VOL. 9, NO. 2

Special Issue: Sensing and Control

Edited by Gerard McKee

NOVEMBER 2000

SPIE's International Technical Group Newsletter

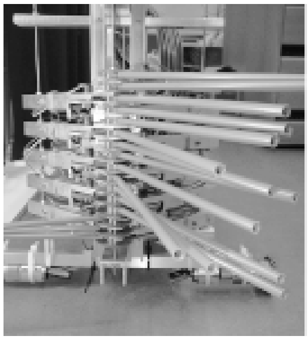
Robotics and Machine Perception

Adaptive hardware for evolving the sensor morphology of a compound eye

Robots are always designed to perform a certain range of specific tasks in a given environment. The design choices for many of the system's parameters are usually done "by hand" and depend on models of the expected robot-environment interaction. In most cases these models cannot describe actual real-world conditions very accurately. Adaptive methods (such as learning and artificial evolution) can be employed to automatically re-adjust some of the system's parameters to match the conditions found in the real world. However, this is usually limited to the control architecture of the robot. When it comes to exploring alternative solutions for the robot's morphology (body construction, placement and properties of individual sensors and actuators, etc.) the options for adjustments are quite limited once the robot has been designed. By contrast, all biological systems can automatically adapt their morphology (through evolution and development) to a specific task environment. It turns out that an adequate morphology can often considerably simplify the control effort required by solving problems already on the "physical level". This allows for designs that are "cheaper" with regard to computation, energy, and cost.

For a systematic exploration of morphological solutions for a given task environment, we built a robot[1] that is inspired by the compound eye of insects. Insect compound eyes show a large morphological diversity and parameters like the density distribution of individual facets (ommatidia) differ strongly between species, and sometimes even between the sexes. Our robot (see Figure 1) consists of a chassis, an on-board controller, and sixteen independently-controllable facet units, which are all mounted on a common vertical axis. Each facet unit consists of a motor, a potentiometer, two

continued on p. 7



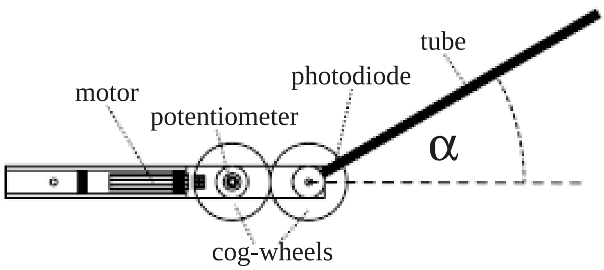


Figure 1. (Left) A robot that can automatically adapt its sensor morphology. Each of the 16 long tubes (facet units) contains a light sensor. The tubes can be rotated about a common vertical axis using electrical motors. (Right) A single facet unit.

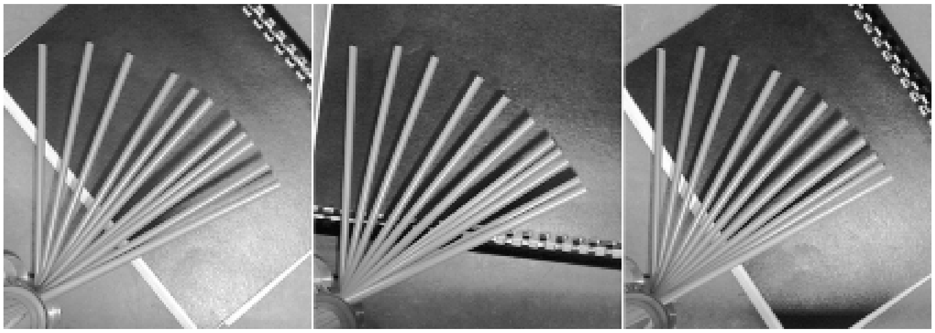


Figure 2. Non-homogeneous compound eye morphologies evolved for the task of estimating lateral distance to obstacles. Best individuals from three different runs are shown. The robot is moving from left to right.

Newsletter now available on-line

Technical Group members are being offered the option to receive 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.

Editorial

Sensing and control

Welcome to this issue of the R&MP technical group newsletter. The theme of the issue is sensing and control. The articles cover a range of topics in this area that are developed in more detail at a forthcoming ISAM (Intelligent Systems and Advanced Manufacturing) meeting, part of SPIE's Photonics East, Boston, 5-8 November 2000.

We hope you will enjoy reading the articles and we welcome your own contributions to ISAM and to future issues of this newsletter.

We have been promising for some time now to set up a wwweb site for the Technical Group. I am pleased to announce that the site, housed and supported by SPIE, is now live. My thanks to SPIE staff for their assistance in setting it up. The site is to be of benefit to members, so please forward any comments and suggestions you might have regarding content and presentation. The URL is given below.[1]

I would also like to alert you to the forthcoming Technical Group meeting which will be held during ISAM 2000. In recent years we have included an invited presentation. The subject of the forthcoming talk is Microsoft's EasyLiving project, which aims to bring vision and robotics technology together in the living room.[2] You are all welcome to attend. The 1999 talk was given by Lynne Parker and Robin Murphy, and entitled "We Don't See Eye to Eye". The slides from Lynne and Robin's presentation can be found on the R&MP site.[1]

Finally, we are developing a range of activities to reflect and focus the interests of the technical group. These will be announced through the wwweb site and the newsletter. We welcome your input and your suggestions.

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

References

1. http://www.spie.org/info/robotics
2. http://www.research.microsoft.com/easyLiving/

A new hierarchical approach to pattern matching

Applications of pattern matching in computer vision range from visual inspection of printed circuit boards to robot navigation. In such applications, the pattern matching algorithm has to exhibit a certain robustness against performance degrading influences such as changes in illumination, partial occlusion of objects, and spurious data. Whereas pattern matching systems for industrial vision applications were often based on correlation methods, more recent research has focused on object shapes rather than on texture.[1,2] In shape-based systems, an object is represented by a model typically comprised of a set of salient features found during the training process.

A typical example application is shown Figure 1. During the training process (left image), a region of interest (ROI) is selected by the user as indicated by the white rectangle. Features found within this part of the training image are then used to build an object model. The task of the pattern matching algorithm is to find a possibly rotated and scaled instance of the object in a search scene as indicated by the white rectangle in Figure 1(b). The spatial relationship between the object model and its corresponding instance in the search scene can be described by means of a mathematical transform. The problem of feature-based pattern matching, therefore, involves searching the scene for instances of the object model and acquiring the associated transform parameters.

Figure 1. Matching example. (a) training image, (b) test image.

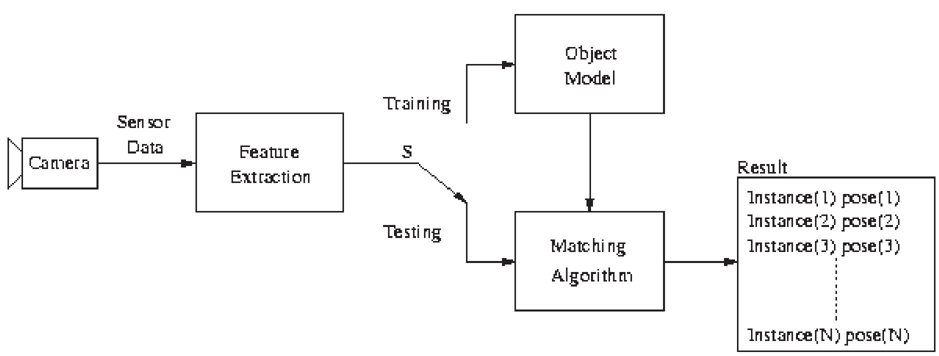


Figure 2. Typical pattern matching structure.

Research in the field of feature-based pattern matching performed at the Institute of Electrical Measurement and Measurement Signal Processing at Graz University of Technology, in collaboration with Sensotech Research and Development, has lead to a new hierarchical approach to 2D pattern matching.[3] The project goals are to build a real-time pattern matcher for 2D scenes that allows for five degrees of freedom (DoF) in object transformations and up to 50% missing features. Initial experiments show this new approach is feasible.

continued on p. 8

A wearable computer for mobile augmented-reality-based robot control

In many situations an intelligent robot can be used to perform tasks that are either hazardous or unpleasant for humans. Such tasks include working in disaster areas or conditions that are, for example, too hot. An intelligent robot can work on its own to some extent, but in some cases the aid of humans is necessary. This means the robot must be controlled from somewhere else, i.e. teleoperated. Several different user interfaces are currently available for teleoperation applications, but many of these do not give the controller a good feel of what is happening in the remote environment. This can cause difficulties in controlling the robot, inefficient performance of tasks, and—if something is not noticed by either the robot's sensors or the controller—can even lead to equipment damage.

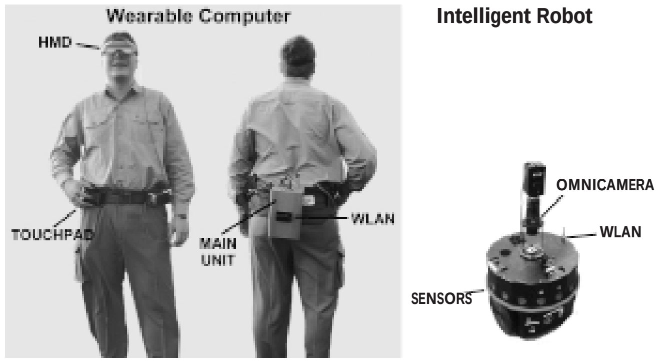


Figure 1. Wearable computer for mobile AR (Left) and an intelligent robot (Right).

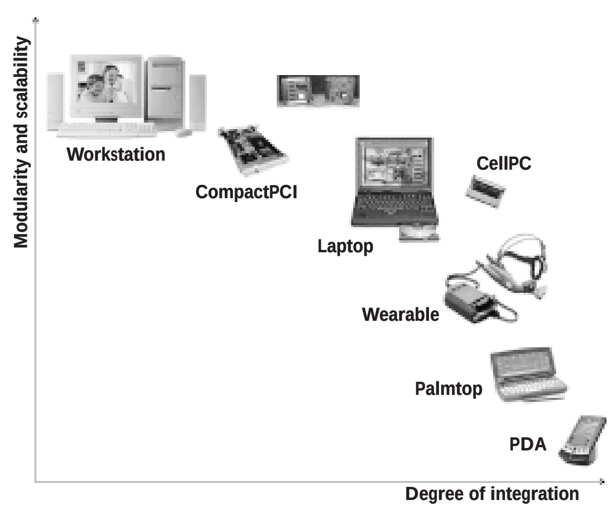


Figure 2. Degree of integration versus modularity and scalability.

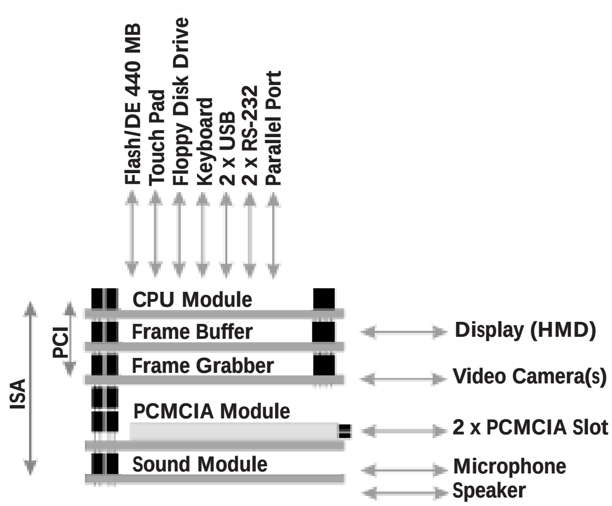


Figure 3. The PC/104+ stack of the Oulu wearable computer.

Augmented reality

Augmented Reality (AR) can provide an interface between the user and the environment that is perceptually much stronger than in other methods. AR is a method that combines virtual objects into user's perception of the real world. These virtual objects are typically textual information—graphical images, force feedback, or sound—provided to the user based on the information gained from the real world and on preprogrammed rules in the computer system. In a well-realized teleoperation application, the goal is to achieve a feeling (immersion) so strong that the controller can truly operate the robot as if they were on site. In the case of cooperation between an intelligent robot and a human, the need for good immersion is not as great as in the case of conventional robots, but mobile AR still makes controlling the robot easier. The technique allows robot-human cooperation without rooting the controller to any particular place, and thus enabling the person in control to perform other tasks while controlling intelligent robots.

The requirements of AR, and especially mobile AR, applications are extensive. A typical mobile AR system must provide all the characteristics required for AR as well as all the properties of mobile devices. This sets high requirements for both the system hardware and software that keep the devices and services going. The basic requirements of a mobile AR terminal are: a Head Mounted Display (HMD) for rendering the computer generated images to the user's view of the world; a wireless communication medium for transmission of information needed in mobile AR applications; several sensor systems for detecting the position and orientation of the user (and, in some applications, also the user's limbs); enough processing power to handle the calculations needed in the applications; and devices for the user to interact with the system without restricting movement. In addition to all these, the terminal must be easy to use, light enough to be carried, and, of course, self-sustained in terms of energy and usable everywhere. Not many systems currently fulfill these requirements, but some devices come close.

Wearable computers

Wearable computers—by definition, devices that the user can carry along like clothing—can be used as terminals in mobile AR applications. Currently available commercial wearable computers do not, however, fulfill the requirements listed above. Therefore we have designed and implemented a wearable computer in the Computer Engineering Laboratory at the University of Oulu as part of the CyPhone project. The wearable computer, shown in Figure 1, was built as a prototype of a future mobile phone. We believe that the current cellular phones will evolve into versatile mobile AR terminals capable of performing several tasks: including teleoperating an intelligent robot.

When designing a wearable computer, we had to choose between several technologies. Our goal was to build a wearable computer that would be as modular and scalable as possible, so as to allow the use of the device in a variety of different mobile AR applications. Many alterna-

continued on p. 7

Object-oriented and model-based approach to objects recognition

An important area of research at the Institute for Measurement Science at the Bundeswehr University, Munich, is the realization of intelligent vision-guided robots that can recognize and manipulate differently shaped objects (see Figure 1), in an arbitrary orientation, without quantitative knowledge about the objects and their subsystems. However, with existing 5-DoF manipulators (see Figure 2) objects are mostly grasped from above. This requires not only an appropriate method for controlling the manipulator to achieve a grasp, but also a suitable method for recognizing the objects. Object recognition here not only includes the actual recognition of the gripper and the object to be grasped—i.e., detecting them from the background and other irrelevant image information—but also the classification of objects and the estimation of their pose to decide which of the robot's behaviors need to be executed to accomplish the grasping. It also has to determine suitable grasping points. The object-oriented and model-based approach introduced here is suitable for this task.

Figure 1. Objects used in recognition experiments.

Object-oriented and model-based object recognition

The main task of object recognition, in the context of vision-guided grasping of objects, is to deliver continuously—and in real time—the image coordinates and orientations of the gripper and the object to be grasped.

According to our concept, the task of object recognition is decomposed into several largely independent modules, each being responsible for a limited subtask. The following are needed to achieve object recognition and to obtain the information necessary for robot control:

- Object detection, i.e., separating objects from the background and other irrelevant image information[1]
- Classification and recognition of differently shaped objects: cylindrical, noncylindrical, convex and concave, etc.[2]
- Determination of the object position in 3D space, its corresponding suitable reference point, and the orientation in the images[3]
- Identification of the gripper
- Determination of the gripper reference point and orientation in the images
- Determination of the priority of grasping multiple objects, i.e., the selection of the object to be grasped first when there is more than one object to be grasped in the images

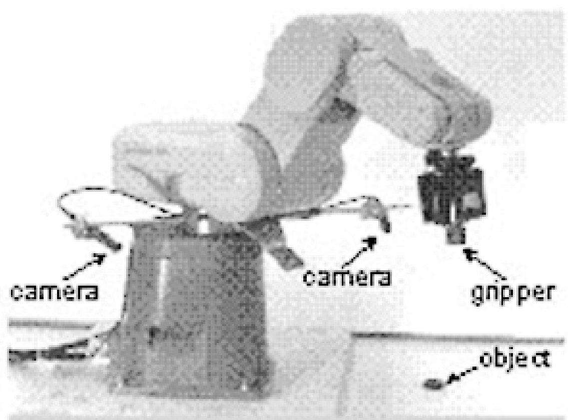


Figure 2. Mitsubishi Movemaster RV-M2 with mounted cameras.

Actually, each of the above subtasks is a separate subroutine or process. These may run parallel to each other or sequentially, so enabling the continuous and simultaneous detection and classification of objects, and the estimation of their pose. Features are extracted in the left and the right image separately. They are, then, evaluated in another process, the stereo vision evaluation process, to recognize objects.

Experimental results and conclusions

This approach has been successfully tested and demonstrated in real-word experiments with a vision-guided uncalibrated manipulator (see Figure 2). In these experiments, objects of various shapes (see Figure 1), in different positions unknown by the robot, were recognized and then grasped. Objects were always recognized reliably if the grey level difference between the object and the background was greater than 10 levels.

The key point of our approach is that features extracted directly from images in real time were used for object recognition. It does not require object reconstruction, and therefore saves time and is suitable for realtime systems.

In sharp contrast to the classical approach, our approach does not require any knowledge regarding:

- The exact locations of the cameras
- The exact viewing directions and the internal parameters of the cameras
- Quantitative knowledge about objects and system
- The surrounding environment: e.g. lighting (within reason) and surrounding landmarks.

Minh-Chinh Nguyen
Institut fuer Infomatik IX
Der Technischen Uni. Muenchen
Orleansstr. 34
81667 Muenchen, Germany
E-mail: Nguyen@in.tum.de

References

1. M.-C. Nguyen and V. Graefe, *Visual recognition of objects for manipulation by calibration-free robots*, **Proc. SPIE 3966,** pp. 290-298, 2000.
2. M.-C. Nguyen and V. Graefe, *Stereo-vision-based object classification*, accepted for presentation at **IROS2000, Takamatsu, Japan, October 30 - November 2000.**
3. M.-C. Nguyen and V. Graefe, *Stereo vision-guided object grasping,* **Proc. Advanced Manufacturing in the Automotive Industry, ISBN 1-902856-023,** pp. 77-85, 1999.

Virtual environments for machine vision system programming

Virtual environments have been used to create workplans and quality inspection programs, interactively, for industrial robots and non-vision-based coordinate measurement machines (CMM). These programs can be executed without modification to the physical work platform, reducing system downtime and operating costs and risks. We propose an approach to extend the use of virtual environments to the interactive creation of vision inspection workplans.

3D computer graphic and realistic physical modeling is used to emulate traditional components of a machine vision system. These include illumination, cameras, lenses, framegrabbers, motion control, and the workpiece under test. We describe a simulation engine that can generate a video stream corresponding to the view of a camera/lens subsystem in the virtual environment. This simulation engine can be combined with an existing video-based measurement products to create a virtual inspection environment. Consequently, operators familiar with the original system may use the virtual system without further training. Alternatively, novice operators may be trained using the virtual system.

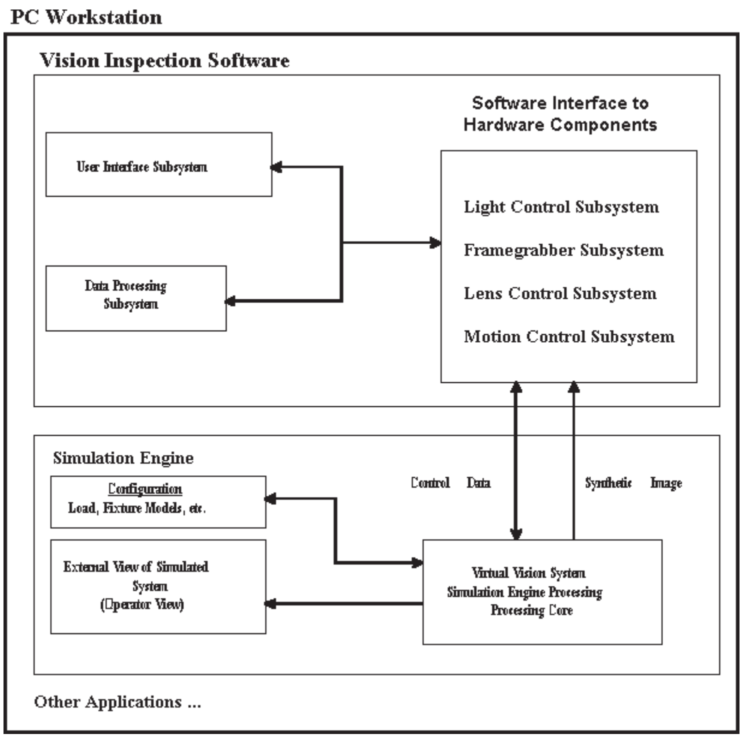


Figure 1. System architecture using VSSE.

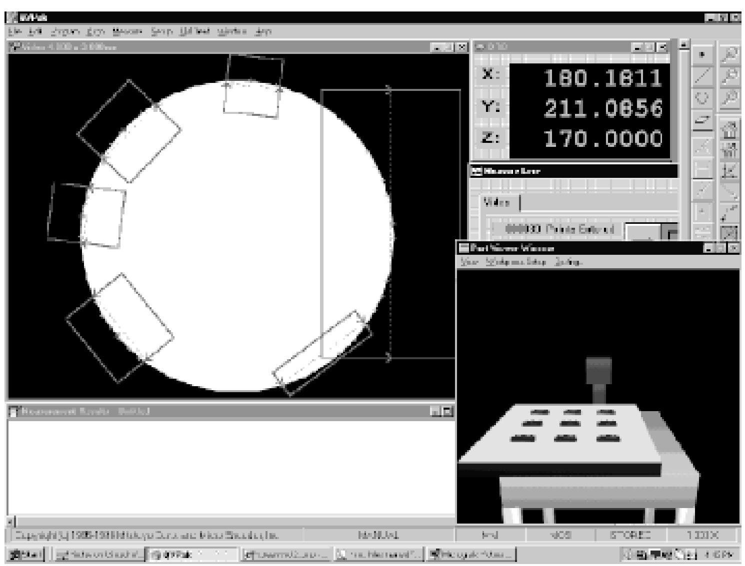


Figure 2. User interface of prototype virtual-machine-vision system.

Background

Several offline systems exist for programming non vision-based CMMs and industrial robots using 3D virtual environments. Vision-based systems have typically been seen as too complex to allow for interactive, offline training of measurement routines.

One of the limiting factors in producing a virtual environment for vision applications is the difficulties in modeling lens effects. Historically, most standard graphics 3D rendering algorithms have produced images with infinite depth of field. Several researchers have addressed the problem of creating realistic imagery for systems with non-idealized lenses.[1-5] One of the first methods of producing depth of field effects was proposed by Potmesil and Chakravarty.[3] Their approach assumes that a 3D-graphics rendering engine generates an image and a corresponding Z-buffer entry for each pixel. Post processing of this data generates a final, defocused image. The quality of the results can be relatively poor but are good enough for many applications. Advances in hardware and algorithms are proceeding towards the goal of realistic camera-model-based rendering in real-time.

Vision System Simulation Engine (VSSE)

The proposed virtual-machine-vision system is based upon a simulation engine (VSSE) that produces synthetic imagery based on a set of environmental factors. This processing engine will be utilized by an existing machine-vision installation via software interfaces that are typically used to access physical system components. An example the architecture of a system using a VSSE is presented in Figure 1.

Graphical and physical modeling techniques can be used to create a complete virtual model of an existing vision inspection system. Three-dimensional CAD models of workpieces may then be imported into this virtual environment. The simulation subsystem synthesizes the information content of a system's lighting, motion control, workpiece model, lens, and camera configuration data to produce still or real-time synthetic images of the camera's view of a workpiece.

Machine-vision-software image acquisition, motion control, and automated lens change requests will be directed automatically to the encapsulated simulation engine. Imaging algorithms may then be executed upon the resultant synthetic data. Minimal changes should be required in the core machine vision application.

continued on p. 9

Controller-driven VRML animation of a real-time inspection system

Virtual objects in a web-based environment can be interfaced to, and controlled by, external real world controllers. A Virtual Reality Modeling Language (VRML) inspection cell was created that models a real-time inspection system. The latter consists of a Cordax coordinate measuring machine (CMM), an inspection probe, a vision system for determining the part position and orientation, and an open architecture controller.[1] The controller performs real-time processing of sensor data for feedback control of the inspection probe and permits access to data, such as the probe position and the part position and orientation, to drive the VRML model of the system.

The real world controller drives the VRML CMM via a socket connection that connects it to the collaborator's web browser. When a client visits the remote access web page (Figure 1), a VRML model of the Cordax CMM is downloaded to their machine. From then on, only the current joint positions need to be sent across the socket from the controller (server) to the web browser (client) to move the joints of the VRML CMM model to the current controller position. All of the graphics are handled on the client machine. The data sent across the socket consists of 3 floating point numbers (x, y, and z position of the measurement tool center point): this information (12 bytes) is sent across the socket every 30 ms, giving a data rate of 400 bytes/sec. This low bandwidth is usually handled easily, even with a slow network connection.

The current probe position, which is stored in a world model buffer in the controller, is collected by a Java applet running on the web page. For communication between a VRML world and its external environment an interface between the two is needed. This is called the External Authoring Interface (EAI)[2] and it defines the set of functions on the VRML browser that the external environment can perform to affect the VRML world. In essence, the EAI provides a method for developing custom applications that interact with, and dynamically update, a 3D scene. The interface is designed to allow an external program (referred to here as an applet) to access nodes in a VRML scene using the existing VRML event model.

Before a part can be inspected, the transformation between the part and the machine coordinate systems must be calculated during a setup process. The vision system automates setup using monocular vision for parts with 2D features. The automated setup algorithm consists of two parts: an image processing algorithm, called Lola,[3] and a pose estimation algorithm. The former produces line segment and constant curvature arc features, while the latter matches sensed with model features and performs pose clustering and verification (Figure 2).

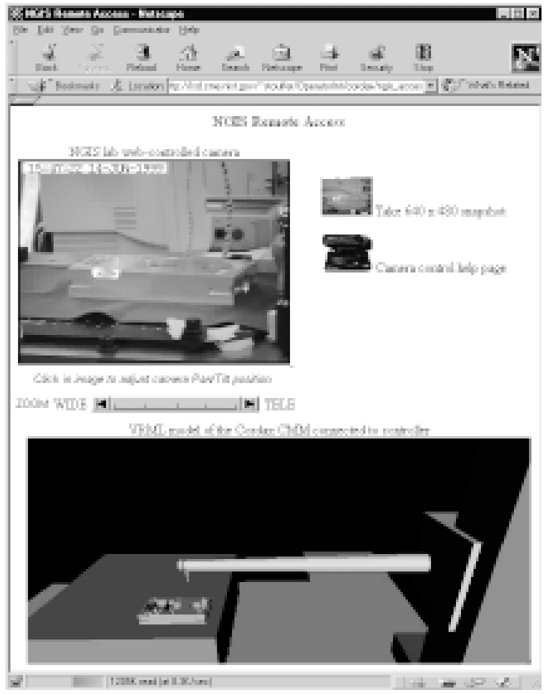


Figure 1. Remote access page with the client-controllable pan/tilt/zoom camera and controller-driven VRML model of the inspection cell.

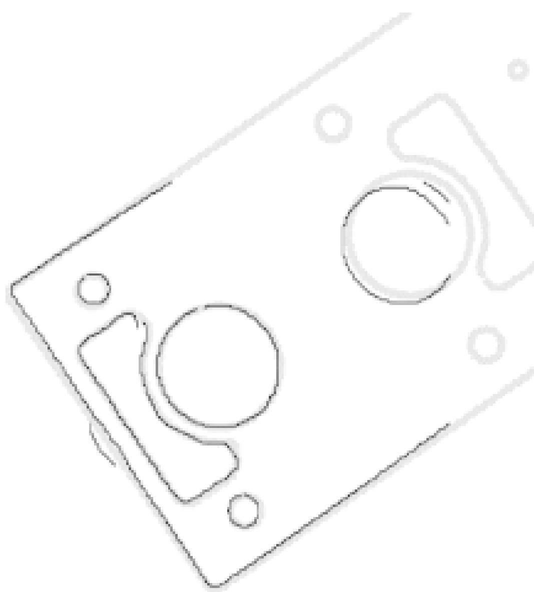


Figure 2. The black features are those sensed, while the gray are model features that have been translated and rotated by the computed pose estimate. Note that a correct match is made in spite of warped, occluded, and spuriously- sensed features.

The matching segment relates the problem of finding a correspondence between sensed and model features to that of finding a word in a dictionary. A dictionary of words is formed where the letters consist of translation and rotation invariant geometric attributes of all possible sets of k model features. Each word in the dictionary is sorted by canonical order of the letters within the word, and the entire dictionary is sorted according to the canonical order of the words. At run-time, $k < s$ sensed features are randomly selected from s sensed features. The sensed word is formed and all matches between sensed and model words are found. After the part position and orientation is obtained from the vision system, it is then updated in the VRML model to represent the part's real world position and orientation.

The remote access web site also contains a client-controlled pan/tilt/zoom camera that sends video to the client. This allows a client to monitor a remote inspection with a PC and an internet connection. The video within the remote access page is a server-pushed JPEG image with an image-map overlaid on it. The resolution of each image is 320×240 pixels. Each image is fairly large (20 Kbytes), so frame rates are slow: just 1 frame per second. The overlaid image-map allows the collaborator to click on the image to determine where the new center of interest should be. The required pan/tilt angles are calculated and the camera moves to recenter the image.

Disclaimer: Certain commercial equipment, instruments, or materials are identified in this paper, Such identification does not imply recommendation or endorsement by NIST, nor does it imply that the materials or equipment identified are necessarily best for the purpose.

continued on p. 10

Adaptive hardware

continued from cover

cog-wheels, and a thin tube containing a sensor (a photodiode) at the inner end. By means of the cog-wheels, the motor can position the facet within a range of about 200˚, and the potentiometer provides feedback about the actual angle. The thin opaque tube reduces the sensor's aperture to about two degrees. These tubes are the primitive equivalent of the biological ommatidia. The robot's compound eye is actually one-dimensional: tubes are stacked vertically only, to allow for free rotation (we are thus using vertical lines as "obstacles").

The robot was tested on a task of using motion parallax to estimate a critical lateral distance to obstacles. The phenomenon of motion parallax has the advantage that it is very well understood and favors non-trivial sensor morphology: when an observer's path passes by the side of a stationary obstacle, he can estimate the distance between his path and the obstacle from the apparent motion of the obstacle through his visual field if he knows his own speed. However, the obstacle's apparent (angular) velocity depends on the viewing angle: when it is seen towards the front it seems to move more slowly than when it passes by the side of the observer. This nonlinear effect could be calculated and the visual data could be corrected accordingly. However, this requires unnecessary computation: an alternative way is to use an eye morphology with a higher density of visual receptors towards the front. There even exists a special sensor density distribution that exactly compensates the above effect so that the apparent speed of the obstacle over the observer's retina becomes independent of the viewing angle. This greatly simplifies distance estimation and further processing of the visual data. It is believed that motion parallax, in combination with an adequate morphology, is used by insects such as the house fly to avoid obstacles.

In our experiments we used a control architecture consisting of a homogeneous array of identical elementary motion detectors inspired by the brain of the house fly. While the control architecture was kept fixed throughout the experiment, the robot's sensor morphology (the angular distribution of facets) could be modified by an evolutionary algorithm. The fitness function was chosen such that the robot had to avoid obstacles too close to its path and not avoid the ones further away than a critical distance. Since evolution requires a considerable amount of time on the real robot, we performed a set of simulations[2] to optimize parameters such as the specific type of evolutionary algorithm used, the phenotype encoding scheme employed, as well as the settings for the initial parameters. It turned out that adopting specific parameters (such as the exact noise level on the fitness function) from the physical robot was essential. Finally, using a (1,6) evolution strategy directly on the physical robot we were able to evolve non-homogeneous facet distributions that could perform the required task and qualitatively resemble the ones found in the house fly (see Figure 2). An evolutionary run on the physical robot took on the order of five hours to converge (for 9 facets).

We are currently investigating more complex tasks where the "optimal" morphology is less evident. Also a 2D version of the artificial compound eye, with more variable parameters (like sensitivity, receptive fields, etc.), is in preparation.

This project was supported by SNF Grant No. 2000-053915.98.

Lukas Lichtensteiger
Artificial Intelligence Lab
Department of Computer Science
University of Zurich
Winterthurerstrasse 190
CH-8057 Zurich, Switzerland
E-mail: llicht@ifi.unizh.ch

References

1. L. Lichtensteiger and P. Eggenberger, *Evolving the morphology of a compound eye on a robot*, **Proc. Third European Workshop on Advanced Mobile Robots (Eurobot '99),** 6-8 September 1999, Zurich, Switzerland, pp. 127-134.
2. L. Lichtensteiger and R. Salomon, *The evolution of an artificial compound eye by using adaptive hardware,* **Proc. 2000 Congress on Evolutionary Computation,** 16-19 July 2000, La Jolla Marriott, San Diego, CA, USA, pp. 1144-1151.

A wearable computer

continued from p. 3

tives are available, and their common denominator is that modularity and scalability typically diminish as the degree of integration increases (see Figure 2). We selected the PC/104+ architecture as the base of our design as it is small, compatible with personal computers, low energy, and has great modularity and scalability and a wide selection of commercial modules, so allowing the whole system to be built using off-the-shelf components. The PC/104+ stack of our wearable computer, including most of the system electronics, is shown in Figure 3.

Our wearable computer has been used in several different applications at the University of Oulu. It has been proved to be quite suitable for testing and demonstrating mobile AR applications in practice. It both complies with the requirements listed earlier and provides the abilities required of future terminals using today's technology. Naturally, some compromises had to be made in order to build a device of the future now. Our compromises were the size and weight of the device. Even though the sturdy belt makes it quite bearable, and it is ideal for research purposes, the device is definitely not suitable for consumer use.

Tuukka Turunen and **Juha Röning**
Infotech Oulu and Department of Electrical Engineering
University of Oulu
PO Box 4500
FIN-90014, Finland
E-mail:{tuukka.turunen, juha.roning}@oulu.fi
http://www.ee.oulu.fi/research/isg
http://www.ee.oulu.fi/projects/CyPhone

References

1. T. Turunen, J. Röning, S. Ahola, and T. Pyssysalo, *A wearable computer for mobile augmented reality based controlling of an intelligent robot*, **Proc. SPIE 4197,** conf. to take place in Boston, MA, 5–8 November 2000.
2. T. Pyssysalo, T. Repo, T. Turunen, T. Lankila, and J. Röning, *CyPhone – bringing augmented reality to next generation mobile phones*, **Proc. DARE2000,** conf. in Elsinore, Denmark, 12 – 14 April 2000.

Hierarchical approach to pattern matching

continued from p. 2

A hierarchical structure

Traditional pattern matching systems are based on a structure as shown in Figure 2. Given the sensor input, a set of features is calculated using algorithms of a given accuracy. This set of features forms the basis for further processing, which is either the model building process (training phase) or the matching process (test phase). There is no interaction between the matcher algorithm and the feature extractor involved. Possible drawbacks in computation time result from the fact that features are processed at a fixed accuracy for the whole region of interest. In real world scenes, however, features do not only originate from object instances but also from spurious data. Processing of spurious features at subpixel accuracy does not contribute to the quality of the matcher result but extends matching time.

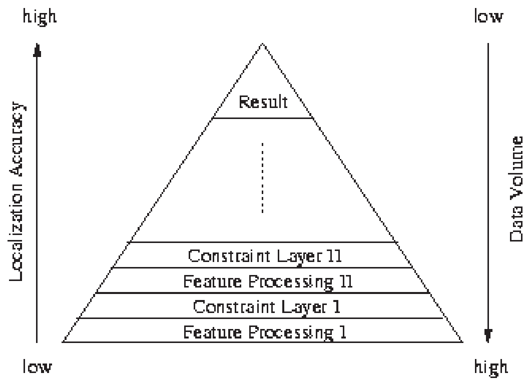


Figure 3. A hierarchical structure of intermediate feature processing layers and constraint layers.

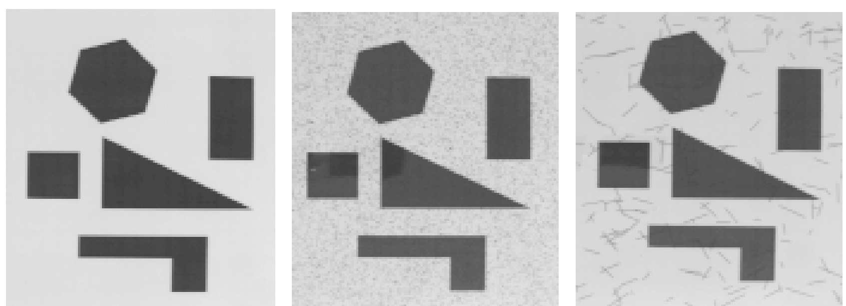


Figure 4. Different test images used to evaluate the hierarchical structure. (a) undistorted image, (b) additive salt and pepper noise, (c) superimposed random lines.

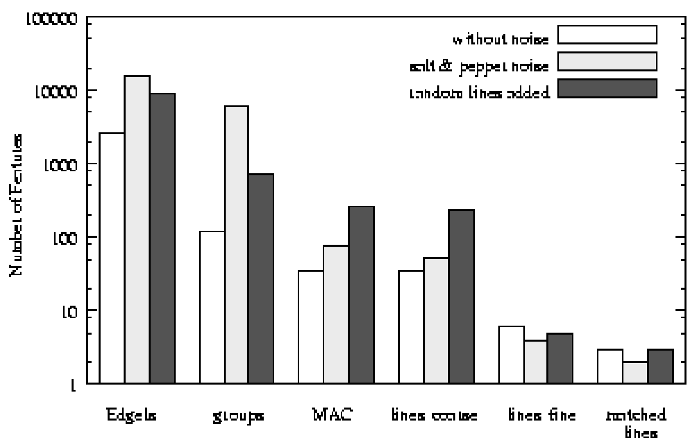


Figure 5. Reduction of features after passing the constraint layers of the structure.

We have introduced a new hierarchical approach to pattern matching as depicted in Figure 3. The matching system is split up into a number of distinct feature processing and constraint layers, where the latter prevent features that do not meet certain requirements from further processing. Subsequent feature processing can, therefore, focus on the important parts of the image. Through the use of feature processing algorithms that are scalable in terms of computational costs and accuracy, we can apply low-accuracy but fast feature extractors at the bottom of the structure, and more time-consuming but highly accurate feature extractors in subsequent layers. Another advantage of this hierarchical structure is the possibility to use *a priori* knowledge gained from previous feature extraction and constraint layers to enhance the extraction of further features. It is, therefore, possible to optimize feature extraction based on, for example, a scale or rotation estimate obtained by previous layers.

Experiments using both artificial and real world images have been carried out for a limited set of specifications. Simple polygons were matched using an eight-layer structure, with the matching algorithm based on object edges. The hierarchy consists of the following layers:

- Edge pixel grouping
- Minimum Area Constraint (MAC)
- 'Coarse' line processing
- Minimum length constraint
- Collinear lines
- Interpretation tree
- 'Fine' line processing
- Hypothesis verification

Our test objects were allowed to undergo Euclidean transformations with no restriction in object rotation. To investigate the behavior of the algorithm under various kinds of spurious data, two different noise sources (salt and pepper noise, random lines) were superimposed to the images as shown in Figure 4. Figure 5 shows the decrease in features to be processed as the matcher algorithm passes each layer. The results correspond to the original image without spurious data (white bars), a search scene with superimposed salt and pepper noise (light gray bars), and a search scene with added random lines (dark gray bars).

The experimental results indicate that by using intermediate constraint layers in a matching algorithm, it is possible to focus computationally expensive feature-extraction algorithms on subimages that are needed to obtain the matching result reliably.

Future work

The next step of our project will be to extend the matcher specifications to allow for 'quasi-affine' object transformations. Both constraint and feature extraction layers will be extended to include higher level features and corresponding constraints under this new group of transforms. In addition to finding new sets of features, the incorporation of *a priori* knowledge into a particular feature extractor is a major research issue.

Markus Brandner and **Axel Pinz**
Institute of Electrical Measurement and Measurement Signal Processing
Graz University of Technology
Schiessstattgasse 14B
A-8010 Graz, Austria
E-mail: {brandner, pinz}@emt.tu-graz.ac.at

Wolfgang Pölzleitner
Sensotech Research and Development
Scheigergasse 74
A-8010 Graz, Austria
E-mail: wp@sensotech.at

References

1. C.A. Rothwell, A. Zissermann, D.A. Forsyth, and J.L. Mundy, *Planar object recognition using projective shape representation,* **Int'l J. of Computer Vision 16** (1), pp. 57-99, September 1995.
2. T. Lourens and R.R. Würtz, *Object recognition by matching symbolic edge graphs,* **Lecture Notes in Computer Science 1352,** pp. II-193 - II-200, 1998.
3. M. Brandner, A. Pinz, and W. Pölzleitner, *A concept for real-time, robust pattern matching,* **24th Workshop of the Austrian Association for Pattern Recognition,** pp.145-152, May 2000.

Virtual environments

continued from p. 5

Implementation and results

A virtual vision inspection system based upon a commercially-available vision system and a custom simulation engine was developed and tested. The Mitutoyo QuickVision line of video CMMs was used as the target system for simulation. It is comprised of a CNC stage viewed by a camera on an orthogonal CNC axis. Vision inspection programs may be generated by example and stored as a part program.

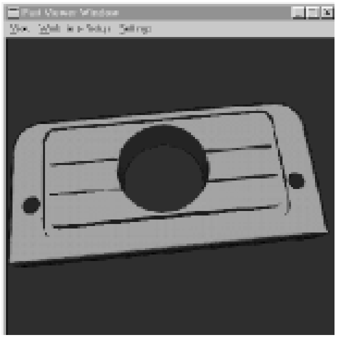


Figure 3. Workpiece under test.

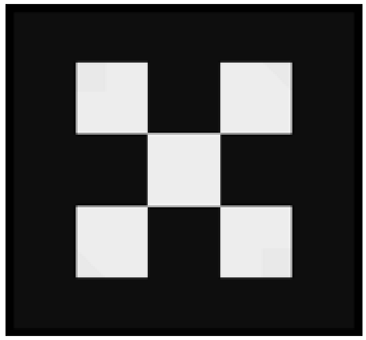


Figure 4. Target in focus.

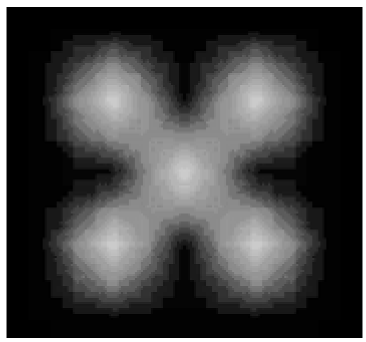


Figure 5. Target with significant defocus.

In particular, the system emulated a 400×400mm stage, fixed lens system, and three independent lighting sources (stage, coaxial, and ring illumination). Workpieces were imported from Pro-Engineer CAD model data. All code was executed on a 450Mhz Pentium II processor running Windows 98 and DirectX 6. A simplified version of Potmesil and Chakravarty's algorithm was used.[3] The user interface of the system is presented in Figure 2. A view of the part under test is presented in Figure 3. Examples of an in-focus and out-of-focus target are illustrated in Figures 4 and 5, respectively.

After implementation and integration of the VSSE, the resultant virtual vision system was evaluated relative to its ability to produce functional vision inspection programs, which were generated on the virtual system and executed successfully after minor modifications to lighting values. Skilled vision-system operators responded positively to the implementation's usability and value in generating inspection programs absent in the physical hardware. In addition such an implementation is expected to have application for training, multimedia documentation, and sales demonstrations.

Conclusion

The VSSE is an effective method for building virtual-machine-vision inspection programs that emulate a specific hardware platform. However, there are many future challenges to overcome before VSSE methods can realize their full potential. Three key areas of future investigation include: increasing realistic real-time lens-effect simulation algorithms; improved consistency between simulated and actual lighting systems; and inclusion of accurate workpiece reflection data. We are currently conducting research in each area of interest.

Richard M. Wasserman
Automated Measurement Systems Group
Micro Encoder Inc.
11533 NE 118th Street
Kirkland, WA 98034
E-mail: richw@microen.com
Phone: 425/821-3906
Fax: 425/821-3228

References

1. C. Kolb, D. Mitchell, and P. Hanrahan, *A realistic camera model for computer graphics*, **Computer Graphics** (Proc. SIGGRAPH '95), ACM SIGGRAPH, pp. 317-324 , 1995.
2. M. Shinya, *Post-filtering for depth of field simulation with ray distribution buffer*, **Graphics Interface '94**, pp. 59-66, 1994.
3. M. Potmesil and I. Chakravarty, *A lens and aperture camera model for synthetic image generation*, **Computer Graphics 15** (3), pp. 297-305, August 1981.
4. M. Woo, J. Neider, and T. David, **Open-GL Programming Guide - 2nd Edition**, Addison-Wesley, Massachusetts, pp. 402-406, 1997.
5. P. Rokita, *Generating depth-of-field effects in virtual reality applications*, **IEEE Computer Graphics and Applications 16** (2), pp. 18-21, March 1996.

— Calendar —

2000

Optical Science and Technology

SPIE's Annual Meeting

30 July - 4 August
San Diego Convention Center
San Diego, California, USA
Technical Exhibit: 1-3 Aug.

Photonics East

3-8 November
Boston, Massachusetts USA
Including international symposia on
- **ISAM '00—Intelligent Systems and Advanced Manufacturing**
- **Information Technologies—Voice, Video, and Data Communications**
- **EIS '00—Environmental and Industrial Sensing**

Law Enforcement
Technical Exhibit: 6-8 Nov.

2001

Photonics West

20-26 January
San Jose Convention Center
San Jose, California USA
Including international symposia on:
- **LASE '01—High-Power Lasers and Applications**
- **Optoelectronics '01—Integrated Devices and Applications**
- **BiOS '01—International Biomedical Optics Symposium**
- **SPIE/IS&T's EI '01—Electronic Imaging: Science and Technology.**

For information on SPIE meetings contact
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

CPU-less robotics

continued from p. 12

Adaptive CPG for Legged Locomotion

The basic notion of an autonomous neural circuit generating the sustained oscillations needed for locomotion was first articulated in the early part of this century. The key idea is that an autonomous system of neurons can generate a rhythmic pattern of neuronal discharge that can drive muscles in a fashion similar to that seen during normal locomotion. These neurons make up a Central Pattern Generator (CPG). Under normal conditions, the CPG will make extensive use of sensory feedback from the muscles and skin, as well as descending input.

It is well known that sensory input can modulate the activity of the CPG. Modulation of the CPG by this sensory input can be seen quite clearly in the resetting of the CPG phase. For example, as a walking cat pushes its leg back, sensors in the leg muscles detect stretching. These sensors (called stretch receptors) signal this stretch to the nervous system. Their firing initiates the next phase of the CPG causing the leg to transition from stance to swing phase.

CPGs are most often modeled as distributed systems of non-linear oscillators. In our implementation, the basic coordination in the leg is achieved by phasically coupling two neurons together to achieve oscillations. When coupled together, they are alternately active. This alternating activity is the basic coordination needed to drive the hip of a robot. A phase control circuit governs the phase difference between the neurons.

These oscillator neurons drive two integrate-and-fire spiking moto-neurons, which are used to drive an actuator. The spiking neuron could also drive biological muscle or a pneumatic cylinder, a McKibben actuator, or biomuscle directly.

In our experimental setup, the robot under control uses servomotors. To be compatible with this technology, it was necessary to low-pass filter the spiking neurons and then integrate the resulting smooth graded velocity signal. Figure 1 shows the control loop, the CPG chip and the pair of legs under control.

The CPG Chip

The CPG chip is designed to provide biologically plausible circuits for controlling motor systems. The chip contains electronic analogs of biological neurons, synapses and time-constants. In addition, the chip also contains dynamic analog memories and phase modulators. Using these components, non-linear oscillators—based on the central pattern generators of biological organisms—can be constructed.

The dynamic properties of the neural circuits can also be adapted using direct sensory information. In this first version of the chip, shown in Figure 1, all the components are individually accessible such that they can be connected with off-chip wiring to realize any desired circuit. In future versions, tested neural CPG circuits will be integrated with completely hardwired or programmable circuits.

Running Experiments

Running with a passive: In this experimental setup, the CPG circuit drives the actuator in the hip joint. The knee joint is passive and rotates with very little friction. (This is similar to an amputee with a thigh level prosthetic leg.) The assembly is suspended above a rotating drum. The CPG circuit is started.

A remarkable feature of this system is that the knee joint adapts the correct dynamics to enable running (!). As the upper limb swings forward, the lower limb rotates so that the foot comes off the ground. When the upper limb is suddenly accelerated backward, the momentum in the lower limb forces the knee to lock in place. At just the correct moment, the foot contacts the ground and the subsequent loading keeps the knee joint locked in place. As the foot travels backward it eventually begins to unload. Stored energy in the elastic foot causes it to 'kick up' and smartly snap off the ground, an effect most noticeable at higher velocities.

Sensory feedback lesioning: This experimental setup is similar to the first experiment. The difference is that sensor feedback is lesioned (turned off) periodically.

After lesioning, the leg drifts backward significantly due to a bias built into the chip. When the sensory input is restored, the leg returns to a stable gait. Perturbations to the leg cause momentary disturbances. Several of the trajectories are clear "outliers" to the typical orbit, and result from environmental disturbances.

Summary and conclusions

We have demonstrated the first experimental results of an adaptive a VLSI neural chip controlling robotic legs. Using sensory feedback, the circuit can adapt the gait of the leg to compensate both for chip bias and for environmental perturbations. This work represents the first experimental results of an adaptive VLSI neural chip controlling robot legs.

Basic rhythmic movements in animals are generated by a network of neurons in the spinal cord called the Central Pattern Generator or CPG. CPGs have been studied extensively and are beginning to be better understood. Our work could lead to implantable biocompatible sensorimotor controllers. The majority of work on implantable microelectronics have focused on sensory prosthesis such as retinas, and cochleae, although cardiac pace-makers are also available. Our approach may one day help paraplegics walk or provide amputees full control of artificial limbs. In addition, there is a wide range of applications in personal robotics and the toy industry.

Ralph Etienne-Cummings
Dept of Electrical and Computer Engineering
Johns Hopkins University
Baltimore, MD 21218
E-mail: etienne@ece.jhu.edu

M. Anthony Lewis
Iguana Robotics, Inc.
P.O. Box 628
Mahomet, IL 61853
E-mail: tlewis@iguana-robotics.com

Controller-driven VRML animation

continued from p. 6

Keith Stouffer and **John Horst**
National Institute of Standards
and Technology
100 Bureau Drive, Stop 823
Gaithersburg, MD 20899-8230
Phone: 301/975-{3877, 3430}
{keith.stouffer, john.horst}@nist.gov

References

1. John Horst, *Architecture, design methodology, and component-based tools for a real-time inspection system,* **3rd IEEE Int'l Symp. on Object-oriented Real-time distributed Computing (ISORC 2000),** Newport Beach, CA, March 15-17, 2000.
2. External Authoring Interface Working Group: http://www.vrml.org/WorkingGroups/vrml-eai/
3. S. Sarkar and K.L. Boyer, *Perceptual organization in computer vision: a review and a proposal for a classificatory structure***, IEEE Trans. on Systems, Man, and Cybernetics 23** (2), pp. 382-399, March 1993.

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: info-robo@spie.org
Anonymous FTP: 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

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.

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. Phone: (1) 360/676-3290. Fax: (1) 360/647-1445.

European Office: Contact SPIE International Headquarters.

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

In Russia/FSU: 12, Mokhovaja str., 121019, Moscow, Russia. Phone/Fax: (095) 202-1079. E-mail: edmund@spierus.msk.su.

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.

CPU-less robotics: distributed control of biomorphs

Challenges for robotics in the future include the miniaturization of machines that can walk, run, and fly—increasing the real-time adaptability of robots to the environment—and the creation of mass-market consumer devices (like the Sony Dog). These new technologies will require small, low-cost, power-efficient, and adaptive controllers, and may therefore benefit greatly from computational support that is radically different from current microprocessor-based technology. Biologically-inspired robots (or biomorphic robots) exploit the computational, functional and architectural qualities of living organism to realize smart agents of the future.

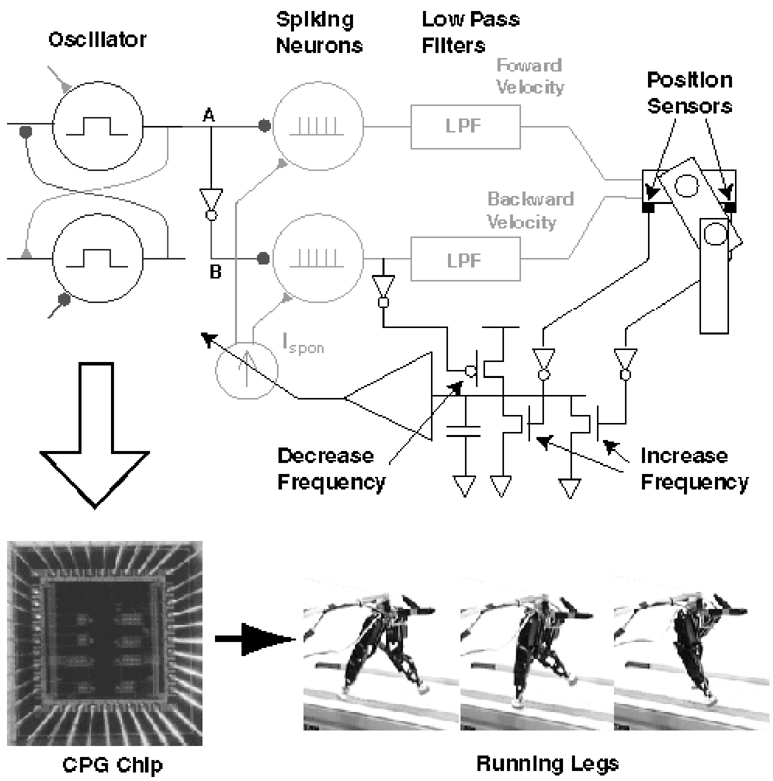


Figure 1. The control loop, the CPG (central pattern generator) chip and the pair of legs under control.

The current state-of-the-art in biologically-inspired robotics, represented by the Honda's humanoid robot or Sony's Dog, depends heavily on microprocessors for all computation. Consequently, the availability of embedded high-performance CPUs has only recently allowed the development of these untethered and "autonomous" robots. Despite their supposed biomimetic nature, the computational mechanisms employed in either of these robots are, to our knowledge, decisively non-biological. The CPUs do not model neural circuits nor do they employ biological control strategies or architectures, other than external appearance.

Does bio-mimicry require that every single detail of the biological master be incorporated in the engineered system? The answer is resoundingly "no." Information processing in the brain can be modeled as many levels. Traditionally, neuromorphic engineering deals with reverse engineering a small subset of the details. One example is the silicon neuron of Mahowald and Douglas from the ETH in Zurich.

We choose to model these systems at a functional level which is more amenable to an engineering solution. In our paradigm, the 'brain' of our robots is modeled as a number of interconnected sub-processors. Each sub-processor roughly reflects the functionality embedded in a brain region or aggregate of brain regions. Furthermore, we require that each subsystem is realizable with compact, low-power Very Large Scale Integrated (VLSI) circuits. The latter requirement reflects the reality of developing autonomous robots that can be small, lightweight, fast and power efficient. This is in stark contrast to the Honda humanoid robot.

Our main thesis is that distributed processing with autonomous computational sub-processors, those that are adaptive and responsive to local sensor information, is necessary in realtime robots that interact with their environments. Local and distributed learning capabilities are a fundamental requirement of such systems. They allow the robot to adapt to variations in the electronics, mechanics and surroundings. Secondly, we believe that these sub-processors are naturally realized in custom VLSI circuits. The resulting advantage will be a combination of highly robust, intelligent behavior, implemented in compact, low power circuits.

Iguana Robotics, Inc., and Johns Hopkins University are developing next generation visually-guided walking machines. These walkers will be more robust, smarter, faster and have more elegant movement than the current state-of-theart. They will handle rugged environments, learn to jump over 3D obstacles, and smoothly change gaits from walking to running to jumping based on passive computational vision. The environment will be sensed using custom-designed compact vision systems, realized primarily in VLSI. Hardware models of neural sub-circuits in motor nuclei, spinal cord, lower brain, cerebellum and visual cortex of vertebrates will be used to control the robot. Because legged locomotion is one of the hardest competences to realize in biomorphic robots, it has been tackled first. We have developed one of the first adaptive Central Pattern Generation (CPG) chips and applied it to control a pair of running legs.

continued on p. 10

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