Neural Computation 11(2):347-374, 1999.

Implicit learning in 3D object recognition: The importance of temporal context

Suzanna Becker
Department of Psychology
McMaster University

November 25, 2001

Abstract

A novel architecture and set of learning rules for cortical self-organization is proposed. The model is based on the idea that multiple information channels can modulate one another's plasticity. Features learned from bottom-up information sources can thus be influenced by those learned from contextual pathways, and vice versa. A maximum likelihood cost function allows this scheme to be implemented in a biologically feasible, hierarchical neural circuit. In simulations of the model, we first demonstrate the utility of temporal context in modulating plasticity. The model learns a representation that categorizes people's faces according to identity, independent of viewpoint, by taking advantage of the temporal continuity in image sequences. In a second set of simulations, we add plasticity to the contextual stream and explore variations in the architecture. In this case, the model learns a two-tiered representation, starting with a coarse view-based clustering and proceeding to a finer clustering of more specific stimulus features. This model provides a tenable account of how people may perform 3D object recognition in a hierarchical, bottom-up fashion.

1 Introduction: context, coherence and plasticity

Context effects, both spatiotemporal and top-down, are ubiquitous in behavior and can also be observed at the neuronal level. The ability of context to influence perception has been demonstrated in many domains. For example, letters are recognized more quickly and accurately in the context of words (see e.g. McClelland & Rumelhart, 1981), while words are recognized more efficiently when preceded by related isolated words (see e.g. Neely, 1991), sentences or passages (Hess et al., 1995). In the compelling "McGurk effect" (McGurk & MacDonald, 1976; MacDonald & McGurk, 1978), a person is presented with a videotape of auditory information for one utterance simultaneously paired with visual information for another utterance. However, the mismatch typically goes unnoticed. What happens is that for some sound pairs, the person's percept tends to be dominated by the auditory cues, in other cases the visual cues dominate, and in still other cases, various fusions and/or alternations of the two sources are perceived. Apparently, when the two modalities provide contradictory information, people choose which modality to believe and which to ignore, or whether to fuse the modalities, according to the context.

Further, the importance of contextual information in modulating neuronal response profiles is becoming increasingly apparent. For example, some visual cortical cells (in the deepest layer of area V1) have been found that are excited by an oriented stimulus in the centre of their receptive field, and show an enhanced response to a similarly oriented stimulus in the surrounding region; on the other hand, the response is suppressed by an orthogonally oriented stimulus in the surround (Cudeiro & Sillito, 1996). In contrast, some cells show just the opposite pattern: they are antagonized by a similarly-oriented stimulus in the surround, and facilitated by an orthogonally-oriented stimulus (Sillito et al., 1995). On the other hand, about 40% of complex cells (in the superficial layers of area V1) are facilitated by the conjunction of a line segment in their classical receptive field and a colinear line segment placed nearby, outside their classical receptive field (Gilbert et al., 1996). Moreover, even in primary visual cortex, cells' tuning curves (in all cortical layers) are sensitive to the temporal history of the input signal and can show bimodal peaks and even complete reversals in tuning over time (Ringach et al., 1997). These examples demonstrate that neuronal responses can be modulated by secondary sources of information in complex ways.

Why would contextual modulation be such a pervasive phenomenon? One obvious reason is that if context can influence *processing*, it can help in disambiguating or cleaning up noisy stimuli. However, an over-reliance on contextual cues leaves the system open to the possibility of information loss, for example, by smearing information across discontinuities. A less obvious reason why context is so pervasive may be that if context can influence *learning*, it may lead to more compact and powerful representations, whereby units encode complex stimulus configurations.

In this paper, we focus particularly on *temporal context*. Most unsupervised classifiers

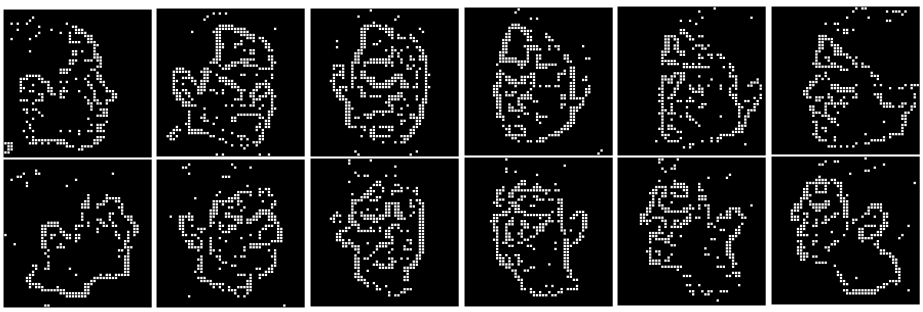

Figure 1: Two sequences of 48 by 48 pixel images digitized with an IndyCam and preprocessed with an edge filter using SGI's Image Works. Eleven views of each of four to ten faces were used in the simulations reported here. The alternate (odd) views of two of the faces are shown above.

are insensitive to temporal context; that is, they group patterns together solely on the basis of spatial overlap. This may be reasonable if there is very little shift or other form of distortion between one time step and the next, but is not a reasonable assumption about the sensory input to the cortex. Pre-cortical stages of sensory processing, certainly in the visual system and probably in other modalities, tend to remove low-order correlations in space and time (see, e.g., Dong and Atick's (1995) model of LGN cells). Consider the images in Figure 1. The top row shows a series of snapshots of one person's face being rotated through 180 degrees. The bottom row shows a series of snapshots of another person's face, also being rotated through 180 degrees. They have been preprocessed by a simple edge-filter, so that successive views of the same face have relatively little pixel overlap. Even in these low-resolution images, we can see certain regularities in the features of each individual. For example, each person's head shape remains consistent across changes in viewpoint. With respect to raw pixel overlap, however, two snapshots of a given individual's face taken from very different viewpoints often have less in common than snapshots of two different individuals' faces taken from the same viewpoint. This creates a difficult challenge for unsupervised learning systems. Unsupervised learning procedures like principal components analysis and clustering can only model lower-order structure (e.g. covariance or Euclidean proximity). How could a self-organizing system discover the higher-order structure shared by radically different views of the same object, and ignore the lower-order structure shared by identical views of different objects? Clearly, we have a long way to go in understanding what sort of learning procedures are employed by the brain, to form distributed representations and account for our high-level perceptual abilities.

One powerful cue for real vision systems is the temporal continuity of objects. Novel objects typically are encountered from a variety of angles, as the position and orientation of the observer, or objects, or both, vary smoothly over time. It would be very surprising if the

visual system did not capitalize on this temporal continuity in learning to group together visual events that co-occur in time. In the Discussion section, we mention several lines of empirical evidence in support of this notion. In the model of cortical self-organization proposed here, we postulate that *contextual modulation* plays a critical role in guiding unsupervised class formation. The term "context" is used very generally here to mean any secondary source of input; it could be from a different sensory modality, a different input channel within the same modality, a temporal history of the input, or top-down information from descending pathways. Although in the simulations reported here we specifically focus on temporal context in the visual system, the same ideas should be applicable to a variety of other sources of context (see Discussion) in a variety of cortical areas.

2 Maximum likelihood cost function

Given that we have identified context as an important cue in learning, the next step is to formalize this notion. We propose maximizing a log likelihood cost function, as in (Nowlan, 1990; Jacobs et al., 1991). In this framework, the network is viewed as a probabilistic, generative model of the data. The learning serves to adjust the weights so as to maximize the log likelihood of the model having generated the data:

$$L = \log P(data \mid model). \tag{1}$$

If the training patterns, $I^{(\alpha)}$, are independent,

$$\begin{aligned} L &= \log \prod_{\alpha=1}^{n} P(I^{(\alpha)} \mid model) \\ &= \sum_{\alpha=1}^{n} \log P(I^{(\alpha)} \mid model). \end{aligned} \tag{2}$$

However, this assumption of independence is not valid under natural viewing conditions. If one view of an object is encountered, a similar view of the same object is likely to be encountered next. In this paper, we propose an extension to the above model in which the independence assumption is relaxed, so that the inputs are only assumed to be independent given the context. In the most general case, the context could be any additional source of information. In the simulations reported here, we explore the special case where the temporal history of the input acts as the context.

There are several advantages to this approach. First, having a global cost function for the learning provides a principled basis for deriving learning rules in a network. Second, the maximum likelihood cost function sets up a very reasonable goal for the learning: modelling the probability distribution of the data. Third, by choosing an appropriate parametric form for the model, that is, the network architecture and associated statistical

assumptions, we can incorporate the added goal of allowing contextual input to modulate the learning.

2.1 Maximum Likelihood Competitive Learning (MLCL)

In Maximum Likelihood Competitive Learning (MLCL) (Nowlan, 1990), the units have Gaussian activations, y_i, and the network forms a mixture-of-Gaussians model of the data. The result is a simple and elegant network implementation of a widely used statistical clustering algorithm. A "soft competition" among the units, rather than a winner-take-all, "hard competition", determines the relative activation levels of the units and hence their learning rates for each pattern. This causes each unit to become selective for a different region of the input space.

The following cost function forms the basis for MLCL:

$$\begin{aligned} L &= \sum_{\alpha=1}^{n} \log \left[\sum_{i=1}^{m} P(I^{(\alpha)} \mid submodel_i)\ P(submodel_i) \right] \\ &= \sum_{\alpha=1}^{n} \log \left[\sum_{i=1}^{m} {y_i}^{(\alpha)}\ \pi_i \right] \end{aligned} \tag{3}$$

where the π_i's are positive mixing coefficients that sum to one, and the y_i's are the unit activations:

$${y_i}^{(\alpha)} = N(\vec{I}^{(\alpha)}, \vec{\mathbf{w}}_i, \Sigma_i) \tag{4}$$

where $N()$ is the Gaussian density function, with mean $\vec{\mathbf{w}}_i$ and covariance matrix Σ_i. Here and throughout the paper, we use the term "submodel" to refer to a Gaussian component in the mixture model. So y_i represents the probability of the input vector under the ith submodel, a Gaussian centred on the ith unit's weight vector, $\vec{\mathbf{w}}_i$. The ith mixing coefficient, π_i, represents the prior probability of the ith Gaussian having generated the data. In MLCL, the Gaussian means, $\vec{\mathbf{w}}_i$, are obtained by maximizing over L, and the mixing coefficients are either fixed to equal values or alternatingly re-estimated after each update of the model parameters as in the EM algorithm (Dempster et al., 1977). For simplicity, Nowlan typically used a single global variance parameter for all input dimensions, and allowed it to shrink during learning. L can be maximized by online gradient ascent[1] with learning rate ε:

$$\Delta w_{ij} = \varepsilon\ \frac{\partial L}{\partial w_{ij}} = \varepsilon \sum_{\alpha} \frac{\pi_i\ {y_i}^{(\alpha)}}{\sum_k \pi_k\ {y_k}^{(\alpha)}}\ \left({I_j}^{(\alpha)} - w_{ij} \right) \tag{5}$$

[1] Nowlan (1990) used a slightly different online weight update rule that more closely approximates the batch update rule of the EM algorithm.

The term $\frac{\pi_i {y_i}^{(\alpha)}}{\sum_k \pi_k {y_k}^{(\alpha)}}$ represents the ith submodel's probability given the current pattern and context. It is normalized over all competing units (submodels), hence the term "soft competition". A long-time average of this probability over many data items represents π_i, the overall probability of the ith submodel. Thus, this rule is quite biologically plausible. It consists of a Hebbian update rule with weight decay, using normalized post-synaptic unit activations.

2.2 Contextually modulated competitive learning (CMCL)

MLCL assumes the input patterns are independent. If we remove this restriction, allowing for temporal dependencies amongst the input patterns, the log likelihood function becomes:

$$\begin{aligned} L &= \log P(data \mid model) \\ &= \sum_\alpha \log P(I^{(\alpha)} \mid I^{(1)}, \ldots, I^{(\alpha-1)}, model) \end{aligned} \tag{6}$$

To incorporate a contextual information source into the learning equation, we extend MLCL by introducing a contextual input stream into the likelihood function:

$$\begin{aligned} L &= \log P(data \mid model, context) \\ &= \sum_\alpha \log P(I^{(\alpha)} \mid I^{(1)}, \ldots, I^{(\alpha-1)}, model, context) \end{aligned} \tag{7}$$

Unlike the model underlying standard MLCL, we want to deal with input streams that may contain arbitrarily complex temporal dependencies. Suppose the input and context represent two separate streams of observable data, with unknown interdependencies. This situation is depicted in Figure 2 a). Taken together, the input and context can be viewed as an ordered sequence of pairs, $(I^{(\alpha)}, C^{(\alpha)})$, where $C^{(\alpha)}$ is the contextual input pattern on training case α.

We now consider several simplifying assumptions that result in a tractable model. Our first assumption is that the model consists of a mixture of submodels. The log likelihood then becomes:

$$\begin{aligned} L = \sum_\alpha \log \Bigg[& \sum_j P(I^{(\alpha)} \mid I^{(1)}, \ldots, I^{(\alpha-1)}, C^{(1)}, \ldots, C^{(\alpha)}, submodel_j) \\ & P(submodel_j \mid I^{(1)}, \ldots, I^{(\alpha-1)}, C^{(1)}, \ldots, C^{(\alpha)}) \Bigg] \end{aligned} \tag{8}$$

Second, let us assume that the probability of observing a particular input pattern is independent of other patterns when conditioned on the context sequence, and vice versa. In other words, all of the temporal dependencies in the input stream can be accounted for

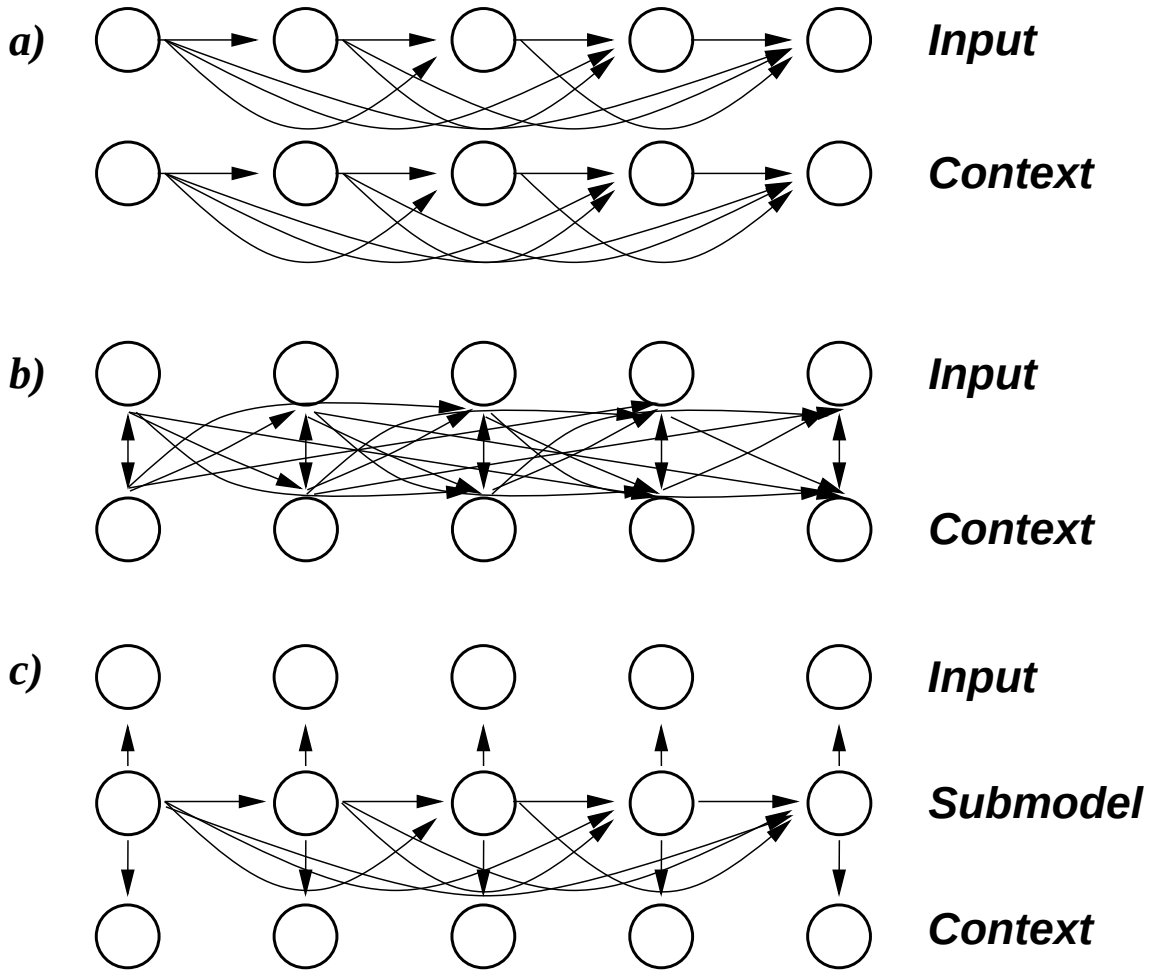

Figure 2: The conditional dependencies amongst the observable variables (context and input) are depicted in three situations. In a), the long-range dependences within the two sequences are shown. In b), the interdependencies within the two sequences disappear when each element in the top sequence is conditioned on the bottom sequence, and vice versa. In c), the sequences become independent of each other when conditioned on the hidden variables (the "submodel" indices).

by knowing the context, and vice versa. This situation is depicted in Figure 2 b). Now we have:

$$L = \sum_{\alpha} \log \left[\sum_{j} P(I^{(\alpha)} \mid C^{(1)}, \ldots, C^{(\alpha)}, submodel_j) \right. \\ \left. P(submodel_j \mid I^{(1)}, \ldots, I^{(\alpha-1)}, C^{(1)}, \ldots, C^{(\alpha)}) \right] \quad (9)$$

Finally, let us assume that given the submodel, the input and context are independent. In other words, all the remaining dependencies in the observable data are explained away by knowing which submodel generated the data at each point in time. This situation is depicted in Figure 2 c). Now the likelihood equation simplifies to:

$$L = \sum_{\alpha} \log \left[\sum_{j} P(I^{(\alpha)} \mid submodel_j)\ P(submodel_j \mid I^{(1)}, \ldots, I^{(\alpha-1)}, C^{(1)}, \ldots, C^{(\alpha)}) \right] \\ = \sum_{\alpha=1}^{n} \log \left[\sum_{j} {y_j}^{(\alpha)}\ {g_j}^{(\alpha)} \right] \quad (10)$$

where ${g_j}^{(\alpha)}$ represents the probability of the jth submodel given the input and context, and ${y_j}^{(\alpha)}$ represents the probability of the input under the jth submodel.

3 Network implementation

The CMCL cost function given in equation 10 could be implemented in a variety of architectures, depending upon how much computational power is allocated to individual units. In the Discussion section, we explore this issue further, and consider the potential advantage of more powerful units with nonlinear synaptic interactions. In the simulations reported here, we used multi-layer circuits consisting of an input layer, a layer of clustering units, and a layer of *gating units* as in Figure 3. We chose the term "gating units" because their role here is analogous to that of the gating network in the "competing experts" model (Jacobs et al., 1991). In fact, the model proposed here could be viewed as an unsupervised version of the mixture of competing experts architecture. Jacobs et al.'s competing experts network performs supervised learning, and can be interpreted as fitting a mixture of Gaussians model of the training signal. In contrast, here the clustering units (experts) are fitting a mixture model to the input signal, while the gating units simultaneously are adapting to the context signal, in order to help the clustering units divide up the input space. This is very different from a model that separately clusters the input and context signals because here, contextual features are used to modulate the partitioning of the input space. As our simulations show, this results in a very different clustering of the inputs.

The clustering units receive the primary source of input to the network. As in MLCL, each clustering unit produces an output ${y_i}^{(\alpha)}$ proportional to the probability of the input pattern, $I^{(\alpha)}$, given the ith submodel (this would be exactly equal to the probability if it were normalized). Each ${y_i}^{(\alpha)}$ is computed as a Gaussian function of its current input:

$$ {y_i}^{(\alpha)} = e^{-\|I^{(\alpha)} - \vec{w_i}\|^2 / {\sigma^2}_i} \tag{11} $$

where $\| \cdot \|$ is the L2 norm, $\vec{w_i}$ is the weight vector for the ith clustering unit representing the mean of the ith Gaussian, and ${\sigma^2}_i$ is the variance of that Gaussian, assuming all Gaussians are spherical. The gating units receive the contextual stream of input, and produce outputs ${g_i}^{(\alpha)}$ representing the probability of the ith submodel given the current context, $C^{(\alpha)}$. For the simulations reported here, the gating units compute their outputs according to a "softmax function" (Bridle, 1990) of their weighted summed inputs ${x_i}^{(\alpha)}$:

$$ {g_i}^{(\alpha)} = \frac{e^{{x_i}^{(\alpha)}}}{\sum_j e^{{x_j}^{(\alpha)}}} \tag{12} $$

$$ {x_i}^{(\alpha)} = \sum_k {C_k}^{(\alpha)} \, v_{ik} \tag{13} $$

where j indexes over all gating units in the network, and v_{ik} is the weight on the connection from the kth contextual input to the ith gating unit. Here, we have made a further simplifying assumption that the prior probabilities of the submodels (the $p(submodel_i)$ terms in equation 10) are all equal and fixed, and can therefore be folded into the gating units' activations g_i. Alternatively, assuming the probabilities of choosing each submodel form a Markov chain, that is, they depend only on knowledge one step back in time, one could then estimate the true probabilities of submodels under a Hidden Markov Model (HMM) (as suggested by Hinton, personal communication). This would allow for temporal dependencies between the submodels over time to be modelled explicitly. Cacciatore and Nowlan (1994) have extended the mixture of competing experts model in this way, to allow recurrent gating networks. See the Discussion for further comments on the relation between HMMs and our model.

4 The learning equations

Given the likelihood function defined by equation 10, online learning rules for the clustering and gating units can be derived by differentiating L with respect to their weights. The variances of each of the Gaussians, σ_i^2, could be approximated by their maximum likelihood estimates under a mixture model, as in the EM algorithm (Dempster et al., 1977). Instead, we used a simple online approximation to the true variance of the input vector about each

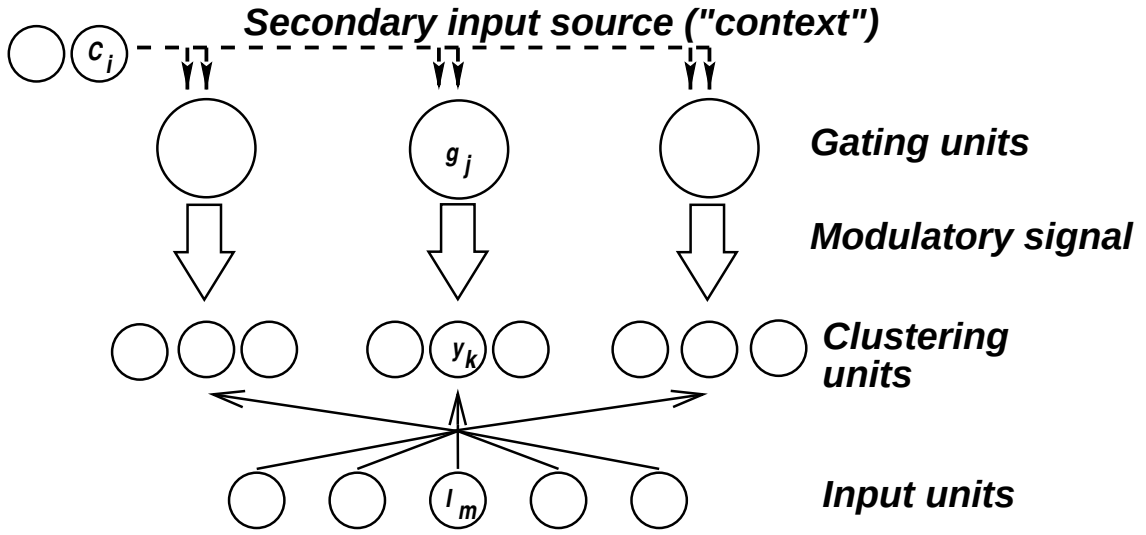

Figure 3: A neural circuit for implementing CMCL.

clustering unit's weight vector:

$$\sigma_i{}^{2(\alpha)} \;=\; k \sum_j \left(w_{ij}^2 \;+\; (I^{(\alpha)}{}_j)^2 \right) \tag{14}$$

where k is a constant. This approximation would be exact, to within a constant factor, if the input vectors were of fixed length and uncorrelated with the weight vectors. In the first set of simulations reported here, $k = 0.05$, and in the second set, $k = 0.03$. The main role of the adaptive variance in the learning is to scale the clustering unit activations, to prevent them from overfitting the training patterns.

The learning rule for the weight from the jth input to the ith clustering unit for input pattern α is:

$$\begin{aligned} \Delta w_{ij} \;&=\; \varepsilon \; \frac{\partial L}{\partial y_i{}^{(\alpha)}} \, \frac{\partial y_i{}^{(\alpha)}}{\partial w_{ij}} \\ &=\; \varepsilon \frac{g_i{}^{(\alpha)} \; y_i{}^{(\alpha)}}{\sum_k g_k{}^{(\alpha)} \; y_k{}^{(\alpha)}} \; \frac{1}{\sigma_i{}^{2(\alpha)}} \; \left(I_j{}^{(\alpha)} - w_{ij} \;+\; w_{ij} \; \frac{\| \; \mathbf{I}^{(\alpha)} - \mathbf{w}_i \; \|^2}{\sum_k (I_k{}^{(\alpha)})^2 + w_{ik}{}^2} \right) \end{aligned} \tag{15}$$

where ε is a learning rate constant.

The learning rule for the weight from the jth contextual input to the ith gating unit for input pattern α is:

$$\Delta v_{ij} \;=\; \varepsilon \; \frac{\partial L}{\partial g_i{}^{(\alpha)}} \, \frac{\partial g_i{}^{(\alpha)}}{\partial v_{ij}}$$

$$= \varepsilon \left(\frac{{g_i}^{(\alpha)} \, {y_i}^{(\alpha)}}{\sum_k {g_k}^{(\alpha)} \, {y_k}^{(\alpha)}} - {g_i}^{(\alpha)} \right) \, {I_j}^{(\alpha)} \tag{16}$$

As a consequence of the multiplicative interaction between the gating and clustering units' activations in the cost function (Equation 10), each gating unit's activation modulates the corresponding clustering unit's learning. Thus, the clustering units are encouraged to discover features that agree with the current contextual gating signal (and vice versa). At any given moment in time, if their contextual gating signal is weak, or if they fail to capture enough activation from their bottom-up input, they will do very little learning. Only when a clustering unit's weight vector is sufficiently close to the current input vector *and* its corresponding gating unit is strongly active will it do substantial learning.

5 Simulations with network 1

Our first set of simulations was designed to demonstrate the utility of temporal context in contributing to higher-order feature extraction and viewpoint-invariant object recognition. For these simulations, the gating connection weights were held fixed. Our second set of simulations was designed to generalize these findings to a network with adaptive links in the gating layer, and to show that by varying the architectural constraints, the network could develop pose-tuned rather than viewpoint-invariant face-tuned units.

For our first set of simulations, we used networks of the form shown in Figure 4. The network is subdivided into *modules*. Here, each module consists of one or more clustering units and one gating unit. In our second set of simulations, modules contain multiple gating units and only one clustering unit. The contextual inputs are time-delayed, temporally blurred versions of the outputs of a module (including both gating and clustering units' outputs). The gating units' outputs are softmax functions of their weighted summed blurred inputs. The temporal blurring on the contextual input lines was achieved by accumulating the activation on each connection as follows:

$${C_i}^{(\alpha)} = 0.5({C_i}^{(\alpha-1)} + {input_i}^{(\alpha-1)}) \tag{17}$$

where ${input_i}^{(\alpha)}$ is the ith input to the gating unit before blurring for pattern α; this input could be equal to the output of either a clustering unit in the layer below or the output of the gating unit itself (see Figure 4). However, more general forms of context are possible, as mentioned in the Discussion section. We have deviated from the general form of the architecture shown in Figure 3 in an important way: There is now a many:one mapping from clustering units to gating units, so that clustering units within the same module i receive a *shared gating signal*, g_i, and produce outputs y_{ij}. Thus, clustering units in the same module are responsible for learning different submodels, but they predict the same

contextual feature. The likelihood equation now becomes:

$$L = \sum_{\alpha=1}^{n} \log \left[\sum_{i=1}^{m} g_i^{(\alpha)} \frac{1}{l} \sum_{j=1}^{l} {y_{ij}}^{(\alpha)} \right] \quad (18)$$

To relate this to the original mixture model given by equation 10, we still have a single mixture of Gaussian submodels, with each clustering unit corresponding to a submodel. However, the probabilities over submodels (the g_is) given the inputs and contexts have some equality constraints imposed, so that clustering units in the same module share the same submodel probability.

One might predict that clustering units with a shared source of contextual input would all come to detect exactly the same feature. Fortunately, there is a disincentive for them to do so: They would then do poorly at modelling the input. Thus, clustering units in the same module should come to encode a common part of the context but detect different features.

Our network architecture was designed with several goals in mind. First, the modular, layered architecture is meant to constrain the network to develop hierarchical representations and functional modularity, as observed in the cortical laminae and columns respectively (see e.g. (Calvin, 1995)). That is, we should see a progression from simple to higher-order features in the clustering and gating layers, with functional groupings of similar features in units within the same module. Second, we expect the temporal context to influence the sort of features learned by the clustering layer; each clustering unit should detect a different range of temporally correlated features.

To test the predictions of our model, we performed simulations with networks like the one shown in Figure 4 trained on sequences of patterns like the ones shown in Figure 1. The training patterns consisted of a set of image sequences of ten centered, gradually rotating faces. In our first set of simulations, there were four modules and only four of the ten faces were used; in the final simulations, the generality of our findings was extended by training a larger network of ten modules like the ones shown in Figure 4 on all ten faces. In both cases, there were three clustering units per module. It was predicted that the clustering units should discover "features" such as temporally correlated views of specific faces. Further, different views of the same face should be represented by different clustering units within the same module because they will be observed in the same temporal context, while the gating units should respond to particular individual's faces, independent of viewpoint.

The training and testing pattern sets were created by repeatedly visiting each of the ten faces in random order. For each face, an ordered sequence of views was presented to the network, by randomly choosing either a left-facing or right-facing view as the initial view in the sequence, and then presenting the remaining views of that face in an ordered sequence. For a given face sequence, views were presented in an ascending order and then a descending order (e.g. rotating through 180 degrees to the right and then to the left), so the

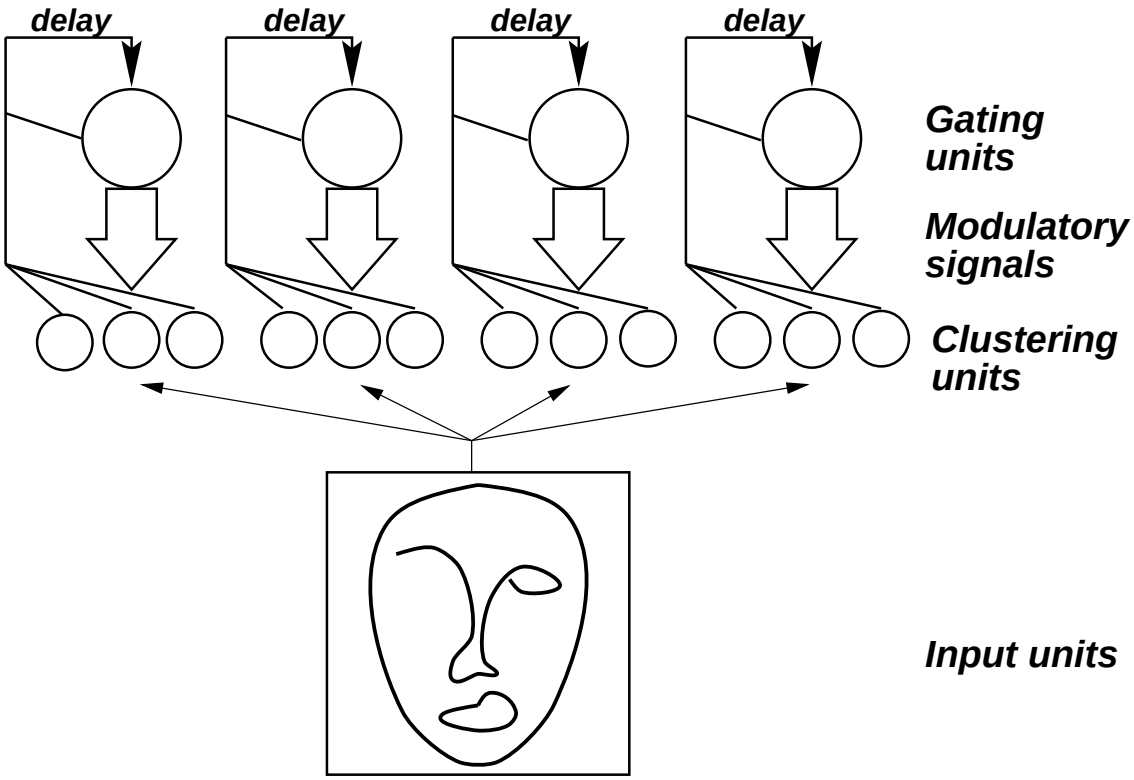

Figure 4: The architecture used in the first set of simulations reported here. The gating units received all their inputs across unit delay lines with fixed weights of 1.0. For these simulations, some of the networks had an architecture with four modules exactly as shown here, and were trained on sequences of images of four individuals' faces. For the remaining simulations, the networks had ten modules like the ones shown above, and were trained on sequences of ten individuals' faces.

initial view was always the final view in each sequence. At the end of each face sequence, a new face and starting view were randomly selected. The network had no knowledge of when a new face would occur or that the training set actually contained ordered sequences. Thus, although the network assumes that temporal context is smooth everywhere, in these data, it is actually discontinuous across the boundaries between sequences.

Gating units had self-links, as well as links from all the clustering units within the same module, all of which had unit time delays. All the gating unit connections had fixed weights of 1.0. Thus, each gating unit received a temporal history of its own output and of the outputs of the clustering units in the same module.

Tuning curves for all units in the network in a typical run are plotted in Figures 5 and 6. The clustering units became specialized for detecting particular faces in a narrow range of views, as shown in Figure 5. Simply by accumulating a temporal history of the clustering units' activations within a module, each gating unit was then able to respond to an individual face, independent of viewpoint, as shown in Figure 6. Of course, the tuning curves for the gating layer shown here depend upon there being continuity in the context signal both during training and testing.

One might wonder how much of the network's ability to discriminate faces was due to the temporal context, and how much was simply due to unsupervised clustering, independent of the contextual modulation. To answer this question, the baseline effect of the temporal context on clustering performance was assessed by comparing the network shown in Figure 4 to the same network with all connections into the gating layer removed. The latter is equivalent to MLCL with fixed, equal mixing proportions (π_i's). First, networks with four modules were trained on sequences of four faces. To quantify clustering performance, each unit was assigned to predict the face class for which it most frequently won (was the most active). Then for each pattern, the layer's activity vector was counted as correct if the winner correctly predicted the face identity. Generalization performance was assessed by training the network only on the odd-numbered views, and testing classification performance on the even-numbered views.

The results are summarized in Table 1. As one would expect, the temporal context provides incentive for the clustering units to group successive instances of the same face together, and the gating layer can therefore do very well at classifying the faces with a much smaller number of units - i.e., independently of viewpoint. In contrast, the clustering units without the contextual signal are more likely to group together instances of different people's faces.

Next, a network like the one shown in Figure 4 but with 10 modules was presented with a set of 10 faces, 11 views each. As before, the odd-numbered views were used for training and the even-numbered views for testing. Without the influence of the context layer, the network's classification performance was very poor. With the addition of contextual modulation, this network still had difficulty classifying all ten faces correctly, and seemed to be somewhat more sensitive to the weights on the gating connections. However,

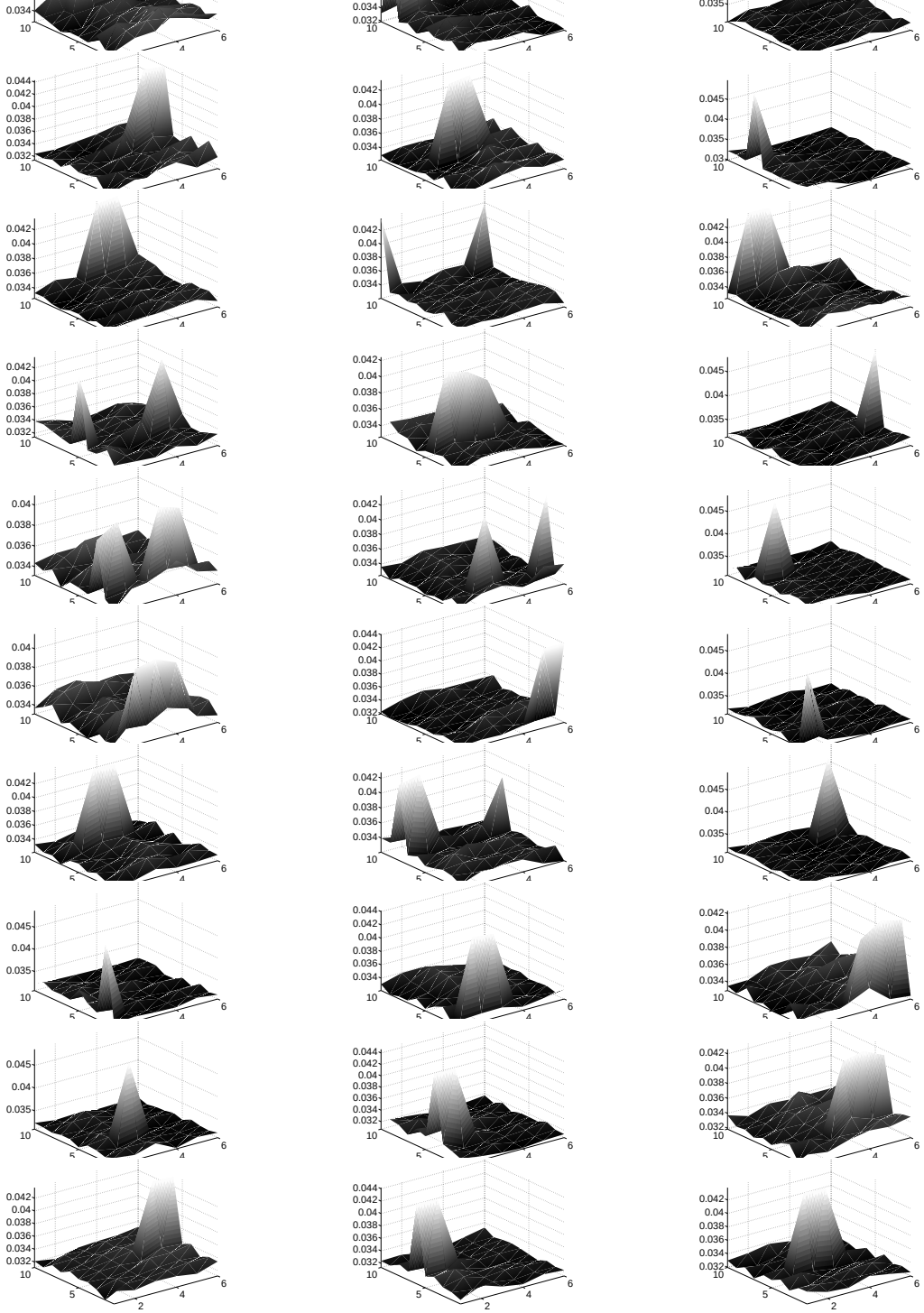

Figure 5: Thirty clustering units' normalized activations are plotted against face identity (bottom left axis) and viewing angle (bottom right axis) of patterns. Each graph shows the activations of a single unit over the entire set of training patterns. Units in the same row were trained with a common contextual gating signal (see Figure 4), and have learned to respond to different views of the same face.

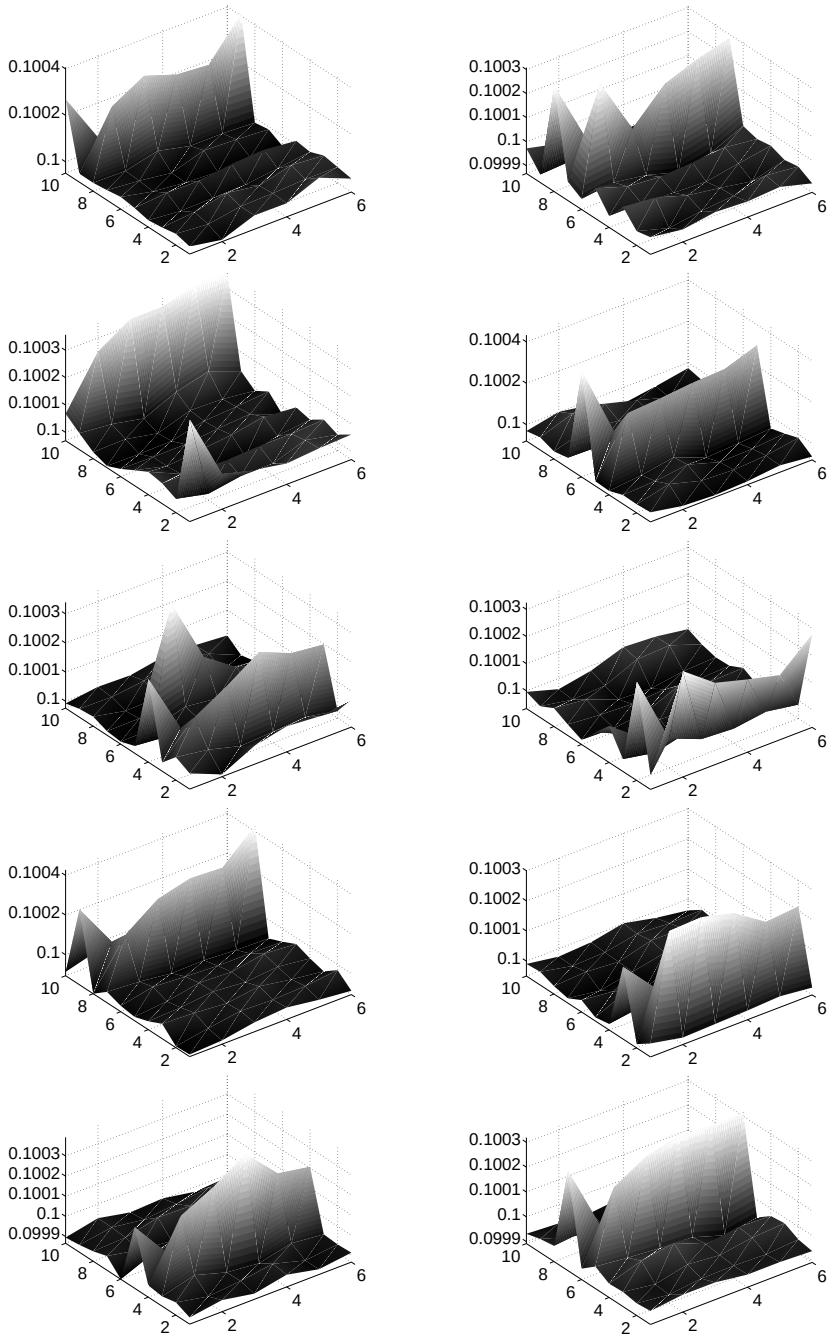

Figure 6: Ten gating units' activations are plotted against face identity (bottom left axis) and viewing angle (bottom right axis) of the training patterns. Each graph shows the activations of a single unit over the entire set of training patterns. Each gating unit provided a contextual gating signal to three clustering units (see Figure 4), and learned to respond to a single face, independent of view.

		Train	Test
no context, 4 faces:	Layer 1	59.2 (2.4)	65.0 (3.5)
no context, 10 faces:	Layer 1	15.0 (0.0)	12.0 (0.0)
context, 4 faces:	Layer 1	88.4 (3.9)	74.5 (4.2)
	Layer 2	88.8 (4.0)	72.7 (4.8)
context, 10 faces:	Layer 1	96.3 (1.2)	71.0 (3.0)
	Layer 2	91.8 (2.4)	70.2 (4.3)

Table 1: Mean percent (and standard error) correctly classified faces, across 10 runs, for unsupervised clustering networks trained for 2000 iterations with a learning rate of 0.5, with and without temporal context. Layer 1: clustering units. Layer 2: gating units.

when the weights on the self-pointing connections on the gating units were increased from 1.0 to 3.0, to increase the time constant of temporal averaging, the network performed extremely well. On average, the top layer units achieved 96% correct classification on the training set and 70% correct on the test set. In further simulations, reported in Becker (1997), the generalization performance of the above unsupervised network was shown to be substantially superior to that of supervised back-propagation networks with similar architectures; however, when a temporal smoothness constraint was imposed on the hidden layer units' states, even feed-forward back-propagation networks performed as well as our unsupervised model.

6 Simulations with network 2

The network shown in Figure 4 learned a "grandmother cell" representation, where each clustering unit learned to specialize for a single face at a particular viewpoint, and each gating unit therefore responded to a single face over a wide range of viewpoints. Although "face cells" have indeed been identified now by many labs, e.g. (Gross et al., 1971; Perrett et al., 1982; Desimone et al., 1984; Yamane et al., 1988; Tanaka et al., 1991) these cells only rarely exhibit either viewpoint invariance or selectivity for a single individual; the vast majority of face cells are tuned to one of only four views (front, back, left and right) and respond roughly equally to the heads of different individuals (Perrett et al., 1992).

There are several reasons why it is unlikely that the brain uses a grandmother cell representation as a matter of course. For one, it is very expensive with respect to neural machinery. Further, it does not scale well; each time a new face is encountered, new representational units would need to be added. Finally, this type of representation exhibits poor generalization.

In our second set of simulations, we sought to explore the interaction between the architecture and the cost function in constraining the representation learned by the network. This time, we used the architecture shown in Figure 7. This network differs from the one used in the fist set of simulations in two important ways, chosen to encourage more distributed representations of faces. First, the network has fewer modules than the previous one: only three modules were trained to encode all ten faces. Now, the network must form a more compact encoding of the face stimuli. Second, there is now only one clustering unit per module, and there are multiple gating units per module (four per module in the simulations reported here). Thus, rather than a many-to-one relationship between clustering and gating units in each model, the relationship is one-to-many. The clustering units should therefore be encouraged to develop broader tuning curves, and might be expected to cluster faces based on viewpoint (pose) rather than face identity, given the low pixel overlap between successive views of the same face. Further, because there are multiple gating units for each clustering unit, the gating units might be expected to learn a more distributed representation of faces.

To accommodate the one-to-many relationship between the clustering and gating units, the cost function was modified so that each clustering unit takes as its gating signal the average of the activations over the gating units in the same module:

$$L = \sum_{\alpha=1}^{n} \log \left[\sum_{i=1}^{m} y_i^{(\alpha)} \frac{1}{l} \sum_{j=1}^{l} {g_{ij}}^{(\alpha)} \right] \tag{19}$$

As in the first network, we still have a single mixture of Gaussian submodels, with each clustering unit corresponding to a submodel. Now, the probability over each submodel, i, given the inputs and contexts, is computed by averaging the activations g_{ij} of gating units within the same module. As before, the gating units received time-delayed, temporally blurred inputs from the clustering layer. Unlike in the previous simulations, the gating units also received time-delayed, temporally blurred inputs directly from the input layer. This extra source of context was provided so that gating units in the same module would have some basis for developing differential responses.

The clustering units' connection weights were updated for 2000 iterations with a fixed learning rate of 0.1 while the gating units' connection weights were initially held fixed. Typical response profiles for the clustering units are shown in Figure 8. As predicted, these units exhibited broad face-tuning but relatively narrow pose-tuning.

The gating units' connection weights from the input layer were then updated for 2000 further iterations with a fixed learning rate of 0.02. No constraints were placed on these weights, so they could potentially grow larger than the weights from the clustering to the gating layer. Networks with different numbers of gating units per module (but always three or four modules) were experimented with, and produced qualitatively similar results. The gating units tended to respond to combinations of one or more faces at similar poses.

However, the responses were not convincingly distributed. Rather, different gating units became selective for narrow, relatively non-overlapping regions of the face-pose-space. To further encourage the gating units to develop more distributed responses, the time-delay and blurring from the direct input connections to the gating layer was removed. Thus, like the clustering units, the gating units could now access only a single time slice of the input at a given moment. As predicted, this decreased the tendency for gating units to group faces of particular individuals over time, resulting in more multi-modal response profiles as in Figure 9. In this case, gating units in the same module (plotted in the same row in Figure 9) tended to have similar pose-tuning, and multi-modal, somewhat overlapping face-tuning profiles. This architecture actually violates the conditional independence assumption about the input and context streams, by using the same signal for both input and context. This would be of greater concern if the clustering and gating layers were adapted simultaneously, in which case they could achieve agreement in trivial ways, e.g. by only attending to small subsets of their inputs. To address this issue of independence, similar results were obtained in networks in which the clustering and gating layers were randomly connected to the input layer, which provided an approximation to independence.[2]

To summarize our second set of simulations, we sought to extend our basic findings by exploring several variations in the architecture which were predicted to lead to more distributed representations of faces. In particular, fewer modules were used, and there were multiple gating units per module. As predicted, the clustering units became less tuned to individuals' faces. Instead, they developed pose-tuning and were broadly selective to a wide range of individuals. It was also predicted that the gating units would form distributed codes for faces. However, although their tuning curves were multi-modal in face-pose-space, they were not strongly overlapping, but instead, remained relatively local. This representation would be good for recognizing general features common to many faces, but would not be as appropriate for face classification as compared to that learned by the first architecture.

7 Discussion

The simulation results with our model demonstrate that temporal context can markedly alter the sort of features or classes learned by an unsupervised network. When combined with appropriate architectural constraints, a range of representations can be learned. But does this have anything to say about self-organization in the cortex? In this section, we consider behavioral and physiological lines of evidence in support of our model. Finally, several related computational models are considered.

[2]This approximation is still not exact. A better solution would be to connect the clustering and gating layers to physically different parts of the input. For example, the gating units could be connected to the spatial context surrounding the input to the clustering unit(s) in the same module.

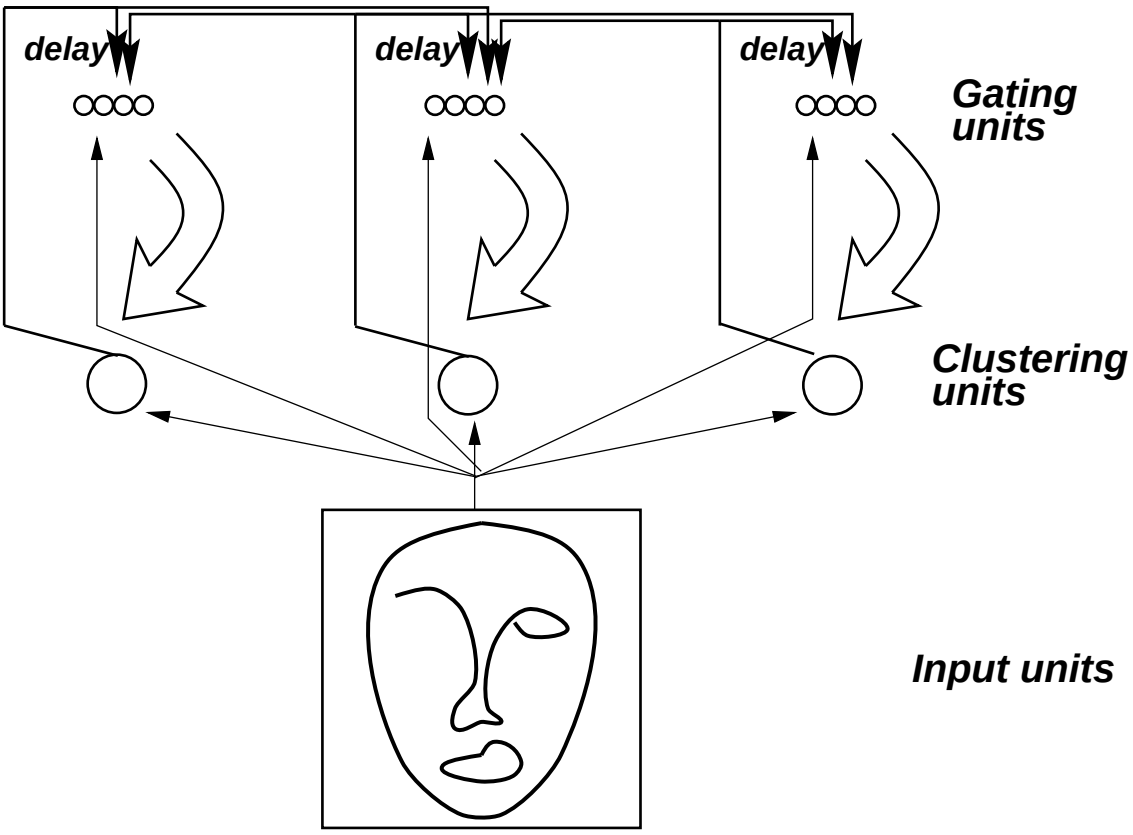

Figure 7: The architecture used in the second set of simulations reported here. The gating units received normalized, temporally blurred input from clustering units in the same module and neighboring module(s), and direct connections from the input layer. The connections from the clustering units to the gating units had fixed weights of 0.6 for within-module connections, 0.2 for between-module connections to the middle module, and 0.4 for between-module connections to the end modules. The weights on the direct input connections to the gating layer were fixed at zero while the clustering layer was trained, and were subsequently adapted during a second training phase.

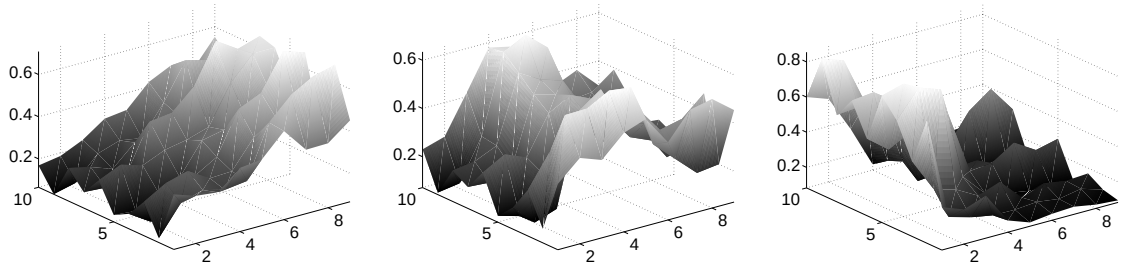

Figure 8: Three clustering units' normalized activations are plotted against face identity (bottom left axis) and viewing angle (bottom right axis) of patterns. Each graph shows the activations of a single unit over the entire set of training patterns. Each clustering unit received contextual input from three gating units (see Figure 5), and learned to respond to faces from a particular viewpoint, independent of face identity.

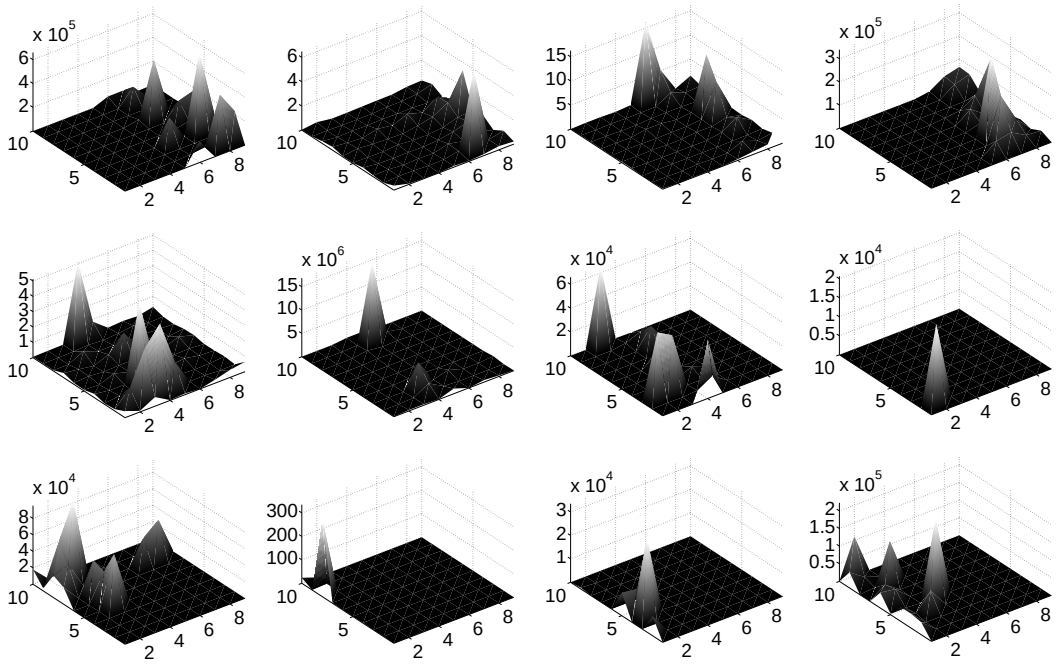

Figure 9: Twelve gating units' activations, before normalization, are plotted against face identity (bottom left axis) and viewing angle (bottom right axis) of patterns. Each graph shows the activations of a single unit over the entire set of training patterns. Units in the same row were trained to provide a common contextual gating signal to a single clustering unit (see Figure 5). For the most part, each has learned to respond to multiple faces from a narrow range of views.

7.1 Empirical evidence for the use of temporal context

As mentioned in the introduction, there is evidence that single cells' tuning curves exhibit complex temporal dynamics (Ringach et al., 1997; De Angelis et al., 1995). But are these effects hard-wired, or might temporal context play a role in the *learning* of receptive fields? Physiological evidence from Miyashita (1988) would support the latter contention. Miyashita repeatedly exposed monkeys to a fixed sequence of 97 randomly generated fractal images during a visual memory task, and subsequently recorded from cells in the anterior ventral temporal cortex. Many cells responded to several of the fractal patterns, and the grouping of patterns was based on temporal contiguity rather than geometric similarity. This is rather striking evidence for learning based on temporal associations rather than pattern overlap.

Furthermore, recent behavioral evidence suggests that temporal context is important to human learning about novel objects. Seergobin, Joordens and Becker (unpublished data) exposed experimental participants to sequences of images of faces of the same sort used in the simulations reported here. In one condition, faces were viewed "coherently", that is, in ordered sequences from left to right or right to left. In another condition, faces were viewed "incoherently", that is, each face was presented in a scrambled sequence with the views randomly ordered. Participants demonstrated a significant benefit in face-matching from the more coherent temporal context during study.[3] Given that there may be differences in the way humans process faces as compared to other types of objects (Bruce, 1997), Seergobin et al. extended their results in a further set of experiments using static image sequences of novel, artificially generated bumpy objects resembling asteroids. In this case, a similar advantage for coherent temporal context in implicit learning was shown.

7.2 Justification for a modular, hierarchical architecture

The hierarchical, modular architecture shown in Figure 3 is motivated by several features widely considered to be ubiquitous throughout all regions of the neocortex: a laminar structure (see e.g. Douglas & Martin, 1990), and a functional organization into "cortical clusters". As Calvin (1995, pp. 269) succinctly puts it, "... the bottom layers are like a subcortical 'out' box, the middle layer like an 'in' box, and the superficial layers somewhat like an 'interoffice' box connecting the columns and different cortical areas". We tentatively suggest a correspondence between the clustering units in our model and layer IV cells, and between the gating units and the deep and superficial layer cells. With respect to

[3]One might then wonder whether fully animated video sequences would confer a further benefit on object learning, over and above that of temporally coherent sequences of static images. Interestingly, for the case of animated versus statically studied faces, Bruce and colleagues found no such advantage in two different experiments (Christie & Bruce, 1998; Bruce & Valentine, 1998). Note, however, that dynamic viewing at the time of *testing* does improve face recognition performance (Christie & Bruce, 1998; Bruce & Valentine, 1998).

functional modularity, in many regions of cortex, spatially nearby columns tend to cluster into functional groupings with similar receptive field properties (see e.g. Calvin, 1995), including visual area V2 (Levitt et al., 1994), and inferotemporal cortex (Tanaka et al., 1993). We experimented with two different means of inducing functional modularity in our model: In the first set of simulations, subsets of clustering units shared a common gating unit, and learned to predict similar regions of the contextual space. Consequently, they became tuned to temporally coherent features: different views of the same individual's face. In the second set of simulations, subsets of gating units shared a common clustering unit, and learned to detect different contextual features that predicted a common region of the input space. In this case, different gating units in the same module became specialized for similar views but different faces. Further, clustering units in nearby modules had partially overlapping contextual inputs; this resulted in a similarity of function across neighboring modules: clustering units in adjacent modules were selective for similar views. It remains to be seen which, if either, of these architectures is a good model of cortical self-organization and modularity.

Another possibility is that the functionality of an entire module of clustering and gating units in our model could be computed by a single neuron. The neuron would then require nonlinear interactions among synaptic inputs, so that the context could act in a modulatory fashion, rather than as a primary driving stimulus. A number of models of cortical cell responses have proposed multiplicative interactions between modulatory and primary input sources (Nowlan & Sejnowski, 1993; Mel, 1994; Mundel et al., 1997; Pouget & Sejnowski, 1997).

7.3 Face processing and shape recognition in the cortex

The model in its present implementation is not meant to be a complete account of the way humans learn to recognize faces. Viewpoint-invariant recognition is probably achieved, if at all, in a hierarchical, multi-stage system. In ongoing work, we are exploring this possibility by training, in series, a sequence of networks like the one shown in Figure 3, with progressively larger receptive fields at each stage.

Oram and Perrett (1994) have proposed a roughly hierarchical, multi-stage scheme for decomposing the ventral visual pathway into a functional processing hierarchy. Of particular relevance to the results reported here is their proposal for the organization of object recognition in the infero-temporal cortex (IT). A large body of physiological evidence supports the notion that IT cells are responsible for complex shape coding. After Tanaka and colleagues (Tanaka et al., 1991), Oram and Perrett propose that object recognition is accomplished in a distributed network in IT (particularly area AIT) as follows: each module or column codes for a particular shape class. A given object activates many modules, corresponding to different complex visual features. Within a module, different cells exhibit slightly different selectivities, and can thereby signal more precisely the stimulus features.

For example, cells in a given column might all code for a pair of small round objects aligned horizontally. Within a column, different cells might further specialize for a pair of eyes or a pair of headlights. Responses across many such columns, taken together, could thereby code a great many different objects uniquely. Only under special circumstances would a grandmother cell be devoted to recognizing a unique conjunction of stimulus features.

The network shown in Figure 7 learned a representation that is consistent, at least in broad terms, with the scheme for representing objects proposed by Tanaka et al, and Oram and Perrett. Units in the same module learned to code for a particular class of stimuli - faces over some wide range of views. Different gating units in the same module became further specialized to detect particular features of different faces. These units were usually not tuned to one specific face, but each tended to respond to several specific individuals' faces. A question for future research is whether the model presented here could encode different uncorrelated features, or different classes of objects, across many different modules.

7.4 Related work

Phillips, Kay and Smyth (Phillips et al., 1995; Kay & Phillips, 1997) have proposed a model of cortical self-organization they call *coherent Infomax* that incorporates contextual modulation. In their model, the outputs from one processing stream modulate the activity in another stream, while the mutual information between the two streams is maximized. They view this algorithm as a compromise between Imax (Becker & Hinton, 1992) and Infomax (Linsker, 1988). A number of other unsupervised learning rules have been proposed based on the assumption of temporally coherent inputs. Becker (1993) and Stone (1996) proposed learning algorithms that maximize the mutual information in a neuron's output at nearby points in time. Földiák (1991) and Weinshall, Edelman and Bülthoff (Weinshall et al., 1990; Edelman & Weinshall, 1991) proposed variants of competitive learning that used blurred outputs and time delays, respectively, to associate items over time. Several investigators (Seergobin, 1996; Wallis & Rolls, 1997; Stewart Bartlett & Sejnowski, 1998) have shown that Földiák's model, when applied to faces, develops units with broad pose-tuning. Temporal smoothing has also been shown to broaden pose-tuning to faces in feed-forward back-propagation networks (Becker, 1997) and in Hopfield-style attractor networks (Stewart Bartlett & Sejnowski, 1997). O'Reilly and Johnson (1994) used feedback inhibition and excitation to achieve temporal smoothing and pose-invariance in a multi-layer model that is perhaps most similar to the one proposed here. Their network used excitatory feedback from the top-layer units to pools of middle-layer units, so that position-invariance was achieved to progressively greater degrees in higher layers. O'Reilly and Johnson's model could be viewed as a more biologically constrained approximation to the more formal learning model proposed here.

As mentioned earlier, Hidden Markov Models provide another way to implement the

model proposed here (Geoff Hinton, personal communication). However, current techniques for fitting HMMs are intractable if state dependencies span arbitrarily long time intervals. Saul and Jordan (1996) have proposed an elegant generalization of HMMs they call Boltzmann chains, for modelling discrete time series. In one special case, they show that the learning is tractable for *coupled parallel chains*, that is, parallel discrete time series of correlated features, coupled by common hidden variables. This case would correspond exactly to the one assumed here (see Figure 2 c), if the temporal dependencies were restricted to adjacent points in time.

One limitation of the model proposed here is that it does not provide a complete account of the role of feedback between cortical layers. Although top-down feedback could be viewed as just another source of context, and thereby incorporated into the present model, the solution might not be globally optimal in a multi-stage system. The work of Hinton and colleagues on the Helmholtz machine (Hinton & Zemel, 1994; Dayan et al., 1995) and Rao and Ballard's Extended Kalman Filter model (Rao & Ballard, 1997) provide two different solutions to this problem.

8 Conclusions

A "contextual input" stream was implemented in the simplest possible way in the simulations reported here, using fixed delay lines and recurrent feedback. However, the model we have proposed provides for a very general way of incorporating arbitrary contextual information, and could equally well integrate other sources of input. A wide range of perceptual and cognitive abilities could be modelled by a network that can learn features of its primary input in particular contexts. These include multi-sensor fusion, feature segregation in object recognition using top-down cues, and semantic disambiguation in natural language understanding. Finally, our model may be able to account for the interaction between multiple memory systems in the brain. For example, it is widely believed that memories are stored rapidly in the hippocampus and related brain structures, and more gradually stored in the parahippocampal and neocortical areas (McClelland et al., 1995). The manner in which information is represented in the hippocampal system is undoubtedly very different from that of the cortex. A major question is how the two systems interact. The model proposed here may be able to explain how interactions between disparate information sources such as the hippocampal and cortical codes are integrated into a unified representation in the cortex. The output of the hippocampus, a rapidly formed novel code, could be treated simply as another source of context, to be integrated with bottom-up information received by various cortical areas.

Acknowledgements

The ideas in this paper arose in the context of many discussions with Ron Racine and Larry Roberts about cortical circuitry and plasticity. Thanks to Geoff Hinton for contributing several valuable insights about the model, and to Gary Cottrell, Peter Dayan, Darragh Smyth and four anonymous reviewers for invaluable comments on earlier drafts of this paper. The face images were collected by Ken Seergobin. All simulations were carried out using the Xerion neural network simulator developed in Hinton's lab, and additional software written by Lianxiang Wang. Financial support for this work was provided by the McDonnell-Pew Program in Cognitive Neuroscience and the Natural Sciences and Engineering Research Council of Canada.

References

Becker, S. (1993). Learning to categorize objects using temporal coherence. In S. J. Hanson, J. D. Cowan, & C. L. Giles (Eds.), *Advances in Neural Information Processing Systems 5* (pp. 361–368). San Mateo, CA: Morgan Kaufmann.

Becker, S. (1997). Learning temporally persistent hierarchical representations. In M. Mozer, M. Jordan, & T. Petsche (Eds.), *Advances in Neural Information Processing Systems 9* (pp. 824–830). MIT Press.

Becker, S. & Hinton, G. E. (1992). A self-organizing neural network that discovers surfaces in random-dot stereograms. *Nature*, *355*:161–163.

Bridle, J. S. (1990). Training stochastic model recognition algorithms as networks can lead to maximum mutual information estimation of parameters. In D. S. Touretzky (Ed.), *Neural Information Processing Systems, Vol. 2* (pp. 111–217). San Mateo, CA: Morgan Kaufmann.

Bruce, V. (1997). Human face perception and identification. In *NATO ASI on Face Recognition.*

Bruce, V. & Valentine, T. (1998). When a nod's as good as a wink. the role of dynamic information in facial recognition. In M. Gruneberg, P. Morris, & R. Sykes (Eds.), *Practical aspects of memory: Current research and issues (Volume 1)* (pp. 169–174). Wiley.

Cacciatore, T. W. & Nowlan, S. J. (1994). Mixtures of controllers for jump linear and non-linear plants. In J. D. Cowan, G. Tesauro, & J. Alspector (Eds.), *Advances in Neural Information Processing Systems 6* (pp. 719–726). San Mateo, CA: Morgan Kaufmann.

Calvin, W. H. (1995). Cortical columns, modules, and Hebbian cell assemblies. In M. Arbib (Ed.), *The handbook of brain theory and neural networks*. Cambridge, MA: MIT Press.

Christie, F. & Bruce, V. (1998). The role of dynamic information in the recognition of unfamiliar faces. *Memory and Cognition*, *26*(4):780–790.

Cudeiro, J. & Sillito, A. M. (1996). Spatial frequency tuning of orientation -discontinuity-sensitive corticofugal feedback to the cat lateral geniculate nucleus. *Journal of physiology*, *490.2*:481–492.

Dayan, P., Hinton, G. E., Neal, R., & Zemel, R. S. (1995). The helmholtz machine. *Neural Computation*, *7*:1022–1037.

De Angelis, G. C., Ohzawa, I., & Freeman, R. D. (1995). Receptive-field dynamics in the central visual pathways. *Trends in Neurosciences*, *18*(10):451–458.

Dempster, A. P., Laird, N. M., & Rubin, D. B. (1977). Maximum likelihood from incomplete data via the EM algorithm. *Proceedings of the Royal Statistical Society*, *B-39*:1–38.

Desimone, R., Albright, T. D., Gross, G., & Bruce, C. (1984). Stimulus-selective properties of inferior temporal neurons in the macaque. *The Journal of Neuroscience*, *4*(8):2051–2062.

Dong, D. W. & Atick, J. J. (1995). Temporal decorrelation: a theory of lagged and nonlagged responses in the lateral geniculate nucleus. *Network: Computation in neural systems*, *6*:159–178.

Douglas, R. & Martin, K. (1990). Neocortex. In G. M. Shepherd (Ed.), *The Synaptic Organization of the Brain* (pp. 389–438). New York: Oxford University Press.

Edelman, S. & Weinshall, D. (1991). A self-organizing multiple-view representation of 3D objects. *Biological Cybernetics*, *64*(3):209–219.

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

Gilbert, C. D., Das, A., Ito, M., Kapadia, M., & Westheimer, G. (1996). Spatial integration and cortical dynamics. *Proceedings of the National Academy of Sciences*, *93*:615–622.

Gross, C. G., Rocha-Miranda, C. E., & Bender, D. B. (1971). Visual properties of neurons in inferotemporal cortex of the macaque. *Journal of Physiology*, *35*:96–111.

Hess, D. J., Foss, D. J., & Carroll, P. (1995). Effects of global and local context on lexical processing during language comprehension. *Journal of Experimental Psychology: General*, *124*(1):62–82.

Hinton, G. E. & Zemel, R. S. (1994). Autoencoders, minimum description length, and helmholtz free energy. In J. D. Cowan, G. Tesauro, & J. Alspector (Eds.), *Advances in Neural Information Processing Systems 6* (pp. 3–10). San Mateo, CA: Morgan Kaufmann.

Jacobs, R. A., Jordan, M. I., Nowlan, S. J., & Hinton, G. E. (1991). Adaptive mixtures of local experts. *Neural Computation*, *3*(1):79–87.

Kay, J. & Phillips, W. A. (1997). Activation functions, computational goals, and learning rules for local processors with contextual guidance. *Neural Computation*, *9*(4):895–910.

Levitt, J. B., Kiper, D. C., & Movshon, J. A. (1994). Receptive fields and functional architecture of macaque v2. *Journal of neurophysiology*, *71*(6):2517–2541.

Linsker, R. (1988). Self-organization in a perceptual network. *IEEE Computer,* March, *21*:105–117.

MacDonald, J. & McGurk, H. (1978). Visual influences on speech perception processes. *Perception and Psychophysics*, *24*(3):253–257.

McClelland, J. L., McNaughton, B. L., & O'Reilly, R. C. (1995). Why there are complementary learning systems in the hippocampus and neocortex: Insights from the successes and failures of connectionist models of learning and memory. *Psychological Review*, *102*(3):419–457.

McClelland, J. L. & Rumelhart, D. E. (1981). An interactive activation model of context effects in letter perception, part I: An account of basic findings. *Psychological Review*, *88*:375–407.

McGurk, H. & MacDonald, J. (1976). Hearing lips and seeing voices. *Nature*, *264*:746–748.

Mel, B. W. (1994). Information processing in dendritic trees. *Neural Computation*, *6*(6):1031–1085.

Miyashita, Y. (1988). Neuronal correlate of visual associative long-term memory in the primate temporal cortex. *Nature*, *335*:817–820.

Mundel, T., Dimitrov, A., & Cowan, J. D. (1997). Visual cortex circuitry and orientation tuning. In *Advances in Neural Information Processing Systems 9*. MIT Press.

Neely, J. (1991). Semantic priming effects in visual word recognition: A selective review of current findings and theories. In D. Besner & G. W. Humphreys (Eds.), *Basic processes in reading: Visual Word Recognition* (pp. 264–336). Hillsdale, NJ: Lawrence Erlbaum Associates.

Nowlan, S. J. (1990). Maximum likelihood competitive learning. In D. S. Touretzky (Ed.), *Neural Information Processing Systems, Vol. 2* (pp. 574–582). San Mateo, CA: Morgan Kaufmann.

Nowlan, S. J. & Sejnowski, T. J. (1993). Filter selection model for generating visual motion signals. In S. Hanson, J. D. Cowan, & L. Giles (Eds.), *Advances in Neural Information Processing Systems 5* (pp. 369–376). San Mateo, CA: Morgan Kaufmann.

Oram, M. & Perrett, D. (1994). Modeling visual recognition from neurobiological constraints. *Neural Networks*, *7*(6/7):945–972.

O'Reilly, R. C. & Johnson, M. H. (1994). Objection recognition and sensitive periods: A computational analysis of visual imprinting. *Neural Computation*, *6*(3):357–389.

Perrett, D. I., Hietanen, J. K., Oram, M. W., & Benson, P. J. (1992). Organization and functions of cells responsive to faces in the temporal cortex. *Philosophical transactions of the royal society of London, B*, *335*:23–30.

Perrett, D. I., Rolls, E. T., & Caan, W. (1982). Visual neurones responsive to faces in the monkey temporal cortex. *Experimental Brain Research*, *47*:329–342.

Phillips, W. A., Kay, J., & Smyth, D. (1995). The discovery of structure by multi-stream networks of local processors with contextual guidance. *Network*, *6*:225–246.

Pouget, A. & Sejnowski, T. J. (1997). Spatial transformations in the parietal cortex using basis functions. *Journal of Cognitive Neuroscience*, *9*(2):222–237.

Rao, R. P. N. & Ballard, D. H. (1997). Dynamic model of visual recognition predicts neural response properties in the visual cortex. *Neural Computation*, *9*(4):721–764.

Ringach, D. L., Hawken, M. J., & Shapley, R. (1997). Dynamics of orientation tuning in macaque primary visual cortex. *Nature*, *387*:281–284.

Saul, L. K. & Jordan, M. I. (1996). Exploiting tractable substructures in intractable networks. In D. Touretzky, M. Mozer, & M. Hasselmo (Eds.), *Advances in Neural Information Processing Systems 8* (pp. 486–492). MIT Press.

Seergobin, K. (1996). Unsupervised learning: The impact of temporal and spatial coherence on the formation of visual representations. Master's thesis, Department of Psychology, McMaster University.

Sillito, A. M., Grieve, K. L., Jones, H. E., Cudeiro, J., & Davis, J. (1995). Visual cortical mechanisms detecting focal orientation discontinuities. *Nature*, *378*:492–496.

Stewart Bartlett, M. & Sejnowski, T. J. (1997). Viewpoint invariant face recognition using independent component analysis and attractor networks. In M. Mozer, M. Jordan, & T. Petsche (Eds.), *Neural Information Processing Systems 9* (pp. 817–823). MIT Press.

Stewart Bartlett, M. & Sejnowski, T. J. (1998). Learning viewpoint invariant face representations from visual experience by temporal association. In *Face recognition: From theory to applications, NATO ASI Series F*. Springer-Verlag.

Stone, J. (1996). Learning perceptually salient visual parameters using spatiotemporal smoothness constraints. *Neural Computation*, *8*:1463–1492.

Tanaka, K., Fujita, I., Kobatake, E., Cheng, K., & Ito, M. (1993). Serial processing of visual object-features in the posterior and anterior parts of the inferotemporal cortex. In T. Ono, L. R. Squire, M. E. Raichle, D. I. Perrett, & M. Fukuda (Eds.), *Brain Mechanisms of Perception and Memory, From Neuron to Behavior* (pp. 34–46). New York, NY: Oxford University Press.

Tanaka, K., Saito, H., Fukada, Y., & Moriya, M. (1991). Coding visual images of objects in the inferotemporal cortex of the macaque monkey. *Journal of Neurophysiology*, *66*(1):170–189.

Wallis, G. & Rolls, E. T. (1997). Invariant face and object recognition in the visual system. *Progress in Neurobiology*, *51*(2):167–194.

Weinshall, D., Edelman, S., & Bülthoff, H. H. (1990). A self-organizing multiple-view representation of 3D objects. In D. S. Touretzky (Ed.), *Advances in Neural Information Processing Systems 2* (pp. 274–282). Morgan Kaufmann.

Yamane, S., Kaji, S., & Kawano, K. (1988). What facial features activate face neurons in inferotemporal cortex of the monkey. *Experimental Brain Research*, *73*:209–214.