A Neural Model of Episodic and Semantic Spatiotemporal Memory

Gerard J. Rinkus (rinkus@comcast.net)
468 Waltham St.
Newton, MA USA

Abstract

A neural network model is proposed that forms sparse spatiotemporal memory traces of spatiotemporal events given single occurrences of the events. The traces are distributed in that each individual cell and synapse participates in numerous traces. This sharing of representational substrate provides the basis for similarity-based generalization and thus semantic memory. Simulation results are provided demonstrating that similar spatiotemporal patterns map to similar traces. The model achieves this property by measuring the degree of match, *G*, between the current input pattern on each time slice and the expected input given the preceding time slices (i.e., temporal context) and then adding an amount of noise, inversely proportional to *G*, to the process of choosing the internal representation for the current time slice. Thus, if *G* is small, indicating novelty, we add much noise and the resulting internal representation of the current input pattern has low overlap with any preexisting representations of time slices. If *G* is large, indicating a familiar event, we add very little noise resulting in reactivation of all or most of the preexisting representation of the input pattern.

Introduction

Any realistic cognitive model must exhibit both episodic and semantic memory. And, as emphasized by Ans, Rousset, French, & Musca (2002), it must demonstrate these properties for the spatiotemporal (or, sequential) pattern domain. Thus, the model must be able to recall, without significant interference, large numbers of spatiotemporal patterns, which we will call episodes, given only single presentations of those episodes. Furthermore, it must exhibit human-like similarity-based generalization and categorization properties that underlie many of those phenomena classed as semantic memory.

We propose a sparse, distributed neural network model, TESMECOR (Temporal Episodic and Semantic Memory using Combinatorial Representations), that performs single-trial learning of episodes. The degree of overlap between its distributed memory traces increases with the similarity of the episodes that they represent. This latter property provides a basis for generalization and categorization and thus, semantic memory. The model achieves this property by computing, on each time slice, the similarity, *G*, between the expected and actual input patterns and then adding an amount of noise inversely proportional to *G* into the process of choosing an internal representation (IR) for that time slice. When expected and actual inputs match completely, no noise is added, allowing those IR cells having maximal input via previously modified weights to be reactivated (i.e., fully deterministic recall). When they completely mismatch, enough noise is added to completely drown out the learned, deterministic inputs, resulting in activation of an IR having little overlap with preexisting traces.

The opposing purposes of episodic memory and pattern recognition (i.e., semantic memory)—i.e., remembering what is unique about individual instances vs. learning the similarities between instances—has led other researchers to propose that the brain uses two complementary systems. McClelland et al (1995) and O’Reilly & Rudy (1999) propose that the purpose of the hippocampus is to rapidly learn new specific information whereas the purpose of neocortex is to slowly integrate information across individual instances thus coming to reflect the higher-order statistics of the environment. The hippocampus then repeatedly presents its newly acquired memory traces to neocortex, acting as trainer facilitating the gradual transfer of information to neocortex during the period of memory consolidation. We point out that TESMECOR is not such a two-component model. Rather, it is a monolithic model, i.e., it has a single local circuit architecture and processing algorithm (envisioned as an analog of the cortical mini-column) that satisfies the opposing needs.

Episodic Spatiotemporal Memory

Rinkus (1995) introduced a neural network model, TEMECOR, of episodic memory for spatiotemporal patterns. As shown in Figure 1, the model’s. Layer 1 (L1) consists of binary feature detectors and its layer 2 (L2) consists of competitive modules (CMs). The L2 cells are nearly completely connected via a horizontal matrix (H-matrix) of binary weights.

The model operates in the following way. On each time step, a pattern is presented to L1. On that same time step, one L2 cell is chosen at random to become active in each CM corresponding to an active L1 cell. In addition, the horizontal weights from the L2 cells active on the prior time slice to those that become active on the current time are increased to their maximal value of one. In this way, spatiotemporal memory traces are embedded in the H-matrix. Later on, if we reinstate a set of L2 cells that was coactive in the past while learning an episode, the remainder of that episode will be read out in time. That is, the model recalls spatiotemporal memories.

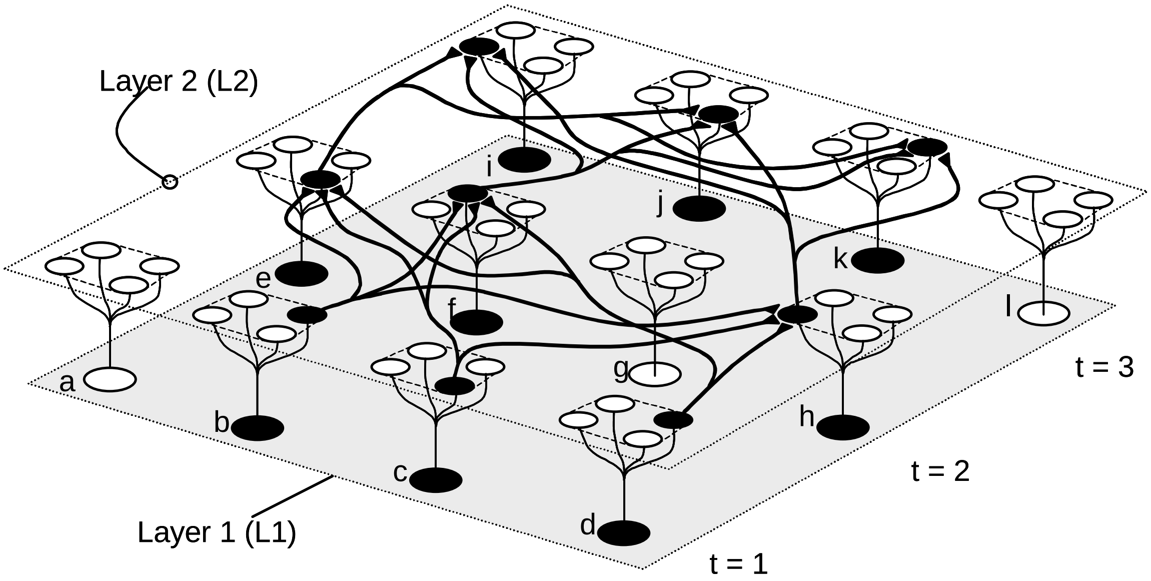

Figure 1: TEMECOR architecture showing how spatiotemporal memory traces are laid down amongst the horizontal connections of Layer 2. Features {b,c,d} are active at t = 1, {e,f,h} at t = 2, and {i,j,k} at t = 3. Each L2 cell has horizontal connections to all other L2 cells except those in its own CM. Only the connections increased while processing this particular spatiotemporal pattern (episode) are shown. Note that although this figure shows each time slice of the episode being handled by a separate portion of the network, this is purely to keep the figure uncluttered. In fact, all L1 cells and all L2 CMs are eligible to become active on every time slice.

TEMECOR exhibits high capacity, as shown in Figure 2, as well as other essential properties of episodic memory, e.g., single-trial learning. The model's beneficial properties derive principally from its use of a sparse distributed, or *combinatorial*, representational framework, a framework underlying many other models—Willshaw, Buneman & Longuet-Higgins, 1969; Lynch, 1986; Palm, 1980; Moll & Miikkulainen, 1995; Coultrip & Granger, 1994. The key to its high capacity is that by randomly choosing winners in the CMs, it minimizes the average overlap amongst the memory traces.

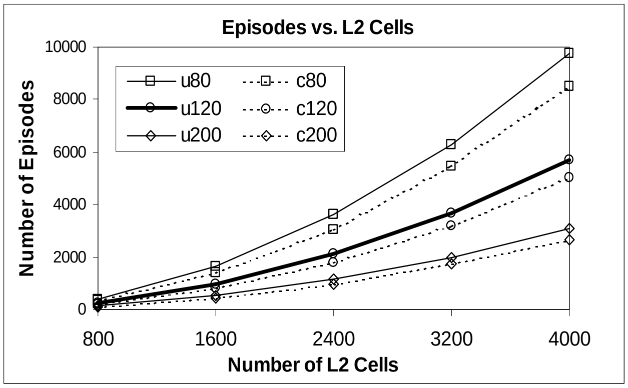

Figure 2: Capacity Results

Table 1 provides the data for the bold curve in the figure. It gives the maximal capacity, E, and other statistics for networks of increasing size, L. All episodes had T = 6 time slices and each time slice had S = 20 (out of M = 100) active features, chosen at random. The bottom row of the table shows that a network containing 4000 L2 cells, i.e., 100 CMs having K = 40 cells each, can store 5693 such episodes.

Table 1: Capacity Test Results

E	E/L	F	K	L	V	R_{set}	H
237	0.30	285	8	800	36	96.3	52.3
943	0.59	1132	16	1600	71	97.0	52.1
2104	0.88	2524	24	2400	105	97.0	51.8
3691	1.15	4430	32	3200	138	97.2	51.4
5693	1.42	6831	40	4000	171	97.4	50.9

Table 1 was generated as follows. For each K, the maximal number of episodes, E, which could be stored to criterion average recall accuracy, 96.3%, was determined. Recall accuracy, R_e, for a given episode *e*, is defined as:

$$R_e = (C_e - D_e) \big/ (C_e + I_e) \qquad (1)$$

where C_e is the number of L2 cells correctly active during recall of e^{th} episode, D_e is the number of deleted L2 cells, and I_e is the number of intruding L2 cells. The table reports R_{set}, the average of the R_e values for a whole set of episodes. All episodes were presented only once.

The other statistics in Table 1 are as follows. E/L is the ratio of stored episodes to the number of cells in L2, which increases linearly. F is the average number of instances of each feature across the entire set of episodes. V is the average number of times each L2 cell in a given CM became active to represent the corresponding feature. H is the percentage of weights increased, which is nearly constant, at just over 50%, across rows. As we allow the fraction of weights to increase beyond 50%, more episodes are stored, but with a lower average recall accuracy due to the increase in intrusion errors resulting from saturation of the weights.

While TEMECOR exhibited the major properties of episodic memory it was not initially intended to model semantic memory and, due to its completely random method of choosing sparse internal representations at L2, it did not exhibit the generalization and categorization properties that underlie semantic memory. The successor version of the model, TESMECOR was developed to address this shortcoming (Rinkus, 1996).

Semantic Spatiotemporal Memory

TESMECOR is shown in Figure 3. It has some architectural differences with the original version (essentially, relaxations of some of the original's structural constraints) and a greatly modified winner selection process. The H-matrix of L2 is as before but the vertical projection is generalized. There is no longer a 1-to-1 correspondence between L1 cells and L2 CMs. Rather; each L1 cell connects to a fraction of all the L2 cells chosen at random in simulations. In TESMECOR, all CMs are active on every time slice. In addition, the bottom-up, or forward, connections (F-weights) and the top-down, or reverse, connections (R-weights) are now modeled separately and are modifiable.

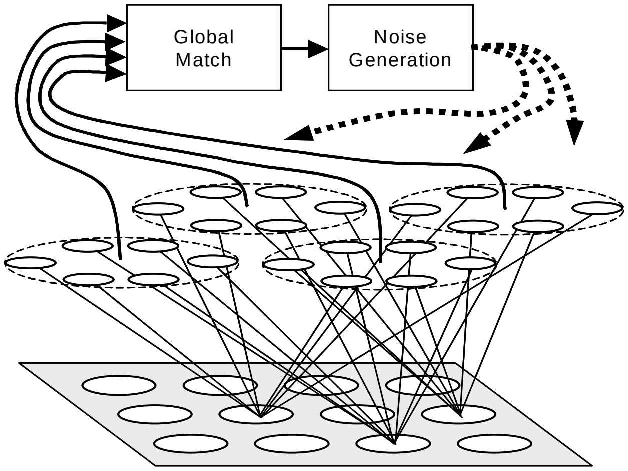

Figure 3: TESMECOR architecture.

The most significant change between TEMECOR and TESMECOR however is in the processing algorithm. Specifically, TESMECOR adds circuitry implementing spatiotemporal matching operations, both locally within each CM and globally over the entire L2. On each time slice, the global degree of match between the actual current input and the expected input, given the spatiotemporal context of the current input, modulates the amount of noise injected into the process of selecting which L2 cells will become active. The smaller the match, the more noise that is added and the greater the difference between the internal representation (IR) that would have become active purely on the basis of the deterministic inputs reflecting prior learning and the IR that actually does become active. The greater the match, the less noise added and the smaller the difference between the most highly implicated IR (on the basis of prior learning) and the actually chosen IR.

Figure 4 illustrates the basic principles by which the model computes, on each time slice, the degree of match, *G*, between its expected input and the actual input and then uses *G* to determine how much noise to add to the internal representation selection scheme. Figure 4a shows a pattern, A, presenting at t = 1. The H-weights are increased (represented by the dotted lines) from the active L1 cells onto an internal representation, IR_A, comprised of the three L2 cells that emerge as winners in their respective CMs. For purposes of this example, these three winners can be assumed to be chosen at random.

Figure 4b shows another pattern, B, presenting at t = 2. As with IR_A, IR_B can be assumed to be chosen at random. Here, we see the both H- and F-weights being increased.

Figure 4c shows another trial with pattern A presenting at t = 1. This time, IR_A becomes active due to the deterministic effects of the previously increased weights (which are now shown as solid lines). The cells of IR_A now send out signals via the H-matrix which will arrive at the other CMs at t = 2.

At this point, it is convenient to portray the t = 2 time slice in two steps. Figures 4d and 4e show these two steps. Figure 4d shows the signals arriving via the H-matrix at the same time that that signals arrive via the F-matrix from currently active L1 cells. Thus, the L2 cells in the three CMs on the right simultaneously receive two vectors each carrying possibly different expectations about which IR should become active (or equivalently, different hypotheses about what the current state of the world is). It is these two vectors that TESMECOR compares. In this particular case, the three cells of IR_B are receiving full support via the H-matrix. In other words, the temporal context says that IR_B should become active. However, these cells are receiving only partial support (two out of four L1 cells) via the F-matrix. Indeed, this is a novel input, pattern C, which has presented. Thus, the current spatial context does not contain sufficient information (given this network's history of inputs) to clearly determine what IR should become active. We represent this less-than-maximal support for IR_B by the gray shading of its cells. Because of this mismatch, i.e., *G* < 1.0, we add some noise into the winner selection process. The final result is that a different L2 cell than the one most strongly implicated by the deterministic inputs ends up winning the competition in one of the three CMs (the bottom right-hand one) active at t = 2. Thus, Figure 4e shows a new IR, IR_C, representing the novel pattern, C.

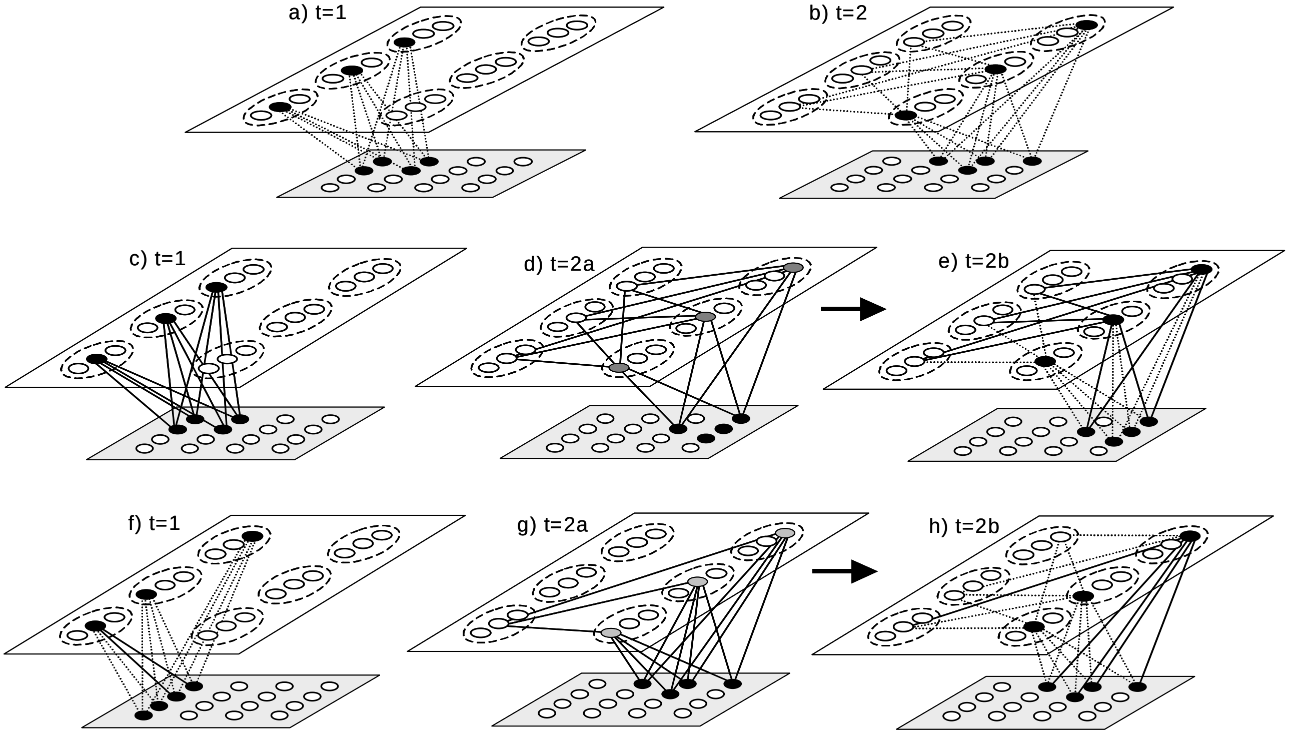

Figure 4: Sketch of TESMECOR’s spatiotemporal pattern comparison and noise-modulated internal representation selection scheme. See text for explanation. As in Figure 2, the division of the L2 CMs into separate groups for different time slices is purely to avoid clutter. In the model’s actual operation, all CMs are active on every time slice.

Figures, 4f, 4g, and 4h, show another possible scenario. This time, we will again present pattern B at t = 2. However a novel pattern, D, having only two features in common with A, presents at t = 1. As this is the first time slice of this new trial, there is no prior context vector active in the H-matrix. For concreteness, let’s assume that this degree of mismatch causes a new winner to be chosen in two of the three CMs active at t = 1, resulting in a new IR, IR_D. When B presents at t = 2, the F-vector lends maximal support for IR_B but the H-vector has great uncertainty; only 1/3 of the maximal possible horizontal input arrives at the cells of IR_B. This seems like even a worse match than in Figure 4d (shown by an even lighter shading of the IR_B cells than in Figure 4d). Consequently, more noise is added to the winner selection process. Let’s assume that this degree of mismatch leads to a new winner in two of the three CMs active at t = 2, resulting in a new IR, IR_{B*}, for pattern B.

With this example of the desired behavior in mind, we now give TESMECOR’s processing algorithm, which is computed on each time slice for each L2 cell.

1. $$\psi_{i,t} = \sum\nolimits_{j \in \Gamma_t} w_{ji} \qquad (2)$$

2. $$\Psi_{i,t} = \frac{\psi_{i,t}}{\max(\max_{j \in CM}(\psi_{j,t}), {}^F\Theta_t)} \qquad (3)$$

3. $$\phi_{i,t} = \sum\nolimits_{j \in \Delta_{t-1}} w_{ji} \quad ,t > 0 \qquad (4)$$

4. $$\Phi_{i,t} = \frac{\phi_{i,t}}{\max(\max_{j \in CM}(\phi_{j,t}), {}^H\Theta_t)} \quad ,t > 0 \qquad (5)$$

5. $$\chi_{i,t} = \begin{cases} \Psi_{i,t}{}^u \Phi_{i,t}{}^v & ,t > 0 \\ \Psi_{i,t}{}^w & ,t = 0 \end{cases} \qquad (6)$$

6. $$\mathrm{X}_{i,t} = \frac{\chi_{i,t}}{\max(\max_{j \in CM}(\chi_{j,t}), {}^\chi\Theta)} \qquad (7)$$

7. $$\pi_{k,t} = \max_{j \in CM_k} \mathrm{X}_{j,t} \quad ,1 \le k \le Q \qquad (8)$$

8. $$G_t = \sum\nolimits_{k=1}^{Q} \pi_{k,t} \Big/ Q \qquad (9)$$

9. $$p_{i,t} = \frac{f(\mathrm{X}_{i,t}, G_t)}{\sum_{j \in CM} f(\mathrm{X}_{i,t}, G_t)} \qquad (10)$$

In step 1, each L2 cell, *i*, computes its total weighted input, $\psi_{i,t}$, from the set, Γ_t, of currently active L1 cells. In step 2, the ψ values are normalized within each CM. That is, we find the maximum ψ value, in each CM and divide all the individual values by the greater of that value and

F-matrix threshold, ${}^{F}\Theta_t$. ${}^{F}\Theta_t$ is needed to ensure that small feedforward signals are not amplified in subsequent normalization steps. ${}^{F}\Theta_t$ is a parameter that can vary from one time slice to the next but we omit discussion of this detail in this paper due to space limitations.

Steps 3 and 4 perform analogous operations for the horizontal inputs. In step 3, *i*, computes its total weighted input, $\phi_{i,t}$, from the set, Δ_{t-1}, of L2 cells active on the prior time slice. In step 4, the ϕ values are normalized within each CM. That is, we find the maximum ϕ value, in each CM and divide all the individual values by the greater of that value and an H-matrix threshold, ${}^{H}\Theta_t$. ${}^{H}\Theta_t$ is needed to ensure that small H values are not amplified in subsequent normalization steps. ${}^{H}\Theta_t$ also varies from one time slice to the next but again, space limitations force us to omit discussion of this detail. Note that steps 3 and 4 are only applicable on non-initial time slices ($t > 0$) of episodes.

Step 5 works differently on the first time slices of episodes than on the rest. When $t > 0$, we multiply the two pieces of evidence, $\Psi_{i,t}$ and $\Phi_{i,t}$, that cell *i* should become active but we do this after passing them through separate exponential filters. Since $\Psi_{i,t}$ and $\Phi_{i,t}$, are both between 0 and 1, the final $\chi_{i,t}$ values output from this step are also between 0 and 1. The exponential filters effect a generalization gradient: the higher the exponents, *u* and *v*, the sharper the gradient and the more sensitive the model is to differences between inputs (i.e., the finer the spatiotemporal categories it would form) and the less overlap between the internal representations chosen by the model. When $t = 0$, we do not have two vectors to compare. Instead, we simply pass the Ψ values through an exponential gradient-controlling filter. The three different exponent parameters, *u*, *v*, and *w*, simply let us fine-tune the model's generalization gradients. For example, we might want the model's sensitivity to featural similarity to be stricter at the beginning of episodes than on the successive time slices of episodes; thus we would set *w* higher than *u*.

In step 6, we normalize the combined evidence vector, again subject to a threshold parameter, ${}^{\chi}\Theta_t$, that prevents small values from erroneously being amplified. In step 7, we simply determine the maximum value, $\pi_{i,t}$, of the $X_{i,t}$ values in each CM. These π values constitute local, i.e., within each CM, comparisons between the model's expected and actual inputs. In step 8, we compute the average of these local comparison results across the *Q* CMs of L2, resulting in the model's global comparison, G_t, of its expected and actual inputs.

In step 9, we convert the $X_{i,t}$ values back into a probability distribution whose shape depends on G_t. We want to achieve the following: if G_t is 1.0, indicating that the actual input has perfectly matched the model's expected input, then, in each CM, we want to choose, with probability 1.0, the cell belonging to the IR representing that expected input. That cell, in each CM, is the one having the highest X value. Since, in general, other cells in that cell's CM could have non-zero or even high X values, we need to filter the values by an expansive nonlinearity, *f*, so that the cell with the maximal X value maps to a probability, $p_{i,t}$, of 1.0 and the rest of the cells end up mapping to $p_{i,t} = 0.0$. On the other hand, if $G_t = 0$, indicating that the actual input is completely unexpected in the current temporal context given all of the model's past experience, then we want to make all the cells, in any given CM, be equally likely to be chosen winner. Thus, in this case, *f* should be a compressive nonlinearity that maps all cells in the CM to $p = 1/K$, where *K* is the number of cells in the CM. Without going into details, the function, *f*, is a sigmoid that meets the above goals. In the last stage of step 9, we simply choose the winner in each CM according to the resulting distribution.

To summarize, on each time slice, every L2 cell compares two evidence vectors, the H-vector, reflecting the sequence of patterns leading up to the present time slice (temporal context), and the F-vector, reflecting the current spatial pattern (spatial context). These vectors are separately nonlinearly filtered and then multiplicatively combined. The combined evidence vector is then renormalized and nonlinearly filtered before being turned into a probability distribution that governs the final selection of L2 cells to become active. Note that this basic scheme can be extended to simultaneously compare other evidence vectors as well. This is one of our intended lines of future research: specifically, we will examine incorporating a hippocampal component to the model, which will provide a third evidence vector to the L2 cells.

The concept of controlling the embedding of internal representations (IRs) based on comparing the expected and actual inputs is common to other cognitive models, e.g., Grossberg (1987). However, TESMECOR's use of distributed IRs, rather than singleton IRs, requires a generalized comparison scheme. Specifically, with distributed IRs, there exists a range of possible degrees of overlap between IRs. We want to use that range to represent the spatiotemporal similarity structure of the environment to which the model has been exposed. Therefore, rather than having a single threshold for judging the similarity of the current input and expected inputs (e.g., ART's vigilance parameter), TESMECOR's continuous-valued similarity measure, *G*, is used to inject a variable amount of noise into the IR-selection process, which in turn allows for selecting IRs whose degrees of overlap are correlated with their spatiotemporal similarities.

Simulation Results

In this section, we provide the results of preliminary investigations of the model demonstrating that it performs similarity-based generalization and categorization in the spatiotemporal pattern domain.

The four simulations described in Table 2 were performed as follows. In the learning phase, E episodes were presented, once each. Each episode consisted of 5 time slices, each having 20 (out of 100) randomly selected features present. Then, perturbed versions, differing by d = 2, 4, 6, or 8 (out of 20) features per time slice from the original episodes were generated. The model was then tested by presenting the first Z time slices of the perturbed episodes as prompts. Following the prompt time slices, the model entered a *free-running* mode (i.e. cutting off any further input) and processing continued from that point merely on the basis of signals propagating in the H-projection.

Table 2: Generalization/Categorization Results

Simulation	**E**	***d***	**Z**	R_{set}
1	27	2	1	92.3%
2	13	4	1	98.0%
3	7	6	1	98.3%
4	13	8	2	82.7%

These results indicate that the model was extremely good at locking into the trace corresponding to the most-closely-matching original episode. The accuracy measure, R_{set} (eq. 1) measures how close the recall L2 trace is to the L2 trace of the most-closely-matching original episode. The accuracy for simulation 4 (82.7%) may seem low. However, if the accuracy measure is taken only for the final time slice of each episode then it is close to 100% for all four simulations. The view taken herein is that given that the pattern to be recalled are spatiotemporal, the most relevant measure of performance is the measure of accuracy on the last time slice of the test episode. If the model can “lock into" the correct memory trace by the end of the recalled trace, then that should be sufficient evidence that model has recognized the input as an instance of a familiar episode.

Table 3: Per-Time-Slice L2 Accuracy for the Test Trials of Simulation 4 of Table 2

Episode	**T=1**	**T=2**	**T=3**	**T=4**	**T=5**
1	0.9	0.9	1.0	1.0	1.0
2	0.82	1.0	1.0	1.0	1.0
3	0.67	1.0	1.0	1.0	1.0
4	0.82	0.9	1.0	1.0	1.0
5	0.67	0.82	1.0	1.0	1.0
6	0.67	0.9	1.0	1.0	1.0
7	0.9	1.0	1.0	1.0	1.0
8	0.74	1.0	1.0	1.0	1.0
9	0.74	1.0	1.0	1.0	1.0
10	0.67	0.82	1.0	1.0	1.0
11	0.54	0.67	0.22	0.0	0.0
12	0.48	0.21	0.0	0.0	0.0
13	0.82	0.9	1.0	1.0	1.0

Table 3 shows the details of the simulation 4 in Table 2. Specifically, it shows the L2 accuracy on each time slice of each episode during the recall test. For each recall trial the model received a prompt consisting of degraded versions of the first two time slices of the original episode—specifically, 4 out of 20 features were substituted on each time slice (for a total of 8 featural differences). In all but two cases, the model ‘locks into’ the L2 trace corresponding to the most-closely-matching original episode (i.e., the episode from which the degraded prompt was created.

These simulations provide preliminary evidence that TESMECOR exhibits generalization, and in fact categorization, in the spatiotemporal domain, while at the same time exhibiting episodic memory since the episodes are learned with single trials.

Acknowledgments

Thanks to Daniel H. Bullock, Michael E. Hasselmo, Michael A. Cohen and Frank Guenther, for discussions that provided many helpful ideas during the formulation of this theory and preparation of the doctoral dissertation from which this work was derived.

References

Ans, B., Rousset, S., French, R.M., & Musca, S. (2002) Preventing Catastrophic Interference in Multiple-Sequence Learning Using Coupled Reverberating Elman Networks. *Proc. of the 24th Annual Conf. of the Cognitive Science Society*. LEA, NJ.

Carpenter, G. & Grossberg, S. (1987) Massively parallel architectures for a self-organizing neural pattern recognition machine. *Computer Vision, Graphics and Image Processing*. **37**, 54-115.

Coultrip, R. L. & Granger, R. H. (1994) Sparse random networks with LTP learning rules approximate Bayes classifiers via Parzen’s method. *Neural Networks*, **7**(3), 463-476.

Lynch, G. (1986) Synapses, Circuits, and the Beginnings of Memory. The MIT Press, Cambridge, MA.

McClelland, J. L., McNaughton, B. L., & O’Reilly, R. C. (1995) Why there are complementary learning systems in the hippocampus and neocortex: Insights from the successes and failures of connectionist models of learning and memory. *Psychological Review*, **102**, 419-457.

Moll, M. & Miikkulainen, R. (1995) Convergence-Zone Episodic Memory: Analysis and Simulations. Tech. Report AI95-227. The University of Texas at Austin, Dept. of Computer Sciences.

O'Reilly, R. C. & Rudy, J. W. (1999) Conjunctive Representations in Learning and Memory: Principles of Cortical and Hippocampal Function. TR 99-01. Institute of Cognitive Science, U. of Colorado, Boulder, CO

Palm, G. (1980) On Associative Memory. *Biological Cybernetics*, **36**. 19-31.

Rinkus, G. J. (1995) TEMECOR: An Associative, Spatiotemporal Pattern Memory for Complex State Sequences. *Proc. of the 1995 World Congress on Neural Networks*. LEA and INNS Press. 442-448.

Rinkus, G. J. (1996) A Combinatorial Neural Network Exhibiting both Episodic Memory and Generalization for Spatio-Temporal Patterns. Ph.D. Thesis, Graduate School of Arts and Sciences, Boston University.

Willshaw, D., Buneman, O., & Longuet-Higgins, H. (1969) Non-holographic associative memory. *Nature*, **222**, 960-962