*LETTER* **Communicated by Ko Sakai**

# Learning Invariant Features in Modulatory Networks through Conflict and Ambiguity

**W. Shane Grant**
*wgrant@usc.edu*
*Department of Computer Science, University of Southern California, Los Angeles, CA 90089, U.S.A.*

**Laurent Itti**
*itti@usc.edu*
*Department of Computer Science and Department of Psychology and Neuroscience Graduate Program, University of Southern California, Los Angeles, CA 90089, U.S.A.*

**This work lays the foundation for a framework of cortical learning based on the idea of a competitive column, which is inspired by the functional organization of neurons in the cortex. A column describes a prototypical organization for neurons that gives rise to an ability to learn scale, rotation, and translation-invariant features. This is empowered by a recently developed learning rule, conflict learning, which enables the network to learn over both driving and modulatory feedforward, feedback, and lateral inputs. The framework is further supported by introducing both a notion of neural ambiguity and an adaptive threshold scheme. Ambiguity, which captures the idea that too many decisions lead to indecision, gives the network a dynamic way to resolve locally ambiguous decisions. The adaptive threshold operates over multiple timescales to regulate neural activity under the varied arrival timings of input in a highly interconnected multilayer network with feedforward and feedback. The competitive column architecture is demonstrated on a large-scale (54,000 neurons and 18 million synapses), invariant model of border ownership. The model is trained on four simple, fixed-scale shapes: two squares, one rectangle, and one symmetric L-shape. Tested on 1899 synthetic shapes of varying scale and complexity, the model correctly assigned border ownership with 74% accuracy. The model's abilities were also illustrated on contours of objects taken from natural images. Combined with conflict learning, the competitive column and ambiguity give a better intuitive understanding of how feedback, modulation, and inhibition may interact in the brain to influence activation and learning.**

*Neural Computation* **31, 344–387** (2019)
doi:10.1162/neco_a_01156

## 1 Introduction

Invariant processing is a hallmark of the visual system, which effortlessly recognizes objects at nearly any combination of scale and viewpoint. Despite the proliferation of biologically inspired work on object recognition, how invariance is achieved remains an unresolved question and a challenging problem for computational models. Scale invariance is especially problematic, as it introduces questions of not only how multiple scales should be represented but how responses to those should dynamically interact together.

Scale is often addressed by using a feature pyramid, where the input itself or processed versions thereof are scaled and fed through the network at multiple scales before the resulting output is integrated into a single response (Dollár, Appel, Belongie, & Perona, 2014; Itti, Koch, & Niebur, 1998; Lowe, 1999). This technique is practical but does not correlate to a directly plausible explanation for how the brain is scale invariant, and it does not allow for much interaction between different scales. In object recognition systems, if scale is considered, it is typically handled by duplicating features at varying resolutions and then performing a pooling or max operation to select a single winner (Serre, Wolf, & Poggio, 2005) in what essentially becomes a scale pyramid over features. Because the pooling operation discards information, this type of approach is also limited in the interaction between varying scales. Modern approaches using deep networks generally rely on some sort of region proposal step, such as a sliding window, an attention system (Cheng et al., 2015) or a network specialized to find candidate regions (Ren, He, Girshick, & Sun, 2015). Other approaches combine region proposal and recognition into a single network (Liu et al., 2016; Redmon, Divvala, Girshick, & Farhadi, 2016), though ultimately all of these approaches resize the selected region to the canonical size of the network, which expects a near-perfect crop.

This work introduces two primary contributions—the competitive column and ambiguity—that resolve some of the complications introduced by multiscale processing. They are demonstrated on a model of border ownership, extending the previous work of Grant, Tanner, and Itti (2017). Border ownership is an early scale-invariant visual process that involves the assignment of object boundaries to objects (Zhou, Friedman, & von der Heydt, 2000). As border ownership responses begin to occur before object recognition takes place (Brincat & Connor, 2006; Williford & von der Heydt, 2016), polarity (or ownership) assignment must be decided in the face of what is often locally ambiguous input. Combined with its putative feedback structure (Craft, Schütze, Niebur, & von der Heydt, 2007), border ownership is a challenging but tractable target for learning.

The first contribution of this work, the competitive column, is a solution for scale invariance that takes inspiration from both scale pyramids as well as the highly interconnected wiring of the visual system (Markov et al., 2014). The resultant architecture is a hierarchical cascade with the addition of connections between every layer of the network, much like the concept of skip connections, which have become popular in deep learning (He, Zhang, Ren, & Sun, 2016) and have previously been referred to as shortcut connections (Bishop, 1995). Using the learning rule developed in the previous work (conflict learning; Grant et al., 2017) the competitive column can be used to learn invariant responses through visual experience alone.

The second contribution of this work is the development of a new notion of ambiguity that dampens the activity of neurons that cannot reach a reliable consensus. The ability to resolve ambiguity is critical for multiscale interaction, which can introduce conflicting responses generated at different scales. In the figure-ground system of Layton, Mingolla, and Yazdanbakhsh (2015), one of the few systems to use scale in a nontrivial fashion, larger-scale responses inhibit smaller-scale responses. In this work, inhibitory feedback will be used in the computation of ambiguity, which can ultimately inhibit the activation of a neuron in a similar fashion.

These primary contributions are supported by an adaptive threshold scheme that addresses many of the complications introduced by a multilayer network where input can arrive at varying times. The threshold operates over multiple timescales in a fashion analogous to the weights in conflict learning.

Together, these contributions are used to learn an invariant model of border ownership that addresses the shortcomings of the model of border ownership in the previous work (Grant et al., 2017) which was restricted to specific scales incapable of handling overly ambiguous input. The competitive column architecture is used to train a new, deeper model on four simple fixed size shapes, which is tested on nearly 2000 procedurally generated shapes of varying complexity and scale. The performance of the model is analyzed as a function of scale, rotation, and translation invariance. It is additionally demonstrated on a sample of contours taken from natural images.

The ability of the network to learn invariant responses through visual experience alone, without an explicit teacher, demonstrates the capabilities of the competitive column architecture, as well as the benefits of ambiguity. The implications of the architecture are discussed as it relates to activation dynamics as well as learning, especially in regard to the concept of proto-objects.

## 2 Background

**2.1 Border Ownership.** As mentioned in section 1, border ownership is an early visual process that assigns borders to owning objects. In terms of neural responses, border ownership (BO) neurons, which are typically edge selective, respond strongly when an object is observed on one side of them and weakly when presented to the nonpreferred side (Zhou et al., 2000).

The computation of border ownership requires information from outside the classical receptive field, suggesting that it necessitates the cooperation of BO neurons spread across an entire figure to ensure a correct polarity assignment. Border ownership is believed to be a critical component of figure-ground segmentation, a process that is itself likely critical for higher-level tasks such as object recognition (Kogo & van Ee, 2014).

The grouping hypothesis (Martin & von der Heydt, 2015) has emerged as a dominant theory for how neurons compute border ownership. Craft et al. (2007) developed a model where pairs of BO neurons compete over a polarity assignment while receiving feedback from higher-level grouping neurons, which pool over a wide range of BO responses. The grouping neurons have annular receptive fields and are essentially tuned to fire when a mostly contiguous arrangement of BO neurons supports the interior of the figure occurring at the retinotopic position of the grouping neuron (see Figures 2A and 2B). Another fundamental concept behind the computation of border ownership is complementary facilitatory and suppressive input to border ownership neurons (Sakai & Nishimura, 2006), which supports push-pull dynamics between competing BO neurons. Subsequent models (Mihalas, Dong, von der Heydt, & Niebur, 2011; Qiu, Sugihara, & von der Heydt, 2007; Russell, Mihalaş, von der Heydt, Niebur, & Etienne-Cummings, 2014) have applied attention to grouping models of border ownership, but relatively little concern has been paid to how such border ownership could be learned. The previous work (Grant et al., 2017) thus developed a learning rule that could be used to learn a model of border ownership, but it lacked a formal network architecture and did not display the scale invariance seen in actual BO neurons.

To address scale invariance, this work develops the competitive column architecture, detailed in section 3, and applies it to learning a model of border ownership. It should be emphasized that the competitive column architecture is not itself a model of border ownership, but rather a framework in which border ownership can be learned. The model of border ownership in Grant et al. (2017) featured grouping neurons at a single scale, which resulted in BO neurons that could not generalize across object sizes regardless of training input. Here, using the competitive column architecture, the hierarchy will be deepened. The interaction between a new layer of neurons, referred to as proto-object neurons, the previously used grouping neurons, and the border ownership neurons lead to a model that can generalize over a wide range of object scales and complexities. Although these neurons are given different names for illustrative purposes, they are all identical within the competitive column architecture; crucially, it is a combination of learning and visual experience that gives rise to border ownership and grouping behaviors of the neurons.

### 2.2 Conflict Learning.

This section provides a brief overview of conflict learning, which is fully detailed in Grant et al. (2017). Conflict learning, which is used as the learning rule throughout this work, is a learning rule for artificial neural networks designed around the complications of modulatory input. Modulatory input signals are those that affect the output of a neuron only when coincidental with driving input, which is input that can directly trigger the firing of a neuron (Brosch & Neumann, 2014). The previous work discusses in detail why modulatory input necessitates a new learning rule and why traditional Hebbian-like associative rules are insufficient.

Conflict learning consists of three components, which work in tandem to differentiate activity among driving and modulatory inputs. In the following, $x_i$ refers to the activation value of a neuron $i$, and $w_{ij}$ the weight between neurons $i$ and $j$:

1. *Spreading*. Neurons are restricted to increasing weight on only those connections that overlap with their existing preferred stimulus. A coefficient, $\kappa_i$, applied to the weight update, is set equal to the maximum activation among a neuron's strongly learned connections,

$$\kappa_i = \max_{j|(w_{ij}(t) > \frac{1}{2} \max\limits_{j} w_{ij}(t))} x_j, \tag{2.1}$$

where strongly learned connections are those whose weight exceeds half the strength of the largest weight among that individual neuron's connections.

2. *Unlearning*. Conflict learning treats inhibition as an error signal indicating that the inhibited neuron has mistakenly strengthened any currently active connections. A neuron competing with its neighbors via inhibition exerts pressure on those neurons to unlearn the connections driving its activation if those neighbors are coincidentally active. In this context unlearning is expressed as a decrease of weight toward some initial nonnegative minimum. The amount of inhibition a neuron receives is used to interpolate between a positive and a negative associative weight update, $\delta_{ij}$,

$$\delta_{ij} = (1 - \text{Inhib}) * \alpha \eta x_i x_j \kappa_i - \text{Inhib} * \beta \eta x_i x_j, \tag{2.2}$$

where $\alpha$ and $\beta$ (set to 1 in all experiments) can be used to control the rate of learning versus unlearning and $\eta$ is the learning rate. Inhib represents the inhibition received and is defined formally in section 3.3.

3. *Short and long-term (SLT)*. Connection weights are adjusted on short-term and long-term timescales. The short-term weight $w_{ij}$ adjusts rapidly to the current stimulus, but decays toward and fluctuates around the more stable, slowly adapting long-term weight $w_{ij}^{\text{ltm}}$. The only visible weight for a neuron is its short-term weight; long-term weights are internal and observed only via their effect on short-term weights. The entire neuron weight update process has four steps:

a. Compute short-term weight updates $\delta_{ij}$.
b. Move long-term weights toward short-term weights, interpolating between the updated short-term weight $w_{ij}(t) + \delta_{ij}$, and the previous long-term weight $w_{ij}^{\text{ltm}}(t)$:

$$w_{ij}^{\text{ltm}}(t+1) = (1 - s_{\text{ltm}})(w_{ij}(t) + \delta_{ij}) + s_{\text{ltm}} w_{ij}^{\text{ltm}}(t). \tag{2.3}$$

c. Move short-term weights toward long-term weights, interpolating between the updated short-term weight and the updated long-term weight $w_{ij}^{\text{ltm}}(t+1)$:

$$w_{ij}(t+1) = (1 - s_{\text{stm}})(w_{ij}(t) + \delta_{ij}) + s_{\text{stm}} w_{ij}^{\text{ltm}}(t+1). \tag{2.4}$$

d. Normalize short- and long-term weights independently,
where $s_{\text{ltm}}$ and $s_{\text{stm}}$ are smoothing factors, and all weight updates are clamped between a nonnegative, nonzero lower bound and an upper bound of 1.

An accumulator of lifetime short-term weight updates is used for computing the smoothing factor $s_{\text{ltm}}$ for the long-term weight update:

$$\text{acc}_{ij}(t+1) = \text{acc}_{ij}(t) + \delta_{ij}. \tag{2.5}$$

The smoothing factor for the long-term update, $s_{\text{ltm}}$, is computed by comparing a neuron's proportion of long-term weight against its proportion of lifetime accumulator value (normalized $w_{ij}^{\text{ltm}}(t)$ versus $\text{acc}_{ij}(t+1)$). When the $w_{ij}^{\text{ltm}}(t)$ update would move the long-term weight proportion toward that of the accumulator, $s_{\text{ltm}}$ is decreased, proportional to the remaining distance between them. In cases where the $w_{ij}^{\text{ltm}}$ update would move the proportion away from the accumulator, $s_{\text{ltm}}$ is increased. This has the effect of decreasing the rate of change of the long-term weight when it diverges too much from the lifetime accumulated value $\text{acc}_{ij}$.

## 3 The Competitive Column Model

The competitive column model is a framework for a prototypical organization of neurons that supports invariant learning. The model has two key components. The first is a column structure and associated wiring that work in concert with conflict learning to use various sources of inhibition as teaching signals and learn features among driving and modulatory inputs. The second is a new activation dynamic called ambiguity that is used to resolve locally ambiguous input, which is supported by novel threshold dynamics that mirror the mechanics of short- and long-term weights in conflict learning.

**3.1 The Competitive Column.** The organization of neurons into columns to support competition has long been a component of more biologically plausible models of the visual system (Fukushima, 1980). In the primary visual cortex, neurons are arranged in perpendicular slabs such that neurons with similar receptive fields but different orientation preferences are adjacent to each other (Blasdel & Salama, 1986; Hubel & Wiesel, 1974). In this sense, the column is more of a conceptual grounding to describe organization.

The column structure developed here, called the competitive column, is a way to organize neurons such that a diversity of features can be learned while maintaining invariance to rotation, translation, and scale. An illustration of the model is shown in Figure 1. Much like the columns that Fukushima (1980) used, neurons here will be organized into a column if they are within some radius of each other. Though the mental model of a column often (and indeed here as well) has neurons stacked in a cylinder, the neurons need not actually be laid out like this.

Competitive columns have winner-take-all dynamics such that only one neuron can be dominant in activation at a time. Neurons that lose out to more active neurons have their activation diminished but not extinguished. This property, which is seen in opposing polarities of actual border ownership responsive cells (Zhou et al., 2000), is essential for the correct operation of conflict learning. Conflict learning utilizes the competition in the column to drive differentiation among the learning of modulatory features. The model developed here bears some resemblance to the selective tuning attention model of Tsotsos et al. (1995) in that both use a hierarchical framework with winner-take-all dynamics. Their model differs in that it uses a backward pass of these dynamics to refine relevant features to the uppermost winning location, pruning other activations away, whereas the model developed here has constant winner-take-all dynamics in every column.

Neurons within columns also have lateral connections that extend outside the column to other nearby neurons, which are organized into other competitive columns. The weight distribution of these lateral connections determines the overall topographic organization of a layer. The competitive column model makes a distinction between driving and modulatory inputs, which are learned as two independent sets of connections. Lateral inhibition is used as the error signal for driving input and within-column inhibition for modulatory input. This allows inhibition received through lateral connections to cause a differentiation over driving input, analogous to how within-column inhibition differentiates modulatory input. The inhibition affects the neuron activation in the same fashion as that received from within the column; neurons can only be inhibited by those that are more active than them. Grant et al. (2017) demonstrated that center-surround lateral connectivity could be used with conflict learning to cause neurons to develop a smooth pinwheel-like configuration analogous to that often seen in mammalian primary visual cortex. Jain, Millin, and Mel (2015)

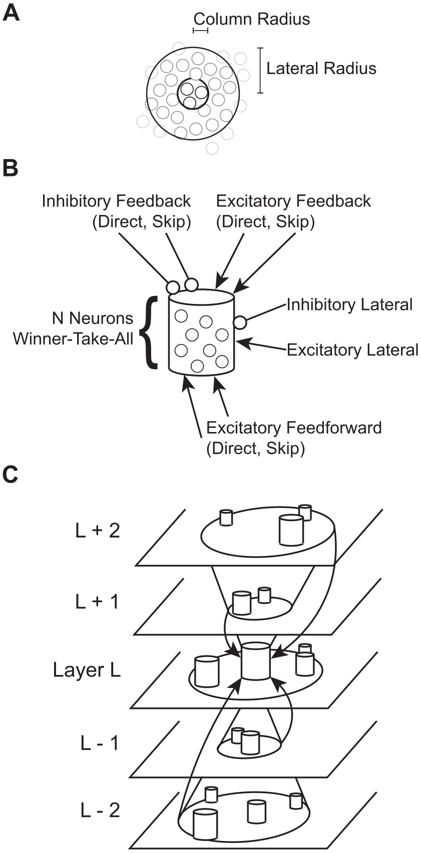

Figure 1: The competitive column and associated network topology. A competitive column is an organization for neurons into local units of competition that receive driving feedforward, modulatory lateral, and modulatory feedback input. (A) Columns are created by wiring together neurons in a local neighborhood with mutual inhibition such that a winner-take-all-like network is created. The winner-take-all dynamics are such that losing neurons are not fully deactivated because of competition. Columns also have lateral connections to other neurons residing in other columns within a larger local neighborhood. (B) A diagram showing the types of input a column receives. Feedforward and feedback input come from both immediately adjacent layers in the hierarchy (direct) as well as distant layers (skip). Feedforward and feedback that trickle through layers in a cascade also ultimately affect the column by influencing the direct input it receives. (C) A diagram showing how the column receives input from other layers in a hierarchy. Input is received from directly adjacent layers, as well as from every distant layer. Input thus follows a pattern of coming from increasingly large receptive fields with increasingly weak weights. Input also arrives from within the same layer in the form of lateral connections.

demonstrated that the structure of lateral connectivity is key to differentiating between these V1-like maps and more differentiated multimaps where a multitude of different features can be learned. Thus, the role of lateral connections in the competitive column model is to drive the overall configuration of the learned feedforward features of a layer.

Columns receive feedforward and feedback input from every other layer in the network. As the layer-to-layer distance increases, so does the receptive field size; neurons deeper in the hierarchy have large receptive fields over earlier levels. This is accompanied by a similar reduction in weight, much like what is seen in real neurons (Markov et al., 2014). This is key to the learning of scale invariance as it allows for differentiation over the influence of input from multiple scales. The network thus combines a typical hierarchical cascade with the ideas of a scale pyramid. The increasing receptive field size of neurons at deeper levels allows the neurons to learn a mixture of scaled-up features along with more "parts-like" features composed of input from intermediary layers.

Since feedback is modulatory and weights decrease with layer-to-layer distance, neurons can exert a great deal of control over their received feedback by inducing neurons in adjacent layers to deactivate. A neuron that receives feedback from the next layer likely also supplies driving input to that layer, and this allows for interesting dynamics to occur between different scales in the network. This is key to the operation of ambiguity developed in section 3.2.

**3.2 Ambiguity.** Border ownership is an example of a visual process that requires the resolution of locally ambiguous information to make a correct decision regarding edge polarities (Kogo & van Ee, 2014). Sometimes this ambiguity is due to a bistable representation of figure and ground, such as the famous Rubin's vase illusion (Rubin, 1915), in which either a vase or two faces can be seen. More often, this ambiguity can arise as a consequence of local features of an object. For example, the neurons that respond to border ownership have limited receptive field sizes yet must make a decision dependent on global context they may not have direct access to. A putative mechanism for the computation of border ownership is briefly reviewed in Figures 2A and 2B.

The problem of resolving ambiguity is not one that can simply be pushed to a deeper level in a hierarchy and solved independently by a larger scale. Eventually there will be a decision between multiple choices where there appear to be several good candidates given the current state of the network. Thus, a useful method of breaking ambiguity should operate at every level of computation, and indeed at every neuron. Further, this issue is not isolated to the problem of border ownership but is a general problem faced when making decisions; there are often many competing choices that cannot be decided on without the influence of some additional factor (consider

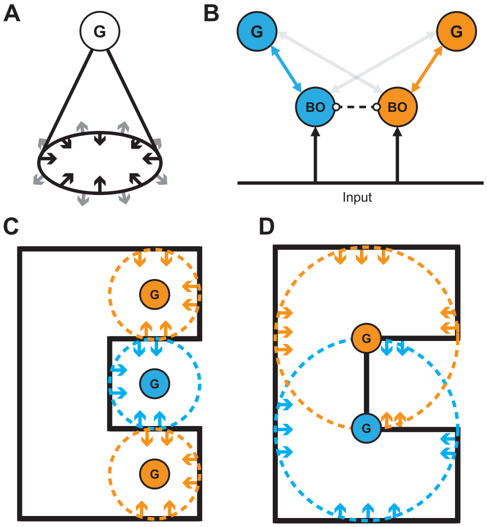

Figure 2: The grouping neuron–based computation of border ownership and an example of a locally ambiguous shape. (A) The output of border ownership (BO) neurons for multiple orientations and polarity assignments are grouped together by a grouping cell, G. The grouping cell reinforces all BO neurons with polarities that support evidence of a closed object located at the grouping cell. Grouping cells receive feedforward input from BO neurons, and BO neurons receive feedback input from grouping cells. Black arrows represent polarities that support the grouping neuron; gray arrows represent polarities inconsistent with that grouping neuron. (B) BO neurons receive identical driving input (e.g., an oriented edge), but their association with a specific grouping cell ties them to a specific polarity (e.g., object left or right). BO neurons responding to the same stimulus compete with each other (dashed line), with feedback from grouping neurons driving differentiation over polarity. (C) A classical locally ambiguous shape for the assignment of border ownership. The top and bottom corners of the c shape have identical local features to the concavity missing from the middle. Each region has three edges making a nearly closed convexity and has the same number of corner features supporting the interior of an object. Each colored grouping cell has a receptive field (dashed circle) from which it receives input from BO neurons with appropriate polarities (like-colored arrows). Note that along the concavity, BO neurons of opposing polarities receive equal amounts of feedback and supply equal amounts of driving input to the grouping neurons. Only a small subset of BO and grouping neurons are drawn for illustrative purposes. (D) The same shape as in panel C with grouping performed over larger receptive fields. Note that the polarity assignment supported by the larger grouping would give an incorrect polarity to the concavity of the c shape. Without explicitly modeling the entire object, the assignment of edge polarities requires a way of quantifying the local ambiguity of decisions and moving the network toward an unambiguous assignment. (Portions of the figure were inspired by Craft et al., 2007.)

relatable situations like deciding on a restaurant to eat at or dealing with conflicting navigation suggestions while driving a car).

A prototypical example of this problem for border ownership is the "c shape," which has a concavity with the same local features as the top and bottom portions of the shape (illustrated in Figures 2C and 2D). Given a feedback model for border ownership, which uses grouping cells to collect local evidence of objectness, strong responses will be elicited within both the concavity and the actual c shape. These will reinforce decisions made at the border ownership neurons that drive those grouping neurons, and ultimately it will not be possible to make a reliable polarity choice along the concavity. Even deepening the hierarchy, as will be done here by the addition of proto-object neurons with larger receptive field sizes, will not break this ambiguity; at a slightly larger scale, this object is still highly symmetric and ambiguous.

The proposed way to solve this problem is to come up with a measure of how ambiguous the activation of a neuron is and then dampen its activity. In essence, the ambiguous neurons will cease to contribute to the state of the network, allowing unambiguous neurons to reach consensus. The border ownership neurons are ambiguous because the neurons for each polarity are receiving strong feedback from grouping cells on opposing sides of the object. However, this alone does not give an individual neuron the ability to decide it is ambiguous. Consider that before these neurons even receive feedback, their common feedforward driving input will put them into a similar state of high activation that is resolved through competition and the contribution of noise to drive an initial winner. It is not until these neurons both become reinforced in their decisions through modulatory feedback that they can be said to be ambiguous.

To give neurons a way to measure this individually, a combination of inhibitory and excitatory feedback is used. Neurons are expected to learn a preferred feedback stimulus (in the case of border ownership, from a single polarity) and a nonpreferred stimulus (from the opposite polarity). Although the model of Craft et al. (2007) already proposed using inhibitory feedback to drive polarity decisions for border ownership, inhibitory feedback alone is insufficient to resolve this problem and unnecessary to compute border ownership in largely unambiguous inputs, as Grant et al. (2017) demonstrated.

It is the balance of excitatory and inhibitory modulatory feedback that provides the best measure of whether a neuron is in an ambiguous state. A neuron receiving a high amount of both types of input is receiving a signal from neurons deeper in the hierarchy that it should be both highly active and highly inhibited. This is precisely what causes an ambiguous assignment of border ownership: high feedback from grouping neurons on competing polarities.

To best capture this description of ambiguity, ambiguity is defined to be the minimum of the modulatory excitation and inhibition. Defined this way, a neuron that receives high amounts of both modulatory excitation and inhibition is considered to be ambiguous, while a neuron that receives a high amount of either excitation or inhibition is unambiguous. In addition, a neuron receiving no modulatory input is also unambiguous. Using a minimum for this operation, a semantic difference can be made between high inibition in isolation and high inhibition coincidental with high excitation.

Ambiguity is detailed mathematically in section 3.3, and a demonstration of border ownership assignment on the c shape with and without a notion of ambiguity can be found in section 5.2.

**3.3 Neuron Activation.** As mentioned in section 3.1, neurons receive many sources of input that can be classified as driving or modulatory. Driving input comes purely from excitatory feedforward connections. Modulatory input comes from either lateral or feedback connections and can be excitatory or inhibitory. Inhibitory lateral input, which can be from within-column or intercolumn neurons, is divisive and provides the source of the control signal used for unlearning in conflict learning. Inhibitory feedback input is subtractive and used in the computation of ambiguity.

A model neuron $j$ has a continuous firing rate $x$ based on integrating weighted inputs,

$$x_j = f\left(g\left(\frac{\mathrm{FF} + (\mathrm{Lat} \cdot \mathrm{FF}^2) + (\mathrm{FB} \cdot \mathrm{FF}^2) + \epsilon}{1 + \mathrm{Inhib} + \mathrm{Ambiguity}}\right), \theta_j^{\mathrm{fast}}\right), \tag{3.1}$$

where FF, Lat, and FB represent the sum of weighted inputs of all feedforward, excitatory lateral, and feedback inputs, respectively. Each sum is calculated as $\sum_{i \in \mathrm{type}} w_{ij} x_i$, where $w_{ij}$ is the weight between neurons $i$ and $j$. Inhibitory lateral inputs are excluded here and instead apply divisively as Inhib. Note that feedback and lateral connections are gated by feedforward input; they cannot activate a neuron in the absence of feedforward driving input.

Inhib represents the sum of weighted input of all inhibitory lateral connections that are at least as active as neuron $j$: $\sum_{i\,|\,x_i \geq x_j,\, i \in \mathrm{lateral}} w_{ij} x_i$. The activation requirement supports the winner-take-all dynamic of a column by preventing the winner from being inhibited. These inhibitory inputs come from both within-column and intercolumn sources and are used in the computation of $\kappa$ in conflict learning (see equation 2.2).

Ambiguity is defined to be the minimum of the total excitatory and inhibitory feedback: $\min(\mathrm{FB}^+, \mathrm{FB}^-)$, where $\mathrm{FB} = \mathrm{FB}^+ + \mathrm{FB}^-$.

$\epsilon$ is a noise term sampled from a normal distribution: $\mathcal{N}(0, \sigma_{noise}^2)$. For neurons within the same column that share similar or identical driving input, noise causes one to be more active than the other, allowing inhibition to take place even in the absence of modulating input. This is especially crucial during learning, when modulatory input has not yet been differentiated over neurons in a column. Noise provides a mechanism to drive winner-take-all dynamics, which in turn drives differentiation of input. After learning has occurred, noise can cause the initial winner in a column receiving only driving input to be random, but this is quickly corrected through later-arriving modulatory input.

$g(x)$ is a gain control function that dampens any activation that exceeds 1.0 and applies a gain normalization term $\gamma$:

$$g(x) = \frac{\min(x, 1.0) + \max(0.0, \log_{10} x)}{\gamma}. \tag{3.2}$$

The desired activation range for neurons is [0, 1.0], though neurons are allowed to exceed this upper level of activation temporarily. The dampening function $g(x)$ does nothing to activation within the nominal range but pushes overactivation back toward 1.0. Overactive output is thus quickly quelled over several network iterations.

The goal of the gain normalization term, $\gamma$, is to enable balanced winner-take-all dynamics to occur within a competitive column. It is essential that the activations of neurons within the column can be compared fairly, and this can happen only if the neurons are operating in the same range of activation. This is much the same effect as homeostatic synaptic scaling within real neurons (Turrigiano, 2008) and works similar to the normalization model proposed by Heeger (1992). To achieve this in the model, $\gamma$ is set proportionately to the highest activation of any neuron in the column: $\gamma = \gamma_{t-1} * \max_j x_j,\ j \in$ column. Note that gain control mechanisms like $\gamma$ must be averaged over time to avoid rapid oscillations caused by changing activation values with each model iteration (Heeger, 1992). A simple exponential average (see equation 3.3) suffices for this.

$f(x, \theta)$ sets the output to zero if it is less than a threshold value. Thresholds are described in section 3.4. Thresholds are further bound between a minimum ($\theta_{\text{min}}$) and maximum ($\theta_{\text{max}}$) value. The minimum threshold value is set such that the noise term $\epsilon$ in the activation function is unlikely to spuriously activate the neuron.

**3.4 Thresholds.** Biological neurons have adaptive thresholds that control whether their input is high enough to trigger activation (Nicholls, Martin, Wallace, & Fuchs, 2001). Often a model will set neuron thresholds to be a rolling average of activation (Stevens, Law, Antolík, & Bednar, 2013), but there are disadvantages to such a mechanism. If a long period passes without a neuron receiving input, it can begin to lower its threshold and fire for nonpreferred stimuli. This contributes to a network forgetting its learned weights when examples of certain features are sparse.

Additionally, a typical threshold does not work well when the input to a neuron changes dynamically over a short period of time. As the model neurons in this work receive feedback and even feedforward inputs at different time steps, it is essential that the threshold can capture this change in activation and still ensure the input is within some expected range. Thus, a threshold needs to behave much like the weights in conflict learning: it needs a long-term stable value but also needs to be adaptable on a short-term timescale.

To address both issues, a new threshold scheme was created that uses the short- and long-term weights in conflict learning, which provide a form of hysteresis for the neurons. Neurons have three thresholds based on long-term activations $x_j^{\text{ltm}}$ (i.e., activations calculated using the long-term as opposed to short-term weights) of their driving input:

- $\theta_j^{\text{max}}$: a rolling average of the maximum long-term activation
- $\theta_j^{\text{active}}$: a rolling average of above-threshold long-term activation
- $\theta_j^{\text{decay}}$: a rolling average of subthreshold long-term activation

These three thresholds effectively classify the activation of the neuron into three states: an active regime, where the long-term activation exceeds $\theta_j^{\text{active}}$; a subthreshold regime, where the long-term activation is between $\theta_j^{\text{active}}$ and $\theta_j^{\text{decay}}$; and a decay regime, where the long-term activation is less than $\theta_j^{\text{decay}}$. Much like the long-term weights, these thresholds are "hidden" state, in that they do not directly control whether the neuron activates. A fourth, faster-moving threshold, $\theta_j^{\text{fast}}$, controls whether the neuron actually activates based on its short-term activation $x_j$ and is determined by the input and the other three thresholds.

Thresholds are updated with the following equations, which use an exponential average that moves the threshold toward some new target value with a smoothing factor $s$:

$$\text{avg}(\theta_{old}, \theta_{new}, s) = (1 - s)\theta_{old} + s\theta_{new}. \tag{3.3}$$

When the neuron is in the active regime, $\theta_j^{\text{max}}$ is adjusted to $\text{avg}(\theta_j^{\text{max}}, x_j^{\text{ltm}}, s)$. $\theta_j^{\text{fast}}$ is adjusted to $\text{avg}(\theta_j^{\text{fast}}, \theta_j^{\text{max}}, s)$. The smoothing factors are chosen such that these thresholds rise rapidly and fall slowly.

For the subthreshold regime, if $x_j^{\text{ltm}} < \theta_j^{\text{max}}$, $\theta_j^{\text{active}}$ is adjusted to $\text{avg}(\theta_j^{\text{active}}, x_j^{\text{ltm}}, s)$. If $x_j^{\text{ltm}} < \theta_j^{\text{active}}$, $\theta_j^{\text{decay}}$ is adjusted to $\text{avg}(\theta_j^{\text{decay}}, \theta_j^{\text{active}}, s)$. The smoothing factors are chosen such that these adjustments occur slowly.

Finally, in the decay regime, $\theta_j^{\text{decay}}$ is adjusted to $\text{avg}(\theta_j^{\text{decay}}, \theta_{\text{minimum}}, s)$, and $\theta_j^{\text{fast}}$ is adjusted to $\text{avg}(\theta_j^{\text{fast}}, \theta_j^{\text{active}}, s)$. The smoothing factors are chosen such that the passive decay rate is very slow, while the reset of $\theta_j^{\text{fast}}$ occurs quickly.

The overall effect of these thresholds is a neuron that maintains a stable activation point through rapidly changing input. This is especially important given the short-term weights of conflict learning, which adapt as the neuron is firing. Due to the decay threshold, the neurons will not drop their thresholds quickly in the absence of input and will lower their threshold only if exposed to long periods of subthreshold activation. Since the thresholds are based on driving input only, neurons will not deactivate when their output is decreased by competition. It is only possible to affect the driving input of competing neurons indirectly through recurrent modulatory feedback to the sources of such driving input, which can indeed lead to deactivation. This inability to directly deactivate a competing neuron is critical, however, for the proper function of conflict learning, which relies on inhibition received through competition as a teaching signal for active neurons. This behavior of competing neurons receiving a nonpreferred stimulus (e.g., receiving the correct orientation but with opposite ownership polarity) to still activate can be seen experimentally in border ownership neurons (Zhou et al., 2000). Furthermore, this behavior allows two significant effects to take place with conflict learning: the first is that neurons within a column can learn the same driving input but differentiate on modulatory input, and the second is that lateral interaction between columns can drive differentiation over entire columns, leading to different driving input preferences among different columns.

## 4 Network Construction

This section details how the competitive column architecture is applied to a model of border ownership. The construction and wiring of the network are first detailed, followed by the training regimen that is used to learn features in an unsupervised fashion. The model builds on the previous work of Grant et al. (2017), which uses a feedback-based model of border ownership based on grouping (see Figure 2 for an illustration of grouping neurons). The model developed here introduces an additional layer of grouping, called the proto-object layer, which pools responses from both border ownership neurons and grouping neurons. As will be demonstrated, this additional layer gives the network increased scale invariance and the ability to respond to more complex input.

As mentioned in section 2.1, the network has no a priori knowledge or biasing toward learning border ownership. All neurons that undergo learning are identical, and the naming of the different layers is based on a semantic interpretation of their responses. A combination of local competition, learned connection patterns, and training input causes the network to become selective to border ownership.

**4.1 Network Construction and Wiring.** The network is constructed of up to five layers: an input layer, an edge response layer, a border ownership

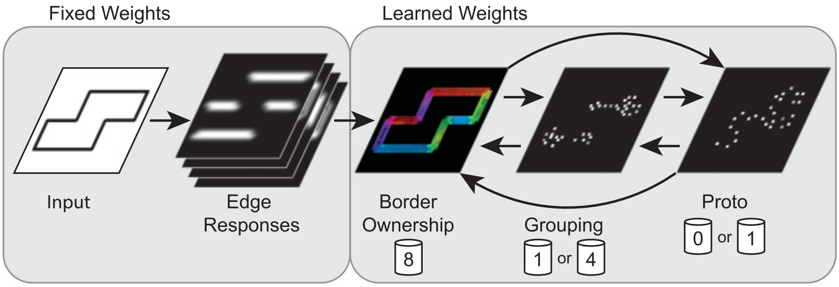

Figure 3: A schematic of the network configuration used for the experiments. Each layer of the network has the same dimensions, though the density and receptive field size tend to increase with layer depth. The input and edge response layers have fixed weights, as does the feedforward input to the border ownership layer. The border ownership, grouping, and proto-object layers all perform learning. These layers are interconnected to each other through feedforward and feedback projections. Each layer is parameterized by the number of neurons in its competitive columns, with different network topologies having different configurations. Layers show actual responses for depicted input, with the polarity colored according to preferred orientation for the border ownership layer.

layer, a grouping layer, and a proto-object layer. The border ownership, grouping, and proto-layers are parameterized by the number of neurons in their competitive columns. An overall schematic of the network can be seen in Figure 3.

Three topologies of networks are used in the experiments: 1G0P, which has one grouping neuron per column and no proto-object layer; 1G1P, which has one grouping neuron per column and one proto-object neuron per column; and 4G1P, which has up to four grouping neurons per column and one proto-object neuron per column. The border ownership layer is always fixed to have exactly eight neurons per column.

The input and edge response layers have fixed weights. Input is provided as a grayscale image that is held constant by the input layer until the next stimulus is presented. The edge responses are computed at four orientations (0, 45°, 90°, and 135°) by log Gabor filters (Field, 1987) parameterized by $\theta_{\text{gabor}} = \pi$ and $f = \sqrt{\pi}$. To reduce artifacts from edge filtering that can occur at small resolutions (e.g., shifts of edges), the network input is upsampled by a factor of 10 to be 800 × 800 pixels. The results of the filtering are then downsampled back to the 80 × 80 network size. This gives a total of 25,600 edge-responsive neurons, with four for each location in the network.

The border ownership layer consists of 80 × 80 columns, each containing eight border ownership neurons, for a total of 51,200 neurons. Each column receives driving feedforward input from every orientation at the same location in the edge-response layer. The input from the four orientation-selective neurons is duplicated within the column such that there are two border ownership neurons for each edge response.

The dimensions of the network are held constant across each layer, though beginning with the grouping layer, the density of neurons decreases. Additionally, beginning with the grouping layer, neurons are no longer arranged on a grid but are instead placed randomly using a Poisson-disc algorithm (Bridson, 2007), which aims to maintain a minimum distance between any two neurons. This random placement means that the grouping and proto-object layers are populated by a target number of neurons, though the actual number may vary slightly.

The border ownership layer is a slight exception to the others because its feedforward is fixed and its neurons are arranged on a grid, as opposed to random placement. This was done to ensure that the network could be trivially probed to determine polarity assignments; each border ownership column is known to have two neurons for each orientation, so comparing their activations and weight distributions directly provides the column's polarity.

For the 1G configurations, the grouping layer has a target of 25% of the density of the border ownership layer, with 1750 neurons. For the 4G configuration, four times as many neurons are generated as in the 1G configuration. For the 1P configuration, the proto-layer is populated at a density target of 10% of the border ownership layer, with 750 neurons.

The neurons in the border ownership, grouping, and proto-layers are wired in a similar fashion:

- Neurons receive excitatory and inhibitory feedback projections from every downstream layer. Feedback arrives from every neuron at the same retinotopic position within some radius in each layer. The radius used for immediately adjacent layers (i.e., $L + 1$), $r_s$, determines the base preferred scale of the network. Every successive layer doubles the size of the radius.
- Neurons receive excitatory feedforward projections from every upstream layer in the same fashion as feedback, with the radius doubling for successive layers.
- Lateral projections (excitatory and inhibitory) arrive from every neuron not in the same competitive column on the same layer within the base scale radius.

The incoming weights to each neuron are organized by their source (e.g., direct feedback, skip feedback, direct feedforward). Each group of weights is given some amount of uniform initial weight, as well as a maximum pool of total learnable weight. If the total weight within a group exceeds the maximum weight for that group, it is normalized back down to the maximum total weight. The number of connections in a group scales with the radius of the projection, which increases as layer-to-layer distance increases. This causes each individual connection to have less impact on the activation of the neuron the further the source of input.

Columns are wired as follows:

- Neurons are considered to be in the same column if they are within some radius of each other. This radius is set for each layer such that the number of neurons within a column hits a specific target.
- Every neuron in a column receives an inhibitory projection from every other neuron within the column.

The networks used in the experiments learn all feedback and column weights beginning with the border ownership layer. Feedforward weights are learned starting with the grouping layer. The base scale is set such that a 10 × 10 pixel square (prior to upsampling) can be fully contained within the input of a grouping neuron. This gives an $r_s$ of $5\sqrt{(2)}$. The radius used to determine if two neurons occupy the same column is $r_s/8$. For the 1G1P network, the above configuration leads to a network with around 54,000 neurons and 18 million synapses that participate in learning. Full details of the implementation, such as source code and detailed parameter information, can be found online (Grant & Itti, 2018).

**4.2 Training.** The network is trained by repeatedly exposing it to moving closed-shape outlines. The scale of shapes shown to the network is described in terms of the preferred stimulus size of the grouping neurons ($r_s$, which is set to optimally respond to 10 × 10 pixel square); a shape with a scale of 1 is one that ideally matches a grouping neuron, a scale of 2 is twice its preferred size, and so on. Shapes are always sized such that their height and width are whole multiples of the preferred scale.

Shapes are drawn from a 2 × 2 shape generator that can create all possible combinations of binary pixels on a square 2 × 2 grid, such that the resulting shapes are a single connected component and contain no holes. For a 2 × 2 generator, this means that there are four total possible shapes: a 1 × 1 square, a 2 × 1 rectangle, a 2 × 2 corner (symmetric L), and a 2 × 2 square. When these shapes are presented to the network, they are appropriately translated to pixels based on the parameterization $r_s$ (e.g., $1 \times 1 \rightarrow 10 \times 10$ pixels, $2 \times 2 \rightarrow 20 \times 20$ pixels, and so on).

After a shape has been randomly selected, it is given both a random orientation and a random position. The position is chosen such that the centroid will be within the network input. A random direction is then picked, and the shape is repeatedly translated in that direction until no portion of it can be viewed by the network. Each position of the shape is presented for 13 iterations of the network, which provides ample time for the influences of feedback to circulate through the network. A blank stimulus is applied for 13 iterations in between selecting a new random shape. All results

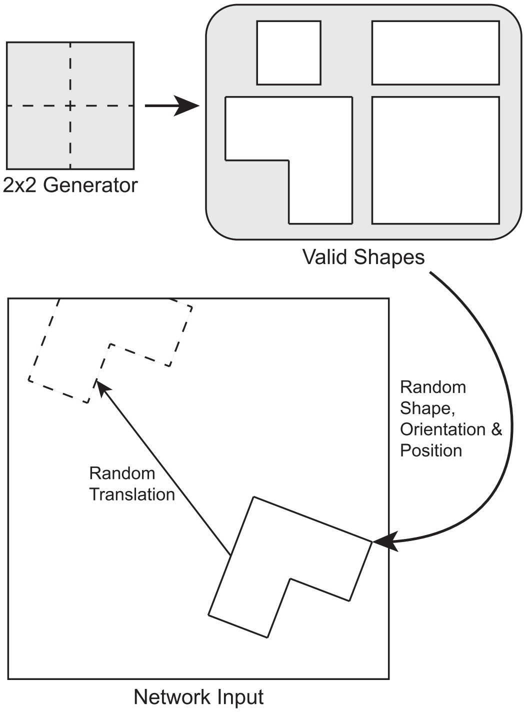

Figure 4: A schematic of the training procedure for the border ownership network. Shapes are generated from a 2 × 2 shape generator to produce all possible single component shapes with no holes. The generator yields shapes that are sized proportional to the base preferred scale ($r_s$) of the network. The valid shapes inset depicts all valid shapes that a 2 × 2 generator can yield. These shapes are then repeatedly sampled from, given a random orientation, and placed randomly on the network. The shape is then translated in a random direction until it leaves the field of view. This process is repeated, with a blank stimulus in between selecting a new shape, until the network is converged.

presented in the next section are on networks that were trained on 15,000 such randomized presentations of the four shapes from the 2 × 2 generator. These networks are then tested on 1895 novel shapes from a 4 × 4 generator. An illustration of this training process is provided in Figure 4.

## 5 Experiments

The competitive column and ambiguity are tested on various border ownership tasks using three network variants, with topologies as described in section 4.1. Networks were each trained as described in section 4.2.

For all experiments, the accuracy of a given border ownership assignment is computed by comparing a ground-truth assignment against the polarity assigned by the network. If the polarity points to the same side (plus or minus 90°) of the ground truth, it is considered correct, as in Teo, Fermuller, and Aloimonos (2015).

**5.1 Border Ownership Assignment.** To study the scale invariance and ability to generalize to challenging shapes, the networks were tested on shapes sampled from a 4 × 4 shape generator. The trained networks were only ever exposed to shapes from a 2 × 2 generator during training, so an overwhelming majority (99.8%) of the testing input was novel to the network.

Shapes generated from a 4 × 4 generator can be categorized by the strictest subset generator that could have created them. This is done by looking at the maximum of the shape's width or height. For example, a 3 × 2 rectangle would be classified as scale 3, whereas a 2 × 2 corner would be scale 2. Using this scheme, scale 1 has 1 shape, scale 2 has 3 shapes, scale 3 has 40 shapes, and scale 4 has 1,855 shapes. It should be noted that as the scale of the shape increases, so does the potential complexity of the shape. Scale 4 shapes can range from a simple 4 × 4 square to a snaking path with many locally ambiguous regions.

For evaluating the performance of the networks on these shapes, each shape was presented in the center of the network with no rotation. The polarity assignments were recorded after 13 iterations of the network, the same number of iterations a shape is presented for at a particular location during training.

Figure 5 shows the median polarity accuracies for each scale of shapes on all network configurations. All networks had essentially ideal performance on objects from the same scale at which they were trained. However, as the scale increased, performance significantly decreased for the network that lacked the proto-object layer (1G0P). At scale 3, the 1G0P network had a median accuracy of 87% compared to 100% and 98% for the 1G1P and 1G4P networks, respectively. At the highest scale, the 1G0P fared considerably worse, with a median accuracy of 61% compared to 73% and 74%, respectively.

The 1G0P network degrades in performance because its grouping neurons, which have a preferred stimulus scale of 1, are unlikely to activate as objects grow beyond this size. From the perspective of a grouping neuron, the stimulus goes from a fully closed contour to some portion of the presented object. While, depending on the learned thresholds, a grouping neuron may still activate for a corner or three-sided end to a shape, as the scale increases, it becomes increasingly likely that the grouping neuron is exposed to only a linear contour, which will fail to excite it above its threshold. Since the network additionally lacks proto-object neurons, which have

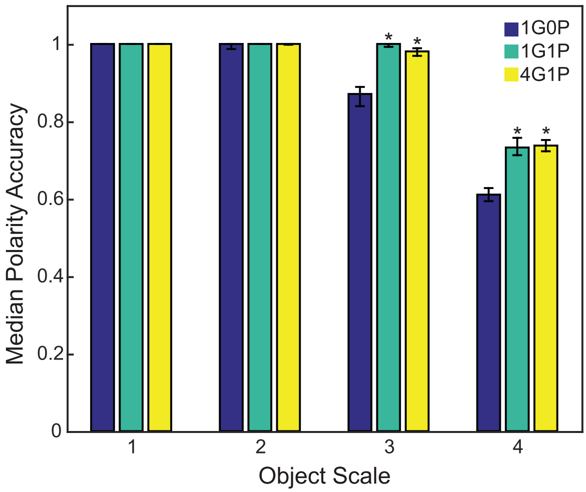

Figure 5: A histogram of network performance against different scales of generated shapes. All shapes from a 4 × 4 generator were tested and placed into scales by the strictest subset generator that could have created them. The networks with the proto-object layer (1G1P, 1G4P) show significant improvements in median polarity accuracy compared to the network without (1G0P) for all scales 2 and larger. Asterisks indicate extreme significance ($p$ value $< 1e - 5$, computed via Wilcoxon signed rank test) when compared to the 1G0P model. Error bars denote 95th percentile cutoffs.

larger receptive field sizes, it ultimately cannot generalize as well to larger input.

In the networks that contain a proto-object layer, the proto-object neurons have preferred stimulus sizes that are at least of scale 2, which gives the network a greater deal of scale invariance. Because proto-object neurons receive feedforward projections from both the border ownership layer and the grouping layer, their behavior is more nuanced than a scaled-up grouping neuron. Proto-object neurons learn a conjunction of scaled-up features, from the border ownership input, as well as more localized grouping responses. The interaction of feedforward and feedback between the grouping and proto-object layers also enhances scale invariance by allowing mutual reinforcement of activations.

The 4G1P network differs from the 1G1P network only in that it has considerably more grouping neurons, which are arranged into competitive columns. This causes the grouping neurons to compete over modulatory feedback from the proto-object layer much the same way border ownership neurons do from the grouping layer. The increase in grouping neuron density also creates extra lateral competition, which causes differentiation in learned feedforward features; some grouping neurons will learn features more receptive to corners or other subsets of a larger object. Since the overall topology of the network is the same as in the 1G1P network, not much difference should be expected between the two networks, which is confirmed by the median polarity accuracies, which are not significantly different from the 1G1P network. The more nuanced differences in the 1G1P and 4G1P networks are discussed in more detail in section 6.1.

Figures 6 and 7 show polarity assignments taken from a variety of shapes using the 1G1P network. These figures show all possible shapes from a 2 × 2 generator, but only a small subset of scale 3 shapes and a fraction of scale 4 shapes. As seen in Figure 5, performance degrades slightly as scale increases, which is largely because the increased scale makes it possible for the generator to create shapes that have multiple competing ambiguities. Since the network was never exposed to such examples during training, it is limited in its ability to resolve such ambiguities. However, in many cases, it is still able to resolve a large number of difficult assignments.

The network shows a slightly diminished activation for neurons that primarily receive support from the proto-object layer, such as the middle portions of the larger areas in Figures 6E, 6J, and 7D. This happens because the strength of direct feedback diminishes as layer-to-layer distance increases, and feedback from the proto-object that routes through the grouping neurons is gated by the driving input to those grouping neurons, which is diminished due to the scale of the input.

In other cases, competing ambiguities can cause a diminished activation of the border ownership response. In Figure 6J, grouping feedback on the outside of the figure competes with grouping and proto-object feedback on the inside of the figure. The combined support of the two scales causes the interior to outcompete the exterior feedback. As losing neurons in a column do not fully deactivate, it is possible for there to be residual activation of grouping or proto-object neurons supporting the incorrect polarity until their adaptive thresholds adjust.

The majority of incorrect responses tend to occur in regions where there is strong local evidence of a concavity and weak competing evidence, at any scale, for the correct assignment. This can be seen in Figures 7G and 7H, where some of the polarities along the interior concavity are either very weak or incorrect when the opposing side is an interior T junction for the shape. Figure 7F demonstrates that deep concavities are difficult to correctly assign, though this particular figure presents additional challenges: it has three concavities that all interfere with each other such that an incorrect assignment on one can affect the others.

### 5.2 Ambiguity.

Figure 8 demonstrates the impact of ambiguity on deciding the border ownership of a 3 × 2 c shape. The network used for this example is the 1G1P network, which has both grouping and proto-object

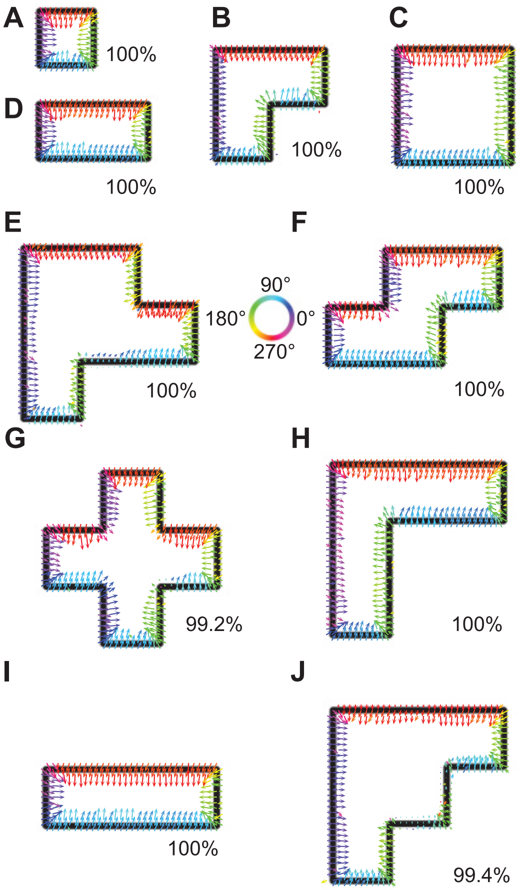

Figure 6: Sample shape responses from the 1G1P network on shapes up to scale 3. The input to the network is shown as a black outlined shape with arrows representing the network-assigned polarity overlaid. Shapes A through D are the only shapes that the network was exposed to during training and are created by a 2 × 2 generator. All shapes starting at E are scale 3. Each shape is accompanied by the percentage of border ownership neurons that had correct polarity assignments when the result was probed.

neurons. Ambiguity was disabled for the "no ambiguity" network by setting the ambiguity term in the activation equation 6 to 0.

Although both networks initially have a large grouping response within the concavity of the c shape, the network with ambiguity dampens the activation of neurons that give rise to this activation. Border ownership (BO) neurons along the concavity lower their activation, which in turn decreases

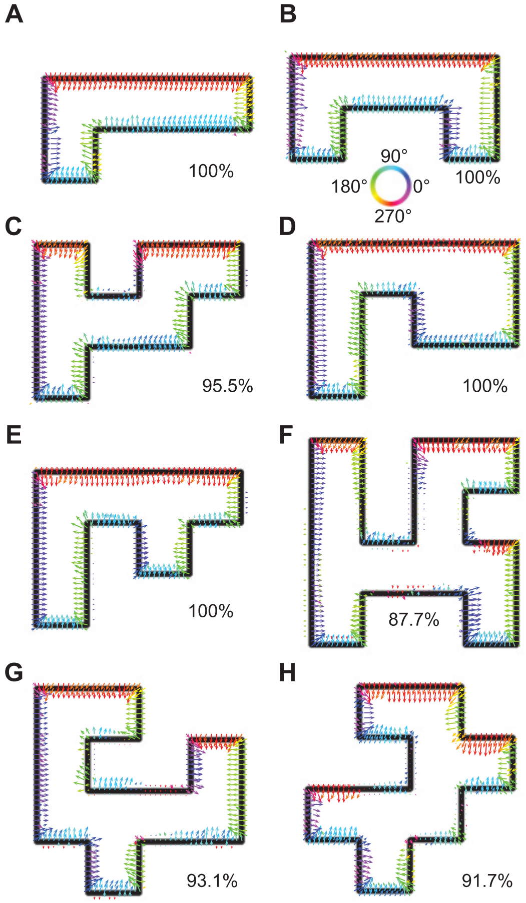

Figure 7: Sample shape responses from the 1G1P network on shapes of scale 4. Scale 4 shapes are not only considerably larger than shapes the network was trained on, but can contain very complex and locally ambiguous arrangements of features. The ability of the network to resolve ambiguity is diminished as the scale increases or if multiple portions of the input are ambiguous. Each shape is accompanied by the percentage of border ownership neurons that had correct polarity assignments when the result was probed.

the driving input to the grouping layer. This causes the grouping-layer neurons that were receiving input from solely ambiguous BO neurons to fail to meet their thresholds and turn off. With these grouping neurons deactivated, competition within the previously ambiguous BO columns causes the polarity to shift to the now unambiguous interior of the shape, which has uncontested grouping activation. This process takes a few iterations to

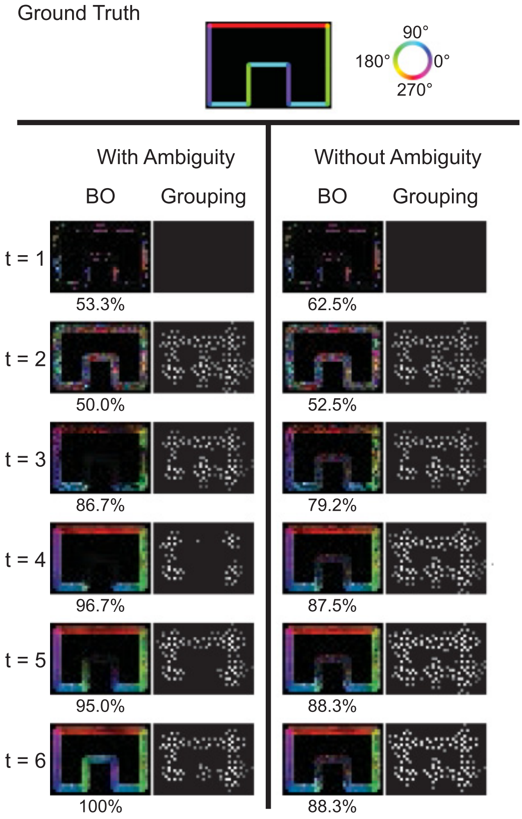

Figure 8: The progression of border ownership (BO) assignment for a 3 × 2 c shape with and without ambiguity. The top of the figure shows the ground-truth data, with edges colorized by the correct polarity assignment. The bottom portion of the image is split into two columns: with and without ambiguity. Each row shows the network state at a different time step, beginning with the onset of the stimulus and ending with the first iteration of correct polarity assignment. BO responses are accompanied by the percentage of BO neurons that had correct polarity assignments when the result was probed. At $t = 2$, the polarity assignment is driven by random noise and purely feedforward input. At $t = 3$, feedback begins to arrive from the grouping and proto-object layers. Note that the grouping neurons are initially responsive for the concavity of the c shape for both networks. The network with ambiguity shows dampening of ambiguous BO neurons starting at $t = 3$, when feedback input is received from both sides of each neuron. This dampened activity propagates to the grouping neurons at $t = 4$, which lowers the ambiguity of the BO neurons. At $t = 6$, the neurons are fully unambiguous, and the correct assignment emerges. Without ambiguity, the assignment in the concavity is both random and muted due to constant competition.

Table 1: Network Results with Ambiguity Disabled.

| Network | Scale 1 | Scale 2 | Scale 3 | Scale 4 |
|---|---|---|---|---|
| 0G1P, no ambiguity | 100% | 89% | 73% | 59% |
| 1G1P, no ambiguity | 100% | 97% | 86% | 67% |

Note: Percentages indicate the median polarity accuracy.

complete, but eventually an interior-only activation of the c shape drives an unambiguous response from the BO neurons.

The network without ambiguity has no way of resolving the polarity assignment of BO neurons along the concavity. The basic winner-takes-all dynamics of the column will indeed be choosing winners each iteration, but without a way to prevent feedback from applying equal amounts of excitation and inhibition to competing polarities, no stable winner can emerge.

Although not shown in the figure, proto-object neurons are also responsive to the input and supply feedback to both the BO neurons and the grouping layer. Since the strength of feedback decreases with layer-to-layer distance, the majority of the feedback influence the proto-object neurons exert on the BO neurons is channeled through the grouping neurons. Since feedback is modulatory and the lowered activation of the BO neurons causes inactivation of some grouping neurons, any feedback the proto-object neurons applied to those deactivated grouping neurons no longer affects the previously ambiguous BO neurons. Thus, not only does the grouping response converge with the unambiguous activation of BO neurons, but the proto-object response does as well. All layers of the network settle in an unambiguous state.

There is some residual activation among the grouping neurons even after a nonambiguous result is reached because neurons in a competitive column are not fully deactivated even when losing the winner-take-all competition. This causes a small amount of input to reach the grouping layer from the incorrect polarity BO neurons around the concavity, which may be enough to turn on some grouping neurons. The long-term thresholds of the neurons need to be such that the neurons will turn on from the initial driving feedforward stage, in which it can be expected that half of the neurons are randomly incorrect. Thus, the thresholds are often low enough to see some activation at this stage. This residual activation causes a slight oscillation in activation among the ambiguous neurons (due to increased ambiguity), which eventually is removed from the network by two factors: the dampening function (see equation 3.2) and the fast-moving component of the threshold (see section 3.4).

To further test the importance of the ambiguity mechanism, the experiment of section 5.1 was repeated using the 0G1P and 1G1P networks with the ambiguity disabled. The resulting median scores are listed in Table 1. In both cases, the networks with ambiguity disabled performed significantly worse than their counterparts with ambiguity enabled (see Figure 5). This result demonstrates that the mechanisms that benefit the polarity assignment in the c shape generalize to a wide variety of the tested shapes.

**5.3 Detailed Rotation, Scale, and Translation Invariance.** Although the scores in Figure 5 capture the overall trend of the network to be invariant, it is useful to look at more detailed metrics of invariance. The network is evaluated on two example shapes of differing base scale and complexity to give insight into the network's invariance to scale, translation, and rotation. All results in this section are from the 1G1P network that was trained on shapes from a $2 \times 2$ generator.

A $1 \times 1$ square shape and a $3 \times 2$ c shape were presented to the 1G1P network with various transformations applied. To test rotation, the shape was placed in the center of the network and then rotated in continuous steps up to $360^\circ$. As with previous tests, results were taken after letting the network settle for 13 iterations. A blank stimulus was applied in between each rotation to prevent any memory of the previous rotation from influencing the result. The results can be seen in Figure 9A. Due to the way the network was trained with random rotations of generated shapes, the network is fully rotation invariant. This is largely due to the nature of grouping and proto-layer neurons, which optimally respond to a closed contour in their receptive fields.

To test scale invariance, the shapes were presented to the center of the network and scaled down by a factor of 0.5 up to a factor of 2.0. Unlike the shapes yielded from the shape generators, these shapes were allowed to have fractional sizes. A blank stimulus was applied between each successive scale. The results are presented in Figure 9B. For simpler shapes, the network is scale invariant over a wide range of object scales. As the complexity of objects increases, such as with the c shape, explicit activation dynamics driven by ambiguity are required for an accurate polarity assignment. This is more sensitive to changes in scale since there is a limited range in which the grouping, proto, and border ownership neurons can interact with each other. However, the network still displays an impressive band of high accuracy and does not catastrophically fall off. When scaled up by a factor of 2.0, the c shape is six times larger than the preferred grouping neuron stimulus size along its longest dimension.

Finally, to test translation invariance, the shapes were positioned at 121 locations, and the network's predicted polarity assignments were recorded. Shapes were initially presented with their centroids in the upper-left corner of the network before systematically traversing the network horizontally and vertically for each sampled location, as seen in Figure 10. The training regime translates shapes randomly all over the network, so it is unsurprising that the final network shows translation invariance. So long as the shape can be fully displayed within the network, the resulting polarity assignment

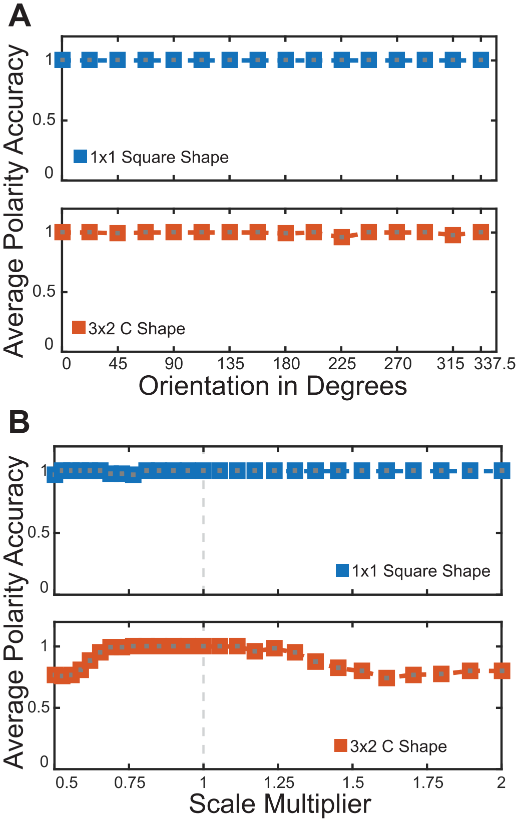

Figure 9: Orientation and scale invariance for a 1 × 1 square shape and a 3 × 2 c shape on the 1G1P network. (A) The average polarity accuracy of the network is plotted as a function of the rotation of the shape. (B) Presented shapes were scaled from half to twice their original size. Presented sizes thus included many fractional amounts of the preferred scale of the network, to which it was never exposed in training. The average polarity accuracy of the network is plotted as a function of the multiplier applied to the shape scale. A vertical gray line indicates the base scale for the shapes.

is accurate. There is some fall-off at the edges of the network due to specific implementation details. The trained networks used for all tests were sized such that no artifact of this could affect the results.

**5.4 Contours from Natural Images.** Natural images represent a substantial step up in contour complexity compared to the procedurally generated shapes used in the previous experiments. In this experiment, the

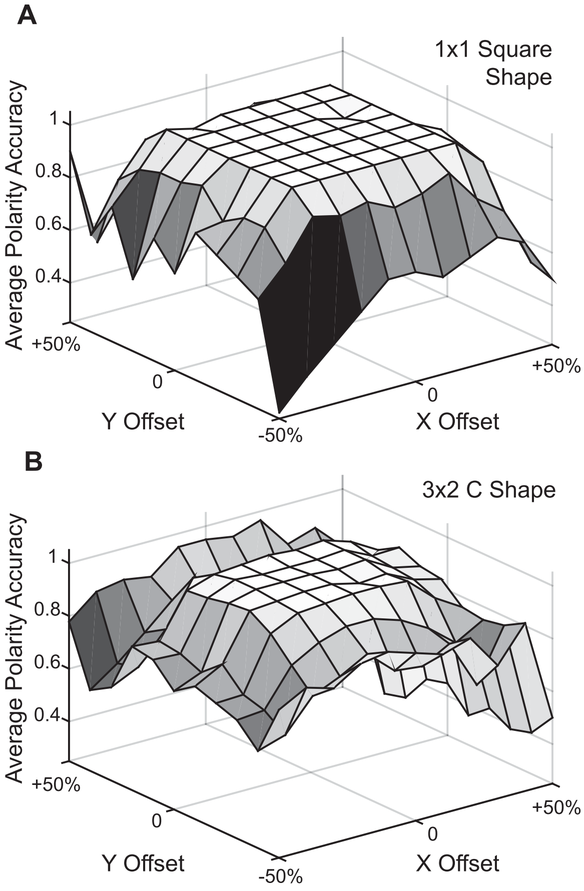

Figure 10: Translation invariance for a 1 × 1 square shape and a 3 × 2 c shape on the 1G1P network. Presented shapes were translated such that their centroid was placed in 121 sampled locations spanning the entire network input. The average polarity accuracy is plotted as a function of the offset of the centroid in terms of network size. The upper graphic is for the 1 × 1 square shape and the lower graphic is for the 3 × 2 c shape.

network is given a sample of line drawings of natural images from the Berkeley Segmentation Data Set (BSDS500; Arbelaez, Maire, Fowlkes, & Malik, 2011). These line drawings were annotated by human subjects.

The 1G1P network is used for all natural contour examples, with a minor adjustment made to neuron thresholds. Since this network was trained on artificially generated shapes, its thresholds have adapted to the expected activation driven by the Gabor filter-edge responses on these shapes, which were computed at only one spatial frequency. The natural images have entirely different statistics and edges that may not correspond to the thickness of the artificial shapes. To address this, the activation thresholds of all neurons are decreased toward their decay thresholds. This allows the neurons to respond to stimuli that may give decreased driving input when compared to the procedurally generated shapes the neurons were trained on. Given enough time with subthreshold input, the neurons would naturally adjust their thresholds in a similar manner (see section 3.4 for details on how the thresholds work).

Results are shown in Figure 11, with the original color images provided for context. Ground-truth responses were created to calculate the percentage of correct polarity assignments by hand-labeling the contours of each figure such that the polarity pointed roughly orthogonal to the contour, toward the interior of the object. For the car image, only the exterior contours were considered, as the correct assignment of the interior contours is more subjective.

These images showcase examples with many features the network has never seen in its training: curves, scales that are not whole multiples of the preferred stimulus size, missing contours, holes, and occlusions. As such, the network does make some mistakes and is unable to give an unambiguous response for some of the contours, but the overall outlook of the responses is highly favorable given the limited training on just four procedurally generated shapes.

The network makes few mistakes in Figure 11A, with the only noticeable errors occurring near the tail as well as the propeller. Competition among different scales leads to the mistake near the propeller. The propeller concavity is supported mostly by grouping neurons, whereas the largest portion of the plane interior is roughly the scale of the proto-object neurons. Direct feedback from the proto-object neurons to the border ownership neurons is weaker than direct feedback from the grouping neurons, as the layer-to-layer distance is greater. In addition, indirect feedback from proto-object neurons, routed through intermediate grouping neurons, is modulated by the driving input to those grouping neurons. Thus, in cases where intermediate grouping neurons are not very active, such as is the case along the largest portions of the plane interior, the total feedback received is weak. Because of these factors, the proto-object response on the interior of the airplane loses to the stronger but incorrect grouping neuron response in the propeller concavity, causing incorrect or ambiguous polarity assignment.

Figure 11B demonstrates that a portion of the input can be missing, and correct polarity assignments can still be computed. In the 1G1P network, grouping and proto-object neurons learn to fire given sufficient input from a roughly annular distribution of border ownership neurons, with no requirement that those firing neurons are contiguous.

Figure 11C is largely correct, with some diminished response along the bottom of the car. The car is significantly larger than the proto-object

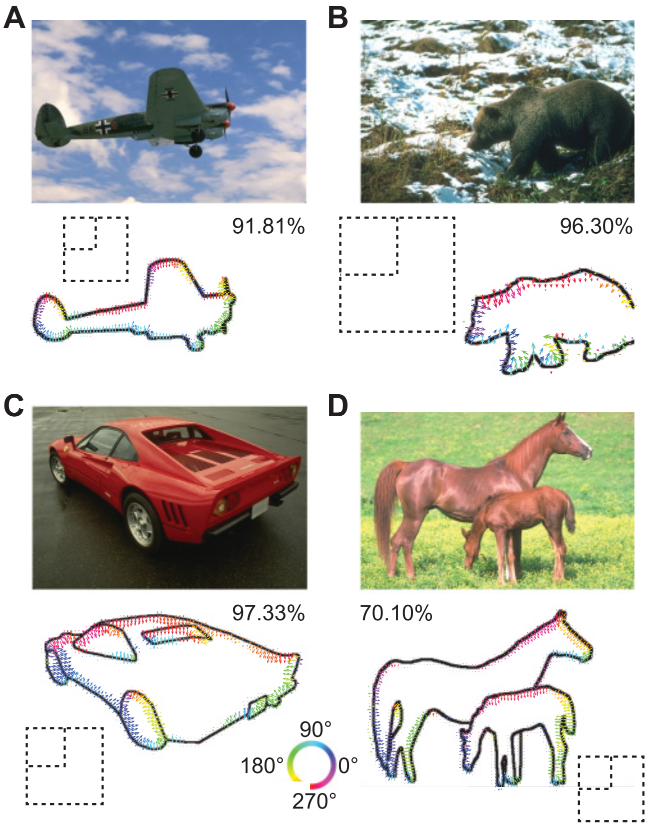

Figure 11: Border ownership assignment on contours of selected natural images. The upper portion of each panel shows the full color natural image, and the bottom shows the ground-truth human-drawn line drawing that served as input to the model, overlaid with the border ownership assignment generated by the network. Each subfigure is also accompanied by the percentage of border ownership neurons that had correct polarity assignments when the result was probed, as well as an outlined box that shows the preferred stimulus size of the grouping and proto-object neurons, scaled to the input. The figure is best viewed zoomed in, for detail. (A) Contours from an airplane. The network has never previously been exposed to curved contours or even shapes that did not match a whole multiple of its referred stimulus size. (B) Contours from a bear. The network still manages to assign correct ownership even though a portion of the contour is missing. (C) Contours from a car. Note that this object contains several "holes," or enclosed regions within the overall silhouette, which are an entirely novel input. (D) Contours from a pair of horses. Note that this image contains two objects at vastly different scales, with the smaller horse occluding the larger one. The network has never before been exposed to occlusion.

receptive field size. It is possible that a deeper hierarchy, which would in turn have neurons with even larger receptive field sizes, could alleviate this situation. The proto-object neurons would then receive feedback from some deeper layer, and thus feedback weight would increase through both direct and indirect connections back to the grouping neurons.

Figure 11D encapsulates the difficulty of scale-invariant processing in the context of border ownership. The legs of the horses are smaller than the preferred grouping stimulus size, and the body of the larger horse is at about the limit of the larger proto-object receptive field. In addition, the legs create long concavities that lead to ambiguous local inputs. The contour of a horse can be seen as a natural image analog of the c shape and would serve well as a litmus test for future work.

## 6 Discussion

This competitive column architecture, combined with the previously developed conflict learning rule, provides a glimpse into how feedback, modulation, and inhibition may shape learning. Although recent efforts have made some progress toward understanding some aspects of feedback processing (Lillicrap, Cownden, Tweed, & Akerman, 2016), how feedback influences activation and learning remains a fundamental question. The competitive column and ambiguity, as applied to a model of border ownership, give some intuition that may be useful in probing the understanding of the brain and developing more powerful models of object recognition. Although it is unsupervised, conflict learning allows the model to have a built-in teaching signal, which as backpropagation-based approaches have shown, is an excellent way to assign blame and modulate learning.

This focus on the importance of feedback for both learning and network dynamics is a departure from a large body of prior work focusing on feedforward processing, particularly with respect to object recognition (DiCarlo, Zoccolan, & Rust, 2012). Animal models (Zhang et al., 2011), human psychophysical experiments (Isik, Meyers, Leibo, & Poggio, 2013; Kheradpisheh, Ghodrati, Ganjtabesh, & Masquelier, 2016), and feedforward computational models (Serre et al., 2007) have shown that feedforward processing is sufficient for at least some level of object recognition. These claims have largely focused on the ability of feedforward information, decoded late in the visual hierarchy, to be useful for rapid categorization tasks (e.g., animal versus nonanimal, 100 ms or less of visual input). However, these types of tasks cover only a minimal set of visual experience, whereas scenes often contain multiple objects, have complex backgrounds, or are driven by high-level goals. Feedback is likely necessary for object recognition in these more challenging contexts, as well as intermediate visual processes such as figure-ground segmentation or directing top-down attention (Kheradpisheh et al., 2016). Zhang et al. (2011), for example, showed that when top-down attention was directed to a single object in an array of multiple objects, an initially ambiguous decoding of neurons in IT became unambiguous for the attended object.

The competitive column model does not contend with the idea that some core of object recognition can be resolved with purely feedforward processing (DiCarlo et al., 2012), and indeed the recent successes of cortical-inspired deep learning models are nearly all purely feedforward (Kheradpisheh et al., 2016). Consider the grouping response with ambiguity in Figure 8. At $t = 2$, the grouping response is driven entirely by feedforward activation. At $t = 5$, when the border ownership response has resolved ambiguity through several iterations of recurrent processing, the change in grouping neuron activation is mostly restricted to the ambiguous portion of the input. The change in the proto-object responses, which is not depicted, is even less discernable. It is entirely conceivable that a classifier trained on the initial feedforward response could still learn that the pattern of grouping activity at $t = 2$ is consistent with a c shape or some category associated with a c shape.

The same cannot be said about the border ownership responses, which dramatically change over time as recurrent processing takes place. The initial feedforward response of the border ownership neurons is ambiguous and covers multiple semantic interpretations (i.e., polarity) of the input simultaneously. Similar behavior occurs in any layer of the network that has multiple neurons competing within a column (see section 6.1 for discussion on competition among grouping neurons).

The competitive column model thus predicts that the initial wave of feedforward information activates a plurality of neurons with shared features that can have different semantic interpretations. Recurrent processing causes the network to move toward a selection of these features that share an unambiguous semantic interpretation. This suggests that a rapid feedforward response is useful for high-level categorization but that more nuanced information, such as border ownership, is both unavailable at the top of the hierarchy as well as not immediately computed earlier in the hierarchy.

This viewpoint of feedback processing does not quite fit into the dichotomy described by DiCarlo et al. (2012). Information does not need to reach the end of the hierarchy before it can influence earlier stages; in the competitive column model, recurrent feedback processing permeates every level of the hierarchy, is continuous, and occurs nearly immediately. What ultimately matters is how ambiguous the scene is: the lower the ambiguity, the more reliable the feedforward pass of information is. In the context of the competitive column model, ambiguous scenes are those that promote multiple simultaneous interpretations of shared features.

To summarize, both feedforward and feedback processing play important roles. Feedforward information drives the initial response of the network, while a continuous interaction of recurrent feedback and feedforward processing provides refinement. Information is encoded at varying levels of the hierarchy, and a densely connected network of direct and skip connections makes it available throughout (Markov et al., 2014). It is likely that much of this feedback processing can be biased through top-down attention, context, or other mechanisms but that it remains a critical component for visual experience.

**6.1 Toward Proto-Objects.** The competitive column architecture encourages competition and a partitioning of learned features at every level of the hierarchy. Figure 12A shows a representative set of feedback receptive fields for a single border ownership column. The learning rule, combined with the column architecture, causes neurons to learn conjunctions of high- and low-level features—in this case, orientation selectivity combined with polarity preference. While this effect is most noticeable at the border ownership layer, especially due to their structured feedforward input, it is also evident at deeper layers of the network.

The new layer of neurons added to the network to support scale invariance is called the proto-object layer. The goal of these neurons is to provide a grouping mechanism over a larger receptive field size. If these neurons received input only from the border ownership layer, they may have acted just like grouping neurons and learned large annular receptive fields. However, proto-object neurons also receive direct input from the grouping neurons, which gives them interesting visual features, seen in Figures 12B to 12E. While it is difficult to give a precise description of their preferred stimulus, proto-object neurons learn conjunctions of midlevel (grouping) and low-level (border ownership) features. This can be seen especially in Figures 12C to 12E, which appear to learn a general large-scale surround (from direct border ownership input) with a preference for more localized input (from grouping input). The features learned here are dependent on the visual experience of each neuron, as well as the competition it receives from between-column lateral connections. Some neurons, such as in Figure 12B, happen to learn their grouping input in alignment with their border ownership input, giving the perception of a large annular receptive field.

This use of the term *proto-object* is similar to that used by von der Heydt and others, though it differs somewhat in that von der Heydt (2015) treats the grouping neuron response as a proto-object. In the 1G1P model, the grouping neurons respond to annular, convex configurations of border ownership neurons, which are very general responses that have little unique relation to any particular object. The 4G1P model, however, learns grouping features that are more in line with parts or subcomponents of objects and are a better match to what the name *proto-object* implies.

This specialization into subcomponents can be seen in Figures 12F to 12I, which depict grouping columns of varying sizes and the learned feedforward and feedback receptive fields. In cases where a column contains a single neuron, there is little competition over feedforward and none over feedback, so the neuron learns an annular receptive field and associated feedback. This is the most general grouping feature of a border ownership

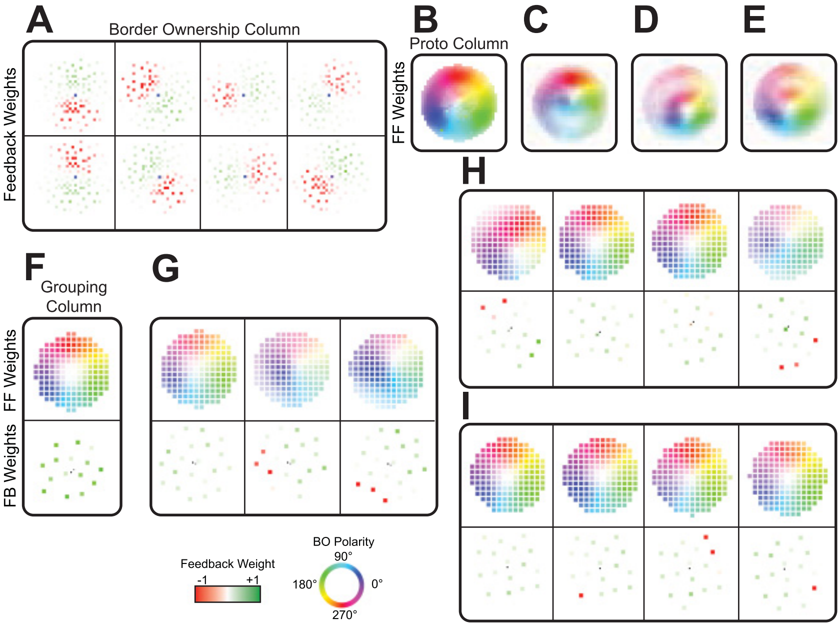

Figure 12: Example receptive fields of neurons in competitive columns of the 4G1P model. (A) The learned feedback (FB) weights of a single border ownership column. Eight neurons are depicted, with two for each orientation. The neurons learn to compete over polarities for each orientation. Green indicates an excitatory connection and red an inhibitory connection. Feedback is learned from both the grouping and the proto-object layer. (B–E) The learned feedforward (FF) weights of four proto-object neurons. The proto-object layer has columns with single neurons. Feedforward input is colored based on the preferred polarity of the border ownership neuron that supplied the input. Input comes from both border ownership and grouping neurons: any input from a grouping neuron is colored by tracing its feedforward inputs to the border ownership layer. (F–I) Learned feedforward and feedback connections for four grouping neuron columns. Column size is dependent on the random distribution of neurons in the grouping layer, so some columns have fewer than four neurons. The top row in each inset depicts learned feedforward weights, colored as in the proto-object columns. The bottom row depicts the learned feedback weights from the proto-object layer, colored as in the border ownership column. The figure is best viewed digitally, zoomed in.

network and a prototypical grouping neuron response. In cases where there are multiple neurons, competition drives a differentiation over both feedforward and feedback features. In Figures 12G to 12I, the learned feedforward receptive fields initially learn very similar distributions before some of the neurons specialize. This specialization is driven by both a partitioning of feedback, much as in the border ownership neurons, as well as between-column lateral competition over driving feedforward input. Figures 12G and 12H illustrate columns where some neurons have begun to specialize both their feedforward and feedback receptive fields, while other neurons still retain features similar to a neuron without competition, such as that seen in Figure 12H. In Figure 12I, the neurons are beginning to show specialization over feedback while maintaining similar feedforward preferences, much as the border ownership neurons operate. Unlike the border ownership neurons, the feedforward connections here are learned. Thus, the competitive column architecture shows the ability to replicate the same behavior that was forced in the border ownership neurons by fixing the feedforward input: a replication of driving input with diversification among modulatory input.

This increased differentiation of feedforward and feedback input occurring with increased competition is a hopeful sign for future expansion of the model. The experiments also demonstrated that there is likely a benefit to having increased representation within the network for generalization to larger scales and more complex features.

**6.2 Modeling Border Ownership.** As mentioned in section 2.1, the model of border ownership in this work shares similarities with several others (Mihalas, Dong, von der Heydt, & Niebur, 2011; Russell et al., 2014) that are based on the grouping hypothesis (Martin & von der Heydt, 2015). Both of these models and the current work build on a theory of using opposing regions of facilitatory and suppressive input to determine border ownership polarity (Sakai & Nishimura, 2006; Sakai, Nishimura, Shimizu, & Kondo, 2012). In the context of border ownership neurons, these facilitatory and suppressive regions are inputs that either agree or disagree with the preferred polarity of the neuron. For example, excitatory feedback from a grouping neuron on the same side as the preferred polarity is facilitatory, while inhibitory feedback from the opposing polarity's grouping neuron is suppressive. Sakai et al. (2012) focus on the general concept of facilitatory and suppressive input and do not dictate the source (e.g., feedforward, feedback) of such information.

The work of Sakai et al. (2012) is especially relevant as their model of border ownership is tested on artificial shapes generated in a similar fashion to those in this work, as well as natural images from a related data set. The results are not directly comparable however, as their shapes are categorized by the number of 1 × 1 squares the shapes are constructed from, and border ownership is measured only at a single location. Their shapes consist of 4, 6, or 8 1 × 1 squares, whereas the shapes in this work are sampled from a 4 × 4 matrix of squares, with generated shapes having between 1 and 16 1 × 1 squares.

Their model uses the average response of a large population of BO neurons with randomly generated regions of facilitation and suppression to assign polarity. This averaging over a large number of random features gives a high degree of consistency to the border ownership response. The learned receptive fields for the current work (see Figure 12), match the most consistent neurons in Sakai et al. (2012), which have symmetric opposing regions of facilitation and suppression. One important difference is that Sakai et al. (2012) emphasize the relative importance of suppressive input, whereas the current model relies on a balance of facilitatory and suppressive input to drive the ambiguity mechanism. The notion of ambiguity is unique to the current model. Sakai et al. (2012) note how assignment of polarity within the concavity of a c shape demonstrates lower consistency, which ambiguity is directly designed to address. As demonstrated in the more complex examples of Figure 7, the surround configuration to a concavity influences the ability of the network to resolve the ambiguity (Sajda & Finkel, 1995), which is also true of actual border ownership neurons (Sakai et al., 2012).

The overcomplete nature of the randomly modulated BO neurons of Sakai et al. (2012) likely affords the network a higher amount of invariance over border ownership assignment. It would be worthwhile to investigate if additional randomness could benefit the current model. The size of the competitive columns in the current work, in conjunction with conflict learning, results in receptive fields that are highly symmetric. However, if the number of neurons in the border ownership columns were increased, it is likely that a more diverse set of neurons could be learned, much like what is seen by increasing the size of the grouping columns (see Figure 12).

The model of Mihalas et al. (2011) uses feedback grouping mechanisms to compute border ownership. The largest structural difference from the current model is the existence of inhibitory feedforward connections from BO neurons to grouping neurons that support nonpreferred polarity assignments. The competitive column model does not currently have a notion of inhibitory driving connections, though it is not difficult to imagine potential benefits from such a mechanism. In the context of ambiguous shapes, for example, inhibitory feedforward could reduce the oscillatory behavior of ambiguity seen in the grouping neurons in Figure 8 and discussed in section 5.2. It is unclear what the implications would be on learning for such a mechanism, however.

Russell et al. (2014) develop a feedforward model of border ownership that combines traditional grouping mechanisms (Craft et al., 2007) with center-surround attention mechanisms (Itti et al., 1998). Neurons with large center-surround receptive fields in opposing polarities (on-off and off-on) are used to compute a feedforward-driven notion of objectness, which biases the activation of BO neurons, as a feedforward analog to grouping neurons. Inhibitory feedforward is again utilized, with the center-surround neurons inhibiting inconsistent BO neurons. The model achieves invariance through the use of successive feature pyramids, much like the Itti et al. (1998) model of saliency.

Although it shares various similarities to the models discussed, one of the most important contributions of this work is the demonstration that the complex features and connections of border ownership can be learned through visual experience alone. The competitive column framework, which has no built-in knowledge of border ownership, learns highly invariant border ownership selective neurons supported by grouping mechanisms. This gives additional credibility to the functional organization of border ownership supported by the discussed models.

**6.3 Generalizability.** Remarkably, the network shows a capability to generalize from the simple 2 × 2 shapes it was trained on to the complex contours of the tested natural images, as seen in section 5.4. In modern deep convolutional networks, the ability of a network to generalize to unseen classes of input is a fundamental problem that which requires retraining and risks catastrophic forgetting (Kirkpatrick et al., 2017). The network used here, with three active layers, is still quite shallow by deep learning standards, which can be hundreds of layers deep (He et al., 2016), or by biology, which suggests that up to 10 levels of processing are involved in the visual hierarchy (Felleman & Van Essen, 1991; Markov et al., 2014). Perhaps some ability of the network to generalize to larger and more complex shapes and contours is due to this shallow nature and lack of features tied to overly specific input (as is blamed in some deep networks; see Long, Cao, Wang, & Jordan, 2015; Sun, Feng, & Saenko, 2016), but the experiments suggest that the invariance and generalizability of the network should increase as more layers and competition are introduced.

Although the network does display a high amount of scale invariance, scale is still one of the most challenging aspects of more natural input. Figure 11D best demonstrates the difficulty of multiscale interaction and processing. The network is unable to strongly classify the polarity of the larger horse in particularly wide regions of its back and similarly unable to settle on an unambiguous assignment in the long concavities created by the younger horse's legs. Figure 11C also presents challenging questions about what the correct assignment of polarity is for enclosed objects, or "holes," should be.

It remains to be seen if the competitive column approach will generalize to fundamentally different tasks outside of what it has been demonstrated on, that is, tasks other than orientation selectivity (see Grant et al., 2017) or border ownership. This type of adaptability is one of the great strengths of deep learning (Neyshabur, Bhojanapalli, McAllester, & Srebro, 2017). Though the competitive column model and conflict learning were primarily designed around solving orientation selectivity and border ownership, the methodology underlying them is general; it is purely the statistics of input that drive the learning and emergent behavior.

**6.4 Biological Implications.** The proposed mechanism of ambiguity is entirely hypothetical. Yet this method is not outside the realm of what is plausible for a neuron to compute; it is based on local computation and operates in a similar fashion to divisive inhibition. Given that the general model fits in with current ideas of border ownership, it is reasonable to hypothesize that a similar mechanism to ambiguity exists in real neurons. The model in its current form predicts that neurons receiving ambiguous input should dampen their activity before reaching a stable response and, further, that this stable response should show a decaying oscillation due to a feedback loop.

The model of border ownership demonstrated here is inspired by the feedback model of Craft et al. (2007), which is predicated on the grouping hypothesis (Martin & von der Heydt, 2015). Grouping cells, as originally described, are hypothetical units. Although synchrony that supports the grouping hypothesis has been observed, individual grouping cells have yet to be identified (von der Heydt, 2015). The results of the 1G4P network, as seen in section 6.1, suggest that it may not be strictly necessary for the canonical grouping neuron to be present for a border ownership response. It is possible that the same relationships encoded by a grouping neuron are also present in lower-level features such as curves and corners. In the 1G4P network, grouping neurons initially show canonical, annular receptive fields, but these are largely replaced by more specialized features over time as more competition occurs. The feedback circuit between these specialized units and the border ownership neurons remains intact, however, maintaining an association with a particular polarity. This thus paints the picture of border ownership as a process supported by an increasing amount of modulatory recurrent activity as the hierarchy deepens and responses become more object-like.

**6.5 Related Frameworks.** The competitive column model presented here has many similarities with two recent models of cortical processing: the capsule model of Sabour, Frost, and Hinton (2017) and the column-based model of Hawkins, Ahmad, and Cui (2017). Common to all three models is a notion of creating complex responses to input that bind to multiple features. In this work, edge-responsive features are bound to polarity information, in Sabour et al. (2017), features associated with numeric characters are bound to location information, and in Hawkins et al. (2017), sensory features are bound to allocentric location signals. The capsule model learns using supervised backpropagation of error, following a long-established path of artificial neural networks. Although it has a notion of prediction and alignment between successive layers, the capsule model does not truly feature recurrent or modulatory processing.

The competitive column model and the model of Hawkins et al. (2017) are more similar. Both feature an architecture centered around the abstract notion of a cortical column, and both use modulatory input sourced from lateral and feedback connections. While the model here places an emphasis on the influence of modulatory feedback, Hawkins's model is more focused on lateral connectivity. A primary difference, aside from the learning rules and activation dynamics, is that the competitive column model learns conjunctions of features on its own. Hawkins's model requires that the location signal, which is ultimately bound to learned features, be presented along with sensory input. Though an argument is presented for this being a fundamental computation of a column, it remains an a priori signal for the model as it currently is implemented. In addition, their model does not explain how features within a column can come to have similar feedforward receptive fields while differentiating over modulatory input; the competitive column model is capable of doing this (see section 6.1).

Ultimately all three models explore a more fundamental role for column-like organization and contain dynamics that differ greatly from established feedforward artificial neural networks. Perhaps these dynamics, particularly those of modulation, will enable better models of cortical processing and give greater intuition for how the brain works.

## 7 Conclusion

The competitive column model, combined with conflict learning, provides a framework for learning invariant features that can differentiate over modulatory as well as driving inputs. The architecture was demonstrated on a large-scale model of border ownership, which generalized from training on four simple shapes to nearly 2000 shapes of varying scale and complexity, as well as a small demonstration on contours taken from natural images. The presented notion of ambiguity provides a way to process features that are locally ambiguous without the need for an explicit representation of the ambiguity to be present. Interactions between different scales of the network provide a way for lower-level neurons to dampen their activity. The effects of this dampened input propagate throughout the network, and the consequences of modulatory input result in low-level decisions affecting the impact of activation deeper in the network. The adaptive thresholds, combined with the short- and long-term weights of conflict learning, give the network a form of hysteresis useful for the varied timings of feedforward and feedback input. In the demonstrated model of border ownership, the thresholds were essential for learning similar responses to driving input while allowing differentiation over modulatory input.

The competitive column is likely a promising avenue for future work on challenging problems. The results suggest that increased competition combined with an increase in hierarchy depth could lead to the learning of complex features and potential applications beyond border ownership, such as object recognition.

## Acknowledgments

This work was supported by the National Science Foundation (grants CCF-1317433 and CNS-1545089), C-BRIC (one of six centers in JUMP, a Semiconductor Research Corporation program sponsored by DARPA), and the Intel Corporation. We affirm that the views expressed here are solely our own and do not represent the views of the U.S. government or any agency thereof.

## References

Arbelaez, P., Maire, M., Fowlkes, C., & Malik, J. (2011). Contour detection and hierarchical image segmentation. *IEEE Trans. Pattern Anal. Mach. Intell.*, *33*, pp. 898–916. doi:10.1109/TPAMI.2010.161

Bishop, C. M. (1995). *Neural networks for pattern recognition*. New York: Oxford University Press.

Blasdel, G. G., & Salama, G. (1986). Voltage-sensitive dyes reveal a modular organization in monkey striate cortex. *Nature*, *321*(6070), 579–585.

Bridson, R. (2007). Fast poisson disk sampling in arbitrary dimensions. In *Proceedings of the SIGGRAPH O7 ACM*, art. 22. New York: ACM.

Brincat, S. L., & Connor, C. E. (2006). Dynamic shape synthesis in posterior inferotemporal cortex. *Neuron*, *49*, 17–24.

Brosch, T., & Neumann, H. (2014). Interaction of feedforward and feedback streams in visual cortex in a firing-rate model of columnar computations. *Neural Networks*, *54*, 11–16.

Cheng, M.-M., Mitra, N. J., Huang, X., Torr, P. H., & Hu, S.-M. (2015). Global contrast based salient region detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *37*, 569–582.

Craft, E., Schütze, H., Niebur, E., & von der Heydt, R. (2007). A neural model of figure-ground organization. *Journal of Neurophysiology*, *97*, 4310–4326.

DiCarlo, J. J., Zoccolan, D., & Rust, N. C. (2012). How does the brain solve visual object recognition? *Neuron*, *73*, 415–434.

Dollár, P., Appel, R., Belongie, S., & Perona, P. (2014). Fast feature pyramids for object detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *36*, 1532–1545.

Felleman, D. J., & Van Essen, D. C. (1991). Distributed hierarchical processing in the primate cerebral cortex. *Cerebral Cortex*, *1*(1), 1–47.

Field, D. J. (1987). Relations between the statistics of natural images and the response properties of cortical cells. *JOSA A*, *4*, 2379–2394.

Fukushima, K. (1980). Neocognitron: A self-organizing neural network model for a mechanism of pattern recognition unaffected by shift in position. *Biological Cybernetics*, *36*, 193–202.

Grant, W. S., & Itti, L. (2018). Competitive column architecture source code. ilab.usc.edu/competitivecolumn/

Grant, W. S., Tanner, J., & Itti, L. (2017). Biologically plausible learning in neural networks with modulatory feedback. *Neural Networks*, *88*, 32–48.

Hawkins, J., Ahmad, S., & Cui, Y. (2017). Why does the neocortex have columns: A theory of learning the structure of the world. bioRxiv. doi:10.1101/162263

He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep residual learning for image recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (pp. 770–778). Piscataway, NJ: IEEE.

Heeger, D. J. (1992). Normalization of cell responses in cat striate cortex. *Visual Neuroscience*, *9*, 181–197.

Hubel, D. H., & Wiesel, T. N. (1974). Sequence regularity and geometry of orientation columns in the monkey striate cortex. *Journal of Comparative Neurology*, *158*, 267–293.

Isik, L., Meyers, E. M., Leibo, J. Z., & Poggio, T. (2013). The dynamics of invariant object recognition in the human visual system. *Journal of Neurophysiology*, *111*, 91–102.

Itti, L., Koch, C., & Niebur, E. (1998). A model of saliency-based visual attention for rapid scene analysis. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *11*, 1254–1259.

Jain, R., Millin, R., & Mel, B. W. (2015). Multimap formation in visual cortex. *Journal of Vision*, *15*(16), 3.

Kheradpisheh, S. R., Ghodrati, M., Ganjtabesh, M., & Masquelier, T. (2016). Deep networks can resemble human feed-forward vision in invariant object recognition. *Scientific Reports*, *6*, 32672.

Kirkpatrick, J., Pascanu, R., Rabinowitz, N., Veness, J., Desjardins, G., Rusu, A. A., . . . Hadsell, R. (2017). Overcoming catastrophic forgetting in neural networks. *Proceedings of the National Academy of Sciences*, *114*, 3521–3526. doi:10.1073/pnas.1611835114

Kogo, N., & van Ee, R. (2014). *Neural mechanisms of figure-ground organization: Border-ownership, competition and perceptual switching*. New York: Oxford University Press.

Layton, O. W., Mingolla, E., & Yazdanbakhsh, A. (2015). Neural dynamics of feed-forward and feedback processing in figure-ground segregation. *Frontiers in Psychology*, *5*, 972. doi:10.3389/fpsyg.2014.00972

Lillicrap, T. P., Cownden, D., Tweed, D. B., & Akerman, C. J. (2016). Random synaptic feedback weights support error backpropagation for deep learning. *Nature Communications*, *13276*. doi:10.1038/ncomms13276

Liu, W., Anguelov, D., Erhan, D., Szegedy, C., Reed, S., Fu, C.-Y., & Berg, A. C. (2016). SSD: Single shot multibox detector. In *Proceedings of the European Conference on Computer Vision* (pp. 21–37). Berlin: Springer.

Long, M., Cao, Y., Wang, J., & Jordan, M. (2015). Learning transferable features with deep adaptation networks. In *Proceedings of the International Conference on Machine Learning* (pp. 97–105).

Lowe, D. G. (1999). Object recognition from local scale-invariant features. In *Proceedings of the Seventh IEEE International Conference on Computer Vision* (pp. 1150–1157). Piscataway, NJ: IEEE.

Markov, N. T., Vezoli, J., Chameau, P., Falchier, A., Quilodran, R., Huissoud, C., . . . Kennedy, H. (2014). Anatomy of hierarchy: Feedforward and feedback pathways in macaque visual cortex. *Journal of Comparative Neurology*, *522*, 225–259.

Martin, A. B., & von der Heydt, R. (2015). Spike synchrony reveals emergence of proto-objects in visual cortex. *Journal of Neuroscience*, *35*, 6860–6870.

Mihalas, S., Dong, Y., von der Heydt, R., & Niebur, E. (2011). Mechanisms of perceptual organization provide auto-zoom and auto-localization for attention to objects. *Proceedings of the National Academy of Sciences*, *108*, 7583–7588.

Neyshabur, B., Bhojanapalli, S., McAllester, D., & Srebro, N. (2017). Exploring generalization in deep learning. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, & R. Garnett (Eds.), *Advances in neural information processing systems*, *30* (pp. 5949–5958). Red Hook, NY: Curran.

Nicholls, J. G., Martin, A. R., Wallace, B. G., & Fuchs, P. A. (2001). *From neuron to brain*. Sunderland, MA: Sinauer Associates.

Qiu, F. T., Sugihara, T., & von der Heydt, R. (2007). Figure-ground mechanisms provide structure for selective attention. *Nature Neuroscience*, *10*, 1492–1499.

Redmon, J., Divvala, S., Girshick, R., & Farhadi, A. (2016). You only look once: Unified, real-time object detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (pp. 779–788). Piscataway, NJ: IEEE.

Ren, S., He, K., Girshick, R., & Sun, J. (2015). Faster R-CNN: Towards real-time object detection with region proposal networks. In C. Cortes, N. D. Lawrence, D. D. Lee, M. Sugiyama, & R. Garnett (Eds.), *Advances in neural information processing systems*, *28* (pp. 91–99). Red Hook, NY: Curran.

Rubin, E. (1915). *Synsoplevede figurer*. Copenhagen: Gyldendal.

Russell, A. F., Mihalaş, S., von der Heydt, R., Niebur, E., & Etienne-Cummings, R. (2014). A model of proto-object based saliency. *Vision Research*, *94*, 1–15.

Sabour, S., Frost, N., & Hinton, G. E. (2017). Dynamic routing between capsules. In I. Guyon, U. V. Luxburg, H. Wallach, R. Fergus, S. Vishwanathan, & R. Garnett (Eds.), *Advances in neural information processing systems*, *30* (pp. 3859–3869). Red Hook, NY: Curran.

Sajda, P., & Finkel, L. H. (1995). Intermediate-level visual representations and the construction of surface perception. *Journal of Cognitive Neuroscience*, *7*, 267–291.

Sakai, K., & Nishimura, H. (2006). Surrounding suppression and facilitation in the determination of border ownership. *Journal of Cognitive Neuroscience*, *18*, 562–579.

Sakai, K., Nishimura, H., Shimizu, R., & Kondo, K. (2012). Consistent and robust determination of border ownership based on asymmetric surrounding contrast. *Neural Networks*, *33*, 257–274.

Serre, T., Oliva, A., & Poggio, T. (2007). A feedforward architecture accounts for rapid categorization. *Proceedings of the National Academy of Sciences*, *104*, 6424–6429.

Serre, T., Wolf, L., & Poggio, T. (2005). Object recognition with features inspired by visual cortex. In *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition* (vol. 2, pp. 994–1000). Piscataway, NJ: IEEE.

Stevens, J.-L. R., Law, J. S., Antolík, J., & Bednar, J. A. (2013). Mechanisms for stable, robust, and adaptive development of orientation maps in the primary visual cortex. *Journal of Neuroscience*, *33*, 15747–15766.

Sun, B., Feng, J., & Saenko, K. (2016). Return of frustratingly easy domain adaptation. In *Proceedings of the 30th AAAI Conference on Artificial Intelligence*. Palo Alto, CA: AAAI Press.

Teo, C., Fermuller, C., & Aloimonos, Y. (2015). Fast 2D border ownership assignment. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (pp. 5117–5125). Piscataway, NJ: IEEE.

Tsotsos, J. K., Culhane, S. M., Wai, W. Y. K., Lai, Y., Davis, N., & Nuflo, F. (1995). Modeling visual attention via selective tuning. *Artificial Intelligence*, *78*, 507–545.

Turrigiano, G. G. (2008). The self-tuning neuron: Synaptic scaling of excitatory synapses. *Cell*, *135*, 422–435.

von der Heydt, R. von der. (2015). Figure–ground organization and the emergence of proto-objects in the visual cortex. *Frontiers in Psychology*, *6*, 1695. doi:10.3389/fpsyg.2015.01695

Williford, J. R., & von der Heydt, R. (2016). Figure-ground organization in visual cortex for natural scenes. *eNeuro*, *3*(6). doi:10.1523/ENEURO.0127-16.2016

Zhang, Y., Meyers, E. M., Bichot, N. P., Serre, T., Poggio, T. A., & Desimone, R. (2011). Object decoding with attention in inferior temporal cortex. *Proceedings of the National Academy of Sciences*, *108*, 8850–8855.

Zhou, H., Friedman, H. S., & von der Heydt, R. (2000). Coding of border ownership in monkey visual cortex. *Journal of Neuroscience*, *20*, 6594–6611.

Received April 8, 2018; accepted September 24, 2018.