Analog neural networks and behavior based robotics*

Andreas Bühlmeier and Christoph Herwig

Zentrum für Kognitionswissenschaften

FB 3 Informatik, Universität Bremen,

Postfach 330 440, D-28334 Bremen, Germany

email: {andreas,herwig}@informatik.uni-bremen.de

http://www.informatik.uni-bremen.de/~{andreas,herwig}/{andreas,herwig}.html

1 Introduction

Designing autonomous systems is still a challenge to various disciplines involving engineering, computer science, artificial intelligence, control theory, learning theory and neuroscience. Natural and artificial agents share the problem that not all situations can be foreseen. Therefore, learning combined with built-in knowledge should give best results. Learning should be performed on-line, i.e. adaptation should be possible during the whole agent's lifetime. Very successful autonomous systems are biological beings. Since traditional AI did not reach its goals in the area of robotics, especially concerning the link to the world (sampled by sensors), more attention has been paid to "natural" information processing. Artificial neural networks are used in order to provide an adaptive coupling between sensors and actuators ("perception–action cycle"). In most cases however, neural networks are simulated sequentially on digital computers and the advantage of parallelism is lost. Also, biologically implausible learning rules like backpropagation are implemented, propagating signals in both directions.

The underlying structure of learning in natural systems seems to be different from commonly used neural networks. We propose here to emphasize the link between neurobiology and analog neural networks. Both, the natural nervous system and analog networks underly the same constraints: Keeping the wiring cost low and using connections unidirectionally. However, not all details of a solution have to be learned. At least an assessment function must be provided, if a system is to enhance its performance. Feature extraction and some behaviors may be fixed to reduce the amount of learning time.

This paper describes neural networks based on biological findings and, as an example, the application to tasks of a mobile robot. Some examples are tested in simulations others in physical systems. The aim is to develop neural networks suitable for hardware implementations and real-world problems. The notion "analog" is used here for continuous valued neurons. Spikes are not considered.

*In: Proc. Workshop 'Kognitive Robotik', B. Krieg-Brückner and Ch. Herwig (eds.), ZKW Bericht 3/95, Zentrum für Kognitionswissenschaften, Universität Bremen, März 1995.

In the first part of this paper the motivation of analog neural networks is described and implementations of this approach are presented. We will then discuss a possible application area: the (semi-) autonomous wheelchair. Finally, we outline an implementation of interactive object selection and tracking. All approaches presented in this paper share the property of being at the signal level. No explicit model of the world is used.

2 Background

2.1 Analog neural networks

Carver Mead (1989) proposed the notion of neuromorphic information processing, i.e. to study the structure of natural networks and to transform it to VLSI hardware implementation. Sensor signals are directly fed to electronic components (resistors, capacitors, transistors etc.) and are processed due to their physical properties. By this, otherwise complex calculations become very simple. One example of this approach is the "Silicon Retina".

However, for learning systems it is not easy to follow this approach straightforward. Especially, the storage of weights is difficult. One possibility to store a weight (load) are *floating gate* transistors, a technology used in EEPROMs (Electrically Erasable Programmable Read Only Memories). However, the number of write cycles is limited; the floating gates are aging. Another disadvantage is a limited accuracy, which depends on the implementation. Backpropagation networks can not be easily implemented in analog hardware, because these networks need an accuracy of about 10 bit and the weights are used bidirectionally. Therefore, a deeper understanding of how natural systems adapt behavior should be considered to develop neuromorphic learning hardware. The aim is to connect only few cheap chips to sensors and actors of a robot system and to obtain a very fast performance.

The main advantages of this approach may be summarized:

- True coincidence of signals is possible.
- Real-time performance is inherent. Response delays do not depend on the number of incoming signals (e.g., senors, weighted outputs of other neurons).
- Physical properties of electronic components can be exploited to perform otherwise expensive computations (e.g. exponential function, short-term memory).
- Higher robustness against electro-magnetic noise compared to a standard software solution is achieved.
- Similar restrictions as in natural systems provide a better link to neuroscience.

Disadvantages of an analog parallel implementation compared to digital sequential computation are:

- lower flexibility,
- lower accuracy (depending on the implementation) and

- a pure analog technique is difficult to miniaturize (e.g. capacitors).

It seems most promising to combine both approaches. Therefore, many implementations use AD and DA converters and standard RAM (e.g. off-chip) for weight storage. Learning systems often need a short-term-memory. This can be simply achieved using low-pass filters (resistor-capacitor combination) and has been used in experiments reported in section 3.1. However, in a VLSI design only very small capacitors are available. Therefore, an early hybrid approach at the STM-level might be more appropriate and is currently investigated in collaboration with Dortmund University.

2.2 Behavior and instinct-based robotics

Brooks (1986, see also Brooks & Stein, 1993) introduced the approach of behavior based robotics. Sensor signals are transformed into motor signals by behavior modules, each of which serves a specific task. Several behavior modules are connected hierarchically. As in neural networks, the signal-level is not left. However, the basic approach did not include adaptivity. Mahadevan and Connell (1991) and Mataric (see, e.g., Mataric, 1994) and others later combined Brook's approach with reinforcement learning and Nehmzow et al. (1989) proposed a related architecture based on instincts to combine predefined knowledge and learning.

The advantages of behavior based robotics may be summarized:

- No explicit world-model is required.
- Fast response time is obtained by direct, parallel asynchronous, sensor-to-motor coupling
- Robust and opportunistic performance.
- Lower levels remain unchanged, if more complex behaviors are added.

Advantages of classical approaches:

- An explicit world model is a good user interface.
- Explicit rules enable the fulfillment of safety requirements.

Since in neural networks and behavior based robot controllers no explicit representation is required, both can be combined easily. One example how biologically plausible learning and built-in knowledge can be combined is conditioning.

3 Applying neuromorphic control

3.1 Conditioning

Conditioning is one principle of animal learning, which is well understood and has been applied to mobile robot control several times (see Bühlmeier & Manteuffel, 1995).

One basic paradigm is classical conditioning. An unconditioned response (UR), like closing the eye-lid when an air puff (unconditioned stimulus US) occurs can be associated to a conditioned stimulus (CS), which did not cause any response before. This can be achieved by presenting repeatedly the CS-signal (e.g. a tone), before the US. After a couple of trials, the response, like closing the eye-lid, can be triggered by the conditioned stimulus alone, the effect of the unconditioned stimulus is anticipated.

Introducing two conditioned stimuli other effects can be observed. If a stimulus (CS_A) was associated by classical conditioning and is then presented simultaneously with another stimulus (CS_B), the second CS may not become associated, it can be blocked. Many effects have been reported, some of which can be attached to specific brain regions. One important brain region is the *hippocampus*. It is involved in *attentional* effects, e.g. blocking.

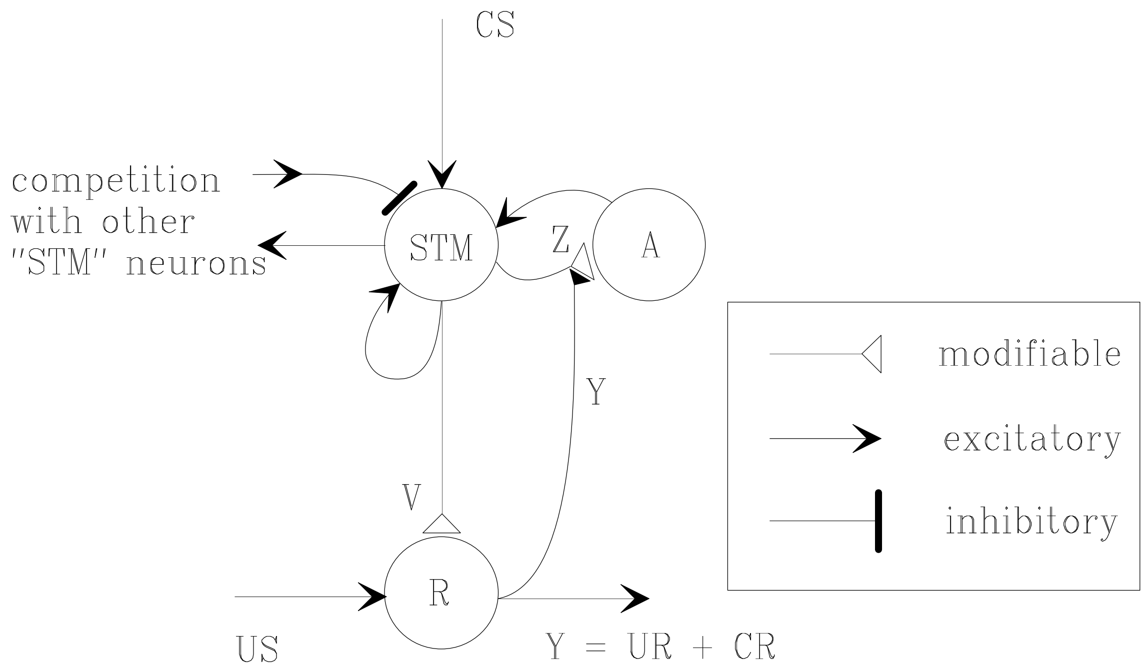


Figure 1: Schmajuck and DiCarlo's (1991) approach to hippocampal function. Representations of stimuli compete among each other by lateral inhibition. Adaptive synapses ("Z") mediate positive feedback of sensory representations.

A model of attentional effects (Schmajuck & DiCarlo, 1991) was extended and tested in a mobile robot. In the network, shown in Fig. 1, learning takes place at two different sites. Classical conditioning is performed by "V" synapses and competition of sensory representation is mediated by "Z" weights.

The robot's reflexes were implemented for safety reasons, the robot retreated to the left when it hit something on the right and vice versa. These collisions were indicated by tactile sensors, which formed US-signals, the retreat movements can be regarded as unconditioned responses. (The reflexes can be interpreted as very basic behaviors modules.) The robot, shown in Fig. 2 was also equipped with a small camera containing six

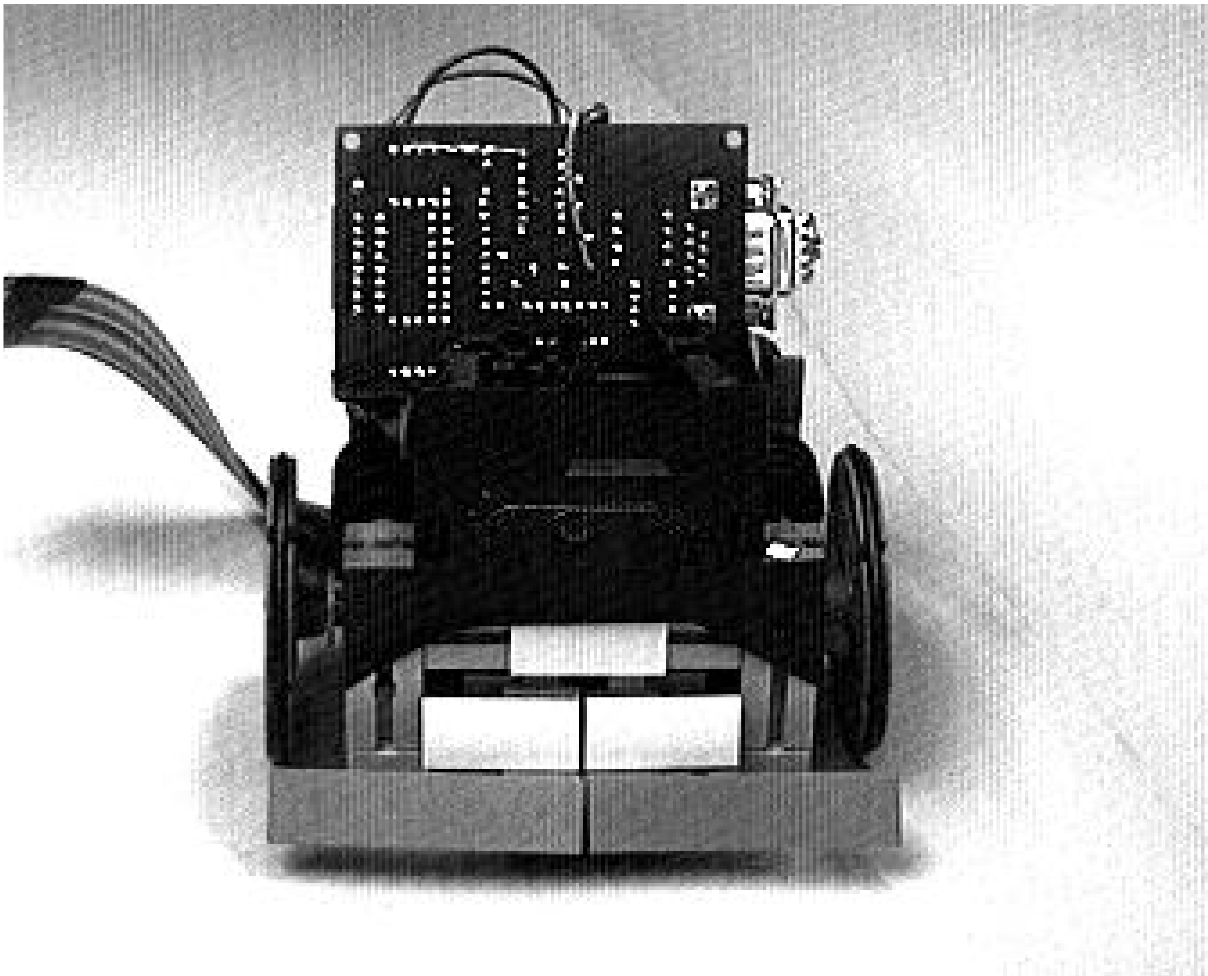


Figure 2: The mobile robot is constructed from "Fisher-Technik" parts.

photoreceptors. The visual information was preprocessed using edge detection and elementary movement detectors. Proprioception was also provided to the robot's network. Visual information and proprioception served as CS-signals, however, the naive robot did not know how to use them.

After some collisions (about 30 sec.) the robot had learned to avoid obstacles by anticipating the reflex-like retreat movements. A combination of visual and proprioceptive signals was successfully associated to response neurons (labeled "R" in Fig. 1).

It was of special interest whether the attentional mechanism (adaptation of "Z" synapses) did enhance the robot's performance. Therefore, a simple test was constructed. The robot was placed in two different positions in front of the obstacle. When the robot had performed a turn of 90 degrees or passed the obstacle, it was stopped and set to the other starting position. The performance was assessed computing the ratio between the time the robot followed a target direction and the time collisions occurred.

The result of 16 runs of 100 seconds each demonstrated, that learning to pay more attention to *significant* stimuli enhances the robot's performance by 30 % in this setup. The network in this case was simulated by a PC (Bühlmeier, 1994a-c).

To test the basic principle of association, we implemented a similar experiment with hardware neurons and a wheelchair. Figure 7 displays the wheelchair; in the background the hardware neurons can be seen. These neurons were mainly designed for teaching (Manteuffel, 1992), but will be miniaturized for other applications in collaboration with Dortmund University.

Again, tactile senors are mounted on the front. Instead of a "camera" we used sonar. A signal was given, when an echo delay was beneath a threshold (an obstacle was in a certain range). The reflex was now to decrease velocity to zero, if a collision, indicated by tactile sensors, occurred. After some collisions the sonar signal was associated and sufficient to decrease velocity.

In artificial physical systems, protections are always implemented. These can be regarded as unchanging knowledge. Triggering protective actions can be anticipated by the principle of conditioning. Classical conditioning may be used in more complete architectures as a basis. The example of the adaptive short-term memory demonstrated how plastic preprocessing can be achieved, driven rather by actions than by sensor signals. Many extensions of the network are possible, one of which is the inclusion of hidden inhibitory neurons (Bühlmeier, 1995).

3.2 Adaptive self-localization

Animals use different modalities to estimate their location in an environment. This is performed by exploration and learning. However, associations to places may not be unequivocal. Therefore, it is necessary to possess a system being able to merge associations to different locations into a single vote (Wan et al., 1994). Since rats with impaired hippocampus are able to perform simple route-following, it is assumed that for some tasks other brain structures are sufficient. Navigation with and without a map might be two interacting strategies (Burgess et al., 1995).

The aim of the work presented here is to enable a robot to adaptively use different sensor modalities to estimate its current position, which may serve as a basis for path-planning.

Figure 3 depicts a simple version of the proposed network. A central part of the model is the lateral inhibition layer ("F", fusion layer) consisting of many neurons each of which represents a part of the environment. After some learning occurred, sensory associations activate different locations on the "F" layer. The combination of different modalities generally contains less error than a single one. After having calculated its activation, this layer undergoes a lateral inhibition process, similar to Amari's networks (Amari, 1980). By this, activations close to each other are merged to a weighted average location and estimations with a large distance to a cluster do not influence the result. After fluctuations of activations due to the lateral inhibition process are below a limit, the associative weights are updated and the odometer is reset, i.e. further integration is added to the current estimation. By this, occasional odometer errors can be compensated (Bühlmeier & Herrling, 1995). Moreover, the difference between odometer based position estimation and the result of the fusion may be used to adapt, e.g., an odometer gain.

Figure 4 shows a simulated robot on a one-dimensional track limited by two walls. The robot moves from one wall to the other; it changes its direction after having sensed

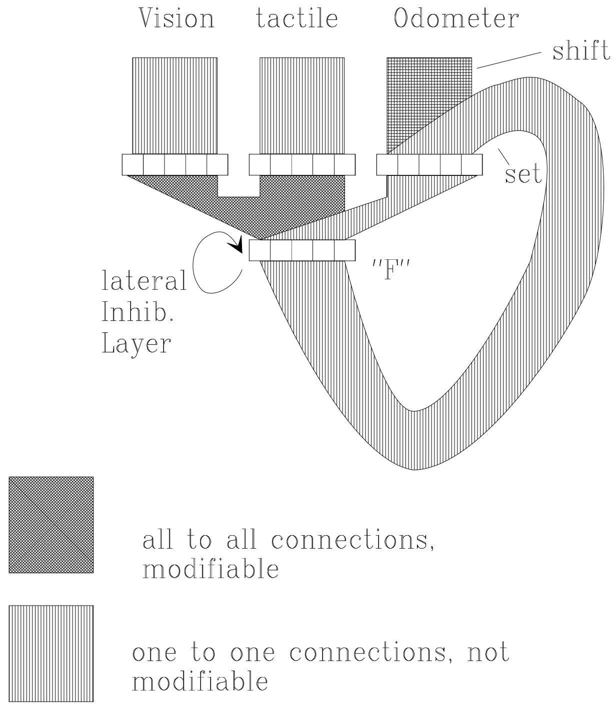


Figure 3: Proposed neural network for self-localization.

a collision by its tactile sensors. We assumed different odometer errors in each direction in this example. The robot underestimates the path to the left and has correct odometry moving to the right wall. (The error might occur when a robot moves on a carpet.) After the first wall approach (scene (1) in Fig. 4) the error equals the odometry error when moving to the left. Since the tactile sensor is touched at the wall, it becomes associated to the location, which was activated by the odometry. Forming the association is depicted by the thick line. In the next part (2) the agent locates at the right wall. The odometer is assumed to be without error in this direction. However, the error of the first scene is still present. Here, the second tactile sensor becomes associated. Scene (3) shows a more interesting case: odometry and associations activate the fusion layer at different locations. By lateral inhibition, the activations merge and the resulting location's node strengthens its associations to the tactile sensors and is used for further path integration. Scenes (5) and (6) display the procedure for the right wall. The error never reaches zero, but step by step the error increments are decreased and we did not assume which sensors might deliver faulty signals and which work correctly. Without using associations and the fusion process, the error would accumulate (multiplied with approaches).

The properties of the network proposed here were verified in simulations and in a physical robot. First of all we simulated the setup described in the previous section. A chain of 100 neurons was used to simulate the fusion layer. Figure 5 shows the robot's self-localization error over wall contacts. The error's peaks correspond to the left side, since the error is accumulated when moving left. When the odometer layer is reset, the difference between the position estimated by the "F"-layer and the odometer information

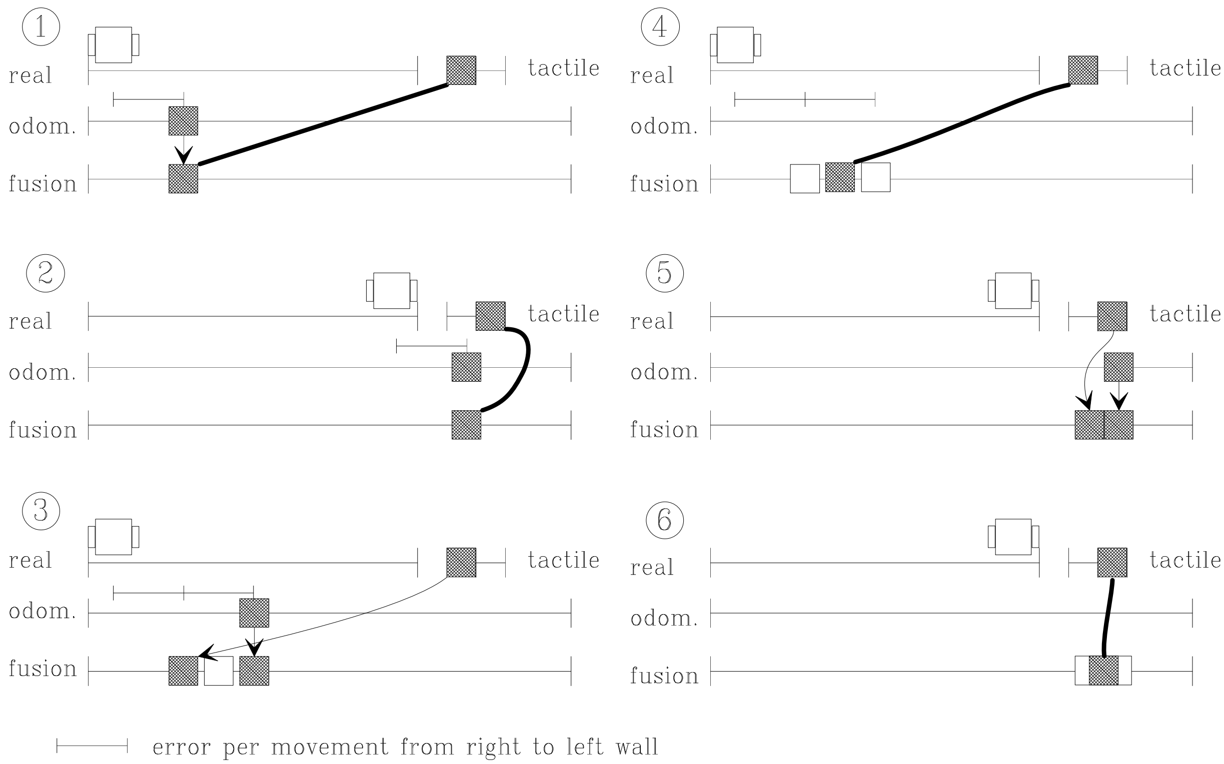


Figure 4: Six scenes of a robot on a one-dimensional track. The first line depicts the robot, its real position (left line) and its two tactile sensors (right part). Position estimations by odomter and fusion layer are shown below (see text for more details).

can be used to adapt a gain parameter. Here, we used different gains for left and right movement. Adapting the gains by gradient descent, the average error is cut in half. Of course, the error might be reduced to zero, but this would imply assumptions about which sensor delivers more correct information, which is not adequate in many situations. This approach has also been tested successfully using the 3D robot simulator `SimRobot` (see Fig. 6).

4 The semi-autonomous wheelchair

To focus some of the research activities in our doctoral study programme (see Krieg-Brückner, 1994), we considered the task of an (semi-)autonomous wheelchair. Meanwhile, the (semi-)autonomous wheelchair has become a central part of the studentship project SAUS (Sensomotorik AUtonomer Systeme). Experiments are conducted on a normal wheelchair, on which we added odometer, tactile and sonar sensors (see Fig. 7). Sensor data acquisition and preprocessing is implemented in CAN-bus nodes (CAN =Controller Area Network).

Applying "intelligent control" to a non-holonomic vehicle is a great challenge but it

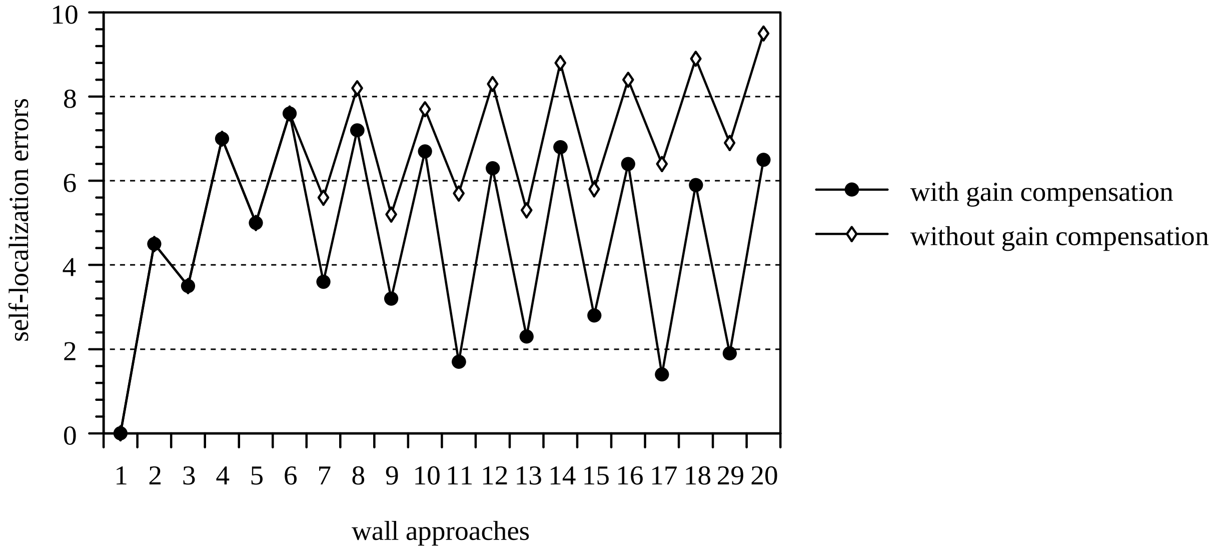


Figure 5: The Robot's self-localization error on a linear track. Without using associations and the fusing procedure, the accumulated error would equal 40 units after 20 wall approaches in this example.

is very time consuming to integrate sensors and computers. Therefore, a proposal was submitted to purchase a "plug and play" platform, like, for example the RWI B21 mobile robot of Real World Interface, Inc. By this, we hope to obtain verifications of theoretical work and simulations much faster. Results on the RWI platform might be implemented later on the wheelchair. Nevertheless, the wheelchair is actually used by SAUS-project students and for testing hardware neurons.

With the background of the wheelchair we are able to discuss the relevance of research results more concretely. By this, we may be able to demonstrate some capabilities of intelligent systems that might help handicapped persons to enhance their independence. Of course, many tasks and problems are the same in this particular application and other areas and can be transfered from one to another. The wheelchair is one example of a service robot.

Experiments are conducted in both, simulations (see Fig. 6) and hardware (see Fig. 7). Neural networks are implemented in software and hardware. We plan to add a robot arm to reach a high degree of functionality as discussed in the next section.

4.1 Aimed functionality

The functionality we wish to reach is threefold:

- *Task-driven navigation*: the user selects tasks (e.g. "wash hands"), the wheel chair plans appropriate movements and executes them.

- *Task-driven manipulation*: the user interactively selects objects on a screen (e.g. a book on a bookshelf) and the object is made accessible by an arm mounted on the wheelchair (elaborated upon below).

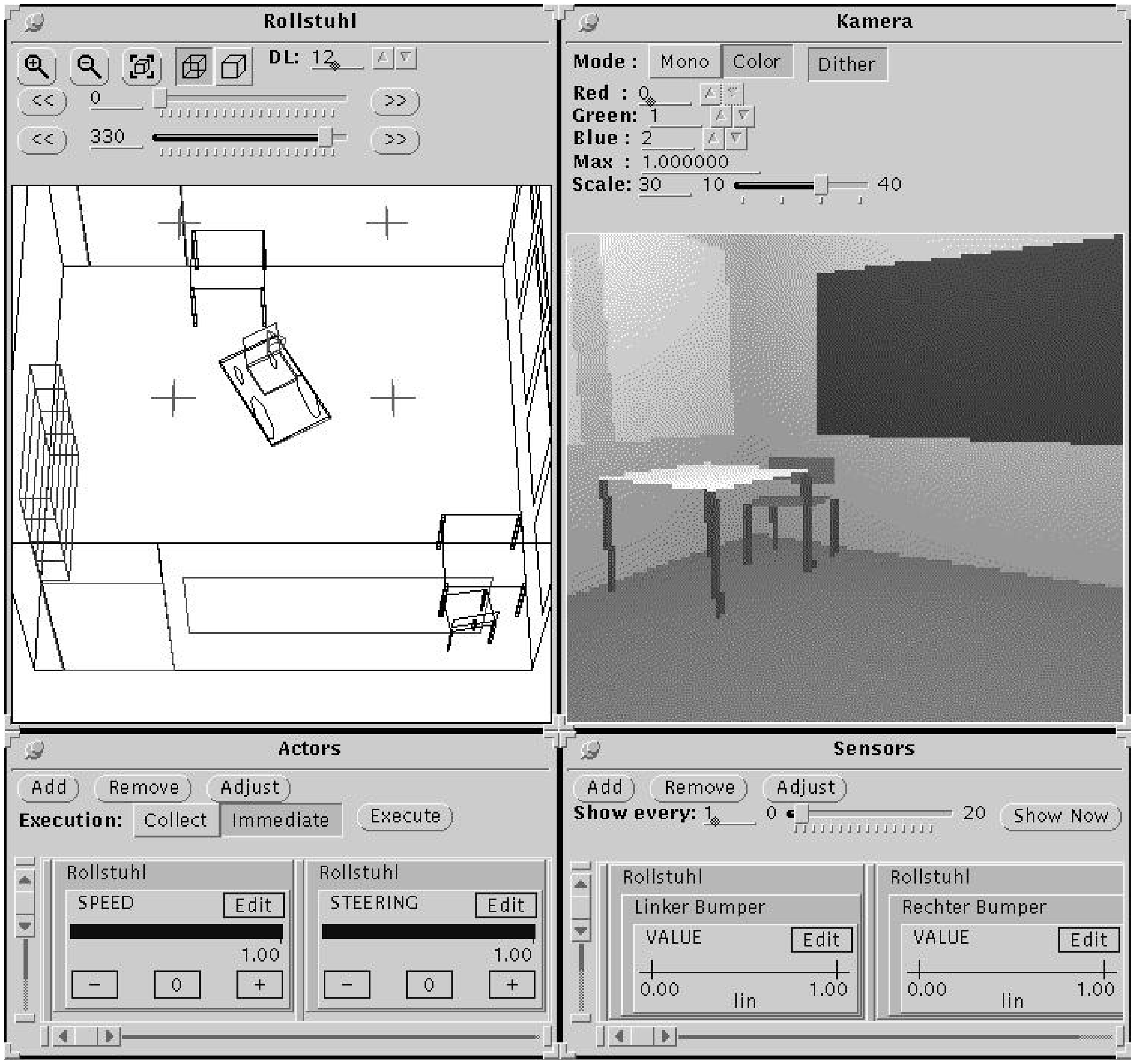


Figure 6: Simulating a wheelchair with `SimRobot` (Siems, Herwig, Röfer; 1994).

- *Autonomous task execution*, combining navigation and manipulation: door opening, putting paper in a printer, automatic runaway (fire-alarm).

A summary of all research activities in the graduate study programme, many of which can be applied to the wheelchair, is given by (Krieg-Brückner, 1994). This research includes the consideration of risk in reinforcement learning (Heger, 1994), modularization of neural networks, neural networks for control applications (Peinemann, 1994), computer vision (Herwig & Carmesin, 1994; Henkel 1994a, 1994b) and navigation (Wittmann 1994).

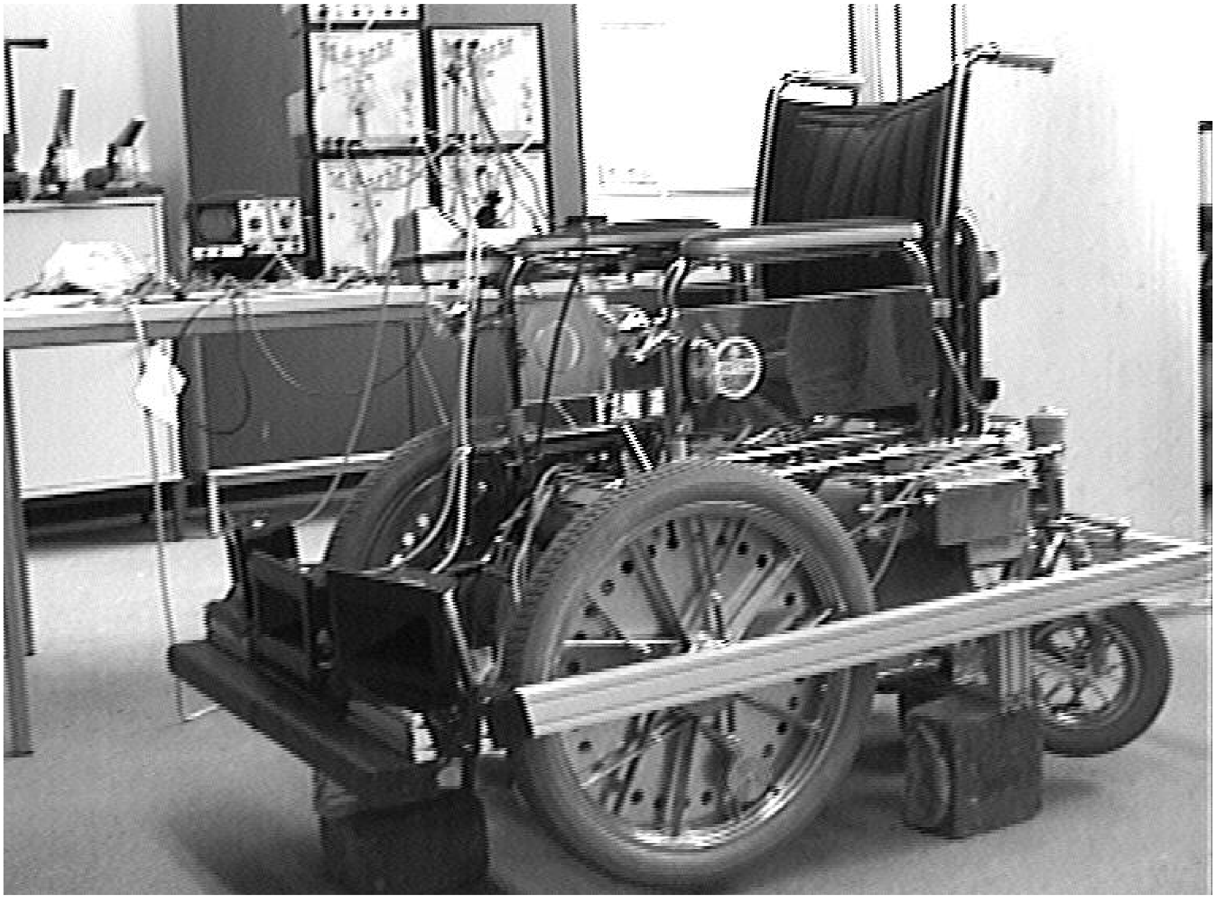


Figure 7: Wheelchair experimental setup. Hardware implementation of analog neural networks can be seen in the background.

4.2 Visual preprocessing for task driven manipulation

4.2.1 Motivation and approach

Aside from navigation, the task–driven manipulation plays an equally important role for the autonomous "every day survival" in man made environments. For this purpose, wheelchairs may be equipped with multijoint robot arms which enable the handicapped to enlarge their action radius. Research in tight collaboration with the handicapped is underway[1] in order to evaluate the needs of the user in conjunction with the technological possibilities. In the following, we elaborate upon a computer vision approach to support the coarse positioning of the robot arm relative to user selected objects. It is characterized by real time and data reduction capability. We emphasize the

- interactive man machine interface and the
- model–free object selection.

Fig. 8 provides a box diagram of the approach, which may be cross referenced during the following explanation.

[1] Forschungsinstitut Technologie – Behindertenhilfe (FTB) der Ev. Stiftung Volmarstein.

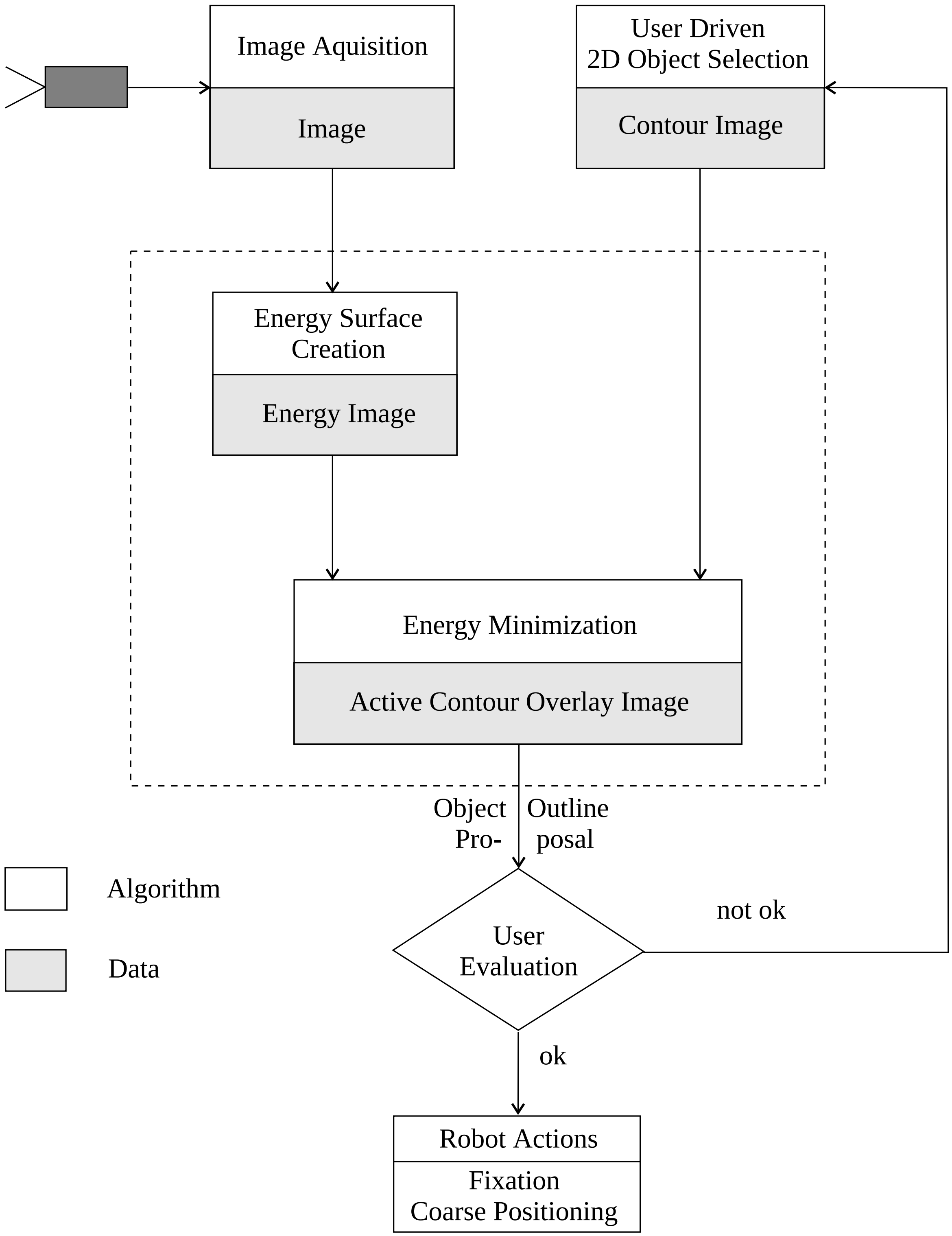


Figure 8: Interactive object selection process.

Scenario. The wheelchair is equipped with a multijoint robot arm such as the MANUS[2] version. A camera will be mounted close to the manipulator such that at least three degrees of freedom (DOF) are available for positioning the camera in the 3D workspace of the manipulator. The user views the camera image on a monitor mounted onto the wheelchair.

Interactive Object Selection. Depending on the abilities, the user must be enabled to select objects of choice on the screen. The output of the selection process is a highlighted contour around the object. This coarse object information is input to a refined object selection routine based on *active contour models* (see below): The system proposes an exact contour around the prospective object. The user may then revise the selection and the interaction continues until the user is satisfied with the choice. This interaction process exhibits several desirable features:

- The user is an integral part of the process.
- Initial contour selection can be tailored to the abilities of the handicapped, e.g. positioning of (variable size) box frames, touch screen input, forehead laser pointer input.
- User deficiencies in object selection are compensated through an automated improving of object outlining via *active contours*.
- Algorithmic maloperation, as is common in many real-world computer vison applications, is compensated through the interaction process.

Coarse Positioning. After interactive object selection, the arm begins to move towards the desired object. During movement, the contour is active and continuously locks onto the desired object which can be constantly centered in the cameras view. The module finishes after positioning the camera and the manipulator close to the object.

4.2.2 Active contour model

The core of the object selection process constitutes the system *object outline proposal routine*. Based on an initial contour in the image provided by the user, its task is to modify the contour such that it "moves" to the outline of the desired object. The algorithmic method of choice here is the *active contour energy minimization*.

Active contours (Kass et al., 1988) are energy minimizing curves which move by internal and image forces towards image features such as lines and edges. Traditionally, they have been used for tracking deformable objects such as biological cells in microscopic images since any ridgid body assumptions fail for these cases. Recently, they have become more popular in the robotic visual tracking domains (Blake et al., 1992) (Denzler & Niemann, 1994). Here they allow model free 3D object tracking in real time.

An active contour is parametrically described as

$$\underline{v}(s) = (x(s), y(s)), s \in \{0, ..., N-1\}$$

[2]Trademark Exact Dynamics (Zevenaar / Niederlande)

where s denotes a discrete metric 1D location along the contour and (x, y) specifies a 2D location within the image plane. The static energy equation

$$E(\underline{v}) = E_{int}(\underline{v}) + E_{im}(\underline{v})$$

is composed of *internal* and *image* forces. The internal forces are typically specified as

$$E_{int}(\underline{v}) = \sum_{0}^{N-1} \frac{1}{2} \left(\alpha(s) ||\underline{v}_s(s)||^2 + \beta(s) ||\underline{v}_{ss}(s)||^2 \right) ds$$

Two parameters characterize the energy term: α controls the contour elongation and equal distance of neighboring contour points. If it is set to zero, discontinuities in terms of image plane locations are permitted and not punished with high energy values. β controls the contour shape. If it is set to zero, the contour is permitted to turn sharply at edges, whereas otherwise it attempts smooth curvature.

The image energy is important as a link to the underlying data. It specifies the affinity of the contour towards image features. Here we give two common examples: the strive towards bright (-) or dark (+) image regions:

$$E_{im}(\underline{v}) = \sum_{0}^{N-1} \pm c(s) \left(G_\sigma * I(x, y) \right) ds$$

where $I(x, y)$ denotes the image intensity, $*$ denotes the convolution operator and G_σ specifies the gaussian function for low pass filtering, and towards high intensity variations, i.e. edges:

$$E_{im}(\underline{v}) = \sum_{0}^{N-1} -c(s) |\underline{\nabla} \left(G_\sigma * I(x, y) \right)| ds$$

where $\underline{\nabla} = (\frac{\partial}{\partial x}, \frac{\partial}{\partial y})^T$ denotes a spatial derivative operator.

The energy may be minimized by various strategies, including variational calculus (Kass et al., 1988), dynamic programming (Amini et al., 1990), finite element techniques (Cohen & Cohen, 1991) or straightforward greedy techniques (Williams & Shah, 1990).

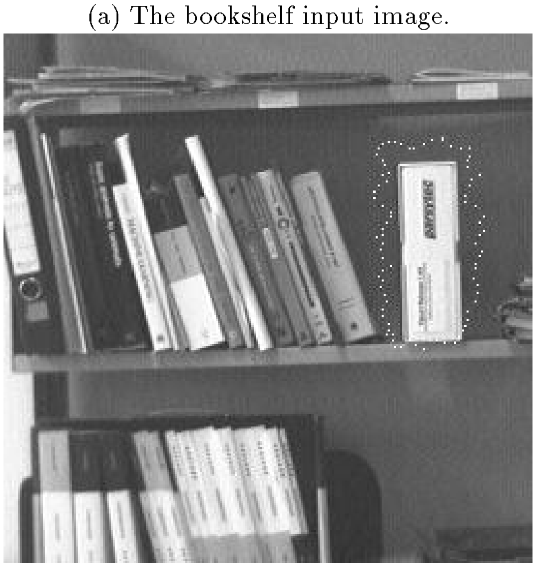
4.2.3 Experiments

A prototypical operational example is to take a book off a shelf. The goal of the modules proposed here is to position the manipulator close to the desired book. The energy minimization was programmed according to the *greedy algorithm* (Williams & Shah, 1990). Fig. 9a shows the image of a bookshelf and in Fig. 9b the image force energy function (inverse of edge detection on low pass filtered image) is displayed. Fig. 9c shows the discrete initial contour (possibly a sampled version of one provided by the user) and in Fig. 9d the dynamic minimization process is visualized. The final contour outlines the selected book.

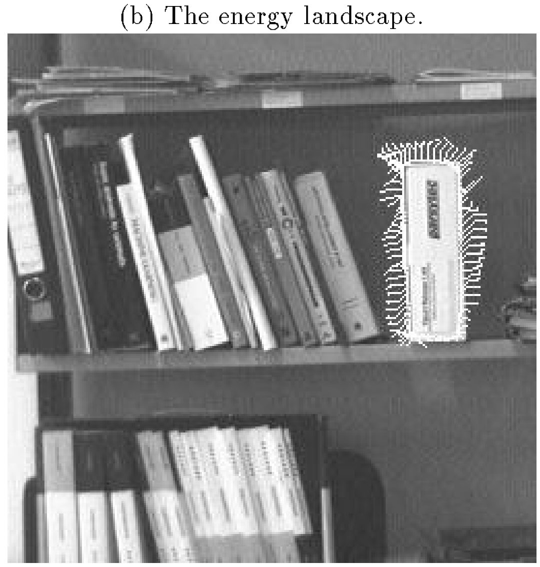
One of the problems with the approach is the dependence upon good initialization (Denzler & Niemann, 1994). Our interactive object selection process however guarantees good initialization depending upon user capabilities. The mentioned energy function leads to convergence towards intensity discontinuities. Often, these features are not equal

(a) The bookshelf input image.

(b) The energy landscape.



(c) The image with the initial active contour.



(d) Time integrated image of the energy minimization process.

Figure 9: Example application of active contour minimization.

to object boundaries, which poses a problem for our application. It has been an issue in computer vision ever since to attempt to bridge the representational gap in order to recognize the distinction and achieve a fusion. The straightforward way to tailor the algorithm towards specific applications is to modify the energy function. We propose here to include depth information (provided for example from image sequence flow field modules (Herwig & Carmesin, 1995) or by high resolution and registered range devices) as an energy term in order to better distinguish object boundaries from simple intensity or texture changes. This constitutes an area of future research.

5 Future work and conclusion

This paper sketched different work in the context of the semi-autonomous wheelchair. Future work will include the VLSI-design of analog neural networks and the integration of local and global navigation and path-planning. The aspect of module interaction will deserve special attention. We will investigate how knowledge gathered by one module can be transfered to another. We further plan to equip the wheelchair with active color cameras for navigational tasks. Functional aspects for the near future will be the autonomous steering through a door. The intention is to use less a priori knowledge than usual. As in the adaptive obstacle avoidance example, we will implement only unchanging knowledge, i.e., protections and a global signal for the assessment of the current situation. Which sensors are to be used to solve the problem is to be learned. We think this approach can be transfered more easily to different implementations compared to single and fixed programmed solutions. However, more complex tasks like tracking an object need more a-priori knowledge. The approach of active contours is one example how preprocessing may achieve a drastic reduction of sensor data, which may then be more easily combined with learning. Since the active contours do not need an explicit model, they fit into the framework of behavior based robotics and (analog) neural networks. The problem of scaling up to more sophisticated tasks is an important aspect in the area of neural networks and robotics and we intend to investigate this problem in the context of mobile manipulation.

Acknowledgements

This work was supported by the Deutsche Forschungsgemeinschaft and the Freie Hansestadt Bremen through the Graduiertenkolleg "Raumorientierung und Handlungsorganisation autonomer Systeme". The authors further thank Prof. Manteuffel and Prof. Krieg-Brückner for discussions and helpful comments.

6 References

Amari S.I. (1980) Topographic organization of nerve fields. *Bulletin of Mathematical Biology*, Vol. 42, (1980) 339-364. Pergamon Press.

Amini A.A., Weymouth T.E., Jain R.C. (1990) Using dynamic programming for solving variational problems in vision. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, Vol. 12, No. 9, 855–867.

Blake A., Zisserman A., Cipolla R.C. (1992) Dynamic contours: real-time active splines. In *Active Vision* (ed. A. Blake & A. Yuille). MIT Press.

Brooks R.A. (1986) Achieving artificial intelligence through building robots. *MIT AI Memo* No. 899, Massachusetts Institute of Technology.

Brooks R. A., Stein L.A. (1993) Building brains for bodies. *AI Memo* No. 1439, Massachusetts Institute of Technology.

Bühlmeier A. (1994a) Conditioning and robot control. In: M. Marinaro, P. Morasso (Eds.): *Proc. Int. Conf. Artificial Neural Networks* Springer Verlag. 1283–1286.

Bühlmeier A. (1994b) From behavioral studies to sensorfusion. Proc. From Perception to Action Conference (PerAc 94). IEEE Computer Society Press. 404–407.

Bühlmeier A. (1994c) Conditioning of a robot. *ZKW–Bericht Nr. 9/94.* Krieg–Brückner, Roth, Schwegler (Eds.). Zentrum für Kognitionswissenschaften, Universität Bremen.

Bühlmeier A. (1995) Negative patterning in classical and operant conditioning: a neural network model of the cerebellum. In N. Elsner, M. Heisenberg (Eds.): *Proceedings of the 23rd Göttingen Neurobiology Conference.*

Bühlmeier A., Herrling U. (1995) A neural network for adaptive self-localization. ECAL95, Abstract Book, (to appear).

Bühlmeier A., Manteuffel G. (1995) Operant conditioning in robots. In: P. P. van der Smagt (Ed.):*Neural systems for robotics.* Ablex (to appear).

Burgess N., Recce M., O'Keefe J. (1995) Hippocampus-spatial models: in: M.A. Arbib (Ed.): The Handbook of Brain Theory and Neural Networks. Bradford Books/MIT Press, to be published.

Cohen L.D., Cohen I. (1991) Using a finite element method for active contour models and 3D reconstruction from cross sections. *Artificial Intelligence and Computer Vision.* Feldman Y.A., Bruckstein A. (Eds.). Elsevier Science Publisher, 237–247.

Denzler J., Niemann H. (1994) A Two Stage Real–Time Object Tracking System. *Journal of Computing and Information Technology* – CIT 2, 3, 211–221.

Heger M. (1994) Risk and reinforcement learning: Concepts and dynamic programming. *ZKW–Bericht Nr. 8/94.* Krieg–Brückner, Roth, Schwegler (Eds.). Zentrum für Kognitionswissenschaften, Universität Bremen.

Henkel R. (1994a) Segmentation with synchronizing neural oscillators. *ZKW–Bericht Nr. 4/94.* Krieg–Brückner, Roth, Schwegler (Eds.). Zentrum für Kognitionswissenschaften, Universität Bremen.

Henkel R. (1994b) Hierarchical calculation of 3d-Structure. *ZKW–Bericht Nr. 5/94.* Krieg–Brückner, Roth, Schwegler (Eds.). Zentrum für Kognitionswissenschaften, Universität Bremen.

Herwig Ch., Carmesin H.-O. (1995) Direct convergent egomotion determination.*ZKW–*

Bericht Nr. 1/95. Krieg–Brückner, Roth, Schwegler (Eds.). Zentrum für Kognitionswissenschaften, Universität Bremen.

Kass M., Witkin A., Terzopoulos D. (1988) Snakes: Active Contour Models. *Int. Journal of Computer Vision.* Kluwer Academic Publishers. 321–331.

Krieg-Brückner (Ed.) (1994) Graduiertenkolleg Raumorientierung und Handlungsorganisation Autonomer Systeme an der Universität Bremen, Berichte der Arbeitsgruppen. *ZKW–Bericht Nr. 10/94.* Zentrum für Kognitionswissenschaften, Universität Bremen.

Mahadevan S., Connell J. (1991) Scaling reinforcement learning to robotics by exploiting the subsumption architecture. In Birnbaum L.A., Collins G.C. (Eds.) *Machine Learning: Proceedings of the 8th International Workshop*, Morgan Kaufman. 328–332.

Manteuffel G. (1992) Neuronal analog models with plastic synapses for neurobiological teaching and robotics. In Schuster H.G. (Ed.): Applications of neural networks. VCH Verlagsgesellschaft. 155–162.

Mataric M. (1994) Interaction and intelligent behavior. Dissertation. Massachusetts Institute of Technology.

Mead C. (1989) Analog VLSI and neural systems. Addison-Wesley.

Nehmzow U., Hallam J., Smithers T. (1989) Really useful robots. In Kanade, Groen, Hertyberger (Eds.): Intelligent autonomous systems, Proceedings of IAS 2, Amsterdam 1989.

Peinemann F. (1994) Neuronale Identifikation und Regelung dynamischer Systeme. *ZKW–Bericht Nr. 2/94.* Krieg–Brückner, Roth, Schwegler (Eds.). Zentrum für Kognitionswissenschaften, Universität Bremen.

Schmajuck N.A., DiCarlo J.J. (1991) A neural network approach to hippocampal function in classical conditioning. *Behavioral Neuroscience*, 1991, Vol. 105, No. 1, 82–110.

Siems U., Herwig Ch., Röfer T. (1994) `SimRobot` – Ein System zur Simulation sensorbestückter Agenten in einer dreidimensionalen Umwelt, *ZKW–Bericht Nr. 1/94.* Krieg–Brückner, Roth, Schwegler (Eds.). Zentrum für Kognitionswissenschaften, Universität Bremen.

Wan H.S., Touretzky D.S., Redish A.D. (1994) Computing goal locations from place codes. in: *Proceedings of the 16th Annual Conference of the Cognitive Science Society* August 13-16, (1994), Atlanta, GA.

Williams D.J., Shah, M. (1990) A Fast Algorithm for Active Contours. *Proc. 3rd Int. Conf. on Computer Vision*, Osaka, Japan. 592–595.

Wittmann T. (1995) Insektennavigation: Modelle und Simulationen Teil 1: Wegintegration. *ZKW–Bericht Nr. 2/95.* Krieg–Brückner, Roth, Schwegler (Eds.). Zentrum für Kognitionswissenschaften, Universität Bremen.