

0031-3203(94)00181-2

AN ARTIFICIAL NEURAL NETWORK SYSTEM FOR
TEMPORAL-SPATIAL SEQUENCE PROCESSING

LIPO WANG*† and DANIEL L. ALKON*‡

*National Institutes of Health, NINDS, 9000 Rockville Pike, Building 36, Room 4A22, Bethesda, MD 20892,
U.S.A

(Received 20 May 1994; in revised form 13 December 1994; received for publication 3 January 1995)

Abstract—An artificial network is described that can learn, recognize, and generate higher-order temporal-spatial sequences. It consists of three parts: (1) comparator units, (2) a parallel array of artificial neural networks that are derived from the visual-vestibular networks of the snail *Hermisenda*, as well as hippocampal neuroanatomy, and (3) delayed feedback lines from the output of the system to the neural network layer. Its advantages include short training time, fast and accurate retrievals, toleration of spatial noise and temporal gaps in test sequences, and ability to store a large number of temporal sequences consisting of *non-orthogonal* spatial patterns.

Temporal sequences Neural networks Comparator Dystal Delay Noise

1. INTRODUCTION

The brain is constantly processing temporal-spatial information, since the environment is constantly changing with respect to time. In particular, the brain often needs to dynamically learn and recall information. Hence understanding of temporal adaptive processes in the brain is of paramount importance. One fruitful way to achieve this understanding is to build and investigate various artificial neural network (ANN) models. A large body of work has been generated on ANN for *static* processing [e.g., references (1–12)] and on temporal processing [e.g., references (13–36)].

Temporal processing may mean one or more of the following: learning, recalling, classifying, generalizing, or generating time-dependent phenomena. Existing ANN algorithms for temporal processing may be divided into two categories, (i) with time delays,^(13–20,22–25,29–32) (ii) without time delays.^(21,26–28,33–36)

In a variety of systems including physical and chemical systems [e.g., reference (37)], and artificial neural networks,^(13–20,22–25,29–32) time-delays help represent temporal sequences. Grossberg's Avalanche model^(13–19) was presented in the form of delayed partial-differential-difference equations. Although it was mathematically proven that this model is capable of learning temporal-spatial sequences after infinite number of presentations of the training sequences, the model is computationally expensive and its practical

abilities to process temporal-spatial sequences have not been demonstrated. Fukushima⁽²⁰⁾ presented a temporal processing system, in which a number of McCulloch-Pitts neurons are fully connected with Hebbian-type synapses. There are multiple synapses between any two neurons and different time-delays in these synapses. Fukushima's system is capable of associating a spatial pattern with a pattern present at a previous time [see reference (20) for details of the formulation]. This formulation shows limited ability to store sequences, i.e., it rapidly saturates because it uses a Hebbian-type inner-product learning rule and it has *fixed* number of processing units. Furthermore, it requires many iterations for sequence retrieval and discriminates non-orthogonal patterns with difficulty. Images retrieved by this system are often obscured by noise (spurious memories). Time delays have been incorporated into Hopfield networks^(3,4) to generate temporal-spatial sequences^(22–25) and to process speech signals.⁽³²⁾ These systems also use Hebb-type⁽³⁸⁾ learning rules and have problems similar to those of Fukushima's system. The ANN discussed by Guyon *et al.*⁽²⁹⁾ requires that all stored sequences are known analytically *a priori*. After synaptic connections are calculated, any additional sequences that need to be stored in the system require reconstruction of the entire synaptic connectivity. Time delays have also been used together with back-propagation networks in processing temporal speech signals,⁽³⁰⁾ although back-propagation networks are known to have long training times.

A number of ANNs are capable of generating temporal sequences without time delays.^(21,26–28,33–36) Stochastic noise⁽²⁶⁾ has been used to induce transitions between attractors in Hopfield networks. Other exist-

† Present address: School of Computing and Mathematics, Deakin University, 662 Blackburn Road, Clayton, Victoria 3168 Australia.

‡ Author to whom correspondence should be addressed.

ing mechanisms are time-dependent,^(21,27) asymmetric^(28,33) and diluted higher order⁽³⁴⁻³⁶⁾ synaptic interactions. These ANNs also require that all stored sequences are known analytically *a priori* and therefore are not suited for practical applications in which numerical databases are used and training may be mixed with testing.

As discussed above, despite the great effort invested in the area of temporal sequence processing with artificial neural networks, various problems, such as slow and inaccurate training and testing, rapid memory storage saturation, strict orthogonality requirements, remain in the existing approaches. The primary goal of the present study is to provide a temporal processing scheme that eliminates these difficulties thereby markedly improving the efficiency of temporal sequence recognition.

In this paper we describe a network system which uses *biologically-derived* learning as incorporated into the Dystal network^(8,12) to learn temporal associations. As described below, this new system for learning and recognizing temporal sequences shows previously unachieved fast and accurate training and retrieval, does not require individual images in sequences to be orthogonal, and shows minimal saturation. Although the system is not a model of a specific biological network, it does include features that are directly derived from biological networks and are incorporated into the Dystal model used in the present system. These biological features, which will be discussed briefly in the next section, together with a novel design containing a parallel array of neural networks and a comparator layer efficiently solve the problem of sequence recognition with an artificial network system.

2. A BRIEF INTRODUCTION TO THE DYSTAL MODEL

Many recent neurobiological observations [see, e.g., references (39-42)] suggest that in biological neuronal networks learning can occur locally and independently of whether the post-synaptic neuron fires, i.e., independent of the output of a neuron. Furthermore, even for one association, memory involves the interaction of changes in more than one spatially distinct compartment of the same neuron.⁽⁴³⁾ The visual-vestibular network of the snail *Hermisenda*, for example, has been demonstrated to mediate Pavlovian conditioning induced by training with visual and vestibular stimuli precisely associated in time [Fig. 1(a) and (b)]. It is the correlation in time of the input stimuli rather than the output of the network neurons that changes the weights of synaptic signals. Visual-vestibular correlation transforms a non-Hebbian GABAergic synapse (between visual and vestibular neurons) from inhibitory to excitatory. These network properties are modeled in the DNN to eliminate the requirement for:

- (a) Hebbian correlation of input activation with output activation,
- (b) synaptic feedback from output to input layers, and

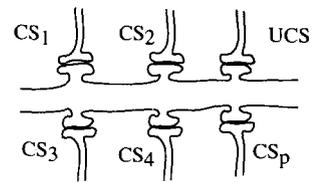
a. *Hermisenda* soma membrane change during learning



b. Model of dendritic change during learning



c. Model of a synaptic patch



d. Model of an artificial patch

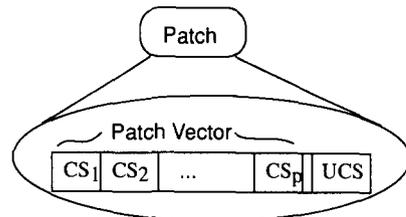


Fig. 1. (a) Schematic diagram for classical conditioning in *Hermisenda*. Paired light and rotation are followed in the type B cell soma by an enhanced long-lasting depolarizing response to light. Repeated pairings lead to cumulative membrane depolarization, enhanced response during and after the light, decreased voltage-dependent K^+ currents, increased input resistance, and thus increased excitability. On the days after training, i.e., during retention of the learning, the cumulative depolarization is no longer present, whereas the other changes remain (Alkon, 1984). (b) A simplified representation of (a). (c) A dendritic compartment of multiple synapses called a "patch". (d) A mathematical representation of an artificial patch.

(c) independent inhibitory and excitatory pathways for correlation and anticorrelation. Each of these learning network features reduces the complexity of the network architecture and improves computational efficiency.

In the snail network, correlation of visual and vestibular inputs causes elevation of intracellular calcium in spatially separated compartments of the same post-synaptic (Type B) neuron.^(43,44) Evidence has recently become available⁽⁴⁵⁾ that such non-Hebbian post-synaptic interaction also occurs hip-

pocampal pyramidal cell neurons during associative learning.^(42,46,47) The compartmental interaction of two correlated stimuli was, therefore, generalized to correlation and anti-correlation of multiple inputs on a shared post-synaptic dendritic branch, that we called a “patch”⁽⁴⁰⁾ [see Fig. 1(c) and (d)].

Since patches in the DNN are virtual, they are generated only by the patterns of stimuli received by the network during training. This allows a comparatively simple network organization to generate much greater complexity that is not explicitly preprogrammed. The patch allows for combinatorial specificity of distributed inputs in an image matrix and is quantitatively described by a patch vector. As demonstrated by Alkon *et al.*⁽⁸⁾ and Blackwell *et al.*⁽¹²⁾, combinatorial correlation of inputs described as patch vectors allows for extremely rapid convergence of the learning algorithm, no orthogonality requirement on input patterns, as well as a large memory capacity.

A Dystal neuron consists of a number of dendritic compartments (Fig. 2), or synaptic “patches”, which are created and modified during learning (see below). There are N_{CS} CS synapses (the patch vector) and one UCS synapse in each patch. A Dystal neural network (DNN) consists of N such Dystal neurons arranged in

parallel that share common CS and UCS input fields (Fig. 3). Each Dystal neuron operates independently during training and testing. We denote the dimension of the CS input vector by N_I . Usually $N_{CS} \leq N_I$, i.e., a patch evaluates only a portion of the input field.

In the training stage, paired CS and UCS training patterns are presented to the Dystal network’s CS and UCS synapses, respectively. The Dystal learning rule, which describes how patches are created and updated during learning, is as follows. Prior to training no patches exist; the first patch within a given Dystal neuron is created by correlating the incoming CS input pattern with an accompanying UCS value. Subsequently, the incoming CS input pattern is compared to the patch vector of every patch within each neuron.

Suppose that P^i is the i -th patch vector and S^i is the similarity of that CS input to the i -th patch. At the present time, Pearson’s R is used as the similarity measure; a normalized dot product would also be suitable. The patch with the greatest S , independent of the value of the UCS, is designated P_{m0} . The patch with the greatest S and with a sufficiently similar UCS is designated P_m with similarity S_m . If S_m exceeds T_H then that patch, which stores the particular correlation between the UCS and CS input patterns, is up-

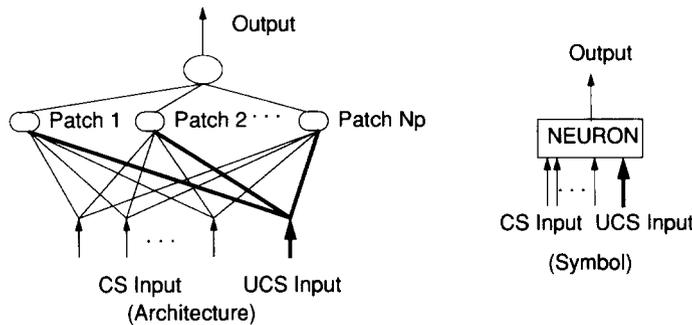


Fig. 2. A Dystal neuron and its patches. Each patch consists of the patch vector that contains the running average values of the CS and one component of the UCS. The left hand side shows the internal architecture of a Dystal neuron. The right hand side defines a symbolic representation of a Dystal neuron and this symbol is used in Fig. 3 to illustrate the design of a Dystal network.

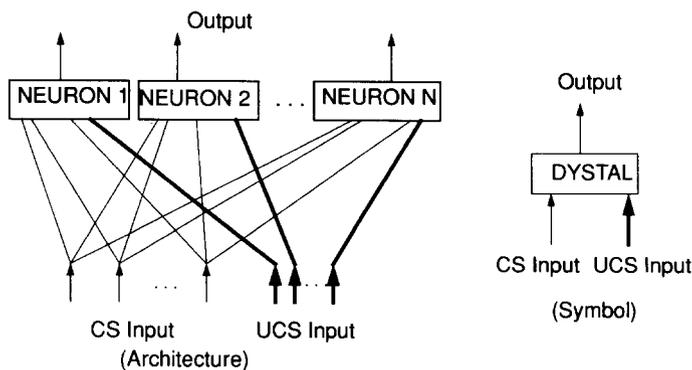


Fig. 3. A Dystal neural network. The right hand side shows its internal architecture. The right hand side defines its symbolic representation and this symbol is used in Fig. 5 to illustrate the design of the proposed temporal-spatial processing system.

dated by a running average:

$$P_m(\tau) = \frac{(\tau - 1)P_m(\tau - 1) + I_m(\tau)}{\tau},$$

where τ is the number of times that patch pattern P_m has been updated and the CS training input to patch P_m at time τ is $\{I_m(\tau)\}$. If S_m is less than T_L a new patch is created using the incoming CS input pattern and the corresponding UCS component. If S_m lies between T_L and T_H then a new patch is created only if P_m is not the same patch as P_{m_0} . Learning described above occurs simultaneously in all neurons in the network.

In the *testing* stage, only CS synapses of the Dystal network are presented with input patterns and the patches do not change. The more *similar* an input pattern is to the stored CS patch pattern, the stronger the patch responds. The most excited patch shunts all others and transmits the signal as the output. The output of a Dystal neuron equals the product of the UCS value of the most excited patch and the similarity of the CS input to the CS patch pattern in the most excited patch.

For more detailed mathematical descriptions on the Dystal design, learning rules, and applications, we refer the reader to references (8) and (12).

In the next section, we present a system that incorporates time delays, comparator units, and a parallel array of DNNs. This system is capable of learning and recognizing temporal-spatial sequences.

3. DESIGN AND OPERATION OF THE PROPOSED SYSTEM FOR TEMPORAL PROCESSING

The desired system should learn a number of temporal-spatial sequences after repeated presentations of these sequences or some variations, e.g., noisy versions, of these sequences. After learning is completed, the system should recall an entire sequence when presented only with a small portion of that sequence which may also be obscured by noise. Temporal-spatial sequences refer to spatial patterns that occur in sequence that can be referenced to successive time intervals. Figure 5 shows twenty examples of such sequences,

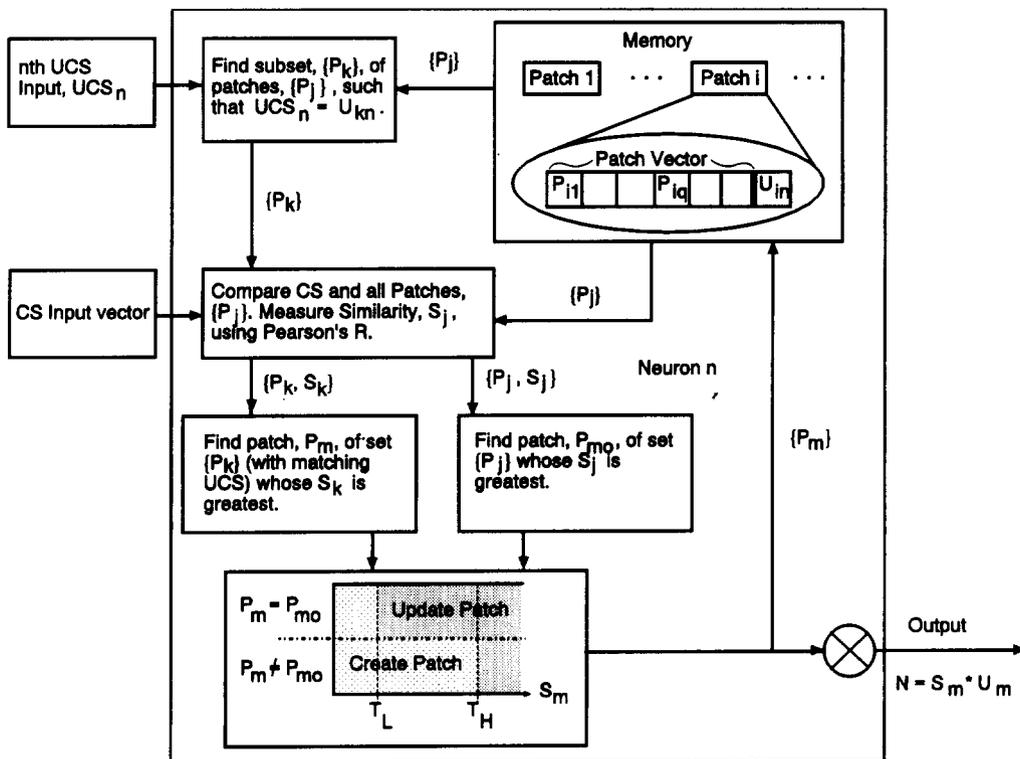


Fig. 4. An overview of the Dystal algorithm, which is executed independently for each output unit (the n th output unit is shown). A patch consists of the patch vector and the component of the UCS vector that is assigned to that output unit. Thus, the number of components in the UCS vector and the number of output units is the same. The incoming CS input pattern is compared to the patch vector of every patch stored in the output unit. P^i is the i th patch vector and S^i is the similarity of that CS input to the i th patch. At the present time, Pearson's R is used as the similarity measure; a normalized dot product would also be suitable. The patch with the greatest S , independent of the value of the UCS, is designated P_{m_0} . The patch with the greatest S and with a sufficiently similar UCS is designated P_m with similarity S_m . If S_m exceeds T_H then that patch is updated by a running average; if S_m is less than T_L a new patch is created. If S_m lies between T_L and T_H then a new patch is created only if P_m is not the same patch as P_{m_0} .

- (1) A B C D E F G H I
- (2) 1 2 3 4 5 6
- (3) J K L J M N J O P
- (4) Q R S T U S
- (5) V W X Y W Z
- (6) 7 F 8 7 9 0
- (7) 凹 八 巴 白 八 百
- (8) 半 办 北 半 贝 本
- (9) 比 币 并 丙 卜 并
- (10) 不 才 仓 厂 才 长
- (11) 尺 虫 丑 尺 出 十
- (12) 串 从 寸 达 丛 寸
- (13) 打 大 呆 石 大 旦
- (14) 当 刀 电 当 刁 吊
- (15) 丁 冬 斗 朵 厄 斗
- (16) 尔 耳 二 尔 而 凡
- (17) 儿 反 方 飞 分 方
- (18) 风 丰 夫 风 干 六
- (19) 之 一 七 万 三 七
- (20) 五 戈 平 久 戈 个

Fig. 5. Twenty sequences used to train the system before testing. Some of the testing results are presented in Fig. 7. Each "..." represents repetitions from the beginning of the sequence.

which consist of spatial images of English alphanumerals and Chinese characters.

Figure 6 shows schematically the architecture of our model for temporal processing that is able to achieve the above goal. It consists of three parts: (1) the comparator units, (2) the DNN layer, and (3) the

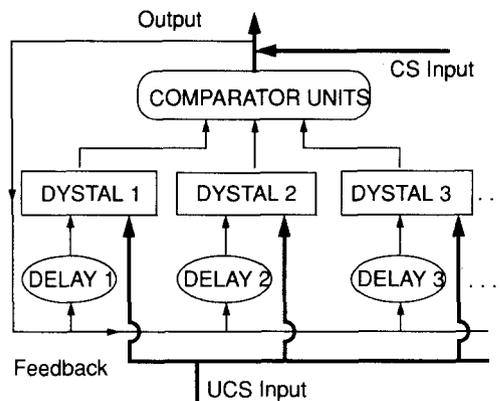


Fig. 6. Architectural design of the proposed system for temporal processing. The system consists of three parts: (1) comparator units, (2) a parallel array of Dystal neural networks (DNN), and (3) feedback lines with time delays from the output of the system to the DNN layer. The system has two input channels: the CS and the UCS channels. During training, pairs of sequences of spatial patterns are presented to the CS and the UCS channels simultaneously. After training, a test pattern, which is presented to the CS channel only, will cause the system to output images in a sequence that has been learned.

time delays. Let there be N_L DNNs in the network layer, N neurons in each DNN, and N comparator units in the system. The time delay associated with the l -th DNN delays the signal by l time steps with respect to the current time, where $l = 1, 2, \dots, N_L$. The system has two input channels: the CS and the UCS channels, which are analogous to classical conditioning and the DNN. There are two stages of operations: training and testing. During training, pairs of sequences of spatial patterns are presented to the CS and the UCS input channels simultaneously, whereas during testing sequences are presented only to the CS input channel.

Each comparator unit carries out a weighted average over the outputs of the corresponding neurons in all the DNNs that carry signals, i.e., $S_i(t) = \phi[\sum' a_l O_{il}(t) / \sum' a_l]$, $i = 1, 2, \dots, N$. Here $S_i(t)$ is the state of the i -th comparator unit of the system, $O_{il}(t)$ is the state of the i -th neuron in the l -th DNN, \sum' means a sum over only signal-carrying DNNs, and the function $\phi(x)$ rounds up x to the nearest gray shade value. The coefficients that measure the relative importance, i.e., $\{a_l, 1 \leq l \leq N_L\}$, are assumed in the present paper to be fixed and do not change during training. There are no particular requirements on how these coefficients should be chosen. The following rules for coefficient determination are, however, reasonable: (i) All DNNs are equally important, i.e., all coefficients are the same. (ii) The coefficients decrease monotonically for DNNs with larger delays, i.e., the earlier the events the less influence they have on the present processes. (iii) More excited, i.e., better matched, DNNs have larger coefficients. It is essential to use more than one DNN and time delay in the system when there are common spatial patterns among different sequences or when a spatial pattern appears

more than once in a sequence (higher order sequence), such as the cases shown in Fig. 5 where pattern "I", which is the same as pattern "1", appears in both sequences (1) and (2), "J" appears more than once in sequence (3), and "7" appears more than once in sequence (6)... The larger the number of DNNs in the network layer, the better the performance, and of course, the more computationally intensive the system will be. We emphasize that the feedback signals through the time delays do not perform any error minimization functions.

During testing, if an unknown or ambiguous sequence is presented to the system, the DNNs in the system output conflicting or "don't know" answers. A threshold, which may vary from task to task depending on the acceptable level of conflict or ambiguity, may be applied to allow a "don't know" answer for the present system so as to reduce the overall error rate. For example, if more than one-third of the signal-carrying DNNs in the system output "don't know" or conflicting answers, the comparator units also output a "don't know" answer and the sequence is unclassified by the system.

During training, paired sequences are presented to the system through the two input channels and learning is achieved through the hetero-associations in the embedded DNNs. One of the two sequences in each training pair, the "CS sequence", is fed into the "CS input" channel. The UCS sequence in the pair, the expected output of the system corresponding to the signal sequence, is fed into the "UCS input" channel. The two sequences in each training pair may be the same, or one may be a variation, e.g., a noisy or distorted version, of the other. Thus each training pair represents one temporal sequence to be stored in the system. After training, each DNN in the network layer has learned the correlations among pairs of patterns corresponding to different time steps and thereby the system has learned the temporal sequence.

During testing, a small piece of stored sequence, that need not be contiguous but may include gaps in the sequence, and which may or may not be obscured by noise, is presented to the system through the CS input channels, while the UCS input channels are not used during testing. The output will be the corresponding expected output sequence in a successful retrieval. The learning and recalling mechanisms of the system can be more clearly demonstrated through the following explicit examples.

We use three DNNs ($N_L = 3$) and choose the same connections between the comparator units of the system and the DNNs ($a_1 = a_2 = a_3 = 1$). We also choose both dimensions of the CS and UCS patterns to be $N = 11 \times 11$. We train the system to store twenty sequences given in Fig. 5 and then test the system.

During training, sequence (1) is presented simultaneously to the CS input and UCS input channels. At time $t = 1$, the output of the system is pattern A . The UCS input for the first DNN is also A , however, learning does not occur at $t = 1$, since there are no CS

inputs to any of the three DNNs from the delayed feedback. At time $t = 2$, the output of the system and the UCS input for the first DNN are both pattern B . The CS input to this DNN from the delayed feedback is the output of the system at the previous time step, which is pattern A . Hence the first DNN learns the association between the CS A and the UCS B at $t = 2$. The second and the third DNNs do not learn at $t = 2$. Similarly, at time $t = 3$, the first DNN learns the association between the CS B and the UCS C , the second DNN learns the association between the CS A and the UCS C , while the third DNN remains inactive. At $t = 4$, the UCS D is associated with C by the first DNN, with B by the second DNN, and with A by the third DNN... Sequence (1) is thus stored into the system. The system learns sequences (2) through (20) in the same way.

To illustrate how the system operates at the testing stage, let us consider the examples given in Fig. 7.

If " \bar{A} ", which denotes a noisy " A ", is presented to the system at time $t = 1$ [Fig. 7(a)], the system outputs an " \bar{A} ", at time $t = 1$. At $t = 2$, the CS input for the first DNN is the output of the system 1 time step before, which is " A ". Since the UCS corresponding to an " A " in the training stage is " B " and the amount of noise in \bar{A} is shown to be tolerable by the DNN, the output vector of the first DNN is a " B ". We notice that at time $t = 2$ other DNNs do not respond since they do not receive any input. Hence the output of the system at time $t = 2$ is " B ". At time $t = 3$ the input to the first DNN is the output of the system at time $t = 2$, which is " B ". Hence the output of the first DNN is " C ". Similarly the output of the second DNN is also " C " after receiving a CS input of " A ". There are still no inputs for the third DNN. Therefore the output of the system is " C " at time $t = 3$. We see that the system outputs sequence (1) when presented with a noisy pattern " A ". We observe that all retrieved images in sequence (1) are noise-free, whereas some retrieved images, i.e., " E " and " F ", are imperfect in Fukushima's system.⁽²⁰⁾

When an image " D " [Fig. 7(b)] is presented to Fukushima's system, the retrieval of sequence (1) becomes very difficult: it takes many iterations and many retrieved images are imperfect.⁽²⁰⁾ This is because Fukushima uses a Hebbian-type (dot-product) learning rule which imposes strict orthogonality requirement on all images in stored sequences.⁽²⁰⁾ To facilitate a performance comparison, we have used the same two sequences used by Fukushima⁽²⁰⁾ as our sequences (1) and (2), where pattern "1" is the same as pattern "I".

When an unknown sequence, e.g., " $\bar{D}\bar{C}\bar{B}$ " [Fig. 7(c)], is presented to the above trained system, the first, the second, and the third DNNs output " C ", " E ", and " G ", respectively. Since less than two-thirds of the DNNs give the same output at a given time step, the comparator units halt the feedback operation and output a "don't know" answer. Fukushima's system was not tested in this type of situation, but can be expected to

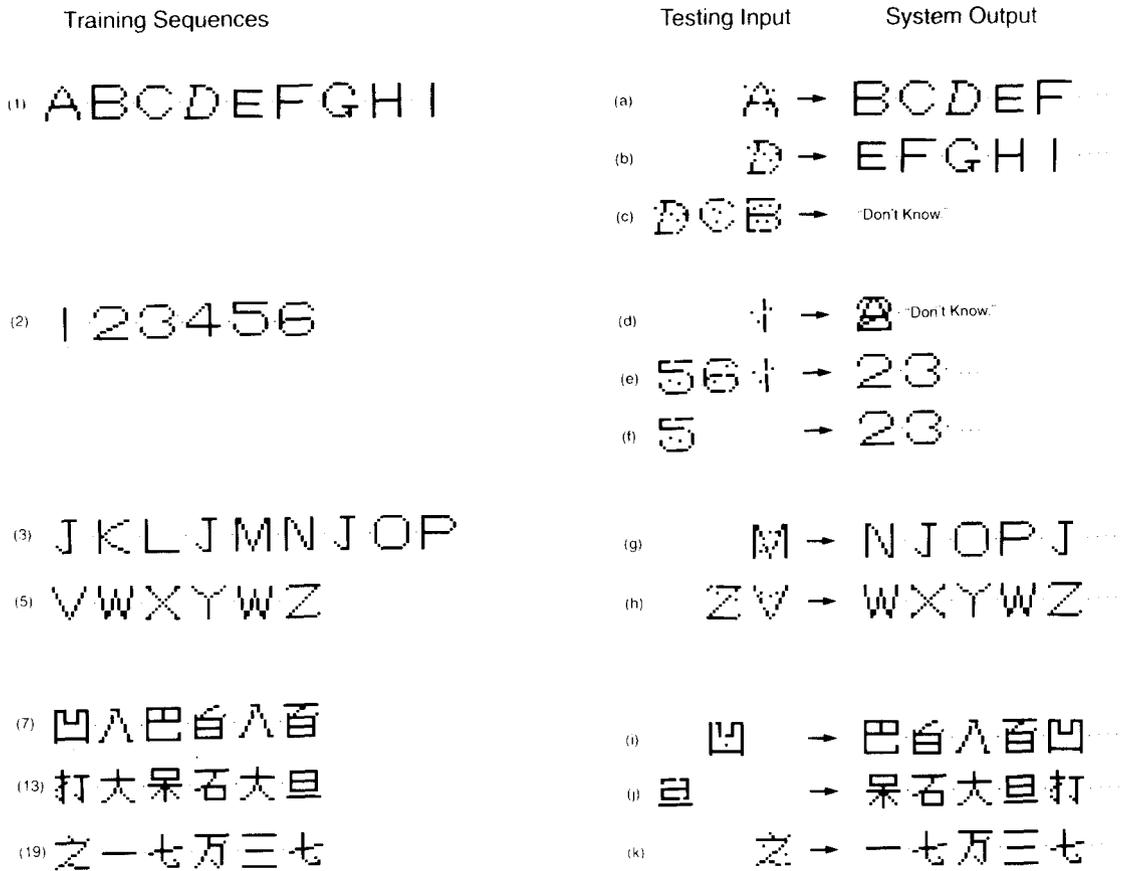


Fig. 7. Examples of the training and testing sequences, together with responses of the system to various input signals after training with the sequences shown in Fig. 5. Each sequence used in the computer experiment consists of a number of individual spatial patterns (English and Chinese characters) that are 11 × 11 pixels in size. Note that pattern "I", which is the same as pattern "1", appears in both sequences 1 and 2, and "J" appears more than once in sequence 3 (higher order sequence). All other sequences used in training, including sequences 5, 7, 13 and 19 shown in Fig. 7, are higher order sequences. The testing responses are as follows: (a) a noisy pattern "A" retrieves sequence (1); (b) a noisy pattern "I" is insufficient to make a retrieval; (c) more information is required in (b) and leads to a successful retrieval; (d) response to a non-orthogonal pattern "D"; (e) response to an unknown sequence; (f) response to input with missing images; (g)–(k) retrievals of high order sequences consisting of English and Chinese characters.

yield meaningless output since it is unable to give a "don't know" answer.

If initially at time $t = 1$ a "1" (a noisy "1" note that "I" is the same as "1") is presented to the system, DNN 1 becomes "confused" and outputs an "average" of "2" and "A", which is also the system output at $t = 2$. At $t = 3$, DNN 1 outputs "don't know", since its input, an "average" of "2" and "A", is not recognized by the first DNN. At $t = 3$, DNN 2 outputs an "average" of "3" and "B", since its input is a "1". The system outputs a "don't know" since one out of two signal-carrying DNNs outputs "don't know" [Fig. 7(d)]. The ability of outputting a "don't know" answer often can significantly reduce error rate in practical applications. We note that Fukushima's system outputs a meaningless sequence in this kind of situation.⁽²⁰⁾

If a longer piece of the sequence, e.g., "561", is presented to the system instead of a "1" alone, however, the system is able to recognize the sequence and

retrieve sequence (2) in the following way. The system outputs "5", "6", "1", at $t = 1, 2, 3$, respectively. At $t = 4$, the output of DNN 1 is an "average" of "2" and "A", however, the outputs of DNNs 2 and 3 are both "2". Hence the system outputs a "2" at time $t = 4$, etc. [Fig. 7(e)]. Similarly, when "GH1" is presented to the system instead of a "1" alone, the retrieval is Sequence (1).

Figure 7(f) shows an example of the proposed system's response, i.e., successful retrieval of sequence 2, when two frames of input sequence are missing. Figure 7(g) shows that the high order sequence learned during training is successfully retrieved when presented with a noisy M. Figure 7(h)–(k) show learning and recalling of higher order sequence consisting of other English letters and Chinese characters.

The other seventeen sequences stored in the system were also successfully retrieved.

4. SUMMARY AND DISCUSSION

We have proposed a system that can learn, recognize, and generate temporal-spatial sequences. After training with temporal sequences, the DNN system is able to recognize and generate the whole sequence after being presented with a small piece, which may or may not be obscured by noise and may or may not contain gaps, of a stored sequence. Or equivalently, after training and when a sequence of events is presented to the system, the system *predicts* the sequence of events in the future. Compared to the Fukushima⁽²⁰⁾ and Hopfield-type neural network temporal processing systems⁽²²⁻²⁵⁾, the novel features of the present temporal processing system include: fast and accurate training and response, noniterative function, few constraints on individual spatial images in sequences, and minimal saturation. Compared with back-propagation systems,⁽³⁰⁾ the DNN system is free from the problems of convergence and long training time. Compared with the system proposed by Guyon et al.,⁽²⁹⁾ the DNN system does not require analytical expressions of all training sequences and does not require reconstruction of the *entire* synaptic connectivity when additional training sequences are presented to the system. These properties demonstrated by the present system are very desirable for practical applications such as real time speech processing.

In some practical applications, signal patterns such as words occur at different rates. Existing temporal systems handle this problem with difficulty. To apply the present system for speech processing, we plan to preprocess the speech signals so that the signals are presented to the system at a pre-determined rate. We plan to address the rate-independence problem by incorporating the present system in a subsequent implementation with (1) variable time delays and (2) temporally persistent responses of biologically realistic neuronal elements.⁽⁸⁾

The biologically-based artificial neural network Dystal (DNN) is an important component of the present temporal processing system. As discussed briefly in Section 2, the compartmental features in the DNN and associated non-Hebbian learning rules that were derived directly from biological networks, have been shown by Alkon *et al.* (1990) and Blackwell *et al.* (1992) to be computational efficient in static pattern processing. We show in the present work that this compartmental model and learning algorithm, together with a novel design of a comparator layer and a parallel array of neural networks, are also powerful in temporal sequence processing. The design of the present system is not *necessary*, in a mathematical sense, for any form of temporal processing, although it provides very favorable functional properties compared with other existing approaches, e.g., those using simple two-state neurons.

Although the model for temporal processing presented here did not saturate after learning twenty sequences, it should on a theoretical basis store many

more sequences with minimal saturation. This is due to the fact that DNNs within the present model create memory "patches" as they learn new associations.^(8,12) The theoretical memory capacity, i.e., the maximum total number of different spatial images in all stored sequences, is 2^N , N being the number of neurons, whereas the memory capacities of systems using Hebbian-type learning rules^(20,22-25) are on the order of N . There are no additional limits on the number of sequences and the length of a sequence that can be stored. Since back-propagation systems (BPSs)⁽³⁰⁾ do not create new memories as they are trained, they saturate quickly, though conclusive investigations on memory capacities of BPSs are still lacking.

Although the artificial Dystal neurons are more complex compared to conventional artificial neurons, e.g., two-state neurons, the structural and computational complexities of a Dystal network are not higher than those of conventional artificial neural networks, since there are no recurrent interactions among Dystal neurons and the complexity of the Dystal neuron is more than offset by decreased complexity of connectivity among neurons thereby achieving substantial savings in software and hardware implementation. Compared to the Hopfield network, for example, the recurrent N^2 synapses among neurons are absent in the DNN (detailed discussions are the subject of another work).

When the DNNs in the present system are replaced by other kinds of hetero-associative neural network, e.g., a back-propagation network, the system, with a comparator layer and a parallel array of neural networks, can still process complex sequences, e.g., the ones shown in Fig. 5 where one image may occur in more than one stored sequences or one image may occur more than once in a stored sequence. However, other computational advantages originated from the DNN, such as minimal saturation, fast and accurate training and testing, are lost.

REFERENCES

1. W. S. McCulloch and W. Pitts, A logical calculus of the ideas immanent in nervous activity, *Bull. Math. Biophys.* **5**, 115-133 (1943).
2. G. A. Carpenter, S. Grossberg, N. Markuzon, J. H. Reynolds and D. B. Rosen, Fuzzy ARTMAP: a neural network architecture for incremental supervised learning of analog multidimensional maps, *IEEE Trans. Neural Networks* **3**, 698-713 (1992).
3. J. J. Hopfield, Neural networks and physical systems with emergent collective computational abilities, *Proc. Natl. Acad. Sci. U.S.A.* **79**, 2554-2558 (1982).
4. J. J. Hopfield, Learning algorithms and probability distributions in feed-forward and feed-back networks, *Proc. Natn. Acad. Sci. U.S.A.* **84**, 8429-8433 (1987).
5. D. E. Rumelhart and D. Zipser, Feature discovery by competitive learning, *Cognition Sci.* **9**, 75-112 (1985).
6. T. Kohonen, *Self Organization and Associative Memory*. 2nd edn, Springer Series in Information Science, Springer Verlag, Berlin (1988).
7. K. Fukushima, Neocognitron: a hierarchical neural net-

- work capable of visual pattern recognition, *Neural Networks* **1**, 119–130 (1988).
8. D. L. Alkon, K. T. Blackwell, G. S. Barbour, A. K. Rigler and T. P. Vogl, Pattern recognition by an artificial network derived from biologic neuronal systems, *Biol. Cybern.* **62**, 363–376 (1990).
 9. L. Wang and J. Ross, Synchronous neural networks of non-linear threshold elements with hysteresis, *Proc. Natn. Acad. Sci. U.S.A.* **87**, 988–992 (1990).
 10. L. Wang and J. Ross, Interactions of neural networks: Models for distraction and concentration, *Proc. Natn. Acad. Sci. U.S.A.* **87**, 7110–7114 (1990).
 11. L. Wang and J. Ross, A variable threshold as a model of selective attention, (de)sensitization, and anesthesia in associative neural networks, *Biol. Cybern.* **64**, 231–241 (1991).
 12. K. T. Blackwell, T. P. Vogl, S. D. Hyman, G. S. Barbour and D. L. Alkon, A new approach to hand-written character recognition, *Pattern Recognition* **25**, 655–666 (1992).
 13. S. Grossberg, Nonlinear difference-differential equations in prediction and learning theory, *Proc. Natn. Acad. Sci. U.S.A.* **58**, 1329–1334 (1967).
 14. S. Grossberg, On learning of spatiotemporal patterns by networks with ordered sensory and motor components, I: excitatory components of the cerebellum, *Studies Appl. Math.* **48**, 105–132 (1969).
 15. S. Grossberg, Some neural networks that can learn, remember, and reproduce any number of complicated space-time patterns, *Studies Appl. Math.* **49**, 135–166 (1970).
 16. S. Grossberg, Neural pattern discrimination, *J. Theor. Biol.* **27**, 291–337 (1970).
 17. S. Grossberg, *Studies of mind and brain: neural principles of learning, perception, development, cognition and motor control*. Reidel Press, Boston (1982).
 18. S. Grossberg, The adaptive self-organization of serial order in behavior: speech, language, and motor control, in *Pattern Recognition by Humans and Machines*, E. C. Schwab and H. C. Nusbaum, eds, Academic Press, New York, 187–294 (1986).
 19. S. Grossberg, *The Adaptive Brain II: Vision Speech, Language and Motor Control*, Elsevier/North-Holland, Amsterdam (1987).
 20. K. Fukushima, A model of associative memory in the brain, *Kybernetik* **12**, 58–63 (1973).
 21. P. Peretto and J. J. Niez, Collective properties of neural networks, in *Disordered Systems and Biological Organization*, Bienenstock E., eds, Springer-Verlag, New York (1985).
 22. H. Sompolinsky and I. Kanter, Temporal association in asymmetric neural networks, *Phy. Rev. Lett.* **57**, 2861–2864 (1986).
 23. D. Kleinfeld, Sequential state generation by model neural networks, *Proc. Natn. Acad. Sci. U.S.A.* **83**, 9469–9473 (1986).
 24. D. W. Tank and J. J. Hopfield, Neural computation by concentrating information in time, *Proc. Natn. Acad. Sci. U.S.A.* **84**, 1896–1900 (1987).
 25. A. V. M. Herz, Z. Li and J. L. van Hemmen, Statistical mechanics of temporal association in neural networks with delayed interactions, in *Advances in Neural Information Processing Systems*, R. P. Lippmann, J. E. Moody and D. S. Touretzky, eds, Morgan Kaufmann Publishers, San Mateo (1991).
 26. J. Buhmann and K. Schulten, Noise-driven temporal association in neural networks, *Europhys. Lett.* **4**, 1205–1209 (1987).
 27. S. Dehaence, J. P. Changeux and J. P. Nadal, Neural networks that learn temporal sequences by selection, *Proc. Natl. Acad. Sci. U.S.A.* **84**, 2727–2731 (1987).
 28. A. C. C. Coolen and Th. W. Ruijgrok, Image evolution in Hopfield networks, *Phys. Rev. A* **38**, 4253–4255 (1988).
 29. I. Guyon, L. Personnaz, J. P. Nadal and G. Dreyfus, Storage and retrieval of complex sequences in neural networks, *Phys. Rev. A* **38**, 6365–6372 (1988).
 30. R. P. Lippmann, Review of neural networks for speech recognition, *Neural Comput.* **1**, 1–38 (1989).
 31. A. Waibel, T. Hanazawa, G. Hinton and K. Shikano, Phoneme recognition using time-delay neural networks, *IEEE Trans. Acoustics, Speech Signal Process.* **37**, 328–339 (1989).
 32. K. P. Unnikrishnan, J. J. Hopfield and D. W. Tank, Connected-digit speaker-dependent speech recognition using a neural network with time-delayed connections, *IEEE Trans. Signal Process.* **39**, 698–713 (1991).
 33. H. Nishimori, T. Nakamura and M. Shiino, Retrieval of spatio-temporal sequence in asynchronous neural network *Phys. Rev. A* **41**, 3346–3354 (1990).
 34. L. Wang, E. E. Pichler and J. Ross, Oscillations and chaos in neural networks: an exactly solvable model, *Proc. Natn. Acad. Sci. U.S.A.* **87**, 9467–9471 (1990).
 35. L. Wang and J. Ross, Chaos, multiplicity, crisis and synchronicity in higher order neural networks, *Phys. Rev. A* **44**, R2259–2262 (1991).
 36. L. Wang and J. Ross, Physical modeling of neural networks, in *Methods in Neurosciences: Computers and Computations in the Neurosciences*, P. M. Conn, ed., Academic Press, San Diego (1992).
 37. J. Ross, S. Pugh and M. Schell, Spectral kinetics and the efficiency of (bio)chemical reactions, in *From Chemical to Biological Organization*, Markus, M., Müller, S. C. and Nocolis, G., eds, Springer, Berlin (1988).
 38. D. O. Hebb, *The Organization of Behavior*. John Wiley, New York (1949).
 39. D. L. Alkon, *Memory Traces in the Brain*. Cambridge University Press, Cambridge (1987).
 40. D. L. Alkon, Memory storage and neural systems. *Sci. Am.* **42**, 42–50 (July 1989).
 41. D. L. Alkon, *Memory's Voice*. Harper-Collins, New York (1992).
 42. J. L. Olds, M. L. Anderson, D. L. McPhie, L. D. Staten and D. L. Alkon, Imaging memory-specific changes in the distribution of protein kinase C within the hippocampus. *Science* **245**, 866–869 (1989).
 43. D. L. Alkon, J.-V. Sanchez-Andres, E. Ito, K. Oka, T. Yoshioka and C. Collin, Long-term transformation of an inhibitory into an excitatory GABA-ergic synaptic response, *Proc. Natn. Acad. Sci. U.S.A.* **89**, 11862–11866 (1992).
 44. E. Ito, K. Oka, C. Collin, B. G. Schreurs, M. Sakakibara and D. L. Alkon, Intracellular calcium signals are enhanced for days after Pavlovian conditioning, *J. Neurochem.* **90**, 8209–8213 (1993).
 45. C. J. Lee and D. L. Alkon, Modification of GABAergic synaptic response following associative pairings in rat hippocampal slice. *Soc. Neurosci. Abstr.* **19**, 1323 (1993).
 46. J. J. LoTurco, J. L. Olds, B. Bank and D. L. Alkon, Biophysical and biochemical records of associative memory in rabbit CA1 pyramidal neurons and Hermisenda B cells, in *Correlates of Classical Conditioning*, I. Gormezano and E. Wasserman, eds, Ch. 11, 231–251 (1992).
 47. D. A. Coulter, J. LoTurco, M. Kubota, J. F. Disterhoft, J. W. Moore and D. L. Alkon, Classical conditioning alters the amplitude and time course of the calcium-dependent afterhyperpolarization in rabbit hippocampal pyramidal cells. *J. Neurophysiol.* **61**, 971–981 (1989).

About the Author—LIPO WANG is a lecturer at the School of Computing and Mathematics, Deakin University, 662 Blackburn Road, Clayton, Victoria 3168, Australia. He received his BS in laser optics from the Changsha Institute of Technology, and his PhD in physics from Louisiana State University. His research interests include theories and applications of neural networks. He is the author or co-author of over 30 journal publications and co-editor of "Artificial Neural Networks: Oscillations, Chaos and Sequence Processing" (IEEE Computer Press, 1993, Los Alamitos, California). He can be reached by E-mail: lwang @ deakin.edu.au.

About the Author—DANIEL L. ALKON received his undergraduate degree in chemistry at the University of Pennsylvania and his M.D. at Cornell University. After finishing an internship in medicine, he joined the staff at the National Institute of Neurological Disorders and Stroke at the National Institutes of Health, Bethesda, MD, U.S.A. where he has remained for the past 24 years. He is now a Medical Director and Laboratory Chief at the Institute where he leads a multidisciplinary research program on the molecular and biophysical basis of memory in the brain. Alkon and his colleagues have uncovered cellular and molecular mechanisms of associative memory that are common to diverse species. These mechanisms include persistent K^+ current inactivation, PKC activation, protein phosphorylation, long-term synaptic transformation and alteration of neuronal branch architecture. Based on these mechanisms, Alkon's group has recently identified the first potentially clinically useful laboratory diagnosis of Alzheimer's disease. In recent years, in collaboration with Drs Thomas Vogl, Kim Blackwell and more recently Dr Lipo Wang, he has incorporated some of the principles he and his group have found in biological networks into the design of artificial pattern recognition networks.