

$$\left\{ \theta_i, r_i, l_i, \alpha_i \;\Big|_{i=1\cdots14} \right\}_i$$

and, hence, errors in these parameters reflect in errors in T_{14}. We refer the reader to Paul [29] and Wu [28] to learn more about how to quantify those errors. In Wu [28], the author defines the errors in link parameters $\theta_i, r_i, l_i, \alpha_i$, as $\Delta\theta_i, \Delta r_i, \Delta l_i, \Delta\alpha_i$ respectively. These errors reflect in a differential change $\mathrm{d}A_i$ between two joint coordinates. Consequently, the relationship between the two coordinate systems is given by $A_i + \mathrm{d}A_i$ where A_i is defined in (2) and:

$$\mathrm{d}A_i = \frac{\partial A_i}{\partial \theta_i}\Delta\theta_i + \frac{\partial A_i}{\partial r_i}\Delta r_i + \frac{\partial A_i}{\partial l_i}\Delta l_i + \frac{\partial A_i}{\partial \alpha_i}\Delta\alpha_i. \tag{4}$$

Substituting (4) in (3) results an error in T_{14}, denoted as $\mathrm{d}T_{14}$ given by:

$$T_{14} + \mathrm{d}T_{14} = (A_1 + \mathrm{d}A_1) * \cdots * (A_{14} + \mathrm{d}A_{14}) = \prod_{i=1}^{14} (A_i + \mathrm{d}A_i), \tag{5}$$

where $\mathrm{d}T_{14}$ represents the total differential change at the end of the kinematic chain due to the 4×14 kinematic errors.

In order to calculate the kinematic error envelop of the end of the manipulator, the author assumes that $\Delta\theta$, Δr, Δl, $\Delta\alpha$ are independent 14-variables (as the number of joints), non-zero, normal distributions with $E[\Delta\theta] = E[\Delta r] = E[\Delta l] = E[\Delta\alpha] = 0$. Moreover, V_θ, V_r, V_l, V_α are each 14×14 diagonal covariance matrixes of $\Delta\theta$, Δr, Δl, $\Delta\alpha$, respectively, with components: $(\sigma^2_{\theta 1}, \ldots, \sigma^2_{\theta 14})$, $(\sigma^2_{r1}, \ldots, \sigma^2_{r14})$, $(\sigma^2_{l1}, \ldots, \sigma^2_{l14})$ and $(\sigma^2_{\alpha 1}, \ldots, \sigma^2_{\alpha 14})$ on their diagonal, respectively. From this, one can find that:

$$E\left\lfloor d^N \right\rfloor = M_1 E[\Delta\theta] + M_2 E[\Delta r] + M_3 E[\Delta l] + M_4 E[\Delta\alpha] = 0, \tag{6}$$

$$E\left\lfloor \delta^N \right\rfloor = M_2 E[\Delta\theta] + M_3 E[\Delta\alpha] = 0, \tag{7}$$

$M_{1,\ldots,4}$ are defined in Wu [28] and the resulting variances are:

$$V_\delta = M_2 V_\theta^{-1} M_2' + M_3 V_\alpha^{-1} M_3', \tag{8}$$

$$V_d = M_1 V_\theta^{-1} M_2' + M_2 V_r^{-1} M_2' + M_3 V_l^{-1} M_3' + M_4 V_\alpha^{-1} M_4'. \tag{9}$$

By using theses results, one can define the total translational and rotational errors, and their extreme values as a function of the kinematics error parameters resulting from manufacturing and assembly tolerances reflected in $E[\Delta\theta]$, $E[\Delta\alpha]$, $E[\Delta r]$, $E[\Delta l]$, $\sigma^2_{\theta,i}$, $\sigma^2_{\alpha,i}$, $\sigma^2_{r,i}$, $\sigma^2_{l,i}$.

We would like to point out that these parameters are not known, but can be evaluated by estimating the manufacturing tolerances; however, it is only reasonable that the designer be able to evaluate them. Moreover, it is also important to notice that the accuracy is configuration dependent, i.e. it depends on the current joint parameters and hence provides us with information on the instantaneous accuracy that can be taken into account in the control laws running the snake.

In order to impliment this method, we apply it while moving the robot's end-effector between sequential predetermined configurations. Table 1 summarizes the kinematic error parameters which characterize our robot design. These values were determined based on our best estimation of the manufacturing tolerances, and motor and joint accuracy. The parameters are given in Table 1 and the robot's Denavit–Hartenberg parameters are given later in Fig. 23.

Table 1.
Mean and variance estimated values for the HRR parameters (in mm and deg)

Parameter	θ_i	α_i	l_i	r_i
Mean $E[\Delta_{\text{parameter}}]$	0	0	0	0
Variance $V_{\text{parameter},i}$	$(0.0067)^2$	$(0.003)^2$	$(0.085)^2$	$(0.17)^2$

The results (Table 2) were obtained while the robot end-effector was following a predetermined trajectory defined by rotating, simultaneously, θ_{2i+1} from 0° to 30° (every second joint in θ degree) where $0° \leqslant \theta \leqslant 30°$ for $(i = 0, \ldots, 6)$. The resulting motion is a curving motion, starting with a straight line configuration (Fig. 9) and ending in a curve (Fig. 10). Table 2 summarizes the calculated error envelope for the starting and end configuration of the snake, in terms of angular and translational errors.

However, perhaps a more informative result would be the error magnitude, i.e. the norm of the translational error and rotational error, while following the

Table 2.
Resulting end-effector error envelop as shown in Fig. 5 (in mm and deg)

Parameter	dx^N	dy^N	dz^N	δx^N	δy^N	δz^N
Starting configuration	0.1	38.01	38.07	0.64	0.94	1.08
End configuration	13.4	9.3	8.6	0.82	0.94	0.95

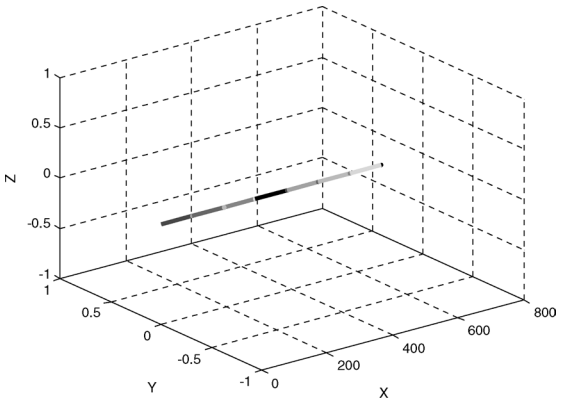


Figure 9. The snake starting in a horizontal configuration.

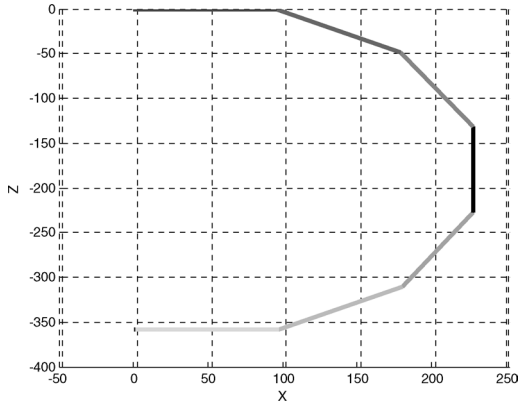


Figure 10. The snake ending in a curved configuration.

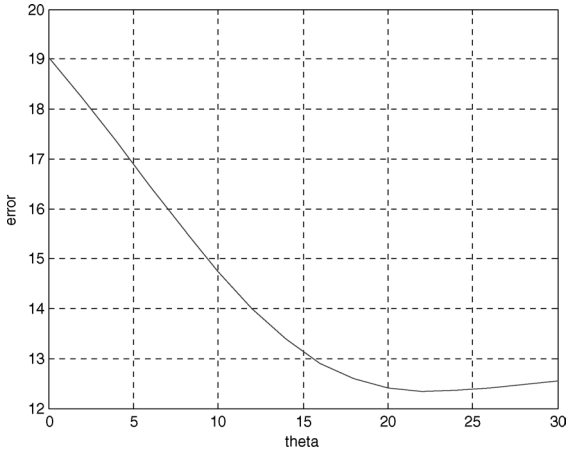


Figure 11. Translational error magnitude (in mm and deg).

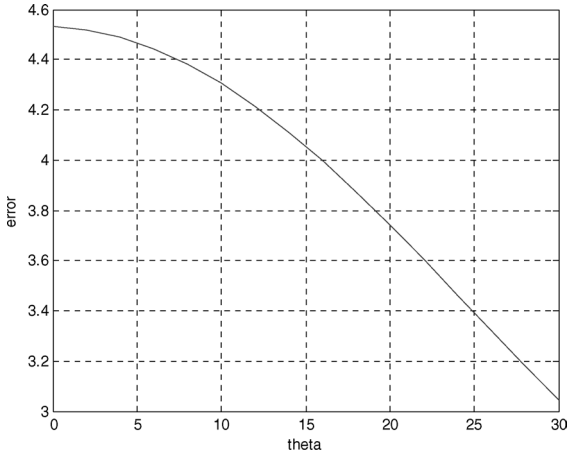


Figure 12. Rotational error magnitude (in mm and deg).

trajectory. For the simulation we took the error in each variable to be equal to 1 SD. Figures 11 and 12 present the results starting in a straight line configuration (0°) and ending in a curved configuration (Fig. 10). Also given in Figs 13 and 14 are the Frobenius norm of the covariance matrices V_θ and V_d, which represent the translational error envelope and the rotational error envelope, respectively. Note that the errors, both in translation and in rotation, receive an extreme value when in a straight configuration. This may be counter-intuitive, yet let us remember that this is a singular configuration of the robot, meaning configurations corresponding to singularities of the Jacobian matrix relating the robot's joint rates to the end-effector Cartesian velocities [30]. As the robot moves away from singularity, i.e. the joint angles increase, the translational and rotational error magnitudes decrease. The same holds for the Frobenius norm of the translational error, meaning not only

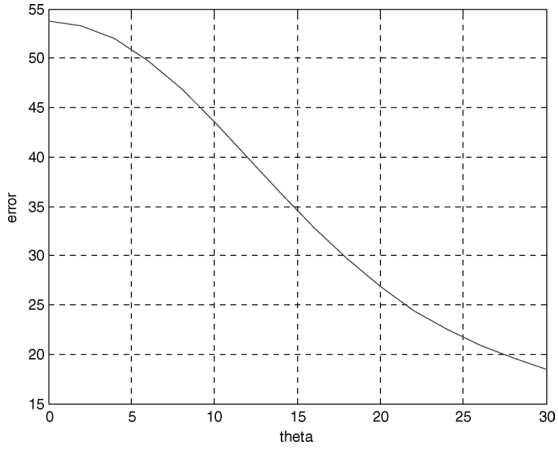


Figure 13. Frobenius norm of the covariance translational error matrix (in mm and deg).

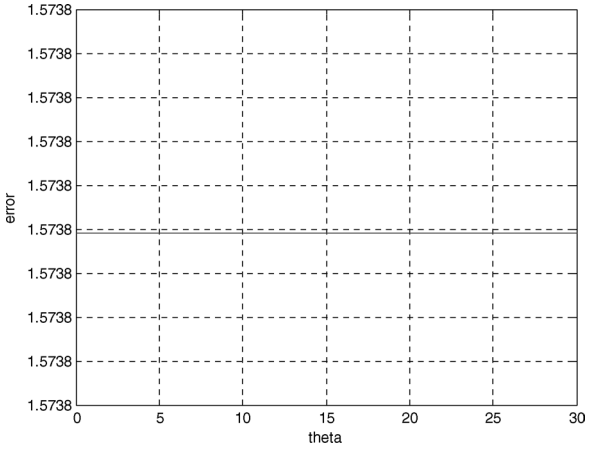
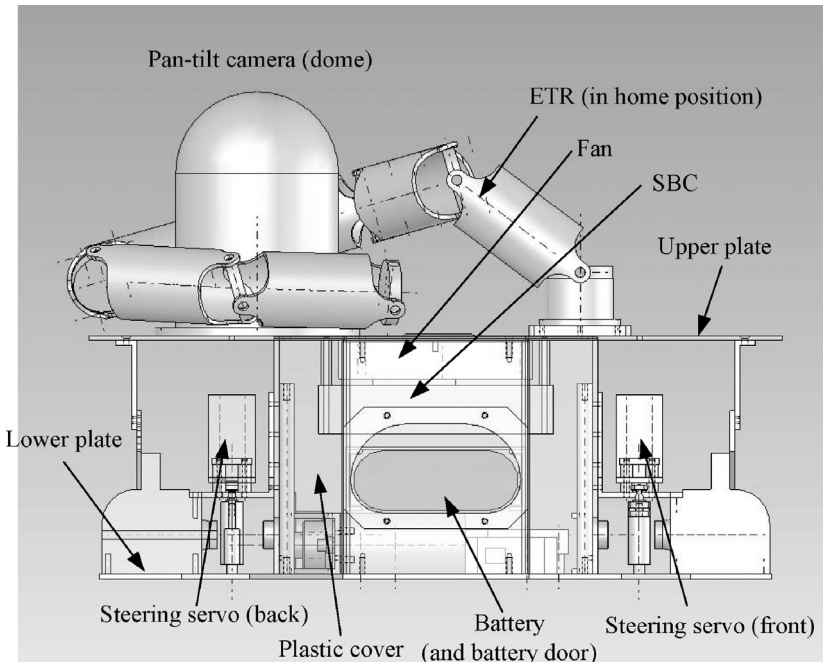


Figure 14. Frobenius norm of the covariance rotational error matrix (in mm and deg).

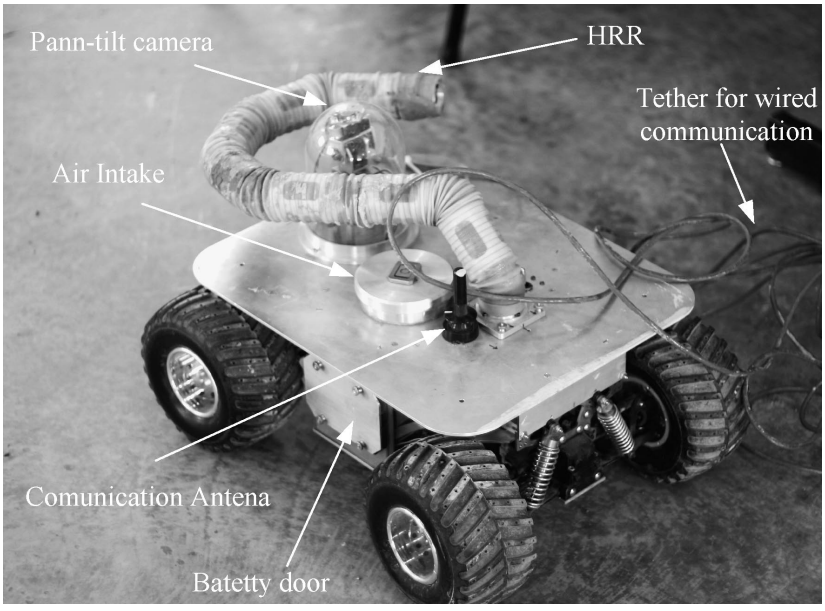
that the translational error is large, but also the error distribution gets the biggest value and the robot end-effector location can vary over a large range. On the other hand, while there is a change in the rotational error magnitude, the Frobenius norm of the covariance rotational error matrix is constant throughout the motion.

3.3. The mobile base

Although the mobile base is not the highlight of this work (the HRR is), a lot of thought was put in the development and adjustment of the mobile base to support the HRR system, and to be used in USAR. The mobile base is a four-wheeled drive system with four-wheel steering capability for high maneuverability. There are two main plates in the structure that are connected by a mechanical structure. The volume between the two plates is completely sealed by plastic cover walls to create



(a)



(b)

Figure 15. (a) Schematics of the mobile base and HRR. (b) The mobile base.

a protected and water proof volume. This volume incorporates all the low-level control parts, communication circuits, and power sources to run both the mobile base and the HRR (Fig. 15a and b). The base contains the on-board single-board computer (SBC) which is responsible for sending and receiving commands form the remote computer, and sending back sensor information to the remote computer. The communication can be either wireless using the communication antenna (Fig. 15b) or wired (Fig. 15b), according to the needs and environment. Moreover, the mobile base is equipped with a pan–tilt camera system which is protected by a dome. This camera is used for maintaining visual information while driving the mobile base. The pan–tilt system is capable of rotating in 180° × 360°, i.e. it can be flipped back when driving backwards. The HRR is mounted on the upper platform and wraps around the pan–tilt camera while is in its store configuration. The system is also equipped with a fan to remove heat from the electronic parts. The fan is installed under the upper plat and sucks air trough the air intake which is equipped with a filter to filter the air streamed trough the system. The air is exhausted trough the HRR. This has two advantages: (i) cooling the HRRs onboard controllers and (ii) the system maintains a positive pressure inside the HRR to prevent entry of dirt through its open end. As can be seen in Fig. 15b, the HRR is also covered by a flexible skin to protect its mechanical and electrical parts from dirt and water. The system parts, i.e. mobile base, HRR and pan–tilt camera, are all hose-able as required by search and rescue regulations.

4. SYSTEM CONTROL

4.1. Low-level control

Hard wiring to all 14 actuators and encoders would require 84 (14 × 6) conductors to run inside the HRR envelope. This solution was deemed unfeasible; hence, it was decided to use an I^2C control bus [31] (by Philips) in order to reduce the number of wires by connecting micro-controller circuit boards placed inside the HRR (Fig. 16). Although I^2C technology is an available technology, packaging the required components (H-bridge, decoder chip, PIC microcontroller plus passive components) to fit within the link envelope and providing interconnects between controllers are both challenging problems.

The current electronics design includes one custom circuit board per actuator (two per link). The boards are mounted on the motors and are connected by a data bus. This attempts to maintain electronic modularity at the ‘link’ level and allows disassembly of the links at the crosses without desoldering wires. Each circuit board includes a Microchip PIC16F876 microcontroller, an LSI/CSI LS7166 24-bit (quadrature) counter, an Allegro A3953 H-bridge amplifier IC, a linear voltage regulator for logic power supply and several passive components. The overall physical size of the board is 36 mm × 23 mm (Fig. 17). The ‘bus’ that runs through the robot consists of a total of seven wires: two supply voltages (approx. 14 and approx. 7 V), separate grounds for each voltage, I^2C communication lines (SCL; SDA), and a camera

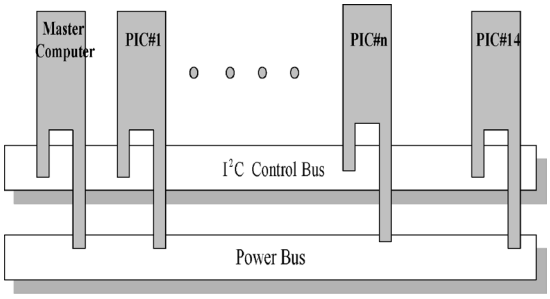


Figure 16. I^2C control bus.

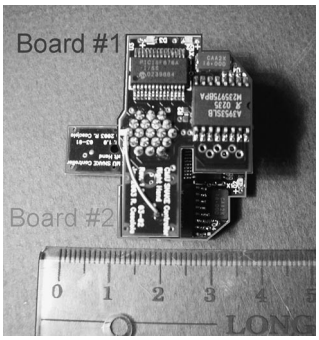


Figure 17. Two PIC boards for two motors.

signal and ground for it. The firmware running on the PIC microcontroller includes support for the I^2C protocol, in both send and receive mode. Moreover, the code also performs PID control loop for the motor, connected to it, locally. For this use, the PIC reads the joint angle (quadrature encoder counter) and drives the H-bridge amplifier with a PWM signal according to the data received on the I^2C bus. It also has the ability to communicate with the host computer over the bus and provides, upon request, the current joint angle. The I^2C bus runs at 120 kb/s which is used for a 50-Hz data update rate on the bus. The PID control loop on board the PIC microcontroller runs at a 1 kHz rate. This frequency is limited by the current microchip that is being used and can be increased using a different model. Both the software and the control parameters can be updated over the I^2C bus.

4.2. High-level control and user interface

The command interface for the HRR system can be teleoperated either using a peer-to-peer network or over the internet. We refer the reader to Fig. 18 for a block diagram of the system. The HRR system contains a camera mounted on the end-effector. One of the key requirements for the system was providing a low-latency, high-frame-rate video from the HRR camera. This was made possible using the Java Media Framework (JMF) to send video capture streams over RTP. JMF also provided a convenient way to add image rotation through the use of a custom codec.

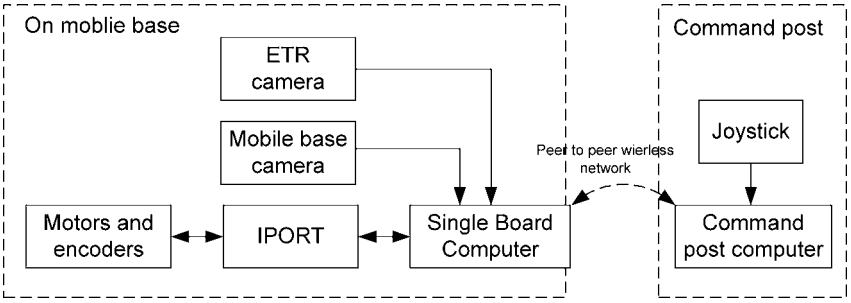


Figure 18. System block diagram.

Figure 19. HRR camera: a 'free-flying' camera.

Image rotation helps to prevent the operator from becoming disoriented due to HRR orientation and will be discussed later. Basically, the camera operates like a 'free-flying camera' (Fig. 19). With the current rotation scheme, the operator can trust that a vertical line in the image is in fact vertical in the real world. The rotation angle calculation depends on the HRR instantaneous configuration and is described in the next sections.

The on-board computer which captures video from the HRR camera compresses the raw video streams into JPEG over RTP streams using JMStudio, which is part of JMF. This base computer also hosts a server which receives HRR motion instructions from the command station (Figs 20 and 21). The command station sends the desired joint angles to this server, which then delivers this information over the I^2C bus to the local controllers. The signals to the motors and from the encoders are transferred over the I^2C control bus using IPORT™ as an interface. IPORT™ is a hardware-based 120 kb/s full-speed I^2C bus interface featuring the I^2C master and slave mode. The command interface uses a joystick to control the mobile base, a pan–tilt camera and the the motion of the HRR (Fig. 21). Moreover, the operator can pick between the pan–tilt camera image or the HRR camera image to be streamed back and displayed on the screen. Furthermore, the command station

Figure 20. User interface and control joystick.

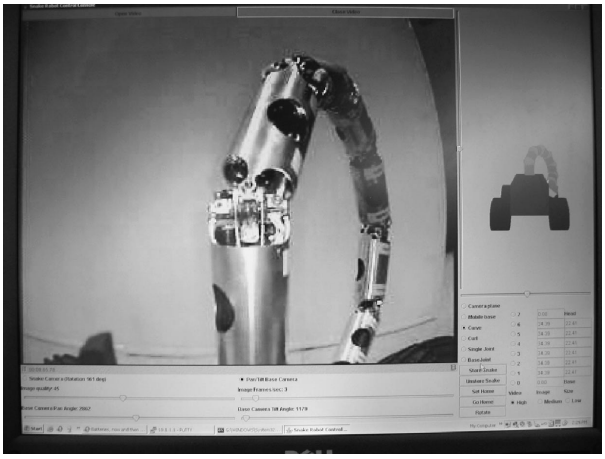


Figure 21. High-level control and user interface, and a 3D graphic animation of the robot (circled in yellow).

also displays a 3D animation of the robot (Fig. 21) — this animation is dynamic and is being updated as the robot moves (Fig. 21).

As mentioned, the PID control loop runs on the local PIC microcontroller. The command post computer reads the joystick position and system configuration, and accordingly sends a packet to the on-board computer, which contains commands addressed to one of the 14 motors driving the robot or the four servos driving the mobile base system. When receiving a packet on the local network, the SBC sends a position command, over the I^2C control bus, to the addressed microcontroller. The local PID control loop then generates a value to the local HRR motor which represents joint velocity; this value is then translated, on-board the microcontroller, to a PWM signal to drive the motor. In the case that the packet is addressed to one of the mobile-base servos, the local PIC microcontroller does not initiate a PID

control loop, but generates a pulse whose length is proportional to the required servo position.

4.3. Image representation

One of the most challenging issues for all search devices that carry a camera in their end-effector, including HRRs, is on-line image interpretation. One can see how this is both difficult and critical. Imagine a situation where the robot is being used for urban search and rescue, and its camera detects a survivor trapped under or perhaps on top of a large metal beam (Fig. 22a and b). Knowing which interpretation is correct immediately impacts on the decisions of the rescue workers [32]. As the camera is mounted at the end-effector of a HRR (Fig. 19), the camera is oriented in space as a function of the joint parameters and thus the instantaneous image is being constantly rotated. In this section, we present an algorithm that solves this problem by resolving the angle of rotation of the image as a function of the robot instantaneous configuration. Using this angle, the image is then rotated on-line and the viewer is provided with an image as if it were taken by a human eye, such that a vertical line in the real world will appear also as vertical in the image.

Given an instantaneous configuration of the HRR, i.e. a set of kinematic parameters of the robot's joints, one can derive the transformation matrix, ${}^0A_{\mathrm{T}}$, from world coordinate frame (WCF) to tool coordinate frame (TCF), which is, in our case, also the camera coordinate system such that the image is facing the positive $\hat{Y}_{\mathrm{T}}$ direction. Figure 23 describes the kinematic skeleton of the HRR and the local coordinate frames used for the Denavit–Hartenberg method. Given the Denavit–Hartenberg parameters, one can calculate the ${}^{i-1}A_i$ homogeneous transformation matrices between coordinate frames i and $i-1$, and then calculate ${}^0A_{14}$ as given in (2):

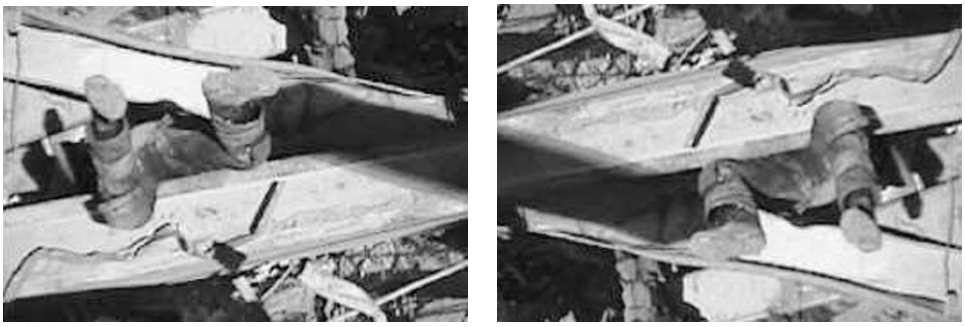
$$T_N = {}^0A_{14} = \prod_{i=1}^{14} {}^{i-1}A_i = \begin{bmatrix} R_{3\times3} & p_{3\times1} \\ 0_{1\times3} & 1_{1\times1} \end{bmatrix}. \tag{10}$$

Let us define the vectors $\hat{X}_{14}, \hat{Y}_{14}, \hat{Z}_{14}$ in coordinate system 14 (tool coordinate system: T). $\hat{X}_{\mathrm{T}}, \hat{Y}_{\mathrm{T}}, \hat{Z}_{\mathrm{T}}$ can be expressed in the WCF, i.e. coordinate system 0, as:

$$\begin{aligned} {}^0\hat{X}_{\mathrm{T}} &= [x_1, x_2, x_3] = R \cdot \hat{X}_{\mathrm{T}}, \\ {}^0\hat{Y}_{\mathrm{T}} &= [y_1, y_2, y_3] = R \cdot \hat{Y}_{\mathrm{T}}, \\ {}^0\hat{Z}_{\mathrm{T}} &= [z_1, z_2, z_3] = R \cdot \hat{Z}_{\mathrm{T}}. \end{aligned} \tag{11}$$

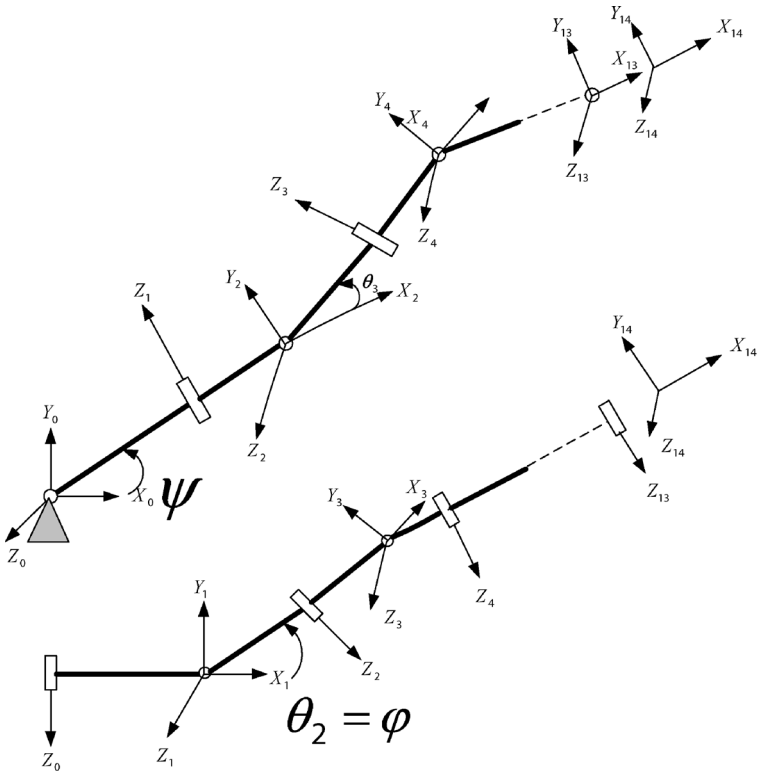
Also, let us define a plane, the P-plane, which is normal to the ground and contains $\hat{X}_{\mathrm{T}}$ (Fig. 24), a normal vector, $\hat{N}$, to P, (in WCF) has its y-component equal to zero and is given by:

$$\hat{N} = [n_x, n_y, n_z] = \frac{\left[\dfrac{-n_z \cdot x_3}{x_1}, 0, n_z\right]}{\left\|\left[\dfrac{-n_z \cdot x_3}{x_1}, 0, n_z\right]\right\|}. \tag{12}$$



(a) (b)

Figure 22. Two different scenarios: (a) a metal beam on a survivor and (b) the survivor is on the metal beam.



i	α_i	a_i	d_i	θ_i
1	$-\pi/2$	l	0	θ_1
2	$\pi/2$	0	0	θ_2
3	$-\pi/2$	l	0	θ_3
4	$\pi/2$	0	0	θ_4
5	$-\pi/2$	l	0	θ_5
6	$\pi/2$	0	0	θ_6
7	$-\pi/2$	l	0	θ_7
8	$\pi/2$	0	0	θ_8
9	$-\pi/2$	l	0	θ_9
10	$\pi/2$	0	0	θ_1
11	$-\pi/2$	l	0	θ_1
12	$\pi/2$	0	0	θ_1
13	$-\pi/2$	l	0	θ_1
14	0	C	0	0

Figure 23. Kinematic skeleton of the HRR and its Denavit–Hartenberg parameters.

Substituting $n_z = 1$ in (12), one can determine $\hat{N}$ for every given instantaneous configuration of the robot.

Given that the camera is mounted on the HRR such that, in the home position, the image is facing the positive $\hat{Y}_{\mathrm{T}}$ direction, we want $\hat{Y}_{\mathrm{T}}$ to be contained in the P-plane

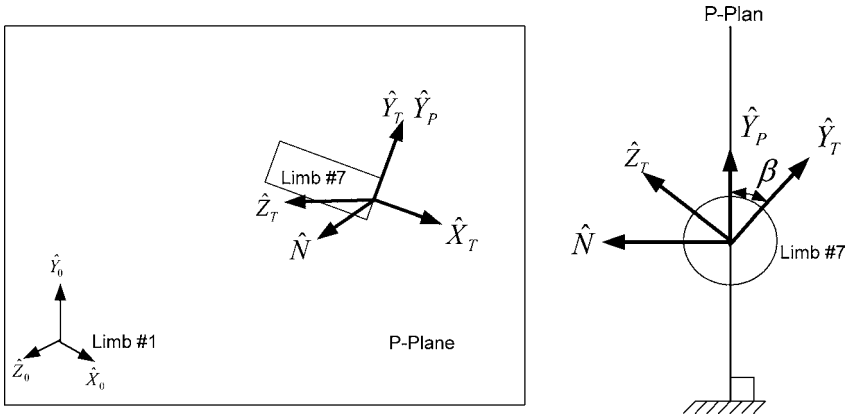


Figure 24. Tool coordinate frame.

such that the image is always facing the 'up' direction and is horizontal. In order to accomplish this, one needs to rotate the image β degrees (Fig. 24) with respect to $\hat{X}_{\mathrm{T}}$ such that both $\hat{Y}_{\mathrm{T}}$ and $\hat{Y}_{\mathrm{P}}$ are collinear. Let us define $\hat{Y}_{\mathrm{P}}$, a vector in P and normal to ${}^{0}\hat{X}_{\mathrm{T}}$, such that:

$$\hat{Y}_{\mathrm{P}} = [yp_1, yp_2, yp_3] = \hat{N} \times {}^{0}\hat{X}_{\mathrm{T}}, {}^{0}\hat{X}_{\mathrm{T}} \times \hat{N} \mid yp_2 > 0. \tag{13}$$

Observing Fig. 24 and (12) and (13) one can detect that $\hat{Y}_{\mathrm{P}} \perp \hat{X}_{14,0}$ and ${}^{0}\hat{Y}_{\mathrm{T}} \perp {}^{0}\hat{X}_{\mathrm{T}}$; hence, one can rotate ${}^{0}\hat{Y}_{\mathrm{T}}\beta$ degrees, with $\hat{X}_{\mathrm{T}}$ as the axis of rotation, such that it is collinear with $\hat{Y}_{\mathrm{P}}$, where β is given by:

$$\beta = a\cos\left({}^{0}\hat{Y}_{\mathrm{T}} \cdot \hat{Y}_{\mathrm{P}}\right). \tag{14}$$

In the case that (14) reveals $\beta < 0$ $\beta = (\pi/2) - \beta$. An example of the effect of this algorithm is given in Fig. 25.

4.4. Motion control

Motion planning for a HRR, in general, has always been a great challenge, due to the redundancy of joint parameters and limbs one has to control in order maneuver in a confined 3D environment. This issue is a greater challenge when motion planning takes place in real-time, while driving a real system. The main challenge here is to have an algorithm that can be implemented without introducing much of a lag to the system.

Curving/curling of the HRR is performed in the P-plane given in Fig. 24. When curving, the robot forms a parabolic shape (Fig. 26). When curling, each limb is rotating separately until reaching its limit and only then does the successive limb follows the same motion.

The P-plane can then be oriented 360° with respect to $\hat{y}_0$ of Fig. 23, resulting in a hemispherical work-envelope for the HRR. The HRR is also capable of moving in the local end-effector coordinate frame, i.e. the image coordinate frame. This mode

Figure 25. An image taken by the camera on the HRR before rotation (left) by β and after (right).

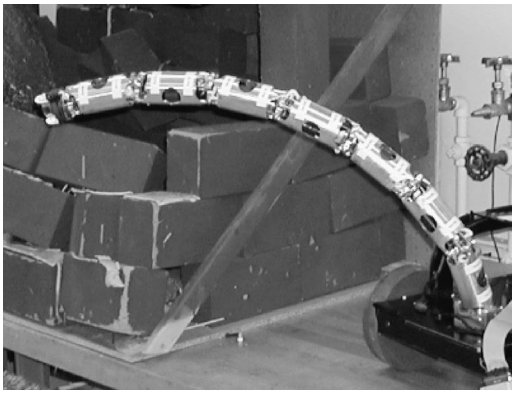


Figure 26. The snake robot curving.

of motion is used when using the image received from the camera in order to steer the head of the robot. We do this by calculating the Penrose pseudo-inverse of the robot's Jacobian matrix J.

Given that:

$$\dot{q} = J^{-1}\dot{x}, \tag{15}$$

where $\dot{x}$ is the end-effector Cartesian velocities and $\dot{q}$ is the robot's joint rates, one needs to calculate J^{-1}. We do this by calculating the Penrose pseudo-inverse of J such that:

$$\dot{q} = J^{\mathrm{T}}\left(JJ^{\mathrm{T}}\right)^{-1}\dot{x}. \tag{16}$$

Observing (16), one can see that we calculate the inverse of JJ^{T}. Given that J has the dimension of 6×14 (6 rows by 14 columns), we invert JJ^{T} which has the dimension of 6×6; this is an advantage as we need to calculate J^{-1} in real-time. In order to move the robot end-effector in the camera frame we do the next steps (starting in step i):

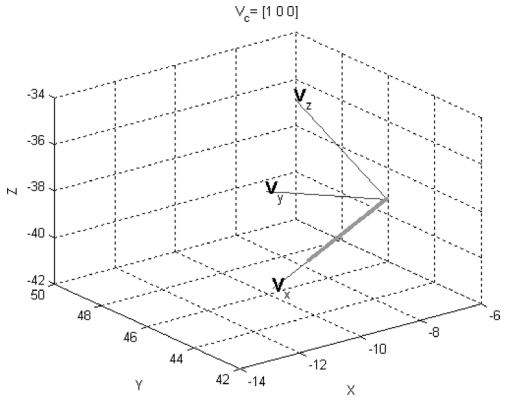


Figure 27. Motion in $\hat{V}_{\mathrm{c}} = [1, 0, 0]$ direction.

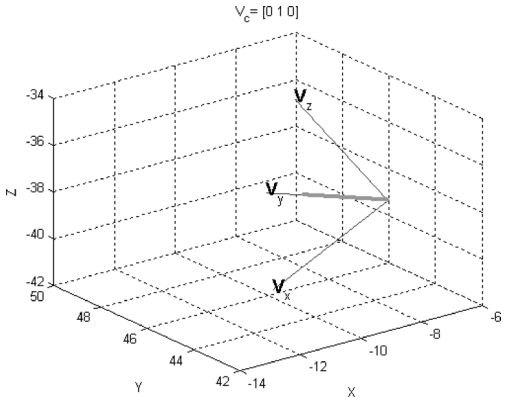


Figure 28. Motion in $\hat{V}_{\mathrm{c}} = [0, 1, 0]$ direction.

(i) Define the motion direction, $\hat{V}_{\mathrm{c}}$, in the camera coordinate frame, by rotating the end-effector coordinate system by β degrees to be collinear with the camera axis and then define $\hat{V}_{\mathrm{c}}$ in the rotated system.

(ii) Write $\hat{V}_{\mathrm{c}}$ in the world coordinate frame, i.e. coordinate frame 0: ${}^0\hat{V}_{\mathrm{c}} = T^{\mathrm{R}}_{N,i} \cdot \hat{V}_{\mathrm{c}}$, where $T^{\mathrm{R}}_{N,i}$ is the rotational part of $T_{N,i}$ in step i (10).

(iii) Calculate J_i^{-1} given configuration vector q_i.

(iv) Define $\dot{x}_{i+1}$ as $\dot{x}_{i+1} = [v_x\hat{V}_x, v_y\hat{V}_y, v_z\hat{V}_z, \omega_x\hat{V}, \omega_y\hat{V}_y, \omega_z\hat{V}_z]$, where v_x, v_y, v_z, ω_x, ω_y, ω_z are the instantaneous linear and angular velocities in the $\hat{V}_x$, $\hat{V}_y$, $\hat{V}_z$ directions, respectively.

(v) Calculate $\dot{q}_{i+1}$ as $\dot{q}_{i+1} = J_i^{-1}\dot{x}_{i+1}$ using (16).

(vi) Calculate q_{i+1} as $q_{i+1} = q_i + \dot{q}_{i+1} \cdot \Delta t$, where Δt is a predefined time step.

(vii) Go to Step (i).

Figures 27–30 demonstrate a motion in the x-, y- and z-directions, respectively, and the bold dots represents the origin of the end-effector coordinate system. As

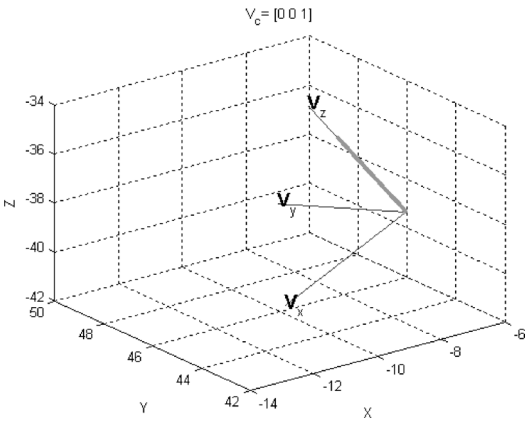


Figure 29. Motion in $\hat{V}_c = [0, 0, 1]$ direction.

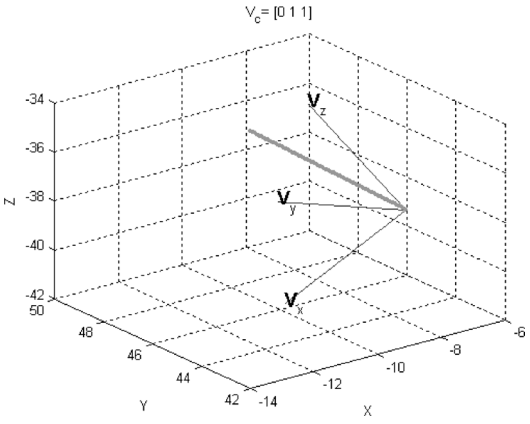


Figure 30. Motion in $V_c = [0, 1, 1]$ direction.

one can detect, the robot performs a motion in each of the camera frame directions. It is also possible to perform motion in any other direction which is defined in image coordinate systems. This motion is demonstrated in Fig. 30 where the desired motion is along a vector in the yz-plane forming a 45° angle with each of the axes.

5. CONCLUSION

In this work we presented a novel search and rescue robotic system that is composed of a HRR which is mounted on a mobile base. This allows us to extend the reach ability of the mobile base by deploying the HRR and using its capabilities to inspect unreachable areas such as small cracks, and pipes, using the camera and sensors mounted on its end-effector. The HRR has 14 d.o.f., yet it is still compact and strong enough to allow the robot to cantilever-lift itself. We base our design on rigorous analysis based on a kinematic analysis technique that allows us to

evaluate the accuracy of the HRR. The robot accuracy is affected by errors related to motor accuracy and manufacturing accuracy. These errors which can be evaluated, *a priori*, are taken into account in the forward kinematic calculations that indicate the expected accuracy of the robot as a function those errors. In the future, this data can be integrated into the robot's control laws in order to compensate for these errors and increase the robot's accuracy.

We also presented the low-level system control which is based on a novel, high-speed, I^2C data bus technology. This enabled us to maintain a compact structure. The system is composed of 14 embedded mini-circuit boards, each containing a microchip microcontroller, a driver and a counter, which run the PID control loop for each motor.

The mechanism and integration, again, are not enough to construct a successful HRR system — it requires control and path planning. This paper presents the beginning of such work. The high-level control algorithms described in this paper and integrated in the system implement kinematic methods to do on-line motion control both in the global coordinate frame and in the camera coordinate frame which is mounted on the robot end-effector. Both methods are useful depending upon the situation and the performance of the person maneuvering the robot. The motion control algorithms allow the user easy steering of the head of the HRR and the rest of its body, taking advantage of the high maneuverability and redundancy of the system. Future work will include developing more 'automated' path planners where the user simply specifies a surface to be inspected, and the planner automatically plans a path for both the head of the robot as well as its internal d.o.f. to maneuver the head, as desired.

The user interface described in this paper was designed to control the HRR. It allows the user full control and interpretation of the data received from the camera as well as easy operation using a single joystick. The system runs from a remote computer on a local wireless network and can also run over the Internet. The command interface uses a single joystick to control the motion of the HRR. It allows the user to switch between several modes of motion including motion in the camera coordinate system. Moreover, the image received from the camera which is mounted on the snake's end-effector is displayed and updated in real-time, such that it is always horizontal. The user can also get a sense about the current robot configuration from a 3D graphics display of the robot. This display is also updated in real-time.

This system has been deployed in several USAR scenarios. Our first experience was in a collapsed building in August 2003 in Lebanon, Indiana, USA. The deployment was part of an exercise organized by the Center for Robotic Assisted Search and Rescue and the Indiana Task Force I. Our second experience was in early 2004 with the New Jersey Task Force 1, and our latest experience was in May 2004 in NASA Ames USA USAR facilities. Although our device performed quite well in all scenarios, we learned that there are still many features that the system requires, such as streaming back audio signals, and hardening the steering

and driving mechanisms. Consequently, we are constantly updating our device. Furthermore, We believe that a HRR will have many applications, beyond USAR. Currently, we are working with the Department of Energy and the Office of Naval Research to develop a HRR to inspect underground tanks and engines on naval vessels.

REFERENCES

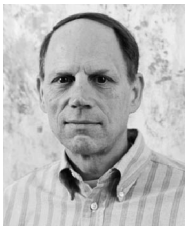
1. US Fire Administration and National Fire Association, *Rescue Systems I* (1993).
2. J. Casper, M. Micire and R. Murphy, Issues in intelligent robots for search and rescue, *SPIE Ground Vehicle Technol. II* **4**, 41–46 (2000).
3. R. Murphy, Marsupial and shape-shifting robots for urban search and rescue, *IEEE Intelligent Syst.* **15**, 14–19 (2000).
4. R. Murphy and J. J. Martinez, Lessons learning from the USF REU site grant: multiple autonomous mobile robots for search and rescue applications, in: *Frontier in Education* (1997).
5. S. Hirose, Snake, walking and group robots for super mechano-system, in: *IEEE SMC'99 Conf. Proc.*, pp. 129–133 (1999).
6. A. Kobayashi and K. Nakamura, Rescue robots for fire hazards, in: *Proc. Int. Conf. on Advanced Robotics*, pp. 91–98 (1983).
7. H. Amano, K. Osuka and T. Tran, Development of vertically moving robot with gripping handrails for fire fighting, in: *Proc. Int. Conf. on Intelligent Robots and Systems*, pp. 661–662 (2001).
8. A. Castano, W. M. Shen and P. Will, CONRO: toward deployable robots with inter-robot metamorphic capabilities, *Autonomous Robots* **8**, 309–324 (2000).
9. R. M. Voyles, TerminatorBot: a robot with dual-use for manipulation and locomotion, in: *Proc. Int. Conf. on Robotics and Automation*, pp. 61–66 (2000).
10. J. S. Jennings, G. Whelan and W. F. Evans, Cooperative search and research with a team of mobile robots, in: *Proc. 8th Int. Conf. on Advanced Robotics*, pp. 193–200 (1997).
11. R. Murphy, J. Casper, J. Hyams and M. Micire, Mixed-initiative control of multiple heterogeneous robots for search and rescue, Technical Report, University of South Florida (2002).
12. R. Masuda, T. Oinuma and A. Muramatsu, Multi-sensor control system for rescue robot, in: *Proc. IEEE/SICE/RJS Int. Conf. on Multisensor Fusion and Integration for Intelligent Systems*, pp. 381–387 (1996).
13. G. S. Chirikjian and J. W. Burdick, A modal approach to hyper-redundant manipulator kinematics, *IEEE Trans. Robotics Automat.* **10**, 343–354 (1994).
14. S. Hirose, *Biologically Inspired Robots: Snake-like Locomotors and Manipulators*. Oxford University Press, Oxford (1993).
15. W. M. Kier and K. K. Smith, Tongues, tentacles, and trunk: the biomechanics of movement in Muscularhydrostas, *Zool. J. Linnean Soc.* **8**, 307–324 (1985).
16. M. A. Hannan and I. D. Walker, A novel 'elephant's trunk' robot, in: *Proc. IEEE/ASME Int. Conf. on Advanced Intelligent Mechatronics*, Atlanta, GA, pp. 410–415 (1999).
17. G. S. Chirikjian and J. W. Burdick, Kinematically optimal hyper-redundant manipulator configurations, *IEEE Trans. on Robotics Automat.* **11**, 794–806 (1995).
18. http://robotics.stanford.edu/users/mark/bio.html (06/2003).
19. G. Miller, Snake robots for research and rescue, in: *Neurotechnology for Biomimetic Robots*, J. Ayers, J. L. Davis and A. Rudolph (Eds), pp. 271–284. MIT Press, Cambridge, MA (2002).
20. http://www.doctorgavin.com/ (06/2003).

21. G. L. Haith, H. Thomas and A. Wright, A serpentine robot for planetary and asteroid surface exploration, oral presentation at: *4th IAA Int. Conf. on Low-Cost Planetary Missions*, Laurel, MD (2000).
22. H. Ikeda and N. Takanashi, Joint assembly movable like a human arm, US Patent No. 4,683,406 (1987).
23. N. Takanashi, K. Aoki and S. Yashima, A gait control for the hypre-redundant robot O-RO-CHI, in: *Proc. ROBOMEC'96*, Ube, pp. 79–80 (1996).
24. http://technology.jpl.nasa.gov/gallery/techGallery/gallery/gl_pages/P44487.html (12/2003).
25. R. Cieslak and A. Morecki, Elephant trunk type manipulator — a tool for bulk and liquid materials transportation, *Robotica* **17**, 11–16 (1999).
26. S. Ma, H. Hirose and H. Yoshinada, Development of a hyper-redundant multijoint manipulator for maintenance of nuclear reactors, *Advanced Robotics* **9**, 281–300 (1995).
27. J. Denhavit and R. S. Hartenberg, A kinematic notation for lower-pair mechanisms based on matrices, *ASME J. Appl. Mech.* 215–221 (1955).
28. C. H. Wu, A kinematic CAD tool for the design and control of a robot manipulator, *Int. J. Robotics Res.* **3**, 58–67 (1984).
29. R. Paul, *Robot Manipulators: Mathematical Programming and Control*. MIT Press, Cambridge, MA (1981).
30. M. J. D. Hayes, M. L. Husty and P. J. Zsomnar-Murray, Singular configurations of wrist-partitioned 6R serial.
31. http://www.semiconductors.philips.com/buses/i2c/ (05/2004).
32. Robots: a geometric perspective for users, *Trans. CSME* **26**, 41–55.

ABOUT THE AUTHORS

Alon Wolf is Co-director of the Biorobotics and Biomedical Robotics lab in Carnegie Mellon University. He also holds an Adjunct Faculty Appointment at the Robotics Institute in Carnegie Mellon University and a Research Faculty Appointment at the Institue for Computer Assisted Orthopaedic Surgery at Western Pennsylvania hospital. He received his PhD from the Mechanical Engineering department at the Technion Israel Institute of Technology. His research areas focus on kinematics, design and control of mechanisms, biomechanics, and medical robotics. In his work he elaborates and develops fundamental kinematics tools which are based on screw theory and line geometry for the study of mechanisms, and especially parallel robots.

Howard H. Choset is an Associate Professor of Mechanical Engineering and Robotics at Carnegie Mellon University where he conducts research in motion planning and design of serpentine mechanisms, coverage path planning for demining and painting, mobile robot sensor-based exploration of unknown spaces, and education with robotics. In 1997, the National Science Foundation awarded him its Career Award to develop motion planning strategies for arbitrarily shaped objects. In 1999, the Office of Naval Research started supporting him through its Young Investigator Program to develop strategies to search for land and sea mines. Recently, the MIT Technology Review elected him as one of its top 100 innovators in the world under 35. He directs the Undergraduate Robotics Minor at Carnegie Mellon and teaches an overview course on Robotics which uses a series of custom-developed Lego Labs to complement the course work. His students have won best paper awards at the RIA in 1999 and ICRA in 2003. He is a member of an urban search and rescue response team using robots with the Center for Robot Assisted Search and Rescue.



H. Benjamin Brown, Jr. is a Project Scientist in the Robotics Institute at Carnegie Mellon University. He received his BS and MS degrees in Mechanical Engineering from Carnegie Mellon in 1967 and 1976. After 12 years in various industry R&D and design-related positions, he came to the Robotics Institute in 1981. He began working in the Leg Lab headed by Marc Raibert, and was primarily responsible for the mechanical design of a number of dynamically stabilized, legged, running robots of unequaled performance. He is currently pursuing the development of dynamic hopping and running robots that exploit elastic energy storage in the legs, and snake robots for search-and-rescue applications. He is interested in the analysis, design and control of robots and electro-mechanical systems, and specializes in the development of high-performance structures and devices, dynamically stabilized robots in particular. He has two patents and has authored or co-authored approximately 40 conference and journal papers, mainly in the field of mobile robotics.

Randall W. Casciola is VP Engineering for AugmenTech, Inc., Pittsburgh, PA. After receiving his BS degree in Electrical Engineering at Carnegie Mellon University in 1991, he joined the Robotics Institute at Carnegie Mellon as a Research Engineer where he designed and built the electronics systems for several robots, including the Self Mobile Space Manipulator (SM2), the DOE-funded Reconfigurable Modular Manipulator System (RMMS) and the Gyrover. In 1995, he became a Senior Research Engineer at the Engineering Design Research Center (EDRC), also at Carnegie Mellon. There, he was responsible for the electronics design of several wearable computers, including the Navigator 2, VuMan 3R, and the DARPA-funded Language Translator and Speech Recognizer Smart Modules. There he was also a Co-instructor for an interdisciplinary course in rapid prototyping. In 1998, he left Carnegie Mellon to join AugmenTech, Inc., where he is developing electronic and software systems to monitor the care given to nursing home residents. He is interested in digital and analog circuit design, especially as related to microprocessor controlled, ultra-miniature micropower systems. He is also interested in RF, non-contact position/moisture sensing and has several patents pending in this area.