

Multi-associative Neural Networks and Their Applications to Learning and Retrieving Complex Spatio-Temporal Sequences

Lipo Wang

Abstract—Based on the previous work of a number of authors, we discuss an important class of neural networks which we call *multi-associative neural networks* (MANN's) and which associate one pattern with multiple patterns. As a computationally efficient example of such networks, we describe a specific MANN, that is, a multi-associative, dynamically generated variant of the counterpropagation network (MCPN). As an application of MANN's, we design a *general* system that can learn and retrieve *complex* spatio-temporal sequences with *any* MANN. This system consists of comparator units, a parallel array of MANN's, and delayed feedback lines from the output of the system to the neural network layer. During learning, pairs of sequences of spatial patterns are presented to the system and the system learns to associate patterns at successive times in sequence. During retrieving, a cue sequence, which may be obscured by spatial noise and temporal gaps, causes the system to output the stored spatio-temporal sequence. We prove analytically that this system is capable of learning and generating any spatio-temporal sequences within the maximum complexity determined by the number of embedded MANN's, with the maximum length and number of sequences determined by the memory capacity of the embedded MANN's. To demonstrate the applicability of this general system, we present an implementation using the MCPN. The system shows desirable properties such as fast and accurate learning and retrieving, and ability to store a large number of complex sequences consisting of nonorthogonal spatial patterns.

Index Terms— Auto-associative, hetero-associative, multi-associative, neural network, noise, spatio-temporal sequence.

I. INTRODUCTION

A *HETERO*-associative neural network (HANN) associates a spatial pattern $\vec{P}^{(1)}$ with another pattern $\vec{P}^{(2)}$ which may or may not be the same as pattern $\vec{P}^{(1)}$, whereas an *auto*-associative neural network (AANN) associates a spatial pattern with itself, i.e., $\vec{P}^{(1)} = \vec{P}^{(2)}$ in an AANN. For example, the multilayer perceptron network [40], the counterpropagation network [25], and the bidirectional associative memory [32] are HANN's, whereas the Hopfield network [27] is an AANN. One pattern may often be associated with many patterns. For example [7], an image of a banana may be associated with not only a banana, but also a yellow object, a fruit, an oblong object, etc. Let us call a neural network that associates one spatial pattern with multiple patterns a *multi-associative*

neural network (MANN). Hirai [26] proposed a network that associates a pattern with a collection of patterns. Hagiwara [22] and coworkers [37] extended the bidirectional associative memory [32] to store associations among multiple patterns. These networks used covariance (inner-product) learning rules similar to that in the Hopfield network [27] and therefore have limited memory capacity. Owens *et al.* [38] attempted to reduce training time and improve the classification capability of the multilayer perceptron by incorporating multiple output layers. In this paper, we discuss the MANN in general, as well as a computationally efficient exemplar MANN. As a demonstration of the applicability of MANN's, we present a design of a system that is capable of learning and retrieving complex spatio-temporal sequences using any static MANN. A specific implementation of this general system shows a number of desirable advantages over existing spatio-temporal systems.

There has been extensive research activity in learning and generating temporal phenomena with artificial neural networks. The main motivation for creating and studying these models is to build intelligent artificial systems that mimic or even surpass certain aspects of biological intelligence. Spatio-temporal phenomena are particularly interesting due to their abundance in both nature and practical applications. For instance, a particular spatio-temporal scene triggers a response sequence in a robot. The survival of many animals also depends on the ability to learn and produce spatio-temporal sequences, e.g., a prey escapes from a predator with a series of maneuvers.

Grossberg [14]–[20] proved mathematically that his Avalanche model is capable of learning spatio-temporal sequences after an infinite number of presentations of the training sequences; however, the model, presented in the form of delayed partial-differential-difference equations, is computationally expensive and its practical capabilities to effectively generate spatio-temporal sequences, such as the ones discussed in the present paper, are yet to be demonstrated.

Fukushima's [13] spatio-temporal system consists of a number of binary neurons connected with delayed Hebb-type [24] covariance (inner-product) synapses [1]. This system required many iterations for sequence retrieval and retrieved nonorthogonal patterns with difficulty. Images generated by this system are often obscured by noise, which may be attributed to spurious memories characteristic of covariance (inner-product) learning rules [1], [27]. Time delays have been used in Hopfield networks [27], [28] to generate time-dependent sequences

Manuscript received January 6, 1997; revised April 10, 1998. This work was supported by the Australian Research Council and Deakin University.

The author was with the School of Computing and Mathematics, Deakin University, Victoria 3168, Australia. He is now with the School of Electrical and Electronic Engineering, Nanyang Technological University, Singapore 639798 (e-mail: elpwang@ntu.edu.sg).

Publisher Item Identifier S 1083-4419(99)00900-0.

of spatial patterns [30], [41], [42] and to process speech signals [45]. Kosko [32] proposed the bidirectional associative memory that consists of interconnected networks and is able to produce spatio-temporal sequences. Buhmann and Schulten [5] used stochastic noise to induce transitions between spatial patterns in Hopfield networks and these transitions formed spatio-temporal sequences. These systems also use Hebb-type learning rules and thus have rather low memory capacity and spurious memories.

Guyon *et al.* [21] presented a spatio-temporal system that required *a priori* analytical expressions of all stored sequences. Other mechanisms for temporal sequence generation are time-dependent [10], [39], asymmetric [8], [36], and diluted higher order [50], [51], [56], [57], synaptic interactions.

Bradski *et al.* [2]–[4] coupled two ART networks [7] to form sequence producing systems (also see [23]). The capabilities of these networks in handling *complex* sequences, where one spatial pattern in a sequence may be followed by different spatial patterns, have not been mathematically proven or clearly substantiated with examples such as the ones given in the present paper. D. Wang and Yuwono [49] recently proposed a novel network to generate complex sequences [47], [48]. They proved that the model can learn to generate any complex sequences within a certain limit determined by the network architecture. Like many other sequence processing systems [59], their network handles sequences of *symbols* (scalars) rather than spatial patterns (vectors). To learn and retrieve spatio-temporal sequences, their network requires preprocessing to transform sequences of spatial patterns to sequences of symbols as network inputs and post-processing to transform sequences of symbols to sequences of spatial patterns as network outputs. This work has the following resemblance with a concurrent effort taken by Nigrin [35]. These networks memorize sequences with the competitive learning algorithm used in the ART networks and anticipate upcoming components in a sequence with feedback connections. They use decreasing activation in neurons to represent order information in a sequence. Nigrin's network *classifies* sequences (also see [7]), rather than generates sequences as the network presented in [49] does. In this paper, we concentrate our discussions to the *storage* and *retrieval* of *spatio-temporal* sequences only. Extension of the present work to sequence classification and discrimination will be the subject of future studies.

Many authors, inspired by Elman [12] and Jordan [29], incorporated time-delayed feedbacks into backpropagation networks [40] to form recurrent networks (for a review see [44]). These networks were used in processing temporal speech signals [33], [46]. DeVries and Principe [9] proposed an interesting temporal model combining the existing approaches such as backpropagation, time-delays, and Grossberg's work. Both backpropagation and recurrent networks have long training times, which makes real-time learning very difficult.

Despite intensive research activities in learning and retrieving spatio-temporal sequences with neural networks, various difficulties, such as slow and inaccurate learning and retrieving, strict orthogonality requirements, limited memory capacity, and difficulties in dealing with complex sequences, remain in the existing approaches. Furthermore, these ex-

isting networks for spatio-temporal sequence generation are all based on some *specific* learning rules and network architecture. In this paper, as an application of MANN's, we propose a *general* framework for learning and generating spatio-temporal sequences with any arbitrary MANN which may be based on any arbitrary architecture and learning algorithm. To demonstrate the applicability of the this general system, we present an implementation using a specific MANN, that is, our multi-associative, dynamically generated variant of the counterpropagation network [25], and we show that this system significantly improves the efficiency of spatio-temporal sequence learning and generation in comparison with other existing systems. The present work also increases the capability of dealing with *complex* sequences when compared to a system that we proposed using AANN's and HANN's, as evidenced by the Theorem in Section III when compared to our previous work based on AANN's and HANN's [53], [54], as evidenced by the Theorem in Section III and the example given in Tables I–III.

This paper is organized as follows. In Section II, we discuss general features of MANN and provide a computationally efficient exemplar MANN based on the counterpropagation network. As an application of MANN's, we describe in Section III the architectural design and the operating mechanisms of our general system for learning and generating spatio-temporal sequences, together with analytical discussions on the maximum complexity of the sequences learned by the system and the memory capacity of the system (the maximum lengths and number of sequences stored). An implementation of the general system, as well as the learning and the retrieval of some explicit examples, will be described in Section IV. We end the paper with some closing remarks in Section V.

II. MULTIASSOCIATIVE NEURAL NETWORKS AND AN EXAMPLE

As defined in the previous section, a multi-associative neural network (MANN) associates one pattern with multiple patterns. There are many situations where multi-associations may occur. For instance, a facial image may be associated with not only the person's name, but also a friend, a humorous person, etc. A car may be associated with a red object, a tool for transport, or even traffic hazard! Although one pattern \vec{C} may be associated with a set of patterns, which we call the *association set* of pattern \vec{C} , we assume that the network outputs only one pattern in its association set at any given instance of time and a signal external to the MANN determines which pattern in the association set is the overall output of the MANN.

We assume that there are two separate input channels in a MANN, i.e., the conditioned stimulus (CS) and the unconditioned stimulus (US) channels, in analogy with classical conditioning [31], that is, after repeated presentations of a US together with a CS, the CS alone can generate the response caused by the US. During learning, paired CS and US input patterns are presented to a MANN through the CS channel and the US channel, respectively, and the MANN learns the association between the CS and the US patterns in each pair. After learning, each CS pattern $\vec{C}^{(k)}$ may be associated with

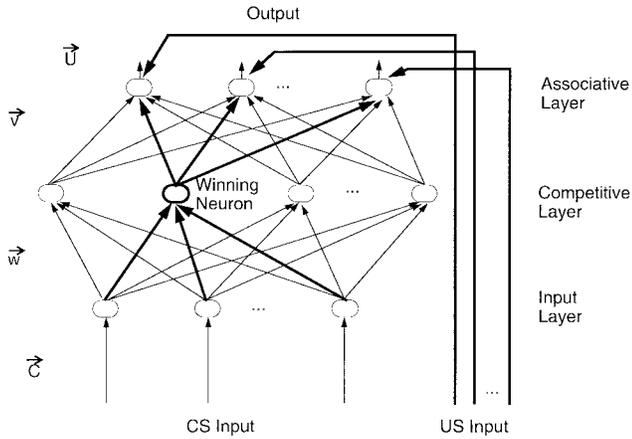


Fig. 1. The forward-only counterpropagation neural network (FOCPN) of Hecht-Nielsen.

multiple US patterns, i.e., the association set

$$\{\vec{U}^{(k,1)}, \vec{U}^{(k,2)}, \dots, \vec{U}^{(k,m_k)}\} \equiv e^{(k)} \quad (1)$$

where the size of the association set m_k can be any positive integer. In a hetero-associative network, one has $m_k = 1$, whereas in an auto-associative network, one has both $m_k = 1$ and $\vec{C}^{(k)} = \vec{U}^{(k)}$, for all k . Note that the terms CS and US are interpreted rather narrowly in the present paper, since many other biological features of classical conditioning are not used here [31].

During retrieving after training, when the MANN is presented with CS pattern $\vec{C}^{(k)}$ which may be obscured with spatial noise, the MANN first outputs $\vec{U}^{(k,1)}$, if the noise is within the tolerance level of the MANN. An external control signal then determines whether the MANN “fixates” the overall output on $\vec{U}^{(k,1)}$, or outputs the next US pattern in association set $e^{(k)}$, until fixation on one of the associated US patterns occurs or the last US patterns in association set $e^{(k)}$ has been reached. This external control signal results from the evaluation of the current output of the MANN, e.g., by another processing system that accepts the output of the MANN as input. We will describe explicitly how the overall output of a MANN is determined in our application of MANN’s to spatio-temporal sequence generation in the subsequent two sections. We assume that the MANN outputs a “don’t know” answer if the noise in the input CS pattern is over the tolerance level. The tolerance level of the MANN is determined according to practical requirements in each specific application.

We now present a computationally efficient exemplar MANN by modifying the forward-only counterpropagation network (FOCPN) invented by Hecht-Nielsen [25]. We first briefly describe the architecture and the learning algorithm of the FOCPN. As shown in Fig. 1, the FOCPN consists of an input layer, a competitive layer, and an associative layer which also serves as the output layer in the original FOCPN. All neurons in the network are standard McCulloch–Pitts binary neurons [34]. When a CS input pattern \vec{C} is presented to the input layer during training, the competitive layer performs winner-take-all competitive learning, and the neuron whose

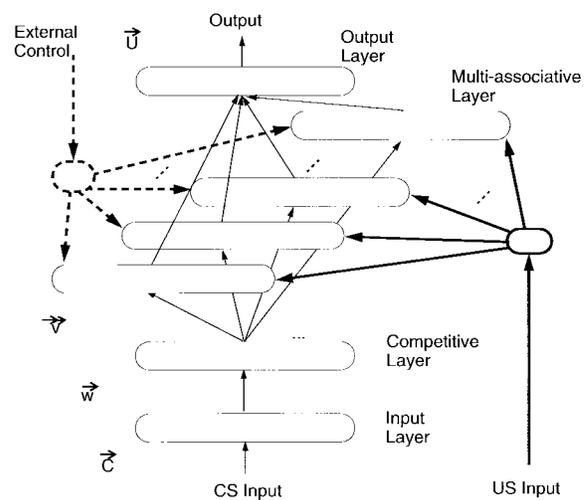


Fig. 2. The present multi-associative, dynamically generated variant of the forward-only counterpropagation neural network (MCPN).

synaptic weights are the most similar to the input pattern changes its incoming weight vector as follows (Fig. 1)

$$\vec{w}_i^{\text{new}} = (1 - \alpha(t))\vec{w}_i^{\text{old}} + \alpha(t)\vec{C}. \quad (2)$$

The incoming weights of other neurons in the competitive layer remain unchanged. All outgoing weights originating from this winning neuron to the associative layer are modified toward the associated US training pattern (the “correct” or the expected output pattern) \vec{U} in a similar fashion

$$\vec{v}_i^{\text{new}} = (1 - \alpha'(t))\vec{v}_i^{\text{old}} + \alpha'(t)\vec{U}. \quad (3)$$

All other weights connected to the associative layer do not change at this time. Hence after training, the incoming weights of each neuron in the competitive layer represent a cluster in the CS training patterns and the outgoing weights originating from this neuron to the associative layer represent the associated US training pattern. Note that \vec{v}_i is *not* the weight vector of the i th associative neuron. Usually the learning rates α and α' assume large values at the beginning of training and gradually decrease toward zero as time elapses. After training, when a testing pattern $\vec{C}^{(k)}$ is presented to the FOCPN through the CS input channel only, the neuron in the competitive layer with weights most similar to the testing pattern wins the competition and broadcasts the associated US pattern $\vec{U}^{(k)}$ to the associative (output) neurons. The original FOCPN is a hetero-associative neural network.

We make the following two modifications in our variant of the FOCPN. First, the competitive layer and all weights of the network are dynamically generated according to the competitive learning algorithm used in the ART network [6]. This dynamically generated FOCPN not only achieves desired training and retrieving objectives, but also is efficient to implement with software [43] and has practically unlimited storage capacity. Second, a multi-associative layer is also dynamically generated according to the rule described below, to allow for multi-associations (Fig. 2).

Suppose the dimensions of the CS and the US patterns are N_{CS} and N_{US} , respectively. Before training, the network

has N_{cs} input neurons and one group of N_{us} associative neurons, but the network has no competitive neurons or weights. When the first pair of CS and US training patterns, i.e., $\vec{C}^{(1)}$ and $\vec{U}^{(1)}$, is presented to the network, the first competitive neuron is generated, with its incoming weights being the CS training pattern, i.e., $\vec{w}_1 = \vec{C}^{(1)}$, and the outgoing weights connecting this neuron to all associative neurons being the US training pattern, i.e., $\vec{v}_1^{(1)} = \vec{U}^{(1)}$. It will become clearer that each competitive neuron may be connected to more than one group of associative neurons to allow for multi-associations. $\vec{v}_k^{(j)}$ denotes the outgoing weight vector connecting the k th competitive neuron with the j th group of associative neurons, and, as shown below, it is actually the j th US pattern associated with the CS pattern represented by the incoming weight vector of the k th competitive neuron.

For each subsequent pair of CS–US association, a competition is carried out in the competitive layer and the neuron whose incoming weights are the most similar to the CS training pattern is located. If the similarity between the CS training pattern and the weight vector of this winning neuron is below a vigilance threshold θ_{us} [6], which signifies that the US training pattern belongs to a category that has not been learned, a new competitive neuron is generated in the same way in which the first competitive neuron was generated, namely, with its incoming and outgoing weight vectors being the CS and the US training vectors, respectively.

If the similarity between the CS training pattern and the incoming weight vector of this winning neuron is above the vigilance threshold θ_{cs} , which signifies that the CS training pattern belongs to a learned category, the incoming weight vector of the winning neuron (\vec{w}) is modified according to (1). The outgoing weight vector of this neuron is then compared with the training US pattern, if the winning neuron is connected to only one set of associative neurons, that is, if the CS pattern represented by the winning neuron has been associated with only one US pattern. If the winning neuron is connected to multiple groups of associative neurons, namely, if the CS pattern represented by the winning neuron has been associated with multiple US patterns, then only the outgoing weight vector that is most similar to the training US pattern is considered. If the similarity between this outgoing weight vector and the training US pattern is over another vigilance threshold θ_{us} , which signifies that the training vector belongs to a learned category, then this outgoing weight vector is updated according to (2). If the similarity between the US training pattern and the most similar outgoing weight vector of the winning neuron is below the vigilance threshold θ_{us} , a new group of N_{us} associative neurons are created and the weight vector connecting these new associative neurons to the winning competitive neuron is set to be the US training pattern, thereby establishing a new multi-association (Fig. 2).

In (2) and (3), we choose a special type of learning rates [52] and [59]

$$\alpha(t) = 1/\tau_i, \quad \alpha'(t) = 1/\tau'_i \quad (4)$$

where $(\tau_i - 1)$ and $(\tau'_i - 1)$ are the numbers of times that the incoming and outgoing weights of neuron i has been modified, respectively, so that the incoming and outgoing

weights connected to neuron i are exactly the *overall* averages of the CS and the US training patterns used to modify the weights of this neuron [52], [59]

$$\vec{w}_i(\tau_i) = \frac{1}{\tau_i} \sum_{\tau'=1}^{\tau_i} \vec{C}(\tau')$$

$$\vec{v}_i^{(j)}(\tau_i)^{(j)} = \frac{1}{\tau_i} \sum_{\tau'=1}^{\tau_i} \vec{U}^{(j)}(\tau').$$

After training, each competitive neuron in the network is connected to at least one group of associative neurons. Some competitive neurons may be connected to multiple groups of associative neurons, if the CS patterns which these competitive neurons represent have been associated with multiple US patterns during training (Fig. 2). The associative neurons in the multi-associative layer of the network are connected to the output layer of the network. When a CS input pattern is presented during retrieving, a winner-take-all competition is carried out in the competitive layer. If the similarity between the input CS pattern and the incoming weights of the winning competitive neuron is above the vigilance threshold θ_{cs} , the US association set represented by the outgoing weights of this winning competitive neuron is activated. An external control signal permits only one group of associative neurons to be active at any instance of time and the US pattern represented by the active group of associative neurons becomes the overall output of the network. If the similarity between the input CS pattern and the incoming weights of the winning competitive neuron is below the vigilance threshold θ_{cs} , the network outputs a “don’t know” answer.

In this section, we have discussed in general an important class of neural networks, i.e., the multi-associative neural networks (MANN’s), and we have built a multi-associative, dynamically generated variant of the counterpropagation network (MCPN) as an example of MANN’s. In the subsequent sections, as an application of MANN’s, we will propose a general system for learning and retrieving spatio-temporal sequences with any MANN, and we will implement this general system with the MCPN.

III. A GENERAL SPATIO-TEMPORAL SYSTEM WITH MULTIASSOCIATIVE NEURAL NETWORKS

Spatio-temporal sequences in the present paper denote time-dependent sequences in which each frame or element at any given time is a *spatial pattern*. Throughout this paper, we treat time as a discrete entity. Fig. 3 shows three examples of such sequences, which we have created to demonstrate the functionality of the system to be proposed: a) $\{ABCDEFGHIABCDEFGHI\dots\}$, b) $\{12345612345\dots\}$, and c) $\{JKLJMNJOPJKLJM\dots\}$. Note that in Fig. 3 pattern I , which is the same as pattern 1, appears in both sequences a) and b), pattern b in sequence a) is somewhat similar to pattern 6 in sequence b), and pattern J appears more than once within sequence c). Hence these sequences are *complex* sequences in which one spatial pattern may be followed by multiple patterns. The cyclic nature of the sequences is not required in order to use our model. We are interested

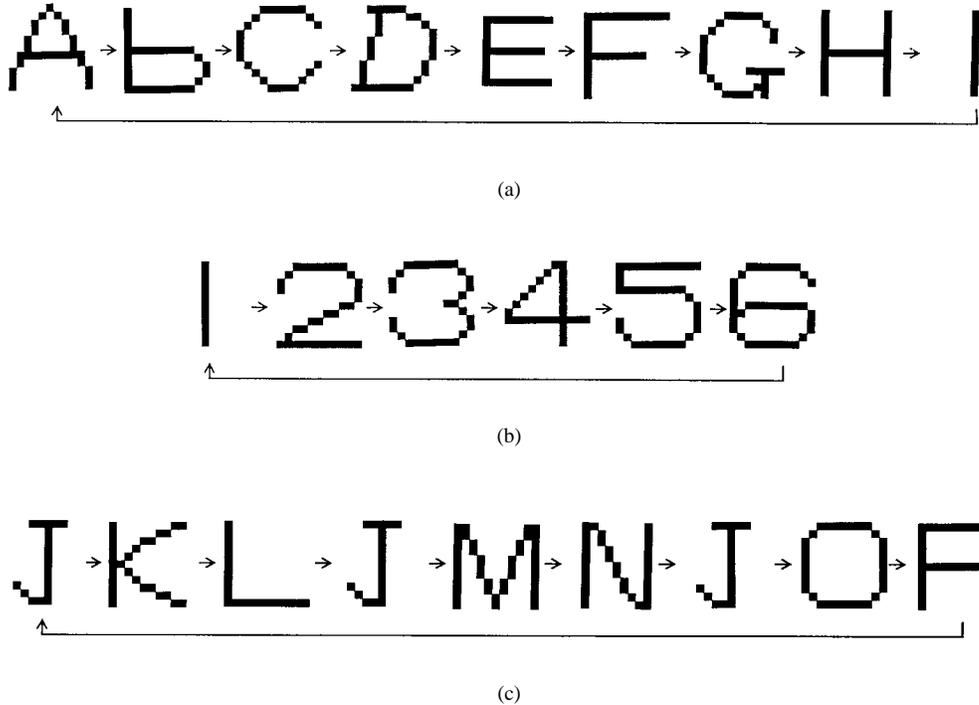


Fig. 3. Three examples of spatio-temporal sequences. Note the complexities in the sequences: (a) pattern *I* appears in both sequence, (b) sequence, and (c) pattern *J* appears more than once in sequence.

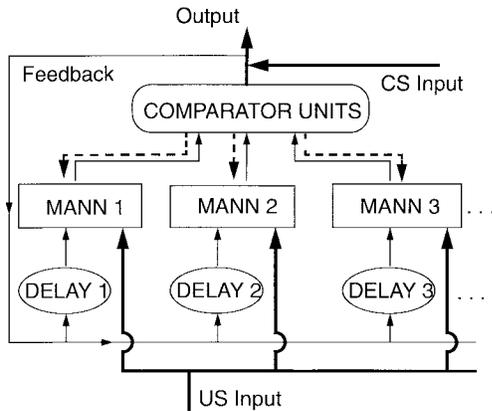


Fig. 4. An application of multi-associative neural networks (MANN's): the general design of the present system for learning and generation of complex spatio-temporal sequences.

in a system for spatio-temporal sequence generation that is able to memorize these sequences after some presentations of these sequences or some variations of these sequences. After learning, the system should be able to recall an entire sequence after being presented with only a small portion of this sequence which may also be obscured by spatial noise and/or temporal gaps.

As we will prove below, the general system shown in Fig. 4 is able to achieve the above objective. The system consists of three major components: a parallel array of N_L MANN's, N comparator units, and time delays that feed the overall output of the system back to the neural network

TABLE I
THE CS AND US INPUTS TO EACH MANN, AS WELL AS THE OVERALL CS AND US INPUTS TO THE SYSTEM, AS A SYSTEM OF THREE EMBEDDED MANN'S LEARNS THE SEQUENCE *REFEREE*

t	SYSTEM		MANN 1		MANN 2		MANN 3	
	CS	US	CS	US	CS	US	CS	US
1	R	R		R		R		R
2	E	E	R	E		E		E
3	F	F	E	F	R	F		F
4	E	E	F	E	E	E	R	E
5	R	R	E	R	F	R	E	R
6	E	E	R	E	E	E	F	E
7	E	E	E	E	R	E	E	E

layer. There are N output neurons and N input neurons in each MANN. The time delay leading to the l th MANN delays the signal by l time steps with respect to the current time, where $l = 1, 2, \dots, N_L$. Output neuron i in each MANN is connected to comparator unit i , which is then connected to input neuron i in each MANN through delayed feedback, where $i = 1, 2, \dots, N$. The system has two separate input channels: the CS and the US channels. This system represents a generalization of those described in [53] and [54] from AANN's and HANN's to MANN's. As we shall show below, this generalization increases the ability to handle *complex* sequences. For example, the systems in [53] and [54] are not able to handle the sequence shown in Table I, whereas the present system is able to learn and retrieve it successfully.

There are two stages of operation for the present spatio-temporal system: the learning stage, at which the system learns spatio-temporal sequences, and the retrieving stage, at which the system retrieves sequences after being presented with cues. Learning and retrieving may be mixed; however, we assume that only one operation occurs at a give instance of time. We will first present some formal discussions on the learning and retrieving mechanisms of the system, which will then be made clearer with some special exemplar sequences and a specific implementation of the general design.

At the learning stage, pairs of sequences of spatial patterns are presented to the CS and the US input channels of the system simultaneously, one spatial pattern at each time step. The two sequences in each training pair may be the same, or one may be a variation of the other. If there is an external input to the CS input channel of the system, the system simply outputs these external signals, regardless the outputs from the comparator units. Since there are always CS inputs to the system during learning, it is equivalent to disabling the comparator units at the learning stage, that is, the comparator units are used only during retrieval.

The CS sequence in each training pair is fed into the CS input channel and is then directed into each embedded MANN through appropriate delayed feedback (Fig. 4). The US sequence in the pair, the expected output of the system corresponding to the CS sequence, is fed into the US input channel and is then directed into each embedded MANN without any delays.

Consider a spatio-temporal sequence of length n , i.e., $(\vec{p}^{(1)}, \vec{p}^{(2)}, \dots, \vec{p}^{(i)}, \dots, \vec{p}^{(n)})$. Suppose the length of the subsequence necessary to unambiguously determine spatial pattern $\vec{p}^{(i)}$ in the sequence is d_i , which is called the degree of pattern $\vec{p}^{(i)}$ [49]. The maximum degree for the entire sequence, i.e., $d \equiv \max\{d_1, d_2, \dots, d_n\}$, is called the degree of this sequence. For example, in the sequence *REFEREE*, the subsequence *ERE* is required to unambiguously determine the last *E*, hence the degree of the last *E* is 3 [49]. One can verify that the degree of this sequence is also 3.

When spatio-temporal sequence $(\vec{p}^{(1)}, \vec{p}^{(2)}, \dots, \vec{p}^{(i)}, \dots, \vec{p}^{(n)})$ is presented simultaneously to the CS input and the US input channels of the system during learning, the overall system CS input and the overall system US input are the same as the US inputs for each MANN at all times. Learning does not occur for the first MANN, i.e., MANN 1, until $t = 2$, since there is no CS input for MANN 1 at $t = 1$ due to the time-delay in the CS pathway of MANN 1 (Fig. 4). Similarly, MANN l does not learn until $t = l + 1$. The CS input pattern for MANN 1 at $t = 2$ is $\vec{p}^{(1)}$, whereas the US input pattern to MANN 1 is $\vec{p}^{(2)}$. Hence MANN 1 learns to associate CS pattern $\vec{p}^{(1)}$ with US pattern $\vec{p}^{(2)}$ at $t = 2$. At $t = 3$, MANN 1 learns to associate CS pattern $\vec{p}^{(2)}$ with US pattern $\vec{p}^{(3)}$ and MANN 2 learns to associate CS pattern $\vec{p}^{(1)}$ with US pattern $\vec{p}^{(3)}$, and so on. Once this sequence is presented to the system, other sequences can be presented to the system in exactly the same way.

At the retrieving stage after learning, a cue sequence, which is a small piece of a stored sequence and which may or may not be obscured by spatial noise and/or temporal gaps, is

presented to the system through the CS input channel only, one spatial pattern at each time step. The comparator units are again disabled during the presentation of the cue sequence, because there exist overall CS inputs to the system. The US input channel of the system is not used during retrieving.

After the last pattern in the cue sequence has been presented, the comparator units are made functional and their task is to find and to output the *common* pattern existed in all of the activated US association sets of the MANN's at each subsequent time step. Suppose the length of the cue sequence, i.e., the number of spatial patterns in the cue sequence excluding any temporal gaps, is c , and at time $t = c + 1$, l_c MANN's have CS input patterns, where $l_c = \min(c, N_L)$ and we recall N_L is the number of embedded MANN's. Suppose MANN l receives a CS input pattern, which has been associated to a set of n_l US patterns $\{\vec{U}^{(1)}, \vec{U}^{(2)}, \dots, \vec{U}^{(n_l)}\} \equiv e^{(l)}$ during training, where $l = 1, 2, \dots, l_c$. If the cue sequence is a part of a training sequence, with each spatial pattern having a tolerable noise level, and the cue sequence is sufficient to unambiguously determine the next spatial pattern \vec{U} in the sequence, then the following must hold: i) pattern \vec{U} must be in every US association set $e^{(l)}$, with $l = 1, 2, \dots, l_c$; and ii) there must not exist any other spatial pattern common to every US association set. This result is not difficult to prove, since i) is a direct consequence of the assumptions that the cue sequence is a part of a training sequence and that each spatial pattern in the cue sequence has a tolerable noise level. In addition, the violation of ii) would contradict the assumption that the cue sequence is sufficiently long to unambiguously determine pattern \vec{U} . We therefore obtain the following theorem concerning the *complexity* of the spatio-temporal sequences that can be learned and generated by the present general system:

Theorem: A system shown in Fig. 4 with N_L embedded MANN's is able to learn and generate any spatio-temporal sequence of degree $d \leq N_L$.

Proof: The proof is similar to the arguments in the previous paragraph. After presenting a sequence to the CS and the US channels simultaneously during training, MANN l learns to associate each spatial pattern in the sequence with the pattern(s) l steps later in the sequence. During retrieving, if a cue sequence with length d or longer is presented to the system, d being the degree of the sequence, the next spatial pattern in the sequence following the last pattern in the cue sequence must be present in all association sets in the MANN's, as long as $d \leq N_L$. Furthermore, this pattern must be the only common pattern present in all association sets in the MANN's, since the cue sequence is sufficiently long to unambiguously determine the next pattern in the sequence. One can infer that the entire sequence can be retrieved by this cue sequence. \square

Note that the mathematical analysis of our system is much less complicated compared to that of Wang and Yuwono [49]. Let us consider the example used by Wang and Yuwono [49], the sequence *REFEREE*; however, we assume that each letter in the sequence is a spatial pattern (vector), rather than a symbol (scalar) as assumed by [49]. Since the degree of the sequence is $d = 3$ as discussed above, we will use 3

TABLE II
THE CS AND US MULTI-ASSOCIATIONS LEARNED BY EACH MANN,
AS A SYSTEM OF THREE EMBEDDED MANN'S LEARNS THE
SEQUENCE *REFEREE*. THE NUMBERS IN "()" SHOW THE TIME
STEPS AT WHICH THE MULTIASSOCIATIONS ARE LEARNED

MANN 1		MANN 2		MANN 3	
CS	US	CS	US	CS	US
R	E(2,6)	R	F(3),E(7)	R	E(4)
E	F(3),R(5),E(7)	E	E(4,6)	E	R(5),E(7)
F	E(4)	F	R(5)	F	E(6)

TABLE III
THE CS INPUT TO AND THE OUTPUT OF EACH MANN, AS WELL AS THE CS
INPUT TO THE OUTPUT OF THE OVERALL SYSTEM, AS A SYSTEM OF
THREE EMBEDDED MANN'S RETRIEVES THE LAST PATTERN *E* IN
SEQUENCE *REFEREE* AFTER IT IS PRESENTED WITH A CUE SEQUENCE *ERE*

t	SYSTEM		MANN 1		MANN 2		MANN 3	
	CS	Output	CS	Output	CS	Output	CS	Output
1		E E						
2	R	R	E	{F,R,E}				
3	E	E	R	E	E	E		
4		E	E	{F,R,E}	R	{F,E}	E	{R,E}

embedded MANN's to learn and generate this sequence, i.e., $N_L = 3$, according to the above theorem. During training, the sequence is presented to the system through the CS and the US input channels simultaneously, one spatial pattern at each time step. The CS and the US inputs to each MANN, as well as the CS and the US inputs to the overall system, are given in Table I for each time step. The multi-associations learned by each MANN are presented in Table II, with the numbers in the parenthesis representing the times at which the multi-associations are learned. After training and during retrieving, a cue sequence *ERE* is presented to the system through the CS input channel only and the US input channels are not used. Table III shows the CS input to and the output of each MANN, as well as the CS input to and the output of the overall system, at each time step of retrieving. From $t = 1$ to $t = 3$, the overall system outputs are the same as the overall system CS inputs, i.e., the comparator units are disabled when there are overall system CS inputs. At $t = 4$, the CS input to MANN 1 is the overall system output at $t = 3$ due to the time-delay, which is pattern *E*. According to Table II, pattern *E* has been multi-associated with a set of three patterns $\{F, R, E\}$ during training. Similarly, the US association sets for MANN2 and MANN3 are $\{F, E\}$ and $\{R, E\}$, respectively. The comparator units find and output the common pattern present in the activated US association sets of all three MANN's. Thus the overall system output at $t = 4$ is pattern *E*, which shows how the system successfully retrieves the last pattern *E* in the sequence *REFEREE* when presented with a cue sequence *ERE* after training.

If an unknown or ambiguous sequence is presented to the system, or a common pattern does not exist in all of the

activated US association sets of the MANN's, the MANN's in the system may output "don't know" answers. A confidence threshold θ_c , which may vary from task to task depending on the acceptable level of conflict or ambiguity, may be applied to the comparator units to allow for an overall "don't know" answer for the present system, so as to reduce the error rate. For example, when $\theta_c = 2/3$, if the most common pattern, which is the pattern appearing in the greatest number of activated US association sets, exists in the activated US association sets of less than $2/3$ of the signal carrying MANN's, or more than one-third of the signal-carrying MANN's in the system output "don't know" answers, then the comparator units also output an overall "don't know" answer and the system stops feeding back. Similar to the "unclassified" result in a classification system, the "don't know" answer in the present system reduces the probability to output meaningless or erroneous sequences, as further shown in a specific implementation presented in the next section, thereby reducing the error rate, i.e., the percentage of sequences retrieved that are incorrect ("unclassified" or "don't know" answers from an artificial system can be examined further, e.g., by human experts, and therefore are not considered as errors).

While retrieving a noisy sequence, the network has two levels of noise tolerance. First, spatial noise in individual patterns in a sequence is tolerated by the embedded MANN's and the quantitative amount of spatial noise tolerated depends factors related to the specific MANN used in the implementation. For example, if the MCPN described in the previous section is chosen, as discussed in detail below, the amount of spatial noise tolerated in each image depends on the vigilance threshold θ_{cs} . Secondly, temporal noise is tolerated at the sequence level by the comparator units and the quantitative amount of temporal noise tolerated is determined by the confidence threshold θ_c for the comparator units. Thus the spatial and temporal noise tolerance can be adjusted according to each specific practical requirement by varying the MANN parameters (e.g., the vigilance threshold) and the comparator confidence threshold, respectively.

Whilst the maximum complexity of the stored spatio-temporal sequences is determined by the number of embedded MANN's, as described in the above theorem, there are no restrictions on the maximum length or the maximum number of the spatio-temporal sequences that can be learned and retrieved by the system, provided that all multi-associations necessary for storing the sequences can be memorized by the embedded MANN's. Thus the memory capacity of the system depends on that of the MANN's used in each specific implementation of the general framework. For example, if the MCPN is used, the theoretic memory capacity of the system is infinite, since the storage size of this MANN is dynamically created on an as-needed basis.

IV. IMPLEMENTATION OF THE GENERAL SYSTEM WITH A MULTIASSOCIATIVE, DYNAMICALLY GENERATED COUNTERPROPAGATION NETWORK

To demonstrate the applicability of the *general* system for spatio-temporal sequence generation proposed in the previous section, we now present the results of an implementation of

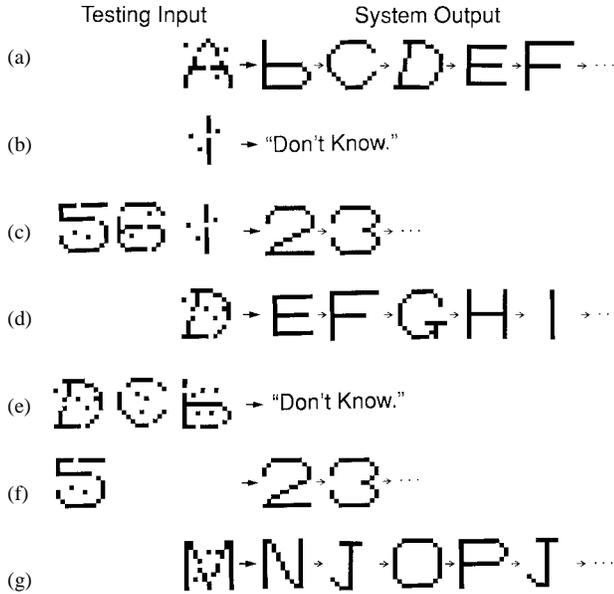


Fig. 5. Spatio-temporal sequence retrieval after training: (a) a retrieval of sequence a); (b) a response to an ambiguous cue; (c) an unambiguous retrieval of sequence b); (d) a response to a nonorthogonal cue pattern; (e) a response to an unknown cue sequence; (f) a response to a cue sequence containing temporal gaps; and (g) a retrieval of sequence c).

TABLE IV
THE CS AND US INPUTS TO EACH MANN, AS WELL AS THE OVERALL CS AND US INPUTS TO THE SYSTEM, AS A SYSTEM OF THREE EMBEDDED MANN'S LEARNS THE SEQUENCE a) IN FIG. 1

t	SYSTEM		MANN 1		MANN 2		MANN 3	
	CS	US	CS	US	CS	US	CS	US
1	A	A	A	A	A	A	A	A
2	b	b	A	b	b	b	b	b
3	C	C	b	C	A	C	C	C
4	D	D	C	D	b	D	A	D
5	E	E	D	E	C	E	b	E
6	F	F	E	F	D	F	C	F
7	G	G	F	G	E	G	D	G
8	H	H	G	H	F	H	E	H
9	I	I	H	I	G	I	F	I
10	A	A	I	A	H	A	G	A
11	b	b	A	b	I	b	H	b
12	C	C	b	C	A	C	I	C
...

the general system with a specific MANN, that is, the MCPN described in Section II.

We choose both dimensions of the CS and the US spatial patterns, as well as the numbers of input neurons and initial associative neurons in the MCPN, to be 11×11 , i.e., $N_{cs} = N_{us} = 121$. Three such MCPN's are used in this implementation ($N_L = 3$). We train the system to store the sequences consisting of manually created spatial images of alpha-numeral characters shown in Fig. 3, and then test the system in situations shown in Fig. 5. We present each training sequence *only once* to the system. The detailed inputs and outputs at each time step of training or retrieval for cases shown in Fig. 5(a)–(c) are given in Tables IV–IX. Compared

TABLE V
THE CS AND US INPUTS TO EACH MANN, AS WELL AS THE OVERALL CS AND US INPUTS TO THE SYSTEM, AS A SYSTEM OF THREE EMBEDDED MANN'S LEARNS THE SEQUENCE b) IN FIG. 1

t	SYSTEM		MANN 1		MANN 2		MANN 3	
	CS	US	CS	US	CS	US	CS	US
1	1	1	1	1	1	1	1	1
2	2	2	1	2	2	2	2	2
3	3	3	2	3	1	3	3	3
4	4	4	3	4	2	4	1	4
5	5	5	4	5	3	5	2	5
6	6	6	5	6	4	6	3	6
7	1	1	6	1	5	1	4	1
8	2	2	1	2	6	2	5	2
9	3	3	2	3	1	3	6	3
...

TABLE VI
THE CS AND US INPUTS TO EACH MANN, AS WELL AS THE OVERALL CS AND US INPUTS TO THE SYSTEM, AS A SYSTEM OF THREE EMBEDDED MANN'S LEARNS THE SEQUENCE c) IN FIG. 1

t	SYSTEM		MANN 1		MANN 2		MANN 3	
	CS	US	CS	US	CS	US	CS	US
1	J	J	J	J	J	J	J	J
2	K	K	J	K	K	K	K	K
3	L	L	K	L	J	L	L	L
4	J	J	L	J	K	J	J	J
5	M	M	J	M	L	M	K	M
6	N	N	M	N	J	N	L	N
7	J	J	N	J	M	J	J	J
8	O	O	J	O	N	O	M	O
9	P	P	O	P	J	P	N	P
10	J	J	P	J	O	J	J	J
11	K	K	J	K	J	K	O	K
12	L	L	K	L	K	L	P	L
...

TABLE VII
TESTING THE SYSTEM FOR CASE a) IN FIG. 3: THE CS INPUT TO AND THE OUTPUT OF EACH MANN, AS WELL AS THE CS INPUT TO AND THE OUTPUT OF THE OVERALL SYSTEM

t	SYSTEM		MANN 1		MANN 2		MANN 3	
	CS	Output	CS	Output	CS	Output	CS	Output
1	\bar{A}	\bar{A}						
2		b	\bar{A}	b				
3		C	b	C	\bar{A}	C		
4		D	C	D	b	D	\bar{A}	D
...

to Fukushima's results [13], our system is capable of fast and accurate storing and generating complex sequences consisting of nonorthogonal spatial patterns. Our system can also output "don't know" answers, thereby reducing error rate.

Sequences a) and b) were first used by Fukushima [13] in training and testing his spatio-temporal system. We present here a comparison between our results and Fukushima's

TABLE VIII
TESTING THE SYSTEM FOR CASE b) IN FIG. 3

t	SYSTEM		MANN 1		MANN 2		MANN 3	
	CS	Output	CS	Output	CS	Output	CS	Output
1	$\bar{1}$	$\bar{1}$						
2		{2,A}	$\bar{1}$	{2,A}				
3		Don't know	{2,A}	Don't know	$\bar{1}$	{3,b}		

TABLE IX
TESTING THE SYSTEM FOR CASE c) IN FIG. 3

t	SYSTEM		MANN 1		MANN 2		MANN 3	
	CS	Output	CS	Output	CS	Output	CS	Output
1	$\bar{5}$	$\bar{5}$						
2	$\bar{6}$	$\bar{6}$	$\bar{5}$	6				
3	$\bar{1}$	$\bar{1}$	$\bar{6}$	1	$\bar{5}$	1		
4		2	$\bar{1}$	{2,A}	$\bar{6}$	2	$\bar{5}$	2
5		3	2	3	$\bar{1}$	{3,b}	$\bar{6}$	3
6		4	3	4	2	4	1	{4,C}
7		5	4	5	3	5	2	5
...

results. In Fig. 5(a), all retrieved images in sequence a) are noise-free, whereas some retrieved images, i.e., E and F , are imperfect for Fukushima's system. Since pattern 1, which is the same as pattern I , appears in both sequences a) and b), the input given in Fig. 5(b) is insufficient to unambiguously retrieve a sequence and additional information is required [Fig. 5(c)]. The ability of outputting a "don't know" answer often can significantly reduce error rate in practical applications [Fig. 5(b)], whereas Fukushima's system outputs meaningless sequences in this kind of situation. When pattern D is presented to Fukushima's system, the retrieval of sequence a) is very difficult: it takes many iterations and many retrieved images are imperfect. This is because Fukushima used a Hebbian-type inner-product learning rule which imposes strict orthogonality requirement on all spatial patterns in stored sequences. The present system retrieves sequence a) accurately and quickly, i.e., retrieved images are free of noise in the first iteration [Fig. 5(d)]. The present system thus has less restriction on training sequences and responds faster in retrieving. Fukushima's system has not been tested with an unknown sequence [Fig. 5(e)], but can be expected to yield meaningless output since it is unable to give a "don't know" answer. The cases with temporal gaps [Fig. 5(f)] and complex sequences [Fig. 5(g)] have not been tested in Fukushima's system.

V. CONCLUSION

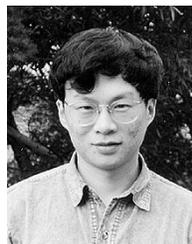
In summary, we have discussed in general an important class of neural network, that is, the *multi-associative neural networks* (MANN's). We have provided a multi-associative, dynamically generated counterpropagation network (MCPN) as a computationally efficient example of this type of neural

network. As an application of the MANN's, we have designed a general system that can learn and generate spatio-temporal sequences with any MANN. After learning, the system is able to retrieve an entire stored spatio-temporal sequence after being presented with a small piece, which may or may not be obscured by spatial noise and may or may not contain temporal gaps. Or equivalently, after learning and when a sequence of events is presented to the system, the system *predicts* the sequence of events in the future. We have mathematically proven that a general system of N_L embedded MANN's is capable of learning and retrieving any spatio-temporal sequence of a complexity no greater than N_L . To demonstrate the applicability of the general system for spatio-temporal sequence generation, we have presented an implementation using the MCPN. This system shows a number of desirable properties, such as short learning time (only one epoch is required per sequence), fast and accurate retrievals, and ability to store a large number of complex sequences consisting of nonorthogonal spatial patterns. These computational advantages signify a marked improvement over existing approaches to spatio-temporal sequence generation and are important for practical applications such as real time robotic control and speech production. Topics of future work include explorations of new types of MANN's other than the MCPN, new applications of MANN's in addition to spatio-temporal sequence generation, implementations of the present general design for spatio-temporal sequence generation with new MANN's, and practical applications of all implementations of the present spatio-temporal system.

REFERENCES

- [1] S.-I. Amari, "Learning patterns and pattern sequences by self-organizing nets of threshold elements," *IEEE Trans. Comput.*, vol. C-21, p. 1197, 1972.
- [2] G. Bradski, G. A. Carpenter, and S. Grossberg, "Working memories for learning temporal order with application to three-dimensional visual object recognition," *Neural Computat.*, vol. 4, pp. 270-286, 1992.
- [3] ———, "STORE working memory networks for storage and recall of arbitrary temporal sequences," *Biol. Cybern.*, vol. 71, no. 6, pp. 469-480, 1994.
- [4] G. Bradski and S. Grossberg, "Fast-learning VIEWNET architectures for recognizing three-dimensional objects from multiple two-dimensional views," *Neural Networks*, vol. 8, no. 7/8, pp. 1053-1080, 1996.
- [5] J. Buhmann and K. Schulten, "Noise-driven temporal association in neural networks," *Europhys. Lett.*, vol. 4, pp. 1205-1209, 1987.
- [6] G. A. Carpenter and S. Grossberg, "Neural dynamics of category learning and recognition: Attention, memory consolidation, and amnesia," in *Brain Structure, Learning, and Memory*, J. Davis, R. Newburgh, and E. Wegman, Eds., AAAS Symp. Series, 1986, pp. 233-290.
- [7] G. A. Carpenter, S. Grossberg, and J. H. Reynolds, "ARTMAP: Supervised real-time learning and classification of nonstationary data by a self-organizing neural network," *Neural Networks*, vol. 4, no. 5, pp. 565-588, 1991.
- [8] A. C. C. Coolen and T. W. Ruijgrok, "Image evolution in Hopfield networks," *Phys. Rev. A*, vol. 38, pp. 4253-4255, 1988.
- [9] B. De Vries and J. C. Principe, "The gamma model—A new neural model for temporal processing," *Neural Networks*, vol. 5, pp. 565-576, 1992.
- [10] S. Dehaene, J. P. Changeux, J. P. Nadal, "Neural networks that learn temporal sequences by selection," in *Proc. Nat. Acad. Sci.*, 1987, vol. 84, pp. 2727-2731.
- [11] K. Doya and S. Yoshizawa, "Adaptive neural oscillator using continuous-time backpropagation learning," *Neural Networks*, vol. 2, pp. 375-385, 1989.
- [12] J. L. Elman, "Finding structure in time," *Cogn. Sci.*, vol. 14, pp. 179-211, 1990.

- [13] K. Fukushima, "A model of associative memory in the brain," *Kybern.*, vol. 12, pp. 58–63, 1973.
- [14] S. Grossberg, "On learning of spatiotemporal patterns by networks with ordered sensory and motor components, I: Excitatory components of the cerebellum," *Stud. Appl. Math.*, vol. 48, pp. 105–132, 1969.
- [15] ———, "Nonlinear difference-differential equations in prediction and learning theory," in *Proc. Nat. Acad. Sci.*, 1967, vol. 58, pp. 1329–1334.
- [16] ———, "Some networks that can learn, remember, and reproduce any number of complicated space-time patterns," *Stud. Appl. Math.*, vol. 49, pp. 135–166, 1970.
- [17] ———, "Neural pattern discrimination," *J. Theor. Biol.*, vol. 27, pp. 291–337, 1970.
- [18] ———, *Studies of Mind and Brain: Neural Principles of Learning, Perception, Development, Cognition, and Motor Control*. Boston, MA: Reidel, 1982.
- [19] ———, "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. New York: Academic, vol. 1, pp. 187–294, 1986.
- [20] ———, *The Adaptive Brain, II: Vision, Speech, Language, and Motor Control*. Amsterdam, The Netherlands: Elsevier/North-Holland, 1987.
- [21] I. Guyon, L. Personnaz, J. P. Nadal, and G. Dreyfus, "Storage and retrieval of complex sequences in neural networks," *Phys. Rev. A*, vol. 38, pp. 6365–6372, 1988.
- [22] M. Hagiwara, "Multidirectional associative memory," in *Proc. Int. Joint Conf. Neural Networks*, 1990, vol. 1, pp. 3–6.
- [23] M. J. Healy, T. P. Caudell, and S. D. G. Smith, "A neural architecture for pattern sequence verification through inferencing," *IEEE Trans. Neural Networks*, vol. 4, pp. 9–20, Jan. 1993.
- [24] D. O. Hebb, *The Organization of Behavior*. New York: Wiley, 1949, p. 44.
- [25] R. Hecht-Nielsen, *Neurocomputing*. Reading, MA: Addison-Wesley, 1990.
- [26] Y. Hirai, "A model of human associative processor (HASP)," *IEEE Trans. Syst., Man, Cybern.*, vol. SMC-13, pp. 851–857, Sept./Oct. 1983.
- [27] J. J. Hopfield, "Neural networks and physical systems with emergent collective computational abilities," in *Proc. Nat. Acad. Sci. USA*, Apr. 1982, vol. 79, pp. 2554–2558.
- [28] ———, "Neurons with graded response have collective computational properties like those of two-state neurons," in *Proc. Nat. Acad. Sci.*, May 1984, vol. 81, pp. 3088–3092.
- [29] M. I. Jordan, "Attractor dynamics and parallelism in a connectionist sequential machine," in *Proc. 8th Annu. Conf. Cognit. Sci. Soc.* Hillsdale, NJ: Lawrence Erlbaum, 1986, pp. 531–546.
- [30] D. Kleinfeld, "Sequential state generation by model neural networks," in *Proc. Nat. Acad. Sci.*, 1986, vol. 83, pp. 9469–9473.
- [31] A. H. Klopff, "A neuronal model of classical conditioning," *Psychol. Rev.*, vol. 88, pp. 135–170, 1988.
- [32] B. Kosko, "Bidirectional associative memories," *IEEE Trans. Syst., Man, Cybern.*, vol. 18, pp. 49–60, Jan./Feb. 1988.
- [33] R. P. Lippmann, "Review of neural networks for speech recognition," *Neural Computat.*, vol. 1, pp. 1–38, 1989.
- [34] W. S. McCulloch and W. H. Pitts, "A logical calculus of ideas immanent in nervous activity," *Bull. Math. Biophys.*, vol. 5, pp. 115–133, 1943.
- [35] A. Nigrin, *Neural Networks for Pattern Recognition*. Cambridge, MA: MIT Press, 1993.
- [36] H. Nishimori, T. Nakamura, and M. Shiino, "Retrieval of spatio-temporal sequence in asynchronous neural network," *Phys. Rev. A*, vol. 41, pp. 3346–3354, 1990.
- [37] Y. Osana, M. Hattori, and M. Hagiwara, "Chaotic bidirectional associative memory," in *Proc. IEEE Int. Conf. Neural Networks*, 1996, pp. 816–821.
- [38] F. J. Owens, G. H. Zheng, and D. A. Irvine, "A multi-output-layer perceptron," *Neural Comput. Appl.*, vol. 4, pp. 10–20, 1996.
- [39] P. Peretto and J. J. Niez, "Collective properties of neural networks," in *Disordered Systems and Biological Organization*, E. Bienenstock *et al.*, Eds. Berlin, Germany: Springer-Verlag, 1986, pp. 171–185.
- [40] D. E. Rumelhart, J. L. McClelland, and the PDP Research Group, *Parallel Distributed Processing*. Cambridge, MA: MIT Press, 1986.
- [41] H. Sompolinsky and I. Kanter, "Temporal association in asymmetric neural networks," *Phys. Rev. Lett.*, vol. 57, pp. 2861–2864, 1986.
- [42] D. W. Tank and J. J. Hopfield, "Neural computation by concentrating information in time," in *Proc. Nat. Acad. Sci.*, 1987, vol. 84, pp. 1896–1900.
- [43] G. L. Tarr, S. K. Rogers, M. Kabrisky, M. E. Oxley, and K. L. Priddy, "Effective neural network modeling in C," in *Artificial Neural Networks*, T. Kohonen, K. Mäksä, O. Simula, and J. Kangas, Eds. Amsterdam, The Netherlands: Elsevier, 1991, pp. 1497–1500.
- [44] A. C. Tsoi and A. D. Back, "Locally recurrent globally feedforward networks: a critical review of architectures," *IEEE Trans. Neural Networks*, vol. 5, pp. 229–239, Mar. 1994.
- [45] 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 Processing*, vol. 39, pp. 698–713, 1991.
- [46] A. Waibel, T. Hanazawa, G. Hinton, and K. Shikano, "Phoneme recognition using time-delay neural networks," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. 37, pp. 328–339, 1989.
- [47] D. L. Wang and M. A. Arbib, "Complex temporal sequence learning based on short-term memory," *Proc. IEEE*, vol. 78, pp. 1536–1543, Sept. 1990.
- [48] ———, "Timing and chunking in processing temporal order," *IEEE Trans. Syst., Man, Cybern.*, vol. 23, pp. 993–1009, July/Aug. 1993.
- [49] D. L. Wang and B. Yuwono, "Anticipation-based temporal pattern generation," *IEEE Trans. Syst., Man, Cybern.*, vol. 25, pp. 615–628, Apr. 1995.
- [50] L. Wang, "Oscillatory and chaotic dynamics in neural networks under varying operating conditions," *IEEE Trans. Neural Networks*, vol. 7, pp. 1382–1388, Nov. 1996.
- [51] ———, "Suppressing chaos with hysteresis in a higher-order neural network," *IEEE Trans. Circuits Syst.*, vol. 43, pp. 845–846, Dec. 1996.
- [52] ———, "On competitive learning," *IEEE Trans. Neural Networks*, vol. 8, pp. 1214–1217, Sept. 1997.
- [53] ———, "Learning and retrieving spatio-temporal sequences with any static associative neural network," *IEEE Trans. Circuits Syst. II*, vol. 45, pp. 729–738, June 1998.
- [54] L. Wang and D. L. Alkon, "An artificial neural network system for temporal-spatial sequence processing," *Pattern Recognit.*, vol. 28, no. 8, pp. 1267–1276, 1995.
- [55] L. Wang, E. E. Pichler, and J. Ross, "Oscillations and chaos in neural networks: an exactly solvable model," in *Proc. Nat. Acad. Sci.*, Dec. 1990, vol. 87, pp. 9467–9471.
- [56] L. Wang and J. Ross, "Chaos, multiplicity, crisis, and synchronicity in higher order neural networks," *Phys. Rev. A*, vol. 44, pp. R2259–2262, Aug. 1991.
- [57] ———, "Physical modeling of neural networks," in *Methods in Neuroscience, Computers and Computations in the Neurosciences*, P. M. Conn, Ed. San Diego, CA: Academic, 1992, vol. 10, pp. 549–567.
- [58] *Time Series Prediction: Forecasting the Future and Understanding the Past*, A. S. Weigend and N. A. Gershenfeld, Eds. Reading, MA: Addison-Wesley, 1993.
- [59] L. Wu and F. Fallside, "On the design of connectionist vector quantizers," *Comput. Speech Lang.*, vol. 5, pp. 207–229, 1991.



Lipo Wang received the B.S. degree in laser technology from National University of Defense Technology, Changsha, China, in 1983, and the Ph.D. degree in physics from Louisiana State University, Baton Rouge, in 1988 (with a CUSPEA assistantship). His Ph.D. dissertation was on low-dimensional electron systems in semiconductor devices such as MOSFET.

Since then, he has been interested in the theory and applications of neural networks. In 1989, he worked at Stanford University, Stanford, CA, as a Postdoctoral Fellow. In 1990, he was a Lecturer in electrical engineering at the University College, University of New South Wales, Australia. From 1991 to 1994, he was on the Research Staff at the National Institute of Health, Bethesda, MD. From 1995 to August 1998, he was a Computing Lecturer at the School of Computing and Mathematics, Deakin University, Melbourne, Australia. Since September 1998, he has been an Associate Professor at the School of Electrical and Electronic Engineering, Nanyang Technological University, Singapore. His research interests include combinatorial optimization, control, temporal behavior, pattern recognition, data mining, fuzzy networks, and biological networks. He is author or co-author of over 40 journal publications and 30 conference presentations, holds a U.S. patent on neural networks, and is co-author/editor of *Artificial Neural Networks: Oscillations, Chaos, and Sequence Processing* (Los Alamitos, CA: IEEE Comput. Soc. Press, 1993) and *Proceedings of the 1998 World Multiconference on Systemics, Cybernetics and Informatics*. He is an Associate Editor for *Knowledge and Information Systems: An International Journal*, and has served on the program committees of a number of conferences. More information is available at his home page at <http://www.ntu.edu.sg/home/elpwang/>.