Neural Networks for Robot Hand Control

E. Littmann, A. Meyering, J. Walter, T. Wengerek, H. Ritter

Technische Fakultät
Universität Bielefeld
4800 Bielefeld

1 Introduction

If we compare present day information processing technology with the abilities of biological neural networks, we notice several puzzling differences. From an engineering perspective, biological neurons are slow and exhibit formidably large tolerances. However, even with our most powerful computer devices we have nowhere come even remotely close to the information processing capabilities of even simple animals, such as bees or birds, if issues of perception or movement control are concerned (in other fields, such as theorem proving or algebra, the performance of computers might seem much more acceptable. At least, biological brains do not deal effortlessly with such problems either). Therefore, to understand the working principles of neural networks, the field of robotics seems to be a particularly adequate and promising area, both for studying the kind of information processing tasks that are well solved by biological networks, and for exploring ways to replicate these capabilities with artificial neural networks.

These considerations are the motivation behind the research project described in this paper that aims at exploring the synthesis and the use of artificial neural networks for the control of *grasping movements* for robots. The process of grasping is a central issue for the application of robots in many domains and therefore, also of considerable practical interest. Currently employed robots use only very simple types of grippers that are limited in many respects. The main reasons for this state of affairs is the still very limited availability of more sophisticated, articulated grippers that are suited for an industrial environment, and the lack of suitable control strategies to make efficient use of such devices. This is mainly due to the fact that a multi-fingered gripper alone usually comprises more joints than the robot arm to which it is attached and the simultaneous control of several fingers raises many of the issues encountered when trying to coordinate the movements of multiple robots. The difficulty of the task is further aggravated by the fact that the combined system robot arm – hand is kinematically redundant, and that dextrous manipulation movements must take into account a fair amount of sensory information, primarily touch and vision. However, even if suitable sensors allow the acquisition of sufficient sensory information (a highly non-trivial task in itself), using this information for the sake of controlling the movement is a very difficult task, since the tactile and visual information encodes the mechanical interaction between complex geometrical shapes under conditions of substantial uncertainty about e.g. frictional properties of object surfaces and their precise shapes.

2 The Role of Learning Approaches

A precise theoretical analysis of these interactions and processes is only feasible in principle, but unlikely to succeed in practice, since in real world situations many parameters will not be known or will be too variable. Even in cases where such full analysis might succeed it may be too costly, since changing some aspects of the task might require to start all over again, which is not compatible with the idea that dextrous robots should ultimately add flexibility e.g. to a production process instead of absorbing large amounts of manpower into reprogramming.

On the other hand, we all are highly competent experts for dextrous manipulation tasks. Consequently, a very natural approach would be to create interfaces which can use this knowledge to teach robots grasping movements shown by human "experts", and to use the capabilities of neural networks for adaptively creating complex, nonlinear mappings between e.g. sensory signals and required joint movements to learn the required knowledge from examples and to refine it by further explorative motions of the robot.

To realize this goal, we have set out to investigate neural network archictures for the task of learning sensori-motor mappings under a variety of different conditions. The key theme is to exploit visual information about the hand movements of a human "teacher" for the control of robot hand movements. In a later stage of the work it is intended to integrate tactile information from the robot manipulator into the control loop in order to allow the control of forces which cannot be inferred from the visual image alone.

In order to learn the required transformations, an important issue is the proper choice of representations for the sensory and the joint data. Usually, many different degrees of freedom interact in a complex, non-linear way, and learning in the resulting, high-dimensional spaces is a hard problem which cannot be solved if the dimensionality of the input space becomes too high, since then it becomes infeasible to provide sufficiently many training examples to achieve a sufficient good generalization ability for new inputs. Closely related with this issue of representation is the issue of choosing a suitable network architecture which on the one hand must be sufficiently general to allow learning of the required mapping and which, on the other hand, is sufficiently modular to facilitate pre-structuring by the designer to employ a-priori-knowledge about the intended task.

3 Recognition of Hand Postures from Images

To meet these requirements, we have developed network architectures that are based on the so-called LLM-networks [14, 15]. An LLM-network can be thought as consisting of two layers (Fig. 1). The first layer (Fig. 1, bottom), which receives the input and which may be a self-organizing map [5] or a GRBF-network [13], acts as a kind of "gating network" for the elements of the second layer (Fig. 1, top). The first layer is trained in an unsupervised fashion. To each element of the first layer there corresponds an element of the second layer which computes an output quantity from the same or a different set of inputs as the input layer and contributes to the output of the system in proportion to the activation received from the first layer. By using a linear mapping network for each element of this second layer, training of the second layer can be based on a well understood, perceptron-like learning rule that requires no back-propagation steps. Still, for the entire system a high degree of flexibility results, since the gating layer can learn to decompose the overall mapping into a set of locally valid, linear mappings between which it will interpolate smoothly. The resulting architecture can be viewed as a special case of the architecture of "competing experts" as proposed by Jacobs and Jordan [4].

In recent work, we have investigated the capabilities of LLM-networks to extract the three-dimensional shape of complex, multijointed robot manipulators from monocular, perspective pixel images [11]. To solve this task with conventional robot vision algorithms constitutes a difficult problem, since the 3d-shape of a complex object of variable shape and varying amounts of self-occlusion must be recognized. The neural network approach allows to solve this task on the basis of only a set of training images of known manipulator postures. In a preprocessing stage, each image is reduced to a 36-dimensional feature vector that then forms the input of a LLM-network. No explicit geometrical model of the manipulator is needed and the performance of the trained network degrades gracefully if the geometry of the manipulator is changed or if parts of the manipulator are missing. In addition, the approach is not limited to the processing of visual images. In a very similar way, the method might e.g. be applied to the task of processing *haptic information* from tactile sensors. In this case, the availability of traditional algorithms is much more restricted than in the vision case, and consequently learning algorithms are of even higher interest.

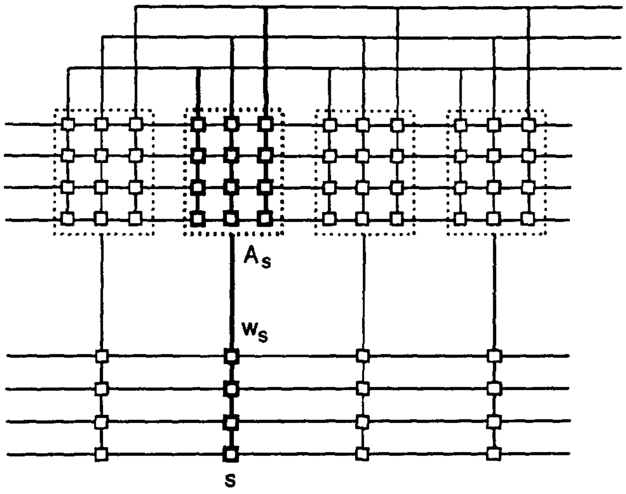

Fig. 1: Structure of LLM-network

To systematically explore the influence of different factors, such as changes in illumination, changes in the geometry of the hand, network size, number of training samples and learning parameters, on the recognition accuracy of the network, we have studied this approach extensively with artificially generated, computer rendered images of multi-joint robot manipulators of different, anthropomorphic shapes [11].

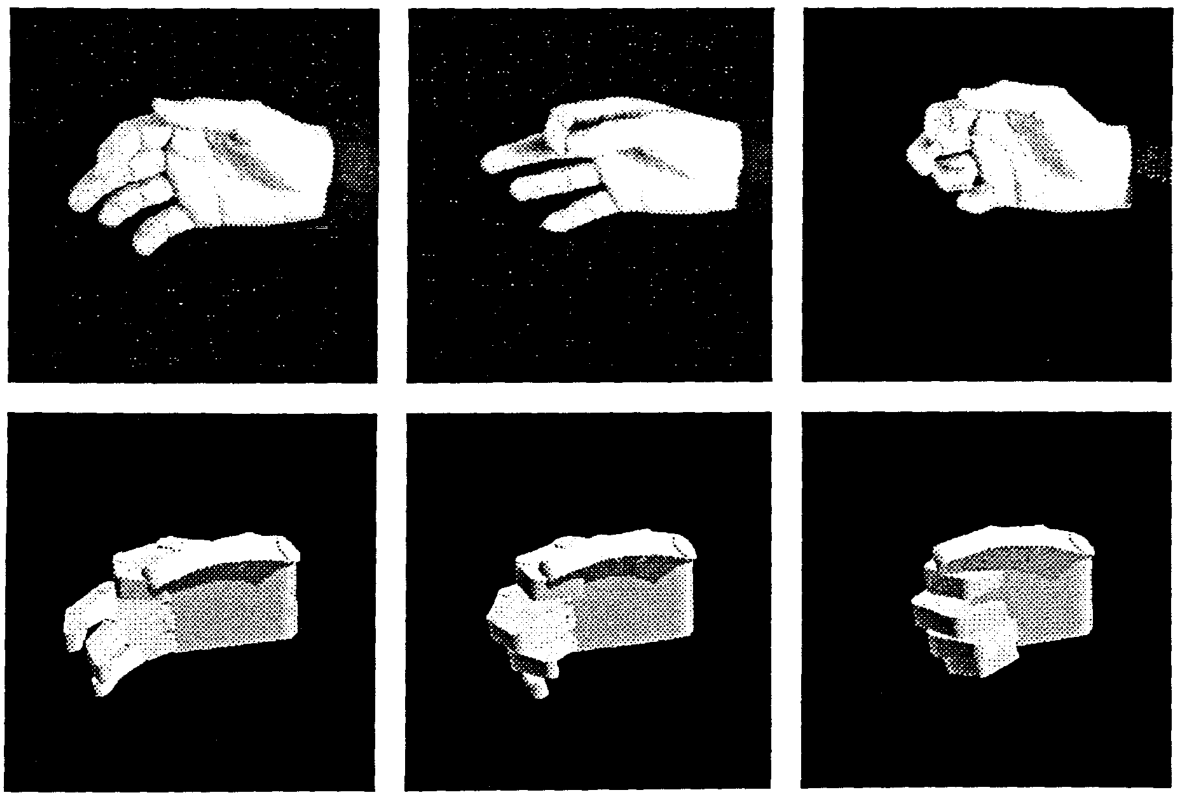

Fig. 2: Recognition of human hand postures with LLM-networks. Top row: input images. Bottom row: visualization of recognized hand postures using a 16-jointed hand model.

We also have begun to investigate the feasability of this approach for images of human hands. First results are reported in [12]. Fig.2 gives some visual impression of the resulting recognition performance for a typical case (10 units and 1000 training samples). Currently we are extending this work further, in particular to allow focal attention on parts of the entire image and to use an internal hand model to improve the recognition further.

4 Neural Architectures

The previous section has reported results demonstrating that already single LLM networks can be trained to extract important and nontrivial control information from pixel images after only very little preprocessing. However, controlling grasping movements is a complex process which most likely will require the integration of several neural modules into a larger system. Therefore, it is important to study systematic approaches to combine subnetworks into larger network architectures. In his experimental study of control architectures for robots Brooks has proposed a subsumption architecture which consists of a set of layers of simple agents [2]. In his case, each agent is non-adaptive and is implemented as an augmented finite state automaton. Agents on the lower levels perform simple behaviours, while agents on higher levels modify these behaviours by supressing inputs or outputs of lower level agents.

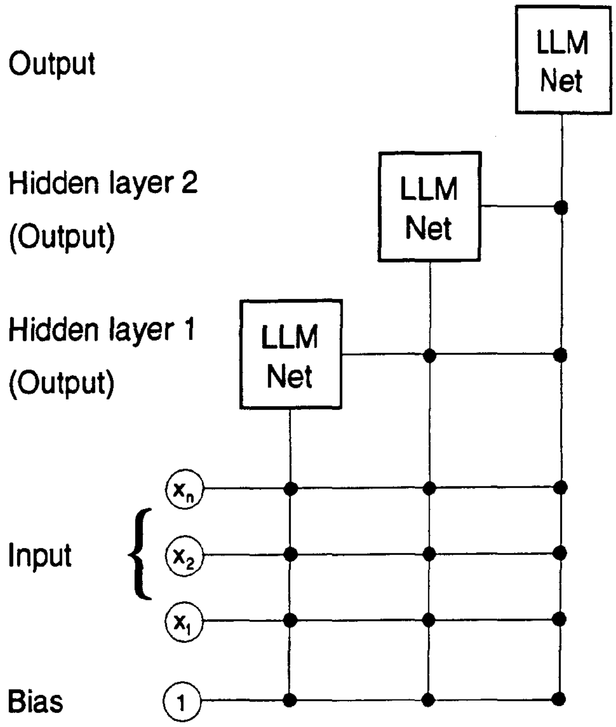

Fig. 3: Cascaded Network with LLM Modules.

Brooks has demonstrated that this concept is very well suited for implementing many important aspects of sensori-motor behaviour in real world environments. This success makes it attractive to look into the possibility to construct similar architectures from adaptive neural modules. Building on an earlier approach by Fahlman [3], we have developed a scheme to incrementally construct such *cascaded architectures* which start with at single low level network, to which incrementally further, higher-level networks are added, so that each new network can compute its transformation from all existing input lines together with the outputs of all its predecessors in the cascade. The resulting architecture is shown in Fig. 3, where LLM-networks are used as cascaded modules.

We have studied this architecture on a variety of benchmark problems with promising results [6, 7, 8]. Currently we are investigating the approach for learning *figure-ground separation* in visual images. A preprocessing stage that performs various figure ground separation tasks will significantly simplify the task of a subsequent hand recognition network and allow its successful operation under more variable conditions of background and illumination than presently, where all these factors must be taken into account by the same network.

Parallel to this work we are investigating neural network architectures the structure of which is inspired by the structure of the visual system of the toad. This work is carried out in close collaboration with the group of Prof. Ewert at the University of Kassel and attempts to construct a computer model of the visual recognition of prey in toads that allows to be used for robotics applications. The current simulation uses an approach which models the synaptic interactions between tectal and pretectal neurons in a reduced, low-dimensional parameter space (Fig. 4, 5). This makes it possible to apply learning algorithms for optimizing the prey recognition performance. Since an essential step in this process is the figure-ground separation, we expect that this work will also provide valuable insights for solving the segmentation task described in the previous paragraph.

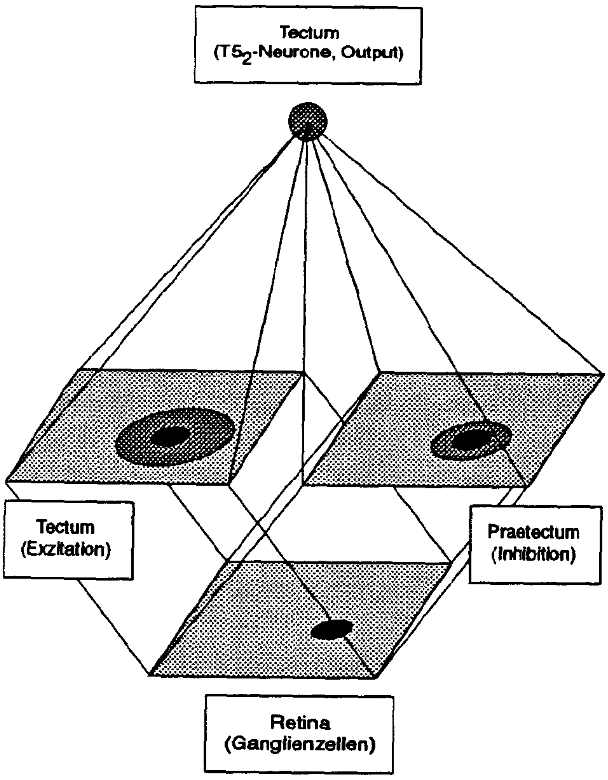

Fig. 4: Sketch of the synaptic interactions between tectal and pretectal neuro ns.

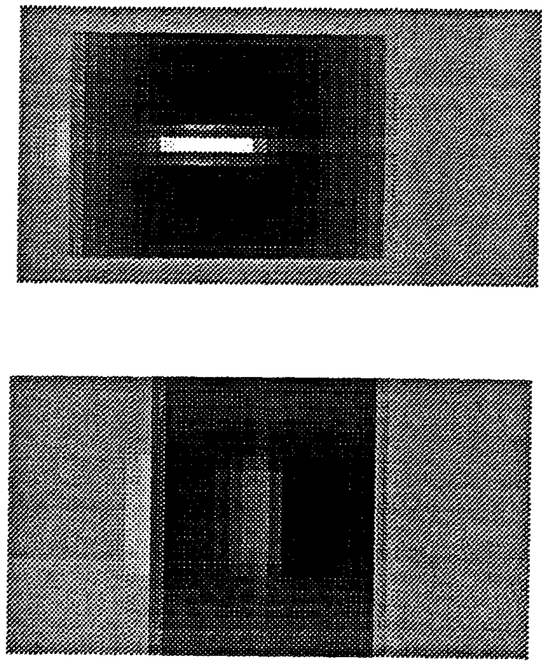

Fig. 5: Simulated Activity. Upper: response to prey, Lower: response to predator. White pixels indicate strong excitation, black pixels inhibition.

5 Experimental Environment

The previous research relies heavily on extensive computer simulation, since reconfiguring a robot or a sensory system is more easily done if the system still is implemented in software instead of in hardware. However, many aspects of real systems cannot be modelled faithfully enough, or simulations would become too complicated and slow. Therefore, it is of central importance to study the performance of algorithms also for actual hardware implementations and to set up facilities with which neural networks can be interfaced easily and flexibly with commercial robot and vision hardware [17, 9]. To this end we have set up a robot laboratory comprising a PUMA 562 robot arm with 6 degrees of freedom and an image processing facility consisting of two ANDROX ICS400 Boards. An important consideration was to overcome the limitations of the commercially available robot controllers with respect to computational power and especially to the transparency of their programming languages. In our implementation we chose to employ a Unix workstation to directly control the robot in real-time via a high speed communication link.

Fig. 6 illustrates the main hardware components of our installation. The Puma 6 DOF manipulator is connected to the Unimation Controller (Mark III, bottom left) which itself contains several controllers. The separate servo controllers for each revolute joint are commanded by a main controller CPU. This CPU usually runs the industrial robot software VAL II which is rather inflexible for our purpose, as it does not support control by an auxiliary computer. To achieve real-time control through an Unix workstation we employed the powerful software package RCCL/RCI (*R*obot-*C*ontrol-*C*-*L*ibrary and *R*eal-time *C*ontrol *I*nterface) [10]. The Unix workstation is a SUN SparcStation, which hosts some interfacing hardware and two image processing boards (ICS-400, Androx Inc.), each based on four digital signal processors. These boards provide sufficient computing power for fast data extraction and preprocessing.

All subsystems are directed by a control program ("planning task"), written and executed as an

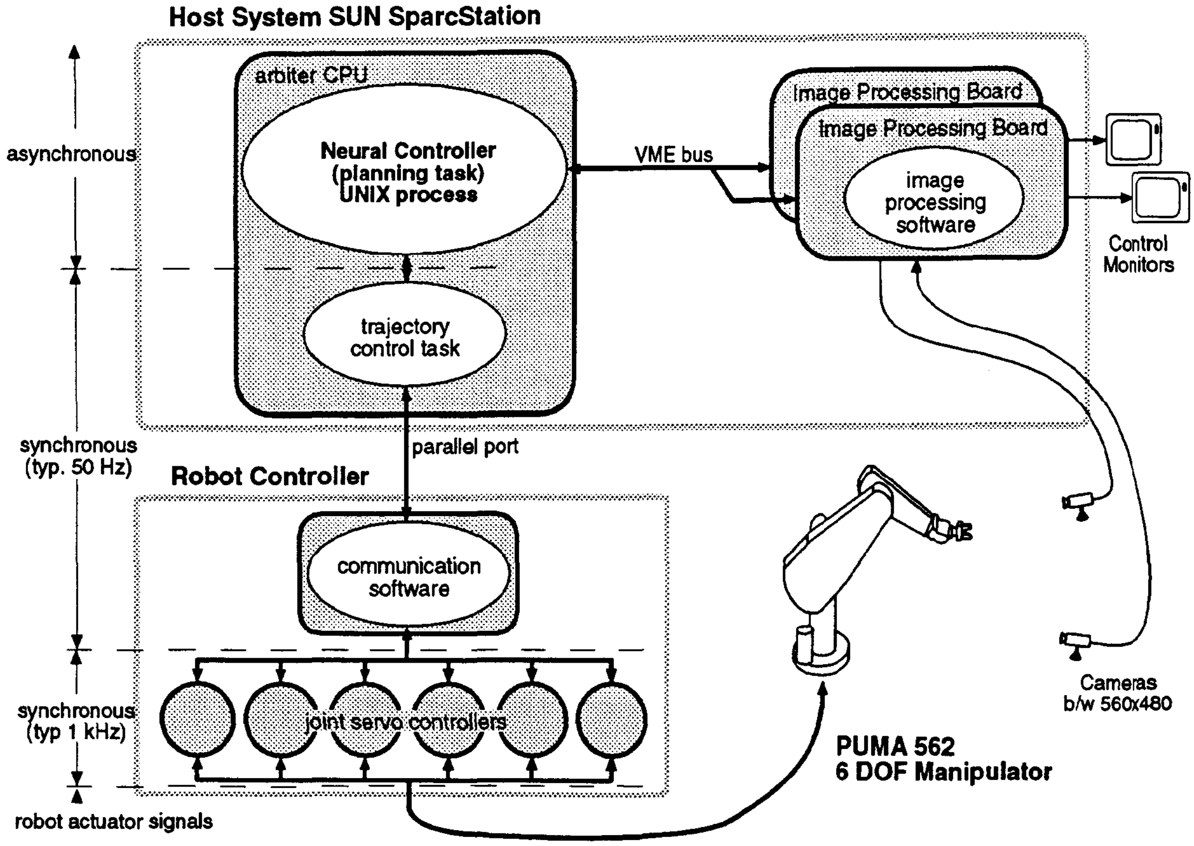

Fig. 6: The main hardware components of the robot system.

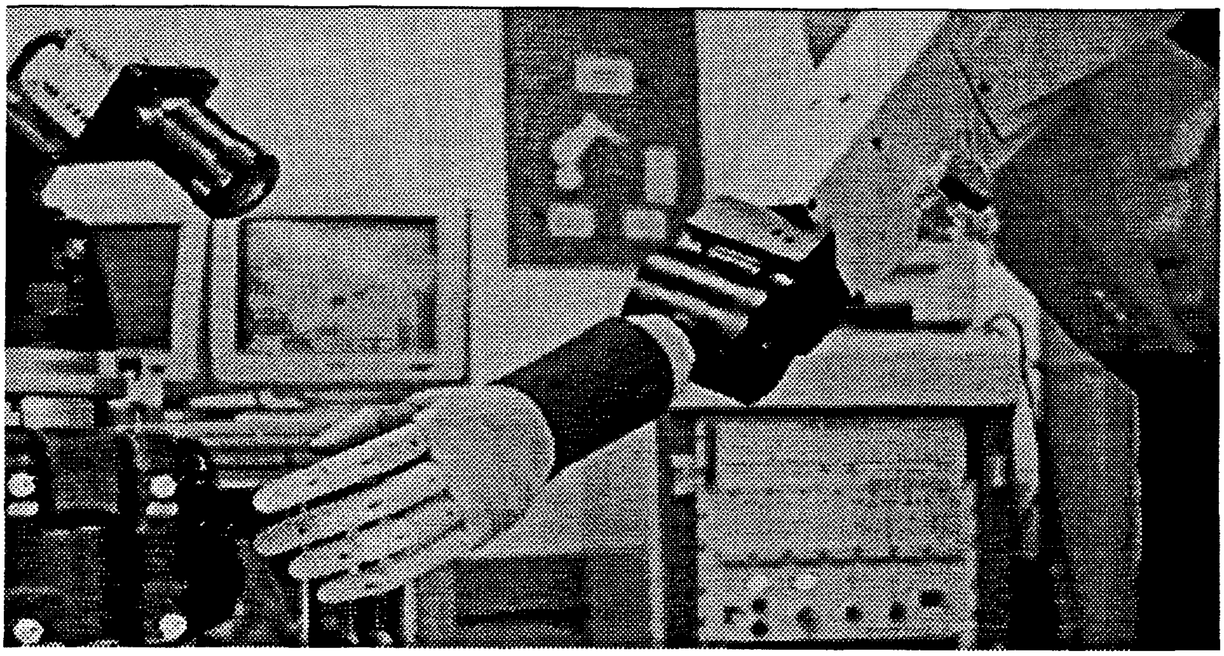

Fig. 7: View of the experimental set-up.

ordinary C program in a Unix environment. The program issues motion requests to the trajectory control level. The control task is executed periodically at a high priority and is responsible for reading input data, generating intermediate joint setpoints, and carrying out a "watchdog" function (collision avoidance).

For the neural network simulations and for graphics output we use several RS/6000 workstations which can communicate with the SUN and access the robot interface and the vision system via remote procedure calls.

Currently we use the system for testing the networks described in the previous sections with real images. To generate reproducible training images of known hand postures, we currently employ an articulated wooden model of a human hand. This model is mounted as "end effector" (Fig. 7) to the robot and can be moved under program control, thereby providing rather realistic images of hand shapes of known orientation.

6 Visually Guided Manipulator Movements

Recently we have integrated several of the previously described components into a system which allows to guide the motions of a robot manipulator by human hand movements that are performed in front of a camera. The system makes use of a so-far hand-optimized segmentation preprocessing stage that is implemented on the graphics processors of the ICS400-subsystem and that can perform segmentation and image centering in close-to-real-time. The resulting images form the input of a laplacian edge detection stage followed by a logarithmic intensity compression and a gaussian filtering stage. These preprocessing steps reduce the input dimensionality of the signal from 19200 (the number of pixels used) to 36. The resulting feature vector is fed to an LLM-network that has been trained with similar feature vectors obtained from hand images generated with the wooden hand model described in the previous section. Using these data, the network was trained to estimate the robot joint angles that gave rise to the observed hand orientation. The trained network can then be used to control the orientation of the robot manipulator such that it follows the orientation shown in novel hand images (Fig. 8). Even though trained with the wooden hand model only, the network is able to generalize to other hand shapes, such as the hand of a human operator, if these are not too different from the model. Currently we are about to improve this generalization ability of the network and to prepare it for controlling a hydraulically operated three-finger manipulator currently under development in the collaborating group of Prof. Pfeiffer at Technical University, Munich.

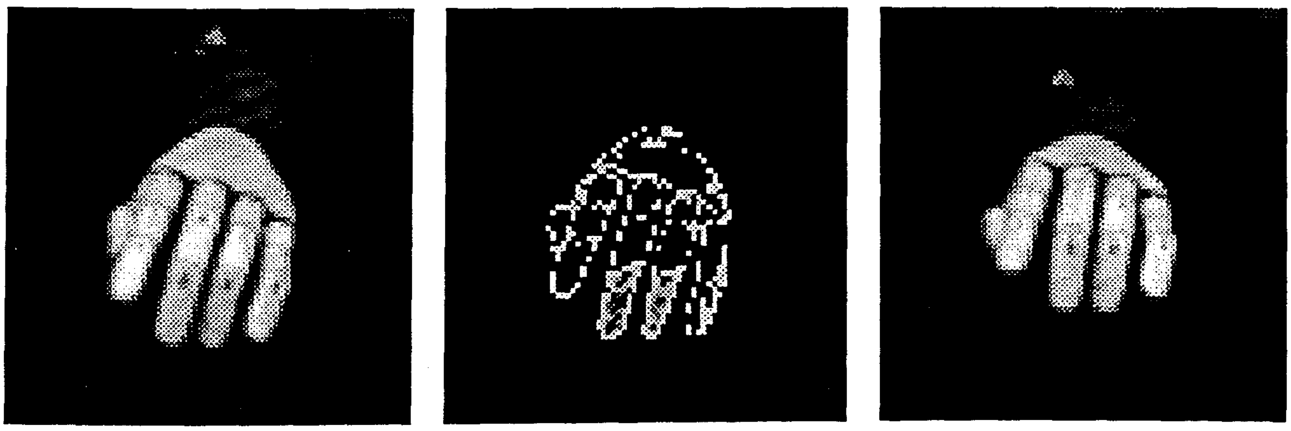

Fig. 8: Recognition of orientation of wooden hand replica from camera image (left: input image; center: after preprocessing; right: hand posture corresponding to recognized orientation).

7 Learning Basis Movements

The work described in the previous sections used supervised learning to achieve the desired performance. However, there will be many aspects of grasping movements that cannot be easily "taught" by supervised methods based on visual images alone. The reason is that important information, such as reaction and friction forces, are not available in a visual image. This makes it necessary

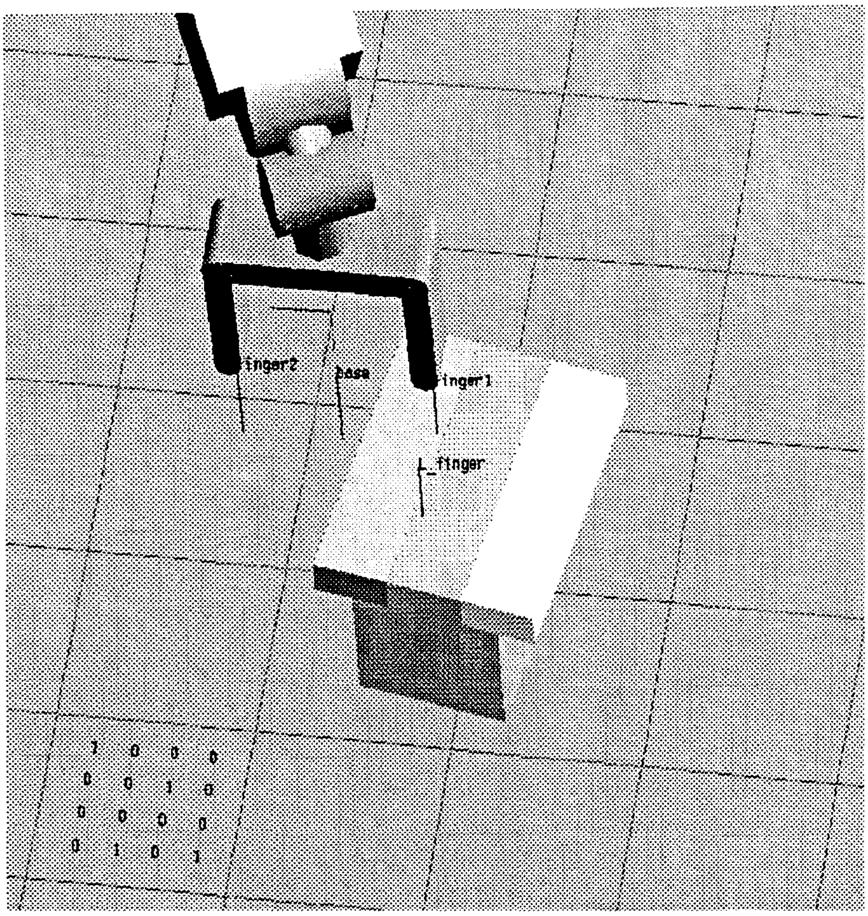

Fig. 9: Learning to grasp an object in computer simulation. Shown is a two-jaw gripper with four optical sensors (the matrix at bottom left encodes the position of the gripper with respect to the object).

to complement the present approach by unsupervised reinforcement learning methods that enable the robot to learn the use of such sensory information by trial and error [1]. However, a common problem of reinforcement learning methods is their low speed due to the enormously large search space in which learning takes place. This has limited existing demonstrations to rather small systems. Though important for their conceptual contribution, they are still far from what is needed to solve e.g. most tasks of interest in robotics. To overcome this difficulty, we have investigated strategies to decompose the full task into a set of simpler subtasks, such that these subtasks can be learnt separately and that the solutions for the subtasks can then serve as *basis elements* to span the solution space of the entire task [18]. Learning then can proceed as a *two-step process*: first, reinforcement learning is used to solve *each of the subtasks* individually. Then, a first guess for an approximated solution of an instance of the overall task is formed by a suitable *superposition of the solutions to the subtasks*. Finally reinforcement learning is used to further refine the approximate solution. We have explored the feasability of this approach using the classical maze-learning task, which is a widely used "benchmark problem" for studying reinforcement learning algorithms [16]. The results are reported in [18]. More recently, we have been able to demonstrate the feasability of this method for the harder task of learning to grasp a rectangular object, using a (simulated) two-jaw gripper that has only 4 optical sensors that are able to discriminate distinct parts of the object (Fig. 9). In this more complex situation, a phenomenon termed "perceptual aliasing" [19] occurs: as a consequence of the very restricted sensory information the state of the world cannot be uniquely recovered from the available sensor information alone. Previous approaches have found this to be a significant difficulty for ordinary reinforcement learning schemes; however, preliminary experiments indicate that our approach can cope with this difficulty and may even be able to exploit this phenomenon to improve its generalization ability for new situations.

8 Outlook

This paper has reported some of our work to develop neural network components to control robot grasping movements. So far, our strategy has mainly focused on the extraction of relevant control information, such as orientation and posture, from visual images and on the investigation of network architectures for combining several neural modules into a larger system. With the availability of an articulated multi-fingered hand in the near future, we will be able to use these components to control complex movements of the manipulator from visual images of a human hand. This seems to be a necessary and also realistic first step towards more intelligent robots that can be taught instead of programmed how to interact dextrously with their environment.

Acknowledgement

This work was supported by the German Ministry for Research and Technology (BMFT) under grant number ITN9104AO. Any responsibility for the content of this publication is with the authors.

References

[1] Barto A.G.,Sutton R.S.,Anderson C.W.(1983) Neuronlike elements that can solve difficult Learning control problems.
IEEE Trans. SMC-13:834-846.

[2] Brooks R (1986) A robust layered control system for a mobile robot. IEEE Journal of Robotic and Automation RA-2 (1), 14-23

[3] Fahlman, S.E., and Lebiere, C. (1989), "The Cascade–Correlation Learning Architecture", in *Advances in Neural Information Processing Systems 2 /*, ed. D.S. Touretzky, pp. 524–532.

[4] R.A. Jacobs, M.I. Jordan (1991): A competitive modular connectionist architecture. In: Advances in Neural Information Processing 3 (Eds. R.P.Lippmann, J.E. Moody, D.S. Touretzky) pp. 767-773

[5] Kohonen T. (1982) Self-organized Formation of Topologically Correct Feature Maps. Biol. Cybern. 43:59-69.

[6] Littmann E., Ritter H. (1992) Cascade LLM-Networks. Artificial Neural Networks 2 (I. Aleksander, J. Taylor Eds.) pp.253-257, Elsevier Science Publishers B.V. (North Holland)

[7] Littmann E., Ritter H. (1992) Generalization Abilities of Cascade Network Architectures. In: *Advances in Neural Information Processing Systems 5 /*, Denver

[8] Littmann E., Meyering A., Ritter H. (1992) Cascaded and Parallel Neural Network Architectures for Machine Vision – A Case Study. 14. DAGM Symposium, Dresden (S. Fuchs Ed.) pp. 81-87. Springer Verlag, Heidelberg, Berlin, New York.

[9] Littmann E., Meyering A., Walter J., Wengerek Th., Ritter H. (1992) Neural Networks for Robotics. In: Applications of Neural Networks (K. Schuster Ed.) pp. 79-103. VCH Verlag Weinheim.

[10] J. Lloyd, M. Parker, and R. McClain (1988) Extending the RCCL programming environment to multiple robots and processors. In *IEEE Conf. Robotics and Automat. (Philadelphia, Pa)*, pages 465–469, April 1988.

[11] Meyering, A., Ritter, H. (1992). Learning 3D-Shape Perception with Local Linear Maps. Proceedings of the International Joint Conference on Neural Networks '92, Vol. IV:432-436, Baltimore.

[12] Meyering, A., Ritter, H. (1992). Learning 3D-Hand Postures from Perspective Pixel Images. Proceedings of the International Conference on Artificial Neural Networks '92, Brighton.

[13] Moody, J., Darken, C. (1988). "Learning with Localized Receptive Fields", in *Proc. of the 1988 Connectionist Models Summer School*, Pittsburg, pp. 133-143, Morgan Kaufman Publishers, San Mateo, CA.

[14] Ritter, H. (1991), "Learning with the Self-organizing Map", in *Artificial Neural Networks I*, eds. T. Kohonen, K. Mäkisara, O. Simula, J. Kangas, pp. 357-364. Elsevier Science Publishers (North-Holland).

[15] Ritter, H., Martinetz, T., Schulten, K. (1992). *Neural Computation and Self-organizing Maps*, Addison-Wesley, Reading, MA.

[16] Sutton R.S.(1990) Integrated Architectures for Learning, Planning, and Reacting Based on Approximating Dynamic Programming. In: Proc. of the 7th Int. Conf. on Machine Learning, San Mateo, CA. Morgan Kauffman.

[17] Implementation of self-organizing neural networks for visuo-motor control of industrial robot. *IEEE Trans. Neural Networks* (in press) 1992.

[18] Reinforcement Learning and Subtasks. Artificial Neural Networks 2 (I. Alexander, J. Taylor Eds.) pp.259-262, Elsevier Science Publishers B.V. (North Holland).

[19] Whitehead S.D., Ballard D.H. (1990) Active Perception and Reinforcement Learning. In: Proc. of the 7th Int. Conf. on Machine Learning, San Mateo, Ca: Morgan Kauffman.