Lecture Notes for

Neural Networks and Machine Learning

CNN Circuits Continued

Logistics and Agenda

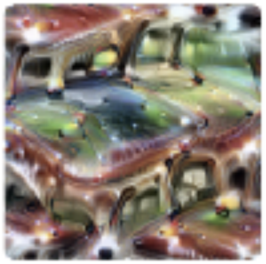
- Logistics
    - Grading Update
- Agenda
    - Last Time: Circuits in CNNs
    - Continued Circuits
    - Lab Three Town Hall
    - Next Time:
        - Student Paper Presentation
        - Fully Convolutional Networks

Last Time

- *Features are connected by weights, forming circuits*
- *"All neurons in our network are formed from linear combinations of neurons in the previous layer, followed by ReLU. If we can understand the features in both layers, shouldn't we also be able to understand the connections between them?"*
- *"Once you understand what features they're connecting together… You can literally read meaningful algorithms off of the weights."*

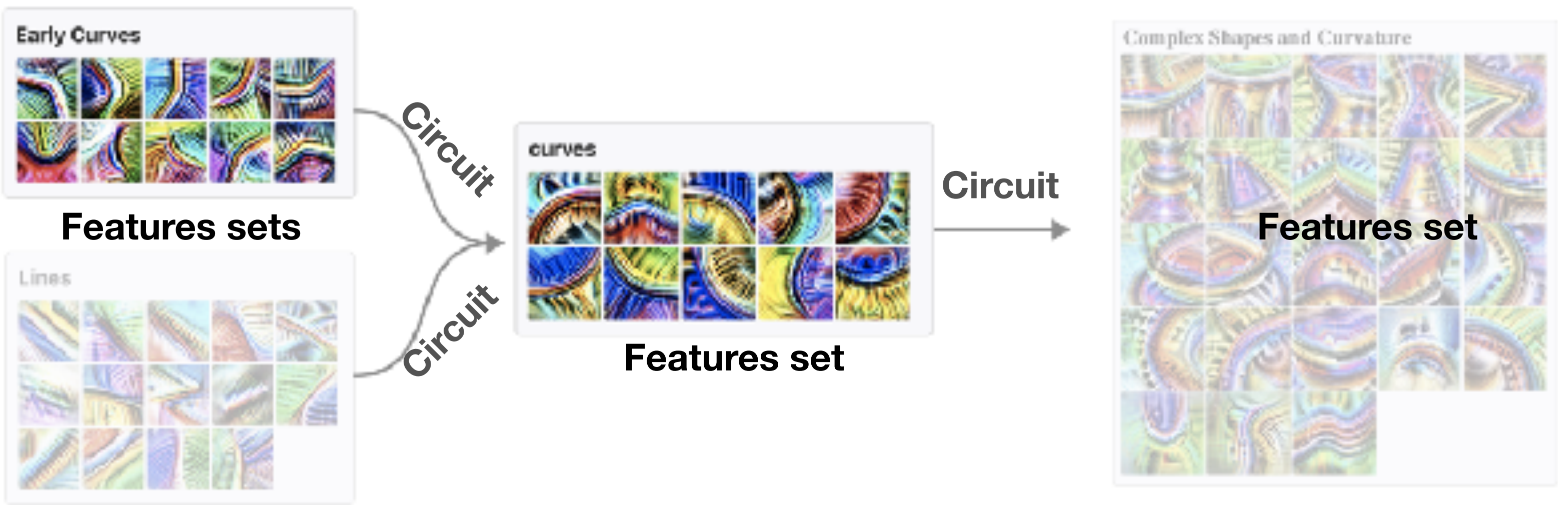
Neuron 4b:409

Dataset examples for neuron 4b:409



From Features to Circuits

- *Features are connected by weights, forming circuits*
- *"All neurons in our network are formed from linear combinations of neurons in the previous layer, followed by ReLU. If we can understand the features in both layers, shouldn't we also be able to understand the connections between them?"*
- *"Once you understand what features they're connecting together… You can literally read meaningful algorithms off of the weights."*



https://microscope.openai.com/models/inceptionv1/

Review of CNN Structure

```
Model: "vgg16"
Layer (type)              Output Shape              Param #
input_1 (InputLayer)      [(None, None, None, 3)]   0
```

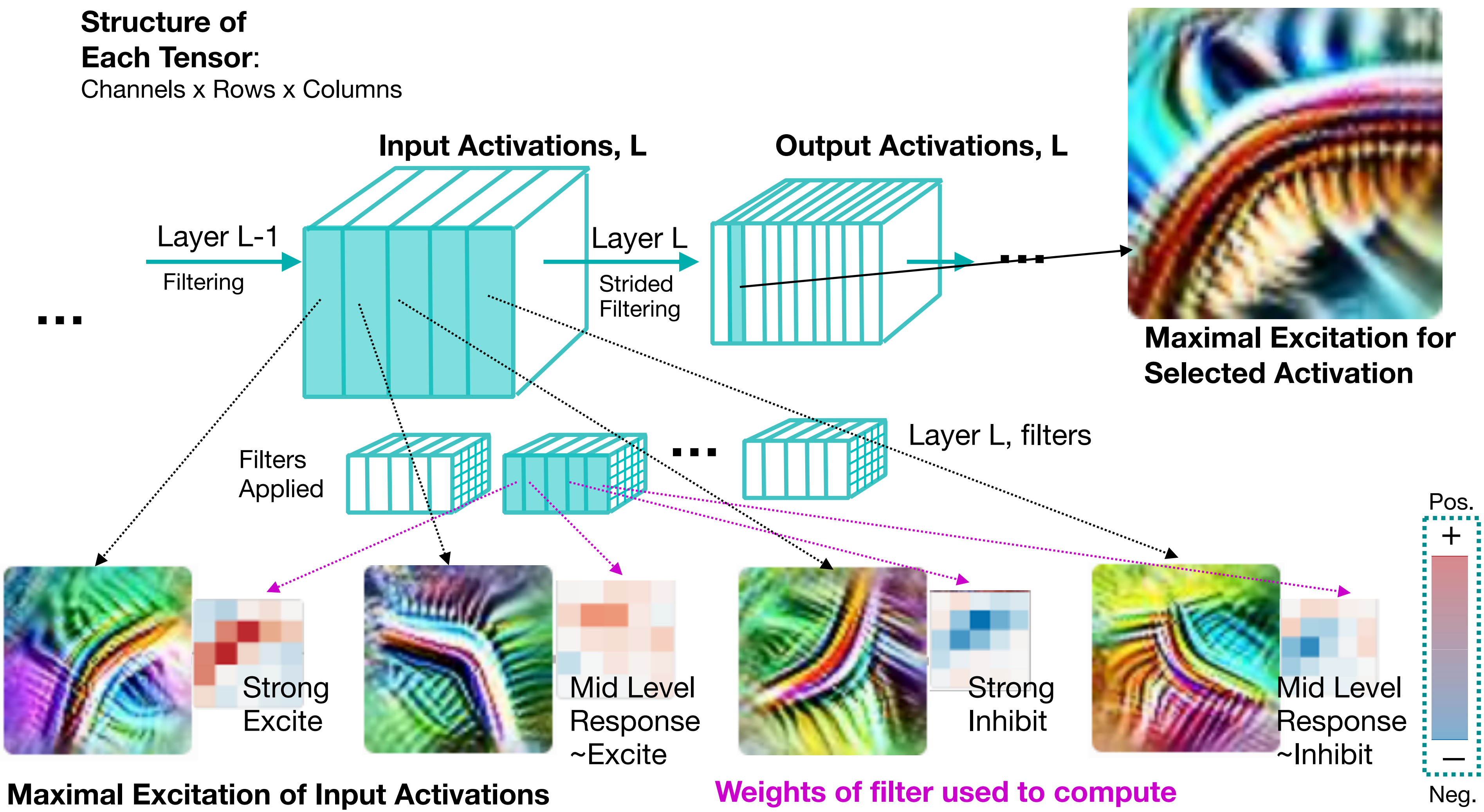
```python
#  lets look at the shapes of some of the filters above
keras_layer = model.get_layer('block3_conv1')
layer_output = keras_layer.output
weights_list = keras_layer.get_weights() # list of filter, the biases
filters = weights_list[0]
biases = weights_list[1]

# print out some specifics of how the filter is saved
print('block4_conv1 activation size is ', layer_output.get_shape(), '(batch x H x W x
print('block4_conv1 filters is of shape',filters.shape, '...(k x k x channels x filter
print('block4_conv1 biases is of shape',biases.shape)

idx = 32
print('one filter in block4_conv1 is ', filters[:,:,:,idx].shape )
channel = 2
print('one channel in the the filter is', filters[:,:,channel,idx].shape)
print('The weights of that channel in the filter are:\n', filters[:,:,channel,idx])
print('The bias of the filter is:',biases[idx])
```

```
block4_conv1 activation size is  (None, None, None, 256) (batch x H x W x filter)
block4_conv1 filters is of shape (3, 3, 128, 256) ...(k x k x channels x filters)
block4_conv1 biases is of shape (256,)
one filter in block4_conv1 is  (3, 3, 128)
one channel in the the filter is (3, 3)
The weights of that channel in the filter are:
 [[-0.03330493  0.01174345  0.03184387]
 [-0.04050588 -0.02253938  0.02304637]
 [-0.00191393 -0.01501364  0.02783429]]
The bias of the filter is: 0.030420048
```

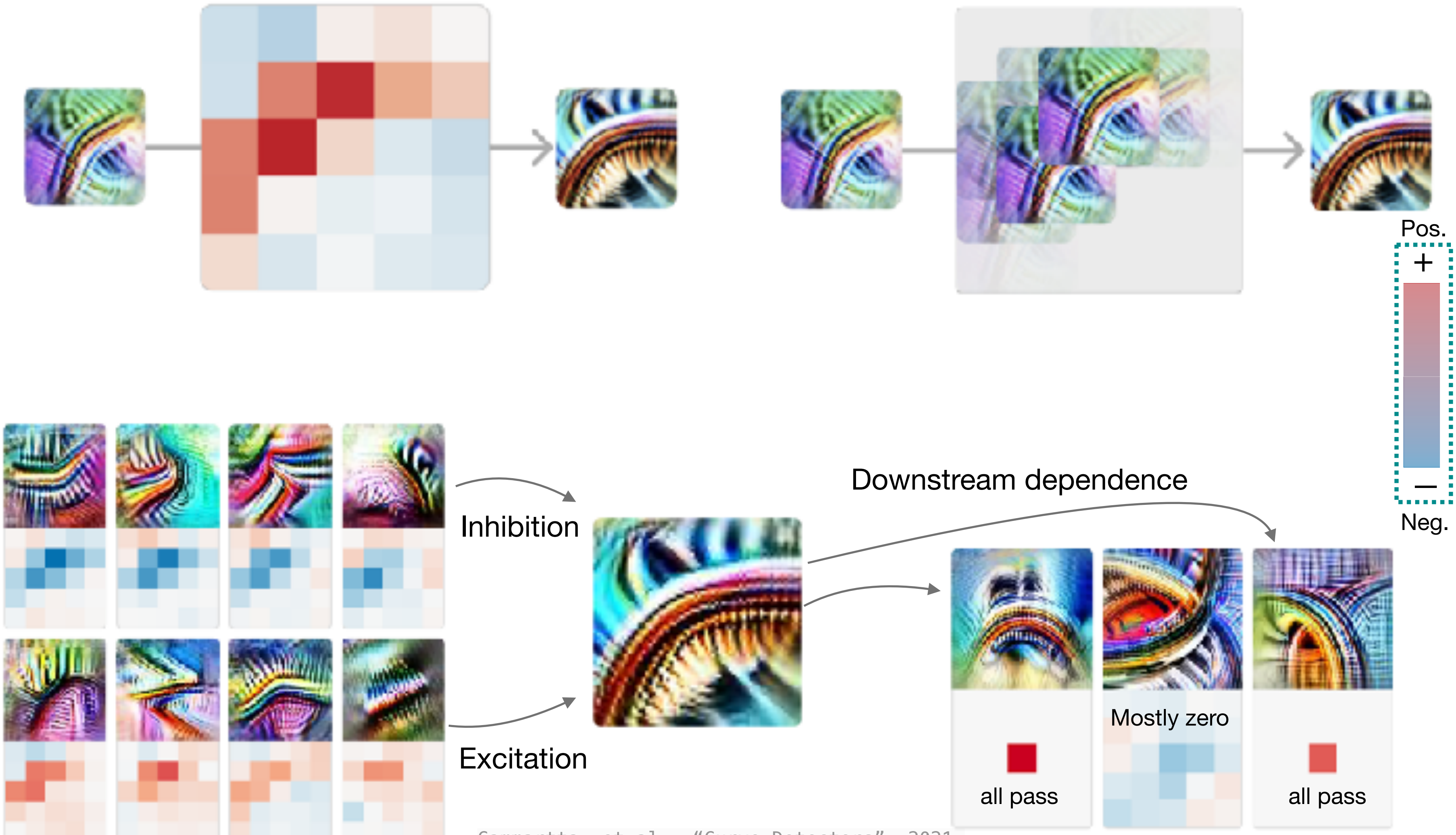
What weights comprise a circuit?



https://distill.pub/2020/circuits/curve-circuits/

Example: Circuit for Better Curve Detection

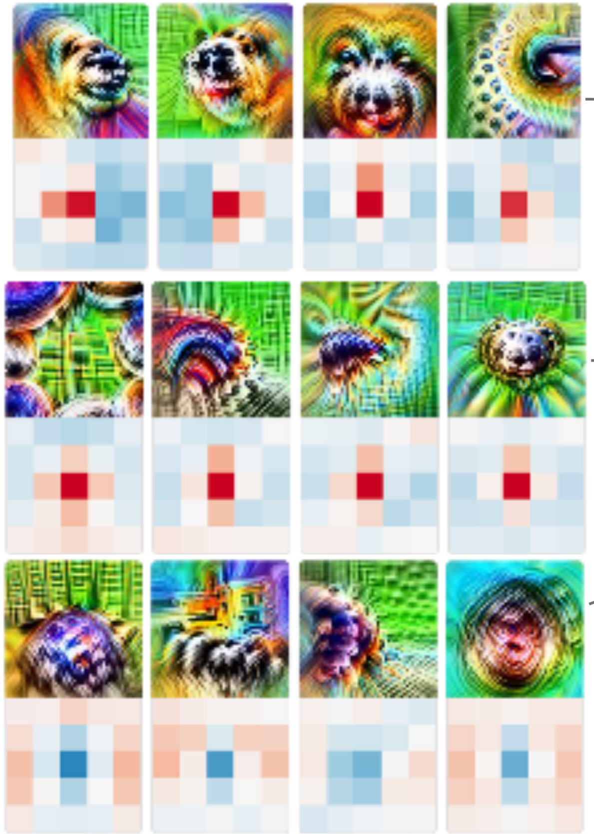
If we visualize the 5x5 Conv Filter, we can see that this becomes a Superposition of Early Curves



Cammartta, et al., “Curve Detectors”, 2021.

Another Example: Dog head

Compact Circuit Visualization



This example is also **polysemantic** due to the “**espresso maker**” class also being excited by this…

Olah, et al., “Naturally Occurring Equivariance in NN”, 2021.

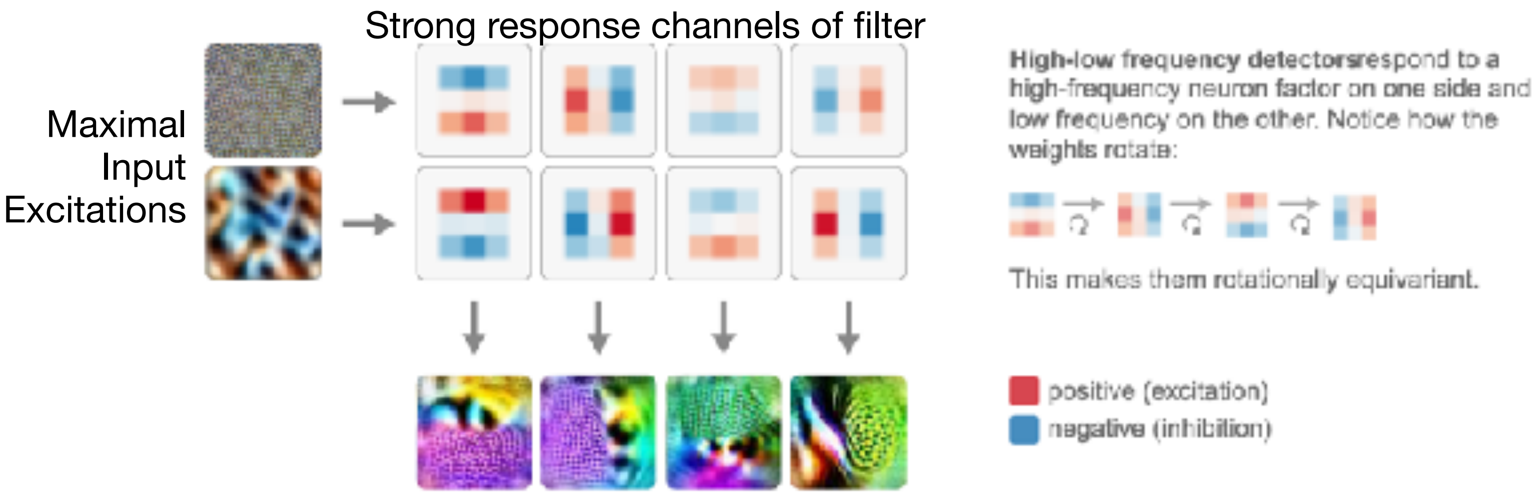
Equivariant Circuits

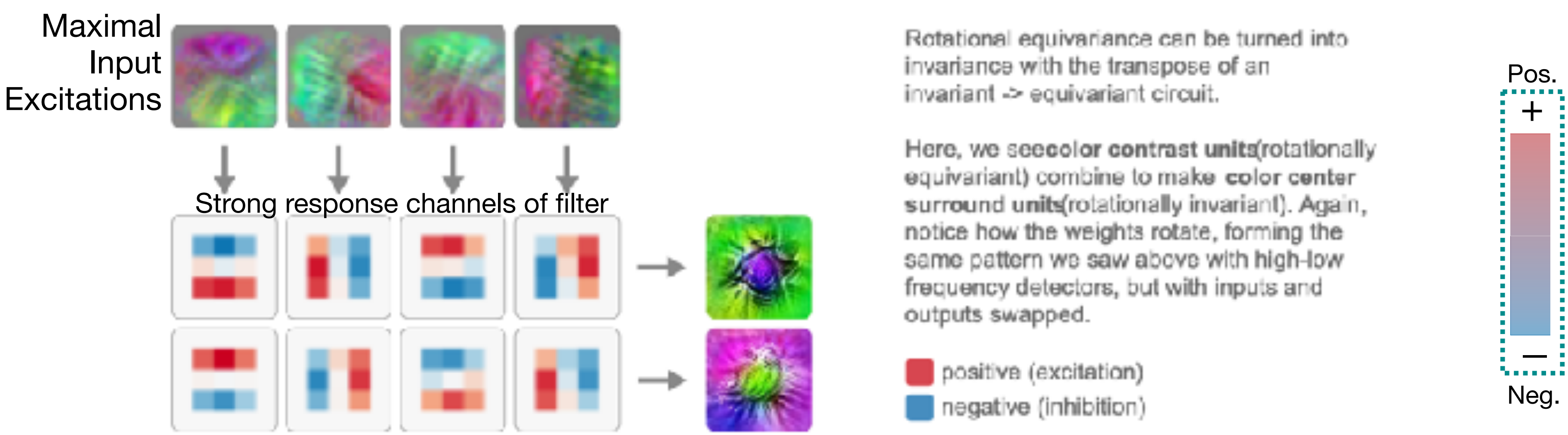
- Many features that are part of a circuit are clearly designed for rotation, hue, and other invariance

Olah, et al., "Naturally Occurring Equivariance in NN", 2021. https://distill.pub/2020/circuits/equivariance/

Equivariant circuits: a Motif

- Possible to reveal patterns of circuits via sets of weights





Olah, et al., "Naturally Occurring Equivariance in NN", 2021.

Lab Three Town Hall

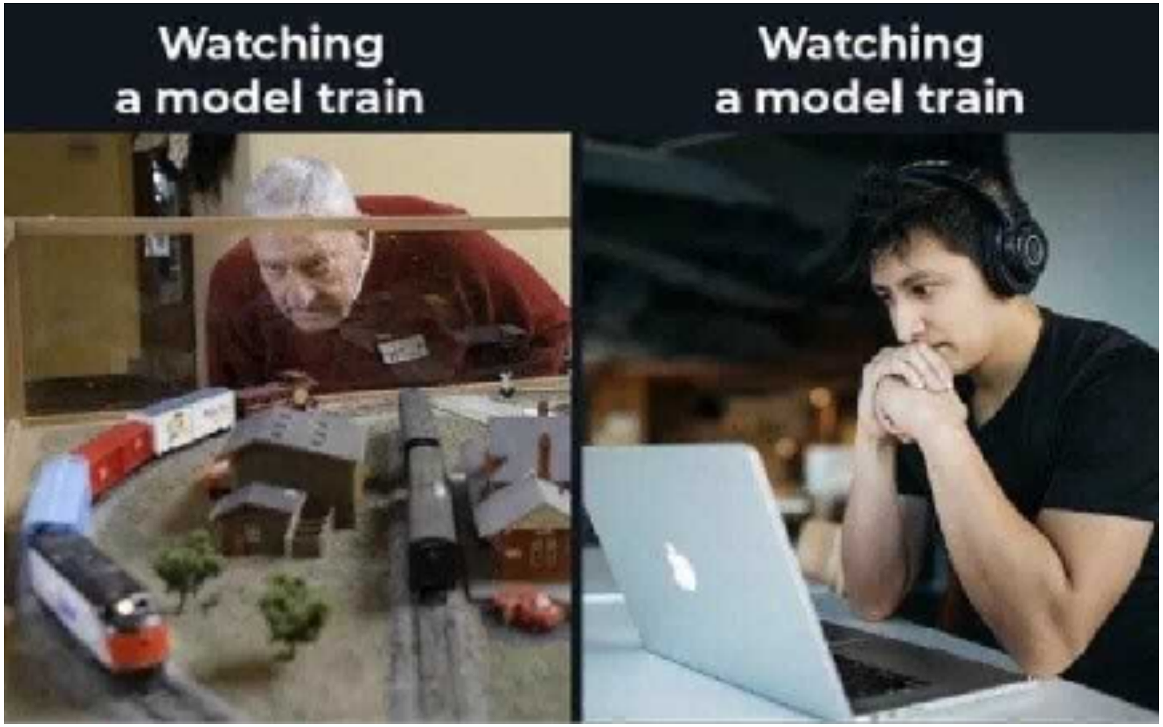
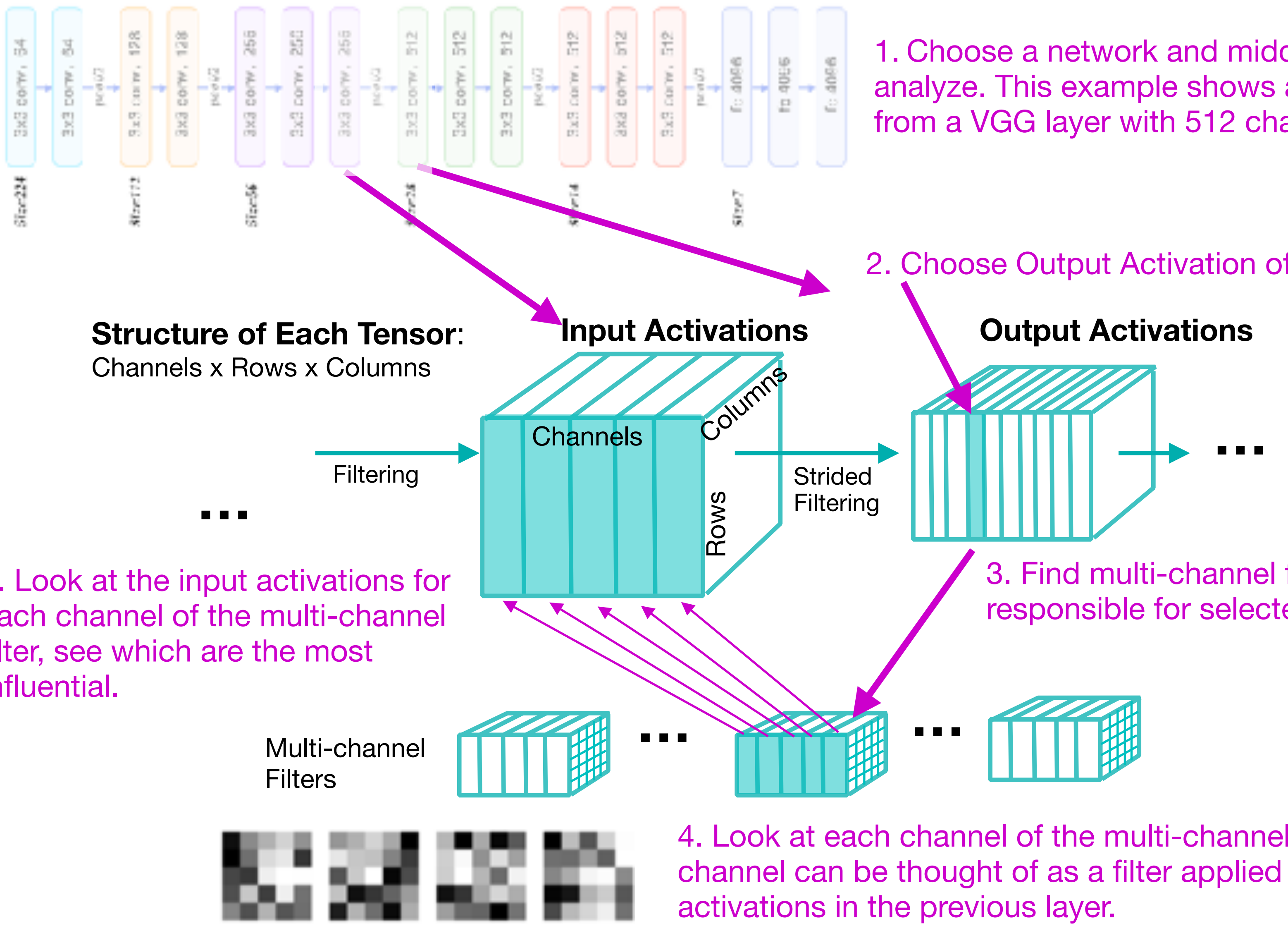


Figure for Circuits Lab

1. Choose a network and middle layer to analyze. This example shows activations from a VGG layer with 512 channels.

2. Choose Output Activation of Interest



3. Find multi-channel filter that is responsible for selected activation

4. Look at each channel of the multi-channel filter. Each channel can be thought of as a filter applied to the activations in the previous layer.

5. Look at the input activations for each channel of the multi-channel filter, see which are the most influential.

Office Hours

- Questions on current lab?

Lecture Notes for

Neural Networks and Machine Learning

CNN Circuits

Next Time:
Fully Convolutional Learning

Reading: Chollet 5.4