DLR Hand II: Experiments and Experiences with an Anthropomorphic Hand

Ch. Borst, M. Fischer, S. Haidacher, H. Liu and G. Hirzinger

German Aerospace Center - DLR
Institute for Robotics and Mechatronics
82230 Wessling, Germany
Email: [Christoph.Borst, Max.Fischer, Steffen.Haidacher, Hong.Liu, Gerd.Hirzinger]@dlr.de

***Abstract*— At our institute, two generations of antropomorphic hands have been designed. In quite a few experiments and demonstrations we could show the abilities of our hands and gain a lot of experience in what artificial hands can do, what abilities they need and where their limitations lie. In this paper, we would like to give an overview over the experiments performed with the DLR hands, our hands abilities and the things that need to be done in the near future.**

I. INTRODUCTION

In many robotics research institutes all over the world new robotic hands are developed [2]. Grasp theory is quite well funded. There is quite a lot of work in the field of grasp synthesis. Different control strategies to robustly and stably grasp an object were implemented and their efficiency is demonstrated in more or less complicated experiments. But still there are not too many hands that do at least some part of the things that man can do with his hands.

At our institute, two generations of antropomorphic hands have been designed. In quite a few experiments and demonstrations we could show the abilities of our hands and gain a lot of experience in what artificial hands can do, what abilities and features they need and where their limitations lie. In the sequel, we would like to give an overview over the applications and experiments performed with the DLR hands, our hands abilities and the things that need to be improved.

II. FEATURES OF THE DLR HAND II

In this section we want to describe all the features that enable the hand to be used in these different experiments, applications and demonstrations.

1) Fingers Can Be Bent Backwards: Although this feature introduces singularities in the Cartesian (fingertip position command) mode, it has proved to increase the grasping abilities of the DLR Hand II very much: While usual finger design only allows for point contacts at the finger tips in case of precision grasps, bending the fingers backwards allows for much more robust line contact with the distal finger links. This grasp type is known as pinch grasp and has proved to be much more robust than pure precision grasps.

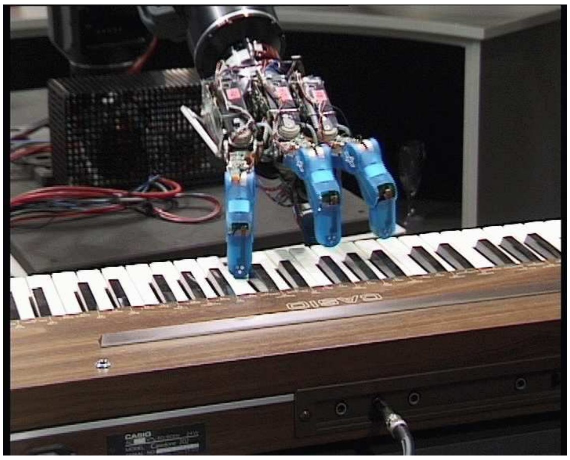

Fig. 1. The DLR Hand II playing piano

2) Palm Design and Shape: For power grasps the palm design is crucial, as the palm adds a lot to the stability of a power grasp. We used rapid prototyping to optimize the shape of the palm. Within our work on power grasp planning, palm design is still a topic of research.

3) Reconfigurable palm: One design feature of the DLR Hand II is the reconfigurable palm: Similar to human hands the DLR Hand has an additional degree of freedom in the palm to adapt the hand pose to the actual need: For power grasps, a “flat” palm is needed, while for precision grasps and fine manipulation it is desirable to have a hand configuration with opposing fingers, especially thumb and ring finger. Currently, we can switch between these two hand configurations.

4) Speed: The high joint speed of about 382 deg/s in each of the joints enables the hand to perform actions where high finger speed is of great importance. This holds especially true for the piano playing demonstration as well as for the ball catching. Our experience is that the mentioned joint speed is sufficient.

5) High Degree of Integration: Especially when putting together more complex robotic systems as e.g. a mobile platform with the DLR LBR and the DLR Hand II mounted on it, easy integrability proved to be an important point. The fact that there is no need for additional hardware (except from a VME controller computer), the small number of communication and power supply cables together with a customized tool adapter [12] which allows the hand to be removed from the arm in seconds turned out to be a major advantage.

6) Flexible Control Software Architecture: One very important feature of our hand is the flexible control software architecture as described in [12]: We can quickly realize new applications and consistently switch between controllers and applications in real-time, command the change of controllers and applications in real-time from an external interface, but at the same time an operator can interrupt each action using a graphical user interface.

7) Motion Teaching: Pre-planning of grasps and grasp movements is one possibility to plan finger trajectories. In practice, however, there is the need of teaching finger trajectories (coordinated for all fingers) or hand poses manually. Therefore we realized a motion and pose teaching facility: Using a data-glove, we can teach a hand motion. The teaching facility is used to record and save this teached motion in a file. We also can store given hand poses which can be teached either using a data-glove or a zero torque control mode of the fingers and manually bringing the fingers to their desired pose.

The previously stored trajectories and hand poses can be commanded by their file name in real-time using a pre-fetching and caching mechanism.

A. Hand and Finger Control

In this section we summarize the different basic control schemes that are used in the above experiments.

1) Joint Stiffness and Zero Torque Control: Each finger joint has a strain-gauge based torque sensor. Using this sensor a joint stiffness control has been realized:

$$J_d\ddot{\theta}_e + B_d\dot{\theta}_e + K_d\theta_e = \tau_{ext} \tag{1}$$

where $\theta_e = \theta_d - \theta_m$ is the position error, while q_d, q_m, and τ_{ext} are the desired position, actual joint position, and the actual joint torque, respectively. J_d, B_d, K_d are the desired target impedance parameters of the robot finger; impedance control specifies this desired impedance relationship as a generalization of the second order dynamics of a damped spring. In order to fulfill the equation 1, we can also introduce an explicit force control scheme, i.e. , let $\tau_d = J_d(\ddot{\theta}_d - \ddot{\theta}_m) + B_d(\dot{\theta}_d - \dot{\theta}_m) + K_d(\theta_d - \theta_m)$ where τ_d is the desired torque. let τ_e be the error function: $\tau_e = \tau_d - \tau_{ext}$

Now we can introduce a simple PI control scheme with τ_e as input. If τ_e converges to zero, the actual impedance parameters will converge to the desired values automatically. In steady state, all measured and desired velocity and acceleration values are zero. This induces that the value of the steady state torque is the stiffness multiplied by the steady state deformation $(\theta_d - \theta_m)$, and the joint behaves like a programmable spring. If τ_d is set to zero then we get a zero joint torque control mode.

2) Cartesian Impedance Control: When a robot hand executes a fine manipulation task, the fingertip should behave like a programmable spring, being soft in the contact normal direction and hard in the tangential directions. Using the impedance control scheme described in [13], we can control the complete impedance property of a finger.

3) Coordinated Grasp Control: One common task in robotic grasping is to perform a precision grasp with only the finger tips contacting the object. In order to maintain best possible contact positions in the presence of disturbances Salisbury [14] proposed a controller that has been widely used and extended. The principle idea is to relate position and velocity errors in control to the object, implement impedance control at the position and velocity of the object and also respect zero space motions of the object:

$$\begin{aligned}
d\mathbf{x}_{fing} &= \mathbf{x}^d_{fing} - \mathbf{x}_{fing} \\
d\mathbf{x}_{obj} &= \mathbf{G}d\mathbf{x}_{fing} \\
\mathbf{v}_{obj} &= \mathbf{G}\mathbf{v}_{fing} \\
\mathbf{f}_{fing} &= \mathbf{G}^T\left(\mathbf{K}_x d\mathbf{x}_{obj} + \mathbf{K}_v(\mathbf{v}^d_{obj} - \mathbf{v}_{obj})\right) \\
\mathbf{G} &= \begin{bmatrix} \left[\mathbf{W}\mathbf{W}^T\right]^{-1}\mathbf{W} \\ kernel(\mathbf{W})^T \end{bmatrix}
\end{aligned} \tag{2}$$

with $\mathbf{x}^d_{fing}$, $\mathbf{x}_{fing}$ being the desired and measured Cartesian finger tip position, $d\mathbf{x}_{obj}$, $d\mathbf{x}_{fing}$ being the object and finger tip displacement, $\mathbf{v}_{obj}$, $\mathbf{v}_{fing}$ being the object and finger velocity. The matrix $\mathbf{W}$ is the grasp matrix as defined in [14]. $\mathbf{f}_{fing}$ is the control output per finger. This controller has been implemented and tested in conjunction with an algorithm to detect contact to perform autonomous grasping processes.

III. GRASP PLANNING

One very desired functionality for a dexterous hand is autonomous grasping. In the following we want to shortly introduce the problem of grasp planning and show which aspects lead us to our grasp planner design.

1) What is needed to plan grasps?: Any grasp planning approach needs kinematic and geometric information about the hand and the object to be grasped. That means it has to be known how the hand can move and how hand and object are shaped.

Fig. 2. Power and precision grasp configuration

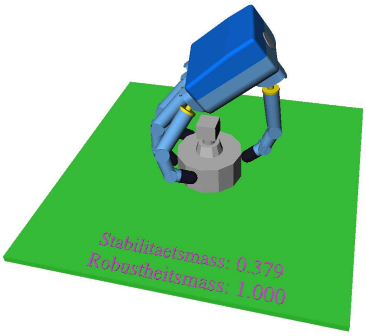

Fig. 3. Planned precision grasp

Fig. 4. Planned pinch grasp

The kinematic and geometric model of an artificial hand is usually available from CAD. On choosing the representation for the object model we had in mind that the model can be obtained also autonomously: We use a polyhedral model and allow gaps and unconnected faces. Such models can be created with common 3D reconstruction methods like structured light, laser scanners or stereo vision.

2) The grasp planning problem: Grasps for dexterous hands are reasonably divided in two main grasp categories [7]: Precision grasps for high manipulability, where only the fingertips are in contact with the object and power- or enveloping grasps where high forces must be resisted or exerted, where the whole hand can be in contact with the object (fig. 2).

The planning of these types of grasps is very different: With precision grasps one searches for a few fingertip contacts (3-5) that allow for a stable grasp. For most of these contact sets one can compute more than one valid hand configuration as we show in [5].

Power grasps are mainly determined by the geometrical constraints of hand and object so one tries to find a suitable hand configuration to “wrap the fingers around the object” and then calculates the resulting contacts.

3) The grasp planner for the DLR Hand: As stated above it is possible to start a precision grasp planner with finding contacts and then calculate a valid hand configuration to realize the grasp. We also observed that about 20 % of a set of 4 randomly chosen contacts on a set of geometrical and real world objects (cube, sphere, cylinder, coffee mug, martini glass, etc.) result in force-closure grasps, which is the most common quality measure for precision grasps [15], [3].

Therefore we decided to implement a random based modular grasp planner. The algorithm can be summarized as follows, for details see [4]:

1) Choose four contact points randomly on the object
2) Calculate a kinematically valid hand configuration using an optimization approach [5] (see fig. 3 for a precision and fig. 4 for a pinch grasp); Start at 1) if there is none.
3) Perform a collision test (hand-object) to see if geometric constraints fail (hand intersects the object); Start at 1) in case of a collision.
4) Compute a quality measure for the grasp
5) Store the grasp in a list sorted by the grasp quality and start at 1) until there are enough grasps in the list or a given time limit is exceeded.

The most interesting step is the quality computation. We use a measure introduced by Ferrari and Canny [9] which physically means: The quality of a grasp is as good as the minimal wrench that breaks the grasp if all fingers can press with unit forces. We developed an incremental convex hull constructing algorithm to calculate this measure instead of using the widely used measure where the sum of all finger forces is limited to unit force which is harder to be motivated physically. With this approach we can compute about 100 valid and force-closure grasp candidates in 20 - 60 s depending on the complexity of the object on a Pentium III 700 MHz Linux PC.

We then choose the best of the candidates which makes force-closure a minimum quality measure for our grasp planner. With the modular design we can change contact generation methods from random to heuristic and add additional quality measures easily, depending on the demonstration tasks.

IV. APPLICATIONS

In the context of a robotic hand there are mainly two independent fields of applications. In the first area, an operator controls a robotic hand and arm to perform tasks at places he cannot be present himself for various reasons (dangerous, hazardous environments, long distances). In the second area, the robotic system performs autonomous tasks, that were previously taught and have a number of variable parameters to adjust to the actual environment.

A. Teleoperation Experiments

Here we present several experiments related to teleoperation using DLR Light Weight Robot II (LBR) and DLR Hand II.

1) Experiments with a Dataglove: The most intuitive way to control a robotic hand is by making it follow the movements of a human hand, e.g. by the use of a dataglove. However, a *robotic* hand differs kinematically from human hands. Thus, a mapping between the measured postures of the human hand and the desired posture of the robotic hand has to be constructed. One possible mapping was proposed in [10]. Here the goal was to map tip positions of the human and robotic hand one by one, neglecting posture of the remaining links. The mapping was trained using a neural net. This approach is feasible for grasping the same object in the remote robotic environment and in the local human control environment simultaneously by precision grasps or to train motions of the hand in a simulation environment and consecutively executing this task in the robotic environment. However, it is strongly dependent on the training set of the neural net and thus on the user it was trained for. Additionally, this approach is not suitable for power grasps, because the position of the links is not directly controllable. Another approach to map human and robotic motions is to linearly map joint angles between the human and robotic hand. This way, the absolute positions of the robotic hand and the human hand differ, but motions of joints are transferred directly to the hand. Thus the operator can adjust the posture of the robotic hand interactively. This approach is suitable for powergrasps, is intuitively and easily adjustable to different users.

2) Telemanipulation using Stereovision, Dataglove and Force-feedback: In one setup that was used in several demonstrations an operator was in another city, several 100 km away from the robotic environment. The goal of this experiment was to prove the suitability of DLR Hand II and LBR II to perform every day tasks in a human environment through telemanipulation. In this setup, the operator could control position and orientation of LBR II using a control ball. The hand was controlled using a dataglove as described in section IV-A.1. Additionally a VTI Cybergrasp exo-skeleton was used to create one dimensional force feedback per finger. The data connection to these three devices was implemented in two serial interfaces. The data traffic of these interfaces was routed through a coupling to an ISDN line, connecting the remote operator's site to the robotic scenery in our laboratory (fig. 5). The remote operator was provided with visual information about the scene through a stereo image that was transmitted via ISDN or a satellite link.

Different tasks were performed as e.g. pouring a glass of wine from a bottle, opening and closing drawers, grasping objects within the drawer, putting them back inside, switching light switches, handing over a flower, a microphone, etc.

There are several requirements for a robotic hand experienced in this setup. First of all, the hand needs to be able to perform powergrasps which is essential for handling large objects as bottles and boxes. To support this type of grasp, a soft finger surface with sufficient friction, for example silicone, is useful in order to improve adjustment to the object's surface and keep required normal forces small. Additionally, a sufficiently strong hand is required because the usual gravitational load put on the hand is about 2-3 kg and enough reserve is required to apply internal forces holding the object to provide enough friction. Restrictions occurred in these experiments where a higher quality force feedback was required. This is mainly the case when handling tiny objects.

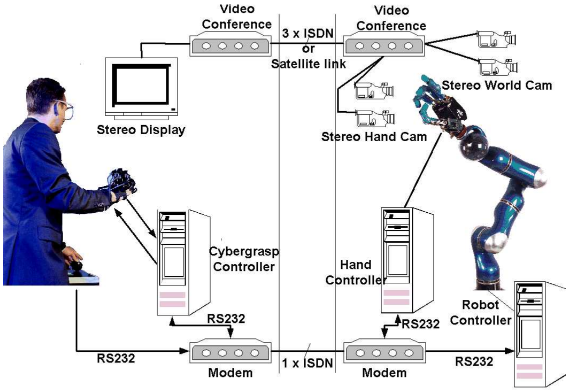

Fig. 5. Telemanipulation setup with hand and arm

B. Task Level Programming Experiments

In this section we present experiments performed with DLR Hand II and LBR II that consist of multiple operations fulfilling a higher order task. A task level programming system proposed in [6] was used. These experiments were performed in relatively autonomous manner, performing and adjusting previously taught operations.

1) Catching a Softball with DLR LBR II and Hand II: As a demonstration of efficient programming, control system design and velocity in execution we developed a setup that tracks and catches a soft ball which has been thrown by any volunteer. Parts of the experiment has been published in [11]. Hereby, two cameras track a thrown ball. Using an extended Kalman Filter, the trajectory of the ball can be predicted. Optimizing the intersection of the trajectory with the robot's catching region returns a desired catching point. Extending [11], we now use DLR Hand II to catch the ball.

Previous examinations showed, that the average available time span for closing the hand before the ball bounces off again is around 50 - 80ms. Restrictions in this experiment clearly occurred in the size of the ball. Experiments to catch a soft soccer ball right in the flight did not work completely reliable due to large size as well as experiments with small soft balls that could not be completely retained by the hand. Requirements for this experiment was the capability of the system to withstand moderate impacts of a thrown object, a time-wise deterministic behaviour of the control system itself and a sufficiently high velocity of the fingers.

Fig. 6. Arm and hand catching a ball

2) Tracking and Grasping an Object: One thing that man repeatedly does with his arms and hands is to pick up and grasp slowly moving or still objects. To demonstrate this ability, we realized a visual servoing setup. The DLR arm (hand and robot) used and a pair of cameras was mounted on the DLR Hand's wrist (see fig. 6). The arm then approaches and tracks an object (a ball in our case) and the hand performs a grasping movement to pick up the ball. When human beings pick up an object, the hand performs a characteristic movement from a resting pose to an open pregrasp pose and finally grasps the targeted object. The actual hand pose depends on the distance of the hand from the object to be grasped. To mimic this behavior with the DLR arm, we derived a distance measure from the visual servoing module described below and use this measure to interpolate between three given hand poses (rest, pregrasp, grasp). This simple approach leads to an impressive human-like grasping behavior.

For the visual servoing module we use a simple Jacobian based approach: We "teach" the desired goal position simply by bringing the object to be grasped in the desired goal position, measure its coordinates in the stereo images, then move the arm a small distance in three orthogonal directions and get the object's image coordinates again. From this information we easily can estimate the Jacobian J that maps Cartesian deviations to deviations in image coordinates. It is straightforward to use it's inverse to realize a visual servoing facility.

3) Playing the Piano: To demonstrate the abilities and suitability of a robotic hand to perform general tasks in an environment originally designed for humans we taught DLR Hand II to play a standard piano keyboard (fig. 1). In order to be able to play multiple songs without having to program all pieces in robot control commands, we taught one complete scale of notes and stored these trajectories in a sample file. This was done using impedance control with a very low value of impedance and by moving the fingers manually. This way a complete scale could be taught and stored in a sample file. This file was combined with a music arrangement suitable for the robotic hand to present a trajectory for a given piece. In this context we excluded the thumb for kinematic reasons. We were able to play most songs. Restrictions occurred only in situations were 4 or more fingers would be necessary. In our experiments, this is one of the very few situations, where the standard kinematic setup and the mostly used number of four fingers is not suitable for all occasions. On the other hand for this experiment we required the hand to be easily teachable, have a sufficient accuracy in reproducing once trained positions, achieve a reasonable speed in finger motions and be able to contact a these high velocities with a rigid environment.

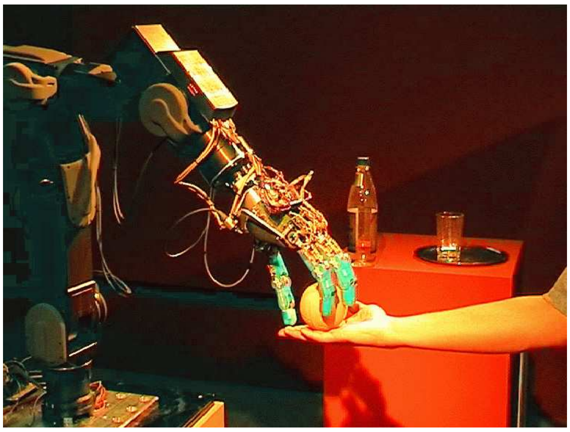

Fig. 7. Arm and hand tracking a ball

4) Service Robot in Human Environments: One of the major development interests of humanoid robotic systems is to develop assistants for elderly or bodily challenged people. The goal of a project called DIROKOL (low-cost light weight service robot) project, funded by the Bavarian Research Foundation, was to realize a mobile platform equipped with hand and arm and to demonstrate the task oriented programming of such a system. DLR arm and hand were mounted on a mobile platform equipped with on-board cameras. One of the tasks to be demonstrated was to navigate to a designated room, locate the doorknob, open the door, navigate to a table, locate a desired object, in this example a can, and grasp it. Apart from the integration of the hand controller in a task level programming system, the key problem to be solved was the robust contact of hand and arm with the environment and robust grasping under the constraint of world model errors. This could especially be solved by using impedance control modes of arm and especially hand.

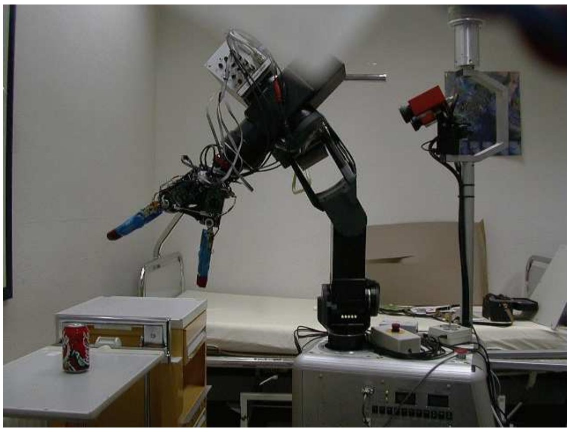

Fig. 8. DIROKOL

V. CONCLUSION AND FUTURE WORK

From our experience in the experiments described above we found that our hand already can do impressive and useful things, but we also learned about the limitations and possible improvements:

Fine manipulation and grasping of small objects is still a difficult task. To our opinion, this is not due to the sensor and control quality, but to the finger tip design: Although we use a silicon coating for the finger tips, we think that finger tips need to be much softer to increase grasp robustness and stability. Our finger tip design already resembles finger nails, but should be much more distinct.

We are very content with the planning results of our grasp planners, but integrating a grasp planner in real world systems has its own challenges: In the field of grasp planning we are about to integrate the grasp planner with a state-of-the-art path planner [1]. Precision grasps are fine in theory, in practice pinch and power grasps are much more important. We currently integrate the pinch grasp planning facility in our grasp planner and develop a randomized autonomous power grasp planner. Another important issue is creating geometric models of objects to be grasped online and make them usable for a grasp planner. We currently experiment with laser scanner and stereo vision based generation of partial object models.

To increase the robustness and stability of grasps, we are integrating a grasp force optimization module, described in [8], in our grasp controller. One precondition for a grasp force optimizer is the knowledge about the actual contact models and grasp forces. We currently try to derive the contact model information (i.e. contact on the finger tip, surface normal) from position and force/torque sensors.

All in all we can state that trying to make artificial hands work in real world situations provides us with a lot of interesting work for the future.

VI. REFERENCES

[1] B. Baginski. *Motion Planning for Manipulators with Many Degrees of Freedom - The BB-Method*. Number 195 in DISSKI. Infix, 1999.

[2] A. Bicchi. Hands for Dexterous Manipulation and Robust Grasping: A Difficult Road Towards Simplicity. *IEEE Transactions on Robotics and Automation*, 16(6), December 2000. .

[3] A. Bicchi and V. Kumar. Robotic grasping and contact: A review. In *Proc. IEEE Conf. on Robotics and Automation*, pages 348 – 353, San Francisco, California, April 2000.

[4] C. Borst, M. Fischer, and G. Hirzinger. A fast and robust grasp planner for arbitrary 3d objects. In *Proc. IEEE Conf. on Robotics and Automation*, pages 1890–1896, Detroit, Michigan, May 1999. .

[5] C. Borst, M. Fischer, and G. Hirzinger. Calculating hand configurations for precision and pinch grasps. In *Proc. of the 2002 IEEE/RSI/GI International Conference on Intelligent Robots and Systems*, page accepted, Lausanne, Switzerland, 2002. IEEE.

[6] B. Brunner, K. Landzettel, B. Steinmetz, and G. Hirzinger. Tele sensor programming - a task-directed programming approach for sensor-based space robots. In *Proc. ICAR 1995 7th International Conference on Advanced Robotics*, Sant Feliu de Guixols, Spain, 1995.

[7] M. R. Cutkosky and R. D. Howe. *Human Grasp Choice and Robotic Grasp Analysis*, chapter 1. Springer Verlag, 1990.

[8] M. B. et al. Dextrous hand grasping force optimization. *IEEE Trans. Robotics and Automation*, 12(3):406–418, 1996.

[9] C. Ferrari and J. Canny. Planning Optimal Grasps. In *Proc. IEEE Conf. on Robotics and Automation*, pages 2290–2295, Nice, France, May 1992.

[10] M. Fischer, P. van der Smagt, and G. Hirzinger. Learning techniques in a dataglove based telemanipulation system for the DLR hand. In *Proceedings of the IEEE Int. Conference on Robotics and Automation*, pages 1603 – 1608, Leuven, Belgium, May 1998.

[11] U. Frese, B. Bäuml, S. Haidacher, G. Schreiber, I. Schäfer, M. Hähnle, and G. Hirzinger. Off-the-shelf vision for a robotic ball catcher. In *Proceedings of the IEEE/RSJ Intl. Conference on Intelligent Robots and Systems Oktober 2001, Maui, Hawaii, USA*, 2001.

[12] S. Haidacher, J. Butterfass, M. Fischer, M. Grebenstein, K. Jöhl, K. Kunze, M. Nickl, N. Seitz, and G. Hirzinger. DLR Hand II: Hard- and software architecture for information processing. In *to appear in ICRA 2003*, 2003.

[13] H. Liu and G. Hirzinger. Cartesian impedance control for the DLR Hand. In *Proc. of the 1999 IEEE/RSI/GI International Conference on Intelligent Robots and Systems*, pages 106–112, Kyongyu, Korea, 1999.

[14] M. T. Mason and J. K. Salisbury. *Robot Hands and the Mechanics of Manipulation*. The MIT Press, Cambridge, MA, 2 edition, 1986.

[15] B. Mishra and N. Silver. Some Discussion of Static Gripping and Its Stability. *Transactions on Stystems, Man and Cybernetics*, 19(4):783 – 796, 1989. .