Neural Comput & Applic (1999)8:163–176
© 1999 Springer-Verlag London Limited

Neural Computing & Applications

Winner-Take-All Network Utilising Pseudoinverse Reconstruction Subnets Demonstrates Robustness on the Handprinted Character Recognition Problem

J. Körmendy-Rácz[1], S. Szabó[2], J. Lörincz[2], G. Antal[2], G. Kovács[3] and A. Lörincz[1,4]

[1]Department of Chemical Physics of the Institute of Isotopes, Hungarian Academy of Sciences, Budapest, Hungary; [2]Department of Mathematics, Eötvös Loránd University Budapest, Hungary; [3]Department of Physiology, Albert Szent-Györgyi Medical University, Szeged, Hungary; [4]Department of Adaptive Systems, Attila József University, Szeged, Hungary

Wittmeyer's pseudoinverse iterative algorithm is formulated as a dynamic connectionist Data Compression and Reconstruction (DCR) network, and subnets of this type are supplemented by the winner-take-all paradigm. The winner is selected upon the goodness-of-fit of the input reconstruction. The network can be characterised as a competitive-cooperative-competitive architecture by virtue of the contrast enhancing properties of the pseudoinverse subnets. The network is capable of fast learning. The adopted learning method gives rise to increased sampling in the vicinity of dubious boundary regions that resembles the phenomenon of categorical perception. The generalising abilities of the scheme allow one to utilise single bit connection strengths. The network is robust against input noise and contrast levels, shows little sensitivity to imprecise connection strengths, and is promising for mixed VLSI implementation with on-chip learning properties. The features of the DCR network are demonstrated on the NIST database of handprinted characters.

Keywords: Categorical perception; Competitive-cooperative behaviour; Hand-printed character recognition; On-chip learning; Pseudoinverse method; Reconstruction

Correspondence and offprint requests to: J. Kormendy-Rácz, Department of Chemical Physics of the Institute of Isotopes, Hungarian Academy of Sciences, Budapest, P.O. Box 77, Hungary H-1525.

1. Introduction

Character recognition is closely related to decision making in that they select one or more options out of two or many possibilities. Decisions are based on certain criteria, e.g. by setting decision surfaces with multilayer perceptrons [1–3], or by utilising the winner-take-all paradigm that selects the winning unit based on the neural activities [4–6]. Here, a winner-take-all paradigm is proposed, which makes decisions on the input reconstruction abilities of subnets. The subnets can be considered as coarse coded memories of the input space or, equivalently, as population codes of the different clusters, with the number of coding units being smaller than the dimension of the input. This means that, for every subnet, the problem is equivalent to data compression and reconstruction. Different approaches use different paradigms, which try to deal with data compression and reconstruction. One part of this enormous field is the family of parallel techniques using simple, possibly adaptive, computational elements, i.e. the field of Artificial Neural Networks (ANN). A brief listing of these might include the auto-associative multilayer perceptron method [7], the Adaptive Resonance Theory (ART) scheme [8] and Principal Component Analysis (PCA) [9,10], each being formulated in ANN terms in different ways [11–13]. Both the multilayer perceptron method and PCA feature slow learning (tuning) properties [14]. The ART network is a fast learning architecture using external behavioural feedback via the vigilance parameter, and it quickly memorises or replaces memories if the input and the reconstructed input do not match properly. Flexibility may be increased by adding external feedback to tune the vigilance parameter [8]. The primary goal of ART is not, however, the reconstruction, but adaptive clustering: ART matches the input by means of the internal long-term memory system, and the matching procedure gives rise to the selection of the appropriate winning unit.

In the present paper, we investigate Wittmeyer's iteractive scheme [15] for solving the input reconstruction problem [16] and combine it with a winner-take-all method, where the winner is selected upon a least squares Goodness-Of-Fit (GOF) of the input reconstruction. At this point, the method resembles the self-organising and reconstructing local PCA architecture of Joutsensalo et al. [17]. The main difference between the local PCA method and the present architecture is both in the purpose and in the training method. Here, a categorising architecture is suggested that, accordingly, exhibits 'categorical perception' sometimes at the expense of increased reconstruction error. Also, the training procedure utilised here is adjusted to the needs of mixed VLSI technologies exhibiting on-chip learning [18–20]. The training method features the attractive properties of ART, i.e. the architecture utilises input vectors directly.

It should be noted that algorithms which are related to Wittmeyer's method by extending it with a lossy term show robustness [21], and have been suggested as information transfer maximising and sparse representation promoting methods [22]. Another connection to the literature is the recognition architecture of Stewart-Bartlett and Sejnowski [23]. It is mentioned here that there is some dichotomy in the literature on the use of the words 'recognition' and 'categorisation'; recognition means 'I have seen this before', whereas in the case of categorisation, the actual input may not have been encountered before. Even so, the categorisation of characters is known as character recognition. We shall try to make the distinction by using the abbreviation 'CR' for the character categorisation problem. The method of Stewart-Bartlett and Sejnowski does indeed, mimic recognition, and makes use of memory vectors developed by PCA and also by Independent Component Analysis (ICA). The test inputs are reconstructed by means of the PCA/ICA memories and the element of the training set is selected that has the smallest Euclidean distance to the reconstructed image. The method we present here can be viewed as the categorisation counterpart of this recognition method with the special intent that the memory vectors should suit VLSI technology. Thus, the memory matrix is not inverted directly and, instead, Wittmeyer's iterative algorithm is transformed into a dynamic architecture to use iconic memories that can make use of single bit connections. The paper is organised as follows. In the next section, the Data Compression and Reconstruction (DCR) architecture is reviewed. In Section 3, the DCR subarchitectures are extended by a Winner-Takes-All (WTA) layer. Computer illustrations are presented in Section 4, and the main features of the architecture are discussed in Section 5. Conclusions are drawn in the last section.

2. The DCR Scheme

Consider that one has an input-output system. The task is that the output should be used for data reconstruction, and the only tool available for the reconstruction is the information contained by the input-output system itself. To make the example clear, consider the problem of video-email transmission or teleconferencing [24]: there are two or more nodes on the network that can be equipped with the same input-output systems. Assume further that the input-output system is made up of 'samples' or 'memories'. The input on one side could be the face of the speaker, and the input-output system can be thought of as a set of faces of different speakers from different views and with different gestures. What kind of output should be transmitted along the network to the other side(s) that promotes reconstruction of the actual face, actual view and actual gesture at the other node(s)?

Assume, that our input-output system has a finite number of memories, $\mathbf{q}_i$ with ($i = \mathbf{1}, \ldots, n$). Each memory has N elements, i.e. memory $\mathbf{q}_i^T = (q_{il}, \ldots, q_{iN})$, where superscript T denotes the transposed form of the vector. Let us denote the matrix formed by the q_{ij} elements by Q with $Q(i, j) = q_{ij}$. For any input vector $\mathbf{x}$ one can form the direct internal representation or direct internal activities by means of the memories:

$$a_{d_i} = (\mathbf{q}_i, \mathbf{x}) \tag{1}$$

where (.,.) denotes the dot product of the arguments. Components a_{d_i} with $i = 1, \ldots, n$, can be collected in the vector $\mathbf{a_d} \in \mathbf{R}^n$. In ANN terminology, one has n neurons each having N connections to the N dimensional (sensory) input. Another way to define the same system in the ANN literature is to say that we have n neurons, each representing an N dimensional memory. From now on, neurons will sometimes be called processing units. Connection vectors may also be called input filters or memories, and Q will be known as the memory matrix.

The problem is formulated as follows: every partner taking part in the teleconferencing has the same memory matrix. Activities, not necessarily the direct activities, can be entered into the network. What then is the 'best' set of internal activities that should be entered into the network to promote reconstruction?

The particular features of the reconstruction procedure are important. After the information is entered into the network the recipient of that information reconstructs the initial input by adding up the different memories based on the activities sent. Either the sender or the recipient can ask whether the reconstructed input gives rise to the same internal activity vector or not. From this point of view, the problem is not symmetrical, and we shall proceed by assuming that the question is asked by the sender.

The following procedure then takes place: (1) the sender reconstructs the input since he/she is also equipped with the memory matrix; (2) the sender tries the reconstructed input to see if the same internal activity arises; and if not (3) he/she modifies the internal activity until it does. This procedure ensures that when sending the information the receiving side, will have exactly the same reconstructed image as the sender. In fact, the recipient has the option of measuring the quality of reconstruction since upon receiving the internal representation he/she can try to reconstruct the input. If the same procedure is executed on the receiving side, then the procedure stops at the very first iteration by construction. If it does not stop, then it indicates that either the quality of the transmission is not perfect, or that the memory matrices of the sender and the receiver are mismatched. It may be worth noting here that we set out to design a procedure with the following properties. With perfect transmission and identical memory matrices on the two sides, the procedure stops at the very first iteration on the receiver side; in other words, we designed a projection method.

With this protocol in mind, we can try to find the appropriate dynamic equations that give rise to the required property, *viz*, that of finding a relaxation equation that gives the internal representation with the immediate reconstruction (projection) property. In other words, the internal representation that we develop should give rise to zero correction when it is used for reconstruction. A set of suitable equations can be designed as follows:

$$\mathbf{a_d} = Q^T\mathbf{x}$$

$$\mathbf{y} = Q\mathbf{a}$$

$$\mathbf{a_e} = Q^T\mathbf{y}$$

$$\mathbf{a} = \mathbf{a_d} + \mathbf{w}$$

$$\dot{\mathbf{w}} = \lambda(\mathbf{a_d} - \mathbf{a_e}) \tag{2}$$

The input forms the 'direct' internal representation, $\mathbf{a_d}$. That vector is used to compute the reconstructed image, $\mathbf{y}$. Vector $\mathbf{y}$ is then tried on the memory system and it gives rise to an 'experienced' internal representation, $\mathbf{a_e}$. The difference between the two vectors, i.e. the difference between vectors $\mathbf{a_d}$ and $\mathbf{a_e}$, is used as a correction to improve internal representation $\mathbf{a}$. In Eq. (2) the N-dimensional vectors ($\mathbf{x}$ and $\mathbf{y}$) and the n-dimensional vectors ($\mathbf{a_d}$, $\mathbf{a_e}$ and $\mathbf{a}$) are all denoted by boldface characters, but that should not cause any confusion. The structure of Eq. (2) closely resembles the wording of our goals. The procedure stops if the difference $\mathbf{a_d} - \mathbf{a_e}$ disappears. This stopping condition ensures that the reconstructed vector $\mathbf{y}$ gives rise to an internal representation that needs no correction, since on stopping the following equation holds:

$$Q^T\mathbf{y} = \mathbf{a_e} = \mathbf{a_d} = Q^T\mathbf{x} \tag{3}$$

Thus,

$$Q^T\mathbf{x} = Q^TQ\mathbf{a} \tag{4}$$

Equations (3) and (4) show that the dynamical equation gave rise to internal representation $\mathbf{a}$, which is equivalent to a solution of the overdetermined equation.

$$Q\mathbf{a} = \mathbf{x} \tag{5}$$

and according to our goals we have a projection method, since $\mathbf{y}$ may be expressed as

$$\mathbf{y} = Q(Q^TQ)^{-1}\, Q^T\mathbf{x} \tag{6}$$

and the combination of the Q matrices $P = Q(Q^TQ)^{-1}\, Q^T$ features the property $P^2 = P$, and thus forms a projector matrix. In short, we have designed a relaxing dynamical equation that utilises three identical copies of the same memory matrix, and can invert the overdetermined equation $Q\mathbf{a} = \mathbf{x}$ without direct access to the pseudoinverse of matrix Q. The architecture and its special features are detailed below.

2.1. Convergence of the Data Compression and Reconstruction (DCR) Architecture. Introducing discrete time iterations with Δt time steps instead of the continuous time relaxation equations (Eq. 2) one can write that

$$\mathbf{a}^{(i+1)} = \kappa\mathbf{a_d} + (\mathrm{I} - \kappa \mathrm{Q}^T\mathrm{Q})\mathbf{a}^{(i)} \tag{7}$$

where κ is the gain factor, which is equivalent to λ times Δt of (Eq. 2) with Δt denoting the discretisation step size and with $\mathbf{a}^{(0)} = \mathbf{a_d}$ giving rise to the observation that the data compression and reconstruction scheme is equivalent to Wittmeyer's iterative algorithm for solving matrix Eq. (15). If the additive correction of each iteration is supplemented by a negative term which is proportional to a function of the actual value of **a**, then one has the starting equation of the information maximizing sparse representation promoting method [22].

One can show that the algorithm minimises the mean square error of the expression

$$J = |\mathbf{x} - \mathbf{Qa}|^2 \tag{8}$$

by changing **a** along the gradient of J.

The connectionist architecture that implements the DCR algorithm in the form of a local, recurrent artificial neural network is depicted in Fig 1. The architecture will be called the DCR scheme. The full network is made up of three replicas of the original memory matrix Q. The first replica, the transposed form of memory matrix Q receives the input vector **x** and gives rise to the internal vector $\mathbf{a_d}$; the second utilises matrix Q and computes the reconstructed input **y**; the third replica deals with the reconstructed image and computes the experienced internal representation $\mathbf{a_e}$. The rest is the differencing between $\mathbf{a_d}$ and $\mathbf{a_e}$, integration with gain factor λ, and then summation to form the corrected internal representation **a**.

It should be noted that the correction vector **w** alone can minimise the mean square error J. It can still be useful to keep the direct term as the initial guess for **a** ($\mathbf{a} = \mathbf{a_d} + \mathbf{w}$), since if matrix Q is itself a projection matrix (or if it is close to a projection matrix), then the architecture becomes feedforward because the correction term, $\mathbf{a_d} - \mathbf{a_e}$, becomes zero upon the first iteration step. This is the case, for example, if memory matrix Q is being tuned by the PCA method. Provided that matrix Q is of full rank, it can be shown that Wittmeyer's algorithm converges in an exponential fashion. The convergence time is determined by the gain factor λ and the inverse of the smallest eigenvalue of matrix Q^TQ.

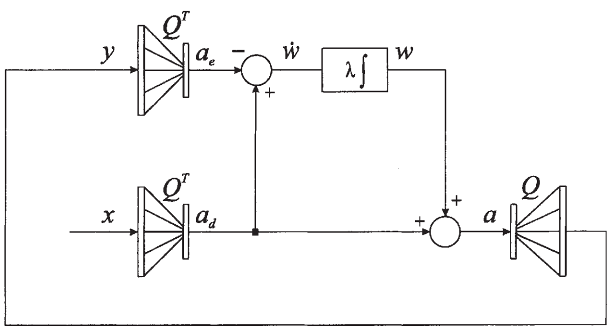

Fig. 1. Architecture of data compression and reconstruction network. The network is made up of three replicas of the original memory matrix Q: the first replica receives the input vector **x** and gives rise to the internal vector $\mathbf{a_d}$; the second utilises the same matrix and computes the reconstructed input **y**; the third deals with the reconstructed input and computes the experienced internal representation $\mathbf{a_e}$. The internal representation undergoes (1) differencing between $\mathbf{a_d}$ and $\mathbf{a_e}$, (2) integration with gain factor λ, and (3) summation to form the corrected internal representation **a**.

2.2. Dynamic Contrast Enhancing Properties of the DCR Scheme. Assume that all of the memories of the DCR architecture are normalised to one. An interesting property of the DCR scheme follows if one considers the case when the actual input **x** is equal to one of the memories of matrix Q. In this case – and provided that matrix Q is of full rank – the relaxed internal representation will have one single active neuron: the one that corresponds to the input; in contrast, all the other neurons will assume zero activities, since this single neuron is able to reconstruct the input by itself. Here the DCR scheme works as a WTA algorithm, since many neurons can be activated at the initial step (depending on the overlap between the input and the memories), and only one neuron stays active. It is also true for the normalised memories that the neuron with the largest initial activity will be 'selected' as the winner [16].

The DCR scheme that utilises overlapping memories can thus be viewed as a *linear* network that promotes contrast enhancement sometimes up to the extreme of WTA behaviour at the level of the internal representation via indirect inhibitory action between neurons (expressed by the term $-\mathbf{a_e}$), and the inhibitory action is mediated by the reconstruction procedure [25].

3. Extension using the Winner-Take-All Procedure

It is straightforward to extend the DCR scheme to a categorising architecture by designing DCR subnets for each category and utilising a WTA layer for decision making. The WTA decision should be based on the goodness of reconstruction. The DCR subnet that produces the smallest reconstruction error can win and provide the output of the categorising scheme. One can say that reconstruction networks provide a natural Goodness-Of-Fit (GOF) measure, and that measure serves as the tool of decision making. In the DCR+WTA network, each separate cluster (or winning domain) is represented by several memories. A traditional WTA procedure would compute the distances between the input and the memories, determine which memory is the closest to the input (this is the winning memory), and determine which category that memory belongs to (this is the winning category). The WTA procedure that works together with the DCR scheme is, however, different. In this case, the memories belonging to a category compete together with memories that belong to other categories. This property gives rise to a different arrangement for the memory vectors when compared with the original WTA procedure, since WTA decision surfaces correspond to the surfaces of Voronoi polyhedra defined by the memory vectors, whereas the DCR method is a subspace method.

4. Computer Illustrations

The set of samples we have used for the computational studies is the subset of the NIST handprinted digit database. This subset is publicly available on the world wide web (ftp://sequoyah.ncsl.nist.gov/pub/databases/data/fl3.tar.Z). The digits comprise 32 by 32 pixels. The number of handprinted digits is different for the different digits. The respective numbers are given in the caption of Fig. 2. The figure depicts the averages of the samples for each category. These are grey scale images, and thus such grey scale images should be well reconstructed by such memories. In other words, the DCR scheme can use single bits as connection strengths, and can still deal with grey scale inputs, at least for the CR problem.

Figure 3 depicts the case when 10 subnets, each having five memories, were utilised parallely to reconstruct the actual input (digit 3) shown at the top of the figure. Each subnet creates reconstruction vectors (shown in grey scale in the figure) that best match the input. The reconstructed inputs look like the actual category, since these types of input can be reconstructed based on the memory content. That is, the reconstructed input of the first subnet looks like a 'zero'; that of the second subnet looks like a 'one'; and so on. Also, the different memories give rise to odd combinations, and thus in most cases, to hazy reconstructed images, since the input

Fig. 2. Averaged digits. The figure depicts the averaged digits from 0 to 9 for 406, 404, 378, 384, 331, 209, 369, 347, 304 and 339 handprinted samples, respectively. The source of the samples is ftp://sequoyah.ncsl.nist.gov/pub/databases/data/fl3.tar.Z.

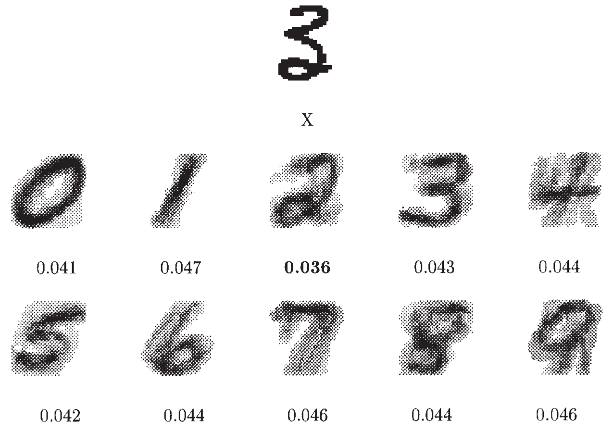

Fig. 3. Reconstruction example. The input to the 10 subnets is shown at the top. The best reconstructed images together with the respective Goodness-Of-Fit (GOF) measures are shown. The winner, erroneously, is the subnet belonging to digit 2.

does not match the memories. The lower loop of the actual input '3' is not typical, and the best subnet that has the lowest GOF measure in this particular example happens to be the subnet representing digit 2. By replacing one of the memories of the subnet of digit 3 by the actual input, then upon the presentation of the same input this subnet will have zero reconstruction error (and will thus win). Thus, the DCR scheme allows the following 'tuning' protocol: if an input is miscategorised, then that input can be placed as a new memory vector into the appropriate subnet, either by increasing the number of memories or by simultaneously dropping one of the memories of the same subnet.

One feature of this 'tuning' procedure is that the stored memories will become more ambiguous than the average memories, thus emphasising the cooperative capabilities of the DCR scheme.

Two tuning procedures are depicted in Fig. 4. Circles without labels represent the memory vectors of the correct category that should have won the

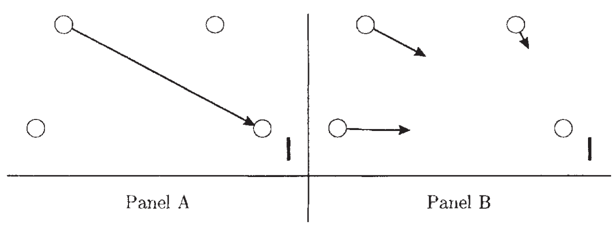

Fig. 4. Tuning procedures. Circles without labels represent the memories. Circle *I* represents the input. Direct tuning (panel A). In the case of improper categorisation, the subnet belonging to the actual input is 'tuned' by placing the input into the subnet (the input is memorised), whereas one of the memories is dropped (forgotten). Smooth tuning (panel B). In this case, the memories are tuned in a smooth fashion that sacrifices the 'iconic' feature of the stored memory vectors.

competition. Circle *I* represents the actual input that has been miscategorised.

Training Rule 1. The arrow on panel A of Fig. 4 originates from a memory vector that has been selected randomly from the subnet and will be dropped, and thus the miscategorised element *I* can be inserted into the memory. The arrow in panel A of Fig. 4 represents this 'drop and insert tuning' protocol. In the computation experiments on categorical perception, this 'tuning' procedure will be utilised.

Training Rule 2. A less strict version of Training Rule 1 replaces the memory vector which is the closest (in Euclidean distance) to the actual input with this input. This 'tuning' procedure will be used in all the other computer experiments. In both cases, we use inputs to represent memories and thus the memories are 'iconic'.

Further smoothing of the tuning can be developed if the memory vectors are tuned towards the input vector and the actual and miscategorised input is not saved. This procedure is depicted in panel B of Fig 4. Such smooth tuning procedures, however, sacrifice the iconic feature of the algorithm.

Figure 5 illustrates the information filling properties of the architecture utilising 10 memories in each subnet. The memories used in this test were selected according to the following procedure. First, the database was divided into two parts. The first 100 samples of each category were selected as the training set and the rest were used as test samples. Out of the respective 100 samples of each digit, the first memory was selected for each subnet as the first memory of that subnet. Then, the other samples were input to the network in accordance with their respective natural order, one from each set of 100, and the winning subnet was selected. If the winning subnet was the correct subnet, then the procedure continued. In the event of miscategorisation, then

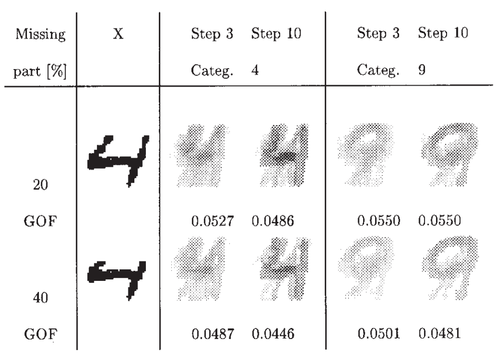

Fig. 5. Information filling based on memories. Inputs with missing information can be corrected by the subnets. The Goodness-Of-Fit (GOF) measure indicates that a relatively high level of missing information does not prevent correct categorisation.

the actual sample was added to the memory set of the subnet. The procedure was continued until all of the samples of the training set had been input once. All the memory sets were limited to 10, and if a new miscategorisation occurred after that number had been reached then no memory was added. After this part of the training, in two other runs the samples of the training set were input again in their natural order and Training Rule 2 was applied. A subset of the memories is shown in Fig. 6. The subnet representing digit 1 grew to an eight-member subnet and did not require any further memories.

Samples representing digit 4 but deleting 20% and 40%, respectively, were input to the DCR subnets of digit 4 and digit 9. The reconstructed input is shown for both cases together with the GOF measures along the relaxation of the subnets at the third and the tenth iterations, respectively. This information filling property is a simple consequence of the fact that the architecture is a subspace selecting architecture.

Another advantage is that the DCR net relaxes the contrast problem of the inputs, since different contrasts belong to the same subspace. Also, a dark character in a light background, or the same character but light within a dark background, belongs to the same subspace. These advantages also hold for PCA algorithms. An alternative to the present scheme is to develop the PCA components for each category separately, and to add a decision making WTA layer on top of the PCA networks and to determine the winner on the basis of the reconstruction error: the subnet having the smallest reconstruction error wins.

Figure 7 illustrates the background resistive properties of the architecture. Non-zero background, in general, moves the input out of the subspace defined by the memory vectors of zero-background, and *vice versa*. Uniform backgrounds (0.2 and 0.3) were added to each pixel of pixel values either 0 or 1. It can be seen in the figure that subnets 'try' to reconstruct the background. The larger the number of memories, the better this reconstruction will be, and the winning subnet can become independent from the family of memories the subnet should represent. This is also the case if the background level is increased. For example, the winning subnet is the subnet of digit 0 for background level 0.3. The solution to the background problem is to add a uniform background memory to each subnet, since this background is again made of single bit connection strengths. The memory number problem, however, is more complex, and we shall return to it in Section 5.

The DCR scheme exhibits robustness against

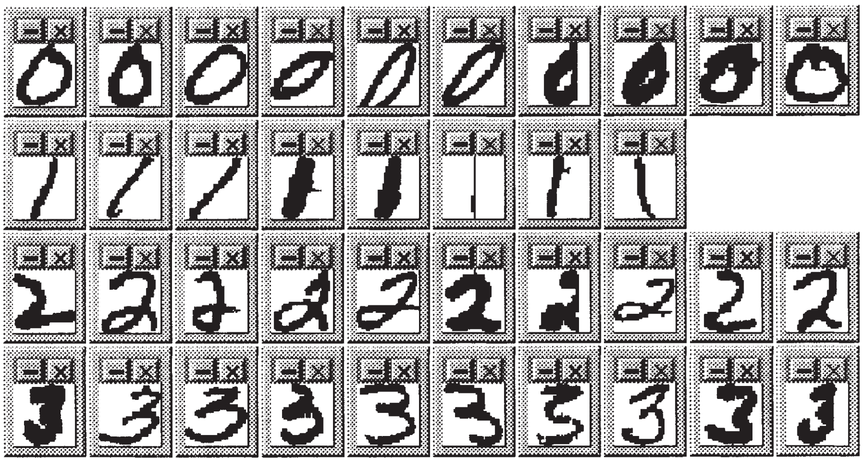

Fig. 6. Subset of the memories belonging to four subnets used in the tests.

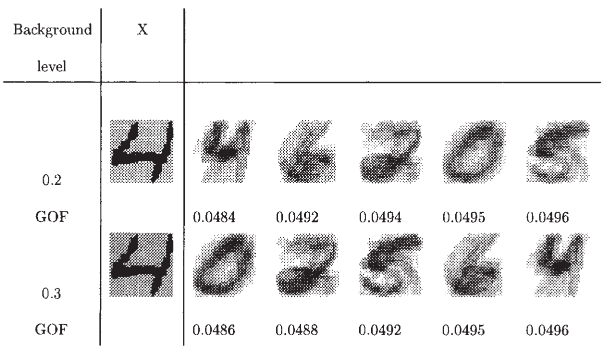

Fig. 7. Background resistivity. The subnets show relatively high resistivity to background. For a 20% background level the GOF measure still categorises properly.

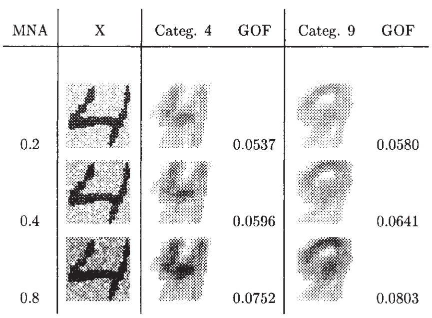

Fig. 8. Noise resistivity. To each pixel random noise between 0.0 and a Maximal Noise Amplitude (MNA) was added. The figure depicts the inputs and the reconstructed inputs for subnets corresponding to digits 4 and 9 together with the GOF values for MNA = 0.2, 0.4, and 0.8, respectively.

noise – as is demonstrated in Fig. 8. The figure was constructed by means of subnets utilising 10 memories for each of the subnets of digit 4 and digit 9. Noise was added to a randomly selected input. The noise content of each pixel was chosen as a random number between zero and the Maximal Noise Amplitude (MNA). The figure depicts the cases when the MNA value equals 0.2, 0.4 and 0.8. The signal content of each pixel is either 0 or 1, depending on the sample. The reconstructed inputs together with their GOF measures are shown. The difference between reconstruction errors is about 10% for MNA = 0.2 and that lowers to *circa* 7% for MNA = 0.8, still allowing correct categorisation.

Table 1 presents the categorisation error as a function of the MNA value. In this test, the trained 10 memory sets were used. To each test sample noise was added with MNA values 0, 0.2, 0.4 and 0.8. The results on categorisation performance are shown. It can be seen from the table that the error rate shows robust insensitivity up to MNA = 0.2 and a degradation beyond that value.

In one of the experiments the 10 member sets were compared with sets having three members and five members. The categorisation performances (as percentages) are shown in Table 2 for these three memory sets. The number of errors for each category is also listed. The percentage of correct categorisations grows steadily with memory number, how-

Table 1. Performance vs. noise. To each input pixel noise was added between 0.0 and the Maximal Noise Amplitude (MNA) value. Percentage of correct categorisation is given for MNA = 0.2, 0.4, and 0.8, respectively.

MNA	Correct categorisation (%)
0	87.1
0.2	87.4
0.4	85.6
0.8	73.4

Table 2. Percentage of correct categorisation and number of miscategorised elements vs. memory number. Top row: number of memories in the subnets. Bottom row: percentage of correct categorisation. Left column: numbers the subsets according to the digits they represent. Other numbers indicate the number of miscategorised elements from the test set.

	3	5	10
0	4	11	4
1	5	7	0
2	33	20	18
3	11	19	12
4	29	9	16
5	56	32	11
6	3	13	9
7	20	21	5
8	38	49	34
9	34	18	20
%	75.7	80.1	87.1

ever, the sample set is too small to enable us to examine higher memory numbers. The table demonstrates the competition between categories: miscategorisation is a somewhat arbitrary function of memory number.

One important question concerns the precision of the bit values of the stored memories. The issue is how robust the scheme is for mixed VLSI design where the memories are stored as bits, whereas the computational units are imprecise, and cannot use the precision of that bit or, otherwise, for a fully analog circuit that bit itself may be imprecise. Table 3 depicts the error rate of the network as a function of the imprecision (as a percentage) of the bit values. The imprecision was modelled by noise added to each connection value. The noise ranged between 0.0 and the MNA value. Performance is given as a function of the MNA value. The result is also shown for the case when, instead of this dynamic architecture, the pseudoinverse matrix itself was utilised and the connection strengths of the pseudoinverse matrix were spoilt in a similar fashion. The comparison was made as follows. In the case of the dynamic network, the noise content of each pixel was computed in terms of the maximal pixel value (1.0). The average pixel value was also computed and the average noise content was estimated. The pseudoinverse was spoilt by adding noise to each connection strength again in a homogeneous fashion, and determining the noise content according to the average connection strength. The pseudoinverse matrix showed high sensitivity to noise under these settings (for 5% MNA value performance decreased to 65.5% from 87.1%), whereas the DCR architecture showed high noise resistivity.

Table 3. Performance with spoilt connection strengths for the dynamic pseudoinverse network. Connection strengths were spoilt by adding noisy contributions to each value. The noisy contributions were randomly chosen between zero and the maximal random amplitude. The noise content was uniformly distributed over all connections and the noise content was normalised to the connection strengths (see text).

MNA	Correct categorisation with DCR (%)
0.00	87.1
0.10	84.3
0.20	81.5

The cooperative-competitive feature of the architecture with cooperating category elements and competing categories shows up upon tuning. We illustrate this feature for categories 4 and 9 (this being a ‘hard task’) and for categories 0 and 1 (this being an ‘easy task’). Figure 9 shows the former case. For both categories, the average of the input vectors and the Euclidean distance between these averages were computed. In turn, for each memory (and for each input) the Euclidean distance for these averages was computed too. In other words, for each memory (and each input) we have a triangle determined by the three distances: the distance between the averages and the distances between the given memory (or the input) and the averages. The base of these triangles is always the same, and is placed on the horizontal axis of the subfigures with small vertical ticks denoting the end points of the basic set symmetrical to the vertical axis. The opposite corner of the triangles is denoted by dots. The left (right) hand side subfigures show the data for digit 4 (9). The top row depicts all the samples of the database that belong to these categories. The middle row depicts just 100 elements for each categories that was selected in 20 different training sessions utilising Training Rule 1 and allowing for five memories in each case. The bottom row depicts the area normalised histograms for these cases with white lines depicting the full distribution and black lines depicting the distribution of the selected memories. One can see that in this latter case there is a shift towards the boundary. The shift can be characterised by the difference between the average positions of all of the samples and the average position of the selected samples (denoted by $\overline{\chi}$). Also, the standard deviation of the histograms ($\sigma[\chi]$) is larger for the selected memories than for the full sample set. The

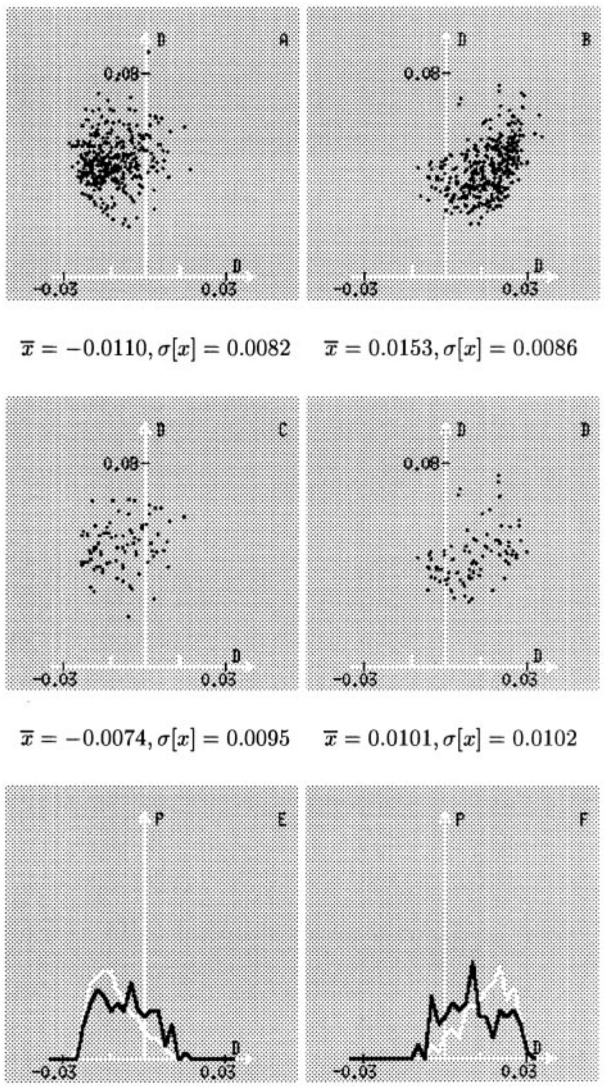

Fig. 9. Distribution of samples and memories for digits 4 and 9. Top row: all samples. Middle row: memories selected according to Training Rule 1. Bottom row: histograms for both cases (white: all samples, black: selected memories). Left (right) column: data for digit 4 (9). $\bar{x}$: average position from centreplane, $\sigma[x]$: standard deviation. For details, see text.

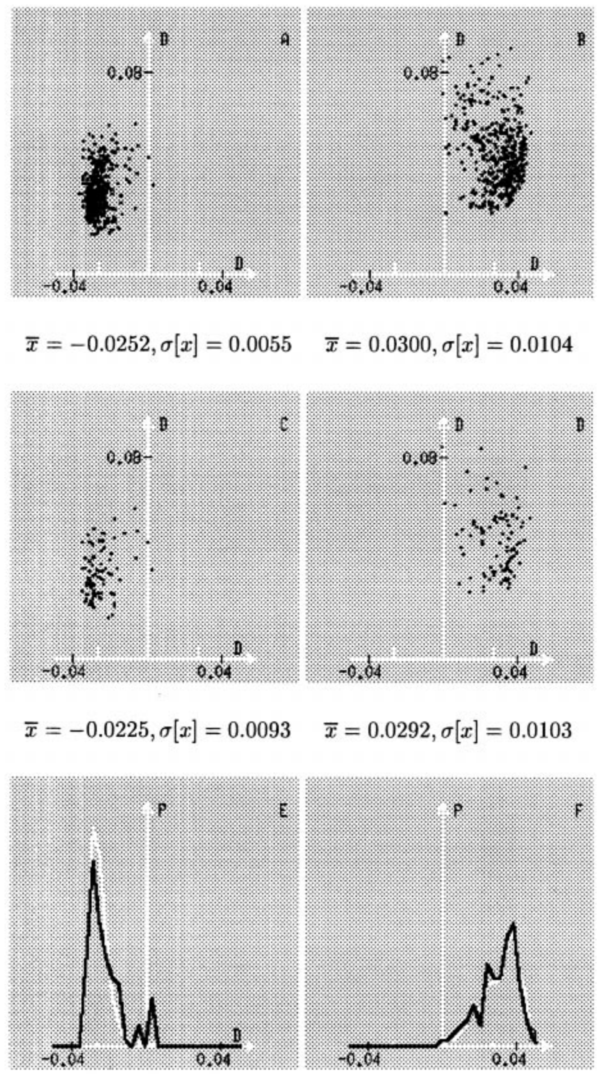

Fig. 10. Distribution of samples and memories for digits 0 and 1. Top row: all samples. Middle row: memories selected according to Training Rule 1. Bottom row: histograms for both cases (white: all samples, black: selected memories). Left (right) column: data for digit 0 (1). $\bar{x}$: average position from centreplane, $\sigma[x]$: standard deviation. For details, see text.

shift is less pronounced for the 'easy task', i.e. for the same test made for categories 0 and 1 (Fig. 10). The relationships with the phenomenon called Learned Categorical Perception (LCP) will be discussed in the next section.

5. Discussion

The pseudoinverse procedure and the WTA procedure together are proposed for categorisation purposes, as this combination establishes cooperative-competitive behaviour: memories belonging to a given category cooperate and compete with other memory sets belonging to other categories. The DCR pseudoinverse architecture allows one to use inputs as memories without any further tuning. It also introduces a natural GOF measure that determines the winner. The pseudoinverse activity optimisation, and thus the decision making is not formulated in terms of neural activities nor in terms of similarities (or closeness) between inputs and memory vectors, but rather in terms of the reconstruction error. The pseudoinverse procedure can be important, for example, for nets coding independent components [26–30], since these components are generally not orthogonal.

The disadvantage of the DCR pseudoinverse architecture is that it uses a recurrent network. It has been illustrated that for categorisation purposes, and for the digit task, a small number of iterations (around 10) seems suitable with the appropriate choice of the gain factor λ times Δt. This choice, and the resulting number of necessary iterations, is a function of the memory content. It can be shown that the inverse of the convergence rate is given by λ times the smallest eigenvalue of matrix $Q^T Q$. The closer the memories are the longer the iteration will be. It is thus reasonable to choose memories that are as different as possible from each other. Also, if the inputs themselves are to be stored at any of the levels of a DCR architecture, the boundaries should be more densely represented for correct decision – as has been demonstrated by the numerical study. That is, one is forced to make a compromise between convergence rate and correct decision making. It seems reasonable to design a tuning protocol that does not admit very similar memories and rejects new memories with small Euclidean distance to any of the old memories.

It should be mentioned here that the number of different memories in the categories cannot be increased without limit, since the more different the memories the larger the subspace of perfect reconstruction and the smaller the difference between GOF measures of different categories will be. This feature points to the need for hierarchical reconstruction architectures that allow 'local feature extraction'. One option is to extend the Neocognitron architecture [31] with the DCR scheme.

A similar note is that that preprocessing stage that inputs dissimilar activations to the DCR net is strongly recommended. Independent component analysis is thus attractive, since it is closely related to sparse representation and prefers concentrated neural activities [22,32]. In this paper, no effort was made to develop such a preprocessing stage, e.g. in the form of Gabor filters: it means that the results could be improved.

If one considers how simple and fast the tuning procedure is, and also that pseudoinverse computation offers many advantages over other methods, the results are promising. Some of the advantages are given here. Namely, the dynamic pseudoinverse computation is robust against corrupting effects that do not strongly influence the subspace determined by the memory vectors. For example, it is robust against noise. It is also robust against contrast and against the change of sign since those keep the same subspace. Consider also the situation when an input is given by the difference of two similar memory vectors. For example, the outline of a figure could be considered as the difference between an input and the same input somewhat enlarged. Such inputs are hard to deal with for any network that utilises thresholding, since the outline has a much smaller level of excitatory input than the full figure. In other words, the outline is close to orthogonal for each of the memory vectors, and gives a small similarity measure, i.e. a small activation in feedforward nets. At the same time, the difference of the two input vectors is within the subspace determined by these two vectors and the pseudoinverse computation will provide zero error in the reconstruction.

The present tuning procedure could be improved in many ways, e.g. by slow tuning instead of selecting. The important feature of this tuning protocol, however, is that the memories store the original inputs. In this particular case, the original inputs were developed from grey scale images, and upon segmentation and thresholding bit valued inputs were formed (by NIST), and these bit valued inputs used as memories are capable of dealing with grey scale inputs. The single bit connection strength feature can save memory [33]. Moreover, this property allows for on-chip learning. Consider, for example, mixed (analogue and digital) VLSI chips [18–20] that utilise bit valued memories for each connection in an analogue network, and can set the bit values from outside. The dynamic pseudoinverse procedure suggests that a similar construct might suit the requirements of the pseudoinverse computation with highly relaxed requirements on connection strengths. The studies on the requirements of the precision of the memories where the compressing matrix and the reconstruction matrix may differ are fairly promising (see Table 3), since network performance is still hardly influenced by imprecisions that fall very far from the margins of analogue VLSI technology.

The selection procedure gave rise to memories that are more closely placed to the boundaries. Closer investigation of these memories reveals that some of the examples that were miscategorised were created by erroneous segmentation. If the upper right side of digit 9 is cut the result is very close to digit 4, for example. The turning procedure should be rather sensitive to this type of error, since it mostly represents the boundary as opposed to learning vector quantisation [34], which withdraws memories from dubious regions. The results could be improved by careful preselection of training samples.

5.1. Implications on Human Psychophysical Experiments. The central feature of our model is that it includes better representation of the digits, further away from the average representation of a given digit category. This increases stimulus sensitivity at the dubious region of the category boundary, and runs parallel with the psychophysical phenomenon of Learned Categorical Perception (LCP). LCP is the phenomenon when – after learning to categorise the stimuli – people perceive interstimulus differences as smaller within categories and larger between categories. In other words, this categorisation process, the active learning of new concepts, influences the perceptual capacity of the system as well. After people became familiar with a category, they tend to perceive the differences between the members of the same class as being smaller, i.e. they look more alike (within category compression), whereas they tend to perceive the differences between members of different categories as being larger, i.e. they look more different (between category expansion). This phenomenon of changes perceptual sensitivity is called categorical perception [35], and can also be induced by complicated learning of categorisation and naming of unfamiliar stimuli LCP [36–39].

A basic feature of our model is that after the reconstruction process is completed, the system receives a corrective feedback signal or reinforcement about its decision by the supervisor. If the winning category of the digit reconstruction procedure is correct, then the system stops and waits for the next incoming digit. However, if the winning category is incorrect (as in the example of Fig. 3) something interesting happens: as a consequence of the feedback signal, the internal representation of one of the five neurons of the digit category that was not recognised correctly is replaced by the representation of the input digit that was miscategorised. To demonstrate the consequences of the above procedure, let us make the task of the system simpler: suppose the system has to decide if the input digit was one of two possible digits, either a ‘four’ or a ‘nine’. It is obvious from the average representations of Fig. 2 that the digits ‘4’ and ‘9’ are physically rather similar to each other. This relatively close similarity is shown in numerical form in Table 4 that presents the data for Euclidean distances between normalised averages of digit sets 0 and 1 and digit sets 4 and 9. In view of the similarity of the two digits, it is likely that an input image will be miscategorised. Miscategorisation of, say, digit ‘4’ occurs when the Euclidean distance of the reconstructed image to the average representation of digit ‘9’ is smaller than that of the image to digit ‘4’. This probably happens when the input digit is not typical of its category (such as the atypical digit ‘3’ of Fig. 3, that resembles a digit ‘2’ rather than a ‘3’). In such cases, the corrective feedback signal instructs the model to put the input digit into one of the memories of its digit category. After a few such implementations where the categorisation process fails, the memory representation of digit ‘4’ will contain solely miscategorised or atypical digits of our sample. This has one important advantage: it enables the system to respond correctly to atypical elements in the future if a previously miscategorised digit reappears, since it is already stored in the memory and will thus correctly match its category, leading to correct categorisation. In other words, after the learning process, the memory units contain the representation of digits that are displaced from the digit’s own category’s central tendency or prototype in the direction of the boundary between the two categories. As a consequence, the average Euclidean distance between the memory units for the two digit categories to be judged decreases. LCP is the consequence of the redistribution of the memory units, and this redistribution manifests itself by better categorisation and also in relative tests that are independent of the categorisation task itself, and would measure the capacity of the full memory sets when a set of inputs is presented in pairs so that we measure how hard it is to tell them apart. If we consider, for example, the change of the reconstructing neural activities of the two digit categories as the measure of sensitivity for judging whether two consecutive patterns are the same or different, we can estimate (according to Section 2.2) that the sensitivity of the system is increased in the vicinity of the category boundary upon training. This means that the sensitivity for those digits that straddle the two categories became higher, whereas the sensitivity for digits that fall into the same digit category became lower. This is even more so when considering the dynamic contrast enhancement properties of the DCR scheme. This behaviour of our model is similar to that of Goldstone et al. [40]. In their model a cooperative/neighbour training algorithm and feedback of classification error in the learning rule were utilised. The result of an incorrect categorisation of a hidden unit is that, by neighbour learning, the unit attracts other units to its region of representation. Since incorrect categorisation happens more frequently at around the category boundary, this

Table 4. Euclidean distances between averages of sample sets 0 and 1 and of sample sets 4 and 9 using normalised samples.

Problem	Euclidean distance
0–1	0.053815
4–9	0.025144

region will be far better populated by detectors than other regions of the feature space and this explains LCP. The present model also increases the representation density at around the boundary, and offers an alternative model which is based on input reconstruction instead of neighbour training.

5.2. Relations to Other Neural Networks. The WTA network made of DCR subnets can be related to other networks. Important extensions arise when the WTA principle is extended with neighbour training properties alike to Kohonen networks [34]. In the Kohonen network prewired local connections, called neighbour connections, exist among the competing neural units. These connections define the topology of the network. Learning information is shared between neighbours. The Kohonen-like neighbour training speeds up learning, an attractive feature of these connections. Neighbour training diminishes during training giving rise to the original WTA-like training rule. We see the DCR idea that can be used to extend both the WTA and the Kohonen networks: units of the WTA and the Kohonen networks can be replaced by DCR subnets, giving rise to improved response properties for each unit (now a DCR subnet) and a hierarchical network construction.

It has been shown that prewired neighbour connections improve training properties of the network. This improvement, however, is limited by the prewired nature of these neighbouring connections. It has been shown that neighbouring connections may assume adaptivity, and can learn the topology of the external space [41–43]. This is another possibility for the hierarchical construction: use self-organising neighbour training to keep adaptivity in a changing world.

Finally, we would like to mention that there are close similarities as well as important differences between our network and the Adaptive Resonance Theory (ART) network of Carpenter and Grossberg [8]. The ART is a two layer network utilising a layer that makes contrast enhancement, and that can work as a WTA layer. This is the similarity between the two networks: both networks may be viewed as a set of WTA subnets. The difference between the two types of networks is in the dynamics. The ART network utilises resonances between input and the categorising units. Our network utilises the concept of reconstruction within the WTA subnets. This important difference can be illustrated with the 'EF problem'. ART is sensitive to the 'EF' problem, the problem when one of the inputs ('F') is a subset of the other input ('E'). In this case, input 'F' will excite the categorising unit responsible for the recognition of input 'E'. Without special care, this second unit may win the competition because it has more/stronger connections to the input layer. The DCR network solves the 'EF' problem in a natural fashion. The reconstruction principle renders smaller output to the wrong subnets: subnet 'E' will respond less to input 'F' than subnet 'F' when the input is 'F' and *vice versa*.

6. Conclusions

Pseudoinverse computation that can be considered as a Data Compression and Reconstruction (DCR) scheme was combined with the winner-take-all paradigm architecture for categorisation purposes. It has been shown that this hierarchy forms a competitive-cooperative-competitive scheme with competing (inhibitory) interactions between memories that belong to the same category, and that the result of this competition is the cooperative reconstuction of a given input by the memories of the category that, in turn, is followed by the competition between the categories.

The pseudoinverse procedure has many advantages, such as relatively low sensitivity to noise, insensitivity to contrast changes, low sensitivity to background levels and memory imprecisions. The price paid is the iterative feature of the algorithm. One could use the memory matrix and its pseudoinverse to avoid iteration, however, this solution loses the attractive features of the iconic (single bit) memories, for the pseudoinverse of the memory matrix. The architecture is of exponential speed, and the convergence rate is governed by the eigenvalues of the memory correlation matrix. With careful choice of the memories fast convergence (i.e. small iteration number) may be kept. This desire is, however, in conflict with the optimal choice of memory vectors that increases the densities of the memory vectors in dubious regions. This issue clearly calls for hierarchical constructions where the small differences become features at a higher level, and do not disturb the rate of the lower level reconstruction. The way how the competitive WTA layer modifies the distribution of the selected memories resembles the phenomenon of categorical perception found in human psychophysical experiments.

If the memory matrices are tuned so that they become orthogonal, e.g. by means of PCA tuning, then the network becomes a feedforward network and the pseudoinverse approach includes this option. The disadvantage of the orthogonal memories is that very small activities should sometimes be developed [22], and such activities can easily be buried by noise. The preferred choice is a representation that concentrates the neural activities into just a few units at a time, i.e. the preferred choice is sparse representation. Sparse representation, however, is not necessarily orthogonal, and the internal activities may be rather similar (think of similar memory vectors such as overlapping Gabor filters) unless the input is reconstructed and the false activities are suppressed by the requirement of reconstruction. Thus, noise considerations favour sparse representation, sparse representation calls for input reconstruction, and input reconstruction with sparse representation is possible with the DCR architecture.

It has been shown that the generalisation properties of the pseudoinverse procedure allows one to utilise connections represented by single bits, and that such bits can be imprecise. This feature is attractive for computers, since bit operations can be utilised instead of floating points, and also for mixed VLSI technology and on-chip learning.

Acknowledgements. This work was partially supported by Hungarian OTKA grants T14566 and T17100, by grant #JF No. 519 of the US-Hungarian Joint Fund and by grant #B4/97 of Hungarian-Flemish International Scientific and Technological Cooperation.

References

1. Rosenblatt F. Principles of Neurodynamics. Washington, DC: Spartan Books, 1962
2. Werbos PJ. Beyond regression: New tools for prediction and analysis in the behavioral sciences. PhD thesis, Harvard University, Cambridge, MA, 1974
3. Rumelhart DE, McClelland JL. Parallel Distributed Processing: Explorations in the microstructure of cognition, Vol. 1. Cambridge, MA: MIT Press 1986
4. Amari S. A theory of adaptive pattern classifiers. IEEE Transactions on Systems, Man and Cybernetics 1967; 2: 643–657
5. von der Malsburg C. Self-organisation of orientation sensitive cells in the striate cortex. Kybernetik 1973; 14: 85–100
6. Grossberg S. Adaptive pattern classification and universal recoding: I. Parallel development and coding of neural detectors. Biological Cybernetics 1976; 23: 121–134
7. Ackley DH, Hinton GE, Sejnowski TJ. A learning algorithm for Boltzmann machines. Cognitive Science 1985; 9: 147–169
8. Carpenter GA, Grossberg S. A massively parallel architecture for a self-organizing neural pattern recognition machine. Computer Vision, Graphics and Image Processing 1987; 37: 54–115
9. Pearson K. On lines and planes of closest fit to systems of points in space. Philosophical Magazine 1991; 2: 559–572
10. Karhunen K. Ueber lineare Methoden in der Wahrscheinlich-keitsrechnung. Annales Academiae Scientiarum Fennicae, Series A1: Mathematica-Physica 1947; 37: 3–79
11. Oja E. A simplified neuron model as a principal component analyzer. Mathematical Biology 1982; 15: 267–273
12. Sanger TD. Optimal unsupervised learning in a single-layer linear feedforward neural network. Neural Networks 1989; 12: 459–473
13. Kung SY, Diamantaras CI. A neural network learning algorithm for adaptive principal component extraction. International Conference on Acoustics, Speech and Signal Processing, vol. 2, Albuquerque, NM, 1990, pp 861–864
14. Oláh B, Szepesvári C. Complexity of learning: The case of everyday neural networks. Proceedings of IEEE International Conference on Neural Networks. IEEE World Congress on Computational Intelligence, Orlando. FL 1994, pp 61–65
15. Wittmeyer H. Ueber die Loesung von linearen Gleichungssystemen durch Iteration. Z Angew Mat Mech 1936; 16: 301–310
16. Fomin T, Körmendy-Rácz J, Lörincz A. Towards a unified model of cortical computation I. Data compression and data reconstruction using dynamic state feedback. Neural Network World 1997; 7(2): 121–135
17. Joutsensalo J, Miettinen A, Zeindl M. Nonlinear dimension reduction by combining competitive and distributed learning. International Conference on Artificial Neural Networks, Paris, France, 1995, pp 395–400
18. Hollis PW, Paulos JJ. Artificial neural networks using MOS analog multipliers. IEEE Journal of Solid State Circuits 1990; 25(3)
19. Masa P, Hoen K, Wallinga H. A high-speed analog neural processor. IEEE Micro 1994: 40–50
20. Tsividis Y. Mixed Analog Digital VLSI Devices and Technology: An Introduction. New York: McGraw Hill, 1996
21. Rao RPN, Ballard DH. Kalman filter model of the visual cortex. Neural Computation 1997; 9(4): 721–763
22. Olshausen A, Field DJ. Emergence of simple-cell receptive field properties by learning a sparse code for natural images. Nature 1996; 381: 607–609
23. Stewart-Bartlett M, Sejnowski TJ. Viewpoint invariant face recognition using independent component analysis and attractor networks. Advances in Neural Information Processing Systems 1997; 9: 817–823
24. Beymer D, Poggio T. Image representations for visual learning. Science 1996; 272: 1905–1909
25. Lörincz A. Towards a unified model of cortical computation II. From control architecture to a model of consciousness. Neural Network World 1997; 7(2): 137–152
26. Jutten C, Herault J. Blind separation of sources, Part I: An adaptive algorithm based on neuromimetic architecture. Signal Processing 1991; 24: 1–10
27. Wang L, Karhunen J, Oja E. A bigradient optimisation approach for robust PCA, MCA, and source separation. Proc IEEE Int Conf on Neural Networks, Perth, Australia, November 1995, pp 1684–1689
28. Comon, P. Independent component analysis – A new concept? Signal Processing 1994; 36: 287–314

29. Berns GS, Sejnowksi TJ. How the basal ganglia make decisions. The Neurobiology of Decision Making. Springer-Verlag, 1995
30. Karhunen J, Oja E, Wang I, Vigário R, Joutsensalo J. A class of neural networks for independent component analysis. IEEE Transactions on Neural Networks 1997; 8: 486–504
31. Fukushima K. Neocognitron: A self-organizing neural network model for a mechanism of pattern recognition unaffected by shift in position. Biological Cybernetics 1980; 36: 193–202
32. Olshausen A, Field DJ. Sparse coding with an overcomplete basis set: A strategy employed by v1? Vision Research 1997; 37: 3311–3325
33. Aleksander I, Clarke T, Braga A. Binary neural systems: combining weighted and weightless properties. IEE Journal of Intell Syst Eng 1994; 3: 211–220
34. Kohonen T. Self-Organisation and Associative Memory. Berlin: Springer-Verlag, 1984
35. Harnad S. Categorical Perception: The groundwork of cognition. Cambridge University Press, 1987
36. Andrews J, Livingston K, Harnad S, Fisher U. Are concepts grounded in categorical perception? Some relevant empirical results. Annual Meeting of Society for Philosophy and Psychology, Memphis, TN, June 1994
37. Livingston K, Andrews J, Harnad S. Categorical perception effects induced by category learning. Journal of Experimental Psychology: Learning, Memory and Cognition 1998; 24: 732–759
38. Beale J, Keil F. Categorical perception as an acquired phenomenon: What are the implications? Workshops in Computing Series. Springer-Verlag, Berlin, 1996
39. Goldstone R. Influences of categorisation on perceptual discrimination. J Exp Psych Gen 1994; 123(2): 178–200
40. Goldstone R, Steyvers M, Larimer K. Categorical perception of novel dimensions. Proceedings of the Eighteenth Annual Conference of the Cognitive Science Society. Hillsdale, NJ: Lawrence Erlbaum, 1996, pp 243–248
41. Martinez T, Schulten K. Topology representing networks. Neural Networks 1994; 7: 507–522
42. Szepesvári Cs, Balázs L, Lörincz A. Topology learning solved by extended objects: A neural network model. Neural Computations 1994; 6: 441–448
43. Szepesvári Cs, Lörincz A. Approximate geometry representations and sensory fusion. Neurocomputing 1996; 12: 267–287