RESEARCH ARTICLE

Learning divisive normalization in primary visual cortex

Max F. Burg[1,2,6]*, **Santiago A. Cadena**[1,2,3], **George H. Denfield**[3,4], **Edgar Y. Walker**[3,4], **Andreas S. Tolias**[2,3,4,5☯], **Matthias Bethge**[1,2,3☯], **Alexander S. Ecker**[6,7☯]

1 Institute for Theoretical Physics and Centre for Integrative Neuroscience, University of Tübingen, Tübingen, Germany, **2** Bernstein Center for Computational Neuroscience, Tübingen, Germany, **3** Center for Neuroscience and Artificial Intelligence, Baylor College of Medicine, Houston, Texas, United States of America, **4** Department of Neuroscience, Baylor College of Medicine, Houston, Texas, United States of America, **5** Department of Electrical and Computer Engineering, Rice University, Houston, Texas, United States of America, **6** Institute of Computer Science and Campus Institute Data Science, University of Göttingen, Göttingen, Germany, **7** Max Planck Institute for Dynamics and Self-Organization, Göttingen, Germany

☯ These authors contributed equally to this work.
* max.burg@bethgelab.org

Citation: Burg MF, Cadena SA, Denfield GH, Walker EY, Tolias AS, Bethge M, et al. (2021) Learning divisive normalization in primary visual cortex. PLoS Comput Biol 17(6): e1009028. https://doi.org/10.1371/journal.pcbi.1009028

Editor: Blake A. Richards, McGill University, CANADA

Received: July 24, 2020

Accepted: April 30, 2021

Published: June 7, 2021

Data Availability Statement: The data analyzed in this study is available in a GIN repository at https://doid.gin.g-node.org/2e31e304e03d6357c98ac735a1fe5788/ (DOI: 10.12751/g-node.2e31e3). Code is available from https://eckerlab.org/code/burg2021_learning_divisive_normalization.

Funding: The research was supported by the German Federal Ministry of Education and Research (BMBF) via the Competence Center for Machine Learning (FKZ 01IS18039A) [https://tuebingen.ai]; the Deutsche

Abstract

Divisive normalization (DN) is a prominent computational building block in the brain that has been proposed as a canonical cortical operation. Numerous experimental studies have verified its importance for capturing nonlinear neural response properties to simple, artificial stimuli, and computational studies suggest that DN is also an important component for processing natural stimuli. However, we lack quantitative models of DN that are directly informed by measurements of spiking responses in the brain and applicable to arbitrary stimuli. Here, we propose a DN model that is applicable to arbitrary input images. We test its ability to predict how neurons in macaque primary visual cortex (V1) respond to natural images, with a focus on nonlinear response properties within the classical receptive field. Our model consists of one layer of subunits followed by learned orientation-specific DN. It outperforms linear-nonlinear and wavelet-based feature representations and makes a significant step towards the performance of state-of-the-art convolutional neural network (CNN) models. Unlike deep CNNs, our compact DN model offers a direct interpretation of the nature of normalization. By inspecting the learned normalization pool of our model, we gained insights into a long-standing question about the tuning properties of DN that update the current textbook description: we found that within the receptive field oriented features were normalized preferentially by features with similar orientation rather than non-specifically as currently assumed.

Author summary

Divisive normalization (DN) is a computational building block throughout sensory processing in the brain. We currently lack an understanding of what role this normalization mechanism plays when processing complex stimuli like natural images. Here, we use

Forschungsgemeinschaft (DFG, German Research Foundation) via grant EC 479/1-1 (A.S.E.) [http://www.dfg.de/], the Collaborative Research Centers SFB 1233 (Robust Vision, project number 276693517) [https://uni-tuebingen.de/en/research/core-research/collaborative-research-centers/crc-1233/] and SFB 1456 (Mathematics of Experiment, project number 432680300) [https://www.uni-goettingen.de/en/628179.html], and the Cluster of Excellence "Machine Learning – New Perspectives for Science" (EXC 2064/1, project number 390727645) [http://www.ml-in-science.uni-tuebingen.de/]; the National Eye Institute of the National Institutes of Health under Award Numbers R01EY026927 (A.S.T.) and DP1 EY023176 (A.S.T.) [https://nei.nih.gov/], and NIH-Pioneer award DP1-OD008301 (A.S.T) [https://commonfund.nih.gov/pioneer/]. This research was also supported by NEI/NIH Core Grant for Vision Research (EY-002520-37) [https://nei.nih.gov/], NEI training grant T32EY00700140 (G.H.D) and F30EY025510 (E.Y.W.) [https://nei.nih.gov/], DARPA grant N66001-17-C-4002 [https://www.darpa.mil], and the Intelligence Advanced Research Projects Activity (IARPA) via Department of Interior/Interior Business Center (DoI/IBC) contract number D16PC00003 [https://www.iarpa.gov]. The funders had no role in study design, data collection and analysis, decision to publish, or preparation of the manuscript.

Competing interests: The authors have declared that no competing interests exist.

modern machine learning methods to build a general DN model that is directly informed by data from primary visual cortex (V1). Contrary to high-predictive deep learning models, our DN-based model's parameters offer a straightforward interpretation of the nature of normalization. Within the receptive field, we found that neurons responding strongly to a specific orientation are preferentially normalized by other neurons that are highly active for similar orientations, rather than being normalized by all neurons as it is currently assumed by textbook models.

Introduction

A crucial step towards understanding the visual system is to build models that predict neural responses to arbitrary stimuli with high accuracy [1]. The classical standard models of primary visual cortex (V1) are based on linear-nonlinear models [2], energy models [3] and subunit (LN-LN) models [4–9]. Fueled by advances in machine learning technology, recent studies have shown that multi-layer convolutional neural networks (CNNs) can significantly improve the prediction of neural responses to complex images and videos at several stages of the visual pathway, outperforming classical models [10–17]. The current state-of-the-art data-driven model of single-unit activity in monkey V1 is a three-layer CNN [14]. However, it is challenging to gain insights into V1 function from the features produced by deeper layers of such models. In particular, we do not have first principles explaining the kind of nonlinearities approximated in successive layers of CNNs, or if these nonlinearities can be described in a compact way in the first place.

A promising candidate to facilitate a more succinct description of V1 neurons is to replace the computations from the CNN's deeper layers by a divisive normalization nonlinearity [18] that is easier to understand. Divisive normalization has been proposed as a canonical neural computation present throughout the visual pathway [19] because it (1) explains a wide variety of neurophysiological phenomena using simple stimuli [19, 20] and (2) can be derived from first principles of redundancy reduction [21, 22]. The significance of DN is also supported by its recent success in computer vision where it enables state-of-the-art natural image compression with high perceptual quality [23].

Divisive normalization has been invoked to explain several nonlinear neurophysiological phenomena in V1, which can be classified into (1) phenomena that are restricted to the receptive field (RF) only or (2) phenomena that involve an interaction between receptive field and its surround. A prominent example for a mechanism restricted to the receptive field is cross-orientation inhibition [18, 24–29], whereas an example for an interaction of the RF with its spatially adjacent surround is surround suppression [30–34]. In our work, we focus on the first aspect: divisive normalization *within* the RF and cross-orientation inhibition. In this phenomenon, the response of a neuron to a driving grating stimulus in the RF is suppressed by superimposing a second grating also within the RF that does not elicit a response when presented alone: for instance, a grating with orientation orthogonal to the neuron's preferred orientation.

The basic idea of divisive normalization (Fig 1A) is that a neuron's driving input is normalized divisively by a weighted sum over nearby neurons' responses [18, 19]. While the general idea is simple, elegant and powerful, our current knowledge of DN is limited in two important ways: (1) DN has been studied mostly with simple, synthetic stimuli, but its implications under natural image stimulation are unclear. For example we do not know whether incorporating DN into system identification models predicting neural responses to natural stimuli improves

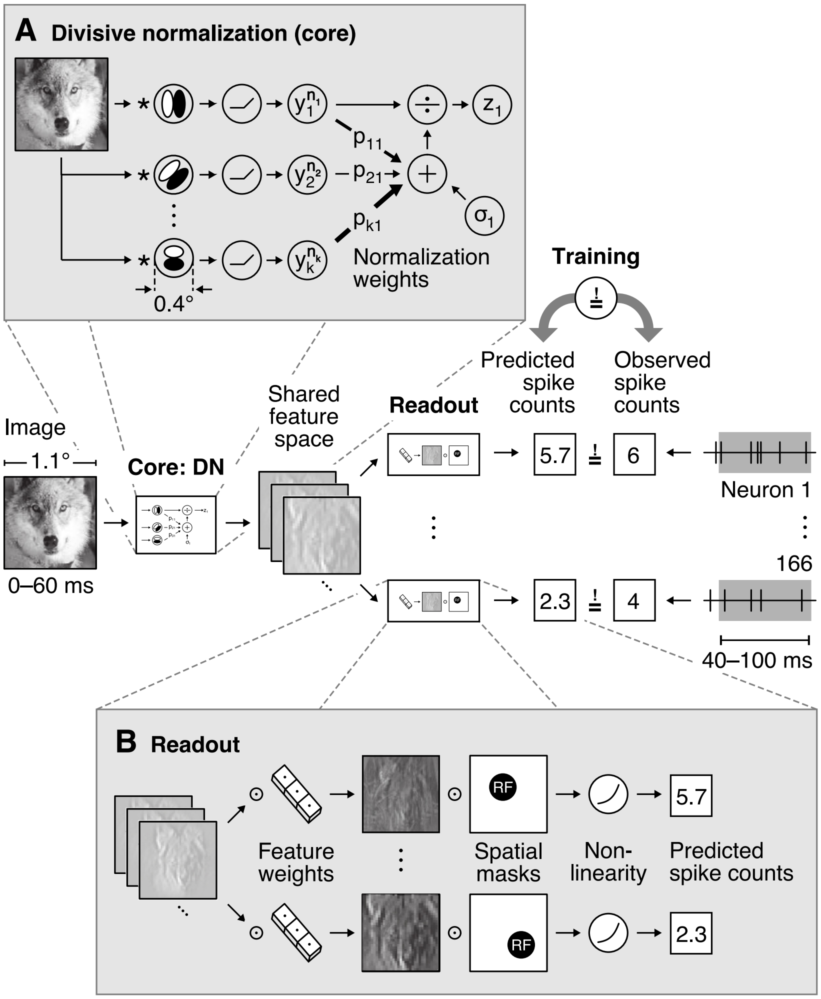

Fig 1. Overview of our divisive normalization (DN) model. The model takes as input an image covering 1.1° of visual field and predicts neurons' spike counts in response to this image (details in Fig 2). The model is split into two parts: a *core* that computes a shared nonlinear feature space and a *readout* that maps the shared feature space individually to each neuron's spike count. **A**. Divisive normalization mechanism (simplified). The visual input is convolved with 32 filters covering 0.4° of visual field and then rectified and exponentiated to produce an excitatory output. The output of each filter is then divided by a weighted sum of the excitatory outputs of all filters with normalization weights p_{kl} and a semi-saturation constant σ_l. In our general formulation, all weights and constants are learned from the data. **B**. Readout that maps the shared feature space to each neuron's spike count through an individual weighted sum over the entire shared feature space and a pointwise output nonlinearity. The readout weights are factorized into a feature vector—capturing the nonlinear feature(s) that a neuron computes—and a spatial mask—localizing each neuron's receptive field (RF).

https://doi.org/10.1371/journal.pcbi.1009028.g001

performance. (2) We currently do not know how response properties of neurons with overlapping receptive fields determine whether a neuron contributes to the normalization pool of another neuron and, if so, with which strength [27].

To explain normalization phenomena within the receptive field like cross-orientation inhibition, current models of divisive normalization assume that all nearby neurons with diverse orientation tuning preferences and with similar receptive field locations contribute equally to the normalization pool [18, 27, 29]. However, some earlier experimental studies suggest that this assumption may not be correct for some neurons [24, 26], and normative models of normalization predict that the magnitude with which a given neuron contributes to another neuron's normalization should depend on the relationship of their response properties [21].

In this paper, we address the two main questions raised above: (1) How well does a data-driven model with divisive normalization predict V1 responses to natural images, compared to classical subunit models and deep convolutional neural networks, and (2) what stimulus features are learned for normalizing a V1 neuron's response in relation to its favoured feature selectivity within the receptive field?

We focused on effects restricted to the receptive field and on models that only account for local normalization interactions between neurons with overlapping receptive fields. We developed an end-to-end trainable model predicting V1 spike counts from natural stimuli, replacing the CNNs deeper layers by the canonical neural DN computation [19] and learning the filters of all neurons as well as their normalization weights directly from the data. We also explored how our DN model could be extended in a way that might enable it to capture surround interactions from outside of the RF, however, our control experiments demonstrate that our results are very unlikely to include the extra-classical surround.

We fitted our DN model to monkey V1 responses to natural images and found that it outperforms linear-nonlinear and subunit models, suggesting that DN mechanisms are useful for capturing nonlinear computations unlocked by natural stimuli beyond those evoked by simpler gratings. The power of our model was further strengthened as it captured cross-orientation inhibition within the receptive field, brought to light via *in silico* experiments. In contrast to the three-layer CNN, which also reproduced this phenomenon, the single divisive layer of the DN model provides direct interpretations of the normalization pool. These revealed that oriented features were preferentially normalized by other features with similar orientation, which quantitatively improved predictive performance over a model with nonspecific normalization. However, the three-layer CNN outperformed our DN models, suggesting that there are additional non-linear mechanisms beyond the ones captured by our DN model (e. g. phase invariance and divisive effects), that are required to achieve optimal performance in modeling V1 responses. Thus, our work advances our understanding of V1 function, highlighting the importance of orientation-specific gain control to predict responses to natural images, and shows how embedding theories of neural processing into the architectures of generic data-driven models can improve our understanding of sensory processing in the brain.

Results

Learning divisive normalization

The basic idea of divisive normalization (Fig 1A) is that the response of neuron l

$$z_l(x) = \frac{y_l^{n_l}(x)}{\sigma_l^{n_l} + \sum_{k \in \mathcal{K}} p_{kl} \cdot y_k^{n_k}(x)} \tag{1}$$

is given by its driving input activity $y_l(x)$ exponentiated by n_l and divisively normalized by a weighted sum over nearby neurons' exponentiated responses $y_k^{n_k}(x)$ [18, 19], where x represents the stimulus, σ_l is a semi-saturation constant, and all quantities are non-negative. Here, the set of normalizing neurons $\mathcal{K}$ and the normalization weights p_{kl} define which neurons k contribute to the normalization pool of neuron l and with what strength, respectively.

While this formulation is straightforward to write down, it is challenging to build quantitative models based on it that are applicable to arbitrary inputs. The denominator depends on a potentially large population of neurons—which is unknown in general—and the structure of the normalization weights has been studied only using very restricted sets of simple stimuli such as oriented gratings and bars. Previous modeling work on divisive normalization has therefore made specific assumptions about the filter properties of the underlying neuronal population and either modeled only a closed set of stimuli such as gratings of different orientation [18, 27–29, 35] or evaluated models only qualitatively [21, 36, 37].

We developed a general, image-computable predictive model of divisive normalization following Eq (1), which is applicable to arbitrary images and whose parameters are learned by optimizing the accuracy of the model in predicting the spiking activity of a large number of neurons in response to natural images (see Fig 1). Our model builds on a recent innovation in predictive modeling [12, 14, 38–40], jointly modeling all recorded neurons instead of learning a predictive model for each neuron individually. Because many neurons perform similar computations—up to shifts in receptive field location—jointly modeling them makes more efficient use of the data and we can learn more complex models. The basic idea is to split the model into two parts (Fig 1): (A) a *core* that transforms the input image into nonlinear features shared among all neurons, and (B) a *readout* that linearly combines the features to produce a prediction of each neuron's response.

We use a convolutional network for the core, whose architecture lends itself very well to model divisive normalization. By construction, we have a model that contains all filters necessary to account for the recorded neurons' responses. All of these filter responses are automatically extracted at each location, providing a good approximation of the underlying population of neurons in the brain although it is only sparsely sampled during the experiment. As a consequence, we can optimize the pool of neurons providing normalizing inputs and their corresponding weights p_{kl} (Eq 1) to account for the neural responses. In this work we mostly focus on normalization effects from neurons with overlapping receptive fields, rather than investigating interactions with the RFs' surround.

In summary, our model's core (Fig 1A) consists of a set of convolutional filters w_l and biases o_l (we use 32) that provide the driving inputs y_l from an image x at all locations in the neurons' RFs,

$$y_l = \max(0,\ w_l * x + o_l)\ . \tag{2}$$

As a specific feature is extracted by its according convolutional filter at each spatial location of the image, the driving responses y_l form a map of responses for the according features, hence named feature map. This operation is followed by a DN stage (Eq 1) in which all operations are performed element-wise across space. This core is shared among all neurons and converts the image into a set of feature maps z_l containing information from the neurons' RFs. We mapped those normalized feature maps into response predictions $\hat{r}$ by a linear readout step (Fig 1B) that picks the relevant features by readout weights b and spatial locations by readout weights a for each neuron i, followed by a pointwise output nonlinearity ϕ,

$$\hat{r}_i = \phi_i\Big(\sum_{u,v,l}(a_{uv,i}\, b_{l,i})\, z_{uvl}\Big)\ , \tag{3}$$

where u, v index space (see Methods for additional details). To ensure that the readout does not model any inhibitory or complex interactions, we constrained its weights a, b to be non-negative and added a sparsity prior. The non-negativity ensures that activations can only add, preventing the readout stage from accounting for any suppressive effects. The readout can, however, account for response invariances such as phase invariance of complex cells; see Methods for an in-depth explanation. While our model reflects the general formulation of divisive normalization, mathematically it is not identical to the classical formulation of DN due to the additional linear-nonlinear readout after the model's core.

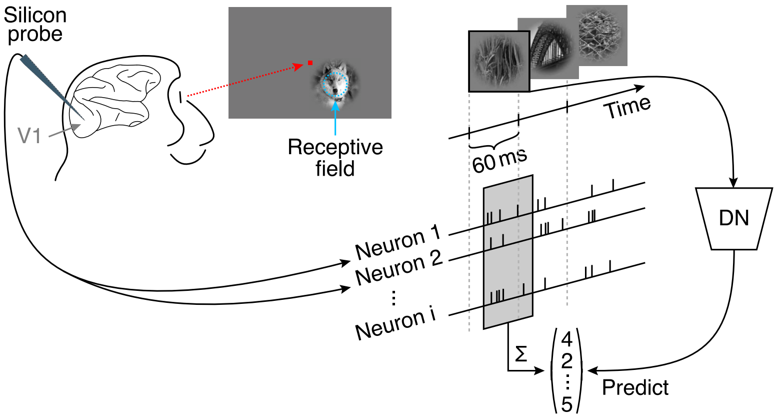

Fig 2. Experimental paradigm from Cadena and colleagues [14]. Natural images were flashed to a monkey covering 2° of their visual angle, and located at the center of the multi-unit receptive field. Multiple neurons were isolated from recordings with silicon probes inserted into V1 [41]. Natural images were shown in a fast sequence without blanks, each presented for 60 ms. Spike counts from all isolated neurons corresponding to each image were extracted from a window 40 ms after the image onset lasting 60 ms.

https://doi.org/10.1371/journal.pcbi.1009028.g002

DN model outperforms subunit model, half-way closing the gap to state-of-the-art performance

We fitted the model described above to a dataset of 166 neurons recorded in V1 of two awake, fixating monkeys, who viewed a fast sequence of localized natural images and textures (Fig 2; data from Cadena and colleagues [14]). The stimuli were placed at the neurons' receptive fields and primarily stimulated the center (i. e. the RF) as they covered one degree of visual field, quickly fading off reaching zero contrast at two degrees (see Discussion and Methods for additional details). Images were shown for 60 ms each, without blanks in between. Single unit activity was recorded with laminar silicon probes sampling from all cortical layers. We fit the model jointly to the responses of all neurons. As neurons were recorded in 17 recording sessions, the dataset sampled a diverse range of preferred orientations. The objective function during training was to minimize the difference between the model's prediction and the observed spike counts of the neurons in a time window 40–100 ms after image onset (to account for response latency).

To evaluate model performance, we estimated the fraction of explainable variance explained (FEV), which quantifies the fraction of the stimulus-driven response variance that is accounted for by the model, and ignores unexplainable trial-to-trial variability in the response of the neurons (see Methods). A perfect model would reach a FEV of 100%. Since model performance depends on the parameters' initial values before optimization, we initialized all weights randomly before model training and repeated this procedure multiple times. To ensure that the reported performance is not a spurious characteristic of one particular model, we picked the best ten models of a model class (assessed in terms of validation set accuracy) and report the mean performance together with its 95% confidence interval. We perform this analysis for each of the four model classes.

Subunit models are an established approach to model primary visual cortex responses [4–9]. In addition to capturing a fair portion of the explainable variance [14], their computations are easily understandable because they are rather shallow models consisting of a first stage of rectified linear filtering followed by a static nonlinearity, and then a pooling stage with a subsequent output nonlinearity. Although the final output nonlinearity is capable to learn various pointwise functions like sigmoids, in most cases (across all model types) it leads to only slight expansive deviations of the identity function. Structurally, the subunit model is the same as our DN model, but without the normalization stage. Therefore, we considered a convolutional subunit model as a strong baseline for our model. This subunit model accounted for (46.6 ± 0.2)% FEV (mean and 95% confidence interval over the ten best models selected in terms of validation set accuracy). In comparison, a regularized linear nonlinear Poisson model (LNP) only accounted for 16.3% FEV on the same dataset [14] due to its inability to model complex cells.

As recent developments in machine learning technology have allowed us to improve predictive performance, we used the current best data-driven model as a gold standard. This model is a convolutional neural network with three convolutional layers and a linear-nonlinear readout [14]. To enable fair comparison of this model to all other models in this study, we optimized the CNN comparable to our DN model, reaching (52.3 ± 0.1)% FEV (mean and 95% confidence interval over the ten best models selected in terms of validation set accuracy; outperforming previously reported performance of 49.8% FEV [14]). However, although this CNN model outperforms the simpler subunit model, it is of higher complexity requiring a higher number of parameters.

To evaluate how well our DN model accounts for the data, we placed its performance on a scale between 0% (baseline: subunit model) and 100% (gold standard: CNN). On that scale, our DN model achieved a score of (52 ± 3)% between the baseline and gold standard (Fig 3; (49.6 ± 0.1)% FEV; mean and 95% confidence interval over the ten best models selected in terms of validation set accuracy). The performance differences between all model pairs were statistically significant (pairwise Wilcoxon signed rank test on best models in terms of validation accuracy: $p < 0.024$, $N = 166$ neurons, family-wise error rate $\alpha = 0.05$ using Holm-Bonferroni correction to account for multiple comparisons). Notably, we achieved this performance gain by simply adding the trainable DN stage to the convolutional subunit model, which

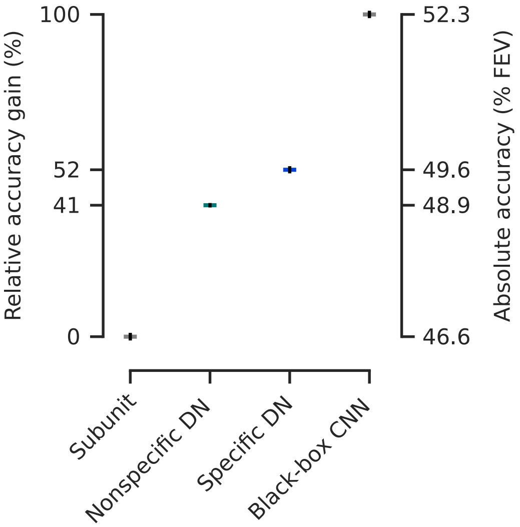

Fig 3. Performance comparison of our models fitted to the data from Cadena and colleagues [14] relative to the gap between the best shallow model—a subunit one layer convolutional neural network (CNN)—and the deeper data-driven state-of-the-art three-layer CNN [14]. Non-specific divisive normalization (DN) accounts for 41% of this gap, while specific DN improves it up to 52%. Absolute values in terms of percentage of explainable variance explained (FEV) on the right (mean over the ten best models selected in terms of validation set accuracy, see main text for details). Error-bars (black) indicate the standard error of the mean. Model performance is significantly different between each model class (pairwise Wilcoxon signed rank test on best models in terms of validation accuracy: $p < 0.024$, $N = 166$ neurons, family-wise error rate $\alpha = 0.05$ using Holm-Bonferroni correction).

https://doi.org/10.1371/journal.pcbi.1009028.g003

quantifies the importance of divisive normalization as computational mechanism to predict neural responses in V1 under stimulation with complex, natural images. At the same time, the CNN outperformed the DN model, suggesting that further non-linear dependencies in addition to phase invariance (complex cells) and divisive normalization are required to reach optimal predictive performance.

Divisive normalization and CNN models learn cross-orientation inhibition

We wanted to test if our DN model captured non-linear interactions between the neurons inside the receptive field leading to cross-orientation inhibition, a phenomenon that was explained by DN within the RF before [18, 24–29]. As the gold standard three-layer convolutional neural network explains more variance than the divisive normalization model, we asked if the CNN captured cross-orientation inhibition, too. To assess if our models learned cross-orientation inhibition, we performed the experiments done before by experimentalists *in silico*. Specifically, for each unit we determined the optimal Gabor maximizing our models' predictions. Subsequently, we combined the optimal Gabor by adding the same Gabor pattern with orthogonal orientation, obtaining a plaid stimulus (Fig 4A insets). We then presented the plaids to our models varying the contrasts of each Gabor component and measured the resulting tuning curve for all models *in silico*, averaging across the orthogonal Gabor's phase to get independent of any phase dependency. For the DN model and the CNN, we found cross-orientation inhibition in a number of cells: the response elicited by presenting the optimal Gabor was inhibited by increasing the contrast of the orthogonal Gabor component (Fig 4A). However, such behavior was not clearly visible for the subunit model.

To quantify the models' capability to learn cross-orientation inhibition, we defined a cross-orientation inhibition index measuring the percentage an orthogonal Gabor inhibits response when added to an optimal Gabor presented alone (Fig 4A left; see Methods for details). To determine how many cells are cross-orientation inhibited by at least 10%, we assess the highest cross-orientation inhibition index across all contrast combinations of the two component Gabors. To ensure we report a general property of all models, we quantified cross-orientation inhibition across the ten best models of each model-type (assessed in terms of performance on the validation set, only differing in the initialization of the model parameters before training). In the subunit model we found no significant cross-orientation inhibition (Fig 4B). Although the 3% FEV performance gain of the DN model over the subunit model might seem small, the phenomenological difference between those models was quite substantial, as in the DN model half of the cells showed cross-orientation inhibition. The gold-standard three-layer CNN captured cross-orientation inhibition in fewer cells, likely approximating divisive normalization through its deeper layers as its filters in the first convolutional layer are qualitatively similar to the DN model, showing Gabor-like and blob patterns. At the same time, the CNN seems to capture additional effects from the data that are not explained by DN in the RF center, leading to the CNN's higher performance.

Normalization within the receptive field is feature-specific

Having established that both the DN and the CNN models learned a phenomenon explained by divisive normalization within the RF, namely cross-orientation inhibition, we next asked which features and how strongly they contribute to the normalization pool. In the CNN model there is no explicit notion of a normalization pool that we could comprehensively analyze. On the other hand, our DN model offers a way to investigate the structure of the normalizing input within the receptive field, i. e. the image-averaged products $\langle p_{kl} \cdot y_k^{n_k}(x) \rangle_x$ in the denominator of Eq (1), to quantify the strength by which different features contribute to it (see

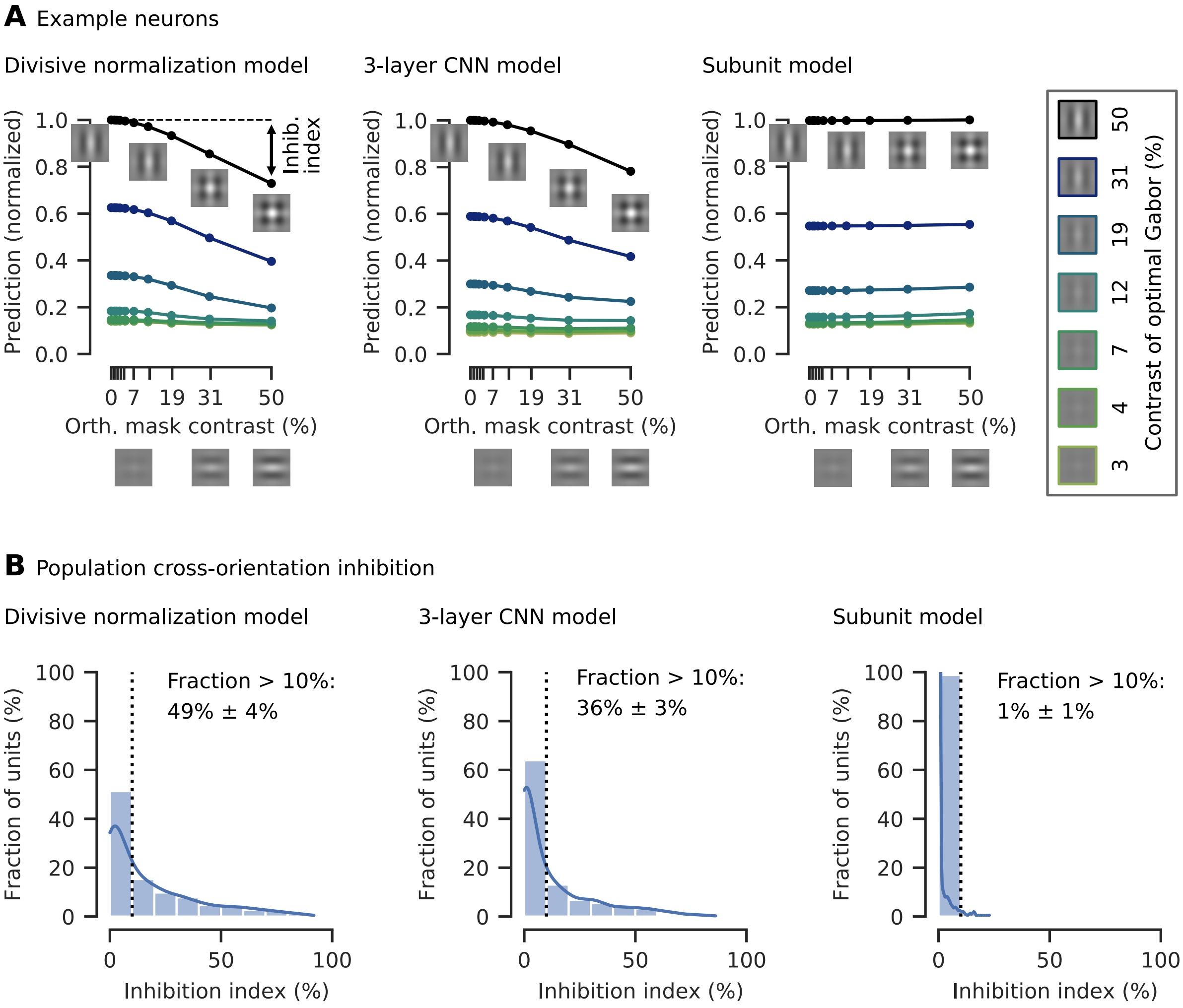

Fig 4. Cross-orientation inhibition was learned by our DN model and the three-layer CNN, but not significantly by the subunit model. **A**. Tuning curves for an example neuron of all three models and various contrast combinations of the optimal Gabor (box on the right, examples for contrasts of 0%, 1% and 2% not shown) and an orthogonal Gabor masking. As the contrast of the orthogonal mask increases, the model prediction (normalized by the maximum response) decreases. The cross-orientation inhibition (inhib.) index measures the percentage of response inhibition by adding the masking compared to the optimal Gabor presented alone, in this case approximately 20%. **A**. **insets**: Illustration of plaid stimuli, created by overlaying an optimally oriented Gabor with an orthogonal mask. **B**. Histograms of the cross-orientation inhibition indices accumulated across the best ten models (in terms of validation set accuracy) per model type, with kernel density estimate of the underlying distribution. The fraction of cells that show more than 10% cross-orientation inhibition is displayed right of the dotted line (mean and 95% confidence interval over the ten best models selected in terms of validation set accuracy). For the DN model, more cells show cross-orientation inhibition compared to the other models. The subunit model shows almost no cross-orientation inhibition.

https://doi.org/10.1371/journal.pcbi.1009028.g004

Methods for details). When visualizing the feed-forward features learned by the DN model we found that several exhibited clear orientation selectivity (i. e. features with an anisotropic structure; see Fig 5A). Thus, for these features we first wondered about the relationship between their preferred orientation and the orientation of the features by which they are preferentially normalized. To this end, we visualized the feature pair-wise normalizing input (Fig 5A) and found that oriented features are normalized preferentially by features with similar orientation preference, while orthogonal features seem to contribute less.

First, we coarsely quantified the difference with which these two groups contribute to normalization. We split the sum in Eq (1) into two parts and collected the normalizing input of

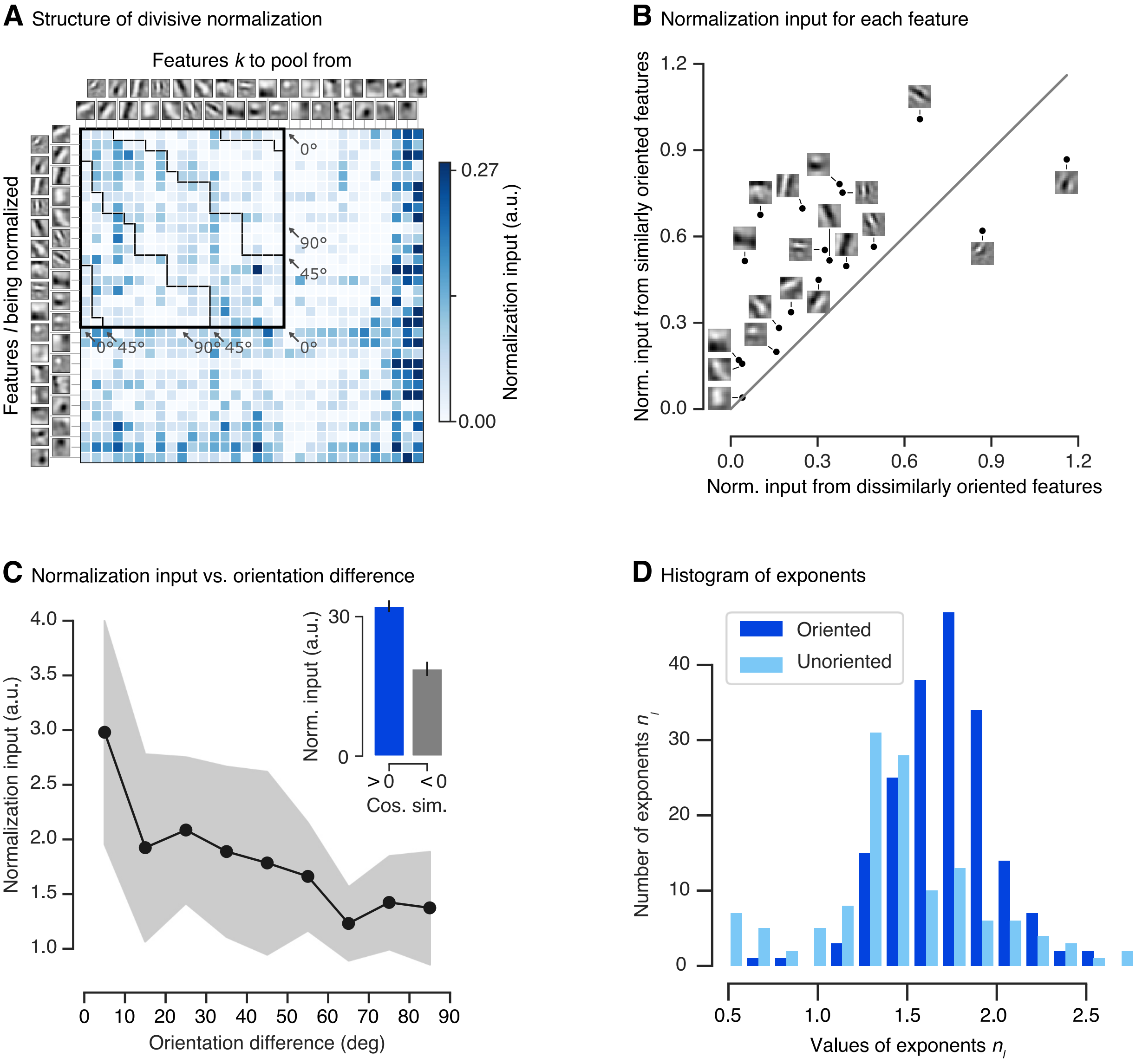

Fig 5. Structure of divisive normalization. A. The matrix shows the average strength of the normalizing inputs (products $\langle p_{kl}\, y_k^{n_k}(x)\rangle_x$ in denominator of Eq (1) averaged across images; see Methods) for each combination of filter response being normalized (rows) and filter response providing normalizing input (columns). Darker shades of blue indicate stronger normalization. Orientation-selective filters are grouped at the top, ordered by preferred orientation and marked by the black square. The dashed black lines within the square separate pairs of filters with similar ($< 45^{\circ}$) and dissimilar ($\leq 45^{\circ}$) orientations. Normalizing inputs are stronger for similarly tuned filters. Unoriented filters mainly accounting for orientation-unspecific contrast are sorted by total normalization input. Darkest blue color corresponds to the maximum normalization input for the group of oriented filters, higher normalization input values for the unoriented filters are clipped to that value. Data of the model with highest accuracy on the validation set is shown. **B**. Normalization input from similar orientations ($< 45^{\circ}$) compared to the normalization input from dissimilar orientations ($\geq 45^{\circ}$) for each oriented linear filter. Grey line: identity. Most features are normalized preferentially by the responses of filters with similar preferred orientations. Data of the model with highest accuracy on the validation set is shown. **C**. Normalization input, binned into orientation difference of 10°. Each bin was averaged over the top-10 models (assessed on the validation set). The shaded area depicts the standard deviation per bin. **C inset**. Normalization input (norm. input) vs. cosine similarity between linear filters (cos. sim.) averaged across the top-10 models (assessed on the validation set). A cosine similarity greater than zero corresponds to similar features. Error bars: standard error of the mean. **D**. Histogram of DN exponents (n_l in Eq 1) of the ten best performing models in terms of validation set accuracy. Darker/lighter color: exponents corresponding to driving inputs due to oriented/unoriented linear filters. Most values are larger than one, with a few exceptions mainly corresponding to unoriented filters.

https://doi.org/10.1371/journal.pcbi.1009028.g005

features with similar orientation as the driving feature ($< 45^\circ$) and that of features with dissimilar orientations ($\geq 45^\circ$). By analyzing the normalization of each oriented feature individually, we found that most oriented features are more strongly normalized by features with similar orientations (Fig 5B). To assess whether our qualitative observation above is a general property of the data or a spurious characteristic of the best model chosen, we repeated this analysis for the top-10 performing models (assessed in terms of accuracy on the validation set, only differing in the initialization of the model parameters before training) and observed a similar behavior. Summing up the normalizing input across the features, we found that, for all of these models, similar orientations contributed more strongly than dissimilar orientations. Taking the data of all top-10 models into account, we found that similarly oriented features contributed (1.5 ± 0.1)-fold more normalizing input than dissimilar features (mean and 95% confidence interval over the ten best models selected in terms of validation set accuracy; Wilcoxon signed rank test: $p < 0.006$, $N = 10$ models; Cohen's $d = 1.2$).

We further quantified the orientation-specific nature of normalization at a more fine-grained level. Instead of lumping orientation differences into two groups as before, we split up the normalizing inputs into nine bins of 10° width each and averaged those bins across the top-10 models. This analysis revealed that the strength of the normalizing inputs decreased as the difference in orientation increased (Fig 5C). Hence, the more similar a normalizing feature's orientation was to the feature to be normalized, the stronger was its contribution to normalization (linear regression analysis on all normalization input vs. orientation difference pairs of best model in terms of validation accuracy: $p < 10^{-6}$). In fact, features in the group most similar to the driving input contributed (2.3 ± 0.4)-fold more than those in the orthogonal group (mean and 95% confidence interval over the ten best models selected in terms of validation set accuracy; Wilcoxon signed rank test: $p < 0.006$, $N = 10$ models; Cohen's $d = 1.9$).

An immediate question that follows from our previous findings relates to the contribution of more general feature properties beyond orientation selectivity. Is the normalization input higher only for similar orientation, or is normalization input generally higher for similar filters, without pre-specifying a certain property like orientation? To answer that question, we split the input to normalization in terms of cosine similarity between all oriented and unoriented filters into a group of similar (cosine similarity > 0) and dissimilar (cosine similarity < 0) features. Summing up the normalizing input across the features and averaging across the best ten models, we found that in general similar features contributed (1.9 ± 0.5)-fold more normalizing input than dissimilar features (Fig 5C inset; mean and 95% confidence interval over the ten best models selected in terms of validation set accuracy; Wilcoxon signed rank test: $p < 0.006$, $N = 10$ models; Cohen's $d = 2.9$).

Analysing the exponents of our divisive normalization model, we found a mean value of 1.61 (averaged across the best ten models on the validation set) with most values being greater than one (Fig 5D), which previously has been connected to an intensified winner-take-all behavior of responses [19, 29]. In a few exceptions exponents learned by our model are smaller than one, mainly corresponding to driving inputs from unoriented filters.

Control: Nonspecific divisive normalization reduces accuracy. To determine how important orientation-specific normalization is, we performed a control experiment: For each feature l being normalized, we constrained all of its incoming normalization weights p_{kl} to be identical. This model is an instantiation of the previous models [18, 27, 29] assuming non-specific normalization from all features. Note that, since the feature orientations were non-uniformly distributed (Fig 5A), this model's normalization was weakly orientation tuned despite equal weights for all features (orientation difference $< 45^\circ$ contributed 1.1-fold stronger than $\geq 45^\circ$; Wilcoxon signed rank test; $p < 0.007$, $N = 10$ models; Cohen's $d = 0.3$). This model achieved a performance of (41 ± 2)% between the baseline and gold standard

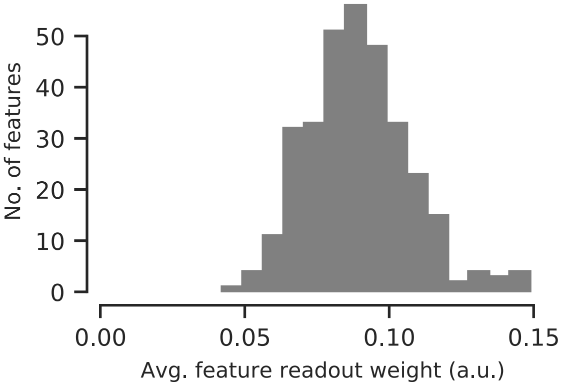

Fig 6. Histogram of feature readout weights of the ten best performing models in terms of validation set accuracy. For each model, feature weights are normalized across channels and averaged across individual neurons. All model's channels are used to predict neural activity.

https://doi.org/10.1371/journal.pcbi.1009028.g006

((48.9 ± 0.1)% FEV, mean and 95% confidence interval over the ten best models selected in terms of validation set accuracy). The unspecific DN model was included in the hypothesis test showing statistically significant performance differences between all model types (pairwise Wilcoxon signed rank test on best models in terms of validation accuracy: $p < 0.024$, $N = 166$ neurons, family-wise error rate $\alpha = 0.05$ using Holm-Bonferroni correction). While the unspecific control model does not match the performance of our more general DN model, it does outperform the subunit baseline. Thus, stronger orientation-specific normalization is necessary to achieve the full DN model performance, but a simpler form of uniform divisive normalization can account for a large fraction of the improvement over the subunit baseline.

Control: All channels contribute to our model's prediction. One potential caveat of our analyses so far is that we analyzed the orientation specificity of DN in terms of the convolutional feature maps in our model's core rather than the actual neurons we recorded. These features provide a much more compact view of the population of neurons, because they are invariant to the receptive field locations and the neural responses are simple linear combinations of those features. Moreover, the model's core offers a direct interpretation of its normalization weights. However, it is not clear a-priori whether all features are equally important for predicting the activity of the neurons in our population. Thus, considering convolutional features instead of actual neurons may lead to a skewed view of the population. To verify that this is not an issue, we quantified how much each feature contributed to the overall activity of all neurons by normalizing the feature readout weights across channels and averaging across neurons. The resulting distribution (Fig 6) containing these averaged feature readout weights for the best ten models had a coefficient of variation of 0.2. We therefore concluded that all features were read out by roughly the same number of neurons and hence were similarly important to predict neural activity. Thus, our interpretation of orientation-specific normalization is unlikely to be an artifact of analyzing the convolutional features rather than the actual neurons.

Control: Surround interactions are unlikely driving orientation-specific normalization in the RF center. We observed orientation-specific divisive normalization in the receptive field. Surround suppression is known to be orientation-specific as well [30–34], so a potential concern would be that some of the extra-classical surround of a unit's RF contributed to the results presented above. To rule out this possibility, we explicitly verified *in silico* that our model learning DN inside the RF learned no significant surround-suppression. For each

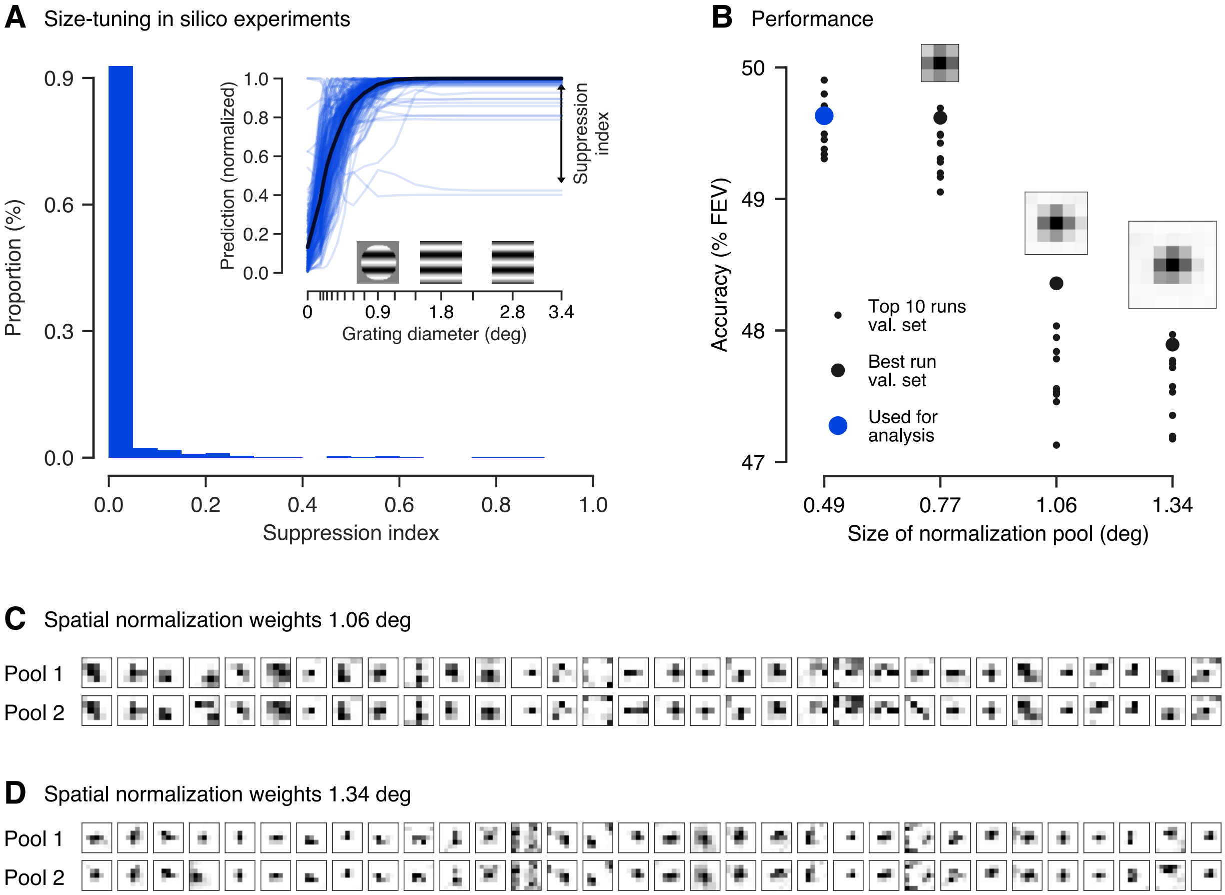

Fig 7. Size-tuning *in silico* experiments and spatially extended DN control models. A. inset: Prediction of the best DN model (chosen by validation set accuracy) for all neurons to gratings of increasing size. The gratings' properties were determined from the units' optimally stimulating Gabor pattern. As grating diameter increased, only very few neurons showed mostly weak suppression. Predictions normalized to maximum response per neuron. Suppression index measures asymptotic suppression relative to the maximum prediction **A. main panel**: Across all neurons and the ten best DN models (chosen by validation set accuracy), almost no neurons show significant surround suppression. **B**. Test set performance of the ten best performing DN models. The model's performance rapidly decreases for spatially increasing normalization pool size (in units of visual angle in degrees). The best model on the validation set is indicated by a blue dot. **C. & D**. Weights of the spatial normalization pool for the best performing model with pool size of (**C**.) 1.06˚ of visual field (5 px × 5 px) and (**D**.) 1.34˚ of visual field (7 px × 7 px; all evaluated in terms of the validation set accuracy). For each feature (columns), the two components (rows) of the in total 32 spatial normalization pools are shown. Darker color corresponds to higher weights. Both components are similar. **B. insets**: Average across features and normalization pool components. The model learned normalization from the receptive field center (on average).

https://doi.org/10.1371/journal.pcbi.1009028.g007

neuron, we generated a circular grating stimulus with optimal orientation and spatial frequency, which we presented to our model. As we increased the grating diameters, our DN model's spike count predictions were suppressed only for very few cells for very large stimuli (Fig 7A, inset), suggesting only weak surround-suppression in very few cells. To quantify this observation, we computed a surround suppression index reporting the reduction from the highest measured response to the response for a grating completely filling the whole image for each neuron (Fig 7A, inset). Thus, no suppression is indicated by a suppression index of zero, whereas a value of one would mean maximum suppression (see Methods for details). Across the best ten DN models and all neurons, we found only very few exceptions from no suppression (Fig 7A) with an average suppression index of (1.6 ± 0.8)% (mean and 95% confidence interval over the ten best models selected in terms of validation set accuracy). If our DN model would have learned surround suppression, we would have expected that the predictions of a substantial number of neurons would significantly decrease for larger grating sizes leading to substantially higher suppression indices [32], which is not the case for our DN model. In conclusion, we expect the influence of very few neurons showing mostly very weak surround-suppression to our results to be very small making it unlikely that surround suppression could have led to the substantial orientation-specific normalization we observed. For the non-specific DN model and the three-layer CNN we found very similar behavior.

As a further control to verify that the orientation-specific normalization we observed in the RF is unlikely to be caused by surround influence, we fit a more general DN model which could additionally learn the spatial structure of the normalization pool, instead of limiting it only to neurons with overlapping receptive fields. To account for the RF's surround, we spatially expanded the model's normalization weights $p_{kl} \rightarrow p_{kluv}$ and introduced a convolution across space into the normalizing sum in Eq (1):

$$z_l(x) = \frac{y_l^{n_l}(x)}{\sigma_l^{n_l} + \sum_{k\in\mathcal{K}} p_{kl\cdot\cdot} * y_k^{n_k}(x)} \quad . \tag{4}$$

Here, the normalization weights $p_{kl\cdot\cdot} = p_{kluv}$ encode which input drives $y_k^{n_k}$ are pooled over what spatial extent (indexed by u, v) to normalize the input drive $y_l^{n_l}$. Note that for the special choice of normalization weights p_{kl11} we perform a $u \times v = 1 \times 1$ convolution resulting in Eq (1) describing a DN model without surround. To keep the number of parameters computationally tractable, we factorized the normalization weights p_{kluv} allowing for two normalization pools (indexed by m) per output channel l, each consisting of a set of spatial integration weights c and feature weightings d,

$$p_{kluv} = \sum_{m=1}^{2} c_{luv,m} \cdot d_{kl,m}. \tag{5}$$

We constrained c and d to be non-negative to make sure the denominator in Eq (4) is non-negative (recall that $y_k^{n_k} \geq 0$ due to Eq (2)). The two normalization pools indexed by m could have different patterns of weights along the feature and spatial dimensions. Our extended model is therefore general enough to account for the standard model of DN with a nonspecific center normalization pool and orientation-specific surround suppression (see Methods for details).

In line with what we expect from our small stimuli mostly covering the RF center, spatially expanding the normalization pool to cover larger surround areas did not increase our model's accuracy; in fact, for lager normalization pools the performance even decreased (Fig 7B; for the best models on the validation set statistically significant beyond 0.77° of visual field, pairwise Wilcoxon signed rank test: $p < 0.025$, $N = 166$ neurons, family-wise error rate $\alpha = 0.05$ using Holm-Bonferroni correction). This could be due to the increased number of parameters, which might lead to over-fitting or non-optimal learning, since adding more weights to the model makes optimization harder. The best performance was achieved by models with the smallest normalization pool mostly within the units' RFs (0.25°–0.75° diameter estimated by spike triggered average, see Methods; 0.86° estimated from the smallest grating that leads to at least 95% of the maximum response in our *in silico* analysis, averaged across neurons and the best ten DN models with smallest normalization pool). The normalization weights of the extended spatial normalization pools showed no visible separation into center and surround and exhibited no or only weak contributions from the RF's surround (Fig 7). If leakage from the surround was the source of orientation-specific normalization, we would expect to observe spatial profiles that cover the surround, which is not the case (except for a few exceptions, which are mostly untuned features).

In summary, we controlled that the orientation-specific normalization learned by our spatially constrained DN model is unlikely caused by orientation-specific influence from the RF surround but mainly from the RF center. The reason for this limitation could be in our dataset, as the stimuli we used were mostly restricted to the receptive field masking out any surround influences and their short presentation time compared to the delay after which surround influences kick in (see Discussion and Methods), preventing any model to capture information from the surround.

Discussion

To improve our understanding of primary visual cortex, we asked what condensed nonlinear function state-of-the-art CNNs might implement for predicting V1 responses to localized natural stimuli. To answer this question, we developed an end-to-end learnable divisive normalization model and fit it to neural responses. Both the unspecific control model and the full model that learned the normalization pool outperformed the subunit model, with the full DN model outperforming its non-specific variant. Through the superior performance of the DN model, we quantify the relevance of divisive normalization for predicting V1 responses to natural images. With the help of *in silico* experiments, we found that both, the DN and the CNN model learned cross-orientation inhibition, showing that the deeper layers of the CNN model approximate the DN nonlinearities. Compared to the multi-layer CNN, the DN model offers a direct, easily understandable interpretation of its normalization pool. For the DN model, we found that normalization by similar orientations, and in general by similar features, is higher compared to dissimilar features within the receptive field.

Along those lines, one may ask whether the difference between the non-specific DN model and the full model that learns orientation specific normalization weights is relevant, because the full model may simply be able to better account for some insignificant biological heterogeneous imperfection due to its additional parameters. We believe that this explanation is unlikely, because oriented features are preferentially normalized by channels with similar orientation. If the model was simply picking up some biological imperfection, we would expect random deviations from uniform normalization weights rather than weights that depend systematically on preferred orientation.

Previous experimental work investigated suppressive phenomena within the receptive field only qualitatively or used simple stimuli that mainly consisted of a combination of driving and mask gratings. Morrone and colleagues [25] found suppression at all orientations, but did not investigate orientations similar to preferred orientation. Bonds [24] reported predominantly orientation-nonspecific suppression, although three of fourteen cells exhibit stronger suppression with masks oriented similarly to the neurons' preferred orientations, and a few other cells were suppressed most strongly by mask orientations orthogonal to the preferred orientation. Similarly, DeAngelis and colleagues [26] found suppression to be predominantly independent of orientation, although for some cells an increased suppression for a range of orientations near the optimal excitatory orientation was apparent. Heeger [18] explained those results by proposing an orientation-nonspecific divisive normalization model. Carandini and colleagues [27] considered the possibility of orientation-specific normalization which provides a marginal improvement in the quality of their model fits to the data. However, they concluded that their dataset was not specifically designed to provide a strong test of this question and their results were inconclusive in this respect. Busse and colleagues [29] developed a quantitative model for the response of a population of neurons to a combination of gratings. Assuming nonspecific normalization by overall contrast, their model predicted the collective action of the whole neuron population better than linear and winner-take-all baselines, but they did not test against an orientation-specific alternative model. To summarize, these studies found phenomena that are predominantly explained by nonspecific normalization [18], some of them encountered only weak orientation-specific phenomena and only in relatively few cells. Until now, a quantitative analysis of orientation-specificity on a data set of natural images was missing.

Our findings are largely consistent with previous experimental results and quantitatively refine them using a larger dataset, place them in the context of other models of V1 and show that the same normalization mechanisms observed with simple stimuli also apply under more natural stimulus conditions. Interestingly, and somewhat in contrast to earlier work, features with preferred orientations within 10° of the driving feature provided twice as strong normalizing input than those with orthogonal preferred orientations.

The reason for this difference between our findings and previous studies could be that we used natural stimuli, which have different image statistics compared to simple stimuli used in earlier studies. Furthermore, most previous studies of divisive normalization were performed in cats [24–26, 29] and the results therein may not generalize to monkeys, for which preceding studies are inconclusive regarding orientation specificity [27].

Recent work modeling a large set of classical psychophysical data also suggests an orientation-specific divisive normalization: Schütt and Wichmann [42] developed an image-computable model of early vision very similar in structure to ours, and found that in order to explain classical data on contrast detection, contrast discrimination and oblique masking, their model required divisive normalization to be orientation-specific. Similar results had been reported in an earlier study by Itti and colleagues [43]. In contrast to those psychophysical studies involving experiments with human observers, in our paper we studied individual spiking neurons in primary visual cortex in response to natural stimulation, providing physiological evidence for orientation-specific divisive normalization. Moreover, the previous psychophysical models used a predefined Gabor filter bank and assumed that divisive normalization's orientation specificity is Gaussian distributed, learning only the standard deviation of the distribution. In contrast, our model is far more flexible, not making such strong assumptions, relying solely on the data to learn oriented filters and their divisive interactions.

Following a normative approach, Schwartz and Simoncelli [21] derive an ecologically justified divisive normalization model from the efficient coding hypothesis [44] that is able to qualitatively describe the orientation masking data of Bonds [24]. Reducing the statistical redundancy of responses to natural stimuli predicts that normalization should be stronger for neurons that exhibit a higher dependency in their unnormalized responses. This theoretical result implies that normalization weights should not be uniform, consistent with our empirical findings. Iyer and Burge [45] compare the output statistics of a divisive normalization model with broadband and narrowband normalization to natural images, $1/f$ stimuli and white noise stimuli. They report that feature-specific divisive normalization improves the discriminability among natural images compared to unspecific normalization.

Other research motivated by the efficient coding hypothesis [44] found that a recurrent sparse coding model reproduces several phenomena explained by DN, like cross-orientation inhibition [46]. In a recurrent excitation-inhibition network trained with Hebbian learning rules the inhibitory connections yielded stronger inhibition for similarly oriented cells, while the responses of excitatory cells were decorrelated [47]. This result of orientation specific normalization is qualitatively consistent with the findings of Schwartz and Simoncelli [21] and our results. These earlier models were trained using an unsupervised objective. Future research could focus on investigating the relationship between DN and recurrence within cortical circuits using recurrent models trained to predict neural responses in V1 to natural stimuli. Another interesting question could be to which degree efficient coding models match our findings of orientation specific normalization.

Is our discovery of divisive normalization by similar orientations actually implemented by the connectivity of neurons in primary visual cortex? The answer to this question could be reflected in the connectivity from inhibitory parvalbumin-expressing (PV) interneurons to pyramidal cells and their relation to neurons' tuning properties. Hofer and colleagues [48] find that, in the mouse, pyramidal cells and PV cells are homogeneously connected. Although a weak bias towards orientation tuning is apparent, they conclude that local inhibition in V1 is primarily non-specific. However, despite the connection probability between PV and pyramidal cells being homogeneous, connection strengths are quite heterogeneous: Individual PV cells strongly inhibit those pyramidal cells that share their visual selectivity [49]. This result is in line with our finding of orientation-specific normalization. Similarly, recent work employs single-neuron perturbations in layer 2/3 of mouse V1 and directly measures higher suppression of neighbouring excitatory neurons with a similar preferred orientation [50].

The stimuli in our dataset were specifically designed to investigate nonlinear processing in the RF center and to minimize surround suppression. The control analyses showing that surround suppression is unlikely in our dataset serve only the purpose of verifying that our conclusions about orientation-specific normalization in the receptive field are not due to leakage of surround suppression (which is known to be orientation-specific [30–34]). Without question, surround suppression is an important component of nonlinear processing in V1 and deserves further investigation and modeling using a dataset designed for that purpose.

In our stimuli the fully visible part was spatially restricted to approximately the size of the receptive field. For our dataset the RF diameters of 0.25°—0.75° have been roughly estimated by spike-triggered average. RF size estimates based on the grating summation field (GSF) tend to be larger by approximately a factor of two [32]. The GSF is defined by the smallest grating diameter that leads a unit to respond with at least 95% of its maximum activity [32]. Accordingly, the average GSF size we estimated based on our *in silico* size-tuning experiments had a diameter of 0.86°, roughly corresponding to the fully visible part of our stimuli (1° of visual field). For the remaining stimulus portion (1°—2°) outside of the GSF there could be normalizing influences from the surround (Figs 1A and 2 in ref. [32]). However, it is rather unlikely that this region strongly influenced our data, as in our stimuli a cosine mask fading to zero was applied in this area (see Methods). Rather, we would expect significant surround effects beyond 2° of visual field, which is not covered by our stimuli. Furthermore, in the results we have shown here, our model saw only the central 1.14° of the images (except for the variants extended to the surround, which did not learn significant surround interactions either). Thus, as we expected, our DN model did not exhibit significant surround suppression in our size-tuning control experiments and spatially extended DN control models suggested no surround interaction (Fig 7). Summarizing our control analysis, literature suggests that in general there are surround interactions under stimulation with natural images [34, 51], however, our spatially constrained stimuli likely restricted us to study DN within the receptive field and very likely prevented us from learning the influence of the RF's surround on normalization, making the surround unlikely to affect our result of orientation-specific normalization within the RF center.

Another property of our dataset that minimized the effect of surround suppression is that we focused on single images to predict a spike count in a relatively short time window covering the transient response, and ignored any temporal aspects or more sustained periods of the response. If there was some interaction from the partly masked surround, we would expect the onset of these effects to be delayed. Several studies report that the onset of surround suppression exhibits a 15—60 ms delay relative to the onset of center RF responses [52]. Bair and colleagues [53] reported average delays as short as 9 ms, however, surround suppression's peak effect was typically reached 40—50 ms after its onset. As responses to the center RF occur on average 52 ms after stimulus onset [53], in the dataset we used [14] we would expect significant surround influences to set in at the end of the recording window, which was 40—100 ms after stimulus onset. Hence, if there might be any contribution from the surround, we would expect it to contribute only very weakly to the overall spike counts determined in this time window. Note, the limitations in studying surround effects are imposed by the available data—the modeling approach may well generalize to cover both the surround and the temporal structure —and thus the extra-classical receptive field should be addressed in future work with a dataset containing larger stimuli.

Compared to our divisive normalization model, the three-layer CNN performs a few percentage points FEV better, showing that it captures additional (non-linear) phenomena that divisive normalization cannot account for. Future work will be necessary to find out what these unexplained effects are. This result shows the importance of quantitative, data-driven modeling: our work suggests that a model that can account for pooling of subunits (complex cells) and divisive normalization is structurally insufficient to achieve optimal performance in modeling V1 responses.

The performance difference between subunit and DN model measured in terms of FEV is also just a few percentage points. However, the phenomenological difference between those models is quite substantial, as the DN model correctly captures cross-orientation inhibition while the subunit model does not. Therefore, one should probably not only measure model performance by considering FEV over a set of randomly sampled natural images, but instead also consider other metrics or specifically chosen test sets that highlight certain aspects of neural computation—such as cross-orientation inhibition.

Important future research directions that follow our methods include studying the role of divisive normalization in higher areas of the ventral stream, like V4 and inferotemporal cortex. Building a predictive model by stacking successive DN layers or including a final divisive nonlinearity for the shared feature space are great candidates for future consideration, especially as earlier studies already report signs of normalization in these higher areas [54–56].

In conclusion, we developed a model consisting of one layer of subunits followed by learned orientation-specific divisive normalization, which accounted remarkably well for V1 data. We hope that this quantitative approach of evaluating theories of computation in the brain by formalizing them as (components of) trainable predictive models will be used more widely in the future, so the field will (slowly) converge to an accurate general-purpose model of the visual system applicable to natural inputs.

Methods

Experimental details

We used the dataset described in detail by Cadena and colleagues [14] and provide a summary of the most important characteristics here. Electrophysiological recordings from two healthy, male rhesus macaque monkeys aged 12 and 9 years were performed with a 32-channel linear silicon probe. The monkeys were head-fixed and placed in front of a screen. They were trained to fixate on a target located at the center of the screen. The start of a trial was determined by maintained fixation on the target for 300 ms. The fixation tolerance was set to 0.42° around the center of the target. At the beginning of each recording session, population receptive fields were mapped with a sparse random dot stimulus. Each dot was of size 0.12° of visual angle and was presented over a uniform gray background, changing location and light intensity (black or white) randomly every 30 ms. The receptive field profiles per electrode channel were then obtained via reverse correlation (i. e. spike-triggered average). The center location of the population receptive field was subsequently estimated by averaging over channels and fitting a twodimensional Gaussian to the reverse correlation profiles. Afterwards, this location was used to place the images of the natural stimulus paradigm.

The dataset by Cadena and colleagues [14] consists of 7 250 distinct natural, greyscale images which were presented two to four times each. A fifth of these images (1 450) were taken from ImageNet [57]. Four additional texturized images were synthesized from each of them, preserving varying degrees of higher-order statistics. The images were cropped to 2 × 2 degrees of visual angle (140 px × 140 px). Before displaying the images on the screen, the images were normalized such that the central 1° (70 px) of each image had the same mean (111.5) and standard deviation (45) determined across the central portion of all original images. Pixels with an intensity that fell outside the display's range [0, 255] where clipped. Afterwards, all images were overlaid with a circular mask with a soft cosine fade-out fading to the screen's mean gray intensity (128) and an aperture with a diameter of 1°.

Images were presented for 60 ms with no blanks in between. Neural responses were extracted in time windows of 40–100 ms after image onset (Fig 2), accounting for typical response latencies in primary visual cortex. The image sequence was randomized with the restriction that consecutive images do not belong to the same type (i. e. natural or one of the four texturized versions).

A few isolated neurons were discarded if their stimulus driven variability was too low [14]. The explainable variance in a dataset is smaller than the total variance because the observation noise prevents even a perfect model to account for all the variance in the data. Thus, targeting neurons that have sufficient explainable variance is necessary to train meaningful models of visually driven responses. For a neuron's spike count r, the explainable variance $\mathrm{Var}_{\mathrm{exp}}[r]$ is the difference between the the total variance $\mathrm{Var}[r]$ and the variance of the observational noise $\sigma^2_{\mathrm{noise}}$,

$$\mathrm{Var}_{\mathrm{exp}}[r] = \mathrm{Var}[r] - \sigma^2_{\mathrm{noise}} \ . \tag{6}$$

We estimated the variance of the observational noise by computing the variance of a neuron's response r_t in multiple trials t in which we presented the same stimulus x_j and subsequently taking the expectation E_j over all images,

$$\sigma^2_{\mathrm{noise}} = \mathrm{E}_j[\mathrm{Var}_t[r_t|x_j]] \ . \tag{7}$$

Neurons for which the ratio between the explainable to total variance was below 0.15 were removed. The resulting dataset includes spike count data for 166 isolated neurons, with an average ratio of explainable to total variance of 0.285. These neurons were recorded at 1°–3° eccentricities and estimated receptive field size diameters were between 0.25° and 0.75°. Since RF sizes were roughly estimated using spike-triggered average, it is likely that the values reported here underestimate the grating summation field defined by the smallest grating diameter that leads a unit to respond with at least 95% of its maximum activity by a factor of approximately two (see Discussion; similar to the minimum response field (MRF) underestimating the GSF as reported by Cavanaugh and colleagues [32]).

To keep the results of our models consistent and comparable to the gold standard baseline from Cadena and colleagues [14], we down-sampled the images by a factor of two to train our models. Likewise, images were cropped symmetrically, keeping the 40 × 40 central pixels (1.14° of visual angle). This size covers all of the recorded neurons' receptive fields, with a slight variability in their spatial location. Furthermore, the stimuli light intensities across all pixels and all images were centered around zero and normalized to have unit standard deviation. Additionally, we used the same random dataset splits of Cadena and colleagues [14] into training (64%), validation (16%) and testing (20%). We assessed our models' accuracy for a specific architecture or set of hyper-parameters in the validation set and we report performance on the test set. We consistently used the same split throughout our study.

Divisive normalization model

Our model consists of two parts, a nonlinear core and a linear readout (see Results and Fig 1). The core (Fig 1A) processes the input stimulus x by convolving it with 32 filters w_k of size 13 px × 13 px (0.37° of visual angle) without padding, defining a bank of features indexed by k. Subsequently, we apply batch normalization [58] without re-scaling (BN*) which adds a bias term (o_l in Eq 2) and scales the responses to be of unit variance. This operation does not affect the overall computation and biological interpretation of our model, since scaling the driving inputs (y in Eq 1) by a factor β whilst scaling the normalization weights (p in Eq 1) and linear readout weights (Eq 3) by $1/\beta$ yields the same output. The batch normalization step is followed by a rectified linear unit (ReLU) nonlinearity

$$f(\cdot) = \max(0, \cdot) \ . \tag{8}$$

Hence, equivalent to Eq (2), the resulting 32 feature maps of size 28 px × 28 px for the excitatory drive are given by

$$y_l = f(\,\mathrm{BN} * (w_l * x)) \tag{9}$$

and contain information about the full input space covering 1.14° of visual field. Many neurons perform similar computations but respond at different localized areas of the visual field. Those receptive fields are represented by the kernels w_l, which we implemented convolutionally to make use of this knowledge. Furthermore, the ReLU nonlinearity (Eq 8) ensures that all feature maps are non-negative, $y_l \geq 0$ being coherent to the biological interpretation of an excitatory drive.

The feature maps y_l are then normalized divisively to produce 32 output feature maps

$$z_l = \frac{y_l^{n_l}}{\sigma_l^{n_l} + \sum_k p_{kl} \, \langle y_k^{n_k} \rangle} \tag{10}$$

shared by all neurons. Here, all operations are element-wise and the scalar semi-saturation constant $\sigma_l \geq 0$ is learned from the data. To include normalization by other channels k, we first exponentiate the excitatory feature maps y_k by the scalar $n_k \geq 0$ element-wise, which is learned from the data as well. Subsequently, low-pass filtering $\langle y_k^{n_k} \rangle$ is performed through average pooling in space with pool-size 5 px × 5 px (0.14°, covering 0.49° in the input space taking the convolutional layer into account). We perform this pooling in order to achieve (approximate) phase invariance of the normalizing input without requiring a large number of filters with different phases. Subsequently, the results of the low-pass filtering are summed up, weighted by the normalization weights p_{kl}, and added into the denominator, resembling Eq (1). Furthermore, the normalization weights are constrained to be non-negative, $p_{kl} \geq 0$. Together with $y_k \geq 0$ and $\sigma_l \geq 0$, this ensures that the denominator in Eq (10) is non-negative, hence having a well-defined biological interpretation.

We converted the core's output feature maps z_l, shared by all neurons, to the activity of individual neurons via a readout for each of them (Fig 1B). To do so, we factorized the readout into spatial readout weights $a_{uv,i} \geq 0$ and feature readout weights $b_{l,i} \geq 0$ that pick the relevant locations and features plus a bias q_i,

$$g_i = \sum_{u,v,l} (a_{uv,i} \, b_{l,i}) \, z_{uvl} + q_i \ . \tag{11}$$

Here, u, v index space and i indexes neurons. This factorization is beneficial because it reduces the number of parameters in the readout. Also, we wanted to ensure that the readout does not model any complex computations, which we achieved by this factorization and the non-negativity of the readout weights. Additionally, we limited complexity by imposing a sparseness prior on both weights, because each neuron should only respond to its receptive field which is represented by a sparse spatial readout weight and should not mix many different features which corresponds to a sparse feature readout weight. The readout can, however, model a complex cell [59] by linearly combining multiple channels of the shared feature space.

We fitted an output nonlinearity to obtain the final prediction of a neuron's activity $\hat{r}_i$ as suggested by Cadena and colleagues [14] from which we paraphrase the description here. Optimizing the output nonlinearity improves data-driven models, but has to be done carefully when simultaneously learning the shared core and the readout weights end-to-end. We therefore construct the output nonlinearity inspired by the shifted exponential linear unit (ELU*) [60],

$$\hat{r}_i = h_i(g_i) \cdot \mathrm{ELU}^*(g_i) \;, \quad \mathrm{ELU}^*(g_i) = \begin{cases} g_i & \text{if } g_i \geq 1 \\ \exp(g_i - 1) & \text{otherwise} \;, \end{cases} \tag{12}$$

keeping the resulting activity non-negative, and multiply it by a non-negative function

$$h_i(g_i) = \sum_j t_j(g_i)\, \mathrm{e}^{\alpha_{ji}} \tag{13}$$

with parameters α_{ji} which are learned for each neuron i independently. The basis functions $t_j(x)$ in the argument of h are tent functions leading to a piecewise linear function inside the exponential. The tent functions are defined as

$$t_j(x) = \min\left(\max\left(0, \frac{x - x_{j-1}}{\Delta x}\right), \max\left(0, \frac{x_{j+1} - x}{\Delta x}\right)\right) \;, \tag{14}$$

with interpolation points $x_j = -3, -2.82, \ldots, 6$ and $\Delta x = 0.18$. By regularizing the coefficients α_j with a L_2 and a smoothness penalty, the model is pushed towards using the identity function for inputs larger than one and a shifted exponential function for smaller inputs. If the data provides strong evidence in favor of a different output nonlinearity, the piecewise function h_i can be used to modify it. For example, the output nonlinearity is general enough to learn a sigmoidal shape, although for our best-fit models across model types it learned a slight expansive deviation from the identity in most cases and in a few cases a mix of slight expanding and slight compressing deviations from the identity function.

To optimize our model's parameters, we maximized the log-likelihood of the model's predictions given the data. To do so, we assumed that neurons' spikes are produced by a Poisson process. Our model predicts the average spike count $\hat{r}$ of a neuron, hence the probability of observing r spikes in the experiment is

$$P(r|\hat{r}) = \frac{\hat{r}^r}{r!} \mathrm{e}^{-\hat{r}} \;. \tag{15}$$

From that follows the Poisson log-likelihood

$$\ln P(r|\hat{r}) = \sum_{i,j} (r_i(x_j) \ln \hat{r}_i(x_j) - \ln(r_i(x_j)!) - \hat{r}_i(x_j)) \tag{16}$$

for all neurons i and all stimuli x_j. A neuron's response $r_i \equiv r_i(x_j)$ depends on the stimulus x_j, which we suppress in our further notation for better readability. For implementation reasons, we wanted to minimize the Poisson loss function

$$\mathcal{L}_{\text{Poisson}} = \sum_{i,j} (\hat{r}_i - r_i \ln \hat{r}_i) \ , \tag{17}$$

which is the negative of the Poisson log-likelihood (Eq 16), where we omitted $\ln(r_i!)$ since this term does not depend on our model.

Furthermore, three terms regularizing the model's parameters were applied to the loss. We imposed a smoothness prior on the kernels w_k to ensure the spatial continuity of the predictors' receptive fields. The according penalty on the loss for weights being not smooth was determined with a Laplace filter L to be

$$\mathcal{L}_{\text{smooth}} = \sqrt{\sum_{u,v,k} (L * w_k)^2_{uv}} \ , \quad L = \begin{pmatrix} 0.25 & 0.5 & 0.25 \\ 0.5 & -3 & 0.5 \\ 0.25 & 0.5 & 0.25 \end{pmatrix} . \tag{18}$$

Due to their receptive fields, neurons only respond to a small, localized area of the visual field, which is why we imposed a sparsity prior on the spatial readout weights a_{uv}. Furthermore, neurons should only pool from a small set of feature maps to ensure that the readout does not perform complex computations. Thus we imposed a sparsity prior on the feature readout weights b_l as well. We achieved this by adding the L_1-norm of both weights

$$\mathcal{L}_{\text{sparse}} = \sum_i \sum_{u,v,l} |a_{uv,i}| \cdot |b_{l,i}| \tag{19}$$

to the loss function. As described by Cadena and colleagues [14], we regularized the output nonlinearity for all neurons by penalizing the sum of squares of the first and second finite differences of the weights α_{ji} to keep the learnable exponential function $h_i(x)$ smooth,

$$\mathcal{L}_{\text{out}} = \frac{1}{N} \sum_i \sum_{j=1}^{N} \left((\alpha_{j,i} - \alpha_{j-1,i})^2 + (2\alpha_{j,i} - \alpha_{j-1,i} - \alpha_{j+1,i})^2 \right) . \tag{20}$$

The final loss function to minimize with respect to our model's parameters is

$$\mathcal{L} = \mathcal{L}_{\text{Poisson}} + \lambda_{\text{smooth}} \mathcal{L}_{\text{smooth}} + \lambda_{\text{sparse}} \mathcal{L}_{\text{sparse}} + \lambda_{\text{out}} \mathcal{L}_{\text{out}} \ , \tag{21}$$

where λ_{smooth}, λ_{sparse} and λ_{out} are hyper-parameters which set the strength of the smoothness, the sparsity and the output nonlinearity penalty, respectively.

Divisive normalization model extended to normalization from surround

To extend our DN model to capture normalization from the spatial surround of a unit's RF, we replaced the weighted sum accounting for normalization (Eq 10) by a convolution that also covers space, keeping the rest of the original DN model unchanged,

$$z_l = \frac{y_l^{n_l}}{\sigma_l^{n_l} + s_l} \ , \quad s_l = \sum_k p_{kl\cdot\cdot} * \langle y_k^{n_k} \rangle \ , \tag{22}$$

equivalent to Eq (4). The new shared feature maps z_l consist of all element-wise operations where the newly introduced normalization feature maps s_l represent the strength by which the excitatory drive $y_l^{n_l}$ is normalized. The normalization feature maps are the result of a convolution between $\langle y_l^{n_l} \rangle$ and normalization pool kernels p_{kluv}.

For a larger convolutional kernel p, the feature maps s have smaller spatial dimensions than the excitatory feature maps y due to the convolution without padding. To be able to perform the element-wise division, we symmetrically cropped the excitatory feature maps y so that the resulting feature maps had the same spatial dimensions as s. Additionally, we wanted to keep the complexity (number of parameters) of the linear readout constant for all the size choices of the normalization kernel p. To this end, we symmetrically cropped the input images to a size that corresponds—after a forward pass through our model—to a shared feature space of spatial dimensions 28 px × 28 px (0.80°, covering 1.14° of the input image taking the convolutional layer into account). Overall, this process enabled a fair comparison across all sizes of p.

To keep the kernel size of p computationally tractable, we used dilated convolutions which have defined gaps between the weights of the convolutional kernels. We chose to skip four pixels between each weight which is reflected by a dilation factor of five. Thereby, we were able to pool from a relatively large extra-classical RF while using few parameters. If we would compute this convolution directly on the feature maps $y_k^{n_k}$, the dilation would cause us to skip some elements in the feature map $y_k^{n_k}$ in the convolution's inner product for one specific position of the convolutional kernel, i. e. for one specific element in the suppression feature maps s_l. To consider all those elements we preceded the inner product computation of the convolution with an average pooling of 5 px × 5 px pool size (same as the dilation factor; 0.14°, covering 0.49° in the input space taking the convolutional layer into account) which we performed at every spatial position in the input feature maps $y_k^{n_k}$. Then, exactly one weight of the convolutional kernel accounts for one 5 px × 5 px pool. In this view, the pools of the dilated kernel's neighbouring weights have coinciding boundaries. So in addition to implementing shift invariance, the average pooling makes sure that we do not loose information for the extended DN model. Due to this pooling, a normalization kernel p of spatial size 3 px × 3 px would spatially cover a normalization pool of size 15 px × 15 px (0.43°, covering 0.77° in the input image taking the convolutional layer into account). We investigated models with normalization kernel sizes of 1 px × 1 px, 3 px × 3 px, 5 px × 5 px and 7 px × 7 px which spatially covered a five times larger normalization pool due to dilation. Those normalization pools covered visual angles of 0.49°, 0.77°, 1.06° and 1.34°, respectively.

Baseline models

Convolutional neural network. Since the divisive normalization computation in our model was completely learned from the data, we wanted to compare to a baseline model that is purely data-driven as well. For this, the current state-of-the-art model is a convolutional neural network with three layers [14]. Its first convolutional layer consists of a kernel with spatial size of 13 px × 13 px (0.37° of visual field) and for the second and third layer of size 3 px × 3 px (0.09°) each, covering 0.43° and 0.49° of the input image, respectively. All layers use 32 channels, batch normalization [58] and ELU nonlinearity [60],

$$\mathrm{ELU}(x) = \begin{cases} x & \text{if } x \geq 0 \\ \exp(x) - 1 & \text{otherwise} \end{cases}. \tag{23}$$

Similar to our model's architecture, the core part of the CNN model results in a nonlinear feature space shared by all neurons which is mapped to each neuron's activity with individual readout weights factorized in spatial and feature weightings. Sparseness of both of them is achieved by adding an L_1-penalty to the according loss function. This readout employs the same final output nonlinearity as in our DN model but differs from ours in having no non-negativity constraints on the weights.

Convolutional subunit model. Our convolutional subunit baseline model is structurally a one-layer convolutional neural network with multiple filters followed by a readout. It is exactly the same as our divisive normalization model but with the normalization function (Eq 10) replaced by the identity function

$$z_l = \mathrm{id}(y_l) = y_l \ . \tag{24}$$

Hence, the only difference to our DN model is the lack of normalization. The shared feature maps z_l consist of rectified outputs of linear filters (Eq 9) which approximate simple cells. The subsequent readout can sum up those simple cell responses with additional weightings, enabling the model to approximate complex cells [59]. We trained the model with the same loss function (Eq 21) as the divisive normalization model.

Hyper-parameter optimization

Our model's accuracy depends on several hyper-parameters. We set the initial learning-rate to 10^{-3} and used an early stopping training scheme: We evaluated the Poisson loss (Eq 17) every 100 training steps, after ten iterations of no improvement we decayed the learning-rate by a factor of three, and we repeated this four times. For the filters w_k in the first convolution, we found that a size of 13 px × 13 px (0.37° of visual field) was optimal, the same is true for the number of 32 channels.

The weights λ_{smooth} of the smoothness penalty (Eq 18), λ_{sparse} of the readout sparsity penalty (Eq 19) and λ_{out} of the output nonlinearity's penalty term (Eq 20) in the full loss function (Eq 21) were extensively cross-validated using the validation set of our data. We randomly sampled the smooth-weight λ_{smooth} from a logarithmic uniform distribution in the interval $[10^{-9}, 10^{-3.5}]$ and the readout sparse-weight λ_{sparse} from a logarithmic uniform distribution in the interval $[10^{-9}, 10^{-4.5}]$ for all models. For the divisive normalization model constrained to the receptive field center and the subunit model, we drew the output nonlinearity penalty weight λ_{out} from a logarithmic uniform distribution in the interval $[10^{-8}, 10^{2}]$. For those two models, we sampled 1 000 runs. For the non-specific divisive normalization model and the DN models extended to the spatial surround, we narrowed down the relevant parameter space of λ_{out} and sampled them from a logarithmic uniform distribution in the interval $[10^{-5}, 10^{0}]$. Since this halved the search space of the hyper-parameters, we halved the number of samples drawn for those models. Thus, the density of sample points in the search space was the same for all models and should lead to a fair comparison between their performance. For a fair comparison of all models in this study to the state-of-the-art three-layer CNN, we retrained this model with the same optimization procedure. After running optimization for a few 100 hyper-parameter samples, we observed that the best performing models were close to the upper bound of the readout sparsity penalty weight. Thus, we discarded those data-points and started searching again with a shifted search-interval for $\lambda_{\text{sparse}} \in [10^{-5.5}, 10^{-1}]$, sampling 1 000 model runs as before. In layer 2 and 3, we set the smooth-weights to zero and group-sparsity weights to $2.5 \cdot 10^{-4}$ since Cadena and colleagues [14] found best performance for those values and these parameters are specific to the three-layer CNN, being not applicable to the other models used in this study.

Accuracy evaluation

Average correlation. For architecture search, hyper-parameter optimization and the selection of specific models for analysis we evaluated models' accuracies on the validation set with the Pearson correlation coefficient between the measured spike counts and our models' predictions, averaged over neurons. If the prediction for one neuron is constant, the according standard deviation is zero. Hence, for such neurons the correlation coefficient was not computable due to division by zero. For those neurons, we set the correlation coefficient to zero before averaging. This average correlation measure does not consider observational noise (Eq 7).

Fraction of explainable variance explained. For reporting accuracy values in this paper, we used the data's test set to compute the fraction of explainable variance explained (FEV) per neuron

$$\mathrm{FEV} = 1 - \frac{\mathrm{Var}_{\mathrm{res}}[r]}{\mathrm{Var}_{\mathrm{exp}}[r]} \tag{25}$$

which utilizes the variance that is explainable in principle, $\mathrm{Var}_{\mathrm{exp}}[r]$ (Eq 6), and the variance of the residuals corrected by the observation noise,

$$\mathrm{Var}_{\mathrm{res}}[r] = \frac{1}{N}\sum_{j}^{N}(r_j - \hat{r}_j)^2 - \sigma^2_{\mathrm{noise}} , \tag{26}$$

where j indexes images. This measure corrects for observation noise, which variance $\sigma^2_{\mathrm{noise}}$ we estimated with Eq (7). To compute model performance we averaged the FEV across neurons.

In silico experiments

To perform cross-orientation and size tuning experiments *in silico*, we first had to determine the optimal Gabor pattern that elicited maximum response for all units in the models we investigated. We specified an image containing a Gabor pattern by the Gabor's center location, size, spatial frequency, phase, orientation and contrast. For each neuron and model, we obtained the optimal Gabor image by finding the parameter combination that lead to maximum model prediction. For the center location we tested all positions in the 40 px × 40 px (1.14° of visual field) image using 12 different orientations with equally spaced angles $\phi/\pi = 0, 1/12, 2/12, \ldots, 11/12$ (in units of π) and 8 phases $\psi/2\pi = 0, 1/8, 2/8, \ldots, 7/8$ (in units of 2π). We searched for size, spatial frequency and contrast with equidistant values in log-space. Specifically, we sampled 8 sizes $s_i = 4\ \mathrm{px} \cdot (1.3895)^i$, $i = 0, 1, \ldots, 7$, resulting in a maximum size of 40px, where the size of the Gabor is defined as ±2 standard deviations of its Gaussian envelope. For spatial frequency, we used 10 values $f_i = (1.3)^{-1} \cdot (1.3)^i$, $i = 0, 1, 2, \ldots, 9$ measured in cycles per four standard deviations of the according Gaussian envelope. We determined the maximum contrast from the images in our training set. After normalization, the mean across all pixel values was zero, the minimum and maximum pixel values were −2.52 and 3.20, respectively. Thus, making sure that the Gabor's pixel values are within the value range seen by the models during training, we set the maximum contrast to $c_{\max} = 2 \cdot 2.52$ corresponding to a maximum amplitude of $a_{\max} = 2.52$. We chose 6 amplitude (contrast) values $a_i = c_i/2 = 0.01\ a_{\max} \cdot (2.51189)^i$, $i = 0, 1, 2, \ldots, 5$ resulting in $c_5 = c_{\max}$.

Cross-orientation inhibition. For the cross-orientation inhibition experiment, we used the parameters determined for the optimal Gabor to generate plaids (Fig 4A, insets) by linear superposition of the optimal Gabor with a 90° rotated version of it. We sampled the contrasts of both Gabor components from all combinations out of $\mathcal{C} = \{0, c_i\}$ with $c_i = 0.01\ c_{\max} \cdot (1.63069)^i$, $i = 0, 1, 2, \ldots, 8$ where the highest contrast was 50% of the maximum contrast, $c_8 = 0.5\ c_{\max}$, making sure the pixel values of the combined plaid stimulus stayed within the value range seen by our models during training. In addition, for the orthogonal Gabor we used 8 phases $\psi/2\pi = 0, 1/8, 2/8, \ldots, 7/8$ (in units of 2π). We obtained contrast tuning curves presenting all plaid stimuli averaging across phases to get results that are independent of relative phase between optimally and orthogonally oriented Gabor components. To quantify cross-orientation inhibition strength, we defined a cross-orientation inhibition index (COI) measuring the relative reduction in response due to adding an orthogonal Gabor with contrast c_i to the driving Gabor of optimal orientation with contrast c_j for all contrast combinations $(c_i, c_j) \in \mathcal{C}_{\text{orthogonal}} \times \mathcal{C}_{\text{optimal}}$,

$$\text{COI}\,(c_i, c_j) = 1 - \frac{\hat{r}(c_i, c_j)}{\hat{r}(c_i = 0, c_j)} \quad . \tag{27}$$

We define a cell to be cross-orientation inhibited if the cross-orientation inhibition index COI is at least 10% for any combination of optimal and orthogonal Gabor contrasts, i.e. the presence of the orthogonal Gabor reduces response by at least 10%.

Size tuning and surround-suppression control analysis. To obtain size tuning curves, we generated gratings using the parameters of the optimal Gabor for each cell. Then, instead of presenting a Gabor with a Gaussian envelope, we present a sinusoidal grating with full contrast and a sharp cut-off to zero intensity (50% grey) outside of a circular area with predefined diameter (Fig 7). We set the maximum grating diameter to d_{max} = 120px (3.43° of visual field), making sure to cover the whole 40 px × 40 px (1.14° of visual field) image even if the grating's center would be in one of the image corners. Then, we chose diameters $d_i = 0.05\ d_{\text{max}} \cdot (1.23859)^i$, $i = 0, 1, 2, \ldots, 14$. As almost all optimal Gabors had maximum contrast, we presented the gratings in this experiment with maximum contrast, too. To quantify any surround suppression, we used the suppression index (SI) proposed by Cavanaugh and colleagues [32], measuring the reduction from the highest model prediction $\hat{r}_{\text{max}}$ to the prediction $\hat{r}_{\text{supp}}$ for a grating completely filling the whole image for each neuron,

$$\text{SI} = 1 - \frac{\hat{r}_{\text{supp}}}{\hat{r}_{\text{max}}} \quad . \tag{28}$$

Hence, a SI of zero corresponds to no suppression, whereas a fully suppressed cell would lead to a SI of one.

Evaluation of orientation-specific normalization

To analyse how the preferred orientation of the features being normalized depend on that of the features providing normalizing inputs (Fig 5), we determined for each feature map whether it extracts oriented features and—if so—its preferred orientation. To do so, we windowed each convolutional kernel of size 13 px × 13 px (0.37° of visual field) with a Gaussian window (standard deviation: 3 px, corresponding to 0.09°), normalized it and then computed its 2D power spectrum (using the discrete Fourier transform with 64 × 64 samples). We then quantified how power spectral density is distributed across orientations by computing a mean resultant vector m given by:

$$m = \sum_{u,v \in R} F_{uv} e^{2\mathrm{i}\phi}, \tag{29}$$

where F_{uv} is the Fourier transformed kernel, $R = \{(u, v) : 0.3 < \sqrt{u^2 + v^2} < 0.7\}$ contains all frequencies between 0.3 and 0.7 (with 1.0 being the Nyquist frequency), $\phi = \text{atan2}(v, u)$ is the orientation, i the imaginary unit and the factor 2 in the complex exponential accounts for the fact that we are interested in orientation, which is periodic in 180° or π. If all power in a kernel is concentrated in one orientation, the mean resultant vector will be long, whereas an unoriented kernel will have a mean resultant vector near zero. Based on visual inspection of the kernels in model fits of the top 11 to 20 model runs (in terms of validation set accuracy), we found $|m| = 0.125$ to be a reasonable threshold for separating oriented from unoriented features and used it as a heuristic for further analyses of the best ten model runs, avoiding issues with post-hoc statistical testing.

To quantify how strong a feature l is normalized by other features k, we computed the average normalizing input, which is given as the expected value (over images) of the product $p_{kl} \cdot y_{kuv}(x)$ in Eq (1). Since this normalization input depends on the stimulus, we computed its expected value of all images in the validation set. We removed the dependence on space by averaging over all locations within the feature map.

Control: All channels contribute to our model's prediction

To verify that all features contribute to normalization, we analyzed the readout feature weights for the best ten models (assessed in terms of performance on the validation set). However, there are two issues that prevent a direct comparison across models and neurons of the feature weightings. First, the factorization of the readout into spatial and feature weightings is not unique: scaling the spatial weights (a in Eq (11)) by a factor β whilst scaling the feature weights (b in Eq (11)) by $1/\beta$ yields the same output limiting comparisons across neurons. Second, a similar exercise between the normalization weights p and the semi-saturation constant σ (Eq 10) impedes comparison across models. To solve these issues, we normalized the feature readout weights of each neuron for this control analysis so that the resulting vectors for each neuron and model convey how much a certain feature map contributes to predict a neuron's response compared to the other feature maps, making the feature readout weights comparable across neurons. Next, we averaged these weights across neurons to assess the importance of the features in a model. Since these normalized feature readout weights were comparable across both neurons and models, we calculated a collective distribution of the averaged feature readout weights from the best ten models. To make sense of this distribution's absolute values, we evaluated its width in terms of the coefficient of variation, which is the standard deviation in units of the mean.

Implementation details

We used Tensorflow [61] to implement models as well as Python, which we additionally used for data analysis. We optimized models with the Adam optimizer [62] using mini-batches of size 256. In addition, we used the Python packages Numpy/Scipy [63], Pandas [64], Matplotlib [65], Seaborn [66], Datajoint [67, 68] and the tools Jupyter [69] and Docker [70].

Acknowledgments

We thank Fabian H. Sinz, Felix A. Wichmann, David A. Klindt, Alexander Böttcher, Matthias Kümmerer, Dylan M. Paiton, Robert Geirhos, Claudio Michaelis and Ivan Ustyuzhaninov for valuable discussions. M.F.B. and S.A.C. thank the International Max Planck Research School for Intelligent Systems (IMPRS-IS). Disclaimer: The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies or endorsements, either expressed or implied, of IARPA, DARPA, DoI/IBC, or the U. S. Government.

Author Contributions

Conceptualization: Max F. Burg, Santiago A. Cadena, Andreas S. Tolias, Matthias Bethge, Alexander S. Ecker.

Data curation: Max F. Burg, Santiago A. Cadena, George H. Denfield, Edgar Y. Walker.

Formal analysis: Max F. Burg, Santiago A. Cadena, Alexander S. Ecker.

Funding acquisition: Andreas S. Tolias, Matthias Bethge, Alexander S. Ecker.

Investigation: Max F. Burg, Santiago A. Cadena, George H. Denfield.

Methodology: Max F. Burg, Santiago A. Cadena, Andreas S. Tolias, Matthias Bethge, Alexander S. Ecker.

Project administration: Max F. Burg, Andreas S. Tolias, Matthias Bethge, Alexander S. Ecker.

Resources: Santiago A. Cadena, George H. Denfield, Edgar Y. Walker, Andreas S. Tolias, Matthias Bethge, Alexander S. Ecker.

Software: Max F. Burg, Santiago A. Cadena, George H. Denfield, Edgar Y. Walker, Alexander S. Ecker.

Supervision: Andreas S. Tolias, Matthias Bethge, Alexander S. Ecker.

Validation: Max F. Burg, Santiago A. Cadena.

Visualization: Max F. Burg, Santiago A. Cadena, Alexander S. Ecker.

Writing – original draft: Max F. Burg, Santiago A. Cadena.

Writing – review & editing: Max F. Burg, Santiago A. Cadena, George H. Denfield, Edgar Y. Walker, Andreas S. Tolias, Matthias Bethge, Alexander S. Ecker.

References

1. Carandini M, Demb JB, Mante V, Tolhurst DJ, Dan Y, Olshausen BA, et al. Do we know what the early visual system does? Journal of Neuroscience. 2005; 25:10577–10597. https://doi.org/10.1523/JNEUROSCI.3726-05.2005 PMID: 16291931
2. Simoncelli EP, Paninski L, Pillow J, Schwartz O. Characterization of neural responses with stochastic stimuli. The Cognitive Neurosciences. 2004; 3:327–338.
3. Adelson EH, Bergen JR. Spatiotemporal Energy Models for the Perception of Motion. Journal of the Optical Society of America A. 1985; 2:284–299. https://doi.org/10.1364/JOSAA.2.000284
4. Rust NC, Schwartz O, Movshon JA, Simoncelli EP. Spatiotemporal Elements of Macaque V1 Receptive Fields. Neuron. 2005; 46:945–956. https://doi.org/10.1016/j.neuron.2005.05.021
5. Touryan J, Felsen G, Dan Y. Spatial structure of complex cell receptive fields measured with natural images. Neuron. 2005; 45(5):781–791. https://doi.org/10.1016/j.neuron.2005.01.029
6. Willmore B, Prenger RJ, Wu MCK, Gallant JL. The berkeley wavelet transform: a biologically inspired orthogonal wavelet transform. Neural Computation. 2008; 20(6):1537–1564. https://doi.org/10.1162/neco.2007.05-07-513
7. Butts DA, Weng C, Jin J, Alonso JM, Paninski L. Temporal precision in the visual pathway through the interplay of excitation and stimulus-driven suppression. Journal of Neuroscience. 2011; 31(31):11313–11327. https://doi.org/10.1523/JNEUROSCI.0434-11.2011
8. McFarland JM, Cui Y, Butts DA. Inferring nonlinear neuronal computation based on physiologically plausible inputs. PLoS Computational Biology. 2013; 9(7):e1003143. https://doi.org/10.1371/journal.pcbi.1003143
9. Vintch B, Movshon JA, Simoncelli EP. A Convolutional Subunit Model for Neuronal Responses in Macaque V1. Journal of Neuroscience. 2015; 35:14829–14841. https://doi.org/10.1523/JNEUROSCI.2815-13.2015
10. Yamins DLK, Hong H, Cadieu CF, Solomon EA, Seibert D, DiCarlo JJ. Performance-Optimized Hierarchical Models Predict Neural Responses in Higher Visual Cortex. Proceedings of the National Academy of Sciences. 2014; 111:8619–8624. https://doi.org/10.1073/pnas.1403112111
11. Khaligh-Razavi SM, Kriegeskorte N. Deep supervised, but not unsupervised, models may explain IT cortical representation. PLoS Computational Biology. 2014; 10:e1003915. https://doi.org/10.1371/journal.pcbi.1003915

12. McIntosh L, Maheswaranathan N, Nayebi A, Ganguli S, Baccus S. Deep learning models of the retinal response to natural scenes. In: Advances in Neural Information Processing Systems 29; 2016. p. 1369–1377.

13. Zhang Y, Lee TS, Li M, Liu F, Tang S. Convolutional neural network models of V1 responses to complex patterns. Journal of Computational Neuroscience. 2019; 46(1):33–54. https://doi.org/10.1007/s10827-018-0687-7

14. Cadena SA, Denfield GH, Walker EY, Gatys LA, Tolias AS, Bethge M, et al. Deep convolutional models improve predictions of macaque V1 responses to natural images. PLoS Computational Biology. 2019; 15(4):e1006897. https://doi.org/10.1371/journal.pcbi.1006897 PMID: 31013278

15. Kindel WF, Christensen ED, Zylberberg J. Using deep learning to probe the neural code for images in primary visual cortex. Journal of Vision. 2019; 19(4):29–29. https://doi.org/10.1167/19.4.29

16. Walker EY, Sinz FH, Cobos E, Muhammad T, Froudarakis E, Fahey PG, et al. Inception loops discover what excites neurons most using deep predictive models. Nature Neuroscience. 2019; 22(12):2060–2065. https://doi.org/10.1038/s41593-019-0517-x PMID: 31686023

17. Sinz F, Ecker AS, Fahey P, Walker E, Cobos E, Froudarakis E, et al. Stimulus domain transfer in recurrent models for large scale cortical population prediction on video. In: Advances in Neural Information Processing Systems 31; 2018. p. 7199–7210.

18. Heeger DJ. Normalization of Cell Responses in Cat Striate Cortex. Visual Neuroscience. 1992; 9:181–197. https://doi.org/10.1017/S0952523800009640

19. Carandini M, Heeger DJ. Normalization as a Canonical Neural Computation. Nature Reviews Neuroscience. 2012; 13:51–62. https://doi.org/10.1038/nrn3136

20. Sawada T, Petrov AA. The Divisive Normalization Model of V1 Neurons: A Comprehensive Comparison of Physiological Data and Model Predictions. Journal of Neurophysiology. 2017; 118:3051–3091. https://doi.org/10.1152/jn.00821.2016

21. Schwartz O, Simoncelli EP. Natural Signal Statistics and Sensory Gain Control. Nature Neuroscience. 2001; 4:819–825. https://doi.org/10.1038/90526

22. Sinz F, Bethge M. The Conjoint Effect of Divisive Normalization and Orientation Selectivity on Redundancy Reduction. In: Advances in Neural Information Processing Systems 21; 2008. p. 1521–1528.

23. Ballé J, Laparra V, Simoncelli EP. End-to-end optimized image compression. In: International Conference on Learning Representations; 2017.

24. Bonds AB. Role of Inhibition in the Specification of Orientation Selectivity of Cells in the Cat Striate Cortex. Visual Neuroscience. 1989; 2:41–55. https://doi.org/10.1017/S0952523800004314

25. Morrone MC, Burr DC, Maffei L. Functional Implications of Cross-Orientation Inhibition of Cortical Visual Cells. I. Neurophysiological Evidence. Proceedings of the Royal Society of London Series B Biological Sciences. 1982; 216:335–354.

26. DeAngelis GC, Robson JG, Ohzawa I, Freeman RD. Organization of Suppression in Receptive Fields of Neurons in Cat Visual Cortex. Journal of Neurophysiology. 1992; 68:144–163. https://doi.org/10.1152/jn.1992.68.1.144

27. Carandini M, Heeger DJ, Movshon JA. Linearity and Normalization in Simple Cells of the Macaque Primary Visual Cortex. Journal of Neuroscience. 1997; 17:8621–8644. https://doi.org/10.1523/JNEUROSCI.17-21-08621.1997

28. Freeman TCB, Durand S, Kiper DC, Carandini M. Suppression without Inhibition in Visual Cortex. Neuron. 2002; 35:759–771. https://doi.org/10.1016/S0896-6273(02)00819-X

29. Busse L, Wade AR, Carandini M. Representation of Concurrent Stimuli by Population Activity in Visual Cortex. Neuron. 2009; 64:931–942. https://doi.org/10.1016/j.neuron.2009.11.004

30. Blakemore C, Tobin EA. Lateral Inhibition between Orientation Detectors in the Cat's Visual Cortex. Experimental Brain Research. 1972; 15:439–440.

31. DeAngelis GC, Freeman RD, Ohzawa I. Length and Width Tuning of Neurons in the Cat's Primary Visual Cortex. Journal of Neurophysiology. 1994; 71:347–374. https://doi.org/10.1152/jn.1994.71.1.347

32. Cavanaugh JR, Bair W, Movshon JA. Nature and Interaction of Signals From the Receptive Field Center and Surround in Macaque V1 Neurons. Journal of Neurophysiology. 2002; 88:2530–2546. https://doi.org/10.1152/jn.00692.2001

33. Cavanaugh JR, Bair W, Movshon JA. Selectivity and Spatial Distribution of Signals From the Receptive Field Surround in Macaque V1 Neurons. Journal of Neurophysiology. 2002; 88:2547–2556. https://doi.org/10.1152/jn.00693.2001

34. Coen-Cagli R, Kohn A, Schwartz O. Flexible Gating of Contextual Influences in Natural Vision. Nature Neuroscience. 2015; 18:1648–1655. https://doi.org/10.1038/nn.4128

35. Heuer HW, Britten KH. Contrast dependence of response normalization in area MT of the rhesus macaque. Journal of Neurophysiology. 2002; 88(6):3398–3408. https://doi.org/10.1152/jn.00255.2002

36. Wainwright MJ, Schwartz O, Simoncelli EP. Natural image statistics and divisive normalization: Modeling nonlinearity and adaptation in cortical neurons. In: Rao R, Olshausen B, Lewicki M, editors. Probabilistic Models of the Brain: Perception and Neural Function. MIT Press; 2002. p. 203–222.

37. Froudarakis E, Berens P, Ecker AS, Cotton RJ, Sinz FH, Yatsenko D, et al. Population code in mouse V1 facilitates readout of natural scenes through increased sparseness. Nature Neuroscience. 2014; 17 (6):851. https://doi.org/10.1038/nn.3707 PMID: 24747577

38. Antolík J, Hofer SB, Bednar JA, Mrsic-Flogel TD. Model constrained by visual hierarchy improves prediction of neural responses to natural scenes. PLoS Computational Biology. 2016; 12(6):e1004927. https://doi.org/10.1371/journal.pcbi.1004927

39. Klindt D, Ecker AS, Euler T, Bethge M. Neural System Identification for Large Populations Separating "What" and "Where". In: Advances in Neural Information Processing Systems 30; 2017. p. 3506–3516.

40. Batty E, Merel J, Brackbill N, Heitman A, Sher A, Litke A, et al. Multilayer Recurrent Network Models of Primate Retinal Ganglion Cell Responses. In: International Conference on Learning Representations; 2017.

41. Denfield GH, Ecker AS, Shinn TJ, Bethge M, Tolias AS. Attentional fluctuations induce shared variability in macaque primary visual cortex. Nature Communications. 2018; 9(1):2654. https://doi.org/10.1038/s41467-018-05123-6

42. Schütt HH, Wichmann FA. An Image-Computable Psychophysical Spatial Vision Model. Journal of Vision. 2017; 17.

43. Itti L, Koch C, Braun J. Revisiting spatial vision: Toward a unifying model. Journal of the Optical Society of America A. 2000; 17(11):1899–1917. https://doi.org/10.1364/JOSAA.17.001899

44. Barlow HB. Possible Principles Underlying the Transformations of Sensory Messages. In: Rosenblith WA, editor. Sensory Communication. Cambridge, Massachusetts: MIT Press; 1961. p. 217–234.

45. Iyer A, Burge J. The statistics of how natural images drive the responses of neurons. Journal of Vision. 2019; 19(13):4–4. https://doi.org/10.1167/19.13.4

46. Zhu M, Rozell CJ. Visual nonclassical receptive field effects emerge from sparse coding in a dynamical system. PLoS Computational Biology. 2013; 9(8):e1003191. https://doi.org/10.1371/journal.pcbi.1003191

47. King PD, Zylberberg J, DeWeese MR. Inhibitory interneurons decorrelate excitatory cells to drive sparse code formation in a spiking model of V1. Journal of Neuroscience. 2013; 33(13):5475–5485. https://doi.org/10.1523/JNEUROSCI.4188-12.2013

48. Hofer SB, Ko H, Pichler B, Vogelstein J, Ros H, Zeng H, et al. Differential Connectivity and Response Dynamics of Excitatory and Inhibitory Neurons in Visual Cortex. Nature Neuroscience. 2011; 14:1045–1052. https://doi.org/10.1038/nn.2876 PMID: 21765421

49. Znamenskiy P, Kim MH, Muir DR, Iacaruso MF, Hofer SB, Mrsic-Flogel TD. Functional Selectivity and Specific Connectivity of Inhibitory Neurons in Primary Visual Cortex. bioRxiv. 2018; p. 294835.

50. Chettih SN, Harvey CD. Single-neuron perturbations reveal feature-specific competition in V1. Nature. 2019; 567(7748):334–340. https://doi.org/10.1038/s41586-019-0997-6

51. Vinje WE, Gallant JL. Sparse Coding and Decorrelation in Primary Visual Cortex During Natural Vision. Science. 2000; 287:1273–1276. https://doi.org/10.1126/science.287.5456.1273

52. Angelucci A, Bressloff PC. Contribution of feedforward, lateral and feedback connections to the classical receptive field center and extra-classical receptive field surround of primate V1 neurons. Progress in Brain Research. 2006; 154:93–120. https://doi.org/10.1016/S0079-6123(06)54005-1

53. Bair W, Cavanaugh JR, Movshon JA. Time course and time-distance relationships for surround suppression in macaque V1 neurons. Journal of Neuroscience. 2003; 23(20):7690–7701. https://doi.org/10.1523/JNEUROSCI.23-20-07690.2003

54. Zoccolan D, Cox DD, DiCarlo JJ. Multiple object response normalization in monkey inferotemporal cortex. Journal of Neuroscience. 2005; 25(36):8150–8164. https://doi.org/10.1523/JNEUROSCI.2058-05.2005

55. Kaliukhovich DA, Vogels R. Divisive normalization predicts adaptation-induced response changes in macaque inferior temporal cortex. Journal of Neuroscience. 2016; 36(22):6116–6128. https://doi.org/10.1523/JNEUROSCI.2011-15.2016

56. Reynolds JH, Desimone R. Interacting roles of attention and visual salience in V4. Neuron. 2003; 37 (5):853–863. https://doi.org/10.1016/S0896-6273(03)00097-7

57. Russakovsky O, Deng J, Su H, Krause J, Satheesh S, Ma S, et al. ImageNet Large Scale Visual Recognition Challenge. International Journal of Computer Vision. 2015; 115:211–252. https://doi.org/10.1007/s11263-015-0816-y

58. Ioffe S, Szegedy C. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In: International Conference on Machine Learning; 2015. p. 448–456.

59. Hubel DH, Wiesel TN. Receptive fields, binocular interaction and functional architecture in the cat's visual cortex. The Journal of Physiology. 1962; 160:106–154. https://doi.org/10.1113/jphysiol.1962.sp006837

60. Clevert DA, Unterthiner T, Hochreiter S. Fast and accurate deep network learning by exponential linear units (elus). arXiv preprint arXiv:151107289. 2015.

61. Abadi M, Agarwal A, Barham P, Brevdo E, Chen Z, Citro C, et al. TensorFlow: Large-Scale Machine Learning on Heterogeneous Distributed Systems; 2015.

62. Kingma D, Ba J. Adam: A method for stochastic optimization. In: International Conference on Learning Representations; 2015.

63. Walt Svd, Colbert SC, Varoquaux G. The NumPy array: a structure for efficient numerical computation. Computing in Science & Engineering. 2011; 13(2):22–30. https://doi.org/10.1109/MCSE.2011.37

64. McKinney W. Data Structures for Statistical Computing in Python. In: van der Walt S, Millman J, editors. Proceedings of the 9th Python in Science Conference; 2010. p. 51–56.

65. Hunter JD. Matplotlib: A 2D graphics environment. Computing in Science & Engineering. 2007; 9(3):90–95. https://doi.org/10.1109/MCSE.2007.55

66. Waskom M, Botvinnik O, O'Kane D, Hobson P, Lukauskas S, Gemperline DC, et al. mwaskom/seaborn: v0.8.1 (September 2017); 2017.

67. Yatsenko D, Reimer J, Ecker AS, Walker EY, Sinz F, Berens P, et al. DataJoint: managing big scientific data using MATLAB or Python. BioRxiv. 2015; p. 031658.

68. Yatsenko D, Walker EY, Tolias AS. DataJoint: a simpler relational data model. arXiv preprint arXiv:180711104. 2018.

69. Kluyver T, Ragan-Kelley B, Pérez F, Granger B, Bussonnier M, Frederic J, et al. Jupyter Notebooks—a publishing format for reproducible computational workflows. In: Loizides F, Schmidt B, editors. Positioning and Power in Academic Publishing: Players, Agents and Agendas. IOS Press; 2016. p. 87–90.

70. Merkel D. Docker: Lightweight Linux Containers for Consistent Development and Deployment. Linux Journal. 2014; 2014(239).