The Self-Organization of Spatially Invariant Representations

Randall C. O'Reilly
&
James L. McClelland

Technical Report PDP.CNS.92.5
December 1992

Parallel Distributed Processing and Cognitive Neuroscience

Department of Psychology
Carnegie Mellon University
Pittsburgh, PA

Western Psychiatric Institute and Clinic
University of Pittsburgh
Pittsburgh, PA

Neural and Behavioral Sciences
University of Southern California
Los Angeles, CA

MRC Applied Psychology Unit
Cambridge, England

Abstract

The problem of computing object-based visual representations can be construed as the development of invariancies to visual dimensions irrelevant for object identity. This view, when implemented in a neural network, suggests a different set of algorithms for computing object-based visual representations than the "traditional" approach pioneered by Marr (1982). A biologically plausible self-organizing neural network model that develops spatially invariant representations is presented. There are four features of the self-organizing algorithm that contribute to the development of spatially invariant representations: temporal continuity of environmental stimuli, hysteresis of the activation state (via recurrent activation loops and lateral inhibition in an interactive network), Hebbian learning, and a split pathway between "what" and "where" representations. These constraints are tested with a backprop network, which allows for the evaluation of the individual contributions of each constraint on the development of spatially invariant representations. Subsequently, a complete model embodying a modified Hebbian learning rule and interactive connectivity is developed from biological and computational considerations. The activational stability and weight function maximization properties of this interactive network are analyzed using a Lyapunov function approach. The model is tested first on the same simple stimuli used in the backprop simulation, and then with a more complex environment consisting of right and left diagonal lines. The results indicate that the hypothesized constraints, implemented in a Hebbian network, were capable of producing spatially invariant representations. Further, evidence for the gradual integration of both featural complexity and spatial invariance over increasing layers in the network, thought to be important for real-world applications, was obtained. As the approach is generalizable to other dimensions such as orientation and size, it could provide the basis of a more complete biologically plausible object recognition system. Indeed, this work forms the basis of a recent model of object recognition in the domestic chick (O'Reilly & Johnson, Submitted).

Contents

Introduction **2**

Network Properties That Produce Spatially Invariant Representations **5**

Simulation 1: Testing the effects of the four factors **9**

Methods 9

Results 12

Discussion 13

Modified Hebbian Learning Mechanism **14**

Computation Performed by Hebbian Learning 15

Problems With the Oja Rule 17

Strong Constraints From Neurobiological Data 18

A Pattern-Covariance Alternative 18

Zero-Sum Interpretation 21

Dual Mechanisms at the Synapse 25

Activation Function And Network Convergence 26

Weight Update Rule Maximization Properties 29

Simulation 2 **31**

Methods 31

Results 32

Discussion 34

Simulation 3 **36**

Methods 36

Results 37

Discussion 40

Conclusions **40**

Analysis of Goodness Function **42**

References **44**

Introduction

Object-based visual representations are those which allow us to visually recognize different objects. While most people take it for granted that they can tell what something is just by looking at it, those who have attempted to reproduce this ability in artificial systems have made us realize that the problem is much more difficult than it seems. This disparity between effortless perception on the one hand and the complicated, computationally intensive solutions suggested by researchers such as Marr and Biederman on the other leads one to wonder if there might be a different way of looking at the problem that would make some sense of the apparent ease with which we can see objects. Recent advances in self-organizing neural network models (e.g. Linsker, 1986; Miller, 1990a; Marshall, 1990; Földiák, 1991) have demonstrated that it is possible for a network to automatically develop many different types of representations useful for visual processing. These models hold out the promise that perhaps there is an "effortless" (i.e., self-organizing) solution to the problem of visual object recognition.

One of the principal reasons for the difficulty of the problem is also a clue to a potential solution: there are a very large (practically infinite) number of different images that a given object can project on to the retina. Deciphering which of the many thousands of familiar objects a given image represents is difficult because of this many-to-many correspondence. However, the ways in which a given object can produce different images on the retina are limited to a few dimensions of variability. This suggests that one could focus on eliminating the systematic variability due to these dimensions as a method of computing object-based representations.

These dimensions of variability arise principally from the projection of three-dimensional objects onto our two-dimensional retinas. Thus, the image produced by a given object can appear in a different location, orientation, and size depending on where it is located relative to our eyes. There are other dimensions of variability resulting from different lighting conditions, and from changes in the shape of the object itself. Thus, an object representation must be invariant with respect to all those dimensions on which the image can vary and still represent the object, while at the same time being selective enough to distinguish between different objects. Therefore, one can re-phrase the object-recognition problem as that of producing representations which exhibit invariance under transformations along the above-mentioned dimensions (spatial location, orientation, size, etc.).

In order to compare the kinds of algorithms suggested by the invariance approach to object recognition with the more direct computational approaches, we will briefly consider the work of investigators such as Marr (1982) and Biederman (1987), which are typical of a symbolic approach to computational modeling. The general scheme they employ is to apply several mathematical transformations on the output of sophisticated retinal image pre-processing systems (e.g., Marr's 3-D model), and arrive at a parameterization of that image which uniquely identifies the object based on its geometrical properties. Both Marr and Biederman suggest the use of generalized cylinders (Biederman calls them *Geons*), which are parameterized by the axis and the shape of the cross-section along the axis. The specific values of these parameters for a given image can then be used to search a lookup table which will identify the corresponding object for those parameters. The principal difference between Marr and Biederman's approach is that Marr applies the generalized cylinder transformation to an unlabeled, three-dimensional model reconstructed from the 2-D retinal image data, while Biederman uses the non-accidental properties of the retinal image to identify which of several standardized Geon components exist in the image. However, the basic strategy underlying these two approaches is one of parameterization and lookup tables.

In contrast to these approaches, there is another category which involve the use of neural networks to perform the transformation between image and object. Many researchers (e.g. Fukushima, 1988; Mozer, 1987; Zemel et al., 1989; Sandon & Uhr, 1988; Hinton, 1981) have proposed such models, which all share several important features, and can be thought of as employing the same basic algorithm. The essential character of this algorithm is the "divide and conquer" approach where the image of an object is transformed for the purposes of invariance and identification in either combined or intertwined stages of processing. In contrast to the symbolic approaches outlined above, these models focus on the problem of invariance to spatial variability instead of the parameterization of an object according to geometrical primitives. This shift in emphasis results from the fact that neural networks can easily identify spatially invariant image patterns without the need to parameterize these images according to explicit geometrical models. While any approach must, by logical necessity, arrive at some form of canonical representation, the means to this end can be quite different.

Figure 1 illustrates the "divide and conquer" approach with respect to spatial invariance. This algorithm has been most directly implemented in Mozer's BLIRNET, which performs spatially invariant recognition of words:

> BLIRNET's architecture consists of a hierarchy of processing levels, starting at the lowest level with location-specific detectors for primitive visual features—the retinal representation—and progressing to a level composed of location-independent detectors for abstract letter identities. Units at intervening levels register successively higher order features over increasingly larger regions of retinotopic space. The effect of this architecture is that both location invariance and featural complexity increase at higher levels of the system. (p. 99 Mozer & Behrmann, 1990)

Thus, as is illustrated in the figure[1] this algorithm produces invariant representations by gradually encompassing more and more of the retina in a single representation, and simultaneously endowing these representations with greater internal complexity in terms of the number of features they contain.

At the highest layer in BLIRNET, there are representations for letter-triples in any spatial location. These representations encode the presence of any three letters (including the

[1] Note that this figure shows the representation of a single letter, while BLIRNET goes all the way to complete words at the highest level

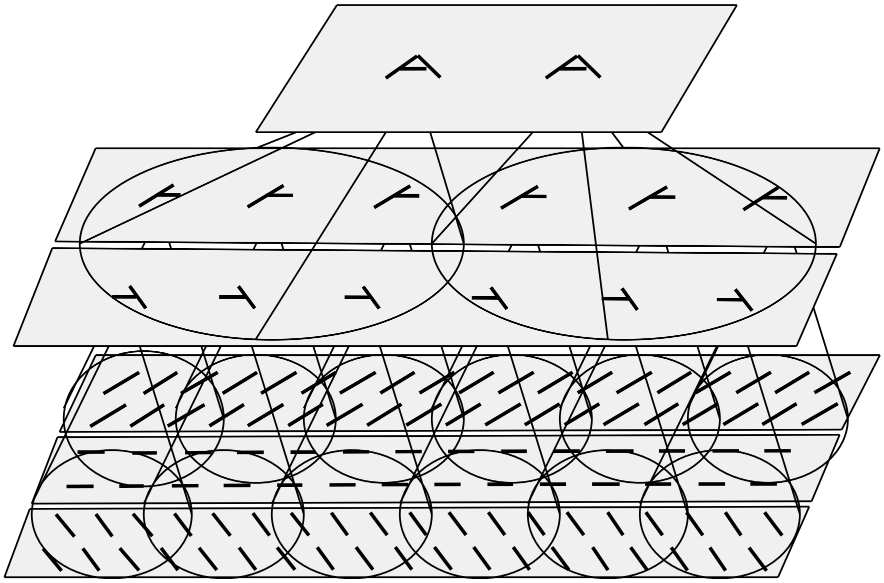


Figure 1: The “Divide and Conquer” Algorithm for producing spatially invariant representations, illustrated with the letter “A”. The first layer contains a retinotopic feature-map similar to that found in the first layers of visual cortex. The second layer integrates over both features and locations, resulting in partially invariant, more complex representations in layer 2. Finally, in layer 3, these representations are integrated over a larger area of spatial invariance, and further complexity as well.

possibility of a single “wild card” letter) in a particular sequence. Thus, the representations encode local relationships between letters, but not the global position of these letters on the retina. Because of this representational scheme, words that share many letters will not be confused unless they also have these letters in the same order. Mozer & Behrmann (1990) give an example of the high-level representations would be activated by the presentation of the word MONEY: “**M, **_O, *MO, *_ON, *M_N, MON, M_NE, MO_E, ONE, O_EY, ON_Y, NEY, NE_*, N_Y*, EY*, E_**, and Y**” (p. 99). Because of the multiplicity of these three-letter sequences activated by the input, the resulting pattern will be distinct for each word, and it is this pattern which is used to identify which word is actually being seen.

Because the invariance transformations (incorporating larger regions of the retina) are combined with the identification transformations (incorporating more and more complex features), it is possible to recognize objects based on the spatial relationships among the features, while at the same time remaining invariant over the position of the entire object on the retina. In contrast, it is difficult to imagine the symbolic approaches having this capability, since they perform the invariance transformation and the identification transformation separately. In addition, the parallelism and distributed nature of the neural network representation offers all the reliability, generalization, and graceful degradation properties typical of these kinds of models. An added advantage over the symbolic approach is the biological realism of the algorithm—it is well established that lower visual cortex employs a massive “divide and conquer” approach with millions of neurons encoding edges at each location in the retina, and others encoding only edges at a certain orientation, etc (e.g. Hubel & Wiesel, 1962).

However, there are several drawbacks to the specific architecture of BLIRNET. Primary among them is that it was designed and hand-wired for a lexical environment. The network does not learn these invariant representations on its own, and this limits the generality of the model, both in terms of the variety of visual stimuli that it can represent in a spatially invariant manner, and also in variety of invariance transformations that can be performed (e.g., rotational invariance in both 2 and 3 dimensions, and size invariance). Other drawbacks include the absence of top-down connections among units, which would allow for attentional and other effects that depend on such connections (see Phaf et al., 1990; McClelland, 1981; Rumelhart, 1982 for a discussion, and Mozer, 1988 for a different way of implementing spatial attention in his network).

As for the other object recognition networks cited above, both the Sandon & Uhr (1988) and Hinton (1981) networks are hand-wired. The *Neocognitron* model (Fukushima, 1988), even though it does possess some degree of self-organizing capability, has an elaborate prespecified architecture that does most of the work. Also, the learning mechanism is quite complicated, biologically implausible, and does not appear to work well unless given explicit teaching signals, and the architecture requires a compromise between invariance and identification (Barnard & Casasent, 1990).

The model presented in this paper attempts to overcome some of these limitations by showing how spatially invariant representations of the kind used in BLIRNET can develop naturally from an interactive Hebbian network combined with certain environmental constraints. These constraints are general enough that they would be adaptable to other forms of invariance, including rotational and size invariance. However, the research presented in this paper focuses on spatial invariance because the kinds of representations needed for this kind of transformation have been discussed and tested previously (in BLIRNET and the other networks cited above), and because these representations appear more straightforward than the other kinds of invariance. Future work will be needed to see if this approach does indeed generalize to other dimensions.

Network Properties That Produce Spatially Invariant Representations

There are two principal types of learning in neural networks: supervised and unsupervised. Supervised learning involves training a network to perform a mapping between many different input/output patterns. Unsupervised or self-organizing networks instead use algorithms that capture regularities in the input stimuli to develop representations without supervision. It is these self-organizing networks that are most appropriate for learning invariant representations, as invariance is a form of regularity that might be present in the visual environment. Also, self-organizing networks are more realistic for studying the brain because they do not require the use of the "teaching" stimulus needed by supervised algorithms, the existence of which in the brain is an unlikely assumption.

Most self-organizing networks use a variant of the Hebbian associative learning algorithm (e.g. Hebb, 1949; Rumelhart & Zipser, 1986; Bienenstock et al., 1982; Linsker, 1986; Miller, 1990a; Marshall, 1990), which modifies the weight between two units in proportion to the product of their activations, yielding a correlation-style[2] learning. Thus, the regularities in the environment that can be used by this type of learning algorithm are those which can be expressed in terms of a correlation between the activities of different units. Is it possible to find such environmental regularities that would enable a Hebbian network to develop spatially invariant representations?

One possibility is to use the continuity of a given object with itself over time as an environmental regularity. Since a given object will be either at rest or in continuous motion relative to the observer, it will produce a series of images on the observer's retina that all have one thing in common: the object itself. If the object is moving relative to the observer, or the observer's eyes are saccading around the object, then the series of images will have this regularity of the presence of the object, while at the same time differing in the position of the object on the retina. In theory, the object-regularity could be extracted from the series of images, resulting in a spatially invariant representation. This idea has recently been suggested by Földiák (1991), although we had independently come up with the same idea at around the same time. Our approach differs from his in that we emphasize biologically plausible mechanisms, and we directly address the multi-layer algorithm that is essential for any practical implementation of this idea.

In order for a self-organizing neural network to "recognize" the temporal regularity of an object, it would need to maintain temporal continuity of the activations in the set of units coding for the object. These units would remain active while the image of the object appeared in different locations on the retina. With a correlation-based learning rule as discussed above, these active units would come to be correlated with the different activity patterns lower in the visual system produced by the object, resulting in the desired many-to-one mapping between the various images of the object and the invariant representation of the object. Thus, the environmental fact of the temporal continuity of objects would be mirrored by internal representations that possess this quality as well, and the two would be linked by a Hebbian associative learning rule.

While it may sound as though this kind of mechanism requires an object to have been seen previously in every possible retinal position before it can be recognized in that position, this is not the case if the network learns to perform the invariance transformation in conjunction with the object-identification transformations. To the extent that the object-identification transformations result in the representation of an object in terms of visual subregularities ("features") shared by many different objects, one only needs to learn the invariant pattern of such features for each object once, since the system will already know how to transform these features into an invariant representation from any given retinal position. For the BLIRNET system, these features are letters and letter-triples, which the system has encoded in all possible locations. Any novel word will simply activate a different pattern of these features, which the system already knows how to represent in a spatially invariant manner. The

[2] It is not a true correlation unless the units have a mean activation of 0

proposed learning algorithm will operate gradually over several layers, such that a gradient of invariance and featural integration will develop, corresponding to the weakening level of influence the retinal changes have on increasing layers of the network, as is depicted in Figure 2b.

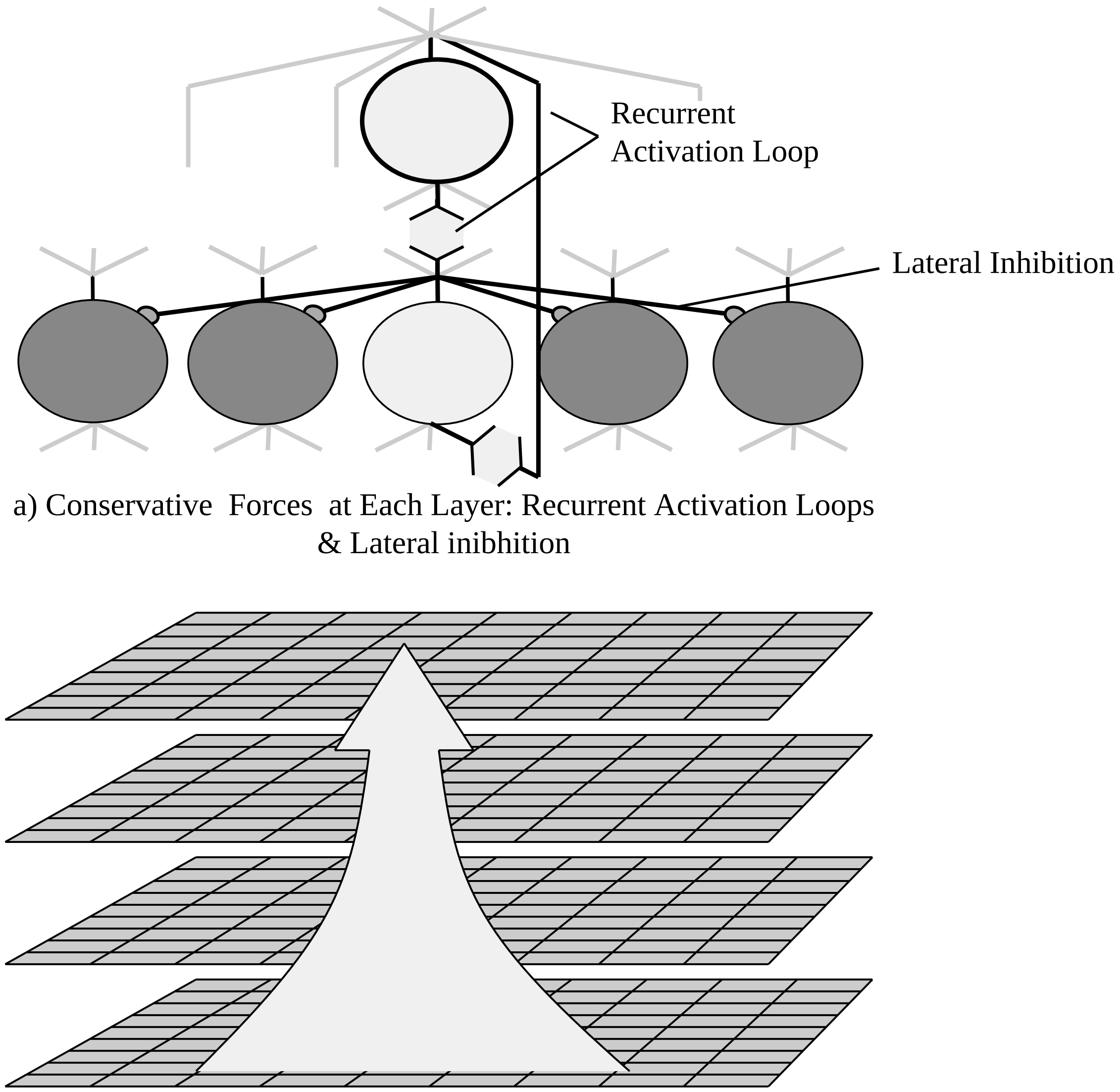
In summary, the following environmental and network properties form a system which should be capable of producing spatially invariant representations:

1. **Temporal Continuity of Environmental Stimuli:** The continued existence of objects in the world is an assumed property of the environment. The organism must direct its visual attention towards objects for contiguous periods of time in order to capitalize upon this property. The direct incorporation of environmental constraints in this model is reminiscent of Gibson's (1979) perceptual theories, and Anderson's (1990) rational analysis perspective on the mind.

2. **Hysteresis in the Activation State of Object Units:** Hysteresis, which is the incorporation of previous states of a variable into subsequent ones, mirrors the continuity of environmental stimuli. In interactive neural networks, hysteresis is the result of recurrent activation loops and lateral inhibition. Specifically, a set of units which are mutually interconnected with excitatory connections will continue to send activation to each other if they have previously been excited, even if the input pattern which originally excited them is no longer active. Further, the active units in a pool of units connected by inhibitory weights will suppress the activity of the other units in the pool, resulting in the maintenance of the present pattern of activity. The mutual excitation will typically decay over time, or when another unit in the inhibitory pool becomes active, at which point the recurrent activation loop will be broken. See Figure 2a for an illustration. That these kinds of wiring patterns are found in the visual cortex is well established (e.g. Douglas & Martin, 1990).

3. **Hebbian Associative Learning:** works by strengthening the connections between concurrently active units. There is considerable evidence from the literature on LTP that this is the kind of learning being performed in many parts of the brain (e.g., Brown et al., 1989; Stanton & Sejnowski, 1989; Bear & Cooper, 1987; Artola et al., 1990)

4. **Split "What" and "Where" Pathways:** There is evidence that visual processing is physiologically divided along two broad categories of information corresponding to object identification and spatial location (see Ungerleider & Mishkin, 1982; Rueckl et al., 1989; O'Reilly et al., 1990 for details). Thus, the spatial location information lost in the spatially invariant representations of the object-based "what" system can be captured in the location-based "where" system. To the extent that both kinds of information are necessary for later processing in the brain, the existence of a "where" system might cause a "what" system to exhibit spatial invariance.



a) Conservative Forces at Each Layer: Recurrent Activation Loops & Lateral inibhition

b) Additive Effect of Conservative Forces Yields a Gradient of Resistance to Change over Increasing Layers.

Figure 2: **a)** Shows how a unit is connected to its neighbors by lateral, inhibitory connections, and to units in other layers by recurrent, excitatory connections. Both of these connections result in a preservation of the current pattern of activation by directly activating each other through the recurrent excitatory connections, and preventing other units from becoming active through the lateral inhibition. **b)** Illustrates how the cumulative effect of these conservative forces at each layer will result in a gradient of stability over increasing layers.

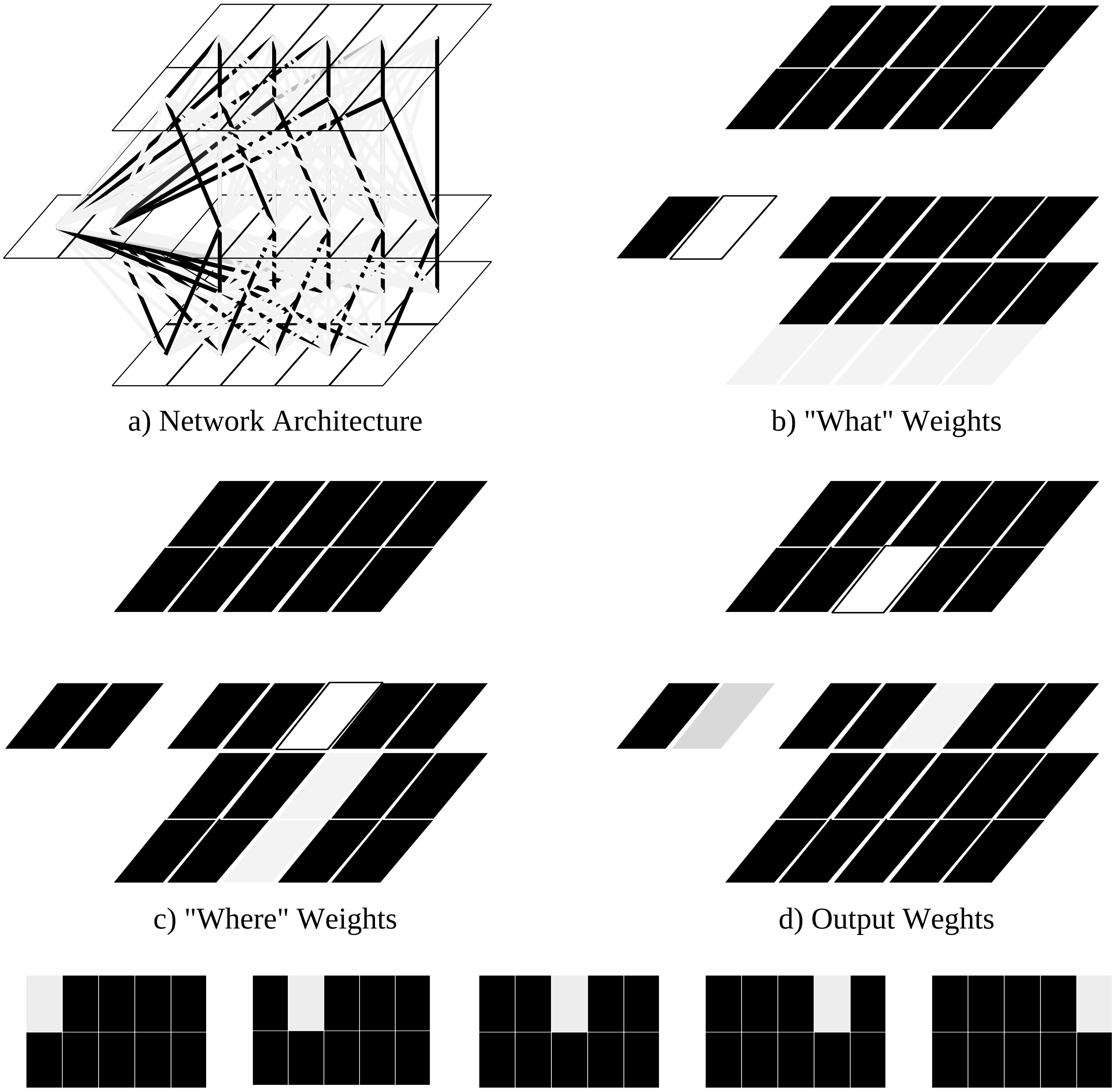
Simulation 1: Testing the effects of the four factors

The interaction of all four of the above-mentioned factors in concert should be capable of producing spatially invariant representations in the "what" pathway of a neural network. In order to test this hypothesis, a backprop network was constructed with the architecture shown in Figure 3. This network has split "what" and "where" pathways, and is given an auto-encoder task (the output is the same as the input) to ensure that these two pathways can capture all the information in the input. The stimuli presented to the network consisted of single positions of the $5x2$ input and output arrays being active ($a = 1.0$) with all other units inactive ($a = 0$). Thus, each unit represents one of two "objects" depending on which of the two rows it is in, and a different position of that object on the retina depending on which of the 5 columns it is in. The input representation is intended to be as simple as possible in order to eliminate any other confounding effects that might be introduced by trying to implement a more realistic model. The level of encoding at the input layer should be conceptualized as the output of lower visual areas which process direct retinal input and produce retinotopic feature representations. Thus, there would be distinct units active for distinct features in distinct positions, just as in the input layer of this model.

Methods

The experimental manipulation performed on the network described above consisted of selectively varying the presence of some of the proposed network properties that are hypothesized to create spatially invariant representations in the "what" pathway. These factors were implemented in the following ways:

Temporal Continuity of Environmental Stimuli: In order to test the "best-case" scenario, temporal continuity for a given stimulus was perfect, so that there was a 100% chance that the same object would appear in a different position in the next time step (except for the last position of that object, after which the other stimulus would be presented). The effects of lower same-object probabilities are investigated later. Thus, all of the stimuli for one row are shown in random order until all positions of that stimulus have been shown, then a series of blank stimuli are shown, allowing the network activations to relax before showing the next object. Without such a demarcation between objects, there is no way for this kind of network to distinguish between the different objects, and object-specific invariant representations do not develop. This blank period is possibly unrealistic, given that the images of objects do not conveniently disappear until the brain "settles," but is intended to summarize the probable effects of attentional mechanisms, which would modulate patterns of activation in object based representations (for evidence of such object-based attention, see Duncan, 1984). A switch in attention would effectively inhibit the current pattern of activation, having the same effect as the settling used in the model.



a) Network Architecture

b) "What" Weights

c) "Where" Weights

d) Output Weghts

e) Stimuli for "top row" unit, showing all locations.

Figure 3: **a)** The architecture of the backprop network, showing the input layer, which consists of two rows, each of which represents a different object, and five columns, each of which represents a different retinal position for a given object. The “what” pathway is on the left in the second layer, with two units, one for each object, and the “where” pathway is on the right, with five units, one for each location. **b)** Shows the weights for one of the “what” units in a trained network, where the weight is represented by the color of the unit to which it is connected, with lighter colors indicating stronger weights. **c)** Shows the weights for one of the “where” units. **d)** The output layer reproduces the stimulus presented on the input by combining the “what” and the “where” representations, as is shown here. **e)** The stimuli for one object in all positions (they were actually presented to the network in random order)

Hysteresis: Implemented by modifying the activation function used in the "what" pathway to include a portion of the previous activation as follows:

$$\begin{aligned} a_j(t+1) &= ha_j(t) + (1-h) f_{sigmoid}(I^{net}) && \text{(1a)} \\ f_{sigmoid}(I^{net}) &= \frac{1}{1+e^{-(I^{net})}} && \text{(1b)} \\ I^{net} &= \sum_i w_{ij} a_i && \text{(1c)} \end{aligned}$$

where Equation 1b is the standard backprop activation function[3] and Equation 1a simply weights the influence of the activation at the previous time step by the hysteresis factor h, which is typically 0.60. In addition to this hysteresis mechanism, the momentum factor on the weights contributes a degree of temporal continuity, in that weights moving in one direction will tend to continue in that direction. This implementation of hysteresis is neither biologically realistic nor computationally as effective as the more biologically plausible mechanism proposed previously (for reasons that will be explained below), but recurrent activation loops and lateral inhibition are not easily incorporated into the backpropagation framework.

Split "What" and "Where" Pathways: This was implemented by fixing the weights to the "where" pathway so that each "where" unit encoded the corresponding column of the input (strong weights to each of the two units in a given column, and 0 weights to all other units, see Figure 3c for an illustration of what these weights look like.)

Note that Hebbian associative learning has been replaced with the backpropagation learning rule. The hysteresis mechanism described above has a similar effect on the backprop learning scheme as it does in Hebbian learning—an increased activation value will increase the weight change to that unit.

Five combinations of these factors were run for 10 different networks each, with random initial weights. These conditions were selected to evaluate the impact of different influences on the creation of spatially invariant representations.

1. **Temporal continuity, Hysteresis, and Fixed "where" weights:** This condition tests the effect of the full complement of factors. It should produce the highest degree of spatially invariant representations.

2. **Temporal continuity and Fixed "where" weights:** When compared to Condition 1, this should indicate the differential importance of the hysteresis factor.

3. **Temporal continuity and Hysteresis:** Again in comparison to Condition 1, this should indicate the differential importance of the split pathway. Also, comparing this condition to Condition 5 should indicate the combined effect of temporal continuity and hysteresis, which should work together.

[3] Note that bias weights were not used in this model because they are not necessary for learning the problem, and they make the analysis of the learned weights more difficult.

4. **Fixed "where" weights:** Having only the split pathway, this should indicate if off-loading the burden of representing location is enough to produce spatially invariant representations in the "what" pathway (the findings of (Rueckl et al., 1989) suggest that this should be the case, but they did not use the encoder network format, so their results depend on the validity of the output representations used).

5. **None:** This is the control condition which will be used to determine the probability of an unconstrained network developing spatially invariant representations.

The parameters used for these simulations were as follows: learning rate (ϵ) = .25, momentum factor (α) = .90, h = .60.

Results

The results were scored as follows: First, all the weights to each of the two "what" units from the input layer were coded as either *on* or *off* by comparing each weight to the average of all weights to the input layer (those above average were *on*, and those below were *off*). For a given "what" unit, the row having the most *on* weights was considered to be the object that the unit represented. The strength of this representation is the number of *on* weights in that object's row, minus the number of *on* weights in the other row. Thus, a perfect representation would have 5 *on* weights in a given row and 5 *off* weights in the other row, resulting in a score of 5. The total score is just the sum of the individual scores for each "what" unit, resulting in a maximum of 10. Since there are two "what" units, each had to be assigned to one of the two different objects. If each "what" unit became selective to the *same* object, then the one with the lowest score was assigned to the other object. In the case of two perfect representations of the same object, the individual scores would be a 5 and a -5 (for the other object), resulting in a total score of 0.

Applying this scoring scheme, an interesting trend developed in the data. For Condition 1, the "best-case" network, only one of the two "what" units reliably developed invariant representations, while the other was somewhat random. In retrospect, this makes sense if one considers that the on-off state of one of the "what" units contains enough information to distinguish between the two objects. If a "what" unit has positive weights to one row of the input layer and negative weights to the other, then it will be positive for one object and zero for the other. The output units can make use of both of these states, so that the other unit is essentially redundant. This explains the pattern of results, and suggests that instead of a total of both units, the best of the two units should be used as the final score.

Figure 4 shows the scores for this simulation. The only significant differences were between the "All Constraints" condition and every other condition ($p < .0001$, Bonferroni/Dunn) for the "best-of" measure.

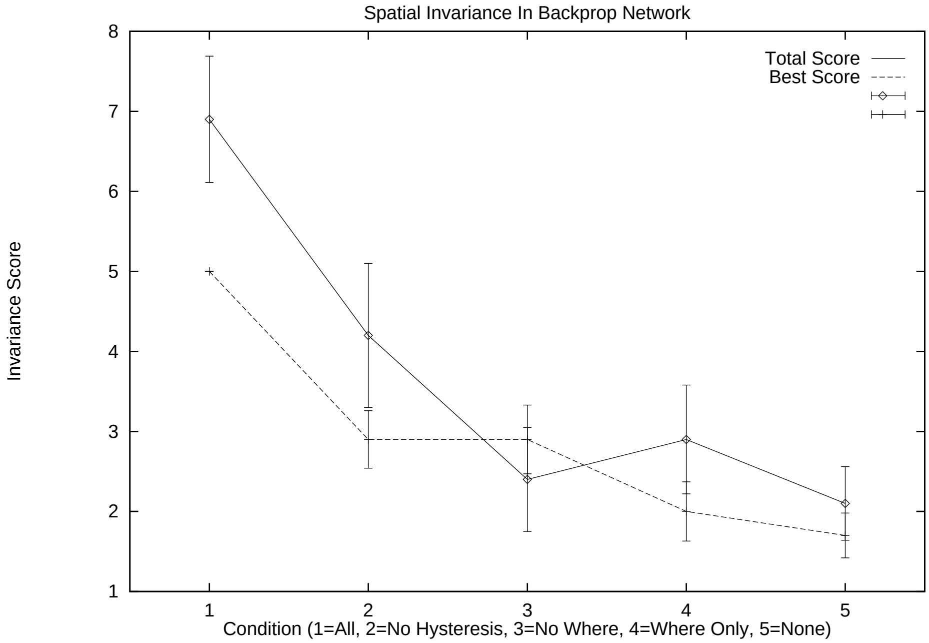


Figure 4: Results of manipulating the presence of three constraints, showing decreasing invariance developing with less constraints. The "Best" score, representing the best of the two "what" units, is the more accurate measure.

Discussion

The most interesting aspect of the results from Simulation 1 is the critical dependence of the spatial invariance effect on the presence of all of the factors hypothesized to be important. Once either the hysteresis (Condition 2) or the split pathway (Condition 3) is removed, the network produces significantly less spatial invariance in the "what" pathway. This indicates the systemic nature of the mechanism, as it requires all elements of the system to be intact for the effect to occur, dropping off significantly when any critical element is removed. Also interesting is that the results are generally in the direction of the number of constraints applied. Thus, each constraint adds something to the effect.

This pattern of results is encouraging, as it is consistent with the hypothesized nature of the spatial invariance mechanism. However, there are several problems with the backprop implementation that make it a less than satisfying model of a biologically realistic network. One of the great appeals of neural network models is their biological realism, and the algorithm suggested here was developed around the specific wiring patterns (recurrent activation loops and lateral inhibition) and learning mechanisms (Hebbian associative learning) known to be operating in the brain. Thus, instead of trying to further the backprop implementation, we will instead develop a Hebbian implementation. However, the backprop network has provided a very general framework for testing multiple factors, which is not as feasible in more specialized networks.

Modified Hebbian Learning Mechanism

The Hebbian learning mechanism proposed by D. O. Hebb (Hebb, 1949), is the most plausible form of synaptic modification given the current state of knowledge about the underlying biochemical mechanisms. Many researchers have noted the parallels between features of *Long Term Potentiation* (LTP) in hippocampal neurons and the Hebbian learning rule (e.g. Rolls, 1989; Levy et al., 1990; Bear & Cooper, 1990; Miller, 1990a; McNaughton & Nadel, 1990). However, both the biochemical data and the original theory proposed by Hebb are inadequate as complete descriptions of a computationally effective learning rule. As has been pointed out by many theorists, Hebb's original rule is unstable because the conditions under which synapses decrease in efficacy are not specified, so that all weights would eventually saturate. Also there are features which have important computational implications that are not specified by Hebb's rule either, including the locus of synaptic modifications, the activation dynamics of neurons employing this learning rule, and the nature of lateral interactions between neurons. Many researchers have proposed solutions to these various problems, but a biologically plausible, interactive network employing a Hebbian learning rule has yet to emerge. We will approach this problem by examining the difficulties with a simple rule proposed by Oja (1982), which has a clear computational interpretation.

In developing our learning rule, we will rely on both neurobiological and computational constraints. As such, we will use the terms *neuron* and *unit* interchangeably, and the term *weight* to refer to the net efficacy of a synapse. Weights are denoted by the term w_{ij}, which is the weight from unit i to the unit j. Also, both the pre and postsynaptic components of synaptic efficacy will be examined. These components are denoted by the suffix *pre* and *post.* Most neural network models developed for psychological modeling summarize the synapse into a single weight value, when in fact both a presynaptic (release of neurotransmitter (NT) into the synapse) and a postsynaptic (the effect of the NT on dendritic receptors) mechanism are at work. The significance of these two different weight components will be discussed below.

The meaning of the term *activation* in biological terms is a little trickier, because the actual thresholded spiking behavior of real neurons is not analytically convenient to work with in abstract network models. Instead, a continuous real number (denoted by a_j for the jth unit) in the range (-1,1) and centered on the firing threshold for the neuron is used to represent the average depolarization level over a relatively short period of time (10's of milliseconds, perhaps). It is assumed that there is some thresholded relationship between this average depolarization level and the probability that the neuron will spike. Increased depolarization above the threshold translates into an increased rate of firing, and below it no firing occurs. This is modeled such that the positive range of the activation value corresponds to increased depolarization above the threshold, and can be interpreted as an average spiking rate. Since, in real neural networks, information is only transmitted through the spiking action of axons, the output function for a given unit is as follows:

$$o_j = \begin{cases} a_j & \text{if } a_j > 0 \\ 0 & \text{otherwise} \end{cases} \tag{2}$$

The o variable is used instead of the raw activation value a whenever a presynaptic neuron is involved in transmitting information. This interpretation of activation is more flexible than those that use only a (0,1) range and associate it directly with spiking probability, as it allows one to model the effects of hyperpolarization and the dynamics of sub-threshold activations.

Computation Performed by Hebbian Learning

A Hebbian learning rule of one form or another has been employed in many models, most of which perform some kind of self-organizing pattern recognition or categorization (also known as vector quantization) on a set of input patterns. The simplest form of the Hebbian rule is:

$$\frac{1}{\epsilon}\Delta w_{ij} = a_j o_i \tag{3}$$

where ϵ controls the learning rate, and i is an index over a set of input units, and j an index of units in an output layer connected to the input units by weights w_{ij}. Intuitively, this rule will lead to an increase in weights between input units whose activation is correlated with the activation of the output unit. Further, because the activation of the output unit is a function of all the input units it is connected to, the Hebbian rule will tend to capture correlations between input units. Over a series of different input patterns, this function will cause the weights to unit j to represent the central tendency of the input unit activations. Indeed, with a modification proposed by Oja (1982), a variant of this simple rule can be shown to be performing principal components analysis (PCA) on the inputs, and extracting the first principal component over all the patterns presented to the input array. The specific modification involves the inclusion of a weight decay term:

$$\frac{1}{\epsilon}\Delta w_{ij} = a_j(o_i - a_j w_{ij}) \tag{4}$$

which will tend to decrease the weights whenever w_{ij} (modulated by the output unit's activation) is greater than the input activation o_i. Since Oja used a linear activation rule, a_j is just the sum of all the inputs, $\sum_k o_k w_{kj}$, so that the equilibrium condition (when $\Delta w_{ij} = 0$, which occurs when $o_i = a_j w_{ij}$) results in the following weight value:

$$\breve{w}_{ij} = \frac{o_i}{\sum_k o_k w_{kj}} \tag{5}$$

where ($\breve{}$) denotes equilibrium, and k is an index over all the input units. Thus, each weight will come to represent the normalized proportion of the total input to unit a_j for a given o_i. The weight decay term will serve to normalize the weights to the unit, preventing the infinite growth of weight that would occur under the simplest Hebbian function shown in equation 3. We will take the normalized learning rule shown in equation 4 as the simplest case of a practical implementation of a Hebbian learning rule.

Given that we are considering the domain of visual object recognition, we will now specify the computational goal of the Hebbian learning algorithm in terms of recognizing individual patterns of activation over a set of *input* units which represent the output of visual preprocessing such as that produced by the lower layers of visual cortex. We consider the process to take place over several *layers* (i.e., areas) of visual cortex, where the *output* layer contains visual object representations. We assume that individual visual objects produce distinct but somewhat noisy and possibly incomplete patterns of active units in the input layer. There is a many-to-one mapping between visual patterns and objects, such that a given object can project many visual patterns onto the retina. These patterns vary along dimensions such as size, spatial location, and orientation. The critical components of this task are:

1. Different visual objects should be represented by different patterns of activation in the subsequent layers, while partial and slightly noisy versions of the same object should be represented by similar patterns of activation.

2. The different images of a given object in different locations, sizes, and orientations should not be confused for other objects. These two constraints act in opposition, and a balance must be struck between them.

3. Subsequent layers should produce more condensed representations of the pattern of activation on the previous layer. In information-theoretical terms, the visual channel contains highly redundant information with respect to the representation of visual objects, given that (in the simplified case presently being considered) there is only one object being viewed, but many units activated by this object at the lowest levels of representation. Thus, the goal is to reduce this mass of redundant information into a compact, stable representation at the highest levels.

According to this set of requirements, the Oja rule by itself would fail miserably, as it would cause all output units to represent the average of all the input patterns combined. They would all discover the main principal component over all the input patters seen by the network. There are two related but distinct components of this problem: One is that each output unit is averaging over all the input patterns, and the other is that all the output units are representing the same thing.

These problems are related because if only some output units were allowed to become active at a given time (through lateral inhibition, for example), then equation 4 would cause those units that were active to represent the central tendency of the patterns presented on the input layer during the period while they were active. To the extent that different input patterns were distinct enough to activate different output units due to the random initial weights, this system would work well. Indeed, this is the idea behind the competitive learning algorithm and its many antecedents and descendants (Rumelhart & Zipser, 1986; Bienenstock et al., 1982; von der Malsburg, 1973), which works through a *Winner Take All* (WTA) mechanism whereby the unit which responds strongest to a given input pattern adjusts its weights according to a function very similar to equation 4.

The central problem with this kind of competitive mechanism is that overlapping but distinct input patterns (like those found in distributed representations) tend to be represented by the same output unit, as the increase of weights to a given input pattern makes it more likely than the other output units to respond to any subsequent input pattern having some common units active. There have been several proposals suggested for eliminating this problem, which are discussed below. However, it is worth mentioning that the averaging behavior of the Oja learning rule plays into this problem, rather than acting to counteract it.

Finally, a WTA-style lateral inhibition mechanism will reduce the number of output layer units active to 1, satisfying the final requirement that the output be a reduced representation of the input. However, a more useful reduction of the information would not eliminate the distributed nature of the representation, rather it would reduce the variability of activity patterns on the input corresponding to a given object, and enhance the variability of the output pattern with respect differences between objects. To this end, generalizations of the strict WTA constraint to N active units (N-WTA) are possible, which retain distributed representations at all levels of the network.

Problems With the Oja Rule

Despite the promise of a competitive activation mechanism, there are two main problems intrinsic to the Oja rule itself. First, it requires a linear activation function in order for the equilibrium condition shown in equation 5 to hold and thereby normalize the weights. If a logistic activation function which squashed the input into the range (-1,1) were used, the denominator in equation 5 would no longer represent the total input to the output unit, and would not produce the desired normalization effect. This would cause the learning to become unstable. A nonlinear activation function is a prerequisite for multi-layer mappings to be effective, however, so this is not a trivial problem. Also, attempts to simply clip the weights at a certain ceiling level would eventually result in the saturation of all weights to that level, as even low positive activation levels in the input units would result in increasing weights.

It might also be possible to change the definition of the rule slightly so that the total input to the unit is used in equation 5 as is required for normalization, while at the same time using the logistic function on the other activation term. The principal disadvantage of this mechanism would be that the neuron would have to somehow retain the net input information, which is not likely to happen in the biological system given that the origin of activational nonlinearity is due to the electrical properties of dendrites (e.g., Rall, 1990). Thus, the neuron computes the sum of a number of non-linear functions of its inputs, instead of somehow taking the sum first, and then applying a non-linearity to it.

Additionally, the Oja rule does not make provision for bidirectional weights. These are crucial for the recurrent activation loops phenomenon, and for continuous activation models which require these weights to achieve stability of the activation function. In order to accommodate recurrent weights and a continuous activation function, the convergence properties of this weight rule in an interactive network would need to be analyzed.

Strong Constraints From Neurobiological Data

In addition to the problems specific to the Oja rule, there are several very well established properties of neurons that must be accommodated by any realistic learning rule. These are as follows:

1. **Fixed sign weights:** Because weights are actually a combination of the release of NT from the presynaptic neuron and the subsequent effect of that NT on the postsynaptic neuron's receptors. Neither of these factors can change the direction of their effect on the postsynaptic neuron (Miller, 1990b; O'Reilly, 1989). This means that the same weight can not represent directly either a correlation or an anti-correlation, but must represent one or the other only. In most areas of cortex, the weights between layers are excitatory (positive), and between layers a mix of inhibitory (negative) and excitatory. To simplify, we will assume positive inter-layer weights, and inhibitory intra-layer weights (which implement the WTA activation function).
2. **Positive-only presynaptic activations available to the postsynaptic neuron:** As was discussed above, a postsynaptic neuron will only receive information from a presynaptic neuron if that neuron fires.
3. **Positive-only postsynaptic activations involved in weight change:** The leading candidate for a mechanism underlying synaptic modification is the NMDA receptor, which only opens when the postsynaptic neuron is depolarized (activated) beyond some threshold determined by the unblocking of a Mg^{2+} ion from the receptor (Collingridge & Bliss, 1987; Rolls, 1989; Levy et al., 1990; Bear & Cooper, 1990; Miller, 1990a; McNaughton & Nadel, 1990).

The net effect of these constraints is that a learning rule must rely on only positive correlations to govern the adjustment of weights, given that both neurons involved must be positively activated to be represented in the standard Hebbian term $a_j o_i$.

A Pattern-Covariance Alternative

We now return to the standard Hebbian rule from equation 3 to see if an alternative formulation to Oja's weight normalization routine can be found that fits more naturally with the constraints discussed so far. First, if the Hebbian rule is to be computing correlations among the input units, then the activations used in equation 3 should be replaced by the temporal covariance of the input unit activations: $(o_i - \langle o_i \rangle^t)$ (where $\langle o_i \rangle^t$ represents the expected value of the variable in the angle brackets over some sample of the variable above the brackets, t=time in this case). This makes equation 3 actually:

$$\frac{1}{\epsilon}\Delta w_{ij} = o_j(o_i - \langle o_i \rangle^t) \qquad (6)$$

Note that we are now using the output function for the receiving unit (o_j) in the weight update rule to restrict the weight changes to positive-only postsynaptic activations (see item 2 above). The covariance of the activation term is important because it enables variation in activation below the mean for that unit to be registered as anti-correlated. Also, if an input neuron had some background level of firing, this would show up in the $\langle o_i \rangle^t$ term, and only activation above this mean level would represent a valid signal.

However, there are several problems with using the $\langle o_i \rangle^t$ term in a biologically realistic neural network. It implies that the statistics of the input environment are either known in advance or stable enough over time to allow a floating average calculation to approximate this term with sufficient accuracy. Further, given that we want to establish different representations for different input patterns, the critical dimension on which to compare activation values should be the activation over a given input pattern, rather than the behavior of a given unit over time.

One solution to these problems, used by Oja and others (c.f. Hertz et al., 1991), is to simply drop the $\langle o_i \rangle^t$ term by assuming that all input units have a mean of 0 over all input patterns. However, given the constraint that only positive activations are transmitted to the receiving unit (see constraint 2 above), the mean value of o_i will always be positive. Also, most neurons that have been subjected to single cell recordings have a basic low-level firing rate independent of the presence of the stimulus to which they respond maximally, which would add to this $\langle o_i \rangle^t$ term and make the zero mean assumption even less viable.

Another alternative can be found in a similar formulation of the basic Oja rule proposed by Linsker (1988; 1986), which also uses a linear activation function as a critical component in the learning rule. Linsker starts with the following learning rule:

$$\Delta w_{ij} = \epsilon_1 (a_j - \epsilon_2)(o_i - \epsilon_3) + \epsilon_4 \tag{7}$$

(where the ϵ_x's are constants) and, because a_j is just the linear sum of all $o_i w_{ij}$'s, this function can be re-written entirely in terms of the inputs to a given output unit. When the values for ϵ_2 and ϵ_3 are set to appropriate functions of the temporal average over patterns of the input unit activities $\langle o_i \rangle^t$, this equation is actually adjusting the weights in proportion to the correlation between input unit i and all the other input units that feed into output unit j. The Linsker rule suffers the same problems as the Oja rule in terms of needing a linear activation function, but it does not offer the normalization properties of the Oja rule. Also, it uses the temporal average term for computing correlations, so the same problems apply.

Another alternative exists, however, and that is to use the *pattern covariance* instead of the usual temporal covariance described above. Pattern covariance is defined here as $(o_i - \langle o \rangle^{in})$, where $\langle o \rangle^{in}$ represents the mean ($\frac{\sum_k o_k}{N_{in}}$, k indexing over input units) over the current pattern of input activations impinging on the output unit. The weight update rule would then be:

$$\frac{1}{\epsilon} \Delta w_{ij} = o_j (o_i - \langle o \rangle^{in}) \tag{8}$$

This form of covariance, while not formally equivalent to temporal covariance, shares many of the same properties which make temporal covariance preferable to straight activations, and it overcomes the problems with temporal covariance. Specifically, any signal that exists above the low-level random firing of the input neurons will show up as a positive covariance term over the current input pattern, regardless of the long-term statistics of each individual input unit.

One way to think about the properties of pattern covariance is to imagine that the statistical variable one is trying to measure is the input pattern, instead of the input unit. The temporal covariance measure described above assumes that each input unit is a variable that is somehow being measured and compared to the other variables (input units). Each measurement of this variable is the activation at a different point in time. In this situation, it makes sense to use the mean activation over the lifetime of each unit in the covariance term because this establishes the baseline for the variable in question.

However, if a given input *pattern* over all input units is the variable of interest, and there are several different patterns in the environment, then each pattern is a different variable. The measurement of this variable is performed over the set of input units, so that there are N_{in} measurements of each pattern variable, one for each input unit. In this situation, a reasonable mean signal for the pattern is the average activation over all input units for that pattern, as this is the mean of all measurements of this variable. This corresponds to the function shown in equation 8.

The emphasis on the input pattern versus the individual input units is important, as it is this pattern which is to be recognized and represented, not the relative statistics of each input unit relative to the others. Intuitively, the pattern covariance measure is simply a way of differentiating the signal of a given pattern from the noisy and inactive regions of the pattern.

As an added benefit, the pattern covariance term is unaffected by the positive-only output function imposed by real neurons because any neuron that is not firing will be below the mean over all the inputs to a given unit (given that some other input units are active), and will thus be anti-correlated with an active output unit. Finally, the computation needed to implement the $\langle o \rangle^{in}$ term is local both in space and time, and even has an interesting biological interpretation, which will be discussed below.

Having argued in favor of using a pattern covariance term for the input units, it would be logical to extend this argument to the output units, so that a true correlation-like measure between the input pattern and the output pattern would be computed by the learning rule. This would change the rule to:

$$\frac{1}{\epsilon}\Delta w_{ij} = (o_j - \langle o \rangle^{out})(o_i - \langle o \rangle^{in}) \qquad (9)$$

where $\langle o \rangle^{out}$ is the average of all positive activations over the output layer. There are two conceivable ways to implement the $\langle o \rangle^{out}$ term using localized information available to the two units linked by weight w_{ij}. One is to use additional lateral inhibitory connections between units in the output layer such that the total inhibition received by a given unit is equal to the mean positive activation level over the layer. If this were the case, the *activation level* of the units would reflect the covariation term directly, and the $\langle o \rangle^{out}$ term would be represented in the activation function instead of the learning rule as it is here.

There is a problem with this lateral inhibition formulation, however, because it does not have the proper effect on the weight itself. Specifically, it does not provide for a decrease in weight when the output of unit j is below the mean output for the entire layer, as should be the case according to equation 9. With lateral inhibition, a sub-mean unit would simply not change its weights at all, as its output function would be 0. Indeed, it should be evident that the addition of lateral inhibition of this type does not affect the kind of learning performed by equation 9 much at all, as the absolute value of the activation is not as important as its relationship to the mean activation over the layer, which is preserved with lateral inhibition.

Thus, we are led to an alternative formulation that can be computed locally, which is to use the *presynaptic* weights of unit i in the input layer in addition to the postsynaptic weights (which we have been tacitly assuming) to unit j in the output layer. Because the presynaptic unit i is connected to all the output units, it potentially has access to their activation levels (more on the biological plausibility of this below), and could be computing the pattern covariance over the output layer. The resulting weight change equation for the net weight between units i and j would be a combination of the presynaptic and postsynaptic components:

$$\frac{1}{\epsilon^2}\Delta w_{ij}^{net} = \Delta w_{ij}^{pre} \Delta w_{ij}^{post} \quad (10a)$$

$$\frac{1}{\epsilon}\Delta w_{ij}^{pre} = o_j - \langle o \rangle^{out} \quad (10b)$$

$$\frac{1}{\epsilon}\Delta w_{ij}^{post} = o_i - \langle o \rangle^{in} \quad (10c)$$

This formulation is equivalent to equation 9 (replacing ϵ with ϵ^2), but it is re-written to make the computation at each synapse depend on information local to the synapse and neuron involved. Only the presynaptic neuron has both the activation of postsynaptic unit j and the average over the entire output layer (in the fully interconnected case we are considering here), so the weight adjustment based on these two factors must be located in the presynaptic component of the weight. The same argument holds for the postsynaptic component of the weight adjustment.

Zero-Sum Interpretation

There is an interesting interpretation of the correlation form of the Hebbian learning rule shown in equations 10b and 10c when they are re-written as:

$$\frac{1}{\epsilon}\Delta w_{ij}^{pre} = o_j - \overline{\Delta w_i^{pre}} \quad (11a)$$

$$\frac{1}{\epsilon}\Delta w_{ij}^{post} = o_i - \overline{\Delta w_j^{post}} \quad (11b)$$

which shows that the second terms of these equations can be re-written as the average weight change that would have been computed for either the pre or postsynaptic portion of the weight for the entire set of weights belonging to a unit, if the weight change was simply a function of the activation state of the units on the other end of the weight (o_{out} or o_{in}, respectively). So, for presynaptic unit i having weights to all of the o_{out} output units, the term $\langle o \rangle^{out}$, is really just $\frac{\sum_l o_l}{N_{out}}$ (with l as an index over all output units), which is just the average activation of this population of units. The same argument can be made for the postsynaptic weights from all of the o_{in} input units to a given output unit. Note that these equations are computed in two steps, the first of which enables the computation of the $\overline{\Delta w_i^{pre}}$ and $\overline{\Delta w_j^{post}}$ terms according to the output and input unit activations respectively, and a second step which is shown in the equations.

This equation represents a *Zero-Sum Hebbian* (ZSH) rule because it will result in a net weight change of 0 for the total of each weight component. Zero-sum effects like these occur often in nature, as they are a direct consequence of having limited resources available, or from the competition between two different systems (as is suggested by Bear & Cooper, 1990). While the specific circumstances under which a synapse will decrease in efficacy, or *Long Term Depression* (LTD) are not as clearly understood as the LTP mechanism, many researchers have computational or theoretical models of LTD (Rolls, 1989; Levy et al., 1990; Bear & Cooper, 1990; Miller, 1990a; McNaughton & Morris, 1987), and empirical evidence is accumulating (Artola et al., 1990; Stanton & Sejnowski, 1989; Bradler & Barrionuevo, 1990; Frégnac et al., 1988). Several of the models (Rolls, 1989; Levy et al., 1990; Miller, 1990a) specifically hypothesize a competitive mechanism based on a limit of total postsynaptic efficacy, which is what the ZSH equations postulate.

While several researchers have employed normalized weight update schemes similar to that shown in Oja's rule in simulated neural networks (e.g. Rumelhart & Zipser, 1986; von der Malsburg, 1973; Bienenstock et al., 1982; Grossberg, 1976), to our knowledge only two others (Miller, 1990a; Rolls, 1989) have used the weight conservation approach detailed here. Thus, the computational properties of this learning rule as implemented in working neural network simulations have not been extensively explored. However, both of these researchers employed this algorithm in simulations whose goal was similar to that stated above regarding visual object recognition.

Rolls (1989), in simulating the function of the hippocampus, developed a simulation that "effectively selects different output neurons to respond to different combinations of active input patterns", which is quite similar to the objective of recognizing distinct patterns of activity over the input layer as distinct visual objects. Miller (1990a) applied a ZSH-style rule to the development of ocular dominance columns in the visual cortex, showing that under certain conditions the zero-sum constraint was critical for the differentiation of a given neuron's receptive field between the right and left eyes.

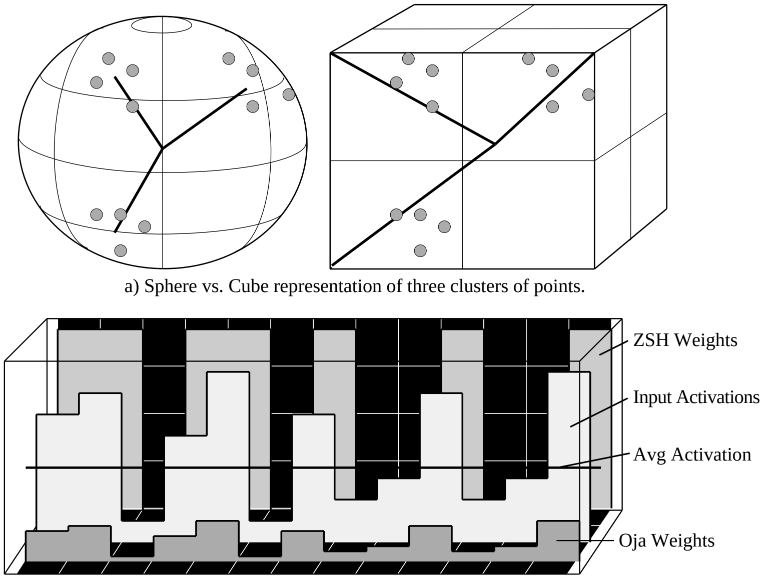
What both of these previous applications of a ZSH-style learning rule demonstrate is that this constraint is important for making a neuron make a "forced-choice" kind of decision regarding which input units to respond to. This kind of behavior is a direct result of the fact that once the weights from an active output unit to one input pattern have been increased, there will be a correspondingly diminished amount of weight over the other input units to that output unit, so that another output layer unit which did not respond to the first input pattern will be more likely to respond to a different pattern because there will be less competition from the units that responded to the first pattern.

The effects of the ZSH rule on individual weight values can be seen by examining what would happen with a stable input pattern which always activates a given output unit. It should be clear that the weights will be driven to extremum by equations 11a and 11b, with those units having above-average activation values always driving the weights upwards, and the below-average inputs driving the weights monotonically downwards. This is in contrast to the weight normalization performed by Oja's rule, where the weight update rule imposes a built-in constraint on the weight magnitudes. Also unlike Oja's rule, the instability of the ZSH rule is amenable to simply fixing the weights within certain boundaries because only the weights that "should" be increased are being increased, while the others are being decreased, so that the weights will not become completely saturated over time. With the constraint that weights do not change sign, the obvious choice for a lower weight bound is 0, and the upper bound can be any arbitrary value (1.0 is used for convenience).

While it might be considered advantageous to have self-bounded weights, it is important to note that the pattern-differentiation effect in a ZSH model will not occur with a weight-normalization approach (e.g., the Oja rule), because the weight from each input unit to a given output unit will approximate its relative frequency of firing, whereas the ZSH rule will make an active delineation between those input units which have above-average output, and drive those weights to the upwards boundary while driving the remaining weights downwards. The result is a contrast enhancement of the input pattern.

The difference between weight normalization and weight conservation effects for a given input pattern is illustrated in Figure 5. This figure shows that weight normalization results in a vector pointing to the center of a set of pattern vectors on a (n-dimensional) sphere, while weight conservation results in a vector pointing towards the nearest corner of an (n-dimensional) cube. This difference is also shown in the bar-chart portion of the figure, where the active (above-average) components of the input vector are pushed to 1, while the others go to 0 under weight conservation, but weight normalization has each weight vector strictly proportional to the relative activity of the unit. The two different representations of the difference between a ZSH and a Oja learning rule are equivalent, but the bar-chart allows one to graph a higher dimensional example.

The value of going to a corner on a cube *vs* a point on a sphere is that there are an infinite number of points on a sphere, but only a fixed (2^n) number of corners on a cube, so that weight conservation has the effect of reducing the information content of the input vector (where information is the number of possible states of a system). While the reduction of information is a stated goal for visual representations, there is an inevitable cost associated with reductions of this sort, namely a distortion of the data. However, this form of distortion is similar to that performed by a sigmoidal "squashing" activation function—it forces a continuous domain into a more binary (n-ary in this case) domain. For categorization and pattern recognition, we feel that it is more important to differentiate between different stimuli



a) Sphere vs. Cube representation of three clusters of points.

b) Bar-Graph rendition, where light grey is the activations of input units, medium grey in background is result of ZSH learning, dark grey is result of Oja-style normalization.

Figure 5: **a)** Weight normalization produces (n-dimensional) spherical weight vectors pointed at the center of cluster of input patterns, while weight competition produces weight vectors pointing at the corner of a (n-dimensional) cube closest to the input patterns. **b)** A bar-graph rendition of the difference between weight normalization and weight competition for a fixed set of inputs.

than accurately capture the precise direction an input vector points, as is the case with weight normalization.

This perspective contrasts with that of Linsker (1988), who claims that the goal of representations is to preserve the maximum amount of information in the input. In the linear case, his *Infomax* principle is equivalent to the PCA representation formed by Oja's rule, so the difference is captured by the sphere *vs* cube distinction. We would say instead that the goal of representations is to maximally differentiate between truly different input patterns, while generalizing over small distortions of the same pattern. This is achieved in the cube analogy by realizing that the decision surface of a trained network will tend to cause a given input vector to be categorized into its nearest corner.

Another category of modifications made to simple Hebbian learning schemes which address the problem of having differentially selective units develop in the output layer involves adding an activity-modulated gain function to a unit's activation function. The gain for a given unit moves in inverse proportion to the activity of the unit, much like a sensitization / desensitization effect. Thus, units which are responding to too many of the input patterns will have their gain reduced, and will therefore be less likely to be activated, while those which have not been active have a higher gain, making them more likely to respond (Bienenstock et al., 1982; DeSieno, 1988). This scheme results in different input patterns activating different output units, but it does so through the activation function. The consequence of this is that it depends on the frequency of pattern presentation being roughly equal across patterns, which seems to us to be an unrealistic constraint to place on the environment. Therefore, we feel that the weight-modulated ZSH selectivity mechanism is a more robust way of differentiating input patterns.

Dual Mechanisms at the Synapse

In order to obtain a true correlation term between the input and output layer in the learning rule, it was necessary to introduce a modifiable presynaptic weight component. Biologically, this weight corresponds to the release of neurotransmitter (NT) at the synapse. The pattern covariation formulation of the weight update rule requires that there be a fixed amount of NT which is distributed over the synapses of a given axon in a competitive, zero-sum manner. Further, the competitive mechanism involves the presynaptic unit having access to the postsynaptic activation levels of all the neurons to which it projects.

There are many contradictory findings regarding the locus of synaptic modification, with some researchers claiming it is presynaptic (e.g., Bliss, 1990; Bekkers & Stevens, 1990) and others supporting a postsynaptic mechanism (e.g., Lynch & Baudry, 1984; Perkel & Perkel, 1985; Rall & Segev, 1987). Current thinking is that both pre and postsynaptic processes must be involved, given the weight of the evidence. Postsynaptic changes are probably due to either the exposure of additional receptors or changes in the shape of dendritic spines, while presynaptic changes are due to changes in the efflux of NT. Further, there is evidence that presynaptic changes depend on the postsynaptic NMDA receptor complex (which itself depends on the depolarization state of the postsynaptic cell) through a retrograde messenger that goes from the postsynaptic to the presynaptic cell (see Fazeli, 1992 for a recent review).

We feel that the biological evidence sufficiently supports the use of two synaptic mechanisms. Our imposition of a zero-sum constraint on these two mechanisms is probably a more rigorous constraint than is actually warranted, but the general character of the competitive interactions among synapses of the same neuron is likely to correspond to biological reality.

The other remaining issue regarding the dual synaptic mechanisms is the way in which a net synaptic efficacy results from the individual efficacies of the two components. According to the correlation model developed above, the derivative of the net weight should be the product of the derivative of the two individual synaptic efficacies, so that the function itself is just the integral of both sides

$$\int dw_{ij}^{net} dt = \int dw_{ij}^{pre} dt \; dw_{ij}^{post} dt \tag{12}$$

which, given that each of the two right-hand terms are separate functions of dt, can be written as $\int dw_{ij}^{pre} dt \int dw_{ij}^{post} dt$, which is simply,

$$w_{ij}^{net} = w_{ij}^{pre} w_{ij}^{post} \tag{13}$$

Biologically, this function says that there is an interaction between the amount of NT released into the synapse (w_{ij}^{pre}), and the efficacy of the postsynaptic receptors, (w_{ij}^{post}) on the net effect of a given synapse on the postsynaptic cell. Most models of synaptic efficacy do indeed postulate such a multiplicative interaction (e.g., McNaughton, 1988; Shepherd, 1990).

Activation Function And Network Convergence

Because we will be using continuous-valued activations in order to capture interactive and inhibitory effects, we need to demonstrate that the activation update function will stabilize for a given set of weights, and further that the weight adjustments made will act to move the stable patterns of activation in the desired direction. Following the previous approaches to this problem (e.g. Hopfield, 1984; Movellan, 1990; Hinton, 1989), this can be proven analytically by showing that both activation and weight updates will drive a global *Goodness* (also referred to as *Energy*) measure in the appropriate direction. A global function of this nature is known as a Lyapunov function if it can be shown that the system will always converge on a minima or maxima of it. In general, this function is not related specifically to the underlying dynamics, but merely acts as a "metric" function similar to Euclidean distance in Cartesian space. For neural networks, a simple Goodness measure takes into account how well the activations reflect the current weights, so the following function is typically used (Hopfield, 1984)[4]

$$H = \frac{1}{2}\sum_i \sum_j a_i w_{ij} a_j \tag{14}$$

where the H represents the *Harmony* of the fit between the weights [5] and the activations.

According to the Harmony function, the signs of the activations on either side of the weight and the weight itself must all be positive or any two negative and the third positive for a positive contribution to the Harmony measure. Any other combination would result in a negative value. However, once positive, there is nothing to constrain the activations from taking on increasingly large values to increase the Harmony measure. For this reason, a second term is added which penalizes large activations[6]:

$$S = \sum_i \int_{rest}^{a_i} f^{-1}(a)da \tag{15}$$

and the resulting overall Goodness function is $G = H - S$, so that increasing Harmony due to larger activations is offset by increasing *Stress* to the activation levels in the network.

Hopfield (1984) has shown that these two terms when combined with a suitable logistic activation function result in G being the Helmholtz free energy function, which allows such

[4] Note that we are using a positive *Harmony* measure instead of Hopfield's original negative Energy measure

[5] We are using w_{ij} to represent the net weight between units, which is actually w_{ij}^{net} in this and all subsequent equations in this analysis.

[6] Note that the letter S was originally used to represent Entropy, but we use it here to indicate *Stress*

a system to be interpreted according to known physical principles. Movellan (1990) has shown using this function that the standard *Interactive Activation and Competition* (IAC) activation function (McClelland, 1981; Rumelhart et al., 1986a) will result in the network finding and settling into stable states for a given set of weights, and that a simple Hebbian learning rule (such as that shown in equation 3) will perform gradient ascent in terms of the global Goodness function G.

The IAC activation function (in a slightly modified form) contains a decay term and an input term (with the relative contribution of these two factors controlled by modifying the gain of the input term), with a rate parameter λ:

$$\frac{da_j}{dt} = \lambda(-a_j + f(I_j^{net})) \tag{16}$$

where λ is the parameter controlling the rate of change in activation, and $f(net_j)$ is the following input function:

$$f(I_j^{net}) = \begin{cases} I_j^{net}(max - a_j) & I_j^{net} > 0 \\ I_j^{net}(a_j - min) & I_j^{net} < 0 \end{cases} \tag{17}$$

and I_j^{net} is the net input to the unit:

$$I_j^{net} = \sum_i o_i w_{ij} + \sum_{l,l \neq j} o_l w_{lj} \tag{18}$$

(with l indexing over the other units in the same layer with inhibitory connections). It turns out that the actual form of the $f(I_j^{net})$ input function is not critical for the proof, as any monotonic function would produce the same results.

Presumably, we could simply use this activation function and the weight update rule from equation 8, but the Harmony measure used in these other models does not correspond exactly to the pattern covariation model being used here. Therefore, we will modify the Harmony term to include the pattern covariation, and we will also reformulate the function into three separate terms: one for the feedforward weights to the output layer, one for the feedback weights from the output layer to the input, and another for the lateral inhibition between units in the output layer:

$$H = \frac{1}{2}\left(\sum_j (a_j - \langle a \rangle^{out}) \sum_i (a_i - \langle a \rangle^{in}) w_{ij} + \sum_i (a_i - \langle a \rangle^{in}) \sum_k (a_j - \langle a \rangle^{out}) w_{ji} + \sum_j a_j \sum_{l,l \neq j} a_j w_{lj}\right) \tag{19}$$

where j and l are indices over the output layer, and i over the input layer. Because we are considering the input and output layers separately, we also need to update the Stress function to sum over both of these layers:

$$S = \sum_i \int_{rest}^{a_i} f^{-1}(a)da + \sum_j \int_{rest}^{a_j} f^{-1}(a)da \tag{20}$$

The Goodness function is then just $G = H - S$, as before. If it can be shown that this function is being maximized by the activation and weight functions, then we can prove the stability of the system. This is true despite the changes made to the Goodness function, as any function can serve as a Lyapunov function (although not many will actually work!).

In writing these functions, we are using activations instead of the output functions that should be used in order to simplify the derivation with respect to the activation function. In order to make this work, we temporarily assume that the activations are bounded in the range (0,1), instead of the (-1,1) range postulated at the outset. Also remember that the w_{ij} and w_{ji} weights are positive, while the lateral inhibitory w_{lj} weights are negative and fixed (we are not interested in learning the inhibitory weights, and it is not clear that they are subject to modification in the brain).

In order to determine if this Harmony function combined with the Stress term will result in the network achieving a stable equilibrium with a given set of weights, we simply take the derivative of G with respect to changes in the activation of a given output unit. While we will focus on the output layer units, the same analysis will hold for the input units if they also have lateral inhibitory connections. The details of this derivative can be found in the Appendix. Summarizing these results, the following equation shows the derivative of G with respect to a given output activation change:

$$\frac{\partial G}{\partial a_j} = I_j^{net} - \langle a \rangle^{in} \sum_i w_{ij} - f^{-1}(a_j) \tag{21}$$

In order to obtain this simplified expression, it was necessary to assume that the weights between a input and output unit were symmetric with each other (i.e., that $w_{ij} = w_{ji}$). While it is not difficult to imagine the inhibitory weights being symmetric, as we are assuming they are fixed (they could all just be fixed at the same value), the assumption of symmetry between the inter-laminar weights is more difficult to accept. However, like the networks examined by Hopfield (1984) and Movellan (1990), our weight adjustment rule is symmetric for both the input and output units so that the weight adjustment rule will tend to produce symmetric inter-laminar weights, making this assumption less problematic. In the simulations done on the network the effect of strictly enforcing this constraint versus letting the learning rule take care of it was explored, and found not to make a difference on the convergence or stability of the network.

In order to determine if the IAC activation function will always cause this derivative to increase over time, we must apply the chain rule using the derivative of the activation function over time (see equation 16) to get the derivative of G over time. When this is done (again, see the Appendix for details), the sign of the resulting expression will always be positive given that a certain condition holds. Assuming this condition holds for the moment, we can conclude that in addition to satisfying the original Goodness function specified by Hopfield (1984) and Movellan (1990), the IAC activation function will serve to maximize our global fitness function G, as the derivative of G with respect to time will always be positive. Further, since G is bounded and always increasing, it will eventually settle to a point where $\frac{dG}{dt}$ is 0, which, according to equation 32, means that both $\frac{da_j}{dt}$ and $\frac{\partial G}{\partial a_j}$ must be 0 for all j (Movellan, 1990). This means that the activations will end up in an equilibrium point, which will be a maximum of G.

However, these results depend on the following condition being true for all units j in the output layer:

$$\begin{aligned} I_j^{net} > \ & \langle a \rangle^{in} \textstyle\sum_i w_{ij} \quad ; I_j^{net} > 0 & \text{(22a)} \\ I_j^{net} < \ & \langle a \rangle^{in} \textstyle\sum_i w_{ij} \quad ; I_j^{net} < 0 & \text{(22b)} \end{aligned}$$

In order to analyze this condition, we can first apply some of the strong neurobiological constraints on the signs of these variables, which dictate that w_{ij} is positive, and that $\langle a \rangle^{in}$ is positive as well, so that the $\langle a \rangle^{in} \sum_i w_{ij}$ term will always be positive. Thus, this condition will be violated only in the first case when $I_j^{net} > 0$, but less than $\langle a \rangle^{in} \sum_i w_{ij}$. This will happen when the feedforward input is greater than the lateral inhibition coming from the other output layer units (because $I_j^{net} > 0$), but less than the sum of the weights to the input times the average activation value over the input $\langle a \rangle^{in}$. Ignoring the lateral inhibition for the moment, this results in the following inequality for a given output unit a_j:

$$\sum_i a_i w_{ij} < \sum_i \langle a \rangle^{in} w_{ij} \tag{23}$$

which says that this unit must have had stronger than average weights to the input units which were *not* active than to those that were. Given that there is lateral inhibition operating in the output layer, it is unlikely that such a unit would have a $I_j^{net} > 0$, because some other output unit(s) will have weights that correlate stronger with the present input pattern than a unit meeting the conditions of equation 23. Thus, the inclusion of sufficient lateral inhibition will ensure that the network will converge on a stable activation state for a given set of weights, assuming there are other output units that represent the input pattern. In the initial, random condition, approximately half of the output units will have above-average weights to a given input pattern, so this constraint should not be a problem. Further, the learning rule will act to correct this inequality as the weights will be adjusted to any active output unit in the direction of the present input vector, so that the net input will probably be larger than $\langle a \rangle^{in} \sum_i w_{ij}$ next time around.

This dependency of the pattern covariation learning rule on sufficient lateral inhibition to prevent instability in the activation space is interesting, as it was unanticipated and is not the case with the original Harmony term used by Hopfield (1984) and Movellan (1990). While this conditional stability might be considered a weakness by some, it does have the advantage of bolstering the computational importance of lateral inhibition beyond the intuitive notions of selecting an active output unit.

Weight Update Rule Maximization Properties

Having established that the IAC activation function will result in a stable pattern of activation for a given set of weights, we must now determine if the weight update rule will move the global Goodness function upwards at every step. Given that the weight update rule was designed to take advantage of the pattern covariance, it should not be surprising that it will adjust the weights in the direction of increasing Goodness, which includes the pattern covariance over the input layer. This can be shown by the following analysis, which proceeds in the same way as the stability of the activations proof, except that we now use the equilibrium state of the network as the point at which the weights are changed. If weights were changed prior to the network reaching equilibrium, the activation patterns would not necessarily reflect the optimum Goodness level for the current weights, and it would be difficult to ensure that the change in weights reflects a step towards greater Goodness.

Of course, it would be nice not to have to assume that weights only change at points of activational equilibrium, but two points can be made regarding this problem. First, one could relax the constraint somewhat when simulating the network, and explore the practical consequences of updating the weights prior to the equilibrium point. Second, since a large Harmony term contributes to a larger overall G term at equilibrium (which is a point where G is at a maximum according to the previous analysis), this will be a time when a unit is receiving a relatively high level of input from other units, which would correspond in real neurons to high frequency synaptic transmissions and a high level of postsynaptic depolarization. This correlates well with what is known about the synaptic modification mechanisms involved in LTP, which depend on the postsynaptic neuron being depolarized, and most studies of this phenomenon induce synaptic modification with high frequency bursts of electricity. So, while speculative, there is some evidence that nature may in fact be implementing something akin to the equilibrium weight adjustment constraint.

We begin the analysis by computing $\frac{d\breve{G}}{dw_{ij}}$ (where ˘ again indicates equilibrium) for a given weight from the input layer unit i to the output layer unit j. There are several variables in $\breve{G}$ which depend on weight w_{ij}, so we first decompose the derivative into these partial terms:

$$\frac{d\breve{G}}{dw_{ij}} = \frac{\partial \breve{G}}{\partial w_{ij}} + \sum_k \frac{\partial \breve{G}}{\partial \breve{a}_k} \frac{\partial \breve{a}_k}{\partial w_{ij}} \tag{24}$$

where k is now an index over *all* units in the network. The first of these terms is just

$$\frac{\partial \breve{G}}{\partial w_{ij}} = (\breve{a}_j - \langle \breve{a} \rangle^{out})(\breve{a}_i - \langle \breve{a} \rangle^{in}) \tag{25}$$

which is exactly the correlational learning rule proposed at the outset. Fortunately, several different analyses (Movellan, 1990; Hinton, 1989) have shown that the $\frac{\partial \breve{G}}{\partial \breve{a}_k}$ is zero at equilibrium, so that this second term is 0. Intuitively, this should be so because evaluating $\frac{\partial G}{\partial a_k}$ at $a_k = \breve{a}_k$ will be 0 according to the activation stability proof given above. While this form of the proof is not entirely rigorous, it provides at least an intuitive justification for only considering the first term in equation 24. Refer to Hinton (1989) for a geometrical argument of the proof. This concludes our proof that the ZSH weight update function will result in an increase in the Goodness function G.

Having shown that the activations are stable according to the modified Goodness measure and that the ZSH learning rule works to increase the overall Goodness of a network, one can be reasonably confident that a ZSH network will converge on a stable, "good" mapping for a given set of input patterns. Exactly what kinds of representations will develop depends on the specific environment a network is exposed to. However, the general result will be that output units will tend to differentially represent different input patterns.

Simulation 2

The purpose of this simulation is to evaluate the potential of the zero-sum Hebbian learning rule just developed on the same task as was used in Simulation 1. However, since there is no error-driven component to the ZSH learning rule, it is not appropriate to give it the auto-encoder task. In dropping this component of the network, we also lose the value of the separate "where" pathway, since the total information present in the input does not need to be preserved for the auto-encoder output. Essentially, we are only interested in exploring the role of hysteresis and temporal contiguity in the environment in the real Hebbian system. The architecture used was simply a three-layer network consisting of 5x2 units in each layer. The large number of units in the upper two layers were used because a reduced number would have been an additional assumption and constraint on our part about the kinds of representations that are to develop. One of the benefits of a self-organizing network of this kind is that it will use just as many units in each layer as it requires, and additional units will become inactive "dead" units.

The use of two layers above the input layer was important as it allowed for hysteresis from the recurrent activation loops to develop between units in these two layers, which are connected with recurrent, excitatory weights. Within each layer, there are inhibitory weights. It was not clear at the outset where the invariant representations would develop, as they could be in either of the upper two layers (see Figure 6 in the Results section below for a diagram of the architecture).

Using this architecture, two goals were pursued. One was to determine what the relative impact the recurrent activation loops and lateral inhibition have on the development of spatial invariance, and the other was to ask a similar question regarding the same-object temporal continuity probability (i.e., the probability that the next image represents the same object as the previous one) in the environment. The answer to the first question will further establish the validity of the hypothesized mechanism, and determine what levels of these variables are necessary for it to work. This information can then be compared to what is known about the physiology of visual cortex to determine if the mechanism is biologically plausible.

The answer to the second question will tell us how robust the mechanism is. If it requires 100% same-object probability to work at all, then the value of this mechanism will be very limited. However, if it can work with relatively low same-object probabilities, then the mechanism holds real-world promise. Average same-object probabilities can be computed for the looking behavior of young children, and these values tested in the network.

Methods

For the first part of this simulation, only half of the desired variable can be manipulated, because lateral inhibition is an intrinsic part of the stability of the network, as was proven above. Therefore, we will only manipulate the relative strength of the recurrent activation connections between layers 1 and 2 (the input layer is 0). The gain of these weights can be set by a layer-wide parameter. While both the feed-forward weights from layer 1 to layer 2 and the feed-back weights from layer 2 to layer 1 are involved in a recurrent activation loop, only the feed-back weights were manipulated, as it is these weights which determine the stabilizing influence of the activation of a layer 2 unit on units in layer 1. This gain value was varied from .75 to 0 in steps of .25, where a gain of 1 was used on the feed-forward weights in the system, and a lateral inhibition gain of 4 was used. This relatively large degree of lateral inhibition was found to be necessary to ensure that only one unit in each layer was active at any given time. While a distributed representation having multiple units active would be more realistic, the simplicity of analyzing the local representations encouraged by strong lateral inhibition was preferable in the present situation.

In the second part of the present simulation, the same-object probability was varied from 1.0 to .25, with intermediate values of .90, .75, .60, and .50. In the "best-case" condition with a SOP (Same-Object Probability) of 1.0, there was one blank stimulus in between each set of objects. As was discussed in Simulation 1, this blank stimulus simulates the (presumed) effects of attention, which would deactivate the currently active units during a shift of attention, such as that which would accompany viewing a different object. However, for the lower SOP values, it was not possible to preserve the presence of this blank stimulus after *every* shift in object type, so that the blank stimulus appeared randomly after a shift from the present object. The algorithm for computing which stimulus to present simply selected another position of the same object with the given SOP probability, or another stimulus that was *not* the same object. This included the various positions of the other objects, and the two blank stimuli. In order to determine the relative importance of the blank stimuli, their presence was manipulated for the "best-case" and .90 probability networks.

For networks with lower SOP values, it is important to balance the perseveration of the higher-layer activations with the ability to change state upon presentation of a different object. For this reason, all simulations included a short decay period during which no stimuli were presented was inserted between each stimulus presentation (i.e., one position of one object). This decay period lasted for 50 time steps. This period would correspond to the time during the actual movement of the eye during a saccade.

Each network was run with 10 random sets of initial weights, and the following parameters: weight learning rate (dW) = .1, activation step (λ) = .1, decay = .1, max = 1, min = -1. The weights were adjusted after the network settled in activation space so that the maximum change in activation was less than .0001, or after 500 time steps, which ever came first. Typical settling times were on the order of 100-150 time steps.

Results

The scores for spatial invariance were computed in much the same way as those in Simulation 1, except that the total score could now be used as a unit only sends information when it has a positive activation, thus ensuring that both objects will be represented by

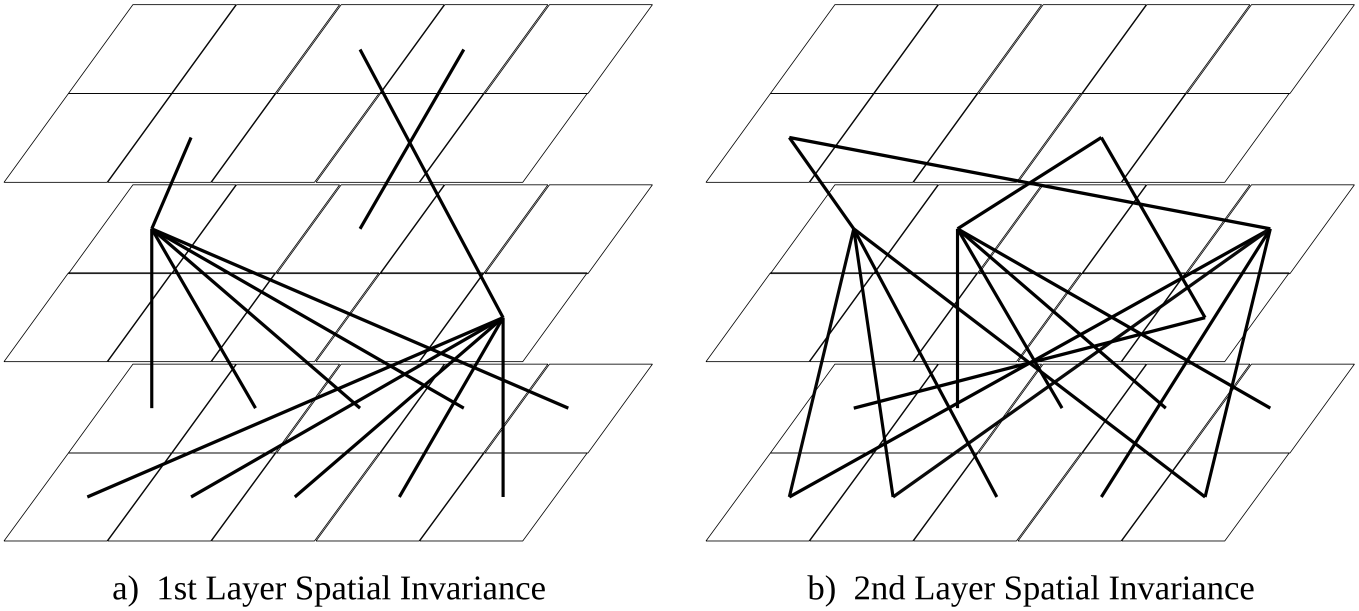

a) 1st Layer Spatial Invariance
b) 2nd Layer Spatial Invariance

Figure 6: **a)** The architecture of the network used in Simulation 2, showing only the strong weights which developed after training. This network developed spatial invariance in Layer 1 (Layer 0 is the input), as is indicated by the presence of strong weights to each position of one object to a single Layer 1 cell. **b)** Shows a different network which developed complete spatial invariance only in Layer 2, which combined partially invariant representations from Layer 1.

units with positive weights. As there were 10 units to choose from in each of the two layers, the score was computed on that unit which actually responded to a given object the most. In most cases, this unit was in layer 2, but in some cases the layer 2 representations captured both objects, so that layer 1 units were scored in this case. Figure 6 shows two examples of trained networks that developed spatially invariant representations in different layers.

Figure 7 shows the results from the first part of the simulation. The direction of the effect is as predicted, with lower gain values reducing the degree of spatial invariance produced. Only the .75 condition was significantly different than the 0 condition ($p < .05$, Bonferroni/Dunn), but the .75 condition was also nearly significantly different than the .25 condition ($p < .06$). No other differences were significant.

Figure 8 shows the results from the second part of the simulation. Again, the direction of the effect is as predicted, with decreasing SOP leading to lower spatial invariance scores. Indeed, the effect is linear with the function slope and intercept shown in the figure with a fit of $R^2 = .98$ to the data. In order to compare these SOP levels with real-world behavior, one can estimate the average time that an individual would need to be looking at a given object to produce a particular probability that the next saccade will be of the same object. This probability is dependent upon the rate of saccading, as this determines the effective time step length. As an example, if one saccades every 200ms, and continues to look at a given object for two seconds (2000ms), then there will be 9 consecutive instances of a given object

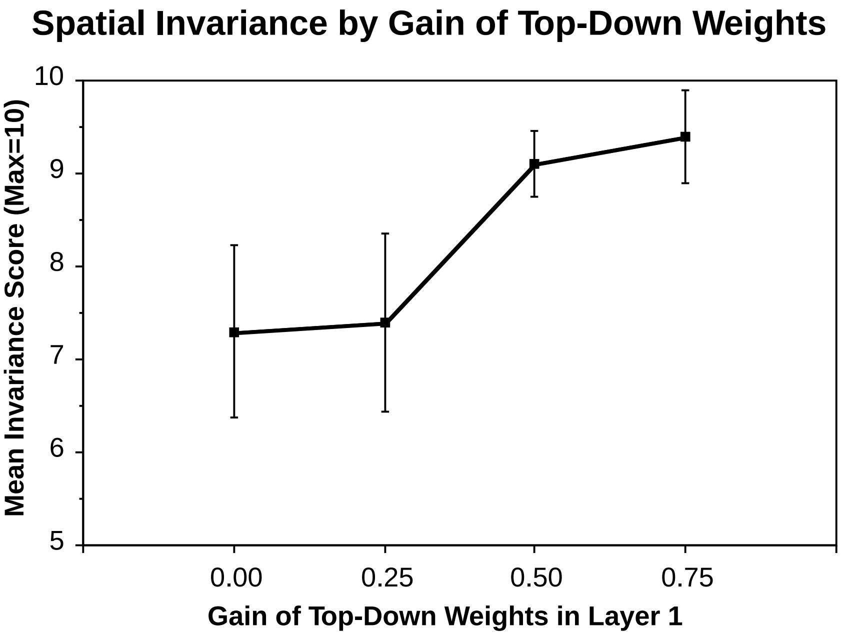


Figure 7: Results of manipulating gain of top-down weights for layer 1 of zero-sum network, showing existence of a critical level of top-down weights

followed by a 10th on a different object. Thus, there would be a .90 SOP level associated with this level of observation. This level would, according to these simulations, result in a high degree of spatial invariance developing.

As for the use of a blank stimulus to simulate attentional effects, the mean invariance score over 10 "best-case" (SOP = 1.0) networks dropped from 9.4 to 6.2 with the elimination of the blank stimuli, with a significance level of ($p < .0001$). For the .90 SOP network, the score dropped from 8.2 to 6.1 with the elimination of the blank stimuli, (significant at $p < .01$). Thus, the effect of the blank stimulus was less significant for the lower SOP network, as would be predicted from the more random nature of its appearance.

Discussion

The principal lessons to be learned from this simulation are that the predicted effects of manipulating the strength of the recurrent activation loops were found, and a surprisingly linear relationship between the spatial invariance and the SOP statistics of the environment was observed. The first of these findings confirms that the spatial invariance effect is developing according to the hypothesized mechanism. The second of these findings was not predicted, but is a suggestive result in light of the Rational Analysis theory of human cognition recently discussed by Anderson (1990). Perhaps such sensitivities to environmental statistics are not indicative of rationality as much as they are of certain underlying mechanisms which depend critically on such statistics.

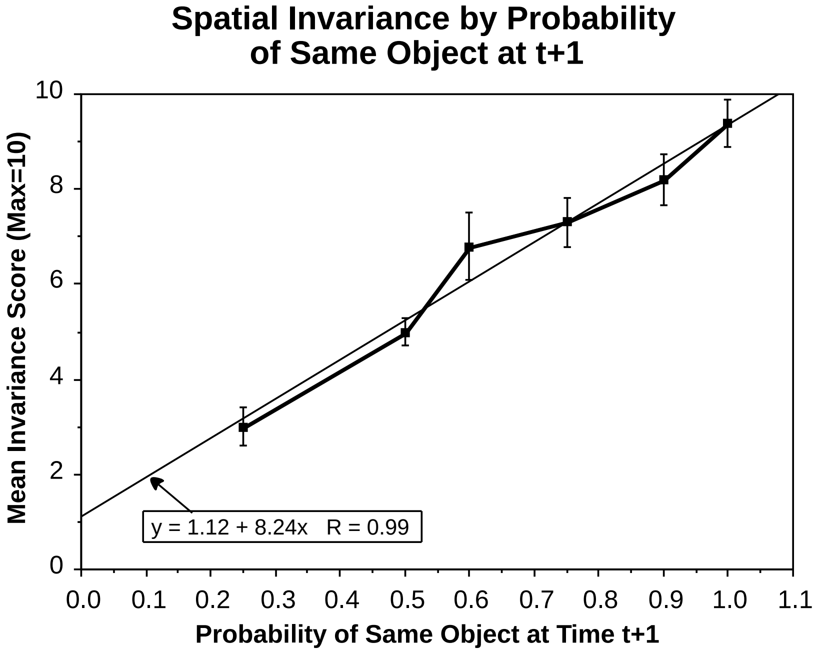


Figure 8: Results of manipulating the probability of seeing the same object at the next time step (t+1), showing linear relationship between SOP and amount of spatial invariance developed

The need for a blank stimulus between objects to prevent the perseveration of the higher-order activation patterns over multiple different objects was clearly demonstrated for these networks. The central problem with this requirement is that it could be construed as circular, in that one must somehow know that one's gaze has shifted to a different object in order for some kind of attentional mechanism to be activated. An object recognition system can not rely on attentional mechanisms which in turn rely on the existence of an object recognition system. However, several arguments could be made for why this would not necessarily hold for larger, more realistic networks. First, the spatial invariance transformation is supposed to occur gradually over features of increasing complexity. For most levels of this transformation, the features will be common to many different objects, so that perseveration over different objects would not be as damaging as it is in the network studied. Second, there are many potential lower-order visual features that could be used to signal a shift in the object being observed, without actually requiring the identification of the object. These features center around figure/ground separation, and include such things as cohesivity of motion, illuminance, and other stimulus qualities over a region of space. Differences in these lower order features could signal the presence of a different object, and serve to inhibit the higher order object recognition units.

While the results from this simulation are encouraging and interesting, they do not yet constitute a convincing demonstration that a more realistic environment could be handled by such a mechanism. Thus, in the next simulation, we will take the environment one step further and introduce very simple visual objects at the level of a retinal image as input to the system.

Simulation 3

The critical difference between the model explored in the present simulation and that of Simulation 2 is in the stimuli presented to the network. The stimuli for this model were intended to more closely represent some of the complexity of actual retinal images. In addition, we wanted to capture at least some of the feature-level properties of the proposed invariance algorithm, so that the images were of single features from which objects might be composed, in this case a diagonal line of either the right or left-leaning orientation. Unlike the stimuli for Simulations 1 and 2, the same input units are used to represent both "objects," so it is not possible for the network to simply develop strong excitatory weights to an entire portion of the input and still distinguish between the two stimuli (as was the case in the previous simulations). Instead, the network must develop retinotopic representations of the features, which can then be combined into spatially invariant representations at higher levels. These retinotopic representations of features were assumed as input to the previous models, but now we want to see if the network will develop them in response to the environment.

As can be seen in Figure 9, the network has five layers of $6x3$ units, except the input layer (the "retina"), which has $8x4$ units. Based on the findings of Simulation 2 regarding the different layers at which invariance developed, it was anticipated that a hierarchical structure of spatially invariant representations would develop, with higher layers encompassing more positions. The figure shows a trained network which has developed spatially invariant representations in layer 3. These representations are built upon the locally spatially invariant representations developed in layer 2 (shown in Figure 9e), which in turn are built upon the retinotopic feature representations developed in layer 1 (shown in Figure 9f).

Methods

The network was trained in much the same way as in the previous simulation, with five different positions of the two diagonal lines presented with a specified SOP. The SOP was varied from 1.0 to .5 with levels of .9 and .75 in between. The same activation function parameters and learning rate from Simulation 2 were used. The gain factors for the weights were the same as well, with .75 for top-down weights, 4 for lateral inhibitory weights, and 1 for bottom-up weights, except for the units in layer 1, which had a gain of .5 for the bottom-up weights to compensate for the larger number of active units projecting to these units.

Initially, a difficulty arose in the development of the layer 1 retinotopic feature representations due to the large receptive fields of these units. Because the zero-sum learning algorithm decreases weights to inactive units in proportion to the average activation over a unit's receptive field, a large number of inactive input units will dilute the amount of weight taken from each inactive input unit. As a layer 1 unit developed strong weights to the four input units comprising a given line, each of these accentuated units was shared with a different diagonal of the opposite direction, causing this unit to respond more favorably to those other diagonals. The zero-sum Hebbian algorithm will compensate for this effect by reducing the weights to all the other units that were not part of a given diagonal line, including the three other units comprising a diagonal going the other way. However, because this weight decrease is normalized over all the inactive units (28 out of the 32 total), more weight increase than decrease results to this other diagonal line. This effect caused a given layer 1 unit which responded to a diagonal line of one orientation to be more likely to respond to a diagonal line of the opposite orientation which shared one of the input units with the other line, as the weight had increased to that unit, and a proportionally large amount of weight had not been taken from the other inactive input units. This, compounded with the perseverative effects from the lateral inhibition and top-down weights caused many layer 1 units to respond to multiple diagonal lines of both orientations.

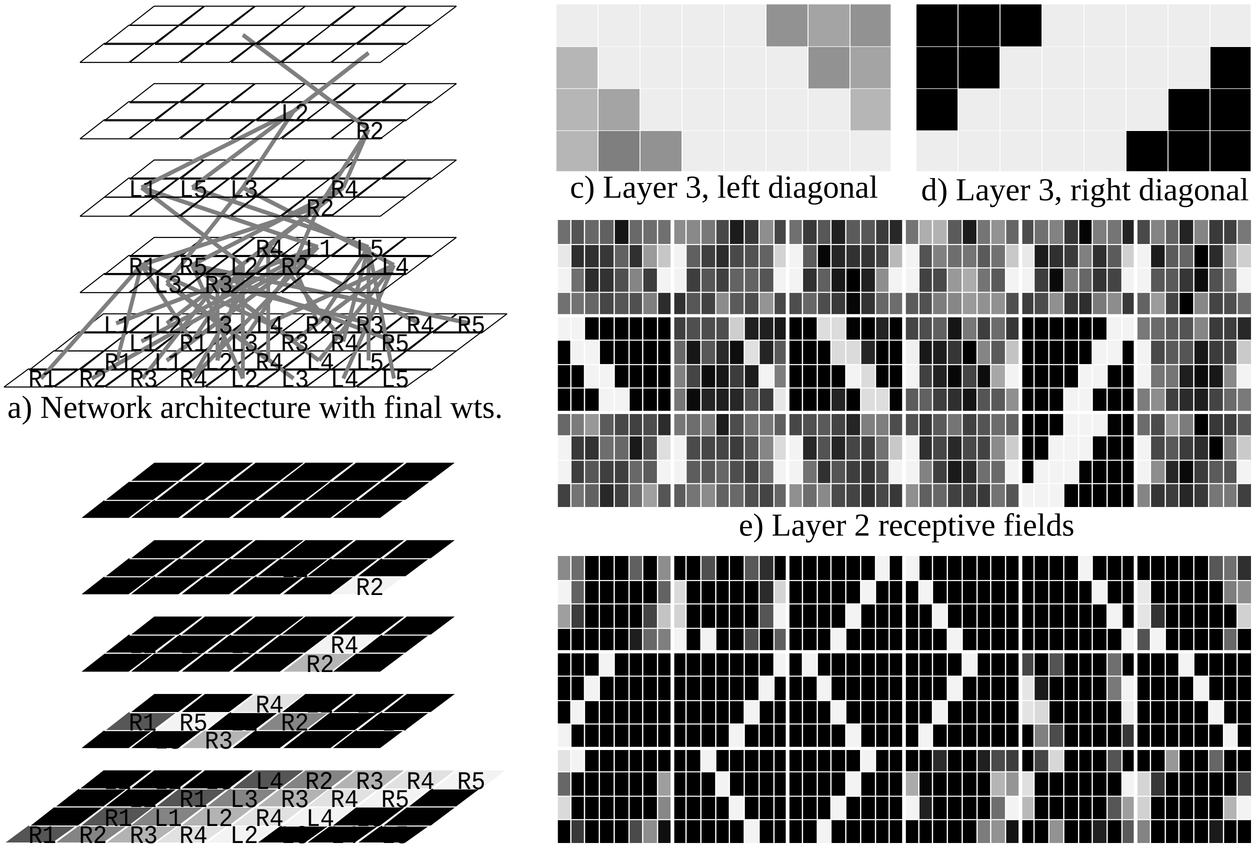
In order to reduce the number of "blends" in the retinotopic feature representations, the receptive fields of the layer 1 units were reduced to cover only 50% of the input layer. The connections were distributed according to a Gaussian distribution based on the distance between a given layer 1 unit and the input units, with a sigma of .15 (defined over the maximum distance across the the layer), yielding retinotopic receptive fields centered below the layer 1 unit. This reduced the blending problem significantly, but it did not eliminate it entirely. Given that the problem was due to an insufficient degree of weight decrease to inactive units, we simply multiplied the computed amount of weight decrease by a factor of 2. In a more realistic case where features were represented by distributed patterns of activity, blends would not be a problem, and these manipulations would be unnecessary.

In addition, the top-down weights for layer 1 were set to zero to obtain a minimum of blending over different features at this layer. However, a set of 10 1.0 SOP networks were run with this parameter set to .75 to determine the relative importance of this factor.

Results

The most important result of this simulation is to demonstrate that the entire pathway from retina to object recognition can be simulated in the same network, and that the representations which developed follow a gradient of increasing spatial invariance, as can be seen in Figure 9. In order to ensure that this network has the same robust qualities for lower SOP values, a range of SOP values were used (1.0, .90, .75, .50). The performance of the network was similar to those of Simulation 2, as can be seen in Figure 10. This network had a negative Y intercept for the linear fit, but a larger slope, as compared to Simulation 2. However, this line fit the data well at $R^2 = .98$.

The best-case performance of this network was similar to the best-case network from Simulation 2, with a mean of 9.2 *vs* 9.4 *ns*. As with the Simulation 2 best-case networks, the only deviation from a perfect invariance over all 10 networks came from a network



a) Network architecture with final wts.

c) Layer 3, left diagonal

d) Layer 3, right diagonal

e) Layer 2 receptive fields

b) Trace of right diagonals.

f) Layer 1 receptive fields

Figure 9: Network architecture for Simulation 3, showing a network that developed spatially invariant representations in layer 3 (with the input layer defined as layer 0). **a)** shows the weights for the network, giving an idea of the hierarchical nature of the representations. **b)** shows an "activation trace" of the right-diagonal stimulus as it sweeps across the input layer. In this figure, the greyscale represents when the unit was last active, with lighter shades representing those that were active most recently. This view shows how activity on the retina has a decreasing influence as one goes higher into the network, so that by layer 3, the same unit is active for all positions of the stimulus. **c** & **d** show what the receptive fields for the two spatially invariant units in layer 3 look like. These receptive fields were generated by projecting (convolving) the input weights to these units through the input weights to the units in the layer below, and so on down to the input layer. **e)** shows the receptive fields for all of the layer 2 units. Each unit in layer 2 is represented in its respective location, with its receptive field. The units in layer 2 are separated by thicker white lines, while those in the receptive field are separated by thin white lines. **f)** shows the receptive fields for layer 1.

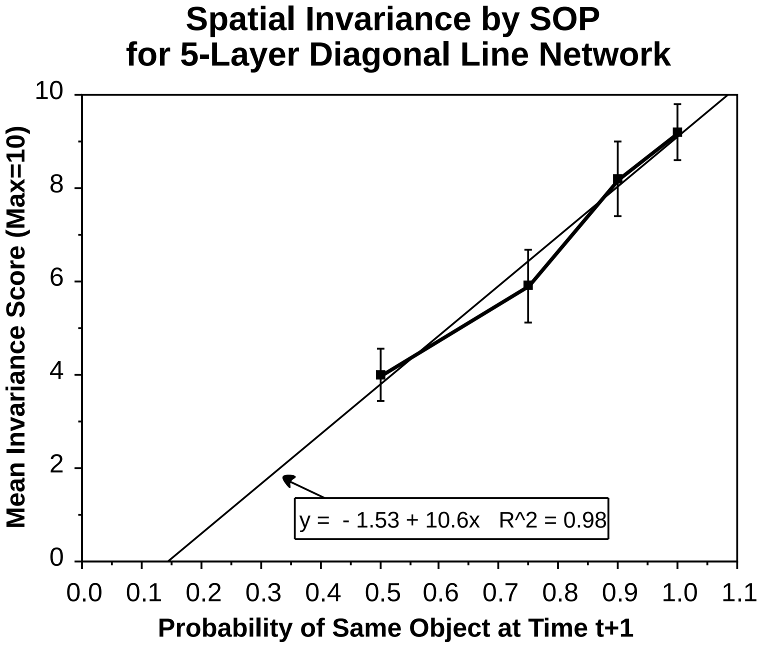


Figure 10: Results of manipulating the probability of seeing the same object at the next time step (t+1), showing linear relationship between SOP and amount of spatial invariance developed

which developed too much invariance, such that it summed over both types of objects in all positions. In this situation, the maximal spatially invariant representations were at a lower level in the network, which caused them to encompass less of the different positions of a given object.

Another notable feature of these networks was the degree to which individual lines got combined in the retinotopic feature representations in Layer 1, as was discussed above. To compare the relative importance of the various Layer 1 manipulations used in the networks reported above, the following conditions were run and analyzed: **1.** Uniform parameters for all layers, with full receptive field size in Layer 1, weight decrease = 1.0, and top-down weight gain = .75. **2.** Same as 1 but with 50% receptive fields, and top-down gain of 0. **3.** Same as 1 but with 50% Layer 1 receptive fields, and weight decrease factor of 2 (note that the weight decrease factor affects all units in the network, as it is a global parameter). Comparing these conditions to the best-case network (Layer 1 receptive field = 50%, weight decrease = 2, top-down gain = 0), the spatial invariance scores drop off to 5.5 for condition 1, 5.1 for condition 2, and 6.6 for condition 3. Only the difference between the first two conditions and the best-case network was statistically significant ($p < .05$, Bonferroni/Dunn). Further, the average number of blended diagonal line representations was significantly higher for conditions 1 and 2 ($p < .05$, Bonferroni/Dunn), going from 2.0 for the best-case network (one blend encompassing two diagonal lines) to 4.5 for condition 1 and 3.8 for condition 2. Condition 3 yielded 2.6 *ns.* Thus, both the smaller receptive fields and increased level of weight decrease were important, but the top-down weight gain was less so.

Even in the best-case network, an average of one retinotopic feature blend per network was observed. However, the network was able to still distinguish which line was present by using the context of a previously activated line position. This was an interesting finding, as it suggests a mechanism for the representation of temporal context effects.

Discussion

The critical insight for explaining why the hierarchical structure of spatially invariant representations develops is illustrated in Figure 9.b, which shows how the changes in activation in the input layer have a decreasing influence on subsequent layers of the network, such that by layer 3, a single unit can remain active during the presentation of all 5 positions of a given feature. This gradient of stability over the layers of the network comes from the combined effects of recurrent activation loops between layers and the lateral inhibition within layers, and it is this gradient which is hypothesized to be capable of producing a corresponding gradient of spatially invariant representations.

The network modifications necessary to make the feature detectors in layer 1 local and distinct were important for this network and the stimuli used, as is clear from the results when they were not in place. However, the relative merits of these modifications is not important, given that more realistic stimuli and representational structures (e.g., distributed representations) should not require them. These more realistic models should be explored, however, to determine if this is actually the case.

In addition to the question of distributed representations, the role of the integration of increasingly complex representations of object features over increasing layers of the network needs to be explored in order to completely duplicate the algorithm suggested by Mozer. Both of these considerations require larger networks than those used here. Having established the validity of the learning mechanism in these simpler networks, it would not be unreasonable to scale up the models to include more complex stimuli that can be decomposed into features for recognition purposes.

Conclusions

Having demonstrated in simulated neural networks that certain patterns of neural interconnectivity combined with a Hebbian learning rule are capable of exploiting temporal regularities in the environment, many interesting predictions of this model can be explored. For example, the model would predict that areas of the brain responsible for object-based representations would have higher degrees of lateral inhibition and recurrent activation loops in order to maintain a perseveration of object-based representations. Further, areas of the brain that are designed to represent other aspects of visual information, such as spatial location (the “where” pathway), should not have as much of these effects in order to *avoid* developing object-based representations.

Several pieces of physiological data fit these requirements. First, it has been shown that the earliest projections from the retina to the LGN can be divided into two basic types, known as the *magnocellular* and *parvocellular* pathways (Livingstone & Hubel, 1988; Maunsell et al., 1990). These pathways have many different properties, temporal response characteristics being one of them. The magno cells respond quickly and briefly to a given stimulus, while the parvo cells have slower, longer lasting responses. While these properties do not derive directly from the wiring patterns of cortex such as lateral inhibition or recurrent connections, they do represent different kinds of input to higher visual areas. According to the theory developed herein, one would predict that the parvo pathway, because of its temporally extended response properties, would project more to the areas responsible for object recognition, while the magno pathway would target the location-specific areas. This is exactly what happens. Maunsell et al. (1990) have shown by differentially disabling the magno and parvo pathways in LGN that disabling the magno pathway has large effects on the Middle Temporal (MT) visual area, which subsequently projects to the parietal "where" pathway, while these effects are not found for the parvo pathway.

Livingstone & Hubel (1988) have shown that many behavioral dissociations between the kinds of stimulus properties that people can use to generate different kinds of visual information such as depth and motion correlate with the differences in the magno and parvo pathways. These results support the notion that these pathways project to different areas of higher visual cortex. However, at the end of their paper, Livingstone & Hubel (1988) ask "Is the existence of separate pathways an accident of evolution or a useful design principle?" (p. 748). The present analysis suggests that indeed these two different pathways could have important computational implications for the kinds of representations developed from inputs differing in the temporal continuity of their responses.

Finally, it appears that the portion of the domestic chick brain responsible for object recognition in filial imprinting and other tasks has the specific form of neural interconnectivity hypothesized in our model: recurrent excitatory connections and lateral inhibition. Because of the relative simplicity of the avian brain compared to that of the human, it is an ideal system in which to explore the behavioral and neural implications of an object recognition system like the one we have proposed. Indeed, in collaboration with Mark Johnson, the present model has been successfully applied to several behavioral phenomena associated with imprinting. The details of this work are available in O'Reilly & Johnson (Submitted). The success of our chick model in accounting for a range of behavioral data supports the hypothesis that at least some biological systems perform object recognition using the algorithm proposed herein.

Analysis of Goodness Function

The following is the derivative of the Goodness function suggested in the text for the ZSH learning algorithm with respect to a given output unit a_j (where $j \in \{k\}$):

$$\frac{\partial G}{\partial a_j} = \frac{\partial H}{\partial a_j} - \frac{\partial S}{\partial a_j} \tag{26}$$

Taking the $\frac{\partial H}{\partial a_j}$ term first results in the following equation, given that $\langle a \rangle^{out}$ is really just a sum over all output units k. Thus, it reduces to $\frac{\partial \frac{a_j}{N}}{\partial a_j}$, which is just $1/N$.

$$\frac{\partial H}{\partial a_j} = \frac{N-1}{2N} \left(\sum_i (a_i - \langle a \rangle^{in}) w_{ij} + \sum_i (a_i - \langle a \rangle^{in}) w_{ji} + \sum_{l, l \neq j} a_l w_{lj} + \sum_{l, l \neq j} a_l w_{jl} \right) \tag{27}$$

For the remainder of the analysis, we will assume that the number of output layer units N is large enough to consider $\frac{N-1}{2N} \approx \frac{1}{2}$ (the difference is not important for the results).

Clearly, this equation would be much simpler to analyze if the feedforeward and feedback weights were the same (i.e., that $w_{ij} = w_{ji}$), and that the lateral inhibition weights were also symmetric ($w_{lj} = w_{jl}$). This condition is discussed in the text.

Assuming symmetric weights, equation 27 can be re-written as:

$$\frac{\partial H}{\partial a_j} = \sum_i a_i w_{ij} + \sum_{l, l \neq j} a_l w_{lj} - \sum_i \langle a \rangle^{in} w_{ij} \tag{28}$$

which is just a combination of the net input to a given unit (I_j^{net}, see equation 18) plus an additional term:

$$\frac{\partial H}{\partial a_j} = I_j^{net} - \langle a \rangle^{in} \sum_i w_{ij} \tag{29}$$

This leaves the $\frac{\partial S}{\partial a_j}$ term, which is relatively simple because the derivative of the integral of a function is the function itself

$$\frac{\partial S}{\partial a_j} = f^{-1}(a_j) \tag{30}$$

Which makes the derivative of G

$$\frac{\partial G}{\partial a_j} = I_j^{net} - \langle a \rangle^{in} \sum_i w_{ij} - f^{-1}(a_j) \tag{31}$$

This gives us the derivative of our global Goodness function in terms of any output layer unit. In order to determine if the IAC activation function will always cause this derivative to increase over time, we must apply the chain rule using the derivative of the activation function over time (see equation 16) to get the derivative of G over time:

$$\frac{dG}{dt} = \sum_j \frac{\partial G}{\partial a_k} \frac{da_k}{dt} \tag{32}$$

where j indexes over all units in the output layer. This reduces to:

$$\frac{dG}{dt} = \lambda \sum_j \left[\left(I_j^{net} - \langle a \rangle^{in} \sum_i w_{ij} - f^{-1}(a_j) \right) \left(-a_j + f(I_j^{net}) \right) \right] \tag{33}$$

Because there are similar terms in both of these expressions, it is possible to predict the conditions under which this expression will have the desired positive sign. Following (Movellan, 1990), we realize that the $\frac{\partial H}{\partial a_j}$ term $(I_j^{net} - \langle a \rangle^{in} \sum_i w_{ij} - f^{-1}(a_j))$ will have the same sign as $f(I_j^{net} - \langle a \rangle^{in} \sum_i w_{ij}) - a_j$ because the activation function f is monotonic and will therefore preserve the sign of its parameter. In comparing this new expression to the $\frac{da_j}{dt}$ term $(-a_j + f(I_j^{net}))$, it should be clear that they will both have the same sign whenever $I_j^{net} - \langle a \rangle^{in} \sum_i w_{ij}$ has the same sign as I_j^{net}. Thus, the sum in equation 33 is over the product of two terms having the same sign, which must be positive, whenever $I_j^{net} - \langle a \rangle^{in} \sum_i w_{ij}$ has the same sign as I_j^{net}. The implications of this condition are explored in the text.

References

Anderson, J. (1990). *The Adaptive Character of Thought.* Lawrence Earlbaum Associates, Inc., Hillsdale, NJ.

Artola, A., Brocher, S., & Singer, W. (1990). Different voltage-dependent thresholds for inducing long-term depression and long-term potentiation in slices of rat visual cortex. *Nature*, 347:69–72.

Barnard, E. & Casasent, D. (1990). Shift invariance and the neocognitron. *Neural Networks*, 3:403–410.

Bear, M. F. & Cooper, Leon N.and Ebner, F. F. (1987). A physiological basis for a theory of synapse modification. *Science*, 237:42–48.

Bear, M. F. & Cooper, L. N. (1990). Molecular mechanisms for synaptic modification in the visual cortex: Interaction between theory and experiment. In (Gluck & Rumelhart, 1990), chapter 2, pages 65–93.

Bekkers, J. M. & Stevens, C. F. (1990). Presynaptic mechanism for long-term potentiation in the hippocampus. *Nature*, 346:724–729.

Biederman, I. (1987). Recognition-by-components: A theory of human image understanding. *Psychological Review*, 94(2):115–147.

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

Bliss, T. (1990). Maintenance is presynaptic. *Nature*, 346:698–699.

Bradler, J. & Barrionuevo, G. (1990). Heterosynaptic correlates of long-term potentiation induction in hippocampal CA3 neurons. *Neuroscience*, 35(2):265–271.

Brown, T. H., Ganong, A., Kairiss, E., Keenan, C., & Kelso, S. (1989). Long-term potentiation in two synaptic systems of the hippocampal brain slice. In (Byrne & Berry, 1989), chapter 14, pages 266–306.

Byrne, J. H. & Berry, W. O., editors (1989). *Neural Models of Plasticity: Experimental and Theoretical Approaches.* Academic Press, Inc., San Diego, CA.

Collingridge, G. & Bliss, T. (1987). NMDA receptors - their role in long-term potentiation. *Trends In Neurosciences*, 10:288–293.

CSS, editor (1988). *Proceedings of the 10 th Conference of the Cognitive Science Society*, Hillsdale, NJ. Lawrence Earlbaum Associates, Inc.

DeSieno, D. (1988). Adding a conscience to competitive learning. In IEEE, editor, *Proceedings of the IEEE International Conference on Neural Networks*, pages 117–124, New York. IEEE.

Douglas, R. J. & Martin, K. A. C. (1990). Neocortex. In (Shepherd, 1990), chapter 12, pages 389–438.

Duncan, J. (1984). Selective attention and the organization of visual information. *Journal of Experimental Psychology: General*, 113:501–517.

Fazeli, M. (1992). Synaptic plasticity: on the trail of the retrograde messenger. *Trends In Neurosciences*, 15(4):115–117.

Földiák, P. (1991). Learning invariance from transformation sequences. *Neural Computation*, 3(2):194–200.

Frégnac, Y., Shulz, D., Thorpe, S., & Bienenstock, E. L. (1988). A cellular analogue of visual cortical plasticity. *Nature*, 333:367–370.

Fukushima, K. (1988). Neocognitron: A hierarchical neural network capable of visual pattern recognition. *Neural Networks*, 1:119–130.

Gibson, J. J. (1979). *The Ecological Approach to Visual Perception.* Houghton Mifflin, Boston.

Gluck, M. A. & Rumelhart, D. E., editors (1990). *Neuroscience and Connectionist Theory.* Lawrence Earlbaum Associates, Inc., Hillsdale, NJ.

Grossberg, S. (1976). Adaptive pattern classificaiton and universal recoding. I: Parallel development and coding of neural feature detectors. *Biological Cybernetics*, 23:121–134.

Hebb, D. O. (1949). *The Organization of Behavior.* Wiley, New York.

Hertz, J., Krogh, A., & Palmer, R. G. (1991). *Introduction to the Theory of Neural Computation.* Addison-Wesley, Redwood City, CA.

Hinton, G. E. (1981). Shape representation in parallel systems. In *Proceedings of the 7th IJCAI*, pages 1088–1096, Vancouver.

Hinton, G. E. (1989). Deterministic boltzmann learning performs steepest descent in weight space. Technical Report CRG-TR-89-1, University of Toronto.

Hopfield, J. J. (1984). Neurons with graded response have collective computational properties like those of two-state neurons. *Proc. Natl. Acad. Sci. USA*, 81:3088–3092.

Hubel, D. & Wiesel, T. N. (1962). Receptive fields, binocular interaction, and functional architecture in the cat's visual cortex. *Journal of Physiology*, 160:106–154.

Levy, W. B., Colbert, C. M., & Desmond, N. L. (1990). Elemental adaptive processes of neurons and synapses: A statistical/computational perspective. In (Gluck & Rumelhart, 1990), chapter 5, pages 187–235.

Linsker, R. (1986). From basic network principles to neural architecture (a three-part series). *Proc. Natl. Acad. Sci. USA*, 83:7508–7512, 8390–8394, 8779–8783.

Linsker, R. (1988). Self-organization in a perceptual network. *Computer*, pages 105–117.

Livingstone, M. & Hubel, D. (1988). Segregation of form, color, movement, and depth: Anatomy, physiology, and perception. *Science*, 240:740–749.

Lynch, G. & Baudry, M. (1984). The biochemistry of memory: a new and specific hypothesis. *Science*, 224:1057–1063.

Marr, D. (1982). *Vision.* Freeman, New York.

Marshall, J. A. (1990). Self-organizing neural networks for perception of visual motion. *Neural Networks*, 3:45–74.

Maunsell, J. H. R., Nealey, T. A., & DePriest, D. D. (1990). Magnocellular and parvocellular contributions to responses in the middle temporal visual area (MT) of the macaque monkey. *The Journal of Neuroscience*, 10(10):3323–3334.

McClelland, J. L. (1981). An interactive activation model of context effects in letter perception: Part 1. an account of basic findings. *Psychological Review*, 88(5):375–407.

McNaughton, B. L. (1988). Neuronal mechanisms for spatial computation and information storage. In Nadel, L. A., Cooper, L. A., Culicover, P., & Harnish, R. M., editors, *Neural Connections, Mental Computation*, chapter 9, pages 285–350. MIT Press, Cambridge, MA.

McNaughton, B. L. & Morris, R. G. M. (1987). Hippocampal synaptic enhancement and information storage within a distributed memory system. *Trends In Neurosciences*, 10(10):408–415.

McNaughton, B. L. & Nadel, L. (1990). Hebb-marr networks and the neurobiological representation of action in space. In (Gluck & Rumelhart, 1990), chapter 1, pages 1–63.

Miller, K. D. (1990a). Correlation-based models of neural development. In (Gluck & Rumelhart, 1990), chapter 7, pages 267–353.

Miller, K. D. (1990b). Derivation of linear hebbian equations from a nonlinear hebbian model of synaptic plasticity. *Neural Computation*, 2:321–333.

Movellan, J. R. (1990). Contrastive hebbian learning in the continuous hopfield model. In Touretsky, D. S., Hinton, G. E., & Sejnowski, T. J., editors, *Proceedings of the 1989 Connectionist Models Summer School*, pages 10–17, San Mateo, CA. Morgan Kaufmann.

Mozer, M. C. (1987). Early parallel processing in reading: A connectionist approach. In Colthheart, M., editor, *Attention and Performance XII: The Psychology of Reading.*, pages 83–104, Hillsdale, NJ. Lawrence Earlbaum Associates, Inc.

Mozer, M. C. (1988). A connectionist model of selective attention in visual perception. In (CSS, 1988), pages 195–201.

Mozer, M. C. & Behrmann, M. (1990). On the interaction of selective attention and lexical knowledge: A connectionist account of neglect dyslexia. *Journal of Cognitive Neuroscience*, 96(2):96–123.

Oja, E. (1982). A simplified neuron model as a principal component analyzer. *Journal of Mathematical Biology*, 15:267–273.

O'Reilly, R. C. (1989). *Properties of a Self-Organizing Neural Network: The Unsupervised Generation of Hierarchical Representations.* PhD thesis, On File at Hilles Library, Harvard University.

O'Reilly, R. C. & Johnson, M. H. (Submitted). Object recognition and sensitive periods: A computational analysis of visual imprinting. Submitted.

O'Reilly, R. C., Kosslyn, S. M., Marsolek, C. J., & Chabris, C. F. (1990). Receptive field characteristics that allow parietal lobe neurons to encode spatial properties of visual input: A computational investigation. *Journal of Cognitive Neuroscience*, 2(2):141–155.

Perkel, D. H. & Perkel, D. (1985). Dendritic spines: Role of active membrane in modulating synaptic efficacy. *Brain Research.*

Phaf, R. H., Heijden, A., & Hudson, P. T. (1990). SLAM: A connectionist model for attention in visual selection tasks. *Cognitive Psychology*, 22:273–341.

Rall, W. (1990). Cable properties of dentrites. In Sejnowski, T. J., editor, *Methods in Neuronal Modeling*, chapter 1. Lawrence Earlbaum Associates, Inc., Hillsdale, NJ.

Rall, W. & Segev, I. (1987). Functional possibilities for synapses on dendrites and on dendritic spines. In Edelman, G., Gall, E., & Cowan, W., editors, *Synaptic Function*, pages 605–636. Wiley, New York.

Rolls, E. T. (1989). Functions of neuronal networks in the hippocampus and neocortex in memory. In (Byrne & Berry, 1989), pages 240–265.

Rueckl, J. G., Cave, K. R., & Kosslyn, S. M. (1989). Why are "what" and "where" processed by separate cortical visual systems? a computational investigation. *Journal of Cognitive Neuroscience*, 1:171–186.

Rumelhart, D. E. (1982). An interactive activation model of context effects in letter perception: Part 2. the contextual enhancement effect and some tests and extensions of the model. *Psychological Review*, 89:60–94.

Rumelhart, D. E., Hinton, G. E., & McClelland, J. L. (1986a). A general framework for parallel distributed processing. In (Rumelhart et al., 1986b), chapter 2, pages 45–76.

Rumelhart, D. E., McClelland, J. L., & PDP Research Group, editors (1986b). *Parallel Distributed Processing. Volume 1: Foundations.* MIT Press, Cambridge, MA.

Rumelhart, D. E. & Zipser, D. (1986). Feature discovery by competitive learning. In (Rumelhart et al., 1986b), chapter 5, pages 151–193.

Sandon, P. A. & Uhr, L. M. (1988). An adaptive model for viewpoint-invariant object recognition. In (CSS, 1988), pages 209–215.

Shepherd, G. M., editor (1990). *The Synaptic Organization of the Brain.* Oxford University Press, Oxford.

Stanton, P. K. & Sejnowski, T. J. (1989). Associative long-term depression in the hippocampus induced by hebbian covariance. *Nature*, 339:215–218.

Ungerleider, L. G. & Mishkin, M. (1982). Two cortical visual systems. In Ingle, D. J., Goodale, M. A., & Mansfield, R. J. W., editors, *The Analysis of Visual Behavior.* MIT Press, Cambridge, MA.

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

Zemel, R. S., Mozer, M. C., & Hinton, G. E. (1989). TRAFFIC: A model of object recognition based on transformations of feature instances. In Touretsky, D. S., Hinton, G. E., & Sejnowski, T. J., editors, *Proceedings of the 1988 Connectionist Models Summer School*, pages 452–461, San Mateo, CA. Morgan Kaufmann.