Master in Computer Vision
Barcelona

Module 3: Machine Learning for Computer Vision

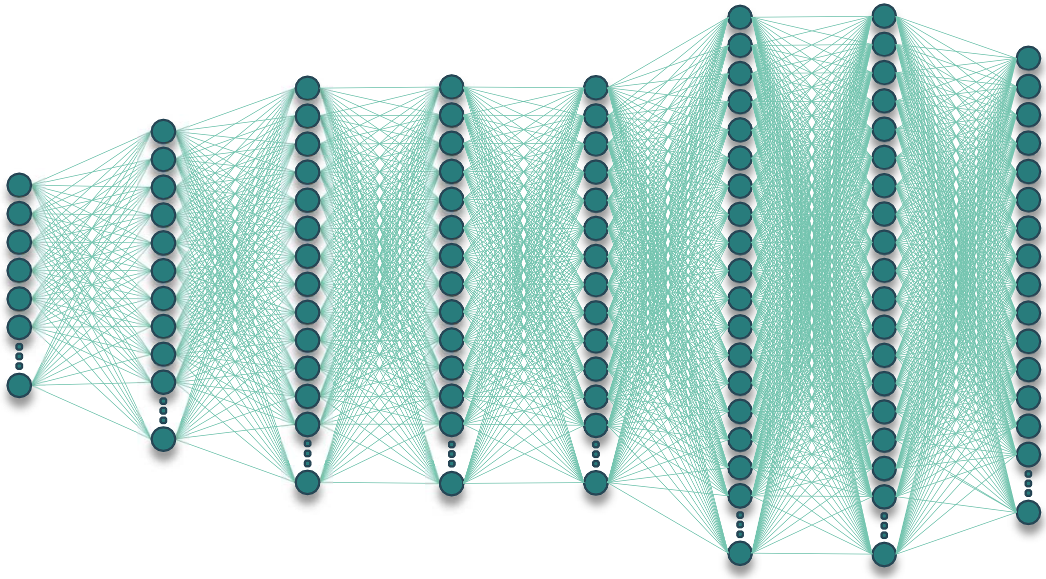
Lecture : Understanding and visualizing CNNs

Lecturers: Maria Vanrell / Guillem Arias

Credits for Some Slides to: Ivet Rafegas

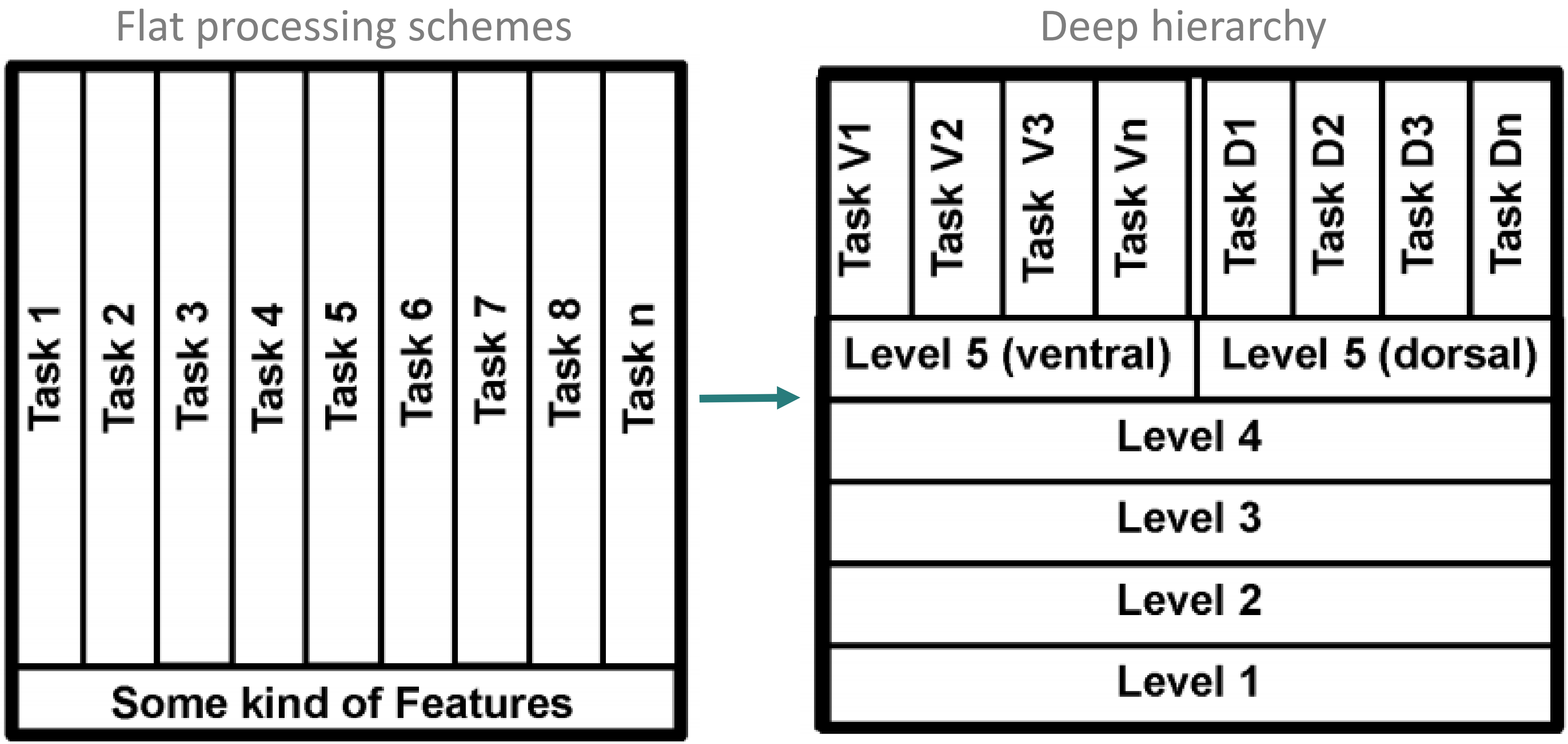
Motivation

Since 2010 Convolutional Neural Networks have overcome all previous image descriptors



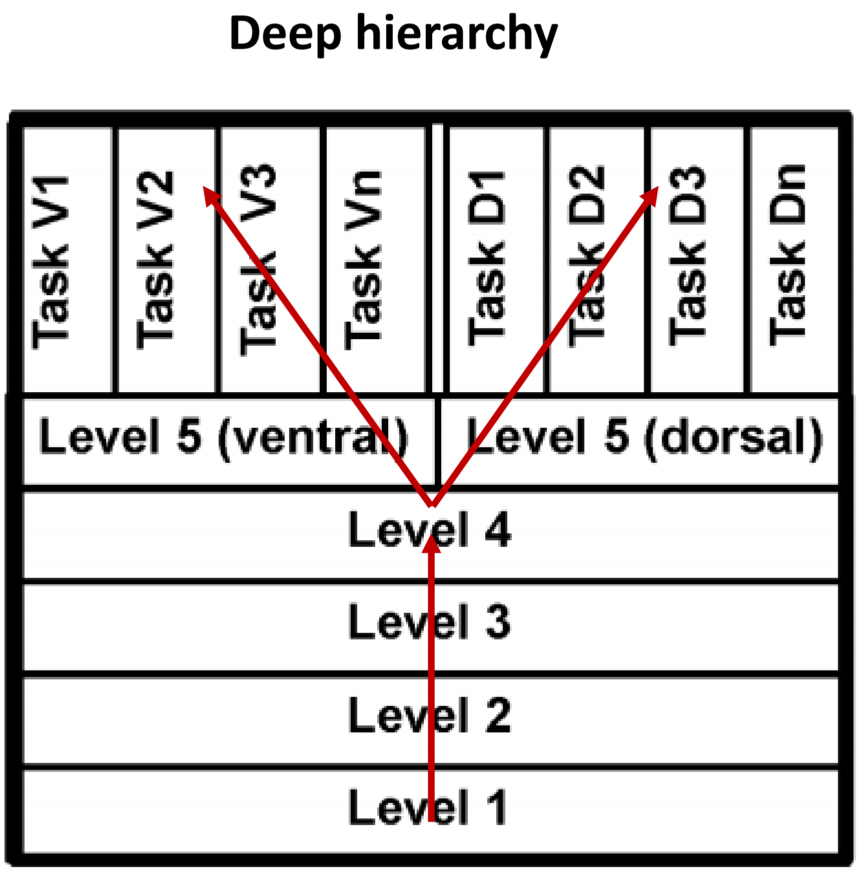
Previous **Image descriptors** where designed to represent specific spatial features, such as edges at different directions, blobs, and combined with first order statistics

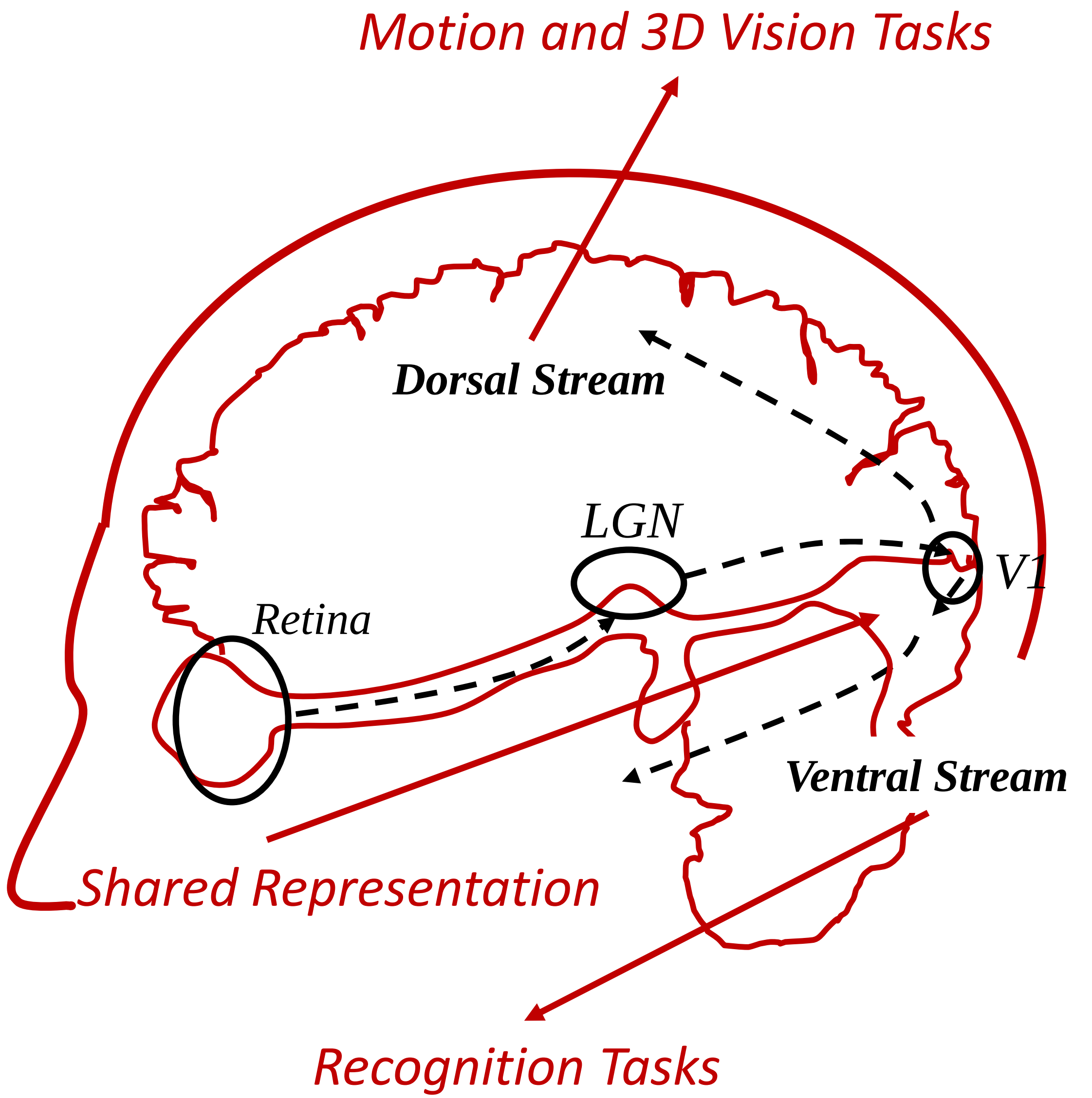
The CNN paradigm Change *According to Kruger et-al 2013*



[Kruger,13]N. Kruger, P. Janseen, S. Kalkan, M. Lappe, A. Leonardis, J. Piater, A. J. Rodriguez-Sanchez, L. Wiskott.
Deep hierarchies in the primate visual cortex: What can we learn for computer vision?.
IEEE Trans. Pattern Anal. Mach. Intell., 35 (8). 2013

The CNN paradigm has more parallelisms with human brain





The paradigm change from the features point of view

Handcrafted features → *Learned features*

Handcrafting allows to **analyse and design** what are the best intuitively useful features

Learning implies **no idea which features are optimizing the loss function**

Flat features → *Hierarchical features*

Concatenating simple feature is **easy to understand and visualize**

Hierarchical features are introducing a high level of abstraction encoded in convolution that brings **more complexity to be understood**

The paradigm change has brought a new problem:

Explainability, interpretabily, understanding …

There is not a single method or tool to explain the black-box nature of the DNN models yet

The most updated Survey:

On Interpretability of Artificial Neural Networks: A Survey

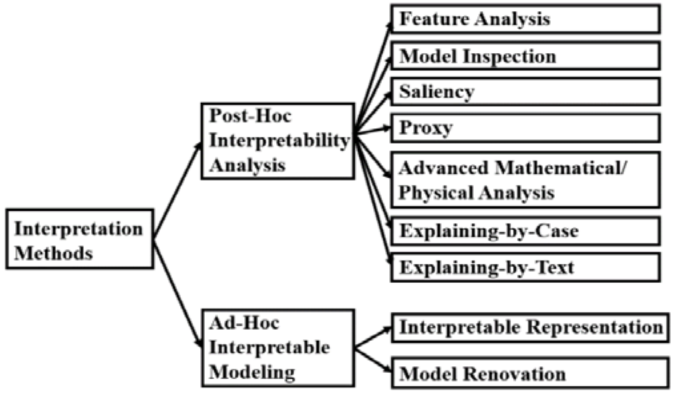
Feng-Lei Fan, *Student Member, IEEE*, Jinjun Xiong, *Senior Member, IEEE*, Mengzhou Li, *Student Member, IEEE*, and Ge Wang, *Fellow, IEEE*

***Abstract*— Deep learning as represented by the artificial deep neural networks (DNNs) has achieved great success in many important areas that deal with text, images, videos, graphs, and so on. However, the black-box nature of DNNs has become one of the primary obstacles for their wide acceptance in mission-critical applications such as medical diagnosis and therapy. Due to the huge potential of deep learning, interpreting neural networks has recently attracted much research attention. In this paper, based on our comprehensive taxonomy, we systematically review recent studies in understanding the mechanism of neural networks, describe applications of interpretability especially in medicine, and discuss future directions of interpretability**

provides in-depth perspectives but is limited in scop example, only 49 references are cited there. The review Du *et al.* (2018) has a similar weakness, only cov papers which are divided into post-hoc and explanations, as well as global and local interpretatio taxonomy is coarse-grained and neglects a nu important publications, such as *explaining-by-text*, *ex by-case*, etc. In contrast, our review is more deta comprehensive, which includes the latest results publications in L. H. Gilpin *et al.* (2018) are classi understanding the workflow of a neural

https://arxiv.org/vc/arxiv/papers/2001/2001.02522v2.pdf

Proposed Taxonomy:



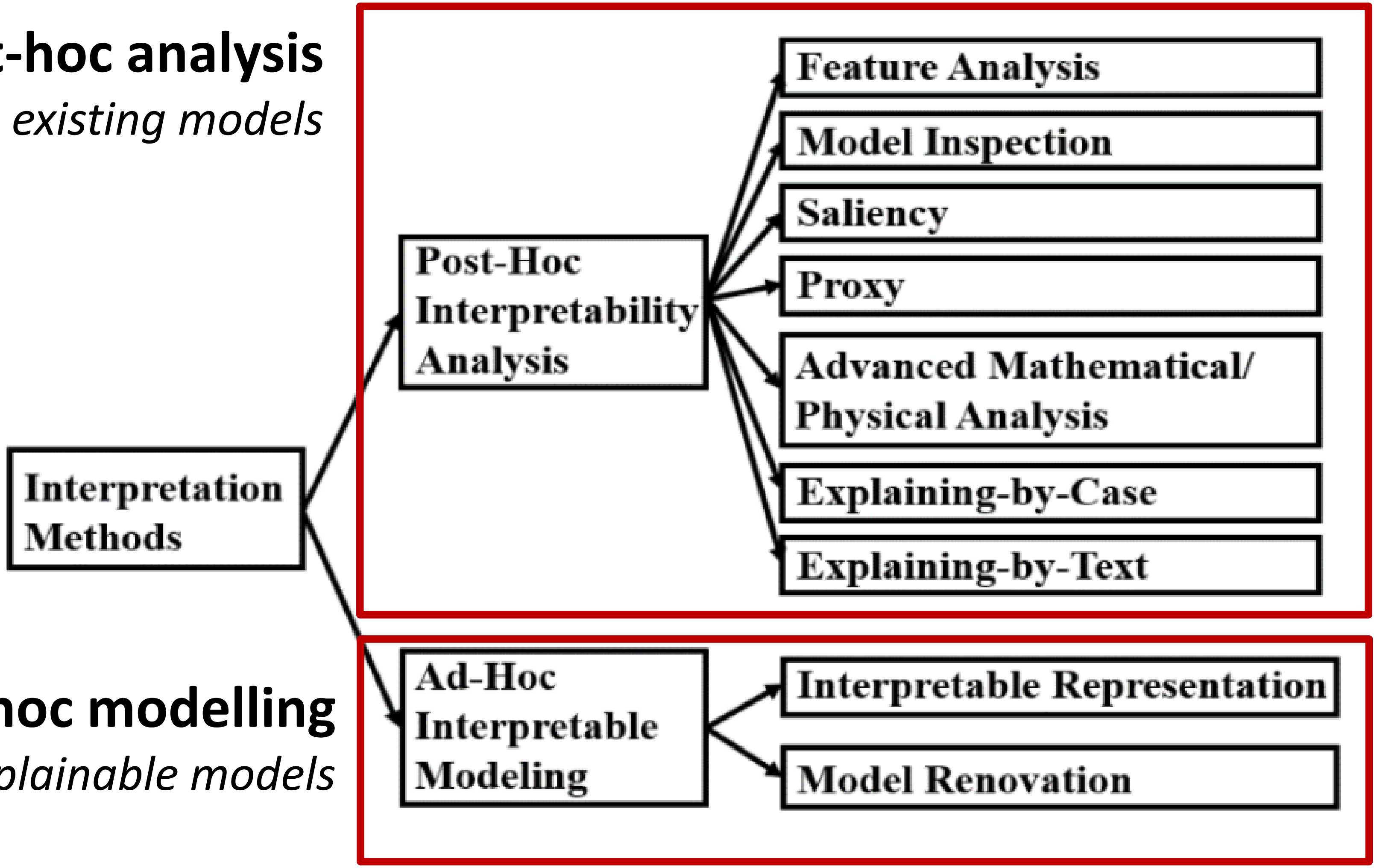
Proposed Taxonomy:

Post-hoc analysis

To explain existing models

Ad-hoc modelling

To build explainable models



Index of this Lecture:

Preliminary considerations

Post-hoc analysis

- Neuron Analysis
- Data Inspection
- Saliency based
- Proxy models
- Modifications
- Theoretical Analysis

Ad-hoc modelling

- Interpretable representation
- Model Renovation

A case study on a single feature *(post-hoc analysis)*

How color is represented in a CNN? and parallelisms with HVS

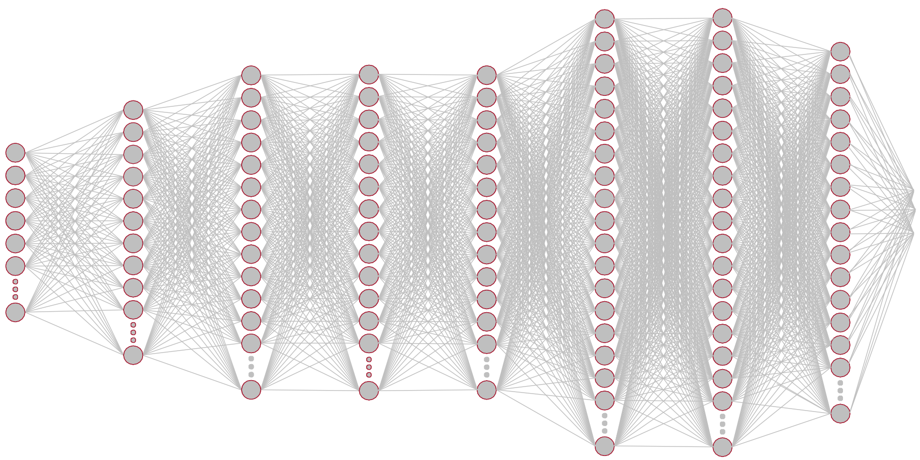
Preliminary considerations:

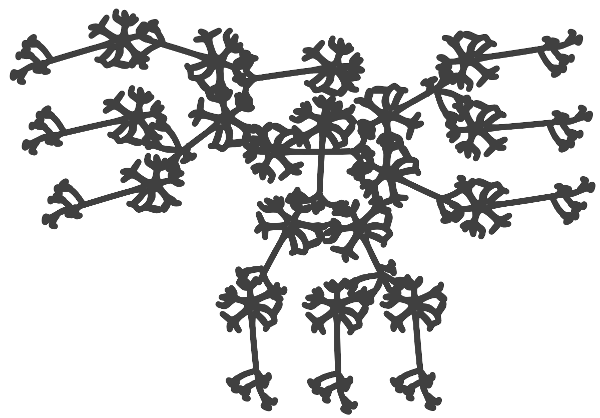
Explainability is necessary due to the black-box nature of CNNs

Usual questions to be asked about the CNN based models:

- How a deep model concludes such a prediction ?
- Why are some features favored over others ?
- What changes are needed to improve model performance ?

If we open the box of our CNN we see Neurons:





Preliminary considerations

1. What is a neuron?
2. What information can we use to characterize neurons?

What is a neuron?

We already know some concepts related to neurons,

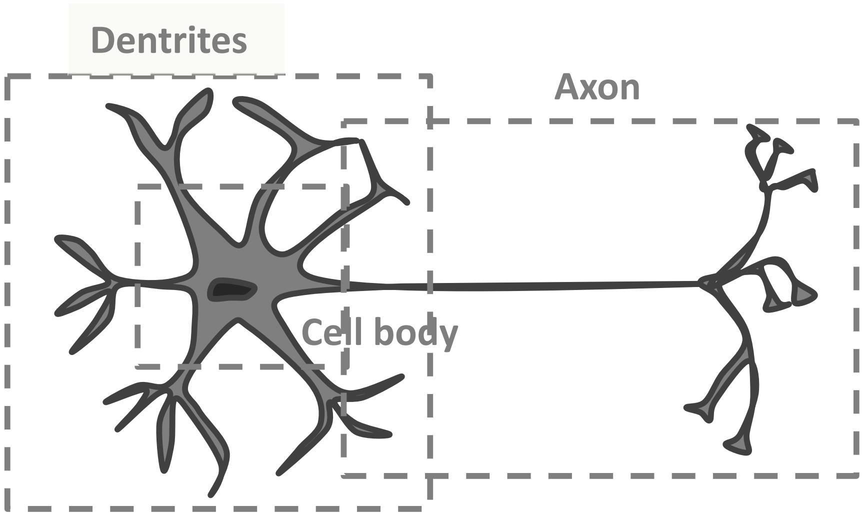
- **Neurons or Units** are the basis of the layers that a CNN relies on
- They are the main responsible for the ability of representing the input image in **high-dimensional feature spaces**
- Each neuron is **associated to some weights** that has been learnt during the CNN training step
- Neurons **give activation responses** depending on their inputs and their weights

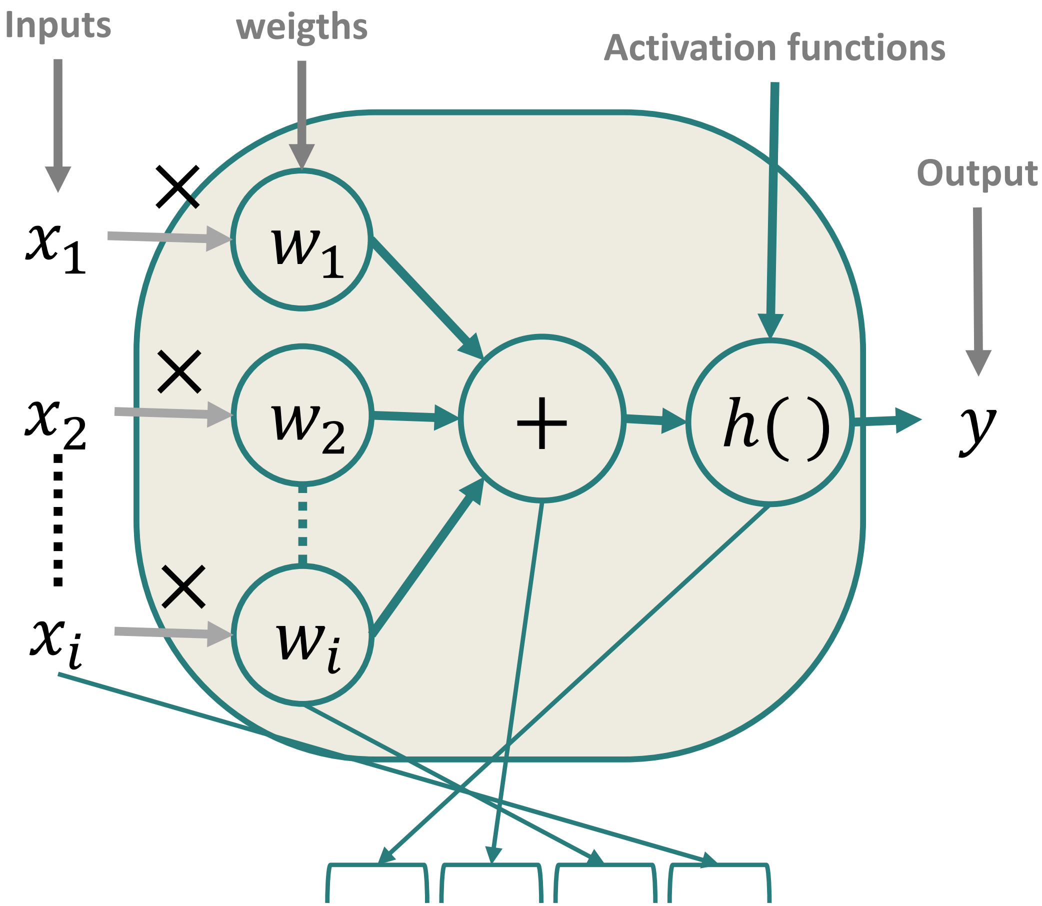
So,

what these weights or activations are explaining about the neuron?

Usual parallelisms between biological and computational neurons

Real Neuron → Artificial Neuron

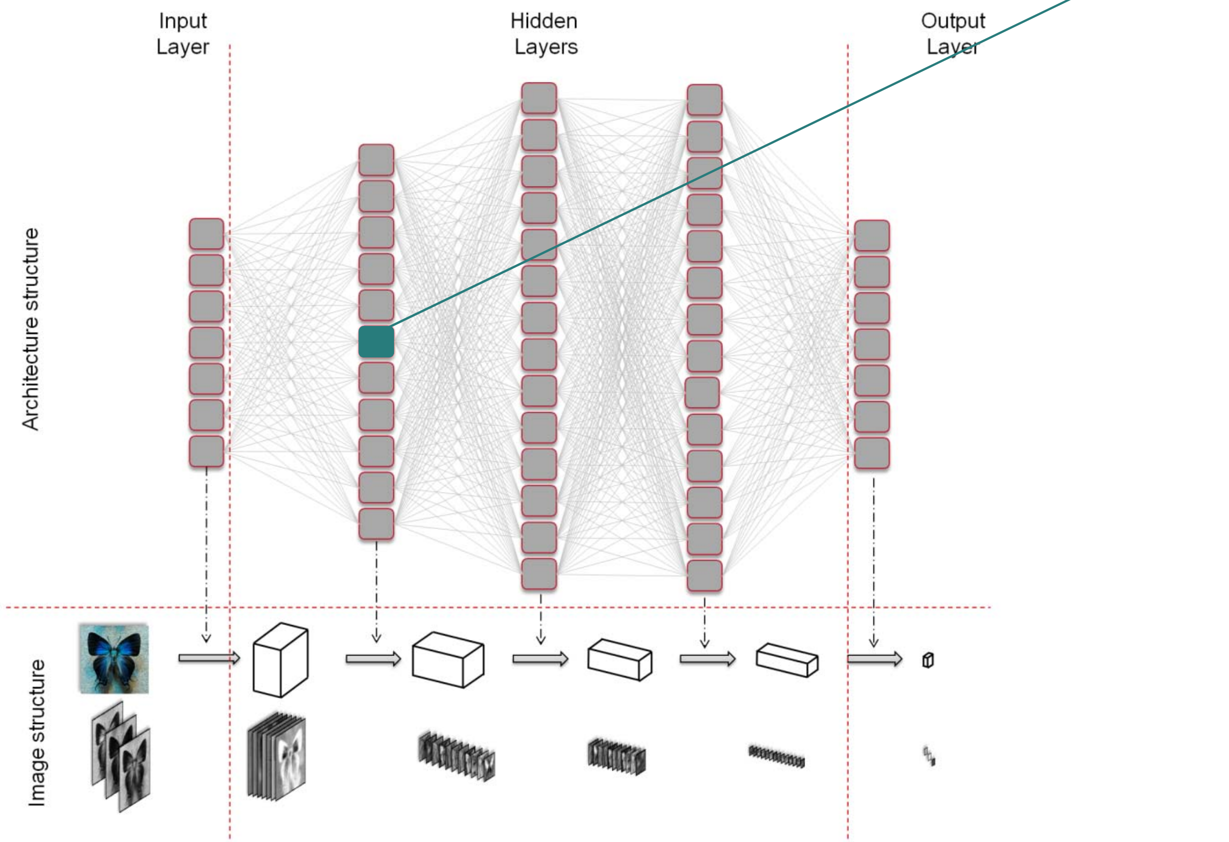




$$y = h\left(\sum_j w_j \cdot x_j\right)$$

When neurons are grouped in layers ...

All **neuron outputs** are grouped in large tensors



Usual questions about individual neurons:

Which feature is this neuron selecting from the input image?

Which is the task of this neuron within the global CNN task?

***In summary,** what characterizes this neurons?*

Preliminary conosiderations :

1. What is a neuron?

2. What information can we use to characterize neurons?

Let's take a look to:

- Neuron weights
- Neuron outputs

In Convolutional Neural Networks

Product → Convolution

$$y = h\Big(\sum_j w_j \boxed{\cdot} x_j\Big) \longrightarrow y = h\Big(\sum_j \boxed{w_j}\boxed{*}\boxed{x_j}\Big)$$

Neuron weights

Image

Convolution returns a high value when the pattern we have in the **kernel fits with the image pattern**

w_1	w_2	w_3
w_4	w_5	w_6
w_7	w_8	w_9

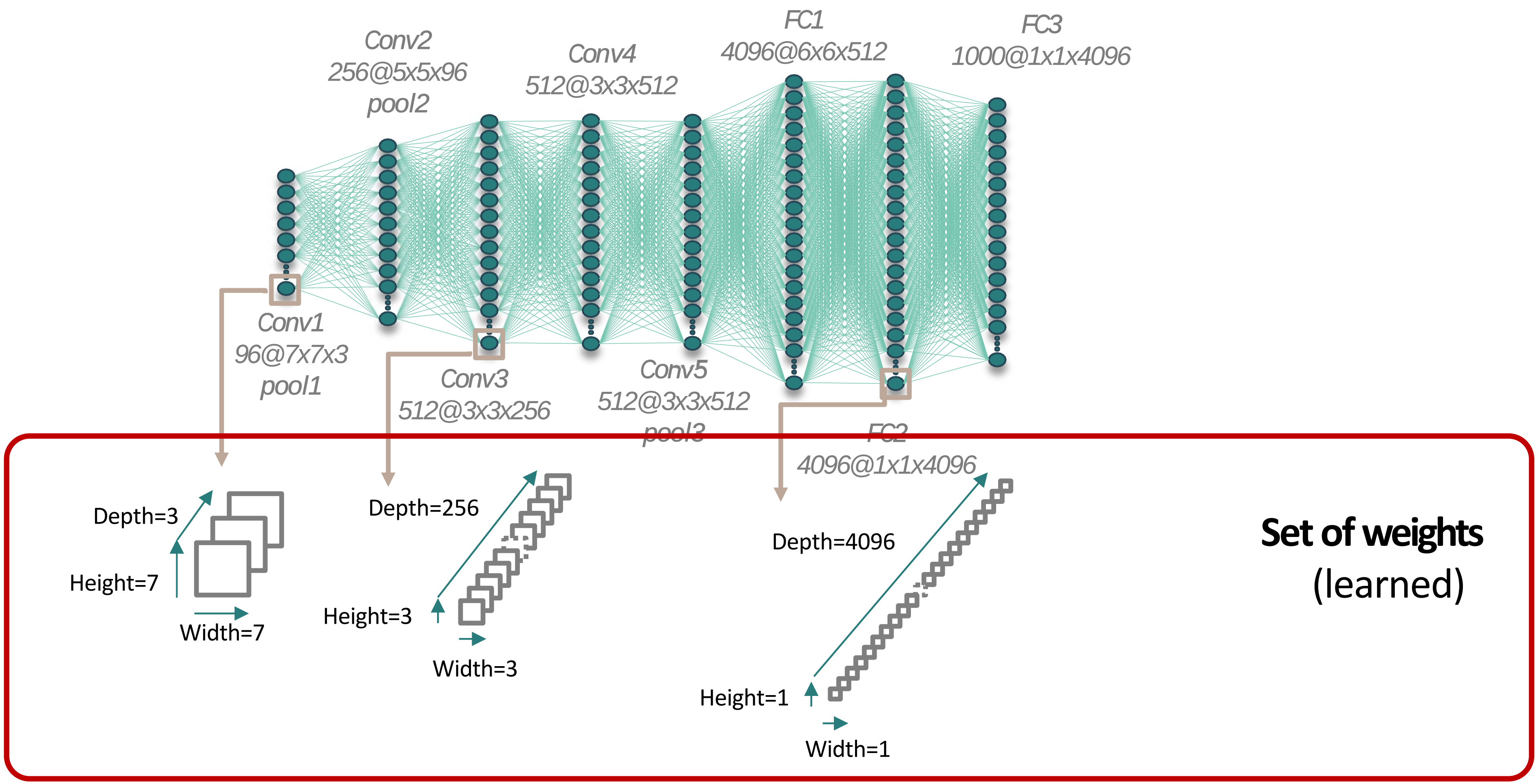
$n \times n$

Filter

125	127	132	2	1	…
127	130	131	2	3	…
130	131	132	2	2	…
125	127	2	2	4	…
125	127	2	4	5	…
…	…	…	…	…	…

How many weights for each Neuron?

we have a **3D tensor of weigths** in the 1st layer neurons



Size (height,width) encode the spatial dimension of the convolution kernels

Depth of a neuron is fixed by the number of neurons (channels) in the previous convolutional layer.

Preliminary definitions :

1. What is a neuron?

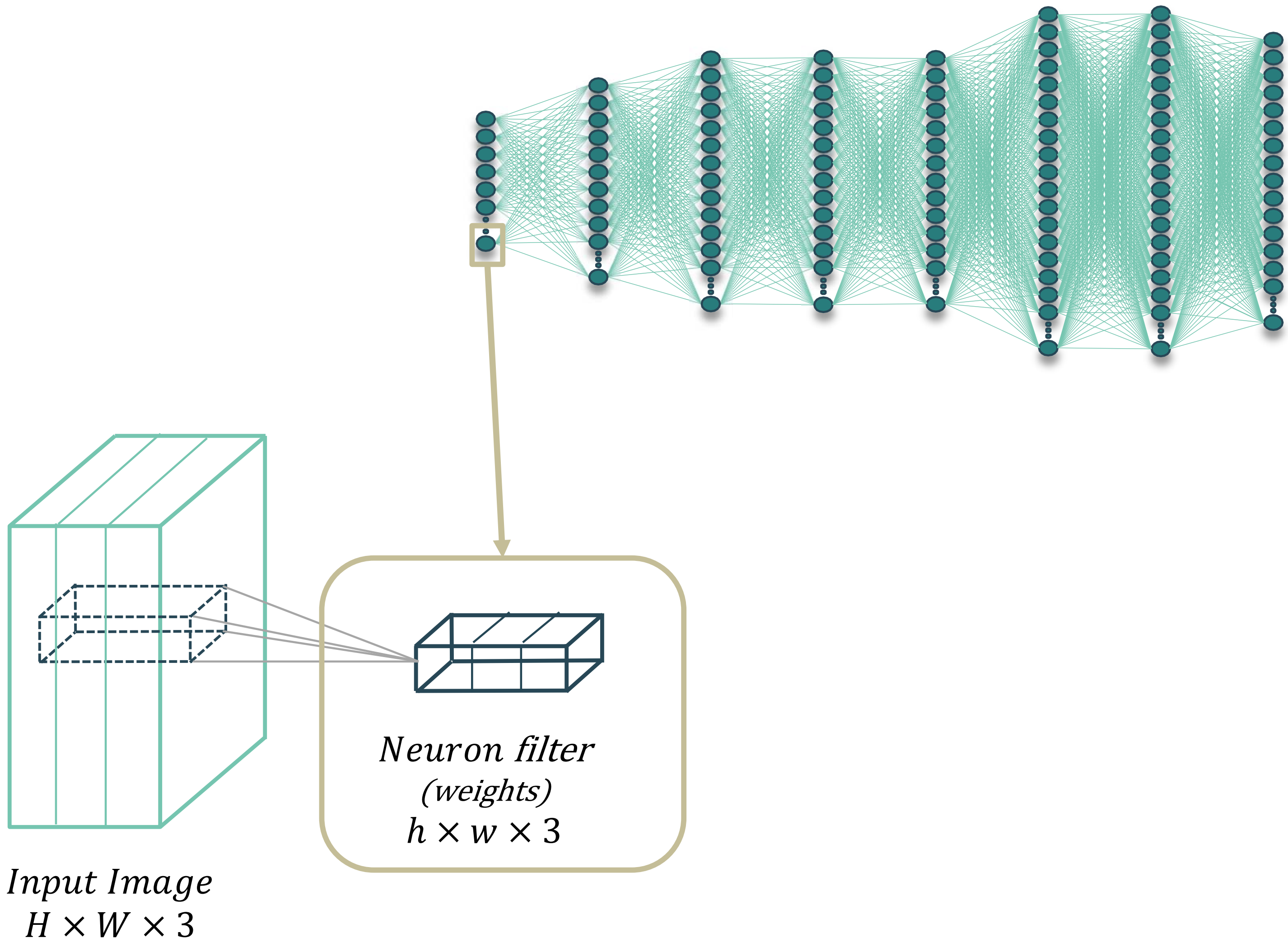
2. What information can we use to characterize neurons?

   Let's take a look to:

   - Neuron weights
   - Neuron outputs

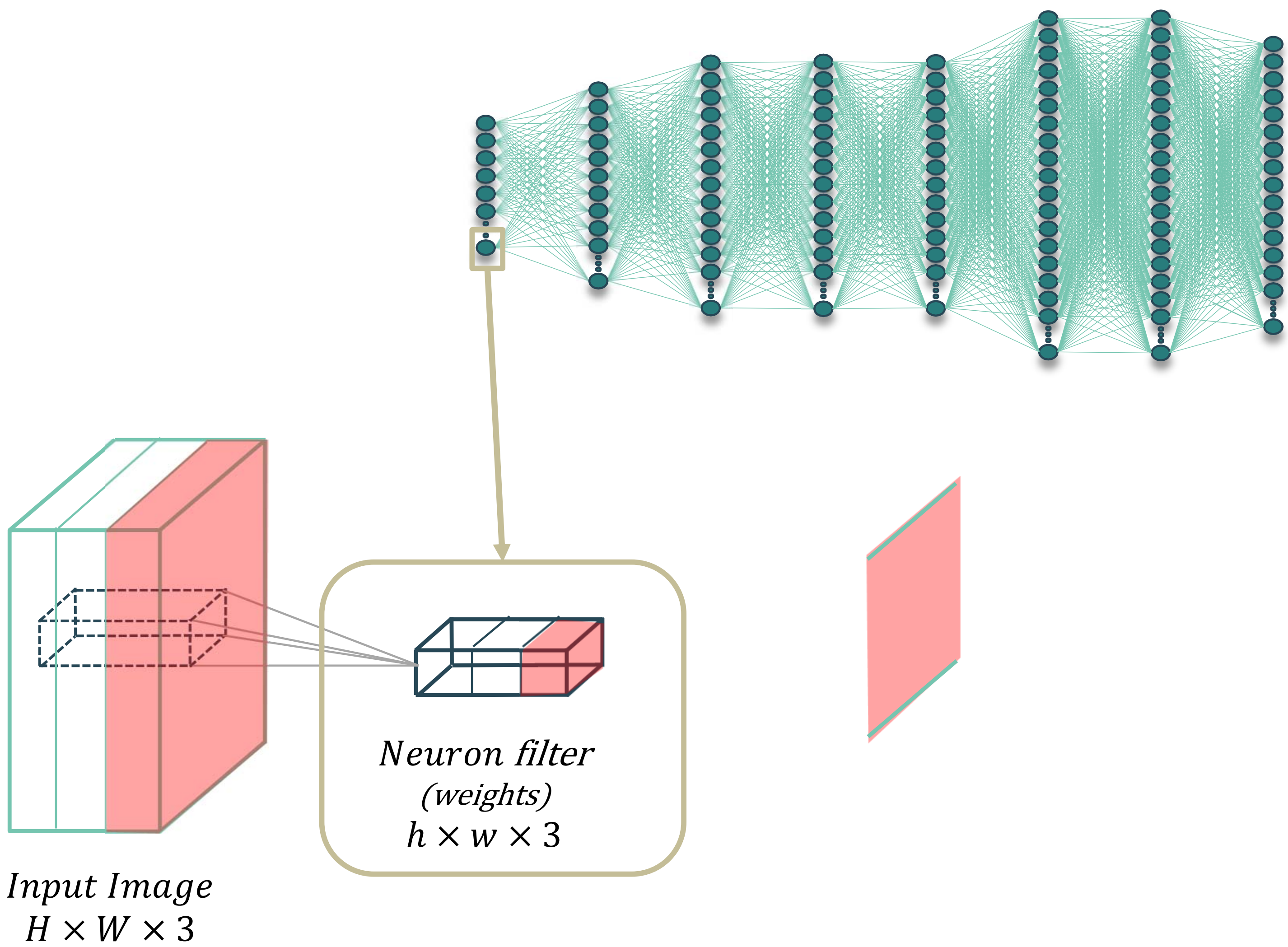
How is the Neuron Output?

each neuron has an activation for every image pixel



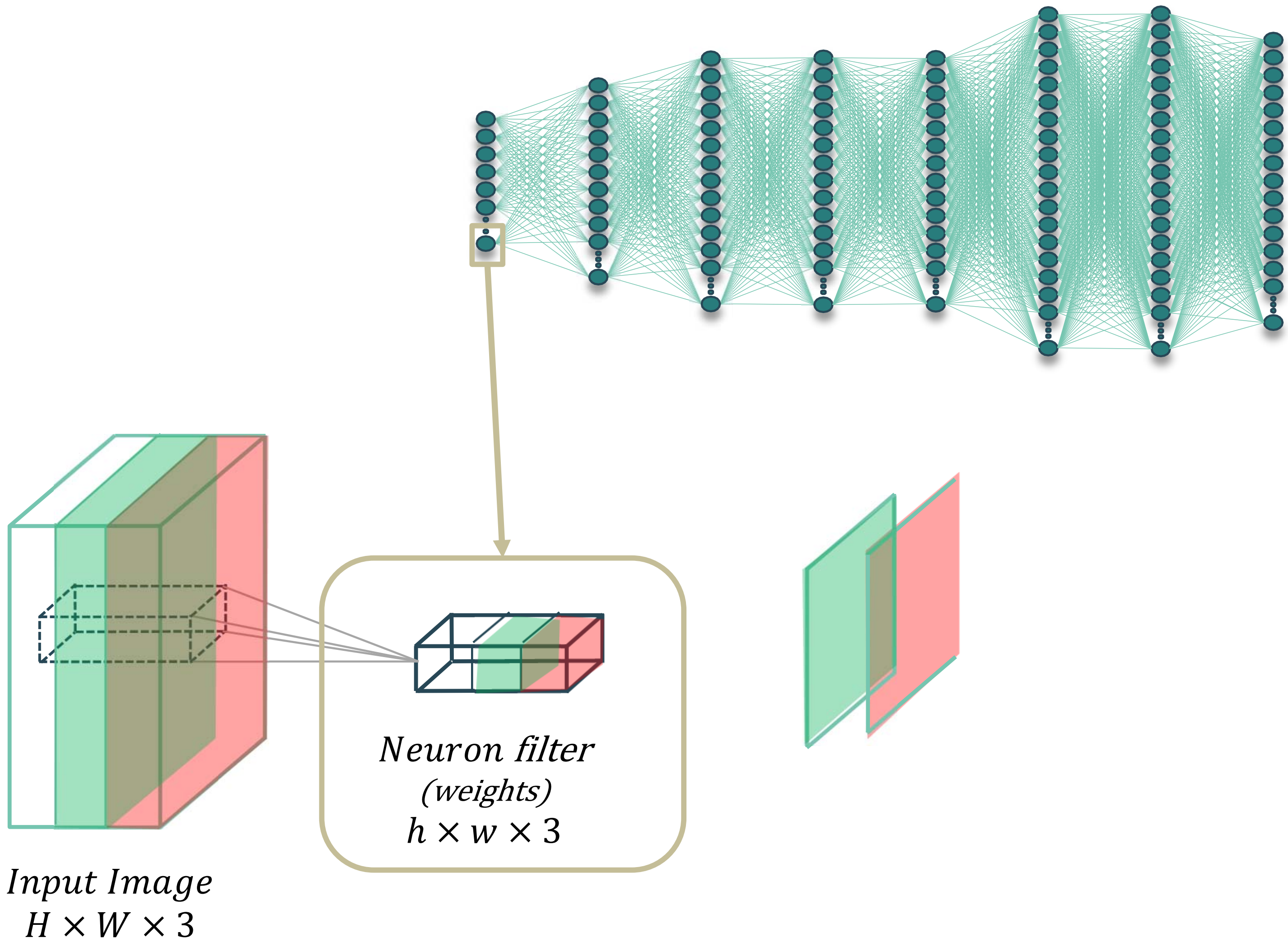
What is the Neuron Output?

each neuron has an activation for every image pixel



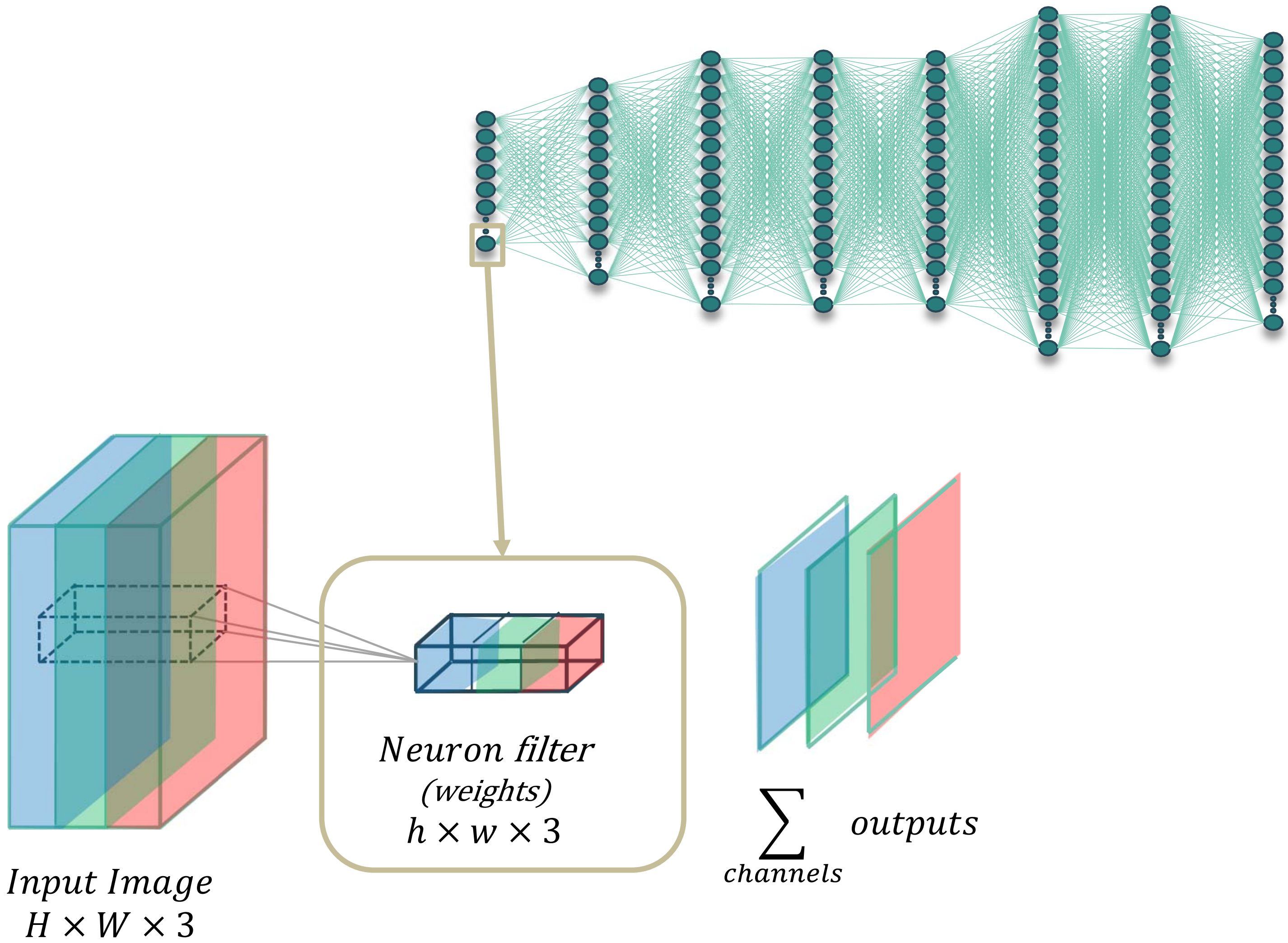
What is the Neuron Output?

each neuron has an activation for every image pixel



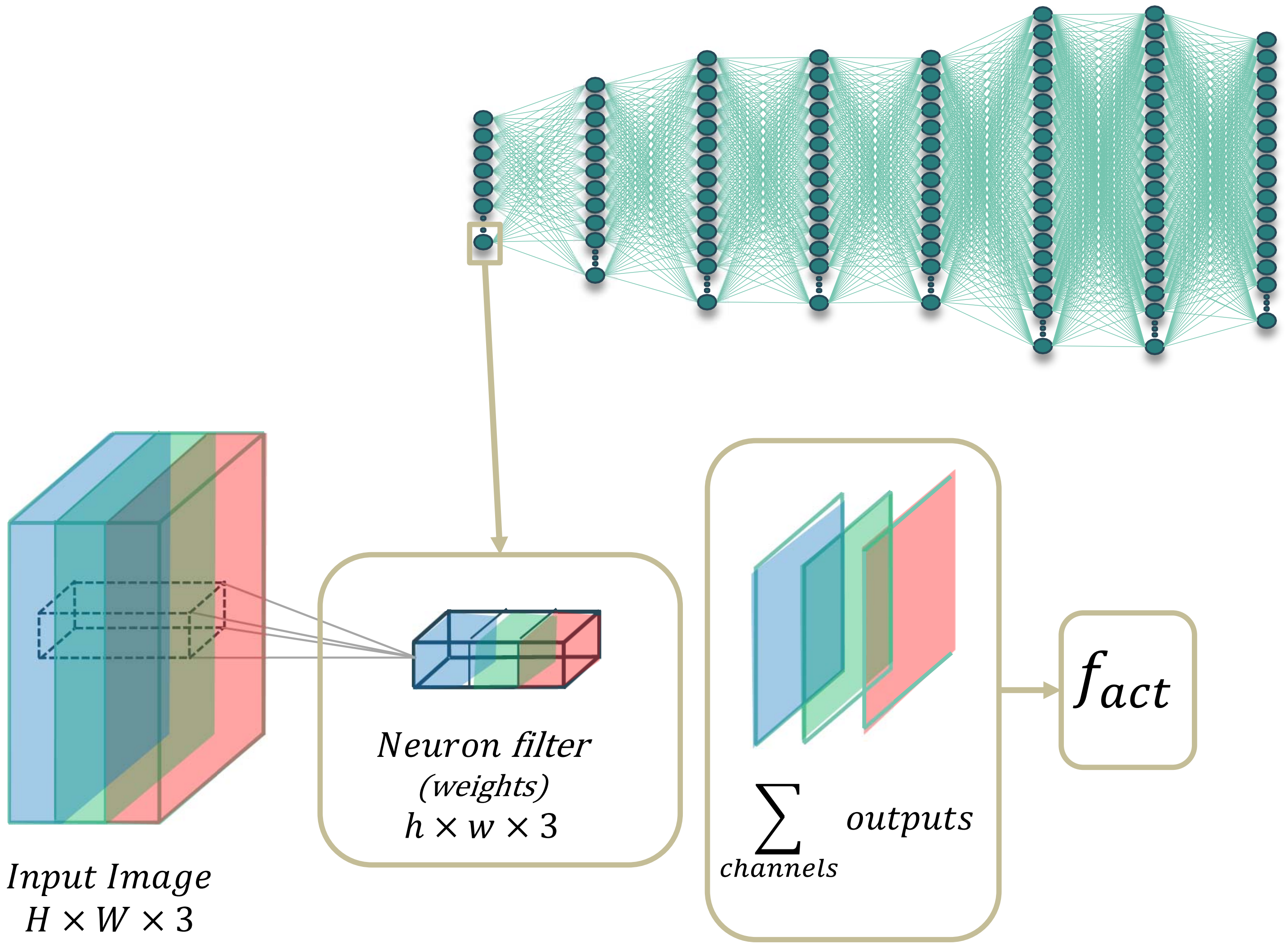
What is the Neuron Output?

each neuron has an activation for every image pixel



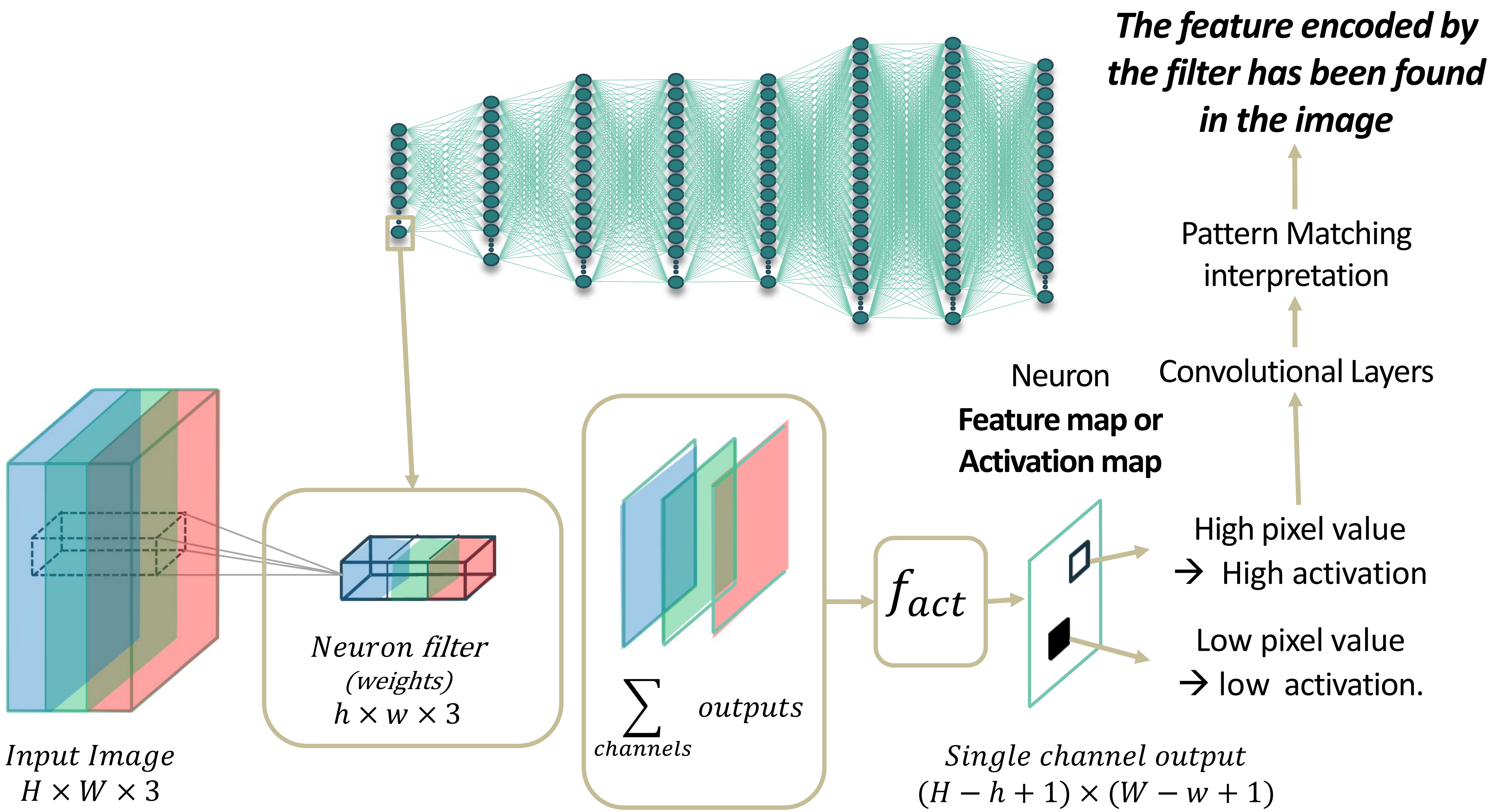
What is the Neuron Output?

each neuron has an activation for every image pixel



What is the Neuron Output?

each neuron has an **activation map** for all the image pixels



Idea: The feature encoded by the filter associated to a neuron has been found in the image

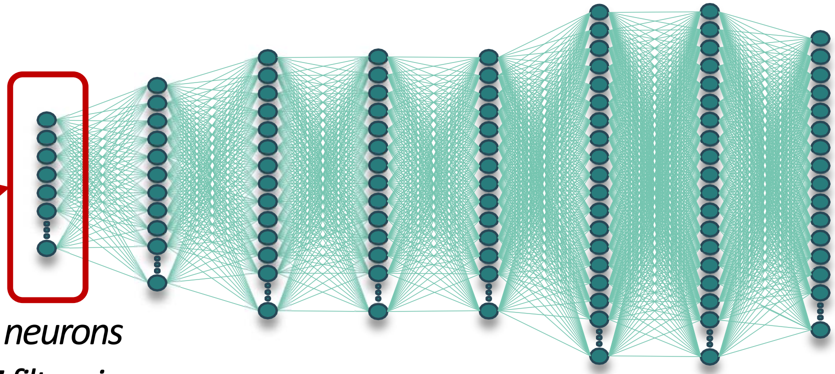
Considering this <u>Pattern Matching</u> interpretation,

if we visualize the weights,
we can visualize what is detected in the image when a neuron activates .....

Solution: Let's visualize the neuron weights

Visualizing weights

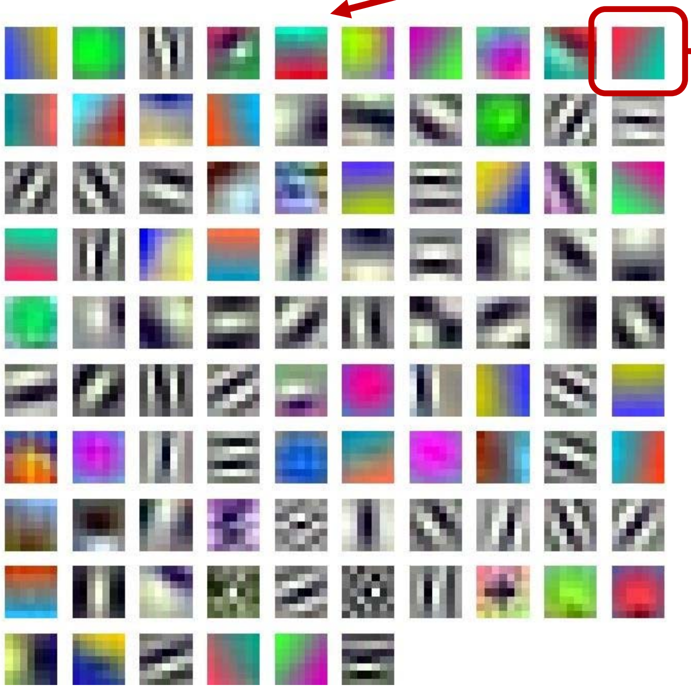
First convolutional layer



96 neurons

7x7 filter size

Filter weights visualization



1	1	1	1	1	1	0
1	1	1	1	1	0	0
1	1	1	1	0	0	0
1	1	1	1	0	0	0
1	1	1	0	0	0	0
1	1	0	0	0	0	0
1	1	0	0	0	0	0
1	0	0	0	0	0	0

Channel 1 (RED)

0	0	0	0	0	0	1
0	0	0	0	0	1	1
0	0	0	0	1	1	1
0	0	0	1	1	1	1
0	0	0	1	1	1	1
0	0	1	1	1	1	1
0	0	1	1	1	1	1
0	1	1	1	1	1	1

Channel 2 (GREEN)

0	0	0	0	0	0	1
0	0	0	0	0	1	1
0	0	0	0	1	1	1
0	0	0	1	1	1	1
0	0	0	1	1	1	1
0	0	1	1	1	1	1
0	0	1	1	1	1	1
0	1	1	1	1	1	1

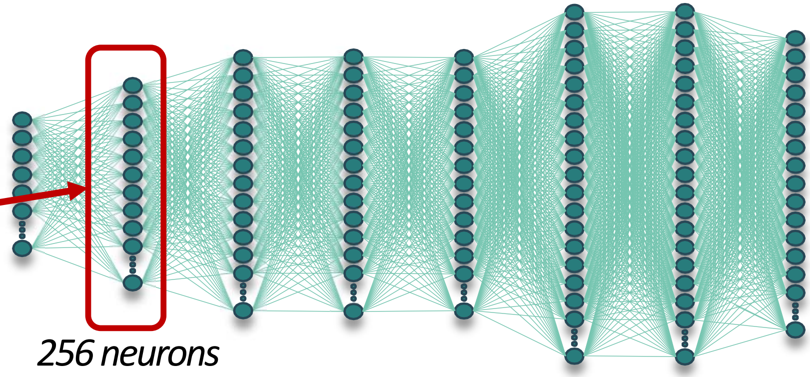
Channel 3 (BLUE)

Interpretation: Red-Cyan 45º Edge Detector

"Convolution = Pattern Matching on RGB image space"

Visualizing weights

2nd convolutional layer



256 neurons

5x5 filter size

Channel 1 .... *Channel 96*

Filter 1

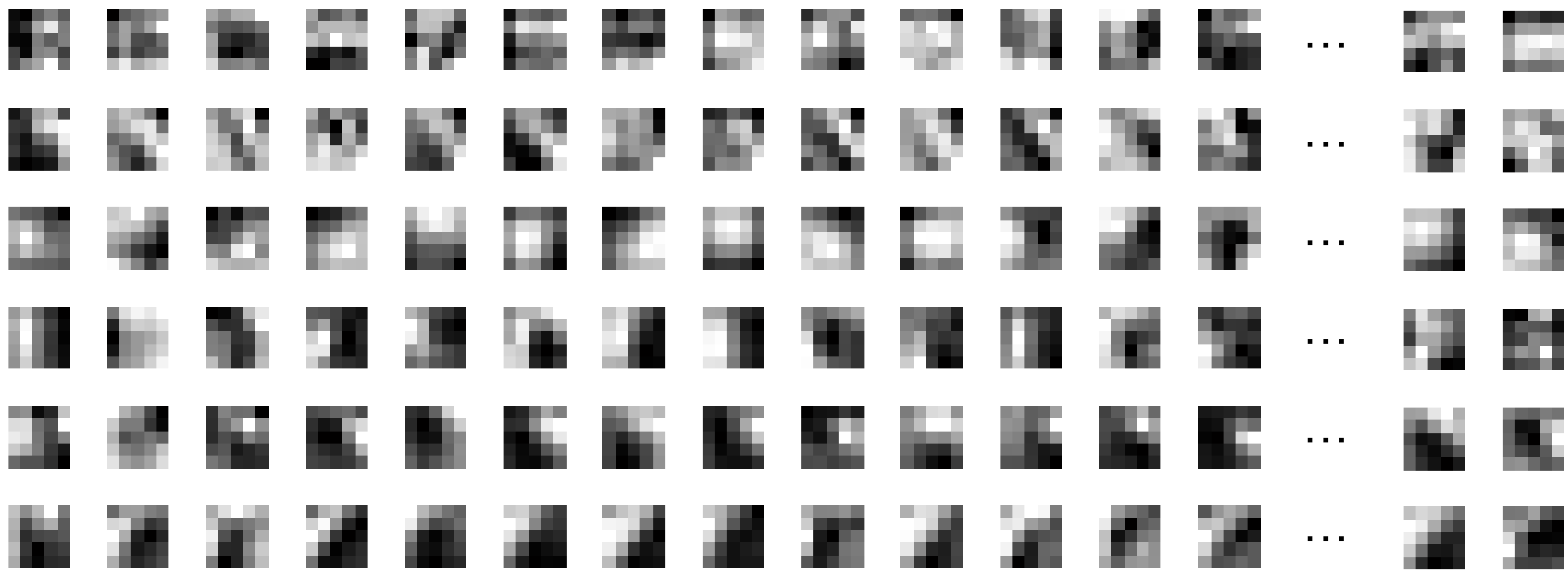
Filter 2

Filter 3

Filter 4

Filter 5

Filter 6



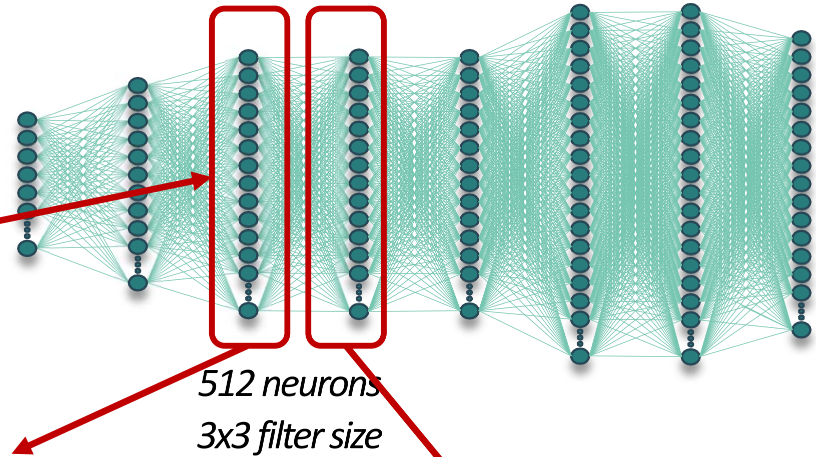
⋮

Filter 256 ....

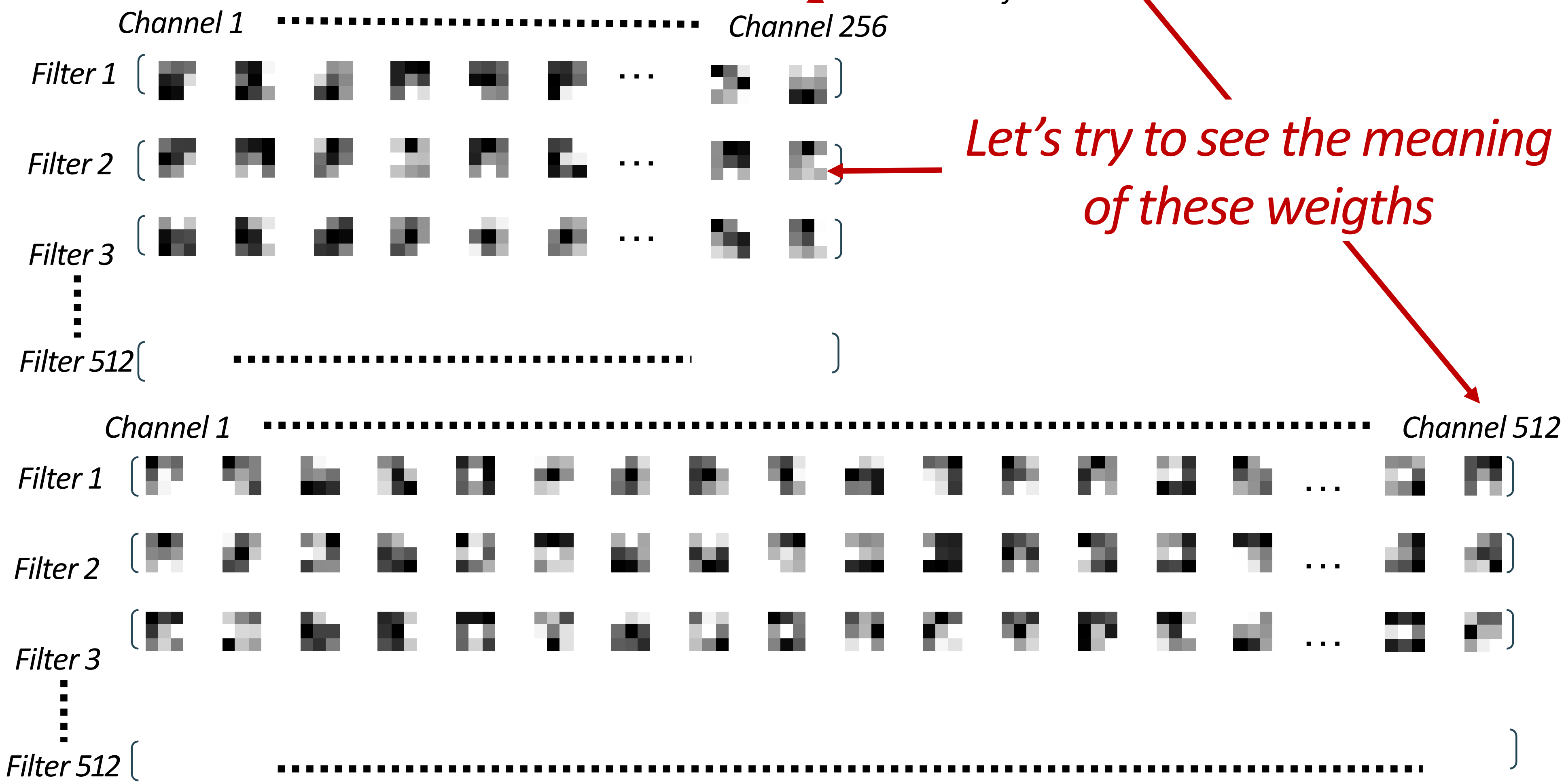
Interpretation? , we can not plot an image of 96 channels!!

Visualizing weights

3rd and 4th convolutional layers

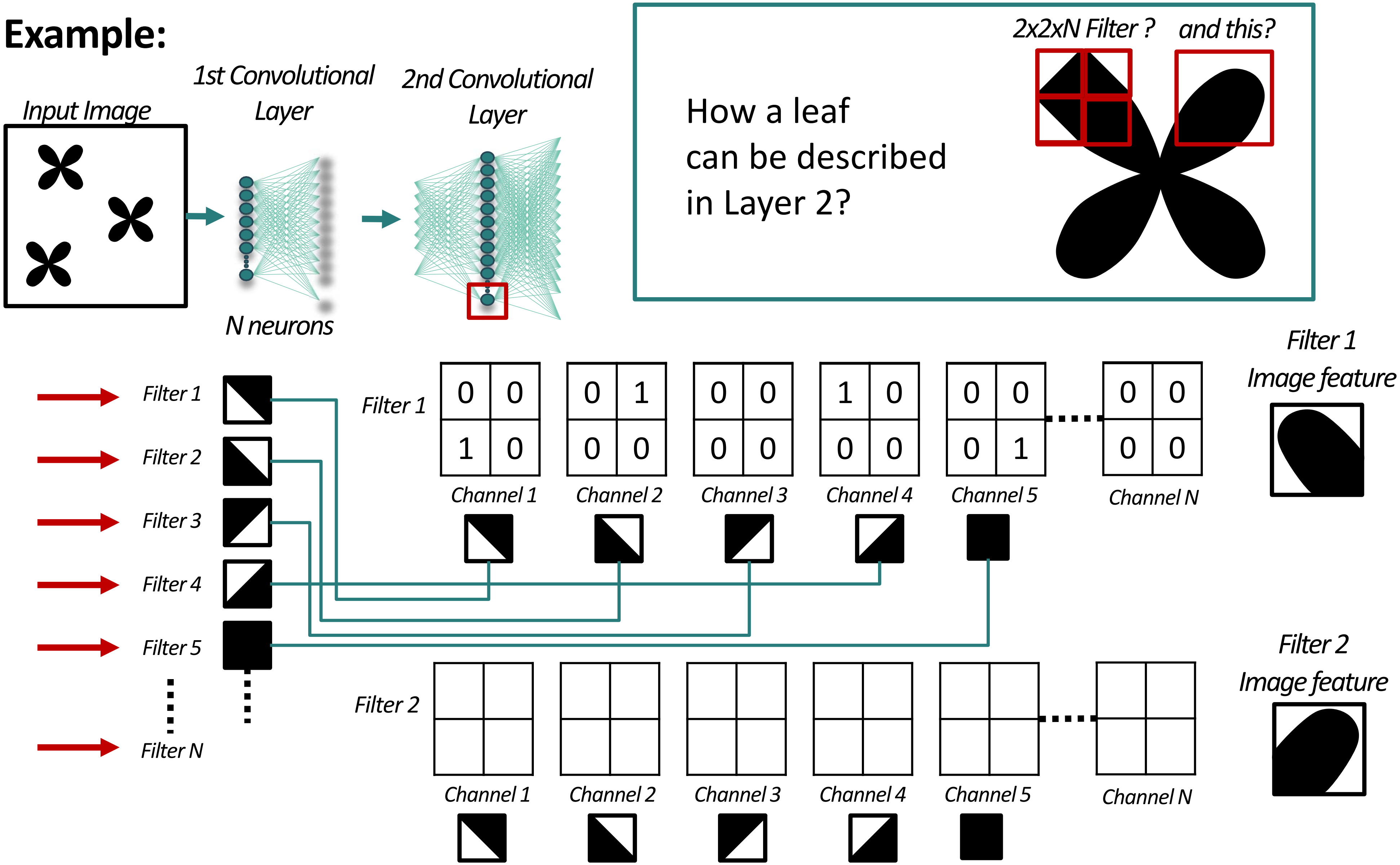


Let's try to see the meaning of these weigths



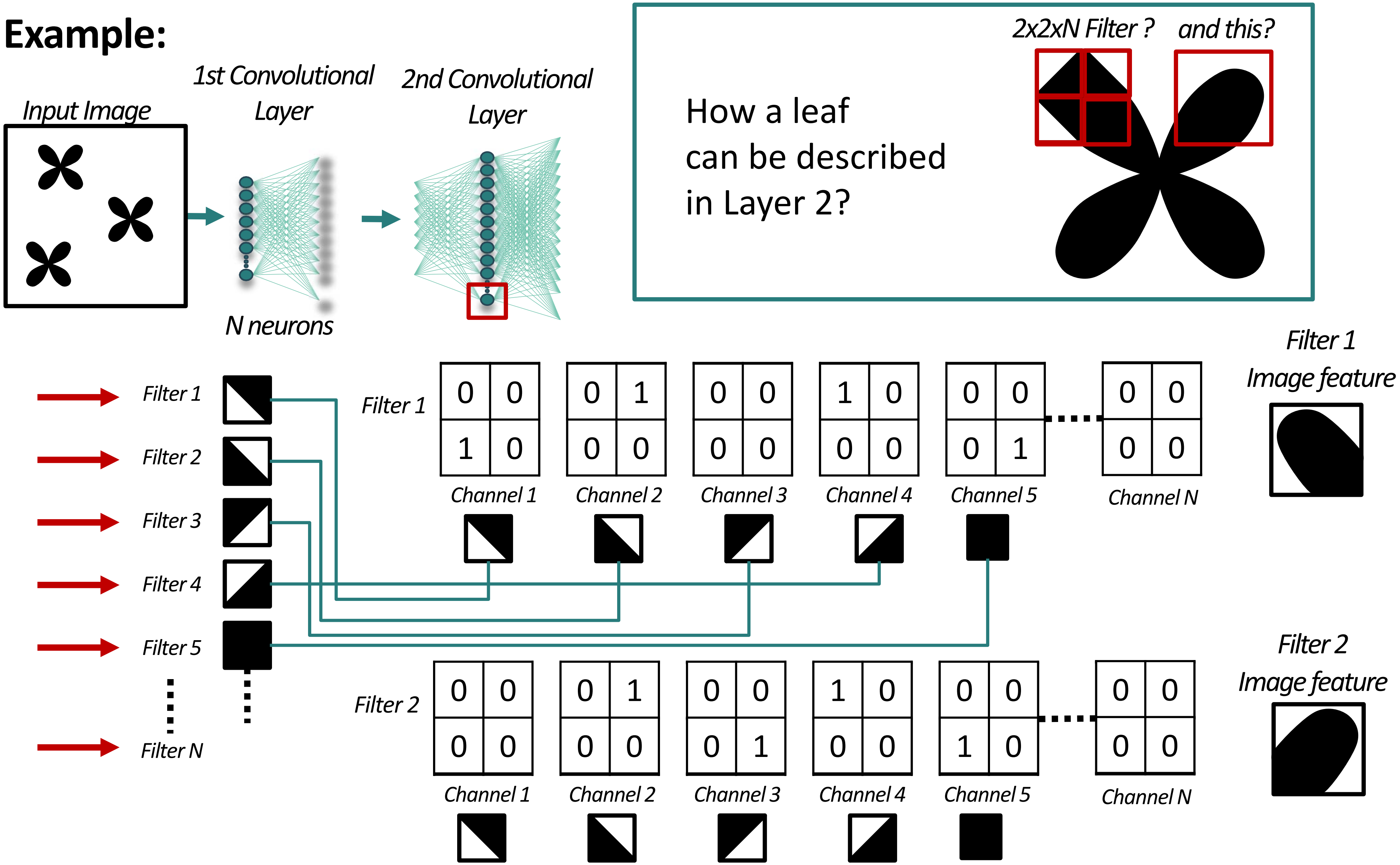
Understanding Composition of Convolutions

Example:

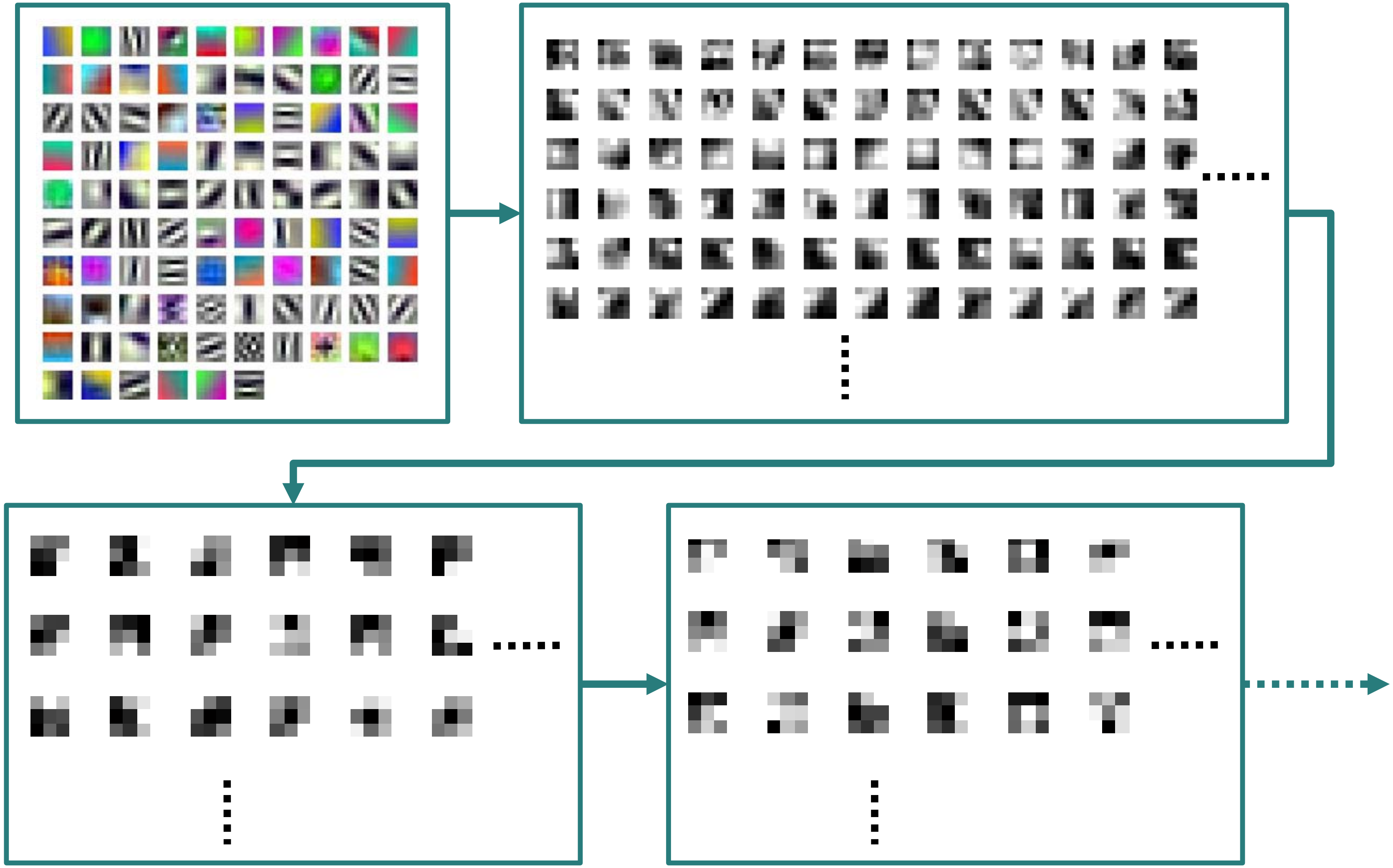


Understanding Composition of Convolutions

Example:



Understanding Composition of Convolutions through layers



It is a powerful representation, but deeply non-understandable

Conclusion: Composition of Convolutions is a powerful representation of complex spatial features where the hierarchy of layers is maximizing the representation power

But, explaining based on visualizing weights ...

Pros:

- Easy and Understandable **for the 1st layer**

Contras:

- No interpretation for the rest of layers

We need to use other tools to understand ...

Index of this Lecture:

Preliminary considerations

Post-hoc analysis

- Neuron Analysis
- Data Inspection
- Saliency based
- Proxy models
- Modifications
- Theoretical Analysis

Ad-hoc modelling

- Interpretable representation
- Model Renovation

A case study on a single feature *(post-hoc analysis)*

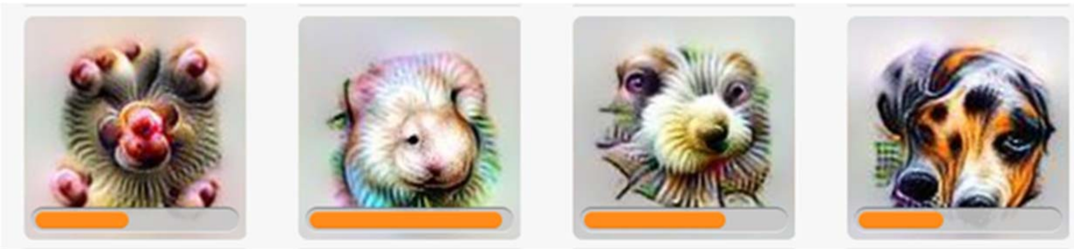
How color is represented in a CNN? and parallelisms with HVS

Neuron Analysis

IDEA: Understand a network by visualizing the individual neurons

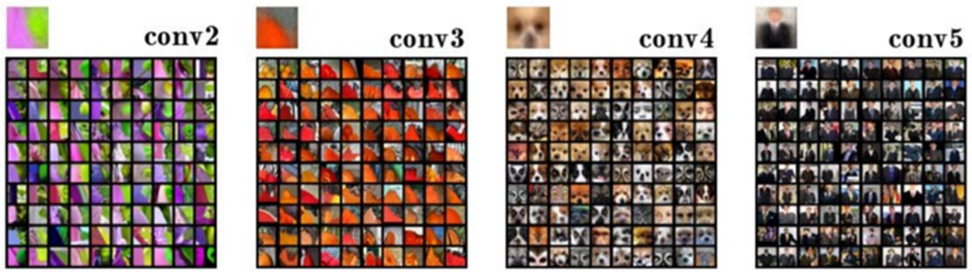
There are two main metodologies used to visualize the neuron preference:

- **Inverting-based methods:** Generate the image that produces a specific activation



Olah, et al., "The Building Blocks of Interpretability", Distill, 2018

- **Activation maximization methods:** Find the images that maximally activate a neuron



Rafegas, I.,et.al (2020). Understanding trained CNNs by indexing neuron selectivity. *Pattern Recognition Letters*, *136*, 318-325.

Receptive Field

What is a the receptive field of a neuron?

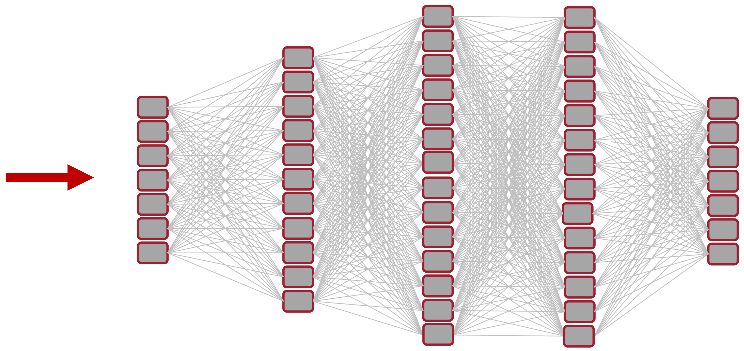
Image Patch that provokes a specific activation of a neuron for a given image

Why is it important?

Gives us the exact region of the image responsible of an activation

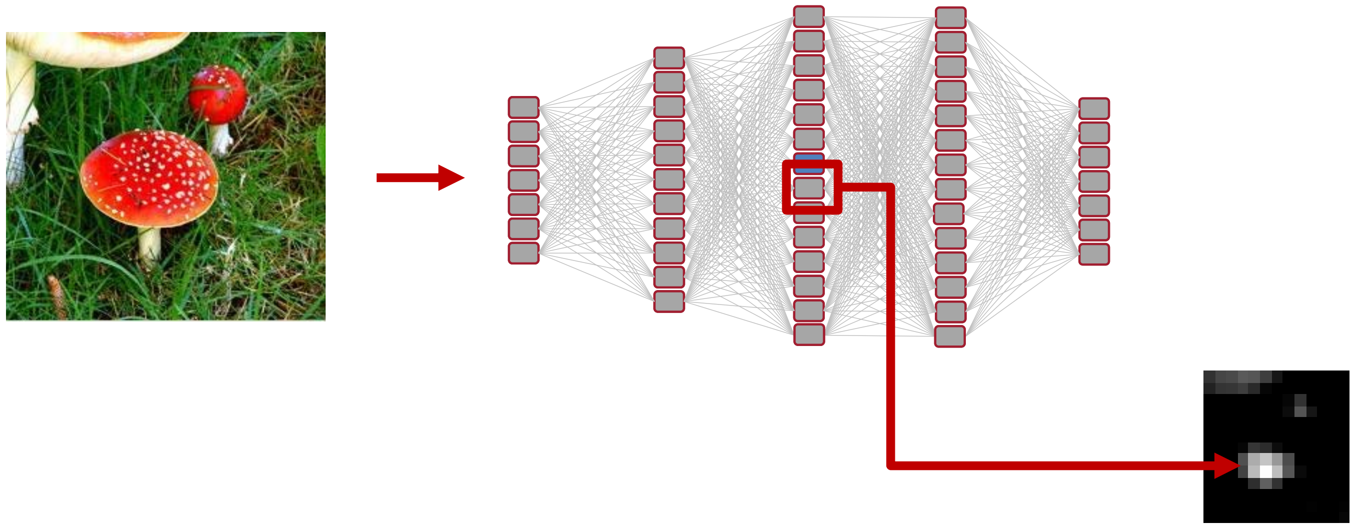
Building the Receptive field of a neuron

1) Run an image through a neural network.



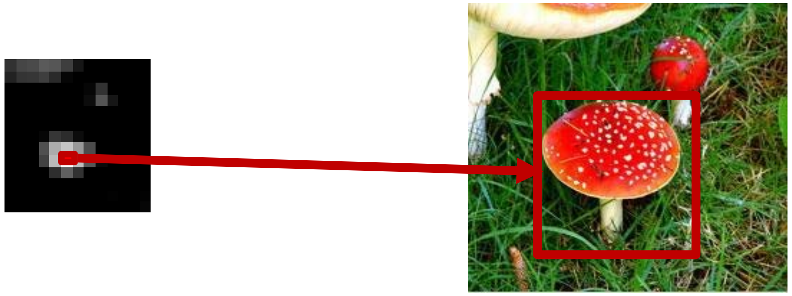
Building the Receptive field of a neuron

1) Run an image through a neural network.
2) Obtain the output of a neuron (Activation Map)



Building the Receptive field of a neuron

1) Run an image through a neural network.
2) Obtain the output of a neuron (Activation Map)
3) Find the part of the image (with size of the receptive field) that triggered the **maximum** activation in the feature map.



Receptive field
(image patch)
of a neuron
for a given image

Building the Receptive field of a neuron

1) Run an image through a neural network.
2) Obtain the output of a neuron (Activation Map)
3) **Find the part of the image (with size of the receptive field) that triggered the maximum activation in the feature map.**

Computing the corresponding indexes from Layer L to Layer L-1:

$$Region(i,j) = (stride_h \cdot (i-1) \colon stride_h \cdot (i-1) + kernel_h - 1, \\ stride_w \cdot (i-1) \colon stride_w \cdot (i-1) + kernel_w - 1)$$

Example: *kernel = 3x3 Stride = 2x2*

(1,1)	(1,2)	(1,3)	(1,4)	(1,5)	(1,6)	(1,7)	(1,8)
(2,1)	(2,2)	(2,3)	(2,4)	(2,5)	(2,6)	(2,7)	(2,8)
(3,1)	(3,2)	(3,3)	(3,4)	(3,5)	(3,6)	(3,7)	(3,8)
(4,1)	(4,2)	(4,3)	(4,4)	(4,5)	(4,6)	(4,7)	(4,8)
(5,1)	(5,2)	(5,3)	(5,4)	(5,5)	(5,6)	(5,7)	(5,8)
(6,1)	(6,2)	(6,3)	(6,4)	(6,5)	(6,6)	(6,7)	(6,8)
(7,1)	(7,2)	(7,3)	(7,4)	(7,5)	(7,6)	(7,7)	(7,8)
(8,1)	(8,2)	(8,3)	(8,4)	(8,5)	(8,6)	(8,7)	(8,8)

Region (3,4) =
(2(3-1) : 2*(3-1) +3 – 1,*
2(4-1) : 2*(4-1) +3 – 1)*
= (4:6,6:8)

Indexes of the Feature map of layer l

(1,1)	(1,2)	(1,3)	(1,4)	(1,5)
(2,1)	(2,2)	(2,3)	(2,4)	(2,5)
(3,1)	(3,2)	(3,3)	(3,4)	(3,5)
(4,1)	(4,2)	(4,3)	(4,4)	(4,5)
(5,1)	(5,2)	(5,3)	(5,4)	(5,5)

Neuron Analysis

USEFUL TOOLS

- **Distill.pub -->** Web-journal dedicated to the understanding CNN at a neuron level
  https://distill.pub/2018/building-blocks/

- **Network Dissection -->** Tool to get ativation information of each neuron
  http://netdissect.csail.mit.edu/

Index of this Lecture:

Preliminary considerations

Post-hoc analysis

- Neuron Analysis
- Data Inspection
- Saliency based
- Proxy models
- Modifications
- Theoretical Analysis

Ad-hoc modelling

- Interpretable representation
- Model Renovation

A case study on a single feature *(post-hoc analysis)*

How color is represented in a CNN? and parallelisms with HVS

Data Inspection

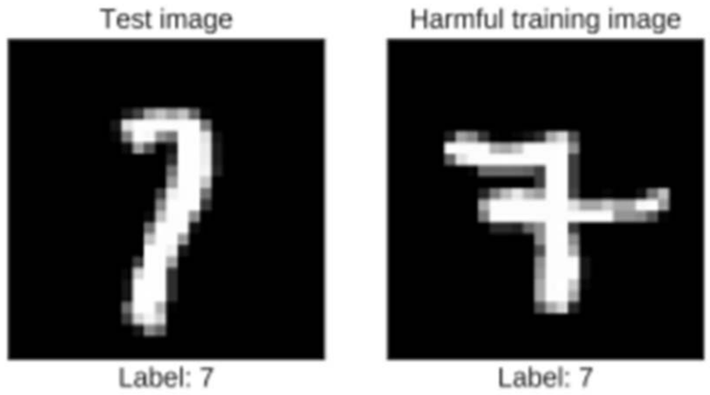
IDEA: Study the dataset to understand possible training bias

- **Study harmful Images on predictions**
  - Find the most similar images in the dataset
  - Positive influence if they share label / Negative otherwise

•**Why is it usefull**: help identify mis-annotated labels and outliers existing in the data

- •Incorrectly labeled images
- •Similar images belonging to different classes
- •Context over object

•



Koh, P. W., & Liang, P. (2017, July). Understanding black-box predictions via influence functions. In *International Conference on Machine Learning* (pp. 1885-1894). PMLR.

Data Inspection

Example

- Find incorrectly classified test data
- Use algorithm to find similar images in the train set (handcrafted descriptor KNN)
- Check if the error comes from wrongly labeled train data / similar class

Prediction: Polecat
True: mouse

Polecat

Most similar image
in the training set

Class: Polecat

Wrong label!!!

Index of this Lecture:

Preliminary considerations

Post-hoc analysis

- Neuron Analysis
- Data Inspection
- Saliency based
- Proxy models
- Modifications
- Theoretical Analysis

Ad-hoc modelling

- Interpretable representation
- Model Renovation

A case study on a single feature *(post-hoc analysis)*

How color is represented in a CNN? and parallelisms with HVS

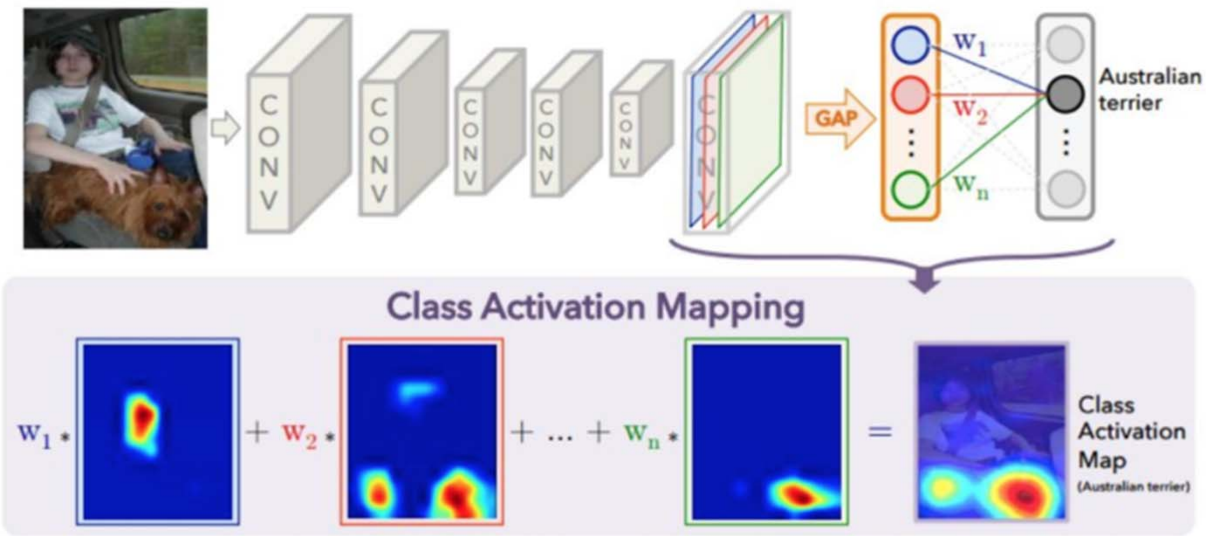
Saliency based

IDEA: Visualize the attributes of input data being more relevant to a prediction. Relevant attributes are highlighted as saliency maps

Saliency maps can be used for understanding and for improving CNN:

- **Class Activation Maps (CAM):** For each possible class prediction, highlights the regions of the image that contributed the most

- **Use saliency maps to improve performance:** Add saliency information as an aditional imput of the CNN to ensure that the CNN is taking into acount only the selected areas of interest

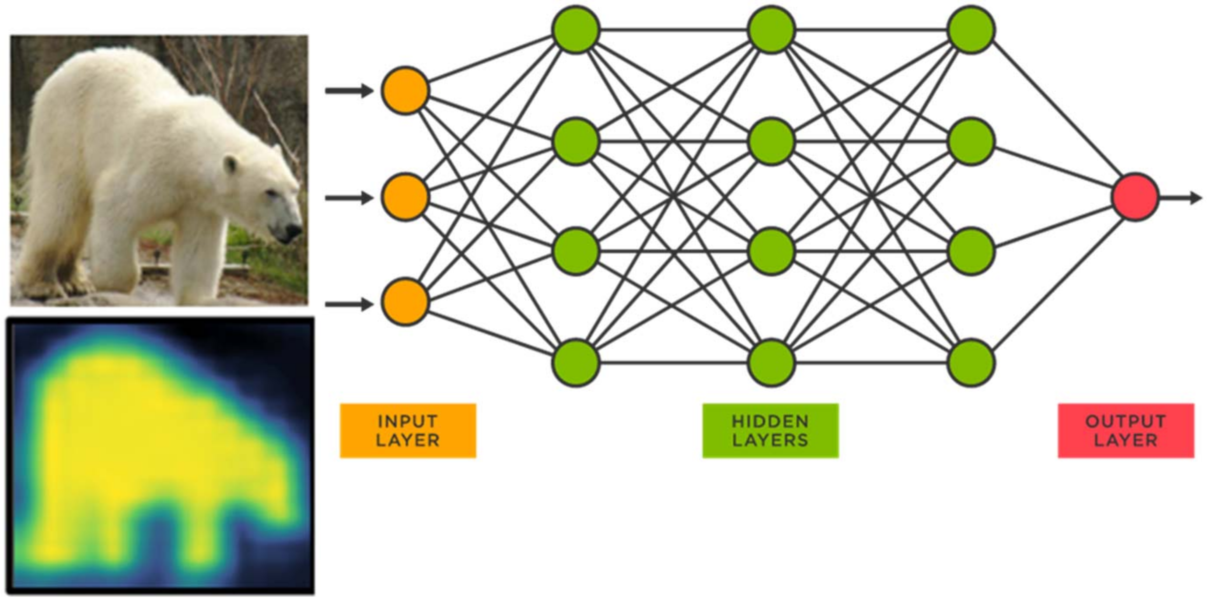


Selvaraju, R. R., Cogswell, M., Das, A., Vedantam, R., Parikh, D., & Batra, D. (2017). Grad-cam: Visual explanations from deep networks via gradient-based localization. In Proceedings of the IEEE international conference on computer vision (pp. 618-626).

Saliency based

Example metodology

- Check the saliency map of an incorrectly classified image
- Use saliency to correct what should be important



Polar Bear

Index of this Lecture:

Preliminary considerations

Post-hoc analysis

- Neuron Analysis
- Data Inspection
- Saliency based
- Proxy models
- Modifications
- Theoretical Analysis

Ad-hoc modelling

- Interpretable representation
- Model Renovation

A case study on a single feature *(post-hoc analysis)*

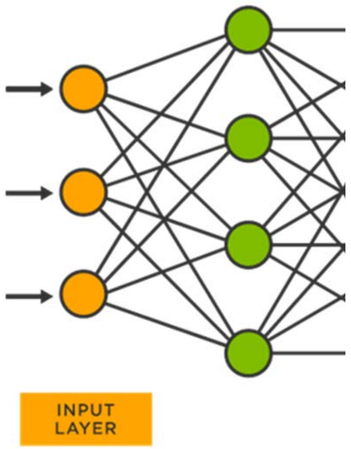
How color is represented in a CNN? and parallelisms with HVS

Proxy model

IDEA: Create an alternative model that performs similarly to the DNN

There are three main methodologies:

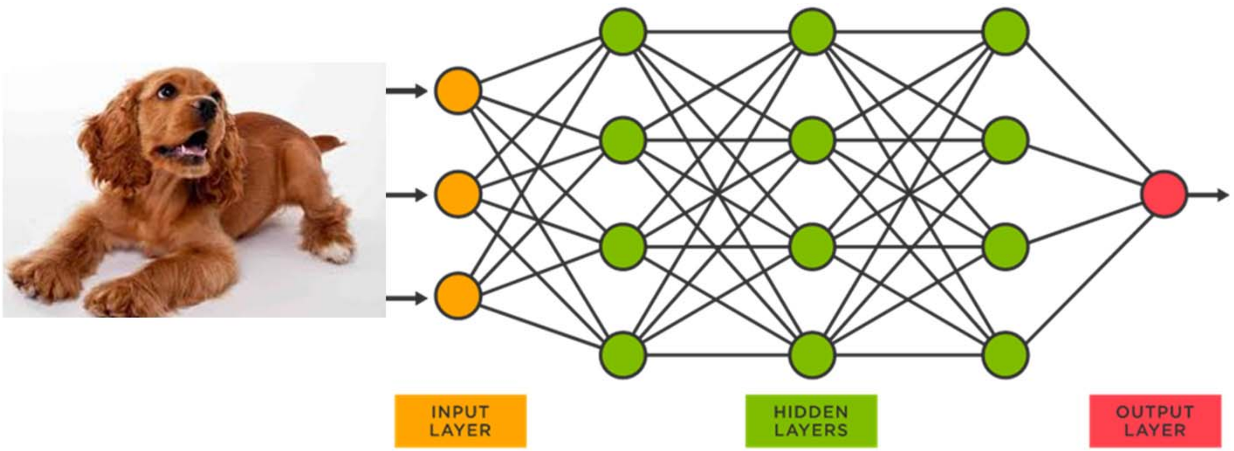
- **Simplistic models:** Create a decision tree or a set of rules that mimics the behaviour of the CNN
- **Knowledge distillation:** Use soft labels to better understand the classification process (I.e: Breed, specie, animal). Can be used to improve training.
- **Local Interpretable Modelagnostic Explanation:** Cut the NN at a certain point to study intermediate representations



Proxy model

Example (Knowledge distilation)

- Use soft labels to locate the error in classification
- Use soft label information in the loss function to improve training



Canine 90%
Feline 50%
Human 20%
...
Furniture 5%

Lion 90%
Cocker 85%
Cavalier 80%
...
Table 5%

Index of this Lecture:

Preliminary considerations

Post-hoc analysis

- Neuron Analysis
- Data Inspection
- Saliency based
- Proxy models
- Modifications
- Theoretical Analysis

Ad-hoc modelling

- Interpretable representation
- Model Renovation

A case study on a single feature *(post-hoc analysis)*

How color is represented in a CNN? and parallelisms with HVS

Modifications

IDEA: Neural networks inputs aditional information providing some insight of why a decision was taken

There are two main methodologies:

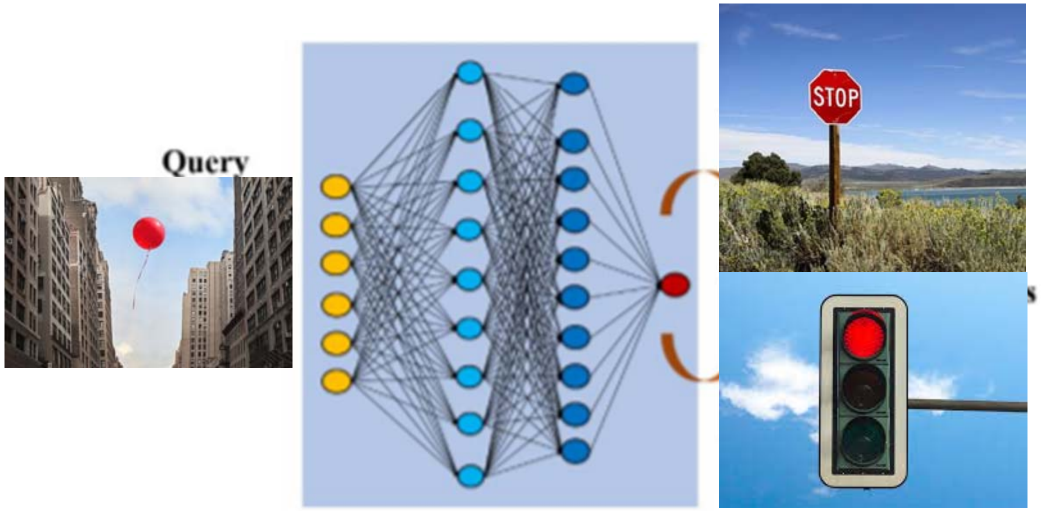
- Explaining by case
- Explaining by text

Explaining-by-Case

IDEA: Give similar examples of training instances that have a similar label

- Given a query the CNN outputs a label as well as nearest neighbors of that activation values in the training set

- Visual information provided by this function may help the user understand why a certain decision was made

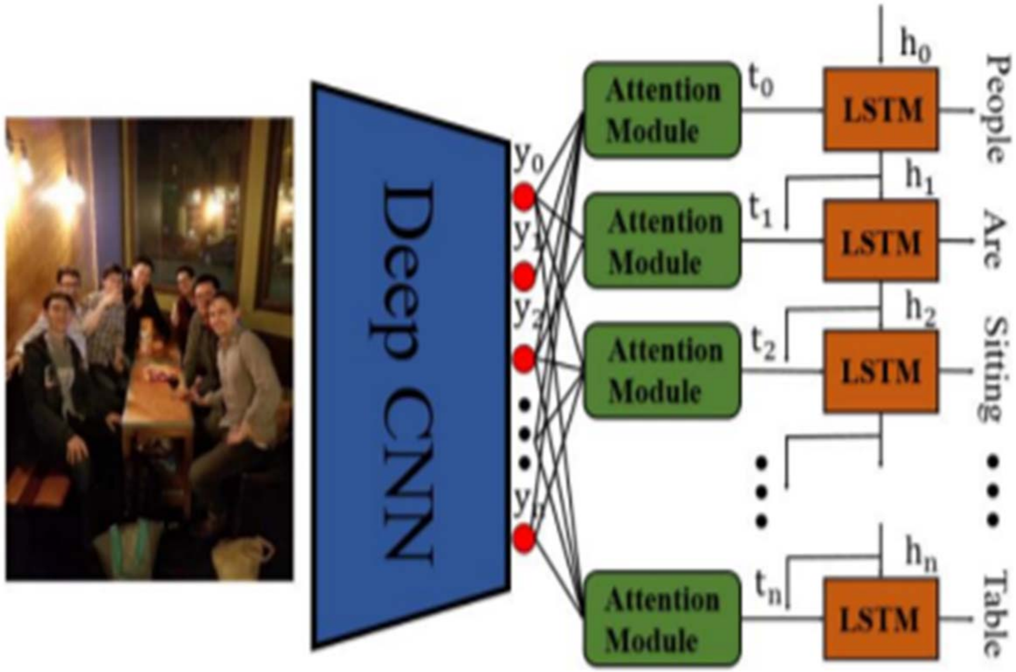


S. Wachter, B. Mittelstadt and C. Russell, "Counterfactual Explanations without Opening the Black Box: Automated Decisions and the GPDR," Harv. JL & Tech., vol. 31, no. 841, 2017

Explaining-by-Text

IDEA: Give a text explanation of the output instead of just a label

- Combination of Attention modules and LSTM:
  - Attention: Detects the most important objects in the image
  - LSTM: Produces a sentence based on ordered attention



A. Karpathy, L. Fei-Fei, "Deep visual-semantic alignments for generating image descriptions," In CVPR, pp. 3128-3137, 2015

Index of this Lecture:

Preliminary considerations

Post-hoc analysis

- Neuron Analysis
- Data Inspection
- Saliency based
- Proxy models
- Modifications
- $\boxed{\text{Theoretical Analysis}}$

Ad-hoc modelling

- Interpretable representation
- Model Renovation

A case study on a single feature *(post-hoc analysis)*

How color is represented in a CNN? and parallelisms with HVS

Theoretical Analysis

IDEA: Study of the DNN architecture from a mathematical point

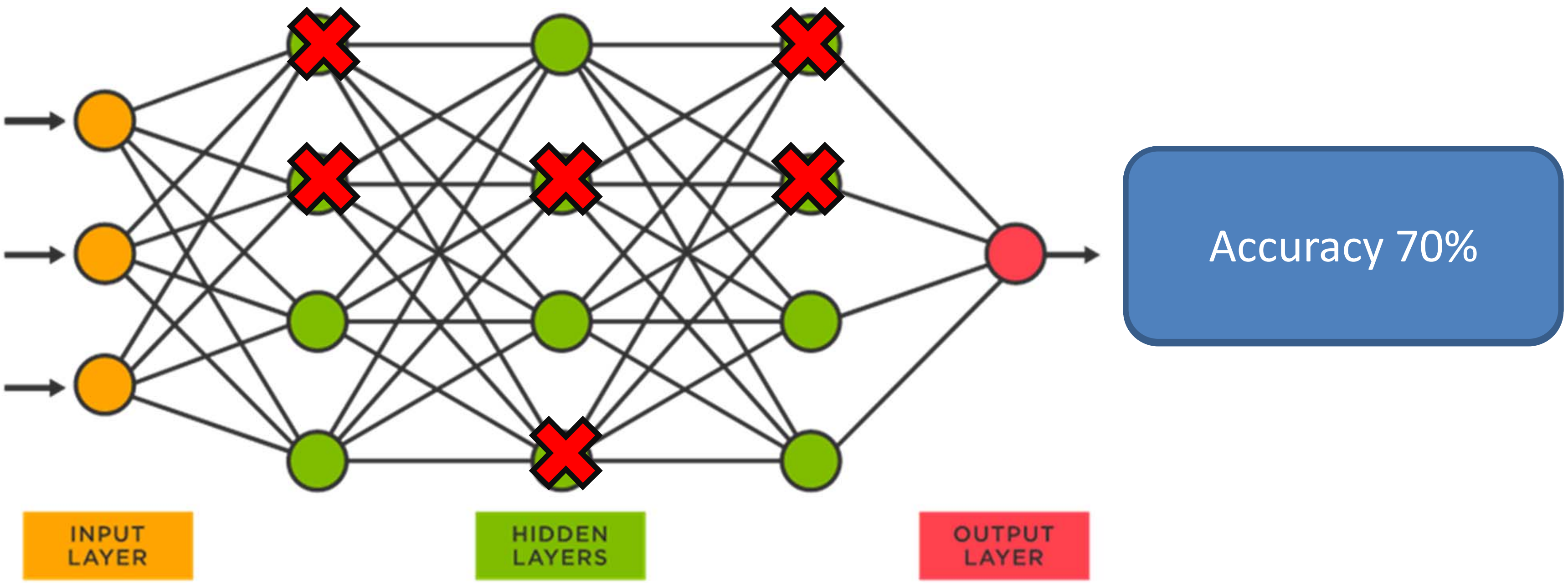
There are three main areas of study

- **Representation:** Deeper networks are more expressive than shallow ones
- **Optimization:** The number of parameters in a network exceeds the number of data instances (try to find the minimum number of parameter with best performance)
- **Generalization:** Explain why a deep network can generalize well despite the number of parameters is greater than the number of data samples

Theoretical Analysis

Example: Ablation

- Use ablation to find the optimal number of parameters
- In Pytorch: **import** torch.nn.utils.prune **as** prune
  - https://pytorch.org/tutorials/intermediate/pruning_tutorial.html



Index of this Lecture:

Preliminary considerations

Post-hoc analysis

- Neuron Analysis
- Data Inspection
- Saliency based
- Proxy models
- Modifications
- Theoretical Analysis

Ad-hoc modelling

- Interpretable representation
- Model Renovation

A case study on a single feature *(post-hoc analysis)*

How color is represented in a CNN? and parallelisms with HVS

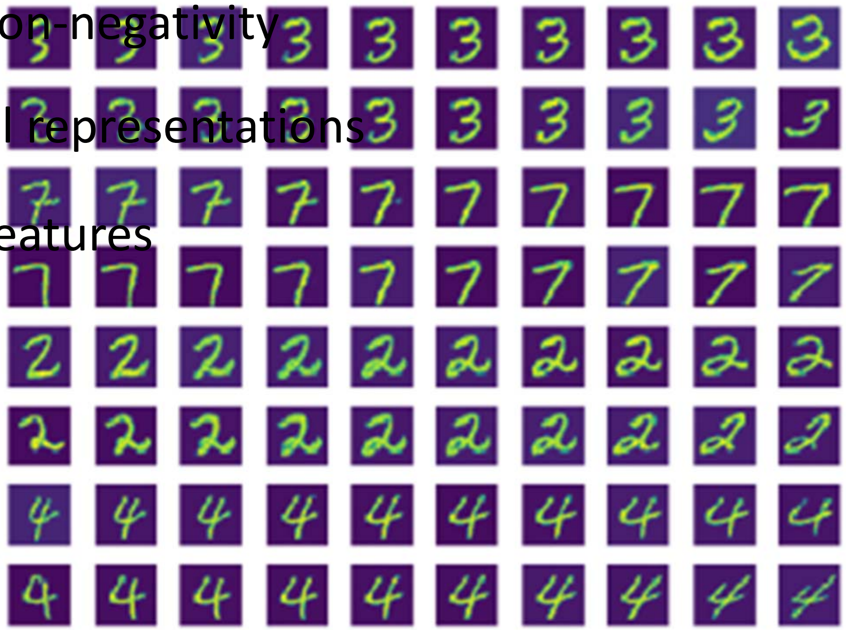
Interpretable representation

IDEA: Create a DNN ensuring that the neurons are sensitive to known or controlled stimulus by using regularization techniques

Regularization techniques steer the optimization towards more interpretable representations with the following properties:

- Decomposability: Neurons represent simple concepts
- Mathematical constrains: monotonicity or Non-negativity
- Sparsity: Maximice the difference of internal representations
- Human-in-the-loop prior: Use handcrafter features

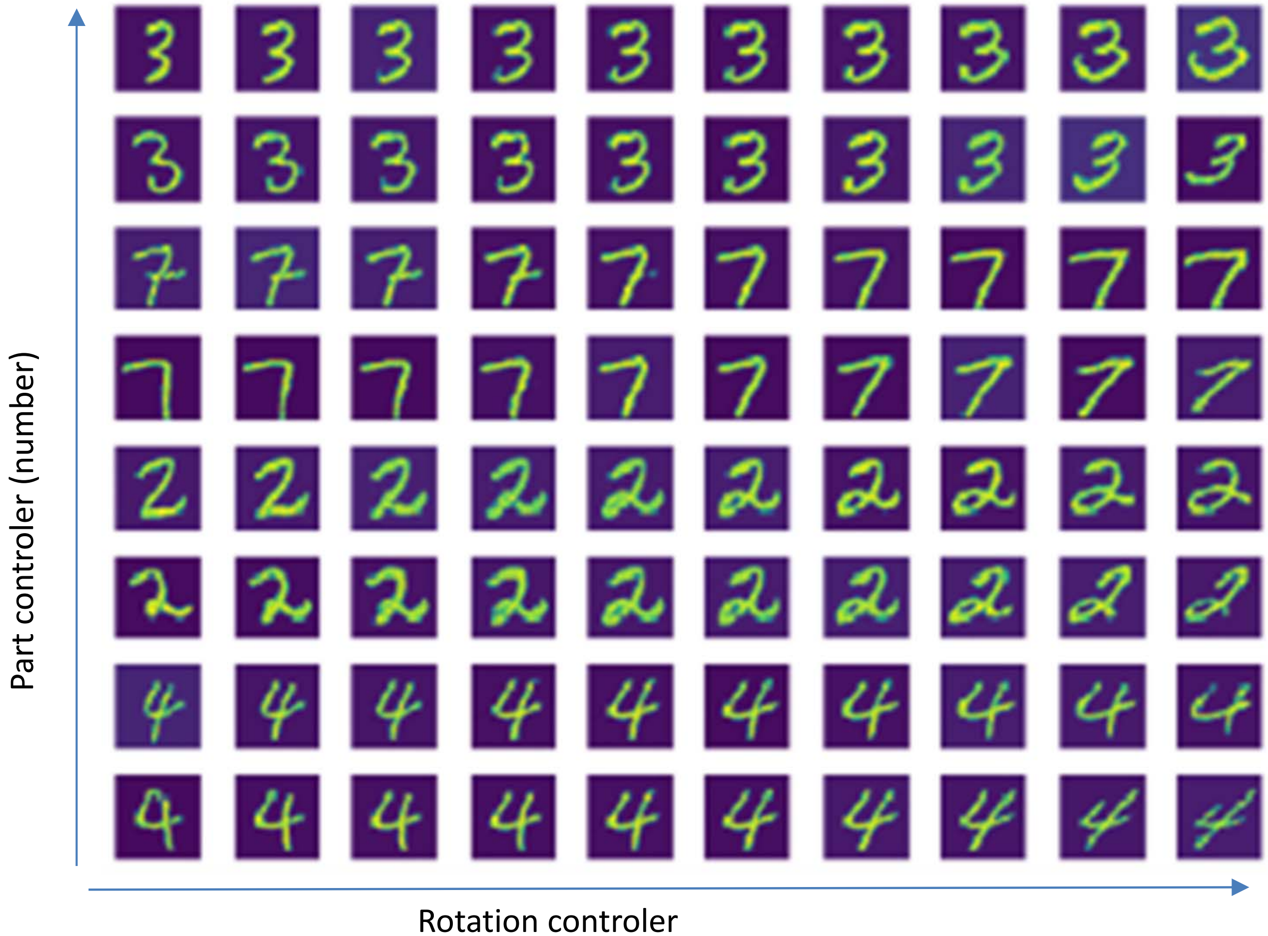
In an InfoGAN, two latent codes control the localized parts and rotation parts respectively



X. Chen, Y. Duan, R. Houthooft, J. Schulman, I. Sutskever, and P. Abbeel, “Infogan: Interpretable representation learning by information maximizing generative adversarial nets,” In NeurIPS, 2016

Interpretable representation

IDEA: Create a DNN ensuring that the <u>neurons</u> are sensitive to known or controlled stimulus by using regularization techniques



In an InfoGAN, two latent codes control the localized parts and rotation parts respectively

Internal features aligned with the PCA of the training data

X. Chen, Y. Duan, R. Houthooft, J. Schulman, I. Sutskever, and P. Abbeel, “Infogan: Interpretable representation learning by information maximizing generative adversarial nets,” In NeurIPS, 2016

Index of this Lecture:

Preliminary considerations

Post-hoc analysis

- Neuron Analysis
- Data Inspection
- Saliency based
- Proxy models
- Modifications
- Theoretical Analysis

Ad-hoc modelling

- Interpretable representation
- Model Renovation

A case study on a single feature *(post-hoc analysis)*

How color is represented in a CNN? and parallelisms with HVS

Model renovation

IDEA: Seek interpretability by means of designing and deploying more interpretable machineries into a network

There are three main methodologies

- **Neuron with purposely designed activation function**
  Some parameters of the neurons can be modified externally

- **Inserted layer with a special functionality**
  Intermediate layers output or interlayer connections

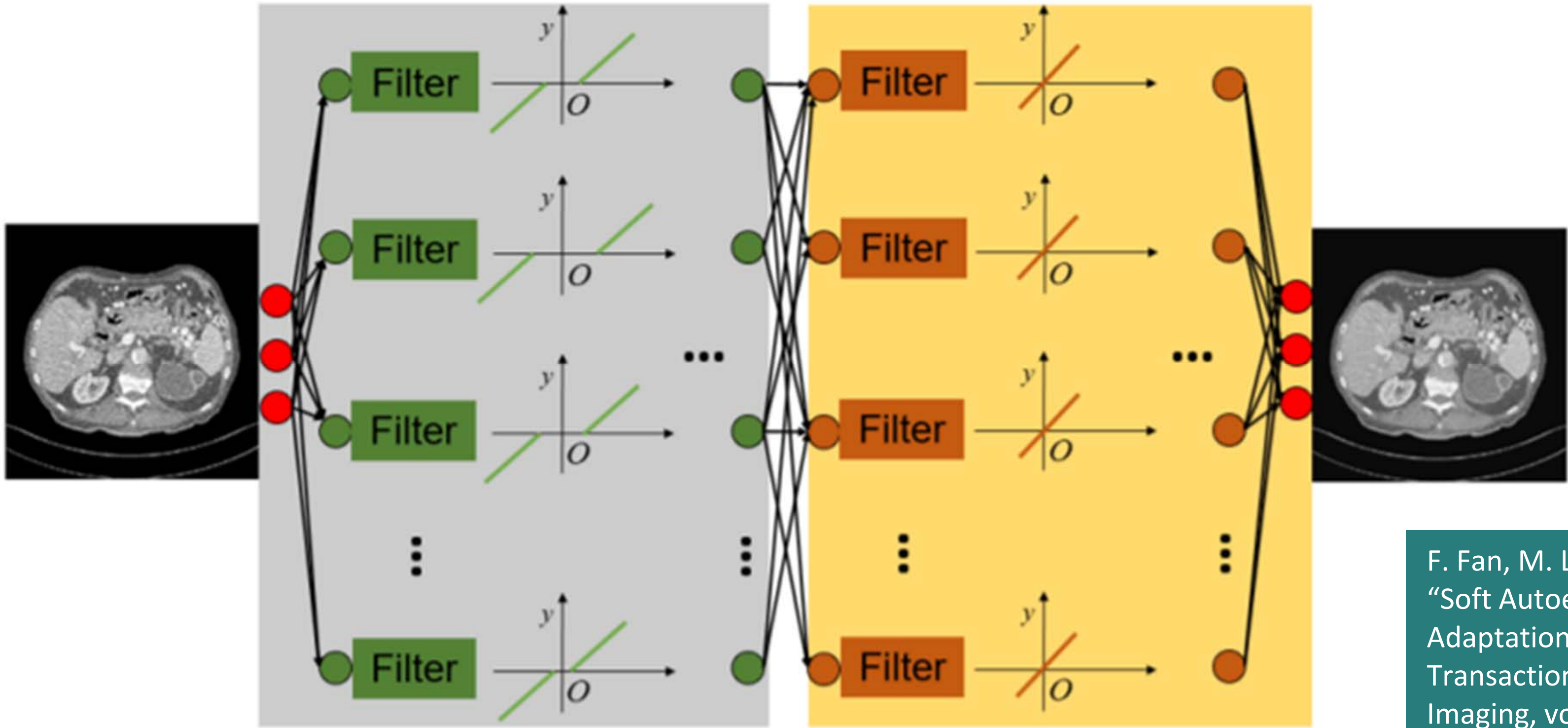
- **Modularized architecture**
  Neural Network is a sum of simple modules

Model renovation

IDEA: Seek interpretability by means of designing and deploying more <u>interpretable machineries</u> into a network

Example of Neurons with a soft Autoencoder

The threshold values of the RELU function can be modified at any point and act as a wavelet filter



F. Fan, M. Li, Y. Teng and G. Wang, "Soft Autoencoder and Its Wavelet Adaptation Interpretation," IEEE Transactions on Computational Imaging, vol. 6, pp. 1245-1257, 2020

Interpretation Model Properties

In order to create a good interpretation model we need to take into account this properties

Exactness: how accurate an interpretation method is.

Consistency: there is no contradiction in an explanation.

Completeness: show effectiveness in support of the maximal number of data instances and data types.

Universality: universal interpreter that deciphers many models.

Reward: What are gains from the improved understanding.

Why Is Interpretability Difficult?

•**Human Limitation**: As humans we are limited to understanding only basic features

•**Algorithmic Complexity**: The complexity and combination of algorithms makes it very dificult to follow the dataflow.

•**Commercial Barrier**: There is an effort to make algorithms hard to understand

Index of this Lecture:

Preliminary considerations

Post-hoc analysis

- Neuron Analysis
- Data Inspection
- Saliency based
- Proxy models
- Modifications
- Theoretical Analysis

Ad-hoc modelling

- Interpretable representation
- *Model Renovation*

A case study on a single feature *(post-hoc analysis)*

How color is represented in a CNN? and parallelisms with HVS