Learning Slowness in a Sparse Model of Invariant Feature Detection

Thusitha N. Chandrapala
tnc@ust.hk
Bertram E. Shi
eebert@ee.ust.hk
Department of Electronic and Computer Engineering, Hong Kong University of Science and Technology, Clear Water Bay, Kowloon, Hong Kong SAR

Primary visual cortical complex cells are thought to serve as invariant feature detectors and to provide input to higher cortical areas. We propose a single model for learning the connectivity required by complex cells that integrates two factors that have been hypothesized to play a role in the development of invariant feature detectors: temporal slowness and sparsity. This model, the generative adaptive subspace self-organizing map (GASSOM), extends Kohonen's adaptive subspace self-organizing map (ASSOM) with a generative model of the input. Each observation is assumed to be generated by one among many nodes in the network, each being associated with a different subspace in the space of all observations. The generating nodes evolve according to a first-order Markov chain and generate inputs that lie close to the associated subspace. This model differs from prior approaches in that temporal slowness is not an externally imposed criterion to be maximized during learning but, rather, an emergent property of the model structure as it seeks a good model of the input statistics. Unlike the ASSOM, the GASSOM does not require an explicit segmentation of the input training vectors into separate episodes. This enables us to apply this model to an unlabeled naturalistic image sequence generated by a realistic eye movement model. We show that the emergence of temporal slowness within the model improves the invariance of feature detectors trained on this input.

1 Introduction

Many neurons in the visual cortex can be thought of as invariant feature detectors: they exhibit selectivity along certain stimulus dimensions and invariance along others. Processing in the visual cortex is often thought to be hierarchical, with neurons in higher areas elaborating on sensory representations encoded by lower areas. Neurons in higher areas are generally selective to more complex stimuli. For example, neurons in the inferior temporal (IT) cortex respond selectively to complex objects like faces (Perrett,

Neural Computation **27, 1496–1529** (2015)
doi:10.1162/NECO_a_00743

Rolls, & Caan, 1982), whereas complex cells in the primary visual cortex (V1) respond to simpler stimuli, like bars and sinusoidal gratings (Hubel & Wiesel, 1962, 1968). Neurons in higher areas often exhibit invariance along stimulus dimensions along which neurons in lower layers are selective. For example, IT neuron responses are orientation and scale invariant, whereas V1 complex cells are orientation and scale selective.

Two different computational principles have been proposed as the computational basis for the development of invariant feature detectors: temporal slowness and sparsity. The principle of temporal slowness or temporal coherence assumes that neurons encode information about the environment, which is relatively stable in comparison to the raw sensory signals. For example, even during fixation at a single point in a static environment, the input to one photoreceptor varies dramatically due to fixational eye movements, such as drift, microsaccades, and microtremors (Rolfs, 2009). Slowness has been shown to be a plausible criterion for learning the invariance exhibited by complex cells in the primary visual cortex (Földiák, 1991). For example, slow feature analysis (SFA) tries to uncover slowly varying features from quickly varying sensory input signals by finding functions of the input whose temporal variation is small (Berkes & Wiskott, 2005; Wiskott & Sejnowski, 2002). Slowness may also underlie the formation of position-invariant object representations in higher cortical areas, such as inferior temporal cortex (Li & DiCarlo, 2008).

Sparsity is inspired by the efficient coding hypothesis, which posits that neural population responses represent sensory data using as few active neurons as possible (Barlow, 1961; Olshausen, 1996; Olshausen & Field, 1997). Since neural firing is metabolically expensive, sparse representations are energy efficient (Hasenstaub, Otte, Callaway, & Sejnowski, 2010). For linear models, sparsity is closely related to the independent component analysis (ICA) (Bell & Sejnowski, 1997; Hyvarinen, Oja, Hoyer, & Hurri, 1998). Subspace learning models based on maximizing sparsity or independence lead to invariant feature detectors with properties similar to visual cortical complex cells (Hyvärinen & Hoyer, 2000; 2001; Kavukcuoglu, Ranzato, Fergus, & Le Cun, 2009).

Although both slowness and sparsity might lead to the development of invariant feature detectors, there is no clear agreement on their relative contributions in neuronal development. It has been argued that sparsity should be preferred, as it provides a more parsimonious explanation since it makes no assumptions about the temporal structure of the input image sequences (Hyvärinen & Hoyer, 2000). Recently Lies, Häfner, & Bethge (2014) found further support for sparsity over slowness in a study comparing the outputs of two different models: one that maximizes sparsity and one that maximizes slowness. Their experiments suggest that maximizing sparsity leads to the localized Gabor-like receptive fields commonly associated with visual cortical complex cells, whereas maximizing slowness does not. The results of their experiments maximizing a combined sparseness/slowness criterion did not reveal any advantage to incorporating slowness.

These results suggest that slowness does not play a strong role in the development of invariant feature detectors. However, it seems unlikely that the processes underlying neuronal development would consider only the spatial statistical structure while ignoring the temporal statistical structure available in the input. We suggest that the role of slowness may be undervalued because past work has largely approached sparsity and slowness similarly: as objectives that are explicitly maximized during learning and development. This viewpoint makes sense for sparsity, given that maximizing the sparsity of neural representation of the input minimizes energy consumption, and thus has clear survival benefits for the organism, independent of the statistics of the input being represented. However, slowness is based on an assumption about the relationship between environment and the sensory input, which depends critically on the organism's behavior in the environment. Thus, dictating that slowness is maximized during development is essentially hard-coding an assumption about the agent's behavior in the environment into the developmental rules of the organism. A more parsimonious approach would be to allow the organism to discover the underlying temporal structure in the sensory input.

In this letter, we describe the generative adaptive subspace self-organizing Map (GASSOM) model, a statistical generative model for time-varying sensory input that combines sparsity and slowness in the following manner. Sparsity is explicitly encoded as an assumption in the model likelihood. Slowness is not explictly encoded in the model, but rather can be reflected through particular model parameter choices. Model parameters can be learned by maximizing the likelihood given an input data sequence.

When the GASSOM model is exposed to an image sequence generated by a model of human eye movements during free viewing, it learns model parameters that correspond to slowness. Thus, rather than being externally imposed, slowness emerges naturally as the model learns the spatial and temporal structure of the input statistics.

Using the GASSOM framework, we examine the relative contributions of several components that are commonly found in developmental models of cortical neuron populations, including slowness and topological organization. In contrast to Lies et al. (2014), who found no advantage to incorporating slowness, we find that slowness improves the invariance of the model responses. We find no significant benefit of topological organization in the development of invariance. We also find that nodes within the network preferentially connect to other nodes with similar preferred orientations.

The GASSOM model was originally proposed in Chandrapala and Shi (2014). Here we extend that prior work with the introduction of new learning rules, including a fully probabilistic learning rule that does not select a single winner and include more extensive comparisons between different learning rules and a study of the role of topographical organization of the maps learned by the network.

2 Methods

In this section, we describe a generative model of image sequences that incorporates both sparsity and temporal slowness. Since this model is based on the adaptive subspace self-organizing map (ASSOM) proposed by Kohonen (1996), we call it the generative ASSOM (GASSOM). Elements of the model have been introduced before (Chandrapala & Shi, 2014), but we describe the model in detail for completeness.

Invariant feature detection is closely related to the concept of subspaces or manifolds of the input space. Natural inputs, such as auditory or visual signals, are usually high dimensional but are distributed along many lower-dimensional manifolds or subspaces (Hyvarinen, Hurri, & Hoyer, 2009; Kohonen, 1996). For example, although objects in the visual field may be quite complex, neurons in the initial stages of visual processing respond to relatively small spatial regions. At this scale, the typical inputs may be relatively simple and regular (e.g., oriented bars or edges), but vary according to a class of transformations, such as translation. The inputs corresponding to one common pattern subject to different transformations within this class are referred to as an invariance class and can be approximated as lying close to a manifold of the high-dimensional input space. For small transformations and similar patterns, this manifold can often be approximated as a linear subspace. Invariant features correspond to subspaces, with the strength of each feature depending on the length of the projection of the input onto the subspace.

The GASSOM removes the ASSOM's requirement that the input be explicitly separated into training episodes, each consisting of inputs from the same invariance class. This enables us to apply the new model to naturalistic unlabeled image sequences, such as those generated by the eye movement model described below. We begin by reviewing the ASSOM model before describing the GASSOM. We conclude this section with a description of an eye movement model that includes the effect of drift and saccades, which we used to generate the image sequence inputs to the models.

2.1 The Adaptive Subspace Self Organizing Map.

Kohonen's (ASSOM) (Kohonen, 1996; Kohonen, Kaski, Lappalainen, & Saljärvi, 1997; Kohonen, Kaski, & Lappalainen, 1997) is an extension to his self-organized Map (SOM) (Kohonen, 1990). Instead of finding a set of vectors that model the distribution of a set of N-dimensional observations, the ASSOM finds a set of low-dimensional subspaces.

Like the SOM, the ASSOM consists of a set of S nodes indexed by $i \in \{1, \ldots, S\}$. As illustrated in Figure 1a, the nodes are organized in a 2D latent space with locations $\mathbf{l}_i \in \mathbb{R}^2$. Associated with each node is an H-dimensional

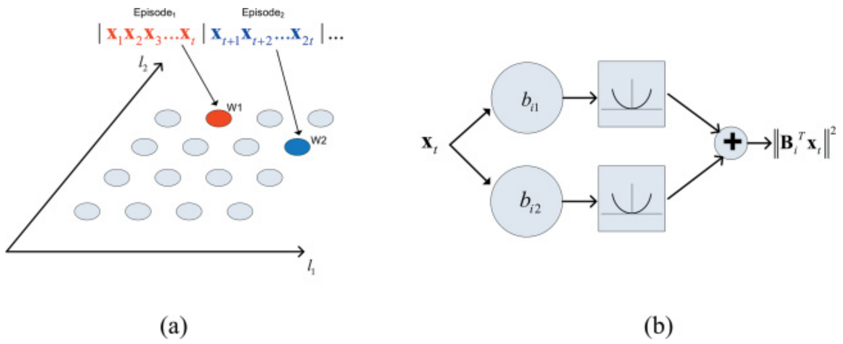

Episode$_1$
Episode$_2$
$| \mathbf{x}_1 \mathbf{x}_2 \mathbf{x}_3 \ldots \mathbf{x}_t | \mathbf{x}_{t+1} \mathbf{x}_{t+2} \ldots \mathbf{x}_{2t} | \ldots$
l_2
W1
W2
l_1
(a)
$\mathbf{x}_t$
b_{i1}
b_{i2}
$\|\mathbf{B}_i^T \mathbf{x}_t\|^2$
(b)

Figure 1: (a) The latent nodes of the ASSOM are organized in a two-dimensional latent space. Input vectors are organized into training episodes, with each episode being assigned to one "winning" node. (b) Each node is an invariant feature detector. Its "output" is the squared length of the projection of the input vector onto the subspace associated with the node. For two-dimensional subspaces, the computation of the squared projection length is similar to computations used in energy models of complex cell responses in the visual cortex. The two basis vectors b_{i1} and b_{i2} are analogous to linear receptive fields of simple cells whose squared outputs are summed to obtain the complex cell response.

subspace of $\mathbb{R}^N$ defined by a set of H orthonormal basis vectors specified by the columns of the matrix $\mathbf{B}_i = [\mathbf{b}_{i1} \ldots \ \mathbf{b}_{ih} \ \ldots \ \mathbf{b}_{iH}]$.

The input to the network is a sequence of vectors $\mathbf{x}(t) \in \mathbb{R}^N$ where t represents time. As illustrated in Figure 1b, we define the response of each node to an input as the squared length of the projection of an input vector $\mathbf{x}(t)$ onto the subspace:

$$\|\mathbf{B}_i^T \mathbf{x}(t)\|^2 = \sum_{h=1}^{H} (\mathbf{b}_{ih}^T \mathbf{x}(t))^2. \tag{2.1}$$

This calculation is similar to that used in energy models of the visual cortical complex cell responses. Basis vectors are analogous to simple cell linear receptive fields.

The ASSOM learns features that are invariant to transformations of the input patterns. The input data are organized into episodes, each consisting of training vectors that are similar but vary along the invariant dimension. For example, in order to learn translation-invariant features, each episode consists of patterns generated by randomly translating an image patch in different directions.

For each episode, the node whose subspace minimizes the total squared projection error over all input patterns in the episode is selected as the "winner." If we denote the index of the winning node by c, then

$$c = \arg\min_{j} \sum_{\tau \in \mathcal{E}} \|\tilde{\mathbf{x}}_j(\tau)\|^2, \tag{2.2}$$

where $\mathcal{E}$ is the set of time indices within the episode and $\tilde{\mathbf{x}}_j(t) = \mathbf{x}(t) - \hat{\mathbf{x}}_j(t)$ is the difference between the input pattern at time t and its projection onto subspace j, where the projection is given by

$$\hat{\mathbf{x}}_j(t) = \mathbf{B}_j \mathbf{B}_j^T \mathbf{x}(t). \tag{2.3}$$

The ASSOM captures the concept of temporal slowness through the episode, which consists of a temporal sequence of the same input transformed along the invariant dimension. The ASSOM captures the concept of sparsity by assigning each episode to a single node in the network.

For each observation vector, we compute a correction to the basis vectors of each subspace:

$$\Delta\mathbf{B}_i(t) = h_i \cdot \tilde{\mathbf{x}}_i(t) \cdot \frac{\mathbf{x}(t)^T \mathbf{B}_i}{\|\hat{\mathbf{x}}_i(t)\| \, \|\mathbf{x}(t)\|}. \tag{2.4}$$

The coefficient h_i determines the amount that subspace i is updated toward the observation $\mathbf{x}(t)$ and is determined by a gaussian neighborhood function around the winning node:

$$h_i = \frac{g(\mathbf{l}_i | \mathbf{l}_c, \sigma\mathbf{I})}{\sum_{k=1}^{S} g(\mathbf{l}_k | \mathbf{l}_c, \sigma\mathbf{I})}, \tag{2.5}$$

where $g(\mathbf{l}|\mathbf{m}, \mathbf{C})$ is an n-dimensional gaussian density function with mean $\mathbf{m}$ and covariance matrix $\mathbf{C}$ and $n = 2$. The subspaces of the winning node, c, and its neighbors are updated the most. The width of the neighborhood is controlled by σ. This update rule creates a topological map of smoothly varying subspaces. This learning rule is slightly modified from the original to improve computational efficiency (Zheng, Lefebvre, & Laurent, 2008).

Subspaces are updated after each episode by accumulating the corrections for all observations in the episode,

$$\mathbf{B}_i^{\text{new}} = \mathbf{B}_i^{\text{old}} + \lambda \sum_{t \in \mathcal{E}} \Delta B_i(t), \tag{2.6}$$

where $\lambda > 0$ is the learning rate.

2.2 The Generative ASSOM Model.

The explicit separation of the training data into episodes required by the ASSOM has several disadvantages. First, it makes it difficult to update the subspaces online. Since the decision about the winning node is made only after all inputs in the episode have been seen, each input within the episode must be stored in order to compute the subspace update in equation 2.6. Second, it explicitly encodes temporal slowness in the model, which makes it difficult to examine the effect of slowness. Finally, it requires extra information that must be supplied during training.

The GASSOM algorithm seeks to overcome these disadvantages through a generative formulation based on the hidden Markov model (HMM) (Rabiner & Juang, 1986; Rabiner, 1989). As in the standard ASSOM, we assume a set of S nodes arranged in a 2D latent space. At each time t, the observation $\mathbf{x}(t)$ is generated by one of the nodes, which is identified by the indicator vector $\mathbf{z}(t) \in \{0, 1\}^S$ according to a 1 of S coding,

$$\mathbf{z}(t) = [z_1(t) \ldots z_i(t) \ldots z_S(t)]^T, \tag{2.7}$$

where

$$z_i(t) = \begin{cases} 1 & \text{if } \mathbf{x}(t) \text{ is generated by subspace } i \\ 0 & \text{otherwise} \end{cases} . \tag{2.8}$$

The GASSOM includes sparsity by the assumption that each observation is generated by only one node.

The sequence of generating nodes evolves according to a Markov chain. The probability that each node generated the initial observation is given by

$$\pi_i = P(z_i(0) = 1). \tag{2.9}$$

We will assume that $\pi_i = S^{-1}$ for all i. For subsequent observations, the generating node changes according to a transition probability,

$$a_{ij} = P(z_j(t) = 1 | z_i(t-1) = 1). \tag{2.10}$$

As we describe below, the transition probabilities can be learned. We can incorporate slowness into the model by particular choices of the transition probabilities. One possibility is to choose the transition probability to be a mixture of a uniform distribution and a discrete delta distribution:

$$a_{ij} = \frac{\rho}{S} + (1-\rho)\delta(i-j). \tag{2.11}$$

The mixture weights are determined by $\rho \in [0, 1]$. The discrete delta distribution captures the concept of temporal slowness. The node generating $\mathbf{x}(t)$ is likely to be the same as the node that generated $\mathbf{x}(t-1)$. The uniform distribution allows for arbitrary changes in the locations of the generating node, as assumed at episode boundaries in the original ASSOM. The mixing parameter ρ controls the amount of slowness in the model. For example, we can remove slowness by setting $\rho = 1$. In this case, the sequence of generating nodes is assumed to be an independent and identically distributed process.

Given the latent state $\mathbf{z}(t)$, we assume that the observation $\mathbf{x}(t)$ is the sum of two components: one lying in the subspace and one orthogonal to the subspace,

$$\mathbf{x}(t) = \mathbf{B}_i \mathbf{w}(t) + \mathbf{B}_i^{\perp} \mathbf{n}(t), \tag{2.12}$$

where $\mathbf{w}(t) \in \mathbb{R}^H$, $\mathbf{B}_i^{\perp} \in \mathbb{R}^{N \times (N-H)}$ is a matrix with orthogonal columns spanning the orthogonal complement of $\mathbf{B}_i$ and $\mathbf{n}(t) \in \mathbb{R}^{N-H}$. We assume that $\mathbf{w}(t)$ and $\mathbf{n}(t)$ are independent and gaussian distributed with diagonal covariance matrices. By orthogonality, we have that $\mathbf{w}(t) = \mathbf{B}_i^T \mathbf{x}(t)$ and $\mathbf{n}(t) = (\mathbf{B}_i^{\perp})^T \mathbf{x}(t)$. Thus,

$$P(\mathbf{x}(t) | z_i(t) = 1) = g(\mathbf{w}(t) | 0, \sigma_W^2 \mathbf{I}) \cdot g(\mathbf{n}(t) | 0, \sigma_N^2 \mathbf{I}). \tag{2.13}$$

As we describe below, the variance parameters, σ_W^2 and σ_N^2, can be learned. We generally expect that $\sigma_N^2 \ll \sigma_W^2$, so that input vectors lying in the subspace have large probability and input vectors off the subspace have low probability.

2.2.1 Subspace Updates. The equation determining the update to each subspace in the network is very similar to that of the ASSOM:

$$\Delta \mathbf{B}_i(t) = h_i(t) \cdot \tilde{\mathbf{x}}_i(t) \cdot \frac{\mathbf{x}(t)^T \mathbf{B}_i}{\|\hat{\mathbf{x}}_i(t)\| \, \|\mathbf{x}(t)\|}. \tag{2.14}$$

The key differences between this update rule and the update rule for the ASSOM in equation 2.4 are in the calculation of $h_i(t)$. First, the coefficient $h_i(t)$ changes for different observations, whereas in equation 2.4, it was identical for all observations within the same episode. Second, as we discuss in more detail below, the probabilistic formulation suggests several ways to calculate $h_i(t)$.

To obtain subspace updates similar to those of the ASSOM, we calculate the vector of coefficients $\mathbf{h}(t) = [h_1(t) \ \ \ldots \ \ h_S(t)]^T$ according to

$$\mathbf{h}(t) = \mathbf{G} \cdot \mathrm{WTA}(\boldsymbol{\gamma}(t)). \tag{2.15}$$

The vector $\boldsymbol{\gamma}(t) = [\gamma_1(t) \ \ \ldots \ \ \gamma_S(t)]^T$ contains the conditional probabilities that node i generated the observation $\mathbf{x}(t)$ given a set of observations, $\mathcal{X}$:

$$\gamma_i(t) = P(z_i(t) = 1 | \mathcal{X}). \tag{2.16}$$

We elaborate on the computation of $\boldsymbol{\gamma}(t)$ at the end of this section. The WTA (winner-take-all) function identifies the node most likely to have generated $\mathbf{x}(t)$,

$$\boldsymbol{\omega}(t) = \mathrm{WTA}(\boldsymbol{\gamma}(t)), \tag{2.17}$$

where the elements of $\boldsymbol{\omega}(t)$ are given by

$$\omega_i(t) = \delta\left(i - \arg\max_j[\gamma_j(t)]\right), \tag{2.18}$$

where $\delta(i)$ indicates the discrete delta function, which equals one if $i = \mathrm{argmax}_j[\gamma_j(t)]$ and zero otherwise. The matrix $\mathbf{G}$ is a gaussian smoothing matrix whose elements are given by

$$G_{ij} = \frac{g(\mathbf{l}_i|\mathbf{l}_j, \sigma\mathbf{I})}{\sum_{k=1}^{S} g(\mathbf{l}_k|\mathbf{l}_j, \sigma\mathbf{I})}. \tag{2.19}$$

We refer to the conditional probability $\gamma_i(t)$ in equation 2.16 as the responsibility of node i for generating $\mathbf{x}(t)$. We consider two definitions of the responsibility, batch and online, depending on what conditioning observations are used. In the batch calculation, we assume that the algorithm has access to a batch of T observations, $\mathcal{X} = [\mathbf{x}(0)\ldots\ \mathbf{x}(t)\ldots\mathbf{x}(T-1)]$, and we define the responsibility by

$$\gamma_i(t) = P(z_i(t) = 1|\mathcal{X}) \text{ for } 0 \leq t < T. \tag{2.20}$$

In the online calculation, we assume the algorithm has access only to the current and past observations, $\mathcal{X}(t) = [\mathbf{x}(0)\ldots\ \mathbf{x}(t)]$ and define the responsibility by

$$\hat{\gamma}_i(t) = P(z_i(t) = 1|\mathcal{X}(t)). \tag{2.21}$$

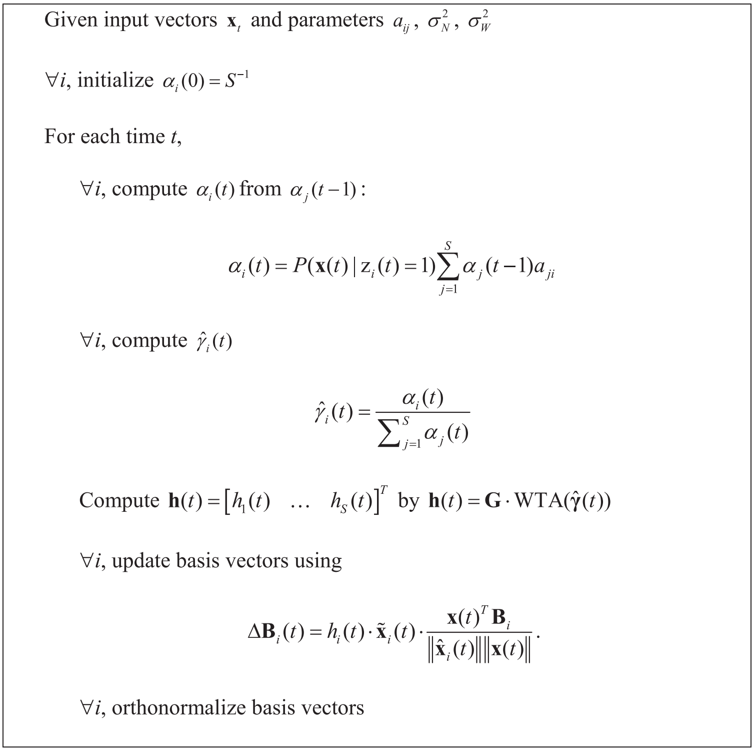
In the batch method, we can use the forward-backward algorithm (Rabiner, 1989) to compute the values of the responsibility. The batch responsibility is given by

$$\gamma_i(t) = \frac{1}{K(t)}\alpha_i(t)\beta_i(t), \tag{2.22}$$

where

$$\alpha_i(t) = P(\mathbf{x}(0), \ldots, \mathbf{x}(t), z_i(t) = 1), \tag{2.23}$$

$$\beta_i(t) = P(\mathbf{x}(t+1), \ldots, \mathbf{x}(T-1)|z_i(t) = 1), \tag{2.24}$$



```
Given input vectors $\mathbf{x}_t$ and parameters $a_{ij}$, $\sigma_N^2$, $\sigma_W^2$

$\forall i$, initialize $\alpha_i(0) = S^{-1}$

For each time $t$,

    $\forall i$, compute $\alpha_i(t)$ from $\alpha_j(t-1)$:

        $\alpha_i(t) = P(\mathbf{x}(t) \mid \mathrm{z}_i(t) = 1) \sum_{j=1}^{S} \alpha_j(t-1) a_{ji}$

    $\forall i$, compute $\hat{\gamma}_i(t)$

        $\hat{\gamma}_i(t) = \frac{\alpha_i(t)}{\sum_{j=1}^{S} \alpha_j(t)}$

    Compute $\mathbf{h}(t) = \left[h_1(t) \quad \ldots \quad h_S(t)\right]^T$ by $\mathbf{h}(t) = \mathbf{G} \cdot \mathrm{WTA}(\hat{\boldsymbol{\gamma}}(t))$

    $\forall i$, update basis vectors using

        $\Delta \mathbf{B}_i(t) = h_i(t) \cdot \tilde{\mathbf{x}}_i(t) \cdot \frac{\mathbf{x}(t)^T \mathbf{B}_i}{\|\hat{\mathbf{x}}_i(t)\| \|\mathbf{x}(t)\|}$.

    $\forall i$, orthonormalize basis vectors
```

Figure 2: Pseudocode for the online winner selection based GASSOM algorithm.

and $K(t) = \sum_{j=1}^{S} \alpha_j(t)\beta_j(t)$. For the online method, since only current and past information is available, the responsibility values can be calculated using only the forward algorithm. The online responsibility is given by

$$\hat{\gamma}_i(t) = \frac{\alpha_i(t)}{\sum_{j=1}^{S} \alpha_j(t)}. \tag{2.25}$$

The values of $\alpha_i(t)$ and $\beta_i(t)$ can be computed recursively—for example,

$$\alpha_i(t) = P(\mathbf{x}(t) | z_i(t) = 1) \sum_{j=1}^{S} \alpha_j(t-1) a_{ji}. \tag{2.26}$$

A similar backward recursion in time can be used to compute $\beta_i(t)$. Pseudocode for the online method is given in Figure 2.

Based on the HMM formulation presented here, it is also possible to use the Viterbi algorithm to assign each observation to a node rather than applying the WTA to the responsibility. Our preliminary experiments (not reported here) indicated little difference between the results of the two approaches, so we present only results using the forward-backward algorithm, as it is most consistent with the original ASSOM algorithm.

2.2.2 The Relationship between the GASSOM and the ASSOM. The subspace update rule for the batch GASSOM is equivalent to that for the ASSOM when (1) the batch contains a single training episode, (2) $\rho = 0$, and (3) $\sigma_W^2 \to \infty$. We prove this by showing that the coefficients $h_i(t)$ in equation 2.15 are identical to the h_i in equation 2.5.

Under assumption 2.2, the transition probability matrix reduces to the identity matrix. This implies that the generating node does not change over time. In this case, equation 2.26 simplifies to $\alpha_i(t) = P(\mathbf{x}(t)|z_i(t) = 1)\alpha_i(t-1)$, which implies that

$$\alpha_i(t) = \prod_{\tau=1}^{t} P(\mathbf{x}(\tau)|z_i(\tau) = 1). \tag{2.27}$$

A similar analysis for the backward algorithm implies that

$$\beta_i(t) = \prod_{\tau=t+1}^{T} P(\mathbf{x}(\tau)|z_i(\tau) = 1). \tag{2.28}$$

Combining equations 2.22, 2.27, and 2.28, we find that for all t,

$$\gamma_i(t) \propto \prod_{\tau=1}^{T} P(\mathbf{x}(\tau)|z_i(\tau) = 1). \tag{2.29}$$

Note that the responsibility is constant in t. By assumption 2.4,

$$P(\mathbf{x}(\tau)|z_i(\tau) = 1) \propto \exp\left(-\frac{\|\tilde{\mathbf{x}}_i(\tau)\|}{2\sigma_N^2}\right). \tag{2.30}$$

Combining equations 2.29 and 2.30, we find that

$$\log \gamma_i(t) = -\frac{1}{2\sigma_N^2}\sum_{\tau=1}^{T}\|\tilde{\mathbf{x}}_i(\tau)\|^2 + k, \tag{2.31}$$

where k is a normalizing constant. Thus, the node with maximum responsibility is the same as the winner, c, in equation 2.2, which minimizes the projection error. The output of the WTA in equation 2.18 is given by $\omega_i(t) = \delta(i - c)$. Thus,

$$h_i(t) = \sum_{j=1}^{S} \frac{g(\mathbf{l}_i|\mathbf{l}_j, \sigma^2\mathbf{I})}{\sum_{k=1}^{S} g(\mathbf{l}_k|\mathbf{l}_j, \sigma^2\mathbf{I})} \omega_j(t) = \frac{g(\mathbf{l}_i|\mathbf{l}_c, \sigma^2\mathbf{I})}{\sum_{k=1}^{S} g(\mathbf{l}_k|\mathbf{l}_c, \sigma^2\mathbf{I})}. \tag{2.32}$$

2.2.3 Variants of the Subspace Update. Different learning rules can be obtained by choosing different combinations of online or batch computation of the responsibility, hard or soft winner selection, and whether to apply topological smoothing.

We can obtain an online algorithm for subspace learning by replacing $\boldsymbol{\gamma}(t)$ in equation 2.15 by $\hat{\boldsymbol{\gamma}}(t)$:

$$\mathbf{h}(t) = \mathbf{G} \cdot \mathrm{WTA}(\hat{\boldsymbol{\gamma}}(t)). \tag{2.33}$$

The batch algorithm requires storage of all observations in the batch in order to compute both $\alpha_i(t)$ and $\beta_i(t)$. Thus, we typically update each subspace once per batch using an equation similar to equation 2.6. On the other hand, since the online algorithm requires only $\alpha_i(t)$, it does not require storage of past or future observations. Thus, we are free to update the subspaces after each observation, that is, $\mathbf{B}_i^{\text{new}} = \mathbf{B}_i^{\text{old}} + \lambda \Delta B_i(t)$. Our results in the next section demonstrate that there is little difference between subspaces learned with the online versus batch algorithms.

Rather than making a hard decision about the generating node using the WTA, we can obtain a soft update rule by using the responsibility directly:

$$\mathbf{h}(t) = \mathbf{G} \cdot \boldsymbol{\gamma}(t). \tag{2.34}$$

Finally, we can choose not to apply topological smoothing, resulting in the simplest update rule:

$$\mathbf{h}(t) = \boldsymbol{\gamma}(t). \tag{2.35}$$

2.2.4 Learning the Transition and Emission Probabilities. For the batch algorithm, the transition probabilities can be estimated using the Baum-Welch algorithm (Rabiner, 1989). We calculate

$$\begin{aligned}\xi_{i,j}(t) &= P(z_i(t) = 1, z_j(t+1) = 1|\mathcal{X}) \\ &= \frac{\alpha_i(t) a_{ij} P(\mathbf{x}(t+1)|z_j(t+1) = 1) \beta_j(t+1)}{P(\mathcal{X})}\end{aligned} \tag{2.36}$$

and update the estimate of the transition probabilities a_{ij} by

$$a_{ij}^{\text{new}} = (1 - \lambda_{tp})a_{ij}^{\text{old}} + \lambda_{tp} \frac{\sum_{\tau=1}^{T-1} \xi_{i,j}(\tau)}{\sum_{\tau=1}^{T-1} \gamma_i(\tau)}. \tag{2.37}$$

The parameters σ_N and σ_W of the emission probability $P(\mathbf{x}(t)|z_i(t) = 1)$ can be estimated using

$$\hat{\sigma}_N^2 = \frac{1}{S(N-H)} \cdot \sum_{i=1}^{S} \left[\frac{\sum_{\tau=1}^{T} \gamma_i(\tau) \|\tilde{\mathbf{x}}_i(\tau)\|}{\sum_{\tau=1}^{T} \gamma_i(\tau)} \right], \tag{2.38}$$

$$\hat{\sigma}_W^2 = \frac{1}{SH} \cdot \sum_{i=1}^{S} \left[\frac{\sum_{\tau=1}^{T} \gamma_i(\tau) \left\|\hat{\mathbf{x}}_i(\tau)\right\|}{\sum_{\tau=1}^{T} \gamma_i(\tau)} \right]. \tag{2.39}$$

2.3 Eye Movement Model for Input Sample Generation. In the human visual system, drift during fixation produces a set of visual samples with a linear displacement around the fixation point (Rolfs, 2009). Saccades introduce large shifts in the fixation point. Thus, fixations are analogous to the training episodes, with successive fixations being delineated by saccades. The input stimuli in our experiments are generated by a model based on saccades and ocular drifts as described in Kuang, Poletti, Victor, and Rucci (2012). Saccade amplitudes are modeled using an exponential distribution with mean 2 degrees. The saccade direction is uniform over $[0, 2\pi)$. The intersaccadic interval follows an exponential distribution with mean 300 ms. Ocular drift is modeled as a diffusion process with a diffusion constant of 40 arcmin^2/sec.

The sequence of image patches presented to the model is taken from the natural images in Van Hateren database (van Hateren & van der Schaaf, 1998), which are first prewhitened (Olshausen & Field, 1997). Time in the eye movement model is discretized at 25 ms per frame. Visual space is quantized at 1 arcmin to one pixel. At the beginning, an image from the database and a gaze point in the image are chosen randomly. The 10×10 pixel image patch input to the model is obtained by interpolating the image around the gaze point. Patches are normalized to have zero mean and unit norm. At each time point, a new gaze point in the image is chosen according to the diffusion process or saccadic shift. This new gaze point is then used to obtain another image patch. Every 20 saccades, a new image from the database is chosen and the gaze point is reinitialized randomly.

3 Results

3.1 Parameter Settings. The maps consist of 16×16 arrays of latent nodes. The subspace of each latent node is two-dimensional. For each subspace, basis vectors are initialized by first choosing two independent and identically distributed (i.i.d.) vectors whose components are chosen from a uniform distribution between -1 and $+1$ and then orthonormalizing them.

Inputs to the model are sequences of 100-dimensional input vectors encoding the 10×10 pixel image patches generated by the eye movement model. For the ASSOM simulations, training episodes are defined by the period between two successive saccades, and subspaces are updated after each episode. For the batch GASSOM, one batch contains 20 saccades, and subspaces are updated after each batch. For the online GASSOM, updates are accumulated and updated every 12 frames. The accumulation is done only to increase computational speed. Neither the batch nor the online GASSOM has any information regarding the saccade locations. Training sequences contained 4×10^3 batches (i.e., 8×10^4 saccades or approximately 9.6×10^5 frames).

Learning rates for the subspace updates, λ, decayed exponentially from 1 to 0.05 with a time constant of 400 batches for the batch algorithm and 8000 saccades for the online algorithm. For topographical smoothing, the width of the 2D gaussian in equation 2.19 decayed from $\sigma = 4$ to 0.5 with a time constant of 400 batches for the batch algorithm and 8000 saccades for the online algorithm. In algorithms where the transition probabilities are learned, we use constant learning rate of 1×10^{-2} for the transition probability updates.

Figure 3 shows the evolution of the model likelihood over 1.2×10^4 batches using the batch GASSOM with soft winner selection and topological smoothing (see equation 2.34) using two fixed learning rates (1 and 0.05), as well as an exponentially decaying learning rate from 1 to 0.05. The larger fixed learning rate gives a faster initial increase in the likelihood but settles at a lower value with larger oscillation than the smaller fixed learning rate. The exponentially decaying learning rate has the advantages of both a rapid initial increase and a higher final value with small oscillation. We see that the likelihood values are stable after 4×10^3 batches, the length of the training sequence in the rest of our experiments.

3.2 Learning the Transition and Emission Probabilities. We demonstrate here that the GASSOM learns model parameters that are consistent with the concept of slowness when exposed to image sequences generated by the eye movement model.

We used equations 2.36 through 2.39 for the batch GASSOM to learn the transition probabilities and the parameters σ_N and σ_W of the emission probability distributions. The elements of the transition probability matrix were initialized to be equal with small random fluctuations to break symmetry,

$$a_{ij} = \frac{1}{S} + n_{ij}, \tag{3.1}$$

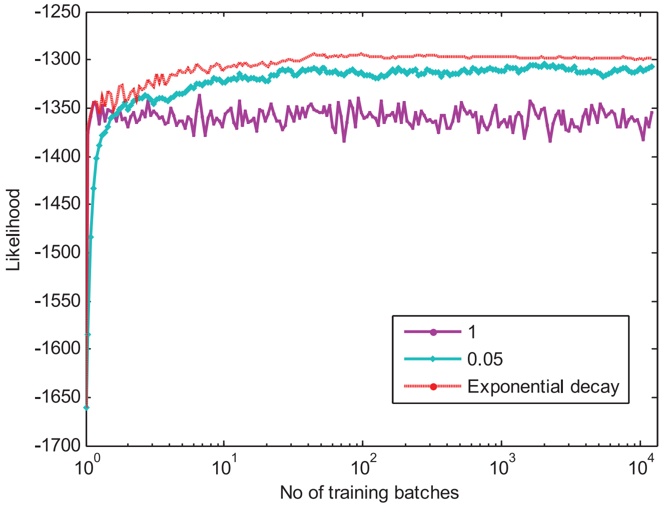


Figure 3: The evolution of model likelihood of the batch GASSOM with soft winner selection and topological smoothing for two fixed learning rates (1 and 0.05) and a learning rate that decays exponentially from 1 to 0.05 with a time constant of 4×10^2 batches. The likelihood was evaluated at 200 checkpoints on a separate testing data set not seen during training.

where the n_{ij} were chosen from a uniform distribution between $\pm 5 \times 10^{-4}$. A small offset was added to ensure that each row of the transition probability matrix summed to one. Basis vectors were initialized randomly as described above and were learned at the same time. The standard deviations of the gaussian emissions were initialized to $\sigma_N = 0.25$ and$\sigma_W = 1.25$. After learning, the emission probability distribution parameters were $\sigma_N = 0.08$ and $\sigma_W = 0.4$.

Figure 4a shows the learned transition probabilities learned by the batch GASSOM with soft winner selection and without topological smoothing (see equation 2.35). These transition probabilities are similar to those in equation 2.11, which encode slowness into the GASSOM. The cutout shows a large self-transition probability a_{ii}. This is further illustrated in Figure 5, which shows that the self-transition probabilities are about four orders of magnitude larger than transition probabilities to other nodes.

Figure 4b shows the transition probabilities learned by the batch GASSOM with soft winner selection and topographical smoothing (see equation 2.34). The topographical smoothing leads neighboring nodes to have similar subspaces. Thus, if the observations vary slowly due to drift, the node generating the current observation will likely be close to the node that generated the past observation. This leads the transition probabilities

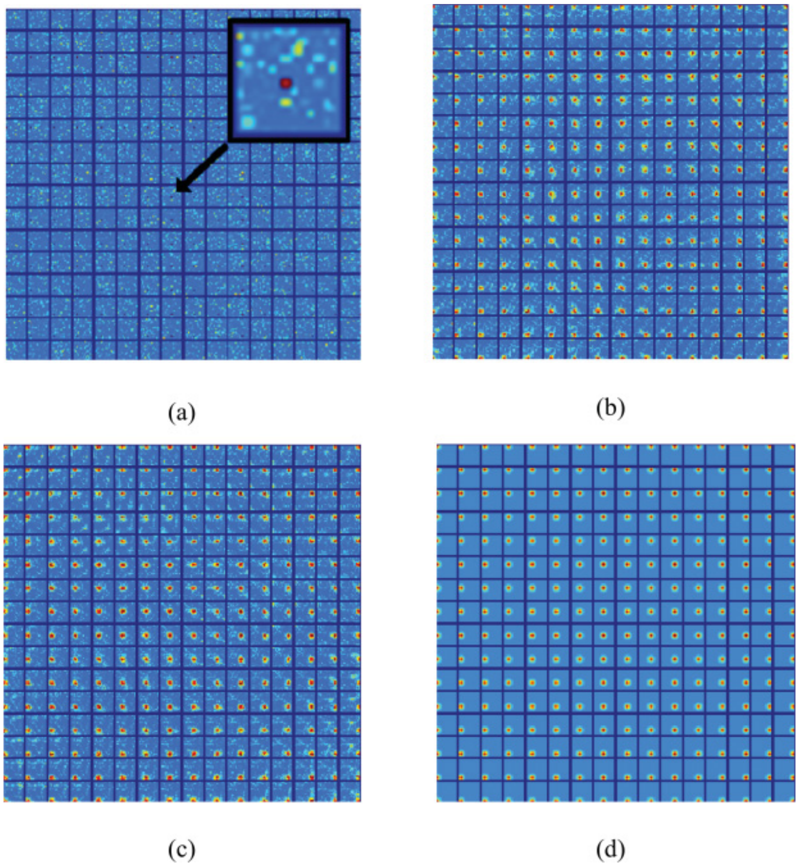
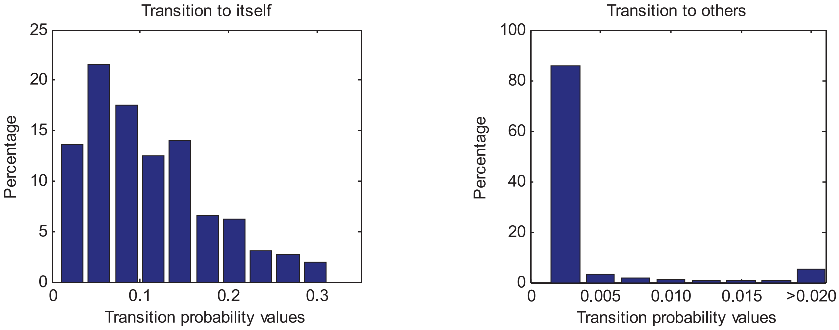


Figure 4: Learned transition probabilities. The probabilities are presented as a 16×16 array of subimages. Each 16×16 pixel subimage represents the transition probabilities $a_{ij} = P(z_{t,j} = 1 | z_{t-1,i} = 1)$ for node i in the latent space using the jet color map. Blue indicates probabilities near zero, and red indicates probabilities near the maximum over all i and j. (a) Batch GASSOM with soft winner selection and without topological smoothing. (b) Batch GASSOM with soft winner selection and topological smoothing. (c) Batch GASSOM with hard winner selection and topological smoothing. (d) The least square fit of equation 3.2 to the probabilities in panel c.

a_{ij} to be concentrated around $j = i$. We can approximate this by modifying the discrete delta function distribution in equation 2.11 to a discrete gaussian-like distribution centered at i:

$$a_{ij} = \frac{\rho}{S} + (1 - \rho) \frac{g(\mathbf{l}_j | \mathbf{l}_i, \sigma_{Tr}^2)}{\sum_k g(\mathbf{l}_k | \mathbf{l}_i, \sigma_{Tr}^2)}. \tag{3.2}$$



(a) (b)

Figure 5: Histograms of learned transition probabilities from Figure 4a. Note the difference in horizontal scale for the self-transition probability (a) and the transition probability to other nodes (b).

A least squares fit of equation 3.2 to the learned transition probabilities for five trials gives $\sigma = 1.34 \pm 0.02$ and $\rho = 0.40 \pm 0.02$. Figure 4c shows that similar transition probabilities are learned with hard winner selection, where the WTA operation is added to equation 2.34. A least squares fit of figure 3.2 over five trials gives $\sigma = 1.29 \pm 0.02$ and $\rho = 0.41 \pm 0.01$. Figure 4d shows the fit of equation 3.2 to the learned transition probabilities in Figure 4c.

The low standard deviations of the fitted parameters and the consistency of the fitted parameters with both hard and soft winner selection indicate that the emergence of slowness in the GASSOM is a robust phenomenon. Thus, for convenience in most of the remaining experiments, we will simply set the transition probabilities to equation 3.2 with $\sigma = 1.25$ and $\rho = 0.4$ and the emission probability distribution parameters to $\sigma_N = 0.08$ and $\sigma_W = 0.4$.

To illustrate further that the transition probabilities in equation 3.2 encode slowness into the GASSOM, Figure 6 shows the distribution of the Euclidian distances between temporally adjacent winning nodes for the online GASSOM with hard winner selection and topological smoothing (equation 2.33) using transition probabilities fixed by equation 3.2. Across saccades, which are analogous to boundaries between episodes, winning nodes are more widely separated. During fixations when the gaze location drifts slowly, transitions tend to occur either to the same node or nearby nodes.

3.3 Subspaces Learned by the ASSOM and GASSOM. As reported previously (Kohonen, 1996), the basis vectors learned by the original ASSOM algorithm, shown in Figure 7a, are Gabor-like in structure. Figure 7 show only one basis vector from each subspace since the other basis vector

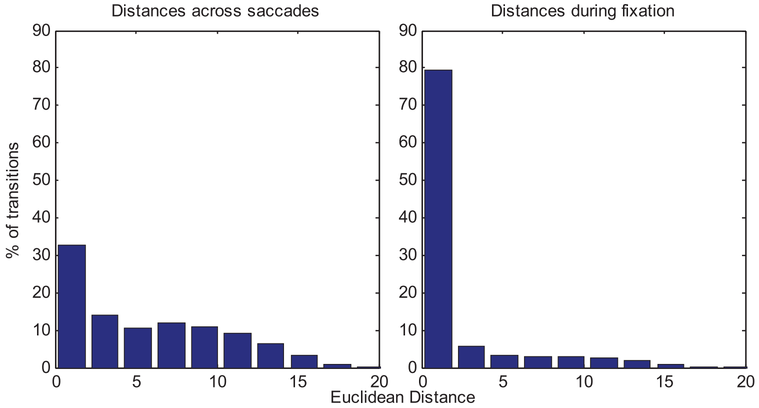


Figure 6: Histograms of the distances between temporally adjacent winners across saccades and during saccades.

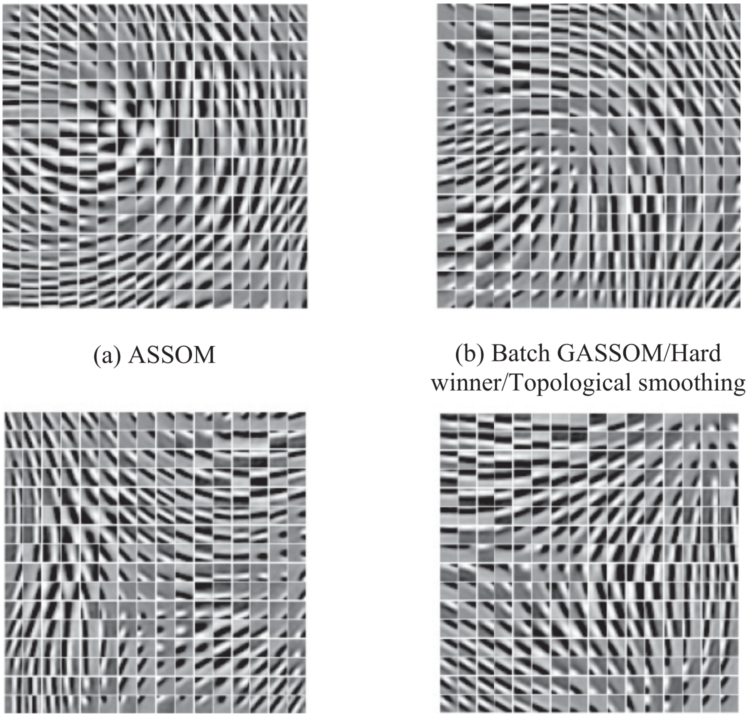
in the subspace is usually similar, except for a phase shift, as discussed in more detail below.

In order to characterize the statistical distribution of the basis vectors learned, we fit each basis vector from five independent trials with a two-dimensional Gabor function, and extracted the wavelength and orientation parameters from the fits. Figure 8a shows that the distribution of orientations is fairly uniform with a slight overrepresentation of the cardinal directions. Figure 9a shows a diversity of wavelengths, with most falling around 6 to 9 pixels.

Figures 7, 8, and 9b show that the batch GASSOM with hard winner selection and topographical smoothing, which we argued in section 2.2.2 was most similar to the ASSOM, learns subspaces with basis vectors with very similar properties to the ASSOM algorithm. This demonstrates that the GASSOM, which eliminates the disadvantages from segmenting the training data into explicit training episodes, can be used in place of the ASSOM.

All variants of the GASSOM subspace learning rule with topological smoothing that we considered result in basis vectors with properties similar to that of the ASSOM algorithm. By comparing Figures 7, 8, and 9c with Figures 7, 8, and 9b, we see little difference between hard and soft winner selection. By comparing Figures 7, 8, and 9d with Figures 7, 8, and 9c, we see little difference between batch and online versions of the GASSOM.

To show the relationship between the two basis vectors of each subspace, Figure 10a plots half of the basis vector pairs learned by the online GASSOM with soft winner selection and topographical smoothing. We see that the two basis vectors are usually both Gabor-like and with similar wavelength and orientation, but they differ in phase. To illustrate, Figures 10b and 10c



(a) ASSOM

(b) Batch GASSOM/Hard winner/Topological smoothing

(c) Batch GASSOM/Soft winner/Topological smoothing

(d) Online GASSOM/Soft winner/Topological smoothing

Figure 7: Basis vectors learned by the ASSOM algorithm and three GASSOM variants. One basis vector from each subspace is shown as a 10×10 sub image. The basis vectors are arranged in the 16×16 topology of the latent space. All basis vectors were learned with fixed transition probabilities and emission probability parameters.

show magnified views of the two basis vectors from two of the subspaces from Figure 10a, as well as their cross-sections in the direction perpendicular to the preferred orientation. We can observe that the two basis vectors in a subspace are in approximate phase quadrature.

To quantify the statistical distribution of the phase relationships between the basis vectors of the subspaces in the entire population, we fit the two basis vectors of each subspace by common Gabor functions that shared the same frequency, orientation, and bandwidth parameters but had independent phase parameters. We define a subspace to be a good fit to the common Gabor functions if the squared error of the fit was smaller than half the squared length of the two basis vectors. Figure 11 shows that for the

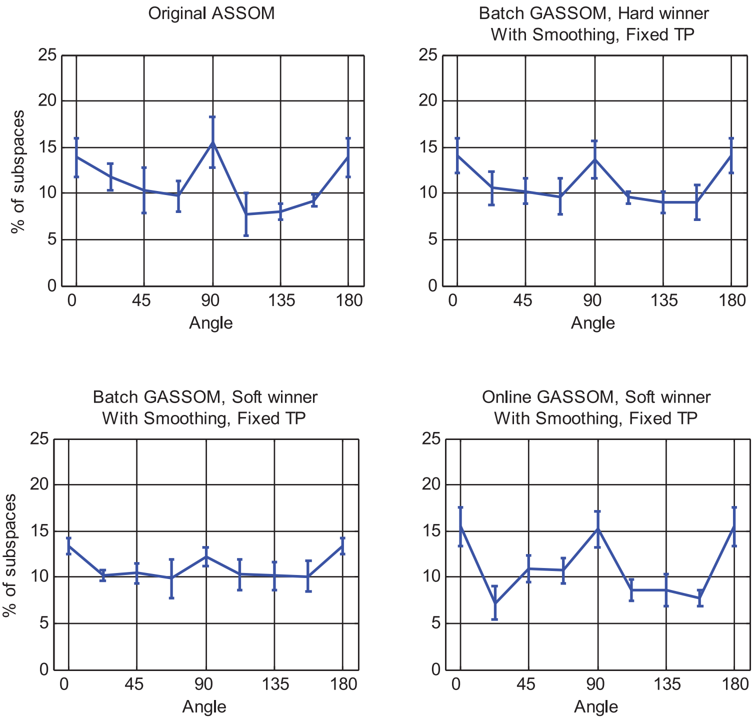


Figure 8: Histograms of the spatial orientations of Gabor fits to the basis vectors in the subspaces learned by the original ASSOM and three GASSOM variants. Error bars indicate the standard deviations of the histogram values computed over five experiments.

ASSOM and the three GASSOM variants, most subspaces were good fits to the common Gabor and that phase differences are clustered around 90 degrees. This is consistent with the phase quadrature Gabor-like receptive fields used to model complex cells in the visual cortex.

3.4 Topological Arrangement of Basis Vectors. Figure 12a shows that the properties of the subspaces learned by the GASSOM with topographical smoothing and learned transition probabilities vary smoothly in the 2D latent space. On the other hand, Figure 12c shows that the subspaces learned by the GASSOM without topographical smoothing and learned transition probabilities do not show this smooth variation. This is expected, since the topographical smoothing applied to the basis vector updates ensures that neighboring subspaces are updated in similar directions.

As discussed in section 2.2.4, the learned transition probabilities of the GASSOM with topological smoothing mirror the topological arrangement

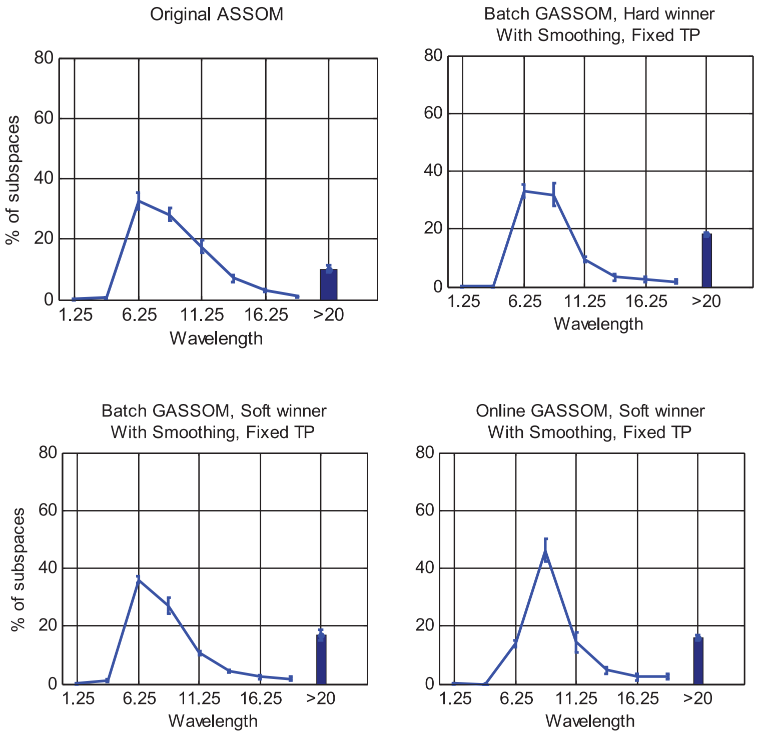


Figure 9: Histograms of the distribution of the wavelengths of the basis vectors. Error bars indicate the standard deviations of the histogram values computed over five experiments.

of the basis vectors. As shown in Figure 4b, the transition probabilities a_{ij} learned by the GASSOM with topological smoothing display a similar smooth variation, with large values concentrated around $j = i$, since nodes neighboring the node generating the current observation are likely to generate the next observation because their subspaces are similar.

Although the transition probabilities learned by the GASSOM without topological smoothing do not have a clear topological arrangement, their structure still reflects similarity between the subspaces in the network. As shown in Figure 4a, in addition to the large self-transition probability a_{ii}, a small number of the large transition probabilities to nodes are scattered apparently randomly in the network. It turns out that these larger transition probabilities connect node i to nodes j with similar subspaces.

To show this, we divide pairs of different nodes into two groups, pairs with high transition interactions and pairs with low transition interactions, by thresholding the symmetric interaction matrix $S(i,j)$ computed from the

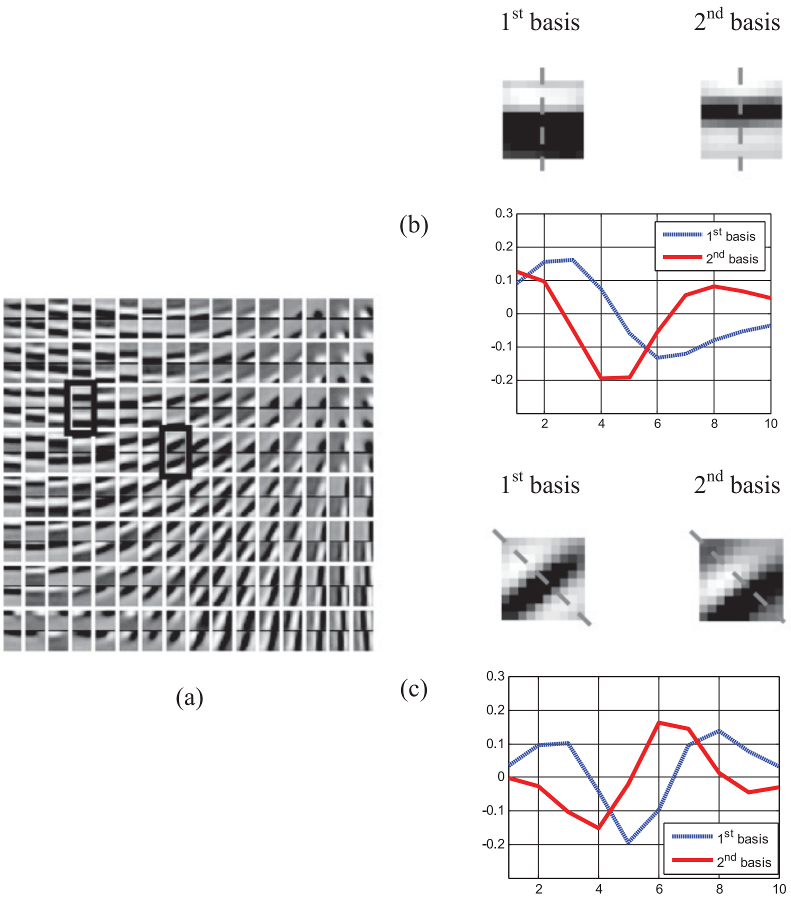


Figure 10: (a) Pairs of basis vectors from the top 8 × 16 array of subspaces learned by the online GASSOM with soft winner selection, topological smoothing, and fixed transition probabilities. Each pair of basis vectors is shown as a 20 × 10 pixel subimage where the top half corresponds to one basis function and the bottom to the other. (b, c) Magnified and cross-sectional views of the basis vector pairs highlighted in panel a.

learned transition probabilities,

$$S(i, j) = \frac{a_{ij} + a_{ji}}{2}, \tag{3.3}$$

at a threshold value of 1×10^{-3}. For each pair, we calculate the maximum principal (canonical) angle between their subspaces. The maximum

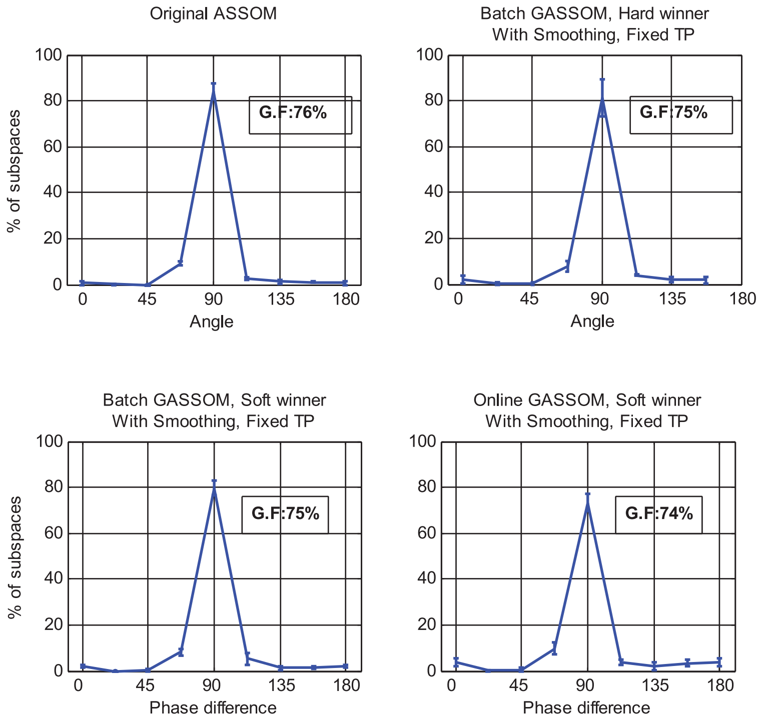
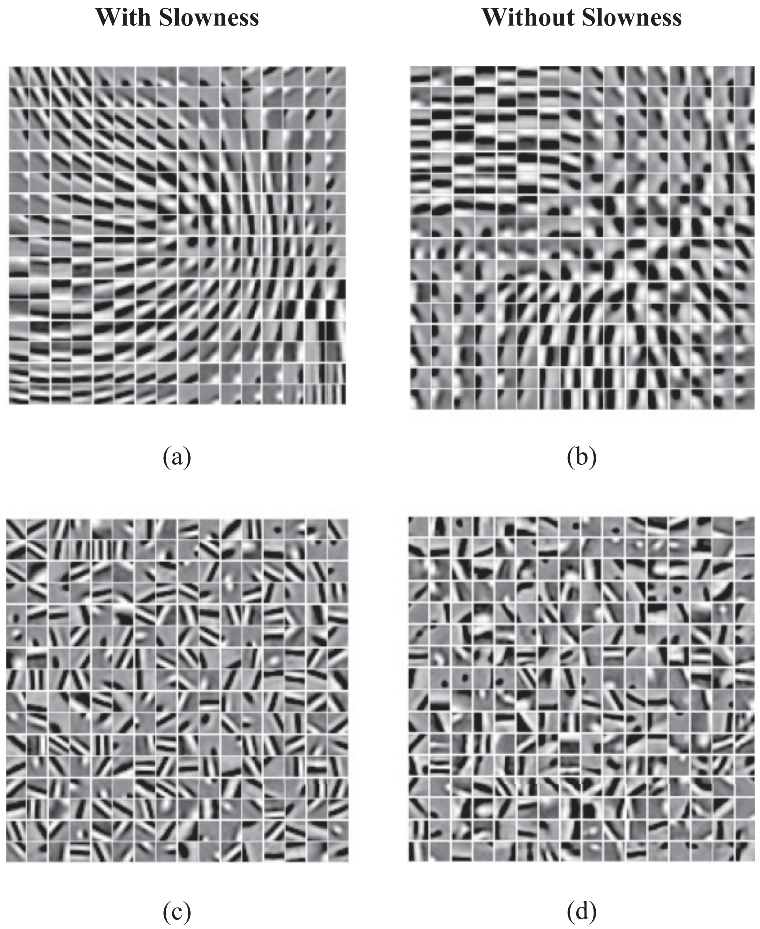


Figure 11: Histograms of the distribution of phase differences between the basis vectors in each subspace for ASSOM and three GASSOM variants. The percentage of good fits (G.F.) is given in each graph. Error bars indicate the standard deviations of the histogram values computed over five experiments.

principal angle is a measure of similarity between two subspaces, which ranges from 0, indicating the subspaces are the same to $\pi/2$, indicating that they are orthogonal.

Figure 13a highlights the nodes that have high transition interaction with the node in the upper-left corner. The subspaces of these nodes have similar orientation and spatial frequency as the basis function of the upper-left-hand-corner node. The histograms in Figure 13b show that pairs of nodes in the high interaction group generally have more similar subspaces (lower maximum principal angles) than the low interaction group.

Despite the differences in their topological arrangement, the subspaces in Figures 12a and 12c exhibit similar properties. Figure 14a shows the distributions of the difference between the spatial orientations of independent Gabor fits to the basis vectors in each subspace. The left two figures show that in both cases, nearly all of the subspaces (93% and 93%) have basis vectors with similar orientations (difference less than 22.5°). The two

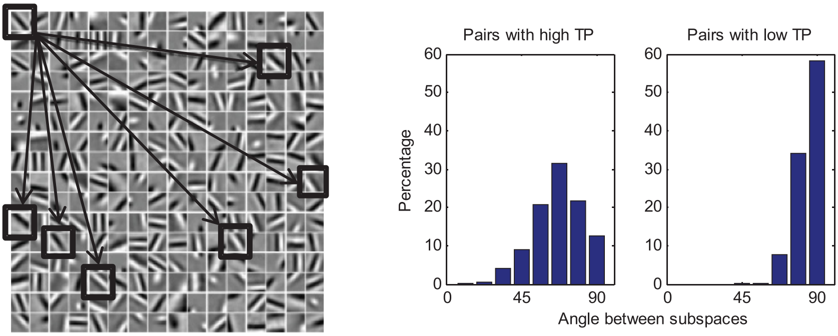


With Slowness

Without Slowness

Figure 12: Basis vectors learned with and without topological smoothing and slowness. Each plot shows one of the pair of basis vector for each subspace learned by (a) batch GASSOM with soft winner selection and topographical smoothing where transition probabilities were learned, (b) batch GASSOM with soft winner selection and topographical smoothing where transition probabilities were fixed to be uniform, (c) batch GASSOM with soft winner selection and without topographical smoothing where transition probabilities were learned, and (d) K-subspaces.

distributions have nearly the same entropy (0.41 and 0.36 bits). Figure 14b shows the distribution of phase differences, which were extracted using the common Gabor fit described previously. The left two figures show in both cases, that the vast majority of subspaces (84% and 81%) have basis vectors



(a)

(b)

Figure 13: (a) The basis functions shown were learned by the batch GASSOM with soft winner selection and without topological smoothing. Basis functions of nodes with high transition interactions with the node in the upper-left-hand-corner are outlined in black. (b) Histograms of the maximum principal angle between pairs of subspaces in the groups with high transition interactions (left) and low transition interactions (right).

in phase quadrature (phase difference within the range 90° ± 11.25°). The two distributions have nearly the same entropy (1.02 and 0.96 bits).

3.5 Removing Slowness Reduces Invariance. We can prevent the emergence of slowness in the model by fixing the transition probabilities to be uniform, that is, by setting $\rho = 1$ in equation 2.11. In this case, the sequence of nodes responsible for the input is assumed to be generated by an i.i.d. process.

Removing slowness does not significantly affect the spatial localization, the spatial frequency tuning, or the orientation tuning of the individual basis vectors. Figure 12b shows basis vectors learned by the batch GASSOM algorithm with soft winner selection and topological smoothing where the transition probabilities were uniform. We observe Gabor-like basis vectors with similar degrees of localization and a similar distribution of orientations as shown in Figure 12a showing basis vectors learned by the same GASSOM algorithm but with transition probabilities that encode slowness.

However, removing slowness reduces the shift invariance of the resulting subspaces. First, the two basis vectors within the learned subspaces are less likely to share the same orientation. Comparing the top two figures of Figure 14a, we observe that without slowness, a significantly smaller percentage of subspaces (32% versus 93%) has basis vectors with similar orientations (difference less than 22.5°). The reduced similarity is also reflected by the higher entropy of the distribution for the GASSOM without slowness: 1.81 bits versus 0.41 bits.

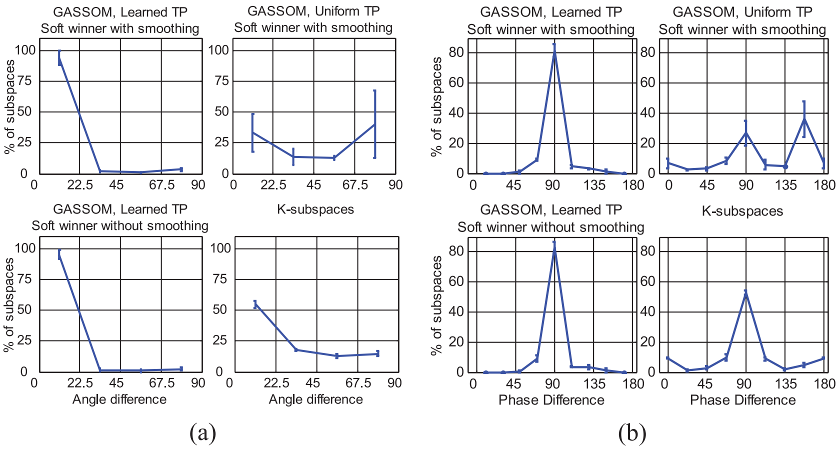


Figure 14: Histograms of (a) the differences between the spatial orientations of independent Gabor fits to the two basis vectors in each subspace and (b) phase differences between the two basis vectors in each subspace extracted by the common Gabor fit.

Second, the basis vectors are less likely to be in approximate phase quadrature. Since the basis vectors were less likely to share a common orientation, fewer of the subspaces from the GASSOM without slowness were good fits to the common Gabor (63%) than from the GASSOM with slowness (75%). In addition, even among the subspaces that were good fits to the common Gabor, fewer of the basis vector pairs learned by the GASSOM without slowness (28% versus 78%) were in phase quadrature (phase difference within the range $90^\circ \pm 11.25^\circ$). The reduced clustering around 90°is also reflected by the higher entropy of the distribution for the GASSOM without slowness: 2.57 bits versus 1.12 bits.

3.6 Removing Both Slowness and Topology. If we remove both slowness and topological smoothing from the batch GASSOM with hard winner selection, we obtain an algorithm similar to the *K*-subspaces algorithm (Vidal, 2011). The K-subspaces algorithm is an extension of the K-means algorithm that finds a set of subspaces, rather than vectors, to best capture the statistics of the input data. The algorithm is iterative, with iteration consisting of two steps. In the first step, each data point is assigned to the subspace with minimum projection error. In the second step, the subspaces adapt to best fit their assigned data points. Data are presented to the system in batches that contain 2000 saccades.

As expected, Figure 12d shows that the resulting subspaces have no topological arrangement. Consistent with our results with removing slowness, only 45% of the subspaces were good fits to the common Gabor functions. Figures 14a and 14b show that the fewer basis vectors learned by K-subspaces have similar orientations (54%) or are in phase quadrature (55%). The entropies of the distributions (1.74 and 2.27) were also similar to the GASSOM without slowness.

Independent subspace analysis (ISA) (Hyvärinen & Hoyer, 2000) is another algorithm that can learn localized orientation-selective receptive fields from natural image data. Like K subspaces, this model includes no assumptions about temporal slowness or topological arrangement. The resulting subspaces show similar properties.

We used the ISA implementation available from Hoyer and Hyvärinen (2000), to learn subspaces on image sequences from the eye movement model. Unless otherwise specified, default parameters were used. We presented data in batches of 2000 frames. The subspace dimensionality was set to two. PCA was used to whiten the data and reduce dimensionality to 80. The number of subspaces learned was 36. We used independent Gabor fits to extract orientation information and common Gabor fits to extract phase differences between basis vectors in a subspace.

Similar to *K* subspaces, only 58% of the subspaces were good fits to the common Gabor. Although many (78%) of the subspaces had basis vectors with similar orientations, very few (17%) were in phase-quadrature. Therefore, the subspaces from ISA did not show the same degree of invariance as those from GASSOM with slowness.

3.7 Comparison of Shift Invariance. The image patches presented to the algorithm during each fixation are slightly shifted versions of each other due to the drift process. Thus, we expect the responses of the GASSOM networks to exhibit shift invariance, where we define the response of the network to an input patch as the set of squared projection lengths onto the subspaces as defined in equation 2.1.

To measure shift invariance, we calculated the root mean square difference (RMSD) between the GASSOM responses to an image patch and a shifted version of the image patch, averaged over 5000 test image patches. To enable comparison across different networks, we normalize the RMSD by the value obtained at the shift of 10 pixels, where there is no overlap between the image patches. Figure 15 plots the RMSD as a function of the image shift for the basis vectors obtained through the four algorithms presented in Figure 12. Better invariance corresponds to a lower RMSD at the same image shift.

The subspaces learned by the algorithms that include slowness display better shift invariance. This is consistent with our results showing that these algorithms lead to basis vectors within each subspace that are more closely matched in orientation and are more likely to be in approximate phase quadrature. Removing topological smoothing has no significant impact on the shift invariance of the representations.

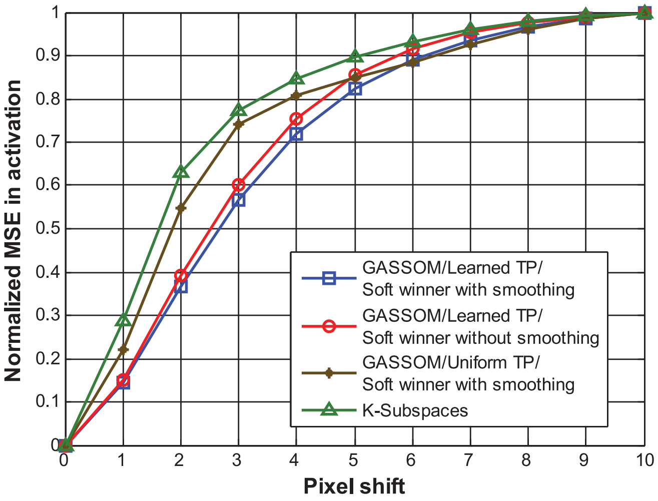


Figure 15: Normalized MSE between the responses of subspaces to input patches with different pixel shifts learned by GASSOM algorithms with and without topological smoothing or slowness and the K-subspaces algorithm.

4 Discussion

The GASSOM is an ASSOM based on a generative model of input sequences. The map consists of a set of topologically organized nodes. Each input vector is assumed to be generated by one of the nodes as the sum of a gaussian distributed vector lying within a low-dimensional subspace associated with the node and a small gaussian noise vector orthogonal to that subspace. Transitions between generating nodes evolve according to a first-order Markov chain.

Each node in the network can be thought of as an invariant feature detector, where the squared length of the projection of the input onto the corresponding subspace can be interpreted as the strength of the feature. The computation of this squared length is similar to the computations used in energy models of complex cell responses in the primary visual cortex. The basis vectors defining the subspaces are analogous to the receptive fields of the orientation-tuned simple cells whose responses are combined to achieve phase-invariant complex cell responses. The learned basis vectors exhibit many properties of the receptive fields used in models of visual cortical cells. They have spatially localized and orientation-tuned Gabor-like shapes. In addition, the two basis vectors of the subspace have the same preferred orientation and are in phase quadrature. This suggests that complex cells in visual cortex play a critical role in extracting a stable representation of the environment from retinal input, which is highly variable due to eye movements, which are present even during fixation.

This generative model has some similarity with the mixture of probabilistic PCA (MPPCA) algorithm (Vidal, 2011). MPPCA assumes that the data are generated from a mixture of subspaces with a multivariate gaussian noise. There are two key differences between the GASSOM model and MPPCA. First, the generating subspaces of the GASSOM are assumed to evolve according to a Markov process rather than according to an i.i.d. process. This difference enables the GASSOM to capture the concept of slowness in the transition probabilities in the Markov chain. Second, the GASSOM model assumes the additive noise lies orthogonal to the generating subspace rather than being an isotropic N-dimensional random vector. This is a relatively minor change, but it dramatically simplifies the computation of the posterior means of the latent variables $\mathbf{w}(t)$ and $\mathbf{n}(t)$ given the observations $\mathbf{x}(t)$. It also enables us to obtain the original ASSOM model as a limiting case of the GASSOM. The GASSOM also has some similarity with the generative topographic mapping (GTM) (Bishop, Svensén, & Williams, 1998). However, the GTM does not include any concept of temporal continuity or slowness.

Our results are consistent with the generative model of image sequences presented in Hurri and Hyvärinen (2003), which learns an autoregressive model of the temporal dependencies between the activity of model simple cells. Similar to our finding of stronger transition probabilities between similar subspaces, the authors found that the model learned stronger connections between cells with similar preferred orientations, frequencies, and locations, However, since the model considered only simple cells, the roles of sparsity and slowness in the learning of invariance were not addressed.

There are several important findings in our study. First, this work demonstrates that slowness does not need to be externally imposed as an objective to be maximized; rather, it can be viewed as a property that emerges as the GASSOM seeks a good statistical representation of its input. In particular, we have shown that slowness emerges when the model is presented with image sequences generated by a realistic model of human eye movements, suggesting that slowness emerges in response to input sequences typically acquired by biological visual systems.

In the GASSOM, slowness is encoded as particular forms of the transition probabilities. We have proposed an algorithm for learning these transition probabilities. Transition probabilities are initialized to discrete uniform distribution with small fluctuations to break symmetry. Since the uniform distribution has maximum entropy, the principle of maximum entropy implies that this initialization makes the fewest prior assumptions. Initializing the transition probabilities as uniform implies that the generating nodes follow an i.i.d. process, which is intuitively far from a slow process. The robust emergence of transition probabilities encoding slowness in our experiments suggests that this is a robust phenomenon in response to the properties of the input and not due to hidden prior assumptions.

Second, this work demonstrates that models that incorporate slowness result in representations that exhibit better invariance. Prior work comparing the effects of sparsity and slowness was unable to identify any advantage to incorporating slowness (Lies et al., 2014). We inhibited the emergence of slowness in the GASSOM by fixing the transition probabilities to be uniform. Although we find that the individual basis vectors learned without slowness still have spatially localized and orientation tuned Gabor-like shapes, we find that the pairs of basis vectors defining the two-dimensional subspaces are less likely to approximate phase quadrature pairs either because their spatial orientation tunings differ or the phase difference between them is not close to 90 degrees. This reduces the invariance of the representation to position shifts.

Third, this work suggests that a smoothly varying topographical organization of orientation-selective nodes is not critical to the development of invariant feature detectors. Our results indicate that subspaces developed by the GASSOM both with and without topographical smoothing display similar degrees of shift invariance. This is consistent with findings that the prevalence of orientation preference maps (OPMs) is not universal among mammals. While cats, primates, and ferrets have smoothly varying orientation maps, several rodent species, such as mice, rats, and gray squirrel, have orientation-selective neurons but lack topological arrangement (Bonin, Histed, Yurgenson, & Reid, 2011; Espinosa & Stryker, 2012). Although OPMs have raised significant attention among theoretical neuroscientists (Bonhoeffer & Grinvald, 1991; Ohki et al., 2006), their functionality and the mechanisms underlying their emergence are still not clear (Kaschube, 2014).

Nonetheless, in both cases with and without maps, compelling evidence suggests that cortical neurons with similar orientation selectivity have a higher degree of connectivity with each other. This is consistent with our results showing that the learned transition probabilities linking nodes with similar orientations are higher than the learned transition probabilities linking nodes with differing orientations in both cases. If the transition probabilities are indeed analogous to lateral interconnections between neurons, the clustering of large transition probabilities to neighboring nodes shown in Figures 4b and 4c suggests that one reason for orientation maps may be to minimize wiring length (Koulakov & Chklovskii, 2001). Since transition probabilities encode slowness in the GASSOM, this analogy also suggests that lateral interconnections may serve as the neural mechanism implementing slowness.

Finally, this work has demonstrated that many possible learning rules result in invariant feature detectors. These vary depending whether they operate in online or batch mode, whether they use hard or soft winner selection, and whether they use topological smoothing. Table 1 summarizes

Table 1: Effect of Different Settings of the GASSOM Algorithm on the Learning of Invariant Subspaces.

Algorithm	Batch/Online Updates	Hard/Soft Winner	Topological Smoothing	Transition Probabilities	Similar Orientation	Good Fit to Common Gabor	Phase Quadrature
ASSOM 7(a)	NA	H	✓	NA	93%	76%	82%
GASSOM 4(a), 12(c), 13(a)	B	S		L	93%	78%	84%
GASSOM 4(b), 12(a),	B	S	✓	L	93%	74%	81%
GASSOM 4(c)	B	H	✓	L	93%	75%	82%
GASSOM 7(b)	B	H	✓	S	94%	74%	78%
GASSOM 7(c)	B	S	✓	S	93%	75%	80%
GASSOM 7(d), 10	O	S	✓	S	92%	74%	78%
GASSOM 12(b)	B	S	✓	U	50%	61%	50%
K-subspaces 12(d)	B	H		U	54%	45%	55%
Independent subspace analysis	B	S		U	60%	58%	17%

Notes: The first column indicates the algorithm and which figure(s) the learned transition probabilities or basis functions are shown in. The next four columns indicate the properties possessed by the different algorithms. In the fourth column, L indicates that the transition probabilities were learned, S indicates that they were fixed to encode slowness, and U indicates that they were fixed to be uniform. The fifth column gives the percentage of the subspaces whose basis vector orientations were well fitted by Gabor functions differed by less than 22.5°. The sixth column gives the percentage of subspaces whose basis vectors were well fit by a common Gabor. The final column gives the percentage of subspaces among those that were good fits to the common Gabor whose fitted basis vectors had phase differences in the range 90° ± 11.25°.

the results of the algorithms studied in this letter. All algorithms, except those without slowness (uniform transition probabilities), had comparable performance. The emergence of invariant feature detectors with similar properties in all of these cases suggests that the emergence of invariance is a robust phenomenon in the GASSOM model.

Acknowledgments

This work was supported in part by the Hong Kong Research Grants council through the Hong Kong PhD Fellowship Scheme (T.N.) and through grant # 619111.

References

Barlow, H. B. (1961). Possible principles underlying the transformation of sensory messages. In W. A. Rosenblith (Ed.), *Sensory communication*, (pp. 217–234). Cambridge, MA: MIT Press.

Bell, A. J., & Sejnowski, T. J. (1997). The "independent components" of natural scenes are edge filters. *Vision Research*, *37*(23), 3327–3338.

Berkes, P., & Wiskott, L. (2005). Slow feature analysis yields a rich repertoire of complex cell properties. *Journal of Vision*, *5*(6), 9.

Bishop, C. M., Svensén, M., & Williams, C. K. (1998). GTM: The generative topographic mapping. *Neural Computation*, *10*(1), 215–234.

Bonhoeffer, T., & Grinvald, A. (1991). ISO-orientation domains in cat visual cortex are arranged in pinwheel-like patterns. *Nature*, *353*(6343), 429–431.

Bonin, V., Histed, M. H., Yurgenson, S., & Reid, R. C. (2011). Local diversity and fine-scale organization of receptive fields in mouse visual cortex. *Journal of Neuroscience*, *31*(50), 18506–18521.

Chandrapala, T. N., & Shi, B. E. (2014). The generative adaptive subspace self-organizing map. In *Proceedings of the IEEE International Joint Conference on Neural Networks* (pp. 3790–3797). Piscataway, NJ: IEEE.

Espinosa, J. S., & Stryker, M. P. (2012). Development and plasticity of the primary visual cortex. *Neuron*, *75*(2), 230–249.

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

Hasenstaub, A., Otte, S., Callaway, E., & Sejnowski, T. J. (2010). Metabolic cost as a unifying principle governing neuronal biophysics. *Proceedings of the National Academy of Sciences of the United States of America*, *107*(27), 12329–12334.

Hoyer, P., & Hyvärinen, A. (2000). *Independent component analysis and its extensions as models of natural image statistics*. http://research.ics.aalto.fi/ica/imageica/

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

Hubel, D. H., & Wiesel, T. N. (1968). Receptive fields and functional architecture of monkey striate cortex. *Journal of Physiology*, *195*(1), 215–243.

Hurri, J., & Hyvärinen, A. (2003). Temporal and spatiotemporal coherence in simple-cell responses: A generative model of natural image sequences. *Network: Computation in Neural Systems*, *14*(3), 527–551.

Hyvärinen, A., & Hoyer, P. (2000). Emergence of phase-and shift-invariant features by decomposition of natural images into independent feature subspaces. *Neural Computation*, *12*(7), 1705–1720.

Hyvärinen, A., & Hoyer, P. O. (2001). A two-layer sparse coding model learns simple and complex cell receptive fields and topography from natural images. *Vision Research*, *41*(18), 2413–2423.

Hyvarinen, A., Hurri, J., & Hoyer, P. O. (2009). *Natural image statistics*. New York: Springer.

Hyvarinen, A., Oja, E., Hoyer, P., & Hurri, J. (1998). Image feature extraction by sparse coding and independent component analysis. In *Proceedings of the International Conference on Pattern Recognition* (Vol. 2, pp. 1268–1273). Piscataway, NJ: IEEE.

Kaschube, M. (2014). Neural maps versus salt-and-pepper organization in visual cortex. *Current Opinion in Neurobiology*, *24*, 95–102.

Kavukcuoglu, K., Ranzato, M., Fergus, R., & Le Cun, Y. (2009). Learning invariant features through topographic filter maps. In *Proceedings of the IEEE Conference in Computer Vision and Pattern Recognition* (pp. 1605–1612). Piscataway, NJ: IEEE.

Kohonen, T. (1990). The self-organizing map. *Proceedings of the IEEE*, *78*(9), 1464–1480.

Kohonen, T. (1996). Emergence of invariant-feature detectors in the adaptive-subspace self-organizing map. *Biological Cybernetics*, *75*(4), 281–291.

Kohonen, T., Kaski, S., & Lappalainen, H. (1997). Self-organized formation of various invariant-feature filters in the adaptive-subspace SOM. *Neural Computation*, *9*(6), 1321–1344.

Kohonen, T., Kaski, S., Lappalainen, H., & Saljärvi, J. (1997). The adaptive-subspace self organizing map (ASSOM). In *Proceedings of the International Workshop on Self-Organizing Maps* (pp. 191–196).

Koulakov, A. A., & Chklovskii, D. B. (2001). Orientation preference patterns in mammalian visual cortex: A wire length minimization approach. *Neuron*, *29*(2), 519–527.

Kuang, X., Poletti, M., Victor, J. D., & Rucci, M. (2012). Temporal encoding of spatial information during active visual fixation. *Current Biology*, *22*(6), 510–514.

Li, N., & DiCarlo, J. J. (2008). Unsupervised natural experience rapidly alters invariant object representation in visual cortex. *Science*, *321*(5895), 1502–1507.

Lies, J., Häfner, R. M., & Bethge, M. (2014). Slowness and sparseness have diverging effects on complex cell learning. *PLoS Computational Biology*, *10*(3), e1003468.

Ohki, K., Chung, S., Kara, P., Hübener, M., Bonhoeffer, T., & Reid, R. C. (2006). Highly ordered arrangement of single neurons in orientation pinwheels. *Nature*, *442*(7105), 925–928.

Olshausen, B. A. (1996). Emergence of simple-cell receptive field properties by learning a sparse code for natural images. *Nature*, *381*(6583), 607–609.

Olshausen, B. A., & Field, D. J. (1997). Sparse coding with an overcomplete basis set: A strategy employed by VI? *Vision Research*, *37*(23), 3311–3326.

Perrett, D., Rolls, E., & Caan, W. (1982). Visual neurones responsive to faces in the monkey temporal cortex. *Experimental Brain Research*, *47*(3), 329–342.

Rabiner, L. R. (1989). A tutorial on hidden Markov models and selected applications in speech recognition. *Proceedings of the IEEE*, *77*(2), 257–286.

Rabiner, L., & Juang, B. (1986). An introduction to hidden Markov models. *IEEE ASSP Magazine*, *3*(1), 4–16.

Rolfs, M. (2009). Microsaccades: Small steps on a long way. *Vision Research*, *49*(20), 2415–2441.

van Hateren, J. H., & van der Schaaf, A. (1998). Independent component filters of natural images compared with simple cells in primary visual cortex. *Proceedings of the Royal Society of London, series B: Biological Sciences*, *265*(1394), 359–366.

Vidal, R. (2011). Subspace clustering. *IEEE Signal Processing Magazine*, *28*(2), 52–68.

Wiskott, L., & Sejnowski, T. J. (2002). Slow feature analysis: Unsupervised learning of invariances. *Neural Computation*, *14*(4), 715–770.

Zheng, H., Lefebvre, G., & Laurent, C. (2008). Fast-learning adaptive-subspace self-organizing map: An application to saliency-based invariant image feature construction. *IEEE Transactions on Neural Networks*, *19*(5), 746–757.

Received August 4, 2014; accepted February 28, 2015.