Convolutional Neural Networks

Manish Gupta

Visiting Faculty at ISB
Senior Applied Scientist at Microsoft, India
Adjunct Faculty at IIIT-H

Credits: Stanford CS224D course, CS231N Stanford course, Geoffrey Hinton's course, http://deeplearning.net/tutorial

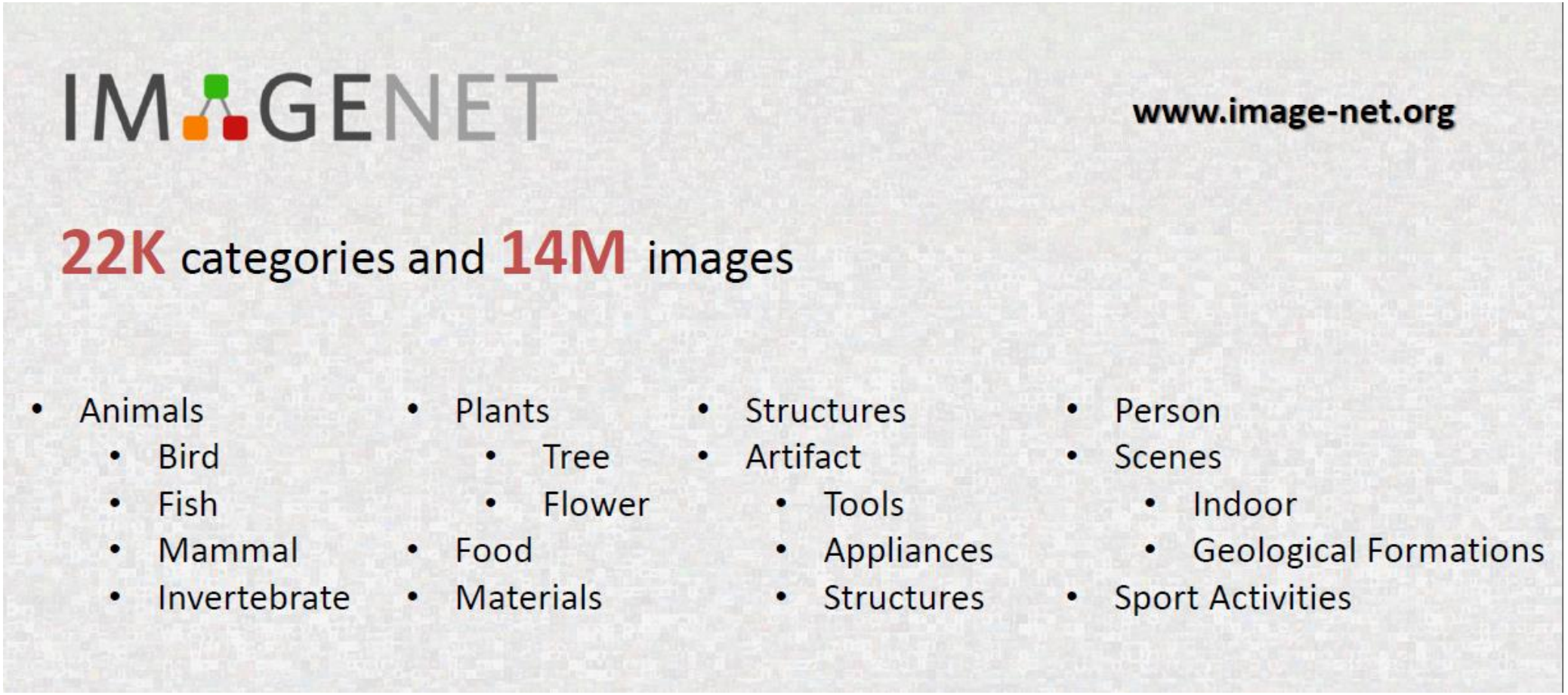
Today’s Agenda

- ImageNet and visual recognition problems
- Introduction to CNNs and applications
- Technical details of a CNN

Today's Agenda

- **ImageNet and visual recognition problems**
- Introduction to CNNs and applications
- Technical details of a CNN

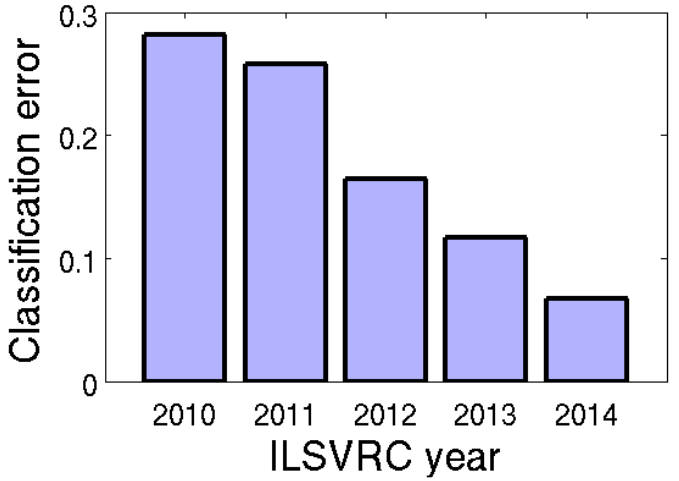
ImageNet



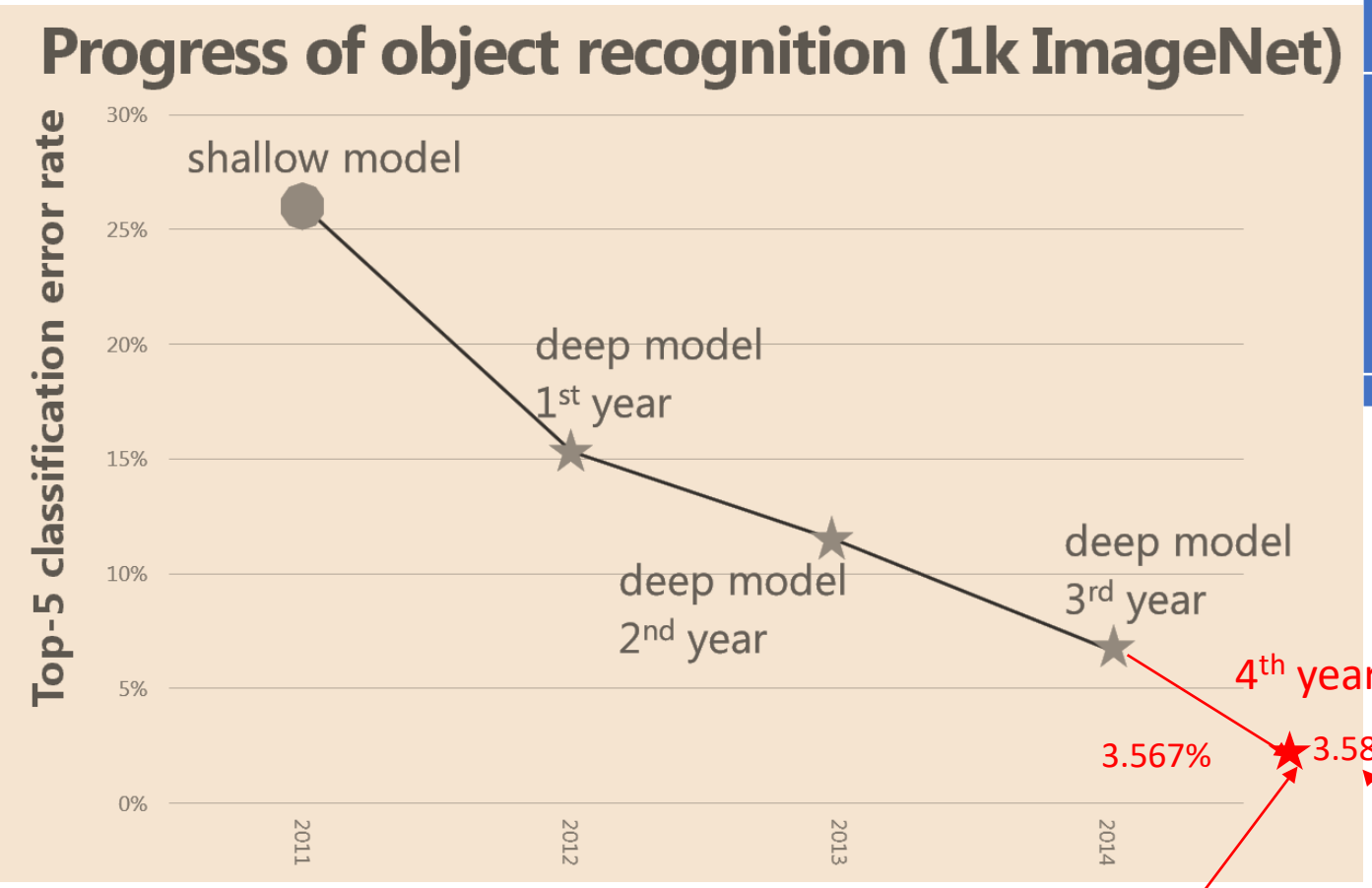
http://image-net.org/download-faq

ImageNet Classification Challenge

- 1,000 object classes, 1,431,167 images



Year	2010	2012	2014	2015	2016	2017
Model, Institution	Linear SVM, NEC-UIUC	AlexNet, SuperVision	Visual Geometry Group (VGG) Oxford, Googlenet	Resnet, MSRA	Trimps-Soushen, The Third Research Institute of the Ministry of Public Security, P.R. China.	Squeeze-and-Excitation Networks, NUS-Qihoo_DPNs (CLS-LOC)
#layers	Not a neural network	8 layers	22 layers	152 layers	Ensemble of Inception-v3 (48 layers), Inception-v4 (~114 layers), Residual Network (152 layers), Inception-ResNet-v2 (200+ layers), Wide Residual Network (~16 layers)	Integrated SE blocks to stacked ResNet-152
Accuracy	28%	16%	7%	3.567%	2.99%	2.25%



Announced
Dec.2015

2012 - 2015

Super-deep: 152 layers

Microsoft Research

Google

ImageNet Challenge Winning Architectures

Year 2010

NEC-UIUC

Dense grid descriptor: HOG, LBP

Coding: local coordinate, super-vector

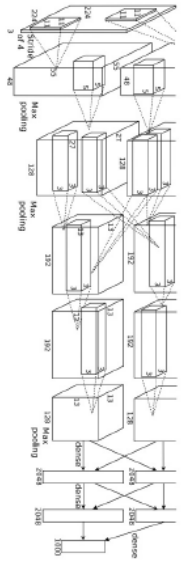
Pooling, SPM

Linear SVM

[Lin CVPR 2011]

Year 2012

SuperVision



[Krizhevsky NIPS 2012]

7 layers

Year 2014

GoogLeNet

[Szegedy arxiv 2014]

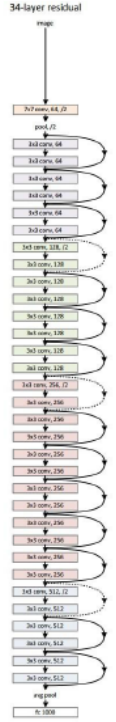
27 layers

VGG

[Simonyan arxiv 2014]

Year 2015

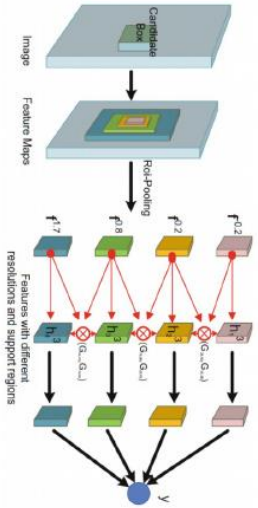
MSRA



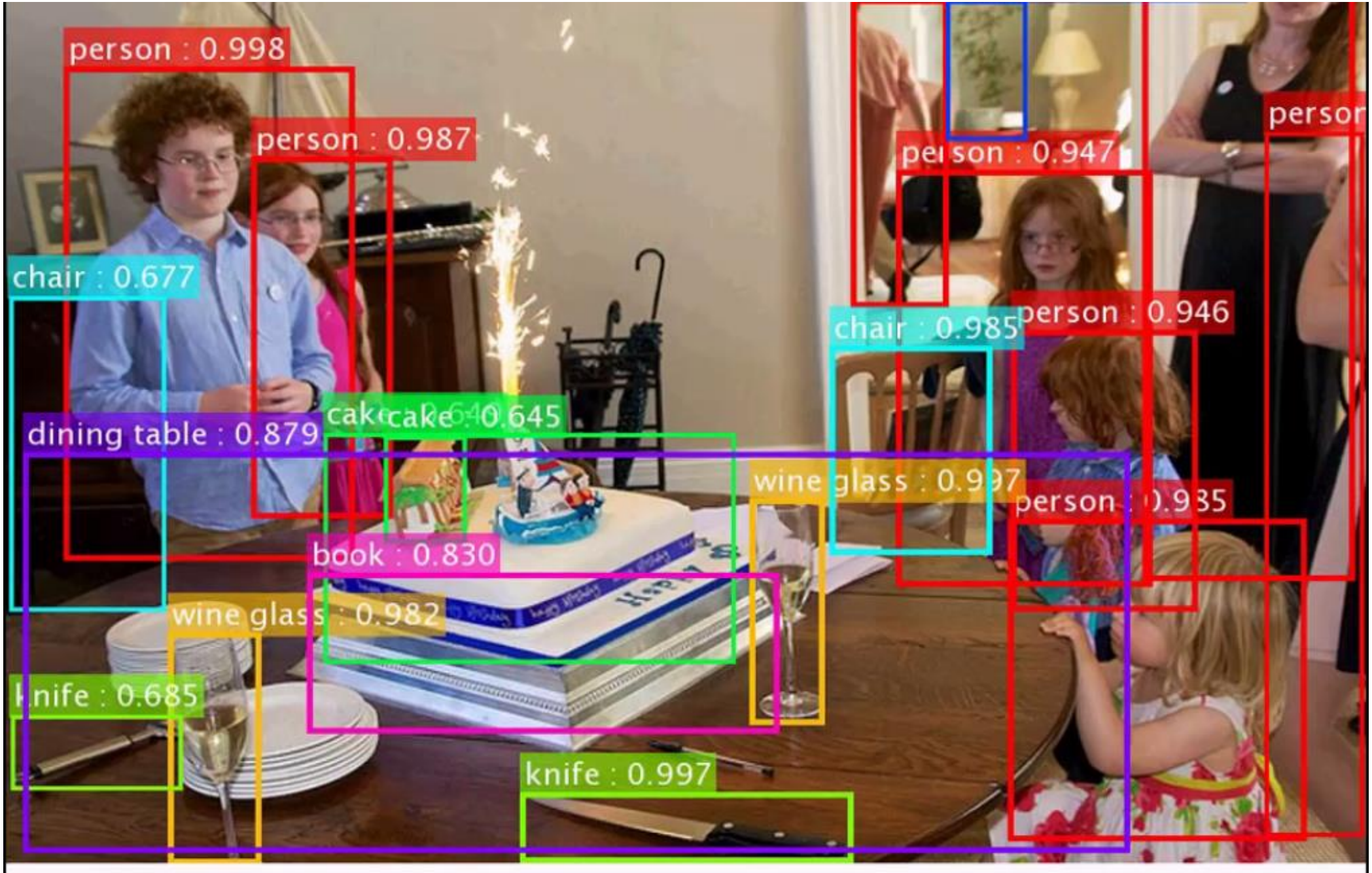
151 layers

Year 2016

CUImage, Chinese University of Hong Kong

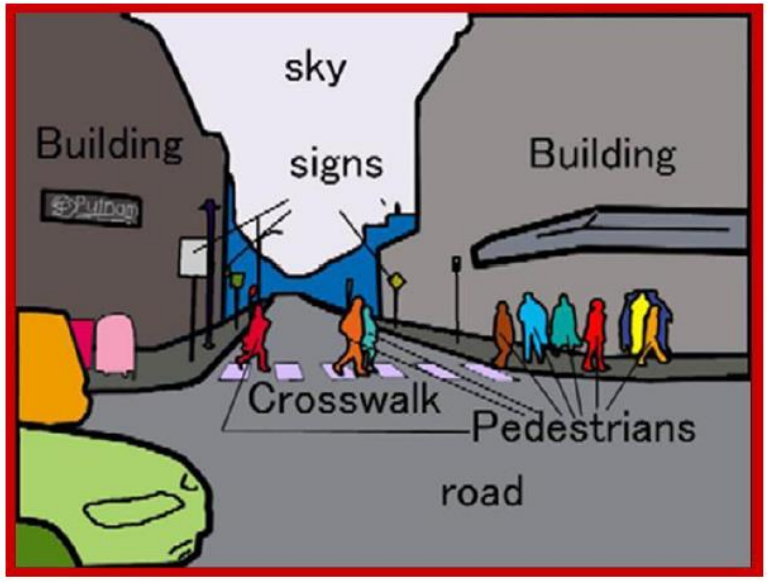


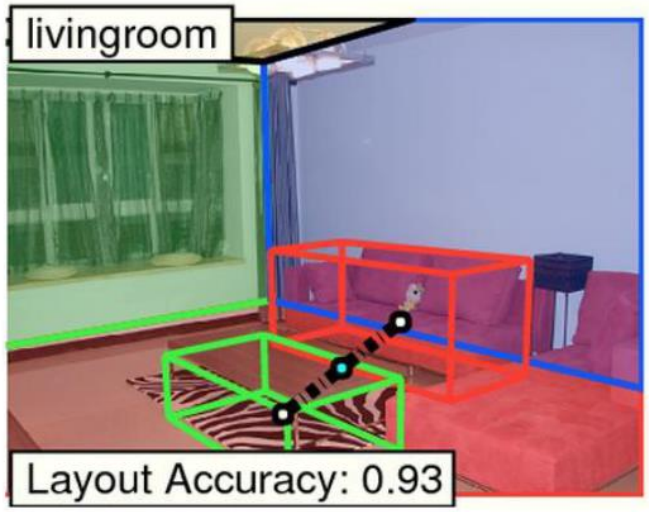
Gated Bi-directional CNN: 269 layers

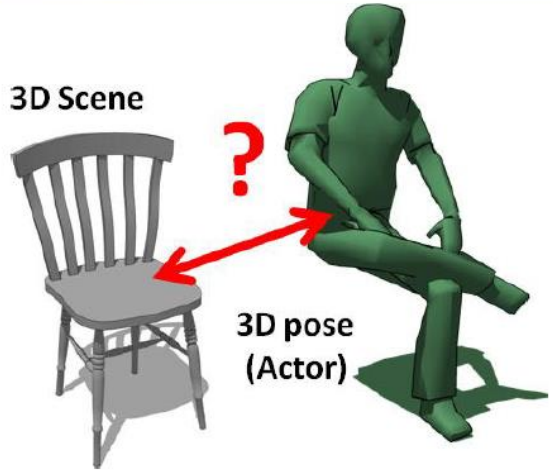


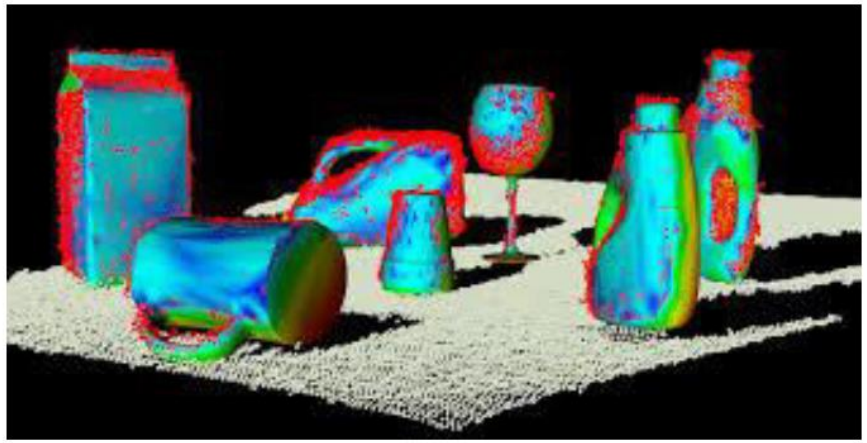
(slide credit: Jian Sun, MSR)

More difficult vision problems

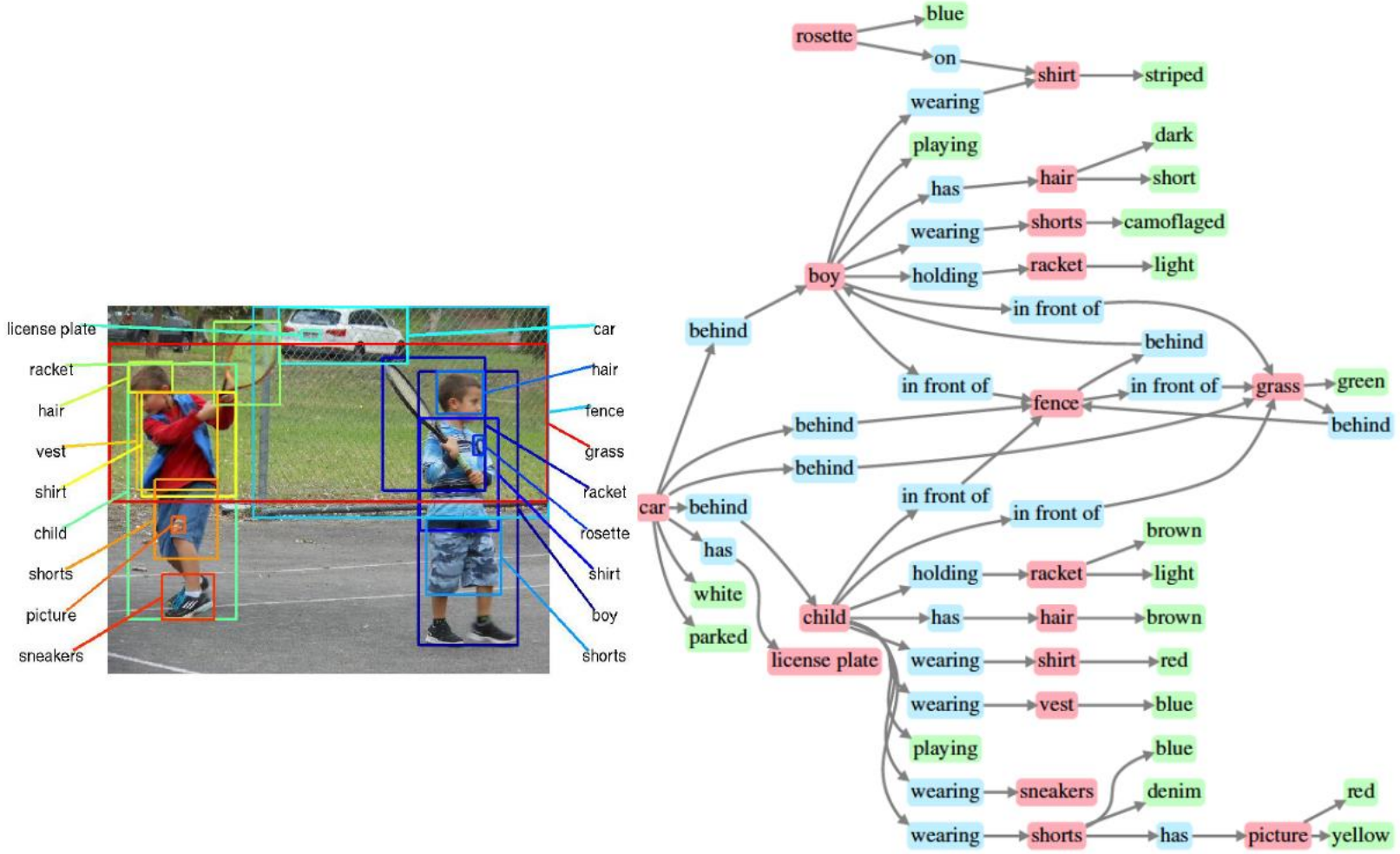








More difficult vision problems



More difficult vision problems

http://karpathy.github.io/2012/10/22/state-of-computer-vision/

- You recognize it is an image of a bunch of people and you understand they are in a hallway
- You recognize that there are 3 mirrors in the scene so some of those people are “fake” replicas from different viewpoints.
- You recognize Obama from the few pixels that make up his face. It helps that he is in his suit and that he is surrounded by other people with suits.
- You recognize that there’s a person standing on a scale, even though the scale occupies only very few white pixels that blend with the background. But, you’ve used the person’s pose and knowledge of how people interact with objects to figure it out.
- You recognize that Obama has his foot positioned just slightly on top of the scale. Notice the language I’m using: It is in terms of the 3D structure of the scene, not the position of the leg in the 2D coordinate system of the image.
- You know how physics works: Obama is leaning in on the scale, which applies a force on it. Scale measures force that is applied on it, that’s how it works => it will over-estimate the weight of the person standing on it.
- The person measuring his weight is not aware of Obama doing this. You derive this because you know his pose, you understand that the field of view of a person is finite, and you understand that he is not very likely to sense the slight push of Obama’s foot.
- You understand that people are self-conscious about their weight. You also understand that he is reading off the scale measurement, and that shortly the over-estimated weight will confuse him because it will probably be much higher than what he expects. In other words, you reason about implications of the events that are about to unfold seconds after this photo was taken, and especially about the thoughts and how they will develop inside people’s heads. You also reason about what pieces of information are available to people.
- There are people in the back who find the person’s imminent confusion funny. In other words you are reasoning about state of mind of people, and their view of the state of mind of another person. That’s getting frighteningly meta.
- Finally, the fact that the perpetrator here is the president makes it maybe even a little more funnier. You understand what actions are more or less likely to be undertaken by different people based on their status and identity.

Today's Agenda

- ImageNet and visual recognition problems
- **Introduction to CNNs and applications**
- Technical details of a CNN

Hierarchical Approach

VISION

pixels → edge → texton → motif → part → object

SPEECH

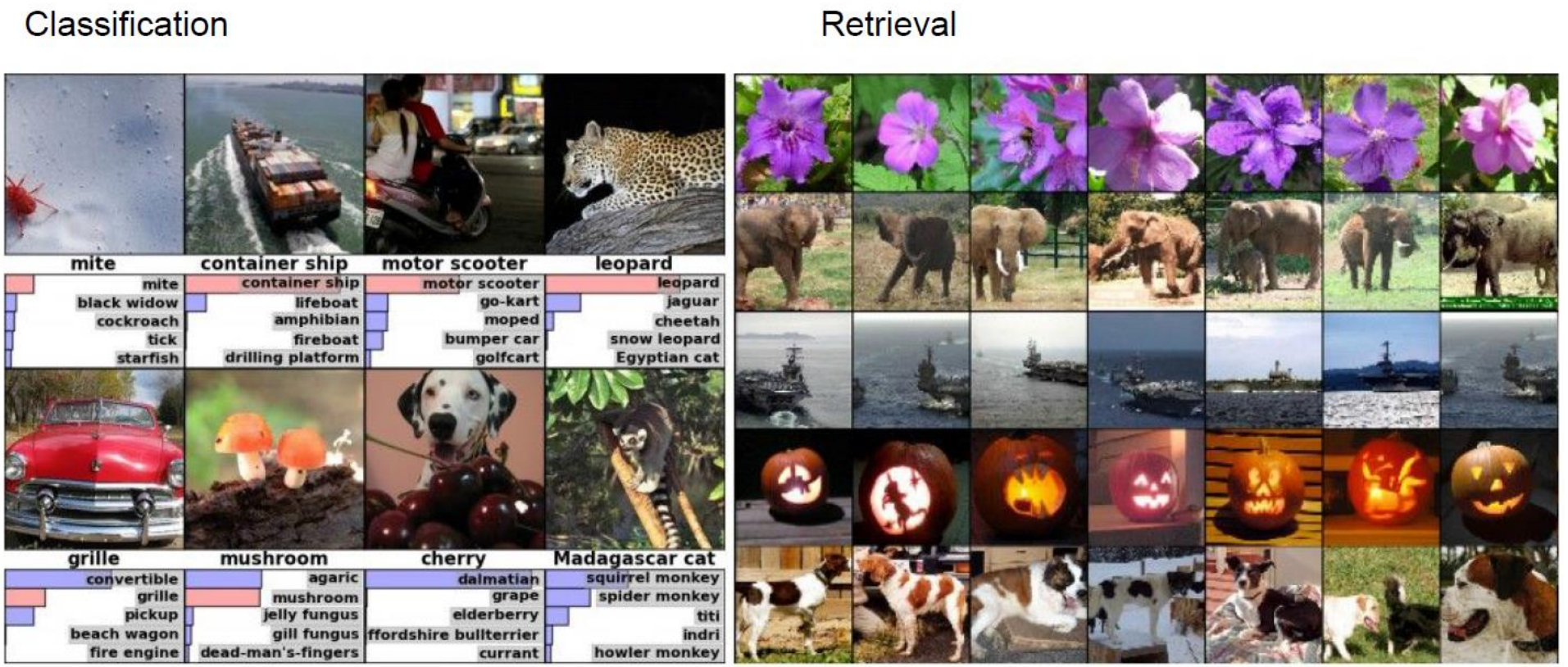
sample → spectral band → formant → motif → phone → word

NLP

character → word → NP/VP/.. → clause → sentence → story

From: Ranzanto

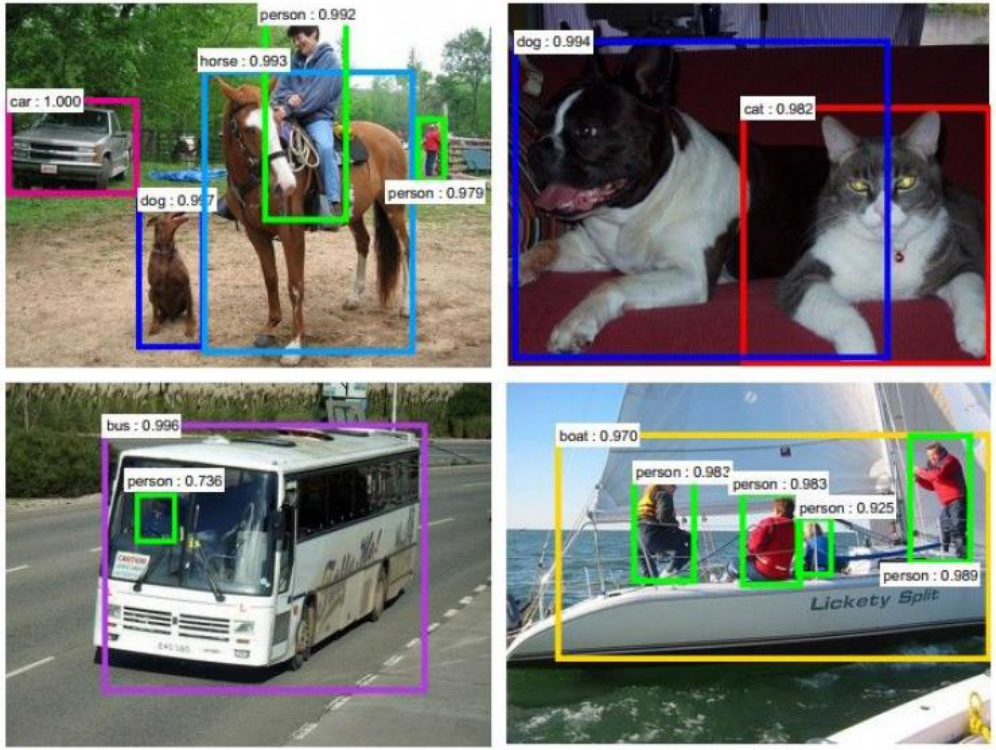
Convolution Networks are everywhere now



[Krizhevsky 2012]

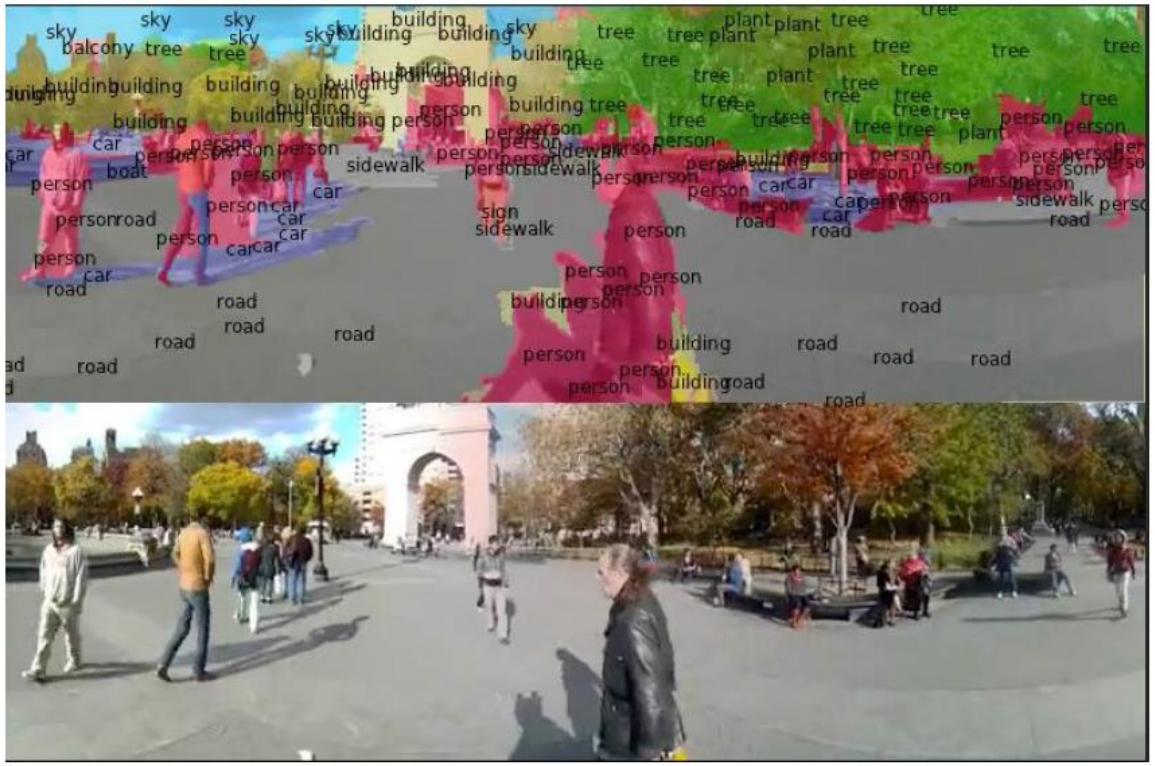
Convolution Networks are everywhere now

Detection



[Faster R-CNN: Ren, He, Girshick, Sun 2015]

Segmentation

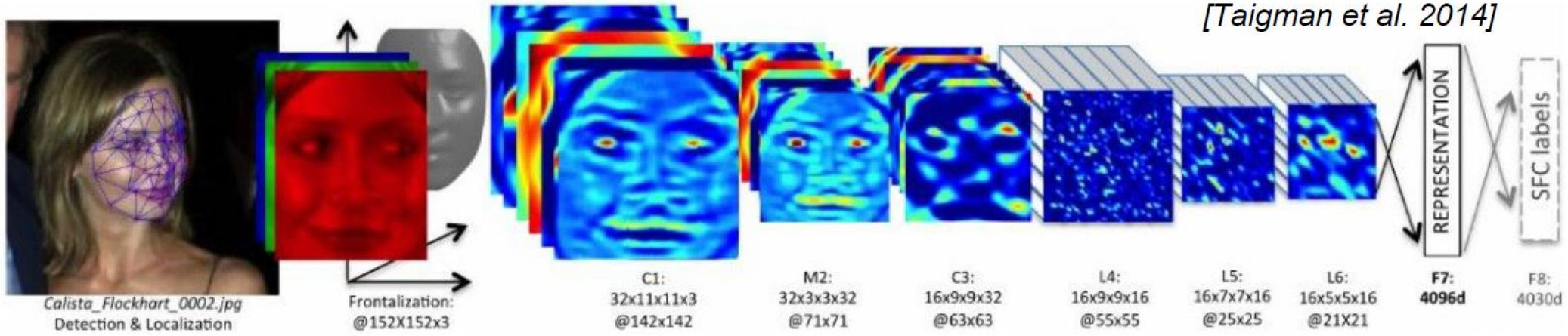


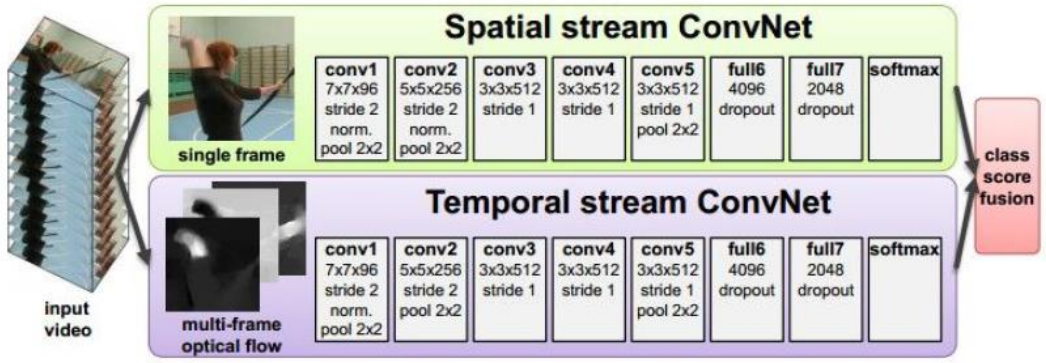
[Farabet et al., 2012]

Convolution Networks are everywhere now

self-driving cars

Convolution Networks are everywhere now

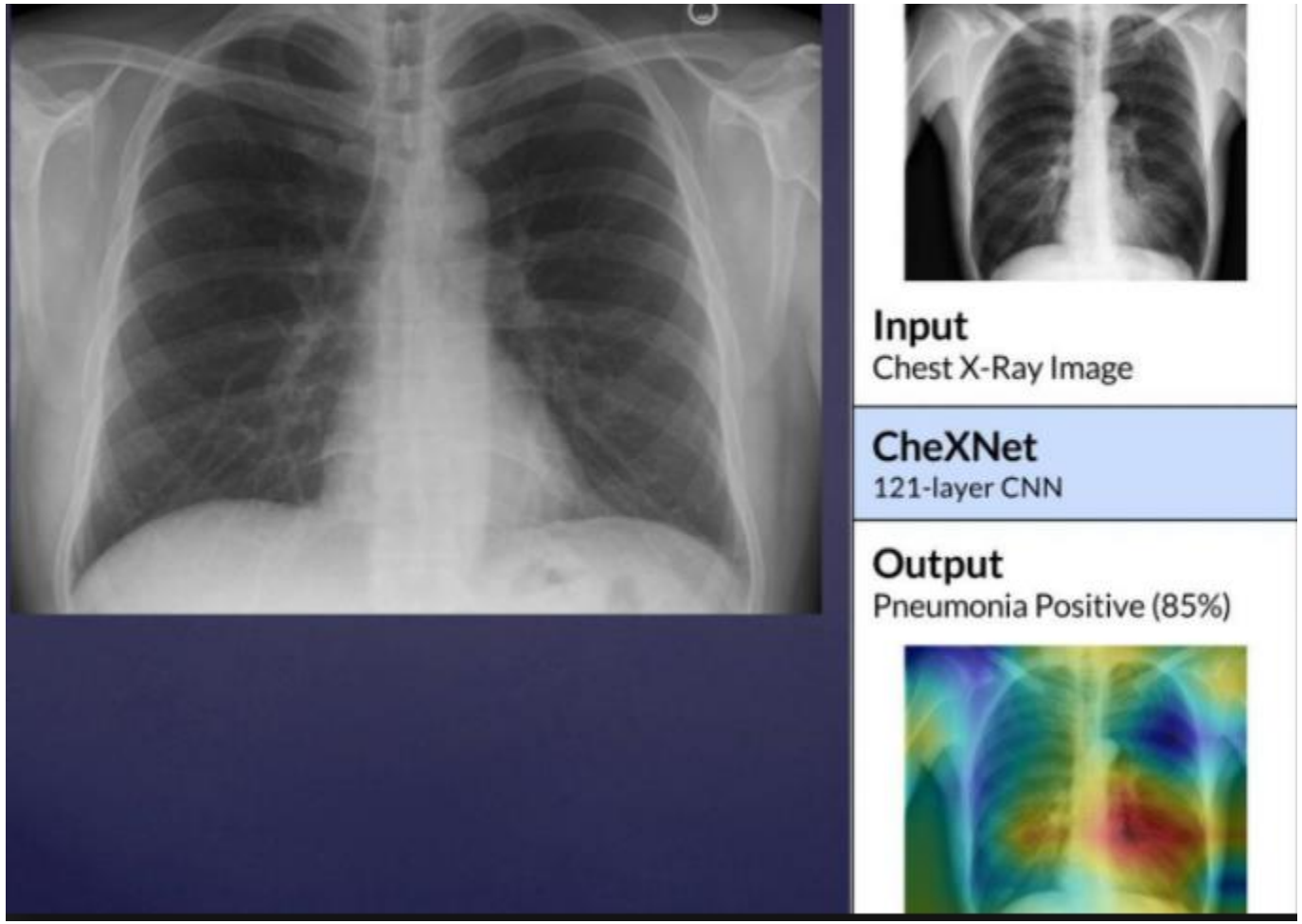




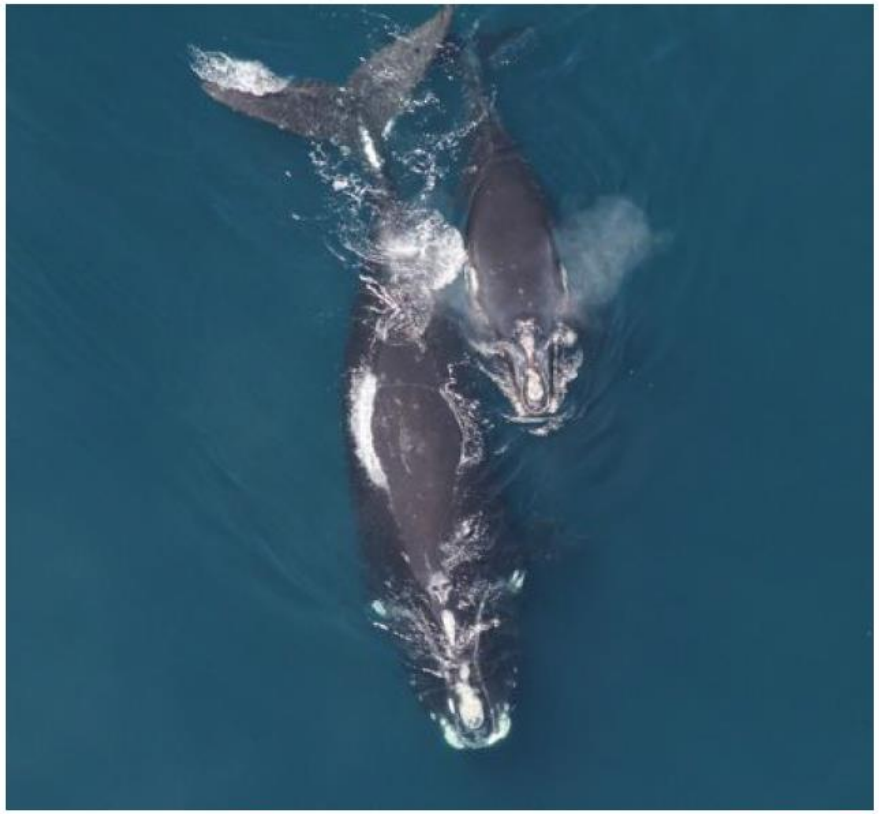
[Simonyan et al. 2014]

[Goodfellow 2014]

CNNs in healthcare



Convolution Networks are everywhere now



Whale recognition, Kaggle Challenge

Mnih and Hinton, 2010

Convolution Networks are everywhere now

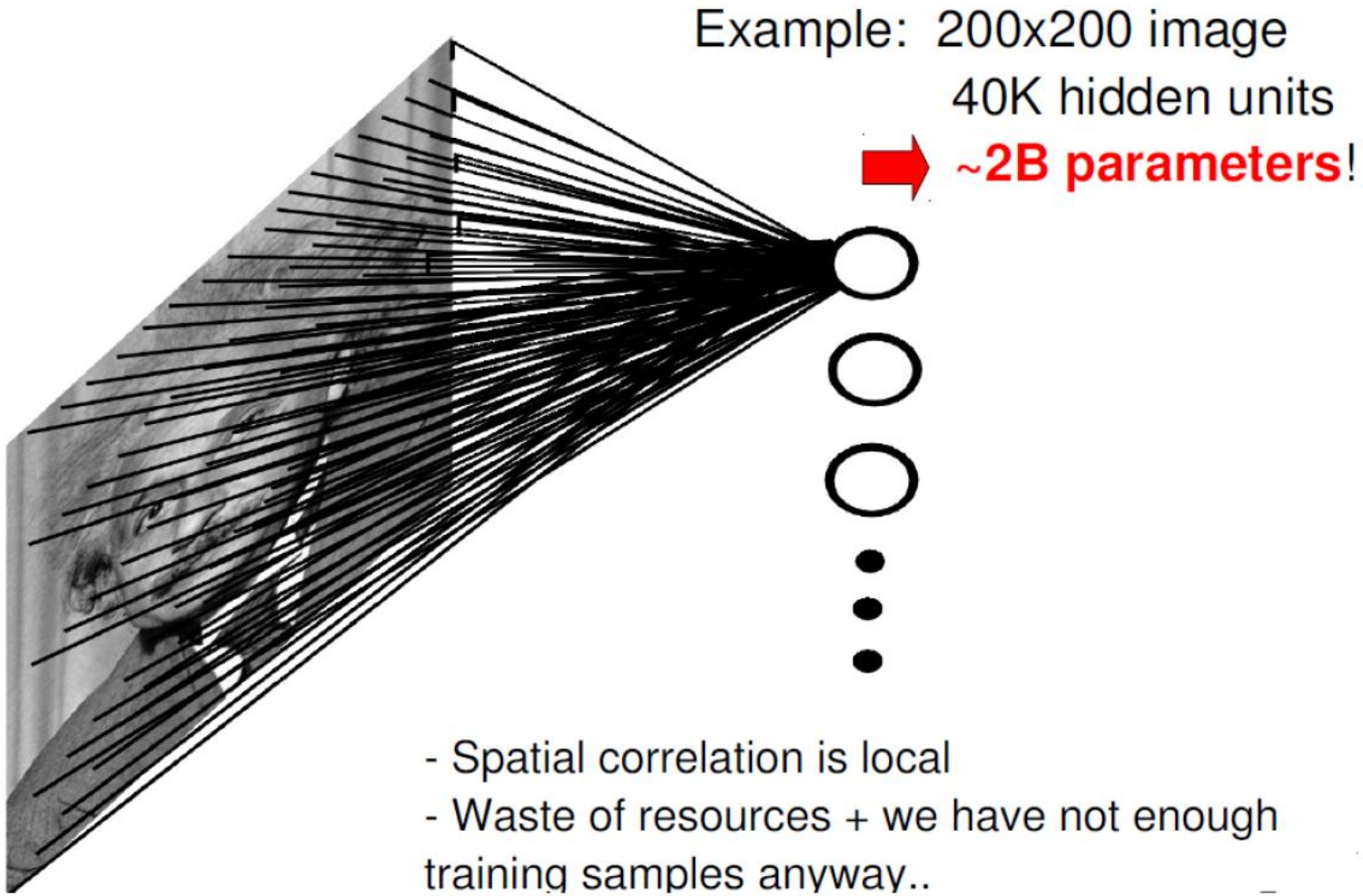
Image Captioning

[Vinyals et al., 2015]

Classifying an image using MLPs?

- In CIFAR-10 dataset, images are only of size 32x32x3 (32 wide, 32 high, 3 color channels), so a single fully connected neuron in a first hidden layer of a regular neural network would have 32*32*3 = 3,072 weights.
- A 200x200 image, however, would lead to neurons that have 200*200*3 = 120,000 weights.
- Such network architecture does not take into account the spatial structure of data, treating input pixels which are far apart and close together on exactly the same footing.
- Clearly, the full connectivity of neurons is wasteful in the framework of image recognition, and the huge number of parameters quickly leads to overfitting.

Parameter explosion with MLPs



Today's Agenda

- ImageNet and visual recognition problems
- Introduction to CNNs and applications
- **Technical details of a CNN**

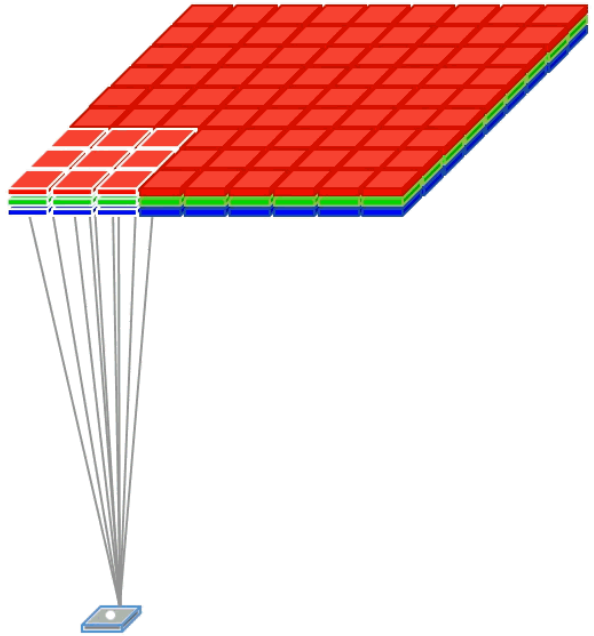
What is convolution of an image with a filter?

1	1	1	0	0
0	1	1	1	0
0	0	$1_{\times 1}$	$1_{\times 0}$	$1_{\times 1}$
0	0	$1_{\times 0}$	$1_{\times 1}$	$0_{\times 0}$
0	1	$1_{\times 1}$	$0_{\times 0}$	$0_{\times 1}$

Image

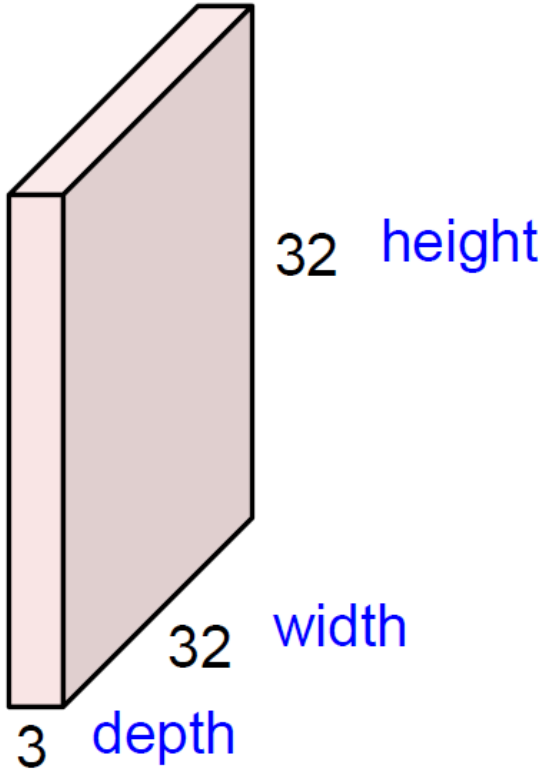
4	3	4
2	4	3
2	3	4

Convolved Feature



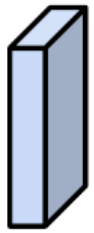
Convolution Layer

32x32x3 image



Filters always extend the full depth of the input volume

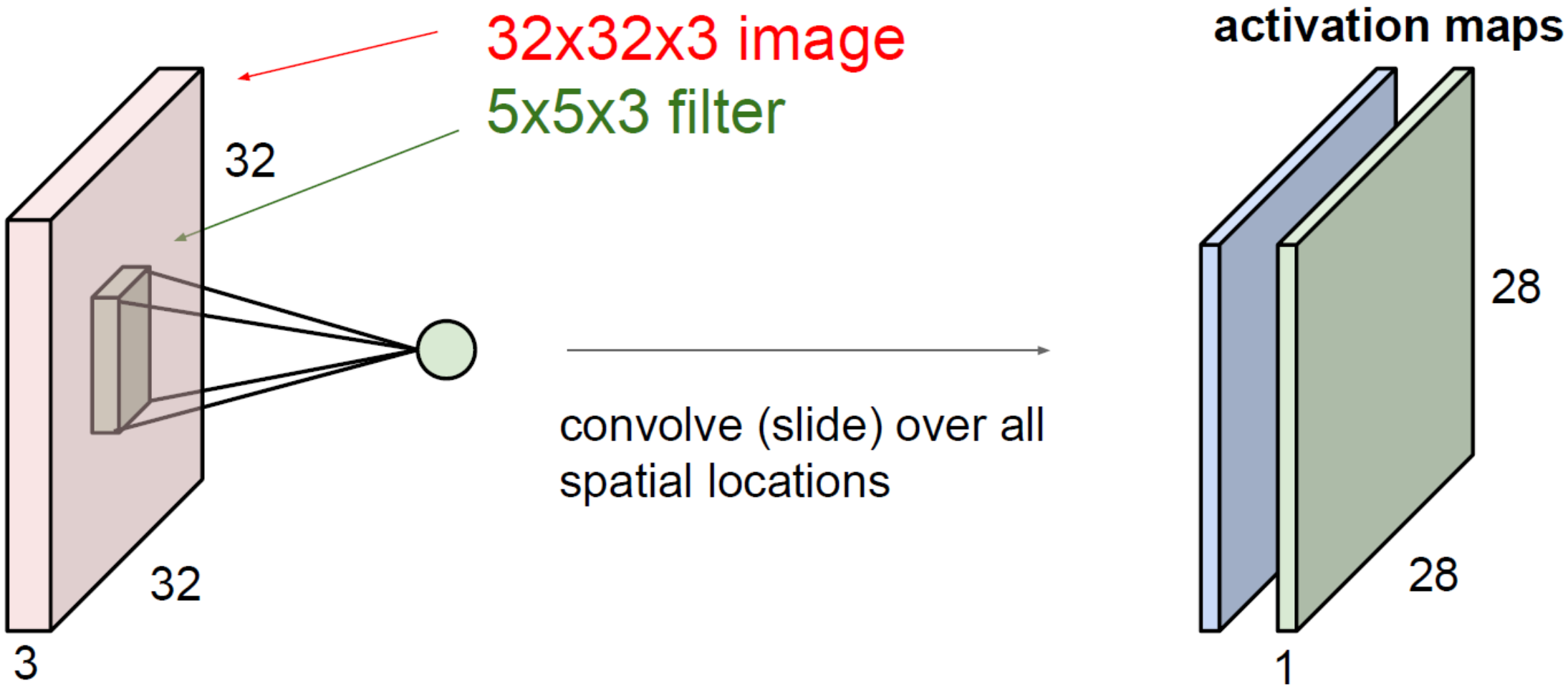
5x5x3 filter



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

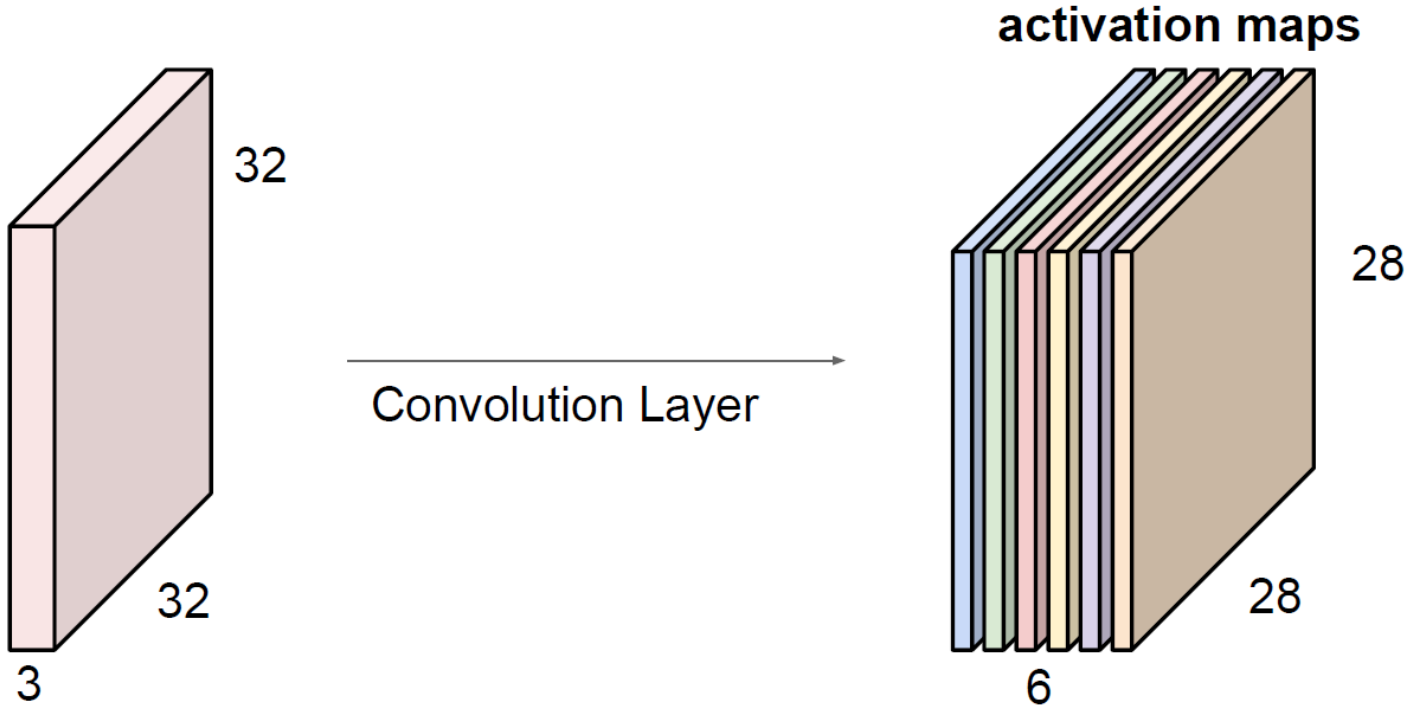
consider a second, green filter

Convolution Layer



Convolution Layer

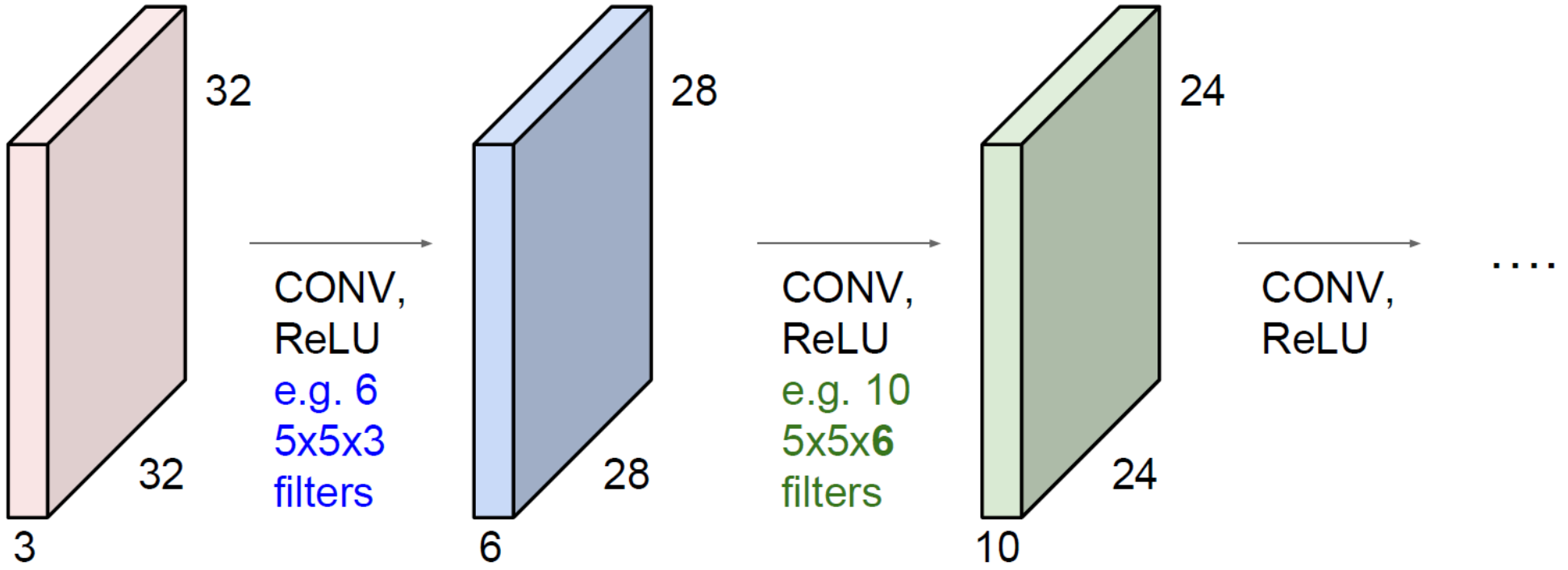
For example, if we had 6 5x5 filters, we'll get 6 separate activation maps:



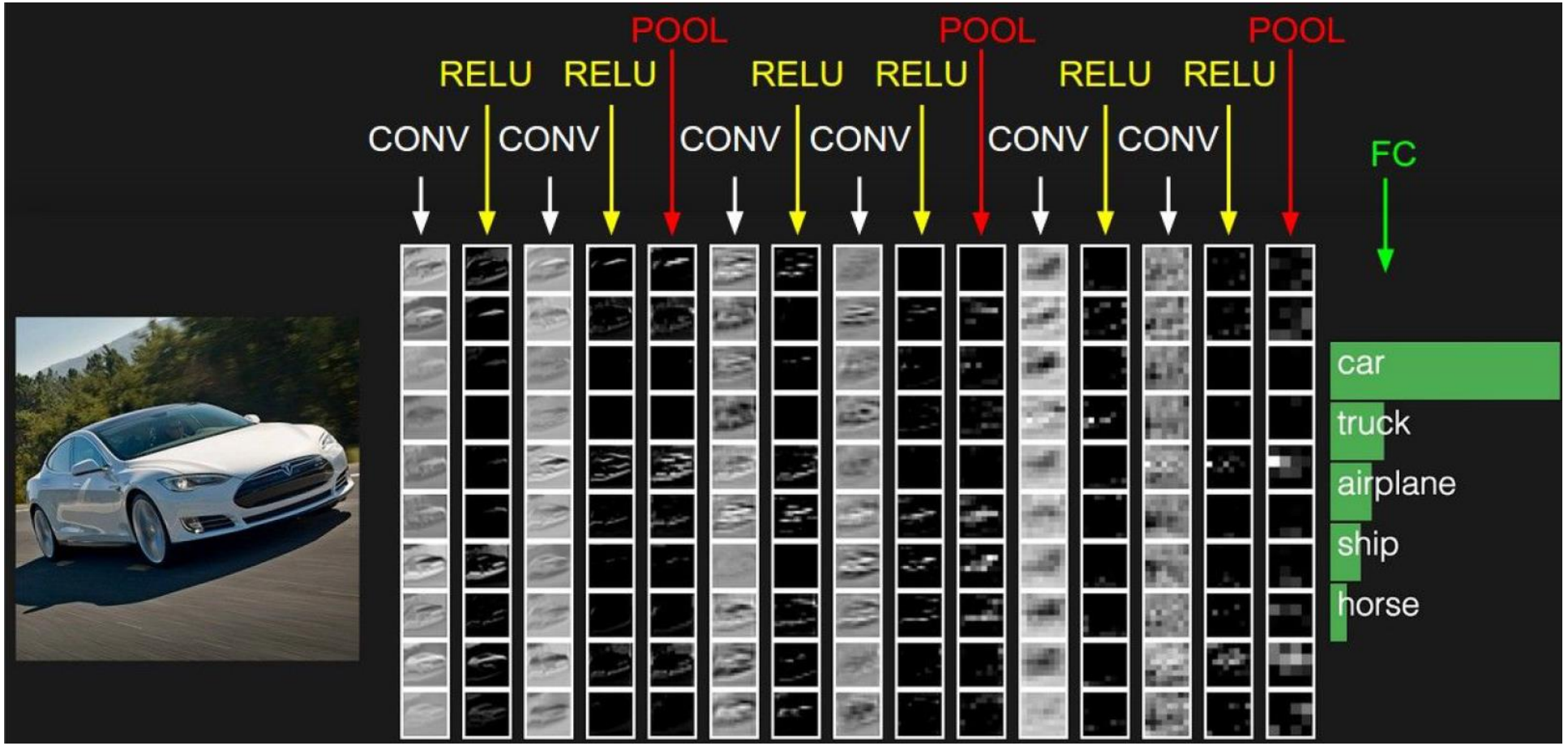
We stack these up to get a "new image" of size 28x28x6!

A Basic ConvNet

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

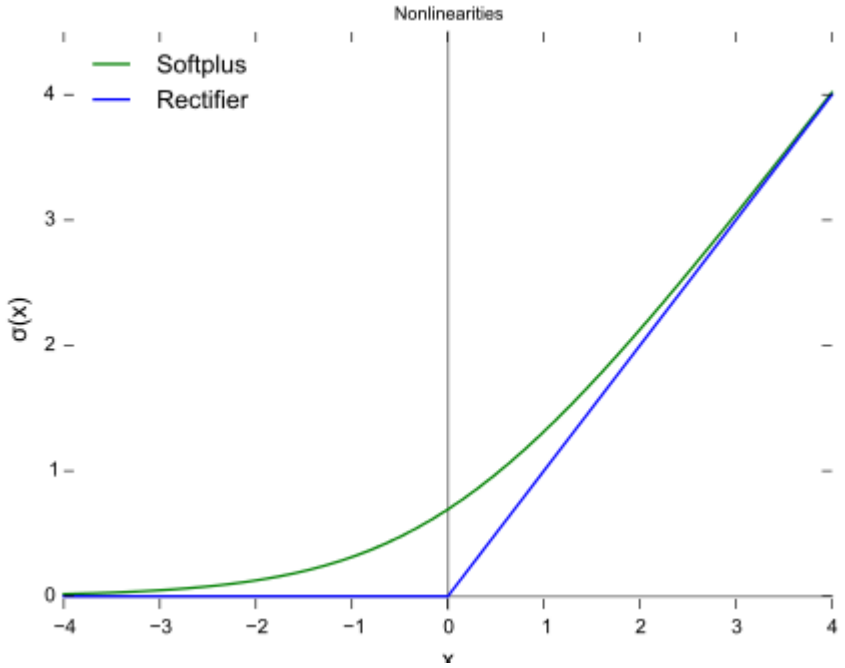


ConvNet with Pooling and FC Layers



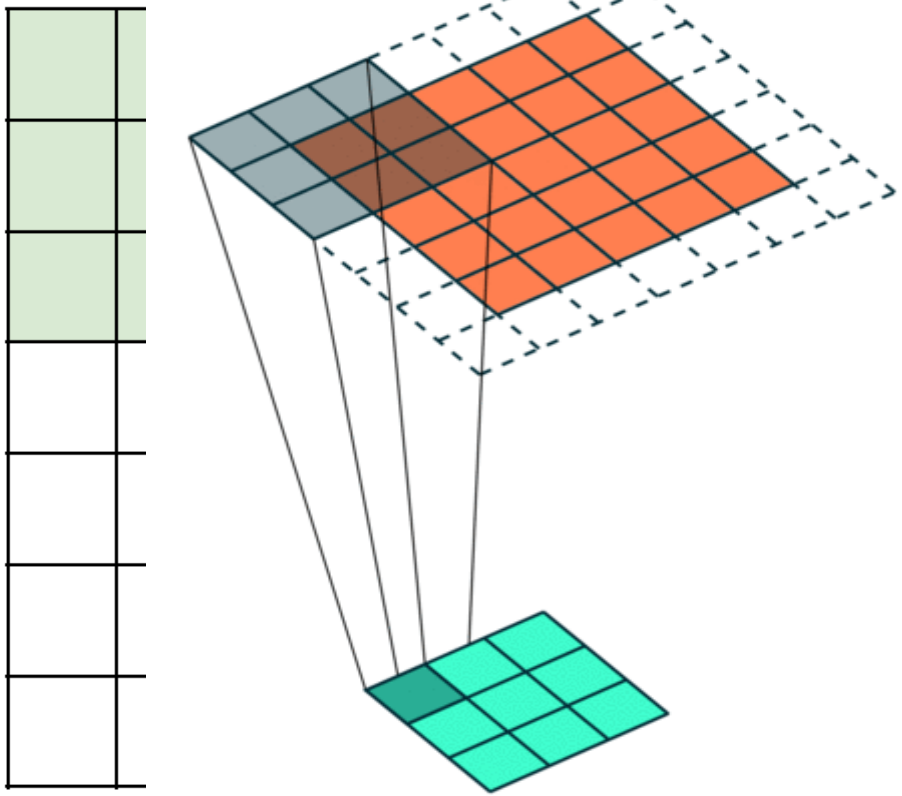
ReLU (Rectified Linear Units) Layer

- This is a layer of neurons that applies the activation function f(x)=max(0,x).
- Other functions are also used to increase nonlinearity, for example the hyperbolic tangent f(x)=tanh(x), and the sigmoid function.
- This is also known as a ramp function.

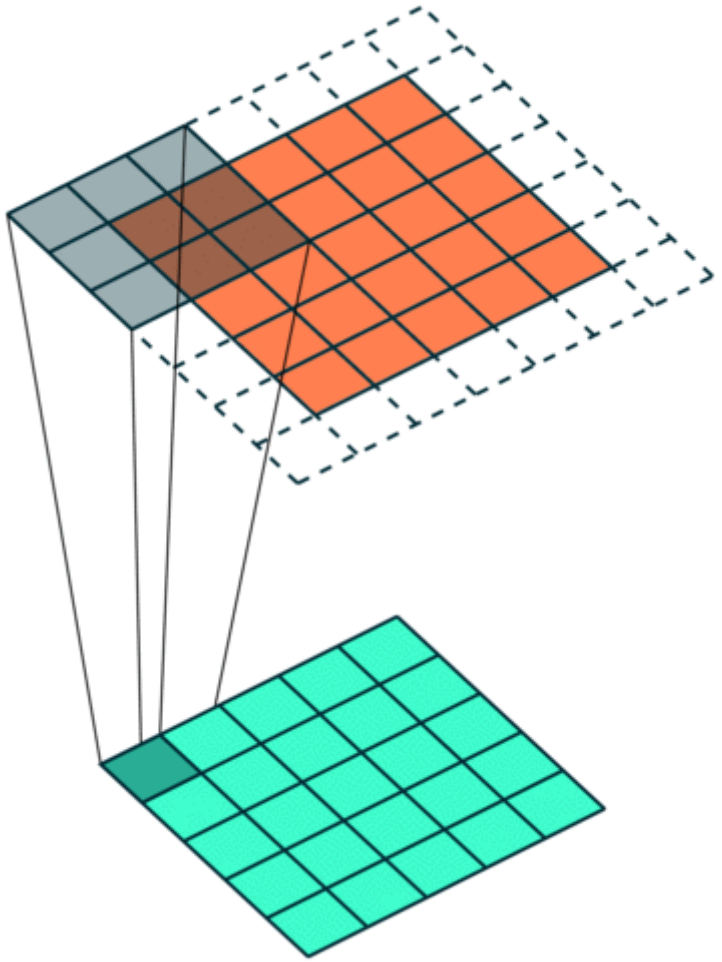


Details about the convolution layer

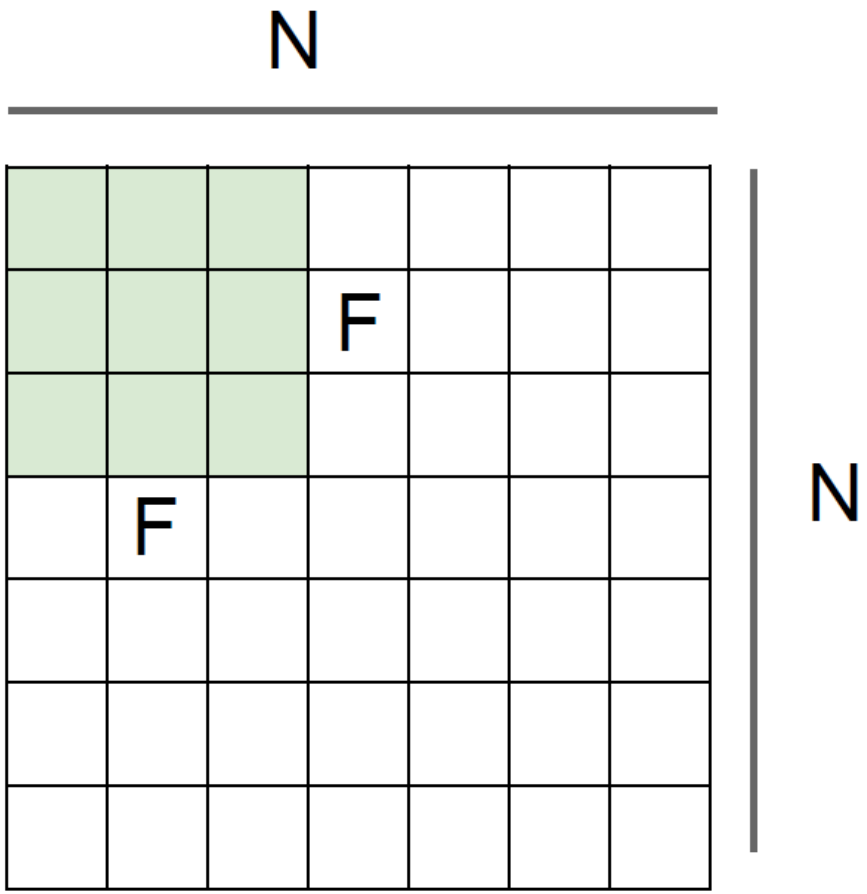
Stride=2



Stride=1



Details about the convolution layer

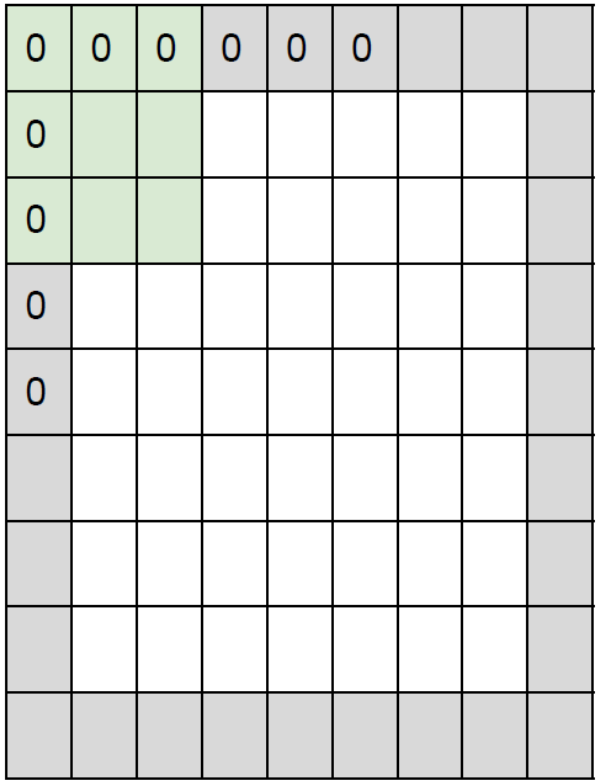


Output size:
(N - F) / stride + 1

e.g. N = 7, F = 3:
stride 1 => (7 - 3)/1 + 1 = 5
stride 2 => (7 - 3)/2 + 1 = 3
stride 3 => (7 - 3)/3 + 1 = 2.33

Details about the convolution layer

In practice: Common to zero pad the border



e.g. input 7x7
3x3 filter, applied with **stride 1**
pad with 1 pixel border => what is the output?

7x7 output!
in general, common to see CONV layers with stride 1, filters of size FxF, and zero-padding with (F-1)/2. (will preserve size spatially)
e.g. F = 3 => zero pad with 1
F = 5 => zero pad with 2
F = 7 => zero pad with 3

Convolution layer examples

Input volume: **32x32x3**
10 5x5 filters with stride 1, pad 2

Output volume size: ?

(32+2*2-5)/1+1 = 32 spatially, so
32x32x10

Convolution layer examples

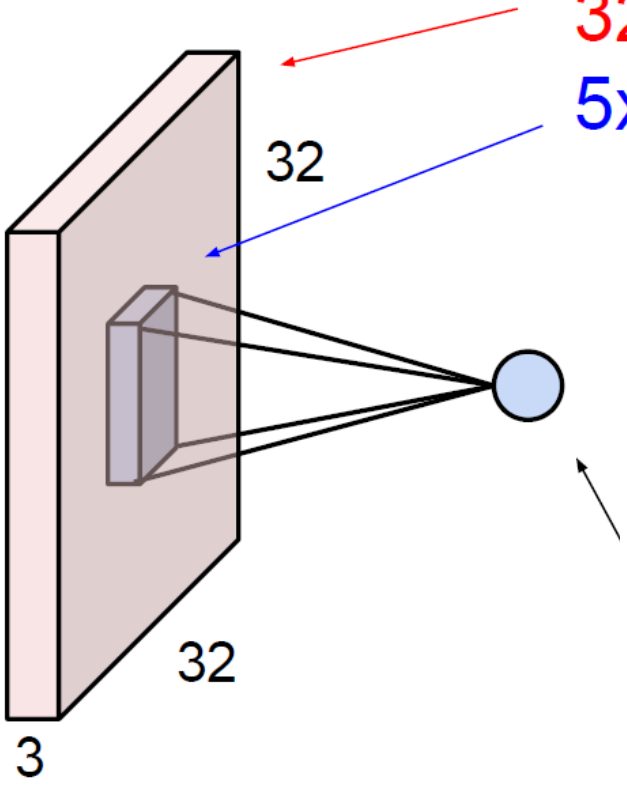
Input volume: **32x32x3**
10 5x5 filters with stride 1, pad 2

Number of parameters in this layer?

each filter has 5*5*3 + 1 = 76 params (+1 for bias)
=> 76*10 = **760**

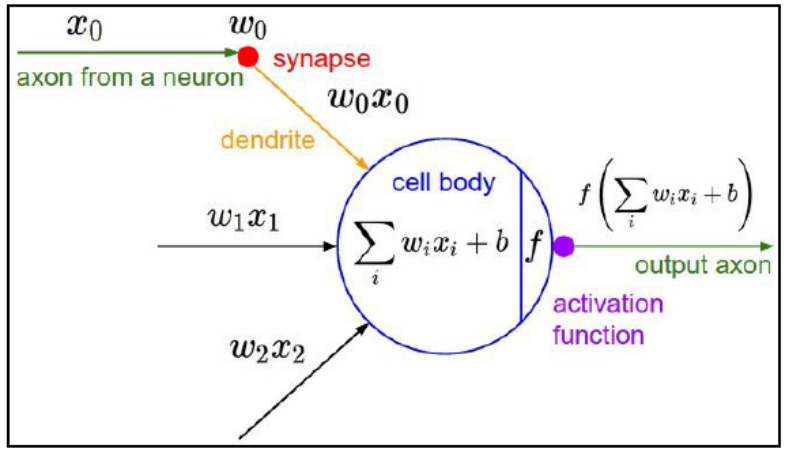
Convolutional networks are simply neural networks that use convolution in place of general matrix multiplication in at least one of their layers.

Neuron view of the convolution layer



32x32x3 image

5x5x3 filter

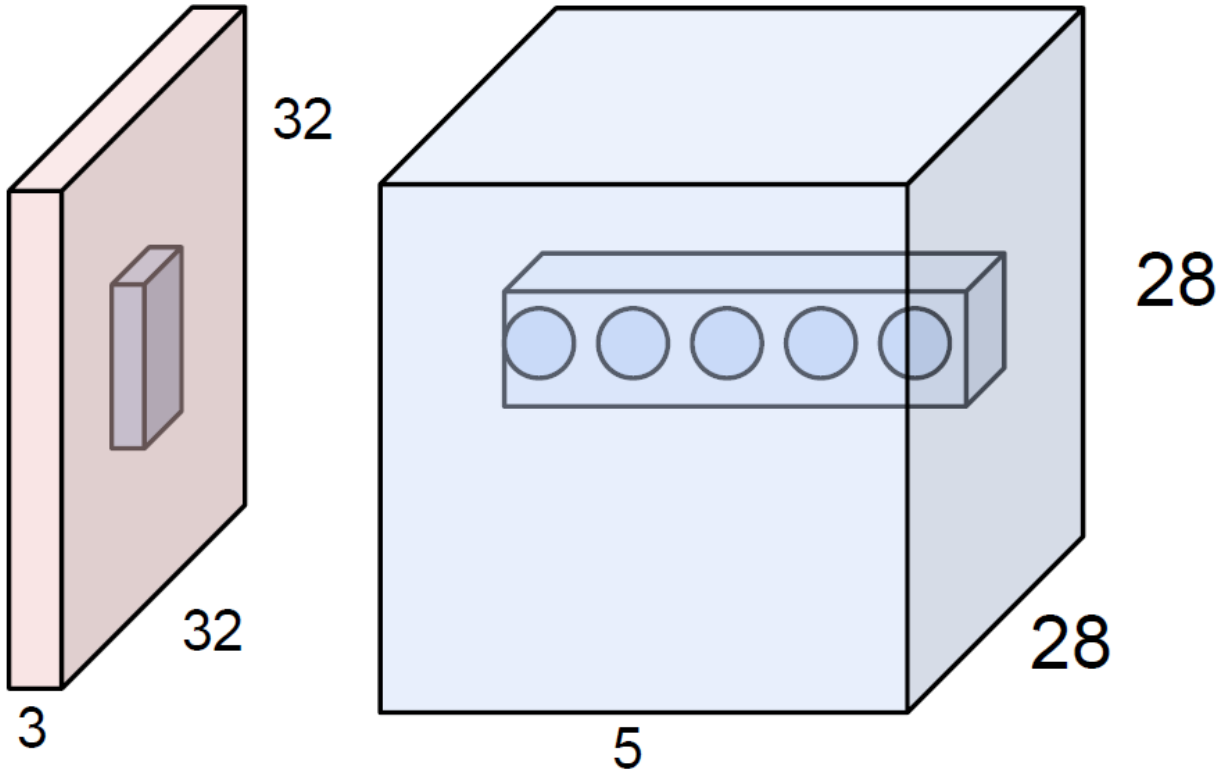


An activation map is a 28x28 sheet of neuron outputs:

1. Each is connected to a small region in the input
2. All of them share parameters

"5x5 filter" -> "5x5 receptive field for each neuron"

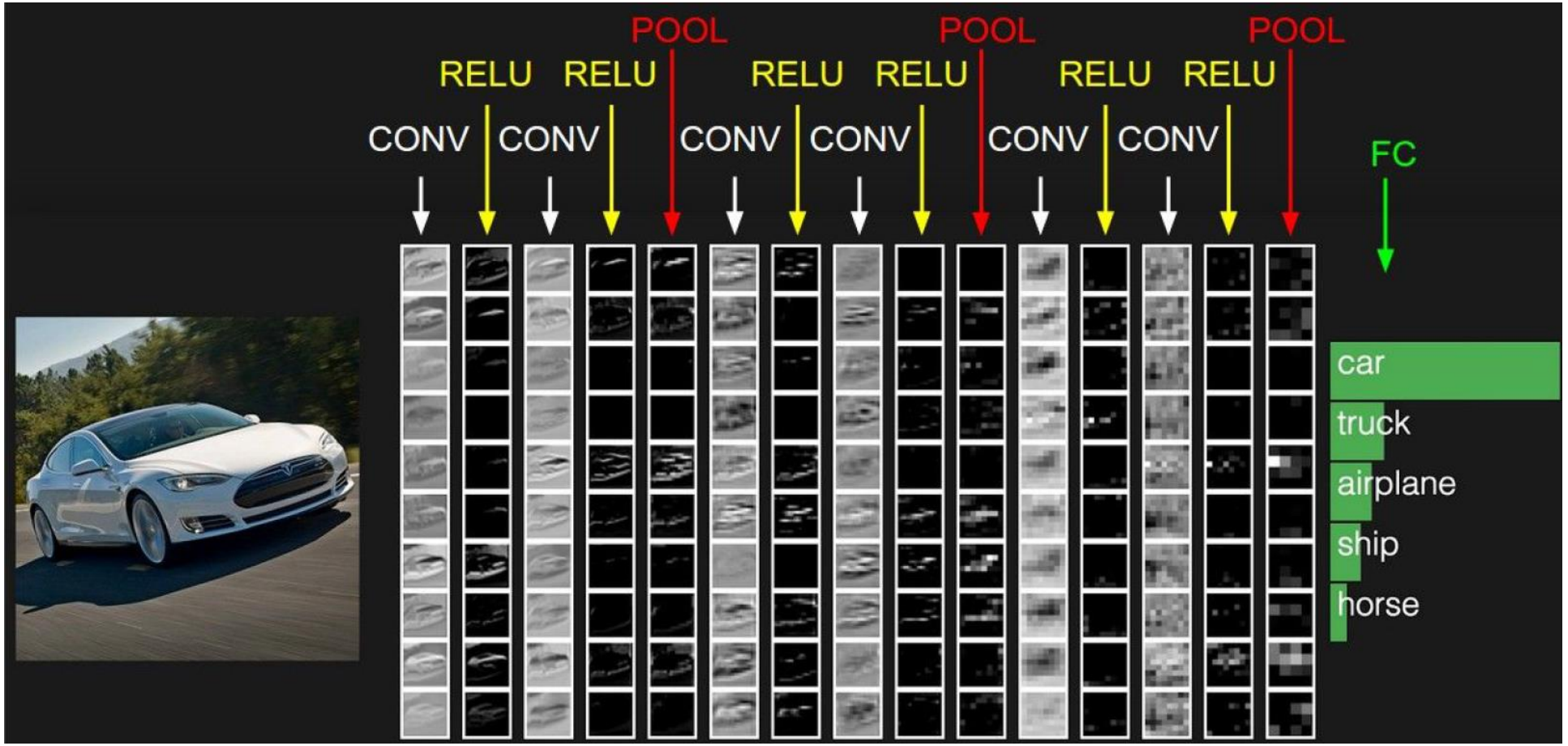
Neuron view of the convolution layer



E.g. with 5 filters,
CONV layer consists of
neurons arranged in a 3D grid
(28x28x5)

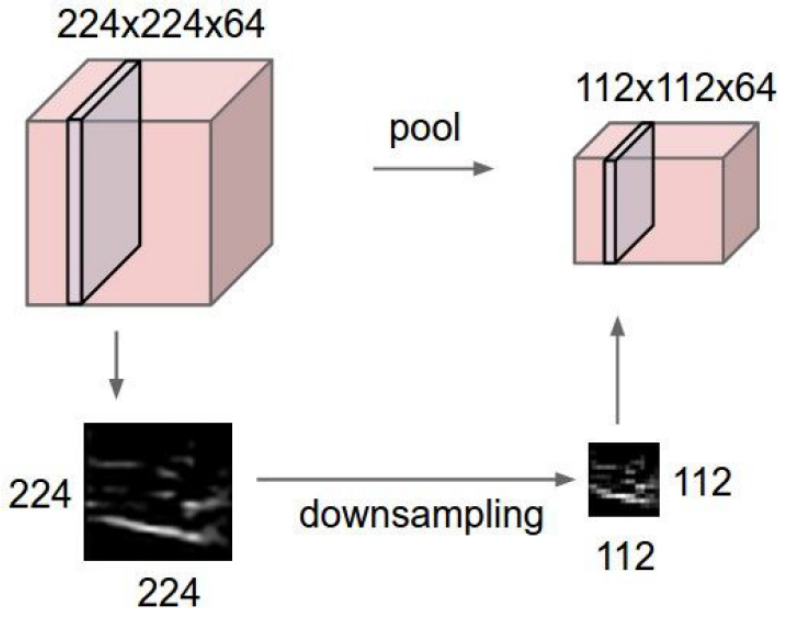
There will be 5 different
neurons all looking at the same
region in the input volume

ConvNet with Pooling and FC Layers



Pooling Layer

makes the representations smaller and more manageable
operates over each activation map independently:



Single depth slice

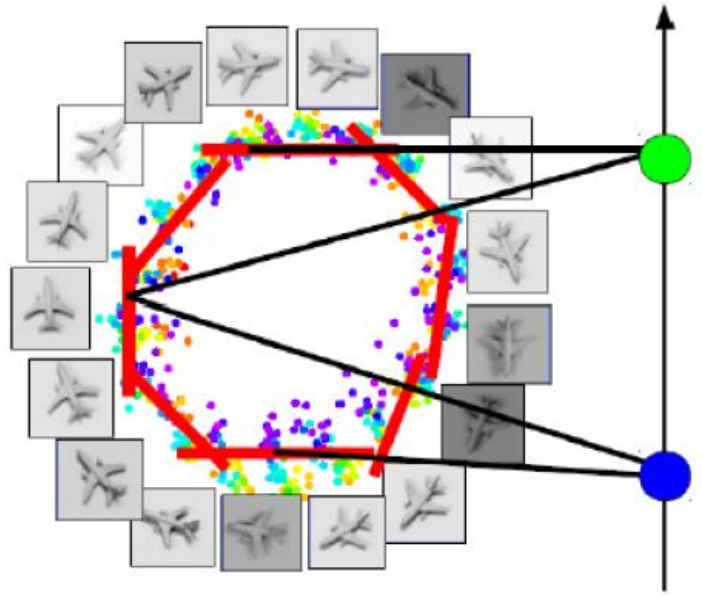
1	1	2	4
5	6	7	8
3	2	1	0
1	2	3	4

x

y

max pool with 2x2 filters
and stride 2

6	8
3	4

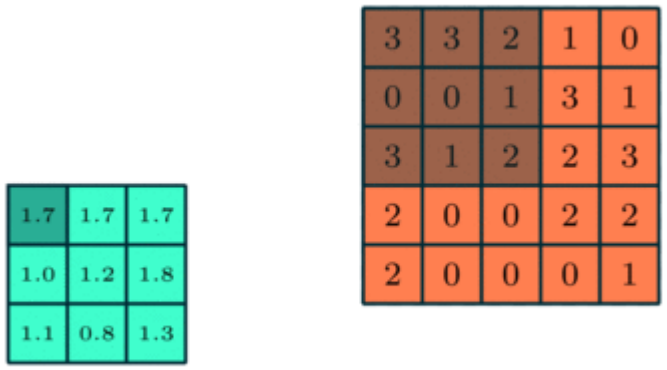


- Invariance to image transformation and increases compactness to representation.
- Pooling types: Max, Average, L2 etc.

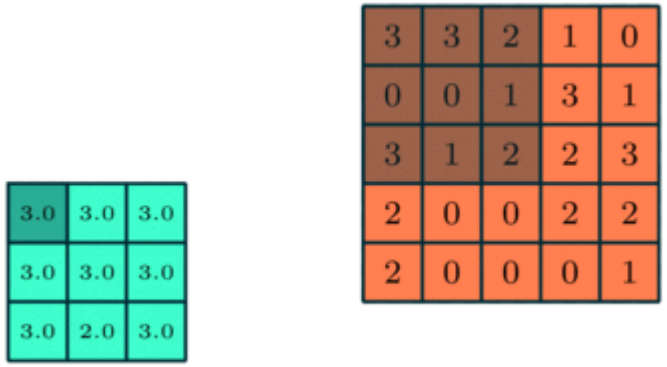
Invariance to local translation can be a very useful property if we care more about whether some feature is present than exactly where it is.

Average and Max Pooling

Average Pooling

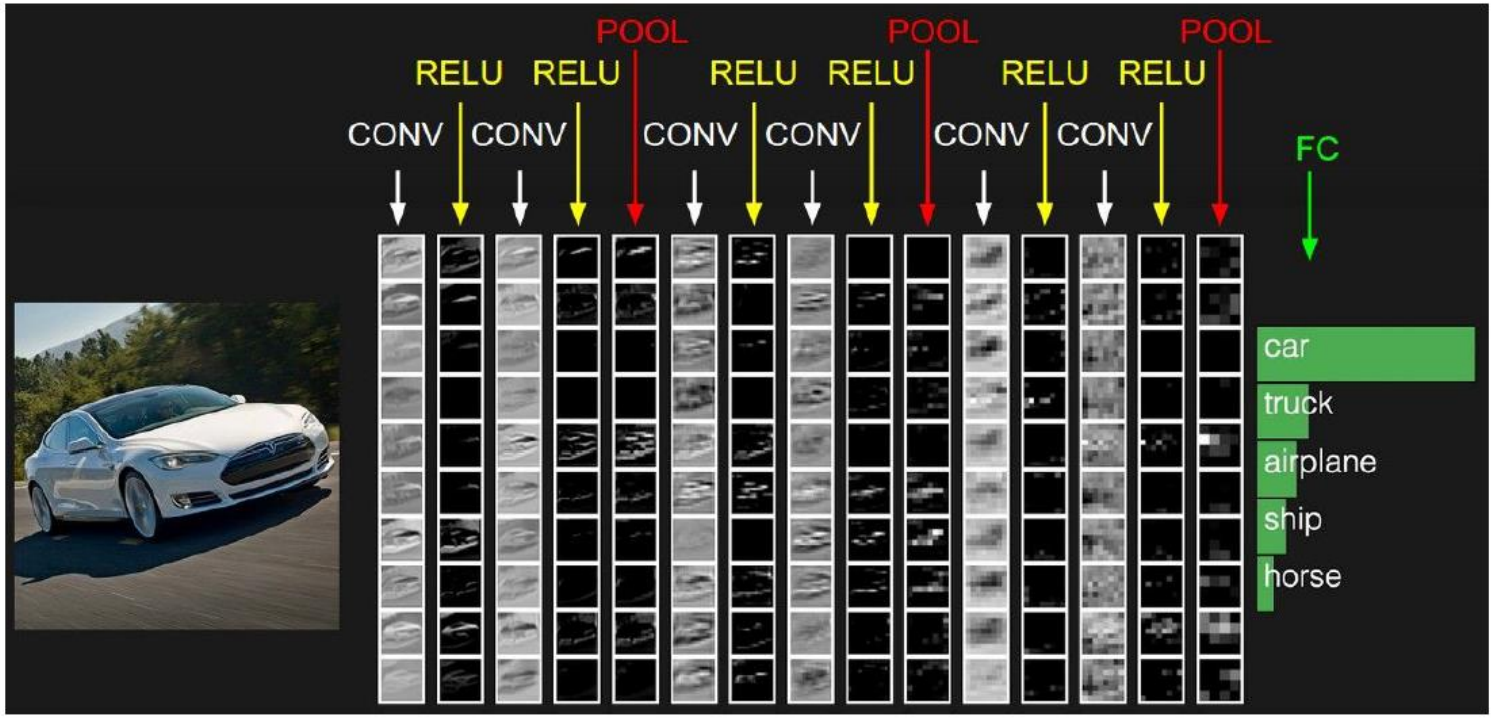


Max Pooling



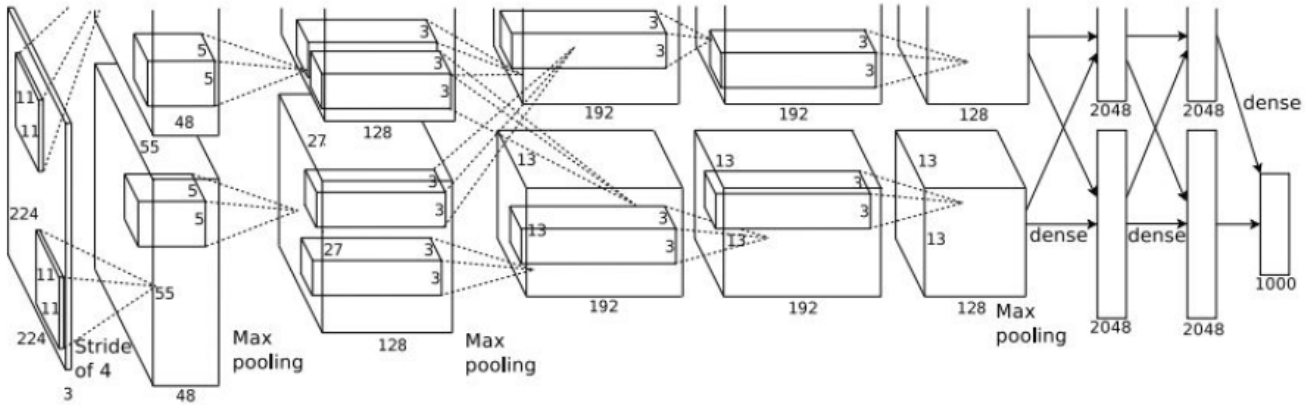
Fully Connected Layer

Contains neurons that connect to the entire input volume, as in ordinary Neural Networks



Case Study: AlexNet

[Krizhevsky et al. 2012]



Full (simplified) AlexNet architecture:
[227x227x3] INPUT
[55x55x96] CONV1: 96 11x11 filters at stride 4, pad 0
[27x27x96] MAX POOL1: 3x3 filters at stride 2
[27x27x96] NORM1: Normalization layer
[27x27x256] CONV2: 256 5x5 filters at stride 1, pad 2
[13x13x256] MAX POOL2: 3x3 filters at stride 2
[13x13x256] NORM2: Normalization layer
[13x13x384] CONV3: 384 3x3 filters at stride 1, pad 1
[13x13x384] CONV4: 384 3x3 filters at stride 1, pad 1
[13x13x256] CONV5: 256 3x3 filters at stride 1, pad 1
[6x6x256] MAX POOL3: 3x3 filters at stride 2
[4096] FC6: 4096 neurons
[4096] FC7: 4096 neurons
[1000] FC8: 1000 neurons (class scores)

Details/Retrospectives:
- first use of ReLU
- used Norm layers (not common anymore)
- heavy data augmentation
- dropout 0.5
- batch size 128
- SGD Momentum 0.9
- Learning rate 1e-2, reduced by 10 manually when val accuracy plateaus
- L2 weight decay 5e-4
- 7 CNN ensemble: 18.2% -> 15.4%

Take-aways

- CNNs are very popular these days across a large variety of tasks.
- Convolution networks are inspired by the hierarchical structure of the visual cortex.
- Things that differentiate CNNs from DNNs are Sparse connectivity, shared weights, feature maps and pooling.
- CNN feature visualization: http://people.csail.mit.edu/torralba/research/drawCNN/drawNet.html?path=imagenetCNN