Object Recognition – Part 1

Danna Gurari
University of Colorado Boulder
Fall 2021

Review

- Last lecture:
  - Ways of seeing: image and video acquisition
  - Evolution of computer vision (before versus after 2012)
  - Background of machine learning and neural networks
  - Training deep neural networks: hardware & software

- Assignments (Canvas)
  - Reading assignment due this Wednesday

- Questions?

Object Recognition: Today's Topics

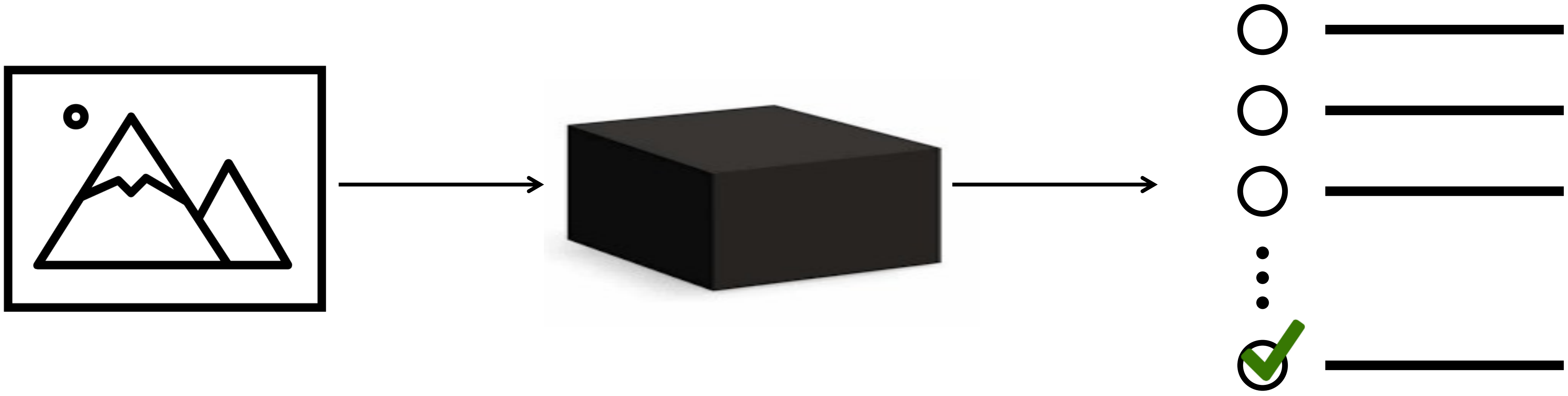
- Problem
- Applications
- Datasets
- Evaluation metric
- Typical Solution: Convolutional Neural Network

Object Recognition: Today's Topics

- Problem
- Applications
- Datasets
- Evaluation metric
- Typical Solution: Convolutional Neural Network

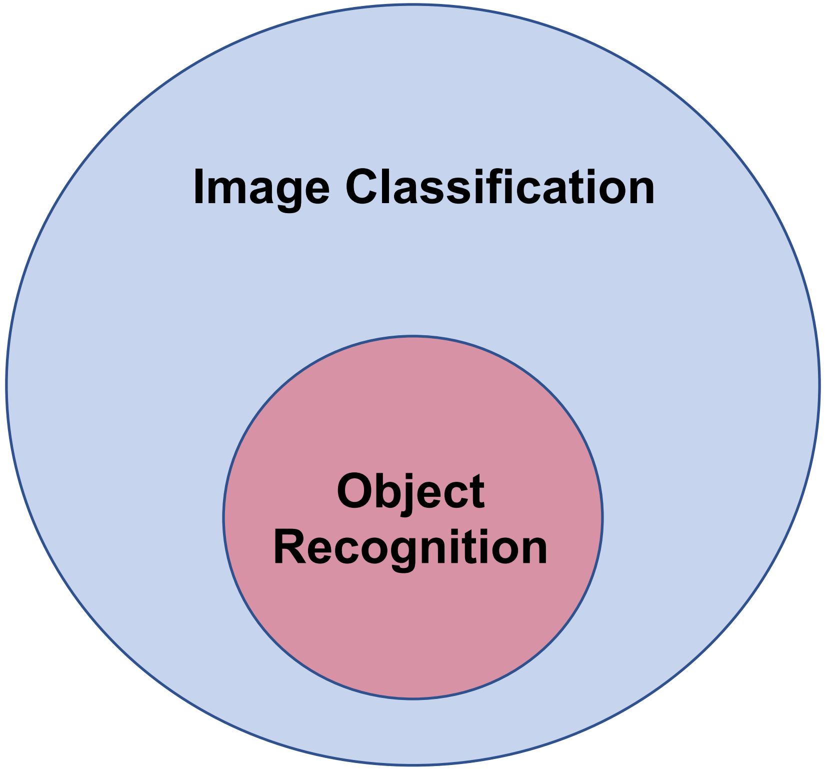
Object Recognition: Image Classification Problem

- Assign an image a label from a set of categories (i.e., multiple choice)



Object Recognition: Image Classification Problem

- Assign an image a label from a set of categories (i.e., multiple choice)



Object Recognition: Image Classification Problem

- Problem: What object is in the image?

Object Recognition: Today's Topics

- Problem
- Applications
- Datasets
- Evaluation metric
- Typical Solution: Convolutional Neural Network

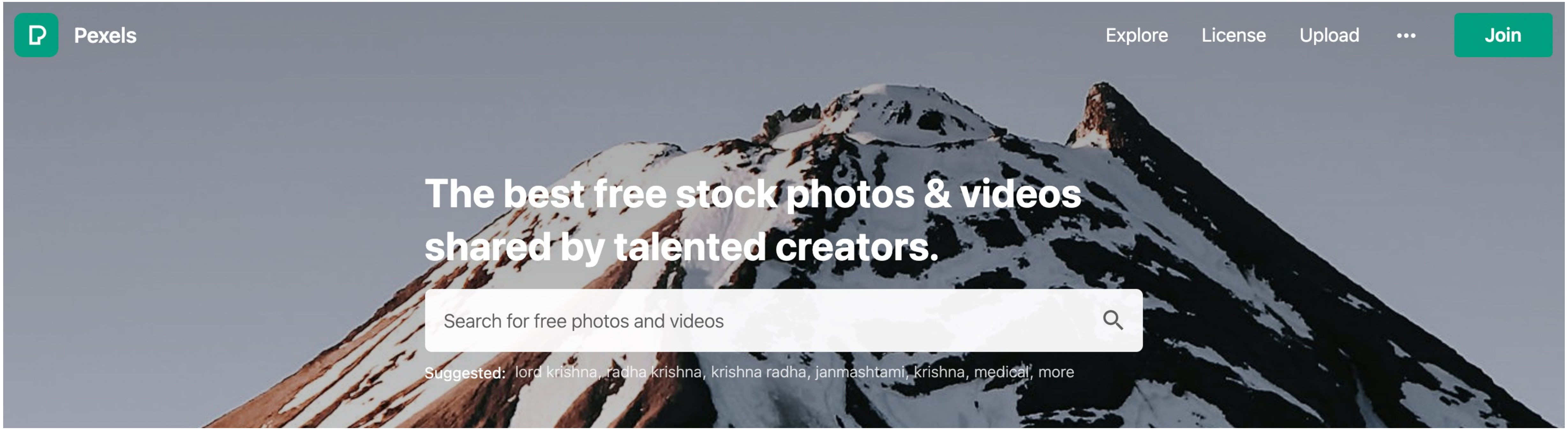
Shopping

Take a picture of an object and find where to buy it

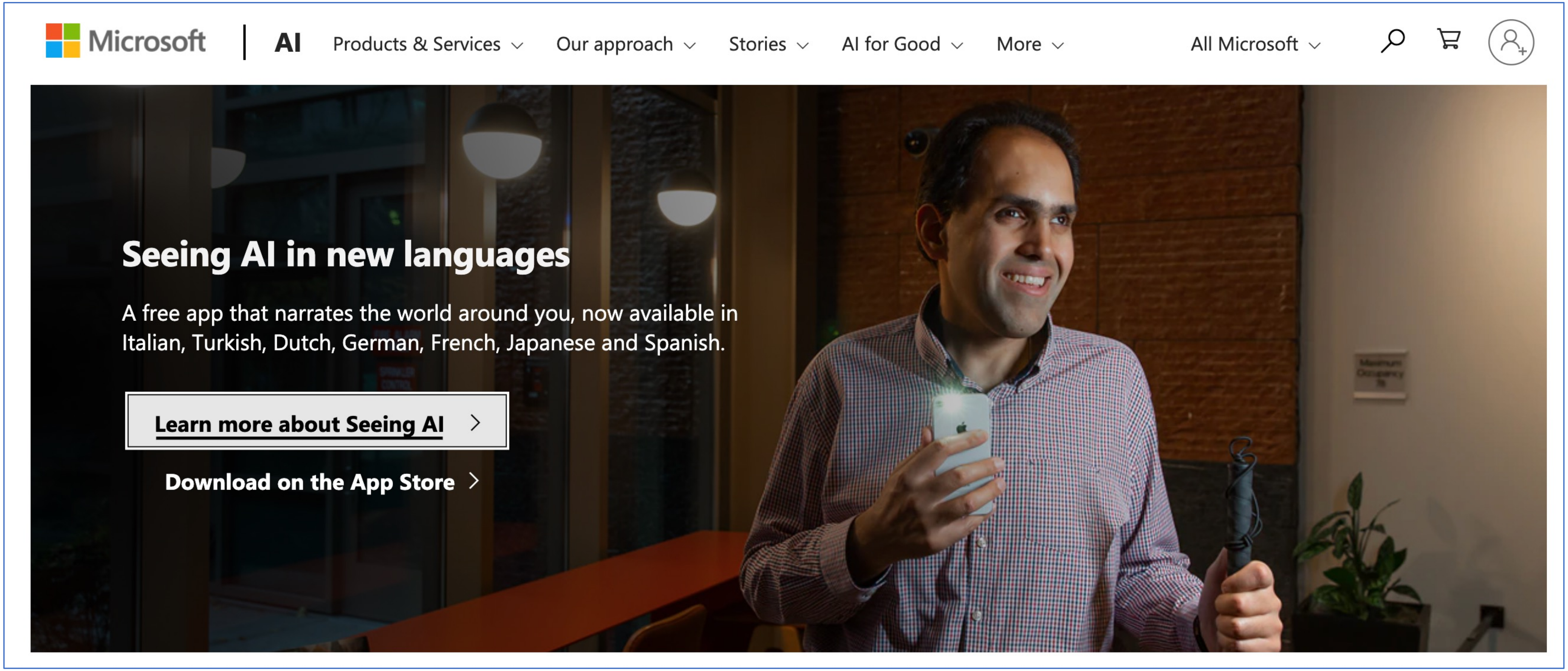
Photo Organization

Demo: https://www.youtube.com/watch?v=aBqmWUalnho
(start video at 1:46)

Image Search



Assistive Technology



Seeing AI Demo: https://www.youtube.com/watch?v=R2mC-NUAmMk

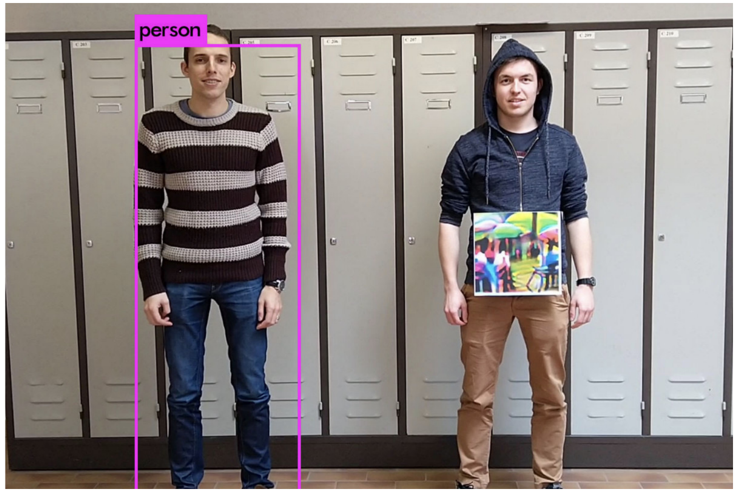
Applications Gone Wrong

- Ethical mistake: people tagged as "gorillas"

http://www.usatoday.com/story/tech/2015/07/01/google-apologizes-after-photos-identify-black-people-as-gorillas/29567465/

- Security risk: people mis-recognized or invisible when wearing special designs

https://www.cs.cmu.edu/~sbhagava/papers/face-rec-ccs16.pdf



https://www.theverge.com/2019/4/23/18512472/fool-ai-surveillance-adversarial-example-yolov2-person-detection

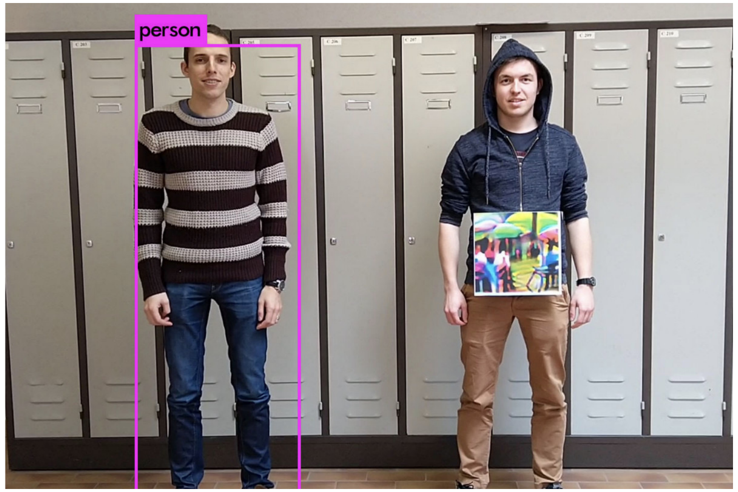
Applications Gone Wrong

1) Why are these mistakes occurring?

2) If you were the CEO providing these products, how would you respond to these issues?

http://www.usatoday.com/story/tech/2015/07/01/google-apologizes-after-photos-identify-black-people-as-gorillas/29567465/

https://www.cs.cmu.edu/~sbhagava/papers/face-rec-ccs16.pdf



https://www.theverge.com/2019/4/23/18512472/fool-ai-surveillance-adversarial-example-yolov2-person-detection

Object Recognition: Today's Topics

- Problem
- Applications
- Datasets
- Evaluation metric
- Typical Solution: Convolutional Neural Network

Research Until Early 2000s: Typical Approach

1945

1957

1966

Computer Vision

1983

CVPR

1987

ICCV

1990

ECCV

Algorithm Dataset

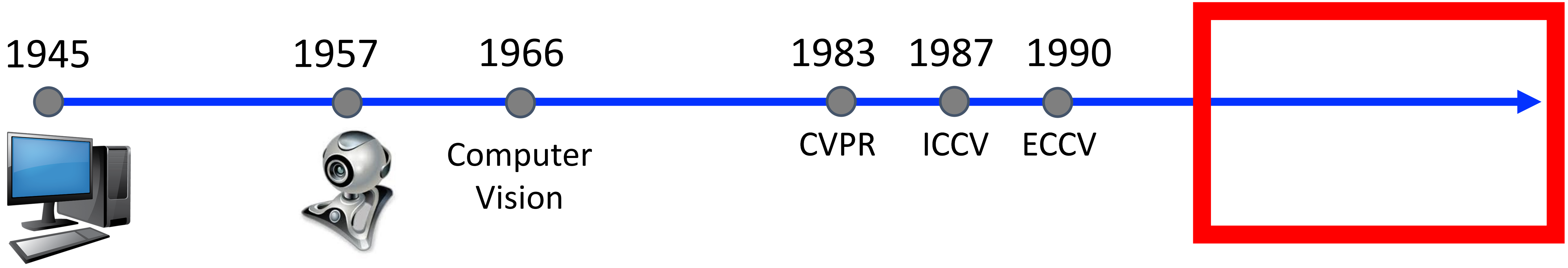
Algorithm Dataset

Algorithm Dataset

Algorithm

Dataset

Research Since Early ~2000s: Public Datasets



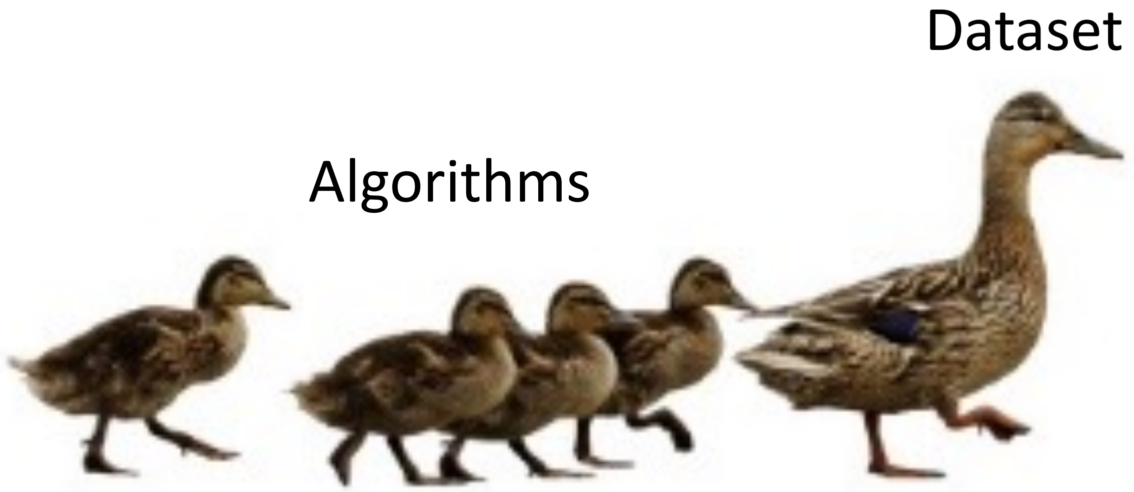


Image Source: http://larryzitnick.org/Talks/CVPR15_Dataset.pptx

Research Since Early ~2000s: Public Datasets

1945 — 1957 — 1966 — 1983 — 1987 — 1990 — 1999 — 2004 — 2009

1966: Computer Vision

1983: CVPR

1987: ICCV

1990: ECCV

e.g.,

1999: Caltech-6

2004: Caltech-101

2009: ImageNet

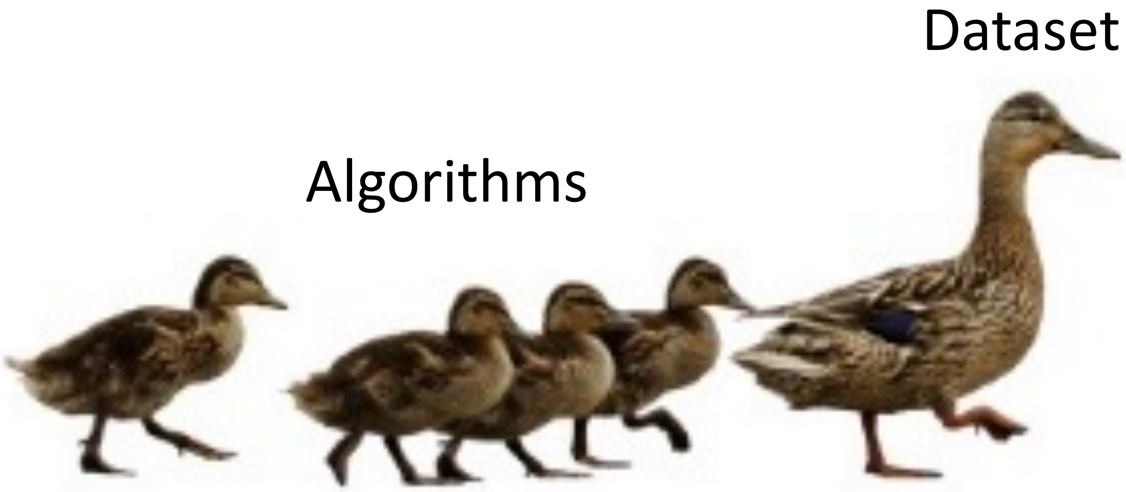


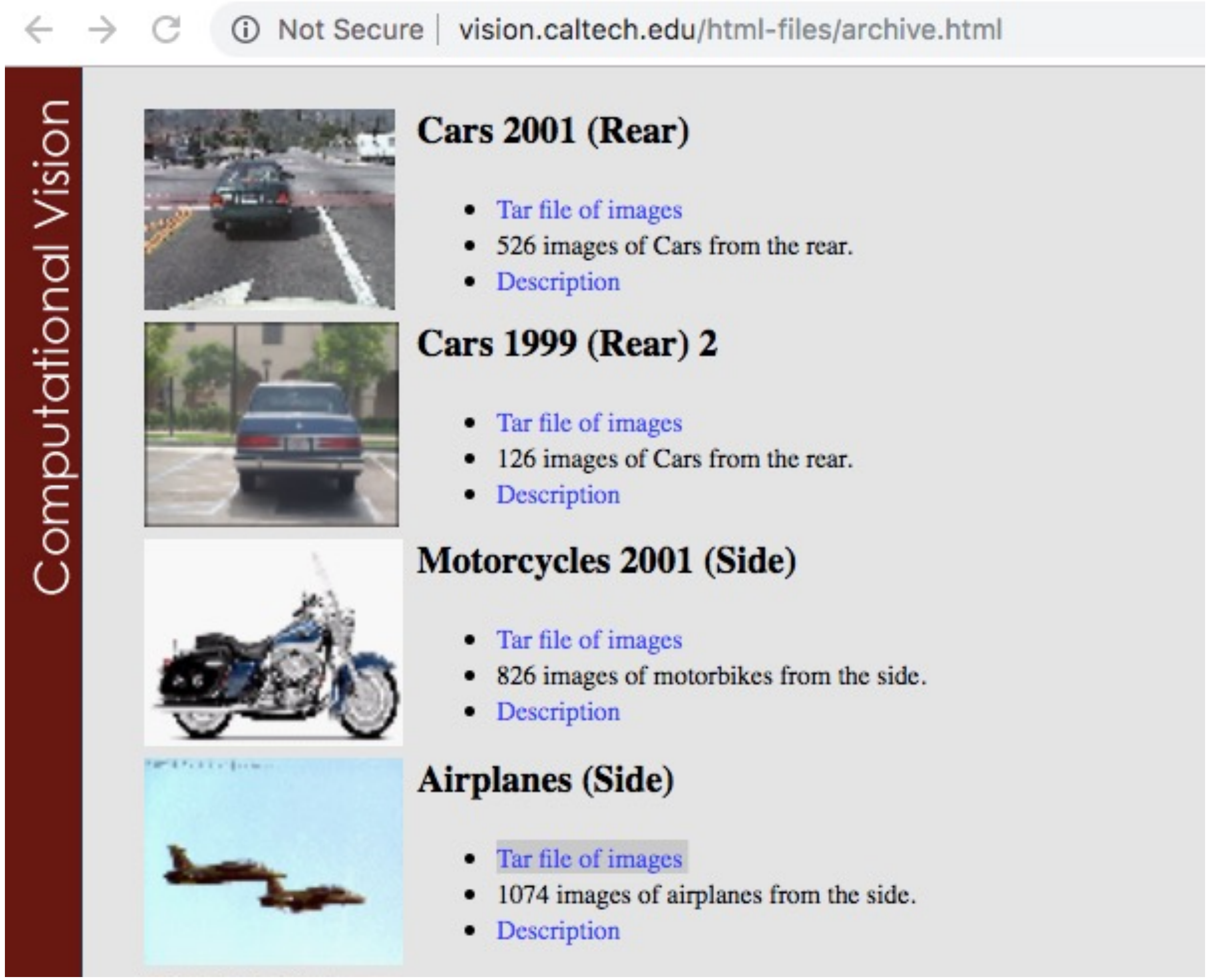
Image Source: http://larryzitnick.org/Talks/CVPR15_Dataset.pptx

Research Since Early ~2000s: Public Datasets

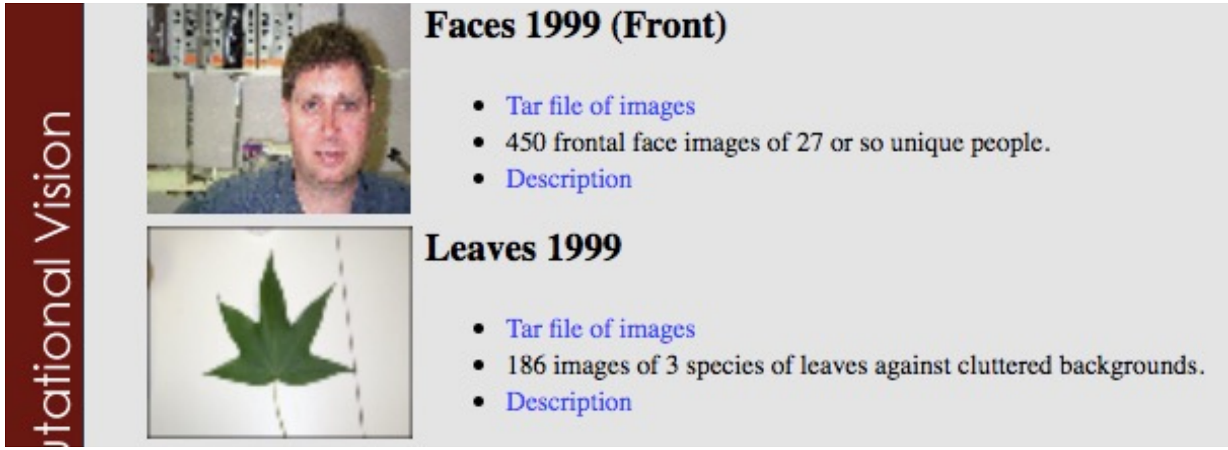
Typical steps for creating object recognition datasets:

1. Category Selection → 2. Image Collection → 3. Human Verification

Caltech-6



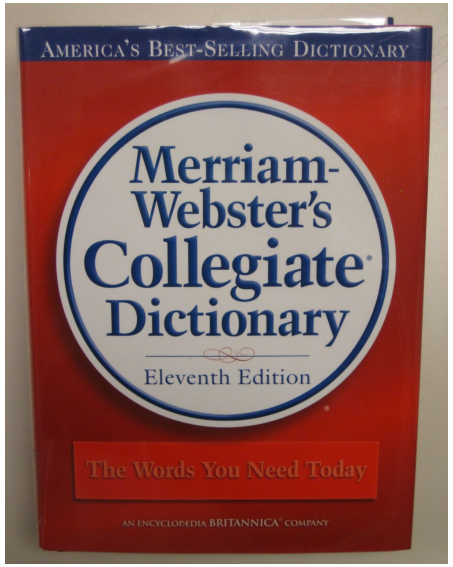
(1) Six categories selected and then (2) students took pictures or collected images from the web



http://www.vision.caltech.edu/html-files/archive.html

Caltech-101

1. Category Selection

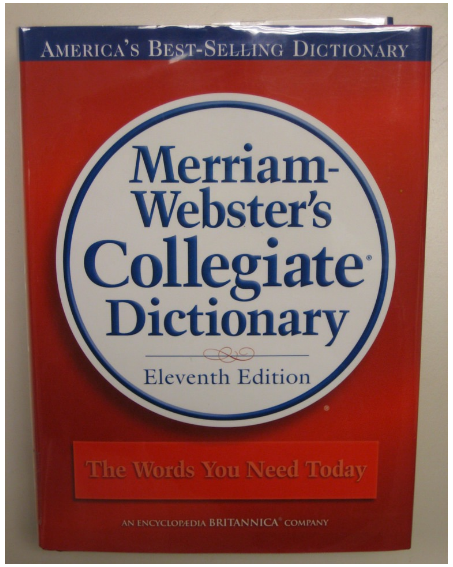


Flipped through a dictionary and chose 101 categories associated with a drawing

Li Fei Fei, Rob Fergus, & Pietro Perona. CVPR 2004.

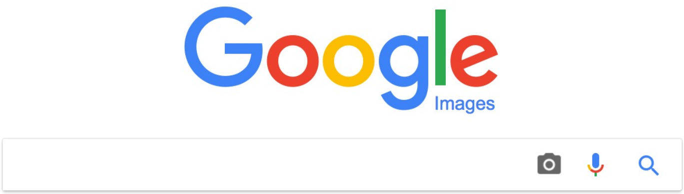
Caltech-101

1. Category Selection



Flipped through a dictionary and chose 101 categories associated with a drawing

2. Image Collection

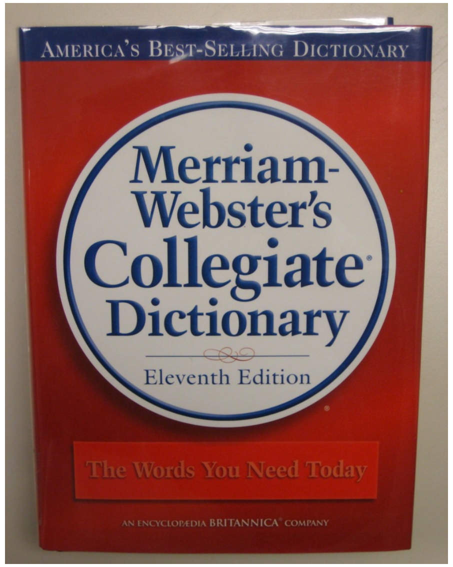


Search for each category

Li Fei Fei, Rob Fergus, & Pietro Perona. CVPR 2004.

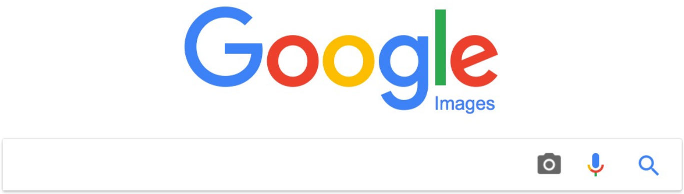
Caltech-101

1. Category Selection



Flipped through a dictionary and chose 101 categories associated with a drawing

2. Image Collection



Search for each category

3. Human Verification

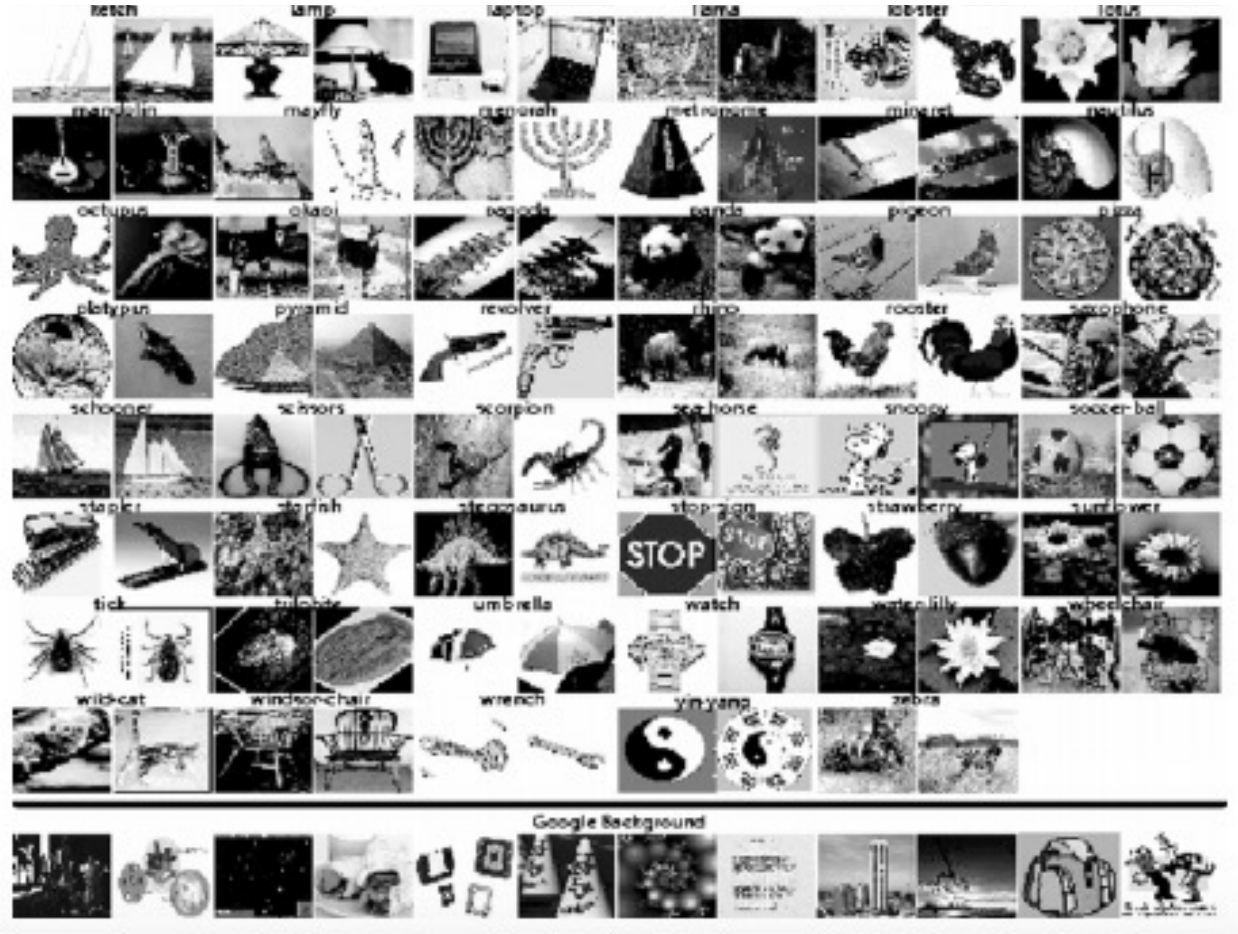
- 2 graduate students reviewed & discarded irrelevant images

- Result is 9,144 grayscale 300x200 pixel images with 45-400 images per category

Li Fei Fei, Rob Fergus, & Pietro Perona. CVPR 2004.

Caltech-101

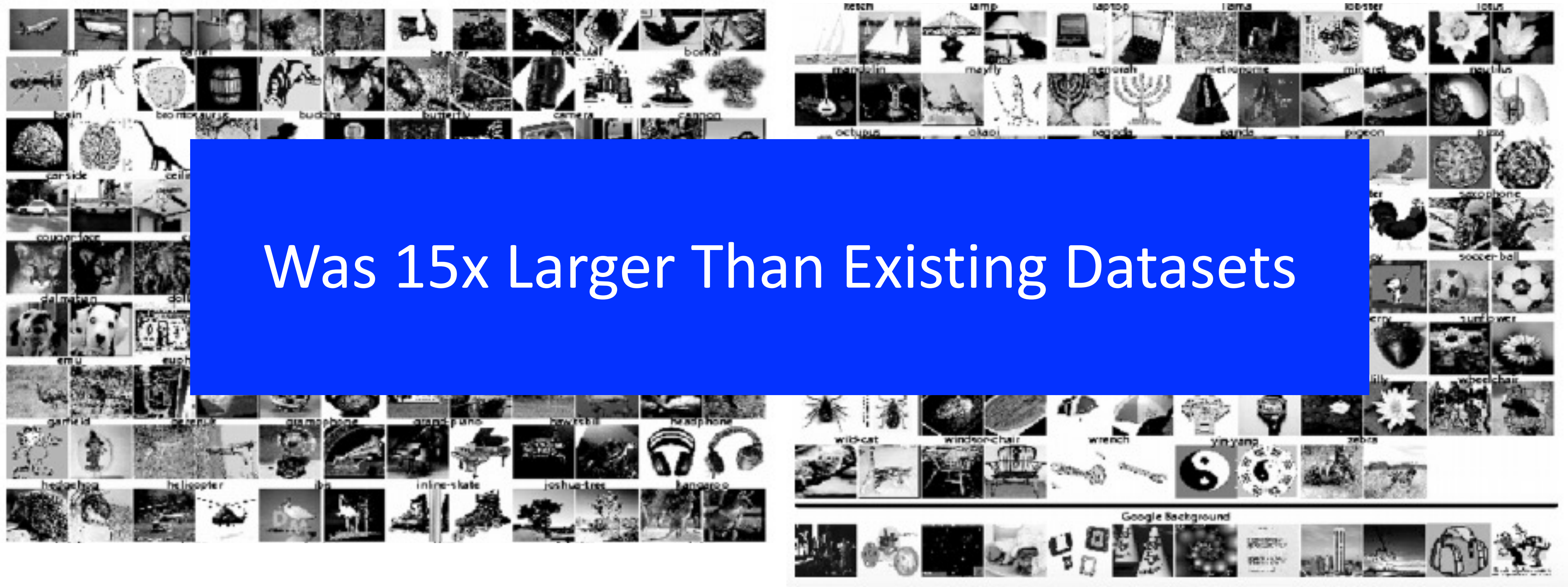
Two random samples per category



Dataset location: http://vision.caltech.edu

Caltech-101

Two random samples per category

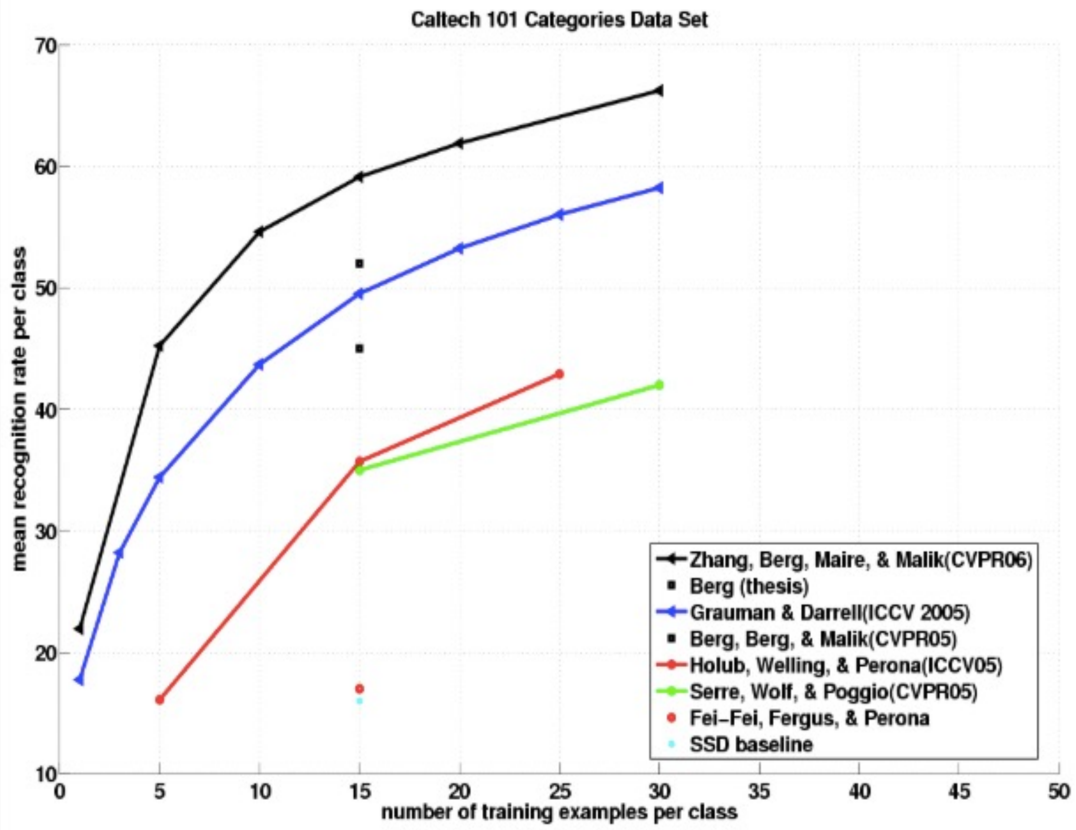


Was 15x Larger Than Existing Datasets

Dataset location: http://vision.caltech.edu

Caltech-101

Progress of algorithms charted



Latest results (March 2006) on the Caltech 101 from a variety of groups. (published results only).

If you would like to include your algorithm's performance please email us at holub@caltech.edu or greg@vision.caltech.edu with a citation and your results. Thanks!

We are also interested in the time it takes to run your algorithm. Both during the training and during the classification stage

Plot courtesy of Hao Zhang.

Update by holub, April 2006.

Li Fei Fei, Rob Fergus, & Pietro Perona. CVPR 2004.

ImageNet

After creating Caltech-101 and finishing her PhD, Fei-Fei Li began her career as an assistant professor creating ImageNet.

Hear her tell her story:

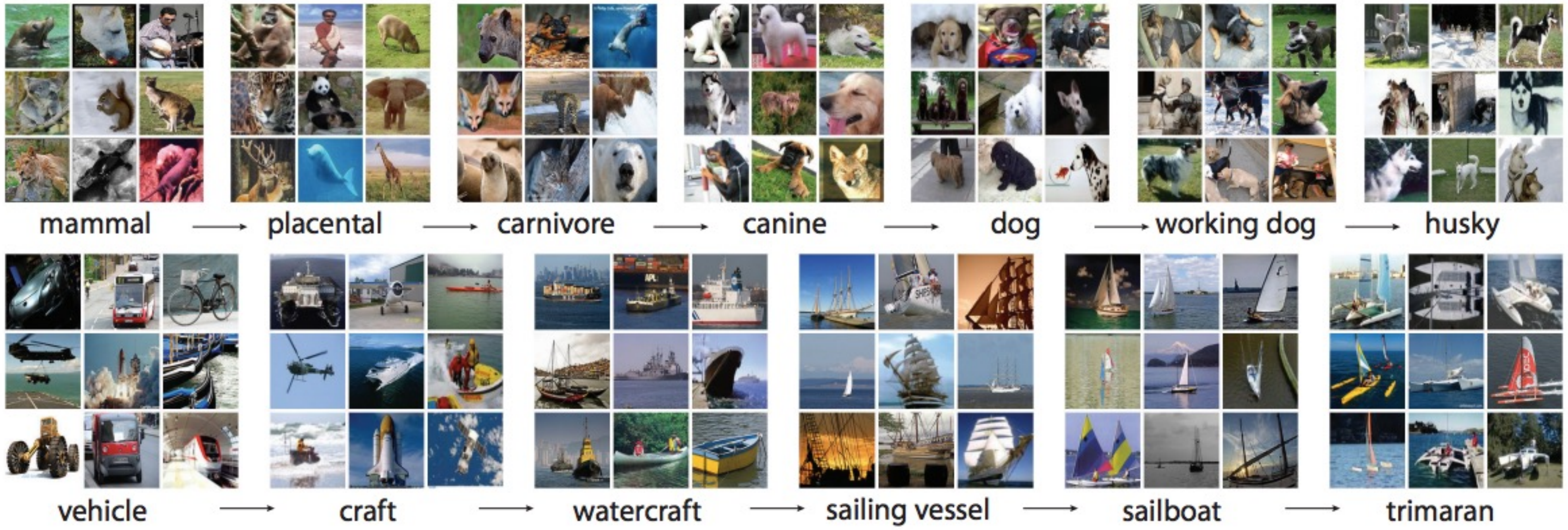
https://www.youtube.com/watch?v=40riCqvRoMs
(5:44 – 9:35)

ImageNet

1. Category Selection

~10% of concepts (synonym sets) in WordNet taxonomy

e.g., two root-to-leaf branches of ImageNet with nine examples for each “synonym set”



Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, & Li Fei-Fei. ImageNet: A Large-Scale Hierarchical Image Database. CVPR 2009.

ImageNet

1. Category Selection

~10% of concepts (synonym sets) in WordNet taxonomy

→

2. Image Collection

flickr

(& more search engines)

Query expansion:

- Augment queries
- Translate queries to different languages

Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, & Li Fei-Fei. ImageNet: A Large-Scale Hierarchical Image Database. CVPR 2009.

ImageNet

Key Insight: use crowdsourcing to recruit many people to verify images

1. Category Selection

~10% of concepts (synonym sets) in WordNet taxonomy

→

2. Image Collection

flickr

(& more search engines)

Query expansion:

- Augment queries
- Translate queries to different languages

→

3. Human Verification

- Humans verify if image contains queried object

- Use majority vote decision from multiple humans to support high quality results

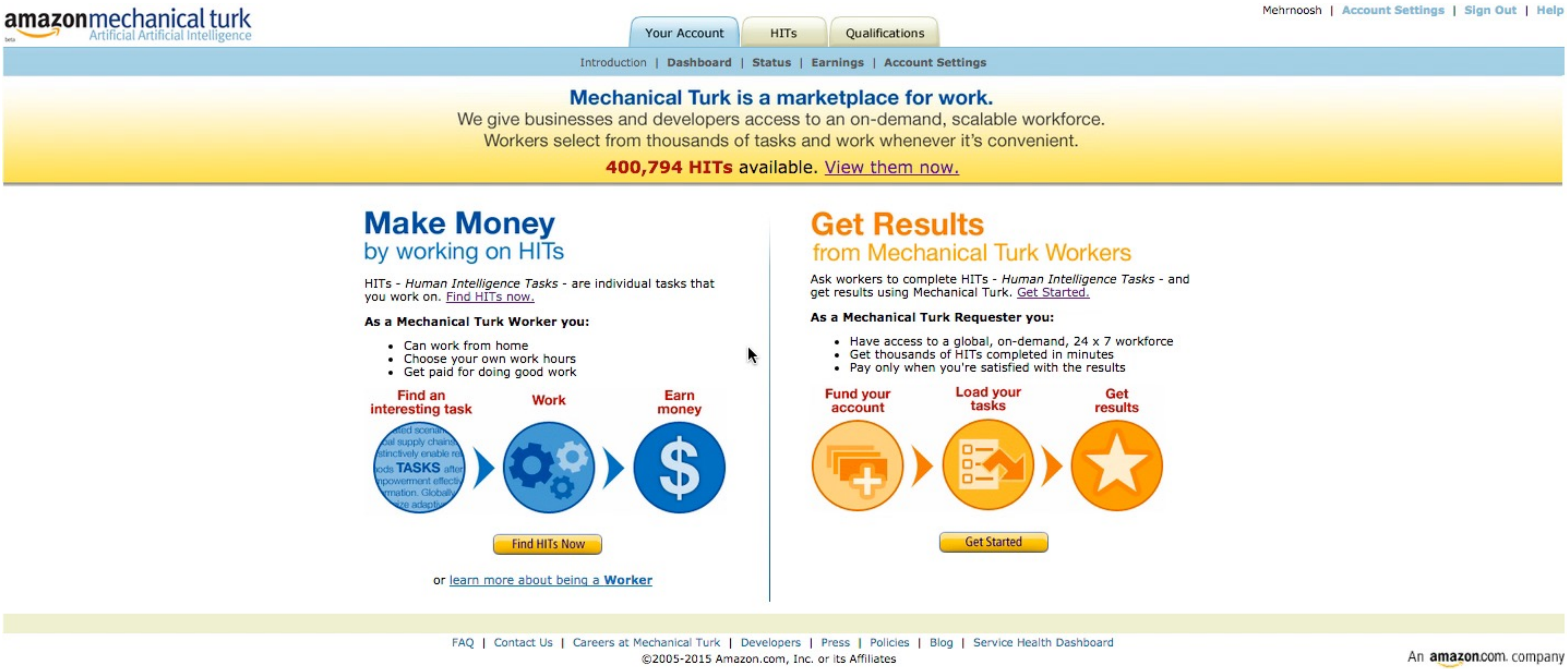
Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, & Li Fei-Fei. ImageNet: A Large-Scale Hierarchical Image Database. CVPR 2009.

ImageNet Task

Definition of the target synonym set with link to Wikipedia.

Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, & Li Fei-Fei. ImageNet: A Large-Scale Hierarchical Image Database. CVPR 2009.

ImageNet Workers



Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, & Li Fei-Fei. ImageNet: A Large-Scale Hierarchical Image Database. CVPR 2009.

Research Since Early ~2000s: Public Datasets

1945

1957

1966

Computer Vision

1983 CVPR

1987 ICCV

1990 ECCV

e.g.,

1999 Caltech-6

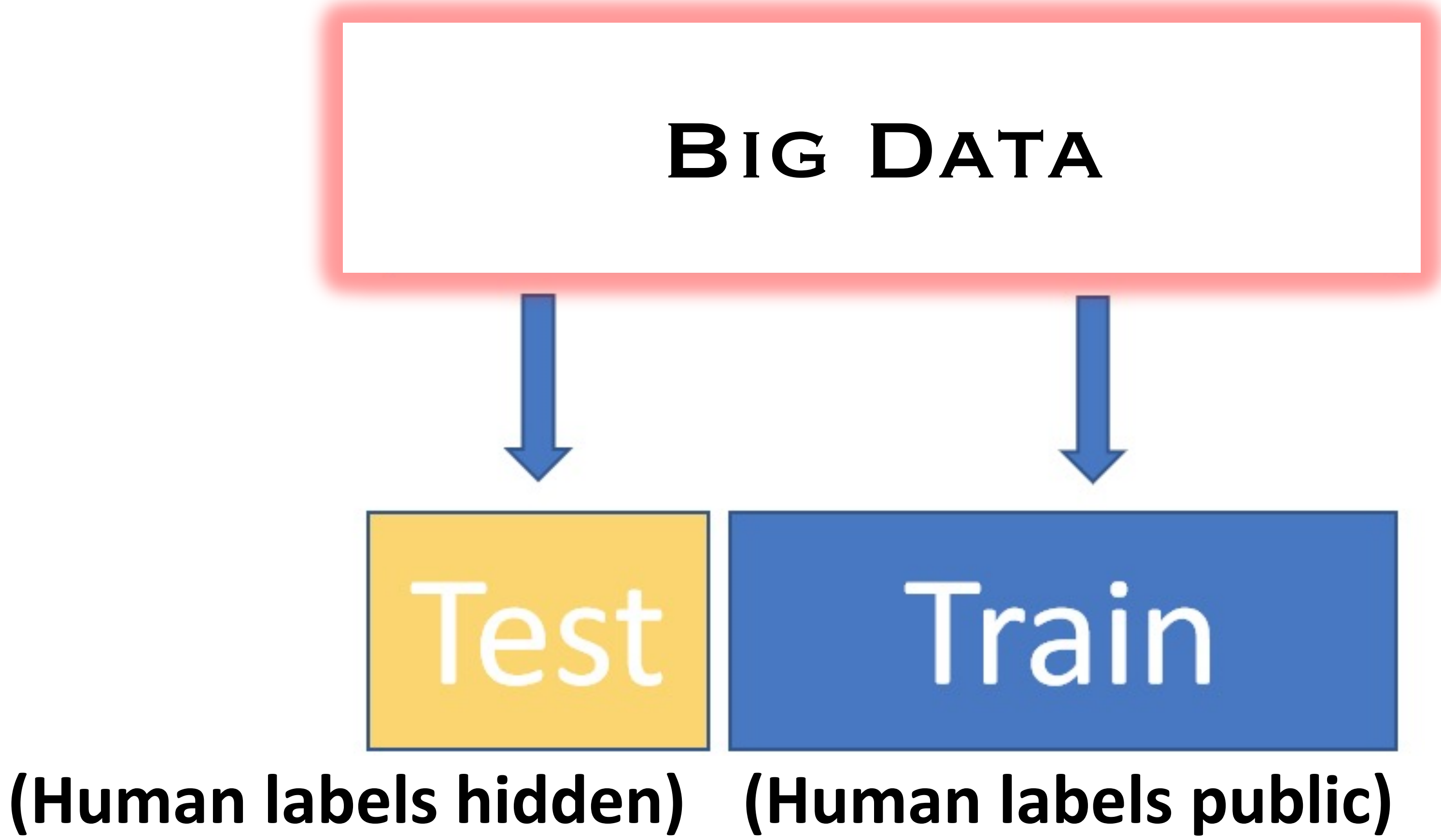
2004 Caltech-101

2009 ImageNet

# Categories:	6	101	1000
# Images:	3,738	9,144	1,461,406

Trend: build bigger datasets

ImageNet Challenge



Winner: highest scoring method on the hidden test set

ImageNet Challenge with Evaluation Server

Not Secure | image-net.org

IMAGENET

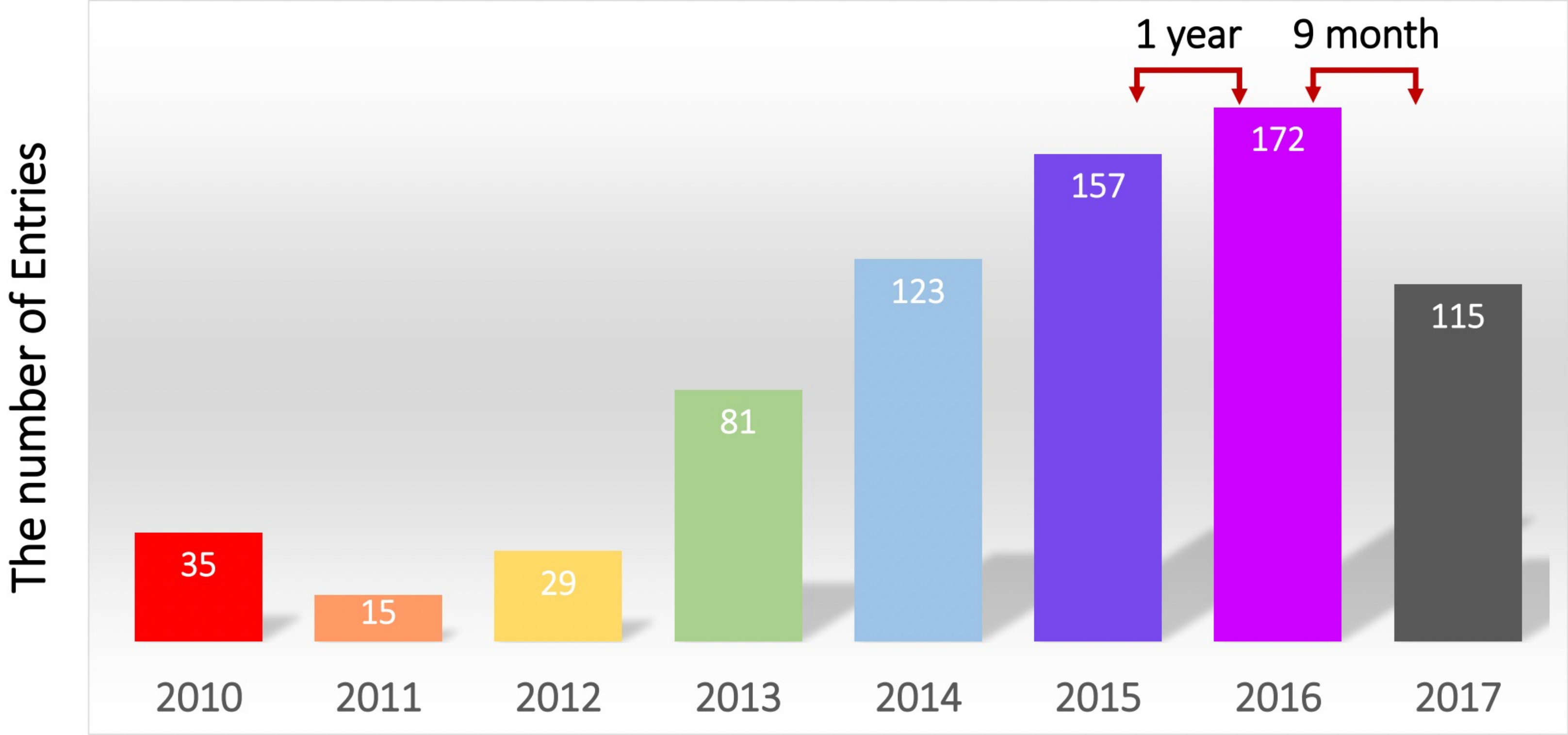
14,197,122 images, 21841 synsets indexed

Explore Download Challenges Publications CoolStuff About

Not logged in. Login | Signup

ImageNet is an image database organized according to the WordNet hierarchy (currently only the nouns), in which each node of the hierarchy is depicted by hundreds and thousands of images. Currently we have an average of over five hundred images per node. We hope ImageNet will become a useful resource for researchers, educators, students and all of you who share our passion for pictures.
Click here to learn more about ImageNet, Click here to join the ImageNet mailing list.

ImageNet Challenge Community Engagement



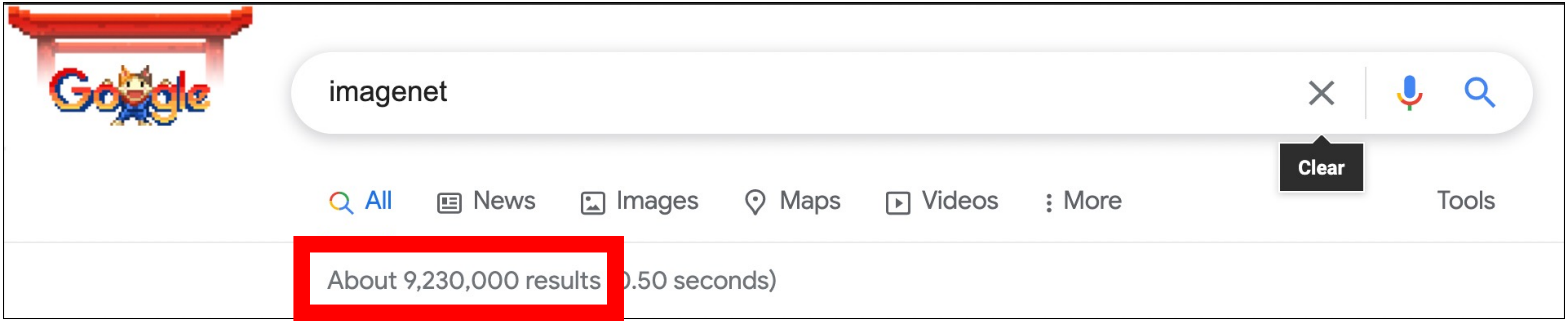
- 727 entries (plus an entry that famously was kicked out in 2015 for cheating from Baidu)
- Labor cost ~$110 million: assuming 3 people contribute to each entry and $50k cost per person

Source: https://image-net.org/static_files/files/ILSVRC2017_overview.pdf

ImageNet Impact Recognized

“Suddenly people started to pay attention, not just within the AI community but across the technology industry as a whole.”

- Economist



“From not working to neural networking". *The Economist*. 25 June 2016. Retrieved July 15, 2021.

ImageNet Impact Recognized

1945 — 

1957 — 

1966 — Computer Vision

1983 — CVPR

1987 — ICCV

1990 — ECCV

2009 — ImageNet

2019 — Award

PAMI Longuet-Higgins Prize

Retrospective Most Impactful Paper from CVPR 2009

ImageNet: A large-scale hierarchical image database

Jia Deng, Wei Dong, Richard Socher,
Li-Jia Li, Kai Li, and Li Fei-Fei

https://syncedreview.com/2019/06/18/cvpr-2019-attracts-9k-attendees-best-papers-announced-imagenet-honoured-10-years-later/

ImageNet Impact Recognized

1945 — 1957 — 1966 — 1983 — 1987 — 1990 — 2009 — 2019

Computer Vision (1966)

CVPR (1983)

ICCV (1987)

ECCV (1990)

ImageNet (2009)

Award (2019)

PAMI Longuet-Higgins Prize

Retrospective Most Impactful Paper from CVPR 2009

ImageNet: A large-scale hierarchical image database

Jia Deng, Wei Dong, Richard Socher,
Li-Jia Li, Kai Li, and Li Fei-Fei

"In 2009, ImageNet was not the most mainstream work, but all of us who did this project believed that it would have a big impact, so we put in a lot of efforts. One of the revelations it gives me is that you don't have to do the most popular things, but do what you believe will have an impact."

-First author, Jia Deng

https://syncedreview.com/2019/06/18/cvpr-2019-attracts-9k-attendees-best-papers-announced-imagenet-honoured-10-years-later/

Trend Started With ImageNet: Progress Charted by Progress on Community Shared Datasets

1. Identify an AI problem

2. **Create infrastructure to work on the problem**: a big labelled dataset with a quantitative approach to evaluate algorithms

3. **Scale**: encourage community involvement in developing algorithms by publicly sharing the data with evaluation server and hosting a workshop to announce winners

Trend Started With ImageNet: Progress Charted by Progress on Community Shared Datasets

Why have dataset challenges?

- Provide “fair” comparison between algorithms
- Create a community around a shared goal

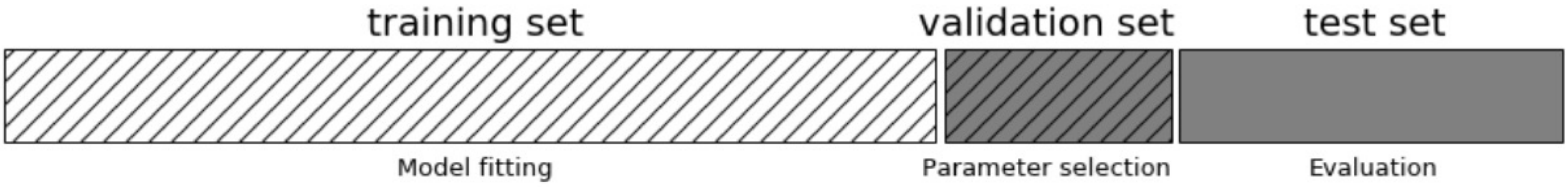
Trend Started With ImageNet: Progress Charted by Progress on Community Shared Datasets

How dataset challenges often are designed:

1. Publicly-shared train (and validation) dataset with “ground truth” labels

2. Publicly-shared test dataset (“ground truth” labels are hidden)

3. Metrics for evaluating algorithm-generated results on the test set



Many Public Datasets Available; e.g.,

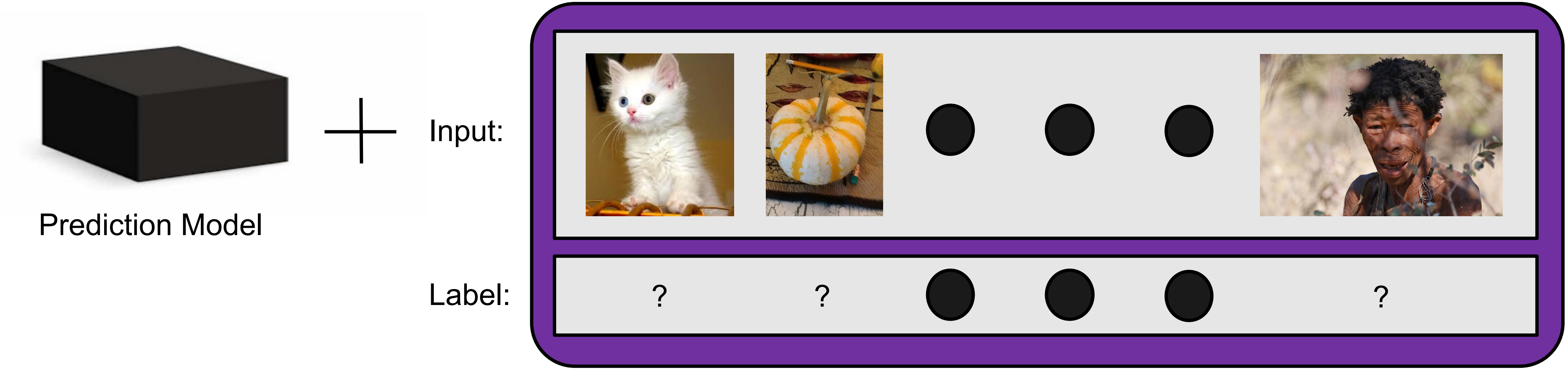
- Google Dataset Search
- Amazon's AWS datasets
- Kaggle datasets
- Wikipedia's list
- UC Irvine Machine Learning Repository
- Quora.com
- Reddit
- Dataportals.org
- Opendatamonitor.eu
- Quandl.com

Object Recognition: Today's Topics

- Problem
- Applications
- Datasets
- Evaluation metric
- Typical Solution: Convolutional Neural Network

Goal: Design Models that **Generalize** Well to New, Previously Unseen Examples

Apply model on “test set” to measure generalization error



Evaluation Metric for ImageNet Challenge

Assumption: 1 ground truth label per image

Error is average over all test images using this rule per image:

* 0 if any predictions match the ground truth
* 1 otherwise

e.g., top 5 error

Steel drum

Output:
Scale
T-shirt
<u>Steel drum</u>
Drumstick
Mud turtle

✔

Output:
Scale
T-shirt
Giant panda
Drumstick
Mud turtle

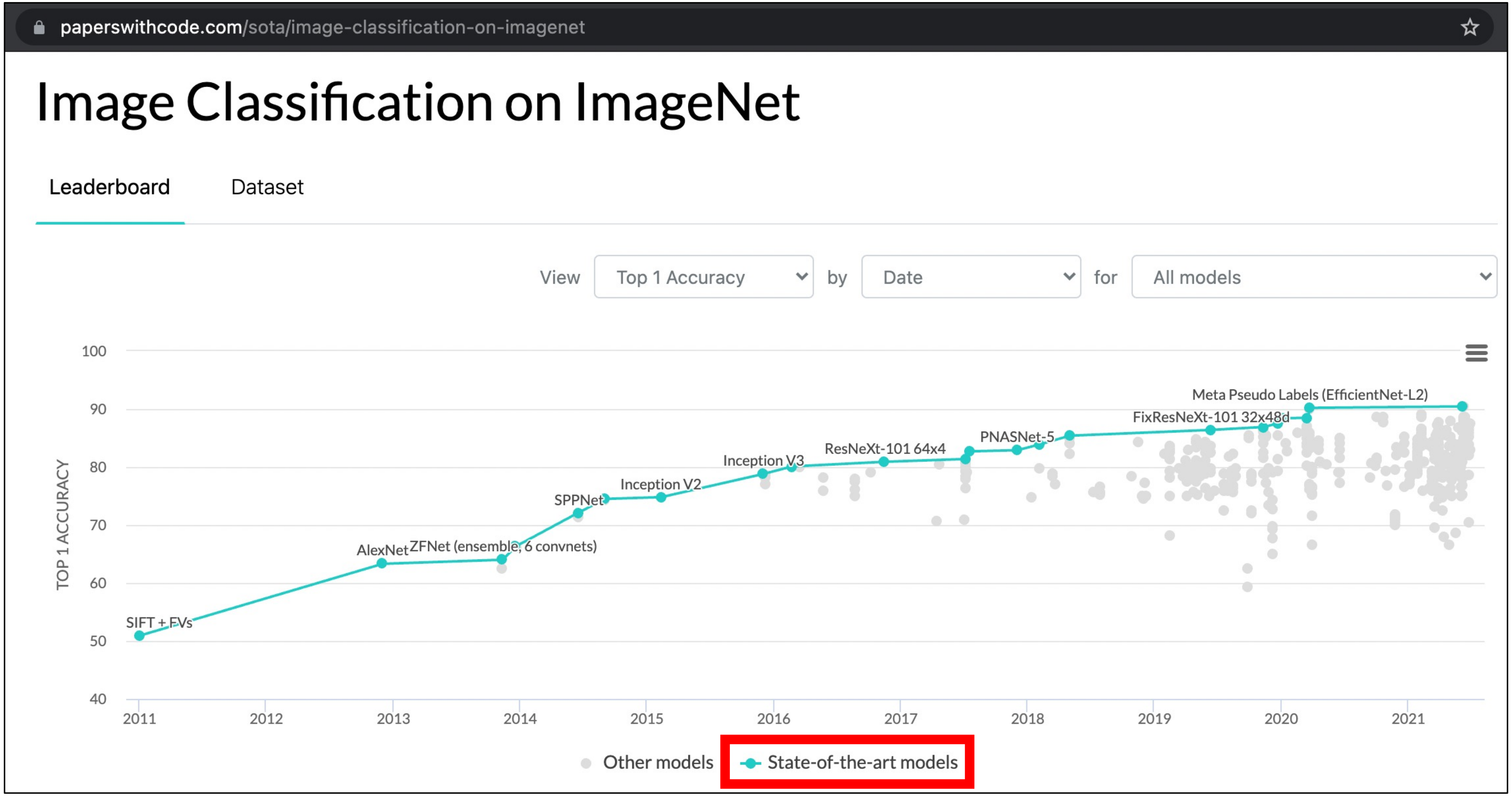
✗

Source: https://image-net.org/static_files/files/ILSVRC2017_overview.pdf

Object Recognition: Today's Topics

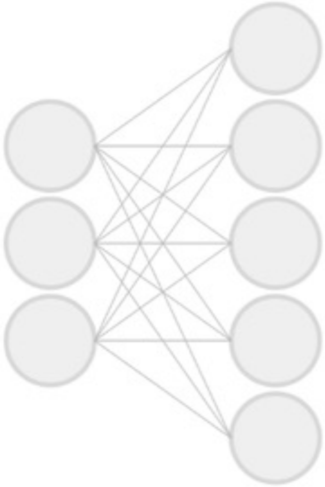
- Problem
- Applications
- Datasets
- Evaluation metric
- Typical Solution: Convolutional Neural Network

Rise of Convolutional Neural Networks (CNNs)



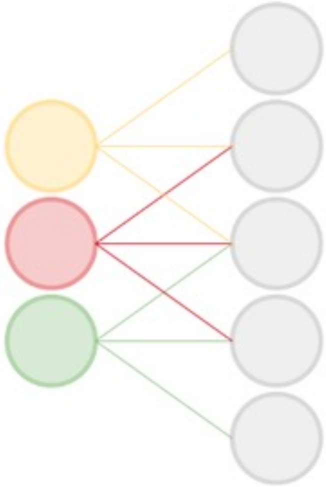
Fully-Connected Neural Networks vs CNNs

Fully-connected:



Rather than have each node provide input to each node in the next layer…

Convolutional:

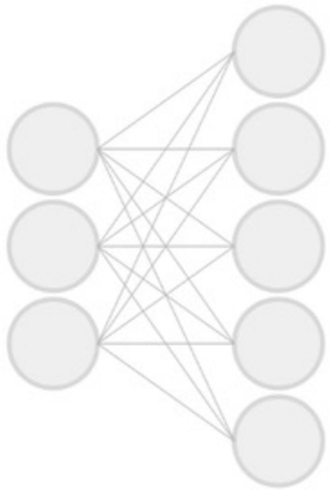


each node receives input only from a small neighborhood in previous layer (and there is parameter sharing)

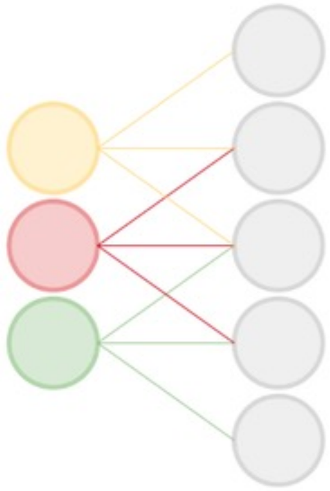
Figure Source: https://qph.fs.quoracdn.net/main-qimg-2e1f0071ca9878f7719ed0ea8aeb386d

Fully-Connected Neural Networks vs CNNs

Fully-connected:



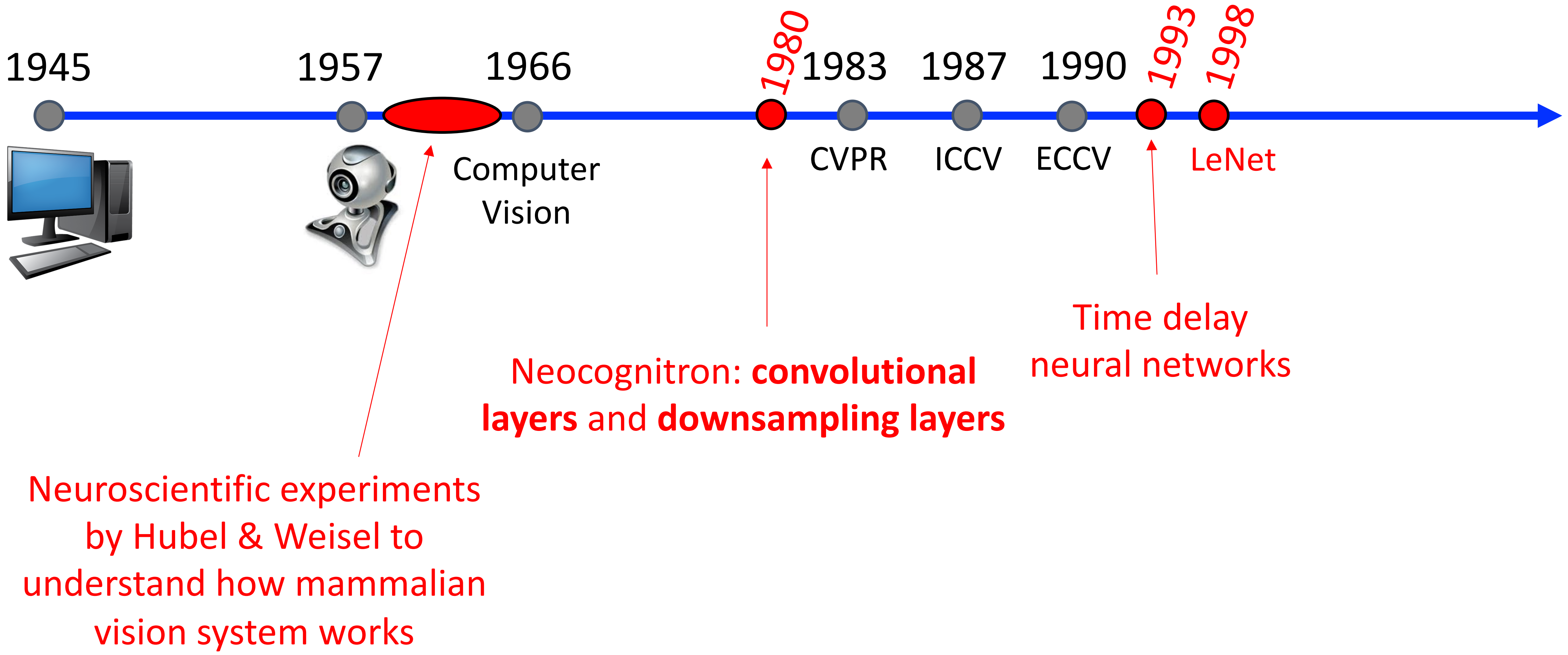
Convolutional:



CNNs dramatically reduce number of model parameters!

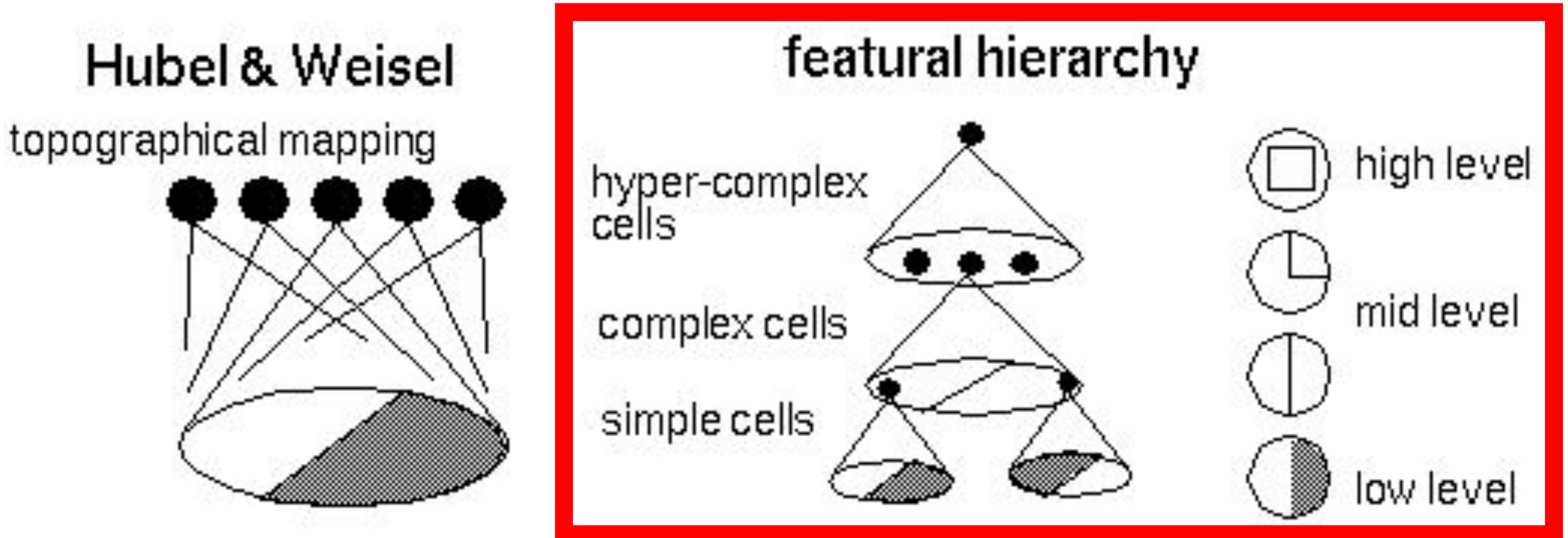
Figure Source: https://qph.fs.quoracdn.net/main-qimg-2e1f0071ca9878f7719ed0ea8aeb386d

CNN Origins


1945
1957
1966
1980
1983
1987
1990
1993
1998
Computer
Vision
CVPR
ICCV
ECCV
LeNet
Neuroscientific experiments by Hubel & Weisel to understand how mammalian vision system works
Neocognitron: convolutional layers and downsampling layers
Time delay neural networks

Inspiration: Biology

Insights come from Nobel Prize winning work by Hubel & Weisel to understand how mammalian vision system works



Key Observation: cells are organized as a hierarchy of feature detectors, with higher level features responding to patterns of activation in lower level cells

Source: https://bruceoutdoors.files.wordpress.com/2017/08/hubel.jpg

LeNet: Core Components of Modern CNNs

Extracts useful features to pass to a MLP using:

- Convolutional layers
- Pooling Layers

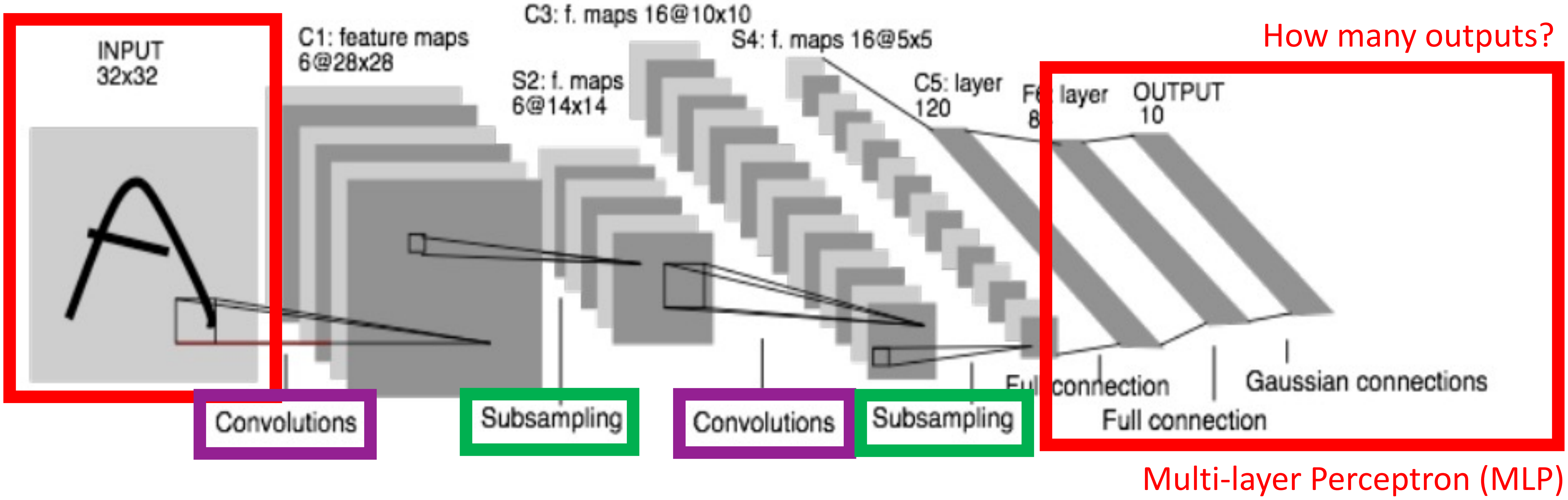


Figure Credit: https://people.eecs.berkeley.edu/~jrs/189/lec/cnn.pdf

Y. Lecun ; L. Bottou ; Y. Bengio ; P. Haffner; Gradient-based learning applied to document recognition; 1998

CNN: Convolutional Layers

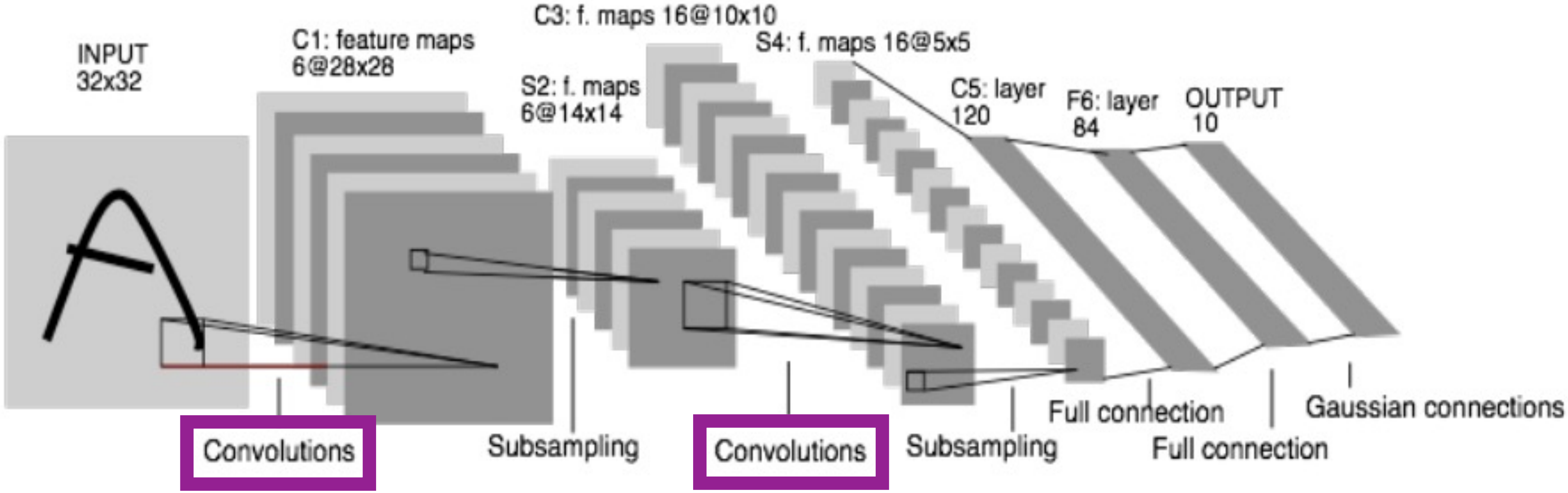
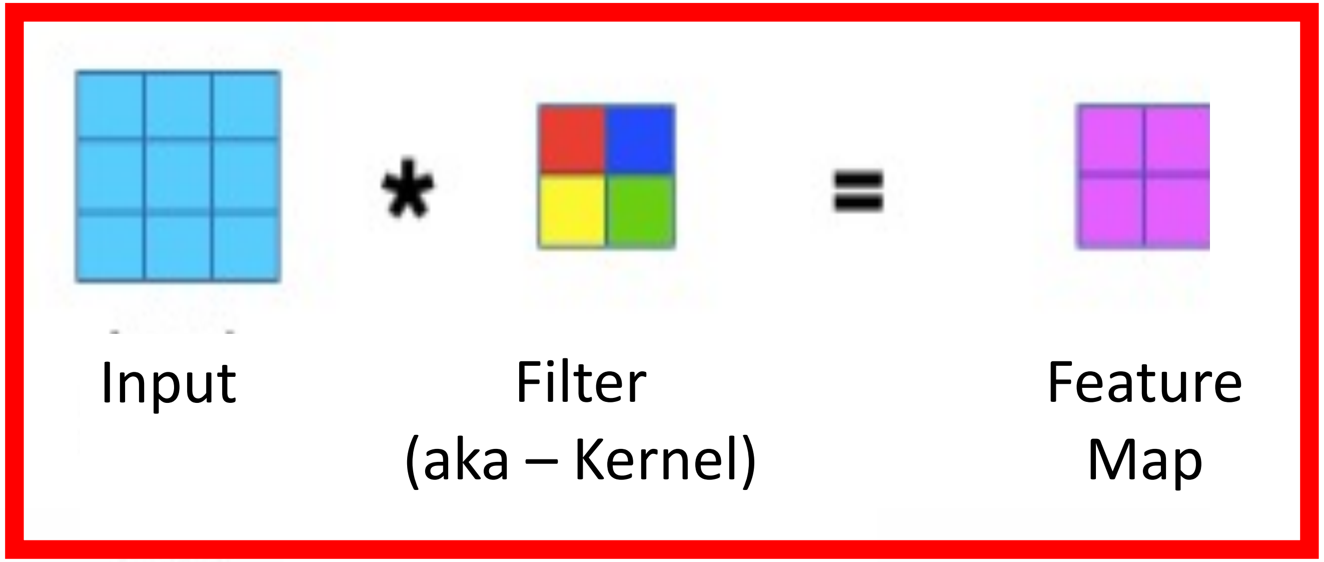


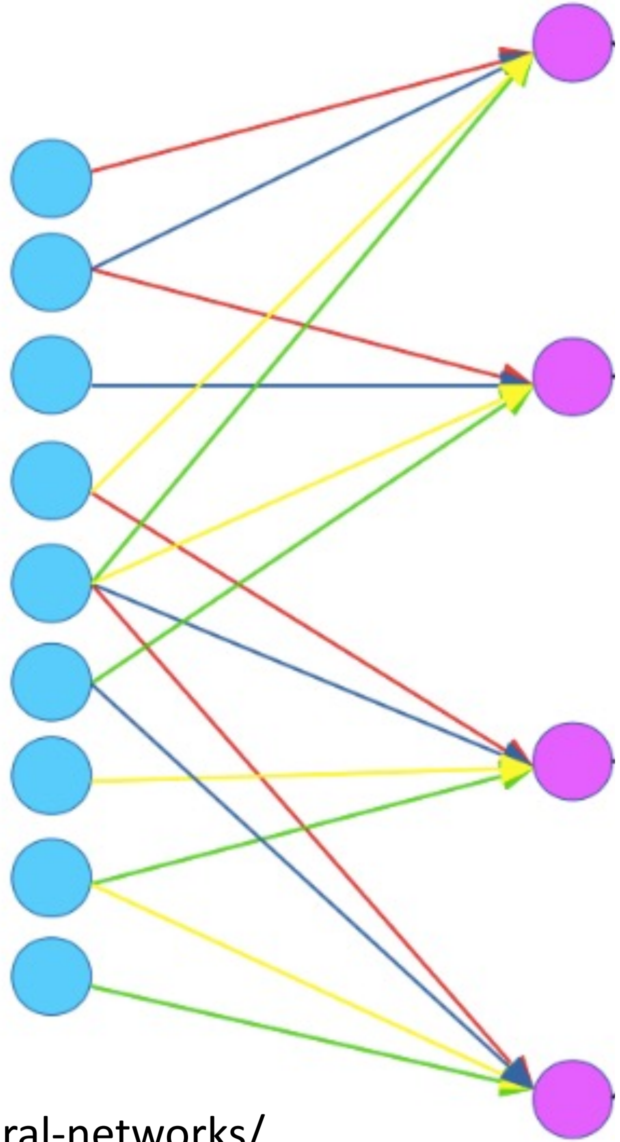
Figure Credit: https://people.eecs.berkeley.edu/~jrs/189/lec/cnn.pdf

Y. Lecun ; L. Bottou ; Y. Bengio ; P. Haffner; Gradient-based learning applied to document recognition; 1998

Convolutional Layer: Applies Linear Filter



Way to Interpret
Neural Network

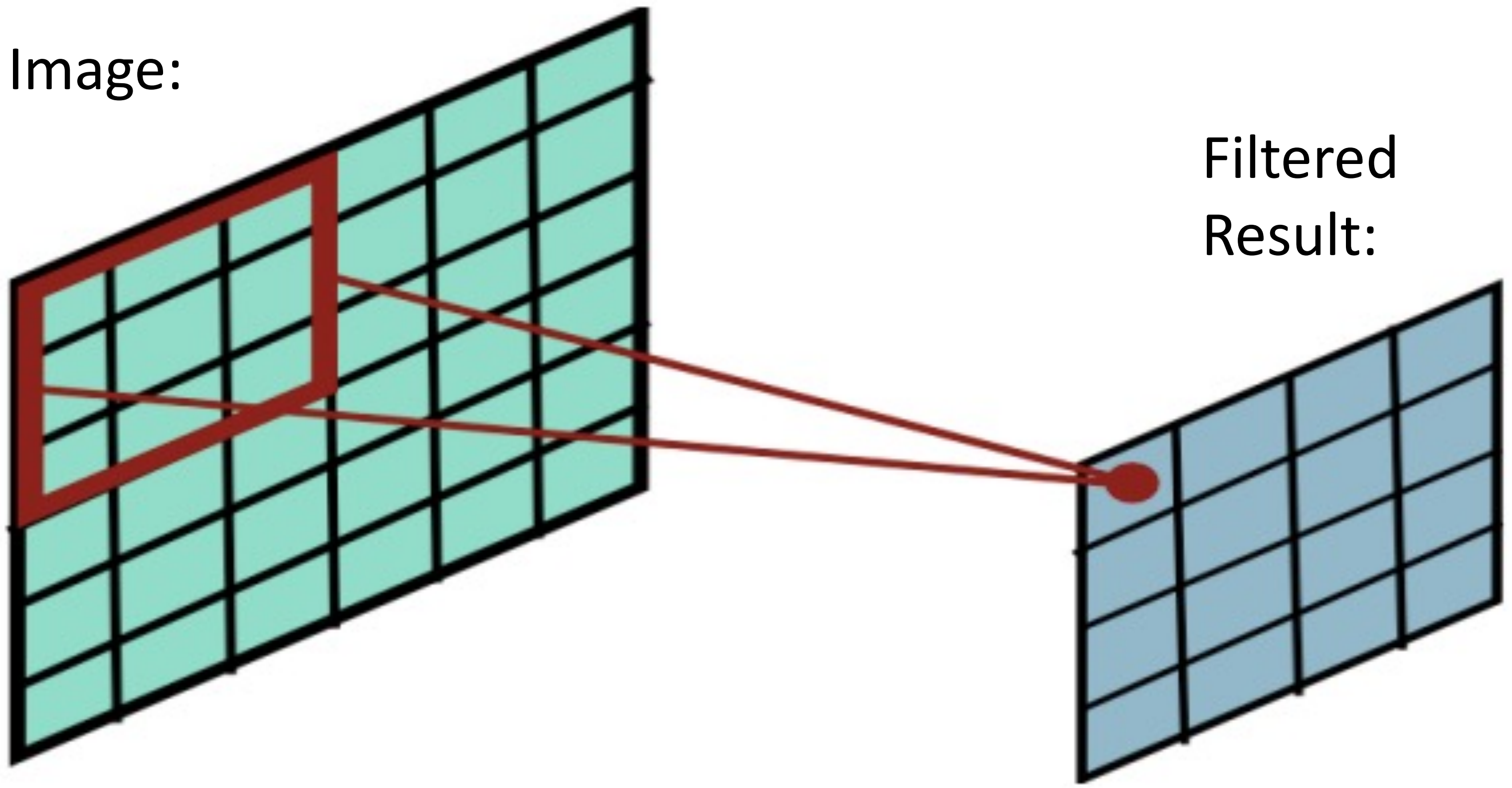


https://www.jefkine.com/general/2016/09/05/backpropagation-in-convolutional-neural-networks/

Image Filtering

- A filter specifies the function for how to combine neighbors' values
- Applying a filter to an image means computing a function of the local neighborhood for each pixel in the image

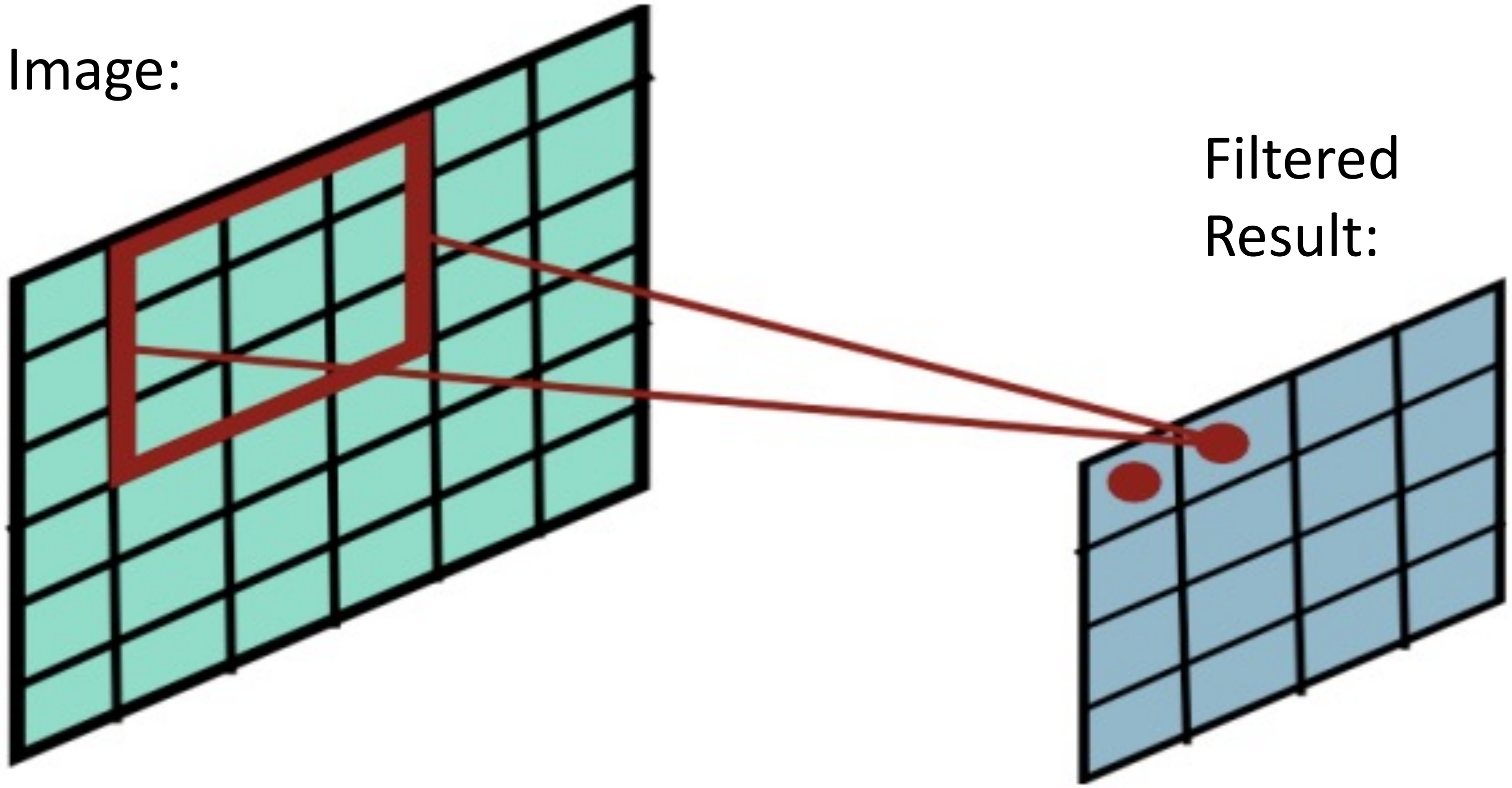
Image Filtering



Slides filter over the image and computes dot products

https://people.eecs.berkeley.edu/~jrs/189/lec/cnn.pdf

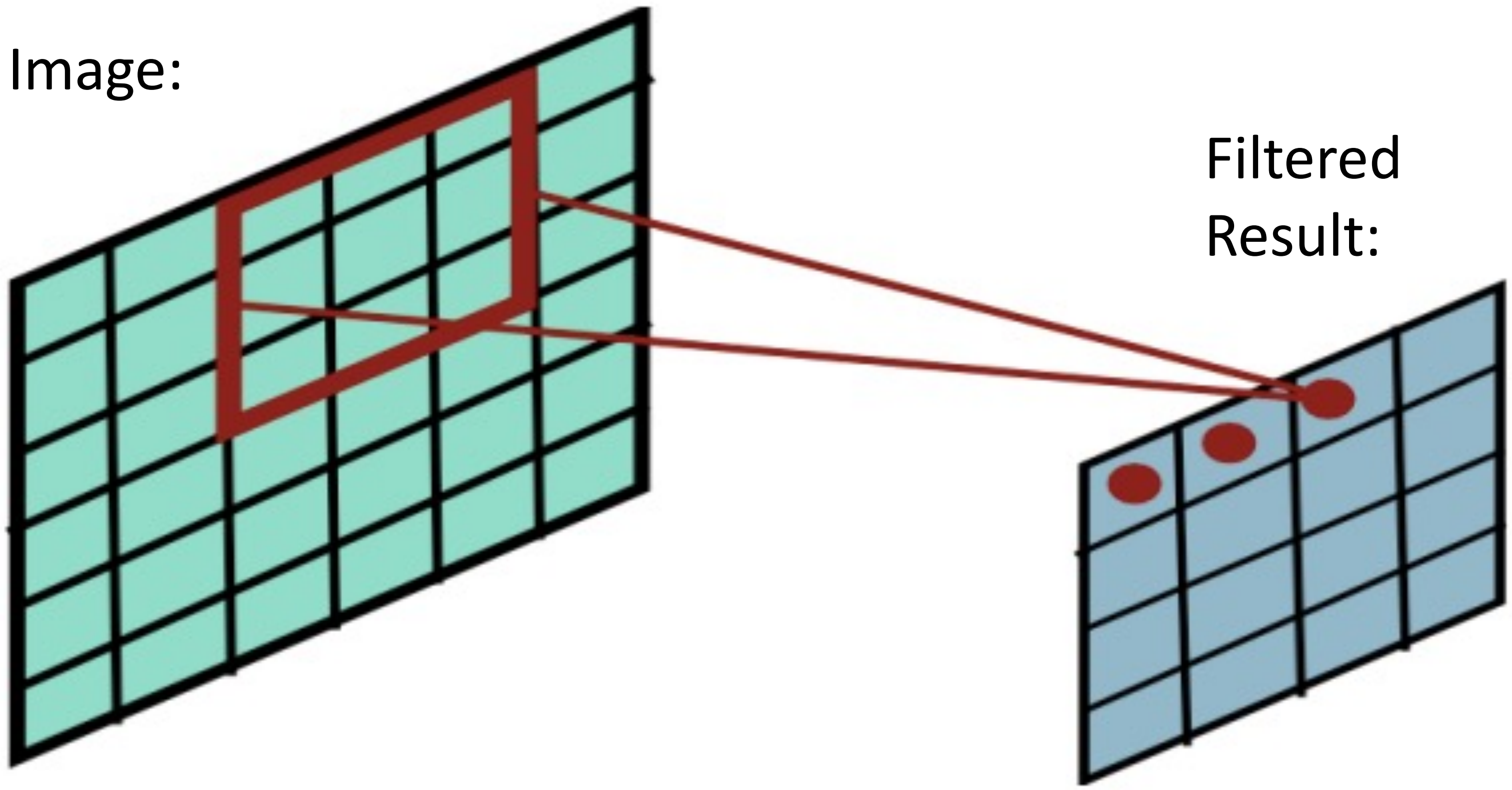
Image Filtering



Slides filter over the image and computes dot products

https://people.eecs.berkeley.edu/~jrs/189/lec/cnn.pdf

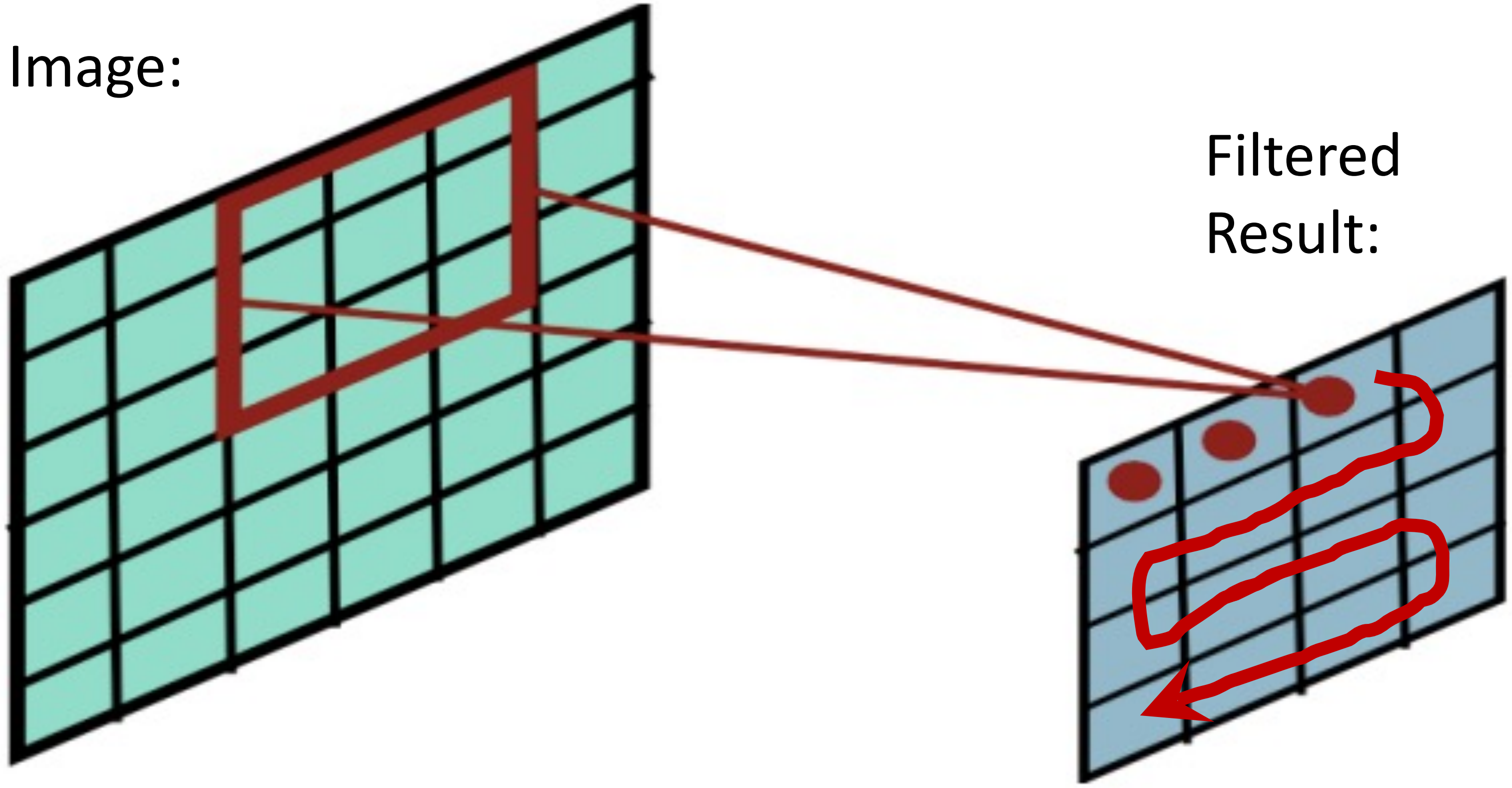
Image Filtering



Slides filter over the image and computes dot products

https://people.eecs.berkeley.edu/~jrs/189/lec/cnn.pdf

Image Filtering



Slides filter over the image and computes dot products

https://people.eecs.berkeley.edu/~jrs/189/lec/cnn.pdf

Image Filtering: Toy Example

Image

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

Filter

1	0	1
0	1	0
1	0	1

Feature Map

?	?	?
?	?	?
?	?	?

Dot Product = 1*1 + 1*0 + 1*1 + 0*0 + 1*1 + 1*0 + 0*1 + 0*1 + 0*0 + 0*0 + 1*1

Dot Product = 4

Image Filtering: Toy Example

Image

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

Filter

1	0	1
0	1	0
1	0	1

Feature Map

4	3	?
?	?	?
?	?	?

Image Filtering: Toy Example

Image

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

Filter

1	0	1
0	1	0
1	0	1

Feature Map

4	?	?
?	?	?
?	?	?

Image Filtering: Toy Example

Image

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

Filter

1	0	1
0	1	0
1	0	1

Feature Map

4	3	4
?	?	?
?	?	?

Image Filtering: Toy Example

Image

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

Filter

1	0	1
0	1	0
1	0	1

Feature Map

4	3	4
2	?	?
?	?	?

Image Filtering: Toy Example

Image

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

Filter

1	0	1
0	1	0
1	0	1

Feature Map

4	3	4
2	4	?
?	?	?

Image Filtering: Toy Example

Image

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

Filter

1	0	1
0	1	0
1	0	1

Feature Map

4	3	4
2	4	3
?	?	?

Image Filtering: Toy Example

Image

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

Filter

1	0	1
0	1	0
1	0	1

Feature Map

4	3	4
2	4	3
2	?	?

Image Filtering: Toy Example

Image

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

Filter

1	0	1
0	1	0
1	0	1

Feature Map

4	3	4
2	4	3
2	3	?

Image Filtering: Toy Example

Image

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

Filter

1	0	1
0	1	0
1	0	1

Feature Map

4	3	4
2	4	3
2	3	4

Image Filter: What Does It Do? (Where's Waldo?)

Filter

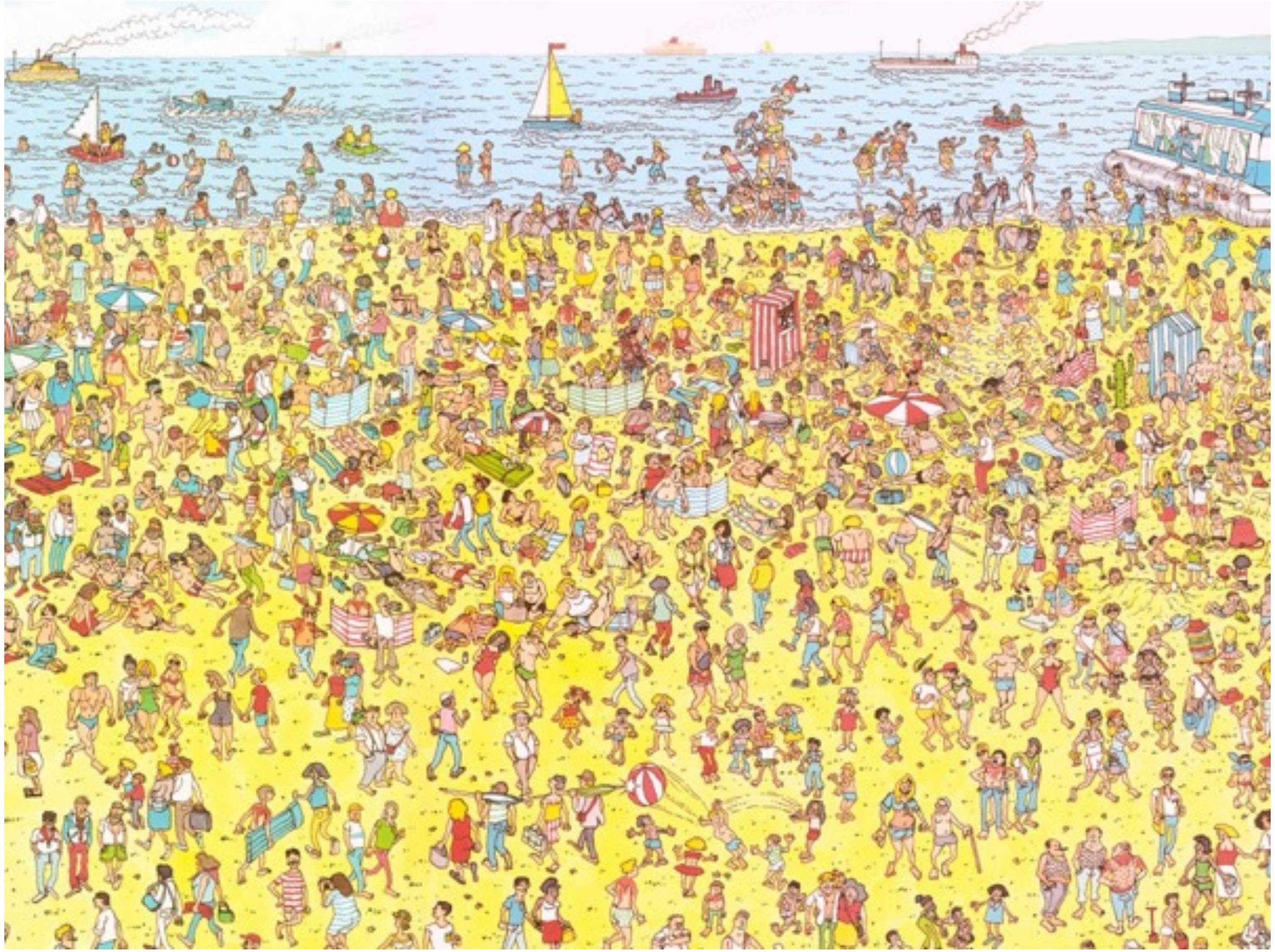


Image Filter: What Does It Do?

- e.g.,

Filter

0	0	0	0	0	30	0
0	0	0	0	30	0	0
0	0	0	30	0	0	0
0	0	0	30	0	0	0
0	0	0	30	0	0	0
0	0	0	30	0	0	0
0	0	0	0	0	0	0

Visualization of Filter

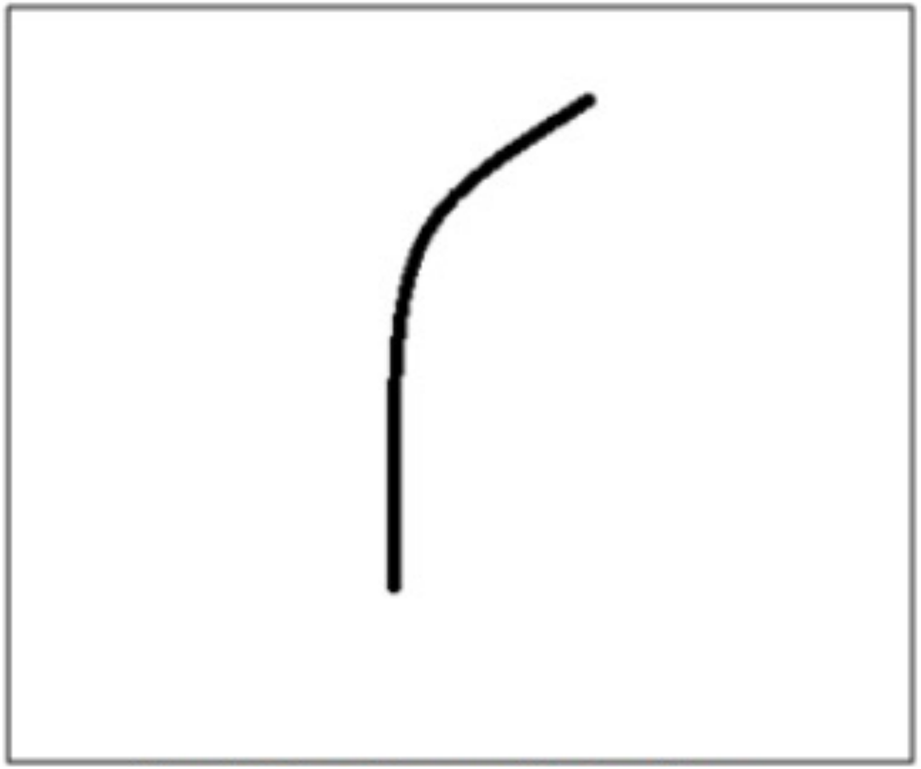
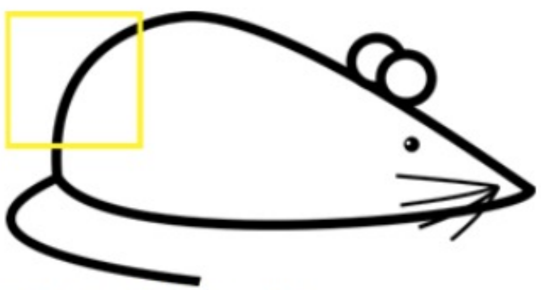


Image Credit: https://adeshpande3.github.io/A-Beginner%27s-Guide-To-Understanding-Convolutional-Neural-Networks/

Image Filter: What Does It Do?

- e.g.,

Filter Overlaid on Image



Image

0	0	0	0	0	0	30
0	0	0	0	50	50	50
0	0	0	20	50	0	0
0	0	0	50	50	0	0
0	0	0	50	50	0	0
0	0	0	50	50	0	0
0	0	0	50	50	0	0

*

Filter

0	0	0	0	0	30	0
0	0	0	0	30	0	0
0	0	0	30	0	0	0
0	0	0	30	0	0	0
0	0	0	30	0	0	0
0	0	0	30	0	0	0
0	0	0	0	0	0	0

Weighted Sum = ?

Weighted Sum = (50x30) + (20x30) + (50x30) + (50x3) + (50x30)

Weighted Sum = 6600 **(Large Number!!)**

Image Credit: https://adeshpande3.github.io/A-Beginner%27s-Guide-To-Understanding-Convolutional-Neural-Networks/

Image Filter: What Does It Do?

- e.g.,

Filter Overlaid on Image

Image

0	0	0	0	0	0	0
0	40	0	0	0	0	0
40	0	40	0	0	0	0
40	20	0	0	0	0	0
0	50	0	0	0	0	0
0	0	50	0	0	0	0
25	25	0	50	0	0	0

$*$

Filter

0	0	0	0	0	30	0
0	0	0	0	30	0	0
0	0	0	30	0	0	0
0	0	0	30	0	0	0
0	0	0	30	0	0	0
0	0	0	30	0	0	0
0	0	0	0	0	0	0

Weighted Sum = ?

Weighted Sum = 0 **(Small Number!!)**

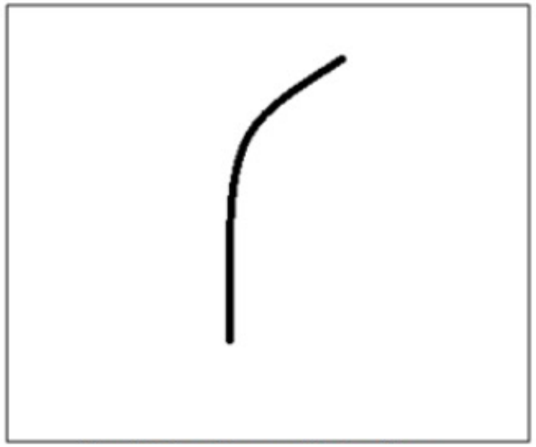
Image Credit: https://adeshpande3.github.io/A-Beginner%27s-Guide-To-Understanding-Convolutional-Neural-Networks/

Image Filter: What Does It Do?

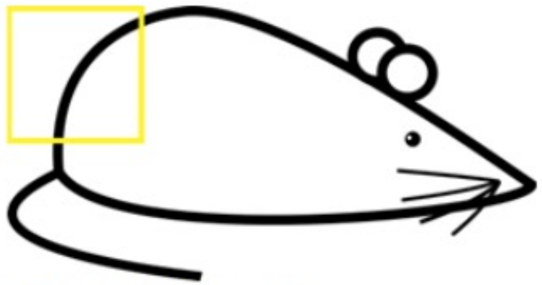
- e.g.,

This Filter is a Curve Detector!

0	0	0	0	0	30	0
0	0	0	0	30	0	0
0	0	0	30	0	0	0
0	0	0	30	0	0	0
0	0	0	30	0	0	0
0	0	0	30	0	0	0
0	0	0	0	0	0	0



Filter Overlaid on Image (Big Response!)



Filter Overlaid on Image (Small Response!)

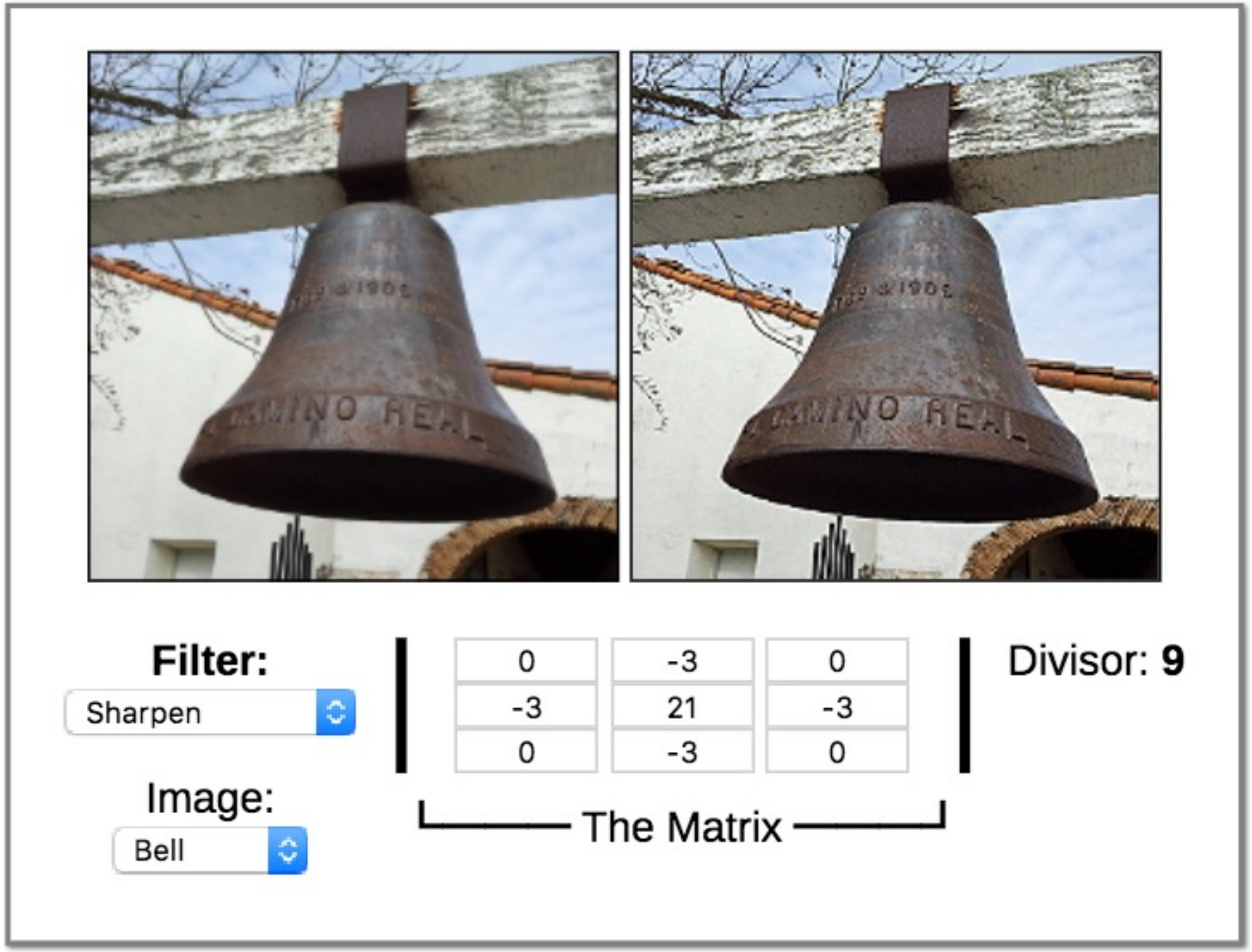
Image Credit: https://adeshpande3.github.io/A-Beginner%27s-Guide-To-Understanding-Convolutional-Neural-Networks/

Different Filters Detect Different Features

Operation	Filter	Feature Map
Identity	$\begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}$	
Edge detection	$\begin{bmatrix} 1 & 0 & -1 \\ 0 & 0 & 0 \\ -1 & 0 & 1 \end{bmatrix}$	
	$\begin{bmatrix} 0 & 1 & 0 \\ 1 & -4 & 1 \\ 0 & 1 & 0 \end{bmatrix}$	
	$\begin{bmatrix} -1 & -1 & -1 \\ -1 & 8 & -1 \\ -1 & -1 & -1 \end{bmatrix}$	
Sharpen	$\begin{bmatrix} 0 & -1 & 0 \\ -1 & 5 & -1 \\ 0 & -1 & 0 \end{bmatrix}$	
Box blur (normalized)	$\frac{1}{9}\begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix}$	
Gaussian blur (approximation)	$\frac{1}{16}\begin{bmatrix} 1 & 2 & 1 \\ 2 & 4 & 2 \\ 1 & 2 & 1 \end{bmatrix}$	

https://ujjwalkarn.me/2016/08/11/intuitive-explanation-convnets/

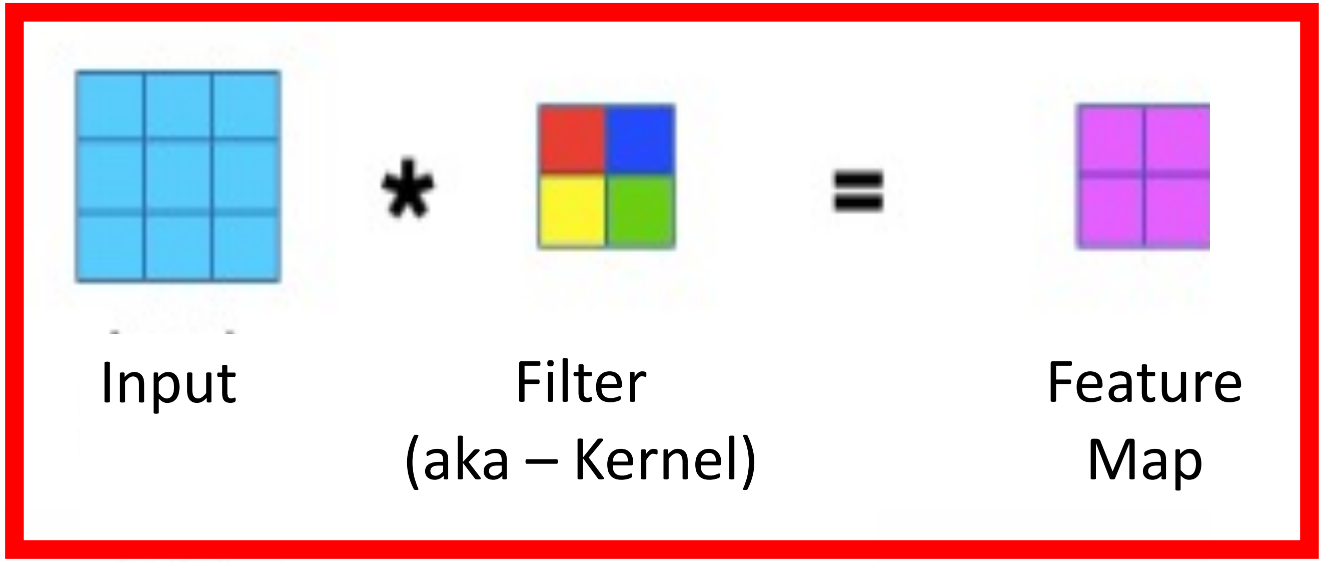
Different Filters Detect Different Features



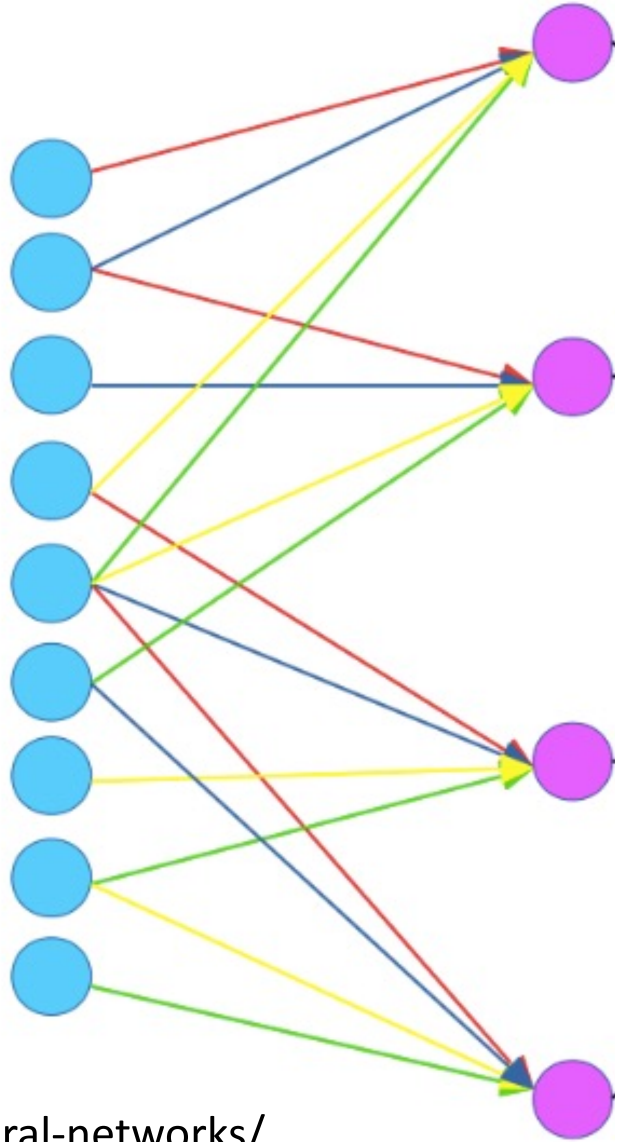
Demo: http://beej.us/blog/data/convolution-image-processing/

Convolutional Layer: Applies Linear Filter

- Note, previous examples show the "cross-correlation" function
- Many neural network libraries use "cross correlation" interchangeably with "convolution"; for mathematicians, these are technically different



Way to Interpret Neural Network



https://www.jefkine.com/general/2016/09/05/backpropagation-in-convolutional-neural-networks/

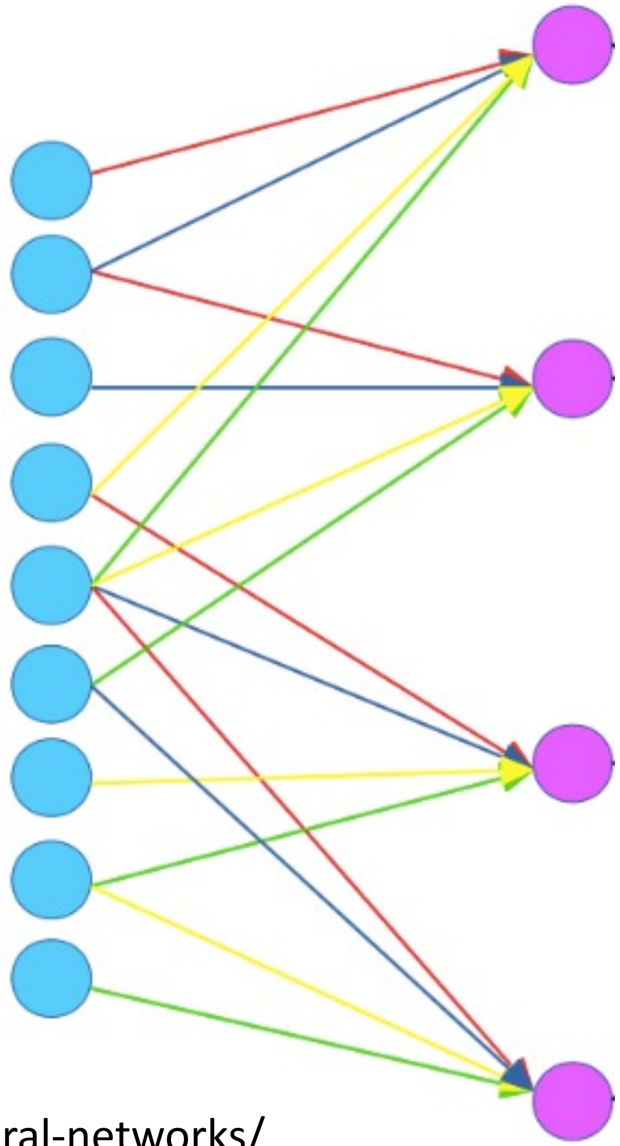
Convolutional Layer: Applies Linear Filter

Input

Filter
(aka – Kernel)

Feature
Map

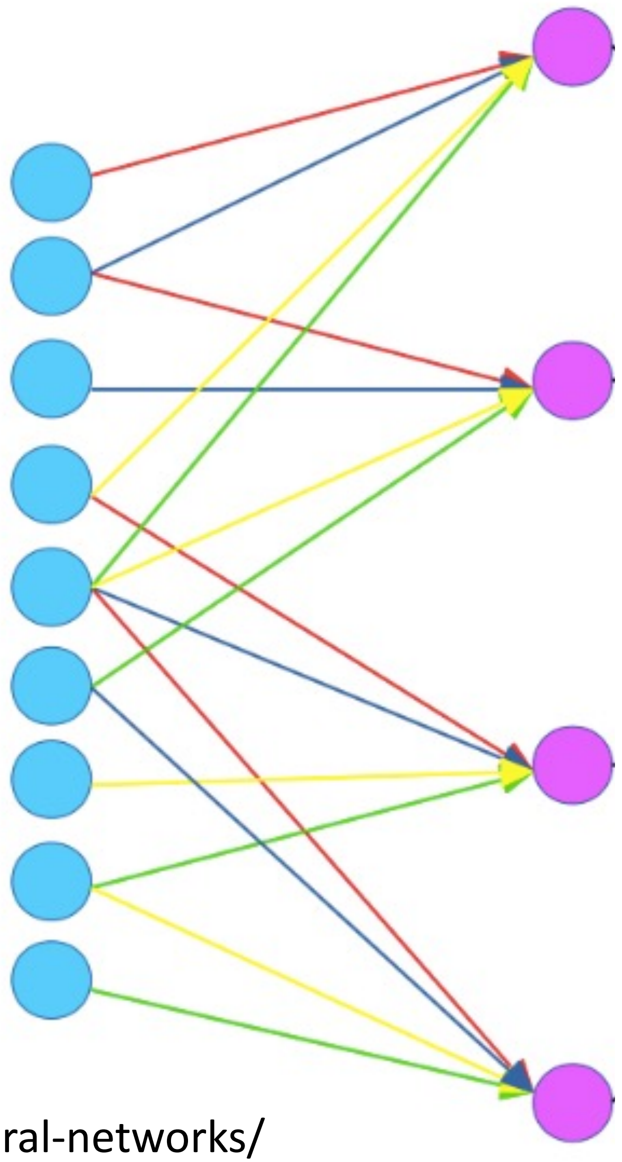
Way to Interpret
Neural Network



https://www.jefkine.com/general/2016/09/05/backpropagation-in-convolutional-neural-networks/

Convolutional Layer: Parameters to Learn

- For shown example, how many weights must be learned?
  - 4 (red, blue, yellow, and green values)
- If we instead used a fully connected layer, how many weights would need to be learned?
  - 36 (9 turquoise nodes x 4 magenta nodes)
- For shown example, how many parameters must be learned
  - 5 (4 weights + 1 bias)
- If we instead used a fully connected layer, how many parameters would need to be learned?
  - 40 (36 weights + 4 bias)

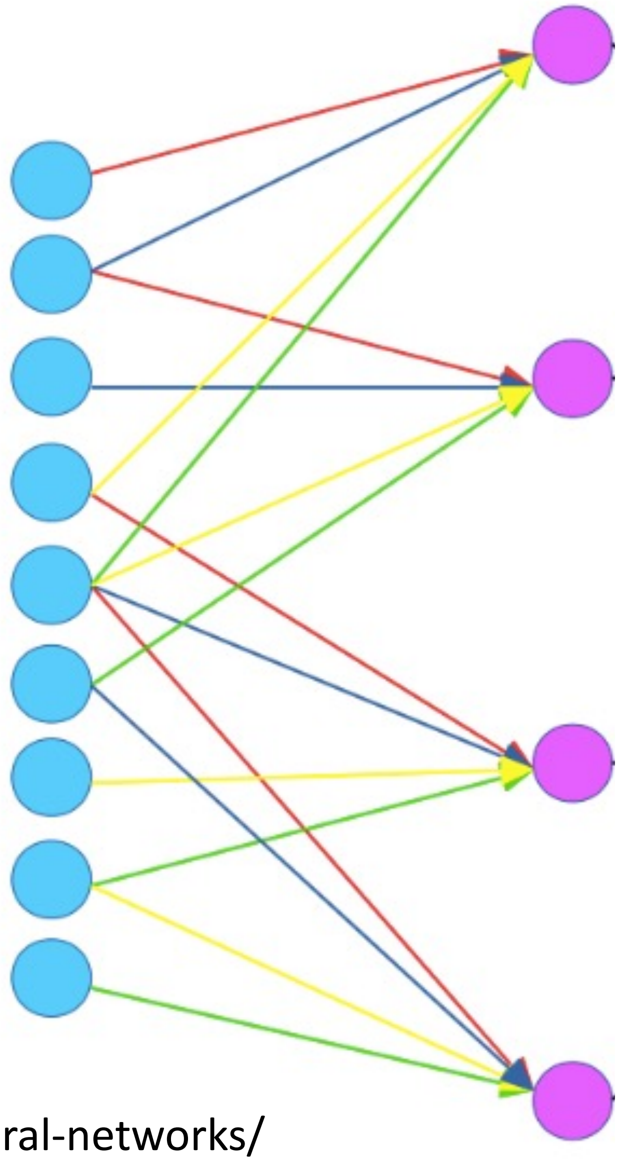


https://www.jefkine.com/general/2016/09/05/backpropagation-in-convolutional-neural-networks/

Convolutional Layer: Parameters to Learn

- Parameter sharing significantly reduces number of weights to learn and so storage requirements

- Sparse (rather than full) connectivity also significantly reduces the number of computational operations required



https://www.jefkine.com/general/2016/09/05/backpropagation-in-convolutional-neural-networks/

Convolutional Layer: Implementation Details

- **Padding**: add values at the image boundaries to preserve image size

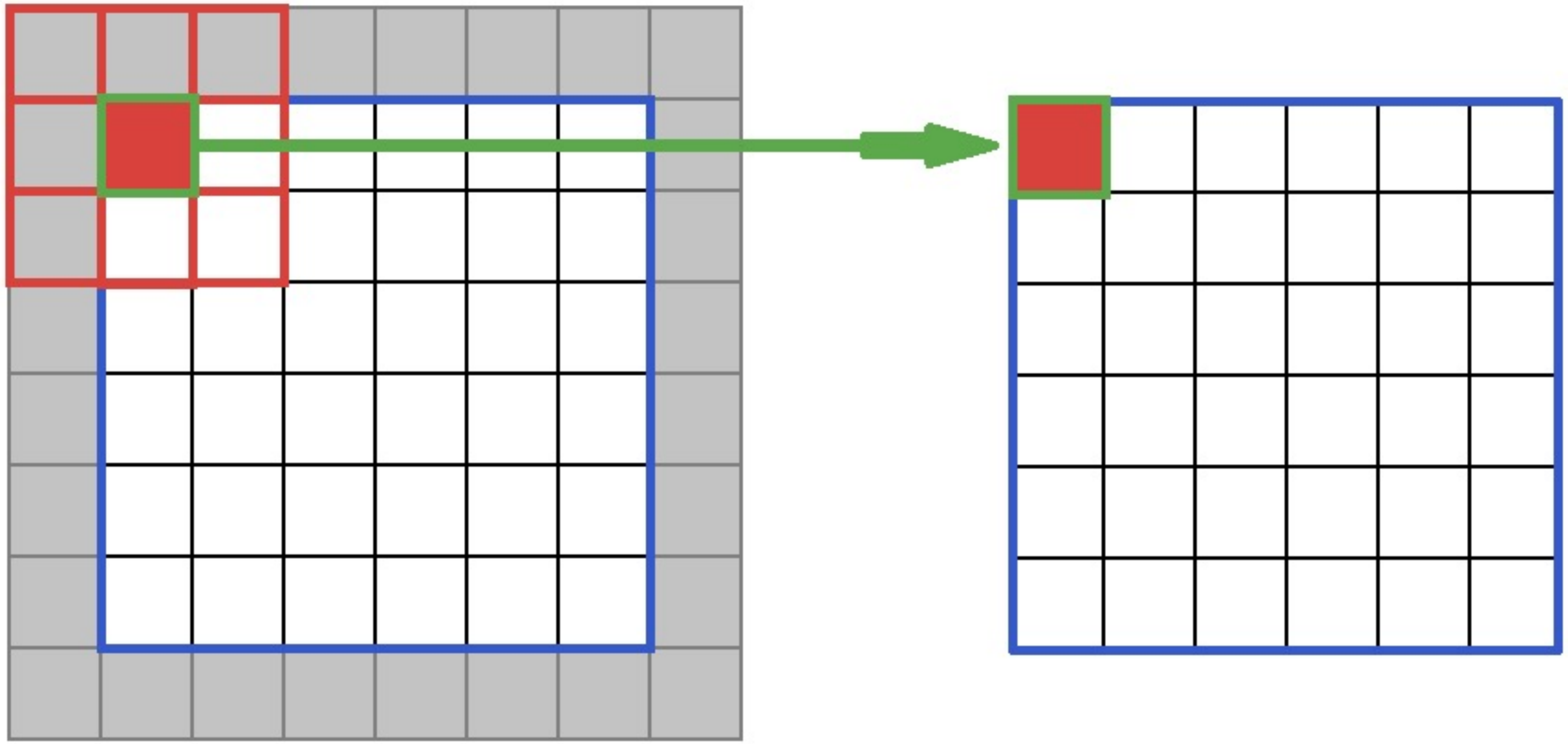
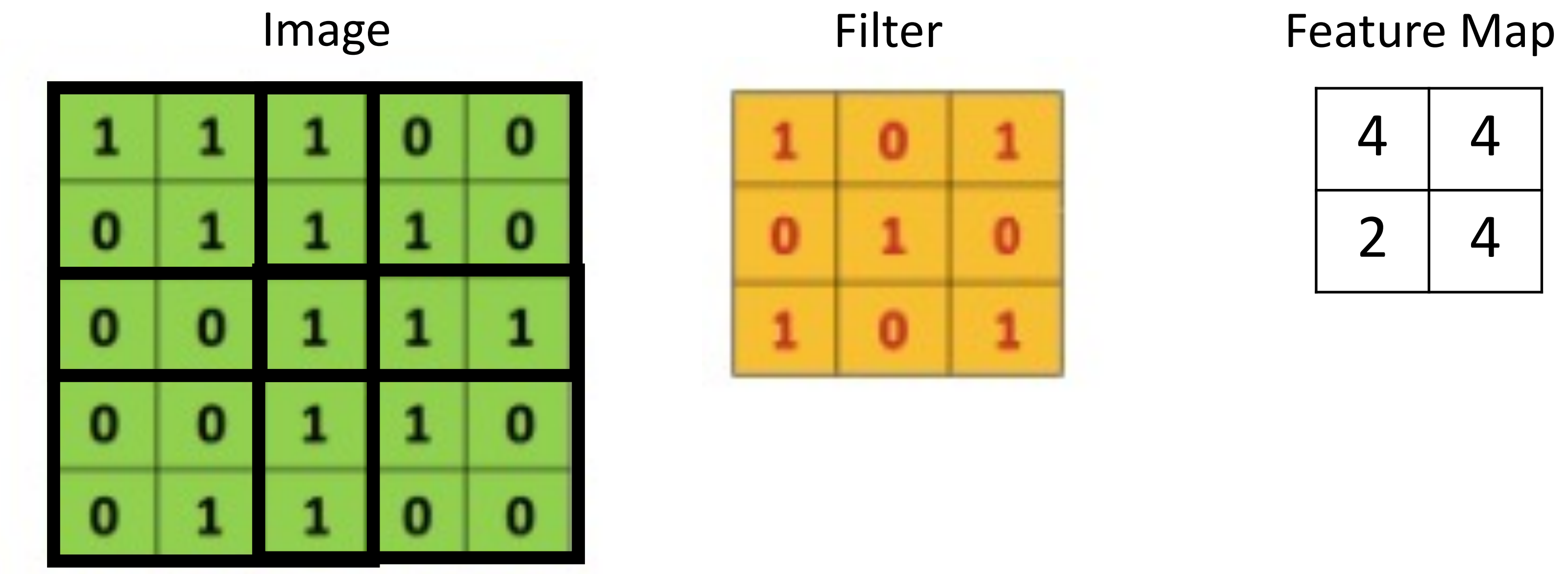


Image Credit: https://software.intel.com/en-us/node/586159

Convolutional Layer: Implementation Details

- **Stride**: how many steps taken spatially before applying a filter
  - e.g., 2x2

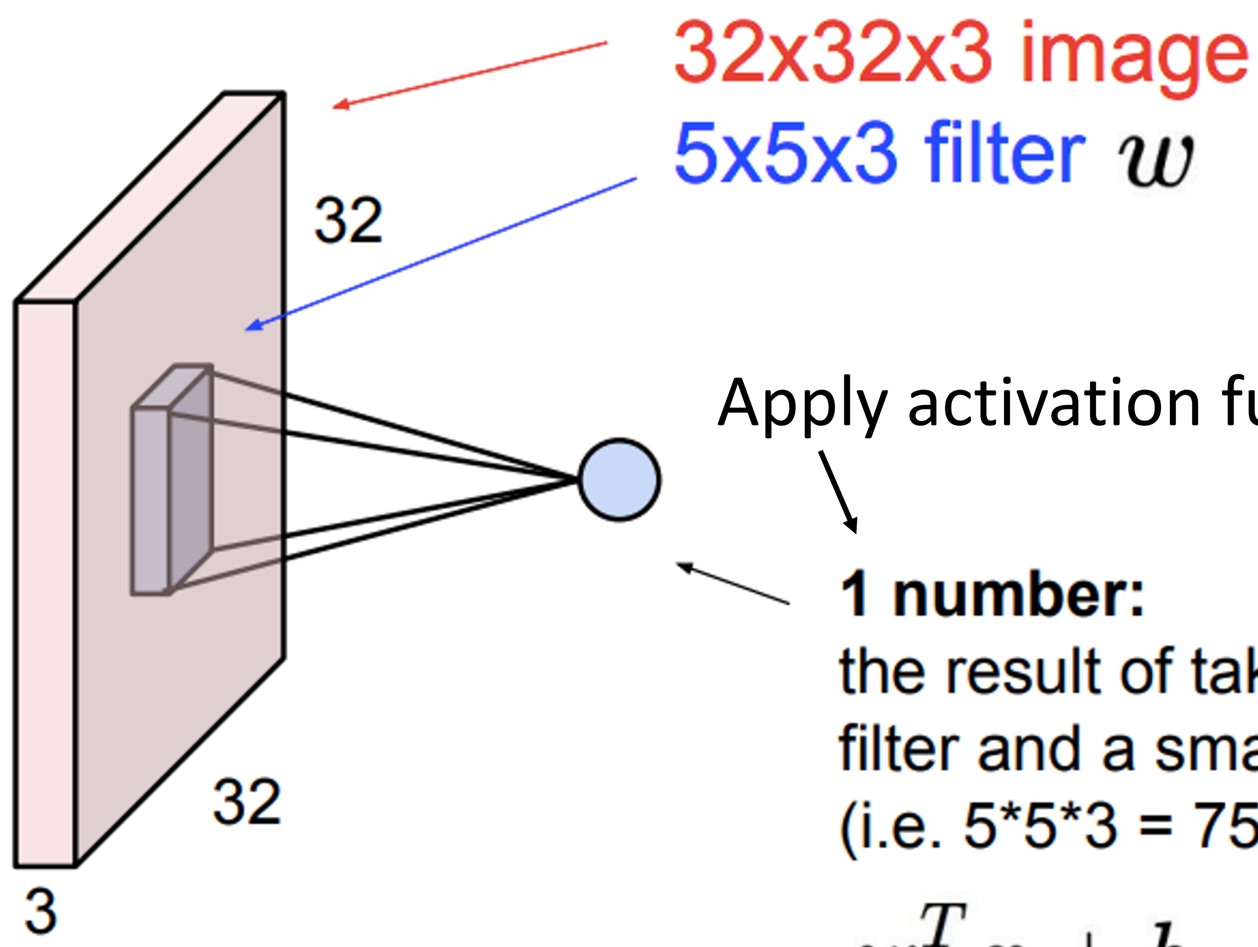


http://deeplearning.net/software/theano/tutorial/conv_arithmetic.html

Convolutional Layer: Implementation Details

- Demo:
  - https://theano-pymc.readthedocs.io/en/latest/tutorial/conv_arithmetic.html

Convolutional Layer: Introduce Non-Linearity



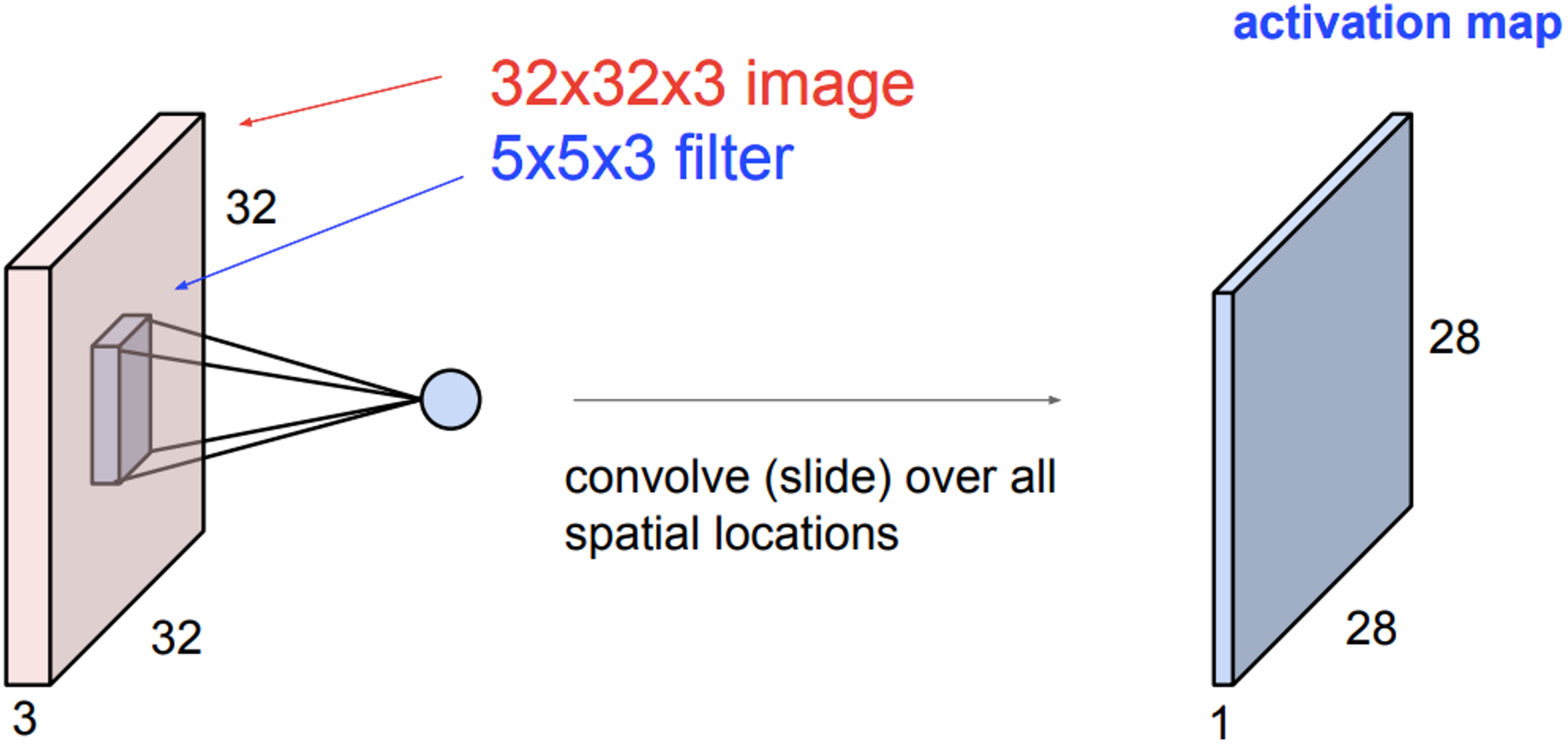
Apply activation function to number

1 number:
the result of taking a dot product between the filter and a small 5x5x3 chunk of the image
(i.e. 5*5*3 = 75-dimensional dot product + bias)

$$w^T x + b$$

Slide Credit: http://cs231n.stanford.edu/slides/2018/cs231n_2018_lecture05.pdf

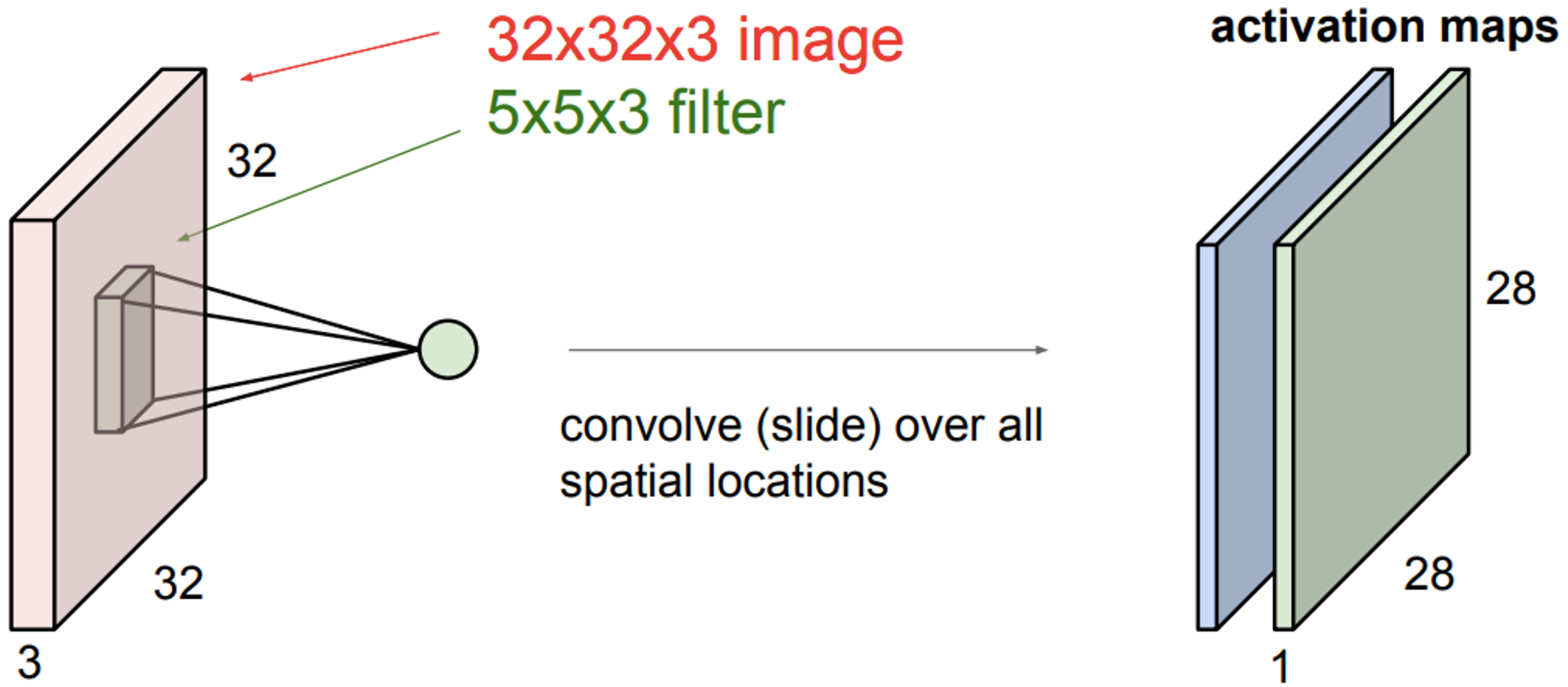
Convolutional Layer



Slide Credit: http://cs231n.stanford.edu/slides/2018/cs231n_2018_lecture05.pdf

Convolutional Layer

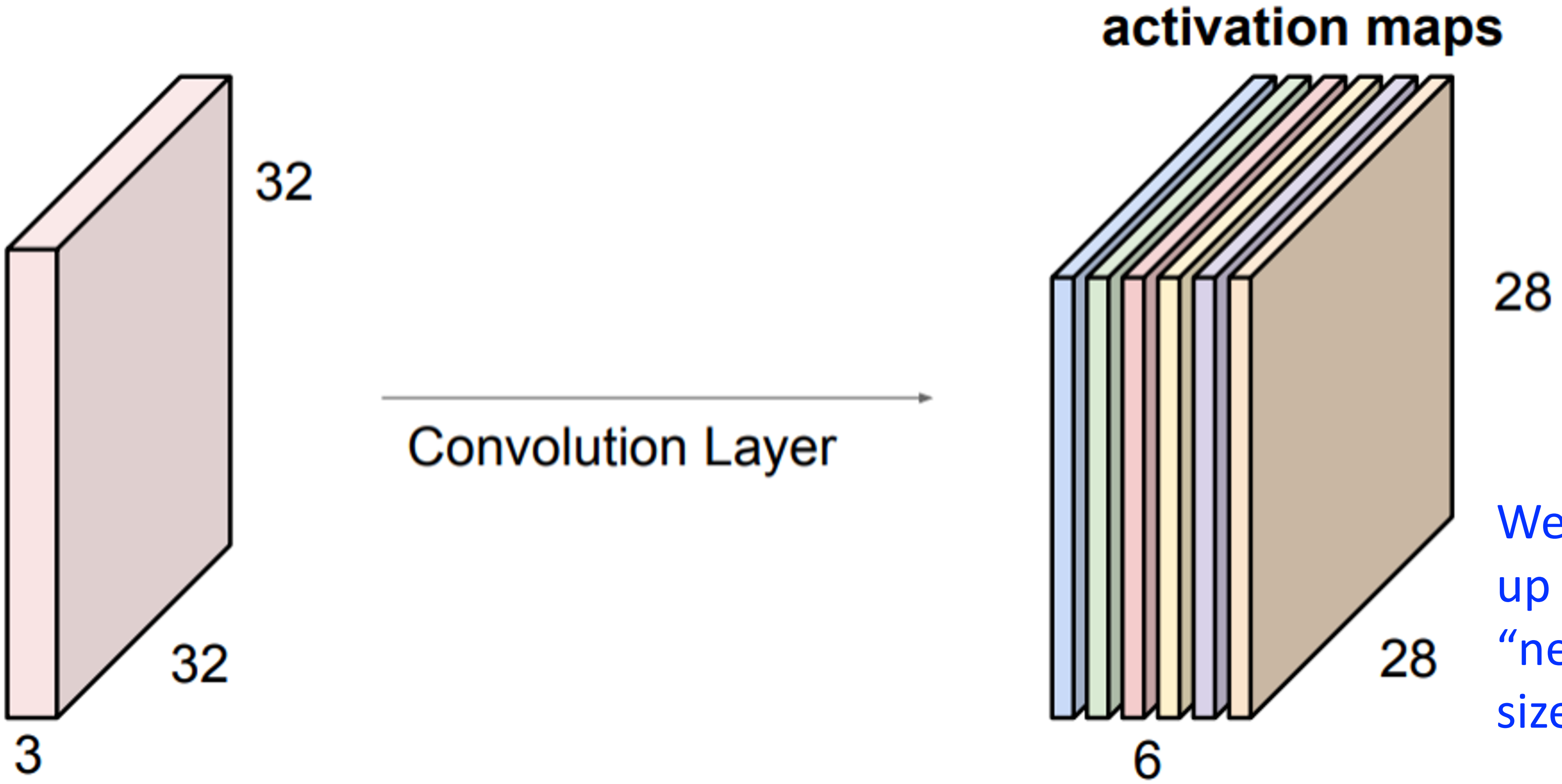
consider a second, green filter



Slide Credit: http://cs231n.stanford.edu/slides/2018/cs231n_2018_lecture05.pdf

Convolutional Layer

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

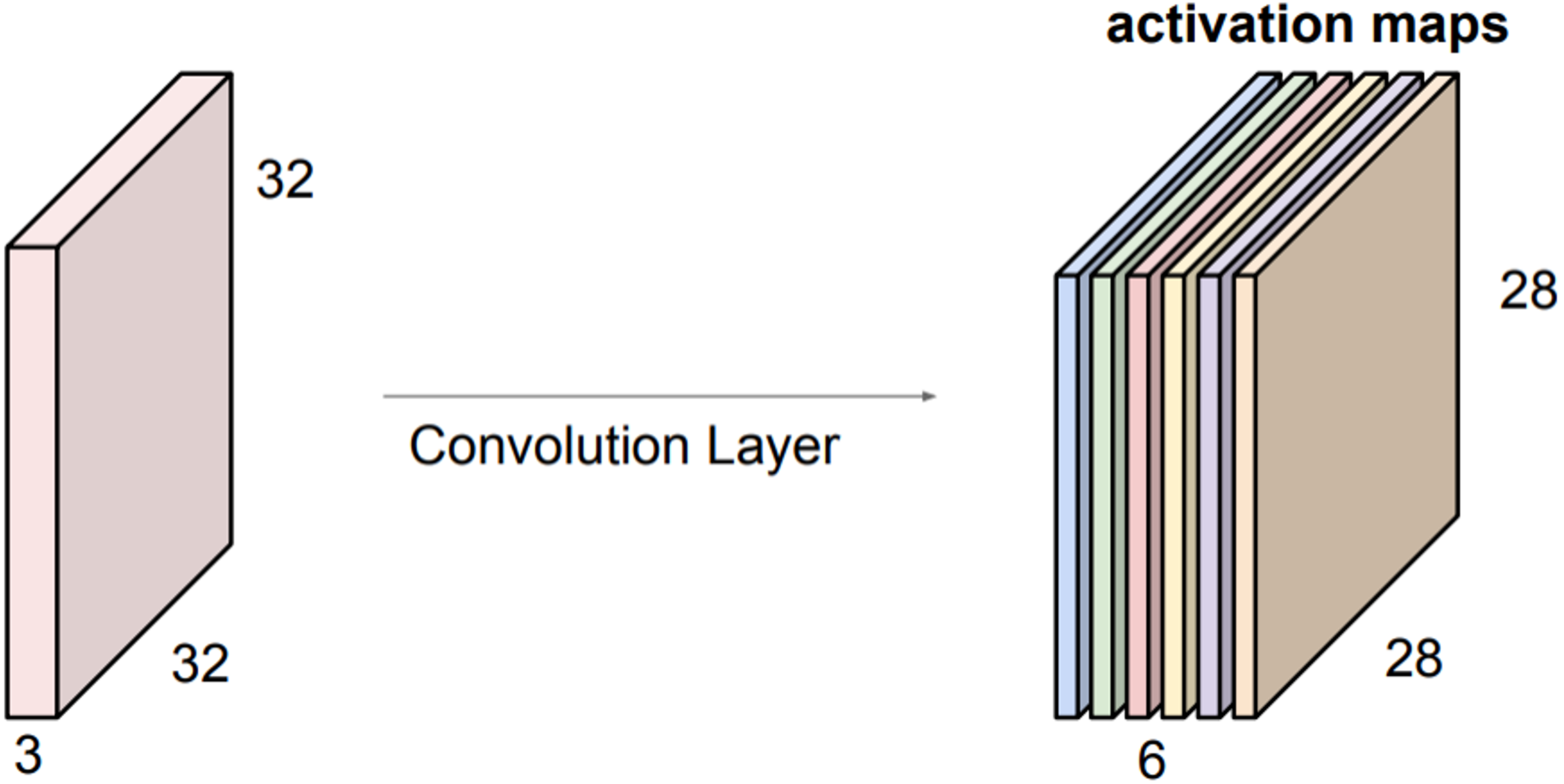


We stack these up to get a "new image" of size 28x28x6!

Slide Credit: http://cs231n.stanford.edu/slides/2018/cs231n_2018_lecture05.pdf

Convolutional Layer: Parameters to Learn

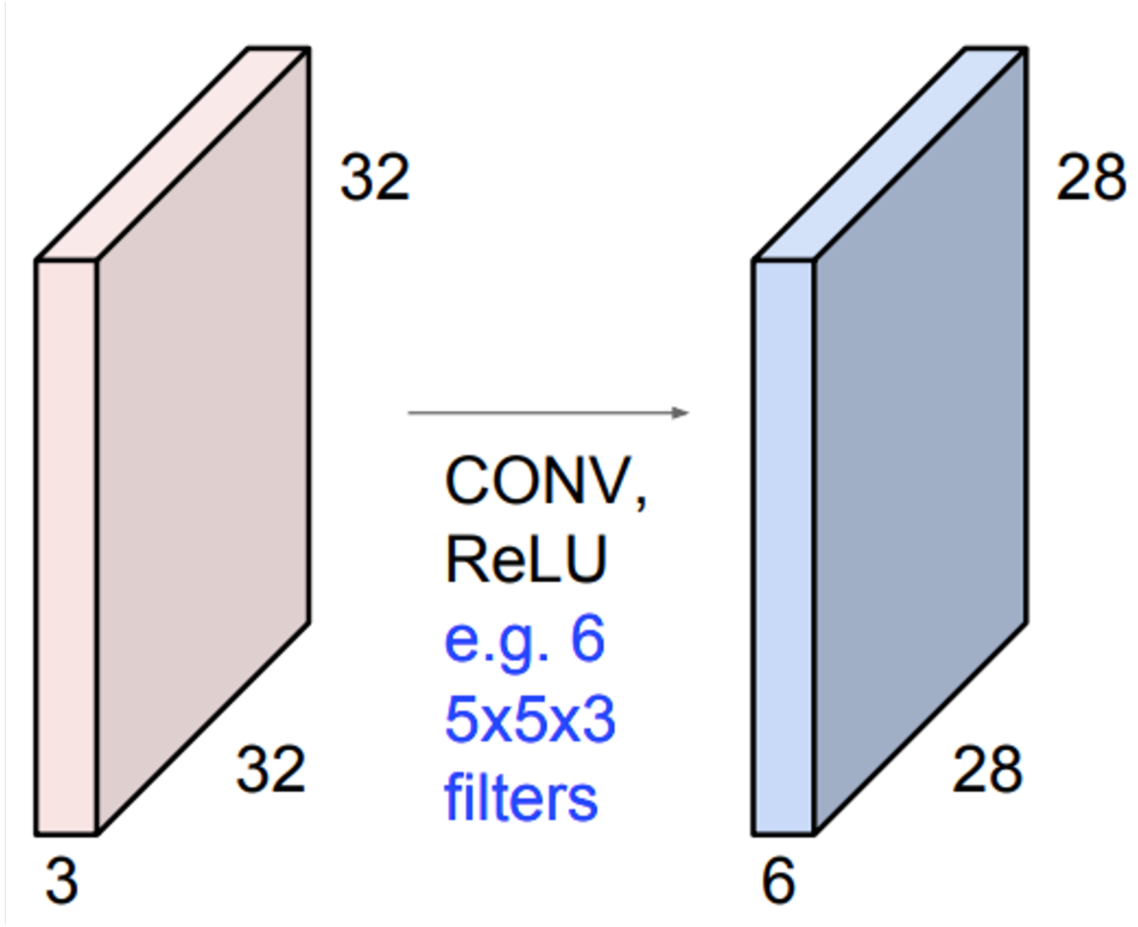
Parameters: bank of filters and biases used to create the activation maps (aka – feature maps)



Slide Credit: http://cs231n.stanford.edu/slides/2018/cs231n_2018_lecture05.pdf

Convolutional Neural Networks (CNNs)

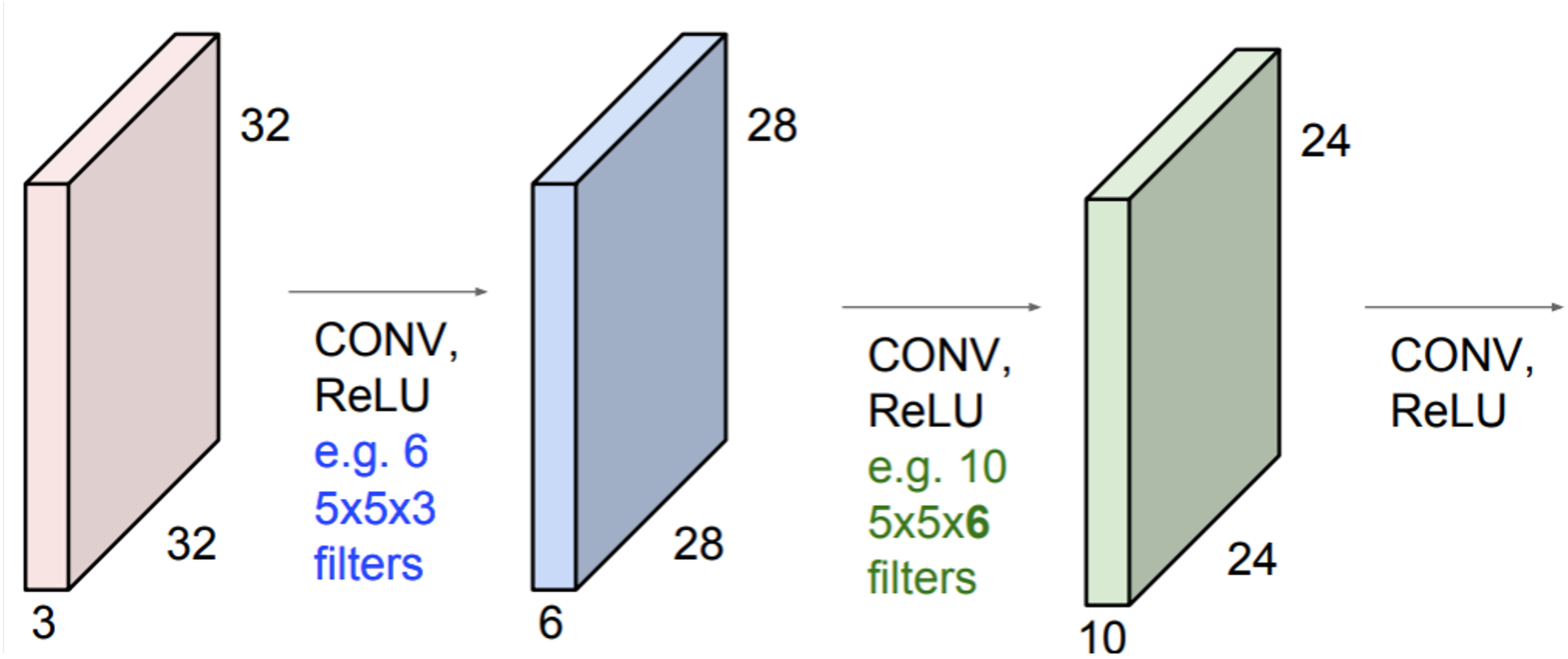
Can then stack a sequence of convolution layers, interspersed with activation functions:



Slide Credit: http://cs231n.stanford.edu/slides/2018/cs231n_2018_lecture05.pdf

Convolutional Neural Networks (CNNs)

Can then stack a sequence of convolution layers, interspersed with activation functions:

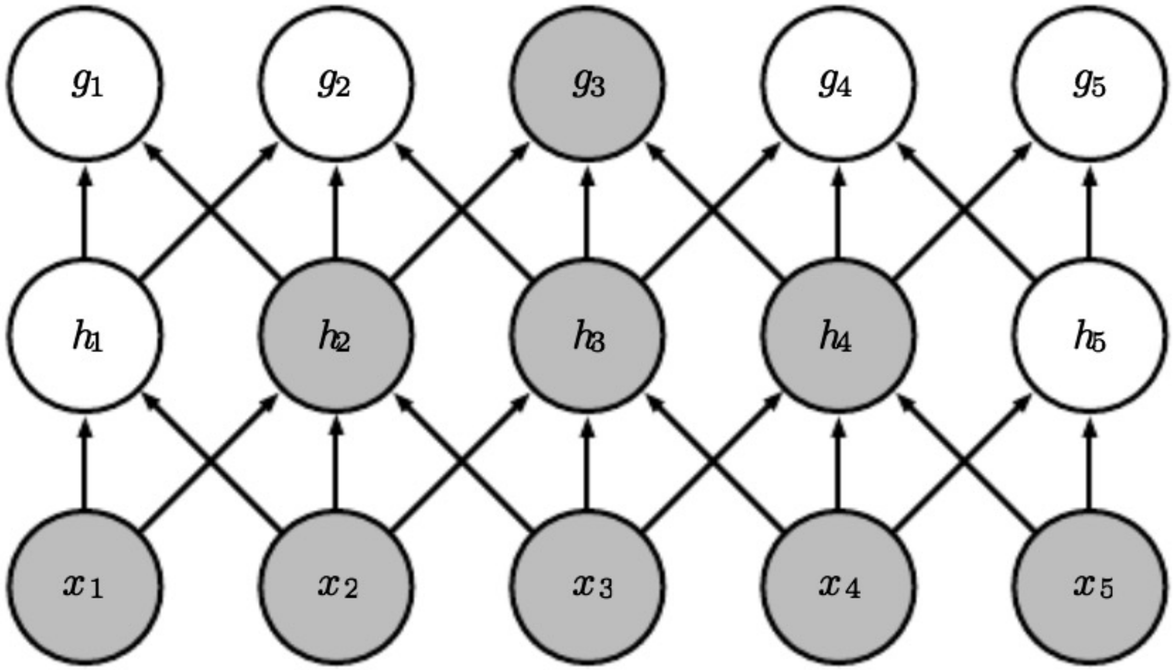


Slide Credit: http://cs231n.stanford.edu/slides/2018/cs231n_2018_lecture05.pdf

Convolutional Neural Networks (CNNs)

Can then stack a sequence of convolution layers, interspersed with activation functions:

Stacking many convolutional layers leads to learning patterns in increasingly **larger regions of the input (e.g., pixel) space.**



https://www.deeplearningbook.org/contents/convnets.html

CNN: Pooling Layers

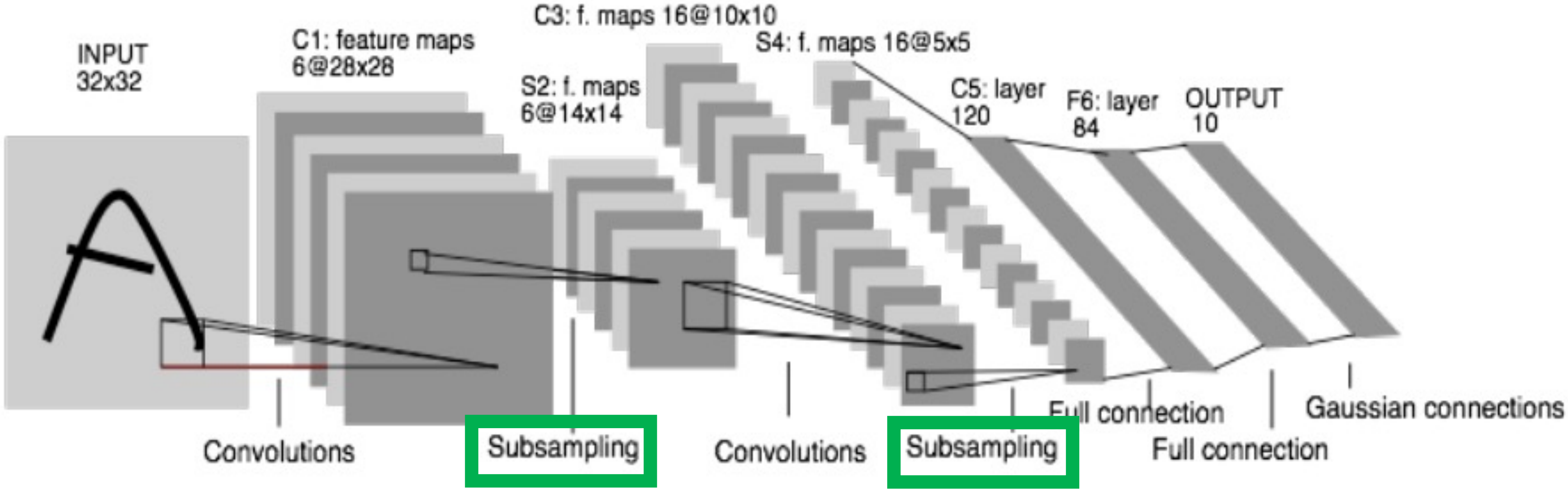
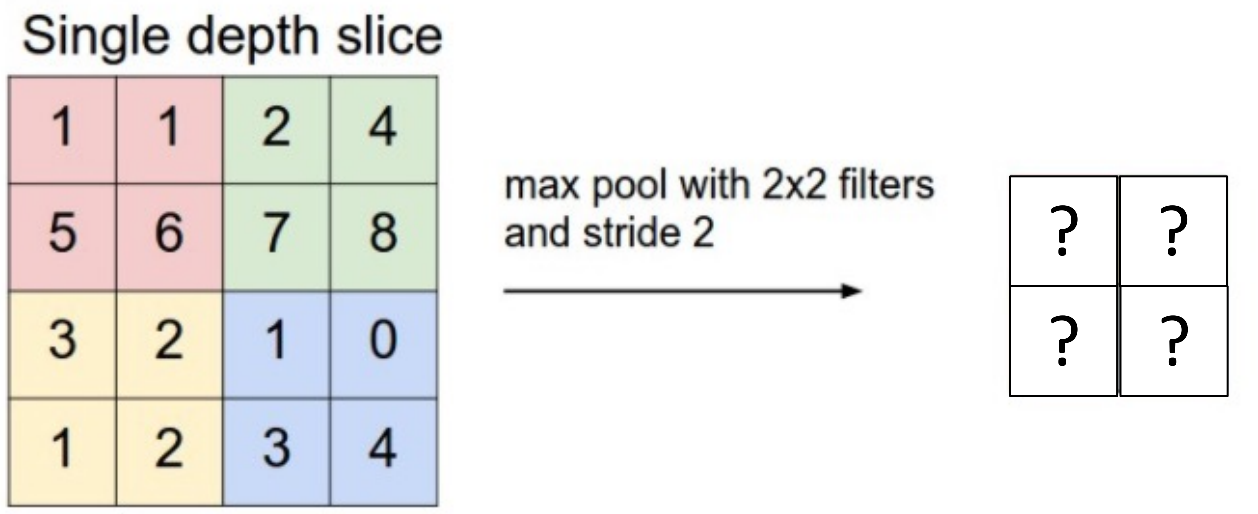


Figure Credit: https://people.eecs.berkeley.edu/~jrs/189/lec/cnn.pdf

Y. Lecun ; L. Bottou ; Y. Bengio ; P. Haffner; Gradient-based learning applied to document recognition; 1998

Pooling Layer: Summarizes Neighborhood

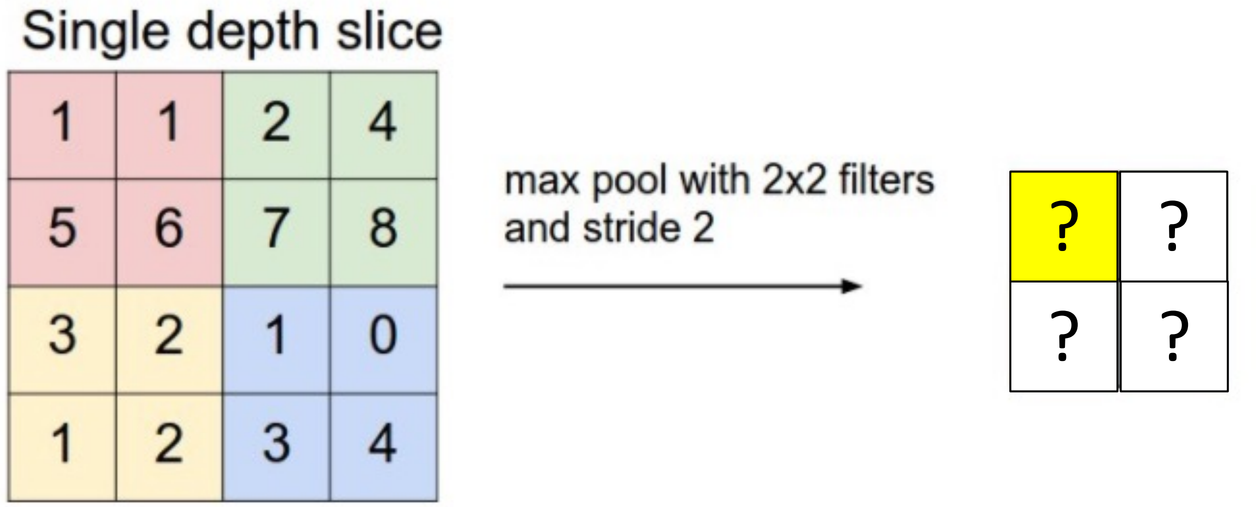
- **Max-pooling***:* partitions input into a set of non-overlapping rectangles and outputs the maximum value for each chunk



http://cs231n.github.io/convolutional-networks/#pool

Pooling Layer: Summarizes Neighborhood

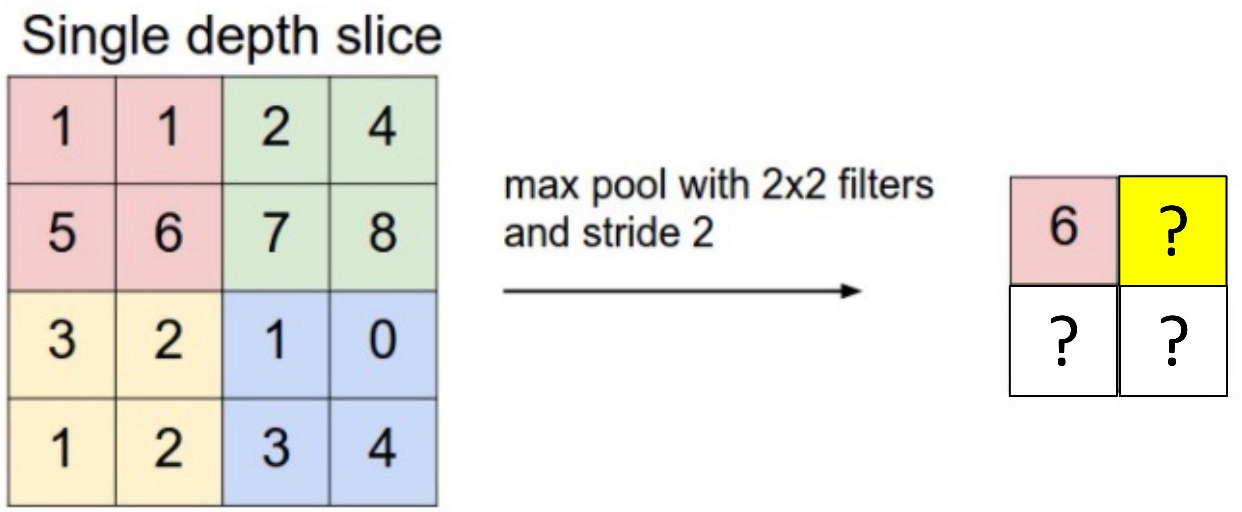
- **Max-pooling**: partitions input into a set of non-overlapping rectangles and outputs the maximum value for each chunk



http://cs231n.github.io/convolutional-networks/#pool

Pooling Layer: Summarizes Neighborhood

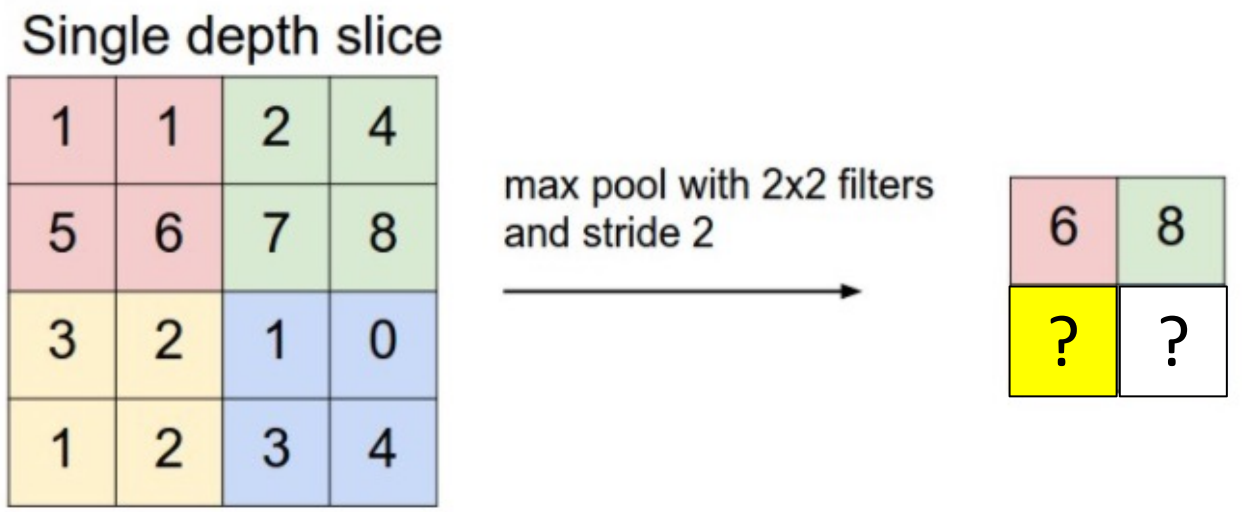
- **Max-pooling**: partitions input into a set of non-overlapping rectangles and outputs the maximum value for each chunk



http://cs231n.github.io/convolutional-networks/#pool

Pooling Layer: Summarizes Neighborhood

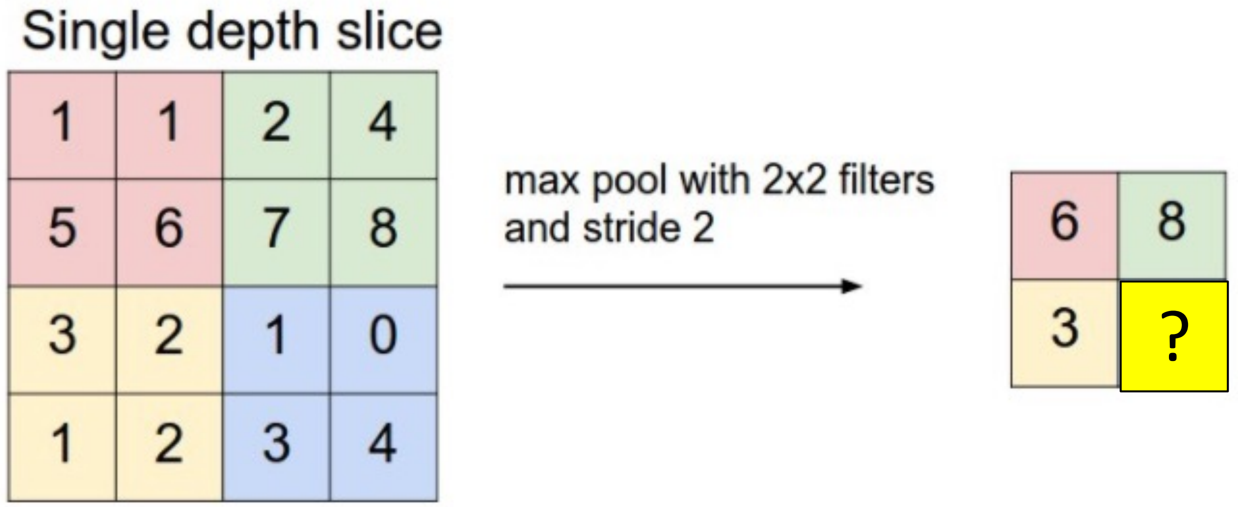
- **Max-pooling**: partitions input into a set of non-overlapping rectangles and outputs the maximum value for each chunk



http://cs231n.github.io/convolutional-networks/#pool

Pooling Layer: Summarizes Neighborhood

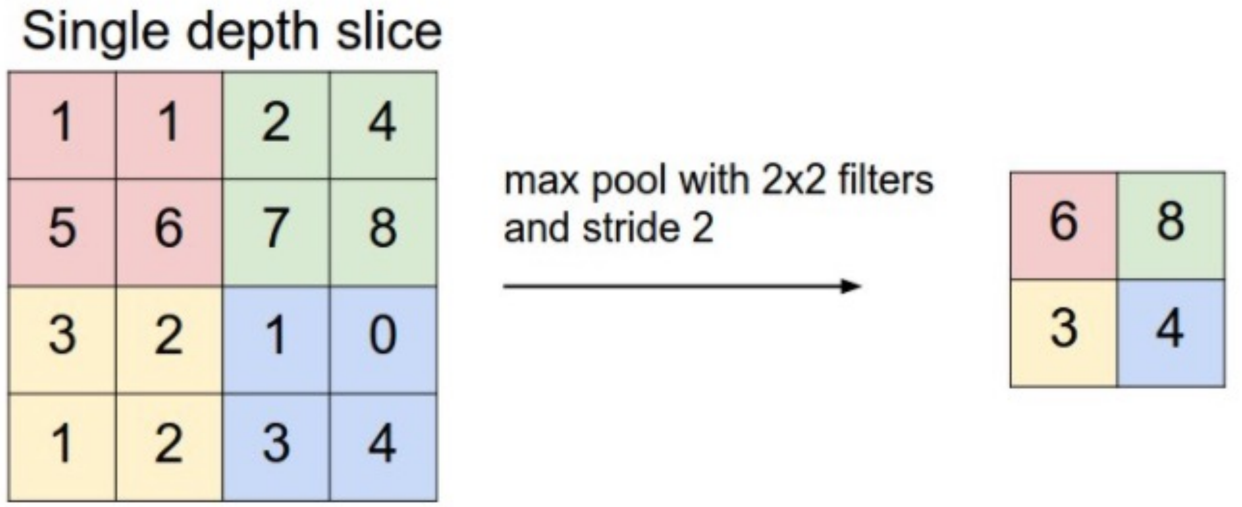
- **Max-pooling***:* partitions input into a set of non-overlapping rectangles and outputs the maximum value for each chunk



http://cs231n.github.io/convolutional-networks/#pool

Pooling Layer: Summarizes Neighborhood

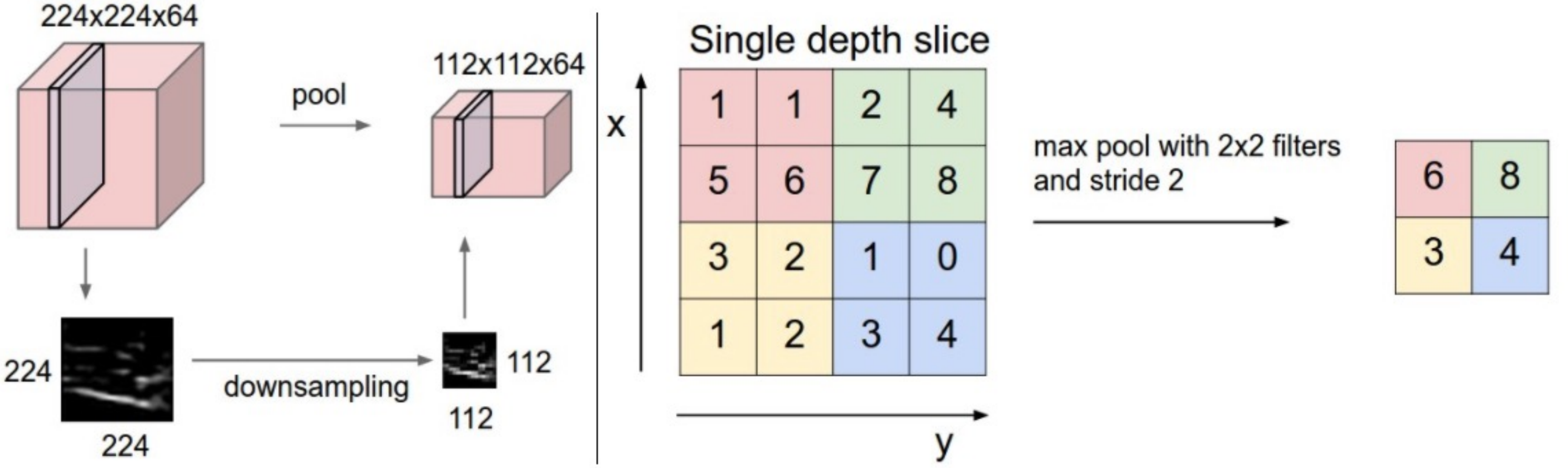
- **Max-pooling***:* partitions input into a set of non-overlapping rectangles and outputs the maximum value for each chunk



http://cs231n.github.io/convolutional-networks/#pool

Pooling Layer: Summarizes Neighborhood

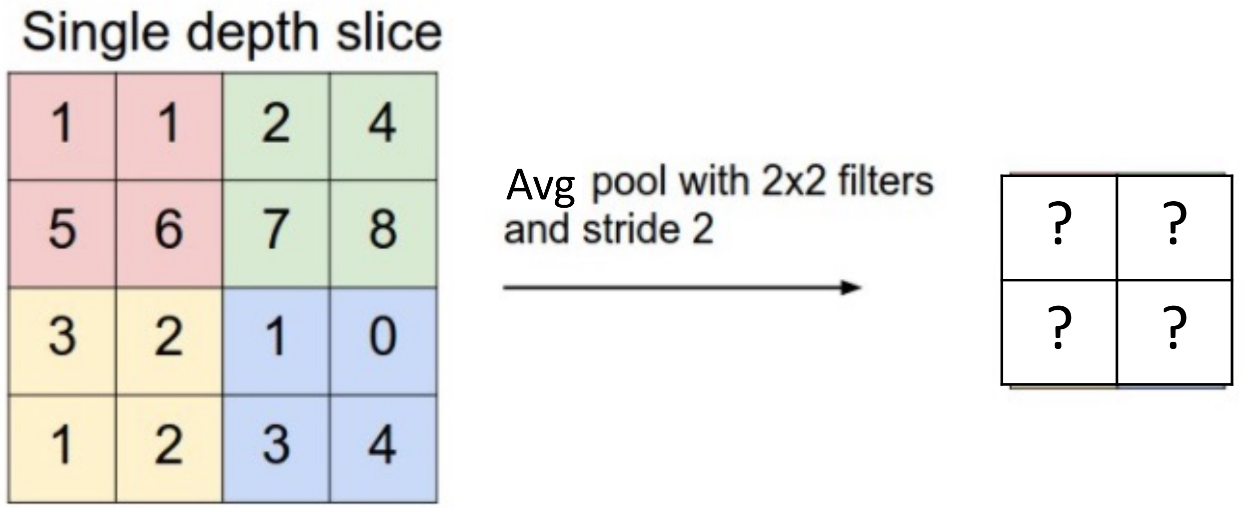
- **Max-pooling**: partitions input into a set of non-overlapping rectangles and outputs the maximum value for each chunk



http://cs231n.github.io/convolutional-networks/#pool

Pooling Layer: Summarizes Neighborhood

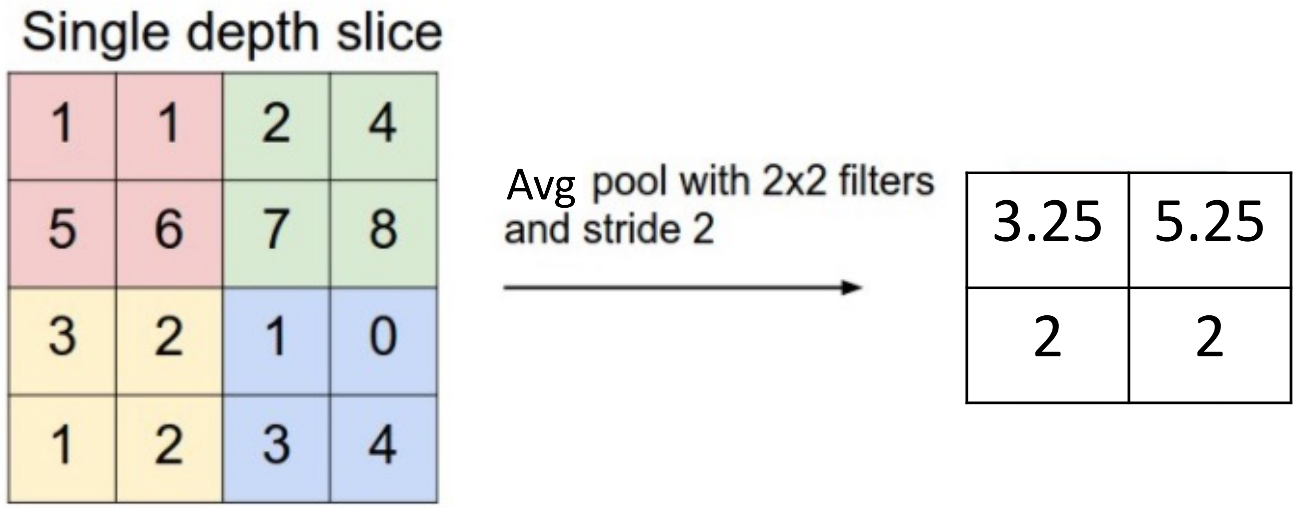
- **Max-pooling***:* partitions input into a set of non-overlapping rectangles and outputs the maximum value for each chunk
- **Average-pooling***:* partitions input into a set of non-overlapping rectangles and outputs the average value for each chunk



http://cs231n.github.io/convolutional-networks/#pool

Pooling Layer: Summarizes Neighborhood

- **Max-pooling***:* partitions input into a set of non-overlapping rectangles and outputs the maximum value for each chunk
- **Average-pooling***:* partitions input into a set of non-overlapping rectangles and outputs the average value for each chunk



http://cs231n.github.io/convolutional-networks/#pool

Pooling Layer: Benefits

- How many parameters must be learned?
  - None
- Benefits?
  - Reduces memory requirements
  - Reduces computational requirements

Core Components of Modern CNNs

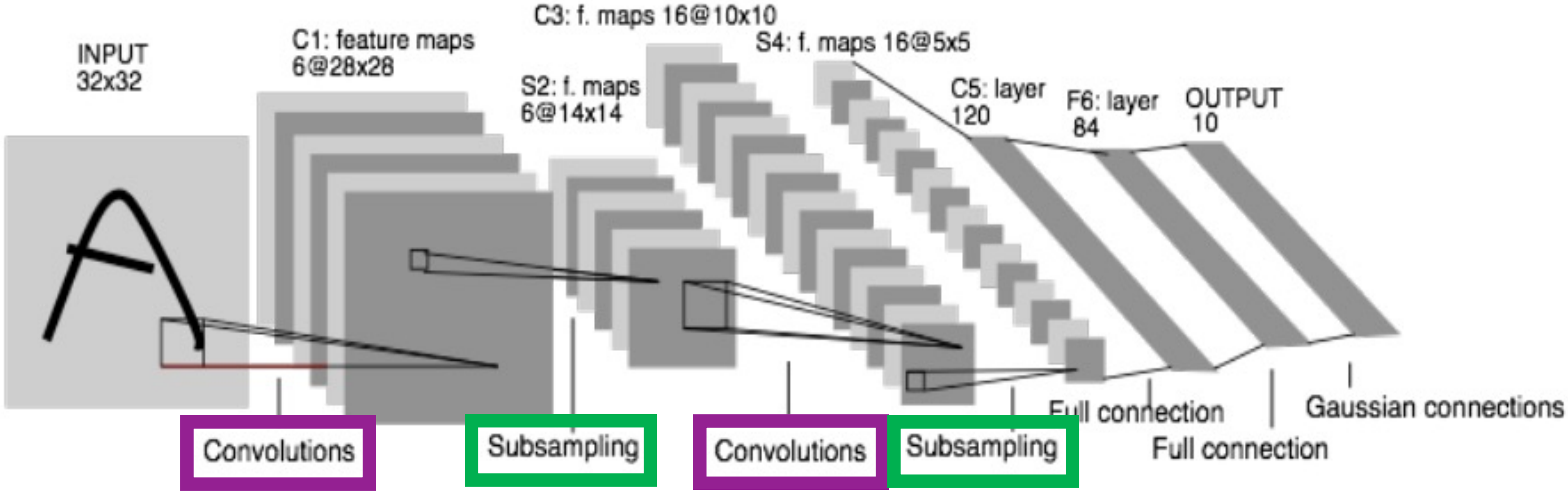


Figure Credit: https://people.eecs.berkeley.edu/~jrs/189/lec/cnn.pdf

Y. Lecun ; L. Bottou ; Y. Bengio ; P. Haffner; Gradient-based learning applied to document recognition; 1998

Object Recognition: Today's Topics

- Problem
- Applications
- Datasets
- Evaluation metric
- Typical Solution: Convolutional Neural Network


The End