Data-based large-scale models provide a window into the organization of cortical computations

Guozhang Chen*, Franz Scherr*, Wolfgang Maass[1]

Institute of Theoretical Computer Science, Graz University of Technology,
Inffeldgasse 16b, Graz, Austria
* Contributed equally.
[1] To whom correspondence should be addressed; E-mail: maass@igi.tugraz.at.

Abstract

Understanding how the brain solves demanding computational tasks is one of the most exciting scientific challenges of our times. So far, recurrently connected artificial neural network models (RANNs) were primarily used to reverse-engineer brain computations. We show that it is now also feasible to reverse-engineer computations of detailed data-based large-scale models of cortical microcircuits. Furthermore, results of these analyses produce hypotheses that can readily be tested in biological experiments since they clarify from which neurons one needs to record and what type of information can be expected at specific time points during a trial. We apply this approach to a demanding visual processing task that has often been used in mouse experiments. Both the cortical microcircuit model and RANNs can solve this task as well as the mouse. But the resulting network dynamics matches only for the cortical microcircuit model experimental data on the sparseness of network activity and the impact of individual neurons on the network decision. Reverse-engineering of the computation in the cortical microcircuit model suggests that a particular subset of neurons causes a bifurcation of the network dynamics that triggers the network decision. Altogether, our results introduce a new type of neural network model for brain computations.

Short title: Reverse engineering of cortical computations

Teaser: Large-scale modeling on supercomputers provides new tools for understanding how the brain computes

1 Introduction

We tackle a central problem in computational neuroscience: How do neural networks of the neocortex compute? A major insight into brain function was the discovery that the mammalian neocortex is in first approximation a continuous 2D sheet consisting of rather stereotypical local circuits that consist of many different types of neurons that are located in 6 parallel layers (laminae) with numerous vertical connections between these layers and primarily local horizontal connections (Mountcastle 1998; Douglas and Martin 2004; Harris and Shepherd 2015). This architecture offers hope that one can understand how the neocortex solves demanding computational tasks by understanding the organization of computations in a representative patch of the 2D sheet that makes up the neocortex. Such representative patches are

usually referred to as cortical microcircuits, in spite of the fact that models for them are typically quite large in order to provide a representative picture of interconnection patterns for many different types of neurons. Intense research in several labs on the anatomy of cortical microcircuits (Mountcastle 1998; Thomson and Lamy 2007; Markram et al. 2015) and subsequent further work at the Allen Institute has recently culminated in a detailed publicly available model for a cortical microcircuit consisting of 51,978 neurons (Billeh et al. 2020). We refer to this model in the following as the Billeh et al. model.

We address the fundamental question how this cortical microcircuit model computes. More precisely, since it is a model for a patch of V1 and comes together with a model for the LGN that serves as a gateway for visual input from the retina to V1, we examine how the model of Billeh et al. solves a demanding computational task that has frequently been used in experimental studies for visual processing in the mouse (Garrett et al. 2020; Joshua H. Siegle et al. 2021): the visual-change-detection task. In this task the subject receives a long sequence of natural images, with intermediate phases where just a gray screen is shown. The task is to report after each image presentation whether it differs from the previously shown image. This is a really demanding computational task since complex natural images are shown. Furthermore, the computational performance is tested for novel images that never occurred during training. Since a neural network is not likely to be able to retain all pixel values of the preceding image in its working memory, it has to adopt a more sophisticated strategy that amounts to extracting and retaining features of natural images that are generally useful for telling images apart, even for novel images.

We trained both the data-based model of Billeh et al. and a generic RANN of the same size with the same training method, stochastic gradient descent, to solve this computational task. Successful application of stochastic gradient descent is less standard for a network of spiking neuron models that have been fitted to biological data. But it can be made to work with a suitable modification of BPTT (backpropagation through time) with pseudo-derivatives for spiking neurons as in (Bellec et al. 2018), and further modifications from (Chen et al. 2022) for the more complex generalized leaky integrate-and-fire ($GLIF_3$) neuron models from Billeh et al. that had been fitted to recordings from diverse neurons from the Allen Brain Atlas Allen Institute 2018. Still, the application of stochastic gradient training to such data-based cortical microcircuit models is computationally substantially more demanding than for RANNs. But it becomes feasible through the use of advanced software, TensorFlow (Martin Abadi et al. 2015), and computer hardware (GPUs) that have been developed to support fast training of artificial neural networks. Note that the connectivity structure of the cortical microcircuit model was not changed through this training process, only the values of synaptic weights within a biologically reasonable range.

We adopt a biologically realistic convention for extracting the network decision for each image, change or no-change, from the model: Network decisions have to be reported by a rather small subset of pyramidal cells on layer 5 of the model that represent projections of V1 to subcortical targets. This modeling convention is supported by experimental data of (Houweling and Brecht 2008; Marshel et al. 2019) and others which showed that stimulation of just 1 or 2 pyramidal cells on layer 5 suffices for triggering a behavioral response. This readout convention has a substantial impact on the computational analysis of a cortical microcircuit model, since the customarily used linear readout from all neurons in the network tends to mask the computational contribution of the network itself, as we will show.

We find that the computation for the visual-change-detection task achieves in the model a similar performance as in-vivo. Furthermore, each computation engages the network in a way that has been reported in numerous experimental data on cortical computation but which provides a stark contrast to typical computations in generic recurrent artificial neural network models (RANNs): Neural activity is very sparse, and therefore implemented in a very energy-efficient manner, with most neurons firing mostly during a rather short phase within a trial. Furthermore, the temporal order of their peak activity is different for different task conditions. In addition, a surprisingly small subset of neurons extracts from the currently presented image the information whether it agrees with the previous one, and controls the

bifurcation of the trajectory of network states, thereby triggering the network decision. The sensitivity of the network decision to the activity of very small subsets of neurons is another characteristic feature of computations of biological neural networks (Houweling and Brecht 2008; Doron et al. 2014; Marshel et al. 2019; Dalgleish et al. 2020; Doron et al. 2020) which is reproduced by the model of Billeh et al. but not by RANN control models of the same size that are trained for the same task. This result provides evidence that the model of Billeh et al. operates in a critical regime, in spite of its very sparse activity, where it is highly sensitive to even a few spikes of individual neurons in the network. A closer analysis of these pivot neurons reveals that most of them have slowly changing internal variables, which most biological neurons have according to the Allen Brain Atlas (Allen Institute 2018), that provide implicit information about the preceding image while the network processes the current one.

We expect that the analysis and supercomputing methods that we present pave the way for research on a new generation of models in computational neuroscience that integrate a substantially larger body of experimental data, and can reproduce features of cortical computations that are difficult to reproduce in artificial neural network models. Also, their predictions can be tested more directly through biological experiments because neurons in the model can be immediately related to the specific types and laminar locations of neurons that are examined in wetlab neuroscience. Going forth and back between detailed modeling and biological experiments is likely to be needed to elucidate the computational function of the neocortex. Since the neocortex achieves its superior computational performance with an energy consumption that is by several orders of magnitudes lower than that of current computer hardware, an understanding how the cortex is able to combine energy-efficient very sparse activity with superior computational performance is likely to have also important technological implications.

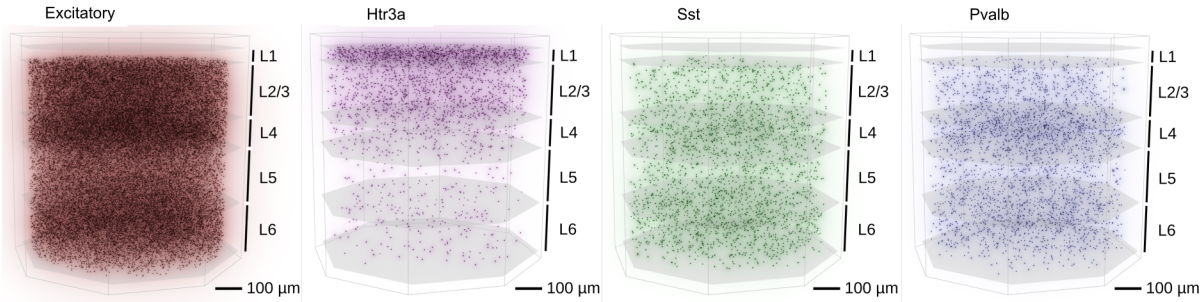
2 Results

2.1 Setting up a data-based V1 model for reverse engineering of cortical computations

The V1 model of (Billeh et al. 2020) represents one of the most comprehensive efforts to integrate the available experimental data on the anatomy and neurophysiology of area V1 in mouse that is currently available (Fig. 1A-C). It distinguishes 17 different neuron types (listed in each row and column of Fig. 1B). These neuron types are further split into 111 different variations based on response profiles of individual neurons from the Allen Brain Atlas (Allen Institute 2018), to which generalized leaky integrate-and-fire ($GLIF_3$) neuron models with 3 internal variables had been fitted. These neuron models have in addition to the membrane potential two further internal variables that model after-spike currents (Fig. 1D). The resulting model for a patch of V1 receives visual input from an LGN model that consists of 2,589 filters (Billeh et al. 2020) that had been fitted to experimental data. This LGN model produces input currents to neurons of the V1 model in a retinotopic and lamina-specific manner (Billeh et al. 2020). We will refer in the following to this model of V1 in conjunction with the LGN model of (Billeh et al. 2020) as the V1 model of Billeh et al.

We employed a data-driven noise model based on experimental data from area V1 of the awake mouse (Stringer et al. 2019). This noise model was not present in (Billeh et al. 2020) and had subsequently been introduced in (Chen et al. 2022). It models both quickly changing forms of noise and slower forms of noise that contribute to experimentally found trial-to-trial variability (Methods). The resulting V1 model was trained to solve the visual-change-detection task, which has frequently been used in mouse experiments (Garrett et al. 2020; Joshua H. Siegle et al. 2021). A sequence of natural images was presented to the model, interleaved by periods without visual input. The subject had to report whenever

A

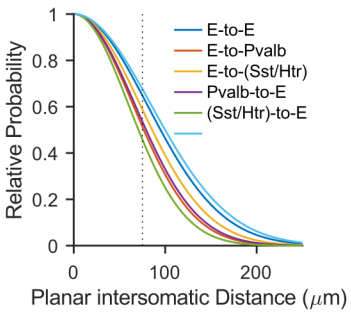


B

Connection probabilities at 75 µm

Trg / Src	i1Htr3a	E2/3	i2/3 Pvalb	i2/3 Sst	i2/3 Htr3a	E4	i4 Pvalb	i4 Sst	i4 Htr3a	E5	i5 Pvalb	i5 Ssst	i5 Htr3a	E6	i6 Pvalb	i6 Sst	i6 Htr3a
i1Htr3a	0.656	0.356	0.093	0.068	0.464	0.148				0.148	0	0		0.148			
E2/3	0	0.16	0.395	0.182	0.105	0.016	0.083	0.083	0.083	0.083	0.081	0.102		0			
i2/3Pva	0.024	0.411	0.451	0.03	0.22	0.05	0.05	0.05	0.05	0.07	0.073						
i2/3Sst	0.279	0.424	0.857	0.082	0.77	0.05	0.05	0.05	0.05	0.021		0					
i2/3Htr	0	0.087	0.02	0.625	0.028	0.05	0.05	0.05	0.05	0							
E4		0.14	0.1	0.1	0.1	0.243	0.43	0.571	0.571	0.104	0.101	0.128	0.05	0.032			
i4Pvalb		0.25	0.05	0.05	0.05	0.437	0.451	0.03	0.22	0.088	0.091	0.03	0.03				
i4Sst	0.241	0.25	0.05	0.05	0.05	0.351	0.857	0.082	0.77	0.026	0.03	0	0.03				
i4Htr3a		0.25	0.05	0.05	0.05	0.351	0.02	0.625	0.028	0	0.03	0.03	0.03				
E5	0.017	0.021	0.05	0.05	0.05	0.007	0.05	0.05	0.05	0.116	0.083	0.063	0.105	0.047	0.03	0.03	0.03
i5Pvalb	0	0	0.102			0	0.034	0.03	0.03	0.455	0.361	0.03	0.22	0.03	0.01	0.01	0.01
i5Sst	0.203	0.169		0.017		0.056	0.03	0.006	0.03	0.317	0.857	0.04	0.77	0.03	0.01	0.01	0.01
i5Htr3a						0.03	0.03	0.03	0.03	0.125	0.02	0.625	0.02	0.03	0.01	0.01	0.01
E6		0				0				0.012	0.01	0.01	0.01	0.026	0.145	0.1	0.1
i6Pvalb		0.1				0.1				0.1	0.03	0.03	0.03	0.1	0.08	0.1	0.08
i6Sst										0.03	0.03	0.03	0.03	0.1	0.05	0.05	0.05
i6Htr3a										0.03	0.03	0.03	0.03	0.1	0.05	0.05	0.03

C



D

Dynamics:

$$V'(t) = \frac{1}{C}\left(I_e(t) + \sum_j I_j(t) - \frac{1}{R}\left(V(t) - E_L\right)\right),$$

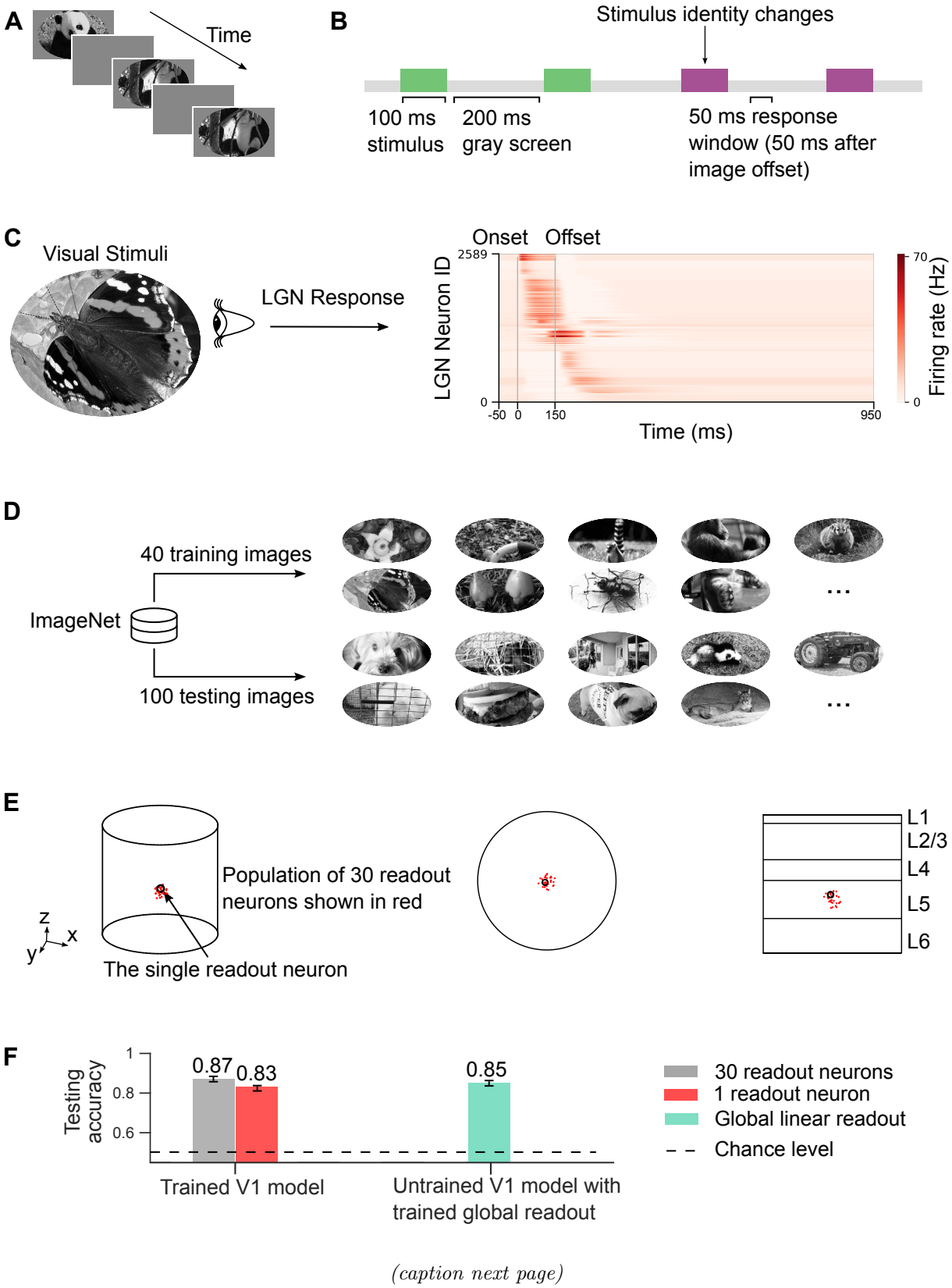
$$I_j'(t) = -k_j I_j(t);\ j = 1, 2,$$

Spike $V(t) > \Theta_\infty$:

$$V(t_+) \leftarrow E_L + f_v \times \left(V(t_-) - E_L\right) - \delta V,$$

$$I_j(t_+) \leftarrow f_j \times I_j(t_-) + \delta I_j,$$

Figure 1: Structure and components of the data-based cortical microcircuit model of (Billeh et al. 2020). **(A)** Side view of the 3D architecture of the model, which consists of with 51,978 neurons from 1 excitatory and 3 inhibitory neuron classes (Htr3a, Sst, Pvalb) in a column with 800 µm diameter. These neurons are distributed over 5 laminar sheets (L1 to L6, where L2 and L3 are lumped together). **(B)** Base connection probabilities between these neuron classes on different laminae, which are valid if the horizontal distance between neurons is at most 75 µm. **(C)** Scaling function for connection probabilities in dependence of the class to which the pre- and postsynaptic neuron belongs. The probability of a synaptic connection between the two neurons is the product of the base connection probability from panel **(B)** and this scaling function. **(D)** Main equations defining the GLIF_3 neuron models of (Billeh et al. 2020) with 3 internal variables. Assignments to their parameters (highlighted in red) define the 111 neuron models of the networks, based on experimental data from the Allen Brain Atlas (Allen Institute 2018).



(caption next page)

Figure 2 ***(previous page)*****: Visual-change-detection task. (A)** A sequence of natural images is presented, interleaved with gray screen periods. **(B)** The visual-change-detection task requires to give during a 50 ms response window that begins 50 ms after image offset an output signal whenever the current image is different from the preceding one. **(C)** We used the LGN model from (Billeh et al. 2020) to transform visual stimuli into the outputs of 2,589 filters that model firing rates of LGN neurons and are connected to neurons in the V1 model according to data-based rules. These outputs provide input currents to V1 neurons using data-based rules. **(D)** We used 40 natural images for training and a separate set of 100 natural images for testing. Images were drawn from ImageNet (Russakovsky et al. 2015) and presented in grayscale because of the insensitivity of LGN model of Billeh et al. to color. To ensure the robustness of our findings, we trained 10 V1 models using 10 different sets of training data (each with its own 40-image set). We found that all results presented below were consistent across all models, demonstrating that our conclusions are not reliant on specific training datasets or coincidences. **(E)** The network was trained to provide the network decision through projection neurons within the network, either a single (black circle) or 30 (red dots) randomly selected pyramidal cells in layer 5 (within a sphere of 55 µm) depending on the experiment. Their task was to report an image change through an increased sum of firing rates during the response window. **(F)** Testing accuracy of the trained V1 model of Billeh et al. for reporting network decisions through increased firing activity of 1 or 30 projection neurons is shown on the left. One sees that the chosen number of projection neurons has little impact on the network performance. The green bar on the right shows a control result for the case when one uses instead of projection neurons from within the network a global linear readout from all neurons in the network. One sees that in case it is not even necessary to train the V1 model for the task: Training of the weights of the artificial global linear readout suffices, indicating that the computation within the V1 model can be effectively masked by such a global linear readout. The error bars, which are small (< 0.01), represent the standard error of the mean (SEM) across 10 models with different training datasets.

the most recently presented image differed from the one before (Fig. 2A-C). We randomly selected a pool of 140 natural images from the Imagenet dataset (Deng et al. 2009) that we used as network inputs. We used 40 of them for training, similar to the biological experiments of (Garrett et al. 2020), and 100 of them for testing (Fig. 2D).

Projections neurons on layer 5 of cortical microcircuits extract computational results of the microcircuit and transmit them to other brain areas, thereby triggering behavioral responses (Harris and Shepherd 2015; Marshel et al. 2019). Therefore we selected a set of pyramidal cells on layer 5 as readout neurons of the V1 model (Fig. 2E). These readout neurons had the task to fire during a specific response window after an image presentation if the preceding image differed from the one that had been presented before that. The size and spatial distribution of this readout population had little impact on the results (Fig. 2F). For simplicity, we used a single readout neuron unless stated otherwise.

A more common output convention in modeling brain computations is to use an external readout neuron that receives synaptic input from all network neurons. We found that this convention is not suitable for probing the computational capability of a network model. First, such global readout neurons that receive synaptic inputs from all neurons in a large patch of the neocortex have not been found in the brain. Secondly, a global linear readout neuron that receives synaptic inputs from a large set of neurons in a recurrent neural network through trained synaptic weights is a too-powerful device that masks the computational contribution of the recurrent neural network. Instead, the recurrent neural network plays in this case just the role of a liquid or reservoir (Maass et al. 2002; Maass and Markram 2004). Concretely, if one takes the V1 and LGN model as defined in (Billeh et al. 2020), without changing any of its synaptic weights or other parameters, and just trains a linear readout from all of V1 neurons for the visual-change-detection tasks, one gets already a very high average accuracy of 0.85 (see the rightmost bar in Fig. 2F).

We trained all synaptic weights from the LGN to V1 and within the V1 model using stochastic gradient descent for a suitable loss function that assumed a low value only when the readout neuron(s) fired within

a short response window 50 ms after the presentation of an image in case that this image differed from the preceding one (Methods). We included regularization terms similarly as in (Chen et al. 2022) in the loss function in order to keep the firing activity of the network in a biologically realistic sparse firing regime. Synaptic connectivity was not changed through this training process. We also did not allow synaptic weights to change their sign, thereby obeying Dale's law. In particular, the average firing rate after training was 3.86 Hz. Hence the model computed in an energy-efficient sparse firing regime. The new values of synaptic weights after training remained in a biologically reasonable range, see Fig. S1 and S2.

The trained V1 model achieved high performance for the visual-change-detection task (Fig. 2F), lying in the same range as the performance achieved by mice (Garrett et al. 2020). The model was also able to generalize well, achieving almost the same performance for images that were not used during training (Fig. 2F). Hence the model had acquired a network algorithm that generalized, i.e., was not constrained to a particular set of previously seen images.

2.2 The data-based V1 model reproduces characteristic features of cortical computations

Simultaneous recordings from large numbers of neurons in the awake brain show that neural networks of the neocortex exhibit a peculiar type of network dynamics that is rarely seen in artificial neural networks: Most neurons fire only during a rather short time window during a trial, see e.g. (Driscoll et al. 2017) and Fig. 2 of (Koay et al. 2022). Furthermore, different neurons have different preferred firing times, and the relative order in which they fire depends on the task condition and the sensory input. Hence, similarly as in synfire chain models (Abeles et al. 2004), the network activity has a prominent sequential character, but in contrast to synfire chain models only a very small fraction of neurons is active during each segment of the sequence.

Since this type of network activity is hard to reproduce in neural network models, we wondered whether the data-based V1 model would be able to do that. We considered trial-averaged neural activity as in (Koay et al. 2022) and followed the same routine to order the neurons according to the time of the peak activity. Furthermore, as in their data analysis we normalized the firing activity, averaged over 200 trials, of each neuron over the 300 ms of the computation on each image, consisting of a segment of the continuous input- and processing stream that contained the 100 ms of the image presentation, a 50 ms delay, the subsequent 50 ms of the network response window, and a 100 ms delay before the presentation of the next image, marked at the top of Fig. 3 A-C. We found that the neural activity in the V1 model was indeed very similar to that in the experimental data. Fig. 3 A-C show that most neurons of the V1 model participated in the network computation, but focused their firing activity to a very short segment of the 300 ms time window. Furthermore, as in the experimental data, the relative order of these short segments of high activity depended on whether the currently processed image was the same as the preceding one or not, see Fig. 3. The neurons are plotted in panel C in the same order as in panel A, but in panel A for the change condition and in panel C for the no-change condition. Panel B, where we have ordered the neurons according to their preferred firing time for the no-change condition, shows that the neurons have also for the no-change condition a clear order of preferred firing times. But the order is different from that for the change condition, as panel C shows: The resulting sequence is blurred and lacks the characteristic thin-line pattern observed in Fig. 3A and B. In other words, the network employed temporal coding through the relative timing of the peak-firing activity of neurons for distinguishing between these two experimental conditions. Furthermore, this temporal coding was simultaneously expressed in all layers of the laminar cortical microcircuit model. Note that this type of temporal coding, through the order to peak activity of different neurons, takes place in spite of substantial noise both in the brain and in our V1 model, see (Chen et al. 2022) for details, that substantially affects the timing of individual spikes in

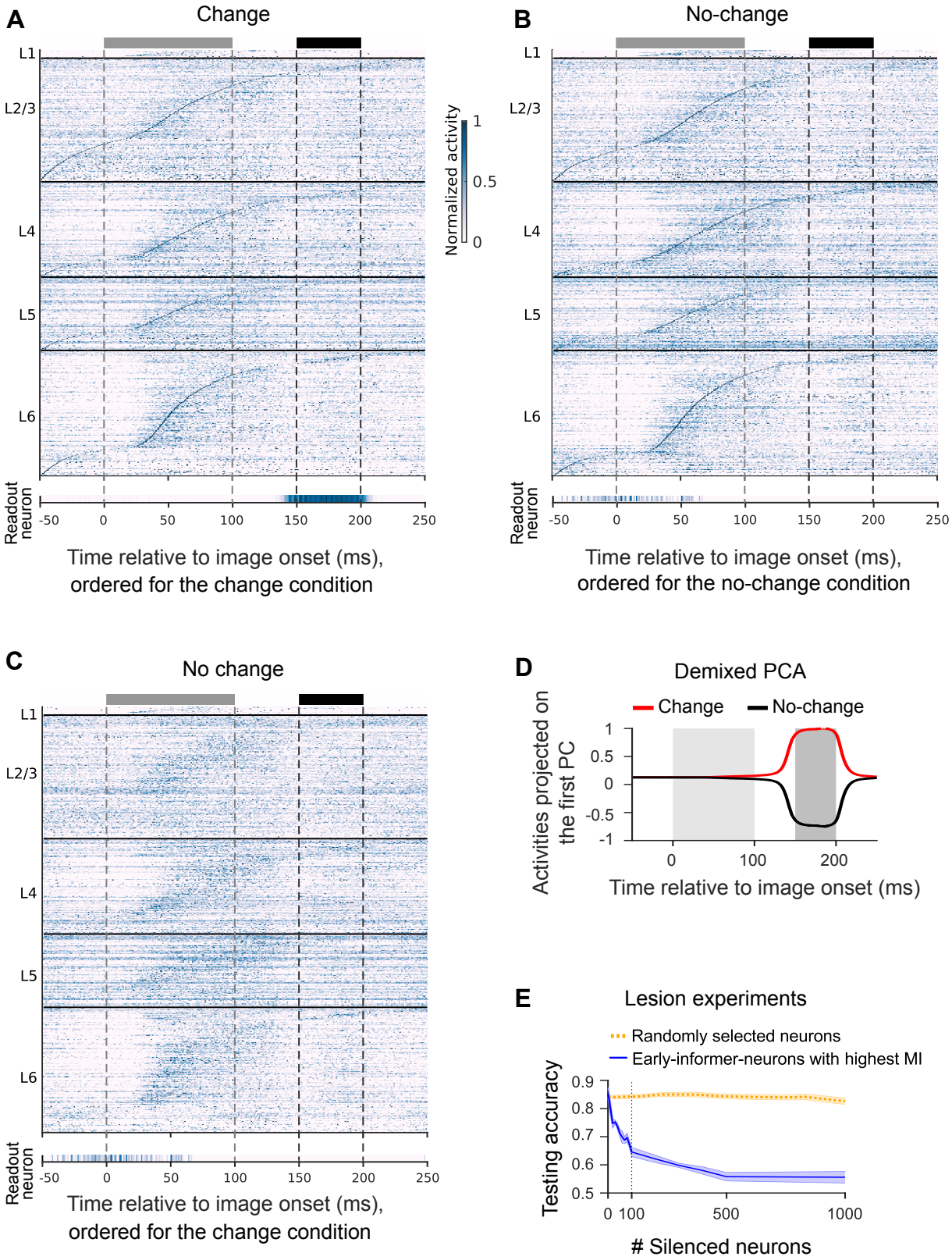
a single trial. The V1 model predicts that cortical microcircuits employ an even more refined type of temporal coding: The order of peak activity also depends on the identity of the current image (Fig. S3). This prediction needs to be tested through neurobiological experiments.

The experimental data of (Houweling and Brecht 2008; Doron et al. 2014; Marshel et al. 2019; Dalgleish et al. 2020; Doron et al. 2020) have elucidated another characteristic feature of cortical computations: The network computation is surprisingly sensitive to the activity of a tiny fraction of neurons of these very large networks, since stimulation of a few selected neurons could change the result of the network computation, i.e., the behavioral response. This high sensitivity of the neocortex on the activity of particular neurons is surprising insofar as it has to cope with a substantial amount of noise, see the analysis and resulting noise model of (Chen et al. 2022) that we also used in this study. Since artificial activation of a small set of neurons in a network is likely to activate also other neurons, a factor that had not been quantified in these biological experiments, we used a slightly different paradigm for testing the sensitivity of the network decision of the V1 model to the activity of a small set of neurons: We silenced, rather than activated, selected neurons. Like in the experimental data on the neocortex, we found that the result of the network computation is not equally sensitive to the activity of all of the neurons, but that there are particular neurons that are pivotal for the network decision. In order to avoid artifacts resulting from silencing of the readout neurons or neurons that directly activate them, we focused on early-informer-neurons which produced the first information on whether the current image was the same as the preceding one well before the response window, while the image was still processed. Fig. 3D shows that the first information about this arises during the time window from 50 to 100 ms after image onset, when the first information about the image reaches the V1 part of the model. Hence we defined early-informer-neurons as neurons whose spike output contained already during this time window substantial mutual information (MI, Methods) with the upcoming network decision. Fig. 3E shows that while the network is not very sensitive to silencing of randomly selected neurons, silencing of just 100 early-informer-neurons has a drastic impact on the network performance. Also, the number of readout neurons does not affect the conclusion of the lesion experiment (Fig. S4).

2.3 RANN models cannot reproduce these characteristic features of cortical computations

Randomly connected recurrent networks of artificial neurons (RANNs) have commonly been used as models for computations in cortical neural networks (Sussillo and Barak 2013; Sussillo et al. 2015; Yang et al. 2019; Yang and X.-J. Wang 2020; Pollock and Jazayeri 2020). We show here that RANNs are not able to reproduce the two previously discussed fingerprints of cortical computations: a short period of peak activity for most neurons, a characteristic sequential order of this peak activity according to the trial type, and the sensitivity of the network to the activity of small subsets of neurons. In order to eliminate a possible impact of differences in the network size or training procedure, we trained through stochastic gradient descent a randomly connected RANN with the same number of neurons and synapses as the V1 model of (Billeh et al. 2020), for the same computational task. Furthermore, we used the same preprocessor (the LGN model of (Billeh et al. 2020)) for transforming images into temporally dispersed inputs to random subsets of neurons in the network. We used a standard neuron model from RANN models for neural networks of the neocortex (Sussillo et al. 2015; Pollock and Jazayeri 2020): A non-spiking neuron with tanh as activation function and a membrane time constant of 50 ms. For extracting the network decision we used the same global linear readout from all neurons in the RANN as in these paradigms.

This RANN model was after training able to perform the visual-change-detection task at a higher performance level than the data-based V1 model and the subjects in the neurobiological experiments (Garrett et al. 2020). But it could not reproduce the two previously discussed fingerprints of cortical computations.



(caption next page)

Figure 3 ***(previous page)*****: Temporal organization of computations in the V1 model. (A)** As in the experimental data, neural activity is sparse and exhibits a clear sequential organization with high temporal resolution. Shown are normalized average responses over 200 trials with the change condition but different images, with neurons ordered according to the time of their peak activity under the change condition. The gray and black bars at the top denote the image presentation and response windows, respectively. **(B)** Same as in **(A)**, but for the no-change condition, with neurons ordered according to the time of their peak activity for the no-change condition. **(C)** The same data as in **(B)** for the no-change condition, but with neurons ordered as in **(A)**. The resulting blurred sequence indicates that the order of peak activity of neurons is quite different for the change and no-change conditions. **(D)** Demixed principal components analysis. In order to visualize the formation of the network decision, we carried out demixed principal component analysis for trial-averaged network activity (Methods). Its projection onto the first principal component is shown. One sees that the network decision starts to emerge during the time window from 50 to 100 ms after image onset. The light and dark gray rectangles denote the window of image presentation and response, respectively. **(E)** Causal impact of specific neurons on the network decision: Task performance quickly decreases when early-informer-neurons are silenced (in the order of their MI with the network decision), see the blue curve. On the other hand, task performance is robust to silencing the same number of randomly selected neurons (dotted yellow curve). Both curves show average values for 10 V1 models where different sets of training data were used. The shaded area represents the SEM across 10 models.

Fig. 4 A and B show that neurons of the RANN do not have a similarly short time period of high activity during the computation on the same task as the V1 model. Furthermore, the same analysis of the firing order as in (Driscoll et al. 2017; Koay et al. 2022) does not reveal a substantial dependence of the order of peak activity of neurons on the trial type, see Fig. 4C. We also tested a different way of plotting the activity of the RANN where we did not normalize the activity of each neuron as in the data analyses of (Driscoll et al. 2017; Koay et al. 2022). Resulting plots (Fig. S5) show that neurons in the RANNs have a wide range of different activity levels. But the analysis of the role of temporal order as in (Driscoll et al. 2017), which was also employed in Fig. 3, did still not provide any indication that the temporal order of peak activity in the RANN depended in a significant way on the trial type. Since the activity level of neurons in the RANN depends on the weight of the regularization term in the loss function for stochastic gradient descent, we repeated the training of the RANN with different weights of this regularization term that controls the average activity of neurons in the network (Fig. S6); see Fig. S7 for the same results without normalizing the activity of each neuron. Also in these controls, the neurons of the RANN do not constrain their activity to a short time window like in the experimental data and the V1 model. The same holds for a further control (Fig. S8) where we changed the threshold of the regularization term in the loss function (Methods). These results suggest that it is not possible to reproduce in the RANN model under common configurations the sparse neural activity with trial-type-dependent temporal sequences of neural peak activity that has been found in the neocortex.

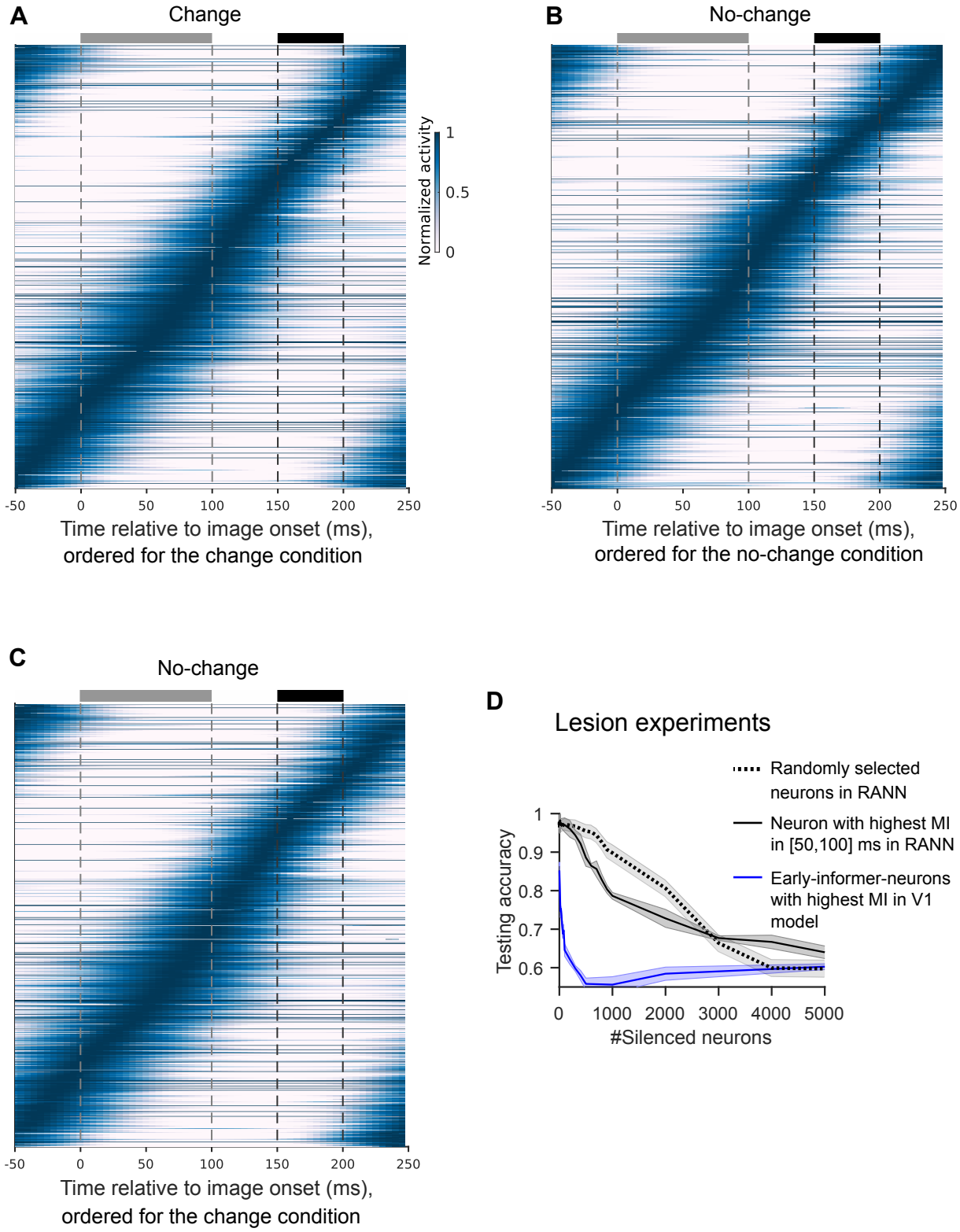
We also tested the sensitivity of the RANN to the activity of small sets of its neurons. We found that about 6,000 neurons need to be silenced in the RANN to reduce the accuracy of the network computation to 0.6 (Fig. 4D), a performance level which was reached in the V1 model according to Fig. 3D by silencing just 200 neurons. Note that we silenced here RANN neurons in descending order of the MI of their activity during the second half of an image presentation with the upcoming network decision, like in the V1 model. The RANN has more high-MI neurons than the V1 model (Fig. S9). These results suggest that the RANN is substantially less sensitive to the activity of small subsets of its neurons.

Altogether, these results suggest that the computation for the visual-change-detection task is organized in the RANN quite differently than in the mouse brain and in the V1 model. In particular, the in-vivo data and the V1 model suggest that neurons become "experts" for particular phases of a particular computation and otherwise remain silent, reminiscent of mixture-of-experts models (Yuksel et al. 2012) and hidden Markov models (Kappel et al. 2014). In contrast, information and impact on the network

output are distributed in the RANN over substantially larger subsets of neurons. This is likely to result from the random connectivity of the RANN, which makes it harder to accumulate specific information in specific parts of the network, the neuron models of the RANN, which do not induce its neurons to restrict their activity to a particular phase of computation, and the use of a global readout from all neurons for extracting the network decision.

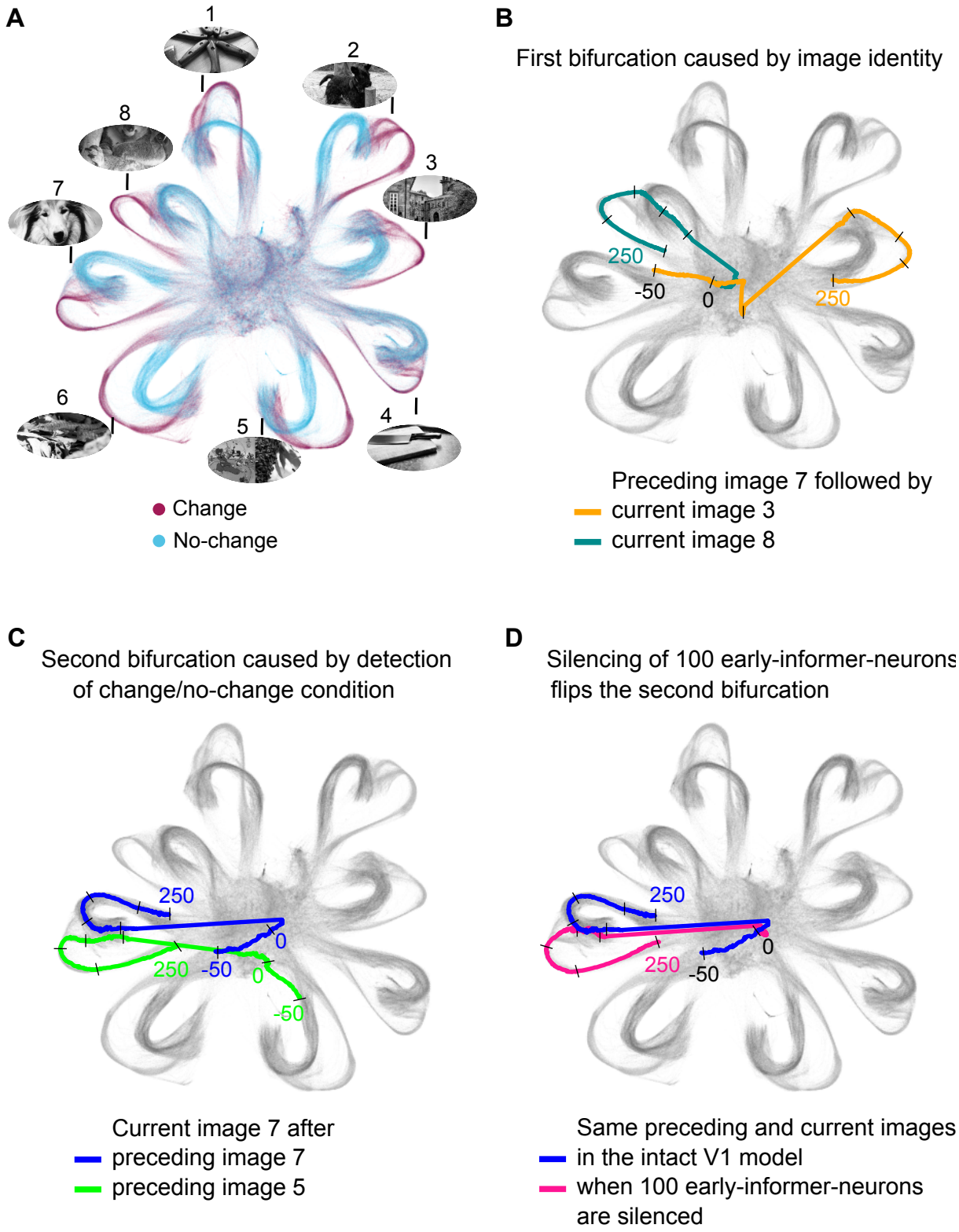
2.4 Computational progress of the V1 model becomes visible as nested bifurcations of the network dynamics

Analyses of results of simultaneous recordings from large numbers of neurons in the brain have shown that low-dimensional projections of the high-dimensional network activity provide interesting links between the network dynamics and the computations that it performs (Broome et al. 2006; Kato et al. 2015; Allen et al. 2017; Steinmetz et al. 2019). We wondered whether similar analyses could elucidate computations of the V1 model. We embedded its activity vectors, defined for every ms by the low-pass filtered spiking activity of its 51,978 neurons, into 2 dimensions with the help of PCA and a subsequent application of UMAP (McInnes et al. 2018). The processing of each particular image gives rise to a bundle of trajectories, with trial-to-trial variability resulting from the preceding images and noise within the network. Two nested bifurcations in these bundles of trajectories mark the computational progress of the dynamical system, see Fig. 5A. First, the trajectories of network states bifurcate during the first 50 ms of an image presentation from the mid-region of the plotted state space and move into a region that is characteristic for the identity of the currently presented image, see Fig. 5B. Afterward, between 50 and 100 ms after image onset, the second bifurcation occurs in dependence on whether the current image is the same or different from the previously presented image, see Fig. 5C. These modeling results provide concrete predictions for the way how these computations are carried out by cortical microcircuits of the brain, viewed as dynamical systems. Among various options on how these could compute, as discussed in (Rabinovich et al. 2006), nested bifurcations of bundles of trajectories emerge as the clearest visible fingerprint of these network computations.



(caption next page)

Figure 4 ***(previous page)*****: Temporal organization of the same computations in the RANN model.** **(A)** Neuronal activity in the equally large RANN model was plotted for the same network inputs and task condition as in Fig. 3A. Here, the activation regularization was not used in the RANN; in other words, the weight of activation regularization is 0 (Methods). **(B)** Same as in **(A)**, but for the no-change condition as in Fig. 3B. **(C)** Same data as in panel **(B)**, but with neurons ordered as in panel **(A)**. In contrast to Fig. 3, little difference emerges between panels **(B)** and **(C)**, indicating that the order of peak activity is less dependent on the task condition in the RANN. **(D)** Lesion experiments corresponding to those in the V1 model (Fig. 3E). The blue curve for the V1 model is the same as in Fig. 3E. One clearly sees that the network decision is substantially less sensitive to the activity of 100 neurons. The shaded areas represent SEM across 10 RANNs where different sets of training data are used.



(caption next page)

Figure 5 ***(previous page)*****: Nested bifurcations of trajectories of network states provide links between network dynamics and network computations.** Spiking activity of the 51,978 neurons was first filtered with an exponentially decaying kernel (time constant: 20 ms), and then projected onto its first 1,500 PCA dimensions (capturing 38% of the variance, which indicates that the network activity is quite high-dimensional). These data were then projected into 2 dimensions with the UMAP method. Each dot represents network activity at a particular ms of a 250-ms long fragment of the computation of the V1 model on a sequence of natural images. Short black bars mark 50-ms long subsections of network trajectories. **(A)** Network trajectories of the V1 model during the presentation of 8 (out of the 100) test images that had not been presented during training. Two colors indicate whether the current image was identical (blue) or different (brown) from the preceding image. One sees that the network state moves for each image into a different region of the state space, no matter whether it was the same or different from the preceding image. **(B)** The first bifurcation occurs according to the identity of the current image, highlighted here for the case where the trajectory starts in both cases from the same state, which largely results from the identity of the preceding image. This first bifurcation occurs within the first 50 ms after image onset. **(C)** The second bifurcation after image onset. It does not depend on the identity of the current image, but on whether it is the same or different from the preceding image. This 2nd bifurcation is more difficult to visualize since the trajectory arrives from two different regions that are characteristic of the identity of the preceding image. **(D)** A small set of early-informer-neurons is causal for the second bifurcation. Here two times the image 7 is shown, both in the intact model and when 100 early-informer neurons are silenced. One clearly sees that these neurons are causal for the second bifurcation occurring within 50 to 100 ms after image onset, since silencing them lets the trajectory flip to the bundle for the no-change condition. Silencing of these neurons also flips a trajectory for the no-change condition to the bundle for the change condition, see Fig. S10.

2.5 Causal relations between the activity of individual neurons and the network decision

The results of our lesion experiments in Fig. 3E indicate that the firing activity of a specific small set of neurons was causal for the network decision. These early-informer-neurons were distinguished by the fact that their firing activity during 50 to 100 ms after image onset contained substantial MI with the upcoming network decision. We show in Fig. 5D that their firing activity was also pivotal for the network dynamics. Silencing the 100 early-informer-neurons with the highest mutual information (MI) with the upcoming network decision flips the trajectory of the network dynamics at the 2nd bifurcation to another bundle, see the magenta curve in Fig. 5D. This suggests that these bundles of trajectories had a certain attractor quality for time-varying network states. The model received for both trajectories shown in this panel exactly the same network input, hence differences were only caused by the silencing of the 100 early-informer-neurons (with some further variance possibly caused by ongoing noise in the network). This was a no-change trial, as the blue curve in Fig. 5D indicates for the intact network (compare with the trajectory bundles in Fig. 5A). Silencing of these 100 neurons can also flip the trajectory of a change trial to the bundle of trajectories for no-change trials, see Fig. S10.

We then investigated what mechanism enabled these 100 early-informer-neurons to decide whether the currently presented image differed from the previously presented one. These neurons were primarily located in layers 2/3 and 4, see Fig. 6A. We selected 7 of them (Methods) from the 4 major neuron classes so that these had during the interval from 50 to 100 ms after image onset the largest MI with the subsequent network decision. Their firing activity is shown in Fig. 6B. One sees that they fired at different rates during the interval from 50 to 100 ms after image onset for the change and no-change conditions. These rate differences continued to be present during the subsequent delay and response window.

In order to determine the mechanism by which these neurons acquired their early information about the relationship between the identity of the current and preceding image we analyzed the dynamics of their internal variables. Fig. 6C depicts the time course of their internal variable with the largest time constant. One sees that this internal variable had in many trials for 5 of these neurons already at the beginning of the presentation of the current image (indicated by the beginning of the light-gray zone) a strongly negative value. In order to understand the computational role of these internal variables, we analyzed whether their strongly negative values at image onset provided information about the identity of the preceding image. The result of this analysis is plotted for the first 4 of these neurons, whose locations are marked in Fig. 7A, in Fig. 7B. One sees that for those 3 among these 4 neurons that had an internal variable with a large time constant, shown here in the 2nd to 4th column, this slow internal variable assumed its lowest values at image onset when a particular image (number 5) had been presented before. Note however that these neurons could not specialize in reporting this particular preceding image, because the network was tested on new test images that had not been shown during training. Therefore, in order to serve as early-informer-neurons, the image features that produced the lowest values of their internal variables at the beginning of the next image had to tile the image space. The neuron whose analysis is plotted in the first column has only short time constants, and the value of its internal variable is less characteristic for a particular preceding image. Hence it is likely to collect and transmit information that it receives from other early-informer-neurons. In order to further elucidate the causal relation between the network decision and preceding activity (or non-activity) of early-informer-neurons with and without long time constants, we separately silenced from each of these two neuron classes those which had the largest MI with the network decision. The result in Fig. 7C shows that the network performance is significantly reduced when we silence 100 neurons with long time constants, and more than 500 neurons with short time constants need to be silenced to produce a similar reduction of network performance.

The impact of early-informer-neurons on the firing or non-firing of the readout neuron was in general rather indirect, because they were in general not directly connected to the readout neuron (Fig. S11).

This had to be expected, because the response window came 100 ms after the critical period of early-informer neurons (50 to 100 ms after image onset). Nevertheless, a direct impact of silencing the 100 early-informer-neurons on the membrane potential of the readout neuron can be detected. The result is shown in Fig. 7D. One sees in the 2nd and 3rd row show that this silencing significantly moved the membrane potential of the readout neuron during the response window, thereby explaining why their silencing caused errors in the network decision.

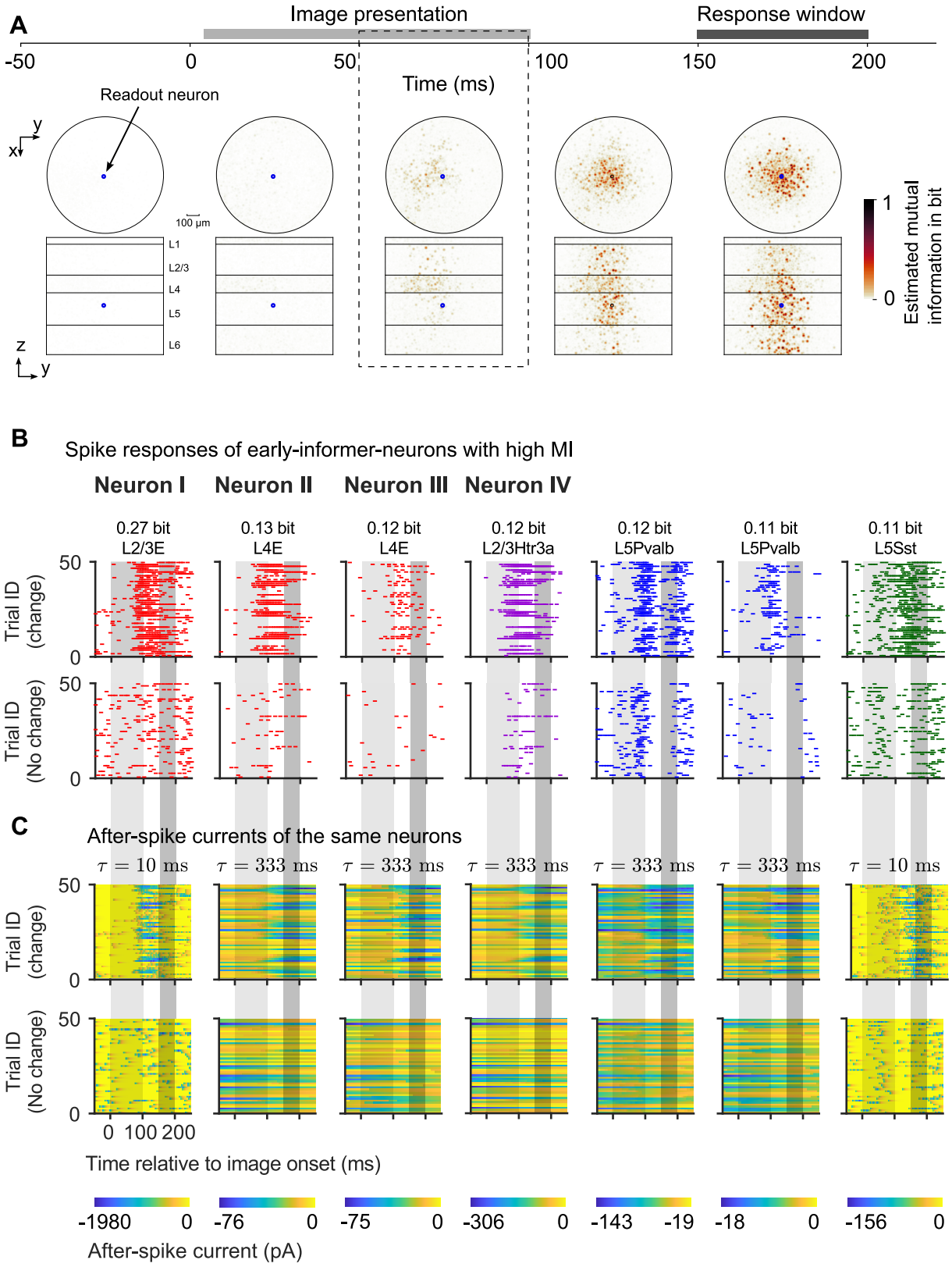
We have shown that the silencing of 100 neurons in the V1 model is able to change the result of a network computation. This high sensitivity of the network output to the firing activity of individual neurons is consistent with experimental data. More specifically, it has been shown that artificial activation of very few neurons on layer 5 in the brain is able to switch a behavioral decision (Houweling and Brecht 2008; Doron et al. 2014; Marshel et al. 2019; Dalgleish et al. 2020; Doron et al. 2020). Note that it is difficult to estimate how many further neurons were indirectly activated in these in-vivo experiments, hence it remains open exactly how many neurons need to be manipulated in order to switch a network decision of a cortical microcircuit.

Finally, we would like to emphasize that the computational analyses in Figs. 3, 5 and 6 were carried out for a new set of images that had not been shown during training of the network. Hence our reverse engineering has elucidated a generic computational network mechanism, rather than a mechanism that only works for specific images. In particular, the diverse selectivity of the working memory of these neurons shown in Fig. 6C in combination with their causal role for the network decision demonstrates that their working memory specializations tile the image space, thereby enabling correct network decisions for generic images.

3 Discussion

We have presented a new paradigm for modeling and analyzing computations in the neocortex. More specifically, we have trained a detailed model for a patch of neocortex with a diameter of 800 µm to carry out a demanding computational task that has often been used in mouse experiments: Deciding for a sequence of natural images, interleaved with delays where no image is shown, whether the most recent image was the same as the preceding one. The V1 model was able to solve this task after training on one set of images, like the mouse, and also for never presented new images with high accuracy. This task is a demanding computational task for any neural network since salient information needs to be extracted from each image and stored, compared with information from the next image while simultaneously storing information from this next image that will be needed to compare it with the subsequent image, and the result of the comparison has to be reported at a time when none of the images are present. This task is especially demanding for a neural network that has to cope with a substantial amount of internal noise, which is the case both for neural networks in the brain and our model.

We have shown that a detailed model for a patch of V1 (Billeh et al. 2020), equipped with a data-based and more challenging noise model from (Chen et al. 2022), can be trained with stochastic gradient descent to solve this visual-change-detection task. We found that the resulting organization of its computation differed in essential aspects from that of a RANN with the same number of neurons and synapses that has been trained with stochastic gradient descent for solving the same task. In particular, the computations of the data-based cortical microcircuit model revealed an exquisite temporal organization, where most neurons focused their activity onto a particular phase of the computation, as seen in Ca-imaging data from the awake brain (Driscoll et al. 2017; Koay et al. 2022). The V1 model also reproduced the finding of (Driscoll et al. 2017) that this temporal order of peak activity depends on the trial type (in our case: change or no-change), see Fig. 3. In addition, the rank order of peak activity depended in the data-based



(caption next page)

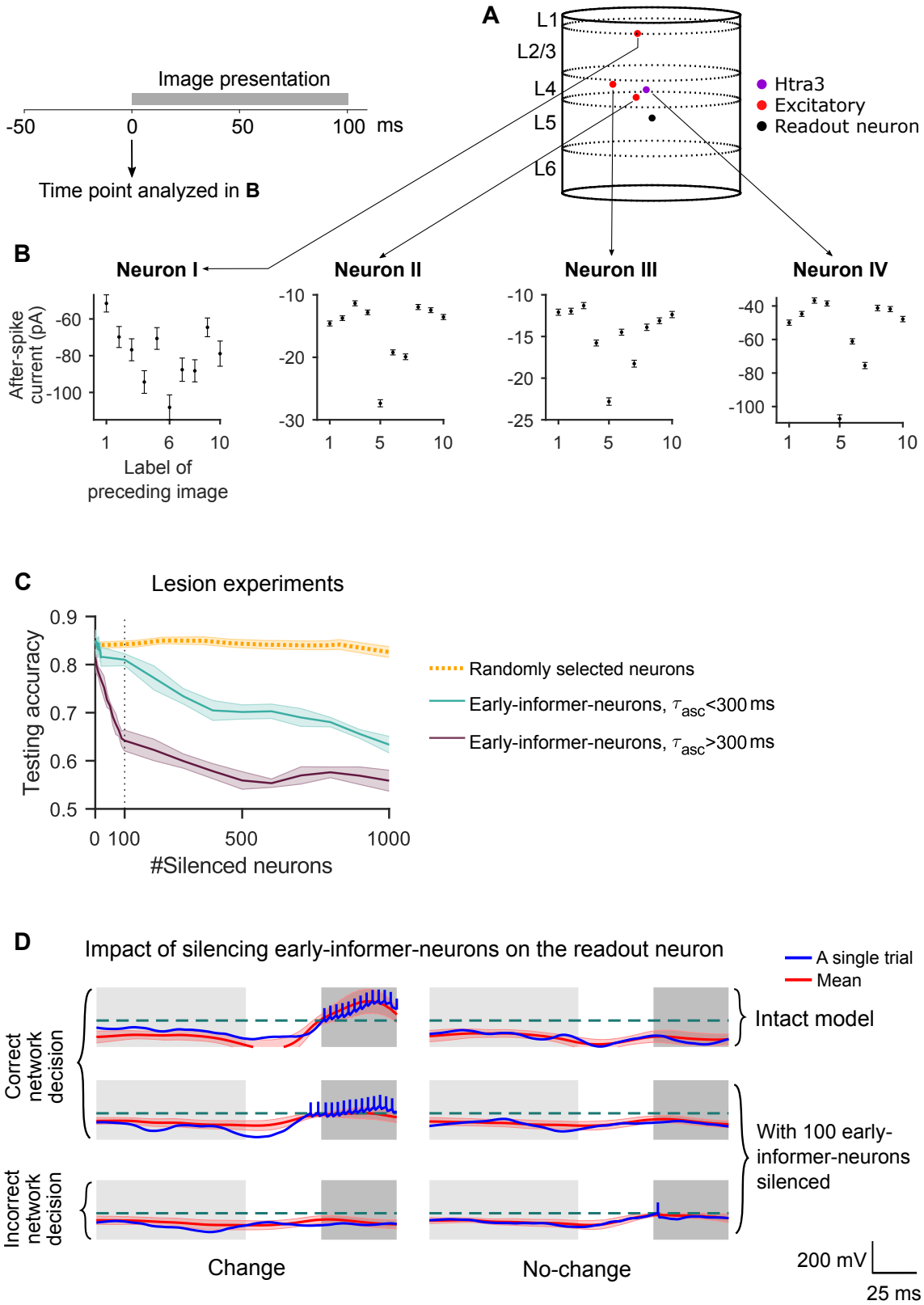
Figure 6 ***(previous page)*****: Fine-scale analysis of the emergence of a network decision in the V1 model. (A)** The mutual information (MI) between activities in 50 ms windows of single neurons and the change/no-change decision of the network (as emerging during the subsequent response window) is estimated for each neuron. For neurons that overlap in the projection from 3D to 2D, the maximum value is visualized to avoid dark points arising from accumulation of small contributions from several neurons. The critical time period from 50 to 100 ms after image onset is marked by dashed lines. **(B)** Spike responses of 7 early-informer-neurons, selected to cover all four basic neuron types: excitatory (E), Htra3, Sst, and Pvalb neurons (color-coded as in Fig. 1A). L2/3, L4, and L5 represent layer 2/3, 4, and 5, respectively. The value of the MI of the firing activity of each of these neurons during $[50, 100]$ ms with the network decision is indicated in bits at the top of each panel. The periods of image presentation and subsequent response window are shaded in grey. Trials are separated according to the change/no-change condition. Condition-dependent differences in the firing responses of these neurons first appear at the start of the critical time window, 50 ms after image onset. **(C)** Time course of the values of the internal variables of these neurons that had the longest time constant (and modeled after-spike currents), for the same trials as in **(B)** (time constant shown on top). As GLIF_3 neurons have two after-spike currents that could have different time constants, we only show the after-spike currents with the largest time constant. Their values at image onset could potentially contain information about the identity of the preceding image, shown 200 ms before the current one. This information will be analyzed in the next figure.

model not only on the trial type, but also on the identity of the current image (Fig. S3). This additional dependence of the temporal order of peak activity on the sensory input (for the same trial type) is a prediction for future neurobiological experiments.

We found that the RANN was not able to reproduce the experimental data on the temporal organization of cortical computations: Most of its neurons were active during rather long segments of a trial (see Fig. 4A), and a fairly large fraction of neurons was simultaneously active at the same time. The temporal order of their periods of high activity appeared to have a stereotypical character (see Fig. 4B, C) with no apparent dependence on the trial type.

Apart from this different temporal organization of the network computation, the data-based model also reproduced another characteristic feature of cortical computations: In spite of the large number of neurons in the model (about 52,000), a rather small subset of neurons had a decisive impact on the time course and outcome of the computation. Silencing of 100 neurons was sufficient to switch the trajectory of network states during the computation (Fig. 5D), thereby drastically reducing the accuracy of the network computation (Figs. 4D, 7C, D). This modeling result is consistent with experimental data which have shown that activation of a small number of neurons in a cortical microcircuit is able to switch the behavioral response of the mouse (Houweling and Brecht 2008; Doron et al. 2014; Marshel et al. 2019; Dalgleish et al. 2020; Doron et al. 2020). This high sensitivity of network decisions to the activity of very small subsets of neurons is also of interest from a theoretical perspective: It suggests that the V1 model operates, like the brain, in a critical regime. Importantly, this can be reproduced in the V1 model in spite of the substantial level of noise and trial-to-trial variability that we have placed, based on experimental data from (Stringer et al. 2019), into the model, see (Chen et al. 2022) for details. In contrast, we found that the RANN does not operate in a critical regime: Very large subsets of its neurons have to be silenced in order to strongly reduce its task performance. Altogether, our results show that the organization of computations is in RANNs substantially different from the brain, and that data-based cortical microcircuit models provide a new family of models that can close this gap.

Reverse-engineering will be essential for understanding the organization of cortical computations. Doing that in the living brain is still handicapped by limitations of current experimental techniques. But one can spearhead such research by exploring and fine-tuning methods for reverse-engineering computations in data-based models that employ a similar architecture and neuron types as the brain. We have shown that low-dimensional projections provide substantial information about the organization of computations



(caption next page)

Figure 7 ***(previous page)*****: Inner workings of 4 sample neurons that are pivotal for the network decision.** **(A)** Spatial location and type of the 4 neurons labeled as Neuron I-IV in Fig. 6B and C whose firing rates have during $[50, 100]$ ms after image presentation the largest MI with the network decision that is made 100 ms later during the response window. **(B)** Analysis of the information contained in the after-spike currents of the four neurons. The after-spike current with the largest time constant is analyzed at the onset of the current image (see time point marked in the scheme above), in dependence on the identity of the PRECEDING test image, marked on the horizontal axis. Its mean value is shown with error bars denoting the SEM across 1000 trials. One sees that this value depends strongly on the identity of the preceding image, especially for neurons II - IV. **(C)** Silencing early-informer-neurons with after-spike currents that have large time constants ($\tau_{\text{asc}} > 300$ ms) in the descending order of their MI reduces testing performance much faster than silencing those with shorter ones ($\tau_{\text{asc}} < 300$ ms). The shaded areas represent the SEM across 10 models with different training datasets. **(D)** The membrane potential of the readout neuron is significantly changed under both the change and the no-change condition when 100 early-informer-neurons are silenced. We used the same stimuli but different realizations of the noise model and initial states. The dashed green lines represent the firing threshold. Blue curves represent the membrane potentials of single trials; the extended vertical bars denote the spikes in these trials. Red curves represent the average membrane potentials across 100 trials. The shaded red area represents the standard deviation across 100 trials.

in the data-based model of Billeh et al., see Fig. 5. These results suggest that among the numerous potentially relevant dynamical principles (Rabinovich et al. 2006) bifurcations of the network activity turn out to be highly relevant for analyzing computational progress in these computations. While the functional role of bifurcations of neural activity had previously focused on single neuron models, more recent experimental data provide evidence that bifurcations are also essential for understanding the functional role of populations of neurons in the living brain (Z. Wang et al. 2022). We have exhibited in Fig. 5D a further-going prediction: These bifurcations are highly sensitive to the firing activity of small sets of neurons: Silencing of just 100 neurons is able to flip a trajectory to a different bundle of trajectories that produces a different network decision. This causal relationship between the activity of individual neurons during a network computation and the resulting network decision needs to be tested in future neurobiological experiments.

The data-based model allows us also to take a closer look at the inner workings of these pivotal neurons, and to analyze how they can collect and transmit cues from two sequentially presented images which indicate whether the second image is a different one. We found that the values of internal variables with long time constants, which are abundantly present in the generalized leaky integrate-and-fire (GLIF_3) neuron models of Billeh et al. that had been fitted to data of specific neurons from the Allen Brain Atlas (Allen Institute 2018), assume values at the beginning of the processing of the current image that contain salient information about the identity of the preceding image (Fig. 6C and 7B). The specific causal impact of these neurons with long time constants on the network decision was verified through further lesion experiments in our model (Fig. 7D).

Altogether, our work demonstrates the feasibility of a new methodology for understanding the organization of computations in the neocortex: One can achieve a direct alignment of computational modeling and neurobiological experiments by analyzing computations in detailed large-scale models, whose spatial organization and neuron types match directly those found in the corresponding region of the neocortex, and which solve the same computational task on the same ensemble of stimuli as the subjects in the neurobiological experiments. This approach has now become feasible through recent advances in software design, such as TensorFlow, and computing hardware, such as graphical processing units (GPUs), that have been produced for the purpose of accelerating deep learning applications in AI.

An obvious next step in the direction of this work is an analysis of computations in detailed models of cortical microcircuits in other cortical areas such as motor cortex, and of distributed computations

in interconnected cortical microcircuits from different cortical areas. Furthermore, at least simplified models for the dendritic arborization of selected classes of neurons need to be added in order to make the impact of top-down inputs more realistic. But this needs to be done in a way that still supports fast simulations of large-scale models, so that training of these models for specific computational tasks remains computationally feasible. In addition, the computational role of projections to and synaptic inputs from subcortical areas such as basal ganglia and the thalamus (see (Cruz et al. 2023) for a recent review) needs to be modeled and analyzed through integrated large-scale models. Altogether, this work is likely to complement, challenge, and enhance experimental work that aims at clarifying the organization of brain computations. In addition, it will provide paradigms for a new generation of artificial neural network models that can capture both the astounding functional capabilities and the energy efficiency of sparsely active neural networks in the brain.

4 Methods

4.1 Neuron models

As in (Chen et al. 2022) we are focusing on the "core" part of the point-neuron version of the realistic V1 model introduced by (Billeh et al. 2020). To make it gradient-friendly, we replaced the hard reset of membrane potential after a spike emerges with the reduction of membrane potential $z_j(t)\,(v_{\text{th}} - E_L)$, where $z_j(t) = 1$ when neuron j fires at time t and $z_j(t) = 0$ otherwise. v_{th} is the firing threshold of membrane potential. E_L the resting membrane potential. This causes no significant change in the neural response (Chen et al. 2022). We simulated each trial for 600 ms. The dynamics of the modified GLIF_3 model was defined as

$$\begin{aligned} v_j(t+\delta t) &= \alpha v_j(t) + \frac{1-\alpha\tau}{C}\left(I_j^e(t+1) + \sum_m I_j^m(t+1) + gE_{\text{L}} + I_j^{\text{syn}}(t)\right) - z_j(t)\,(v_{\text{th}} - E_L) \\ z_j(t) &= H\left(v_j(t) - v_{\text{th}}\right) \\ I_j^e(t) &= \sum_i W_{ji}^{\text{in}} x_i(t) + qK_j^{\text{quick}}(t) + sK_j^{\text{slow}}, \end{aligned} \tag{1}$$

where C represents the neuron capacitance, I^e the external current, I^{syn} the synaptic current, g the membrane conductance, and v_{th} the spiking threshold. W_{ji}^{in} is the synaptic weight from LGN neuron i to V1 neuron j. The scales of the quick noise $K_j^{\text{quick}}(t)$ and the slow noise K_j^{slow} to neuron j are $q = 2$ and $s = 2$, respectively, unless otherwise stated. K_j was randomly drawn from the empirical noise distribution which will be elaborated on later. The decay factor α is given by $e^{-\delta t/\tau}$, where τ is the membrane time constant. δt denotes the discrete-time step size, which is set to 1 ms in our simulations. H denotes the Heaviside step function. To introduce a simple model of neuronal refractoriness, we further assumed that $z_j(t)$ is fixed to 0 after each spike of neuron j for a short refractory period depending on the neuron type. The after-spike current $I^m(t)$ was modeled as

$$I^m(t+\delta t) = f^m I^m(t) + z(t)\delta I^m;\ m = 1, \ldots, N_{\text{asc}}, \tag{2}$$

where the multiplicative constant $f^m = \exp\left(-k^m \delta t\right)$ and an additive constant, δI^m. In our study, $m = 1$ or 2. Neuron parameters have been fitted to experimental data from 111 selected neurons according to the cell database of the Allen Brain Atlas (Allen Institute 2018), see (Teeter et al. 2018; Billeh et al. 2020), including neuron capacity C, conductance g, resting potential E_{L}, the length of the refractory period, as well as amplitudes δI^m and decay time constants k^m of two types of after-spike currents, $m = 1, 2$.

4.2 Synaptic inputs

The V1 model utilizes experimental data to specify the connection probability between neurons. The base connection probability for any pair of neurons from the 17 cell classes is provided in (Billeh et al. 2020) in a table (shown in Fig. 1B), where white cells denote unknown values. The values in this table are derived from measured frequencies of synaptic connections for neurons at maximal 75 µm horizontal inter-somatic distance. The base connection probability was then scaled by an exponentially decaying factor based on the horizontal distance between the somata of the two neurons (Fig. 1C), also derived from experimental data. The synaptic delay was spread in $[1, 4]$ ms, as extracted from Fig. 4E of (Billeh et al. 2020) and rounded to the nearest integer as the integration step is 1 ms.

The postsynaptic current of neuron j was defined by the following dynamics (Billeh et al. 2020):

$$I_j^{\text{syn}}(t+\delta t) = e^{-\frac{\delta t}{\tau_{\text{syn}}}} I_j^{\text{syn}}(t) + \delta t e^{-\frac{\delta t}{\tau_{\text{syn}}}} C_j^{\text{rise}}(t) \tag{3}$$

$$C_j^{\text{rise}}(t+\delta t) = e^{-\frac{\delta t}{\tau_{\text{syn}}}} C_j^{\text{rise}}(t) + \sum_i W_{ji}^{\text{rec}} z_i(t) \frac{e}{\tau_{\text{syn}}}, \tag{4}$$

where τ_{syn} is the synaptic time constant, W_{ji}^{rec} is the recurrent input connection weight from neuron i to j, and z_i is the spike of presynaptic neuron i. The τ_{syn} constants depend on neuron types of pre- and postsynaptic neurons (Billeh et al. 2020).

4.3 Initial conditions

The default initial conditions for spikes and membrane potentials were set to zero, unless otherwise specified. The initial conditions for $\mathbf{W}^{\text{in}}$ and $\mathbf{W}^{\text{rec}}$ were taken from (Billeh et al. 2020), unless otherwise stated.

4.4 Data-driven noise model

We used a noise model that was introduced in our previous study (Chen et al. 2022). The model was based on an empirical noise distribution that was obtained from experimental data of mice responses to 2,800 nature images (Stringer et al. 2019). The noise currents $K_j^{\text{quick}}(t)$ and K_j^{slow} in Eq. 1 were drawn independently for all neurons from this distribution. The quick noise $K_j^{\text{quick}}(t)$ was drawn every 1 ms while the slow noise K_j^{slow} was drawn once every 600 ms. The empirical noise distribution was derived from the variability (additive noise) collected from the experimental data. A detailed mathematical analysis of this method is available in the methods and supplementary materials of (Stringer et al. 2019).

4.5 Readout neurons

We employed a readout population in the V1 model, whose firing activity during the response window encoded the network decisions for the visual-change-detection task. Each population consisted of a certain number (30 or 1) of randomly selected excitatory neurons in layer 5, located within a sphere of a radius of 55 µm (Fig. 2E).

4.6 Visual-change-detection task

LGN model. The visual stimuli were processed by a qualitative retina and LGN model, as depicted in Fig. 2C and following (Billeh et al. 2020). Their full LGN model consists of 17,400 spatiotemporal filters that simulate the responses of LGN neurons in mice to visual stimuli (Durand et al. 2016). Each filter generates a positive output, which represents the firing rates of a corresponding LGN neuron. We used only a subset of 2,589 of these LGN filters that provide inputs from a smaller part of the visual field to the core part of the V1 model, on which we are focusing in this study. The input images were first converted to grayscale and scaled to fit in the interval $[-Int, Int]$, where $Int > 0$. The output of the LGN model was then used as an external current input in the V1 model as follows:

$$I_{\text{sti}} = \boldsymbol{W}^{\text{in}} \cdot \text{LGN}(G_{Int}), \tag{5}$$

where G_{Int} represents images scaled into $[-Int, Int]$ for $Int = 2$.

Visual-change-detection task with natural images. We designed the visual-change-detection task to be as close as possible to corresponding biological experiments while keeping them as simple as possible. In the mouse experiments of (Garrett et al. 2020; Joshua H. Siegle et al. 2021), mice were trained to perform a visual change detection task using static natural images presented in a sequence of 250 ms with short phases (500 ms) of gray screens in between. The mice had to report whether the most recently presented image was the same as the previously presented one. To replicate this task while taking into account GPU memory limitations, we presented natural images for 100 ms each with delays between them lasting 200 ms (Fig. 2A, B). The first image was presented after 50 ms, and all images were selected from a set of 40 randomly chosen images from the ImageNet dataset (Deng et al. 2009). The model had to report within a 50 ms time window starting 150 ms after image onset (response window) if the image had changed.

In the response window, we defined the mean firing rate of readout population as

$$r_{\text{readout}} = \frac{1}{T_{\text{resp}} \cdot N_{\text{readout}}} \sum_{t=1}^{T_{\text{resp}}} \sum_{j=1}^{N_{\text{readout}}} z_j(t), \tag{6}$$

where the sum over j is over the $N_{\text{readout}} = 30$ readout neurons and the sum over t is over the time length of response window $T_{\text{resp}} = 50$ ms. If $r > r_0 = 0.01$, the model reported a network decision that the image had changed. Otherwise, it reported no-change.

4.7 Loss function

The loss function was defined as

$$L = L_{\text{cross-entropy}} + \lambda_f L_{\text{rate reg.}} + \lambda_v L_{\text{v reg.}}, \tag{7}$$

where $L_{\text{cross-entropy}}$ represents the cross-entropy loss, λ_f and λ_v represent the weights of firing-rate regularization $L_{\text{rate reg.}}$ and voltage regularization $L_{\text{v reg.}}$, respectively. As an example, the cross-entropy loss of visual change detection tasks was given by

$$L_{\text{cross-entropy}} = -\sum_m \left[T^{(m)} \log \sigma \left(\theta \left(r^{(m)}_{\text{readout}} - r_0 \right) \right) + \left(1 - T^{(m)} \right) \log \sigma \left(\theta \left(r_0 - r^{(m)}_{\text{readout}} \right) \right) \right], \tag{8}$$

where the sum over m is organized into chunks of 50 ms and $r^{(m)}_{\text{readout}}$ denotes the mean readout population firing rate defined in Eq. 6. Similarly, $T^{(m)}$ denotes the target output in time window m, being 1 if a

change in image identity should be reported and otherwise 0. The baseline firing rate r_0 was 0.01. σ represents the sigmoid function. θ is a trainable scale ($\theta > 0$) of firing rate.

We used regularization terms in the loss function to penalize very high firing rates as well as values of membrane voltages that were not biologically realistic. The default values of their weights were $\lambda_{\mathrm{f}} = 0.1$ and $\lambda_{\mathrm{v}} = 10^{-5}$. The rate regularization is defined via the Huber loss (Huber 1992) between the target firing rates, y, calculated from the model in (Billeh et al. 2020), and the firing rates, r, sampled the same number of neurons from the network model:

$$\begin{aligned} L_{\text{rate reg.}} &= \sum_{j}^{N} \left|\tau_j - \mathbb{I}\left\{\delta_j < 0\right\}\right| \frac{\mathcal{L}_\kappa\left(\delta_j\right)}{\kappa}, \quad \text{with} \\ \mathcal{L}_\kappa\left(\delta_j\right) &= \begin{cases} \frac{1}{2}\delta_j^2, & \text{if } |\delta_j| \leq \kappa \\ \kappa\left(|\delta_j| - \frac{1}{2}\kappa\right), & \text{otherwise} \end{cases} \end{aligned} \tag{9}$$

where j represents neuron j, N the number of neurons, $\tau_j = j/N$, $\delta = 0.002$, $\delta_j = r_j - r_j^{\text{target}}$. $\mathbb{I}(x) = 1$ when x is true; $\mathbb{I}(x) = 0$ when x is false.

The voltage regularization is defined through the term

$$L_{\text{v reg.}} = \frac{1}{N} \sum_{j=0}^{j=N} \left(\left[\frac{v_j - E_{\mathrm{L}}}{v_{\mathrm{th}} - E_{\mathrm{L}}} - 1 \right]^+ \right)^2 + \left(\left[-\frac{v_j - E_{\mathrm{L}}}{v_{\mathrm{th}} - E_{\mathrm{L}}} + 1 \right]^+ \right)^2, \tag{10}$$

where N represents the total number of neurons, v_j, the membrane potential of neuron j, $[\cdots]^+$, the rectifier function. v_{th} is the firing threshold of membrane potential. E_L the resting membrane potential.

4.8 Training and testing

We applied back-propagation through time (BPTT) (Chen et al. 2022) to minimize the loss function. The non-existing derivative $\frac{\partial z_j}{\partial v_j}$ was replaced in simulations by a simple nonlinear function of the membrane potential that is called the pseudo-derivative. Outside of the refractory period, we chose a pseudo-derivative of the form

$$\begin{aligned} \psi^t &= \frac{\gamma_{\mathrm{pd}}}{v_{\mathrm{th}} - E_{\mathrm{L}}} \exp\left(\frac{-\left(v_{sc}^t\right)^2}{\sigma_p^2} \right), \\ v_{sc}^t &= \frac{v^t - v_{\mathrm{th}}}{v_{\mathrm{th}} - E_{\mathrm{L}}}, \end{aligned} \tag{11}$$

where the dampening factor $\gamma_{\mathrm{pd}} = 0.5$, the Gaussian kernel width $\sigma_p = 0.28$. During the refractory period, the pseudo derivative was set to 0.

To demonstrate how sensitive the performance is to the scale of the surrogate derivative, I trained the model with $\gamma_{\mathrm{pd}} = 0.25$ and 0.75 and kept all other hyperparameters the same. When $\gamma_{\mathrm{pd}} = 0.25$, the testing accuracy is 0.7; when $\gamma_{\mathrm{pd}} = 0.75$, the testing accuracy is 0.75. Compared with the case of $\gamma_{\mathrm{pd}} = 0.5$ where the testing accuracy is 0.83, other values are worse. This demonstrates that the choice of the derivative's scale can substantially affect gradient-based learning performance in spiking neural networks (Zenke and Vogels 2021).

We drew a batch of visual stimuli (64) and calculated the gradient after every trial for each synaptic weight whether an increase or decrease of it (but without changing its sign) would reduce the loss function.

Weights were then updated by the average gradient across the batch. This method had originally only been applied to neuron networks with differentiable neuron models and was normally referred to as stochastic gradient descent.

During the training, we added the sign constraint on the weights of the neural network to keep Dale's law. Specifically, if an excitatory weight was updated to a negative value, it would be set to 0; vice versa. In every training run, we used a different random seed in order to draw fresh noise samples from the empirical distribution, and to randomly generate/select training samples.

4.9 Other simulation details

The BPTT training algorithm was implemented in TensorFlow, which is optimized to run efficiently on GPUs, allowing us to take advantage of their parallel computing capabilities. We distributed the visual-change-detection task trials over batches, with each batch containing 64 trials, and performed independent simulations in parallel. Each trial lasted for 600 ms of biological time, and computing gradients for each batch took around 5 s on an NVIDIA A100 GPU. Once all batches had finished (one step), gradients were calculated and averaged to update the weights by BPTT. We define an epoch as 500 iterations/steps. This computation had to be iterated for 22 epochs to make sure the performance was saturated. This took 12 h of wall clock time on 32 GPUs.

4.10 Recurrent artificial neural network models (RANNs)

Model

The dynamics of a RANN can be defined as

$$\tau \frac{d\boldsymbol{x}}{dt} = -\boldsymbol{x}(t) + \boldsymbol{W}_r \boldsymbol{r}(t) + \boldsymbol{W}_{ff}\mathbf{I}(t) + \boldsymbol{b}_r, \tag{12}$$

where the $\boldsymbol{x}$ is the activation of the network units and the corresponding firing rate is defined as

$$\boldsymbol{r} = \tanh(\boldsymbol{x}) \tag{13}$$

τ is the single-unit timescale, I used 50 ms as in (Sussillo et al. 2015; Pollock and Jazayeri 2020). $\boldsymbol{W}_r$ is the recurrent synaptic weights; $\boldsymbol{W}_{ff}$ is the feedforward synaptic weights; $\boldsymbol{b}_r$ is bias. The initialization of $\boldsymbol{W}_r$, $\boldsymbol{W}_{ff}$ and $\boldsymbol{b}_r$ are Gaussian noise $\mathcal{N}(0, \sigma^2)$. $\sigma = 1/\sqrt{\text{average \# fan-in}}$ for $\boldsymbol{W}_r$ and $\boldsymbol{W}_{ff}$; $\sigma = 0.1$ for $\boldsymbol{b}_r$. $\mathbf{I}(t)$ is the input to the network at time t. It is the output of LGN model (Billeh et al. 2020).

The linear readout $\boldsymbol{y}(t)$ from activities of all neurons $\boldsymbol{r}(t)$,

$$\boldsymbol{y}(t) = \boldsymbol{W}_y^\top \boldsymbol{r}(t) + \boldsymbol{b}_y, \tag{14}$$

where $\boldsymbol{b}_y$ is bias. $\boldsymbol{W}y$ is the readout weight. $\boldsymbol{W}y$ is a $N \times 2$ matrix; N is the number of recurrent neurons; 2 is the number of possible decisions.

The number of neurons and synapses in our model are the same as those in Billeh's model (Billeh et al. 2020). We randomly shuffled the connectivity or kept it the same as in Billeh's model.

Loss function

The loss function was defined as

$$L = L_{\text{cross-entropy}} + \lambda_{\text{reg.}} L_{\text{reg.}}, \tag{15}$$

where $L_{\text{cross-entropy}}$ represents the cross-entropy loss which was defined in Eq. 8, $\lambda_{\text{reg.}}$ represents the weight of activation regularization $L_{\text{reg.}}$. The activation regularization was defined as

$$L_{\text{reg.}} = \left(\left[\frac{1}{N} \sum_i |r_i| - \theta \right]^+ \right)^2, \tag{16}$$

where r_i is the "firing rate" of neuron i. N is the number of neurons and θ is a threshold ($\theta = 0.01$), unless otherwise stated. $[\cdots]^+$ is the rectifier linear unit function. The value of θ was determined as $\frac{1}{4}$ of the mean value of $|r|$ when training the RANN model without regularization.

Training details
We used the same training methods as in the V1 model but the learning rate is 10^{-4}. We also used $dt = 5$ ms to alleviate the vanishing gradient problem, as the one used in (Sussillo et al. 2015).

Details for calculating normalized averaged activity in Fig. 4A-C
As the "firing rate" of neurons in RANN, $\boldsymbol{r}$ is in $[-1, 1]$, we took the absolute value of $\boldsymbol{r}$ to compare with the neural activity in the V1 model.

4.11 Demixed principal component analysis

Demixed principal component analysis (Demixed PCA) is a statistical method that decomposes high-dimensional neural data into a set of orthogonal latent variables, each of which captures a unique aspect of the neural response Kobak et al. 2016. Briefly, let $\mathbf{X} n \times T$ be the neural data matrix, where n is the number of neurons and T is the number of time points. Let $\mathbf{S} n \times C \times T$ be the tensor of stimulus conditions, where C is the number of experimental conditions. The goal of Demixed PCA is to find a low-dimensional latent space $\mathbf{Y}_{d \times T}$, where d is the number of latent variables, that captures the majority of the variance in the neural data $\mathbf{X}$, while also separating out the variance that is specific to each experimental condition in $\mathbf{S}$.

In Fig. 3D, the principal component used for the projection arises by analysis of the first eigenvector of the covariance matrix that reflects variation through joint dependencies of the network decision and relative timing, see marginalization procedure of (Kobak et al. 2016). Hence, this matrix does not reflect variation that is caused only through the course of time within a trial or through the network decision alone. Moreover, to emphasize the formation of the network decision, we include in the computation of the aforementioned covariance matrix only data within [-50, 50] ms of the image presentation.

4.12 Mutual information

To estimate the mutual information between single neuron activity and the network decision, we binned the spike counts of each neuron into 10 uniformly distributed bins between the minimum and maximum spike count observed for that neuron within 50 ms windows. We then established an empirical joint distribution for the binned spike count and the network decision and computed the mutual information using the below formula.

$$MI(X;Y) = \sum_{y \in Y} \sum_{x \in X} P(x,y) \log_2 \frac{P(x,y)}{P(x)P(y)}, \tag{17}$$

where X is the set of firing activities of each neuron within a 50 ms window, and Y is the set of network decisions (either change or no-change). $P(x, y)$ is the joint probability distribution of spike count and

network decision, while $P(x)$ and $P(y)$ are the marginal probability distributions of spike count and network decision, respectively.

To estimate $P(x, y)$, we calculated the spike count of each neuron within a 50 ms window and established an empirical joint distribution by counting the number of occurrences of each possible combination of spike counts and network decisions across 100,000 trials. We then normalized the joint distribution to obtain a probability distribution. We estimated $P(x)$ and $P(y)$ in a similar way by counting the number of occurrences of each possible value of spike count and network decision, respectively, across all trials.

4.13 UMAP

We applied an exponential filter with a time constant of 20 ms to the spike output of each neuron for 8 new images that had not been used during training. We then discarded all but the 1,500 most important principal components of these network states (explain 38% of variance; compromise to the memory consumption), and embedded these into 2D space by UMAP. The number of neighbors is 200. These projected network states were recorded for every ms, represented by a dot in Fig. 5.

UMAP (Uniform Manifold Approximation and Projection) is a nonlinear dimensionality reduction algorithm that aims to preserve the local structure of high-dimensional data in low-dimensional space (McInnes et al. 2018). In this study, UMAP was used to embed the network activity during task performance into 2D space.

First, we applied an exponential filter with a time constant of 20 ms to the spike output of each neuron for 8 new images that had not been used during training. We then computed the principal components of the resulting network states, discarding all but the 1,500 most important components that explained 38% of the variance.

Next, we used UMAP to embed the high-dimensional network states into 2D space, while preserving the local structure of the data. Specifically, we used the UMAP implementation from the Python library umap, with the following parameters: `n_neighbors` = 200, `min_dist` = 0.1, and metric = euclidean.

The resulting 2D embeddings represent the low-dimensional trajectories of the network states during image processing, and were recorded for every ms. Each point in the 2D space represents a network state at a given time point, and is displayed as a dot in Fig. 5. The trajectory of the network states can be visualized by connecting these dots in chronological order.

4.14 Additional figure description

Normalized activity in Fig. 3, 4, and S6
Spiking activity at a specific relative time step, regarding image presentation, was averaged over 200 trials. These average activities per time step were then normalized with the maximum values of their average activation.

Fig. 6B and C
7 neurons shown in Fig. 6B and C were selected from the 20 early-informer-neurons with the largest MI that represented each of the 4 neuron classes, taking within each neuron class (excitatory, PV, Htra3, or Sst neurons) the ones with the largest MI.

Acknowledgements

We would like to thank Yuqing Zhu for helpful discussions. We also thank Sandra Diaz for advice and help in using supercomputers. This research was partially supported by the Human Brain Project (Grant Agreement number 785907) of the European Union and a grant from Intel. Computations were carried out on the Human Brain Project PCP Pilot Systems at the Jülich Supercomputing Centre, which received co-funding from the European Union (Grant Agreement number 604102).

References

Abeles, Moshe, Gaby Hayon, and Daniel Lehmann (2004). "Modeling compositionality by dynamic binding of synfire chains". In: *Journal of computational neuroscience* 17, pp. 179–201.

Allen, William E, Isaac V Kauvar, Michael Z Chen, Ethan B Richman, Samuel J Yang, Ken Chan, Viviana Gradinaru, Benjamin E Deverman, Liqun Luo, and Karl Deisseroth (2017). "Global representations of goal-directed behavior in distinct cell types of mouse neocortex". In: *Neuron* 94.4, pp. 891–907.

Allen Institute (2018). "© 2018 Allen Institute for Brain Science. Allen Cell Types Database, cell feature search. Available from: celltypes.brain-map.org/data". In:

Bellec, Guillaume, Darjan Salaj, Anand Subramoney, Robert Legenstein, and Wolfgang Maass (2018). "Long short-term memory and Learning-to-learn in networks of spiking neurons". In: *Advances in Neural Information Processing Systems*. Ed. by S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett. Vol. 31. Curran Associates, Inc., pp. 787–797. URL: https://proceedings.neurips.cc/paper/2018/file/c203d8a151612acf12457e4d67635a95-Paper.pdf.

Billeh, Yazan N, Binghuang Cai, Sergey L Gratiy, Kael Dai, Ramakrishnan Iyer, Nathan W Gouwens, Reza Abbasi-Asl, Xiaoxuan Jia, Joshua H Siegle, Shawn R Olsen, et al. (2020). "Systematic integration of structural and functional data into multi-scale models of mouse primary visual cortex". In: *Neuron*.

Broome, Bede M, Vivek Jayaraman, and Gilles Laurent (2006). "Encoding and decoding of overlapping odor sequences". In: *Neuron* 51.4, pp. 467–482.

Chen, Guozhang, Franz Scherr, and Wolfgang Maass (2022). "A data-based large-scale model for primary visual cortex enables brain-like robust and versatile visual processing". In: *Science Advances* 8.44, eabq7592.

Cruz, K Guadalupe, Yi Ning Leow, Nhat Minh Le, Elie Adam, Rafiq Huda, and Mriganka Sur (2023). "Cortical-subcortical interactions in goal-directed behavior". In: *Physiological reviews* 103.1, pp. 347–389.

Dalgleish, Henry WP, Lloyd E Russell, Adam M Packer, Arnd Roth, Oliver M Gauld, Francesca Greenstreet, Emmett J Thompson, and Michael Häusser (2020). "How many neurons are sufficient for perception of cortical activity?" In: *Elife* 9, e58889.

Deng, Jia, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei (2009). "Imagenet: A large-scale hierarchical image database". In: *2009 IEEE conference on computer vision and pattern recognition*. Ieee, pp. 248–255.

Doron, Guy, Jiyun N Shin, Naoya Takahashi, Moritz Drüke, Christina Bocklisch, Salina Skenderi, Lisa de Mont, Maria Toumazou, Julia Ledderose, Michael Brecht, et al. (2020). "Perirhinal input to neocortical layer 1 controls learning". In: *Science* 370.6523, eaaz3136.

Doron, Guy, Moritz Von Heimendahl, Peter Schlattmann, Arthur R Houweling, and Michael Brecht (2014). "Spiking irregularity and frequency modulate the behavioral report of single-neuron stimulation". In: *Neuron* 81.3, pp. 653–663.

Douglas, Rodney J and Kevan AC Martin (2004). "Neuronal circuits of the neocortex". In: *Annu. Rev. Neurosci.* 27, pp. 419–451.

Driscoll, Laura N, Noah L Pettit, Matthias Minderer, Selmaan N Chettih, and Christopher D Harvey (2017). "Dynamic reorganization of neuronal activity patterns in parietal cortex". In: *Cell* 170.5, pp. 986–999.

Durand, Séverine, Ramakrishnan Iyer, Kenji Mizuseki, Saskia de Vries, Stefan Mihalas, and R Clay Reid (2016). "A comparison of visual response properties in the lateral geniculate nucleus and primary visual cortex of awake and anesthetized mice". In: *Journal of Neuroscience* 36.48, pp. 12144–12156.

Garrett, Marina, Sahar Manavi, Kate Roll, Douglas R Ollerenshaw, Peter A Groblewski, Nicholas D Ponvert, Justin T Kiggins, Linzy Casal, Kyla Mace, Ali Williford, Arielle Leon, Xiaoxuan Jia, Peter Ledochowitsch, Michael A Buice, Wayne Wakeman, Stefan Mihalas, and Shawn R Olsen (Feb. 2020). "Experience shapes activity dynamics and stimulus coding of VIP inhibitory cells". In: *eLife* 9. Ed. by Brice Bathellier, Joshua I Gold, Brice Bathellier, and Georg B Keller, e50340. ISSN: 2050-084X. DOI: 10.7554/eLife.50340. URL: https://doi.org/10.7554/eLife.50340.

Harris, Kenneth D. and Gordon M. G. Shepherd (Feb. 2015). "The neocortical circuit: themes and variations". In: *Nature Neuroscience* 18.2, pp. 170–181. ISSN: 1546-1726. DOI: 10.1038/nn.3917. URL: https://doi.org/10.1038/nn.3917.

Houweling, Arthur R and Michael Brecht (2008). "Behavioural report of single neuron stimulation in somatosensory cortex". In: *Nature* 451.7174, pp. 65–68.

Huber, Peter J (1992). "Robust estimation of a location parameter". In: *Breakthroughs in statistics*. Springer, pp. 492–518.

Kappel, David, Bernhard Nessler, and Wolfgang Maass (2014). "STDP installs in winner-take-all circuits an online approximation to hidden Markov model learning". In: *PLoS computational biology* 10.3, e1003511.

Kato, Saul, Harris S Kaplan, Tina Schrödel, Susanne Skora, Theodore H Lindsay, Eviatar Yemini, Shawn Lockery, and Manuel Zimmer (2015). "Global brain dynamics embed the motor command sequence of Caenorhabditis elegans". In: *Cell* 163.3, pp. 656–669.

Koay, Sue Ann, Adam S Charles, Stephan Y Thiberge, Carlos D Brody, and David W Tank (2022). "Sequential and efficient neural-population coding of complex task information". In: *Neuron* 110.2, pp. 328–349.

Kobak, Dmitry, Wieland Brendel, Christos Constantinidis, Claudia E Feierstein, Adam Kepecs, Zachary F Mainen, Xue-Lian Qi, Ranulfo Romo, Naoshige Uchida, and Christian K Machens (Apr. 2016). "Demixed principal component analysis of neural population data". In: *eLife* 5. Ed. by Mark CW van Rossum, e10989. ISSN: 2050-084X. DOI: 10.7554/eLife.10989. URL: https://doi.org/10.7554/eLife.10989.

Maass, Wolfgang and Henry Markram (2004). "On the computational power of circuits of spiking neurons". In: *Journal of computer and system sciences* 69.4, pp. 593–616.

Maass, Wolfgang, Thomas Natschläger, and Henry Markram (2002). "Real-time computing without stable states: A new framework for neural computation based on perturbations". In: *Neural computation* 14.11, pp. 2531–2560.

Markram, Henry, Eilif Muller, Srikanth Ramaswamy, Michael W Reimann, Marwan Abdellah, Carlos Aguado Sanchez, Anastasia Ailamaki, Lidia Alonso-Nanclares, Nicolas Antille, Selim Arsever, et al. (2015). "Reconstruction and simulation of neocortical microcircuitry". In: *Cell* 163.2, pp. 456–492.

Marshel, James H, Yoon Seok Kim, Timothy A Machado, Sean Quirin, Brandon Benson, Jonathan Kadmon, Cephra Raja, Adelaida Chibukhchyan, Charu Ramakrishnan, Masatoshi Inoue, et al. (2019). "Cortical layer–specific critical dynamics triggering perception". In: *Science* 365.6453.

Martin Abadi, Ashish Agarwal, Paul Barham, Eugene Brevdo, Zhifeng Chen, Craig Citro, Greg S. Corrado, Andy Davis, Jeffrey Dean, Matthieu Devin, Sanjay Ghemawat, Ian Goodfellow, Andrew Harp, Geoffrey Irving, Michael Isard, Yangqing Jia, Rafal Jozefowicz, Lukasz Kaiser, Manjunath Kudlur, Josh Levenberg, Dandelion Mané, Rajat Monga, Sherry Moore, Derek Murray, Chris Olah, Mike Schuster, Jonathon Shlens, Benoit Steiner, Ilya Sutskever, Kunal Talwar, Paul Tucker, Vincent Vanhoucke, Vijay Vasudevan, Fernanda Viégas, Oriol Vinyals, Pete Warden, Martin Wattenberg, Martin

Wicke, Yuan Yu, and Xiaoqiang Zheng (2015). *TensorFlow: Large-Scale Machine Learning on Heterogeneous Systems*. Software available from tensorflow.org. URL: https://www.tensorflow.org/.

McInnes, L., J. Healy, and J. Melville (Feb. 2018). "UMAP: Uniform Manifold Approximation and Projection for Dimension Reduction". In: *ArXiv e-prints*. arXiv: 1802.03426 [stat.ML].

Mountcastle, Vernon B (1998). *Perceptual neuroscience: The cerebral cortex*. Harvard University Press.

Pollock, Eli and Mehrdad Jazayeri (2020). "Engineering recurrent neural networks from task-relevant manifolds and dynamics". In: *PLoS computational biology* 16.8, e1008128.

Rabinovich, Mikhail I, Pablo Varona, Allen I Selverston, and Henry DI Abarbanel (2006). "Dynamical principles in neuroscience". In: *Reviews of modern physics* 78.4, p. 1213.

Russakovsky, Olga, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, Alexander C. Berg, and Li Fei-Fei (2015). "ImageNet Large Scale Visual Recognition Challenge". In: *International Journal of Computer Vision (IJCV)* 115.3, pp. 211–252. DOI: 10.1007/s11263-015-0816-y.

Siegle, Joshua H., Xiaoxuan Jia, Séverine Durand, Sam Gale, Corbett Bennett, Nile Graddis, Greggory Heller, Tamina K. Ramirez, Hannah Choi, Jennifer A. Luviano, Peter A. Groblewski, Ruweida Ahmed, Anton Arkhipov, Amy Bernard, Yazan N. Billeh, Dillan Brown, Michael A. Buice, Nicolas Cain, Shiella Caldejon, Linzy Casal, Andrew Cho, Maggie Chvilicek, Timothy C. Cox, Kael Dai, Daniel J. Denman, Saskia E. J. de Vries, Roald Dietzman, Luke Esposito, Colin Farrell, David Feng, John Galbraith, Marina Garrett, Emily C. Gelfand, Nicole Hancock, Julie A. Harris, Robert Howard, Brian Hu, Ross Hytnen, Ramakrishnan Iyer, Erika Jessett, Katelyn Johnson, India Kato, Justin Kiggins, Sophie Lambert, Jerome Lecoq, Peter Ledochowitsch, Jung Hoon Lee, Arielle Leon, Yang Li, Elizabeth Liang, Fuhui Long, Kyla Mace, Jose Melchior, Daniel Millman, Tyler Mollenkopf, Chelsea Nayan, Lydia Ng, Kiet Ngo, Thuyahn Nguyen, Philip R. Nicovich, Kat North, Gabriel Koch Ocker, Doug Ollerenshaw, Michael Oliver, Marius Pachitariu, Jed Perkins, Melissa Reding, David Reid, Miranda Robertson, Kara Ronellenfitch, Sam Seid, Cliff Slaughterbeck, Michelle Stoecklin, David Sullivan, Ben Sutton, Jackie Swapp, Carol Thompson, Kristen Turner, Wayne Wakeman, Jennifer D. Whitesell, Derric Williams, Ali Williford, Rob Young, Hongkui Zeng, Sarah Naylor, John W. Phillips, R. Clay Reid, Stefan Mihalas, Shawn R. Olsen, and Christof Koch (Jan. 2021). "Survey of spiking in the mouse visual system reveals functional hierarchy". In: *Nature*. ISSN: 1476-4687. DOI: 10.1038/s41586-020-03171-x. URL: https://doi.org/10.1038/s41586-020-03171-x.

Steinmetz, Nicholas A, Peter Zatka-Haas, Matteo Carandini, and Kenneth D Harris (2019). "Distributed coding of choice, action and engagement across the mouse brain". In: *Nature* 576.7786, pp. 266–273.

Stringer, Carsen, Marius Pachitariu, Nicholas Steinmetz, Matteo Carandini, and Kenneth D Harris (2019). "High-dimensional geometry of population responses in visual cortex". In: *Nature* 571, pp. 361–365.

Sussillo, David and Omri Barak (Mar. 2013). "Opening the Black Box: Low-Dimensional Dynamics in High-Dimensional Recurrent Neural Networks". In: *Neural Computation* 25.3, pp. 626–649. ISSN: 0899-7667. DOI: 10.1162/NECO_a_00409. eprint: https://direct.mit.edu/neco/article-pdf/25/3/626/881886/neco_a_00409.pdf. URL: https://doi.org/10.1162/NECO%5C_a%5C_00409.

Sussillo, David, Mark M Churchland, Matthew T Kaufman, and Krishna V Shenoy (2015). "A neural network that finds a naturalistic solution for the production of muscle activity". In: *Nature neuroscience* 18.7, pp. 1025–1033.

Teeter, Corinne, Ramakrishnan Iyer, Vilas Menon, Nathan Gouwens, David Feng, Jim Berg, Aaron Szafer, Nicholas Cain, Hongkui Zeng, Michael Hawrylycz, et al. (2018). "Generalized leaky integrate-and-fire models classify multiple neuron types". In: *Nature communications* 9.1, pp. 1–15.

Thomson, Alex M and Christophe Lamy (2007). "Functional maps of neocortical local circuitry". In: *Frontiers in neuroscience* 1, p. 2.

Wang, Zhaoxiang, Zhouyan Feng, Yue Yuan, Gangsheng Yang, Yifan Hu, and Lvpiao Zheng (2022). "Bifurcations in the firing of neuronal population caused by a small difference in pulse parameters during sustained stimulations in rat hippocampus in vivo". In: *IEEE Transactions on Biomedical Engineering* 69.9, pp. 2893–2904.

Yang, Guangyu Robert, Madhura R Joglekar, H Francis Song, William T Newsome, and Xiao-Jing Wang (2019). "Task representations in neural networks trained to perform many cognitive tasks". In: *Nature neuroscience* 22.2, pp. 297–306.

Yang, Guangyu Robert and Xiao-Jing Wang (2020). "Artificial neural networks for neuroscientists: A primer". In: *Neuron* 107.6, pp. 1048–1070.

Yuksel, Seniha Esen, Joseph N Wilson, and Paul D Gader (2012). "Twenty years of mixture of experts". In: *IEEE transactions on neural networks and learning systems* 23.8, pp. 1177–1193.

Zenke, Friedemann and Tim P Vogels (2021). "The remarkable robustness of surrogate gradient learning for instilling complex function in spiking neural networks". In: *Neural Computation* 33.4, pp. 899–925.

Supplementary Information

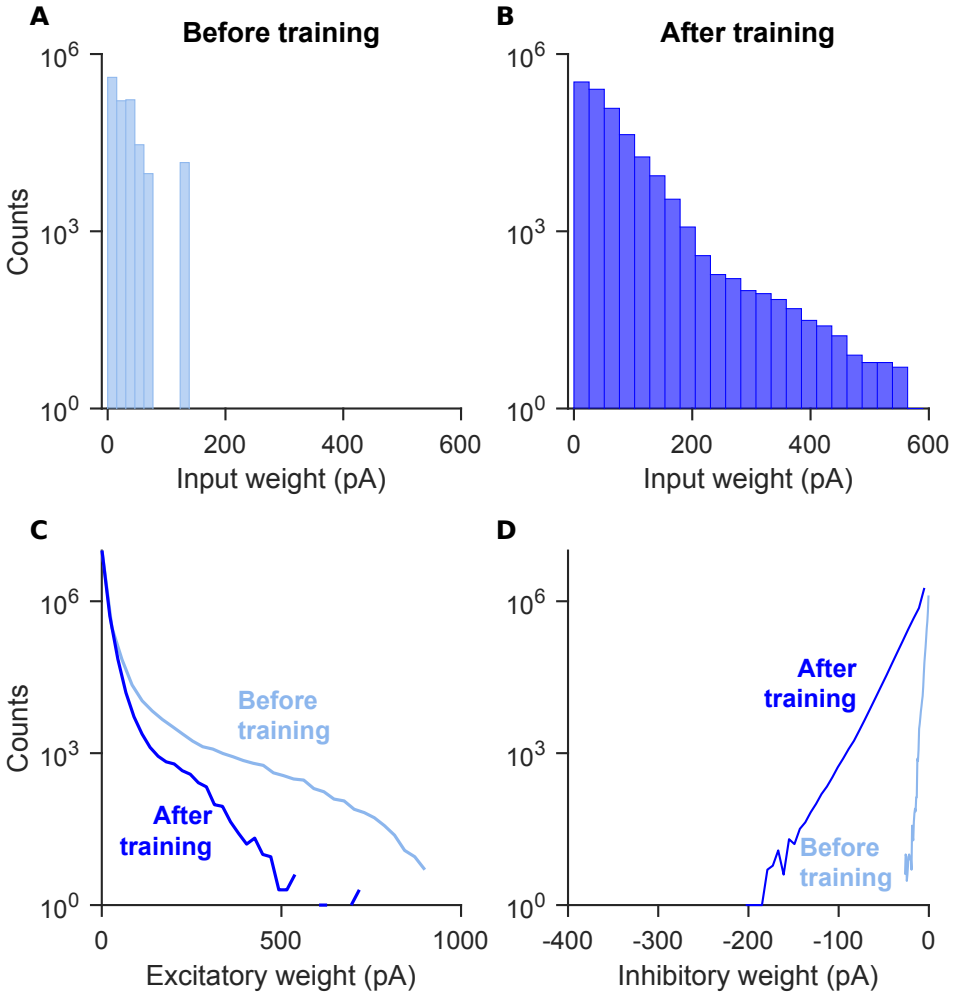


Figure S1: Changes in the distribution of synaptic weights through training. (A) Distributions of input weights before training. **(B)** Same as **(A)**, but after training. The mean of input weights increases from 24.1 to 25.2 through training. **(C)** Distributions of excitatory weights in V1 model before and after training. **(D)** Distributions of inhibitory weights in V1 model before and after training. Note that the weights before training were given by (Billeh et al. 2020).

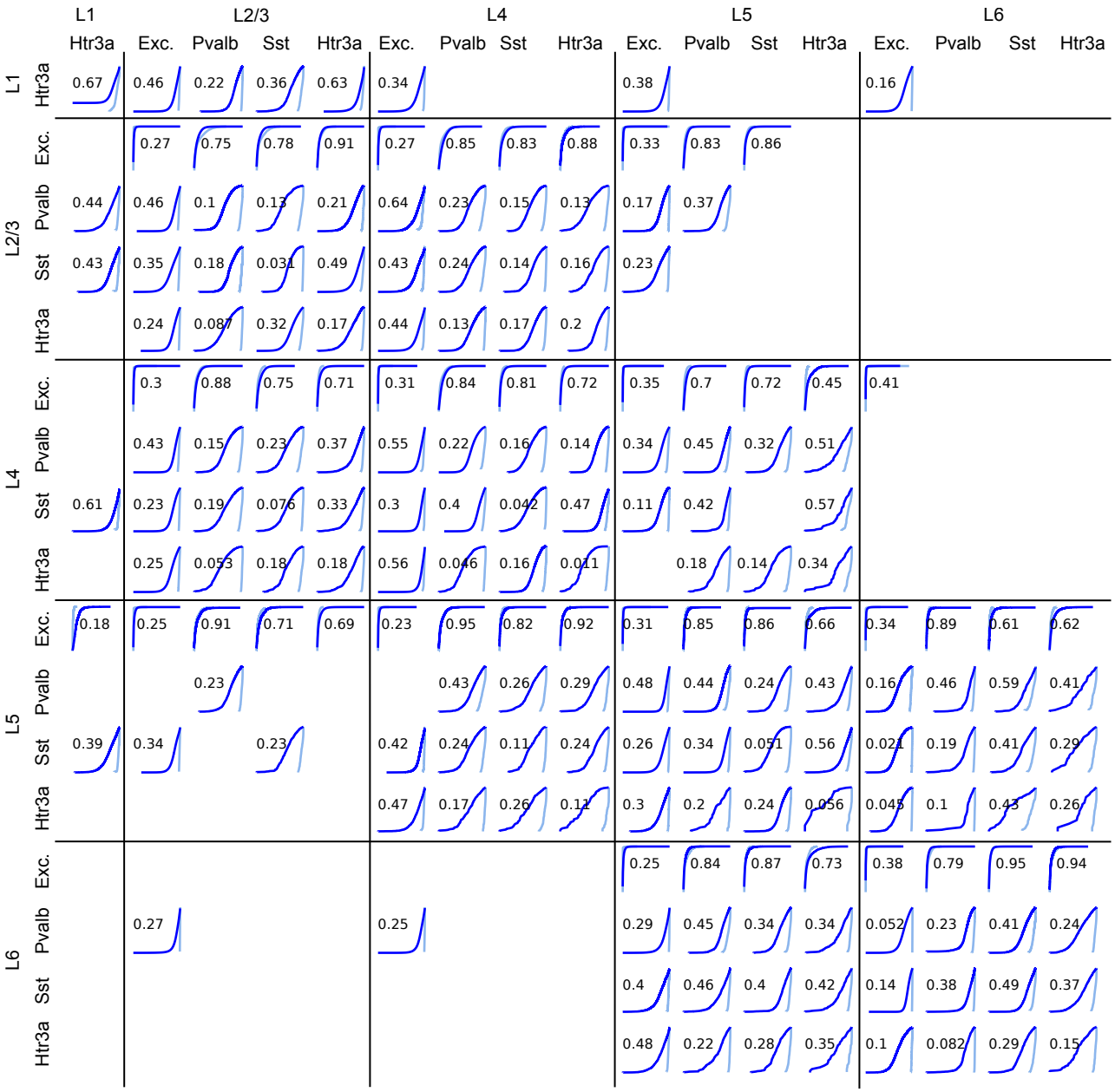
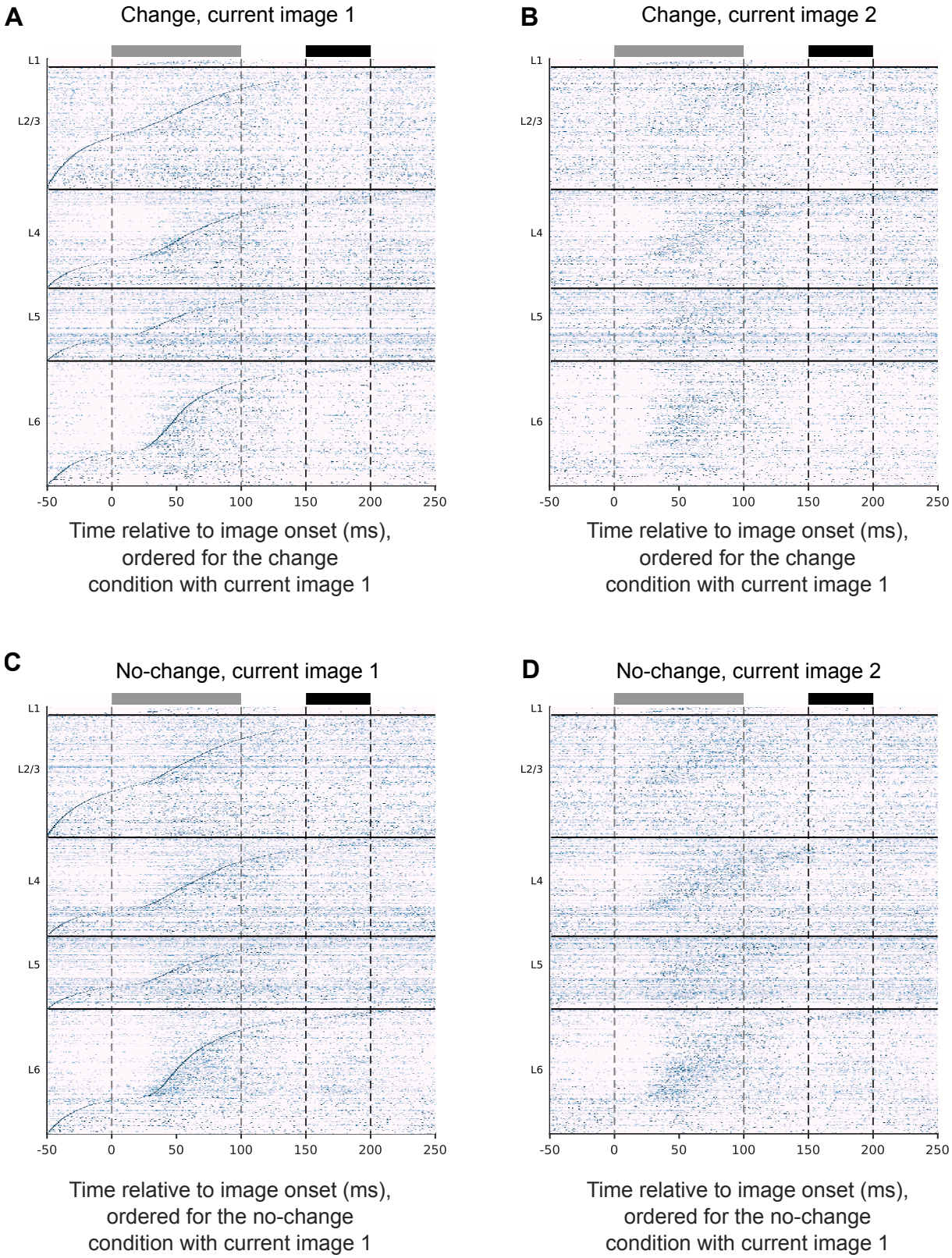


Figure S2: Distribution of recurrent synaptic weights between each pair of populations before (light blue) and after learning (dark blue). Each row represents a pre-synaptic neuron population, and each column represents a post-synaptic neuron population. The histogram represents the distribution of synaptic weights of all synaptic connections that share the same pre-synaptic and post-synaptic neuron population. Vertical axis in each panel is log-scale. Horizontal axis is linear scale and horizontal range is from the smallest value to the largest value of each population. The number is $1 - D$ where D is from the Kolmogorov–Smirnov test, quantifying the similarity between distributions (Billeh et al. 2020). Exc., excitatory neurons.



(caption next page)

Figure S3 ***(previous page)*****: The temporal order of peak activity encodes in the V1 model also information about the identity of the current image.** We have shown in Fig. 3 that the temporal order of peak activity in the V1 model has characteristic differences for different trial types (change or no-change). We show here that this temporal or rank-order coding is even more refined: the order contains in addition information about the identity of the current image. **(A)** Normalized average responses over 200 trials with the change condition and the same current image (image 1) but different preceding images, with neurons ordered according to the time of their peak activity. The gray and black bars at the top denote the image presentation and response windows, respectively. **(B)** Same as in **(A)**, but all trials have the same current image 2 and neurons were ordered as in **(A)**. The resulting blurred sequence indicates that the order of peak activity of neurons is different for images 1 and 2, also within the same trial type (change condition). **(C)** and **(D)** Same as in **(A)** and **(B)**, respectively, but for the no-change condition.

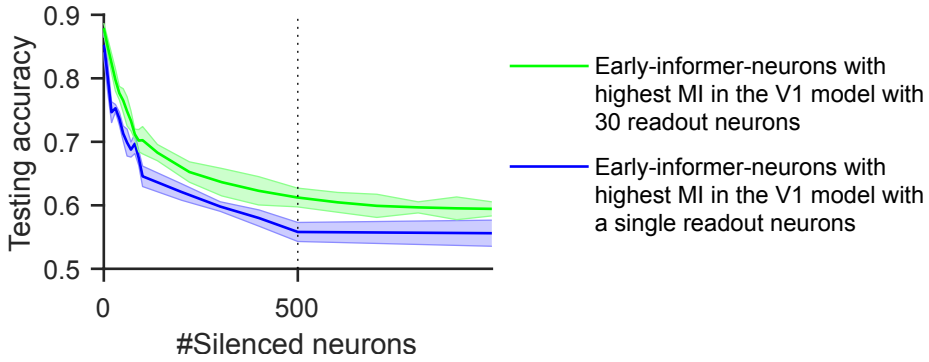
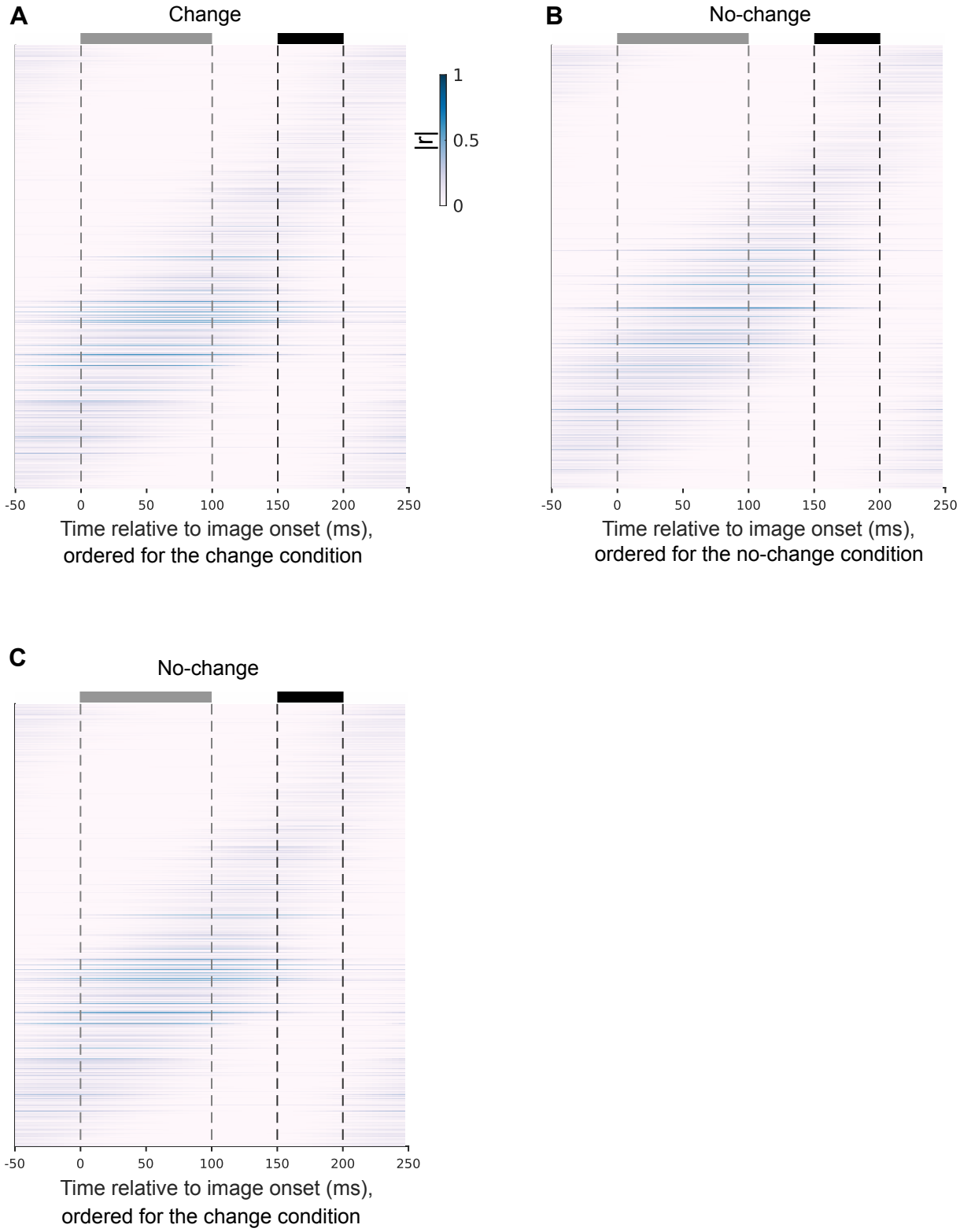
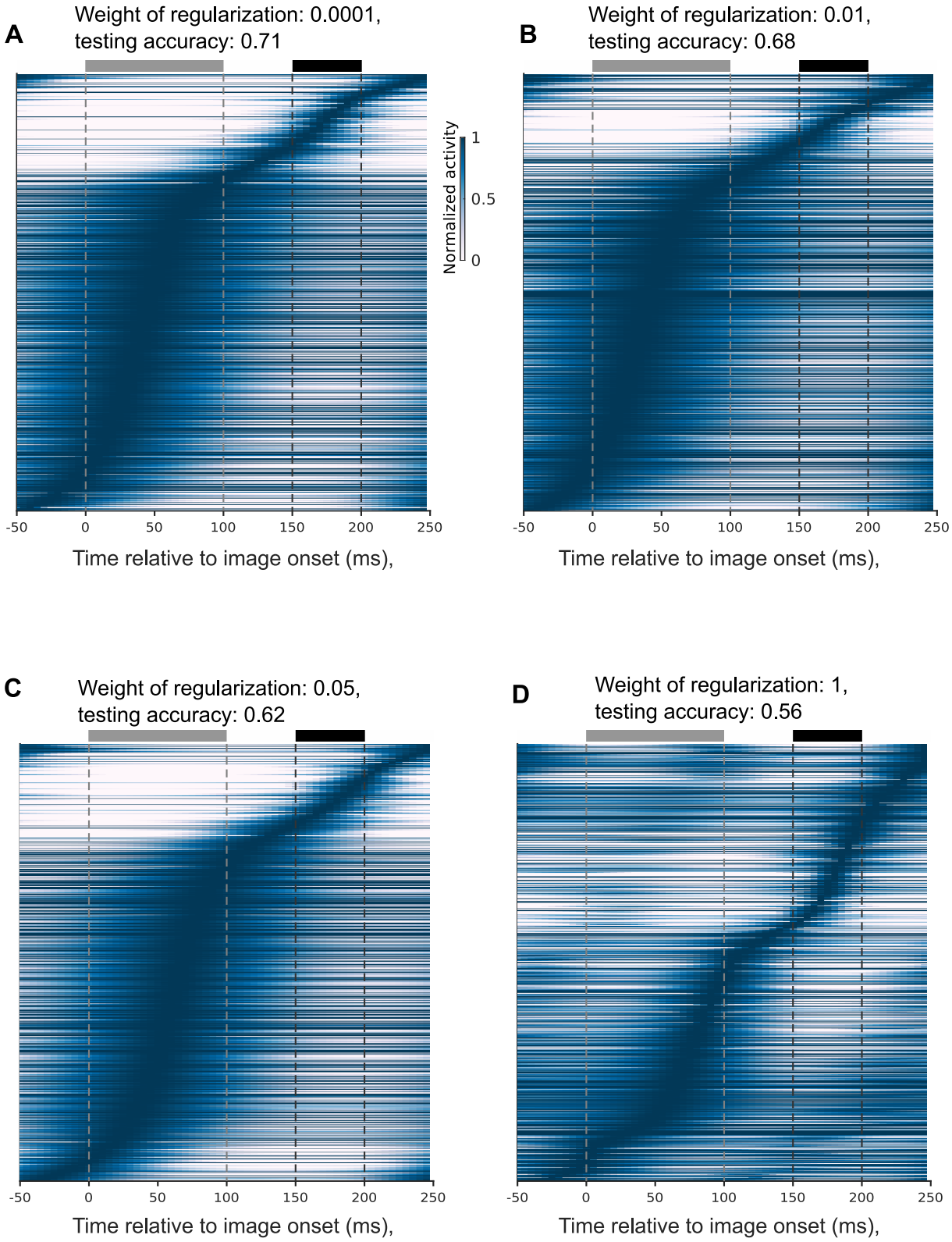


Figure S4: Lesion experiments for two versions of the trained V1 model with 30 and 1 readout neurons that are trained to report an image chance. Task performance quickly decreases when early-informer-neurons are silenced (in the order of their MI with the network decision) in both versions. The 30-readout-neuron model has also after silencing a given number of neurons slightly higher accuracy on test images than the single-readout-neuron model, consistent with Fig. 2F. Both curves show average values for 10 V1 models where different sets of training data were used. The shaded area represents the SEM across 10 models.



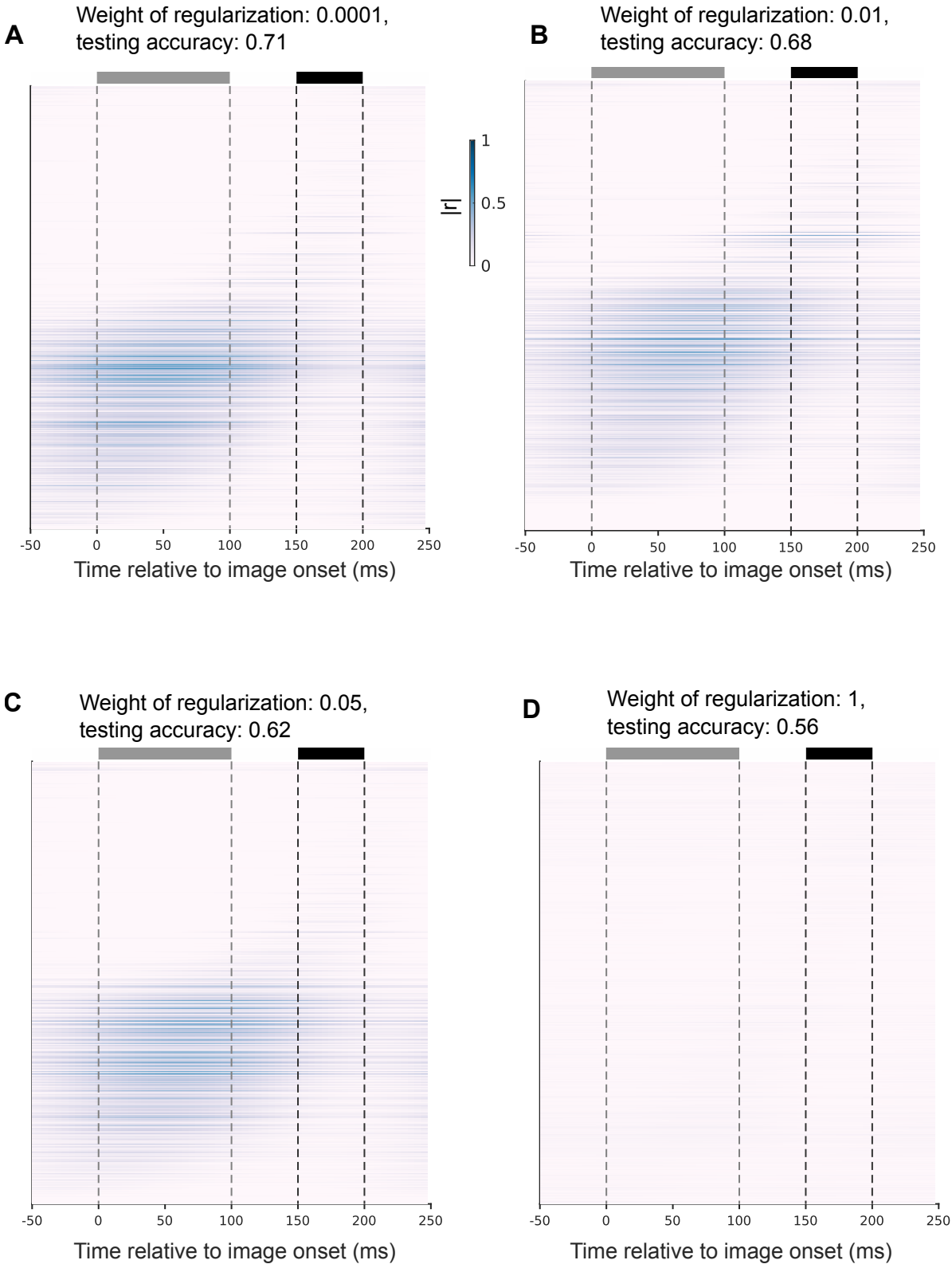
(caption next page)

Figure S5 ***(previous page)*****: Another attempt to identify trial-type dependent sequential activity in the RANN, similarly as shown in Fig. 3 for the V1 model (A-C)** Same RANN and conditions as in Fig. 4A-C but visualized without neuron-wise normalization. The fuzzy sequential order does not get sharper in panel **(B)** than in panel **(C)**, although the neurons are ordered in **(B)** for this particular trial type (no-change).



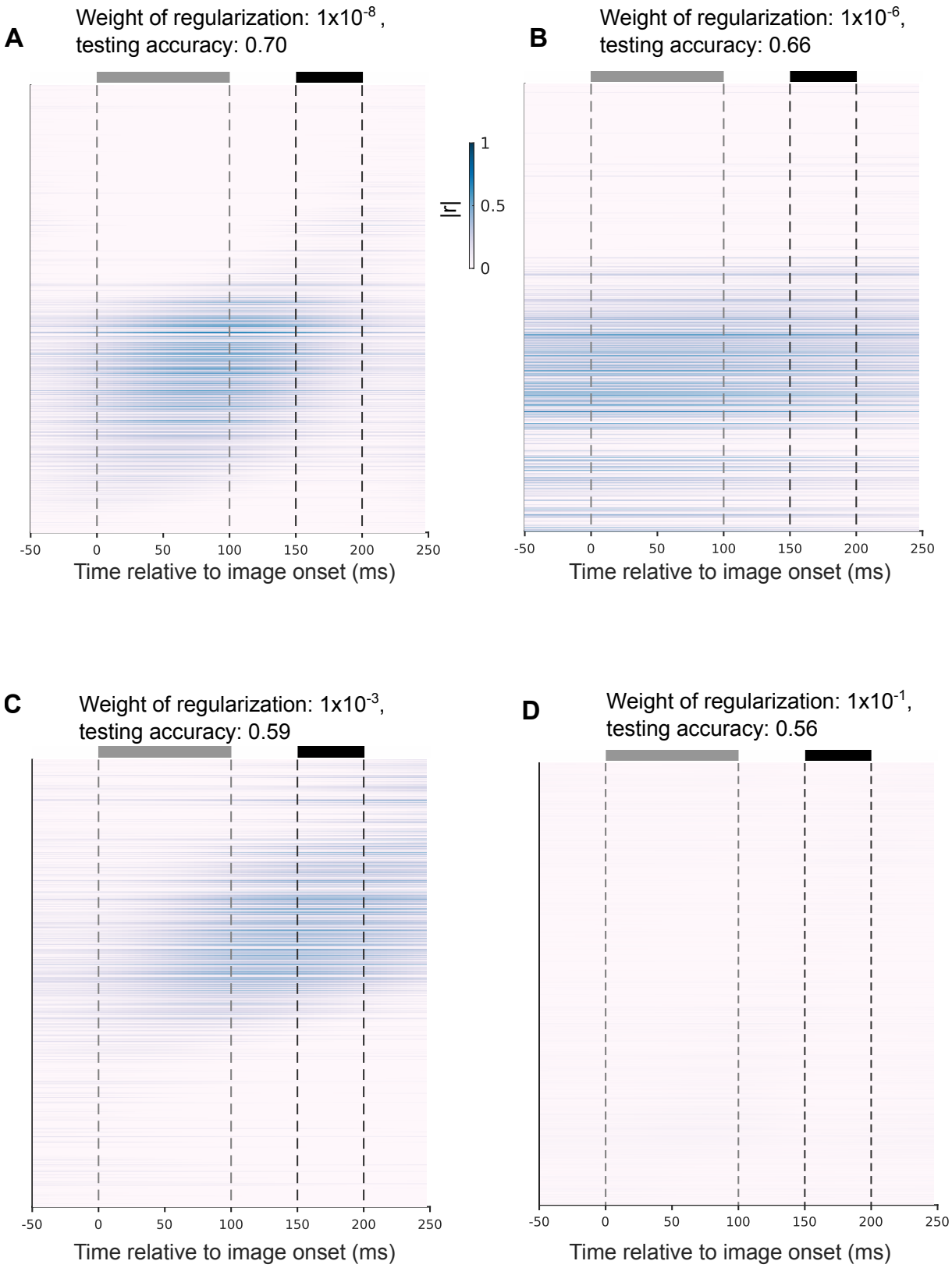
(caption next page)

Figure S6 ***(previous page)*****: Different weights of the regularization term cannot produce sparse sequential neural activity while maintaining high task performance. (A-D)** The normalized average responses of 200 change-condition trials in the RANN with different weights of regularization are plotted over time, with neurons ordered based on the time of their peak activities under the change condition. One sees that training the RANN for the same task with different weights of the regularization term (Methods) does not produce a sparse sequential neural activity as in the experimental data and the V1 model. Furthermore, more aggressive regularization strongly reduces task performance.



(caption next page)

Figure S7 ***(previous page)*****: Plotting the same results as in Fig. S6 without neuron-wise normalization still does not indicate a RANN regime with sparse sequential neural activity. (A-D)** Here the average absolute responses $|r|$ of 200 change-condition trials in RANN with different weights of regularization are plotted over time, with neurons ordered based on the time of their peak activities under the change condition.



(caption next page)

Figure S8 ***(previous page)*****: Setting the value of the threshold in the activation regularization term to 0 also does not produce sparse sequential activity in the RANN.** The threshold, θ in Eq. 16 was changed here from its default value 0.01 to 0. Average responses of 200 change-condition trials in RANN with different weights of the regularization term are plotted over time, with neurons ordered based on the time of their peak activities under the change condition. Similarly as in Fig. S7, none of the weights of the regularization term that we tried produces sparse sequential activity in the RANN while maintaining high task performance.

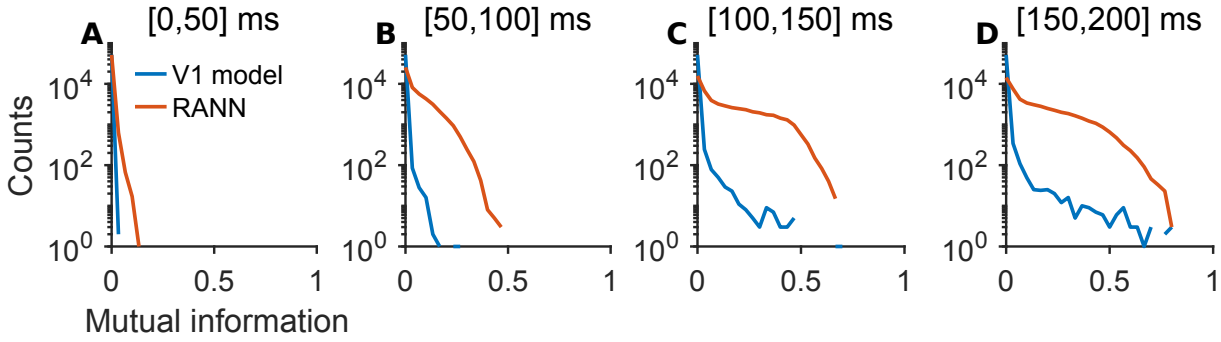


Figure S9: Comparison of mutual-information distributions among neurons in the V1 Model and the RANN. (A-D) The number of neurons that has a given level of MI with the network output is shown for each 50-ms window in the V1 model and the RANN (image onset was at time 0). One clearly sees that for each of these time windows that are by 1 or 2 orders of magnitude more neurons that have high MI with the network output. This explains also why so much more neurons need to be silenced in the RANN in order to reduce the task performance of the network to a given level.

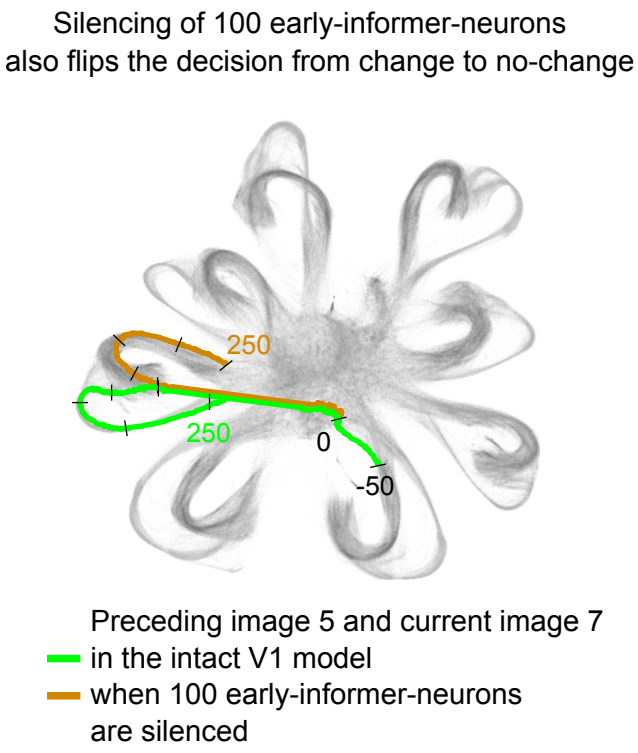


Figure S10: Silencing of 100 neurons can also flip a trajectory from the bundle for change to the no-change bundle of trajectories. In Fig. 5D, we demonstrated that silencing 100 early-informer-neurons can cause the trajectory of network states to flip from the bundle for no-change to the bundle for change trials. We show here that silencing of the same 100 neurons can also flip the network bifurcation in the other direction.

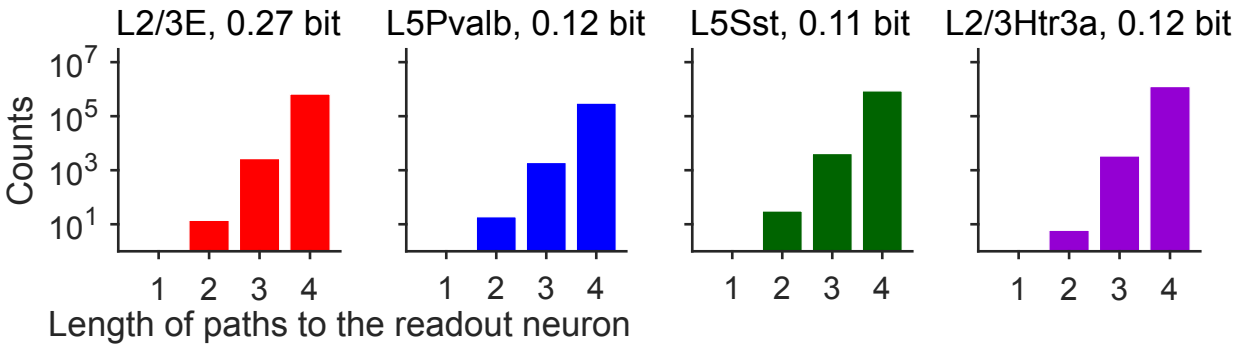


Figure S11: Histogram of the lengths (= number of synapses) of paths from 4 sampled early-informer-neurons with high MI to the readout neuron. We selected early-informer-neurons with the highest MI in four basic neuron types (excitatory, Pvalb, Sst, and Htra3 neurons). The paths from early-informer-neurons to the readout neuron were found by the MATLAB function "allpaths" in the directed graph. One can also find arbitrarily long paths; here we only demonstrate the short paths (length < 5) to the readout neuron for each of the early-informer-neurons. This distribution of path lengths suggests that the firing activity of an early-informer-neuron affects the membrane voltage of the readout neuron in multiple and diverse indirect ways.