Structured hierarchical models for neurons in the early visual system

by

Brett Vintch

A dissertation submitted in partial fulfillment
of the requirements for the degree of
Doctor of Philosophy
Center for Neural Science
New York University
May 2013

Eero P Simoncelli, PhD

J Anthony Movshon, PhD

Dedication

To my parents, Dean and Joanne

Acknowledgements

The work presented in this thesis is the product of a tremendously supportive community of colleagues, friends, and family. I have benefited immensely from the training provided by a number of great scientists. Perhaps foremost I would like to thank my two advisors, Eero Simoncelli and Tony Movshon. To have received backing from either of them individually would have been more than sufficient, but their complementary traits and skills combined nonlinearly to great effect. Eero's intuitive, geometrical approach to math and science is fascinating and inspiring, and Tony's endless reserve of knowledge and ability to synthesize vast and varied information is enviable and instructive. I must also thank Justin Gardner for an incredible learning experience. He cannot be thanked enough for his great insights and generous support.

I have received just as much scientific support from my early-career colleagues, who are also my close friends. I am indebted to Yan Karklin, Umesh Rajeshanker, Atul Mallik, Pascal Wallisch, Yasmine El Shamayleh, Rob Young, and Romesh Kumbhani for their caring instruction, and to Deep Ganguli, Andrew Zaharia, Chaitu Ekanadham, Adam Weiss, and Christopher Shooner for Le Basket and Brooklyn.

I would like to express my gratitude to my friends and family for their consistent encouragement. Thanks to my parents and my sister for their frequent expression of confidence. Thanks to all of my climbing buddies across the globe for keeping me grounded, but not too grounded. This is especially true of Jill Cai, for her strong support on and off the rocks. Finally, special thanks to Joel Weitzman for sharing a parallel journey, Rohini Sen for the day-to-day support of a veteran flatmate, Chris Flory for a worldly belaying and leading partnership, and to everyone in the greater Hopkins crew for many, much-needed distractions.

Abstract

The early visual system is composed of a set of anatomically distinct areas that are linked together in a hierarchy. This structure uses simple rules at each stage but supports an impressive array of processing capabilities. In order to capture the full range of these computations, neuronal models in these areas should include this hierarchical architecture. Neurons in the earliest stages receive information directly from sensory transducers, yielding linear-like visual representations that are closely tied to visual stimulation. Neurons further downstream are more abstract and nonlinear in their representation, being both more selective for relevant stimulus visual and invariant across irrelevant features. Despite these computational differences, individual neurons among all areas are anatomically similar and they can be described in simple terms; inputs are summed across dendritic synapses and arbors and outputs are generated by a spiking nonlinearity in the soma and axon hillock. This regularity can be exploited to build simple but powerful hierarchical models that approximate the stages of visual processing in cortex.

A realistic model architecture can reduce, and in some cases eliminated altogether, the need for ad-hoc priors or regularizers. Incorporating physiological and anatomical constraints, and careful experimental design (including the choice of stimuli), simplifies models and allows for more direct and efficient estimation procedures. In this thesis I present a series of hierarchical models for neurons in the early visual system (V1 & V2) and show that they can accurately capture the computations performed by real neurons. I also demonstrate that a stage-wise structure avoids overfitting and that it allows for a more efficient estimation procedure than generic statistical models.

Contents

List of Figures

Introduction

This thesis presents a series of models that describe receptive fields and response properties of neurons in the first two cortical stages of visual processing, V1 and V2. Though the scope of the models are limited, due in part to the inclusion of properties that are specific to each area, the principles embodied in these models should be relevant to a wide range of neuroscientific inquiry. A common view of sensory cortex is that it is a hierarchy, using simple, local computations to accomplish increasingly sophisticated processing at each level. Our models mimic this hierarchical architecture, with stages that each mirror their anatomical counterpart. The model parameters are estimated so that they can reproduce the firing patterns of individual neurons, and we demonstrate that they outperform the current standard quantitative models. The stacked nonlinear structure of the models allows them to attain a variety of complex selectivities and invariances that would otherwise by impossible with single-stage models. This suggests a general framework for thinking about the transformation of signals between areas in the brain: simple operations performed in succession.

Background

Receptive field models for V1 neurons

Neurons in V1 are tuned for spatial position, local orientation, spatial frequency, and temporal frequency [1, 2, 3, 4]. Some V1 cells also have a preference for spatiotemporal phase and are selective for the specific patterns of light and dark regions in an image. The collection of these properties, usually measured with drifting sine wave gratings, are a type of descriptive model for neurons in V1. Each grating can be thought of as a single point in Fourier (spatiotemporal frequency) space, and the model can be 'estimated' for real neurons in this domain by mapping with an exhaustive search [5, 6]. Though this description is informative for many cells, it fails to give us much insight into how these properties are computed in the brain. Orientation selectivity in V1 is an emergent property that is not found in the preceding stages of visual processing. V1 receives its predominant feedforward input from the Lateral Geniculate Nucleus (LGN), whose neurons have spatially isotropic tuning properties; receptive fields have a center-surround spatial profile with bandpass frequency tuning [7, 8, 9], but no intrinsic preference for feature orientation due to their radial symmetry. How is it, then, that oriented receptive fields are generated from unoriented inputs?

Hubel & Wiesel were the first to describe orientation tuning in primary visual cortex. The receptive fields of simple cells, when mapped as a preference for light or dark spots of light as a function of spatial position, resemble Gabor-like edge filters with elongated excitatory and inhibitory lobes [1, 10]. Hubel & Wiesel were also the first to provide a plausible mechanism for its computation in the form of a qualitative model. They predicted that V1 simple cells could become selective for orientation by summing

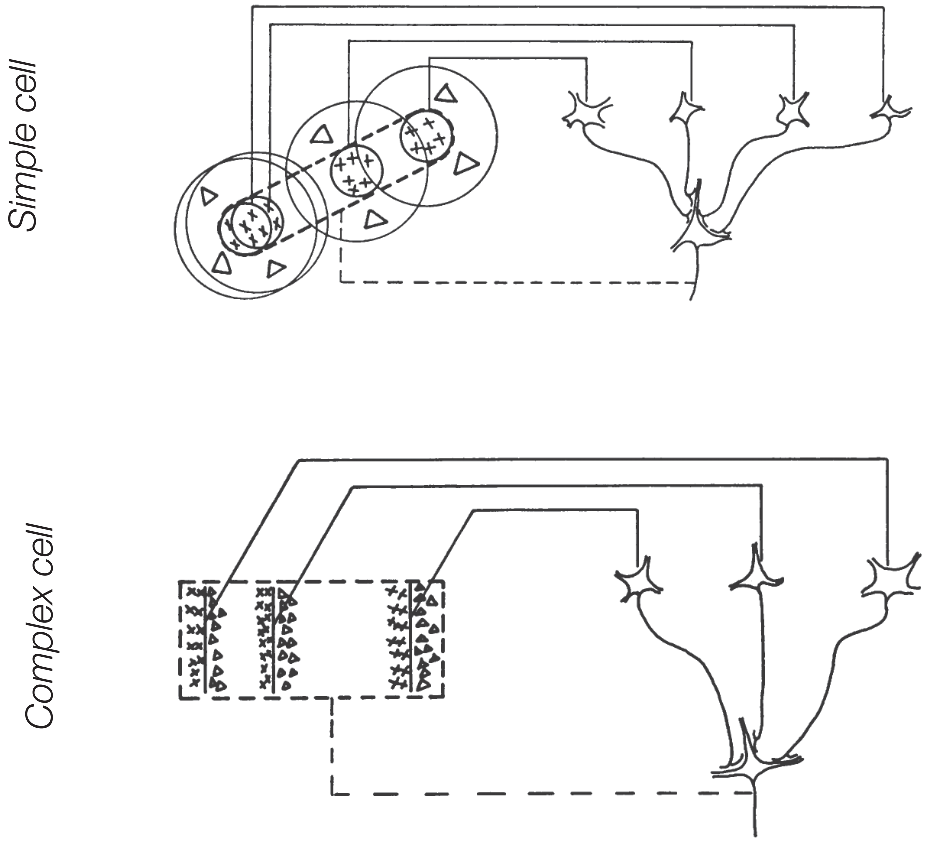


Figure 1: Hubel & Wiesel were the first to describe orientation selective neurons in V1. They speculated that phase-sensitive simple cells (top) generate orientation tuning by selectively pooling LGN afferents that lie along a line in space. Phase-invariant complex cells (bottom) then pool over an array of similarly tuned simple cells to generate a position invariant response that retains orientation selectivity. Receptive fields (left) are depicted in only their spatial domain. (Adapted from [1])

together a set of precisely aligned LGN cells (Figure 1, top). Though this qualitative model was not readily testable at the time, later simultaneous recordings in LGN and V1 were able to give experimental support for this conceptual model [11].

Not all neurons in V1 are selective for spatial phase; complex cells retain the orientation and spatiotemporal tuning of V1 simple cells but are not sensitive to contrast

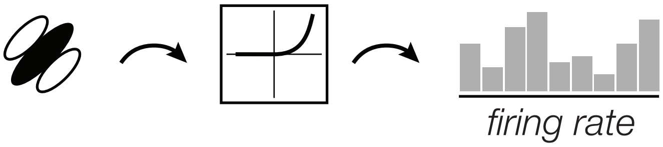


Figure 2: The Linear-Nonlinear model for V1 simple cells. Stimuli are first projected onto a linear filter. In general, the filter is 3-dimensional (two in space and one in time), but is depicted here in only two, such as one of space and one of time. The filter response is passed through an output nonlinearity to generate an average spike rate.

polarity or the precise location of edge-like elements within the receptive field envelope [6]. This response property cannot be generated through a linear combination of LGN afferents because any cell preferring an ordered white-black combination would be inhibited by a stimulus that was instead black-white. Hubel & Wiesel conjectured that a related pooling mechanism to simple cells could account for position invariance in complex cells. V1 cells that linearly pool over V1 simple cells, rather than directly pooling LGN afferents, could exhibit phase invariance if they sum over a family of overlapping, similarly tuned units (Figure 1, bottom). Together, these qualitative feedforward descriptions of simple and complex cells have provided an invaluable anchoring paradigm for vision research, but further progress requires testable mechanistic models.

The prototypical model for simple cells is the Linear-Nonlinear (LN) cascade, which operates by projecting the stimulus onto a fixed linear filter with a monotonic, rectifying spiking nonlinearity (Figure 2). Though the connection is slightly abstract, the model is closely related to Hubel & Wiesel's qualitative model. The processing stream from retina to V1 can be largely approximated as a single linear operation. LGN neurons pool over retinal ganglion cells, which in turn pool over an intricate retinal network, yet to a first-order approximation each of these stages is linear; individual retinal and

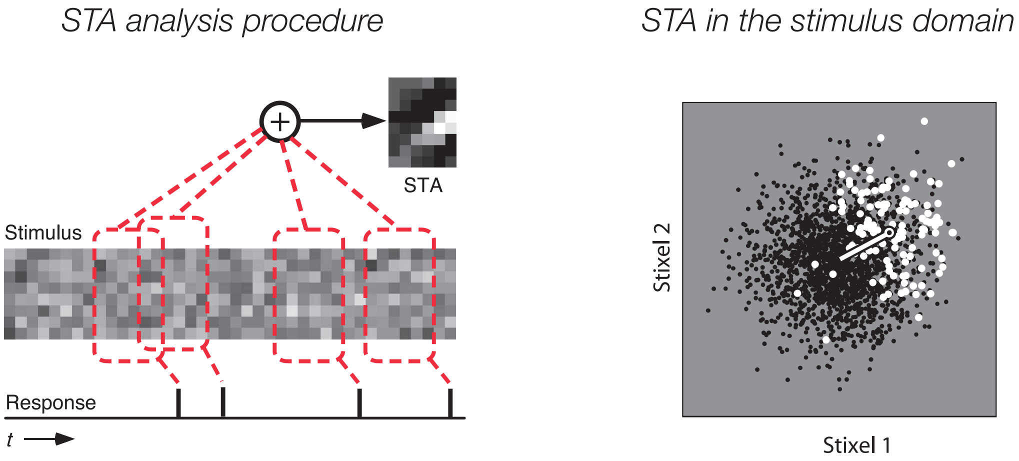


Figure 3: Spike-Triggered Averaging (STA) works by averaging together the stimuli in a window preceding the spikes. This produces a one-dimensional filter (left) that depicts the stimulus the optimal linear filter. We can also imagine this procedure in the stimulus domain, in which each stimulus window is characterized as a vector, or a point, in a high dimensional space (right). Each axis in this example is the contrast of a pixel within a particular window of time. The STA is the direction in this space where the stimulus projection optimally correlates with the observed spiking response. (Adapted from [12])

LGN neurons have half-wave rectifying responses, but on- and off-cell pairs can capture the full range of contrast values. Because cascades of linear operations can always be collapsed into a single linear operation, Hubel & Wiesel's model of pooling over LGN cells can be extended to the linear combinations of pixel values that create an oriented filter.

The linear parameters of simple cells can be estimated with Spike Triggered Averaging (STA). STA is an analysis technique that averages together all stimulus frames that precede a spike (Figure 3, left; see [12, 13, 14]), and for a cell with reasonable nonlinearities and probed with elliptically symmetric stimuli, the calculated STA filter will provide an unbiased estimate of the true linear receptive field [15]. It is often more informative to consider these types of analysis techniques in the stimulus domain, where we plot the raw distribution of stimuli and the spike-triggered distribution as points in the high-dimensional stimulus space (Figure 3, right). In this view, STA is simply the vector average of the spike-triggered stimulus ensemble [12]. Thus, STA is a subspace method that projects a high dimensional stimulus onto a single stimulus dimension, leaving the model blind to any stimulus deviations orthogonal to this direction.

Complex cells are not sensitive to any linear component of a stimulus; they respond equally well to pairs of a stimulus and it's contrast-reversed counterpart. Models for complex cells must account for this strong nonlinearity. They should also be sensitive to more than one stimulus dimension if they wish to capture position-invariance. The classic description of a complex cell that instantiates these two properties is the Adelson & Bergen 'Energy' model. It is a phenomenological model that combines the response of two oriented filters in phase quadrature [16]. Specifically, the responses of two filters separated in phase by 90 degrees are squared and added to produce a firing rate that is invariant to phase (Figure 4). This model is attractive for both theorists and physiologists because it is uncomplicated and could be plausibly implemented with real neural components. For example, a cell that dendritically adds the responses of four simple cells even spaced in phase, and with half-squaring output nonlinearities, will be completely phase invariant.

Like the LN model, the energy model is an example of a subspace approach; it is driven by activity in a low, two-dimensional stimulus projection. Unlike the LN model, the response is not driven by a monotonic function of those projections. Rather, the squaring operation that follows the linear filters acts to detect the *variance* of the projection from zero, both positive and negative, and precludes the model from responding linearly to any image features. This strong nonlinearity can make the model difficult to work with; it is not commonly fit to white-noise data because there is no

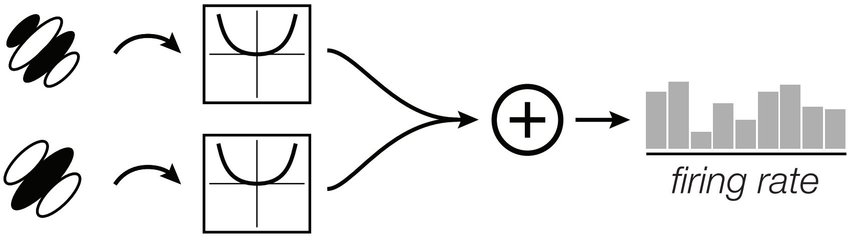


Figure 4: An energy model for complex cells. The response of two phase-quadrature filters are squared and added together to produce a response that is invariant to stimulus phase [16].

standard estimation procedure (though see [17] for a nice example of fitting this type of model to two-bar data, and Chapter 1 of this thesis in which we describe a procedure for fitting the model to data from white-noise experiments).

Real V1 cells are usually somewhere between completely phase invariant and completely phase selective, and models for these cells should be flexible enough to capture the full range of response types. The Rust-STC model is one successful hybrid example that blends together components of both the LN and energy models [18]. Figure 5 shows how an idealized version of this model generates a response by first passing stimuli through LN units with nonlinearities that are either half-squaring (linear response) or full squaring (energy response). Units are then pooled together into an excitatory channel or a suppressive channel, depending on their tuning properties, and the channels are combined with a joint nonlinearity to generate a firing rate. The output nonlinearity can typically accommodate suppression that is linear and divisive.

The parameters of the Rust-STC model are estimated with STA and STC (Spike-Triggered Covariance) analyses. As described above, STA captures the linear, phase-selective kernel of V1 receptive fields by computing the mean of the spike-triggered stimulus ensemble. STC works in an analogous manner, but computes the *covariance*

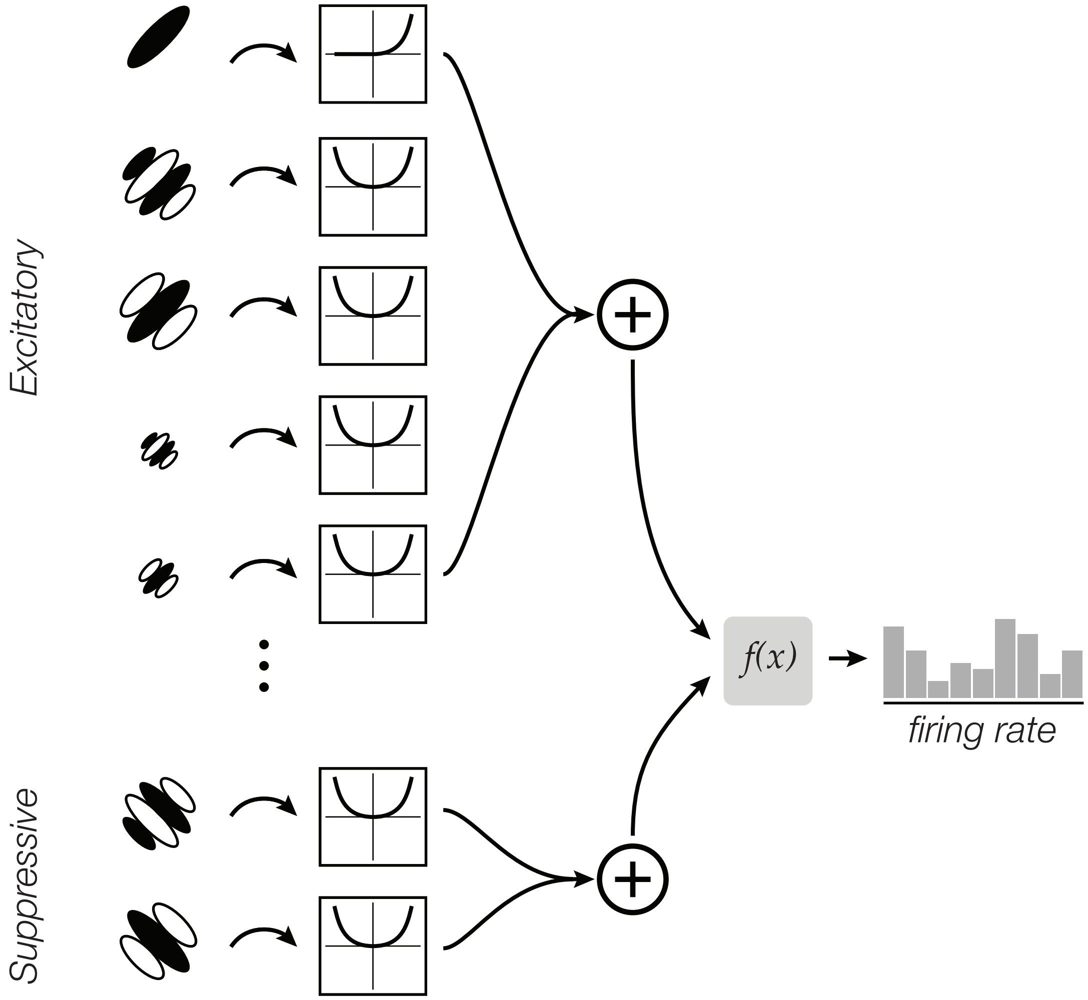


Figure 5: The Rust-STC model for neurons in V1. LN units are combined into an excitatory channel and a suppressive channel. A joint nonlinearity, $f(\cdot)$, converts the channel responses to a firing rate.

of the spike-triggered distribution [12, 18]. The eigenvectors of this covariance matrix describe the orthogonal axes, or filters, along which stimulus *variance* excites or suppresses spiking activity (Figure 6). The Rust-STC model is also a subspace model because a handful of filters usually suffice to describe the activity of most cells.

Though the Rust-STC model is equally adept at describing the responses of both simple and complex cells, the parameters of the model fit to complex-like cells are

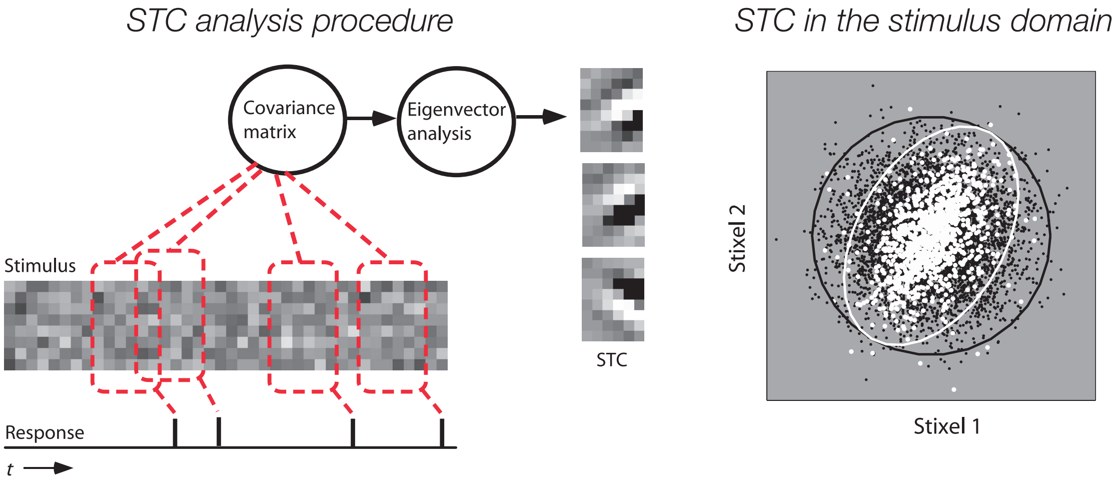
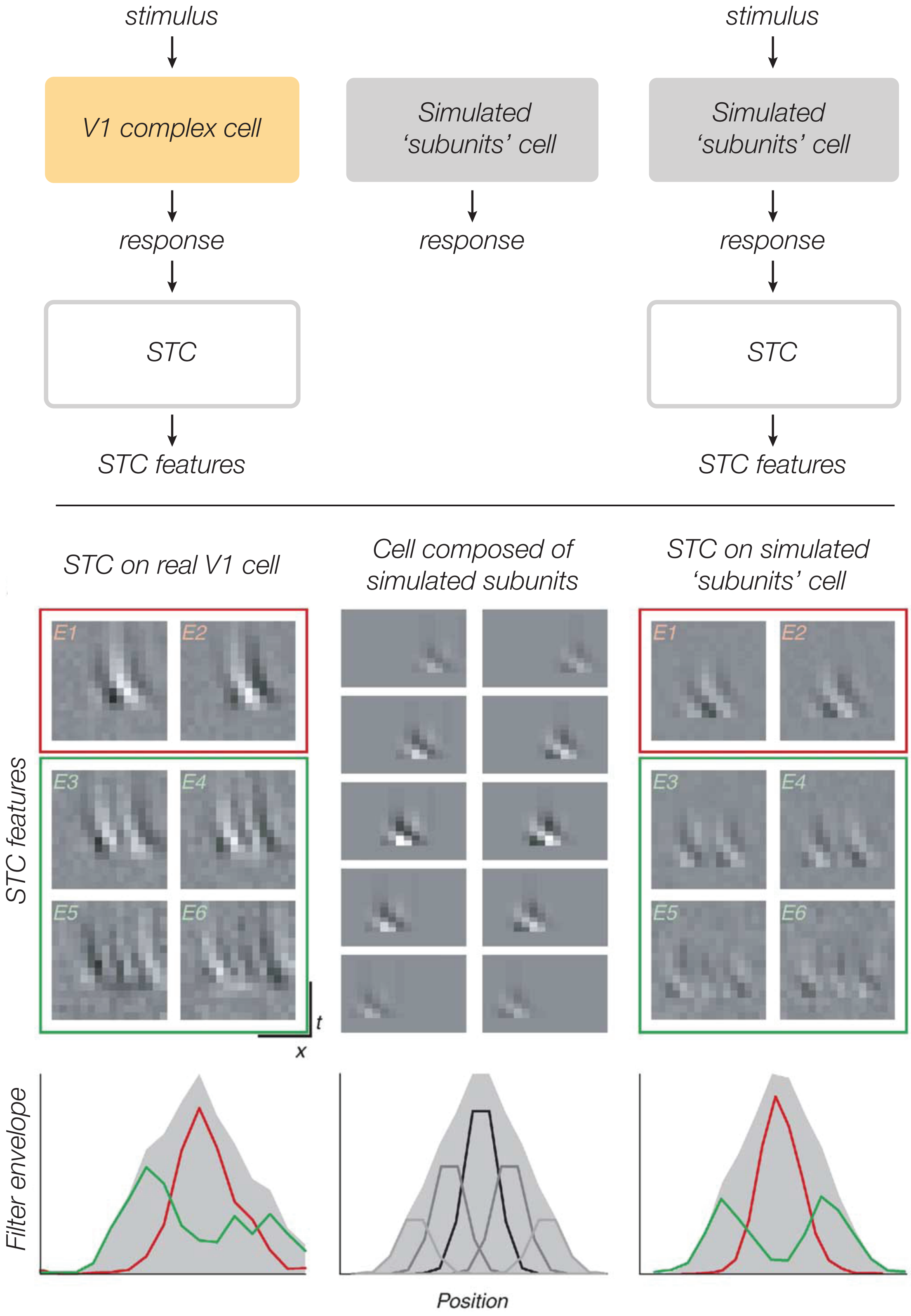


Figure 6: Spike-Triggered Covariance also operates on the spike-triggered stimulus ensemble. Rather than operating on the *mean* of the spike-triggered stimulus ensemble, STC operates on the covariance of this ensemble to produce a set of stimulus directions whose variance maximally excites or suppresses the cell. (Adapted from [12])

difficult to interpret in biological terms. The idealized portrait of a Rust-STC V1 model is depicted in Figure 5, with well localized filters that resemble realistic V1 simple cells. In reality, the model regularly produces filters that are unlocalized in both space-time and spatiotemporal frequency, due to the constraint that forces components to be orthogonal (see [18] and Chapter 1 of this thesis). This issue is not unique to the Rust-STC model, as it extends to many other types of orthogonal subspace methods

Figure 7 *(following page)*: The Rust-STC model on a simulated 'subunits' cell. [Left column] A complex cell is presented with a stimulus that is composed of flickering bars in space and time (xt). STC yields a series of quadrature-pair filters. Higher order filters in green are not localized in space-time or spatiotemporal frequency. [Middle column] Imagine that the complex cell is actually composed of simple filters, each just a shifted copy of some canonical oriented filter. [Right column] STC on the simulated subunit cell does not return the true subunits of the cell. Rather, the STC filters are consistent with those found for the real cell. Thus, it seems plausible that real complex cells could be composed of a set of shifted subunits. (Adapted from [18])


stimulus
V1 complex cell
response
STC
STC features
Simulated 'subunits' cell
response
stimulus
Simulated 'subunits' cell
response
STC
STC features
STC on real V1 cell
Cell composed of simulated subunits
STC on simulated 'subunits' cell
STC features
E1
E2
E3
E4
E5
E6
t
x
E1
E2
E3
E4
E5
E6
Filter envelope
Position

[19, 20, 21]. Nonetheless, Rust et al. were able to confirm that the results from their model were consistent with a 'subunit' architecture, which is a more realistic depiction of V1 receptive fields.

The classic Hubel & Wiesel model for a complex cell pools over a multiple copies of similarly tuned V1 simple cells. These simple cell afferents are termed the *subunits* of the complex cell receptive field, and they are formed by shifting identical copies of a linear filter over space to generate position invariance. Rust et al. ran an experiment to determine what would happen if they fit the Rust-STC model to a simulated cell that was actually composed of a series of shifted subunits (Figure 7). They found that the estimated filters for this simulation matched those observed for real cells, which suggests that though the firing patterns of real cells could be computed by a subunit computation [18], their model would not be able to return the true subunit filters.

Chapters 1, 2, and 3 of this thesis extend this concept into a direct subunit model that can be fit to V1 spiking data from reverse correlation experiments. We find that subunits are indeed an adequate description of the receptive fields across the entire response-type spectrum of neurons in V1. Including this architecture allows the model to avoid overfitting, and it provides a reliable substrate for further scientific investigation.

Receptive field models for V2 neurons

Visual area V2 is the largest cortical region in the brain, yet it's function is largely mysterious. It is located in inconvenient territory for scientific study; receiving it's predominant input from V1, it is too far removed from phototransduction to be closely tied to visual stimulation, and at several synapses from area IT, it is also too far from

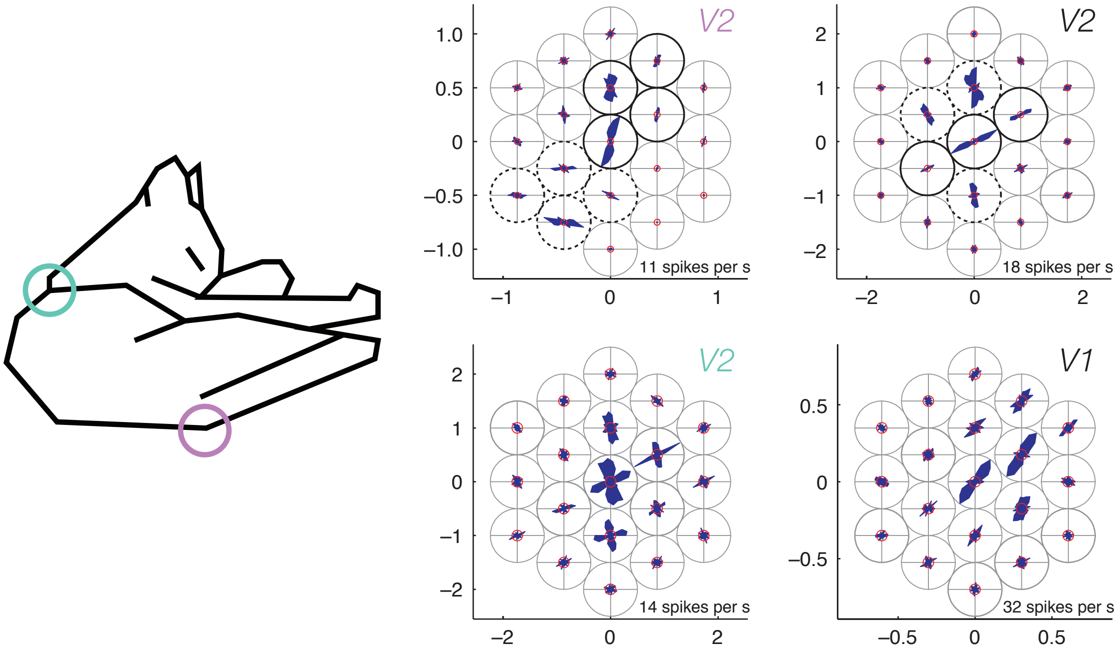


Figure 8: [Left] Attneave [22] demonstrated that image regions with high curvature are important for recognizing objects. If the high-curvature regions of an image are connected with straight lines, the image of a cat is still readily perceptible. [Right] Anzai et al. [23] probed neurons in V1 and V2 with hard-edged local drifting gratings. They found that some neurons in V2 had heterogenous orientation selectivity over space. Some neurons appeared to respond to angles or X- and T-junctions. Could neurons with these types of feature-selectivities underly object perception? (Adapted from [22] and [23])

object perception to be easily probed with behavioral tasks. Most attempts at receptive field characterization generally find that neurons in V2 have properties that resemble those in V1. However, a number of studies suggest that some neurons in V2 possess selectivities or invariances to image features beyond local oriented elements.

V1 receives center-surround afferents from LGN and uses them to generate selectivity to local orientation. Analogously, because V2 neurons pool over V1 inputs, it is tempting to imagine that V2 could have receptive fields that are selective to a combination of local edges, like angles or T-junctions. Perceptually, observers appear to be sensitive to image regions with either high curvature or conjunctions [24]. Attneave [22] noted that the edges with high curvature in an image are especially salient to observers, and simply connecting these regions with straight lines is enough to create an object that is readily identifiable (Figure 8, left). Hedge & van Essen [25], in an early V2 tuning experiment, found that some neurons responded selectively to angles and curves. Later, Anzai et al. [23] probed single units in V1 and V2 with *local* drifting sine-wave gratings and found a subset of neurons in V2, but not V1, that appeared to be selective to nonlinear contours over space (Figure 8, right). One could theorize that neurons of these kinds could be the link between V1 feature selectivity and object perception. Yet, only a small proportion of cells in V2 show this kind of response selectivity.

Another salient higher-order feature of images is texture boundaries. It is well known that observers are capable of detecting first-order, luminance-defined edges in images. However, observers are also very good at detecting second-order edges which can be defined as a difference in texture over space [27], such as the bark of a tree against background foliage (Figure 9). Von der Heydt & Peterson [26] found that some neurons in V2, but not V1, are selective to both first-order and second-order image features. For example, the neuron depicted in Figure 9 is selective for horizontal line segments, but also for horizontal illusory contours. A population of invariant V2 neurons such as these could underly the perception of texture boundaries. But again, only a small percentage of neurons display this type of interesting flexibility, and other studies fail to show a large difference in V2 at all [28]. So what does the rest of V2 do?

Fitting receptive field models for V2 is a complementary scientific program to the experiments described above. Those tuning experiments have a top-down design, first

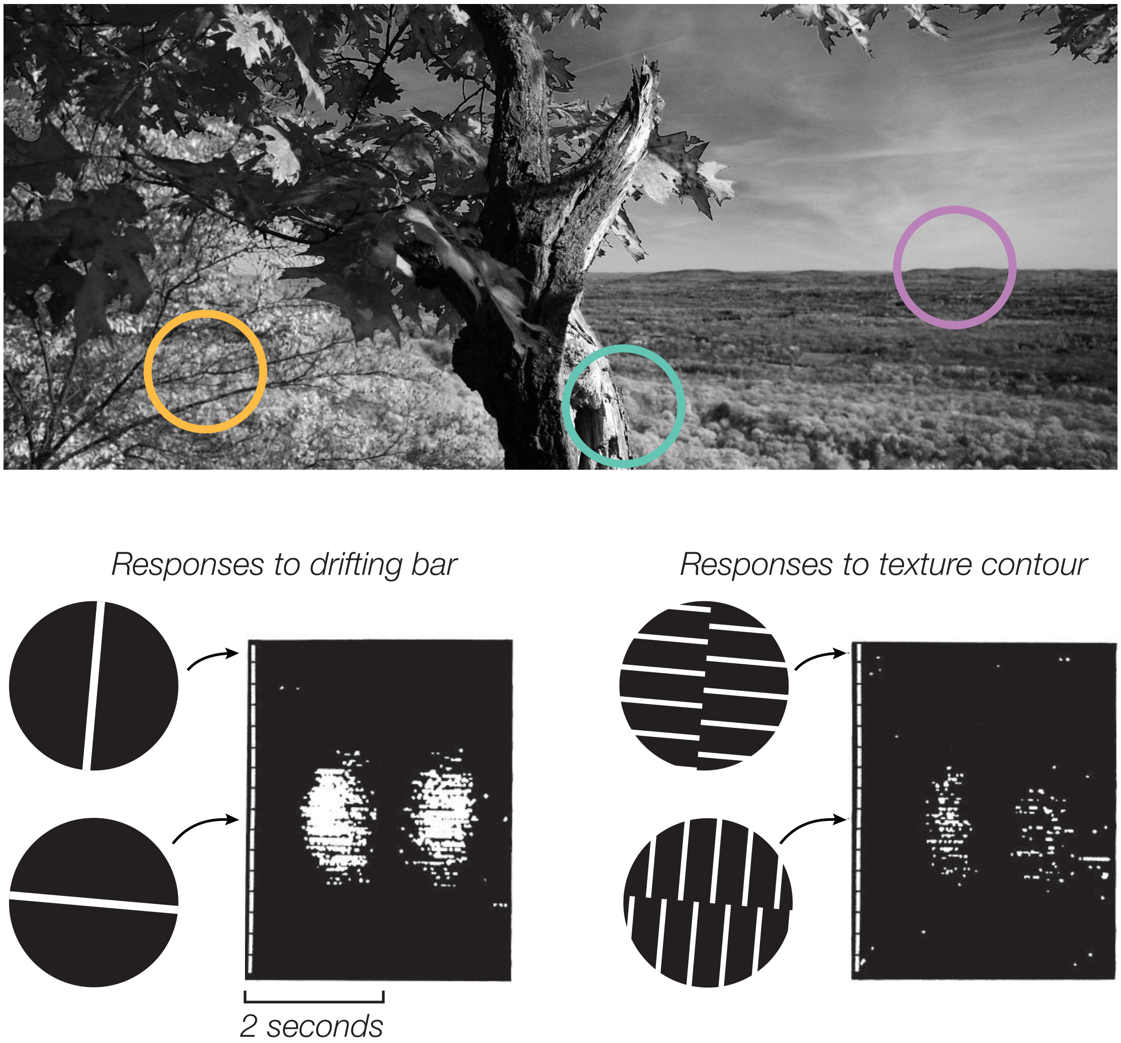


Figure 9: Images contain a variety of cues for object boundaries. Luminance boundaries (purple) and texture boundaries (blue) are both salient signals. Feature continuity (yellow) can also indicate which contrast edges belong to which objects. Von derHeydt and Peterhans [26] showed that some neurons in V2 are selective not only to luminance-defined edge orientation (left), but also texture-defined edge orientation (right). (Adapted from [26])

requiring a hypothesis about the set of visual or perceptual features that should selectively drive V2. The physiologist must then hunt for neurons tuned to these features, but all non-selective neurons remain outside the scope of understanding. Receptive

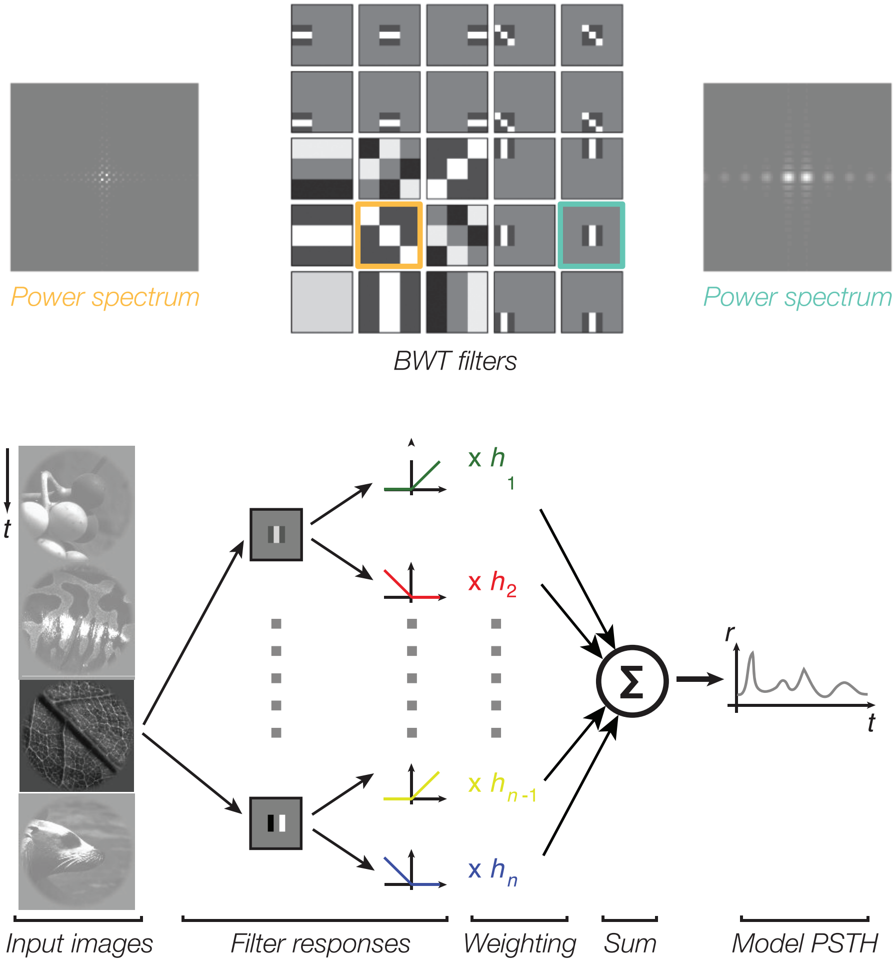


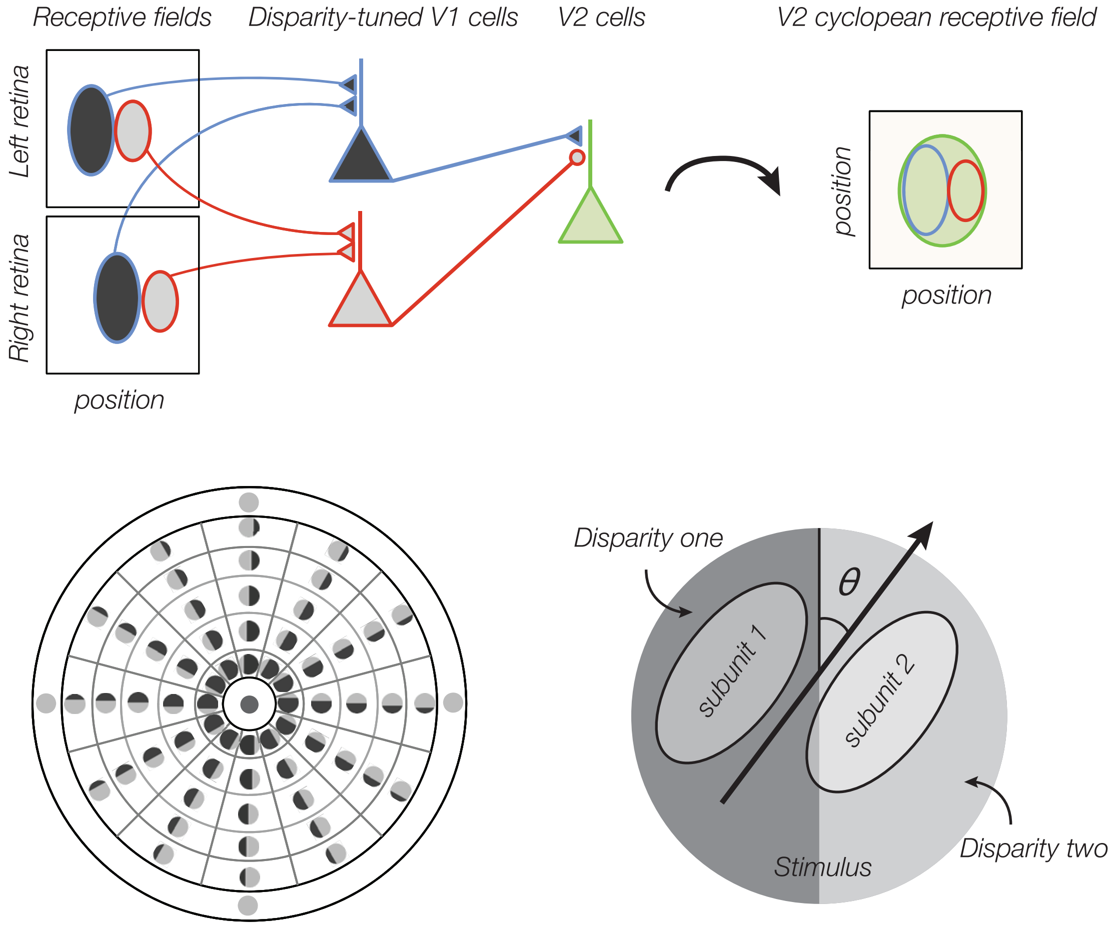
Figure 10: Willmore et al. [29] designed a hierarchical LN-L model to describe the firing rate of V2 neurons. Images are first passed through a set of orthogonal filters that are localized in space, but not frequency. A subset of filters are depicted at top. Two filters are also depicted, enlarged, in the Fourier domain, left and right. To generate a model response, the filter outputs are half-wave rectified, weighted, and summed (bottom). By fitting the weights, h_i, to spiking data for individual neurons they were able to derive wavelet-based depictions of the neuronal receptive fields. (Adapted from [29])

field modeling experiments usually have more of a bottom-up design. Starting from an agnostic position, they seek to understand the *mechanisms* that convert the photons that fall on the retina to firing rates in V2 before understanding the functionality. The ultimate goal is to find differences in the receptive fields of V1 and V2 neurons that could support novel classes of response properties.

A good example of this approach is the Willmore hierarchical model for V2 neurons [29]. Receptive fields are described as a linear combination of simple, half-rectified filter responses, and weights to each filter are fit with a sparse regression algorithm (Figure 10). The authors then try to determine if there is a difference in the type of filters that are pooled together by V1 and V2 cells. They find that V2 cells are more likely to have strong, tuned suppression than V1 cells. Yet, despite this success, the model front-end filters were chosen for engineering considerations [30] rather than for biological realism (Figure 10, top), which leaves the parameters of the model difficult to connect to real physiological properties.

A related study by Bredfeldt et al. builds models for V2 that are sensitive to disparity-defined depth edges [31]. Just as neurons in V1 and V2 are tuned to luminance edges, some neurons in V2 show a preference for depth edges created by a difference in horizontal disparity over space [32]. The Bredfeldt model pools over two simulated V1 units with independent parameters for position and disparity preference (Figure 11). These units receive their input from the stimulus parameters rather than computing them directly from the raw stimulus images. Nevertheless, the model is capable of capturing the subtle tuning properties of some neurons in response to oriented depth edges. In contrast to the Willmore model, the pooling function over V1 neurons is designed to be realistic. In fact, in adjusting the parameters of their model, Bredfeldt et al. describe how a purely linear pooling mechanism cannot explain the data as well as a pooling rule that includes a half-squaring intermediate output nonlinearity.

Chapter 4 of this thesis presents a new model for V2 receptive fields that shares some commonalities with the modeling approaches described above. Like Willmore et



Full stimulus ensemble
(disparity-defined edges)

Example receptive field
over an example stimulus

Figure 11: Many neurons in V2 are selective for horizontal disparity, and some are selective for disparity-defined edges. Bredfeldt et al. posited that a V2 neuron could generate this type of selectivity by pooling inputs from two V1 subunits (top), each with a different disparity and spatial position. They probed a population of neurons with a parameterized set of disparity-defined depth edges (bottom left) and fit a subunit model to the responses (bottom right). They found that the model could only explain the neuronal responses if the subunits were combined nonlinearly. Specifically, each subunit had to have a half-squaring nonlinearity. (Adapted from [31])

al. we describe spatial properties of V2 neurons by building a model that is defined by a sparse linear combination of simple LN units, and like Bredfeldt et al. we try to structure the model with realistic physiological elements. These insights allow us to construct a model that performs markedly better than other V2 receptive fields models. Because the parameters of the model are also easy to interpret in a biological context, we are able to find examples of neurons in V2 that are tuned to complex nonlinear spatial structure in images.

Part I

Models for V1

Chapter 1

Linear-Nonlinear cascade models for V1 receptive fields

1.1 Introduction

Advances in sensory neuroscience rely on the development of testable functional models for the encoding of sensory stimuli in neural responses. Such models require procedures for fitting their parameters to data, and should be interpretable in terms both of sensory function and of the biological elements from which they are made. The most common models in the visual and auditory literature are based on linear-nonlinear (LN) cascades, in which a linear stage serves to project the high-dimensional stimulus down to a one-dimensional signal, where it is then nonlinearly transformed to drive spiking. LN models are readily fit to data, and their linear operators specify the stimulus selectivity and invariance of the cell. The weights of the linear stage may be loosely interpreted as representing the efficacy of synapses, and the nonlinearity as a transformation from membrane potential to firing rate.

For many visual and auditory neurons, responses are not well described by projection onto a single linear filter, but instead reflect a combination of several filters. In the cat retina, the responses of Y cells have been described by linear pooling of shifted rectified linear filters, dubbed “subunits” [33, 34]. Similar behaviors are seen in guinea pig [35] and monkey retina [36]. In the auditory nerve, responses are described as computing the envelope of the temporally filtered sound waveform, which can be computed via summation of squared quadrature filter responses [37]. In primary visual cortex (V1), simple cells are well described using LN models [14, 38], but complex cell responses are more like a superposition of multiple spatially shifted simple cells [1], each with the same orientation and spatial frequency preference [6]. Although the description of complex cells is often reduced to a sum of two squared filters in quadrature [16], more recent experiments indicate that these cells (and indeed most 'simple' cells) require multiple shifted filters to fully capture their responses [18, 39, 21]. Intermediate nonlinearities are also required to describe the response properties of some neurons in V2 to stimuli (e.g., angles [40] and depth edges [31]).

Each of these examples is consistent with a canonical but constrained LN-LN model, in which the first linear stage consists of convolution with one (or a few) filters, and the first nonlinear stage is point-wise and rectifying. The second linear stage then pools the responses of these “subunits” using a weighted sum, and the final nonlinearity converts this to a firing rate. Hierarchical stacks of this type of “generalized complex cell” model have also been proposed for machine vision [41, 42]. What is lacking is a method for validating this model by fitting it directly to spike data.

A widely used procedure for fitting a simple LN model to neural data is reverse correlation [43, 44]. The spike-triggered average of a set of Gaussian white noise stimuli provides an unbiased estimate of the linear kernel. In a subunit model, the initial linear stage projects the stimulus into a multi-dimensional subspace, which can be estimated using spike-triggered covariance (STC) [45, 12]. This has been used successfully for fly motion neurons [46], vertebrate retina [47], and primary visual cortex [48, 18]. But this method relies on a Gaussian stimulus ensemble, requires a substantial amount of data, and recovers only a set of orthogonal axes for the response subspace—not the underlying biological filters. More general methods based on information maximization alleviate some of the stimulus restrictions [19] but strongly limit the dimensionality of the recoverable subspace and still produce only a basis for the subspace.

Here, we develop a specific subunit model and a maximum likelihood procedure to estimate its parameters from spiking data. We fit the model to both simulated and real V1 neuronal data, demonstrating that it is substantially more accurate for a given amount of data than the current state-of-the-art V1 model which is based on STC [18], and that it produces biologically interpretable filters.

1.2 The architecture of V1 receptive fields

We assume that neural responses are based on a linear summation of the responses of a set of nonlinear subunits. Each subunit operates by filtering the input (which can be either the raw stimulus, or the responses arising from a previous stage in a hierarchical cascade), and transforming the filtered response into a firing rate using a memoryless rectifying nonlinearity. The output at time t can be written as

$$\hat{r}(t) = \sum_i w_i \; f_{\boldsymbol{\Theta}} \left(\mathbf{k}_i^T \mathbf{x} \right) + \ldots + b, \tag{1.1}$$

where the $\mathbf{k}$'s are the subunit filters, f_Θ is a point-wise function parameterized by $\boldsymbol{\Theta}$

(e.g. a polynomial function, or a piecewise function), w_i are weights to the subunits, b is an additive baseline, and bold fonts represent vectors. The ellipsis indicates that we allow for multiple subunit channels, each with perhaps its own type of filter, nonlinearity, and pooling weights. Note that the nonlinearity, f, is not indexed by i, which indicates that the most simplified model assumes that all subunits share the exact same nonlinearity shape. We interpret $\hat{r}(t)$ as a 'generator potential', (e.g., time-varying membrane voltage) which is converted to a firing rate by another rectifying nonlinearity.

For the models that follow, we will fit parameters to data from reverse-correlation experiments and we will assume that neuronal firing is measured as a rate, binned at the same frequency as the stimulus. The response rate, $\mathbf{r}$, is a vector of T time points. The stimulus is a matrix in $\mathbb{R}^{T,P}$, with P stimulus parameters (such as pixels). For this section, stimulus parameters are generally just screen pixels, but the model is flexible, allowing for more abstract parameters derived from the stimulus if desired.

1.3 Linear-Nonlinear model

The subunit model defined in Eq. (1.1) may be seen as a general example of a subspace model, in which the input is initially projected onto a linear multi-dimensional subspace. In the simplest instantiation, there is only one subunit filter, leading to a one-dimensional subspace. This model is known as the Linear-Nonlinear, or LN, model. It produces a firing rate by filtering the stimulus with a linear kernel whose output is then passed through a nonlinearity that converts the membrane generator potential to a spike rate.

The standard approach to estimating the linear filter of the LN model uses Spike-

Triggered Averaging (STA), in which each frame of the stimulus in a causal time window is averaged with a weighted sum according to the observed spike count [44]. Under reasonable stimulus constraints and model assumptions, the STA filter will be the unbiased estimate of the cell's true linear component [15]. Unlike for the linear stage, there is no widely accepted approach to estimating the model nonlinearity. This is partly because there is no accepted canonical function for output nonlinearities, and partly because there are many ways to fit a point-wise nonlinear function once it has been described. We choose to define the function as piecewise-linear over 9 nodes that are chosen to span the entire response distribution and estimate the parameters with regularized least-squares regression. Specially, we encourage the nonlinearity to be smooth by penalizing the function's second derivative [49]. All three subsequent models will follow this convention for fitting an output nonlinearity.

1.4 Energy model

The energy model is the canonical description of complex cells in primary visual cortex [16]. It is another example of a subspace model, but projects the stimulus down to a four dimensional subspsace rather than one (only two dimensions are required if there is no suppressive channel). The responses of two direction-selective filters with differing phase preferences (typically, odd-symmetric and even-symmetric) are squared and summed to generate phase-invariant selectivity. Similarly, the summed and squared responses of a second pair of filters can be subtracted to capture suppressive selectivity. Although this model is the classic qualitative form for complex cells, there exists no standard method for fitting it in reverse correlation experiments (though see Emerseon et al. for a method that uses pairs of bar stimuli [17]). We develop a novel optimization procedure to fit such a model to spiking data which we describe in two dimensions (XT) - the methodology readily generalizes to three dimensions (XYT). The idea is to first find the single filter whose squared response best matches the observed firing rate in terms of mean-square error (MSE). We then find the direction preference of this filter and take the 2-dimensional Hilbert transform to obtain a paired quadrature filter.

The model response can be written as

$$\hat{r}(t) = \Big[(\mathbf{x}^T\mathbf{k})^2 + (\mathbf{x}^T\mathbf{k}_H)^2\Big] - \Big[(\mathbf{x}^T\mathbf{s})^2 + (\mathbf{x}^T\mathbf{s}_H)^2\Big], \tag{1.2}$$

where $\mathbf{k}$ and $\mathbf{s}$ are the excitatory and inhibitory filters, and the subscript H denotes a directional Hilbert transform. We wish to solve for the excitatory filter by gradient descent in the Fourier domain, and so we use Parsevals Theorem to make the following substitution:

$$\left(\mathbf{x}^T\mathbf{k}\right)^2 = \left(\tilde{\mathbf{x}}_r^T\tilde{k}_r + \tilde{\mathbf{x}}_i^T\tilde{k}_i\right)^2. \tag{1.3}$$

Here, $\tilde{\mathbf{x}}$ represents the Fourier transform of $\mathbf{x}$, and the subscripts r and i refer to the real or imaginary part respectively. Assuming a squared-error objective function, the gradient of the objective function with respect to the excitatory kernel is

$$\frac{d}{d\mathbf{k}} = 4\left(\hat{r} - r\right)\left(\tilde{\mathbf{x}}_r^T\tilde{k}_r + \tilde{\mathbf{x}}_i^T\tilde{k}_i\right)\tilde{\mathbf{x}}. \tag{1.4}$$

There is an analogous equation for the suppressive filter, s, and the optimization for both is performed simultaneously. Because of symmetry in the Fourier domain, we need only fit half of the coefficients.

To obtain the directional 2D quadrature filters, $\mathbf{k}_H$ and $\mathbf{s}_H$, we first must estimate the predominant orientation of each filter. The problem is formulated in the Fourier

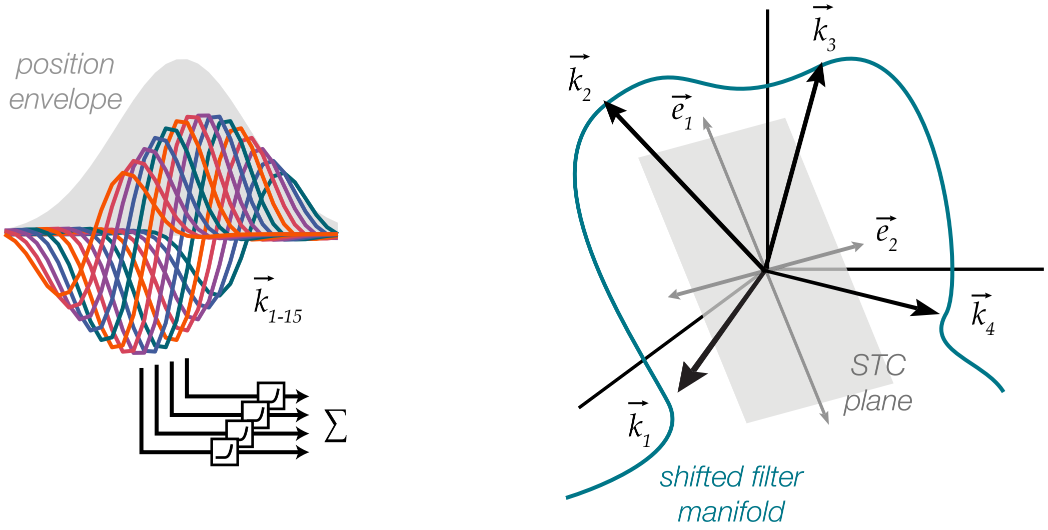


Figure 1.1: Spike-triggered covariance analysis for a hypothetical V1 complex cell. Left, the model output is formed by summing the rectified responses of multiple linear filter kernels which are shifted and scaled copies of a canonical form. Right, the shifted filters lie along a manifold in stimulus space (four shown), and are not mutually orthogonal in general. STC recovers an orthogonal basis for a low-dimensional subspace that contains this manifold by finding the directions in stimulus space along which spikes are elicited or suppressed.

domain as an orthogonal least-squares regression in which we seek the unit vector, $\mathbf{u}$, with the largest eigenvalue of the matrix $M^T M$, where M is power-spectrum weighted grid of Fourier frequencies. Specifically, each row of M is $|F(\omega_x, \omega_y)| \cdot [\omega_x,\ \omega_y]$. Then, the quadrature filter of k is $\mathcal{F}^{-1}[\tilde{\mathbf{k}}\mathbf{H}]$, where $\mathbf{H}$ is $0 + 1i$ for positive values of $\omega_{x,y}\mathbf{u}$ and $0 - 1i$ for negative values of $\omega_{x,y}\mathbf{u}$.

1.5 Spike-triggered covariance methods

Bialek and colleagues [45, 46] introduced spike-triggered covariance as a means of recovering an arbitrarily large multi-dimensional subspace of Eq. 1.1. Specifically, a generalized eigenvector analysis of the covariance matrix of the spike-triggered input ensemble exposes orthogonal axes for which the spike-triggered ensemble has a variance that differs significantly from that of the raw ensemble. These axes define a subspace, and may be separated into those along which variance is greater and those along which variance is smaller (excitatory and suppressive).

Rust et al. took this very general analysis procedure and built a model to convert stimulus images to firing rates [18]. The subspace axes are viewed as filters whose responses are half-squared (STA) or fully squared (STC), and they are weighted and added together into an excitatory channel, E, and a suppressive channel, S, depending on their variance. We use cross-validated least-squares fits to determine how many filters to use for each channel, and how to weight them. The two channels are then combined with a joint Naka-Rushton nonlinearity,

$$\hat{r}(t) = \alpha + \frac{\beta E^{\rho} - \delta S^{\rho}}{\gamma E^{\rho} + \epsilon S^{\rho} + 1}. \tag{1.5}$$

The functional form allows for suppression to work in both a subtractive and a divisive manner, and the parameters $\{\alpha, \beta, \delta, \gamma, \epsilon, \rho\}$ are fit to the data to minimize mean squared error.

Figure 1.1 demonstrates the geometry of the Rust-STC model applied to a simulated complex cell with 15 spatially shifted subunits. The simulated response is $\hat{r}(t) = \sum_i \lfloor w_i \left(\mathbf{k}_i \cdot \mathbf{x}(t)\right) \rfloor^2$, where the $\mathbf{k}$'s are shifted filters, w weights filters by position, and $\mathbf{x}$ is the stimulus vector (for this example $f_{\Theta}(\cdot)$ is implicitly defined as the half-squaring operation, $\lfloor \cdot \rfloor^2$). Note that the shifted filters are not orthogonal by construction. As a result, the recovered axes, the quantity of which depends on the amount of data collected, do not directly reflect the filters used to build the model (Figure 1.2). This is not a surprise: the recovered axes are forced to be orthogonal, and need only span the same subspace as the set of shifted model filters. This can be

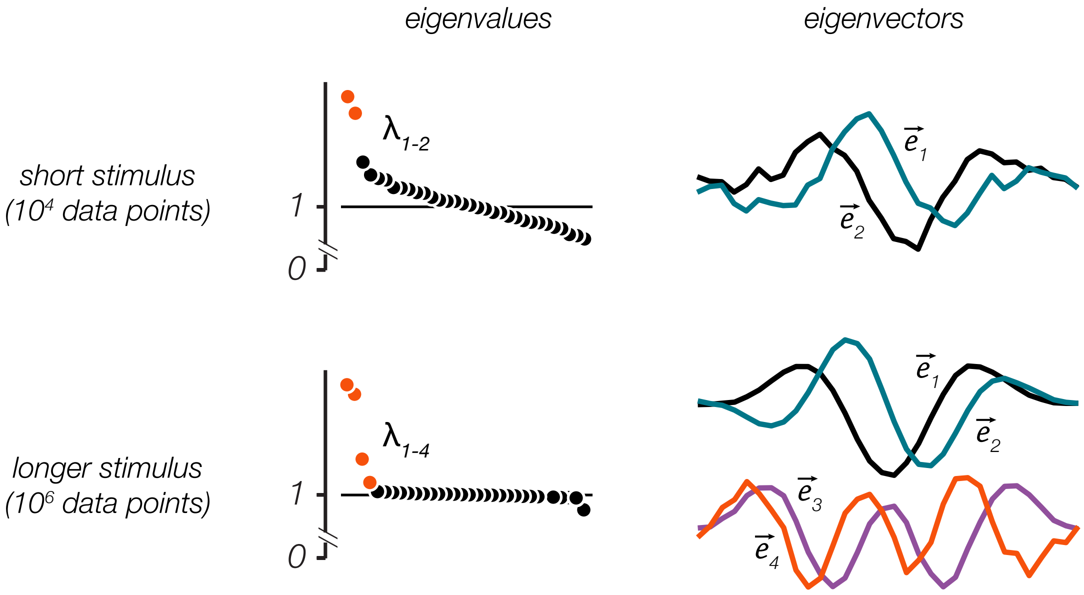


Figure 1.2: STC analysis of the simulated cell in Figure 1.1 returns a variable number of filters dependent upon the amount of acquired data. A modest amount of data typically reveals two strong STC eigenvalues (top), whose eigenvectors form a quadrature (90-degree phase-shifted) pair and span the best-fitting plane for the set of shifted model filters. These will generally have tuning properties (orientation, spatial frequency) similar to the true model filters. However, the manifold does not generally lie in a two-dimensional subspace [50], and a larger data set reveals additional eigenvectors (bottom) that serve to capture the deviations from the $\vec{e}_{1,2}$ plane. Due to the constraint of mutual orthogonality, these filters are usually not localized and they have tuning properties that differ from true model filters.

seen in data from V1 [18, 39]. Although one may follow the STC analysis by indirectly identifying a localized filter whose shifted copies span the recovered subspace [18, 21], the underlying STC method remains limited by the stimulus and data requirements discussed above.

1.6 A direct subunit model

A generic subspace method like STC does not exploit the specific structure of the subunit model. We therefore developed an estimation procedure explicitly tailored for this type of computation.

A critical simplification is that the subunit filters are related by a fixed transformation; here, we assume that they are spatially translated copies of a common filter, and the population of subunits can be viewed as computing a convolution. For example, the subunits of a V1 complex cell could be simple cells in V1 that share the same orientation and spatial frequency preference, but differ in spatial location, as originally proposed by Hubel & Wiesel [1, 6]. We also assume that all subunits use the same rectifying nonlinearity, further simplifying the model. We write the response to input defined over two spatial dimensions and time, $x(i,j,t)$, as,

$$\hat{r}(t) = \sum_{i,j,t} w_{i,j,t}\ f_{\Theta}\left(\sum_{m,n,\tau} k(m,n,\tau) \cdot x(i-m, j-n, t-\tau)\right) + \ldots + b, \qquad (1.6)$$

where the m and n indices implement the convolution operation.

Next, we first introduce a piecewise-linear parameterization of the subunit nonlinearity. Piecewise-linear functions can be implemented on a 1-dimensional signal by first decomposing the signal with a nonlinear basis set and then weighting the outputs of each basis function. The nonlinear basis resembles a series of triangles, or 'tents'. This parameterization of f is written as,

$$f(s) = \sum_{l} \alpha_l T_l(s), \qquad (1.7)$$

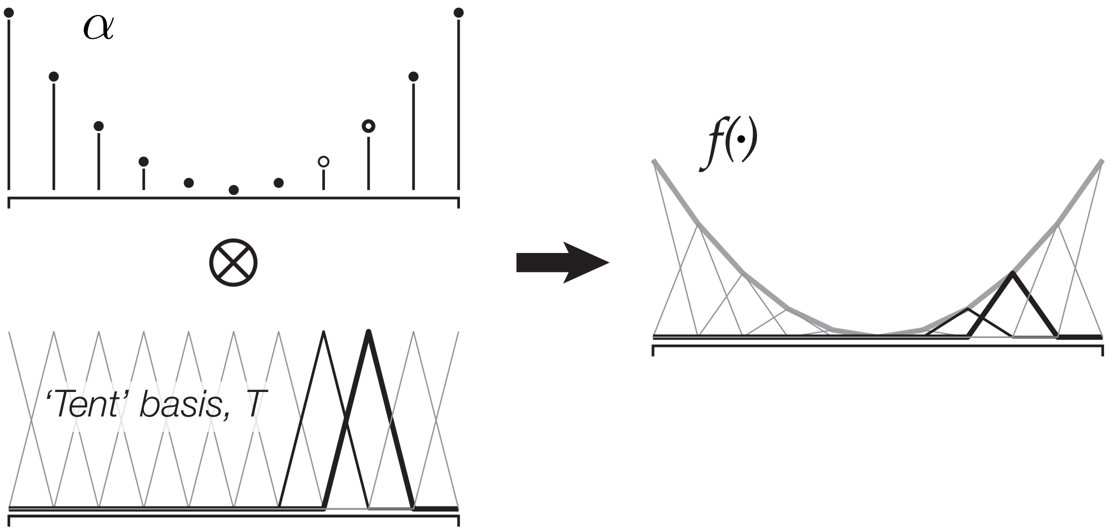


Figure 1.3: Construction of a piecewise nonlinearity. A 'tent' basis is a series of nonlinear triangle functions that tile one another; they sum to a constant value. Individually scaling these functions amounts to multiplying the tent basis by a series of α's and allows for the approximation of arbitrary nonlinearities with the resulting piecewise linear function. For example, a squaring nonlinearity can be constructed with the series of α's depicted here.

where the α's scale the small set of overlapping 'tent' functions, $T_l(\cdot)$, that represent localized portions of $f(\cdot)$ (Figure 1.3). We find that a dozen or so basis functions are typically sufficient to provide the needed flexibility. Incorporating this into the model response of Eq. (1.6) allows us to fold the second linear pooling stage and the subunit nonlinearity into a single sum:

$$\hat{r}(t) = \sum_{i,j,t,l} w_{i,j,t}\alpha_l \; T_l\left(\sum_{m,n,\tau} k(m,n,\tau)\cdot x(i-m, j-n, t-\tau)\right) + ... + b. \qquad (1.8)$$

The model is now partitioned into two linear stages, separated by the *fixed* nonlinear

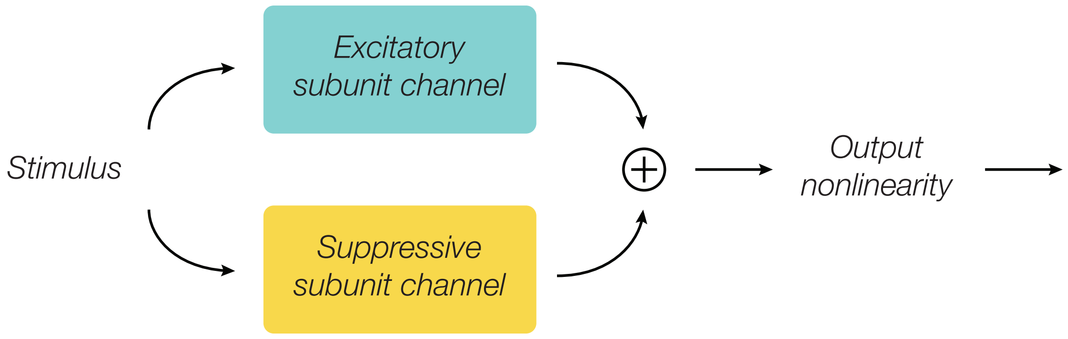


Figure 1.4: Subunit channels. The generic subunit model can accommodate an arbitrary number of channels. Each channel is composed of a bank of convolutional linear filters and nonlinearities along with a spatial pooling function. In this thesis, each subunit model that we build will have two channels: one excitatory and one suppressive. The two channels are linearly combined and passed through an output nonlinearity.

functions $T_l(\cdot)$. In the first partition, the stimulus is convolved with k, and in the second, the nonlinear responses are summed with a set of weights that are separable in the indices l and n, m. This formulation motivates the use of an iterative coordinate descent scheme: the linear weights of each portion are optimized in alternation, while the other portion is held constant. For each step, we minimized the mean square error between the observed firing rate of a cell and the firing rate predicted by the model. For models that include two subunit channels we optimize over both channels simultaneously (see section 1.7.4 for comments regarding two-channel initialization).

Schematics of the full subunit model are depicted in Figures 1.4 and 1.5. For the purpose of this thesis, each subunit model will have two channels, one excitatory and one suppressive, though in general the model can accommodate any number of channels. Example computations of each stage of the model for a simulated complex cell are shown in Figure 1.5, but for illustrative purposes we plot only a single, excitatory channel.

We also show the fit for a representative 2-channel model for a cell from V1. Figure

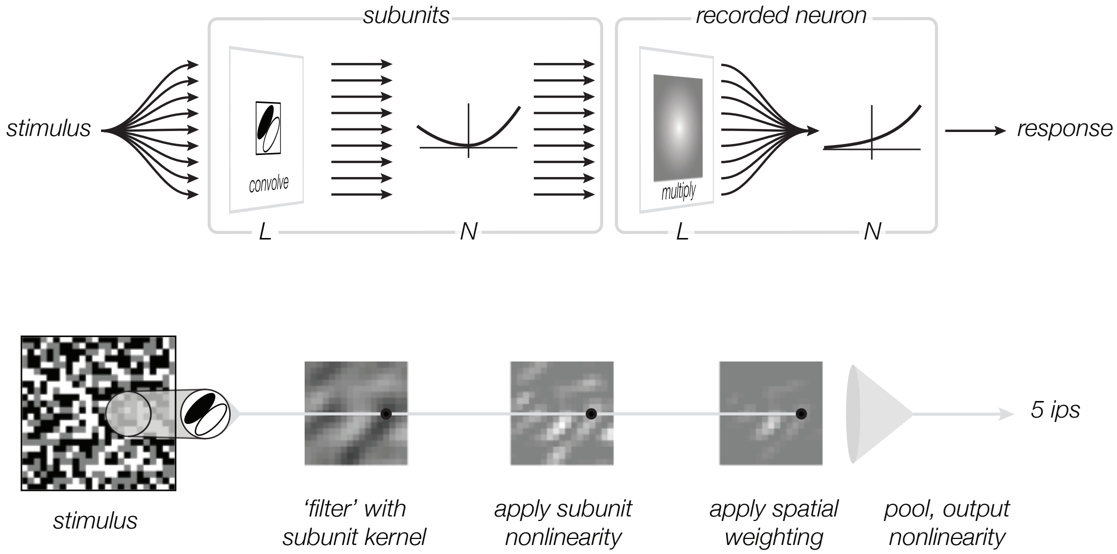


Figure 1.5: Subunit computations for a single channel. [Top] Diagram of a single subunit channel. The stimulus is convolved with a linear filter and each point is passed through an identical pointwise nonlinearity. The responses are weighted (here, over space only) and added together, before running through a final output nonlinearity. [Bottom] Computations performed by the model at each processing stage. Subsequent to the stimulus, the gray-scale value of each 'pixel' at each stage represents the activity of one hypothetical neuron. The stimulus is first convolved with a linear subunit filter. The output of each filter is then passed through the subunit nonlinearity. Finally, the subunits are weighted spatially, summed together, and the output is passed through a spiking nonlinearity.

1.6 illustrates the subunit kernels and their associated nonlinearities and spatial pooling maps, for both the excitatory channel (top row) and the suppressive channel (bottom row). The two channels show clear but opposing direction selectivity, starting at a latency of 50 ms. The fact that this cell is complex is reflected in two aspects of the model parameters. First, the model shows a symmetric, full-wave rectifying nonlinearity for the excitatory channel. Second, the final linear pooling for this channel is diffuse over space, eliciting a response that is invariant to the exact spatial position and phase of the stimulus.

1.7 Estimating the parameters of the subunit model

We optimized the parameters to minimize the mean square error between the observed firing rate of a cell and the firing rate predicted by the model. This choice of objective function implicitly assumes that the neural noise, or firing rate probability distribution, is Gaussian distributed. Write $\hat{r}_\Theta(\mathbf{x}_t)$ as the model prediction for the average firing rate for a particular cell in response to a stimulus $\mathbf{x}_t$ at time t (Θ is the vector of model parameters in Eq. 1.8 that includes $\mathbf{w}$,α,$\mathbf{k}$, and b). Then the probability of observing a vector of independent spike counts $\mathbf{r}$ over time is the product of the probability of observing each individual spike count,

$$p(\mathbf{r} \,|\, \hat{\mathbf{r}}_\Theta(\mathbf{x})) = \prod_{\mathbf{t}} \exp\left(\frac{-\left(\hat{\mathbf{r}}_{\boldsymbol{\Theta}}(\mathbf{x_t}) - \mathbf{r}(\mathbf{t})\right)^{\mathbf{2}}}{\mathbf{2}\sigma^{\mathbf{2}}} \right). \tag{1.9}$$

To estimate the model parameters Θ, we wish to maximize the probability of the model parameters with respect to the data, $\mathbf{X}$ and $\mathbf{r}$. To make this easier, we can take the negative logarithm of Eq. 1.9.

$$-\log p(\mathbf{r} \,|\, \hat{\mathbf{r}}_\Theta(\mathbf{x})) = \frac{\mathbf{1}}{\mathbf{2}\sigma^{\mathbf{2}}} \sum_{\mathbf{t}} \left(\hat{\mathbf{r}}_{\boldsymbol{\Theta}}(\mathbf{x_t}) - \mathbf{r}(\mathbf{t})\right)^{\mathbf{2}}, \tag{1.10}$$

leading us to the familiar metric of Mean-Squared Error (MSE). The argument that minimizes the negative log-likelihood objective function in Eq. 1.10 is the same as the argument that maximizes Eq. 1.9. Henceforth, we will drop the subscript Θ from $\hat{r}$ for notational clarity, but it is to be understood that this value represents the estimate of the firing rate given a set of specific model parameters.

To form the entire objective function, we substitute Eq. 1.8 into Eq. 1.10. We minimize the function by coordinate descent, alternately fitting the convolutional sub-

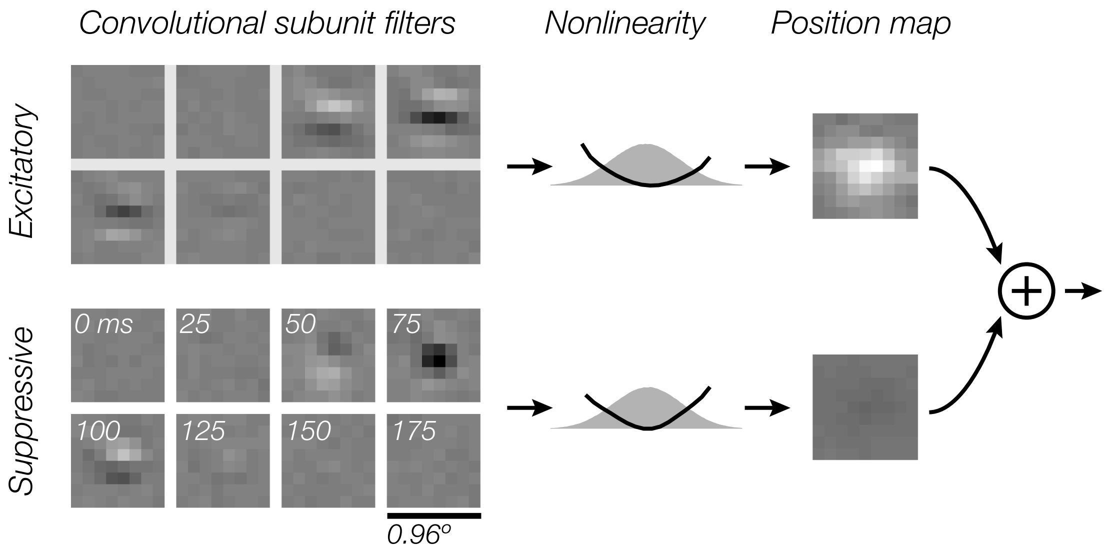


Figure 1.6: Two-channel subunit model fit to a representative cell in V1. Fitted parameters for the excitatory (top row) and inhibitory (bottom row) channel, including the space-time subunit filter (8 grayscale images correspond to different time frames), a point piece-wise nonlinearity, and a weighting function $w_{n,m}$ that is used to pool the subunit responses over space.

unit kernel and jointly fitting the subunit nonlinearity and the spatial pooling. These two steps are iterated until convergence:

Algorithm 1 Estimating the parameters of the subunit model

```
Initialize the model parameters:
$w \leftarrow w_0$, $\alpha \leftarrow \alpha_0$, $k \leftarrow k_0$, $b \leftarrow b_0$
while $-\log p(\mathbf{r} \mid \hat{\mathbf{r}}(\mathbf{x}))$ is decreasing do
    Fix $w$ and $\alpha$. Optimize subunit kernels, $k$, with gradient descent
    Fix $k$. Alternately update $w$ and $\alpha$ in closed-form with Ordinary Least Squares
end while
Fit an output nonlinearity to the entire function: $g(\hat{r})$
```

For models that include two subunit channels we optimize over both channels simultaneously.

At this point It is worth noting that real neurons do not exhibit Gaussian-distributed neural noise. For one, neurons cannot fire with a negative rate, and Gaussian distributions have infinite tails that include a prediction for negative firing. Moreover, there is much evidence that neuronal noise is better described as a Poisson distribution, and it may be still more complicated [51, 52]. Why then do we insist on using Mean Squared Error as our objective function? The subunit estimation algorithm hints at why this choice is advantageous; updating w and α with Ordinary Least Squares (OLS) only makes sense with a squared error loss function, and the ability to solve for these two parameters in closed-form is extremely attractive from an algorithmic standpoint. OLS is both less susceptible to local minima than comparable algorithms and faster to compute. We have experimented with other objective functions, such as one that assumes Poisson-distributed spike noise, but the resulting model fits are nearly identical to those computed with the MSE objective function, and it thus fails to justify the more complicated estimation process.

1.7.1 Estimating the convolutional subunit kernel

The first coordinate descent leg optimizes the convolutional subunit kernel, k, using gradient descent while fixing the subunit nonlinearity and the final linear pooling. We write the convolution operation between the stimulus and the kernel as a matrix multiplication of the stimulus with a circulant matrix and restrict this operation to the valid convolution region.

To optimize the convolutional kernel we perform gradient descent on the objective function (Eq. 1.10). Because we parameterize the nonlinearity as a linear combination of nonlinear basis functions, and this 'tent' basis is fixed, we can propagate the gradient easily. If we let S be the matrix of linear subunit responses at each time point,

$$S_{i,j,t} = \sum_{m,n,\tau} k(m,n,\tau) \cdot x(i-m, j-n, t-\tau), \tag{1.11}$$

then $d\mathbf{f}/dS$ is the derivative of the subunit nonlinearity with respect to the linear subunit responses. Calculating this derivative is trivial, as it only involves finding the slope of each line segment that composes f. With the chain rule,

$$\frac{d}{dk_{m,n,\tau}} = 2\left(\hat{r} - r\right) \left(\sum_{m,n} w_{i,j,t} \cdot \frac{df}{dS_{i,j,t}} \cdot x(i-m, j-n, t-\tau) \right). \tag{1.12}$$

This property also ensures that gradient descent is locally convex: assuming that updating k does not cause any of the the linear subunit responses to jump between the nodes of the tent functions representing f, then the derivative is linear and the objective function is quadratic. In practice, the full gradient descent path does cause the linear subunit responses to move slowly across bins of the piecewise nonlinearity. However, we include a regularization term to impose smoothness on the nonlinearity (see below) and this yields a well-behaved minimization problem for k.

1.7.2 Estimating the subunit nonlinearities and linear subunit pooling

The second leg of coordinate descent optimizes the subunit nonlinearity (more specifically, the weights on the tent functions, α_l), and the subunit pooling, $w_{i,j,t}$. As described above, the objective is bilinear in α_l and $w_{i,j,t}$ when k is fixed. Given that we assume the output noise of $\hat{r}(t)$ is Gaussian, standard bilinear estimation procedures can be used. Estimating both α_l and $w_{i,j,t}$ can be accomplished with Alternating Least

Squares (ALS), which assures convergence to a (local) minimum. In this coordinate descent procedure, we alternately and iteratively solve for w and then α in closed form with Ordinary Least Squares (OLS). Ahrens et al. [49] have described and discussed this procedure in great detail in the context of estimating input nonlinearities to one-dimensional time-varying signals.

The matrix of linear subunit responses S, in $\mathbb{R}^{i,j,t}$, is first passed through the fixed tent basis, T. This embedding creates a new matrix F of higher dimensionality, in $\mathbb{R}^{i,j,t,l}$. The advantage of this fixed nonlinear embedding is that the parameters of the nonlinearity, α, can now be applied linearly. In fact, w and α are combined within a single linear sum,

$$\hat{r}(t) = \sum_{i,j,t,l} w_{i,j,t}\alpha_l F_{i,j,t,l}, \tag{1.13}$$

and we can alternately solve for the remaining parameters by simply combining the linear functions. For example, if we want to optimize w, we can fold α into F to form a new matrix A:

$$\begin{aligned} &= \sum_{i,j,t} w_{i,j,t}\left(\sum_{l} \alpha_l F_{i,j,t,l}\right) \\ &= \sum_{i,j,t} w_{i,j,t}\mathbf{A}_{i,j,t}. \qquad (1.14) \end{aligned}$$

We then use Ordinary Least Squares to solve for w, which is fast, efficient, and amenable to regularization.

Alternate Least Squares is an example of a coordinate-descent optimization procedure, and it is not guaranteed to obtain a global minimum. Ahrens et al. [49] report that for their simulations of input nonlinearities in auditory neurons, it is rare to observe multiple local minima. Although one cannot be sure that a result is a global minimum, the optimization procedure at multiple starting points should generate the same solution. We use this method with our data in primate visual cortex and find a similar insensitivity to model initialization.

1.7.3 Regularization

There is no guarantee that the full coordinate descent algorithm will reach a global minimum, though we can ensure reasonable solutions through a variety of regularization methods. The subunit kernels, nonlinearities, and pooling functions can be regularized separately. We find that the most important regularizer is a prior for smoothness in the nonlinearity.

The subunit nonlinearity f is encouraged to be smooth by penalizing the second derivative of the function in the least-squares fit [49]. This smoothness helps to guarantee that the optimization of k is well behaved, even where finite data sets leave the function poorly constrained. Recall that the optimization of w requires the propagation of the gradient through the subunit nonlinearity. If the nonlinearity is jagged, the gradient steps will cause this value to careen wildly at each iteration, but if it is smooth, w is less sensitive to gradient steps that push the function between the nodes of the nonlinearity.

We also include a cross-validated ridge prior for the pooling weights to bias $w_{i,j,t}$ toward zero. The subunit responses are correlated because the subunit filters overlap one another in space and time. The Ridge prior helps to keep the pooling weights smooth over space by reducing the high-frequency terms that come from the inversion of the subunit covariance matrix in OLS. The filter kernel k can also be regularized to ensure smoothness, but for the examples shown here we did not find the need to include such a term.

1.7.4 Model initialization

Our objective function is non-convex and contains local minima, so the selection of initial parameters may affect the value of the solution and the rapidity of convergence. We found that initializing our two-channel subunit model to have a positive pooling function for one channel and a negative pooling function for the second channel allowed the optimization of the second channel to proceed much more quickly. This is probably due in part to a suppressive channel that is much weaker than the excitatory channel in general. We initialized the nonlinearity to halfwave-rectification for the excitatory channel and fullwave-rectification for the suppressive channel.

To initialize the convolutional filter we use a novel technique that we term 'convolutional STC'. The subunit model describes a receptive field as the linear combination of nonlinear kernel responses that spatially tile the stimulus. Thus, the contribution of each localized patch of stimulus (of a size equal to the subunit kernel) is the same, up to a scale factor set by the weighting used in the subsequent pooling stage. As such, we compute an STC analysis on the union of all localized patches of stimuli. Ignoring time for simplicity, for each subunit location, $\{i, j\}$, we extract the local stimulus values in a window, $g_{i,j}(m, n)$, the size of the convolutional kernel and append them vertically in a 'local' stimulus matrix. As an initial guess for the pooling weights, we weight each of these blocks by a Gaussian spatial profile, chosen to roughly match the size of the receptive field. We also generate a vector containing the vertical concatenation of copies of the measured spike train, $\vec{r}$ (one copy for each subunit location).

$$\begin{pmatrix} w_{1,1}X_{g_{1,1}(m,n)} \\ w_{1,2}X_{g_{1,2}(m,n)} \\ \vdots \end{pmatrix} \rightarrow X_{loc}\ ; \qquad \begin{pmatrix} \vec{r} \\ \vec{r} \\ \vdots \end{pmatrix} \rightarrow \vec{r}_{loc}. \tag{1.15}$$

After performing STC analysis on the localized stimulus matrix, we use the first (largest variance) eigenvector to initialize the subunit kernel of the excitatory channel, and the last (lowest variance) eigenvector to initialize the kernel of the suppressive channel. In practice, we find that this initialization greatly reduces the number of iterations, and thus the run time, of the optimization procedure.

1.8 Model validation

We estimate each model on training data and validate each model with testing data. It is important to perform this second step because the spiking data that is used to fit the model contains both a stimulus-driven component (the signal) and random fluctuations (the noise). Models that are under-constrained (i.e. they have too many parameters for the amount of collected data) tend to find the noise as well as the signal, showing high accuracy for the training data but failing to generalize to the testing data.

For models trained and validated on XT data we perform 5-fold cross-validation. That is, we take ${}^{4}/_{5}$ ths of the data and use it to train the model and then test the fit on the remaining ${}^{1}/_{5}$ th of the data. We do this five times and average the results over all five sets (the tranches of data are taken randomly over time and do not correspond to the first fifth of data in time, etc). Performance is measured as the correlation coefficient between the actual firing rate and the predicted firing rate, and over-fitting can be assessed by comparing the training performance to the test performance; the former should never exceed the later. Though valuable for this data set, this form of cross-validation is relatively cumbersome because the model must be estimated five times in succession.

For models trained and validated on the XYT data we perform a different kind of validation that is more efficient. For a subset of cells we collected responses to 20 repeats of a novel, 25-second stimulus. We then test the fitted model on each repeat and take the average correlation coefficient over all 20 trials.

These validation methods are valuable because they allow us to compare performance across models and they give us an idea of how much each model is over-fitting, but they cannot tell us how well the models are performing on an absolute scale. Each recorded spike rate includes many sources of noise (Poisson-like spiking statistics, input-noise, measurement-noise, etc), and this noise creates a performance ceiling above which no stimulus-driven model could hope to achieve. We estimate this ceiling and compare our models to this theoretical maximum.

We call the ceiling on stimulus-driven model performance the *oracle* prediction, and it works by seeking to minimize the effects of noise. Because our measure of model performance is the correlation coefficient, or mean-square error, we are sensitive only to errors of the first and second moments: bias and variance. In order to perform well, an oracle prediction needs to average out the trial-by-trial noise and arrive at the best prediction for the response *mean*. Specifically, to predict the response to trial n, we average together the responses of all of the other trials. Many neurons are very unreliable, and their trial-to-trial variability is high. For these neurons, the oracle is expected to perform poorly, and so is every stimulus-driven model. By comparing our models to this theoretical maximum, we can get a sense for how much of the explainable variance each model is capturing.

1.9 Simulations

Initially, we use simulated V1 cells to compare the performance of the subunit model to that of the Rust-STC model [18], which is based upon STC analysis. We simulated the responses of canonical V1 simple cells and complex cells in response to white noise stimuli. Stimuli consisted of a 16x16 spatial array of pixels whose luminance values were set to independent ternary white noise sequences, updated every 25 ms (i.e. 40 Hz). The simulated cells use spatiotemporally oriented Gabor filters: The simple cell has one even-phase filter and a half-squaring output nonlinearity while the complex cell has two filters (one even and one odd) whose squared responses are combined to give a firing rate. Spike counts are drawn from a Poisson distribution, and overall rates are scaled so as to yield an average of 40 ips (i.e. one spike per time bin).

For consistency with the analysis of the physiological data (XYT), we fit the simulated data using a subunit model with two subunit channels (even though the simulated cells only possess an excitatory channel). When fitting the Rust-STC model, we followed the procedure described in [18]. Briefly, after the STA and STC filters are estimated, they are weighted according to their predictive power and combined in excitatory and suppressive pools, E and S (we use cross-validation to determine the number of filters to use for each pool). These two pooled responses are then combined using a joint output nonlinearity: $\hat{r}(t)_{Rust} = \alpha + (\beta E^{\rho} - \delta S^{\rho})/(\gamma E^{\rho} + \epsilon S^{\rho} + 1)$. Parameters $\{\alpha, \beta, \delta, \gamma, \epsilon, \rho\}$ are optimized to minimizing mean squared error between

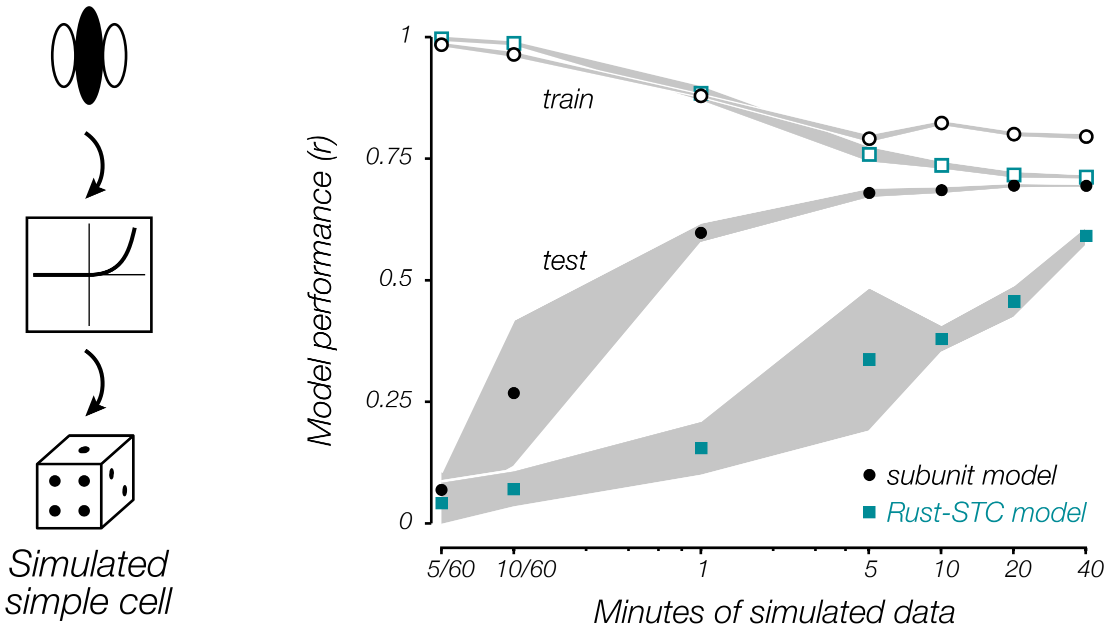


Figure 1.7: Model fitting performance for simulated V1 simple cell with Poisson spike noise. Shaded regions are ± 1 s.d. for 5 random stimulus ensembles. The time scales are derived from a simulated presentation rate of 40 Hz. Simulated simple cell was constructed from a single oriented filter with a half-squaring output nonlinearity that averaged 40 spikes per second (e.g. one spike per simulated time bin).

observed spike counts and the model rate.

Model performances, measured as the correlation between the model rate and spike count, for an example simple cell is shown in Figure 1.7. In low data regimes, both models perform nearly perfectly on the training data, but poorly on separate test data not used for fitting, a clear indication of over-fitting. But as the data set increases in size, the subunit model rapidly improves, reaching near-perfect performance for modest spike counts. The Rust-STC model also improves, but much more slowly; It requires more than an order of magnitude more data to achieve the same performance as the subunit model.

This inefficiency is more pronounced for the complex cell. Figure 1.8 shows how

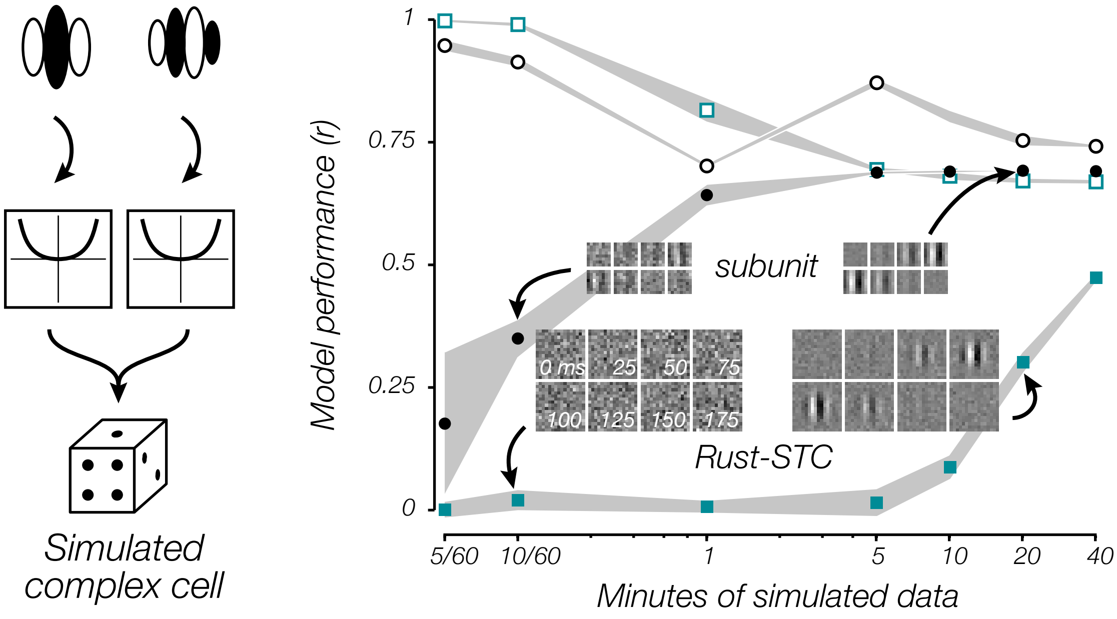


Figure 1.8: Figure details are as in 1.7. The complex cell was constructed from a sum of squared Gabor filters arranged in (sine and cosine) spatial quadrature and averaged 40 spikes per second. Image insets show the parameters for the subunit and Rust-STC model with two amounts of data.

the subunit model and the Rust-STC model improve as a function of data size for a simulated complex cell. The rate of improvement for the simple cell and the complex cell is largely the same for the subunit model. However, the Rust-STC model is more sluggish to improve for the complex cell. This is because the simple cell is fully explained by the STA filter, which can be estimated much more reliably than the STC filters for small amounts of data.

The asymmetry in performance for the Rust-STC between cell types suggest that the model fits will be biased under normal experimental conditions. Real experiments are constrained by the need to maintain cell isolation over an extended period of time; under average experimental conditions, a recording only lasts for about an hour or two. Simple cells are more likely to be accurately characterized by the Rust-STC model, and the model fits for complex cells will be biased towards phase-sensitive versions for short physiological recordings. We conclude that directly fitting the subunit model is much more efficient in the use of data than using STC to estimate a subspace model.

Chapter 2

Spatiotemporal subunit models (XT)

2.1 Introduction

The response properties of neurons in primary visual cortex (V1) are generally described using orientation-selective linear filters. Simple cells combine ON and OFF afferents arriving from the lateral geniculate nucleus (LGN) [1, 11]; the resulting behavior can be captured using a cascade of a linear filter and a rectifying nonlinearity in an LN cascade [14, 53]. Complex cell responses can be described by summing the responses of a set of simple cells with identical orientation tuning, but differing in spatial position or phase [1, 16].

This simple-complex dichotomy has been recently brought into question [54], and use of new "subspace" characterization methodologies [45, 19, 20] have shown that the responses of individual V1 cells may be fit by a number of linear filters that can vary from one to more than a dozen [18, 39]. Although such models provide a useful generalization of the conventional simple and complex cell descriptions, the large set of free parameters makes the corresponding estimation procedures particularly noise sensitive. An additional effect of these methods is that the extracted filters are forced by the procedure to be orthogonal, and as a result they generally do not resemble shifted copies of one another. Although the resulting orthogonal set can be transformed into an equivalent shifted set [18, 21], this transformation can further exacerbate the noise sensitivity of the solution.

Here, we use the general model for V1 neurons along with a direct method for fitting this model to spiking data that was developed in Chapter 1. The model is composed of two channels (one excitatory, one suppressive), each formed by summing a set of LN subunits, as has been used in modeling the so-called Y ganglion cells in rabbit and cat retina [33, 55], and similar in concept to Hubel and Wiesels original description of complex cell responses as resulting from a spatial combination of simple cells. To constrain the model complexity, we assume that the subunit filters of each channel differ only in spatial position, and that their nonlinearities are identical. The responses of the two channels are computed as weighted sum of their associated subunits, and the difference between them is then passed through a final rectifying nonlinearity to determine the firing rate of the neuron. We develop a method for directly and efficiently estimating all parameters of the model and we test this on responses of V1 neurons to white noise stimuli, demonstrating that the fitted model outperforms previously published LN, energy, and STC-based models on all cells.

2.2 Methods

(See chapter 1 for methods relating to the models discussed in the current chapter)

2.2.1 Physiology experiments

Flickering bar (XT) data are taken from Rust et al. [18]. The recording methods are similar to those described in section 3.1.1 of this thesis. Briefly, data was collected from primary visual area (V1) in adult macaque monkeys (*Macaca nemistrina* and *Macaca fasciularis*). Single units (n = 52) were isolated on microelectrodes that were lowered through a craniotomy and durotomy centered over V1. Units were determined with a dual-window discriminator.

Stimuli were white bar-noise, displayed by a Silicon graphics octane 2 workstation at 100 Hz. Each frame contained a 1-D white, binary stimulus array that varied across the receptive field orthogonal to the preferred direction. Thus, the stimulus appeared as flickering bars in which each bar was aligned to the optimal receptive field orientation. We refer to this class of stimuli as XT noise, referring to the two relevant dimensions of space (X) and time (T).

2.2.2 Measuring properties of the fitted models

Accuracy of each model is determined by how well it predicts the responses to data not used in model fitting. We perform 5-fold cross-validation by randomly breaking the data into 5 tranches, and iteratively fitting the model with 4 fifths of the data and testing with the holdout fifth. Performance is summarized by the average correlation coefficient between the actual firing rate and the model-predicted firing rate over the 5 tranches. Training performance is the average correlation for the 4 fifths of training data.

The structure of the subunit model is convenient for dissociating the spatiotemporal position of the receptive field from its tuning properties. We measure the spatial and temporal extent of the subunit filter and linear pooling for the XT models by fitting a 2-dimensional Gaussians with 6 parameters (1 for amplitude, 2 for spatial position, and 3 for the covariance matrix in a Cholesky decomposition [**?**]). Fitting a Guassian to the pooling maps is straight forward because the shapes closely match Gaussian bumps. The subunit filters are bandpass and contain both positive and negative components, so we first compute spatial power envelope before fitting a Gaussian. First, we find the 2-dimensional Hilbert transform in the direction of the subunit filters preferred orientation. The power envelope is then the square-root of the sum of squares of the original subunit filter and its Hilbert transform. We compute the relative influence of the excitatory channel and the suppressive channel by comparing the standard deviation of the model responses with the other channel removed.

2.3 Standard models for V1 neurons (XT)

The linear-nonlinear (LN) cascade model is the simplest and most well-known descriptive model for sensory neuron responses. At each moment in time, the linear stage computes a weighted sum (or integral, if the stimulus space is continuous) of the recent stimuli. The weights determine the stimulus selectivity of the model cell, and the linear operation can be interpreted as passive dendritic combination of incoming signals weighted by their associated synaptic efficacies. This response is then transformed with a rectifying nonlinear function, that can be interpreted as transforming membrane voltage to firing rate. When applied to neurons from area V1, this framework is only practical for simple cells, which are known to have a monotonic contrast-polarity response functions. An example fit to a simple cell is shown in Figure 2.1 (left). The sharp, rectifying, piecewise-linear output function shows that the cell will be strongly

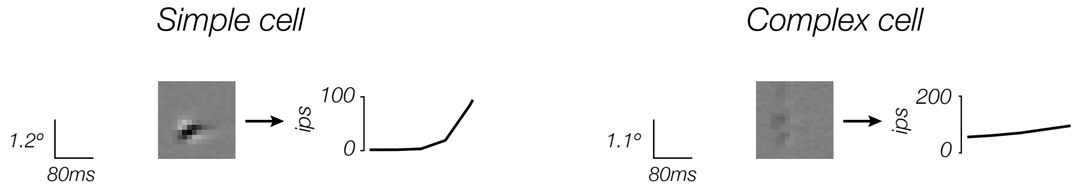


Figure 2.1: The Linear-Nonlinear (LN) model fit to an example simple cell and complex cell. The model fits the simple cell well. The linear filter shows a clear preference for direction selectivity because it is tilted in space-time, and the deep, rectifying nonlinearity shows that the cell's firing rate will be significantly modulated as a function of stimuli and contrast. However, the model does not do a good job at fitting the complex cell. The weak output nonlinearity is indicative of a model that cannot produce strong response variance, and thus cannot capture the real neuron's output range.

modulated by preferred stimuli. In terms of cross-validated performance (i.e. averaged over 5-fold cross-validated data tranches), the model performs relatively well for this cell with an r-value of 0.52.

Complex cells, which respond equally well to stimuli of opposite contrast polarity, are not well fit by these types of models. Figure 2.1 (right) shows an example complex cell fit with an LN model. The linear filter is weak and oddly structured with a multimodal spatiotemporal envelope, and the nonlinearity provides for very little modulation with stimulus drive. Indeed, the model poorly accounts for the cellular response, with cross-validated accuracy of $r = 0.28$. Higher-dimensional, nonlinear models are required for these neurons.

Adelson & Bergen [16] proposed the "energy" model as a means of computing directionally-selective complex cell responses in order to explain a variety of observations on human motion perception (see Figure 4 in the Background section). The model computes the sum of squares of two space-time oriented linear filters, chosen as a quadrature pair that have the same frequency response but different symmetry (one is even-symmetric, the other odd-symmetric). This Linear-Nonlinear-Linear (LN-L) construction is an instantiation of the qualitative description given by Hubel and Wiesel [1], and can be thought of as combining the responses of four simple cells with identical retinotopic location, orientation, and frequency selectivity, but differing in the precise spatial arrangement of excitatory and inhibitory lobes. The response of this model to sinusoidal gratings is phase insensitive, but retainis selectivity for spatial orientation and direction. The model is extended to include suppression by subtracting the responses of an energy unit tuned for opposite directions.

Though not typically fit to data in reverse correlation experiments, we develop a gradient method to estimate the parameters of an opponent energy model for spiking data (see Chapter 1). The procedure returns two quadrature filter pairs that correspond to excitatory and suppressive pairs described above. Because the model is designed to produce phase-invariance, it is only able to reproduce the example complex cell responses and fails to fit the simple cell (Figure 2.2; $r_{simple} = 0.08$, $r_{complex} = 0.41$).

Neurons in V1 have been traditionally categorized as simple or complex [1], but recent analyses suggest that their properties may lie on a continuum [54]. The model developed by Rust et al. [18] combines aspects of the LN and energy models into a single common framework, in which a set of LN and energy responses are additively combined into two channels that are classified as excitatory or suppressive. The channels are then combined using a spiking nonlinearity that includes both subtractive and divisive interactions. The filters for the LN and energy units are obtained from spike-triggered average (STA) and covariance (STC) analyses, respectively. Unlike either the LN or energy models alone, the combination of elements allows the Rust-STC model to accommodate both extremes of simple and complex cells (Figures 2.3 and 2.4; $r_{simple} = 0.56$, $r_{complex} = 0.47$ for the example cells).

A drawback of the Rust model is that the parameters are difficult to interpret

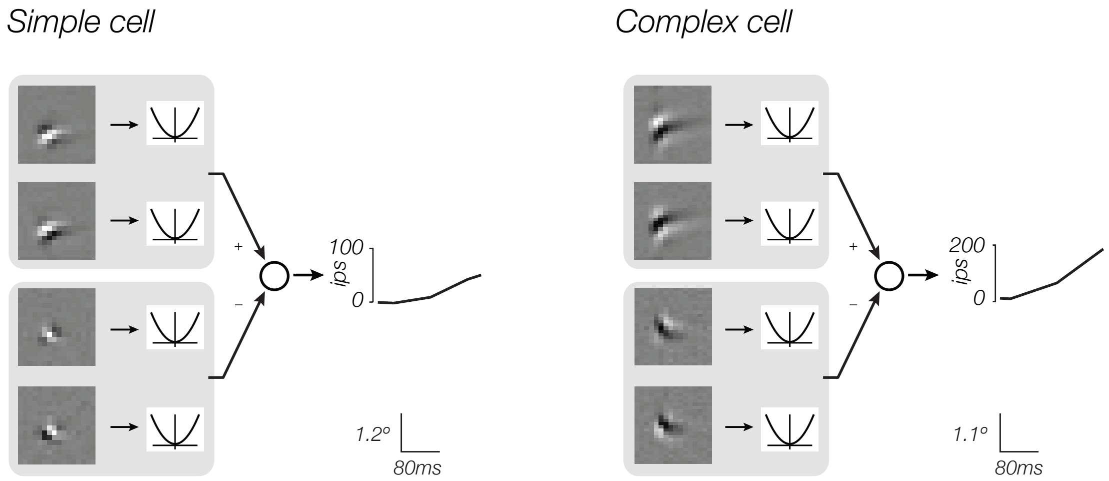


Figure 2.2: The Energy model fit to an example simple cell and complex cell (same cells as in Figure 2.1). Pairs of quadrature filters are squared and added together. The excitatory pairs are depicted at top and the suppressive pairs are depicted at the bottom. The suppressive filters for both cells show tuning for motion direction that is opposite of the excitatory filters. Although the simple cell has well defined filters, the squaring operation embodies the model with phase invariance, which is a poor description for these types of cells.

in a biologically meaningful way, primarily because the STC filters are forced to be orthogonal by construction. As a consequence, pairs of filters beyond the first pair are delocalized in both space-time and spatiotemporal frequency (Figure 2.4), which are properties that are not generally observed in real cells [18, 56]. Furthermore, the Rust model has many more degrees of freedom in its parameterization than the other models. This makes it difficult to obtain clean and unbiased model fits for individual neurons given the limits of data acquisition in real experiments [18]. Note that the two example cells shown were fit to data from two of the longer recordings in the dataset, each lasting more than one hour.

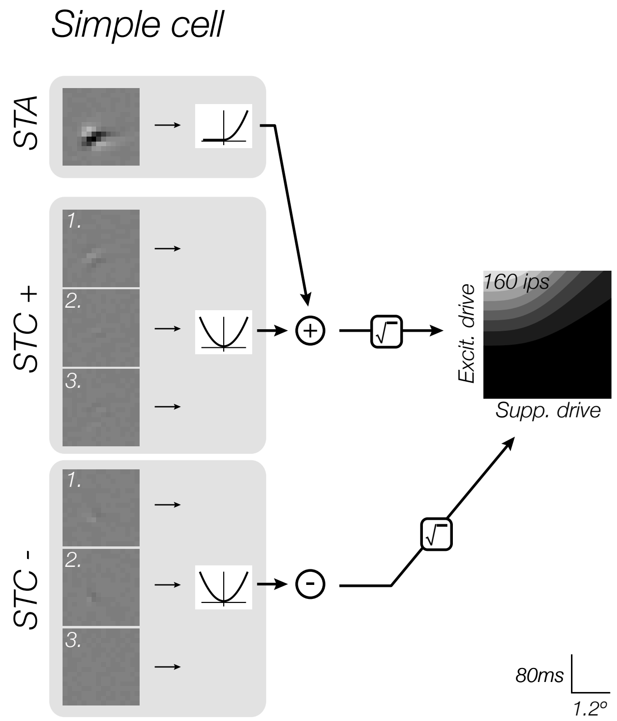


Figure 2.3: The Rust-STC model fit to the example simple cell. The responses of the excitatory filters are squared, weighted and pooled along with the STA response. The suppressive filters are similarly combined. Images are scaled to represent how the model weights each component before summation. Note that the predominant component of this cell is the STA filter; the STC filters contribute very little information. The excitatory and suppressive channels are combined with a joint nonlinearity to generate a firing rate, which for this cell shows both subtractive and divisive inhibition.

2.4 The generalized subunit model in primary visual cortex (XT)

We have developed a subunit model that retains many of the advantages of the Rust model, but with an architecture that has a more natural biological interpretation, and requiring far less data to achieve a satisfactory fit [56]. The model sums over two channels, one excitatory and one inhibitory; each channel is constructed from a bank of LN 'subunits' with identical receptive field structure but differing in their spatial and

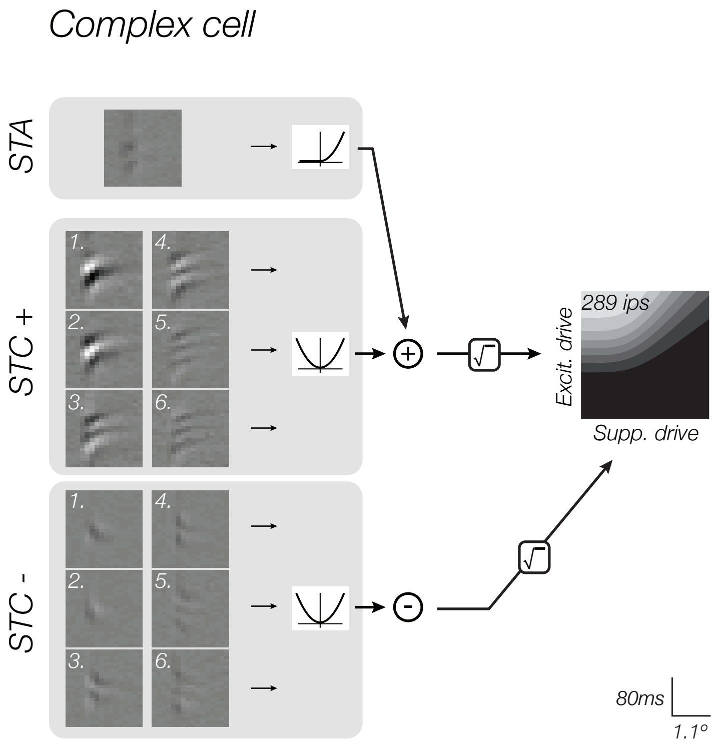


Figure 2.4: The Rust-STC model fit to the example complex cell. Figure conventions follow Figure 2.3. Note that the STC filters are much strong for the complex cell, and that the STA filter contributes very little drive. Also notice the structure of the higher-order STC filters, which are not well localized in space.

temporal position (in other words, the set of linear filters perform a convolution). For each channel, the set of linear filter responses are passed through a common output nonlinearity before they are linearly weighted and summed to generate the channel response. The model framework, along with examples of the computation performed at each stage, is depicted in Figure 1.5. We fit the linear subunit filter, the subunit nonlinearity, and the spatial pooling functions (weights) of both the excitatory and suppressive channels for each cell (see Chapter 1).

The parameters of the subunit model can be interpreted in terms of hypothetical biological components. The field of LN subunits can be viewed as a population of afferents from upstream visual areas or layers, and the channel response arises from a linear combination of these afferents, weighted by their associated synaptic efficacies. We find that the fitted subunit filters are tightly localized in space and time, a widely-observed feature of real receptive fields in visual cortex [1]. Moreover, the model can easily be configured to produce cells with a continuum of response properties, ranging from simple to complex and beyond. For example, if the subunit nonlinearity is half-rectifying and the spatial pooling is punctate, the model will exhibit phase selectivity and would be classified as simple. Conversely, if the subunit nonlinearity is full-rectifying and/or the spatial pooling is broad, relative to the size and/or preferred spatial period of the subunits, the model will be selective to the feature expressed by the subunit filter but invariant to exact spatial position. Fitted model parameters for the example simple and complex cell are plotted in Figure 2.5, for which they exhibit good cross-validated performance ($r_{simple} =$ 0.55, $r_{complex} =$ 0.42).

On average the subunit model produces the most accurate cross-validated fits of all the tested models ($< r_{LN} >=$ 0.18, $< r_{Energy} >=$ 0.17, $< r_{STC} >=$ 0.26, $< r_{subuni}t >=$ 0.29; Figure 2.6, filled circles). Through the Rust model performs reasonably well, it is highly susceptible to overfitting, due to the large number of parameters. This is demonstrated by comparing the cross-validated performance to that on training data (Figure 2.6, open circles). On training data, the Rust model outperforms the subunit model, but it underperforms on a held-out test set, a standard indication of "overfitting", in which the model is explaining noise along with signal. In comparison, the pure LN model and the energy model each have many fewer parameters than the Rust model and do not exhibit this telltale signs of overfitting. Rather, these models (with dimensionality that is on par with the subunit model) fail because they are insufficiently flexible to account for the behaviors of all neurons.

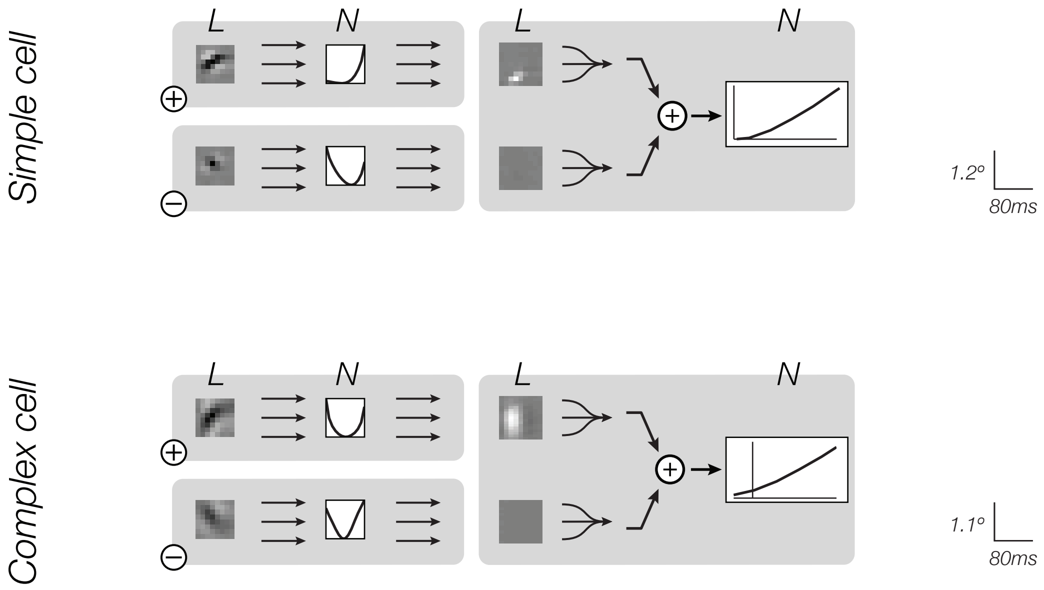


Figure 2.5: Subunit model fit to example simple cell and complex cell (same cells as in Figure 2.1). For each model, the top left box represents the excitatory LN convolutional subunits (linear filter and output nonlinearity), and the bottom left box represents the suppressive channel. The subunits are weighted in space and time with the pooling maps in the right box before being added and passed through an output nonlinearity. The excitatory pooling map for the simple cell is small and punctate. The excitatory pooling map for the complex cell is broad, summing subunits over a wide swath of space and time, indicating spatiotemporal invariance.

The number of parameters needed to estimate each model is substantially different. For a cell presented with a 16 x 16 spatiotemporal stimulus grid, the Rust model requires the estimation of about 32,000 parameters (16^2 x 16^2 / 2 for the covariance matrix, and about 10 for the filter weighting and output nonlinearity), while a two-channel subunit model with an 8 x 8 convolutional subunit filter requires only about 300 parameters (2 · [8 x 8] for the subunit filters, 2 · 9 for the subunit nonlinearities, 2 · [9 x 9] for the spatial pooling functions, and 9 for the final output nonlinearity).

It is also worth noting that while both the subunit and Rust models are able to

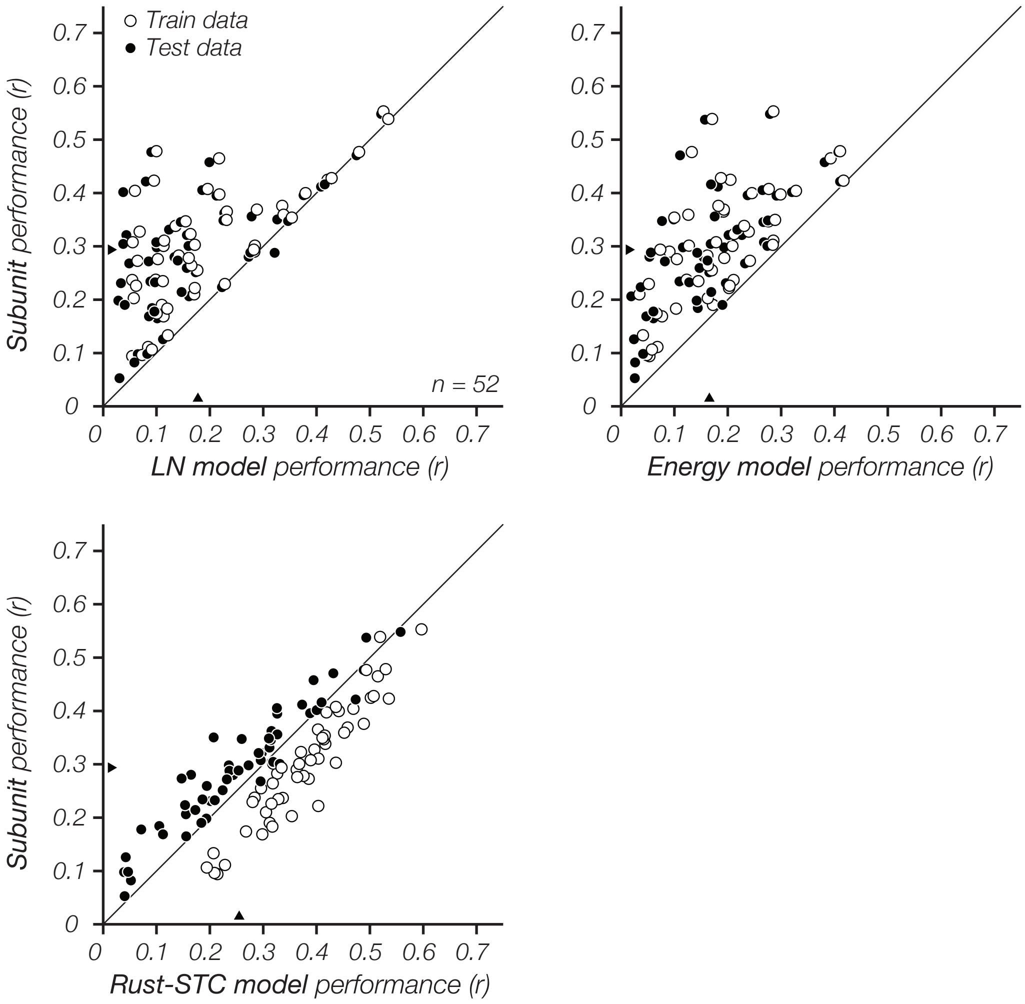


Figure 2.6: Comparison between the four models for both testing and training data. The subunit model outperforms all other models on test (cross-validated) data. All models except the Rust-STC model perform similarly well on test and training data, but the Rust-STC model overfits, giving a substantially better result on training data than test data.

accommodate cells along the entire spectrum of simple to complex (Figure 2.7), they have a philosophically different manner of doing so. The Rust model is flexible because it blends the LN model with an energy-like model in proportions that are relevant to each cell; in this view, every cell is composed of both a linear part and a quadratic part, and each part performs a very different type of computation. The subunit model

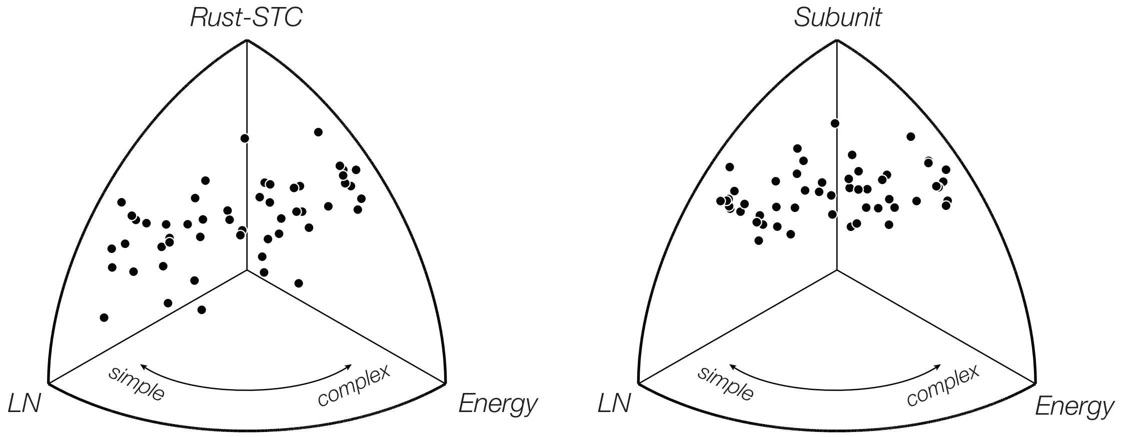


Figure 2.7: Comparing the Rust-STC model (left) and the subunit model (right) across the simple-complex response spectrum. Both models are able to accommodate the full range of response types. In the triplot on the left, the distribution of points is not horizontal because for simple cells the LN model performs nearly as well as the Rust-STC model. This is because the LN model is a subset of the Rust-STC model, and both use the STA to compute their linear response. STA is a very efficient analysis technique and is much less susceptible to noise than STC. In comparison, on average, the subunit model outperforms even the LN model for simple cells because it uses the available data more efficiently.

is flexible across cell type with a different approach; a simple cell is built from a single subunit, and the response can become gradually more complex by incorporating more subunits, each constructed from a filter with the same selectivity, and an identical nonlinearity.

Regardless of how well the Rust-STC model can be fit to individual cells, there is a noted curiosity regarding the shape of the STC filters. Real cells in V1 are expected to have localized receptive fields in both the spatial and spectral domains [57]. STA filters for real cells are mostly well localized, as are the first two (quadrature) pairs of STC filters. However, subsequent pairs of quadratic filters are usually multimodal in space and frequency [18], shown for the example complex cell in Figure 2.8 (left). This property is due to an artifact of the analysis rather than real afferents into the complex

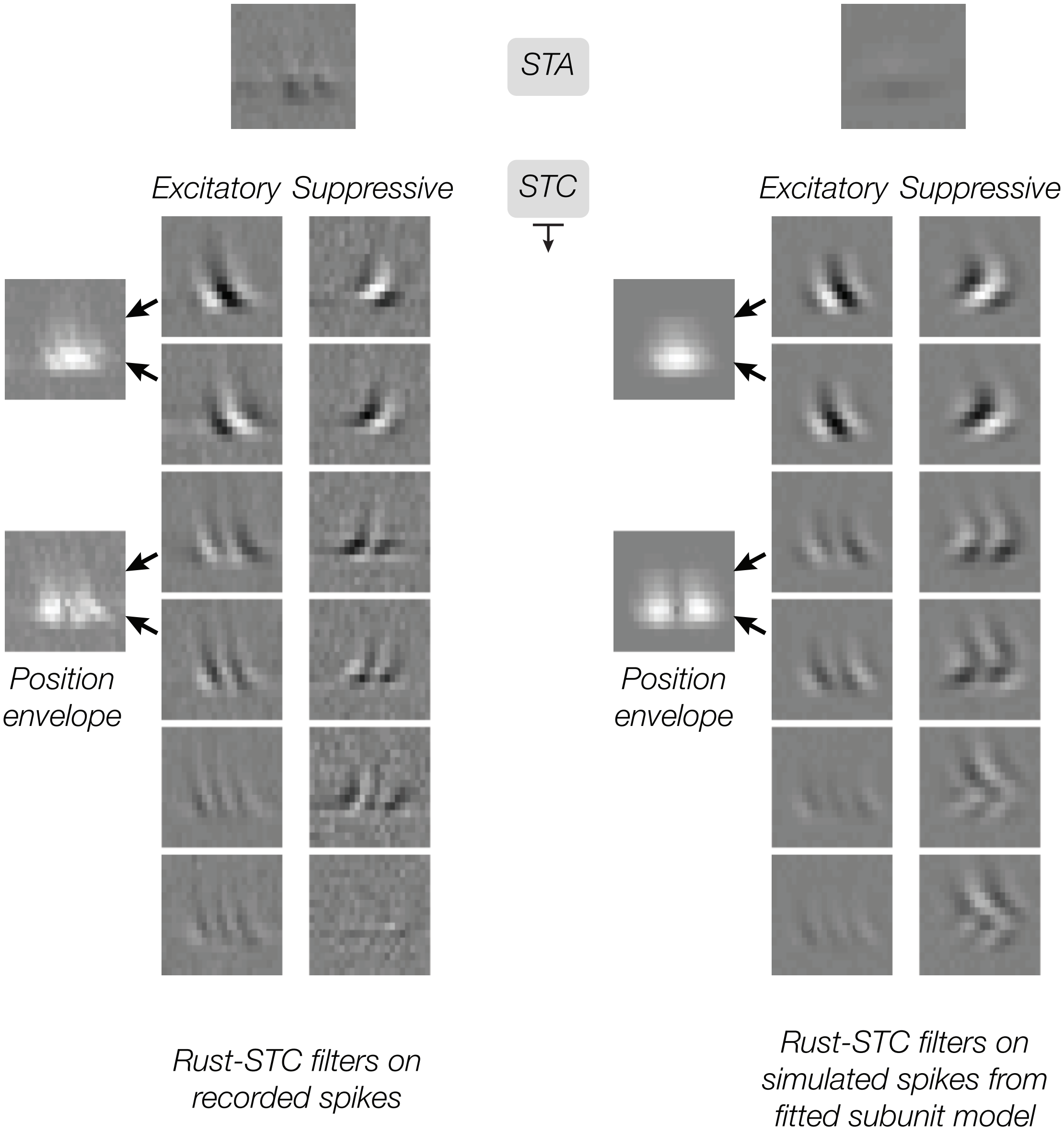

STA
STC
Excitatory
Suppressive
Excitatory
Suppressive
Position envelope
Position envelope
Rust-STC filters on recorded spikes
Rust-STC filters on simulated spikes from fitted subunit model

Figure 2.8: Performing STC on a fitted subunit model reproduces the finding in Rust et al. that complex cells have multiple quadratic filters, and these higher-order filters have unlocalized, bimodal spatial and spectral envelopes. On the left is the STA (top) and STC (bottom) filters for the example complex cell (for clarity, each STC filter has been normalized to have the same contrast). To the far left is show the spatial envelope of pairs of excitatory STC filters, which is simply their sum of squares. We then fit the subunit model to the same cell and simulate a spike rate from the model for a white-noise input. The STC components of this simulated cell closely match those of the real cell, including the bimodal spatial envelope of the higher-order filters.

cell. If we fit a subunit model to the real cell (Figure 2.5, bottom), simulate responses of the model to Gaussian noise, and then fit the Rust-STC model to the simulated rates, we obtain STC filters that are qualitatively similar to the real STC filters (Figure 2.8, right). Thus, the subunit model is capable of capturing similar receptive field information as the Rust-STC model, but it's interpretation is much easier to connect to a biology: namely, spatiotemporally and spectrally localized afferents.

The subunit model also has the advantage that it's structure allows for a dissociation between receptive field position and spatial extent, and stimulus tuning properties. Imagine a simple cell and a complex cell with identical subunit filters, and thus identical tuning properties, but that differ in the number of subunits that they pool together. The complex cell should pool over comparatively more subunits, arranged broadly over space, to generate position invariance. We measured the spatial and temporal extent of the subunit filter and pooling map for each cell (see Methods, section 2.2.2). Indeed, our data shows that most complex cells pool over a larger spatial region than simple cells (Figure 2.9, top left), in both the excitatory channel and the suppressive channel ($r_{excitatory} = 0.42$, $p < 0.05$; $r_{suppressive} = 0.39$, $p < 0.05$). The extent of pooling over time is not correlated with cell type (Figure 2.9, top right), but the suppressive channel tends to sum over time more broadly than the excitatory channel ($p < 0.05$, t-test).

The smaller spatiotemporal pooling envelope for simple cells is not because simple cells have less suppression. For each cell we calculate a 'channel strength' index that is 0 if excitation and suppression are balanced and near 1 if excitation is much stronger than suppression. Though it has previously been reported that simple cells and complex cells have equal amounts of inhibition [58, 59], our data with white-noise stimuli show a tendency for simple cells to be more balanced and for complex cells to have a stronger

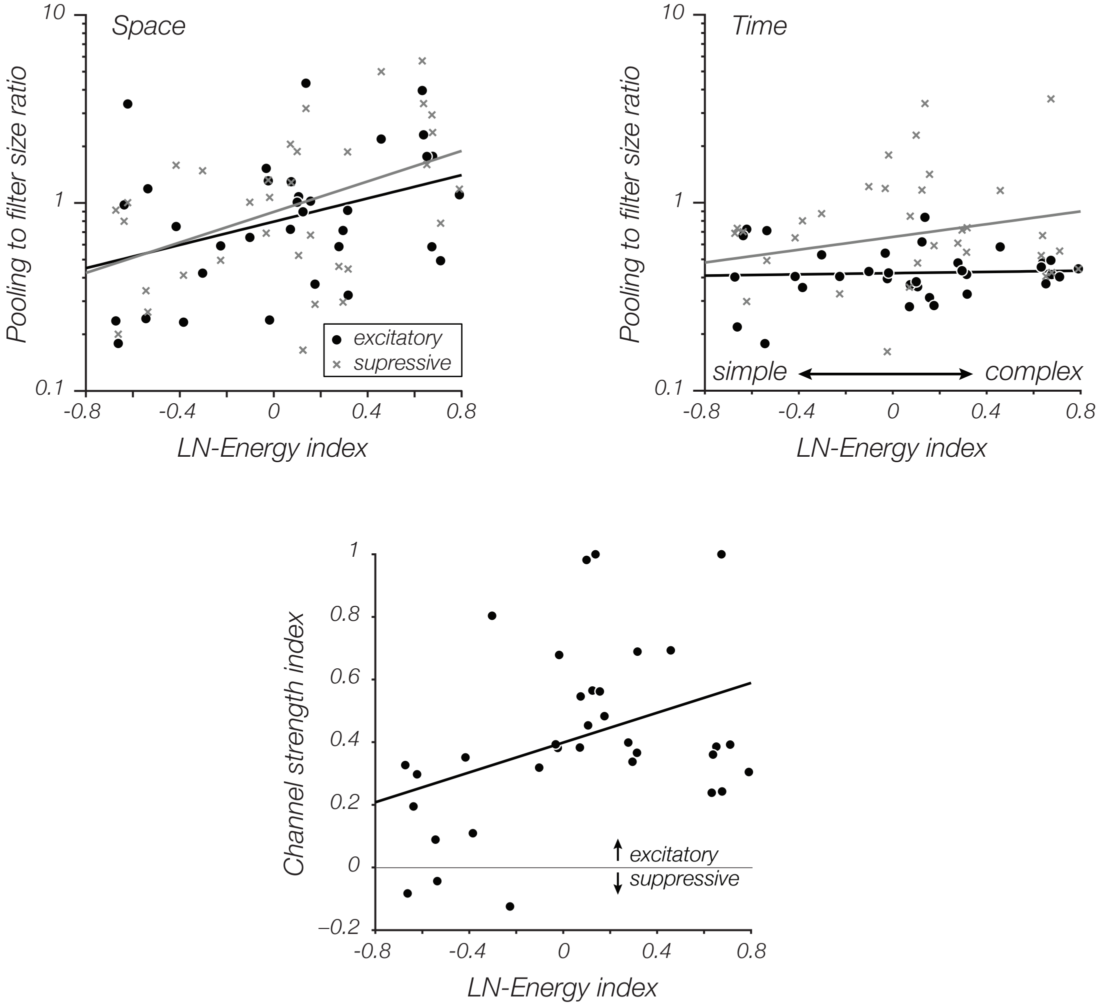


Figure 2.9: Subunit pooling over space and time. We fit a Gaussian to the subunit filter envelope and the spatiotemporal pooling function and compare the sizes of these elements. The abscissa for each plot is an index that compares model performance for the LN and Energy models ($(r_{Energy} - r_{LN})/(r_{Energy} + r_{LN})$), and can thus be viewed as an estimate of cell complexity. Complex cells (i.e. the right end of each graph) pool over a larger area in space than simple cells, but not in time. Complex cells tend to have less suppression than simple cells.

excitatory drive (Figure 2.9, bottom; r = 0.38; $p <$ 0.05), though the correlation is weak.

Chapter 3

3D spatiotemporal subunit models (XYT)

3.1 Methods

3.1.1 Electrophysiology

Flickering pixel (XYT) data were recorded from primary visual area (V1) in adult macaque monkeys (*Macaca nemistrina* and *Macaca fasciularis*; 6 animals). Typical experiments spanned 5-7 days during which animals were maintained in an anesthetized and paralyzed state through a continuous intravenous infusion of sufentanil citrate and vercronium bromide. Core temperature was kept fixed within the physiological range and vital signs were continuously measured (e.g. heart rate, end-tidal pCO2 levels, blood pressure, EEG activity, and urine quantity and specific gravity). Eyes were treated with topical gentamicin, dilated with topical atropine, and protected with gas-permeable hard contact lenses. Subsequent corrective lenses were chosen via direct ophthalmoscopy to make the retinas conjugate to the experimental monitor. All experimental processes and animal care were performed in accordance to protocols approved by the New York University Animal Welfare Committee, and are in compliance with the National Institute of Healths Guide for the Care and Use of Laboratory Animals.

Single units were isolated on quartzplatinumtungsten microelectrodes (Thomas Recording) that were lowered through a craniotomy and durotomy centered between 10 and 16 mm lateral to the midline and roughly 4-6 mm behind the lunate sulcus. We collected units across all cortical depths with receptive fields that were located parafoveally, at about 5-10 degrees from the center of gaze. The amplified signal from the electrode was bandpassed (300 Hz to 8 kHz) and routed through a dual window time-amplitude discriminator (EXPO) from which single-unit spike times were recorded at a resolution of 0.1 ms.

3.1.2 Visual Stimulation

XYT pixel-noise stimuli were presented on a gamma-corrected CRT monitor (Eizo T966; mean luminance around 35 cd/m^2). The resolution of the display was 1280 x 960 pixels, set to refresh at 120 Hz. We generated stimuli pseudorandomly using Expo software on an Apple Macintosh computer (http://corevision.cns.nyu.edu). The stimulus was a spatial pixel array (usually of dimensions 16 x 16) of white, ternary noise that was continuously refreshed at 40 Hz. The size of the array was chosen to be roughly double that of the receptive field as measured with optimal moving gratings while still maintaining adequate pixel resolution with respect the the receptive field features. For a subset of cells we also presented a repeated noise stimuli to act as cross-validation. These stimuli had identical statistics to the main white-noise stimuli, but lasted 25 s each and were repeated 20 times in succession.

3.1.3 Orientation tuning experiments

We also collected direction tuning curves to drifting grating stimuli for each cell. Most direction tuning curves were collected at the preferred spatial frequency (SF) and temporal frequency (TF) determined by hand, though many cells were then also run at their preferred spatiotemporal frequency as determined by their separable SF-TF tuning curves. Model-predicted direction tuning curves were obtained by generating drifting sinusoidal images near the model-optimal SF and TF and presenting them to each fitted model. For the models, the sinusoidal images have a higher *effective* contrast than the noise images to which the models were fit because gratings are a closer match to each receptive field. As a result, we fit a new output nonlinearity for each model and each cell so account for the output scaling. In addition, the subunit nonlinearities must also be able to extrapolate to higher effective contrasts, because the subunit outputs for grating stimuli is often outside the support of the nonlinearity defined with noise stimuli. Thus, we fit each subunit nonlinearity with a piecewise power-law function; the nonlinearity for each channel is split at zero and both the positive and negative side are fit with an independent nonlinearity of the form $x^p - b$, where b is an offset and p is a power between 0 and 5. This allows us to describe nonlinearities that are either half-wave or full-wave rectifying.

3.2 The subunit model in XYT

The subunit model is easily extended to handle stimuli with two spatial dimensions in addition to the temporal dimension. We presented 38 cells with ternary pixel noise presented at 40 Hz on a spatial grid with a typical extent of 16 x16 pixels (we looked back 8 frames in time yielding an embedded stimulus matrix of 2048 dimensions). Due

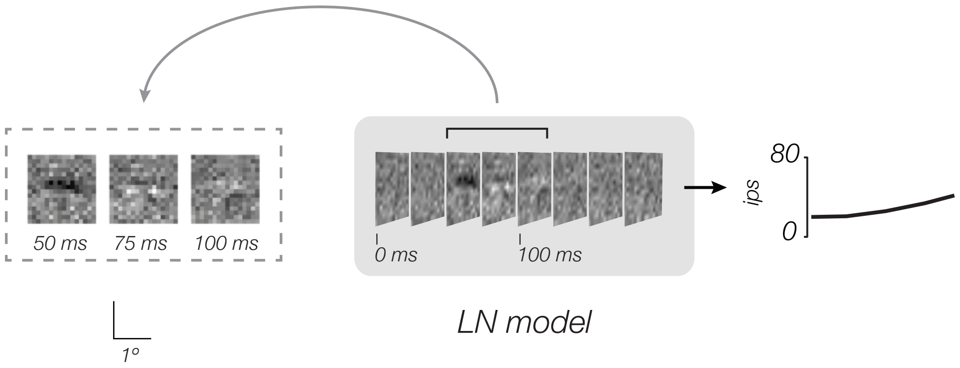


Figure 3.1: LN model for example V1 complex cell probed with 3 dimensional XYT stimuli. The linear filter is presented as a stack of spatial maps at different slices of time, preceding a spike. Three time slices at the optimal delay are highlighted at left. The stimulus is projected onto the filter and the response is passed through the output nonlinearity at right.

to the relatively slow frame rate we fit a subunit model for each cell that is convolutional in space but not in time - in principle, temporally shifted subunits could also be included if the stimulus frame rate were increased.

Unlike for the XT stimuli, we cross-validated models for a subset of neurons against responses to a novel 25 second stimulus, which was repeated 20 times. Test performance is measured as the average correlation (r-value) between the actual binned firing rate and the model-predicted firing rate. For these same cells we also calculate *training* performance as the correlation between these values for the stimuli on which the models were trained (see Figure 3.6). We test the LN, energy, Rust-STC, and subunit models.

The four model fits for an example directional cell are shown in Figures 3.1 – 3.4. The filters for each model are now three-dimensional; they are plotted as 25 ms slices through time over a range of 200 ms preceding spikes. The LN model filter for this cell shows a preference for horizontal features that drift upwards over time (Figure 3.1). This preference is apparent in all models: the energy model filters (Figure 3.2), the

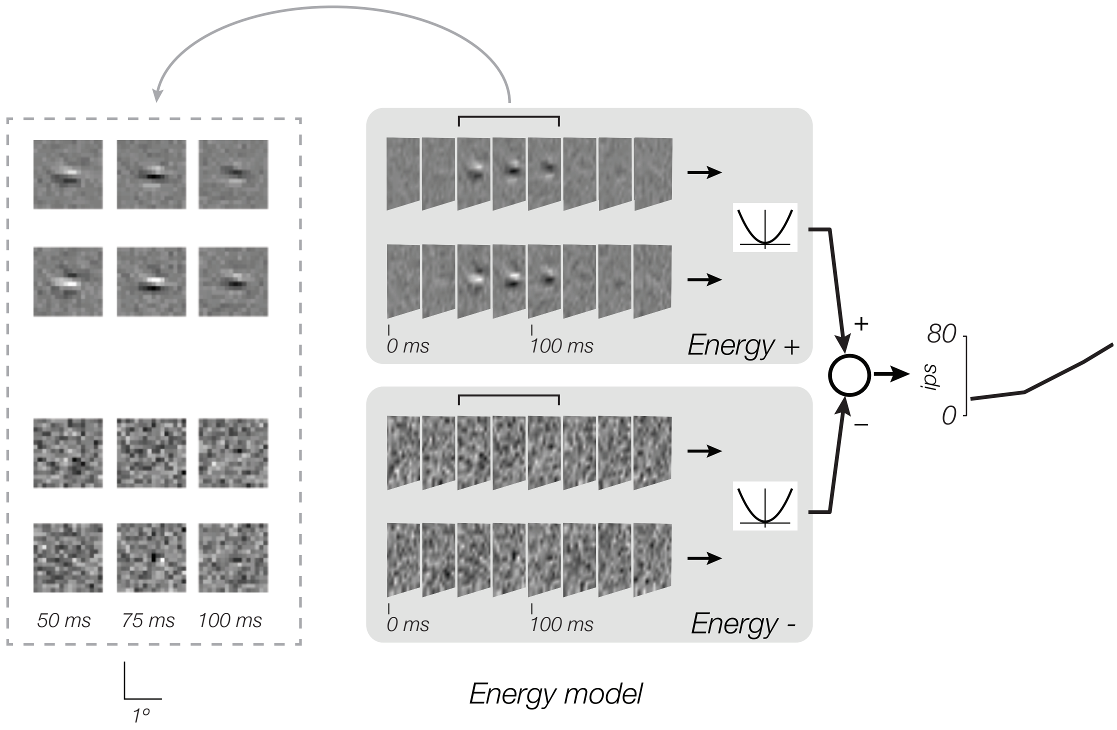


Figure 3.2: Energy model for example V1 complex cell. Conventions follow Figure 3.1. The outputs of two excitatory filters are squared and summed into an excitatory channel (top). A suppressive channel (bottom) is subtracted, and the output is passed through an output nonlinearity.

Rust model second-order filters (Figure 3.3), and the subunit kernels (Figure 3.4). The Rust model filters for this cell are much noisier than than the filters for any of the other models, stemming from large sample covariance matrix that must be estimated from the stimulus. For this example cell, the subunit model exhibits best cross-validated performance ($r_{subunit} =$ 0.54, $r_{STC} =$ 0.43, $r_{energy} =$ 0.28, $r_{LN} =$ 0.16).

Like the models fit to the XT data, on average the subunit model outperforms the other models in cross-validated tests across all cells ($< r_{subunit>} =$ 0.27, $< r_{STC} >=$ 0.16, $< r_{energy} >=$ 0.12, $<_{rLN}>=$ 0.12; Figure 8). For these high-dimensional stimuli, the difference in model performance is more substantial than for the XT stimuli: here, the subunit model performs about 70% better than the Rust model on average. The

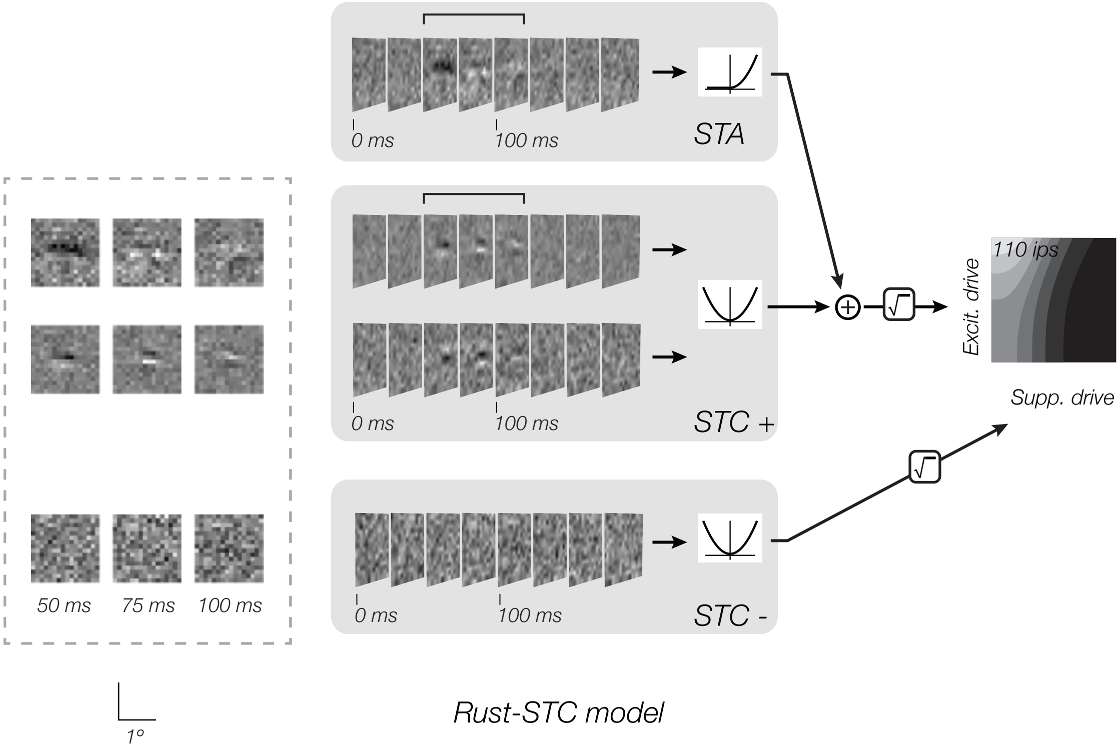


Figure 3.3: Rust-STC model for example V1 cell. Conventions follow Figure 3.1. An excitatory channel is computed by combining the responses of an LN unit and multiple excitatory STC units (only two depicted). A suppressive channel is also formed in an analogous manner. The two channels are combined with a joint nonlinearity (right) that includes subtractive and divisive suppression.

Rust model also shows a larger gap between training and testing performance than for the other models (Figure 3.5, open and closed circles).

We can also compare the performance of the subunit model to that of an estimated upper bound, which we refer to as the 'oracle' prediction. The oracle uses the mean response over 19 repeated presentations to predict the firing rate for the 20th presentation (Figure 3.6). Predictably, the oracle outperforms the subunit model (Figure 3.7; two-tailed t-test, $p << 0.05$), but by only a modest amount. On average, the

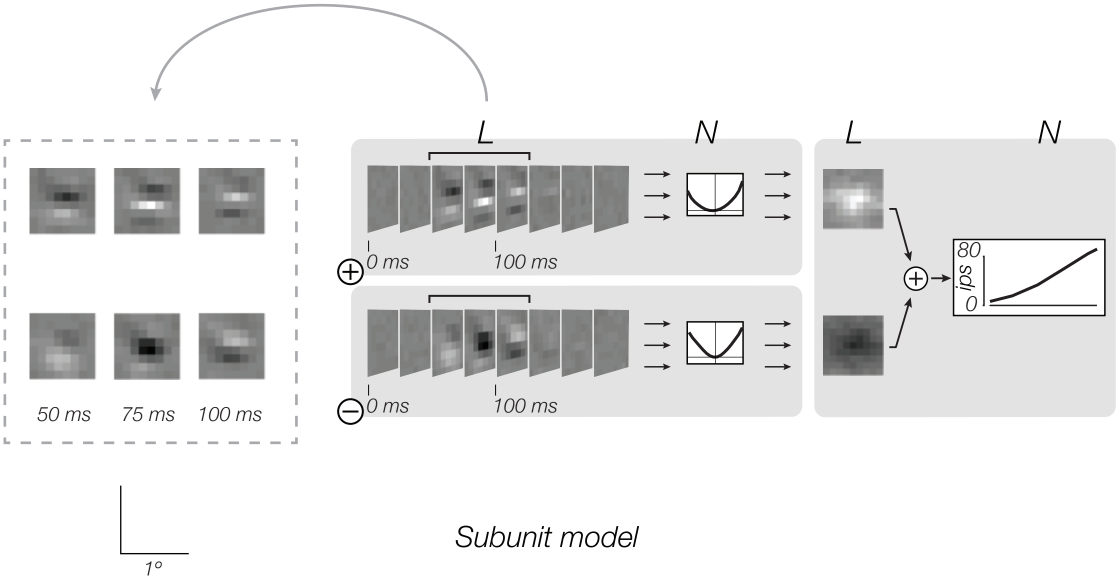


Figure 3.4: Subunit model for example V1 cell. Conventions follow Figure 3.1. A convolutional subunit filter is applied to the stimulus image and passed through a pointwise, rectifying nonlinearity for both an excitatory and suppressive channel. The subunits for both channels are separately pooled over space and then combined. Finally, the response is passed through an output nonlinearity. Note the clear selectivity for upward motion (left), and suppression by low-frequency downward motion.

subunit model performs 76% as well as the oracle model, while the Rust model only performs 49% as well as the oracle. In summary, the subunit model is able to capture a significant percentage of the explainable variance in each cells stimulus-modulated firing rate, and can do so much more efficiently (i.e., with less training data) than the Rust model.

Each model can also be used to predict direction tuning curves for each cell in response to drifting grating stimuli. We measured responses to synthetic sine-wave gratings at the optimal spatial and temporal frequency for each cell, and use each model, with parameters fitted to the white noise training data, to predict the grating responses averaged across phase (i.e. the DC response). The grating stimuli have higher effective contrast than the white noise stimuli that are used to estimate the

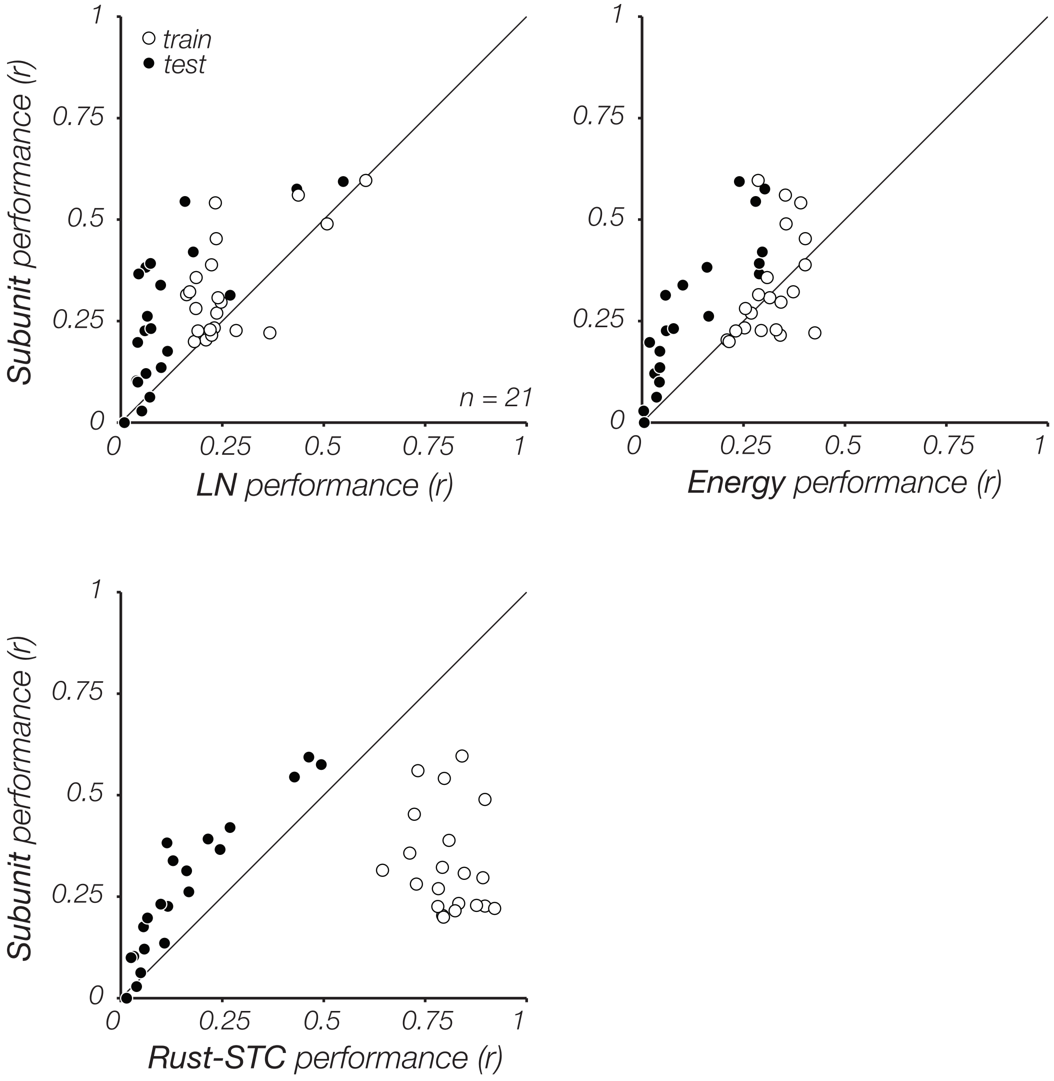


Figure 3.5: Subunit model performance is compared against the LN, Energy, and Rust-STC model performances. Given the large dimensionality of the stimulus, there is a tendency for most models to overfit, depicted as the difference between the performance for training data and test data. The Rust-STC model overfits the most of the four models.

model parameters, and so we allow for the estimation of a new output nonlinearity that maps the model drive (sum of excitatory and inhibitory channels) to the observed firing rate. We then compare these model-predicted direction tuning curves to the actual measured tuning curves (Figure 3.8 shows this for an example cell).

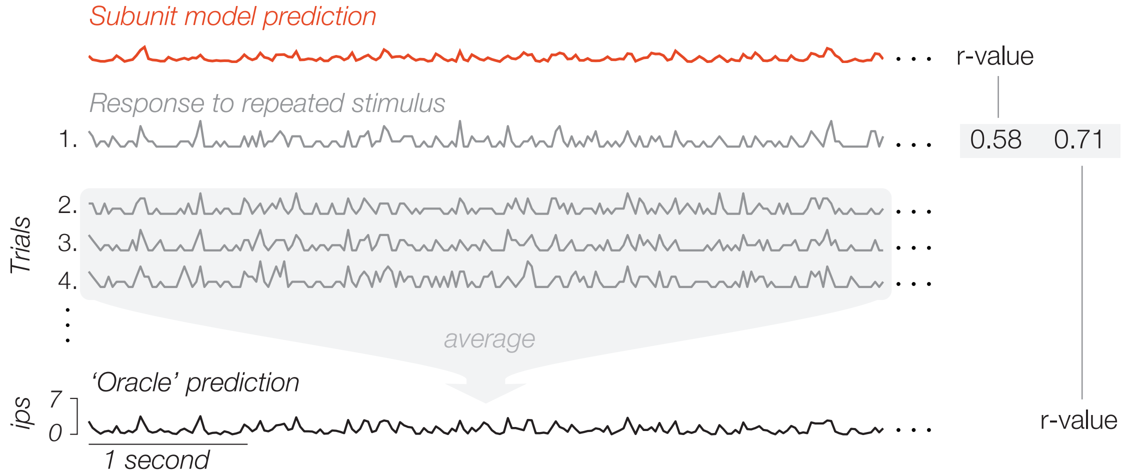


Figure 3.6: For a subset of cells we collected neuronal responses to repeated stimulus trials. The subunit model accuracy is the average of the correlation between the subunit model prediction and all twenty trials. We also compute an oracle prediction which attains the best possible performance that any stimulus-driven model could hope to attain. The oracle for trial n is computed by summing the responses to all other nineteen trials. This value is averaged for all twenty trials.

The subunit model is consistently more accurate than the others in predicting the direction tuning curves, as measured by correlation coefficient (Figure 3.9, top). The superior performance of the subunit model comes both from a estimation of direction preference (Figure 3.9, left) as well as tuning width (Figure 3.9, right). In general, each model predicts broader tuning curves than the real cell exhibits, though the subunit model is the most accurate in this regard. Of note is that the cells for which the model provides accurate tuning curve estimates are also the cells for which the subunit model predicts novel white-noise stimuli well ($r =$ 0.49, $p =$ 0.025; Pearson r-value, data not shown). These cells are also the cells with the highest average firing rate ($r =$ 0.57, $p =$ 0.007; Pearson r-value) and the highest number of total recorded spikes ($r =$ 0.59, $p =$ 0.005; Pearson r-value).

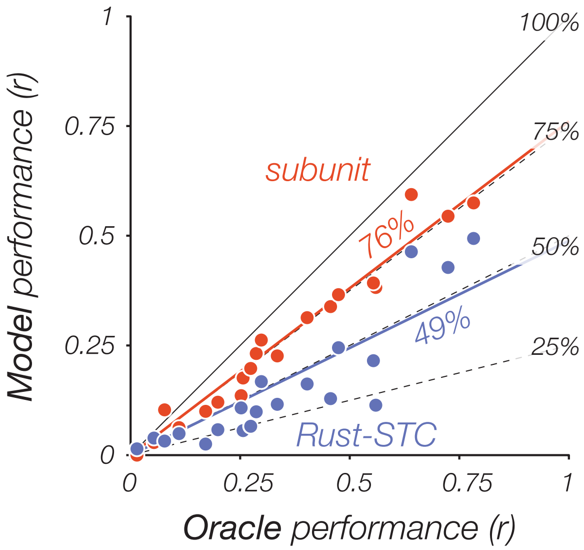


Figure 3.7: The oracle performance is compared to the subunit model (orange) and the Rust-STC model (purple). Cells would fall along the identity line if the model was capturing all of the explainable variance. The subunit model comes closer to reaching this upper bound of performance (76%) than does the Rust-STC model (49%) on average.

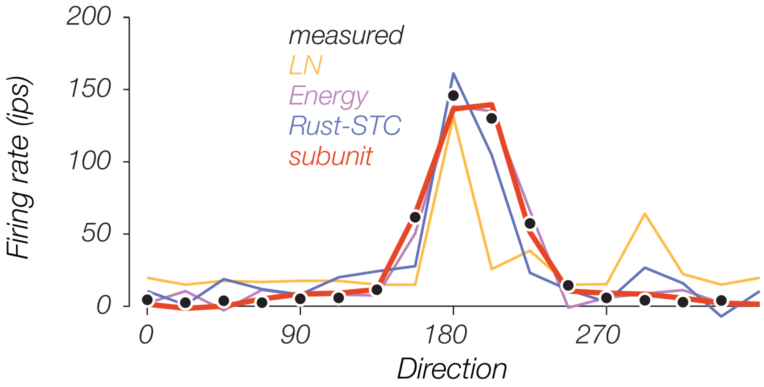


Figure 3.8: Direction tuning curve for an example cell. Black dots show the tuning curve for drifting gratings over 16 directions. This neuron was direction tuned because the tuning curve is unimodal. We also simulate the response of each model drifting gratings and normalize the response range to match the amplitude of the measured tuning curve. The subunit model provides the best fit to the actual tuning curves.

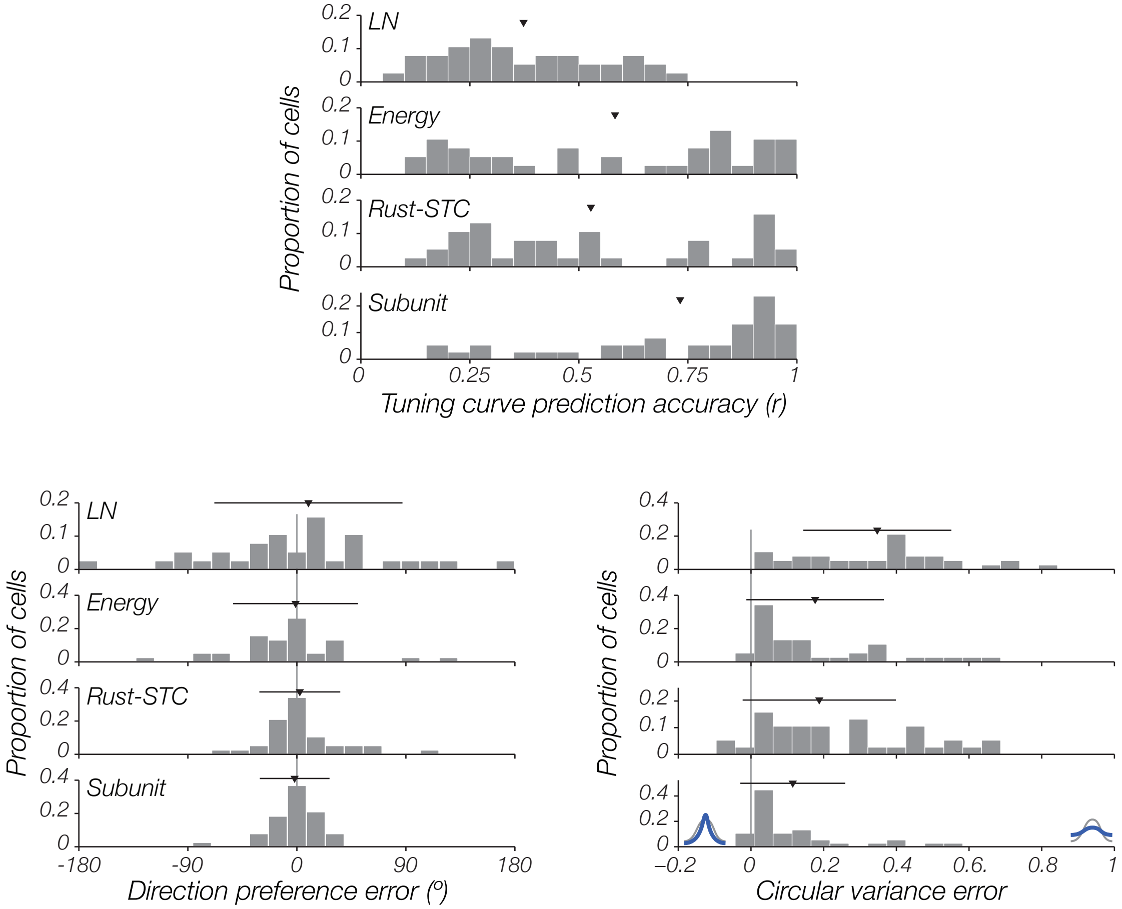


Figure 3.9: We quantify tuning curve prediction as the correlation between the actual tuning curve and the model-predicted tuning curve. The Subunit model performs the best. Part of this performance increase comes from smaller errors in predicting the preferred orientation (bottom left), and part comes from smaller errors in predicting the circular variance (bottom right; the black and blue curves are cartoons of the actual and model predicted tuning curves, showing that at the right end of the plot the model is less tuned to gratings than the actual cell). All models tend to predict flatter tuning curves than predicted from drifting gratings. Error bars (horizontal lines) are standard deviation. (n=52)

3.3 Discussion

We've developed a generalized subunit model for neurons in primary visual cortex, along with a method to directly and efficiently estimate the parameters of this model from measured firing rates. The model includes the LN model, widely used for simple cells, and the energy model, used for complex cells, as a special cases. When fit to relatively short segments of white noise stimuli responses, the new method significantly outperforms both classic models in terms of cross-validated accuracy in macaque primary visual cortex. We also compared the model to that of Rust et al. [18], which has previously been shown to provide good qualitative fits regardless of cell type; the subunit model performs significantly better on validation data because it is more compactly parameterized, and thus less susceptible to noise. The parameters of the subunit model are also easy to interpret as the hypothetical building blocks of a biologically-instantiated receptive field. Our results indicate a continuum of pooling behaviors across V1 populations, from cells that are extremely localized (simple cells) to complex cells that gather rectified responses over far larger regions than those predicted by the energy model.

Subunit models have a long history and date back to the work of Hubel and Wiesel, who proposed that the lack of distinctive excitatory and suppressive subregions in complex cell receptive fields could be explained by combining the response of a set of spatially distributed simple cells, each with identical orientation and spatial frequency tuning [1]. In the retina of the cat, Hochstein and Shapley showed that spatial frequency responses of Y cells arose from spatial filters that were significantly smaller than the receptive field size; the authors proposed a model for these responses based on a sum of nonlinear subunits, and this appears to be the first use of the nomenclature [34]. Adelson and Bergen introduced the spatio-temporal energy model as a simplification of the Hubel and Wiesel model that used only two filters in quadrature phase and whose responses were squared and summed [16]. Fukishima suggested that the spatial pooling of rectified filter responses is a canonical operation that is repeated in all visual cortical areas to allow translation-invariance in the representations [41], a notion that was generalized to include pooling over other dimensions by Perrett and Oram [60] and is widely used in object recognition systems [61, 62, 63, 64, 65, 42]. It has also been proposed as a means of representing statistical quantities (local variances or covariances), which can support the representation and recognition of visual texture [66].

Methods for fitting V1 models directly to physiological recordings originate with the work of Jones and Palmer, who used reverse-correlation methods [43, 67] to obtain spatial or spatiotemporal receptive fields for simple cells [14]. Emerson and colleagues introduced a sparse noise methodology for fitting the energy model to complex cell responses [17]. Lau and colleagues fit complex cell responses using an artificial neural network model that included multiple unconstrained filters [68], revealing families of oriented receptive fields of differing phase similar to the energy model. Parallel work by Touryan et. al. used spike-triggered covariance analysis [45] to estimate orthogonal pairs of oriented filters underlying complex cell responses [48], and later work by Rust et al. extended this analysis to reveal larger sets of orthogonal filters underlying most complex cells [18]. This generality engenders experimental limitations: STC methods generally require Gaussian stimulus ensembles, and recording durations that strain existing experimental capabilities (although Park and Pillow have recently introduced regularization methods for STC that partially alleviate these problems [57]). Information-theoretic methods have also been introduced to find orthogonal filters that capture the subspace of stimuli responsible for neural responses [15, 19]. Compared with STC, these methods are less restrictive in terms of the stimuli (for example, they have been used with natural movies), but the estimation of information generally makes them even more data intensive.

The method introduced here greatly reduces the data requirements of the subspace methods, by exploiting the simplified parameterization that results from banks of spatially shifted (convolutional) filters. Analogous advantages have been realized in training artificial neural networks for pattern classification [62]. One can approximate this approach by first finding a response subspace (e.g. with methods described in the previous paragraph such as STC) and then solving for a filter whose shifted copies span that same subspace [18, 21]. However, such a two-step procedure does not generally realize the gains in accuracy that should accompany the reduction in dimensionality, because the second step does not take into account the estimation errors of the first. There are also several related analysis techniques in recent literature that incorporate the concept of subunits without building an explicit receptive field model. Local Spectral Reverse Correlation (LSRC) computes a spike-triggered local Fourier spectrum over localized, stimulus windows [69]. The method computes both localized spectral tuning profiles (which can be linked to squared responses of spectrally-tuned filters), as well as a position map. But as with the subspace methods, spectral responses are estimated independently, and so it requires substantial amounts of data to achieve clean results. In addition, the spectral characterization of local regions does not allow for a separation of excitatory and suppressive influences, complicating any biological interpretation. Similar benefits and drawbacks may be attributed to the method of Sasaki & Ohzawa, which estimates local second-order subunit kernels that respond in a contrast invariant manner to windowed stimulus regions [70].

Perhaps the closest published method to our own is that of Eickenberg and colleagues, who used an information-theoretic objective function to fit a model constructed from an OR-like combination of shifted LN subunits [71]. As in our results, the authors find that model-fitting is substantially more data-efficient than for a general subspace model. The model differs from ours in a number of aspects, including the use of a different combination rule (Or-like product, instead of a sum), a general nonlinear combination of the responses of multiple filter outputs (instead of the linear combination of nonlinear channels used in our model), and lack of a final nonlinearity. The authors find that the model works best for complex cells, necessitating the use of a separate model to handle simple cells.

The model and methods presented here offer a number of possibilities for extension and generalization. In particular, our current fitting procedure optimizes squared error between model response and observed spike counts. Incorporating a spike generation stage, even a simple one such as a Poisson model, would more accurately capture the precision associated with different spike count observations. Incorporating other known nonlinear properties, such as gain control and adaptation, would also be of interest, though it is likely to greatly increase the complexity of the fitting procedure. Other, additions could include the responses of other neurons in an array recording (such as in recent retinal modeling work [**?**]) or explicitly modeling feedback from the spike-rate output (i.e. adaptation) or from downstream cortical areas.

The list of possible extensions to the current subunit model derive from an important principle; no quantitative model of a neural process will ever be perfect. It is easy to enumerate the deficiencies of a model just by noting the common components that are not included. More fundamentally, a model should be falsifiable, because not all of the components that *are* included will be correct, and we would like to know how incorrect they are and how they can be improved. The two largest assumptions of the subunit model are that the subunits of a cell are well localized and that their tuning properties and nonlinearities are identical. Testing these assumptions will require further experimentation with non-white datasets, as correlated inputs will be more likely to independently activate the putative subunits of the cell.

Finally, though the subunit model is at its core a *functional* model and not a detailed biophysical model, we believe its general structure reflects computations throughout the early stages of visual computation. The retina has long been described with similar mechanisms [34, 55], but later stages, such as V2 or MT (V5), are also likely to use a similar computation. There are current explorations of the subunit models to data from such areas [72, 73, 56].

Part II

Models for V2

Chapter 4

Sparse afferent models

4.1 Introduction

Models of neurons in areas V1 and V2, the largest cortical regions in the brain, are particularly important in neuroscience because they give insights into the types of transformations and abstractions that occur in the visual stream from sensory periphery to cortex. Hubel & Wiesel provided a qualitative model that describes how V1 simple cells could combine a series of LGN afferents to generate their emergent property of orientation tuning [1]. Because V2 neurons receive their predominant input from neurons in V1, there is a general expectation that the selective pooling of V1 neurons may yield V2 receptive fields with higher-order selectivities and invariances [42, 74, 75].

Some neurons in V2 are responsive to the mid-to-high level visual features that make up natural images. Subpopulations of V2 neurons are selective for the conjunction of local oriented features [23] and curvature [25]. Other neurons are modulated by scene context such as figure-ground labelling or border ownership [76] and anomalous contours [26, 77]. Freeman et al. also recently demonstrated that neurons in V2 can distinguish naturalistic image properties from spectrum-matched controls, while V1 neurons cannot [73]. Yet, despite these experiments based upon tuning preferences, most models for V2 neurons yield receptive fields that are largely indistinguishable from V1, and there are few functional models that are specifically designed to work in both V1 and V2.

Receptive field models are an important tool for understanding how visual information is transformed through the cortical hierarchy. All receptive field analysis techniques implicitly assume some family of functional models, and not all models are appropriate for all neurons. For example, Spike-Triggered Averaging (STA) assumes a linear-nonlinear cascade model [15] and is incapable of capturing the phase-invariant responses of V1 complex cells [16]. Other models, such as those based on filters derived from Spike-Triggered Covariance (STC, [12]), are too flexible because they do not incorporate enough constraints to be interpretable, making them unreliable with realistic amounts of experimental data [56]. To be both pragmatic and accurate, receptive field models should assume a meaningful architecture. For example, many neurons in V2 are selective for the orientation and position of stereoscopic edges [78], but a recent study found that only a receptive field model that combined V1-like responses with realistic output nonlinearities could describe the tuning properties of these neurons [31].

In this chapter we describe a hierarchical receptive field model that can account for response properties in both V1 and V2. The model selectively sums over a bank of simulated V1 simple cells that are tuned to local orientation, phase, position, and spatial frequency, and it is fit to individual neurons in both V1 and V2 cells. Neurons are probed with a complex stimulus and the spiking response is recorded. We then estimate the parameters of the model by finding the sparse set of the simulated V1 afferents that, when pooled together, represent the receptive field structure of the recorded neuron. We find that this model is able to accurately describe the receptive fields of most neurons in V1 and V2. The receptive fields of V2 neurons cover a diverse array of selectivity to form, and we are able to discriminate cells in V1 and V2 based upon these properties.

4.2 Methods

4.2.1 Electrophysiology

All recordings are from the first two areas of visual cortical processing, V1 and V2, in the adult macaque monkey (Macaca nemistrina and Macaca fascicularis; 8 animals). Experiments spanned 5-7 days and animals were maintained in an anesthetized and paralyzed state throughout via a continuous intravenous infusion of sufentanil citrate and vercronium bromide. Core temperature and vital signs (e.g. heart rate, end-tidal pCO2 levels, blood pressure, EEG activity, and urine quantity and specific gravity) were kept within the physiological range. Topical gentamicin was applied to the eyes and they were dilated with topical atropine. Gas-permeable hard contact lenses served to protect the eyes. Corrective lenses were then chosen via direct ophthalmoscopy to make the retinas conjugate to the experimental monitor. Experimental processes and animal care were all performed in accordance to protocols that were approved by the New York University Animal Welfare Committee, and they are in compliance with the National Institute of Healths Guide for the Care and Use of Laboratory Animals.

Quartzplatinumtungsten microelectrodes (Thomas Recording) were used to isolate single-units. The amplified signal from the electrode was bandpassed (300 Hz to 8 kHz) and routed through a dual window time-amplitude discriminator (EXPO) from which single-unit spike times were recorded at a resolution of 0.1 ms. For V1 recordings, electrodes were lowered through a craniotomy and durotomy centered between 10 and 16 mm lateral to the midline and roughly 2-6 mm behind the lunate sulcus. For V2 recordings, we extended the electrodes through the output layers (5,6) of V1, and the subsequent tract of white matter, to reach the output layers of V2. We collected units across all cortical depths for both areas, and found receptive fields that were located parafoveally, at about 5-10 degrees from the center of gaze.

4.2.2 Visual stimulation

XYT pixel-noise stimuli were presented on a gamma-corrected CRT monitor (Eizo T966; mean luminance around 35 cd/m2). The resolution of the display was 1280 x 960 pixels, set to refresh at 120 Hz. We displayed stimuli pseudorandomly at 10 Hz using Expo software on an Apple Macintosh computer (http://corevision.cns.nyu.edu), at custom sizes for each cell intended to capture both the center and part of the surround for each receptive field. The stimuli were generated as AVI movie files using MATLAB computing software (Mathworks, Natick, MA).

Stimulus frames were generated as droplet noise images. 19 droplets were spatially arranged in a hexagonal grid, where the envelope of each droplet was a two-dimensional raised-cosine, corrected so that the entire stimulus envelope had a flat-top contrast profile. White pixel-noise stimuli (usually 64 x 64) were decomposed into oriented-noise images by filtering with 4 oriented filters (orientation filters tiled frequency space in wedges of 45 degrees each), and then randomly recombined within each droplet to give a stimulus with independent local contrast and orientation content (which could include multiple orientations). Effectively, each droplet independently drew a selection of orientations and a contrast and were recombined into a global stimulus image. An example of the stimulus design is shown in Figure 4.1.

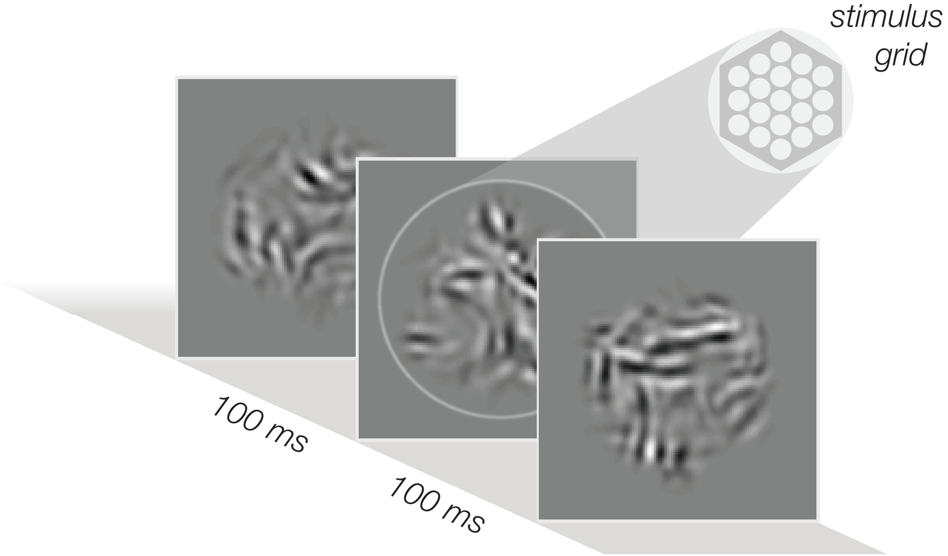


Figure 4.1: Stimuli for the V2 experiment consist of local regions with independent orientation and contrast content. Specifically, there are 19 circular locations arranged in a hexagonal grid, each with a flat-topped contrast envelope that together evenly tile space.

4.2.3 V1 filter bank

We model a population of V1 simple cells as an over-complete basis of locally oriented bandpass filters, augmented with a basis of low-pass Gaussian-envelope filters. The bandpass filters are computed with the complex-valued steerable pyramid wavelet basis [79] that generates both an odd and an even filter for each location and orientation. Each filter has an orientation bandwidth of 53 degrees, a spatial frequency bandwidth of 1.33 or 1.23 octaves (for the low- and high-frequency units), and an aspect ratio of 0.74 (Figure 4.2). We use two spatial scales of filters that together span the narrow-band frequency content of the stimulus. The high frequency filters tile space evenly on a 16 x 16 square grid, and the low frequency filters tile on an 8 x 8 grid. We use each bandpass filter to model two separate cells, one that captures the negative response and one that captures the positive response, by passing the output of each filter through two half-squaring nonlinearities ($\lfloor -x \rfloor^2$ and $\lfloor x \rfloor^2$). If the output of these two half-rectified filters are summed together along with their quadrature-phase counterparts, the result is a nonlinear unit that is invariant to local phase, identical to the energy model for V1 complex cells [16].

We use our model to characterize both V1 and V2 cells, and so we also included low-pass Gaussian filters in our simulated population of V1 afferents. These filters tile the stimulus on a 16 x 16 grid and approximate LGN neurons or layer-4 'input' neurons in V1. We believe that a realistic V1 filter bank is important for studies such as ours that build hierarchical models. Filters that are designed for compression or reconstruction (such as orthogonal wavelet bases) are good for maximizing the explainable variance for a given model class, but they are prone to artifacts. Orthogonal filters generally do not resemble real V1 units, often meaning that multiple filters must be weighted toghether, both positively and negatively, to approximate a single, smooth linear simple cell. Such bimodal weighting schemes are difficult to interpret, as there need be no real inhibitory inputs to a neuron in order to produce model fits with suppressive components. Our filter basis, in comparison, is specifically designed for analysis. Because they are overcomplete and steerable, simple linear additions of filters can reproduce simple cell filters oriented in arbitrary directions without the need for negative weights.

4.2.4 Sparse-afferent model

V1 and V2 cells are modeled as a linear combination of fixed V1 afferents (see Methods: V1 filter bank), where each afferent is composed of a linear filter and an output nonlinearity. The output of this model is a binned firing rate, calculated at an

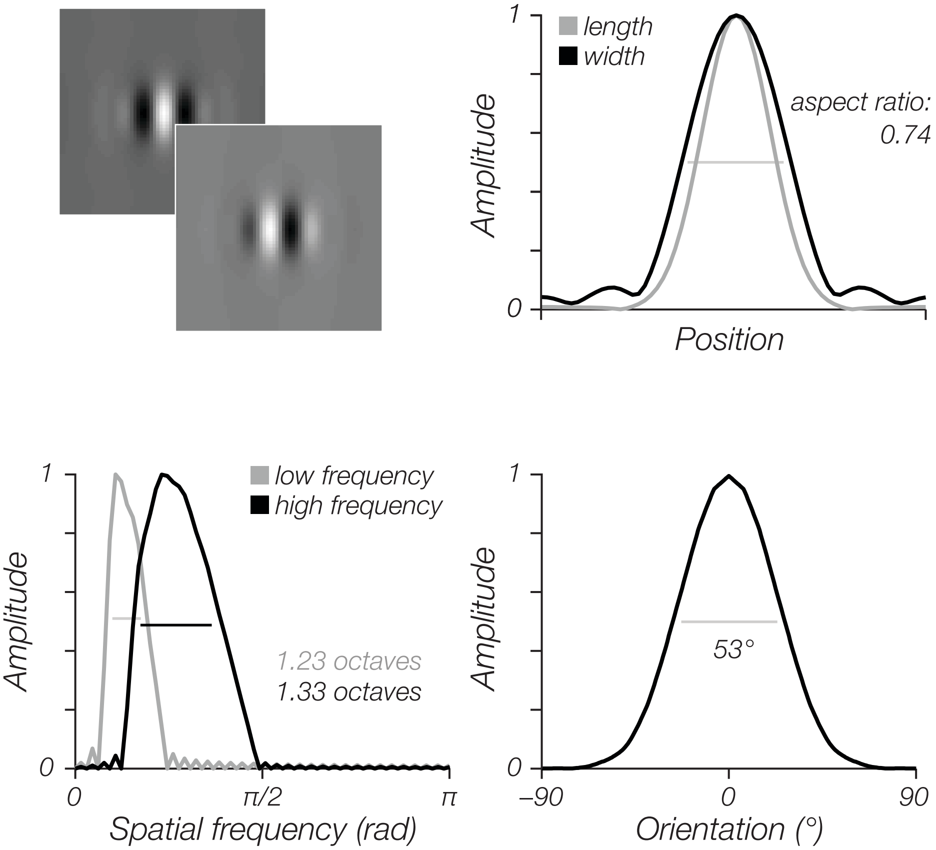


Figure 4.2: The first stage of the V2 model is a bank of V1-like simple cells at two scales, in addition to a set of Gaussian-windowed units (not depicted). [Top left] Odd an even example filters from the high-frequency scale. All distribution widths are measured at half-maximum. [Top right] Spatial envelope of the filters in length and width. [Bottom left] Spatial frequency profile of the bandpass filters. [Bottom right] Orientation tuning of bandpass filters from both scales.

optimal delay (the delay is calculated by fitting a reduced model at 5 ms delay intervals and choosing the delay with the best prediction accuracy) and can be written as

$$r = f\left(A\mathbf{w}\right), \tag{4.1}$$

where $\mathbf{w}$ is the vector of linear pooling weights, $f(\cdot)$ is the output nonlinearity, and A is a matrix of simulated V1 afferent responses. We first fit $\mathbf{w}$, then subsequently fit

f as a sigmoidal function. If B_{high}, B_{low}, and G are the bases for the high-frequency bandpass, low-frequency bandpass, and Gaussian afferent filters, A is computed from a stimulus image, **x**, as

$$A = \begin{pmatrix} \lfloor \mathbf{x} B_{high} \rfloor^2 & \lfloor \mathbf{x} B_{low} \rfloor^2 & \mathbf{x} G \end{pmatrix}. \tag{4.2}$$

By design, the afferent responses, A, are correlated because the stimulus has spatial correlations and because the filter bank is over-complete. Unconstrained Ordinary Least Squares (OLS) is not an appropriate method to estimate the linear model weights for this type of experimental design because its computation involves inverting the covariance matrix of the V1 afferents:

$$\begin{aligned} \hat{\mathbf{w}} &= \arg\min_{w} |\mathbf{r} - A^T \mathbf{w}|^2 \\ &= (A^T A)^{-1} A^T \mathbf{r}, \end{aligned} \tag{4.3}$$

which is highly sensitive to corruption by underrepresented directions in the stimulus matrix (i.e. eigenvectors of $A^T A$ with very small eigenvalues). We mitigate this problem with a regularization term that biases towards sparse solutions. Specifically, we include both a Ridge (L_2) and Lasso (L_1) term in the Least Squares objective function which forms an Elastic Net penalty [80]:

$$\begin{aligned} \hat{\mathbf{w}} &= \arg\min_{w} |\mathbf{r} - A^T \mathbf{w}|^2 + \lambda_1 |\mathbf{w}|_1 + \lambda_2 |\mathbf{w}|^2 \\ &\equiv \arg\min_{k} |\mathbf{r} - A^T \mathbf{w}|^2, \text{ subject to } [\alpha |\mathbf{w}|_1 + (1-\alpha)|\mathbf{w}|^2] < t \text{ for some } t \end{aligned} \tag{4.4}$$

In the later representation, t serves to set the overall strength of the regularization, and α governs the relative contribution of Ridge [81] and Lasso [82] penalties (see [83]). We fit both hyper-parameters by trying a fixed set within a relevant operating range and comparing their cross-validated accuracy (*reported* accuracy is computed on yet another holdout set).

In a stimulus matrix such as ours, with bandpass image content and low-frequency correlations, the Ridge penalty acts to keep the solution smooth and grouped over space and orientation, and the Lasso penalty acts to keep the solution sparse. In practice, both of these effects tend to decrease the prevalence of overfitting and they allow for good performance on novel holdout data. These constraints also have a clear biological interpretation – we expect that each downstream neuron should only pool over a small subset of similarly tuned afferents.

4.2.5 Validating the sparse model with a playback experiment

We perform a playback experiment on a subset of cells to ensure that the estimated model parameters are meaningful. The experiment is designed to generate a series of images from the model that should maximally excite or suppress the cell, and then we show these images back to the cell and record the response. To build the model we first measure the response to a stimulus that is 30 minutes long (18,000 frames). This data is used to estimate a *simplified* receptive field model that pools only over a fixed set of V1 complex cells (i.e. each model V1 afferent is phase invariant [16], unlike the full model described above that includes phase-selective responses). We simulate the response of the model to white, Gaussian noise and then use Spike-Triggered Covariance to determine the 10 images (5 excitatory, 5 suppressive) that will most excite or suppress the model. The relative order of the covariance matrix eigenvalues determines the relative strength in which the images should modulate the cellular response. Assuming that throughout this analysis procedure the isolation of the cell is maintained, we then 'play back' these images to the cell and record the firing rate. There should be good agreement in the rank order of the stimuli and the neuronal firing rate if the model is a good fit for the cell.

Besides showing each individual image by itself, we also show pairs of images to assess interaction terms. The design matrix for the experiment is 10 x 10, because each image is also superimposed on every other image, where the diagonal of this matrix represents each image in isolation. We randomize the phase of each image presentation, whether presented in isolation or in pairs. Each stimulus condition is sampled 48 times.

4.2.6 Calculating statistics of the receptive field models

A model for cortical receptive fields is constructed from the linear summation of V1-like afferents. These afferents are local in both space and frequency and are characterized by spatial position, orientation, and phase. We can compute a set of informative statistics for each cell by examining the set of afferents that the model selectively pools. For example, we can examine the relative influence of excitatory drive to suppressive drive by summing the magnitude of the positive weights and the negative weights into two separate pools (E for excitation, S for suppression) and computing an index of their relative contribution:

$$\text{excitation index} = \frac{E - S}{E + S}. \tag{4.5}$$

A cell with an equal balance of excitation and suppression will have an excitation index of 0, while one that has no suppression will have a value of 1.

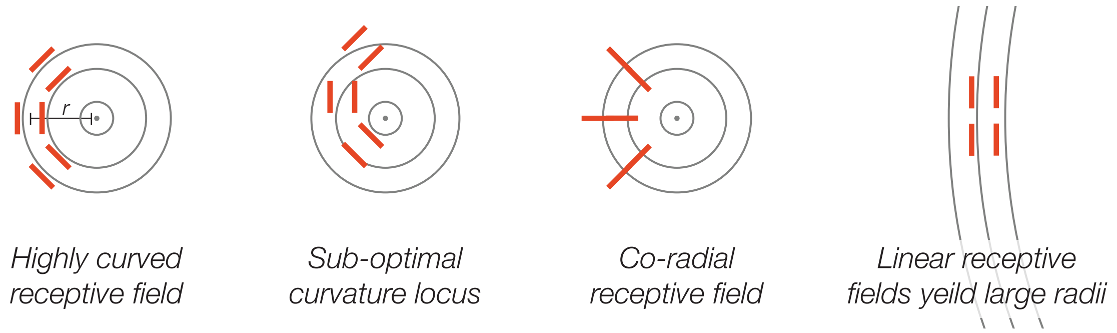
Local statistics characterize the average tuning properties of afferents at each location in space. In each discretized receptive field location there are 32 possible afferents from which to pool (8 orientations at intervals of ${}^{\pi}/_{8}$, and 4 phases), and there are thus 32 estimated weights to these afferents, $\mathbf{w}$. We can estimate how tuned the cell is for local orientation by measure the orientation selectivity at each location in space, which is computed as the circular variance of the weights, averaged over phase [84],

$$\mathrm{CV} = 1 - \frac{|\Sigma_k w_k e^{i2\Theta_k}|}{\Sigma_k |w_k|}. \tag{4.6}$$

A value of 1 indicates a flat, non-selective orientation tuning profile and a value near 0 indicates a highly tuned location. (Note that this particular definition of circular variance ignores phase and treats negative weights as identical to an excitatory weight for the orthogonal orientation). Average local circular variance is a summary over the entire receptive field by taking the weighted average over all spatial locations. Similarly, local phase selectivity can be computed as a the circular variance of local phase weights, with a weighted average over local orientation and spatial position.

Global statistics consider receptive field tuning across the entirety of the receptive field. Global orientation homogeneity is computed in a similar manner to local orientation tuning, but it computes tuning over the entire receptive field at once. Specifically, we compute the circular variance of the entire receptive field regardless of spatial position. We find that this measure of global orientation homogeneity is qualitatively similar to a method that first computes the local orientation of each receptive field location and then calculates circular variance over these summary values.

Receptive field curvature is another informative measure of global structure. Imag-



Highly curved receptive field

Sub-optimal curvature locus

Co-radial receptive field

Linear receptive fields yeild large radii

Figure 4.3: We measure curvature tuning by fitting concentric circles (gray) to the local orientation of the receptive field (red). We find the best locus of the concentric circles for each receptive field by trying all possible locations. The distance between the locus of the curves and the receptive field center-of-mass defines the preferred curvature radius. Cells with a co-radial arrangement will not be fit well by curves. Parallel receptive fields will also be well fit by concentric circles, but the radius will be near-infinite.

ine a series of concentric circles emanating from a particular locus somewhere near a receptive field (Figure 4.3). We can determine how well the receptive field is fit by these circles by angular difference between each receptive field segment (i.e. oriented afferent) and the local tangent of the circles at each point. Goodness of fit is defined as the weighted average of the angular correlation over all locations and determines how well the field conforms to that particular type of curvature (1 is the best, 0 is random). We find the best locus for each receptive field by trying all possible locations of the concentric circles and report this value as the curvature index. This also gives us an estimate of the curvature radius of the receptive field by computing the distance between the receptive field center-of-mass and the best concentric circle locus (to compare between cells, we normalize this metric to be in units of stimulus diameters). Note that cells that prefer a parallel edges will also be fit well by curves, but the radius will be nearly infinite. We fit a similar Parralelism index by determining how well each receptive field matches a parallel field of afferents.

The envelope of a receptive field provides information about how the receptive field is distributed over space. We measure the envelope as average of the weight magnitudes at each location, and it can be parametrically summarized with fit to a Gaussian distribution (see section 2.2.2). The goodness-of-fit tells us something about the complexity of the receptive field shape. Receptive fields that are well fit can be described as roughly circular or elliptical. We can also measure the aspect ratio, which is the ratio of the length of the receptive field to it's width (i.e. the major and minor axes of the Gaussian covariance ellipse).

4.2.7 Discriminating V1 and V2 receptive fields

We would like to know if V1 receptive fields are different than V2 receptive fields. The general procedure is to determine if the distribution of receptive fields statistics (see the previous section) is different for V1 and V2 neurons. We start by constructing a matrix of receptive field statistics that has as many rows as there are neurons and as many columns as the number of statistics that we choose. A simple, unsupervised method to visualize this space is to perform Principle Component Analysis (PCA) on the normalized distribution matrix (i.e. zero mean and unit variance) to reduce the dimensionality and then see if there is clustering by cell type. PCA serves to find the linear combinations of statistics that vary together the most, and does not require any prior knowledge about which statistics belong with which cell type. We project the statistics distribution onto the first two principle components and measure the difference between V1 and V2 cells.

We can also take a supervised approach to distinguishing V1 and V2. By using the labels for each data point (i.e. V1 or V2), we can find a projection in the multi-dimensional statistics distribution that optimally discriminates V1 and V2. This linear projection is known as the Fisher Linear Discriminant, and is calculated as

$$\mathrm{FLD} = C_w^{-1}(\mu_{V2} - \mu_{V1}). \tag{4.7}$$

C_w describes the average within-class covariance matrix and μ is the mean of the statistics distribution for V1 or V2. We can then find an optimal boundary along the discriminant by trying all possible boundaries between the data points. Percent correct is defined as the average correct classifications for V1 and V2.

As the dimensionality of the statistics distribution grows, it becomes easier to find a linear projection that can optimally discriminate two subpopulations. We construct a null hypothesis through permutation to get an idea of how well we are doing compared to chance. Specifically, we shuffle the data labels (V1 or V2) for all data points and construct an optimal discriminator for this altered distribution. We perform this procedure 10,000 times and generate a null distribution of percent-correct estimates. We then determine whether or not our test statistic, measured with the un-shuffled data, is greater than 95% of the null estimates.

4.3 A hierarchical model for neurons in V1 and V2

We analyze the activity of 144 neurons in visual cortex (43 in V1 and 101 in V2) in response to images with random local orientation and contrast content (Figure 4.1; see Methods: Visual stimulation). Stimuli were presented at 10 frames per second and the measured spike times were binned at this same frequency, but with an optimal delay. We then fit a hierarchical model for each cell whose goal was to take the stimulus as an input and use it to reproduce the observed firing rate. The model consisted of a fixed V1-like stage that filtered the stimulus with locally oriented neural units, and a

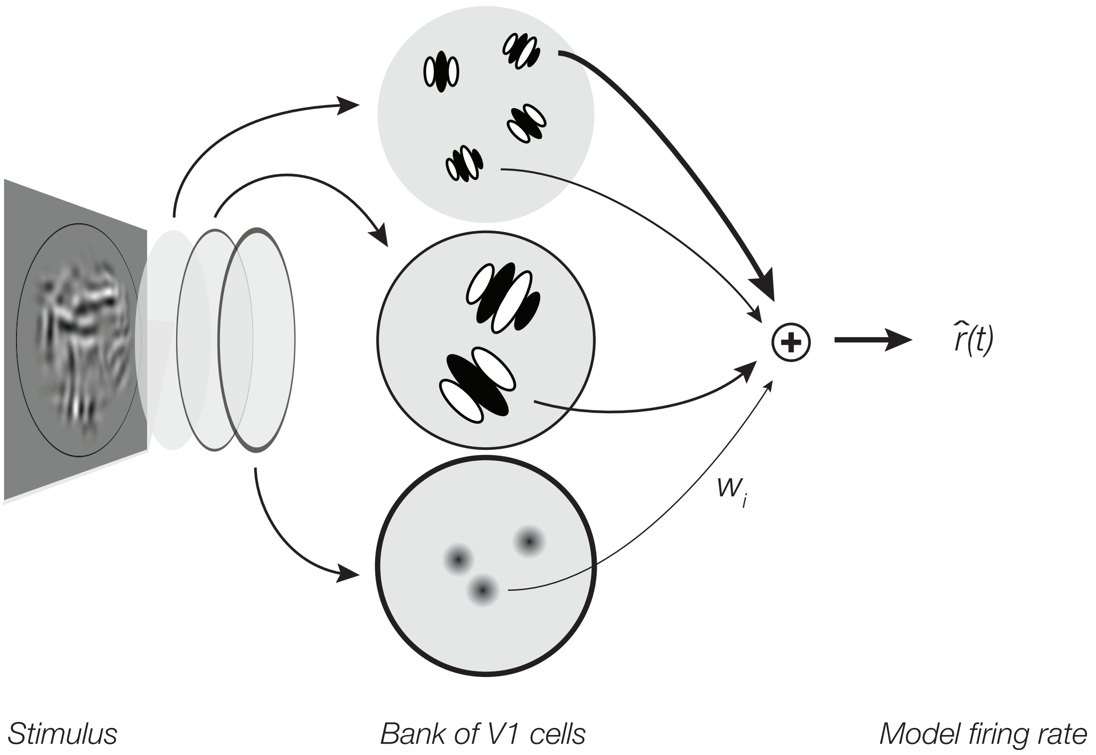


Figure 4.4: Sparse-afferent V2 model. A stimulus is decomposed into the responses of a set of V1-like afferents. The overcomplete basis for the afferent's linear filters come in three families: high-frequency bandpass (top), low-frequency bandpass (middle), and Gaussian (bottom). The bandpass filter responses are half-squared. These afferent units are combined together with a sparse weighting scheme to generate a model output.

second stage that took sparse linear combinations of these afferents.

Neurons in V2 receive their predominant feedforward input from V1 neurons, yet while the basic properties of V1 receptive fields are well known, the characteristics of V2 receptive fields remain enigmatic. V1 simple cells possess oriented, narrowband linear filters that are localized in space [6]. V1 complex cells combine multiple simple cells of different phases and positions to generate responses that are invariant to precise feature location [16, 56]. This well defined architecture provides a convenient substrate for modeling neurons downstream of V1; first, model a fixed bank of V1 filters, along

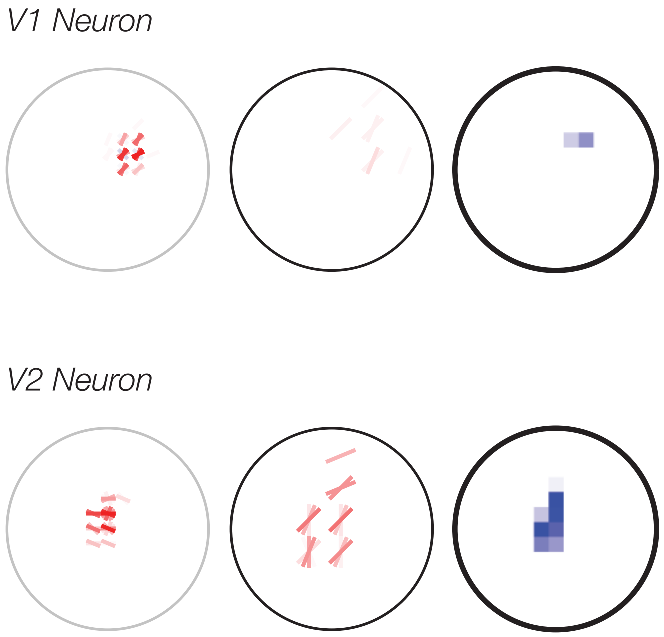


Figure 4.5: Example cells. Each cell is defined as the overlap of three set of V1-like afferents (separated for illustration purposes), where the large circles indicate the rough location of the stimulus in space and oriented line segments and pixels represent simulated afferents. The left column corresponds to the high-frequency bandpass afferents, the middle column represents the low-frequency band-pass units, and the right column represents the low-pass Gaussian units. Red colors are positively weighted and blue colors are subtractive. The V1 neuron prefers oblique orientations with a small amount of linear suppression. The V2 neuron has a different orientation preference for high- and low-frequency image content.

with a standard output nonlinearity, and then determine which afferents are selectivity pooled together to generate the observed firing rate of a neuron in V2.

For the model to perform well, the V1 filter bank should be complete or overcomplete to ensure no loss of stimulus information. Furthermore, each filter should match the general properties of V1 neurons as closely as possible. The most prominent features of real V1 cells are that they are well localized in both space and frequency preference, and they are tuned to orientation. We use two scales of a well known wavelet basis (the Complex Steerable Pyramid; [79]) to capture these basic properties, providing a set of even- and odd-phase filters that evenly tile space and frequency (Figure 4.2; the outputs of these bandpass filters are half-squared to create realistic V1 unit responses). This basis is augmented with a set of unimodal Gaussian filters to also capture the LGN-like input into V1 cells. A diagram of the model is depicted in Figure 4.4. The generality of this framework provides flexibility and ensures that the model is capable of describing both V1 neurons and V2 neurons; degenerate models that pool over only one or a few afferents can approximate simple or complex cells.

Because the V1 stage of the model is a fixed set of linear-nonlinear (LN) computations, and the V1 unit responses are fixed given a particular stimulus, fitting the full hierarchical model simply amounts to fitting the linear pooling function from the V1 units to the output firing rate. We use the Elastic Net to solve this linear regression ([80]; see Methods). This algorithm finds sparse solutions with the minimal amount of V1 unit afferents needed to explain the output firing rate by assuming that most input units are not weighted at all. We cross-validate the afferent weights and the hyper-parameters that control sparsity to avoid overfitting.

The model parameters fit to an example V1 cell and V2 cell are presented in Figure 4.5. The parameters that correspond to each type of afferent unit are plotted separately (left, high-frequency bandpass V1 units; middle, low-frequency bandpass V1 units; right, low-pass Gaussian units). Each large circle corresponds to the approximate location of the stimulus in space, and each line segment or pixel represents an afferent unit from the model V1 filter bank. The orientation of the line segment depicts the orientation preference of the V1 unit and its color indicates how heavily it is weighted by the model (red is positive, blue is negative). Thus, the V1 neuron (top) is selective for stimuli that have a local orientation of 45 degrees. The V2 neuron (bottom) shows selectivity for horizontal orientations at high spatial frequencies, but oblique orientations

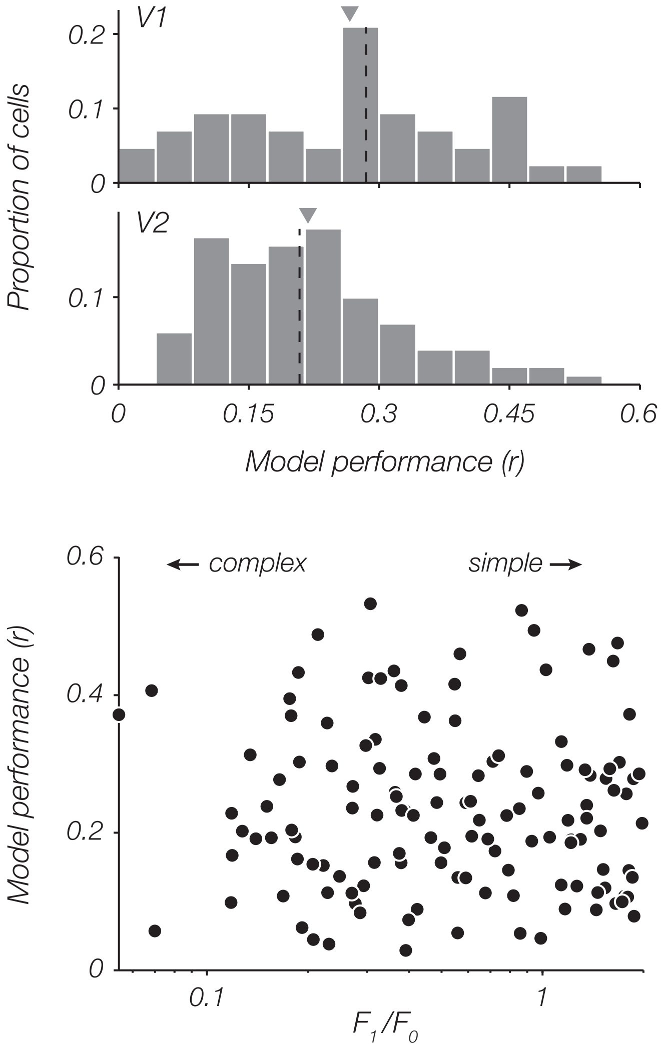


Figure 4.6: Model performance for each cell is defined as the correlation between the measured firing rate and the model predicted rate (r). V1 cells ($n = 43$) are are fit better than V2 cells ($n = 101$) on average (top). In this and following figures, arrows indicate the mean of each distribution and the dotted lines indicate the median. In both V1 and V2, cell sensitivity to phase, as measured by the F_1/F_0 ratio at the optimal orientation, is not correlated with model fit (bottom).

at lower spatial frequencies.

We measure cross-validated model performance, or accuracy, as the correlation coefficient between the observed firing rate and the model-predicted firing rate on

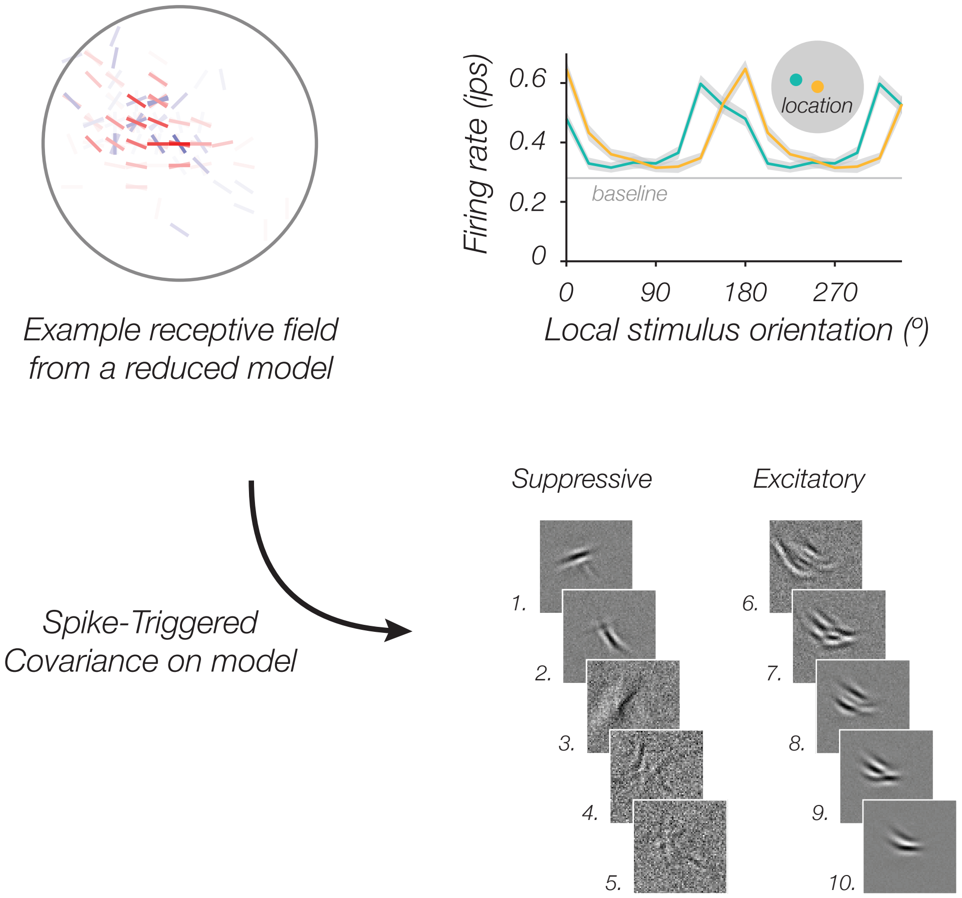


Figure 4.7: Playback experiment procedure. For an example V2 cell, we estimate the parameters of a reduced model that pools over phase-invariant afferent neurons (top left). This cell appears to be tuned for curves, which we confirm by binning the cell's firing rate according to the local stimulus orientation at two different locations. The two locations have tuning curves with different preferred orientations (top right; error bars are SEM). We generate rank-order stimuli for the playback experiment by passing white-noise into the model and computing the STC filters from the response.

holdout data. On average, this model obtains higher accuracy for V1 neurons than V2 neurons ($< r_{\mathrm{V1}} >=$ 0.27 is greater than $< r_{\mathrm{V2}} >=$ 0.22, $p <$ 0.05 , t-test), but there is no correlation with cell phase selectivity as measured by F_1/F_0 ($r =$ -0.04, $p >$ 0.05; Figure 4.6). We note that these r-values are considerably lower than those reported by other groups, particularly Willmore et al. who used the Berkeley Wavelet Transform

(BWT) to decompose their images rather than a realistic model V1 population [29]. However, the accuracy values for these two models are not directly comparable because of a difference in cross-validation methodology. Willmore et al. take the mean of ten responses to a set of novel images to obtain a noise-reduced validation set. Our model is validated on a noisy single-trial images, similar in kind to the training data (if as a control, we use the BWT in place of our Gabor-like V1 units within our model, accuracy decreases significantly by 30%; $p << 0.05$, t-test; see section 4.5).

4.4 Validating the sparse-afferent with playback and tuning experiments

We validated our experimental paradigm with a *playback* experiment (see section 4.2.5). For a subset of V2 cells (n=19) we estimated the parameters of a simplified model and and used the model to generate a set of rank-order images that should uniquely drive or suppress each cell. We measured the tuning to these stimuli by presenting the images back to the same cell at various phases and in various pairwise combinations (see Methods). We demonstrate this procedure with an example cell in Figure 4.7.

The example cell, fit with a phase-insensitive, one-scale model, appears to show a preference for curves. Indeed, local tuning functions, triggered on the stimulus orientation at two independent locations, show a distinct difference in orientation preference as a function of position (Figure 4.7, top right), where local stimulus orientation is computed from the vector average of Steerable Pyramid filter activity. The playback stimuli for this cell, which are five excitatory and five suppressive image sets, are created by performing spike-triggered covariance [12] on simulated responses from the model

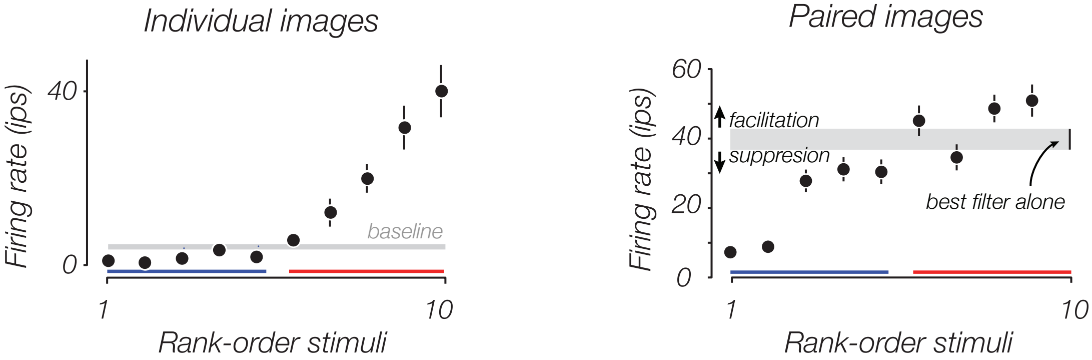


Figure 4.8: We show the playback images (Figure 4.7) to the example cell and measure the firing rate (left). The five suppressive images (blue) tend to suppress firing below baseline, and the five excitatory images (red) generate increased firing in the same magnitude order as predicted by the model (error bars are SEM). Next, we pair each image with the most excitatory image (right). This has the effect of raising the baseline firing rate and uncovering the strong suppressive effects of the first two stimuli. The excitatory stimuli weakly facilitate the response.

to white noise stimuli. The eigenvalues of the STC matrix determine the rank order in which these images should drive and suppress the cell; for this example cell there is a strong correlation between the cells firing rate and the image rank order ($r_{Spearman} =$ 0.98, $p <<$ 0.05; Figure 4.8, left). In isolation it is difficult to assess how strongly the suppressive images affect a cells response because cells cannot fire less than zero spikes per second. If we look at the neuronal response to each image in combination with the cells preferred image, we can get a more complete picture of cellular suppression. The rank-order response to the playback images in the paired paradigm for the example cell is well preserved ($r_{Spearman} =$ 0.97, $p <<$ 0.05; Figure 4.8, right).

Over all cells, the playback experiment confirms that sparse afferent models captured relevant spatial features of V2 cells. The average rank-order correlation between the ten test images and the observed firing rate was 0.85 (Figure 4.9, top left). The average rank-order correlation for each image paired with the most excitatory image

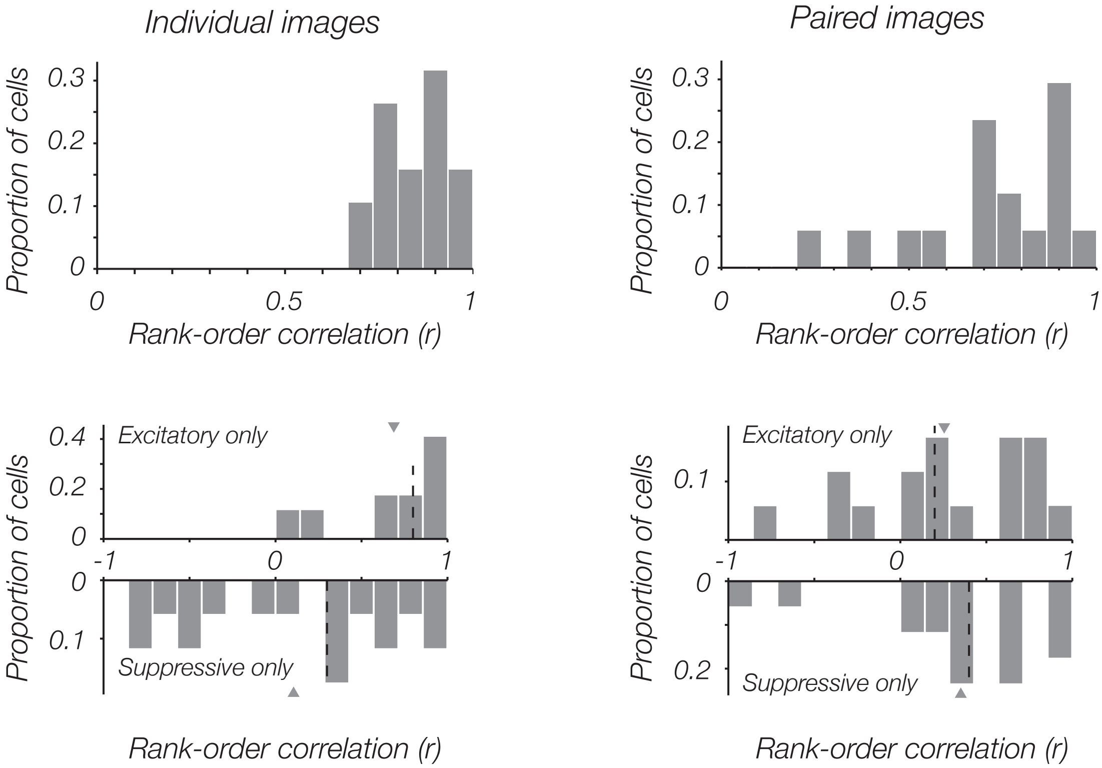


Figure 4.9: Average playback performance for 19 V2 cells. [Top] Average Spearman rank-order correlation for images presented alone (left) and images paired with the most excitatory stimuli (right). [Bottom] Performance among the solitary stimuli is best for the excitatory images (left) because they are not obscured by sub-threshold membrane potentials. For the paired stimuli, suppressive images are the most correlated with observed firing rate (right), and we see little correlated facilitation.

was 0.72 (Figure 4.9, top right). We can also examine these two measures for only the excitatory images and only the suppressive images (Figure 4.9, bottom). Again, there is a positive correlation between cell response and image rank order, but for the solitary stimuli, the excitatory images provide the best correlation ($r_{excitatory} = 0.69$ versus $r_{suppressive} = 0.11$), and for the paired stimuli, the suppressive images provide the best correlation ($r_{excitatory} = 0.26$ versus $r_{suppressive} = 0.35$).

For each cell we also measured direction tuning curves with drifting sine wave gratings. The DC response (F_0) and phase-tuned response (F_1) are plotted for an

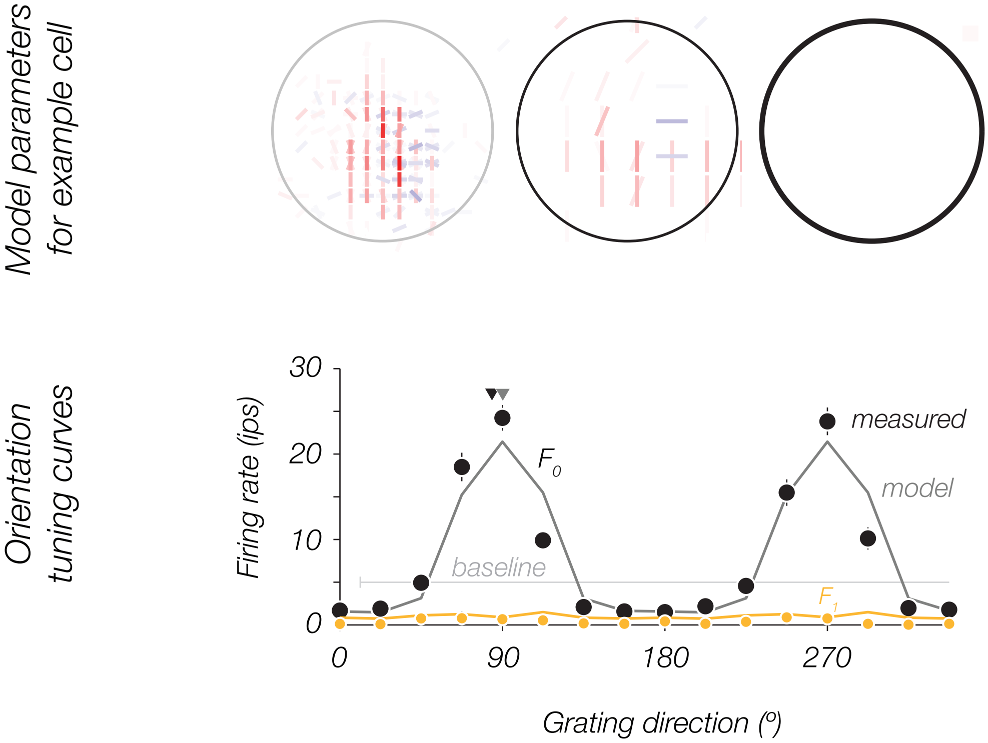


Figure 4.10: We use the receptive field model for an example V2 cell (top) to predict the direction tuning curve (bottom, solid lines). There is good agreement with the tuning curve as measured with drifting gratings (dots). Black points and the gray line depict average tuned response (F_0), while the yellow points and line depict the phase-tuned response (F_1). Error bars are SEM.

example complex V2 cell in Figure 4.10 (F_1/F_0: 0.019). The model for this cell can be used to generate a predicted orientation tuning curve and it shows good agreement with actual tuning (the model does not include time-varying responses and thus it cannot differentiate drift direction. We allow for independent scaling and offset of the model-predicted tuning curve for illustration purposes). Across all cells, the model is capable of reliably predicting preferred orientation tuning (Circular correlation = 0.53; Figure 4.11). The models also predict the circular variance of the tuning curves ($r =$ 0.71, $p <<$ 0.05), but they do less well at predicting their phase selectivity ($r =$ 0.26, $p <$ 0.05; not shown). There is no bias in predicting orientation tuning ($n.s.$, $p =$ 0.88,

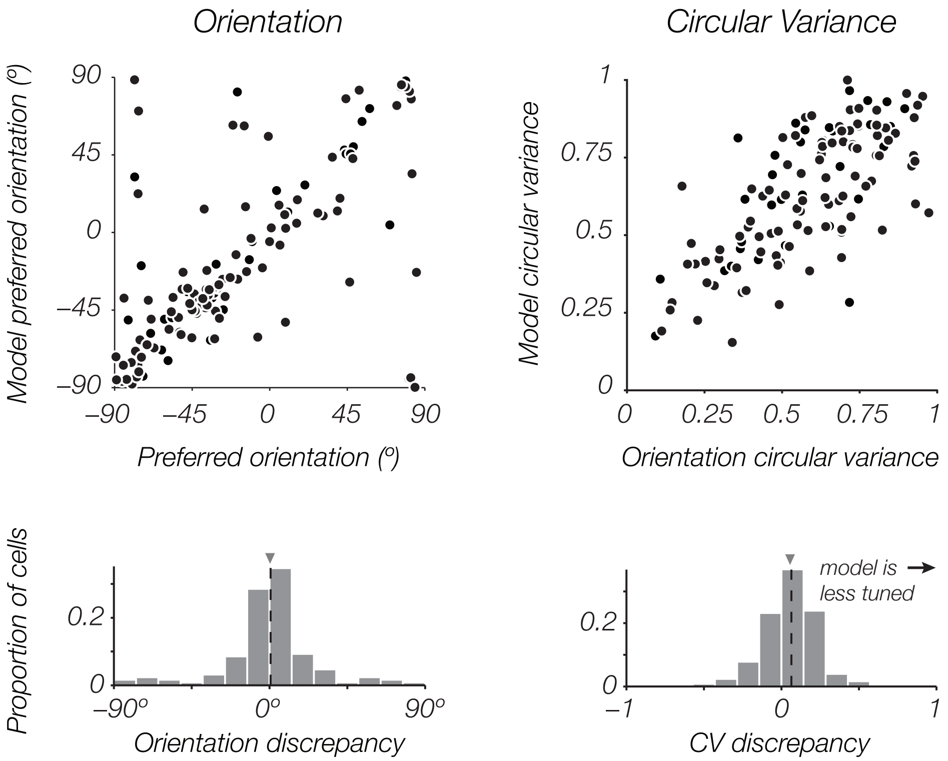


Figure 4.11: Over all cells in V1 and V2, actual orientation preference as measured with drifting gratings matches the direction preference predicted from the model. There is no systematic bias in the error. The circular variance of the model also closely matches the circular variance of the actual tuning curve, but tends to predict slightly flatter tuning curves on average.

t-test; Figure 4.11, bottom), but the model tends to predict a flatter tuning curve than observed in tuning to drifting gratings ($p << 0.05$, t-test).

4.5 Sensitivity to design aspects of the sparse-afferent model

The sparse-afferent model captures important tuning properties of cells in V1 and V2, but a number of design choices were made in the construction of the model. For

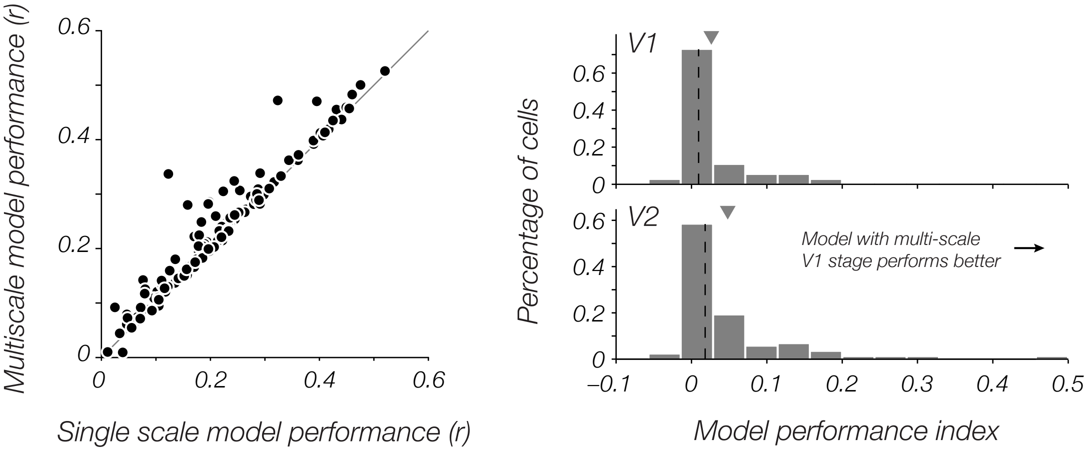


Figure 4.12: [Left] A model with only the high-frequency bandpass V1 filters (abscissa) underperforms the full model that includes low-frequency bandpass and low-pass filters (ordinate). This is pronounced for only a subset of cells. [Right] The improvement gained by using the 3-scale afferent bank is similar between V1 and V2.

example, the model uses a multi-scale bank of V1-like afferents that are constructed to resemble the local, oriented filters found in V1, and their output is half-squared. How sensitive is the model to these parameters?

We test model sensitivity to scale by constructing a reduced model that pools over only a single scale of V1 afferents. Specifically, we remove the low frequency bandpass filters and the low-pass Gaussian filters and refit the model. Because the single-scale model is a subset of the multi-scale model, we predict that the multi-scale model will perform better, but we are interested to see for how many cells this is true and if there is a difference between V1 and V2. We find that while the multi-scale model does improve the fit quality for most cells, substantial improvement is found for only a subset of cells (Figure 4.12). We calculate an improvement index as $(r_{multi} - r_{single})/(r_{multi} + r_{single})$ and find no significant difference between in V1 and V2 ($p = 0.11$, Figure 4.12, right).

We also consider the model parameters chosen by Willmore et al. in their sparse model of V2 receptive fields [29]. The Willmore model is conceptually similar to

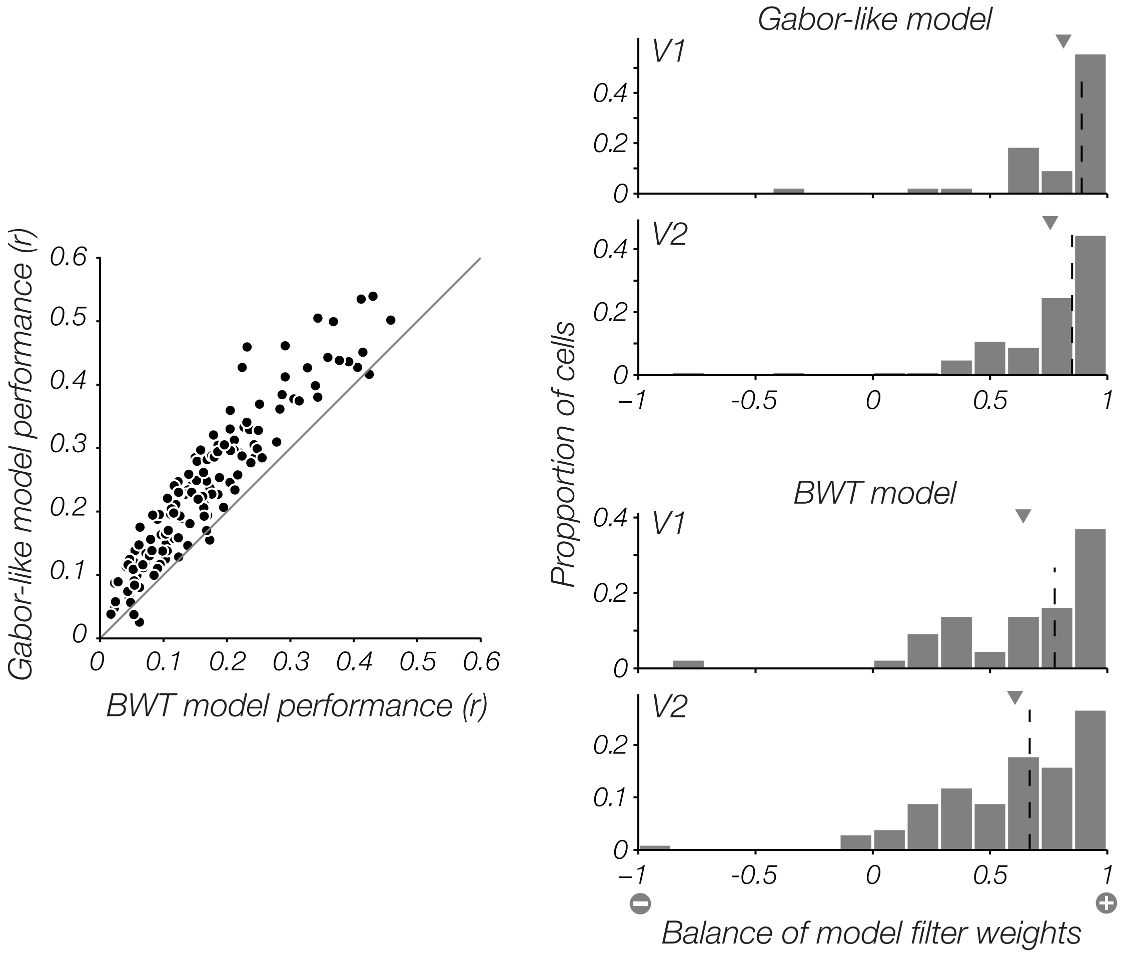


Figure 4.13: [Left] A model with BWT V1 filters (abscissa) underperforms the model Steerable Pyramid Wavelet filters (ordinate). [Right] The balance of model filter weights is an index that is 1 if all weights are excitatory, -1 if all weights are suppressive, and 0 if there is an equal balance between the two. Models that use the BWT filters have an increased amount of suppressive weights for both V1 and V2 cells.

ours because they both find a sparse weighting function over a series of LN afferents. However, their model uses LN filters from the hard-edged BWT (Berkeley Wavelet Transform [30]), and a hard half-rectifying nonlinearity. In contrast, our model uses a half-squaring nonlinearity, and filters from the SPWT (Steerable Pyramid Wavelet Transform [79]) that are well localized in space and frequency.

If we replace the afferent filters in our model with the BWT filters, we find that model performance significantly decreases (-30%, $p << 0.05$; Figure 4.13). Some of the difference in performance is due to overfitting, because the models with the BWT filters overfit more than our models (fitting performance on *training* data for the BWT model increases the r-value by 42% over the cross-validated data, versus 32% for our model). Willmore et al. also find that there is more suppression in V2 cells versus V1 cells, which is quantified by constructing a channel balance index that compares the sum of the excitatory weights and inhibitory weights (see Methods, section 4.2.6). With our model on our data, we find much less suppression for both V1 and V2 cells than Willmore et al. report. However, if we construct a model with BWT afferents on our data instead, we find significantly more suppression than in our model with Gabor-like SPWT filters ($p < 0.05$, t-test; Figure 4.13, right bottom). To us, this suggests that the real afferents being pooled by V2 cells do not resemble the BWT filters, and so the model with these afferents requires a greater number of suppressive inputs to adequately 'shape' the receptive field.

Finally, we investigate model sensitivity to the choice of V1 afferent output nonlinearity. We chose to implement a half-squaring nonlinearity so that cells that are truly phase invariant could be modeled as a linear sum of phase-quadrature filters (from the identity $cos(x)^2 + sin(x)^2 = 1$; see [16]). If we substitute half-wave rectification for an example cell we find that though the basic tuning appears to remain constant, the hard rectification model exhibits much stronger suppression than the half-squaring model (Figure 4.14). Over all cells, the percentage of non-zero weights for each model averaged about 12% and is not significantly different between the models ($p = 0.64$; t-test), but the half-wave rectified afferent stage yields significantly more suppressive weights on average (Figure 4.15).

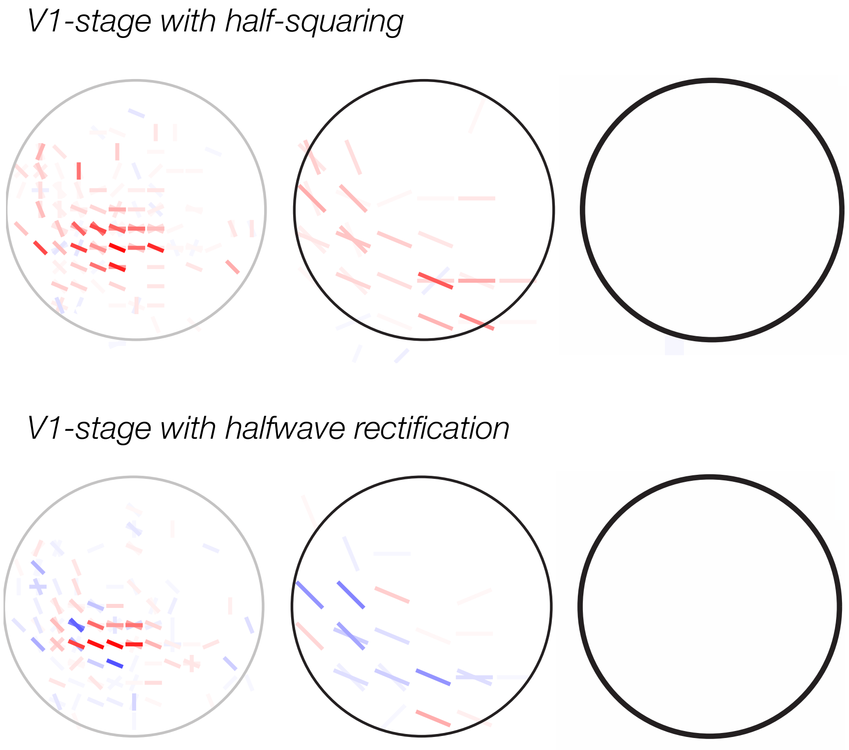

V1-stage with half-squaring
V1-stage with halfwave rectification

Figure 4.14: An example cell plotted for two models with different afferent nonlinearities. The model with half-wave rectification has many more suppressive weights than the half-squaring model, though the overall tuning is similar.

4.6 Properties of V1 and V2 receptive fields

Receptive fields in V1 and V2 range in complexity, from those that possess homogeneous tuning over a well localized region of space, to those that demonstrate selectivity for a conjunction of simple features. Most cells in V1 prefer images with homogeneous tuning properties over space (Figure 4.5). These are reminiscent of the classic simple cells and complex cells that were originally described by Hubel & Wiesel. Like in V1, many V2 cells show selectivity for a single orientation, but others show a preference for curves (Figure 4.16, top), multi-scale patterns (Figure 4.16, bottom), T-junctions (Figure 4.17, top), and other types of contour or texture-like patterns (Figure 4.17,

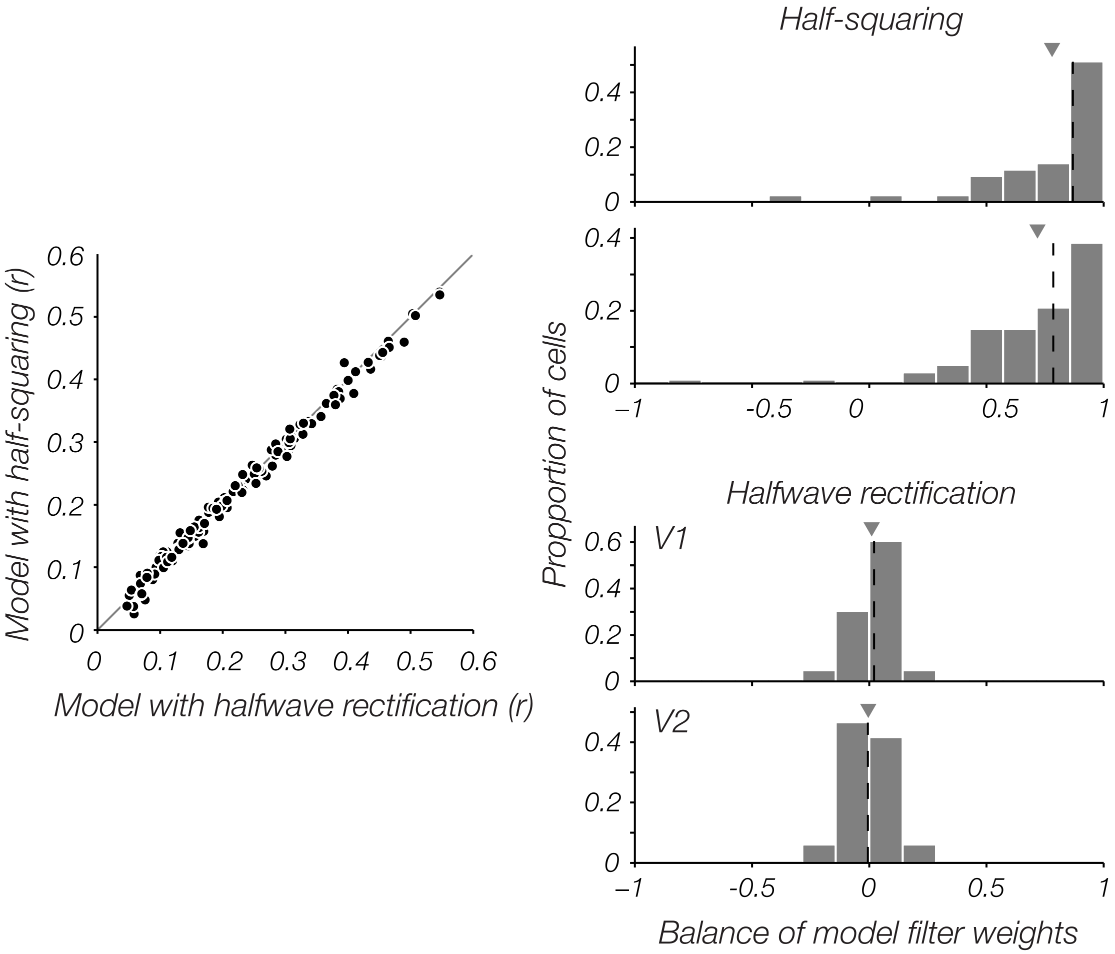


Figure 4.15: Models with half-wave rectification have roughly comparable performance to models with half-squaring nonlinearities. However, half-wave rectified models have significantly more suppression than half-squaring models.

bottom). We summarize model receptive fields with a series of shape statistics (see Methods, section 4.2.6) and examine the difference between neurons in V1 and V2.

V2 neurons do not pool over model afferents in the same way as V1 neurons. A simple observation is that V2 neurons sum over more afferents than V1. The model contains 6432 V1-like afferents, and V2 cells require a significantly greater percentage of these units to describe their receptive fields ($p << 0.05$; t-test; Figure 4.18, left). On average, V1 cells pool over 7.7% of the afferents and V2 cells pool over 13.5%.

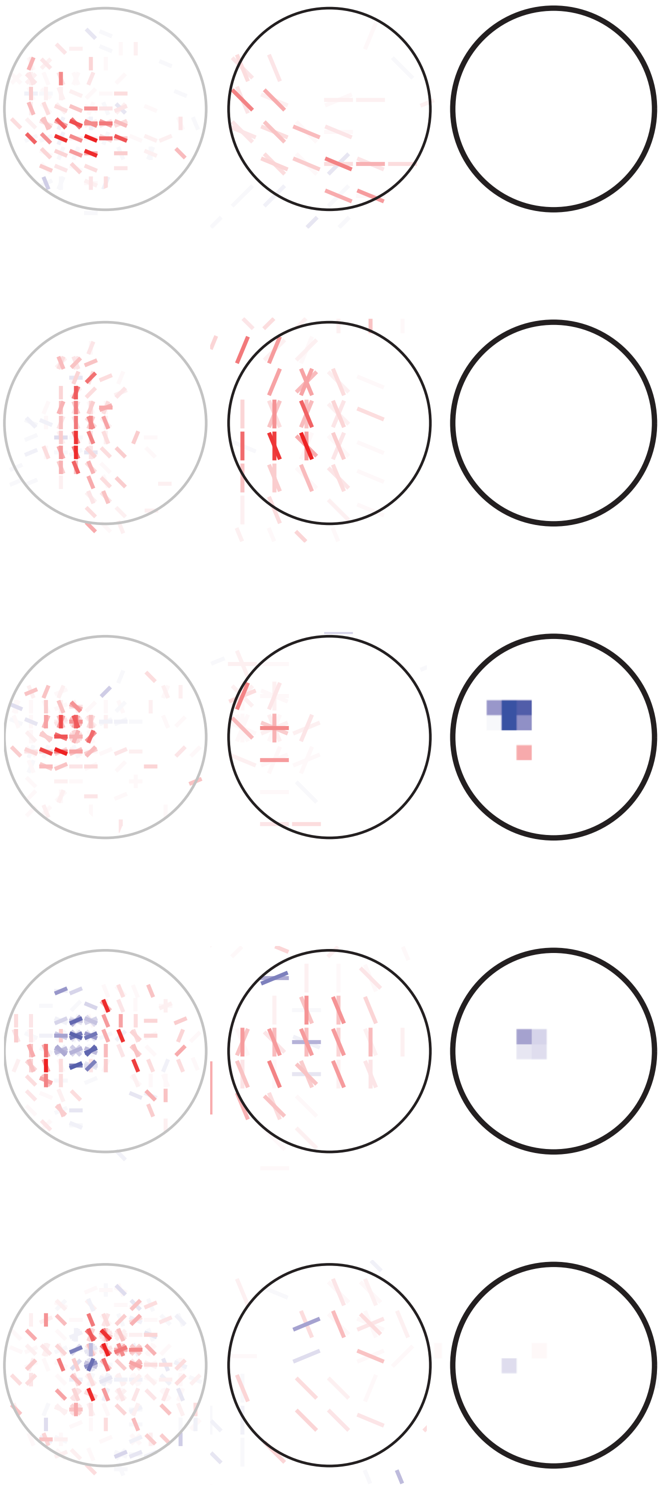


Figure 4.16: Example V2 receptive fields (I). Plotting conventions follow Figure 4.5

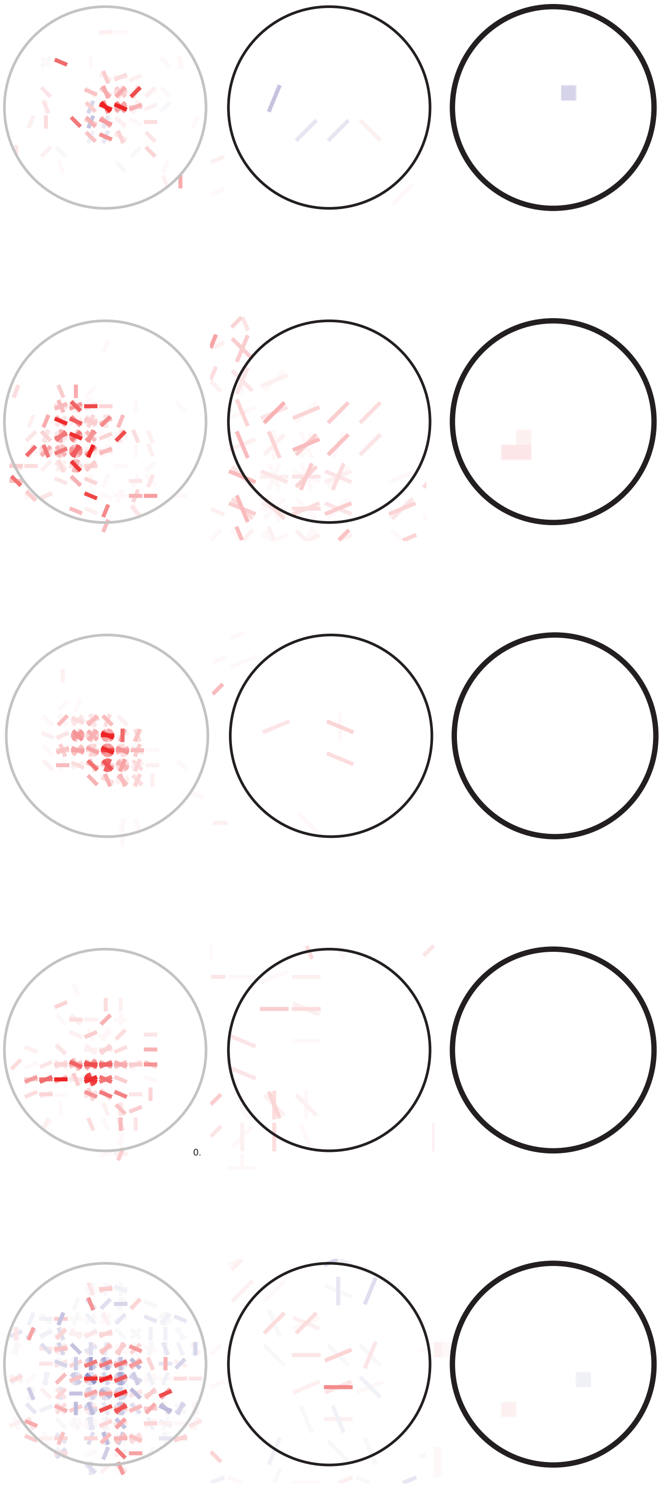
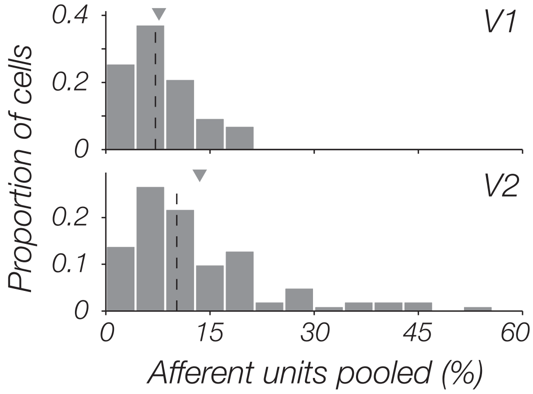


Figure 4.17: Example V2 receptive fields (II). Plotting conventions follow Figure 4.5



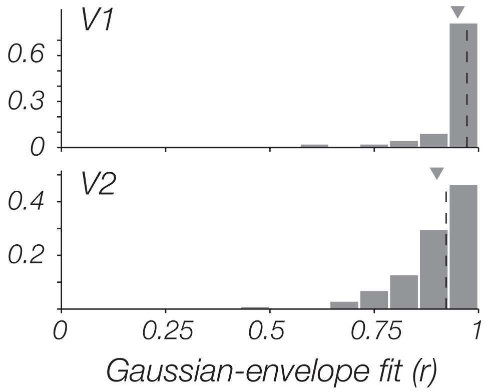


Figure 4.18: V2 cells pool over significantly more afferents, as measured by the percentage of V1-like units with non-zero weights in receptive field models (left). The spatial profile of V2 receptive fields are also less likely to be well described by a simple 2 dimensional Gaussian profile (right).

The two areas also differ in the spatial profile of their receptive fields. For each cell we sum the afferent weights at each spatial position to create an envelope of the receptive field in space. We then fit a 2 dimensional Gaussian to this distribution. V2 cells are significantly less well described by a Gaussian envelope ($p <$ 0.05, t-test; Figure 4.18, right).

V2 neurons also tend to be less tuned to oriented features than V1 cells, though the differences are not significant. We can measure orientation tuning for the model parameters in an analogous procedure to measuring orientation tuning to drifting gratings. Circular variance captures the width of the tuning curve, with higher values indicating broader tuning and less selectivity. The (weighted) average circular variance over all locations in a receptive field describes how strongly each cell is tuned to local orientation. V1 cells tend to be slightly more tuned that V2 cells with a mean local circular variance of 0.23 versus 0.26 ($p = n.s.$, t-test; Figure 4.19, left). Computing circular variance simultaneously over the entire receptive field returns a global measure of orientation homogeneity over space. V2 cells are also more heterogenous over

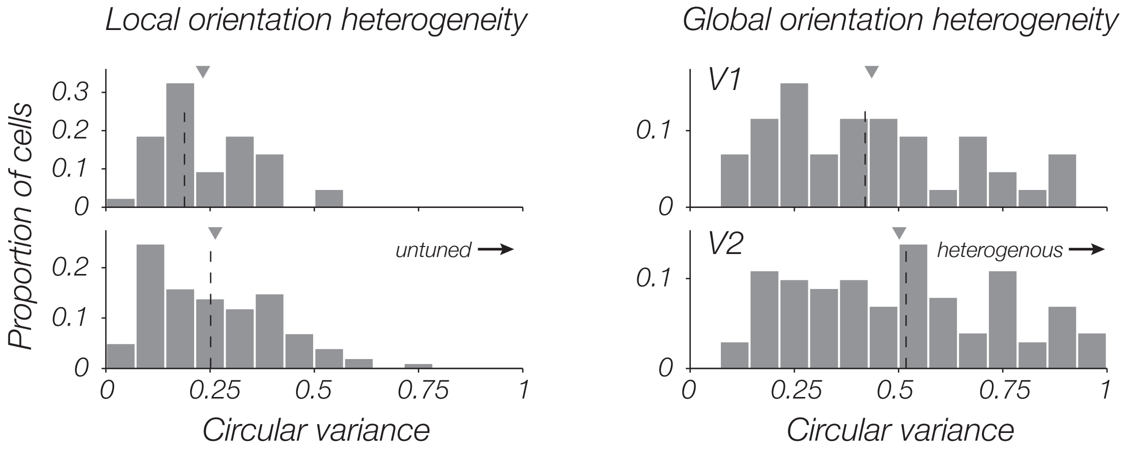


Figure 4.19: Orientation tuning can be measured with circular variance, where flat tuning curves have high variance. As measured from the model, V2 cells tend to have broader average orientation selectivity at each local position in their receptive field (left). Globally across all position in the receptive field, V2 cells also tend to be less homogeneous in their orientation selectivity (right). Neither of these differences are significant.

space, but again the difference in not significant ($p = n.s.$, t-test; Figure 4.19, right). Local and global tuning measures are heavily correlated with each other (r = 0.84, $p <<$ 0.05; t-test), and are both correlated with the circular variance measured from drifting-grating tuning curves (r_{local} = 0.52, $p <<$ 0.05; r_{global} = 0.55, $p <<$ 0.05; t-test).

Sitting in the middle of the ventral stream, V2 is often considered as a potential hub for early form processing. Human observers are sensitive to image regions with second-order statistics such as high curvature [22, 24] and texture boundaries [27], and some neurons in V2 appear to be capable of signaling these forms [23, 25, 26]. We measure a spatial index for parallelism and curvature for each cell by determining how well their receptive fields conform to a set of idealized curves (see section 4.2.6). V1 cells tend to be better fit by both parallel features and curved features (Figure 4.20, top), though the difference is not significant. While this may be surprising, remember that V1 cells

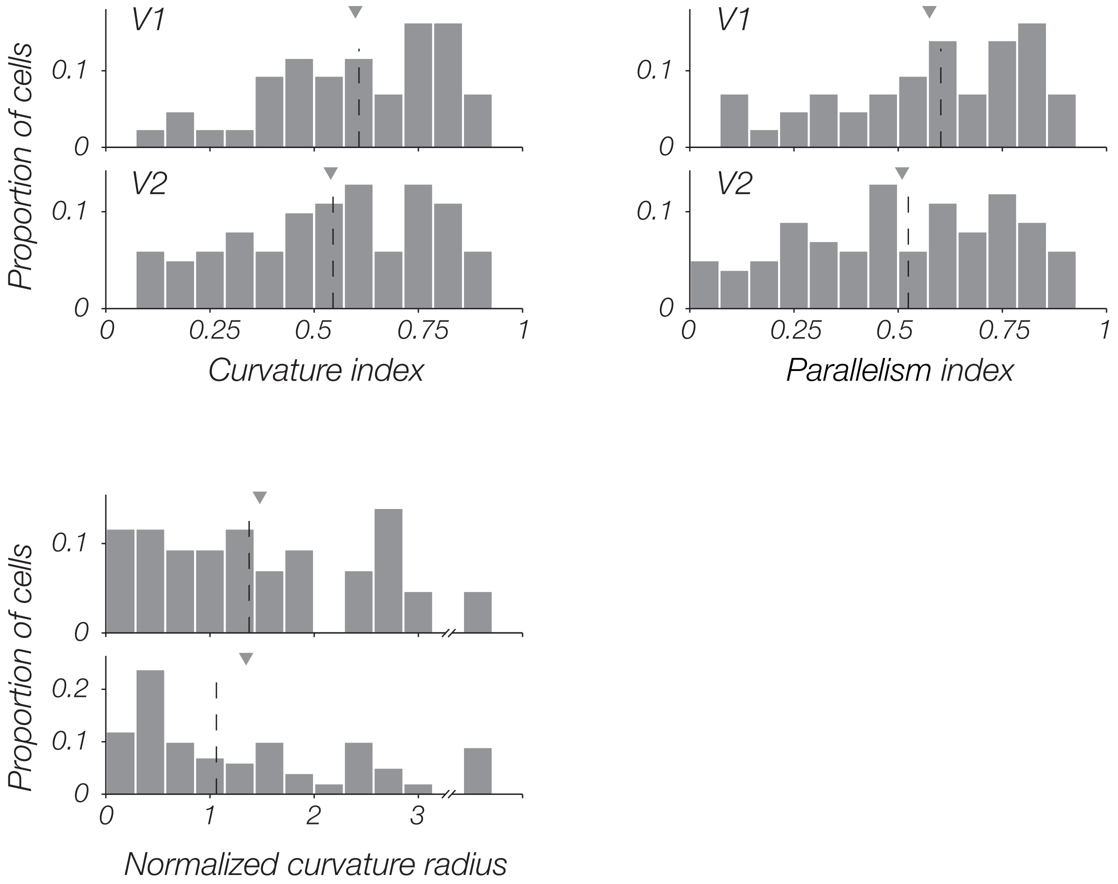


Figure 4.20: Spatial linearity and curvature indices are computed by measuring the best set of curves or lines that conform to the receptive field. V1 cells are better fit by both linear features and curves (top), but V2 cells show a preference for curves of smaller radii (bottom). The 'normalized radius' measure is defined in units of the diameter of the stimulus, and cells after the break have a curvature radius of greater than 3 times the stimulus diameter. None of the differences between V1 and V2 are significant.

tend to be more uniform over space in general, and note that curved features with very large radii can approximate linear features to a first-order comparison. Comparing curvature radius reveals that there is a trend for V2 cells to be selective for features with tighter curvature (Figure 4.20, bottom), though the difference is also not significant.

4.7 Discriminating V1 and V2 receptive fields

There are small differences between receptive fields in V1 and V2, but it is difficult to distinguish the two areas from individual statistics alone. This may be because there are many different ways that a V2 cell could differ from the canonical model of a V1 cells, and each statistic can only capture one aspect of these differences. In order to determine if V1 and V2 cells are genuinely different in quality, we must either consider new sets of discriminating stimuli, or we must look for combinations of statistics that can differentiate the two areas. (For much of this section we use only the cells that were fit well by the model, excluding the 13% of cells with $r < 0.1$ to yield $n_{V1} = 36$ and $n_{V2} = 89$).

A recent study by Freeman et al. show that V2 cells have a clear preference for 'naturalistic' images over spectrum-matched noise images, a selectivity that is not shared by V1 neurons [73]. Naturalistic images are stimuli that are generate from a seed of Gaussian white- noise, but the stimuli are adjusted until they share the same high-order texture statistics as a referential natural texture image [85]. The texture parameters are marginal and joint statistics of Gabor-like wavelet responses. Noise images start with the same natural texture image but only seek to match the marginal statistics of the wavelet response, which is equivalent to matching the texture image power spectrum. The logic of the Freeman study is similar to our own; if V1 filters are Gabor-like, they should not be sensitive to joint statistics of several filters, unlike V2, which may combines a select set of afferents from V1 to generate novel types of selectivity or invariance.

We tested whether our model of V2 cells were able to distinguish between naturalistic and noise images as the physiological evidence suggests that they should. We

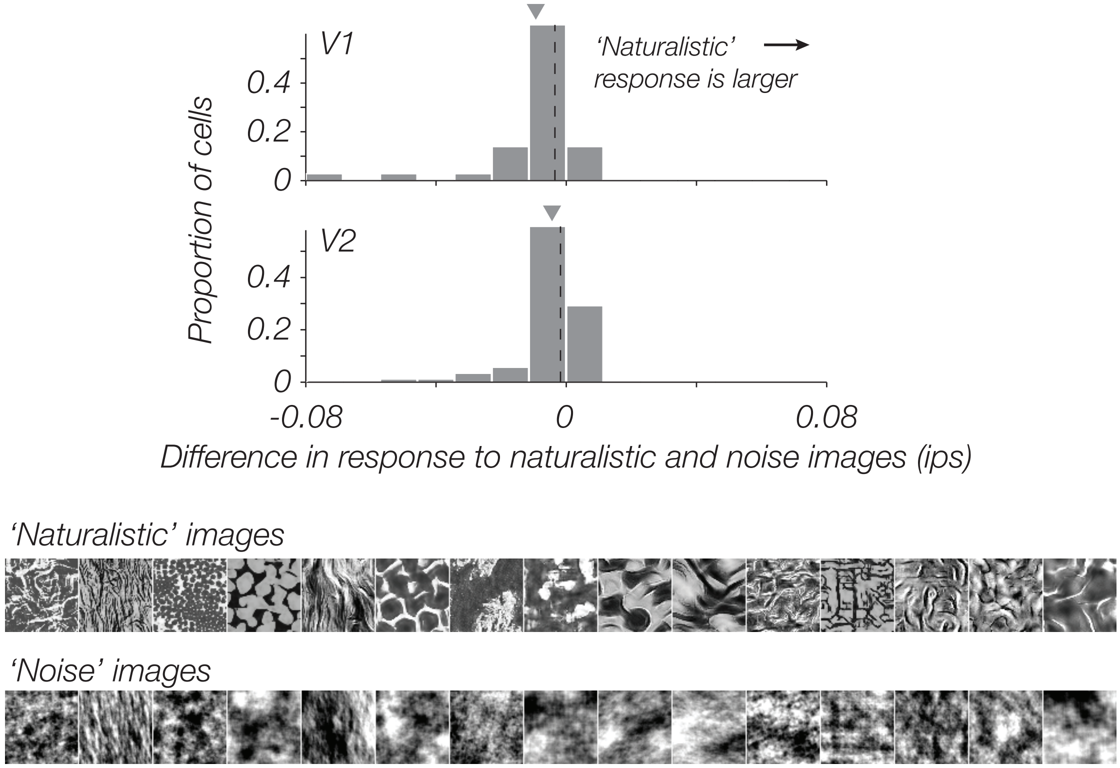


Figure 4.21: [Bottom] We present the model V1 and V2 cells with the same naturalistic and noise images (i.e. texture-parameter matched and spectrum-matched) that were presented to real cells in an experiment by Freeman et al. [73]. Top and bottom image pairs were generated from the same reference texture. [Top] Both V1 and V2 cells showed little difference in response to the two classes of stimuli.

presented the images from Freeman et al. to the model cells in V1 and V2 and measured the difference in the firing rate output (Figure 4.21). In general, both V1 and V2 cells respond with similar firing rates to each class of images. However, though the difference is small, there is a slight trend for V2 cells to show a larger response to naturalistic images than noise images, a trend that is statistically significant if we exclude the cells that were not fit well by our model ($p <$ 0.05; t-test). The lack of a strong difference with our model could be due to a number of factors. First, our model *linearly* combines V1-like afferents. It's possible that more complicated models with nonlinear interactions are required to capture the higher-order texture statistics.

Second, the stimuli that we used in our experiment were designed to elucidate receptive field shape and are narrow-band in spatial frequency. Freeman et al. find that cross-scale interactions often appear to drive V2 cell differentiation, which would be difficult for us to detect.

A more natural way to distinguish V1 and V2 cells within the context of our experiment is to look for a combination of receptive field statistics that discriminate the two areas. We take both an unsupervised and a supervised approach to looking for these combinations in a hand-selected set of statistics. The statistics we chose are the 7 from the histograms plotted in section 4.6, along with the envelope aspect ratio, and a curvature tuning index which measures the difference in tuning to the preferred curve and a curve in the opposite direction.

In the unsupervised method we use Principle Component Analysis (PCA) to reduce the dimensionality of the statistics from 9 dimensions to 2 without knowledge of cell type. We first normalize each statistic to have zero mean and unit variance over all cells and then compute the eigenvectors of the covariance of this matrix. The first two principle components, plotted in Figure 4.22, describe the combinations of statistics that vary the most over all cells, and capture over 70% of the overall variance. Plotting the data projected onto these components for both V1 and V2 shows overlapping but distinct distributions. We also display a discriminant, which is the vector of PCA eigenvalues, $[\lambda_1 \;\; \lambda_2]$. V1 and V2 projected onto this discriminant are significantly different from one another ($p < 0.05$, t-test), suggesting an intrinsic difference between V1 and V2 receptive fields.

To get an idea of *how* different V1 and V2 are, we also build a discriminant that is designed to maximally separate the distributions of V1 and V2 statistics. Linear Discriminant Analysis (LDA), unlike PCA, uses the label of each cell to find the statistics

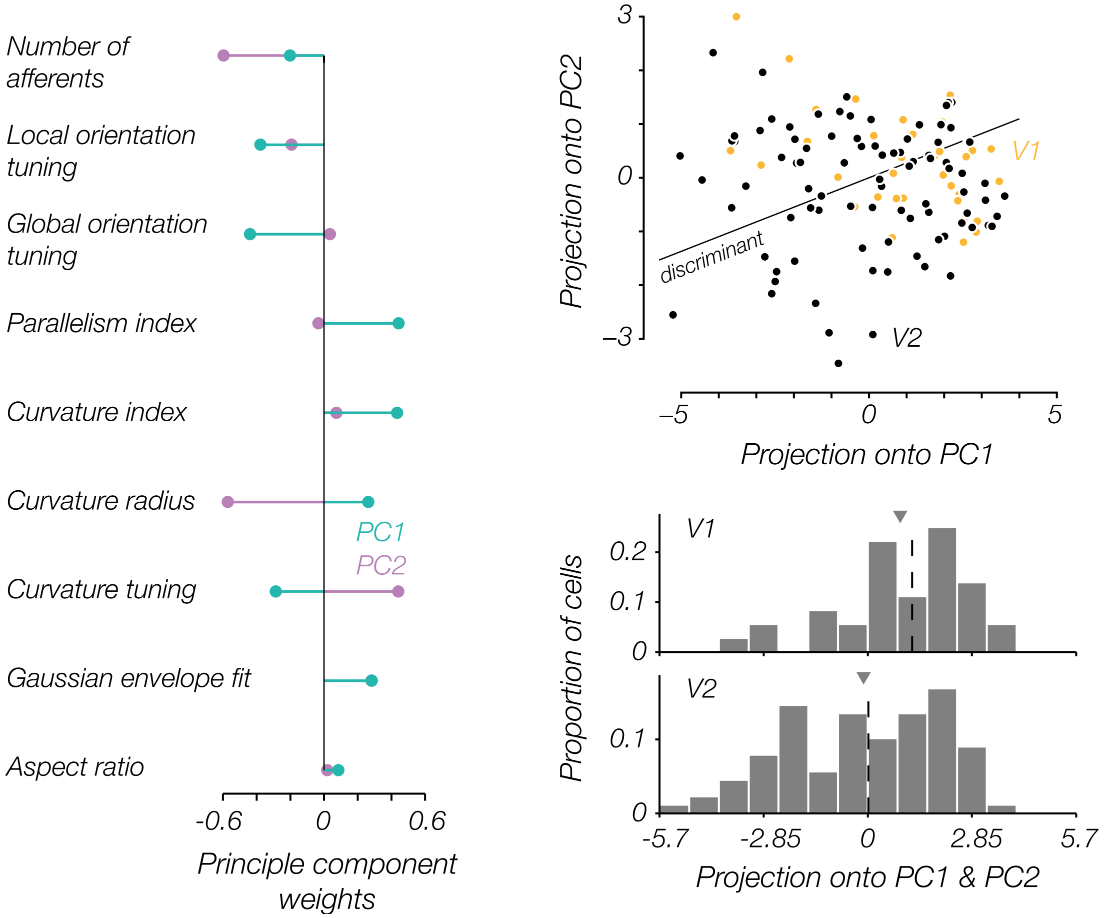


Figure 4.22: The principle components (left) describe the statistics combinations that capture the most variance across cells. When the data is projected onto the two components (top right) there is a difference in the distribution of V1 and V2 neurons. The black line indicates the discriminant, which is simply a vector of the eigenvalues of the PCA decomposition. When the data is again projected onto this vector and split between V1 and V2 (bottom right) there is a clear difference between the two areas.

matrix projection that optimally discriminates V2 from V1 (see section 4.2.6). The linear discriminant, plotted in Figure 4.23, is not as easy to interpret as the principle components because the weights to each statistic are affected by the in-class covariance structure. For example, the curvature tuning parameter is higher for V2 on average but shows up as a negative weight in the linear discriminant. This is because curvature tuning is strongly correlated with some of the other statistics, like global orientation

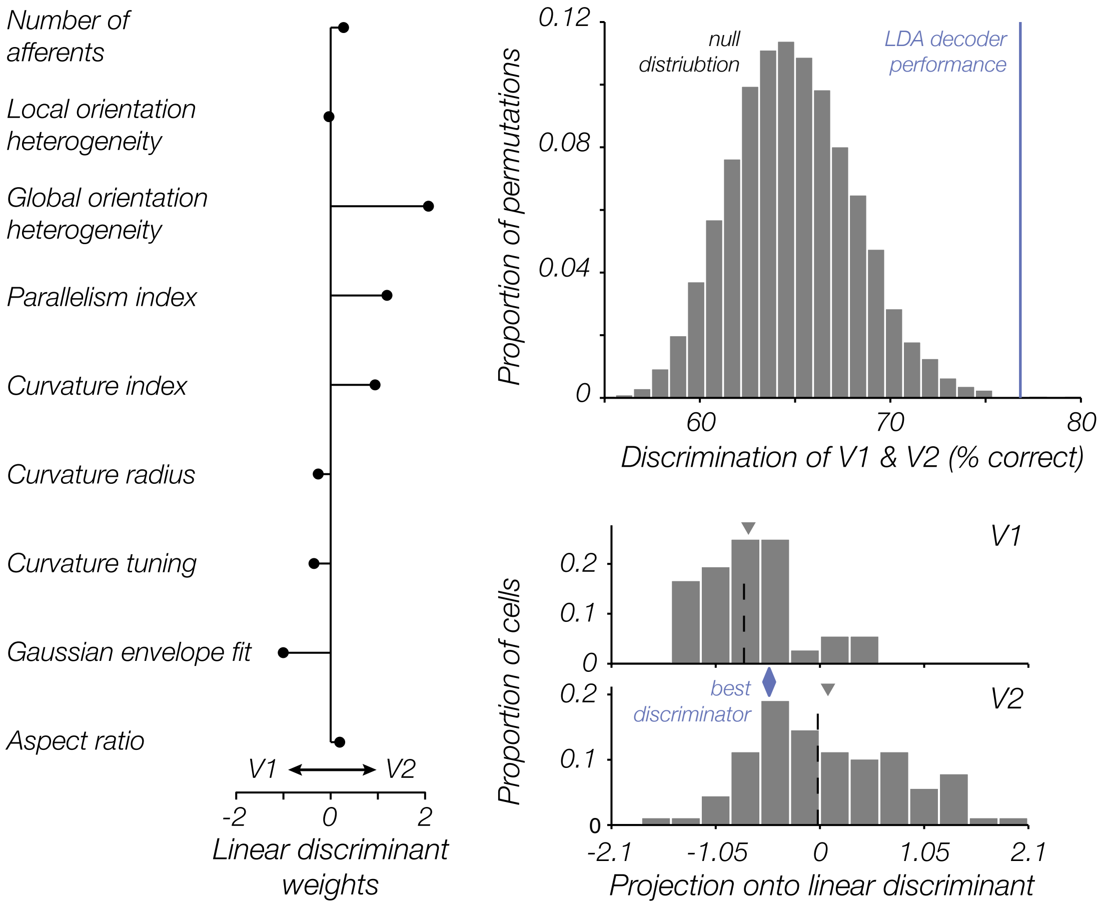


Figure 4.23: The optimal linear discriminant between V1 and V2 assigns weights to each individual receptive field statistic (left). When V1 and V2 are projected onto this discriminant (bottom right) the two distributions are significantly different. An optimal discriminator can split cells from the two areas with 77% accuracy. We permute the cell labels 10,000 times and perform a Linear Discriminant Analysis (LDA) to obtain a null distribution for classification accuracy (top right).

homogeneity (r = 0.51), and negatively correlated with others such as the parallelism index (r = -0.65). However, as an optimal discriminator, we can use the discrimination performance to determine how different V1 is from V2.

Projecting the statistics onto the linear discriminant shows that V1 is significantly different than V2 ($p <<$ 0.05, t-test; Figure 4.23, bottom right). An optimal decision boundary can discriminate the two cell types with 77% accuracy. LDA can be prone to overfitting in high dimensions, and so we test the hypothesis that this discrimination can occur by chance. The null hypothesis of discrimination accuracy is constructed by permuting the data labels 10,000 times and constructing a classifier for each new data set. The distribution of accuracy values is shown in Figure 4.23 (top right). The actual performance for discriminating V1 and V2 falls well outside this null distribution, indicating statistical significance by any reasonable criterion.

4.8 Discussion

We describe a hierarchical model for V2 neurons that combines specific sets of V1 afferents to generate form-selective receptive fields. The model is validated with a playback experiment, and with orientation tuning data, to ensure that it is capturing relevant tuning information. Many cells in V2 are difficult to distinguish from large V1 simple or complex cells, but some V2 cells show spatially heterogenous tuning that resembles a preference for curves, angles, or complex forms. Though it is difficult to discriminate V1 and V2 cells based on solitary receptive fields statistics, the aggregate of many statistics can reliably determine cell type with up to 77% accuracy.

In section 4.5 we show that model performance is sensitive to the alteration of some aspects of it's design. For example, replacing the linear afferent filters with blocky BWT filters, or changing the nonlinearity from half-squaring to half-wave linear, can have a profound affect on model performance and interpretation. The goal of each model from the vantage of its objective function is to accurately reproduce the observed firing rate while using as few afferents as possible. This can be best accomplished if the simulated afferents closely match the properties of the real afferents. Otherwise, it is likely that a less parsimonious solution with many more afferents, and with diverse tuning properties and a bimodal weight distribution, will be required to successively approximate the receptive field. This observation, though subtle, is a key difference between our study and previous modeling experiments in V2 (e.g. [30]).

Building receptive models is one way of learning about the properties of sensory area neurons, but it is not the only method. Tuning experiments, that look for selectivity (or invariance) over a select set of sensory attributes, are a complementary approach. For example, previous V2 experiments have found tuning for stereoscopic edges [78], illusory contours [26], and texture parameters [73]. Tuning experiments can be relatively easy to perform, but they require a strong hypothesis about cellular computation at the outset; an experiment seeking to elucidate color tuning properties would not also vary orientation or disparity, and combinations of properties are rarely presented or explored. This means that a great deal of statistical power can be leveraged onto the property of interest because no other properties are varied. The disadvantage is that, because these types of experiments are mostly agnostic to computational mechanism, it can be difficult to generalize results to the full population of recorded cells. Typical tuning experiments generally find interesting tuning properties in only a subset of cells. What do the rest of the cells do, and how do they do it?

Receptive field model experiments try to work from the opposite direction, building mechanistic descriptions of information processing that are largely blind to the *purpose* of the computation. Though fitting these models can be difficult, progress can occur incrementally. Model complexity can be increased until reaching a desired performance criterion, over generations of models and generations of scientists. However, model failure can be difficult to interpret because it is often not known if it should be attributed to the model structure, or to the model estimation procedure. The lack of a functional hypothesis can also be limiting. For example, in this study, concise summaries can be elusive because there is a variety of ways that V2 neurons can differ from V1 neurons.

Many receptive field models for V2 neurons in our experiment show selectivity for shapes that are very different than the Gabor-like receptive fields of V1 neurons. Curvature, angles, and other contours are are visually striking examples, but other, less structured forms are also common. Without a clear hypothesis of what these neurons are computing, interpreting these receptive fields can be difficult. Karklin & Lewicki analyzed the covariance structure of natural images and predicted that some higher-order visual neurons should be selective to complex image features [86]. Many of the types of units that they predicted possessed texture-like elements, and qualitatively resemble some of our V2 receptive fields. We find that our models are not able to differentiate between naturalistic textures and noise, which seems like a necessary condition in order to claim true selectivity, but perhaps the addition of simple nonlinearities to the model may be able to add this functionality.

Nonlinear interactions between receptive field elements may be an important aspect of V2 functionality. In a computational analogy to V1, the V2 model presented in this chapter characterizes 'simple' V2 cells because afferents are added together *linearly*. In V1, many of the interesting computations are performed in a stage subsequent to the simple cells. Complex cells pool together many such cells which generates the novel property of position invariance. Similarly, there may be 'complex' V2 cells that pool together the outputs of the simpler V2 units, and these cells could be the substrate for robust contour and texture discrimination.

Part I of this thesis describes how an LN-LN subunit operation can yield position invariance in complex cells. Similar models, working from a bank of simulated V1 afferents rather than stimulus pixels, may be useful to describe the properties of some V2 neurons. For example, cells that combine curve-selective subunits with broad spatial pooling profile could implement position-invariant curve detection. Subunit models can also be designed to be invariant to different types of transformations. Though position invariance is the clear choice for V1 cells that operate on pixels, it is possible to imagine that some V2 neurons are 'convolutional' over scale or orientation. For example, such a neuron could be invariant to the precise orientation of a stimulus, but selective for the particular combination of elements that generate acute angles.

General Discussion

This thesis presented two new functional models of single neurons in V1 and V2. The common thread that connects these models is they are hierarchical, and both are designed to match the general architecture of their respective area. The earliest cortical neurons receive inputs that are closely tied to the stimulus and are relatively easy to fit. For example, V1 simple cells receive afferents from a population of LGN neurons, which is largely linear under white-noise stimulus conditions. However, neurons further downstream receive afferents that are more nonlinear, and this can make receptive field fitting difficult. By explicitly incorporating the hierarchical structure of the early visual system, our models are more accurate and efficient than the standard models for these types of neurons. Moreover, the parameters of the models are easy to interpret because they represent the qualities of hypothetical afferents.

The subunit model for V1 receptive fields is a quantitative instantiation of the classic qualitative model. Each neuron in V1 receives a series of subunit inputs. For complex cells, there are many subunits that each represent a simple cell afferent, but for a simple cell there is only a single subunit (in this case, the model devolves into a classic LN computation). Neurons that are neither simple nor complex fit neatly within this paradigm because their receptive fields can be modeled as the sum of an intermediate number of simple-cell subunits. We find that for fixed amounts of data the subunit model is more accurate and efficient than alternative models, such as the Rust-STC (Spike Triggered Covariance) model. In the limit of infinite amounts of data, it is likely that both models would be able to fit the data with similar accuracy, but given the constraints on physiological experimentation, the subunit model better able to make use of data collected during realistic studies.

The subunit model owes its efficiency to its parameterization. We expect that all the afferents to a single V1 cell will have similar tuning properties. This expectation is enforced by computing the field of afferents with a convolution, which is very efficient to calculate. This assumed structure also makes the model easy to interpret. A salient feature of receptive fields in visual cortex is that they are well localized in both space and time. The convolutional structure of the afferent pool ensures this feature, unlike orthogonal subspace methods. Though we do not claim that the subunits from the model are the true subunits of the cell, the model afferents should be a close match to the tuning properties of the actual subunits.

The sparse-afferent V2 model is also accurate and efficient. By combining together a set of plausible V1 afferents we can explain receptive field shape selectivity, and we can also capture both the phase-selective and phase-invariant aspects of the neuronal responses. It is important that model afferents match the properties of the true afferents as closely as possible. As with the V1 subunit model, this design principle allows the model to be accurate and efficient because only a few afferents need to be pooled together. Perhaps more importantly, it also allows the results to be interpretable. Each afferent is well localized and can be visualized by its spatial position and tuning properties (for orientation, spatial frequency, and phase). Thus, we can create a *map* of the receptive preference over space by simply plotting the relevant afferents. If, by counter-example, the model afferents were not localized and came from a set of global basis functions, it would be exceedingly difficult to summarize the estimated receptive field properties and understand how they could generate tuning to shape.

We judge our models on accuracy, efficiency, and interpretability, but these three measures are not independent. Rather, they interact in interesting ways. For a given model, accuracy and efficiency can trade off during estimation; models that are more accurate are less likely to be fit efficiently, and efficient models are less likely to be accurate. However, if we are allowed to choose between models with different structures, it may be possible to find a model class that is both more accurate and more efficient than its rivals. We generally assume that the data we collect comes from a real, physical generating process. For neurons, this deterministic process includes the biophysical elements and synaptic connections that turn stimulus inputs into firing rate outputs. When we model of this process, we are performing a regression analysis by trying to explain the dependent data (firing rates) through its connection to the independent data (stimuli). The regression will be easier and more accurate when the model closely matches the generating process of the true physical system. This property also allows for simple models, which decreases the probability that they will overfit. The subunit and sparse-afferent model presented in this thesis are designed to match the expected neural architecture of each area, and this allows them to be accurate and efficient in their descriptions.

Bibliography

[1] D. Hubel and T. Wiesel. Receptive fields, binocular interaction and functional architecture in the cat's visual cortex. *The Journal of Physiology*, 160(1):106–154, 1962.

[2] J. Robson. Spatial and temporal contrast-sensitivity functions of the visual system. *JOSA*, 56(8):1141–1142, 1966.

[3] F. Campbell, B. Cleland, G. Cooper, and C. Enroth-Cugell. The angular selectivity of visual cortical cells to moving gratings. *The Journal of Physiology*, 198(1):237–250, 1968.

[4] F. W. Campbell, G. F. Cooper, and C. Enroth-Cugell. The spatial selectivity of the visual cells of the cat. *The Journal of Physiology*, 203(1):223, July 1969.

[5] J. A. Movshon, I. D. Thompson, and D. J. Tolhurst. Spatial summation in the receptive fields of simple cells in the cat's striate cortex. *The Journal of Physiology*, 283(1):53–77, 1978.

[6] J. A. Movshon, I. D. Thompson, and D. J. Tolhurst. Receptive field organization of complex cells in the cat's striate cortex. *The Journal of Physiology*, 283(1):79–99, 1978.

[7] R. A. Linsenmeier, L. J. Frishman, H. G. Jakiela, and C. Enroth-Cugell. Receptive field properties of X and Y cells in the cat retina derived from contrast sensitivity measurements. *Vision Research*, 22(9):1173–1183, 1982.

[8] Y. T. So and R. Shapley. Spatial tuning of cells in and around lateral geniculate nucleus of the cat: X and Y relay cells and perigeniculate interneurons. *Journal of neurophysiology*, 45(1):107–120, 1981.

[9] R. Shapley and P. Lennie. Spatial frequency analysis in the visual system. *Annual review of neuroscience*, 8(1):547–581, 1985.

[10] J. G. Daugman. Two-dimensional spectral analysis of cortical receptive field profiles. *Vision Research*, 20(10):847–856, 1980.

[11] J.-M. Alonso, W. M. Usrey, and R. C. Reid. Rules of connectivity between geniculate cells and simple cells in cat primary visual cortex. *The Journal of neuroscience*, 21(11):4002–4015, 2001.

[12] O. Schwartz, J. W. Pillow, N. Rust, and E. Simoncelli. Spike-triggered neural characterization. *Journal of Vision*, 6(4):13–13, Feb. 2006.

[13] E. De Boer and P. Kuyper. Triggered Correlation. *IEEE Transactions on Biomedical Engineering*, BME-15(3):169–179, July 1968.

[14] J. P. Jones and L. A. Palmer. The two-dimensional spatial structure of simple receptive fields in cat striate cortex. *Journal of neurophysiology*, 58(6):1187–1211, 1987.

[15] L. Paninski. Maximum likelihood estimation of cascade point-process neural encoding models. *Network: Computation in Neural Systems*, 15(4):243–262, 2004.

[16] E. H. Adelson and J. R. Bergen. Spatiotemporal energy models for the perception of motion. *Journal of the Optical Society of America A*, 2(2):284, 1985.

[17] R. C. Emerson, J. R. Bergen, and E. H. Adelson. Directionally selective complex cells and the computation of motion energy in cat visual cortex. *Vision Research*, 32(2):203–218, Feb. 1992.

[18] N. C. Rust, O. Schwartz, J. A. Movshon, and E. P. Simoncelli. Spatiotemporal Elements of Macaque V1 Receptive Fields. *Neuron*, 46(6):945–956, June 2005.

[19] T. Sharpee, N. Rust, and W. Bialek. Analyzing neural responses to natural signals: maximally informative dimensions. *Neural computation*, 16(2):223–250, 2004.

[20] J. W. Pillow and E. Simoncelli. Dimensionality reduction in neural models: An information-theoretic generalization of spike-triggered average and covariance analysis. *Journal of Vision*, 6(4):9–9, 2006.

[21] T. Lochmann, T. Blanche, and D. Butts. Construction of direction selectivity in V1: from simple to complex cells. *Computational and Systems Neuroscience (CoSyNe)*, 2011.

[22] F. Attneave. Some informational aspects of visual perception. *Psychological Review*, 61(3), 1954.

[23] A. Anzai, X. Peng, and D. C. Van Essen. Neurons in monkey visual area V2 encode combinations of orientations. *Nature neuroscience*, 10(10):1313–1321, 2007.

[24] I. Biederman. Human image understanding: Recent research and a theory. *Computer vision, graphics, and image processing*, 32(1):29–73, 1985.

[25] J. Hegde and D. C. Van Essen. Strategies of shape representation in macaque visual area V2. *Visual Neuroscience*, 20(03):313–328, 2003.

[26] R. von der Heydt and E. Peterhans. Mechanisms of contour perception in monkey visual cortex. I. Lines of pattern discontinuity. *The Journal of neuroscience*, 9(5):1731–1748, 1989.

[27] M. S. Landy and N. Graham. Visual perception of texture. *The visual neurosciences*, 2:1106–1118, 2004.

[28] Y. El-Shamayleh and J. A. Movshon. Neuronal responses to texture-defined form in macaque visual area V2. *The Journal of neuroscience*, 31(23):8543–8555, 2011.

[29] B. D. Willmore, R. J. Prenger, and J. L. Gallant. Neural representation of natural images in visual area V2. *The Journal of neuroscience*, 30(6):2102–2114, 2010.

[30] B. Willmore, R. J. Prenger, M. C. K. Wu, and J. L. Gallant. The berkeley wavelet transform: a biologically inspired orthogonal wavelet transform. *Neural computation*, 20(6):1537–1564, 2008.

[31] C. Bredfeldt, J. Read, and B. Cumming. A quantitative explanation of responses to disparity-defined edges in macaque V2. *Journal of neurophysiology*, 101(2):701–713, 2009.

[32] C. E. Bredfeldt and B. G. Cumming. A simple account of cyclopean edge responses in macaque V2. *The Journal of neuroscience*, 26(29):7581–7596, 2006.

[33] H. B. Barlow and W. R. Levick. The mechanism of directionally selective units in rabbit's retina. *The Journal of Physiology*, 178(3):477, 1965.

[34] S. Hochstein and R. Shapley. Linear and nonlinear spatial subunits in Y cat retinal ganglion cells. *The Journal of Physiology*, 262(2):265–284, 1976.

[35] J. Demb, K. Zaghloul, L. Haarsma, and P. Sterling. Bipolar cells contribute to nonlinear spatial summation in the brisk-transient (Y) ganglion cell in mammalian retina. *The Journal of neuroscience*, 21(19):7447–7454, 2001.

[36] J. Crook, B. Peterson, O. Packer, F. Robinson, J. Troy, and D. Dacey. Y-cell receptive field and collicular projection of parasol ganglion cells in macaque monkey retina. *The Journal of neuroscience*, 28(44):11277–11291, 2008.

[37] P. Joris, C. Schreiner, and A. Rees. Neural processing of amplitude-modulated sounds. *Physiol. Rev.*, 84:541–577, 2004.

[38] G. C. DeAngelis, I. Ohzawa, and R. Freeman. Spatiotemporal organization of simple-cell receptive fields in the cat's striate cortex. I. General characteristics and postnatal development. *Journal of neurophysiology*, 69(4):1091–1117, 1993.

[39] X. Chen, F. Han, M. m. Poo, and Y. Dan. Excitatory and suppressive receptive field subunits in awake monkey primary visual cortex (V1). *Proceedings of the National Academy of Sciences*, 104(48):19120–19125, 2007.

[40] M. Ito and H. Komatsu. Representation of angles embedded within contour stimuli in area V2 of macaque monkeys. *The Journal of neuroscience*, 24(13):3313–3324, 2004.

[41] K. Fukushima. Neocognitron: A self-organizing neural network model for a mechanism of pattern recognition unaffected by shift in position. *Biological cybernetics*, 36(4):193–202, 1980.

[42] M. Riesenhuber and T. Poggio. Hierarchical models of object recognition in cortex. *Nature neuroscience*, 2:1019–1025, 1999.

[43] E. De Boer. *Reverse correlation I. A heuristic introduction to the technique of triggered correlation with application to the analysis of compound systems*. Proc. Kon. Nederl. Akad. Wet, 1968.

[44] E. Chichilnisky. A simple white noise analysis of neuronal light responses. *Network: Computation in Neural Systems*, 12(2):199–213, 2001.

[45] R. D. R. V. Steveninck and W. Bialek. Real-Time Performance of a Movement-Sensitive Neuron in the Blowfly Visual System: Coding and Information Transfer in Short Spike Sequences. *Proceedings of the Royal Society B: Biological Sciences*, 234(1277):379–414, Sept. 1988.

[46] N. Brenner, W. Bialek, and R. de Ruyter van Steveninck. Adaptive Rescaling Maximizes Information Transmission. *Neuron*, 26(3):695–702, June 2000.

[47] O. Schwartz, E. Chichilnisky, and E. Simoncelli. Characterizing neural gain control using spike-triggered covariance. *Advances in neural information processing systems*, 1:269–276, 2002.

[48] J. Touryan, B. Lau, and Y. Dan. Isolation of relevant visual features from random stimuli for cortical complex cells. *The Journal of neuroscience*, 22(24):10811–10818, 2002.

[49] M. Ahrens, L. Paninski, and M. Sahani. Inferring input nonlinearities in neural encoding models. *Network: Computation in Neural Systems*, 19(1):35–67, 2008.

[50] C. Ekanadham, D. Tranchina, and E. Simoncelli. Recovery of sparse translation-invariant signals with continuous basis pursuit. *IEEE Trans Signal Processing*, 59(10):4735–4744, 2011.

[51] R. Goris, E. P. Simoncelli, and J. A. Movshon. Using a doubly-stochastic model to analyze neuronal activity in the visual cortex. In *Computational and Systems Neuroscience (CoSyNe)*, Salt Lake City, UT, Feb. 2012.

[52] J. Pillow and J. Scott. Fully Bayesian inference for neural models with negative-binomial spiking. *Advances in neural information processing systems*, pages 1907–1915, 2012.

[53] D. J. Heeger. Half-squaring in responses of cat striate cells. *Visual Neuroscience*, 9(05):427, 2009.

[54] F. Mechler and D. L. Ringach. On the classification of simple and complex cells. *Vision Research*, 42(8):1017–1033, 2002.

[55] J. D. Victor and R. M. Shapley. The nonlinear pathway of Y ganglion cells in the cat retina. *The Journal of General Physiology*, 74(6):671–689, Dec. 1979.

[56] B. Vintch, A. Zaharia, J. Movshon, and E. P. Simoncelli. Efficient and direct estimation of a neural subunit model for sensory coding. *Advances in neural information processing systems*, 2012.

[57] M. Park and J. W. Pillow. Receptive field inference with localized priors. *PLoS computational biology*, 7(10):e1002219, 2011.

[58] M. P. Sceniak, M. J. Hawken, and R. Shapley. Visual spatial characterization of macaque V1 neurons. *Journal of neurophysiology*, 85(5):1873–1887, 2001.

[59] G. A. Walker, I. Ohzawa, and R. D. Freeman. Suppression outside the classical cortical receptive field. *Visual Neuroscience*, 17(3):369–379, 2000.

[60] D. I. Perrett and M. W. Oram. Neurophysiology of shape processing. *Image and Vision Computing*, 11(6):317–333, July 1993.

[61] T. Poggio and S. Edelman. A network that learns to recognize 3D objects. *Nature*, 343(6255):263–266, 1990.

[62] Y. LeCun and Y. Bengio. Convolutional networks for images, speech, and time series. *The handbook of brain theory and neural networks*, 3361, 1995.

[63] K. Jarrett, K. Kavukcuoglu, M. A. Ranzato, and Y. LeCun. What is the best multi-stage architecture for object recognition? In *IEEE International Conference on Computer Vision (ICCV)*, pages 2146–2153. IEEE, 2009.

[64] M. D. Zeiler, D. Krishnan, G. W. Taylor, and R. Fergus. Deconvolutional networks. *IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2010*, pages 2528–2535, 2010.

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

[66] J. Freeman and E. P. Simoncelli. Metamers of the ventral stream. *Nature neuroscience*, 14(9):1195–1201, Aug. 2011.

[67] P. Z. Marmarelis, V. Z. Marmarelis, P. Z. Marmarelis, and V. Z. Marmarelis. *Analysis of Physiological Systems*. Springer US, Boston, MA, 1978.

[68] B. Lau. Computational subunits of visual cortical neurons revealed by artificial neural networks. *Proceedings of the National Academy of Sciences*, June 2002.

[69] S. Nishimoto, T. Ishida, and I. Ohzawa. Receptive field properties of neurons in the early visual cortex revealed by local spectral reverse correlation. *The Journal of neuroscience*, 26(12):3269–3280, 2006.

[70] K. S. Sasaki and I. Ohzawa. Internal spatial organization of receptive fields of complex cells in the early visual cortex. *Journal of neurophysiology*, 98(3):1194–1212, 2007.

[71] M. Eickenberg, R. J. Rowekamp, M. Kouh, and T. O. Sharpee. Characterizing Responses of Translation-Invariant Neurons to Natural Stimuli: Maximally Informative Invariant Dimensions. *Neural computation*, 24(9):2384–2421, Sept. 2012.

[72] J. Freeman, G. Field, P. Li, M. Greschner, L. Jepson, N. Rabinowitz, E. Pnevmatikakis, D. Gunning, K. Mathieson, A. Litke, E. J. Chichilnisky, and E. Simoncelli. Spatial structure and organization of nonlinear subunits in primate retina. In *Computational and Systems Neuroscience (CoSyNe)*, Salt Lake City, UT, Feb. 2013.

[73] J. Freeman, C. Ziemba, J. A. Movshon, and E. P. Simoncelli. A functional and perceptual signature of the second visual area in primates. *Nature Neuroscience, accepted for publication*, 2013.

[74] H. Lee, C. Ekanadham, and A. Ng. Sparse deep belief net models for visual area V2. *Advances in neural information processing systems*, 2008.

[75] T. Serre, L. Wolf, S. Bileschi, M. Riesenhuber, and T. Poggio. Robust object recognition with cortex-like mechanisms. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 29(3):411–426, 2007.

[76] R. Baumann, R. Zwan, and E. Peterhans. Figure-Ground Segregation at Contours: a Neural Mechanism in the Visual Cortex of the Alert Monkey. *European Journal of Neuroscience*, 9(6):1290–1303, 1997.

[77] E. Peterhans and R. von der Heydt. Mechanisms of contour perception in monkey visual cortex. II. Contours bridging gaps. *The Journal of neuroscience*, 9(5):1749–1763, 1989.

[78] F. T. Qiu and R. von der Heydt. Figure and Ground in the Visual Cortex: V2 Combines Stereoscopic Cues with Gestalt Rules. *Neuron*, 47(1):155–166, 2005.

[79] E. Simoncelli and W. Freeman. The steerable pyramid: A flexible architecture for multi-scale derivative computation. *Image Processing, 1995. Proceedings., International Conference on*, 3:444–447 vol. 3, 1995.

[80] H. Zou and T. Hastie. Regularization and variable selection via the elastic net. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 67(2):301–320, 2005.

[81] A. E. Hoerl and R. W. Kennard. Ridge Regression: Biased Estimation for Nonorthogonal Problems. *Technometrics*, 12(1):55–67, Feb. 1970.

[82] R. Tibshirani. Regression Shrinkage and Selection via the Lasso. *Journal of the Royal Statistical Society. Series B (Methodological)*, 58(1):267–288, Jan. 1996.

[83] T. J. Hastie, R. J. Tibshirani, and J. J. H. Friedman. *The elements of statistical learning*. Springer, 2009.

[84] D. L. Ringach, M. J. Hawken, and R. Shapley. Dynamics of orientation tuning in macaque primary visual cortex. *Nature*, 387(6630):281–284, 1997.

[85] J. Portilla and E. P. Simoncelli. A parametric texture model based on joint statistics of complex wavelet coefficients. *International Journal of Computer Vision*, 40(1):49–70, 2000.

[86] Y. Karklin and M. S. Lewicki. Emergence of complex cell properties by learning to generalize in natural scenes. *Nature*, 457(7225):83–86, Nov. 2008.