Technical Report SFB360–TR–96–4

The Neuroinformatics Robot Laboratory

Jörg Walter and **Helge Ritter**

September 26, 1996

SFB-360 · Project D4 · Introductory Report of the
Arbeitsgruppe Neuroinformatik
Technische Fakultät
Universität Bielefeld
D–33615 Bielefeld

Please send any comments, remarks or feedback via
Email: walter@techfak.uni-bielefeld.de
Tel: 0521-106-6064 · Fax: +49-521-106-6011
Further information URL
http://www.techfak.uni-bielefeld.de/∼walter/

Abstract:

We describe the developed hardware concepts and set-up of the Neuroinformatics robot laboratory. It provides part of the infrastructure for the SFB-360 project D4 “Multisensor Based Exploration and Manipulation” (Additional facilities for this project are contributed by the working group of Prof. Alois Knoll and will be described elsewhere).

Central part of the described hardware infrastructure is a classical 6 degrees-of-freedom (DOF) robot manipulator and a multi-fingered hydraulic TUM robot hand. The robot's proprioceptive sensors and their limitations are described. Several additional sensory systems became available, among them are a 6 D force-torque wrist sensor, the developed prototype of a new tactile sensor sub-system, video imaging systems, and active camera systems.

The current communication infrastructure is presented with regards to the physical transport layer, providing the base for various kinds of control and communication processes (asymmetric multiprocessing). The developed high-level software integration layer for the special needs of our robotics laboratory is reported in SFB360-TR-96-3 “SORMA” (Walter and Ritter 1996).

Contents

1 Introduction

This report describes the developed concept and set-up of our robotic laboratory. It is aimed at the technically interested reader and explains some of the hardware aspects of this work. The construction and working of system components shall be described as well as ideas, difficulties and solutions which accompanied the development.

The domain for setting up this robotics laboratory is the domain of manipulation and exploration with a 6 degrees-of-freedom (DOF) *robot manipulator* in conjunction with a *multi-fingered robot hand*. Before presenting the details of our achieved approach, we want to consider some issues and characteristics of an actuator in general. What are the kinds of control goals? What kinds of sensors are of interest?

1.1 Actuator Issues

What are the main issues for the actuators of interest? They comprise a particular mechanical structure and systems to generate and transmit mechanical energy.

The geometric sizing was oriented at a human, manipulating objects without extra tools (up to 1 kg, in contrast to micro-manipulation or lifting heavy automobile parts.) For cost efficiency reasons, preferred materials of the mechanical structure are various kinds of metals (steel and aluminum, in contrast, e.g., to silicon used for nano-structures or carbon structure for light-weight applications.) The stiff links are connected by revolving or prismatic joints and actuated by some type of motor. Here the possible solutions of actuation systems spread widely. The goals are simple, but conflicting: fast, strong, power-efficient, and light-weight. Low motor weight is increasingly important with growing distance to the resting robot torso, since the inertial torque load for - in the link chain - previous joint motors is quadratic in distance.

Various system are used for transmitting the mechanical energy, generated in most cases by electrical motors. A common design are *gear trains* converting the high rotational speeds to high torques at the joints. The motor is mounted here close to the torso or, in heavier, low-friction "direct-drive" motors, at the joint shaft. One interesting axis of research is the development of brush-less motor systems with extremely high power to weight ratio (Hirzinger, Dietrich, and Brunner 1990). The other main direction is the spatial separation of the actuation system in a heavy and maybe bulky mechanical power generation part and transmission to the joints. Prominent are transmission via tendons and oil hydraulics.

Analog to the biological muscle actuators, two *tendons* are pulling as agonist and antagonist at a pulley exerting joint torques according to the difference in pulling force. The advantage is very high actuation speed by high power weight ratio. For an increasing number of links the tendon guiding structure gets complicated to accommodate in a compact and stiff construction. For example, the Utah-MIT hand is rather fast and strong, but restricted by its bulky link to the remote motor box (Jacobsen 1984).

Hydraulic systems are prominent for heavy duty machinery due to their excellent power density and ease of delivering centrally generated mechanical energy (also preferred in explosion endangered areas.) Usually, these systems are built with double side powered cylinders where valves control the power supplied by a compressor. Alternatively, the pressured oil drives directly revolving motors.
More exotic actuator systems, for example built with heated memory alloys, we didn' t consider.

1.2 What to control and command?

Obviously, an actuator system must be commanded somehow. At the most basic level *motor power signals* are commanded (e.g. on/off, direction, voltage or puls-width coded torque signals). Using this level is rather tedious, since it usually does not reflect any *intention*.

More convenient is the notion of moving to a certain location, or, more general, a goal *pose* (position and orientation) – within a certain time. Pose commands can be desired in various coordinate systems: joint angles or Cartesian (different frames of reference, e.g., world and tool). Pose control involves the inverse kinematics and suitable trajectory generation from the current to the defined goal pose. Further it involves proprioceptive *position sensors* (e.g. joint angle encoders) and a *control loop* to gain appropriate motor power commands.

Today, most industrial robots on the shop floors are working solely with this paradigm: absolute pose control with high repetition precision and speed. This is based on high resolution sensors for the absolute robot position and works successfully in highly structured environments like precisely defined work cells.

The next advanced level of commands specify desired action in terms of measurable patterns, which are more complex. These robot actions aim at manipulation and exploration of objects. Therefore, the pose specification are extended from absolute to relative commands – relative to the object of interest. The second important extension considers forces and torques. Exerted forces manipulate the object immediately and must be controlled for articulated objects interaction[1].

Complex robot tasks are decomposed in *elementary motion primitives* where

[1]In industrial manufacturing the urge for general active force control is often avoided by artificially structuring, reproducible environments in combination with specially configured, passive, compliant mount devices.

we specify *nominal sensory patterns* as target functions. In the context of force / torque control patterns this problem is know as the *compliant motion* problem. Compliant robot motions are non-fixed trajectory specifications. The intention encoding allows a family of possible trajectories – the actually executed one is depended on the sensory input. For example, one commands a specific force or torque in order to grasp, push an object or turn a screw. How far the robot will move, is controlled by the sensed force. This form of control is often selected in certain control sub-spaces and combined with position / velocity control along orthogonal axes.

The goal can also be denoted as a general target functions on combined sensory patterns: e.g. *active stiffness* or impedance control: “Move to a position and behave like a spring”. A spring stretches in a linear - or non-linear - fashion depending on the exerted forces. This is a natural compromise for exerting a desired force without drifting unbounded far away from the desired location.

The sensory target functions and patterns can be generalized to a pattern on a set of *virtual sensors*. A virtual Sensor can be the classical absolute pose sensors (based on joint enscoders), a wrist force-sensors, any kind of advanced, direct or derived sensory information device (see below).

Additional specification of the elementary motion primitives concern start and stopping criteria in form of sensory pattern templates. This is required to: *(i)* specify prerequisites (e.g. force comply motions requires an available sensor); *(ii)* to specify termination conditions (enter the next task execution phase); *(iii)* safeguard the motion (*“guarded motion”*) and trigger a reaction when, e.g., specified force (or position) thresholds are exceeded. For example, induce a reflex on collision or start the next motion when contact is reached (branch condition).

1.3 Sensor Taxonomy

A very important prerequisite for these advanced autonomous robot action primitives are suitable sensors. Sensors are often distinguished between *proprioceptors* and *exteroceptors*.

(i) **Proprioceptors** sense *internal state* variables. They are the fundament of most realized control loops. In particular, joint position sensors must have high resolution to allow fine pose control of robot motions. Joint torque sensors can be used to close a desired force control loop around disturbances, thus achieving better control performance. Further, internal sanity checks, e.g., (over-) temperature sensors, communication checksumming, etc., can be counted as proprioceptive sensory information. They helps assuring performance and signal possible malfunction (“pain”) and need for extra attention from the operator.

(ii) **Exteroceptor** are sensitive to environmental signals from the outside, which includes mechanical, thermal, optical, acoustical, and olfactory stimuli. The notion

of proprio- versus exteroceptor originates in neuroanatomy. Certainly, also exteroceptor can measure only the internal physical effect caused by an immediate or remote environment interaction (via touch, light, sound). In contrast to the proprioceptors, the exteroceptive sensor devices are built such, that they are selectively sensitive to the desired information – directed towards the outside. Due to the physical inside - outside coupling, this distinction remains somehow vague and leads to different classifications in the robotics literature[2]. For a more practical solution, we suggest here the notion of a $(ii.i)$ *contact oriented sensors* for the immediate object interaction. During exploration and manipulation, the tactile sense plays a key role. There importance results intuitively when considering how the human hand can perform innumerable tasks effectively.
$(ii.ii)$ *Contact-less, remote* sensors gather information from the distance. In particular, relative distance values are of great interest. Proximity can be detected by laser triangulation or time of flight measurement using ultra sound or laser light pulses. Optical imaging by mono- or binocular camera systems, are useful for various kinds of tasks, in particular, object identification and object localization.

The abstract concepts of possible action and perception systems were introduced in the previous paragraphs. Fig. 1 introduces the realization: it displays the actuator components robot arm and the three-fingered hand. Among the perception components only the wrist sensor and the end-effector based camera are apparent.

Last not least, these components must be connected in some suitable manner - on a hardware as well as on a software level. These connections should by efficient, easy to maintain and control from a higher planning level comprising the cognitive component.

The integration of these non-standard, highly complex components into robust and flexible modules bears a number of interesting issues (see also Walter and Ritter 1996). The next sections are devoted to the description of the major lab components, some of the problems and solution are reported.

2 Actuation: The Puma Robot

The manipulator should serve a a multi-purpose general 6 DOF positioning device, as already pointed out before, able to carry a multi-fingered robot hand plus a pay load of about one kg. It should be a robust, “open”, “real-time” system.

An **open system** means, that sufficient and precise information is provided, telling how things are done. Additional to a good external system interface speci-

[2]E.g., touch sensors are considered as exteroceptor in (Vassura and Bicchi 1989) and proprioceptive in (Koeppe and Hirzinger 1995). Section 4.2 discusses the exteroceptive information gain of a proprioceptive force/torque wrist sensor device.

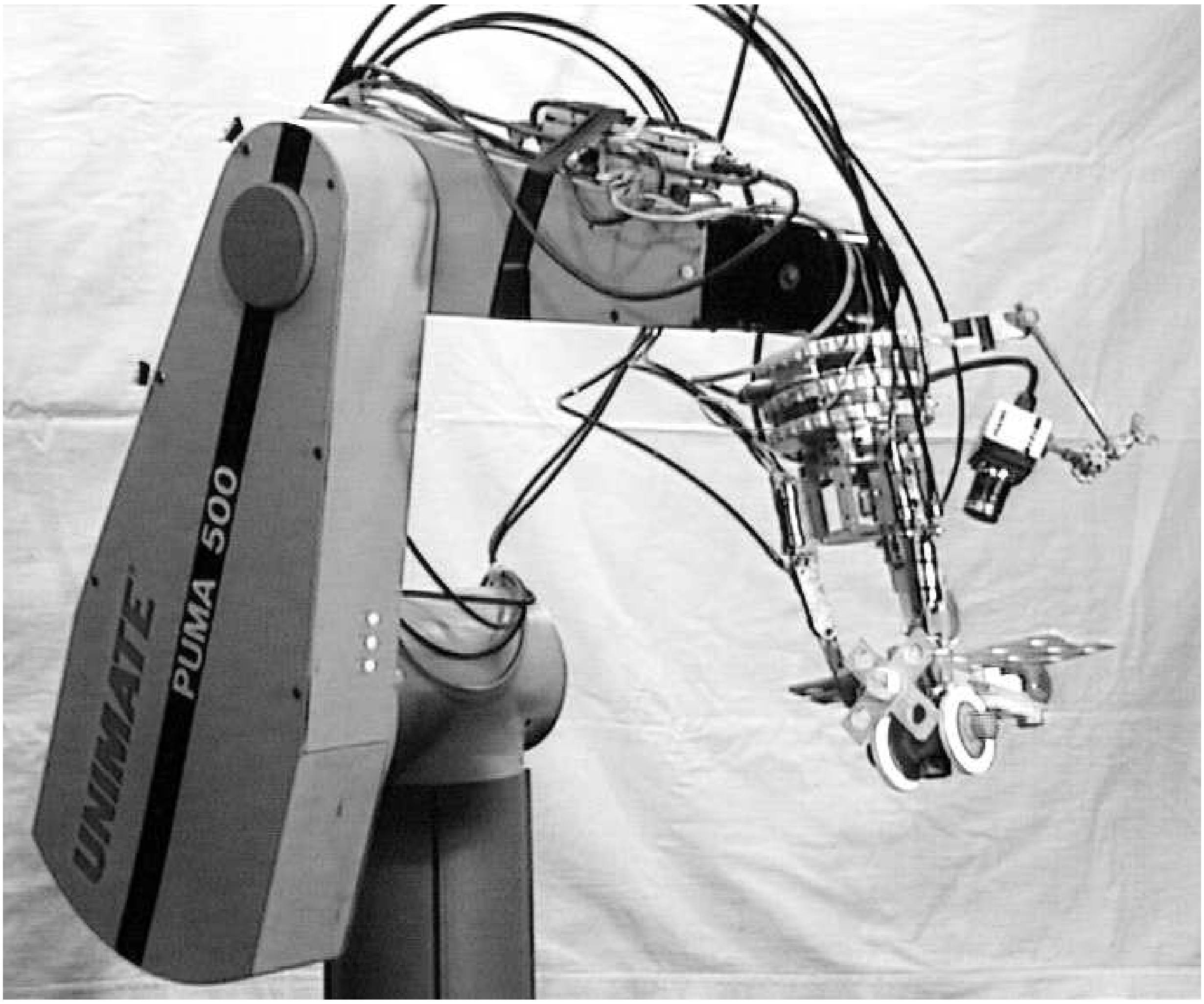


Figure 1: The six axes Puma robot arm with the TUM multi-fingered hand fixating a wooden “Baufix” toy airplane. The 6 D force-torque sensor (FTS) and the end-effector mounted camera is visible, in contrast to built-in proprioceptive joint encoders.

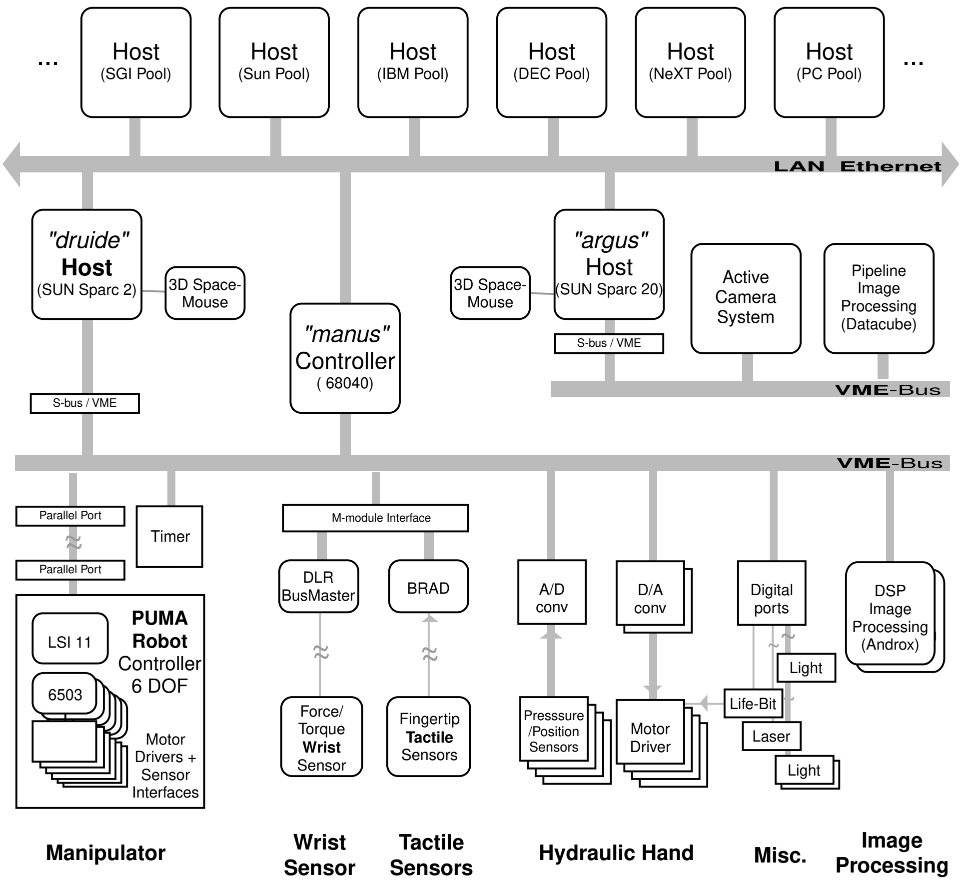


Figure 2: The main hardware "roads" connect the heterogeneous system components and lay ground for various types of communication links. The LAN Ethernet ("Local Area Network" with TCP/IP and max. throughput 10 Mbit/s) connects the pool of Unix computer workstations with the primary "robotics host" *"druide"* and the "active vision host" *"argus"* . Each of the two Unix SparcStation is bus master to a VME-bus (max 20 MByte/s, with 4 MByte/s S-bus link). *"argus"* controls the active stereo vision platform and the image processing system (Datacube, in pipeline architecture). *"druide"* is the primary host, which controls the robot manipulator, the robot hand, the perception systems including the force/torque wrist sensor, the tactile sensors, and the second image processing system. The hand sub-system electronics is coordinated by the *"manus"* controller, which is a second VME bus master and also accessible via the Ethernet link. Boxes with rounded corners indicate semi-autonomous sub-systems with CPUs enclosed. For more details, see text.

fication, also the documentations to internal interfaces are accessible. As a result, it is open to inspection of internal affairs, as well as additions and replacements of particular components – hardware as well as software. As conflicting this openness is to the economy inherent idea of sustainable profit making, as essential it is to incremental progress in science and technology.

This dilemma is rather fundamental, but it gets very apparent when looking for a mechanically and electrically mature robot. As already indicated, todays industrial robots rarely use advanced sensory equipment and they are not open systems. Larger and solid research machines (like e.g., the Sarcos arm) are rather budget unfriendly and come with an open system interface, but only on a low joint motor control level.
We find that the lack of a common, high level, mature, and open robot systems in combination with the non-disclosure attitude of the industrial manufactures is a substantial hurdle for incremental research in robotics.

The compromise solution was found with a Puma 560 Mark II robot. It is probably "the" classical industrial robots with six revolve joints. Its geometry and kinematics[3] is subject of standard robotics textbooks (Paul 1981; Fu, Gonzalez, and Lee 1987). It can be characterized as a medium fast (0.5 m/s straight line), very reliable, robust "work horse" for medium pay loads. The action radius is comparable to the human arm, but the arm is stronger and heavier (3 kg permanently in any position; max. radius 0.9 m; 63 kg arm weight). The Puma Mark II controller comprises the power supply and the servo electronics for the six DC motors. They are controlled by six parallel microprocessors and coordinated by a DEC LSI-11 as central controller. Each joint microprocessor (Rockwell 6503) implements a digital PD controller, correcting the commanded joint position periodically. The *decoupled joint position control* operates with 1 kHz and originally receives command updates (setpoints) every 28 ms by the LSI-11.

The typical industrial applications for a robots of this type are for example, welding, part manipulations and mating (assembly), and spray painting. Here, the desired trajectory is programmed by recording a sequence of via-points shown by the instructor. Or, nowadays more and more important, by direct computation from CAD-model data of the scene. In the production phase, the goal trajectory is repeated with high precision and no fatigue. The control process checks consistency of certain state signals (e.g., "part in place", "ready for take away" or "force overload") and can react by program branching (e.g., "waiting for part", or emergency stop).

In the standard application the Puma is programmed in the interpreted lan-

[3] Designed by Joe Engelberger, the founder of Unimation, sometimes called the father of robotics. Unimation was later sold to Westinghouse Inc., AEG and last to Stäubli.

guage VAL II, which is considered a flexible programming language by industrial standards. But running on the main controller (LSI-11 processor), it is not capable of handling high bandwidth sensory input itself (e.g., from a video camera) and furthermore, it does not support flexible control by an auxiliary computer. To achieve a tight real-time control directly by a Unix workstation, we installed the software package RCI/RCCL (Hayward and Paul 1986; Lloyd 1988; Lloyd and Parker 1990; Lloyd and Hayward 1992). With the help of the main author, John Lloyd, we gained a reliable and open robot system including all source code and an exemplary documentation.

The acronym RCI/RCCL stands for *Real-time Control Interface* and *Robot Control C Library*. The package provides the following valuable components:

Reprogramming the robot controller: By reverse engineering the joint micro controller code, the internal control flow could be redirected via the added high speed parallel link to the external host computer. At power up time the LSI controller is reprogrammed via the PROM boot monitor and a terminal emulation program. Instead of running VAL it serves as a command dispatcher to the joint micros. The LSI collects feedback data form them, performs elementary safety checks, and handles the communication to the new host via a high speed parallel link.

Robot control in C: RCCL allows to issue robot motion requests from a high level control program ("planning task") which is written and executed as an ordinary C program on a Sun SparcStation 2. By shared memory communication (RCI) these requests are handed to the trajectory control level. The control task is executed periodically at the highest priority and is responsible for reading feedback data, generating intermediate joint setpoints and sending them to the robot controller.

Real-time Control under Unix: We patched the Sun operating system OS 4.1[4]to sufficient real-time capabilities for serving a reliable control process up to about 200 Hz. **Unix** is a multitasking operating system, sequencing several processes in short time slices. Initially, Unix was not designed for real-time control, therefore it provides a regular process only with timing control on a coarse time scale (e.g. low priority alarm interrupts). But **real-time** processing requires, that the system reliably responds within a certain time frame. RCI succeeded here by anchoring the control task at a special device driver

[4]RCI was originally developed for a shared memory multi-processor system. The single processor port to SunOS 4.0 was done in conjunction with the PUMA-Sparc installation in the Beckman Institute at UIUC (Walter 1991). Since RCCL is also capable of controlling several real or simulated robots simultaneously, this robot system architecture was successfully duplicated in the robotics laboratory of Prof. Knoll, serving there for the cooperation of a pair of PUMA 262 robots.

serving the interrupts from a timer card. The **control task** is thus running independently and outside the **planning task**. RCI handles all the set-up and proper shutdown of the required interrupts. The time critical code segments get memory locked, in order to avoid time consuming disk swapping.

Trajectory requests: RCCL provides a variety of ways to specify a desired trajectory. Besides specifying goal positions by joint angle tuples, they can be directly written as a kinematic chain of homogeneous coordinate transformations:

$$\mathbf{T}_{basis}\,\mathbf{T}_6\,\mathbf{T}_{tool} = \mathbf{T}_{table}\,\mathbf{T}_{object}\,\mathbf{T}_{graspPosition} \qquad (\mathbf{T}_i \in \mathbb{R}^4 \times \mathbb{R}^4). \quad (1)$$

$\mathbf{T}_6$ denotes the desired transformation matrix describing the space rotation and translation from the robot shoulder to the wrist. The robot path is interpolated from the current position to the new position linearly either in joint or in Cartesian space. Honoring desired velocity, acceleration, and/or time constraints, the trajectory is formed by fifth order spline polynomials.

Trajectory real-time control: Two features facilitate convinient ways ofreal-time robot control: *(i)* Extra application functions can be registered to be executed in each control cycle, for example, for carrying out watch-dog functions . *(ii)* Sensory feedback data can be used to modify the goal position definition during motion execution. For example, $\mathbf{T}_{object}$ in Eq. 1 can be periodically updated by camera or force feedback, see also (Walter and Ritter 1996)..

The resulting robot control system allows us to implement hybrid control architectures using the position control interface. This includes multi-sensor compliant motions with mixed force controlled motions as well as controlling an artificial spring behavior (impedance control). The main restriction is the difficulty in controlling forces with high robot speeds. High speed motions with environment interaction need quick response and therefore requires, a very high frequency of the digital force control loop. The bottleneck is given by the Puma controller structure. The realizable force control includes a fast inner position loop (joint micro controller) with a slower outer force loop (involving the Sun *"druide"*). But still, by generating the robot trajectory setpoints on the external Sun workstation, we could double the control frequency of VAL II and establish a stable outer control loop with 65 Hz.

Fig. 3 sketches the two-loop control scheme implemented for the mixed force and position control of the Puma. The inner, fast loop runs on the joint micro controller within the Puma controller, the outer loop involves the control task on

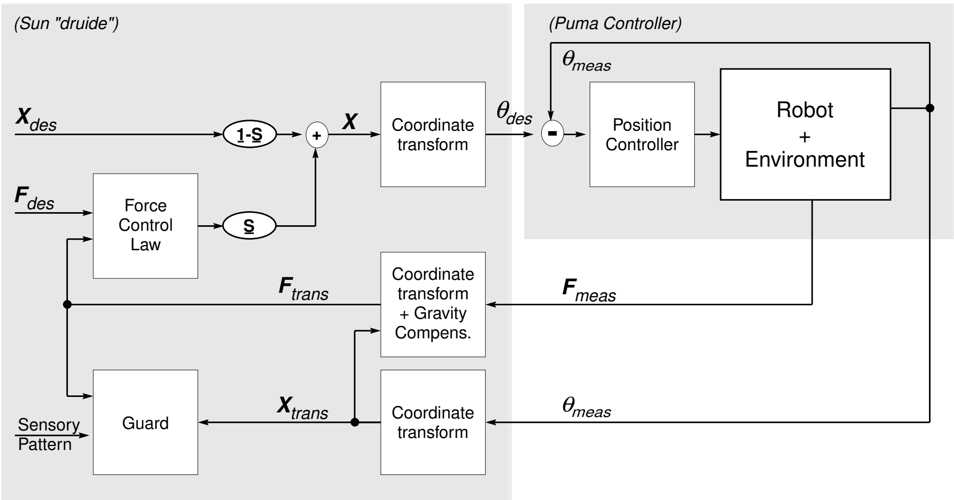


Figure 3: A two-loop control scheme for the mixed force and position control. The inner, fast loop runs on the joint micro controller within the Puma controller, the outer loop involves the control task on *"druide"*.

"druide". The desired position $\mathbf{X}_{des}$ and forces $\mathbf{F}_{des}$ are given for a specified coordinate system (here written as generalized 6 D vectors: position and orientation in roll, pitch, yaw (see also Walter (1996) and Paul 1981) $\mathbf{X}_{des} = (p_x, p_y, p_z, \phi, \theta, \psi)$ and generalized force $\mathbf{F}_{des} = (f_x, f_y, f_z, m_x, m_y, m_z)$). The control law transforms the force deviation into a desired position. The diagonal selection matrix elements in **S** choose force controls (if 1) or position control (if 0) for each axis, following the idea of Cartesian sub-spaces control[5]. The desired position is transformed and signaled to the joint controllers, which determine appropriate motor power commands. The results of the robot - environment interaction $\mathbf{F}_{meas}$ is monitored by the force-torque sensor measurement and transformed after the gravity force computation to the net acting force $\mathbf{F}_{trans}$, see Sect. 4.2. The guard block checks on specified sensory patterns, e.g., force-torque ranges for each axes and whether the robot is within a safe-marked work space volume. Walter and Ritter (1996) describes the integration of supplementary sensors.

Depending on the desired action, one must choose the suitable controller scheme and entire sets of parameters, for example, **S**, gains, stiffness, safe force/position

[5]Examples for suitable selection matrices are: **S**=diag(0,0,1,0,0,0) for a compliant motion with a desired force in z direction, or b S=diag(0,0,1,1,1,0) for aligning two flat surfaces (with surface normal in z). A free translation and z-rotational follow controller in Cartesian space can be realized with **S**=diag(1,1,1,0,0,1). See (Mason and Salisbury 1985; Schutter 1986; Dücker 1995).

patterns). The efficient parameterization is subject to work of Christof Dücker. Here the naming of actions, parameter sets (and sub-sets) turned out to be a very useful mechanism as described in (Walter and Ritter 1996).

3 Actuation: "Manus" – The TUM-Hand

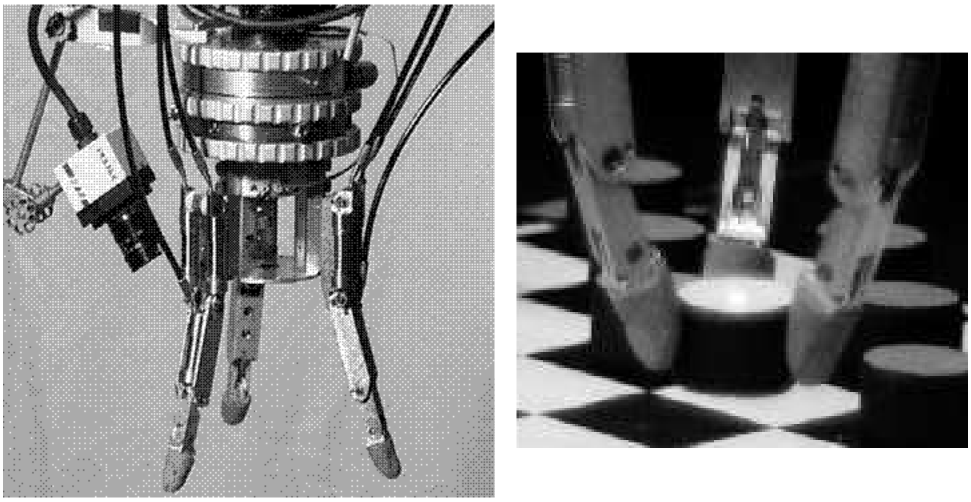
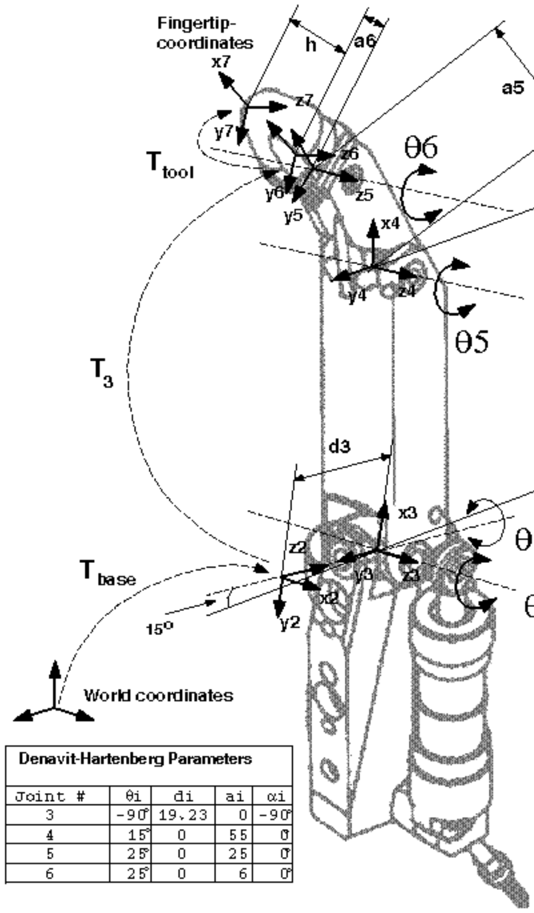


Figure 4: The end-effector. Between the arm and the hydraulic hand, the cylinder shaped device can measure current 6 D force torque values. The three finger modules are mounted here symmetrically at the 12 sided regular prism base. On the right side, the color video camera looks at the scene from an end-effector fixed position. Inside the flat palm, a diode laser is directed in tool axis, which allows depth triangulation in the viewing angle of the camera.

For the purpose of studying dextrous manipulations tasks, our robot lab is equipped with an hydraulic robot hand with (up to) four identical 3-DOF fingers modules, see Fig. 4. The hand prototype was developed and built by the mechanical engineering group of Prof. Pfeiffer at the Technical University of Munich ("TUM-hand"). We received the final hand prototype comprising four completely actuated fingers, the sensor interface, and motor driver electronics. The robot finger's design and its mobility resembles that of the human index finger, but scaled up to about 110 %.

Fig. 5 displays the kinematics of one finger. The particular kinematic mapping (from piston location to joint angles and Cartesian position) of the cardanic joint configuration is hard to invert analytically. Selle (1995) describes an iterative numerical procedure. A learning approach to obain these mappings is discussed in (Walter and Ritter 1995).



Denavit-Hartenberg Parameters

Joint #	θi	di	ai	αi
3	-90°	19.23	0	-90°
4	15°	0	55	0°
5	25°	0	25	0°
6	25°	0	6	0°

Figure 5: The kinematics of the TUM robot finger. The cardanic base joint allows 15° sidewards gyring (θ_3) and full adduction (θ_4) together with two coupled joints ($\theta_5 = \theta_6$).

3.1 Oil model

The finger joints are driven by small, spring loaded, hydraulic cylinders connecting each actuator to the base station by a single oil hose. In contrast to the more standard hydraulic system with a central power supply and valve controlled bi-directional powered cylinder, here, each finger cylinder is one-way powered from a corresponding cylinder at the base station. Unfortunately, the finger design does not foresee integrated sensors directly at the fingers.

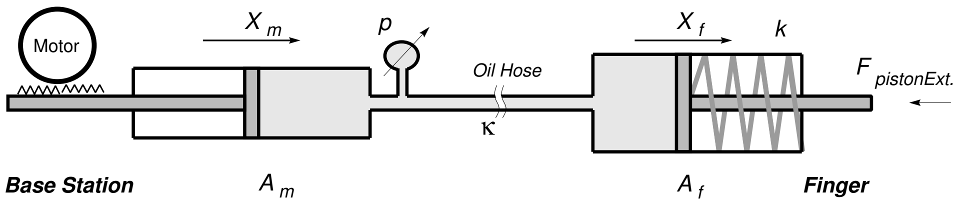


Figure 6: The hydraulic oil system.

The control system has to rely on indirect feedback sensing through the oil system. Fig. 6 displays the location of the two feedback sensors. In each degree of freedom (i) the piston position x_m of the motor cylinder (linear potentiometer) and (ii) the pressure p in the closed oil system (membrane sensor with semi-conductor strain-gauge) is measured at the base station. The long oil hose is not perfectly stiff, which makes this oil system component significantly expandable (4 m, large surface to volume ratio). This bears the advantage of a naturally compliant and damped system but bears also the disadvantage, that even pure position control must consider the force - position coupled oil model (Menzel et al. 1993; Rankers 1994; Selle 1995). The idealized equations are:

$$A_m \, x_m = A_f \, x_f + \kappa \, p \tag{2}$$

$$p \, A_m = F_{pistonExtern} + k_{spring} \, x_f + F_{friction} \tag{3}$$

Eq. 2 describes a closed oil system with overall linear compressibility κ without leakage. Here, A_f, A_m, x_f, x_m denote the piston area A and relative cylinder piston position x on the **f**inger and the **m**otor side. Eq. 3 denotes the force balance at the spring loaded finger cylinder. The last term $F_{friction}$ aggravates the coupling. It summarizes the friction hysteresis effect depending on the piston velocity, the pressure on the sealing, and the local surface roughness in the cylinder.

3.2 Hardware and Software Integration

The modular concept of the TUM-hand encloses its interface electronics. Each finger module has its separate electronics rack with motor and sensor amplifiers, which we connected to analog converter cards in the VME bus system as illustrated in the lower right part of Fig. 2. The digital hand control process is running at manus, a VME based 68040 processor board (Microsys CPU 40). *"manus"* runs with the multi-tasking real-time operating system PSOS+, allowing remote compiling and debugging on the Sun host system (*"druide"*) via the additional Ethernet link. Surprisingly, the TCP/IP and RPC library functions turned out to be insufficiently implemented and do not peacefully coexist with a time critical control task.

Following the example of RCCL, the "Manus Control C Library" (MCCL) was developed and implemented by Rankers (1994) and Selle (1995). To facilitate a arm–hand unified planning level, the Unix workstation *"druide"* is set up to issue finger motion (piston, joint, or Cartesian position), and force control requests to the *"manus"* controller (see Fig. 2, page 9). The communication uses messaging via shared memory located at *"manus"*. Trajectories based on third order polynomials are generated by the *"manus"* controller in the slower outer loop (ca. 50 Hz). Fig. 7 shows the inner loops of the impedance control scheme, executing with 200 Hz. The desired finger piston position $\mathbf{X}_{\mathbf{f},\mathrm{des}}$ and forces $\mathbf{F}_{\mathbf{f},\mathrm{des}}$ are followed by a digital PD controller in the inner loop. The oil model is employed to estimate the state of the remote finger and supplies the feedback control law. The desired elasticity law $\mathbf{K}^{-1}$ transforms force deviations into commensurable position deviations. For pure force control the position correction branch can be switched off (which may lead to drift due to incorrect force state estimations; see also Paetsch (1993)). Further direct fingertip sensory information should improve the finger state estimation, see below.

The TUM hand is controlled at the *motor power signal* level. Operating at this level requires to take **safety** provisions. Despite the fact that the hydraulic actuators are good-natured, naturally damped systems, which are not dangerous for their surroundings (in contrast to the Puma arm) care must be taken about the following system immanent dangers: *(i)* The membrane of the semi-conductor pressure sensor is the weakest part of the hydraulic system. Bursting is prevented by keeping a maximal pressure of safe 50 bar. *(ii)* The cylinder piston sealing is an inward formed lip and tightens under oil pressure. Below atmospheric pressure, air may break into the oil system, forming a partly solved mixture (one cubic-cm (ccm) oil may solve up to three ccm air). Negative relative pressure can occur, when the motor cylinder is retracted too fast (or too far), or the finger is closed by external force. This can be prevented by keeping a minimal pressure of about

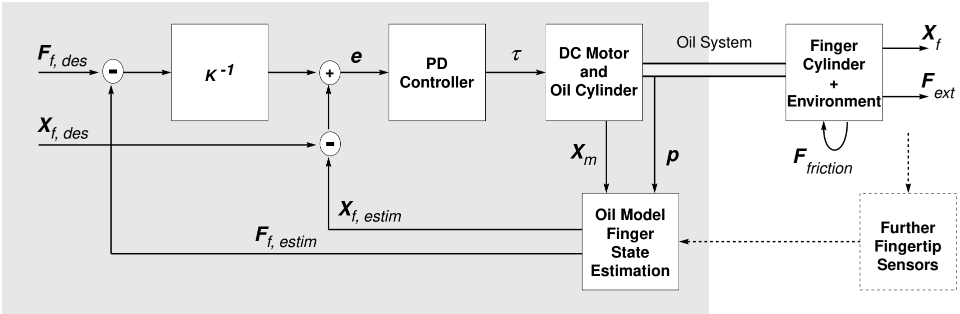


Figure 7: A two-loop control scheme for the mixed force and position control. The inner, fast loop runs on the joint micro controller within the Puma controller, the outer loop involves the control task on *"druide"*.

3 bar. The price is a reduced finger opening velocity (the opening force of the finger cylinder spring is partly compensated). (iii) A further safety measure is concerned about active motor power commands in case the controller software hangs. We implemented a power enable circuit expecting a software generated periodical toggling of a so-called "life-bit" (see Fig. 2). Analog to a dead-man-switch in a train cockpit, the power will be cut if the controller software process does not appear to be alive.

3.3 Problems with Fine Control

The achieved control performance was not satisfying which lead to a more detailed study of the oil system. Selle (1995) wrote a detailed simulator for studying the control system emphasizing on the influence of the major uncertainties and their propagation through the control system. From there, the calibration process could be refined and parameters identified, which describe the dominant friction effects (Coulomb and Stokes) in detail.

Two main sources of uncertainty from hysteresis are within the control loop. *(i)* One source of position uncertainty is the backlash in the four joint axes. In the domain of low contact forces, it spans up to 4 mm tolerance in fingertip position. *(ii)* The other source is the unknown but significant friction forces in the *finger* cylinders. The described position-force coupling in the actual oil system propagates the uncertainties into the control loop. As depicted in Fig. 8, the effect on the estimated finger force estimation is dependent on the joint configuration (Jacobian matrix).

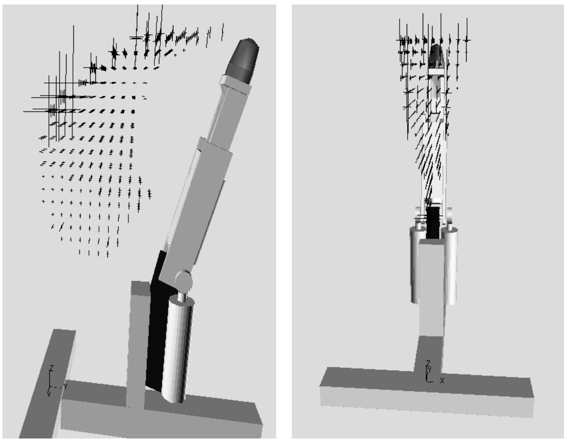


Figure 8: Errors of the external force computation due to bad friction hysteresis estimation are dependent on the joint configuration. The error bars are drawn at the finger tip scaled to 20 N w.r.t. the xyz axes cross. This graphical animation tool package is able to visualizing simulated as well as live data from the hand.

The friction hysteresis effect is often mathematically formulated[6] as a $F_{friction} = \mathsf{sgn}(\dot{x}_f)(\gamma_1 + \gamma_2 p)$ term, but this is not always correct. In case the finger piston is steady, the term $\mathsf{sgn}(\dot{x}_f) = \mathsf{sgn}(0)$ is zero, but the force $F_{friction}$ is – in general – not. Then, the physically transmitted friction force is apriori *unknown* within the sticktion force bounds. The sticktion forces can be significantly higher than the friction forces.

3.4 Strategies Against Friction Hysteresis

One way of dealing with this friction effect is to *estimate* the friction force and *compensate* it. This could be learned (including the piston position depended cylinder surface properties) and combined with the idea of signal-preshaping (de Wit, Noel, Aubin, and Brogliato 1991; Hyde and Cutkosky 1993). A pre-requisite is a suitable sensation to base this estimation of the friction state. For the reasons explained above, piston velocity monitoring turns out to be insufficient. The achievable control loop is instable, since the friction estimation is unreliable as soon as the finger comes to rest.

A principal alternative is continuously **rocking the finger**. By adding a sinusoidal bias term to the motor power signal, the static friction (sticktion) at the finger actuator can be periodically overcome (Selle 1995). This concept indeed helps to improve the force and active stiffness control, but is bears some disadvantages: *(i)* The disturbance propagates through the whole oil system and therefore

[6]E.g., (Menzel, Woelfl, and Pfeiffer 1993; de Wit, Noel, Aubin, and Brogliato 1991)

interferes with the other sensors. Principally, high frequency injection is desired, but high frequencies are damped too much and lead to more energy dissipation and fatigue of the motor-gear train. *(ii)* The motor piston oscillation amplitude must be such, that the sticking forces in the (4 m far away) finger piston are exceeded, and - at the same time - it should be of minimal dose, in order not to endanger any desired fingertip sticktion in a grasp. *(iii)* Constantly injecting mechanical energy heats the oil system due to dissipation (friction, oil viscosity, and a small but measurable, non-elastic expansion of the oil hose). Thermal expansion – as any oil volume changes – directly affects the finger state estimation (Eq. 2), but can not be measured with the sensory equipment[7], currently at hand. These undesired consequences make finger-rocking only a temporary solution.

3.5 TUM Hand: Open Questions

The oil system actuation system is characterized by a high (closing) strength and good power to weight ratio. The spring driven opening motion denies forceful pushing with the finger backs. Despite many efforts on mechanical precision of the cylinder surface and refined sealing, the oil system's leakage problem is not entirely solved yet.

The overall ability for fine manipulation is still not very satisfying. Experiments show force measurement uncertainties up to several Newton, which are insufficient for reliable fine and micro motion control. For example, grasping a fragilely standing object, obviously requires good force synchronization in order to avoid premature dislocation. What should be improved? The primary gap is the *lack of proprioceptive sensory equipment*. The sensing of force and position through the coupling oil system makes position as well as force control strategies susceptible to all occuring oil system uncertainties.

Where could further sensor system help resolve these ambiguities? The first places for feedback sensors are directly at the finger joints and links. The compactness of the given mechanical design makes insertion of suitable *joint angle encoders* rather difficult. Similarly difficult is the measurement of link stress by deformations (strain-gauges) since the structure is very stiff.

Add-on sensors are useful at the fingertip and generally on all surfaces. The next section will discuss 6 D force-torque sensing and report on the on-going development of localized surface sensors for the fingertips.

Those are very promising, since good localized force state sensing concurrently helps to solve, two other problems of the TUM-hand. Any auxiliary strain on (or stress between) finger links disturbs the fingertip force and (coupled) position mea-

[7] Indirect measurement of the volume expansion by temperature monitoring is practically ruled out by the large surface to oil volume ratio.

surement (through the oil system). Therefore, independent force feedback allows *(i) better disturbance rejection* for forces and torques transmitted by the middle oil hose and *(ii)* the use of a thick *finger glove for the metallic phalanxes*.

The shape and surface smoothness of phalanxes play a key role when controlled rolling/slipping of objects inside the hand is desired. As we can learn from our own hands round shaped cross section of the finger allow to easily move the contact area on each link[8]. The surface between adjacent links should allow to move and enlarge the contact area from link to link. Currently, the metallic phalanxes as well as the area inbetween the adjacent links are of rectangular cross section.

A further question is, how to design an articulated palm which could add the missing "thumb" degrees of freedom? The human thumb can rotate normal to the palm in order to get in opposition to each of the other fingers - with variable basis distance. This is used for *power grasps* on differently shaped and sized objects (ball, handle, thin flat, thick flat, etc. See Cutkosky and Howe (1990) for a taxonomy of static grasps). The thumb's excursion reaches also the *lateral pinch grasp*, the typical key turn pose, which is preferred for exerting high torques on flat objects[9]. Adding two further degrees of freedom to, at least at one finger, would greatly enhance the dexterity of the hand manipulations. This question opens the very exciting topic on articulated hand design. However, following all these issues in detail, would be beyond the scope of this work (for further reading, consult (Mason and Salisbury 1985; Vassura and Bicchi 1989; Venkataraman and Iberall 1990)).

4 Contact Perception: 6 D Force-Torque Sensing

The next two section are devoted to non-standard sensory equipment, which are essential to receive information about the contacts involved in possible object and environment interactions: *(i)* the principles of the 6 D Force-Torque sensing are discussed. The sensor device is inserted the robot and "feels" the currently acting stress in order to determine the contact situation at the surface of the robot geometry. Particularly applicable is this scheme as *wrist* sensor and as miniature device built into the finger tip; *(ii)* first results on the development of a compound multipurpose tactile sensor system are reported in the following section.

[8]See also, e.g., (Kerr and Roth 1986; Vassura and Bicchi 1989; Iberall and MacKenzie 1990)

[9]ditto, and watch our wonderful hand during daily life manipulations. Currently, the TUM index finger can laterally reach its neighbor when properly re-mounted on the palm base.

4.1 Proprioceptive Information from a 6 D FTS

Generally, a 6 D **f**orce-**t**orque **s**ensor module ("FTS") connects two adjacent parts of a robot, and measures the currently acting force $\vec{f}_S$ and moment $\vec{m}_S$ at the (virtual) sensor center point. Particularly interesting locations are *(i)* the wrist – connecting the last manipulator arm segment with the end-effector, and *(ii)* the fingertip (precisely, between the last finger segment and the fingertip). The measure vectors can be transformed to the reference frame of interest, for example at the tool center point. Translations $\vec{t}$ of the reference frame leave the force unchanged $\vec{f}_t = \vec{f}_S$ but the leverage contribute to the moment $\vec{m}_t = \vec{m}_s + (\vec{t} \times \vec{f}_S)$, rotations affect $\vec{f}_S$ and $\vec{m}_S$ in the same manner.

The six components $\mathbf{F}_S = (\vec{f}_S^T, \vec{m}_S^T)^T$ give a *summary information*, to be more exact, two integrals of the force density distribution $d^3\mathbf{f}/d^3r$ over the entire end-effector volume V beyond the sensor center $\mathbf{r}_S$. [10]

$$\vec{f}_S = \int_V \frac{d^3\mathbf{f}}{d^3r} d^3r \tag{4}$$

$$\vec{m}_S = \int_V (\mathbf{r} - \mathbf{r}_S) \times \frac{d^3\mathbf{f}}{d^3r} d^3r. \tag{5}$$

This assumes that the wrist sensor module is the only connection to the robot arm, which means that no other structures transmit forces (stiff or pulling cabels etc.). $\mathbf{F}_S$ includes external forces transmitted over the surface as well as volumetric effects namely gravitation and inertia forces. Fortunately, are the later accessible, as described below, and can be separated from the interesting part, the exteroceptive information, i.e. the environment interaction forces.

What principal contact interactions are possible? Mason and Salisbury (1985) gives a detailed classification: **point**, *line* and *surface* contacts, **with or without friction**. Deviating from them, we like to re-divide here the classes of line and surface contacts into: *(i)* a **spot** contact is a spatially extended point. Each convex, non-rigid surface enlarges the interaction area on pressure, for example allowing our fingertip to turn a coin on the table; *(ii)* The class of **multiple contacts** is a more practical description of a non-perfect planar surface in contact with another possibly non-perfect planar surface or straight edge. It subsumes also textured surfaces.

A non-friction contact will slip until the tangential forces are zero. A contact with friction can transmit tangential forces and if it is spatially extended, the spot can transact also torques [11]. The maximal tangential force is bounded to the normal

[10] The transpose signs ^{T} are omitted, when the correct form is clear.

[11] which is no contradiction to Eq. 5, but understood as a local summary; Eq. 5 can be rewritten as sums of contact spot interactions, so called *wrenches* (Mason and Salisbury 1985).

force times the friction coefficient of the interacting materials (friction cone concept). This incipient slip criterion is of great practical relevance. There the contact is lost which might be intended or not – possibly causing trouble and damage. The local dynamics of a soft contact are rather complex and theoretically only tractable for some very idealized cases. Principal ways to avoid object slippage are: *(i) stable grasp selection* on the task planning level. *(ii)* The *detection of incipient slip* triggers a reflex to secure the grasp by increasing the contact forces.

4.2 Extracting Exteroceptive Information

What can we learn form a 6 D force-torque measurement? This depends on the kind of contact.

Hard rigid bodies – one contact point: Assuming a *single* interaction point, with or without friction, we know that no torque is transferred, and we can derive the contact force and the "line of action". Based on the known tool geometry we can determine a *set* of possible contact points. For example, for a pushed, convex tool this leaves a single solution, which is interesting, e.g, for rigid, convex fingertips.

Despite the fact, that hard point contacts are impractical for stable grasps, lots of theoretical work is published with this presumtion (e.g. Bicchi, Salisbury, and Brock 1993).

Soft bodies with one single contact spot with friction can exert torques. Since we measure remotely, we loose the position information to the extent that an unknown torque is inserted at the contact. If we know the contact position we can conclude the contact torque. The contact force is available in both cases.

Multiple contacts give summary information. Conclusions about contact locations require extra presumptions, which are usually unavailable. All forces beyond the sensor device are summed. For example, *internal grasp forces*, acting within a closed grasp are zeroed out and therefore invisible at the wrist.

This discussion underlines the *importance of getting localized sensor information* in order to obtain insight to complex multi-finger and multi-contact situations – see next section.

However, for the arm control, the integral force situation is indeed the value of interest. There, it gives the relevant overview – which is indispensable as long as other tactile senses cannot assure full spatial coverage (which is hard). E.g. the wrist "view" is not endangered of missing something at insensitive joint regions or finger backs.

4.3 The DLR Wrist Sensor

Fig. 4 shows the DLR force-torque sensor from the outside and Fig. 4.3 from the inside. It was developed by the robotics group of Prof. Hirzinger of the DLR, Oberpfaffenhofen, and is a spin-off from the ROTEX Spacelab mission D2 (Hirzinger, Dietrich, and Heindl 1989; Hirzinger, Brunner, Dietrich, and Heindl 1994). Its

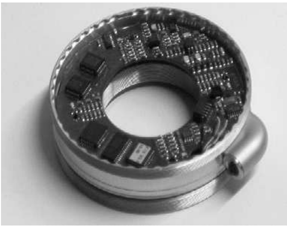


Figure 9: The DLR wrist sensor for sensing all six force and torque components acting in the (virtual) sensor center between the upper and lower part. The on-board electronics measures, pre-filters the primary strain-gauge sensors, and computes and transmits the temperature compensated results.

sensor cell has a new, stiff membrane architecture with strain-gauche sensors. The sensor device is characterized as stiff structure with a small maximum clearance of ± 0.15 mm and ± 0.3 mrad. This is advantageous for a high position accuracy, but for certain compliant operations a soft passive compliance is desirable. To suit both cases, a passive compliance module is mounted adjacent to the sensor, which can be mechanically locked (lower half in Fig. 4). If unlocked, it has a defined low hysteresis spring behavior in all 6 DOF with a clearance of ± 2 mm and $\pm 2°$. Passive compliance is helpful, for part mating, tolerating small misalignments, and, it simplifies control of stiff environment contact. Since the force control loop has an inner, fast position control loop, the limited resolution in position and time becomes less critical with a defined compressible interface module.

The DLR sensor can be characterized as *semi-autonomous sensor sub-system*. A build-in micro-controller takes care of overload detection, pre-filtering, temperature compensation, and the computation of the desired result $\vec{f}_S, \vec{m}_S$. It also communicates via a serial fieldbus system to the DLR-busmaster, as displays in Fig. 2.

This fieldbus concept is a very interesting solution to the cabling problem of a highly complex end-effector, as demonstrated in the ROTEX experiment ((Schott and Dietrich 1992; Dietrich, Gombert, Hirzinger, and Schott 1993)). It implements a strict, *single-master – multiple-slaves* protocol and can control several intelligent sub-systems, all connected to the same four-wire bus (polling-oriented, no interrupts). Two wires serve the information transport, the other two as a central AC power supply. This 20 kHz low voltage line simplifies the derivation of different voltages needs for multiple sub-systems (analog, digital part, reference voltages, etc.)

The particular FTS protocol specifies commands like remote resetting, switching a power saving mode, selecting signal filters, and re-programming the microcontroller (Schott and Dietrich 1992; Dietrich et al. 1993). The integration experience shows, that a handshaked fieldbus protocol has also certain disadvantages: *(i)* It requires one active process to collect the sensor data from the fieldbus. *(ii)* Only one such active process is supported (single master). *(iii)* During this polling process, the central resource VME-bus is needed. More desirable would be a fully autonomous sensor module, collecting the data and keeping it ready for all interested user process. In section 5.2 we return to this idea of keeping time-stamped records of remotely collected sensor data.

The software integration of the FTS is located in the control task of the Puma. The request and collection of the FTS data is weaved into the communication to the Puma controller, minimizing host processor wait time and sensor data age. As depicted in Fig. 3, the current end-effector orientation is used to compute the gravitational force and moments, which is subtracted from the measured force vector $\mathbf{F_S}$. The result is transformed into the selected task frame. Additionally to the force law, the net force $\mathbf{F}_{trans}$ is written to a cyclic shared memory segment for monitoring purpose. Following (Müller 1993), Dücker (1995) implemented an auto-calibration procedure to determine the weight and the relative 3 D location of the center of gravity based a set of different, unloaded hand positions.

5 Tactile Sensors Development

The previous section discussed the benefit and limits of an integral force-torque sensing at the wrist. Here, we turn to possible add-on sensor systems to fill the previously explained sensor gap for dextrous hand control (Sect. 3.5).

The literature suggests several interesting fingertip sensor designs. A prominent architecture is a miniaturized 6 axes force - torque sensors, which gets built into a cylindrical frame with a hemispherical tip, see Brock and Chiu (1985, Vassura and Bicchi (1989, Dario et al. (1990, Bicchi et al. (1993, Bicchi (1992).

This mechanical construction is challenging – fragility and sensitivity are tied together – but it promises full force information. As discussed in section 4.2, the contact location can be inferred only for a single, rigid point – not for an extended contact spot with friction. This is a goal conflict, since a rigid point contact is usually less stable than a soft, compliant interaction surface.

Speeter (1988) and Dario et al. (1990) use piezoresistive touch sensing arrays from Interlink Electronics. The 16×16 sensor sites are made of an interesting force sensitive resistor material (FSR described below). The sequential readout of the 256 "taxels" is reported as demanding due to the cross coupling of all resistors (see also Liu et al. 1995 for further discussion on sensor arrays). The benefit is a tactile image, which is ready for linearization and image processing. Since the flexible sensor film has planar topology it can only be formed to cone shapes (see below).

Another interesting technique utilizes a piezoelectric transducer film. Dario et al. (1984) suggested to use PVDF (details are given below) as a pressure and texture sensor. By moving the sensor over the object, the spatial frequency of the surface grooves are determined by the dominant peak in the FFT spectrum (sampling frequency 85 Hz, Dario et al. 1992). In Dario et al. (1990,1992) a multi-sensor architecture is described including a 6 D force-torque sensor, a tactile image array, a thermal conductivity sensor, and several rings of PVDF strips for estimating surface roughness. An ultrasonic proximity sensor is planned for completeness. The list is impressive and demonstrates the search for a complex exploration tool. However, the practical usefulness as a feedback sensor for the purpose of controlling a robot finger is not entirely demonstrated. Many multi-sensor fingertip designs ignore the fact, that during a typical finger closing motion the tip sensor *rolls* over an object surface. A tactile sensor with an outer shape of a cylinder or cone (axis aligned in the finger direction) can be rendered "blind" as soon as the finger is adducted and the contact rolls over the edge. It is remarkable that this practical kinematic aspect is rarely addressed in the literature.

5.1 Tactile Sensor Design Issues

What are the main design issues for an effective tactile sensor system?

Contact force measurement is the primary goal with the priority steps: moments, tangential and normal forces. For micro motion control a good differential resolution is particularly important.

Contact location detection is the second central aim. In ideally complements a 6 D fingertip force sensor. Good spatial resolution and separability of multiple contact spots is desired.

Coverage: The fingertips are the most interesting spots for all *precision grasps* (see Cutkosky and Howe 1990). *Power grasps* involving whole hand manipulation and in particular, contact transition control require feedback also from the phalanxes and palm surface.
Preferred are sensor techniques which are **inexpensive** and **easy to shape** to the actual geometric need.

Contact slippage detection is essential for a fast reflex to control grasp stability, as mentioned earlier.

Environment identification by active touch: The search and identification process for a light switch in the dark is obviously a feasible strategy. The signatures of move, scratch, or knock induced vibrations, can characterize the surface texture (Allen 1987; Klatzky and Lederman 1989). To allow future research on these surface recognition strategies, the sensor system should be able to deliver entire time series in contrast to quasi-static single value measurements.

A thermal sensor is useful in order to discriminate object materials by their differences in thermal conductivity and capacity. This works particularly fast and reliably, e.g. between metals and non-metals and will be considered as a future supplement.

The sensor geometry must match the kinematic needs of the robot finger. This means, during the rolling motion of the fingertip on a object surface, the sensor should not exhibit a “blind spot” or edge.

Contact imaging opens the field for image processing techniques on tactile images. This can serve to find the location of a gap or hole. Obvious problems of tactile arrays are high costs, high dead-time (for read-out and post-processing) and substantial geometric restrictions (see above).

The cabling problem: High spatial resolution requires many sensors and many cables. Cables need space, create extra stiffness, and are generally endangered to getting ripped-off when the arm is moving. Using arrays of sensors trades required number of cables against effort and time spent for readout. In order to yield a good signal to noise ratio, the cable between the sensor and the first pre-amplification stage should be short. Early signal conditioning, sampling and multiplexed transition appears as an attractive solution.

System Integration: The tactile information serves multiple purposes as explained before. Therefore it should be rapidly accessible to several processes (at *“manus”* and *“druide”*, see communication map Fig. 2) and should not require computing resources for the collection and handshake task. For time series analysis the recent history of the sensor data should be accessible in shared memory.

Despite the obvious importance of good sensor sub-systems, no suitable *and* affordable devices are commercially available. Can the resulting feedback information gap simply be closed by a learning system? It would by very desirable, but they can not, since even open loop control strategies (without feedback) do need reliable sensor data for training and testing.

5.2 The Prototype

In spite of limited resources, we started to develop our own sensor sub-system based on two different refabricated, commercially available film sensors. This work was carried out by Ján Jockusch, who implemented the first prototypes and the necessary interface electronics to use them in our robotics environment. In the following we give an overview on the developed sensor approach, a more detailed account can be found in (Jockusch 1996). The structure of the multi-layer compound sensor is illustrated in Fig. 10.

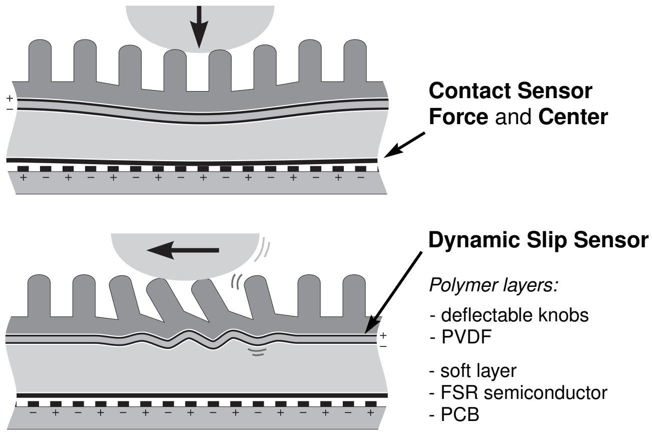


Figure 10: The sandwich structure of the multi-layer tactile sensor. The FSR sensor measures normal force and contact center location. The PVDF film sensor is covered by a thin rubber with a knob structure. The two sensitive layers are separated by a soft foam layer transforming knob deflection into local stretching of the PVDF film. By suitable signal conditioning, slippage induced oscillations can be detected by characteristic spike trains.

5.2.1 Force Sensor

The force sensitive part is made of FSR, fabricated by Interlink Electronic, Santa Barbara, CA. In contrast to the ready-made fixed-size sensor array, the *Force Sensitive Resistor* ("FSR") core material is rather inexpensive. It is a 0.1–1 mm thin, flexible polymer film. It shows an exponentially decreasing resistivity with increasing pressure in the range of 0.007–7 bar. The physical effect is caused by the band-gap in the semi-conducting polymer, which is here strongly dependent on the geometric deformation.

FSR sensors are built in a sandwich structure. At one side of the the FSR material, two patterned electrodes are mounted in the shape of interdigitating finger sets (without direct contact). On both sides the sensor is covered for mechanical and electrical protection. Pressure on the sensor will lower the local resistivity and can be measured as a drop in the (overall) electrical resistance between the two electrodes. Since the functional dependency is not linear in pressure, the superposition principle does not hold and the sensor readout is dependent on the area over which a certain force is applied. Therefore, FSR can not replace an absolute vectorial force sensor, but it can serve as an inexpensive, normal force monitor with high differential resolution.

5.2.2 Force Center Sensor

The FSR sensor can be made position sensitive. By forming one electrode such, that the fingers connect to a linear strip resistor (instead of a cross-connection), one can determine the *center of pressure* along the strip axis (by differential measurement of the resistivity against the end of the stripe resistor). The resulting combined sensor needs only three cables and can measure alternatingly the force and the contact spot position.

5.2.3 Dynamic Slippage Sensor

The slip sensor utilizes the piezo-effect – i.e., the surface charge generation on mechanical strain – in polarized polyvinylidenefluoride (PVDF), a copolymer of Teflon. The PVDF is extruded, mechanically oriented by stretching to thin films, and polarized by exposure to an intense electric field. After roughening the surface, electrodes are applied to the film to collect the charge[12]. PVDF shows a very wide frequency range ($10^{-3} \ldots 10^{9}$ Hz) and is susceptible to many influences. We use the PVDF-film as a supplementary, dynamic sensor to the relatively slow FSR material[13].

The slip sensing concept resembles somehow the combination of skin vibration sensors (Pacinian corpuscles, Johansson 1978) and papillary ridges leading to the finger prints of the human hand. Here, we coat the dynamic sensor by a rubber surface with little knobs. When rubbing over a object surface, rub-oscillations of

[12]The measured charge density is proportional to the applied stress or strain, but the coefficient, the *piezo strain constant*, is anisotrope. Additionally, PVDF shows a strong pyroelectric effect – i.e., charge generation on temperature changes. The pyroelectric effect of PVDF gets employed for passive infrared sensors, which detect human motion for automatized light switching, door control, etc.

[13]The mechanical reaction time of FSR is 1–2 msec.

the knobs are induced and transmitted to the PVDF layer, as illustrated in Fig. 10. These are caused by losing friction contact, slapping back, re-gaining contact, moving with the surface until the bending force breaks contact and regains it.

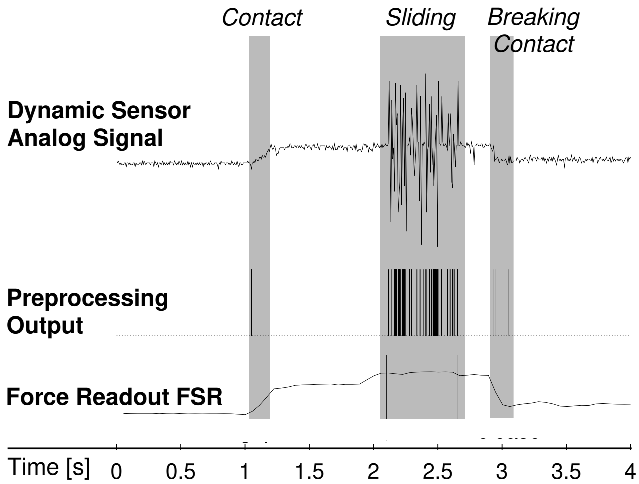


Figure 11: Recordings from the raw and pre-processed signal of the dynamic slippage sensor. A flat wooden object is pressed against the sensor, and after a short rest tangentially drawn away. By band-pass filtering the slip signal of interest can be extracted: The middle trace clearly shows the sudden contact and the slippage phase. The lower trace shows the force values obtained from the second sensor.

Fig. 11 shows first recordings from the sensor prototype. The raw signal of PVDF sensors (upper trace) is bandpass filtered and thresholded. The obtained spike train (lower trace) exhibits already a very promising, characteristic signal shape. The first contact with a flat piece of wood induces a short signal, which is easy to discriminate from the following slip phase.

The first version of a complete fingertip sensor is built on four faces of a polyhedron. Fig. 12 shows several intermediate steps in making the compound sensor. Care is taken, to avoid “blindness” for forces applied at the edges. Along the edges, the interlayer soft foam is shaped in thus a way, that forces are deflected to the adjacent FSR sensors. The complete re-fabrication of the FSR sensor allows to shape the sensor film to the required geometry and to avoid insensitive sensor margins, as shown in (Jockusch, Walter, and Ritter 1996).

5.2.4 Tactile System Integration

As discussed before, the main components of effective tactile system integration are *(i)* multi-channel, sensor-near signal conditioning, *(ii)* multiplexed transfer to

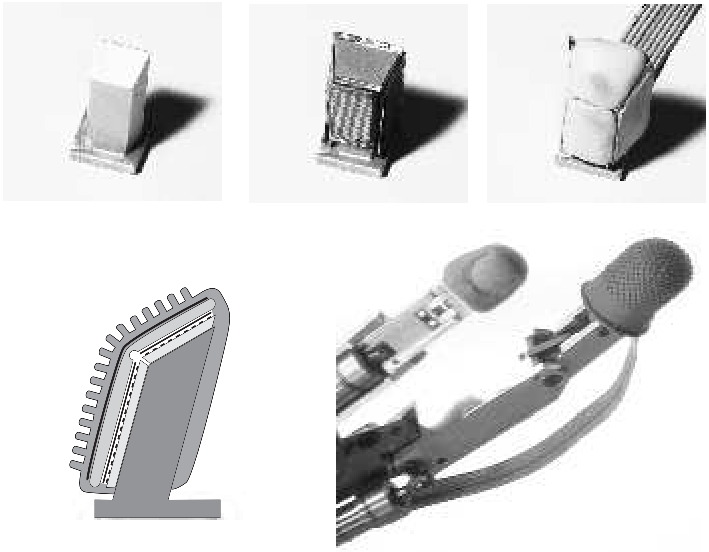


Figure 12: Intermediate steps in making the compound sensor, see text

the control systems, and *(iii)* random access to the demultiplexed signals and signal histories (time series). Fig. 13 illustrates this concept in more detail and Fig. 14 depicts the prototype electronics (designed and realized by (Jockusch 1996)).

Each sensor needs its separate signal conditioning circuitry. Additionally, the FSR contact sensor needs alternating measurement of force and position. The **Multiplexing Analog Signal Sampler** ("MASS") coordinates the data collection process using a Motorola 68HC11 micro-controller. According to a scan-list the desired sequence of analog channels is sampled (8 bit), time-stamped and transmitted via a 500 kBaud synchronous high speed serial data line. MASS supports pre-conditioning of 64 channels and their sampling with 32 kHz. The scan-list can run-time configured via a separate serial line (RS 232, not shown in Fig. 13), which allows to dynamically configure the data sampling process. By selective multi-sampling the band-width of certain channels can be increased (at the expense of a reduced sampling frequency for other channels).

The data stream is sent to the complementing system "BRAD" – the **Buffered Random Access Driver** hosted in the VME-bus rack, see Fig. 2. BRAD writes the time-stamped data packets into its shared memory (8 kByte) in cyclic order. In this way, all processes can conveniently access the most recent sensor data tuple. Furthermore, entire records of the recent history of sensor signals are readily available for time series analysis.

Summarizing, the first results from the new tactile sensor system look very promising. They give rise to expect solutions *(i)* for filling the present gap in

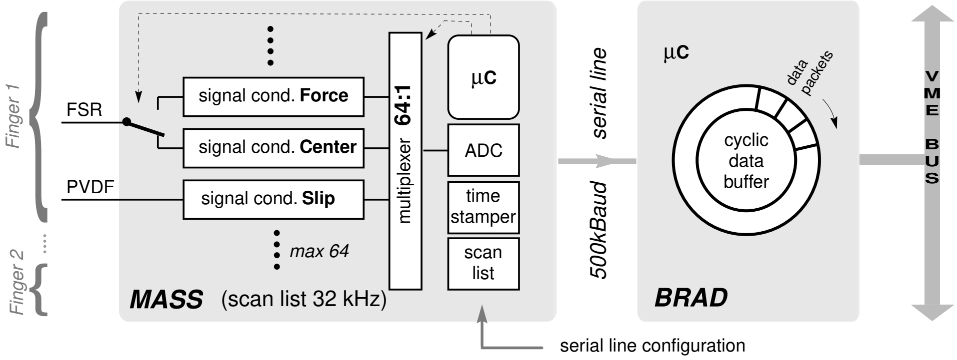


Figure 13: MASS ("Multiplexing Analog Signal Sampler") and BRAD ("Buffered Random Access Driver") mirror up to 64 analog sensor signals for efficient data access to the VME bus. The scan-list determines the sequence of channels, which are samples with 32 kHz.

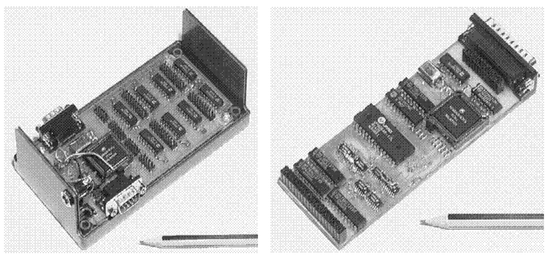


Figure 14: View on the MASS and BRAD electronic boards.

proprioceptive sensory information about the oil cylinder friction state (proprioception); *(ii)* for getting fast contact state information for task-oriented low-level grasp reflexes (exteroception); *(iii)* for obtaining reliable contact state information for signalling higher-level semi-autonomous robot motion controllers. Further task-oriented integration aspects are discussed in (Jockusch, Walter, and Ritter 1996) and (Walter and Ritter 1996).

6 Remote-Sensing: Vision

To empirically qualify vision as an important sensory component, it is useful to consider the effect when human eyesight is missing. The activities of blind persons are most handicapped, when they encounter an unknown or unstructured environment. In known environments, only orientation updates are required for enabling mobility (in- and outside of buildings) and normal manipulations of daily life (cooking with ordered storage etc.) In unknown situations memory does not help – the blind must get in touch. Comparing to the high-bandwidth visual system, this is a very slow process, since it additionally requires to go there (walking/reaching) and gather tactile information (immediately by hand, or via a blind stick).[14]

6.1 The Machine Vision Task

Inspired from the biological example, our own eyesight, intensive research started in the seventies to reach for a comparable, general image processing and understanding system. Fed by monocular and especially binocular gray scale images, the system should yield, as a first step, segmentation, identification, localization and depth information on all objects in the scene. But it turned out, that the problems were greatly underestimated.

A fundamental problem of low-level vision is that the camera does not measure the remote 3 D object geometry but rather its 2 D image – the image of the light reflected from the object in the particular direction towards the camera. This light ray is influenced by the local reflection properties (spectral=color, angular=specular and diffuse reflection etc.) and the local surface lighting situation. The local lighting situation is determined by the distribution of natural and artificial light sources, but also influenced by the reflections from neighboring objects.

It is interesting to note, that another field of research, computer graphics aims at most efficient computation of exactly these physical phenomena. *Ray tracing* techniques render photo-realistic 2 D images given the geometry of all objects,

[14] Additionally, well trained blinds show impressive acoustic orientation capabilities based on ambient noises.

plus their surface properties, plus the distribution of light sources. Photo-realistic scene rendering can be considered as the inverse of low-level machine vision task: Given an image, what is the scene geometry?

Another fundamental problem is the *image segmentation* and the *correspondence problem*, higher level vision tasks, which build the associates object references to image segments. Marr (1982) states it eloquently:

> "Despite considerable effort over a long period, the theory and practice of segmentation remained primitive for two reasons. First, it was well-nigh impossible to formulate precisely in terms of the image or even of the physical world what the exact goals of segmentation were. What, for example, is an object, and what makes it so special that it should be recoverable as a region in an image? Is a nose an object? Is a head one? Is it still one if it is attached to a body? What about a man on a horseback?"

The image analysis of a complex scene needs a suitable amount of world knowledge. Attempts to solve this task in a general, goal-free manner seems to be an ill-posed problem.

Nevertheless, machine vision is a very challenging task, which is superbly solved in our own visual system. As a consequence, today we can discern a number of different research directions. They are motivated within different, narrowed, task-specific domain. A number of applicable techniques emerged in order to facilitate a scene description and the desired information gain on remote objects (see e.g. Fischler and (Eds.) 1987). For a report on recent work on the task of object recognition in the SFB "Baufix" scenario with hybrid approach that combines neural and semantic networks see (Heidemann and Ritter 1996) Depending on the chosen technique, different hardware requirements can arise.

Simplify object – background segmentation

- by choice of the background,
- controlled lighting (diffuse, avoid specular reflexes),
- usage of chromatic imaging,
- adding markers, or coloring of the object (if possible)

Image sequence analysis for moving objects: segmentation of object-ground is facilitated by determining flow fields from differential images.

Active vision paradigm: An alternative to passive vision derives from observations of the biological systems. Active vision, also called "animate vision" systems are inter-active with the environment, they are not general-purpose and goal-free, but rather oriented at specific task and structured as integrated skills and behaviors (Ballard 1991). In "animate vision" the active control of external camera parameters are in the foreground, in contrast to the active probing by selective lighting or echoing (triangulation, laser range scanning etc.). The image analysis problem is simplified

- since an actively verging system gives depth estimate and simplifies stereo matching;
- since it facilitates search of the best object view (seeking, homing, and zooming in)
- supports to generate multiple views of (also of quasi-static) scenes from different camera views (fusing stereo and even more views)
- generating image sequences of scenes simplifies the determine corresponding object locations when fusing images.

In contrast to the processing of force-torque values, is the information gained by image processing system of very high-dimensional nature. In spite the computational simplification of the active vision paradigm, the computational demands are enormous and require all effort to quickly reduce the huge amount of raw pixel values to useful task-specific informations.

6.2 Vision Hardware

Our vision related hardware currently consists of several main parts:

Cameras

A variety of **cameras** gives flexibility in choosing scenarios: two monochrome CCD cameras Pulnix TM-786, two color CCD cameras Pulnix TMC-76 with remote heads, two 3-chip RGB CCD cameras JVC KY-F55. For cost-effectiveness we use standard equipment, where the discrete pixel information is transmitted as an analog signal before re-digitizing[15] (PAL norm, 25 Hz interlaced, 756×580 pixels).

Specialized Image Processors

Two image processing hardware systems allow rapid pre-processing. *(i)* two Androx ICS-400 boards in the VME bus system of *"druide"*(see Fig. 2). *(ii)* A MaxVideo-200 with a DigiColor frame grabber extension from Datacube Inc.

[15] Despite the fact, that most people in the machine vision community have to fight with the information loss and (fully digital) solution is readily available, the market forces exhibit here the interesting phenomena of "technology locking" (a further amazing example is DOS and its derivatives.)

Each system allows simultaneous frame grabbing of several video channels (Androx: 4, Datacube: 3-of-6 + 1-of-4), image storage, image operations, and display of results on a RGB monitor. Image operations are called by library functions on the Sun hosts, which are then scheduled for the *parallel processors*. The architecture differs: Each Androx system uses four DSP operating on shared memory, while the Datacube system uses a collection of special pipeline processors working easily in frame rate (max 20 MByte/s). All these processors and crossbar switches are register programmable via the VME bus. Fortunately there are several layers of library calls, helping to organize the pipelines and there timely switching (pipe altering threads).

Specially the latter machine exhibits high performance if it is well adapted to the task. The price for the speed is the sophistication and the complexity of the parallel machines and the substantial lack of debugging information provided in the fast switching, parallel, and large data streams. This lack in debug tools makes code development somehow tedious.

However, the tremendous growth in *general-purpose computing power* allows to shift already the entire exploratory phase of vision algorithm development to general-purpose high-bandwidth computers. Fig. 2 exposes various graphic workstations and high-bandwidth server machines at the LAN network.

Active Vision Systems

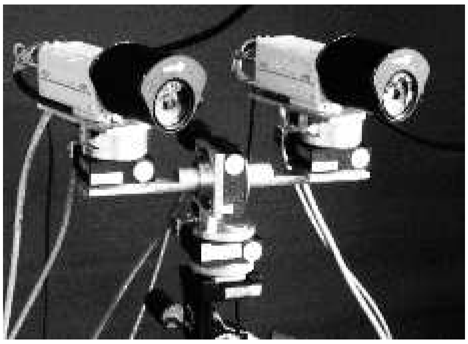


Figure 15: The active stereo camera head offers 4 DOF for head gaze control (pan, tilt, and vergence) as well as 3 DOF in both motor lens system (focus, aperture, zoom).

Two principal active vision systems are available: *(i)* The **Puma robot** carries a color camera and allows a wide range of possible camera views. Fig. 4 depicts the color camera with a fixed focus wide-angle lens, attached to the end-effector. This **camera-in-hand** configuration is most useful for object exploration and manipulation control and surveillance – with or without object contact.

(ii) The **binocular active camera head** is depicted in Fig. 15. It is a copy of the AUC robot head, which was developed at the University of Aalborg, Denmark

(Christenson 1992) and built by Robert Kubisch (1995). Each of the two 3-chip CCD cameras is equipped with an active lens system, which allows to computer control the intrinsic camera parameters focus, aperture, and zoom. For control of the extrinsic camera parameters, two neck motors can pan and tilt the head (fixed eye base, 8°/s). Together with the two fast vergence motors (80°/s) the camera pair can take stereo images in any interesting direction and distance. As depicted in Fig. 2, the total ten degrees of freedom are controlled by hardware residing and interfaced by the second VME-bus hosted by the Sun SparcStation *“argus”* .

6.3 Further Auxiliary Devices

The following components complete the description of hardware components depicted in Fig. 2:

- As already pointed out, **controlled lighting** conditions can tremendously simplify reliable vision processing. Beyond the installation of light shutters, we implemented a remote control system for artificial lighting. By means of a fieldbus running through the lab, eight groups of lamps are controllable by manual buttons or via software interfaced by *“druide”*(on/off and brightness)

- Finding **markers** and landmarks in scene images is particularly reliable and easy, when their identification can be verified using software controlled lamps or LEDs. *“druide”*interfaces to twelve channels of a digital IO used for switching light emitting lamps or diodes (LED). One channel is reserved for the laser pointing device, axially mounted in the hand (see Fig. 4.)

- Intuitive **user input devices** are important. Since the computer keyboard can only be concidered very intuitive for the group of intense computer users and advanced hackers – today the computer mouse plays a key role. With 2 D moves and button-clicks, virtual sliders and buttons can be manipulated in the graphical user interface (GUI). For commanding actions in 3 D space, which simultaneously involve translation and rotation changes, the conventional mouse interaction is cumbersome due to the required toggling of command modes. In these cases, the **3 D SpaceMouse** (Turbo Version, Space-Control GmbH) allows the user to act more intuitively by pushing, pulling, twisting, rolling, etc. a soft mounted cap. Similar to the 6 D wrist sensors, the exerted forces and torques are measured and suitably transformed to the desired kind of action command, see Walter and Ritter (1996).

7 Discussion and Conclusion

We described our hardware architecture design and implementation work carried out in the *Arbeitsgruppe Neuroinformatik* (connectionist research group). As mentioned earlier, the infrastructure of the robotics laboratory of the collaborating group (Arbeitgruppe Praktische Informatik, headed by Alois Knoll) employs the same real-time robot control scheme as described, details are reported elsewhere.

The major actuation components are a 6 DOF robot manipulator (Puma 562) and a multi-fingered hydraulic robot TUM hand. Aiming at intelligent, adaptive, task-oriented, dextrous exploration and manipulation, we find the following issues important:

- very often, the need for good perception gets severely underestimated;[16]
- intelligent actuation can not be expected without suitable sensors – nor can be learning or adaptation;
- the robotics and automation industry usually substitutes intelligent sensing by constructing the environment in a highly predictable manner. This led to a tremendous lack of suitable (and commercially available) sensor systems and of general sensor system integration support.
- Sensing should be:
  - robust and reliable (the real world bears real risks);
  - fast (high band-width and low latency, providing reactiveness and better stability in control loops);
  - multi-aspect ("multi-media") and redundant (cost issue);
  - suitable (configurable and possibly re-configurable);
  - well integrated in control and command architecture;

Currently, our sensory equipment comprises, beside the built-in proprioceptive position encoders an additional 6 D force-torque wirst sensor (DLR) for measuring integral hand forces.

For improving the fine motion control and acquiring spatially resolved contact state information we started to develop a new compound multi-purpose tactile sensor system. We reported results from the first prototype, desinged also for

[16]This is probably due to the extremly versatile structured and well trained human perception mechanisms, which do not reveal their importance. Our own visual, actustic, and haptic senses operate sub-consciously and effortless. Considering a robot system, our attention is primarily attracted to the apparent "action" parts of an robotic system. Since we cannot directly perceive the machine's perception, one easily neglects it.

low-force and short-latency grasp reflex control. Special attention was paid to realize high-bandwidth (re-configurable) data transfer system (including precise time stamps and random access history records) able to serve multiple monitoring and control processes in the VME-bus system.

The control and command architecture is built on general purpose Unix workstations in tight connection with two VME-bus systems. Real-time control is achieved by special SunOS kernel extensions and for the robot hand the real-time OS pSOS+ on the embedded controller *manus*.

The contact-free remote sensing facility is based on video imaging and allows to choose between several cameras and image acquisition and pre-processing systems, enhanced by three different active camera systems (mono-, stereo-head, and the camera-in-hand configuration).

A good software integration plays a key rule for easy experimentation capabilities. The *"Service Object Request Management Architecture"* ("SORMA") was recently developed to serve the (partly) conflicting, special needs of an robotics laboratory, among them are real-time constraints (time-optimal invocation) secure and robust interoperation of hardware resources in an distributed object-oriented computing environment. Further details are reported in SFB360-TR-96-3 (Walter and Ritter 1996).

Acknowledgments

Many thanks to the following contributors: Christof Dücker (robot arm control, zero-gravity compensation), Hartmut Holzgraefe (TUM hand control, Space-Mouse, PowerGlove software, a.o.), Ján Jockusch (tactile sensors, MASS, and BRAD), Rudi Kaatz (safety switch on force sensitive work bench, PowerGlove, a.o.), Michael Krause (interbus communication, a.o.), John Lloyd (RCCL, RCI), Robert Rae (light bus, active camera head, Datacube, a.o.), Stefan Rankers (hand control software MCCL), Dirk Selle (hand kinematics and simulator, a.o.), and Thomas Wengerek (VME rack, a.o.). We further acknowledge inspirations for the conceptual design of the multi-layer tactile sensor, in particular by Owen Holland, who brought our attention to the versatile PVDF sensor material and Jörg Butterfaß for the idea on re-tailoring FSR sensors.

Some of the illustrations were contributed by Dirk Selle [5,8], Ján Jockusch [10,11,12,14], and Robert Rae (former Kubisch) [15]. Thanks to Christof and Jan for their help on lab snapshots and for proofreading earlier version of this manuscript.

This work was supported by the ministry for research and education of NRW and by the federal ministry under grant BMFT-ITN9104AO.

References

Allen, P. (1987). *Robotic object recognition using vision and touch*. Kluwer.

Ballard, D. (1991). Animate vision. *Artificial Intelligence 48*(1), 57–86.

Bicchi, A. (1992). A criterion for optimal design of multi-axis force sensors. *Robotics and Autonomous Systems 10*, 269–286.

Bicchi, A., J. K. Salisbury, and D. L. Brock (1993). Contact sensing from force measurements. *International Journal of Robotics Research 12*(3), 249–262.

Brock, D. and S. Chiu (1985). Environment perception of an articulated robot hand using contact sensors. In *Robotics and Manufacturing Automation, ASME, Nov. Miami, USA*, pp. 89–96.

Christenson, H. (1992). The AUC robot camera head. In *Proc. SPIE Applications of Artificial Intelligence X: Machine Vision And Robotics, Int. Soc. for Optical Engineering*, pp. 26–33.

Cutkosky, M. and R. Howe (1990). Human grasp choice and robotics grasp analysis. In S. Venkataraman and T. Iberall (Eds.), *Dextrous Robot Hands*, pp. 5–31. New York: Springer.

Dario, P. et al. (1984). Ferroelectric polymer tactile sensors with anthropomorphic features. *IEEE*, 332–340.

Dario, P., B. A. A. Sabatini, M. Bergamaco, and G. Buttazzo (1990). A fingertip sensor with proximity, tactile and force sensing capabilities. In *IEEE International Workshop on Intelligent Robots and Systems (IROS'90)*, pp. 883–889.

Dario, P., P. Ferrante, G. Giacalone, L. Livaldi, B. Allotta, G. Buttazzo, and A. Sabatini (1992). Planning and executing tactile exploratory procedures. In *IEEE International Conference on Intelligent Robots and Systems*, pp. 1896–1903.

de Wit, C., P. Noel, P. Aubin, and B. Brogliato (June 1991). Adaptive friction compensation in robot manipulators: Low velocities. *Int. J. of Robotics Research 10*(3), 189–199.

Dietrich, J., B. Gombert, G. Hirzinger, and J. Schott (1993). The endeffector assembly system (EEA). Technical note, Deutsche Gesellschaft für Luft- und Raumfahrt, (DLR) Oberpaffenhofen, Germany.

Dücker, C. (1995). Parametrisierte Bewegungsprimitive für ein Roboter–Kraft/Momenten–Sensor Handsystem. Diplomarbeit, Technische Fakultät, Universität Bielefeld.

Fischler, M. and O. F. (Eds.) (1987). *Readings in Computer Vision – Issues, Problems, Principles, and Paradigms*. Morgan Kaufmann.

Fu, K., R. Gonzalez, and C. Lee (1987). *Robotics : Control, Sensing, Vision, and Intelligence*. McGraw-Hill.

Hayward, V. and R. Paul (1986). Robot manipulator control under unix rccl: A robot control "c" library. *Int. Journal of Robotics Research 5*(4), 94–111.

Heidemann, G. and H. Ritter (1996). Objekterkennung mit neuronalen netzen. Technical Report SFB360-TR-96-2, TF-AG-NI, Universität Bielefeld, D-33615 Bielefeld.

Hirzinger, G., B. Brunner, J. Dietrich, and J. Heindl (1994). ROTEX – the first remotely conrolled robot in space. In *Intern. Conf. on Robotics and Automation (San Diego)*, pp. 2604–2611. IEEE.

Hirzinger, G., J. Dietrich, and B. Brunner (1990, sep). Multisensory telerobotic concepts for space and underwater applications. In *Proc. of the Space and Sea Colloquium, Paris, France*, pp. 151–161.

Hirzinger, G., J. Dietrich, and J. Heindl (1989). Zum stand der robotikarbeiten der DLR. *DLR-Nachrichten* (58), 2–9.

Hyde, J. M. and M. R. Cutkosky (1993). Contact transition control: An experimental study. In *IEEE Int. Conf. on Robotics and Automation, Atlanta*, pp. 363–368.

Iberall, T. and C. L. MacKenzie (1990). Opposition space and human prehension. In S. Venkataraman and T. Iberall (Eds.), *Dextrous Robot Hands*, pp. 32–54. New York: Springer.

Jacobsen, S. (1984). The UTAH-MIT dextrous hand: Work in progress. *Int. J. of Robotics Research 3*(4), 21–50.

Jockusch, J. (1996). Taktile Sensorik für eine Roboterhand – Ein microcontroller basiertes integriertes Sensorsystem. Diplomarbeit, Technische Fakultät, Universität Bielefeld.

Jockusch, J., J. Walter, and H. Ritter (1996). A tactile sensor system for a three-fingered robot manipulator. In *Proc. Int. Conf. on Robotics and Automation (ICRA-97)*, pp. (submitted).

Johansson, R. S. (1978). Tactile sensibility in the human hand: Receptive field characteristics of mechanoreceptive units in the glabrous skin area. *J. Physiol. 281*, 101–123.

Kerr, J. and B. Roth (1986). Analysis of multifingered hands. *Int. J. of Robotics Research 4*(4), 3–17.

Klatzky, R. L. and S. Lederman (1989). Intelligent exploration by the human hand. In S. Venkataraman and T. Iberall (Eds.), *Dextrous Robot Hands*, pp. 66–81. New York: Springer.

Koeppe, R. and G. Hirzinger (1995, Stanford CA, June). Learning compliant motions by task-demonstration in virtual environments. In *Symposium on Experimental Robotics ISER*95*.

Kubisch, R. (1995). Aktives Sehen mittels eines binokularen Kamerakopfes: Ein Ansatz auf der Grundlage neuronaler Netze. Diplomarbeit, Technische Fakultät, Universität Bielefeld.

Liu, H., P. Meusel, and H. G (1995). A tactile sensing system for the DLR three-finger robot hand. In *Proc. of ISMCR*, pp. 92–96.

Lloyd, J. (1988, January). RCI user's guide. report CIM-88-22, McGill Reseach Center for Intelligent Maschines, McGill University, Montréal.

Lloyd, J. and V. Hayward (1992, April). Multi-RCCL user's guide. Technical report, McGill Reseach Center for Intelligent Maschines, McGill University, Montréal.

Lloyd, J. and M. Parker (1990, July). Real time control under Unix for RCCL. In *Robotics and Manufacturing ISRAM'90*, Volume 3, pp. 237–242. ASME Press, New York.

Marr, D. (1982). *Vision*. San Francisco: Freeman.

Mason, M. and J. Salisbury (1985). *Robot hands and the mechanics of manipulation*. MIT Press.

Menzel, R., K. Woelfl, and F. Pfeiffer (1993, Sept). The developement of a hydraulic hand. In *Proc. of the 2nd Conf. on Mechanotronics and Robotics*, pp. 225–238.

Müller, M. (1993). *Roboter mit Tastsinn*. Dissertation, Fakultät für Maschinenbau und Elektrotechnik der Technischen Universität Braunschweig, Germany.

Paetsch, W. (1993). *Exemplarische Untersuchung zu mehrfingerigen Robotergreifern: Aufbau – Regelung – Systemintegration*. Meß- Steuerungs- und Regeltechnik, Nr. 363. VDI-Verlag Düsseldorf.

Paul, R. (1981). *Robot Manipulators: Mathematics, Programming, and Control*. MIT.

Rankers, S. (1994). Steuerung einer hydraulisch betriebenen Roboterhand unter Echtzeitbedingungen. Diplomarbeit, Technische Fakultät, Universität Bielefeld.

Schott, J. and J. Dietrich (1992). Endeffector assembly. Tech. Note D2-RX-TN-89023-DOP, DLR – Institut für Robotik und Systemdynamik, Oberpfaffenhofen.

Schutter, J. D. (1986). *Compliant Robot Motion: Task Formulation and Control*. Ph. D. thesis, Katholieke Universiteit Leuven, Belgium.

Selle, D. (1995). Realisierung eines Simulationssystems für eine mehrfingerige Roboterhand zur Untersuchung und Verbesserung der Antriebsregelung. Diplomarbeit, Technische Fakultät, Universität Bielefeld.

Speeter, T. H. (1988). Flexible, piezoresitive touch sensing array. *SPIE Optics, Illumination, and Image Sensing for Machine Vision 1005*(3), 31–43.

Vassura, G. and A. Bicchi (1989). Whole-hand manipulation: Design of an articulated hand exploiting all its parts to increase dexterity. In P. Dario, G. Sandini, and P. Aebischer (Eds.), *Robots and Biological Systems: Towards a New Bionics?*, pp. 165–177. Springer-Verlag.

Venkataraman, S. and T. Iberall (1990). *Dextrous Robot Hands*. New York: Springer. (Ed.).

Walter, J. (1991). Visuo-motorische Koordination eines Industrieroboters und Vorhersage chaotischer Zeitserien: Zwei Anwendungen selbstlernenden neuronalen Algorithmen. Diplomarbeit, Physik Department der Technische Universität München.

Walter, J. and H. Ritter (1995). Local PSOMs and Chebyshev PSOMs – improving the parametrised self-organizing maps. In *Proc. Int. Conf. on Artificial Neural Networks (ICANN-95), Paris*, Volume 1, pp. 95–102.

Walter, J. and H. Ritter (1996). Service Object Request Management Architecture: SORMA concepts and examples. Technical Report SFB360-TR-96-3, Universität Bielefeld, D-33615 Bielefeld.

Walter, J. A. (1996). *Rapid Learning in Robotics*. , Technische Fakultät, Universität Bielefeld. http://www.techfak.uni-bielefeld.de/∼walter/pub/.