What computations can be done with traveling waves in visual cortex?

Gabriel Benigno
Western University
Roberto Budzinski
Western University
Zachary Davis
Salk Institute for Biological Studies https://orcid.org/0000-0003-4440-9011
John Reynolds
The Salk Institute for Biological Studies https://orcid.org/0000-0001-6988-4607
Lyle Muller (lmuller2@uwo.ca)
Western University https://orcid.org/0000-0001-5165-9890

Article

Keywords:

Posted Date: August 29th, 2022

DOI: https://doi.org/10.21203/rs.3.rs-1903144/v1

What computations can be done with traveling waves in visual cortex?

Gabriel B. Benigno[1,2,3], Roberto C. Budzinski[1,2,3], Zachary Davis[4], John Reynolds[4], Lyle Muller[1,2,3,*]

Affiliations: [1]Department of Mathematics, Western University, London, ON, Canada, [2]Brain and Mind Institute, Western University, London, ON, Canada, [3]Western Academy for Advanced Research, Western University, London, ON, Canada, [4]The Salk Institute for Biological Studies, La Jolla, CA, USA, *Corresponding author: lmuller2@uwo.ca.

Recent analyses have found waves of neural activity traveling across entire visual cortical areas in awake animals. These traveling waves modulate excitability of local networks and perceptual sensitivity. The general computational role for these spatiotemporal patterns in the visual system, however, remains unclear. Here, we hypothesize that traveling waves endow the brain with the capacity to predict complex and naturalistic visual inputs. We present a new network model whose connections can be rapidly and efficiently trained to predict natural movies. After training, a few input frames from a movie trigger complex wave patterns that drive accurate predictions many frames into the future, solely from the network's connections. When the recurrent connections that drive waves are randomly shuffled, both traveling waves and the ability to predict are eliminated. These results show traveling waves could play an essential computational role in the visual system by embedding continuous spatiotemporal structures over spatial maps.

Introduction

Five percent of synapses received by a neuron in visual cortex arrive through the feedforward (FF) pathway that conveys sensory input from the eyes[1]. While these FF synapses are strong[2], "horizontal" recurrent connections coming from within the cortical region make up about 80% of total synaptic inputs, with 95% of these connections arising from a very local patch (2 mm) around the cell[1]. The anatomy of the visual system thus indicates that cortical neurons interact with other neurons across the retinotopically organized maps[3] that assign nearby points in visual space to nearby points in a cortical region, via these horizontal connections. Models of the visual system predominantly focus only on FF[4,5] and feedback (FB)[6] connections. One result of this focus is that neurons in visual cortex are often conceived to be non-interacting "feature

detectors", with fixed selectivity to features in visual input (driven by FF connections) that can be modulated by expectations generated in higher visual areas (driven by FB connections). Neuroscientists have long been interested in how horizontal connections shape neuronal selectivity[7,8] and "non-classical" receptive fields[9–13]. More recently, neuroscientists have also been interested in adding these connections to deep learning models to understand neuronal selectivity in visual cortex[14,15]. It remains unclear, however, how horizontal connections shape the moment-by-moment dynamics and computations in cortex while processing visual input.

Recent analyses of large-scale recordings have revealed that horizontal connections profoundly shape spatiotemporal dynamics in cortex. Traveling waves driven by horizontal connections have been observed in visual cortex of anesthetized animals[16–21]. The relevance of traveling waves had previously been called into question, as they were thought to disappear in the awake state[22] or to be suppressed by high-contrast visual stimuli[19,23]. Recent analyses of neural activity at the single-trial level, however, have revealed spontaneous[24] and stimulus-evoked[25] activity patterns that travel smoothly across entire cortical regions in awake, behaving primates during normal vision. These neural traveling waves (nTWs) shift the balance of excitation and inhibition as they propagate across cortex, sparsely modulating spiking activity as they pass[26]. Because they drive fluctuations in neural excitability[24,27], nTWs show that neurons at one point in a visual area (representing a small section of visual space) can strongly interact with neurons across the entire cortical region. These results thus indicate that cortical neurons may share information about visual scenes broadly across the retinotopic map, through nTWs generated by horizontal connections.

What computations, then, can be done with waves of neural activity traveling across a map of visual space? To address this question, we studied a complex-valued neural network (cv-NN) processing visual inputs ranging from simple stimuli to natural movies. cv-NNs exhibit similar or superior performance to standard, real-valued neural networks in many supervised learning tasks[28], and have been used effectively in explaining biological neural dynamics[29]. Here, we modified the standard FF architecture used in deep learning and computer vision to include horizontal recurrent connections, where neurons in a single processing layer form a web of interconnections similar to the horizontal connections in visual cortex. Horizontal recurrent connections are thought to provide advantages[14] over the standard FF architecture used in computer vision tasks[5,30]; however, current methods for training recurrent models severely limit both the time window over which recurrent activity can be considered and the ease with which

the networks can be trained[31,32]. In recent work, we have introduced a mathematical approach to understand the recurrent dynamics in a specific complex-valued model[33]. Here, we leverage this understanding to train recurrent complex-valued networks to process visual inputs, ranging from simple stimuli to naturalistic movie scenes. The training process for the cv-NN is rapid and efficient, requiring only minutes of desktop computer time. The resulting networks can predict learned movies many frames into the future, entirely from their internal dynamics alone, without external input. During prediction, the recurrent network exhibits prominent nTWs, ranging from simple waves propagating out from a small local input[25] to complex traveling wave patterns[34], raising the possibility that nTWs enable processing spatiotemporally complex, natural, and dynamic visual scenes.

Results

The cv-NN consists of an input layer sending movie frames to a recurrently connected network of model neurons. An individual movie frame, serving as input to the network, is represented by a two-dimensional grid of pixels (input frame, Fig. 1a), and each pixel projects to the recurrently connected layer through FF connections (red lines, Fig. 1a). The recurrently connected layer is arranged on a two-dimensional grid, analogous to the retinotopic arrangement of neurons in visual regions. Horizontal interconnections within the cv-NN then drive recurrent interactions in the network (blue lines, Fig. 1a). Both FF and horizontal recurrent projections in the cv-NN are matched to the approximate scale of connectivity in visual cortex[35,36], so that a single pixel in an input movie drives a local patch of neurons, with overlapping horizontal connections, in the cv-NN. Lastly, neurons in the recurrent layer communicate with time delays approximating axonal conduction speeds along horizontal fibers[37], which have recently been shown to shape spiking neural activity into nTWs[26]. The combination of FF input and dense interconnections generates complex patterns of activity in the recurrent layer (Fig. 1b). Here, we focus on these recurrent activity patterns to understand their computational role for movie inputs ranging from simple to complex.

nTWs can simultaneously encode stimulus position and time of onset over spatial maps

To understand how nTWs propagating over sensory maps could facilitate visual computation, we first studied the dynamics generated in response to a single point stimulus. Without recurrent connections, a short point stimulus generates a small bump of activity that remains centered on

the point of input (“without recurrence”, Fig. 1c). With recurrent connections, however, the point stimulus generates a wave that propagates out from the point of input (“with recurrence”, Fig. 1c). We then studied these stimulus-evoked waves, which are qualitatively similar to those previously observed in visual cortex of awake primate[25], in a simple decoding task. Specifically, we let the point stimulus appear at a random time in a series of input frames, and then trained a linear classifier to decode the time of stimulus onset from the network activity at the final frame. As expected, without recurrent connections, the classifier performed at chance-level accuracy in this task (Fig. 1c, right; Methods - Time-of-stimulus prediction task). With recurrence, however, the classifier selects the correct time of stimulus appearance from the final network state with 100% accuracy. This initial example shows that traveling waves of neural activity, when propagating on an orderly retinotopic map, can allow decoding of stimulus onset time, in addition to stimulus location, even after the stimulus is no longer present.

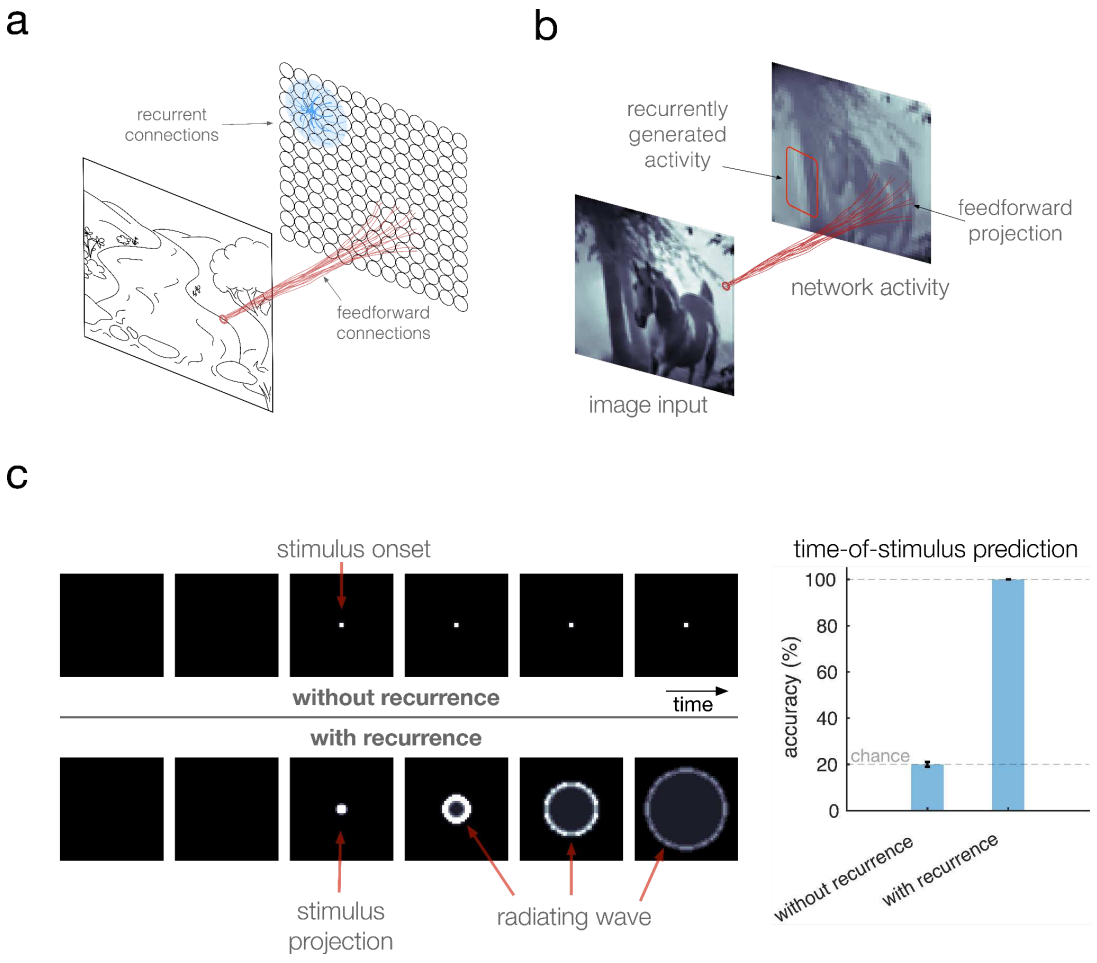

Figure 1: A topographic recurrent network model encodes temporal information of video frames via internal wave activity. (a) Schematic of the network model. Neurons (circles) are arranged on a two-dimensional grid and are

recurrently connected (blue) locally in space like the cortical sheet. A natural image input projects locally into the network via feedforward connections (red), mimicking retinotopy. **(b)** Example dynamic of the network model. Due to the spatially local projection of the input image, an imprint of the image is visible in the grid of network activity. Due to the local recurrent connectivity, intrinsic wave activity is generated alongside the input projection. **(c)** In a sequence of six frames, exactly one of the first five contains a point stimulus, and the other frames do not. These frames are sequentially input to the network. Top row: When the network has no recurrence, the stimulus projection remains stationary. Bottom row: With recurrence, from the time of stimulus, the network activity contains a projection of the stimulus and a wave radiating outward. Right: A linear classifier that received the final network state in the no-recurrence case could not predict the time of stimulus beyond chance-level accuracy. In contrast, using the classifier with the sixth with-recurrence network state allowed 100% accuracy since the feedforward projection of the point stimulus triggered a radiating wave that encoded the time of stimulus in the subsequent network states.

nTWs aid forecasting movie inputs from simple to complex

Can nTWs enable the processing of the complex, dynamic, and non-stationary visual scenes that we encounter in our natural experience? We approached this question in several steps. We first asked whether, given an input frame from a movie, the cv-NN could be trained to accurately predict the following frame. To perform this more complicated task, we introduced a learning rule that requires training only a linear readout of the recurrent layer (Fig. 2a). This procedure is analogous to a complex-valued implementation of the reservoir computing paradigm[38] that has recently proven very powerful for learning the dynamics of chaotic systems in physics[39]. This training process, however, has never before been applied to naturalistic movie scenes. We find the cv-NN can be rapidly, reliably, and efficiently trained to predict the next frame in a movie input (Supplementary Materials - Section III, Table S2, Moving Bump Input). Surprisingly, with a cv-NN trained on a movie input, the predicted next frame generated by the network can then be provided as input, in place of the original movie (Fig. 2b). We call this process *closed-loop forecasting* of entire visual scenes.

The visual cortex readily processes and operates on dynamic visual inputs on timescales of milliseconds to seconds. We then asked whether closed-loop forecasting in this system could work on the scale of tens to hundreds of frames in an input movie. Starting with the first half of a movie containing a simple moving bump stimulus tracing out a trajectory in two-dimensional space (Fig. 2c), we find that the trained cv-NN can produce the entire second half of the movie as output from its trained synaptic weights alone (Fig. 2d and Movie S1). As in the previous example, activity in the recurrent layer exhibits a dynamic spatiotemporal pattern extending beyond the immediate FF imprint of the stimulus and structured by the recurrent connections in the network (Fig. 2e and Movie S1). These results demonstrate that recurrent cv-NNs can produce simple video inputs from their recurrent connections through this rapid and efficient

training process. Finally, when we remove the recurrent connections, the cv-NN produces an activity pattern that represents only the average of FF stimulus imprints, without having learned the underlying spatiotemporal process[40]. In this case, the cv-NN no longer produces an accurate closed-loop forecast (Fig. 2f). These results demonstrate the importance of both the spatiotemporal patterns in the reservoir and the horizontal recurrent dynamics generating them.

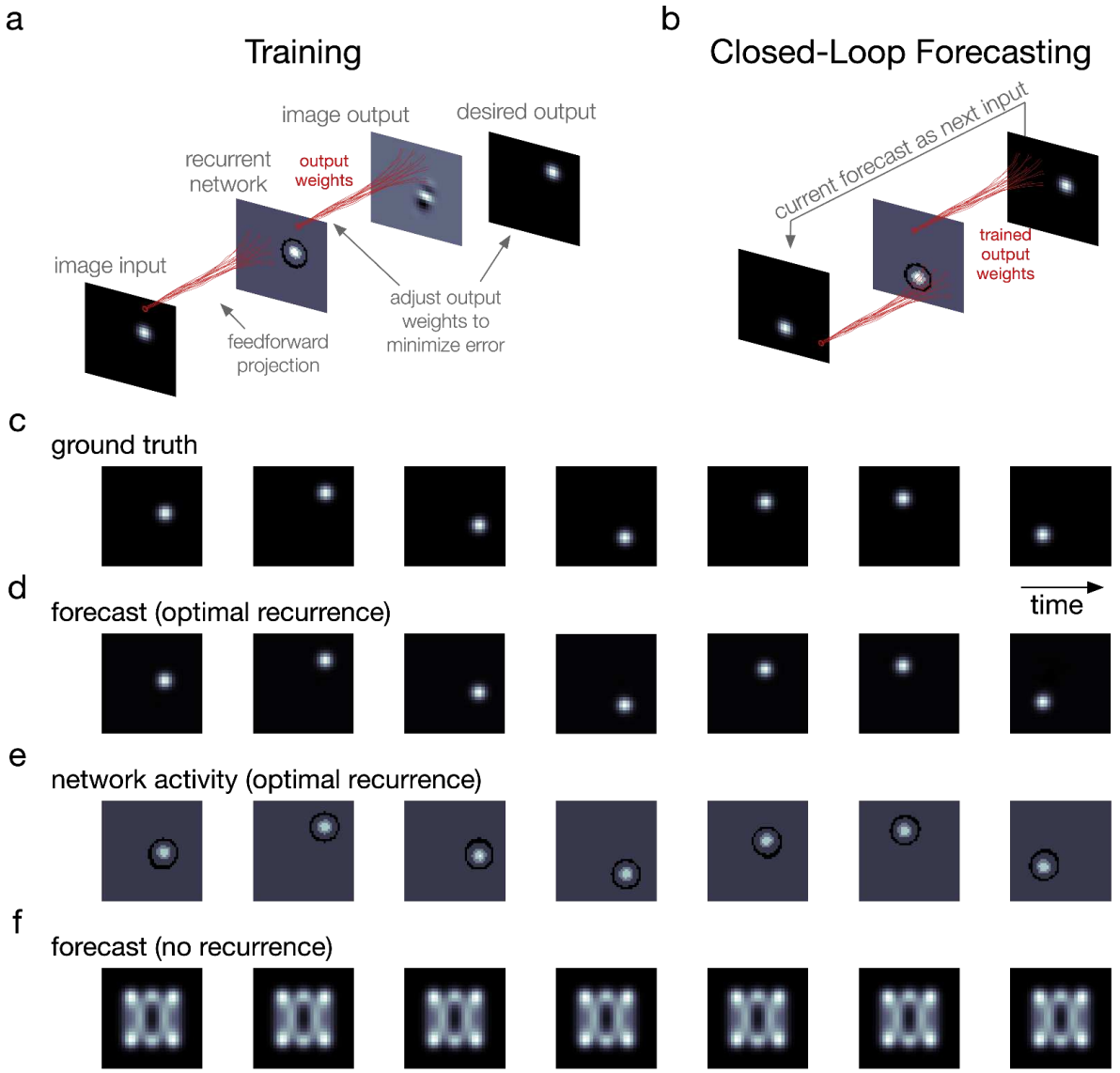

Figure 2: The network can forecast a simple video input many frames into the future. (a) As in the classification example (Figure 1), a video frame projects into the network in a spatially local manner and a recurrent network interaction occurs, generating internal wave activity on top of the projection. The network outputs an image from its network state via a matrix of trainable weights. Training entails one-shot linear regression between a set of network states and the corresponding desired output frames (the one-step-ahead next frames). Shown: a schematic representation of the one-shot linear regression for one time step. **(b)** Once training of the readout weights is complete, closed-loop forecasting begins. To properly test how well the network model learned the underlying spatiotemporal process from the training data, it is deprived of ground-truth data of any kind during this step. Instead,

the forecast next frame at one time step serves as the input frame for the following time step. **(c)** Video frames of the data: a bump tracing an orbit. **(d)** Corresponding closed-loop forecasts generated by the network model with optimal recurrence. **(e)** Network activity for the optimal-recurrence case. Cosine of phase of activation shown. **(f)** Closed-loop forecast in the case without recurrence.

We find that closed-loop forecast performance in this system depends on two key factors: (1) the ratio of horizontal recurrent strength to feedforward input strength and (2) the spatial extent of the recurrence. To study the first factor in detail, we measured closed-loop forecast performance using an index of structural similarity (SSIM)[41], which quantifies the perceptual match between two images. We studied SSIM between movie frames produced by the closed-loop forecast process and the ground truth at different ratios of recurrence to input (Fig. 3a; see also Fig. S1 and Methods - Network connectivity and Network dynamics). Once the stimulus is removed and the closed-loop forecast begins (video frame 1, Fig. 3a), forecast performance in cv-NNs with low recurrent strength quickly drops close to zero (light blue line, Fig. 3a). By contrast, cv-NNs at optimal recurrent strength sustain closed-loop forecasts for long timescales (gray line, Fig. 3a), extending beyond 100 video frames into the future. Importantly, networks where recurrence is too strong also perform poorly, with SSIM dropping near zero within a short timeframe (copper line, Fig. 3a). Systematic quantification of SSIM across ratios of recurrent strength to input strength reveals that performance is best when the recurrence and input are approximately balanced (Fig. 3b), highlighting the importance of the interplay between these two fundamental circuit patterns in visual cortex. We next studied performance as a function of the spatial extent of recurrent connectivity. The best performance occurs for recurrent lengths on approximately the same spatial scale as the moving bump stimulus (Fig. 3c), with performance dropping for recurrent lengths outside this range. This result demonstrates that recurrent connections, which span from local to long-range in visual cortex[6,42], utilize features in the closed-loop forecasting task best when matched to the spatial scale of the input.

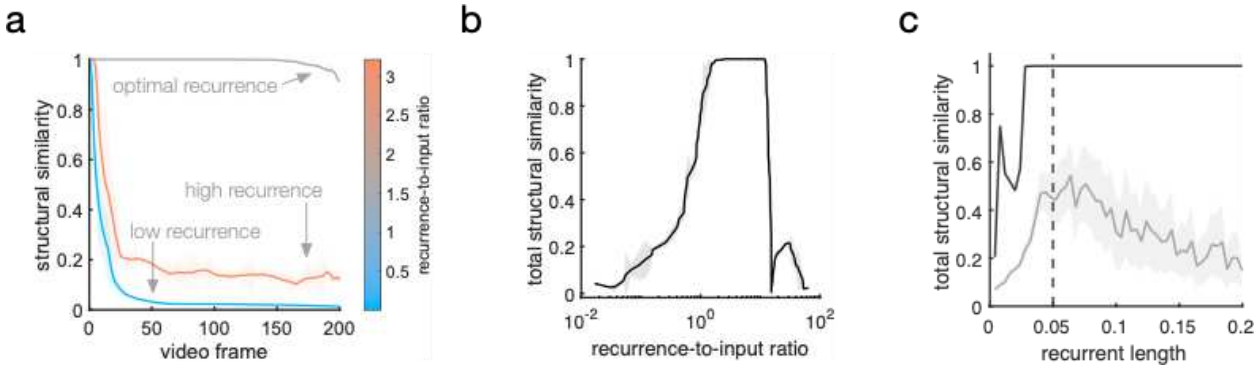

Figure 3: Moving bump forecast performance depends on specific properties of the recurrent connections. **(a)** Structural similarity between a forecast frame and the ground truth as a function of closed-loop forecast video

frame. Each curve corresponds to a different network parameter implementation. Curves have been smoothed by a moving-average filter (filter width of 30 time steps). Shaded error is the absolute difference between filtered and unfiltered. **(b)** Total structural similarity, in which a single SSIM is calculated for the whole movie, as a function of the recurrence-to-input ratio. In the parameter space, each point differs only in recurrent strength. Smoothing and error shading is the same as in **a**. **(c)** Total structural similarity as a function of recurrent length. In the three-dimensional parameter space comprising the recurrent strength (rs), recurrent length (rl), and input strength (is), averages across rs-is planes at fixed rl were computed (gray curve). The peak coincides with the standard-deviation width of the Gaussian bump stimulus (dashed vertical line). Shaded area: variance. Solid black curve: maximum structural similarity at each recurrent length.

The visual system readily processes richly textured and naturalistic visual scenes. To examine this type of stimulus in the cv-NN, we considered naturalistic video inputs for next-frame prediction and closed-loop forecasting. To do this, we used videos from the Weizmann Human Action Dataset[43]. As above, we trained linear readout weights of the cv-NN on these individual naturalistic movie inputs (Fig. 4a) and then tested whether, given the first half of the input movie, the network could produce the second half in a closed-loop forecast (Fig. 4b). Even with a much more sophisticated input than the previous examples, the cv-NN can be trained rapidly and efficiently on the natural movie inputs (Supplementary Materials - Section III, Table S2, Walking Person Input). As in previous examples, at optimal values of the network parameters (Methods - Parameter optimization), the cv-NN accurately produces the natural movie using only its connection weights (Fig. 4c,d and Movie S2). In this case, the recurrent connections in the cv-NN create complex wave patterns (Fig. 4e and Movie S2). The recurrent connections and their resulting complex activity patterns are important for success in this task, as networks without recurrence do not produce accurate closed-loop forecasts (Fig. 4f).

Figure 4: The recurrent network performs next-frame forecasting of a natural video input. (a) Training follows as in the moving bump example (Fig. 2a). **(b)** Next-frame closed-loop forecasting follows as in the moving bump example (Fig. 2b). **(c)** Video frames of the data: a person walking. **(d)** Corresponding closed-loop forecasts generated by the network model in the case of optimal recurrence. **(e)** Corresponding network states for the optimal-recurrence case (panel d). Cosine of phase shown. **(f)** Same as d, but in the absence of recurrence.

We then studied what specific features of the recurrent connections enable predicting naturalistic movie inputs. As in the moving bump example, networks perform best when recurrence and input are approximately balanced, and the performance quickly decays when

the recurrence is too weak or too strong (Figs. 5a,b). This result shows that, as in the simple case of the moving bump, the complex spatiotemporal predictions generated by the network depend on a sophisticated interplay between input and recurrent connections. We next studied the role of connection topography and distance-dependent time delays. To do this, we started with networks that achieve accurate predictions and randomly shuffled both the connections and time delays (Fig. 6a). We then compared the closed-loop forecast performance and network activity in the topographic and shuffled cases. In the topographic case, the cv-NN produces accurate predictions and complex traveling wave patterns, as before (Fig. 6b,c). The shuffled versions of the cv-NN, however, produce spatiotemporally unstructured activity in the recurrent layer (Fig. 6d) and do not achieve accurate closed-loop forecasts, even after retraining (Fig. 6e; see also Supplementary Materials - Table S3 and Movie S3). Finally, the specific spatiotemporal structure of the input movie is also important: a cv-NN at the optimal hyperparameters for a natural movie cannot be retrained to do closed-loop forecasting on a randomized (phase-shuffled) version of the same movie (Supplementary Materials - Table S1), demonstrating that the cv-NN utilizes the specific spatiotemporal correlations in the movie to generate its forecast. Taken together, these results demonstrate that the complex spatiotemporal patterns generated by horizontal recurrent connections in the cv-NN enable performance on next-frame prediction and closed-loop forecasting tasks for sophisticated natural movie inputs.

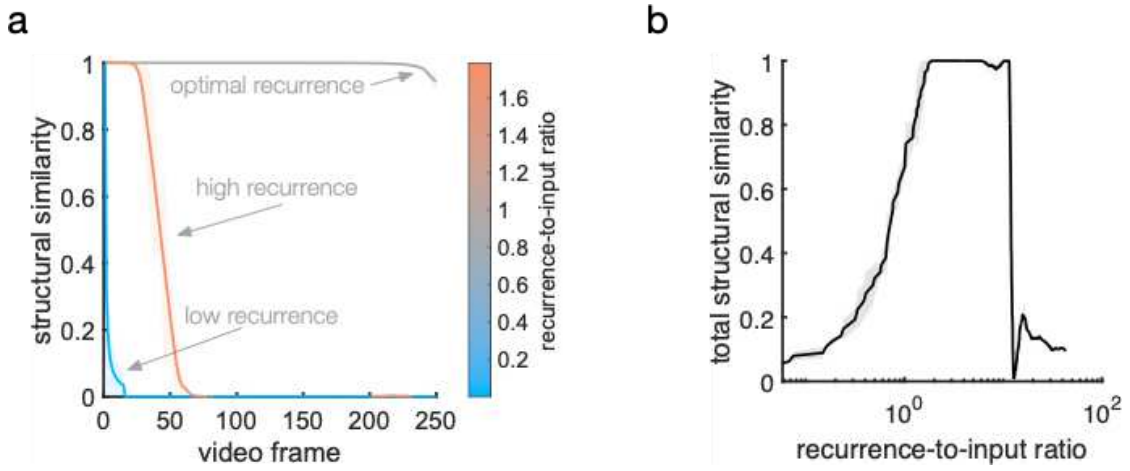

Figure 5: Natural movie forecast performance depends on specific properties of the recurrent connections. **(a)** Several examples of closed-loop forecast performance. Structural similarity between a forecast frame and the ground truth as a function of video frame during closed-loop forecasting. Each curve corresponds to a different ratio of recurrent strength to input strength. Curves have been smoothed by a moving-average filter (filter width of 30 time steps). Shaded error is the absolute difference between filtered and unfiltered. **(b)** Total structural similarity, in which a single SSIM is computed for the whole movie, as a function of the recurrence-to-input ratio. In the parameter space, each point differs only in recurrent strength. Smoothing and error shading is the same as in **a**.

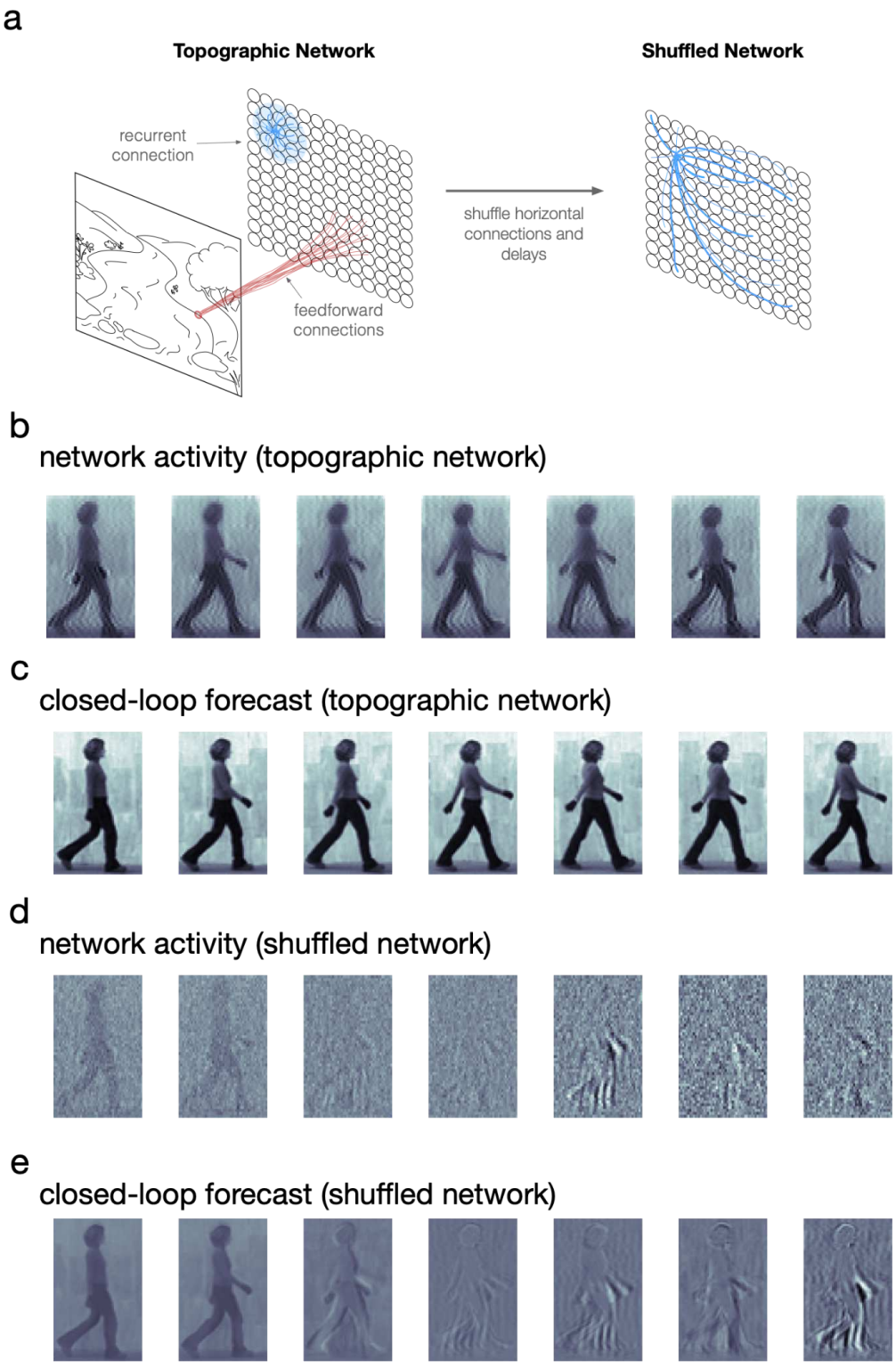

Figure 6: Randomly shuffling recurrent connections eliminates nTWs and ability to forecast. (a) Left: the topographic network model used throughout this study, featuring feedforward projections of the image input (red lines) and local distance-dependent horizontal connectivity (blue lines). There are also synaptic time delays proportional to a neuron pair's separation distance within the horizontal recurrent circuitry. Right: by randomizing the horizontal connection weights and time delays assigned to the network neurons, the topography in the network is

removed. **(b)** The network activity of the topographic network in response to frames of a natural movie input. **(c)** Closed-loop forecasts generated by the topographic network. Forecast frames correspond to network states in **a**. **(d)** Network activity of the shuffled network. **(e)** Closed-loop forecasts generated by the shuffled network.

The nTW network model is capable of forecasting multiple movies without retraining

We lastly sought to understand whether the cv-NN could perform closed-loop forecasts on multiple movies it had previously learned, and switch flexibly with changing inputs. To do this, we implemented a simple competitive process (Methods - Movie switching), so that the network could adapt its output based on the similarity of its prediction to its input (Fig. 7a). When performing a closed-loop forecast, this extended network model can receive a new input from its learned set, and then rapidly switch to closed-loop forecasting this new movie input within a few frames (Fig. 7b and Movie S4). This result demonstrates that the process of closed-loop forecasting, mediated by horizontal recurrent fibers in the network, can generalize to realistic visual conditions with multiple, changing input streams.

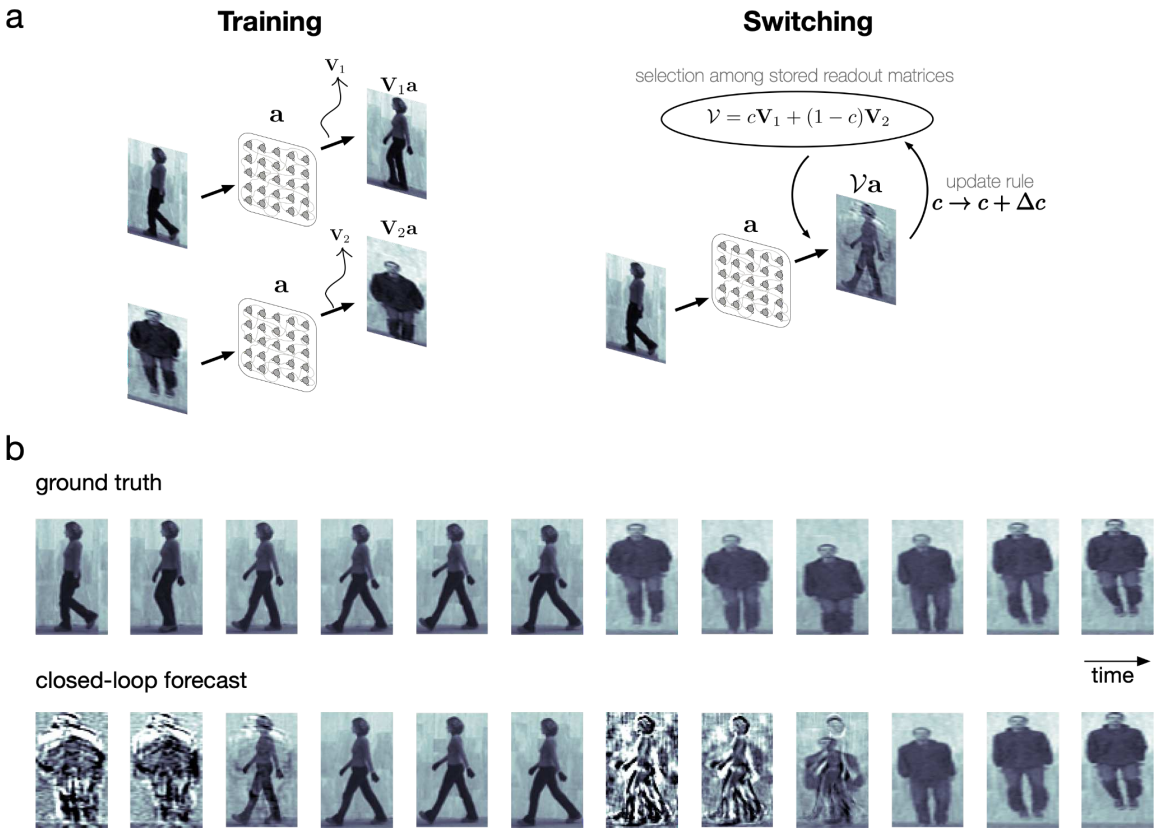

Figure 7: The network is capable of forecasting multiple movies without being retrained. (a) The recurrent network model was adapted to contain a higher-level competitive-learning process. Left: Readout matrices were learned separately for separate examples. Right: Storing the learned readout matrices in a higher-level matrix, the

present network state drove the aggregate matrix toward either of the learned matrices via an unsupervised competitive learning rule. **(b)** Beginning with feeding frames from movie 1, the network takes some time to recall the learned matrix that results in an accurate closed-loop forecast. Quickly switching to a different movie, the network once again takes some time to adjust its output weights before converging to the correct ones for an accurate closed-loop forecast.

Discussion

In this work, we have introduced a model to understand whether traveling waves generated by horizontal connections in visual cortex may play a computational role in processing natural visual inputs. By adapting a recurrent neural network model using a new dynamical update rule and a new learning rule, this model can efficiently learn video inputs ranging from simple visual stimuli to complex natural scenes. Further, this network model is broadly consistent with spatiotemporal dynamics recently observed in the visual system of the alert primate. In the case of a single point stimulus (Fig. 1c), the network produces a traveling wave radiating out from the point of input, similar to the responses observed in primary and secondary visual cortex of the awake monkey in response to a brief visual input[25]. In the case of a moving bump stimulus (Fig. 2), the network produces a bump of activity, reflecting the movie input but embedded within a larger spatiotemporal pattern, consistent with anticipatory responses observed in retinal populations[44] and in primary visual cortex[45]. Finally, in the case of naturalistic movie inputs (Fig. 4), the network produces complex spatiotemporal patterns, which can be mathematically described in this model as the summation of multiple traveling waves[33,34]. The responses of this network to natural movie inputs ranging from simple to complex are thus qualitatively consistent with observations of neuronal dynamics *in vivo*.

These results provide fundamental insight into the function of horizontal recurrent connections, whose effect on the moment-by-moment dynamics in the visual system has remained unexplained. While there has been much interest in the function of recurrent horizontal fibers in visual cortex, for example in explaining direction and orientation selectivity in V1[7,8], or in center-surround models of the receptive field[11,13,46], general computational roles for traveling waves generated by the massive recurrent circuitry in single cortical areas on the single-trial level remain unknown. Successful models of the visual system, including feature-based models and deep convolutional neural networks, have provided insight into how neural systems could process single image inputs, but explain only a fraction of the variance in neural responses to natural sensory stimuli[15,47,48]. The cv-NN may provide new opportunities for understanding how

the visual system processes continuously updated, movie-like visual inputs, where information is extracted from the visual environment moment-by-moment as it comes from the eye. The sophisticated closed-loop movie forecasts produced by this network, and the fact that this closed-loop forecast process can generalize to multiple movie inputs, represent an important step in explaining the computational role of recurrent connections and traveling waves in visual cortex.

Methods

Network connectivity

The recurrent network is arranged on a square grid of *N* nodes. The network grid is treated as a discretized Euclidean plane such that the side lengths span distances of unity. Boundaries are not periodic. The recurrent weight w_{ij} from node j to node i is inversely proportional to their Euclidean distance d_{ij} so as to give local connectivity like that of the neocortical sheet. Specifically, w_{ij} is Gaussian as a function of d_{ij}:

$$w_{ij} = ae^{-d_{ij}^2/(2b^2)}.$$

The coefficient a is called the recurrent strength and the standard deviation b is called the recurrent length. Both are free parameters. The maximum possible value of d_{ij} is $\sqrt{2}$ (corner to corner), and, for example, $b = 1$ means that the recurrent length equals the network side length. Further, all N^2 such weights are strictly positive, and the N-by-N matrix of such weights is symmetric ($w_{ij} = w_{ji}$). Diagonal weights (w_{ii}) are not set to zero.

Network dynamics

Network dynamics are given by a complex-valued equation. A complex number z is of the form $z = x + \mathrm{i}y$, where x is the real part, y is the imaginary part, and i is the imaginary constant defined as $\mathrm{i}^2 = -1$. Equivalently, $z = me^{\mathrm{i}\phi}$, where m is the modulus and ϕ is the argument. A complex number is intuitively visualized as a two-dimensional vector, where (x, y) is its Cartesian representation and (m, ϕ) is its polar representation. What distinguishes a complex number from a standard two-dimensional vector is the multiplication rule: multiplication of two

complex numbers corresponds to both a scaling and a rotation in the so-called complex plane. This property makes complex-valued representations of observable quantities more concise than real-valued representations, and thus, complex numbers are a central tool in physics and engineering. From the perspective of biological vision, a complex-valued representation is useful. Since phase information is important for representing visual inputs, complex-valued models, which efficiently represent phase in the argument ϕ, are ideal. Indeed, complex-valued models of vision are widely explored[49]. Given the practical utility of artificial neural networks and deep learning (including for modeling biological neural networks), complex-valued neural networks, in which the neural activations are complex-valued, are of great interest. However, they are notoriously difficult to train, especially in a recurrent architecture[50]. We make an advance here on this front by choosing a unique dynamical equation and by exploiting the advantages of reservoir computing.

The discrete-time dynamical equation for each node *i* is

$$a_i[t] = a_i[t-1] + \left(x_i[t] - \mathrm{i} \sum_{j=1}^{N} w_{ij} e^{\mathrm{i}\left(a_j[t-1-\tau_{ij}] - a_i[t-1]\right)} \right), \tag{1a}$$

$$a_i[t] := a_i[t] \;/\; \left|a_i[t]\right|. \tag{1b}$$

Here, $a_i[t]$ is the complex-valued activation, $x_i[t]$ is the feedforward input of the image stimulus to node i, and w_{ij} is the recurrent weight from node j to node j (Methods - Network connectivity). Further, τ_{ij} is the discrete time delay between nodes i and j, given by $\tau_{ij} = \mathrm{round}[d_{ij}/v]$ in which the Euclidean distance d_{ij} between nodes i and j (Methods - Network connectivity) is scaled by the parameter v, which represents the speed of activation transmission across the network, and $\mathrm{round}[d_{ij}/v]$ rounds d_{ij}/v to the nearest integer in accord with the discrete-time dynamics. The value of v is randomly sampled between 0 and 0.1 ($v \in (0, 0.1)$), meaning the activation travels a distance of up to one-tenth the network side length per time step. Lastly, the modulus of $a_i[t]$ (*i.e.*, $|a_i[t]|$) is normalized (Equation 1b), which confines $a_i[t]$ on the complex unit circle, and thus, the phase of $a_i[t]$ contains the dynamics. We note that modulus normalization is a common operation used in complex-valued neural networks[50].

The specific form of Equation 1a is unique compared to other complex-valued neural-network equations because it involves a pairwise node attraction $a_j[t-1-\tau_{ij}] - a_i[t-1]$. Another system with pairwise attraction is the Kuramoto model, a popular model for studying synchronization in nonlinear systems[51–53]. Our presented system has a correspondence with the Kuramoto model[54], and allows the description of the dynamics for individual realization in terms of the eigenvalues and eigenvectors of the network[33]. Along with the choice of local network connectivity and distance-dependent delays, the presented system gives rise to meaningful spatiotemporal self-organization dynamics, and for this reason, the recurrent weights $\{w_{ij}\}$ need not be trained.

The initial network state is $a_i[0] = 0 + 0\mathrm{i}$ for all nodes, and the first several time steps contain transient activity associated with the input disrupting the initial steady state of the system. For the time-of-stimulus prediction task, this transient activity is important to the model and was used, while for the next-frame forecasting task, it is distracting to the model and was discarded.

Image read-in

Each discrete time step, a digital grayscale image is read into the network. Prior to read-in, the image is mean-subtracted and divided by its standard deviation across all its pixels (*i.e.*, z-scored). Image read-in is accomplished with a local feedforward projection, which mimics retinotopy and preserves the spatial correlations in the image. Technically, this is a two-dimensional interpolation using the bilinear kernel common in image processing, which takes a weighted average in the nearest 2-by-2 pixel neighbourhood. The projected image has $\sqrt{N}$ rows and $\sqrt{N}$ columns like the network grid, and each pixel intensity of the projected image is given by $x_i[t]$ (Equation 1a). Lastly, $x_i[t]$ is scaled according to $x_i[t] := \epsilon x_i[t]$, where ϵ is called the input strength. In our model, ϵ is the third and final free parameter after the recurrent strength and recurrent length.

Time-of-stimulus prediction task

Classification was performed using the basic perceptron. For an input vector $\vec{v} = (1,\ v_1,\ \cdots,\ v_N)^T$, where $v_1,\ \ldots,\ v_N$ are features, and a label $l \in \{0, 1\}$, the goal is to find a hyperplane $\vec{u}^T\vec{v} = b + u_1v_1 + \cdots + u_Nv_N = 0$, where $\vec{u} = (b,\ u_1,\ \cdots,\ u_N)^T$ is a

vector containing the bias b and weights $u_1, \ldots, u_N$, that separates the data in the N-dimensional feature space according to their binary class (0 or 1). During training, with a sub-optimal $\vec{u}$ vector and one example $\vec{v}$ vector, the output classification $l = H(\vec{u}^T \vec{v})$ is computed, where $H(\cdot)$ is the Heaviside step function defined as unity for positive argument and zero otherwise. For the desired classification d (either 0 or 1), the signed distance $\Delta = d - l$ is computed, where $\Delta \in \{-1, 0, 1\}$. With each new example $\vec{v}$, the $\vec{u}$ vector is updated using the delta rule $\vec{u} := \vec{u} + \lambda \vec{v} \Delta$, where λ is the learning rate. To use the perceptron in multiclass classification, the one-versus-rest scheme is used. That is, for the set of classes $C = \{c_1, c_2, \ldots, c_i, \ldots, c_M\}$, binary classification is performed separately M times. Each time i, the two classes are defined such that $c_i = 1$ and $C \setminus c_i = 0$, where "\" denotes the set difference. Then, there are M weight vectors $\vec{u}_1, \ldots, \vec{u}_i, \ldots, \vec{u}_M$, and M inner products $f_1 = \vec{u}_1^T \vec{v}, \ \ldots, \ f_i = \vec{u}_i^T \vec{v}, \ \ldots, f_M = \vec{u_M}^T \vec{v}$ for a given data vector $\vec{v}$. The multiclass classification is $\mathrm{argmax}_{c_i} \{f_1, \ldots, f_i, \ldots, f_M\}$.

In the time-of-stimulus classification task (Fig. 1c), input frames were 50 by 50 pixels, and the network was 50 by 50 nodes. There were six frames. One of the first five frames was randomly chosen to contain the point stimulus, and the remaining frames were entirely zero intensity. The point stimulus was an isotropic two-dimensional Gaussian of standard deviation 0.05, and the input frames are defined on the Cartesian grid $[-2, 2] \times [-2, 2]$. The stimulus was centered in the frame. The sequence of frames was sequentially input to the network. There are exactly five classes: each of the first five frames in which the point stimulus could occur. The column vector of activations corresponding to the final (sixth) frame was used as predictors for all trials. The task was repeated 100 000 times, with the time of stimulus (1 or 2 or 3 or 4 or 5) randomly rechosen each time.

Next-frame forecasting

The network outputs an image of M_r rows and M_c columns of pixels–the same size as the input image–at each time step. In both examples (moving bump and natural movie), the network was 50 by 50 nodes. Recalling that $a_i[t]$ is the complex-valued activation of node i at discrete time t (Equations 1a and 1b), the output transformation is linear:

$$y_i[t] = \sum_{j=1}^{N} v_{ij} a_j[t]'.$$

Here, $y_i[t]$ is the i^{th} pixel intensity of the output image, and v_{ij} is the $(i,j)^{\text{th}}$ readout weight of the M-by-N matrix V, where $M = M_r M_c$. The prime notation (') indicates that $a[t] = \left(a_1[t],\ a_2[t],\ \cdots,\ a_N[t]\right)^T$ was mean-subtracted, which was done to avoid an intercept term during training.

The readout weights $\{v_{ij}\}$ of V are the only weights trained in our model, making our network a reservoir computer. Reservoir computers are recurrent neural networks that avoid the issues associated with training recurrent weights, and have been shown to perform well in time series forecasting[38]. Suppose training begins at time step 1, after discarding the initial transient, and ends at time step T. Defining $a[t]' = \left(a_1[t]',\ a_2[t]',\ \cdots,\ a_N[t]'\right)^T$, the matrix of regressors is then

$$A = \Big[a[1]'\ a[2]'\ \cdots\ a[T]'\Big],$$

and the matrix of regressands (desired outputs) is

$$D = \Big[f[2]\ f[3]\ \cdots\ f[T+1]\Big].$$

Hence, the desired outputs are simply the set of one-step-ahead frames. Here, $f[t]$ is the column vectorization of the t^{th} input image frame (before read-in), and is also mean-subtracted. Training entails ordinary least-squares linear regression between A and D. Because D is highly underdetermined (containing far fewer frames than pixels per frame), the matrix 2-norm of V was simultaneously minimized during regression to reduce model bias.

Following training is *closed-loop forecasting*. At this point, the network activation has been primed by being driven with the training frames, and the readout matrix V has been trained. In the first time step of closed-loop forecasting, we input the corresponding video frame.

Subsequently for steps $\{t\}$, the predicted output at time step t serves as the input for time step t+1.

In the moving bump example (Fig. 2), the frames are 30 by 30 pixels and defined on a $[-2,2] \times [-2,2]$ Cartesian grid. A two-dimensional isotropic Gaussian of standard deviation 0.2 traced a Lissajous curve given by the parametric equations $x_c(t) = \sin(t/3)$ and $y_c(t) = \cos(t)$, where (x_c, y_c) is the center of the Gaussian in space and t is a continuously valued time variable[55]. The Lissajous trajectory was discretized to have 100 frames per cycle. The first cycle was discarded to omit the initial transient network activity, the network was trained on the subsequent 3 cycles, and closed-loop forecasting was performed on the 2 cycles subsequent to that.

In the natural video example (Fig. 4), a walking video from the Weizmann Human Action Dataset[56] was used, in which a person walks across the scene. We present several key examples here, but note that the model successfully performs closed-loop forecasting for all movies in this dataset. Segmentation masks of the people in the videos are included with this dataset (https://www.wisdom.weizmann.ac.il/~vision/SpaceTimeActions.html). Using these masks, we cropped the frames so that the person was centered throughout the entire walk, giving frames of approximately 80 by 50 pixels. Without performing this step, our network model would fail: the training data would be independent from the closed-loop forecast data since they would occupy exclusive regions of the pixel space, and the model would not generalize to the prediction data. Such nonstationary data have been successfully taught to networks with approximate translation invariance, and translation invariance is likely used in the brain to learn such processes[57]. However, translation invariance is beyond the scope of our study. The frames were then resized to be exactly 80 by 50 pixels. Finally, each video was around 70 frames long. To get more frames without interpolation, we “bookended” each video by concatenating it with its temporal reverse sequence, where one cycle consists of the original frames followed by the bookended frames. The result is a longer video with the same spatiotemporal statistics. The first cycle was discarded to omit the initial transient network activity, the network was trained on the subsequent 3 cycles, and closed-loop prediction was performed on the 2 cycles subsequent to that.

To measure the balance between feedforward input and recurrent interaction, we devised the *recurrence-to-input ratio*. Per Equation 1a, the input and recurrence terms are the column vectors

$$x[t] = \{x_i[t]\}_{i=1}^{N}$$

and

$$r[t] = \left\{ -\mathrm{i} \sum_{j=1}^{N} w_{ij} e^{\mathrm{i}(a_j[t-1-\tau_{ij}]-a_i[t-1])} \right\}_{i=1}^{N},$$

respectively. Further, let the matrices

$$R = \Big[r[t] \; r[t+1] \; \cdots \; r[t'] \Big]$$

and

$$X = \Big[x[t] \; x[t+1] \; \cdots \; x[t'] \Big]$$

be the horizontal concatenations of $r[t]$ and $x[t]$, respectively, over closed-loop forecast times $\{t, t+1, \ldots, t'\}$. The ratio is defined as

$$\frac{\|R\|_F}{\|X\|_F},$$

where $\|G\|_F$ denotes the Frobenius matrix norm of a matrix G, which is equivalent to the Euclidean vector norm of the vectorization of G.

Movie switching

The network was trained on two movie inputs: one of a walking person (movie 1) and one of a jumping person (movie 2), both from the Weizmann dataset. The same recurrent matrix was used in each case–only the learned matrices (V_1 and V_2, respectively) differed. Let $\mathcal{V} = cV_1 + (1-c)V_2$, where $c \in [0, 1]$. $\mathcal{V}$ stores both learned matrices, and the present input modulates the relative contribution of V_1 and V_2 using an update rule for c. The structural similarity between the input and output were computed at each time step t ($\mathrm{SSIM}[t]$), and the change thereof was computed at each time step as $\Delta\mathrm{SSIM} = \mathrm{SSIM}[t] - \mathrm{SSIM}[t-1]$. The update rule is $c := c + \Delta c$, where $\Delta c = -\eta\ \mathrm{sgn}[\Delta\mathrm{SSIM}]$ and η is the learning rate, set to 0.1. Depending on which movie (movie 1 or movie 2) drives the network, c tends toward 1 or 0, respectively. Once this happens, this driving input is removed and closed-loop forecasting commences as described. Switching entails instantaneously transitioning from closed-loop forecasting of one movie to driving the network with the frames of another movie. c then updates as described and is followed by closed-loop forecasting again.

Parameter optimization

The random-search algorithm was used to optimize parameters for closed-loop forecasting. Within specified bounds, each parameter was randomly sampled, giving a point in the parameter space. The parameter space was randomly sampled in this way many times, and each time, the structural similarity index was computed as the performance index. The bounds within which the parameters were sampled are given in Table I.

parameter	sampled interval
recurrent strength	(0, 0.2)
recurrent length	(0, 0.2)
input strength	(0, 0.2)
v	(0, 0.1)

Table I: Intervals over which model parameters were randomly searched during optimization.

Acknowledgements: This work was supported by the Canadian Institute for Health Research and NSF (NeuroNex Grant No. 2015276), BrainsCAN at Western University through the Canada First Research Excellence Fund (CFREF), Gatsby Charitable Foundation, the Fiona and Sanjay Jha Chair in Neuroscience, the Swartz Foundation, Compute Ontario (computeontario.ca),

Compute Canada (computecanada.ca), SPIRITS 2020 of Kyoto University, and the Western Academy for Advanced Research. R.C.B. gratefully acknowledges the Western Institute for Neuroscience Clinical Research Postdoctoral Fellowship. G.B.B. gratefully acknowledges the Canadian Open Neuroscience Platform (Graduate Scholarship), the Vector Institute (Postgraduate Affiliate), and the National Sciences and Engineering Research Council of Canada (Canada Graduate Scholarship - Doctoral). The authors thank Alex Busch for her help with illustrations.

Data and code availability: All data and code associated with this work are available at https://github.com/mullerlab.

Contributions: Conceptualization: G.B.B., L.M.; data curation: G.B.B.; formal analysis: G.B.B., L.M.; funding acquisition: J.R., L.M.; investigation: G.B.B., L.M.; methodology: G.B.B., R.C.B., Z.D., J.R., L.M.; supervision: J.R., L.M.; visualization: G.B.B.; writing–original draft: G.B.B., L.M.; writing–review and editing: G.B.B., R.C.B., Z.D., J.R., L.M.

References

1. Markov, N. T. *et al.* Weight consistency specifies regularities of macaque cortical networks. *Cereb. Cortex* **21**, 1254–1272 (2011).
2. Bruno, R. M. & Sakmann, B. Cortex is driven by weak but synchronously active thalamocortical synapses. *Science* **312**, 1622–1627 (2006).
3. Swindale, N. V. Visual map. *Scholarpedia J.* **3**, 4607 (2008).
4. Hubel, D. H. & Wiesel, T. N. Receptive fields of single neurones in the cat's striate cortex. *J. Physiol.* **148**, 574–591 (1959).
5. Riesenhuber, M. & Poggio, T. Hierarchical models of object recognition in cortex. *Nat. Neurosci.* **2**, 1019–1025 (1999).
6. Angelucci, A. *et al.* Circuits for local and global signal integration in primary visual cortex. *J. Neurosci.* **22**, 8633–8646 (2002).
7. Douglas, R. J., Koch, C., Mahowald, M., Martin, K. A. & Suarez, H. H. Recurrent excitation in neocortical circuits. *Science* **269**, 981–985 (1995).
8. Sompolinsky, H. & Shapley, R. New perspectives on the mechanisms for orientation selectivity. *Curr.*

Opin. Neurobiol. **7**, 514–522 (1997).

9. Blakemore, C. & Tobin, E. A. Lateral inhibition between orientation detectors in the cat's visual cortex. *Exp. Brain Res.* **15**, 439–440 (1972).
10. Allman, J., Miezin, F. & McGuinness, E. Stimulus specific responses from beyond the classical receptive field: neurophysiological mechanisms for local-global comparisons in visual neurons. *Annu. Rev. Neurosci.* **8**, 407–430 (1985).
11. Field, D. J., Hayes, A. & Hess, R. F. Contour integration by the human visual system: Evidence for a local 'association field'. *Vision Res.* **33**, 173–193 (1993).
12. Gilbert, C. D. Adult cortical dynamics. *Physiol. Rev.* **78**, 467–485 (1998).
13. Albright, T. D. & Stoner, G. R. Contextual influences on visual processing. *Annu. Rev. Neurosci.* **25**, 339–379 (2002).
14. Kietzmann, T. C. *et al.* Recurrence is required to capture the representational dynamics of the human visual system. *Proc. Natl. Acad. Sci. U. S. A.* **116**, 21854–21863 (2019).
15. Kar, K., Kubilius, J., Schmidt, K., Issa, E. B. & DiCarlo, J. J. Evidence that recurrent circuits are critical to the ventral stream's execution of core object recognition behavior. *Nat. Neurosci.* **22**, 974–983 (2019).
16. Bringuier, V., Chavane, F., Glaeser, L. & Frégnac, Y. Horizontal propagation of visual activity in the synaptic integration field of area 17 neurons. *Science* **283**, 695–699 (1999).
17. Roland, P. E. *et al.* Cortical feedback depolarization waves: a mechanism of top-down influence on early visual areas. *Proc. Natl. Acad. Sci. U. S. A.* **103**, 12586–12591 (2006).
18. Xu, W., Huang, X., Takagaki, K. & Wu, J.-Y. Compression and reflection of visually evoked cortical waves. *Neuron* **55**, 119–129 (2007).
19. Nauhaus, I., Busse, L., Carandini, M. & Ringach, D. L. Stimulus contrast modulates functional connectivity in visual cortex. *Nat. Neurosci.* **12**, 70–76 (2009).
20. Reimer, A., Hubka, P., Engel, A. K. & Kral, A. Fast propagating waves within the rodent auditory cortex. *Cereb. Cortex* **21**, 166–177 (2011).
21. Townsend, R. G. *et al.* Emergence of complex wave patterns in primate cerebral cortex. *J. Neurosci.* **35**, 4657–4662 (2015).

22. Slovin, H., Arieli, A., Hildesheim, R. & Grinvald, A. Long-term voltage-sensitive dye imaging reveals cortical dynamics in behaving monkeys. *J. Neurophysiol.* **88**, 3421–3438 (2002).
23. Sato, T. K., Nauhaus, I. & Carandini, M. Traveling waves in visual cortex. *Neuron* **75**, 218–229 (2012).
24. Davis, Z. W., Muller, L., Martinez-Trujillo, J., Sejnowski, T. & Reynolds, J. H. Spontaneous travelling cortical waves gate perception in behaving primates. *Nature* **587**, 432–436 (2020).
25. Muller, L., Reynaud, A., Chavane, F. & Destexhe, A. The stimulus-evoked population response in visual cortex of awake monkey is a propagating wave. *Nat. Commun.* **5**, 3675 (2014).
26. Davis, Z. *et al.* Spontaneous traveling waves naturally emerge from horizontal fiber time delays and travel through locally asynchronous-irregular states. *Nature Communications* (2021).
27. Takahashi, K. *et al.* Large-scale spatiotemporal spike patterning consistent with wave propagation in motor cortex. *Nat. Commun.* **6**, 7169 (2015).
28. Trabelsi, C. *et al.* Deep Complex Networks. *arXiv [cs.NE]* (2017).
29. Heeger, D. J. & Mackey, W. E. Oscillatory recurrent gated neural integrator circuits (ORGaNICs), a unifying theoretical framework for neural dynamics. *Proc. Natl. Acad. Sci. U. S. A.* **116**, 22783–22794 (2019).
30. Krizhevsky, A., Sutskever, I. & Hinton, G. E. Imagenet classification with deep convolutional neural networks. *Adv. Neural Inf. Process. Syst.* **25**, (2012).
31. Graves, A. Supervised Sequence Labelling with Recurrent Neural Networks. *Studies in Computational Intelligence* (2012) doi:10.1007/978-3-642-24797-2.
32. Hwang, K. & Sung, W. Online Sequence Training of Recurrent Neural Networks with Connectionist Temporal Classification. *arXiv [cs.LG]* (2015).
33. Budzinski, R. C. *et al.* Geometry unites synchrony, chimeras, and waves in nonlinear oscillator networks. *Chaos* **32**, 031104 (2022).
34. Muller, L., Chavane, F., Reynolds, J. & Sejnowski, T. J. Cortical travelling waves: mechanisms and computational principles. *Nat. Rev. Neurosci.* **19**, 255–268 (2018).
35. Hellwig, B. A quantitative analysis of the local connectivity between pyramidal neurons in layers 2/3 of the rat visual cortex. *Biol. Cybern.* **82**, 111–121 (2000).

36. Binzegger, T., Douglas, R. J. & Martin, K. A. C. A quantitative map of the circuit of cat primary visual cortex. *J. Neurosci.* **24**, 8441–8453 (2004).
37. Girard, P., Hupé, J. M. & Bullier, J. Feedforward and feedback connections between areas V1 and V2 of the monkey have similar rapid conduction velocities. *J. Neurophysiol.* **85**, 1328–1331 (2001).
38. Jaeger, H. & Haas, H. Harnessing Nonlinearity:Predicting Chaotic Systemsand Saving Energy in Wireless Communication. *Science* **304**, (2004).
39. Pathak, J., Hunt, B., Girvan, M., Lu, Z. & Ott, E. Model-Free Prediction of Large Spatiotemporally Chaotic Systems from Data: A Reservoir Computing Approach. *Phys. Rev. Lett.* **120**, 024102 (2018).
40. Mathieu, M., Couprie, C. & LeCun, Y. Deep multi-scale video prediction beyond mean square error. *arXiv [cs.LG]* (2015).
41. Wang, Z., Bovik, A. C., Sheikh, H. R. & Simoncelli, E. P. Image quality assessment: from error visibility to structural similarity. *IEEE Trans. Image Process.* **13**, 600–612 (2004).
42. Stettler, D. D., Das, A., Bennett, J. & Gilbert, C. D. Lateral connectivity and contextual interactions in macaque primary visual cortex. *Neuron* **36**, 739–750 (2002).
43. Moshe, B., Lena, G., Eli, S., Michal, I. & Ronen, B. Actions as space-time shapes. in *Proceedings of the Tenth IEEE International Conference on Computer Vision, Beiging, China* 17–20 (2005).
44. Berry, M. J., Brivanlou, I. H., Jordan, T. A. & Meister, M. Anticipation of moving stimuli by the retina. *Nature* vol. 398 334–338 (1999).
45. Benvenuti, G. *et al.* Anticipatory responses along motion trajectories in awake monkey area V1. 2020.03.26.010017 (2020) doi:10.1101/2020.03.26.010017.
46. Hess, R. F. & Dakin, S. C. Contour integration in the peripheral field. *Vision Res.* **39**, 947–959 (1999).
47. Olshausen, B. A. How Close Are We to Understanding V1? *Neural Comput.* **17**, 1665–1699 (2005).
48. Schrimpf, M. *et al.* Brain-Score: Which Artificial Neural Network for Object Recognition is most Brain-Like? *bioRxiv* 407007 (2020) doi:10.1101/407007.
49. Cadieu, C. F. & Olshausen, B. A. Learning intermediate-level representations of form and motion from natural movies. *Neural Comput.* **24**, 827–866 (2012).
50. Bassey, J., Qian, L. & Li, X. A Survey of Complex-Valued Neural Networks. *arXiv [stat.ML]* (2021).
51. Acebrón, J. A., Bonilla, L. L., Pérez Vicente, C. J., Ritort, F. & Spigler, R. The Kuramoto model: A

simple paradigm for synchronization phenomena. *Rev. Mod. Phys.* **77**, 137–185 (2005).

52. Rodrigues, F. A., Peron, T. K. D. M., Ji, P. & Kurths, J. The Kuramoto model in complex networks. *Phys. Rep.* **610**, 1–98 (2016).
53. Breakspear, M., Heitmann, S. & Daffertshofer, A. Generative models of cortical oscillations: neurobiological implications of the kuramoto model. *Front. Hum. Neurosci.* **4**, 190 (2010).
54. Muller, L., Mináč, J. & Nguyen, T. T. Algebraic approach to the Kuramoto model. *Phys Rev E* **104**, L022201 (2021).
55. Heim, N. & Avery, J. E. Adaptive Anomaly Detection in Chaotic Time Series with a Spatially Aware Echo State Network. *arXiv [cs.NE]* (2019).
56. Gorelick, L., Blank, M., Shechtman, E., Irani, M. & Basri, R. Actions as space-time shapes. *IEEE Trans. Pattern Anal. Mach. Intell.* **29**, 2247–2253 (2007).
57. Anselmi, F. *et al.* Unsupervised learning of invariant representations. *Theor. Comput. Sci.* **633**, 112–121 (2016).

Supplementary Files

This is a list of supplementary files associated with this preprint. Click to download.

- wc1supplement.pdf
- movies1.mp4
- movies2.mp4
- movies3.mp4
- movies4.mp4