

1991-02

Working memory networks for learning multiple groupings of temporally ordered events: applications to 3-D visual object recognition

<https://hdl.handle.net/2144/2062>

"Downloaded from OpenBU. Boston University's institutional repository."

**WORKING MEMORY NETWORKS FOR
LEARNING MULTIPLE GROUPINGS OF
TEMPORALLY ORDERED EVENTS: APPLICATIONS
TO 3-D VISUAL OBJECT RECOGNITION**

Gary Bradski, Gail A. Carpenter, and Stephen Grossberg

February, 1991

Technical Report CAS/CNS-TR-91-007

Permission to copy without fee all or part of this material is granted provided that 1. the copies are not made or distributed for direct commercial advantage, 2. the report title, author, document number, and release date appear, and 3. notice is given that copying is by permission of the BOSTON UNIVERSITY CENTER FOR ADAPTIVE SYSTEMS AND DEPARTMENT OF COGNITIVE AND NEURAL SYSTEMS. To copy otherwise, or to republish, requires a fee and/or special permission.

Copyright © 1991

Boston University Center for Adaptive Systems and
Department of Cognitive and Neural Systems
111 Cummington Street
Boston, MA 02215

WORKING MEMORY NETWORKS FOR LEARNING MULTIPLE GROUPINGS OF TEMPORALLY ORDERED EVENTS: APPLICATIONS TO 3-D VISUAL OBJECT RECOGNITION

Gary Bradski, Gail A. Carpenter, and Stephen Grossberg

Center for Adaptive Systems and Graduate Program in Cognitive and Neural Systems
Boston University, 111 Cummington Street, Boston, MA 02215

ABSTRACT Working memory neural networks are characterized which encode the invariant temporal order of sequential events that may be presented at widely differing speeds, durations, and interstimulus intervals. This temporal order code is designed to enable all possible groupings of sequential events to be stably learned and remembered in real time, even as new events perturb the system. Such a competence is needed in neural architectures which self-organize learned codes for variable-rate speech perception, sensory-motor planning, or 3-D visual object recognition. Using such a working memory, a self-organizing architecture for invariant 3-D visual object recognition is described that is based on the model of Seibert and Waxman [1].

1. INTRODUCTION. Working memory is the type of memory whereby a telephone number, or other novel temporally ordered sequence of events, can be temporarily stored and then performed [2]. Working memory, a kind of short term memory (STM), can be quickly erased by a distracting event, unlike long term memory (LTM). The article describes a working memory architecture for the storage of temporal order information across a series of item representations. There is a large experimental literature about working memory, as well as a variety of models ([1] - [28]).

The present class of models exhibits a set of properties that have heretofore not been available in a dynamically defined working memory. In particular, these working memories are designed to encode the invariant temporal order of sequential events that may be presented at widely differing speeds, durations, and interstimulus intervals. Moreover, despite these variations, the temporal order code is designed to enable *all possible groupings* of sequential events to be stably learned and remembered in real time, even as new events perturb the system. In other words, these working memories enable chunks (also called compressed, categorical, or unitized representations) of variable size to be learned in a manner that is not destabilized by the continuous barrage of new inputs to the working memory. Such chunks are used to represent the most informative combinations of recently stored items, and to predict and control future behavior based upon these representations.

Working memories with these properties are important in many applications wherein properties of behavioral self-organization are needed. Three important applications are real time self-organization of codes for variable-rate speech perception, sensory-motor planning, and 3-D visual object recognition. Architectures for the first two types of application are described in [4] and [5]. Herein we outline how such a working memory can both simplify and extend the capabilities of the Siebert and Waxman model for 3-D visual object recognition ([1], [29]).

2. LTM INVARIANCE PRINCIPLE AND STM NORMALIZATION. The neural network working memories described herein are based upon algebraically characterized working memories that were introduced by Grossberg ([9], [10]). These algebraic working memories were defined to explain a variety of challenging psychological data concerning working memory storage and recall. This analysis led to the articulation of two principles for the design of a working memory, the LTM Invariance Principle and the Normalization Rule.

LTM Invariance Principle: The spatial patterns of STM activation across the item representations of a working memory are stored and reset in response to sequentially

presented events in such a way as to leave the LTM codes of all past event groupings invariant.

The LTM Invariance Principle is algebraically realized as follows. Let field F_1 represent a working memory with activity x_i at the i^{th} item representation, field F_2 represent a chunking network — for example, a masking field [4] —, and let an adaptive filter $F_1 \rightarrow F_2$ compress STM patterns of item and order information across F_1 into F_2 chunks via competitive learning. Let $T_j = \sum_i x_i z_{ij}$ denote the total input to the j^{th} F_2 node, and z_{ij} denote the LTM trace in the ij^{th} pathway. Then past LTM groupings are left invariant under perturbation of new items if

$$x_i(t_{j+1}) = \begin{cases} 0 & \text{if } i > j \\ \mu_i & \text{if } i = j \\ \omega_j x_i(t_j) & \text{if } i < j. \end{cases} \quad (1)$$

In other words, the pattern $(x_1, x_2, \dots, x_{j-1})$ of already stored STM activities is multiplied by a common factor ω_j as the j^{th} item is instated with some activity μ_j . Such a reset event does not change the direction of the vector (T_1, T_2, \dots, T_m) of previously positive inputs to F_2 when new events are stored in F_1 . Solving (1) recursively yields the formula

$$S_i = \sum_{m=1}^i \prod_{r=m+1}^i \omega_r \mu_m \quad (2)$$

for the total activity S_i in F_1 after i items are stored there.

Normalization Rule: This rule algebraically instats the classical property of limited capacity of STM ([3], [21]). One convenient realization is the equation

$$S_i = \mu_1 \theta_i + M(1 - \theta_i), \quad (3)$$

where θ_i decreases towards 0 as i increases. For example, let $\theta_i = \theta^{i-1}$. An explicit equation for the reset parameters ω_k can be derived from (2) and (3); namely,

$$\omega_k = \frac{\mu_1 \theta_k + M(1 - \theta_k) - \mu_k}{\mu_1 \theta_{k-1} + M(1 - \theta_{k-1})}, \quad (4)$$

in terms of the μ_k and θ_k . Remarkably, the LTM Invariance Principle, which enforces temporally stable learning and LTM, implies the characteristic anomalies of STM storage—such as primacy gradients, recency gradients, and bows—that occur in experiments on working memory ([9], [10], [13]).

The multiplicative gating in (1) and the activity normalization in (3) are algebraic versions of properties that arise as emergent properties in shunting competitive feedback networks ([11], [30]). In order to embed working memories into hierarchies of self-organizing chunking networks, they must be defined by real time networks that are computationally consistent with the chunking nets with which they interact.

3. WORKING MEMORIES INVARIANT UNDER VARIABLE INPUT SPEED, DURATION, AND INTERSTIMULUS INTERVAL. At least two types of working memory models can be imagined: *transient* models and *sustained* models. In a transient model, presentation of items of different durations can alter the stored temporal order information. Good temporal order can be achieved using a 1-level model whose input durations are controlled by a preprocessing stage [31]. Such models cannot, however, handle data streams wherein the speeds, durations, and interstimulus intervals of individual

events vary greatly. Sustained models can. Just so long as each input duration exceeds a minimal time interval, that can be chosen arbitrarily small, all other temporal input fluctuations have essentially no effect on memory storage.

Sustained models may be realized by 2-level networks. We call them STORE models, for Sustained Temporal Order REcurrent models. Two illustrative networks are defined below. In both models let I_i equal the i^{th} input (chosen to equal 0 or 1.5 over a variable duration), let x_i equal the activity of the i^{th} node in F_1 , and let y_i equal the activity of the i^{th} node in F_2 .

STORE Model 1

$$\frac{d}{dt}x_i = (\alpha x_i + y_i + I_i)I \quad (5)$$

$$\frac{d}{dt}y_i = (x_i - y_i)I^c \quad (6)$$

where

$$\alpha = A + B + C \sum_k y_k - \sum_k x_k, \quad (7)$$

$$I = \begin{cases} 1 & \text{if } \sum_k I_k > 0 \\ 0 & \text{otherwise} \end{cases} \quad (8)$$

$$I^c = \begin{cases} 1 & \text{if } \sum_k I_k = 0 \\ 0 & \text{otherwise.} \end{cases} \quad (9)$$

STORE Model 2

$$\frac{d}{dt}x_i = (\alpha x_i + y_i + I_i)I \quad (10)$$

$$\frac{d}{dt}y_i = (x_i + \beta y_i)I^c \quad (11)$$

where α , I , and I^c are defined as above, and

$$\beta = A + B \sum_k x_k - \sum_k y_k. \quad (12)$$

Figure 1 summarizes a parametric series of computer simulations of Model 1, which show how the model generates characteristic STM recency gradients (top left rows), STM primacy gradients (bottom left rows), and STM bows (middle left rows), in different parameter ranges, as eight items are presented through time. Input durations were varied randomly from 10 to 40 integration steps with no discernible effect on order storage. Bar height designates equilibrated STM activity of the corresponding item. The middle columns represent the ratios of successive activities x_i/x_{i+1} through time. After a brief initial input phase, they are constant through time, thereby confirming the LTM Invariance Principle. The rightmost columns represent the growth of total STM activity $\sum_k x_k$ across F_1 (solid line) and total STM activity $\sum_k y_k$ across F_2 (dotted line). A partial breakdown of the normalization rule helps to control whether an STM bow occurs and, if so, at what

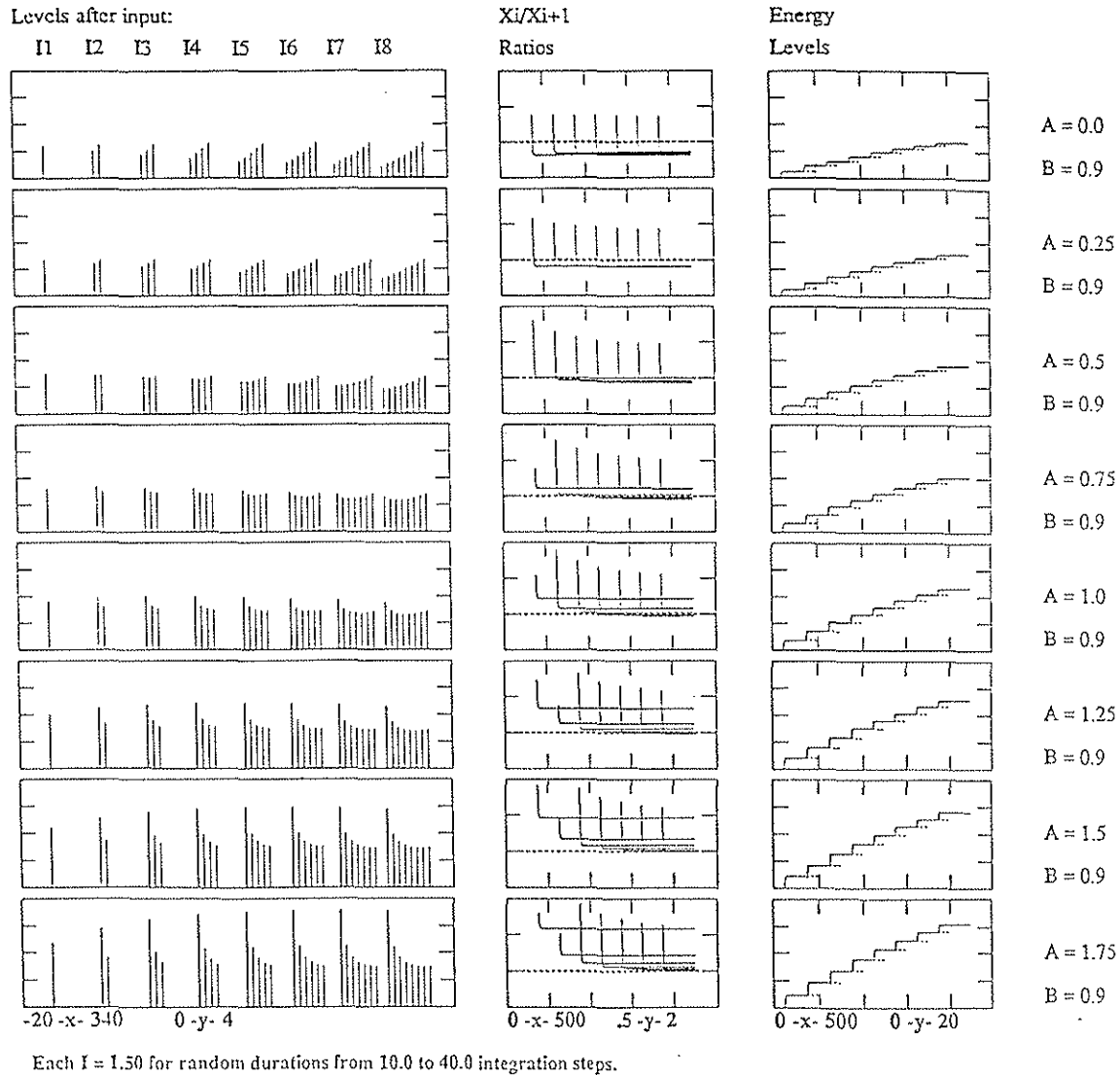


Figure 1.

list position. A variant of this design can be used to encode temporal order of lists in which each item may occur multiple times [31].

4. A SELF-ORGANIZING ARCHITECTURE FOR INVARIANT 3-D VISUAL OBJECT RECOGNITION. Seibert and Waxman ([1], [29]) have developed a self-organizing neural network architecture for invariant 3-D visual object recognition. In response to moving objects in space, preprocessing stages in the architecture automatically generate 2-D patterns that are invariant under changes in object position, size, and orientation. These patterns form the inputs to an ART 2 [32] network that self-organizes learned category representations of the invariant patterns. Each category encodes a 2-D "aspect" of the object; that is, it forms a single compressed representation of a collection of similar 2-D views of the object. The ART 2 vigilance parameter controls how similar

these 2-D views must be in order to activate the same category node J . As the object moves with respect to the camera, a temporally ordered sequence J_1, J_2, \dots, J_m of category nodes is activated. These nodes and their ordering implicitly represent invariant 3-D properties of the object, in much the same manner as an "aspect graph" [33]. Seibert and Waxman developed a specialized network, called an Aspect Network, to represent the temporal order of 2-D aspects. They then fed the output of each temporal order network into a competitive learning network, each of whose 3-D Object Nodes learns to fire only when a correct sequence of 2-D aspects is activated.

Two problems of the Seibert and Waxman model are (1) combinatorial explosion; and (2) sensitivity to fluctuations in input speed, duration, and interstimulus interval. Combinatorial explosion occurs in the Aspect Graph mechanism for encoding temporal order. Each possible temporal order uses a different Aspect Network to compute products of the temporally overlapping STM traces of all successive input pairs. The spatial loci of these intersecting quantities project through an adaptive filter to the corresponding Object Node. In order to compute all possible objects that can be represented by M distinct (and non-repeated) 2-D aspect nodes N_i , one needs to represent $M!$ temporal orderings by $M!$ Aspect Networks. Each Aspect Network computes $O(M^2)$ products which requires $O(M^2)$ pathways to each Object Node. In our modified architecture, only one sustained-type working memory is needed with M nodes to represent all $M!$ temporal orders, no Aspect Network is needed, and only $O(M)$ pathways are needed to each ART 2 Object Node.

The Seibert-Waxman scheme for computing temporal order information, by using products of successive STM traces, is also sensitive to changes in input speed, duration, and interstimulus interval. They partially overcome this problem using a specialized LTM law whose adaptive weight converges to 1 if the corresponding product exceeds a threshold, and zero otherwise. This problem does not arise if a sustained working memory is used. The temporal order code of the working memory can input an ART 2 network, which automatically learns to select different category nodes in response to different analog patterns of temporal order information over the same fixed set of working memory nodes.

We also augment the Seibert-Waxman scheme by letting 3-D Object Nodes be used to learn arbitrary output names via outstar learning ([11], [34]). Each ART 2 Object Node is the source cell of an outstar. All the outstars converge on the same outstar border where an output name can be represented in an arbitrary format by an external teacher. The total self-organizing system is the following cascade of processing stages: (Invariant preprocessor) \rightarrow ART 2 (2-D aspects) \rightarrow (Invariant Working Memory) \rightarrow ART 2 (3-D objects) \rightarrow (Feedforward Outstar Network). This is a self-organizing multi-level instar-outstar map specialized for invariant 3-D object recognition [35].

5. CONTROL OF WORKING MEMORY AND TEMPORAL LEARNING.

Reset of the working memory can be autonomously controlled by the Seibert and Waxman object tracking system. This system enables the camera to continuously track an object, as a sequence of 2-D aspects are learned and encoded in working memory, after which a ballistic camera movement focuses on a new object. We assume that working memory is reset when a ballistic movement occurs; for example, by reducing the gain of the working memory's recurrent interactions. As a result, each sequence of simultaneously stored 2-D aspects represents the same 3-D object.

ART 2 learning of each working memory pattern may be controlled in either of two ways: (1) Unsupervised: each new entry into working memory causes ART 2 to choose and learn a new category. Each subsequence $(J_1), (J_1, J_2), (J_1, J_2, J_3), \dots$ of 2-D aspect nodes can then learn to activate its own ART 2 node. Only those subsequences which are associated with names of 3-D objects generate output predictions. (2) Supervised: an ART 2 learning gate is opened only when a teaching input to an outstar occurs. Here,

only those sequences (J_1, J_2, \dots) that generate 3-D object predictions will learn to activate ART 2 categories and their outstar predictions. The number of learned ART 2 categories is hereby minimized.

ACKNOWLEDGEMENTS Supported in part by the Air Force Office of Scientific Research (AFOSR 90-128, AFOSR 90-0175), British Petroleum (89-A-1204), DARPA (AFOSR 90-0083), and the National Science Foundation (NSF IRI 90-00530, NSF IRI 87-16960). The authors wish to thank Kelly Dumont and Carol Yananakis Jefferson for their valuable assistance in the preparation of the manuscript.

REFERENCES

1. Seibert, M. and Waxman, A.M. In M. Caudill (Ed.). *Proceedings of IJCNN-90*, Washington, D.C., 2, 233-236; Hillsdale, NJ: Erlbaum, 1990.
2. Baddeley, A.D. *The psychology of memory*. New York: Basic Books, 1976.
3. Atkinson, R.C., and Shiffrin, R.M. *Scientific Amer.*, August, 1971.
4. Cohen, M.A. and Grossberg, S. *Applied Optics*, 26, 1866-1891, 1987.
5. Cohen, M.A., Grossberg, S. and Stork, D. In M. Caudill and C. Butler (Eds.) *Proceedings of the IEEE int'l conference on neural networks*, San Diego, IV, 443-454, 1987.
6. Elman, J.L. *Cognitive Science*, 179-211, 1990.
7. Grossberg, S. *J. Math. and Mech.*, 19, 53-91, 1969.
8. Grossberg, S. *Stud. in Applied Math.*, 49, 135-166, 1970.
9. Grossberg, S. *J. Math. Psychol.*, 17, 199-219, 1978a.
10. Grossberg, S. In R. Rosen and F. Snell (Eds.), *Progress in theoretical biology*, 5, 233-374. New York: Academic Press, 1978b. Reprinted in [11].
11. Grossberg, S. *Studies of mind and brain*. Boston: Reidel Press, 1982.
12. Grossberg, S. and Pepe, J. *J. of Statistical Physics*, 3, 95-125, 1971.
13. Grossberg, S. and Stone, G.O. *Memory and Cognition*, 14, 451-468, 1986.
14. Gutfreund, H. and Mezard, M. *Physiology Review Letters*, 61, 235-238, 1988.
15. Guyon, I., Personnaz, L., Nadal, J.P., and Dreyfus, G. *Physiology Review A*, 38, 6365-6372, 1988.
16. Healy, A.F. *J. Verbal Learning and Verbal Behavior* 13, 644-655, 1975.
17. Hogan, R.M. and Hogan, M.M. *Memory and Cognition*, 3, 210-215, 1975.
18. Jordan, M.I. Institute for Cognitive Science, Report 8604. University of California, San Diego, 1986.
19. Sperling, G. and Melchner, J. *Math. Psych.*, 3, 192-213, 1976.
20. Lee, C. and Estes, W.K. *J. Exp. Psychol.: Human Learning and Memory*, 1, 149-169, 1981.
21. Miller, G.A. *Psychol. Review*, 63, 81, 1956.
22. Murdock, B.B. Jr. *J. Exp. Psychol.*, 64, 482-488, 1962.
23. Schreter, Z. and Pfeifer, R. In L. Personnaz and G. Dreyfus (Eds.). *Neural networks from models to applications* Paris: I.D.S.E.T. 1989.
24. Ratcliff, R. *Psychol. Rev.*, 85, 59-108, 1978.
25. Reeves, A. and Sperling, G. *Psychol. Rev.*, 93, 180-206, 1986.
26. Rundus, D. *J. Exp. Psychol.*, 89, 63-77, 1971.
27. Tulving, E. *Elements of episodic memory*. Oxford: Clarendon Press, 1983.
28. Wang, D. and Arbib, M.A. *Proceedings of the IEEE*, 78, 1536-1543, 1990.
29. Seibert, M.C. *Neural networks for machine vision: Learning three-dimensional object recognition*. Boston University, Ph.D. Thesis, 1991.
30. Grossberg, S. *Studies in Applied Math.*, 52, 217-257, 1973.
31. Bradski, G., Carpenter, G.A., and Grossberg, S. Working memory networks for learning multiple groupings of temporally ordered events: Applications to 3-D object recognition. *in preparation*.
32. Carpenter, G.A. and Grossberg, S. ART 2: Self-organization of stable category recognition codes for analog input patterns. *Applied Optics*, 26, 4919-4930, 1987.
33. Koenderink, J.J. and van Doorn, A.J. *Biol. Cybernetics*, 32, 211-216, 1979.
34. Grossberg, S. *Proc. Nat'l Acad. of Sciences*, 59, 368-372, 1968.
35. Carpenter, G.A. and Grossberg, S. *Pattern recognition by self-organizing neural networks*. Cambridge, MA: MIT Press, 1991.