INTEGRATING SYMBOLIC AND NEURAL PROCESSING IN A SELF-ORGANIZING ARCHITECTURE FOR PATTERN RECOGNITION AND PREDICTION

Gail A. Carpenter and Stephen Grossberg

January 1993

Technical Report CAS/CNS–93–002

To appear in:
Artificial Intelligence and Neural Networks:
Steps toward Principled Integration
V. Honavar and L. Uhr (Eds.)
Academic Press

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

INTEGRATING SYMBOLIC AND NEURAL PROCESSING IN A SELF-ORGANIZING ARCHITECTURE FOR PATTERN RECOGNITION AND PREDICTION

Gail A. Carpenter† and Stephen Grossberg‡

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

Technical Report #CAS/CNS-TR-93-002
Boston, MA: Boston University

January 1993

To appear in

Artificial Intelligence and Neural Networks: Steps toward Principled Integration
V. Honavar and L. Uhr (Eds.)
Academic Press

† Supported in part by BP (89A-1204), DARPA (ONR N00014-92-J-4015), NSF (IRI 90-00530), and the Office of Naval Research (ONR N00014-91-J-4100).

‡ Supported in part by the Air Force Office of Scientific Research (F49620-92-J-0225), DARPA (ONR N00014-92-J-4015), and the Office of Naval Research (ONR N00014-91-J-4100).

Acknowledgements: The authors wish to thank Robin Locke for her valuable assistance in the preparation of the manuscript.

1. Integrating Symbolic Processing and Neural Networks

The apparent dichotomy between symbolic AI processing and distributed neural processing cannot be absolute, since neural networks that capture essential features of human intelligence will also model some of the symbolic processes of which humans are capable. Indeed, a primary goal of biological neural network research is to design systems that can self-organize intelligent symbolic processing capabilities. Such a system is summarized in this chapter.

Most if not all of the purported dichotomies between traditional artificial intelligence and neural network research dissolve within these systems. Although these systems are neural networks, they are also a type of self-organizing production system capable of hypothesis testing and memory search. They embody both continuous and discrete, parallel and serial, and distributed and localized properties. Their symbols are compressed, often digital representations, yet they are formed and stabilized through a process of resonant binding that is distributed across the system. They are used to explain and predict data on both the psychological and the neurobiological levels, yet their unique combinations of computational properties are also rapidly finding their way into technology. They are capable of autonomously discovering rules about the environments to which they adapt, yet these rules are emergent properties of network dynamics rather than formal algorithmic statements. On the other hand, these emergent rules can be rewritten as algorithmic if-then rules by a human observer or properly programmed computer.

This synthesis has become possible because such systems embody genuinely new computational principles. These are not the principles of modular construction that have been so popular in artificial intelligence. Rather they are principles of uncertainty, complementarity, symmetry, and resonance — the types of principles that are familiar in theoretical physics. We believe that these principles, which embody a new type of computation, reflect the brain's ability to adapt to the physical processes of the external world. We summarize this conclusion by calling them principles of *natural intelligence*, and anticipate that the study of artificial and natural intelligence will develop in a more cooperative manner in the coming years.

2. Properties of a Self-Organizing Neural Production System

A system architecture has gradually been developed over the past three decades that embodies these new computational principles in rigorously defined networks. The books by Carpenter and Grossberg (1991, 1992), Commons, Grossberg, and Staddon (1991), Grossberg (1982, 1987a, 1987b, 1988), and Grossberg and Kuperstein (1986, 1989) survey some of these developments. The present chapter restricts itself to only one type of model within this system. This family of models is capable of supervised learning, categorization, and prediction within a nonstationary environment of arbitrarily large size. These neural models are generically called ARTMAP (Carpenter and Grossberg, 1991, 1992; Carpenter, Grossberg, Markuzon, Reynolds, and Rosen, 1992; Carpenter, Grossberg, and Reynolds, 1991). ARTMAPs can learn arbitrary analog or binary mappings between learned categories of one feature space (e.g., visual features) to learned categories of another feature space (e.g., auditory features). They exhibit a set of rigorously demonstrated computational properties that have enabled them to perform significantly better in benchmark studies than alternative machine learning, genetic algorithm, or neural network models. We believe that this is so because the heuristics and mechanisms of the Adaptive Resonance Theory components that go into ARTMAPs were derived from a study of cognitive and neural data (Grossberg, 1987a, 1987b, 1988). In particular, ARTMAPs possess properties that an autonomous knowledge system needs to possess, but that do not yet seem to have been described in artificial intelligence algorithms. These properties enable an ARTMAP to autonomously learn, categorize, and make predictions about:

(A) Rare Events: A successful autonomous agent must be able to learn about rare

events that have important consequences, even if these rare events are similar to a surrounding cloud of frequent events that have different consequences. *Fast learning* is needed to pick up a rare event on the fly. For example, a rare medical case may be the harbinger of a new epidemic. A slightly different chemical assay may predict the biological activity of a new drug. Many traditional learning schemes use a form of slow learning that tends to average over similar event occurrences.

(B) Large Nonstationary Data Bases: Rare events typically occur in a nonstationary environment whose event statistics may change rapidly and unexpectedly through time. Individual events may also occur with variable probabilities and durations, and arbitrarily large numbers of events may need to be processed. Each of these factors tends to destabilize the learning process within traditional algorithms. New learning in such algorithms tends to unselectively wash away the memory traces of old, but still useful, knowledge. Using such an algorithm, for example, learning a new face could erase the memory of a parent's face. More generally, learning a new type of expertise could erase the memory of previous expert knowledge. ARTMAP contains a *self-stabilizing memory* that permits accumulating knowledge to be stored reliably in response to arbitrarily many events in a nonstationary environment under incremental learning conditions, until the algorithm's full memory capacity, which can be chosen arbitrarily large, is exhausted.

(C) Morphologically Variable Types of Events: In many environments, some information, including rule-like inferences, is coarsely defined whereas other information is precisely characterized. Otherwise expressed, the morphological variability of the data may change through time. For example, it may just be necessary to recognize that an object is an animal, or you may need to confirm that it is your own pet. Under autonomous learning conditions, no teacher is typically available to instruct a system about how coarse its generalization, or compression, of particular types of data should be. Multiple scales of generalization, from fine to coarse, need to be available on an as-needed basis. ARTMAP is able to automatically adjust its scale of generalization to match the morphological variability of the data. It embodies a Minimax Learning Rule that conjointly *min*imizes predictive error and *max*imizes generalization using only information that is locally available under incremental learning conditions in a nonstationary environment. This property has been used to suggest, for example, how the inferotemporal cortex can learn to recognize both fine and coarse information about the world (Carpenter and Grossberg, 1993), as demonstrated by neurophysiological experiments of Desimone (1992), Miller, Li, and Desimone (1991), Harries and Perrett (1991), Mishkin (1982), and Spitzer, Desimone, and Moran (1988), among others.

(D) Many-to-One and One-to-Many Relationships: Many-to-one learning takes two forms: categorization and naming. For example, during categorization of printed letter fonts, many similar exemplars of the same printed letter may establish a single recognition category, or compressed representation (Figure 1). Different printed letter fonts or written exemplars of the letter may establish additional categories. Each of these categories carries out a many-to-one map of exemplar into category. During naming, all of the categories that represent the same letter may be associatively mapped into the letter name, or prediction. This is a second, distinct, type of many-to-one map due to cultural, not visual, reasons.

Figure 1

One-to-many learning is used to build up expert knowledge about an object or event. A single visual image of a particular animal may, for example, lead to learning that predicts: animal, dog, beagle, and my dog "Rover" (Figure 2). A computerized record of a patient's medical check-up may lead to a series of predictions about the patient's health. In many learning algorithms, the attempt to learn more than one prediction about an event leads to unselective forgetting of previously learned predictions, for the same reason that these algorithms become unstable in response to nonstationary data. In particular, error-based learning systems, including the popular back propagation algorithm (Parker, 1982;

Rumelhart, Hinton, and Williams, 1986; Werbos, 1974), find it difficult, if not impossible, to achieve any of the computational goals (A)–(D).

Figure 2

ARTMAP systems exhibit the properties (A)–(D) because they implement a qualitatively different set of heuristics than error-based learning systems:

(E) Pay Attention: An ARTMAP can learn top-down expectations (also called prototypes, primes, or queries) that can bias the system to ignore masses of irrelevant distributed data. These queries "test the hypothesis" that is embodied by a recognition category, or symbol, as they suppress features not in the prototypical attentional focus. Thus ARTMAP embodies properties of intentionality. A large mismatch between a bottom-up input vector and a top-down expectation can drive an adaptive memory search that carries out hypothesis testing for a better category, as described below.

(F) Hypothesis Testing and Match-Learning: The system actively searches for recognition categories, or hypotheses, whose top-down expectations provide an acceptable match to bottom-up data. The top-down expectation learns a prototype that focuses attention upon that cluster of input features that it deems to be relevant. If no available category, or hypothesis, provides a good enough match, then selection and learning of a new category and top-down expectation is automatically initiated. When the search discovers a category that provides an acceptable match, the system locks into an attentive resonance through which the distributed input and its symbolic category are bound together. During this resonantly bound state, the input exemplar refines the adaptive weights of the category based on any new information in the attentional focus. Thus the Fuzzy ARTMAP system carries out match-learning, rather than mismatch-learning, because a category modifies its previous learning only if its top-down expectation matches the input vector well enough to risk changing its defining characteristics. Otherwise, hypothesis testing selects a new category on which to base learning of a novel event.

(G) Choose Globally Best Symbolic Answer: In many learning algorithms, as learning proceeds, local minima or less than optimal solutions are selected to symbolically represent the data. In ARTMAP, at any stage of learning, an input exemplar first selects the category whose top-down expectation provides the globally best match. This top-down expectation hereby acts as a prototype for the class of all the input exemplars that its category represents. After learning self-stabilizes, every input directly selects the globally best matching category without any search. This category symbolically represents all the inputs that share the same prototype. Before learning self-stabilizes, familiar events gain direct access to the globally best category without any search, even if they are interspersed with unfamiliar events that drive hypothesis testing for better matching categories. A lesion in the orienting subsystem that mediates the hypothesis testing, or memory search, process leads to a memory disorder that strikingly resembles clinical properties of medial temporal amnesia in humans and monkeys after lesions of the hippocampal formation (Carpenter and Grossberg, 1993). These and related data properties provide support for the hypothesis that the hippocampal formation carries out an orienting subsystem function as one of its several functional roles.

(H) Learn Prototypes and Exemplars: The learned prototype represents the cluster of input features that the category deems to be relevant based upon its past experience. The prototype represents the features to which the category "pays attention". In cognitive psychology, an input pattern is called an exemplar. A fundamental issue in cognitive psychology concerns whether the brain learns prototypes or exemplars. Some argue that the brain learns prototypes, or abstract types of knowledge, such as being able to recognize that a particular object is a face or an animal. Others have argued that the brain learns individual exemplars, or concrete types of knowledge, such as being able to recognize a particular face or a particular animal. Recently it has been increasingly realized that some sort of hybrid

system is needed that can learn both types of knowledge (Smith, 1990). Fuzzy ARTMAP is such a hybrid system. It uses the Minimax Learning Rule to control how abstract or concrete — how fuzzy — a category can become in order to conjointly minimize predictive generalization and maximize predictive generalization. The next section indicates how this is accomplished.

(I) Calibrate Confidence: A confidence measure, called *vigilance*, calibrates how well an exemplar matches the prototype that it selects. Otherwise expressed, vigilance measures how well the chosen hypothesis matches the data. If vigilance is low, even poorly matching exemplars can then be incorporated into one category, so compression and generalization by that category are high. The symbol here is more abstract. If vigilance is high, then even good matches may be rejected, and hypothesis testing may be initiated to select a new category. In this case, few exemplars activate the same category, so compression and generalization are low. In the limit of very high vigilance, prototype learning reduces to exemplar learning, so abstraction is minimal.

The Minimax Learning Rule is realized by adjusting the vigilance parameter in response to a predictive error. Vigilance is increased just enough to initiate hypothesis testing to discover a better category, or hypothesis, with which to match the data. In this way, a minimum amount of generalization is sacrificed to correct the error. This process is called *match tracking* because vigilance tracks the degree of match between exemplar and prototype in response to a predictive error.

(J) Rule Extraction by Adaptive Production Systems: This crucial property is directly relevant to recent controversies about putative differences between artificial intelligence and neural networks. At any stage of learning, a user can translate the state of an ARTMAP system into an algorithmic set of rules. These rules evolve as the system is exposed to new inputs. Suppose, for example, that n categories are associated with the m^{th} prediction of the network. Backtrack from prediction m along the associative pathways whose adaptive weights have learned to connect the n categories to this prediction (Figure 1). Each of these categories codes a "reason" for making the m^{th} prediction. The prototype of each category embodies the set of features, or constraints, whose binding together constitutes that category's "reason". The if-then rule takes the form: IF the features of any of these n categories are found bound together, within the fuzzy constraints that would lead to selection of that category, THEN the m^{th} prediction holds. Keeping in mind that ARTMAPs carry out hypothesis testing and memory search to discover these rules, we can see that ARTMAPs are a type of self-organizing production system (Laird, Newell, and Rosenbloom, 1987) that evolves adaptively from individual input-output experiences, as in case-based reasoning.

The if-then rules of Fuzzy ARTMAP can be read off from the learned adaptive weights of the system at any stage of the learning process. This property is particularly important in applications such as medical diagnosis from a large database of patient records, where doctors may want to study the rules by which the system reaches its diagnostic decisions. Some of these rules may already be familiar to the doctors. Others may represent novel constraint combinations which the doctors might want to evaluate for their possible medical significance. This property also sheds new light on how humans can believe that their brains somehow realize rule-like behavior although they are not algorithmically structured in a traditional sense. The Minimax Learning Rule determines how abstract these rules will become in response to any prescribed environment.

Table 1 summarizes some medical and other benchmark studies that compare the performance of Fuzzy ARTMAP with alternative recognition and prediction models. Three of these benchmarks are summarized in Sections 9 and 11. These and other benchmarks are described elsewhere in greater detail (Carpenter, Grossberg, and Iizuka, 1992; Carpenter, Grossberg, Markuzon, Reynolds, and Rosen, 1992; Carpenter, Grossberg, and Reynolds, 1991).

Table 1

(K) Properties Scale: One of the most serious deficiencies of many traditional artificial intelligence algorithms is that their desirable properties tend to break down as small toy problems are generalized to large-scale problems. In contrast, all of the desirable properties of ARTMAPs scale to arbitrarily large problems. It must be emphasized, however, that ARTMAPs solve a particular type of problem. They are not intended to solve all problems of learning or intelligence. The categorization and inference problems that ARTMAP does handle well are, however, core problems in many intelligent systems, and have been technology bottlenecks for many alternative approaches.

(L) Working Memory and Subgoal Planning: The ARTMAP architecture *per se* processes only spatial input patterns. Thus it cannot be used for temporal prediction or planning problems unless temporal input sequences are first transformed into spatial patterns by a preprocessing stage. Such a preprocessing stage takes the form of a working memory. A family of neural network working memories has been designed so that any grouping of its stored events can be stably learned by the system even if new inputs reorganize the working memory in real time (Bradski, Carpenter, and Grossberg, 1992a, 1992b). These working memories, called Sustained Temporal Order REcurrent networks, or STORE models, provide a processing substrate from which temporally evolving rules may be learned. STORE models gain biological support from their ability to explain a variety of cognitive data, such as free recall order and error data (Grossberg, 1978a), order and error data during rapid attention shifts (Grossberg and Stone, 1986a; Reeves and Sperling, 1986), reaction time data during production of planned sequences of speech or motor acts (Boardman and Bullock, 1991) and the fan effect (Grossberg, 1978b).

An architecture that combines ART and STORE modules is generically called an ARTSTORE system. We suggest that many inference and production system problems can be handled by specialized ARTSTORE systems. So far, various problems in speech perception (Cohen and Grossberg, 1986), sensory-motor planning (Grossberg and Kuperstein, 1989), and 3-D visual object recognition (Bradski, Carpenter, and Grossberg, 1992a) have been analysed using this modelling approach. ARTSTORE models provide a way for a future error to select those past subsequences of actions that can correct the error.

A summary is now given of Adaptive Resonance Theory, or ART, networks for unsupervised learning and categorization. Then a connection between certain binary ART systems and fuzzy logic is noted. Fuzzy ART networks for unsupervised learning and categorization are then described. Fuzzy ART modules are next combined into a Fuzzy ARTMAP system that is capable of supervised learning, recognition, and prediction. Along the way, benchmark comparisons of ARTMAP and Fuzzy ARTMAP with machine learing, neural network, and genetic algorithms are summarized.

3. Unsupervised Self-Organizing Feature Map and ART Systems

Adaptive Resonance Theory, or ART, was introduced as a theory of human cognitive information processing (Grossberg, 1976, 1980). The theory has since led to an evolving series of real-time neural network models for unsupervised category learning and pattern recognition. These models are capable of learning stable recognition categories in response to arbitrary input sequences with either fast or slow learning. Model families include ART 1 (Carpenter and Grossberg, 1987a), which can stably learn to categorize binary input patterns presented in an arbitrary order; ART 2 (Carpenter and Grossberg, 1987b), which can stably learn to categorize either analog or binary input patterns presented in an arbitrary order; and ART 3 (Carpenter and Grossberg, 1990), which can carry out parallel search, or hypothesis testing, of distributed recognition codes in a multi-level network hierarchy. Variations of these models adapted to the demands of individual applications have been developed by a number of authors.

Figure 3

Figure 3 illustrates one example from the family of ART 1 models, and Figure 4 illustrates a typical ART search cycle. Level F_1 in Figure 3 contains a network of nodes, each of which represents a particular combination of sensory features. Level F_2 contains a network of nodes that represent recognition codes which are selectively activated by patterns of activation across F_1. The activities of nodes in F_1 and F_2 are also called short term memory (STM) traces. STM is the type of memory that can be rapidly reset without leaving an enduring trace. For example, it is easy to reset the STM of a list of numbers that a person has just heard once by distracting the person with an unexpected event. STM is distinct from LTM, or long term memory, which is the type of memory that we usually ascribe to learning. For example, we do not forget our parents' names when we are distracted by an unexpected event.

As shown in Figure 4a, an input vector **I** registers itself as a pattern **X** of activity across level F_1. The F_1 output vector **S** is then transmitted through the multiple converging and diverging adaptive filter pathways emanating from F_1. This transmission event multiplies the vector **S** by a matrix of adaptive weights, or LTM traces, to generate a net input vector **T** to level F_2. The internal competitive dynamics of F_2 contrast-enhance vector **T**. Whereas many F_2 nodes may receive inputs from F_1, competition or lateral inhibition between F_2 nodes allows only a much smaller set of F_2 nodes to store their activation in STM. A compressed activity vector **Y** is thereby generated across F_2. In ART 1, the competition is tuned so that the F_2 node that receives the maximal $F_1 \rightarrow F_2$ input is selected. Only one component of **Y**, the symbol of the category, is nonzero after this choice takes place. Activation of such a winner-take-all node defines the category, or symbol, of the input pattern **I**. Such a category represents all the inputs **I** that maximally activate the corresponding node. So far, these are the rules of a self-organizing feature map, also called competitive learning, self-organizing feature maps, or learned vector quantization.

Figure 4

In a self-organizing feature map, only the F_2 nodes that win the competition and store their activity in STM can influence the learning process. STM activity opens a learning gate at the LTM traces that abut the winning nodes. These LTM traces can then approach, or track, the input signals in their pathways, by a process called steepest descent. This learning law is thus often called *gated steepest descent*, or *instar learning*. It was introduced by Grossberg into neural network models in the 1960's (Grossberg, 1969) and is the learning law that was used to introduce ART (Grossberg, 1976). Such an LTM trace can either increase or decrease to track the signals in its pathway. It is thus not a Hebbian associative law. It has been used to model neurophysiological data about hippocampal LTP (Levy, 1985; Levy and Desmond, 1985) and adaptive tuning of cortical feature detectors during the visual critical period (Rauschecker and Singer, 1979; Singer, 1983), lending support to ART predictions that both systems would employ such a learning law (Grossberg, 1976).

Self-organizing feature map models were introduced and computationally characterized in Grossberg (1972, 1976, 1978b), Malsburg (1973), and Willshaw and Malsburg (1976). These models were subsequently applied and further developed by many authors (Amari and Takeuchi, 1978; Bienenstock, Cooper, and Munro, 1982; Commons, Grossberg, and Staddon, 1991; Grossberg, 1982, 1987a, 1987b; Grossberg, and Kuperstein, 1989; Kohonen, 1984; Linsker, 1986; Rumelhart and Zipser, 1985). They exhibit many useful properties, especially if not too many input patterns, or clusters of input patterns, perturb level F_1 relative to the number of categorizing nodes in level F_2. It was proved that under these sparse environmental conditions, category learning is stable; the LTM traces track the statistics of the environment, are self-normalizing, and oscillate a minimum number of times; and the classifier is Bayesian (Grossberg, 1976, 1978b). It was also proved, however, that under arbitrary environmental conditions, learning becomes unstable. Such a model could forget

your parents' faces. Although a gradual switching off of plasticity can partially overcome this problem, such a mechanism cannot work in a recognition learning system whose plasticity is maintained throughout adulthood.

This memory instability is due to basic properties of associative learning and lateral inhibition. An analysis of this instability, together with data about categorization, conditioning, and attention, led to the introduction of ART models that stabilize the memory of self-organizing feature maps in response to an arbitrary stream of input patterns (Grossberg, 1976).

4. Search, Attention, and Binding

In an ART model (Carpenter and Grossberg, 1987a, 1992), learning does not occur as soon as some winning F_2 activities are stored in STM. Instead activation of F_2 nodes may be interpreted as "making a hypothesis" about an input **I**. When **Y** is activated, it quickly generates an output vector **U** that is sent top-down through the second adaptive filter. After multiplication by the adaptive weight matrix of the top-down filter, a net vector **V** inputs to F_1 (Figure 5b). Vector **V** plays the role of a learned top-down expectation. Activation of **V** by **Y** may be interpreted as "testing the hypothesis" **Y**, or "reading out the category prototype" **V**. The ART 1 network is designed to match the "expected prototype" **V** of the category against the active input pattern, or exemplar, **I**. Nodes that are activated by **I** are suppressed if they do not correspond to large LTM traces in the prototype pattern **V**. Thus F_1 features that are not "expected" by **V** are suppressed. Expressed in a different way, the matching process may change the F_1 activity pattern **X** by suppressing activation of all the feature detectors in **I** that are not "confirmed" by hypothesis **Y**. The resultant pattern **X*** encodes the cluster of features in **I** that the network deems relevant to the hypothesis **Y** based upon its past experience. Pattern **X*** encodes the pattern of features to which the network "pays attention."

If the expectation **V** is close enough to the input **I**, then a state of *resonance* develops as the attentional focus takes hold. The pattern **X*** of attended features reactivates hypothesis **Y** which, in turn, reactivates **X***. The network locks into a resonant state through the mutual positive feedback that dynamically links **X*** with **Y**. In ART, the resonant state, rather than bottom-up activation, drives the learning process. The resonant state persists long enough, at a high enough activity level, to activate the slower learning process; hence the term *adaptive resonance* theory. ART systems learn prototypes, rather than exemplars, because the attended feature vector **X***, rather than the input **I** itself, is learned. These prototypes may, however, also be used to encode individual exemplars, as described below.

5. 2/3 Rule Matching and Memory Stability

This attentive matching process is realized by combining three different types of inputs at level F_1 (Figure 3): bottom-up inputs, top-down expectations, and attentional gain control signals. The attentional gain control channel sends the same signal to all F_1 nodes; it is a "nonspecific", or modulatory, channel. Attentive matching obeys a 2/3 Rule (Carpenter and Grossberg, 1987a): an F_1 node can be fully activated only if two of the three input sources that converge upon it send positive signals at a given time.

The 2/3 Rule allows an ART system to react to bottom-up inputs, since an input directly activates its target F_1 features and indirectly activates them via the nonspecific gain control channel to satisfy the 2/3 Rule (Figure 4a). After the input instates itself at F_1, leading to selection of a hypothesis **Y** and a top-down prototype **V**, the 2/3 Rule ensures that only those F_1 nodes that are confirmed by the top-down prototype can be attended at F_1 after an F_2 category is selected.

The 2/3 Rule, first and foremost, enables an ART network to realize a self-stabilizing learning process. Carpenter and Grossberg (1987a) proved that ART learning and memory

are stable in arbitrary environments, but become unstable when 2/3 Rule matching is eliminated. Thus a type of matching that guarantees stable learning also enables the network to pay attention.

6. Vigilance, Memory Search, and Category Generalization

The criterion of an acceptable 2/3 Rule match is defined by a parameter ρ called *vigilance* (Carpenter and Grossberg, 1987a, 1992). The vigilance parameter is computed in the orienting subsystem $\mathcal{A}$. Vigilance weighs how similar an input exemplar must be to a top-down prototype in order for resonance to occur. Resonance occurs if $\rho|\mathbf{I}| - |\mathbf{X}^*| \leq 0$. This inequality says that the F_1 attentional focus $\mathbf{X}^*$ inhibits $\mathcal{A}$ more than the input $\mathbf{I}$ excites it. If $\mathcal{A}$ remains quiet, then an $F_1 \leftrightarrow F_2$ resonance can develop.

Vigilance calibrates how much novelty the system can tolerate before activating $\mathcal{A}$ and searching for a different category. If the top-down expectation and the bottom-up input are too different to satisfy the resonance criterion, then hypothesis testing, or memory search, is triggered. Memory search leads to selection of a better category at level F_2 with which to represent the input features at level F_1. During search, the orienting subsystem interacts with the attentional subsystem, as in Figures 4c and 4d, to rapidly reset mismatched categories and to select other F_2 representations with which to learn about novel events, without risking unselective forgetting of previous knowledge. Search may select a familiar category if its prototype is similar enough to the input to satisfy the vigilance criterion. The prototype may then be refined by 2/3 Rule attentional focussing. If the input is too different from any previously learned prototype, then an uncommitted population of F_2 cells is selected and learning of a new category is initiated.

Because vigilance can vary across learning trials, recognition categories capable of encoding widely differing degrees of generalization or abstraction can be learned by a single ART system. Low vigilance leads to broad generalization and abstract prototypes. In a winner-take-all ART classifier, a low vigilance category is still represented by a winner-take-all choice, or symbol, but it can represent a large "fuzzy" set of input exemplars. In contrast, a category chosen under high vigilance is still a "symbol", but high vigilance leads to narrow generalization and to prototypes that represent fewer input exemplars, even a single exemplar. The vigilance parameter hereby permits a reconciliation to be made between symbolic and fuzzy representations. Thus a single ART system may be used, say, to recognize abstract categories of faces and dogs, as well as individual faces and dogs. A single system can learn both, as the need arises, by increasing vigilance just enough to activate $\mathcal{A}$ if a previous categorization leads to a predictive error (Carpenter and Grossberg, 1992; Carpenter, Grossberg, and Reynolds, 1991; Carpenter, Grossberg, Markuzon, Reynolds, and Rosen, 1992). ART systems hereby provide a new answer to whether the brain learns prototypes or exemplars. Various authors have realized that neither one nor the other alternative is satisfactory, and that a hybrid system is needed (Smith, 1990). ART systems can perform this hybrid function in a manner that is sensitive to environmental demands, including cultural conventions.

7. Memory Consolidation, Direct Access, and Neurobiological Correlates

As inputs are practiced over learning trials, the search process eventually converges upon stable categories. The process whereby search is automatically disengaged may be interpreted as a form of memory consolidation. Inputs familiar to the network access their correct category directly, without the need for search. The category selected is the one whose prototype provides the globally best match to the input pattern. If both familiar and unfamiliar events are experienced, familiar inputs can directly activate their learned categories, while unfamiliar inputs continue to trigger adaptive memory searches for better categories, until the network's memory capacity is fully utilized (Carpenter and Grossberg, 1991).

8. The ARTMAP System

The main elements of a supervised ARTMAP system are shown in Figure 5. Two ART modules, ART_a and ART_b, read vector inputs **a** and **b**. If ART_a and ART_b were disconnected, each module would self-organize category groupings for the separate input sets. In the first application described below, ART_a and ART_b are ART 1 modules coding binary input vectors. ART_a and ART_b are here connected by an inter-ART module that in many ways resembles ART 1. This inter-ART module includes a *Map Field* that controls the learning of an associative map from ART_a recognition categories to ART_b recognition categories. This map does not directly associate exemplars **a** and **b**, but rather associates the compressed and symbolic representations of families of exemplars **a** and **b**. The Map Field also controls match tracking of the ART_a vigilance parameter. A mismatch at the Map Field between the ART_a category activated by an input **a** and the ART_b category activated by the input **b** increases ART_a vigilance by the minimum amount needed for the system to search for and, if necessary, learn a new ART_a category whose prediction matches the ART_b category. The search initiated by inter-ART reset can shift attention to a novel cluster of visual features that can be incorporated through learning into a new ART_a recognition category, which can then be linked to a new ART prediction via associative learning at the Map Field.

Figure 5

9. A Binary ARTMAP Benchmark Study: Distinguishing Edible and Poisonous Mushrooms

The ARTMAP system was first tested on a benchmark machine learning database that partitions a set of vectors **a** into two classes. Each vector **a** characterizes observable features of a mushroom as a binary vector, and each mushroom is classified as edible or poisonous (Schlimmer, 1987a). The database represents the 11 species of genus *Agaricus* and the 12 species of the genus *Lepiota* described in **The Audubon Society Field Guide to North American Mushrooms** (Lincoff, 1981). These two genera constitute most of the mushrooms described in the **Field Guide** from the familiy *Agaricaceae* (order *Agaricales*, class *Hymenomycetes*, subdivision *Basidiomycetes*, division *Eumycota*). All the mushrooms represented in the database are similar to one another: "These mushrooms are placed in a single family on the basis of a correlation of characteristics that include microscopic and chemical features..." (Lincoff, 1981, p. 500). The **Field Guide** warns that poisonous and edible species can be difficult to distinguish on the basis of their observable features. For example, the poisonous species *Agaricus californicus* is described as a "dead ringer" (Lincoff, 1981, p. 504) for the Meadow Mushroom, *Agaricus campestris*, that "may be known better and gathered more than any other wild mushroom in North America" (Lincoff, 1981, p. 505). This database thus provides a test of how ARTMAP and other machine learning systems distinguish rare but important events from frequently occurring collections of similar events that lead to different consequences.

The database of 8124 exemplars describes each of 22 observable features of a mushroom, along with its classification as poisonous (48.2%) or edible (51.8%). The 8124 "hypothetical examples" represent ranges of characteristics within each species; for example, both *Agaricus californicus* and *Agaricus campestris* are described as having a "white to brownish cap," so in the database each species has corresponding sets of exemplar vectors representing their range of cap colors. There are 126 different values of the 22 different observable features. For example, the observable feature of "cap-shape" has six possible values. Consequently, the vector inputs to ART_a are 126-element binary vectors, each vector having 22 1's and 104 0's, to denote the values of an exemplar's 22 observable features. The ART_b input vectors are (1,0) for poisonous exemplars and (0,1) for edible exemplars.

9.1. Performance

The ARTMAP system learned to classify test vectors rapidly and accurately, and system performance compares favorably with results of other machine learning algorithms applied to the same database. The STAGGER algorithm reached its maximum performance level of 95% accuracy after exposure to 1000 training inputs (Schlimmer, 1987b). The HILLARY algorithm achieved similar results (Iba, Wogulis, and Langley, 1988). The ARTMAP system consistently achieved over 99% accuracy with 1000 exemplars, even counting "I don't know" responses as errors. Accuracy of 95% was usually achieved with on-line training on 300–400 exemplars and with off-line training on 100–200 exemplars. In this sense, ARTMAP was an order of magnitude more efficient than the alternative systems. In addition, with continued training, ARTMAP predictive accuracy always improved to 100%. These results are elaborated below.

Almost every ARTMAP simulation was completed in under 2 minutes on an IRIS 4D computer, with total time ranging from about 1 minute for small training sets to 2 minutes for large training sets. This is comparable to 2–5 minutes on a SUN 4 computer. Each timed simulation included a total of 8124 training and test samples, run on a time-sharing system with non-optimized code. Each 1–2 minute computation included data read-in and read-out, training, testing, and calculation of multiple simulation indices.

9.2. On-line learning

On-line learning imitates the conditions of a human or machine operating in a natural environment. An input **a** arrives, possibly leading to a prediction. If made, the prediction may or may not be confirmed. Learning ensues, depending on the accuracy of the prediction. Information about past inputs is available only through the present state of the system. Simulations of on-line learning by the ARTMAP system use each sample pair (**a**, **b**) as both a test item and a training item. Input **a** first makes a prediction that is compared with **b**. Learning follows as dictated by the internal rules of the ARTMAP architecture.

Four types of on-line simulations were carried out, using two different baseline settings of the ART_a vigilance parameter ρ_a: $\overline{\rho_a} = 0$ (forced choice condition) and $\overline{\rho_a} = 0.7$ (conservative condition); and using sample replacement or no sample replacement. With sample replacement, any one of the 8124 input samples was selected at random for each input presentation. A given sample might thus be repeatedly encountered while others were still unused. With no sample replacement, a sample was removed from the input pool after it was first encountered. The replacement condition had the advantage that repeated encounters tended to boost predictive accuracy. The no-replacement condition had the advantage of having learned from a somewhat larger set of inputs at each point in the simulation. The replacement and no-replacement conditions had similar performance indices, all other things being equal. Each of the 4 conditions was run on 10 independent simulations. With $\overline{\rho_a} = 0$, the system made a prediction in response to every input. Setting $\overline{\rho_a} = 0.7$ increased the number of "I don't know" responses, increased the number of ART_a categories, and decreased the rate of incorrect predictions to nearly 0%, even early in training. The $\overline{\rho_a} = 0.7$ condition generally outperformed the $\overline{\rho_a} = 0$ condition, even when incorrect predictions and "I don't know" responses were both counted as errors. The primary exception occurred very early in training, when a conservative system gives the large majority of its no-prediction responses.

Table 2

Results are summarized in Table 2. Each entry gives the number of correct predictions over the previous 100 trials (input presentations), averaged over 10 simulations. For example, with $\overline{\rho_a} = 0$ in the no-replacement condition, the system made, on the average, 94.9 correct predictions and 5.1 incorrect predictions on trials 201–300. In all cases a 95% correct-prediction rate was achieved before trial 400. With $\overline{\rho_a} = 0$, a consistent correct-prediction rate of over 99% was achieved by trial 1400, while with $\overline{\rho_a} = 0.7$ the 99% consistent correct-prediction rate was achieved earlier, by trial 800. Each simulation was continued for 8100 trials. In all four cases, the minimum correct-prediction rate always exceeeded 99.5% by trial

1800 and always exceeded 99.8% by trial 2800. In all cases, across the total of 40 simulations summarized in Table 2, 100% correct prediction was achieved on the last 1300 trials of each run.

Note the relatively low correct-prediction rate for $\overline{\rho_a} = 0.7$ on the first 100 trials. In the conservative mode, a large number of inputs initially make no prediction. With $\overline{\rho_a} = 0.7$ an average total of only 2 *incorrect* predictions were made on each run of 8100 trials. Note too that Table 2 underestimates prediction accuracy at any given time, since performance almost always improves during the 100 trials over which errors are tabulated.

9.3. Off-line learning

In off-line learning, a fixed training set is repeatedly presented to the system until 100% accuracy is achieved on that set. For training sets ranging in size from 1 to 4000 samples, 100% accuracy was almost always achieved after one or two presentations of each training set. System performance was then measured on the test set, which consisted of all 8124 samples not included in the training set. During testing no further learning occurred.

The role of repeated training set presentations was examined by comparing simulations that used the 100% training set accuracy criterion with simulations that used only a single presentation of each input during training. With only a few exceptions, performance was similar. In fact for $\overline{\rho_a} = 0.7$, and for small training sets with $\overline{\rho_a} = 0$, 100% training-set accuracy was achieved with single input presentations, so results were identical. Performance differences were greatest for $\overline{\rho_a} = 0$ simulations with mid-sized training sets (60–500 samples), when 2–3 training set presentations tended to add a few more ART_a learned category nodes. Thus, even a single presentation of training-then-testing inputs, carried out on-line, can be made to work almost as well as off-line training that uses repeated presentations of the training set. This is an important benefit of fast learning controlled by a match tracked search.

9.4. Off-line forced-choice learning

The simulations summarized in Table 3 illustrate off-line learning with $\overline{\rho_a} = 0$. In this forced choice case, each ART_a input led to a prediction of poisonous or edible. The number of test set errors with small training sets was relatively large, due to the forced choice. Table 3 summarizes the average results over 10 simulations at each size training set. For example, with very small, 5-sample training sets, the system established between 1 and 5 ART_a categories, and averaged 73.1% correct responses on the remaining 8119 test patterns. Success rates ranged from chance (51.8%, 1 category) in one instance where all 5 training set exemplars happened to be edible, to surprisingly good (94.2%, 2 categories). The range of success rates for fast-learn training on very small training sets illustrates the statistical nature of the learning process. Intelligent sampling of the training set or, as here, good luck in the selection of representative samples, can dramatically alter early success rates. In addition, the evolution of internal category memory structure, represented by a set of ART_a category nodes and their top-down learned expectations, is influenced by the selection of early exemplars. Nevertheless, despite the individual nature of learning rates and internal representations, all the systems eventually converge to 100% accuracy on test set exemplars using only (approximately) 1/600 as many ART_a categories as there are inputs to classify.

Table 3

With 1000-sample training sets, 3 out of 10 simulations achieved 100% prediction accuracy on the 7124-sample test set. With 2000-sample training sets, 8 out of 10 simulations achieved 100% accuracy on the 6124-sample test sets. With 4000-sample training sets, all simulations achieved 100% accuracy on the 4124-sample test sets. In all, 21 of the 30 simulations with training sets of 1000, 2000, and 4000 samples achieved 100% accuracy on test sets. The number of categories established during these 21 simulations ranged from 10 to 22, again indicating the variety of paths leading to 100% correct prediction rate.

9.5. Off-line conservative learning

As in the case of poisonous mushroom identification, it may be important for a system to be able to respond "I don't know" to a novel input, even if the total number of correct classifications thereby decreases early in learning. For higher values of the baseline vigilance $\overline{\rho_a}$, the ARTMAP system creates more ART_a categories during learning and becomes less able to generalize from prior experience than when $\overline{\rho_a}$ equals 0. During testing, a conservative coding system with $\overline{\rho_a} = 0.7$ makes no prediction in response to inputs that are too novel, and thus initially has a lower proportion of correct responses. However, the number of incorrect responses is always low with $\overline{\rho_a} = 0.7$, even with very few training samples, and the 99% correct-response rate is achieved for both forced choice ($\overline{\rho_a} = 0$) and conservative ($\overline{\rho_a} = 0.7$) systems with training sets smaller than 1000 exemplars.

Table 4

Table 4 summarizes simulation results that repeat the conditions of Table 3 except that $\overline{\rho_a} = 0.7$. Here, a test input that does not make a 70% match with any learned expectation makes an "I don't know" prediction. Compared with the $\overline{\rho_a} = 0$ case of Table 3, Table 4 shows that larger training sets are required to achieve a correct-prediction rate of over 95%. However, because of the option to make no prediction, the average test set error rate is almost always less than 1%, even when the training set is very small, and is less than .1% after only 500 training trials. Moreover, 100% accuracy is achieved using only (approximately) 1/130 as many ART_a categories as there are inputs to classify.

This benchmark study illustrates the stability, speed, and accuracy of ARTMAP on a binary data base. Many applications require classification of analog data bases. One way to achieve this using ARTMAP systems is to notice a close connection between the binary operations of ART 1 and the analog operations of fuzzy logic.

10. A Connection between ART Systems and Fuzzy Logic

Fuzzy ART is a generalization of ART 1 that incorporates operations from fuzzy logic (Carpenter, Grossberg, and Rosen, 1991). Although ART 1 can learn to classify only binary input patterns, Fuzzy ART can learn to classify both analog and binary input patterns. Moreover, Fuzzy ART reduces to ART 1 in response to binary input patterns. As shown in Figure 6, the generalization to learning both analog and binary input patterns is achieved by replacing appearances of the intersection operator ($\cap$) in ART 1 by the MIN operator ($\wedge$) of fuzzy set theory. The MIN operator reduces to the intersection operator in the binary case. Of particular interest is the fact that, as parameter α approaches 0, the function T_j which controls category choice through the bottom-up filter reduces to the operation of fuzzy subsethood (Kosko, 1986). T_j then measures the degree to which the adaptive weight vector $\mathbf{w}_j$ is a fuzzy subset of the input vector $\mathbf{I}$.

Figure 6

In Fuzzy ART, as in ARTMAP (see Figure 5), input vectors are normalized at a preprocessing stage (Figure 7). This normalization procedure, called complement coding, leads to a symmetric theory in which the MIN operator ($\wedge$) and the MAX operator ($\vee$) of fuzzy set theory (Zadeh, 1965) play complementary roles. The categories formed by Fuzzy ART are then hyper-rectangles. Figure 8 illustrates how MIN and MAX define these rectangles in the 2-dimensional case. The MIN and MAX values define the acceptable range of feature variation in each dimension. Complement coding uses on-cells (with activity $\mathbf{a}$ in Figure 7) and off-cells (with activity $\mathbf{a}^c$ in Figure 7) to represent the input pattern, and preserves individual feature amplitudes while normalizing the total on-cell/off-cell vector. The on-cell portion of a prototype encodes features that are critically present in category exemplars, while the off-cell portion encodes features that are critically absent. Each category is then defined by an interval of expected values for each input feature. For instance, Fuzzy ART would encode the feature of "hair on head" by a wide interval ([A, 1]) for the category "man", whereas

the feature "hat on head" would be encoded by a wide interval ([0, B]). On the other hand, the category "dog" would be encoded by two narrow intervals, [C, 1] for hair and [0, D] for hat, corresponding to narrower ranges of expectations for these two features.

Figure 7

Learning in Fuzzy ART is stable because all adaptive weights can only decrease in time. Decreasing weights correspond to increasing sizes of category "boxes". Smaller vigilance values lead to larger category boxes. Learning stops when the input space is covered by boxes. The use of complement coding works with the property of increasing box size to prevent a proliferation of categories. With fast learning, constant vigilance, and a finite input set of arbitrary size and composition, learning stabilizes after just one presentation of each input pattern (Carpenter, Grossberg, and Rosen, 1991). A fast-commit slow-recode option combines fast learning with a forgetting rule that buffers system memory against noise. Using this option, rare events can be rapidly learned, yet previously learned memories are not rapidly erased in response to statistically unreliable input fluctuations. When the supervised learning of Fuzzy ARTMAP controls category formation, a predictive error can force the creation of new categories that could not otherwise be learned due to monotone increase in category size through time in the unsupervised case. Supervision permits the creation of complex categorical structures without a loss of stability.

Figure 8

11. Two Analog ARTMAP Benchmark Studies: Letter and Written Digit Recognition

As summarized in Table 1, Fuzzy ARTMAP has been benchmarked against a variety of machine learning, neural network, and genetic algorithms with considerable success. An illustrative study used a benchmark machine learning task that Frey and Slate (1991) developed and described as a "difficult categorization problem" (p. 161). The task requires a system to identify an input exemplar as one of 26 capital letters A–Z. The database was derived from 20,000 unique black-and-white pixel images. The difficulty of the task is due to the wide variety of letter types represented: the twenty "fonts represent five different stroke styles (simplex, duplex, complex, and Gothic) and six different letter styles (block, script, italic, English, Italian, and German)" (p. 162). In addition each image was randomly distorted, leaving many of the characters misshapen (Figure 9). Sixteen numerical feature attributes were then obtained from each character image, and each attribute value was scaled to a range of 0 to 15. The resulting Letter Image Recognition file is archived in the UCI Repository of Machine Learning Databases and Domain Theories, maintained by David Aha and Patrick Murphy (ml_repository@ics.uci.edu).

Frey and Slate used this database to test performance of a family of classifiers based on Holland's genetic algorithms (Holland, 1980). The training set consisted of 16,000 exemplars, with the remaining 4,000 exemplars used for testing. Genetic algorithm classifiers having different input representations, weight update and rule creation schemes, and system parameters were systematically compared. Training was carried out for 5 epochs, plus a sixth "verification" pass during which no new rules were created but a large number of unsatisfactory rules were discarded. In Frey and Slate's comparative study, these systems had correct prediction rates that ranged from 24.5% to 80.8% on the 4,000-item test set. The best performance (80.8%) was obtained using an integer input representation, a reward sharing weight update, an exemplar method of rule creation, and a parameter setting that allowed an unused or erroneous rule to stay in the system for a long time before being discarded. After training, the optimal case, that had 80.8% performance rate, ended with 1,302 rules and 8 attributes per rule, plus over 35,000 more rules that were discarded during verification. (For purposes of comparison, a rule is somewhat analogous to an ART_a category in ARTMAP, and the number of attributes per rule is analogous to the size of ART_a

category weight vectors.) Building on the results of their comparative study, Frey and Slate investigated two types of alternative algorithms, namely an accuracy-utility bidding system, that had slightly improved performance (81.6%) in the best case; and an exemplar/hybrid rule creation scheme that further improved performance, to a maximum of 82.7%, but that required the creation of over 100,000 rules prior to the verification step.

Figure 9

Fuzzy ARTMAP had an error rate on the letter recognition task that was consistently less than one third that of the three best Frey-Slate genetic algorithm classifiers described above. In particular, after 1 to 5 epochs, individual Fuzzy ARTMAP systems had a robust prediction rate of 90% to 94% on the 4,000-item test set. A *voting strategy* consistently improved this performance. This voting strategy is based on the observation that ARTMAP fast learning typically leads to different adaptive weights and recognition categories for different orderings of a given training set, even when overall predictive accuracy of all simulations is similar. The different category structures cause the set of test items where errors occur to vary from one simulation to the next. The voting strategy uses an ARTMAP system that is trained several times on input sets with different orderings. The final prediction for a given test set item is the one made by the largest number of simulations. Since the set of items making erroneous predictions varies from one simulation to the next, voting cancels many of the errors. Such a voting strategy can also be used to assign confidence estimates to competing predictions given small, noisy, or incomplete training sets. Voting consistently eliminated 25%–43% of the errors, giving a robust prediction rate of 92%–96%. Moreover Fuzzy ARTMAP simulations each created fewer than 1,070 ART_a categories, compared to the 1,040–1,302 final rules of the three genetic classifiers with the best performance rates. Most Fuzzy ARTMAP learning occurred on the first epoch, with test set performance on systems trained for one epoch typically over 97% that of systems exposed to inputs for five epochs.

Rapid learning was also found in a benchmark study of written digit recognition, where the correct prediction rate on the test set after one epoch reached over 99% of its best performance (Carpenter, Grossberg, and Iizuka, 1992). In this study, Fuzzy ARTMAP was tested along with back propagation and a self-organizing feature map. Voting yielded Fuzzy ARTMAP average performance rates on the test set of 97.4% after an average number of 4.6 training epochs. Back propagation achieved its best average performance rates of 96% after 100 training epochs. Self-organizing feature maps achieved a best level of 96.5%, again after many training epochs.

In summary, on a variety of benchmarks (see also Table 1, Carpenter, Grossberg, and Reynolds, 1991, and Carpenter, Grossberg, Markuzon, Reynolds, and Rosen, 1992), Fuzzy ARTMAP has demonstrated either much faster learning, better performance, or both, than alternative machine learning, genetic, or neural network algorithms. Perhaps more importantly, Fuzzy ARTMAP can be used in an important class of applications where many other adaptive pattern recognition algorithms cannot perform well (see Section 2). These are the applications where very large nonstationary databases need to be rapidly organized into stable variable-compression categories under real-time autonomous learning conditions.

12. Concluding Remarks

Fuzzy ARTMAP is one of a rapidly growing family of attentive self-organizing learning hypothesis testing, and prediction systems that have evolved from the biological theory of cognitive information processing of which ART forms an important part (Carpenter and Grossberg, 1991, 1993; Grossberg, 1982, 1987a, 1987b, 1988). At the present time, unsupervised ART modules are being used in such diverse applications as the control of mobile robots, learning and search of airplane part inventories, medical diagnosis, 3-D visual object recognition, music recognition, seismic recognition, sonar recognition, and laser radar

recognition (Baloch and Waxman, 1991; Caudell, Smith, Johnson, Wunsch, and Escobedo, 1991; Gjerdingen, 1990; Goodman, Karburlasos, Egbert, Carpenter, Grossberg, Reynolds, Hammermeister, Marshall, and Grover, 1992; Seibert and Waxman, 1991). These applications benefit from the ability of ART systems to rapidly learn to classify large data bases in a stable fashion, to calibrate their confidence in a classification, and to focus attention upon those featural groupings that they deem to be important based upon their past experience. We anticipate that the growing family of supervised ARTMAP systems will find an even broader range of applications due to their ability to adapt the number, shape, and scale of their category boundaries, and to self-organize transparent if-then rules, as they adapt to the on-line demands of large nonstationary data bases.

REFERENCES

Amari, S.-I. and Takeuchi, A. (1978). Mathematical theory on formation of category detecting nerve cells. *Biological Cybernetics*, **29**, 127–136.

Baloch, A.J. and Waxman, A.M. (1991). Visual learning, adaptive expectations, and learning behavioral conditioning of the mobil robot MAVIN. *Neural Networks*, **4**, 271–302.

Bienenstock, E.L., Cooper, L.N., and Munro, P.W. (1982). Theory for the development of neuron selectivity: Orientation specificity and binocular interaction in visual cortex. *Journal of Neuroscience*, **2**, 32–48.

Boardman, I. and Bullock, D. (1991). A neural network model of serial order recall from short-term memory. *Proceedings of the International Joint Conference on Neural Networks*, **II**, Piscataway, NJ: IEEE Service Center, 879–884.

Bradski, G., Carpenter, G.A., and Grossberg, S. (1992a). Working memory networks for learning temporal order with application to 3-D visual object recognition. *Neural Computation*, **4**, 270–286.

Bradski, G., Carpenter, G.A., and Grossberg, S. (1992b). Working memories for storage and recall of arbitrary temporal sequences. *Proceedings of the International Joint Conferences on Neural Networks (IJCNN)-92*, Piscataway, NJ: IEEE Service Center, 57–62.

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

Carpenter, G.A. and Grossberg, S. (1987b). ART 2: Stable self-organization of pattern recognition codes for analog input patterns. *Applied Optics*, **26**, 4919–4930.

Carpenter, G.A. and Grossberg, S. (1987c). Neural Dynamics of Category Learning and Recognition: Attention, Memory Consolidation, and Amnesia. In S. Grossberg (Ed.), **The adaptive brain, I: Cognition, learning, reinforcement, and rhythm**. Amsterdam: Elsevier/North Holland, 238-286.

Carpenter, G.A. and Grossberg, S. (1990). ART 3: Hierarchical search using chemical transmitters in self-organizing pattern recognition architectures. *Neural Networks*, **3**, 129–152.

Carpenter, G.A. and Grossberg, S. (Eds.) (1991). **Pattern recognition by self-organizing neural networks**. Cambridge, MA: MIT Press.

Carpenter, G.A. and Grossberg, S. (1992). Fuzzy ARTMAP: Supervised learning, recognition, and prediction by a self-organizing neural network. *IEEE Communications Magazine*, **30**, 38–49.

Carpenter, G.A. and Grossberg, S. (1993). Normal and amnesic learning, recognition, and memory by a neural model of cortico-hippocampal interactions. *Trends in Neurosciences*, in press.

Carpenter, G.A., Grossberg, S., and Iizuka, K. (1992). Comparative performance measures of Fuzzy ARTMAP, learned vector quantization, and back propagation for handwritten character recognition. **Proceedings of the international joint conference on neural networks**, Baltimore, **I**, 794–799. Piscataway, NJ: IEEE Service Center.

Carpenter, G.A., Grossberg, S., Markuzon, N., Reynolds, J.H., and Rosen, D.B. (1992). Fuzzy ARTMAP: A neural network architecture for incremental supervised learning of analog multidimensional maps. *IEEE Transactions on Neural Networks*, **3**, 698–713.

Carpenter, G.A., Grossberg, S., and Reynolds, J.H. (1991). ARTMAP: Supervised real-time learning and classification of nonstationary data by a self-organizing neural network. *Neural Networks*, **4**, 565–588.

Carpenter, G.A., Grossberg, S., and Rosen, D.B. (1991). Fuzzy ART: Fast stable learning and categorization of analog patterns by an adaptive resonance system. *Neural Networks*, **4**, 759–771.

Caudell, T., Smith, S., Johnson, C., Wunsch, D., and Escobedo, R. (1991). An industrial application of neural networks to reusable design. **Adaptive neural systems**, Technical Report BCS-CS-ACS-91-001, Seattle, WA: The Boeing Company, pp. 185–190.

Cohen, M. and Grossberg, S. (1986). Neural dynamics of speech and language coding: Developmental programs, perceptual grouping, and competition for short term memory. *Human Neurobiology*, **5**, 1–22.

Commons, M.L., Grossberg, S., and Staddon, J.E.R. (Eds.) (1991). **Neural network models of conditioning and action.** Hillsdale, NJ: Erlbaum Associates.

Desimone, R. (1992). Neural circuits for visual attention in the primate brain. In G.A. Carpenter and S. Grossberg (Eds.), **Neural networks for vision and image processing.** Cambridge, MA: MIT Press, pp. 343–364.

Frey, P.W. and Slate, D.J. (1991). Letter recognition using Holland-style adaptive classifiers. *Machine Learning*, **6**, 161–182.

Gjerdingen, R.O. (1990). Categorization of musical patterns by self-organizing neuronlike networks. *Music Perception*, **7**, 339–370.

Goodman, P., Kaburlasos, V., Egbert, D., Carpenter, G.A., Grossberg, S., Reynolds, J.H., Hammermeister, K., Marshall, G., and Grover, F. (1992). Fuzzy ARTMAP neural network prediction of heart surgery mortality. **Proceedings of the Wang Institute Research Conference: Neural Networks for Learning, Recognition, and Control**, Boston, MA: Boston University, p. 48.

Grossberg, S. (1969). On learning and energy-entropy dependence in recurrent and nonrecurrent signed networks. *Journal of Statistical Physics*, **1**, 319–350.

Grossberg, S. (1972). Neural expectation: Cerebellar and retinal analogs of cells fired by learnable or unlearned pattern classes. *Kybernetik*, **10**, 49–57.

Grossberg, S. (1976). Adaptive pattern classification and universal recoding, II: Feedback, expectation, olfaction, and illusions. *Biological Cybernetics*, **23**, 187–202.

Grossberg, S. (1978a). Behavioral contrast in short-term memory: Serial binary memory models or parallel continuous memory models? *Journal of Mathematical Psychology*, **17**, 199–219.

Grossberg, S. (1978b). A theory of human memory: Self-organization and performance of sensory-motor codes, maps, and plan. In R. Rosen and F. Snell (Eds.), **Progress in theoretical biology**, Vol. 5. New York: Academic Press. [Reprinted in Grossberg, S. (1982). **Studies of Mind and Brain: Neural principles of learning, perception, development, cognition, and motor control.** Boston: Reidel Press].

Grossberg, S. (1980). How does a brain build a cognitive code? *Psychological Review*, **1**, 1–51.

Grossberg, S. (1982). **Studies of mind and brain: Neural principles of learning, perception, development, cognition, and motor control.** Boston: Reidel Press.

Grossberg, S. (Ed.) (1987a). **The adaptive brain, I: Cognition, learning, reinforcement, and rhythm.** Amsterdam: Elsevier/North-Holland.

Grossberg, S. (Ed.) (1987b). **The adaptive brain, II: Vision, speech, language, and motor control.** Amsterdam: Elsevier/North-Holland.

Grossberg, S. (Ed.) (1988). **Neural networks and natural intelligence.** Cambridge, MA: MIT Press.

Grossberg, S. and Kuperstein, M. (1986). **Neural dynamics of adaptive sensory-motor control: Ballistic eye movements.** Amsterdam: North-Holland.

Grossberg, S. and Kuperstein, M. (1989). **Neural dynamics of adaptive sensory-motor control: Expanded edition.** Elmsford, NY: Pergamon Press.

Grossberg, S. and Merrill, J.W.L. (1992). A neural network model of adaptively timed reinforcement learning and hippocampal dynamics. *Cognitive Brain Research*, **1**, 3–38.

Grossberg, S. and Stone, G.O. (1986). Neural dynamics of attention switching and temporal order information in short term memory. *Memory and Cognition*, **14**(6), 451–468.

Harries, M.H. and Perrett, D.I. (1991). Visual processing of faces in temporal cortex: Physiological evidence for a modular organization and possible anatomical correlates. *Journal of Cognitive Neuroscience*, **3**, 9–24.

Holland, J.H. (1980). Adaptive algorithms for discovering and using general patterns in growing knowledge bases. *International Journal of Policy Analysis and Information Systems*, **4**, 217–240.

Iba, W., Wogulis, J., and Langley, P. (1988). Trading off simplicity and coverage in incremental concept learning. In **Proceedings of the 5th international conference on machine learning**. Ann Arbor, MI: Morgan Kaufmann, 73–79.

Kohonen, T. (1984). **Self-organization and associative memory**. New York: Springer-Verlag.

Laird, J.E., Newell, A., and Rosenbloom, P.S. (1987). SOAR: An architecture for general intelligence. *Artificial Intelligence*, **33**, 1–64.

Levy, W.B. (1985). Associative changes at the synapse: LTP in the hippocampus. In W.B. Levy, J. Anderson and S. Lehmkuhle (Eds.), **Synaptic modification, neuron selectivity, and nervous system organization.** Hillsdale, NJ: Erlbaum Associates, pp. 5–33.

Levy, W.B. and Desmond, N.L. (1985). The rules of elemental synaptic plasticity. In W.B. Levy, J. Anderson, and S. Lehmkuhle (Eds.), **Synaptic modification, neuron selectivity, and nervous system organization.** Hillsdale, NJ: Erlbaum Associates, pp. 105–121.

Lincoff, G.H. (1981). **The Audubon Society field guide to North American mushrooms.** New York: Alfred A. Knopf.

Linsker, R. (1986). From basic network principles to neural architecture: Emergence of spatial-opponent cells. *Proceedings of the National Academy of Sciences*, **83**, 8779–8783.

Malsburg, C. von der (1973). Self-organization of orientation sensitive cells in the striate cortex. *Kybernetik*, **14**, 85–100.

Miller, E.K., Li, L., and Desimone, R. (1991). A neural mechanism for working and recognition memory in inferior temporal cortex. *Science*, **254**, 1377-1379.

Mishkin, M. (1982). A memory system in the monkey. *Philosophical Transactions Royal Society of London* B, **298**, 85–95.

Parker,D. B. *Learning-logic.* Invention Report, 581-64, File 1, Office of Technology Licensing, Stanford University, October, 1982.

Rauschecker, J.P. and Singer, W. (1979). Changes in the circuitry of the kitten's visual cortex are gated by postsynaptic activity. *Nature*, **280**, 58–60.

Reeves, A. and Sperling, G. (1986). Attentional theory of order information in short-term visual memory. *Psychological Review*, **93**, 180-206.

Rumelhart, D.E., Hinton, G. and Williams, R. (1986). *Learning internal representations by error propagation.* In D.E. Rumelhart and J.L. McClelland (Eds.), **Parallel distributed processing**, Cambridge, MA: MIT Press.

Rumelhart, D.E. and Zipser, D. (1985). Feature discovery by competitive learning. *Cognitive Science*, **9**, 75–112.

Schlimmer, J.S. (1987a). Mushroom database. UCI Repository of Machine Learning Databases. (aha@ics.uci.edu)

Schlimmer, J.S. (1987b). Concept acquisition through representational adjustment (Technical Report 87–19). Doctoral dissertation, Department of Information and Computer Science, University of California at Irvine.

Seibert, M. and Waxman, A.M. (1991). Learning and recognizing 3D objects from multiple views in a neural system. In H. Wechler (Ed.), **Neural networks for perception, Volume 1**. New York: Academic Press.

Singer, W. (1983). Neuronal activity as a shaping factor in the self-organization of neuron assemblies. In E. Basar, H. Flohr, H. Haken, and A.J. Mardell (Eds.), **Synergetics of the brain**. New York: Springer-Verlag, pp. 89–101.

Smith, E.E. (1990). In D.O. Osherson and E.E. Smith (Eds.), **An invitation to cognitive science**. Cambridge, MA: MIT Press.

Spitzer, H., Desimone, R., and Moran, J. (1988). Increased attention enhance both behavioral and neuronal performance. *Science*, **240**, 338–340.

Werbos, P. (1974). *Beyond regression: New tools for prediction and analysis in the behavioral sciences,* PhD Thesis, Harvard University, Cambridge, MA.

Willshaw, D.J. and Malsburg, C. von der (1976). How patterned neural connections can be set up by self-organization. *Proceedings of the Royal Society of London (B)*, **194**, 431–445.

Zadeh, L. (1965). Fuzzy sets. *Information Control*, **8**, 338–353.

FIGURE CAPTIONS

Figure 1. Many-to-one learning combines categorization of many exemplars into one category, and labelling of many categories with the same name.

Figure 2. One-to-many learning enables one input vector to be associated with many output vectors. If the system predicts an output that is disconfirmed at a given stage of learning, the predictive error drives a memory search for a new category to associate with the new prediction, without degrading its previous knowledge about the input vector.

Figure 3. Interactions between the attentional and orienting subsystems of an adaptive resonance theory (ART) circuit: Level F_1 encodes a distributed representation of an event to be recognized via a short-term memory (STM) activation pattern across a network of feature detectors. Level F_2 encodes the event to be recognized using a more compressed STM representation of the F_1 pattern. Learning of these recognition codes takes place at the long-term memory (LTM) traces within the bottom-up and top-down pathways between levels F_1 and F_2. The top-down pathways can read-out learned expectations whose prototypes are matched against bottom-up input patterns at F_1. Mismatches in response to novel events activate the orientation subsystem $\mathcal{A}$, thereby resetting the recognition codes that are active in STM at F_2 and initiating a memory search for a more appropriate recognition code. Output from subsystem $\mathcal{A}$ can also trigger an orienting response. (a) Block diagram of circuit. (b) Individual pathways of circuit, including the input level F_0 that generates inputs to level F_1. The gain control input to level F_1 helps to instantiate the 2/3 Rule (see text). Gain control to level F_2 is needed to instate a category in STM.

Figure 4. ART search for an F_2 recognition code: (a) The input pattern **I** generates the specific STM activity pattern **X** at F_1 as it nonspecifically activates the orienting subsystem $\mathcal{A}$. **X** is represented by the hatched pattern across F_1. Pattern **X** both inhibits $\mathcal{A}$ and generates the output pattern **S**. Pattern **S** is transformed by the LTM traces into the input pattern **T**, which activates the STM pattern **Y** across F_2. (b) Pattern **Y** generates the top-down output pattern **U** which is transformed into the prototype pattern **V**. If **V** mismatches **I** at F_1, then a new STM activity pattern $\mathbf{X}^*$ is generated at F_1. $\mathbf{X}^*$ is represented by the hatched pattern. Inactive nodes corresponding to **X** are unhatched. The reduction in total STM activity which occurs when **X** is transformed into $\mathbf{X}^*$ causes a decrease in the total inhibition from F_1 to $\mathcal{A}$. (c) If the vigilance criterion fails to be met, $\mathcal{A}$ releases a nonspecific arousal wave to F_2, which resets the STM pattern **Y** at F_2. (d) After **Y** is inhibited, its top-down prototype signal is eliminated, and **X** can be reinstated at F_1. Enduring traces of the prior reset lead **X** to activate a different STM pattern $\mathbf{Y}^*$ at F_2. If the top-down prototype due to $\mathbf{Y}^*$ also mismatches **I** at F_1, then the search for an appropriate F_2 code continues until a more appropriate F_2 representation is selected. Then an attentive resonance develops and learning of the attended data is initiated.

Figure 5. Fuzzy ARTMAP architecture. The ART_a complement coding preprocessor transforms the input vector $\mathbf{a}$ into the vector $\mathbf{A} = (\mathbf{a}, \mathbf{a}^c)$ at the ART_a field F_0^a, where $\mathbf{a}^c = (1, 1, ..., 1) - \mathbf{a}$. $\mathbf{A}$ is the input vector to the ART_a field F_1^a. Similarly, the input to F_1^b is the vector $(\mathbf{b}, \mathbf{b}^c)$. When a prediction by ART_a is disconfirmed at ART_b, inhibition of map field activation induces the match tracking process. Match tracking raises the ART_a vigilance ρ_a to just above the F_1^a to F_0^a match ratio $|\mathbf{x}^a|/|\mathbf{A}|$, between the number $|\mathbf{x}^a|$ of active F_1^b nodes and the number $|\mathbf{A}|$ of active input features. This triggers an ART_a search which leads to activation of either an ART_a category that correctly predicts $\mathbf{b}$ or to a previously uncommitted ART_a category node.

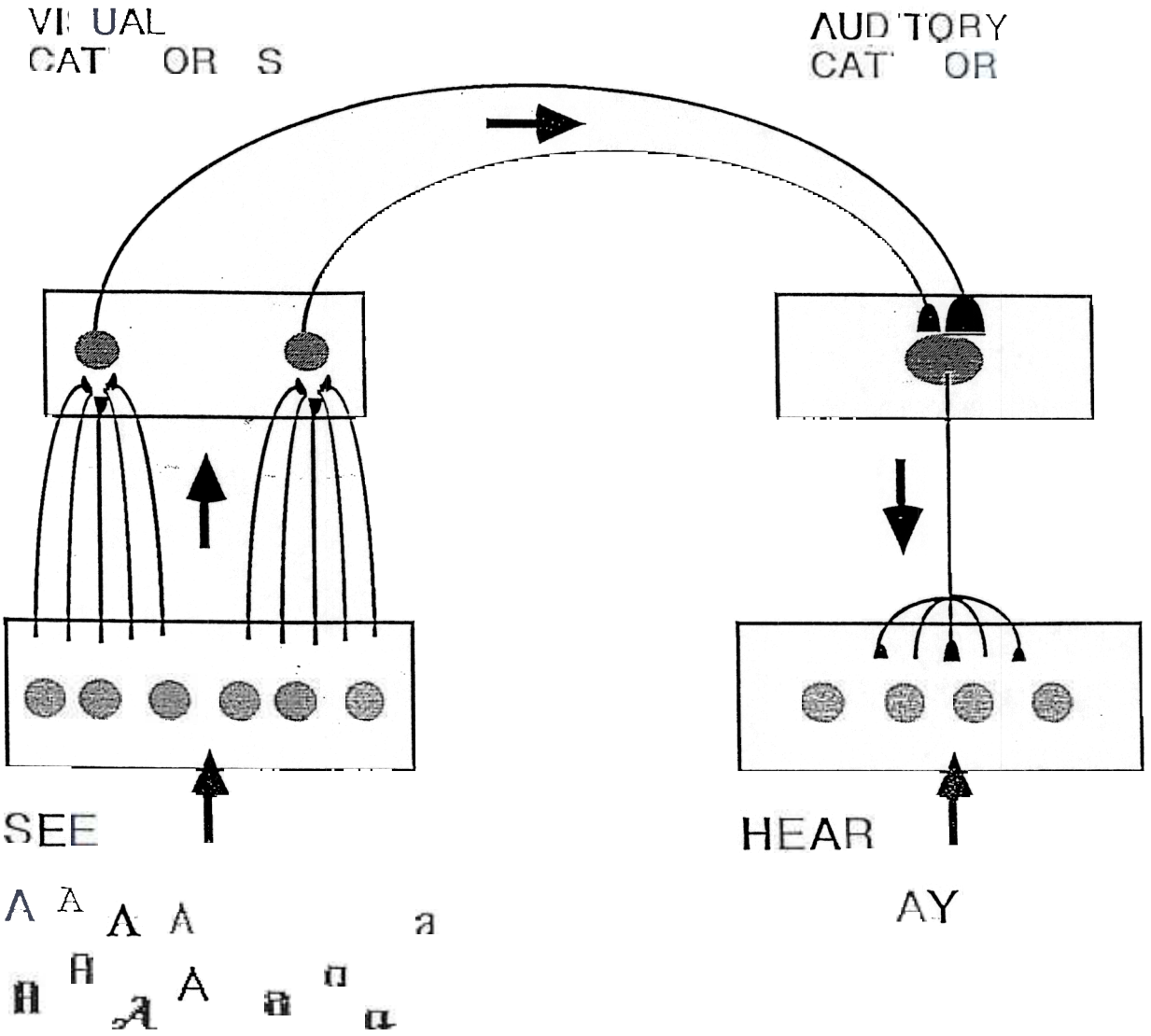
Figure 6. Comparison of ART 1 and Fuzzy ART.

Figure 7. Complement coding uses on-cell and off-cell pairs to normalize input vectors.

Figure 8. Fuzzy AND (or MIN) and Fuzzy OR (or MAX) operations generate category hyper-rectangles.

Figure 9. Illustrative letter fonts used by Frey and Slate (1991).

MANY TO ONE MAP



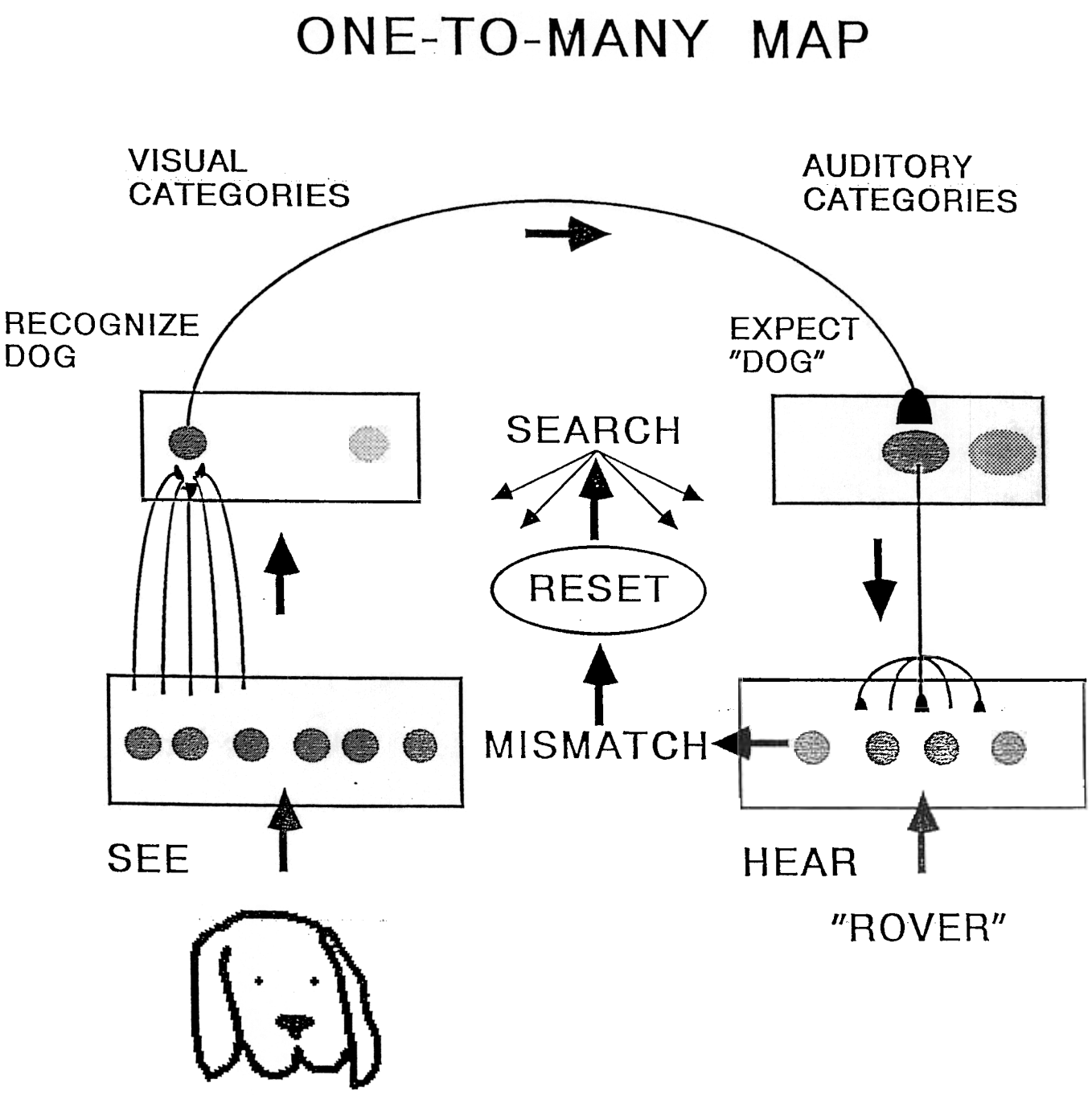

ONE-TO-MANY MAP
VISUAL CATEGORIES
AUDITORY CATEGORIES
RECOGNIZE DOG
EXPECT "DOG"
SEARCH
RESET
MISMATCH
SEE
HEAR
"ROVER"

Figure 2

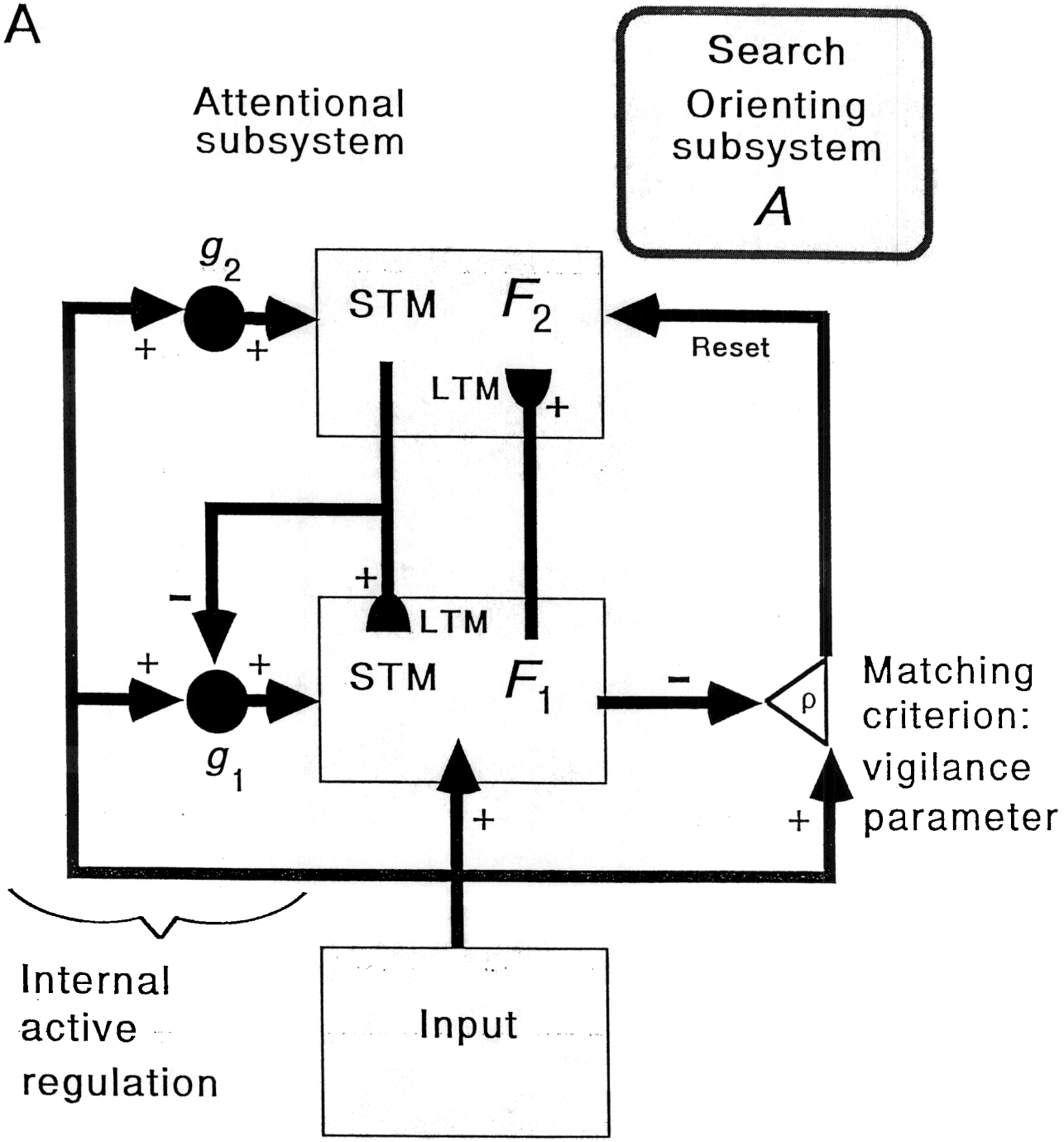
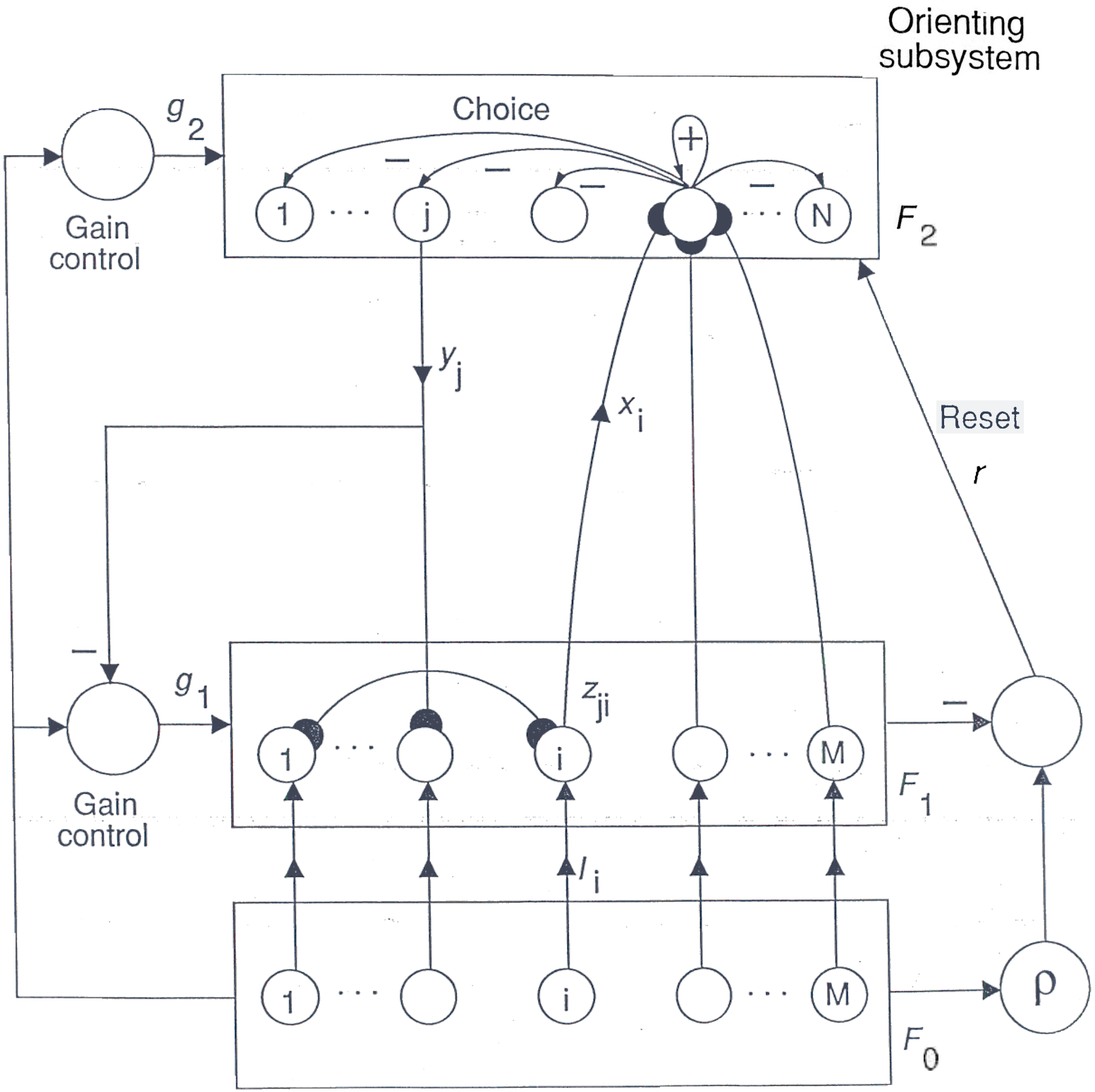
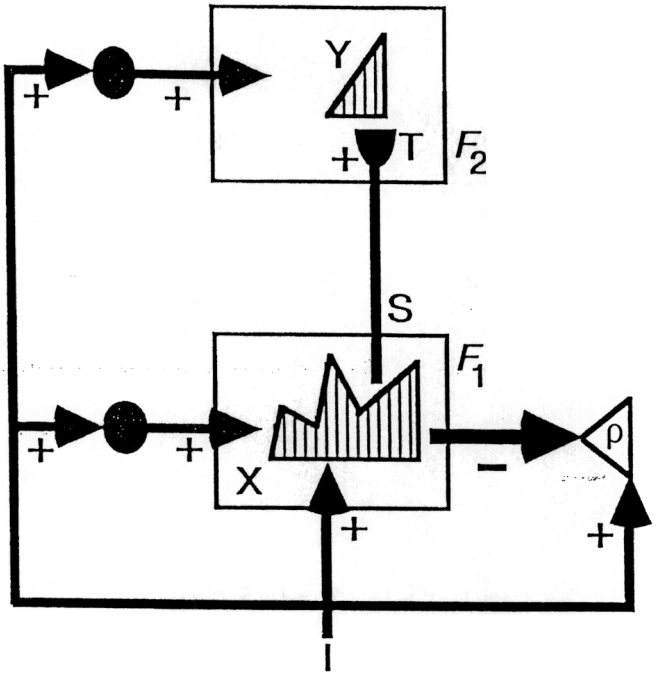

A
Attentional subsystem
Search
Orienting subsystem
A
g_2
STM F_2
LTM
Reset
LTM
STM F_1
g_1
ρ
Matching criterion: vigilance parameter
Internal active regulation
Input

Figure 3a

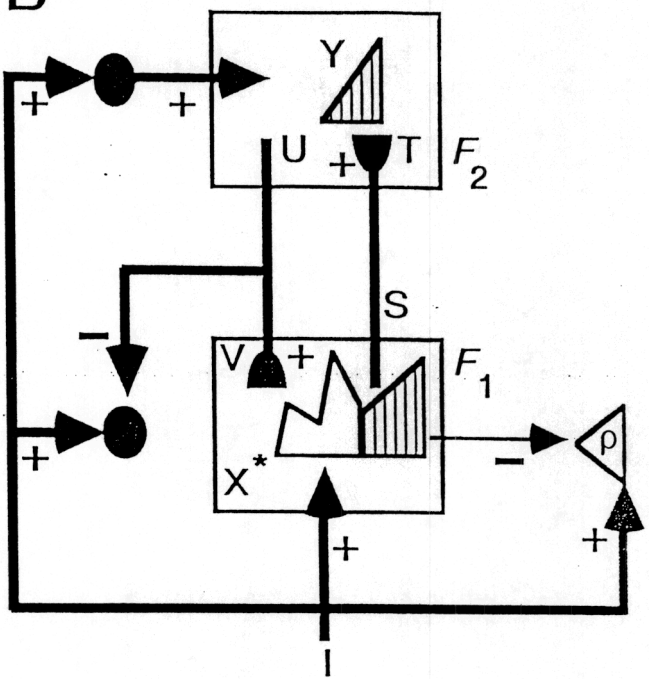
B

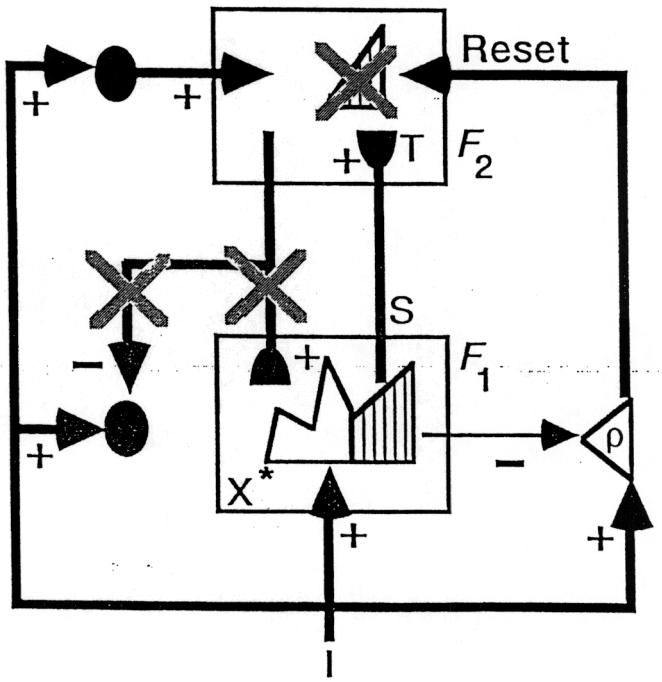

Orienting subsystem
Choice
g_2
Gain control
1
j
N
F_2
y_j
x_i
Reset
r
g_1
z_{ji}
1
i
M
F_1
Gain control
I_i
1
i
M
F_0
ρ

Figure

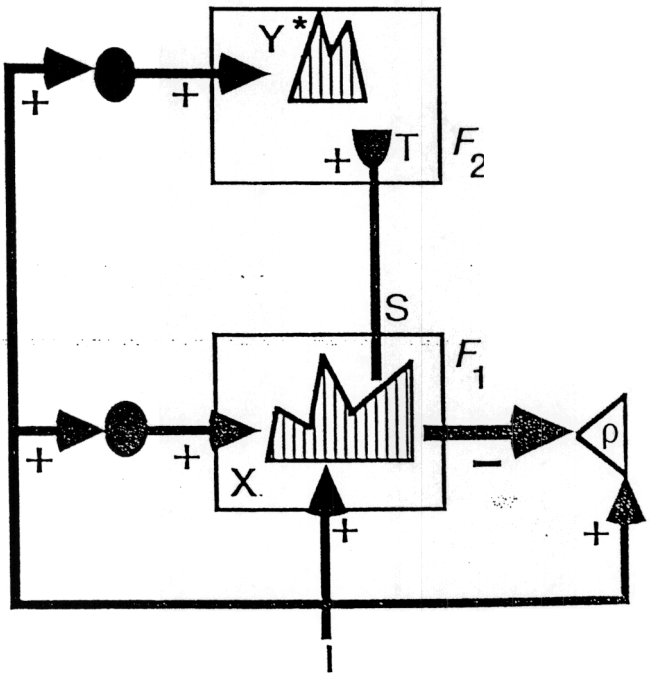


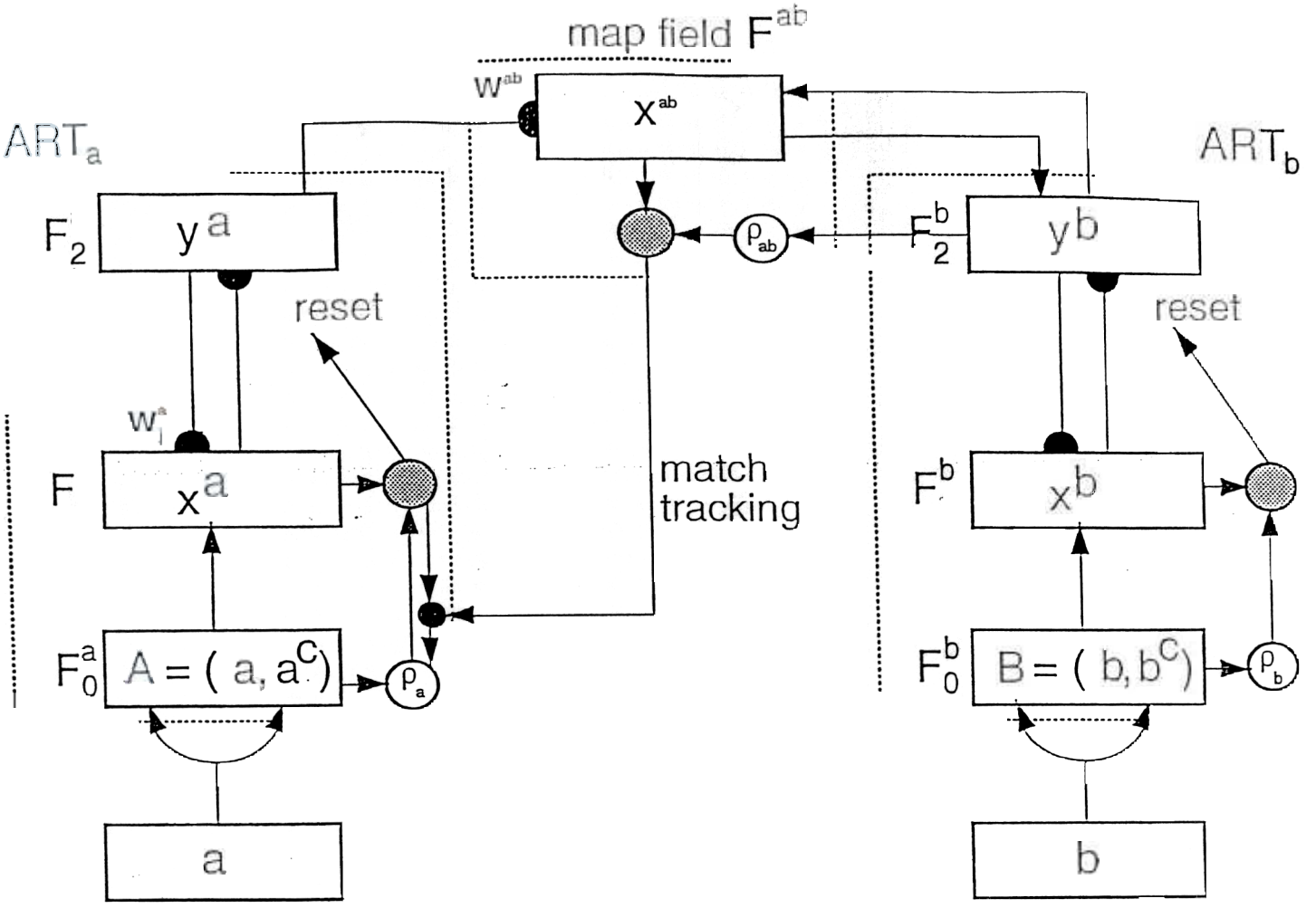
B





D




map field F^{ab}
w^{ab}
x^{ab}
ART_a
ART_b
F_2 y^a
ρ_{ab}
F_2^b y^b
reset
reset
w_i^a
F x^a
match
tracking
F^b x^b
F_0^a $A = (a, a^c)$
ρ_a
F_0^b $B = (b, b^c)$
ρ_b
a
b

Figure

ART 1 (BINARY)	FUZZY ART (ANALOG)

CATEGORY CHOICE

$$T_j \quad \frac{|\mathbf{I} \cap \mathbf{w}_j|}{\alpha + |\mathbf{w}_j|} \qquad T_j = \frac{|\mathbf{I} \wedge \mathbf{w}_j|}{\alpha + |\mathbf{w}_j|}$$

MATCH CRITERION

$$\frac{|\mathbf{I} \cap \mathbf{w}|}{|\mathbf{I}|} \geq \rho \qquad \frac{|\mathbf{I} \wedge \mathbf{w}|}{|\mathbf{I}|} \geq \rho$$

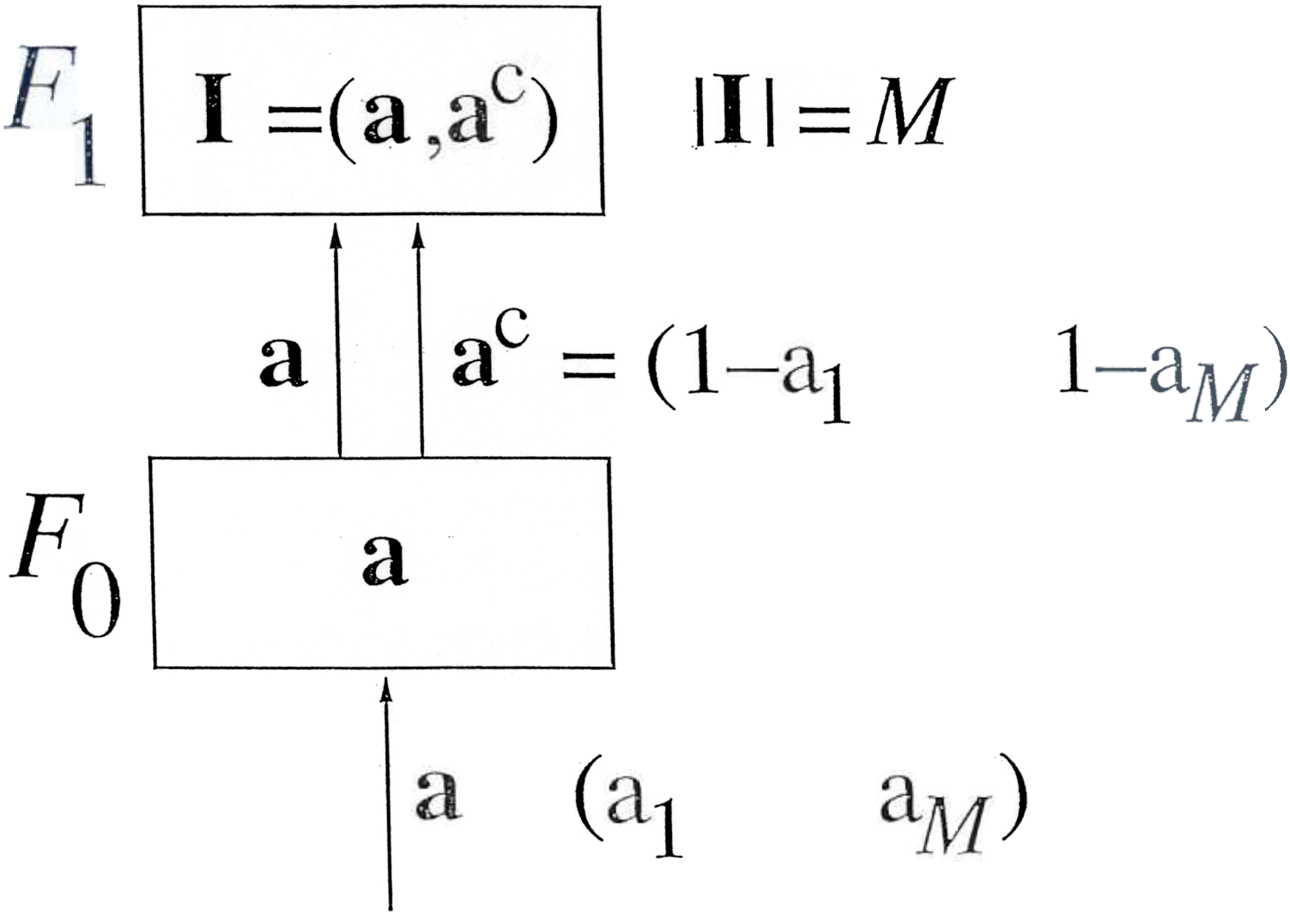
FAST LEARNING

$$\mathbf{w}_j^{(\text{new})} \quad \mathbf{I} \cap \mathbf{w}_j^{(\text{old})} \qquad \mathbf{w}_j^{(\text{new})} \quad \mathbf{I} \wedge \mathbf{w}_j^{(\text{old})}$$

$\cap$ = logical AND intersection

$\wedge$ = fuzzy AND minimum

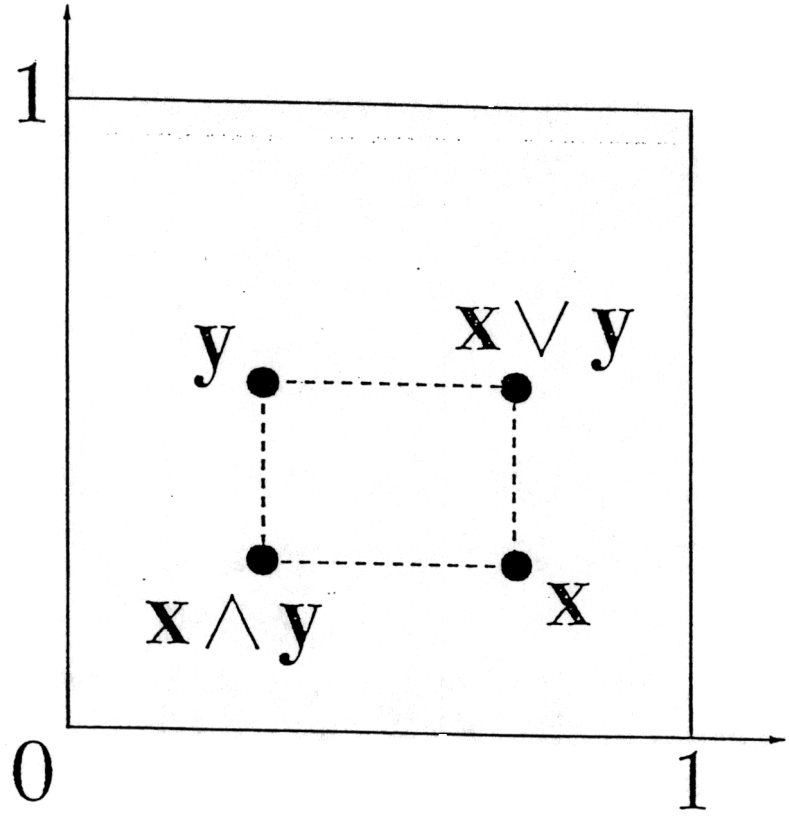
Figure 6


F_1
$\mathbf{I} = (\mathbf{a}, \mathbf{a}^c)$
$|\mathbf{I}| = M$
$\mathbf{a}$
$\mathbf{a}^c = (1-a_1 \quad 1-a_M)$
F_0
$\mathbf{a}$
$\mathbf{a}$
$(a_1 \quad a_M)$

Figure

$\wedge$ Fuzzy AND (conjunction)

$\vee$ Fuzzy OR (disjunction)



$\mathbf{x}$ $(x \quad x_2)$ $\mathbf{y}$ $(y \quad y_2)$

$(\mathbf{x} \wedge \mathbf{y})$ $\min(x \quad y)$ $(\mathbf{x} \wedge \mathbf{y})_2$ $\min(x_2, y_2)$

$\mathbf{x} \vee \mathbf{y}$ $\max(x \quad y)$ $(\mathbf{x} \vee \mathbf{y})_2$ $\max(x_2 \quad y_2$

Figure

TABLE CAPTIONS

Table 1. Some machine learning benchmark studies (Carpenter, Grossberg, and Reynolds, 1991; Carpenter, Grossberg, Markuzon, Reynolds, and Rosen, 1992) which compare the performance of supervised ART, or ARTMAP, models with that of alternative models. These benchmarks describe how well these systems predict test sets when they experience equivalent training sets (as in benchmarks 1–4) and the number of epochs, or repetitions of the training set, that are needed to reach the same level of accuracy (benchmark 5).

Table 2. On-line learning and performance in forced choice ($\overline{\rho_a} = 0$) or conservative ($\overline{\rho_a} = 0.7$) cases, with replacement or no replacement of samples after training.

Table 3. Off-line forced choice ($\overline{\rho_a} = 0$) ARTMAP system performance after training on input sets ranging in size from 3 to 4000 exemplars. Each line shows average correct and incorrect test set predictions over 10 independent simulations, plus the range of learned ART_a category numbers.

Table 4. Off-line conservative ($\overline{\rho_a} = 0.7$) ARTMAP system performance after training on input sets ranging in size from 3 to 4000 exemplars. Each line shows average correct, incorrect, and no-response test set predictions over 10 independent simulations, plus the range of learned ART_a category numbers.

ARTMAP BENCHMARK STUDIES

Medical database - mortality following coronary bypass grafting (CABG) surgery

Fuzzy ARTMAP significantly outperforms:

- Logistic regression
- Additive model
- Bayesian assignment
- Cluster analysis
- Classification and regression trees
- Expert panel-derived sickness scores
- Principal component analysis

2. Mushroom database

   Decision trees (90-95% correct)

   ARTMAP (100% correct; training set an order of magnitude smaller)

3. Letter recognition database

   Genetic algorithm (82% correct)

   Fuzzy ARTMAP (96% correct)

4. Circle-in-the-Square task

   Back propagation (90% correct)

   Fuzzy ARTMAP (99.5% correct)

5. Two-Spiral task

   Back propagation (10,000 - 20,000 training epochs)

   Fuzzy ARTMAP (1-5 training epochs)

Table 1

TABLE 2: On-Line Learning

	Average number of correct predictions on previous 100 trials			
Trial	$\overline{\rho_a} = 0$ no replace	$\rho_a = 0$ replace	$\overline{\rho_a} = 0.7$ no replace	$\overline{\rho_a}$ 0.7 replace
100	82.9	81.9	66.4	67.3
200	89.8	89.6	87.8	87.4
300	94.9	92.6	94.1	93.2
400	95.7	95.9	96.8	95.8
500	97.8	97.1	97.5	97.8
600	98.4	98.2	98.1	98.2
700	97.7	97.9	98.1	99.0
800	98.1	97.7	99.0	99.0
900	98.3	98.6	99.2	99.0
1000	98.9	98.5	99.4	99.0
1100	98.7	98.9	99.2	99.7
1200	99.6	99.1	99.5	99.5
1300	99.3	98.8	99.8	99.8
1400	99.7	99.4	99.5	99.8
1500	99.5	99.0	99.7	99.6
1600	99.4	99.6	99.7	99.8
1700	98.9	99.3	99.8	99.8
1800	99.5	99.2	99.8	99.9
1900	99.8	99.9	99.9	99.9
2000	99.8	99.8	99.8	99.8

TABLE 3: Off-Line Forced-Choice Learning

Training Set Size	Average % Correct (Test Set)	Average % Incorrect (Test Set)	Number of ART_a Categories
3	65.8	34.2	1–3
5	73.1	26.9	1–5
15	81.6	18.4	2–4
30	87.6	12.4	4–6
60	89.4	10.6	4–10
125	95.6	4.4	5–14
250	97.8	2.2	8–14
500	98.4	1.6	9–22
1000	99.8	0.2	7–18
2000	99.96	0.04	10–16
4000	100	0	11–22

TABLE 4: Off-Line Conservative Learning

Training Set Size	Average % Correct (Test Set)	Average % Incorrect (Test Set)	Average % No-Response (Test Set)	Number of ART_a Categories
3	25.6	0.6	73.8	2–3
5	41.1	0.4	58.5	3–5
15	57.6	1.1	41.3	8–10
30	62.3	0.9	36.8	14–18
60	78.5	0.8	20.8	21–27
125	83.1	0.7	16.1	33–37
250	92.7	0.3	7.0	42–51
500	97.7	0.1	2.1	48–64
1000	99.4	0.04	0.5	53–66
2000	100.0	0.00	0.05	54–69
4000	100.0	0.00	0.02	61–73