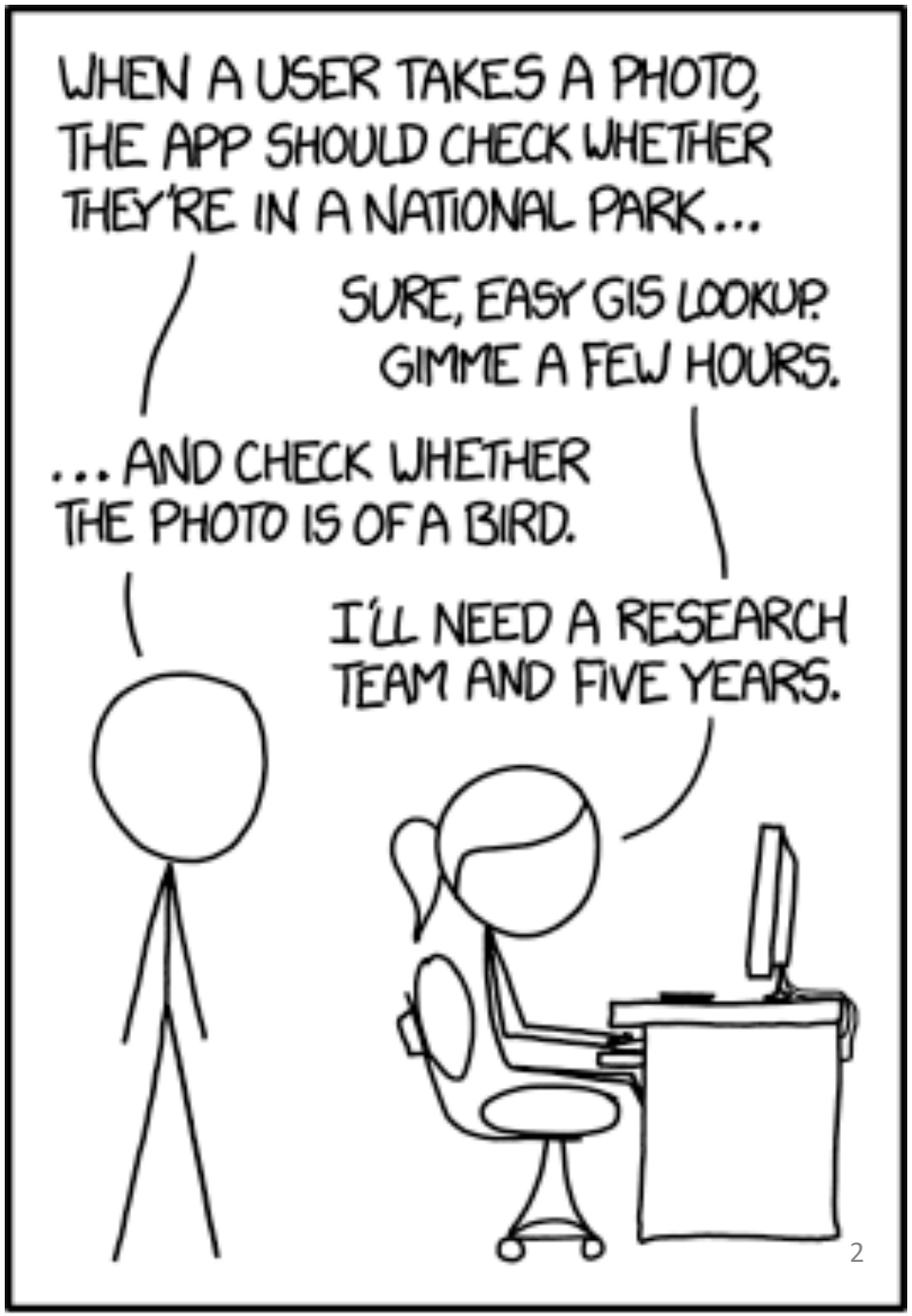
Neural Networks MLPs and CNNs

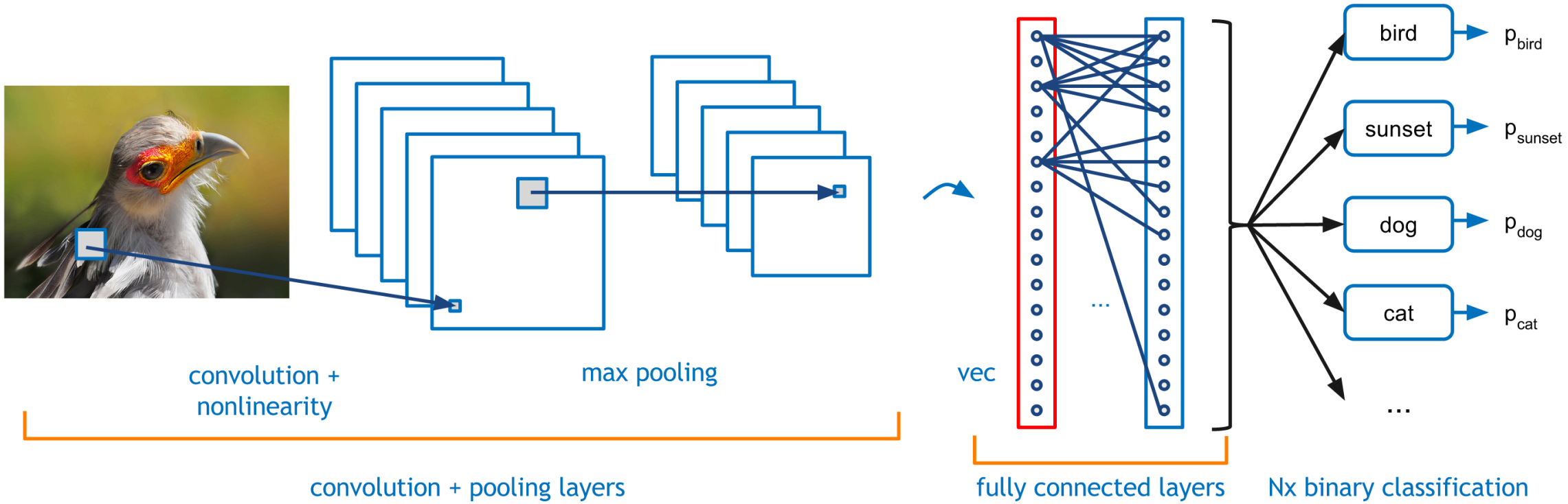
Ardavan Pedram

It can be hard to explain the difference between the easy and the virtually impossible.



https://xkcd.com/1425/

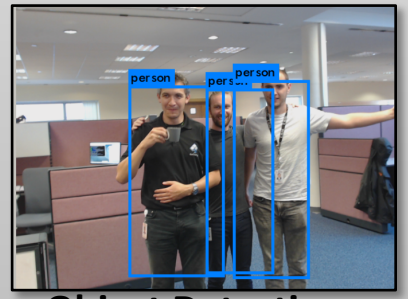
High Level view



Why Neural Networks

- **Conventional Computers**
  - Deductive Reasoning: Analyse the problem in terms of known rules
  - Code the analysis
  - Good for well defined problems
- **Neural Networks Learn From Examples**
  - Inductive Reasoning: Construct the rules
  - **Generalize to handle incomplete data**
  - No requirement for an explicit description of the problem
  - **Noise Resilient**

Image Classification



Object Detection

Semantic Segmentation

Computer Vision
CNNs

Speaker Diarization

Speech Recognition

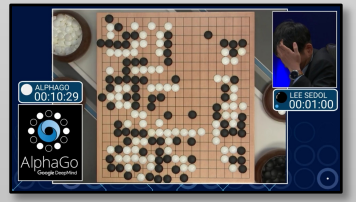
Speech Recognition
RNNs, LSTMs

Translation

Sentiment Analysis

Natural Language Processing
Sequence to sequence

Recommender

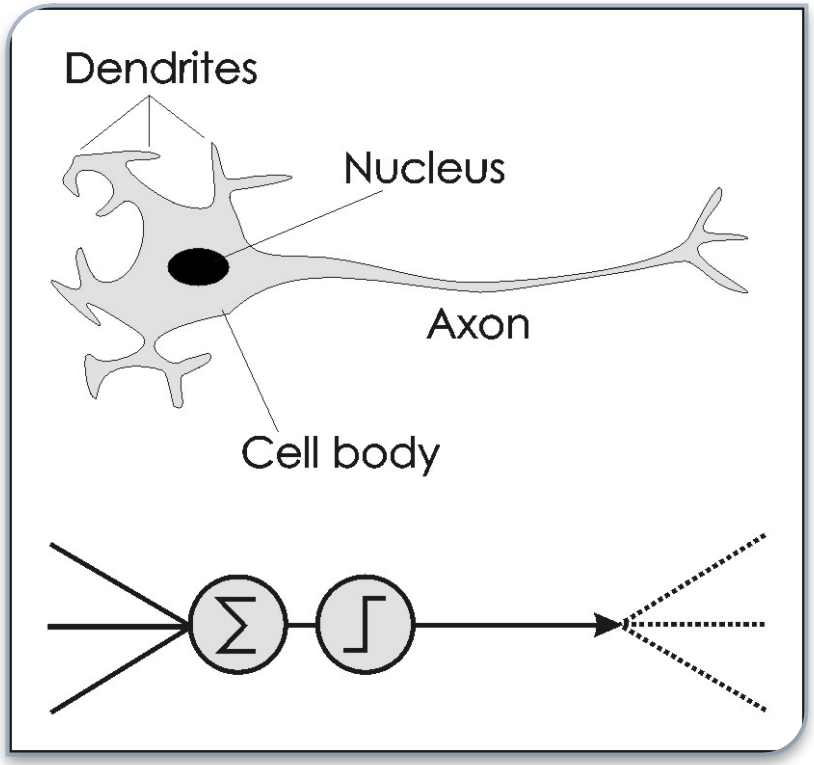


GamePlay

Many more emerging…

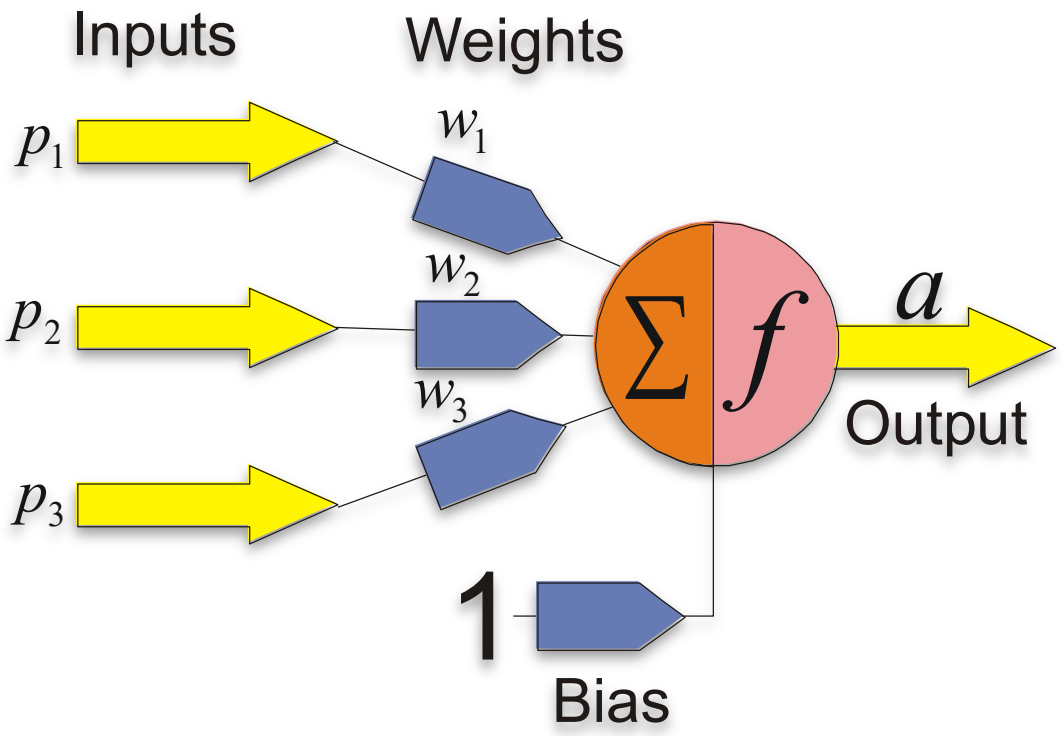
Others

The Neuron



- A neuron's dendritic tree is connected to a thousand neighbouring neurons. When one of those neurons fire, a positive or negative charge is received by one of the dendrites. The strengths of all the received charges are added together through the processes of spatial and temporal summation

Anatomy of Neuron

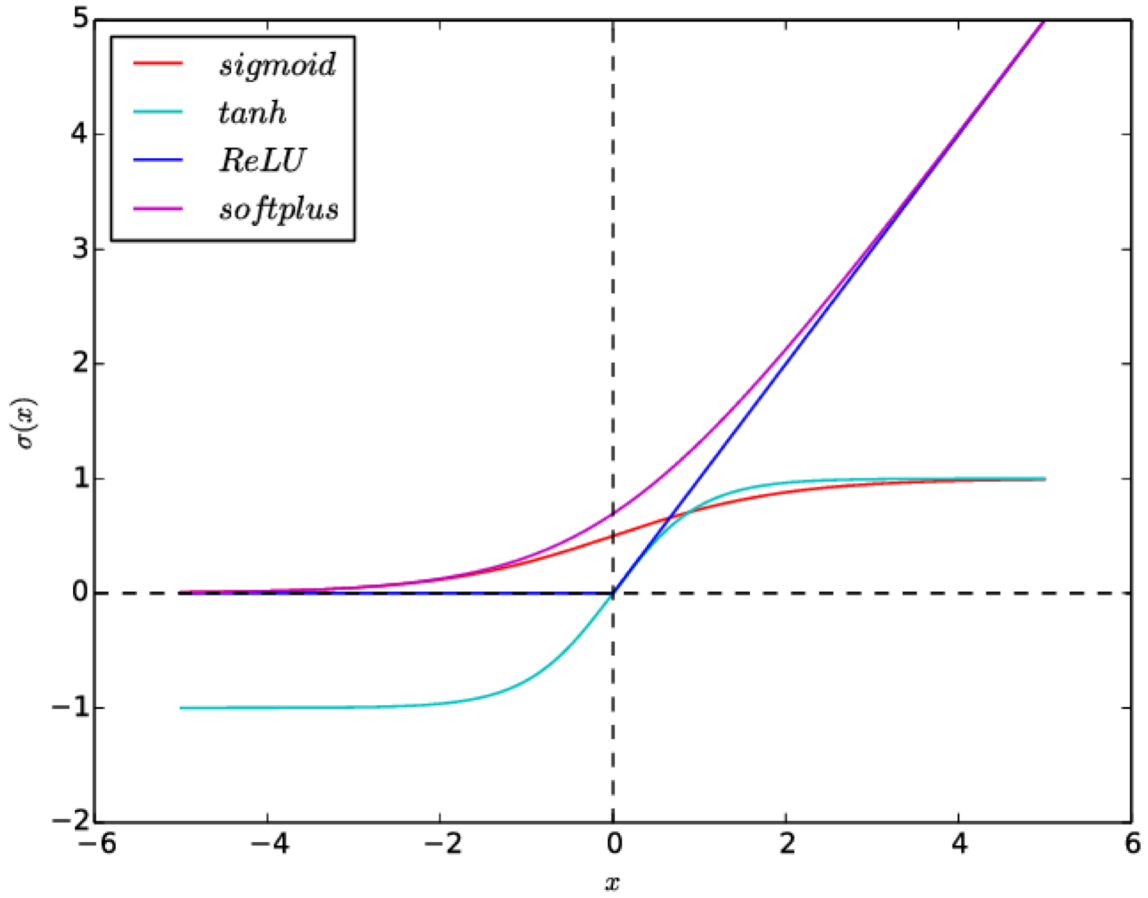


- Basic Computation is Multiply Add
  - Sum of Inputs x weights
- Activation function can vary
  - Differentiable?

$$a = f\left(p_1 w_1 + p_2 w_2 + p_3 w_3 + b\right) = f\left(\sum p_i w_i + b\right)$$

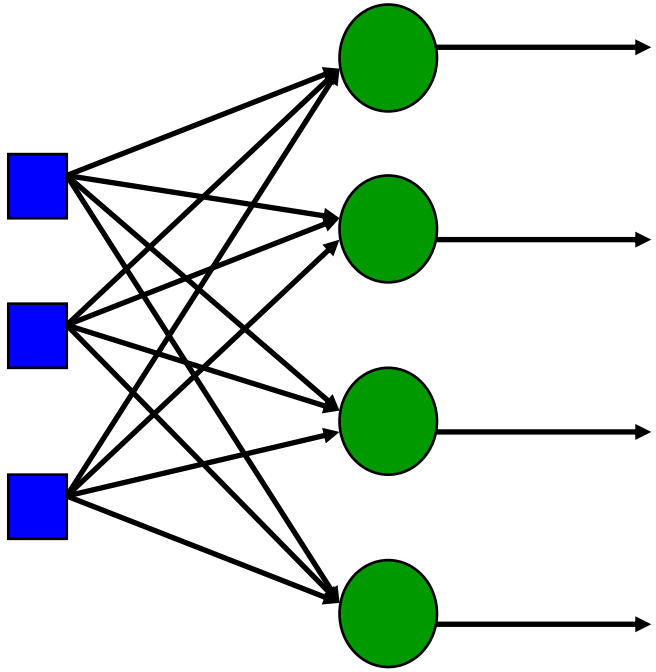
What is the Activation Function F?

- Linear functions
  - limited because the output is simply proportional to the input.
- Non-linear functions
  - Sigmoid
  - Step



Single Layer

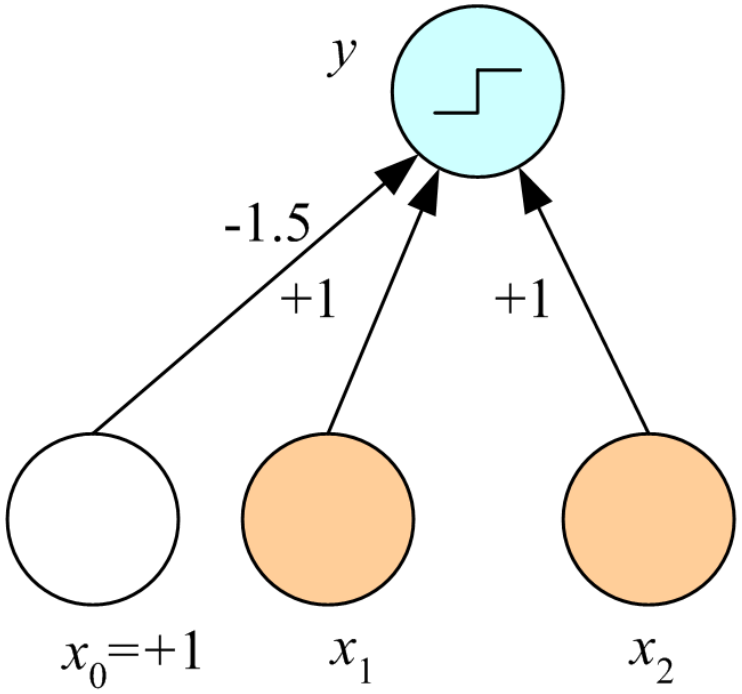
Input layer of source nodes

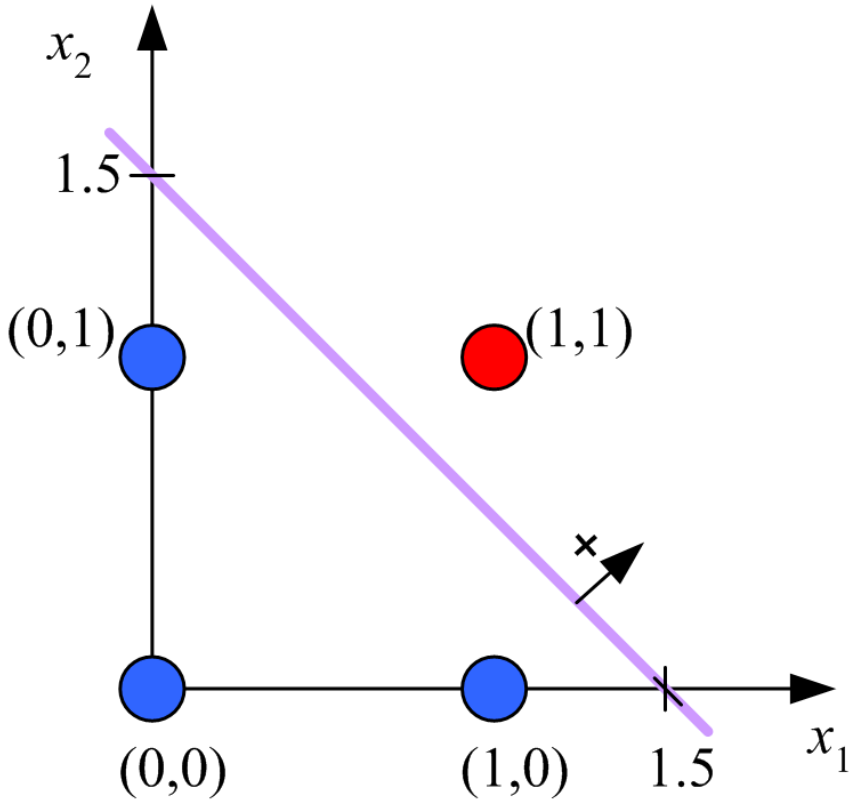


Output layer of neurons

Example: Learning Boolean AND

x_1	x_2	r
0	0	0
0	1	0
1	0	0
1	1	1

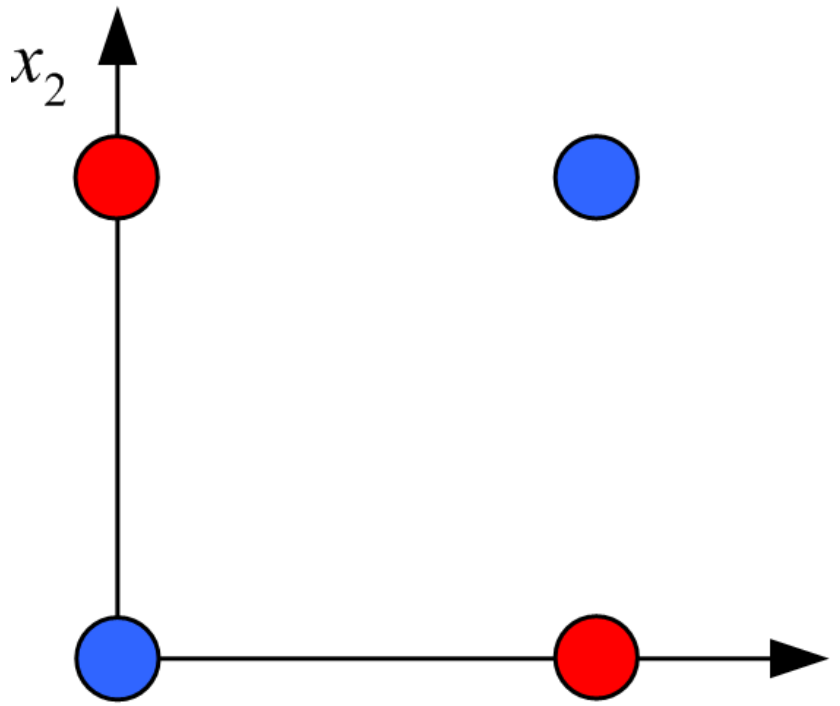




What ABOUT XOR Function?

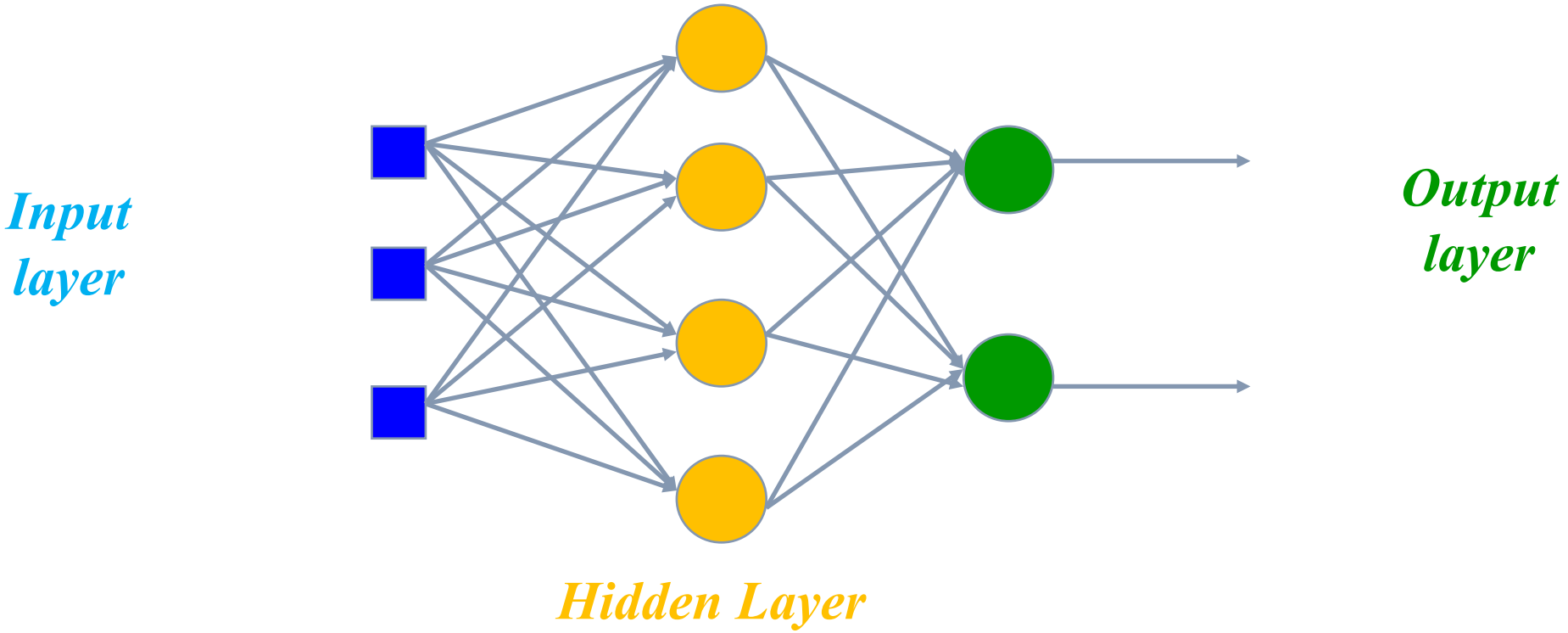
- Linearly **non** separable

x_1	x_2	r
0	0	0
0	1	1
1	0	1
1	1	0

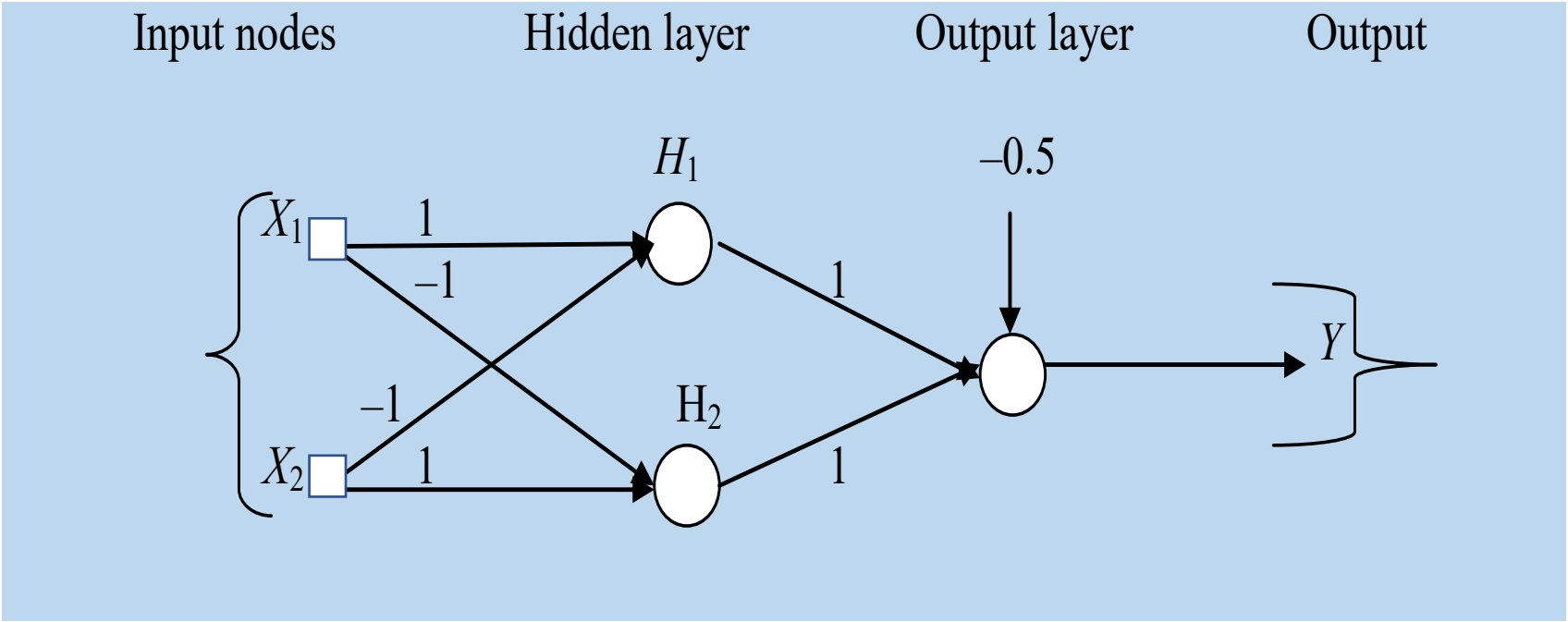


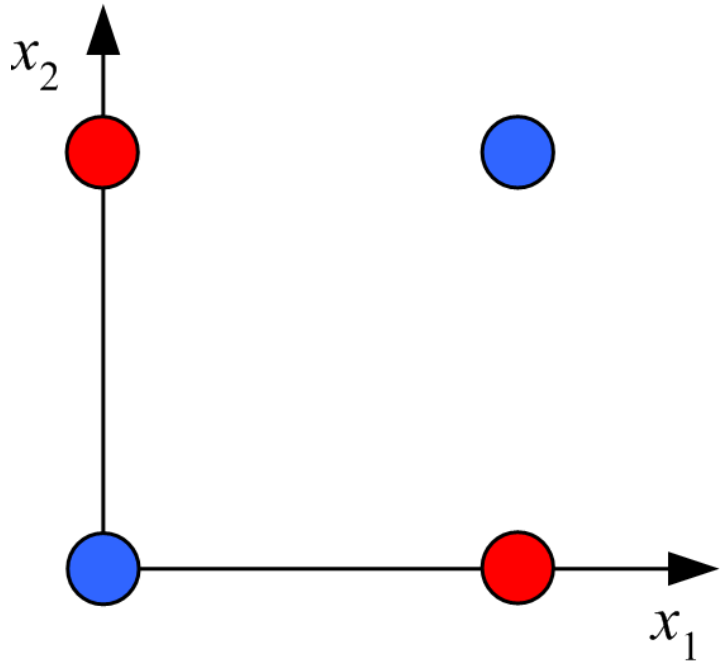
(Minsky and Papert, 1969)

Hidden Layer(s)

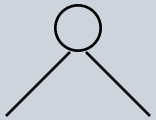
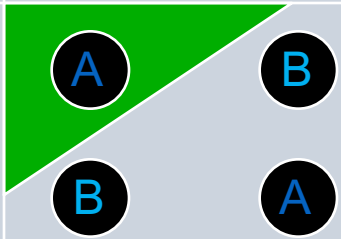
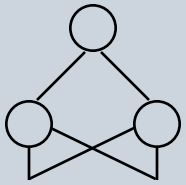
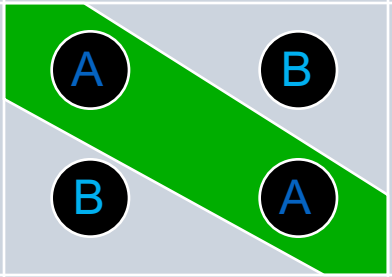
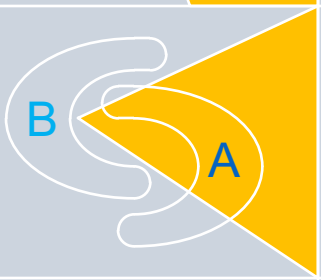
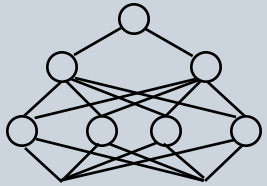
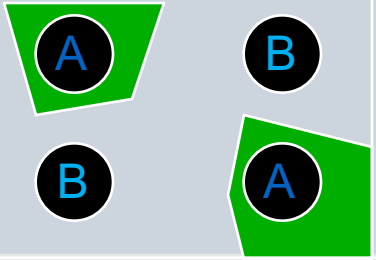
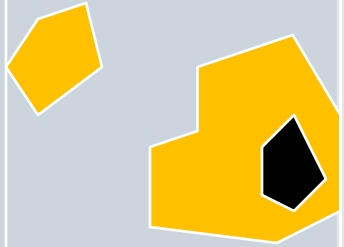


XOR Function



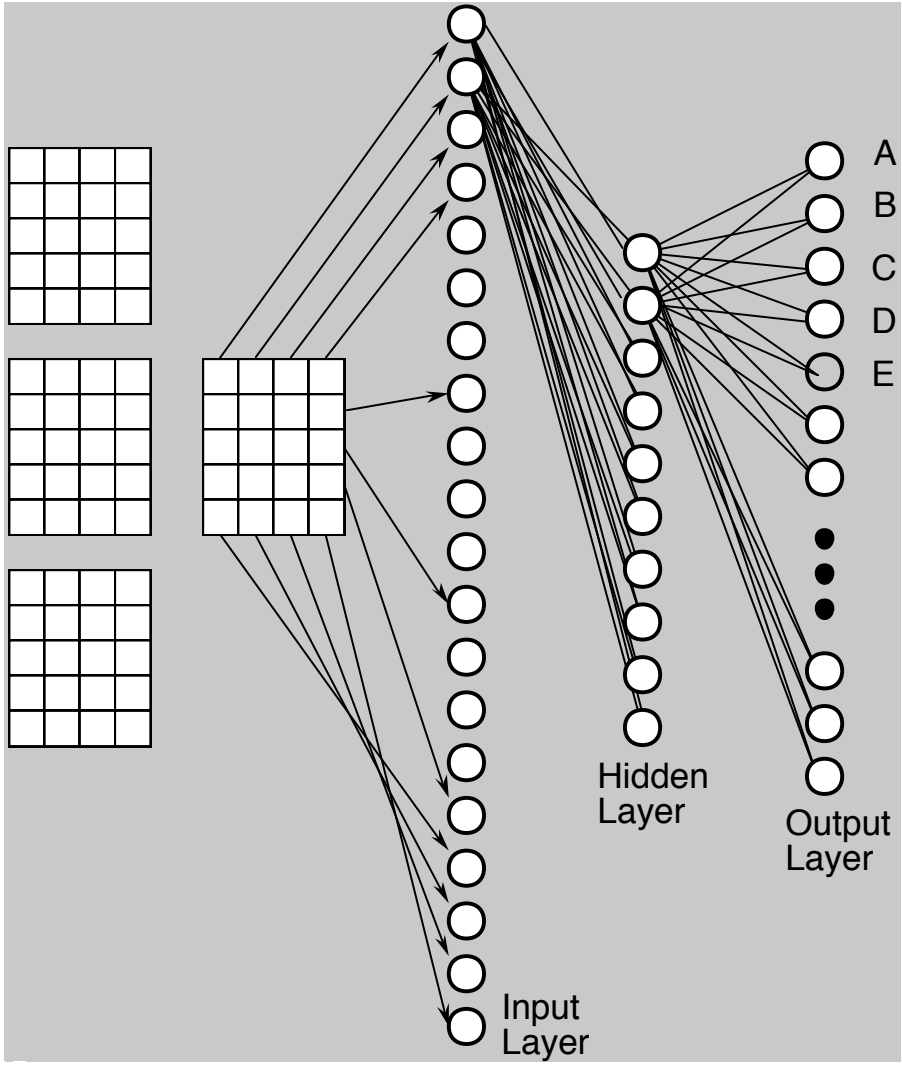


Non-Linearly Separable Problems

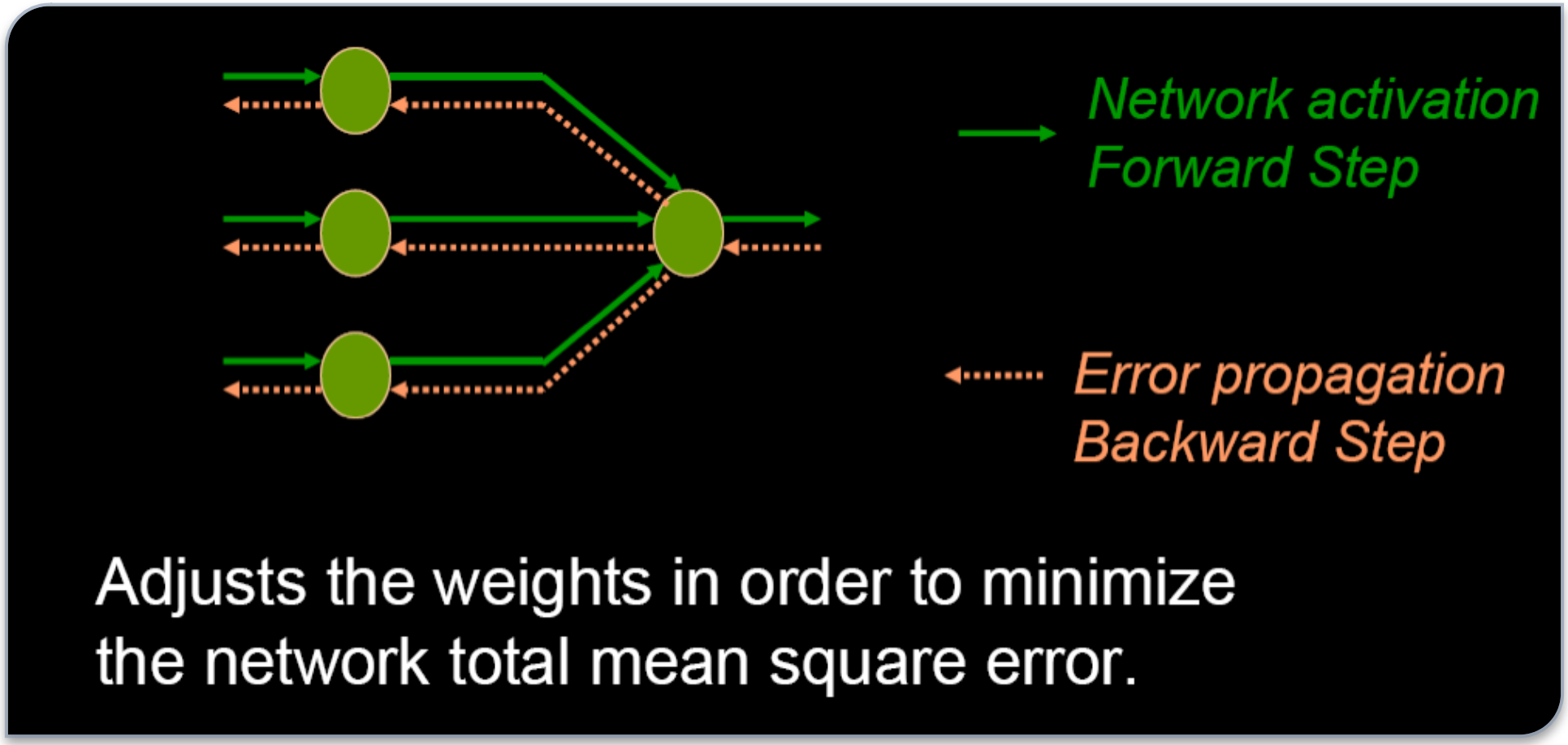
Structure	*Types of Decision Regions*	*Exclusive-OR Problem*	*Classes with Meshed regions*	*Most General Region Shapes*
Single-Layer 	*Half Plane Bounded By Hyperplane*			
Two-Layer 	*Convex Open Or Closed Regions*			
Three-Layer 	Arbitrary (Complexity Limited by No. of Nodes)			

Neural network for OCR

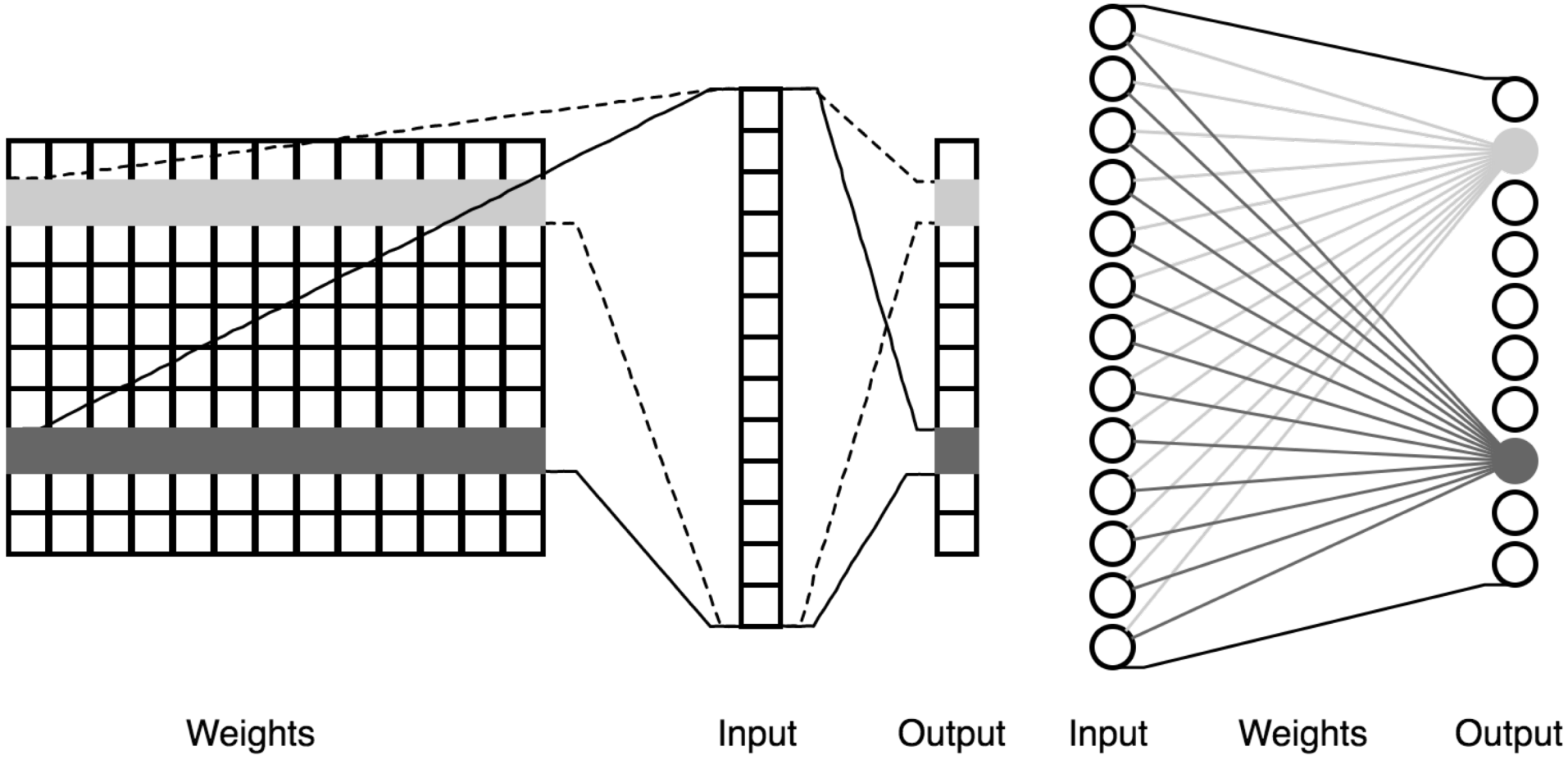
- Feedforward network
- Trained using Back- propagation



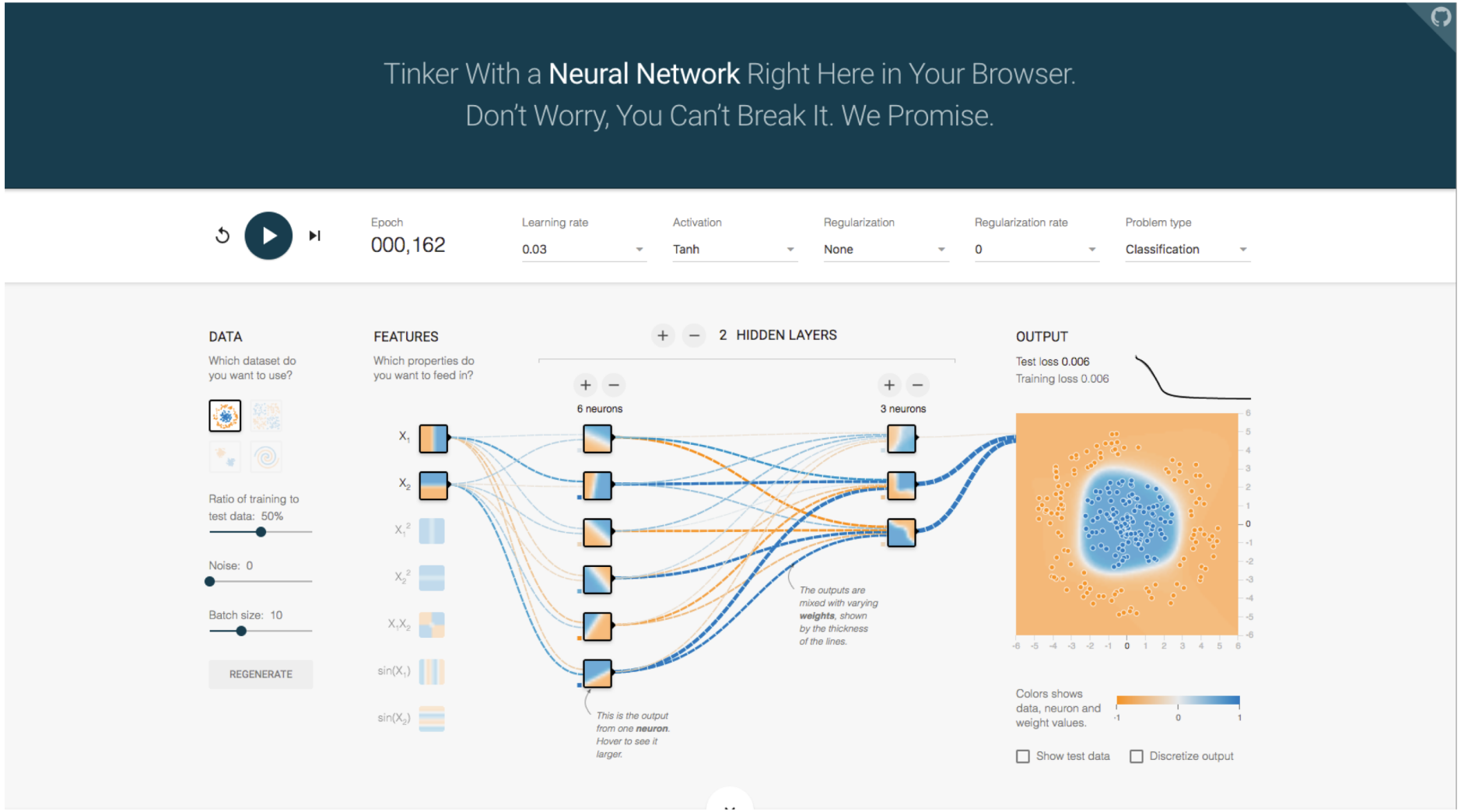
Learning: Back Propagation

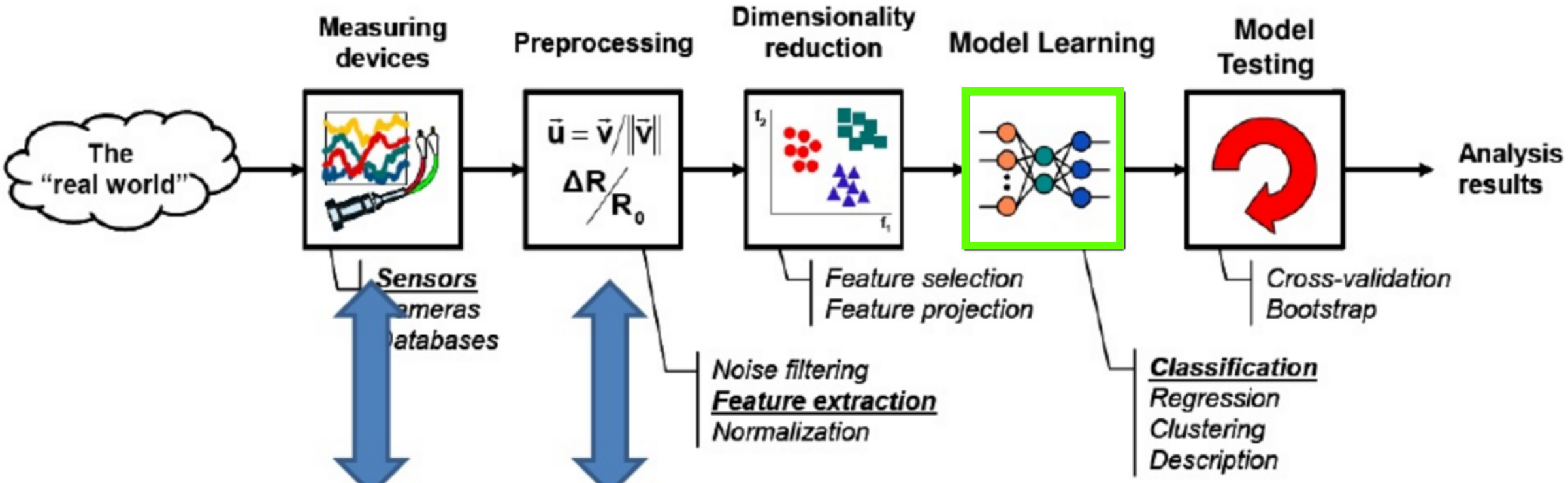


Forward Propagation is GEMV



Playground.tensorflow.com

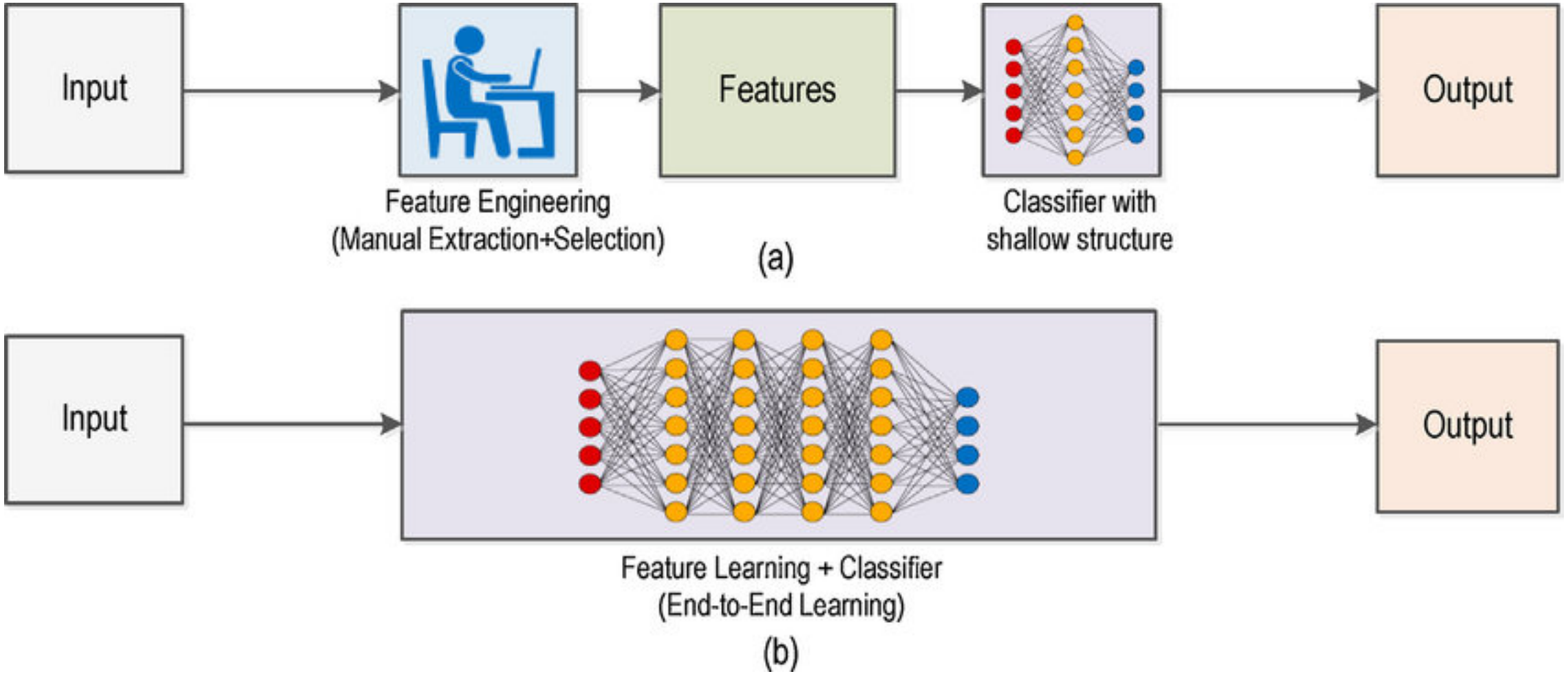


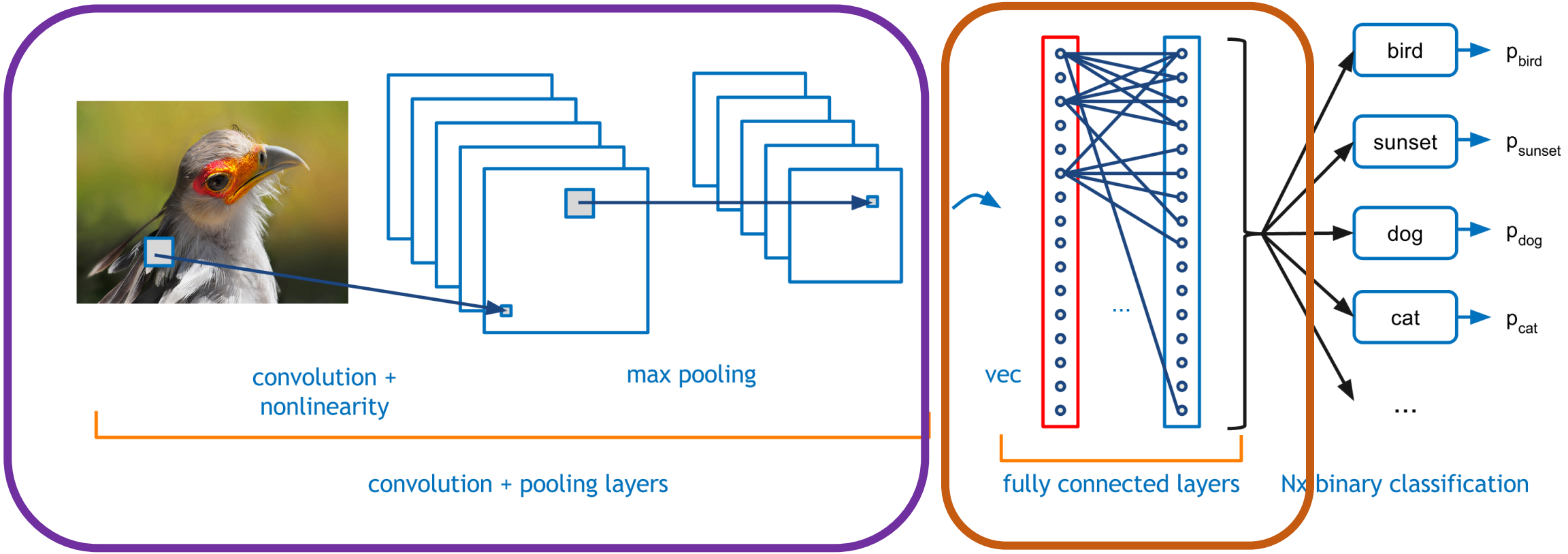


Email Server

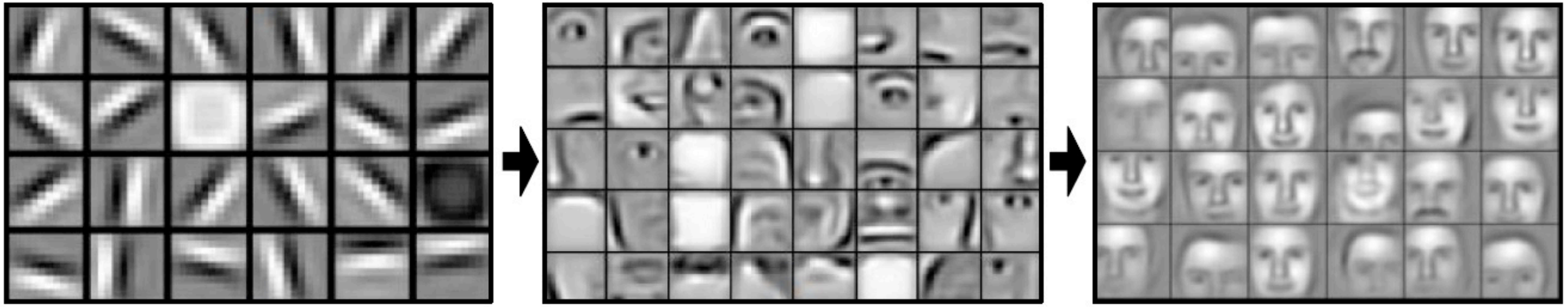
- Number of recipients
- Size of message
- Number of attachments
- Number of "re's" in the subject line
- ...

Shallow & Deep Learning




convolution +
nonlinearity
max pooling
convolution + pooling layers
vec
...
fully connected layers
bird
sunset
dog
cat
...
p_{bird}
p_{sunset}
p_{dog}
p_{cat}
Nx binary classification

Features in End-to-end learning



local appearance

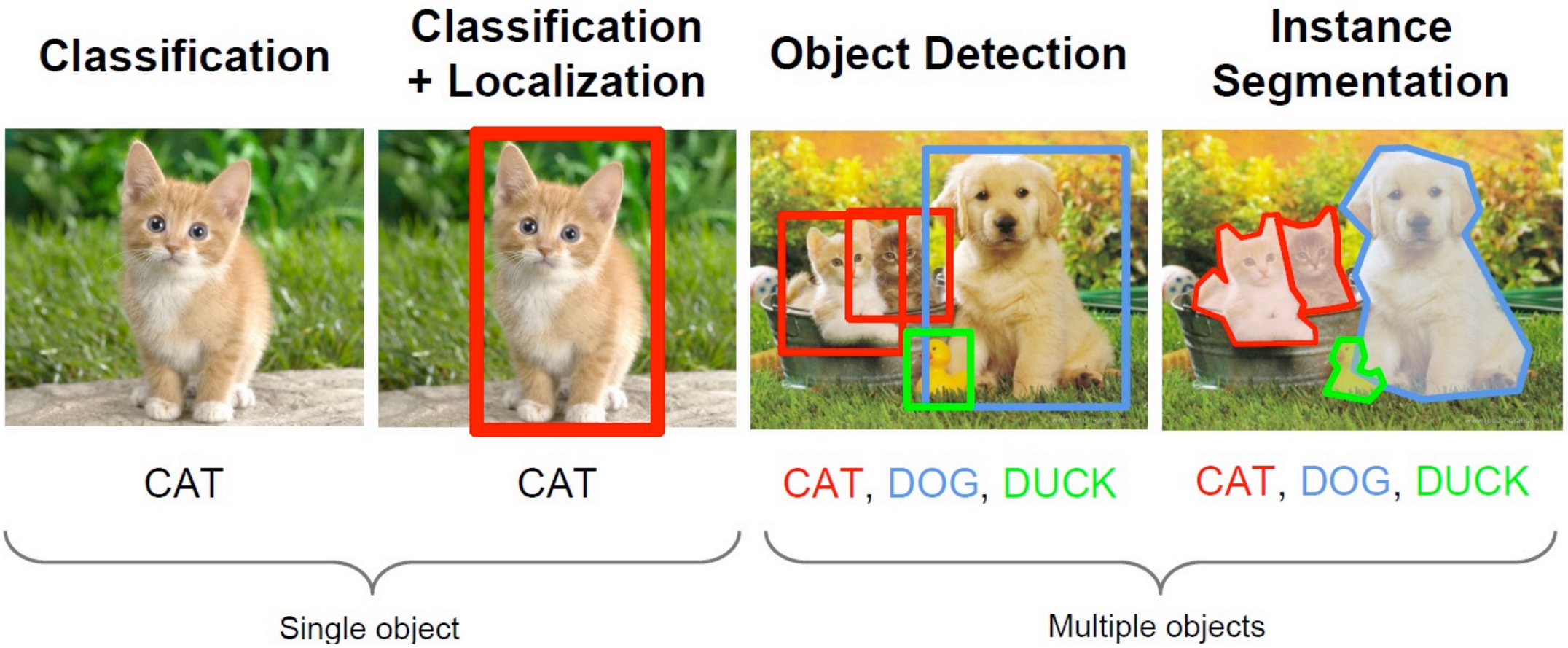
parts and texture

objects and semantics

Learn the representation from data

[figure credit H. Lee]

Computer Vision Tasks

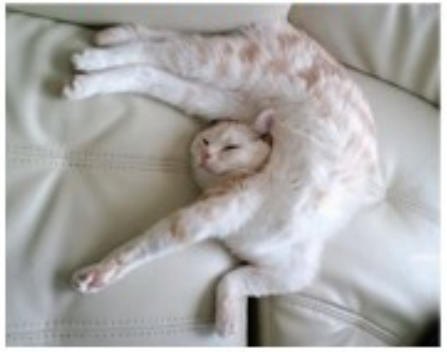


Challenges

Viewpoint variation

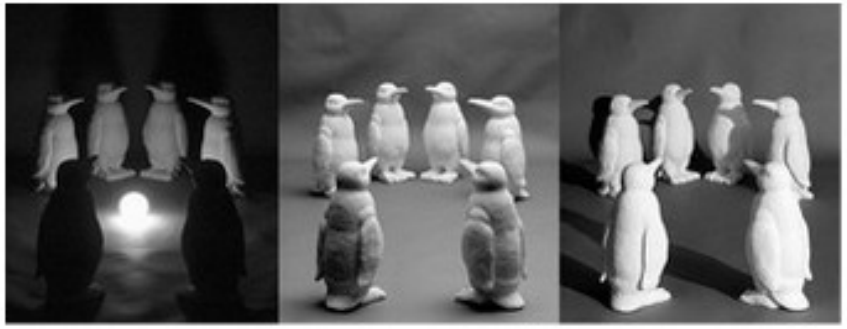
Scale variation

Deformation



Occlusion

Illumination conditions



Background clutter

Intra-class variation

Hand Engineered Features

- Years of Vision algorithms

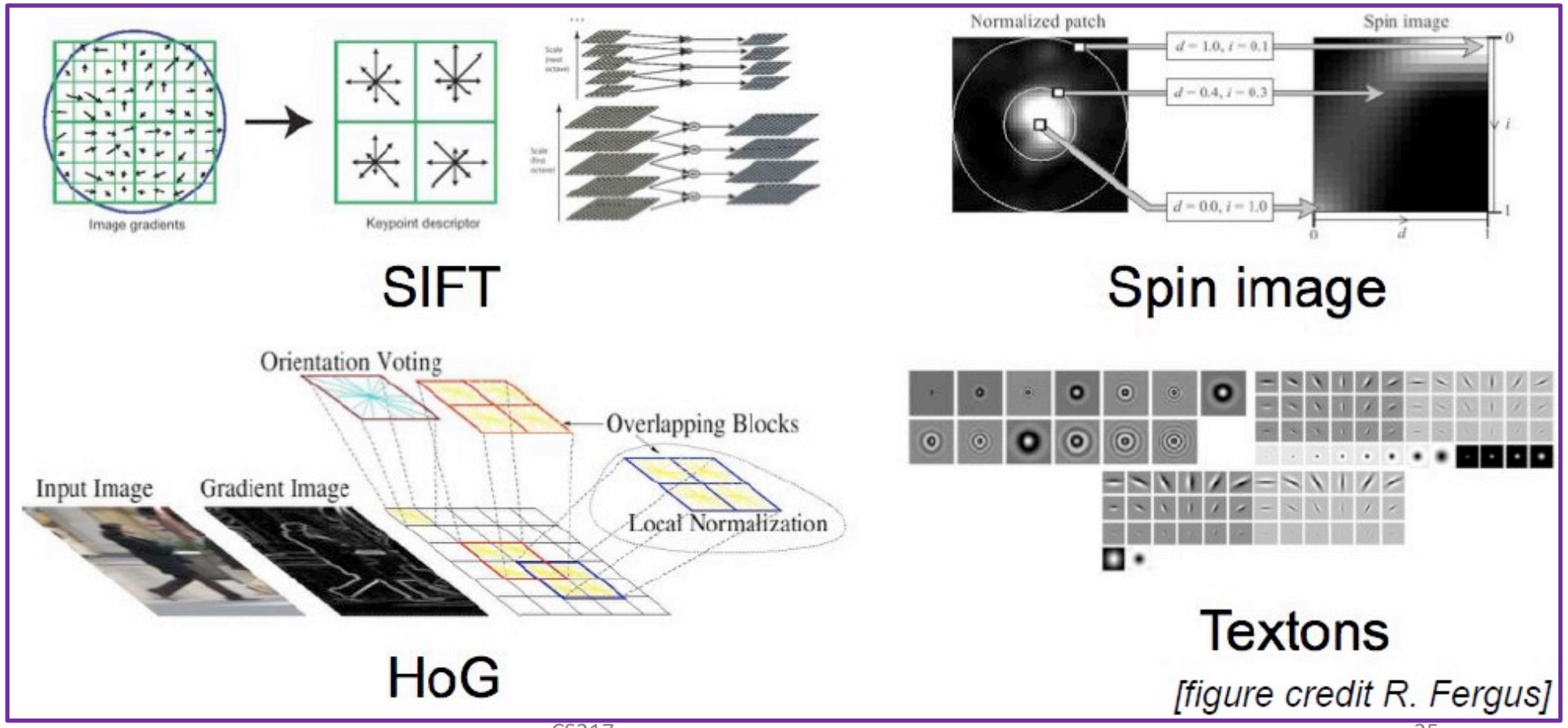


Image Scanning Detectors

Detection Phase

Scan image(s) at all scales and locations

↓

Extract features over windows

↓

Run window classifier at all locations

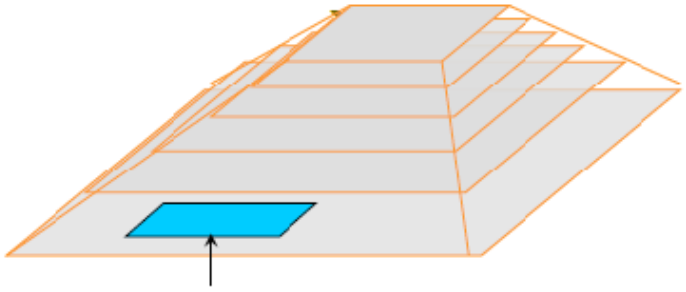
↓

Fuse multiple detections in 3-D position & scale space

↓

Object detections with bounding boxes

Scale-space pyramid



Detection window

Viola Jones Face Detection

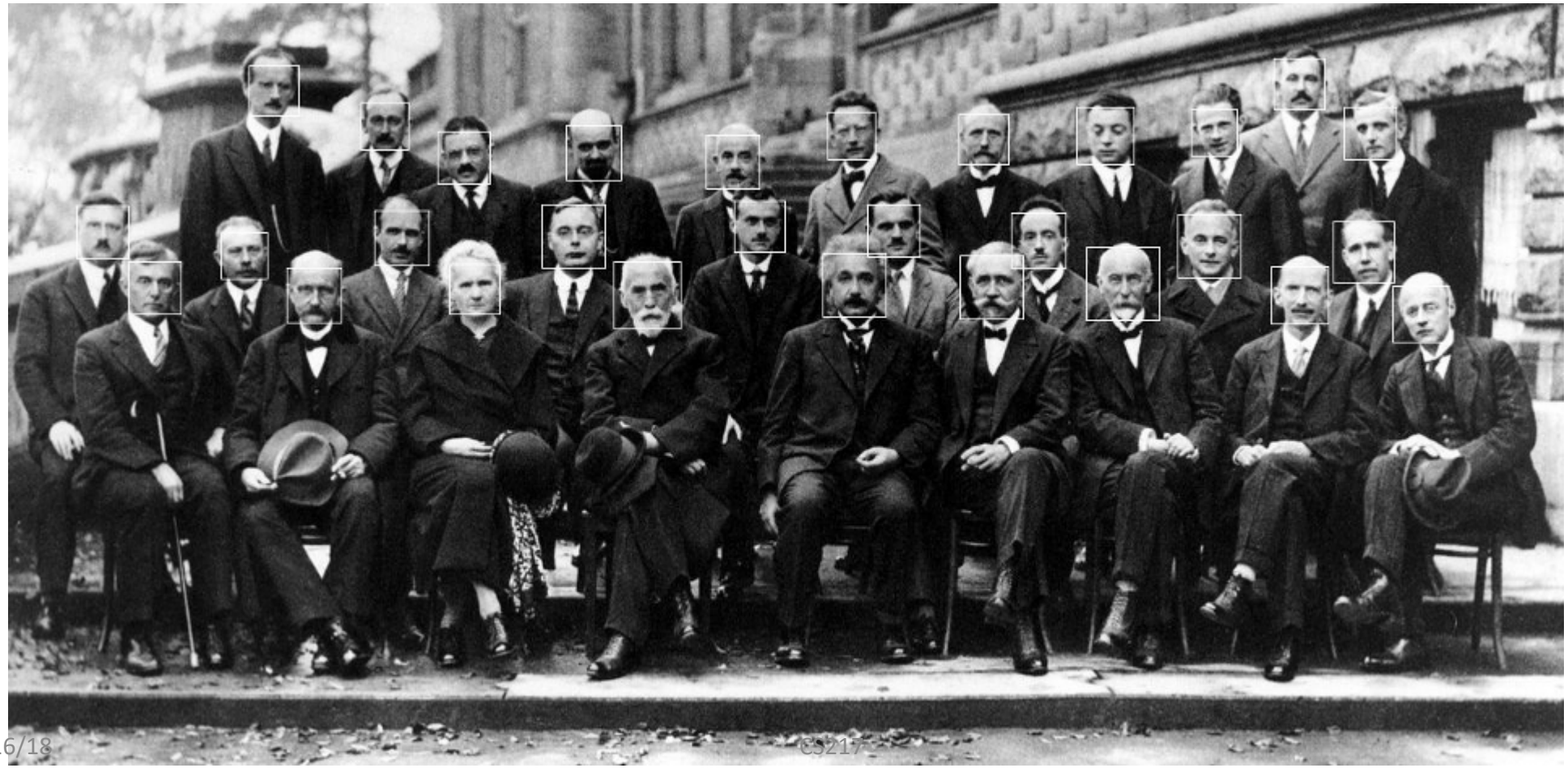
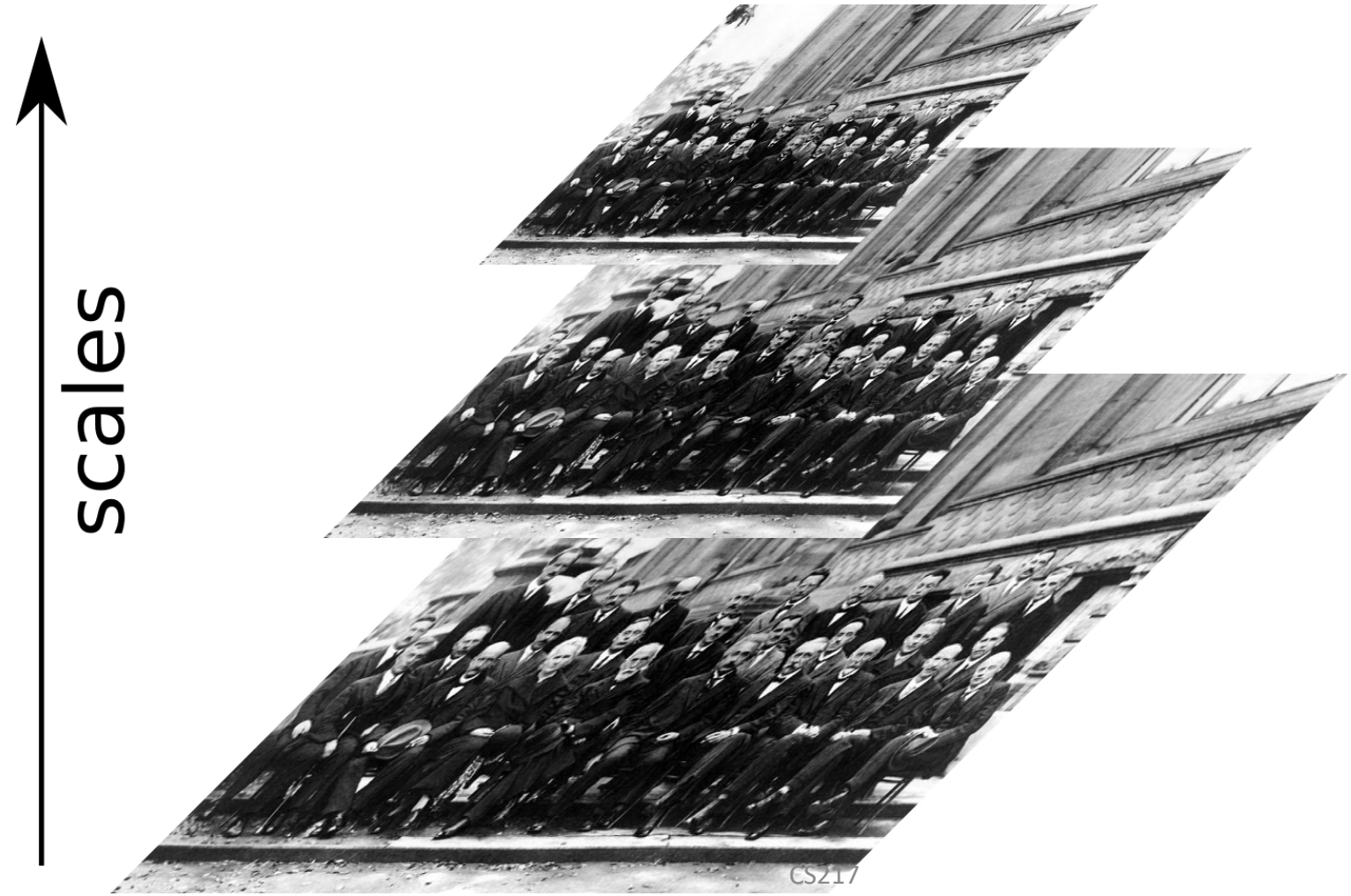
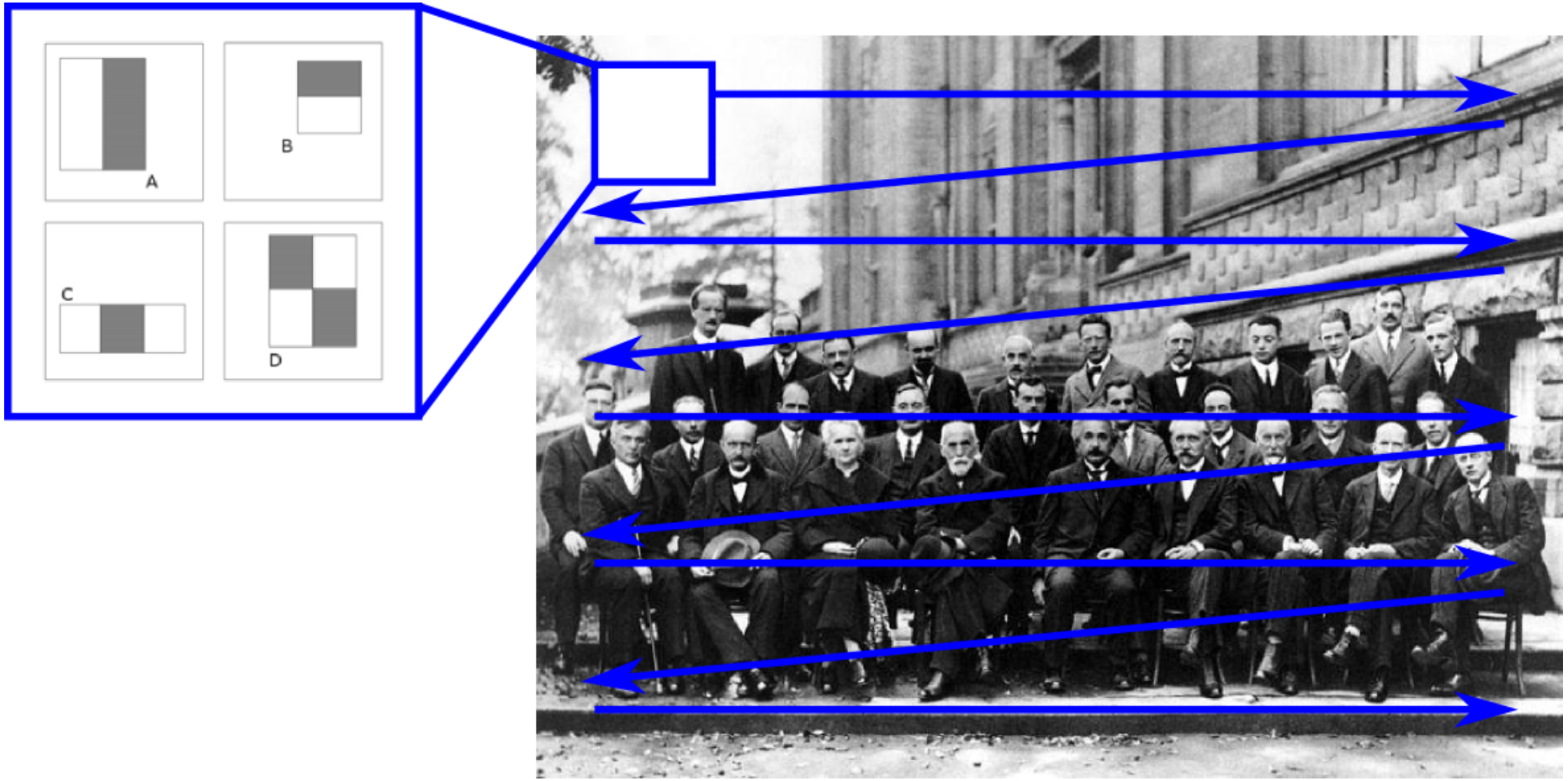
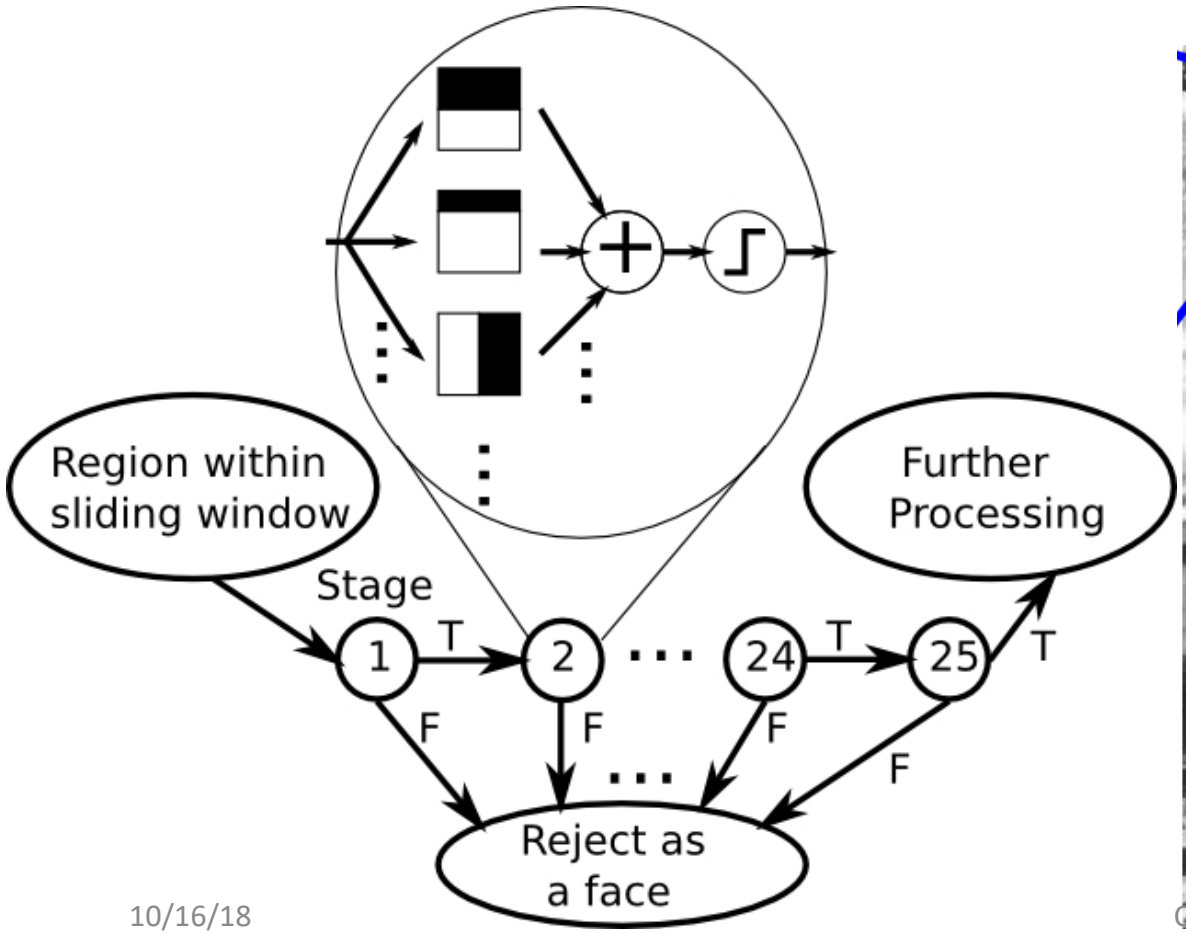


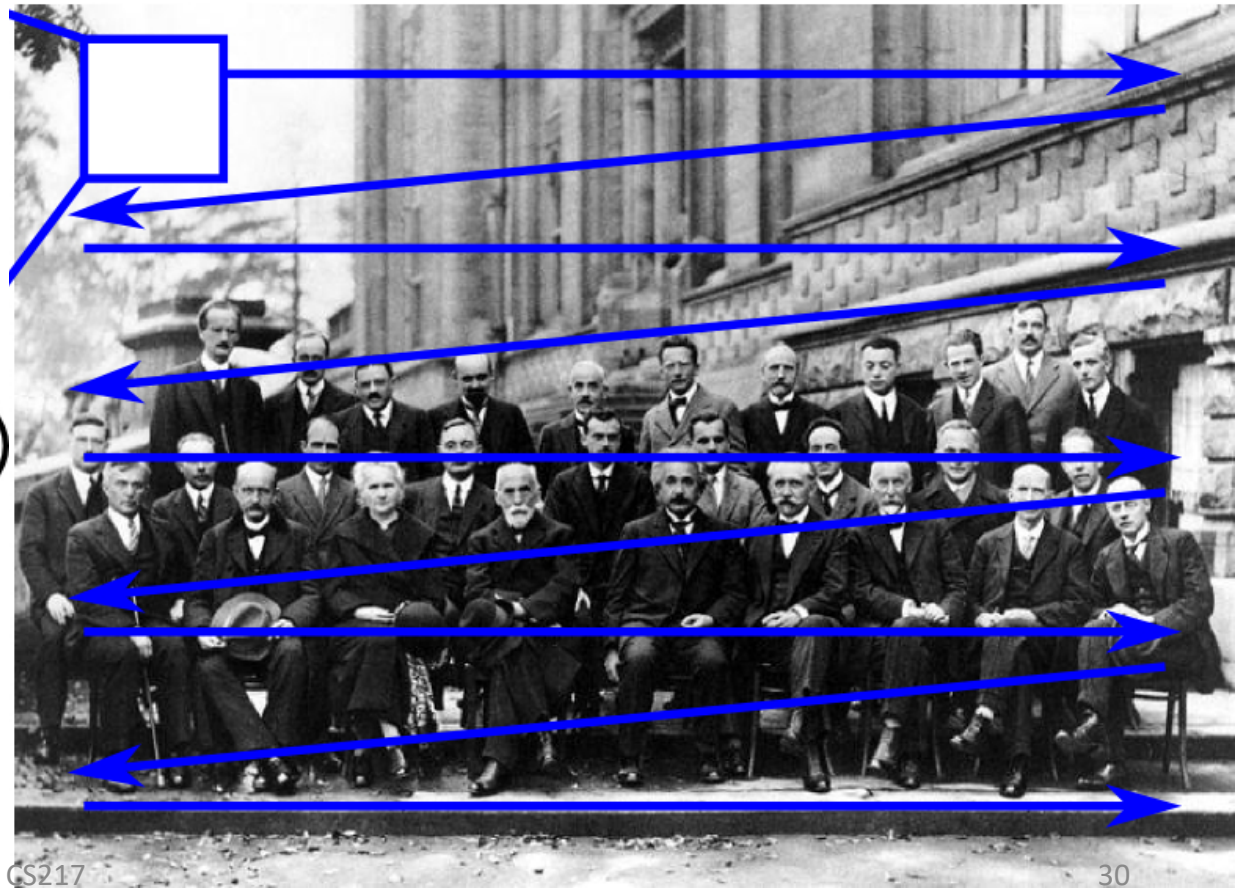
IMAGE PYRAMID



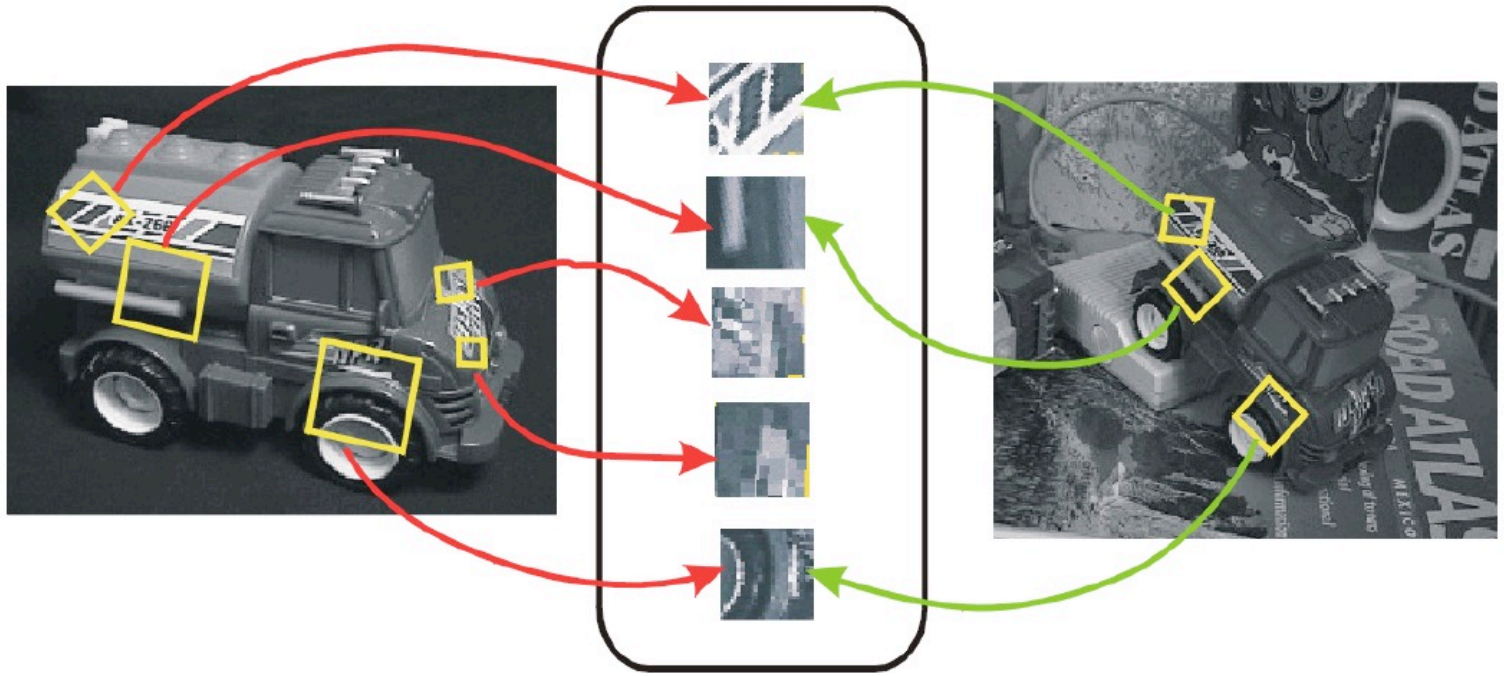
SLIDING WINDOW and COMPARE




Region within
sliding window
Stage
1
T
2
24
T
25
T
Further
Processing
F
F
F
F
Reject as
a face



Feature Descriptors, Detection and Matching



- Encode distinctive local structure at a collection of image points for matching between images
- Modest changes in viewing conditions
  - scale
  - orientation
  - Contrast

Descriptors

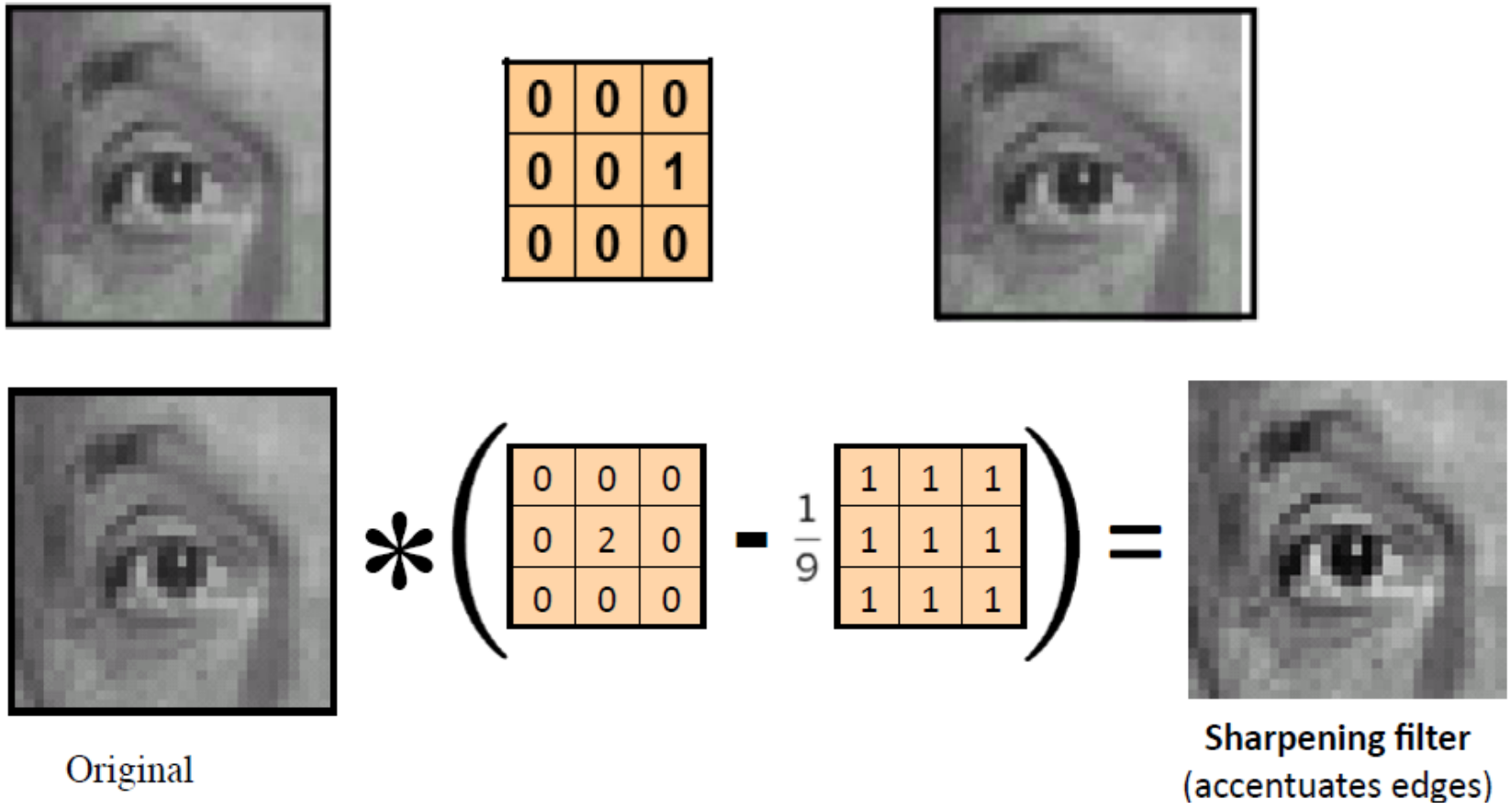
- A good Tutorial:

https://gilscvblog.com/2013/08/18/a-short-introduction-to-descriptors/

- Scale Invariant Feature Transformation (SIFT)
- Histogram of Oriented Gradients (HOG)

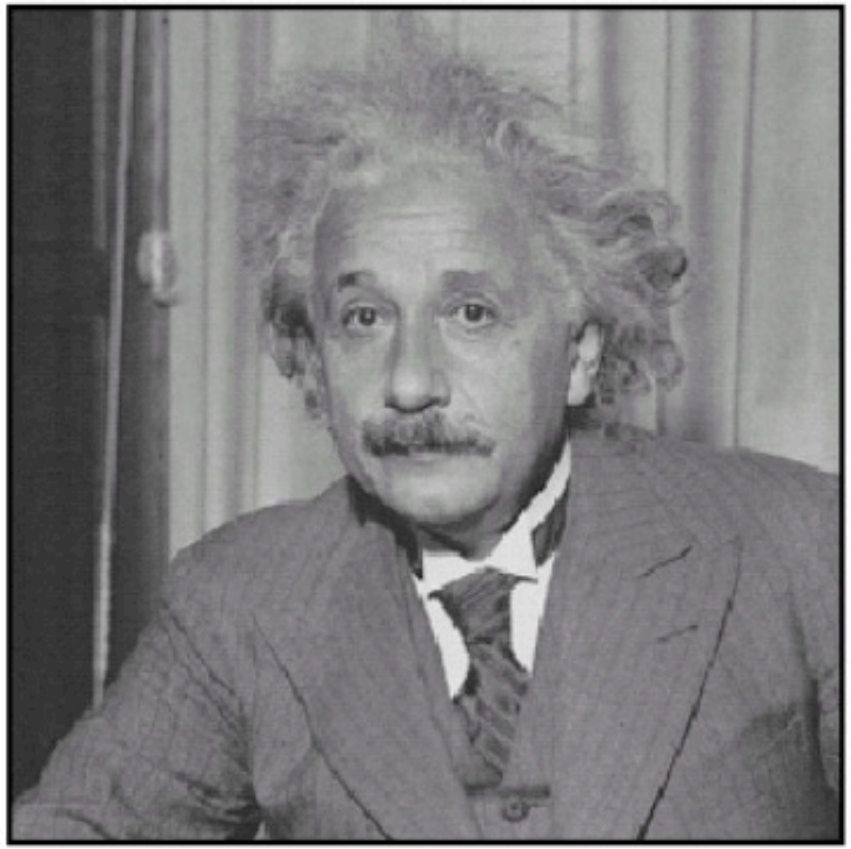
- Feature point detection
- Feature descriptors
- Potential applications –
  - Image panoramas
  - image matching and registration
  - Long range motion (tracking by detection)
  - Stereoscopic vision / 3D reconstruction
  - Object recognition

Convolution and Filtering

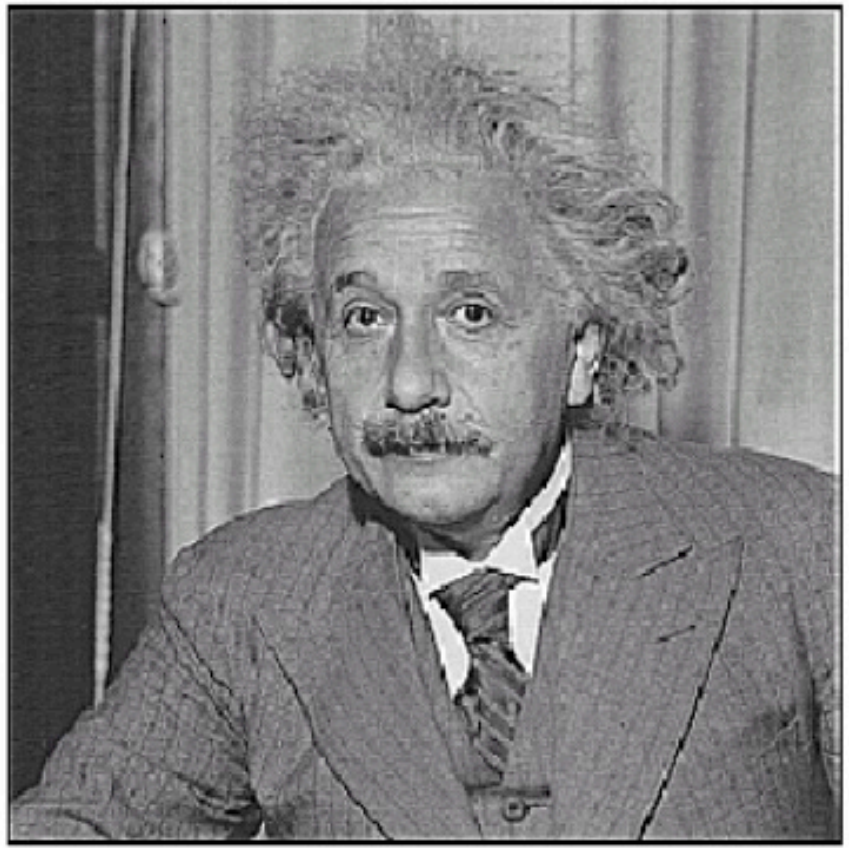


- http://www.cs.toronto.edu/~urtasun/courses/CV/lecture02.pdf

CONVOLUTION AND FILTERING

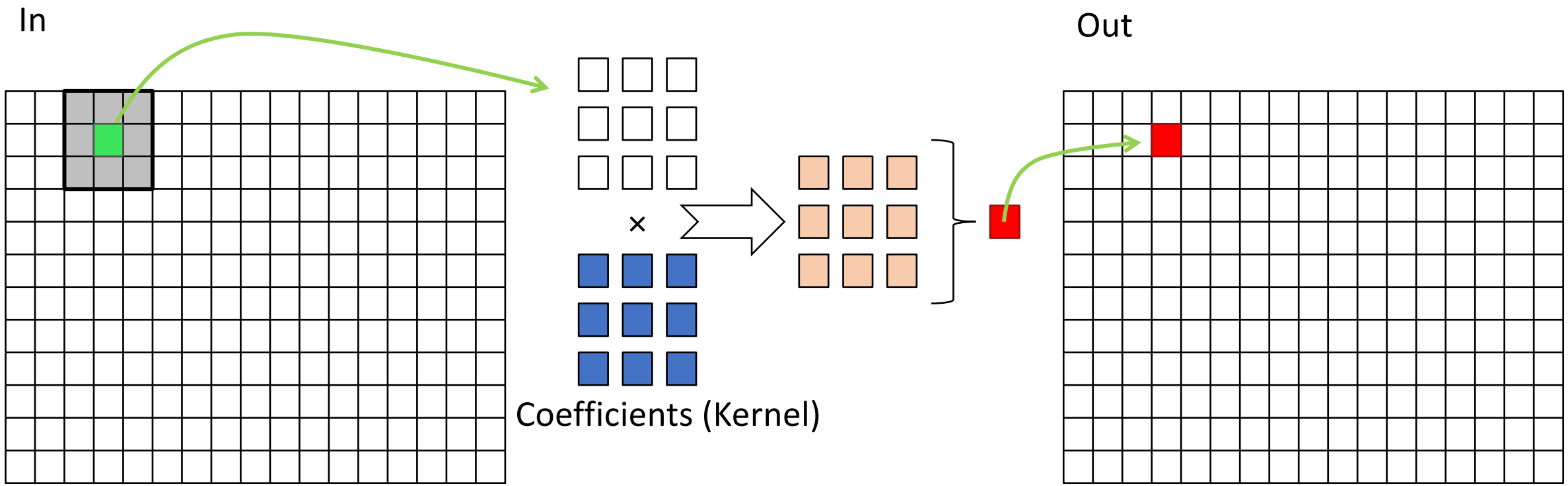


before

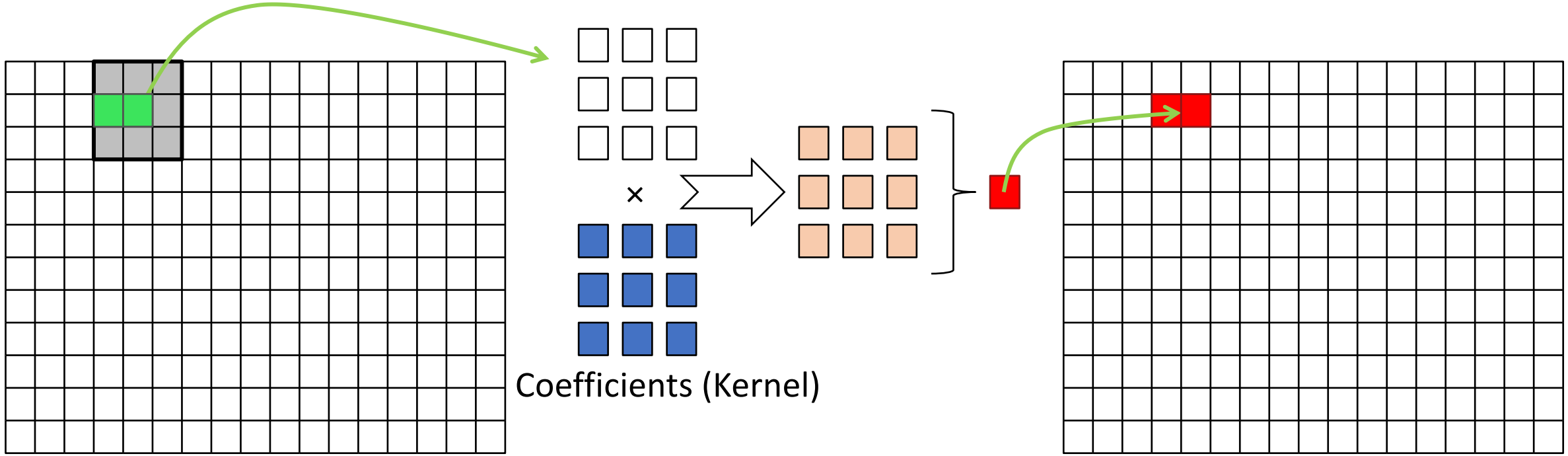


after

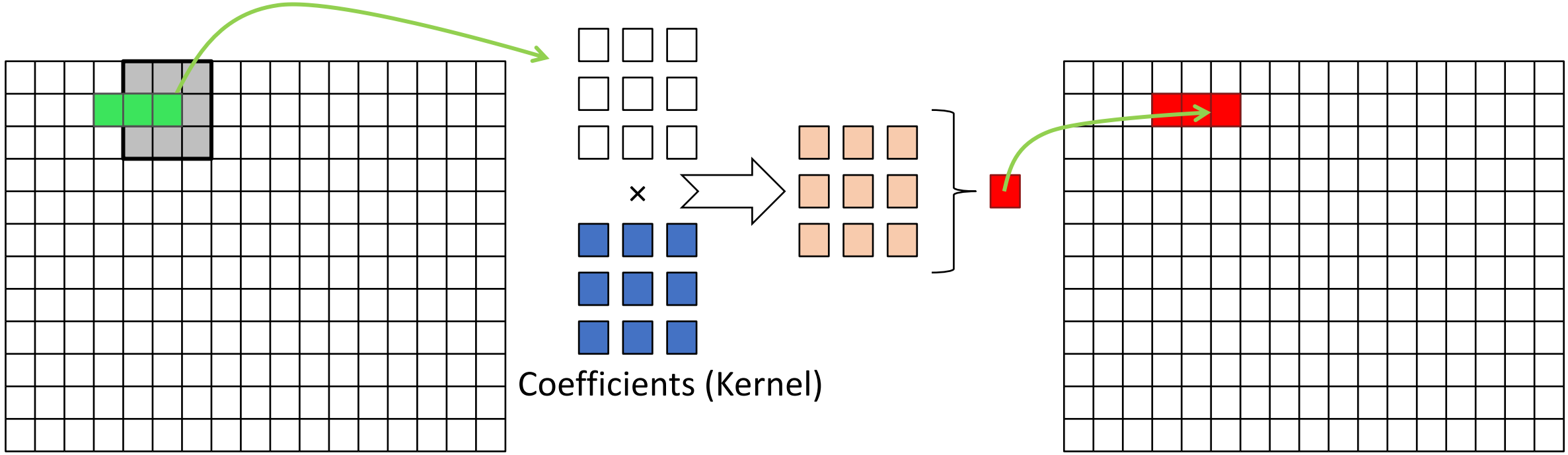
Convolution Operation



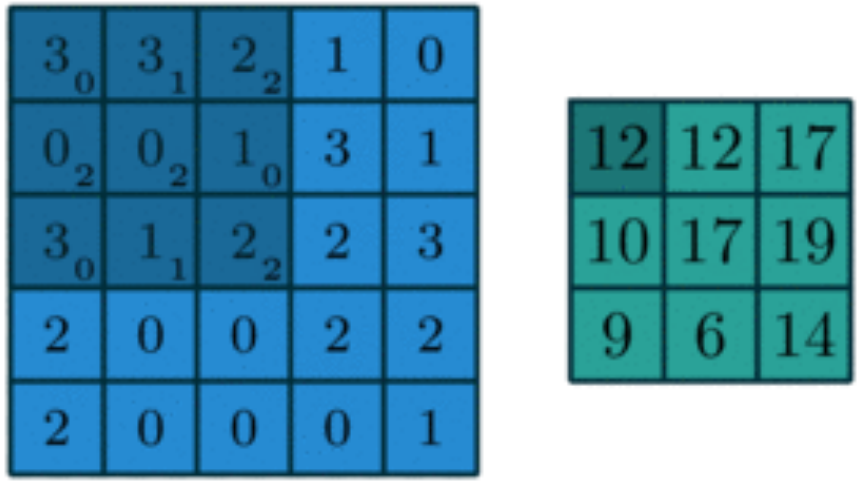
Convolution Operation



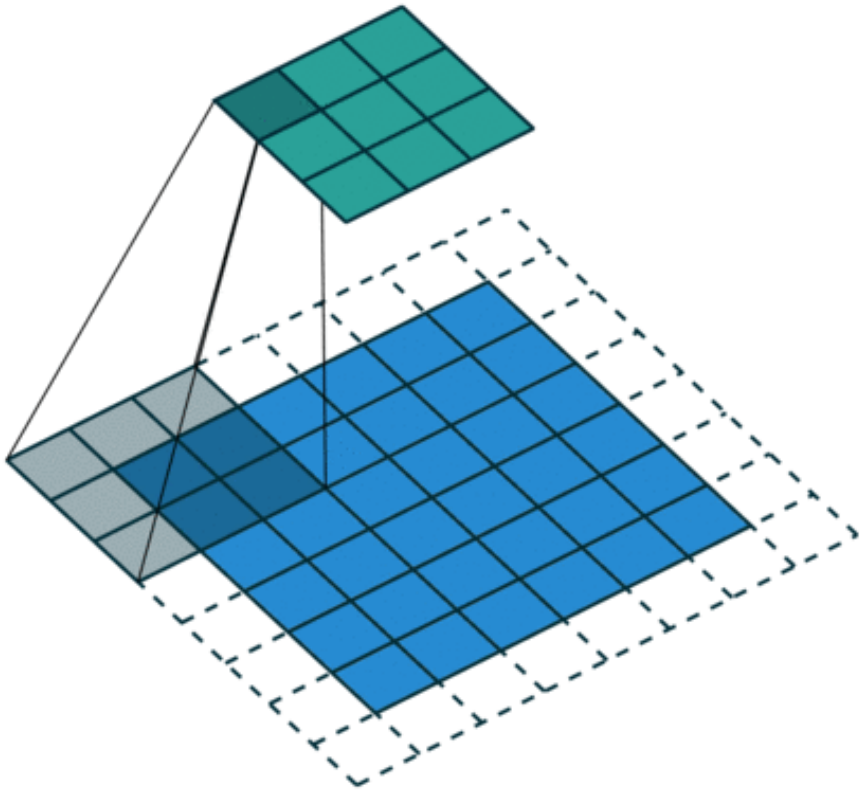
Convolution Operation



Convolution Operation



No zero padding, unit strides



Zero padding, non-unit strides

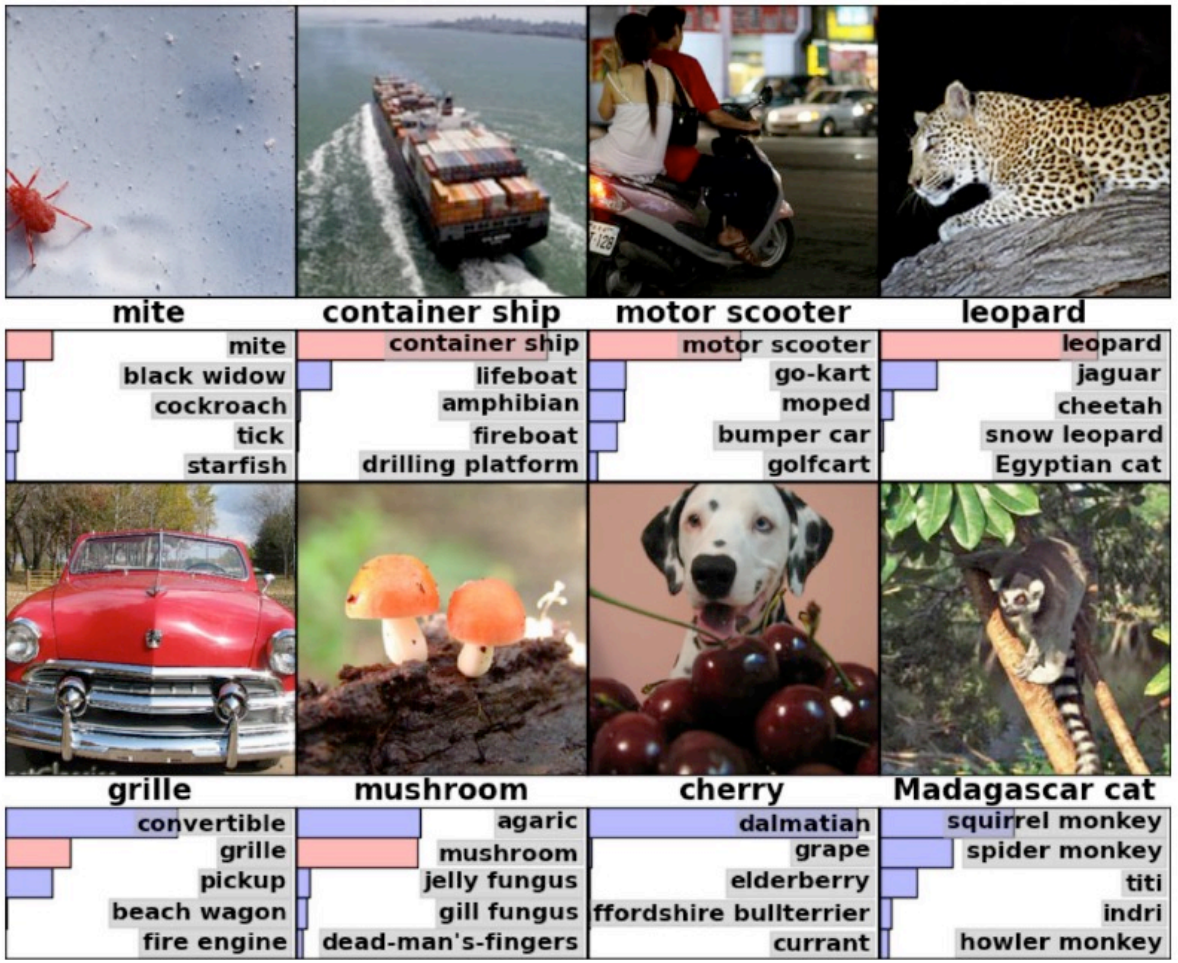
http://deeplearning.net/software/theano_versions/dev/tutorial/conv_arithmetic.html

Chihuahua or Muffin?

That Is The Question

ImageNET challenge

- 1,000 object classes (categories).
- Images:
  - 1.2 M train
  - 100k test.

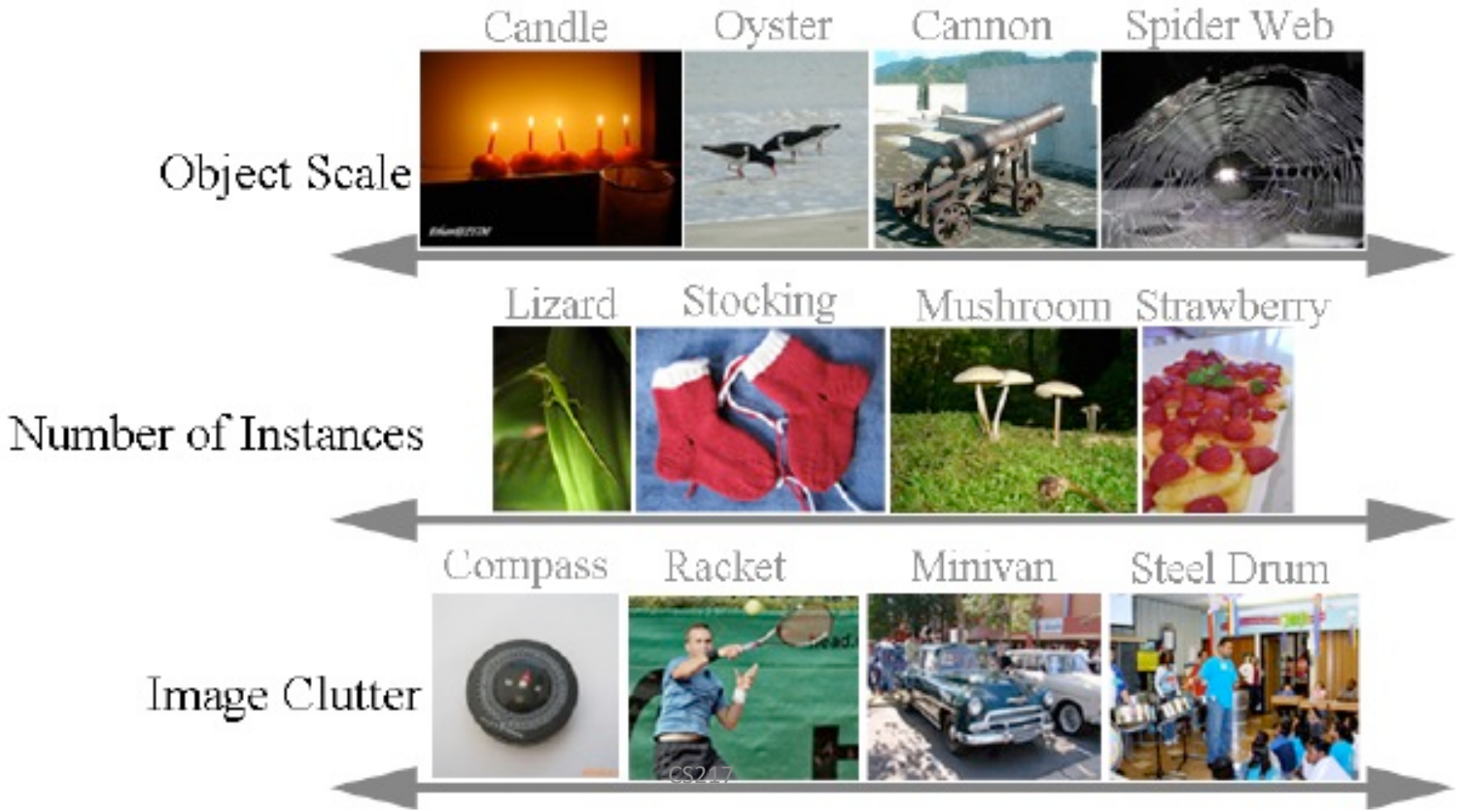


Imagenet dataset

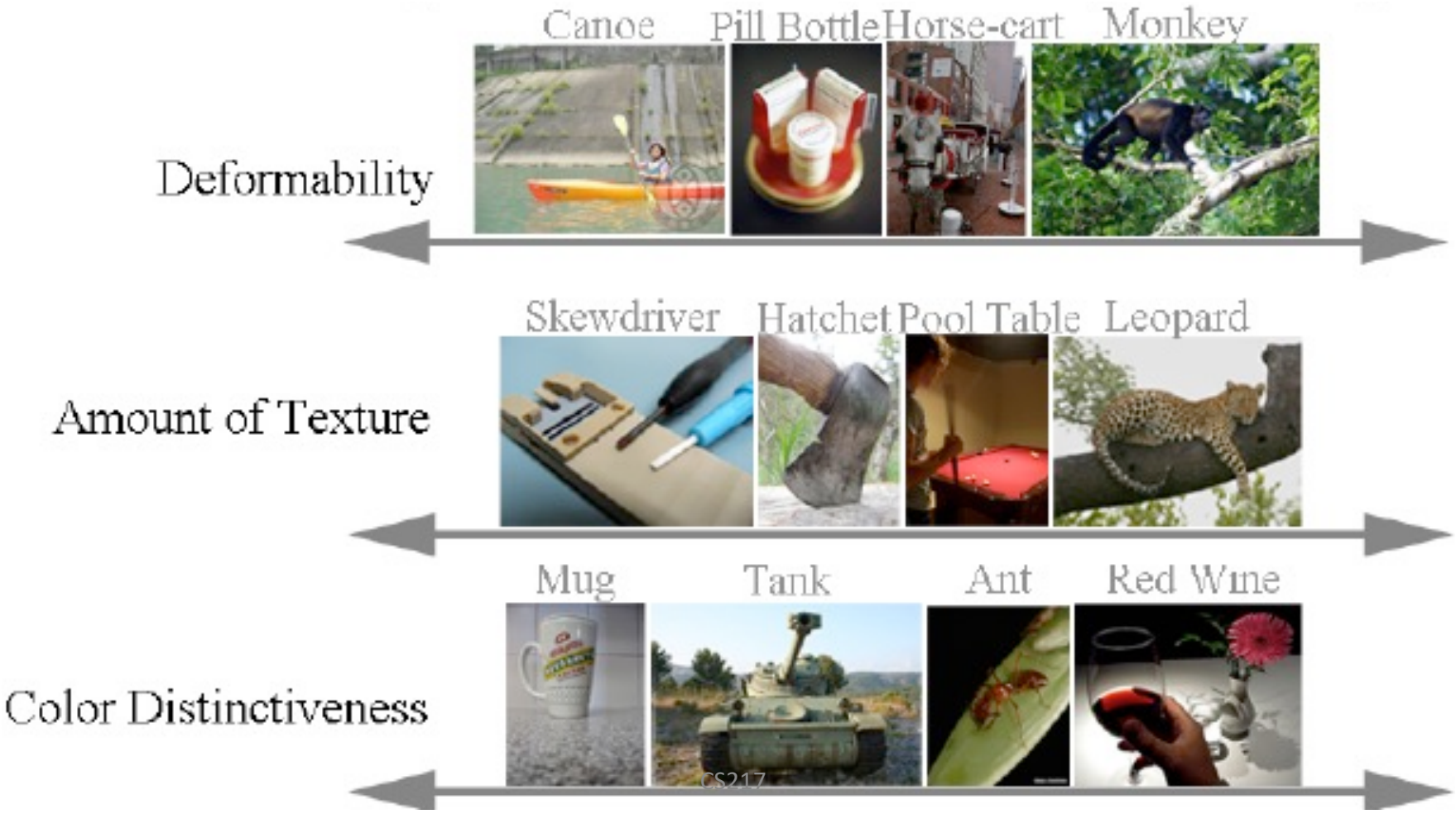
Li Fei-Fei, "How we're teaching computers to understand pictures" TEDTalks 2014.

Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., ... & Fei-Fei, L. (2015). Imagenet large scale visual recognition challenge. *arXiv preprint arXiv:1409.0575*. [web]

IMAGENET


Candle
Oyster
Cannon
Spider Web
Object Scale
Lizard
Stocking
Mushroom
Strawberry
Number of Instances
Compass
Racket
Minivan
Steel Drum
Image Clutter

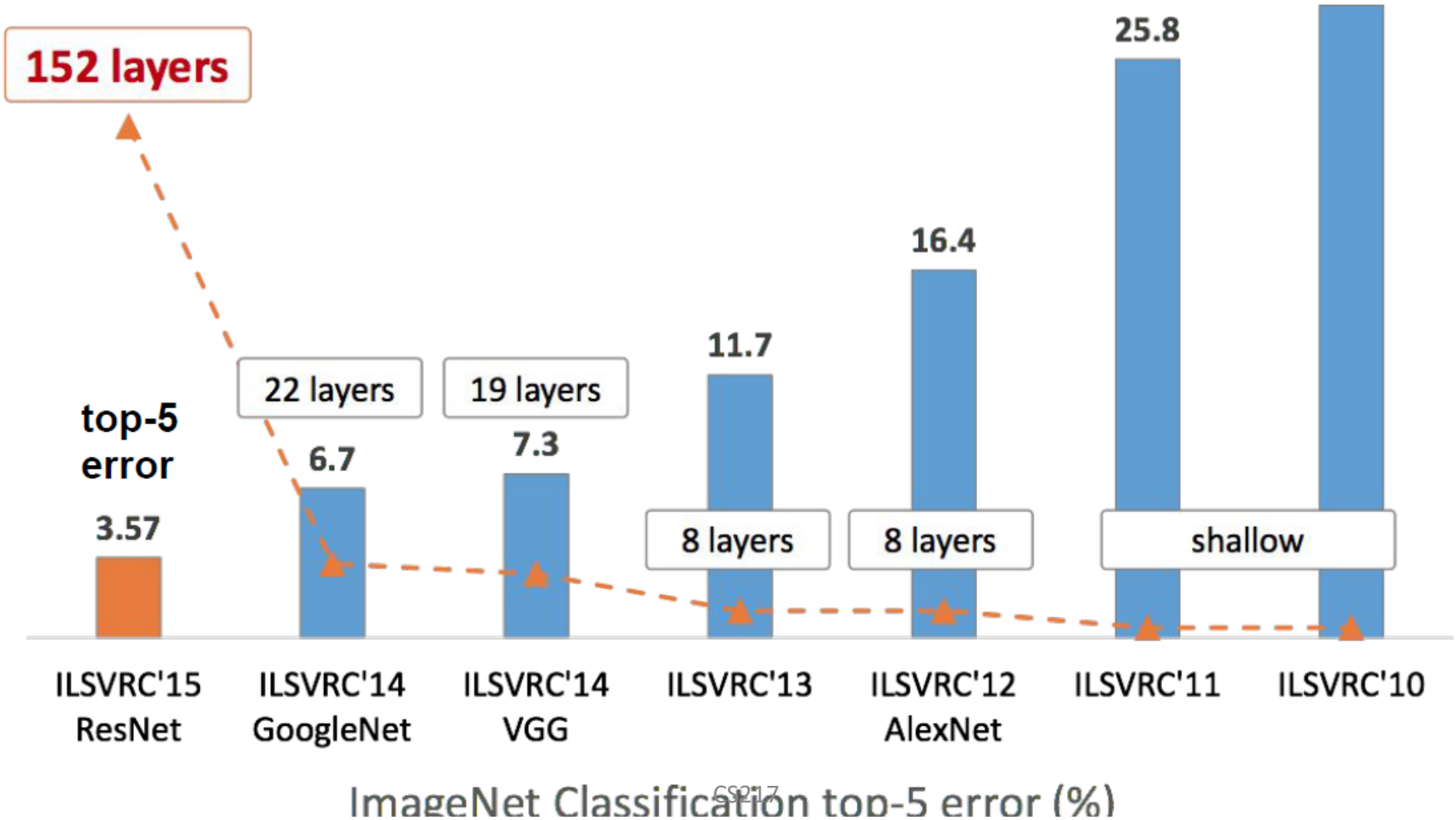
IMAGENET


Canoe
Pill Bottle
Horse-cart
Monkey
Deformability
Skewdriver
Hatchet
Pool Table
Leopard
Amount of Texture
Mug
Tank
Ant
Red Wine
Color Distinctiveness

IMAGENET


Jigsaw Puzzle
Foreland
Lion
Bell
Shape Distinctiveness
Orange
Laptop
Four-poster
Airliner
Real-world Size
Low
High

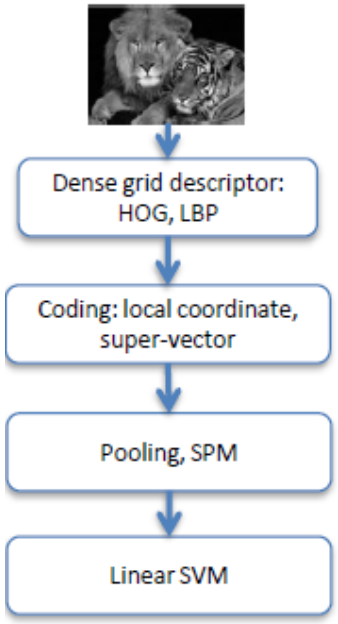
IMAGENET CHALLENGE WINNERS SHALLOW vs. DEEP



DEEP Models Beat Shallow Models

Year 2010

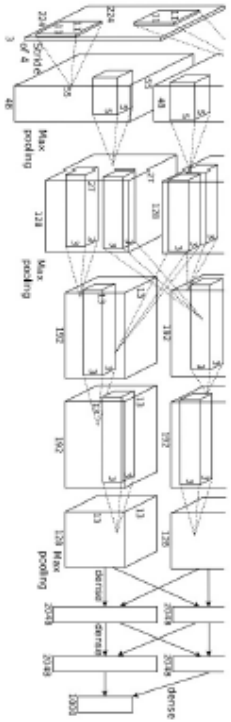
NEC-UIUC



[Lin CVPR 2011]

Year 2012

SuperVision



[Krizhevsky NIPS 2012]

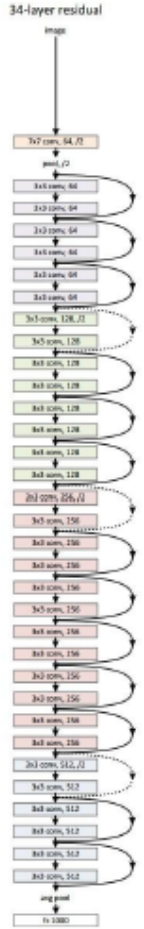
Year 2014

GoogLeNet VGG

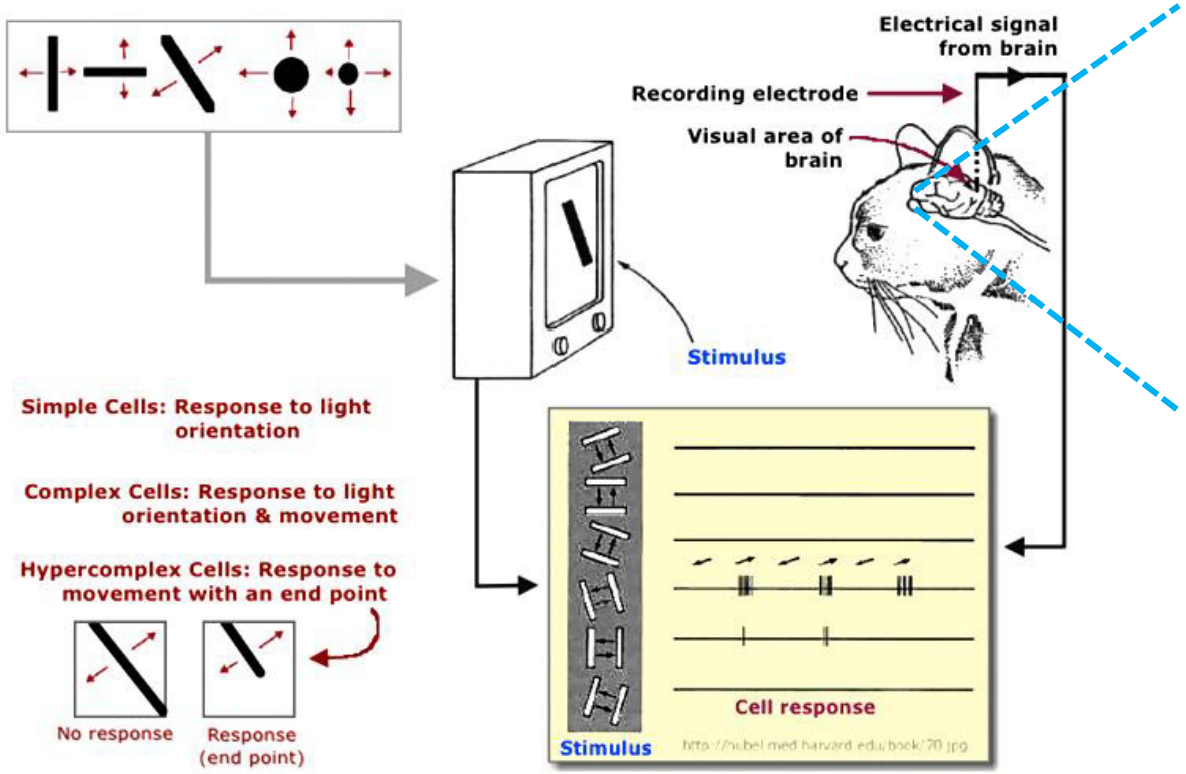
[Szegedy arxiv 2014] [Simonyan arxiv 2014]

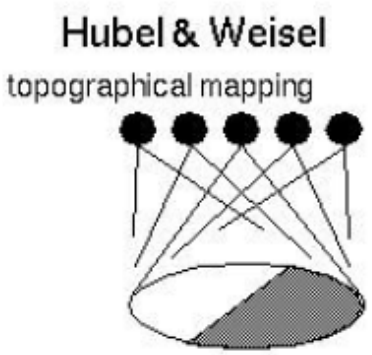
Year 2015

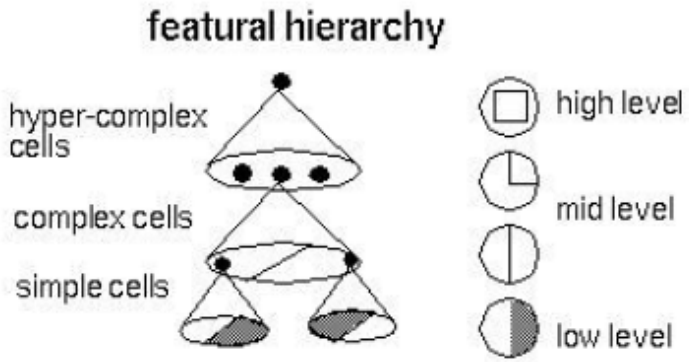
MSRA




WE NEED TO GO
DEEPER
memegenerator.net

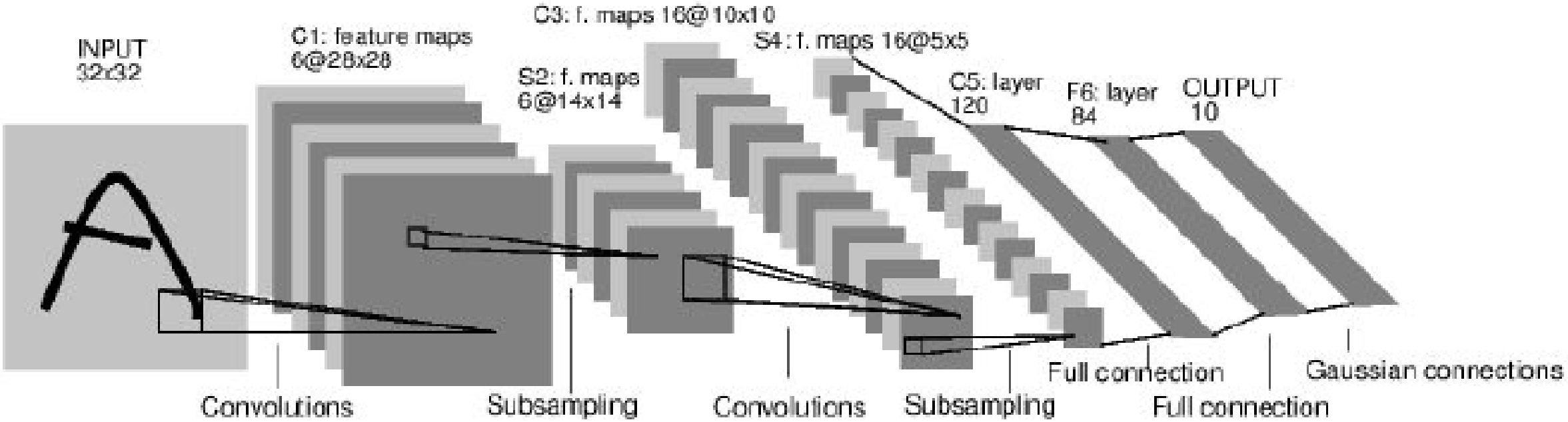




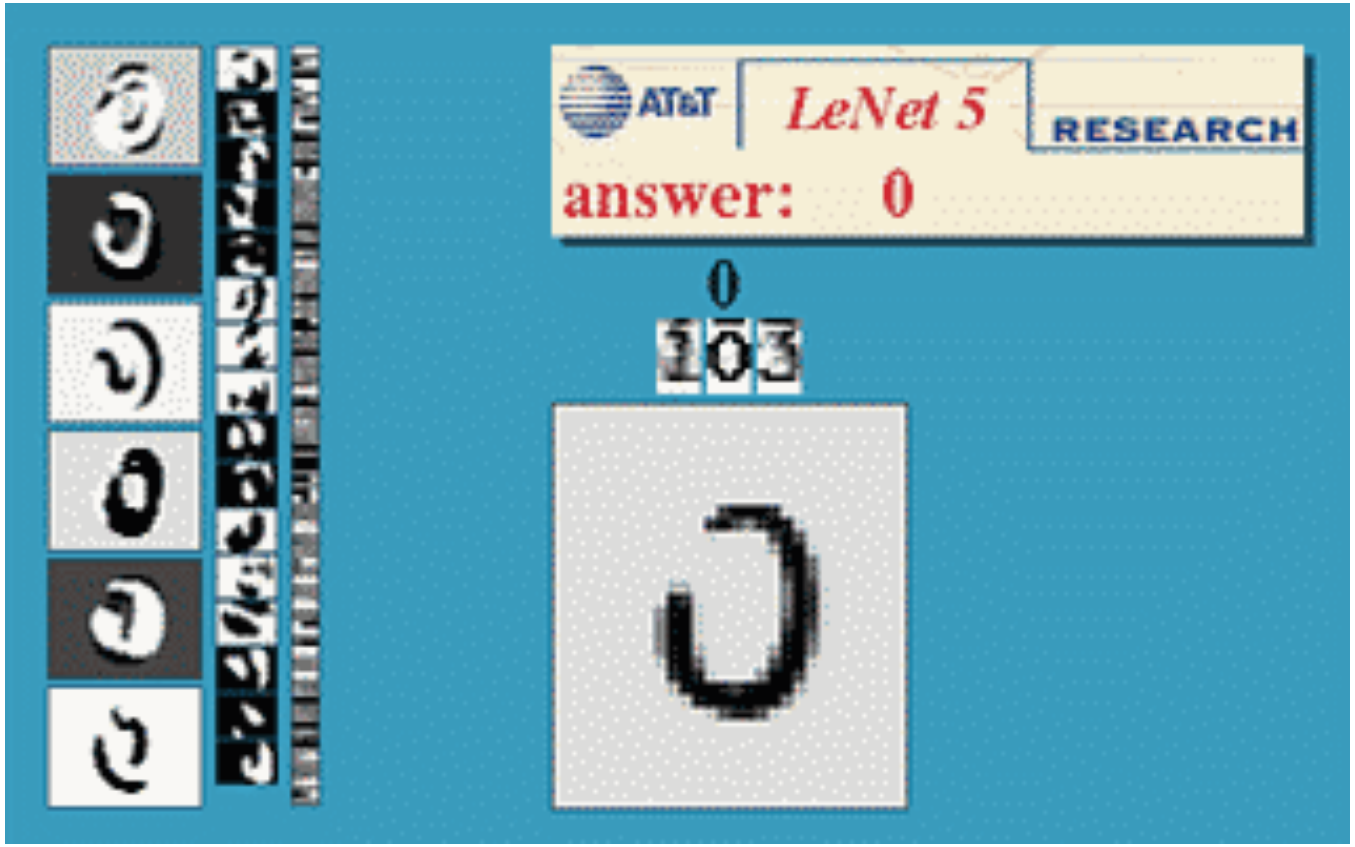


Hubel & Wiesel, 1959

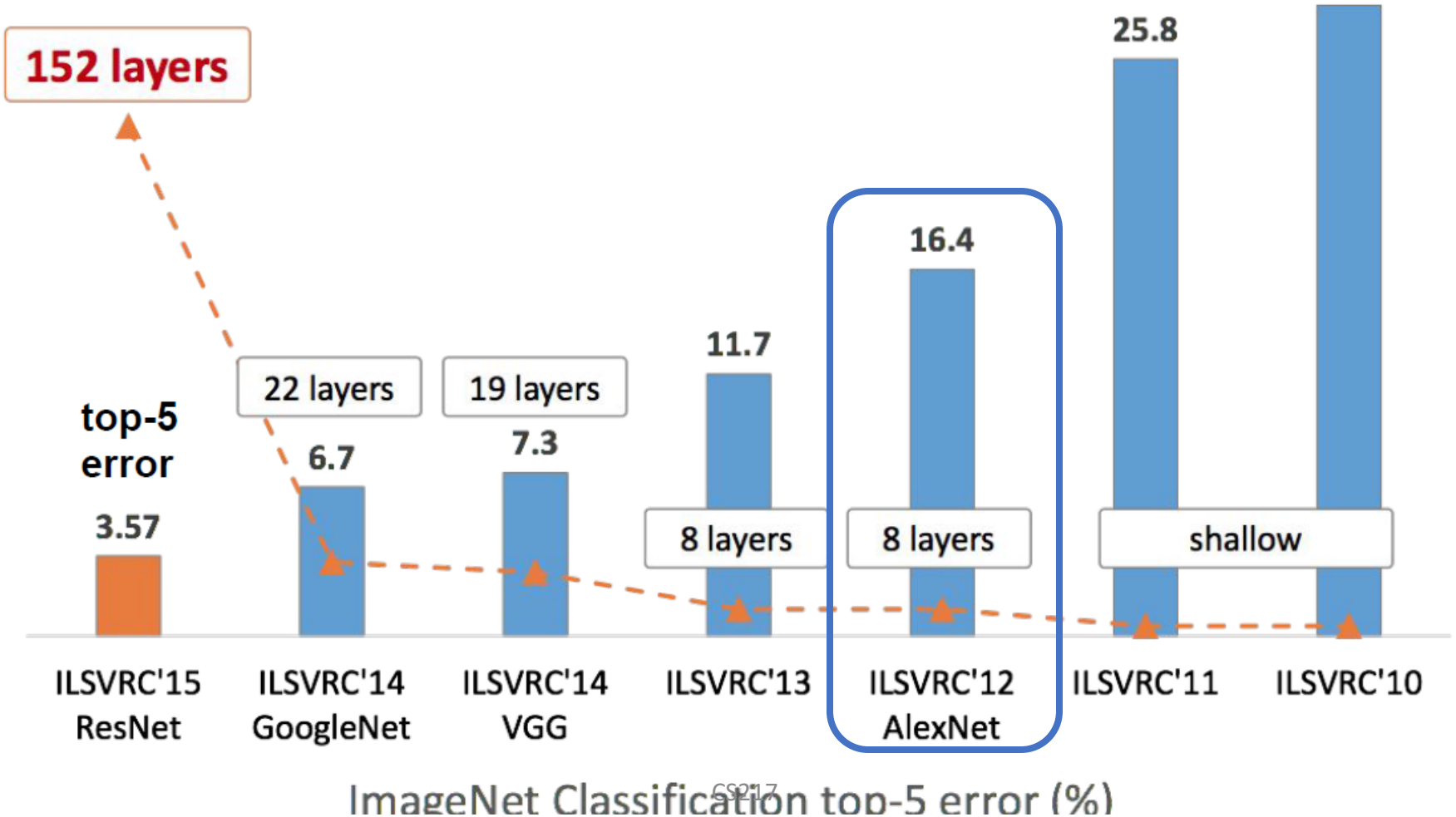
Le-NET 5: 1989


INPUT
32x32
C1: feature maps
6@28x28
S2: f. maps
6@14x14
C3: f. maps 16@10x10
S4: f. maps 16@5x5
C5: layer
120
F6: layer
84
OUTPUT
10
Convolutions
Subsampling
Convolutions
Subsampling
Full connection
Full connection
Gaussian connections

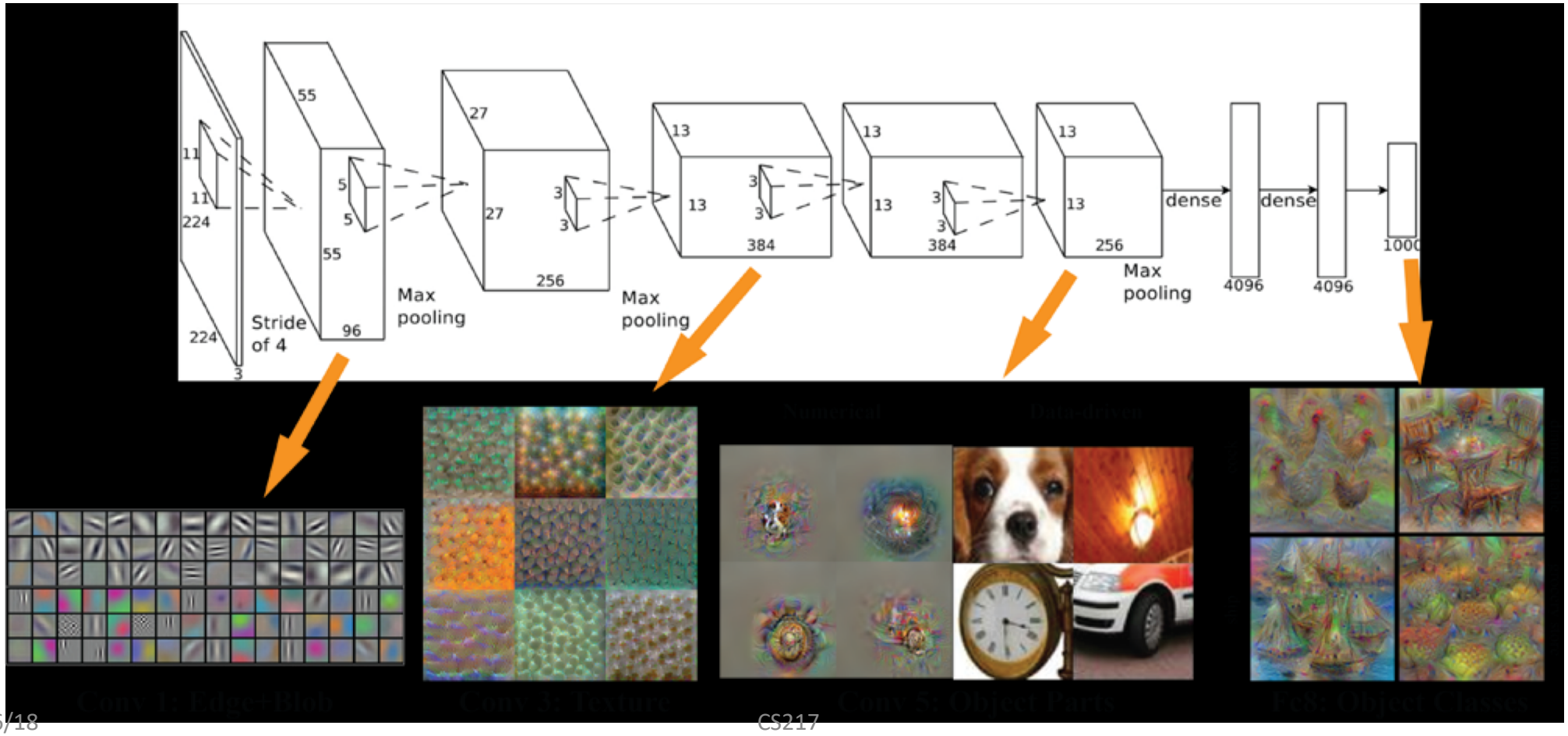
LE-NET 5



IMAGENET CHALLENGE WINNERS SHALLOW vs. DEEP

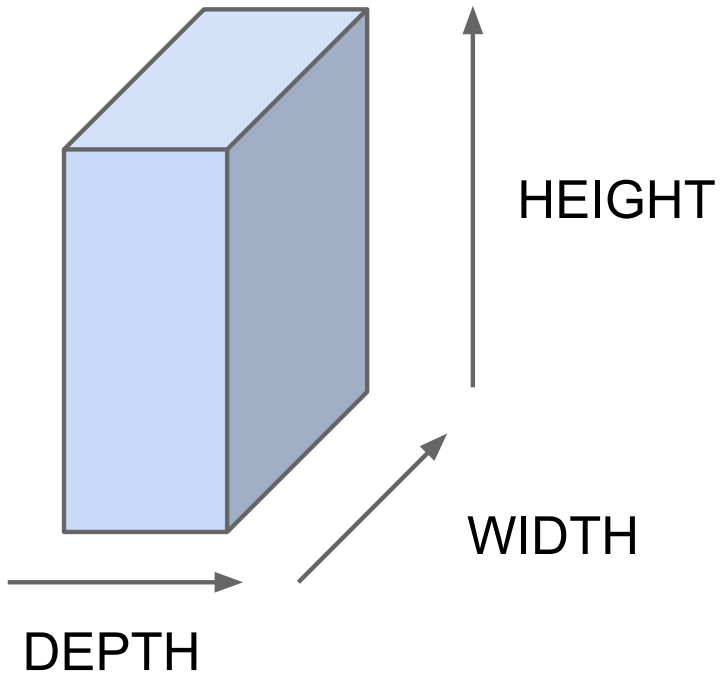


ALEX NET ARCHITECTURE



3D Data

All Neural Net activations arranged in **3 dimensions:**



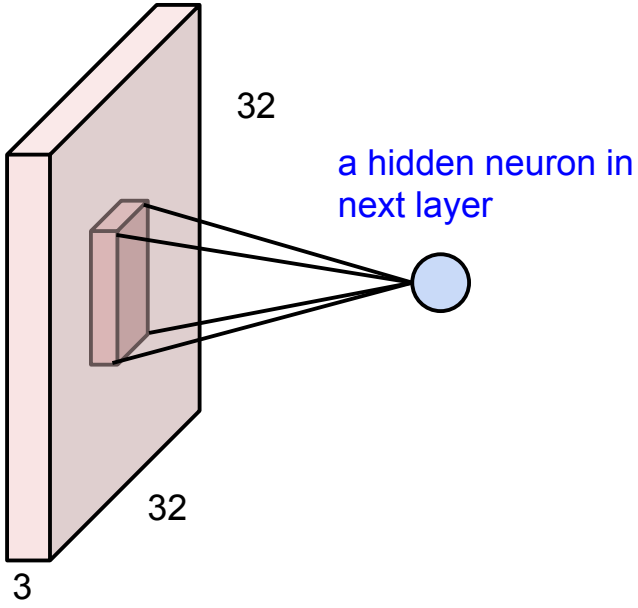
For example, a CIFAR-10 image is a 32x32x3 volume
32 width, 32 height, 3 depth (RGB channels)

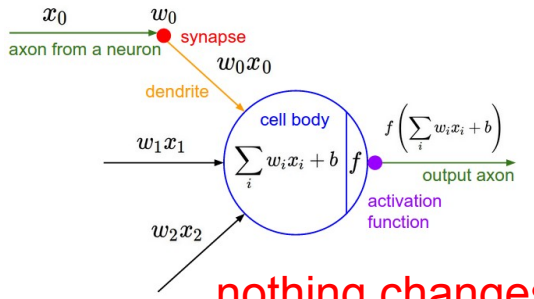
Source: [Fei-Fei Li & Andrej Karpathy]

Local Neighborhoods

Convolutional Neural Networks are just Neural Networks BUT:

1. **Local connectivity**





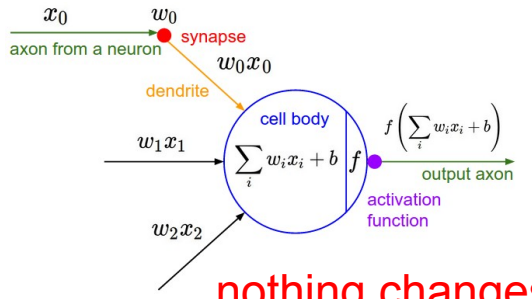
nothing changes

Source: [Fei-Fei Li & Andrej Karpathy]

Local Neighborhoods

Convolutional Neural Networks are just Neural Networks BUT:

1. **Local connectivity**



nothing changes

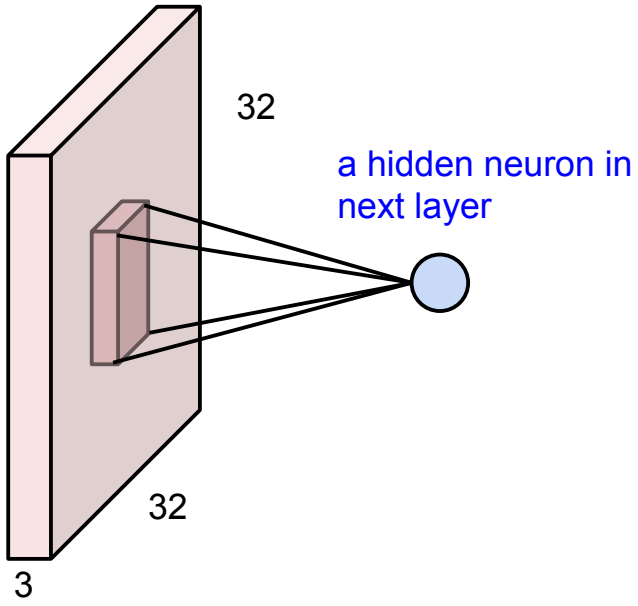


image: 32x32x3 volume
before: full connectivity: 32x32x3 weights

now: one neuron will connect to, e.g. 5x5x3 chunk and only have 5x5x3 weights.

note that connectivity is:

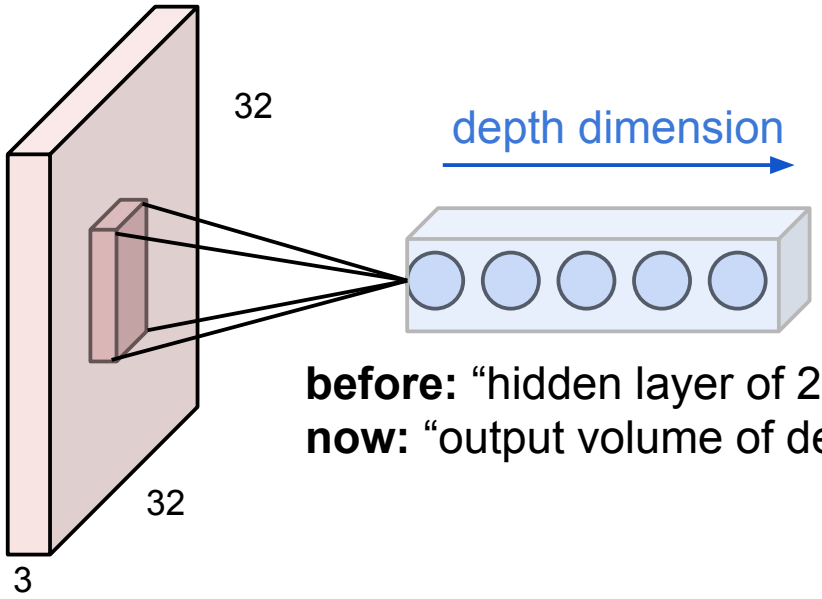
- local in space (5x5 inside 32x32)
- but full in depth (all 3 depth channels)

Source: [Fei-Fei Li & Andrej Karpathy]

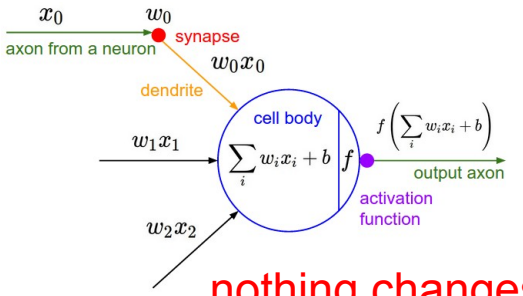
Kernels (Filters) Create Depth

Convolutional Neural Networks are just Neural Networks BUT:

1. **Local connectivity**



before: “hidden layer of 200 neurons”
now: “output volume of depth 200”



nothing changes

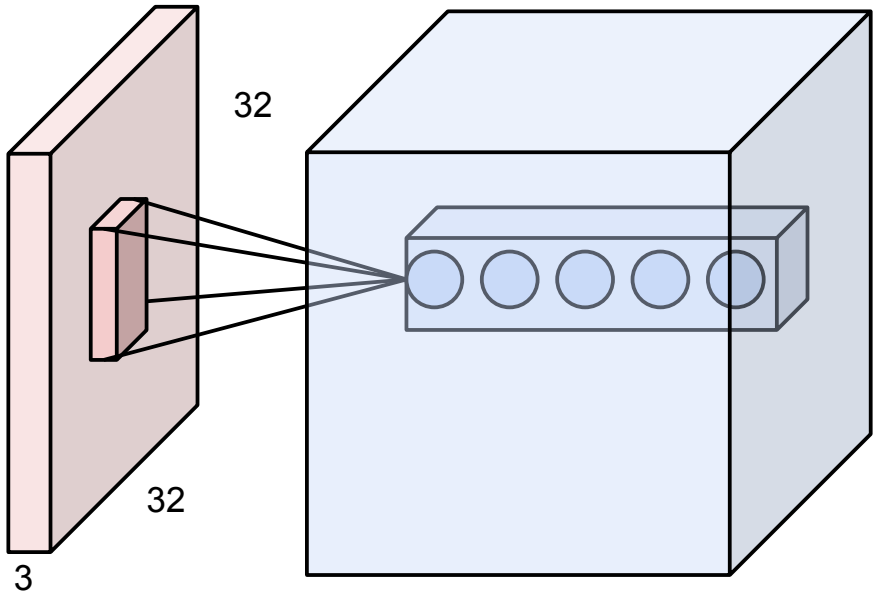
Multiple neurons all looking at the same region of the input volume, stacked along depth.

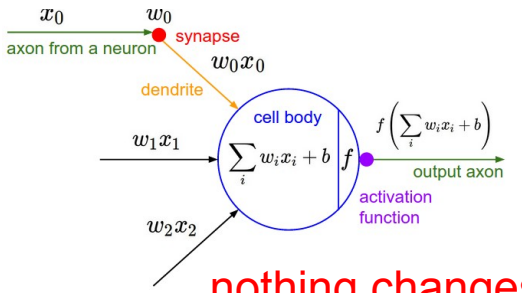
Source: [Fei-Fei Li & Andrej Karpathy]

Kernels (Filters) Create Depth

Convolutional Neural Networks are just Neural Networks BUT:

1. **Local connectivity**





nothing changes

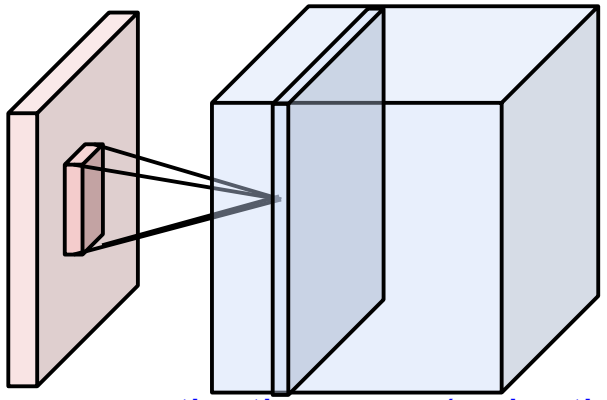
These form a single [1 x 1 x depth] "depth column" in the output volume

Source: [Fei-Fei Li & Andrej Karpathy]

Convolutional Layers

These layers are called **Convolutional Layers**

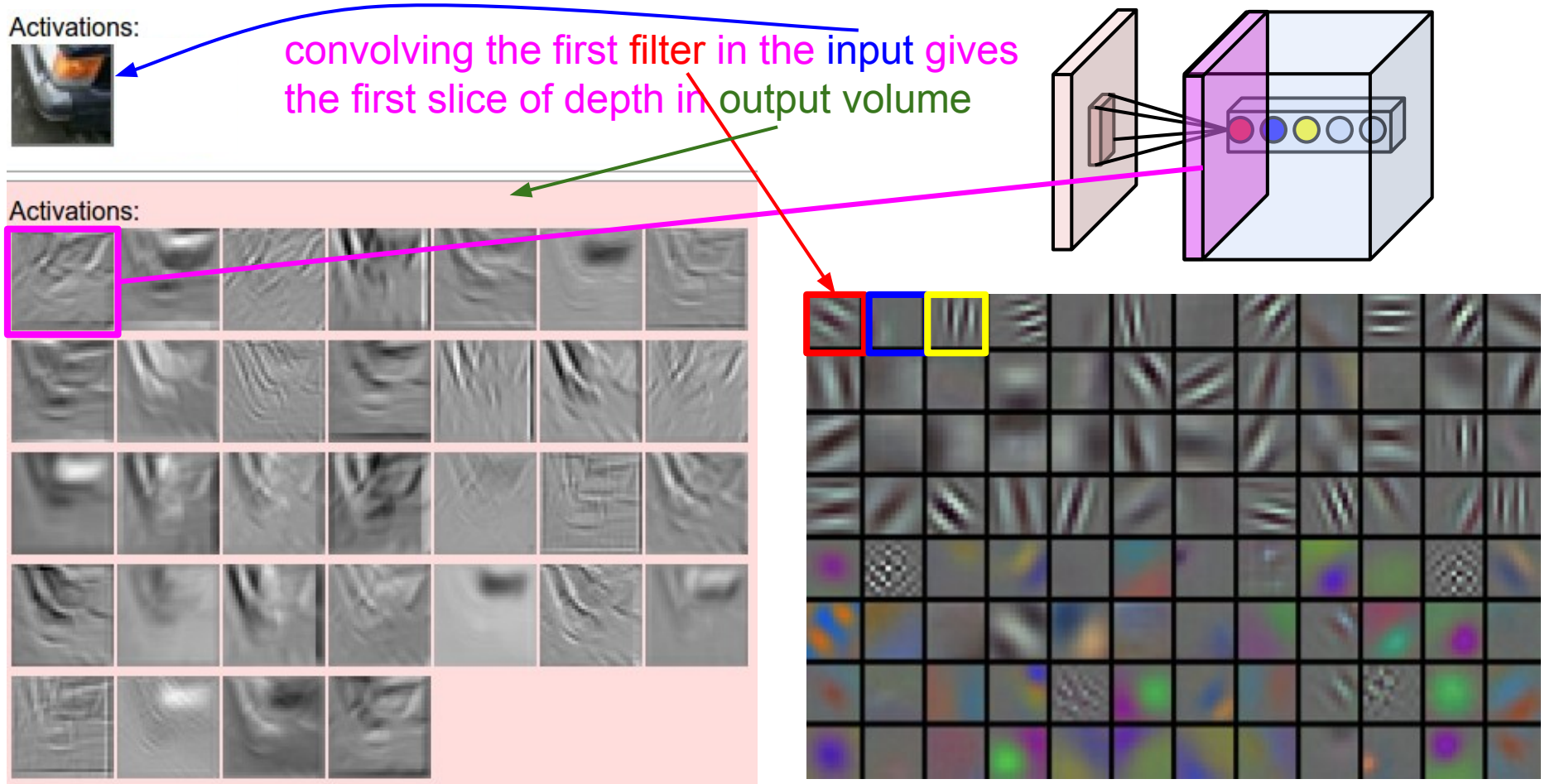
1. Connect neurons only to local receptive fields
2. Use the same neuron weight parameters for neurons in each “depth slice” (i.e. across spatial positions)

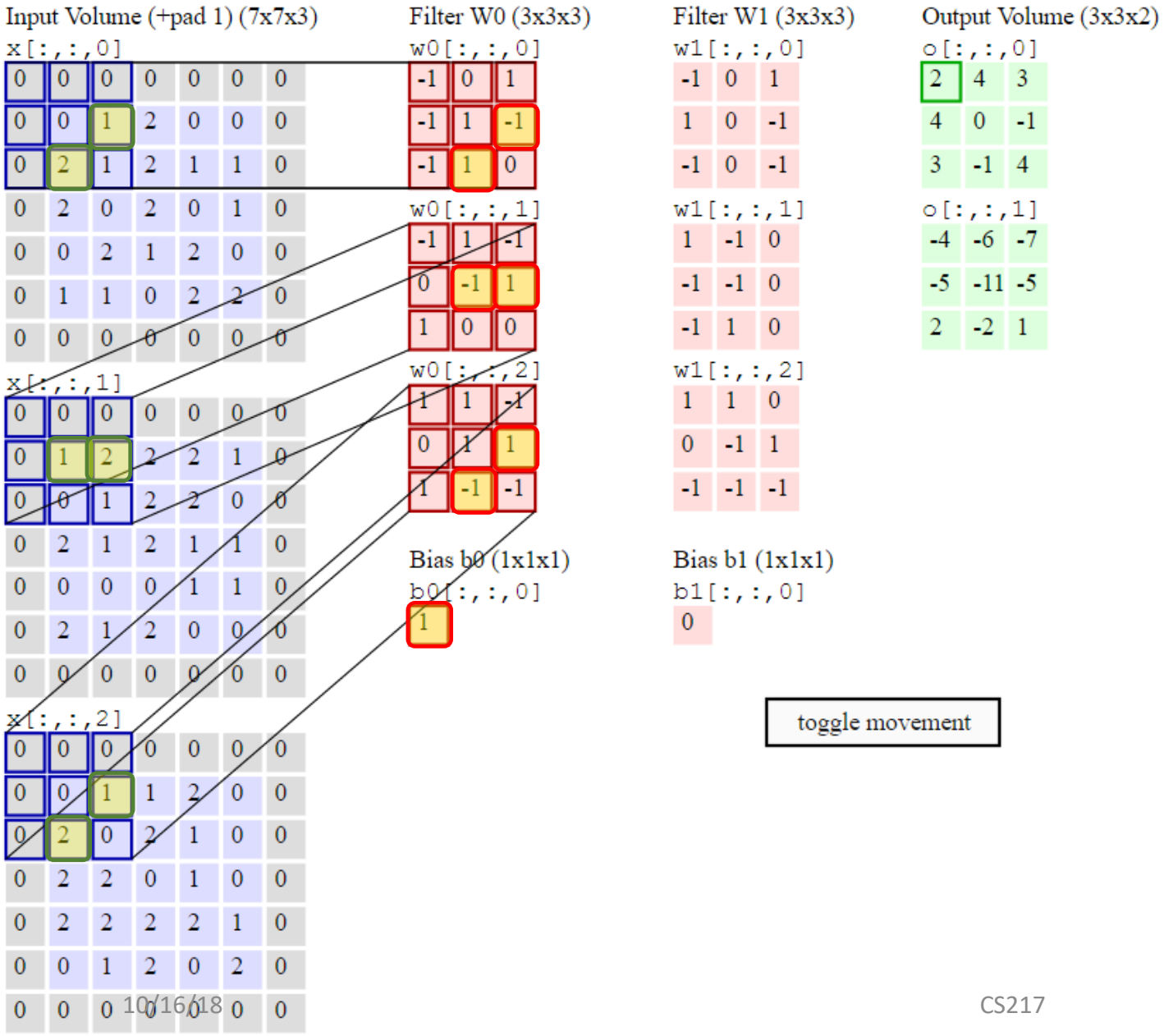


one activation map (a depth slice), computed with one set of weights

Source: [Fei-Fei Li & Andrej Karpathy]

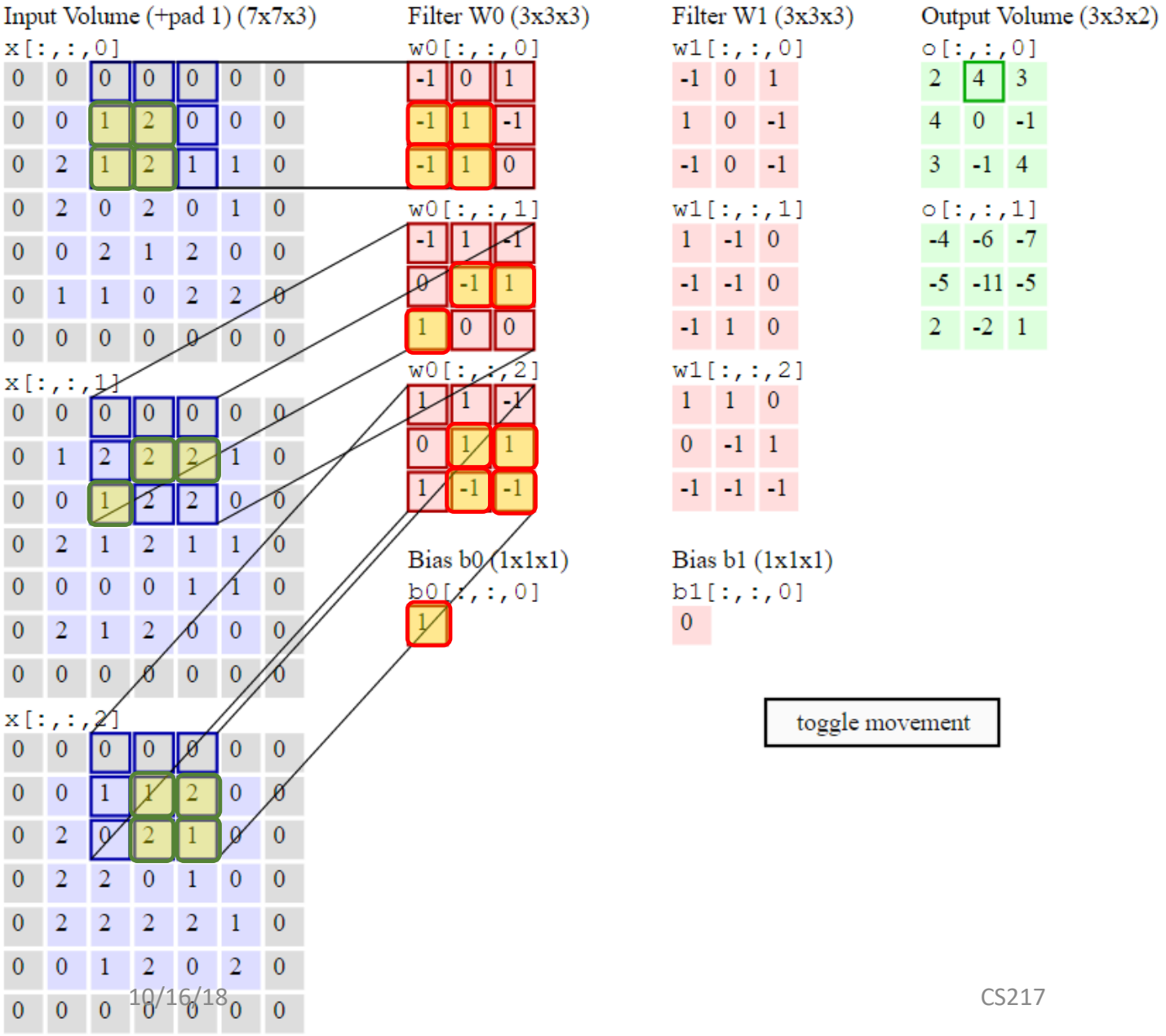
Filters, Convolutions, Depth





- Input 7x7 with 3 Channels
- Kernels 3x3 with 3 Channels
- 2 Kernels (Filters) W0 and W1
- Output has 2 Channels (# of Kernels)

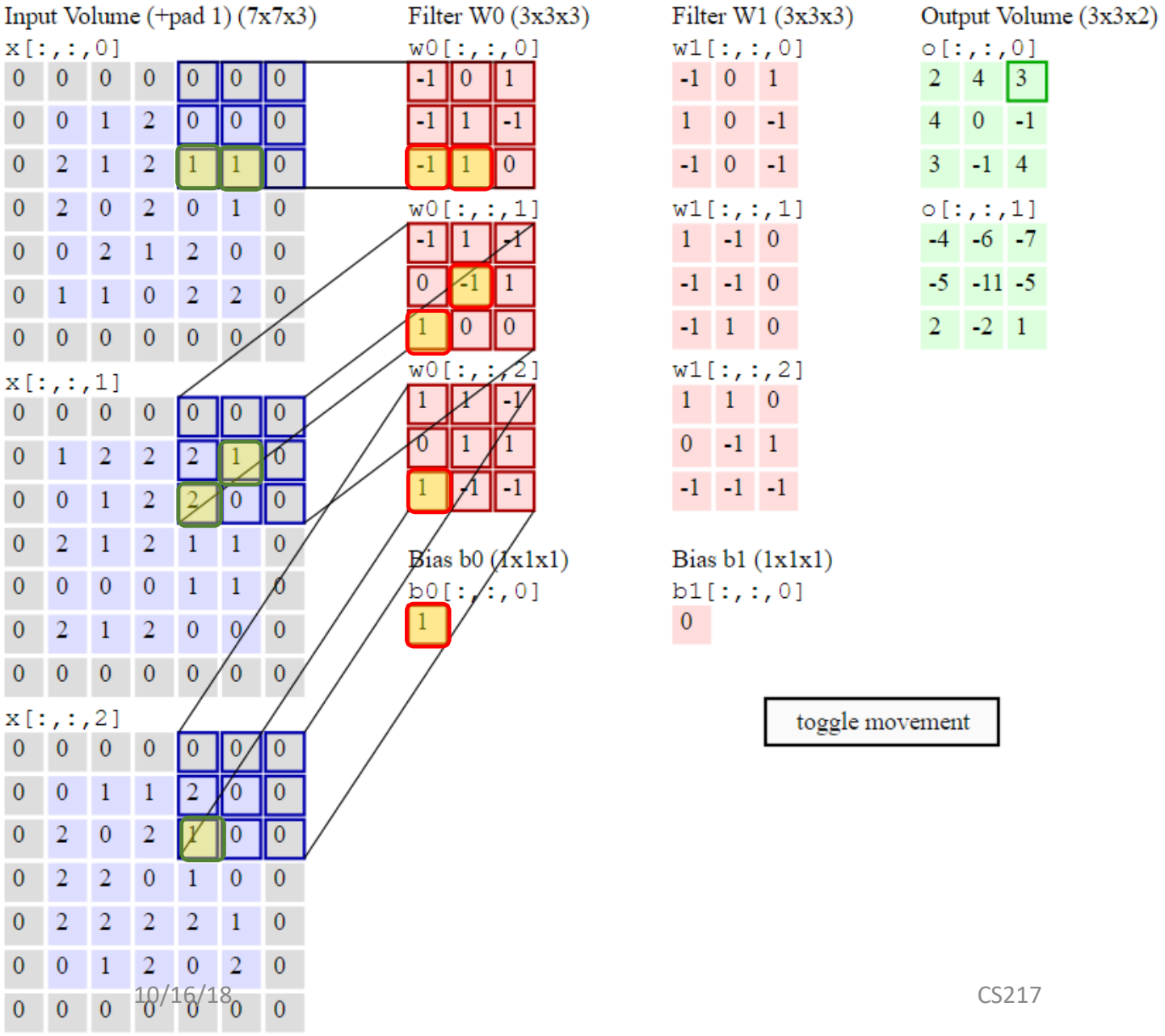
(1x-1+2x1)+
(1x-1+2x1)+
(1x1+2x-1)+
1
=2



Example

- Input 7x7 with 3 Channels
- Kernels 3x3 with 3 Channels
- 2 Kernels (Filters) W0 and W1
- Output has 2 Channels (# of Kernels)

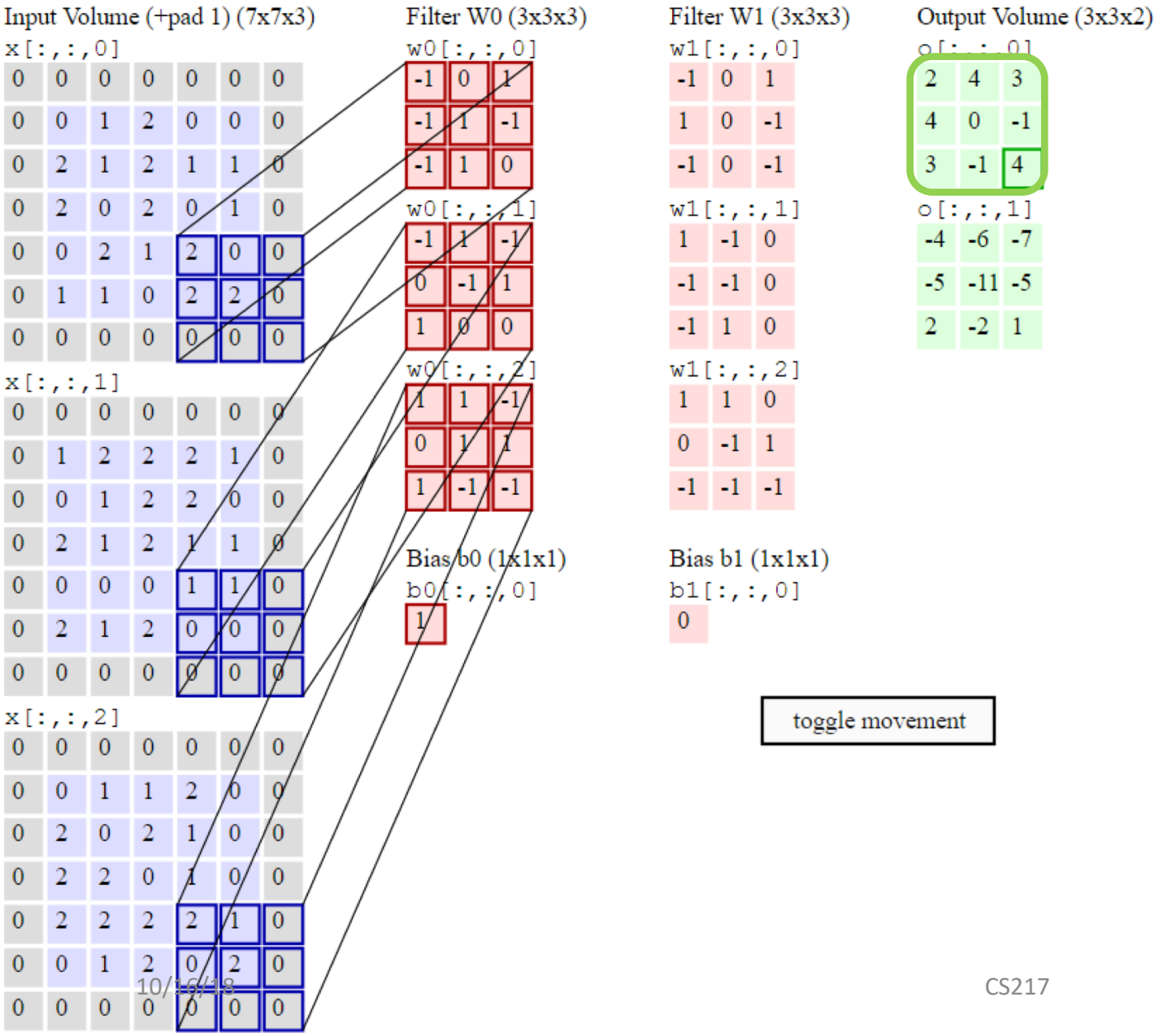
(1x-1+2x1+1x-1+2x1)+
(2x1+2x1+1x1+2x-1+2x-1)+
(1x1+2x1+2x-1+1x-1)+
1
=4



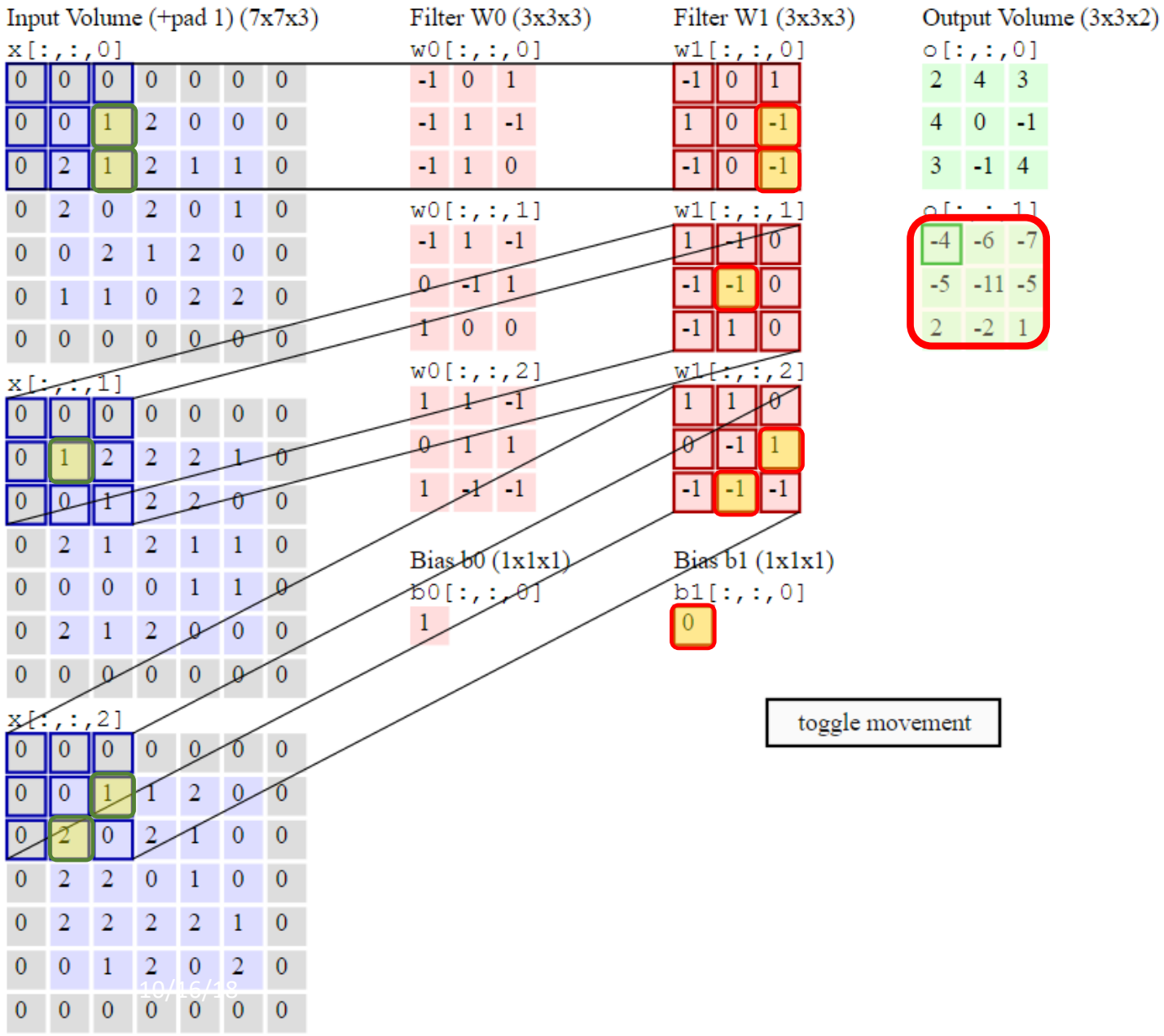
Example

- Input 7x7 with 3 Channels
- Kernels 3x3 with 3 Channels
- 2 Kernels (Filters) W0 and W1
- Output has 2 Channels (# of Kernels)

(1x-1+1x1)+
(1x-1+2x1)+
(1x1)+
1
=3



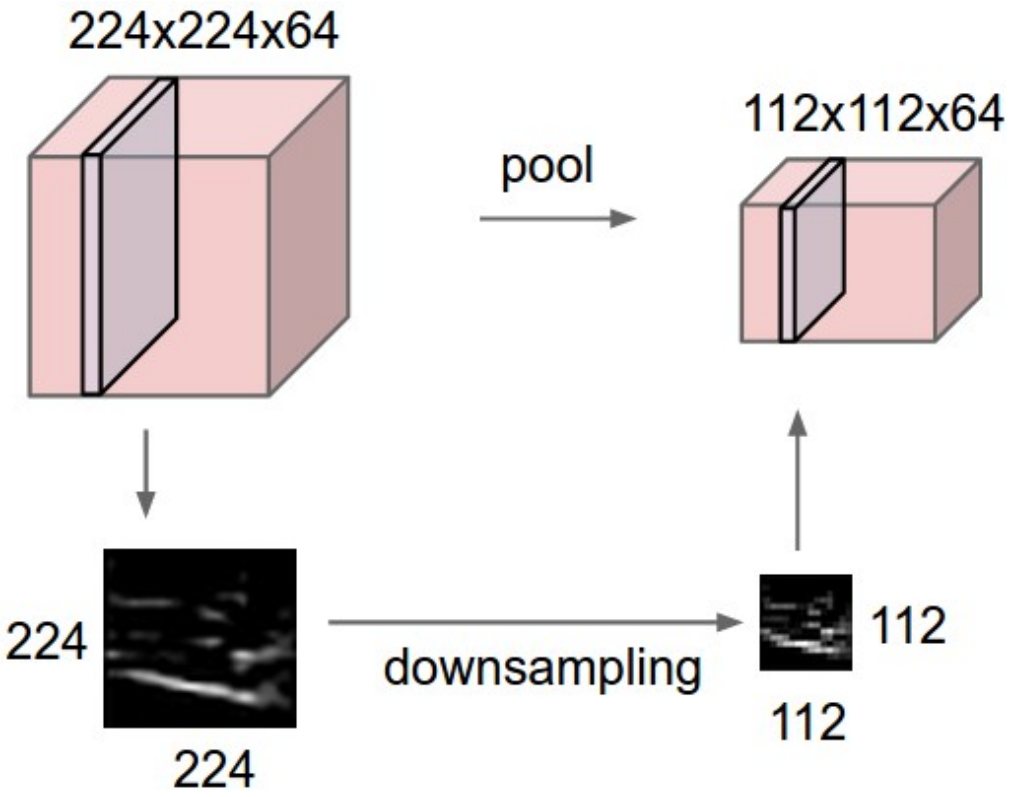
- Computed 1st Channel of Output
- Use Filter W1 with the same input volume

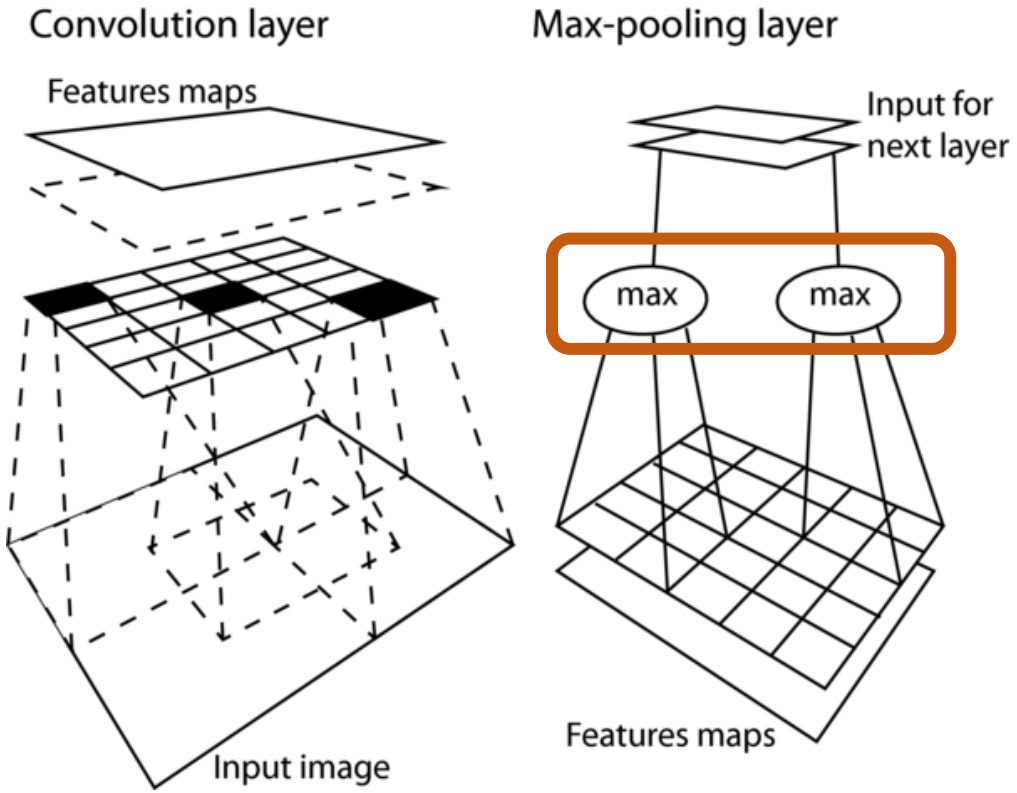


- Input 7x7 with 3 Channels
- Kernels 3x3 with 3 Channels
- Now with W1
- Output's 2nd Channel

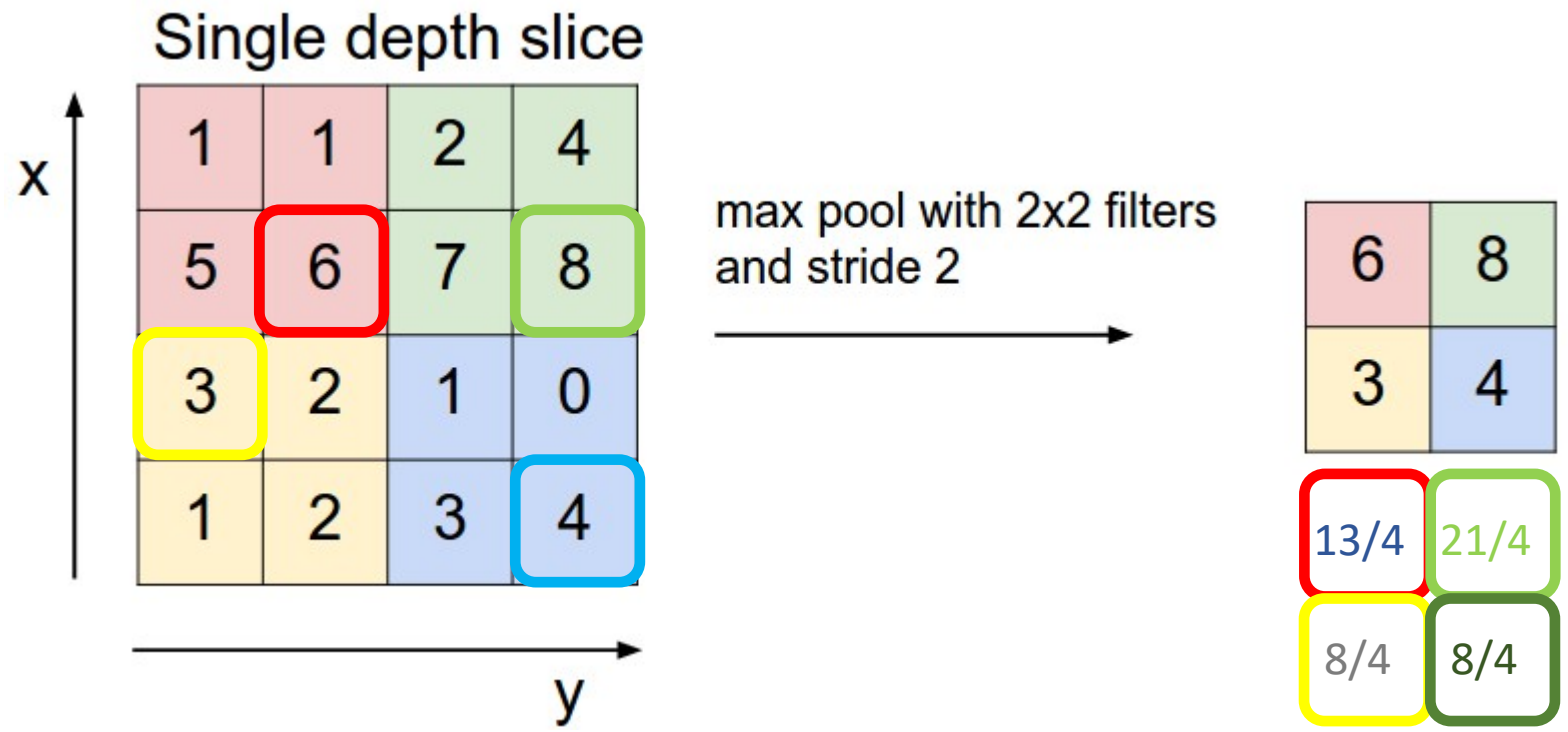
(1x-1+1x-1)+
(1 -1)+
(1x1+2x-1)+
0
=-4

Pooling Layer

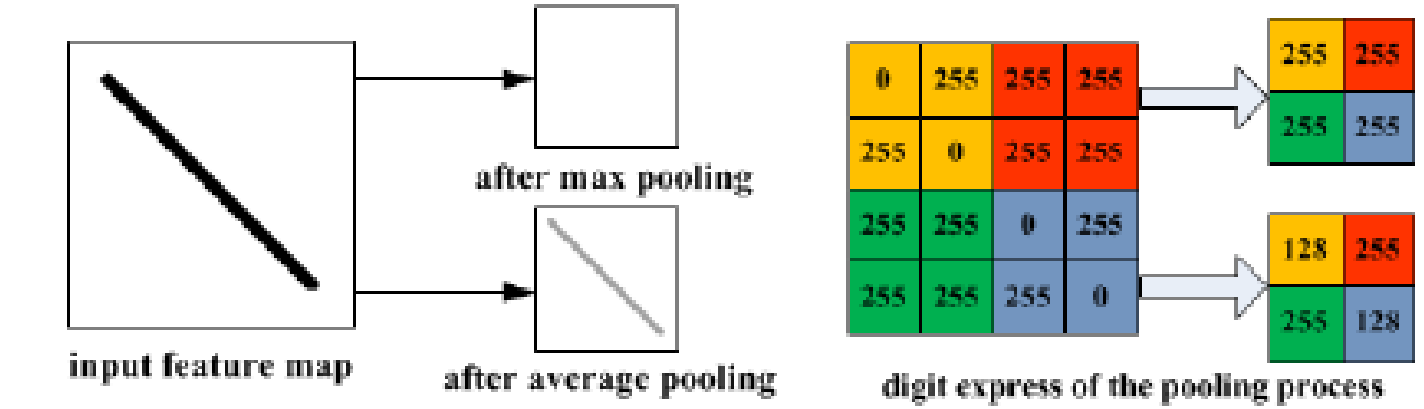




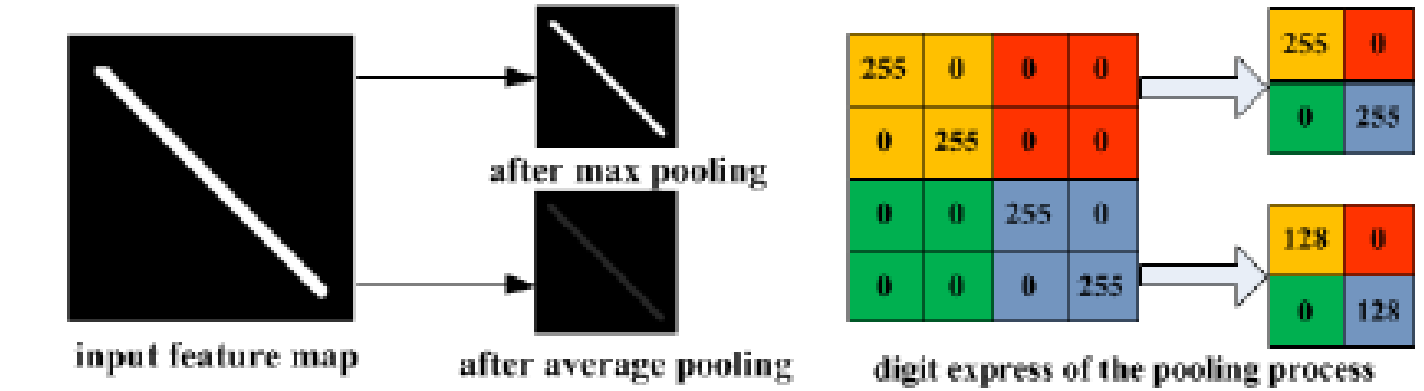
Max Pooling


Single depth slice
x
y
1 1 2 4
5 6 7 8
3 2 1 0
1 2 3 4
max pool with 2x2 filters
and stride 2
6 8
3 4
13/4 21/4
8/4 8/4

Pooling



(a) Illustration of max pooling drawback

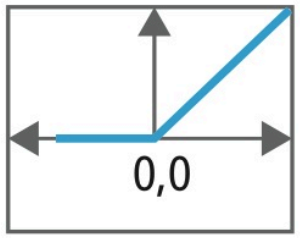


(b) Illustration of average pooling drawback

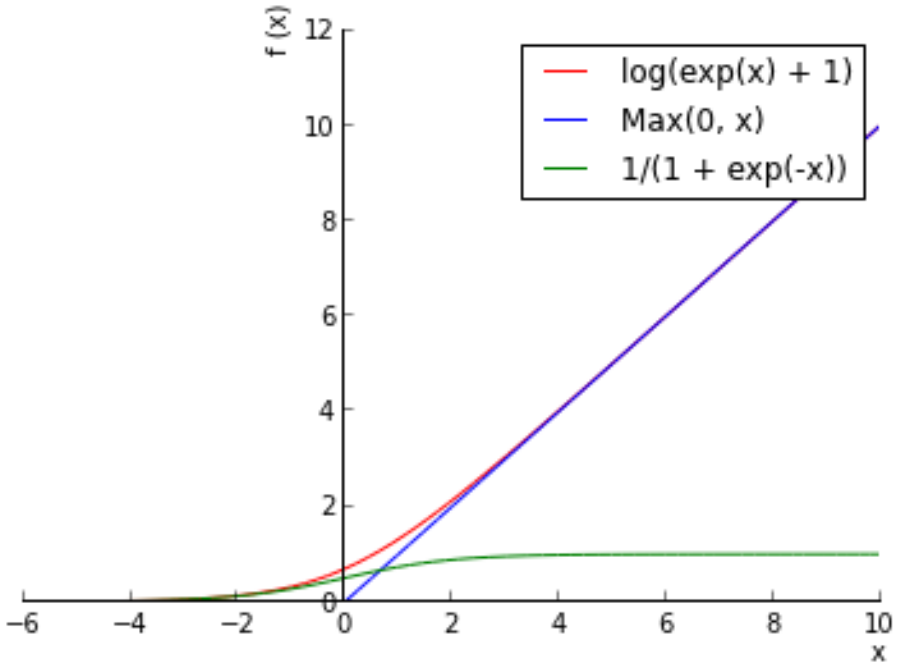
Rectified Linear Units

15	20	-10	35
18	-110	25	100
20	-15	25	-10
101	75	18	23

Transfer Function



15	20	0	35
18	0	25	100
20	0	25	0
101	75	18	23

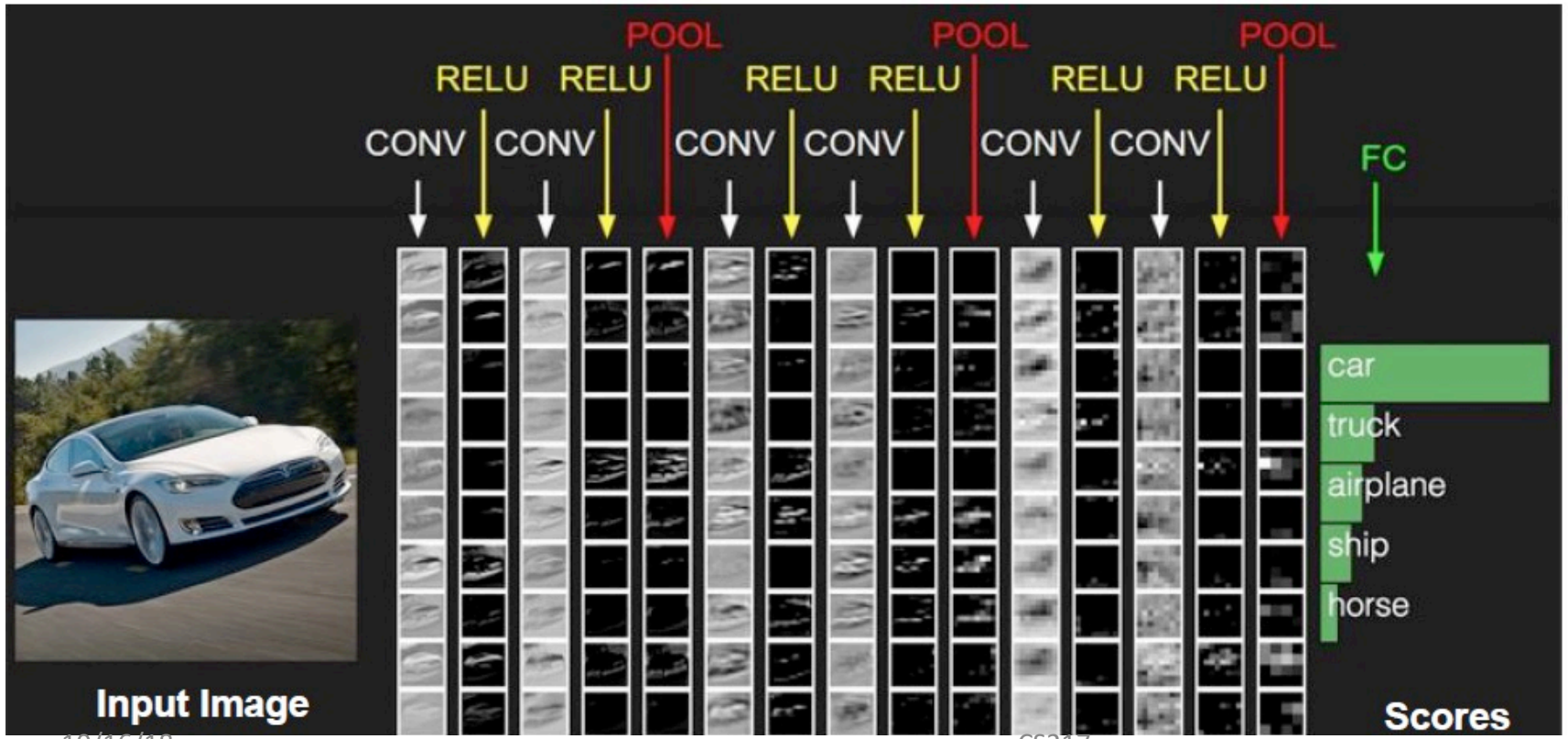


EASY……

RIGHT?

Conv 3x3s1, 10 / ReLU

Type: Conv **Kernel Size**: 3x3 **Stride**: 1 **Channels**:10 **Activation**: ReLU



FC 10

Max Pool 3x3s1

Conv 3x3s1, 10 / ReLU

Conv 3x3s1, 10 / ReLU

Max Pool 3x3s1

Conv 3x3s1, 10 / ReLU

Conv 3x3s1, 10 / ReLU

Max Pool 3x3s1

Conv 3x3s1, 10 / ReLU

Conv 3x3s1, 10 / ReLU

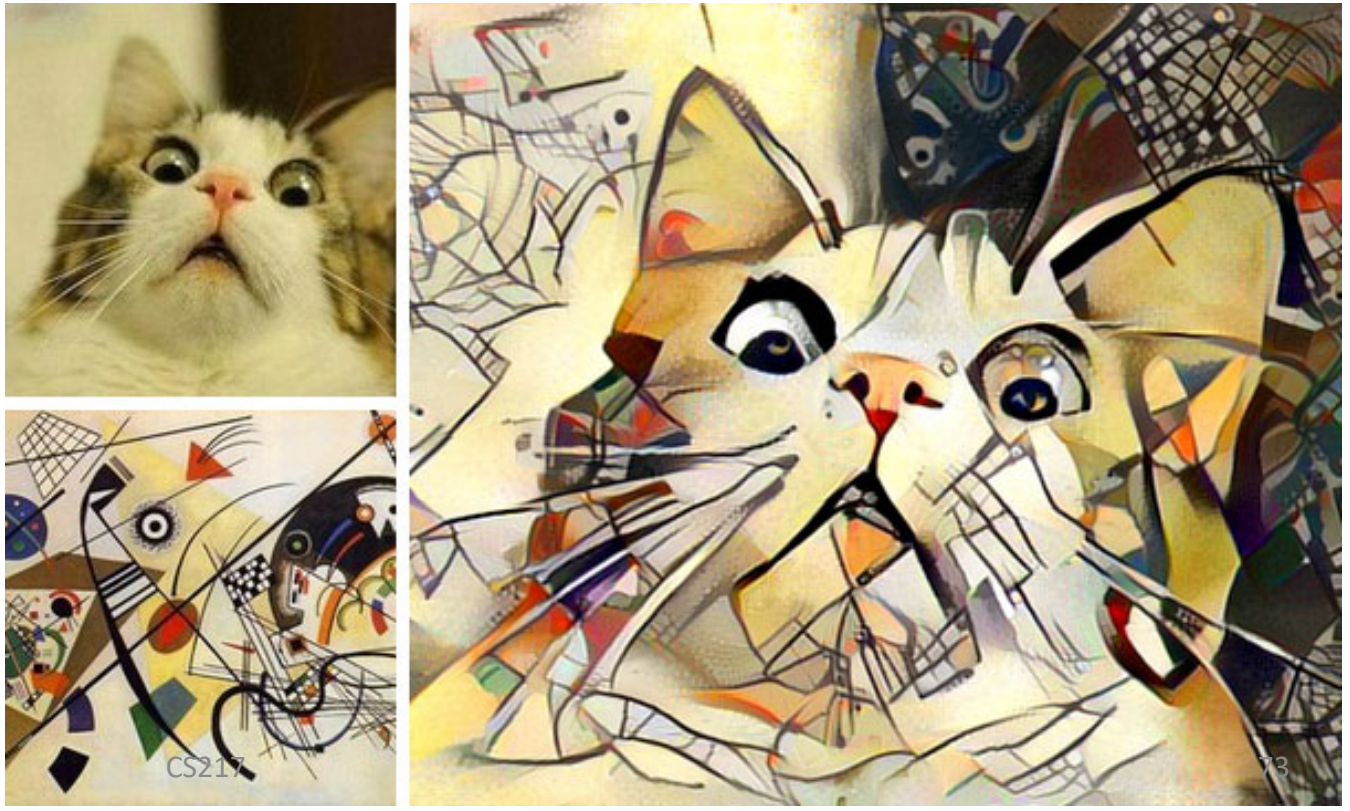
DEMO

Classify **MNIST** digits with a Convolutional Neural Network

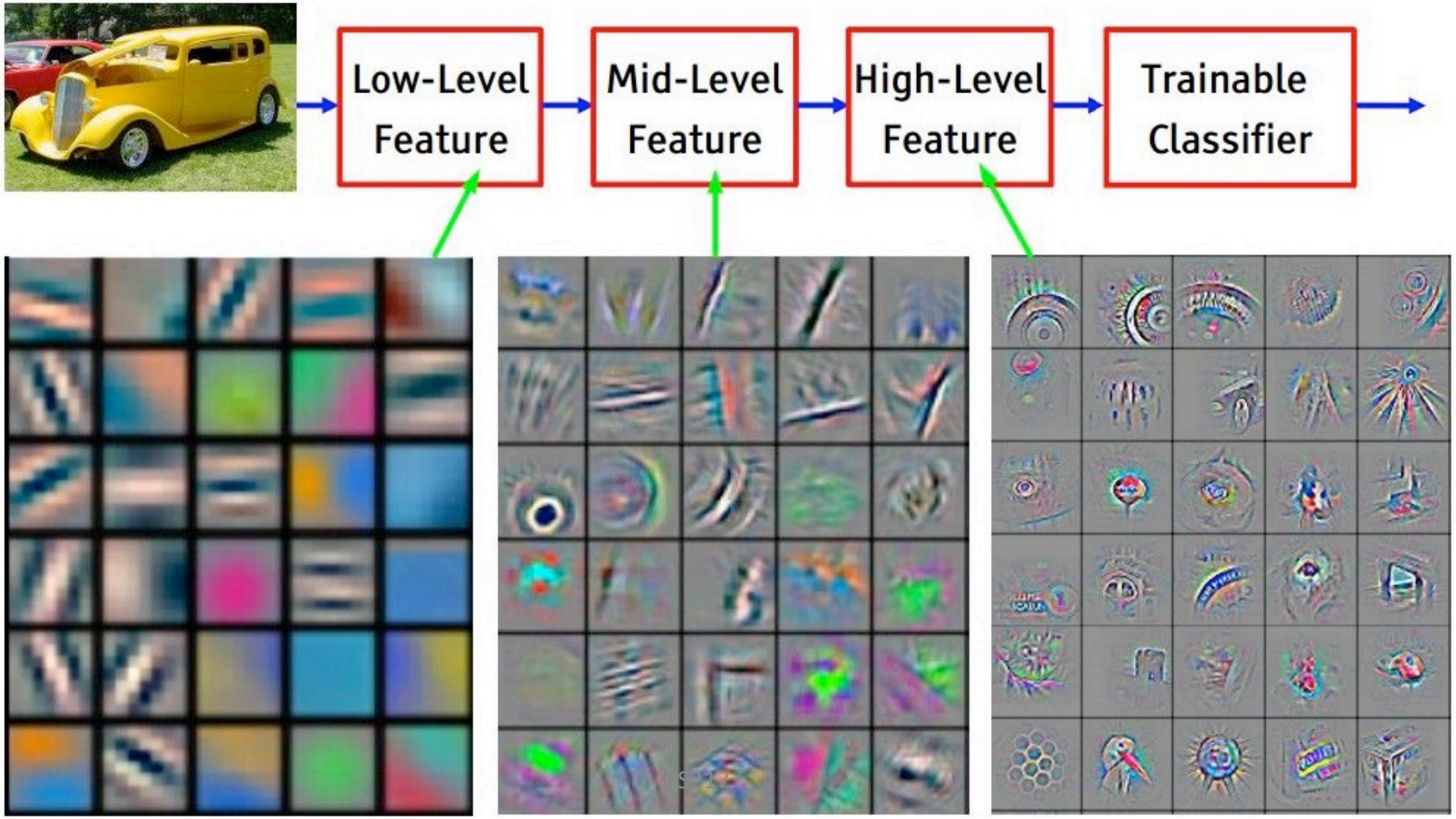
Classify **CIFAR-10** with Convolutional Neural Network

- http://cs.stanford.edu/people/karpathy/convnetjs/

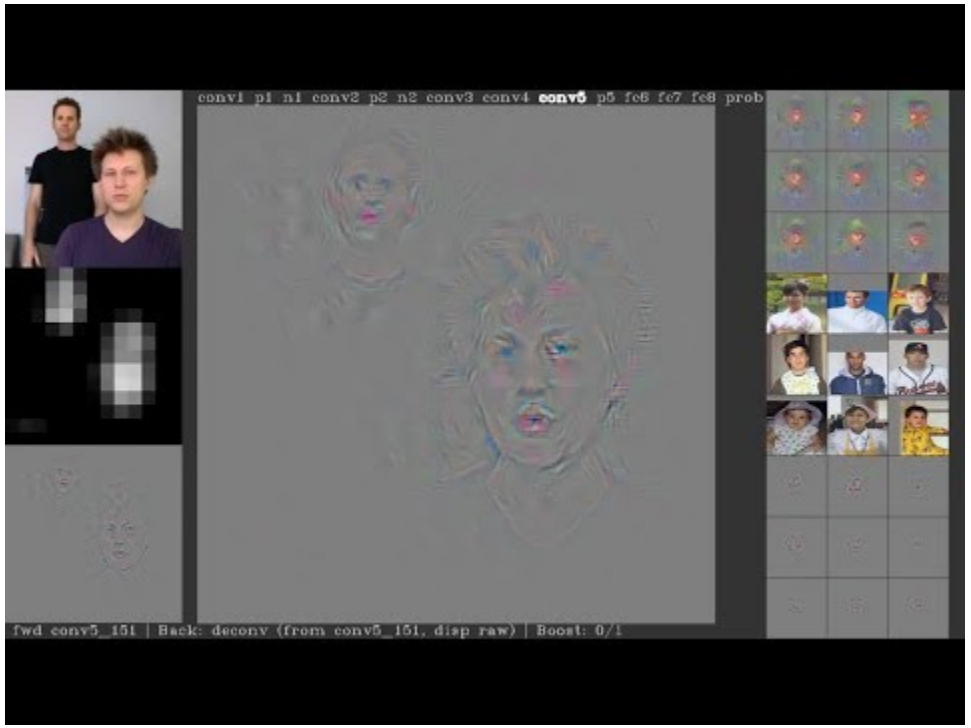
Next:
Visualizing CNNs and more



Features and Layers



Deep Visualization

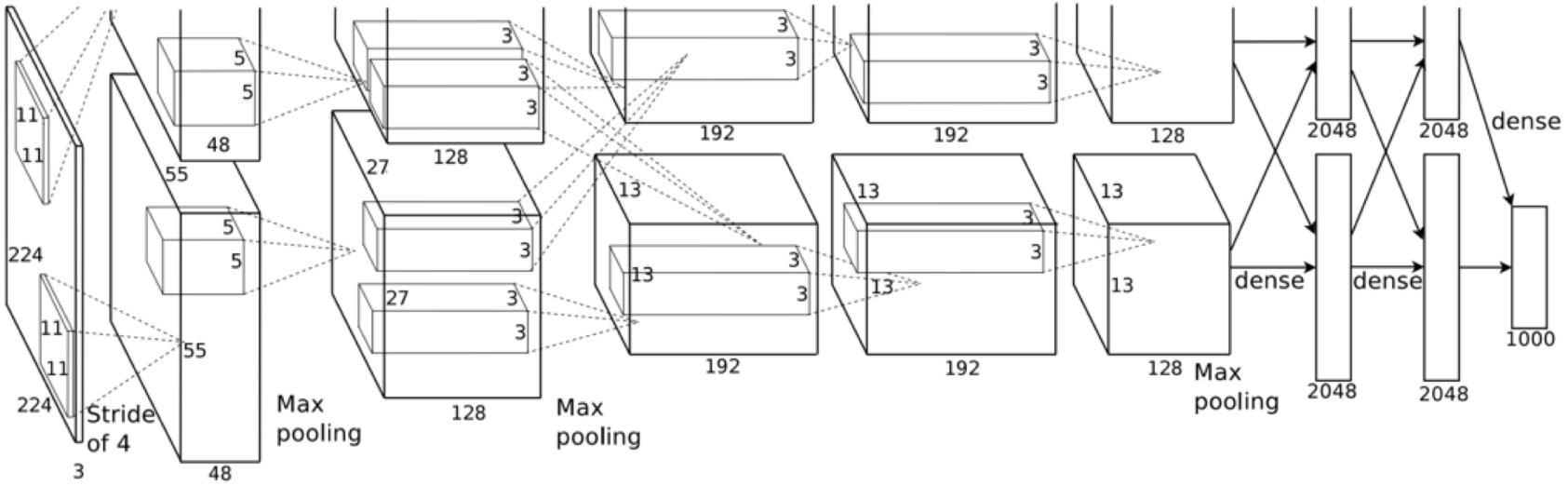


http://yosinski.com/deepvis

- Watch the video for a nice recap of CNNs and what we learnt so far

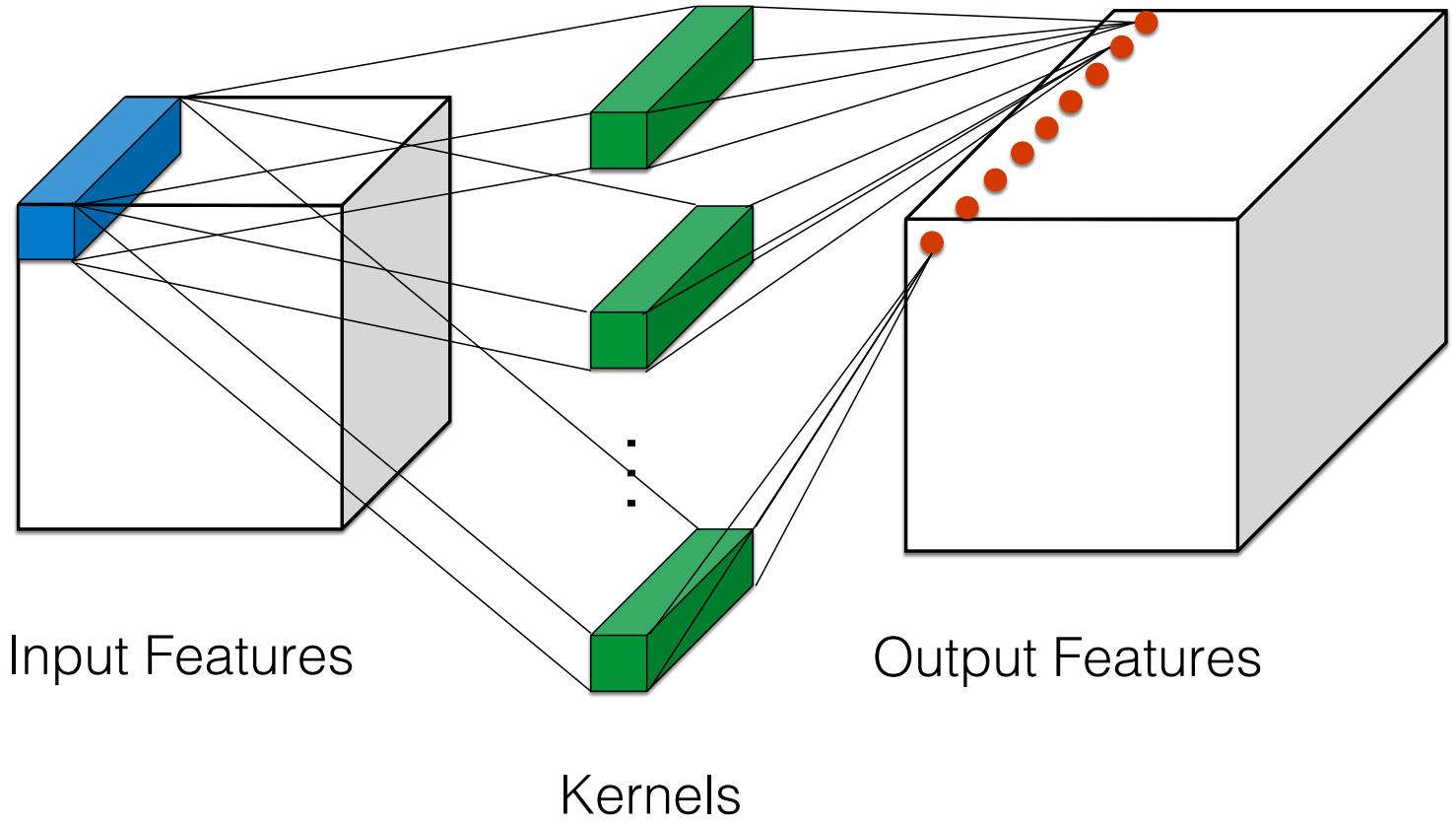
- https://archive.org/details/Redwood_Center_2016_03_01_Leon_Gatys

- Four dimensions in each layer
  - Image Dimensions (X,Y)
  - # Channels (C)
  - # Kernels (K)
  - Filter dimensions (F_h, F_w)



Alex Krizhevsky, Ilya Sutskever, Geoffrey E. Hinton, ImageNet Classification with Deep Convolutional Neural Networks

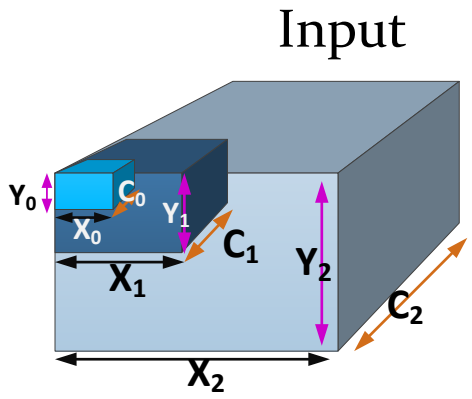
Convolution Layer



Optimize CNNs for Hardware

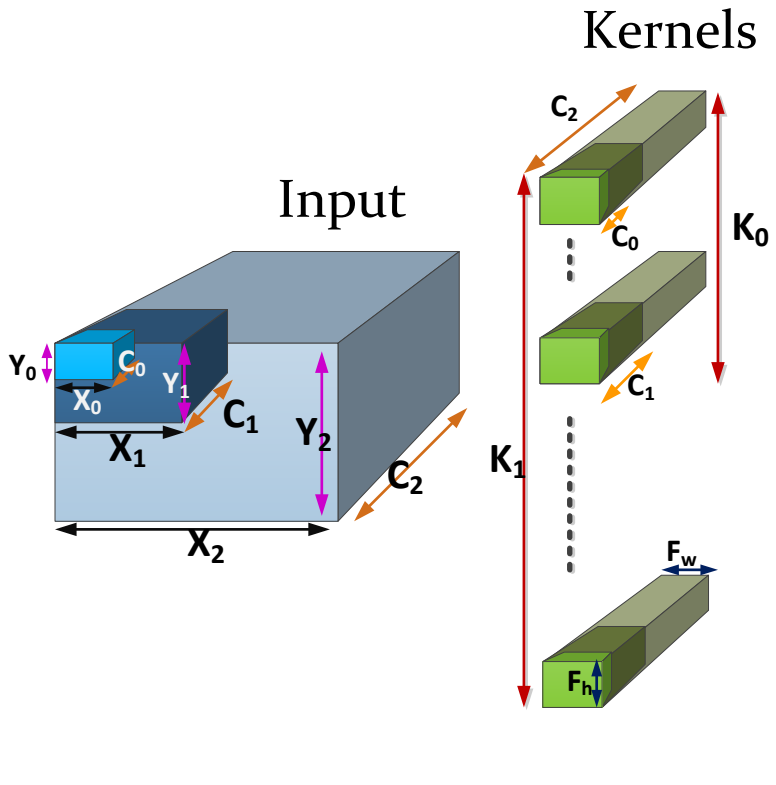
- Turn it to GEMM/GEMV?
- Design Accelerators
- Take advantage of Locality
- A desired Datapath
  - Data Movement
  - PEs
  - Systolic/Non Systolic
- Take advantage of sparsity
- Change Nature of Computation (Winograd)

Multilevel Blocking of Convolutional Layer



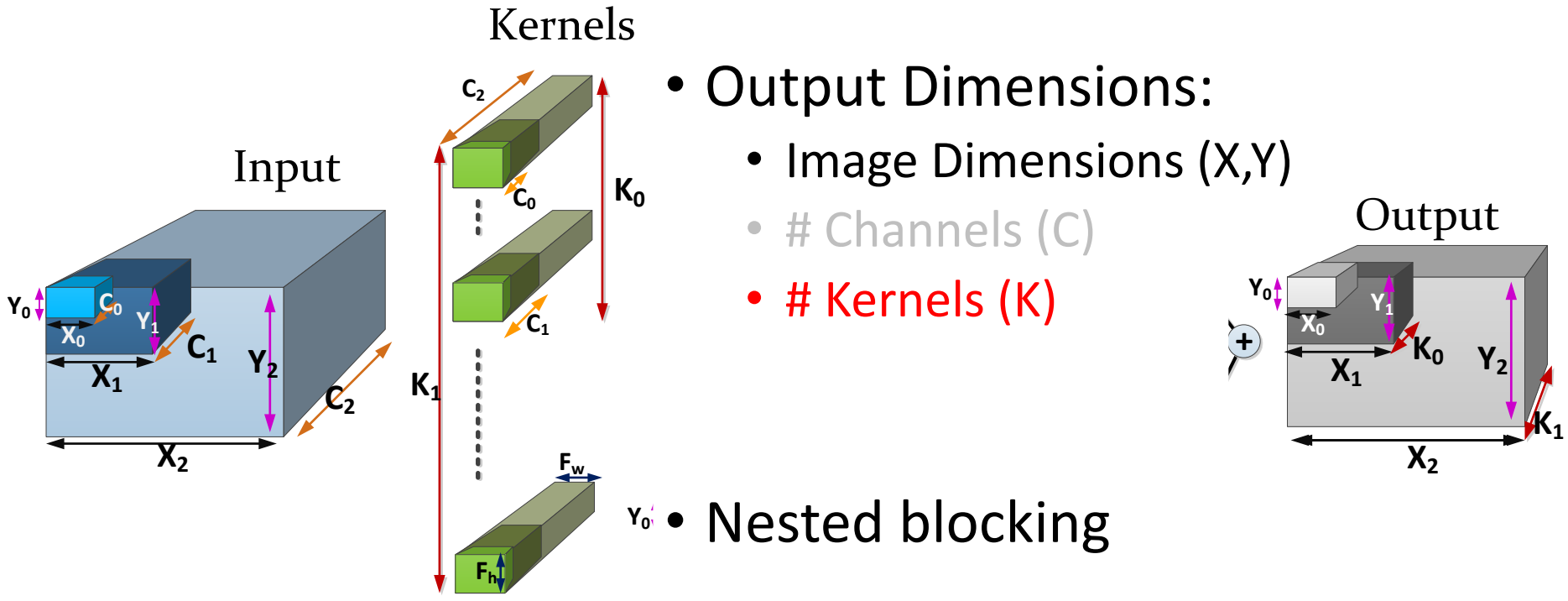
- Input Image Dimensions:
  - Image Dimensions (X,Y)
  - # Channels (C)
  - # Kernels (K)
  - Filter dimensions (F_h, F_w)
- Nested blocking

Multilevel Blocking of Convolutional Layer



- Kernel Dimensions:
  - Image Dimensions (X,Y)
  - # Channels (C)
  - # Kernels (K)
  - Filter dimensions (Fh, Fw)
- Nested blocking

Multilevel Blocking of Convolutional Layer



- Output Dimensions:
  - Image Dimensions (X,Y)
  - # Channels (C)
  - # Kernels (K)
- Nested blocking

Multilevel Blocking of Convolutional Layer