Optimising Visual Input for Cortical Neurons: a CNN-based MEI Approach Compared to Gabor Patches

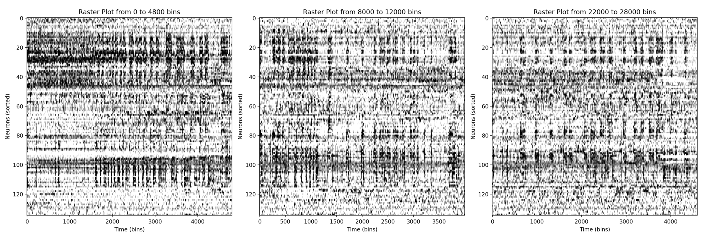
Agnese Adorante (3158575), Martina Del Gaudio (3174092), Tommaso Giacomello (3173616), Kristian Gjika (3116656), Clara Montemurro (3179680)

INTRODUCTION

This research is committed to employing deep learning techniques to identify and generate the Most Excitatory Input (MEI) for a single neuron in the visual system of a mouse, specifically through the use of end-to-end trained convolutional neural networks (CNNs). **Most Excitatory Inputs** (MEIs) represent the specific visual stimuli that are designed to maximally activate the neurons. These in silico stimuli are pivotal for delineating the functional characteristics of a neuronal single-cell, highlighting the features within a neuron's receptive field to which it is most responsive.
By adopting this methodology, we aim to construct a stimulus that, by successfully combining the stimuli's features, is most likely to trigger the highest firing rates in neurons. Finally, we seek to investigate some properties of these artificially optimized stimuli and try to test the robustness of our findings using Gabor filters as benchmarks.

Most Important Neurons (MINs)

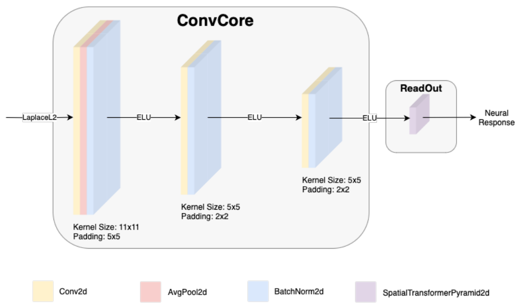
We begin with a preliminary analysis of our dataset, by constructing a RasterMap of the (z-scored) neural activity. There are three rastermaps, which correspond to the 3 time slots in which the mice has been shown natural scenes.



We immediately recognize that there are indeed some neurons which present a pattern behaviour in neural response, which indicates that there are good possibilities to select a neuron which will work well with our CNN. However some neurons have a noisy response, hence the correct selection of neurons is of utmost importance. To understand the neurons' response, we analyzed several aspects of their activity: correlation, entropy, and variance of spike count across images. Additionally, we employed an unsupervised anomaly detection algorithm called Isolation Forest. In order to select the most informative neurons, we developed an oracle correlation method which analyses a neuron's response across different images. It compares a random response to the average response on other trials for the same image and penalises neurons with limited image data. To ensure reliability, we developed a second function which runs this process 50 times (i.e., mean number of trials) and calculates average scores. We focused on high-performing neurons with strong average oracle correlations (above 60th percentile) and high variance (above 50th percentile). The neurons amongst these which will be chosen are those which perform best with the developed CNN. It is interesting to note, that when plotting oracle correlation against entropy, there is an almost-exponential behaviour in the datapoints.

Convolutional Neural Network (CNN)

We utilized a Convolutional Neural Network (CNN) to predict neuronal responses. Recent research has solidified deep CNNs as leading models in this domain, prompting us to develop our own inspired by prior studies. The core of the network consists of 3 blocks, and is followed by a neuron-specific readout layer. Specifically, our network comprises the following components.



Core block

1. A convolutional layer with a kernel size tailored to match available Gabor filters, facilitating an effective comparison between stimuli.
2. A batch normalization layer, essential for preventing overfitting by standardizing and normalizing the inputs of each layer.
3. A final non-linearity activation, fundamental to model the intrinsic non-linear nature of the neuronal responses. Specifically, we used the Exponential Linear Unit (ELU) function, defined as:

$$\mathrm{ELU}(x) = \begin{cases} x & x \geq 0 \\ e^x - 1 & x < 0. \end{cases}$$

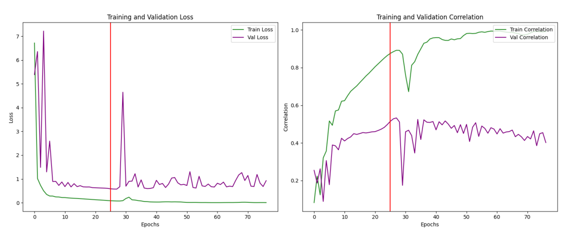
4. A pooling layer on the first block only, positioned before the non-linearity activation to expedite computations and enhance performance. Pooling layers help in downsampling feature maps, thereby reducing dimensionality while retaining essential information.

Readout Layer

The readout layer synthesizes the features extracted by the CNN into a single scalar representing the average spike activity of a neuron. This layer is modeled as an affine transformation of the features, followed by a non-linearity. To ensure the output remains positive, the activation function employed is $ELU(x) + 1$. For further insights into the entire network structure, including the readout layer, refer to Walker et al (Nat. Neurosc. 2019).
This approach encapsulates the essence of the network's functionality, providing a mechanism for interpreting neuron excitations through deep predictive models.

Results

After identifying neurons through oracle correlations, we conducted further classification based on their performance in our CNN. We retained only those neurons that exhibited high performance in this subsequent classification process. During training, we leveraged two key metrics: loss and correlation values. Our highest-performing neuron attained a validation correlation of 62%. To illustrate the distribution of scores, refer to the accompanying figure below.



Most Excitatory Input (MEI)

Currently, we have identified a suitable neuron and constructed an effective model that provides neural responses. The remaining task is to employ an algorithm capable of generating the Most Excitatory Input (MEI) based on these architectures. Here the pseudo code is provided:

Algorithm 1: MEI generation code

$I_{MEI} \leftarrow$ White Noise Image;
$d_c \leftarrow$ Decrease Constant ;
$n_{steps} = 1000$;
$\sigma \leftarrow 7$;
for $i \leftarrow 0$ **to** n_{steps} **do**
  # Gradients computation wrt image pixels;
  grads $\leftarrow$ grad($CNN(I_{MEI})$);
  #Fourier Transform Smoothing;
  fftgrad $\leftarrow$ fftsmooth(grads);
  # Gradient Ascent Step;
  $I_{MEI} \leftarrow I_{MEI} + \alpha \cdot$ fftgrad;
  # Blur the image with a Gaussian Filter;
  $I_{MEI} \leftarrow gaussblur(I_{MEI}, \sigma)$;
  # Decrease σ to reduce the blurring;
  $\sigma \leftarrow \sigma - d_c$;
  # Scale the image in[0,1];
  $I_{MEI} \leftarrow scale(I_{MEI}, 0, 1)$
end for
return I_{MEI};

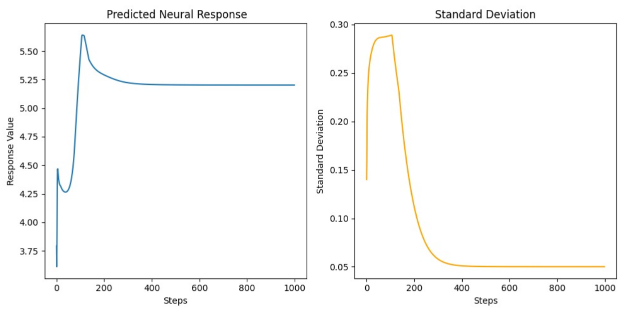
More formally, we start by generating a white noise image distributed as $X \sim N(0, 1)$. Subsequently, we predict the neural response of our image with our CNN. Then, we compute the gradients with respect to each pixel of the image keeping a regularization term to avoid contrast imbalances:

$$(\frac{\partial r}{\partial X})_{i,j} \rightarrow \frac{\partial r}{\partial X_{i,j}} \rightarrow \frac{\partial r - \lambda \cdot |c^* - c|}{\partial X_{i,j}}$$

Where:

- $c^* =$ "optimal standard deviation"
- $c =$ "standard deviation of current image"

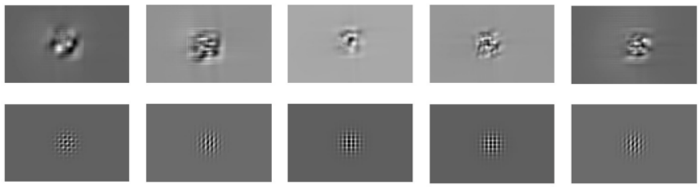
Furthermore, we smooth our computed gradients with a Fast Fourier Transform in order to suppress high frequency noise and blend it with our original image, emphasizing high-gradient areas in a gradient ascent step. Using a Gaussian Filter, we ensure clarity by gradually reducing blurring. Finally, we scale the image and restart the loop.
Analysing how the neural response and the standard deviation change across step confirm the robustness of our algorithm. In fact, both neural response and standard deviation stabilize to a value after following respectively an increasing and decreasing trend. This tell us that the refinement worked without diverging.



Gabor Comparison

The last step of our analysis is to compare our results with Gabor filters. **Gabor filters** are mathematical models that mimic the response of neurons in the mammalian visual cortex to specific visual stimuli. We extracted the gabor filter maximizing the chosen neuron's activation by merging the first 3 gabors that triggered the highest spike count.
We take four key metrices into account:

1. **Luminance:** average pixel intensity within the spatial mask
2. **Contrast:** standard deviation within the spatial mask
3. **1 Fold Symmetry Index:** mimicked by the folding of the image over one axis
4. **2 Fold Symmetry Index:** mimicked by the folding of the image over two axis



Despite the seemingly diverse structure both Gabors and MEI share similar values of the metrics we compute.

	MEI	Gabor
Luminance	0.4496	0.4097
Contrast	0.1511	0.1323
1-Fold Symmetry Index	0.9696	0.9992
2-Fold Symmetry Index	0.9845	0.9975

Conclusions and Limitations of our research

Our analysis investigated the comparative efficacy of Most Excitatory Inputs (MEIs) and Gabor filters in maximizing neuronal responses for specific neurons. While MEIs are generally acknowledged as superior to Gabors, a direct comparison remains crucial for validating MEIs' specific advantages. Additionally, we want to investigate whether MEIs, optimized for visual stimuli response, exhibit parallels with Gabors, designed to evoke maximal responses in V1 neurons of the visual cortex. Our inquiry aimed to uncover shared characteristics between the two. Our findings reveal similarities between MEIs and Gabor patches, affirming the presence of common traits in driving neuronal response. Notably, research by Walker et al. (2019) suggests that MEIs elicit a heightened response compared to Gabor patches among V1 cortex neurons. However, it's essential to underscore that such analysis necessitates in vivo validation.
Our study faced limitations including a small dataset of recorded neurons and natural images, which likely affected model generalization. Deep neural networks may struggle to extend beyond training data statistics, raising uncertainty about in vivo response predictions. Computational constraints led to image resizing, compromising model ability to discern details. Gabor filter parameters were limited, hindering MEI analysis precision. Future work could explore diverse cortical regions, different stimuli, while expanding to predict responses across multiple neurons for improved simulation sophistication.

References

[1] W. et al, "Inception loops discover what excites neurons most using deep predictive models", *Nature Neuroscience*, 2019.

[2] T. et al, "The feature landscape of visual cortex", *bioRxiv*, 2023.