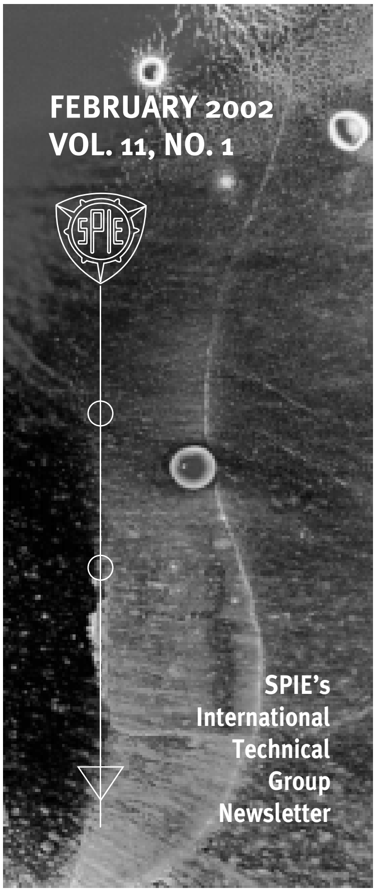

Robotics and Machine Perception

Calendar
—See page 2

Technical Group Registration Form *—See page 10*

Newsletter now available on-line

Technical Group members are being offered the option of receiving the Robotics and Machine Perception Newsletter in an electronic format. An e-mail notice is being sent to all group members advising you of the web site location for this issue and asking you to choose between the electronic or printed version for future issues. If you have not yet received this e-mail message, then SPIE does not have your correct e-mail address in our database. To receive future issues of this newsletter in the electronic format please send your e-mail address to spie-membership@spie.org with the word ROBOTICS in the subject line of the message and the words "Electronic version" in the body of the message.

If you prefer to continue to receive the newsletter in the printed format, but want to send your correct e-mail address for our database, include the words "Print version preferred" in the body of your message.

http://spie.org/web/techgroups/robotics/pdfs/

SAIL: A "mentally" developing robot

The conventional process for robot development is not autonomous—the human programmer must provide a task-specific representation, which relies on how the human programmer understands the task. Even if machine learning is used during this manual developmental process, learning requires the adjustment of some task-specific parameters, designed by human programmer, using sensory data. Thus, such a robot still cannot autonomously derive representations for other tasks, especially in a changing environment.

This task-specific engineering paradigm has met fundamental problems in dealing with complex robot tasks, such as vision-guided navigation and object manipulation, in unknown or unpredictable human environments. The various human-designed representations for vision that have been experimented with in the past are too static to deal with changing environments, and are brittle in performing complex perception-related tasks.

The SAIL robot project is motivated by human autonomous mental development. It is well known that a human child gradually makes sense of its environment by interacting with it through its sensors and effectors.[1] Recent studies of brain plasticity have shown that auditory cortex that receives visual signals right after birth can generate a representation that is found in visual cortex and, further, such a rewired auditory cortex can perform visual tasks.[2] Is it advantageous to enable robots to automatically develop its mental skills, including representation? Some discussion of the related issues were recently presented in an article in Science.[3]

Figure 1. This human-size robot, called SAIL, was made in-house at Michigan State University.

SAIL Developmental Robot

Since early 1995,[4] we have been working on the SAIL robot (short for Self-organizing, Autonomous, Incremental Learner: see Figure 1) and its predecessor (SHOSLIF).[5] The goal of the SAIL project is to automate the process of mental development in robots: however, we do not intend to faithfully emulate biology, which would be impractical at our current stage of knowledge.

A developmental robot needs to learn different tasks through online, real-time interactions with the environment (including humans) without any need to switch the mode of the program. The SAIL developmental algorithm has some "innate" reflexive behaviors built-in. At the "birth" of the robot, this algorithm starts to run. Humans train the robot by interacting with it, in very much the way that human parents interact with their infants. Table 1 outlines the major characteristics of existing approaches to constructing an artificial system, and the new developmental approach.

Sensory signals from each channel first enter a sensory mapping, which derives features (a subset of the overall representation) using incremental principal component analysis (IPCA), thus reducing the dimensionality of the space.[6] Each sensory mapping also has an attention effector that turns signals on or off for transmission to the later cognitive mapping. The attention effector is internal, in that it carries out an internal action. This new concept is at variance with the traditional model where learning agents only interact with the external

continues on p. 9

Editorial

Welcome to our first issue of the R&MP newsletter for 2002.

In this issue we include a set of articles that bring together a diverse range of advanced themes in robotics.

SAIL (Weng et. al.), for example, integrates a wide range of robotics and artificial intelligence techniques with the goal of creating a composite, multi-functional, intelligent robot system that follows a learning, or developmental pattern, modeled on human development. Three articles focus on the traditional theme of robot programming, but within the more recent area of cooperative multirobot systems. Two of these (Bredenfeld and Das) provide exemplars of software systems for programming robot teams, illustrated with applications to the popular RoboCup **(http://www.robocup.org)** competition. The third (Sugar) illustrates decentralised control and programming of cooperative mobile manipulators for transportation tasks.

The paper by Belousov & Clapworthy develop the programming theme further, in the area of Internet robots, demonstrating remote programming and visualisation.

The remaining three papers address advanced themes in newer areas of robotics research, including an important application of microrobotics technology (Nelson & Sun), the characterization of an environment for classifying rock types (Pedersen), and new lightweight, unobtrusive, interfaces for wearable computers (Kosaka & Mohri).

We encourage you to read the articles and to peruse the references provided by the authors to gain more insight into these important areas of advanced robotics.

Finally, now is the time to begin planning for the SPIE Industrial Photonics Stuttgart meeting, October 7-10, 2002, in Stuttgart, Germany. The meeting comprises conferences on many areas of robotics and machine perception. Further details and calls for papers can be obtained from the SPIE web site. We look forward to seeing you in Stuttgart.

Gerard McKee
Technical Group Chair
The University of Reading, UK
E-mail: Gerard.McKee@Reading.ac.uk

Calendar

2002

7th International Conference on Intelligent Autonomous Systems (IAS-7)
25-27 March
Marina del Rey, California USA
http://ias7.cs.umn.edu/

AeroSense
1-5 April
Orlando, Florida USA
http://spie.org/Conferences/Programs/02/or/

IEEE International Conference on Robotics & Automation (ICRA)
11-15 May
Washington USA
http://www.icra2002.org

The Society for the Study of Artificial Intelligence and the Simulation of Behaviour (SSAISB)
AISB'02 Convention
2-5 April
Imperial College, London, United Kingdom
http://www.aisb.org.uk/

The Eighteenth National Conference on Artificial Intelligence
Fourteenth Innovative Applications of AI Conference
28 July-1 August
Edmonton, Alberta, Canada
http://www.aaai.org/Conferences/National/2002/aaai02.html

From Animals to Animats 7
The Seventh International Conference on the Simulation of Adaptive Behavior (SAB'02)
4-9-11 August
Edinburgh, UK
http://www.isab.org/sab02/

CFP: IROS'02, IEEE/RSJ
International Conference of Intelligent Robots and Systems
30 September-4 October
Lausanne, Switzerland
http://IROS02.epfl.ch

Industrial Photonics Stuttgart
7-10 October
Stuttgart, Germany
http://spie.org/info/pe/

ISR (International Symposium on Robotics)
8-11 October
Stockholm, Sweden
http://www.isr2002.com/

For More Information Contact

SPIE • PO Box 10, Bellingham, WA 98227-0010
TEL: +1 360 676 3290 • Fax: +1 360 647 1445 • E-mail: spie@spie.org
www.spie.org

Behavior engineering environment for autonomous robot teams

The speed and complexity of robot behaviors are key factors for mobile autonomous robots acting in unpredictable, dynamic environments. The Behavior Engineering team of the Fraunhofer Institute for Autonomous Intelligent Systems (AIS) focuses on the development of design environments that are suitable for supporting the highly iterative process of robot-behavior programming. The design process includes high-level specification of robot behaviors, support for cooperation in a team of heterogeneous robots, simulation of a robot team, and support for monitoring and debugging during experiments with real robots. We use soccerplaying RoboCup (middle-size league) robots as our application to demonstrate the suitability and performance of our design environment for robot behavior programming.

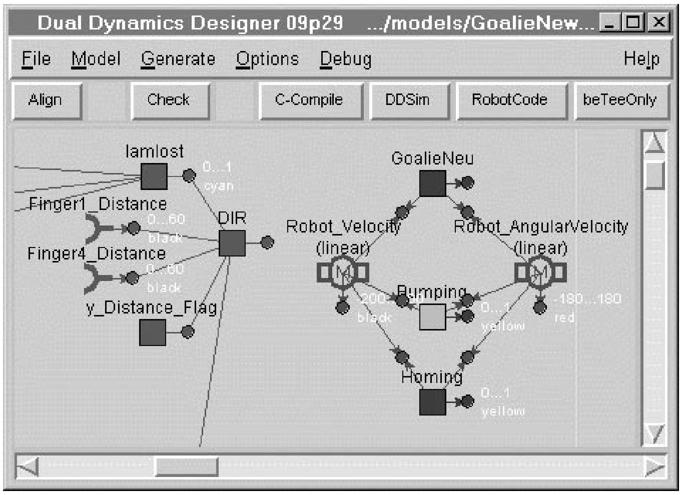

Figure 1. DD-Designer: sensors, sensor filters, behaviors, actuators, entered as a network of typed data-processing elements.

Figure 3. RoboCup, middle-size-league robot.

Robot behavior specification

Our approach to robot programming is based on a mathematical model for robot behaviors that we have developed. It integrates central aspects of a behavior-based approach, robust control, and a dynamical-systems representation of actions. Robot behaviors are specified through ordinary differential equations, forming a global dynamical system made of behavioral subsystems that interact through specific coupling and bifurcation-induction mechanisms. Behaviors are organized such that higher levels have a larger time scale than lower ones. Since, at the elementary level, the activation of behaviors (activation dynamics) are separated from their actuator control laws (target dynamics), we named our approach "Dual Dynamics".[1] An important feature of Dual Dynamics is that it allows for robust and smooth changes between different behavior modes, which results in reactive, fast and natural motions of the robots.

Behavior engineering environment

The successful design of robot software requires means to specify, implement and simulate as well as to run and debug the robot software in real-time on physical robots. The integrated Behavior Engineering Environment we have developed in recent years allows us to specify robot behaviors at a high level of abstraction, hiding low-level programming language details. The specification is used as a central reference to generate all implementation artifacts required during the design process. This includes a robot behavior documentation in HTML, a simulation model of the robotic behavior system in Java, control programs for physical robots in C/C++ and a parameter set for our generic testing and debugging tool. The Behavior Engineering Environment comprises the specification tool DD-Designer, the simulator DDSim and the real-time monitoring tool beTee.

- DD-Designer allows us to specify a robot behavior system in terms of sensors, actuators, sensor processing elements and a hierarchy of coupled behaviors.[2,3] Each of the processing elements and behaviors is formulated using a combination of control data flow and differential equations. This specification is the basis for the generation of all required implementation artifacts in the design flow.
- DDSim allows the simulation of a cooperating team of robots. The simulation includes synthetic sensor stimuli derived from the virtual scene the robots are in. Sensor simulation comprises laser range finders and an emulation of the vision system. Each simulated robot is described by an XML file allowing the reconfiguration of robot shape and sensor equipment. The behavior system of each simulated robot is generated from the abstract behavior specification edited in DD-Designer.
- beTee is a real-time monitoring tool for tracing arbitrary variables of the simulated or (via wireless LAN) physical robot. This tool allows the analysis of the running behavior system on very fine-grained level. We use this tool to monitor internal states of all running behavior systems, in real-time, at a rate of 50Hz.

Behavior design for RoboCup robots

The test bed and demonstrator application for our behavior engineering approach is soccer playing. The robots take part in the middle-size league tournaments of the international RoboCup contest, a very demanding benchmark for mobile robots. Our robots (Figure 3) were custom-built in the Institute for Autonomous Intelligent Systems. Our hardware consist of 2-degree-of-freedom, 1-PC-3-micro, controller-equipped robots with a panning camera for ball and landmark detection, infrared-based distance sensors, standard bumper sensors, odometry and a piezo-gyro. We use our Behavior Engineering Environment to specify, simulate, run, test and debug a team of these autonomous robots. Our new generative approach to robot behavior programming enables us to design and change robot behaviors on the fly within minutes. The reconfigurability of our design environment suggests its use for various mobile robot platforms.

Dr. Ansgar Bredenfeld
Fraunhofer Institute for
Autonomous Intelligent Systems
Schloss Birlinghoven
D-53754 Sankt Augustin, Germany
E-mail: bredenfeld@ais.fraunhofer.de
http://www.ais.fhg.de/be

References

1. H. Jaeger, Th. Christaller, *Dual dynamics: Designing behavior systems for autonomous robots,* **Artificial Life and Robotics 2,** pp.108-112, 1998.
2. A. Bredenfeld and G. Indiveri, *Robot Behavior Engineering using DD-Designer,* **IEEE Int'l Conf. on Robotics and Automation (ICRA 2001),** Seoul, Korea, May 23-26, 2001.
3. A. Bredenfeld and H.-U. Kobialka, *Team communication using Dual Dynamics,* **Balancing Reactivity and Social Deliberation in Multi-Agent-Systems,** Springer, LNAI 2103, 2001.

Tight mobile-robot cooperation

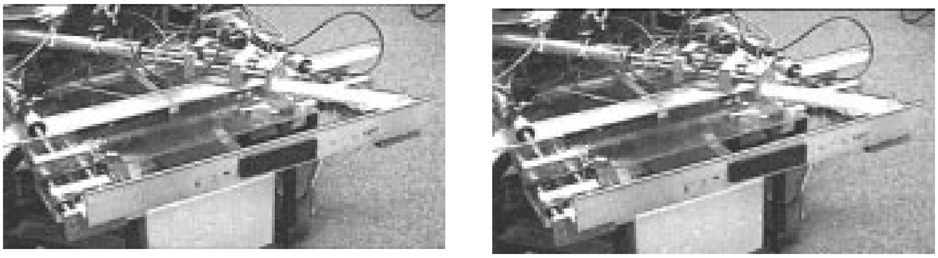

Figure 1. A compliant, 3-degree-of-freedom arm mounted on a mobile base. A position-controlled scheme is used to adjust the equilibrium position of the springs dynamically so that the planar forces and moment can be adjusted. The arm can be controlled to achieve any desired, planar, Cartesian stiffness matrix.

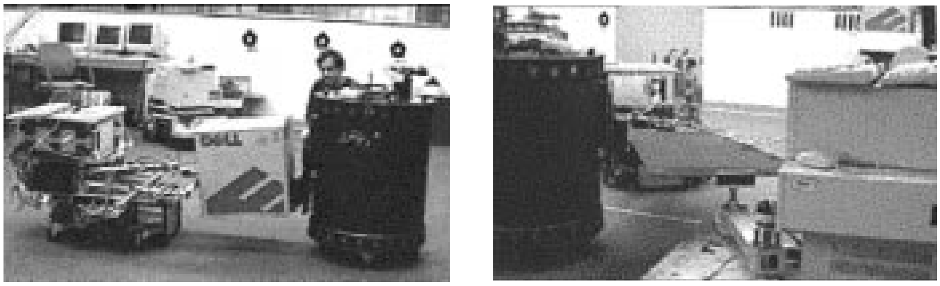

Figure 2. In the picture on the left, a compliant arm is mounted on the rear mobile robot. The arm interacts with a stiff, passive arm mounted on the lead robot. In the picture on the right, the compliant arm interacts with two other, passive, stiff arms in a system of multiple coordinated mobile robots that are capable of picking-up and transporting objects. In the second case, a large, flexible board is carried. In both cases, tightly-coupled manipulation is demonstrated.

The next generation of robotic systems will rely on complex human and machine interactions. Designing robotic systems that effortlessly interact with other robots or users is very complicated, and currently robots have only been very successful at repetitive position control tasks such as welding or painting. An example of machine interaction is a team of mobile robots that are tightly coupled, with force interactions, when lifting or carrying objects. In another example, passive, personal, robotic assistants are being researched, designed, and created to aid people. On the market, there are entertainment robots such as *Aibo®* and dolls such as *My Real Baby®*. There are vacuuming robots, lawn mowing robots, home and office robots, and security robots. It is interesting, though, that all of these personal robots interact solely and passively with a human. There is no force interaction or grasping, and each robot works individually not as part of a team.

Human machine interaction with a team of new mobile manipulators (mobile robots with additional robotic arms that allow manipulation) will allow users to assemble large parts more easily. Other scenarios include: clean-up and transportation of hazardous objects, carrying of bulky items on factory floors, and aiding the elderly in standing or balancing.

Many research problems still need to be addressed for cooperative interaction of multiple manipulators allowing force interactions. Robust sensing of forces and compliant actuation is needed for manipulation. New grasp metrics to determine soft and stable grasps will determine a safe contact stiffness between the object and the team members (robots and humans). The coordination between the manipulation and the locomotion systems (arm and the platform) must be optimized. There needs to be an efficient way of communicating and sharing information in real time. Lastly, a control architecture for the entire system and for each mobile robot must allow for complex interactions. It should be possible to organize the robots differently for different tasks, forcing the controllers to be independent and yet able to function in a tightly-coupled architecture when carrying objects. The robots must coordinate their trajectories in order to maintain a desired formation while maintaining the grasp. Unlike the task of pushing a box, the robots must maintain a formation while grasping and carrying object. Research in mobile robotics has started in all of these areas.

Mobile robotics research, and specifically mobile robot cooperation, can be split into different areas such as tight mobile-robot cooperation, robot cooperation, behavior-based control, and mobile-robot motion planning allowing manipulation.

Our previous work attempted to address some of the important issues in tight mobile-robot cooperation.[1-3] Others, notably, have developed mobile robot systems for transporting objects. One approach to the problem is to control each mobile manipulator by a computed torque scheme, and let the mobile manipulators share real–time servo-level information via a wireless network. Research has focused on the ability to generate system dynamics in a modular fashion: Khatib's robots, Romeo and Juliet, perform a wide variety of tasks using this method.

In my research, the controllers for the robots in the team are decentralized to allow for autonomous actions. The architecture is composed of state diagrams for each behavior ensuring that behaviors such as docking, picking-up, or transporting objects are completed. To simplify the optimization problem to control both the manipulator and the mobile base, the locomotion and grasping are mechanically decoupled. The manipulator data is still used to modify the robot trajectories.

We have focused on developing the compliant actuation needed for manipulation as well as feedback to the user. Manipulators consist of selectively-compliant actuators that allow for soft interactions. A mechanism has been created that can vary the planar spatial compliance.

Because the compliance can be altered at each contact, the design question arises to determine what are the appropriate planar stiffness matrices. The compliance matrix at each contact must be chosen to ensure invariant global properties. Grasp stability is an important property, but other properties are important as well. For example, it must not be assumed that contacts are static in mobile environments. Contact deviations from their initial positions cause force errors that must be reduced. By knowing the force and position relationships that occur during grasping, an inverse problem can be solved to determine the set of compliance matrices needed for the group of manipulators and humans that will interact.

We have built a team of mobile manipulators that are able to pick-up, transport, and carry objects in laboratory environments. The team consists of heterogeneous robots that have soft and stiff manipulators. To accomplish the task, trajectory information is shared. In our approach, the control of each platform is decomposed into the control of the gross trajectory and the control of the grasp. The gross trajectory is shared with all partners at low rates, in order to ensure that each robot carries the object equally. One or more actively-controlled, compliant arms control the grasp forces in the formation, allowing the robot platforms to be position controlled. The compliant arms accommodate the excessive forces due to platform positioning errors and odometry errors.

By understanding mobile grasping, new tasks can be performed such as manipulation of flexible car doors using a team of mobile manipulators. In this example, not only is the object grasped and carried, but the object has its own complex vibration modes as well. I envision a team of multiple mobile manipulators that can safely interact with their physical environment. These sys-

continues on p. 11

Multi-robot research at GRASP

The General Robotics, Automation, Sensing and Perception Lab (GRASP) is a multi-disciplinary research laboratory at the University of Pennsylvania in Philadelphia. Current research at GRASP focusses on dynamics and control of robots, vision and active sensory perception, and system design and automation.

Multi-robot research goals

Our goal is to develop a framework and the support tools for the deployment of multiple autonomous robots in an unstructured and unknown environment with applications to reconnaissance, surveillance, target acquisition, mapping, exploration and cooperative manipulation. The current state-of-the-art in control software allows for supervised autonomy, a paradigm in which a human user can command and control one robot using teleoperation and close supervisory control. The objective here is to develop the software framework and tools for a new generation of autonomous robots. The main components are the methodology and the software that will enable robots to: exhibit deliberative and reactive behaviors in autonomous operation; be reprogrammed by a human operator at run-time; and learn and adapt to unstructured, dynamic environments and new tasks, while providing performance guarantees.

The architecture and tools need to be scalable to tens and hundreds of autonomous robots and allow a single human operator to control an entire fleet. In order to realize these goals we developed and tested our algorithms on the following robot testbeds.

Hardware platforms

The Clodbuster platform (CB, see Figure 1) is based upon a 1/10 scale radio controlled model of a monster truck, made by Tamiya Inc. We have made significant modifications to the vehicle, including the addition of an omni-directional camera, a video transmitter for sending images to a remote workstation for processing, additional on-board power, and an improved suspension. This setup provides us with a robust, low cost (less than $1000 without camera), autonomous, all-terrain mobile robot platform.

Using the same base platform, significant improvements were made to the second CB variant, with all the processing moved on-board. CB-II sports a PIII 850 MHz processor, 128MB RAM, and 802.11B wireless ethernet connectivity. These modifications yield truly distributed platforms, provide additional processing power, and eliminate the radio interference associated with the wireless transmission used in CB-I. An additional sensor suite, integrating IR detectors, accelerometers, and gyroscopes, is presently being added. The third generation robots will be equipped with notebook computers, which will simplify the power requirements.

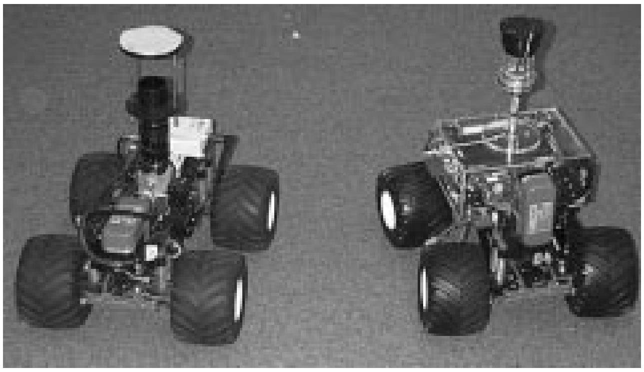

Figure 1. The low-cost CB I (left) and CB II (right) robot platforms developed at GRASP.

Figure 2. A team of four robots locating, trapping and moving a box cooperatively.

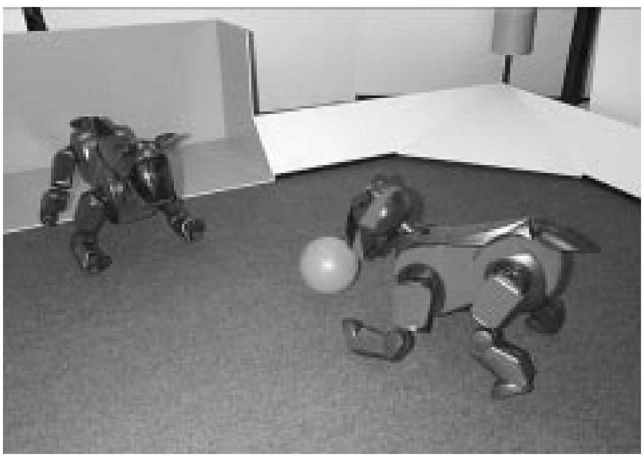

Figure 3. Attacker ready to score on Goalie in a practice match.

We also use the commercially available Cye robot from *Probotics*: we integrated omni-directional vision and on-board computing for our research platform. Our vision algorithms are augmented by the integrated position encoders and torque sensors that come with the robot.

Software framework and architecture

Our software framework divides the overall multi-robot control task into a set of modes (executed by agents), that may be executed sequentially or in parallel. Modes can consist of high level behaviors such as planning a path to a goal position, as well as low level tasks such as obstacle avoidance. We developed CHARON, a high level language, which is used to formally describe how and when transitions between these modes are to take place in order to achieve a set of global objectives.

The modular, hierarchical programs written in CHARON are inherently parallel, both at the agent level and the mode level. To implement this parallelism on our Clodbuster test-bed we have adopted the paradigm of “live objects”. A live object encapsulates algorithms and data in the usual object-oriented manner, together with control of a thread within which the algorithms will execute, and a number of events that allow communication with other live objects. The parallel execution required by CHARON programs is provided by the use of threads, the hierarchy by control of the execution of each object’s thread, and the modularity by the use of standard C++ object-oriented techniques.

Currently we use combinations of these objects in hierarchies to accomplish basic robotic exploration and localization abilities. Once basic sensing and control strategies were implemented as live objects, the combination and re-use of these objects to provide novel functionality was quick (development time) and simple (code complexity).

Cooperative sensing and localization

Simultaneous Localization And Mapping (SLAM) poses a tremendous challenge

continues on p. 11

Wearable user interface based on hand/finger gesture recognition

We have recently developed a wearable user interface that recognizes user hand/finger gestures to generate useful commands for wearable computers. This interface allows the user to operate anywhere and at any time. The current version of wearable user interface includes individual gyro-sensors on the fingertips as well as the combination of acceleration sensors and gyro-sensors on the back of the hand, in order for the interface to efficiently measure the three-dimensional hand and finger gesture.

Because the technologies allow the users to operate computers anywhere and at anytime, wearable computers have become popular rapidly. Figure 1 shows a typical future scenario, where wearable computers will be used with two key components: a wearable viewer and wearable user interface. In this figure, the wearable viewer, the “PC Eyetrek”, was developed by Olympus Optical Co., Ltd..[1,3] This device includes a 0.47inch SVGA (800×600 pixel resolution) LCD monitor. This ultra-light viewer enables the user to see the monitor with one eye, thus allowing the observation of the real environment with the other eye. The user also wears a user interface on the hand and fingers. This generates useful commands to operate the computer through hand/finger gesture recognition, and fully replaces conventional pointing devices such as mice and joysticks.[2]

Figure 1. The user wears a viewer on her face, and an interface on her hand.

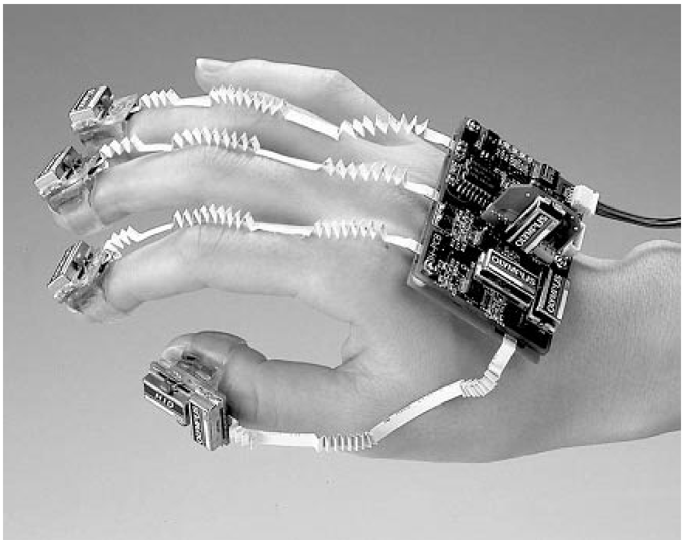

Figure 2. A larger view of the wearable user interface. The finger shape is estimated from the output of the gyro-sensors mounted on each finger.

The emergence of wearable computers has been dramatically changing the concept of a human-computer interfaces by imposing high demands on its ease of use, comfort, size, and reliability. In the past decades, many attempts have been made to develop such user interfaces. The Dataglove, used mostly for virtual reality applications, captures hand and finger motion in conjunction with an additional three-dimensional pose sensor attached to the hand. Although the Dataglove is powerful enough to estimate accurate motion of hand and fingers, its large size makes it unsuitable for wearable computers. Another possibility is a speech-recognition-based user interface, although the would be difficult to use in noisy environments.

Our wearable user interface should solve all these problems. Figure 2 shows a larger view of our prototype wearable user interface. The user wears combined gyroscopic and acceleration sensors on the back of the hand that measure its three-dimensional orientation with respect to the ground. A gyro-sensor on each fingertip measures its angular motion with respect to the back of the hand. Note that the relative bending angle of the figer is computed by comparing the outputs of the hand and fingertip gyro-sensors. Since no other devices need to be worn externally, and particularly not in the palm of the hand, the user can wear this compact interface anywhere and anytime, and even can hold other objects or take notes with pen.

It is also important to mention here that we do not measure all precise joint angles of finger motion in order to estimate the precise hand/finger shape, unlike traditional gesture-recognition devices. Instead, we developed a new algorithm that estimates principal finger motion using kinematic constraints taken from the relative orientation of the fingertip and the back of the hand.

The computer algorithm recognizes the user’s hand/finger gesture, and generates various commands that can simulate existing pointing devices such as mouse, joystick, and trackpad. For example, in the case of a trackpad, if the user locates the back of the hand horizontally and moves the index finger forward, the cursor on the screen will move up. Conversely, if the user moves the finger backward, the cursor on the screen will move down. Moving the finger counterclockwise with the wrist, then the cursor will move from left to right. Mouse clicks can be performed by tapping the thumb and the index finger together. In addition to such conventional pointing device commands, this interface allows the user to define new gesture commands that may be more appropriate for certain applications. For instance, the user could entertain traditional Japanese “rock-paper-scissors” games.

The potential applications for this device are not limited to wearable computers and game devices. One possibility is that it may help hospitalized and handicapped people to live their lives, and to aid rehabilitation.

Akio Kosaka and Ko Mohri
Advanced Technology Institute
Olympus Optical Co., Ltd.
2-3 Kuboyama-cho
Hachioji-shi, Tokyo, 192-8512, Japan
E-mail: {a_kosaka, k_mouri}@ot.olympus.co.jp

References

1. T. Shimoyama, *Wearable micromachine,* **Mstnews,** pp. 29-30, June 2000.
2. K. Mohri, *Wearable human interface based on hand/finger gesture recognition,* **Japanese Journal of Human Interface 2** (4), pp. 283-292, 2000.
3. “PC EyeTrek,” Olympus Optical Co., Ltd., Japan, http://www.olympus.co.jp.

Microrobotics and MEMS technology for biological cell research

An active research area in the Advanced MicroSystems Lab[1] at the University of Minnesota, is the development of microrobotics and MEMS for the manipulation of biological cells. Two ongoing research topics are the realization of the autonomous manipulation of single cells, and the characterization of biomembrane mechanical properties using microrobotic systems with integrated vision and force sensing modules. The objective is to obtain a fundamental understanding of single cell biological systems and provide characterized mechanical models of biomembranes for deformable cell tracking during biomanipulation and cell injury studies.

Biomanipulation—autonomous microrobotic pronuclei DNA injection

To improve the low success rate of manual operation, and to eliminate contamination, an autonomous microrobotic system (shown in Figure 1) has been developed to deposit DNA into one of the two nuclei of a mouse embryo without inducing cell lysis.[2,3] The lab's experimental results show that the success rate for the autonomous embryo pronuclei DNA injection is dramatically improved over manual conventional injection methods. The autonomous microrobotic system features a hybrid controller that combines visual servoing and precision position control, pattern recognition for detecting nuclei, and a precise auto-focusing scheme. Figure 1 illustrates the injection process.

To realize large-scale injection operations, a MEMS cell holder was fabricated using anodic wafer-bonding techniques. Arrays of holes are aligned on the cell holder, which are used to contain and fix individual cells for injection. When well calibrated, the system with the cell holder makes it possible to inject large numbers of cells using position control. The cell injection operation can be conducted in a move-inject-move manner.

A successful injection is determined greatly by injection speed and trajectory, and the forces applied to cells. To further improve the microrobotic system's performance, a multi-axial MEMS-based capacitive cellular force sensor is being designed and fabricated to provide real-time force feedback to the microrobotic system. The MEMS cellular force sensor also aids our research in biomembrane mechanical property characterization.

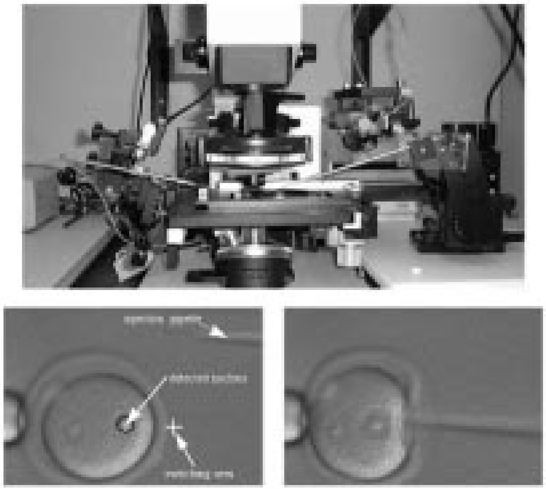

Figure 1. Shown is the microrobotic cell injection system and the injection process.

Multi-axial MEMS-based cellular force sensor

The capacitive cellular force sensor (3.2×3.0mm) being designed and fabricated measures forces up to 490µN with a resolution of 0.1µN. Figure 2 shows the solid model of the force sensor design. The constrained outer frame and the inner movable plate are connected by four curved springs. The probe deforming a cell membrane causes the movable plate to travel in the opposite direction. This changes the gap between each pair of the interdigitated capacitor comb fingers. Consequently, the total capacitance of the capacitor is changed, which represents the applied force on the cell membrane. By varying the dimensions of the springs, which changes the springs' stiffness, a large range of displacements can be achieved for a given applied force. To make the force sensor capable of resolving forces in two directions, the interdigitated capacitors are configured to be orthogonal to each other, as shown in Figure 2. Structural and electrostatic analyses have been conducted to determine the geometry and configuration of the cellular force sensor. Processing the 3-D structure requires only four masks, including the forming of the tip manipulator. The chief fabrication process utilizes SOI technology and Deep Reactive Ion Etching.

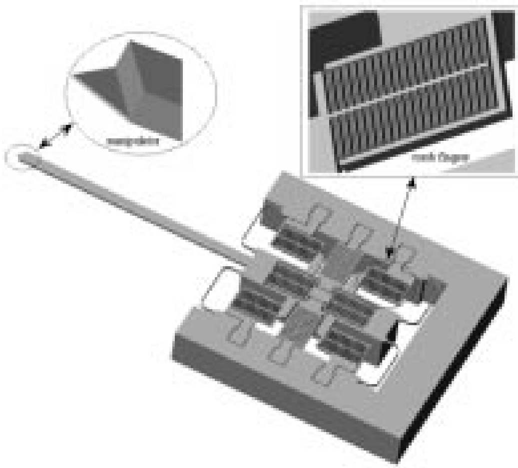

Figure 2. Shown is the multi-axial MEMS-based cellular force sensor (3.2×3.0mm).

The microrobotic system and high-sensitivity cellular force sensor are also being applied to the biomembrane mechanical property studies. The goal is to obtain a general parameterized model describing cell membrane deformation behavior when an external load is applied. This parameterized model serves two chief purposes. First, in microrobotic biomanipulation, it allows online parameter recognition so that cell membrane deformation behavior can be predicted. Second, for a thermodynamic model of membrane damage in cell injury and recovery studies, it is important to appreciate the mechanical behavior of the membranes. This allows the interpretation of such reported phenomena as mechanical resistance to cellular volume reduction during dehydration, and its relationship to injury. The establishment of such a biomembrane model will greatly facilitate cell injury studies.

Conclusion

Experiments demonstrate that microrobotics and MEMS technology can play important roles in biological studies such as automating biomanipulation tasks. Aided by microrobotics, the integration of vision and force-sensing modules, and MEMS design and fabrication techniques, we are conducting investigations in biomembrane mechanical property modeling, deformable cell tracking, and single cell manipulation.

Brad Nelson and Yu Sun
Department of Mechanical Engineering
University of Minnesota, Minneapolis
111 Church Street, S.E.
Minneapolis, MN 55455
E-mail: nelson@me.umn.edu

References

1. http://www.me.umn.edu/divisions/design/adv_microsystems/
2. Y. Sun and B.J. Nelson, *Microrobotic cell injection*, **Proc. Conf. on Robotics and Automation,** pp. 620-625, 2001.
3. Y. Sun and B.J. Nelson, *Autonomous Injection of Biological Cells Using Visual Servoing*, **Int'l Symp. on Experimental Robotics (ISER 2000),** pp. 175-184, 2000.

Remote programming and Java3D visualization for internet robotics

In recent years, advances in computer and internet technology have supported rapid growth in internet robotics. As a result, many systems providing permanent, online, robot control have been developed. However, these systems—particularly those based on visual feedback through camera images—have had the disadvantage of reacting slowly to the operator's input: this situation is due to current limitations on the communication bandwidth of the internet.

The goal of our activity in this area has been to create an efficient teleoperation system that is fast, easy-to-control, and open: that is, capable of running on a variety of computer platforms. To do this, we have developed a dynamic virtual environment in which graphical models of the robot and surrounding objects are used to provide real-time visualization of the current state of the robot site. In addition, an environment for remote robot programming has been developed in order to simplify the implementation of complicated, repeated operations.

During development, we applied our tools and methods to the remote control of a PUMA 560 robot manipulator over the web.[1,2] All our components are realized using open technologies, Java and Java3D, to allow the system to be accessible through any standard browser.

Visualization of the robot and environment

Experiments in controlling the PUMA robot via the web revealed that successful robot contro,l based purely on image information (for instance, from a video camera), is impossible given existing internet communication rates.

An alternative way to provide suitable control information for the operator is by using an online 3D model of the robot and its environment. Data transmission is restricted to small parcels defining the current coordinates of the robot and the objects. Under this regime, the robot can be controlled successfully even when communication rates are extremely low. This visual interface also has the advantage of allowing additional, complex functionality useful for control: these include changing the viewpoint, zooming in/out of the scene, using semitransparent images, etc. (see Figure 1).

The open technology used—Java3D—allows the virtual control environment to run on any type of computer platform through the internet. Java3D features, such as automatic collision detection and the generation of stereo images, also provide attractive possibilities for future progress.

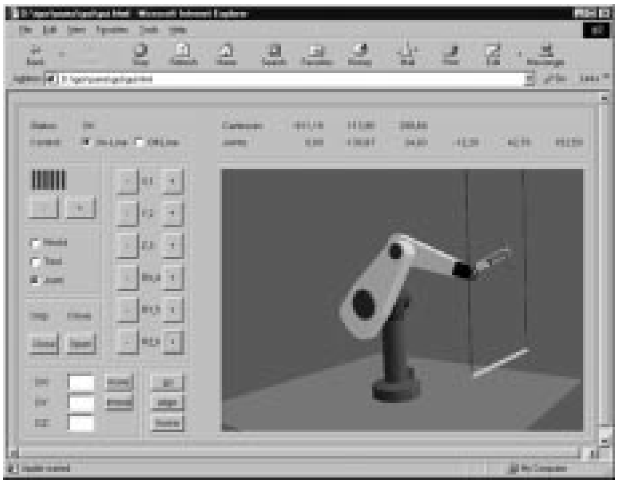

Figure 1. Web interface for remote control of a PUMA robot.

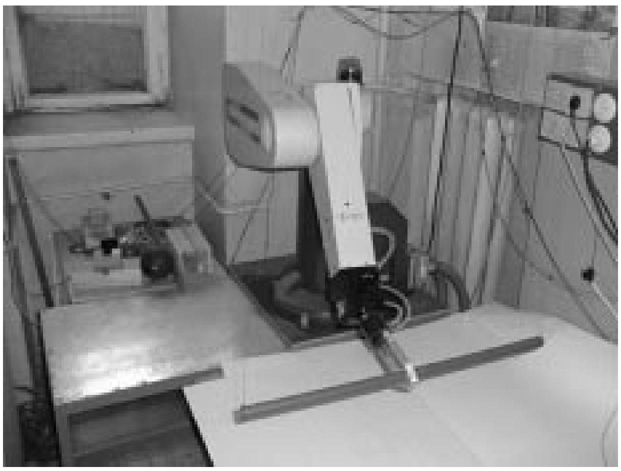

Figure 2. The PUMA grasps the rod through human teleoperation, despite the low bandwidth (internet) connection between operator and robot.

Remote robot programming

The operator's control environment contains a tool for programming, thus providing the useful possibility of setting up complicated robot actions such as pick-and-place, assembly, etc. This significantly simplifies the problem of remote robot control. The remote programming module is organised as an interpreter of the commands of the robot control language (Rcl). The operator can perform both individual commands and arbitrary sets of them (i.e. programs). Rcl was realised using an interpreted scripting language Tcl/Tk. All Tcl standard commands are interpreted inside the Tcl-shell. New commands of Rcl were developed that could be interpreted within the Tcl-shell together with Tcl standard commands. The Java version of Rcl is realized with the use of the Jacl package.

Experiments on internet robot control

The tools and methods described here were applied while developing a system for the remote control of a PUMA 560 robot over the web. The goal of the experiments was to grasp a rod suspended on two threads attached to its ends (Figure 2). The communication rate for all of the sessions was extremely low, about 100 bytes/sec on average, so nearly 30 seconds was needed to receive every portion of the TV data. However, by using the virtual environment, the delay was less than 1s, and the rod was grasped successfully.

Remote control of the PUMA robot via a standard internet line was successfully demonstrated during the IEEE International Conference on Robotics and Automation (ICRA 2001, Seoul, Korea). A Moscow-based robot was controlled from the conference venue: a distance exceeding 10,000km.

Future work will focus on the rapid generation of the 3D model of the robot working environment, and developing methods that will allow interaction with moving objects[3] via the internet, not just static ones.

The main results and Java3D demo programs are presented on: http://www.keldysh.ru/i-robotics/home.html.

Igor Belousov* and Gordon Clapworthy†
*Keldysh Institute of Applied Mathematics
Russian Academy of Sciences
4, Miusskaya Square
Moscow 125047, Russia
E-mail: belousov@spp.keldysh.ru

†De Montfort University
Dept. of Computer & Information Sciences
Hammerwood Gate, Kents Hill
Milton Keynes MK7 6HP, UK
E-mail: gc@dmu.ac.uk

References

1. G. Clapworthy and I. Belousov et al., *Medical Visualisation, Biomechanics, Figure Animation and Robot Teleoperation: Themes and Links,* **Confluence of Computer Vision and Computer Graphics,** Kluwer Academic Publishers, pp. 215-228, 2000.
2. I. Belousov, R. Chellali, and G. Clapworthy, *Virtual Reality Tools for Internet Robotics,* **Proc. IEEE Int'l Conf. on Robotics and Automation ICRA 2001, Seoul, Korea**, pp.1878-1883, May 21-26, 2001.
3. I.Belousov and D.Okhotsimsky et al., *Robot interaction with spherical pendulums,* **Izvestiya of the Russian Academy of Sciences, Mechanics of Rigid Bodis 1,** pp. 194-204, 2001.

SAIL: A “mentally” developing robot

continued from cover

world.

The cognitive mapping accepts input from attention-selected signals from three sensory mappings. It has also an internal state that records the context within a short time past. The cognitive mapping is implemented by the Incremental Hierarchical Discriminant Regression (IHDR). The IHDR is a general mapping approximator. It uses a tree structure to approximate the input and output mapping in a course-to-fine fashion. In each node of the tree, incremental discriminating analysis is carried out to derive the most discriminating features to form a subspace. The input components not related to the output are disregarded by each node, thus enabling each one to generalize according to the probability distribution. More detail of IHDR can be found in Reference 7. IHDR produces action signals not only for external effectors, such as steering or drive, but also internal effectors such as attention selection and control of the internal gating system. Since internal effectors cannot use supervised learning, reinforcement learning is used to learn internal actions.

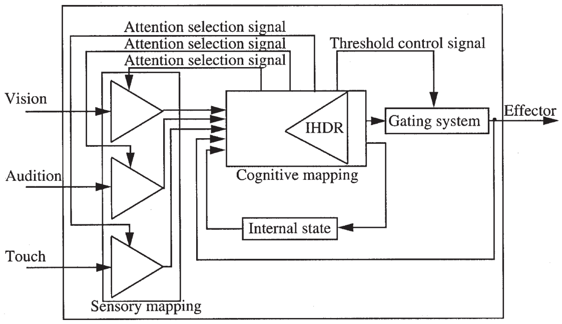

Figure 2. The SAIL-2 developmental architecture.

Table 1. Approaches to artificial intelligence.

Approach	Species architecture	World knowledge	System behavior	Task-specific
Knowledge-based	programme	mod.[a]	mod.	Yes
Behavior-based	programme	avoid mod.	mod.	Yes
Learning-based	programme	p-mod.[b]	p-mod.	Yes
Evolutionary	search	p-mod.	p-mod.	Yes
Developmental	programme	avoid mod.	avoid mod.	No

[a]Modeling.
[b]Modeling with parameters.

The gating system is used to prioritize multiple applicable actions and to release an action only when its potential is sufficiently high, thus avoiding unstable actions. The gating system also enables the developmental robot to “mentally rehearse” actions before they are released for execution by the corresponding effector.

At birth time, no mapping, other than the framework, exists. Experience with interactions in the physical world gradually grows mappings in real time. The more experience the robot has, the more sophisticated its cognitive and behavioral capabilities are.

Developmental experiments

The SAIL robot has been trained to perform a series of challenging tasks, including vision-guided navigation, turning attention towards a moving object and reaching for it, speech recognition, speech-guided object manipulation, and vision and speech directed navigation.To start the developmental process, we use more supervised learning so that the robot can perform some perception-guided action quickly. For example, to teach the SAIL robot how to get around using its vision, the human teacher teaches the robot by taking it for a walk along the hallways of MSU engineering building. The force sensors on the robot body sense the pressure of the teacher’s hands, and its two drive wheels comply by moving at a speed proportional to the force sensed each side. In other words, the robot performs supervised learning, in real-time, through imitation.

The IHDR mapping algorithm processes the input images in real time. It derives features that are related to the action and disregards features that are not. The human teacher does not need to define features. To address the requirement of real-time speed, the IHDR method incrementally constructs a tree architecture that automatically generates and updates the representations in a coarse-to-fine fashion. The real-time speed is achieved by the logarithmic time complexity of the tree, in that the time required to update the tree for each sensory frame is a logarithmic function in the number of fine clusters (prototypes) in the tree. After a few trips along slightly different trajectories along the hallways, the human teacher started to let the robot run free. He still needs to “hand push” the robot at certain places until the robot can reliably navigate along the hallway, without a need for this kind of hand-holding. We found that about 10 trips are sufficient for the SAIL-2 robot to navigate along the hallways using only vision: that is, without using any range sensors. Figure 3 shows SAIL navigating autonomously.

Further, we also trained the SAIL robot using reinforcement learning. For example, a human teacher speaks a command. If the robot does it correctly, the teacher presses its “good” button (a positive reward) . Otherwise, he presses a “bad” button (a negative reward). The SAIL robot uses Q-learning mechanism to back- propagate the reward in time so that later, when the same sensation is received, the backpropagated reward allows the robot to predict the expected future reward. At any state, its developmental program chooses the action that has the best expected reward. However, we do not use reinforcement for long-delayed rewards, because we believe these should be handled by further developed cognitive behaviors. We have used both supervised and reinforcement learning to teach the SAIL robot to recognize spoken commands while producing desired actions. Under this grounded learning, the robot has successfully learned 15 spoken phrases with 12 different speakers, such as “go left,” “go right,” and “freeze”.[6] With this capability, later training for vision-guided navigation was then conducted by verbal commands.

Figure 3. The SAIL robot autonomously navigating, in real time, guided by its vision (through video cameras) and without the use of range sensors.

Juyang Weng, Wey S. Hwang, and Yilu Zhang
Department of Computer Science and Engineering
Michigan State Univ.
East Lansing, MI 48824, USA
Tel: +517-353-4388
E-mail: weng@cse.msu.edu

Reference

1. J.H. Flavell, P.H. Miller, and S.A. Miller, **Cognitive Development,** Prentice Hall, New Jersey, 3rd Edition, 1993.
2. S.L. Pallas, L. von Melchner, and M. Sur, *Visual behavior mediated by retinal projections directed to the auditory pathway,* **Nature 404,** pp. 871-876, 2000.
3. J. Weng, J. McClelland, A. Pentland, O. Sporns, I. Stockman, M. Sur, and E. Thelen, *Autonomous mental development by robots and animals,* **Science 291,** pp. 599-600, 26 January 2001.
4. J. Weng, *Learning in image analysis and beyond: Development,* **Visual Communication and Image Processing,** C.W. Chen and Y.Q. Zhang, editors, Marcel Dekker, pp. 431-487, New York, NY, 1998. Also MSU CPS Tech. Report CPS-96-60, 1996.
5. J. Weng, *Cresceptron and SHOSLIF: Toward comprehensive visual learning,* **Early Visual Learning,** S.K. Nayar and T. Poggio, editors, pp. 183-214. Oxford University Press, New York, 1996.
6. Y. Zhang and J. Weng, *Grounded auditory development by a developmental robot,* **Proc. INNS-IEEE Int’l Joint Conference on Neural Networks,** pp. 1059-1064, Washington, DC, July 14-19 2001.
7. W. S. Hwang and J. Weng, *Hierarchical discriminant regression,* **IEEE Trans. Pattern Analysis and Machine Intelligence 22** (11), pp.1277-1293, 2000.

Join the Technical Group

...and receive this newsletter

Membership Application

Please Print ☐ Prof. ☐ Dr. ☐ Mr. ☐ Miss ☐ Mrs. ☐ Ms.

First Name, Middle Initial, Last Name ______________________________

Position ______________________ SPIE Member Number ______________________

Business Affiliation ______________________________

Dept./Bldg./Mail Stop/etc. ______________________________

Street Address or P.O. Box ______________________________

City/State ______________________ Zip/Postal Code ____________ Country ____________

Telephone ______________________ Telefax ______________________

E-mail Address ______________________________

Technical Group Membership fee is $30/year, or $15/year for full SPIE members.

❏ Robotics & Machine Perception
Total amount enclosed for Technical Group membership $ ____________

❏ **Check enclosed.** Payment in U.S. dollars (by draft on a U.S. bank, or international money order) is required. Do not send currency. Transfers from banks must include a copy of the transfer order.

❏ **Charge to my:** ❏ VISA ❏ MasterCard ❏ American Express ❏ Diners Club ❏ Discover

Account # ______________________ Expiration date ______________________

Signature ______________________________
(required for credit card orders)

This newsletter is printed as a benefit of membership in the **Robotics & Machine Perception Technical Group.** Technical group membership allows you to communicate and network with colleagues worldwide.

Technical group member benefits include a semi-annual copy of the *Robotics & Machine Perception* newsletter, SPIE's monthly publication, *OE Magazine*, membership directory, and discounts on SPIE conferences and short courses, books, and other selected publications related to robotics.

SPIE members are invited to join for the reduced fee of $15. If you are not a member of SPIE, the annual membership fee of $30 will cover all technical group membership services. For complete information and an application form, contact SPIE.

Send this form (or photocopy) to:
SPIE • P.O. Box 10
Bellingham, WA 98227-0010 USA
Phone: +1 360 676 3290
Fax: +1 360 647 1445
E-mail: spie@spie.org

Web: www.spie.org/info/robotics

Please send me
☐ Information about full SPIE membership
☐ Information about other SPIE technical groups
☐ FREE technical publications catalog

Reference Code: 3152

ROBOTICSONLINE

New Robotics Web Discussion Forum launched

You are invited to participate in SPIE's new online discussion forum on Robotics. The **INFO-ROBO** mailing list is being "retired" as we move the discussions to the more full-featured web forums. We hope you will participate. To post a message, log in to create a user account. For options see "subscribe to this forum."

You'll find our forums well-designed and easy to use, with many helpful features such as automated email notifications, easy-to-follow "threads," and searchability. There is a full FAQ for more details on how to use the forums.

Main link to the new Robotics forum: http://spie.org/app/forums/tech/

Related questions or suggestions can be sent to forums@spie.org.

Robotics & Machine Perception

The Robotics & Machine Perception newsletter is published semi-annually by SPIE—The International Society for Optical Engineering for its International Technical Group on Robotics & Machine Perception.

Editor and Technical Group Chair Gerard McKee
Technical Editor Sunny Bains
Managing Editor Linda DeLano
Advertising Sales Roy Overstreet

Articles in this newsletter do not necessarily constitute endorsement or the opinions of the editors or SPIE. Advertising and copy are subject to acceptance by the editors.

SPIE is an international technical society dedicated to advancing engineering, scientific, and commercial applications of optical, photonic, imaging, electronic, and optoelectronic technologies. Its members are engineers, scientists, and users interested in the development and reduction to practice of these technologies. SPIE provides the means for communicating new developments and applications information to the engineering, scientific, and user communities through its publications, symposia, education programs, and online electronic information services.

SPIE—The International Society for Optical Engineering, P.O. Box 10, Bellingham, WA 98227-0010 USA. Tel: +1 360 676 3290. Fax: +1 360 647 1445.

European Office: Karin Burger, Manager, karin@spieeurope.org, Tel: +44 7974 214542. Fax: +44 29 2040 4873.

In Japan: c/o O.T.O. Research Corp., Takeuchi Bldg. 1-34-12 Takatanobaba, Shinjuku-ku, Tokyo 160, Japan. Tel: +81 3 3208 7821. Fax: +81 3 3200 2889. E-mail: otoresco@gol.com

In Russia/FSU: 12, Mokhovaja str., 121019, Moscow, Russia. Tel/Fax: +7 95 202 1079. E-mail: edmund@spierus.relcom.ru

Want it published? Submit work, information, or announcements for publication in future issues of this newsletter to Sunny Bains at sunny@spie.org or at the address above. Please consult http://www.sunnybains.com/newslet.html for submissions guidelines. All materials are subject to approval and may be edited. Please include the name and number of someone who can answer technical questions. Calendar listings should be sent at least eight months prior to the event.

Autonomous robotic meteorite search

continued from p. 12

way for the robot to learn this distribution while acquiring sensor data.[3] Figure 3 shows the automatic characterization of the geological environment at Patriot Hills, Antarctica.

Besides building up a map of target-type probability distributions over a geographic area, the algorithm exploits the dependencies between targets and recalculates their posterior type probabilities as more sensor data is acquired. Figure 5 shows the cumulative number of misclassifications as rocks in the dataset from Figure 4 are examined. Notice how modelling the environment causes the total number of misclassifications to decrease occasionally as more targets are examined. This is not possible if each target is considered independently.

Using the Nomad robot, we have demonstrated fully autonomous meteorite search by a rover in Antarctica. Furthermore, we have shown how a robot can characterize an unknown environment, dramatically decreasing target-classification error rates in the process. This work has applications for future planetary exploration rovers, as well as terrestrial applications such as robotic mine clearance.

This research was carried out by the Field Robotics Center of Carnegie Mellon University, under a grant from the National Aeronautics and Space Administration.

Liam Pedersen
QSS Group, Inc.
NASA Ames Research Center
MS 269-3
Moffett Field, CA 94035-1000
E-mail: pedersen@email.arc.nasa.gov

References

1. D. Apostolopoulos, M. Wagner, B. Shamah, L. Pedersen, K. Shillcutt, and W. Whittaker, *Technology and Field Demonstration of Robotic Search for Antarctic Meteorites*, **International Journal of Robotic Research,** *Special Field and Service Robotics issue,* November 2000.
2. L. Pedersen, D. Apostolopoulos, and W. Whittaker, *Bayes Networks on Ice: Robotic Search for Antarctic Meteorites*, **Neural Information Processing Symp.,** Denver, Colorado, 2000
3. L. Pedersen, *Autonomous Characterization of Unknown Environments*, **IEEE Int'l Conf. on Robotics and Automation,** 2001.

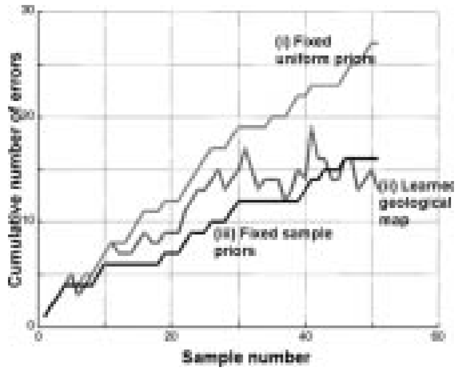

Figure 5. Cumulative number of misclassifications as samples from the Antarctic field data are examined in the order they were encountered by the robot. In (i), samples are each independently classified assuming uniform rock type probabilities. In (ii) the rock type probabilities are learned as samples are acquired, while (iii) indicates performance when rocks are again classified independently, using the known fraction of each rock type in the data set as the priors. This is the best performance possible for the independent classification of the rocks.

Multi-robot research at GRASP

continued from p. 5

to a single robot. However, sharing information amongst a team of robots can greatly simplify the task. Using a team of Clodbusters (CBs), a global map of the environment can be generated up to a scale factor. With a minimum team size of three robots, localization of each in three-dimensional space is possible. This is accomplished by using azimuth and elevation information obtained from the CB omnicams. With this relative information from each platform, localization for each can be accomplished using triangulation. Each team member can then augment a portion of the global map with its local map information.

The same cooperative scheme can be used to localize a team of robots and targets relative to a reference robot, and thus enabling control of formations and tracking targets for manipulation.

Formation control and cooperative manipulation

The problem of controlling multiple autonomous robots arises in many scenarios of current interest: military applications, where vehicles are required to maintain a close formation while avoiding obstacles; IVHS (Intelligent Vehicle Highway Systems), where a platoon of cars needs to maneuver while keeping prescribed inter-car separations; cooperative manipulation tasks, such as trapping and moving objects along prescribed paths.

We consider the problem of controlling a team of nonholonomic mobile robots in formation. The robots are required to follow a prescribed trajectory while maintaining a desired formation. The shape of the formation may change based on environmental conditions or high-level commands. By using the leader-following framework, we formulate the hybrid control problem as a sequential composition of continuous-state control algorithms to achieve a desired formation. The stability of the closed-loop hybrid system is guaranteed by using Lyapunov methods from control theory. Our analytic results have been verified using numerical simulations as well as experiments (Figure 2) using our mobile test-beds.

Legged RoboCup soccer

The Sony AIBOs (http://www.aibo.com) are self-contained autonomus quadruped robotic platforms, donated to the *UPennalizers* team at GRASP competing anually in the International RoboCup Soccer competition. In RoboCup 2001 our team placed third out of sixteen competing teams, and placed second in a series of technical challenges.

We developed a modular hierarchical software framework that allowed us to simultaneously develop low level tasks such as walking, image-processing and sound-based communication, along with high level control tasks and modes such as localization, goal scoring and defending. We are currently aiming to formalize our specification of tasks and modes in CHARON. We are also investigating learning schemes to improve our match-playing strategies.

Aveek K. Das
GRASP Laboratory
University of Pennsylvania
3401 Walnut Street, #301C
Philadelphia, PA 19104
Tel: +1 215 898 0346
Fax: +1 215 573 2048
E-mail: aveek@grasp.cis.upenn.edu
http://www.seas.upenn.edu/~aveek

References

1. http://www.cis.upenn.edu/mars
2. http://www.grasp.upenn.edu

Tight mobile-robot

continued from p. 4

tems will be the next generation of robotic devices capable of being used in unstructured manufacturing and domestic environments.

Thomas G. Sugar
Dept of Mechanical and Aerospace Engineering
P.O. Box 6106
Tempe, AZ 85287-6106
E-mail: thomas.sugar@asu.edu
http://www.eas.asu.edu/~tsugar

References

1. T. G. Sugar and V. Kumar, *Metrics for Analysis and Optimization of Grasps and Fixtures,* **Int'l J. Robotics and Automation,** accepted for publication September 2001.
2. T. G. Sugar and V. Kumar, *Design and Control of a Compliant Manipulator,* **ASME J. Mechanical Design,** accepted for publication September 2001.
3. T. G. Sugar and V. Kumar, *Control of Cooperating Mobile Manipulators,* **IEEE Trans. on Robotics and Automation,** accepted for publication October 2001.

Autonomous robotic meteorite search and characterization of the environment

Figure 1. The Nomad robot examining a potential meteorite with a color camera and visible light spectrometer.

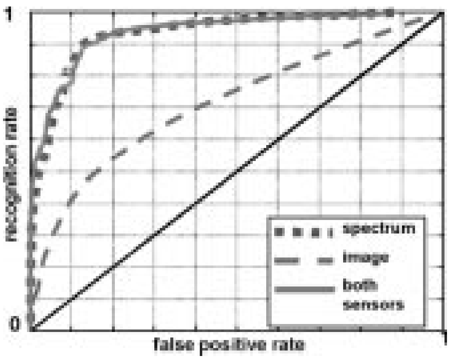

Figure 2. Recognition versus false-positive rate curves for recognizing meteorites, using 20% cross validation, in laboratory training data (color images and reflection spectra).

In January 2000,[1] Nomad, a rover built by Carnegie Mellon University, was deployed in Antarctica where it autonomously searched for meteorites and made the first autonomous identification of a meteorite by a robot.

Nomad searches the ice sheets for rocks using a forward-looking color camera on a pan-tilt mount. Candidates are identified by looking for dark blobs against the white background. Upon encountering an object, a zoomed-in image is obtained both for initial classification and to determine the objects' location relative to the robot. An arm-mounted visible light reflectance spectrometer is deployed onto the target for final target identification (Figure 1). A sunshade and external lamps switch on and off as necessary, enabling the acquisition of good quality spectra in the high contrast conditions of Antarctica. At the same time, they relax the precision with which the instrument has to be placed against rock samples.

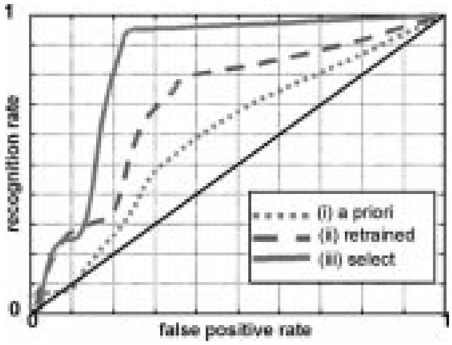

Figure 3. Recognition versus false-positive rate curve demonstrating the performance of the Nomad meteorite classifier on rocks and meteorites at Elephant Moraine, Antarctica, in January 2000, using reflection spectroscopy only. Curve (i) shows performance in the field with the classifier trained only on laboratory data obtained beforehand. Performance is improved with onsite field training (ii), which takes the difficulties of field measurements into account. Classification accuracies approaching those under laboratory conditions are attained when removing hydrothermally-altered basalts from the test set in (iii).

A Bayes-network-based classifier[2] interprets the sensor data (color images and reflection spectra) to determine the rock type (and therefore whether a rock is of extra-terrestrial origin or not). There are distinct advantages of the Bayesian approach for this problem. Evidence from multiple sources (including prior information about the area) is compounded in a natural way as it becomes available. This is particularly important for a mobile robot with multiple sensors, any of which could fail. Furthermore, the prior probabilities of finding different types of rocks change with location, hence it is important to be able to set these directly (done trivially under the Bayesian approach). Figures 2 and 3 show the classifier performance on laboratory data and on data obtained by Nomad in Antarctica in January 2000.

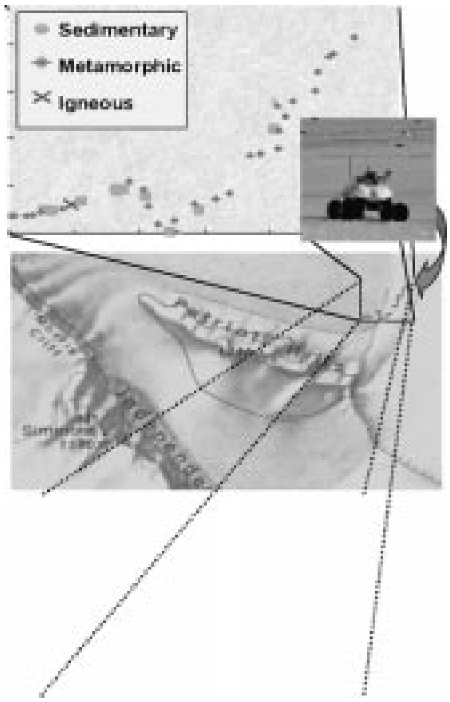

Figure 4. Patriot Hills, Antarctica, data collection site where Nomad collected spectral data from rocks on the ice sheet. The bottom map is the determined likelihood of finding a metamorphic rock, using the algorithm in [3] and the collected spectral data. (The actual rock types shown at the top were determined by a human geologist looking at the rocks. This information is not used to generate the bottom map.)

The classifier on-board Nomad ignores the relationship between rock samples, considering each independently of the others. However, rocks of similar types tend to cluster together geographically, due to the underlying geological processes that deposit them. As the robot moves, it can leave a region dominated by one rock type for a region dominated by another. We developed a method of expressing the rock-type conditional probability distribution given geographic location, and a

continued on p. 11

SPIE Society of Photo-Optical Instrumentation Engineers

P.O. Box 10 • Bellingham, WA 98227-0010 USA

Change Service Requested

DATED MATERIAL

Non-Profit Org.
U.S. Postage Paid
Society of
Photo-Optical
Instrumentation
Engineers