Convolutional Neural Networks

Manish Gupta

Senior Applied Scientist at Microsoft, India

Adjunct Faculty at IIIT-H

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

What is deep learning?

Deep learning

From Wikipedia, the free encyclopedia

Deep learning (*deep machine learning*, or *deep structured learning*, or *hierarchical learning*, or sometimes *DL*) is a branch of machine learning based on a set of algorithms that attempt to model high-level abstractions in data by using model architectures, with complex structures or otherwise, composed of multiple non-linear transformations.[1] (p198)[2][3][4]

The Universal Translator ... *comes true!*

Deep learning technology enabled speech-to-speech translation

The New York Times

Scientists See Promise in Deep-Learning Programs

John Markoff

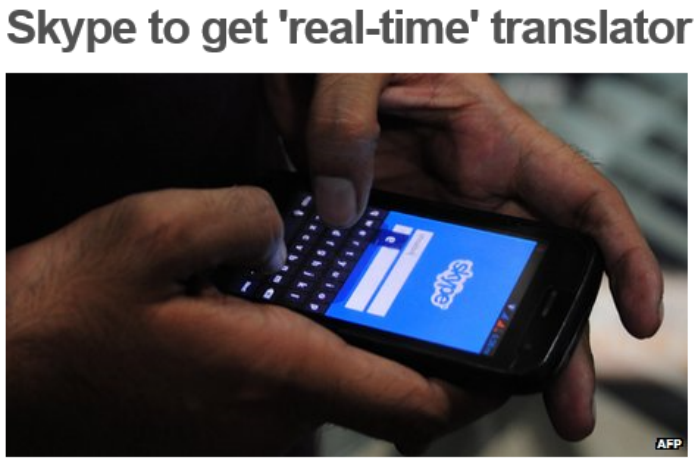
November 23, 2012

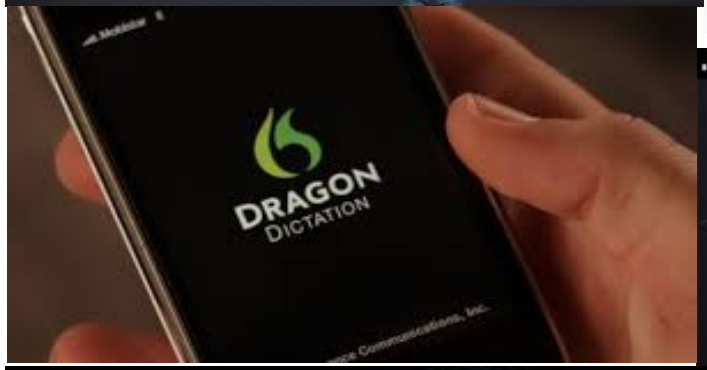
Tianjin, China, October, 25, 2012

Microsoft Research

A voice recognition program translated a speech given by Richard F. Rashid, Microsoft's top scientist, into Mandarin Chinese

Across-the-Board Deployment of DNN in Speech Industry





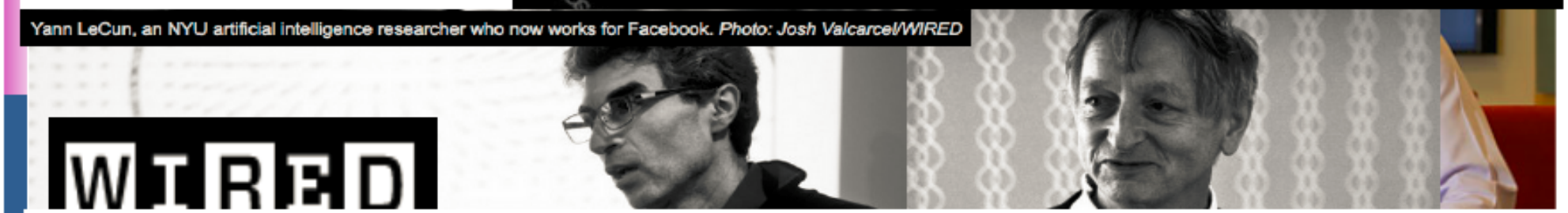
Deep Learning in the News

The New York Tim

Monday, June 25, 2012 Last Update: 11:50 PM ET

EXCLUSIVE

Facebook, Google in 'Deep Learning' Arms Race



Yann LeCun, an NYU artificial intelligence researcher who now works for Facebook. *Photo: Josh Valcarcel/WIRED*

WIRED

NEWS BULLETIN

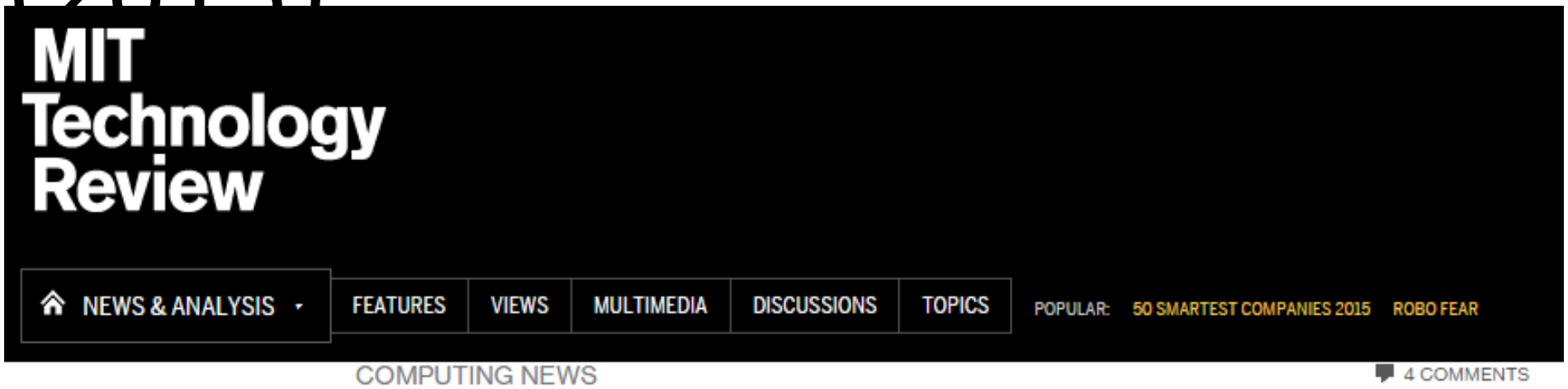
Google Beat Facebook for DeepMind

Google Acquires Artificial Intelligence Startup DeepMind For More Than $500M

Posted Jan 26, 2014 by Catherine Shu (@catherineshu)

(Slide from Bengio, 2015)

More Recent Media Reports (2015)



MIT Technology Review

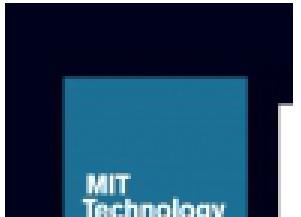
COMPUTING NEWS

IBM Pushes Deep Learning with a Watson Upgrade

IBM is combining different AI techniques, including deep learning, in the commercial version of Watson.

By Will Knight on July 9, 2015

IBM's *Jeopardy!*-playing computer system, Watson, combined two separate areas of artificial intelligence research with winning results. Natural language understanding was merged with statistical analysis of vast, unstructured piles of text to find the likely answers to cryptic *Jeopardy!* clues.



nature International weekly journal of science

LETTER

doi:10.1038/nature14236

Human-level control through deep reinforcement learning

Volodymyr Mnih[1]*, Koray Kavukcuoglu[1]*, David Silver[1]*, Andrei A. Rusu[1], Joel Veness[1], Marc G. Bellemare[1], Alex Graves[1], Martin Riedmiller[1], Andreas K. Fidjeland[1], Georg Ostrovski[1], Stig Petersen[1], Charles Beattie[1], Amir Sadik[1], Ioannis Antonoglou[1], Helen King[1], Dharshan Kumaran[1], Daan Wierstra[1], Shane Legg[1] & Demis Hassabis[1]

The theory of reinforcement learning provides a normative account[1], deeply rooted in psychological[2] and neuroscientific[3] perspectives on animal behaviour, of how agents may optimize their control of an environment. To use reinforcement learning successfully in situations approaching real-world complexity, however, agents are confronted

agent is to select actions in a fashion that maximizes cumulative future reward. More formally, we use a deep convolutional neural network to approximate the optimal action-value function

$$Q^*(s,a) = \max_{\pi} \mathbb{E}\left[r_t + \gamma r_{t+1} + \gamma^2 r_{t+2} + \ldots \mid s_t = s,\ a_t = a,\ \pi\right],$$

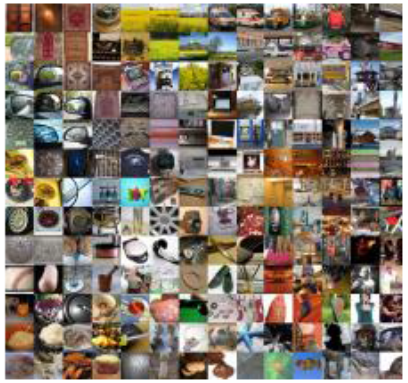
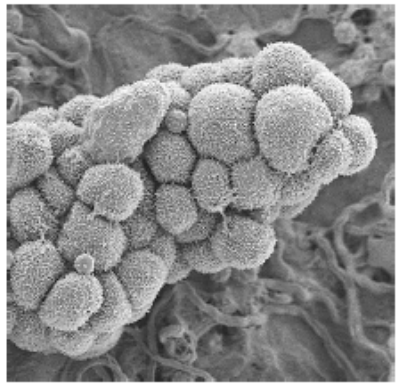
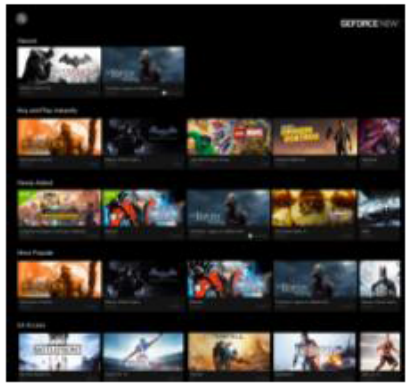
Google DeepMind

ep-Q-Net = DNN + Q-learning

Breakthrough research!

Deep learning is everywhere

DEEP LEARNING EVERYWHERE

INTERNET & CLOUD

Image Classification
Speech Recognition
Language Translation
Language Processing
Sentiment Analysis
Recommendation

MEDICINE & BIOLOGY

Cancer Cell Detection
Diabetic Grading
Drug Discovery

MEDIA & ENTERTAINMENT

Video Captioning
Video Search
Real Time Translation

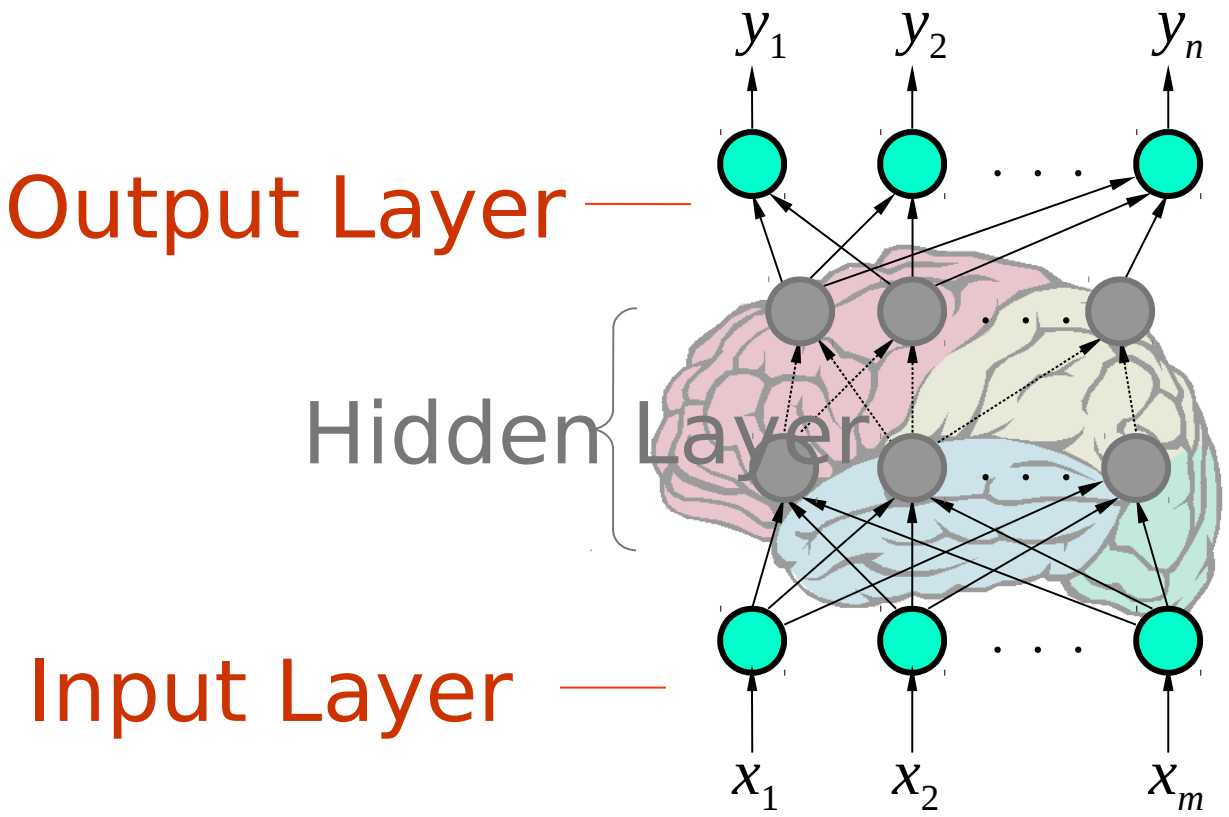
SECURITY & DEFENSE

Face Detection
Video Surveillance
Satellite Imagery

AUTONOMOUS MACHINES

Pedestrian Detection
Lane Tracking
Recognize Traffic Sign

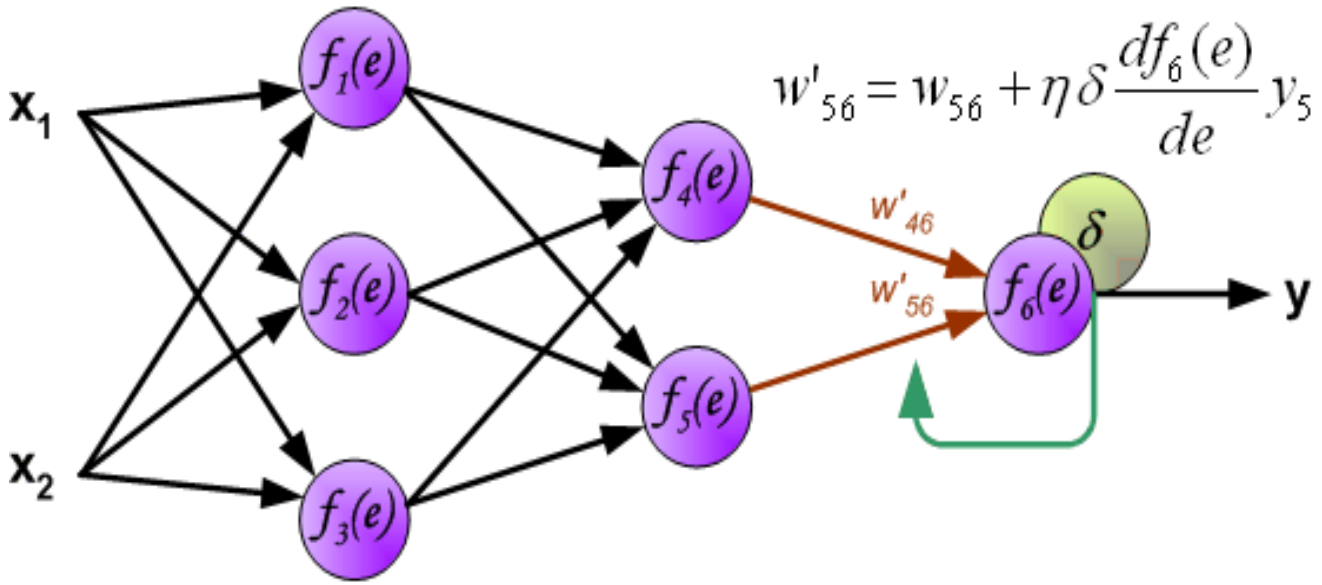
Multi-Layered Perceptrons (MLPs)



Back Propagation Demo

$$w'_{46} = w_{46} + \eta\,\delta\,\frac{df_6(e)}{de}\,y_4$$

$$w'_{56} = w_{56} + \eta\,\delta\,\frac{df_6(e)}{de}\,y_5$$



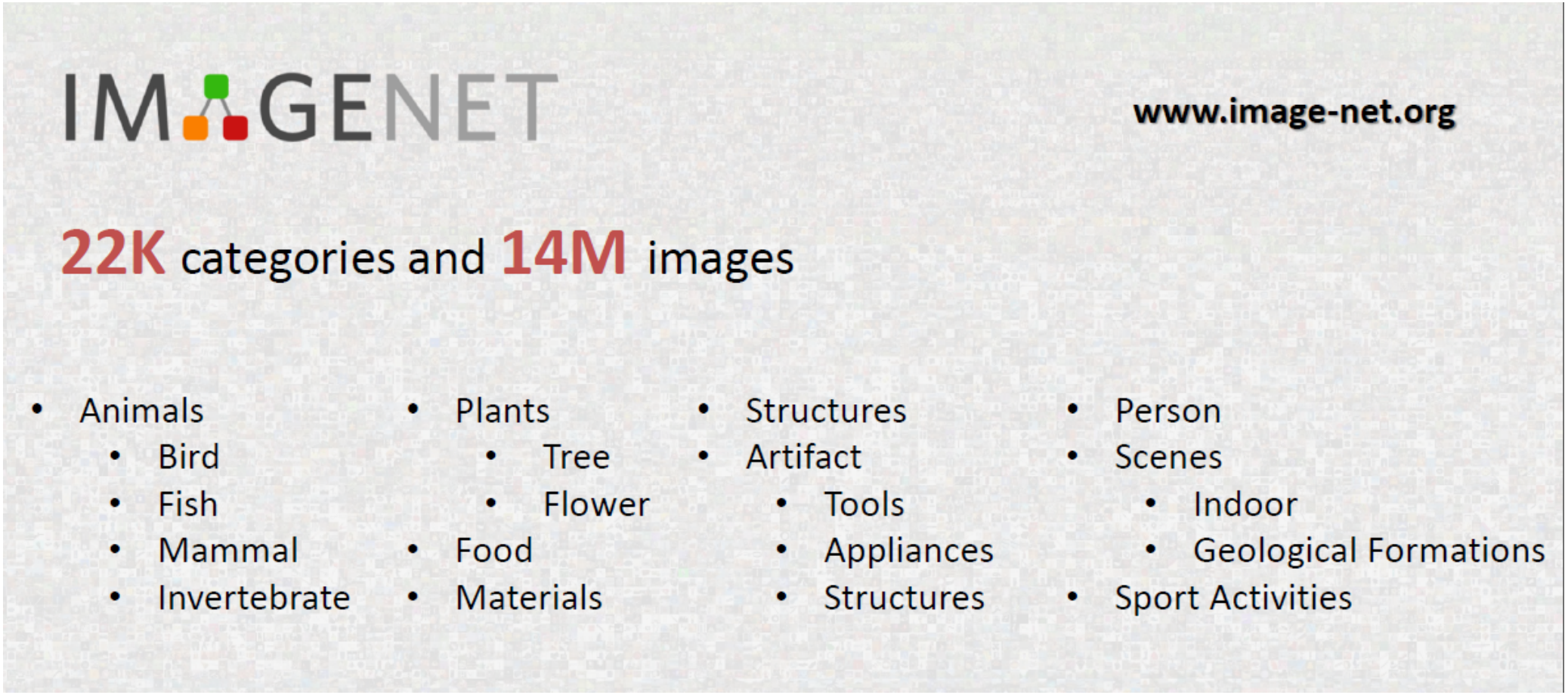
Today’s Agenda

- ImageNet and visual recognition problems
- Introduction to CNNs and applications
- Technical details of a CNN
- Deep Semantic Similarity Model (DSSM)

Today’s Agenda

- **ImageNet and visual recognition problems**
- Introduction to CNNs and applications
- Technical details of a CNN
- Deep Semantic Similarity Model (DSSM)

ImageNet



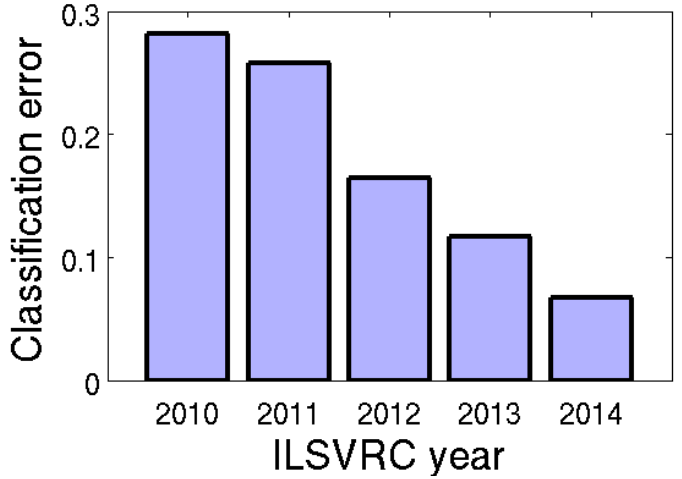
http://image-net.org/download-faq

ImageNet Classification Challenge

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

Steel drum

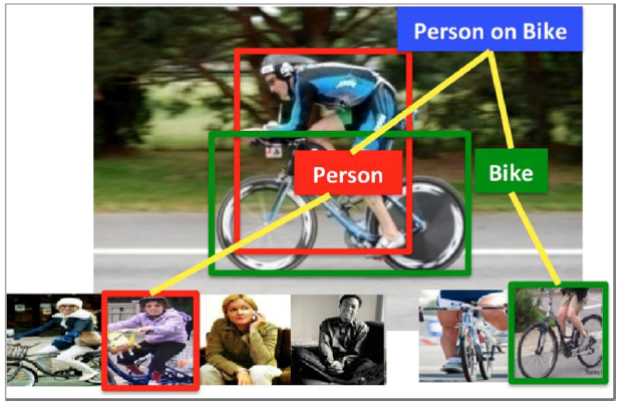
Giant panda



Visual Recognition Problems

- There is a number of visual recognition problems that are related to image classification, such as object detection, im

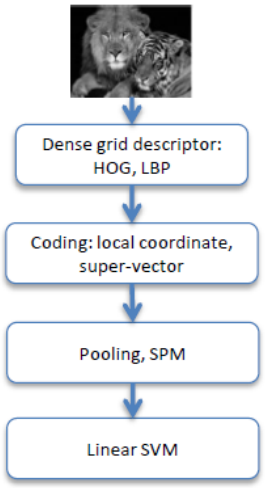
- Object detection
- Action classification
- Image captioning
- …



ImageNet Challenge Winning Architectures

Year 2010

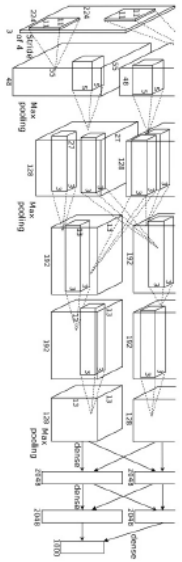
NEC-UIUC



[Lin CVPR 2011]

Year 2012

SuperVision

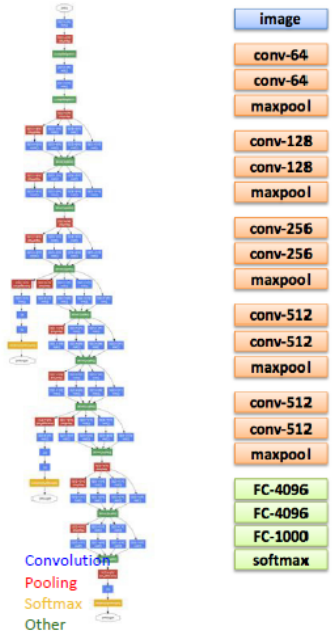


[Krizhevsky NIPS 2012]

7 layers

Year 2014

GoogLeNet VGG

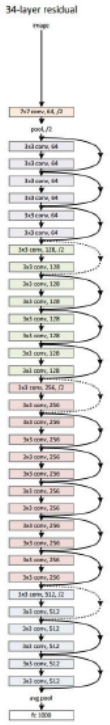


[Szegedy arxiv 2014] [Simonyan arxiv 2014]

27 layers

Year 2015

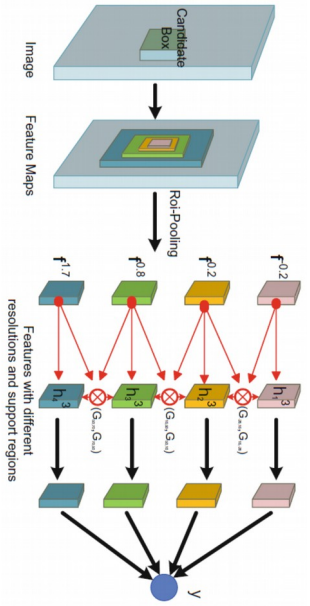
MSRA



Resnet: 151 layers

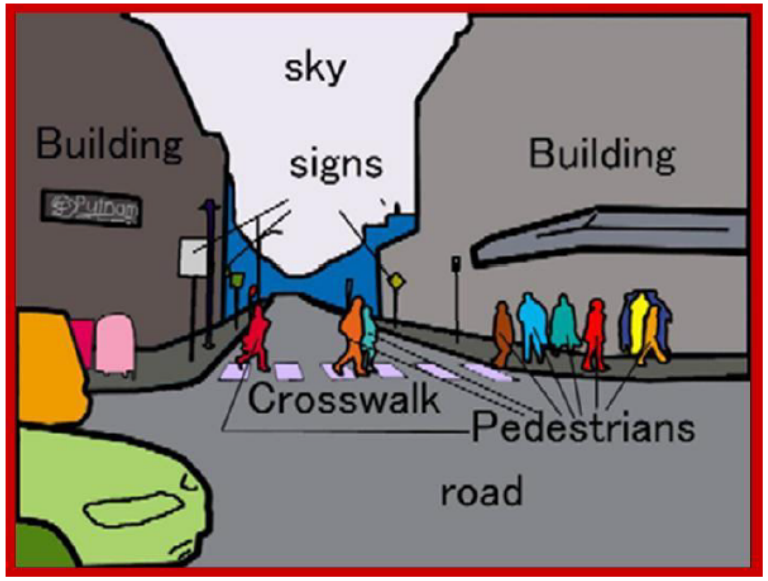
Year 2016

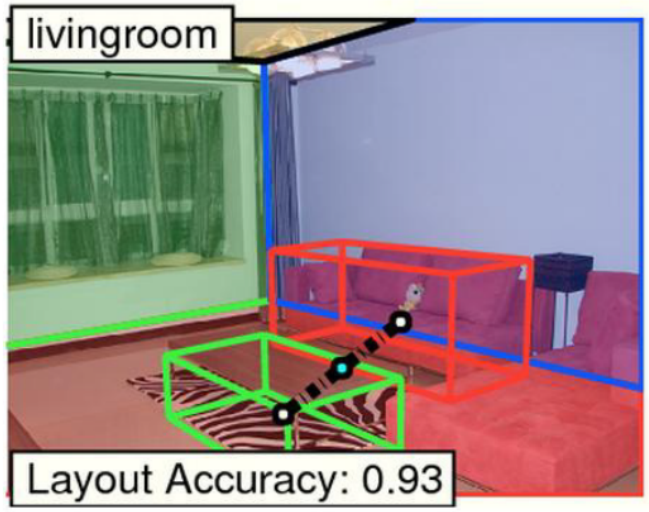
CUImage, Chinese University of Hong Kong

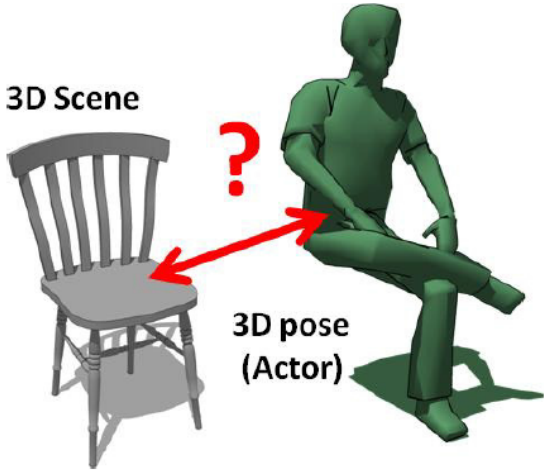


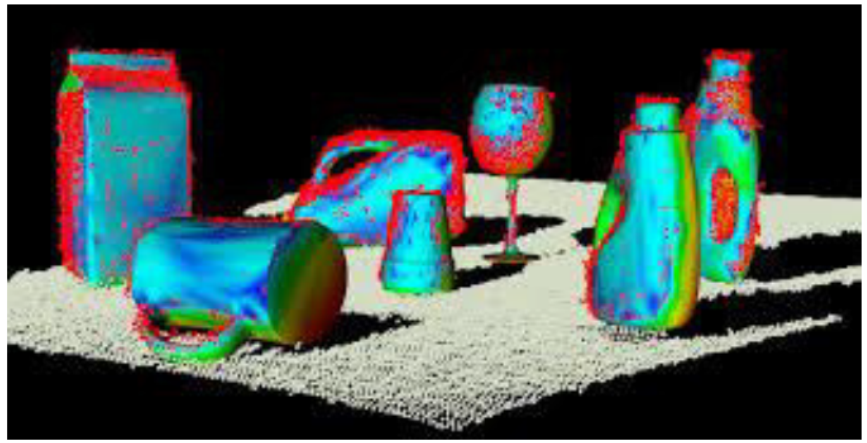
Gated Bi-directional CNN: 269 layers

More difficult vision problems

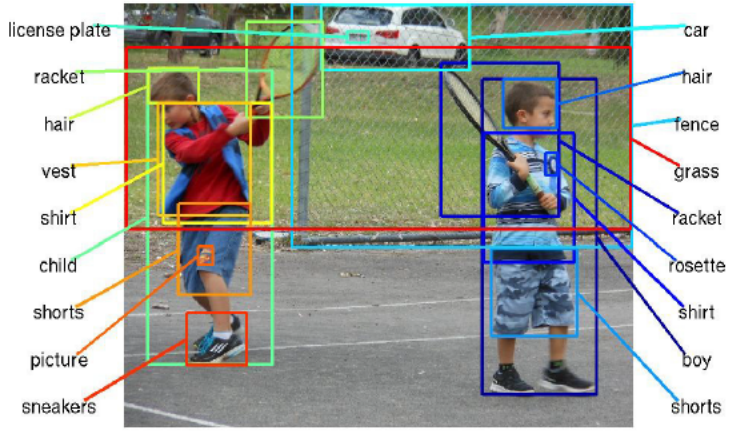


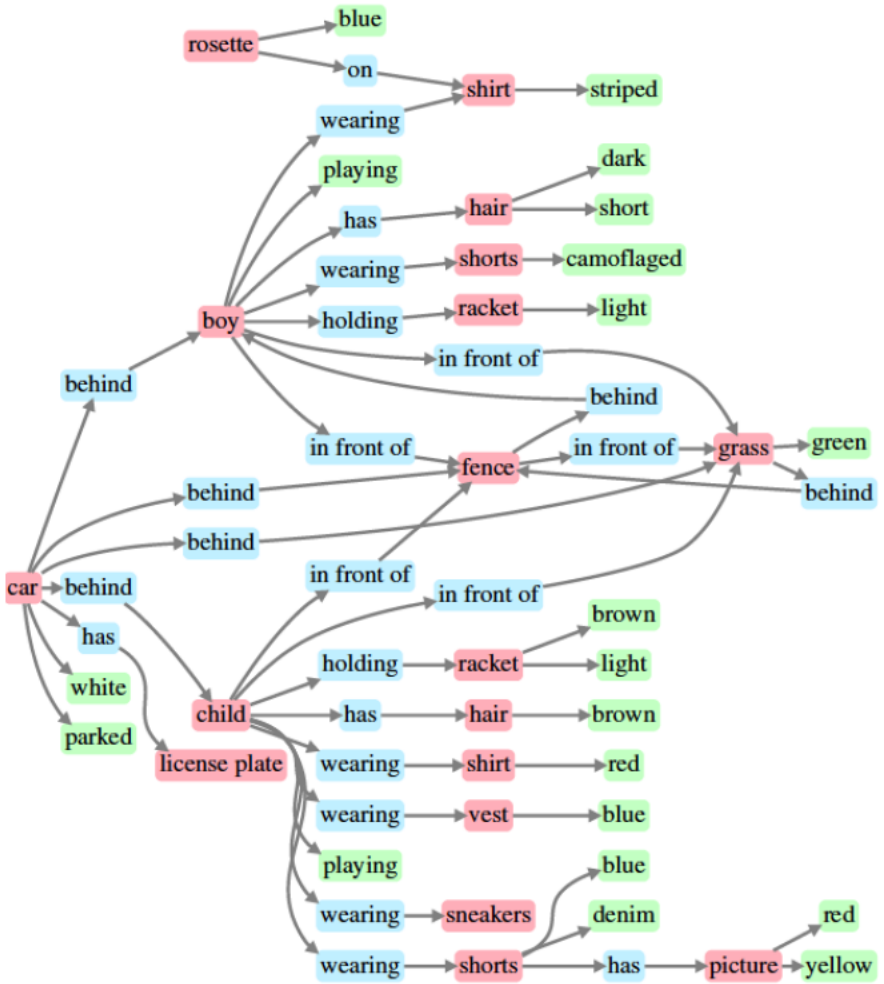






More difficult vision problems





More difficult vision problems

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

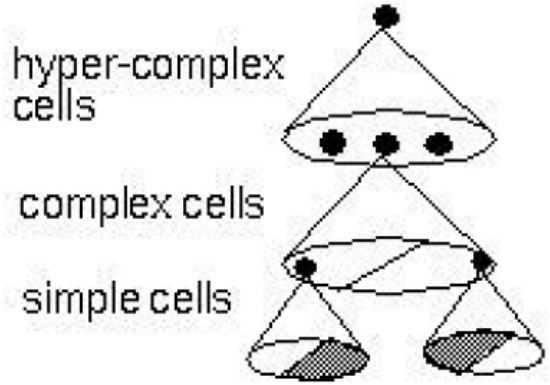
- You recognize it is an image of a bunch of people and you understand they are in a hallway
- You recognize that there are 3 mirrors in the scene so some of those people are “fake” replicas from different viewpoints.
- You recognize Obama from the few pixels that make up his face. It helps that he is 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.

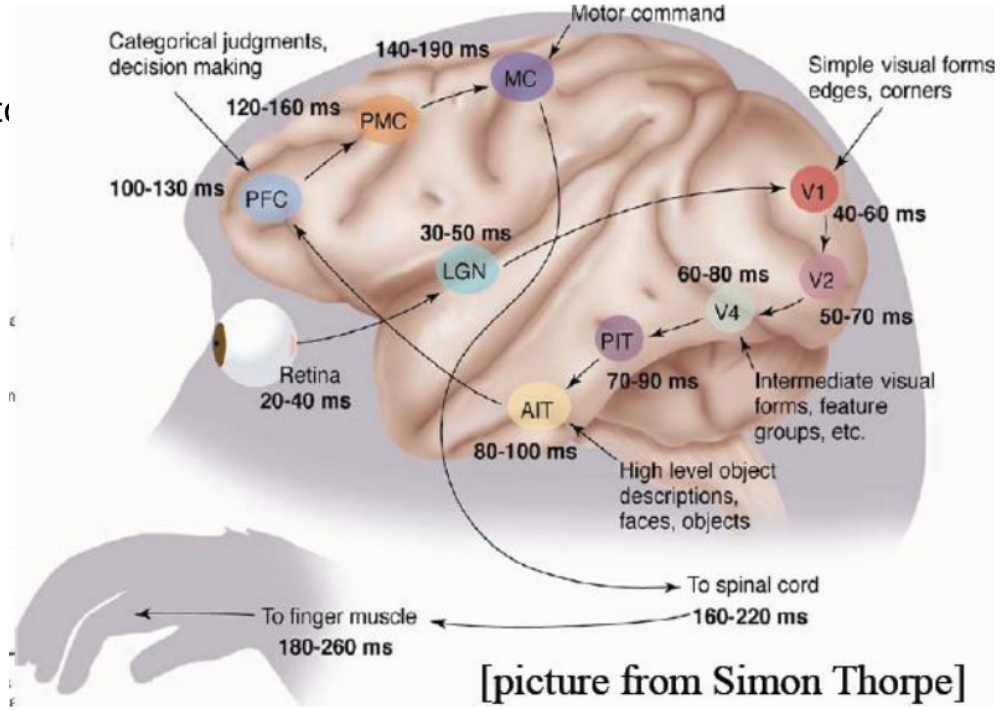
Today’s Agenda

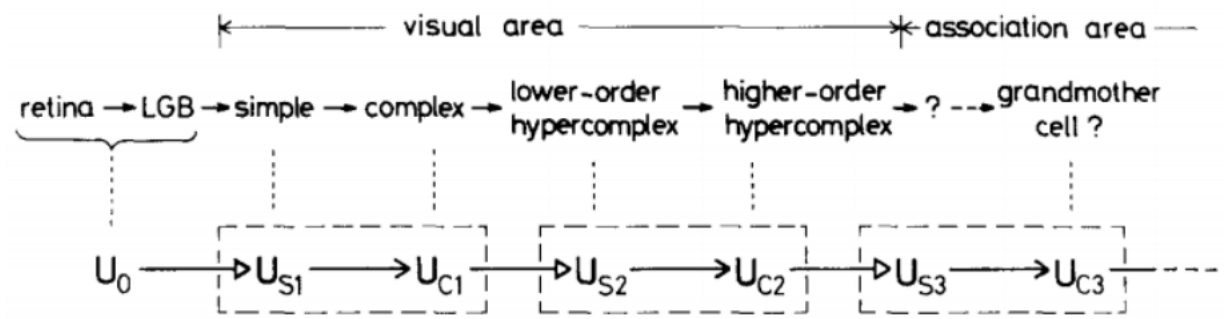
- ImageNet and visual recognition problems
- **Introduction to CNNs and applications**
- Technical details of a CNN
- Deep Semantic Similarity Model (DSSM)

Biological inspiration for CNNs
Topographical mapping in the cortex

- Hubel and Weisel hierarchical mapping [1959]
  - Nearby cells in cortex represented nearby regions in the visual field
  - Visual cortex contains a complex arrangement of cells. These cells are sensitive to small sub-regions of the visual field, called a receptive field. The sub-regions are tiled to cover the entire visual field, and may overlap. These cells act as local filters over the input space and are well-suited to exploit the strong spatially local correlation present in natural images.
  - Additionally, two basic cell types have been identified: Simple cells respond maximally to specific edge-like patterns within their receptive field. Complex cells have larger receptive fields and are locally invariant t the exact position of the pattern.
- The neocognitron is a hierarchical, multilayered artificial neural network proposed by Kunihiko Fukushima in the 1980s.







Hierarchical Approach

VISION

pixels → edge → texton → motif → part → object

SPEECH

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

NLP

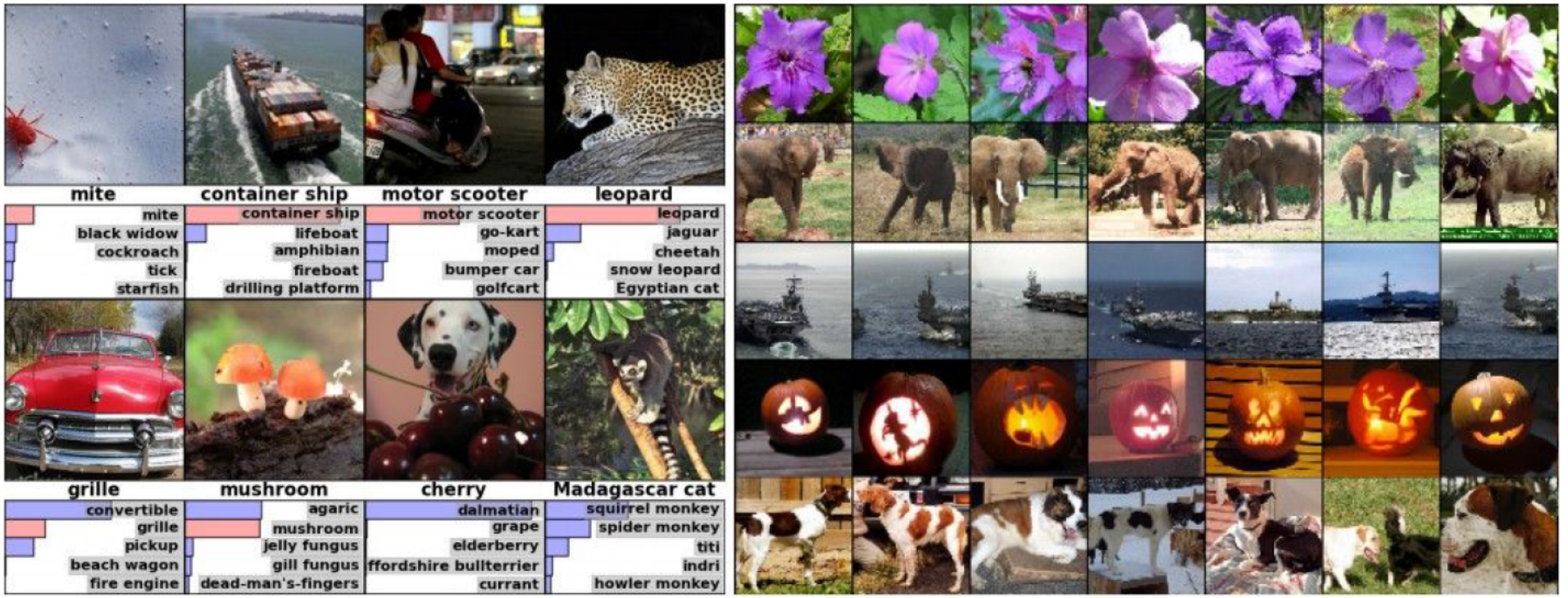
character → word → NP/VP/.. → clause → sentence → story

From: Ranzanto

Convolution Networks are everywhere now

Classification

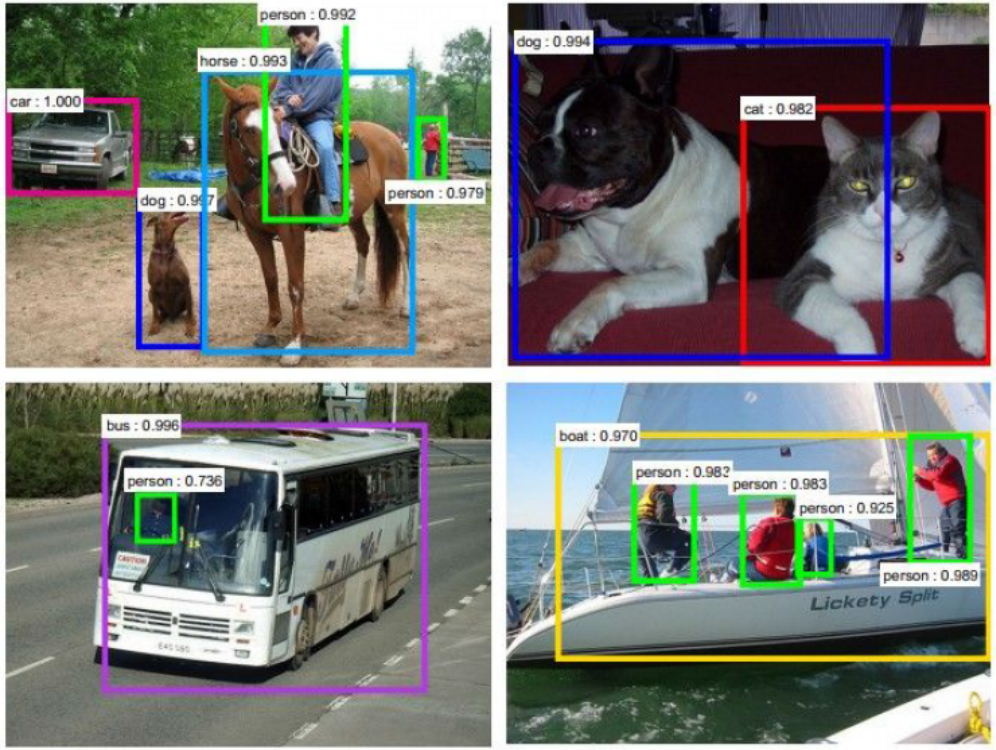
Retrieval



[Krizhevsky 2012]

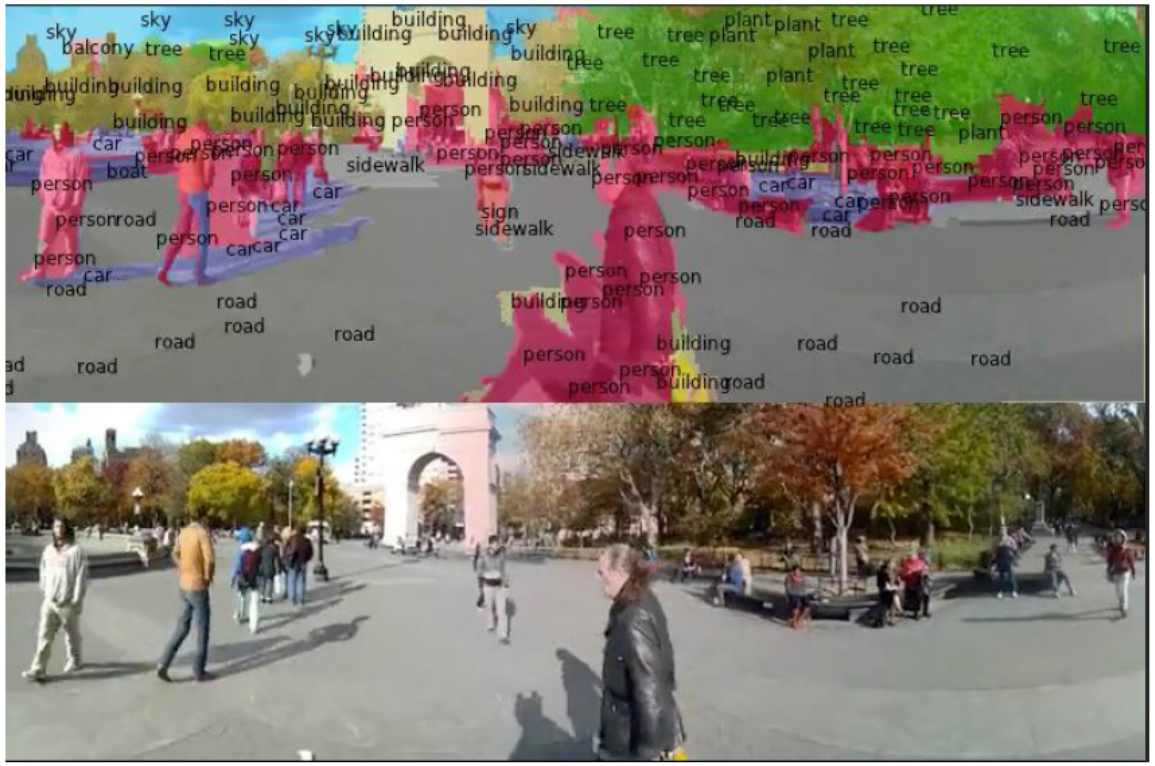
Convolution Networks are everywhere now

Detection



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

Segmentation

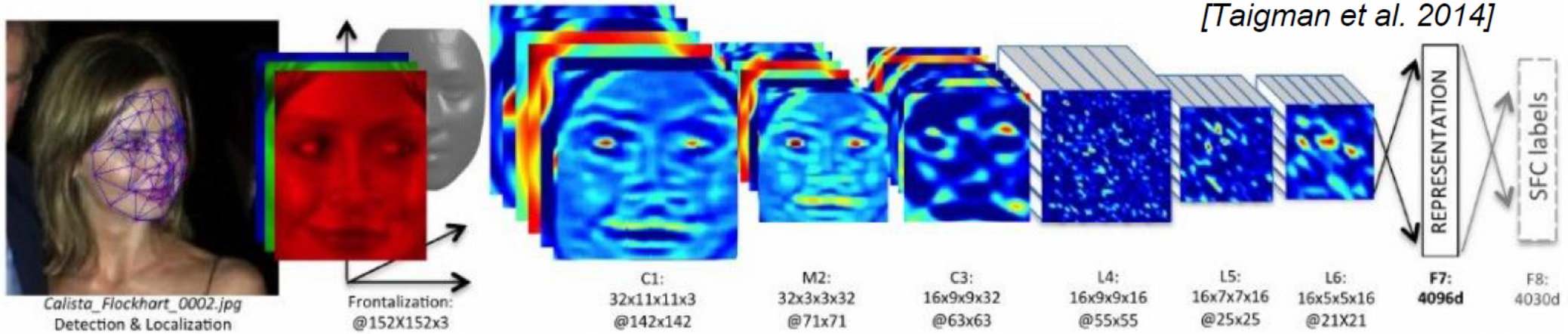


[Farabet et al., 2012]

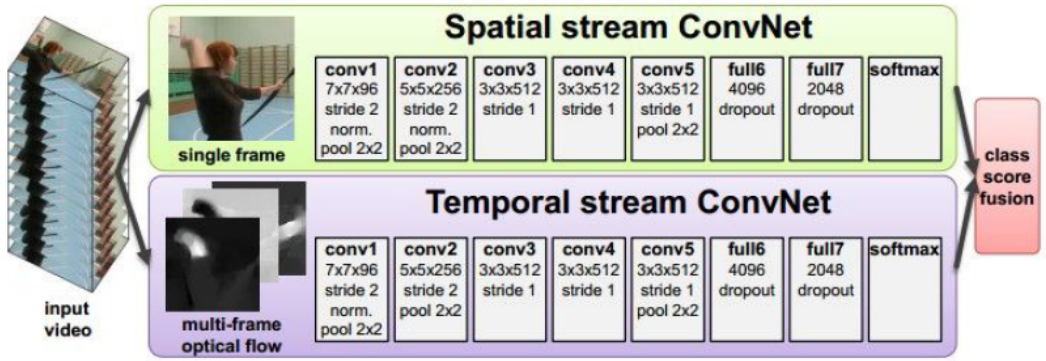
Convolution Networks are everywhere now

self-driving cars

Convolution Networks are everywhere now



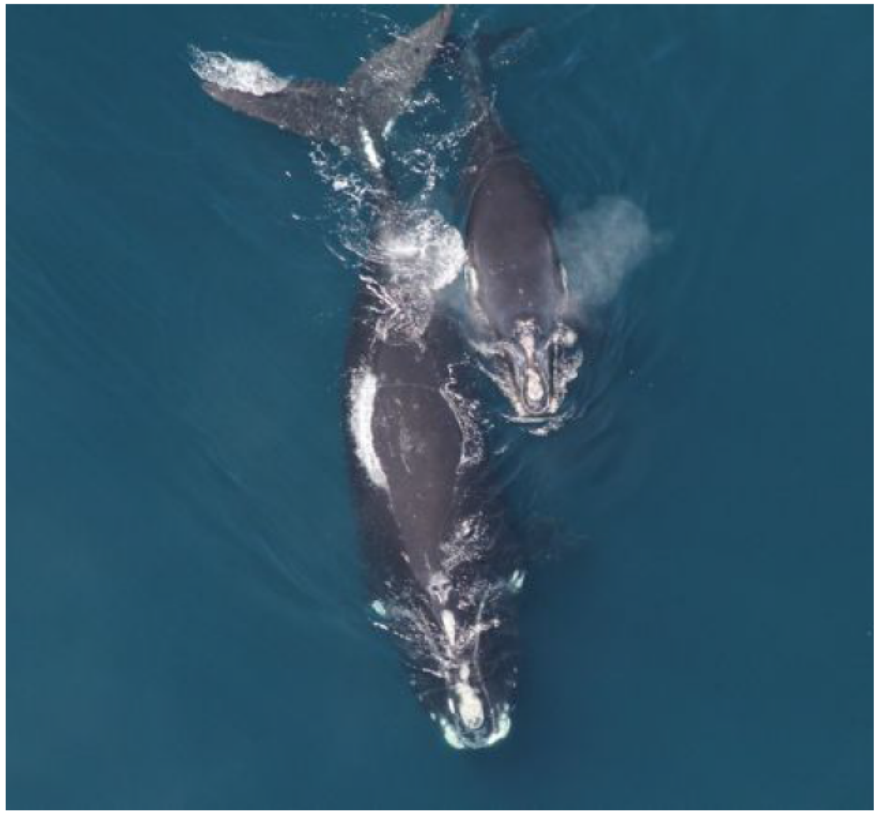
[Taigman et al. 2014]



[Simonyan et al. 2014]

[Goodfellow 2014]

Convolution Networks are everywhere now



Whale recognition, Kaggle Challenge

Mnih and Hinton, 2010

Convolution Networks are everywhere now

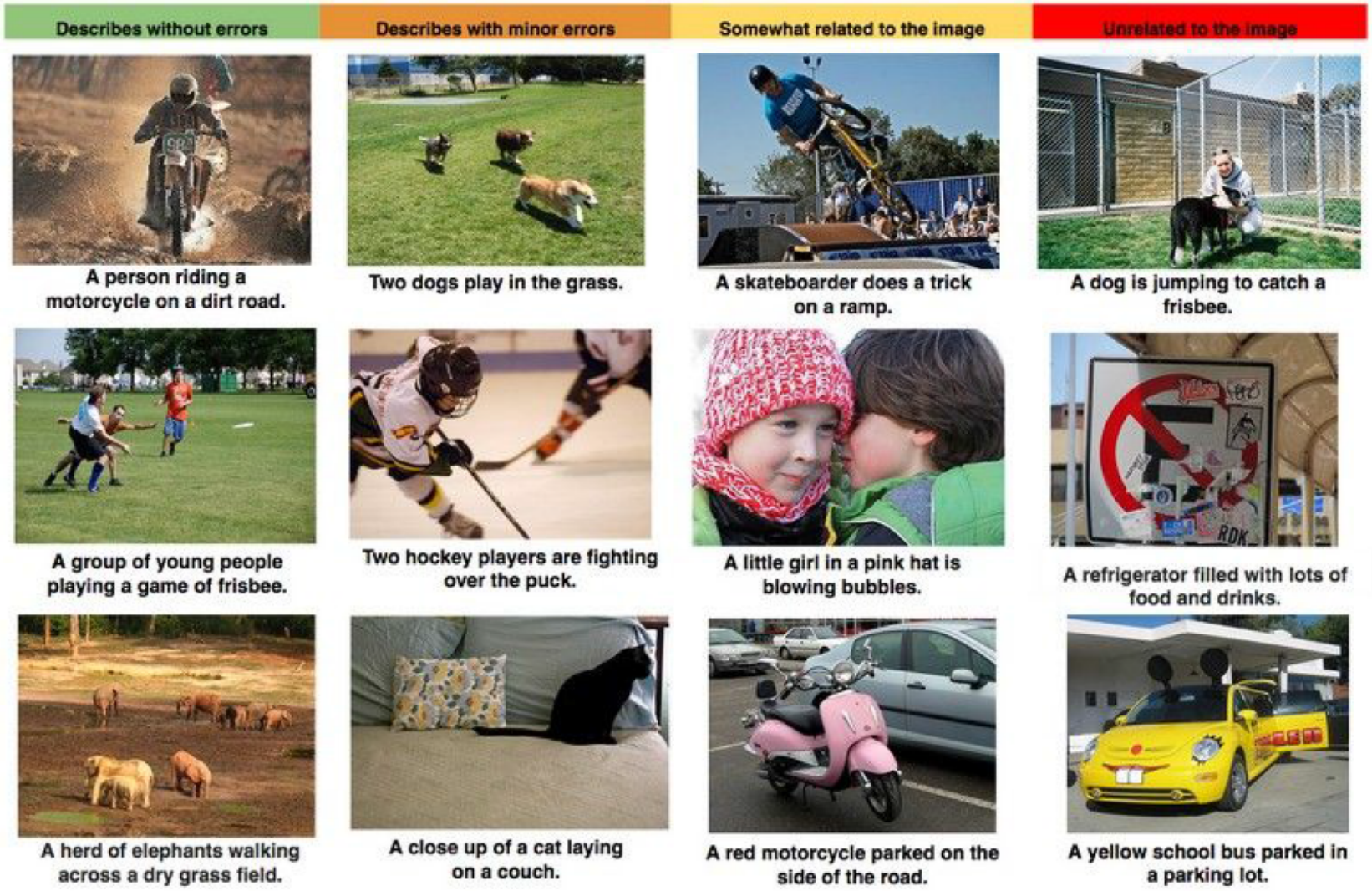


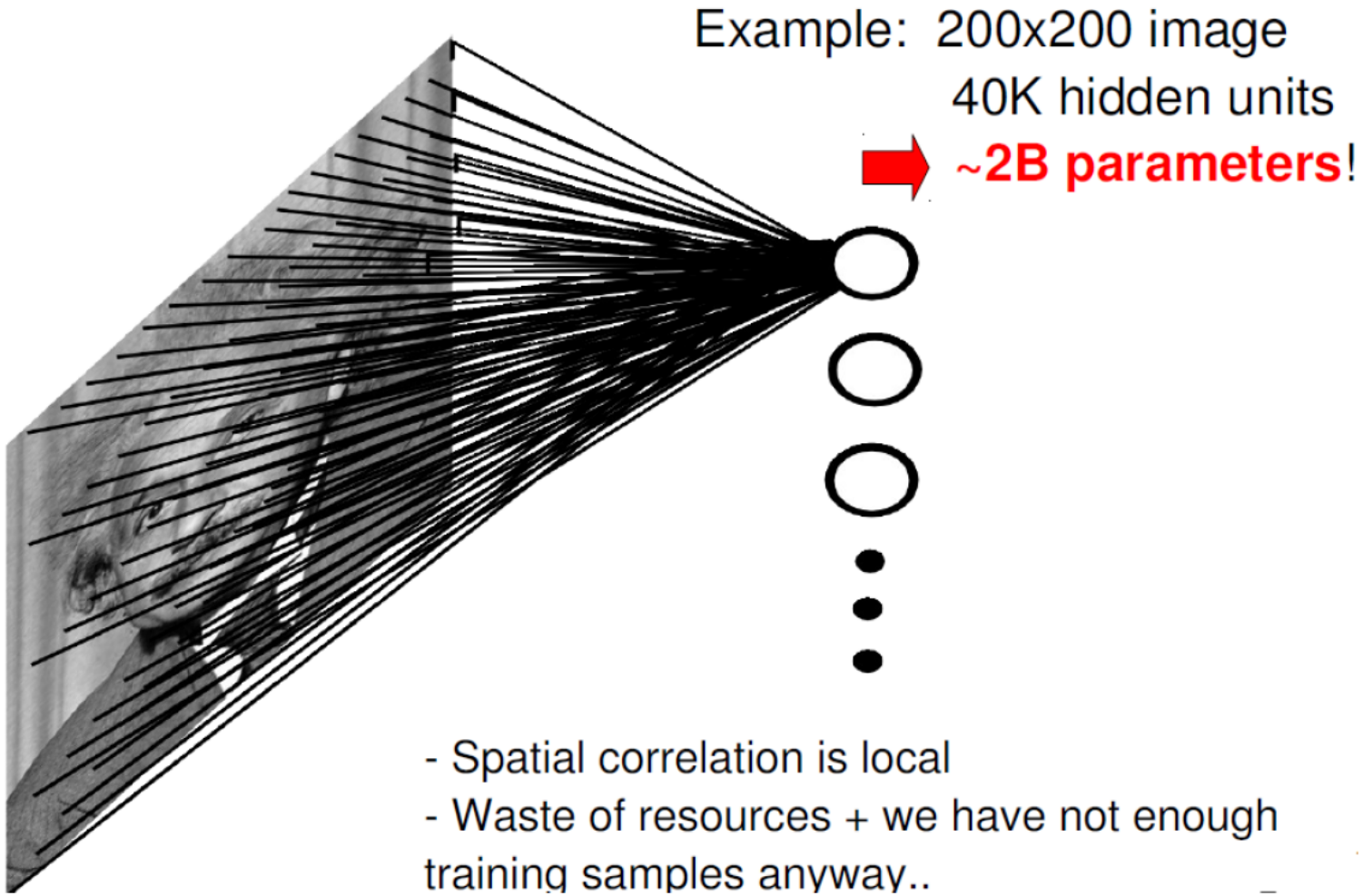
Image Captioning

[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.

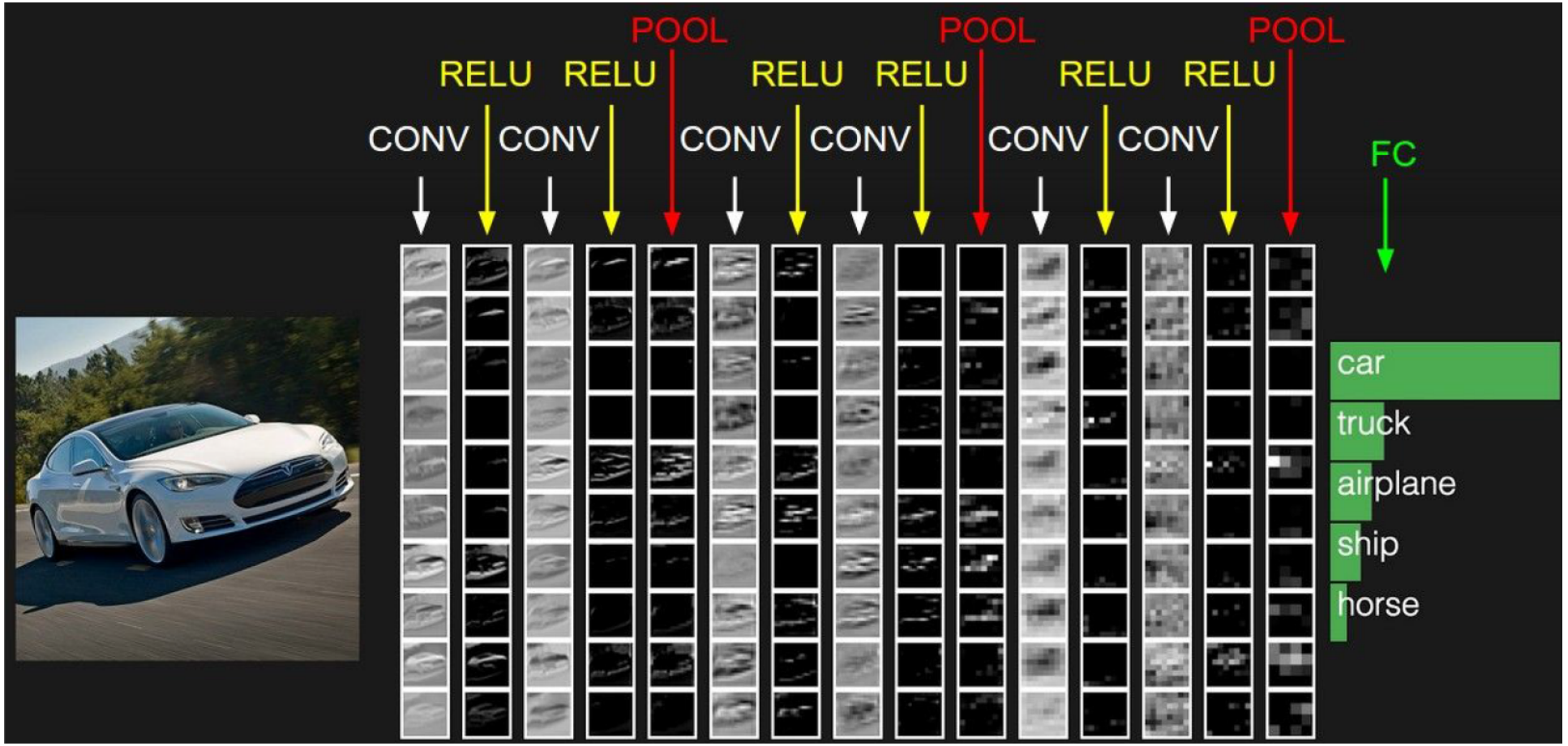
Parameter explosion with MLPs



Today’s Agenda

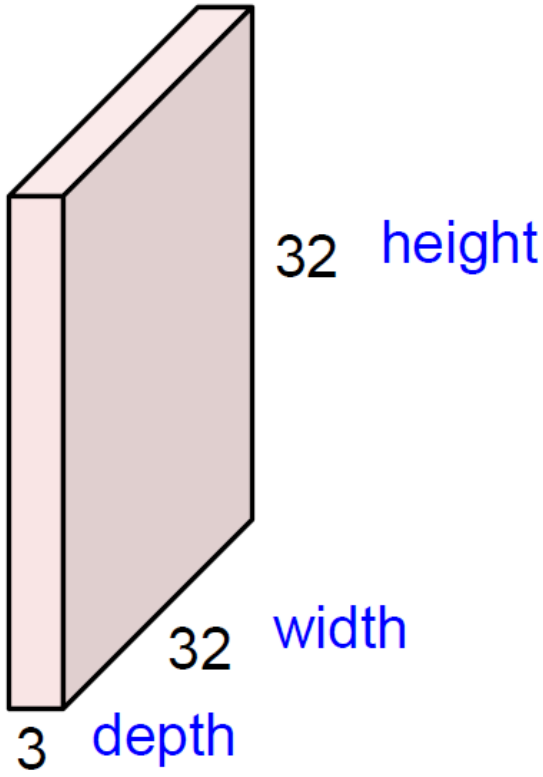
- ImageNet and visual recognition problems
- Introduction to CNNs and applications
- **Technical details of a CNN**
- Deep Semantic Similarity Model (DSSM)

ConvNet: CONV, RELU, POOL and FC Layers



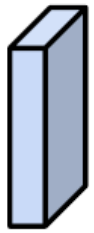
Convolution Layer

32x32x3 image



Filters always extend the full depth of the input volume

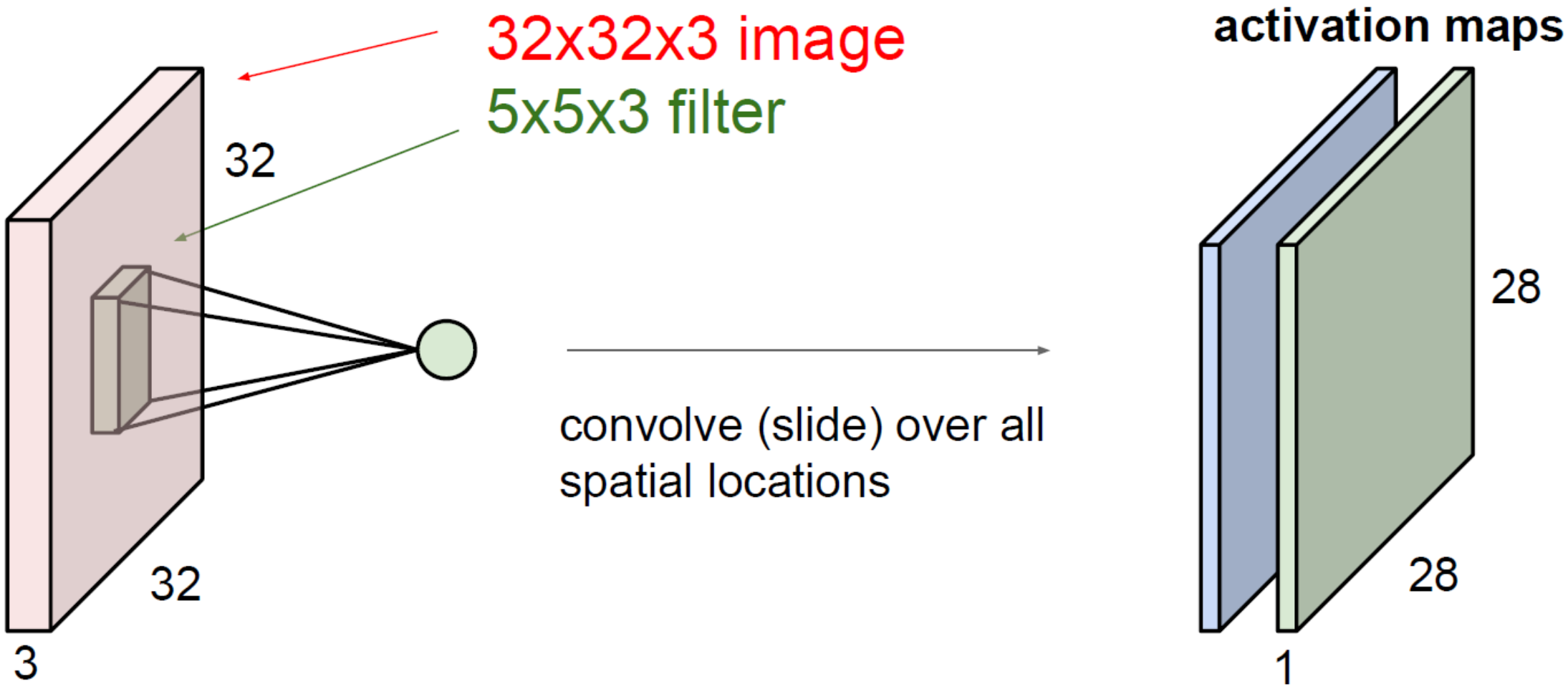
5x5x3 filter



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

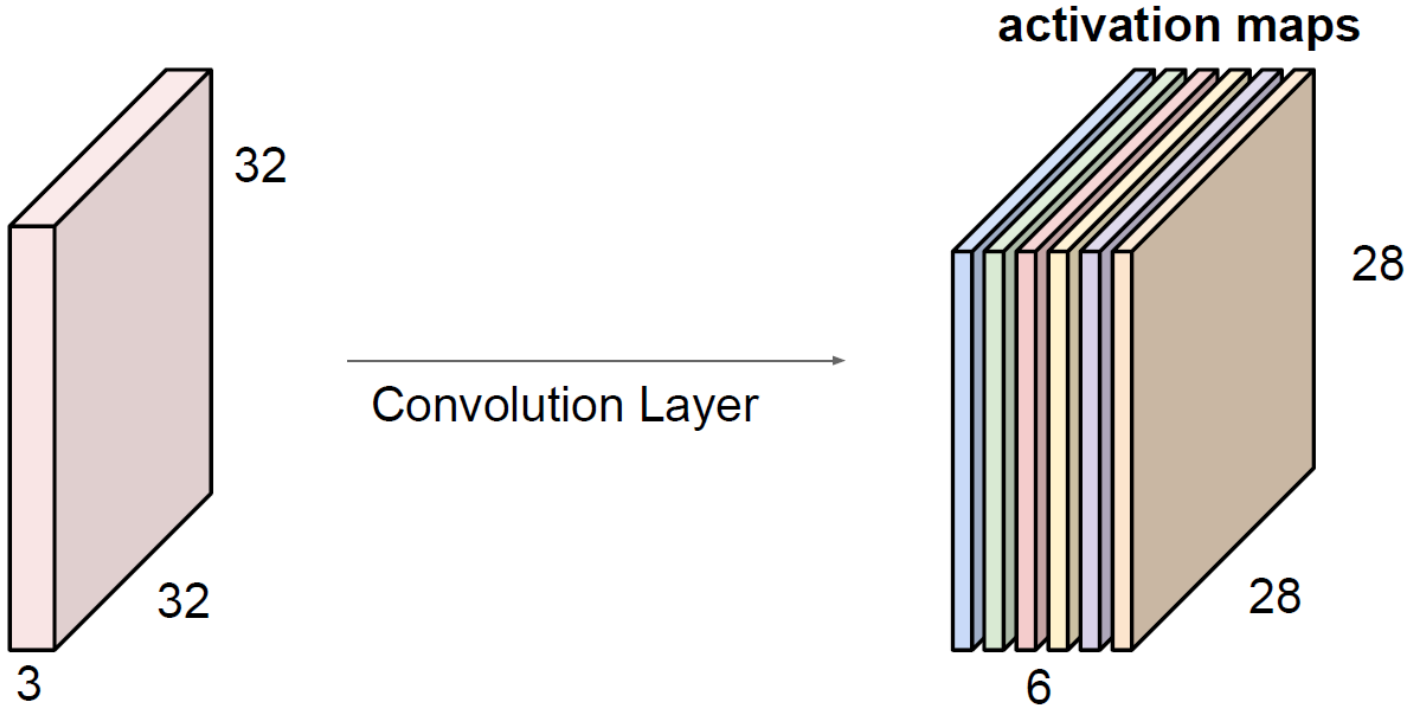
consider a second, green filter

Convolution Layer



Convolution Layer

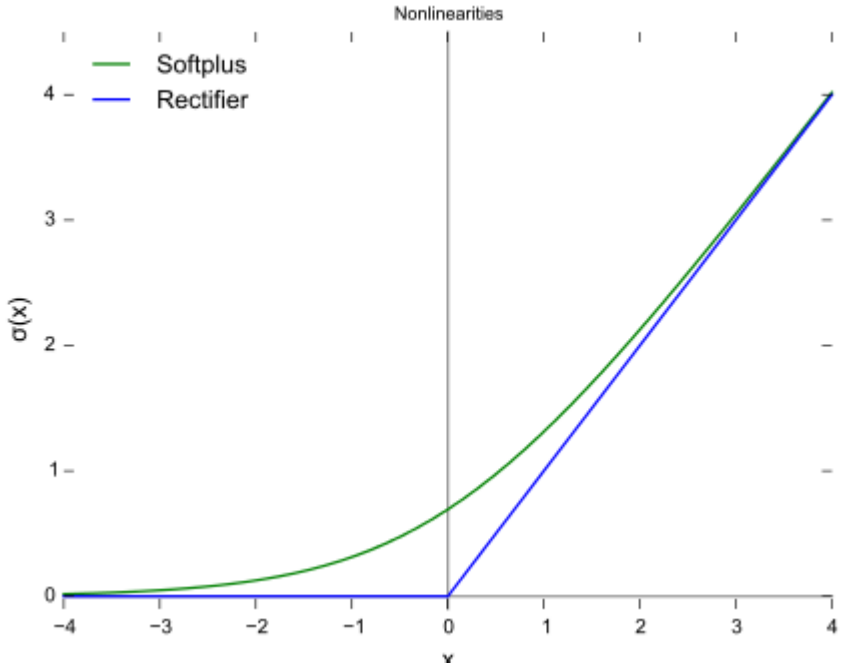
For example, if we had 6 5x5 filters, we'll get 6 separate activation maps:



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

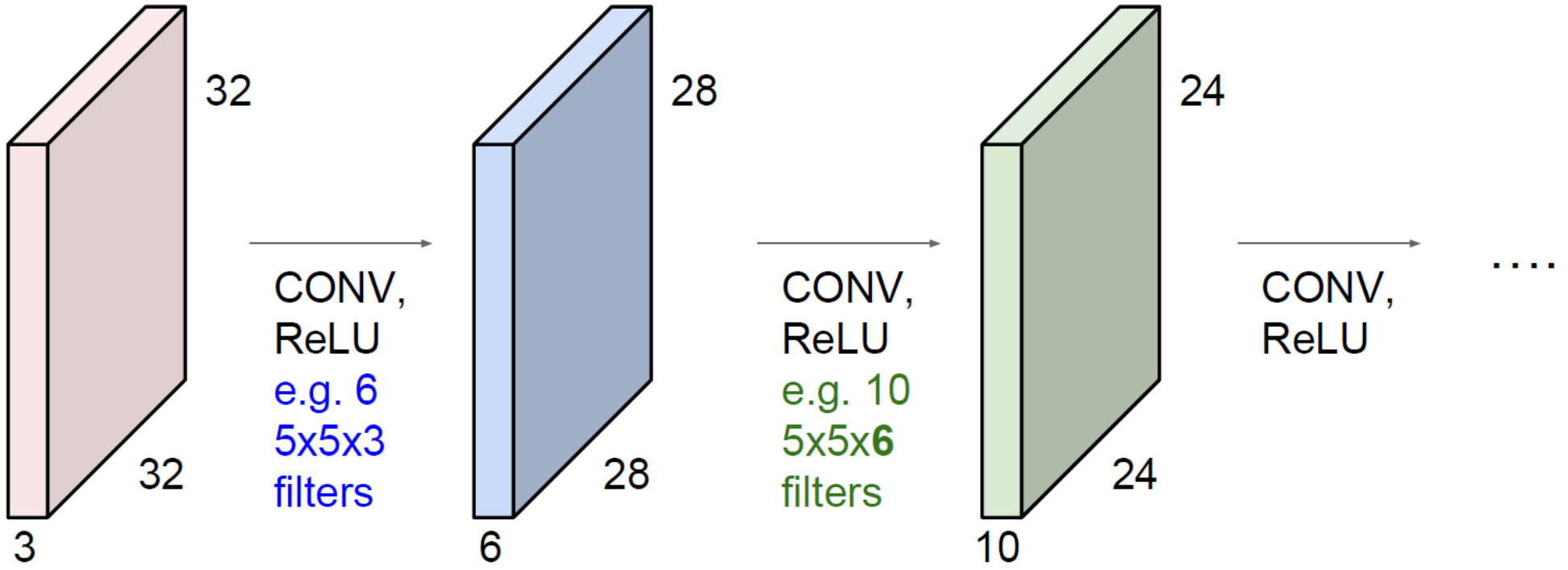
ReLU (Rectified Linear Units) Layer

- This is a layer of neurons that applies the activation function f(x)=max(0,x).
- It increases the nonlinear properties of the decision function and of the overall network without affecting the receptive fields of the convolution layer.
- 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.

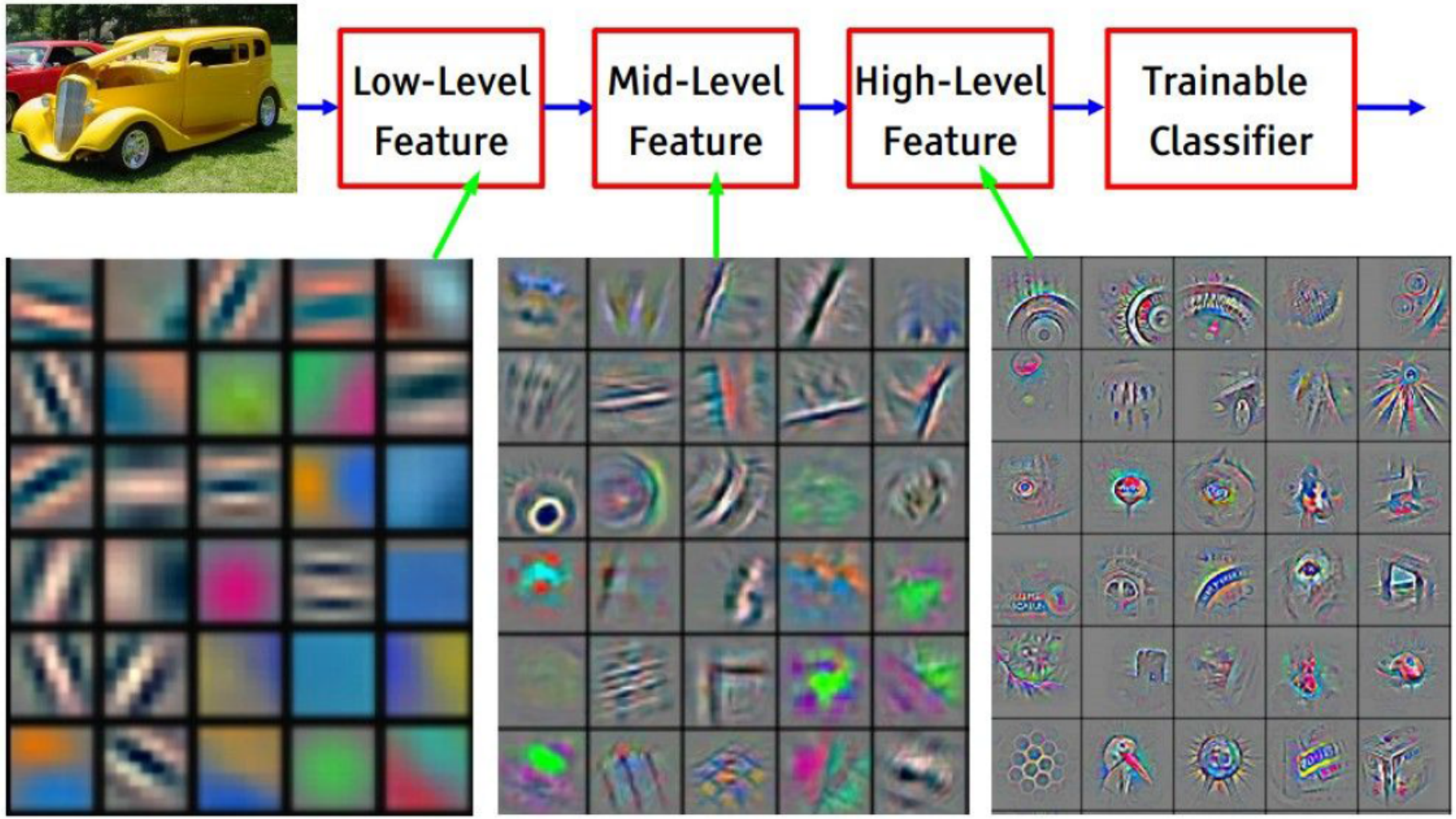


A Basic ConvNet

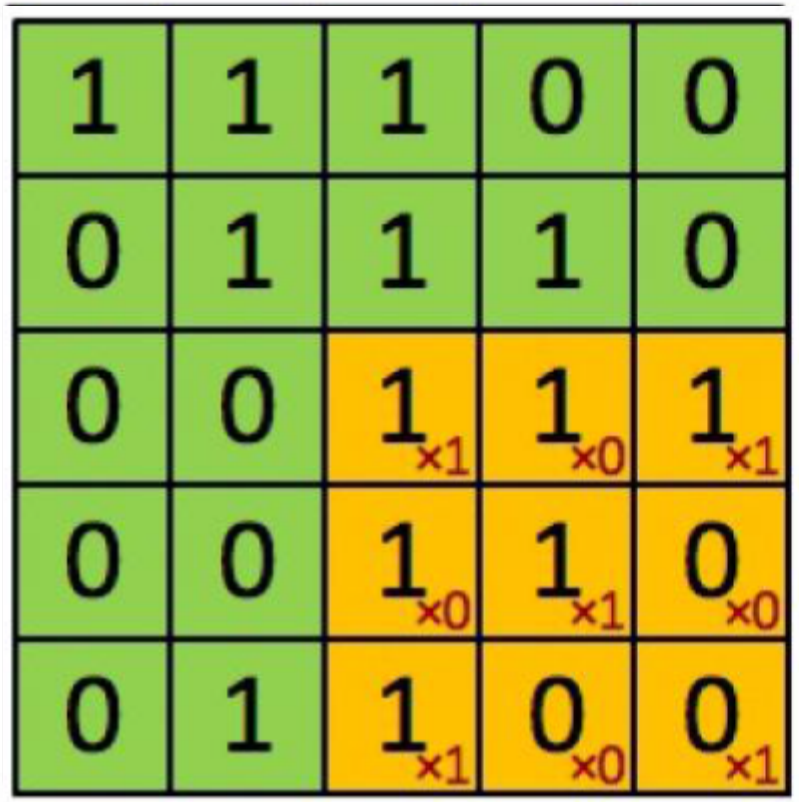
Preview: ConvNet is a sequence of Convolutional Layers, interspersed with activation functions



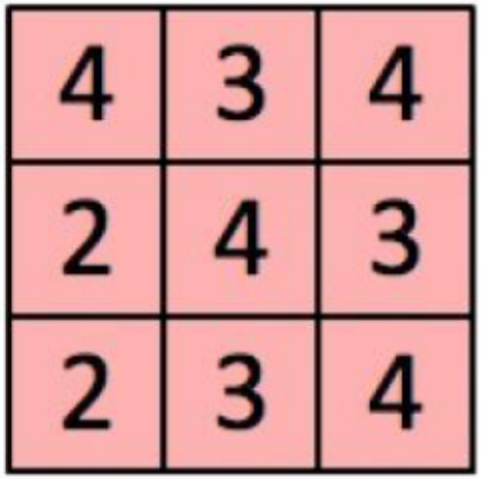
What do those feature/activation maps capture?



What is convolution of an image with a filter?

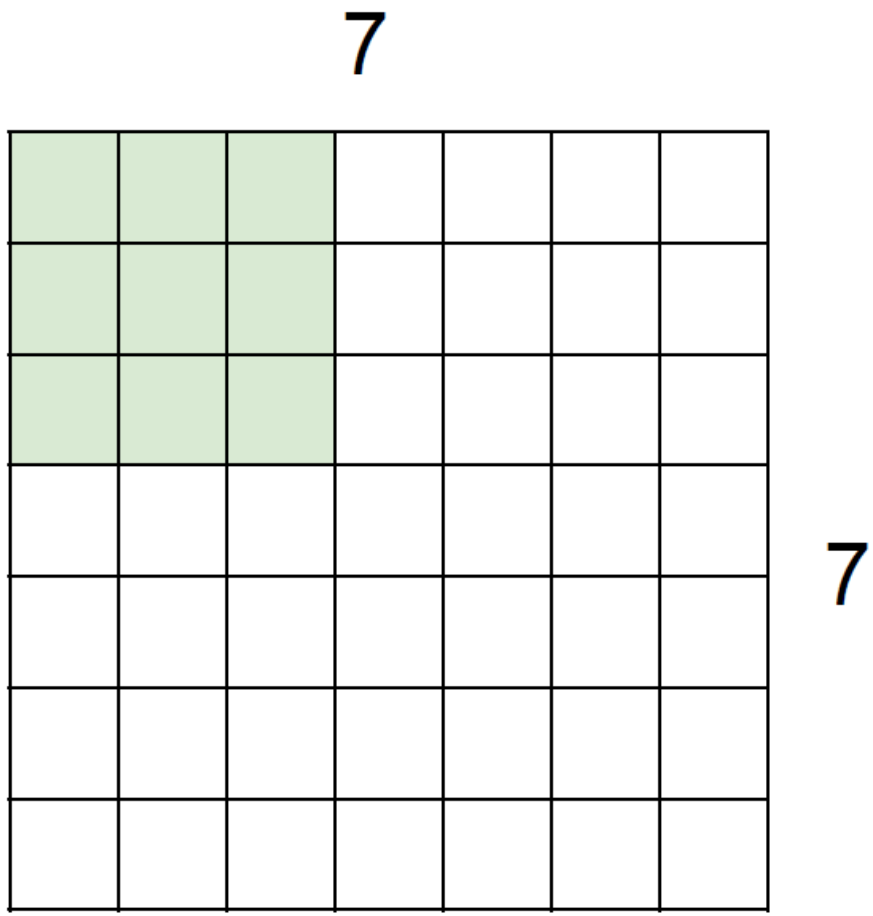


Image



Convolved Feature

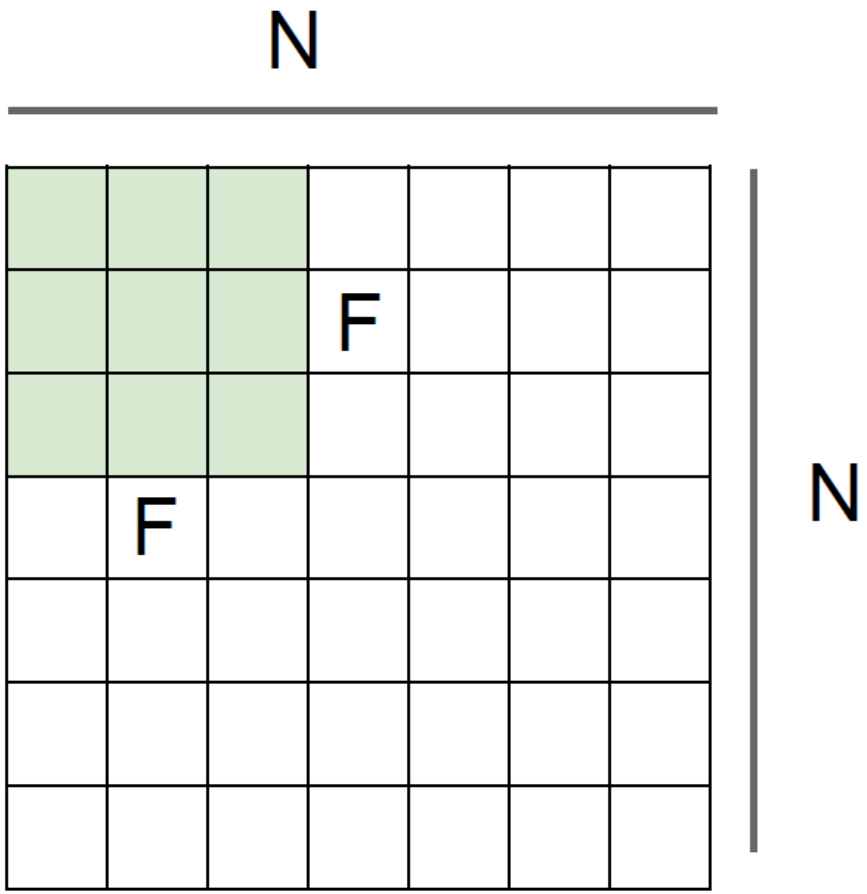
Details about the convolution layer



7x7 input (spatially)
assume 3x3 filter
applied **with stride 3?**

doesn't fit!
cannot apply 3x3 filter on 7x7 input with stride 3.

Details about the convolution layer

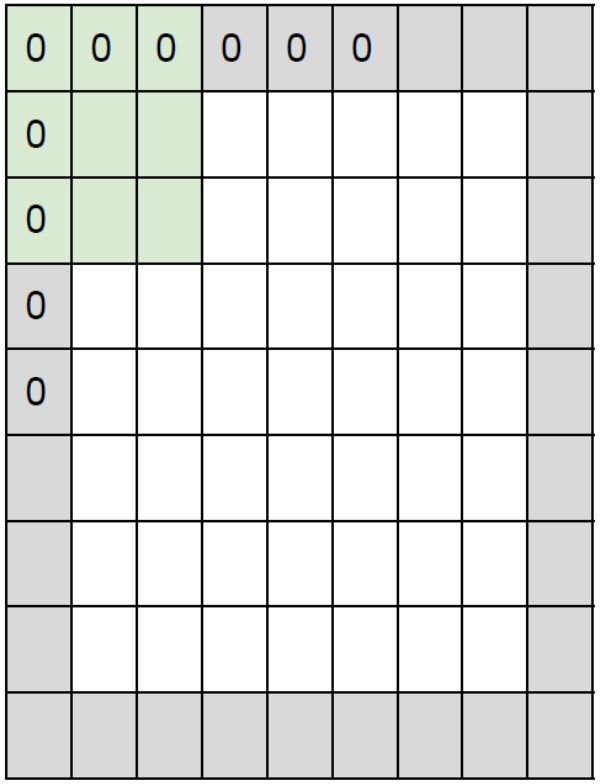


Output size:
(N - F) / stride + 1

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

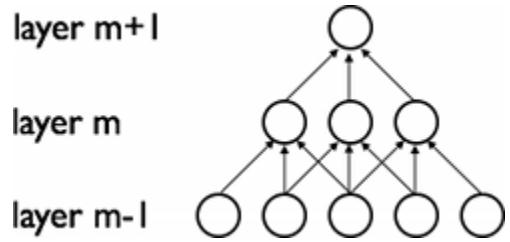
Number of parameters in this layer?

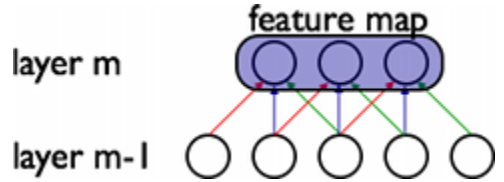
each filter has 5*5*3 + 1 = 76 params (+1 for bias)
=> 76*10 = **760**

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

Sparse Connectivity and Shared Weights

- Sparse Connectivity
  - Enforce a local connectivity pattern between neurons of adjacent layers.
  - The inputs of hidden units in layer m are from a subset of units in layer m-1, units that have spatially contiguous receptive fields.
  - Units in layer m have receptive fields of width 3 in the input retina and are thus only connected to 3 adjacent neurons in the retina layer.
- Shared Weights
  - In CNNs, each filter h_i is replicated across the entire visual field. These replicated units share the same parameterization (weight vector and bias) and form a feature map.
  - Weights of the same color are shared—constrained to be identical. Gradient descent can still be used to learn such shared parameters, with only a small change to the original algorithm. The gradient of a shared weight is simply the sum of the gradients of the parameters being shared.
  - Replicating units in this way allows for features to be detected regardless of their position in the visual field.
  - Weight sharing increases learning efficiency by greatly reducing the number of free parameters being learnt.



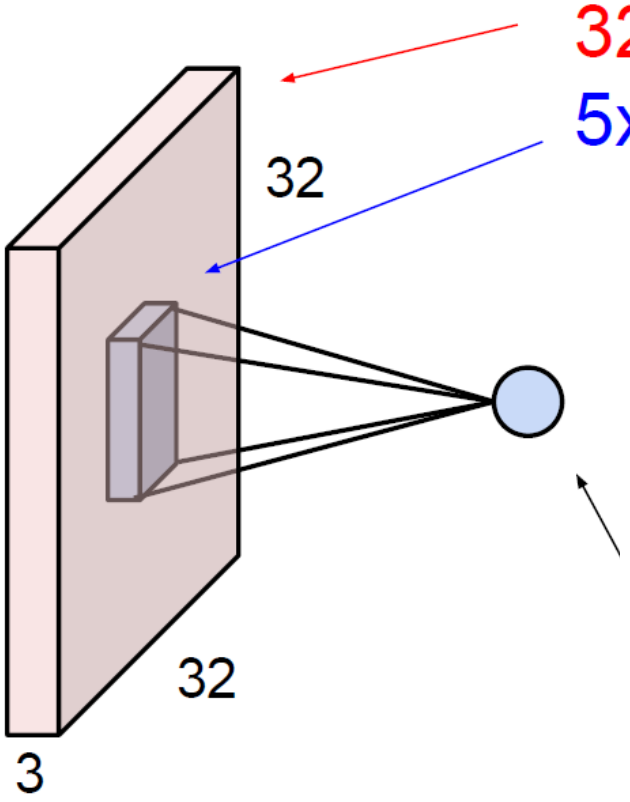


Convolution Layer Summary

Summary. To summarize, the Conv Layer:

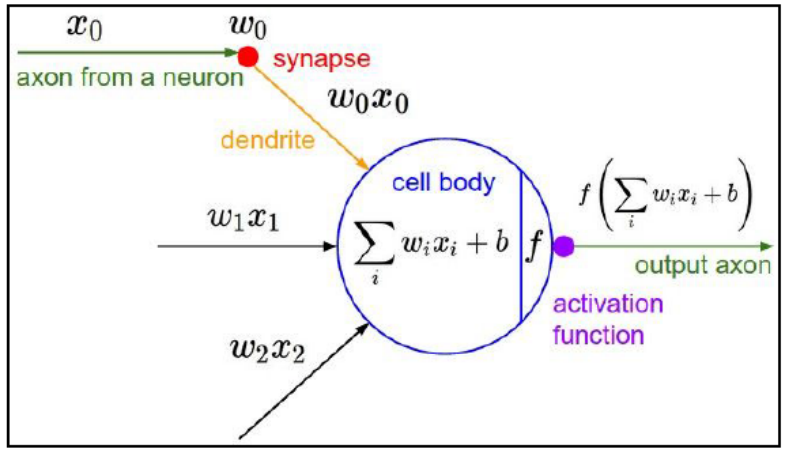
- Accepts a volume of size $W_1 \times H_1 \times D_1$
- Requires four hyperparameters:
  - Number of filters K,
  - their spatial extent F,
  - the stride S,
  - the amount of zero padding P.
- Produces a volume of size $W_2 \times H_2 \times D_2$ where:
  - $W_2 = (W_1 - F + 2P)/S + 1$
  - $H_2 = (H_1 - F + 2P)/S + 1$ (i.e. width and height are computed equally by symmetry)
  - $D_2 = K$
- With parameter sharing, it introduces $F \cdot F \cdot D_1$ weights per filter, for a total of $(F \cdot F \cdot D_1) \cdot K$ weights and K biases.
- In the output volume, the d-th depth slice (of size $W_2 \times H_2$) is the result of performing a valid convolution of the d-th filter over the input volume with a stride of S, and then offset by d-th bias.

Neuron view of the convolution layer



32x32x3 image

5x5x3 filter

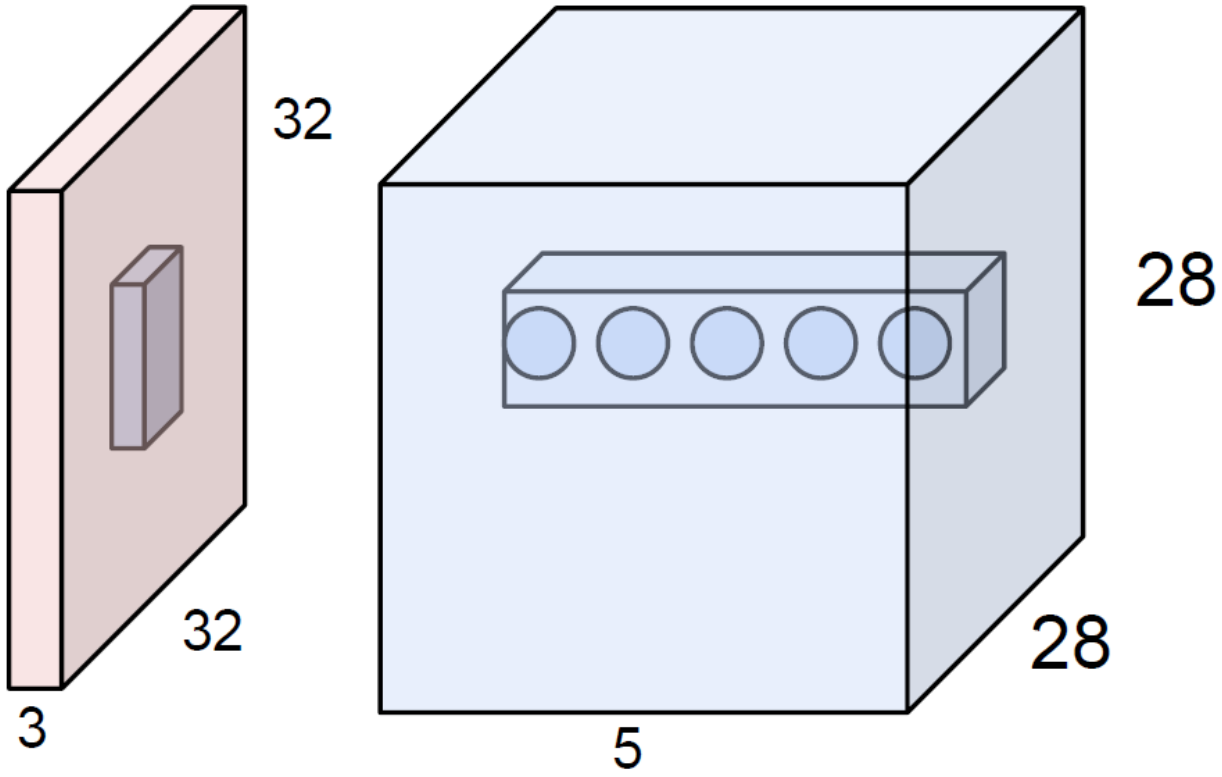


An activation map is a 28x28 sheet of neuron outputs:

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"

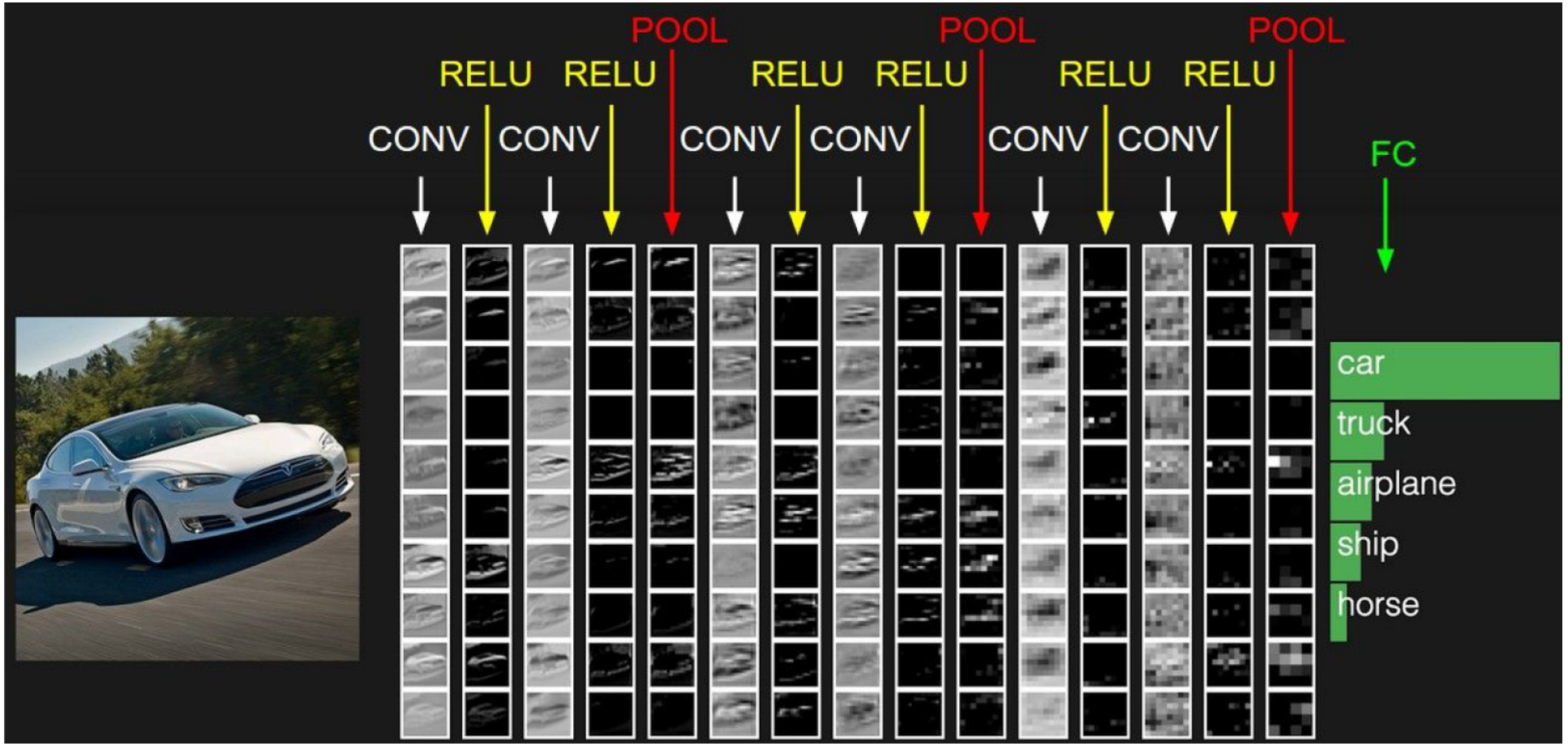
Neuron view of the convolution layer



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

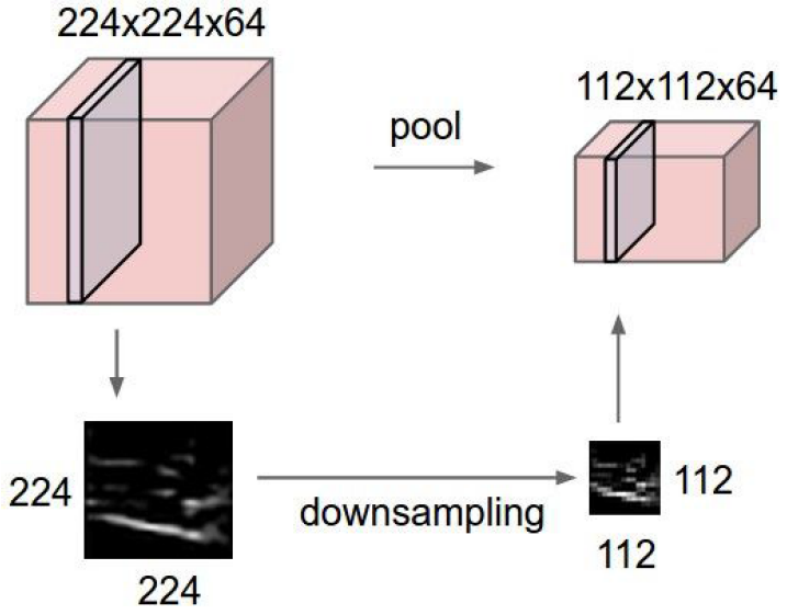
There will be 5 different
neurons all looking at the same
region in the input volume

ConvNet with Pooling and FC Layers



Pooling Layer

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



Single depth slice

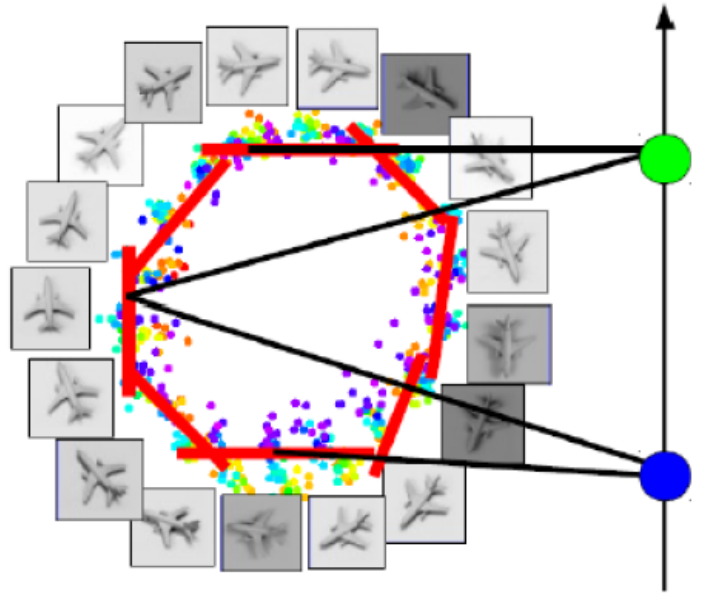
1	1	2	4
5	6	7	8
3	2	1	0
1	2	3	4

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.

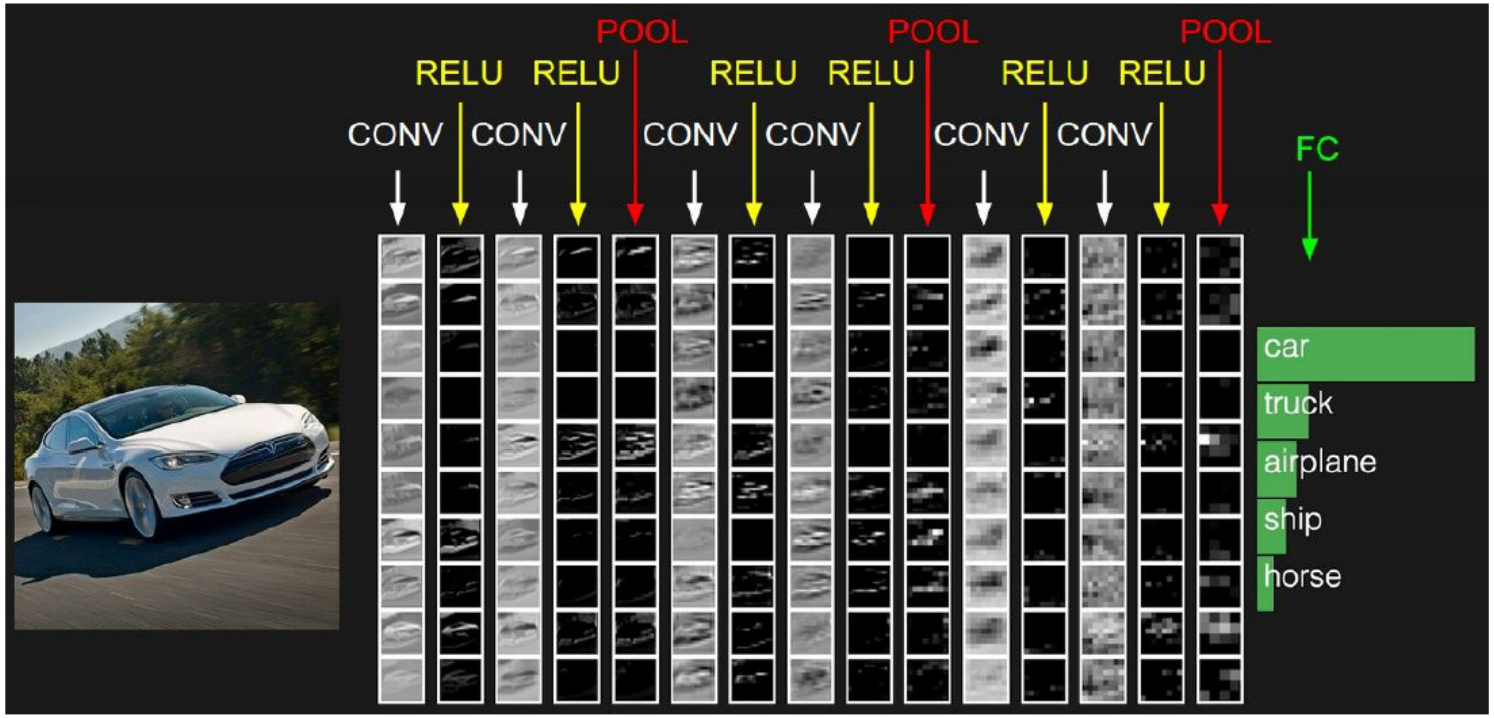
Invariance to local translation can be a very useful property if we care more about whether some feature is present than exactly where it is.

Pooling Layer summary

- Accepts a volume of size $W_1 \times H_1 \times D_1$
- Requires three hyperparameters:
  - their spatial extent F,
  - the stride S,
- Produces a volume of size $W_2 \times H_2 \times D_2$ where:
  - $W_2 = (W_1 - F)/S + 1$
  - $H_2 = (H_1 - F)/S + 1$
  - $D_2 = D_1$
- Introduces zero parameters since it computes a fixed function of the input
- Note that it is not common to use zero-padding for Pooling layers

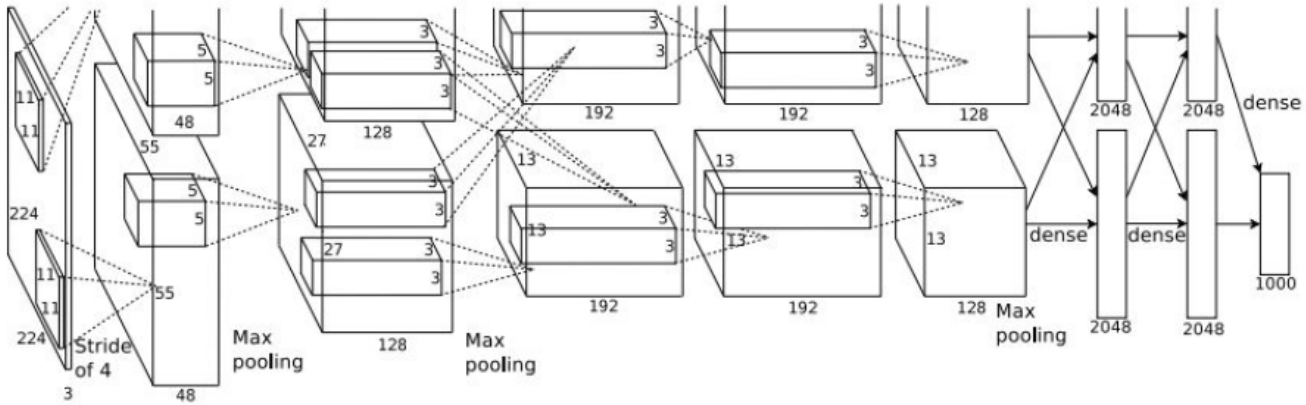
Fully Connected Layer

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%

Today’s Agenda

- ImageNet and visual recognition problems
- Introduction to CNNs and applications
- Technical details of a CNN
- **Deep Semantic Similarity Model (DSSM)**

Deep Semantic Similarity Model (DSSM)

[Huang+ 13; Gao+ 14a; Gao+ 14b; Shen+ 14; Yih+ 15; Fang+15]

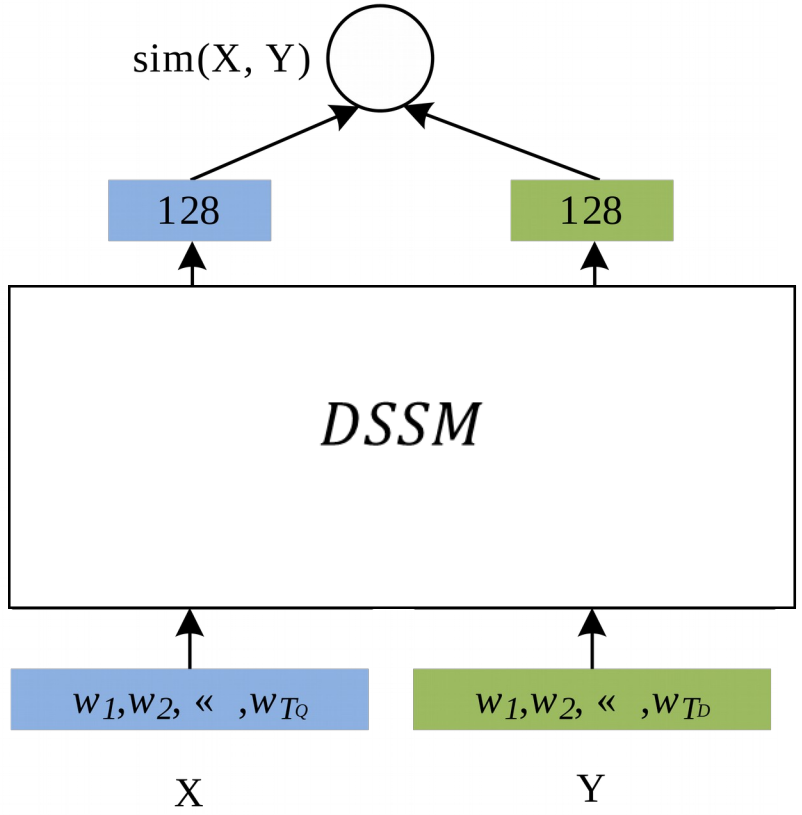
- How to use CNNs to compute semantic similarity between text strings X and Y?
  - Map X and Y to feature vectors in a latent semantic space via deep neural net
  - Compute the cosine similarity between the feature vectors
  - Also called “Deep Structured Similarity Model” in [Huang+ 13]

Tasks	X	Y	Ref
Machine translation	*Text in language A*	*Translation in language B*	[Gao+ 14a]
Web search	*Search query*	*Web document*	[Huang+ 13; Shen+ 14]
Image captioning	*Image*	*Text caption*	[Fang+ 15]
Question Answering	*Question*	*Answer*	[Yih+ 15]
Contextual entity linking	*Mention (in text)*	*Entities (in Satori)*	[Gao+ 14b]
Ad selection	*Search query*	*Ad keywords*	
Sent2Vec (DSSM)	…	…	

http://aka.ms/sent2vec

DSSM: Compute Similarity in Semantic Space

Relevance measured by cosine similarity

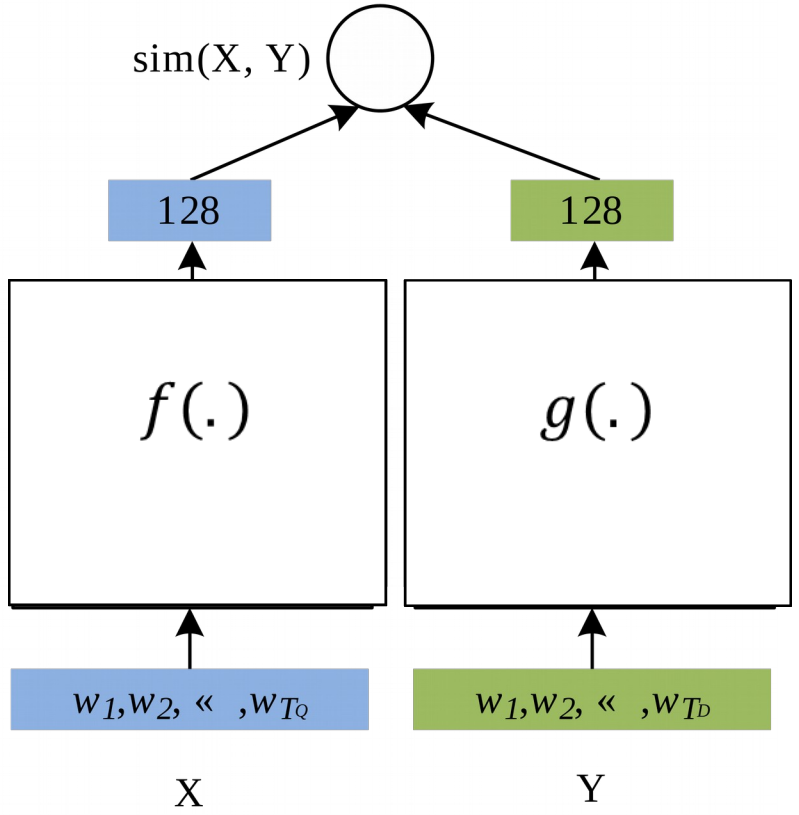


Word sequence x_t

Learning: maximize the similarity between X (source) and Y (target)

DSSM: Compute Similarity in Semantic Space

Relevance measured by cosine similarity



Word sequence x_t

Learning: maximize the similarity between X (source) and Y (target)

Representation: use DNN to extract abstract semantic representations

DSSM: Compute Similarity in Semantic Space

Relevance measured by cosine similarity

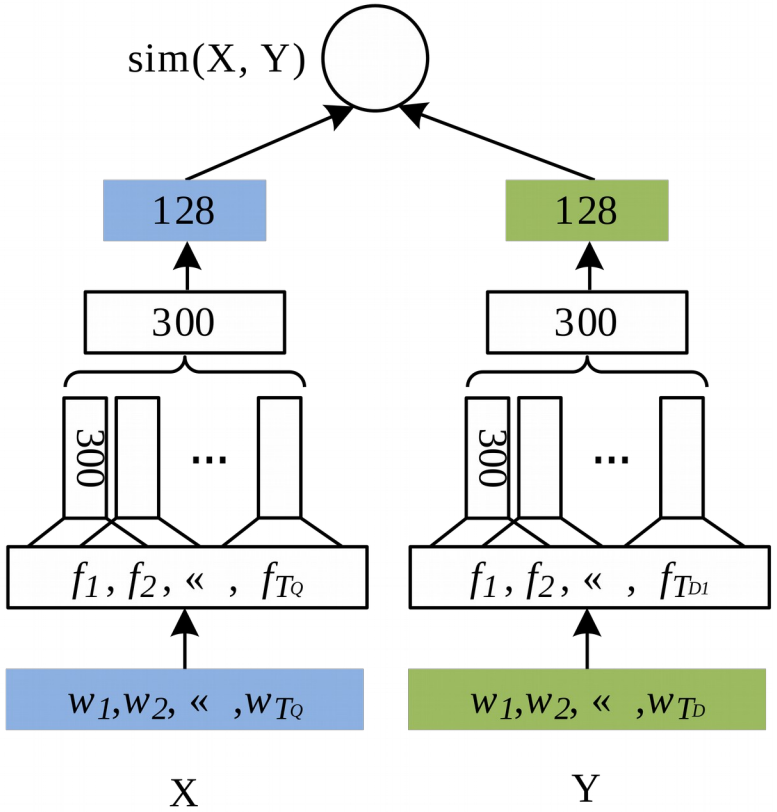
Semantic layer h

Max pooling layer v

Convolutional layer c_t

Word hashing layer f_t

Word sequence x_t



[Gao+ 14b; Shen+ 14]

Learning: maximize the similarity between X (source) and Y (target)

Representation: use DNN to extract abstract semantic representations

Convolutional and Max-pooling layer: identify key words/concepts in X and Y

Word hashing: use sub-word unit (e.g., letter -gram) as raw input to handle very large vocabulary

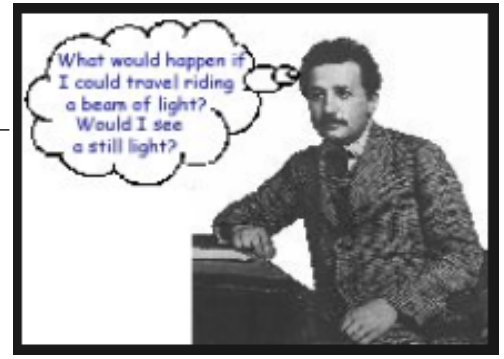
DSSM for Contextual Entity Linking [Gao+ 14b]

The Einstein Theory of Relativity

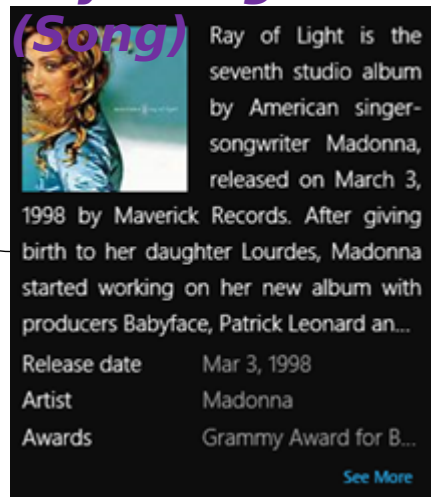
(1) The perihelion of Mercury shows a discrepancy which has long puzzled astronomers. This discrepancy is fully accounted for by Einstein. At the time when he published his theory, this was its only experimental verification.

(2) Modern physicists were willing to suppose that light might be subject to gravitation—i.e., that a ray of light passing near a great mass like the sun might be deflected to the extent to which a particle moving with the same velocity would be deflected according to the orthodox theory of gravitation. But Einstein's theory required that the light should be deflected just twice as much as this. The matter could only be tested during an eclipse among a number of bright stars. Fortunately a peculiarly favour-able eclipse occurred last year. The results of the observations

Ray of Light



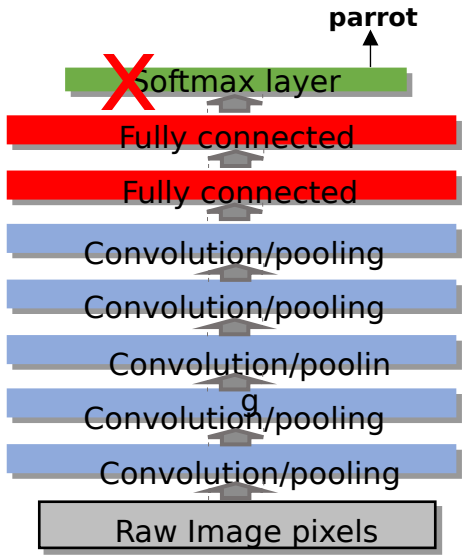
Ray of Light (Song)



Multimodal DSSM for Image-Text Joint Learning

[Fang+15]

- DSSM for text inputs: *s, t*
- Replace text s by image **s**
- Pick complete captions affinitize to complete images



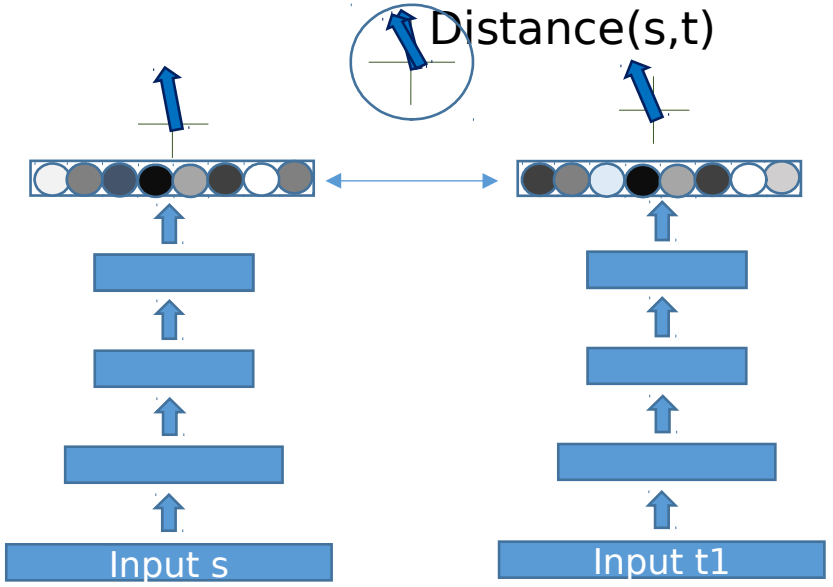


Image features s

Text: *a parrot riding a tricycle*

Q = image, D = caption, R = relevance

Relevance: $R(Q,D) = \text{cosine}(y_Q, y_D) = \frac{y_Q{}^T y_D}{\|y_Q\|\|y_D\|}$

Caption probability: $P(D|Q) = \frac{\exp(\gamma R(Q,D))}{\Sigma_{D' \in \mathbb{D}} \exp(\gamma R(Q,D'))}$

Candidate captions ↗ ↖ Smoothing factor

Objective: $L(\Lambda) = -\log \prod_{(Q,D^+)} P(D^+|Q)$

↖ Correct caption

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.
- Finally, we talked about DSSMs which use CNNs for learning similarity between object pairs.
- CNN feature visualization: http://people.csail.mit.edu/torralba/research/drawCNN/drawNet.html?path=imagenetCNN