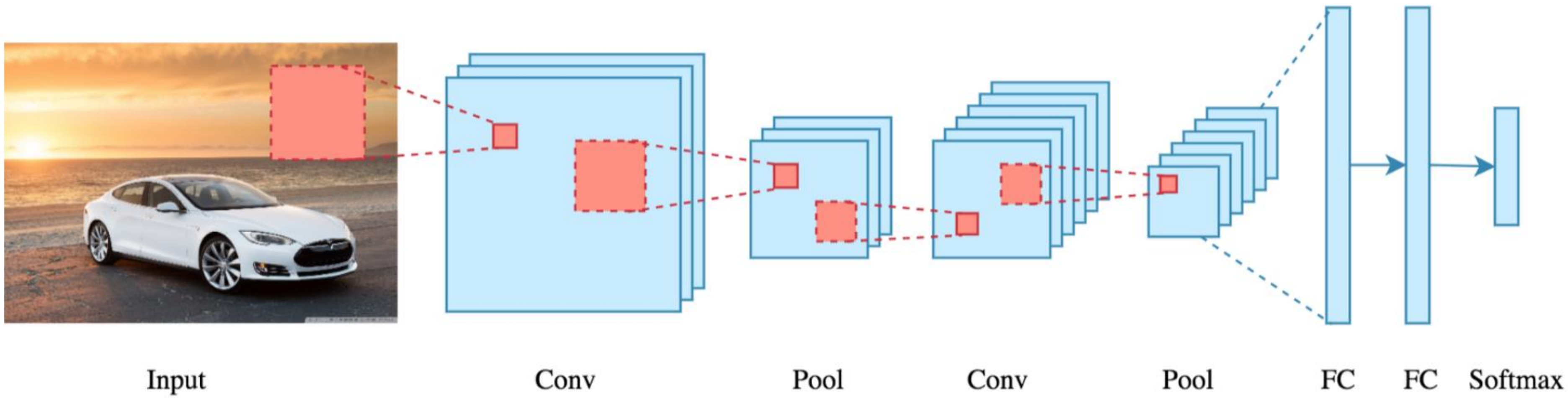
Convolutional Neural Networks

Credits
https://towardsdatascience.com/applied-deep-learning-part-4-convolutional-neural-networks-584bc134c1e2

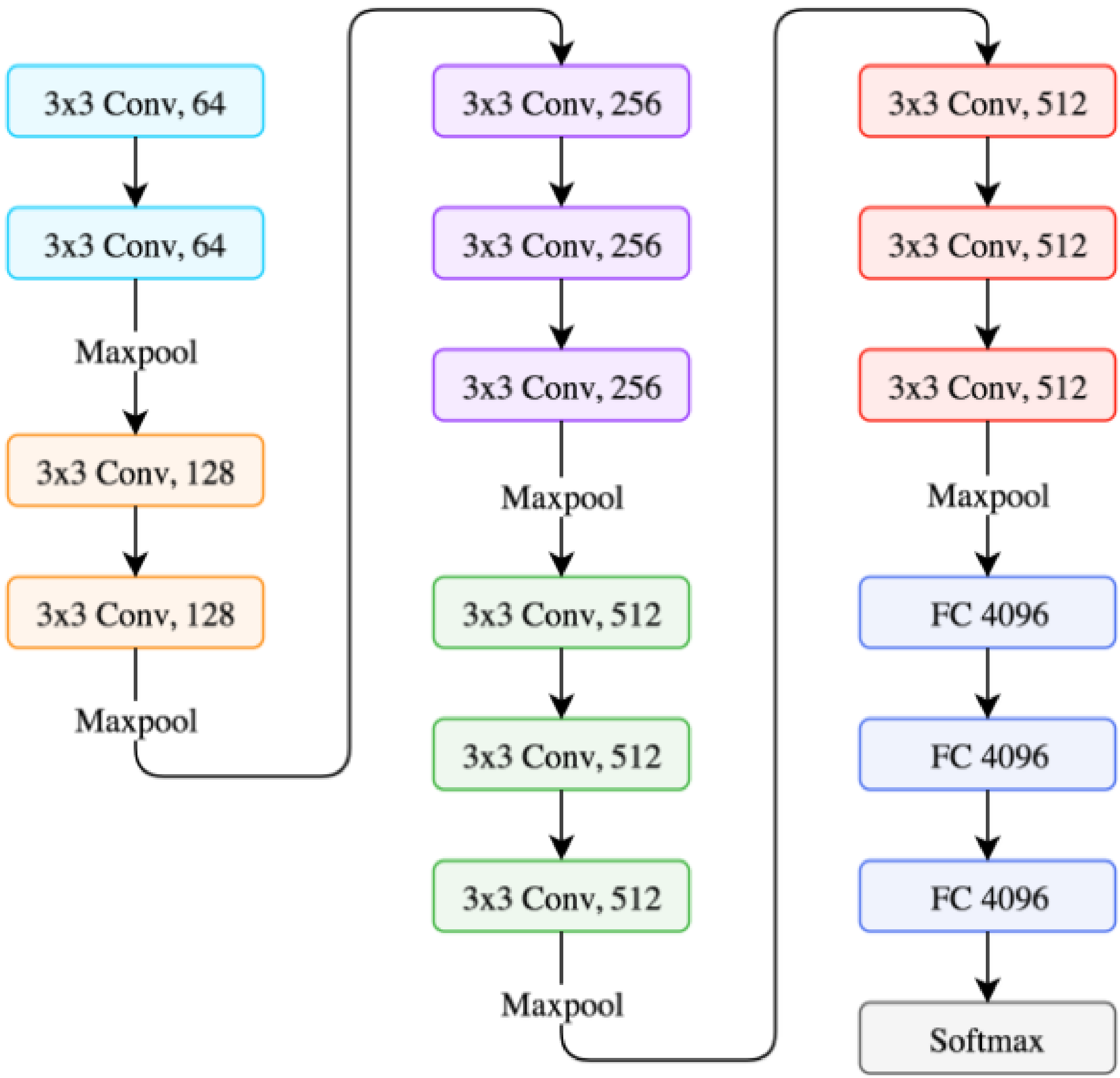
Convolutional Neural Networks (CNN)

- Very popular
- Automatically detects *features*
- Computationally efficient
- A very high level description:
  - Build edges from pixels
  - Build shapes from edges
  - Complex objects from shapes
  - (All this happens automatically – as part of learning)

CNN: Typical Architecture



CNN: VGGNet : A Practical CNN Implementation



Convolution

- It is the mathematical operation to:
  - Merge two sets of information
  - In CNN
    - Input data passed through convolution filter
      - To create a **feature map**

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

Input

1	0	1
0	1	0
1	0	1

Filter / Kernel

Convolution

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

Input

1	0	1
0	1	0
1	0	1

Filter / Kernel

1	1x1	1x0	0x1	0
0	1x0	1x1	1x0	0
0	0x1	1x0	1x1	1
0	0	1	1	0
0	1	1	0	0

Input x Filter

4	3	

Feature Map

Convolution

1	1	1	0	0
0	1	1	1	0
0	0	1x1	1x0	1x1
0	0	1x0	1x1	0x0
0	1	1x1	0x0	0x1

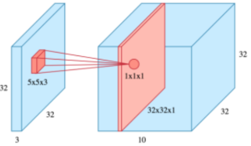
4	3	4
2	4	3
2	3	4

Convolutions and Feature Maps

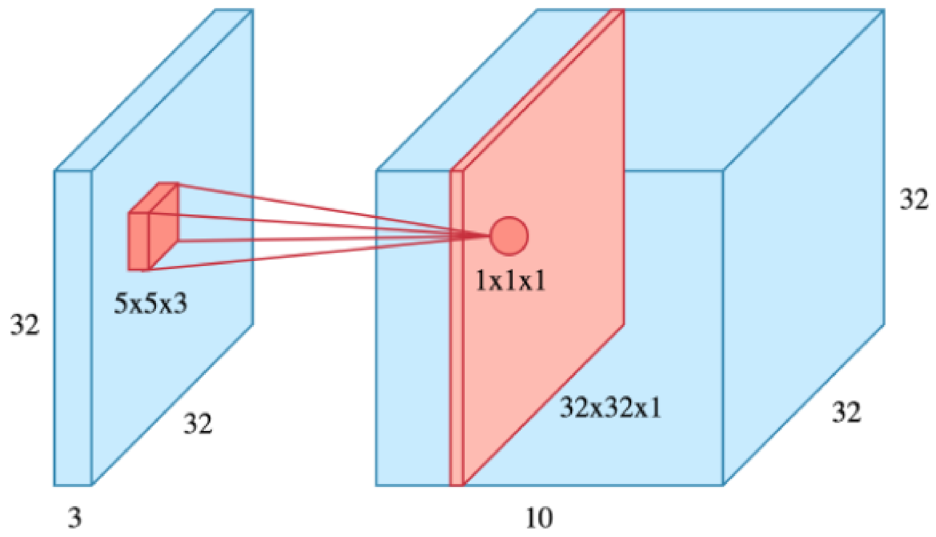
- Filters:
  - 3 X 3 or 5 X 5
  - Cover the entire depth: R-G-B layers (if image …)

- So filters are ‘3D’:
  - Filters: 3 X 3 X 3 or 5 X 5 X 3

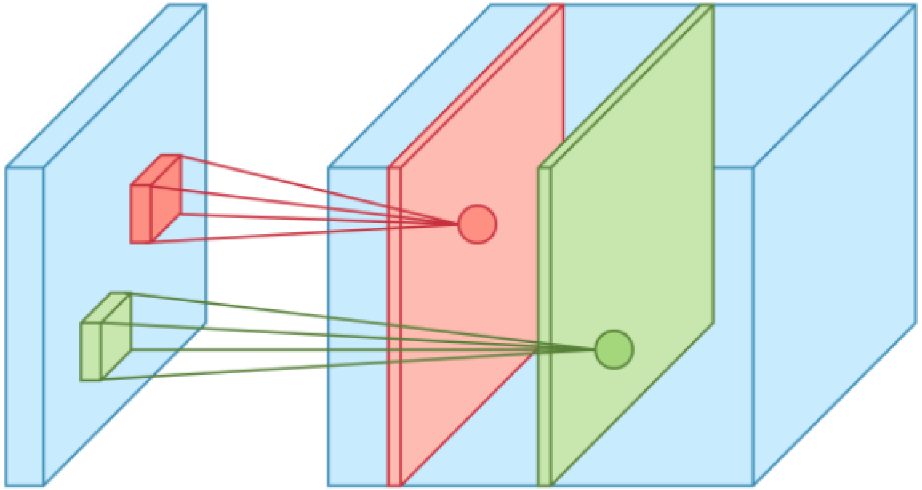
- Feature Maps:
  - Input may be processed with multiple filters
  - Each producing a feature map



Convolutions and Feature Maps

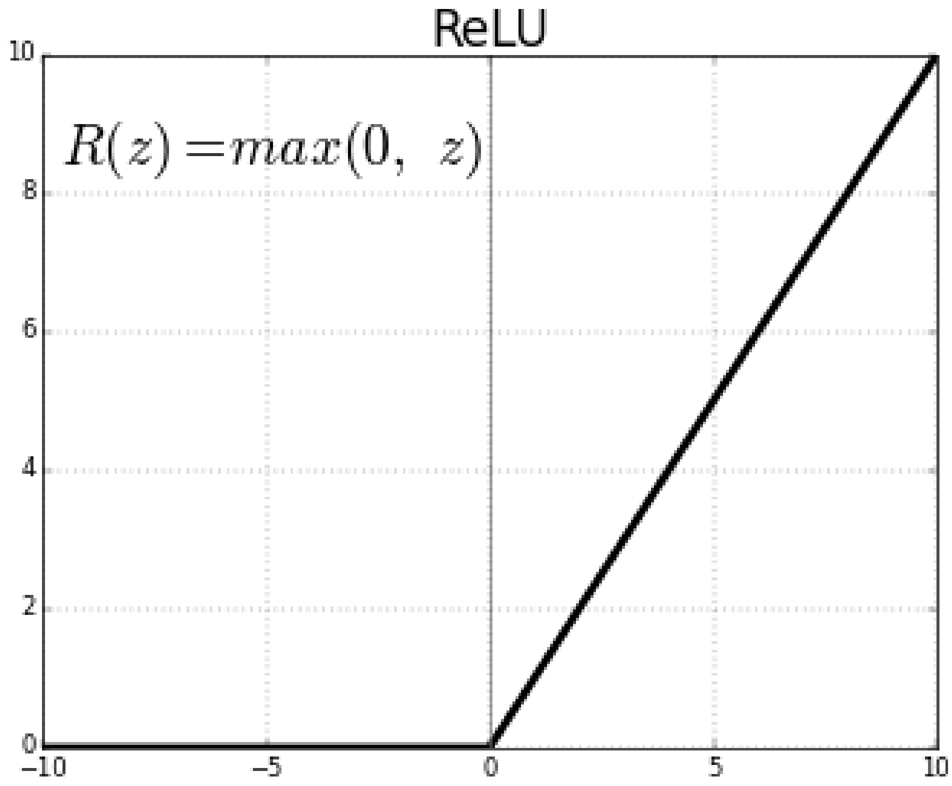


Convolutions and Feature Maps



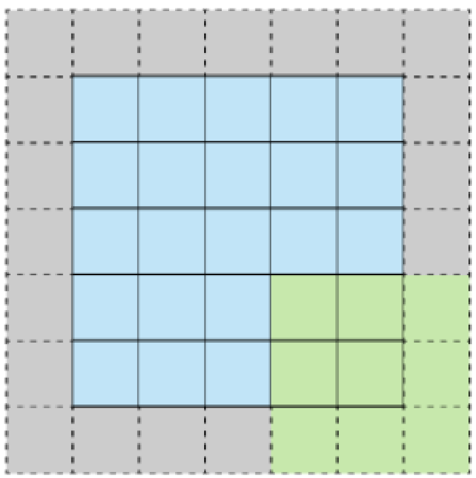
Non-Linearity in CNN

- Values in the feature maps are not sums
  - But, passed through ReLU function
  - RELU acts like a switch
  - This introduces non-linearity and makes CNN powerful

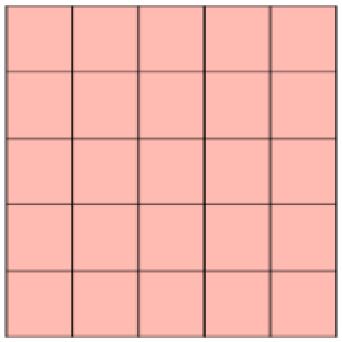


Stride and Padding

- Stride
  - The step by which the filter moves
- Padding
  - Surrounding the input with **0** or with boundary values
  - To make **conv** layer dimensions same as of the **input**



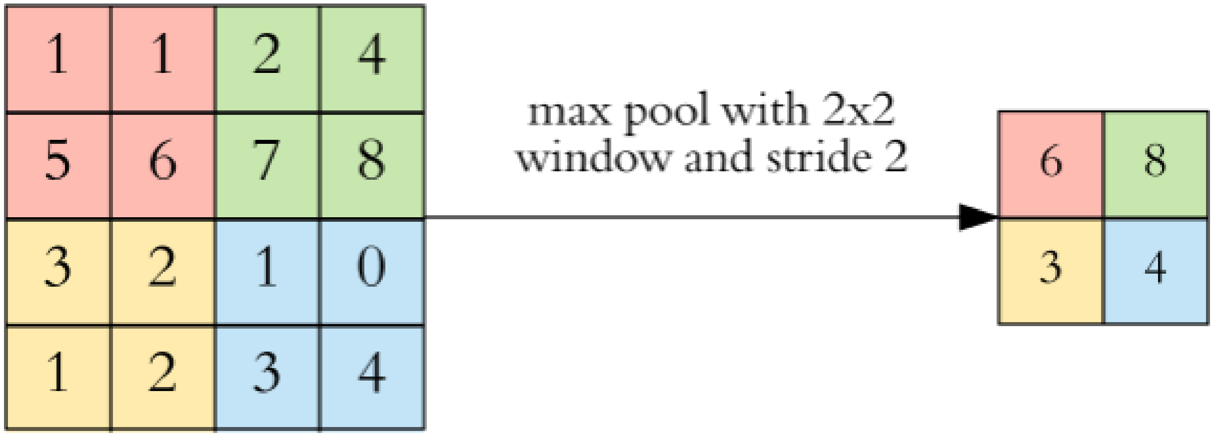
Stride 1 with Padding



Feature Map

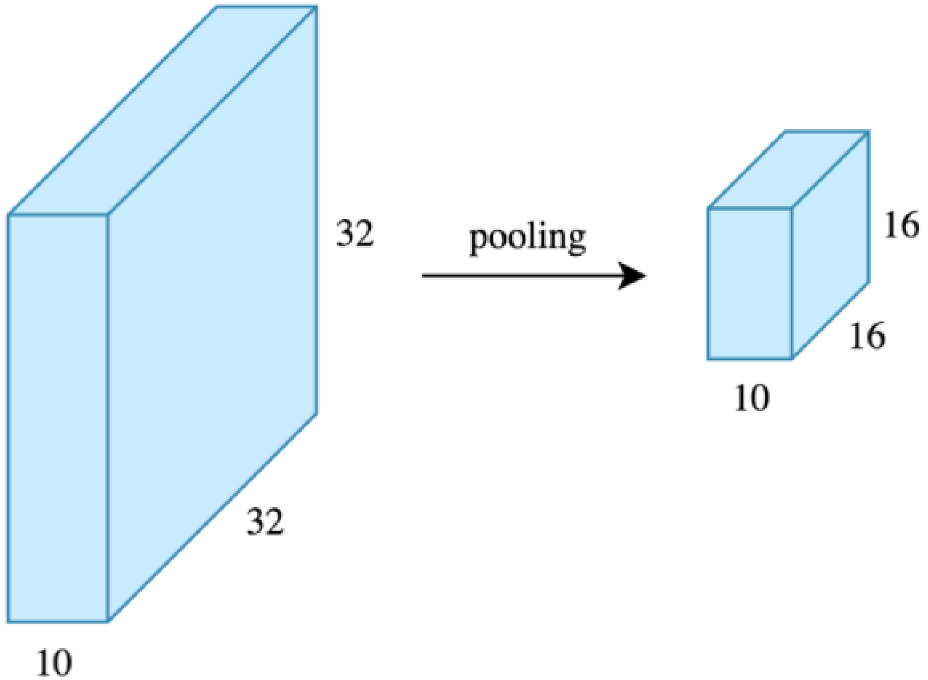
Pooling

- Goal of Pooling
  - Reduce dimensionality
  - Reduces over-fitting
  - Reduces training time
  - Enables translational, rotational, scaling invariance
- Most popular **Pooling** method
  - Max Pooling



Pooling

- Pooling reduces width and height
- But depth remains the same
- Example: effect of 2 X 2 pooling with stride = 2
  - Pooling is done without padding

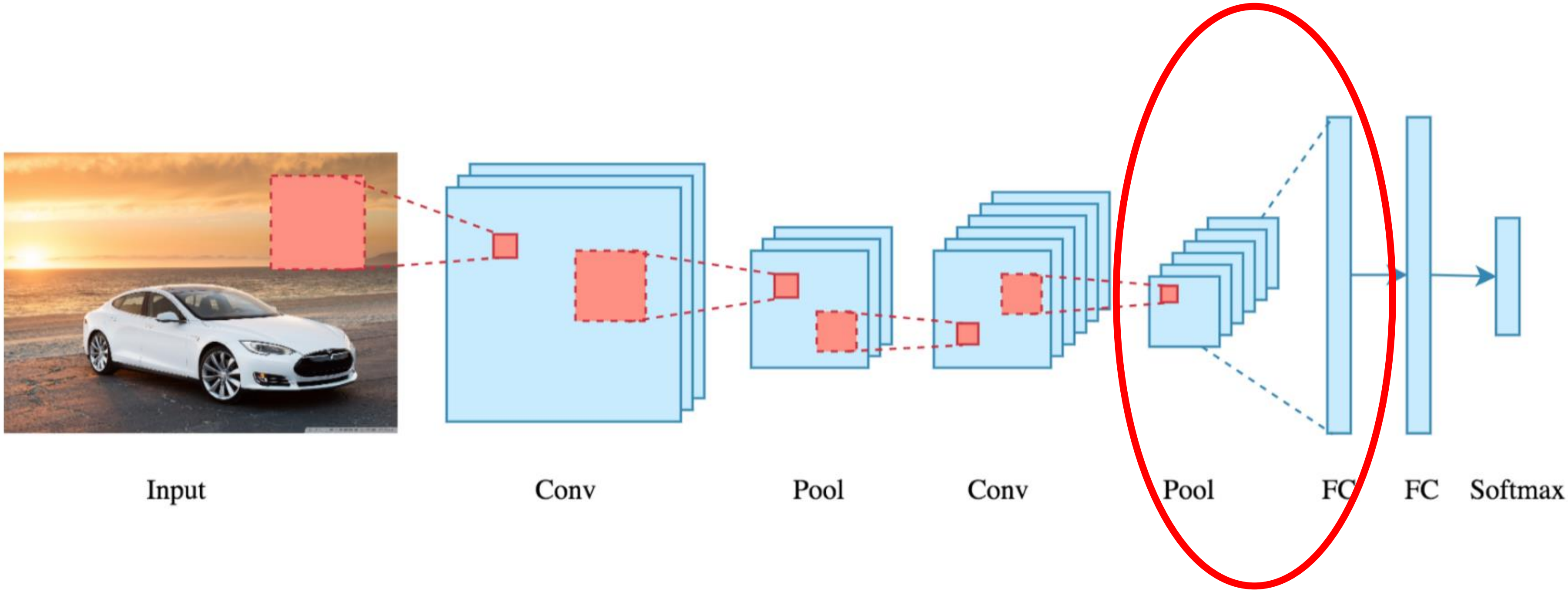


CNN: Hyperparameters

- Filter size
  - 3x3 or 5x5 or 7x7
- Filter depth
  - Same as that of input: eg. 3, corresponding to R-G-B
- Filter count
  - Power of 2
  - Typically Between 32 and 1024
    - Depends on amount of data available
    - Else, higher values ➔ over-fitting
    - Starts with small number and increases with CNN depth
- Stride
  - Typically : 1
- Padding
  - Used for conv operations, not used for pooling

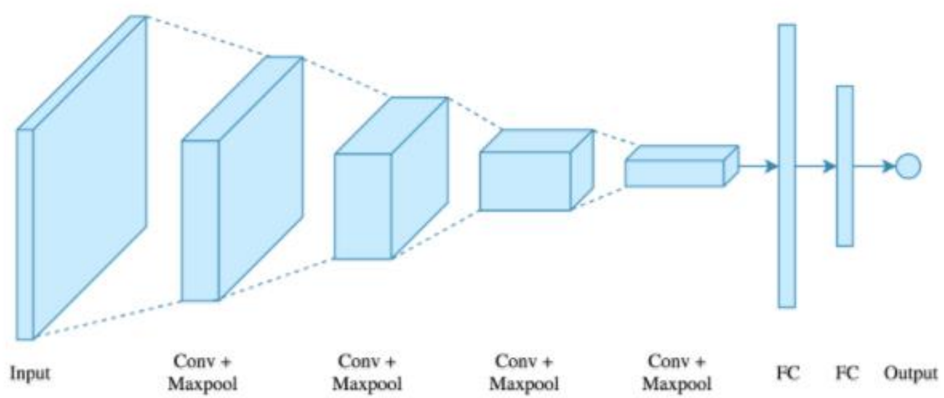
Fully Connected Layers

- Output of the final pooling layer is flattened
- And fed as input to the fully connected layer



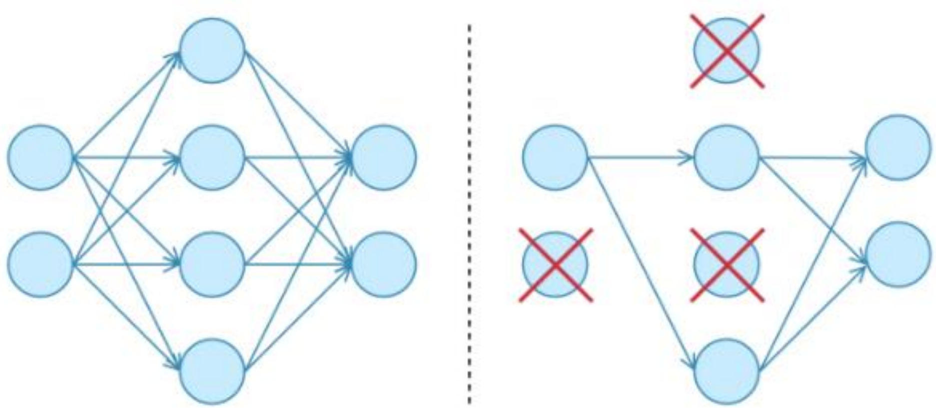
CNN Training

- Same procedure as ANN
  - Backpropagation w/ Gradient Descent
- Training process results in:
  - Feature Extraction from input data:
    - Conv layers + Pooling Layers
  - Classification: Fully Connected layers



CNN Training: Dropouts

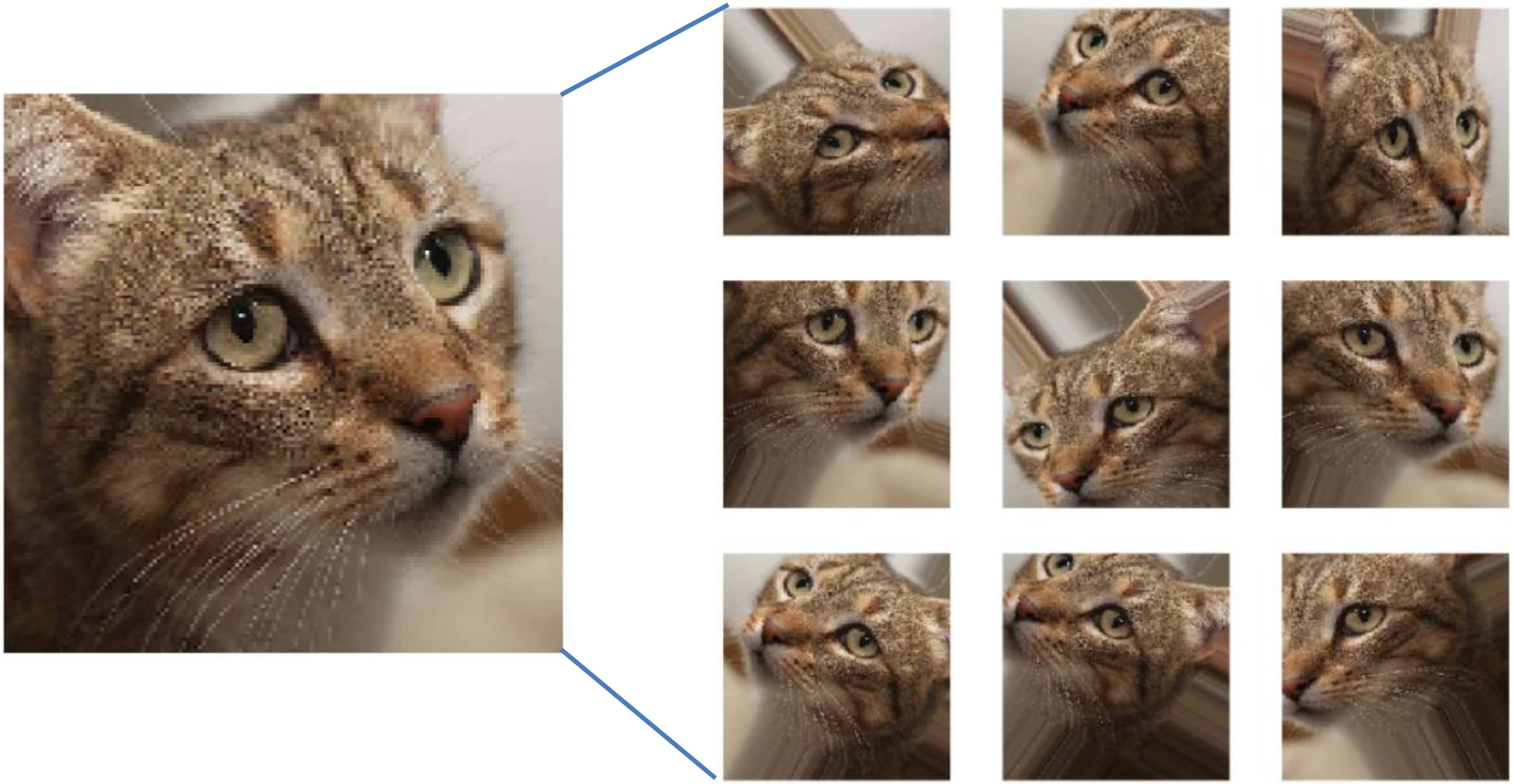
- Dropouts: Regularization technique
- Reduces over-dependence on specific nodes
  - Used to prevent over-fitting
- Method:
  - At each iteration neurons are dropped / disabled
  - Drop-out rate is about 50%: **input or hidden layers**



No Dropout With Dropout

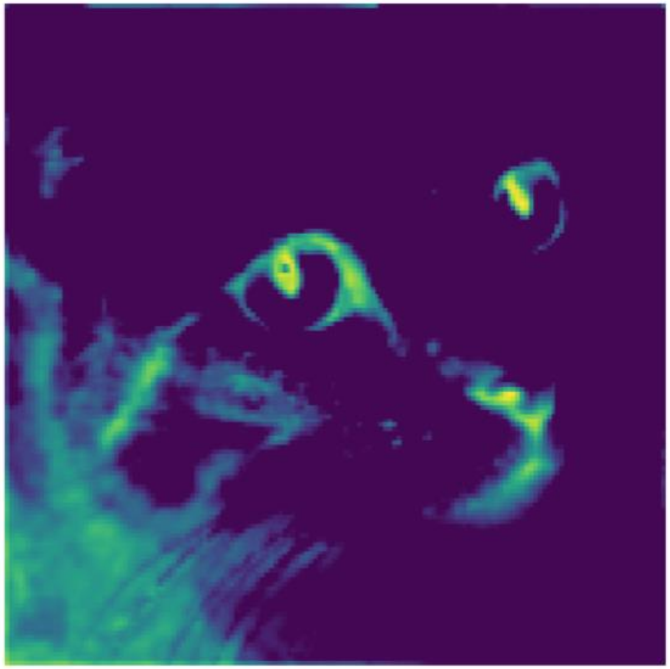
CNN Training: Data Augmentation

- Generating more data from available data-set



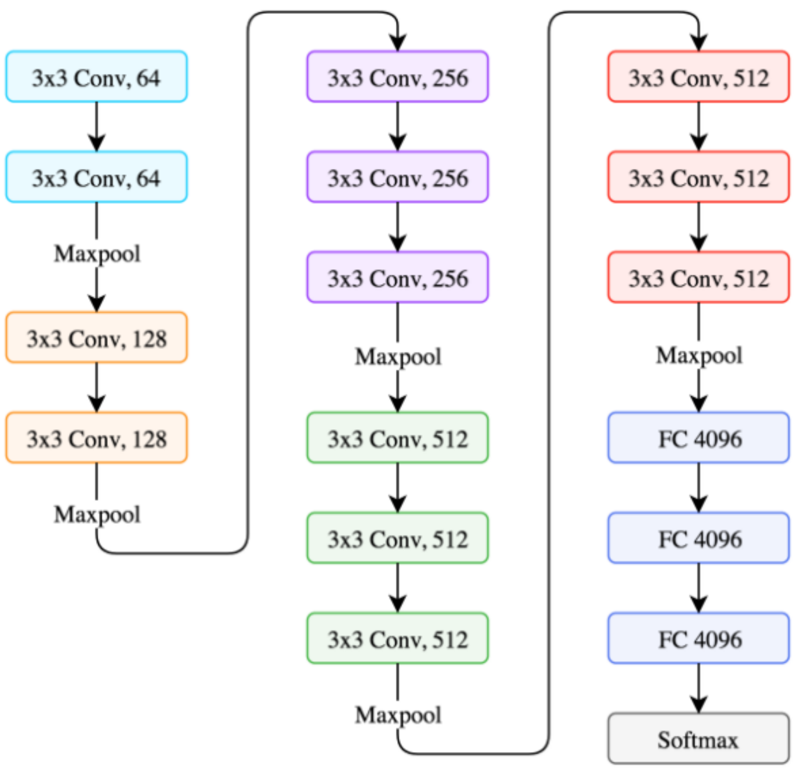
Data may be boosted 50 times !!

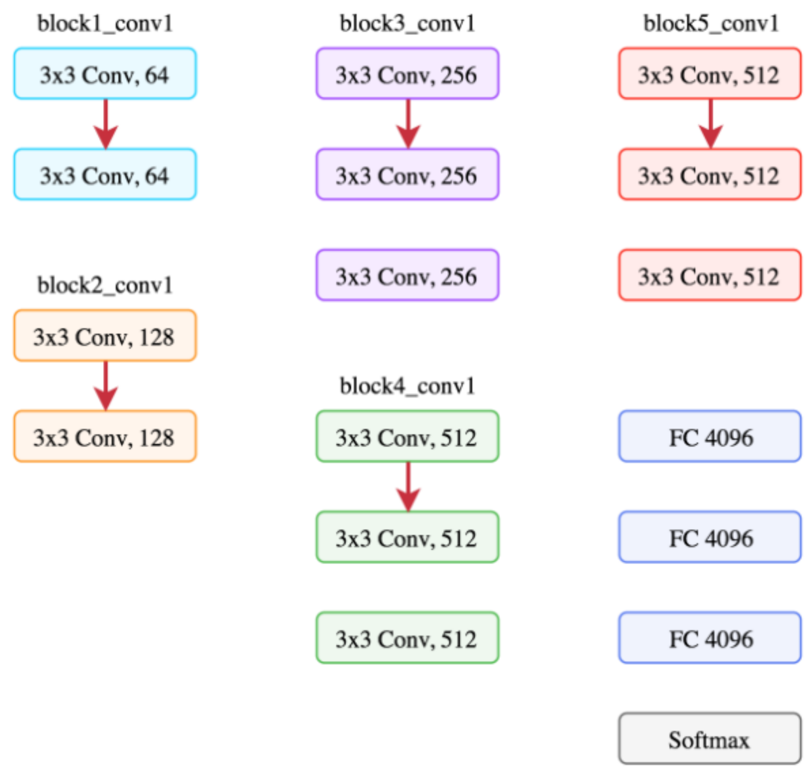
Visualizing CNN Operations: VGGNet



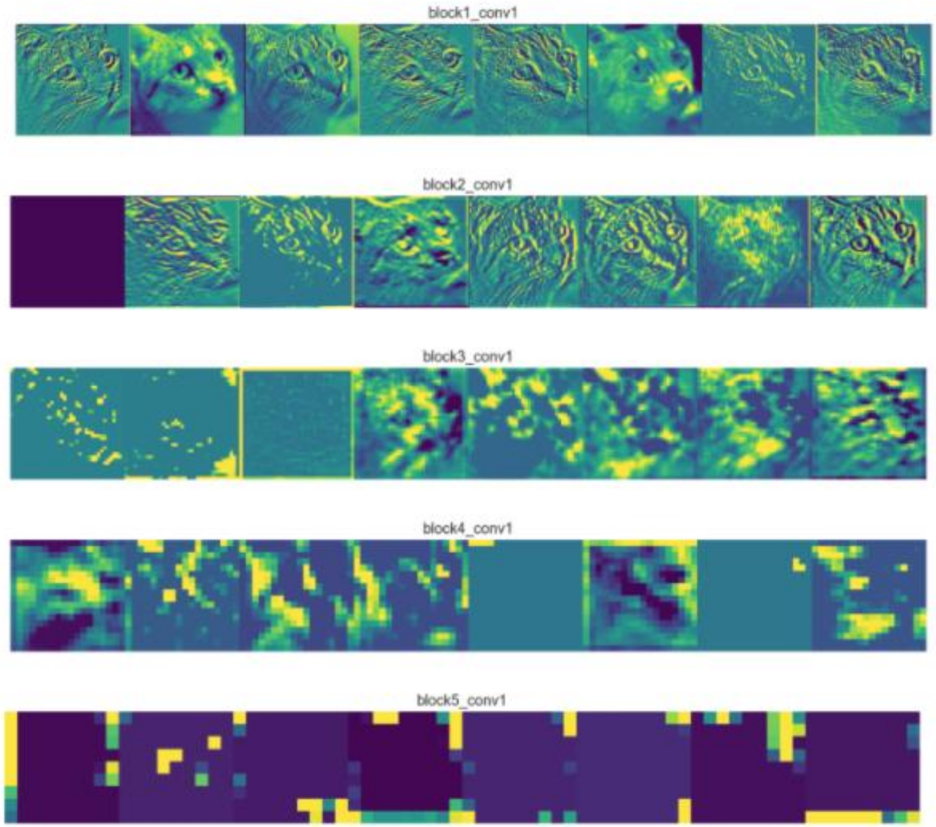
VGGNet

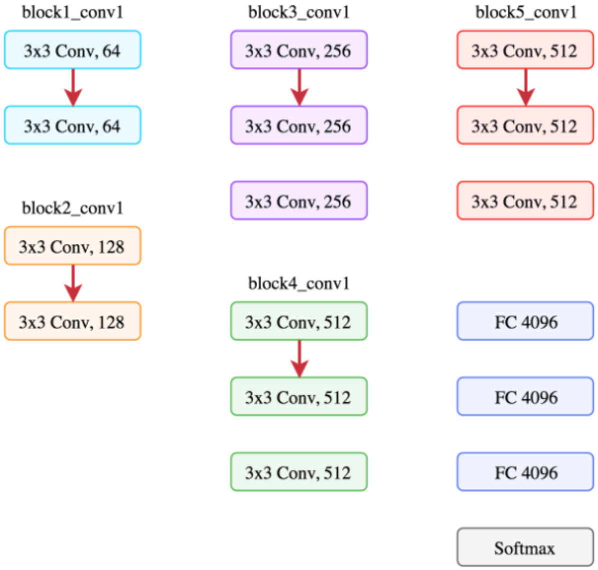
- Visual Geometry Group
- Training time on 4 GPUs for 3 weeks !

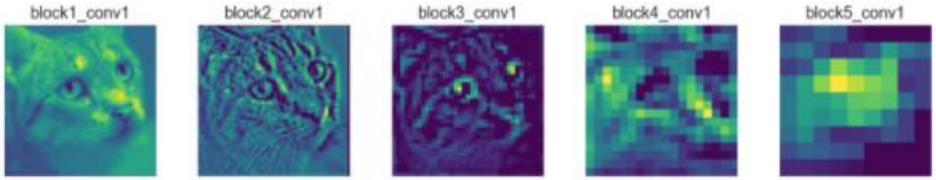




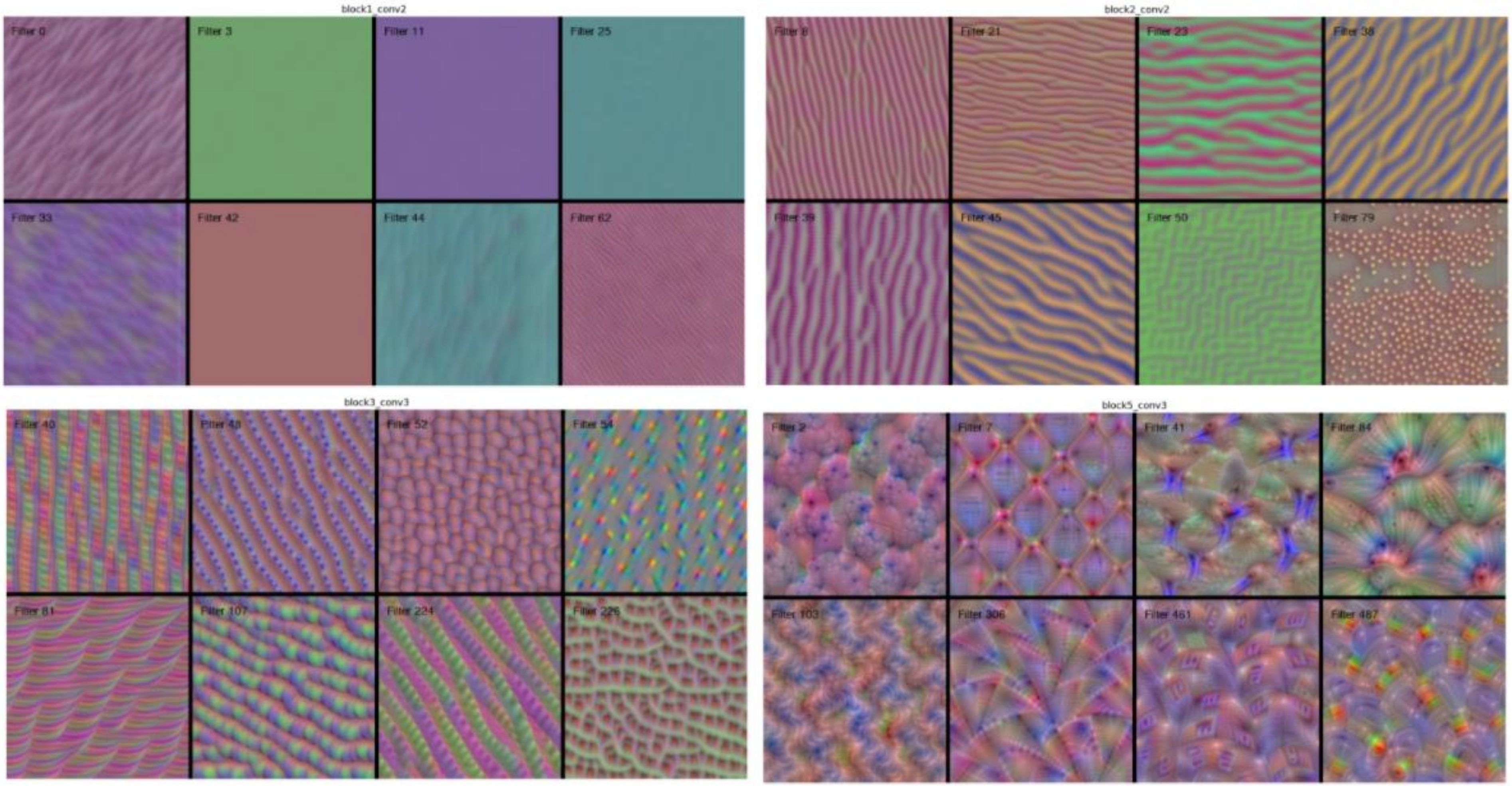
VGGNet: Visualizing Feature Map







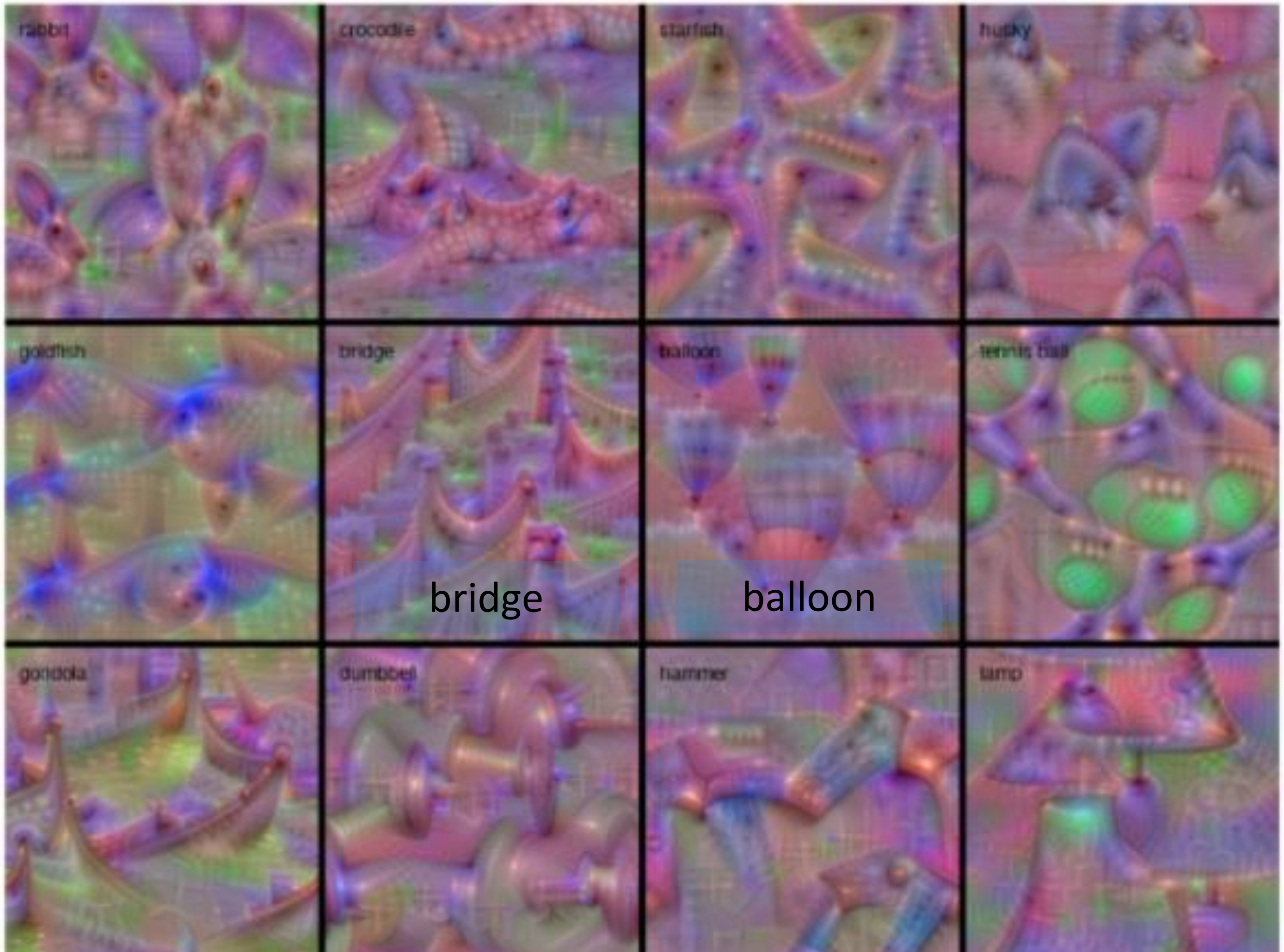
VGGNET: Visualizing Filters



CNN: Visualizing Class Outputs

rabbit crocodile star fish husky

Rabbit



tennis ball

gondola dumbbell hammer lamp

CNN: Visualization