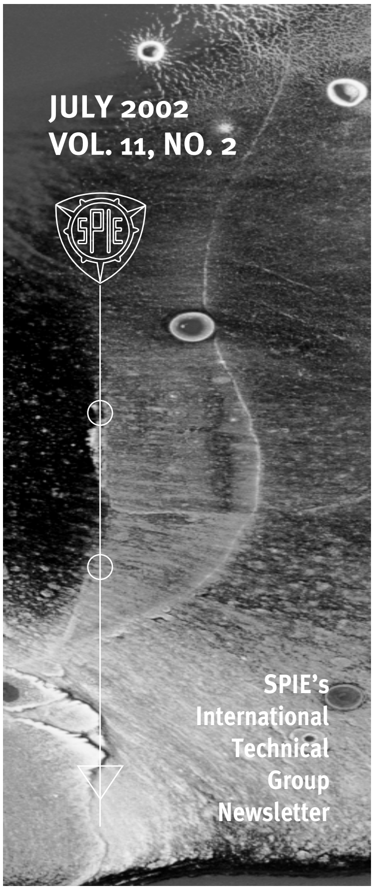

Calendar
—See page 2

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 electronically. An e-mail is being sent to all group members with advice of the web location for this issue, and asking members to choose between the electronic and printed version for future issues. If you are a member and have not yet received this message, then SPIE does not have your correct e-mail address.

To receive future issues electronically, 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 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.

Robotics and Machine Perception

The need for, and application of, 'soft' actuators

Traditionally, the design of robotic systems has been concerned with developing fast, accurate mechanisms. This has resulted in robots having high mass and power requirements and limited capacity for human interaction. Recent advances in computational power and materials, however, have allowed the production and control of lightweight and highly-flexible structures similar to those found in nature. These can be used in robotics design. As a result, bio-mimetics has been developed. Here the trend is to try to emulate the 'soft' compliant structure of muscle, bone, tendons, and skin and combine this with the power, robustness, accuracy, and endurance of traditional mechanical drives.

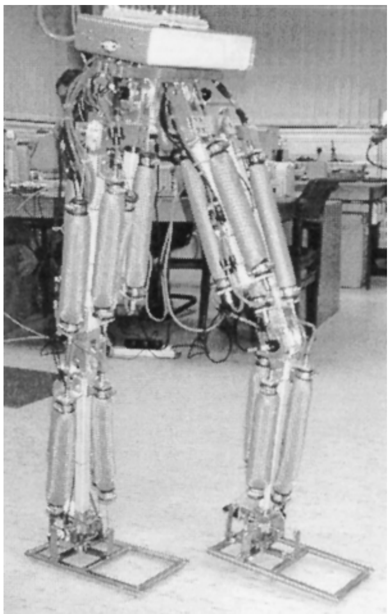
Figure 1. The lower section of a bipedal humanoid robot.

Bio-mimetic actuation

Organic muscle provides power for motion on land, in water, and in the air, in highly variable climatic conditions, and in creatures ranging in size from whales to microbes. Further, its operation in antagonistic pairs permits modulation of stiffness and position. This is vital for safer human interaction, gives more natural motion and control, and enables energy conservation through muscle elasticity. Unfortunately, real muscle is not an engineering technology and is prone to fatigue and damage, characteristics that mean it is not suitable for machine operation. The goal of our research at the University of Salford has thus become the development of an alternative bio-mimetically-inspired robot actuation system able to emulate the 'soft' compliant structure of organic systems.

A particularly interesting actuation system, from the perspective of bio-mimetic systems is the pneumatic muscle actuator (pMA)[1,2] or McKibben muscle.[3] The pMA is formed from a two-layered cylinder consisting of an inner containment liner, usually formed from an elastomeric material, and an outer flexible double helix layer of braided material such as nylon, Kevlar, polyester etc. These are clamped to endcaps that seal the open ends of the muscle and allow the input of compressed air. The detailed construction, operation, and mathematical analysis of these actuators can be found in Caldwell et al..[1,2]

In attempting to use these actuators to duplicate natural muscle, it is useful to compare the attributes of both biological and pneumatic muscles (see Table 1).

This basic structure of the muscles gives the actuator a number of desirable characteristics including:

- Having exceptionally high power- and force-to-weight/volume ratios.
- Achieving displacement of typically 35% of

Continues on page 9.

Editorial

Welcome to this second issue of the R&MP newletter for 2002. In this issue we bring together reports from researchers working in both new and traditional areas of robotics and machine perception. The area of biologically inspired robotics is growing ever more popular and finding support for funded research projects. In this issue we have two papers devoted to very different aspects of this emerging field. The first, Davis et al, reports work on pneumatic muscle actuators (pMAs), emphasising research effort to mimic natural muscles. The second, Iida, reflects research aimed at exploiting insect-like vision for the navigation and guidance of airbourne vehicles.

Three of the papers focus on new models for robot systems, drawing much of their inspiration from developments in software component technology, including mobile code. They could all be collectively listed under the banner of 'networked robotics'. The aim is flexibility through the exploitation of the Internet, but each approaches it in a different way. The first, Wang et al, presents an environment implemented in Java, LOGUE, aimed at sharing task and behavior knowledge between robots. The carrier is mobile code. The second paper, Amigoni, models a robot as being comprised of fixed and mobile elements. The latter embody the components required for cooperation between robots, and are again implemented as mobile code.The third paper, Baker et al, again exploits the concept of mobile code, but focuses on reconfiguration of pools of resources distributed about a network environment.

The remaining three papers develop themes in more established areas of robotics. Lefebvre et al. look at uncertainty modelling for compliant motion, for tasks in which the robot is to maintain contact with objects in its workspace. Cervera et al look at visual servoing using a stereo pair of cameras. Theoretical and experimental work is outlined. Finally, Fiorini outlines workspace analysis research at the ALTAIR laboratory in Italy. An algorithm for computing manipulator workspace is described, and motivated by applications involving surgical robotics.

We hope you enjoy reading all of the articles and we encourage you to take this opportunity to find out more by pursuing the references provided.

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

Tell us about your news, ideas, and events!

If you're interested in sending in an article for the newsletter, have ideas for future issues, or would like to publicize an event that is coming up, we'd like to hear from you. Contact our technical editor, Sunny Bains (sunny@spie.org) to let her know what you have in mind and she'll work with you to get something ready for publication.

Deadline for the next edition, 11.2, is:

27 September 2002: Suggestions for special issues and guest editors.

14 October 2002: Ideas for articles you'd like to write (or read).

13 December 2002: Calendar items for the twelve months starting January 2003.

Calendar

2002

The Seventh International Conference on the SIMULATION OF ADAPTIVE BEHAVIOR (SAB'02)

4–9 August
Edinburgh, UK
Program
http://www.isab.org.uk/sab02/

EPSRC/BBSRC International Workshop
Biologically-Inspired Robotics: The Legacy of W. Grey Walter

14–16 August
Bristol, UK
http://www.ecs.saton.ac.uk/~rid/wgw02/home.html

IEEE Roman 2002
11th IEEE International Workshop on Robot and Human Interactive Communication

25–27 September
Berlin, Germany
http://www.morpha.de/roman2002/

Third International Workshop on Robot Motion and Control

9–11 November
Near Lagow Lubuski, Poland
Call for Papers
http://romoco.put.poznan.pl/

15th International Trade Fair for Machine Vision and Identification Technologies

12–14 November
Messe Stuttgart
Program
http://www.messe-stuttgart.de/vision/

IEEE Workshop on Applications of Computer Vision

3–4 December
Orlando, Florida
http://www.cs.ucf.edu/~vision/workshop/2002/applicationCompVision.html

NIPS 2002
Neural Information Processing Systems

10–12 December
Vancouver, Canada
http://www-2.cs.cmu.edu/Groups/NIPS/

For More Information Contact
SPIE • PO Box 10, Bellingham, WA 98227-0010
Tel: +1 360 676 3290 • +1 360 647 1445 • Email: spie@spie.org • Web: www.spie.org

Robotic manipulation with stereo-visual servoing

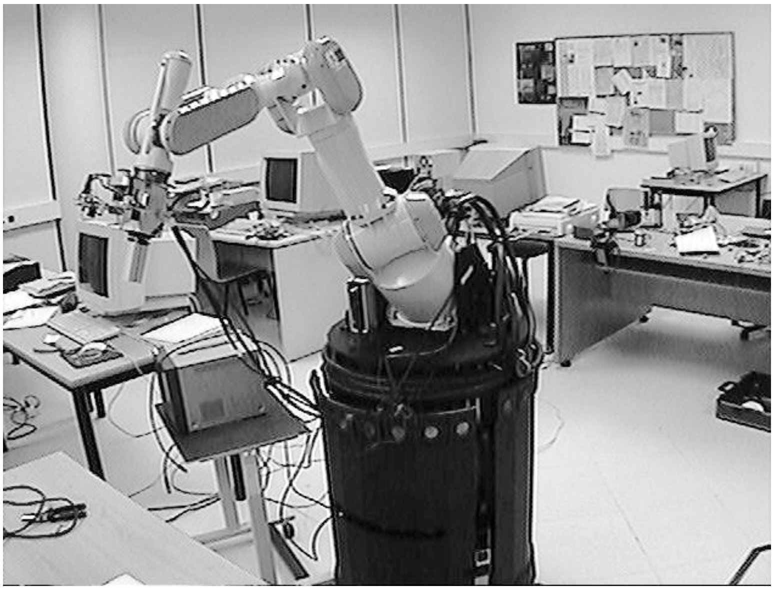

Figure 1. Shown is the set-up of the visual servoing system: a stereo rig mounted on the end-effector of a mobile manipulator.

Visual servoing is an approach to robot control based on visual perception. It involves the use of cameras in the control loop of the robot relative to its environment as required by the task. Its essence is the computation of the matrix of derivatives (the Jacobian) of the visual feature vector with respect to the motion of the robot. Whether to use raw pixel data or estimated 3D-point coordinates is a matter of choice: both approaches require the estimation of camera parameters. However, the resulting dynamic properties of the task may differ. Stereo visual servoing offers some advantages over the classical monocular 2D- and 3D-visual-servoing approaches. Depth information can be recovered without the need for any geometrical model of the observed object. It should be noted that, even in 2D visual servoing, this information would be needed for the computation of the image Jacobian.

In our work we have achieved some theoretical results concerning the modeling of a positioning task.[1] We have tested many approaches to define this kind of task when using a stereo-vision system. The vision system segments the observed object from the scene and computes its center of gravity. The image coordinates of this point, in both cameras, are the output of the vision system to the robot controller. Since 3D coordinates of the observed point can be computed from the image data (and an estimation of the extrinsic and intrinsic parameters of the cameras), they can also be used in the control law, thus getting a linear Jacobian matrix. As a result, some theoretical properties of the trajectory of the end-effector can be obtained.[2] To deal with real objects, orientation has to be taken ii»o account. Classically, axes of inertia are computed from image segmentations. These axes provide information for robot orientation with regard to the object. Though visual servoing formalisms with lines and orientation are theoretically sound, real applications are uncommon due to the difficulty of robustly extracting such primitives.

The mobile manipulator of the Robotic Intelligence Lab consists of a Nomad XR4000 platform and a Mitsubishi PA-10 arm. Attached to the end-effector of the arm is a stereo rig with two miniature CMOS NTSC color cameras, in an eye-in-hand configuration, linked to two video boards that deliver visual features at video rate. Figure 1 shows the configuration of the system. At this moment only the arm is controlled by vision: in future we hope to control the mobile base this way too. The cameras are coarsely positioned, mounted with approximately the same orientation and at equal distances from the end-effector's origin. No calibration procedure is used. The vision system is made of two inexpensive, off-the-shelf boards (*Cognachrome* by Newton Research Labs). These are video-rate color segmentation systems that extract colored regions from an image and deliver the coordinates of its centroid, its aspect ratio, and the orientation of its major axis of inertia. Each camera is connected to it's own processor.

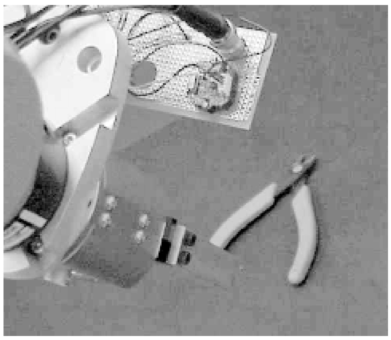

Figure 2. Close view of the end-effector, targeted to a real object (pliers).

The work-place is depicted in Figure 2 where pliers lie on a black surface and the robot is observing the object. Pliers are orange-colored, thus the blobs corresponding to the arms are segmented based on color information. Blobs are not symmetric along their inertia axes, but the system is expected to be robust against minor deviations.

Theoretical developments have shown us how to extract 3D control features from stereo images: the Jacobian matrix is computed for raw pixels, 3D coordinates are estimated, and a new feature vector that uses stereo disparity is obtained. Real experiments in adverse conditions (large rotation, noisy images, coarse calibration) show that the trajectory of the end-effector relies strongly on the features chosen for the control loop. Future work should allow us to state more precisely the robustness of the different approaches with respect to camera parameters and signal loss. We are also interested in considering other visual features and integrating them with other types of sensors.[3]

Enric Cervera*, Philippe Martinet†, and F. Berry†
*Robotic Intelligence Lab
Jaume-I University
12071 Castelló, Spain
E-mail: ecervera@icc.uji.es

†LASMEA - GRAVIR
Blaise Pascal University of Clermont-Ferrand
63177 Aubiere-CEDEX, France
E-mail: {martinet, berry}
@lasmea.univ-bpclermont.fr

References

1. E. Cervera, F. Berry, and P. Martinet, *Image-based Stereo Visual Servoing: 2D vs 3D features,* **Proc. IEEE Int'l Conf. on Robotics and Automation**, Washington DC, 2002, *(in press).*
2. P. Martinet and E. Cervera, *Stacking Jacobians Properly in Stereo Visual Servoing,* **Proc. IEEE Int'l Conf. on Robotics and Automation**, pp. 717-722, Seoul, Korea, 2001.
3. E. Cervera and A. P. del Pobil, *Sensor-Based Learning for Practical Planning of Fine Motions in Robotics,* **Int'l J. of Information Sciences,** Special Issue on Intelligent Learning and Control of Robotics and Intelligent Machines in Unstructured Environments, 2002, *(in press).*

Network robotics: task-directed configurations

Traditional robotics models a task and physical platform as a single robotic entity. The network robotics[1] approach models robotic systems as a collection of resources connected via a network that facilitates inter-resource communication. These resources are modelled as discrete software entities using the MARS[2] language, which exports a set of services that define the capabilities of the resource. The interconnection of these modules to create a *task structure,* forms a *network-robotic* agent with the ability to perform a specific task.

Previous work at the University of Reading has assumed this interconnection of modules, the task-structure, is hard coded. However, given the possibilities presented by networked robotics, this seems far too limiting. New research[3] centers around the concept of quasi-intelligent tasks building their own configuration on an *as needed* basis, introducing flexibility and simplicity to the creation of task structures. Accompanying our theoretical work has been a set of experiments that implement classic robotic tasks using the network-robotic approach. The example we will be discussing here is the problem presented by having a mobile base move safely about its environment. This can then be extended to the notion of having a robot map its environment.

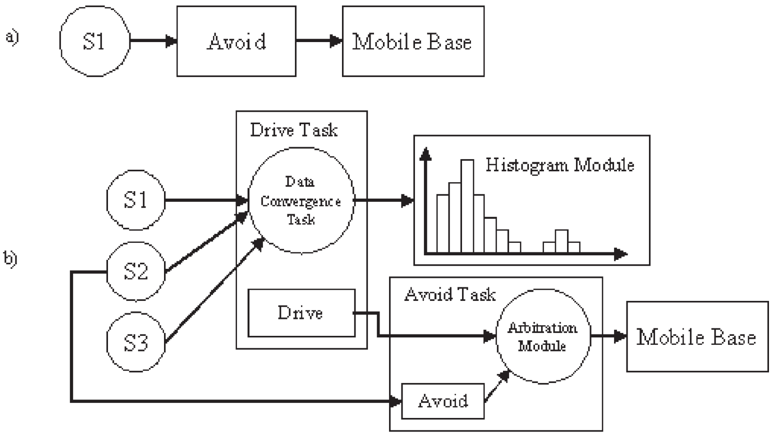

Figure 1. The (a) avoid and (b) drive task modules in an network robotic configuration experiment.

A major component of the new work is the *task module*. These modules embody successively higher-order tasks, yet operate at a similar level to the rest of the module population, allowing them to interface as *equals*. Rather than defining the task structure as a hard-coded set of connections, the task is expressed as a set of queries that will locate modules on the network that can satisfy the task's input and output needs.

Task modules are modelled as three components, namely input channels, output channels, and computational components. It is the input and output channels that form the basis for reasoning about self-configuration. They define the data types they require yet do not specify any specific source. Queries are generated that will locate modules that meet the requirements of the data channels and are broadcast to the module pool. The data channels then link with responding modules to instantiate the network robotic task structure.

The first task module, *avoid,* enables the mobile base to navigate safely about its environment. The actual avoidance behavior is based on the notion that, as the base moves forward, some mechanism should detect weather there are obstacles in its path and, if so, take steps to avoid them. This task can be broken down into two components. First, the mobile base must be in the appropriate location and have an output control channel. Second, a sensor must be mounted on the base, facing forward, with an input data channel.

These queries would be expressed using MARS modelling language as, respectively:

```
(@TransX, RotnZ) &
(is_mounted_on Room1)
```

to locate an appropriate mobile base module and;

```
(@sense 1D) & (is_mounted_on
<LocatedMobileBase> T4)
```

to locate a sensor mounted on the located mobile base module and whose position matches the translation T4, which would be a rotation of 0˚ for a forward-facing sensor. On release into the module pool, the task module would broadcast these queries and await responses. When sufficient responses had been received, satisfying all of the data channels, connections are initiated with the responding modules. The simple architecture instantiated is shown in Figure 1a.

The second task, mapping the environment, builds on the first and demonstrates how the output of a task module can be reasoned about in terms of the functionality it contributes to the module pool. This task requires that a mobile base should be able to move safely about its environment and to gather a set of sensor readings as it travels. It can be broken down into two components: a mobile base with the ability to avoid obstacles, and a set of sensors mounted on that base from which to gather data about the environment. The concept of safe movement about an environment is already embodied by the *avoid* task module. We use this module within this task, rather then re-create its functionality. Thus previously instantiated task modules can simplify the creation of further task modules.

The queries to satisfy the tasks data channels would be expressed as follows:

```
(@TransX, AVOID) &
(is_mounted_on Room1)
```

would locate an *avoid* module. MobileBase1 module would not respond directly, as it fails to fulfil the *avoid* constraint, but it would be used indirectly via its connection to the *avoid* module.

```
(@sense 1D) &
(is_mounted_on
MobileBase1)
```

would ensure that all sensors mounted on the mobile base responded. This means that the sensor used by the *avoid* module for obstacle detection provides data to two separate task modules. The instantiated architecture is shown in Figure 1b.

The above represents our initial work within this area. We intend to expand it to incorporate a wider set of resources and to look more fully at the creation of generic task modules. Specifically, our aim is the creation of a task analyser capable of the formulation of queries based on a given task, and with the ability to reason about the interaction of modules that satisfy those queries. We believe that the flexibility offered by this approach has application to the deployment of robotics technology in a wide range of industrial, office, and home environments.

Duncan I. Baker,[1] Gerard T. McKee,[1] and P. S. Schenker[2]
[1]Department of Computer Science
The University of Reading
Reading, Berkshire, England, UK
E-mail: {d.i.baker, g.t.mckee}@reading.ac.uk

[2]Jet Propulsion Laboratory
Pasadena, CA,USA
E-mail: paul.s.schenker@jpl.nasa.gov

References

1. G. T. McKee and P. S. Schenker, Networked Robotics, **Proc. SPIE 4196,** pp. 197-209, 2000.
2. G. T. McKee, J. A. Fryer, and P. S. Schenker, *Modelling Robot Architectures for Modular Robotics Systems*, **Proc. IEEE Int'l Conf. on Advanced Robotics,** Budapest, August 2001, pp. 515-521.
3. G. T. McKee, D. I. Baker, and P. S. Schenker, *Task-Directed Configuration of Networked Robotic Agents*, **IROS 2002,** 30 September - 4 October, EPFL, Switzerland, *to appear.*

Autonomous execution of force-controlled robot tasks

Compliant motion tasks are those in which the robot moves a manipulated object (a tool or a work-piece) while keeping it in contact with the objects in the environment. In industry, compliant motion tasks still require very structured environments: the positions, orientations and dimensions of tools and work-pieces or parts to be assembled are known precisely. In these cases, the robot receives and executes a nominal task plan.

In the Autonomous Manipulation Group at the Department of Mechanical Engineering, K.U.Leuven, Belgium, we are working towards the autonomous, model-based execution of compliant-motion tasks in less structured environments, such as are encountered in space, sub-sea, or nuclear installations. The same approach is relevant for future service robots operating in environments that have been designed for humans, and where even simple tasks require a lot of sensing.

Figure 1 show an example of an autonomous compliant motion task: the robot has grasped a peg and assembles it: placing it in a hole in the environmental object by executing a sequence of contact formations. None of the positions, orientations, or dimensions of either object are well known at the start. The robot executes its task based on models of the different contact formations, and on measurements of contact forces and torques and of velocities (both translational and rotational).

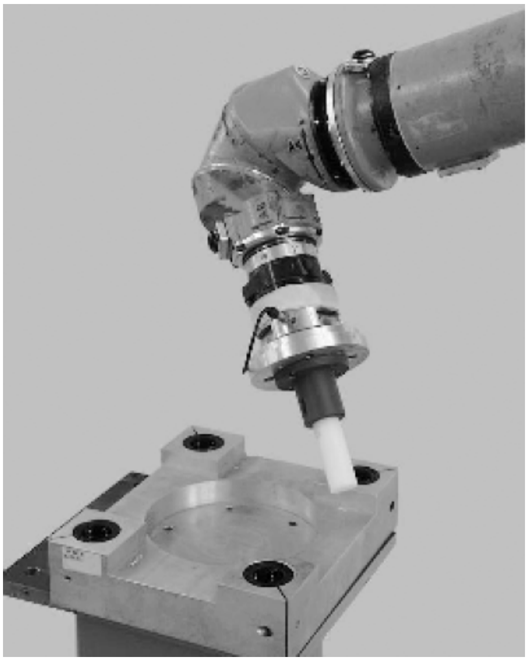

Figure 1. Peg-in-hole insertion.

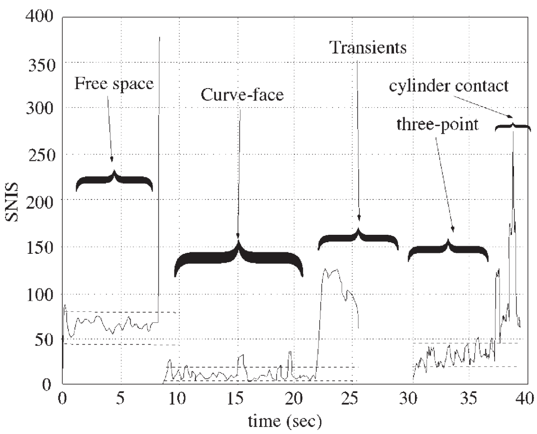

Figure 2. Detecting contact transitions and recognizing the type of contact formations during a peg-in-hole insertion.

Contact models depend both on the type of the contact formation and the geometry of the contacts: they define the space of instantaneous motion, the degrees of freedom of the manipulated object, and the space of possible contact forces. Research at our department has focused on contact modelling for several years. This has resulted in: a general geometric (quasi static) description of any contact between arbitrary contacting objects;[1] and a unified description of such contact models for contacts between polyhedral object parts by decomposition into elementary vertex-face and edge-edge contacts.[2] This allows for an automatic generation of the models during the task execution.

Uncertainty representation

The robot encounters two types of uncertainty:

1. It does not know the established contact formation type, a discrete uncertainty. The transition between two contact formations is usually quite visible in the sensor signals.

2. The robot does not know the exact geometrical parameters (positions, orientations and dimensions) of the contacting parts, this is a continuous type of uncertainty.

Besides passive force-feedback due to the compliance of the system, the robot can have different active components dealing with uncertainty:

A. For small inaccuracies in the geometrical parameters, a basic (hybrid or other) force controller allows the robot to keep the contact and assures a good task execution. We use a hybrid force/position controller where force and tracking control loops are closed around a velocity controller.[3] Tasks are then specified by set points in the force-controlled and velocity-controlled directions, which depend on the contact geometry. Measured velocities and forces are projected on their respective subspaces, the control laws are performed in each subspace separately, and afterwards the results are combined.

B. Larger uncertainties are dealt with by adding an estimation component that extracts from the measurements both the type of the contact formation (see Figure 2 for the peg-in-hole insertion) and estimates for the geometrical parameter values.[4] Due to the non-linearities in the geometrical models, the estimation problem is not straightforward. We developed an efficient Kalman-Filter-based estimator able to process the measurements at measurement frequency (10Hz).

C. We are currently working on a third component for dealing with even less structured environments: the active-sensing (re)planning component. This will allow the robot to deviate from its nominal task execution in order to obtain more persistent measurements: thus reducing the uncertainty of the estimator—both for the recognition of the contact formation and for the estimation of the geometrical parameters—as much as possible. The active-sensing component looks for an optimum between the reduction of uncertainty and the cost of task deviations, while respecting constraints such as the robot work-space limits, maximum velocities, accelerations, robot-joint torques, contact forces, etc..

T. Lefebvre and H. Bruyninckx are, respectively, Doctoral and Postdoctoral Fellows of the Fund for Scientific Research-Flanders (F.W.O.) in Belgium.

Tine Lefebvre, Herman Bruyninckx, and Joris De Schutter
Department of Mechanical Engineering
Katholieke Universiteit Leuven, Belgium
Tel: +32 16 32 24 80
Fax: +32 16 32 29 87
E-mail: Tine.Lefebvre@mech.kuleuven.ac.be
http://www.mech.kuleuven.ac.be/pma/research/manip/

References

1. H. Bruyninckx, S. Demey, S. Dutré, and J. De Schutter, *Kinematic Models for Model-Based Compliant Motion in the Presence of Uncertainty,* **Int'l J. of Robotics Research 14** (5), pp. 465-482, October 1995.
2. T. Lefebvre, H. Bruyninckx, and J. De Schutter, *Estimation and Propagation of Geometrical Parameters During Force-Controlled Execution of Polyhedral Contact Formation Sequences,* **Proc. Int'l Conf. on Advanced Robotics,** Budapest, Hungary, August 2001, pp. 85-90.
3. J. De Schutter and H. Van Brussel, *Compliant robot motion I and II,* **Int'l J. of Robotics Research 7** (4), pp. 3-33 ,August 1988.
4. J. De Schutter, H. Bruyninckx, S. Dutré, J. De Geeter, J. Katupitiya, S. Demey, and T. Lefebvre, *Estimating First Order Geometric Parameters and Monitoring Contact Transitions During Force Controlled Compliant Motion,* **Int'l J. of Robotics Research 18** (12), pp. 1161-1184, December 1999.

Building flexible multi-robot systems

My work, in collaboration with Marco Somalvico[1] aims to define a methodology and architecture to develop flexible multi-robot systems: our approach is called dynamic agency.[2] It allows the development of cooperative multi-robot systems in which the cooperative behaviour of the robots is explicitly programmed by the designer, as opposite to multi-robot systems in which the cooperation behaviour is only implicitly programmed in and emerges as the robots interact with each other and with the environment.

Our main idea is to conceive a robot as an agent structured in two parts. The first part, called an *op semi-agent*, is composed of the hardware and basic software components of the robot. These components provide the abilities to operate within the given environment. The hardware components include sensors, actuators, processing units, and communication devices; the software components include control systems for the sensors and actuators, operating systems for the processing units, programs for managing the communication protocols, and so on.

The second part, called the *co semi-agent*, is composed of high-level software modules devoted to cooperation: they integrate the op semi-agents within a framework that allows uniform and coherent cooperation. In particular, they provide functions for negotiation, for division of tasks, for high-level knowledge exchange, etc..

So, in the dynamic agency approach, each agent of a multi-robot system is composed of the op and co semi-agents. The software modules of the co semi-agents are implemented using mobile-code-system techniques.[3] Accordingly, the co semi-agents are built by execution units (software processes) that can migrate through the network, connecting the robots from one host to another, and resume their execution from the point where they were interrupted. The op semi-agents are the hosts on which the execution unit runs. In our methodology, the co semi-agents are spread on the op semi-agents by a unique execution unit that replicates and evolves on each one of them.

To build a multi-robot system according to the dynamic agency methodology, the robots are firstly adapted to be op semi-agents that can host the execution units, then a cooperation mechanism is installed by spreading the co semi-agents. The dynamic agency methodology enables the easy management and substitution of the co semi-agents providing a number of advantages, only partially demonstrated in our experimental activity. First of all, the designers of the robots (op semi-agents) are independent from the designers of the whole multi-robot system. Moreover, the re-use of the existing op semi-agents for different purposes is facilitated, since different co semi-agents can be installed on them at different times. It is also possible to conceive multi-robot systems that can automatically reconfigure themselves, by adding or eliminating robots, during their operation.

In order to validate our approach, we developed a multi-robot system composed of mobile robots for mapping unknown environments. The multi-robot system is composed of four agents: three mobile robots (shown in Figure 1) and a computer. The agents communicate through a wireless local area network. The co semi-agents, built by a replicating execution unit, first extract knowledge about the corresponding op semi-agents: how to activate their functions, their physical dimensions, and so on. Then the agents are organized in a hierarchical structure, in which the computer coordinates the explorer robotic agents. Then, the co semi-agents negotiate the areas worth exploring and, when the explorer robots have extracted the segments from the images taken by their vision systems at the assigned locations, these segments are integrated by the co semi-agents in a unique global map. Using this system, we have been able to map some of our departmental hallways.

The possibilities offered by the dynamic agency approach extend far beyond what we described here. We are currently investigating the automatic reconfiguration of the multi-robot system: the computer agent might decide to recruit a new small robot to map a newly discovered room accessible by a narrow doorway. Moreover, we developed a first prototype to show how the multi-robot system can automatically switch from the task of exploring to the task of sweeping as more of the floor is mapped.

Francesco Amigoni
Dipartimento di Elettronica e Informazione
Politecnico di Milano
Piazza Leonardo da Vinci
32I-20133 Milano (MI), Italy
Tel: +39 02 2399-3475
Fax: +39 02 2399-3411
E-mail: amigoni@elet.polimi.it
http://www.elet.polimi.it/people/amigoni/

References

1. L. E. Parker, *Current State of the Art in Distributed Robotic Systems*, in L. E. Parker, G. Bekey, and J. Barhen (eds.), **Distributed Autonomous Robotic Systems 4,** pp. 3-12, 2000.
2. F. Amigoni and M. Somalvico, *Dynamic Agencies and Multi-Robot Systems*, in T. Lueth, R. Dillmann, P. Dario, and H. Worn (eds.), **Distributed Autonomous Robotic Systems 3,** pp. 215-224, 1998.
3. A. Fuggetta, G. P. Picco, and G. Vigna, *Understanding Code Mobility,* **IEEE Transactions on Software Engineering 24** (5), pp. 342-361, 1998.

Figure 1. The three mobile robots used in the dynamic agency experiments: those on the left and right were built entirely at the Politecnico di Milano.

Robotics research at ALTAIR

The Laboratory for Teleoperation and Autonomous Intelligent Robotics (ALTAIR) is a newcomer to the family of robotics laboratories, having been founded just a year ago at the Department of Computer Science at the Unviersity of Verona. The laboratory is currently acquiring equipment and personnel to carry out research and development in three main areas: service robotics, teleoperation, and field robotics. Collaborations are being established with leading laboratories to address specific research topics. One of the projects in the area of teleoperation, for example, is cooperating with the Istituti Ortopedici Rizzoli (IOR) in Bologna towards the further development of a robot for orthopaedic surgery. The surgical system was developed by IOR, and is shown in Figure 1. It consists of a supporting cart, a localization device, and the surgical robot. The cart provides a rigid connection between the robot, the measuring device, and the surgical bed. We describe here a general method for computing the workspace of the robot and for its calibration.

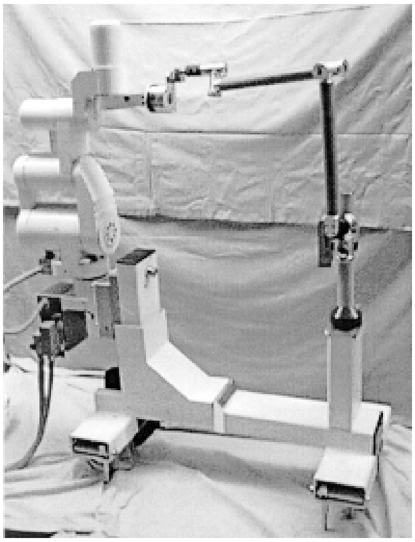

Figure 1. Robot for orthopaedic surgery developed by the Istituti Ortopedici Rizzoli (IOR) in Bologna.

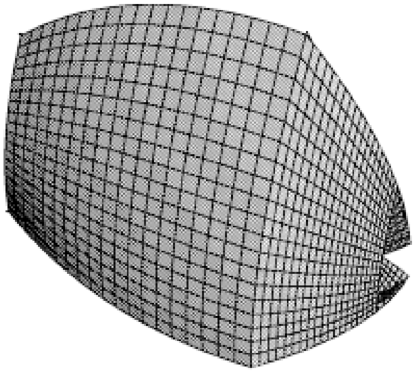

Figure 2. Workspace computation for the IOR surgical robot.

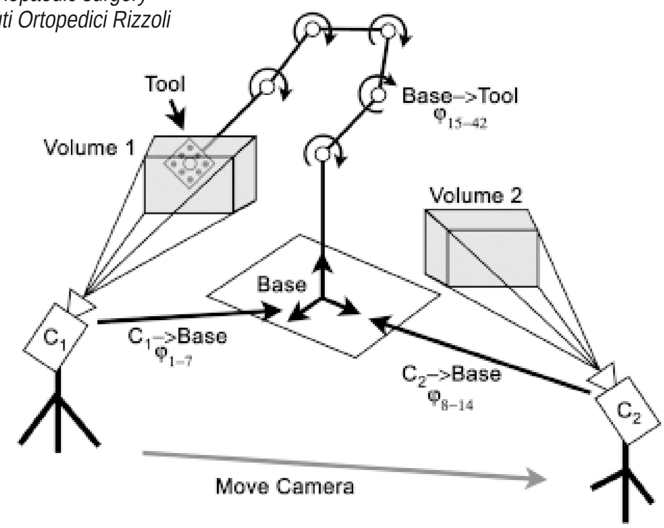

Figure 3. Robot calibration using a mobile camera.

Robot workspace analysis

The applicability of robotic manipulators depends strongly on the characteristics of their workspace. In fact, given a specific task and its space requirements, the manipulator must be positioned (or designed) to satisfy those workspace constraints. In particular, non-conventional robots, such as surgical manipulators, require precise workspace knowledge to satisfy task constraints optimally. To address this problem with reference to the IOR surgical robot, we have developed an iterative algorithm based on analytical and geometric analysis of the workspace. The workspace of each joint is represented as a single volume, possibly degenerate, described by the collection of analytical patches forming its boundary. The volume is rotated or translated, depending on the next joint type. This approach is consistent for all joints, does not require geometric simplifications, and directly accounts for joint limits.

Workspace computation is based on the iterative rigid sweeps of the volume representing the previous workspace. Rigid sweeps are defined as the set product of a generatrix and a directrix: the generatrix is a curve, usually closed, lying on the plane normal to the tangent to the directrix; and the directrix is a planar curve that prescribes the direction and magnitude of the rigid sweep. The computation consists of computing the workspace generated by the the manipulator end effector, and then iteratively sweeping the resulting volumes through the joint ranges, moving toward the base. After an initial step, in which the algorithm computes the surface representing the workspace corresponding to the first two joints, the iterative procedure starts with the third joint, when the workspace becomes a volume. By considering the workspace boundary as a set of independent surfaces, the complete workspace can be computed by sweeping each boundary surface, computing the intersections among the resulting volumes, and removing all internal surfaces.

The result is a workspace represented as a collection of boundary patches, to which the algorithm can be recursively applied until the base joint is reached. The central part of the algorithm is the i^{th} step, which receives as input workspace w(i+1) and computes w(i). Computational complexity analysis shows that the algorithm complexity is $O(n^3)$. This has been proved by analyzing the different parts of the algorithm and verified experimentally by running the algorithm with manipulators with successively increasing joint numbers.

Preliminary results using the algorithm confirm the validity of the idea, since workspace characteristics can be easily analyzed. Furthermore, rapid queries can be developed to verify the reachability of specific points. Workspace can be also visualized from different viewpoints, and with different sections. The workspace computation for the IOR surgical robot is shown in Figure 2.

Robot mobile calibration

Calibration refers to the determination of unknown, or uncertain, parameters in robot kinematics, and to the precise localization of the robot within its environment. Typically, a robot is calibrated when it is set in its final position. However, this approach is not possible if calibration must be frequently repeated. For example, surgical robots cannot be permanently installed in an operating room (OR), and must be moved out for sterilization and use in other ORs. Clearly, transportation can alter the robot and the new OR setting may be different.

The method that we have developed has several new features that also make it interesting for applications outside surgery. It allows the calibration of a large work volume by connect-

Continues on page 11.

Beyond teleoperation: an architecture for networked autonomous robots

In recent years, due to advances in computer and network communication technologies, research in networked robotics—and especially on internet robotics—have become very active. The main trend is to focus on two key areas: telepresentation, where information from remote sensors is transmitted, and teleoperation, where a remote robot device is controlled. Both of these deal with the use of remote hardware resources through a network.

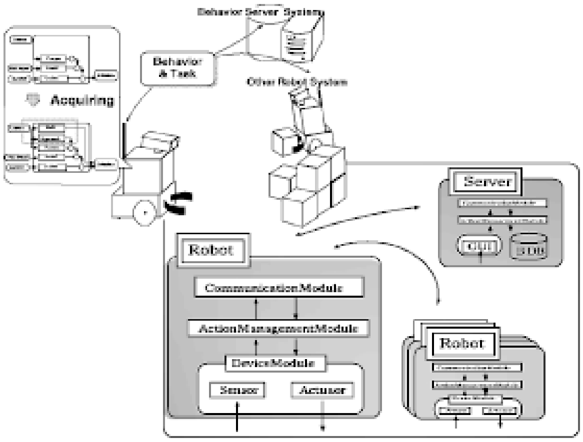

Figure 1. Concept and basic architecture of the Little Object-Oriented Grounded User Environment for robot operation (LOGUE).

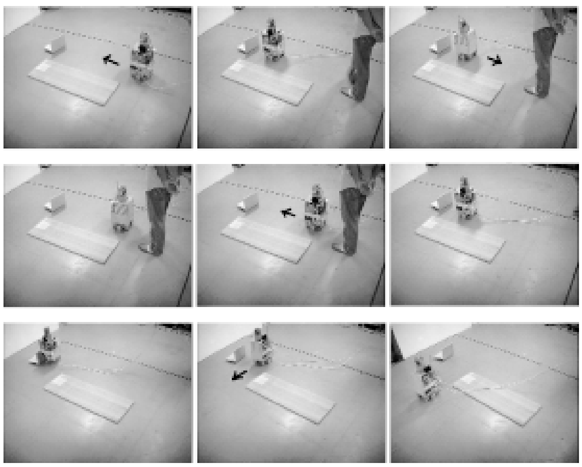

Figure 2. Because of an unknown obstacle (a person), the robot's EscapeBehavior prevents the CruiseLeftBehavior from reaching its goal. The server sends a new BEO called pushObstacle, the robot removes the obstacle, and finally reaches the goal position.

In the case of teleoperation, the methods of socket-based communication and remote procedures are frequently used for transmitting parameters among robots or between robot and host. These can be found in research related to master-slave remote control systems,[1] which have to cope with large time delays caused by the network being in the control loop. Other research,[2] using JAVA RMI, demonstrated how some off-line planning path or command sequence could be transferred to a teleoperated mobile robot from a web-based interface. However, the whole process is still strictly under the operator's control rather than being performed by the robot itself.

When using a system consisting of multiple autonomous robots, each one should have all the necessary information about a given task in order to perform it. However, if the robots are connected to a network, we can construct a robot system that can acquire task information on-line and perform it in real time. This information can either be acquired from other robots that have already performed the same task, or from a server system on a network which has a database of the different object behaviors and tasks.This strategy has two major advantages. First, it will increase the ability of an autonomous robot to perform new tasks and allow it to cope with changes in the environment. Second, the size of the robot control system can be small since it only requires basic control functions: a smaller control system is particularly helpful when designing a multiple robots system, especially as the number of robots increases.

In the software engineering area, plug-in technology and software auto-updates demonstrate the advantages of dynamic reconfigurable systems. These applications hint at possible solutions, but are not enough. The controller of a robot involves many sensor and actuator devices that interact with the real world and have some characteristic that devices in a computer system do not. For example, you cannot simply reset a gripper's input to zero when it is holding an object. Thus, some action recovery procedures must be introduced.

Object representation of behavioral elements and task

We have developed a Java package,[3] called 'Little Object-Oriented Grounded User Environment for robot operation' or LOGUE, and designed a basic architecture that allows task and behavior information to be transferred and which performs checks and executes procedures while the robot is running. Our strategy has concentrated on sharing software resources among agents in a networked robotics system, not hardware resources. The action-decision of the robot is realized using a behavior-based architecture, since this makes the information about the task both sufficiently abstracted and easily reusable.

Two types of information are needed: information about the behavior and information about the task.The former is represented by the behavioral element object (BEO). Each one has a unique name used as the key for searching and checking its existence in a behavior-object database. A BEO has a list of devices native to the robot, including those sensors and actuators specifically involved in the implementation of this behavior. This allows it to check for the existence of those devices when it is transferred to the machine. A task is represented by task object (TO) containing all the information about how the robot behaves. The TO is constructed as a module of the network among the BEOs and also includes starting and ending conditions and priority.

Object transmission and environment management

The whole LOGUE system is shown in Figure 1. It can be run on both autonomous robots and behavior-server systems and includes three component modules. First, the communication module allows for TO and BEO transmission. The action-management module, which includes the behavior-manager and task-manager modules, works on reconstructing objects transferred from the network, checking executive ability, storing them, and running the task objects. The devices module is designed to provide a common interface from the robot's native devices—such as sensors and actuators—and some sepecific robot control parameters.

The LOGUE was implemented on the Java™ JDK1.3 running on Linux 2.2 and has been tested on simulation, experimental robot, and behavior-server systems. For object transmission among robots and the behavior-server, the distributed-object-environment java.rmi package was used. Except for the behavior database and GUI interface included in behavior-server system, the implementations used in the autonomous robots and the server are the same.

In situations where a task must be interrupted in case of emergency, or a BEO must be replaced even while it is still active, special accommoda-

Continues on page 11.

The need for, and application of, 'soft' actuators

Continued from cover.

Table 1. Comparison of pMA and natural muscle.

Parameter	Natural Muscle	pMA Muscle
Displacement	35%	35%
Force/cm^2	20-40N	100-500N
Power/weight	40-250W/kg	500-2kW/kg
Efficiency	45-70%	32-50%
Rate of Contraction	25-2000%/s	35-700%/s
Bandwidth		5Hz
Control	Good	Fair-Good
Operation in water	Yes	Yes
Temperature range	0-40°C	-30-80°C
Robustness	Excellent	Fair-Good
Self repair-regeneration	Yes	No
Antagonistic Operation	Yes	Yes
Compliance/Impedance Control	Yes	Yes
Energy Source	Chemical	Pneumatic
Environmentally Safe	Produces CO_2	Yes
Scalable from	µm-m	cm-m
Linear Operation	Yes	Yes

the dilated length.

- Being pneumatic in nature, the muscles are highly flexible, soft in contact, and have excellent safety potential.
- Operating safely in aquatic or other liquid environments and being safe in explosive/gaseous states.
- Having lateral and rotational tolerances that mean that accurate alignment is not necessary, thus allowing rapid, low-tech construction.
- Producing variable compliance structures and storing energy when operated with an antagonist.

Biologically-inspired robots

The functional similarity between natural and pneumatic muscles suggests they are well-suited to the field of biologically-inspired robotics, and particularly application to anthropomorphic robots. Therefore, the basic anatomical model of a living creature would be a good point of initial mechanical investigation, although it is clear that the range of muscles in organic entities is beyond direct duplication.

To demonstrate the utility of the pMAs, the structures of the human legs (bipedal robot), arms (14-degree-of-freedom or 14-dof twin-armed robot) and hand (20-dof dextrous manipulator) have been built and tested. These robotic units combine the anatomical layout and motion of the human body with the the soft, high-power actuation format of the pMA and engineering composites to provide the functionality (though not the anatomical complexity) of the respective limbs.

Table 2. Characteristics of robot primate.

Weight	25Kg
Height	1.7m
Dof	18
control	On Board PC

These features have recently been united in a single simian robot based on the physical structure of a female gorilla, combining a stable quadrupedal walking platform with manual dexterity: see Figure 2. The characteristics of this primate are shown in table 2.

Despite braided pMAs offering very high power/weight performance, many researchers still resist using them because of perceived failings in terms of the following.

- Low bandwidth: the bandwidth is often considered to be too low for practical success in many applications, particularly robotics.
- Low stiffness: compliance regulation is one of the benefits of the actuators as antagonistic pairs, but some researchers feel that the peak stiffness is not adequate in all applications.
- Limitation of linear displacement: dimensional changes of 30-35% are possible but could this be increased?
- Energy usage: as with all actuators, there is a continual goal of trying to reduce the energy (air) consumption of this actuator.

Recent work at the University of Salford has addressed these problems and shown that stiffness and bandwidth improvements of over 500% are feasible, along with increased displacement and a reduction in power requirements of up to one third. This makes the dynamics of pMAs substantially *better* than that of natural muscle.

Steve Davis, N. G. Tsagarakis, and Darwin G. Caldwell
Department of Electronic Engineering
University of Salford
Manchester M5 4WT, UK.
E-mail: {S.Davis,
D.G.Caldwell}@salford.ac.uk

References

1. D. G. Caldwell, G. A. Medrano-Cerda, and M. Goodwin, *Characteristics and Adaptive Control of a Pneumatic Muscle Actuator for Robotic Elbow,* **ICRA,** 1994.
2. N. Tsagarakis and D. G. Caldwell, *Improved Modelling and Assessment of pneumatic Muscle Actuators,* **ICRA,** 2000.
3. H. F. Schulte, *The Characteristics of the McKibben Artificial Muscle,* **Application of External Power in Prosthetics and Orthotics**, National Academy of Science, NRC, Appendix H, pp. 94-115, 1961.

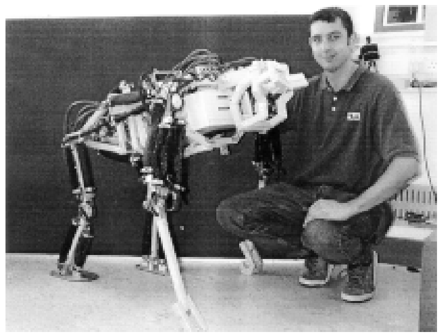

Figure 2. A primate robot combining the advantages of quadrupedal walking with dextrous hands.

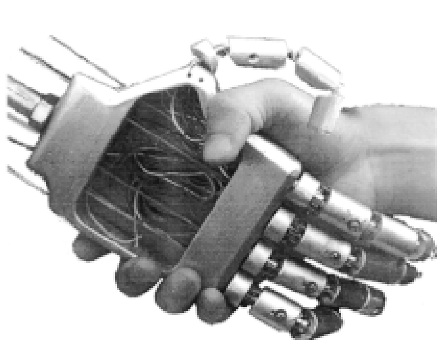

Figure 3. Dextrous manipulator powered by pneumatic muscle actuators (pMAs).

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 the **Robotics & Machine Perception Technical Group.** Membership allows you to communicate and network with colleagues worldwide.

As well as a semi-annual copy of the *Robotics & Machine Perception* newsletter, benefits include SPIE's monthly publication, *oemagazine*, a 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
Tel: +1 360 676 3290
Fax: +1 360 647 1445
E-mail: spie@spie.org

http://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

Robotics Web Discussion Forum

You are invited to participate in SPIE's online discussion forum on Robotics. The **INFO-ROBO** mailing list has been '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 also a full FAQ for more details on how to use the forums.

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

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

Robotics & Machine Perception

This 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	*Managing Editor*	Linda DeLano
Technical Editor	Sunny Bains	*Advertising Sales*	Gayle Lemieux

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.

Copyright ©2002 Society of Photo-Optical Instrumentation Engineers. All rights reserved.

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

Robotics research at ALTAIR

Continued from page 7.

ing several high precision sub-volumes, and it can be adapted into a procedure for a portable calibration, using a mobile device. It uses an external camera that can be positioned without affecting the OR equipment. However, the small viewing volume of the camera necessitates the development of a new geometric model of the robot.

Robot calibration using a mobile camera (see Figure 3) provides the solution to two separate problems: robot calibration and localization with respect to the OR. Our calibration approach requires a model separating the kinematic from the localization parameters to support camera views from different locations. For this purpose, we developed a new linearized kinematic model that supports the merging of data from different locations, with different reference frames, and calibration of the resulting collection of camera-viewing volumes. By analyzing the model Jacobian, the procedure is also able to identify the set of camera positions and robot poses that will yield the best parameter set.

To calibrate the robot over the complete workspace, we need to define the error Jacobian with respect to different camera locations. This Jacobian defines a linear relation between the error in all model parameters and the final tool error in all configurations. By analyzing the singular values of the Jacobian, we can establish whether all the parameters are identifiable. Camera positions and robot poses are selected to avoid instability in parameter identification. This is achieved by computing the Jacobian condition number indicating whether the unknown parameters are easily identifiable within the measured data. The identification computes the kinematic parameters and localization parameters, i.e. the first seven kinematic parameters of the model.

The method was verified using extensive simulations with the kinematic model of the IOR manipulator. To verify convergence properties, initial parameter values are generated by modifying the true values with an array of zero mean and 50mm/mrad variance variables. Simulation results encourage the development of laboratory experiments to verify the characteristics of the algorithm and its real applicability to different robot applications.

The work described here was performed by Debora Botturi and Gianni Campion while undergraduate students at the University of Verona.

Paolo Fiorini
Department of Computer Science
University of Verona
Ca’ Vignal 2, Strada le Grazie 15
37134 Verona, Italy
Tel: +39 045 8027963
Fax: +39 045 8027928
E-mail: paolo.fiorini@univr.it
http://metropolis.sci.univr.it

References

1. C. Casadei, S. Martelli, and P. Fiorini, *A Workcell for the Development of RobotAssisted Surgical Procedures,* **J. Intelligent and Robotic Systems 28,** pp. 301324, 2000.
2. G. Campion and P. Fiorini, *Robot Calibration using Mobile Camera,* **IEEE Int’l Conf. on Robotics and Automation (ICRA 2002),** Washington, DC, USA, May 2002.

Beyond teleoperation: an architecture for networked autonomous robots

Continued from page 7.

tion is necessary. This is because a task or a behavior running on a robot may involve some physical interaction with the real world, such as holding an object or moving at some velocity. As these issues depend on the robot’s native devices, they must be handled carefully: a basic mechanism has been designed for this.

Conclusion

Automatically acquiring and running tasks or behaviors from networked robots or servers not only allows a robot to keep its control system small but also allows it to act beyond its own sensing and data-storage capability. This is particularly significant as access to the wide range of network-connected sensor information can be vital in unpredicted or emergency situations. Also, it may be possible to apply the proposed architecture to a learning-based multiple-robot system as a means of sharing the most successful behaviors.

Zhidong Wang, Takayuki Takahashi, and Eiji Nakano
Graduate School of Information Sciences
Tohoku University, Aobayama 01
Sendai 980-8579, japan
E-mail:{wang, taka, nakano}
@robotics.is.tohoku.ac.jp
http://www.robotics.is.tohoku.ac.jp/

References

1. J. Kikuchi, K. Takeo, and K. Kosuge,*Teleoperation System Via Computer Network for Dynamic Environment,* **IEEE Int’l Conf. on Robotics and Automation,** p. 3534, May 1998.
2. P. G. Backes and K. S. Tso, et.al, *Internet-based operations for the Mars Polar Lander mission,* **IEEE Int’l Conf. on Robotics and Automation,** p. 984, May 2000.
3. Z. D. Wang, T. Takahashi, and et.al, *An Architecture for Task and Behavior Object Transmission Among Multiple Autonomous Robots,* **Intelligent Autonomous Systems 7,** p. 365, March 2002.

Biologically-inspired, visually-guided navigation of a flying robot

Continued from page 12.

Our flight experiments in an unstructured office environment show that this control strategy can be applied to altitude stabilization, route stabilization, and obstacle avoidance.[3,4] In addition to these low-level control experiments, we have also tested a visual odometer that measures the distance travelled, and which has been also observed in bees’ foraging behavior. This work is aimed at the implementation of long-distance navigation. Further quantitative analysis suggests that the method under consideration relies strongly on the spatial structure experienced by robot, but it is also highly adaptable through the tuning of various parameters such as the number of EMDs, spatial and temporal redundancy, sensory-motor connections, and other sensory information. In addition, analysis through simulation has lead to better understanding of both the relationships among these parameters and the general design principles, as characterized by the concept of ‘cheap vision’.[5]

It is true that the control of our blimp-type robotic platform is far simpler than for others (such as helicopters). However, by enhancing this ‘cheap vision’ approach, it should eventually be possible to realize more sophisticated controls for more demanding situations with a simpler architecture: as evolution has found solutions for flying insects.

Fumiya Iida
Artificial Intelligence Laboratory
University of Zurich
Winterthurerstrasse 190
8057 Zurich, Switzerland
E-mail: iida@ifi.unizh.ch
http://www.ifi.unizh.ch/ailab/

References

1. M. V. Srinivasan, M. Poteser, and K. Kral, *Motion detection in insect orientation and navigation,* **Vision Research 39,** pp. 2749-2766, 1999.
2. A. Borst and M. Egelhaaf,*Visual Motion and its Role in the Stabilization of Gaze,* in **Detecting visual motion: Theory and models,** Eds. F.A. Miles and J. Wallman, Elsevier Science, pp. 3-27, 1993.
3. F. Iida, *Goal-directed navigation of an autonomous flying robot using biologically inspired cheap vision,* **Proc. of the 32nd ISR** (Int’l Symp. on Robotics), pp. 1404-1409, 2001.
4. F. Iida, *Biologically inspired visual odometer for navigation of a flying robot,* **Proc. of Intelligent Autonomous Systems 7,** pp.142-149, 2002.
5. R. Pfeifer and D. Lambrinos, *Cheap Vision—Exploiting Ecological Niche and Morphology,* **Theory and practice of informatics: SOFSEM 2000, 27th Conference on Current Trends in Theory and Practice of Informatics,** Milovy, Czech Republic, Vaclav Hlavac et al. Eds., pp. 202-226, 2000.

Biologically-inspired, visually-guided navigation of a flying robot

Figure 1. The autonomous flying robot, Melissa, and its gondola.

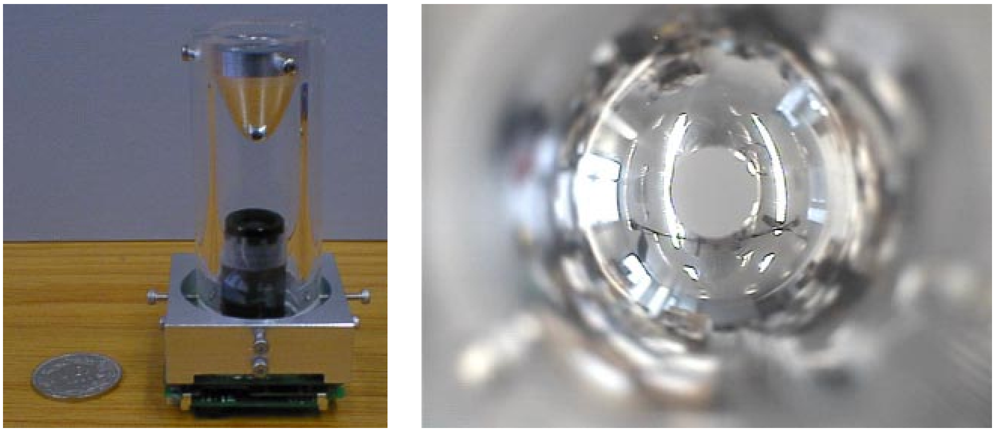

Figure 2. A panoramic vision system with a hyperbolic mirror, and an omni-directional image obtained in an unstructured environment.

In nature, flying insects navigate through a complex environment in a robust manner, despite their tiny brains. Behavioral studies with insects have revealed that the vision systems of flying insects are exquisitely sensitive to motion, because visual motion induced by ego-motion can tell the animal much about its own motion and also about the structure of its environment. Behavioral experiments with flies and bees show a number of different ways in which insects use cues derived from optical flow: from the low-level controls to the high-level cognitive tasks such as flight stabilization, safe landing, landmark navigation, object recognition, etc. (for review, see Reference 1). Compared to nature, however, artificial autonomous aerial vehicles rely heavily on external devices such as beacons, GPS, or other reliable sensory devices (inclinometers, accelerometers, sonars, laser-range-finders etc.), simply because passive visual processing is too complex and computationally demanding. The objective of our project is, therefore, to understand the underlying mechanisms and design principles of visually-mediated navigation of autonomous flying agents by using a synthetic methodology, i.e. the knowing-by-building strategy.

For this purpose, we developed an autonomous flying robot, shown in Figure 1. *Melissa* is a blimp-like flying robot that consists of a helium balloon, a gondola hosting the on-board electronics, and panoramic camera device, and a host computer. The balloon is 2.3m long and has a lift capacity of approximately 500g. Inside the gondola, there are three motors for rotation, elevation and thrust control, a four-channel radio link, a miniature panoramic vision system, and the batteries. A biologically-plausible wide visual field is realized by the panoramic vision system, which has a mirror with a hyperbolic surface. This provides a visual field of 360˚ in the horizontal plane and 260˚ vertically (see Figure 2).

In the controller of our flying robot we used a biologically-based motion detector: the so-called Elementary Motion Detector (EMD), that has been proposed based on neuro-physiological studies of flying insects (for a review, see [2]). We applied this EMD model to the panoramic image obtained from the omni-directional mirror, which measures horizontal and vertical optical flow. The controller of the robot used in the experiment is purely based on vision: optical flow obtained in both right and left lateral visual fields is used to control forward translation speed; course stabilization is realized by balancing the right and left speeds; vertical translation speed is controlled by measuring vertical optical flow; rotation is controlled by measuring front and back image speed.

Continues on page 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