MSc Artificial Intelligence
Master Thesis

# Temporal Adaptation in the Visual Cortex: Bridging the Gap from Circuits to Behavior with Task-Driven AI-Models

by
Georg Lange
13405373

December 22, 2023

48 Credits
February - December 2023

*Daily Supervisor:*
Amber Brands

*Examiner:*
Dr. Iris Groen

Universiteit van Amsterdam

# Contents

**Abstract**

In computational neuroscience, a core objective is to identify canonical computations — fundamental operations that are integral to the brain's processing of behavior and stimuli and central building blocks for neural circuits. The translation of these computations to observable behavior, however, remains elusive.

This work examines temporal adaptation, a normalizing process believed to be universally utilized throughout the brain, that modulates neuronal activity based on recent activation within the same or adjacent neurons. Because manipulating this process in vivo is challenging, we explore its behavioral implications through artificial intelligence models equipped with temporal adaptation.

We demonstrate that reliance on gradient descent for training these models may lead to deceptive outcomes, where model behavior aligns with expectations, but the underlying mechanisms diverge significantly from our intuition. Our experiments involve training models with divisive normalization and additive adaptation enhancements for an object recognition task, specifically designed to suppress stationary noise. These models display markedly different behaviors from those with preset adaptation parameters, suggesting that simply incorporating brain-like features is not sufficient. Detailed examination of the models' internal processes is imperative to establish their congruence with brain-like functionality.

We present a successful application scenario where a model is trained on a novel object recognition task that necessitates adaptation. Through mechanistic analysis, we confirm the model's use of adaptation and its resemblance to human neural activity patterns.

In conclusion, while it is crucial to design task-driven models, a thorough mechanistic investigation is essential to validate whether the model's problem-solving approach is analogous to that suggested by neural measurements in humans.

# Chapter 1

# Introduction

To understand how intelligent systems produce complex behavior, we often try to decompose them into smaller components that are easier to understand. These components, or models, need to satisfy two constraints: They need to be interpretable for humans, and they have to have predictive power. Numerous lines of evidence suggest that the brain, but also AI models, can indeed be decomposed into modular units. For example, brain functions like memory, visual perception, or speech are localized and ablating associated brain regions leads to a loss of function [25, 5]. The same principle applies to AI models: Language models use their first layers for detokenization and compound words, perform text processing and action selection in middle layers, and retokenize in their final layers. Within this framework, concrete and localized circuits that solve a specific task of interest can be found. A good example is the Indirect Object Identification circuit [27] in which only 1.1% of attention heads determine the model's decision between two names. Moreover, factual knowledge can be localized to specific layers [17].

During the past decades, huge efforts were made in Cognitive Neuroscience to map behavior to localized regions and enhance our understanding of how computation is distributed. However, localizationism failed to be specific enough to understand how concrete circuits compute behavior and it seems that we reached a limit in the minimum size of interpretable chunks without understanding how individual patches create behavior. Instead, we argue that researching modularity in computation rather than functional localization is more informative of how neural activity translates to behavior. Specifically, our goal is to identify canonical algorithms that are utilized across regions and circuits that can help explain on a lower level how a circuit's behavior is actually computed. In large language models (LLMs), this approach has already been successfully demonstrated. For example, induction heads have been proposed as such a canonical operation [19]: In a sequence $A, B, ..., A \rightarrow ?$, they predict B by attending to the first occurrence of B and copying its value (attending to B can be achieved by a "previous token" head that copied contents from A to B at position B in a preceding layer). This mechanism is used to e.g. repeat sequences but remarkably, it has been proposed that the concept of "induction" is also utilized for abstract features and canonical operations which is used throughout the model. Most importantly, the authors claim that induction can explain the majority of in-context learning, one of the most important emergent capabilities of LLMs [19].

**Divisive Normalization as a canonical operation** With approaches like this, would it be possible to identify such canonical computations in the brain and link them to concrete behaviors? Here, we investigate one such proposed operation, namely divisive normalization. Divisive normalization is a nonlinear operation that divides a neuron's output based on summed activation values of other units. The units used for division determine the function and can be neurons with adjacent receptive fields, neurons within the same receptive field that represent different or similar features, the neuron's own recent activity, or recurrent connections.

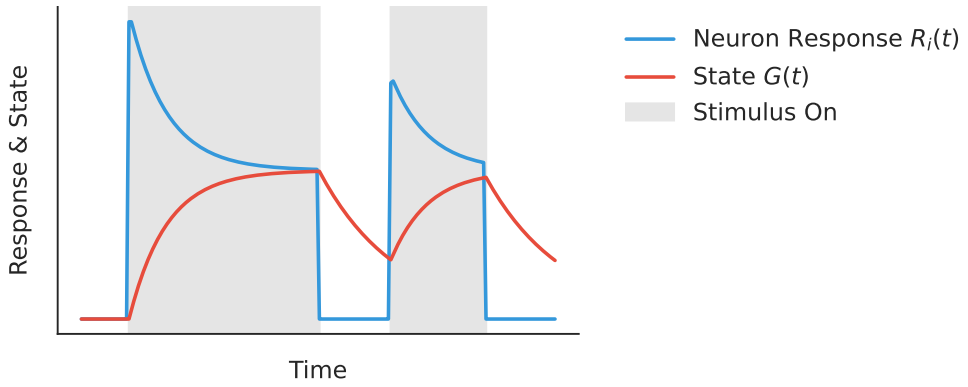



Figure 1.1: **Temporal Dynamics of Neuronal Response and State in Divisive Normalization.** The plot illustrates the neuron's response $R_i$ and the feedback signal variable $G(t)$ over time. Response to the second stimulus is suppressed because of the neuron's recent activity. $R_i$ is normalized with its own historic activity.

Divisive normalization has been observed in numerous brain regions. In the retina, normalization helps to adapt neurons to different light intensities to keep sensitivity for different ranges of lights despite a limited dynamic range [8]. Light intensities range over 10 magnitudes and while anatomical mechanisms help to reduce this range [21], further adaptation is still needed and implemented via normalization [18]. In the primary visual cortex (V1), neural responses saturate with increasing stimulus contrast while keeping sensitivity to low contrasts which can be explained by temporal or spatial normalization [1, 13]. In addition, normalization leads to cross-orientation inhibition [3], a mechanism that suppresses a neuron's response if another stimulus that the neuron wouldn't respond to if presented alone, is present. Normalization in V1 also leads to max-pooling because the presence of multiple stimuli increases the divisive term, effectively suppressing stimuli with low salience or contrast [7]. It is hypothesized that divisive normalization is widely used throughout the visual pathway and the entire cortex. For example, in the velocity-selective middle temporal area (MT), normalization might help to make V1-receiving MT-neurons invariant to spatial patterns [23], and in the ventral pathway (V4, IT), normalization might drive adaptation to complex or even intermodal stimuli [28, 20]. However, obtaining direct evidence is difficult because it's unclear whether the observed normalization effect originates there or is inherited from preceding regions like V1.

Given the widespread application of divisive normalization in various neural circuits, it's crucial to delve deeper into its fundamental mechanics. At its core, different mathematical models have been suggested, each capturing unique aspects of neural processing. Typically [6], the response $R_i$ of neuron $i$ is modelled as

$$R_i = \frac{L_i^{n_i}}{\sigma_i^{n_i} + \sum_{k \neq i} L_k^{n_k}} \tag{1.1}$$

with $L_i$ being the rectified (linear) response of neuron $i$, which is exponentiated by a constant $n_i$. $\sigma$ is a semi-saturation constant that determines the asymptote of the response and effectively limits the amount of suppression. The neuron's linear response $L_i$ is divisively normalized by the sum of neighboring neurons $L_k$. Which neurons are included in the sum determines the function and is often selected to study a particular observation. For example, spatial normalization divides by neurons with neighboring receptive fields. This can lead to an enhancement of spatial features like edges. If multiple neighboring neurons detect an edge, normalization suppresses the less responsive neurons while the winner-takes-it-all which effectively enhances precision. Spatial normalization can also lead to contrast adaptation where stimuli of different contrasts

evoke responses of similar size that effectively utilize the neuron's dynamic range.

While spatial normalization facilitates edge detection and contrast adaptation by considering adjacent receptive fields, feature-wise normalization extends the normalization pool to encompass a range of feature detectors, thereby modulating the neuron's response in a feature-selective manner. This enables a contextual modulation of neural responses that is sensitive to the distribution and strength of multiple different features. For example, the presence of multiple features, or features with different contrasts or saliency can lead to suppression of features that would evoke a great response if presented alone. This stabilizes the overall output scale and enhances salience of important features. Cross-orientation inhibition might be a direct result of this.

Transitioning from the spatial and feature domains, we acknowledge that neural computations are not static but inherently temporal. Temporal divisive normalization represents a fundamental yet frequently understated aspect of neural processing, emphasizing the importance of stimuli dynamics over time in shaping neural responses. Different functions for temporal normalization have been proposed: It might explain 'neural fatigue', the decrease in output for constant stimuli, and it might enable response enhancement by suppressing interfering or noisy stimuli. Lastly, it might contribute to bottom-up attention by decreasing salience of static stimuli. A mathematical framework was originally suggested by Heeger [13, 14] and defines a neuron's response $R_i$ recursively as

$$R_i(t) = \left[ L_i(t) \frac{\sqrt{K - G(t-1)}}{\sigma} \right]^2 \tag{1.2}$$

where $K$ is a constant that determines the maximum attainable response, $t$ the time, and $R_i$, $L_i$, and $\sigma$ as defined in 1.1. $G$ is a feedback signal that determines the amount of suppression based on preceding temporal activity of the same or neighboring neurons and is given by

$$G(t) = (1 - \alpha)G(t-1) + \alpha \sum_k R_k(t-1) \tag{1.3}$$

.

$G(t)$ integrates past activity and discounts it exponentially with a constant $\alpha$ that determines the rate of change (Figure 1.1). $G(t)$ is an inhibitory signal such that strong activity leads to local suppression of $R_i$. $R_j$ is the response of all neurons $j$ that are used to normalize a given neuron. This can be neighboring neurons, the global population, or the neuron of interest itself.

How $K$ and $\sigma$ shape the form and sensitivity of adaptation is illustrated in Figure 1.2. With respect to the maximal response and the saturated response (the response after the stimulus is shown long enough for the response to converge), $K$ doesn't affect adaptation. However, $K$ determines the maximum amount of suppression for a given input value. If the input values are much smaller than $K$, it might not be possible for the feedback signal $G(t)$ to grow to a size that is a considerable fraction of $K$, and in that instance, the total amount of inhibition is limited (Figure 1.2).

Low values for $\sigma$ on the other hand scale the linear response, which leads to a larger response but because of that, also to a faster increase in $G$ and thus to faster adaptation. In addition, it determines the height of the saturated response but while $K$ does this additively, $\sigma$ controls this divisively (Figure 1.2).

While a combination of divisive operations across time, location, and features might best approximate the mechanisms in the brain, our paper will primarily explore temporal normalization, an aspect that remains under-investigated yet holds significant potential for understanding dynamic neural processing. In this study, we augment artificial neural networks with temporal

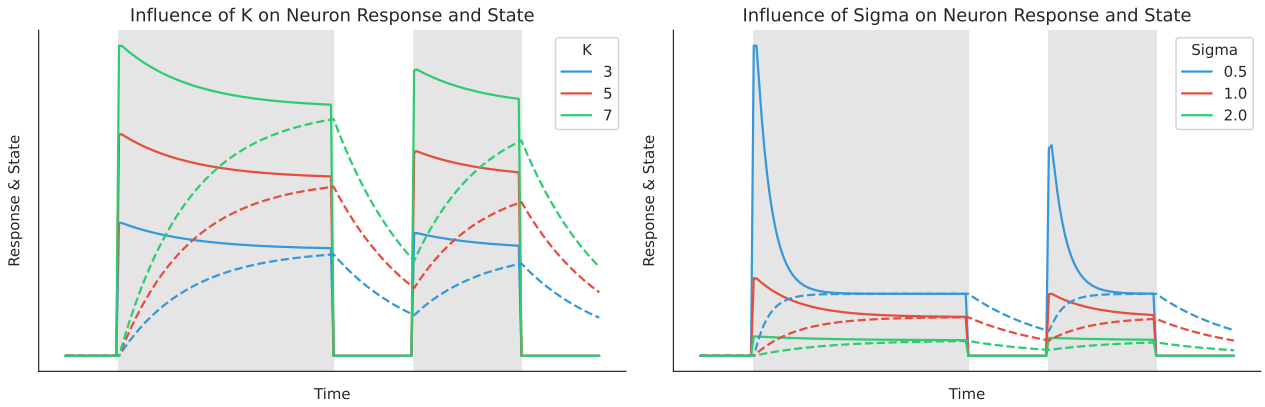



Figure 1.2: **Influence of Parameters K and Sigma on Temporal Divisive Normalization.** This figure displays the neuron's response $R_i(t)$ and feedback signal $G(t)$ across two scenarios: varying $K$ (left) and varying $\sigma$ (right). Each line represents a different value of $K$ or $\sigma$, showcasing how these parameters modulate the neuron's response and feedback dynamics over time. The dashed lines indicate the state variable $G(t)$, illustrating its integral role in the neuron's adaptive response to stimuli.

adaptation mechanisms and study if they can be successfully utilized and if they are a plausible explanation for a range of behavioral observations.

While evidence of divisive computations in the brain is well-documented, the direct link between these computations and the resulting behaviors remains elusive. To bridge this gap, our research focuses on assessing how adaptation can give rise to specific behaviors. To achieve this, we augment AI models with divisive normalization layers. By training these enhanced models on tangible tasks, we aim to shed light on whether divisive normalization can be actively employed by artificial systems and, more importantly if it stands as a plausible model for explaining behavior. This approach marks a significant shift from traditional studies, as we move towards a task-driven model evaluation.

# Chapter 2

# Methods

## 2.1 Models

Throughout the experiments, we use standard Pytorch convolutions and linear layers. Models were trained using the Adam optimizer, a learning rate of 0.001, and were optimized to minimize the categorical cross entropy using stochastic gradient descent with a batch size of 64. For fashion-mnist models trained on the noise suppression task, three convolutional layers were followed by a single linear layer. The convolutions had 32 feature maps each and a kernel size of 5, 5, and 3. The linear layer had a dimensionality of 1024 which was reduced by the decoder to 10 output neurons. For Cifar10 models, we increased the number of feature maps to 32, 64, and 64. For fashion-mnist models trained on the novel object task, we added a fourth convolutional layer between the last convolution and the linear layer with 32 feature maps. After each convolution, ReLU and max-pooling were applied. For the noise suppression experiment, the class was decoded at the last timestep. For the novel object task, every timestep was decoded. 50% dropout was applied after the last convolution. Models for the noise suppression task were trained for 20 epochs with 60000 images per epoch. For the novel object detection task, models with a single adaptation layer were trained 10 epochs, and models with multiple adaptation layers were trained 50 epochs. All models converged well before training ended. For the baseline model, no adaptation mechanism was added such that all images were processed independently.

## 2.2 Datasets

**Noise Suppression:** Each input sample consisted of three sequential images processed by the recurrent CNN. The first image, referred to as the 'adapter image,' contained noise intended for adaptation. This noise, generated uniformly between -0.5 and 0.5, was superimposed onto the mean of the test image, then clamped to the range [0, 1]. This procedure ensured consistency in the mean luminance of the sequence, preventing abrupt transitions in brightness.

The second image in the sequence was blank, its color set to the mean of the test image. The third image, termed the 'test image,' combined the same noise pattern used in the adapter image with the object-containing test image, which was to be classified.

To modulate the contrast of the test image, its contrast was randomly altered to either 100% or 20% for each instance. The contrast adjustment was performed according to the following formula:

$$\text{Adjusted Image} = \text{Mean Image} + \text{Contrast Level} \times (\text{Original Image} - \text{Mean Image}) \quad (2.1)$$

In this formula, "Adjusted Image" represents the contrast-modified image, "Mean Image" is the mean luminance of the original image, "Contrast Level" is the desired contrast ratio (either 100% or 20%), and "Original Image" is the unmodified image.

Different noise patterns were sampled for each epoch to ensure variability and robustness in learning.

**Novel Object Recognition:** Each input sample was composed of a sequence of 20 images, each sequentially fed to the model. Every image in the sequence was associated with a single label derived from the Fashion MNIST dataset. The dimensions of each image were 56x56 pixels, designed to accommodate a grid arrangement of four Fashion MNIST images ('objects') without overlap. For clarity, the term "image" refers to the composite image created for the model, while "object" denotes an individual Fashion MNIST image within the larger image. The location of each object on the composite image was randomly determined, ensuring that the positions of objects were distinct and non-overlapping within an image. Furthermore, the location of a given object remained constant throughout the 20-image input sequence.

The appearance (or 'onset') of each object within the sequence was randomized. However, every sequence was structured to include at least one object that appeared in the very first image. The contrast of each object varied randomly between 0 and 1. The task for the model was to predict the object introduced at the current timestep. In cases where no new object was added at a timestep, the model's objective was to identify the most recently added object. If multiple objects appeared simultaneously in a timestep, the model was trained to predict the object with the highest contrast.

**Novel Object Recognition with augmented images:** This task mirrored the previous experiment, with the addition of image augmentation techniques applied to each object independently. The augmentations were designed to intensify over the sequence, resulting in more substantial changes than those within a single timestep. The augmentation procedures included:

- **Rotation:** Objects were subjected to random rotations, varying between -12 and +12 degrees.
- **Translation:** A random uniform translation was applied in both the x and y directions, ranging from 0 to 4 pixels.
- **Scaling:** Objects underwent random uniform scaling, with factors between 0.9 and 1.1.
- **Shear:** Random shear transformations were applied, with angles from -5 to 5 degrees.
- **Gaussian Noise:** Gaussian noise, with a standard deviation of 0.1, was added to the images.
- **Brightness Adjustment:** The brightness of objects was randomly altered, with adjustment factors from 0.8 to 1.2.
- **Contrast Adjustment:** Contrast was also randomly adjusted, with factors ranging from 0.8 to 1.2.

After applying these augmentations, the image was clamped to maintain valid pixel value ranges.

## 2.3 Adaptation Layers

**Additive Adaptation:** Adaptation was applied after convolution and ReLU and defined as described in [26]:

$$G_i(t) = \alpha G_i(t-1) + (1-\alpha)R_i(t-1) \tag{2.2}$$

$$R_i(t) = \Phi(L_i(t) - \beta G_i(t)) \tag{2.3}$$

$L_i$ is the activation of a single unit $i$ of the model after the convolution and ReLU, $G_i$ is the latent state for unit $i$ that tracks the amount of suppression, $\Phi$ is the activation function ReLU, and $\alpha$ and $\beta$ are parameters that influence the duration and strength of adaptation. They were initialized with $\alpha = 0.5$ and $\beta = 1$. In supplementary experiments, we zero-initialized them and observed that they converge to the same values but take more time to train. The state $G$ was zero-initialized at the first timestep.

**Divisive Normalization:** Adaptation was applied after convolution and ReLU and defined as described in [13, 14]:

$$R_i(t) = \Phi\left[L_i(t)\frac{\sqrt{K - G(t-1)}}{\sigma}\right] \tag{2.4}$$

$$G(t) = (1-\alpha)G(t-1) + \alpha R_i(t-1) \tag{2.5}$$

.

where $K$ is a constant that determines the maximum attainable response, $t$ the time, and $R_i$, $L_i$, and $\sigma$ as defined in 1.1. $G$ is a feedback signal that determines the amount of suppression based on preceding temporal activity of the same or neighboring neurons. We initialized the parameters with $K = 0.3$, $\sigma = 0.3$, and $\alpha = 0.1$.

# Chapter 3

# Results

## 3.1 Augmenting CNNs with temporal adaptation capabilities to suppress recurring noise

Our primary goal is to determine if mechanisms like divisive normalization and additive adaptation can be effectively integrated into the training of specific computational models. By training artificial deep neural networks that demonstrably employ these mechanisms for certain tasks, we can infer that the proposed computations are plausible for accomplishing these tasks. Subsequently, we can examine the trained model further, comparing it with neural or behavioral data from humans to ascertain if it realistically represents certain neural circuits.

This method stands in contrast to other computational strategies that either fit models to neural data or deliberately design them to exhibit specific characteristics. While these traditional approaches are useful for demonstrating the existence of certain computations or for explaining observations, they often fall short of elucidating how known computations or circuits influence behavior.

Our approach addresses this gap by focusing on training models to learn specific tasks or behaviors, rather than directly fitting them to neural data. This allows for a more straightforward examination of the model's internal workings compared to human or animal studies. Essentially, if we can demonstrate that AI models effectively use adaptation mechanisms for real-world tasks, and if the learned algorithms closely align with in-vivo findings, it supports the notion that adaptation is a viable strategy for the tasks under study.

In pursuit of this, we have taken a standard convolutional neural network (CNN) as our base architecture and enhanced it with capabilities for recurrent (temporal) adaptation, aiming to scrutinize the influence of these mechanisms on the model's performance. This approach is inspired by the work presented in Vinken et al. [26], where they posit that intrinsic suppression, modeled as an additive adaptation mechanism, serves to eliminate interfering but temporally-constant noise from the data. The authors use a task-driven approach to train a model to suppress static noise to increase the signal-to-noise ratio in its activations which can push accuracy. A critical aspect of their findings is the superior generalization exhibited by models incorporating this adaptation, supporting the widely shared view that inductive biases and sophisticated architectures lead to leaner (less parameterized) models that exhibit less overfitting and better generalization. Besides these results, they manually augment a pre-trained AlexNet [16] and show that a variety of effects measured in the brain also emerged there. For example, they observe signs of repetition suppression and report visual aftereffects that bias the models' decision away from an adapter.

The crucial detail, however, remains missing: Relating the task-driven model to the observations made on the modified AlexNet. With the latter, they showed that they can manually tune adaptation parameters of a neural network such that desired characteristics emerge. With

the task-driven model on the other hand, they show that deep neural networks can utilize adaptation mechanisms to boost their performance. While one might assume that both observations combine to "Neural networks can use temporal adaptation to suppress continuous noise, similar to adaptation in the human visual cortex", we investigate this claim more thoroughly and show that this is not always the case.

We attempted to replicate Vinken et al's [26] experiments (Figure 7) that show that an intrinsic suppression mechanism can be used to suppress recurring noise patterns and that this generalizes much better than a fully recurrent model.

The task consisted of three images that were presented to the model subsequently (Figure 3.1). The first image, also called "adapter", is uniform noise and is followed by a blank image. The final image, which will be decoded, is the sum of a cifar10 image that the model should classify, and uniform noise. Crucially, the noise pattern between the first and third image is identical, such that remembering information from the first image can help to denoise the third image. To make this transfer of information possible, we added recurrent connections that implemented the adaptation mechanism. We tested two different models. First, we replicate the intrinsic suppression model used in Vinken et al where a neuron's response $R_i$ is given by

$$G_i(t) = \alpha G_i(t-1) + (1-\alpha)R_i(t-1) \tag{3.1}$$

$$R_i(t) = \Phi(L_i(t) - \beta G_i(t)) \tag{3.2}$$

Here, $L_i$ is the neuron response before suppression, $\Phi$ the activation function ReLU, $G_i$ is a latent state that holds information about recent neural activity used for adaptation, and $\alpha$ and $\beta$ are two constants that determine the timescale and strength of adaptation, respectively.

Second, we test a divisive normalization model originally introduced by Heeger [13, 14] and given by

$$R_i(t) = \Phi\left(L_i(t)\frac{\sqrt{K - G(t-1)}}{\sigma}\right) \tag{3.3}$$

$$G(t) = (1-\alpha)G(t-1) + \alpha\sum_k R_k(t-1) \tag{3.4}$$

as outlined in the introduction. Note that the square root and the squaring operation are technically unnecessary because they can easily be implemented by the convolutions if advantageous but we leave them here for better interoperability. While divisive normalization is thought to rely on recurrent or lateral connections in most settings, our definition normalizes solely to a neuron's own historical activity. As such, this mechanism could also be implemented intrinsically. In this case, the main difference between both ideas is the additive or divisive nature of the normalization operation.

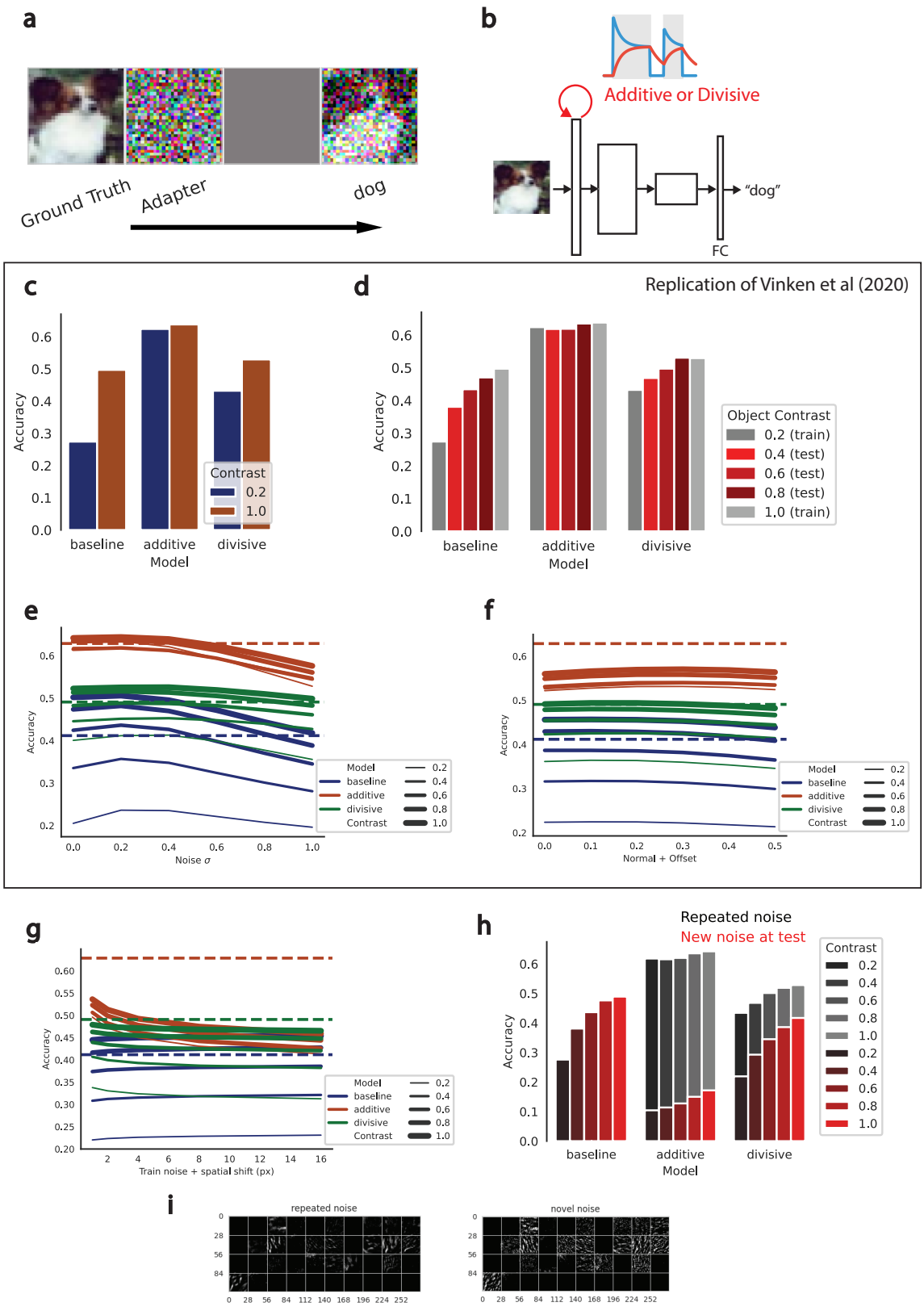
Deviating from Vinken et al, we only apply adaptation in a single layer (Figure 3.1b). Because the noise pattern is static, a single layer is able to solve the task and a single adaptation operation is much easier to interpret and understand. We train the models till convergence and set the contrast of the cifar10 image randomly to 0.2 or 1.0. A baseline model without recurrent connections achieved an accuracy of around 50% for the high-contrast images (Chance level is 10%) that considerably dropped to less than 30% for the low-contrast images, illustrating the challenge to separate a weak signal from a very noisy background. Remarkably, both adaptation models achieve higher accuracies compared to the baseline, indicating that they utilize the adaptation mechanism. The accuracy gain was higher for the low-contrast image, suggesting that normalization helped to reduce noise and to raise the signal-to-noise ratio for these images considerably (Figure 3.1c).

We then test how these models generalize to different types of noise. First, we test them on cifar10 images with a contrast of 0.4, 0.6, and 0.8 and observe that they perform similarly to the contrasts they were trained on (Figure 3.1d). Then, we tested on gaussian noise, a type of noise the models weren't trained on, and varied the variance of the noise (Figure 3.1e) or its mean (Figure 3.1f). Both tests resulted in performance similar to when tested on noise patterns the models were trained on, and only dropped slightly for very high variances, while accuracy dropped stronger for the baseline model. Also, the divisive normalization model showed these generalization abilities though its overall accuracy was a bit lower compared to the additive model. Thus, following Vinken et al [26], we conclude that the models trained with an adaptation mechanism generalize to different noise patterns.

Figure 3.1: **Adaptation-augmented models learn a temporal noise suppression task but collapse a)** An input sample consists of three images that are fed sequentially to the model. The first image is sampled from uniform noise and followed by a blank image. The third image interleaves the same noise with a low-contrast cifar10 image and has to be classified. **b)** A schematic of the model. We train an ordinary 3-layer CNN followed by a linear layer. We introduce an additive or divisive adaptation mechanism in the first layer, the only connection that shares information between images. **c)** Models were trained on a mix of high- and low-contrast images and can learn the classification task. **d)** Generalization to different contrasts. Note that the model was only trained on the grey contrasts. **e** Generalization to different noise patterns. Models were trained on uniform noise but tested on Gaussian noise with different variances. The dashed lines mark the average accuracy of each model when tested on the noise patterns they were trained on. **f)** Same as **e**, but in addition the mean of the noise is offset. **g)** Same as **e** and **f**, but the noise from the model's training distribution is used but spatially shifted between images. **h)** Generalization to unseen noise: Greyscale bars mark accuracy when noise is repeated between images, as is in the training data. Red bars mark accuracy when tested on the same noise pattern but noise is resampled between images. **i)** Resampling noise leads to off-distribution activations. Images show the activation values of all feature maps of a single sample after the first layer. For "novel noise", the noise in the adapter and test step is different.

## 3.2 Training adaptation-augmented neural networks can lead to an illusory sense of interpretability

Are these models faithful representations of adaptation in the visual cortex? While enhancing generalization capabilities in neural networks is a notable objective in itself, our primary aim is to determine whether our adaptation implementation can accurately model brain adaptation mechanisms. Hence, it is crucial to understand how the model employs the adaptation mechanism and whether this mirrors neural observations. According to Vinken et al. [26], models with manually set adaptation parameters exhibit perceptual bias and enhanced discriminability aftereffects, which are commonly linked to adaptation effects in the human brain. However, we argue that these findings may not directly apply to models trained from scratch with adaptation mechanisms. The reason for this caution is the uncertainty about whether the model discovers an alternative, perhaps previously unconsidered, approach to task resolution. Gradient descent, known for its strong optimization capabilities, has led to unforeseen behaviors in various instances. For instance, models trained via reinforcement learning might develop undesirable behaviors that yield high rewards due to imperfectly calibrated reward functions [24]. In convolutional neural networks (CNNs), gradient-based feature visualization techniques can be deceptive, displaying random patterns that create a false sense of interpretability [12]. Similarly, large language models are thought to leverage the basis-dependent momentum in


a
Ground Truth
Adapter
dog
b
Additive or Divisive
"dog"
FC
c
Accuracy
Contrast
0.2
1.0
baseline
additive
divisive
Model
d
Replication of Vinken et al (2020)
Object Contrast
0.2 (train)
0.4 (test)
0.6 (test)
0.8 (test)
1.0 (train)
e
Noise σ
f
Normal + Offset
Model
baseline
additive
divisive
Contrast
g
Train noise + spatial shift (px)
h
Repeated noise
New noise at test
i
repeated noise
novel noise

the Adam optimizer [10] to learn specific "landfill" dimensions in the residual stream, which are used to store output of components that do not contribute data [2, 9]. This highlights the necessity of a detailed mechanistic analysis when training models using gradient descent.
Because a data sample consists of only three images and we added the adaptation mechanisms after only one layer, the learned parameters are readily interpretable: The additive suppression model learned parameters $\alpha = 0.43$ and $\beta = 4.14$. Because the second image is blank and does not activate any neurons, the state of the test image only depends on the adapter image. The response for the test image $R_3$ can thus be written as

$$R_3 = L_3 - L_1 * \alpha * \alpha * \beta \tag{3.5}$$

.
Because $\alpha * \alpha * \beta \approx 1$, the model essentially learned to subtract all noise features from the test image! This behavior is advantageous for the model as completely subtracting the noise will result in the best signal-to-noise ratio. However, it wouldn't resemble visual cortex functioning which is our precondition to studying the behavior of this model. To show that the model's generalization capability indeed arises from simple subtraction of the noise, we perform a simple experiment: Instead of changing the noise patterns we test the network on, we use noise patterns the model was trained on but spatially shift it at test stage by 1, 2, 4, 6, or 16 pixels (Figure 3.1g). The test image now still has the same noise pattern but with a slight offset such that unit-wise subtraction would be compromised. Even when shifting the noise pattern only by a single pixel, we observe a big drop in accuracy of about 10%, suggesting that the model relies on complete subtraction as hypothesized. If the model had learned to generalize to different noise patterns, we would expect it to perform well on this task. We also test the behavior of the model when we resample the noise pattern at the test image such that adapter and test noise differ and accuracy decreases close to chance level (Figure 3.1h). We manually investigated feature maps in the first layer post-adaptation and noticed that activations are extreme and differ greatly from activations with repeated noise.
We conclude that the model didn't learn to generalize to different noise patterns but that it learned to fully subtract the first image which resulted in all noise being cancelled out independent of the pattern. This underlines the importance of a mechanistic investigation of models trained with gradient descent and shows that models can use components in unexpected ways. It showcases how modeling neural concepts using AI models can create an illusory sense of interpretability.
In contrast to additive suppression, the divisive model cannot simply subtract the noise. Instead, it could learn to zero out all features that were activated by the adapter image but this could heavily interfere with the image recognition objective which relies on stable nonzero representations. We repeat the experiments with this model and observe similar effects: $\alpha$, $\sigma$, and $K$ converge to values close to zero, indicating rapid and long-lasting suppression and accuracy decreases if we spatially shift the noise pattern at test step. However, when resampling the noise, accuracy only decreases moderately (Figure 3.1h) and much less compared to the additive suppression model, supporting our idea that there is a trade-off between suppressing features evoked by noise and keeping them responsive to serve the classification task.
Why exactly does this problem arise and how can we train more faithful adaptation models? In the brain, the amount of adaptation is an optimization problem: While adaptation might be helpful to adapt to persisting contrast, brightness, or noise settings, it can have negative side effects. For example, it can create perceptual biases where prolonged exposure to a specific type of stimulus affects the perception of subsequent stimuli. This happens as the sensory system becomes desensitized or attuned to certain features. For instance, if someone looks at a female face for an extended period and then views a gender-neutral face, the neutral face may appear more masculine [26]. Therefore, adaptation is tuned in strength, length, and maximum

for optimal visual perception over time. Our noise suppression task, however, only rewards suppression and there is no incentive to limit the amount of it. Hence, the dataset and task are too simplistic and do not sufficiently model the challenges that give rise to adaptation patterns in the brain.
In the subsequent chapter, we propose a methodology to address this issue. We introduce a more intricate task designed to more closely mimic the real-world challenges that lead to adaptation in the brain. Our findings demonstrate that models can indeed learn an adaptation process more akin to that observed in the human brain. We contend that to develop models that accurately replicate brain functions and circuits, it is crucial to train them with data that simulates the environmental and sensory conditions believed to drive the brain's learning of these functions. Rather than solely focusing on training models for specific tasks that are also used to test human capabilities, a more effective approach involves training models under conditions similar to human learning experiences. Subsequently, these models should be evaluated on the same tasks without prior specific training. This approach provides more robust evidence that the computations and behaviors we observe and study align with our understanding of their operation in the brain.

## 3.3 Stimulus-driven salience

When a novel or unexpected stimulus enters the visual field of humans or animals, attention is rapidly redirected towards it. This process is stimulus-driven and also called bottom-up attention because it operates on raw sensory input, is involuntary, and shifts attention towards salient visual features. It is used to filter out unimportant or interfering stimuli and is possibly driven by temporal divisive normalization: Static scenes and objects might get suppressed over time while a novel stimulus might evoke the full response, thus dominating the preexisting stimuli.
Here, we show that an artificial neural network augmented with additive suppression or divisive normalization capabilities can learn this effect when trained on a task that requires attending to novel stimuli. Then, we confirm mechanistically what the model learned and how it used its adaptation capabilities, and lastly, we investigate if brain-like patterns emerge that could form the connection between neural recordings and behavior.

### 3.3.1 Adaptation to static objects can drive saliency of novel stimuli

We trained the models on a novel object recognition task where the spatiotemporal model is fed with a sequence of images that contain one to four objects from fashion-mnist and has to predict the class of the novel object. If no new object was added, the model should keep predicting the most recently added object (Figure 3.2a). Thus, in theory, an adaptation model could solve this task by suppressing recurring input and then using the resulting scale to determine salience. We trained small CNNs that were augmented with an additive suppression, divisive normalization, or no adaptation mechanism after the first layer (Figure 3.2b).
Without adaptation, the model had a high accuracy of 72% when only one object was present but got quickly worse as more images were added. In contrast to this, both adaptation mechanisms were able to learn the task as accuracy barely dropped after adding novel objects (Figure 3.2c). Accuracy of divisive normalization was slightly higher than exponential decay. We then investigated the behavior of the model more closely: We constructed a test set that only contained a single fixed object at every position. While the baseline model without adaptation performed well throughout, accuracy of both adaptation models dropped after position 10 (Figure 3.2d) which might be explained by the learned suppression effect. Intuitively, sequences of 10 repeating images with no new objects added are rare in the training set, so the model has

little incentive to optimize these cases, and a strong suppression effect learned by the adaptation layer could explain the drop in accuracy. Remarkably, the accuracy of the divisive normalization model improved after seeing the same image multiple times but we found it difficult to map this to a specific property of the type of normalization.
Next, we tested the model's performance around the onset of a novel image. To do this, we made a test set that contained one object from the first timestep and a novel object that occurred at steps 2, 5, or 15. Without adaptation, accuracy dropped significantly as the model cannot know which of the two objects to predict. In contrast to this, accuracy of both adaptation models remained high and had little change which shows that both models utilize their recurrent adaptation mechanism to determine the novel object (Figure 3.2e).

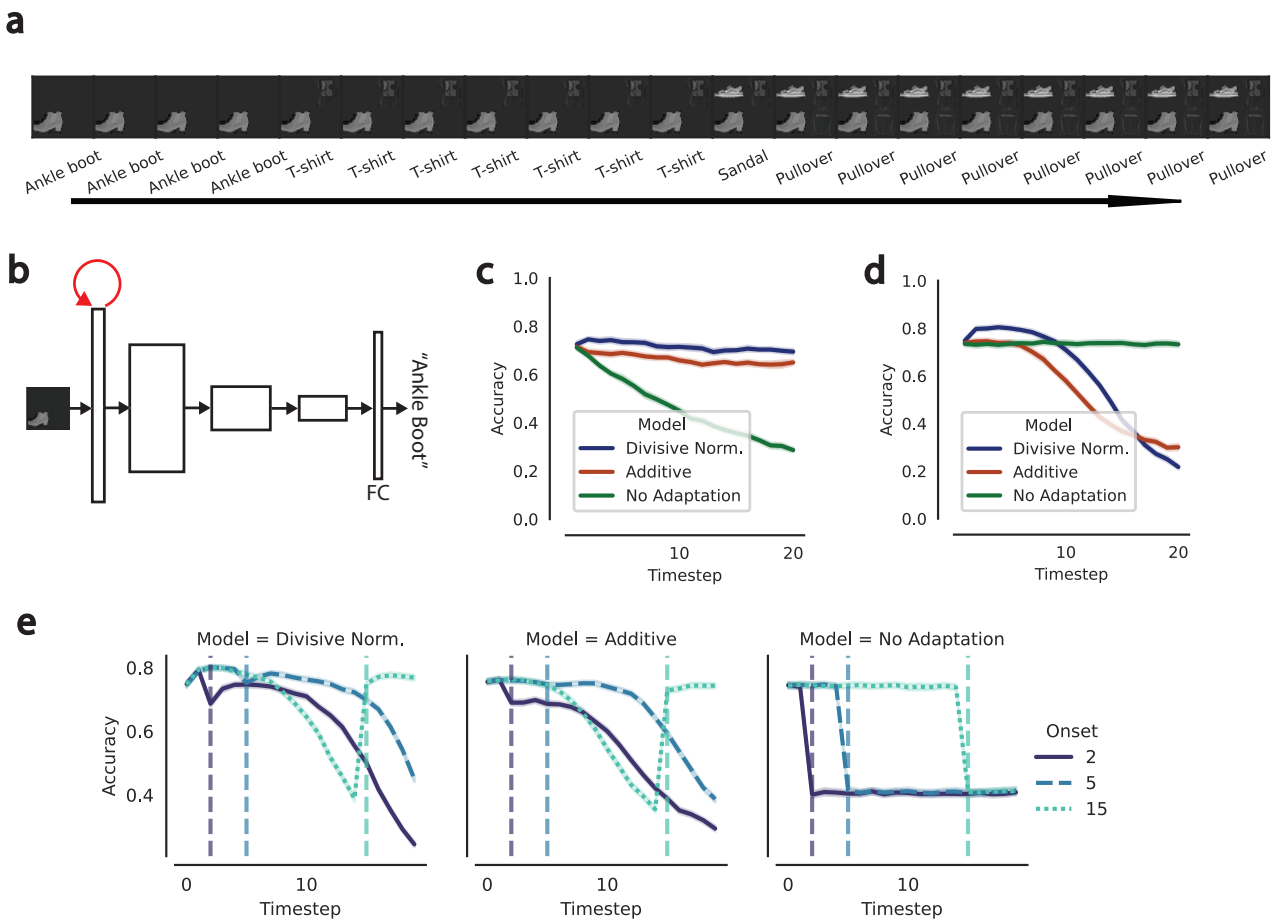



Figure 3.2: **CNNs augmented with adaptation can learn a novel object recognition task end-to-end.** **a)** An example sequence of a single training example with targets **b)** A trainable additive suppression or divisive normalization was added after the first convolution **c** Accuracy per timestep **d)** Accuracy per timestep on a test dataset that contained only one static object per sequence **e)** Accuracies on a test dataset that contained two objects per sequence with the second object appearing at onset.

### 3.3.2 Neuron magnitudes encode salience and is controlled by adaptation

What did the adaptation models learn? While we have an intuition about the implemented mechanisms, this is not necessarily what the model actually learned. Indeed, there are many examples where a model performed well but for a different reason than assumed (e.g. batch

normalization [22]) or where a model found a surprising way to solve a task (e.g. reward hacking in reinforcement learning [24]). In addition, it has been proposed that deep neural networks encode features in superposition where features are not necessarily basis-aligned and where a unit-wise suppression mechanism would not suppress features equally [11]. For this reason, we mechanistically investigated model behavior to show how they utilized the added adaptation layer.

We hypothesized that the models represent salience in each neuron's amplitude and that the adaptation mechanism reduces the amplitude. Indeed, we found that all feature maps decrease their activations over time when the input stays constant (Figure 3.3a). When tested on a single repeating object, activations of both adaptation models decreased but divisive normalization showed more aggressive adaptation. To test if this pattern replicates for multiple objects, we tested on a dataset comprised of two images where the second image occurs 2, 5, or 15 timesteps after the first. We recorded activations of layer 1 post adaptation and divided each feature map into each object's receptive field (Figure 3.3c left panel). Activations of the first image were high at onset and continuously decreased irrespective of the second object. However, activations of the second object's receptive field only deflected with onset, reached the same magnitude, and then started to decline while staying higher than activations of the first image (Figure 3.3c right panel).

Is the relationship between activation size and salience causal? To test this, we intervened on activations post-adaptation and changed each receptive field's activations to match the scale of the other one. We then measured the predicted probability of each object with and without intervention (Figure 3.3d). After intervention, the model's predictions were essentially swapped and the model assigned a higher probability to the first image which occurred earlier in context. Thus, we have shown that both adaptation models decrease activations on repeating input and that this is how the model encodes the novelty of objects.

### 3.3.3 Contrast contributes to neuron magnitudes

In the brain, contrast contributes to the salience of an object. This raises the question of how the adaptation models encode images of different contrasts and if that interferes with salience. Initially, we hypothesized that the model might learn different computational paths for images of different contrasts but showed in supplementary experiments that this is not the case. CNNs utilize the same feature maps for objects of one class but with vastly different contrasts. To investigate if contrast is represented by activation magnitude, we tested again on a dataset containing only a fixed object per sequence. Although accuracy is initially comparable between contrasts (Figure 3.3e), accuracy decreases faster over time for low-contrast objects, hinting at interference between contrast and salience. In addition, activation magnitudes scale with contrast (Figure 3.3f), so activations of low-contrast images vanish faster. Interestingly, there is a large difference in the range of values: while contrast directly translates to activation magnitude for divisive normalization and thus utilizes the range perfectly, differences in activations are much smaller for the additive suppression model. This hints at a possible advantage of divisive normalization: Adaptation scales effectively with magnitude while the additive suppression model subtracts values directly proportional to the previous activation scale which would lead to vastly different results if the range is too big. This behavior also replicates on objects added to preexisting ones (Figure 3.3g).

### 3.3.4 Adaptation effects are stronger in later layers

So far, we focused on tasks that are solvable with a single normalization layer. This was possible because images of an object stayed fixed for every timestep, so a single adaptation

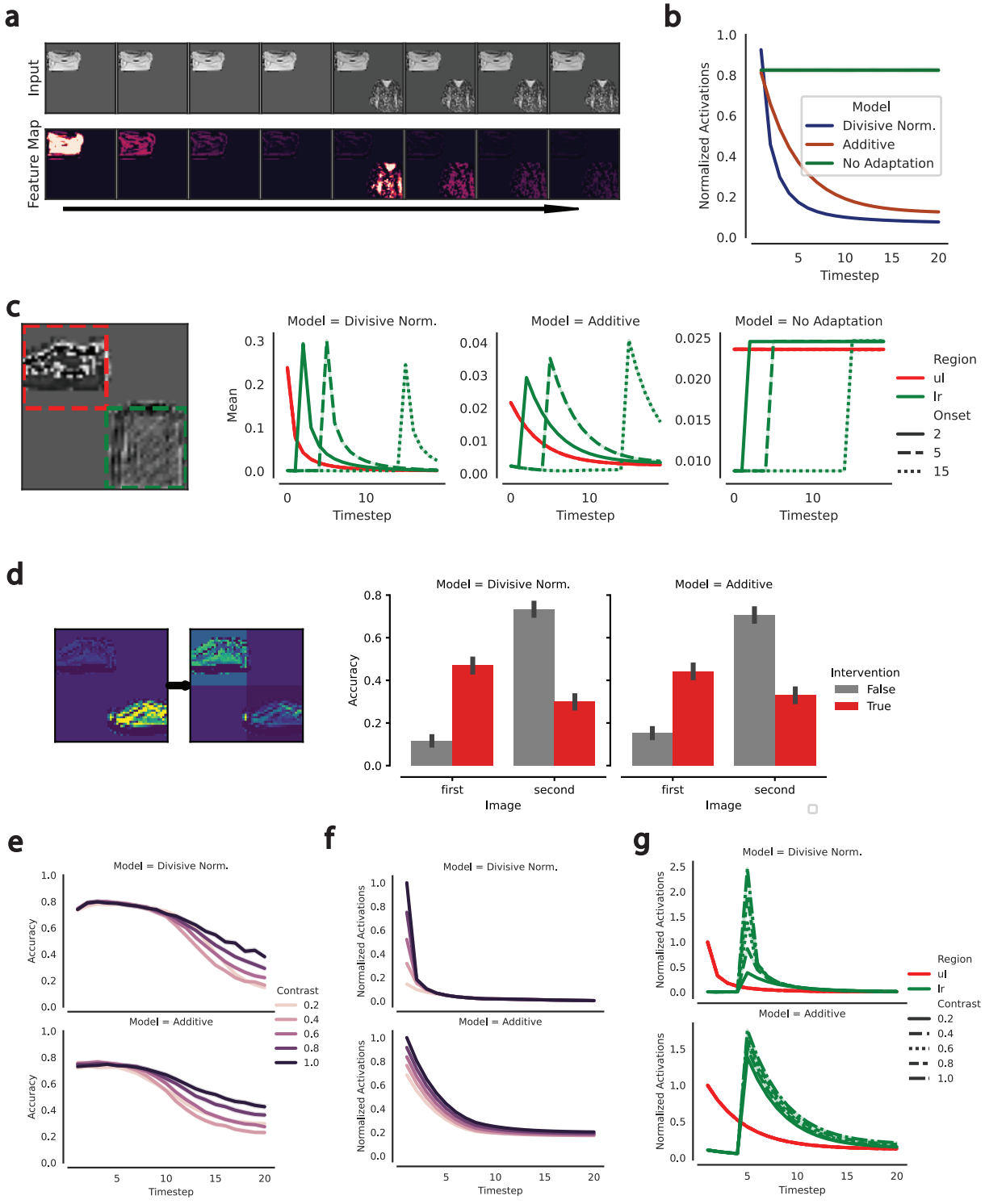



Figure 3.3: **Mechanistic Understanding of Adaptation a)** Activations of a representative feature map **b)** Activations normalized to first timestep at full contrast **c)** Left panel: Representative field for first (red) and second (green) image; Right panel: Activations by representative field and onset of the second image **d)** Left panel: Example intervention that swaps the scale of the representative fields; Right panel: Accuracy per object (grey) and accuracy after intervention (red) **e)** Accuracy by contrast on a test set containing only one static object **f)** Normalized Activations by contrast on a test set containing only one static object **g)** Normalized Activations by contrast and receptive field of the novel image on a test set containing two objects with the second (green) occurring later than the first (red)

layer was sufficient to pick up recurring inputs perfectly and suppress these. However, in the wild, a neuron is unlikely to receive the same inputs multiple times. Instead, visual stimuli are specked with small changes in noise, contrast, and brightness, or the location of the image shifts slightly. For example, a low-level edge detector's output in V! or the first convolution will be vastly different if the input image is moved just a tiny bit. In the human visual cortex, adaptation effects are greater in later layers and it is hypothesized that this is because later layers with larger receptive fields can integrate their input and "repair" the small deviations [4], [15].

Previously, Vinken et al [26] showed that normalization is greater in later layers in Alexnet augmented with additive adaptation. For example, in an oddball sequence, later layers showed a greater change for the deviant stimuli, and a logistic face classifier changed its prediction after adaptation more in later layers. However, they used pre-trained Alexnet with hardcoded adaptation parameters which were tuned to produce desired effects. Thus, their observations are a direct result of their paradigm and they didn't show if and how a model trained with adaptation could benefit from this mechanism.

Here, we aim to close the gap between low-level circuits and the resulting capabilities and show that an adaptation model does indeed develop greater adaptation in later layers. To mimic real-world scenarios better, we added various sources of noise to the images, for example, shifts, zoom, shears, rotations, contrast, and Gaussian noise (Figure 3.4a). Because inputs to the same units in the first layer can now vary between positions, this task is not solvable with a single normalization layer. Instead, we now introduce adaptation with separate learnable parameters after every convolution (Figure 3.4b). The divisive normalization and additive suppression models are able to learn the task well with stable accuracy across positions (Figure 3.4c). Figure 3.4d shows that the models learn to suppress previously shown images. To assess the degree of adaptation, we record activation magnitudes in every layer (Figure 3.4e). As hypothesized, we find that later layers exert a higher relative drop in magnitudes, suggesting that adaptation builds up across layers.

But is this effect a consequence of adaptation acting in different layers or did the convolutions simply learn to scale down small input even further without using additional normalization? To test this causally, we scaled activations of a given layer L to be constant across timesteps. Then, we recorded post-layer L + 1 to see if there is suppression occurring in layer L + 1 (Figure 3.4g). Interestingly, we found that this is the case for layers 0 and 1, supporting the hypothesis that task-driven models can utilize normalization across multiple layers. However, layer 2 and 3 effectively didn't utilize their normalization capability. For divisive normalization, no temporal adaptation is occurring in these layers, and for additive suppression, activations in both layers increase.

Thus, normalization across layers can solve complex tasks by chaining suppression although not every layer is effectively utilized given our current training setup.

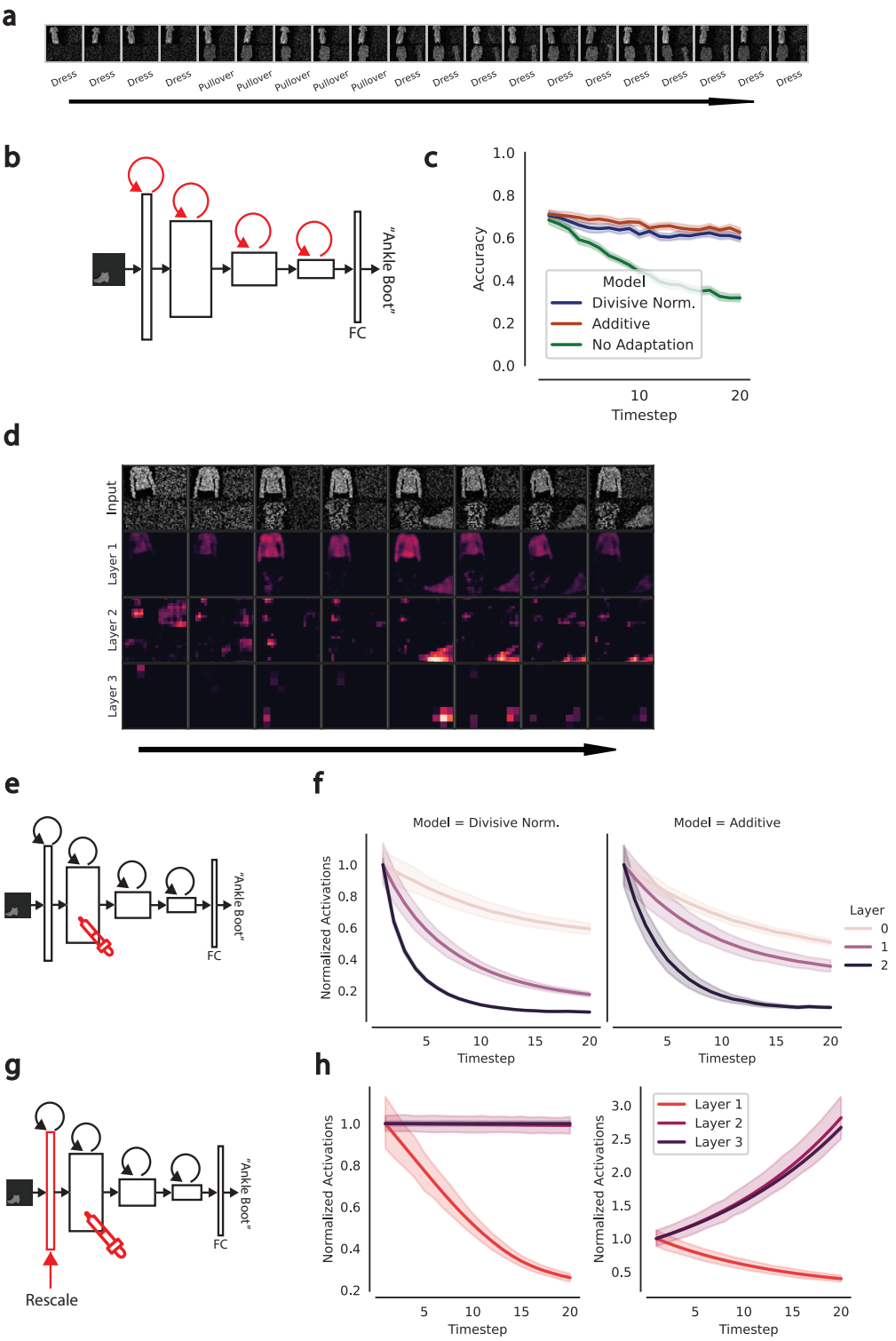



Figure 3.4: **Adaptation across multiple layers can learn more complex tasks a)** Representative training sample containing various sources of noise and image augmentations **b)** Trainable additive suppression or divisive normalization was introduced after every convolution **c)** Accuracy per timestep **d)** A random input sample with activations from one representative feature map per layer **e)** Activations in all layers were recorded **f)** Normalized activations by layer on a dataset containing a fixed object which is augmented across positions **g)** Activation magnitudes were fixed in a specific layer and activations were recorded after the following layer **h)** Normalized activations after intervention (g). Labels denote the layer of measurement.

# Chapter 4

# Discussion

This study was initiated to identify and understand canonical computations in the brain, which are fundamental operations widely utilized in neural processing. Canonical computations are essential not only for processing stimuli but also for influencing and shaping behavior. Our research specifically focused on temporal adaptation, a mechanism hypothesized to be prevalent across various neural regions. This form of adaptation involves neuronal responses adjusting dependent on time from lateral or recurrent afferents, or from itself.

In particular, we concentrated on self-referential temporal adaptation, where a neuron normalizes its activity based on historical responses. This study explored two approaches: additive and divisive normalization. The additive model, aligning with the propositions in [26], suggests an intrinsic mechanism within neurons for suppression, independent of external connections. On the other hand, divisive normalization, traditionally associated with lateral or recurrent neuronal interactions, was implemented in our study to normalize based solely on a neuron's historical activity. This implementation points to the potential for such mechanisms to be either an intrinsic attribute of individual neurons or mediated through recurrent network connections. Our investigation into these forms of temporal adaptation aimed to contribute to a more comprehensive understanding of how neurons can utilize their historical activity for processing inputs, potentially serving as a model for more complex neural circuitry and behavior.

In this study, we have made significant strides in understanding and implementing adaptation mechanisms within convolutional neural networks (CNNs). We successfully integrated both additive and divisive adaptation mechanisms into CNNs, targeting a noise suppression task designed to enhance the signal-to-noise ratio by suppressing static noise. Although the models learned this task effectively, our analysis revealed that they primarily relied on subtracting the noise. This finding suggests that the models do not fully emulate processes similar to those in the visual cortex, highlighting a divergence in their approach. One of the key insights from our study is the potency of gradient descent optimization. We observed that gradient descent often finds unexpected solutions, diverging from our initial hypotheses about how specific layers would function. This outcome underscores the importance of employing mechanistic interpretability methods to unravel the actual operations within neural network layers.

Furthermore, we emphasize the necessity of training models in scenarios that closely resemble real-world conditions, particularly when the goal is to replicate specific brain functions. This approach ensures that the training encompasses the relevant challenges and complexities inherent in the targeted neural operations. To this end, we developed a novel object recognition task where models were tasked with identifying the most recently added object among multiple items. This task required the models to balance the level of suppression - enough to distinguish between new and old elements, but not so much as to lose track of existing information. A detailed investigation into the models' internal mechanisms revealed that they use activation magnitude to encode both contrast and salience. Additionally, we showed that employing mul-

tiple layers of normalization in a model can be effective for complex tasks where single-layer approaches fall short. Overall, these contributions provide a deeper understanding of how adaptation mechanisms can be incorporated into artificial neural networks, enhancing their ability to learn and process information in a brain-like manner.

**Utilization of multiple adaptation layers:** We observed that the model trained on the novel object task does not effectively use its last two convolutional layers for divisive normalization although we would expect such a behavior in the brain. We think this is likely because it allocates these layers for other functions, such as rescaling. The more intriguing question, however, concerns the brain's processing: while we observe normalization in higher brain areas like the middle temporal (MT) visual area, it remains unclear if this stems from lower-level cortices or if higher areas possess their own distinct normalization mechanisms. Our model displays a pronounced normalization effect in its third layer, even in the absence of apparent normalization activity. This might occur if the convolutions inherently scale the input about its magnitude. Observing this in our model suggests the possibility of the brain employing a similar strategy. It seems plausible that higher brain areas could have independent normalization processes to facilitate adaptation to complex stimuli, such as recognizing varying images of the same individual or processing diverse auditory inputs. Yet, the existence of such mechanisms should not be hastily concluded from incomplete experimental data. A more rigorous investigation is necessary to substantiate these hypotheses.

**Choosing the dataset:** The training data are arguably the most important ingredient for training AI models. This is often underappreciated as data collection is expensive and time-consuming while often not directly delivering novel scientific findings. Here, we need to choose between training on toy datasets with simple and engineered data, or information-dense real-world datasets, in our case large video databases. While training on data that is similar to what humans see might give more comparable and more similar results, these models are inherently messy and difficult to interpret and test. Toy datasets, however, are readily interpretable but if they lack crucial complexity, the resulting models might not capture the desired effect, as seen in this study. For temporal adaptation specifically, datasets must at least have the following complexities:

1. **Advantageous Adaptation:** Adapting must provide information that is valuable with respect to the task the network is trained on. This ensures that the adaptation mechanism contributes positively to the model's performance.

2. **Resensitization:** There must be an advantage to sensitize the model again after a certain period. Without this, the model may converge to low values of $\alpha$, $K$, and $\sigma$ (and high $\beta$ in the case of additive adaptation), leading to complete suppression without recovery.

3. **Restriction of Adaptation:** It is crucial to limit the extent of adaptation. In our model, this was implemented by requiring the network to continue outputting the latest object, thereby incentivizing the model to restrict the degree of suppression. Without such a restriction, the model might learn to achieve complete suppression quickly.

4. **Layer-Specific Adaptation in Complex Datasets:** When applying adaptation in multiple layers of a CNN, the dataset needs to be complex enough to necessitate layer-specific adaptation. Early layers typically learn simple features like edges or textures, while later layers learn complex features such as faces or objects. Adaptation in early layers is beneficial if the stimulus to be suppressed is detectable at that level. Conversely, learning adaptation in the final layers is advantageous if it provides new information that earlier layers could not discern.

In this study, we showed how a too shallow dataset can collapse adaptation layers and potentially lead to an illusory sense of understanding. Then, we constructed a novel dataset that

contained these challenges and showed that the resulting models have brain-like attributes. However, while there is a strong incentive to limit the amount of adaptation, the incentive to resensitize after a stimulus disappears might not be strong enough because objects in the image aren't overlapping. To compensate for this, we should consider longer datasets with objects disappearing and other objects appearing at the same location. This would create a real incentive for the model to optimize between continuing or lifting the suppression and should lead to more interesting models. For example, with such a model, it would be possible to study aftereffects, like the adapter class biasing the prediction for the test object. It would also be possible to add noise or set contrast in a way that correlates between subsequent timesteps. This could lead to response enhancement by suppressing noisy repeating stimuli.

# Bibliography


[1] Duane G Albrecht and David B Hamilton. Striate cortex of monkey and cat: contrast response function. *Journal of neurophysiology*, 48(1):217–237, 1982.

[2] Yelysei Bondarenko, Markus Nagel, and Tijmen Blankevoort. Quantizable transformers: Removing outliers by helping attention heads do nothing. *arXiv preprint arXiv:2306.12929*, 2023.

[3] AB Bonds. Role of inhibition in the specification of orientation selectivity of cells in the cat striate cortex. *Visual neuroscience*, 2(1):41–55, 1989.

[4] Amber M Brands, Sasha Devore, Orrin Devinsky, Werner Doyle, Adeen Flinker, Daniel Friedman, Patricia Dugan, Jonathan Winawer, and Iris IA Groen. Temporal dynamics of neural adaptation across human visual cortex. *bioRxiv*, pages 2023–09, 2023.

[5] Paul Broca et al. Perte de la parole, ramollissement chronique et destruction partielle du lobe antérieur gauche du cerveau. *Bull Soc Anthropol*, 2(1):235–238, 1861.

[6] Max F Burg, Santiago A Cadena, George H Denfield, Edgar Y Walker, Andreas S Tolias, Matthias Bethge, and Alexander S Ecker. Learning divisive normalization in primary visual cortex. *PLOS Computational Biology*, 17(6):e1009028, 2021.

[7] Laura Busse, Alex R Wade, and Matteo Carandini. Representation of concurrent stimuli by population activity in visual cortex. *Neuron*, 64(6):931–942, 2009.

[8] Matteo Carandini and David J Heeger. Normalization as a canonical neural computation. *Nature Reviews Neuroscience*, 13(1):51–62, 2012.

[9] Tim Dettmers, Mike Lewis, Younes Belkada, and Luke Zettlemoyer. Llm. int8 (): 8-bit matrix multiplication for transformers at scale. *arXiv preprint arXiv:2208.07339*, 2022.

[10] Nelson Elhage, Robert Lasenby, and Christopher Olah. Privileged bases in the transformer residual stream. *Transformer Circuits Thread*, 2023.

[11] Nelson Elhage, Neel Nanda, Catherine Olsson, Tom Henighan, Nicholas Joseph, Ben Mann, Amanda Askell, Yuntao Bai, Anna Chen, Tom Conerly, Nova DasSarma, Dawn Drain, Deep Ganguli, Zac Hatfield-Dodds, Danny Hernandez, Andy Jones, Jackson Kernion, Liane Lovitt, Kamal Ndousse, Dario Amodei, Tom Brown, Jack Clark, Jared Kaplan, Sam McCandlish, and Chris Olah. A mathematical framework for transformer circuits. *Transformer Circuits Thread*, 2021. https://transformer-circuits.pub/2021/framework/index.html.

[12] Robert Geirhos, Roland S Zimmermann, Blair Bilodeau, Wieland Brendel, and Been Kim. Don't trust your eyes: on the (un) reliability of feature visualizations. *arXiv preprint arXiv:2306.04719*, 2023.

[13] David J Heeger. Normalization of cell responses in cat striate cortex. *Visual neuroscience*, 9(2):181–197, 1992.

[14] David J Heeger. Modeling simple-cell direction selectivity with normalized, half-squared, linear operators. *Journal of neurophysiology*, 70(5):1885–1898, 1993.

[15] Kevin D Himberger, Hsiang-Yun Chien, and Christopher J Honey. Principles of temporal processing across the cortical hierarchy. *Neuroscience*, 389:161–174, 2018.

[16] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. *Advances in neural information processing systems*, 25, 2012.

[17] Kevin Meng, David Bau, Alex Andonian, and Yonatan Belinkov. Locating and editing factual associations in gpt. *Advances in Neural Information Processing Systems*, 35:17359–17372, 2022.

[18] Richard A Normann and I Perlman. The effects of background illumination on the photoresponses of red and green cones. *The Journal of Physiology*, 286(1):491–507, 1979.

[19] Catherine Olsson, Nelson Elhage, Neel Nanda, Nicholas Joseph, Nova DasSarma, Tom Henighan, Ben Mann, Amanda Askell, Yuntao Bai, Anna Chen, et al. In-context learning and induction heads. *arXiv preprint arXiv:2209.11895*, 2022.

[20] John H Reynolds and Robert Desimone. Interacting roles of attention and visual salience in v4. *Neuron*, 37(5):853–863, 2003.

[21] Fred Rieke and Michael E Rudd. The challenges natural images pose for visual adaptation. *Neuron*, 64(5):605–616, 2009.

[22] Shibani Santurkar, Dimitris Tsipras, Andrew Ilyas, and Aleksander Madry. How does batch normalization help optimization? *Advances in neural information processing systems*, 31, 2018.

[23] Eero P Simoncelli and David J Heeger. A model of neuronal responses in visual area mt. *Vision research*, 38(5):743–761, 1998.

[24] Joar Skalse, Nikolaus Howe, Dmitrii Krasheninnikov, and David Krueger. Defining and characterizing reward gaming. *Advances in Neural Information Processing Systems*, 35:9460–9471, 2022.

[25] Larry R Squire. The legacy of patient hm for neuroscience. *Neuron*, 61(1):6–9, 2009.

[26] K Vinken, X Boix, and G Kreiman. Incorporating intrinsic suppression in deep neural networks captures dynamics of adaptation in neurophysiology and perception. *Science Advances*, 6(42):eabd4205, 2020.

[27] Kevin Wang, Alexandre Variengien, Arthur Conmy, Buck Shlegeris, and Jacob Steinhardt. Interpretability in the wild: a circuit for indirect object identification in gpt-2 small. *arXiv preprint arXiv:2211.00593*, 2022.

[28] Davide Zoccolan, Minjoon Kouh, Tomaso Poggio, and James J DiCarlo. Trade-off between object selectivity and tolerance in monkey inferotemporal cortex. *Journal of Neuroscience*, 27(45):12292–12307, 2007.