The functional role of oscillatory dynamics in neocortical circuits: a computational perspective

Authors: Felix Effenberger,[1*] Pedro Carvalho,[1,†] Igor Dubinin,[1,2,†] Wolf Singer[1,2,3*]

Affiliations:

[1]Ernst Strüngmann Institute, Frankfurt am Main, Germany

[2]Frankfurt Institute for Advanced Studies, Frankfurt am Main, Germany

[3]Max Planck Institute for Brain Research, Frankfurt am Main, Germany

[†]Equal contributions

[*]To whom correspondence should be addressed;

E-mail: felix.effenberger@esi-frankfurt.de, wolf.singer@esi-frankfurt.de.

Abstract: Biological neuronal networks exhibit hallmark features such as oscillatory dynamics, heterogeneity, modularity, and conduction delays. To investigate which of these features support computations or are epiphenomena, we simulated recurrent networks and analyzed the effects of implementing these features on performance in benchmark tests for pattern classification. Configuring the network nodes as damped harmonic oscillators reproduced essential features of cortical dy-

namics and led to networks that vastly outperformed non-oscillating RNN architectures in learning speed, task performance, parameter efficiency, and noise tolerance. Implementation of additional biological features further improved performance. Analyses of network dynamics uncovered a novel, powerful computational principle and provided plausible a posteriori explanations for features of natural networks whose computational role has remained elusive for a long time. It is discussed how the computational principle realized in the model could be implemented in analog computing devices capable of unsupervised learning.

One-Sentence Summary: Recurrent networks of damped harmonic oscillators have powerful computational abilities and reproduce biological phenomena.

Main Text: Biological neuronal systems such as the cerebral cortex are characterized by several canonical anatomical and physiological features. Among those are recurrent connections between network nodes within and between processing layers (*1, 2*), the propensity of nodes and subnetworks to oscillate in different preferred frequency ranges (*3, 4*), heterogeneous but tuned conduction delays (*5, 6*), and activity-dependent adjustment of the gain of connections (*7, 8*) by Hebbian synaptic plasticity (*9*).

These features endow neuronal networks with highly complex and nonlinear dynamics capable of generating high-dimensional spatiotemporal patterns in a self-organized manner (*10*), see (*2*) for a review. However, the functional significance of these dynamical phenomena remains unclear as they naturally emerge from neuronal interactions and therefore could be epiphenomena without functional role. Canonical circuit motifs such as lateral or recurrent inhibition have been proposed to serve contrast enhancement, gain control, the expansion of dynamic ranges, and competition. However, these circuits also have the propensity to oscillate. If these microcircuits interact with each other, as is the case in cortical networks, additional dynamical phenomena emerge. In simultaneous recordings from

multiple network nodes, these dynamics manifest as frequency-varying oscillations (*11, 12*), transient synchronization or desynchronization of discharges (*13*), resonance (*14*), entrainment (*15*), phase shifts (*16*), and traveling waves (*17–19*). To determine in physiological experiments whether any of these dynamical phenomena serves a functional role is notoriously difficult because of the complexity, non-stationarity and variability of the dynamics, the technical limitations to monitor the activity of a sufficient number of nodes with the required temporal resolution and the difficulty to apply knock-out or gain-of-function strategies to prove causal relations. Interfering with single variables of such complex, highly integrated systems inevitably impacts the behavior of the entire system, making it difficult to assess the putative function of a particular feature (*20*). Therefore, virtually all evidence for the functional role of the aforementioned dynamical phenomena is correlative in nature.

To overcome this epistemic hurdle and isolate computational principles, we simulated recurrent neuronal networks (RNNs) and analyzed the effects of implementing canonical features of the cerebral cortex on the dynamics of the networks and on their performance in standard benchmark tests on pattern recognition. This approach at the intersection of neuroscience and machine learning differs from most of the previous studies because these either tried to mechanistically model features of biological networks but often without validation of functions by benchmark tests for performance, or they investigated machine learning models with a focus on improving performance, but disregarded aspects of biological realism and system dynamics (*21, 22*).

Because of the marked propensity of cortical circuits to engage in oscillatory activity, we were particularly interested in examining the putative functions of oscillatory phenomena. Although the role of oscillating neurons and microcircuits in the generation of motor patterns is well established (*23*), it remains a matter of discussion whether oscillations support computations in the context of cognitive processes in the cerebral cortex and, if so, how (*24–27*).

In a first step inspired by physiological experiments (*28*), we thus examined the consequences of configuring the network nodes as damped harmonic oscillators (DHOs) rather than as leaky integra-

tors. We call such networks harmonic oscillator RNNs (HORNs).

To our surprise, we found HORNs to outperform, sometimes by orders of magnitude, RNNs without oscillating units with respect to parameter efficiency, task performance, learning speed, and noise tolerance. Encouraged by these findings, we continued to implement additional characteristics of cortical networks in HORNs. We endowed them with nodes that prefer different oscillation frequencies, introduced heterogeneous conduction velocities for the recurrent connections, and implemented Hebbian synapses. In addition, we recurrently coupled two HORNs in order to examine the putative advantages of multilayer architectures as found in the brain. The inclusion of these biologically inspired features resulted in further improved task performance without increasing the number of trainable variables.

In-depth analyses of network dynamics uncovered a genuinely novel (to the best of our knowledge), powerful computational principle that exploits the rich transient and relaxation dynamics of coupled oscillators and casts a radically new view on recurrent network dynamics (see "Oscillating network nodes" below). In addition, the networks reproduced a wide range of physiological phenomena despite their conceptual simplicity and this allowed us to provide plausible a posteriori explanations for a number of anatomical and physiological phenomena whose function has so far remained elusive or has given rise to controversial discussions.

Last but not least, we show how HORNs can store stimulus priors through biologically plausible unsupervised Hebbian learning principles and how their dynamics supports the highly parallel search for optimal matches between sensory evidence and stored priors. As predicted (*2*), these matching operations lead to a fast collapse of high-dimensional spontaneous activity patterns towards stimulus specific dynamic substates amenable to linear classification, which is another hallmark of cognitive biological systems.

Oscillating network nodes

To facilitate the analysis of principles we designed a simple RNN and configured the nodes as damped harmonic oscillators (DHOs). We opted for DHOs for the following reasons: First, as canonical implementation of an oscillatory process, they permit full control of dynamics through few, interpretable parameters (gain, preferred frequency, damping). Second, in neuronal microcircuits, damped harmonic oscillations generically result from excitatory-inhibitory interactions or negative feedback subject to a damping force (*29, 30*). This makes DHOs the most basic choice for introducing oscillations at the level of a single network node. Importantly, a DHO node in our networks should not be considered to represent a single biological neuron, but rather as a microcircuit composed of recurrently coupled populations of excitatory (E) and inhibitory (I) neurons such as the (P)ING circuits (*30*). In this sense, HORNs should be understood as a mesoscale model of a cortical network (see Fig. 1A and Methods).

Each DHO node has one state variable x, the oscillator's time-varying amplitude, and three parameters that jointly control its relaxation dynamics: the natural frequency ω, the damping coefficient γ, and an excitability coefficient α (see Methods). These parameters can be thought of as the node's innate or long-term adaptation to the characteristics of the signals to be processed (Fig. 1A). A node is defined by the second-order ODE $\ddot{x}(t)+2\gamma\dot{x}(t)+\omega^2 x(t) = \alpha\sigma\left(I\left(t\right) + F\left(x(t), \dot{x}(t)\right)\right)$, where t denotes time, σ is an input nonlinearity that models local resource constraints, $F(x(t), \dot{x}(t))$ models the feedback input of the node to itself, capturing characteristics of the dynamics of the microcircuit that a DHO node represents, and $I(t)$ denotes the input to the node (see Methods). Despite their simplicity, DHO nodes capture the essential dynamical characteristics of various neural mass models (*28, 29, 31, 32*) (see Supplementary Materials). Thus, the oscillatory properties of the nodes implemented in our simulations are both physiologically plausible and experimentally confirmed (*28, 29, 31, 32*).

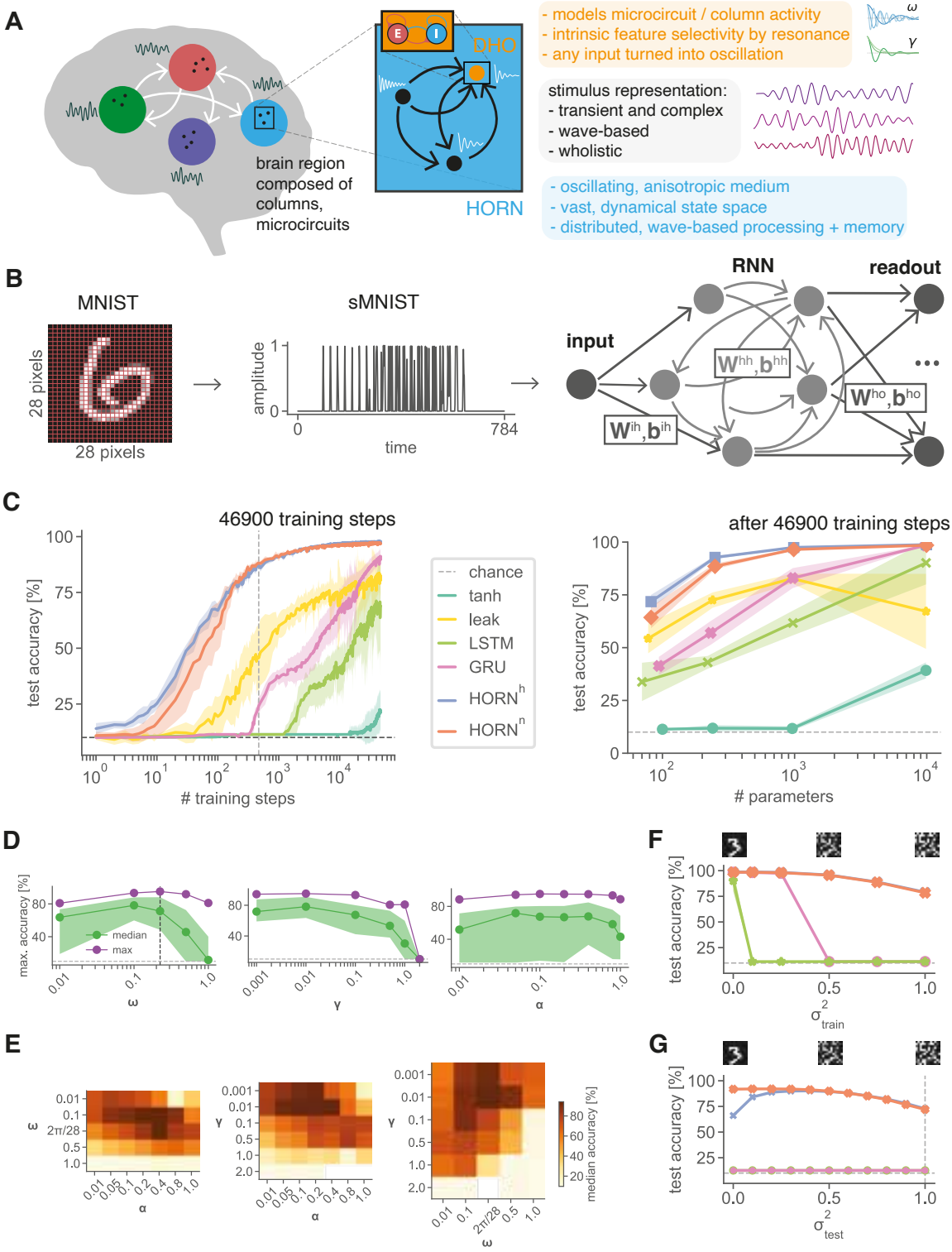
A
brain region composed of columns, microcircuits
DHO
HORN
- models microcircuit / column activity
- intrinsic feature selectivity by resonance
- any input turned into oscillation
stimulus representation:
- transient and complex
- wave-based
- wholistic
- oscillating, anisotropic medium
- vast, dynamical state space
- distributed, wave-based processing + memory
B
MNIST
sMNIST
28 pixels
amplitude
time
RNN
readout
input
C
46900 training steps
after 46900 training steps
test accuracy [%]
training steps
parameters
chance
tanh
leak
LSTM
GRU
D
max. accuracy [%]
median
max
E
median accuracy [%]
F
G

Fig. 1. Architecture of HORN networks and their performance in pattern recognition tasks. **A**. A HORN network composed of DHO nodes as a mesoscale model of brain activity. Each DHO node models the compound activity of a recurrently connected E-I population as damped oscillations controlled by the parameters ω, γ, α. **B**. Serialization of an MNIST digit into an sMNIST stimulus and feeding the stimulus into an RNN with trainable parameters $\mathbf{W}^{\text{ih}}, \mathbf{b}^{\text{ih}}$ (input weights and biases), $\mathbf{W}^{\text{hh}}, \mathbf{b}^{\text{hh}}$ (recurrent weights and biases), and $\mathbf{W}^{\text{ho}}, \mathbf{b}^{\text{ho}}$ (readout weights and biases). For sMNIST, all networks are trained using a readout after 784 time steps. **C**. Test accuracy on sMNIST attained for different RNN architectures as a function of training steps over 100 training epochs (left) and maximal test accuracy after 100 training epochs as a function of trainable system parameters (right). Lines show mean accuracy over 10 network instances with random weight initialization, shaded areas standard deviation. Note the logarithmic scale in the number of training steps. **D**. Distributions of maximum test accuracy on sMNIST of HORN^{h} networks (32 nodes, 10 training epochs) with variable parameter values ω (natural frequency), γ (damping factor), α (excitability). Each panel marginalizes the values for the remaining parameters. Lines show median and maximum, shaded areas indicate 25% and 75% percentiles. Note that the maximal network performance peaks around a value of $\omega = 2\pi/28$ (dashed vertical line, see main text). **E**. Median test accuracies of HORN^{h} networks as in E. Each matrix entry shows the median test accuracy (color scale on the right) for the marginalized remaining parameter. **F**. Maximal test accuracy of different RNNs architectures (10^4 trainable parameters) after training on noisy sMNIST for 100 epochs as a function of training noise level σ^2_{train}. Colors as in D. **G**. Mean test accuracy of different network architectures trained on noisy sMNIST with $\sigma^2_{\text{train}} = 1$ (dashed vertical line) as a function of test stimulus noise level σ^2_{test}, calculated over $n = 1000$ test stimuli. Colors as in D.

It is important to note that already a single DHO node possesses intriguing computational capabilities that are not accessible to non-oscillating nodes such as leaky integrators. DHOs can gain-modulate an input signal in a nonlinear way as a function of its frequency profile and encode stimulus information in the oscillation phase (fig. S1C and Supplementary Information). Moreover, the combination of the

input nonlinearity with the biologically inspired feedback connections endow the DHO nodes with a dynamical repertoire that goes far beyond that of a classical damped harmonic oscillator. It allows nodes to express self-sustained oscillations and to resonate at fractional harmonics of their natural frequency (see fig. S2, S3, S4 and Supplementary Materials).

The natural frequency parameter ω defines a frequency band in which the DHO node selects and amplifies signals (see "Receptive Fields" below). Conceptually, the DHOs implemented in our networks can be understood as antennas, which process signals by modulating their amplitude, frequency, and phase in a non-linear way, as determined by the unit parameters. Furthermore, any input is turned into a wave pattern, that propagates throughout the whole network, allowing for interference. Importantly, their oscillatory dynamics enables DHO units to perform feature extraction through resonance rather than solely through the selective recombination of convergent input connections as is the case in many other RNN architectures. Single nodes or groups of nodes can encode information in distinct frequency bands, and complex networks allow partial information flow between nodes residing in different frequency bands. Together, the nodes make the HORN an anisotropically coupled, nonlinear, analog medium for information processing that computes with wave representations of stimuli and their interference patterns.

Homogeneous networks

We first consider homogeneous HORN (HORN^{h}) networks with all-to-all connectivity, no conduction delays and identical parameter values ω, γ, α, for all nodes. As cortical circuits operate in a balanced state where excitatory and inhibitory drive cancel in the mean (E/I balance), emphasizing the significance of fluctuations, we chose to couple DHO units on their velocity term $\dot{x}$ (see Methods).

With one exception (see below, "Geometric input drives self-organization"), stimuli were presented to the networks as time series. To transform the MNIST images used for benchmarking into

time series, henceforth addressed as sMNIST stimuli, intensity values were collected in scan-line order from top left to bottom right of the images (see Fig. 1B and Methods).

The networks were initially trained for classification of sMNIST stimuli in a supervised way using the back propagation through time (BPTT) algorithm (see Methods). For the sMNIST data set, a grid search was performed on the parameter values ω, γ, α to find a constellation that resulted in high classification performance (Fig. 1D,E, Table S1, and Methods).

Optimal parameter values turned out to be insensitive to network size (data not shown) and networks with low values of the damping factor γ tended to perform best (Fig. 1D,E). Low damping factors put the system in a highly oscillatory state and enhance the memory spans of individual DHOs as well as of the whole network (Fig. 1D,F). The excitability parameter α had no strong effect on performance, intermediate values between 0.2 and 0.4 resulting in the best performance (Fig. 1D,E). The optimal values of the natural frequency parameter ω were around the value $\omega_v = 2\pi/28$ which corresponds to the fundamental frequency of the time series resulting from a straight vertical line in the 28x28 MNIST pixel space (see Fig. 1D,E and Methods). As DHOs resonate with input frequencies around ω_v, this choice of ω enabled the nodes to extract and retain information contained in time series resulting from continuous lines of different slopes. The value ω_v also coincides with a peak in the variance of power spectral densities (PSDs) calculated on a representative set of sMNIST samples (fig. S5). Thus, by adjusting ω the network can be adapted to the statistical regularities of sMNIST stimuli and this setting of priors enhances performance by allowing feature extraction through resonance rather than through selective recombination of convergent input connections. To the best of our knowledge, this strategy of feature extraction has not been proposed previously and crucially depends on units or microcircuits with the propensity to oscillate.

To our surprise these simple homogeneous HORN networks substantially outperformed state-of-the-art gated RNN architectures with respect to learning speed (Fig. 1C, fig. S6), absolute performance, and noise tolerance at the same number of learnable parameters. This superiority was partic-

ularly pronounced in the region of low parameter counts (see Fig. 1C). When making the task more difficult by decreasing signal-to-noise ratios or increasing the number and similarity of stimulus patterns, the performance differences between HORNs and the other networks tended to increase further (fig. S6, S7, S8, S9). HORNs were extraordinarily noise resilient and showed only a gradual decline in task performance with increasing noise levels (Fig. 1F). Furthermore, the stimulus representations in HORNs were robust to a mismatch of noise characteristics between training and inference runs (Fig. 1G). This robustness can be explained by the strong attenuation of high-frequency signals due to the nonlinear frequency dependent gain modulation of input signals at each DHO node (see below "Receptive Fields"). In particular, this remarkable noise tolerance persists even when HORNS are trained on shuffled sMNIST stimuli (psMNIST), in which the dominance of the low-frequency information characteristic for the sMNIST stimuli is removed (fig. S10, S5). As both biological and artificial systems have to be able to learn from noisy stimuli and robustly detect, classify, and process stimuli even under changing noise characteristics, this noise resilience constitutes a very attractive property of HORNs.

We identified several reasons for the high performance of the HORNs. (i) The kinetics of each DHO node turn arbitrary input into harmonic oscillations, thereby structuring the networks' state spaces, and allow for dynamic phenomena such as fading memory, resonance, entrainment, fine-scale synchronization, phase shifts, and desynchronization both on the single node and on the network level which can be exploited for learning. (ii) "Innate" preference for stimulus features (controlled by the values of ω, γ, α) allowed individual nodes to efficiently extract and encode stimulus features already in an untrained network by means of resonance (iii) The discretization scheme used for the oscillator differential equations introduces temporal residual connections that stabilizes gradient propagation in BPTT learning. Furthermore, the oscillating dynamics of each DHO node temporally modulate gradients, implementing a kind of time-varying "attention" that can be used to recall important past information (see Methods and fig. S11).

As expected, before learning the dynamics of homogeneous HORNh networks is dominated by large-scale synchronization among the nodes (Fig. 2A,C). As learning progresses, global synchronization decreases (Fig. 2C), which enhances the dimensionality of network dynamics (Fig. 2D). This reduction in global desynchronization is accompanied by the emergence of complex, spatiotemporally structured correlation and higher-order synchronization patterns that are stimulus specific, well segregated in the high dimensional activity landscape of the network, and well classifiable by a linear read-out (Fig. 2B, 2C). This learning dependent increase in the complexity of network dynamics resembles the shift from global, large amplitude low frequency oscillations towards more local, low amplitude and broad band oscillations that occurs during early stages of experience dependent maturation of cortical networks (reviewed in (*2*)).

Heterogeneous networks

The structural and functional organization of mature cortical networks is characterized by heterogeneity (*33, 34*). However, it is unclear whether this variability plays a functional role. To test whether increasing network heterogeneity facilitates learning, we simulated non-homogeneous HORNs ($HORN^n$) in which each node had a different natural frequency, damping coefficient, and excitability (see Methods). As expected, heterogeneous HORNs responded already in the untrained state with more complex and less globally synchronized patterns (Fig. 2A). As in the homogeneous case, global synchrony decreased further as learning progressed (Fig. 2C). Here again, the decrease in global synchrony led to a significant increase in the dimensionality of the dynamics (Fig. 2D), but the dimensionality of the dynamics was much higher compared to their homogeneous counterparts already in the first training steps.

A comparison between homogeneous and heterogeneous HORNs revealed superior performance of the latter with respect to learning speed in data sets with more complex spectral structure such as

permuted sequential MNIST stimuli (psMNIST, S7), and noise tolerance (Fig. 1F,G). Note that in the case of sMNIST digits that have a reduced complexity of their signal statistics, the final performance of heterogeneous HORNs was found to be on par with the one obtained by homogeneous HORNs whose parameters had been optimized for the specific stimuli (Fig. 1C, S6). Thus, heterogeneity allows one to obtain networks with high task performance without the need to find optimal parameter configurations. Such a parameter optimization is computationally expensive in artificial systems and likely unfeasible for biological systems.

When datasets possessed more complex signal statistics such as the permuted MNIST digits (fig. S10), heterogeneous HORNs performed better than their parameter-optimized homogeneous counterparts (fig. S7). Generally, the advantages of heterogeneity were especially prominent for larger networks, more difficult classification problems, and higher levels of stimulus noise (fig. S6, S7, S8, S9). We suggest that the pronounced heterogeneity of natural systems is not the result of imprecision, but rather reflects an efficient solution to the problem of robustly processing noisy sensory stimuli with highly varying spectral characteristics.

Because heterogeneous HORNs produced highly structured response landscapes already in the untrained state, we hypothesized that they might also serve as efficient reservoirs (*35*). We found that in contrast to RNNs composed of gated units that often have chance level stimulus decodability by a linear readout in the untrained state, heterogeneous HORNs already allow for high levels of decodability even in an untrained state (fig. S12). To determine the reservoir properties of HORNs we simulated heterogeneous HORNs with random but fixed recurrent weights and trained only the input and readout weights with BPTT. Although these reservoir HORNs showed a performance drop compared to the corresponding regular HORN in which recurrent connections were plastic (fig. S13), we hypothesize that heterogeneous HORNs could serve as efficient reservoirs, especially with increasing number of nodes.

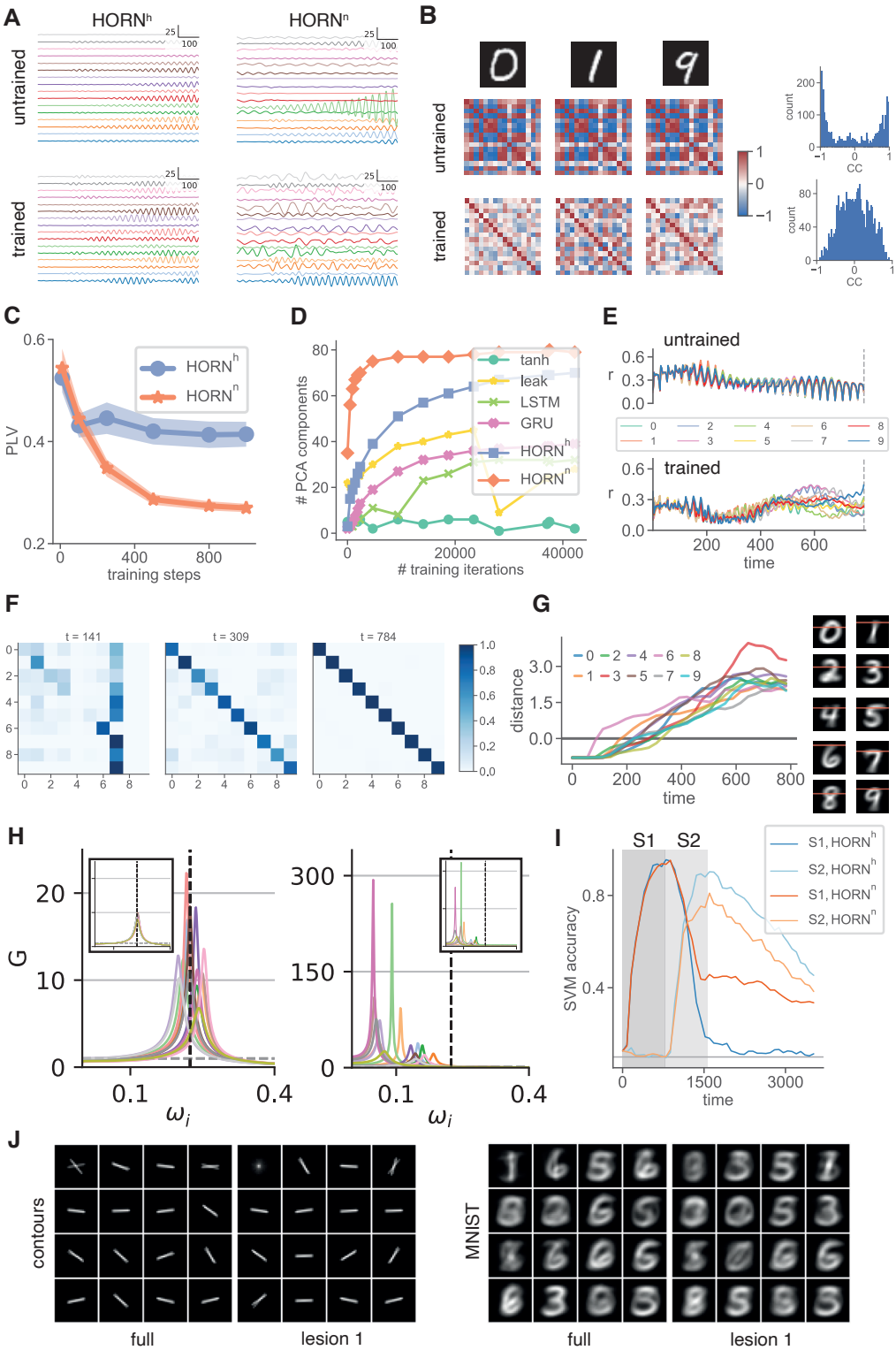
A
HORN^h
HORN^n
untrained
trained
B
untrained
trained
count
CC
C
PLV
training steps
HORN^h
HORN^n
D
PCA components
training iterations
tanh
leak
LSTM
GRU
HORN^h
HORN^n
E
untrained
trained
r
time
F
t = 141
t = 309
t = 784
G
distance
time
H
G
ω_i
I
S1
S2
SVM accuracy
time
S1, HORN^h
S2, HORN^h
S1, HORN^n
S2, HORN^n
J
contours
MNIST
full
lesion 1
full
lesion 1

Fig. 2. Dynamics and computational principles in HORNs. **A**. Network activity elicited by an sMNIST stimulus (digit 0) of 16-node HORNs trained on sMNIST, before (top row) and after (bottom row) training. Left column HORN^{h}, right column HORN^{n}. Note the highly irregular temporal dynamics of HORN^{n} in the trained state. **B**. Pairwise cross-correlation coefficients of node activity for different input stimuli in HORN^{h} (16 nodes) before (left, top row) and after training (left, bottom row) on sMNIST for 10 epochs. Distribution of cross-correlation coefficients before (right, top) and after (right, bottom) training. **C**. Pairwise phase locking value (PLV) calculated on $n = 100$ test stimuli as a function of training steps for a homogeneous (blue) and heterogeneous (orange) HORN network (16 nodes) trained on sMNIST. Lines show mean, shaded region indicates standard deviation. **D**. Number of PCA components explaining 99% of the variance of observed network activity as a function of training steps for different RNN architectures (10^4 trainable parameters). PCA performed on network activity data resulting from 10000 test stimuli (see Methods). **E**. Dynamics of mean Kuramoto order parameter r of HORN^{h} (16 nodes) activity trained on sMNIST in the untrained network (top) and after training for 10 epochs (bottom). Lines show mean values per digit class. The HORN learns to generate dynamic, stimulus-specific patterns of higher-order synchrony. **F**. Normalized confusion matrices of linear SVMs trained on stimulus classification from activity of a HORN^{h} (93 nodes, trained on sMNIST for 100 epochs) activity at different time points of the stimulus presentation period. Matrices show relative frequencies the number of predictions obtained over 10^4 samples. The vanishing of the non-diagonal entries (indicating misclassified stimuli) shows that the network performs evidence accumulation over the stimulus presentation time. **G**. Mean distance to SVM decision boundary for SVMs as a function of presentation time for the dataset depicted in F. The color of the curves refer to different digits (insert). Note the steady accumulation of evidence. Right: Class-averaged MNIST digits with the image row corresponding to the decision boundary crossing of the SVM colored in orange. **H**. Intrinsic RFs (gain functions G) of nodes as a function of input frequency ω_i of 16-node HORNs before (insets) and after training on sMNIST. Homogeneous (left) and heterogeneous (right) HORN variants. Gain values $G > 1$ indicate amplification. Horizontal dashed lines indicate $G = 1$, vertical dashed lines indicate the frequency $\omega_v = 2\pi/28$ that corresponds to a straight vertical bar in pixel space. Colors indicate node identity. **I**. Accuracy of stimulus decoding by linear SVMs trained on the activities of different HORN networks (93 nodes) that were stimulated with a sequence of two temporally segregated stimuli S1, S2

(shaded areas indicate stimulus timing). Light and dark colored curves indicate SVM decoder performance for S1 and S3 for a homogeneous ($HORN^h$) and non-homogeneous ($HORN^n$) network, respectively. Note the longer persistence and superposition of stimulus-specific information in the $HORN^n$. **J**. Effective RFs of a $HORN^h$ (16 nodes) when mapped with contour stimuli from LSDSa (left) and sMNIST digits (right). Both sides show the ERFs of the full network and the ERFs after the deletion (lesion) of the first node. Note the reconfiguration of most of the ERFs.

Fading memory and evidence accumulation

Due to the reverberation of activity, recurrent networks exhibit fading memory, which allows them to integrate information about successive events. Electrophysiological recordings from neurons in the visual cortex have shown that population activity in the visual cortex exhibits fading memory and is capable of simultaneously representing classifiable information on the identity of successively presented stimuli, including their sequence order (*36*). HORNs share this ability to superimpose information about successively presented stimuli. Following the sequential presentation of two different stimuli, a linear classifier (support vector machine, SVM) can decode the two stimuli from the same segment of reverberating activity (Fig. 2I).

As heterogeneous HORNs possess more diverse memory timescales than homogeneous networks, their ability to represent simultaneously information about temporally segregated stimuli is superior, yet another functional advantage of network heterogeneity. Fading memory also allows for gradual accumulation of stimulus-specific information from time series. The response patterns of HORNs evolve over time as a result of recurrent network reverberations. To determine the point in time at which the network converges to states of maximal stimulus specificity, a linear SVM was trained on stimulus classification using activity data from a trained homogeneous HORN at different time points

throughout the stimulus presentation period. We find that the network accumulates evidence, with its dynamics allowing progressively better decoding of stimulus identity as the network approaches the read-out time $t = 784$ on which it was trained (Fig. 2F,G). Furthermore, we find that most sMNIST stimuli can be decoded well before the time step $t = 400$ (Fig. 2F), i.e. in many cases already when only the upper half of the digit has been presented to the network.

Receptive fields

To better understand the principles of computation in HORNs and how stimulus specific activity patterns emerge during training, we investigated how learning changed the response properties of both individual nodes and the entire network. We expected analogies to the experience- or learning dependent modifications of receptive fields (RFs) in natural systems (*37*).

Each DHO node in a HORN has a gain curve $G(\omega_i)$ that describes how the node modulates the amplitude of an input signal as a function of the input signal frequency ω_i (Fig. 2H). The shape of G is determined by the values of ω, γ, and the self-connectivity terms v and W_{ii}, the diagonal terms of the recurrent weight matrix (see Methods, fig. S4 and Supplementary Information). We call G the *intrinsic receptive field* (IRF) of the node because it defines the frequency band in which the node shows feature selectivity. During learning, the adjustment of the self-connection weights v and W_{ii} alters G and thereby drives changes in the IRF of each node, allowing the node to improve its selectivity for stimulus features useful for performing a given task (Fig. 2H). In homogeneous HORNs where the values of ω, γ, α are the same for all nodes, the IRFs of all nodes only differ due to learning-induced changes in the values of the feedback parameters v and W_{ii} that vary between nodes (Fig. 2H, left). For heterogeneous networks, the IRFs of the untrained network already cover a larger portion of the frequency space, with nodes tuned to a wider variety of features in the frequency space (Fig. 2H, right). Note that the value of ω not only influences the IRF, but also sets the frequency band

in which the node codes. This increases the diversity of frequency bands available for processing in heterogeneous HORNs.

For comparison with natural systems, we mapped the receptive fields of the nodes in the same way as is common practice in electrophysiological studies (*38*). We examined which stimuli activate a node most strongly and address the so-defined RF as *effective RF* (ERF). In natural systems, the structure of the ERFs varies greatly depending on the nature of the test stimuli. When mapped with very simple isolated stimuli, neurons in sensory cortices exhibit canonical RFs (*38*), but these are no longer detectable when more complex stimuli are used that induce large-scale network interactions (*39*). The reason is that responses depend not only on the stimulus but also on network connectivity and dynamics. The same dependence of ERFs on test stimulus complexity also holds for trained HORNs. We determined the ERFs of nodes in our models with two canonical choices of mapping stimuli: (i) simple line segments of different orientations, as commonly used for RF mapping in visual experiments (*38*) (fig. S14), and (ii) the original MNIST digits on which the networks were trained, as an example of complex natural stimuli (*39*).

To measure ERFs, we simulated a heterogeneous 16-node HORN^{n} trained on MNIST and calculated for each node the mean stimulus that resulted in maximal node activation (see Methods). When stimulated with simple line stimuli, we find simple orientation-selective ERFs that closely resemble the RFs of neurons in the primary visual cortex (Fig. 2J, left). When mapped with complex stimuli, some nodes exhibited specificity for particular features of the MNIST digits (Fig. 2J, right). Other nodes had ERFs that appeared to be unrelated to the stimuli in the training set, and yet others were inhibited.

To determine the influence of the collective network dynamics on individual ERFs, we silenced one node of the network. This resulted in an immediate reconfiguration of the ERFs of the other nodes, which was quite dramatic in some cases (Fig. 2J). This indicates that after learning, ERFs are to a large extent the result of dynamic interactions in the network. Thus, the ERFs of HORN nodes

undergo context-dependent dynamic modifications similar to those observed in natural systems (*40*).

To obtain a more quantitative description of the ERFs of the nodes in frequency space, we additionally stimulated the networks with harmonic sine-wave stimuli of different frequencies and determined the frequency responses of the nodes (fig. S15). As expected, in untrained networks most of the nodes resonated best to inputs with frequencies close to their preset natural frequencies. However, this preference changed substantially over the course of training with nodes developing complicated and hard to predict resonance patterns, confirming the strong influence of anisotropic network interactions on the structure of ERFs.

To assess the influence of the presence of feedback parameters v, w on network performance, we trained HORNs with and without DHO feedback connections on sMNIST and psMNIST and measured their task performance (fig. S16, S17). We found that even in the absence of feedback in both amplitude and velocity ($v = w = 0$), HORN performance only dropped slightly, but both feedback terms were needed to obtain the fastest learning speed and the highest test accuracy.

Delays

In the cerebral cortex, neurons interact through relatively slow conducting nerve fibers (0.5 to 10 m/s), which introduces considerable, widely scattered, distance-dependent coupling delays (*41*). To test the influence of introducing coupling delays on task performance, we started with a HORN^{h} and endowed all recurrent connections with uniformly distributed variable coupling delays $[1, d_{\text{max}}]$ (see Methods). This manipulation increased HORN performance in both maximal classification accuracy and learning speed on psMNIST (Fig. 3A). Increasing d_{max}, which results in greater heterogeneity, was found to increase task performance, and this gain of function increased with increasing values of d_{max} (Fig. 3A). Thus, as for the preferred oscillation frequencies, heterogeneity in conduction delays enables the generation of more diverse spatiotemporally structured activity landscapes in HORNs,

thereby increasing the dimensionality of the networks' state space. This again results in enhanced HORN performance, and alleviates the need for parameter tuning, in particular for datasets with complex spectral properties or changing noise characteristics. Hence, we hypothesize that the heterogeneity of coupling delays in natural systems is likely not a reflection of unavoidable imprecision, but developed in order to increase the dimension of the available coding space and to make the networks more robust to changes in spectral properties of stimuli.

Multi-layer networks

In the mammalian cerebral cortex, sensory signals are processed in hierarchically organized cortical areas that are reciprocally coupled (*42*). To investigate the potential benefits of distributed multi-stage processing, we generated two-layer homogeneous HORNs consisting of a "lower" (L1) and an "upper layer" (L2), each consisting of 32 nodes, $\text{HORN}^{\text{h}}_{32,32}$), and implemented sparse reciprocal connections between the layers (Fig. 3B). The input signals were presented to L1 as before, and the result was read out at the nodes of L2. We performed a grid search on the inter-layer feed-forward and feed-back connection probabilities f_{F}, f_{B}, as well as the scaling factors f_ω, f_γ that controlled the scaling of the parameters ω, γ of L2 with respect to L1. For each parameter configuration, we trained a $\text{HORN}^{\text{h}}_{32,32}$ and assessed task performance by means of the maximal test accuracy achieved on psMNIST in 10 training epochs. For all values of each parameter, there usually exists a parameter configuration that results in a high-performing two-layer network (Fig. 3B). Furthermore, many two-layer $\text{HORN}^{\text{h}}_{32,32}$ networks outperformed a single-layer HORN^{h} at a comparable number of trainable parameters (see Fig. 3C-E). In particular, higher task performance resulted when both the preferred oscillation frequencies and the damping coefficients were lower in the upper layer (Fig. 3B, bottom row). In this setting, the dynamics of the faster lower layer L1 are partly opaque to the slower upper layer L2 due to the stronger attenuation of high frequencies in L2.

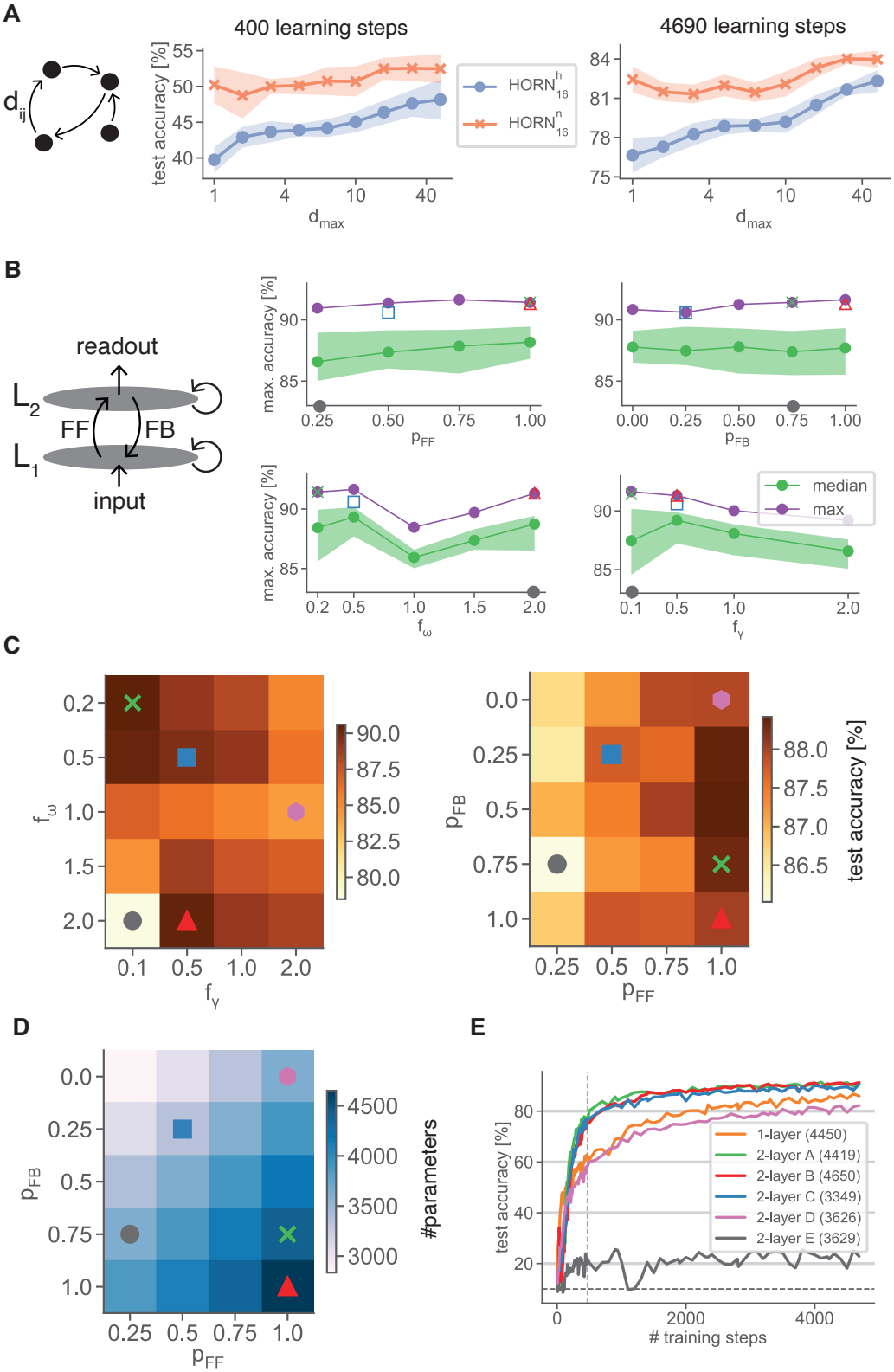
A
400 learning steps
4690 learning steps
test accuracy [%]
d_{max}
d_{ij}
$HORN^h_{16}$
$HORN^n_{16}$
B
readout
input
L_2
L_1
FF
FB
max. accuracy [%]
p_{FF}
p_{FB}
f_ω
f_γ
median
max
C
f_ω
f_γ
p_{FB}
p_{FF}
test accuracy [%]
D
p_{FB}
p_{FF}
#parameters
E
test accuracy [%]
training steps
1-layer (4450)
2-layer A (4419)
2-layer B (4650)
2-layer C (3349)
2-layer D (3626)
2-layer E (3629)

Fig. 3. Biologically-inspired extensions to HORN networks. **A**. Performance of HORNs with connection delays. Maximum test accuracy on psMNIST after 400 training steps (left) and after 4690 training steps (right), respectively, as a function of maximal synaptic delay d_{max}. For each network, connection delays were sampled from a uniform distribution $U([1, d_{\text{max}}])$. A network with $d_{\text{max}} = 1$ corresponds to a regular HORN. Lines show mean performance over 10 randomly initialized networks, shaded areas standard deviation. **B**. Two-layer HORN^{h} (32 +32 nodes) performance as a function of the 4 parameters $p_{\text{FF}}, p_{\text{FB}}$ (fraction of feed-forward and feed-back connections present, upper panels), f_ω, f_γ (scaling factor of ω and γ between the first layer and the second layer, lower panels). Curves represent median and maximal values of marginal distributions of the maximal accuracy on psMNIST attained during training for each value of the 4 parameters. Colored symbols refer to five exemplary networks whose parameter constellations are shown in C. Note the drop in performance when L1 and L2 have the same preferred oscillation frequency and that increasing connection sparsity does not result in a strong performance drop (gray circle marked on axes lies outside of the accuracy range displayed). **C**. Median HORN^{h} performance (color scale on the right) for the different pairs f_ω, f_γ values (left) and p_{FF}, p_{FB} (right). Symbols refer to the same networks as in B. Note that different parameter combinations support equally high performance. **D**. Number of trainable parameters of HORN networks (color code on the right) as a function of various combinations of p_{FF}, p_{FB}. Symbols refer to the same networks highlighted in B and C. Note that certain network configurations show high performance despite lower numbers of trainable parameters (green cross, red triangle, blue circle). **E**. Evolution of accuracy as a function of learning steps of the 5 exemplar two-layer networks marked by symbols in B, C, D (colors of curves correspond to symbol colors) in comparison to a single layer HORN^{h} with a comparable number of trainable parameters. Number of trainable parameters in parentheses.

Yet, activity in the slower upper layer is able to entrain the nodes in the faster lower layer, potentially supporting processes such as top-down attention (see Receptive Fields). Layered networks can therefore operate in different frequency bands and perform parallel analyses of input patterns at different temporal scales. Interestingly, we found that a two-layer network in which the separation of frequency

bands across layers breaks down fails to learn (Fig. 3E). In this case, the computations in the different layers are not well separated into frequency bands, which seems to hinder information processing. Together, these findings provide a plausible explanation for the fact that the preferred oscillation frequencies of cortical areas in the vertebrate brain decrease from low to high levels of the processing hierarchy (*11*). Interestingly, according to basic physics waves with slower frequencies tend to travel over longer spatial distances, in our case over a larger number of network nodes. Higher areas of the cerebral cortex integrate information from increasingly diverse, often spatially remote processing streams as reflected by their large, often polymodal and multi-selective receptive fields. Operating at lower oscillation frequencies would allow these areas to integrate information over larger temporal and spatial scales, favoring holistic processing of information.

Learning priors

In the primary sensory cortices of the mammalian brain, information about statistical regularities of the natural environment is stored in the architecture and the synaptic weight distribution of both feedforward and reciprocal connections. This translation of statistical contingencies of features into functional architectures is achieved during early development by experience-dependent synaptic pruning and continues throughout life through perceptual learning (*2, 43*). The shaping of feedforward connections leads to canonical feature-selective RFs of the nodes, and the selective stabilization of the recurrent connections among nodes leads to preferential coupling of nodes (representing, for example, cortical columns) tuned to features that frequently co-occur in natural environments. In the visual modality, this internal model of the world captures statistical regularities of the visual environment (Gestalt criteria) and is used as a prior for perceptual grouping and scene analysis (see (*2*) for a review).

To test how the installation of priors in HORNs influences their learning and task performance, we

first installed a set of canonical priors by training a heterogeneous HORN^n (93 nodes) to discriminate simple elongated contours of different orientations placed at random locations in the MNIST pixel matrix (see fig. S14 and Methods). We then froze the weights of the input and recurrent connections and trained only the readout layer on the sMNIST classification task with BPTT. We found that the installation of canonical priors allowed the pretrained HORN to generate well-separable representations of the MNIST digits. These representations were directly classifiable with high stimulus specificity by training only the readout layer (fig. S18). When after the installation of the priors, we only froze the weights of the input connections but allowed the pre-structured recurrent connections to participate in further learning, we observed a further increase of the learning speed (fig. S18). In summary, these pre-structured HORNs required substantially fewer training steps than the HORNs trained directly with the MNIST digits to reach the same performance levels. This indicates that a priori knowledge of the statistical contingency of features substantially facilitates and accelerates learning classifiable object representations. We assume that similar processing principles are realized in cortical networks and that these contribute to the one-shot learning capability that is a hallmark of natural systems. We noticed that heterogeneous networks required fewer training steps to reach high performance as their size increased (fig. S7, S8) and predict that sufficiently large and heterogeneous HORNs should be capable of single shot learning.

To test how trained HORNs process unfamiliar stimuli, we trained a HORN on sMNIST data and then presented shuffled sMNIST stimuli (psMNIST) to the trained networks. The network separated well different classes of "natural" sMNIST stimuli, but not the unfamiliar psMNIST stimuli (fig. S19). This is reminiscent of recent experimental findings in the visual cortex of behaviorally trained monkeys. These revealed that population vectors of multi-unit activity contained more stimulus-specific information when evoked by natural scenes than by manipulated stimuli, in which part of the statistical regularities of natural scenes were removed (*44*). Taken together these findings emphasize the beneficial effects that priors have for the orthogonalization of object representations in both artificial

and natural neuronal networks.

Hebbian Learning

Experimental evidence suggests that biological neuronal networks learn to represent characteristic feature constellations of perceptual objects through Hebbian modifications of synaptic gain (*9*); connections strengthen between nodes that repeatedly engage in correlated activity and weaken between nodes exhibiting mostly decorrelated responses. Thus, neurons that encode features that frequently co-occur become more strongly coupled and, therefore, self-organize to form a functionally coherent stimulus-specific assembly (*45*).

To test whether gradient-based training of HORNs produces weight distributions compatible with those predicted by Hebbian learning principles, we compared recurrent weight distributions and the corresponding correlation structure of responses before and after learning. To our surprise, we found that the changes in synaptic weights of HORNs induced by BPTT are similar to those predicted by a Hebbian mechanism (Fig. 4A,C). The BPTT learning algorithm apparently capitalizes on the stimulus-specific correlation structure of network activity and enhances those connections that induce correlation patterns characteristic of a particular stimulus. As expected, heterogeneous HORNs could exploit stimulus-specific correlation structures right from the beginning of training (Fig. 4C). Homogeneous HORNs, by contrast, must first learn to desynchronize to increase the dimensionality of their state space (Fig. 2A,B).

To investigate whether BPTT could be substituted by correlation-based learning, we implemented simple unsupervised additive (conventional) Hebbian, as well as anti-Hebbian rules for the activity dependent modifications of the recurrent connections (see Methods). During training, we disabled BPTT learning on the recurrent connections while keeping it active for the input and readout connections. As a performance baseline, we trained a HORN in which the recurrent connections were kept

fixed, and only the input and readout connections were trained. In the case of homogeneous networks, the application of a fully unsupervised *anti*-Hebbian rule resulted in a task performance surprisingly close to that attained when the recurrent connections were trained by the fully supervised BPTT rule (Fig. 4B). The reason is that the anti-Hebbian rule first promoted selective desynchronization of the homogeneous network, allowing for the later build-up of stimulus-specific correlation structures. For heterogeneous networks that already express more decorrelated activity in the untrained state (Fig. 4C), the conventional Hebbian rule but not the anti-Hebbian rule increased performance above baseline (Fig. 4D). The reason is that the Hebbian modifications enhanced the already existing stimulus-specific correlation structure, whereas the anti-Hebbian rule could not induce further modifications of the already decorrelated activity patterns. In conclusion, the propensity of the network nodes to engage in oscillatory activity and the resulting amplification of specific correlations by resonance enabled the Hebbian mechanism to install stimulus specific priors in the network architecture and to thereby orthogonalize representations in the high dimensional state space.

Taken together, the results provide a proof of principle that unsupervised Hebbian learning at the level of recurrent connections in HORNs supports the segregation of stimulus-specific dynamic states and thus facilitates their classification (Fig. 4B,D).

In natural systems conventional Hebbian rules prevail, but there is also evidence for anti-Hebbian mechanisms (*46*). Our findings are compatible with this evidence because mature natural systems bear more similarities with heterogeneous than homogeneous HORNs (*2*) and thus are likely to profit more from the conventional than from the anti-Hebbian mechanism. It would be interesting to examine whether sign changes of Hebbian learning rules occur in biological systems during development, and if so, whether they are related e.g. to the known conversion of Gabaergic synapses from an excitatory to an inhibitory effect (*47*).

Geometric input drives self-organization

To test how HORNs process simultaneous geometric input as encountered by the visual system, we trained networks that receive spatially organized input in the form of MNIST digits (Fig. 5A). We endowed each node of a homogeneous HORN^{h} with a receptive field (RF) of 2x2 pixels so that the 28x28 pixel matrix of an MNIST digit was covered by the RFs of a network with 196 nodes (Fig. 5A). To provide input to the network, we activated each node for one time step at an intensity corresponding to the sum of the intensity values of the pixels within its RF (Fig. 5A). We then gave the network 150 time steps to process the stimulus before performing a read-out (Fig. 5B). For training, we used the BPTT algorithm as before, started with an all-to-all random connectivity, and kept the weights of the input layer fixed, while the recurrent and readout weights were plastic. After training for 10 epochs, the classification accuracy on the test set was found to be around 90% (data not shown). When stimulated, each of the simultaneously activated nodes responded with a damped oscillation (Fig. 5B). These activities spread throughout the network and led to complex interference patterns and traveling waves. The direction and shape of the waves differed for responses evoked by different stimuli (Movies S1, S2). Traveling waves are a robust and well-studied phenomenon in biological networks (*17*) and many hypotheses concerning their functional role have been proposed (*18*). We expect that HORNs will also be helpful for systematic investigations of these phenomena.

The temporal evolution of responses and of learning-induced changes bear similarities with observations in the mammalian visual cortex. The brief and direct activation of the nodes covered by the stimulus was followed by a relatively low activity period that then gradually increased to levels far above those of the initial stimulus-driven response (Fig. 5B). Similar dynamical effects have been observed in the visual cortex of anesthetized cats in response to flashed stimuli (*45*). Responses were amplified by reverberation and the decodability of the resulting dynamic state increased due to better segregation of the stimulus-specific principal components of the population vector (*45*).

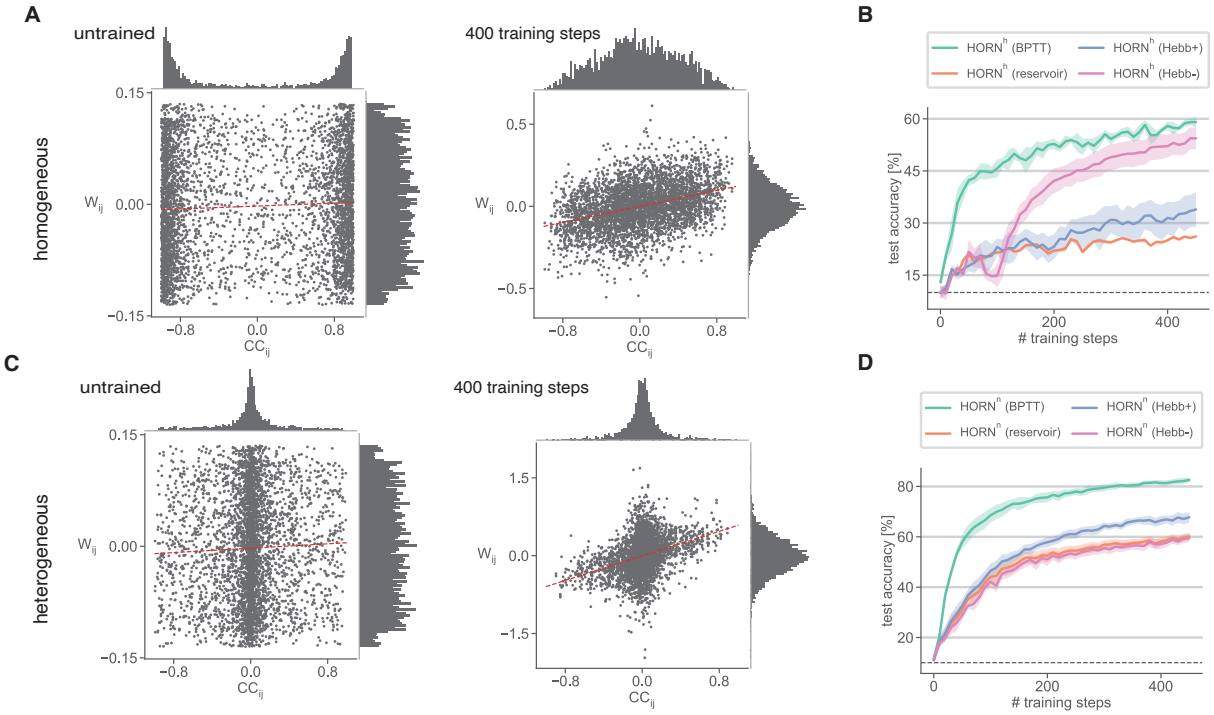

Fig. 4. Backpropagation learning in HORN networks results in Hebbian-like weight changes. **A**. Scatter plots of connection weights $\mathbf{W}^{\mathrm{hh}}_{ij}$ and mean cross-correlation coefficients CC_{ij} of node activities of a homogeneous HORN^{h} (64 nodes) before training (left) and after training on psMNIST for 500 training steps (right). CC_{ij} computed over 100 samples. Linear regression lines in red. Marginal distributions of $\mathbf{W}^{\mathrm{hh}}_{ij}$ and CC_{ij} are shown on the right and top, respectively. Note the bimodal distribution of correlation coefficients with modes around -1, 1 in the untrained state and the more decorrelated network activity as a result of learning. **B**. Performance of a homogeneous HORN^{h} (64 nodes) network as a function of training steps when instances of the same network are trained with correlation-based Hebbian (suffix "Hebb+") or anti-Hebbian (suffix "Hebb-") learning rules compared to instances trained with BPTT (suffix "BPTT") and when $\mathbf{W}^{hh}$ was fixed (suffix "reservoir"). The input and readout parameters are trained with BPTT for all instances. Curves show mean performance over 10 network instances with random weight initialization, shaded areas standard deviation. Note the strong performance of the anti-Hebbian rule for this initially highly synchronized network. **C**. Scatter plots of the connection weights and cross-correlation coefficients of a heterogeneous HORN^{n} as in A. Note the more decorrelated activity compared to a HORN^{h} already in the untrained state. **D**. Performance of different

instances of a heterogeneous HORN^n (64 nodes) network as a function of training steps when trained with different learning rules. Learning rules as in B. Note the baseline (reservoir HORN) performance of the anti-Hebbian rule in this initially more desynchronized network (compared to the case in B) and the performance increase compared to baseline resulting from a Hebbian rule.

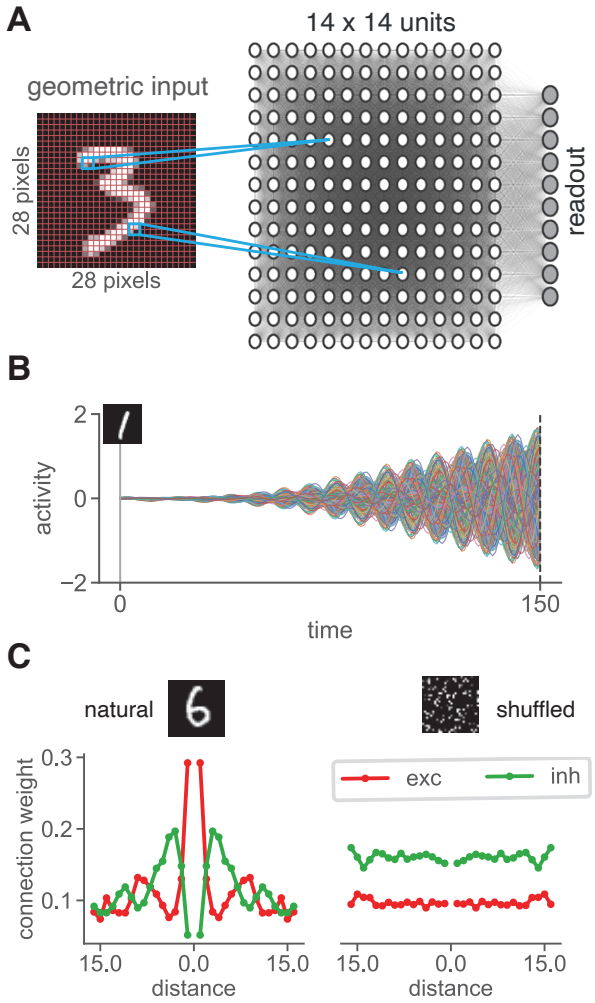

Fig. 5. Self-organization in HORNs driven by geometric input. **A**. Schematic of geometric MNIST input provided to an all-to-all connected homogeneous HORN (196 nodes). Each DHO node receives input from its 2x2 pixel receptive field. Input weights are constant and fixed, recurrent and readout weights are trainable. Readout nodes are connected to the entire HORN. **B**. Example network dynamics of a HORN^{h} (196 nodes) stimulated as in A after training with BPTT for 2000 steps (achieved classification accuracy around 90% for both natural and shuffled stimuli). Input is flashed to the network during one time step at $t = 1$. Readout is performed after 150 time steps of network activity during which no input is given. Curves represent the activity of all nodes in the network (node identity is color coded). Note the increasing amplitudes of reverbrating activity and the drifts in phase of the oscillatory responses. **C**. Distance-dependent connection weights of excitatory (red) and inhibitory (green) connections in HORN^{h} networks after training with BPTT for 2000 steps on geometric stimuli. Stimuli were either natural (left) or shuffled (right) MNIST digits. Note the Mexican-hat-like connectivity structure among nodes in networks trained on natural stimuli and the lack of structure in the connectivity in networks trained with shuffled stimuli.

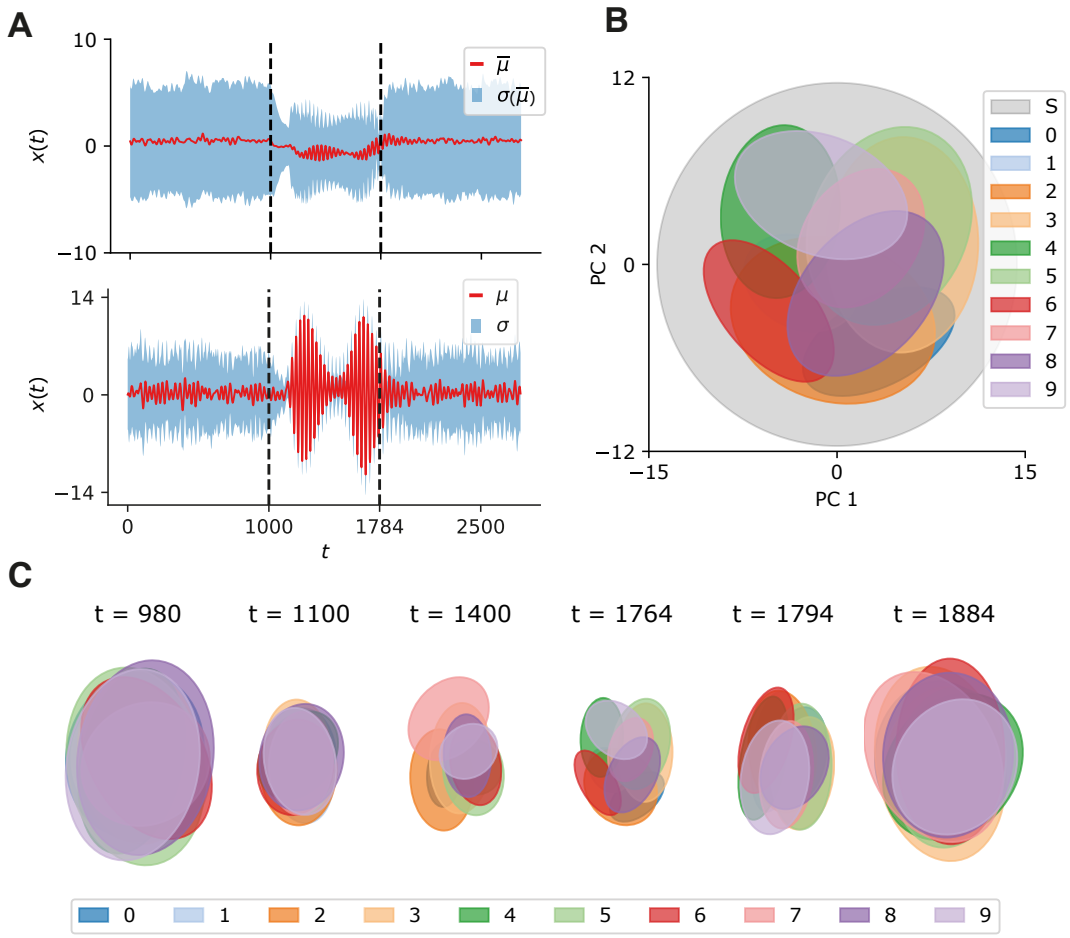

Fig. 6. Spontaneous and evoked activity in a heterogeneous HORN network of 16 nodes trained on sMNIST. **A**. Mean and standard deviation of amplitude dynamics computed over 500 samples of the sMNIST digit 0. Top: Mean amplitude $\bar{\mu}$ and standard deviation of mean amplitude $\sigma(\bar{\mu})$ computed for the entire network. Bottom: Mean and standard deviation of amplitude dynamics of one selected DHO node. Dashed lines mark stimulus onset ($t = 1000$) and offset ($t = 1784$), respectively. **B**. Sliding window analysis of PCs of network activity (window size: 20 time steps). Ellipses show 2σ confidence intervals, stimulus conditions are color-coded (S = spontaneous activity). **C**. Evolution of network dynamics in PC space under different stimulus conditions ($t < 1000$: spontaneous activity; $1000 \leq t \leq 1784$: sMNIST stimulus presentation; $t > 1784$: spontaneous activity). Ellipses show 2σ confidence as in C.

This state can be interpreted as the result of a highly parallelized search for the best match between sensory evidence and the priors installed by learning (*2*). The present results support the notion that this core function of predictive coding can be realized through self-organizing dynamic interactions in oscillatory recurrent networks (*2*). In our model, nodes coactivated by a flashed stimulus exhibit stimulus-locked synchronized oscillations. As the BPTT algorithm mimics Hebbian plasticity in HORNs, this predicts that during learning these synchronously active nodes should increase their mutual coupling. Analysis of learning-dependent circuit modifications revealed a shift from unspecific all-to-all connections to spatially restricted connections with a distant-dependent decay of coupling strength, following a Mexican hat-like shape (Fig. 5C, left). Such connectivity favors nearest-neighbor interactions and captures the essential structural feature of our MNIST stimuli, the continuity of their contours, and the spatial vicinity of activated nodes. Thus, the weight distributions of the coupling connections in the HORN evolved to capture the Gestalt criteria of continuity and vicinity that natural systems are also known to exploit for perceptual grouping. During testing (recall) of the trained model, a particular stimulus again induces synchronized oscillations in a respective constellation of nodes and, due to the enhanced coupling of those nodes in the learned network, these now engage in resonance which leads to an increase in response amplitude. Thus, over the course of learning, the nodes activated by a particular stimulus become associated with one another by self-organizing interactions and form a cooperating, stimulus-specific assembly distinguished by synchronized and jointly enhanced responses. Such stimulus-specific synchronization has been observed among neurons in the visual cortex tuned to features that tend to be perceptually bound (*12,45*) and is at the core of the binding by synchrony hypothesis (BBS) (*24*). In conclusion, by exploiting the resonance properties of coupled oscillators, HORNs also reproduce this important feature of natural cortical networks.

As expected, such distant dependent connectivity patterns did not emerge when training HORNs on shuffled MNIST stimuli (fig. S10), although the networks achieved a comparable classification

accuracy on the test set (Fig. 5C, right). Here, other priors than spatial continuity and vicinity are installed in the architecture of the coupling connections.

Spontaneous and evoked activity

Cortical networks are spontaneously active and in sensory cortices, stimulation typically causes a reduction in variability and leads to the emergence of stimulus-specific substates (*48–50*). To model spontaneous activity in HORNs, we subjected each DHO node to random discrete jumps in its velocity by a given amount according to a Poisson process (see Methods).

After training a spontaneously active heterogeneous HORN on sMINST, we analyzed network activity before, during, and after stimulation with sMNIST stimuli (Fig. 6). During stimulation, spontaneous activity was replaced by the structured time series corresponding to the sMNIST digits. In this way, we mimicked protocols used in physiological experiments. Here, the unstructured resting activity of sensory organs, the main drive for spontaneous cortical activity, is temporarily replaced by stimulus-induced structured activity. Analysis of the activity of both single nodes and the entire network showed that the variance of activity was high before stimulation, decreased during stimulation, and then recovered to pre-stimulus levels after stimulus offset (Fig. 6A). Principal component analysis (PCA) revealed that the state space of spontaneous activity spans a large but confined space that comprised the subspaces of stimulation-induced stimulus-specific response vectors (Fig. 6B). The reason for this confinement of the spontaneous state is that the connectivity of the trained HORN is not random, but highly structured through learning. In trained HORNs, the weight distribution of the recurrent connections reflects the statistical regularities of the learned sMNIST stimuli (see "Learning Priors" above). This structure of the coupling connections in turn shapes the dynamics of the network. Thus, spontaneous activity can be considered as a superposition of fragments of learned stimulus-specific representations. The temporal evolution of network dynamics in PCA space shows

that network dynamics rapidly converges to a stimulus specific substate once a stimulus is presented and that this specificity fades after stimulus offset (Fig. 6C). At the same time, this collapse towards a stimulus-specific substate is associated with a reduction of variance of activity (Fig. 6A). Again, very similar, stimulation-dependent changes in network dynamics have been reported for natural networks (*2, 45, 48*).

Summary and outlook

The implementation of characteristic features of the mammalian cerebral cortex in simulated RNNs uncovered, to the best of our knowledge, a genuinely novel and extremely powerful computational principle, that exploits the unique dynamics of coupled oscillators.

An important step, inspired by experimental findings (*28, 51*), was the configuration of network nodes as damped harmonic oscillators. This boosted the performance of the networks on standard bench mark tests for pattern recognition and classification with respect to learning speed, noise tolerance and paucity of required parameters. The reason for this performance boost is that the residual connections of our biologically inspired networks make them gradient-stable, and that they can take advantage of the rich dynamics of reciprocally coupled oscillators (*11, 12, 14, 16*) that allow them to exploit for computations not only spatial distributions of response amplitudes, but also temporal variables such as resonance, entrainment, synchrony and asynchrony, reverberation, phase shifts, and fading memory. This also sets them apart from models based on attractor dynamics such as the ones realized in Ising models and their associated Hopfield networks (*52*).

Importantly, single network nodes in our model are intrinsically feature selective not as a result of specific combinations of input connections but because of preferred resonance frequencies and non-linear gain modulation. This also contributes to the excellent noise resilience of HORNs. Moreover, the oscillatory nature of the nodes' activity turns any input into a damped harmonic oscillation,

making it available to the network over extended periods of time in a modulated manner (fading memory). Consequently, stimulus representations oscillate, resulting in temporally modulated windows of attention that can be exploited for faster learning (see (*53*) for a detailed analysis).

The DHO units in a HORN network collectively process stimuli in a fully distributed manner by converting sensory input into waves that spread and give rise to complex interference patterns (*17, 18*). This representation provides a coding space of massive dimensionality and allows the networks to cope effectively with the encoding of time series and the superposition of information about spatially and temporally segregated events.

The introduction of physiologically plausible heterogeneity, of coupling delays and of multi-layer architectures further enhanced performance at the same number of trainable parameters because it increased the dimensionality of the networks' state spaces and allowed the networks to process stimuli on several time scales simultaneously. Such heterogeneity endowed even untrained networks with sensitivity to a broader range of correlation structures that could be exploited by subsequent learning, accelerating learning speed. Most importantly, heterogeneity enhanced the ability to process stimuli with novel characteristics, alleviating the need for fine-tuning or the cumbersome search for optimal stimulus-specific parameters. Thus, heterogeneity is functionally beneficial and likely not a byproduct of nature's imprecision.

Quite unexpectedly, gradient-based BPTT learning in HORNs resulted in changes in the coupling strengths of the recurrent connections that closely resemble those predicted by Hebbian principles. This provides strong support for the concept that biologically plausible unsupervised correlation based learning can install priors about statistical regularities of stimuli in the network's synaptic weight distributions, which in turn shape dynamic stimulus representations useful for stimulus discrimination.

Learning on geometrically organized input in HORNs resulted in synaptic weight configurations that decay with distance and that capture the Gestalt criteria of continuity and vicinity, a property also known from natural systems (*2, 45, 48*). In contrast to a most recently proposed model of an

oscillating RNN designed specifically for the purpose of studying traveling waves and in which local connectivity was enforced by design (*54*), we show here that local connectivity is likely not only a biological constraint but also results from the self-organizing installation of stimulus priors.

Finally, spontaneously active HORNs were shown to reproduce important hallmarks of experimental findings such as the stimulus induced reduction of variance (*55*) and the transient convergence of network dynamics to stimulus-specific substates (*2*).

Taken together, these findings not only unveil powerful novel principles based on analog computation, but also allow for a functional interpretation of numerous experimentally verified physiological phenomena whose roles in information processing so far have been elusive or have caused controversial discussions. Plausible functional roles can now be assigned to (i) the propensity of nodes to oscillate and the resulting dynamical phenomena such as synchronization, desynchronization, resonance, entrainment, and traveling waves (*13–16, 18, 19*); (ii) the diversity of preferred oscillation frequencies, their non-stationarity and context dependence (*3, 4*), (iii) the heterogeneity of the conduction velocities of recurrent connections (*5, 6*), (iv) the Hebbian adaptivity of recurrent connections (*7, 8*); (v) the emergence of context-dependent dynamic receptive fields by network interactions (*37, 39*), (vi) the decrease of oscillation frequencies in higher areas of the cortical processing hierarchy (*42*), and (vii) the reduction of variance in network dynamics during stimulus presentation (*55*). The simulations also suggest a physiologically plausible scenario for the rapid and parallel matching of sensory evidence with stored priors through self-organized convergence of network dynamics to classifiable, stimulus-specific, dynamic substates. These substates consist of highly structured, high-dimensional dynamical landscapes that unfold in amplitude, frequency, and phase space.

In essence, such networks of coupled oscillators perform highly parallelized analog computations in a very high-dimensional state space. This allows the networks to simultaneously relate a large number of input variables. We propose that this is the reason why biological systems can solve extremely complex problems such as scene segmentation or speech parsing so fast and efficiently despite the

slowness of their neurons. Problems of this type require the virtually simultaneous analysis of a vast number of nested relations and are notoriously difficult to solve with digital von Neumann computing architectures. One possibility for an efficient solution to such problems was found to be the exploitation of quantum effects. Yet, it is commonly believed that biological systems cannot exploit quantum effects for computation. Therefore, we hypothesize that nature solves such hard problems by means of analog computations of the kind described in this study. At the conceptual level, the highly parallel, analogue and holistic computations realized in HORNs share numerous similarities with the computations performed in quantum systems. The superposition and interactions of phases in case of biological networks share similarities with the superposition and interaction of spins in physical systems (*56*). We think that it will be possible to implement the computational principle presented here in analogue hardware that, in contrast to quantum systems, runs at room temperature, is miniaturizable and highly energy efficient. In combination with novel electrical elements mimicking Hebbian synapses such as memristors, this novel principle will likely enable the design of self-adapting and learning devices for machine learning applications that can ideally complement existing digital technologies.

Acknowledgments

The authors thank Martin Vinck, Andreas Bahmer, and Andreea Lazar for their helpful discussions and comments on an earlier version of this manuscript. We thank Claudia Kernberger for her help with some of the illustrations. We acknowledge computing support from the Ernst Strüngmann Institute and the Max Planck Computing and Data Facility. **Funding:** This study was supported by a Koselleck grant to W.S. from the German Research Foundation (DFG) and the Ernst Struengmann Institute for Neuroscience in Cooperation with Max Planck Society. **Author contributions:** Conceptualization: F.E., W.S.; Formal Analysis: F.E., P.C., I.D.; Funding acquisition: W.S.; Investigation:

F.E., P.C., I.D.; Methodology: F.E., P.C., I.D.; Project administration: F.E., W.S.; Resources: F.E., W.S.; Software: F.E., P.C., I.D.; Supervision: F.E., W.S.; Validation: F.E., P.C., I.D.; Visualization: F.E., P.C., I.D.; Writing - original draft: F.E., P.C., I.D., W.S.; Writing - review and editing: F.E., W.S.; **Competing interests:** F.E. and W.S. have filed a patent application for the hardware implementation of the described network models at the European Patent Office, application number EP22183567.1. **Data and materials availability:** Model implementations and simulation scripts will be made publicly available upon publication. All data sets used in this study are publicly available.

References

1. Nikola T. Markov, Maria Ercsey-Ravasz, Camille Lamy, Ana Rita Ribeiro Gomes, Loïc Magrou, Pierre Misery, Pascale Giroud, Pascal Barone, Colette Dehay, Zoltán Toroczkai, Kenneth Knoblauch, David C. Van Essen, and Henry Kennedy. The role of long-range connections on the specificity of the macaque interareal cortical network. *Proceedings of the National Academy of Sciences*, 110(13):5187–5192, March 2013.

2. Wolf Singer. Recurrent dynamics in the cerebral cortex: Integration of sensory evidence with stored knowledge. *Proceedings of the National Academy of Sciences*, 118(33):e2101043118, August 2021.

3. Rodolfo R. Llinás. The Intrinsic Electrophysiological Properties of Mammalian Neurons: Insights into Central Nervous System Function. *Science*, 242(4886):1654–1664, December 1988.

4. Andreas K. Engel, Peter König, Andreas K. Kreiter, Thomas B. Schillen, and Wolf Singer. Temporal coding in the visual cortex: new vistas on integration in the nervous system. *Trends in Neurosciences*, 15(6):218–226, June 1992.

5. Stephen G. Waxman. Determinants of conduction velocity in myelinated nerve fibers. *Muscle & Nerve*, 3(2):141–150, March 1980.

6. H. A. Swadlow. Physiological properties of individual cerebral axons studied in vivo for as long as one year. *Journal of Neurophysiology*, 54(5):1346–1362, November 1985.

7. Jeffrey C. Magee and Daniel Johnston. A Synaptically Controlled, Associative Signal for Hebbian Plasticity in Hippocampal Neurons. *Science*, 275(5297):209–213, January 1997.

8. Henry Markram, Joachim Lübke, Michael Frotscher, and Bert Sakmann. Regulation of Synaptic Efficacy by Coincidence of Postsynaptic APs and EPSPs. *Science*, 275(5297):213–215, January 1997.

9. D.O. Hebb. *The Organization of Behavior*. Psychology Press, 0 edition, 1949.

10. Valerio Mante, David Sussillo, Krishna V. Shenoy, and William T. Newsome. Context-dependent computation by recurrent dynamics in prefrontal cortex. *Nature*, 503(7474):78–84, November 2013.

11. György Buzsáki and Xiao-Jing Wang. Mechanisms of Gamma Oscillations. *Annual Review of Neuroscience*, 35(1):203–225, July 2012.

12. Charles M. Gray, Peter König, Andreas K. Engel, and Wolf Singer. Oscillatory responses in cat visual cortex exhibit inter-columnar synchronization which reflects global stimulus properties. *Nature*, 338(6213):334–337, March 1989.

13. W Singer. Synchronization of Cortical Activity and its Putative Role in Information Processing and Learning. *Annual Review of Physiology*, 55(1):349–374, October 1993.

14. Jessica A. Cardin, Marie Carlén, Konstantinos Meletis, Ulf Knoblich, Feng Zhang, Karl Deisseroth, Li-Huei Tsai, and Christopher I. Moore. Driving fast-spiking cells induces gamma rhythm and controls sensory responses. *Nature*, 459(7247):663–667, June 2009.

15. Keith B. Doelling and David Poeppel. Cortical entrainment to music and its modulation by expertise. *Proceedings of the National Academy of Sciences*, 112(45), November 2015.

16. Jean-Philippe Lachaux, Eugenio Rodriguez, Jacques Martinerie, and Francisco J. Varela. Measuring phase synchrony in brain signals. *Human Brain Mapping*, 8(4):194–208, 1999.

17. John R Hughes. The phenomenon of travelling waves: a review. *Clinical Electroencephalography*, 26(1):1–6, 1995.

18. Lyle Muller, Frédéric Chavane, John Reynolds, and Terrence J. Sejnowski. Cortical travelling waves: mechanisms and computational principles. *Nature Reviews Neuroscience*, 19(5):255–268, May 2018.

19. Zachary W. Davis, Lyle Muller, Julio Martinez-Trujillo, Terrence Sejnowski, and John H. Reynolds. Spontaneous travelling cortical waves gate perception in behaving primates. *Nature*, 587(7834):432–436, November 2020.

20. Olaf Sporns, Dante R Chialvo, Marcus Kaiser, and Claus C Hilgetag. Organization, development and function of complex brain networks. *Trends in cognitive sciences*, 8(9):418–425, 2004.

21. Yann LeCun, Yoshua Bengio, and Geoffrey Hinton. Deep learning. *Nature*, 521(7553):436–444, May 2015.

22. T. Konstantin Rusch and Siddhartha Mishra. Coupled Oscillatory Recurrent Neural Network (coRNN): An accurate and (gradient) stable architecture for learning long time dependencies. *arXiv:2010.00951 [cs, stat]*, March 2021. arXiv: 2010.00951.

23. Eve Marder and Dirk Bucher. Central pattern generators and the control of rhythmic movements. *Current Biology*, 11(23):R986–R996, November 2001.

24. Wolf Singer. Neuronal Synchrony: A Versatile Code for the Definition of Relations? *Neuron*, 24(1):49–65, September 1999.

25. Michael N. Shadlen and J.Anthony Movshon. Synchrony Unbound. *Neuron*, 24(1):67–77, September 1999.

26. Supratim Ray and John H. R. Maunsell. Differences in Gamma Frequencies across Visual Cortex Restrict Their Possible Use in Computation. *Neuron*, 67(5):885–896, September 2010.

27. Supratim Ray and John H.R. Maunsell. Do gamma oscillations play a role in cerebral cortex? *Trends in Cognitive Sciences*, 19(2):78–85, February 2015.

28. Georgios Spyropoulos, Matteo Saponati, Jarrod Robert Dowdall, Marieke Louise Schölvinck, Conrado Arturo Bosman, Bruss Lima, Alina Peter, Irene Onorato, Johanna Klon-Lipok, Rasmus Roese, Sergio Neuenschwander, Pascal Fries, and Martin Vinck. Spontaneous variability in gamma dynamics described by a damped harmonic oscillator driven by noise. *Nature Communications*, 13(1):2019, December 2022.

29. György Buzsáki and Andreas Draguhn. Neuronal Oscillations in Cortical Networks. *Science*, 304(5679):1926–1929, June 2004.

30. Ben H. Jansen and Vincent G. Rit. Electroencephalogram and visual evoked potential generation in a mathematical model of coupled cortical columns. *Biological Cybernetics*, 73(4):357–366, September 1995.

31. Jack D. Cowan, Jeremy Neuman, and Wim van Drongelen. Wilson–Cowan Equations for Neocortical Dynamics. *The Journal of Mathematical Neuroscience*, 6(1):1, December 2016.

32. R J Douglas and K A Martin. A functional microcircuit for cat visual cortex. *The Journal of Physiology*, 440(1):735–769, August 1991.

33. Yun Wang, Henry Markram, Philip H. Goodman, Thomas K. Berger, Junying Ma, and Patricia S. Goldman-Rakic. Heterogeneity in the pyramidal network of the medial prefrontal cortex. *Nature Neuroscience*, 9(4):534–542, April 2006.

34. John D. Murray, Alberto Bernacchia, David J. Freedman, Ranulfo Romo, Jonathan D. Wallis, Xinying Cai, Camillo Padoa-Schioppa, Tatiana Pasternak, Hyojung Seo, Daeyeol Lee, and Xiao-Jing Wang. A hierarchy of intrinsic timescales across primate cortex. *Nature Neuroscience*, 17(12):1661–1663, December 2014.

35. Mantas Lukoševičius and Herbert Jaeger. Reservoir computing approaches to recurrent neural network training. *Computer Science Review*, 3(3):127–149, August 2009.

36. Danko Nikolić, Stefan Häusler, Wolf Singer, and Wolfgang Maass. Distributed Fading Memory for Stimulus Properties in the Primary Visual Cortex. *PLoS Biology*, 7(12):e1000260, December 2009.

37. Colin Blakemore, Jacob Nachmias, and Peter Sutton. The perceived spatial frequency shift: evidence for frequency-selective neurones in the human brain. *The Journal of Physiology*, 210(3):727–750, October 1970.

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

39. S. V. David. Natural Stimulus Statistics Alter the Receptive Field Structure of V1 Neurons. *Journal of Neuroscience*, 24(31):6991–7006, August 2004.

40. Aniruddha Das and Charles D. Gilbert. Topography of contextual modulations mediated by short-range interactions in primary visual cortex. *Nature*, 399(6737):655–661, June 1999.

41. D Ferster and S Lindström. An intracellular analysis of geniculo-cortical connectivity in area 17 of the cat. *The Journal of Physiology*, 342(1):181–215, September 1983.

42. Nikola T. Markov, Mária Ercsey-Ravasz, David C. Van Essen, Kenneth Knoblauch, Zoltán Toroczkai, and Henry Kennedy. Cortical High-Density Counterstream Architectures. *Science*, 342(6158):1238406, November 2013.

43. Siegrid Löwel and Wolf Singer. Selection of Intrinsic Horizontal Connections in the Visual Cortex by Correlated Neuronal Activity. *Science*, 255(5041):209–212, January 1992.

44. Yang Yiling, Katharine Shapcott, Alina Peter, Huang Xuhui, Andreea Lazar, and Wolf Singer. Robust Encoding of Natural Stimuli by Neuronal Response Sequences in Monkey Visual Cortex. *Nature Communications*, in revision, 2022.

45. Andreea Lazar, Christopher Lewis, Pascal Fries, Wolf Singer, and Danko Nikolic. Visual exposure enhances stimulus encoding and persistence in primary cortex. *Proceedings of the National Academy of Sciences*, 118(43):e2105276118, October 2021.

46. Christian Hansel, David J Linden, and Egidio D'Angelo. Beyond parallel fiber ltd: the diversity of synaptic and non-synaptic plasticity in the cerebellum. *Nature neuroscience*, 4(5):467–475, 2001.

47. Karunesh Ganguly, Alejandro F Schinder, Scott T Wong, and Mu-ming Poo. Gaba itself promotes the developmental switch of neuronal gabaergic responses from excitation to inhibition. *Cell*, 105(4):521–532, 2001.

48. Mark M Churchland, Byron M Yu, John P Cunningham, Leo P Sugrue, Marlene R Cohen, Greg S Corrado, William T Newsome, Andrew M Clark, Paymon Hosseini, Benjamin B Scott, et al. Stimulus onset quenches neural variability: a widespread cortical phenomenon. *Nature neuroscience*, 13(3):369–378, 2010.

49. Pietro Berkes, Gergő Orbán, Máté Lengyel, and József Fiser. Spontaneous cortical activity reveals hallmarks of an optimal internal model of the environment. *Science*, 331(6013):83–87, 2011.

50. Mihály Bányai, Andreea Lazar, Liane Klein, Johanna Klon-Lipok, Marcell Stippinger, Wolf Singer, and Gergő Orbán. Stimulus complexity shapes response correlations in primary visual cortex. *Proceedings of the National Academy of Sciences*, 116(7):2723–2732, 2019.

51. Rosalyn Moran, Dimitris A. Pinotsis, and Karl Friston. Neural masses and fields in dynamic causal modeling. *Frontiers in Computational Neuroscience*, 7, 2013.

52. John J Hopfield. Neural networks and physical systems with emergent collective computational abilities. *Proceedings of the national academy of sciences*, 79(8):2554–2558, 1982.

53. Igor Dubinin and Felix Effenberger. Fading memory as inductive bias in residual recurrent networks, 2023.

54. T Anderson Keller and Max Welling. Neural wave machines: Learning spatiotemporally structured representations with locally coupled oscillatory recurrent neural networks, 2023.

55. Mark M Churchland, Byron M Yu, John P Cunningham, Leo P Sugrue, Marlene R Cohen, Greg S Corrado, William T Newsome, Andrew M Clark, Paymon Hosseini, Benjamin B Scott, David C Bradley, Matthew A Smith, Adam Kohn, J Anthony Movshon, Katherine M Armstrong, Tirin Moore, Steve W Chang, Lawrence H Snyder, Stephen G Lisberger, Nicholas J Priebe, Ian M

Finn, David Ferster, Stephen I Ryu, Gopal Santhanam, Maneesh Sahani, and Krishna V Shenoy. Stimulus onset quenches neural variability: a widespread cortical phenomenon. *Nature Neuroscience*, 13(3):369–378, March 2010.

56. William Moy, Ibrahim Ahmed, Po-wei Chiu, John Moy, Sachin S Sapatnekar, and Chris H Kim. A 1,968-node coupled ring oscillator circuit for combinatorial optimization problem solving. *Nature Electronics*, 5(5):310–317, 2022.

57. Ernst Hairer, Christian Lubich, and Gerhard Wanner. Geometric numerical integration illustrated by the Störmer–Verlet method. *Acta Numerica*, 12:399–450, May 2003.

58. Jeffrey L Elman. Finding structure in time. *Cognitive science*, 14(2):179–211, 1990. Publisher: Wiley Online Library.

59. H. Sompolinsky, A. Crisanti, and H. J. Sommers. Chaos in Random Neural Networks. *Physical Review Letters*, 61(3):259–262, July 1988.

60. Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Kopf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. PyTorch: An Imperative Style, High-Performance Deep Learning Library. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d' Alché-Buc, E. Fox, and R. Garnett, editors, *Advances in Neural Information Processing Systems 32*, pages 8024–8035. Curran Associates, Inc., 2019.

61. Yann LeCun, Corinna Cortes, and CJ Burges. MNIST handwritten digit database. *ATT Labs [Online]. Available: http://yann.lecun.com/exdb/mnist*, 2, 2010.

62. Gregory Cohen, Saeed Afshar, Jonathan Tapson, and Andre van Schaik. EMNIST: Extending MNIST to handwritten letters. In *2017 International Joint Conference on Neural Networks (IJCNN)*, pages 2921–2926, Anchorage, AK, USA, May 2017. IEEE.

63. P. Warden. Speech commands: A dataset for limited-vocabulary speech recognition. *ArXiv e-prints*, April 2018. arXiv: 1804.03209 [cs.CL].

64. Razvan Pascanu, Tomas Mikolov, and Yoshua Bengio. On the difficulty of training Recurrent Neural Networks. *arXiv:1211.5063 [cs]*, February 2013. arXiv: 1211.5063.

65. Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep Residual Learning for Image Recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 770–778, 2016.

66. Martin Arjovsky, Amar Shah, and Yoshua Bengio. Unitary evolution recurrent neural networks. In *International conference on machine learning*, pages 1120–1128. PMLR, 2016.

67. Michael Rosenblum, Arkady Pikovsky, Jurgen Kurths, Carsten Schafer, and Peter A Tass. Phase synchronization: from theory to data analysis. *Handbook of biological physics*, 4:279–321, 2001.

68. Fritz K. Kneubühl. *Oscillations and Waves*. Springer Berlin Heidelberg, Berlin, Heidelberg, 1997.

Materials and Methods

DHO nodes. We model each HORN node as a driven damped harmonic oscillator (DHO) with feedback, defined by the second-order ordinary differential equation (ODE)

$$\ddot{x}(t) + 2\gamma\dot{x}(t) + \omega^2 x(t) = \alpha\sigma\left(I\left(t\right) + F\left(x\left(t\right), \dot{x}\left(t\right)\right)\right), \tag{1}$$

with $x(t)$ denoting the oscillator amplitude, $\gamma > 0$ the damping factor (fig. S1A), $\omega > 0$ the natural angular frequency (fig. S1B), and $\alpha > 0$ an excitability parameter. $\sigma(\cdot)$ denotes an input nonlinearity that models local resource constraints and is chosen as $\tanh(\cdot)$. $I(t)$ denotes a time-varying external input and $F(x(t), \dot{x}(t)) = vx(t) + w\dot{x}(t)$, $v, w \in \mathbb{R}$ the feedback input (see below).

Depending on the quotient $s = \omega/\gamma$, the solutions of (1) have qualitatively different relaxation dynamics (fig. S1A,B): (i) When $s > 1$, solutions are given by oscillations with decaying amplitude (subcritical damping). (ii) When $s = 1$ or $s < 1$, solutions are given by non-oscillating exponentially decaying curves (critical damping and supercritical damping, respectively).

To define a discrete-time update equation for a DHO node, we first rewrite (1) as a system of two first order ODEs

$$\begin{aligned} \dot{x}(t) &= y(t) \\ \dot{y}(t) &= \alpha \tanh\left(I(t) + F(x(t), y(t))\right) - 2\gamma y(t) - \omega^2 x(t) \end{aligned} \tag{2}$$

and then discretize (2) using a microscopic time constant h to obtain the update equation for a discrete-time HORN network in which each node performs an Euler integration of (2) at each time step. As an explicit Euler integration scheme can be numerically unstable when used for the integration of (2), we implemented a symplectic Euler integration scheme (*57*) for each node and obtain the update equations

$$\begin{aligned} x_{t+1} &= x_t + h y_{t+1}, \\ y_{t+1} &= y_t + h\left[\alpha \tanh(I_t + F_t(x_t, y_t)) - 2\gamma y_t - \omega^2 x_t\right]. \end{aligned} \tag{3}$$

Here, t denotes a discrete discrete time index $1 \leq t \leq T$, I_t is the external input at time t, and F_t the feedback input of the node to itself. Initial conditions are chosen as $x_1 = y_1 = 0$ unless otherwise

noted. The natural frequency ω corresponds to a period $\omega_p = 2\pi/(\omega h)$ and the memory half-life time is proportional to $1/(\gamma h)$, both measured in input time steps.

The microscopic time constant h defines a scaling factor of input time to system time. It can be interpreted as the sampling frequency of the system, and the linear Nyquist frequency is $h_{\text{Nyquist}} = 1/2h$. Therefore, the theoretical upper limit of the natural (circular) frequency parameter ω for any DHO node must be $\omega_{\text{max}} < \pi/h$. In practice, this bound is lower as the Euler integration step in (3) becomes numerically unstable (data not shown).

Since the number of integration time steps always equals the length T of the input time series, the numerical integration of the system (3) is invariant under changes in $h \to h'$ when system parameters are rescaled according to

$$\gamma' = c\gamma, \quad \omega' = c\omega, \quad \alpha' = c^2\alpha, \tag{4}$$

where $c = h/h'$. To allow a better interpretability of the natural frequency parameter, we set $h = 1$ without loss of generality to link the two time scales and express the natural frequency ω in period nodes of the input as $\omega_p = 2\pi/\omega$ when convenient. A value of $h = 1$ will be assumed throughout the manuscript if not stated otherwise.

HORN Networks. A HORN network is a system of n additively coupled DHOs of the form (3). Given a time-varying external input $S_{\text{ext}} = (\mathbf{s}_1, \ldots, \mathbf{s}_T)$, $\mathbf{s}_t \in \mathbb{R}^m$, the update equations of a HORN are

$$\begin{aligned} \mathbf{x}_{t+1} &= \mathbf{x}_t + h\mathbf{y}_{t+1} \\ \mathbf{y}_{t+1} &= \mathbf{y}_t + h\left[\boldsymbol{\alpha} \cdot \tanh\left(1/\sqrt{n}\mathbf{I}_t^{\text{rec}} + \mathbf{I}_t^{\text{ext}}\right) - 2\boldsymbol{\gamma} \cdot \mathbf{y}_t - \boldsymbol{\omega}^2 \cdot \mathbf{x}_t\right], \end{aligned} \tag{5}$$

where I_t^{rec} and I_t^{ext} denote the recurrent and external input to each node at time t, respectively, defined

as

$$\mathbf{I}_t^{\text{rec}} = \mathbf{W}^{hh}\mathbf{y}_t + \mathbf{b}^{hh} + \mathbf{v} \cdot \mathbf{x}_t,$$
$$\mathbf{I}_t^{\text{ext}} = \mathbf{W}^{ih}\mathbf{s}_t + \mathbf{b}^{ih}.$$

Above, lowercase boldface symbols denote vectors of length n, uppercase boldface symbols denote matrices, $\cdot$ denotes element-wise multiplication, $\mathbf{W}^{hh}$ is the $n \times n$ matrix of recurrent coupling weights, $\mathbf{W}^{ih}$ is the $n \times m$ matrix of input weights, and $\mathbf{b}^{hh}$, $\mathbf{b}^{ih}$ are bias vectors. As for the single-node case, the update equations implement a symplectic Euler integration scheme. The feedback input from a node to itself $F(x_t, y_t)$ is $F(x_t, y_t) = vx_t + wy_t$ with $w = \mathbf{W}_{ii}^{hh}$. Note that the values of the parameters ω, γ, α can vary between DHO nodes. To train a HORN on a task, an affine readout of the network state $\mathbf{x}_{t_0}$ at time t_0 is defined as a map $\mathbf{R}(\mathbf{x}_{t_0}) = \mathbf{W}^{ho}\mathbf{x}_{t_0} + \mathbf{b}^{ho}$, where $\mathbf{W}^{ho}$ is the $k \times n$ readout matrix and $\mathbf{b}^{ho}$ is a bias vector. As previously, we will set $h = 1$ to link the input and system time scales, unless otherwise stated. See below for a description of network simulations.

Other RNN Architectures. A simple Elman RNN (*58*) with a linear readout (referenced by the label "tanh" throughout all plots) was implemented by the update equation

$$\mathbf{x}_{k+1} = \tanh\big(\mathbf{W}^{hh}\mathbf{x}_k + \mathbf{b}^{hh} + \mathbf{I}_k\big),$$

where the input $\mathbf{I}_k$ is defined as above. Similarly, a "leak" RNN consisting of leaky integrators was implemented by the update equation

$$\mathbf{x}_{k+1} = \mathbf{x}_k + h\left[-a\mathbf{x}_k + \tanh\big(\mathbf{W}^{hh}\mathbf{x}_k + \mathbf{b}^{hh} + \mathbf{I}_k\big)\right],$$

where a is a leak constant that controls memory decay, see also (*59*) in which $a = 1, h = 1$. For the LSTM and GRU based RNNs, the implementations of LSTM and GRU nodes provided by PyTorch

(version 1.9.0) (*60*) were used, see below.

Number of Trainable Parameters. Performance comparisons for different RNN architectures were carried out at the same number of trainable parameters. Assuming an RNN with n_h nodes receiving input of dimensionality n_i through an affine projection layer, and a readout to an n_o-dimensional output vector via an affine readout layer, the number of trainable parameters is $N = c(n_i + b_i)n_h + cn_h(n_h+b_h)+(n_h+b_o)n_o$, where c is the number of state variables per node ($c = 3$ for GRU, $c = 4$ for LSTM, $c = 1$ for the simple and leak RNNs) and b_i, b_h, b_o is either 1 (bias term present) or 0 (no bias term present). HORNs also have one state variable per node (i.e. $c = 1$), but an additional amplitude self-connection term for each node, leading to $N_{\text{HORN}} = (n_i + b_i)n_h + n_h(n_h + 1 + b_h) + (n_h + b_o)n_o$. To construct a network of a given architecture with a given number N^* of trainable parameters, we solve the quadratic equation $N = N^*$ after plugging in the known values for n_i, n_o, b_i, b_h, b_o and take the integer part of the positive solution to obtain the largest number n^* of nodes so that the number of trainable parameters does not exceed N^*. See fig. S20 for the scaling behavior of trainable parameters as a function of the number system nodes for different RNN architectures for $n_i = 1$, $n_o = 10$, $b_i = b_h = b_o = 1$.

Data Sets. The **MNIST** data set (*61*) of hand written digits consists of a training data set of 60,000 samples and a test set of 10,000 samples which was used to assess classification performance (measured by classification accuracy) of different networks. Each sample is given by a 28x28 matrix of intensity values between $[0, 255]$ which are scaled to $[0, 1]$. Two variants of the MNIST data set are considered here, the sequential MNIST (sMNIST), and the permuted sequential MNIST (psMNIST). For sMNIST, the 28x28 intensity values of a sample are collected in scanline order from top left to bottom right to form a time series of length 784 that is then fed to the networks step by step. For psMNIST, a random but fixed permutation on 784 elements is applied to the stimulus before feeding the input to the network. This removes the dominant low frequency structure present in sMNIST digits and distributes stimulus information across all frequencies (fig. S10, S5). Noise was added to

sMNIST or psMNIST time series in form of i.i.d. additive white Gaussian noise on a per-pixel level and sampled from $N(0, \sigma^2)$. Stimulus values were clamped to $[0, 1]$ after noise was added.

The **Line segments data set** (LSDS) consists of samples of 28x28 pixel intensity values with one or more line segments of a given angle. The data set has four parameters that define the nature of its samples: The number of angles n_a, the maximal number of segments per sample n_s, the minimal and maximal length of the line segments l_{min} and l_{max}. It furthermore comes in two flavors: with random segment locations or centered segment locations. To construct one sample, first each entry of a 28×28 pixel canvas is set to 0 and an angle $\alpha \in \{\alpha_1, \ldots, \alpha_{n_a}\}$ is chosen at random with $\alpha_i = (i-1)\pi/n_a$, i.e. each sample only shows line segments of the same angle. Then the number of line segments k to be shown is chosen as a random integer in $1, \ldots, n_s$. For each line segment i, first a random integer length l_i is drawn from $l_{\text{min}}, \ldots, l_{\text{max}}$. Then a line segment of length l_i pixels is drawn on the 28×28 canvas with intensity 1.0 (using the Python package `opencv-python`, version 4.6). If line segments are to be centered, they are drawn so that their central point coincides with the center of the 28×28 canvas. If segment locations are random, a random starting coordinate from the set $\{0, \ldots, 21\}$ is chosen for both the horizontal and vertical starting point of the line segment. The label for each sample is the angle α present in the sample and the networks are trained on a n_a-class classification problem of predicting the angle shown in a given sample. We will write LSDS(n_a, n_s, l_{min}, l_{max}, L) with $L \in \text{r}, \text{c}$ to refer to the data set with random ($L = \text{r}$) and centered ($L = \text{c}$) segment locations. We use two versions of the LSDS in the study, LSDS(32,5,11,c) which we will refer to as LSDSa, and LSDS(10,3,8,24,r) which we will refer to as LSDSb. Samples of line segment data sets are shown in fig. S14.

The **EMNIST** (balanced) data set (*62*) consists of hand-written 131,600 characters belonging to 47 classes, representing hand written digits as well as uppercase and lowercase latin characters. Each character is represented by a matrix of 28x28 pixel intensity values as for MNIST. The training data set is of size 112,800, the test set of size 18,800. EMNIST samples have the same format as MNIST,

and their signal properties as time series are similar to those of MNIST digits.

The of **Spoken Digits Data Set** (SDDS), was chosen as a subset of the Speech Commands data set (version 0.0.2) (*63*) (SCDS). The SCDS consists of 65,000 audio audio recordings (1 second length, mono, 16kHz sampling rate) of spoken words and digits obtained from thousands of people and spans 30 different classes. The SDDS data set was formed by choosing the 10 classes from the SCDS representing spoken digits 0 to 9, forming a data set of 23,666 samples (with each class consisting of around 2730 samples). Of the SDDS, a random selection of 10% of all samples were used as test set, and the rest was used as the training set. To construct the input time series to networks, each spoken digit was first sub-sampled to 4000 Hz using the Python library `librosa`, followed by a normalization of the amplitude to the interval $[-1, 1]$, performed per sample.

The power spectral densities of the time series calculated on samples from each data set is shown in fig. S5.

Simulations. All simulations were performed in PyTorch (version 1.9.0) (*60*). All networks consisted of three trainable affine layers (input, recurrent, readout). For the Elman, leak and HORN networks, custom implementations were used and the code is available at `https://github.com/....` For the LSTM and GRU networks, the standard implementations provided by PyTorch were used.

The initial weights and biases of all affine layers were randomly drawn from a uniform distribution $U(-1/\sqrt{n_{\text{in}}}, 1/\sqrt{n_{\text{in}}})$, where n_{in} denotes the input dimension of the layer (default implementation in PyTorch `torch.nn.Linear`).

All parameters were learned with the BPTT algorithm. As loss function for the studied stimulus classification problems we used the cross entropy loss function (`torch.nn.CrossEntropyLoss`). Weight updates were performed according to the AdamW optimizer with default settings (`betas=(0.9,0.999)`, `eps=1e-8`, `weight_decay=0.01`) using a batch size of 128 for all data sets, unless otherwise stated. For the SDDS data set, we used a batch size of 64.

For each model, a grid search on hyperparameters was performed using a network with 10^4 train-

able parameters to find the parameter values that resulted in a network with the highest classification accuracy on the test set after 10 training epochs. The parameters contained in the grid search were different per model and are described below. For all models, the grid search included the learning rate `lr` $\in \{10^{-5}, 5 \cdot 10^{-5}, 10^{-4}, 10^{-3}, 5 \cdot 10^{-3}, 10^{-2}\}$. Optimal learning rates for HORNs and the leak RNN were found to be 10^{-2} except for SDDS where they were $5 \cdot 10^{-3}$. Optimal learning rates for the gated (LSTM, GRU) RNNs were determined to be 10^{-3} for all data sets (SDDS: 10^{-4}). Learning rates for the Elman RNN were set to 10^{-4} for all data sets (SDDS: 10^{-5}). Gradient clipping (`pytorch.nn.utils.clip_grad_norm_`) was applied in the case of the Elman, GRU, and LSTM networks, as without gradient clipping these architectures showed unstable learning trajectories or failed to depart from chance-level classification performance for sMNIST. For these networks, the grid search included the learning rate and clipping values $c \in \{0, 0.1, 1, 5, 10, 100\}$. A clipping value of 1 was chosen across all data sets. For the leak RNNs, the grid search included the learning rate, the time constant $h \in \{0.01, 0.05, 0.1, 0.2, 0.3, 0.4, 0.5\}$ and the leak constant $a \in \{0.05, 0.1, 0.2, 0.3, 0.4, 0.5, 0.75, 1.0\}$. See Table S1 for the parameter values for each data set used. For homogeneous HORNs, the grid search included the learning rate, the time constant $h \in \{0.01, 0.05, 0.1, 0.2, 0.3, 0.4, 0.5\}$, the natural frequency $\omega \in \{0.1, 0.5, 1.0, 2.0, 3.0, 4.0, 5.0, 10.0\}$, and the damping factor $\gamma \in \{0.001, 0.005, 0.01, 0.05, 0.1, 0.25, 0.5\}$, setting $\alpha = 1.0$. For heterogeneous HORNs, as a first step the spread of values h, ω, γ among the best 10 performing homogeneous HORNs from the previous step were calculated as $[h_{\text{min}}, h_{\text{max}}]$, $[\omega_{\text{min}}, \omega_{\text{max}}]$, $[\gamma_{\text{min}}, \gamma_{\text{max}}]$, respectively. Subsequently, each node of a HORN^n was randomly assigned parameter values h_i, ω_i, γ_i sampled in an i.i.d. manner from $U([h_{\text{min}}, h_{\text{max}}))$, $U([\omega_{\text{min}}, \omega_{\text{max}}))$, $U([\gamma_{\text{min}}, \gamma_{\text{max}})$, respectively, where $U([a, b))$ denotes the uniform distribution over the half-open interval $[a, b)$.

See Table S.1 for dataset-dependent parameters for the HORN and leak networks. For HORN networks, normalized parameters with $h = 1$ are given. For the implementation details of HORN networks with synaptic delays and multi-layer networks see below.

Gradient Stability. Given a loss function L, the gradients of L with respect to a recurrent weight w can be computed as (*64*)

$$\frac{\partial L}{\partial w} = \sum_{1 \leq k \leq T} \frac{\partial L_k}{\partial w} = \sum_{1 \leq k \leq T} \sum_{1 \leq l \leq k} \frac{\partial L_k}{\partial H_k} \frac{\partial H_k}{\partial H_l} \frac{\partial H_l^+}{\partial w}, \tag{6}$$

where H_t is hidden state of the network at the time t, T is depth of recurrent network (i.e. the number of input time steps), $\frac{\partial H_l^+}{\partial w}$ is the immediate derivative and where

$$\frac{\partial H_k}{\partial H_l} = \prod_{k \leq t < l} \frac{\partial H_{t+1}}{\partial H_t}, \tag{7}$$

by iterated application of the chain rule. The propagation of the gradient at time t is defined by the Jacobian's behavior $\frac{\partial H_{t+1}}{\partial H_t}$. The vanishing gradient problem is characterized by $\left\| \prod_{k \leq t < l} \frac{\partial H_{t+1}}{\partial H_t} \right\| \ll 1$, and $\left\| \prod_{k \leq t < l} \frac{\partial H_{t+1}}{\partial H_t} \right\| \gg 1$ is called exploding gradient problem.

To study the dynamics of gradients in HORNN networks, we first make the following substitutions in the expression (5):

$$\mathbf{H}_t = \begin{bmatrix} \mathbf{x}_t \\ \mathbf{y}_t \end{bmatrix}, \tag{8}$$

$$\mathbf{A} = \begin{bmatrix} \mathbf{I} - h^2 \operatorname{diag} \boldsymbol{\omega}^2 & (h - 2h^2) \operatorname{diag}(\boldsymbol{\gamma}) \\ -h \operatorname{diag}(\boldsymbol{\omega}^2) & \mathbf{I} - 2h \operatorname{diag}(\boldsymbol{\gamma}) \end{bmatrix}, \tag{9}$$

and denote by

$$\phi(\mathbf{H}_t, \mathbf{s}_t) = \begin{bmatrix} h \tanh(\mathbf{I}_t^{\text{rec}}(\mathbf{H}_t) + \mathbf{I}_t^{\text{ext}}(\mathbf{s}_t)) \\ \tanh(\mathbf{I}_t^{\text{rec}}(\mathbf{H}_t) + \mathbf{I}_t^{\text{ext}}(\mathbf{s}_t)) \end{bmatrix}, \tag{10}$$

the transformation defined by the $\tanh$ non-linearity, where $\mathbf{s}_t, \mathbf{I}_t^{\text{rec}}, \mathbf{I}_t^{\text{ext}}$ are as in (5), $\mathbf{I}$ denotes the identity matrix, and $\operatorname{diag}(\mathbf{x})$ the diagonal matrix with diagonal entries from the vector $\mathbf{x}$.

These substitutions allow us to write the recurrent update step (5) as

$$\mathbf{H}_{t+1} = \mathbf{A}\mathbf{H}_t + h\boldsymbol{\alpha}\phi(\mathbf{H}_t, \mathbf{s}_t). \tag{11}$$

The sufficient condition to avoid both vanishing and exploding gradients is $\frac{\partial \mathbf{H}_{t+1}}{\partial \mathbf{H}_t} \approx \mathbf{I}$. This condition can be achieved in (11) by choosing an Euler discretization time step $h \ll 1$. The scaling relationships for the system parameters ω, γ, α that result in equivalent system behavior under changes of h is described in (4). Similar arguments hold for the RNN composed of leaky integrators, explaining their increased performance compared to a simple Elman RNN. In essence, the Euler integration scheme introduces residual connections between subsequent steps of HORNN dynamics, which ensure stable gradient propagation (*65*).

Oscillating gradients. L^∞ gradient norms of $\frac{\partial L}{\partial x_i}$ were computed based on network activity of a homogeneous HORN (64 nodes) obtained on the test set of sMNIST (*66*). Gradients were computed for each time step of network activity (fig. S11, right). Such modulated gradient magnitudes implement an attention mechanism that allows networks to pick up on certain spectral properties of stimuli (such as prominent frequencies) while ignoring others.

Phase Locking Value. To analyze the pairwise synchronization between DHO nodes, we used the analytical representation of the amplitude signal $x(t)$ which can be computed by means of the Hilbert transform as $X(t) = x(t) + iH\left(x(t)\right)$, where $H(\cdot)$ is the Hilbert transform and i is the imaginary node. Then, the instantaneous phase $\phi(t)$ and the instantaneous magnitude $A(t)$ can be computed as $S(t) = A(t)\exp\left(i\phi(t)\right)$. As a measure of nonlinear interactions, we use the pairwise phase locking value (PLV) calculated over the time window T, where T denotes the length of the input signal (*67*). The PLV for two signals x_1 and x_2 in analytical representation is defined as

$$\mathrm{PLV}(x_1, x_2) = \frac{1}{T}|\sum_{t=1}^{T} e^{i\Delta\phi(t)}|, \tag{12}$$

where $\Delta\phi(t) = \phi_1(t) - \phi_2(t)$.

The phase locking value PLV takes values on the real interval $[0, 1]$. The case of $\mathrm{PLV}(x_1, x_2) = 1$ indicates fully phase-locked synchronization of x_1 and x_2, while a value of $\mathrm{PLV}(x_1, x_2) = 0$ indicates fully asynchronous signals x_1, x_2.

PCA dimensionality of network activity. To measure PCA dimensionality, network activity vectors were collected using 1000 samples from the sMNIST test dataset. The activity vectors for all 784 time steps of stimulus presentation were treated as independent samples, resulting in $1000 \cdot 784$ samples. PCA was performed on the resulting dataset (Python package `scikit-learn`, `sklearn.decomposition.PCA`). As a result, every principal component constitutes the activation vector which explains the amount of variance as defined by the corresponding eigenvalue of the covariance matrix. The number of PCA components explaining 99% variance of the network's activity was taken as measure of PCA dimensionality. PCA dimensionality measures the linear dimensionality and takes integer values on the interval $[1, N]$, where N is the number of nodes in the network. To show the dynamics in PCA dimensionality over learning, it was computed for a sequence of intermediate instances over training for each network.

Decoding of HORN activity. To decode the stimulus based on RNN activity, activity vectors were collected at each time step t of the stimulus presentation phase ($t = 1, \ldots, 784$ for sMNIST), using 10000 samples from the test dataset. For each time step t, the resulting set of activity vectors was randomly split into a training set and a test set using a 80%/20% ratio. Using this dataset, an SVM classifier was trained on the 10-class stimulus classification problem for each time point t (Python package `scikit-learn`, `sklearn.svm.LinearSVC`). A regularization constant of $C = 0.1$ was chosen to ensure the convergence of all SVM classifiers. The confusion matrices were calculated using `sklearn.metrics.confusion_matrix`. Distance to the SVM decision hyperplane was calculated with `sklearn.svm.LinearSVC.decision_function`.

Receptive Fields. For each DHO node, the intrinsic receptive field (IRF) was calculated as the gain

curve of the node (*68*). The gain curve is computed using harmonic (sinusoidal) inputs. It is defined as the quotient between the DHO node amplitude, i.e. the stationary forced oscillation, and the input amplitude as a function of input frequency ω_i. For the intrinsic receptive field, these gain curves were calculated for an input amplitude of $A = 1$ over the time of $T = 50000$ time steps to allow the transient dynamics to settle.

The stimulus-dependent effective receptive field (ERF) of each DHO node in a HORN network was calculated as the mean most strongly driving stimulus (as the mean absolute amplitude computed over the entire presentation time). For this, network activity was first recorded for $n = 10000$ samples in a test set, and then the average of the 500 stimuli that resulted in the highest activation of a given DHO node was calculated to form the effective receptive field of a DHO node.

Networks with Synaptic Delays. To implement synaptic conduction delays in a network of n nodes, we first defined a maximum delay d_{max} present in a network (measured in the number of discrete time steps) and subsequently assigned to each connection from a node j to a node i a delay of d_{ij} time steps. We then modified the recurrent input $\mathbf{I}_t^{\text{rec}}$ of the HORN update equation (5) to

$$\mathbf{I}_t^{\text{rec}} = \sum_{d=1}^{d_{\text{max}}} \mathbf{M}^d \mathbf{W}^{hh} \mathbf{y}_{t+1-d} + \mathbf{b}^{hh} + \mathbf{v} \cdot \mathbf{x}_t, \tag{13}$$

where $\mathbf{M}^d \in \{0,1\}^{n \times n}$, $1 \leq d \leq d_{\text{max}}$ are binary masks and the value of $\mathbf{M}_{ij}^d = 1$ iff the connection from node j to node i has a delay of d, and $\mathbf{M}_{ij}^d = 0$ otherwise. Furthermore, we assume $\mathbf{y}_i = 0$ for all $i < 0$. This approach allows us to continue using the dense gradient methods implemented in PyTorch while effectively performing operations on d_{max} sparse matrices in each time step.

Multi-Layer Networks. To implement a two-layer network with per-layer connectivity, we first choose the number of nodes per layer as n_1 and n_2. Furthermore, we introduce connection probability parameters $p_1, p_2, p_{\text{F}}, p_{\text{B}}$ where p_1 and p_2 model the recurrent connection probabilities within the layer 1 and 2, respectively, p_{F} is the feedforward connection probability from layer 1 to layer 2, and p_{B} is

the feedback connection probability from layer 2 to layer 1. We set $p_1 = p_2 = 1$ unless otherwise noted, i.e. we choose fully connected layers. We then model the two-layer network by simulating a HORN network of $n = n_1 + n_2$ nodes, where the first n_1 nodes represent layer 1, and the rest of the nodes layer 2. To model layer-specific connectivity patterns, we modify the HORN update equation (5) using masks as above and write

$$\mathbf{I}_t^{\text{rec}} = \mathbf{M}^{hh}\mathbf{W}^{hh}\mathbf{y}_t + \mathbf{b}^{hh} + \mathbf{v} \cdot \mathbf{x}_t, \tag{14}$$

where the masking matrix $\mathbf{M}^{hh} \in \{0, 1\}^{n \times n}$ has a block structure and defines the connections present in the network. The entries of $\mathbf{M}^{hh}$ are sampled from a binomial distributions $B(p)$ in an i.i.d. manner as follows: $\mathbf{M}_{ij}^{hh} \sim B(p_1)$ for $1 \leq i, j \leq n_1$ (recurrent connections of layer 1), $\mathbf{M}_{ij}^{hh} \sim B(p_{\text{F}})$ for $n_1 + 1 \leq i \leq n_1 + n_2$, $1 \leq j \leq n_1$ (feedforward connections from layer 1 to layer 2), $\mathbf{M}_{ij}^{hh} \sim B(p_{\text{F}})$ for $1 \leq i \leq n_1$, $n_1 + 1 \leq j \leq n_1 + n_2$ (feedback connections from layer 2 to layer 1), $\mathbf{M}_{ij}^{hh} \sim B(p_2)$ for $n_1 + 1 \leq i, j \leq n_1 + n_2$ (recurrent connections of layer 2). Note that in a two-layer network the input is only provided to the first layer and activity is only read out from the second layer. To implement this, we masked both the input layer and the readout layer using binary masks as above to only project input to the first layer and to only read out activity from the second layer, respectively.

To simulate the presence and absence of different types of recurrent connections within a single layer HORN network of n nodes, we use the update equation (14) and set the mask matrix $\mathbf{M}^{hh}$ as follows: (i) $\mathbf{M}^{hh} = 0$ for a network consisting of isolated nodes without self-connections, (ii) $\mathbf{M}_{ii}^{hh} = 1$ for $1 \leq i \leq n$ and $\mathbf{M}_{ij}^{hh} = 0$ otherwise for a network of isolated nodes with self-connections, (iii) $\mathbf{M}_{ii}^{hh} = 0$ for $1 \leq i \leq n$ and $\mathbf{M}_{ij}^{hh} = 1$ otherwise, resulting in a fully connected network of nodes without self-connections, and (iv) $\mathbf{M}_{ij}^{hh} = 1$ for all i, j for a fully connected network of nodes with self-connections. We note that case (iv) is equivalent to that of an unmasked network with update equation (5).

Hebbian Learning Rule. To simulate Hebbian weight updates of the recurrent connection matrix $\mathbf{W}^{hh}$, the weights of the input layer $\mathbf{W}^{ih}$ and the read-out layer $\mathbf{W}^{ho}$ were kept plastic for the BPTT algorithm, but the recurrent weights $\mathbf{W}^{hh}$ were frozen. Instead, a simple additive rule was implemented that modifies the weight of a synaptic connection $\mathbf{W}_{ij}$ from node j to node i by an additive term

$$\Delta \mathbf{W}_{ij} = \sigma_{\Delta} \lambda_h a_{ij} r(x_i(t), x_j(t)), \tag{15}$$

where $\sigma_{\Delta} \in \{+1, -1\}$ determines the type of the learning rule ($\sigma_{\Delta} = +1$: Hebb+, $\sigma_{\Delta} = -1$: Hebb-), λ_h denotes the learning rate, $a_{ij} \in \{0, 1\}$ an activity modulator described below, and $r(x_i(t), x_j(t))$ denotes the Pearson correlation between the activity vectors of the nodes i and j computed over the time steps $t = c, \ldots, T$, respectively, where $T = 784$ denotes the stimulus length. The parameter c controls the starting point in time after which the correlation coefficients are computed. During training, the update rule (15) was applied to the entries of $\mathbf{W}_{ij}$ according to the following logic. For each sample, the distribution of pairwise cross-correlation coefficients $r(x_i(t), x_j(t))$ of the evoked HORN activity was calculated for all node pairs. Subsequently, the upper and lower P percentiles of this distribution were determined to identify the most strongly correlated and most strongly anticorrelated nodes, excluding self-connections i, i (the active connections). The activity modulator a_{ij} was set to 0 for all node pairs i, j for which $r(x_i(t), x_j(t))$ did not belong to the set of active connections (the inactive connections). For active connections, the values of a_{ij} were sampled from a binomial distribution $B(p_s)$, where p_s is the selection probability. The simulation was performed using the psMNIST data set, see Table S1 for the base HORN parameters. For all HORNs, we disabled the self-connections in the amplitude as well as velocity terms, as these cannot be meaningfully updated with the correlation-based rules, and also disabled bias vectors for the recurrent connections for the same reason. A grid search on $\lambda_h \in \{5 \cdot 10^{-5}, 10^{-4}, 5 \cdot 10^{-4}, 10^{-3}, 5 \cdot 10^{-3}\}$, $c \in \{0, 300, 600\}$, $P \in \{1, 5, 10, 50\}$, $p_s \in \{0.1, 0.5, 1.0\}$ was performed to determine the optimal parameters for each

case. For homogeneous HORNs, the following parameters were used for the correlation-based learning rules: Hebb+: $\lambda_h = 10^{-4}$, $c = 300$, $P = 5$, $p_s = 0.5$. Hebb-: $\lambda_h = 5 \cdot 10^{-4}$, $c = 300$, $P = 5$, $p_s = 1.0$. For heterogeneous HORNs, the following parameters were used: Hebb+: $\lambda_h = 10^{-4}$, $c = 300$, $P = 5$, $p_s = 0.1$. Hebb-: $\lambda_h = 10^{-4}$, $c = 600$, $P = 5$, $p_s = 0.1$.

Geometric Input. Geometric input was provided to a network by defining non-overlapping receptive fields of size 2×2 pixels and covering the 28×28 intensity matrix of each MNIST digit by 14×14 receptive fields ordered from top left to bottom right of the stimulus matrix in a scanline fashion. The pixel intensities in each of the 196 receptive fields were mapped to 196 DHO nodes that form a fully connected HORN^h with node input connection weights and no input bias in a way that preserves geometric relations between the input receptive fields (so that the first DHO node receives input from the top left receptive field, and the last DHO node from the bottom right receptive field). To simulate a flashed stimulus, the network only received input during the first time step. Following the stimulus presentation period, we let the network dynamics unfold for 150 time steps. The whole network was trained using BPTT on the MNIST digit classification task for a readout at time $T = 150$. We trained a HORN^h network on both the original and a shuffled version of the MNIST digits (with shuffling performed by applying a random but fixed permutation of the 28×28 pixel values of each digit).

Spontaneous Activity. For simulating spontaneous activity in HORNs, we trained heterogeneous HORN networks of size 16 on sMNIST (see Table S.1 for parameter values). We used random initial conditions for each DHO node with both the initial amplitudes x and velocities $\dot{x}$ for $t = 0$ sampled from a Normal distribution $N(0, 30)$. To model spontaneous activity, DHO nodes in the network were subject to random instantaneous deflections to their velocity $\dot{x}$ in the form of delta pulses induced by i.i.d. Poisson processes with a rate of $\lambda = 0.005$ and an amplitude of $\Delta_{\dot{x}} = 5$. To model the transition into and out of a stimulus-driven regime, we simulated network dynamics in three periods. In the first periods, the network was spontaneously active for a period of 1000 time steps, after settling network activity for 500 time steps. In the second periods lasting 784 timesteps,

spontaneous activity was switched off and an sMNIST stimulus was presented. The third period lasting 1000 time steps, represents the time after stimulus offset when the stimulus-dependent input was replaced again by spontaneous activity. PCA analysis was performed on windowed network activity (window length 20 time steps). In particular, windows of spontaneous activity were sampled starting at $t = 500$, and windows of evoked activity starting at $t = 1764$ (i.e. the windows spanning 20 time steps prior to the readout time). PCA was performed using the implementation in the Python package `scikit-learn` (`sklearn.decomposition.PCA`).