Advances in Robotics: The DLR Experience

Gerd Hirzinger
Max Fischer
Bernhard Brunner
Ralf Koeppe
Martin Otter
Markus Grebenstein
Ingo Schäfer

Deutsches Zentrum für Luft und Raumfahrt e.V. (DLR),
Oberpfaffenhofen, D-82234 Wessling, Germany
Gerd.Hirzinger@dlr.de

Abstract

Key items in the development of a new smart robot generation are explained in light of DLR's recent activities in robotics research. These items are the design of articulated hands, ultra-lightweight links, and joint drive systems with integrated joint torque control, sensory feedback including real-time 3-D vision, learning and skill-transfer, modeling the environment using sensorfusion, and new sensor-based off-line programming techniques based on teaching by showing in a virtual environment.

1. Introduction

In the past, there has been a kind of very general disappointment about the fairly slow progress in robotics compared to human performance—despite many years of robotics research involving a large number of scientists and engineers. Industrial robots today in nearly all applications are still purely position-controlled devices, perhaps with some static sensing, but still far away from the human arm's performance with its amazingly low own-weight-against-load ratio and its online sensory feedback capabilities involving mainly vision and tactile information, actuated by force-torque-controlled muscles. However, there are a.number of observations in the area of industrial robots that prove that there have been a lot of advances with respect to certain aspects. In particular, industrial robots are much cheaper now (approximately by a factor of 4!) than 10 years ago, while the peripheral costs (e.g., precise part feeding) have remained nearly the same, thus indicating that this development highlights the features of a dead end, calling for more sensor-based adaptivity. The European manufacturers ABB/Sweden and KUKA/Germany have excellent positions among the leading robot manufacturers that sell 5000-9000 robots per year. But there is also real progress in robot technology other than pricing. And service robotics (space robotics as a special case) will help in accelerating the advances in robotics.

The International Journal of Robotics Research
Vol. 18, No. 11, November 1999, pp. 1064-1087,

2. Advances in Industrial Robotics

For many years, robot manufacturers have not integrated available research results (and requests) into their robot controllers—now they do. The most obvious improvements—for example, in the case of the KUKA robots—we were able to essentially contribute the following:

- dynamics and control optimization, in particular:
- automatic optimal controller tuning (Fig. 1)
- inverse dynamics (including friction models) for time-optimal paths (Fig. 2)
- online load estimation (based, e.g., on Swevers et al. 1997)
- online collision detection
- active vibration damping
- next step: fully adaptive and learning control, teletuning (Lange and Hirzinger 1996)
- fast, open controllers will finally allow online sensory feedback (Fig. 3)
- intuitive programming: 6-DoF manual control devices, like DLR's space mouse, are now becoming standard, at least in European robotics (see below) (Hirzinger 1982).

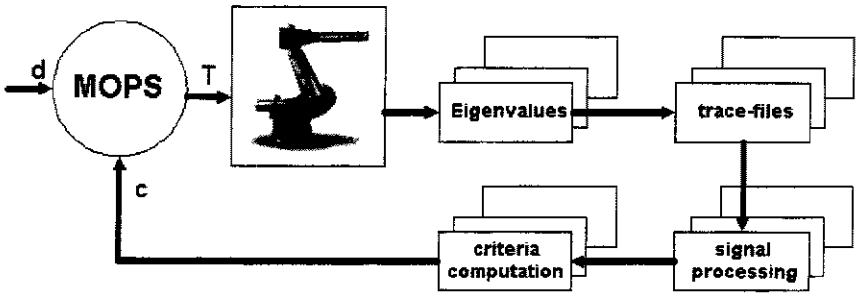
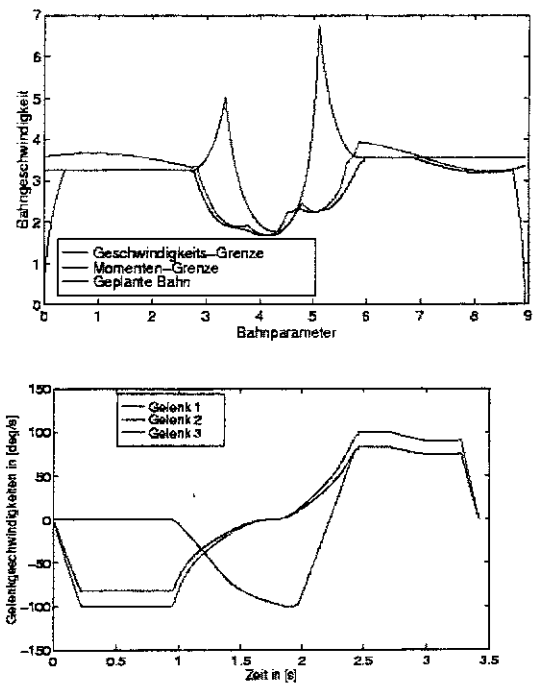


Fig. 1. The KUKA robots are tuning their controllers automatically "overnight" now by optimizing more than 20 performance criteria concurrently.



Optimal Path Planning

Mathematical **time optimal** path based on a **non-linear** robot **model**.

All **constraints** are satisfied (maximum motor torque, gear load).

Guaranteed **overload protection.**

Path planning in **real time**.

Robots now operate **15%-35% faster**.

Teaching the robot is **5 times faster**.

Fig. 2. Time-optimal paths in the KUKA robots are planned using inverse dynamics models.

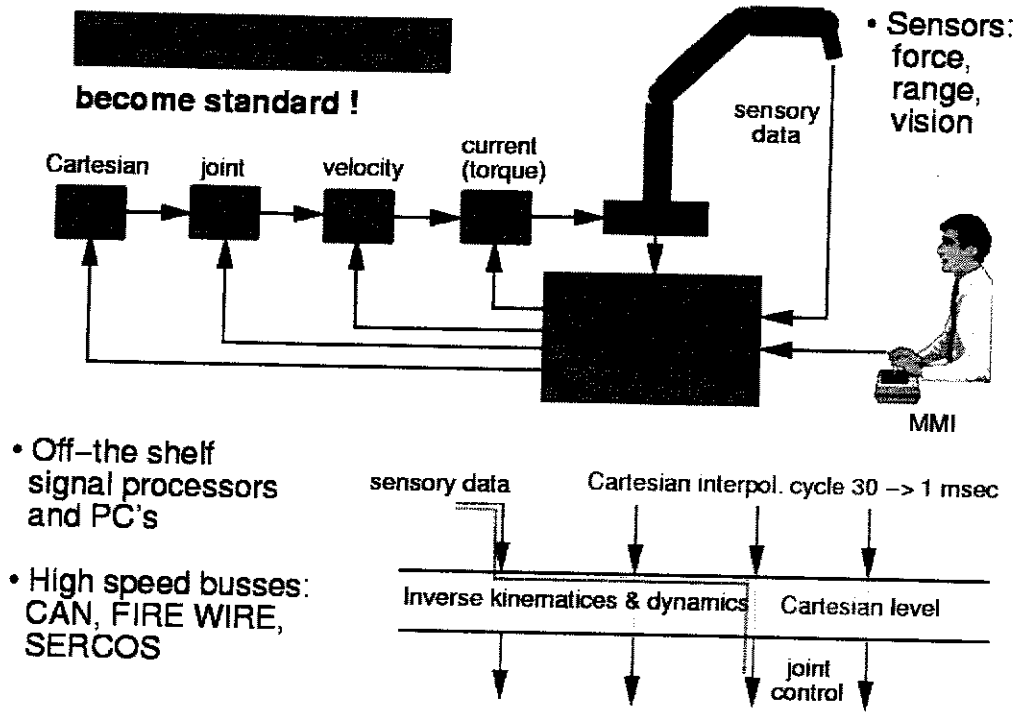


Fig. 3. Open robot controllers are a prerequisite for sensory feedback.

And we now see the Cartesian interpolation cycles that determine the sampling rate of sensory feedback, too, permanently falling from 30 msec a couple of years ago down to a few milliseconds.

3. Advances in Space Robotics

3.1. ROTEX as a First Step

Space robotics (and service robotics in general) might become a major driver for a new robot generation. The experiment we made with ROTEX, the first remotely controlled robot in space, has strongly underlined this. As has been outlined in different papers (Herzinger 1993), ROTEX flew with spacelab mission D2 in April 1993, performed several prototype tasks (assembly, catching a floating object, etc.) in a variety of operational "telerobotic" modes, e.g., online teleoperation on board and from ground (via operators and pure machine intelligence) as well as off-line programming on ground. Key technologies for the big success of this experiment have been

- the multisensory gripper technology, which worked perfectly during the mission (redundant force-torque sensing, 9 laser range finders, tactile sensing, stereo TV).
- local (shared autonomy) sensory feedback, refining gross commands autonomously.
- powerful delay-compensating 3-D-stereo-graphic simulation (predictive simulation), which included the robot's sensory behavior and allowed to master the overall signal delays of 5-7 seconds in case of teleoperation from ground.

3.2. The ETS VII Experience

The next milestone in terms of remotely controlled space robots has been set by Japan's space agency NASDA. Their ETS VII project represents the first free-flying robot satellit (more precisely a larger and a smaller robot on a carrier satellite and a target satellite for rendezvous and docking). The system, launched at the end of 1997 and still operable mid-1999, has allowed a number of NASDA partners to perform their own telerobotic experiments and so far has been very successful (Oda 1999).

Between April 19 and 21, we had the opportunity to perform our own experiments (GETEX) with NASDA's ETS VII free-flying space robot. Our goals were twofold:

- To verify the performance of advanced telerobotic concepts, in particular concerning the implicit task-level programming capabilities as well as the sensor-based autonomy and world-model update features.
- To verify 6-DoF dynamic models for the interaction between a robot and its free-floating carrier satellite.

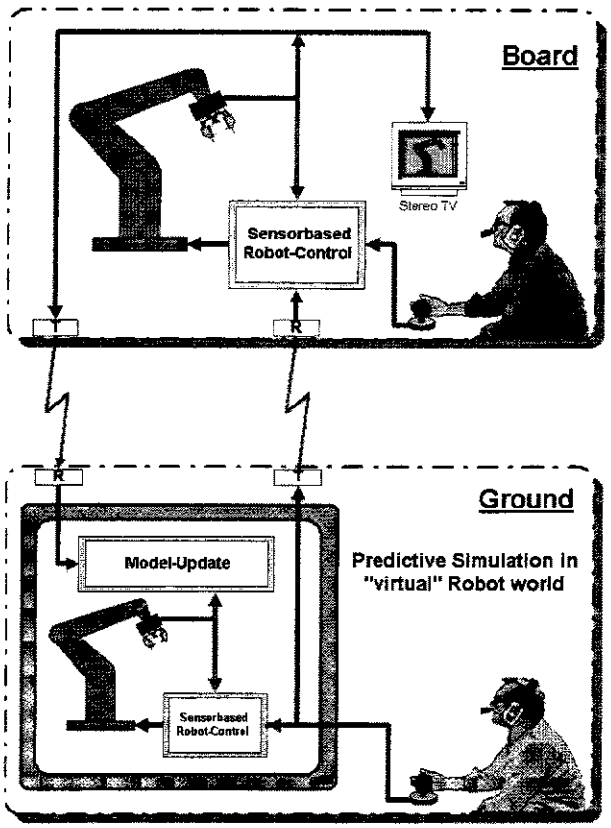


Fig. 4. ROTEX telerobotic control concept.

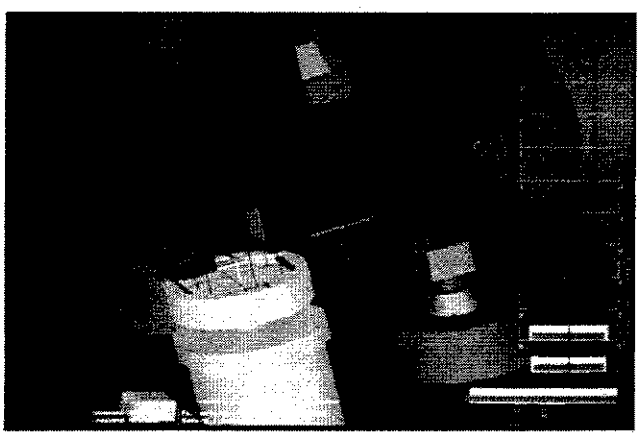


Fig. 5. Predictive simulation of sensory perception in telerobotic ground stations as demonstrated in ROTEX.

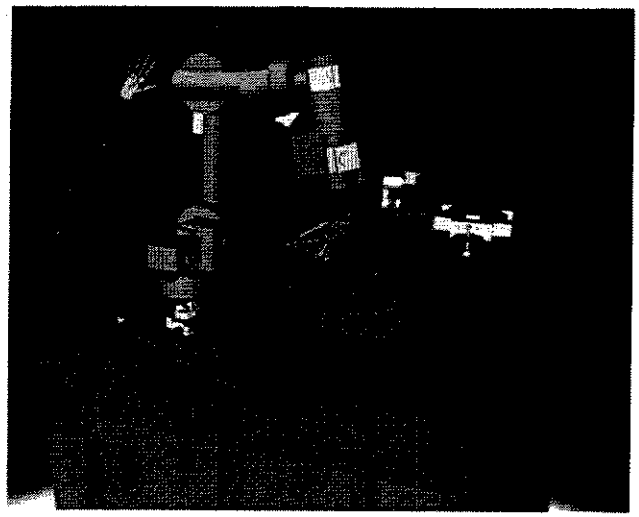


Fig. 6. GETEX/ETS VII: Our goals were (a) vision and force-controlled task-level teleprogramming and (b) verifying the dynamic interaction between robot and satellite.

As for our first goal, a highlight was indeed the teleprogramming of a peg-in-hole task, where in the virtual world we intentionally displaced the standby position of the peg from where the robot had to fetch it. Vision processing on ground using NASDA's tracking markers on the task board and the Jacobian matrix learning beforehand based on real images (as explained below) caused the ETS VII robot to automatically and perfectly adapt to the unexpected situation. The peg-in-hole insertion as such (taking into account the fairly high tolerances) was less critical and made use of NASDA'S compliant motion commands.

As for our second goal, if a robot—mounted on a spacecraft—moves, it generates linear and angular momentum. In the case of an attitude- and position-controlled spacecraft, the attitude control system will permanently produce forces and torques compensating for the arm motion. The spacecraft may then be considered as inertial in the coordinates of a fixed-orbit system, and the problem of robot motion planning can be solved using the same methods as for terrestrial manipulators. Whereas for the control of the spacecraft attitude, electrically powered momentum wheels can be used as well as thrusters, for control of the spacecraft translation fuel, consuming thrusters are the only actuators currently in use. For this reason and because the position errors are generally negligible, most satellites are only attitude controlled. Due to the linear momentum conservation, which states that the center of mass of the system comprising the robot and the satellite is constant, the motion of a manipulator mounted on the satellite will lead to a compensating motion of the satellite. The amount of satellite translation produced depends on the masses of the bodies constituting the system. For space robotic systems that are neither position nor

attitude controlled, the angular momentum conservation law leads further to a rotation of the spacecraft, by an amount that results from the mass and inertia properties of the manipulator links and the spacecraft. It is generally assumed that no external forces act on such free-floating robots (Longman, Lindberg, and Zedd 1987; Dubowsky and Papadopoulos 1993). The free-floating mode of operation is of interest for space robots not only for the reason that attitude control fuel may be saved, which augments the robot life-span, it will also be of importance during repair missions, when the servicing satellite is very close to or in contact with the target satellite: any action of the attitude control system of either of the two satellites during this phase would lead to a collision and thus to potential damage on the two spacecrafts.

As long as the tasks performed with the robot are described in fixed-robot coordinates, the fact that the satellite position remains uncontrolled has no influence. If, however, the task is described with respect to a fixed-orbit coordinate system, as would be the case, for example, for the capturing of a defect satellite, the satellite motion has to be taken into account (see Fig. 7). The equations relating the tool center point motion to the manipulator joint motion, which for robots with an inertially fixed base are purely kinematical equations, become thus dependent on dynamic parameters in the case of free-floating space robots. This influences the path-planning methods that have to be applied. On one hand, singularities, that is, joint configurations in which the robot is not controllable in Cartesian coordinates, are no more a function only of the robot kinematics, but become dependent on the dynamic properties of the robot, too. Therefore, iterative methods based on the direct kinetic equations have to be used instead of the inverse kinematics equations. Moreover, the angular momentum equation makes the system nonholonomic (Li and Canny 1993), which means that the satellite orientation is not a function of the current joint configuration only, but merely a function of the chosen path. Two different paths starting at the same initial configuration of the robot, and leading to the same final configuration, will therefore result in different amounts of satellite rotation—and thus in different final inertial tool center point positions, too. As a consequence, nonholonomy offers the possibility to do a reorientation of the satellite using manipulator motion only, by simply carrying out a closed-loop maneuver in joint space. This kind of maneuver can be employed to significantly augment the workspace of the robot, since it allows turning the satellite into any desired orientation, bringing back the manipulator into its reference configuration. The maximum workspace of a free-floating space robot is thus described by a hollow sphere of which the inner and outer radius are given by the minimum and maximum possible distance between the tool center point and the system center of mass. Another possibility resulting from nonholonomy is that any point that is inside the fixed-base workspace of the robot may be attained with zero satellite attitude error. In the simplest case, this may be done by planning and executing the maneuver as for a robot with a fixed base and adding a closed-loop reorientation maneuver to compensate for the produced attitude error. Path planning for a nonholonomic system has been investigated in the context of cars and wheel-driven robots (Li and Canny 1993). While those systems may generally be considered as planar, the case of free-floating robots demands spatial methods.

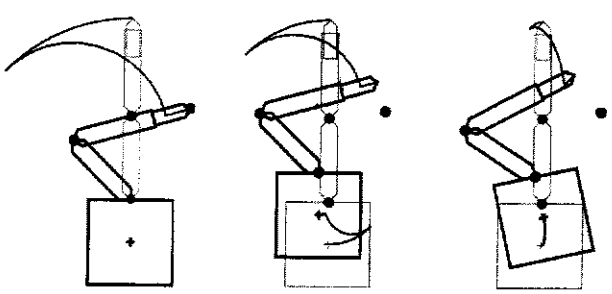


Fig. 7. The influence of the satellite attitude control mode on the path described by the robot end-effector—the same joint motion is carried out by a robot with a fixed base (left), an attitude-controlled base (middle), and a free-floating base (right).

Whatever path-planning method is applied to free-floating robots, it is necessarily highly model-based. The parameters of the dynamic model must therefore be known quite well. While this poses no problem for the geometric parameters and for the mass and inertia of the manipulator, the mass and the inertia of the spacecraft are subject to important changes during the lifetime of a servicing satellite. This is especially the case if the spacecraft is performing capturing or rendezvous/docking-like operations. One goal of GETEX has therefore been to identify the mass properties of the satellite after one year and a half of activity in orbit. Further objectives were the verification of the dynamic models and to obtain some insight into the nature and importance of the disturbances acting on a robotic satellite on low earth orbit. Additionally, the mission aimed at gathering data for the future design of controllers that combine the manipulator motion control with the satellite attitude control. To meet all these objectives, a variety of different maneuvers were executed, which include simple point-to-point operations and closed-loop reorientation maneuvers (examples of which are given in Fig. 8), sequences during which only one joint was active at a time, as well as sequences during which all joints were moving simultaneously. The major constraints, due to mission security aspects, were the maximum satellite attitude error allowed by NASDA, which was limited to $\pm 1.0°$ around each axis, and the fact that the maximum tool center point velocity was limited, too. Furthermore, the reaction wheels were turning at a very low but nonzero constant velocity during the experiments, which introduced undesired torques into

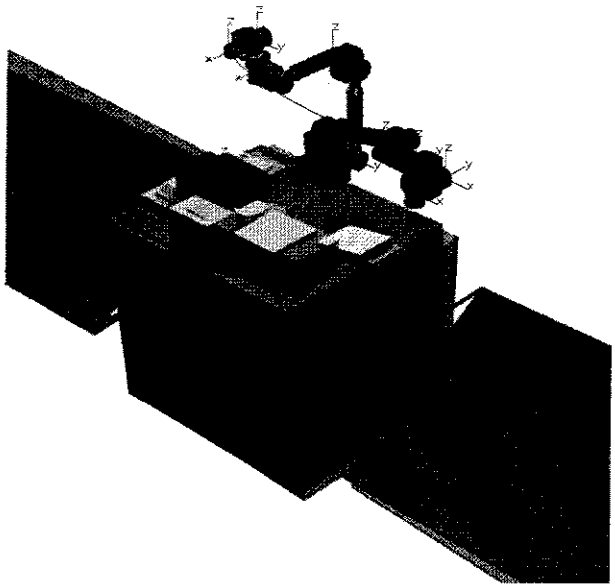


Fig. 8. Examples of dynamic motion maneuvers carried out during the GETEX mission: simple point-to-point maneuver and reorientation maneuver. The shaded robot indicates the reference position. The satellite reaction to the arm motion is scaled by a factor of 10 in this picture.

the system. Their effects will have to be considered during the evaluation of the mission results.

In total, over 110 minutes of dynamic motion experiments were carried out, of which 52 minutes were spent in free-motion mode. The remaining time was used to repeat the experiments in reaction wheel attitude control mode for verification purposes. First evaluations of the measurement data confirm the need to account for external disturbance forces acting on the satellite, such as the gravity gradient torque and magnetic torque.

4. Future Systems

Different manipulator systems operated by the astronauts will work in the construction phase and in routine operations on the International Space Station ISS. The Canadian two-arm system SPDM (special dexterous manipulator system) at the end of a long arm (the space station remote manipulator system SSRMS, mobile along the large truss structure) will presumably be the most remarkable one. In a similar way, the European ERA arm along the Russian part of the ISS and the Japanese arm fixed at NASDA's JEM module are supposed to support astronauts and transport bulky loads. However, it is difficult to understand why—with the ROTEX and ETS VII experience available—so far these arms will have minimal ground-control capabilities only. This bolsters our hope that we will be able to help in making the EuTEF robot at the ISS a demonstrator of highly efficient ground controllability.

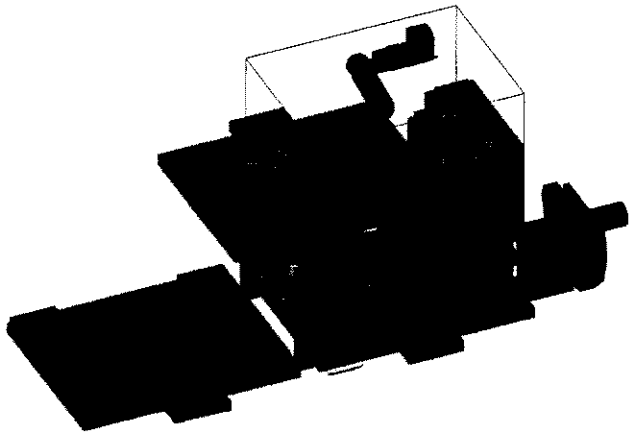


Fig. 9. EuTEF on the space station may become the first remotely operated operational space robot.

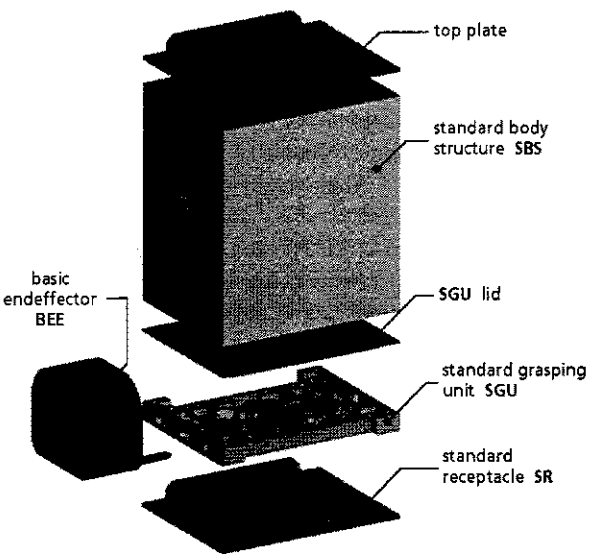


Fig. 10. Payload module PM (exploded view).

The European Technology Exposure Facility, EuTEF, which basically consists of an express pallet on the outer truss structure of the ISS, is supposed to be operated by a robot system.

We are in the process of developing a smart end-effector (stereo-vision and force controlled), designing the complex mechatronic counterpart (the so-called standard grasping units for the pallets and drawers to be handled absolutely safely) and preparing our task-level teleprogramming system MARCO (see Section 7.1) for the high-level remote control of this first purely ground-controlled operational space robot system.

5. Toward a New Service Robot Generation Based on a Mechatronics Approach

5.1. General Remarks

We are deeply convinced that service robotics (and space robotics as a special field) is becoming now a major driver for new, ultralight robots (e.g., on mobile platforms) with manipulative skills and multisensory perception as well as intuitive ways of programming.

Similar to NASA, we have a "robonaut" concept in mind, that might, e.g., relieve astronauts from tedious and dangerous extravehicular activities. However, for us the main difference is not whether this kind of new arm (or two arms if humanoid concepts are pursued) is (are) mounted on a mobile terrestrial platform or on a free-flying satellite. The key issue for us is to have the basic technologies for lightweight arms and articulated multifingered hands available in such a way that these kinds of systems may be duplicated and used by the wide community of robot researchers that is waiting for them.

Our approach in designing multisensory lightweight robots with articulated hands is an integrated, mechatronics one. The new sensor and actuator generation developed over the past few years demonstrates not only a high degree of electronic and processor integration but also a fully modular hard—and software—structure. Analogous signal conditioning, power supply, and digital preprocessing are typical subsystem modules of this kind. The 20 kHz power supply line connecting all sensor and actuator systems in a galvanically decoupled way, and high-speed (optical) serial data bus systems (SERCOS, CAN, or Firewire) are typical features of our multisensory and multiactuator concept.

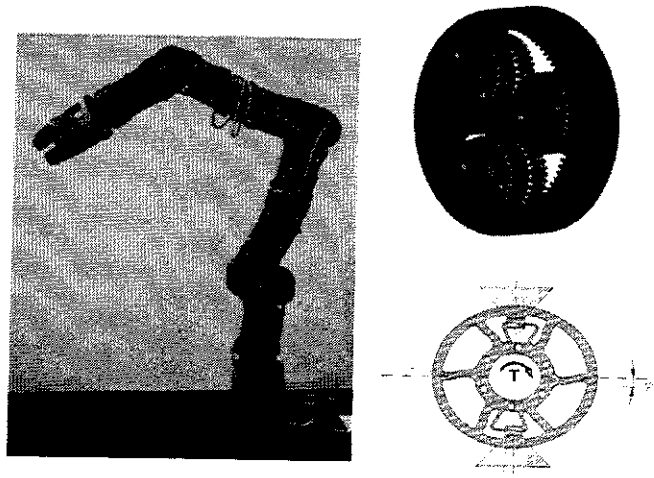


Fig. 11. DLR's first lightweight robot with integrated electronics (left), double planetary gearing, and inductive torque sensing (right).

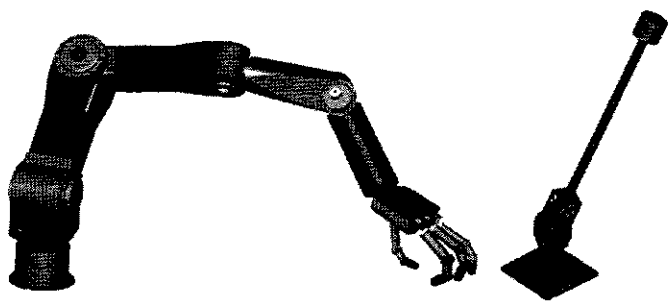


Fig. 12. The CAD model of DLR's new 7-DoF lightweight robonaut arm (left) and the testbed setup of joint 2 (right).

5.2. DLR's Lightweight Robot Design

The design-philosophy of DLR's lightweight robots is to achieve a type of manipulator similar to the kinematic redundancy of the human arm, i.e., with 7 degrees of freedom, a load to weight ratio of between 1:3 and 1:2 (industrial robots $\approx$ 1:20), a total system weight of less than 20 kg for arms with a reach space of up to 1.5 m, no bulky wiring on the robot (and no electronics cabinet as comes with every industrial robot), and a high-dynamic performance. As all modern robot control approaches are based on commanding joint torques, in the first carbon fiber type arm version (Fig. 11) there was an inductive (13 bit, 1 KHz bandwidth) torque-measurement system that was an integral part of a double-planetary gearing system (Gombert et al. 1994). A full inverse dynamics (joint torque) control system including a neural net learning system for compensating gravity modeling errors made use of it.

However, the double-planetary gears (Fig. 11, right) with their extremely high reduction rate of 1:600 were difficult to manufacture.

Meanwhile, a new lightweight robot design (Fig. 12) is under way that tries to make optimal use of all the experience gained with the above "reference" model. Its joints are based on special lightweight Harmonic Drives.

In the drives, we are measuring all relevant state variables, i.e., off-drive position, torque, motor position, and speed (Figs. 13, 14a, and 14b). For torque measurement, we went back to strain gauge–based systems. A first version of this new arm design uses so-called INLAND motors, which were redesigned by us to provide hollow axes where all cabling is fed through.

A second version will use a new motor concept (Fig. 15) as developed in our lab, the optimized external rotor motor (OERM).

The electromagnetic torque generation to be delivered over a wide rotor speed range is realized by a multipole stator assembly interacting with rotor-permanent magnet poles in a nonsymmetrical configuration to virtually eliminate cogging effects. The dynamic performance is significantly enhanced by means of a special commutation control technique based on a single-coil winding technique.

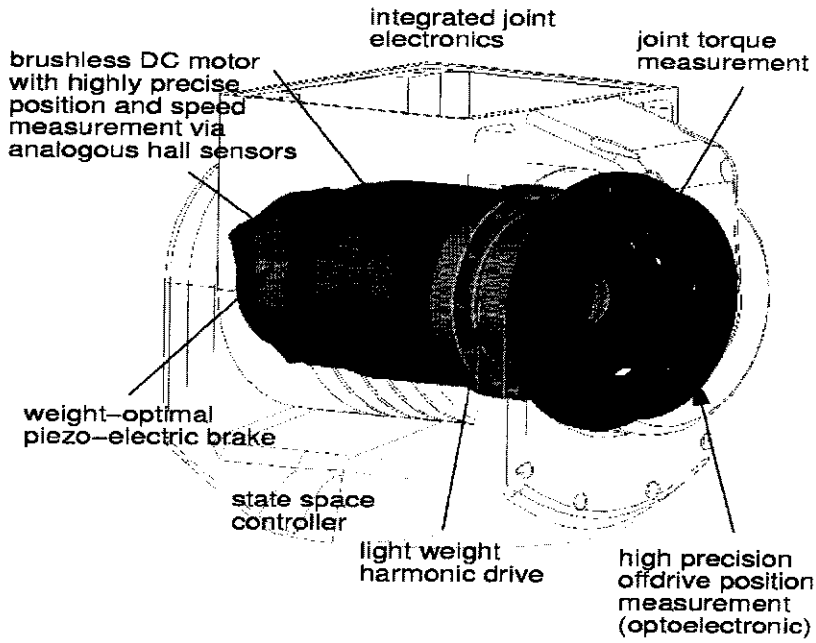


Fig. 13. Mechatronic components of DLR's new intelligent robot joint.

In view of the limited heat exchange to be realized with a compact design, the key design requirement is a large stall-torque-to-input-power ratio. This number can be significantly enhanced as compared to conventional designs by careful tuning of geometrical dimensions and electromagnetic design parameters using magnetic field computations supporting a lumped parameter optimization process.

Not shown here are the hall sensors used for motor position measurement.

The tedious history of weight reduction over the past two years is depicted in Figure 16.

In the first step, the (in our opinion) best commercially available high-end brushless DC-motor was combined with a slightly modified Harmonic Drive gear and a commercially available robotic safety brake.

In the next steps, the total weight was diminished by reducing weight in the Harmonic Drive's circular spline and the development of a weight optimized, modified version of the original, commercially available electromagnetic brake, which was replaced recently by DLR's new piezoelectric brake with a weight of less than half the original brake. Considerable decrease of the drive-unit masses was reached by providing the Harmonic Drive with a new aluminum-crafted wave generator and circular spline as developed in close cooperation with the company Harmonic Drive, so that it finally came out with only half the weight of the original part.

The biggest step toward an extremely lightweight construction was the development of a completely in-house designed optimized external rotor motor (OERM) of high efficiency and stall torque with a highly integrated piezoelectric safety brake. The mass of the motor related to the stall torque at equal power consumption is less than 72% of the originally used high-end motor, and the weight of the integrated brake (30 g) is just one-tenth of the weight of the commercial brake used in the first step (300 g).

The combination of the new OERM, integrated safety brake, and lightweight Harmonic Drive gear yields an extremely powerful lightweight jointdrive with a related mass of just 55% of the weight of the original high-end drive unit and a joint quality measure of $J = 250$, where we have defined this measure as

$$J = \frac{T}{W} \cdot \frac{V_{\max}}{(180°/\text{sec})}, \tag{1}$$

and where

$$T[Nm] = \text{output torque (max)} \tag{2}$$

$$W[kg] = \text{weight of joint} \tag{3}$$

$$V_{\max}[°/\text{sec}] = \text{maximal rotational speed.} \tag{4}$$

Indeed, it is not trivial to compare the performance of lightweight joints, as output torque related to overall weight is

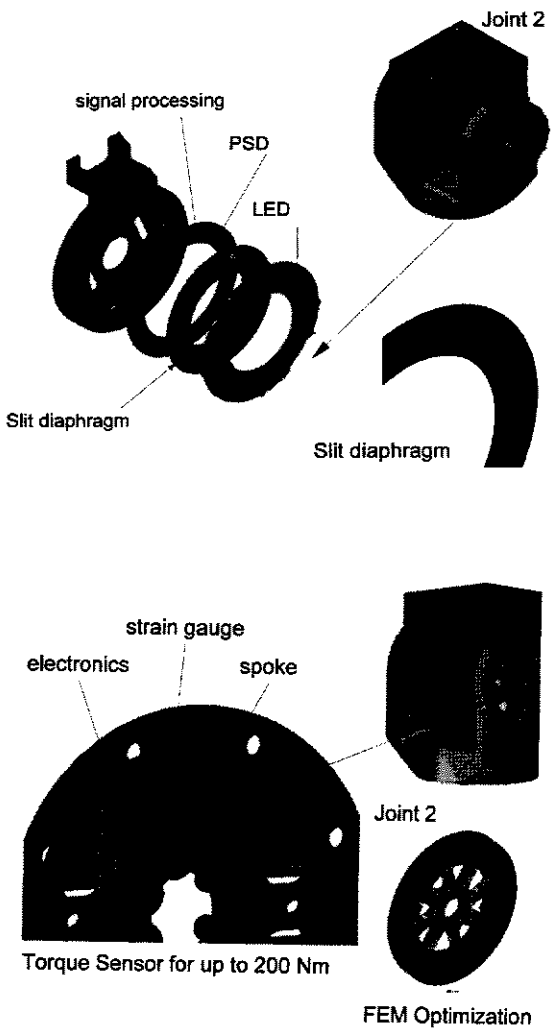


Fig. 14. Two sensors in DLR's new lightweight robot (Fig. 12) (a) Off-drive joint angle sensor (resolution 0.01 degrees) (b) joint torque sensor.

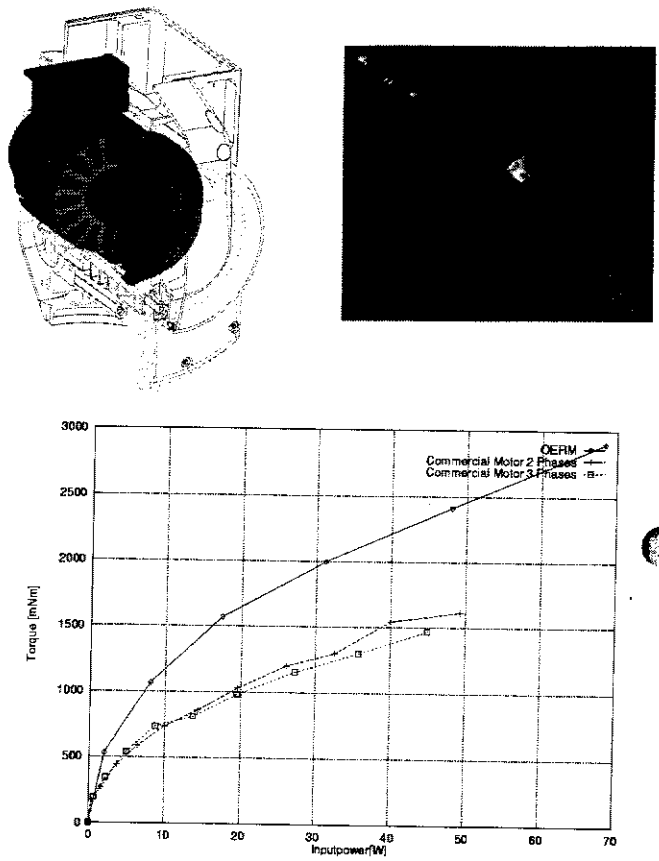


Fig. 15. The optimized external rotor motor (OERM) just needs about 38% of the stall torque input power, which has been required by the best commercial motor used since, and moreover yields 50% higher torques.

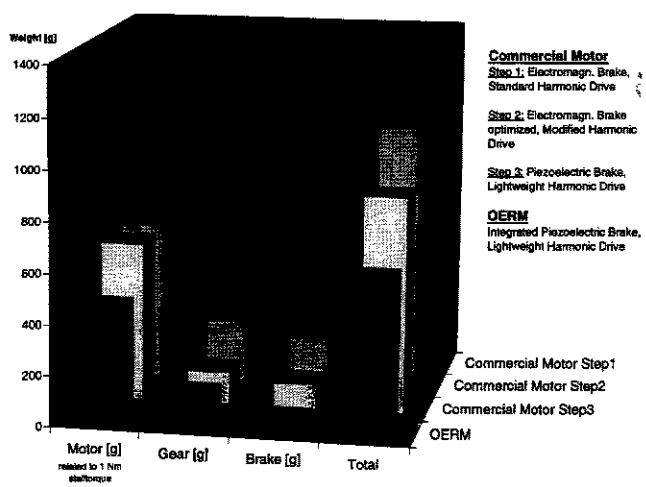


Fig. 16. The history of weight reduction in DLR's new LWR-Drive Units.

meaningless if one does not take into account the joint's maximum rotational speed, which we normalize via 180°/sec, a value that is, e.g., a good standard for terrestrial robots.

In summary, we are convinced that the enormous efforts we made to arrive at joints with approximately 200 Nm output torque, 220°/sec and approximately 1 kg weight including the brake system will pay out in the near future.

5.2.1. *Dynamic Feedback Control*

From the control point of view (Walker and Paul 1980; Khatib 1987), the DLR lightweight robot belongs to the category of flexible joint robots due to the structure of the gear box and the integrated torque sensor (Shi and Lu 1996). The dynamic model can be established by applying Lagrange's equation. For the decoupling of the manipulator dynamics, this model is transformed into a new coordinate system in which the joint torque is treated as a state variable instead of the motor position.

This leads to the so-called singular perturbation formulation of the robot dynamics. As a result, the fast motion corresponds to the joint-torque loop and the slow motion corresponds to the dynamic path concerned with the link position. On the higher levels, particularly interesting control results so far have been achieved with a hybrid learning approach, it is based on a full inverse dynamic model providing torque control, but as no model will ever be perfect, the remaining uncertainties are learned via back-propagation neural nets (Fig. 17). A first impressive demonstration of this type was learning zero-torque control, i.e., pure gravity compensation so that the arm is just able to sustain itself against gravity but reacts softly to any external force at any link without additional force sensing (Hirzinger et al. 1993).

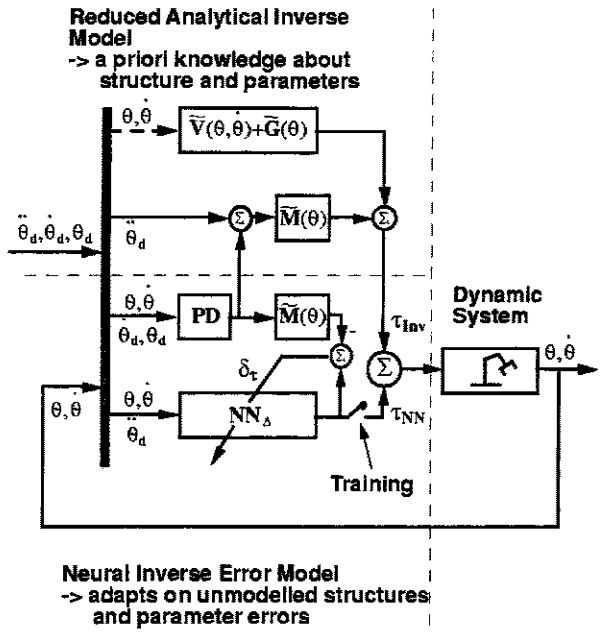


Fig. 17. "Hybrid" learning control scheme applied to gravity compensation of DLR's lightweight robots.

5.3. *DLR's Four-Fingered Articulated Hand*

For many space operations, i.e., handling drawers, doors, and bayonet closures (electric connectors) in an internal lab environment, two-fingered grippers seem adequate and sufficient; the appropriate mechanical counterparts in the lab equipment are easily designed and realized even in a very late design stage. For more complex manipulations, future space robots (robonauts) should use articulated multifingered hands.

In contrast to existing robot hand designs, it was our declared goal to build a multisensory four-fingered hand with a total of 12 degrees of freedom (3 active DoFs in each finger), where all actuators, uniformly based on the position-force-controlled artificial muscle, are integrated in the hand's palm or in the fingers directly (Figs. 19 and 20). This means the hand is fully modular and may be mounted on any robot. Force transmission in the fingers is realized by special tendons (highly molecular polyethylene), which are optimal in terms of low weight and backlash despite fairly linear behavior.

With 112 sensors, around 1000 mechanical, and around 1500 electrical components, the new hand is one of the most

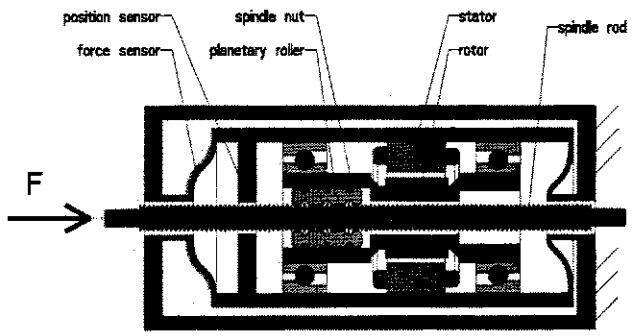


Fig. 18. DLR's planetary roller screw integrated into tiny motors is the basis of the position-force controlled artificial muscle; for use in the hand, special position-torque sensors are used (Gombert et al. 1994). Each finger shows a 2-DoF base joint realized by two artificial muscles and a third actuator of this type integrated into the bottom finger link (phalanx proximal), thus actuating the second link (phalanx medial) actively and, by elaborate coupling via a spring, the third link (phalanx distal) passively. Every finger unit with its 3 active degrees of freedom integrates 28 sensors (Gombert et al. 1994).

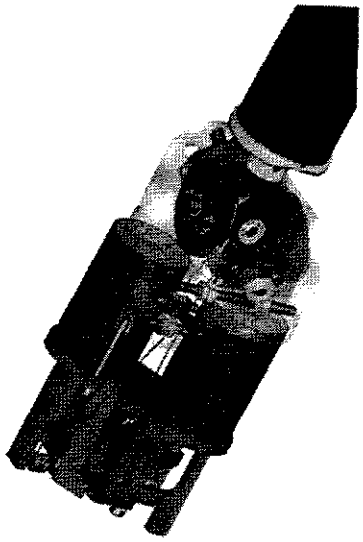


Fig. 19. The 2-degree-of-freedom base joint.

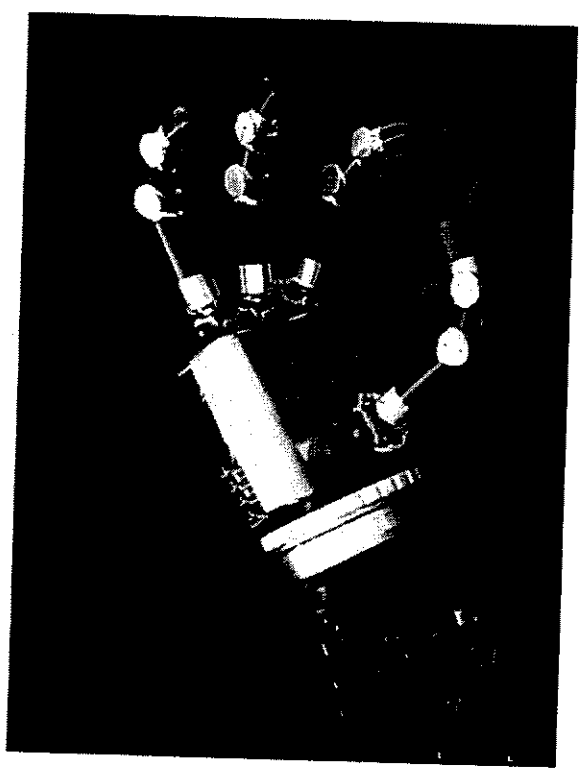


Fig. 20. Our four-fingered hand with its 12 actuators and 112 sensors integrates 1000 mechanical and 1500 electronic components.

complex robot hands ever built. The fingers are position-force-controlled (impedance control), they are gravity compensated, and they are prevented from colliding by appropriate collision-avoidance algorithms. In addition, recently a Cartesian stiffness control scheme on hand level was implemented, which turned out to be of crucial importance for all kinds of manipulation tasks. For more details, see Fischer, van der Smagt, and Hirzinger (1998) and Hirzinger (1997).

A number of telepresence demonstrations have meanwhile been performed using a data glove, a polhemus tracker, and on the "remote" site, the robonaut consisted of a 7-DoF lightweight robot on a three-axis rail system and the four-fingered hand (Fig. 21).

A mockup of the spacelab in our lab allows us to remotely pull drawers, grasp objects in the most natural way, and so on. The robonaut concept of the "prolonged arm" of man in space seems very realistic here (Fig. 12). Needless to say that in case of a long delay, all operations can be programmed and executed in a virtual environment using the MARCO telerobotic system (see Section 4). Mapping the data glove signals into glove finger positions via neural nets (Fig. 22) as well as high- and low-level grasp-planning modules for the position-force controlled fingers are meanwhile available for our hand (Fig. 23).

We have now started with the development of DLR hand II, which will demonstrate an even higher degree of integration. As an example, presently around 400 cables are coming out of the hand; they should be reduced down to fewer than 10 cables in DLR hand II.

6. Issues in Force Sensing and Sensor-Based MMI

6.1. General Remarks

Though hands like the one presented above with more than 100 sensors are of course a real challenge in sensor integration,

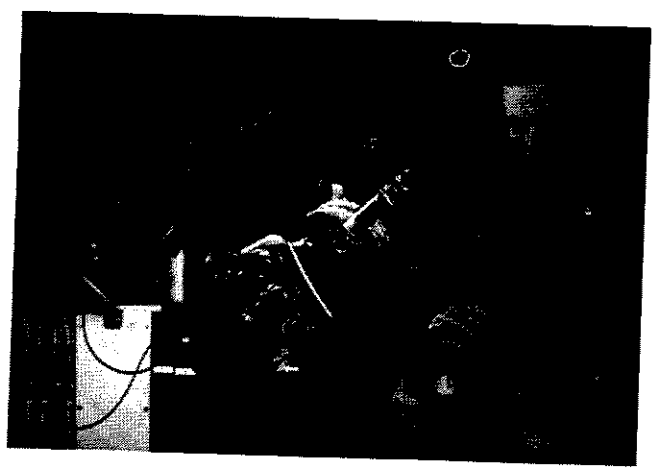


Fig. 21. Skill transfer from human hand to robot hand via data glove.

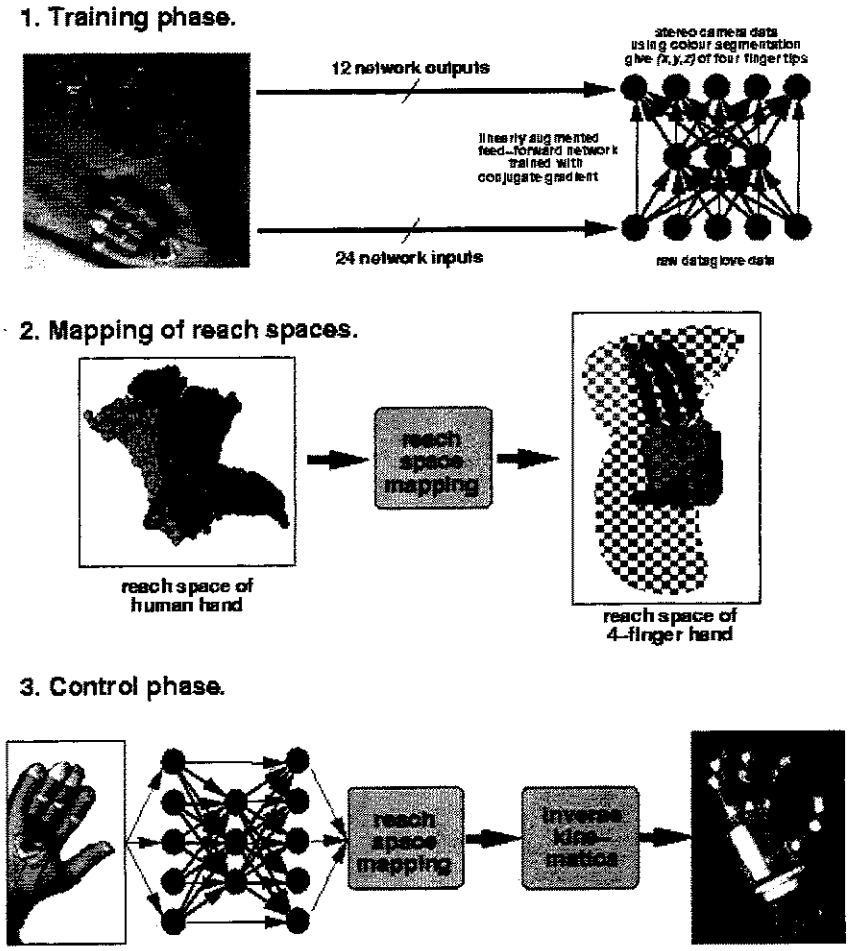


Fig. 22. Data glove control issues for four-fingered hand control.

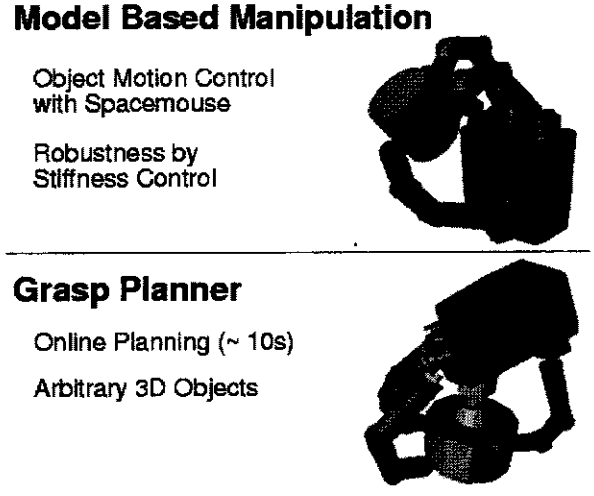
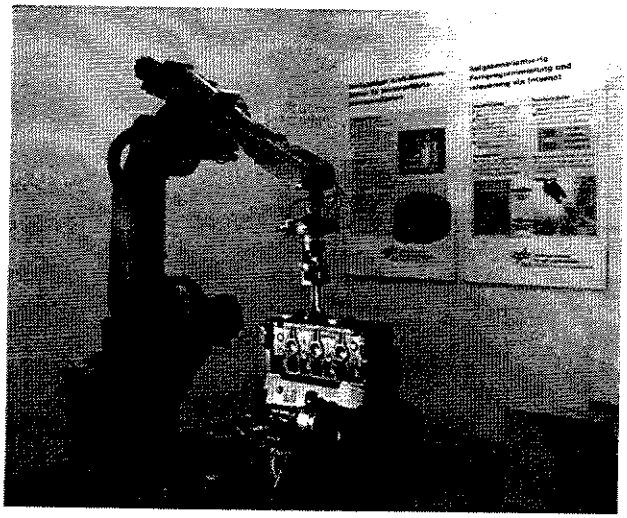


Fig. 23. High-level manipulation and grasp-planning skills are essential for efficient control of DLR's articulated hand.

we should not underestimate the innumerable industrial applications that might attain much higher flexibility and adaptivity by the use and combination of only a few efficient sensory systems. Although we have developed triangulation-based laser range finders and tiny scanners, we are convinced that the combination of real-time 3-D vision and force-torque feedback will have a major impact on industrial robotics over the next few years. In the past, there have been many justified excuses why force-feedback in robotics via positional commands was not really competitive to human performance. Cartesian interpolation cycles (representative for the sampling rates) were longer than 10 msec, and a few of these cycles were needed to feed the reactions on sensory data down to the joint level, while on the other side, we know from numerous experiments performed in the robotics labs, that sampling rates of 250 Hz or higher (i.e., 4 msec or less) without major delays allow the performance of force-controlled assembly operations comparable to the speed humans are used to.

Indeed, we were overwhelmed by the huge interest a force-vision demo generated, which we presented at the 99 Hannover fair, the world's biggest industrial fair. A KUKA



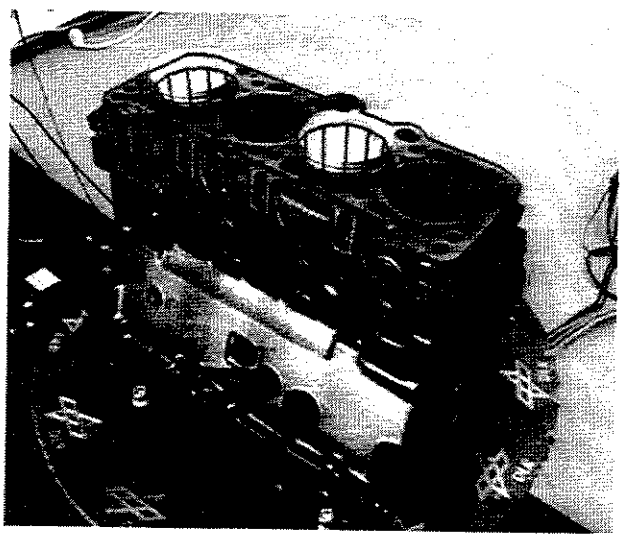


Fig. 24. Vision-force-controlled insertion of noncircular cylinders into rotating motor blocks.

robot equipped with a compliant force sensor (see Fig. 26) and a model-based 3-D-vision system inserted slightly elliptic cylinders into a rotating motor block. The robot did not know a priori the block's position or speed (changeable by spectators), but estimated both via Kalman filters as explained in Section 6.2 and then dived down and performed force-controlled insertion during the motion.

6.2. Force Sensing and Intuitive Motion Control

Stiff force-torque sensors (though not cheap enough) have been commercially available for a couple of years now. However, with our normally stiff industrial robots and a stiff force-torque sensor, force-controllable resolution via positional interfaces may be very low. Sensors that might change their stiffness (as humans do) are not available. So for assembly operations, we propose compliant sensors and we intend to make our own developments in this area commercially available in 2000.

They are based on connecting inner and outer parts via springs and measuring the deflections in the range of 1-2 mm via the opto-electronic system of Figures 25 and 26, which is the core element of Europe's by far most popular 6-DoF man-machine interface for 3-D graphics and 3-D CAD, our Space Mouse.

It is difficult to understand why it took years until robot manufacturers realized that intuitive 6-DoF robot control with

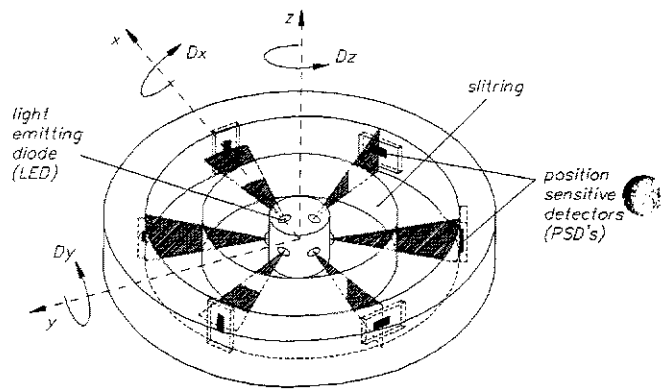


Fig. 25. The patented opto-electronic measuring system.

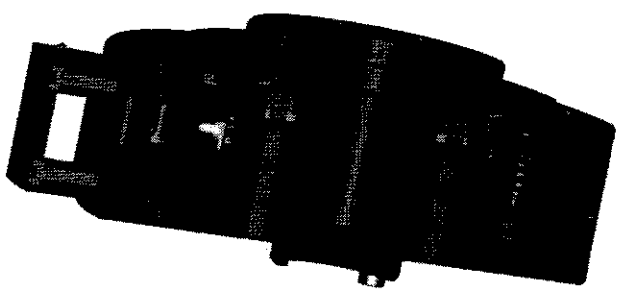


Fig. 26. The compliant force-torque sensor.

such a device is highly efficient in the programming phase (Hirzinger and Heindl n.d.). Now more and more of them are integrating the Space Mouse cap into their programming panels (Fig. 28).

The German manufacturer REIS even allows one to remove it there, fix it magnetically at the robot's wrist (Fig. 28), and guide the robot around in the most natural way (via nulling forces/torques).

Now there is one major drawback if such a compliant-sensing principle is used as a wrist sensor.

In case the robot is moving around the load quickly before starting the assembly process, the thus generated mass-spring system tends to vibrate. Passive or active damping concepts seem to be indispensable here and are under development.

We feel that these types of compliant force sensors should not cost more than, e.g., U.S.$2000, to bring force control into a large number of industrial applications. But again pricing is not the only issue. What we need in force control are standard procedures and software modules so that the normal robot programmers (which are of course not robot researchers) are able to program compliant tasks in an easy, convenient way.

Fig. 27. Space Mouse (in USA and Asia called Magellan by our license partner Logitech).

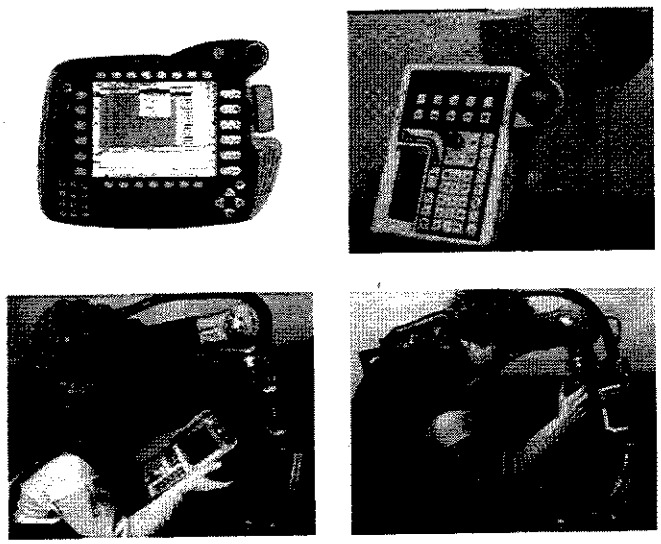


Fig. 28. New generation of programming panels with 6-DoF intuitive control (KUKA and STÄUBLI first row, REIS second row).

6.3. Enabling Innovation in Force Control

Thus, three enabling factors tend to speed up the innovation in applications of robot force control:

- The availability of reliable but low-cost force/torque wrist sensors.
- Fast robot controllers that enable sampling times toward 1 ms.
- Algorithms to ease the setup of force control applications.

To accelerate the setup of compliant motion controlled robots, we have devised an automatic design procedure to compute the structure and the parameters of the force controller for the inner position/outer force implementation (Natale, Koeppe, and Hirzinger 1999). The approach applies to standard position-controlled industrial robots with cascaded controllers that impose Cartesian decoupled behavior on the end-effector and are equipped with a force/torque wrist sensor.

Experiments show that industrial robots can be described by simplified models up to a bandwidth of 10-15 Hz (Hirzinger 1982), which is well above the human bandwidth of approximately 1-2 Hz required to perform sensory-controlled tasks. Such a simplified model can be parameterized by a few but important technology parameters, which can be easily measured or provided by the robot manufacturer. Process parameters are specified in a fuzzy way, which enables one to specify, e.g., the performance type (slow, medium fast, fast) or the surface type (soft, medium stiff, stiff).

The design technique can also handle the presence of a time delay due to the digital implementation of the controller. The designed controller guarantees high gain and phase margins for the closed-loop system, which implies robustness of the system.

We have implemented the force control structure, the automatic design procedure and utility functions like sensor gravity compensation, identification of the end-effector weight and its center of mass, and other functionalities into a modularized object-oriented software library, the DLR Force Control Software Library (FCL) using the C programming language. Even though the implementation in C++ would be straightforward, we have chosen an object-oriented implementation

in C to allow the use of the library on floating-point micro- or signal processors.

We have used the software library to implement force control with different industrial robots equipped with compliant force sensors and have achieved excellent results in a small amount of setup time. The FCL library will be extended with modules for stiffness observation and impedance (stiffness) control.

7. Issues in 3-D Vision

7.1. Multisensory Model-Free Servoing

The focus of our work is on real-time image postprocessing for finding the pose of objects to enable the robot to grasp them, or to get their shape to perform object recognition and world model update. We can give only a few examples here.

The purpose of robot servoing is to position a robot, sensory-controlled, to a desired pose relative to an object, as characterized by nominal sensory patterns. Contrary to conventional methods, which need camera and hand-eye calibration, we have developed two approaches that do not need calibrations yet allow us to process redundant multisensory information (robustness) and are based on learning by showing. In both cases, the control law may be written as

$$v_C = aC(\alpha - s^*)$$

where $(s - s^*)$ is the vector-valued deviation between the actual and the nominal sensory pattern indicating the displacement of the actual robot pose x from the nominal pose, x^*; v_C is the velocity command; α represents a scalar dynamic expression, at least a real constant, determining the closed-loop dynamics; and $\mathbf{C}$ represents a projection operator used for mapping the sensor space onto the control space. $\mathbf{C}$ is either determined by analytical methods (projection operator-based on estimation theory corresponding to the inverse of a Jacobian) or by neural network learning using multilayer perceptrons. Both cases need a kind of training by moving the virtual or real robot around its nominal situation in the graphics or real world (Fig. 29).

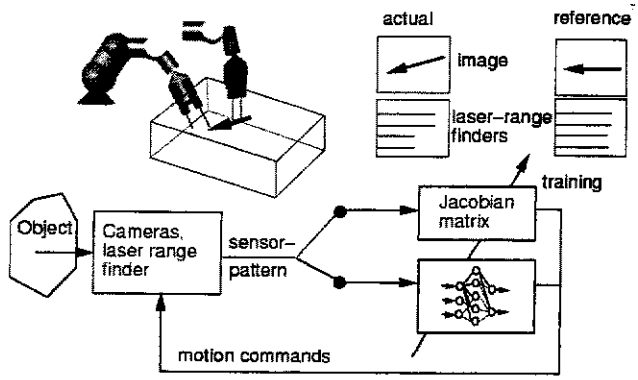


Fig. 29. Multisensory servoing into nominal situations when no 3-D models are available.

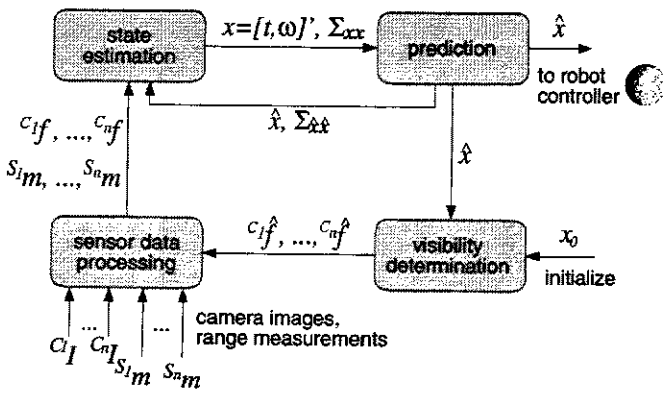


Fig. 30. Processing steps in model-based multisensory 3-D tracking.

7.2. Model-Based Visual Servoing

Here we assume that model information is available in the form of a polyhedral description of the 3-D object geometry, enhanced by information about circular features (multisensory "eye-in-hand" systems).

The structure of the model-based multisensory tracking algorithm (Wunsch and Hirzinger 1997) is illustrated in Figure 30. The task is to track the relative pose x of the target. Image feature vectors ^{C_i}f along with the measurements $^{S}j_m$ of the range-sensing devices are the input to a least squares procedure that computes an estimate x of current pose and its covariance Σ_{xx}. Next, to compensate delay and to prepare feature extraction, a prediction $\hat{\mathbf{x}}$ and its covariance are computed based on assumptions of the object motion, which is used to generate feature values $^{C_i}\hat{f}$ expected in the next processing cycle. Thus, image processing can be limited to small regions of interest and potential occlusions can be predicted by hidden line removal. This structure is equivalent to that proposed by Dickmanns and Graefe (1988) for monocular tracking via 2-D projection.

As a typical example, we investigate the vision-based (model-based) capturing of a target satellite by a free-floating repair robot (Experimental Servicing Satellite, ESS, Fig. 32). The prediction module of Fig. 30 including satellite dynamics here is implemented as a Kalman filter that takes x and S_{xx} as an input (observation) and continuously estimates the parameters of the dynamic equations. From these estimates, both a smoothed version of the object pose and a prediction are computed in each cycle.

In addition we developed different types of visual servoing for laparoscopic camera guidance in minimally invasive surgery based on stereoscopic colour segmentation (Wei, Arbter, and Hirzinger 1997). The surgeon's instruments are marked with colours which normally do not occur in a hu-

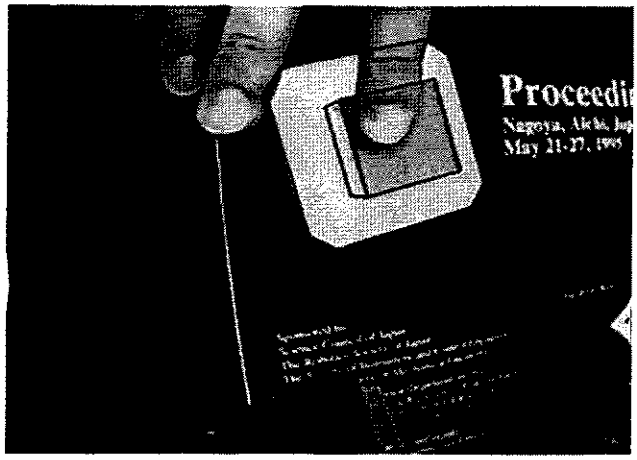


Fig. 31. Monocular real-time 3-D tracking: The wire frame of the object is projected into the image at the estimated pose. The good match indicates high accuracy. Minor occlusions do not affect accuracy.

man's body. Starting in 1996 dozens of patients were safely operated on in a Munich hospital with this type of robotic assistance (Figs. 46 and 47).

7.3. Model-Based Pose Estimation by Registration

Novel algorithms (Wunsch and Hirzinger 1996) have been developed for registering a 3-D model to one or more of two-dimensional images without a priori knowledge of correspondences. The method—a generalization of the iterative closest point algorithm (ICP)—handles fully 6 degrees of freedom, without explicit 3-D reconstruction.

Experiments show that complex, three-dimensional CAD-models can be registered efficiently and accurately to images, even if image features are incomplete, fragmented, and noisy (Fig. 33).

7.4. 3-D Reconstruction

7.4.1. Shape from Shading by Neural Networks

The problem of shape from shading is to infer the surface shape of an object based solely on the intensity values of the image. We proposed a new solution (Wei and Hirzinger 1996) of the shape from shading problem based on the optimization framework (Smagt and Hirzinger 1996). We used a multilayer perceptron to parameterize the object surface.

The weights of the network are updated so that the error between the given image intensity and the generated one is minimized. Figures 34 and 35 illustrate the mechanism with which the shape is recovered from shading. We also showed that knowledge about object surface, e.g., known depths or orientations at some positions, can be easily incorporated into the shape from shading process so that errors due to lack of boundary conditions or self-shadows can be reduced.

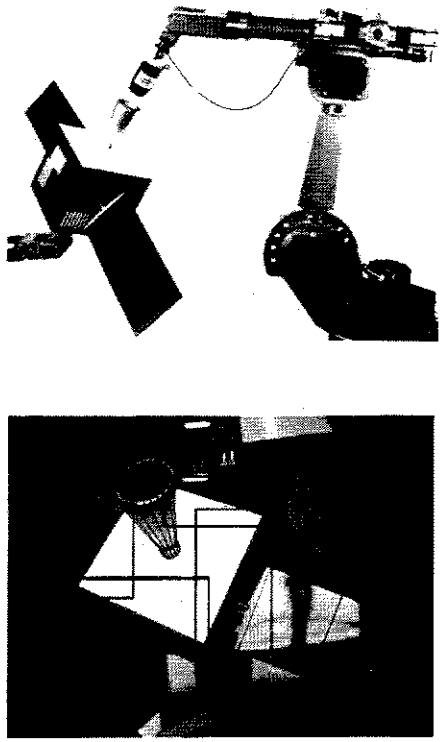


Fig. 32. (a) A two-robot system as testbed for a satellite repair project. The robot on the left carries a mockup of a satellite that is tracked by a camera mounted in the tool of the repair robot (right). (b) Satellite tracking as seen from the repair manipulator's hand camera. The wire frame model of the target is projected into the live video image at the currently estimated pose.

7.4.2. A New Stereo Approach

One of our preferred technologies in 3-D reconstruction from stereo images is sketched in Figure 37; it uses multilayer perceptrons based on radial-basis functions, solves the disparity problem automatically, and needs extremely small contrasts only (Wei and Hirzinger 1998).

7.4.3. World Modeling Using Kohonen's Feature Map

Building a geometric description of a robot's workspace via unordered "clouds" of multisensory data out of the robot gripper is of crucial importance for robot autonomy.

To solve these kinds of problems, a surface reconstruction algorithm, based on Kohonen's self-organizing feature map, was developed and successfully implemented (Baader and Hirzinger 1994). The two-dimensional array of neurons in Kohonen's algorithm may be considered as a discrete parametric surface description. To incorporate different types of

surface information, new training equations were developed for the Kohonen network. Successful surface reconstruction was performed for completely unordered data clouds of different surface information (Fig. 38). This method was used as a core for a complete world perception system for sensor-in-hand robots.

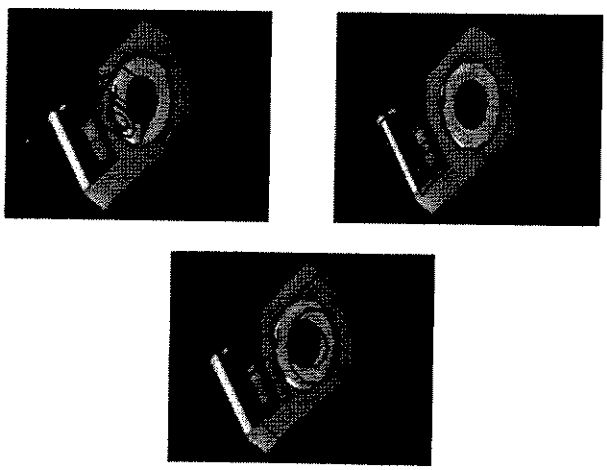


Fig. 33. An example of convergence of the registration algorithm. The initial rotational displacement in ZYX-Euler angles is {66.5°, 49.5°, 32.0°}.

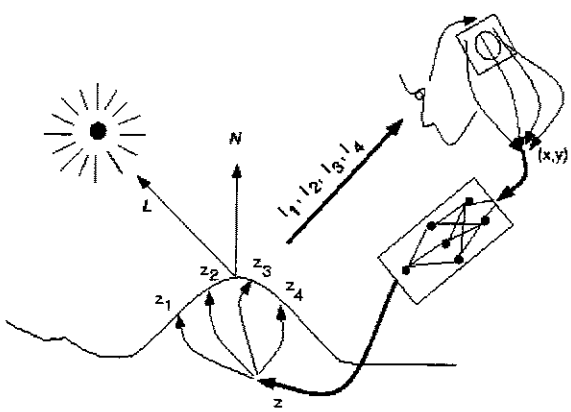


Fig. 34. Shape from shading: A learning scheme minimizes the intensity error.

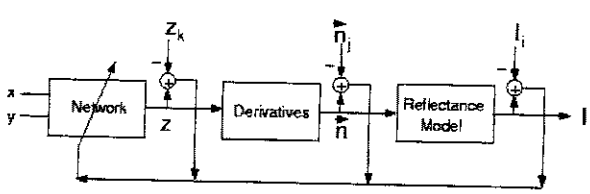


Fig. 35. A unified training framework.

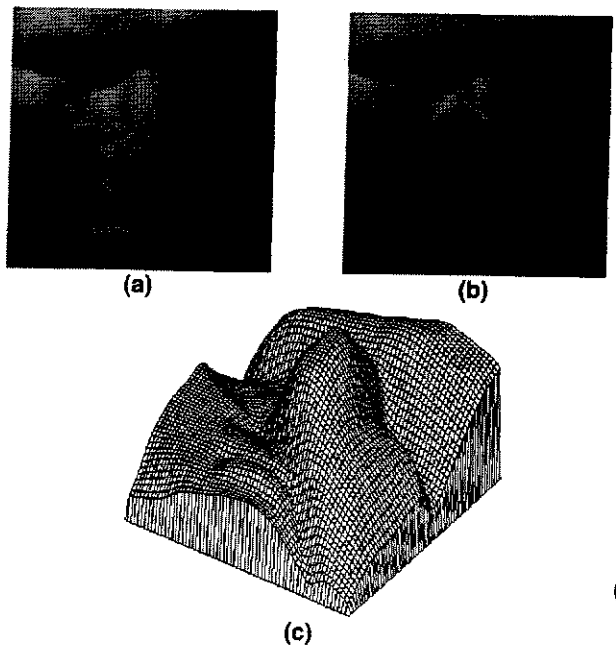


Fig. 36. Surface recovery of the Agrippa statue. (a) Input image; (b) Learned image; (c) Recovered 3-D surface.

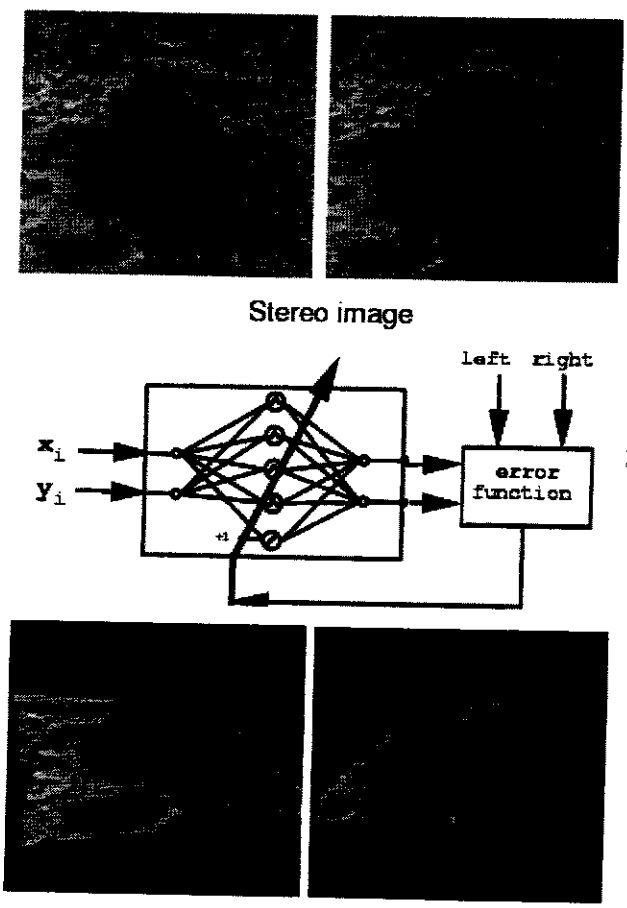
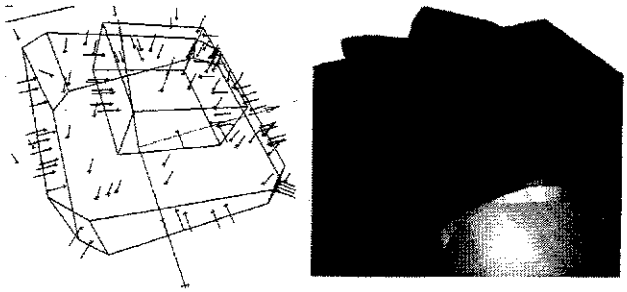


Fig. 37. 3-D reconstruction with neural networks; example: stereo image of stone Yogi/Pathfinder Mars Mission.



(a) (b)

Fig. 38. Self-organizing reconstruction of an object in the robot's work cell. Part (a) shows the wireframe representation of the object and the sampled 110 point and normal vector tuples; Part (b) shows the final surface after 3000 steps and a total time of 9 seconds.

8. Programming by Showing and Demonstration

8.1. MARCO: An Advanced Task Directed Telerobotic System

Following ROTEX, we have focused our work in telerobotics on the design of a high-level task-directed robot-programming system MARCO, which may be characterized as learning by showing in a virtual environment (Brunner et al. 1999) and which is applicable to the programming of terrestrial robots as well. The goal was to develop a unified concept for

- a flexible, highly interactive, online programmable teleoperation station, as well as
- an off-line programming environment, which includes all the sensor-based control and local autonomy features as tested already in ROTEX, but in addition provides the possibility to program a robot system on an implicit, task-oriented level.

A nonspecialist user—e.g., a payload expert—should be able to remotely control the robot system in case of internal servicing in a space station (i.e., in a well-defined environment). However, for external servicing (e.g., the repair of a defect satellite), high interactivity between man and machine is requested.

To fulfill the requirements of both application fields, we have developed a 2in2-layer-model, which represents the programming hierarchy from the executive to the planning level.

On the implicit level, the instruction set is reduced to what has to be done. No specific robot actions will be considered at this task-oriented level. On the other hand, the robot system has to know how the task can be successfully executed, which is described in the explicit layers.

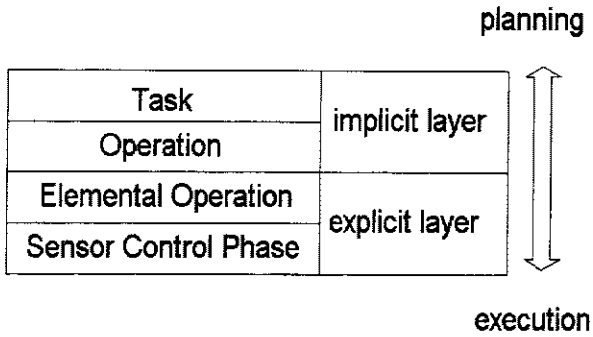


Fig. 39. 2in2-layer-model.

8.1.1. Reflex (Sensor Control Phase)

At the lowest level of the MARCO system, the sensor control mechanism is active. These so-called reflexes guarantee the local autonomy at the remote robot's site via using sensory data processing algorithms in an extensive way. The teaching by showing paradigm is used at this layer to show the reference situation, which the robot should reach, from the sensor's view: in the virtual environment, we store the nominal sensory patterns and generate appropriate reactions (of robot movements) on deviations in the sensor space.

A reflex is described by

- A controller function, which maps the deviation in the sensor space into Cartesian robot move commands
- A state recognition component, which detects the controller's end conditions (success, failure)
- The constraint frame information, which supports the controller function with the task frame data to interpret the sensory data correctly (e.g., for shared control)
- A sensor fusion algorithm, if sensor values of different types have to be transformed into a common reference system (e.g., vision and distance sensors)

8.1.2. Elemental Operations

The explicit programming layer is completed by the elemental operation (ElemOp) level. It integrates the sensor control facilities with position and end-effector control. According to the constraint frame concept, the non-sensor-controlled degrees of freedom of the Cartesian space will be position controlled

- in case of teleoperation directly with a telecommand device like the SpaceMouse.
- in case of off-line programming by deriving the position commands from the selected task. Each object, which can be handled, includes a relative approach position,

determined off-line by moving the virtual end-effector in the simulation into the desired pose with respect to the respective object and storing the geometrical relationship between the object's reference frame and the tool center point.

It should be mentioned that the ElemOp layer aims at a manipulator-independent programming style: if the position and sensor control function are restricted to the Cartesian level, kinematical restrictions of the used manipulator system can be neglected. This implies the general reusability of ElemOps in case of changing the robot type or modifying the work cell.

A model-based online collision detection supervises all the robot activities. For global transfer motions, a computational very fast path-planning algorithm avoids collisions and singularities in the robot's joint space.

8.1.3. Operations

Whereas the Reflex and ElemOp levels require the robotics expert, the implicit, task-directed level provides a powerful man-machine-interface for the nonspecialist user. We divide the implicit layer into the operation and the task level.

An *operation* is characterized by a sequence of ElemOps, which hides the robot-dependent actions. The robot expert is necessary only for the specification of an operation, because she or he is able to build the ElemOp sequence. For the user of an operation, the manipulator is fully transparent, i.e., not visible.

We categorize the operation level into two classes:

- An *object-operation* is a sequence of ElemOps, which is related to a class of objects available within the work cell, e.g., GET <object>, OPEN <door>.
- A *place-operation* is related to an object, which has the function of a fixture for a handled object, e.g., INSERT <object> INTO <place>. Before a place-operation can be activated, the corresponding object-operation has to be executed. <object> is the object, known from the predecessor object-operation, <place> the current fixture, to which the object is related.

Each object in the work cell environment can be connected with an object-operation and/or a place-operation. Because an operation is defined for a class of objects, the instantiation of formal parameters (e.g., the approach frame for the APPROACH-ElemOp) has been done during the connection of the operation with the concrete object instance.

To apply the operation level, the user only has to select the object/place, which she or he wants to handle, and to start the object-operation/place-operation. For that reason, the programming interface is based on a virtual reality (VR) environment, which shows the work cell without the robot system. Via a 3-D interface (data glove or a 3-D cursor, driven by the Space Mouse), an object can be grasped and moved to an appropriate place. If the user has moved all the objects to the places she or he wants, the execution of the generated task can be started by doing a specific VR-hand gesture. For supervision, the system shows the state of the operation execution, i.e., the ElemOp, which is currently active. Also the position and orientation of the currently moved object is fed back.

8.1.4. Tasks

Whereas the operation level represents the subtask layer, the possibility to specify complete robot tasks must be available in a task-directed programming system. A *task* is described by a consistent sequence of operations, which are instantiated with concrete object instances (see Fig. 40). To generate a task, we use the VR-environment as described above. All the operations, activated by selecting the desired objects or places, are recorded with the respective object or place description.

Based on this four-level hierarchy (Brunner et al. 1999), an operator working on the (implicit) task level no longer needs real robotic expertise. With a 3-D cursor (controlled by a Space Mouse) or with a human-hand-simulator (controlled by a data glove), he picks up any desired object in the virtual world, releases it, moves it to a new location, and fixes it there. Sequences of these kinds of operations are easily tied together as complex tasks, and before they are executed remotely, the simulated robot engaging its path planner demonstrates how it intends to perform the task *implying automatic collision avoidance*. To have real stereo-graphic imagination, we use either shutter glasses with stereo monitors or polarized glasses with large screens where many observers can watch at the same time. Stereo impression is perfect in both cases.

Nevertheless, in the explicit layer (the learning phase), the robot expert has to show and demonstrate the elementary operations including the relevant sensory patterns and—if necessary—train the mapping between nonnominal sensory patterns and motion commands that servo into the nominal patterns later on in the real world. He or she performs these demonstrations by moving the robot's simulated gripper or hand (preferably without the arm) into the proximity of the objects to be handled (e.g., drawers, bayonet closures, doors in a lab environment), so that all sensory patterns are simulated correspondingly. The robot expert at this stage, of course, must have knowledge on position- and sensor-controlled subspaces (and must be able to define them, massively supported by MARCO functions), and he or she has to define how operations (e.g., remove bayonet closure) are composed by elementary operations (approach, touch, grasp, turn, etc.).

8.1.5. MARCO's Two-Handed VR Interface Concept

Thus, as a general observation, on the implicit as well as on the explicit layer statement, we have to move around 3-D

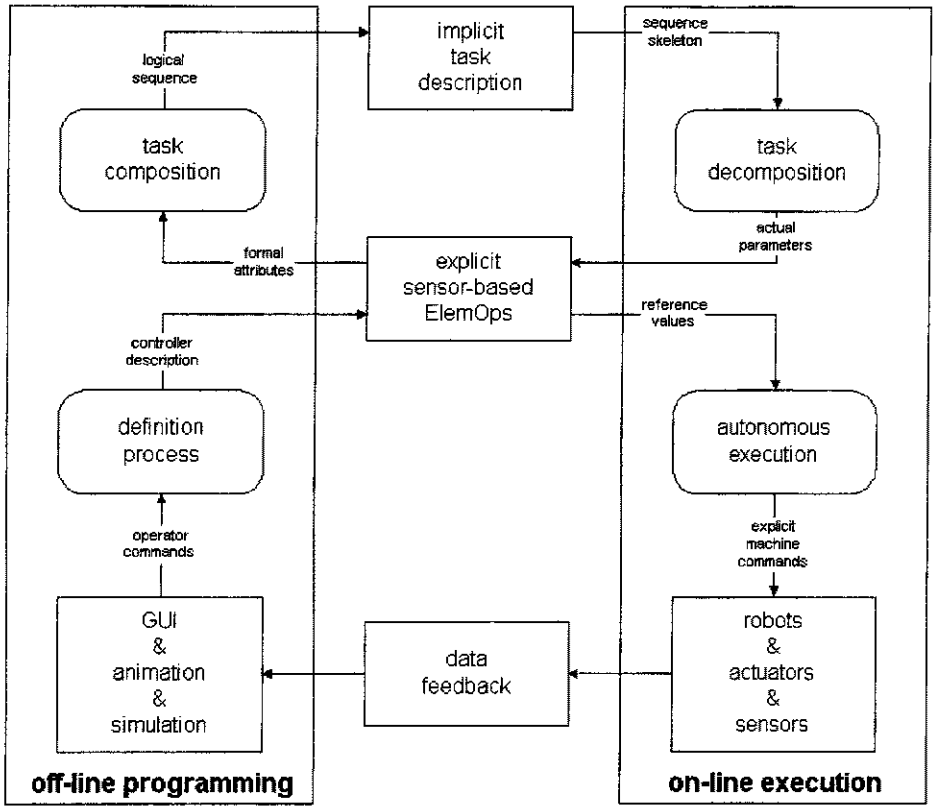


Fig. 40. Task-directed sensor-based programming.

Fig. 41. DLR's universal telerobotic station MARCO (Modular A&R Controller).

pointers or grippers/hands in the virtual lab environment. Using classical "immersive" cyberspace techniques with data glove and helmet was not adequate for our approach, as the human arm's reaching space is fairly small (e.g., in a lab environment), and with head motions only very limited translational shifts of the simulated world are feasible. As a general observation, an alternative to the position control devices, data glove and helmet, is the velocity control device, Space Mouse, particularly if the robot system to be programmed has no articulated hand. Velocity control here means we may easily steer around an object in VR over arbitrary distances and rotations via small deflections (which generate velocities) of an elastic sensorized cap. The second important observation (confirmed by extensive tests of car manufacturers in the context of 3-D CAD-design) is that, just as in real-life, two-handed operations when interacting with 3-D graphics are the optimum. Indeed, whenever humans can make use of both hands, they will (e.g., when carving, modeling, cutting). In the Northern Hemisphere, for around 90% of the people, the right hand is the working hand, while the left hand is the guidance and observation hand, which holds the object to be worked on (and vice versa for left-handers).

This ideal situation for a human is easily transferred to the VR interface scenario. A right-hander preferably moves

around the whole virtual world in 6 DoF with a Space Mouse in his left hand (the guidance hand), while with his right hand he moves around the 3-D cursor with a second Space Mouse (velocity control, Fig. 42) or a simulated hand with a data glove (position control, Fig. 43). One should note that now, even with the glove, the problem of limited workspace disappears, because with the left hand the operator is always able to move the virtual lab world around such that the objects to be grasped are very close, so that even in position control mode with a data glove only, small, convenient motions of the operator's hand are needed to reach them.

More details on MARCO's high-level user interface, as well as Java/VRML client techniques, are given in Brunner et al. (1999).

One of the key features of the MARCO system is the implication of sensor-based autonomy using the above-mentioned storage of nominal sensory patterns.

Indeed, for comparing the real world with the virtual world, based on possible multisensory perception, and thus for either updating the world model and/or servoing into a nominal situation as learned during the explicit layer training phase, MARCO provides several alternatives:

- if we have reliable CAD models of the objects and the environment and if we may assume that spatial 3-D contours are well detectable by a (mono or stereo) vision system, we prefer 3-D model-based real-time tracking algorithms; see Section 6.2.
- if we have no 3-D CAD models, or situations where they are not useful (e.g., if the camera does not see real 3-D contours) or where the sensor fusion aspect is prevailing (e.g., cameras combined with arrays of range finders), we prefer to train a linear mapping from sensory input patterns into corrective motions; see Section 6.1.

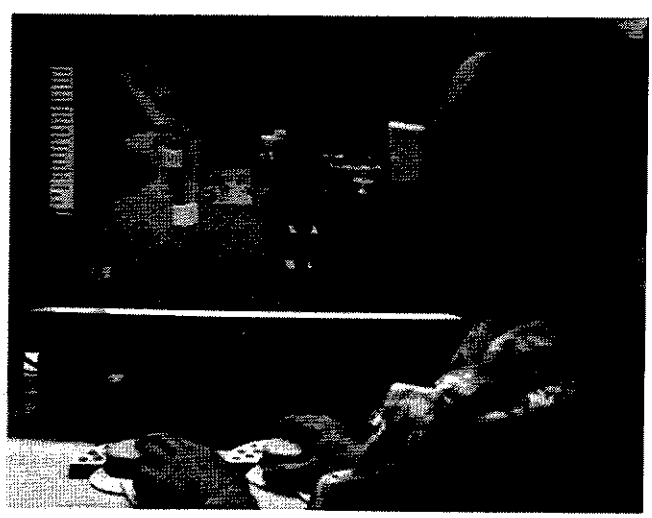


Fig. 42. Two-handed VR-interface using two Space Mice (ETS VII scenario as example).

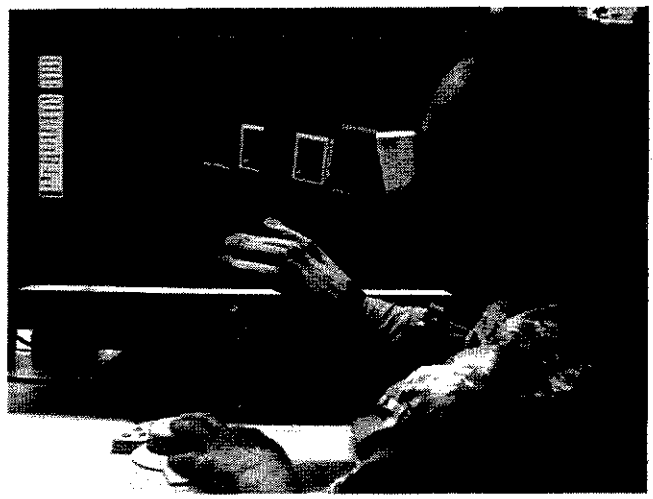


Fig. 43. Two-handed VR-interface using Space Mouse and Data Glove (EuTEF scenario as example).

The latter method has been used in the DLR-NASDA cooperation project GETEX with the ETS VII free-flying robot. In this case, learning of the Jacobian was originally performed with simulated sensory patterns, and then repeated in Tsukuba, Japan, with the real images yielding very similar results.

The sensor-based task-level teleprogramming system MARCO has reached, meanwhile, a high level of universality. It was not only used as a ground control station for the ETS VII experiment, but it is now also used for technology studies of Germany's technology project ESS as well as for remote ground control of EuTEF and for mobile terrestrial (fetch and bring services in hospitals) and planetary robot projects.

8.2. Sensorimotor Skill Transfer of Compliant Motion

The transfer of skill from human to robot through learning by showing as implemented in MARCO is not restricted to contactless elemental moves. For compliant motion tasks, we have developed a consistent approach of sensorimotor skill transfer of compliant motion, taking into account the different properties of human and robot arm dynamics.

We derive robot sensory compliant motion from human demonstration. The sensorimotor behavior of the operator has to be acquired using an acquisition system. Consistent data from the human operator can be observed if the task dynamics are not perturbed by the intervening dynamics of the demonstration system. A "natural feel" is achieved for the operator if the acquisition is designed to display the task dynamics through an ideal response.

Observing the operator directly and acquiring the task within a VR environment (Koeppe and Hirzinger 1995) are the most powerful approaches for implementing the ideal response. For the former one, we have designed a teaching

device for kinesthetic sensing (Koeppe, Breidenbach, and Hirzinger 1996). The operator's hand pose and interaction force are measured during the demonstration (Fig. 44).

The sensorimotor compliant motion skill is identified from perception and action signals of the human operator. As a representation, we use linearly augmented feed-forward neural networks that can represent any nonlinear mapping.

Since the compliant motion skill map defines an implicit representation of contact states and task strategy, the perception signals have to be chosen to allow a unique classification of contact states. We call such signals *geometric perception signals* (as opposed to dynamic perception signals), which enable sensing of the compliant motion dynamics (Koeppe and Hirzinger 1999) (Fig. 45).

We generated compliant motion skill maps for three independent sensory modes of geometric perception: for pose sensing and visual sensing, we used the left and right camera of a stereo vision system mounted on the robot end-effector. This resulted in a redundant system similar to the skilled op-

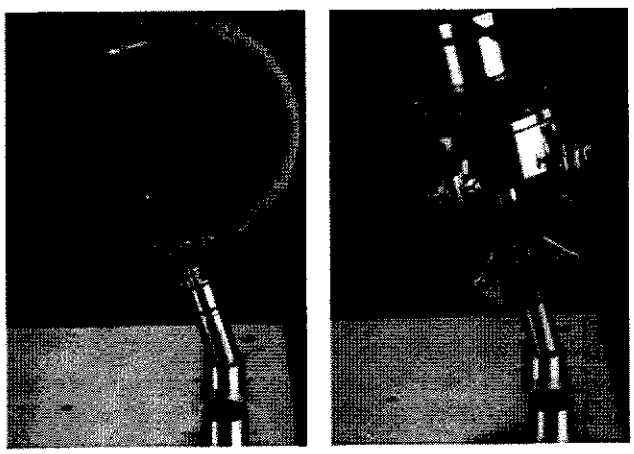


Fig. 44. Demonstration (teaching device) and execution system.

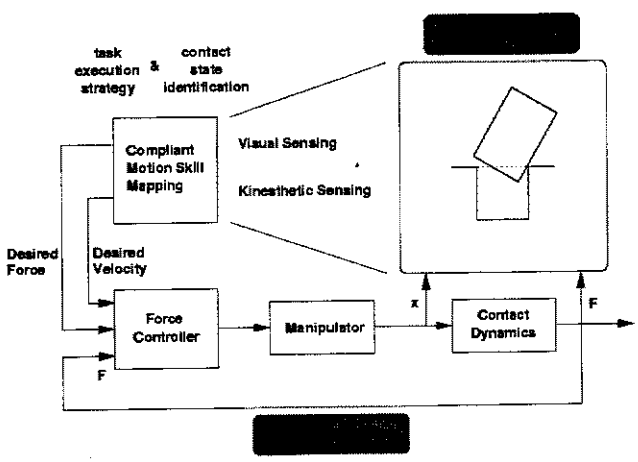


Fig. 45. Sensorimotor compliant motion skill control structure.

Fig. 46. (a) Laparoscopic image out of the abdomen with a color-marked instrument (b) post-processed segmentation

Fig. 47. Minimally invasive surgery using an autonomous robot camera assistant, for the first time tested on humans in September 1996 in the Munich hospital, "Klinikum rechts der Isar"

erator who can perform the task with his or her eyes closed (pure kinesthetic/pose sensing) or with only the left eye, the right eye, or both eyes open. The different sensory modes are optimally combined using sensor fusion techniques (Cortesao and Koeppe 1999).

The technique enables skill transfer for complex tasks comprising attributes like nonlinearity, three-dimensionality, and tight tolerance applicability. We have implemented a peg-in-hole task with 50 μm tolerance for a diameter of 23 mm. This corresponds to an assembly with ISO quality 9. Such tight tolerance may result from processes like fine grinding, milling, or manufacturing on a precision lathe.

9. Conclusion

A couple of years ago, people expected that AI would push robotics quickly forward toward more intelligence; they neglected, e.g., that though logical decisions play an important role, sensory perception and feedback is the real basis for more intelligent behavior. As a matter of fact, industrial robots (representing the broad basis) to a great extent remained as stupid as they were years ago.

And top performances achieved in research labs have not found broad application, as we still are missing kind of commonly accepted standards in robotics research, onto which others might build.

Nevertheless, we believe that the next decade will become the most exciting one in robot history, as the time turns out to be really mature now for robot manufacturers to accept and integrate robot research results on a broad basis. Progress in affordable computational power will result in a breakthrough of real-time 3-D vision, force feedback, and sensor fusion.

Thus, there is justified hope that the big gap, which in the past has separated robot researchers and users, is going to be closed more and more.

References

Baader, A., and Hirzinger, G. 1994. A self-organizing algorithm for multisensory surface reconstruction. Paper presented at Proc. IROS'94 IEEE Int. Conf. on Intelligent Robots and Systems, München.

Brunner, B., Landzettel, K., Schreiber, G., Steinmetz, B. M., and Hirzinger, G. 1999. A universal task-level ground control and programming system for space robot applications. Paper presented at iSAIRAS 5th Int. Symposium on Artificial Intelligence, Robotics and Automation in Space, ESTEC, Noordwijk, June 1-3.

Cortesao, R., and Koeppe, R. 1999. Sensor fusion for skill transfer systems. Paper accepted for presentation at IEEE/RSJ Int. Conf. on Intelligent Robots and Systems, IROS, Kyongju, Korea, October 17-21.

Dickmanns, E. D., and Graefe, V. 1988. Dynamic monocular machine vision. *Machine Vision and Applications* 1:223–240.

Dubowsky, S., and Papadopoulos, E. 1993. The kinematics, dynamics and control of free-flying and free-floating space robotic systems. *IEEE Transactions on robotics and Automation* 5.

Fischer, M., van der Smagt, P., and Hirzinger, G. 1998. Learning techniques in a dataglove based telemanipulation system for the DLR hand. Paper presented at IEEE International Conference on Robotics and Automation (ICRA).

Gombert, B., Hirzinger, G., Plank, G., and Schedl, M. 1994. Modular concepts for a new generation of light weight robots. Paper presented at Prod. IEEE Int. Conf. on Industrial Elektronics, Instrumentation and Control (IECON), Bologna, Italy.

Grunwald, G., and Hager, G. D. 1992. Towards task-directed planning of cooperating sensors. Paper presented at Proc. SPIE OE/Technology: Sensor Fusion V, Conf. 1828, Boston, MA.

Hirzinger, G. Robot-teaching via force-torque-sensors. Proc. EMCSR'82, 6th European Meeting on Cybernetics and Systems Research, Wien, 1982.

Hirzinger, G. Sensor-based space robotics - ROTEX and its telerobotic features. IEEE Transactions on Robotics and Automation, vol. 9, No. 5, Oct. 1993.

Hirzinger, G., Baader, A., Koeppe, R., and Schedl, M. 1993. Towards a new generation of multisensory light-weight robots with learning capabilities. Paper presented at Prod. IFAC 1993 World Congress, Sydney, Australia.

Hirzinger, G., Brunner, B., Dietrich, J., and Heindl, J. 1994. ROTEX—The first remotely controlled robot in space. Paper presented at IEEE Int. Conference on Robotics and Automation, San Diego, California, May 8-13.

Hirzinger, G., Butterfass, J., Knoch, S., and Liu, H. 1997. DLR's multisensory articulated hand. Paper presented at ISER'95 Fifth International Symposium on Experimental Robotics, Barcelona, Catalonia, June 15-18.

Hirzinger, G., and Heindl, J. Device for programming movements of a robot. US-Patent 4,589,810.

Khatib, O. 1987. A unified approach for motion and force control of robot manipulators: The operational space formulation. *IEEE J. Robotics and Automation* 3:43–53.

Koeppe, R., Breidenbach, A., and Hirzinger, G. 1996. Skill representation and acquisition of compliant motions using a teach device. Paper presented at IEEE/RSJ Int. Conf. on Intelligent Robots and Systems, IROS, Osaka, Japan, November.

Koeppe, R., and Hirzinger, G. 1995. Learning compliant motions by task-demonstration in virtual environments. Paper presented at Fourth Int. Symp. on Experimental Robotics, ISER, Stanford, June 30-July 2.

Koeppe, R., and Hirzinger, G. 1999. Sensorimotor compliant motion from geometric perception. Paper accepted for presentation at IEEE/RSJ Int. Conf. on Intelligent Robots and Systems, IROS, Kyongju, Korea, October 17-21.