Theory of cortical function

David J. Heeger[a,b,1]

[a]Department of Psychology, New York University, New York, NY 10003; and [b]Center for Neural Science, New York University, New York, NY 10003

This contribution is part of the special series of Inaugural Articles by members of the National Academy of Sciences elected in 2013.

Contributed by David J. Heeger, December 22, 2016 (sent for review August 19, 2016; reviewed by Peter Dayan, Kenneth D. Harris, and Alexandre Pouget)

Most models of sensory processing in the brain have a feedforward architecture in which each stage comprises simple linear filtering operations and nonlinearities. Models of this form have been used to explain a wide range of neurophysiological and psychophysical data, and many recent successes in artificial intelligence (with deep convolutional neural nets) are based on this architecture. However, neocortex is not a feedforward architecture. This paper proposes a first step toward an alternative computational framework in which neural activity in each brain area depends on a combination of feedforward drive (bottom-up from the previous processing stage), feedback drive (top-down context from the next stage), and prior drive (expectation). The relative contributions of feedforward drive, feedback drive, and prior drive are controlled by a handful of state parameters, which I hypothesize correspond to neuromodulators and oscillatory activity. In some states, neural responses are dominated by the feedforward drive and the theory is identical to a conventional feedforward model, thereby preserving all of the desirable features of those models. In other states, the theory is a generative model that constructs a sensory representation from an abstract representation, like memory recall. In still other states, the theory combines prior expectation with sensory input, explores different possible perceptual interpretations of ambiguous sensory inputs, and predicts forward in time. The theory, therefore, offers an empirically testable framework for understanding how the cortex accomplishes inference, exploration, and prediction.

computational neuroscience | neural net | inference | prediction | vision

Perception is an unconscious inference (1). Sensory stimuli are inherently ambiguous so there are multiple (often infinite) possible interpretations of a sensory stimulus (Fig. 1). People usually report a single interpretation, based on priors and expectations that have been learned through development and/or instantiated through evolution. For example, the image in Fig. 1*A* is unrecognizable if you have never seen it before. However, it is readily identifiable once you have been told that it is an image of a Dalmatian sniffing the ground near the base of a tree. Perception has been hypothesized, consequently, to be akin to Bayesian inference, which combines sensory input (the likelihood of a perceptual interpretation given the noisy and uncertain sensory input) with a prior or expectation (2–5).

Our brains explore alternative possible interpretations of a sensory stimulus, in an attempt to find an interpretation that best explains the sensory stimulus. This process of exploration happens unconsciously but can be revealed by multistable sensory stimuli (e.g., Fig. 1*B*), for which one's percept changes over time. Other examples of bistable or multistable perceptual phenomena include binocular rivalry, motion-induced blindness, the Necker cube, and Rubin's face/vase figure (6). Models of perceptual multistability posit that variability of neural activity contributes to the process of exploring different possible interpretations (e.g., refs. 7–9), and empirical results support the idea that perception is a form of probabilistic sampling from a statistical distribution of possible percepts (9, 10). This noise-driven process of exploration is presumably always taking place. We experience a stable percept most of the time because there is a single interpretation that is best (a global minimum) with respect to the sensory input and the prior. However, in some cases, there are two or more interpretations that are roughly equally good (local minima) for bistable or multistable perceptual phenomena (9, 11, 12).

Prediction, along with inference and exploration, may be a third general principle of cortical function. Information processing in the brain is dynamic. Visual perception, for example, occurs in both space and time. Visual signals from the environment enter our eyes as a continuous stream of information, which the brain must process in an ongoing, dynamic way. How we perceive each stimulus depends on preceding stimuli and impacts our processing of subsequent stimuli. Most computational models of vision are, however, static; they deal with stimuli that are isolated in time or at best with instantaneous changes in a stimulus (e.g., motion velocity). Dynamic and predictive processing is needed to control behavior in sync with or in advance of changes in the environment. Without prediction, behavioral responses to environmental events will always be too late because of the lag or latency in sensory and motor processing. Prediction is a key component of theories of motor control and in explanations of how an organism discounts sensory input caused by its own behavior (e.g., refs. 13–15). Prediction has also been hypothesized to be essential in sensory and perceptual processing (16–18). However, there is a paucity of theories for how the brain performs perceptual predictions over time (19–23), noting that many of the so-called "predictive coding theories" of sensory and perceptual processing do not predict forward in time and are not in line with physiological and psychological phenomena (*Discussion*). Moreover, prediction might be critical for yet a fourth general principle of cortical function: learning (*Discussion*).

The neocortex accomplishes these functions (inference, exploration, prediction) using a modular design with modular circuits and modular computations. Anatomical evidence suggests the existence of canonical microcircuits that are replicated across cortical areas (24, 25). It has been hypothesized, consequently, that the brain relies on a set of canonical neural computations, repeating them across brain regions and modalities to apply similar operations of the same form, hierarchically (e.g., refs. 26 and 27). Most models of sensory processing in the brain, and

Significance

A unified theory of cortical function is proposed for guiding both neuroscience and artificial intelligence research. The theory offers an empirically testable framework for understanding how the brain accomplishes three key functions: (*i*) inference: perception is nonconvex optimization that combines sensory input with prior expectation; (*ii*) exploration: inference relies on neural response variability to explore different possible interpretations; (*iii*) prediction: inference includes making predictions over a hierarchy of timescales. These three functions are implemented in a recurrent and recursive neural network, providing a role for feedback connections in cortex, and controlled by state parameters hypothesized to correspond to neuromodulators and oscillatory activity.

Author contributions: D.J.H. designed research, performed research, and wrote the paper.

Reviewers: P.D., University College London; K.D.H; University College London; and A.P., University of Geneva.

The author declares no conflict of interest.

See Profile on page 1745.

[1]Email: david.heeger@nyu.edu.

This article contains supporting information online at www.pnas.org/lookup/suppl/doi:10.1073/pnas.1619788114/-/DCSupplemental.

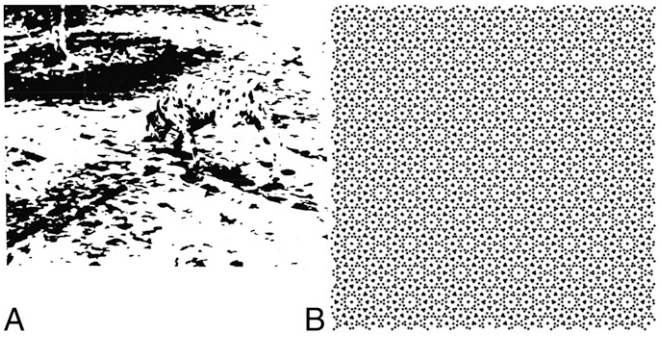

Fig. 1. Perceptual inference. (*A*) Prior expectation. Reprinted with permission from ref. 84. (*B*) Perceptual multistability. Reprinted with permission from ref. 85.

many artificial neural nets (called deep convolutional neural nets), have a feedforward architecture in which each stage comprises a bank of linear filters followed by an output nonlinearity (Fig. 2 *A* and *B*). These hierarchical, feedforward processing models have served us well. Models of this form have been used to explain a wide range of neurophysiological and psychophysical data, and many recent successes in artificial intelligence are based on this architecture. However, neocortex is not a feedforward architecture. There is compelling evidence for a number of distinct, interconnected cortical areas (e.g., 30 or so in visual cortex), but for every feedforward connection there is a corresponding feedback connection, and there is little or no consensus about the function(s) of these feedback connections (28).

Perceptual phenomena also suggest a role for feedback in cortical processing. For example, memory contributes to what we perceive. Take another look at the Dalmatian image (Fig. 1*A*); then close your eyes and try to visualize the image. This form of memory recall (called visual imagery or mental imagery) generates patterns of activity in visual cortex that are similar to sensory stimulation (e.g., ref. 29). One way to conceptualize visual imagery is to think of it as an extreme case of inference that relies entirely on a prior/expectation with no weight given to the sensory input.

This paper represents an attempt toward developing a unified theory of cortical function, an empirically testable computational framework for guiding both neuroscience research and the design of machine-learning algorithms with artificial neural networks. It is a conceptual theory that characterizes computations and algorithms, not the underlying circuit, cellular, molecular, and biophysical mechanisms (*Discussion*). According to the theory, neural activity in each brain area depends on feedforward drive (bottom-up from a previous stage in the processing hierarchy), feedback drive (top-down context from a subsequent processing stage), and prior drive (expectation). The relative contributions of feedforward drive, feedback drive, and prior drive are controlled by a handful of state parameters. The theory makes explicit how information is processed continuously through time to perform inference, exploration, and prediction. Although I focus on sensation and perception (specifically vision), I hypothesize that the same computational framework applies throughout neocortex.

The computational framework presented here, of course, includes components previously proposed in computational/theoretical neuroscience, image processing, computer vision, statistics, and machine learning with artificial neural networks (*SI Appendix*). I was particularly influenced by an underappreciated signal-processing paper by José Marroquin et al. (30).

Results

In a typical feedforward model of visual processing, the underlying selectivity of each neuron is hypothesized to depend on a weighted sum of its inputs, followed by an output nonlinearity (Fig. 2 *A* and *B*). The weights (which can be positive or negative) differ across neurons conferring preferences for different stimulus features. For neurons in primary visual cortex (V1), for example, the choice of weights determines the neuron's selectivity for orientation, spatial frequency, binocular disparity (by including inputs from both eyes), etc. Taken together, neurons that have the same weights, but shifted to different spatial locations, are called a "channel" (also called a "feature map" in the neural net literature). The responses of all of the neurons in a channel are computed as a convolution over space (i.e., weighted sums at each spatial position) with spatial arrays of inputs from channels in the previous stage in the processing hierarchy, followed by the output nonlinearity. The examples in this paper, only for the sake of simplicity, used quadratic output nonlinearities, but a computation called "the normalization model" has been found to be a better model (both theoretically and empirically) of the output nonlinearity (refs. 31 and 32; *SI Appendix*). Neurons in each successive stage of visual processing have been proposed to perform the same computations. According to this idea, each layer 2 neuron computes a weighted sum of the responses of a subpopulation of layer 1 neurons, and then the response of each layer 2 neuron is a nonlinear function of the weighted sum. (I am using the term "layer" to refer to subsequent stages of processing, following the terminology used in the neural network literature, not intended to map onto the layered anatomical structure of the cortex within a brain area.)

Here, I take a different approach from the feedforward processing model, and instead propose a recurrent network (Fig. 2 *C* and *D*). Similar to the feedforward network, there is again a hierarchy of processing stages, each comprising a number of channels. Also similar to the feedforward network, all neurons in a channel perform the same computation, with shifted copies of the same weights, and an output nonlinearity. However, in addition, the network includes a feedback connection for every

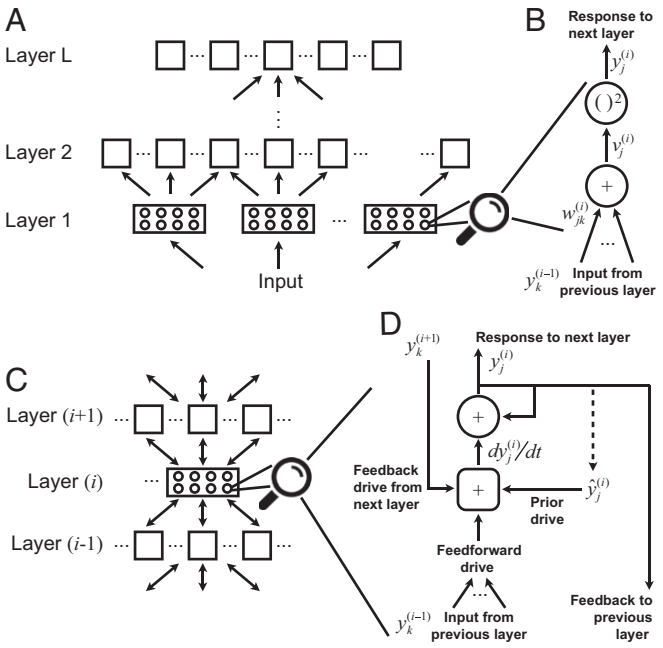

Fig. 2. Neural net architecture and computation. (*A*) Feedforward architecture. Each box represents a channel, comprising a large number of neurons (small circles). All neurons in a channel perform the same computation, with shifted copies of the same weights. (*B*) Neural computation module in the feedforward network. Each neuron computes a weighted sum of its inputs, followed by a squaring output nonlinearity. (*C*) Recurrent architecture. (*D*) Neural computation module in the recurrent network. Feedforward weights (same as *B*) drive the neuron's response to be the same as *A*, but this feedforward drive competes with prior drive and feedback drive. Dashed line, the prior can be computed recursively over time.

feedforward connection (Fig. 2*C*, two-sided arrows, and Fig. 2*D*). Each neuron also has another input that I call a prior, which can be either prespecified or computed recursively (Fig. 2*D*). The response of each neuron is updated over time by summing contributions from the three inputs: feedforward drive, feedback drive, and prior drive (Fig. 2*D*). Each neuron also provides two outputs: feedforward drive to the next layer, and feedback drive to the previous layer. Each neuron performs this computation locally, based on its inputs at each instant in time. However, the responses of the full population of neurons (across all channels and all layers) converge to minimize a global optimization criterion, which I call an energy function. First, I define the energy function. Then, I derive (simply by taking derivatives with the chain rule) how each neuron's responses are updated over time.

The starting point is the hypothesis that neural responses minimize an energy function that represents a compromise between the feedforward drive and prior drive (see Table 1 for a summary of notation):

$$E = \sum_{i=1}^{L} \alpha^{(i)} \left[\lambda^{(i)} \sum_j \left(f_j^{(i)} \right)^2 + \left(1 - \lambda^{(i)} \right) \sum_j \left(p_j^{(i)} \right)^2 \right], \qquad [1]$$

$$f_j^{(i)} = y_j^{(i)} - z_j^{(i)} \text{ (feedforward drive)},$$

$$p_j^{(i)} = y_j^{(i)} - \hat{y}_j^{(i)} \text{ (prior drive)},$$

$$z_j^{(i)} = \left(v_j^{(i)} \right)^2 \text{ (quadratic output nonlinearity)},$$

$$v_j^{(i)} = \sum_k w_{jk}^{(i-1)} y_k^{(i-1)} \text{ (weighted sum)}.$$

The variables (y, v, z, and $\hat{y}$) are each functions of time; I have omitted time in this equation to simplify the notation, but I deal with time and dynamics below.

The values of y are the responses (proportional to firing rates) of the neurons in each layer of the network, where $y^{(0)}$ (layer 0) comprises the inputs to the multilayer hierarchy. The superscript (i) specifies the layer in the hierarchy. For example, layer 1 might correspond to neurons in the lateral geniculate nucleus (LGN) of the thalamus, which receives inputs from the retina (noting that there is no feedback to the retina), and layer 2 might correspond to neurons in V1 that receive direct inputs from the LGN, etc.

The first term of E drives the neural responses to explain the input from the previous layer; f is called the feedforward drive (Eq. **1**, second line). With only this term, the neural responses would be the same as those in a purely feedforward model. The values of v are weighted sums of the responses from the previous layer, and w are the weights in those weighted sums (Eq. **1**, fifth line). The weights are presumed to be the same for all neurons in a channel, but shifted to different spatial locations (i.e., the values of v can be computed with convolution over space). The values of z determine the feedforward drive, after the quadratic output nonlinearity (Eq. **1**, fourth line).

The second term of E drives the neural responses to match a prior; p is called the prior drive (Eq. **1**, third line). With only this term, the neural responses would be driven to be the same as the values of $\hat{y}$. The values of $\hat{y}$ might, for example, be drawn from memory and propagated via the feedback drive to a sensory representation (as detailed below), and/or used to predict forward in time (also detailed below). I show that the values of $\hat{y}$ can be interpreted as an implicit representation of a prior probability distribution (see *Bayesian Inference: Cue Combination* and *SI Appendix*), so I use the term "prior" when referring to $\hat{y}$. For some of the examples, however, it is more appropriate to think of $\hat{y}$ as target values for the responses. For other examples, the values of $\hat{y}$ can be interpreted as predictions for the responses. (I see it as a feature that the various components of the theory can be interpreted in different ways to connect with different aspects of the literature.)

The α and λ ($0 < \lambda < 1$) are state parameters, which I hypothesize change over time under control of other brain systems (*Discussion* and *SI Appendix*). The values of α determine the relative contribution of each layer to the overall energy, and the values of λ determine the trade-off between the two terms in the energy function at each layer. Changing the values of α and λ, as demonstrated below, changes the state of the neural network. With only the first term (i.e., $\lambda = 1$), for example, the neural responses are determined by the feedforward drive, and the network behaves exactly like a conventional feedforward network. With only the second term (i.e., $\lambda = 0$), the neural responses follow the prior and completely ignore the sensory inputs.

For simplicity, Eq. **1** denotes a network with only one channel in each layer, but it can easily be extended to have multiple channels per layer (*SI Appendix*). It is a global optimization criterion; the summation is over all neurons in all channels and all layers, and a summation over time can also be included (see *Prediction* and *SI Appendix*).

The neural responses are modeled as dynamical processes that minimize the energy E over time. Taking derivatives of Eq. **1** (using the chain rule):

$$\tau \frac{dy_j^{(i)}}{dt} = -\frac{dE}{dy_j^{(i)}} = -2\alpha^{(i)} \lambda^{(i)} f_j^{(i)} + 4\alpha^{(i+1)} \lambda^{(i+1)} b_j^{(i)} - 2\alpha^{(i)} \left(1 - \lambda^{(i)} \right) p_j^{(i)}, \qquad [2]$$

$$b_j^{(i)} = \sum_k \left[y_k^{(i+1)} - z_k^{(i+1)} \right] v_k^{(i+1)} w_{kj}^{(i)} \text{ (feedback drive)}.$$

According to this equation, neural responses are updated over time because of a combination of feedforward drive f, feedback drive b, and prior drive p. The first term f is the same feedforward drive as above, and the third term p is the same prior drive as above. The middle term b, the feedback drive, is new. The feedback drive drops out when taking the derivative of Eq. **1** because the response of each neuron appears twice in that equation: (*i*) the derivative of $[y_j^{(i)} - z_j^{(i)}]^2$ gives the feedforward drive; (*ii*) the derivative of $[y_k^{(i+1)} - z_k^{(i+1)}]^2$ gives the feedback drive because $z_k^{(i+1)}$ depends on $y_j^{(i)}$ (*SI Appendix*). The prior drive contributes to minimizing the second term of E in Eq. **1**. The feedforward drive and the feedback drive both contribute to minimizing the first term of E in Eq. **1**. The combined effect of the feedforward drive and the feedback drive is that if the response of a neuron is larger than the value provided by the feedforward processing of its inputs, then its response gets tamped down and its inputs get cranked up; or vice versa if

Table 1. Notation for Eqs. 1 and 2

Symbol	Description
$y_j^{(i)}(t)$	Responses over time of the jth neuron in layer (i)
$y_j^{(0)}(t)$	Inputs (layer 0)
$\hat{y}_j^{(i)}(t)$	Prior expectation (target values) for the responses of the jth neuron in layer (i)
$w_{jk}^{(i-1)}$	Weights from neuron k in layer $(i) - 1$ to neuron j in layer (i)
$v_j^{(i)}(t)$	Weighted sum of the responses from the previous layer
$z_j^{(i)}(t)$	Weighted sum followed by quadratic output nonlinearity
$f_j^{(i)}(t)$	Feedforward drive for the jth neuron in layer (i)
$p_j^{(i)}(t)$	Prior drive for the jth neuron in layer(i)
$b_j^{(i)}(t)$	Feedback drive for the jth neuron in layer (i). See Eq. **2**
$\alpha(t), \lambda(t)$	State parameters

the response of a neuron is smaller than the feedforward value. Specifically, the feedback to layer (i) depends on the mismatch between the responses in the next layer ($i + 1$) and the feedforward drive from layer (i) to layer ($i + 1$); this mismatch is then transformed back to layer (i) through the transpose of the weight matrix (*SI Appendix*). The value of τ is a time constant that I interpret as a combination of the time constant of a neuron's cell membrane and the time constant of synaptic integration.

Inference. Depending on the state parameters (the values of α and λ at each layer), the responses are dominated by sensory input, prior expectation, or a combination of the two.

As a simple example, a three-layer network was implemented that computed a cascade of exclusive-or (XOR) operations (Fig. 3*A*). The response of the layer 3 neuron was 1 if the inputs at layer 0 consisted of a single 1 with three 0s or a single 0 with three 1s. The feedforward drive of each neuron was equal to the square of the difference between its inputs: $(0-0)^2 = 0$, $(0-1)^2 = 1$, $(1-0)^2 = 1$, $(1-1)^2 = 0$. The weights (−1, 1) were the same for each neuron.

The network behaved like a feedforward model for some values of the state parameters (Fig. 3*B*). Responses of the four neurons in layer 1 rapidly converged to values matching the input (Fig. 3*B*, bottom panel). Responses of the two neurons in layer 2 and the neuron in layer 3 each converged more slowly to values determined by the feedforward drive (Fig. 3*B*, top two panels). Because of the choice of state parameters, the energy was dominated by the feedforward drive (Eq. **1**, first term) in layer 1, whereas the prior (Eq. **1**, second term) was ignored in all three layers.

The network behaved like a simple example of memory recall or visual imagery for other values of the state parameters (Fig. 3*C*). The state parameters were set to values so that the energy function (Eq. **1**) was dominated by the layer 3 prior. Consequently, the response of the layer 3 neuron converged to a value determined by its prior (Fig. 3*C*, top panel). The responses of the neurons in layers 2 and 1 converged more slowly to values that were consistent with the layer 3 prior (Fig. 3*C*, bottom two panels). Hence, the value of the prior for the layer 3 neuron propagated back to generate or reconstruct a representation in layer 1. This reconstructed representation in layer 1 corresponded to a sensory input that would have evoked, in a feedforward network, the same layer 3 response. The reconstruction emerged over time; the rise in the neural responses were delayed by ∼100 ms in layer 2 relative to layer 3, and in layer 1 relative to layer 2, even though the time constant was short (τ = 5 ms). Rerunning the simulation yielded different results, depending on the initial conditions (i.e., different initial values for the responses for each of the neurons). However, in all cases, the responses of the layer 1 neurons converged to values that were consistent with the layer 3 prior (i.e., a single 1 with three 0s or a single 0 with three 1s). The layer 3 prior was ambiguous; it did not reconstruct a specific memory but rather a class of memories because there were multiple local minima in the energy function: any input consisting of a single 1 with three 0s or a single 0 with three 1s was consistent with setting the layer 3 prior to 1.

When presented with an ambiguous sensory input, the network was biased by a prior, analogous to the Dalmatian image (Fig. 1*A*). For example, when the input was specified to be (0.5, 0, 0, 0), and the prior for the layer 3 neuron was set to 1, then the global minimum energy state corresponded to responses of the layer 1 neurons of approximately (1, 0, 0, 0). Alternatively, when the prior for the layer 3 neuron was set to 0, then the responses of the layer 1 neurons converged to (0, 0, 0, 0). The sensory input was the same and the state parameters were the same, but the network converged to a different solution, depending on the layer 3 prior.

Exploration. The network explored different possible perceptual interpretations (exhibiting bistability, analogous to Fig. 1*B*) when the input and prior were inconsistent with one another (Fig. 4). The input was specified to be (1, 0, 1, 0) and the prior for the layer 3 neuron was set to 1, such that the inputs were incompatible with the layer 3 prior. Bistability emerged by adding noise to the neural responses. The layer 1 responses remained close to the inputs (Fig. 4, bottom panel) and the response of the layer 3 neuron remained close to its prior (Fig. 4, top panel). However, the responses of the layer 2 neurons changed over time, alternating between (1, 0) and (0, 1), which corresponded to two local minima in the energy function. The noise was statistically independent across neurons and over time, but nonstationary. In particular, the time course of the SD had a 1/*f* amplitude spectrum (*Discussion* and *SI Appendix*), but similar results were obtained when the SD modulated periodically (e.g., at 10 Hz) instead of having a 1/*f* amplitude spectrum, or when the noise SD was a sum of periodic and 1/*f* components.

Bayesian Inference: Cue Combination. These principles, inference based on prior expectation (often formalized as Bayesian inference) and exploration of alternative possible interpretations (hypothesized to be driven by neural response variability), apply not only to perception but also to motor control, motor learning, and cognition (e.g., refs. 33–38). Consequently, there is considerable interest in how neural populations can represent uncertainty and priors, and perform probabilistic inference and probabilistic learning (2–4, 10, 39–44).

The two terms of Eq. **1** are analogous to Bayesian inference, with the first term representing a negative log likelihood and the

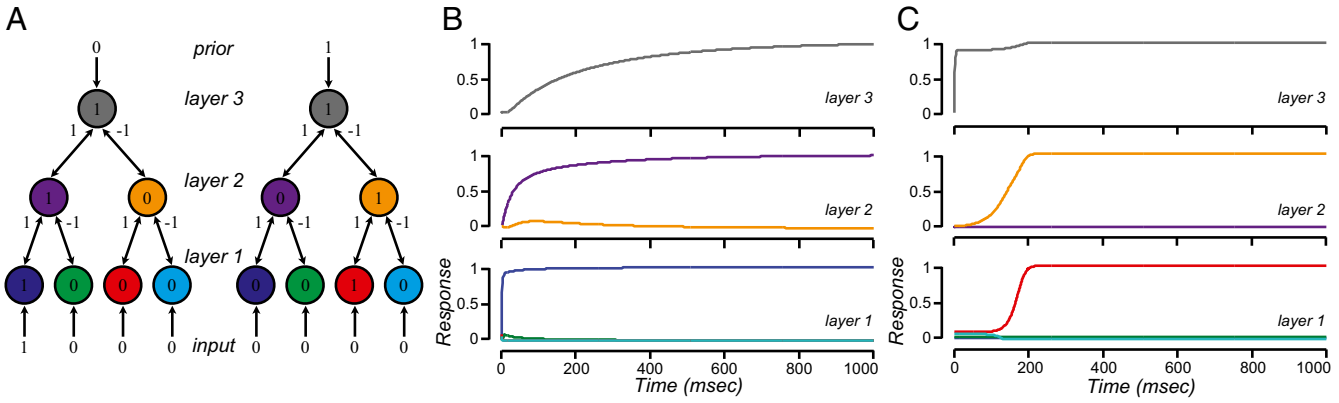

Fig. 3. Inference. (*A*) Network architecture. Feedforward drive of each neuron is the square of the difference between its two inputs. The two examples correspond to responses in *B* and *C*. (*B*) Driven by sensory input. Each panel corresponds to a layer. Each curve corresponds to the response time course of a neuron. Colors correspond to *A*, *Left*. Input: $y^{(0)}$ = (1, 0, 0, 0). Prior: $\hat{y} = 0$ for all neurons in the network. State: λ = (1, 1, 1) and α = (1,0.1,0.1), for layers 1, 2, and 3, respectively. (*C*) Driven by memory. Colors correspond to *A*, *Right*. Input: $y^{(0)}$ = (0,0,0,0). Prior: $\hat{y} = 1$ for the layer 3 neuron and $\hat{y} = 0$ for all other neurons in the network. State: λ = (1, 1, 0.1) and α = (0.001, 0.1, 1). Time constant: τ = 5 ms. See *SI Appendix* for details.

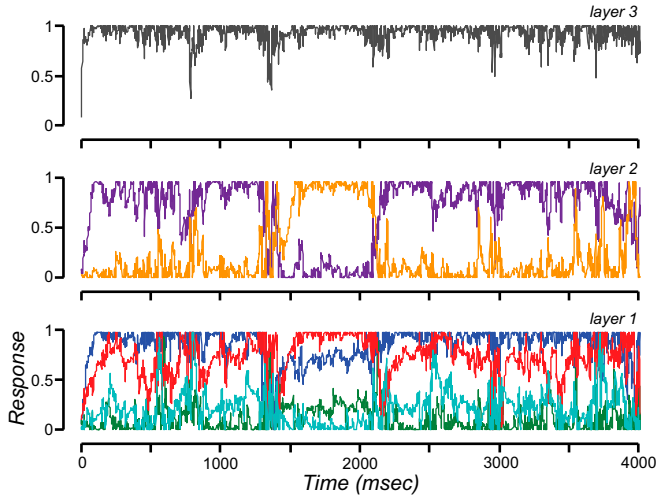

Fig. 4. Exploration. Responses in layer 2 exhibit bistability (same format as Fig. 3 *B* and *C*). Input: $y^{(0)} = (1, 0, 1, 0)$. Prior: $\hat{y} = 1$ for the layer 3 neuron and $\hat{y} = 0$ for all other neurons in the network. State: $\lambda = (1, 1, 0.1)$ and $\alpha = (0.1, 0.1, 1)$, for layers 1, 2, and 3, respectively. Time constant: $\tau = 10$ ms. See *SI Appendix* for details.

second term representing a prior probability. Following previous work on probabilistic population codes (4, 21, 40), the idea is that the neural responses encode an implicit representation of the posterior. Indeed, the values of $\hat{y}$ can be interpreted as an implicit representation of a prior probability distribution, and the values of y can be interpreted as an implicit representation of the posterior (*SI Appendix*). The quadratic function in the first term of Eq. **1** corresponds to a normal distribution for the noise in the feedforward drive and the quadratic in the second term determines the prior probability distribution. Different cost functions (other than quadratics) would correspond to different statistical models of the noise and prior (e.g., refs. 45 and 46). I cannot claim that the theory is, in general, Bayesian, but there are special cases that approximate Bayesian inference.

To make explicit the link to Bayesian inference, I use cue combination as an example. In a cue combination task, an observer is presented with two or more sources of information (cues) about a perceptual variable. For example, early empirical work on cue combination used two depth cues (e.g., stereo and motion parallax) (47). One of the cues is typically more reliable than the other, and the reliability of both cues may vary from one trial to the next of the experiment (e.g., by varying the contrast or visibility of one or both cues). Both cues may be consistent with the same interpretation or they may be in conflict, such that each cue supports a slightly different interpretation. Observers in such an experiment are instructed to indicate their percept (e.g., depth estimate). A series of studies have reported that percepts depend on a combination of the two cues, the reliability of both cues, and a prior. Cue combination tasks have, consequently, been formalized as Bayesian estimation (47), and some empirical results suggest that cue combination is approximately Bayes-optimal (e.g., refs. 5, 48, and 49). The broader literature on psychophysics and perceptual decision-making can be encompassed by this same formalism, with only one cue instead of two.

I implemented a network that combines information from two sensory cues with a prior to simulate a cue combination experiment; the network was designed to approximate optimal Bayesian cue combination. The network consisted of a layer of output neurons and two sets of input neurons (Fig. 5*A*). Each of the input neurons was tuned for depth, responding most strongly to a preferred depth value (Fig. 5*B*). Both sets of input neurons had the same tuning curves but responded to each of two different cues (e.g., stereo and motion parallax). The stimulus strength of each of the two cues scaled the gain of the input neuron's responses, and the input neuron's responses were presumed to be noisy (additive, independent, normally distributed noise). The feedforward drive for each output neuron was a weighted sum of the two input neurons with the corresponding tuning curve (*SI Appendix*), so the output neurons had the same tuning curves as the input neurons. A target value $\hat{y}$ was specified for the response of each output neuron. These target values could be learned, for example, as the mean response of each output neuron, averaged across a series of practice/training trials. These target values for the responses corresponded to a prior probability distribution over the stimulus depth values; each neuron responded selectively to a preferred stimulus depth so a large target value for a particular neuron meant that the corresponding stimulus depth was more likely. Consequently, the vector of $\hat{y}$ values can be transformed to a function that is proportional to a prior probability distribution (Fig. 5*C*; *SI Appendix*). I also defined a readout rule (*SI Appendix*) that transformed the vector of responses of the output neurons to a depth estimate (Fig. 5*D*, approximately equal to the mean of the posterior) and an uncertainty (Fig. 5*E*, approximately equal to the SD of the posterior).

Depth estimates and uncertainties computed with this readout from the network were strongly correlated with optimal Bayesian estimates and uncertainties (estimates: $r = 0.94$; uncertainties: $r = 0.98$). The network was a particularly good approximation to Bayesian inference in two regimes of stimulus strengths (see *SI Appendix* for derivation). (*i*) When the stimulus strength of one or both cues was large, depth estimates and uncertainties depended on the relative reliabilities of the two cues (Fig. 5*D*, *Top*, *Right*, and *Top Right*; Fig. 5*E*, *Bottom*, *Right*, and *Bottom Right*). (*ii*) When the stimulus strengths of both cues were small, depth estimates and uncertainties were dominated by the prior (Fig. 5*D*, *Bottom Left*; Fig. 5*E*, *Top Left*). This network illustrates how the general framework I have laid out can be used to solve fairly complex probabilistic inference near optimally, but it remains to be seen whether this particular model of multisensory integration can account for the experimental data to the same extent as other theories such as the linear probabilistic population code (4, 49).

I am not suggesting that the prior probability distribution (plotted in Fig. 5*C*) and readout (plotted in Fig. 5 *D* and *E*) are explicitly computed and represented in the brain. Rather, the vector of target values $\hat{y}$ (that implicitly encodes a prior) in one channel/layer interacts with the inputs to evoke a vector of neural responses y (that implicitly encodes a posterior). Neural responses in one channel/layer interact (through feedforward and feedback drive) with neural responses in other channels/layers (that implicitly encode their corresponding posteriors), to yield neural responses in all channels and layers that correspond to a globally optimal inference.

Prediction. Prediction requires a model. The idea here is to rely on the generative model embedded in the hierarchical neural network, coupled with the intuition that the relevant timescales are different at each level of the hierarchy (50). The sensory inputs at the bottom of the hierarchy change rapidly but the more abstract representations at successively higher levels change more slowly over time. A simple example is a network in which the responses in one layer are sinusoidal and the feedforward drive to the next layer computes the sum of squares of a pair of neurons that respond with temporal phases offset by 90° (e.g., sine- and cosine-phase). The sinusoids modulate rapidly over time, but the sum of squares is constant over time. The responses of each neuron in the network are computed and predicted recursively over time, for example, with recurrent excitation and inhibition within each module of each channel (Fig. 2*D*, dashed line), and the values of $\hat{y}$ can be interpreted as predictions for the responses. Slow changes at

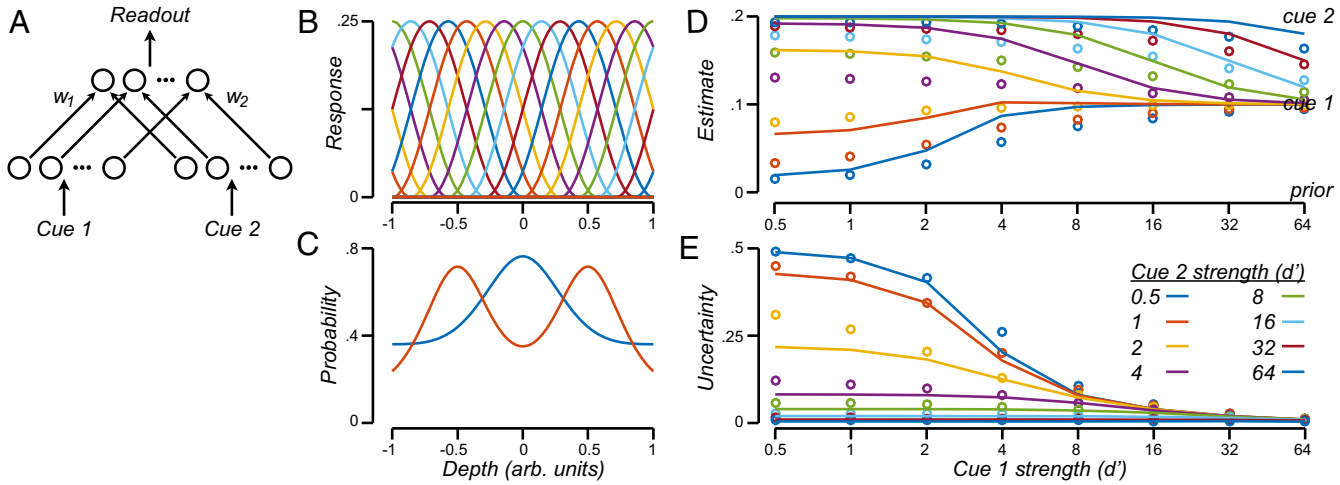

Fig. 5. Bayesian estimation. (*A*) Network architecture. Top row, each circle corresponds to one of 23 output neurons, each tuned for depth (*B*). Bottom row, each circle corresponds to an input neuron. The two sets of input neurons respond to two different cues (e.g., stereo and motion parallax). (*B*) Tuning curves. Each input neuron responds preferentially to a range of depth values with a raised-cosine tuning curve. The tuning curve for the *j*th neuron is denoted $\psi_j(s)$, where *s* is the stimulus depth. (*C*) Example prior probability distributions. Blue, prior corresponding to $\hat{y}_j^{(1)} = \psi_j(0)$ with uncertainty $\sigma_0 = 0.5$. Orange, prior corresponding to $\hat{y}_j^{(1)} = \psi_j(-0.5) + \psi_j(0.5)$ with uncertainty $\sigma_0 = 0.25$. (*D*) Depth estimates. Solid curves, optimal Bayesian estimation. Circles, cue combination network. In the absence of sensory input, the most likely depth was 0 ("prior"). The two sensory cues indicated different depth values ("cue 1" and "cue 2"). Stimulus strengths are specified in units of detectability (d'), where $d' = 1$ corresponds to a stimulus that is barely detectable. (*E*) Uncertainty. Solid curves, optimal Bayesian. Circles, cue combination network. Simulation parameters: cue 1 indicated $s_1 = 0.1$; cue indicated $s_2 = 0.2$; reliability of cue 2 twice that of cue 1 (i.e., $\sigma_1 = 2$, $\sigma_2 = 1$); prior corresponded to blue curve in *C*. See *SI Appendix* for details.

higher layers constrain, via the feedback drive, predictions at lower layers.

The energy function for a one-layer prediction network is expressed as follows (see Table 2 for a summary of notation):

$$E = \sum_t \lambda(t) \left[\left(\sum_m y_{m_1}^{(1)}(t) \right) - y^{(0)}(t) \right]^2 + \sum_t (1 - \lambda(t)) \left[\sum_m \left(y_{m_1}^{(1)}(t) - \hat{y}_{m_1}^{(1)}(t) \right)^2 + \left(y_{m_2}^{(1)}(t) - \hat{y}_{m_2}^{(1)}(t) \right)^2 \right], \quad [3]$$

$$\hat{y}_{m_1}^{(1)}(t) = y_{m_1}^{(1)}(t - \Delta t)\, w_{m_1}^{(1)} - y_{m_2}^{(1)}(t - \Delta t)\, w_{m_2}^{(1)}$$
$$\hat{y}_{m_2}^{(1)}(t) = y_{m_1}^{(1)}(t - \Delta t)\, w_{m_2}^{(1)} + y_{m_2}^{(1)}(t - \Delta t)\, w_{m_1}^{(1)} \quad \text{(predicted responses)},$$

$$w_{m_1}^{(1)} = \cos\left(2\pi\omega_m^{(1)} \Delta t\right)$$
$$w_{m_2}^{(1)} = \sin\left(2\pi\omega_m^{(1)} \Delta t\right) \quad \text{(temporal weights)}.$$

The form of this optimization criterion is borrowed from signal processing (30). The values of y_{m1} and y_{m2} are the responses of a population of neurons that share the same input $y^{(0)}$. The neurons are arranged in pairs (subscripts 1 and 2 with the same value for subscript m). As above, the neural responses are computed dynamically to minimize this energy function over time (*SI Appendix*). The values of $\hat{y}_{m1}$ and $\hat{y}_{m2}$ are the corresponding predictions of the responses from the previous time step (Δt is a discrete time step). I set the priors by hand in the examples above (Figs. 3 and 4), but here they are instead computed recursively. Specifically, they are computed (Eq. **3**, second and third lines) as weighted sums of the responses from the previous time step with temporal weights w_{m1} and w_{m2} (a pair of numbers for each m). The temporal weights confer a 90° phase shift (sine and cosine; Eq. **3**, fourth and fifth lines) between the responses of the two neurons in the pair. Different pairs of neurons (indexed by subscript m) have different dynamics (different temporal frequencies), controlled by the value of ω_m.

A one-layer network was constructed to follow an input for past time, but to predict for future time (Fig. 6, see *SI Appendix* for details). The input was a periodic time series, a sum of sinusoids, until $t = 0$ and then nonexistent for $t > 0$ (Fig. 6*A*, top panel). The network was constructed with five pairs of neurons, each pair corresponding to a different temporal frequency (Fig. 6*B*, blue and green curves). The output of the network (Fig. 6*A*, bottom panel) was computed by summing the responses of these 10 neurons across the five temporal frequencies (i.e., the blue curve in the bottom panel of Fig. 6*A* is the sum of the blue curves in Fig. 6*B*, and likewise for the green curves). The output (Fig. 6*A*, bottom panel, blue curve) followed the input (Fig. 6*A*, top panel) for past time because the state parameter λ was set to a relatively large value ($\lambda = 0.1$ for $t \leq 0$). The network predicted forward in time (Fig. 6*A*, bottom panel), based on the current and past responses, because λ was set to a relatively small value ($\lambda = 0.01$ for $t > 0$).

For a fixed value of λ, each pair of neurons acts like a shift-invariant linear system (i.e., a recursive linear filter). The predicted responses can be computed recursively, but they can also be expressed as a sum of basis functions that I call the "predictive basis functions." The predictive basis functions (damped oscillators of various temporal frequencies) are the impulse response functions of these shift-invariant linear systems, each corresponding to a pair of neurons (indexed by m). Given the responses

Table 2. Notation for Eq. 3

Symbol	Description
$y_m^{(1)}(t)$	Responses over time of the mth pair of neurons, where m specifies to the predictive frequency
$y^{(0)}(t)$	Input over time
$\hat{y}_m^{(1)}(t)$	Predicted responses for the mth pair of neurons
$w_m^{(1)}$	Temporal weights (a pair of numbers that depend on ω_m) for the mth pair of neurons
$\omega_m^{(1)}$	Predictive frequency (a constant) for the mth pair of neurons
$\lambda(t)$	State parameter

of a pair of neurons at only one instant in time, the predicted responses over time are proportional to the predictive basis functions, scaled by the responses at that instant in time. Given the responses over time up to a current instant in time, the predicted responses can be expressed as a sum of scaled copies of the predictive basis functions. For example, when $\omega_m = 0$, the predictive basis function is an exponential decay, the response y_m is a low-pass filtered (blurred over time) copy of the input $y^{(0)}$, and the value of the state parameter λ determines the amount of blurring.

A change in state ($\lambda = 0.1$ versus $\lambda = 0.01$) corresponded to a change in neural response dynamics (Fig. 6 *C* and *D*). Different values of λ corresponded to linear filters with different temporal impulse response functions. During the first part of the simulation ($\lambda = 0.1$ for $t \leq 0$), the temporal impulse response function was relatively brief (Fig. 6*C*) and the temporal frequency bandwidth was correspondingly broad. During the second part of the simulation ($\lambda = 0.01$ for $t > 0$), however, the temporal impulse response function was extended in time (Fig. 6*D*) and the temporal frequency bandwidth was relatively narrow.

As an example of multilayer prediction, a network was implemented that predicted periodic visual motion (Fig. 7). The energy function for this multilayer network can be expressed by combining Eqs. **1** and **3** (*SI Appendix*). The visual stimulus was a sinusoidal grating pattern that moved periodically rightward and leftward. A simplified model of retinal processing consisted of a temporal filter at each spatial location (Fig. 7*A*). The output of this temporal filter at each spatial location served as the input to the network (Fig. 7*C*). Layer 1 of the network was a simplified model of the LGN, layer 2 was a simplified model of direction-selective V1 simple cells (Fig. 7*B*), and layer 3 was a simplified model of direction-selective V1 complex cells. There were two channels in layer 3, responding preferentially to leftward and rightward motion. The layer 3 responses modulated over time with the periodic motion (Fig. 7*E*), and they predicted that the modulation would continue. This modulation of the layer 3 responses fed back through layer 2 to layer 1 and constrained the predicted responses in layer 1 (Fig. 7*D*).

Discussion

This paper outlines a first step toward an empirically testable computational framework for cortical function, in which neural responses depend on a combination of feedforward drive (bottom-up input from the previous processing stage), feedback drive (top-down context from the next stage), and prior drive (expectation). Information processing is continuous and dynamic, and it predicts forward in time (or combines sensory information with different latencies). Noise serves to explore different possible interpretations (a form of stochastic optimization). Special cases of the theory approximate Bayesian inference/estimation in which the neural responses encode an implicit representation of a posterior probability distribution.

The theory is related to previous research in computational neuroscience and artificial intelligence. In some states, neural responses are dominated by the feedforward drive and the theory is identical to conventional feedforward models, thereby preserving all of the desirable features of those models. Specifically, with $\lambda = 1$ and with appropriate choices of weights, the theory is identical to convolutional neural nets used in artificial intelligence systems for object recognition (e.g., refs. 51 and 52), and to conventional hierarchical models of visual perception (e.g., refs. 27 and 53). In other states, the theory is a generative model (e.g., refs. 54–56) that constructs a sensory representation (e.g., in layer 1 of the example networks in this paper) from an abstract representation (e.g., in layer 3) via feedback. In still other states, the computational framework combines prior expectation with sensory input, and it explores different possible perceptual interpretations of ambiguous sensory inputs, akin to models based on Bayesian inference (2–5, 47–49). The noise-driven process of exploration was motivated by stochastic optimization algorithms (57) and is similar to models of sensory neuroscience that draw samples from an underlying probability distribution over possible percepts (9, 10, 39, 43). This sampling idea has been proposed as an alternative to probabilistic population codes (4) for representing uncertainty in neural systems. However, I see these two ideas as complementary, not mutually exclusive. Neural responses evolve dynamically over time in my networks, in part because of noise-driven exploration, while implicitly encoding a posterior probability distribution. I hypothesize that this noise-driven process of exploration is the essence of creativity.

Prediction Versus Predictive Coding. Predictive coding theories of sensory and perceptual processing have been developed to "explain away" the sensory input (e.g., refs. 19, 20, and 58–61). These theories posit two functionally distinct subpopulations of neurons, one representing predictions and the other representing prediction errors. Prediction errors are propagated forward to the next layer in the hierarchy and predictions are transmitted via feedback to the previous layer. The idea is to account for the incoming sensory signal by means of a matching top-down prediction, so that less prediction error propagates up the hierarchy. However, most of these models do not posit how the brain predicts over time (i.e., they do not extrapolate forward in time—see below).

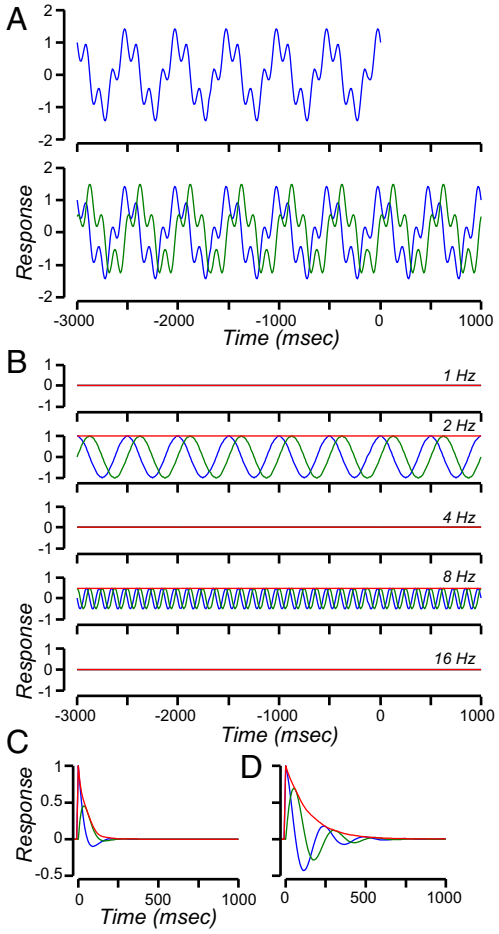

Fig. 6. Prediction. (*A*) Input and output. Top panel, input is a sum of two sinusoids for past time ($t \leq 0$) and nonexistent for future time ($t > 0$). Bottom panel, output. Blue curve, sum of the blue curves in *B*. Green curve, sum of the blue curves in *B*. State: $\lambda = 0.1$ for $t \leq 0$ and $\lambda = 0.01$ for $t > 0$. (*B*) Responses of each individual neuron. Different panels correspond to predictive basis functions with different temporal frequencies (ω_m). Blue and green curves in each panel, responses of pairs of neurons with the same ω_m but with temporal phases offset by 90°. Red curve in each panel, square root of the sum of the squares of the blue and green curves. (*C*) Impulse response functions. State: $\lambda = 0.1$. Predictive basis function temporal frequency: $\omega_m = 4$ Hz. Blue, green, and red curves, same convention as in *B*. (*D*) Impulse response functions for same pair of neurons as in *C* but different state: $\lambda = 0.01$. Time step: $\Delta t = 10$ ms for all four panels. See *SI Appendix* for details.

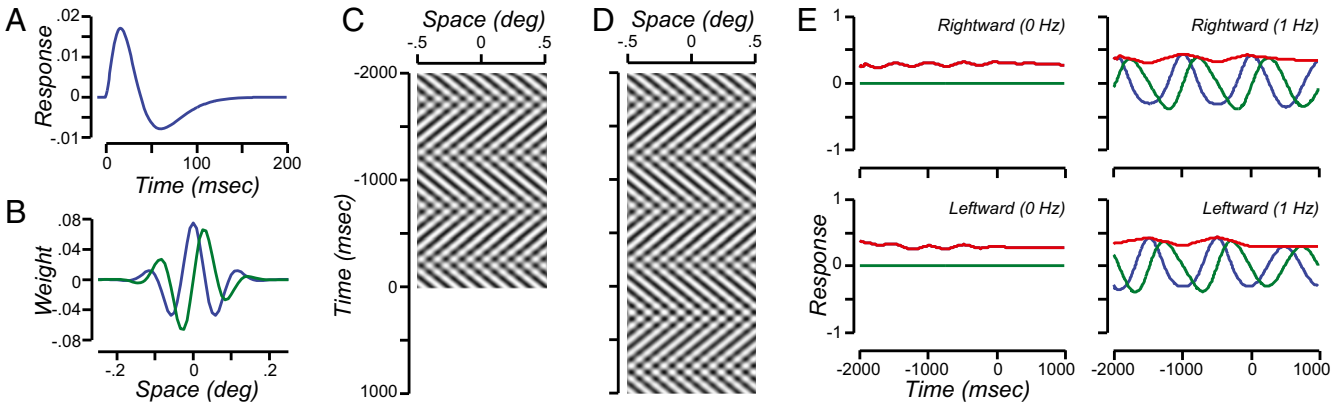

Fig. 7. Multilayer prediction of periodic motion. (*A*) Impulse response of retinal temporal filters. (*B*) Spatial weighting functions of layer 2 neurons. (*C*) Input to the network. Space–time responses of the retinal temporal filters, shown for one dimension of space over time, in response to periodic motion for $t \leq 0$. (*D*) Layer 1 responses. Responses followed the input for $t \leq 0$ and predicted continued periodic motion for $t > 0$. (*E*) Layer 3 responses modulated over time with the periodic motion of the stimulus. Blue, green, and red curves, same convention as in Fig. 6. Blue curves in left-hand panels (0 Hz) are identical to, and hidden by, the red curves. State for $t \leq 0$: $\lambda =$ (0.9, 0.9, 0.9) and $\alpha =$ (1, 1, 1), for layers 1, 2, and 3, respectively. State for $t > 0$: $\lambda =$ (0.9, 0.9, 0.001) and $\alpha =$ (0.001, 0.01, 1). Time step: $\Delta t = 10$ ms. See *SI Appendix* for details.

The feedforward and feedback drive in the current theory are analogous to those in the predictive coding models, but the variables are flipped. The representation is propagated forward and the errors are propagated backward. This is more in line with neurophysiological and psychophysical phenomena than predictive coding models. First, neurons exhibit sustained activity to a predictable visual stimulus (e.g., ref. 62). According to predictive coding theories, the forward-propagating responses correspond to the prediction error, which should rapidly decay to zero for a predictable stimulus. Neural activity typically decreases over time due to adaptation, but the responses do not go to zero, that is, they are not "explained away." Second, imagining a familiar image evokes a sensory representation in visual cortex that is reconstructed from memory (e.g., ref. 29), not explained away.

Moreover, the conceptual approach in predictive coding theories is fundamentally different from the current theory. Predictive coding theories start with a generative model that describes how characteristics of the environment produce sensory inputs. Perception is presumed to perform the inverse mapping, from sensory inputs to characteristics of the environment. The current theory is built the other way around ("synthesis-by-analysis"). I start with a feedforward cascade of signal processing operations, following the success of both feedforward models of sensory neuroscience and feedforward artificial neural nets. The corresponding generative model is the inverse of this feedforward processing model, which can be computed by gradient descent with respect to the input (55, 63). The energy function in the current theory combines a feedforward processing model and the corresponding generative model, so that it can run bottom-up (feedforward signal processing), top-down (generative), or a combination of the two.

There is a paucity of theories for how the brain performs perceptual predictions over time and/or combines sensory information with different latencies in the past (19–23). Most of the so-called predictive coding models (cited above) do not posit how the brain predicts over time. The predictive coding models that do so (19, 20, 22), as discussed above, are inconsistent with empirical phenomena. Other theories that perform predictions over time are neurally inspired implementations of a Kalman filter (21) or a Markov chain (23).

The current theory posits a different process for how the brain might predict over time. It relies on recursive computation similar to a Kalman filter, that is, the predictive basis functions serve the same role as the dynamical system model in a Kalman filter. Also like a Kalman filter, the neural responses in the current theory implicitly represent both estimates and uncertainties over time. However, unlike a Kalman filter, this computational framework comprises processing at multiple temporal scales, with different predictive frequencies at each level of the hierarchy. Multiple temporal scales of processing, across brain areas, have been proposed theoretically (22, 50, 64) and observed empirically (e.g., refs. 65–67). I hypothesize that this hierarchy of timescales is determined by the temporal weights (that specify the temporal frequencies of the predictive basis functions); neurons with temporal weights corresponding to lower temporal frequencies accumulate information over a longer time period in the past and are capable of predicting forward in time over a correspondingly longer timescale.

There is some controversy about whether sensory systems perform prediction versus what has been called "postdiction" in which sensory information acquired at different latencies (all in the past) is used to construct a percept of the past (68). However, there is no distinction between the two in the current theory; both involve extrapolating over time. An intriguing hypothesis is that sensory awareness is the brain's prediction of the present (e.g., ref. 16).

Learning. Most artificial neural nets rely on supervised learning. In computer vision, for example, an image is presented to a neural net, which attempts to categorize the image as one of a fixed set of possibilities. The network produces an output (e.g., at the top of the hierarchy in a deep convolutional neural net), which is compared with a desired output. The desired output is specified a priori (e.g., by hand-labeling the identity of an object in an image). The difference between the output and desired output is used to adjust the weights via "backpropagation" (gradient descent on the weights in every layer with respect to the error in the output). This requires a large library of images, each of which is prelabeled with a category.

The example networks presented in this paper were hand-tuned (ad hoc), but they could instead be learned using an unsupervised learning algorithm that extracts regularities in the inputs without labels. The neural responses are modeled as dynamical processes that compute a weighted sum of feedforward drive, feedback drive, and prior drive (Eq. **2**). The prior drive, in turn, depends on a weighted sum of previous responses over time (Eq. **3**). These spatial and temporal weights can be learned based on prediction errors for a time-varying input (e.g., video). Each neuron in the network produces a prediction for what its response will be later in time, and the weights are adjusted to minimize the difference between these predicted responses and the actual responses that occur later in time. This is similar to what has been called "target propagation" (as opposed to the more familiar backpropagation) in the neural net literature (69–71). Periods of inference, exploration, and prediction, during which the neural responses evolve dynamically (e.g., via gradient descent on the responses as in Eq. **2**) alternate (by changing the state of the network) with periods of learning during which the weights are updated (via gradient descent on the weights). Alternation between inference and learning, and learning based on

different timescales at each level of the hierarchy, are each reminiscent of previous unsupervised learning algorithms (50, 56).

A challenge for backpropagation-based learning is that it is nonlocal, requiring the weights in one channel/layer to be updated based on errors in other channels/layers. Nonlocality is considered by many to be biologically implausible, although there are some proposals for how to implement backpropagation with only local, biologically plausible weight updates (72–74). The idea here is to circumvent this problem entirely by updating each neuron's (spatial and temporal) weights locally, based only on that neuron's prediction errors.

The priors can also be learned (*SI Appendix*).

Brain States, Neuromodulators, and Oscillatory Activity. The values of the state parameters (α and λ) might be controlled by acetylcholine (ACh), given the evidence that ACh plays a role in modulating the trade-off between bottom-up sensory input versus top-down signals related to expectancy and uncertainty (*SI Appendix*). In addition, there is considerable evidence that attention modulates the gain of neural responses, suggesting that α might be controlled also by attention (*SI Appendix*). Neuromodulators might also control changes in state to enable learning (*SI Appendix*). According to the theory, exploration depends on neural response variability, which might be controlled (at least in part) by noradrenaline, and/or by oscillations in brain activity (*SI Appendix*).

Empirical Relevance. There are a number of variations of the computational framework, depending on the network architecture, output nonlinearity, and optimization algorithm (*SI Appendix*). Some of the variants or some of the components of the computational framework might be completely wrong, whereas others are less wrong; that is, falsifying just one variant or one component would not render the entire computational framework worthless. The simulation examples presented in this paper were designed to illustrate the principles of the theory, not to model the responses of any particular neurons or neural systems. It remains to be seen whether these principles can be applied to fit neurophysiological and/or behavioral data, and/or applied in computer vision or artificial intelligence systems. In the meantime, there are some general principles of the theory that are empirically relevant and/or that motivate experiments. Some examples are as follows:

i) According to the theory, prediction is performed recursively (Eq. **3**; Fig. 2*D*, dashed line) with pairs of neurons that have identical response properties except that they respond with different temporal phases (e.g., pairs of neurons with temporal phases offset by 90°, although any two or more phases would suffice). There is evidence that adjacent pairs of simple cells in V1 have receptive fields with 90° or 180° shifts in spatial phase (e.g., ref. 75), but temporal-phase relationships between nearby neurons have not been reported.

ii) The theory posits neurons with similar preferences for sensory stimuli, but with different dynamics, which together make up a basis set for predicting forward in time.

iii) Changing the state (the value of λ) corresponds to a change in neural response dynamics (Fig. 6 *C* and *D*). Such changes in state are hypothesized to be controlled by fluctuations in ACh (*SI Appendix*).

iv) Alternations in perceptual state (e.g., for bistable perceptual phenomena) depend on neural response reliability. Changes in neural response reliability are hypothesized to be driven by fluctuations in noradrenaline and by oscillations in brain activity (*SI Appendix*).

v) Functional connectivity between pairs of neurons is hypothesized to depend on brain state. When in a bottom-up sensory processing state, the feedback connections will appear to be weak. When in a top-down processing state, the feedback connections will be strong but the feedforward connections will appear weak. Functional connectivity is also hypothesized to depend on context. Take, for example, my simple XOR-like inference network (Fig. 3), and imagine an experiment to perturb the neural responses by injecting current in either the first of the layer 1 neurons (Fig. 3*A*, blue circle) or the first of the layer 2 neurons (Fig. 3*A*, purple circle), with state parameters that enable a combination of bottom-up and top-down processing. With input $y^{(0)} = (0.1, 0, 0, 0)$ and prior $\hat{y} = 0$ for all neurons in the network, positive perturbations of either the layer 1 neuron or the layer 2 neuron causes the other neuron to respond more. If the input is instead $y^{(0)} = (0.1, 1, 0, 0)$, then positive perturbations of either the layer 1 neuron or the layer 2 neuron causes the other neuron to respond less. Additionally, if the input is $y^{(0)} = (1, 1, 0, 0)$, then shutting down the layer 2 neuron causes a different neuron in layer 1 (Fig. 3*A*, green circle) to respond more. Given how complicated this is for such a simple network, I worry about how to interpret the results of optogenetic experiments in the absence of predictions from specific computational models.

Computational Theory. The current theory is intended, following the terminology of David Marr (76), to characterize cortical function at a computational level of abstraction (what the brain might be optimizing, e.g., Eqs. **1** and **3**), and at an algorithmic level of abstraction (signal-processing computations to perform the optimization, e.g., Eq. **2**), not in terms of the underlying circuit, cellular, molecular, and biophysical mechanisms. There are a number of variations of the optimization criterion, depending on the architecture of the network and choices for the nonlinearities. For each choice of optimization criterion, there are also a number of possible optimization algorithms (for which Eq. **2** is only one example). For any given choice of network architecture, optimization criterion, and optimization algorithm, there are a variety of mechanisms that might implement the computations embodied in the theory.

For example, I developed the normalization model 25 y ago to explain stimulus-evoked responses of V1 neurons (31). The model has since been applied to explain physiological measurements of neural activity in a wide variety of neural systems, and behavioral/perceptual analogs of those physiological phenomena (32) (*SI Appendix*). However, only recently has there been progress in elucidating the underlying mechanisms, which have been found to be different in different neural systems (*SI Appendix*).

Computational theory is an intermediate level of abstraction between the underlying mechanisms, on the one hand, and physiology and behavior, on the other (77). The field of neuroscience might benefit from the recognition, in other fields of science, that reductionism is inherently limited, and that there are fundamental organizing principles at intermediate levels (e.g., ref. 78). Computation might be such a critical intermediate level for characterizing brain function. Consequently, it may be extremely useful to identify abnormal computations in individuals with particular psychiatric and neurodevelopmental disorders. For example, deficits in normalization (79), deficits in prediction (80), dysfunctional Bayesian inference (81), and uncontrolled neural response variability (82, 83) have each been hypothesized to underlie autism.

ACKNOWLEDGMENTS. Special thanks go to Mike Landy, Eero Simoncelli, E. J. Chichilnisky, Jon Winawer, Weiji Ma, Paul Glimcher, Laura Dugué, Rachel Denison, Wayne Mackey, Matteo Carandini, Carlos Fernandez-Granda, and Marc'Aurelio Ranzato for comments and discussion.

1. von Helmholtz H (1925) *Treatise on Physiological Optics*; translated from the 3rd German Edition (1910) (Optical Society of America, Washington, DC).
2. Heeger DJ (1988) Optical flow using spatiotemporal filters. *Int J Comput Vis* 1: 279–302.
3. Heeger DJ, Simoncelli EP (1993) Model of visual motion sensing. *Spatial Vision in Humans and Robots*, eds Harris L, Jenkin M (Cambridge Univ Press, New York), pp 367–392.
4. Ma WJ, Beck JM, Latham PE, Pouget A (2006) Bayesian inference with probabilistic population codes. *Nat Neurosci* 9(11):1432–1438.

5. Knill DC, Pouget A (2004) The Bayesian brain: The role of uncertainty in neural coding and computation. *Trends Neurosci* 27(12):712–719.
6. Blake R, Logothetis N (2002) Visual competition. *Nat Rev Neurosci* 3(1):13–21.
7. Said CP, Heeger DJ (2013) A model of binocular rivalry and cross-orientation suppression. *PLoS Comput Biol* 9(3):e1002991.
8. Wilson HR, Krupa B, Wilkinson F (2000) Dynamics of perceptual oscillations in form vision. *Nat Neurosci* 3(2):170–176.
9. Moreno-Bote R, Knill DC, Pouget A (2011) Bayesian sampling in visual perception. *Proc Natl Acad Sci USA* 108(30):12491–12496.
10. Hoyer PO, Hyvarinen A (2003) Interpreting neural response variability as Monte Carlo sampling of the posterior. *Adv Neural Inf Process Syst* 15:293–300.
11. Gershman SJ, Vul E, Tenenbaum JB (2012) Multistability and perceptual inference. *Neural Comput* 24(1):1–24.
12. Sundareswara R, Schrater PR (2008) Perceptual multistability predicted by search model for Bayesian decisions. *J Vis* 8(5):12.1–19.
13. Crapse TB, Sommer MA (2008) Corollary discharge across the animal kingdom. *Nat Rev Neurosci* 9(8):587–600.
14. Kawato M (1999) Internal models for motor control and trajectory planning. *Curr Opin Neurobiol* 9(6):718–727.
15. Wolpert DM, Ghahramani Z, Jordan MI (1995) An internal model for sensorimotor integration. *Science* 269(5232):1880–1882.
16. Nijhawan R (2008) Visual prediction: Psychophysics and neurophysiology of compensation for time delays. *Behav Brain Sci* 31(2):179–198, discussion 198–239.
17. Palmer SE, Marre O, Berry MJ, 2nd, Bialek W (2015) Predictive information in a sensory population. *Proc Natl Acad Sci USA* 112(22):6908–6913.
18. Bialek W, Nemenman I, Tishby N (2001) Predictability, complexity, and learning. *Neural Comput* 13(11):2409–2463.
19. Rao RP, Ballard DH (1997) Dynamic model of visual recognition predicts neural response properties in the visual cortex. *Neural Comput* 9(4):721–763.
20. Wacongne C, Changeux JP, Dehaene S (2012) A neuronal model of predictive coding accounting for the mismatch negativity. *J Neurosci* 32(11):3665–3678.
21. Beck JM, Latham PE, Pouget A (2011) Marginalization in neural circuits with divisive normalization. *J Neurosci* 31(43):15310–15319.
22. Kiebel SJ, Daunizeau J, Friston KJ (2008) A hierarchy of time-scales and the brain. *PLoS Comput Biol* 4(11):e1000209.
23. Hawkins J, George D, Niemasik J (2009) Sequence memory for prediction, inference and behaviour. *Philos Trans R Soc Lond B Biol Sci* 364(1521):1203–1209.
24. Douglas RJ, Martin KA (1991) A functional microcircuit for cat visual cortex. *J Physiol* 440:735–769.
25. Douglas RJ, Koch C, Mahowald M, Martin KA, Suarez HH (1995) Recurrent excitation in neocortical circuits. *Science* 269(5226):981–985.
26. Heeger DJ, Simoncelli EP, Movshon JA (1996) Computational models of cortical visual processing. *Proc Natl Acad Sci USA* 93(2):623–627.
27. Simoncelli EP, Heeger DJ (1998) A model of neuronal responses in visual area MT. *Vision Res* 38(5):743–761.
28. Gilbert CD, Li W (2013) Top-down influences on visual processing. *Nat Rev Neurosci* 14(5):350–363.
29. Kosslyn SM, Ganis G, Thompson WL (2001) Neural foundations of imagery. *Nat Rev Neurosci* 2(9):635–642.
30. Marroquin JL, Figueroa JE, Servin M (1997) Robust quadrature filters. *J Opt Soc Am A Opt Image Sci Vis* 14:779–791.
31. Heeger DJ (1992) Normalization of cell responses in cat striate cortex. *Vis Neurosci* 9(2):181–197.
32. Carandini M, Heeger DJ (2011) Normalization as a canonical neural computation. *Nat Rev Neurosci* 13(1):51–62.
33. Körding KP, Wolpert DM (2004) Bayesian integration in sensorimotor learning. *Nature* 427(6971):244–247.
34. Griffiths TL, Tenenbaum JB (2006) Optimal predictions in everyday cognition. *Psychol Sci* 17(9):767–773.
35. Tenenbaum JB, Kemp C, Griffiths TL, Goodman ND (2011) How to grow a mind: Statistics, structure, and abstraction. *Science* 331(6022):1279–1285.
36. Wu HG, Miyamoto YR, Gonzalez Castro LN, Ölveczky BP, Smith MA (2014) Temporal structure of motor variability is dynamically regulated and predicts motor learning ability. *Nat Neurosci* 17(2):312–321.
37. Tumer EC, Brainard MS (2007) Performance variability enables adaptive plasticity of "crystallized" adult birdsong. *Nature* 450(7173):1240–1244.
38. Todorov E, Jordan MI (2002) Optimal feedback control as a theory of motor coordination. *Nat Neurosci* 5(11):1226–1235.
39. Fiser J, Berkes P, Orbán G, Lengyel M (2010) Statistically optimal perception and learning: From behavior to neural representations. *Trends Cogn Sci* 14(3):119–130.
40. Ganguli D, Simoncelli EP (2014) Efficient sensory encoding and Bayesian inference with heterogeneous neural populations. *Neural Comput* 26(10):2103–2134.
41. Zemel RS, Dayan P, Pouget A (1998) Probabilistic interpretation of population codes. *Neural Comput* 10(2):403–430.
42. Yu AJ, Dayan P (2005) Uncertainty, neuromodulation, and attention. *Neuron* 46(4):681–692.
43. Haefner RM, Berkes P, Fiser J (2016) Perceptual decision-making as probabilistic inference by neural sampling. *Neuron* 90(3):649–660.
44. Berkes P, Orbán G, Lengyel M, Fiser J (2011) Spontaneous cortical activity reveals hallmarks of an optimal internal model of the environment. *Science* 331(6013):83–87.
45. Black MJ, Sapiro G, Marimont DH, Heeger D (1998) Robust anisotropic diffusion. *IEEE Trans Image Process* 7(3):421–432.
46. Hampel FR, Ronchetti EM, Rousseeuw PJ, Stahel WA (1986) *Robust Statistics: The Approach Based on Influence Functions* (Wiley, New York).
47. Landy MS, Maloney LT, Johnston EB, Young M (1995) Measurement and modeling of depth cue combination: In defense of weak fusion. *Vision Res* 35(3):389–412.
48. Ernst MO, Banks MS (2002) Humans integrate visual and haptic information in a statistically optimal fashion. *Nature* 415(6870):429–433.
49. Fetsch CR, Pouget A, DeAngelis GC, Angelaki DE (2011) Neural correlates of reliability-based cue weighting during multisensory integration. *Nat Neurosci* 15(1):146–154.
50. Wiskott L, Sejnowski TJ (2002) Slow feature analysis: Unsupervised learning of invariances. *Neural Comput* 14(4):715–770.
51. Krizhevsky A, Sutskever I, Hinton GE (2012) ImageNet classification with deep convolutional neural networks. *Advances in Neural Information Processing Systems (NIPS 2012)*. Available at papers.nips.cc/paper/4824-imagenet-classification-with-deep-convolutional-neural-networks. Accessed October 15, 2015.
52. LeCun Y, Bottou L, Bengio Y, Haffner P (1998) Gradient-based learning applied to document recognition. *Proc IEEE* 86(11):2278–2324.
53. Riesenhuber M, Poggio T (1999) Hierarchical models of object recognition in cortex. *Nat Neurosci* 2(11):1019–1025.
54. Heeger DJ, Bergen JR (1995) Pyramid-based texture analysis/synthesis. *SIGGRAPH '95 Proceedings of the 22nd Annual Conference on Computer Graphics and Interactive Techniques* (ACM, New York), pp 229–238.
55. Portilla J, Simoncelli EP (2000) A parametric texture model based on joint statistics of complex wavelet coefficients. *Int J Comput Vis* 40(1):49–70.
56. Hinton GE, Dayan P, Frey BJ, Neal RM (1995) The "wake-sleep" algorithm for unsupervised neural networks. *Science* 268(5214):1158–1161.
57. Geman S, Geman D (1984) Stochastic relaxation, gibbs distributions, and the Bayesian restoration of images. *IEEE Trans Pattern Anal Mach Intell* 6(6):721–741.
58. Rao RP, Ballard DH (1999) Predictive coding in the visual cortex: A functional interpretation of some extra-classical receptive-field effects. *Nat Neurosci* 2(1):79–87.
59. Lee TS, Mumford D (2003) Hierarchical Bayesian inference in the visual cortex. *J Opt Soc Am A Opt Image Sci Vis* 20(7):1434–1448.
60. Friston K (2005) A theory of cortical responses. *Philos Trans R Soc Lond B Biol Sci* 360(1456):815–836.
61. Bastos AM, et al. (2012) Canonical microcircuits for predictive coding. *Neuron* 76(4):695–711.
62. Burns SP, Xing D, Shapley RM (2010) Comparisons of the dynamics of local field potential and multiunit activity signals in macaque visual cortex. *J Neurosci* 30(41):13739–13749.
63. Gatys LA, Ecker AS, Bethge M (2015) A neural algorithm of artistic style. arXiv:1508.06576.
64. Chaudhuri R, Knoblauch K, Gariel MA, Kennedy H, Wang XJ (2015) A large-scale circuit mechanism for hierarchical dynamical processing in the primate cortex. *Neuron* 88(2):419–431.
65. Hasson U, Yang E, Vallines I, Heeger DJ, Rubin N (2008) A hierarchy of temporal receptive windows in human cortex. *J Neurosci* 28(10):2539–2550.
66. Honey CJ, et al. (2012) Slow cortical dynamics and the accumulation of information over long timescales. *Neuron* 76(2):423–434.
67. Murray JD, et al. (2014) A hierarchy of intrinsic timescales across primate cortex. *Nat Neurosci* 17(12):1661–1663.
68. Eagleman DM, Sejnowski TJ (2000) Motion integration and postdiction in visual awareness. *Science* 287(5460):2036–2038.
69. Ranzato MA, Poultney C, Chopra S, LeCun Y (2006) Efficient learning of sparse representations with an energy-based model. *Advances in Neural Information Processing Systems* (MIT Press, Cambridge, MA), pp 1137–1144.
70. Carreira-Perpinán MA, Wang W (2014) Distributed optimization of deeply nested systems. *Proceedings of the 17th International Conference on Artificial Intelligence and Statistics (AISTATS) 2014*. Available at jmlr.org/proceedings/papers/v33/carreira-perpinan14.pdf. Accessed August 11, 2016.
71. Krogh A, Thorbergsson C, Hertz JA (1989) A cost function for internal representations. *Advances in Neural Information Processing Systems 2 (NIPS 1989)*. Available at https://papers.nips.cc/paper/229-a-cost-function-for-internal-representations.pdf. Accessed August 12, 2016.
72. O'Reilly RC (1996) Biologically plausible error-driven learning using local activation differences: The generalized recirculation algorithm. *Neural Comput* 8(5).
73. Whittington J, Bogacz R (2015) Learning in cortical networks through error back-propagation. bioRxiv:035451.
74. Bengio Y, Mesnard T, Fischer A, Zhang S, Wu Y (January 17, 2017) STDP as presynaptic activity times rate of change of postsynaptic activity approximates back-propagation. *Neural Computation*, 10.1162/NECO_a_00934.
75. Pollen DA, Ronner SF (1981) Phase relationships between adjacent simple cells in the visual cortex. *Science* 212(4501):1409–1411.
76. Marr D (1983) *Vision: A Computational Investigation into the Human Representation and Processing of Visual Information* (W. H. Freeman, New York).
77. Carandini M (2012) From circuits to behavior: A bridge too far? *Nat Neurosci* 15(4):507–509.
78. Laughlin RB, Pines D (2000) The theory of everything. *Proc Natl Acad Sci USA* 97(1):28–31.
79. Rosenberg A, Patterson JS, Angelaki DE (2015) A computational perspective on autism. *Proc Natl Acad Sci USA* 112(30):9158–9165.
80. Sinha P, et al. (2014) Autism as a disorder of prediction. *Proc Natl Acad Sci USA* 111(42):15220–15225.
81. Pellicano E, Burr D (2012) When the world becomes "too real": A Bayesian explanation of autistic perception. *Trends Cogn Sci* 16(10):504–510.
82. Dinstein I, et al. (2012) Unreliable evoked responses in autism. *Neuron* 75(6):981–991.
83. Dinstein I, Heeger DJ, Behrmann M (2015) Neural variability: Friend or foe? *Trends Cogn Sci* 19(6):322–328.
84. Gregory RL (2005) The Medawar Lecture 2001 Knowledge for vision: vision for knowledge. *Philos Trans R Soc Lond B Biol Sci* 360(1458):1231–1251.
85. Marroquin JL (1976) Human visual perception of structure. Master's thesis (MIT, Cambridge, MA).

Supporting Information

Heeger, *PNAS* 2017

Extended discussion

The theoretical framework presented in this paper, of course, includes components previously proposed in computational/theoretical neuroscience, image processing, computer vision, statistics, and machine learning with artificial neural networks. Because of space limitations, a number of influential papers were not cited in the main text (1-27).

Variants and extensions

There are a number of variations of the computational framework, depending on the network architecture, output nonlinearity, and optimization algorithm.

The neural responses were modeled as dynamical processes that minimize an energy function over time, via gradient descent. But other optimization algorithms might converge more rapidly while providing a better characterization of empirical measurements of neural dynamics.

The architecture (number of layers, number of channels per layer, interconnectivity between channels in adjacent layers) and spatial weights determine the selectivity of the neurons. I used an architecture and spatial weights that computed motion (Fig. 7 of the main text), but different choices would extract different features (or statistics) of the input. There need not be a strict hierarchy so that, for example, there can be feedforward connections from V1 to V2 to V4 and also a parallel feedforward connection directly from V1 to V4.

The temporal weights determine the predictive basis functions. I used temporal weights that conferred a set of predictive basis functions that are damped oscillators of various temporal frequencies, but different temporal weights might be used instead, corresponding to different predictive basis functions. An obvious variation is to replace the pair of temporal weights (w_m in Eq. **3** of the main text) with a matrix of weights so that the responses of each neuron are predicted over time by a weighted sum of a large number other neurons, including neighboring neurons in the same channel (e.g., that respond to stimuli at nearby spatial locations), and neurons from different channels in the same layer (e.g., that respond preferentially to different stimulus features). This is similar to classic recurrent network models of working memory that maintain a memory representation with a self-sustaining pattern of persistent activity (28-32), and also to some models of perceptual organization, segmentation, and grouping (33-36).

The convolutions can be replaced with an equivalent computation that encompasses the physiological diversity across individual neurons. The convolution weights that determine the selectivity of each neuron in each channel should be thought of as a basis set, with the first basis function equal to the first channel's weights, etc. For some basis sets and for some output nonlinearities (e.g., squaring), any invertible linear transform of the basis set can be substituted (37-39). A different invertible linear transform can be applied at each location, thereby allowing the weights to be different from one location to the next (and explaining the diversity of tuning properties of neurons), without changing the nature of the representation.

In the current implementation, the same neurons perform both inference and prediction, but an alternative implementation of the same principles would be to have two separate subpopulations of neurons. The first subpopulation would be responsible for inference (minimizing both terms in Eq. **3** of the main text), while the second subpopulation would be continuously predicting forward in time, based on the responses of the first subpopulation (minimizing only the second term in Eq. **3** of the main text). These two subpopulations of neurons might be in the same cortical circuit or the prediction subpopulation of neurons might be in a different brain area.

Normalization and other output nonlinearities

The examples in this paper, only for the sake of simplicity, used quadratic output nonlinearities, but a computation called "the normalization model" has been found to be a better model (both theoretically and empirically) of the output nonlinearity (40). I developed the normalization model 25 years ago to explain stimulus-evoked responses of individual neurons in V1 (41, 42). The model has since been applied to explain physiological measurements of neural activity in a wide variety of neural systems (43-60), and behavioral/ perceptual analogs of those physiological phenomena (e.g., 53, 54, 59, 61-63). The defining characteristic of normalization is that the feedforward drive underlying the response of each neuron is raised to a power (e.g., squaring) and divided by a factor that includes a sum of activity of a pool of neurons, analogous to normalizing the length of a vector (see below, Eq. **S3**). Squaring can be computed with a pair of neurons that have complementary weights (flipped in sign), each of which is half-squared (halfwave rectified and squared) and then summed (41). The half-squaring can be approximated by rectification with a high threshold (64, 65).

It has been known since the normalization model was first introduced that normalization can be implemented in a recurrent neural circuit with biophysically-plausible mechanisms (40, 42, 55, 66-68), but only recently has there been progress in elucidating the cellular and biophysical mechanisms underlying normalization. Normalization is implemented by GABA-mediated presynaptic inhibition in the olfactory system of the fruit fly (48, 69). Normalization in mammalian cortex, however, does not rely on GABA inhibition (70), but rather is caused by a decrease in excitation (71). That is, the mechanisms underlying normalization are different in different neural systems.

Sigmoids, rectified linear units, and max pooling are alternative output nonlinearities, common/popular in computational neuroscience and machine learning, that are each related to normalization. The normalization model, because of the division, confers a saturating (sigmoidal) response as a function of the amplitude of the inputs. A rectified linear unit computes a linear sum of its inputs and subtracts a constant bias, followed by halfwave rectification. The bias acts like a high threshold, that approximates a power function with different values of the bias corre-

sponding to different powers (64, 65). Max pooling (also called softmax) transmits the most active response among a set of inputs (72). Max pooling can be approximated by normalization (73).

Learning the prior

The priors can be learned. For a prior that constitutes a permanent feature of the environment, an elegant solution is to adjust the convolution weights (i.e., "warp" the tuning curves) to match the statistics of the environment (74). The current theory handles the priors in a complementary way. Some priors, rather than being a permanent feature of the environment, are instead context-specific (e.g., matched to a particular task). The cue combination network (Fig. 5 of the main text) provides an example. What I have in mind is that this cue combination network is embedded in a larger hierarchical network. The target values for the responses $\hat{y}$ are learned as the mean responses of the neurons, averaged across a series of practice/training trials in which the cues are consistent with one another (no cue conflict), and both cues are reliable (i.e., with large stimulus strengths). These learned target responses propagate up the hierarchy, transformed to an abstract representation, and stored in memory. Just before each trial of the task, this abstract representation is recalled from memory at the top of the hierarchy, and the state of the network is set to behave like a generative model so the remembered prior is propagated via the feedback drive to a sensory representation, i.e., to reconstruct the target response values. The state is then switched so that this sensory representation of the priors is combined with incoming sensory information to perform inference.

Brain states, neuromodulators, and oscillatory activity

The values of the state parameters (α and λ) determine whether neural responses are driven bottom-up, top-down, or a combination of the two. These parameters also control whether the neurons are primarily processing sensory inputs that occurred in the past versus predicting the future. There is evidence that acetylcholine (ACh) plays a particular role in modulating the trade-off between bottom-up sensory input versus top-down signals related to expectancy and uncertainty (e.g., 75). It has also been hypothesized that ACh signals when bottom-up sensory inputs are known to be reliable (76, 77). Consequently, it is reasonable to hypothesize that α and/or λ might be controlled (at least in part) by ACh. Although ACh is released broadly throughout the cortex, its effect can be regionally specific (78), possibly offering a mechanism for how the values of the state parameters can differ across the hierarchy of brain areas.

In addition, there is considerable evidence that attention modulates the gain of neural responses (51), suggesting that α might be controlled also by attention, perhaps through the feedback drive (see paragraph above about learning the prior) or through a different set of feedback connections that modulate the gain of the convolutions v.

Neuromodulators might also control changes in state to enable learning. During inference, the neural responses are computed dynamically with fixed weights. During learning, the weights are adjusted to minimize the difference between the predicted and the actual neural responses. Neuromodulators might indicate when it is appropriate to adjust the weights (e.g., moments in time corresponding to prediction errors). Dopamine, for example, has been identified as signaling reward prediction-error (79).

According to the theory, exploration depends on neural response variability, which might be controlled (at least in part) by noradrenaline (NA). Specifically, I added non-stationary noise to the simulated neural responses to implement a kind of stochastic optimization. I speculate that the time course of spontaneous NA fluctuations might contribute to the time-varying standard deviation of this non-stationary noise process. Subthreshold fluctuations in NA over time (as assessed by measuring pupil dilation) affect neural response variability (80). Neural response variability exhibits an inverted U-shaped curve as a function of membrane potential depolarization such that responses are most reliable for an intermediate level of depolarization and less reliable when the neural membrane potential is either too close or too far from spike threshold. Neural membrane-potential depolarization and pupil size both depend on NA. For example, NA fluctuations might exhibit a $1/f$ amplitude spectrum (81). Such a noise process can be computed by integrating white noise over time (analogous to the position of a particle undergoing Brownian motion); doing so with a leaky integrator is biologically plausible given the ubiquity of neural integrators (31, 32). It has been hypothesized that NA signals when something unexpected has occurred (76, 77), which would, according to the present theory, transiently increase the noise variance to explore alternative interpretations. NA has also been linked to alternations (i.e., exploration) during bistable perception (82), an observation that might be explained by the current theory if perception is stable when the neural response variability is low and prone to alternations when response variability is high.

This non-stationary noise process might also contribute to variability over time in behavioral performance. Measurements of behavioral performance as a function of arousal exhibit an inverted U-shaped function, which is hypothesized to be caused by the relationship between NA and neural response variability (80, 83-85). It has been reported, for example, that residual reaction time (after subtracting the mean reaction time for any given experimental condition) exhibits a $1/f$ power spectrum for a variety of tasks (86). Behavioral measures of timing and tapping also exhibit $1/f$ power spectra (87, 88).

Neural response variability might also be controlled (in part) by oscillations in brain activity, pseudo-periodic fluctuations in neural membrane potential, correlated across large populations of neurons. Such brain oscillations are readily observed with EEG, a well-known example of which is so-called alpha activity (~10 Hz). Subthreshold fluctuations in neural membrane potential affect neural response variability, as summarized above (80). I presume that such fluctuations have an impact on the reliability of stimulus-evoked activity with little or no impact on the mean responses (i.e., that the fluctuations are small in any given neuron but that they are evident in EEG recordings which measures the correlated component of the mem-

brane potential fluctuations across a large population of neurons). So I hypothesize that oscillations in brain activity might contribute to stochastic optimization for exploring alternative perceptual and/or cognitive interpretations. The oscillation phase corresponding to minimal response variability would correspond to the more stable percepts and the phase corresponding to maximum response variability would correspond to less stable percepts. These periodic fluctuations in response variability (in service of optimization) might, therefore, explain the empirical evidence for perceptual rhythms, i.e., that perception and perceptual performance fluctuate periodically and depend on the frequency and phase of oscillatory activity (89).

Methods and derivations

Feedforward convolutional neural net

Deep convolutional neural nets have an architecture that is based on a common model of sensory processing in the visual system, comprising a feedforward (pipeline processing) hierarchy of stages each comprising a bank of linear filters following by an output nonlinearity (Figs. 2A,B of the main text). This hierarchy of computations can be expressed as follows:

$$y_{jn}^{(i)} = \rho_z\left(v_{jn}^{(i)}\right) \qquad \textbf{[S1]}$$
$$v_{jn}^{(i)} = \sum_{q=1}^{N^{(i-1)}} \sum_{k} w_{jknq}^{(i-1)} y_{kq}^{(i-1)} \quad .$$

The values of y are the responses (proportional to firing rates) of the neurons in each layer, v are the outputs of the linear weighted sums, w are weight matrices, and ρ_z is the output nonlinearity. The superscript (i) specifies the layer in the hierarchy; $y^{(0)}$ are the inputs to the multi-layered hierarchy. The subscripts n and q specify each of the channels in a layer, where $N^{(i)}$ is the number of channels in layer (i). The subscripts j and k specify the different neurons in a channel. The values of w_{jknq} specify a matrix of weights connecting the k^{th} neuron in channel q of layer $(i\text{-}1)$ to the j^{th} neuron of channel n of layer (i). For all neurons in a channel, the weight matrices are assumed to be spatially shifted copies of one another (i.e., performing a spatial convolution, optionally with spatial subsampling). I have included the subscripts n and q in w_{jkqn} only to clarify that the weights are different for different channels.

The examples in this paper use either linear outputs or quadratic output nonlinearities:

$$\rho_z(v) = v \qquad \textbf{[S2]}$$
$$\rho_z(v) = \frac{1}{2}v^2 \quad .$$

Normalization is a more sophisticated model of the nonlinearity (40). The defining characteristic of normalization is that the response of each neuron is divided by a factor that includes a sum of activity of a pool of neurons:

$$\rho_Z\left(v_{jn}^{(i)}\right) = \frac{\left(v_{jn}^{(i)}\right)^2}{\sum_q \sum_k \beta_{kqn}^{(i)} \left(v_{kq}^{(i)}\right)^2 + \left(\sigma^{(i)}\right)^2} \quad . \qquad \textbf{[S3]}$$

The summation in the denominator is a weighted sum (i.e., local average) over neurons in the same layer with weights β. For each neuron j in channel n, these weights β_{kq} are assumed to be spatially shifted copies of one another (i.e., performing a spatial convolution). I have included the subscript n in β_{kqn} only to clarify that the weights β_{kq} are different for different channels. The constant σ determines the contrast gain (the contrast of the visual stimulus that evokes half the maximal response).

Theory of Cortical Function

I hypothesize that neural responses minimize an energy function (or optimization criterion) across all neurons in all channels and layers (and a summation over time can also be included, see below):

$$E = \sum_{i=1}^{L} \sum_{n} \sum_{j} \alpha^{(i)} \lambda^{(i)} \rho_l\left(y_{jn}^{(i)} - z_{jn}^{(i)}\right) + \sum_{i=1}^{L} \sum_{n} \sum_{j} \alpha^{(i)} \left(1-\lambda^{(i)}\right) \rho_p\left(y_{jn}^{(i)} - \hat{y}_{jn}^{(i)}\right) \qquad \textbf{[S4]}$$
$$z_{jn}^{(i)} = \rho_z\left(v_{jn}^{(i)}\right)$$
$$v_{jn}^{(i)} = \sum_{q=1}^{N^{(i-1)}} \sum_{k} w_{jknq}^{(i-1)} y_{kq}^{(i-1)} \quad .$$

This is a generalization of Eq. **1** of the main text with multiple channels in each layer and a flexible choice for the output nonlinearities and cost functions. The values of y are again the neural responses (proportional to firing rates). The values of v are again the outputs of the linear weighted sums from the previous layer. The values of z are now the outputs after the nonlinearity (unlike the more common formulation above in which y are the outputs after the nonlinearity). The function ρ_z is again the output nonlinearity (Eq. **S2**). The values of $\hat{y}$ in the second term represent a prior (or expectation) for the responses. These variables (y, x, v, z, and $\hat{y}$) are each functions of time because the inputs change over time with the sensory input. The functions ρ_l and ρ_p are cost functions, which are quadratic for the examples in this paper:

$$\rho_l(u) = \frac{1}{2}u^2 \qquad \rho_p(u) = \frac{1}{2}u^2 \quad , \qquad \textbf{[S5]}$$

although other cost functions could be readily substituted. The values of α and λ ($0 < \lambda < 1$) are state parameters that determine the tradeoffs between the two terms in the energy function at each layer.

The neural responses are modeled as dynamical processes that minimize this energy function over time (dropping the channel subscript n to simplify notation):

$$\tau \frac{dy_j^{(i)}}{dt} = -\frac{dE}{dy_j^{(i)}} \quad \textbf{[S6]}$$

The derivative of the energy function with respect to each neuron's response (using quadratic output nonlinearities and quadratic cost functions) is:

$$\frac{dE}{dy_j^{(i)}} = \alpha^{(i)}\lambda^{(i)}\left(y_j^{(i)} - z_j^{(i)}\right) + \alpha^{(i)}\left(1-\lambda^{(i)}\right)\left(y_j^{(i)} - \tilde{y}_j^{(i)}\right) + \sum_k \frac{dE}{dz_k^{(i+1)}}\frac{dz_k^{(i+1)}}{dy_j^{(i)}} \quad \textbf{[S7]}$$

Combining the previous two equations yields the following dynamical system in which each neuron's response is updated over time:

$$\begin{aligned} \tau \frac{dy_j^{(i)}}{dt} &= -\alpha^{(i)}\lambda^{(i)} f_j^{(i)} + \alpha^{(i+1)}\lambda^{(i+1)} b_j^{(i)} - \alpha^{(i)}\left(1-\lambda^{(i)}\right) p_j^{(i)} \\ f_j^{(i)} &= y_j^{(i)} - z_j^{(i)} \\ b_j^{(i)} &= \sum_k \left[y_k^{(i+1)} - z_k^{(i+1)} \right] v_k^{(i+1)} w_{kj}^{(i)} \\ p_j^{(i)} &= y_j^{(i)} - \hat{y}_j^{(i)} \end{aligned} \quad \textbf{[S8]}$$

This is the same as Eq. **2** of the main text except that I have included factors of 1/2 in the quadratic output nonlinearity and the quadratic cost function. As noted in the main body of the paper, the first term in this expression is the feedforward drive f; with only this term the neural responses would be the same as the feedforward model outlined above (i.e., $y = z$). The second term is the feedback drive b; this term drives the responses according to the mismatch between the responses at the next layer, i+1, and the feedforward drive from the i^{th} layer. The third term is the prior drive p; with only this term the neural responses would be driven to the value of the prior (i.e., $y = \hat{y}$). The value of τ is a time constant.

Feedback connections

As noted in the main body of the paper, the feedback signals are selective for features that are represented at the earlier layer due to the transpose of the weight matrix. A simplified two-layer example illustrates:

$$\begin{aligned} E &= \frac{1}{2}\sum_j \left(y_j^{(2)} - z_j^{(2)}\right)^2 \\ z_j^{(2)} &= \frac{1}{2}\left(\sum_k w_{jk} y_k^{(1)}\right)^2 = \frac{1}{2}\left(v_j^{(2)}\right)^2 \\ v_j^{(2)} &= \sum_k w_{jk} y_k^{(1)} \\ \frac{dE}{dy_k^{(1)}} &= \sum_j \frac{dE}{dz_j^{(2)}}\frac{dz_j^{(2)}}{dy_k^{(1)}} = -\sum_j \left(y_j^{(2)} - z_j^{(2)}\right) w_{jk} v_j^{(2)} \end{aligned} \quad \textbf{[S9]}$$

In the form of a matrix tableau:

$$\begin{pmatrix} \vdots \\ v_j^{(2)} \\ \vdots \end{pmatrix} = \begin{pmatrix} \ddots & & \\ & w_{jk} & \\ & & \ddots \end{pmatrix} \begin{pmatrix} \vdots \\ y_k^{(1)} \\ \vdots \end{pmatrix} \quad \textbf{[S10]}$$

$$\begin{pmatrix} \vdots \\ \dfrac{dE}{dy_k^{(1)}} \\ \vdots \end{pmatrix} = -\begin{pmatrix} \ddots & & \\ & w_{kj} & \ddots \end{pmatrix} \begin{pmatrix} \vdots \\ \left(y_j^{(2)} - z_j^{(2)}\right) v_j^{(2)} \\ \vdots \end{pmatrix}.$$

The feedforward drive depends on $v_j^{(2)}$, which is computed as a weighted sum of the layer 1 responses $y_k^{(1)}$ with weights w_{jk}. The gradients of the energy function $\mathrm{d}E/\mathrm{d}y_k^{(1)}$, which determine the feedback drives, are computed as a weighted sum of the mismatch between the responses and the feedforward drive $(y_j^{(2)}\text{-}z_j^{(2)})\ v_j^{(2)}$ using the transpose of the weight matrix w_{kj}.

Inference (Fig. 3)

For each of the simulation results in Fig. 3 of the main text, the input, the prior $\hat{y}$, and the network state (determined by the values of λ and α) were all held constant over time. The responses of the neurons were initialized to small, random values ($0 < y < 0.1$) at time $t = 0$. The responses were computed with Eq. **2** of the main text (time constant: $\tau = 5$ ms; time step: $\Delta t = 1$ ms), and the values were clipped ($0 < y < 1$) after each iteration.

Exploration (Fig. 4)

The responses were again computed with Eq. **2** of the main text (time constant: $\tau = 5$ ms; time step: $\Delta t = 10$ ms), the values were again clipped ($0 < y < 1$), and noise was added to each neuron's response at each time step. The noise was statistically independent across neurons and over time, but non-stationary. All neurons had the same noise standard deviation at each moment in time, but the noise standard deviation varied over time. Specifically, the time course of the standard deviation had a $1/f$ amplitude spectrum for frequencies greater than ~1 Hz. The noise process was computed by taking Gaussian white noise and filtering it with a leaky integrator (i.e., a first-order differential equation or exponential low pass filter) with time constant = 100 ms. The noise added to each neuron at each time point was drawn (independently for each neuron and each time point) from a normal distribution with the corresponding standard deviation.

One-layer time-series prediction (Fig. 6)

The one-layer time-series prediction network (Fig. 6 of the main text) optimized the following energy function:

$$E = \frac{1}{2}\sum_t \lambda(t)\left[\left(\sum_m \mathrm{Re}\left(y_m^{(1)}(t)\right)\right) - y^{(0)}(t)\right]^2 + \frac{1}{2}\sum_t (1-\lambda(t))\left[\sum_m \left|y_m^{(1)}(t) - \hat{y}_m^{(1)}(t)\right|^2\right] \quad \textbf{[S11]}$$

$$\hat{y}_m^{(1)}(t) = y_m^{(1)}(t-\Delta t)\, w_m^{(1)}$$

$$w_m^{(1)}(\Delta t) = e^{i 2\pi \omega_m^{(1)} \Delta t}.$$

This is a different way of writing Eq. **3** of the main text using the notational convenience of complex numbers and complex exponentials (instead of sines and cosines). It is a global optimization criterion; the summation is over all neurons and over time. The values of y_m are the complex-valued responses of a subpopulation of neurons that share the same input $y^{(0)}$, the values of ω_m specify the frequencies of the predictive basis functions, w_m are temporal weights (a pair of numbers for a each ω_m), and Δt is a discrete time step. The complex values can be represented by the responses of a pair of neurons (Eq. **3** of the main text), but the complex-exponential notation is convenient. The state parameter λ can change over time.

The derivative of E with respect to $y_m(t)$, can be used to find a local minimum of E by gradient descent:

$$\Delta y_m^{(1)}(t) = -r\frac{\partial E}{\partial y_m^{(1)}(t)} = -r\left[f_m^{(1)}(t) + p_m^{(1)}(t)\right], \quad \textbf{[S12]}$$

where f_m is the feedforward drive (note that there is no feedback drive in this one-layer example), p_m is the prior drive, r specifies a step size, and $y_m(t)$ is updated simultaneously for all time points t. I used Eq. **S12** to implement a batch algorithm, to compute the global minimum for all neurons and all time samples (Fig. 6 of the main text). This batch algorithm updated all of the neural responses at all time samples repeatedly until it converged. Other optimization algorithms could be used instead; For example, I have implemented an incremental approximation (see below).

The expressions for f_m and p_m depend on whether there is an input (for $t \le 0$) or not (for $t > 0$), and whether or not t is an endpoint (e.g., for a finite duration simulation and/or with an incremental algorithm for which $t = 0$ is always an endpoint because the input for the next time step is in the future). The feedforward drive is:

$$f_m^{(1)}(t) = \lambda\left[\mathrm{Re}\left(\sum_m y_m^{(1)}(t)\right) - y^{(0)}(t)\right], \quad \textbf{[S13]}$$

when there is an input and 0 otherwise. The prior drive is:

$$p_m^{(1)}(t) = (1-\lambda(t))\left(2y_m^{(1)}(t) - y_m^{(1)}(t-\Delta t)w_m^{(1)}(\Delta t) - y_m^{(1)}(t+\Delta t)w_m^{(1)}(-\Delta t)\right)$$
$$p_m^{(1)}(t) = (1-\lambda(t))\left(y_m^{(1)}(t) - y_m^{(1)}(t+\Delta t)w_m^{(1)}(-\Delta t)\right) \quad \textbf{[S14]}$$
$$p_m^{(1)}(t) = (1-\lambda(t))\left(y_m^{(1)}(t) - y_m^{(1)}(t-\Delta t)w_m^{(1)}(\Delta t)\right),$$

when t is not an endpoint, when t is the first time sample, and when t is the last time sample, respectively.

For the simulation results (Fig. 6 of the main text), the input was a sum of two sinusoids (2 Hz, amplitude 1; 8 Hz, amplitude 1/2) for past time ($t \le 0$) and nonexistent for future time ($t > 0$). I.e., the first term of E in Eq. **S11** was set to 0 (ignoring the input entirely) for $t > 0$. This could be implemented with two separate subpopulations of neurons, one of which minimizes both terms in Eq. **S11** and is responsive to the input, while the second subpopulation minimizes only the second term in Eq. **S11** and is continuously predicting forward in time (see *Variants and extensions*). Regardless, this is different from setting the input to 0 and minimizing both terms of E. If the input was set to 0 for $t > 0$ (rather than ignoring it entirely) then the responses decayed over time; the value of λ determined rate at which the responses decayed (see below).

Unlike the examples in Figs. 3 and 4 of the main text, the responses were not clipped. The negative values for the responses can be accommodated with positive firing rates by replacing each quadrature pair with 4 neurons, each with halfwave-rectified responses and 4 different temporal phases offset by 90°.

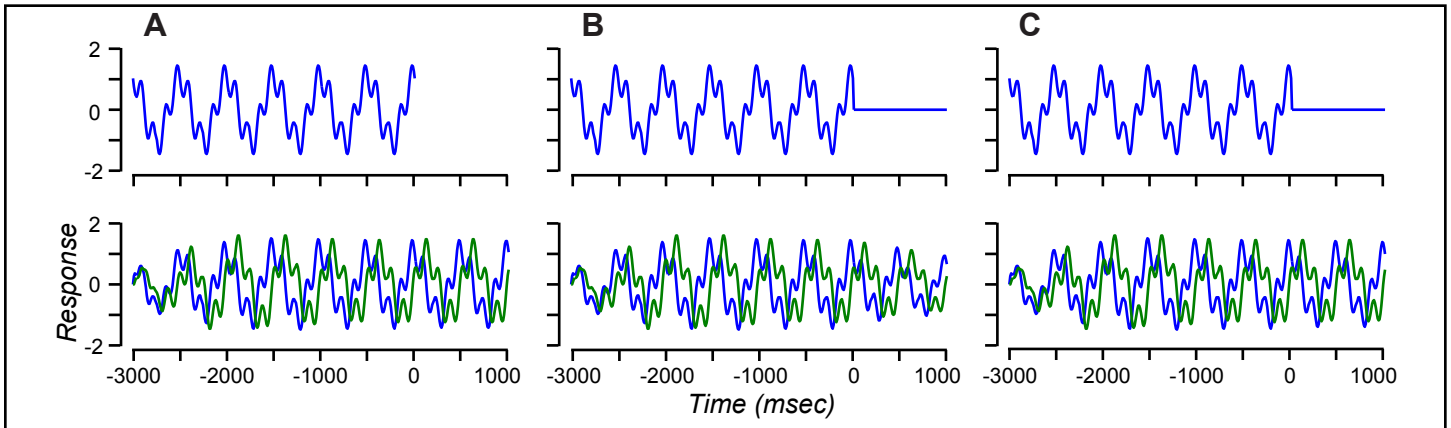

Fig. S1. One-layer time-series prediction with incremental algorithm. Compare with Fig. 6A of the main text. **A.** Input is a sum of two sinusoids for past time ($t \le 0$) and nonexistent for future time ($t > 0$). **B,C.** Input is 0 for $t > 0$. Top row, input. Bottom row, output. Blue curves in the bottom row, sum of the responses of the neurons representing the real parts of y_m. Green curves, sum of neural responses representing the imaginary parts of y_m. **A,B.** State: $\lambda = 0.1$ for $t \le 0$ and $\lambda = 0.01$ for $t > 0$ (same as Fig. 6 of the main text). **C.** State: $\lambda = 0.1$ for $t \le 0$ and $\lambda = 0.001$ for $t > 0$.

An incremental (causal) algorithm was also implemented (Fig. S1), analogous that in Eq. **S8**. The prior drive for the incremental algorithm used the 3rd line of Eq. **S14**, so that the change in responses depended on only the present input and the present and past responses. In practice, fewer than 10 iterations were needed for each time step of the incremental algorithm. The results depended on whether the input was nonexistent for $t > 0$ (Fig. S1A) or 0 for $t > 0$ (Figs. S1B,C). For nonexistent input, the first term of E in Eq. **S11** was set to 0 (ignoring the input entirely) for $t > 0$. If the input was set to 0 for $t > 0$ (rather than ignoring it entirely) then both terms of E were minimized, and the responses decayed over time; the value of λ determined rate at which the responses decayed (Figs. S1B,C).

Multi-layer prediction of periodic motion (Fig. 7)

The multi-layer prediction network optimized the following energy function:

$$E = \frac{1}{2}\sum_{i=1}^{L}\sum_{n}\sum_{t}\alpha^{(i)}(t)\lambda^{(i)}(t)\left[\left(\sum_{m}\mathrm{Re}\left(y_{nm}^{(i)}(x,t)\right)\right) - z_n^{(i)}(x,t)\right]^2 + \frac{1}{2}\sum_{i=1}^{L}\sum_{n}\sum_{t}\alpha^{(i)}(t)\left(1-\lambda^{(i)}(t)\right)\left[\sum_{m}\left(y_{nm}^{(i)}(x,t) - \hat{y}_{nm}^{(i)}(x,t)\right)^2\right]$$

$$\hat{y}_{nm}^{(i)}(x,t) = y_{nm}^{(i)}(x,t-\Delta t)\; w_m^{(i)} \qquad \textbf{[S15]}$$

$$w_m^{(i)}\left(\Delta t\right) = e^{i2\pi\omega_m^{(i)}\Delta t}.$$

Here, I dropped the subscript j and instead use x to denote the different neurons in each channel in terms of the spatial locations of their receptive fields. The rest of the notation is defined above (Eqs. **S4**, **S11**). It is again a global optimization criterion; the summation is over all neurons in all channels and all layers, and over time.

To denote the specific multi-layer motion-prediction network (Fig. 7 of the main text), it is helpful to break it down and write each layer separately. The total energy was the sum of the energies in each layer:

$$E = E^{(1)} + E^{(2)} + E^{(3)}. \qquad \textbf{[S16]}$$

Layer 1 had one channel:

$$E^{(1)} = \frac{1}{2}\sum_{t}\alpha^{(1)}\lambda^{(1)}\left[\mathrm{Re}\left(\sum_{m}y_m^{(1)}\right) - y^{(0)}\right]^2 + \frac{1}{2}\sum_{t}\sum_{m}\alpha^{(1)}\left(1-\lambda^{(1)}\right)\left(y_m^{(1)} - \hat{y}_m^{(1)}\right)^2 \qquad \textbf{[S17]}$$

The values of $y^{(0)}(x,t)$ are the output of a simplified model of retinal processing consisting of a temporal filter at each spatial location (see below for details). The responses y are functions of both space and time, and the state parameters α and λ also change over time, but I have dropped x and t from this equation (and in most of those that follow) to simplify the notation.

Layer 2 had four channels:

$$E^{(2)} = \frac{1}{2}\sum_{t}\sum_{n}\alpha^{(2)}\lambda^{(2)}\left[\mathrm{Re}\left(\sum_{m}y_{nm}^{(2)}\right) - z_j^{(2)}\right]^2 + \frac{1}{2}\sum_{t}\sum_{n}\sum_{m}\alpha^{(2)}\left(1-\lambda^{(2)}\right)\left(y_{nm}^{(2)} - \hat{y}_{nm}^{(2)}\right)^2 \qquad \textbf{[S18]}$$

$$z_n^{(2)}(x,t) = \frac{1}{2}\left(v_n^{(2)}(x,t)\right)^2$$

$$v_n^{(2)}(x,t) = \sum_{\xi}\left[w_{n1}^{(1)}(\xi - x)\sum_{m}\mathrm{Re}\left(y_m^{(1)}(x,t)\right)\right] + \sum_{\xi}\left[w_{n2}^{(1)}(\xi - x)\sum_{m}\mathrm{Im}\left(y_m^{(1)}(x,t)\right)\right],$$

The last line expresses v_n as a sum of convolutions, where n indexes the 4 channels, and w_{n1} and w_{n2} are the spatial weights of the convolution kernels (Fig. 7C of the main text; Eq. **S23**). The derivatives, used for gradient descent, are:

$$\frac{dE^{(2)}}{dz_n^{(2)}} = -\alpha^{(2)}\lambda^{(2)}\sum_{n}\left[\left(\sum_{m}y_{nm}^{(2)}\right) - z_n^{(2)}\right] \qquad \textbf{[S19]}$$

$$\frac{dz_n^{(2)}(x,t)}{dy_m^{(1)}(x,t)} = v_n^{(2)}(x,t)\frac{dv_n^{(2)}(x,t)}{dy_m^{(1)}(x,t)} = v_n^{(2)}(x,t)\left[w_{n1}^{(1)}(x) + iw_{n2}^{(1)}(x)\right].$$

Layer 3 had two channels:

$$E^{(3)} = \frac{1}{2}\sum_{t}\sum_{n}\alpha^{(3)}\lambda^{(3)}\left[\mathrm{Re}\left(\sum_{m}y_{nm}^{(3)}\right) - z_n^{(3)}\right]^2 + \frac{1}{2}\sum_{t}\sum_{n}\sum_{m}\alpha^{(3)}\left(1-\lambda^{(3)}\right)\left(y_{nm}^{(3)} - \hat{y}_{nm}^{(3)}\right)^2 \qquad \textbf{[S20]}$$

$$z_1^{(3)} = \sum_{m}\mathrm{Re}\left(y_{1m}^{(2)}\right) + \sum_{m}\mathrm{Re}\left(y_{2m}^{(2)}\right)$$

$$z_2^{(3)} = \sum_{m}\mathrm{Re}\left(y_{3m}^{(2)}\right) + \sum_{m}\mathrm{Re}\left(y_{4m}^{(2)}\right)$$

$$\frac{dE^{(3)}}{dz_n^{(3)}} = -\alpha^{(3)}\lambda^{(3)}\sum_{n}\left[\left(\sum_{m}y_{nm}^{(3)}\right) - z_n^{(3)}\right]$$

$$\frac{dz_1^{(3)}}{dy_{1m}^{(2)}} = 1 \quad \frac{dz_1^{(3)}}{dy_{2m}^{(2)}} = 1 \quad \frac{dz_2^{(3)}}{dy_{3m}^{(2)}} = 1 \quad \frac{dz_2^{(3)}}{dy_{4m}^{(2)}} = 1,$$

and the other derivatives of $z_j^{(3)}$ with respect to $y_{km}^{(2)}$ are zero.

The simulation results in Fig. 7 of the main text were computed as follows. The input was a sum of two contrast-modulated sinusoids for past time ($t \le 0$):

$$s(x,t) = c(t)\sin(2\pi\omega_x x - 2\pi\omega_t t) + [1-c(t)]\sin(2\pi\omega_x x + 2\pi\omega_t t)$$
$$c(t) = \frac{1}{2}[1+\cos(2\pi\omega_m t)] \quad \textbf{[S21]}$$

where $\omega_m = 1$ Hz was the modulation frequency, $\omega_x = 8$ cycle/deg was the spatial frequency, and $\omega_t = 8$ Hz was the grating temporal frequency, so that the speed of motion was 1 deg/sec. The stimulus was sampled spatially with 120 samples per degree of visual angle (approximately equal to the sampling of cone photoreceptors in the fovea of the primate retina) and with 1 ms temporal sampling.

The input was nonexistent for future time ($t > 0$). As for the simulation in Fig. S1, the responses decayed to 0 over time if the input was set to 0 for $t > 0$ (rather than ignoring it entirely), and the value of λ determined rate at which the responses decayed.

A simplified model of retinal processing was computed as a cascade of exponential low-pass filters (Fig. 7A of the main text):

$$y^{(0)}(x,t) = f_3(x,t) - f_5(x,t)$$
$$\tau_f \frac{df_1(x,t)}{dt} = -f_1(x,t) + s(x,t)$$
$$\tau_f \frac{df_{n+1}(x,t)}{dt} = -f_{n+1}(x,t) + f_n(x,t) \quad \textbf{[S22]}$$

where $y^{(0)}$ was the retinal output (i.e., the input to the multi-layer motion-prediction network) at each spatial sample x, and $\tau_f = 12$ ms was the time constant of each of the low-pass filters.

Layer 1. The layer 1 weights were the identity matrix and the output was linear, so that the layer 1 responses were driven to copy the retinal input. Layer 1 comprised a pair of neurons corresponding to each spatial location, all with the same temporal frequency tuning that matched that of the sinusoidal grating ($\omega_m = 8$ Hz). One neuron in each pair represented the real part of the complex-valued responses and the other neuron in each pair represented the imaginary part. For each such pair of neurons, the time-courses of the responses were offset by a 90° phase shift.

Layer 2. The layer 2 weights were constructed from even- and odd-phase spatial Gabor functions (Fig. 7B of the main text). Each of these 2 spatial weighting functions was convolved with the responses of each of the two spatial arrays of layer 1 responses to yield 4 space-time separable combinations. Direction-selective responses were computed as sums and differences of these space-time separable responses (90), resulting in 4 direction-selective channels, two of which were a quadrature pair that preferred leftward motion, and two of which were a quadrature pair that preferred rightward motion. The layer 2 output nonlinearity was squaring. Each of the 4 direction-selective channels was combined with 2 predictive basis functions: 0 Hz and 16 Hz (i.e., twice the temporal frequency in layer 1 because the output nonlinearity was quadratic).

The Gabor functions used for the spatial weights in layer 2 (Fig. 7B of the main text) were:

$$w_s(x) = \exp(x^2/\sigma^2)\sin(2\pi\omega_x x)$$
$$w_c(x) = \exp(x^2/\sigma^2)\cos(2\pi\omega_x x) \quad \textbf{[S23]}$$

$$w_{11}^{(1)}(x) = w_s(x) \qquad w_{12}^{(1)}(x) = w_c(x)$$
$$w_{21}^{(1)}(x) = w_c(x) \qquad w_{22}^{(1)}(x) = -w_s(x)$$
$$w_{31}^{(1)}(x) = -w_s(x) \qquad w_{32}^{(1)}(x) = w_c(x)$$
$$w_{41}^{(1)}(x) = w_c(x) \qquad w_{42}^{(1)}(x) = w_s(x)$$

where $\omega_x = 8$ cycle/deg was the preferred spatial frequency and $\sigma = 1/16$ degrees of visual angle determined the extent of the spatial weights.

Layer 3. There were two channels in layer 3. The feedforward drive for the first channel summed the quadrature pair of leftward-selective layer 2 responses, and summed across space. Likewise, the feedforward drive for the second channel summed the quadrature pair of rightward-selective layer 2 responses, and summed across space. The layer 3 output was again linear. Layer 3 had two predictive basis functions: 0 Hz and 1 Hz (i.e., matching the frequency of periodic motion in the stimulus).

The feedforward processing in this network, with no feedback and no prior (i.e., with λ=1), computed leftward and rightward "motion energy" (41, 90). It is called "motion energy" because it depends on the local (in space, time, orientation, spatial frequency, and temporal frequency) spectral energy of the stimulus. But the term "motion energy" has nothing to do with the energy function that is being minimized (Eq. **S15**). Layer 1 comprised a pair of neurons at each spatial location, with the same temporal frequency tuning. One of the neurons in each pair responded with a copy of the input (provided by the simplified temporal-filtering model of retinal processing). The other neuron responded with a phase-shifted copy of the input. The phase shift emerged because of the quadrature-phase (sine and cosine) temporal weights. As an aside, this solves a problem for models of visual motion perception, which rely on having pairs of neurons that respond with temporal phases offset by 90° (90). The feedforward drive in layer 2 depended on odd- and even-phase spatial weights, and a quadrature pair of temporal filters (the real and imaginary parts of the layer 1 responses), combined according to Eq. **S18**. This yielded four direction-selective channels: a quadrature pair selective for rightward motion and a quadrature pair selective for leftward motion (90). The feedforward drive in layer 3 computed motion energy, a sum of the squared responses of each quadrature pair.

The neural responses corresponding to the global minimum of E were computed for all neurons and all time steps (time step: $\Delta t = 10$ ms), using the "batch" algorithm (see above). There was a second local minimum for which the network predicted that the periodic motion would dissipate, with a clear local maximum separating the two local minima.

Bayesian cue combination (Fig. 5)

The energy function for the cue-combination network was:

$$E\left(\mathbf{y}^{(1)}\right)=\frac{1}{2}\alpha\lambda\sum_n\left(y_n^{(1)}-z_n^{(1)}\right)^2+\frac{1}{2}\alpha\left(1-\lambda\right)\sum_n\left(\frac{y_n^{(1)}}{\hat{g}}-\hat{y}_n^{(1)}\right)^2$$
$$\hat{g}=\sum_n y_n^{(1)}$$
$$z_n^{(1)}=w_1y_{n1}^{(0)}+w_2y_{n2}^{(0)} \qquad \textbf{[S24]}$$
$$w_1=\sqrt{\frac{\sigma_2^2}{2\left(\sigma_1^2+\sigma_2^2\right)}}$$
$$w_2=\sqrt{\frac{\sigma_1^2}{2\left(\sigma_1^2+\sigma_2^2\right)}},$$

where to $y_{n1}^{(0)}$ and to $y_{n2}^{(0)}$ are the responses of two sets of input neurons and $y_n^{(1)}$ are the responses of the output neurons (Fig. 5A of the main text). Each of the input neurons was tuned for depth, responding most strongly to a preferred depth value (Fig. 5B of the main text). Consequently, each input neuron was from a different channel, indexed by n. (A channel by the nomenclature I've adopted is a spatial array of neurons with identical stimulus selectivity, whereas each of the input neurons in this network responded preferentially to different depths at the same spatial location.) Both sets of input neurons had the same tuning curves, but responded to each of two different cues (e.g., stereo and motion parallax). The output neurons had the same tuning curves as the input neurons because the feedforward drive depended on a weighted sum of the responses of input neurons with identical tuning curves, with weights w_1 and w_2 (Eq. **S24**, 3rd line).

Each tuning curve, denoted $\psi_n(s)$, where s is stimulus depth, was one cycle of a raised cosine, and the spacing, amplitudes, and widths of the raised cosines were chosen so that the tuning curves summed to 1:

$$\psi_n\left(s\right)\propto\cos\left[\frac{2\pi\left(s-s_n\right)}{\nu}\right]+1 \qquad \textbf{[S25]}$$
$$\text{for } -\pi<\frac{2\pi\left(s-s_n\right)}{\nu}<\pi$$
$$\sum_j\psi_n\left(s\right)=1,$$

The value of ν determined the width of the tuning curves, and the values of s_n determined the preferred depths (the peaks of the tuning curves). The preferred depths were evenly spaced and the widths were selected to be even multiples of the spacing. The spacing and width also determined the amount of overlap, overlap = spacing / 2ν; the overlap was 4 for the simulation results in Fig. 5 of the main text.

The responses of the input neurons depended on the strengths of the two cues (g_1 and g_2), and the responses of the input neurons were presumed to be noisy (additive, independent, normally-distributed):

$$y_{n1}^{(0)}\sim N\left(g_1\psi_n\left(s\right),\sigma_1^2\right) \qquad \textbf{[S26]}$$
$$y_{n2}^{(0)}\sim N\left(g_2\psi_n\left(s\right),\sigma_2^2\right),$$

where σ_1 and σ_2 are the standard deviations of the noise. If $\sigma_1=\sigma_2$ then the two cues were equally reliable; otherwise not.

The responses of the output neurons were modeled as dynamical processes (Eq. **S6**) that minimized this energy function (Eq. **S24**) over time, subject to $y_n^{(1)}\geq 0$.

The prior for the response of the n^{th} output neuron was defined in terms of the tuning curves. The two example priors shown in Fig. 5C of the main text corresponded to $\hat{y}_n^{(1)}=\psi_n(0)$ and $\hat{y}_n^{(1)}=\psi_n(-0.5)+\psi_n(0.5)$. Each of these priors for the responses of the output neurons conferred a prior over stimuli:

$$p_0\left(s\right)\propto\exp\left[-\frac{1}{2\sigma_0^2}\sum_n\left(\psi_n(s)-\hat{y}_n^{(1)}\right)^2\right], \qquad \textbf{[S27]}$$

where σ_0 specified the reliability of the prior.

The readout was defined as:

$$h\left(s\mid\mathbf{y^{(1)}}\right)=\exp\left[-\sum_n h_n\left(s\mid y_n^{(1)}\right)\right] \qquad \textbf{[S28]}$$
$$h_n\left(s\mid y_n^{(1)}\right)=\frac{1}{2}\alpha\lambda\left(y_n^{(1)}-\hat{g}\psi_n(s)\right)^2+\frac{1}{2}\alpha\left(1-\lambda\right)\left(\frac{y_n^{(1)}}{\hat{g}}-\psi_n(s)\right)^2.$$

The readout h transformed the vector of responses of the output neurons to a continuous function of s that was approximately proportional to the Bayes-optimal posterior (as derived below). A variant of the readout computed a depth estimate and an uncertainty:

$$\hat{s}=\frac{\sum_k s_k h\left(s_k\mid\mathbf{y^{(1)}}\right)}{\sum_k h\left(s_k\mid\mathbf{y^{(1)}}\right)} \qquad \textbf{[S29]}$$
$$\sigma_{\hat{s}}^2=\frac{\sum_k\left(s_j-\hat{s}\right)^2 h\left(s_k\mid\mathbf{y^{(1)}}\right)}{\sum_k h\left(s_k\mid\mathbf{y^{(1)}}\right)},$$

where the depth estimate (or percept) $\hat{s}$ was approximately equal to the mean of the posterior, and the uncertainty $\sigma_{\hat{s}}$ was approximately equal to the standard deviation of the posterior. The value of k indexes a finite number of sam-

ples of s. Both variants of the readout (Eqs. **S28** and **S29**) depended only the responses of the output neurons $y_n^{(1)}$, the tuning curves $\psi_n(s)$, and the values of the state parameters λ and α. The Bayes-optimal posterior, on the other hand, depends on the responses of the input neurons $y_{n1}^{(0)}$ and $y_{n2}^{(0)}$, the noise standard deviations σ_1 and σ_2, the prior over stimuli $p_0(s)$, and the reliability of the prior σ_0.

Next I show that the readout is approximately proportional to the Bayes-optimal posterior, if the values of the state parameters are chosen correctly. There are two limiting cases corresponding to: 1) when the stimulus strengths of both cues are small, and 2) when the stimulus strengths of one or both cues are large. To begin, we need expressions for the probability distribution of z_n, and for the values of the state parameters.

The values of z_n were normally distributed because they were computed as weighted sums of normally-distributed random variables (Eq. **S24**, 3rd line):

$$z_n^{(1)} \sim N\left(g\psi_n(s), \sigma^2\right) \qquad \textbf{[S30]}$$

$$g = w_1 g_1 + w_2 g_2 \quad \text{(see Eq. S24)}$$

$$\sigma^2 = w_1^2\sigma_1^2 + w_2^2\sigma_2^2 = \frac{\sigma_1^2\sigma_2^2}{\sigma_1^2+\sigma_2^2} \quad \text{(see Eq. S24)}$$

The state parameters were chosen based on the reliabilities of each of the two cues and the reliability of the prior:

$$\alpha = r_0 + r_1 + r_2 \qquad \textbf{[S31]}$$

$$\lambda = \frac{r_1 + r_2}{r_0 + r_1 + r_2}$$

$$r_0 = \frac{1}{\sigma_0^2} \quad r_1 = \frac{1}{\sigma_1^2} \quad r_2 = \frac{1}{\sigma_2^2}$$

where r_1 and r_2 are the reliabilities of each of the two cues, and r_0 is the reliability of the prior. For these values of the state parameters:

$$\alpha\lambda = \frac{1}{\sigma^2} \qquad \textbf{[S32]}$$

$$\alpha(1-\lambda) = \frac{1}{\sigma_0^2}$$

It is helpful to rewrite the readout:

$$h\left(s \mid \mathbf{y^{(1)}}\right) = \exp\left[-\frac{1}{2}\alpha\lambda\sum_n\left(y_n^{(1)} - \hat{g}\psi_n(s)\right)^2 - \frac{1}{2}\alpha(1-\lambda)\sum_n\left(\frac{y_n^{(1)}}{\hat{g}} - \psi_n(s)\right)^2\right]$$

$$= \exp\left[-\frac{1}{2}\alpha\left(\lambda\hat{g}^2 + 1 - \lambda\right)\sum_j\left(\frac{y_n^{(1)}}{\hat{g}} - \psi_n(s)\right)^2\right] \qquad \textbf{[S33]}$$

Case 1: When the stimulus strengths of both cues are small, the readout is approximately proportional to the prior over s. If both cues are weak:

$$\hat{g}^2 \ll (1-\lambda) \qquad \textbf{[S34]}$$

$$\left(\lambda\hat{g}^2 + 1 - \lambda\right) = (1-\lambda)\left(\lambda\frac{\hat{g}^2}{1-\lambda} + 1\right) \approx (1-\lambda)$$

In addition, when both stimulus strengths are weak, then the second term of E (Eq. **S24**) dominates and the responses converge to values that are proportional to the priors, i.e.,

$$\frac{y_n^{(1)}}{\hat{g}} \approx \hat{y}_n^{(1)} \qquad \textbf{[S35]}$$

Consequently, the readout (Eq. **S33**) can be approximated:

$$h\left(s \mid \mathbf{y^{(1)}}\right) \approx \exp\left[-\frac{1}{2}\alpha(1-\lambda)\sum_n\left(\frac{y_n^{(1)}}{\hat{g}} - \psi_n(s)\right)^2\right] \qquad \textbf{[S36]}$$

(see Eqs. **S33** and **S34**)

$$\approx \exp\left[-\frac{1}{2\sigma_0^2}\sum_n\left(\hat{y}_n^{(1)} - \psi_n(s)\right)^2\right]$$

(see Eqs. **S32** and **S35**)

$$\propto p_0(s),$$

(see Eq. **S27**)

where $p_0(s)$ is the prior over s.

Case 2: When the stimulus strengths of one or both cues are large, the readout is approximately proportional to the likelihood. If one or both cues are strong:

$$\hat{g}^2 \gg (1-\lambda) \qquad \textbf{[S37]}$$

$$\left(\lambda\hat{g}^2 + 1 - \lambda\right) = \hat{g}^2\left(\lambda + \frac{1-\lambda}{\hat{g}^2}\right) \approx \left(\lambda\hat{g}^2\right)$$

In addition, when one or both cues are strong, then the first term of E (Eq. **S24**) dominates and the responses converge to minimize the feedforward drive, i.e.,

$$y_n^{(1)} \approx z_n^{(1)}, \qquad \textbf{[S38]}$$

And when one or both cues are strong, then $\hat{g} \approx g$:

$$\hat{g} = \sum_n y_n^{(1)} \approx \sum_n z_n^{(1)} \qquad \textbf{[S39]}$$

$$= \left(w_1\sum_n y_{n1}^{(0)} + w_2\sum_n y_{n2}^{(0)}\right) \quad \text{(see Eq. S24)}$$

$$\approx \left(w_1 g_1\sum_n \psi_n(s) + w_2 g_2\sum_n \psi_n(s)\right)$$

(see Eq. **S26**)

$$= \left(w_1 g_1 + w_2 g_2 \right) = g \qquad \text{(see Eqs. S25 and S30)}$$

Consequently, the readout (Eq. **S33**) can be approximated:

[S40]

$$h\left(s \mid \mathbf{y}^{(1)}\right) \approx \exp\left[-\frac{1}{2} \alpha \left(\lambda \hat{g}^2 \right) \sum_j \left(\frac{y_j^{(1)}}{\hat{g}} - \psi_j(s) \right)^2 \right]$$

(see Eqs. **S33** and **S37**)

$$= \exp\left[-\frac{1}{2} \alpha \lambda \sum_j \left(y_j^{(1)} - \hat{g} \psi_j(s) \right)^2 \right]$$

$$\approx \exp\left[-\frac{1}{2\sigma^2} \sum_j \left(z_j^{(1)} - g \psi_j(s) \right)^2 \right]$$

(see Eqs. **S32**, **S38**, **S39**)

$$\propto p\left(\mathbf{z}^{(1)} \mid s\right)$$

(see Eq. **S30**)

Finally, when one or both cues are strong, then $p(\mathbf{z}^{(1)} \mid s)$ is approximately proportional to the likelihood $p(\mathbf{y}_1^{(0)}, \mathbf{y}_2^{(0)} \mid s)$. The negative log likelihoods are:

[S41]

$$-\log\left[p\left(\mathbf{y}_1^{(0)}, \mathbf{y}_2^{(0)} \mid s \right) \right] = \sum_n \left[\frac{\left(y_{n1}^{(0)} - g_1 \psi_n(s) \right)^2}{2\sigma_1^2} + \frac{\left(y_{n2}^{(0)} - g_2 \psi_n(s) \right)^2}{2\sigma_2^2} \right] - \log(c_0)$$

$$-\log\left[p\left(\mathbf{z}^{(1)} \mid s \right) \right] = \sum_n \frac{\left(z_n^{(1)} - g \psi_n(s) \right)^2}{2\sigma^2} - \log(c_1)$$

,

where c_0 and c_1 are proportionality constants. It suffices to show that each of the terms in the summations are approximately equal to one another:

[S42]

$$\frac{\left(z_n^{(1)} - g\psi_n(s) \right)^2}{2\sigma^2} = \frac{\sigma_1^2 + \sigma_2^2}{2\sigma_1^2 \sigma_2^2} \left(w_1 y_{n1}^{(0)} + w_2 y_{n2}^{(0)} - w_1 g_1 \psi_n(s) - w_2 g_2 \psi_n(s) \right)^2$$

$$= \frac{\left(y_{n1}^{(0)} - g_1 \psi_n(s) \right)^2}{2\sigma_1^2} + \frac{\left(y_{n2}^{(0)} - g_2 \psi_n(s) \right)^2}{2\sigma_2^2} + \frac{\left(y_{n1}^{(0)} - g_1 \psi_n(s) \right)\left(y_{n2}^{(0)} - g_2 \psi_n(s) \right)}{\sigma_1 \sigma_2}$$

$$\approx \frac{\left(y_{n1}^{(0)} - g_1 \psi_n(s) \right)^2}{2\sigma_1^2} + \frac{\left(y_{n2}^{(0)} - g_2 \psi_n(s) \right)^2}{2\sigma_2^2} .$$

The last step relies on an approximation that the cross-term can be ignored. This approximation is reasonable when the stimulus strengths of one or both cues are large, specifically when either: $g_1/\sigma_1 \gg g_2/\sigma_2$, or $g_2/\sigma_2 \gg g_1/\sigma_1$, or both $g_1/\sigma_1 \gg 1$, and $g_2/\sigma_2 \gg 1$.

1. Lecun Y, *et al.* (1989) Backpropagation applied to handwritten zip code recognition. *Neural Comput* 1:541-551.
2. Ackley DH, Hinton GE, & Sejnowski TJ (1985) A learning algorithm for Boltzmann Machines. *Cognitive Science* 9(1):147-169.
3. Lewicki MS & Olshausen BA (1999) Probabilistic framework for the adaptation and comparison of image codes. *J Opt Soc Am A* 16(7):1587-1601.
4. Cadieu CF & Olshausen BA (2012) Learning intermediate-level representations of form and motion from natural movies. *Neural Comput* 24(4):827-866.
5. Mumford D (1992) On the computational architecture of the neocortex. II. The role of cortico-cortical loops. *Biol Cybern* 66(3):241-251.
6. Rumelhart DE, McClelland JL, & Group PR (1988) *Parallel distributed processing* (IEEE).
7. Hinton GE (2007) Learning multiple layers of representation. *Trends Cogn Sci* 11(10):428-434.
8. Freeman J & Simoncelli EP (2011) Metamers of the ventral stream. *Nat Neurosci* 14(9):1195-1201.
9. Friston K (2005) A theory of cortical responses. *Philos Trans R Soc Lond B Biol Sci* 360(1456):815-836.
10. Bastos AM, *et al.* (2012) Canonical microcircuits for predictive coding. *Neuron* 76(4):695-711.
11. Pineda FJ (1987) Generalization of back-propagation to recurrent neural networks. *Phys Rev Lett* 59(19):2229-2232.
12. Carpenter GA & Grossberg S (1987) A Massively Parallel Architecture for a Self-Organizing Neural Pattern-Recognition Machine. *Comput Vision Graph* 37(1):54-115.
13. Bruna J & Mallat S (2013) Invariant scattering convolution networks. *IEEE transactions on pattern analysis and machine intelligence* 35(8):1872-1886.
14. Girshick AR, Landy MS, & Simoncelli EP (2011) Cardinal rules: visual orientation perception reflects knowledge of environmental statistics. *Nat Neurosci* 14(7):926-932.
15. Knill DC & Richards W (1996) *Perception as Bayesian inference* (Cambridge University Press).
16. Weiss Y, Simoncelli EP, & Adelson EH (2002) Motion illusions as optimal percepts. *Nat Neurosci* 5(6):598-604.
17. Mamassian P, Landy MS, & Maloney LT (2002) Bayesian modelling of visual perception. *Probabilistic models of the brain: Perception and neural function*:13-36.
18. Yuille A & Kersten D (2006) Vision as Bayesian inference: analysis by synthesis? *Trends Cogn Sci* 10(7):301-308.
19. Brainard DH & Freeman WT (1997) Bayesian color constancy. *J Opt Soc Am A* 14(7):1393-1411.
20. Laing CR & Chow CC (2002) A spiking neuron model for binocular rivalry. *Journal of computational neuroscience* 12(1):39-53.
21. Moreno-Bote R, Rinzel J, & Rubin N (2007) Noise-induced alternations in an attractor network model of perceptual bistability. *J Neurophysiol* 98(3):1125-1139.
22. Wilson HR (2003) Computational evidence for a rivalry hierarchy in vision. *Proc Natl Acad Sci U S A* 100(24):14499-14503.
23. Wilson HR (2007) Minimal physiological conditions for binocular rivalry and rivalry memory. *Vision Res* 47(21):2741-2750.
24. Jordan MI & Rumelhart DE (1992) Forward models: Supervised learning with a distal teacher. *Cognitive Science* 16(3):307-354.
25. Riesenhuber M & Poggio T (2002) Neural mechanisms of object recognition. *Curr Opin Neurobiol* 12(2):162-168.
26. Fleet DJ, Wagner H, & Heeger DJ (1996) Neural encoding of binocular disparity: energy models, position shifts and phase shifts. *Vision Res* 36(12):1839-1857.
27. Landy MS & Bergen JR (1991) Texture segregation and orientation gradient. *Vision Res* 31(4):679-691.
28. Zipser D, Kehoe B, Littlewort G, & Fuster J (1993) A spiking network model of short-term active memory. *J Neurosci* 13(8):3406-3420.
29. Seung HS (1996) How the brain keeps the eyes still. *Proc Natl Acad Sci U S A* 93(23):13339-13344.
30. Zhang K (1996) Representation of spatial orientation by the intrinsic dynamics of the head-direction cell ensemble: a theory. *J Neurosci* 16(6):2112-2126.

31. Robinson DA (1989) Integrating with neurons. *Annual review of neuroscience* 12:33-45.
32. Major G & Tank D (2004) Persistent neural activity: prevalence and mechanisms. *Curr Opin Neurobiol* 14(6):675-684.
33. Li Z (1998) A neural model of contour integration in the primary visual cortex. *Neural Comput* 10(4):903-940.
34. Li Z (1999) Visual segmentation by contextual influences via intracortical interactions in the primary visual cortex. *Network* 10(2):187-212.
35. Li Z (1999) Contextual influences in V1 as a basis for pop out and asymmetry in visual search. *Proc Natl Acad Sci U S A* 96(18):10530-10535.
36. Coen-Cagli R, Dayan P, & Schwartz O (2012) Cortical Surround Interactions and Perceptual Salience via Natural Scene Statistics. *PLoS Comput Biol* 8(3):e1002405.
37. Freeman WT & Adelson EH (1991) The design and use of steerable filters. *IEEE Pattern Analysis and Machine Intelligence* 13:891-906.
38. Simoncelli EP, Freeman WT, Adelson EH, & Heeger DJ (1992) Shiftable multi-scale transforms. *IEEE Transactions on Information Theory, Special Issue on Wavelets* 38:587-607.
39. Simoncelli EP (1993) Distributed representation and analysis of visual motion. (Massachusetts Institute of Technology, Cambridge, MA).
40. Carandini M & Heeger DJ (2012) Normalization as a canonical neural computation. *Nature reviews. Neuroscience* 13(1):51-62.
41. Heeger DJ (1992) Half-squaring in responses of cat striate cells. *Vis Neurosci* 9(5):427-443.
42. Heeger DJ (1992) Normalization of cell responses in cat striate cortex. *Vis Neurosci* 9(2):181-197.
43. Zoccolan D, Cox DD, & DiCarlo JJ (2005) Multiple object response normalization in monkey inferotemporal cortex. *J Neurosci* 25(36):8150-8164.
44. Rabinowitz NC, Willmore BD, Schnupp JW, & King AJ (2011) Contrast gain control in auditory cortex. *Neuron* 70(6):1178-1191.
45. Ohshiro T, Angelaki DE, & Deangelis GC (2011) A normalization model of multisensory integration. *Nature neuroscience* 14(6):775-782.
46. Louie K & Glimcher PW (2010) Separating value from choice: delay discounting activity in the lateral intraparietal area. *J Neurosci* 30(16):5498-5507.
47. Vokoun CR, Huang X, Jackson MB, & Basso MA (2014) Response normalization in the superficial layers of the superior colliculus as a possible mechanism for saccadic averaging. *J Neurosci* 34(23):7976-7987.
48. Olsen SR, Bhandawat V, & Wilson RI (2010) Divisive normalization in olfactory population codes. *Neuron* 66(2):287-299.
49. Herrmann K, Montaser-Kouhsari L, Carrasco M, & Heeger DJ (2010) When size matters: attention affects performance by contrast or response gain. *Nat Neurosci* 13(12):1554-1559.
50. Itthipuripat S, Garcia JO, Rungratsameetaweemana N, Sprague TC, & Serences JT (2014) Changing the spatial scope of attention alters patterns of neural gain in human cortex. *J Neurosci* 34(1):112-123.
51. Reynolds JH & Heeger DJ (2009) The normalization model of attention. *Neuron* 61(2):168-185.
52. Sundberg KA, Mitchell JF, & Reynolds JH (2009) Spatial attention modulates center-surround interactions in macaque visual area v4. *Neuron* 61(6):952-963.
53. Brouwer GJ & Heeger DJ (2011) Cross-orientation suppression in human visual cortex. *Journal of neurophysiology* 106(5):2108-2119.
54. Brouwer GJ, Arnedo V, Offen S, Heeger DJ, & Grant AC (2015) Normalization in human somatosensory cortex. *J Neurophysiol* 114(5):2588-2599.
55. Carandini M, Heeger DJ, & Movshon JA (1997) Linearity and normalization in simple cells of the macaque primary visual cortex. *J Neurosci* 17(21):8621-8644.
56. Cavanaugh JR, Bair W, & Movshon JA (2002) Nature and interaction of signals from the receptive field center and surround in macaque V1 neurons. *J Neurophysiol* 88(5):2530-2546.
57. Cavanaugh JR, Bair W, & Movshon JA (2002) Selectivity and spatial distribution of signals from the receptive field surround in macaque V1 neurons. *J Neurophysiol* 88(5):2547-2556.
58. Smith MA, Bair W, & Movshon JA (2006) Dynamics of suppression in macaque primary visual cortex. *J Neurosci* 26(18):4826-4834.
59. Zenger-Landolt B & Heeger DJ (2003) Response suppression in V1 agrees with psychophysics of surround masking. *J Neurosci* 23(17):6884-6893.
60. Simoncelli EP & Heeger DJ (1998) A model of neuronal responses in visual area MT. *Vision Res* 38(5):743-761.
61. Foley JM (1994) Human luminance pattern-vision mechanisms: masking experiments require a new model. *J Opt Soc Am A* 11(6):1710-1719.
62. Xing J & Heeger DJ (2000) Center-surround interactions in foveal and peripheral vision. *Vision Res* 40(22):3065-3072.
63. Xing J & Heeger DJ (2001) Measurement and modeling of center-surround suppression and enhancement. *Vision Res* 41(5):571-583.
64. Carandini M & Ferster D (2000) Membrane potential and firing rate in cat primary visual cortex. *J Neurosci* 20(1):470-484.
65. Carandini M (2007) Melting the iceberg: contrast invariance in visual cortex. *Neuron* 54(1):11-13.
66. Carandini M & Heeger DJ (1994) Summation and division by neurons in primate visual cortex. *Science* 264(5163):1333-1336.
67. Heeger DJ (1993) Modeling simple-cell direction selectivity with normalized, half- squared, linear operators. *J Neurophysiol* 70(5):1885-1898.
68. Rubin DB, Van Hooser SD, & Miller KD (2015) The stabilized supralinear network: a unifying circuit motif underlying multi-input integration in sensory cortex. *Neuron* 85(2):402-417.
69. Olsen SR & Wilson RI (2008) Lateral presynaptic inhibition mediates gain control in an olfactory circuit. *Nature* 452(7190):956-960.
70. Katzner S, Busse L, & Carandini M (2011) GABAA inhibition controls response gain in visual cortex. *J Neurosci* 31(16):5931-5941.
71. Sato TK, Haider B, Hausser M, & Carandini M (2016) An excitatory basis for divisive normalization in visual cortex. *Nat Neurosci*.
72. Nowlan SJ & Sejnowski TJ (1995) A selection model for motion processing in are MT of primates. *Journal of Neuroscience* 15:1195-1214.
73. Kouh M & Poggio T (2008) A canonical neural circuit for cortical nonlinear operations. *Neural Comput* 20(6):1427-1451.
74. Ganguli D & Simoncelli EP (2014) Efficient sensory encoding and Bayesian inference with heterogeneous neural populations. *Neural Comput* 26(10):2103-2134.
75. Sarter M & Bruno JP (1997) Cognitive functions of cortical acetylcholine: toward a unifying hypothesis. *Brain Res Brain Res Rev* 23(1-2):28-46.
76. Yu AJ & Dayan P (2005) Uncertainty, neuromodulation, and attention. *Neuron* 46(4):681-692.
77. Marshall L*, et al.* (2016) Pharmacological Fingerprints of Contextual Uncertainty. *PLoS Biol* 14(11):e1002575.

78. Pafundo DE, Nicholas MA, Zhang R, & Kuhlman SJ (2016) Top-Down-Mediated Facilitation in the Visual Cortex Is Gated by Subcortical Neuromodulation. *J Neurosci* 36(10):2904-2914.

79. Schultz W, Dayan P, & Montague PR (1997) A neural substrate of prediction and reward. *Science* 275(5306):1593-1599.

80. McGinley MJ, David SV, & McCormick DA (2015) Cortical Membrane Potential Signature of Optimal States for Sensory Signal Detection. *Neuron* 87(1):179-192.

81. Pisauro MA, Dhruv NT, Carandini M, & Benucci A (2013) Fast hemodynamic responses in the visual cortex of the awake mouse. *J Neurosci* 33(46):18343-18351.

82. Einhauser W, Stout J, Koch C, & Carter O (2008) Pupil dilation reflects perceptual selection and predicts subsequent stability in perceptual rivalry. *Proc Natl Acad Sci U S A* 105(5):1704-1709.

83. Aston-Jones G & Cohen JD (2005) An integrative theory of locus coeruleus-norepinephrine function: adaptive gain and optimal performance. *Annual review of neuroscience* 28:403-450.

84. Teigen KH (1994) Yerkes-Dodson: A law for all seasons. *Theory & Psychology* 4(4):525-547.

85. Yerkes RM & Dodson JD (1908) The Relation of Strength of Stimulus to Rapidity of Habit Formation. *Journal of Comparative Neurology & Psychology*:459-482.

86. Gilden DL (2001) Cognitive emissions of 1/f noise. *Psychol Rev* 108(1):33-56.

87. Gilden DL, Thornton T, & Mallon MW (1995) 1/f noise in human cognition. *Science* 267(5205):1837-1839.

88. Rigoli LM, Holman D, Spivey MJ, & Kello CT (2014) Spectral convergence in tapping and physiological fluctuations: coupling and independence of 1/f noise in the central and autonomic nervous systems. *Front Hum Neurosci* 8:713.

89. VanRullen R (2016) Perceptual Cycles. *Trends Cogn Sci* 20(10):723-735.

90. Adelson EH & Bergen JR (1985) Spatiotemporal energy models for the perception of motion. *Journal of the Optical Society of America A* 2:284-299.