

FORSCHUNGSBERICHTE KÜNSTLICHE INTELLIGENZ

Learning to generate focus trajectories for
attentive vision

J. H. Schmidhuber, R. Huber

Report FKI-128-90

April 1990

T U M

TECHNISCHE UNIVERSITÄT MÜNCHEN

Institut für Informatik, Arcisstr. 21, 8000 München 2, West Germany



Learning to Generate Focus Trajectories for Attentive Vision

Jürgen Schmidhuber*
Rudolf Huber
Institut für Informatik
Technische Universität München
Arcisstr. 21, 8000 München 2, Germany
schmidhu@tumult.informatik.tu-muenchen.de

Abstract

One motivation of this paper is to provide an alternative for inefficient purely static 'neural' approaches to visual target detection. This is done by introducing a more efficient sequential approach. The latter is inspired by the observation that biological systems employ sequential eye-movements for pattern recognition.

The other motivation is to demonstrate that there is at least one principle which can lead to the *learning* of dynamic selective spatial attention.

A system consisting of an adaptive 'model network' interacting with a dynamic adaptive 'control network' is described. The system *learns* to generate focus trajectories such that the final position of a moving focus corresponds to a target to be detected in a visual scene. The difficulty is that no teacher provides the desired activations of 'eye-muscles' at various times. The only goal information is the *desired final input* corresponding to the target. Thus the task involves a complex temporal credit assignment problem, as well as an attention shifting problem.

It is demonstrated experimentally that the system is able to learn correct sequences of focus movements involving translations and rotations. The system also learns to track moving targets.

Some implications for attentive systems in general are discussed. For instance, one can build a 'mental focus' which operates on the set of internal representations of a neural system. It is suggested that *self-referential* systems which model the consequences of their own 'mental focus shifts' open the door for *introspective* learning in neural networks.

The system

There is little doubt that selective attention is essential for large scale dynamic goal-directed learning systems. Here we study an aspect of adaptive attentive vision with neural networks which has not been explored in this general form before: The adaptive control of sequential physical focus-movements. We concentrate on the question: How can an attentive vision system *learn without a teacher* to generate focus trajectories such that the final visual input always looks like a desirable input corresponding to a target? How can it learn to focus on those parts of the visual scene which are relevant for the target detection process?

In our approach there is a neural *control network* C that controls sequential focus movements. The focus has high resolution in its center and low resolution in its periphery. Motoric actions like 'move focus left', 'rotate focus' are based on the activations of C 's output units at a given time. Thus output actions may cause new activations for the input units, and in this case we say that there is *external feedback* (through the environment). The final desired input is an activation pattern corresponding to the target in a static visual scene. The task is to sequentially generate a focus trajectory such that the final input matches the target input. C 's error at the end of a sequential recognition process is given

*Research supported by a scholarship from SIEMENS AG

by the difference between the desired final input and the actual final input. (Control theory calls this a 'terminal control problem'.)

Pure supervised learning techniques for neural networks only work if there is a teacher who provides target *outputs* at every time step of a trajectory. In our case, however, there never are externally given desired outputs. There only is one final desired *input*.

In order to allow credit assignment to past output actions of the control network, we employ a supervised learning *model network* M which learns to represent a model of the visible environmental dynamics. This is done by training M at a given time to predict C 's next input. This prediction is based on previous inputs and outputs of C . For C 's training, C 's output units become identified with the corresponding input units of M , and M 's output units become identified with C 's input units. The model network serves to 'make the world differentiable'. It serves to bridge the gap between output units and input units of the controller.

A learning algorithm for dynamic recurrent networks is employed to propagate gradient information for the weights of the control network back through the model network down into the control network and back through the model network etc... The weights of the model network remain fixed during this procedure. In different contexts and with different degrees of generality this basic principle for credit assignment by *system identification* has been previously described by Munro (1987), Robinson & Fallside (1989), Nguyen & Widrow (1989), Jordan (1988), Werbos (1989), and Schmidhuber (1990b, 1990c, 1990d). It will not be repeated here in detail.

If objects in a visual scene may occupy random positions then it will be impossible for the model network to predict exactly the future focus inputs from previous ones. Unlike with e.g. the 'truck backer upper' (Nguyen and Widrow, 1989) both C and M never 'see' the complete state of the environment, but only some local details. However, this is exactly what attentive vision is good for: The focus of attention should be shifted to parts of the scene that allow to acquire more detailed information about how to go on in the target detection process. The main task of the model network is to help the controller to move the focus into regions of the plane which *allow to continue with more informed moves*. (Although one can not exactly predict what one will see after moving one's eyes to the door, one is setting the stage for additional eye-movements that help to recognize an entering person.)

If the things one attends to never provided *unpredicted* information, then the concept of attention would make no sense. One might say that if the situation was such that the model network could be trained to always make perfect predictions, *then there would be no need for a model network*. In that case a *single* network would be able to store *all* information about the environment. Thus for all interesting cases the model network *necessarily* has to remain imperfect.

So unlike with the 'truck backer upper' (Nguyen and Widrow, 1989) it is not intended to make M a perfect predictor whose output could replace the input from the environment (in that case not much would be gained compared to the static approach to target detection). It suffices if the inner products of the approximated gradients (based on an inaccurate model) for C and the true gradients (according to a hypothetical perfect model) tend to be positive (see also (Jordan, 1988)).

One goal of this work is to demonstrate that imperfect models can contribute to perfect solutions. Another goal is to demonstrate that making a sequential task out of a static one can be very efficient.

Experiments

Target detection without rotations

Consider the figures at the end of this report. A visual scene was made of a dark object on a white background which was placed on a 512 x 512 pixel field. Instead of using hundredthousands of input units (as in a straight-forward inefficient static approach) only 40 input units for C were employed. These were sitting on the focus (a two-dimensional artificial retina) which was controlled by linearly transforming the activations of four output units of the control network. The focus diameter was about equal to the object size. The first figure shows a typical visual scene and the receptive fields of the 40 input units. At a given time step the activation of an input unit was computed by simply averaging the

values of the pixels (black = 1, white = 0) covered by its receptive field. The maximal translation of the focus between two successive time steps was 20 pixels. M had a layer of $40 + 4 = 44$ input units, a layer of 40 hidden units, and a layer of 40 output units. C had a layer of 20 hidden units. Both C and M were *fully* forward-connected. Unlike with the more complicated situations described in (Jordan, 1989) (Robinson and Fallside, 1989) (Schmidhuber, 1990b, 1990c, 1990d) we did not allow internal feedback *within* C or M . In the beginning all weights were initialized randomly between -0.1 and 0.1. A learning rate of 0.1 was employed for both C and M .

After M had been trained (by generating random focus moves), M 's weights were fixed and C 's training phase began. During learning the differences between desired final inputs and *actual* final inputs at the end of each trajectory were used for computing error gradients for the controller. We did *not* use the differences between desired final inputs and *predicted* final inputs. For the experiments reported below it sufficed to propagate errors 'back into time' for not more than 5 time steps.

The object occupied a randomly chosen position in the pixel field. In the beginning of each 'trial' the focus was randomly placed near the object such that the object was partially overlapped by some of the receptive fields of the input units. In the figures the focus center at some time step is given by the center of some arrow. The task was to generate a focus trajectory which leads the center of the focus to a certain predefined detail of the object. For twenty minutes first the model network was trained on a SUN SPARC station, afterwards the controller was trained for another ten minutes.

Our experiments show *that the system described above is able to learn (without a teacher) correct sequences of focus movements although the model network often makes erroneous predictions.* (A precondition for a successful trajectory is a partial overlap between the area covered by the object and the area covered by the 'retina' in the beginning of some trajectory. See the figures at the end of this report.) At the end of a successful trajectory (50 time steps per trajectory were allowed after training) the focus used to have moved towards the target part of the object. The accuracy was nearly perfect: In most cases the difference between the desired position and the actual position was not greater than one or two pixels.

Note that the focus typically did *not* find the *shortest* path to the target. *Instead it developed a preference for edges.* This is reminiscent of certain focus trajectories generated by human subjects.

Each of the 50-steps trajectories depicted in the figures took about one second real time; including graphics output. Using a fully parallel approach for solving similar target detection problems (by considering all pixels at one single time step) would require orders of magnitude more execution time (and probably much more training time, however, this has not yet been tested).

One network for various targets

By providing an additional constant controller input which remains time invariant during the generation of some focus trajectory, various targets can be specified.

The number of C 's input units was doubled: For each original input unit there was another input unit whose constant activation defined the desired activation at the end of a focus trajectory (the goal). (This goal-defining feature is also relevant for 'higher-level' sub-goal generating processes to be addressed later.)

In the experiments, the controller was able to learn to look for parts of a scene which matched the time invariant input. See figure 3 for an illustration of trajectories leading to different targets in the same scene.

Target detection including rotations

Two additional output units for C were introduced for controlling focus *rotations*. Thus the number of M 's input units increased to 46. As it was expected, the learning of focus trajectories which include rotations proved to be more difficult than the learning of pure translation sequences. In order to avoid an experimentally observed 'blow-up' of the error signals during the error propagation phase, error signals were multiplied with a factor of 0.1 whenever they passed a hidden unit of M or C . A learning rate

of 0.19 was employed for both C and M . These were the only modifications of the learning procedure described in the last section.

The maximal focus rotation per time step was 50 degrees. Consider the figures at the end of this report: In the beginning of some trajectory both the focus and the test object composed from a cross and a ball were arbitrarily positioned and rotated in the pixel field. (However, the receptive fields of the input units partially overlapped the object.) The focus rotation at each time step of some trajectory is indicated by the direction of an arrow. The task was to generate a focus trajectory which leads the center of the focus to the center of the cross such that the arrow points to the opposite direction of the center of the ball.

For two hours first the model network was trained on a SUN SPARC station, afterwards the controller was trained for another hour.

The experiments show that the learning of successful focus trajectories involving translations and rotations can be possible, although M uses to make erroneous predictions. See (Huber, 1990) for additional experiments and more detailed accounts of the system behavior.

It should be noted that we currently cannot answer general questions like: How many input units and how many hidden units are necessary for which kind of visual scenes? How to chose optimal learning rates?

Target tracking

Further experiments (not depicted graphically) showed that the system is well-suited for target tracking. The desired detail of the moving object soon is focussed and tracked, as long as the objects velocity does not exceed the maximal focus velocity.

The potential applications are manifold. For instance, it is intended to study the applicability of the system to the following task: Work pieces are transported through an assembly line. Local robots equipped with a non-stationary camera and two interacting networks as above have to detect, visually track, and pick up certain work pieces for further handling.

Parallel learning of C and M : The need for probabilistic output units

With the experiments reported in the last sections there were separate training phases for M and C . The search element that usually is incorporated within reinforcement learning systems by using probabilistic activation rules was buried in the random search of the first phase.

For realistic large scale applications both M and C need to learn in parallel. In general the model network will not be able to explore *all* possible combinations of inputs and actions and their consequences. The control network already should start learning with an incomplete representation of the external dynamics in the model network. M should concentrate just on those parts of the external dynamics that are necessary for achieving C 's goals. Just like Kohonen's self organizing feature maps (Kohonen, 1988) dedicate more storage capacity for fine grained representation of common similar inputs, M should dedicate more storage capacity and time for fine grained modeling of those aspects of the world that are likely to be relevant for the system's main goal, namely, to exist without experiencing pain. (See (Schmidhuber, 1990b) for more reasons for parallel on-line learning of M and C .)

We conducted some experiments with on-line learning. It was found that two interacting conventional *deterministic* networks in the style of (Jordan, 1988) and (Nguyen and Widrow, 1989) were *not* appropriate. Usually a deterministic system soon became trapped in a state where the controller never shifted the focus towards regions which allowed the model network to collect new relevant information about the external world. The randomness *imported from the environment* (Schmidhuber, 1990b) did not suffice for improving performance.

So we introduced some modifications for the controller, in order to provide it with *explicit* search capabilities. Each of the output units was replaced by a little network consisting of two units, one giving the mean and the other one giving the variance for a random number generator which produced random numbers according to a continuous distribution. (We approximated a Gauss distribution by a Bernoulli

distribution.) Weight gradients were computed by applying William's concept of 'back-propagation through random number generators' (Williams, 1988).

It was found that such an on-line learning system can be able to learn appropriate focus trajectories. As it was expected, after training the model network was a good predictor *only* for those situations which the controller typically was confronted with.

Dynamic equilibria 'through the environment'

Since the task is to stop the focus as soon as a certain detail of the environment is focussed, one can draw an interesting analogy to *static* equilibrium networks (like e.g. the Hopfield network, or the Boltzmann machine). To see this, consider the whole combined system consisting of retina, controller, and pixel plane: A given weight vector for C together with a given visual scene defines an 'energy landscape' where the attractors should correspond to solutions for the target detection task. The main difference to conventional equilibrium networks is that the dynamic equilibrium corresponding to a certain attractor involves *external feedback*. A mathematical analysis of such energy landscapes seems to be difficult, since it has to take domain-dependent details of the environment into account.

Ongoing and future research

The approach described above certainly will not remain the last word in the field of adaptive attentive vision. So far our system has been tested only with fairly simple visual scenes involving objects with a rather simple geometrical shape. It is not clear how well the system will do with more complicated scenes. However, there are some promising directions for future research.

Scenes with multiple objects

Scenes with multiple objects or objects with rich internal details require either recurrent connections in both M and C or some other mechanism for escaping certain cases of *local minima*. Local minima can be caused by parts of the pixel plane that look similar to the target input, while the nearby environment does not. In such cases the *relevant* external feedback through the environment becomes highly *non-Markovian*. While there are ongoing experiments with multiple objects and interacting continually running fully recurrent networks (à la (Schmidhuber, 1990d) and (Schmidhuber, 1990c)), there is an approach which in the long run might prove to be even more interesting: The adaptive on-line generation of appropriate sub-goals. Some first work in this direction already has been done (Schmidhuber, 1990e). By using the above-mentioned concept of goal-defining input units with time-invariant activations, it is intended to apply *adaptive sub-goal generators* to the problems of 'local minima' that can arise during the target detection process.

Rewarding temporal perceptual invariances.

To smoothen the error surface of an attentive vision system as described above, in some ongoing research we *impose temporal smoothness constraints on the input units*, too. This can be done by constructing a new error function by *adding differences in successive focus inputs* to the final input error observed at the end of a focus trajectory. (The approach is reminiscent of Jordan's work (Jordan, 1988), however, Jordan imposes temporal constraints on the *output* units.)

The effect is that the system develops a preference for temporal invariances in input space. For attentive vision, such temporal invariances can be caused e.g. by focus movements that follow edges. Thus an unsupervised element (a search for regularities) is introduced into the learning process. (Trivial temporal invariances obtained by stopping the focus are excluded by the goal directed part of the complete error function.)

An empirical motivation for introducing an explicit preference for temporal invariances is given by the experimentally observed fact that even without such a predefined preference the system liked to generate focus trajectories which followed edges.

A comparison of the 'Methods of Temporal Invariances' with the more straight-forward approach described above is under investigation.

Implications for learning selective attention in the general case: An outlook

The only work by other authors that also addresses the problem of *learning* selective attention in reactive environments (and that we are aware of) is the work of Whitehead and Ballard (Whitehead and Ballard, 1990). Their system disposes of adaptive actions that can bind 'markers' to certain features of an environmental state. The context is given by the 'blocks world' example. Markers dynamically mask or emphasize aspects of the visible environment. Attention learning is inspired by Sutton's TD-Methods (Sutton, 1988) which provide some theoretical foundation for adaptive critics as employed in e.g. (Barto et al., 1983) (Anderson, 1986) (Schmidhuber, 1990a).

The system described above (which learns by using the principle of system identification) as well as Whitehead and Ballard's system can be viewed as implementing selective attention by some sort of external feedback. The system described in (Schmidhuber, 1990b), which implements 'curiosity' and 'boredom' by means of adaptive dynamic attention depending on the amount of a model network's ignorance about the external dynamics, also is based on external feedback.

Cognitive scientists often like to refer to a different form of selective attention: Some cognitive systems are supposed to focus on certain parts of their internal representations by means of some 'internal searchlight'. Can we extend the basic concept of *learning* attention from the last section to the case of internal attention?

The answer should be 'yes'. The basic step we have to make is *to introduce new internal adaptive actions which explicitly manipulate the systems internal state as well as the state of internal 'searchlights'*. It is intended to investigate learning systems consisting of a number of networks, where one learns to model the 'usefulness' of certain *internal* actions produced by another network. (Internal actions include actions for stepping through sub-goal hierarchies (Schmidhuber, 1990e) as well as actions for 'mental focus shifts'.) Using the 'model network', a control network should learn to produce 'good' internal actions at appropriate times.

It is suggested that 'internal actions' open the door for *self-referential* and *introspective* learning. It is also suggested that internal actions will be essential for bridging the gap between so-called 'sub-symbolic' and so-called 'symbolic' computation.

References

- Anderson, C. W. (1986). *Learning and Problem Solving with Multilayer Connectionist Systems*. PhD thesis, University of Massachusetts, Dept. of Comp. and Inf. Sci.
- Barto, A. G., Sutton, R. S., and Anderson, C. W. (1983). Neuronlike adaptive elements that can solve difficult learning control problems. *IEEE Transactions on Systems, Man, and Cybernetics*, SMC-13.
- Huber, R. (1990). *Selektive visuelle Aufmerksamkeit: Untersuchungen zum Erlernen von Fokustrajektorien durch neuronale Netze*. Diplomarbeit, Institut für Informatik, Technische Universität München.
- Jordan, M. I. (1988). Supervised learning and systems with excess degrees of freedom. Technical Report COINS TR 88-27, Massachusetts Institute of Technology.
- Jordan, M. I. (1989). Generic constraints on underspecified target trajectories. In *IJCNN International Joint Conference on Neural Networks, Vol 1*.

- Kohonen, T. (1988). *Self-Organization and Associative Memory*. Springer, second edition.
- Munro, P. W. (1987). A dual back-propagation scheme for scalar reinforcement learning. *Proceedings of the Ninth Annual Conference of the Cognitive Science Society, Seattle, WA*, pages 165–176.
- Nguyen and Widrow, B. (1989). The truck backer-upper: An example of self learning in neural networks. In *IEEE/INNS International Joint Conference on Neural Networks, Washington, D.C.*, volume 1, pages 357–364.
- Robinson, T. and Fallside, F. (1989). Dynamic reinforcement driven error propagation networks with application to game playing. In *Proceedings of the 11th Conference of the Cognitive Science Society, Ann Arbor*, pages 836–843.
- Schmidhuber, J. H. (1990a). Applying temporal difference methods to fully recurrent reinforcement learning networks. *In preparation*.
- Schmidhuber, J. H. (1990b). Making the world differentiable: On using fully recurrent supervised learning neural networks for dynamic reinforcement learning and planning in non-stationary environments. Technical Report FKI-126-90, Institut für Informatik, Technische Universität München.
- Schmidhuber, J. H. (1990c). An on-line algorithm for dynamic reinforcement learning and planning in reactive environments. In *Proc. IEEE/INNS International Joint Conference on Neural Networks, San Diego*.
- Schmidhuber, J. H. (1990d). Reinforcement learning with interacting continually running fully recurrent networks. In *Proc. INNC International Neural Network Conference, Paris*.
- Schmidhuber, J. H. (1990e). Towards compositional learning with dynamic neural networks. Technical Report FKI-129-90, Institut für Informatik, Technische Universität München.
- Sutton, R. S. (1988). Learning to predict by the methods of temporal differences. *Machine Learning*, 3:9–44.
- Werbos, P. J. (1989). Backpropagation and neurocontrol: A review and prospectus. In *IEEE/INNS International Joint Conference on Neural Networks, Washington, D.C.*, volume 1, pages 209–216.
- Whitehead, S. and Ballard, D. H. (1990). Active perception and reinforcement learning. Technical Report 331, University of Rochester, Dept. of Comp. Sci.
- Williams, R. J. (1988). On the use of backpropagation in associative reinforcement learning. In *IEEE International Conference on Neural Networks, San Diego*, volume 2, pages 263–270.

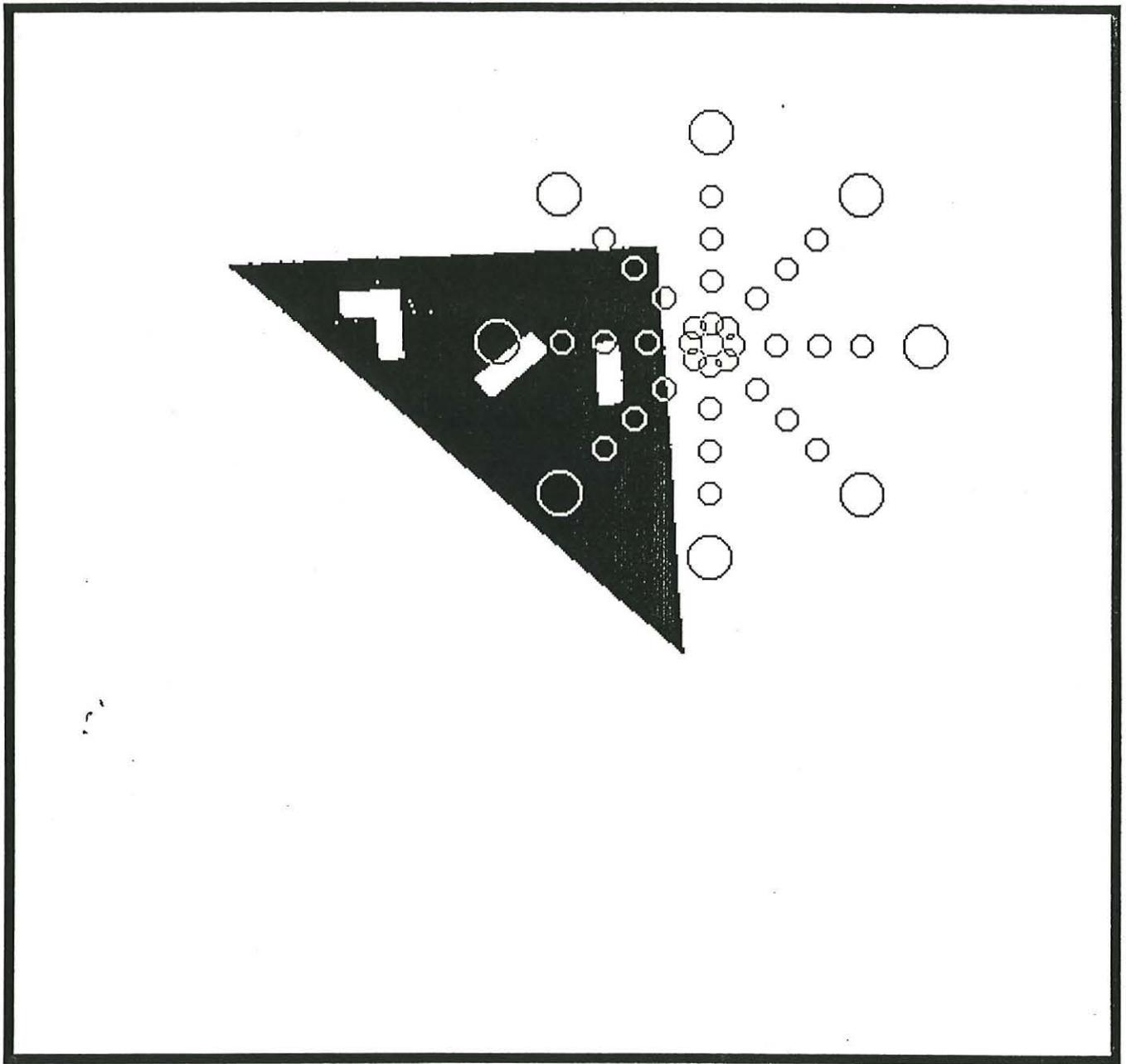


Figure 1: A typical visual scene. The diameters of the receptive fields of the retina's input units are indicated by circles.

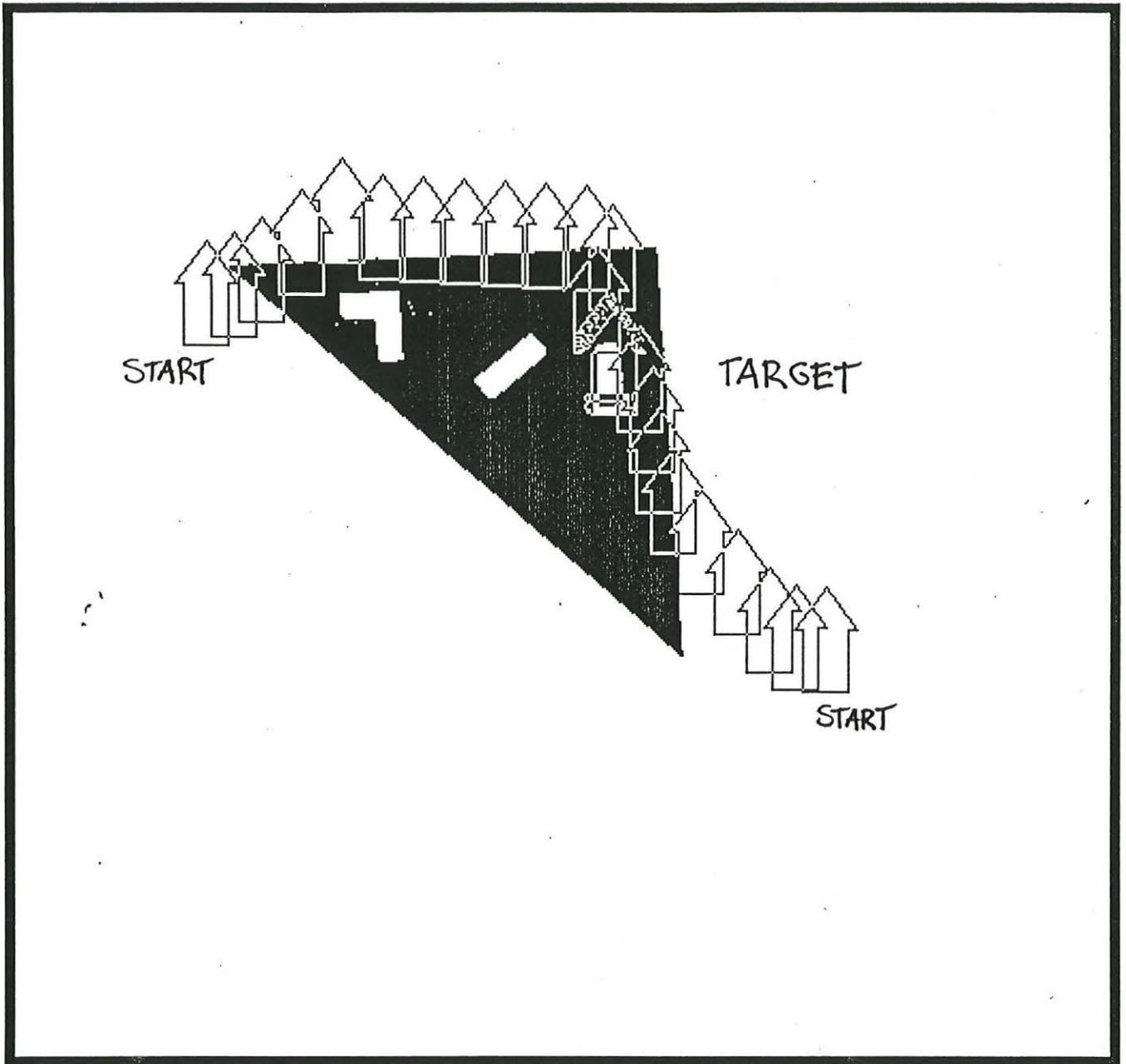


Figure 2: Translations: Examples of focus trajectories leading from various start positions to the target, which is the small white rectangle on the right side of the big black triangle. No teacher told the focus how to do that! Note that the system typically did not find the shortest path to the target. It developed a preference for edges.

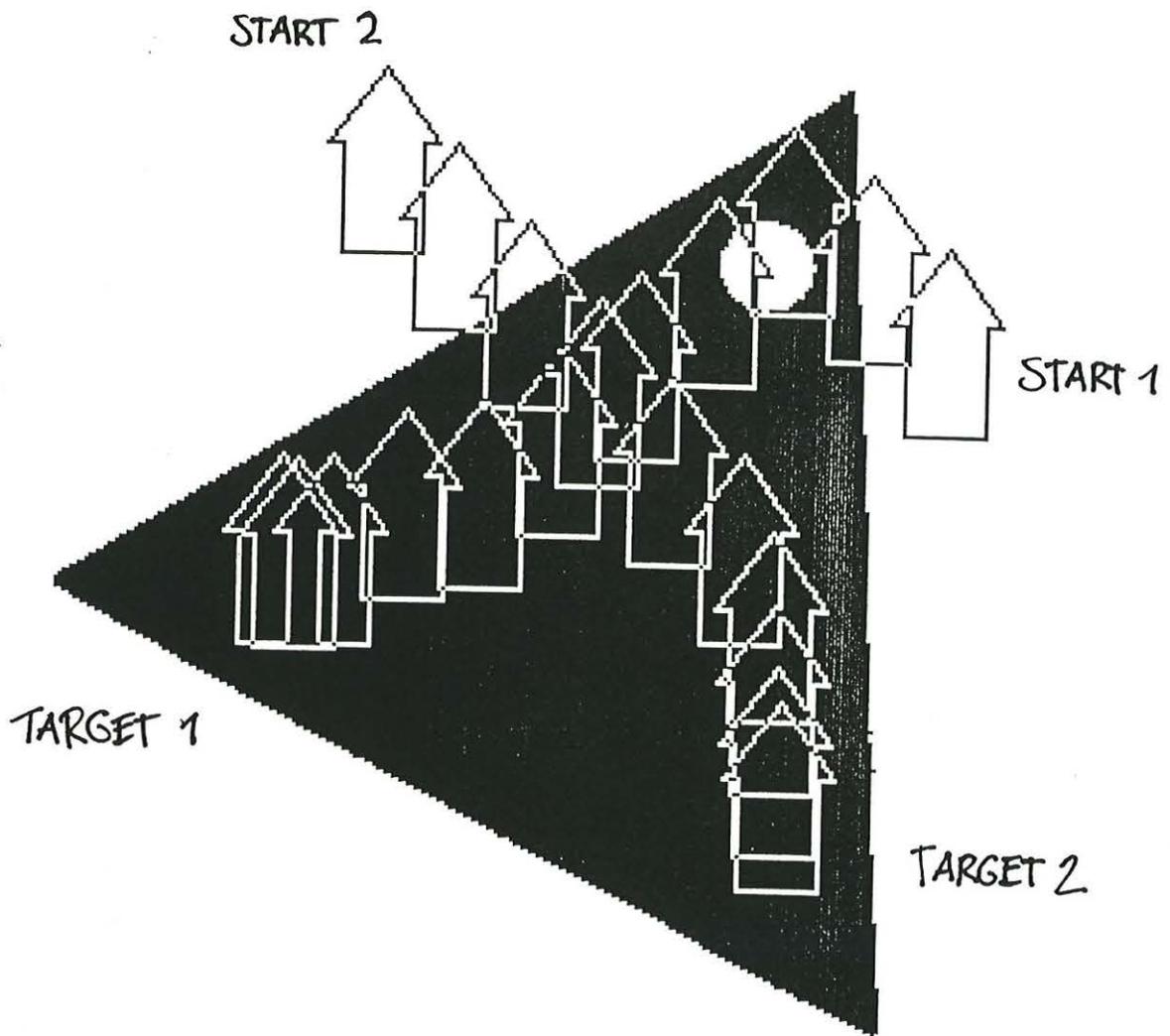


Figure 3: *One* controller for various targets specified by an additional constant input: Examples of focus trajectories leading from various start positions to different targets. The first target is near the left corner of the triangle. The second target is near the lower corner.

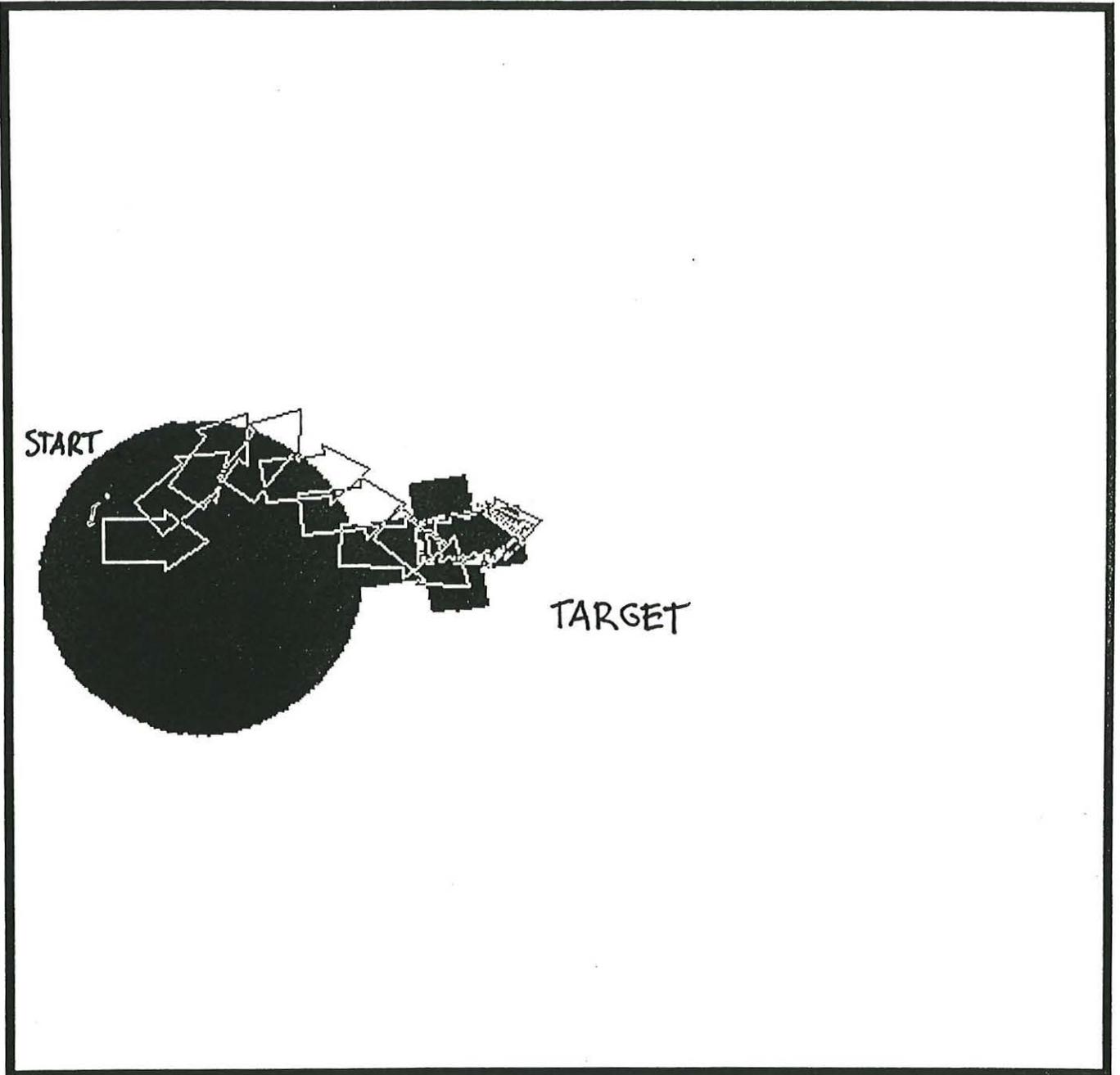


Figure 4: Translations and rotations: The focus finding its way from the inside of the object to the target. No teacher told it how to do that!

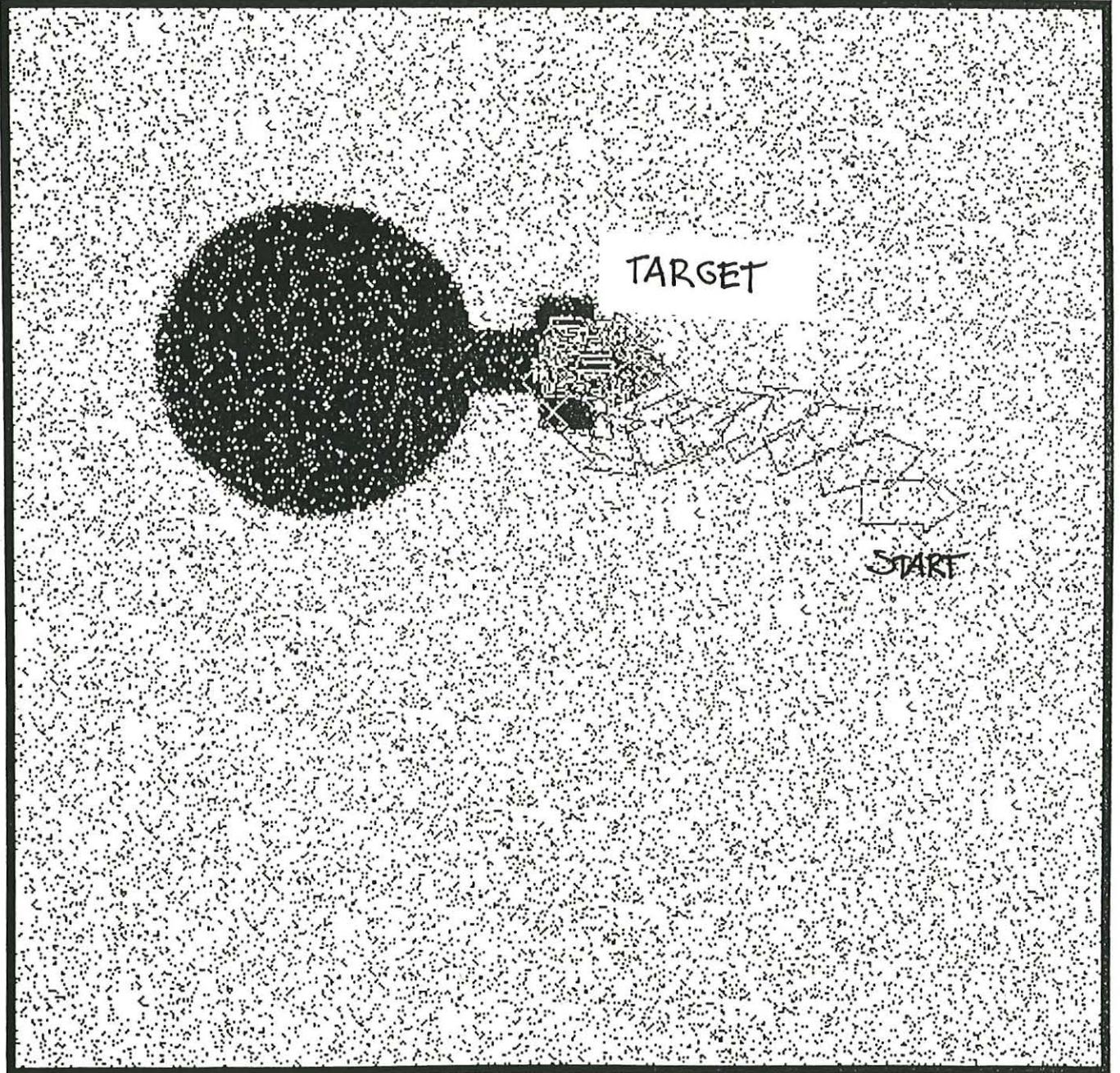


Figure 5: The focus pushing backwards to the target on a noisy pixel plane.

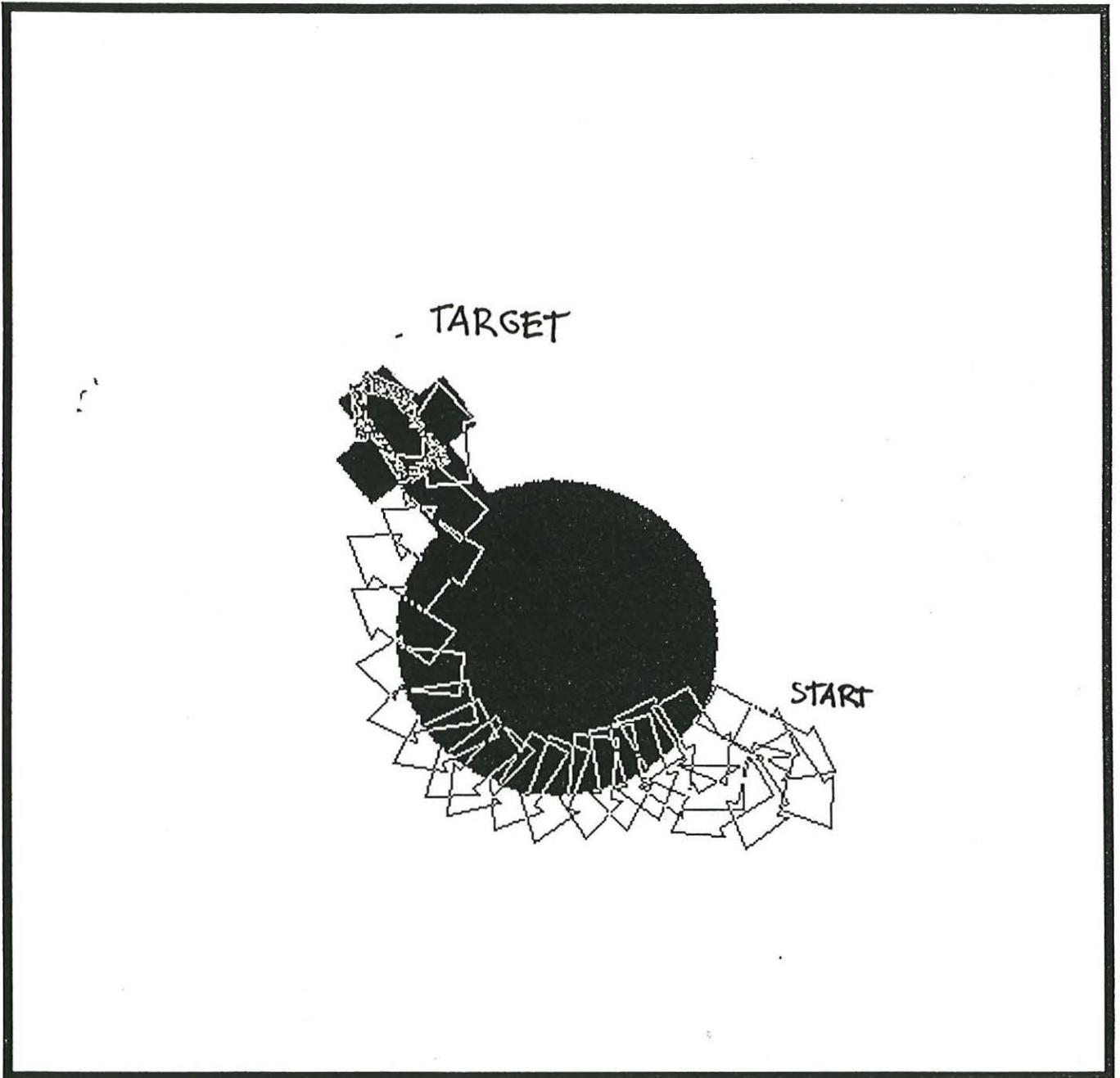


Figure 6: A good strategy discovered by the system was to first find the edge of the ball and then follow the edge until the cross appeared. From there it moved more or less straight to the target. Note again that there was no teacher who told the system that this might be a good strategy!

FORSCHUNGSBERICHTE KÜNSTLICHE INTELLIGENZ

Institut für Informatik Technische Universität München

Phone: +49 - 89 - 2805217
Telex: tumue d 05-22854
Fax: +49 - 89 - 2105 - 8207
e-mail: fki@tumult.uucp
fki@informatik.tu-muenchen.de

The following list comprises reports which have been written in the AI/Cognition group at the chair of Prof. Brauer as well as in the AI group "INTELLEKTIK" at the chair of Prof. Jessen.
The papers can be obtained free of charge by writing to:

"FKI"
Institut für Informatik
TECHNISCHE UNIVERSITÄT MÜNCHEN
Arcisstr. 21
D-8000 München 2

FKI-Reports 1988-1991

- FKI-84-88 Christian Freksa: *Cognitive Science - eine Standortbestimmung*. In: Heyer, G., J. Krems, G. Görz (Hrsg.): *Wissensarten und ihre Darstellung*. Informatik Fachberichte. Heidelberg: Springer, 1988.
- FKI-85-88 Christian Freksa: *Intrinsische vs. extrinsische Repräsentation zum Aufgabenlösen oder die Verwandlung von Wasser in Wein*. In: Heyer, G., J. Krems, G. Görz (Hrsg.): *Wissensarten und ihre Darstellung*. Informatik Fachberichte. Heidelberg: Springer, 1988.
- FKI-86-88 Uli Furbach/ Steffen Hölldobler: *FHCL - Functions in Horn Claus Logic*
- FKI-87-88 Uli Furbach/ Bertram Fronhöfer: *Transformation Systems for Program Synthesis: Knuth Bendix Completion and Fold/Unfold*
- FKI-88-88 Daniel Hernández: *Module Fault Localization in a Software Toolbus based System*.
- FKI-89-88 Thierry Conrad/ Uli Furbach: *Sorts are Nothing but Functions - An Equational Approach to Sorts for Logic Programming*.
- FKI-90-88 Christoph Kreitz/ Gerd Neugebauer/ Bertram Fronhöfer: *Logic Oriented Program Synthesis - Goals and Realization*.
- FKI-91-88 Uli Furbach/ Christian Freksa/ Gerhard Dirlich: *Wissensrepräsentation in Künstlichen Symbolverarbeitenden Systemen*. In: Mandl, H., H. Spada (Hrsg.): *Wissenspsychologie*. München/Weinheim: Psychologie Verlags Union 1988. Auch in: Mandl, H. u. a. (Hrsg.): *Wissenspsychologie*. Ein Lehrbuch. München: Urban & Schwarzenberg, 1988.
- FKI-92-88 Erwin Klöck: *Utterance Generation Without Choice*. In: Hoepfner, W. (Hrsg.): *GWAI-88, 12th German Workshop on Artificial Intelligence*. Berlin: Springer 1988.
- FKI-93-88 Vytautas Cyras: *Loop Program Synthesis Using Array Traversing Modules*.
- FKI-94-88 Kai Zimmermann: *Der Netzeditor. Eine komfortable Umgebung zum Erstellen und Testen von konnektionistischen Netzen*.
- FKI-95-88 Andreas Stolcke: *Generierung natürlichsprachlicher Sätze in unifiktionsbasierten Grammatiken*.
- FKI-96-88 Andreas Strasser: *Zur Portierbarkeit taxonomischer Wissensbasen zwischen heterogenen Systemen*.
- FKI-96-a-88 Stefan Bayerl, Caferra R., F. van der Linden: *Theses on Automated Theorem Proving*.
- FKI-96-b-88 Reinhold Letz, Johannes Schumann: *Global Variables in Logic Programming*.
- FKI-96-c-88 Reinhold Letz: *Expressing First Order Logic within Horn Clause Logic*.
- FKI-97-89 Stefan Bayerl, Reinhold Letz, Johannes Schumann: *SETHEO: a SEquential THEOremprover for first order logic - Version 2 -*.
- FKI-98-89 Wilfried Brauer, Christian Freksa and the AI/Cognition Group: *Connectionist Approach to the Description of Spatial Knowledge and related papers*.
- FKI-99-89 Franz Kurfeß: *Logic and Reasoning with Neural Models*.
- FKI-100-89 Johannes Schumann, Wolfgang Ertel, Christian Suttner: *Learning Heuristics for a Theorem Prover using Back Propagation*. In: Retti, J. u.a. (Hrsg.): *5. ÖGAI-Conference 1989*. Berlin, Springer 1989.
- FKI-101-89 Jochen Hager, Martin Moser: *An Approach to Parallel Unification Using Transputers*. In: Metzging, D. (Hrsg.): *GWAI-89, 13th German Workshop on Artificial Intelligence*. Berlin: Springer 1989.
- FKI-102-89 Eike Jessen: *Leistungsfähigkeit von Parallelrechnern*. (Verbesserte Fassung: Jan.91).
- FKI-103-89 Bertram Fronhöfer: *Plan Schemes in Planlog*. In: Jorrand P., Sgurev V. (eds.): *AIMSA 88*. North-Holland 1988.

- FKI-104-89 Bertram Fronhöfer, Uli Furbach: *ESPRIT PROJECT 973 ALPES - A Programming Environment for Logic Programming*. In: Brauer W. Freksa C. (Hrsg.): *Wissensbasierte Systeme, 3. Internationaler GI-Kongreß*. Berlin: Springer 1989 (Informatik Fachberichte 227)
- FKI-105-89 Andreas Strasser: *Strukturierte Darstellung juristischen Wissens*. In: Paul M. (Hrsg.): *Proc. 19. GI-Jahrestagung*, Berlin: Springer 1989.
- FKI-106-89 Wolfgang Bibel, L. Farinas del Cerro, Bertram Fronhöfer, Andreas Herzig: *Plan Generation by Linear Proofs: on Semantics*. In: Metzging D. (Hrsg.): *GWAI-89, 13th German Workshop on Artificial Intelligence*. Berlin: Springer 1989.
- FKI-107-89 Stefan Lanser: *NEOSIM Der Neocognitron Simulator*.
- FKI-108-89 Bertram Fronhöfer: *Default Connections in a Modal Planning Framework*.
- FKI-109-89 Gerhard Strobl, Dieter Dodenhöft: *A Knowledge-Based Approach for Design Analysis in Mechanical Engineering*. In: Plander I. (ed): *Proceedings of the 5th International Conference on Artificial Intelligence and Information-Control Systems of Robots - 89*. North-Holland, 1989
- FKI-110-89 Uli Furbach, Steffen Hölldobler: *Equations, Order-Sortedness and Inheritance in Logic Programming*. In: Wilkerson (ed.): *Advances in Logic Programming and Automated Reasoning*. Ablex Publishing Corporation, 1990
- FKI-111-89 Uli Furbach, Steffen Hölldobler, Joachim Schreiber: *Linear Paramodulation modulo Equality*. In: Metzging D. (Hrsg): *Proceedings of the 13th German Workshop on Artificial Intelligence*. Berlin: Springer 1989 (Informatik Fachberichte 216)
- FKI-112-89 Andreas Strasser, Gerhard Strobl, Dieter Dodenhöft: *Flexible Classification in ProObj*.
- FKI-113-89 Franz Kurfeß, Xaver Pandolfi, Zoubir Belmesk, Wolfgang Ertel, Reinhold Letz, Johannes Schumann: *PARTHEO and FP2: Design of a Parallel Inference Machine*. In: Treleaven P. (ed): *Parallel Computers: Object-Oriented, Functional and Logic*, Chicester: Wiley, 1989
- FKI-114-89 Wolfgang Ertel: *Backpropagation with Temperature Parameter and Random Pattern Presentation*.
- FKI-115-89 Christian Suttner, Wolfgang Ertel: *Automatic Acquisition of Search Guiding Heuristics*. In: Stickel, M (ed.) *Proc. of the 10th International Conference on Automated Deduction (CADE)*, Kaiserslautern, 1990, LNAI 449, Springer-Verlag, pp. 470-484.
- FKI-116-89 Reinhold Letz, Johannes Schumann, Stefan Bayerl, Wolfgang Bibel: *SETHEO: A High-Performance Theorem Prover*. In: *Journal of Automated Reasoning* (1991)
- FKI-117-89 Stefan Bayerl, Reinhold Letz, Johannes Schumann, Franz Kurfeß, Wolfgang Ertel: - *ESPRIT - 415 Deliverable D16, PARTHEO/6: Full First order Logic Parallel Inference Machine - Language and Design* -.
- FKI-118-89 Wolfgang Ertel, Johannes Schumann, Reinhold Letz, Stefan Bayerl, Franz Kurfeß, M. van der Koelen, Christian Suttner, N. Trapp: - *ESPRIT - 415 Deliverable D15, PARTHEO/6: Parallel Automated Theorem Prover based on the Connection Method for Full First order Logic - Implementation and Performance* -.
- FKI-119-89 Stefan Bayerl, Reinhold Letz, Johannes Schumann: *PARTHEO: A Parallel Inference Machine*. In: Brauer W. Freksa C. (Hrsg.): *Wissensbasierte Systeme, 3. Internationaler GI-Kongreß*. Berlin: Springer 1989 (Informatik Fachberichte 227)
- FKI-120-89 Reinhold Letz, Johannes Schumann: *PARTHEO: A High Performance Parallel Theorem Prover*. In: Stickel, M (ed.) *Proc. of the 10th International Conference on Automated Deduction (CADE)*, Kaiserslautern, 1990, LNAI 449, Springer-Verlag, pp. 40-56.
- FKI-121-89 Johannes Schumann, N. Trapp, M. van der Koelen: *SETHEO/PARTHEO, Users Manual*.
- FKI-122-89 Reinhold Letz, Stefan Bayerl, Johannes Schumann, Bertram Fronhöfer: *The Logic Programming Language LOP*.
- FKI-123-89 Reinhold Letz: *Classical Negation in Logic Programming*.
- FKI-124-90 Jürgen H. Schmidhuber: *A local learning algorithm for dynamic feedforward and recurrent networks*.
- FKI-125-90(revised) Jürgen H. Schmidhuber: *Networks adjusting networks*. A revised and extended version of 'Networks adjusting Networks' in J. Kindermann and A. Linden (eds.), *Proceedings of 'Distributed Adaptive Neural Information Processing'*, St. Augustin, 24. - 25. 5. 1990, pp. 197-208, Oldenbourg, 1990
- FKI-126-90(revised) Jürgen H. Schmidhuber: *Making the world differentiable: On Using Self-Supervised Fully Recurrent Neural Networks for Dynamic Reinforcement Learning and Planning in Non-Stationary Environments*
- FKI-127-90 Gerhard Weiss: *Artificial Neural Learning*.
- FKI-128-90 Jürgen H. Schmidhuber, Rudolf Huber: *Learning to generate focus trajectories for attentive vision*.
- FKI-129-90 Jürgen H. Schmidhuber: *Towards compositional learning with dynamic neural networks*.
- FKI-130-90 Andreas Strasser: *AVIGNON-90, Specialized Conference on Second Generation Expert Systems, ECZ*.
- FKI-131-90 Peter Baumgartner: *Combining horn clause logic with rewrite rules*.
- FKI-132-90 Gerhard Weiss: *Combining neural and evolutionary learning: Aspects and approaches*.
- FKI-133-90 Peter Baumgartner, Siegfried Meggendorfer, Zong Yan Qiu: *Software Specification Methods from the Viewpoint of Reusability*
- FKI-134-90 Kai Zimmermann: *Entwicklung einer bildorientierten Benutzungsoberfläche für wissensbasierte Systeme*, In: K. Kansy, P. Wißkirchen (Hrsg), *Graphik und KI, Informatik-Fachberichte 239*, Springer-Verlag, Berlin, 1990
- FKI-135-90 Daniel Hernández: *Relative Representation of Spatial Knowledge: The 2-D Case*
- FKI-136-90 Christian Freksa: *Qualitative Spatial Reasoning*
- FKI-137-90 Marco Dorigo, Bernhard Schätz: *Mapping a Generator for Neural Network Simulators to a Transputer System*
- FKI-138-90 Petra Bräunling, Christian Freksa, Kai Zimmermann (eds.): *The SpaceGarden Bibliography*

- FKI-139-90 Andreas Strasser: *Probleme regelbasierter Repräsentation von Wissen in juristischen Expertensystemen*. In: *Juristische Expertensysteme: Methoden-Insturmente-Anwendungen*. Heimanns-Verlag, Köln (1991).
- FKI-140-90 Patrick Thomas: *Beyond Hebb Synapses: Biological Building Blocks for Unsupervised Learning in Artificial Neural Networks*
- FKI-141-90 Bertram Fronhöfer: *Implementing Exeption in Inheritance by Concept Transforming Actions*
- FKI-142-90 Bertram Fronhöfer, Max Moser: *Program Optimization by Analyzing Similar Repetitive Subcomputations*
- FKI-143-90 Wolfgang Ertel: *Random Competition: A Simple, but Efficient Method for Parallelizing Inference Systems*
- FKI-144-90 Reinhold Letz, Klaus Mayr: *Proof Schemata in Propositional Calculi*
- FKI-145-90 Peter Baumgartner: *Modelling Software Reuse with Predicate Logig*
- FKI-146-91 Christian Freksa: *Conceptual Neighborhood and its role in temporal and spatial reasoningc*, To appear in: Proceedings of the IMACS Workshop on Desicion Support Systems and Qualitative Reasoning, M. Singh, L. Travé-Massuyès (eds.), 1991, Elsevier Science Publishers
- FKI-147-91 Jürgen Schmidhuber: *Learning to Control Fast-Weight Memories: An Alternative to Dynamic Recurrent Networks*
- FKI-148-91 Jürgen Schmidhuber: *A Neural Sequence Chunker*
- FKI-149-91 Jürgen Schmidhuber: *Adaptive Confidence and Adaptive Curiosity*
- FKI-150-91 Andreas Strasser: *Towards a Lattice of Knowledge Representation Systems*
- FKI-151-91 Jürgen Schmidhuber: *An $O(n^3)$ Learning Algorithm for Fully Recurrent Networks*
- FKI-152-91 Thomas Laußermair, Gerhard Weiß : *Artificial Life - Eine Einführung*
- FKI-153-91 Christian Freksa: *Temporal Reasoning Based on Semi-Intervals*

