

A Firing-Rate Neural Network Model for the Primary Visual Cortex and Its Learning Study

Huimiao Chen
2024-09-13

Outline

01 Introduction

02 Models & Data

03 Experiments & Results

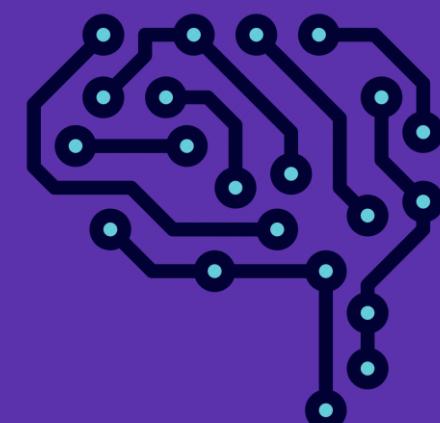
04 Discussion

05 Conclusions

A Firing-Rate Neural Network Model for the Primary Visual Cortex and Its Learning Study

KEY WORDS:

Firing-rate models;
Primary visual cortex;
Neural network learning;
Visual processing;
LGN filters.



01

Introduction

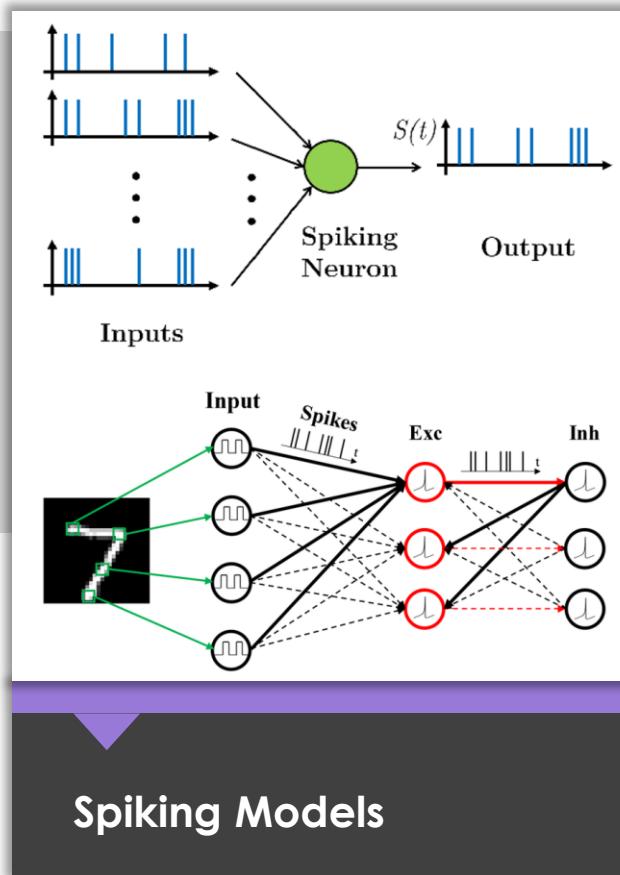
01 Introduction: Background (why firing-rate model?)

Neural network models: Spiking vs. Firing-Rate

Spiking Models

Advantages: Biologically realistic, precise spike timing, complex temporal dynamics.

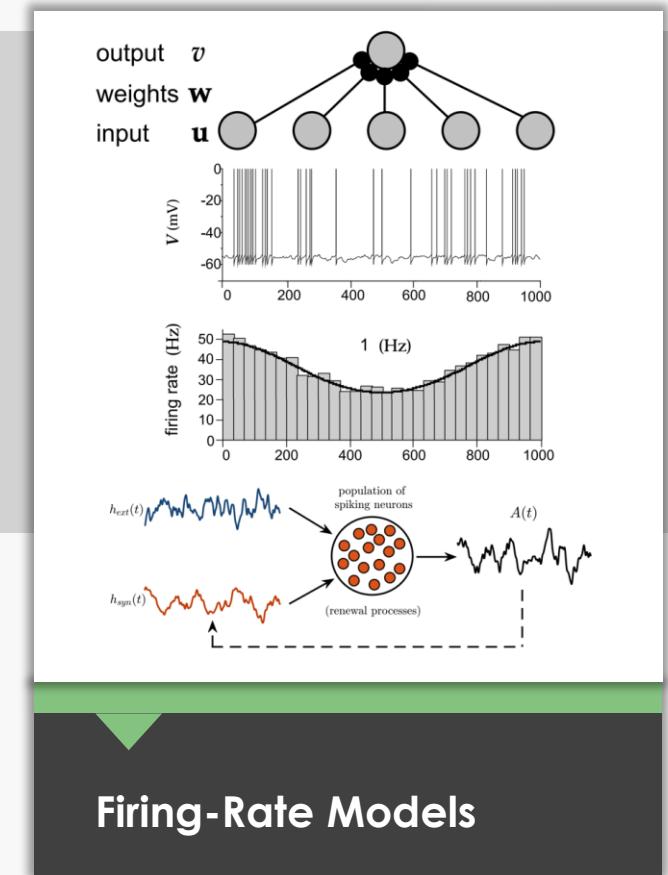
Disadvantages: Computationally intensive, challenging for analysis and integration.



Firing-Rate Models:

Advantages: Computationally efficient, mathematically tractable, easier to integrate, fewer parameters.

Disadvantages: No spike timing, potential oversimplification.



01 Introduction: Background (why visual cortex?)

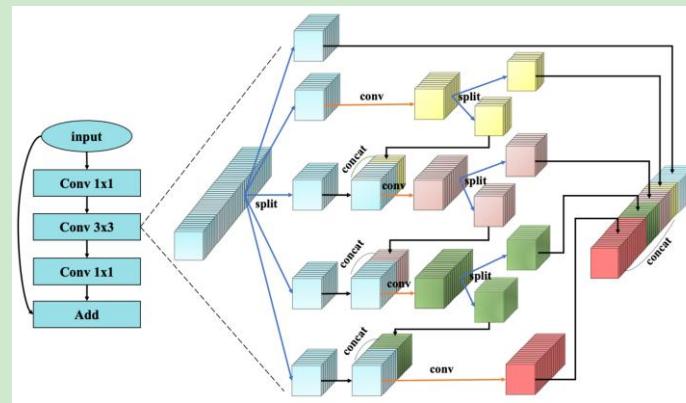
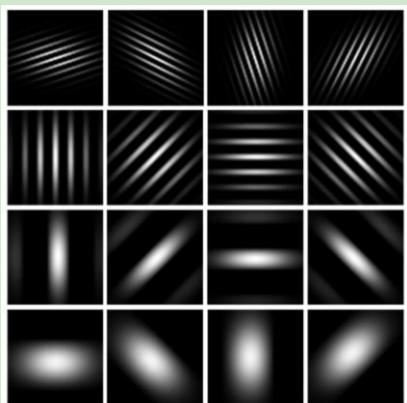
Visual systems are a crucial part of the mammalian brain, highly interconnected with other regions.

✓ Complexity of visual systems

- Reception, processing, and interpretation of visual information;
- Multiple levels from retina to LGN to cortical areas;

✓ Implications for computer vision

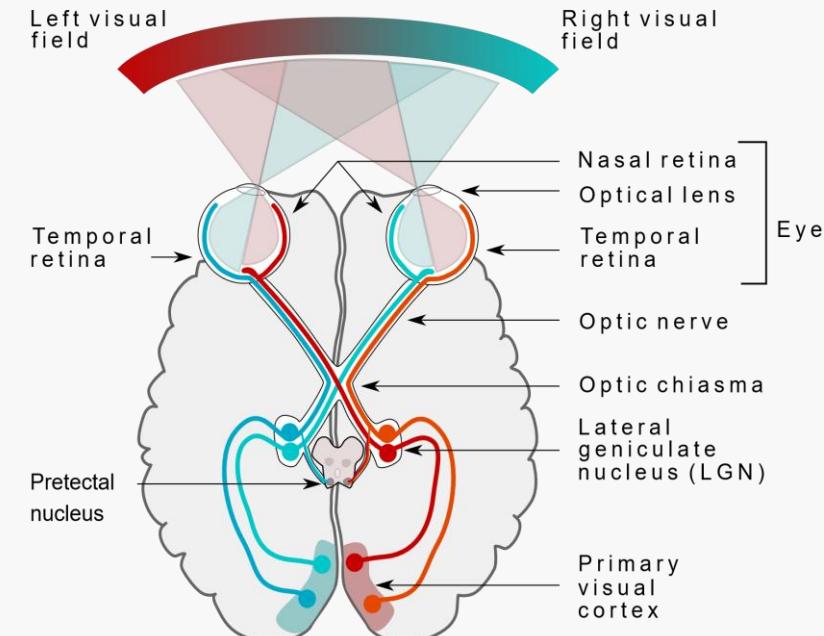
- Pattern recognition: e.g., Gabor filters;
- Imaging processing: e.g., convolutional neural networks (CNNs).



Despite advancements, many aspects of visual processing are not yet fully understood.

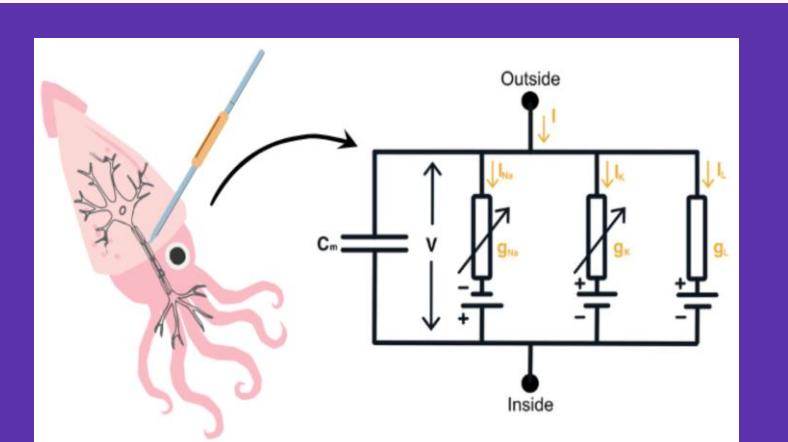
✓ Challenges

- Large-scale simulation and analysis
- Hybrid learning models
- Interaction with other regions



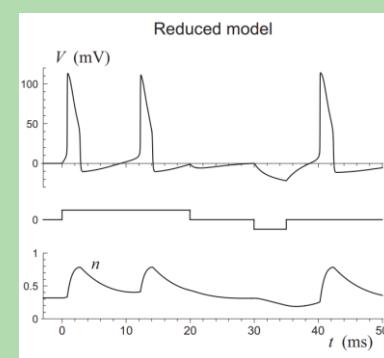
01 Introduction: Existing work

Spiking model development

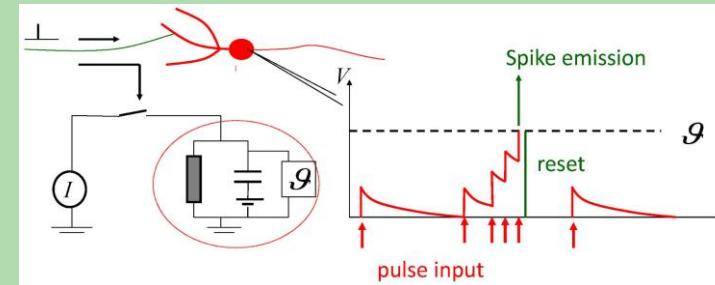


Hodgkin-Huxley (HH) model: A detailed biophysical model that describes the dynamics of sodium and potassium ion channels.

Simplified
→



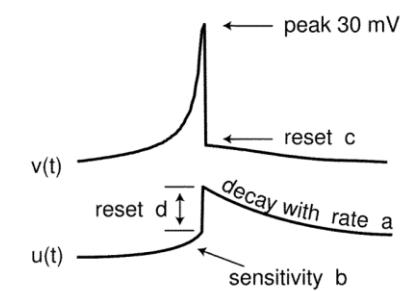
Reduced HH model: Simplified HH model that retains the essential spiking dynamics.



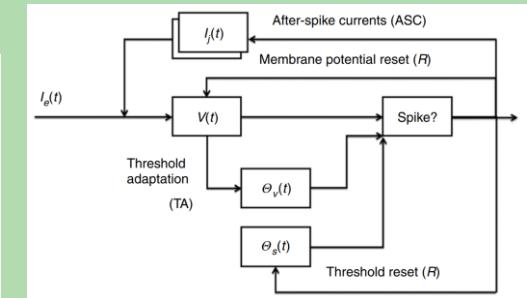
Leaky Integrate-and-Fire (LIF) model: A model that treats the neuron as a capacitor with a leak current, generating spikes with a threshold.

$$v' = 0.04v^2 + 5v + 140 - u + I$$
$$u' = a(bv - u)$$

if $v = 30$ mV,
then $v \leftarrow c$, $u \leftarrow u + d$



Izhikevich model: A two-dimensional spiking model that can reproduce a wide range of spiking patterns.



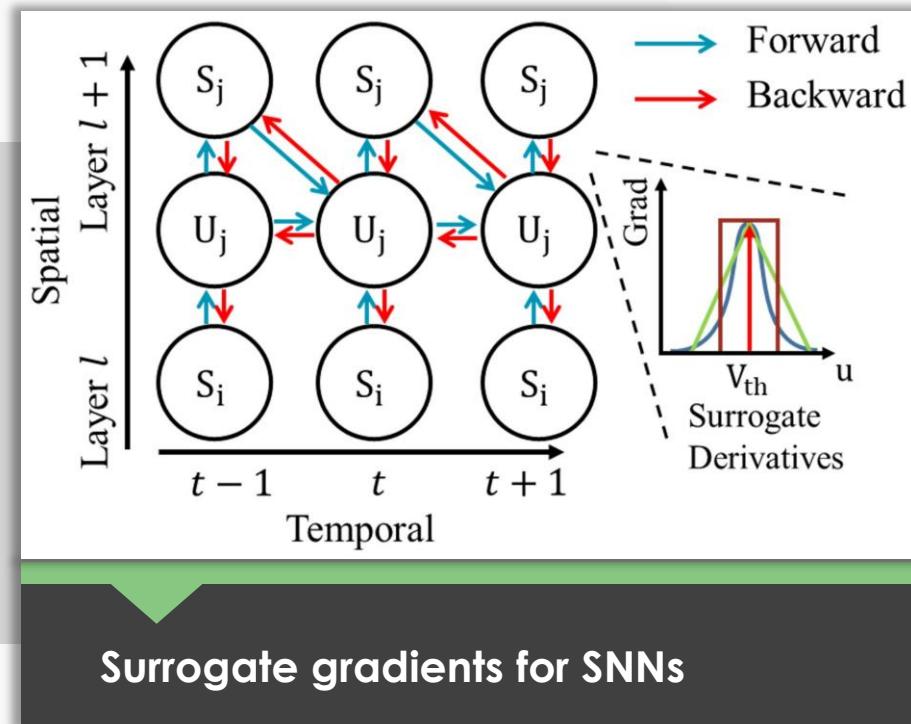
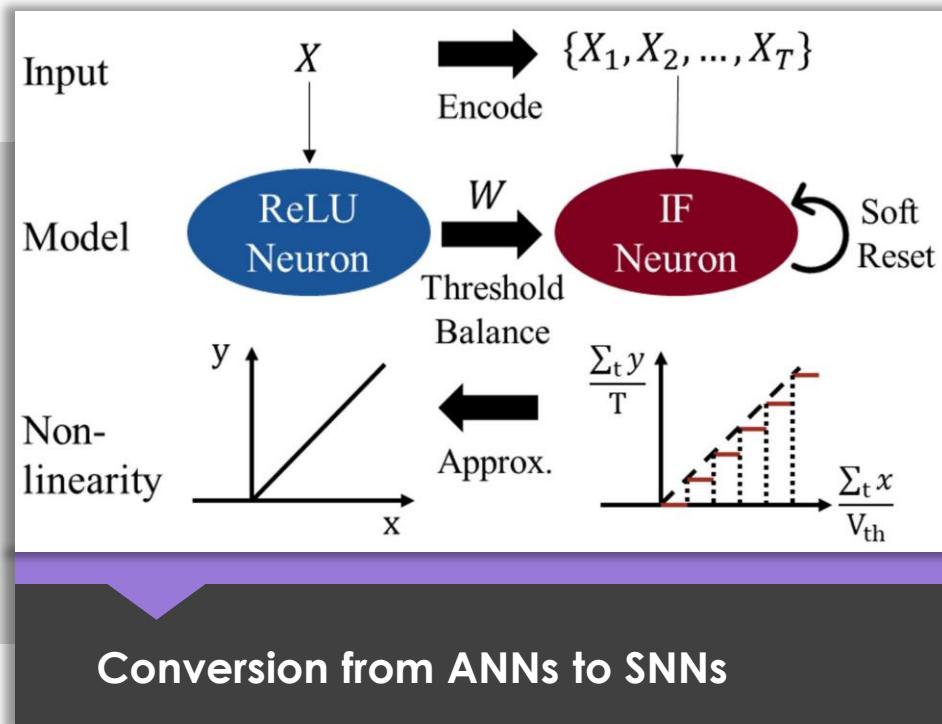
Generalized LIF model: An extension of the LIF model that incorporates additional features like adaptation.

The development of spiking models aims to **enhance computational efficiency** while capturing essential spiking dynamics, but these models **still face challenges in terms of computational complexity**.

01 Introduction: Existing work

Spiking neural network (SNN) Learning

Conversion from ANNs to SNNs and **Surrogate gradients** are two main approaches for training SNNs.



Successfully applied to well-known architectures like ResNet, but requiring longer inference time, and prioritizing performance over neuronal dynamics.

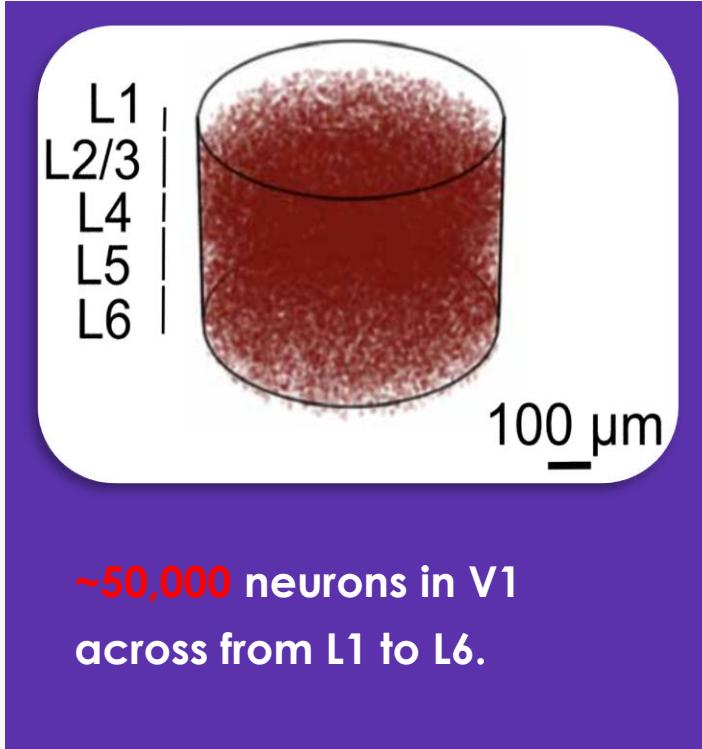
Achieving results comparable to deep learning, but introducing inherent inaccuracies in the descent direction and not compatible with neuromorphic hardware.

01 Introduction: Existing work

Large scale SNN models for V1

SNN models have been used for simulation of the V1 cortex.

However, the model training calls for **many GPUs** and **a large memory**.

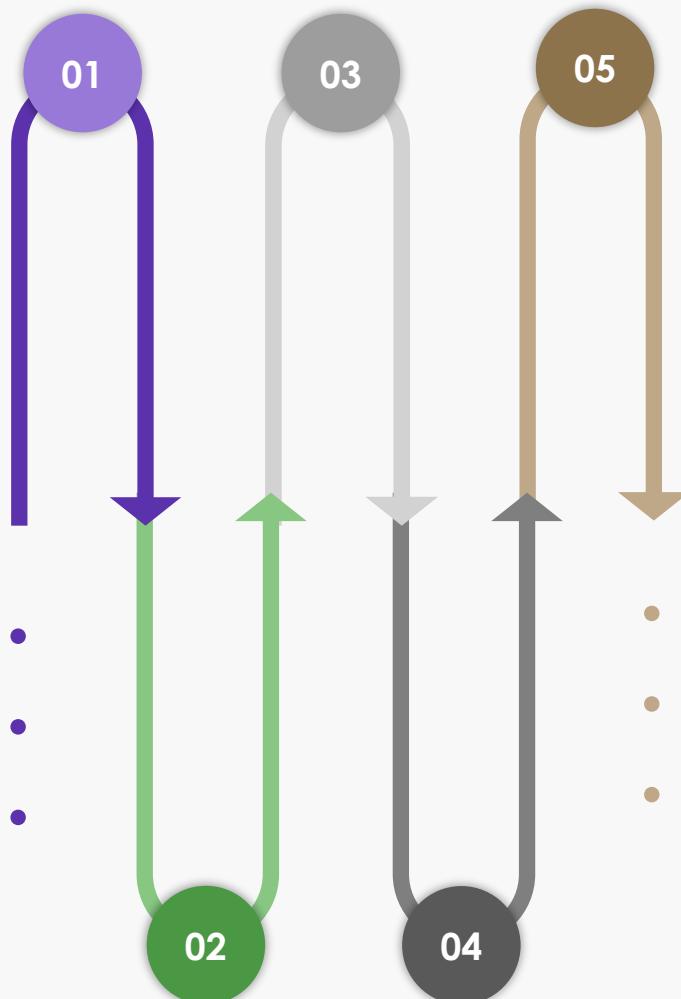


Requires



01 Introduction: Main work

Bridge the gap between experimental data and computational models for V1 learning.



01

Motivation

Building a biologically-constrained, data-driven firing-rate model of V1 to balance biological realism with computational efficiency and study its learning capabilities.

02

Transition from spiking to firing-rate models

Developing discrete form firing-rate models that enable the simulation and learning of biologically plausible visual processing in V1.

03

Use of real data from Allen Institute

Constructing the network using data from the Allen Institute for Brain Science, ensuring a biologically realistic representation of neuronal properties and connectivity.

04

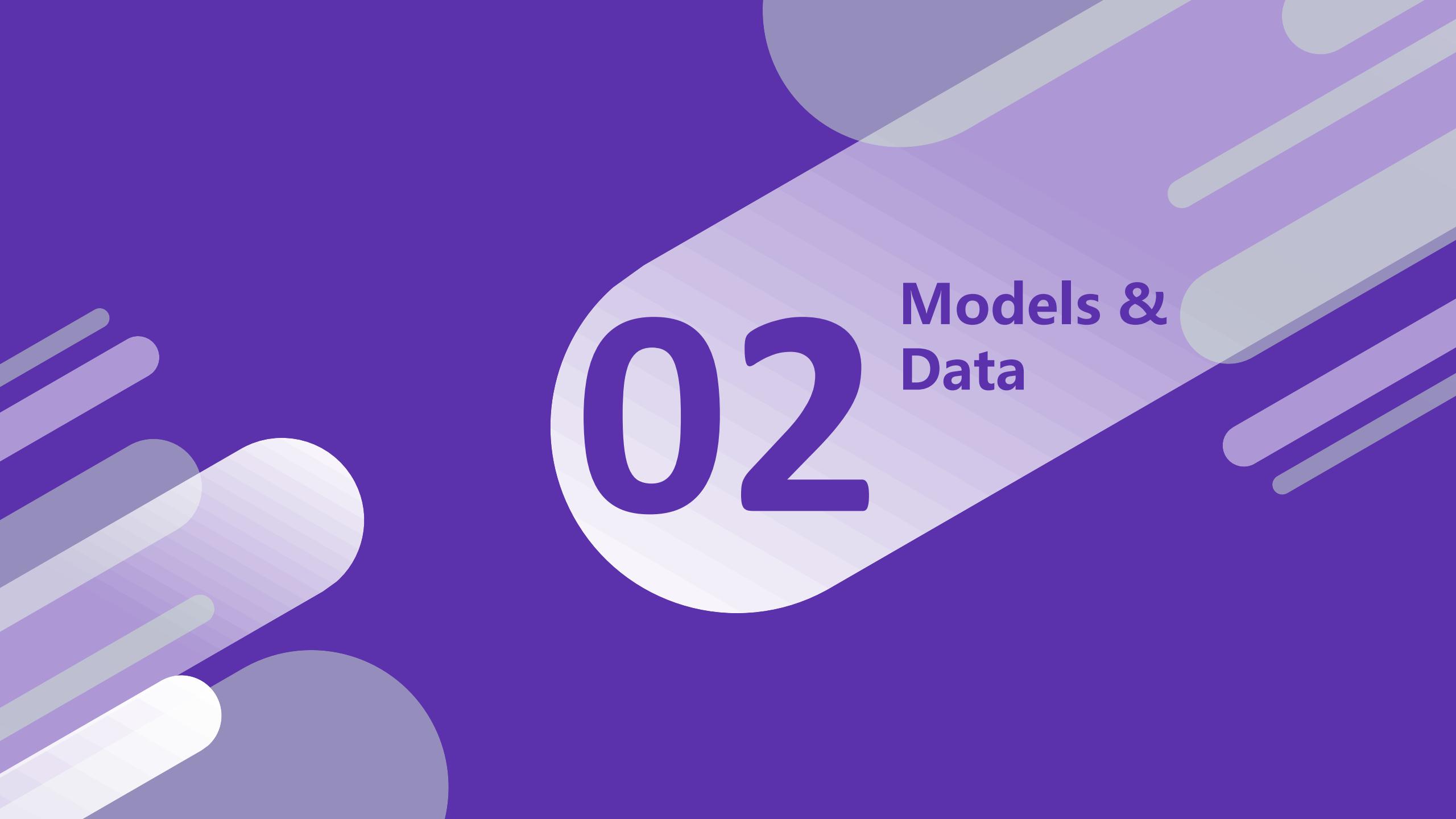
Learning evaluation with well designed cases

The learning of the V1 model is assessed using carefully designed visual tasks, different network sizes, multiple scenarios and other settings.

05

Analysis from various aspects

The model is analyzed from different aspects, including synaptic weight distributions, neuronal responses, robustness to noise, hybrid learning, biological realism, etc.

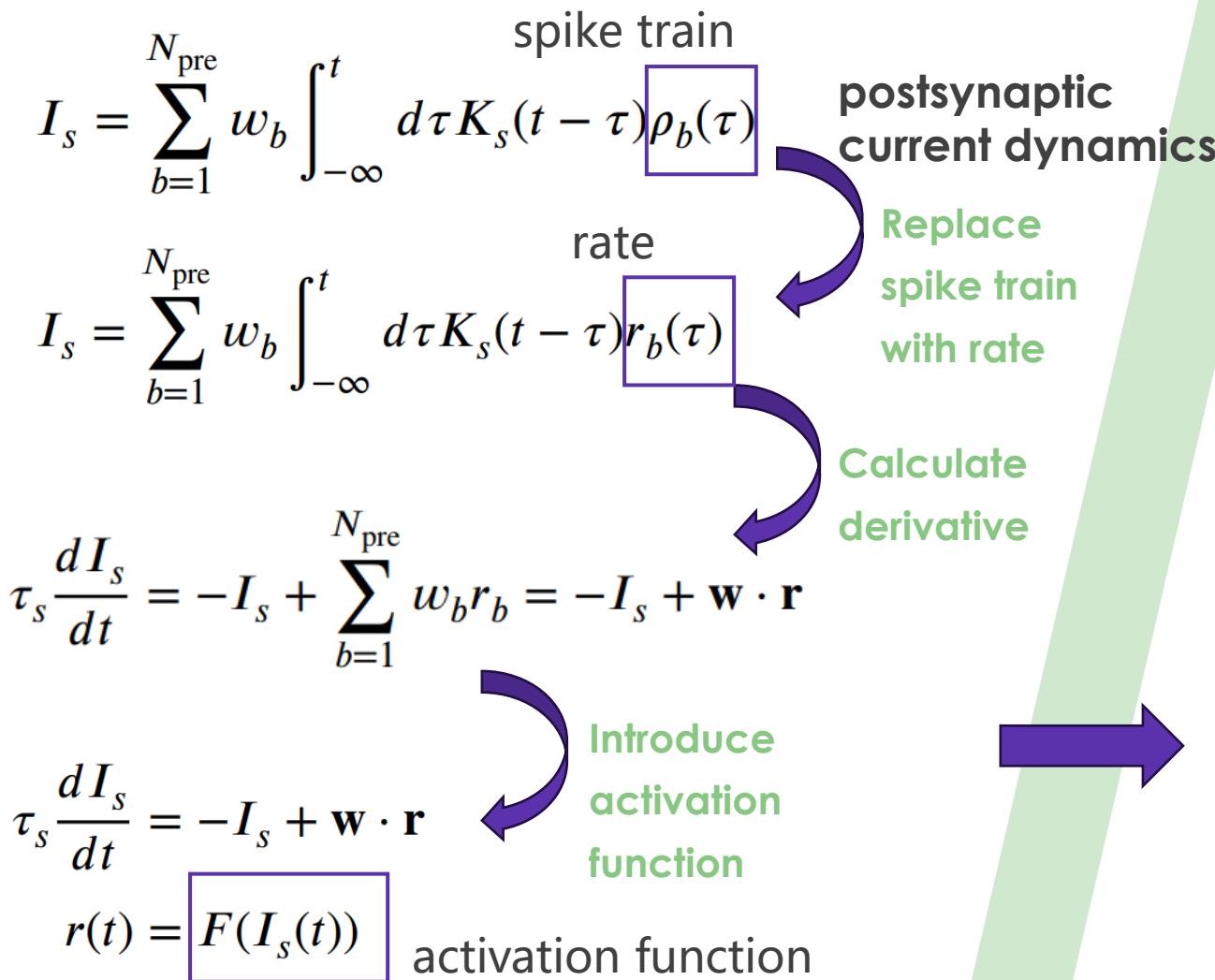


02

Models & Data

02 Models and Data

Models: From spiking to firing-rate



Introduce firing rate time constant τ_r

We have 3 versions of dynamics.

Instantaneous version:

$$\tau_s \frac{dI_s}{dt} = -I_s + \mathbf{w} \cdot \mathbf{r}$$

$$r(t) = F(I_s(t))$$

$$\tau_r \ll \tau_s$$

Low-pass filtered version:

$$\tau_s \frac{dI_s}{dt} = -I_s + \mathbf{w} \cdot \mathbf{r}$$

$$\tau_r \frac{dr(t)}{dt} = -r(t) + F(I_s(t))$$

$$\tau_r \approx \tau_s$$

Current-equilibrium low-pass filtered version:

$$\tau_r \frac{dr(t)}{dt} = -r(t) + F(\mathbf{w} \cdot \mathbf{r})$$

$$\tau_r \gg \tau_s$$

02 Models and Data

Models: From continuous to discrete

Difference (discrete) form:

derived using first-order exponential integrator method.

Instantaneous version:

$$I_s(t + \Delta t) = e^{-\frac{1}{\tau_s} \Delta t} I_s(t) + \left(1 - e^{-\frac{1}{\tau_s} \Delta t}\right) \mathbf{w} \cdot \mathbf{r}(t) \quad \tau_r \ll \tau_s$$

$$\mathbf{r}(t + \Delta t) = F(I_s(t + \Delta t))$$

Low-pass filtered version:

$$I_s(t + \Delta t) = e^{-\frac{1}{\tau_s} \Delta t} I_s(t) + \left(1 - e^{-\frac{1}{\tau_s} \Delta t}\right) \mathbf{w} \cdot \mathbf{r}(t) \quad \tau_r \approx \tau_s$$

$$\mathbf{r}(t + \Delta t) = e^{-\frac{1}{\tau_r} \Delta t} \mathbf{r}(t) + \left(1 - e^{-\frac{1}{\tau_r} \Delta t}\right) F(I_s(t))$$

Current-equilibrium low-pass filtered version:

$$\mathbf{r}(t + \Delta t) = e^{-\frac{1}{\tau_r} \Delta t} \mathbf{r}(t) + \left(1 - e^{-\frac{1}{\tau_r} \Delta t}\right) F(\mathbf{w} \cdot \mathbf{r}(t)) \quad \tau_r \gg \tau_s$$

Differential (continuous) form

We have 3 versions of dynamics.

Instantaneous version:

$$\tau_s \frac{dI_s}{dt} = -I_s + \mathbf{w} \cdot \mathbf{r} \quad \tau_r \ll \tau_s$$

$$\mathbf{r}(t) = F(I_s(t))$$

Low-pass filtered version:

$$\tau_s \frac{dI_s}{dt} = -I_s + \mathbf{w} \cdot \mathbf{r} \quad \tau_r \approx \tau_s$$

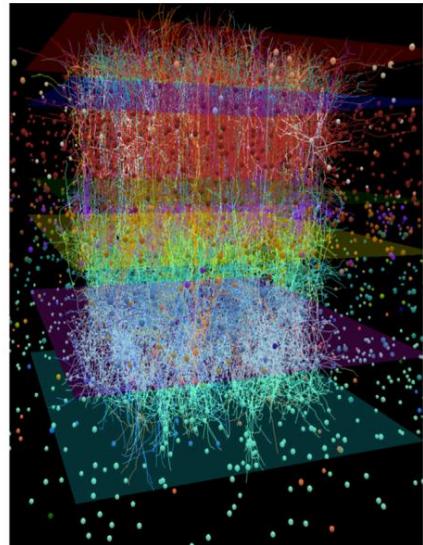
$$\tau_r \frac{dr(t)}{dt} = -r(t) + F(I_s(t))$$

Current-equilibrium low-pass filtered version:

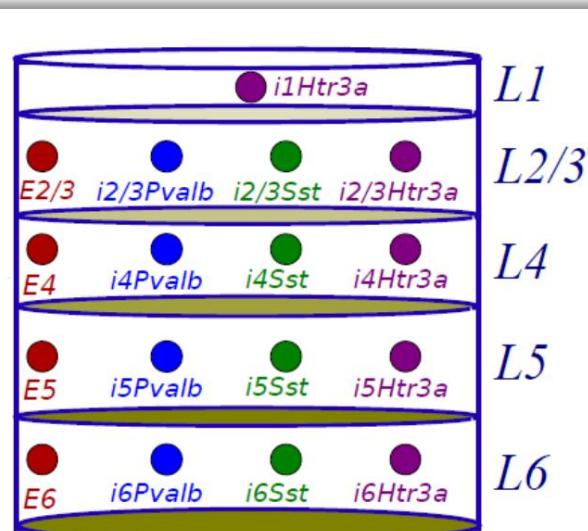
$$\tau_r \frac{dr(t)}{dt} = -r(t) + F(\mathbf{w} \cdot \mathbf{r}) \quad \tau_r \gg \tau_s$$

02 Models and Data

Data: From Allen Institute



(a) A small portion of neurons and their connections in V1 are visualized.



(b) Excitatory and inhibitory neuronal classes across layers from L1 to L6 in V1 are shown. L2/3 to L6 have all four classes of neurons: one excitatory class and three inhibitory classes: Pvalb, Sst, and Htr3a, while L1 has only one inhibitory class: Htr3a.

We select **3k-**
and **6k-neuron**
networks for
study.

A cylinder
column is cut
from the data.

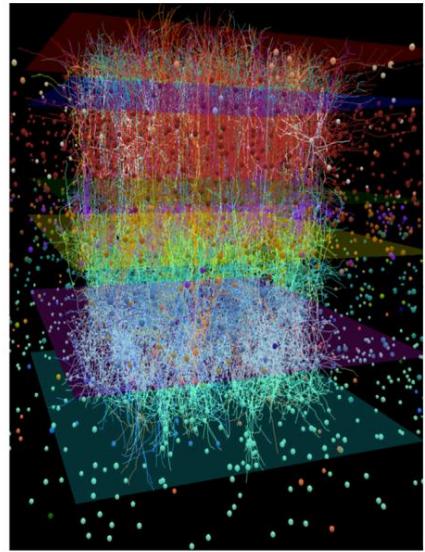
Neuron class	(3k network) Neuron number	(6k network) Neuron number
L1 i	59	113
L2/3 e	753	1508
L2/3 i	129	250
L4 e	593	1189
L4 i	94	213
L5 e	455	907
L5 i	64	133
L6 e	725	1425
L6 i	128	262

Distribution of excitatory and inhibitory neurons in network layers

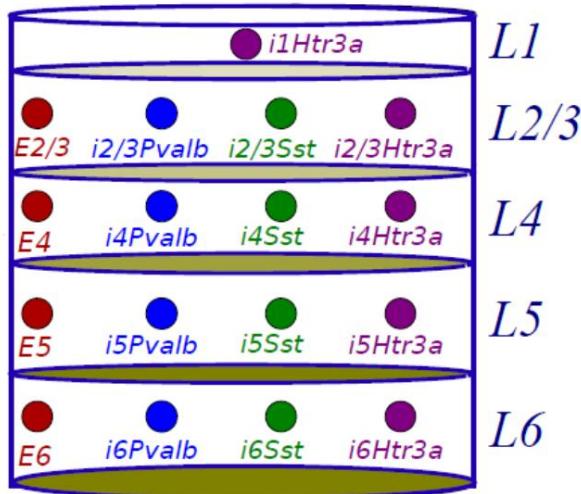
V1 neural network visualization and cell classes (from mouse brain)

02 Models and Data

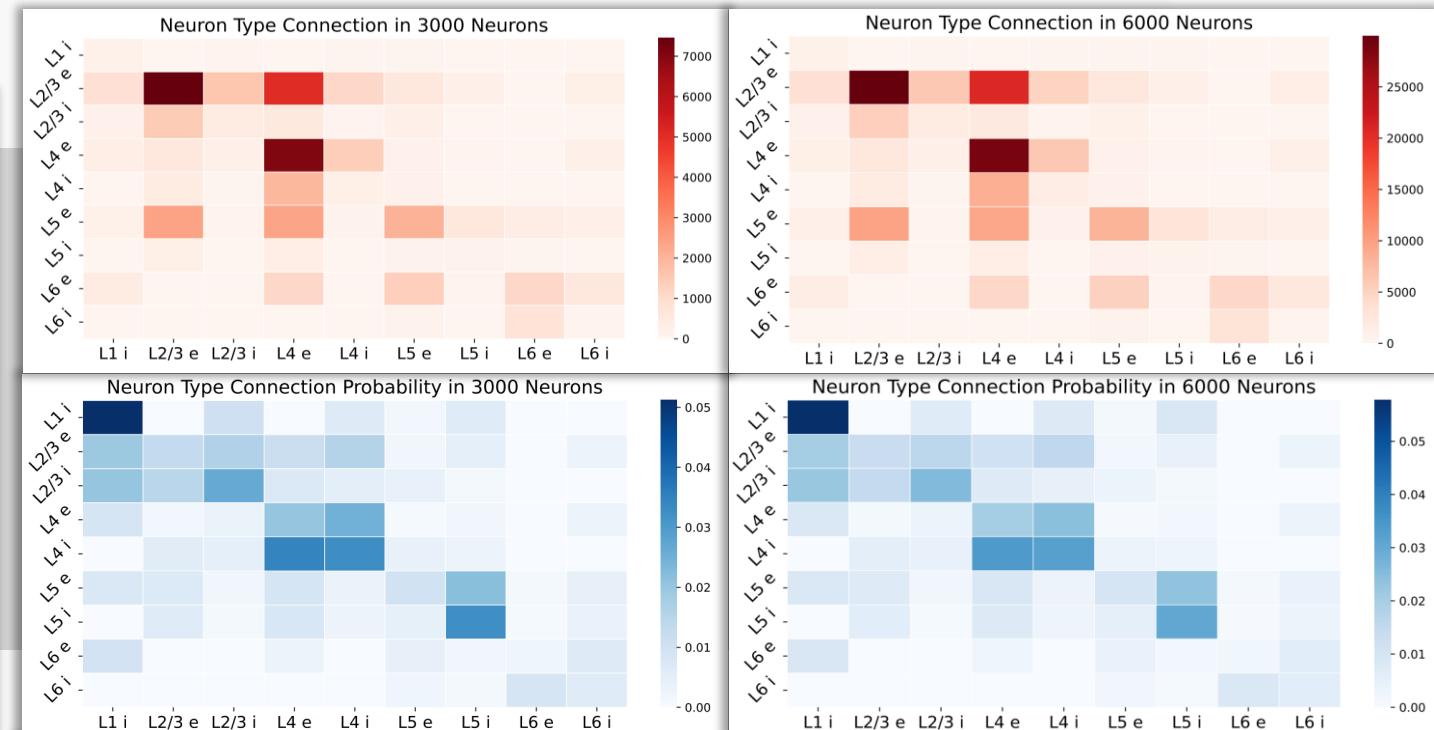
Data: From Allen Institute



(a) A small portion of neurons and their connections in V1 are visualized.



(b) Excitatory and inhibitory neuronal classes across layers from L1 to L6 in V1 are shown. L2/3 to L6 have all four classes of neurons: one excitatory class and three inhibitory classes: Pvalb, Sst, and Htr3a, while L1 has only one inhibitory class: Htr3a.

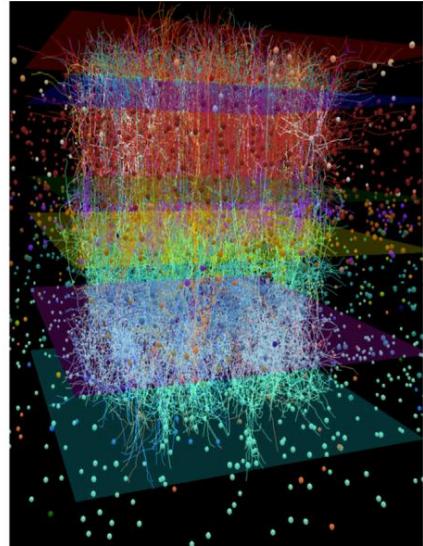


Synaptic connectivity count and probability matrix for neuronal classes (3k and 6k networks)

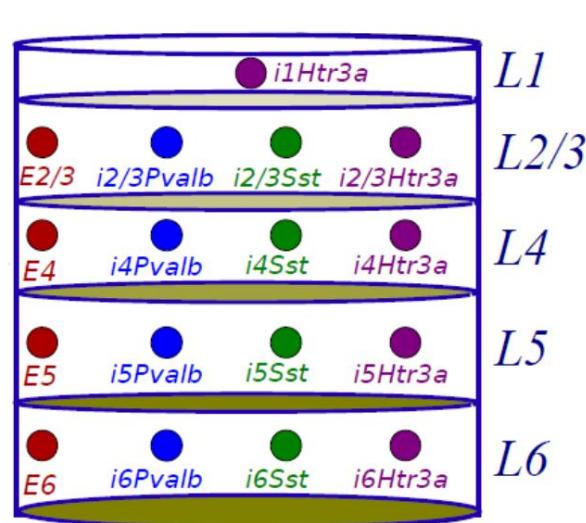
Observe: A forward flow of visual data from L4 to L2/3 and then to L5 and L6.

02 Models and Data

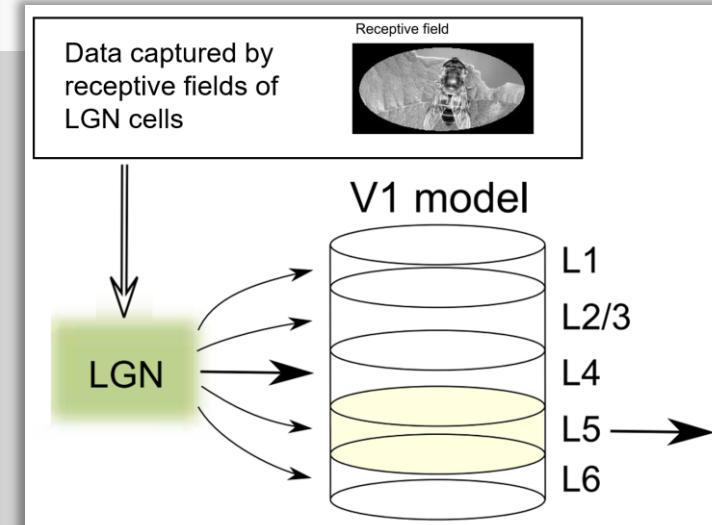
Data: From Allen Institute



(a) A small portion of neurons and their connections in V1 are visualized.



(b) Excitatory and inhibitory neuronal classes across layers from L1 to L6 in V1 are shown. L2/3 to L6 have all four classes of neurons: one excitatory class and three inhibitory classes: Pvalb, Sst, and Htr3a, while L1 has only one inhibitory class: Htr3a.



Inputs: Half are delivered to L4 and the rest are randomly delivered to other layers.

Outputs: All are extracted from L5 by default.

Other parameters:

Synaptic time constants
are adapted from the data;

Firing-rate time constants
are sampled based on the
synaptic time constants.

V1 neural network visualization and cell classes (from mouse brain)

03

Experiments & Results

03 Experiments and Results

Visual tasks



Fine-orientation detection task

The drifting gratings orientation is between 43 and 47 degrees. The model predicts whether the orientation of the gratings exceeded 45 degrees.



Image classification task

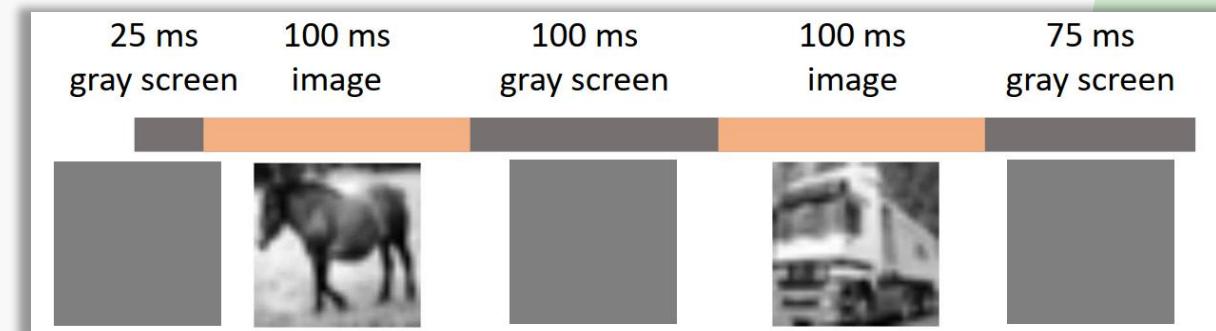
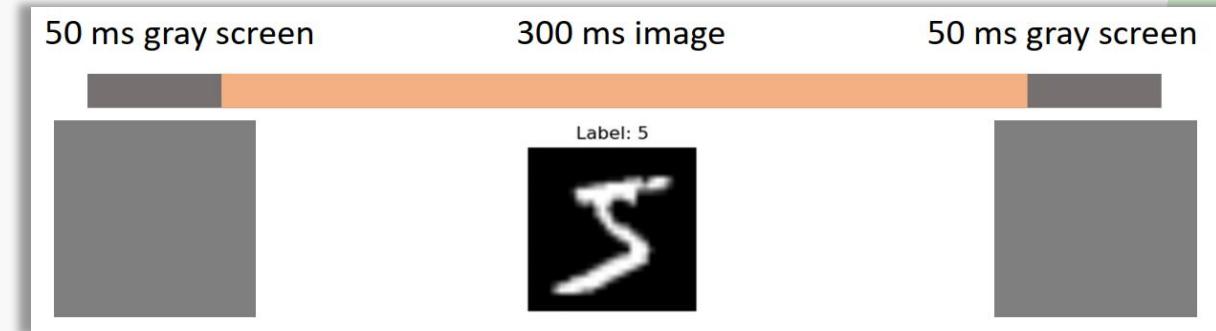
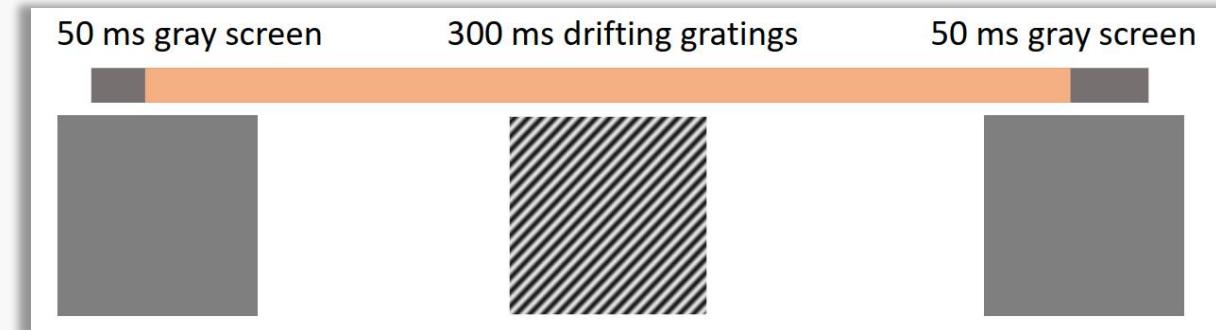
The core stimulus is an image from the MNIST dataset, containing a handwritten digit ranging from 0 to 9. The model performs 10-class classification.



Visual change task

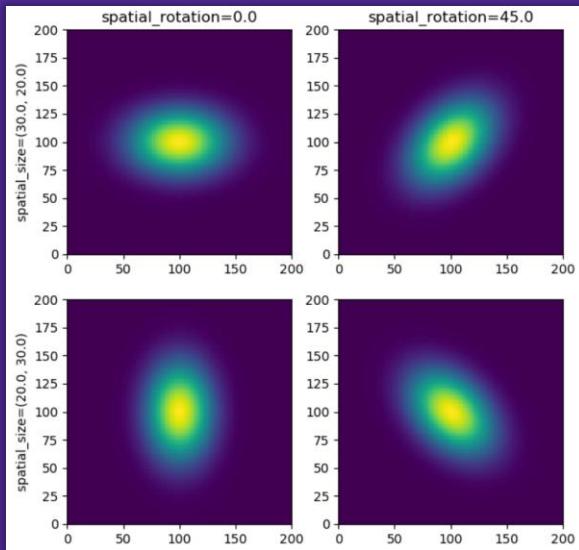
The first and second images are both from the CIFAR-10 dataset, which can be same or different. The model predicts whether the image is changed.

Task schematic diagrams

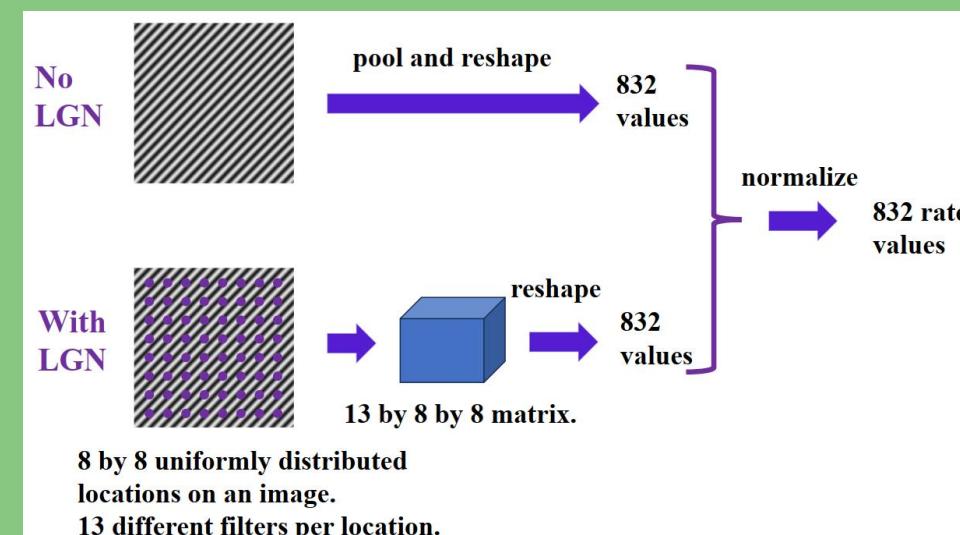


03 Experiments and Results

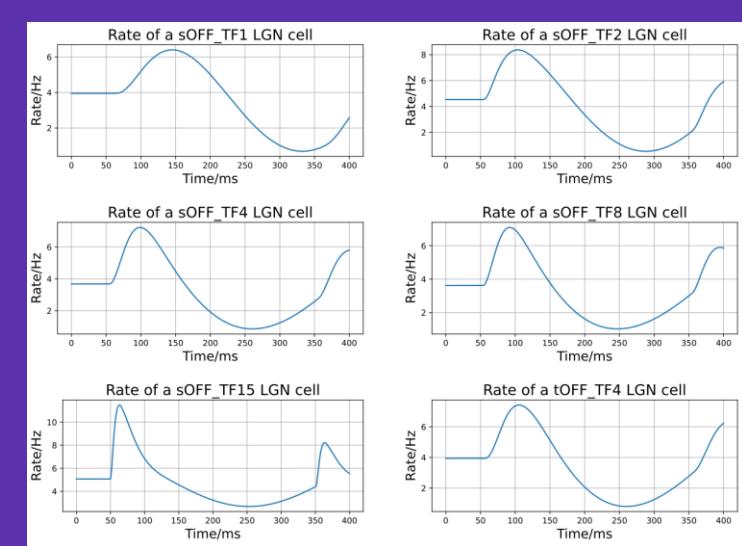
LGN filters



LGN cells with different spatial settings



Data preprocessing from images to rates with and without LGN



LGN cell responses to visual stimuli of the fine-orientation detection task

13 types of LGN cells are used to capture visual features. For each type, 64 same cells are put across an input image, arranged in an 8x8 grid. This generates an output of 832 channels, which then feed into the V1 model.

03 Experiments and Results

Scenarios

01

Default scenario

With neuronal diversity, network connections from data, and Dale's law. Use backpropagation for learning.

02

“Uniform neurons” scenario

Neuronal diversity removed: all excitatory/inhibitory neurons share the same values for their synaptic and firing-rate time constants.

03

“Random synapses” scenario

First organizing all n neurons connected like a tree with $n - 1$ synapses, and then randomly connecting the rest synapses (better connectivity).

04

“No Dale’s law” scenario

Allowing neurons to deliver positive, negative or a mix of positive and negative weights to downstream neurons

05

Hybrid learning (special case)

Hebbian learning replaces backpropagation for recurrent connections within a layer.

02 Models and Data

Results: by scenario and LGN

(3k, no LGN)	All tasks	Fine-orientation detection	Image classification	Visual change detection
Default scenario	81.13%	100.00%	76.48%	68.07%
Uniform neurons	77.71%	100.00%	71.67%	62.67%
Random synapses	95.35%	100.00%	90.05%	96.48%
No Dale's law	95.66%	100.00%	92.43%	94.98%

(3k, with LGN)	All tasks	Fine-orientation detection	Image classification	Visual change detection
Default scenario	85.07%	99.82%	68.00%	89.40%
Uniform neurons	80.84%	96.73%	66.62%	82.90%
Random synapses	95.45%	100.00%	87.75%	99.17%
No Dale's law	95.01%	100.00%	86.37%	98.95%

(6k, no LGN)	All tasks	Fine-orientation detection	Image classification	Visual change detection
Default scenario	80.56%	100.00%	73.72%	69.30%
Uniform neurons	78.08%	100.00%	75.47%	62.30%
Random synapses	97.48%	100.00%	94.57%	98.30%
No Dale's law	95.76%	100.00%	93.28%	94.73%

(6k, with LGN)	All tasks	Fine-orientation detection	Image classification	Visual change detection
Default scenario	90.33%	100.00%	73.30%	98.28%
Uniform neurons	87.58%	100.00%	69.57%	94.77%
Random synapses	95.44%	100.00%	87.40%	99.50%
No Dale's law	95.64%	100.00%	87.58%	99.80%

 "Random synapses" and "no Dale's law" are best-performance scenarios.

 Without LGN, increasing the neuron count from 3k to 6k does not lead to a significant improvement in performance.

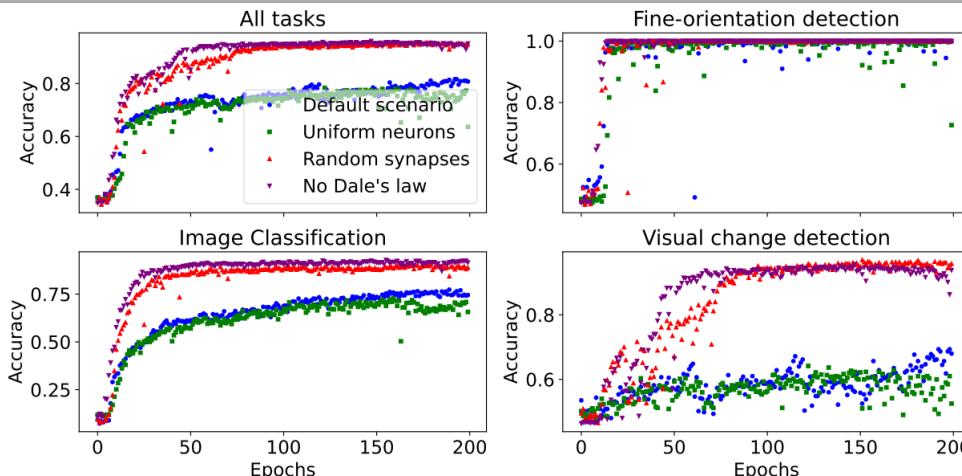
 With LGN, increasing the neuron count from 3k to 6k improves the performance of default and "uniform neurons" scenario.

 LGN enhances the model's ability to capture and utilize temporal information which is crucial for visual change detection.

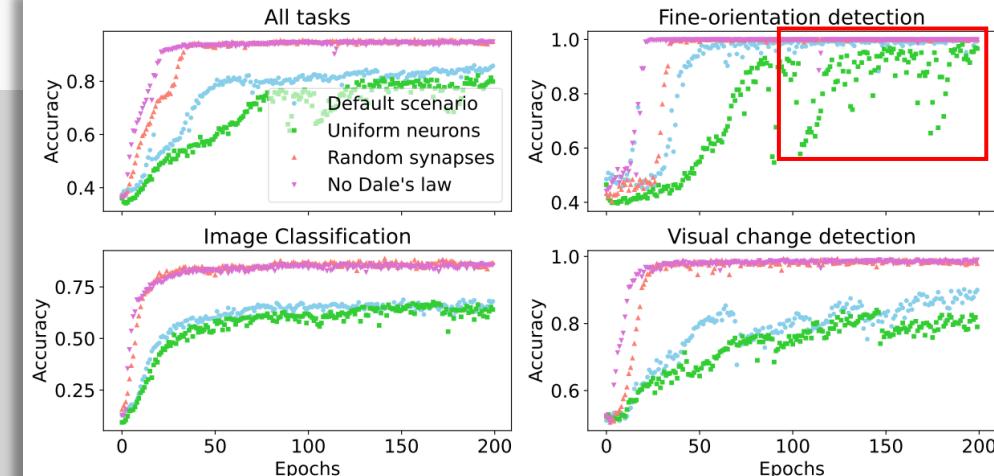
 With LGN, the performance of other two tasks are slightly compromised as a trade-off to improve the performance of all tasks.

02 Models and Data

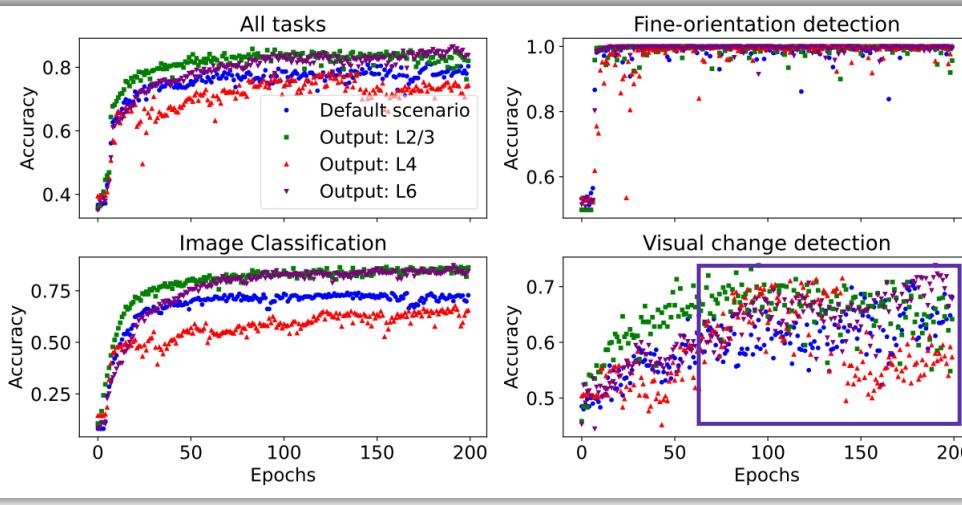
Results: by scenario and LGN



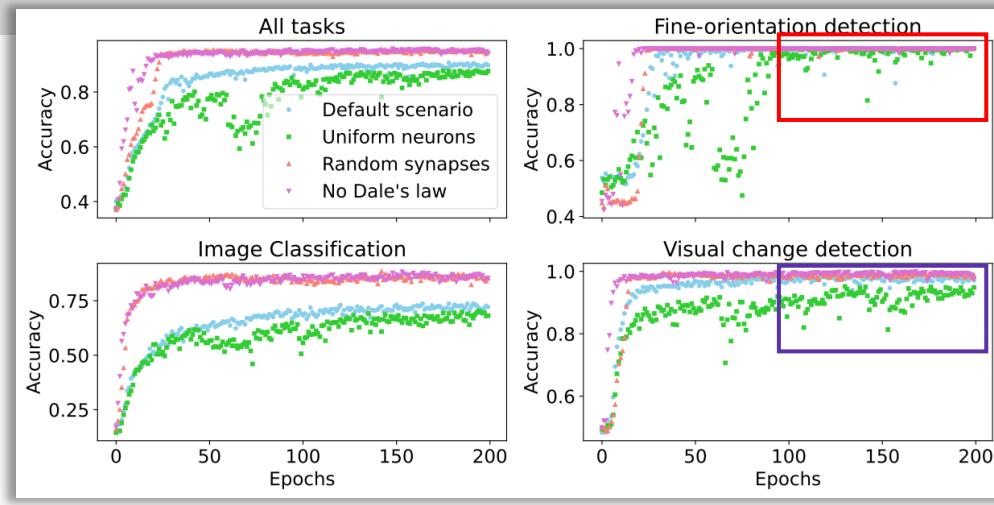
(3k, no LGN)



(3k, with LGN)



(6k, no LGN)



(6k, with LGN)

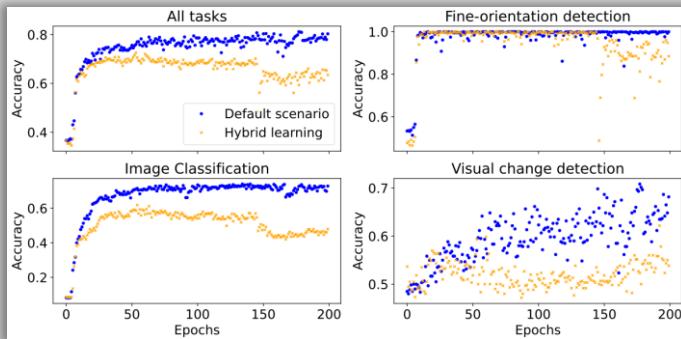
Increasing the network size alone does not significantly improve the performance, and LGN filters alone do not significantly contribute to model convergence, but the combination of increased network size and LGN preprocessing can further improve the model's performance and convergence.



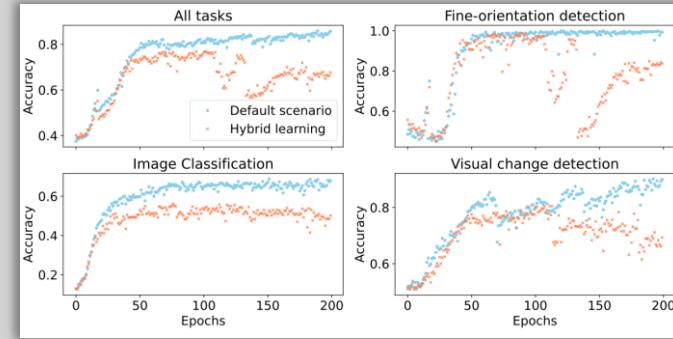
02 Models and Data

Results: by hybrid learning

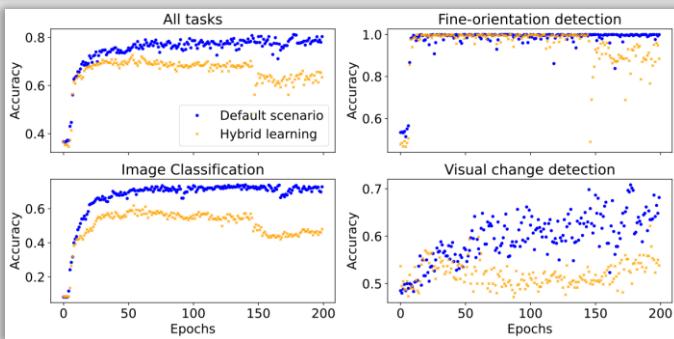
(3k, no LGN)	All tasks	Fine-orientation detection	Image classification	Visual change detection
Default scenario	80.56%	100.00%	74.72%	69.30%
Hybrid learning	71.24%	100.00%	59.52%	57.83%



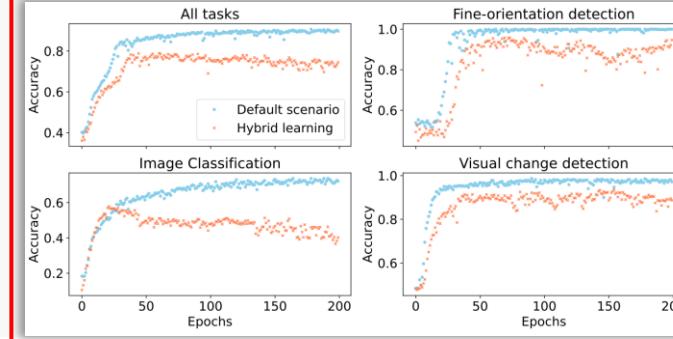
(3k, with LGN)	All tasks	Fine-orientation detection	Image classification	Visual change detection
Default scenario	80.56%	100.00%	74.72%	69.30%
Hybrid learning	71.24%	100.00%	59.52%	57.83%



(6k, no LGN)	All tasks	Fine-orientation detection	Image classification	Visual change detection
Default scenario	85.07%	99.82%	68.00%	89.40%
Hybrid learning	76.61%	98.02%	55.35%	79.90%



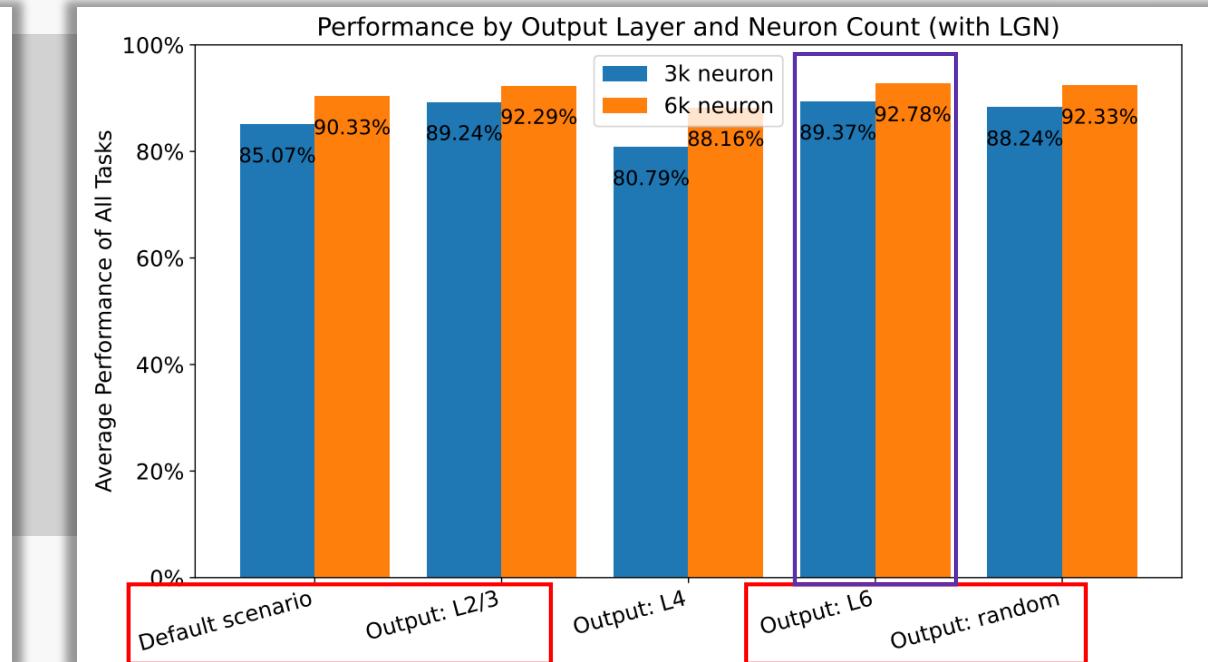
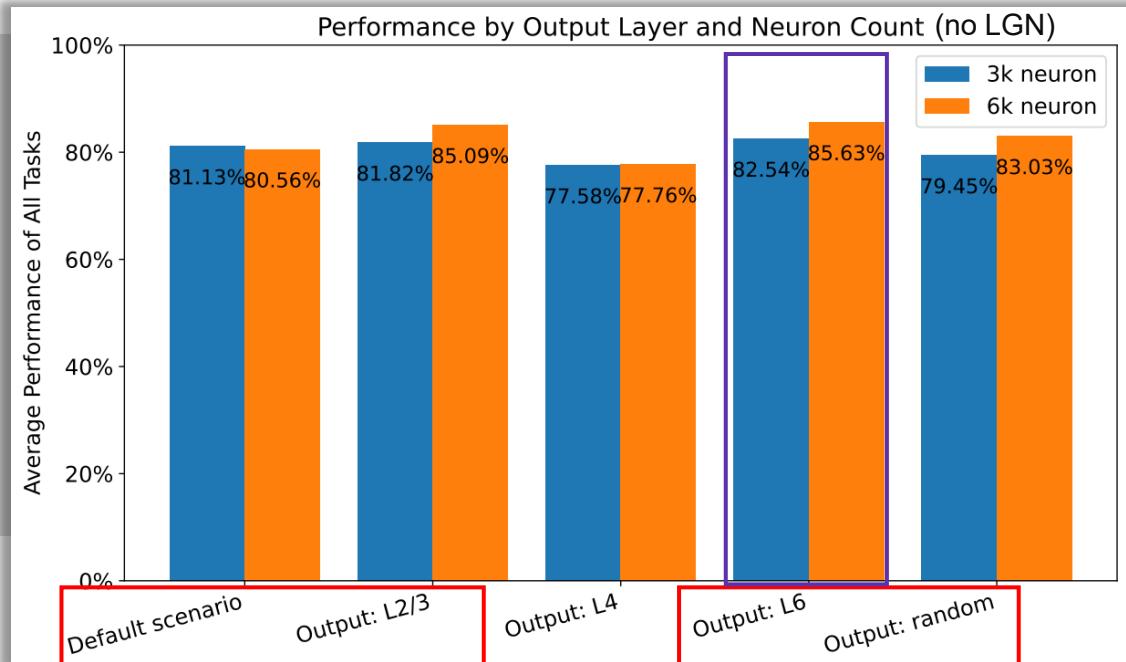
(6k, with LGN)	All tasks	Fine-orientation detection	Image classification	Visual change detection
Default scenario	90.33%	100.00%	73.30%	98.28%
Hybrid learning	78.53%	95.18%	56.58%	92.67%



A larger network size and LGN preprocessing can make hybrid learning work better!

02 Models and Data

Results: by output layer and LGN



Deeper output layers



Deepest output layer



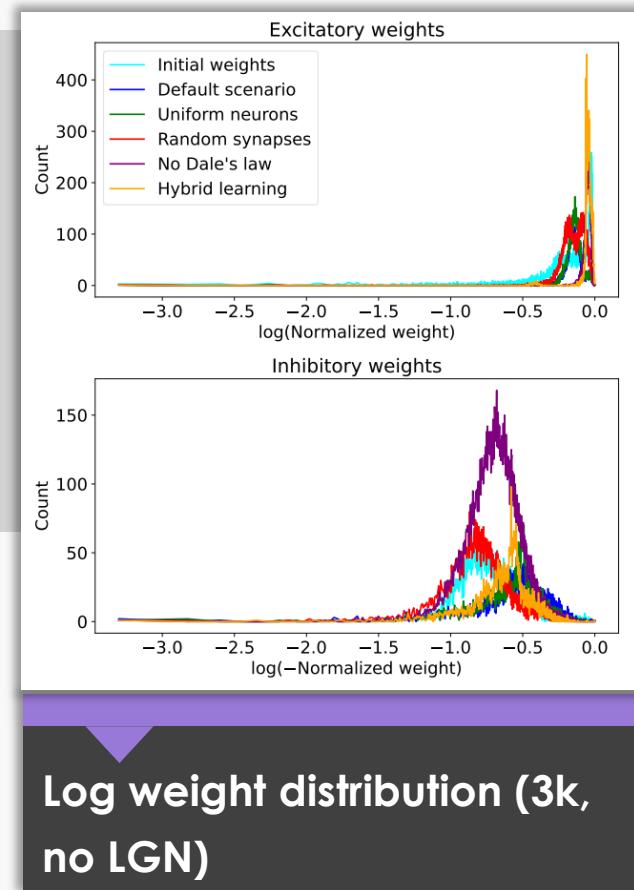
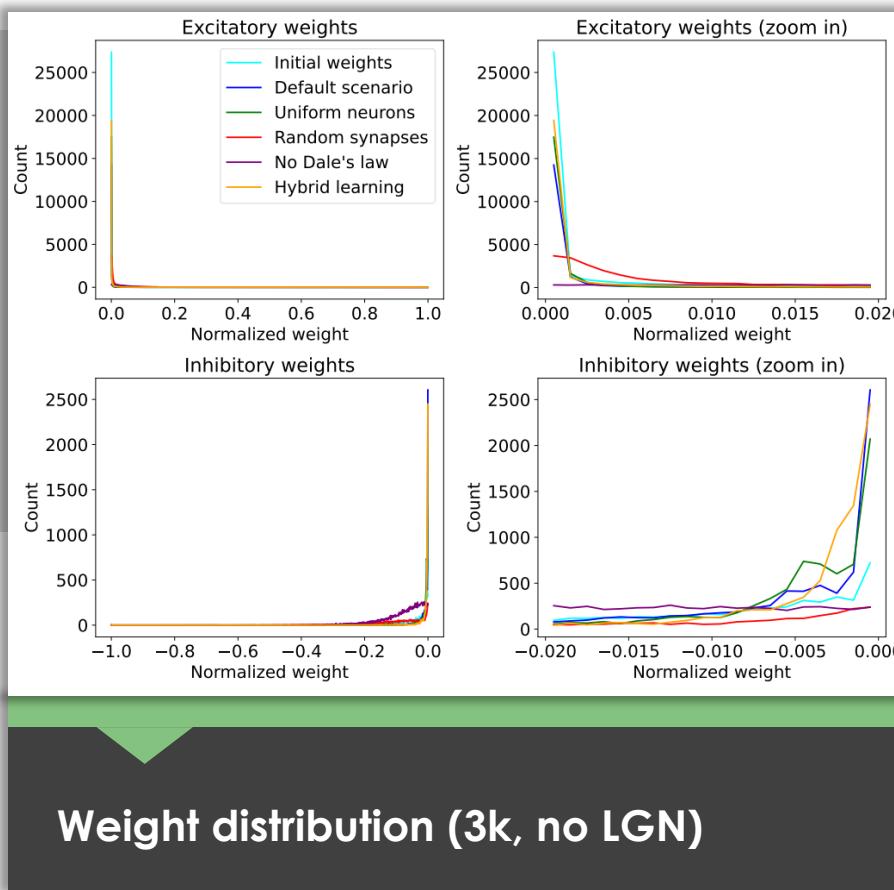
Deeper output, better performance.

Discussion

04

04 Discussion

Weight distribution

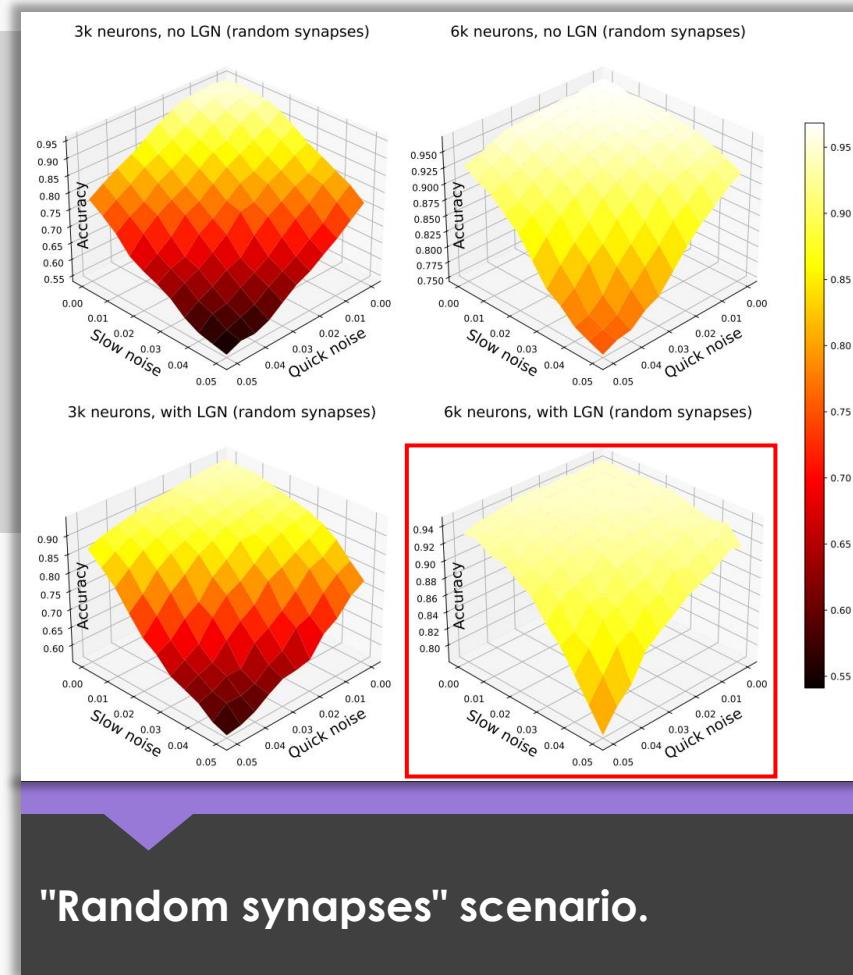
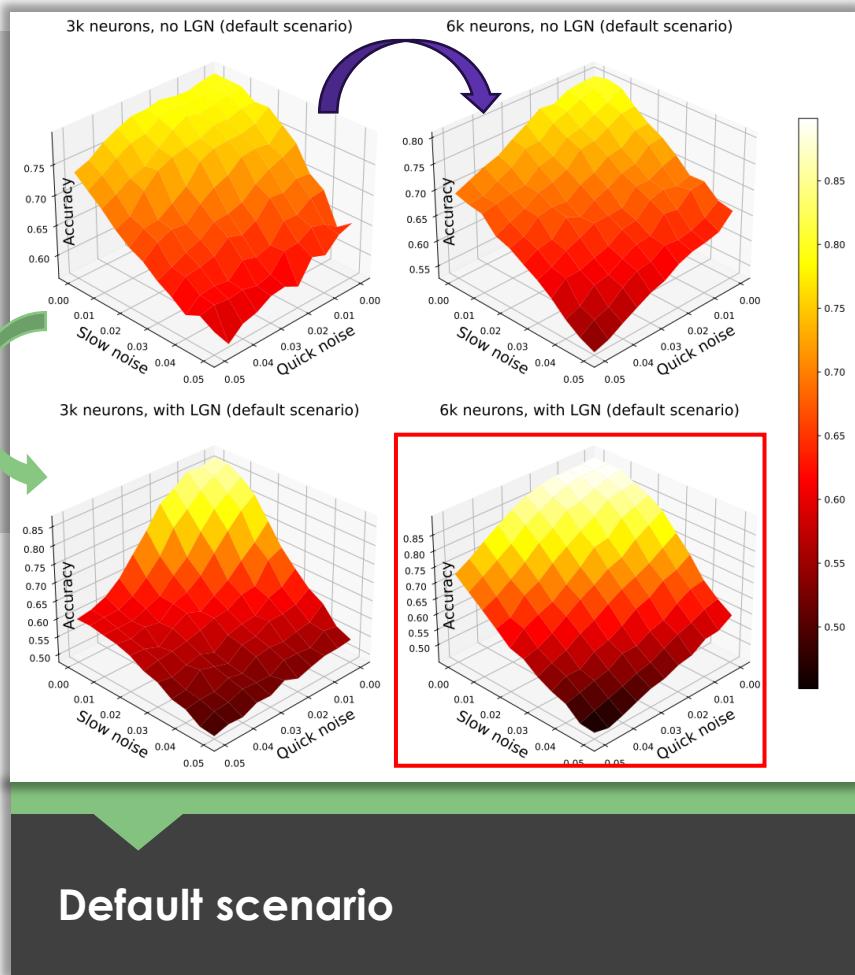


Scenarios with **more widespread weight distributions** tend to perform better, indicating a diverse range of weights may be beneficial for learning complex patterns

The **log-normal distribution** of weights shows an indication of biological realism.

04 Discussion

Internal noise



Slow noise: same for all neurons;
Quick noise: specific to each neuron.

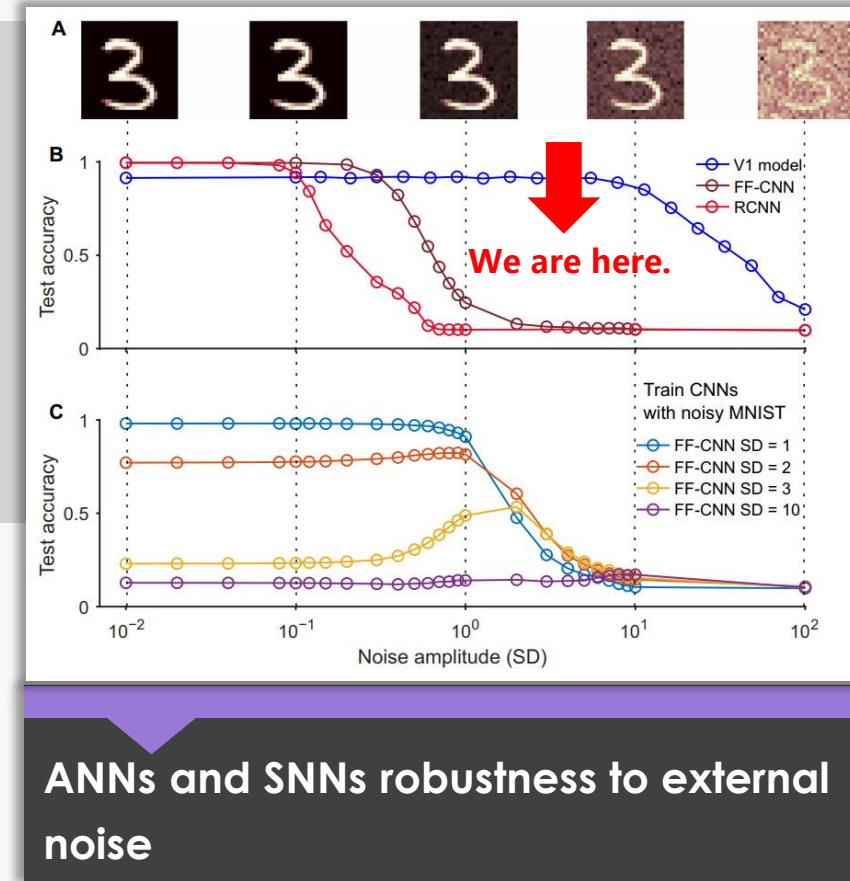
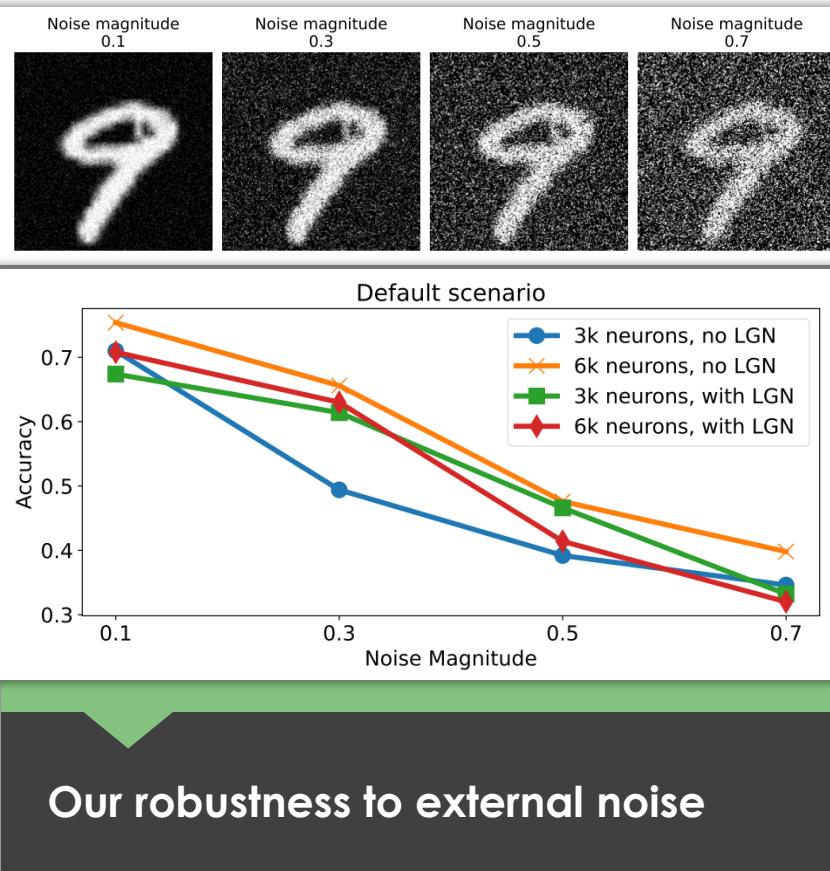
Higher recurrency, better robustness to internal noise.

The effectiveness of **LGN filters** in improving the model's robustness to internal noise is relatively modest compared to the benefits of **increasing the network size**.

"Random synapses" scenario's performance decreases slower

04 Discussion

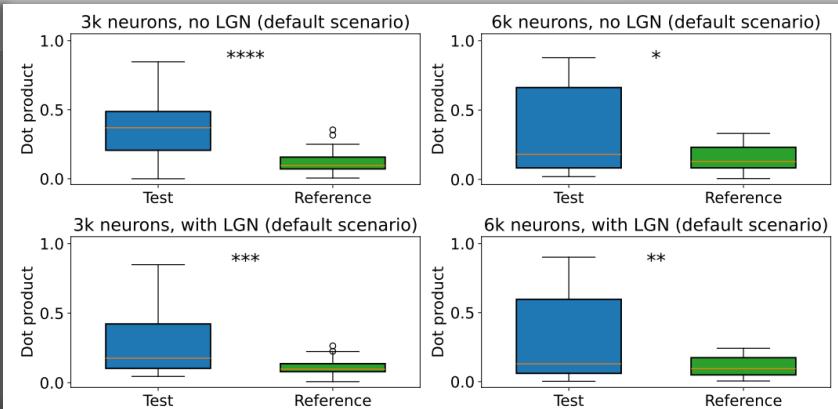
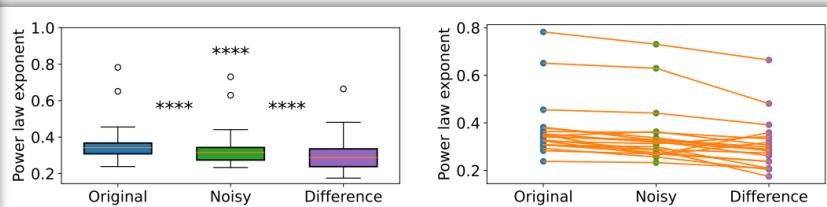
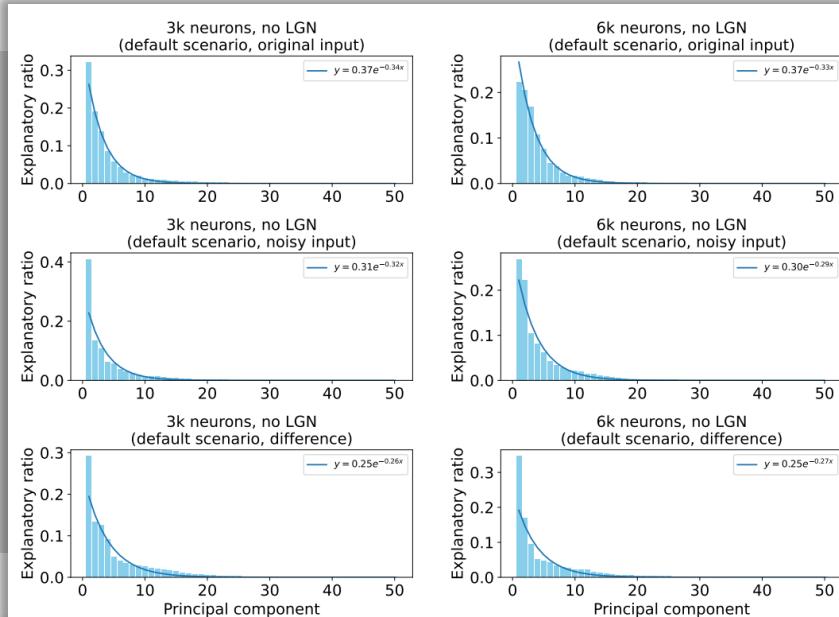
External noise



Our model shows robustness to external noise between typical ANNs and SNNs (10 times larger in scale).

04 Discussion

Neuronal responses



PCA results of neuronal responses to MNIST images

Significance analysis of power law exponents

Despite the external noise, the principal components **capturing the most variance** in the responses remain dominant. The principal components of the neuronal responses follow **a power law distribution**, indicating a balance between high precision and robustness, and suggesting biological realism.

Significance analysis of dot product of between top 5 principal components

05 Conclusions

05 Conclusions

Based on real data of V1 cortex from the Allen Institute for Brain Science, we developed a firing-rate model and investigated its learning capabilities across various visual tasks, network sizes, and scenarios.

Transition from spiking to firing-rate models and development of discrete form firing-rate models.

The transition from spiking models to firing-rate models addresses the need for computationally efficient yet biologically plausible models, which cannot be achieved by existing SNNs. The discrete form firing-rate models developed in this thesis provide a framework for large-scale simulation.

Impact of network size and LGN filter preprocessing on model performance and convergence.

LGN helps the model to capture temporal information. Increasing the network size alone does not significantly improve the performance, and LGN filters alone do not significantly contribute to model convergence, but the combination of increased network size and LGN preprocessing can further improve the model's performance and convergence. A larger network size and LGN preprocessing also enhance hybrid learning.

Evaluation of learning capabilities through visual tasks and scenarios / settings.

The learning capabilities of the models are evaluated using three designed visual tasks: fine-orientation detection, image classification, and visual change detection. Multiple learning scenarios, such as "uniform neurons," "random synapses," "no Dale's law," and different output layer settings, and hybrid learning, are explored to assess the models' performance under different conditions

Discussion of the biological realism and robustness through multiple aspects.

The log-normal distribution of synaptic weights is consistent with biological findings. Widespread weight distributions lead to better performance. The model exhibits resilience to both internal and external noise to some extent. Higher recurrency offering more benefits in improving robustness to internal noise. The neuronal response power law distribution shows the trade-off between precision and robustness, and supports the model's biological plausibility.

Supplementary Part

"FIRING RATES"

1. Effect of LGN: The inclusion of LGN preprocessing leads to a **concentration and reduction** in the firing rate distributions, particularly in excitatory populations. This suggests that LGN may **help regulate and stabilize the firing rates, reducing variability in neuronal activity**.
2. Impact of Neuronal Population Size: An increase in the number of neurons in the model results in **lower and more compact firing rate distributions**, especially for excitatory cells. This could be due to more complex connectivity patterns and increased inhibitory feedback in larger networks, which might suppress excitation and maintain balance.
3. Stabilizing Influence: **Both LGN and increased neuronal population size appear to play a role in stabilizing the network's activity**, which could be crucial for maintaining the model's performance and adapting to different visual tasks.

In summary, LGN preprocessing and larger network sizes lead to more regulated firing patterns across neuron populations, which may enhance the model's ability to learn.

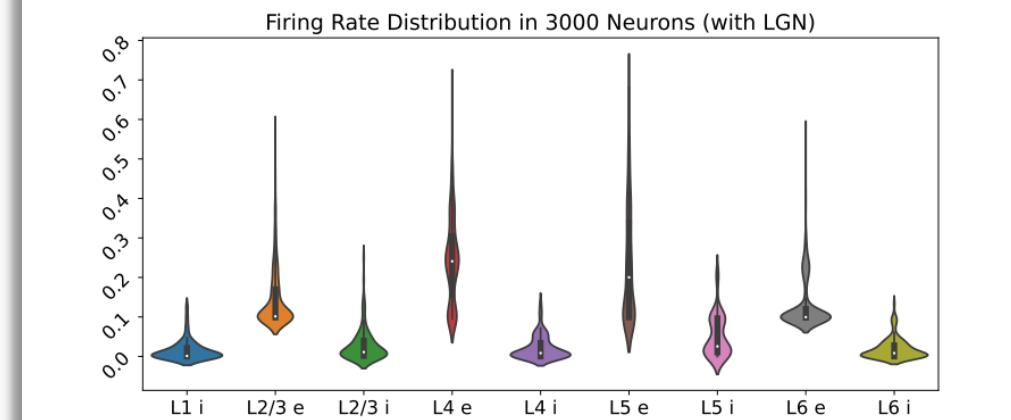


Figure 4.5 Firing rate distribution of all excitatory and inhibitory neuron classes in different layers (3k neurons, with LGN)

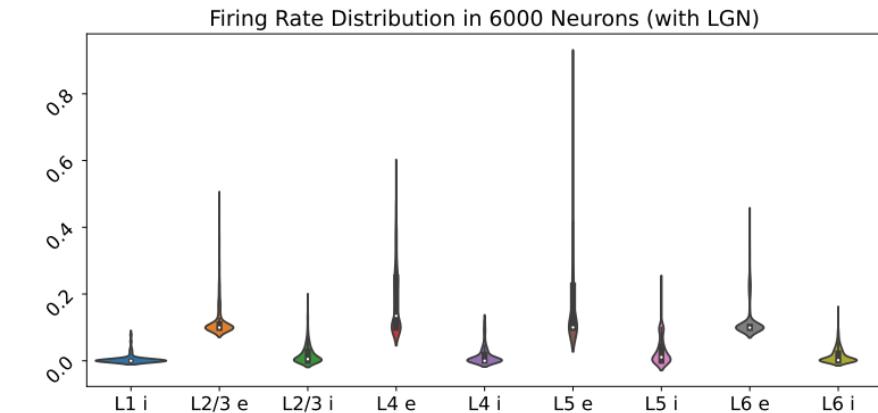


Figure 4.6 Firing rate distribution of all excitatory and inhibitory neuron classes in different layers (6k neurons, with LGN)

Supplementary Part

"WEIGHT MEAN DISTRIBUTION OF NEURONAL CLASS PAIRS"

1. LGN Impact: Without LGN, the weight distributions after training show a balanced distribution of positive and negative weights, capturing both excitatory and inhibitory connections. With LGN, there is a more skewed distribution **towards positive weights**, especially for connections involving the LGN, suggesting that it **enhances excitatory connections and facilitates the propagation of visual information**.

2. Network Size: The weight distributions for the 6,000-neuron network are **more fine-grained and have a smaller range** compared to the 3,000-neuron network, suggesting that a larger network size allows for **more precise weight adjustments and the capture of more intricate features**.

In summary, this part demonstrates how the V1 model's synaptic weights evolve through training, the influence of LGN on these weights, and how network size affects the precision of weight tuning.

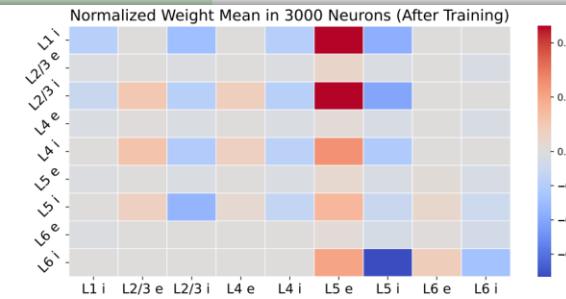


Figure E.2 Synaptic connectivity probability matrix for neuronal classes (3k neurons, after training)

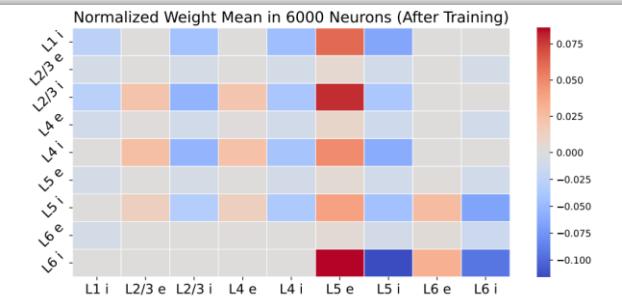


Figure E.4 Synaptic connectivity probability matrix for neuronal classes (6k neurons, after training)

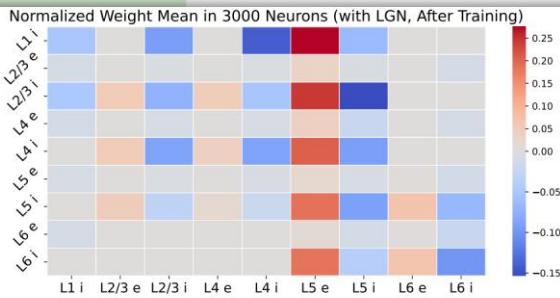


Figure E.6 Synaptic connectivity probability matrix for neuronal classes (3k neurons, with LGN, after training)

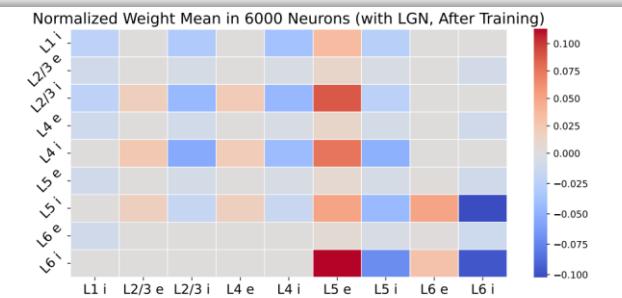


Figure E.8 Synaptic connectivity probability matrix for neuronal classes (6k neurons, with LGN, after training)

Supplementary Part

"SIMPLE AND COMPLEX CELLS"

1. Model Self-Organization: The V1 model developed in the thesis does not explicitly designate neurons as simple or complex. Instead, it is hypothesized that these properties would **emerge naturally after training due to the model's biological plausibility and large network size.**
2. Role of LGN: The incorporation of LGN filters in the model provides biologically realistic input preprocessing. However, it does not predetermine the existence of simple cells. **The emergence of both cell types is a result of the learning process** and the network's self-organization based on visual tasks and input statistics.
3. Future Analysis: the future work could involve analyzing the trained network to identify and characterize emergent simple and complex cells. This would involve examining the response properties of individual neurons to various visual stimuli and comparing them to the established properties of simple and complex cells.
4. Implications: The presence of both simple and complex cells in the trained network would demonstrate the model's ability to self-organize and develop key functional specializations without explicit architectural constraints.



Thank You!