Biologically constrained deep neural networks to parse visual computations in the primary visual cortex

Matthew Jacobsen[1], Jacob L Yates[3], Bevil R. Conway[2*], Daniel A. Butts[1*]

[1]Program in Neuroscience and Cognitive Science, University of Maryland, [2]Laboratory of Sensorimotor Research, National Eye Institute, National Institutes of Health,
[3]Herbert Wertheim School of Optometry and Vision Science, UC Berkeley, *co-PIs

Laboratory of Sensorimotor Research
National Eye Institute (NIH)

UNIVERSITY OF MARYLAND
PROGRAM IN NEUROSCIENCE & COGNITIVE SCIENCE
neurotheory.umd.edu

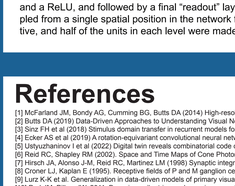
Introduction

Our understanding of how computations are implemented by visual circuits is still limited by our ability to characterize the complex nonlinearities present in the system. Deep convolutional neural networks (**CNNs**) have been able to capture the nonlinear computations represented in neural responses in complex stimulus contexts, but thus far there has been no clear way to relate their ability to predict neural responses to an understanding of the function of neurons they successfully model.

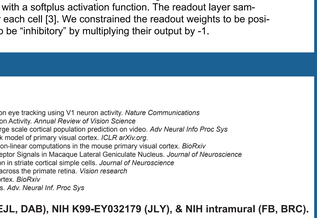
How can we use CNNs to understand neural function?

Here, we exploit known facts of the visual system to constrain build an "interpretable" CNN, and present strategies to characterize the complex computations performed over the population of recorded neurons in primary visual cortex (**V1**).

The visual pathway



Anatomically/physiologically constrained Convolutional Neural Network (CNN)



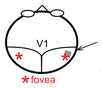
Methods

Electrophysiology Recordings were made in primary visual cortex (V1) from an awake macaque using a 96-electrode Utah array embedded in foveal V1, while the monkey passively fixated over 4-sec trials for a juice reward, as previously described [1]. Our CNNs were fit using data from four experiments, which yielded 598 units (with 306 single units).

Stimuli We presented color-cloud stimuli, which is spatiotemporal white noise updated at 60 Hz (with a 120 Hz monitor refresh), with each frame band-passed in the range of 6-30 cycles per degree to optimally drive foveal V1 responses. The stimulus was constructed in DKL space, with an uncorrelated stimulus frame in luminance, L-M, and S-dimensions, but here we only considered the luminance dimension, which primarily drove V1 responses. **Pixel size = 1 arcmin ~ 1 cone.**

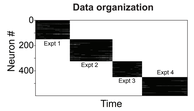
Model-based eye tracking Although the animal was fixating, small shifts in eye position due to fixational eye movements required shifting the stimulus fed into the model. We inferred these eye position shifts using a model-based eye tracking procedure based on neural activity, as previously described [1].

Animal passively viewed color cloud stimuli while we recorded from foveal V1 using via a chronically implanted Utah array.



Utah array
96 contacts,10x10,
400 μm spacing

Data organization



Responses from V1 recorded in day-long experiments. Experiments combined together to fit core model with large numbers of cells.

Maximum a posteriori (MAP) estimation of CNN parameters [2]:

Convolutional neural networks (CNNs) of various configurations were fit using using PyTorch. Parameters were fit using stochastic gradient descent (the AdamW optimizer to maximize the regularization-penalized population Poisson log-likelihood (per spike), given by

$$r_{pop}^{*} = \sum_i \frac{1}{N_{spk}^{(i)}} \sum_t d_i(t) \left[R_{obs}^{(i)}(t) \log_2 r_i(t) - r_i(t) \right] \quad - (\text{regularization penalties})$$

where $r_i(t)$ is the model predicted firing rate, and $d_i(t)$= 1 for all time points where there is recorded data for neuron i, and zero otherwise. Our network consisted of 4 convolutional layers consisting of filters, batch-norm, and a ReLU, and followed by a final "readout" layer with a softplus activation function. The readout layer sampled from a single spatial position in the network for each cell [3]. We constrained the readout weights to be positive, and half of the units in each level were made to be "inhibitory" by multiplying their output by -1.

References

[1] McFarland JM, Bondy AG, Cumming BG, Butts DA (2014) High-resolution eye tracking using V1 neuron activity. *Nature Communications*
[2] Butts DA (2019) Data-Driven Approaches to Understanding Visual Neuron Activity. *Annual Review of Vision Science*
[3] Sinz FH et al (2018) Stimulus domain transfer in recurrent models for large scale cortical population prediction on video. *Adv Neural Info Proc Sys*
[4] Ecker AS et al (2019) A rotation-equivariant convolutional neural network model of primary visual cortex. *ICLR arXiv:org*.
[5] Ustyuzhaninov I et al (2022) Digital twin reveals combinatorial code of non-linear computations in the mouse primary visual cortex. *BioRxiv*
[6] Reid RC, Shapley RM (2002). Space and Time Maps of Cone Photoreceptor Signals in Macaque Lateral Geniculate Nucleus. *Journal of Neuroscience*
[7] Hirsch JA, Alonso J-M, Reid RC, Martinez LM (1998) Synaptic integration in striate cortical simple cells. *Journal of Neuroscience*
[8] Croner LJ, Kaplan E (1995). Receptive fields of P and M ganglion cells across the primate retina. *Vision research*
[9] Lurz K-K et al. Generalization in data-driven models of primary visual cortex. *BioRxiv*
[10] Park IM, Pillow JW. 2011. Bayesian spike-triggered covariance analysis. *Adv. Neural Inf. Proc Sys*

This work supported by NSF IIS-2113197 (FB, EJL, DAB), NIH K99-EY032179 (JLY), & NIH intramural (FB, BRC).

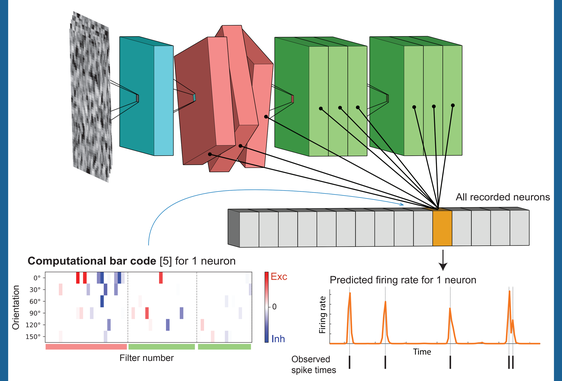
Functionally-inspired model architecure

Model architecture inspired by the functional anatomy of the early visual system:

1. First layer bottleneck inspired by retina/LGN
2. Spatial convolutions to capture similarity in computation across position
3. Orientation convolutions after the first layer to capture the similarity in
4. computation across orientations within V1
5. Units in the network obey Dales Rule: only excitatory or inhibitory
6. "Computational scaffold" to model neurons recorded across cortical layer

Network operates as follows:

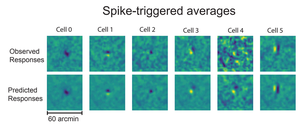
1. Stimulus is spatially convolved over by bottlenecked initial layer.
2. Output of this processed by filters that are rotated with 6 different orientations (30 degree increments) [see 4]
3. Space and orientation dimensions are preserved in each subsequent layer (3-d convolutions)



4. The prediction for each neuron response is readout at a given spatial position [3], sampling sparsely from particular locations within the three V1 layers.

5. Model is fit to predict the observed spike times of all recorded neurons across experiments.

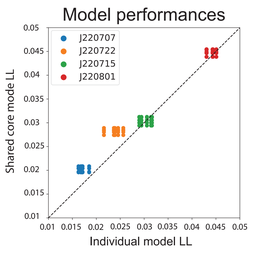
Model predicts neuron's receptive field well



Computational core across multiple datasets

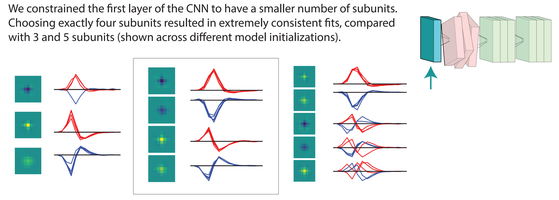
We compared performance for the CNN fit to 4 experiments at once, compared with to the same model fit on each experiment separately. We show the outcomes for 5 initializations of each model (single vs. shared).

Fitting multiple experiments at once improves performance for each experiment, suggesting the CNN learns common V1 computations that generalize across experiments [9].

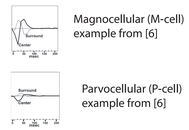


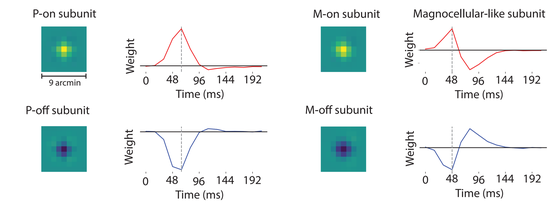
Model infers spatiotemporal properties of magno- and parvocellular LGN pathways

We constrained the first layer of the CNN to have a smaller number of subunits. Choosing exactly four subunits resulted in extremely consistent fits, compared with 3 and 5 subunits (shown across different model initializations).



The resulting filters were not oriented, and resembled the spatiotemporal properties of parvocellular (monophasic and smaller) and magnocellular (biphasic and larger) LGN neurons [8].

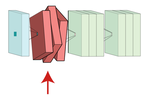


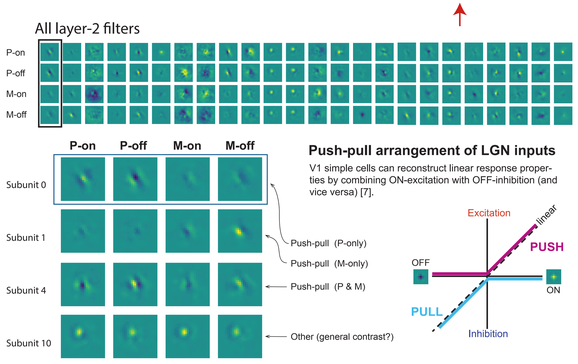


The resemblance of the input layer of our CNN to known LGN pathways facilitates interpretability in the next layer of the CNN.

Spatial filters in second layer have RFs resembling those found in V1

Subunits in the second CNN layer integrated spatially over the 4-subunit types from the first layer. This resulted in 4 spatial maps for each layer-2 subunit, specifying how it combines ON and OFF, and M- and P-like, inputs.





Push-pull arrangement of LGN inputs

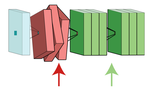
V1 simple cells can reconstruct linear response properties by combining ON-excitation with OFF-inhibition (and vice versa) [7].

Conclusions

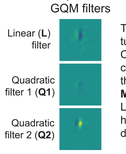
1. Fitting a general "core" model using multiple experiments improves performance by learning generalizable computations shared across neurons.
2. Biological constraints yield CNN internal units with classical properties of LGN inputs to V1 (M- and P-cells)
3. Biologically constrained CNNs also derive Push-Pull combinations of LGN inputs onto V1 input layers
4. Computational barcodes provide a means to characterize more complicated processing across V1 neurons

Computational bar codes as a means to characterize neural function at the population level

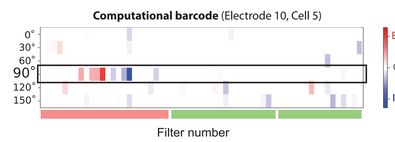
We represent each neuron's computational barcode across orientations. Using a form of sparseness regularization (similar to L1), neurons typically concentrate their filter connectivity at orientations that best matches their preferred orientation.



Example neuron
GQM filters



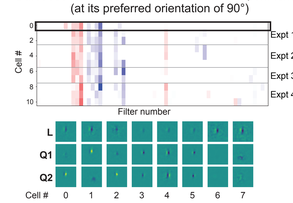
To determine orientation tuning independent of the CNN (and not requiring the cell to be linear), we use the **G**eneralized **Q**uadratic **M**odel (GQM) [10]: an LNLN cascade model that has one linear and two quadratic components.



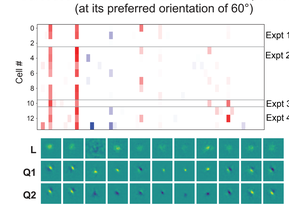
Cell barcode orientation preference matches the orientation identified by its receptive field.

Validation: functional similarities expected for neurons recorded on the same chronically implanted electrode (both within a given experiment, and over multiple days)

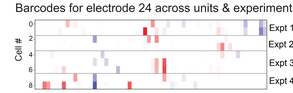
Barcodes for electrode 10 across units & experiments (at its preferred orientation of 90°)



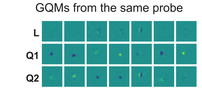
Barcodes for electrode 30 across units & experiments (at its preferred orientation of 60°)



Barcodes for electrode 24 across units & experiments



GQMs from the same probe



Not all electodes have similar units on it.

A metric for CNN interpretability based on barcode similarity?

Here we used a "barcode" similarity metric based on orientation-permuated cosine similarity.

An interpretable CNN will have barcodes:
=> that are different enough to capture large range of function
=> but similar enough to identify neurons as "similar" that have similar functional properties