to be published in **Proc. of 39th Intern. Scientific Colloquium Ilmenau**, Sept. 27 to 30, 1994

A Neural Network Architecture for Episodic Feature Representation *

H.-J.Boehme, U.-D.Braumann, H.-M.Gross
Department of Neuroinformatics
Technical University of Ilmenau
98684 Ilmenau, Germany

e-mail: hans@informatik.tu-ilmenau.de

Introduction

Biological information processing systems reach their solution in an interpretation task by means of continuous interaction between the internal state of the system and the incoming data which can be described as a process of *Sensory Controlled Internal Simulation* (SCIS) (see [2]). The proposed architecture builts up an episodic feature representation (see also [4] and [5]) and uses this representation for flexible generation and continuous verification of hypotheses about the input. In comparison with models ending up in a stationary state of coding (for example resulting from parallel pattern matching and/or pure bottom-up processing) a SCIS process provides evident advantages:

- It allows to modify continuously the systems knowledge dependent on its suitability for an appropriate interaction with its environment.
- The continuous interaction between the internal hypotheses and the incoming data seems to prevent the system from the problem of combinatorial explosion because the system always tries to handle a present input on the base of the knowledge already accumulated.

The Neural Network Architecture

In order to implement and investigate the model sufficiently, we defined several constraints concerning its working environment: The primary input data consists of a basic alphabet of abstract features forming a set of abstract objects. The objects are characterized by a sequence of extracted features from the alphabet whereby the sequence could be provided for instance by an internal or external scan path (see also [9] and [8]). Furthermore we assume that for the present our model is restricted only to the feature domain and position information is reduced to the place of a certain feature inside the scan path. A further assumption is that the features are represented by distributed activity patterns. For better plausibility we used letter patterns as basic features and combined them to words as candidates for significant objects.

*supported by the Federal Department of Research and Technology (BMFT), Grand No. 413-5839-01 IN 101D - NAMOS-Project

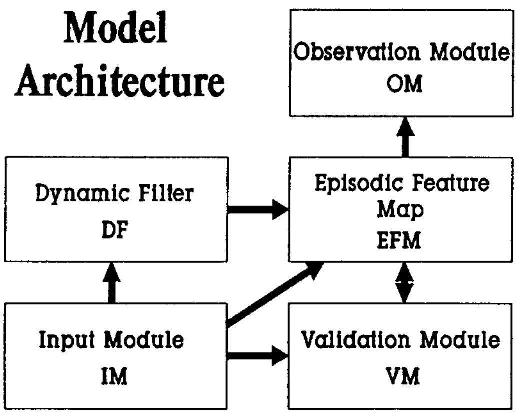

Figure 1 *depicts in a strictly simplified manner the proposed neural network architecture. The Input Module (IM) provides the sequence of features (letter patterns) in form of a closed and reverberating pattern sequence. The Dynamic Filter (DF) transforms the input sequence into a limit cycle that codes highly specific the spatial and temporal (order of the patterns) characteristic of the processed sequence (see also [1]). The Episodic Feature Map (EFM) builts up a very specific memory trace by exploitating the information contained in the input pattern sequence and in the limit cycle of the DF. By means of the Validation Module (VM) a continuous interaction between the internal state of the system (especially inside the EFM) and the input is ensured. The Observation Module (OM) has no functional importance but is necessary to record and visualize the ongoing processes of generation and verification of hypotheses inside the EFM.*

Organization of the Episodic Feature Map

Inside the *Input Module* (IM) the actual input pattern sequence is provided which serves as input signal to the *Dynamic Filter* as well as to the *Episodic Feature Map.* According to the contained patterns (abstract features) the corresponding clusters of the *Episodic Feature Map* (EFM) are activated in a sequential manner.

The *Dynamic Filter* (DF) (see also [1]) consists of an array of formal neurons with the same dimension like the feature patterns. To avoid boarder effects and to ensure that each node is influenced by the same number of nodes from the local neighbourhood the net is arranged as a closed surface. The DF behaves like a Cellular Automata (see [6] and [7]) and is continuously driven by the input pattern sequence. At every time step of processing each filter node calculates its activity as the weighted sum of the activity of the surrounding filter nodes and the external input. After a certain number of time steps the DF reaches a steady state in which a stable limit pattern cycle emerges. The patterns of this cycle carry the information about the spatial and temporal structure of the input pattern sequence as follows: One pattern of the limit cycle corresponds with one pattern (feature) contained in the input sequence. So the transformed patterns of the limit cycle implicitly code both what patterns belong to the sequence and the order of their occurence. Consequently, the information provided by the limit cycle of the *Dynamic Filter* can be used as *context* for the representation of all features forming the actual input. By means of exploiting the temporal correlation between the limit cycle of the DF and the input pattern sequence it is possible to select the best fitting node of each corresponding feature cluster of the EFM (see figures 2 and 3). Via a simple link mechanism (Hebbian learning) a memory trace between the selected nodes is made by an unidirectional increasing of the strength of the concerning synaptic connections. Therefore this memory trace codes highly specific the handled pattern sequence. Following, the same procedure takes place with every presented sequence until each sequence is included into the EFM. Furthermore each memory trace inside the EFM is mapped to one corresponding node of the *Observation Module* (OM) to find out the different traces during the *Sensory Controlled*

Internal Simulation.

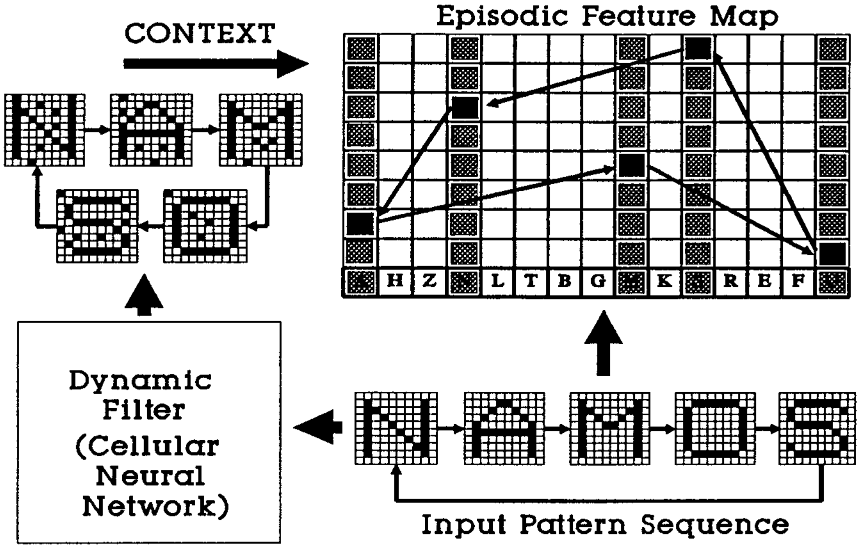

Figure 2 *is to demonstrate the emergence of a highly specific memory trace coding the spatial and temporal characteristic of the handled input pattern sequence. The exploitation of the spatial and temporal characteristics of the inputs allows to establish suitable hypotheses about the actual input.*

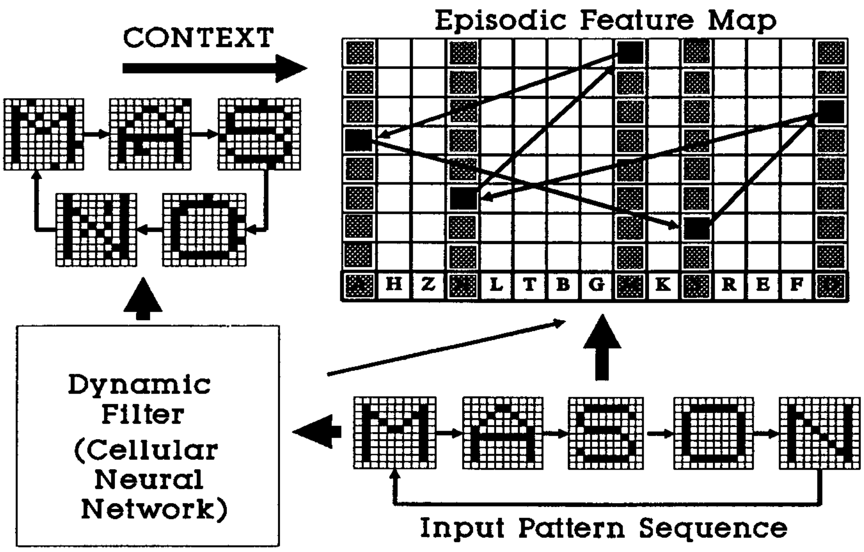

Figure 3 *demonstrates what is happened if another input pattern sequence containing the same feature patterns like the sequence of figure 2 but only differing in the order of the patterns. Consequently, a different memory trace emerges across the Episodic Feature Map.*

Model Behaviour

We used a two phased method in which first a set of pattern sequences was presented to the model and the EFM (26 feature clusters each of them contained 10 nodes) was organized without hypotheses generation and verification. After that the recall phase was started to investigate the performance of the model in handling the represented pattern sequences by *Sensory Controlled Internal Simulation.*

During the SCIS process we presented patterns of a certain sequence already stored. Thereby the EFM was only to make use of its internal structure to built up the right (most likely) memory trace. The activity of one EFM-node is determined by the external input (feature pattern), the incoming projections from other nodes inside the EFM and of the matching via the *Validation Module* (VM). All EFM nodes are synchronuously updated during one step of the SCIS process. The EFM nodes project continuously their expectation about a certain feature pattern to the VM where a matching between this expectations and the actual feature pattern takes place. Depending on this matching the projecting EFM nodes are supported or attenuated. The desired result of this process is that this memory trace of the EFM best fitting with the presented pattern sequence will become the strongest one or is at least among the strongest memory traces.

Simulation Results

The illustrated 3 simulation examples were recorded with 3 different input sets containing 248, 704 and 1124 pattern sequences. In order to demonstrate the powerful representation capacity of our model an alternative feature map was organized for all depicted simulation examples. In this alternative feature map was used no context information and the selection of a node of a certain feature cluster was randomly made during the organization of the map. Hence, a homogeneous usage of all nodes of this alternative feature map was achieved. In the histograms the non-filled bars show the results achieved with the contextual organized EFM whereas the filled bars depict the results made by the randomly organized feature map.
The histograms outline the ranking of the most active memory traces during the recall of all represented traces (observed via the activity distribution across the OM nodes). The left hand side of the figures 4, 5, and 6 shows how much sequences reach the first, second and so on position of ranking during the SCIS process. To get a more realistic view at the right hand side of the figures 4, 5, and 6 the ranking of a certain sequence was normalized to the total amount of all stored sequences. Here the histograms show the ranking of the right hypothesis among the first one to ten percent of all stored sequences.
The upper rows display the recall behaviour while presenting the input sequences in the same spatial and temporal relation as during learning. The middle and the bottom rows indicate the results yielded with input sequences in which 30% of the contained patterns were interchanged (the same patterns as in the original sequence but differing in the order of the patterns) or were exchanged (some patterns were replaced by patterns not contained in the original sequence), respectively.
Based on the depicted results two evident conclusions can be made:

- First, the representation of objects by means of memory traces inside an *Episodic Feature Map* seems to be an appropriate scheme for flexible and robust interaction with a complex and varying environment in general. This conclusion is derived from the fact that the results achieved with the noncontextual feature maps are qualitatively comparable with the results reached by the contextual feature maps, at least for the first two examples (figures 4 and 5).

- Second, the proposed contextual organized feature map maintaining the spatial and temporal structure of the input inside the memory traces makes such a representation much more specific and therefore the emergence of a certain trace can be better supported significantly. This effect becomes the more obvious the more the EFM is loaded by different traces which can be seen very clearly in the simulation example of figure 6. Here the noncontextual organized map is not capable of establishing right hypotheses whereas the contextual organized map can support the emergence of the correct hypotheses (most active memory traces) in almost all cases.

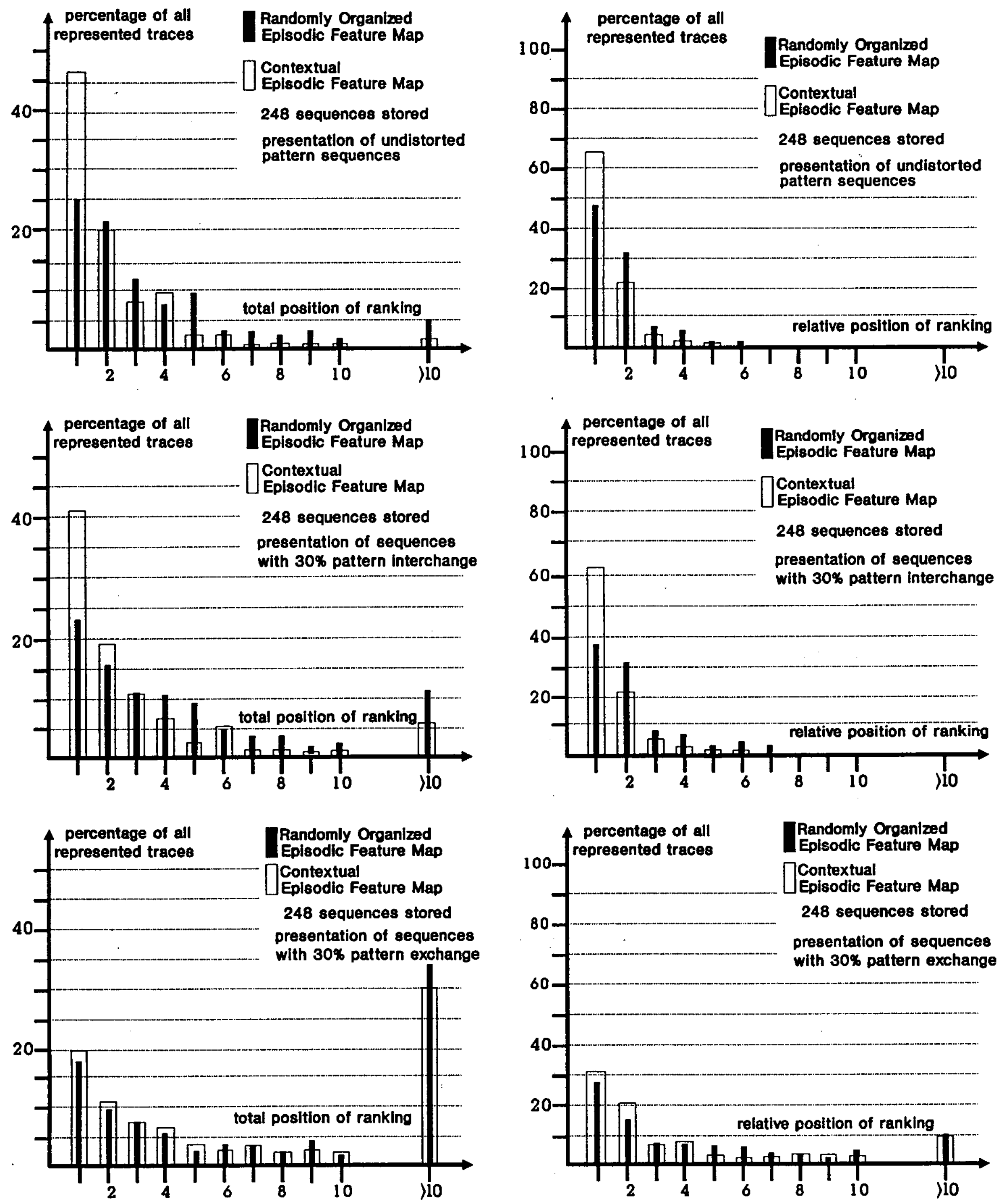

Figure 4 *gives the simulation results for an Episodic Feature Map which is trained with 248 input pattern sequences.* **Upper row:** *Recall behaviour during presentation of undistorted pattern sequences;* **Middle row:** *30% of the patterns of all sequences were interchanged (altering the order of the patterns);* **Bottom row:** *30% of the patterns of all presented sequences were substituted by patterns not contained in the concerning original sequences*

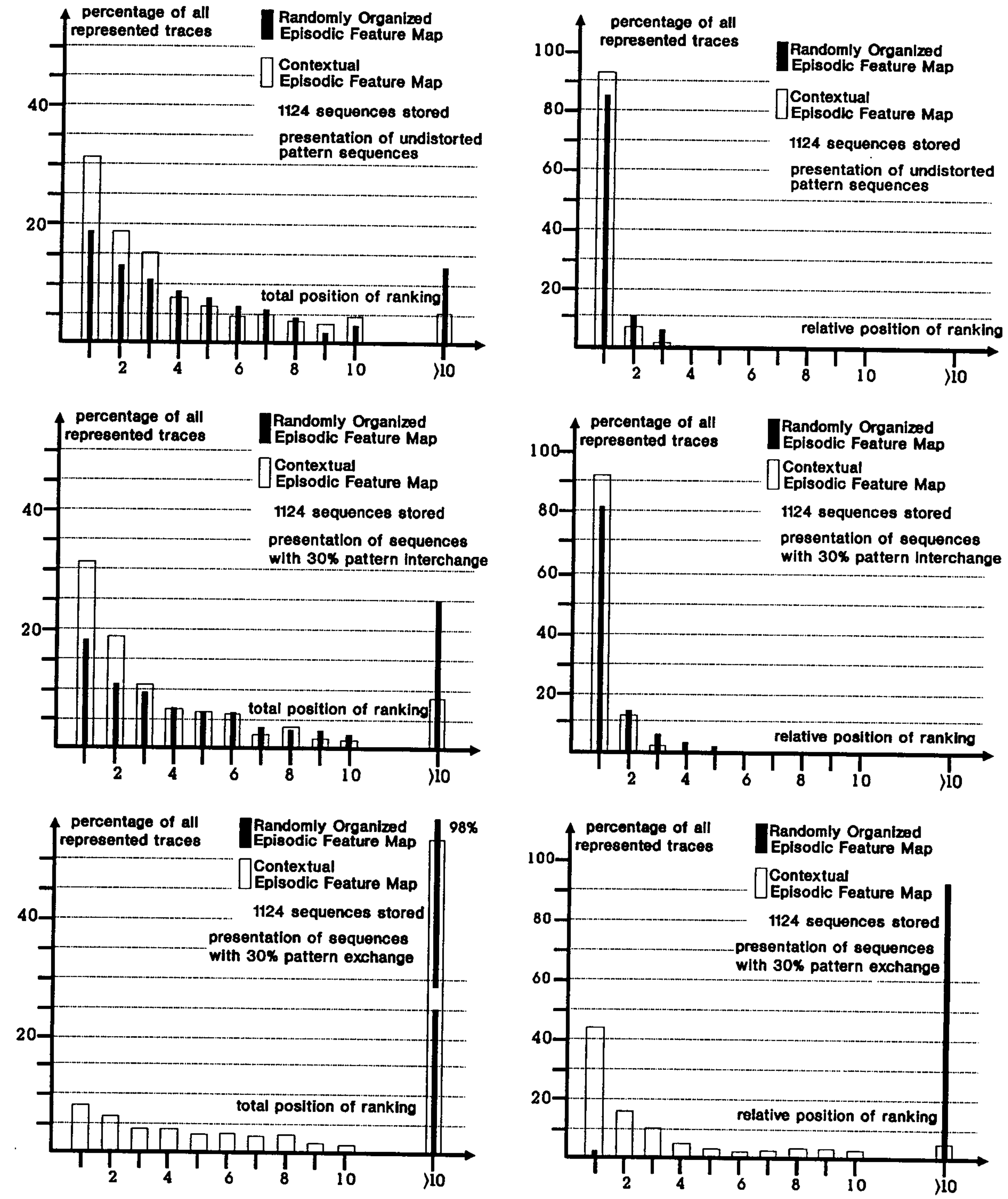

Figure 6 *gives the simulation results for an Episodic Feature Map which is trained with 1124 input pattern sequences.* **Upper row:** *Recall behaviour during presentation of undistorted pattern sequences;* **Middle row:** *30% of the patterns of all sequences were interchanged (altering the order of the patterns);* **Bottom row:** *30% of the patterns of all presented sequences were substituted by patterns not contained in the concerning original sequences*

Outlook

The neural network architecture proposed in this paper forms a part of a very complex model architecture which is to organize a behaviour-oriented scene interpretation in attentional vision (see [8]). In this architecture a network model very similar to the model described above interacts with a network architecture outlined in [9] which is able to aquire stable transitions between certain features with the corresponding spatial relations. Here our proposed model architecture is to establish knowledge about feature transitions that form stable complex visual structures.

It is no question that the separation into a learning phase and a following recall phase cannot be motivated in a biological sense. Therefore we will give up this separation to allow the system to generate hypotheses and to include new traces in the same processing mode. Here we have to use mechanisms for self-organizing the *Episodic Feature Map* in a changing environment, so the simulation will become much more expensive.

References

[1] Boehme, H.-J. and Koerner, E. (1991) A Neural Network Model for Control and Stabilization of Reverberating Pattern Sequences. Proceedings of ICANN-91, 1121-1124

[2] Gross,H.-M., Boehme,H.-J., Heinke,D., Moeller,R., and Pomierski,T. (1992) Steuerung parallel-sequentieller Verarbeitungsprozesse und Strukturierung dynamischer Repraesentationen. Informatik-Fachberichte, BMFT-Neuroinformatik-Statusseminar

[3] Koerner,E. and Boehme,H.-J. (1991) Organization of an Episodic Knowledge Base. Proceedings of ICANN-91, 873–878

[4] Kohonen, T. (1991) The Hypermap Architecture. Proceedings of ICANN-91, 1357-1360

[5] Miikkulainen, R. (1992) Trace feature map: a model of episodic associative memory. Biological Cybernetics, 66, 273-282

[6] Wolfram,S. (1986) Theory and Applications of Cellular Automata, World Scientific Publishing, Singapore

[7] Kauffman, S.A. (1984) Emergent Properties in Random Complex Automata, Physica 10D, pp.145-156

[8] Gross, H.-M., Boehme, H.-J., Heinke, D., and Pomierski, T. (1994) An Approach to Behaviour-Oriented Object-Understanding in Attentional Neurocomputing, this volume

[9] Heinke, D. and Gross, H.-M. (1994) A Neural Network Architecture for Self-Organization of Object Understanding, this volume