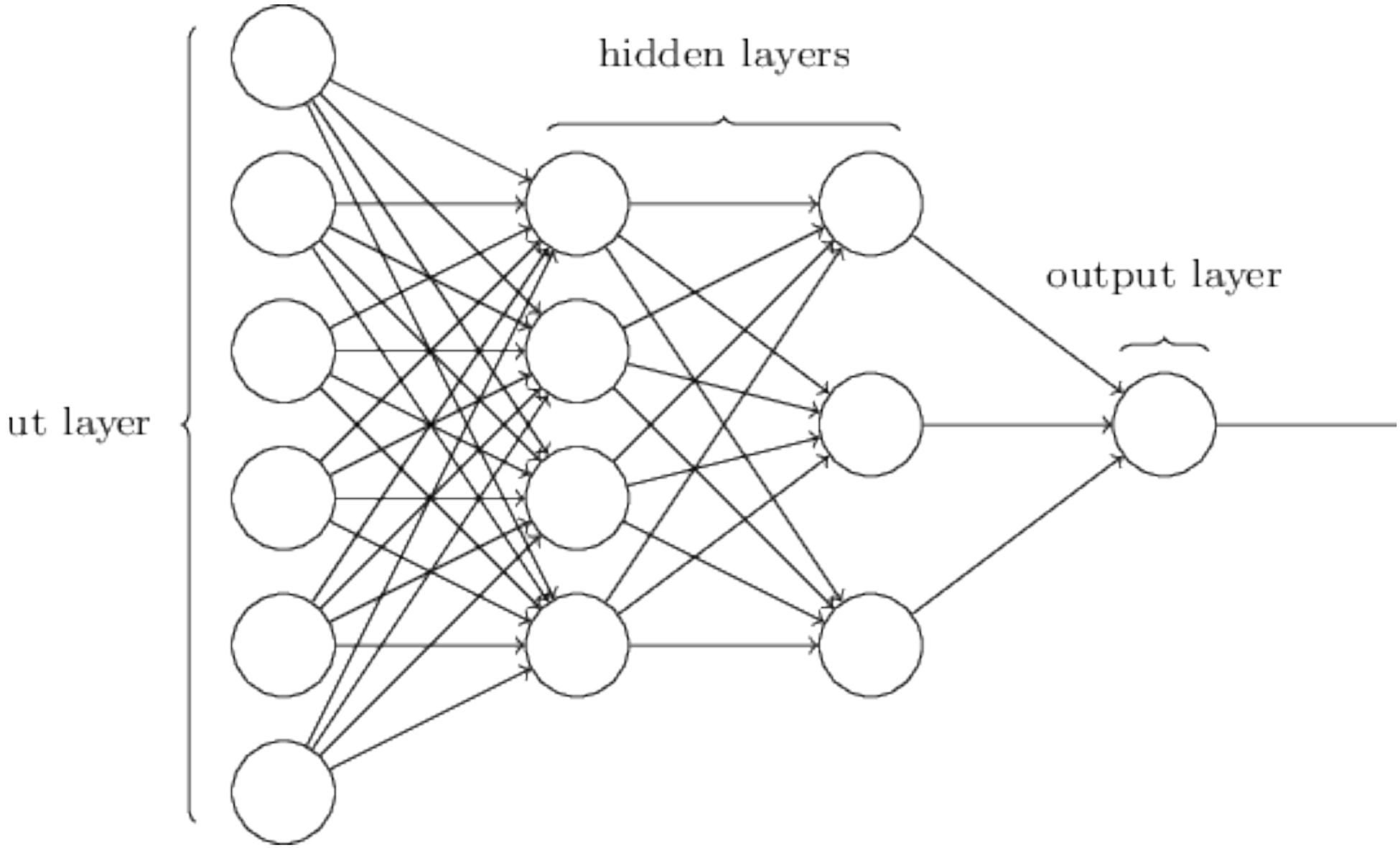
Convolutional Neural Networks

Deep Learning Course
By Felix Kreuk



Recall DNN

(AKA “Fully-Connected”)

How Did We Solve MNIST?

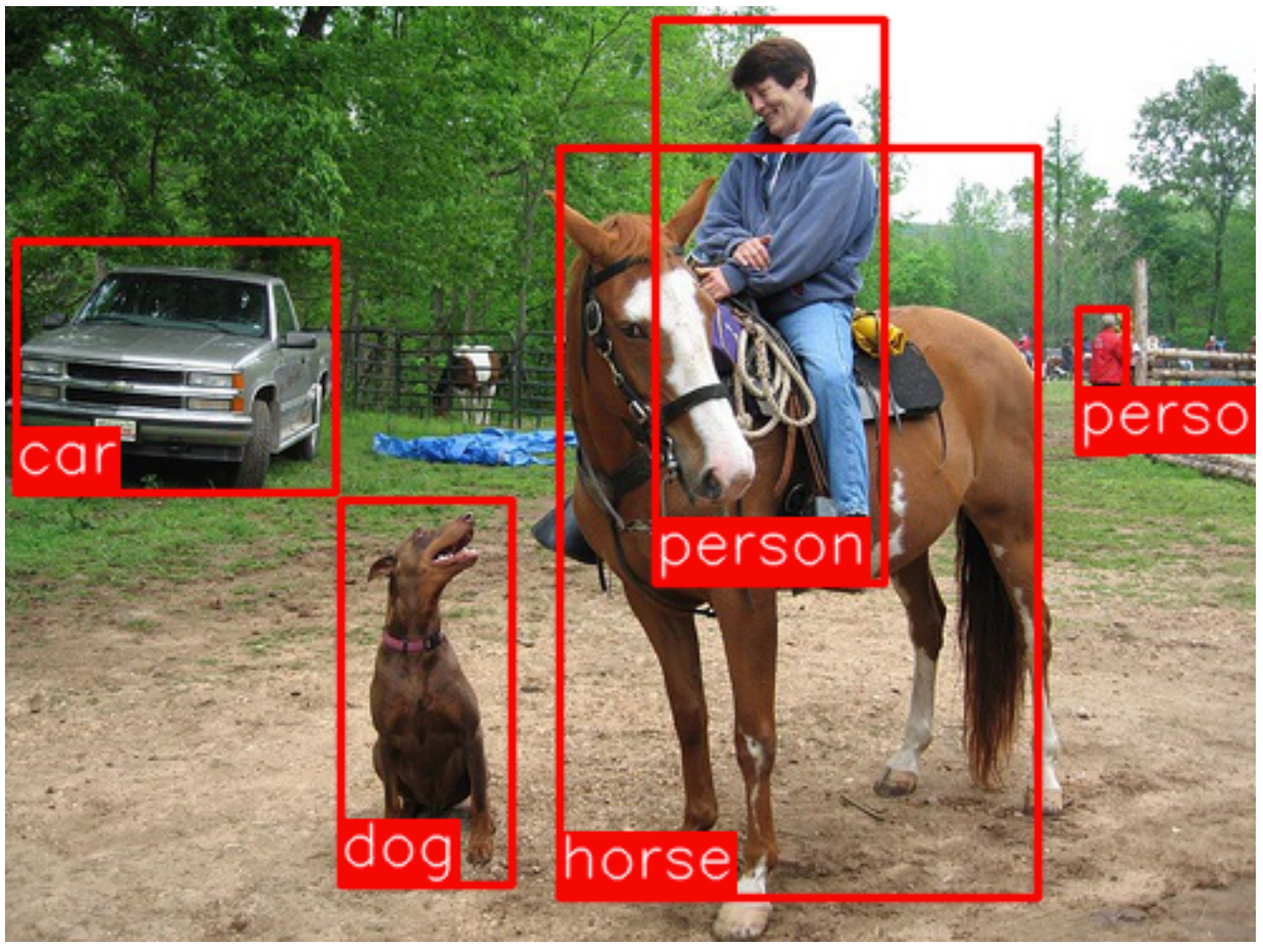
- Transform a 28x28 image to a 1x784 row vector.
- Feed to a DNN.
- Accuracy ~95%.
- Can we do better?

How Did We Solve MNIST?

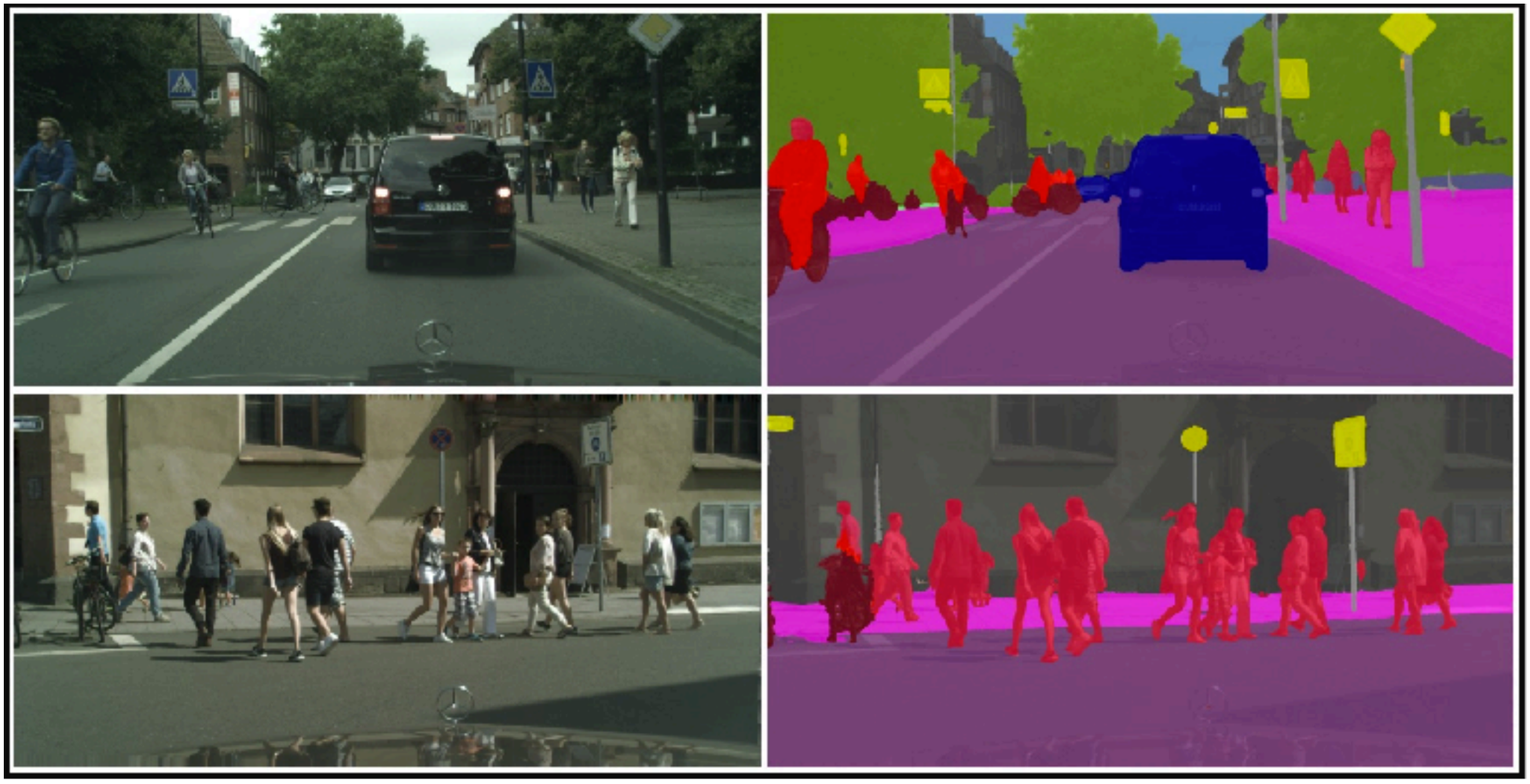
- Transform a 28x28 image to a 1x784 row vector - what about spatial relationship?
- Feed to a DNN - how many parameters? What features did the network learn?
- Accuracy ~95% - is this the best we can achieve?
- Can we do better - yes.
- These are some problems to think about for the next slides.

Introducing CNNs

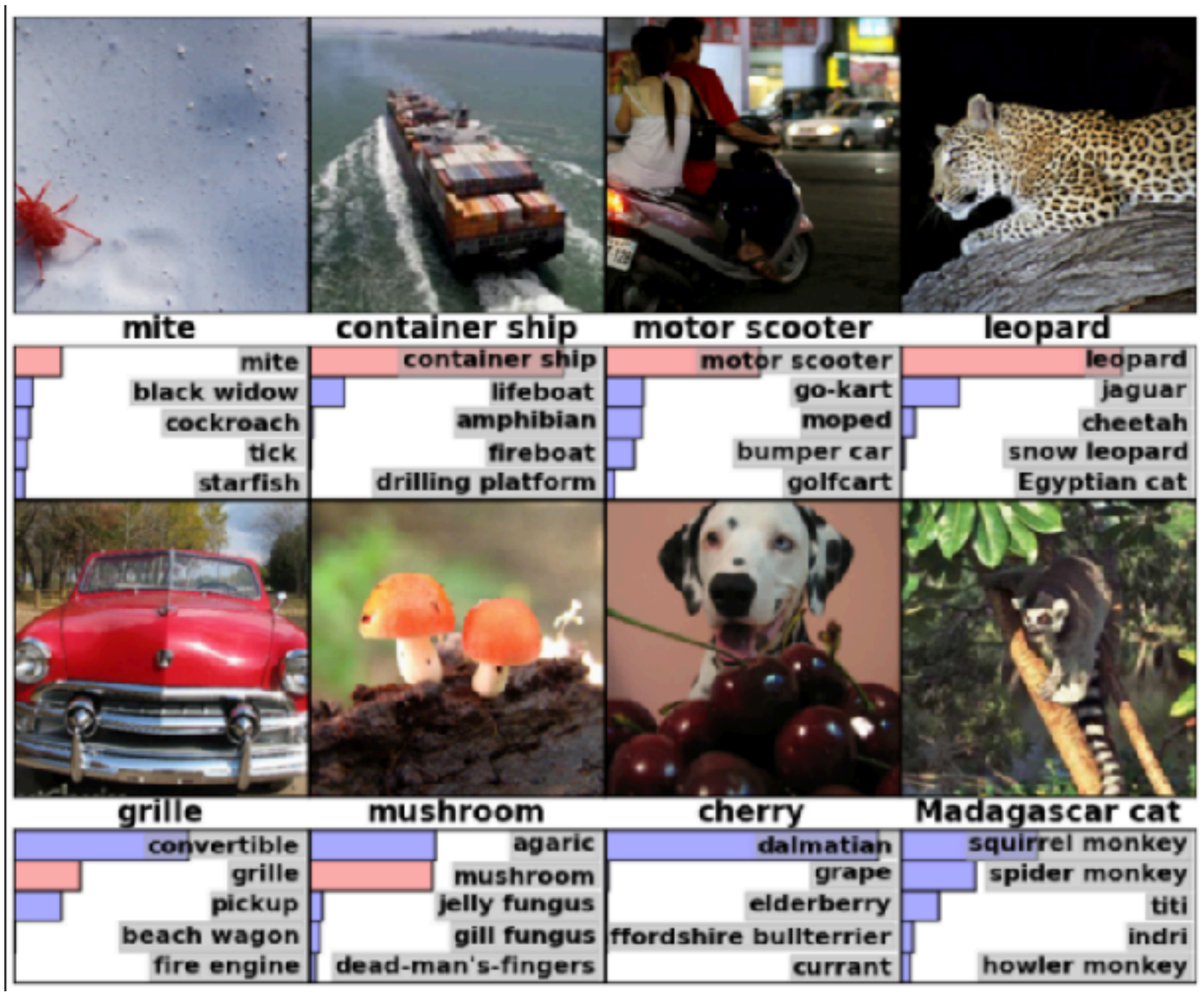
They Are Everywhere



They Are Everywhere



They Are Everywhere



They Are Everywhere

Describes without errors	Describes with minor errors	Somewhat related to the image
		
A person riding a motorcycle on a dirt road.	Two dogs play in the grass.	A skateboarder does a trick on a ramp.
		
A group of young people playing a game of frisbee.	Two hockey players are fighting over the puck.	A little girl in a pink hat is blowing bubbles.

They Are Everywhere

1 **Upload photo**

The first picture defines the scene you would like to have painted.

2 **Choose style**

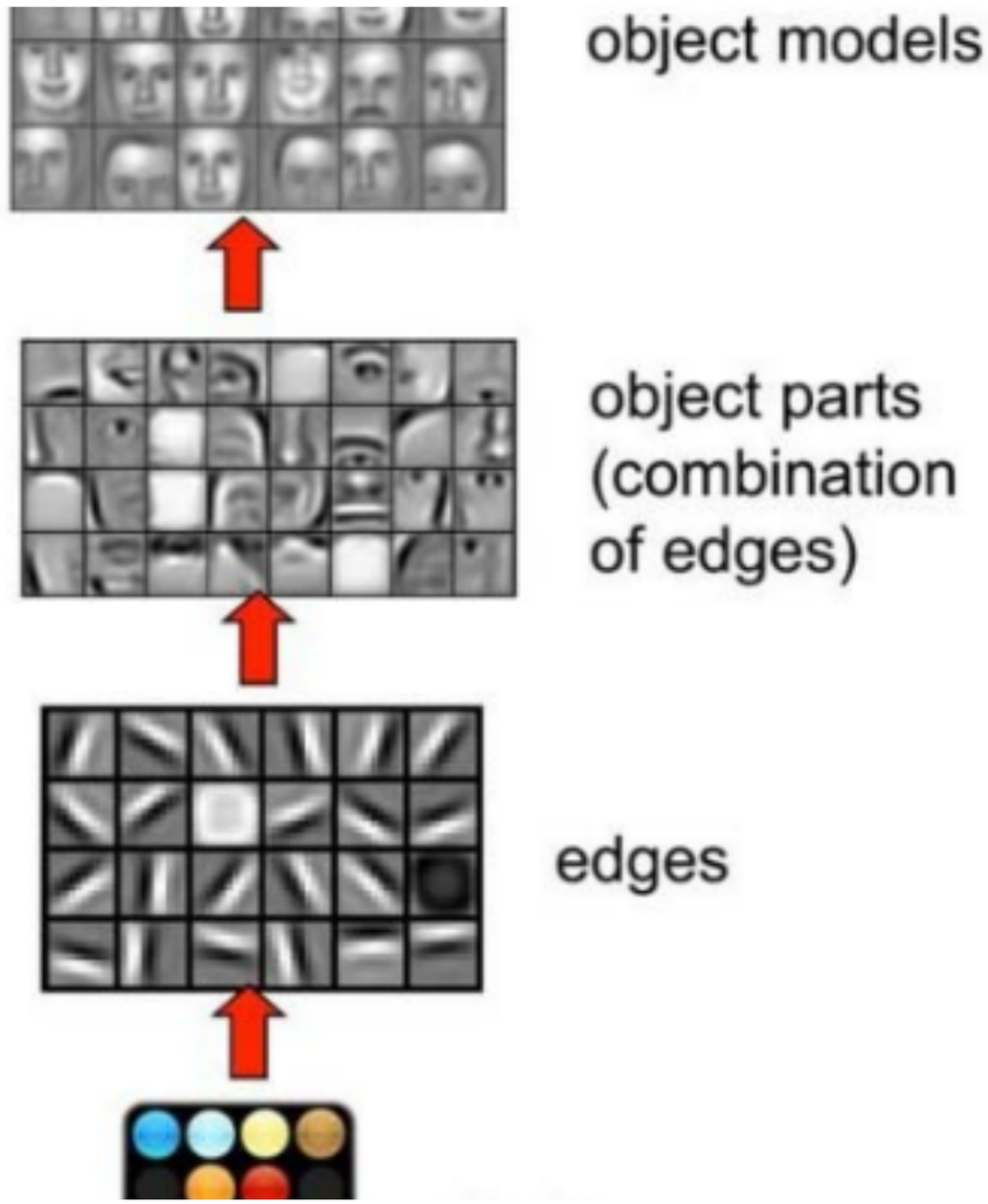
Choose among predefined styles or upload your own style image.

3 **Submit**

Our servers paint the image for you. You get an email when it's done.

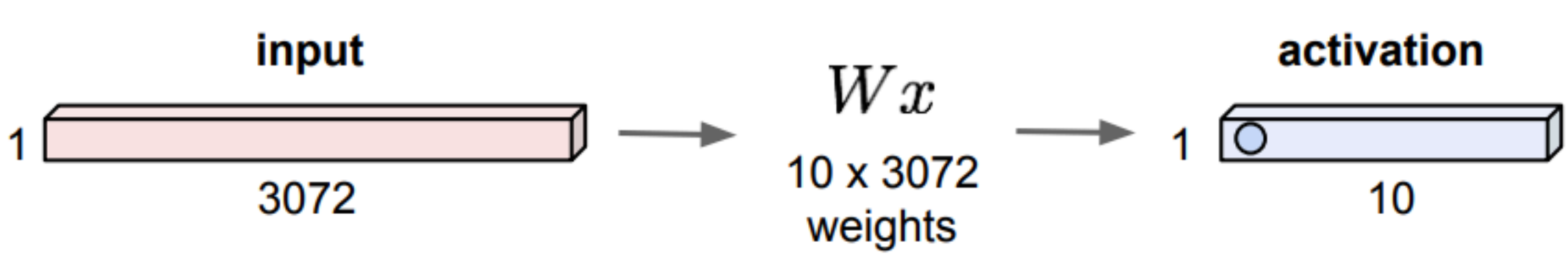
Biological Motivation

- CNNs are biologically inspired variants of DNNs.
- Our visual cortex contains a complex arrangement of cells. These cells are sensitive to sub-regions of the visual field.
- These cells act as local filters of the input space.
- Well suited to exploit strong spatially local correlation in input space.



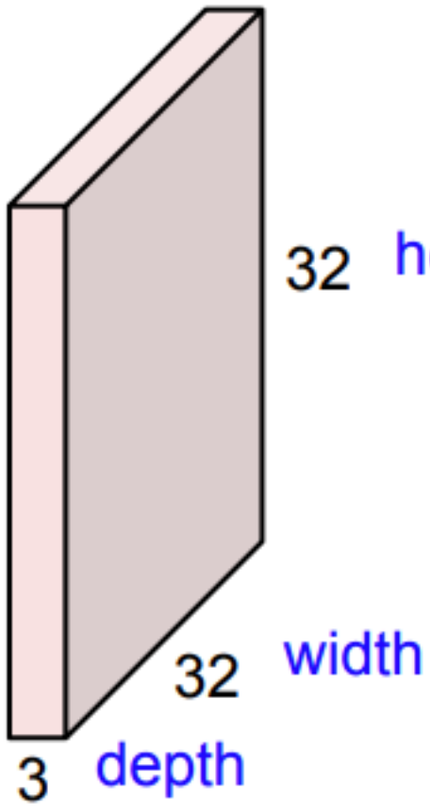
Fully Connected Layer

32x32x3 image -> stretch to 3072 x 1



Convolutional Layer

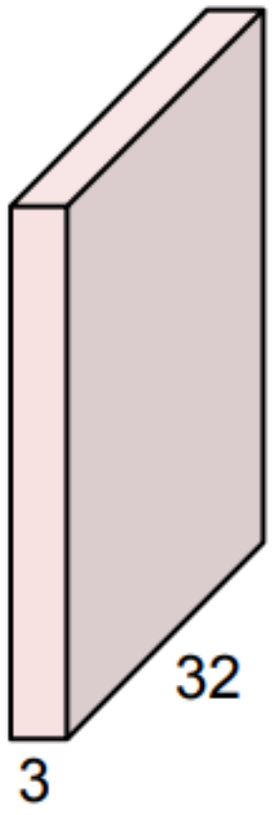
32x32x3 image -> preserve spatial structure



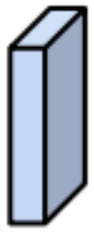
Adapted from cs231 by Fei-Fei Li & Justin Johnson & Serena Yeung

Convolutional Layer

32x32x3 image



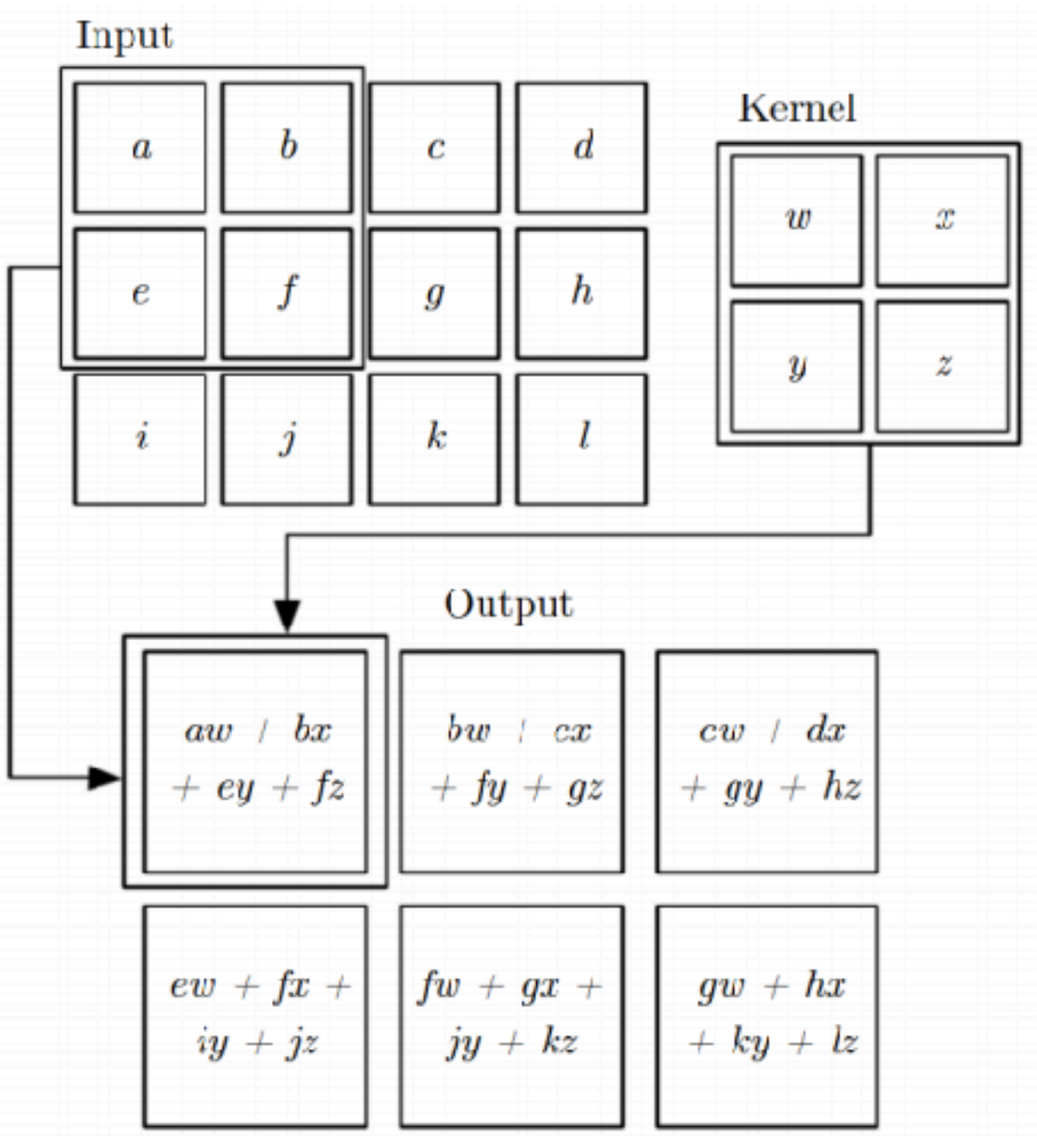
5x5x3 filter



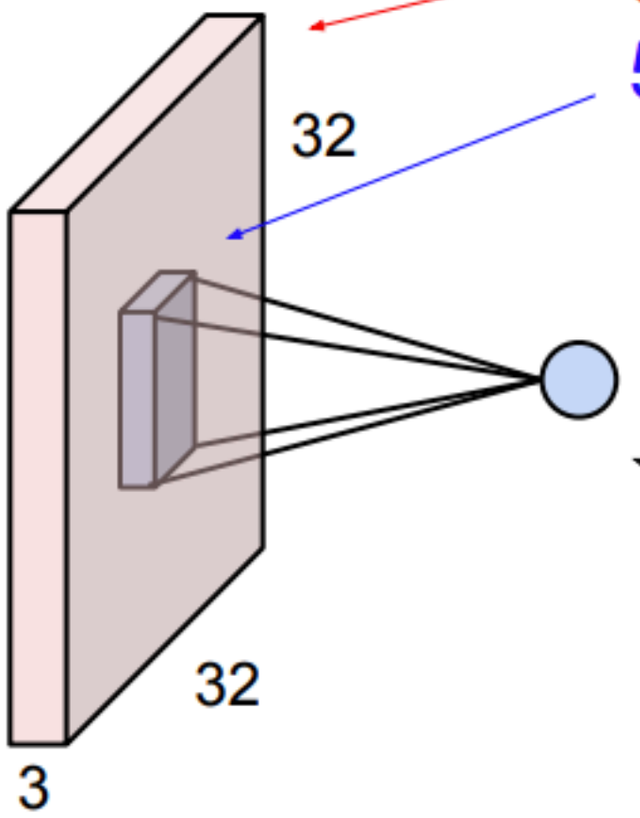
Convolve the filter with the image
i.e. “slide over the image spatially,
computing dot products”

Adapted from cs231 by Fei-Fei Li & Justin Johnson & Serena Yeung

Convolutional Layer



Convolutional Layer



32x32x3 image

5x5x3 filter w

1 number:

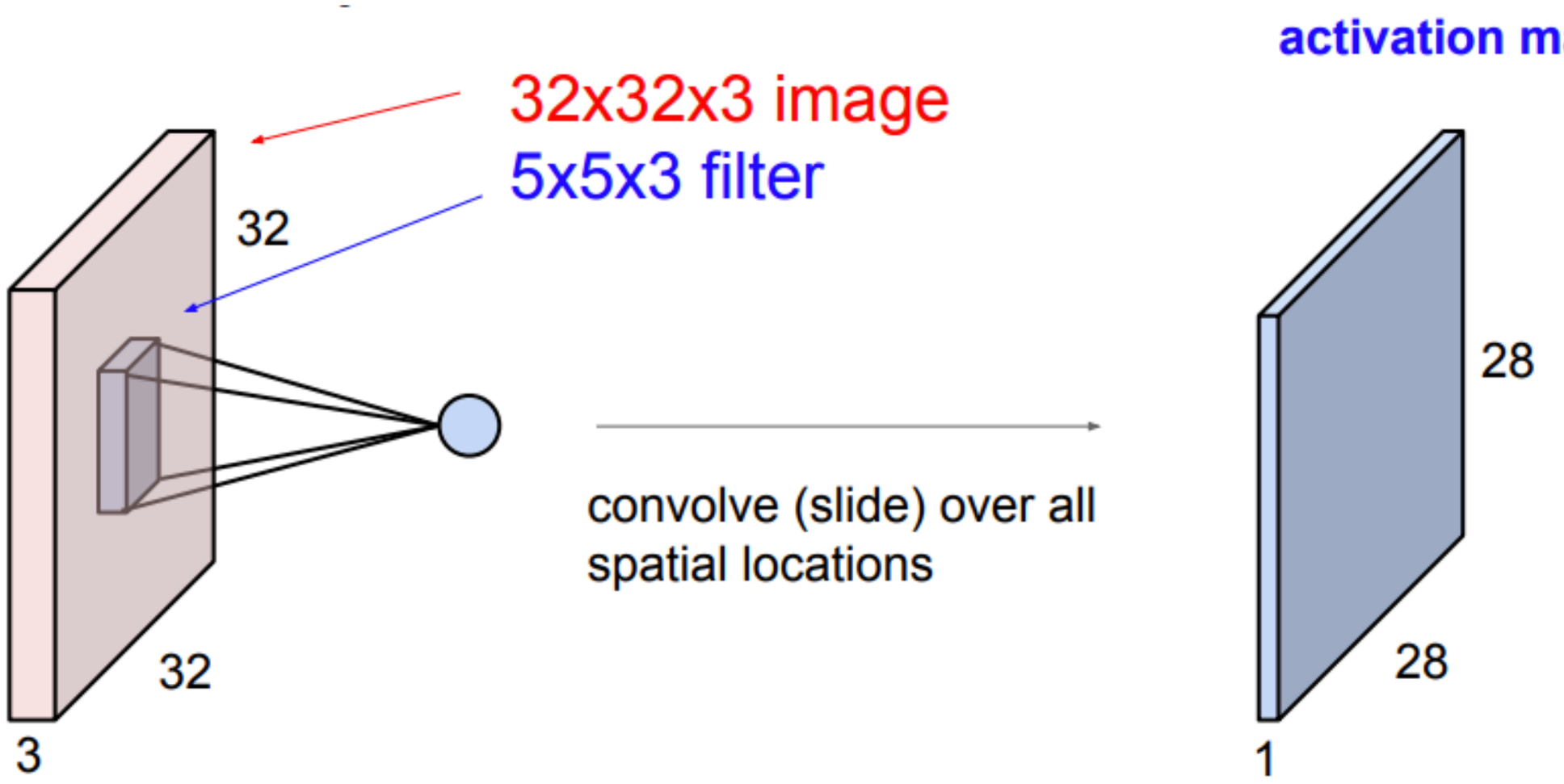
the result of taking a dot product between the filter and a small 5x5x3 chunk of the image

(i.e. 5*5*3 = 75-dimensional dot product + bias)

$$w^T x + b$$

Adapted from cs231 by Fei-Fei Li & Justin Johnson & Serena Yeung

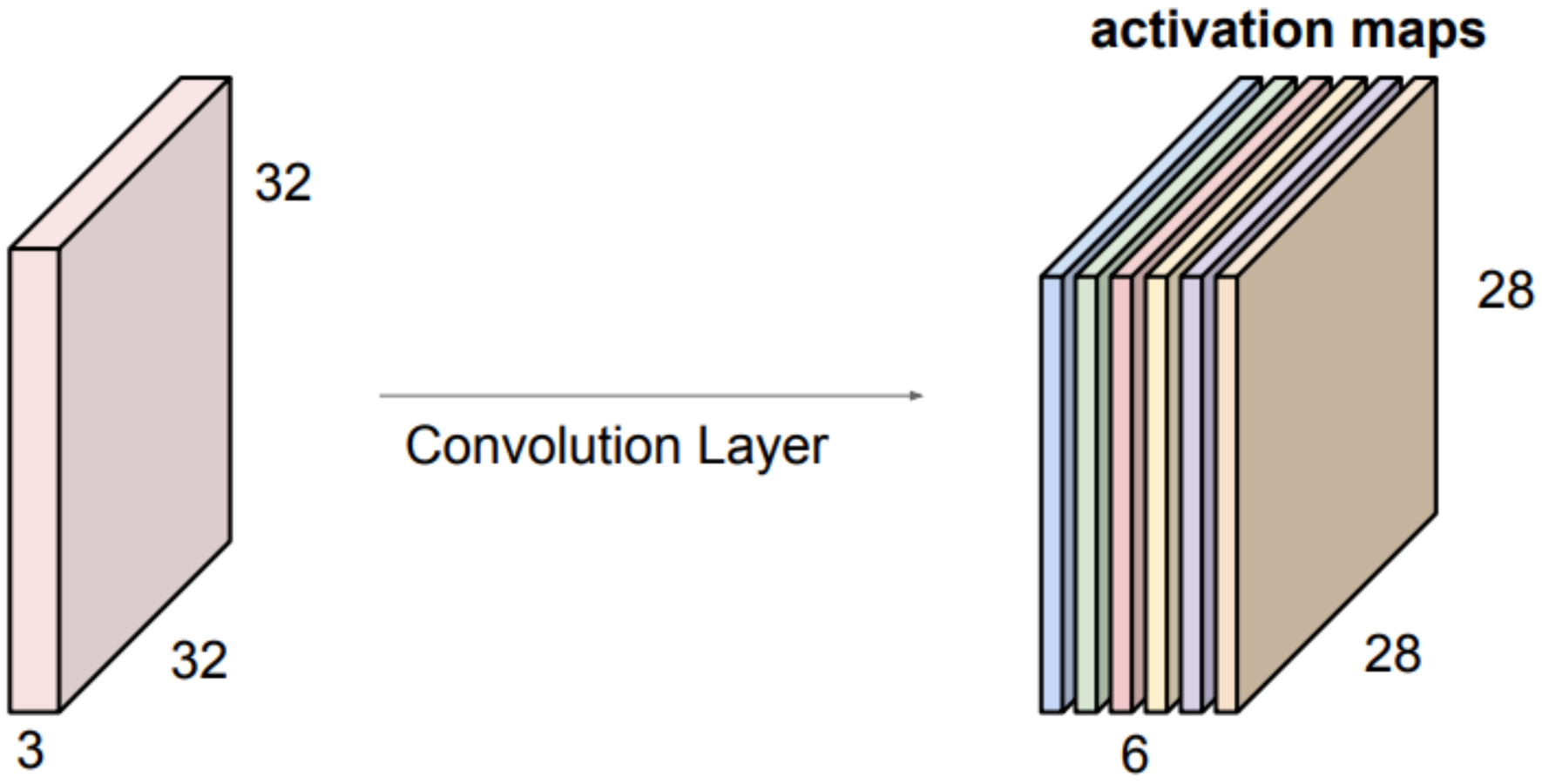
Convolutional Layer



Adapted from cs231 by Fei-Fei Li & Justin Johnson & Serena Yeung

Convolutional Layer

For example, if we had 6 5x5 filters, we'll get 6 separate activation maps:

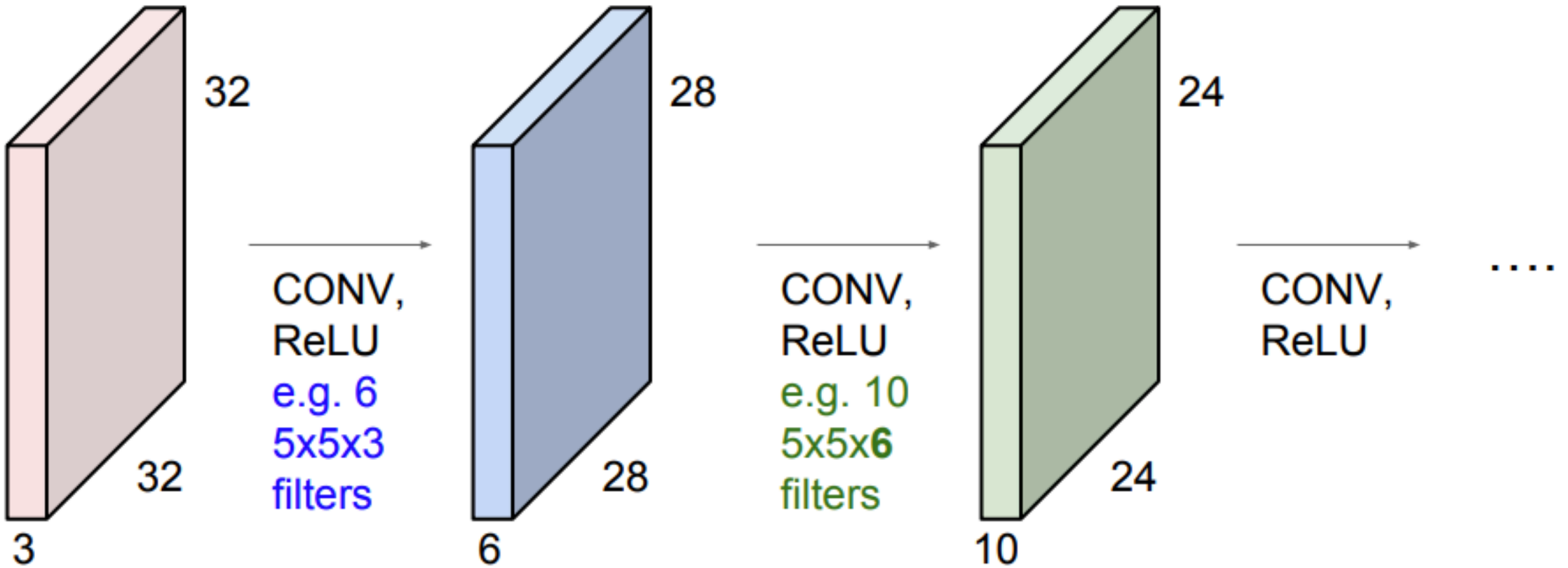


We stack these up to get a "new image" of size 28x28x6!

Adapted from cs231 by Fei-Fei Li & Justin Johnson & Serena Yeung

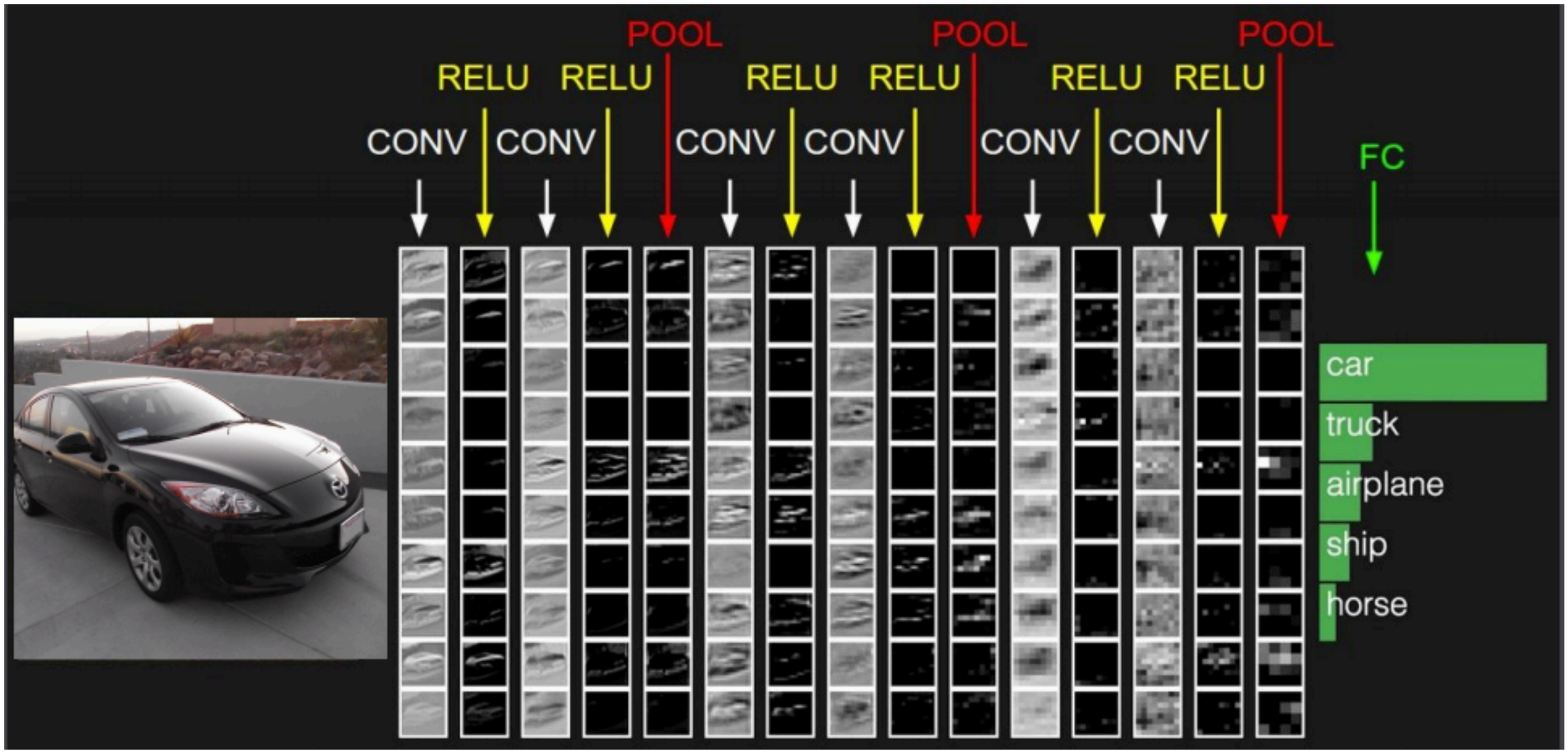
Convolutional Layer

Preview: ConvNet is a sequence of Convolutional Layers, interspersed with activation functions



Adapted from cs231 by Fei-Fei Li & Justin Johnson & Serena Yeung

Convolutional Layer



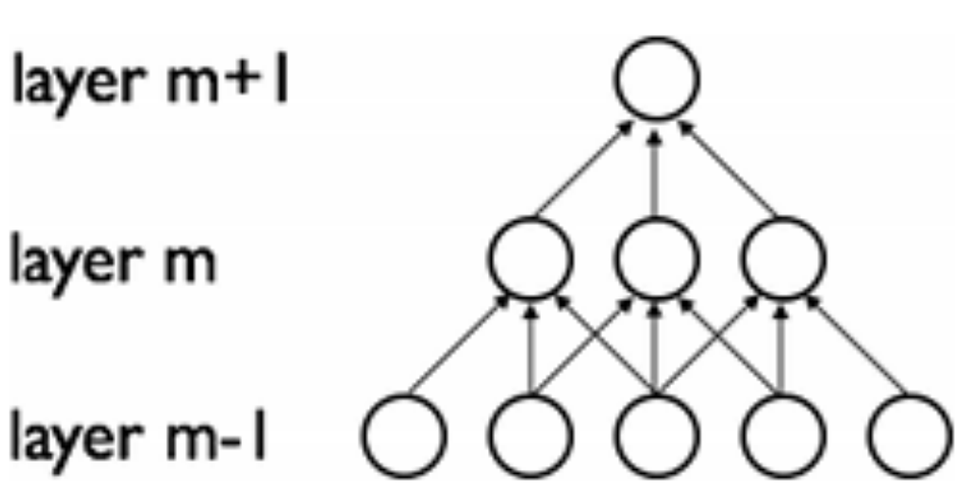
Adapted from cs231 by Fei-Fei Li & Justin Johnson & Serena Yeung

But Why?

Sparse Connectivity

- The inputs of hidden units in layer **m** are from a subset of units in layer **m-1**.

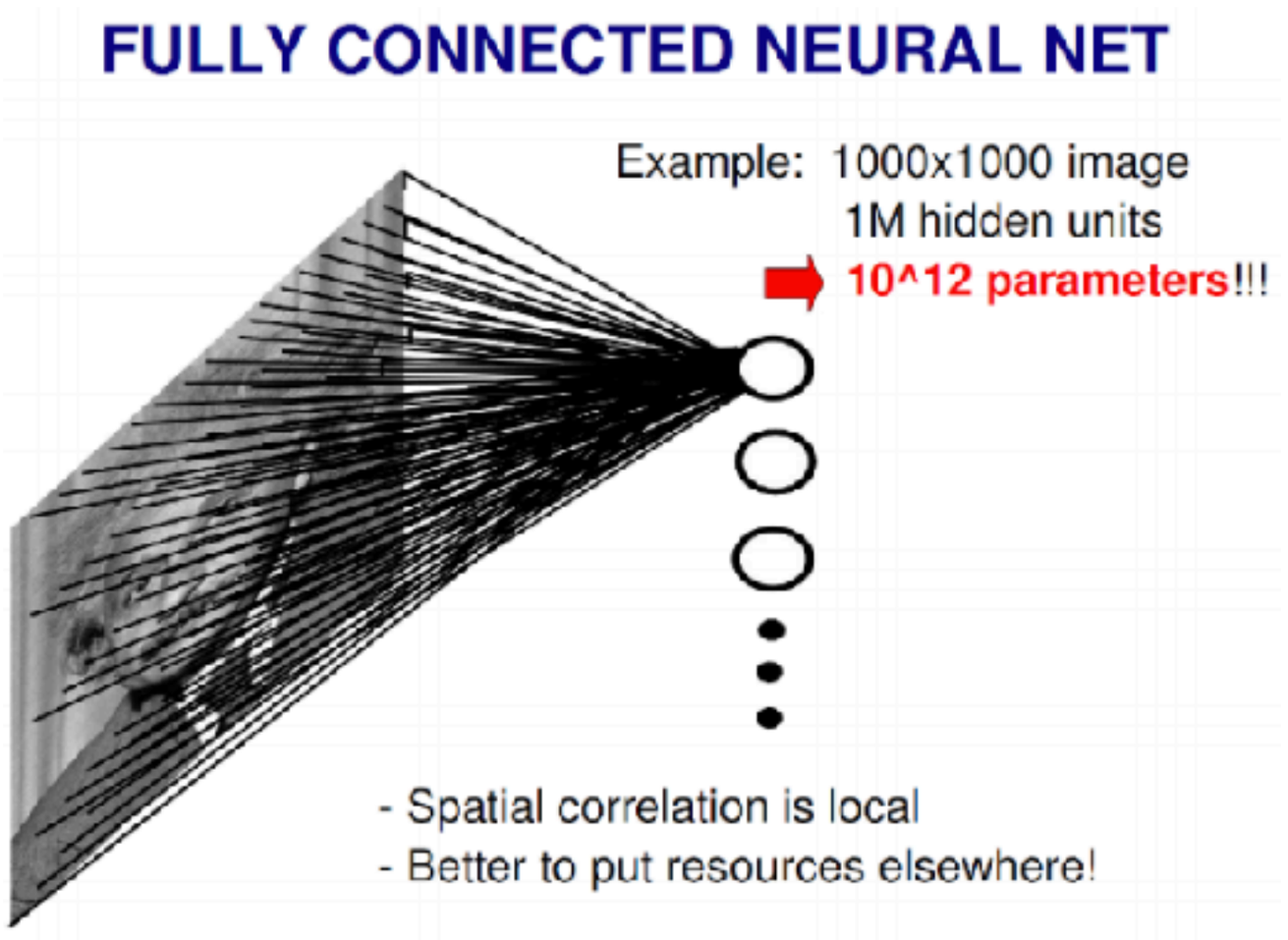
- The architecture thus ensures that the learnt “filters” produce the strongest response to a spatially local input pattern.



Shared Weights

- Replicating units in this way allows for features to be detected **regardless of their position in the visual field.**

- Drastic reduce in the number of free parameters compared to fully connected network **reducing overfitting** and more importantly, computational complexity of the network.

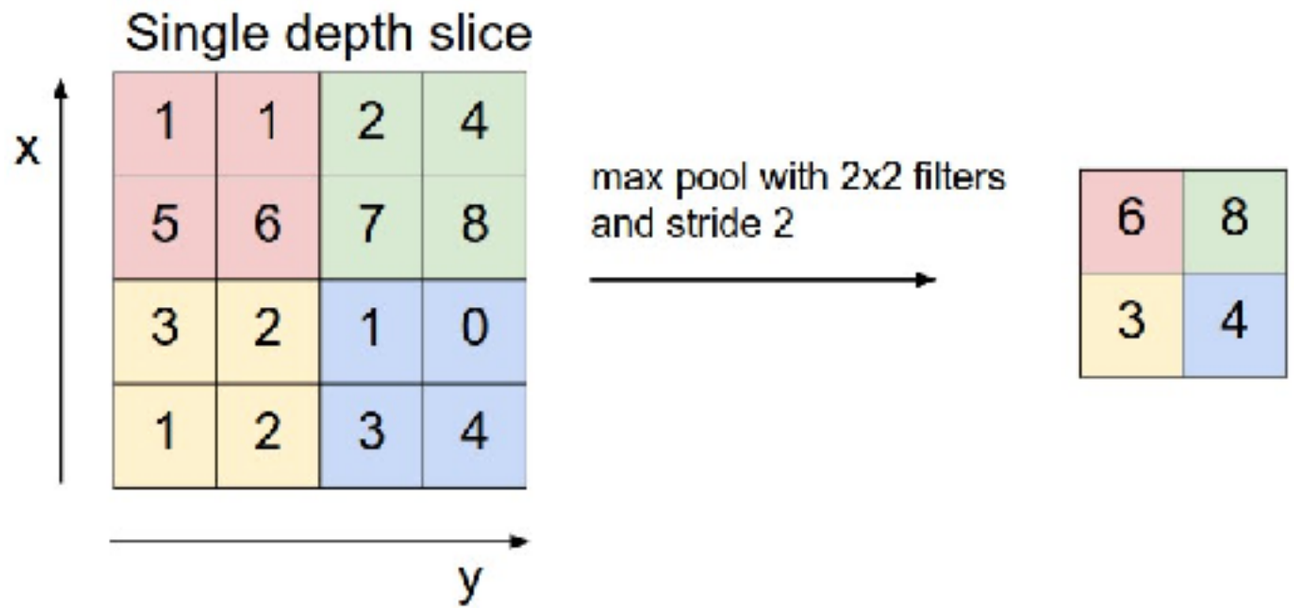


Pooling

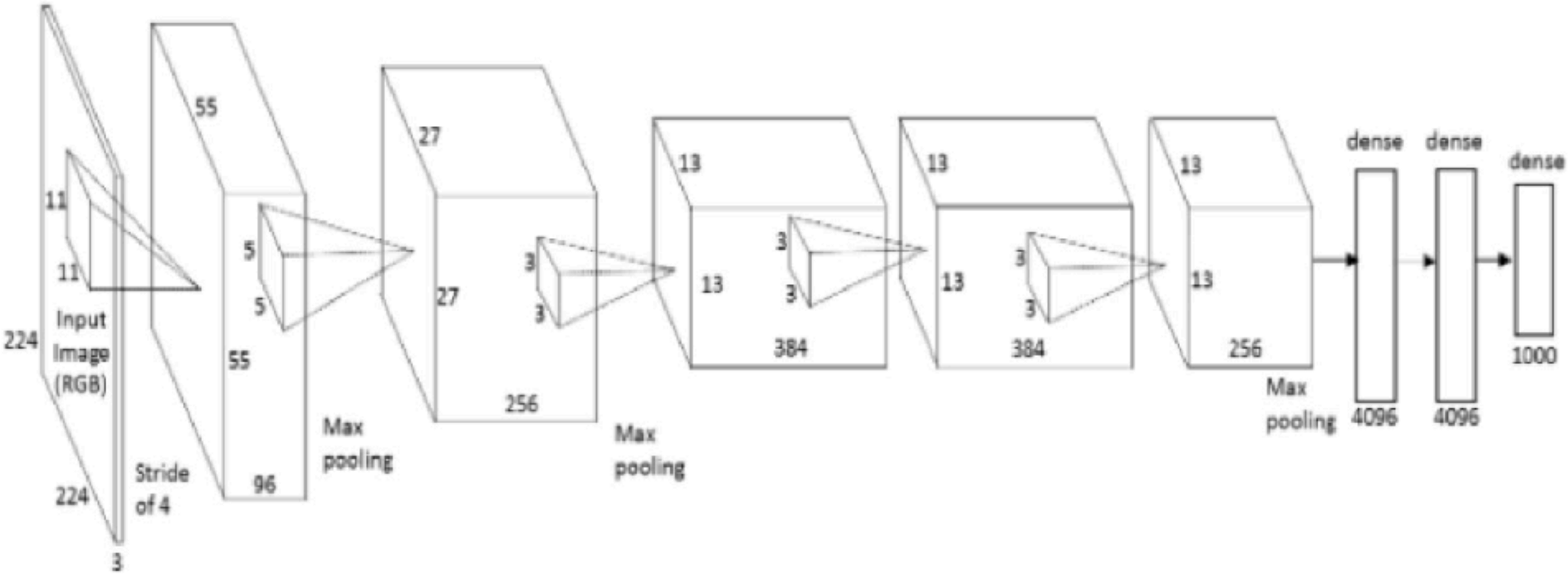
- Non-linear down-sampling to simplify the information in output from convolutional layer.

- Variants: Max pooling (popular), Weighted average, etc.

- Useful property, if we care more about whether some feature is present than exactly where it is, thus adds robustness to position



Real CNNs


224
Input
Image
(RGB)
3
11
11
Stride
of 4
55
55
96
5
5
Max
pooling
27
27
256
3
3
Max
pooling
13
13
384
3
3
13
13
384
3
3
13
13
256
Max
pooling
dense
4096
dense
4096
dense
1000

Real CNNs