Convolutional Neural Networks

Spring 2018

Review

- Last week:
  - Neural Network: Gradient Calculation with Backpropagation
  - Neural Network: Weight Update Methods
  - Neural Network: Training

- Assignments (Canvas):
  - Lab assignment due Friday
  - Project pre-proposal due yesterday
  - Project proposal due next week

- Questions?

Lab 4: VizWiz Challenge Winners

- Class: # Correct Predictions from 30 Test Examples
  - 15
  - 17 (x2)
  - 18 (x4)
  - 19 (x2)
  - 20 (x3)
  - 21 (x2)
  - 26 (x1)
- First place
  - Linli Ding (26/30)
- Second Place
  - Jennie Wolfgang (21/20)
  - Xiaomeng Mu (21/20)

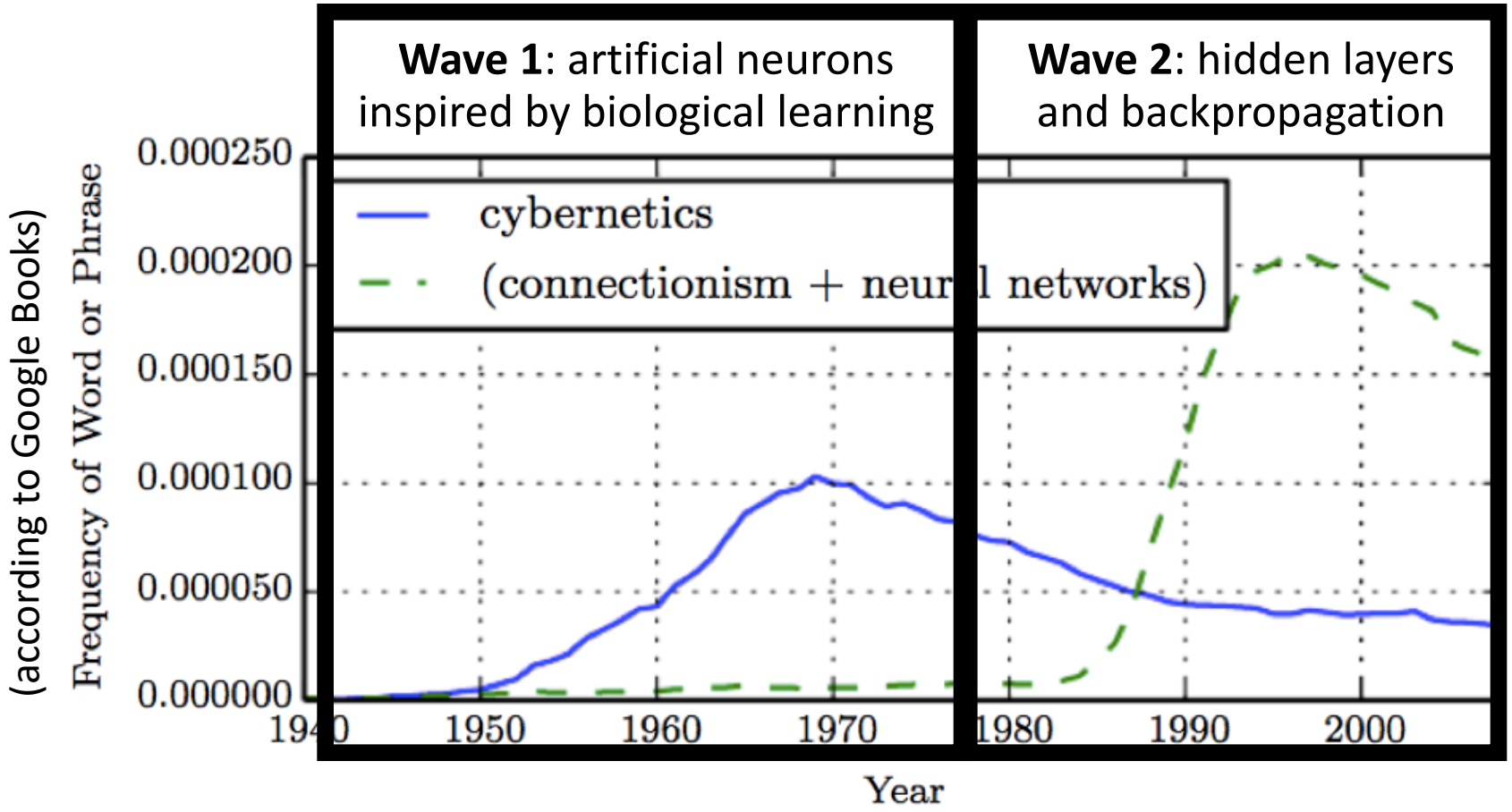
Today's Topics

- Revisiting History of Modern Neural Networks and "Deep Learning"
- Convolutional Neural Networks (CNN)
- Deep Features
- Lab

Today's Topics

- Revisiting History of Modern Neural Networks and "Deep Learning"
- Convolutional Neural Networks (CNN)
- Deep Features
- Lab

The History of “Deep Learning”

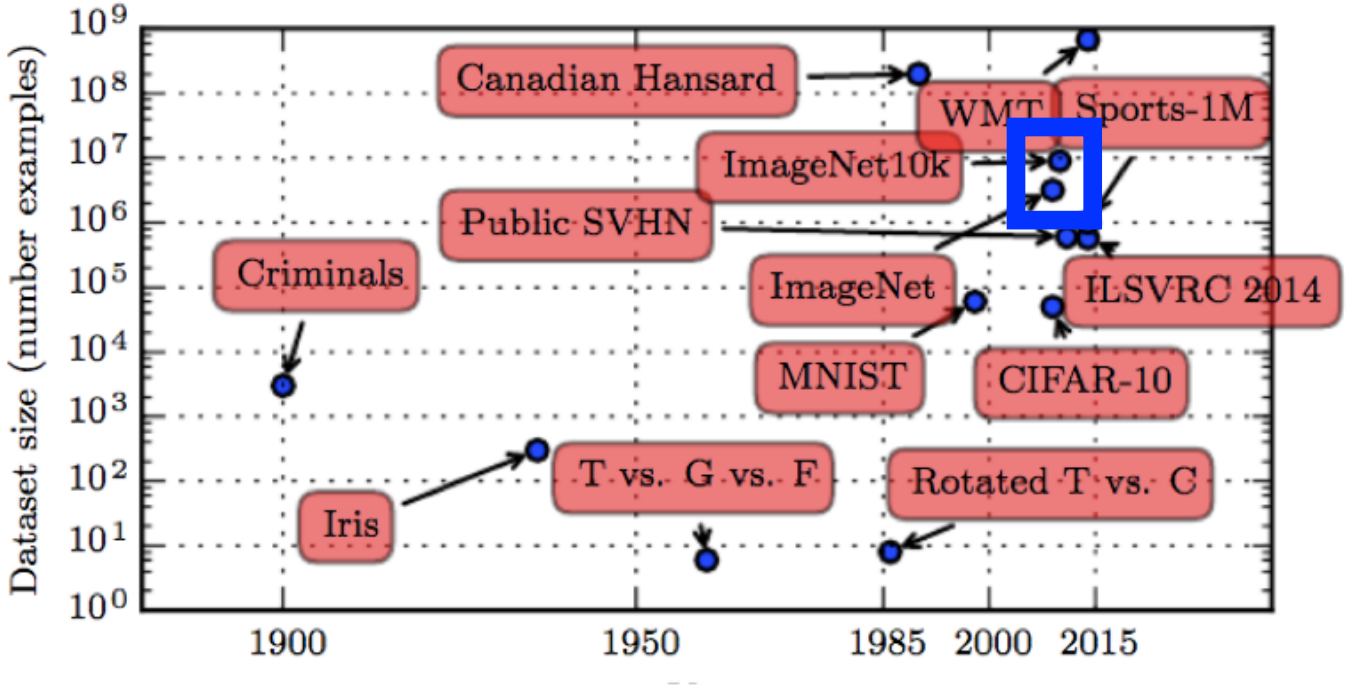


Wave 3: field emerged again as “deep learning” in 2006

Deep Learning, Ian Goodfellow, Yoshua Bengio, and Aaron Courville

Deep Learning: What Prompted the 3rd Wave?

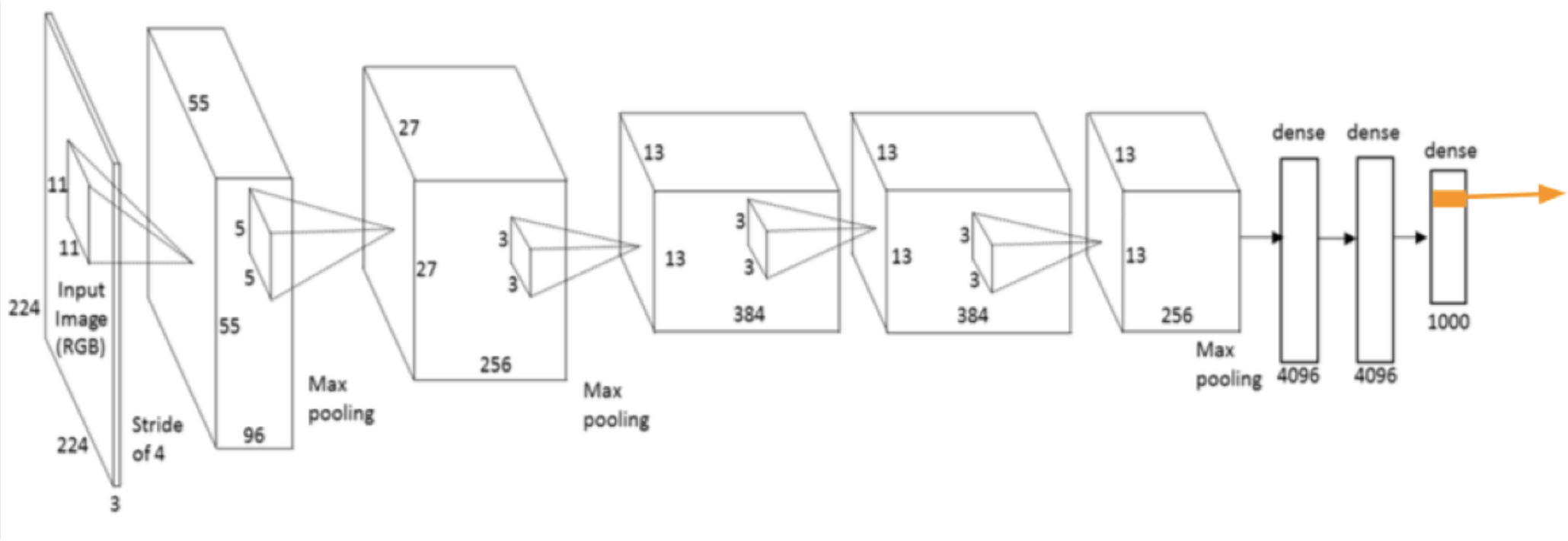
- The ability to provide the algorithms with the resources they need:
  1. "Big data"; e.g.,
  2. Faster Hardware; e.g., GPUs
  3. More memory; e.g., RAM



Deep Learning, Ian Goodfellow, Yoshua Bengio, and Aaron Courville

Why is Deep Learning So Popular Today?

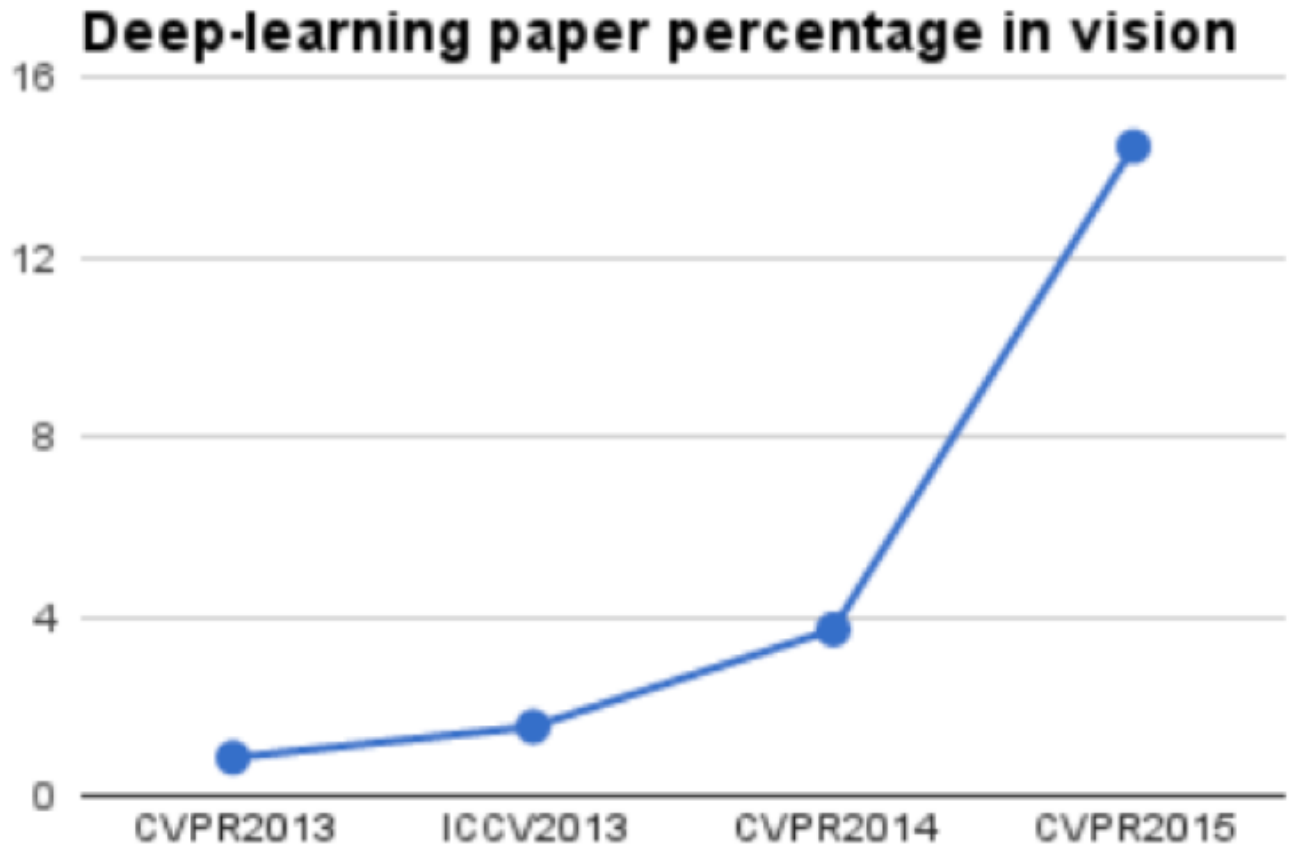
- 2012: a **Convolutional Neural Network** won **ImageNet challenge**
  - state-of-the-art top-5 error rate fell from **26.1% to 15.3%** with “AlexNet”



Slide Credit: https://www.slideshare.net/xavigiro/saliency-prediction-using-deep-learning-techniques

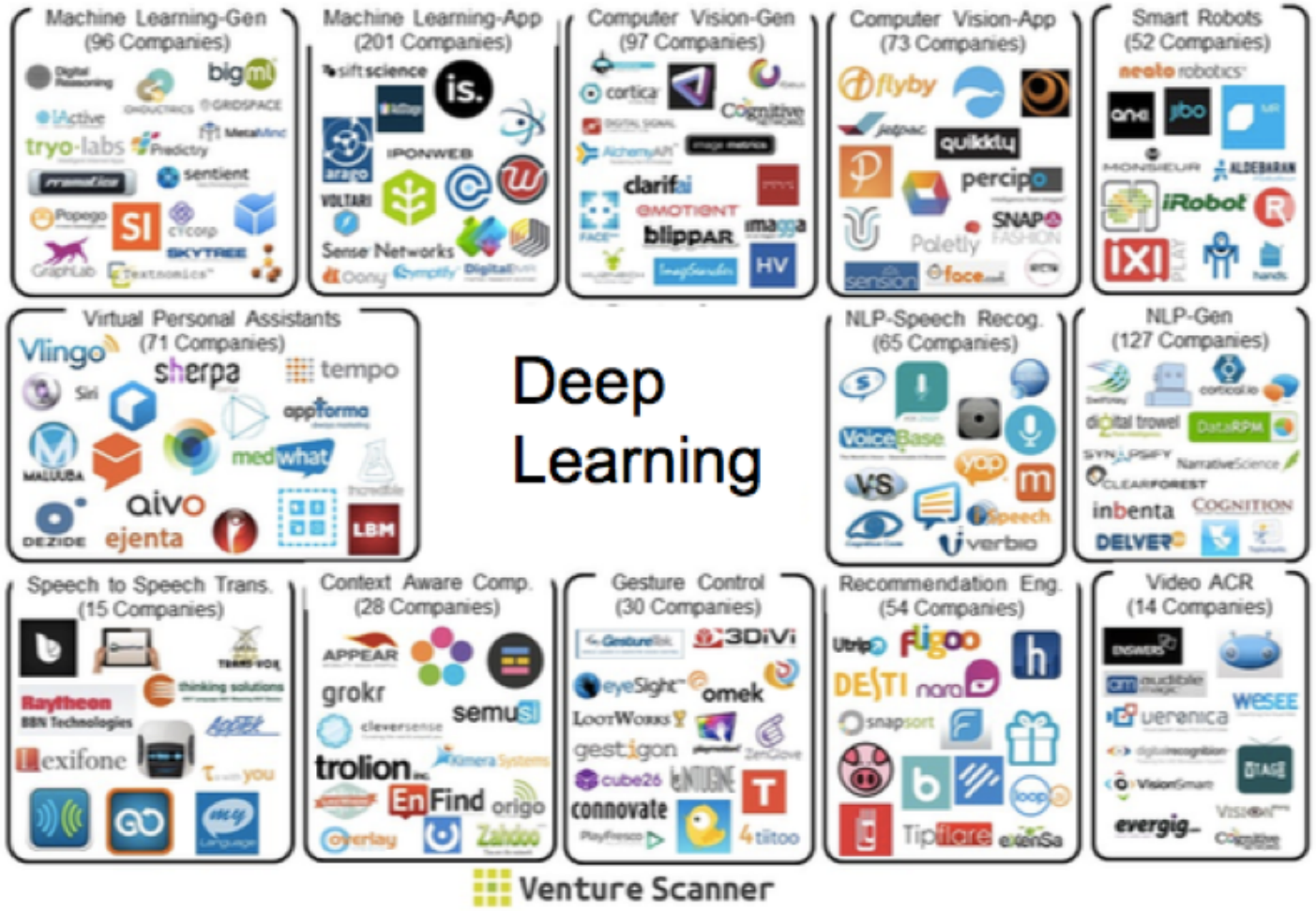
A. Krizhevsky, I. Sutskever, G. E. Hinton “ImageNet classification with deep convolutional neural networks”

Why is Deep Learning So Popular Today?



Slide Credit: https://www.slideshare.net/xavigiro/saliency-prediction-using-deep-learning-techniques

Deep Learning Inspiring New Companies



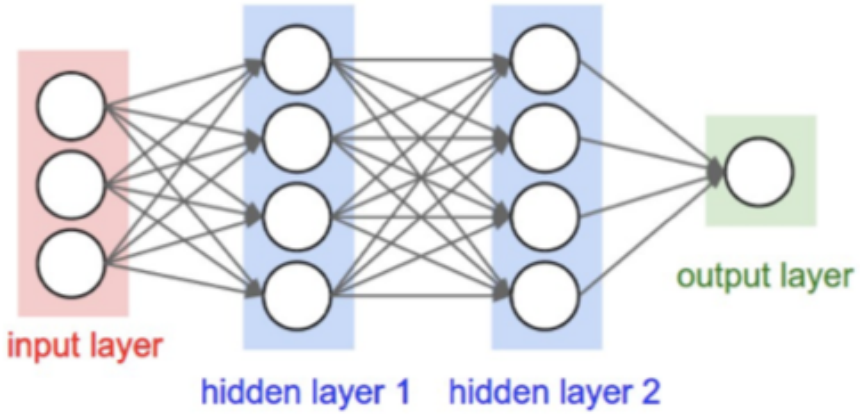
Slide Credit: https://www.slideshare.net/xavigiro/saliency-prediction-using-deep-learning-techniques

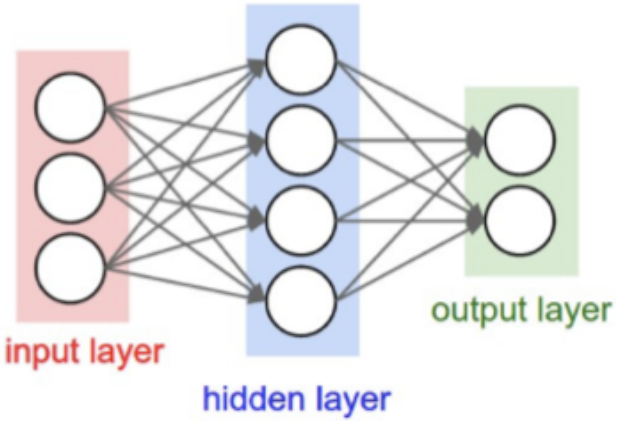
Today's Topics

- Revisiting History of Modern Neural Networks and "Deep Learning"
- Convolutional Neural Networks (CNN)
- Deep Features
- Lab

Why Convolutional Neural Networks (CNNs)?

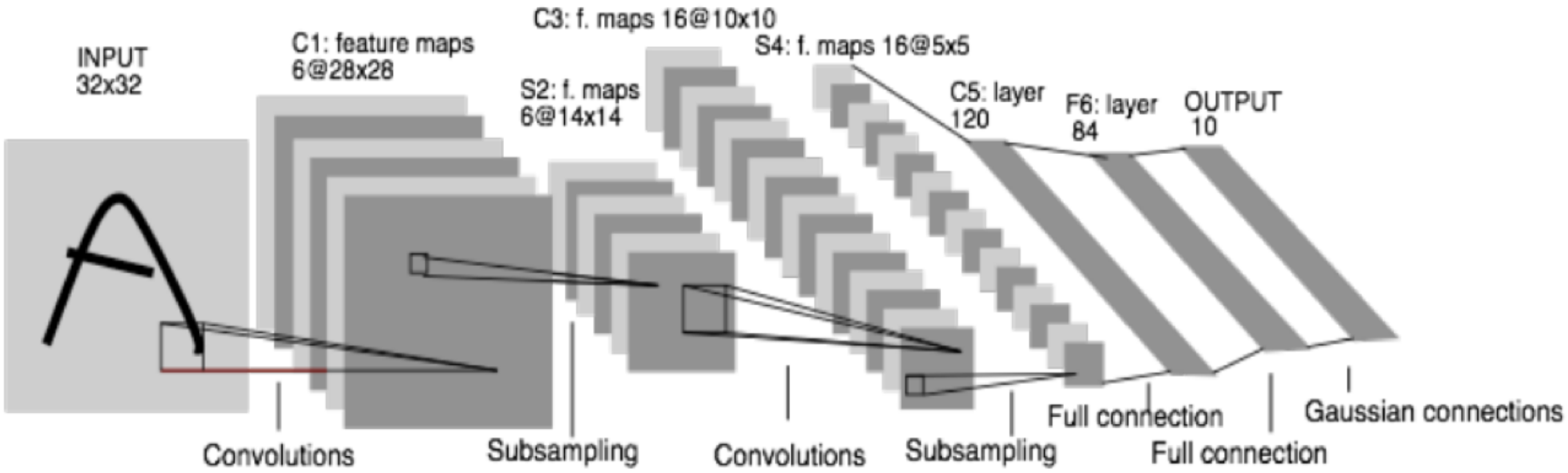
- Goal: reduce number of parameters in fully connected networks; i.e., when each node provides input to each node in the next layer





- Assume 2 layer model with 100 nodes per layer
  - e.g. how many weights are in a 640x480 image?
    - 640x480x3x100 + 100x100 + 100x1 = 92,170,100
  - e.g. how many weights are in a 2048X1536 image (3.1 Megapixel image)?
    - 2048x1536x3x100 + 100x100 + 100x1 = 943,728,500

First CNN: LeNet

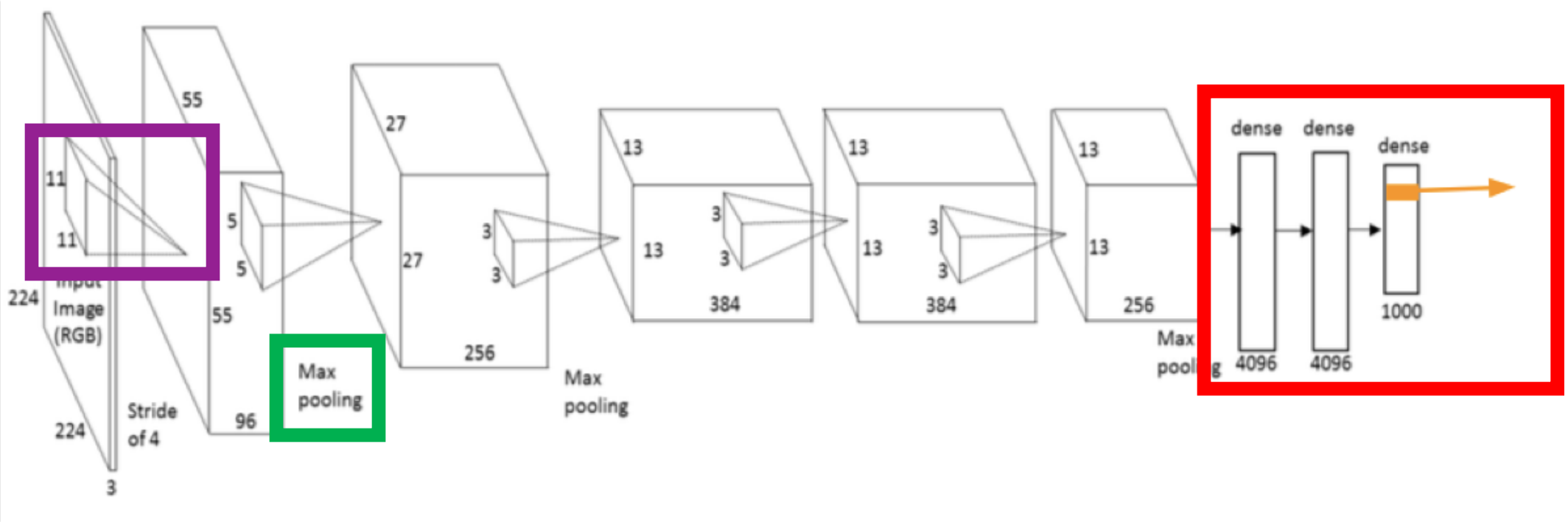


Slide Credit: https://people.eecs.berkeley.edu/~jrs/189/lec/cnn.pdf

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

CNN Architecture: Key Components

- CNN extracts useful features of lower dimension prior to passing it to MLP with:
  - Convolutional layers
  - Pooling Layers

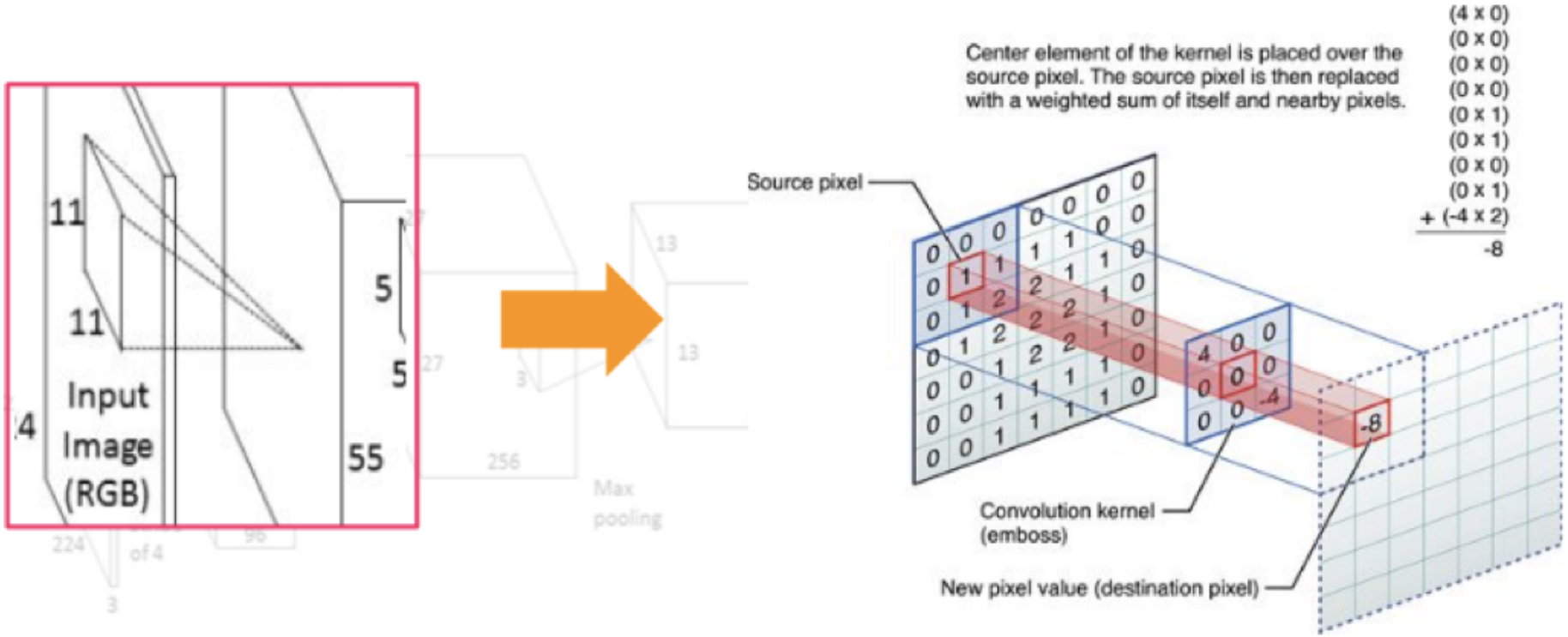


Slide Credit: https://www.slideshare.net/xavigiro/saliency-prediction-using-deep-learning-techniques

A. Krizhevsky, I. Sutskever, G. E. Hinton “ImageNet classification with deep convolutional neural networks”

Convolutional Layer

- Set of kernels: each kernel computes the convolution of the output of a previous layer
- Kernels are learned from a given training dataset via *gradient descent*



Slide Credit: https://www.slideshare.net/xavigiro/saliency-prediction-using-deep-learning-techniques

Convolution Operation: 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

Kernel

1	0	1
0	1	0
1	0	1

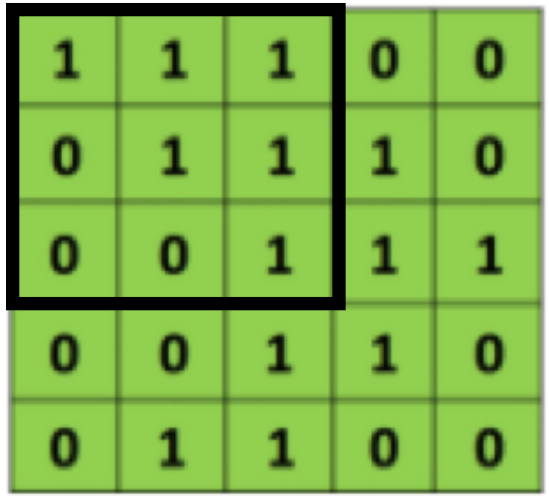
Feature Map

?	?	?
?	?	?
?	?	?

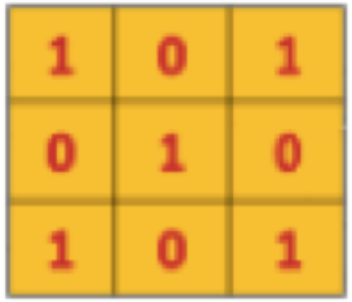
https://ujjwalkarn.me/2016/08/11/intuitive-explanation-convnets/

Convolution Operation: Toy Example

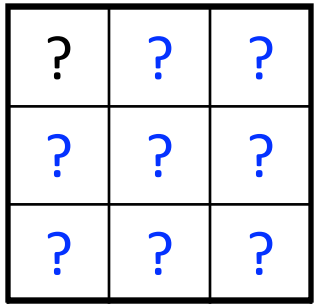
Image



Kernel



Feature Map

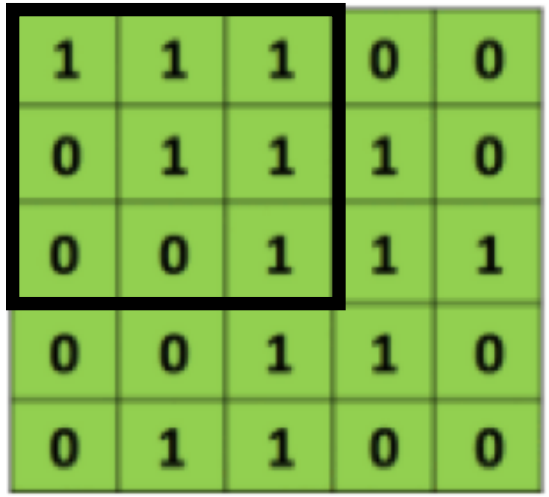


Weighted Sum = 1x1 + 1x0 + 1x1 + 0x0+1x1+1x0 + 0x1 + 0x0 + 1x1 = 4

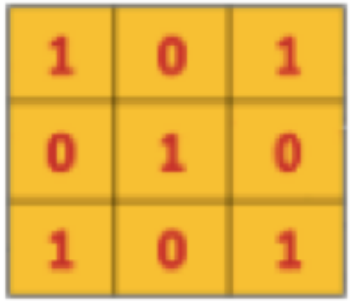
https://ujjwalkarn.me/2016/08/11/intuitive-explanation-convnets/

Convolution Operation: Toy Example

Image



Kernel



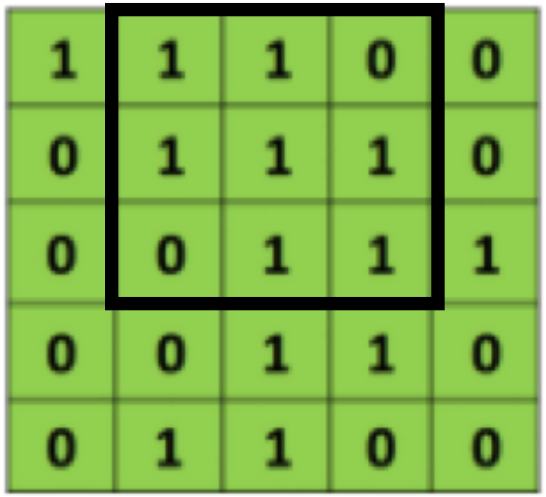
Feature Map

4	?	?
?	?	?
?	?	?

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

Convolution Operation: Toy Example

Image



Kernel

1	0	1
0	1	0
1	0	1

Feature Map

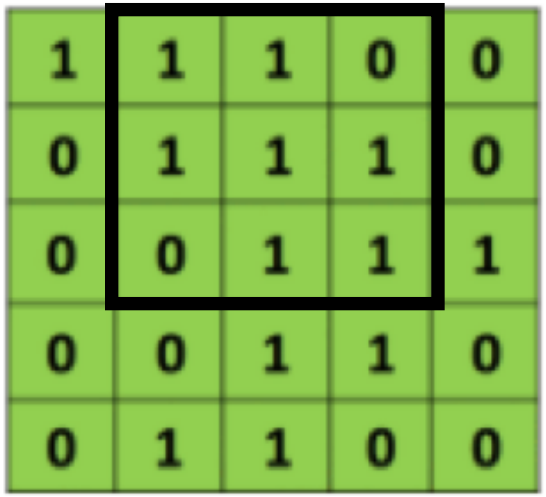
4	?	?
?	?	?
?	?	?

Weighted Sum = ?

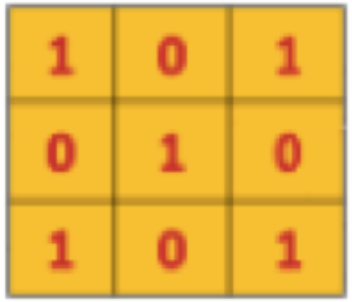
https://ujjwalkarn.me/2016/08/11/intuitive-explanation-convnets/

Convolution Operation: Toy Example

Image



Kernel



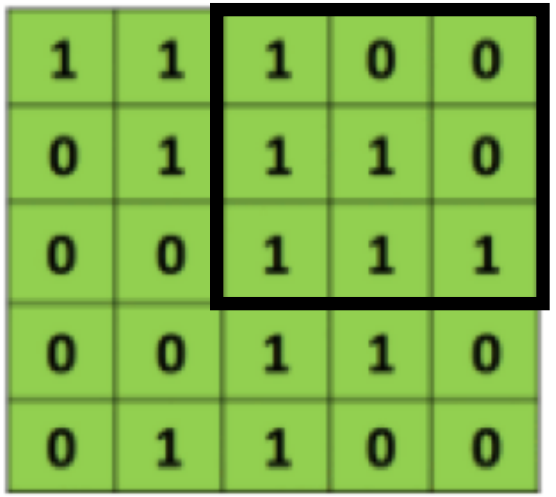
Feature Map

4	3	?
?	?	?
?	?	?

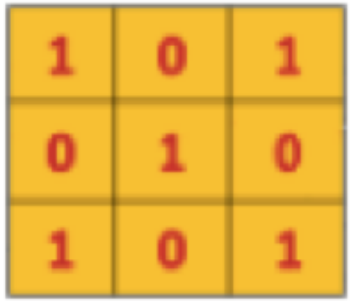
https://ujjwalkarn.me/2016/08/11/intuitive-explanation-convnets/

Convolution Operation: Toy Example

Image



Kernel



Feature Map

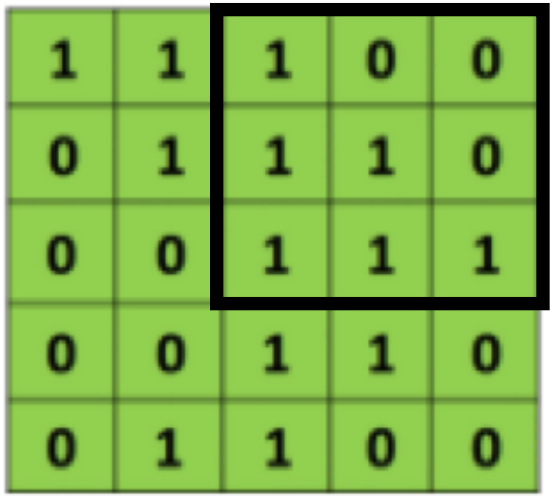
4	3	?
?	?	?
?	?	?

Weighted Sum = ?

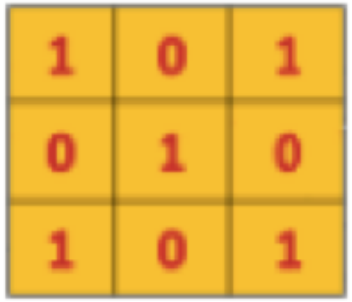
https://ujjwalkarn.me/2016/08/11/intuitive-explanation-convnets/

Convolution Operation: Toy Example

Image



Kernel



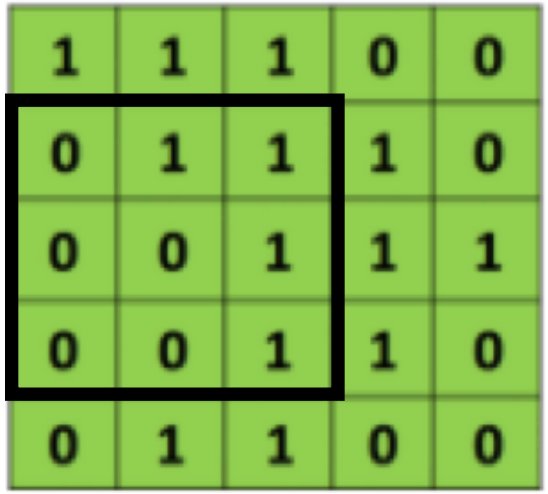
Feature Map

4	3	4
?	?	?
?	?	?

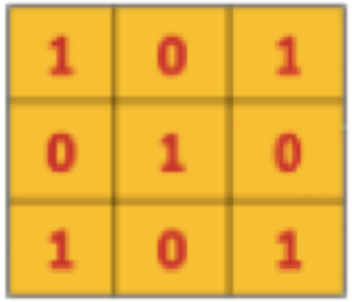
https://ujjwalkarn.me/2016/08/11/intuitive-explanation-convnets/

Convolution Operation: Toy Example

Image



Kernel



Feature Map

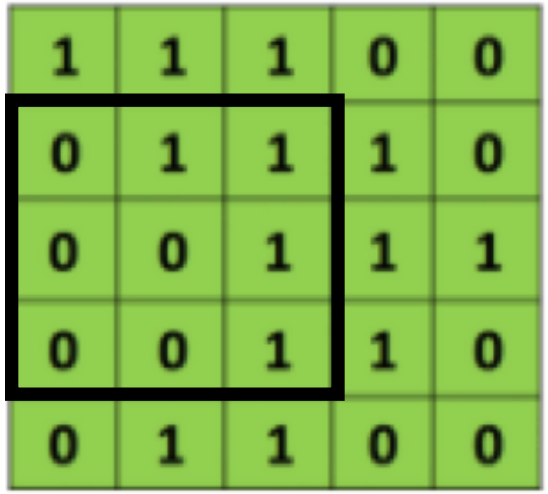
4	3	4
?	?	?
?	?	?

Weighted Sum = ?

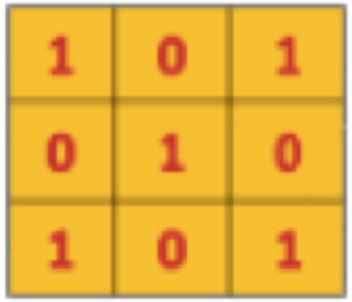
https://ujjwalkarn.me/2016/08/11/intuitive-explanation-convnets/

Convolution Operation: Toy Example

Image



Kernel



Feature Map

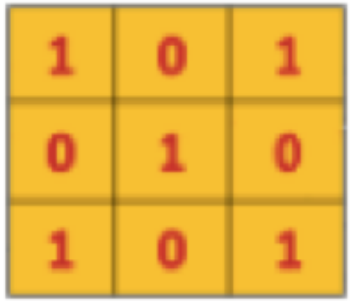
4	3	4
2	?	?
?	?	?

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

Convolution Operation: Toy Example

Image

Kernel



Feature Map

4	3	4
2	?	?
?	?	?

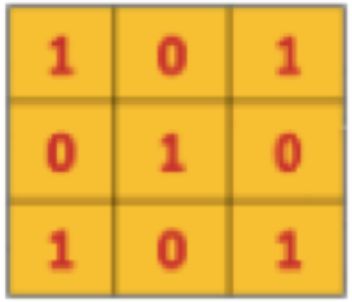
Weighted Sum = ?

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

Convolution Operation: Toy Example

Image

Kernel



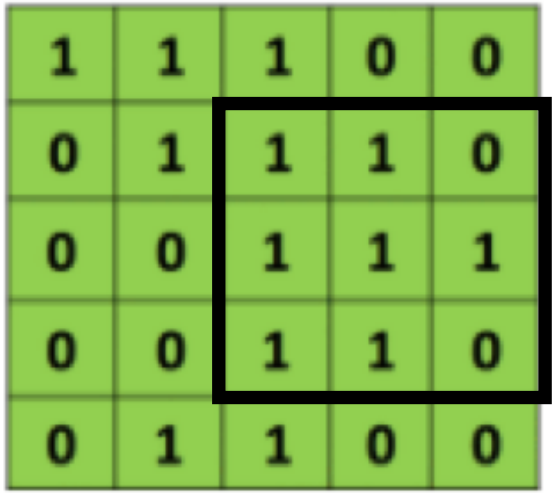
Feature Map

4	3	4
2	4	?
?	?	?

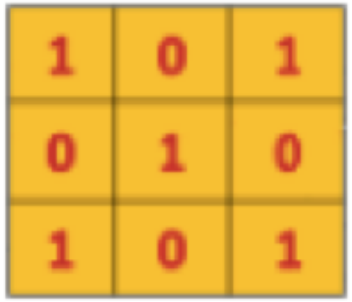
https://ujjwalkarn.me/2016/08/11/intuitive-explanation-convnets/

Convolution Operation: Toy Example

Image



Kernel



Feature Map

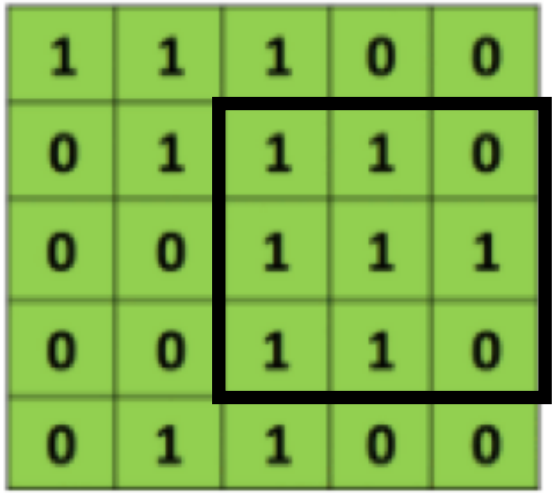
4	3	4
2	4	?
?	?	?

Weighted Sum = ?

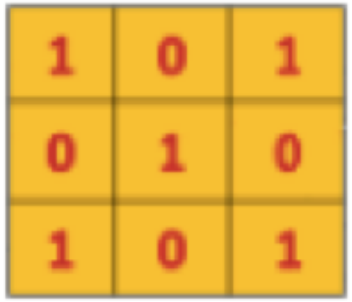
https://ujjwalkarn.me/2016/08/11/intuitive-explanation-convnets/

Convolution Operation: Toy Example

Image



Kernel



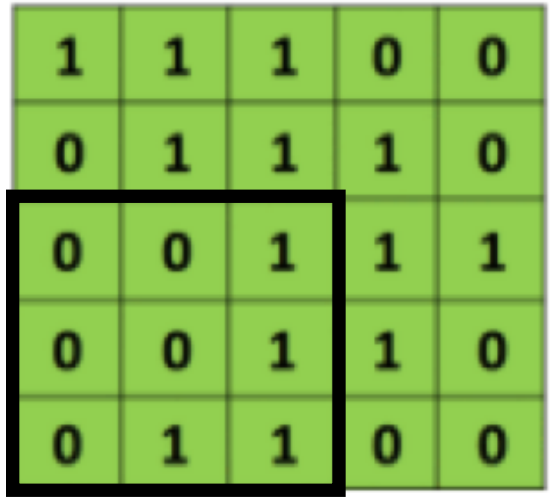
Feature Map

4	3	4
2	4	3
?	?	?

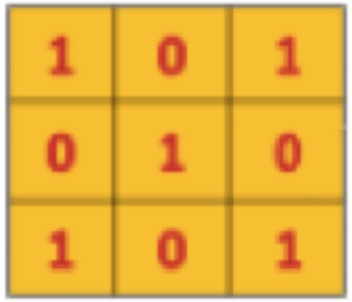
https://ujjwalkarn.me/2016/08/11/intuitive-explanation-convnets/

Convolution Operation: Toy Example

Image



Kernel



Feature Map

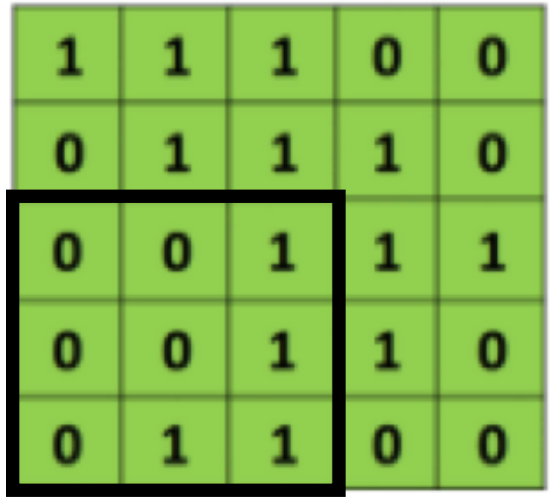
4	3	4
2	4	3
?	?	?

Weighted Sum = ?

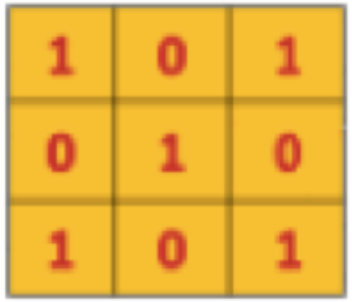
https://ujjwalkarn.me/2016/08/11/intuitive-explanation-convnets/

Convolution Operation: Toy Example

Image



Kernel



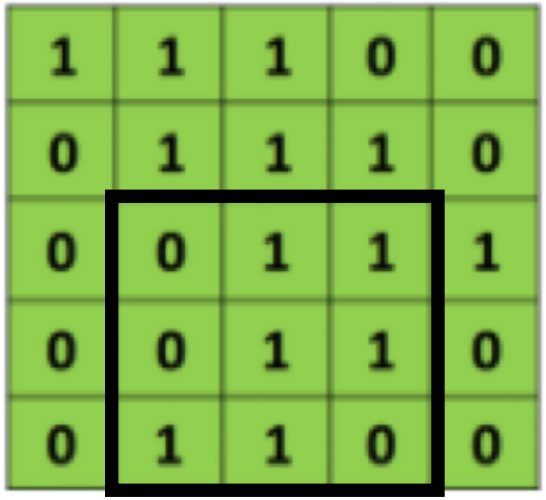
Feature Map

4	3	4
2	4	3
2	?	?

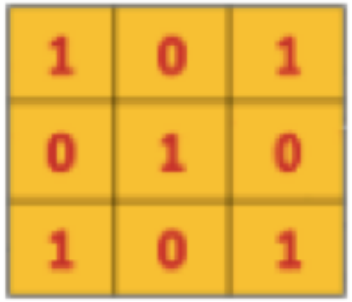
https://ujjwalkarn.me/2016/08/11/intuitive-explanation-convnets/

Convolution Operation: Toy Example

Image



Kernel



Feature Map

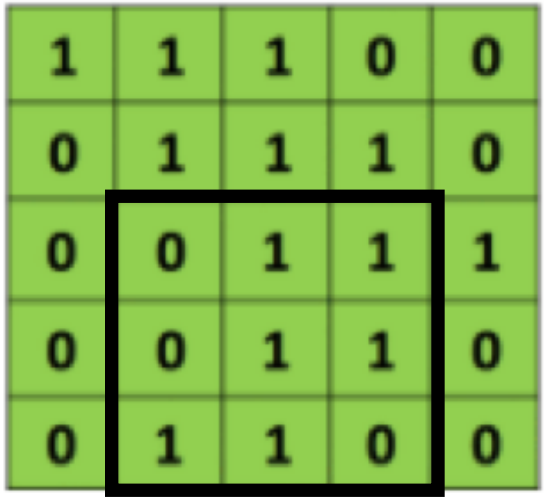
4	3	4
2	4	3
2	?	?

Weighted Sum = ?

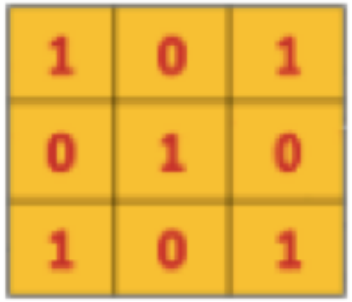
https://ujjwalkarn.me/2016/08/11/intuitive-explanation-convnets/

Convolution Operation: Toy Example

Image



Kernel



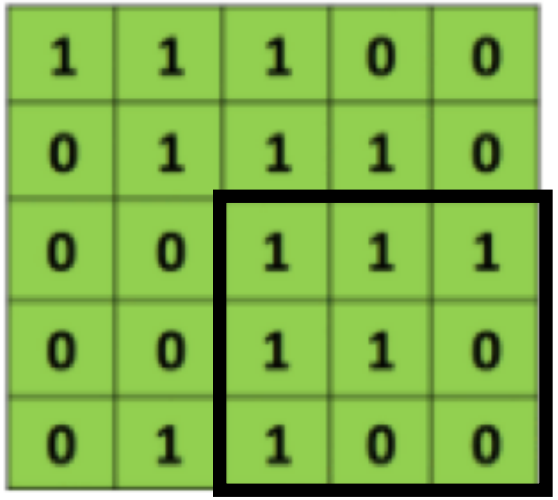
Feature Map

4	3	4
2	4	3
2	3	?

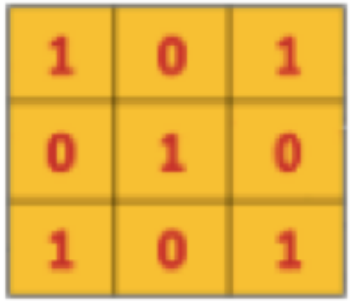
https://ujjwalkarn.me/2016/08/11/intuitive-explanation-convnets/

Convolution Operation: Toy Example

Image



Kernel



Feature Map

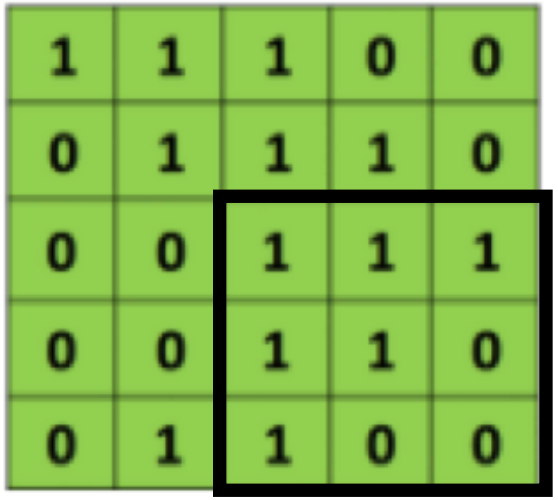
4	3	4
2	4	3
2	3	?

Weighted Sum = ?

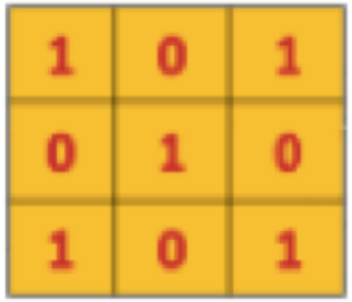
https://ujjwalkarn.me/2016/08/11/intuitive-explanation-convnets/

Convolution Operation: Toy Example

Image



Kernel



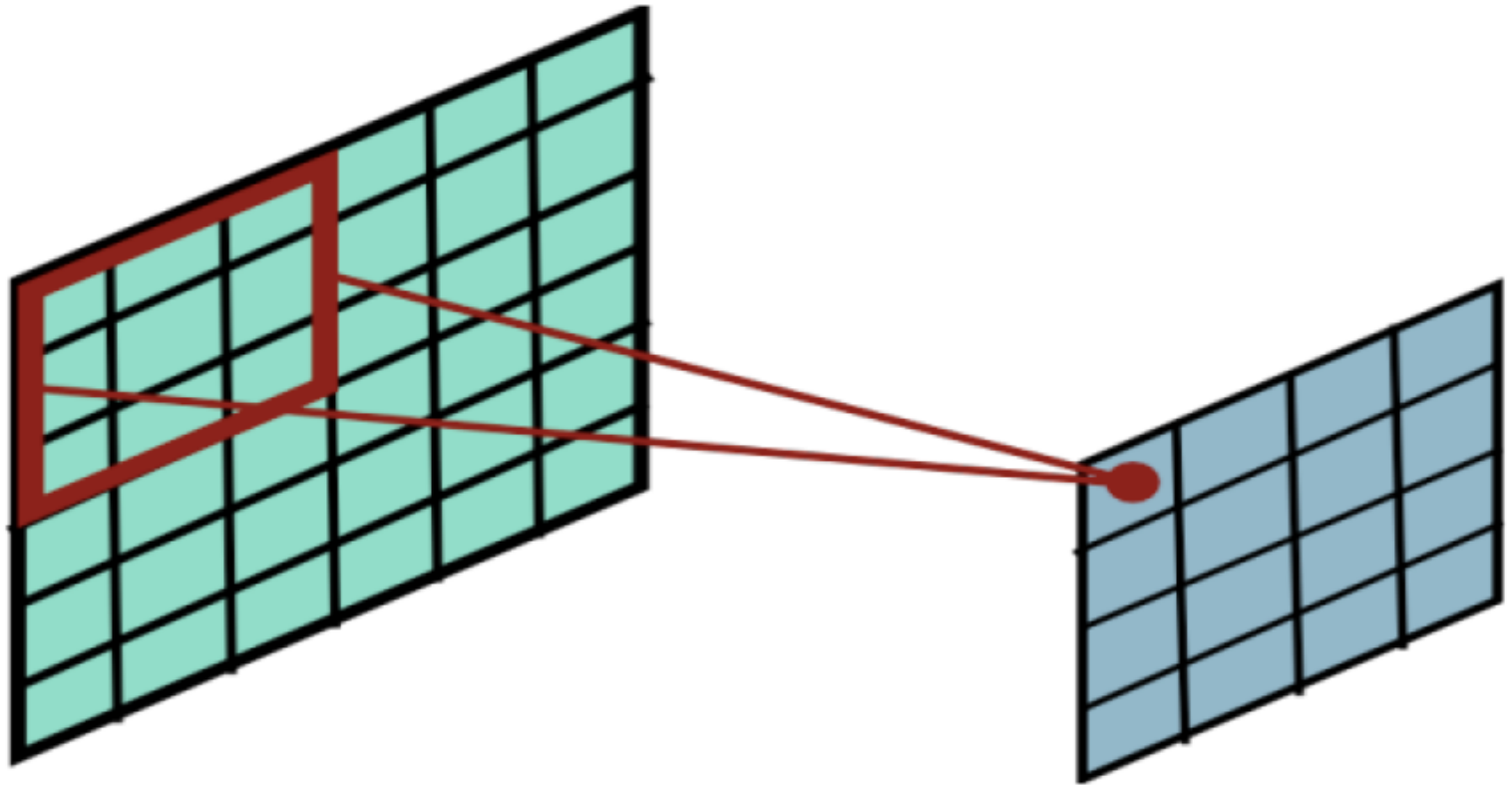
Feature Map

4	3	4
2	4	3
2	3	4

Weighted Sum = ?

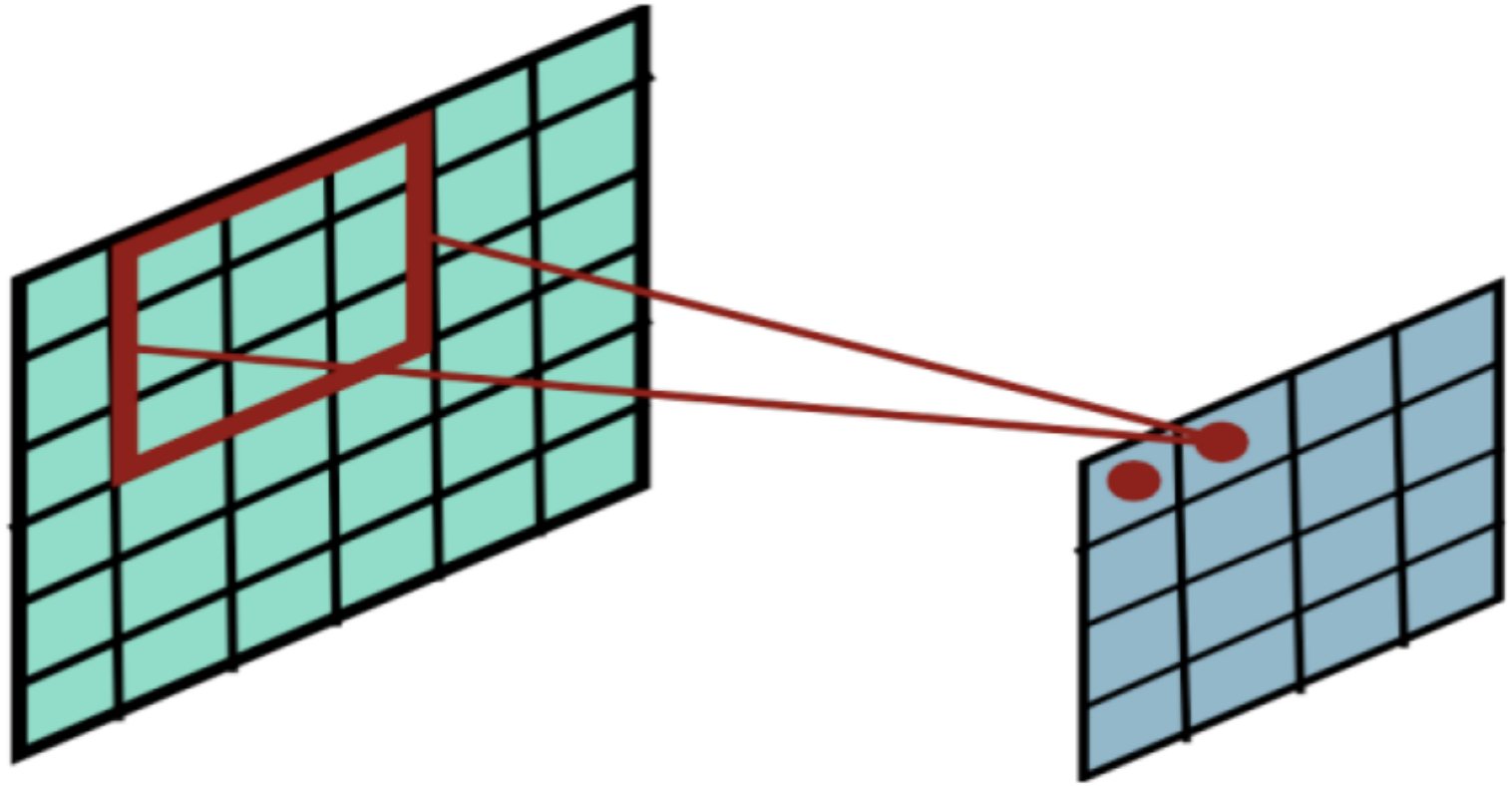
https://ujjwalkarn.me/2016/08/11/intuitive-explanation-convnets/

Convolutional Operation (3x3 kernel)



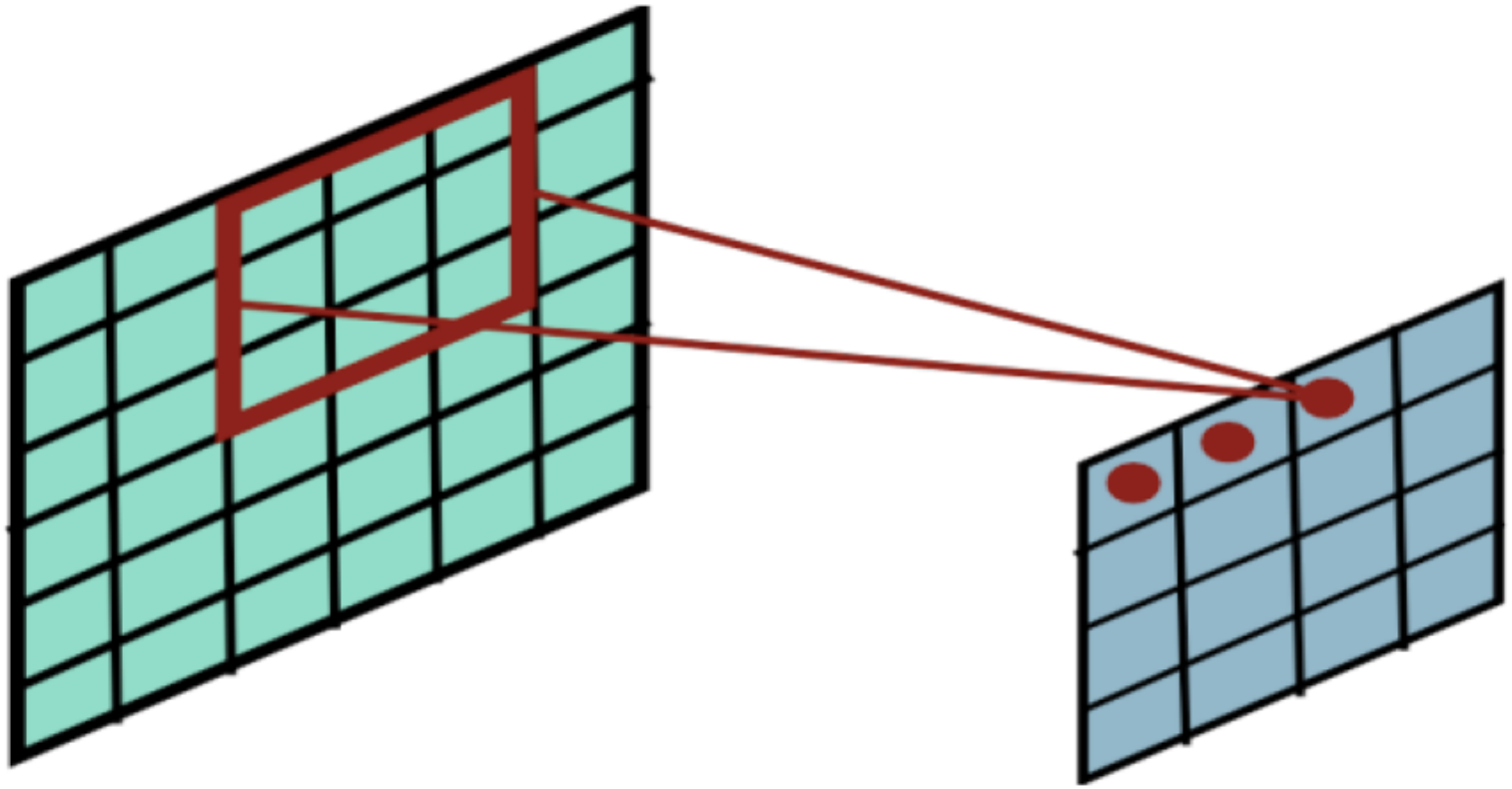
https://people.eecs.berkeley.edu/~jrs/189/lec/cnn.pdf

Convolutional Operation (3x3 kernel)



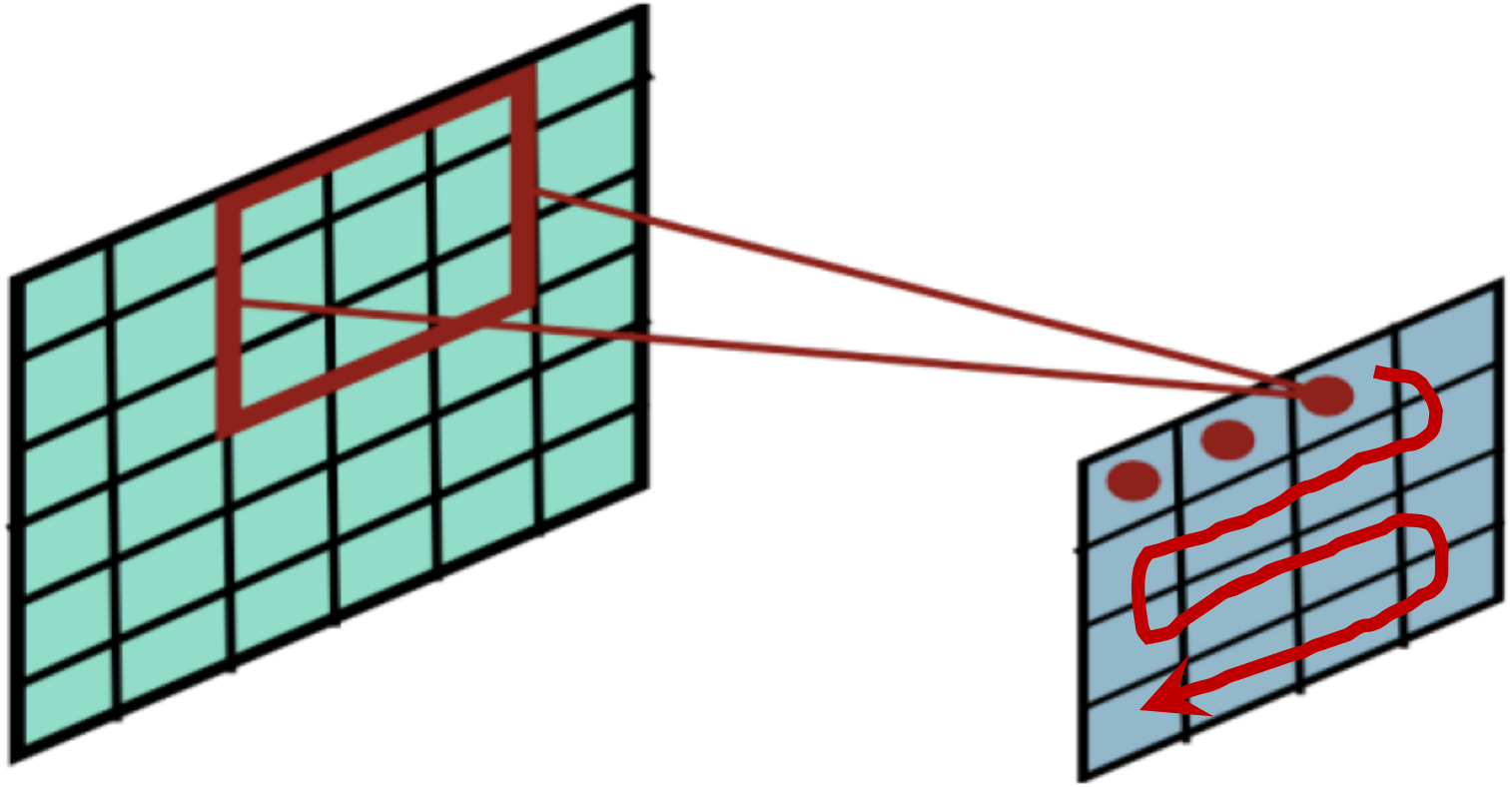
https://people.eecs.berkeley.edu/~jrs/189/lec/cnn.pdf

Convolutional Operation (3x3 kernel)



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

Convolutional Operation (3x3 kernel)



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

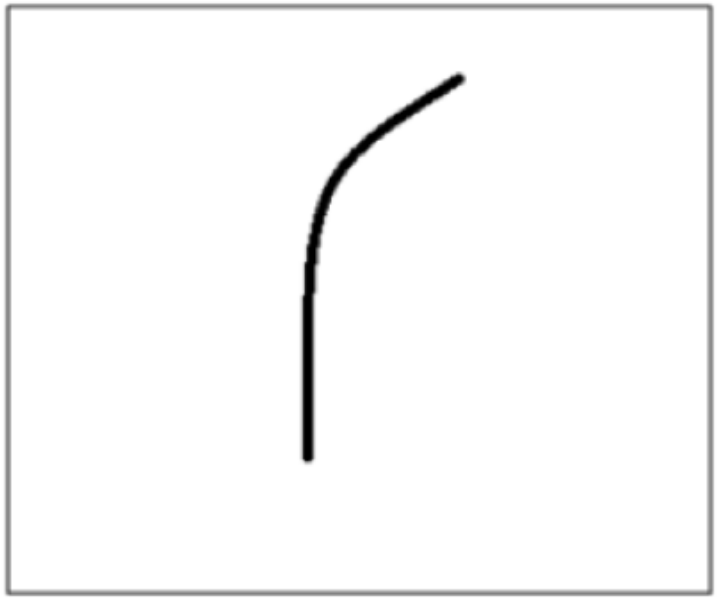
Convolution: What Does It Do?

- 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



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

Convolution: 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!!)**

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

Convolution: 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!!)**

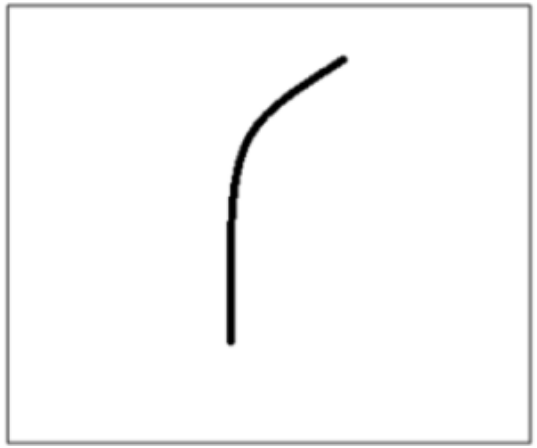
Slide Credit: https://adeshpande3.github.io/A-Beginner%27s-Guide-To-Understanding-Convolutional-Neural-Networks/

Convolution: 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



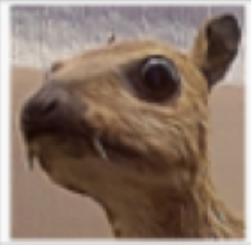
Filter Overlaid on Image (Big Response!)

Filter Overlaid on Image (Small Response!)

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

Different Filters Detect Different Features

	Filter	Feature Map
Identity	$\begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}$	
Edge detection	$\begin{bmatrix} 1 & 0 & -1 \\ 0 & 0 & 0 \\ -1 & 0 & 1 \end{bmatrix}$	
	$\begin{bmatrix} 0 & 1 & 0 \\ 1 & -4 & 1 \\ 0 & 1 & 0 \end{bmatrix}$	
	$\begin{bmatrix} -1 & -1 & -1 \\ -1 & 8 & -1 \\ -1 & -1 & -1 \end{bmatrix}$	

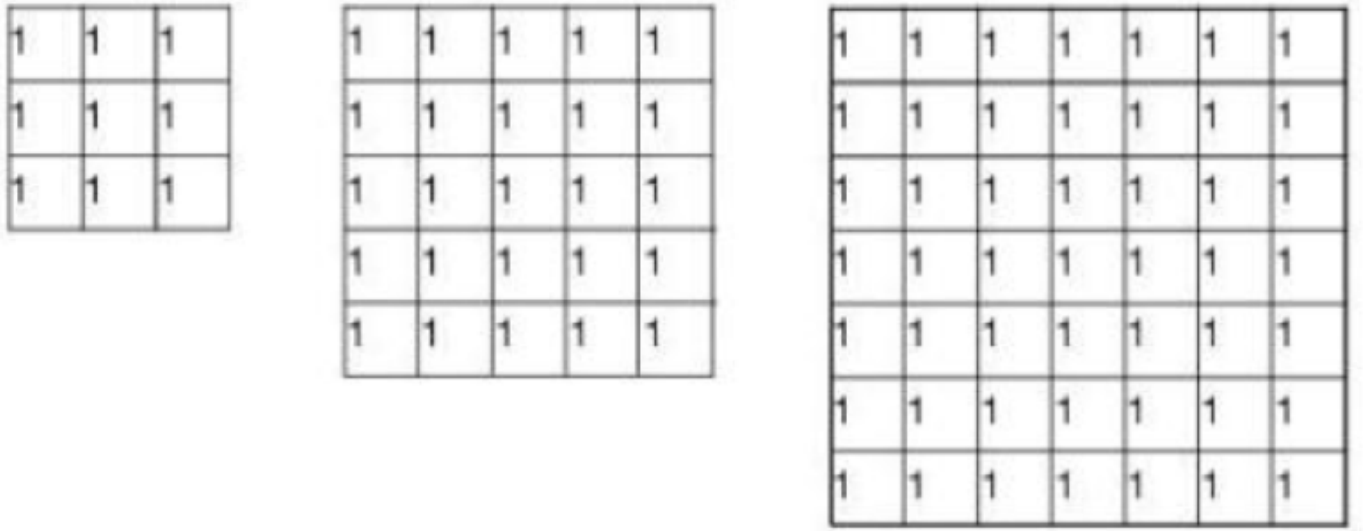
	Filter	Feature Map
Sharpen	$\begin{bmatrix} 0 & -1 & 0 \\ -1 & 5 & -1 \\ 0 & -1 & 0 \end{bmatrix}$	
Box blur (normalized)	$\frac{1}{9}\begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix}$	
Gaussian blur (approximation)	$\frac{1}{16}\begin{bmatrix} 1 & 2 & 1 \\ 2 & 4 & 2 \\ 1 & 2 & 1 \end{bmatrix}$	

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

http://www1.adept.com/main/KE/DATA/ACE/AdeptSight_User/ImageProcessing_Operations.html

Convolution Implementation Details

- **Kernel size**: e.g., 3x3, 5x5, 7x7 averaging filters



- Why choose a larger versus smaller kernel?

http://www1.adept.com/main/KE/DATA/ACE/AdeptSight_User/ImageProcessing_Operations.html

Convolution Implementation Details

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

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

Kernel

1	0	1
0	1	0
1	0	1

Feature Map

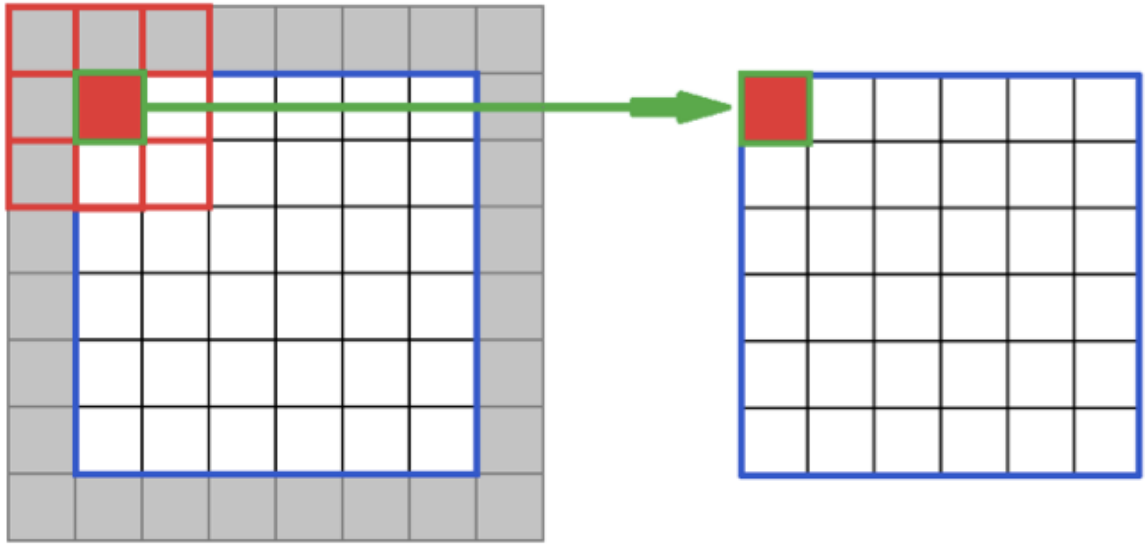
4	4
2	4

- Why choose a larger versus smaller stride size?

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

Convolution Implementation Details

- **Padding**: add zeros at the image boundaries
  - e.g., “same” padding versus “valid” (no) padding



- Why use padding? How much padding to use?

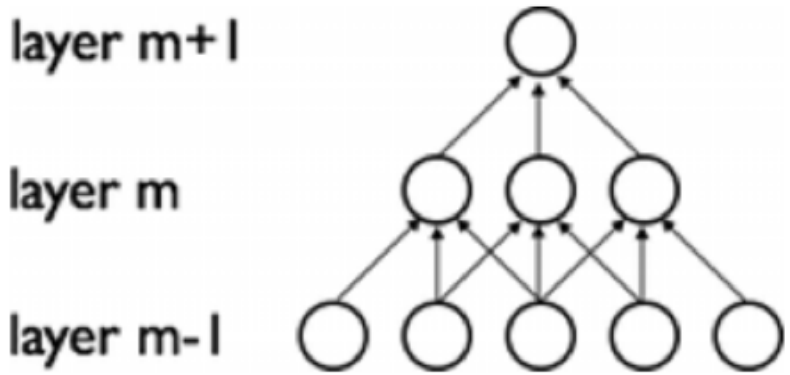
https://software.intel.com/en-us/node/586159

Demo

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

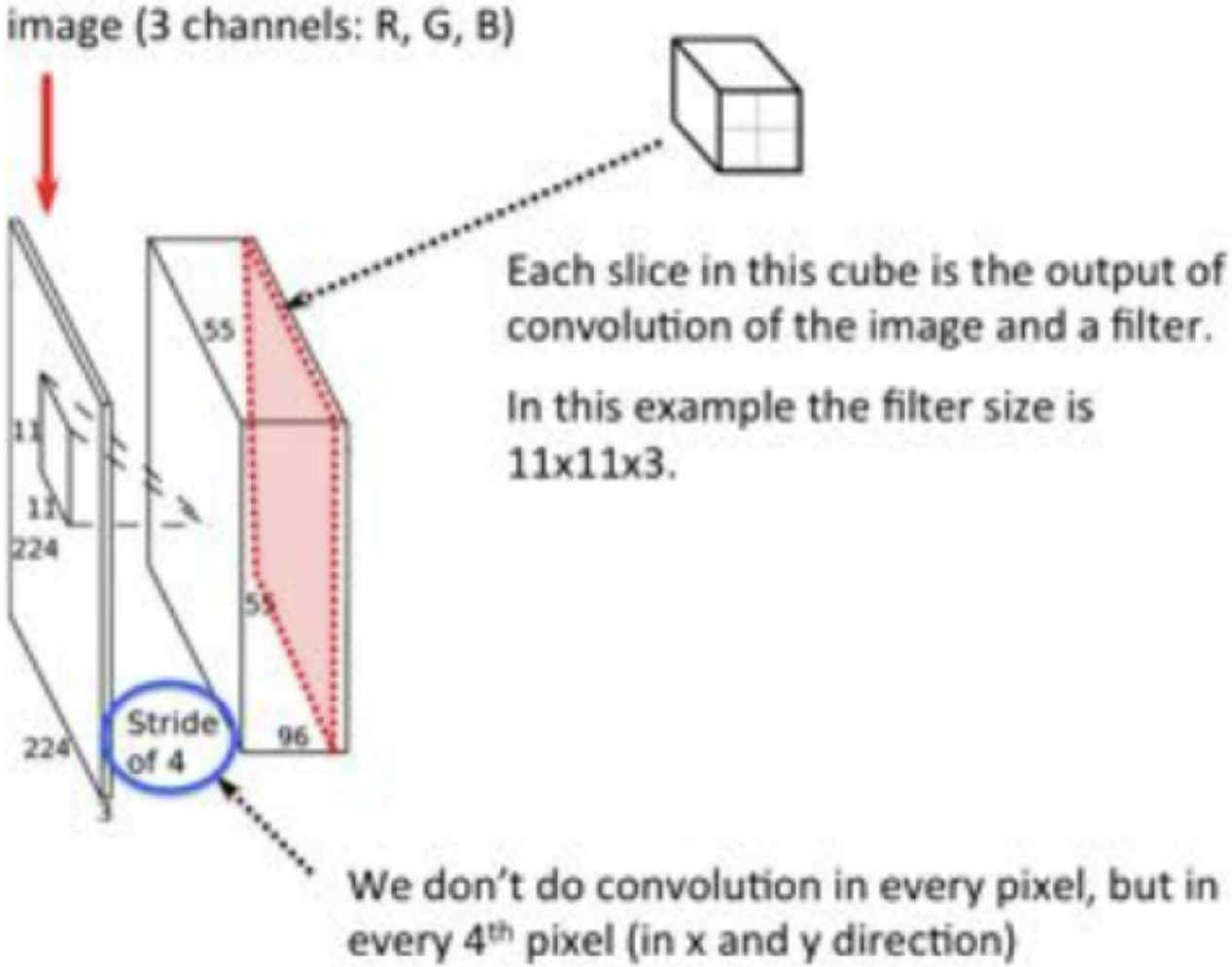
Convolutional Layer

- Implementation:
  1. Apply convolution operation with each filter
  2. Add biases (one per each output image).
  3. Apply an activation function to all the pixels of the output images.
- CNNs learn “filters” based on spatially **local input patterns.**
- Stacking many convolutional layers leads to learning patterns in increasingly **larger regions of the pixel space**



http://deeplearning.net/tutorial/lenet.html

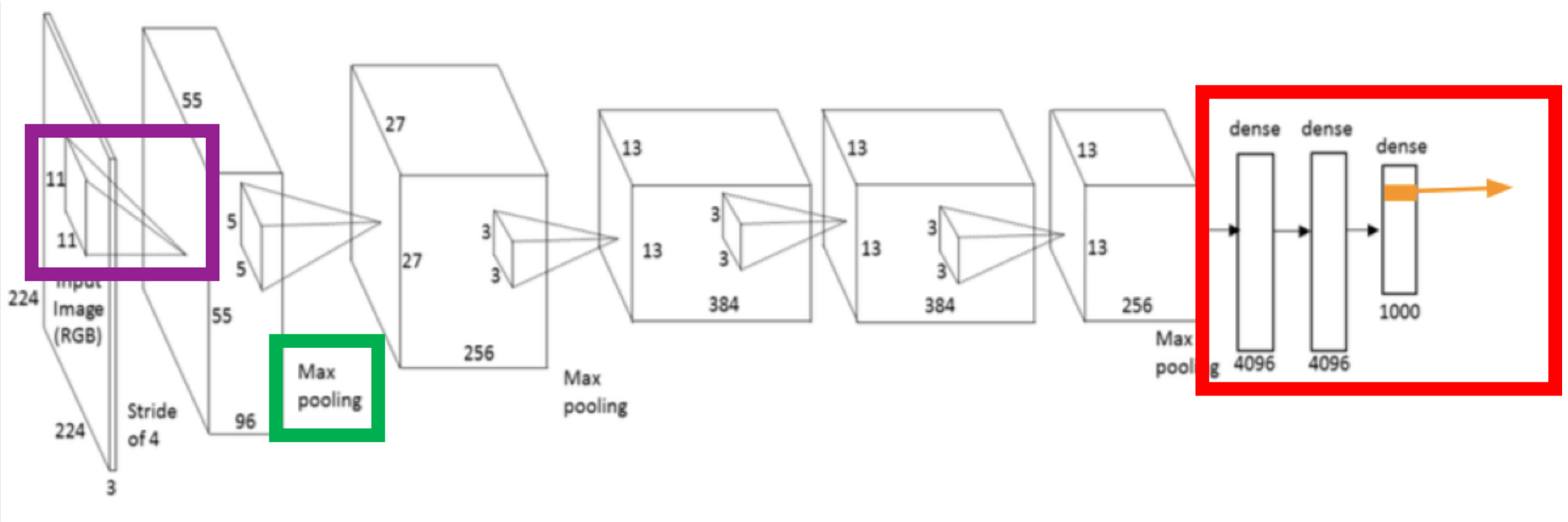
Recall CNN Architecture



Slide Credit: http://www.cs.utoronto.ca/~fidler/teaching/2015/slides/CSC411/11_nn2.pdf

Recall CNN Architecture: Key Components

- CNN extracts useful features of lower dimension prior to passing it to MLP with:
  - Convolutional layers
  - Pooling Layers

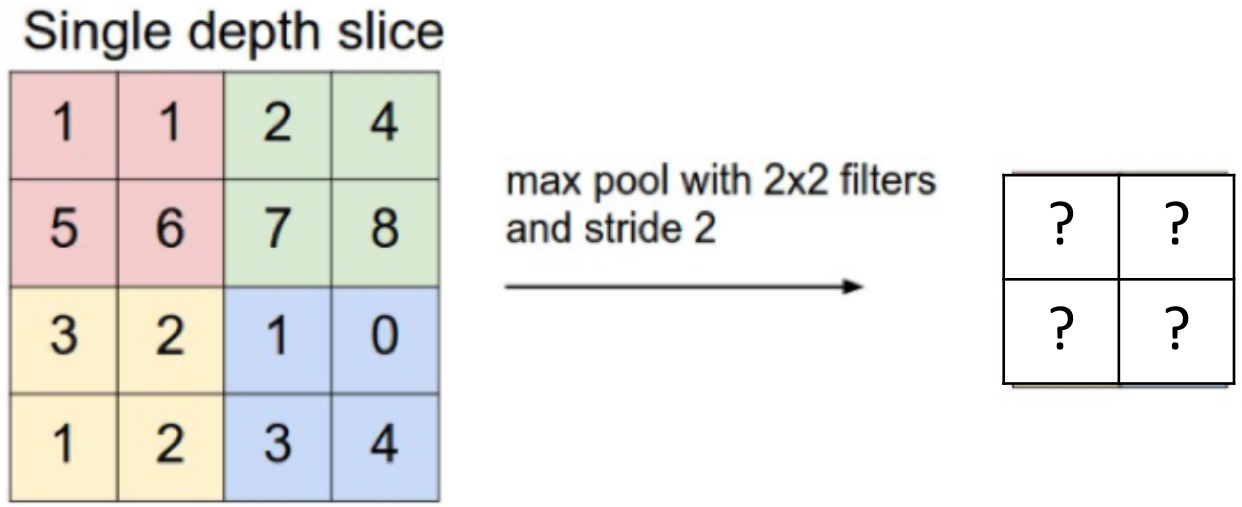


Slide Credit: https://www.slideshare.net/xavigiro/saliency-prediction-using-deep-learning-techniques

A. Krizhevsky, I. Sutskever, G. E. Hinton “ImageNet classification with deep convolutional neural networks”

Pooling Layer

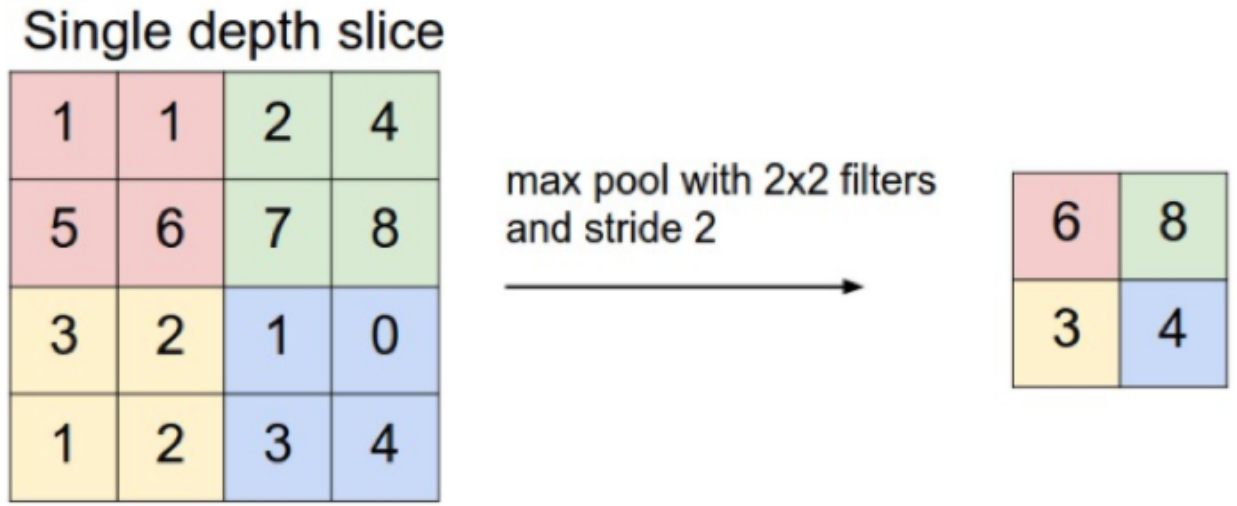
- Reduces feature dimension size
  - **Max-pooling***:* partitions input into a set of non-overlapping rectangles and outputs the maximum value for each chunk



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

Pooling Layer

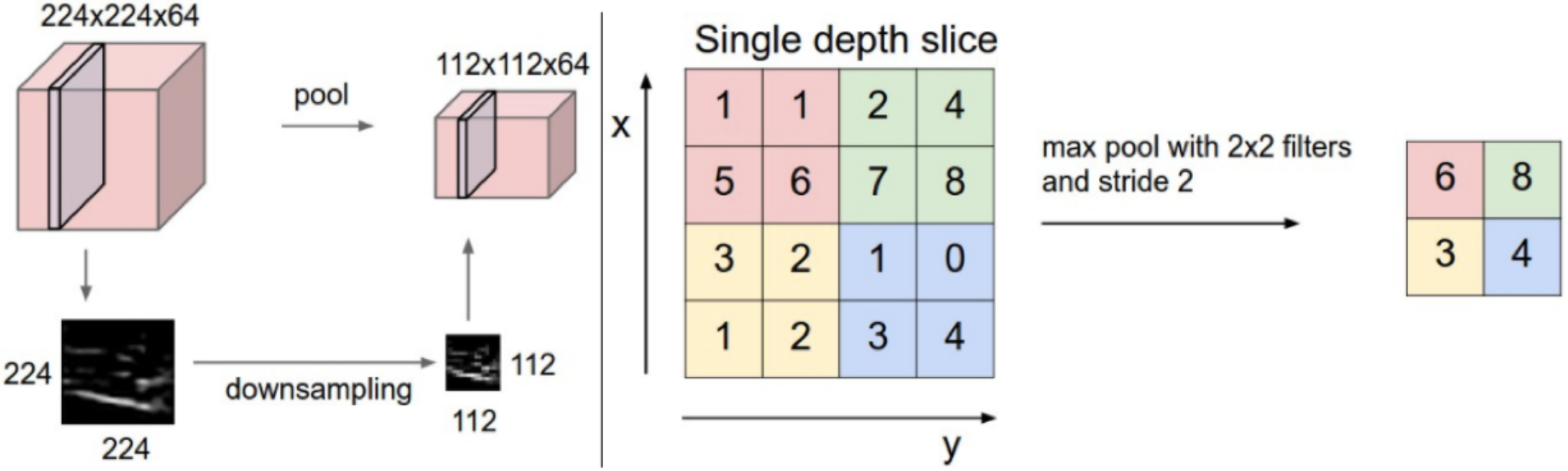
- Reduces feature dimension size
  - **Max-pooling***:* partitions input into a set of non-overlapping rectangles and outputs the maximum value for each chunk



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

Pooling Layer

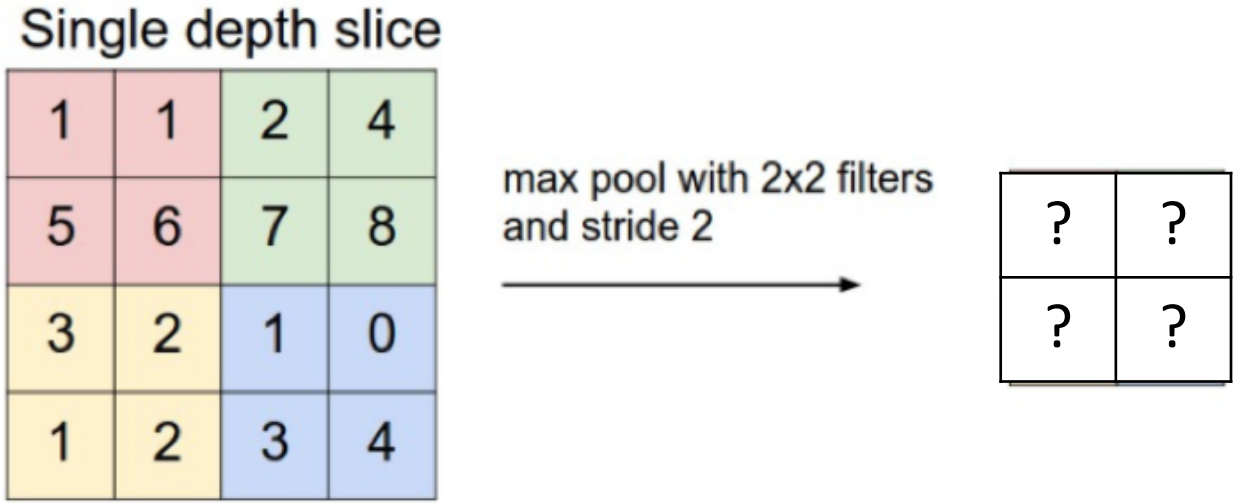
- Reduces feature dimension size
  - **Max-pooling***:* partitions input into a set of non-overlapping rectangles and outputs the maximum value for each chunk



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

Pooling Layer

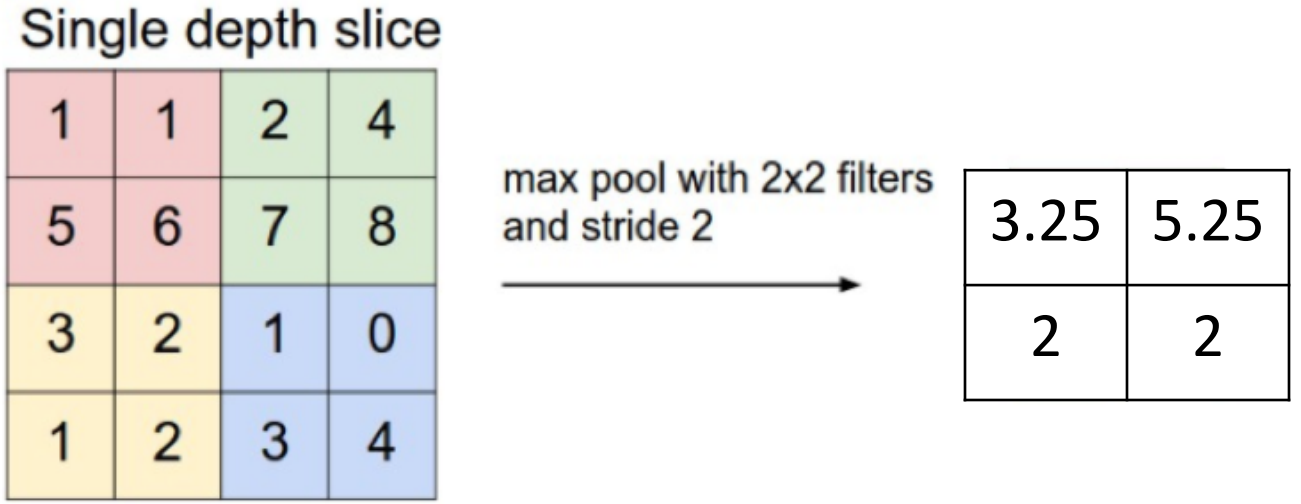
- Reduces feature dimension size
  - **Max-pooling***:* partitions input into a set of non-overlapping rectangles and outputs the maximum value for each chunk
  - **Average-pooling***:* partitions input into a set of non-overlapping rectangles and outputs the average value for each chunk



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

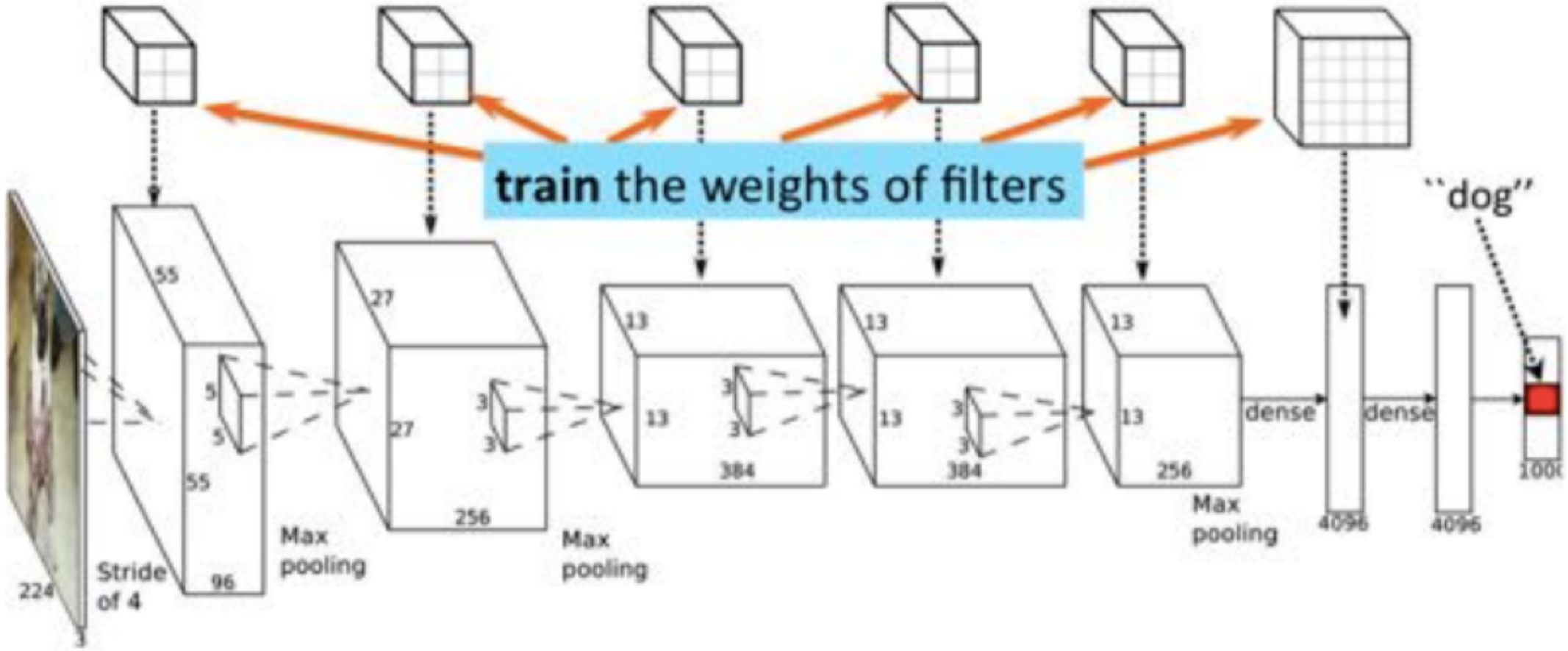
Pooling Layer

- Reduces feature dimension size
  - **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

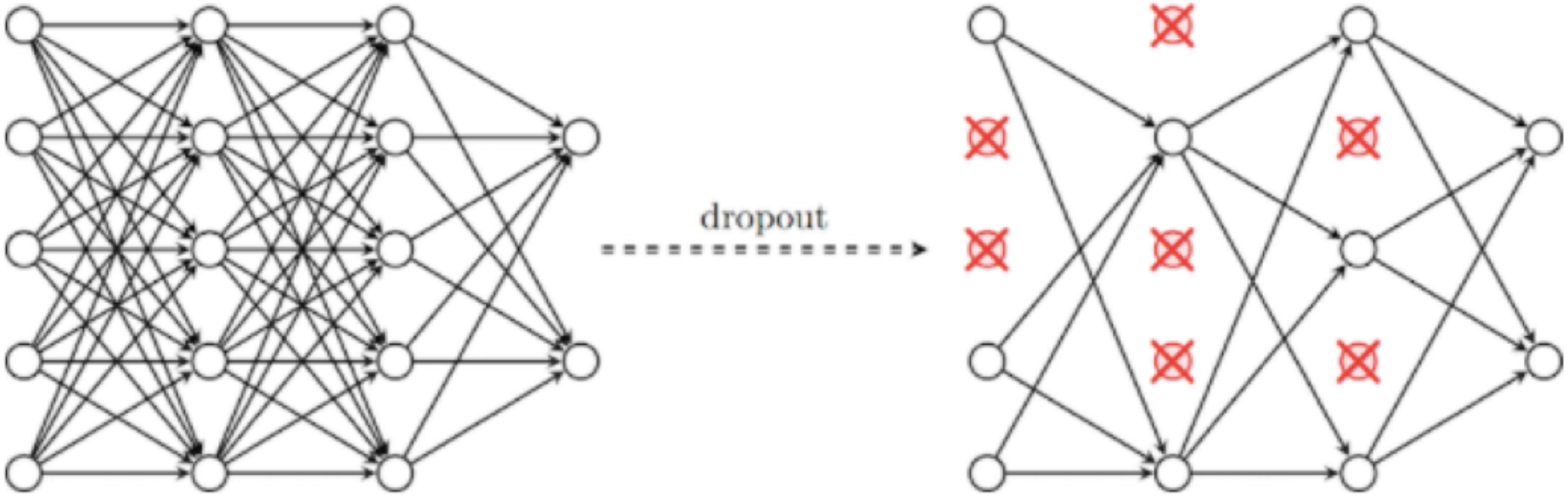
Training CNN



Slide Credit: http://www.cs.utoronto.ca/~fidler/teaching/2015/slides/CSC411/11_nn2.pdf

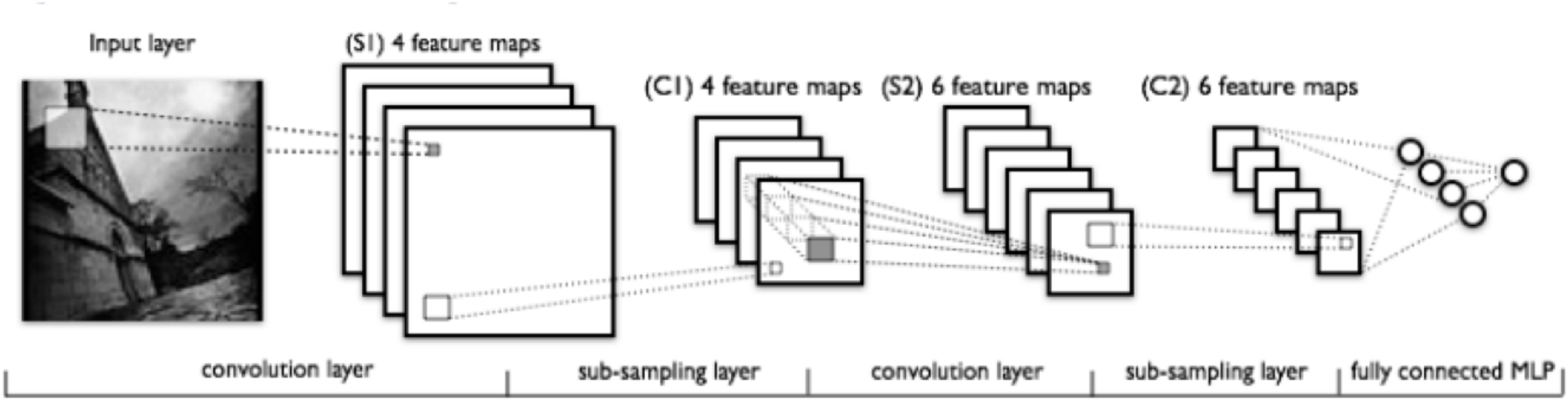
Training CNN: Dropout

- In single training iteration, *eliminate nodes from the network*
- Why? This forces the neural network to learn redundancy by developing a consensus (ensemble) of neurons within a layer that are predictive



https://cambridgespark.com/content/tutorials/convolutional-neural-networks-with-keras/index.html

Revisiting LeNet

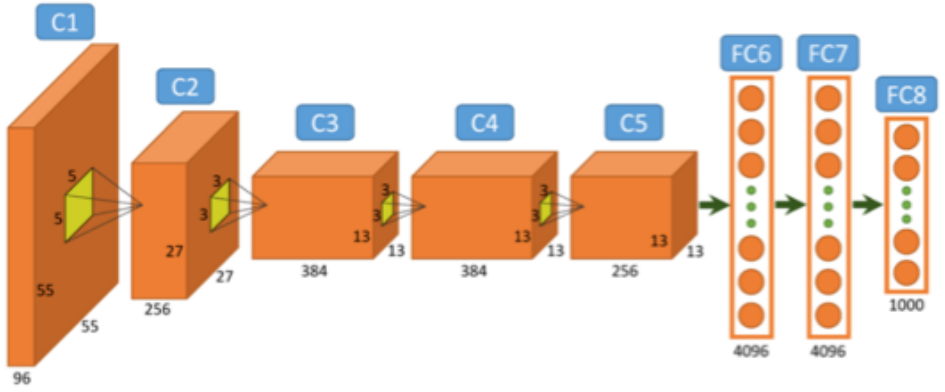


http://deeplearning.net/tutorial/lenet.html

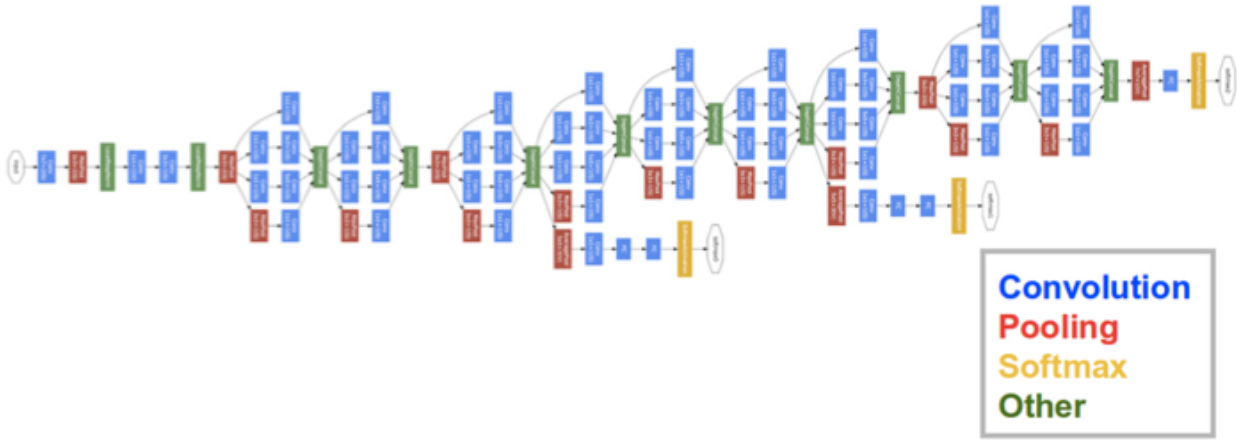
Y. Lecun ; L. Bottou ; Y. Bengio ; P. Haffner; Gradient-based learning applied to document recognition; 1998

Over Time, CNNs are Becoming “Deeper”

AlexNet (2012)



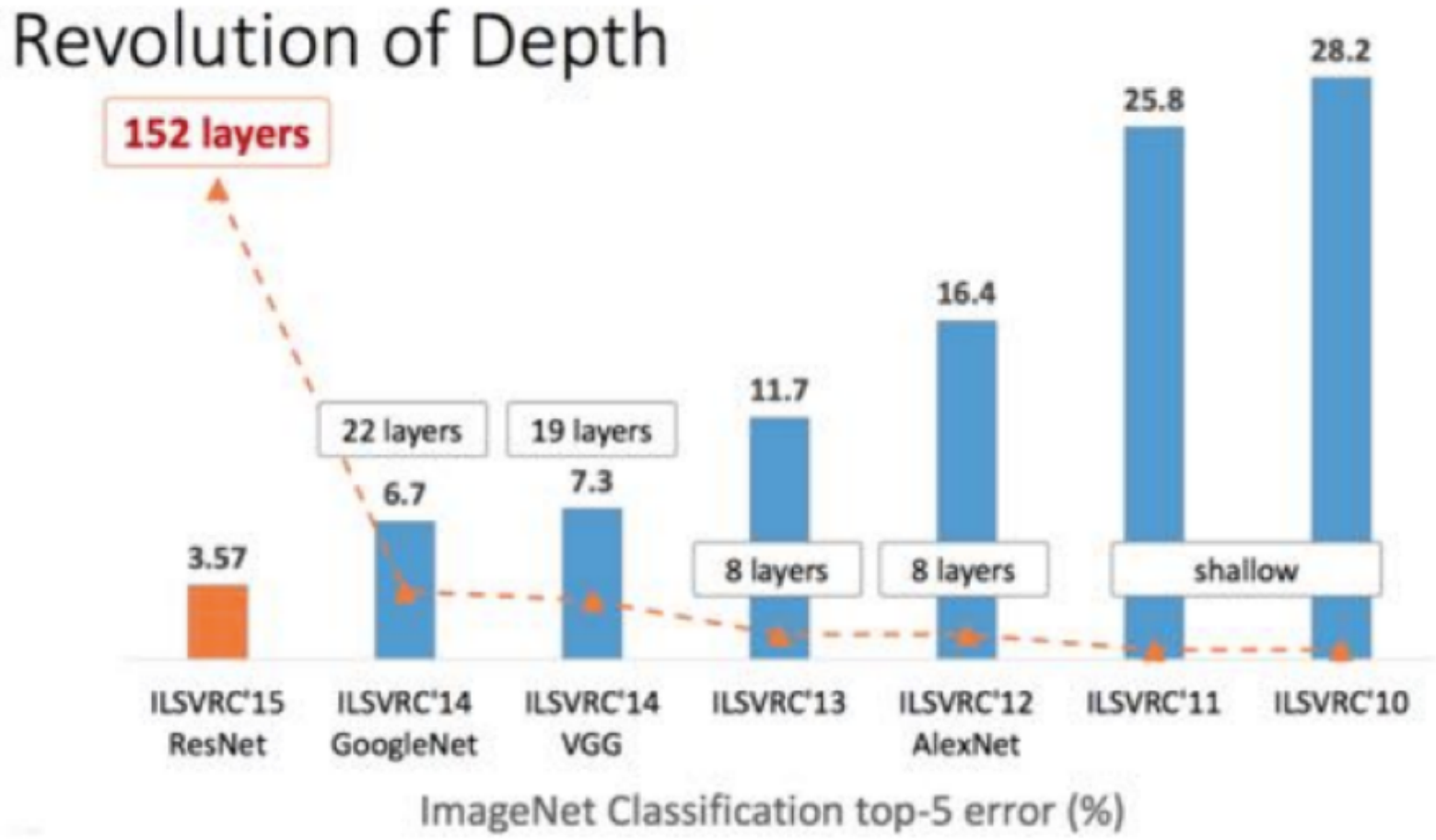
GoogLeNet (2014)



http://joelouismarino.github.io/blog_posts/blog_googlenet_keras.html

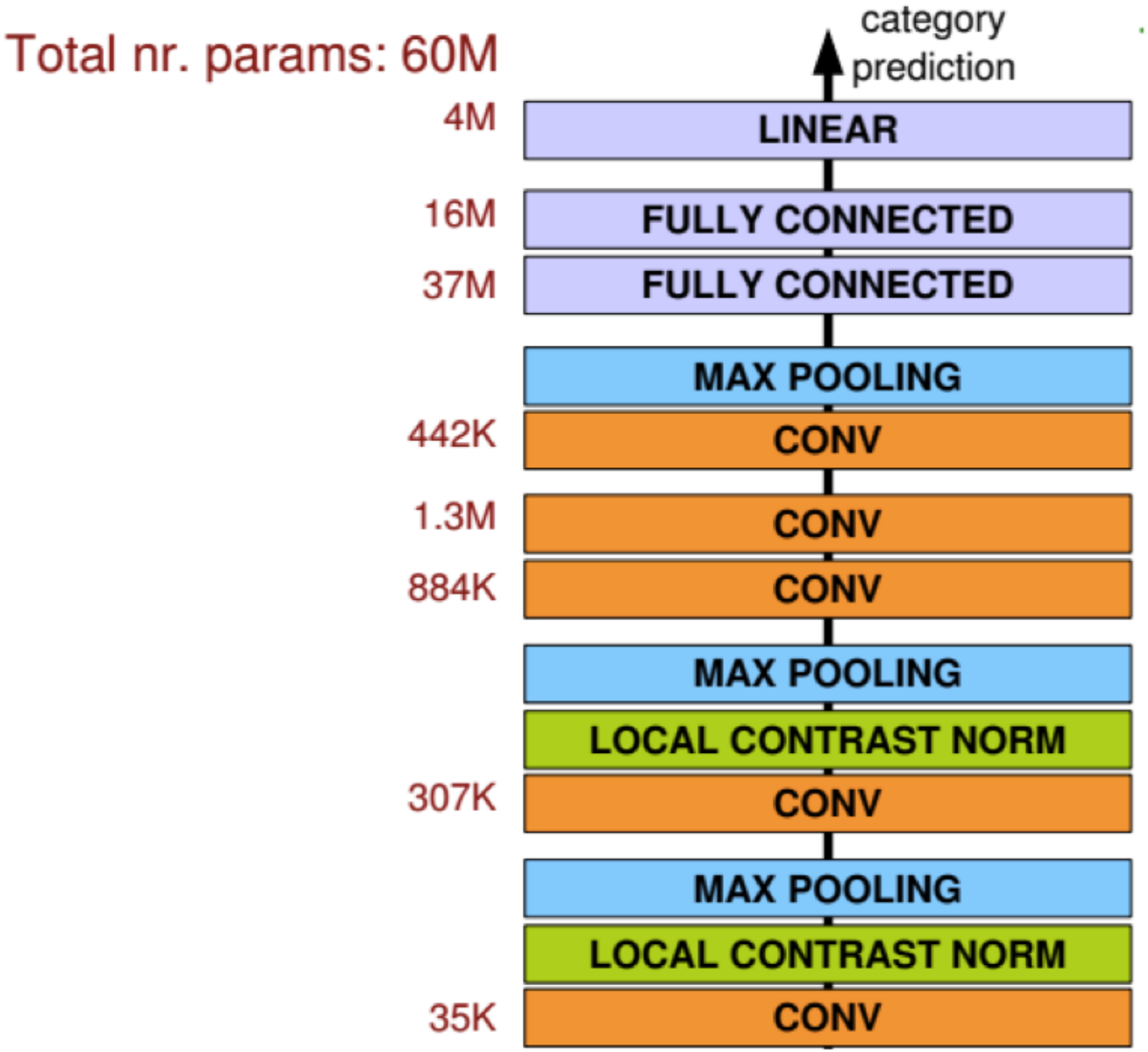
https://www.researchgate.net/figure/Architecture-of-Alexnet-From-left-to-right-input-to-output-five-convolutional-layers_fig2_312303454

Over Time, CNNs are Becoming “Deeper”



Slide: R. Liao, Paper: [He, K., Zhang, X., Ren, S. and Sun, J., 2015. Deep Residual Learning for Image Recognition.arXiv:1512.03385, 2016]

AlexNet

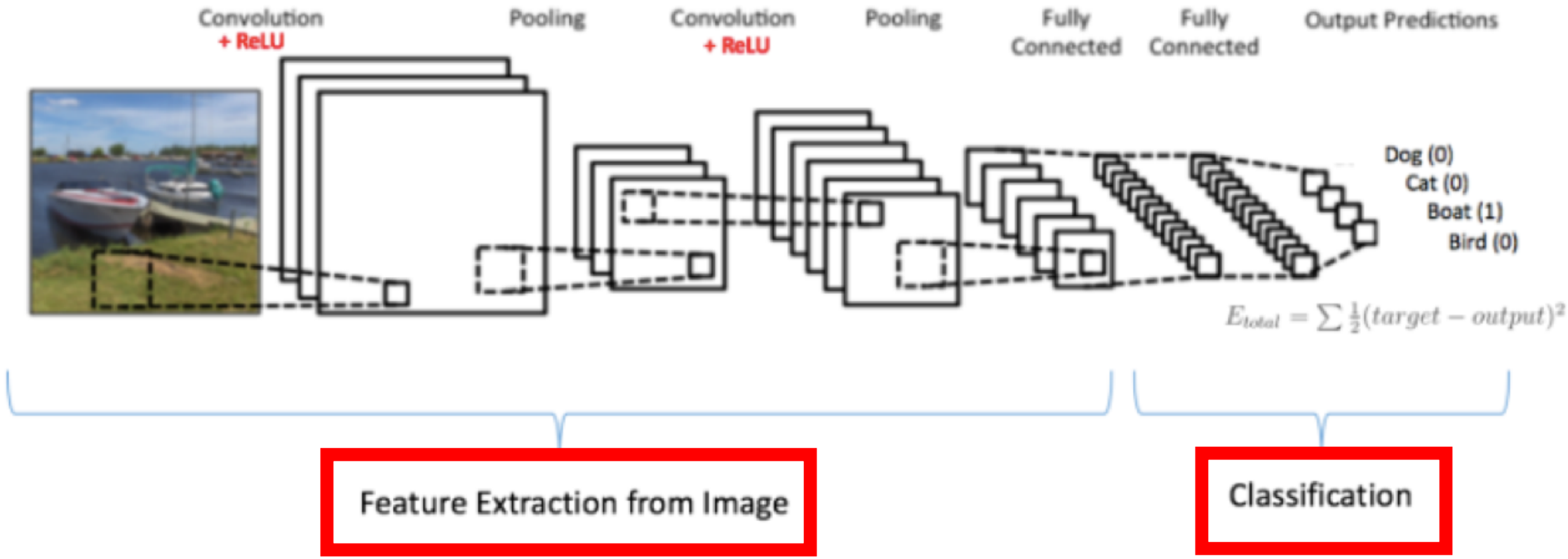


Slide Credit: http://www.cs.utoronto.ca/~fidler/teaching/2015/slides/CSC411/11_nn2.pdf

Today's Topics

- Revisiting History of Modern Neural Networks and "Deep Learning"
- Convolutional Neural Networks (CNN)
- Deep Features
- Lab

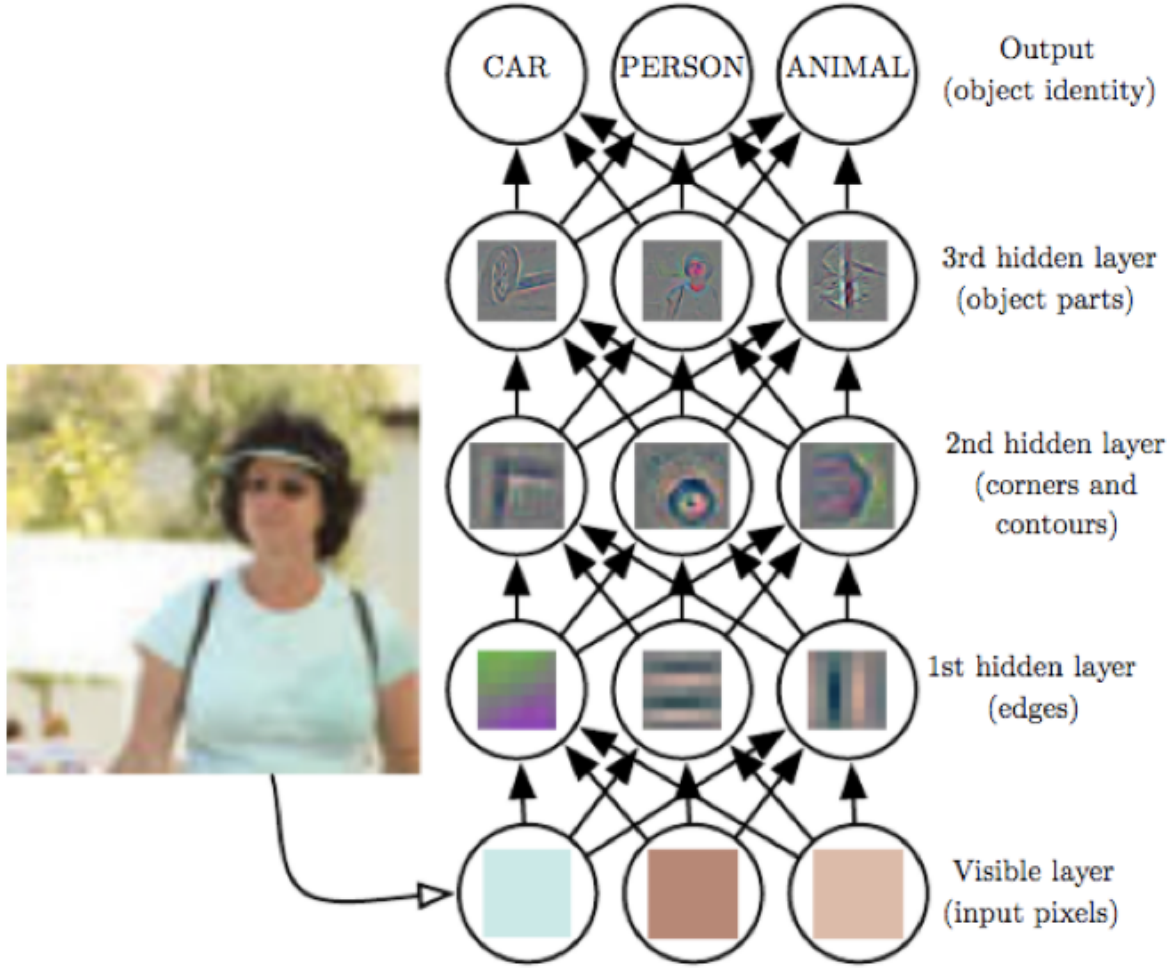
CNN: Intuition of Different Layers



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

Understanding What Deep Learning Learns

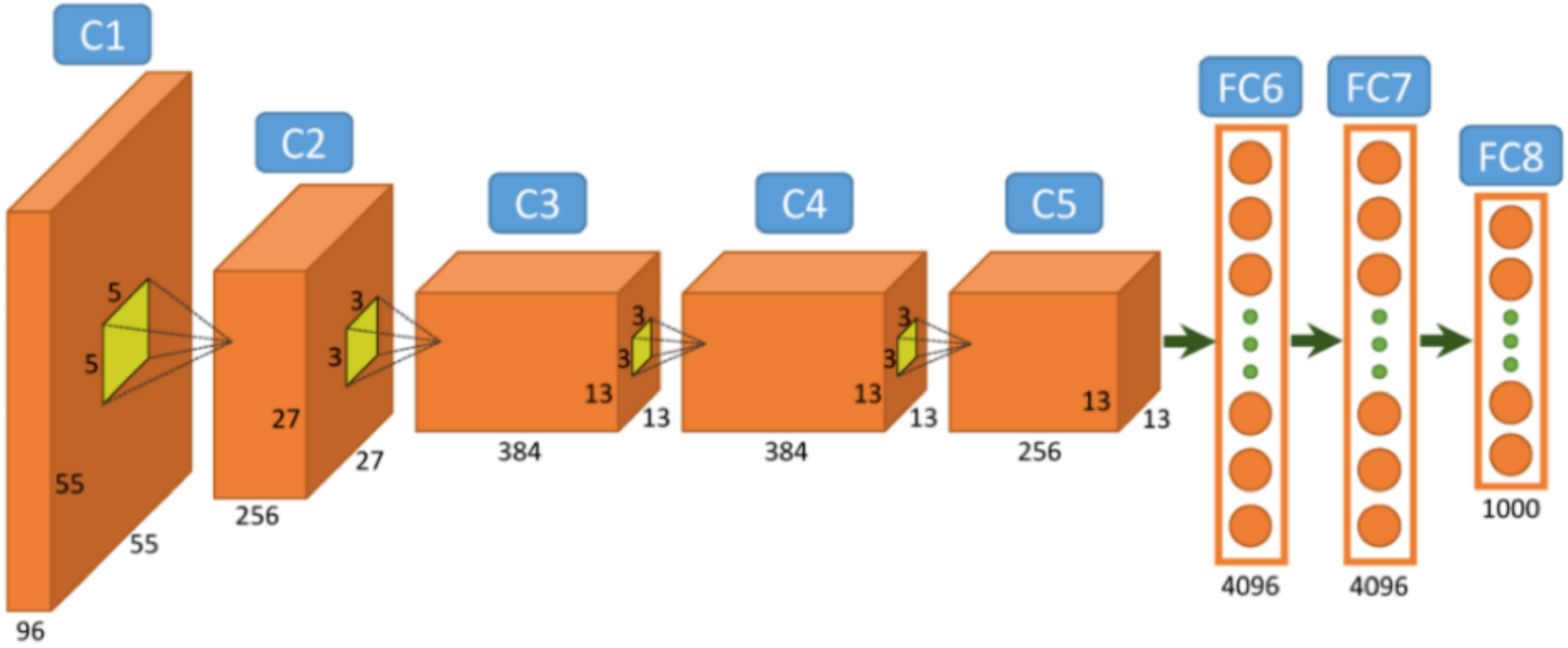
- Distributed representation:
  - e.g., predicting red truck, red car, red bird, green truck, green car, green bird, blue truck, blue car, blue bird
    - Two layers: one layer indicates color (3 nodes) and other layer indicates object (3 nodes)
- Goal: learn “right” representation of the data
- Generally, more convolution layers leads to learning more complex features



Deep Learning, Ian Goodfellow, Yoshua Bengio, and Aaron Courville

AlexNet Deep Features

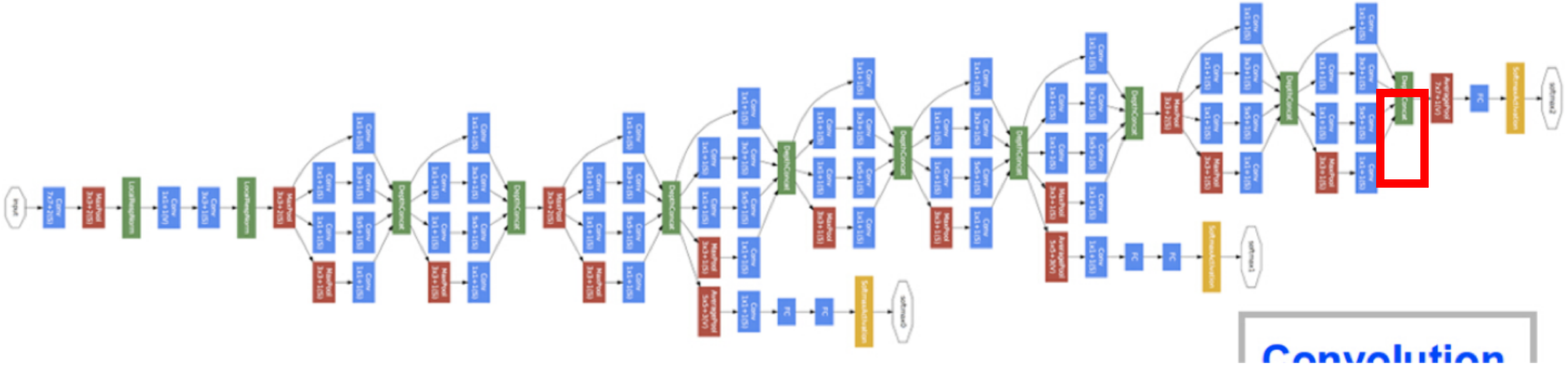
- What is the dimensionality of the fc6 feature?
- What is the dimensionality of the fc7 feature?



https://www.researchgate.net/figure/Architecture-of-Alexnet-From-left-to-right-input-to-output-five-convolutional-layers_fig2_312303454

GoogleNet (Inception) Deep Features

- What is the dimensionality of the inception features?



http://joelouismarino.github.io/blog_posts/blog_googlenet_keras.html

Today’s Topics

- Revisiting History of Modern Neural Networks and “Deep Learning”
- Convolutional Neural Networks (CNN)
- Deep Features
- Lab