ELSEVIER

Neurocomputing 12 (1996) 267–287

NEUROCOMPUTING

Approximate geometry representations and sensory fusion

Csaba Szepesvári [a,b,*], András Lőrincz [b]

[a] *Department of Photophysics, Institute of Isotopes of the Hungarian Academy of Sciences, P.O. Box 77, Budapest, Hungary, H-1525*

[b] *Bolyai Institute of Mathematics, University of Szeged, Szeged, Hungary, H-6720*

Received 20 November 1994; accepted 2 October 1995

Abstract

Information from the external world goes through various transformations. The learning of the original neighbourhood relations of the world using only the transformed information is examined in detail. An approximate representation consists of a finite number of discretizing points and connections between neighbouring points. The goal here is to develop the theory of self-organizing approximate representations. Such a self-organizing system may be considered as a generalization of the Kohonen topographical map that we now equip with self-organizing neigbouring connections. For illustrative purposes an example is presented for sensory fusion: the geometry of the 3D world is learned using the outputs of two cameras.

Keywords: Self-organizing networks; Sensory fusion; Geometry representation; Topographical mapping; Kohonen network

1. Introduction

Knowledge of the geometry of the world external to a system is essential in cases such as navigation when for predicting the trajectories of moving objects [21,16,25,4,11,35]. It may also play a role in recognition tasks, particularly when the procedures used for image segmentation and feature extraction utilize information on the geometry [2,22]. A more abstract example is function approximation, where this information is used to create better interpolation [15,8,3].

This paper summarizes the recent advances in the theory of self-organizing development of approximate geometry representations based on the use of neural

* Corresponding author. Email: szepes@math.u-szeged.hu

SSDI 0925-2312(95)00116-6

networks. The first results of this field were published in [24,9,36,34]. Some theoretical results were developed in [23] and [32].

Part of this work is based on the theoretical approach of [32], which is different from that of [23] and also is somewhat more general. The Martinetz approach [23] treats signals provided by artificial neuron-like entities whereas the present work uses the entities of the external world as its starting point. The relationship between the present work and the Martinetz approach will be detailed.

We approach the problem of approximate geometry representations by first examining the problem of sensory fusion, i.e. the problem of fusing information from different transductors. A straightforward solution is the simultaneous discretization of the output of all transductors, which means the discretization of a space defined as the product of the individual transductor output spaces. However, the geometry relations are defined for the *external world* only, so it is still an open question how to define the metrics on the product of output spaces. It will be shown that simple Hebbian learning can result in the formation of a correct geometry representation. Some mathematical considerations will be presented to help us clarify the underlying concepts and assumptions. The mathematical framework gives rise to a corollary on the "topographical mappings" realized by Kohonen networks. In fact, the present work as well as [23] may be considered as a generalization of Kohonen's topographic maps. We develop topographic maps with self-organizing interneural connections. The crucial point is whether the neighbouring connections, i.e. the geometry representation, is correct or not. To this end the initial step should be to develop the concept of correct geometry representation.

Extensive use will be made of the fact that a neural network may be viewed as finite graph. The concepts of connection based neighbourhood graph (CBN-Graph), the geometry induced neighbourhood graph (GIN-Graph) and the similarity based neighbourhood graph (SBN-Graph) will be introduced and their relationship with geometry representation will be studied.

The paper is organized as follows: In Section 2 we briefly review adaptive pattern space disretization (Section 2.1) and the formation of local filters by 'winner-takes-all' networks (Section 2.2). The issue of geometry representation of the external space is raised in Section 2.3. The GIN-Graph and the CBN-Graph are introduced in Section 3, together with the concept of correct geometry representation of the external space geometry. Section 3.1 considers Hebbian learning for the correct geometry representation of the external space. The representation of pattern space geometry based on the SBN-Graph and its relationship with Kohonen networks are discussed in Section 3.2. In Section 3.3 we analyse how the so-called 'competitive' Hebbian law is linked with the representation of the pattern space.

2. Discretization and neighbourhood

Let us consider Fig. 1 in which the external space is transformed into a multidimensional one by a set of transductors. Transductors are tools used by

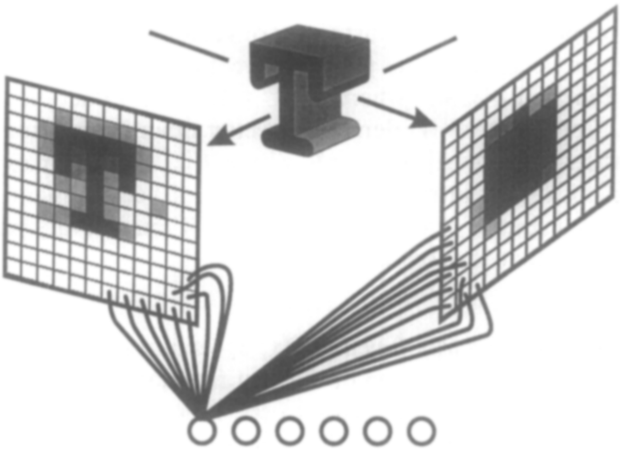

Fig. 1. Two cameras, i.e. two sets of transductors, monitoring the same 3D region. The objects of the external world are 3 dimensional. The input of the cameras corresponds to the projections of the objects. The bottom part of the figure shows neurons and a few feedforward connections of the first neuron. The pattern space that – consists of the output of cameras as a row vector – is 288 dimensional.

embedded systems [17] to measure various properties of their environment. A system only obtains information that comes through its transductors. In the mathematical sense a transductors is a mapping of the external world onto another space that we shall call the *pattern space*. An important question is how to explore geometrical relations in the external world by the information of the transductors, i.e., the information available in the pattern space. Henceforth we shall assume that the pattern space is a subset of the n-dimensional Euclidean space $\mathbf{R}^n$.

Although the dimensionality of the external world is generally low, that of the pattern space can be tremendous. To give an example a 10 by 10 discretization of a 2D world results in a pattern space embedded into a 100 dimensional space. A possible task for the first processing stage of the embedded system is to reduce dimensionality with the help of the correlations in the transductor signals. Vector quantization (or pattern space discretization) is such an example.

2.1 Adaptive vector quantization

Let us first assume that we have a fixed number of fixed pattern vectors, the *prototype vectors*. Let us number these vectors from 1 to k and denote the individual vectors by $\mathbf{w}_1, \ldots, \mathbf{w}_k$. Henceforth the functions whose domain is the pattern space and whose image space is the set $W = \{\mathbf{w}_1, \ldots, \mathbf{w}_k\}$ will be called *vector quantization functions*. Usually, the number of prototype vectors is less than the dimensionality of the pattern space and thus the vector quantization function reduces the dimensionality of the data. Discrete time is assumed and let us also assume that the pattern signals at different time steps are independent samples of a common underlying probability distribution over the pattern space. This distribution function is denoted by F. The distortion of vector quantization, i.e., the cost of replacing a pattern vector $\mathbf{x}$ by the prototype vector $\mathbf{w}_i$ is taken as being somehow defined by the function $d(\mathbf{x}, \mathbf{w}_i)$, where d: $\mathbf{R}^n \times \mathbf{R}^n \to \mathbf{R}$. Function d is assumed to be continuous. Let $G: \mathbf{R}^n \to W$ denote a vector quantization function. The expected cost is then given by $E[d(\mathbf{x}, G(\mathbf{x})]$, where $E[\cdot]$ stands for expectation. This is the expected distortion introduced by the use of vector quantization G. The optimal vector quantization of function $G^*: \mathbf{R}^n \to W$ maps every pattern vector to prototype vectors so that the loss is minimal [12]:

$$d(\mathbf{x}, G^*(\mathbf{x})) \leq d(\mathbf{x}, \mathbf{w}_i), \quad i = 1, \dots, k.$$

A 'reward function' or *similarity function* $S(\mathbf{x}, \mathbf{w}_i)$ is introduced that is a monotonically decreasing, continuous function of the cost function. In terms of the similarity function, G^* maps every pattern vector to the most similar prototype vector. In the rest of the paper we shall use the similarity function instead of the cost function as it fits our description better.

A simple incremental adaptive process that determines optimally distributed prototype vectors is known as a variant of the k-means algorithm of statistics [20], or the 'winner-takes-all' procedure from neural networks [29,39,31,14,10]. To every pattern vector $\mathbf{x}$ the algorithm computes the best matching prototype vector (the winner) and moves it into the direction of the inputted pattern vector:

$$\Delta\mathbf{w}_i = \alpha G_i(\mathbf{w}_1, \dots, \mathbf{w}_k; \mathbf{x})(\mathbf{x} - \mathbf{w}_i). \tag{1}$$

Here $G_i(\mathbf{w}_1, \dots, \mathbf{w}_k; \mathbf{x}) = 1$ only for index i for which $S(\mathbf{x}, \mathbf{w}_i)$ is maximal and 0 otherwise, α is the so called learning rate, $0 < \alpha < 1$. In the language of artificial neural networks the prototype vectors are called weight vectors and the processing units - the artificial neurons - are said to store these weight vectors as the strength of their connections to the input neurons. The signal of the input neurons provides the components of the inputted pattern vector. Determination of the winner, often called the competition, can be implemented by continuous time parallel processes [13].

If the learning rate α decreases in time in an appropriate fashion (see e.g., [40]) then the process is convergent and the prototype vectors $\mathbf{w}_1, \dots, \mathbf{w}_k$ converge to optimal prototype vectors. On the other hand, if α is kept to a small constant value then the prototype vectors become sample trajectories of a stochastic process. These aspects were studied by Amari [1], who showed that the expected values of the stochastic prototype vectors converge to the set of optimal prototype vectors, and the deviation of prototype vectors is proportional to $\sqrt{\alpha}$ in the limit and the speed of convergence is proportional to $1 - \alpha$.

Let D_i denote the winning region of neuron i:

$$D_i = \{\mathbf{x} \in X \mid S(\mathbf{x}, \mathbf{w}_i) \geq S(\mathbf{x}, \mathbf{w}_j),\ j \neq i\} \tag{2}$$

where X denotes the support of F, the probability distribution of input patterns in the pattern space $\mathbf{R}^n$. [1] The set D_i is called the masked Voronoi polyhedron by Martinetz [23]. At (stochastic) equilibrium the ith prototype vector is given by

$$\mathbf{w}_i^* = E[\mathbf{x} \mid D_i], \tag{3}$$

the conditional expectation of pattern vectors over the winning region of the ith prototype vector.

2.2 Spatial filters by competitive learning

Let us assume that the objects of external space are mapped by an n-pixel camera to an n dimensional pattern space (without any reference to the proximity relations between the pixels). Let us also assume that the objects are presented at different random positions that are uniformly distributed in the external space and let us choose the inner product function for a similarity function. $S(\mathbf{p}, \mathbf{q}) = \sum_{i=1}^{n} p_i q_i$: the more the two pattern vectors 'overlap', the more 'similar' they are. Let the 'position' of a pattern vector $\mathbf{x}$ on the camera be

$$\bar{\mathbf{x}} = \sum_{i=1}^{n} x_i \mathbf{r}_i / \sum_{i=1}^{n} x_i,$$

where $\mathbf{r}_i$ is the position of the centre of the ith pixel on the 2D surface of the camera; the 'size' is given by

$$\sigma(\mathbf{x}) = \sum_{i=1}^{n} (\mathbf{r}_i - \bar{\mathbf{x}})^2 x_i / \sum_{i=1}^{n} x_i,$$

which is the average radius of the patterns on the camera. A prototype vector is said to be local on the camera if its size is small compared to that of the camera. Numerical simulations with learning rule given in (1) show that the equilibrium prototype vectors represent localized areas on the camera (Fig. 2) provided that localized input vectors are presented (see e.g. [27,33]).

Schulten et. al. have put forward a heuristic argument [30,27] to show that localized input vectors result in localized prototype vectors. (The learning equation they used is slightly different from ours in that they normalized the prototype vectors after each learning step.) It was argued that the size of prototype vectors defined on the camera corresponds to the average size of input pattern vectors. Numerical simulations were also presented in [26] on a Kohonen network.

2.3 The problem of neighbourhood relations

Since the prototype vectors are local on the camera, it might seem that one can define a neighbourhood relation between them in an obvious fashion: two prototype vectors are considered as neighbours if they overlap on the camera. This definition, however, fails in more general situations. As an example let us consider the case of a pattern space consisting of gray-scale outputs of *two* cameras monitoring the same 3D region (see Fig. 1). The external objects are 3 dimensional now and the cameras only monitor projections of them. (In Fig. 1 the darker pixels

[1] The support of F, supp(F) is defined by $\cap\{U^c \mid F(U) = 0,\ U \in \Omega\}$ where U^c is the complementer of U and Ω is the set of measurable sets.

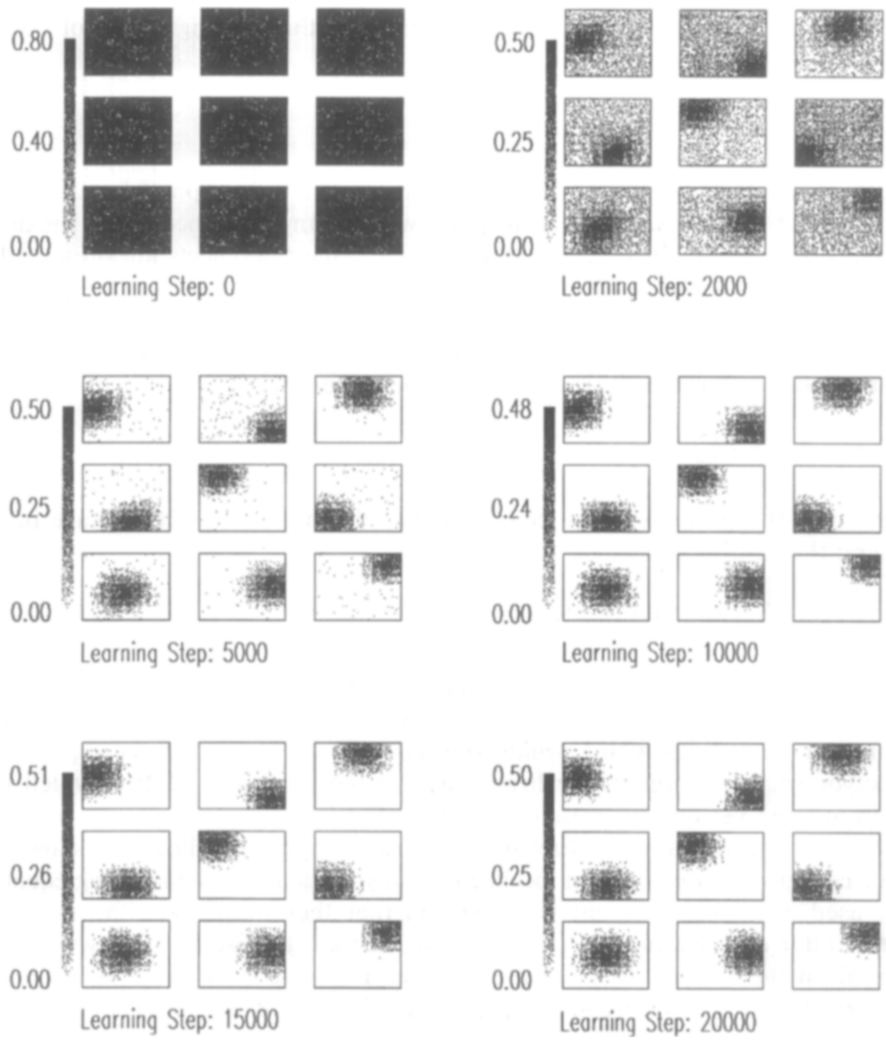

Fig. 2. Self-organized filter formation during training that utilizes learning rule (1). The numbers indicate the training steps. Different localised inputs at different random positions served as training examples. The filters are formed during the first 5000 steps. The learning rate was kept constant. During later steps the configuration undergoes minor modifications in accordance with the random object generation, but stays stable. The figure was drawn by generating pixels randomly with probabilities that correspond to the gray scale values.

correspond to higher outputs.) It is easy to see that two patterns corresponding to distant objects and each representing 3D regions may also overlap: if the surfaces of the two cameras are not parallel then two objects may overlap on the first camera while they may still be arbitrarily far from each other along the direction that is parallel to the surface of the second camera. The crucial point is that the pattern space geometry does not necessarily correspond to the external space geometry.

The general definition of the neighbourhood of pattern vectors induced by the external space geometry may be sketched as follows: The reverse of the pattern formation on the cameras renders a region of the 3D space to every pattern vector, viz. the region obtained by the union of objects that are mapped onto the given pattern vector. This region will be called the *pre-image* of the pattern vector (the exact definition will be given in the next section). Two regions of the external space are neighbours if there exists a local object that overlaps with both of them. Two pattern vectors 'overlap' in the pattern space only if their pre-images are neighbours in the external space.

It is now clear that one cannot define the meaning of preservation of external space neighbourhood relations without considering the mapping $\mathscr{P}$ of the external world to the internal pattern space. $\mathscr{P}$ maps the configurations in the external space into the pattern space. This can be formalized by saying that $\mathscr{P}$ maps subsets of the external space E to the pattern space $\mathbf{R}^n$:

$$\mathscr{P}: 2^E \to \mathbf{R}^n.$$

E is taken to be a topological space. We shall proceed at this level of generality and no further details of the pattern formation process will be assumed. For sensory fusion the example of pattern formation with two cameras is chosen. Accordingly we define a possible mathematical interpretation of this mapping as follows. The external space is divided into a finite number of disjoint parts that correspond to the sensitivity regions of the camera pixels; $E = \bigcup_{i=1}^{n} E_i$, where $E_i \cap E_j = \emptyset$. The pattern space is the n dimensional hypercube and $\mathscr{P}: 2^E \to [0, 1]^n$ is given by

$$\mathscr{P}(O)_i = \frac{\mu(O \cap E_i)}{\mu(E_i)},$$

where O is a subset of E, the ith component of the pattern vector is $\mathscr{P}(O)_i$, and μ is a Borel measure (volume) on E [2].

3. Geometry representation by lateral connections

As outlined above, neighbourship of pattern vectors should be based on the concept of neighbourship in the external space. To this end we project the pattern vectors back to the external world by defining their pre-images:

Definition 3.1. *The* pre-image of pattern vector $\mathbf{x} \in \mathscr{P}(\mathbf{R}^n)$ *is the union of all objects (subsets of the external space) that are mapped by* $\mathscr{P}$ *to* $\mathbf{x}$, *this being the union of the inverse images of* $\mathbf{x}$:

$$R(\mathbf{x}) = \cup \mathscr{P}^{-1}(\mathbf{x}).$$

We postulate the assumption of local presentations that is needed to develop the concept of neighbourship in the pattern space:

[2] O is assumed to be measurable by μ, and $\mu(E_i)$ is assumed to be nonzero for all i.

Assumption 3.1. *We assume that there is a set* $\mathbf{O}$ *of 'local' objects in the external world and a probability distribution on* $\mathbf{O}$ *and also that while learning, the objects are independently presented from* $\mathbf{O}$ *in accordance with the given probability distribution.*

For the rest of the paper we fix both $\mathbf{O}$ and the probability distribution. As a consequence we shall have an induced distribution in the pattern space that is denoted by F. The objects in $\mathbf{O}$ are assumed to be 'local' thus two points of the external space are assumed to be 'close' if there is an object in $\mathbf{O}$ that overlaps with both points. In order to measure the overlap of two sets we fix a Borel measure μ on the external space, E. We say that two sets overlap significantly if their intersection has a nonzero μ-measure. Then the neighbourship in the pattern space may be defined in accordance with the following:

Definition 3.2. *Two vectors of the pattern space are neighbours if the probability that an object from* $\mathbf{O}$ *overlaps significantly with the pre-images of the two vectors is nonzero. That is,* $\mathbf{x}, \mathbf{y} \in \mathbf{R}^n$ *are neighbours if* $\text{Prob}(\{O \in \mathbf{O} \mid \mu(O \cap R(\mathbf{x}))\mu(O \cap R(\mathbf{y})) > 0\}) > 0$.

In other words, *the geometry of the space is assumed to be encoded into the local nature (smallness) of the objects that are inputted into the algorithm.* An approximate geometry representation of the geometry of the *external space* E is a graph that is embedded into the *pattern space.*

Definition 3.3. *Let* $W = \{\mathbf{w}_1, \ldots, \mathbf{w}_k\}$ *be a finite set of points in the pattern space. The* geometry induced neighbourhood graph *(GIN-Graph) of W is the graph that has k nodes: every node corresponds to one element of W and two nodes are connected if and only if the corresponding elements of W are neighbours in the sense of Definition 3.2.*

Embedding of a graph into a space is a mapping from the node set of the graph to the space. An embedded finite graph can be viewed as a finite set of points in the embedding space and the connections between these points. If two embedded graphs are mapped onto the same set of points then we say that they have a *common embedding*. In this case we consider them as having a common node set. From now on if we find that two graphs have a common embedding then we shall assume that they share a common node set. We say that a graph is *perfectly matched* to another graph based on their common embedding if they have a common embedding and the connection structure of the graphs is the same. Similarly, we say that a graph G_1 is a *subgraph* of another graph G_2 *based on a common embedding*, if they have a common embedding and the connection structure of G_2 is a subset of the connection structure of G_2. In this latter case there may exist edges in G_2 that are not in G_1. We can now define the meaning of correct geometry representation.

Definition 3.4. *Let* (G, e) *be an approximate geometry representation, where G is a finite graph and e is its embedding into the pattern space. Let W be the image of the* *node set of G in the pattern space and let G_I be the GIN-Graph induced by the set W. If G is perfectly matched to G_I based on their common embedding W, then we say that graph G represents correctly the geometry of the external space.*

Since a vector quantization network defines a finite number of points in the pattern space, the obvious step is to extend this network by lateral connections in order to obtain a geometry-representing network. Let the lateral connection between neuron i and j be $q_{ij} \in [0, 1]$. Then every configuration of the network defines an approximate geometry representation, which is not necessarily a correct representation. Two prototype vectors, $\mathbf{w}_i$ and $\mathbf{w}_j$ are connected if the lateral connection between the corresponding neurons is nonzero. More precisely, the node set of the graph of the representation is given by $\{1, \ldots, k\}$; two nodes (i, j) are connected in the graph if and only if $q_{ij} > 0$ and the embedding of the graph is given by $e : i \mapsto \mathbf{w}_i$.

Definition 3.5. *The connection based neighbourhood graph (CBN-Graph) is the graph that has k nodes numbered from 1 to k and the arbitrary nodes i,j are connected in the graph only if the connection between the corresponding neurons is nonzero, i.e. if $q_{ij} > 0$.*

Thus we may examine whether a network equipped with lateral connections correctly represents the geometry of the external space or not.

3.1 Theory of correct geometry representations

In this section we shall examine the use of the Hebbian law in the formation of adaptive geometry representations. In its simplest form the Hebbian law changes the connection strength between neurons i and j if they are activated simultaneously: [3]

$$\Delta q_{ij} = \beta(S_i S_j - q_{ij}), \quad i \neq j. \tag{4}$$

Here we used the product of the input intensities of neurons to represent the simultaneous activation, β is (another) learning rate with $0 < \beta < 1$. The input intensity of neuron k (S_k) is given by the similarity of its prototype vector $\mathbf{w}_k$ to the actual input vector $\mathbf{x}$: $S_k = S(\mathbf{x}, \mathbf{w}_k)$. Using this rule the lateral connections converge to the expected joint similarity of neurons in the limit (i.e., to $E[S(\mathbf{x}, \mathbf{w}_i^*)S(\mathbf{x}, \mathbf{w}_j^*)]$), provided that the weight vectors converge to the pattern vectors $\mathbf{w}_k^*$ and β goes to zero in an appropriate fashion [32] [4]. Thus we need not wait until the feedforward weights settle but rather we may use Eqs. (1) and (4) *simultaneously*.

[3] Many authors call this law the signal Hebbian law since the original does not contain a decay term.

[4] The proof, that relies on the law of large numbers, requires a further condition on S, namely its continuity in the second argument independently of the first argument.

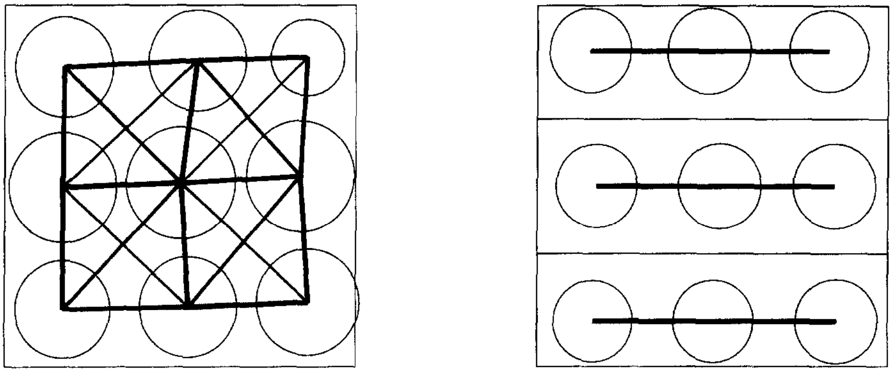

Fig. 3. Learnt approximate geometry representations of different external space geometries. Circles show the 'sizes' of the prototype vectors. Circles are depicted at the 'positions' of the prototype vectors (see text for the explanation of 'size' and 'position'). The connection thickness represents the strength of the topology connections q_{kl}. In the left hand side figure the objects were presented everywhere in the two dimensional box. In the right hand side figure the objects were presented along three horizontal stripes in the two dimensional box with arbitrary ordinates. The lack of a link indicates asymptotically zero strength connections.

Remember that positive lateral connections are required to represent proximity relations of the external space in the limit. As an example let us consider again the case in which the pattern space is the output of a camera that monitors a 2D world and the similarity function is the inner product function. It should also be remembered that the weight vectors tend to form 'local filters' on the retina. The strength of the lateral connection between neurons i and j then converges to a value proportional to the overlap of the sections of each limit filter and the average of its inputs patterns. In this way 'distant' filters will have zero connections whereas 'close' filters tend to have strong connections.

According to Definition 3.3 two prototype vectors should be connected if and only if with non-zero probability there is an object in set **O** that overlaps with the pre-images of the prototype vectors. Fig. 3 shows two examples when the similarity function is the inner product function. Although this similarity function works well in this case it is not always suitable. Consider now the example of two cameras and let us suppose that the monitored region is a 3D cube and that the objects are shown only in a horseshoe shaped region of it. The developed connection structure is depicted in Fig. 4. It can be seen that nonzero connections develop between neurons that represent distant regions in the sense of Definition 3.2 The reason for the incorrect representation is that the 'overlap' on the union of cameras (defined by the inner product of patterns) can be non-zero even if the pre-images of the pattern vectors do not overlap. A little change in the similarity function rectifies the geometry representation and provides a suitable method for sensory fusion. The proposed neural activation function can be computed by multiplying two inner products, one for the first camera and one for the other. This implements a simple AND function: an object must overlap with the prototype vector of a neuron on

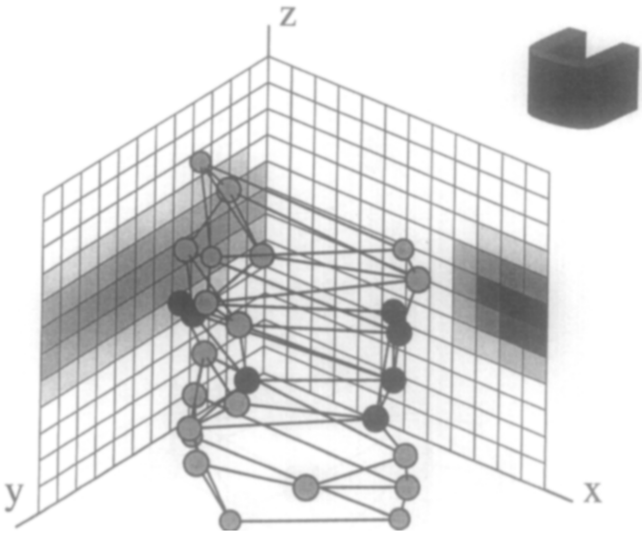

Fig. 4. Learnt neighbouring relations in 3 dimensions, I. Two 2-dimensional projections of randomly positioned 3 dimensional objects were inputted to the neural network. The objects came from a U-shaped 3-dimensional volume. The spheres representing neurons are given by the centre and the radius of their prototype vectors. The neurons that developed lateral connections are linked by lines. One neuron (black) with its prototype vectors projected back to the camera surfaces is shown in the figure. Its neighbours – those neurons developing connections to the 'black' neuron – are coloured dark gray. All other neurons are light gray. The similarity function (the inner product here) is not 'overlap' preserving. A composite pattern vector is formed from the pictures of the cameras and the winner-takes-all paradigm is applied. Pre-images become extended at least on one of the cameras (see grey transductors corresponding to the input connection strengths of the black neuron) and lateral connections develop between neurons from distant regions.

both of the cameras in order to excite it. The connection structure that develops with this similarity function is shown in Fig. 5. In this case the connection structure is a correct representation of the 3D connectivity of the horseshoe.

Now the sufficient conditions that ensure correct geometry representation may be discussed. The central point is the *overlap preserving* property of the similarity function.

Definition 3.6. *Let the external space E be a topological space, $\mathscr{P}: 2^E \to [0, 1]^n$ be an arbitrary function and Y the image of E under $\mathscr{P}$. The function $S: Y^2 \to [0, 1]$ is said to be* overlap preserving *if the following condition holds: whenever two open sets of the external space overlap significantly the similarity of their image under $\mathscr{P}$ (the similarity measured by S) is greater than zero, and vice versa. (This is one form of the separability condition discussed in [32].)*

This means that if $U, V \subset E$ are open sets and $\mu(U \cap V) > 0$ holds, where μ is the fixed Borel measure on E, then $S(\mathscr{P}(U), \mathscr{P}(V)) > 0$ holds, and vice versa. Henceforth we shall assume that function $\mathscr{P}$ is monotonic with respect to the inclusion

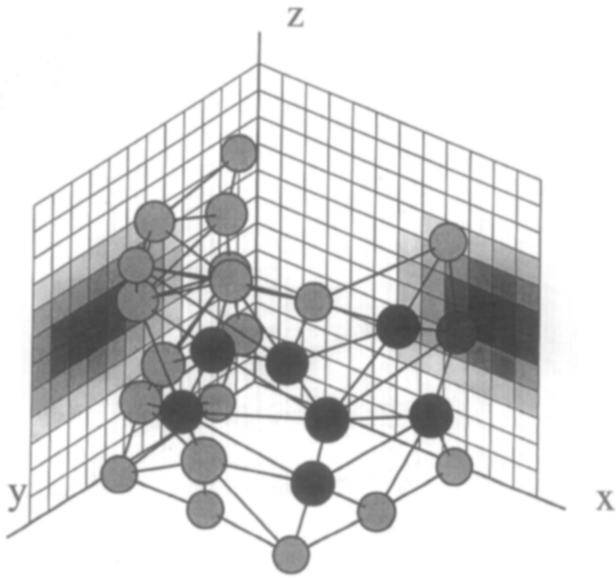

Fig. 5. Learnt neighbouring relations in 3 dimensions, II. For details of simulation and presentation see caption of Fig. 4. Here, the similarity function is the product of separate inner products on both cameras. This function is overlap preserving with respect to the transduction function. Learnt neighbouring relations are correct.

among subsets of the external space: This means that $U \subset V$ implies $\mathscr{P}(U) \leq \mathscr{P}(V)$. [5] Then the following statement holds:

Proposition 3.1. *Let us assume that S is overlap preserving and $\mathscr{P}$ is monotonic. Then for any pair of* $\mathbf{x}$ *and* $\mathbf{y}$ *pattern vectors* $\mu(R(\mathbf{x}) \cap R(\mathbf{y})) = 0$ *implies that* $S(\mathbf{x}, \mathbf{y}) = 0$.

Proof. Remember that $R(\mathbf{x})$ defines the pre-image of pattern vector $\mathbf{x}$ by $\cup \mathscr{P}^{-1}(\mathbf{x})$. Then from $\mathbf{x} = \mathscr{P}(U)$ it follows that $U \subset R(\mathbf{x})$. This together with the assumptions proves the statements. □

Two other conditions to be satisfied are that (i) the similarity function S is monotonic in any of its variables provided that the other is fixed and (ii) $\mathscr{P} \circ R$ preserves the zero components: for all pattern vectors $\mathbf{x} \geq 0$ there exists a constant c (c may depend on $\mathbf{x}$) for which the inequality $\mathscr{P}(R(\mathbf{x})) \leq c\mathbf{x}$ is satisfied for all of the components. [6] To summarize we assume that the following assumption holds:

[5] As usual, the comparison of vectors means the comparison of their components.

[6] Another set of more general but more technical conditions was proposed in [32].

Assumption 3.2. *$\mathscr{P}$ and S are monotonic functions and S is overlap preserving with respect to $\mathscr{P}$, and $\mathscr{P} \circ R$ preserves the zero components.*

The following theorem provides an efficient tool for geometry learning:

Theorem 3.1. *If Assumptions 3.1 and 3.2 hold and a competitive network is trained according to Equations (1) and (4) in such a way that its weights converge, then the GIN-Graph and CBN-Graph will be congruent in the limit.*

The theorem says that a 'winner-takes-all' network equipped with lateral weights trained with the Hebbian law (4) will settle in a configuration where the lateral connection structure of the network represents the geometry of the external world in a correct fashion provided that the similarity function, which is used by the neurons for competition and lateral weight learning, is overlap preserving.

It is easy to see that in the experiments presented the critical property is the overlap preservation. In 2D experiments, overlap preservation holds for the inner product function whereas with two cameras this is no longer true. In this case for example, the proposed 'product of inner products' is overlap preserving. Thus the theory supports that choice.

But how can one find an appropriate similarity function in practice? If the pattern space is the direct sum of separate pattern spaces with overlap preserving similarity functions on them, then the *product* of the similarity functions may be good:

Proposition 3.2. *Let us assume that $\mathscr{P}_1$ and $\mathscr{P}_2$ map objects of the same external space to pattern spaces X_1 and X_2, respectively. Let us assume further that the similarity functions S_1 and S_2, defined over $X_1 \times X_1$ and $X_2 \times X_2$, respectively, are overlap preserving and monotonic functions. Then $S = S_1 S_2$ defined over $X_1 \oplus X_2$ is also overlap preserving and monotonic.*

Note the condition that both pattern spaces are the image spaces of the *same* external space. One question that remains unanswered is how one finds suitable partitions of the pattern space if the partitions are not given in advance.

The Kohonen network can be viewed as a network that has an inherited CBN-Graph to be used for neighbour training. The Kohonen network forms a 'topographic' mapping if the CBN-Graph of the network matches the geometry of the *pattern space*. Concepts related to the approximate representation of the geometry of the pattern space are introduced and examined in the next two sections.

3.2 'Topographic mapping' and Kohonen networks

Kohonen networks are usually defined on a two (or higher) 'dimensional' lattice of neurons. Each neuron has previously defined neighbours. This proximity is further extended to 'distances' between arbitrary pairs of neurons. [7] Let the distance between neurons i and j be d_{ij}. Eq. (1) is then extended to allow *neighbour training*. Let us assume that a pattern vector $\mathbf{x}$ is presented to the network. If $\mathbf{w}_i$ denotes the prototype vector that is the most similar to the pattern vector $\mathbf{x}$ then the prototype vectors are modified according to

$$\Delta \mathbf{w}_j = \alpha K(d_{ij})(\mathbf{x} - \mathbf{w}_j), \tag{5}$$

where K is a positive, monotonically decreasing function that satisfies $K(1) = 1$. Usually K changes in time: it becomes sharper as time proceeds. Note that by taking $q_{ij} = K(d_{ij})$ the integers that represent distances between the neurons are transformed into lateral connections representing closeness. The fact that K becomes sharper as time proceeds may be expressed by saying that those neurons initially 'close to each other' become more and more 'distant' by time. Even though the convergence properties of this net have been studied by several authors [18,5,28,37], one important property of this network, viz. that it induces a topographical mapping that maps the neuronal lattice into the pattern space, has not yet been fully explored. One reason is that the concept of correct topographical mapping was just intuitively defined. In some cases, however, a precise meaning is associated with it: for example, in one dimensional input spaces the topographical mapping corresponds to ordering [19]. Alas, the concept of an ordering based topographical mapping cannot easily be extended to higher dimensional spaces [6]. Often the continuity is used to define intuitively the concept of it as in [15,38]. The domain of the considered mapping, however, consists of only a *finite* set of points (the nodes of a lattice are mapped to the pattern space by assigning the prototype vector of neuron i to node i). If the topology is based on a finite set then the only topologies available are the empty ones and the discrete ones. This is a serious restriction since in the case of empty topology *there is no continuous mapping* (provided that the image space has at least two points); whereas in the case of discrete topology *every mapping is continuous*. Thus the concept of continuity of topographical mapping is a dead end (as long as the domain of mapping is assumed to be the neuronal lattice). On the other hand, the idea that neurons with prototype vectors close to each other in the pattern space should be close to each other in the lattice as well, can be formulated precisely. Let us consider the winning regions of the prototype vectors, i.e. the sets D_i (see Eq. (2)). Two prototype vectors should be considered as *neighbours in the pattern space* if their winning regions share a common side. In this case the *common winning region* of the two prototype vectors is nonempty, since $D_i \cap D_j \subset C_{ij}$. The common winning region of neurons i and j is defined by

$$C_{ij} = \left\{ x \in X \mid \min\left(S(\mathbf{x}, \mathbf{w}_i), S(\mathbf{x}, \mathbf{w}_j)\right) \geq S(\mathbf{x}, \mathbf{w}_k), \ k \neq i, j \right\}. \tag{6}$$

In [23] this set is called the second order masked Voronoi polyhedron. The

[7] Note that the lattice of neurons can be considered as a finite graph. The distance of two nodes in a graph is the length (i.e. the number of edges) of the shortest path between the two nodes.

similarity based neighbourhood graph reflects the proximity of prototype vectors *in the pattern space* [32]:

Definition 3.7. *The* similarity based neighbourhood graph *(SBN-Graph) defined by prototype vectors* $W = \{\mathbf{w}_1, \ldots, \mathbf{w}_k\}$ *is a graph that is embedded into the pattern space onto the set W and two nodes of the graph are connected only if their embedded images are neighbours in the pattern space X.* [8] *In other words, the vectors* $\mathbf{w}_i$ *and* $\mathbf{w}_j$ *are neighbours in the pattern space if and only if* $C_{ij} \neq \emptyset$.

Note that the concept of neighbourship in the pattern space relies on the set of prototype vectors and the similarity function. In other words, we may obtain different "geometries" over the same pattern space X by choosing different similarity functions. (An example is presented in the next section.)

The lattice structure of the Kohonen network may be considered as a graph that defines which nodes are immediate neighbours in the 'topology' of the network. This will be called the *inherited neighbourhood graph* (IN-Graph). (In fact one frame of the so called Kohonen movie is the picture that depicts the embedded IN-Graph in the pattern space.) Now we are in the position to define a correct topological mapping [32]:

Definition 3.8. *The Kohonen network, with a structure defined by the IN-Graph* G_I*, is said to define a correct topographical mapping if the IN-Graph* G_I *is a subgraph of the SBN-Graph* G_S *where the matching of graphs is based on their common embedding.*

If the two graphs are exactly matched then we say that the configuration of the network defines a correct topographical mapping *in the strong sense*. This correctness in the strong sense corresponds to the 'topology preservation' of the work of [23]. Note that sometimes even the broader Definition 3.8 is too strict: there are mappings that one would like to consider to be correct but in the sense of Definition 3.8 they are not (see Fig. 6 that will be discussed later and also the example presented in [7]).

Correct topographical mapping in the sense of Definition 3.8 means that the distance of nodes measured in the IN-Graph G_I is always greater than the distance of the same nodes measured in the SBN-Graph G_S. This can be used to define a measure of defect of the topographic mapping:

$$D = \max_{i,j}\left(d_{G_S}(i, j) - d_{G_I}(i, j)\right). \qquad (7)$$

Here $d_G(i, j)$ is the distance of nodes i, j in graph G. If $G_I \subset G_S$, i.e. if the topographical mapping is correct, then $d_{G_S}(i, j) - d_{G_I}(i, j)$ is non-positive for all pairs of i, j and zero for all edges of the IN-Graph G_I. Thus a positive difference

[8] Note that the pattern space X is assumed to be the support of the probability distribution of pattern vectors and not the whole $\mathbf{R}^n$

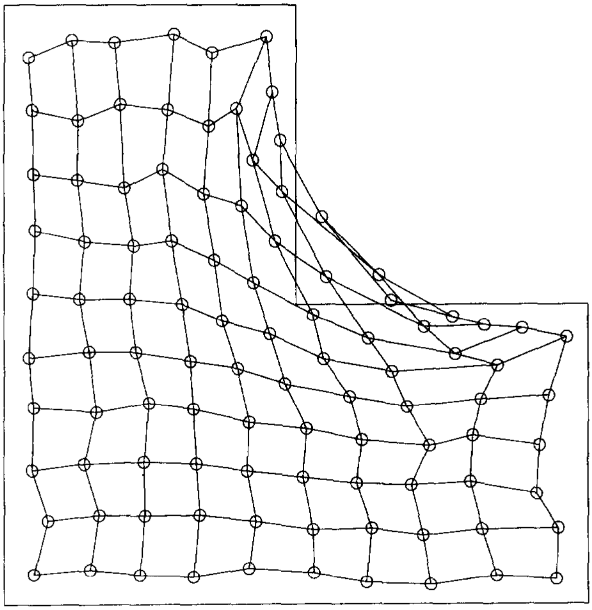

Fig. 6. Inherited 2D lattice-like neighbourhood graph of a Kohonen network after training with pattern vectors from an L-shaped region. The pattern space is restricted to an L-shaped region. Circles represent neural nodes; a line between circles represents a neighbour relation in the inherited neighbourhood graph. As a result of the neighbour training, a few prototype vectors remain outside the pattern space. The neighbourhood graph is a conventional 2D lattice. Note that the 'topology' of the 2D lattice and the pattern space are the same, but the limit configuration is usually not topology preserving. Thus topology preservation (or, rather, geometry preservation) is not granted for the Kohonen network even if the dimensionality of the pattern space and that of the neural lattice are the same. In other words, not topology but a more strict neighbourhood relationship is encoded into the inherited neighbourhood graph.

means a defect in the mapping and the larger the value D the larger the defect. D measures only the extent of the topological defect rather than the number of defective edges.

One cannot exploit fully the advantages of the Kohonen network unless the geometry of the pattern space fits the structure of the network. Thus in order to use the network effectively one must *a priori* know the geometry of the pattern space. We use the words *geometry* and *structure* instead of *topology* for the following reasons. Consider a network with a 2D lattice structure and assume that the inputs of the network are points of a concave, L-shaped region of the 2D space (see Fig. 6). Usually it is undesirable to have prototype vectors outside of the pattern space (see Fig. 6). In this case, however, a few prototype vectors are constrained out the pattern space in the limit. This means that the resulting mapping is not correct. [9] However, the 'topology' of the network (2D lattice,

[9] One may consider the mapping in Fig. 6 as correct. A possible solution is to restrict Definition 3.8 to prototype vectors that are inside the pattern space.

connected) is the same as the topology of the pattern space. Thus the geometry (including metrics) rather than the topology is captured by the network.

If the geometry of the pattern space is not known a priori then the use of Kohonen networks may be disadvantageous. Problems may arise even if the 'dimension' of the lattice structure of the Kohonen network matches the dimension of the pattern space. [10] This may result in several prototype vectors settled outside the pattern space [9]. If the lattice 'dimension' is lower than the dimension of the pattern space then the network 'folds' itself into the pattern space. In these cases the topographical mapping is not correct in the strong sense. This type of defect can be measured similarly to (7). Now the question is how to figure out the dimension or even the geometry of the pattern space a priori. The dimension of the external space that is mapped by the transducers onto the pattern space may be a good starting point. (Note that the resolution of both the adaptive vector quantization and the Kohonen algorithm depend only on the number of prototype vectors and on the distribution of the pattern vectors. Neither of these algorithms can achieve a fine resolution in very high dimensional pattern spaces.) But in general spaces with mixed dimensional parts are also possible and for such spaces no appropriate single dimension can be found [24]. In such cases the development of adaptive lateral connections may be advantageous.

3.3 Learning the geometry of the pattern space

Learning the geometry of the pattern space means to adapt the lateral connections in a way that the CBN-Graph corresponding to the settled network weights matches the SBN-Graph induced by the settled prototype vectors. Since the SBN-Graph is defined by the 'common winning region' of neurons this implies the use of the following rule:

$$\Delta q_{ij} = \begin{cases} \beta(S_i S_j - q_{ij}), & \text{if } \mathbf{x} \in C_{ij}; \\ 0, & \text{otherwise}. \end{cases} \tag{8}$$

This rule, first introduced by Martinetz, is called the 'competitive Hebbian law' for it can be interpreted as a competition between the lateral connections. The rule requires the determination of the best and the second best prototype vectors. The result is that q_{ij} converges to

$$q_{ij}^* = E\left[S(\mathbf{x}, \mathbf{w}_i^*) S(\mathbf{x}, \mathbf{w}_j^*) \mid \mathbf{x} \in C_{ij}\right], \tag{9}$$

which is the conditional expectation value of the product of similarities of the limit weights of neurons to the input vectors if $C_{ij} \neq \emptyset$ and otherwise $q_{ij}^* = 0$. The CBN-Graph associated with the limit configuration of the network will be the same as the SBN-Graph. This is a consequence of the following slightly generalized theorem of [23].

[10] The dimension of the pattern space may be defined as the minimum of the dimensions of vector spaces that the pattern space can be embedded into by topological mapping.

Theorem 3.2. *If* $S(\mathbf{x}, \mathbf{w}_i^*) > 0$ *for all* $\mathbf{x} \in \cup_k C_{ik}$ *then* $q_{ij}^* > 0$ *is equivalent to* $Prob(\mathbf{x} \in C_{ij}) > 0$ *where* q_{ij}^* *is defined by Eq.* (9). [11]

Proof. Since q_{ij}^* is the conditional expectation of a non-negative variable over C_{ij}, $Prob(\mathbf{x} \in C_{ij}) > 0$ follows from $q_{ij}^* > 0$. On the other hand, assume that $Prob(\mathbf{x} \in C_{ij}) > 0$ holds. If this does, indeed, hold then $q_{ij}^* > 0$ holds too since $C_{ij} \subset \cup_k C_{ik}$ and thus $S_i S_j$ is a non-negative variable over C_{ij}. □

Although the similarity based neighbourhood graph is an important concept, for a correct approximate representation of the *external geometry* one needs the matching of the GIN-Graph and the CBN-Graph. An example is presented in Fig. 3 that helps us to discriminate between the matching of external geometry and the similarity based geometry representation. The figure shows the CBN-Graph of two networks trained on different external space geometries. The similarity function was the normalized inner product function. In the case of the right hand side figure, objects of the external space were restricted to belonging to one of three disjoint regions,

$$R_i = \{(x, y) \in [0, 1]^2 \mid i/3 < y < (i+1)/3\}, \quad i = 0, 1, 2.$$

Attention is drawn to the fact that the regions were infinitesimally near to each other. The transduction mapping $\mathscr{P}$ preserved the regions, i.e. if $A \subset R_i$ and $B \subset R_j$, $j \neq i$ then $R(\mathscr{P}(A)) \cup R(\mathscr{P}(B)) = \emptyset$. This ensured that objects from different regions resulted in separable pattern vectors. If only the pattern space is considered and the nature of the external space is neglected then, depending on the similarity function and the horizontal resolution (i.e. the horizontal resolution of the transduction mapping and the number of neurons together), prototype vectors can be either 'distant' or 'close' to each other. In a correct SBN-Graph two prototype vectors may be connected even if there is no continuous path (still in the pattern space) between them. This question is further examined in [32]. In our case the transduction mapping $\mathscr{P}$ and the similarity function S realize the connection between the external space and the CBN-Graph.

4. Conclusion

In this article we have reviewed concepts and theorems relate to the field of geometry representation by neural networks. The central point was the use of finite graphs: we have introduced various graphs embedded into the pattern space. The concept of a correct geometry representation was defined in terms of the coincidence of the edges of graphs. The main feature of our approach is that the considered geometry representations are based on the geometry of the external space instead of that of the pattern space.

[11] Note that since $C_{ij} \subset \mathrm{supp}(F)$, $Prob(\mathbf{x} \in C_{ij}) = F(C_{ij}) > 0$ is equivalent to $C_{ij} \neq \emptyset$.

We have examined winner-takes-all networks equipped with lateral connections. The lateral connections were developed according to the Hebbian law based on the input intensities of neurons. The approximate geometry representations were associated with the configurations of the network. It was shown that appropriately selected neural activation functions may lead to the formation of a correct geometry representation. The resolution of our geometry representation does not depend on the number of neurons but rather on the local nature of the training objects.

Geometry representation is important since (i) Kohonen-type neighbour training may be introduced with the help of these connections [34,35], (ii) the connection structure may be used for path planning [21,4,25,11] as well as for learning motion control [35], and (iii) for feature extraction that utilizes geometry information [2,22].

References

[1] S. Amari, Theory of adaptive pattern classifiers, *IEEE Trans. Elect. Comput.* 16 (1967) 299–307.

[2] H. Blum, Biological shape and visual science (part I), J. *Theoretical Biology* 38 (1973) 205–287.

[3] J. Bruske and G. Sommer, Dynamic cell structure learns perfectly preserving map, *Neural Computation* (submitted).

[4] C.I. Connolly and R.A. Grupen, On the application of harmonic functions to robotics, *J. Robotic Systems* 10(7) (1993) 931–946.

[5] M. Cottrel and J.C. Fort, A stochastic model of retinotopy: a self-organizing process, *Biological Cybernetics* 53 (1986) 405–411.

[6] M. Cottrel, J.C. Fort and G. Pages, Two or three things that we know about the Kohonen algorithm, in: *Proc. ESANN'94*, Brussels (1994) 271–276.

[7] R. Der, M. Hermann and Th. Villmann, Time behavior of topological ordering in self-organized feature mapping, *Biological Cybernetics* (in press).

[8] B. Fritzke, Growing cell structures – a self organizing network for unsupervised and supervised training, Technical Report, TR-93-026, ICSI Berkeley, 1993.

[9] B. Fritzke, Let it grow – self organizing feature maps with problem dependent cell structure, in: *Proc. ICANN'91* (Elsevier, Amsterdam, 1991) 403–408.

[10] K. Fukushima, Character recognition with neural networks, *Neural Computing* 4 (1992) 221–233.

[11] R. Glasius, A. Komoda and S. Gielen, Neural network dynamics for path planning and obstacle avoidance, *Neural Networks* (1994).

[12] R.M. Gray, Vector quantization, *IEEE ASSP Mag.* 2(1) (1984) 4–29.

[13] S. Grossberg, Adaptive pattern classification and universal recording: I. parallel development and coding of neural feature detectors, *Biological Cybernetics* 23 (1976) 121–134.

[14] S. Grossberg, Competitive learning: from interactive activation to adaptive resonance, *Cognitive Science* 11 (1987) 23–63.

[15] R. Hecht-Nielsen, *Neurocomputing* (Addison-Wesley, Reading, MA, 1991).

[16] Y.K. Hwang and N. Ahuja, Gross motion planning – a survey, *ACM Computing Surveys* 24(3) (1992) 219–291.

[17] L.P. Kaelbling, *Learning in Embedded Systems* (MIT Press, Cambridge, MA, 1991).

[18] T. Kohonen, Analysis of a simple self-organizing process, *Biological Cybernetics* 44 (1982) 135–140.

[19] T. Kohonen, Self-organized formation of topographically correct feature maps, *Biological Cybernetics* 43 (1982) 59–69.

[20] P.R. Krischnaiah and L.N. Kanal (eds.), *Classification, Pattern Recognition, and Reduction of Dimensionality, Handbook of Statistics*, Vol. 2 (North-Holland, Amsterdam, 1982).

[21] G. Lei, A neural model with fluid properties for solving labyrinthian puzzle, *Biological Cybernetics* 64(1) (1990) 61–67.
[22] Zs. Marczell, Zs. Kalmár, Cs. Szepesvári, and A. Lőrincz, The generalized skeleton. *Neural Processing Letters* (submitted).
[23] T. Martinetz, Competitive Hebbian learning rule forms perfectly topology preserving maps in: *Proc. ICANN'93* (Springer-Verlag, London, Amsterdam, The Netherlands, 1993) 427–434.
[24] T. Martinetz and K. Schulten, A 'neural-gas' network learns topologies in: *Proc. ICANN* (Elsevier, Amsterdam, 1991) 397–402.
[25] P. Morasso, V. Sanguineti and T. Tsuji, Neural network architecture for robot planning, in: *Proc. ICANN'93* (Springer-Verlag, London, Amsterdam, The Netherlands, 1993) 256–261.
[26] K. Obermayer, H. Ritter, and K. Schulten, Large-scale simulation of self-organizing neural network: formation of a somatopic map, in: *Parallel Processing in Neural Systems and Computers*, (North-Holland, Amsterdam, 1990) 71–74.
[27] K. Obermayer, H. Ritter and K. Schulten, A neural network model for the formation of topographic maps in the CNS: Development of receptive fields, in: *IJCNN-90, Conf. Proc. II*, San Diego (1990) 423–429.
[28] H. Ritter and K. Schulten, Convergence properties of Kohonen's topology conserving maps: fluctuations, stability and dimension selection, *Biological Cybernetics* 60 (1988) 59–71.
[29] F. Rosenblatt, *Principle of Neurodynamics*. (Spartan, New York, 1962).
[30] J. Rubner and K. Schulten, A self-organizing network for complete feature extraction, *Biological Cybernetics* 62 (1990) 193–199.
[31] D.E. Rumelhart and D. Zipser, Feature discovery by competitive learning, *Cognitive Science*, 9 (1985) 75–112.
[32] Cs. Szepesvári, The role of lateral connections in competitive neural networks: collective learning and representation of topology, Master's thesis, University of Szeged, Bolyai Institute of Mathematics, Szeged 6720, Aradi vrt. 1, Hungary, e-mail: szepes@math.u-szeged.hu, 1993 (in Hungarian; extended abstract available in English).
[33] Cs. Szepesvári, L. Balázs and A. Lőrincz, Topology learning solved by extended objects: a neural network model in: *Proc. ICANN'93* (Springer-Verlag, London, Amsterdam, The Netherlands, Sep. 1993) 678.
[34] Cs. Szepesvári, L. Balázs and A. Lőrincz, Topology learning solved by extended objects: a neural network model, *Neural Computation* 6(3) (1994) 441–458.
[35] Cs. Szepesvári, T. Fomin and A. Lőrincz, Self-organizing neurocontrol, in: *Proc. ICANN'94* (Springer-Verlag, Sorrento, Italy, 1994) 1261–1264.
[36] Cs. Szepesvári and A. Lőrincz, Self-organized learning of 3 dimensions, in: *Proc. IEEE WCCI ICNN'94* (IEEE, Orlando, FL, 1994) 671–674.
[37] V.V. Tolat, An analysis of Kohonen's self-organizing maps using a system of energy functions, *Biological Cybernetics* 64 (1990) 155–164.
[38] T. Villmann, R. Der and T. Martinetz, A novel approach to measure the topology preservation of feature maps in: *Proc. ICANN'94* (Springer Verlag, Heidelberg, Sorrento, Italy, May 1994) 298–301.
[39] Ch. von der Malsburg, Self-organization of orientation sensitive cells in the striate cortex, *Kybernetik* 14 (1973) 85–100.
[40] T. Wasan, *Stochastic Approximation* (Cambridge University Press, London, 1969).

Csaba Szepesvári graduated from the Attila József University of Szeged in 1993. He won the Main Prize of the National Conference of Student Researchers for his Master's thesis on 'Role of Local Connections in Competitive Neural Networks: Collective Learning and Representation of Geometry.' Currently he is a Ph.D. student at the Attila József University of Szeged working on self-organizing artificial neural networks and autonomous systems. He is a member of the János Bolyai Mathematical Society.

András Lőrincz graduated from the Loránd Eötvös University, Budapest in 1975. He wrote his Master's thesis on the field of theoretical particle physics, titled 'Statistical Many Particle Creation.' He did experimental Ph.D. work on 'Thermocurrent Dosimetry with Al_2O_3 Single Crystals' in 1978, and wrote his Candidate thesis in both experimental and theoretical studies of 'Ultrafast Energy Redistribution Processes of Molecules in Supersonic Expansions' in 1986. Currently he is working on adaptive autonomous agents and their applications, to, e.g., adaptive optics, acoustic problems and to the control of quantum systems. He is a member of the Loránd Eötvös Society of Physics.