Lesson 2
Convolutional Neural Networks (CNN)

Outline

- Introduction
- CNN history
- CNN layers
- Common CNN architectures

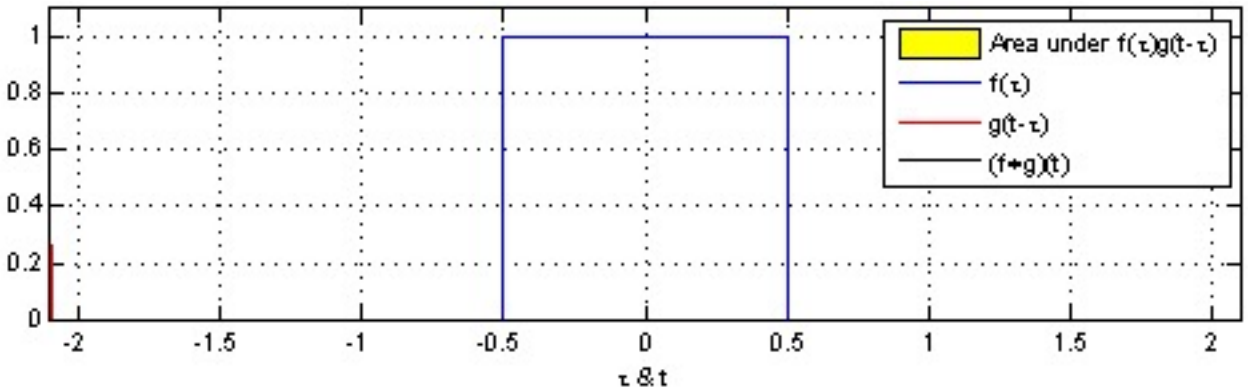
Introduction

Convolution function

- Convolution is a mathematical operation on two functions (f and g) that produces a third function (f*g) that expresses how the shape of one is modified by the other.
  - the integral of the product of the two functions after one is reversed and shifted

$$(f * g)(t) \quad \stackrel{\text{def}}{=} \int_{-\infty}^{\infty} f(\tau)\, g(t-\tau)\, d\tau$$

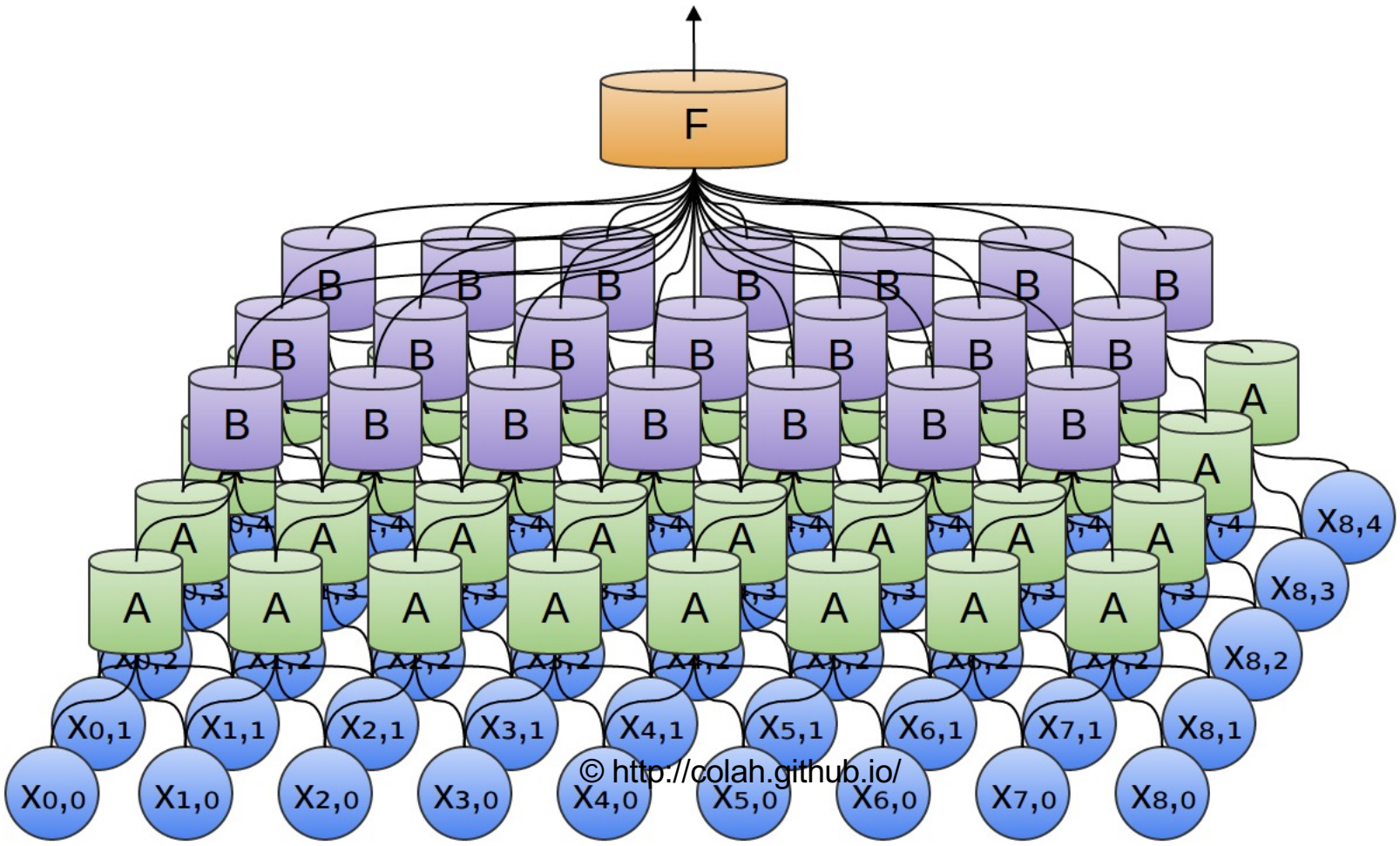
$$= \int_{-\infty}^{\infty} f(t-\tau)\, g(\tau)\, d\tau.$$



Convolutional neural networks

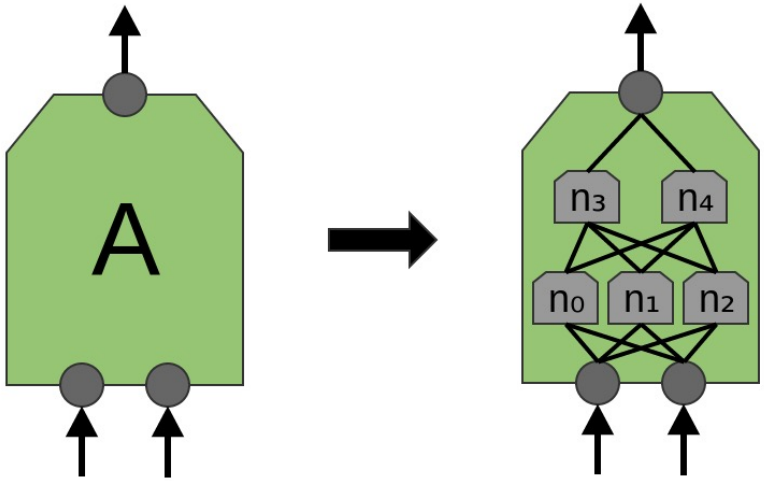
- Multiple convolutional layers
- Exploit the “spatial” structured characteristic of data
- Khai thác đặc trưng cấu trúc ”spatial” của dữ liệu
- Using multiple identical copies of the same neuron block
  - The number of neurons is big
  - The number of layers or the deep of the network is big
  - But the number of weights that need to be learned for the model is significantly smaller

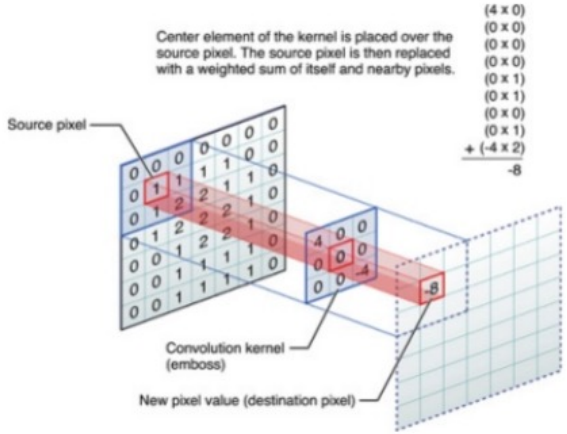
Hình dung về CNN trong không gian 2 chiều

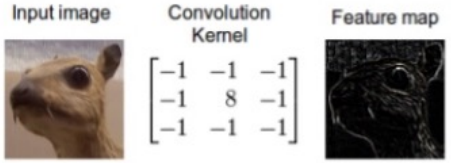


Each neuron block can be a small neural network

- Blocks are called filters or kernels







© http://colah.github.io/

Convolution in image processing

input

Kernel for blurring

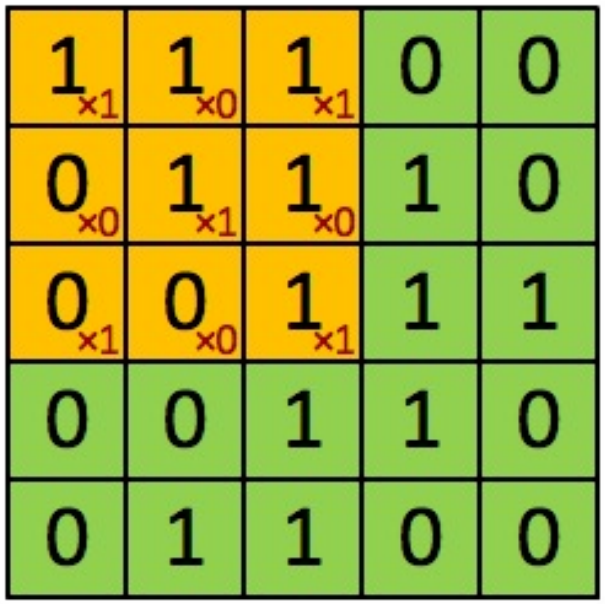
0.0625	0.125	0.0625
0.125	0.25	0.125
0.0625	0.125	0.0625

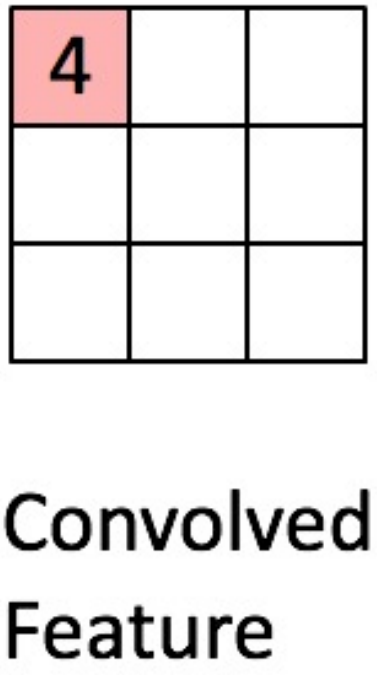
tf.nn.conv2d

output

© http://web.stanford.edu/class/cs20si

How filters work



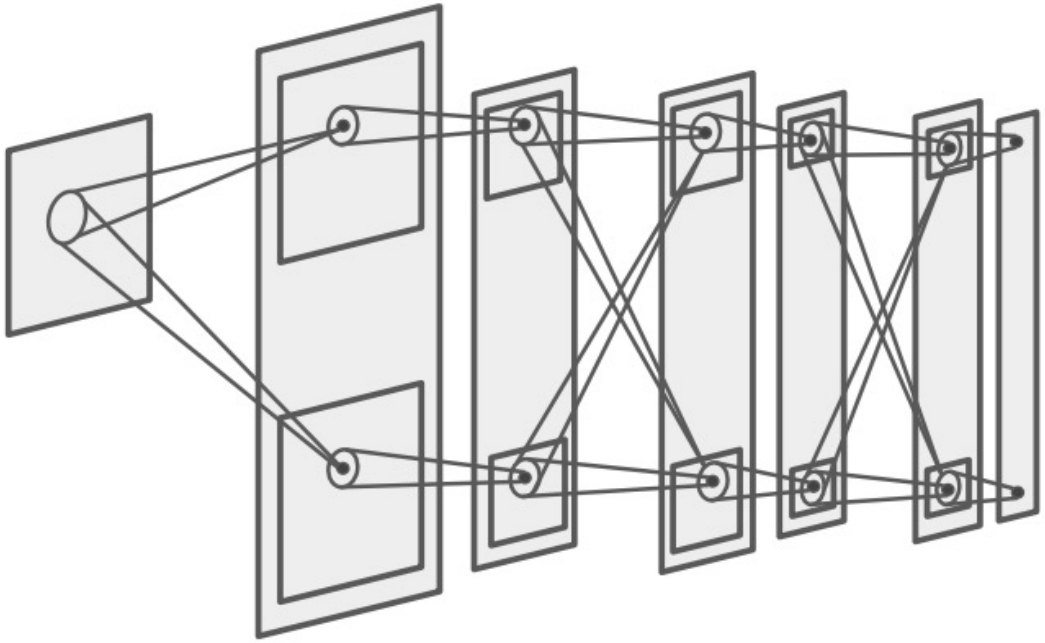


© http://deeplearning.stanford.edu/

CNN history

CNN history

Neocognitron
[Fukushima 1980]



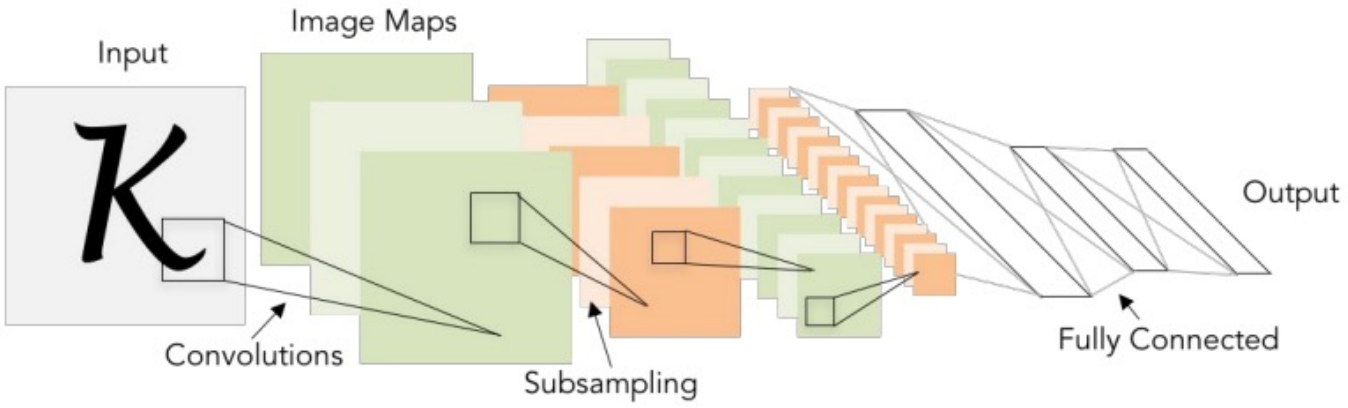
"sandwich" architecture (SCSCSC…)
simple cells: modifiable parameters
complex cells: perform pooling

- The idea of CNNs first came from the work of Fukushima in 1980

CNN history (2)

Gradient-based learning applied to document recognition

[LeCun, Bottou, Bengio, Haffner 1998]



- In 1998, LeCun applied BackProp to train CNNs for the text recognition problem

CNN history (3)

ImageNet Classification with Deep Convolutional Neural Networks
[Krizhevsky, Sutskever, Hinton, 2012]

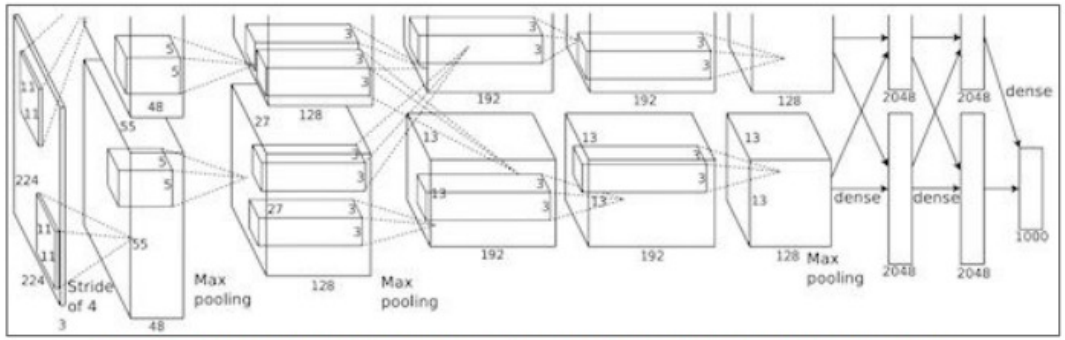
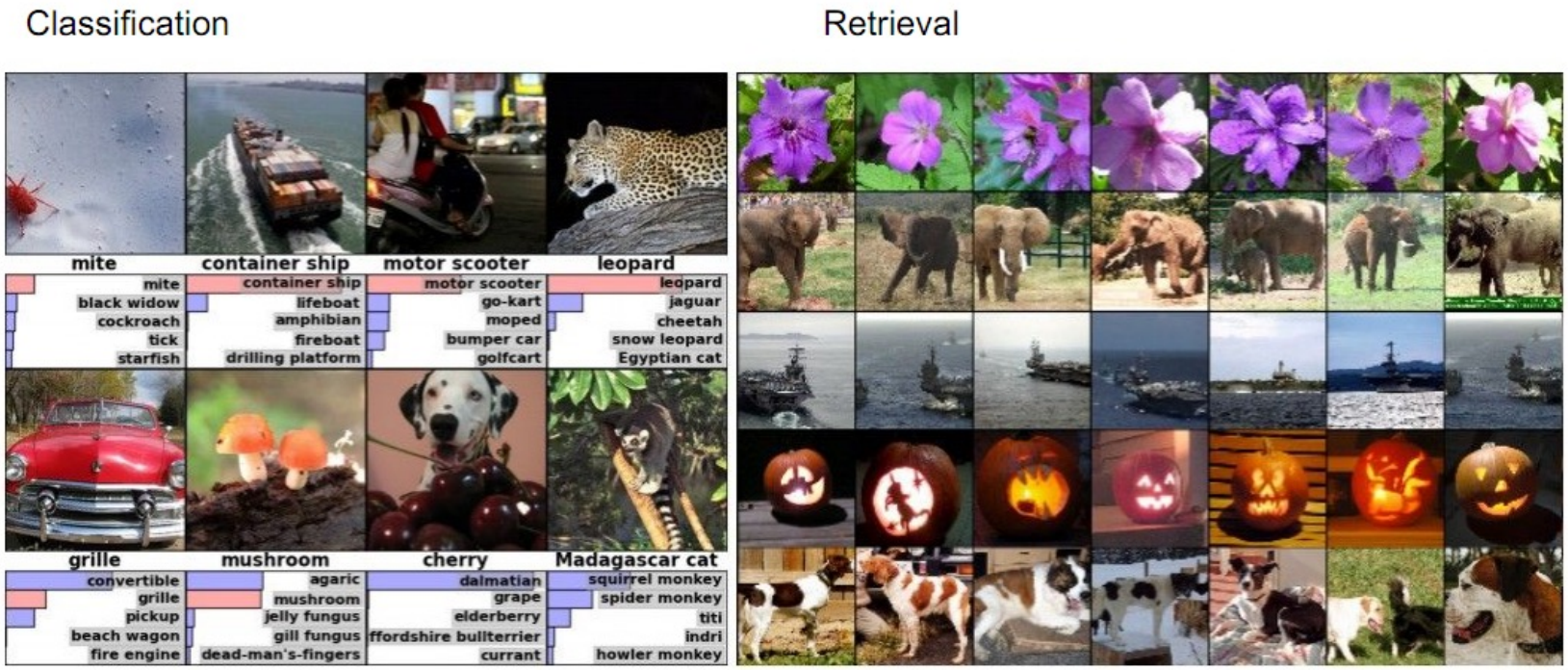


Figure copyright Alex Krizhevsky, Ilya Sutskever, and Geoffrey Hinton, 2012. Reproduced with permission.

"AlexNet"

- In 2012, CNNs made a big splash when they won the ILSRC 2012 competition, far surpassing the 2nd method of the traditional computer vision approach.

CNN history (4)



Figures copyright Alex Krizhevsky, Ilya Sutskever, and Geoffrey Hinton, 2012. Reproduced with permission.

- Currently, CNNs are applied everywhere, for example in image classification problems, image querying

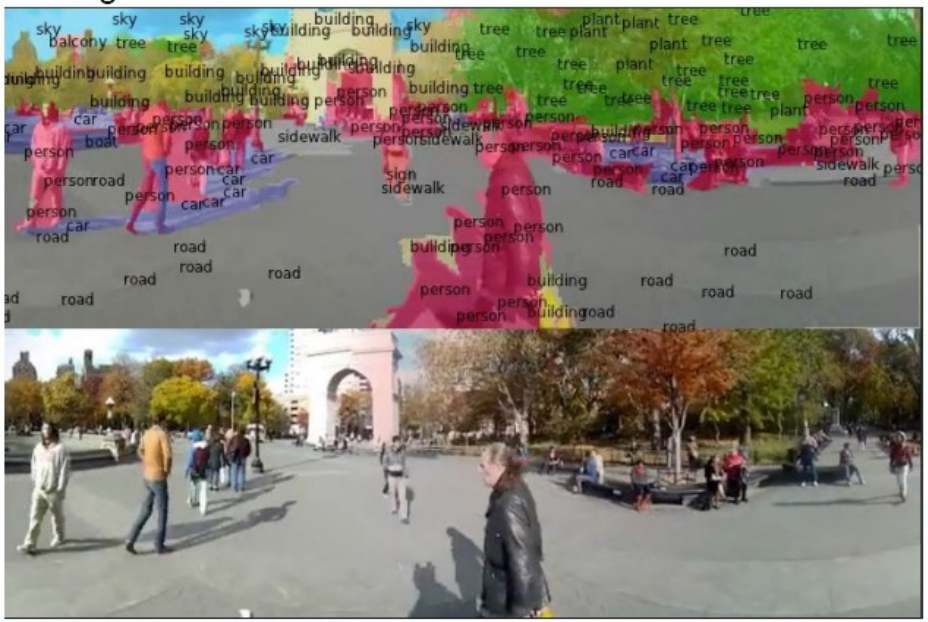
CNN history (5)

Detection

Figures copyright Shaoqing Ren, Kaiming He, Ross Girschick, Jian Sun, 2015. Reproduced with permission.

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

Segmentation



Figures copyright Clement Farabet, 2012. Reproduced with permission.

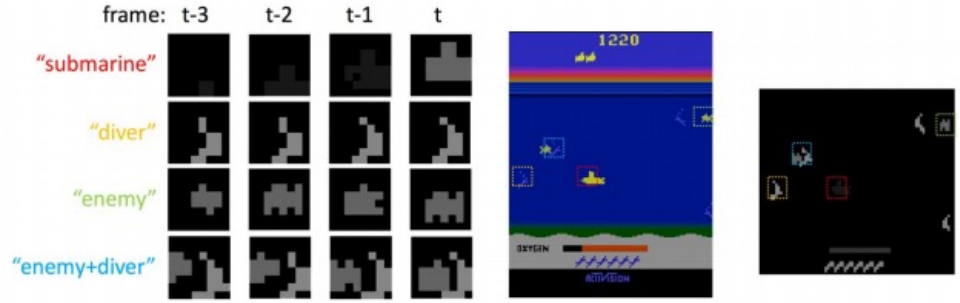
[Farabet et al., 2012]

- Application of CNNs in object detection and image segmentation

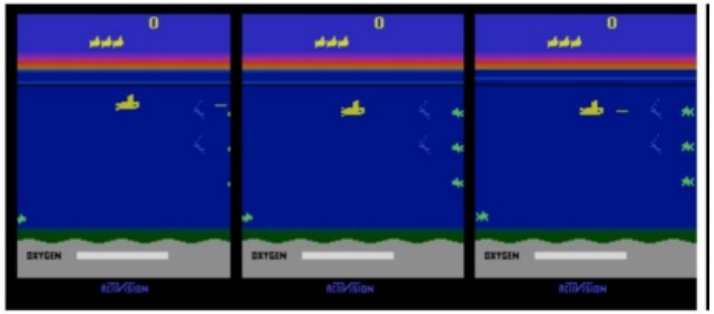
CNN history (6)

Images are examples of pose estimation, not actually from Toshev & Szegedy 2014. Copyright Lane McIntosh.

[Toshev, Szegedy 2014]



[Guo et al. 2014]



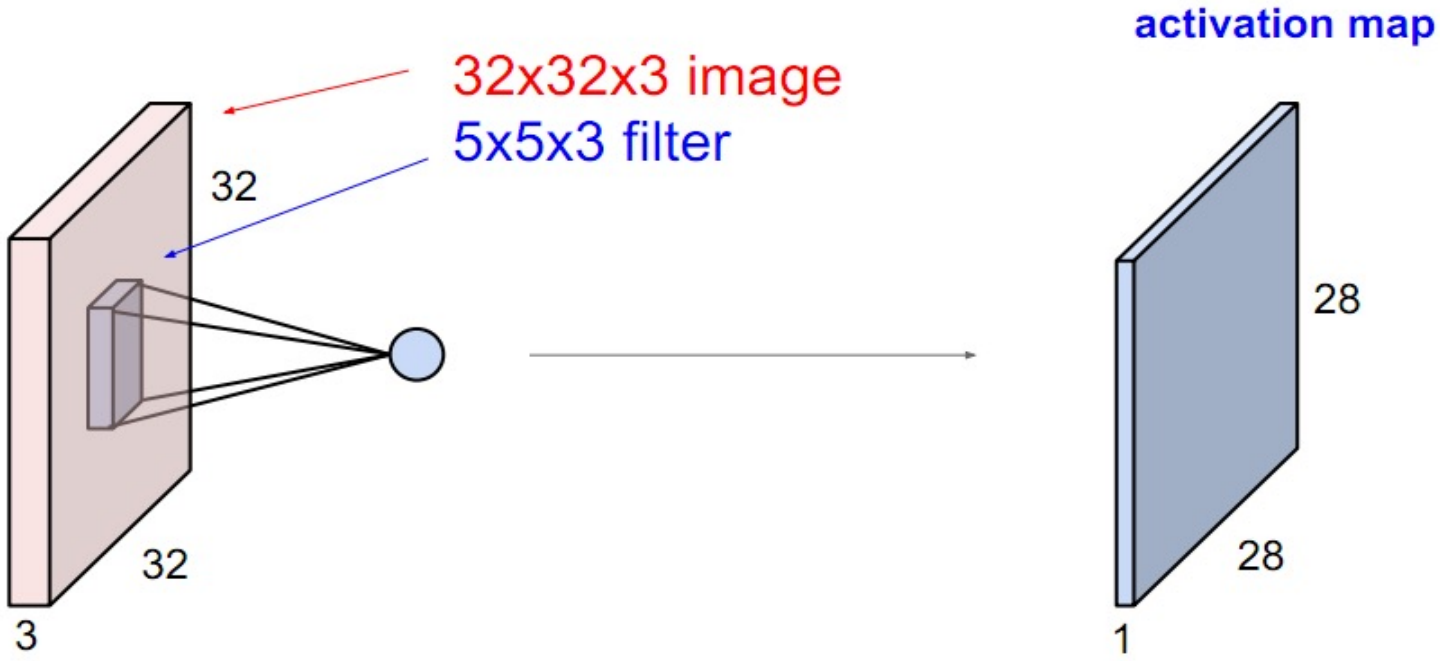
Figures copyright Xiaoxiao Guo, Satinder Singh, Honglak Lee, Richard Lewis, and Xiaoshi Wang, 2014. Reproduced with permission.

- Application of CNNs in human pose recognition, in games...

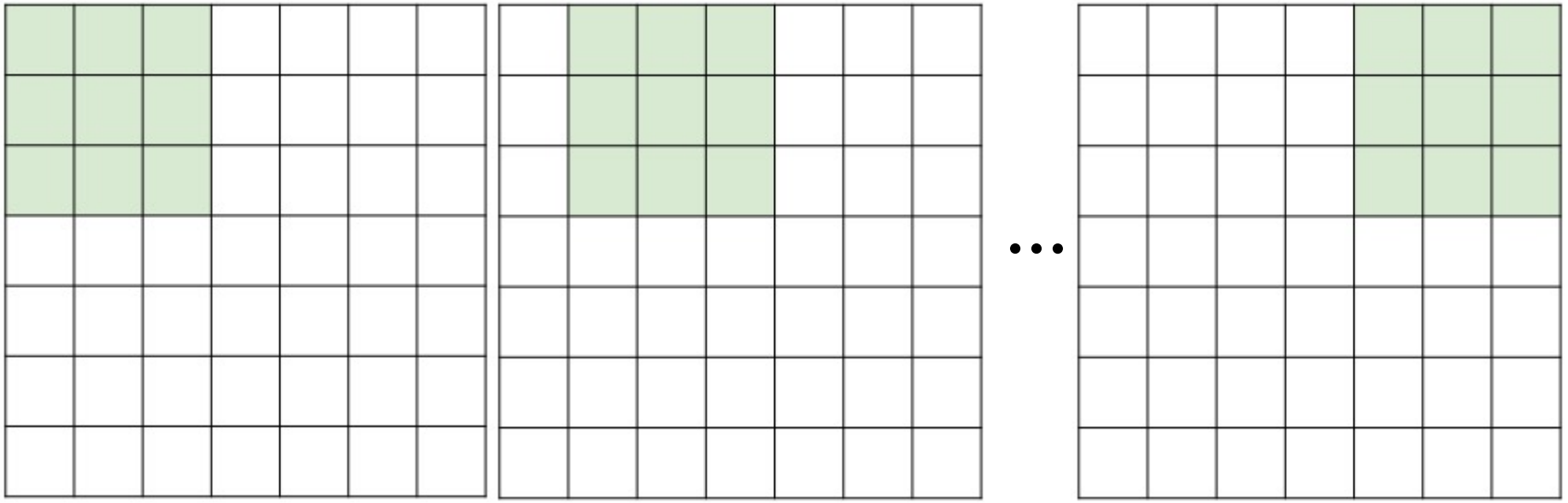
CNN layers

Convolutional layer (Conv)

- Unlike a fully connected neuron, each convolutional neuron (filter) is only locally connected to the input data.
- The convolutional neuron slides from left to right and from top to bottom of the input data block and computes to generate an activation map.
- The depth of the convolutional neuron is equal to the depth of the input data block

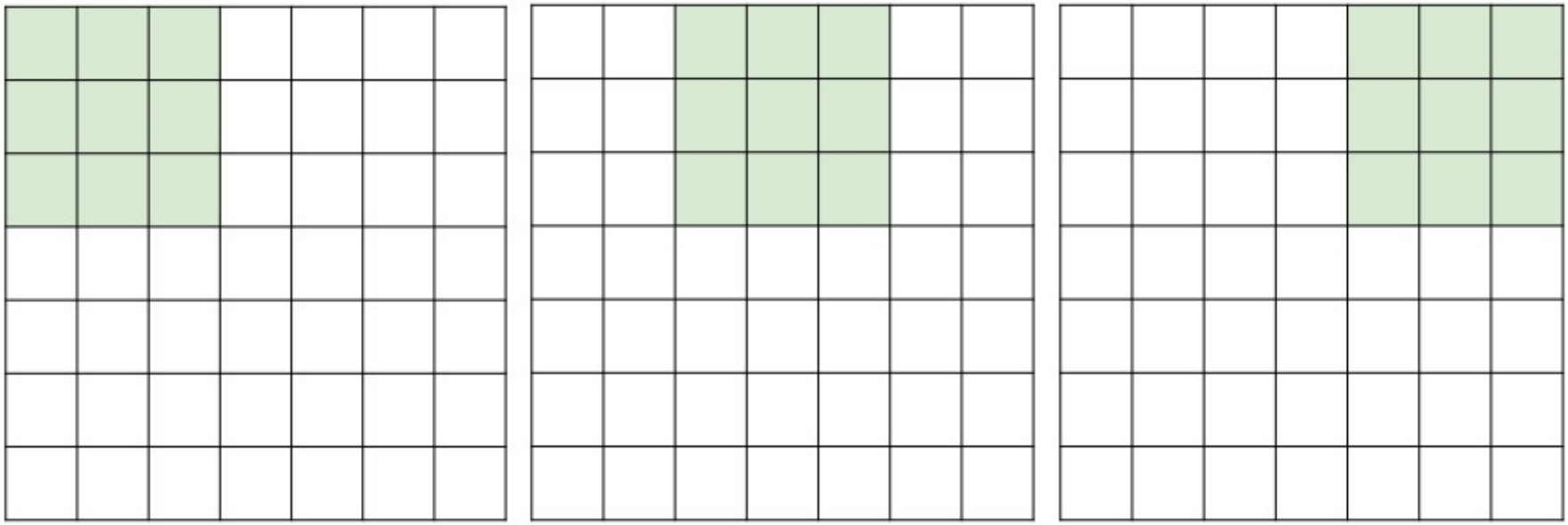


Conv layer



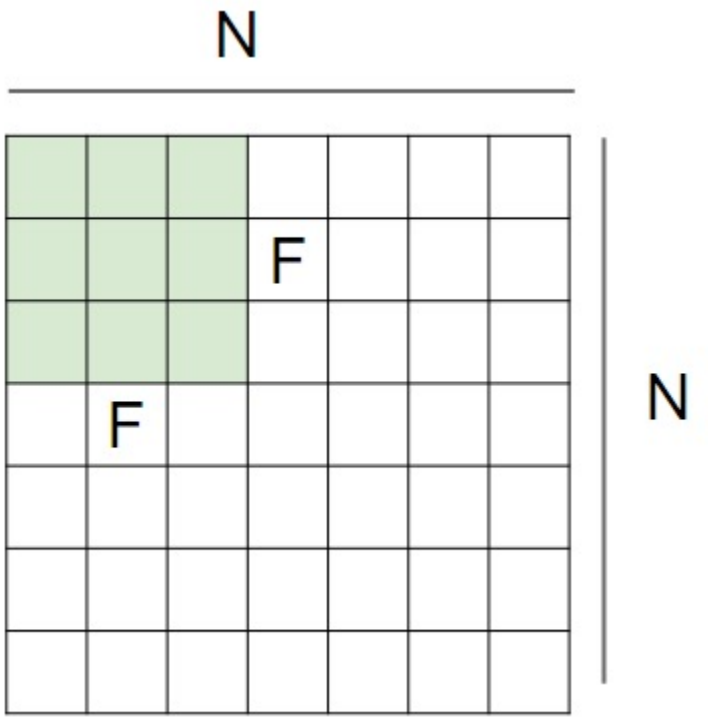
- Stride = 1
- Input size 7x7, neuron size 3x3
- Output size 5x5

Lớp tích chập



- Bước nhảy stride = 2
- Đầu vào kích thước 7x7, nơ-ron kích thước 3x3
- Đầu ra kích thước 3x3

Conv layer (2)

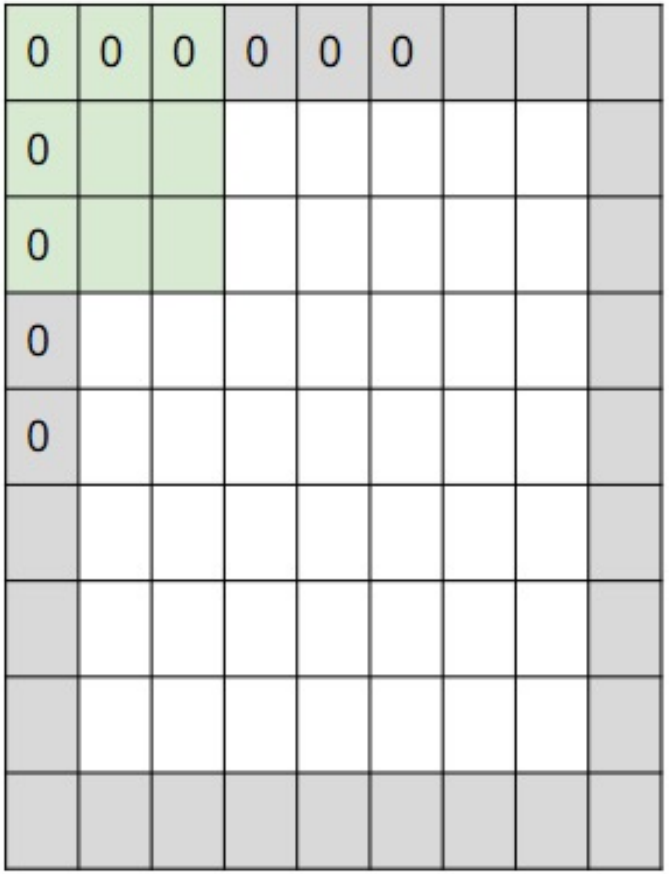


Output size:
(N - F) / stride + 1

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

Conv layer (3)

- To preserve the output size, usually add padding borders with zeros (zero padding).
- Example: input size 7x7, neuron size 3x3, stride 1, padding width 1.
- Then the output size is 7x7

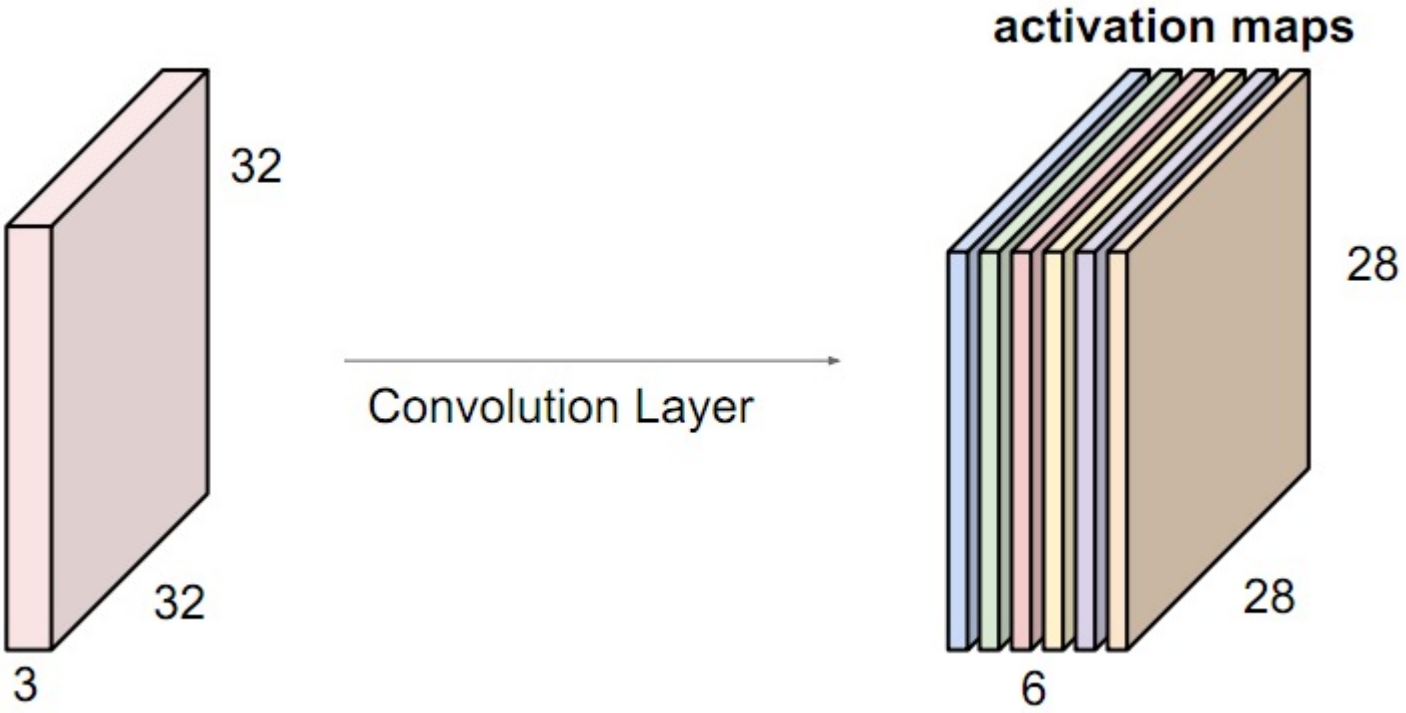


Conv layer (4)

- Assuming another convolutional neuron is added, it behaves similarly and generates a second activation map
- Note that the weights of the convolutional neurons are differen

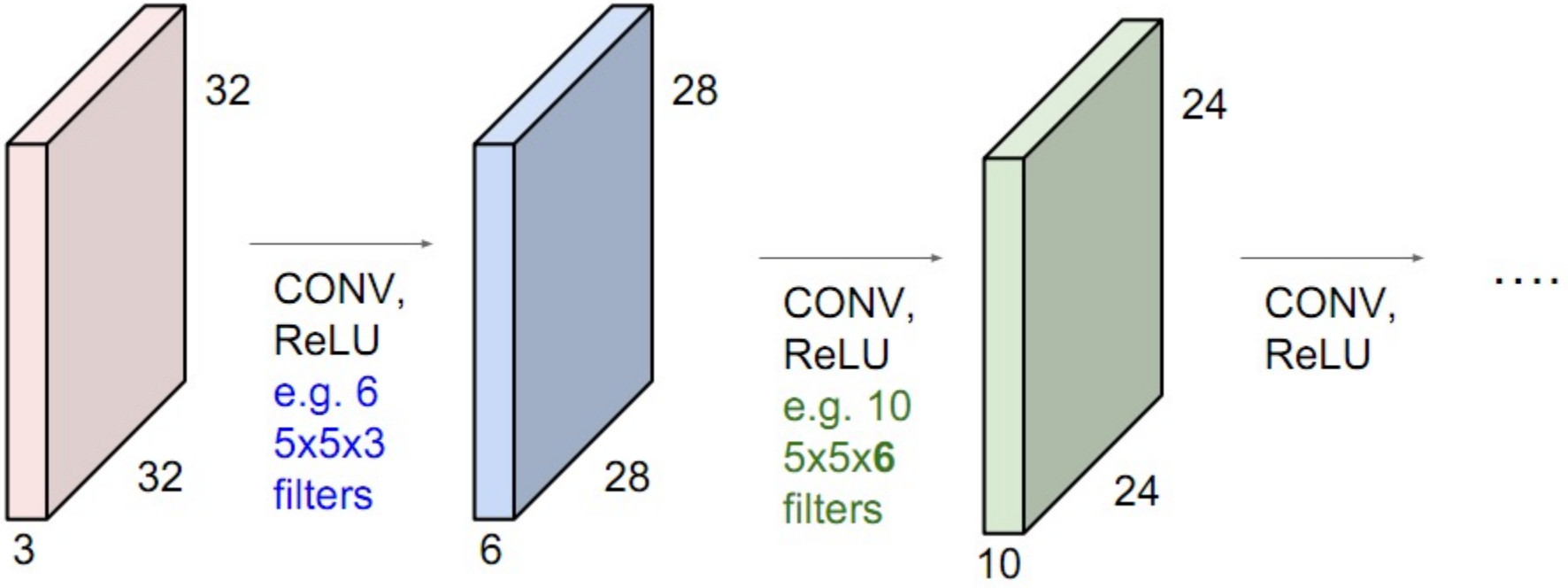
Conv layer (5)

- Assuming there are 6 convolutional neurons, then they will generate 6 activation maps
- Activation maps stitched together into a “new image”



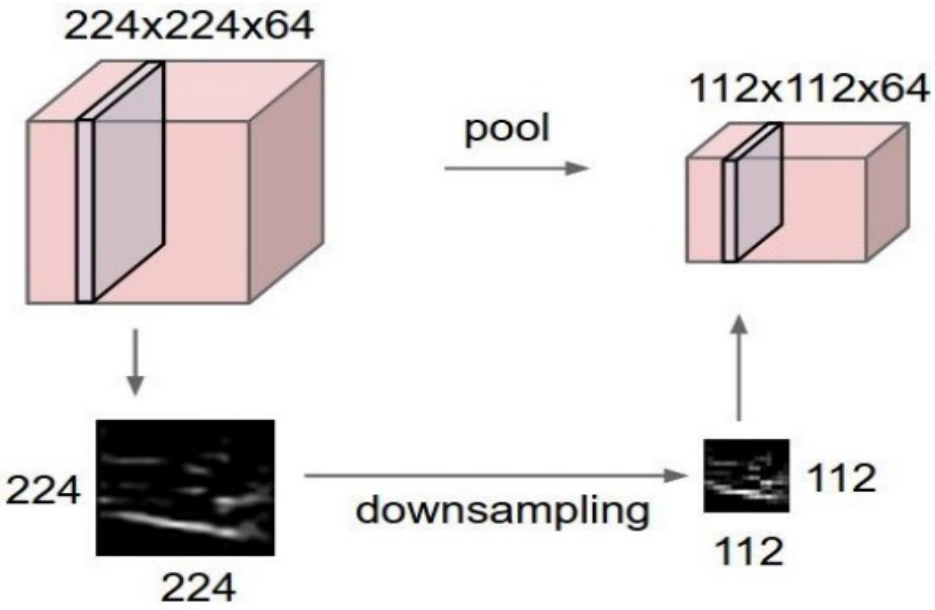
CNNs

- A convolutional neural network is a sequence of convolutional layers stacked on top of each other, and they are interspersed with activation functions (e.g. ReLU).

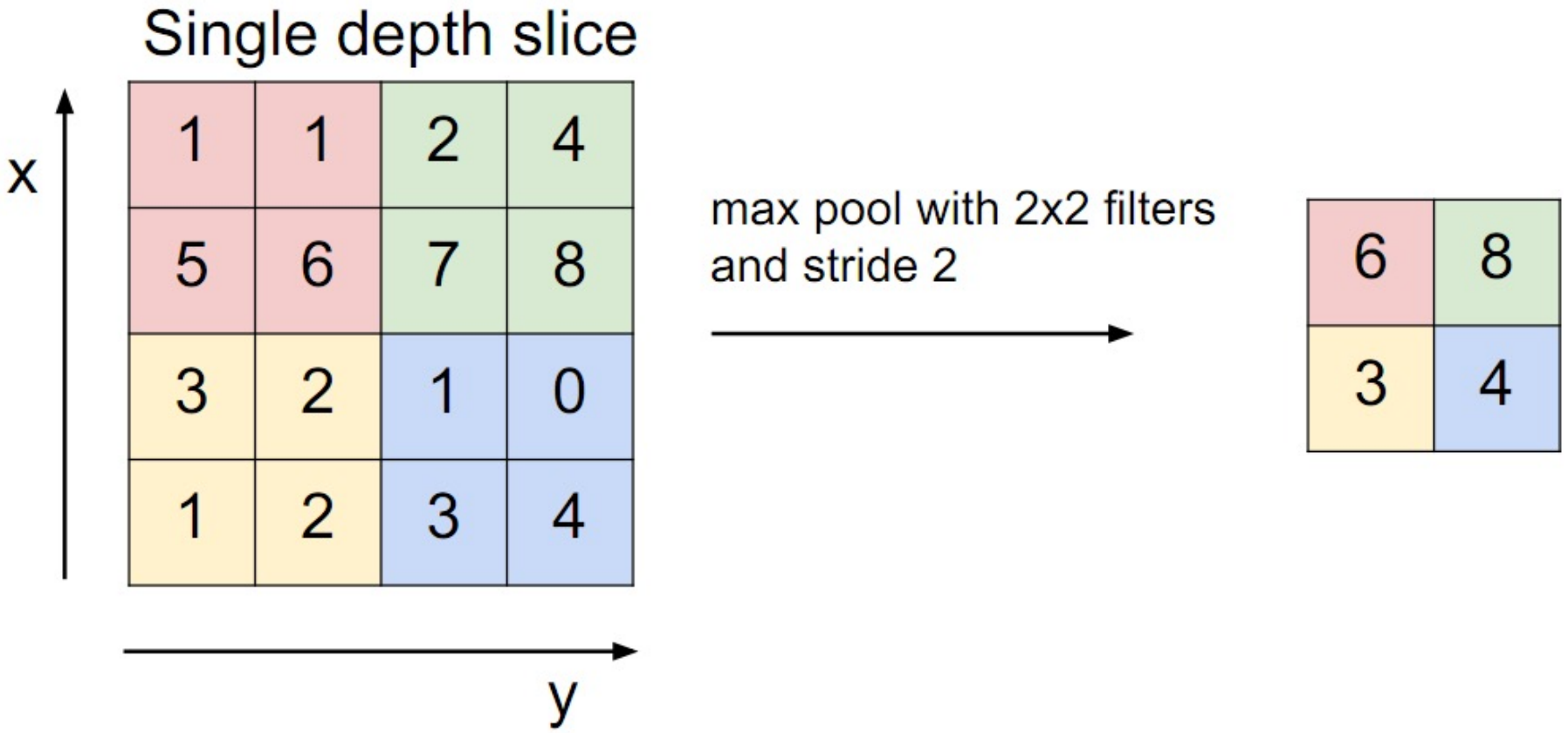


Pooling layer

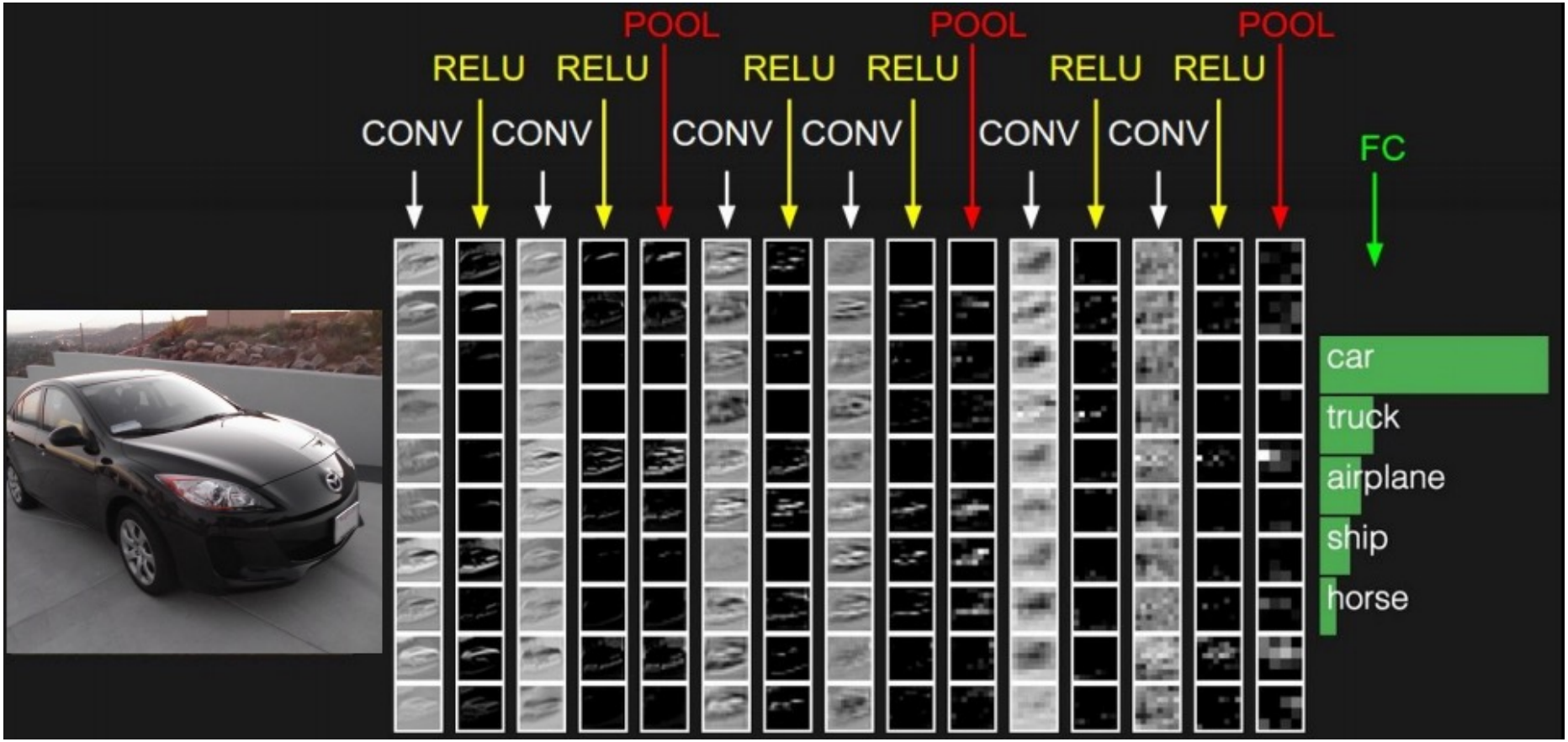
- Pooling is a down-sampling technique.
- Helps reduce the block data resolution to reduce memory and computation consumptions.
- Work independently on each activation map.
- The max pooling layer helps the network to make the representation become invariant to the small (local) translation, or deformation (deformation-invariant) of the input data.



Max pooling layer



CNNs

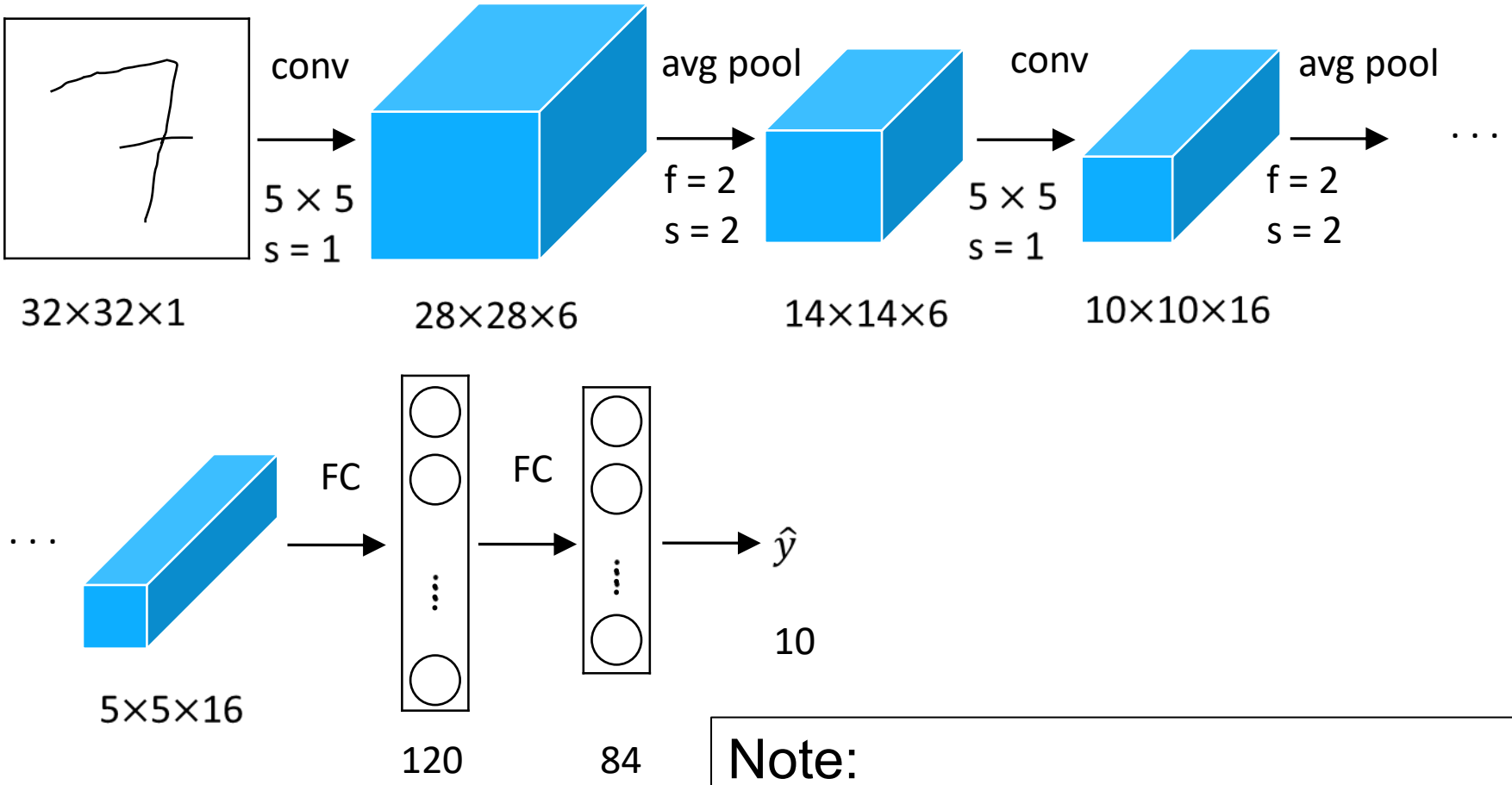

POOL
POOL
POOL
RELU
RELU
RELU
RELU
RELU
RELU
CONV
CONV
CONV
CONV
CONV
CONV
FC
car
truck
airplane
ship
horse

Common CNN architectures

Common CNN architectures

- LeNet-5
- AlexNet
- VGG
- GoogleNet
- ResNet

LeNet-5



Note:
Output size = (N+2P-F)/stride + 1

This slide is taken from Andrew Ng

[LeCun et al., 1998]

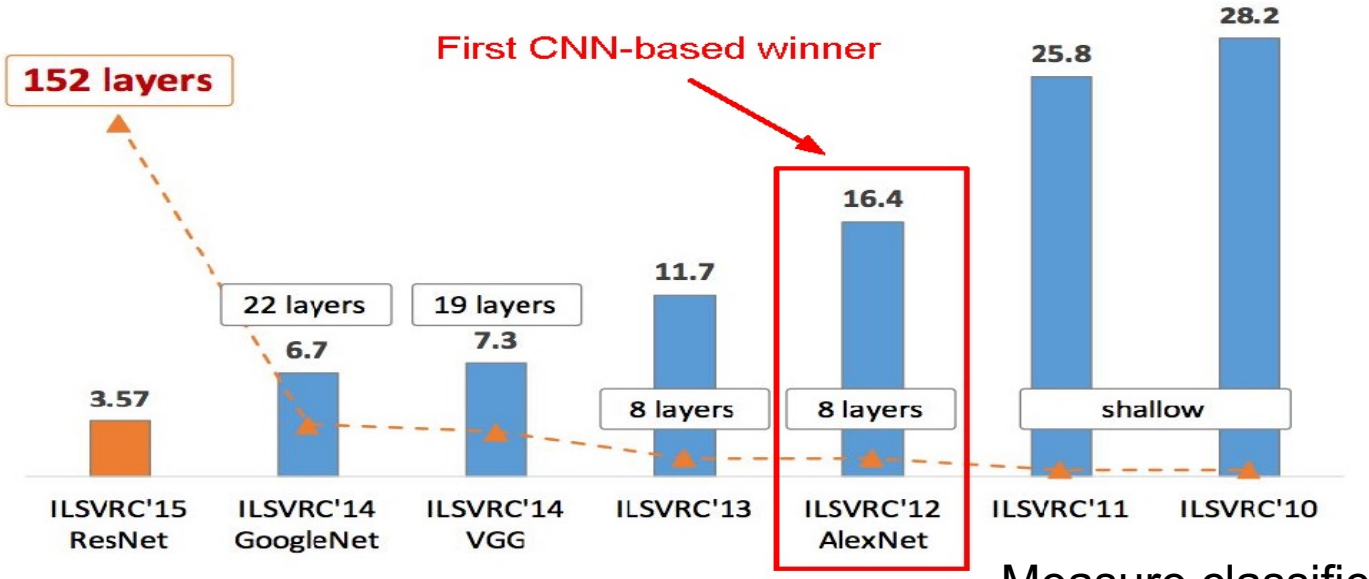
AlexNet

- *ImageNet Classification with Deep Convolutional Neural Networks* - Alex Krizhevsky, Ilya Sutskever, Geoffrey E. Hinton; 2012
- One of the largest CNN networks at that time
- There are 60M parameters compared to 60k parameters of LeNet-5 .

[Krizhevsky et al., 2012]

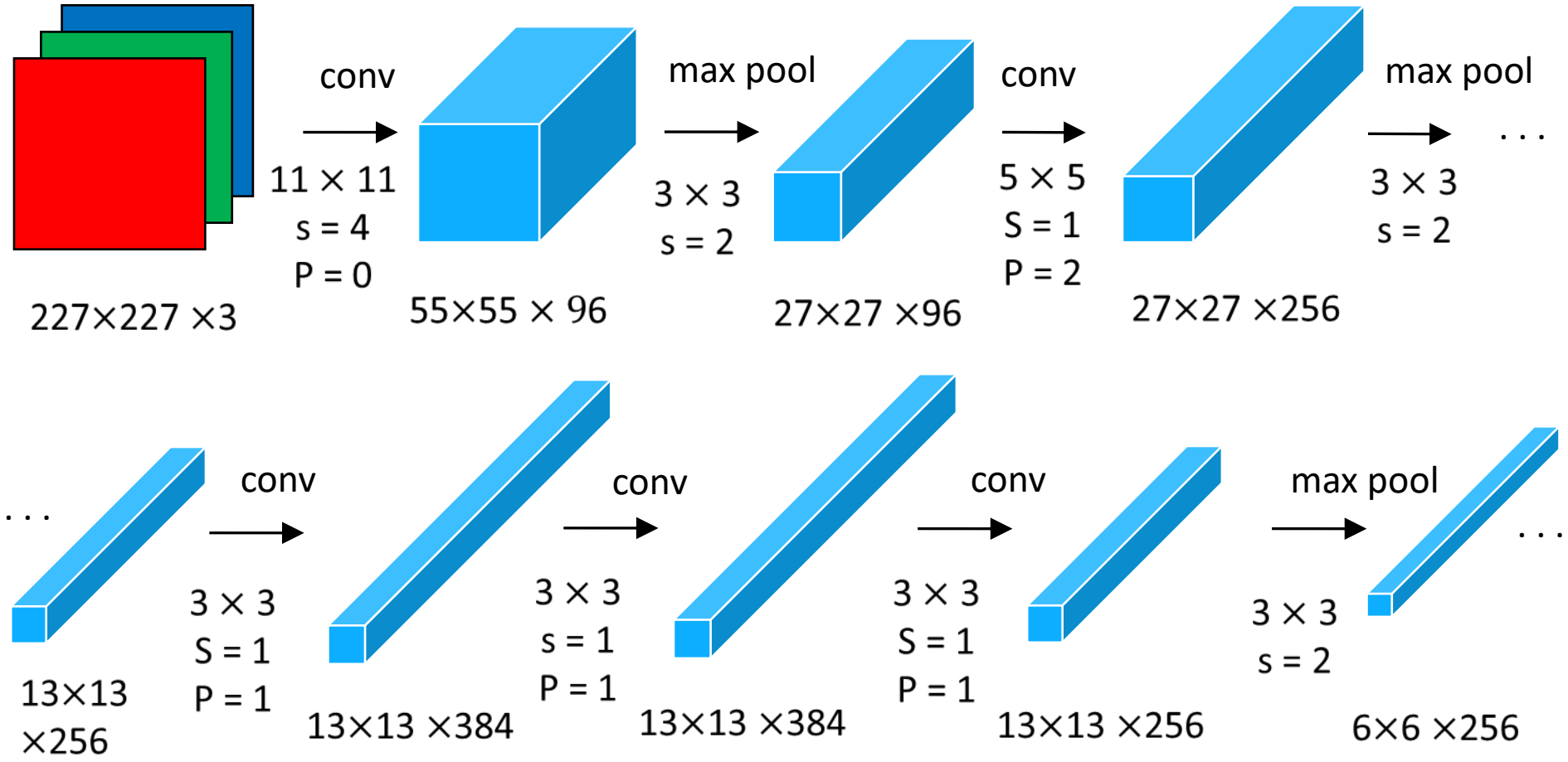
ImageNet Large Scale Visual Recognition Challenge (ILSVRC) winners

- Annual "Olympics" in the field of computer vision.
- Teams around the world compete against each other to see who has the best CV models for problems like image classification, positioning, and object detection in images.



Measure classification error

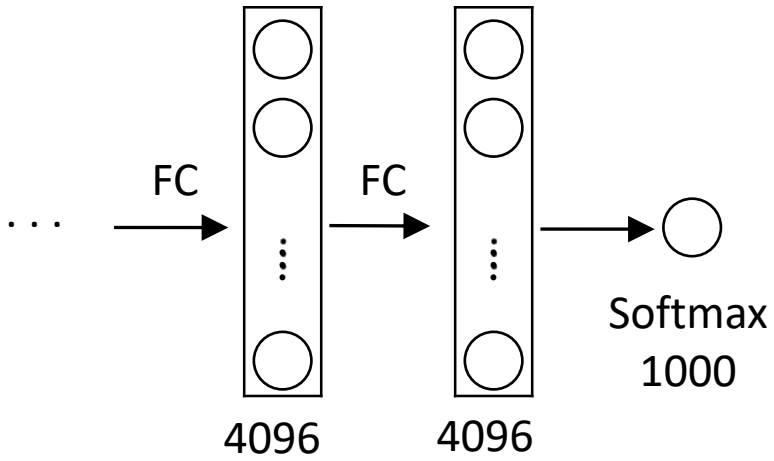
AlexNet



This slide is taken from Andrew Ng

[**Krizhevsky et al., 2012**]

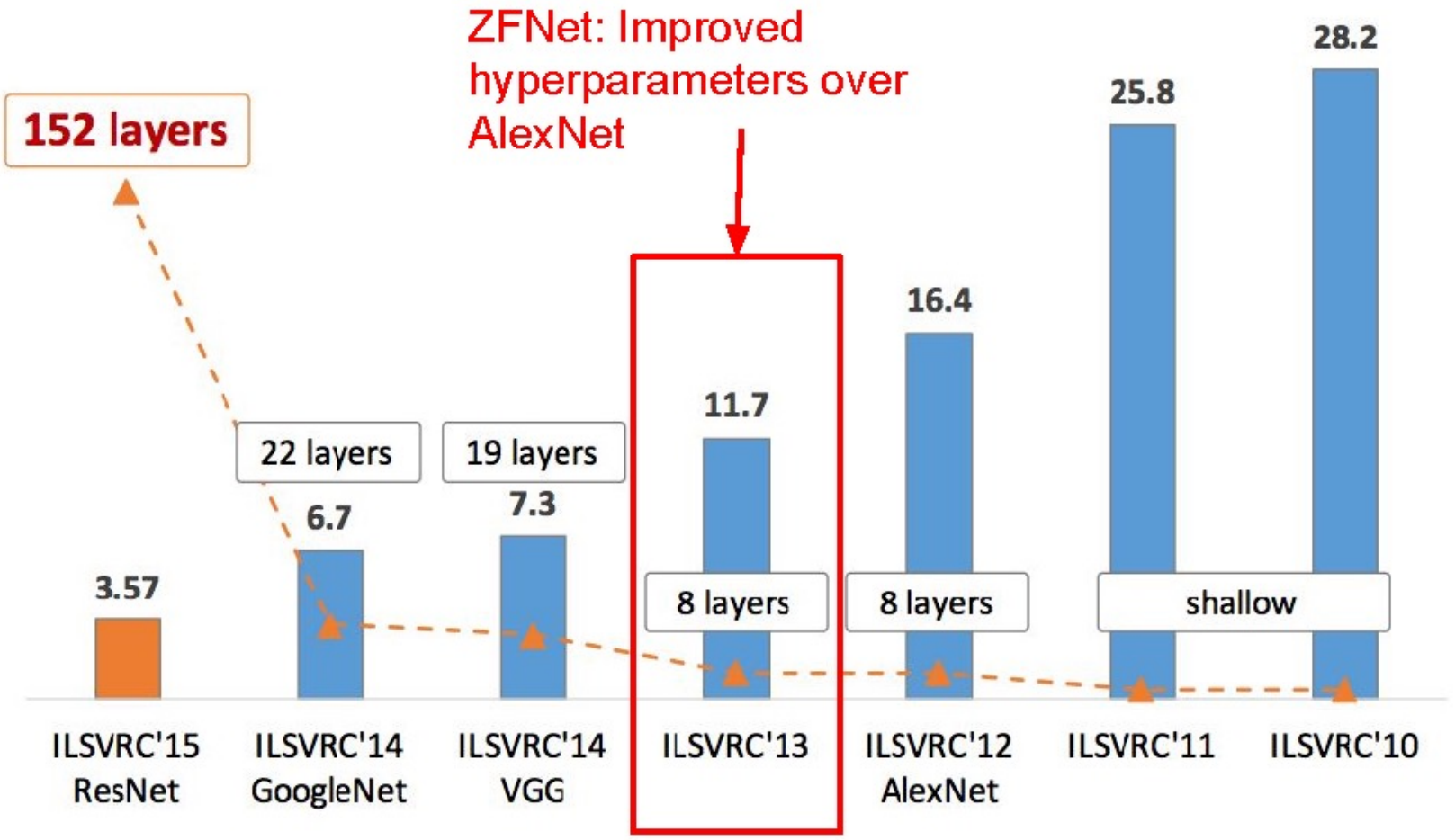
AlexNet



This slide is taken from Andrew Ng

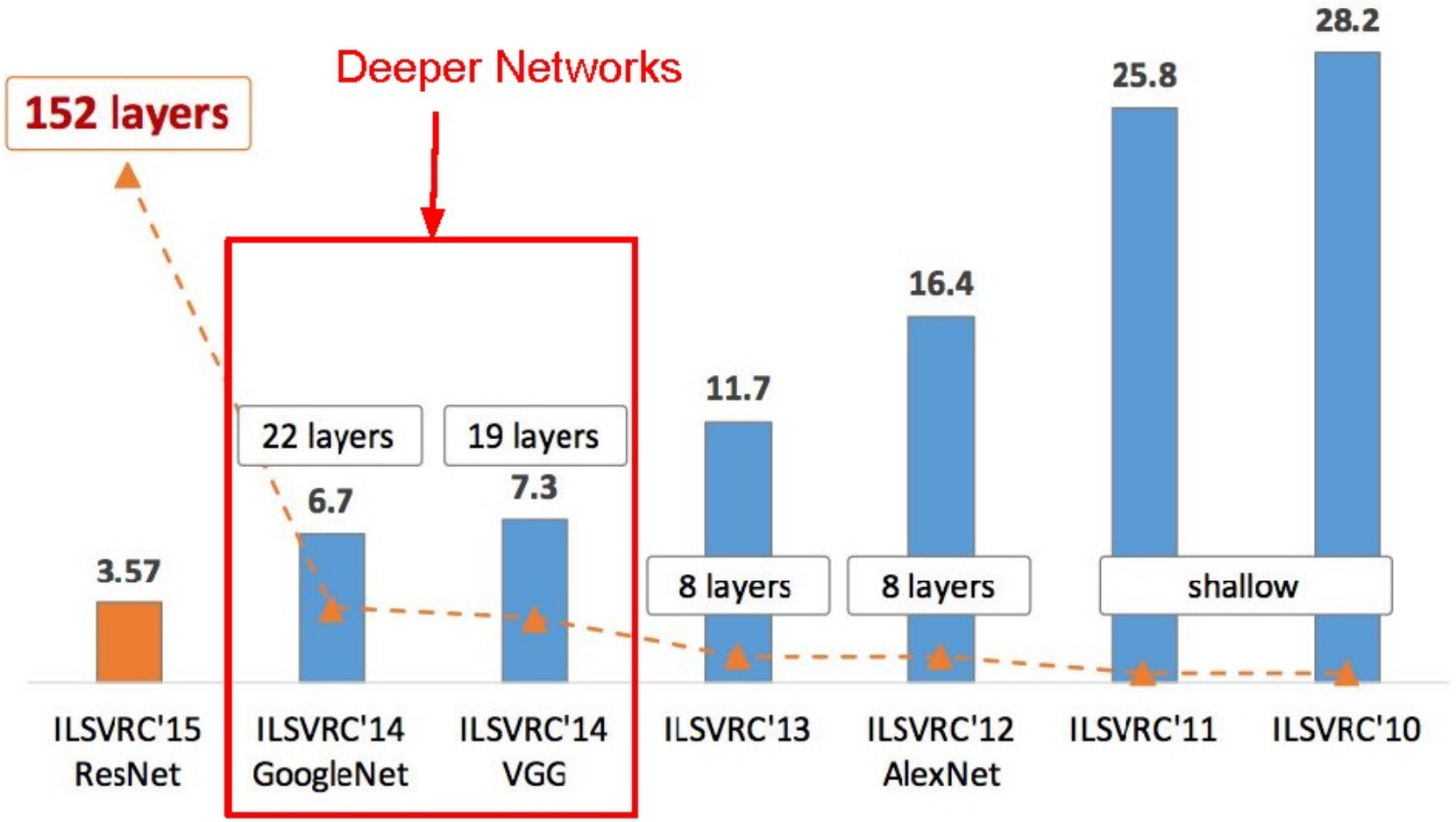
[**Krizhevsky et al., 2012**]

ImageNet Large Scale Visual Recognition Challenge (ILSVRC) winners



Slide taken from Fei-Fei & Justin Johnson & Serena Yeung. Lecture 9.

ImageNet Large Scale Visual Recognition Challenge (ILSVRC) winners



Slide taken from Fei-Fei & Justin Johnson & Serena Yeung. Lecture 9.

VGGNet

- *Very Deep Convolutional Networks For Large Scale Image Recognition* - Karen Simonyan and Andrew Zisserman; 2015
- Runner-up at ILSVRC 2014
- Much deeper than AlexNet
- 140 million parameters

[Simonyan and Zisserman, 2014]

Input
3x3 conv, 64
3x3 conv, 64
Pool 1/2
3x3 conv, 128
3x3 conv, 128
Pool 1/2
3x3 conv, 256
3x3 conv, 256
Pool 1/2
3x3 conv, 512
3x3 conv, 512
3x3 conv, 512
Pool 1/2
3x3 conv, 512
3x3 conv, 512
3x3 conv, 512
Pool 1/2
FC 4096
FC 4096
FC 1000
Softmax

VGGNet

- Small size filters
  Only conv 3x3, stride 1, pad 1
  và 2x2 MAX POOL , stride 2

- Deeper network
  AlexNet: 8 layers
  VGGNet: 16 - 19 layers

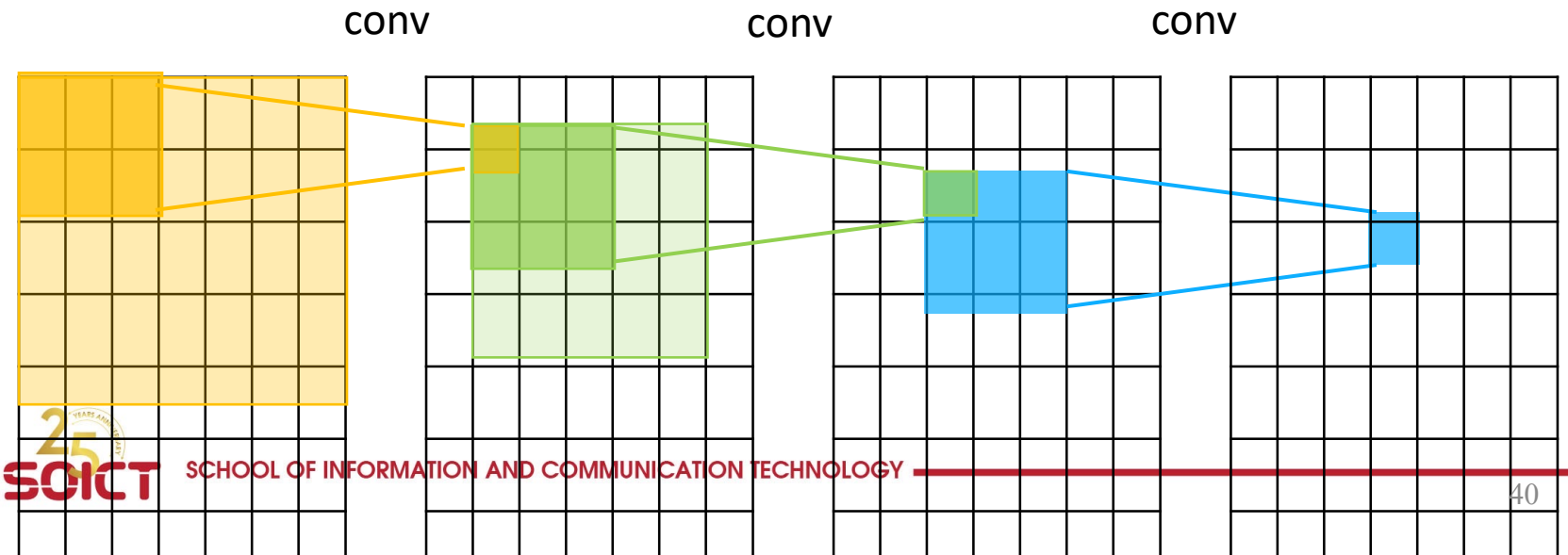
- ZFNet: 11.7% top 5 error in ILSVRC'13
- VGGNet: 7.3% top 5 error in ILSVRC'14

[Simonyan and Zisserman, 2014]

[Simonyan and Zisserman, 2014]

VGGNet

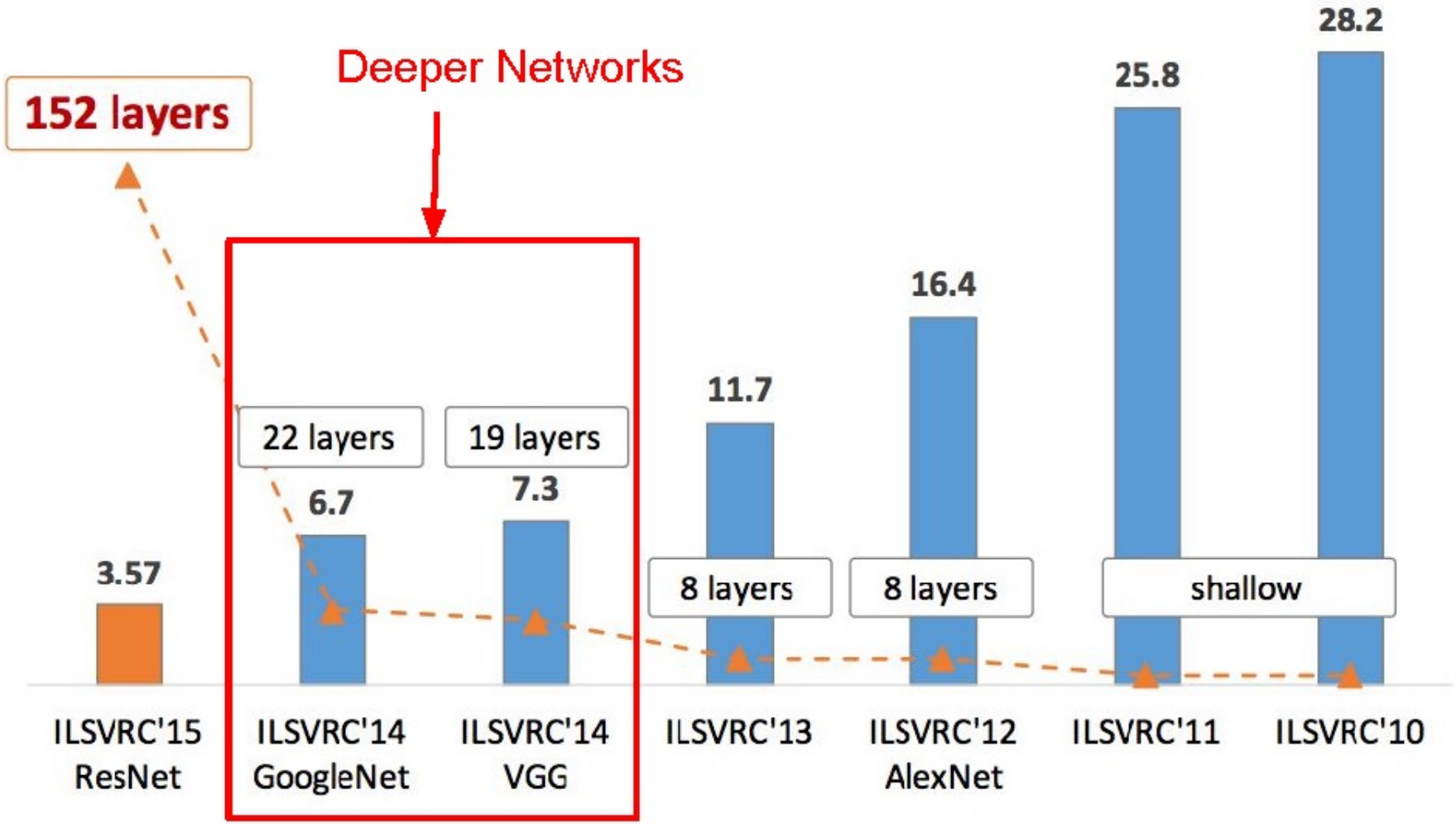
- Why use a small filter? (3x3 conv)
- A stack of 3 3x3 conv layers (stride 1) has the same output representation capacity as a 7x7 conv layer.
- But deeper, mean more nonlinear layers
- And less parameters: 3 * (3^2C^2) vs. 7^2C^2 where C is the number of channels of each layer



Input	memory: 224*224*3=150K	params: 0
3x3 conv, 64	memory: 224*224*64=3.2M	params: (3*3*3)*64 = 1,728
3x3 conv, 64	memory: 224*224*64=3.2M	params: (3*3*64)*64 = 36,864
Pool	memory: 112*112*64=800K	params: 0
3x3 conv, 128	memory: 112*112*128=1.6M	params: (3*3*64)*128 = 73,728
3x3 conv, 128	memory: 112*112*128=1.6M	params: (3*3*128)*128 = 147,456
Pool	memory: 56*56*128=400K	params: 0
3x3 conv, 256	memory: 56*56*256=800K	params: (3*3*128)*256 = 294,912
3x3 conv, 256	memory: 56*56*256=800K	params: (3*3*256)*256 = 589,824
3x3 conv, 256	memory: 56*56*256=800K	params: (3*3*256)*256 = 589,824
Pool	memory: 28*28*256=200K	params: 0
3x3 conv, 512	memory: 28*28*512=400K	params: (3*3*256)*512 = 1,179,648
3x3 conv, 512	memory: 28*28*512=400K	params: (3*3*512)*512 = 2,359,296
3x3 conv, 512	memory: 28*28*512=400K	params: (3*3*512)*512 = 2,359,296
Pool	memory: 14*14*512=100K	params: 0
3x3 conv, 512	memory: 14*14*512=100K	params: (3*3*512)*512 = 2,359,296
3x3 conv, 512	memory: 14*14*512=100K	params: (3*3*512)*512 = 2,359,296
3x3 conv, 512	memory: 14*14*512=100K	params: (3*3*512)*512 = 2,359,296
Pool	memory: 7*7*512=25K	params: 0
FC 4096	memory: 4096	params: 7*7*512*4096 = 102,760,448
FC 4096	memory: 4096	params: 4096*4096 = 16,777,216
FC 1000	memory: 1000	params: 4096*1000 = 4,096,000

[Simonyan and Zisserman, 2014]

ImageNet Large Scale Visual Recognition Challenge (ILSVRC) winners

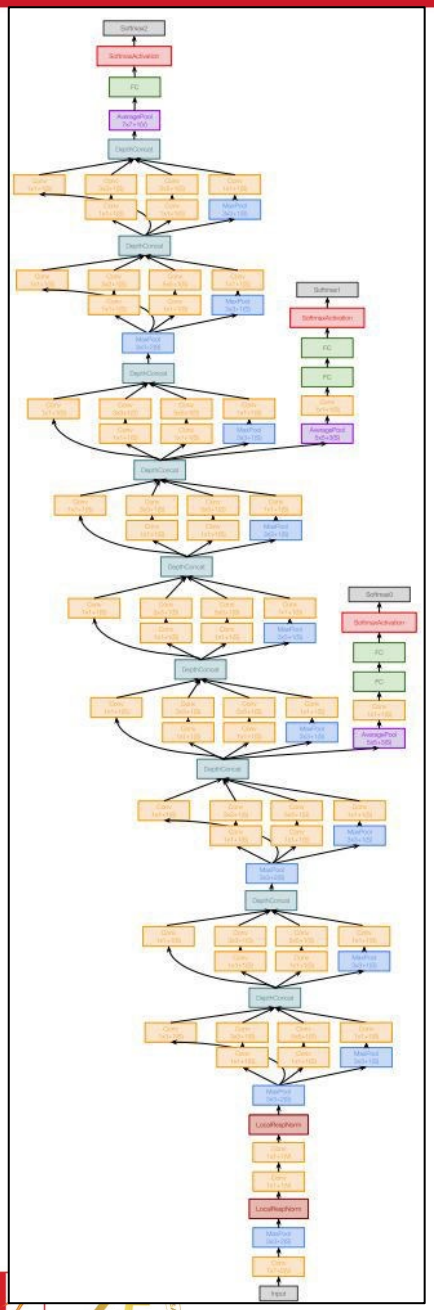


Slide taken from Fei-Fei & Justin Johnson & Serena Yeung. Lecture 9.

GoogleNet

- *Going Deeper with Convolutions* - Christian Szegedy et al.; 2015
- Winner of ILSVRC 2014
- Much deeper than AlexNet
- 12 times fewer parameters than AlexNet
- Focus on reducing computational complexity

[Szegedy et al., 2014]



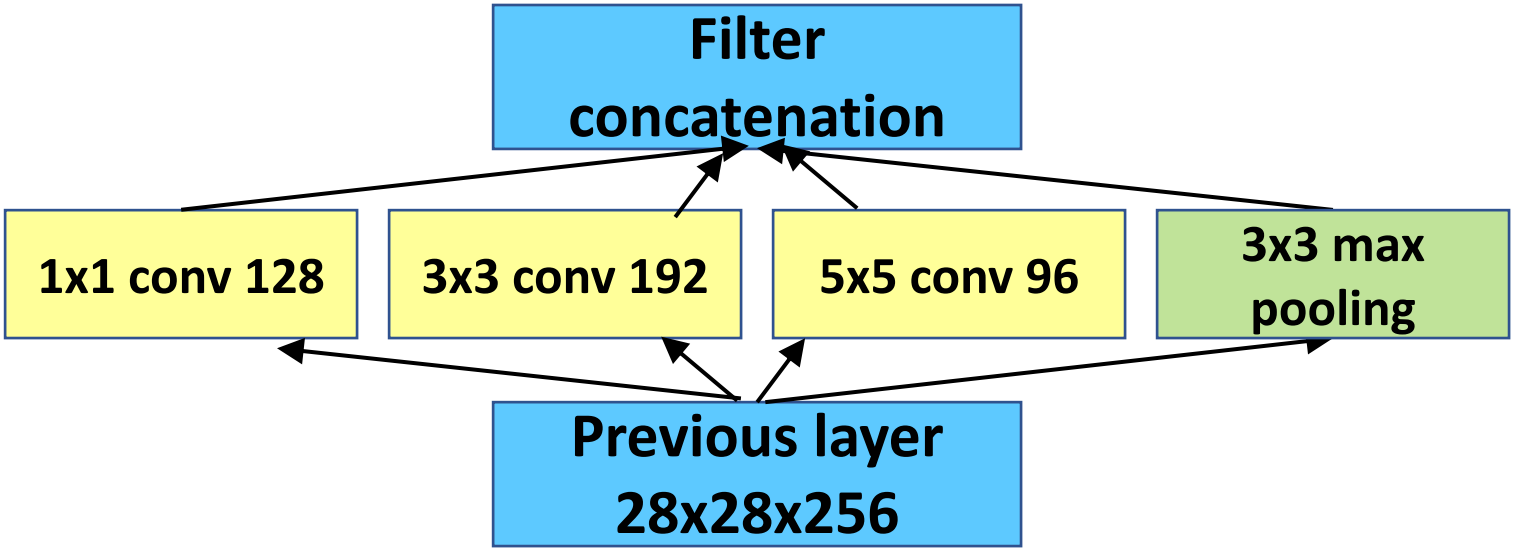
GoogleNet

- 22 layers
- Application of “Inception block”
- No fully connected layer
- Only 5 million parameters
- Champion of the ILSVRC'14 image classification task (6.7% top 5 error)

[Szegedy et al., 2014]

GoogleNet - Naïve Inception Model

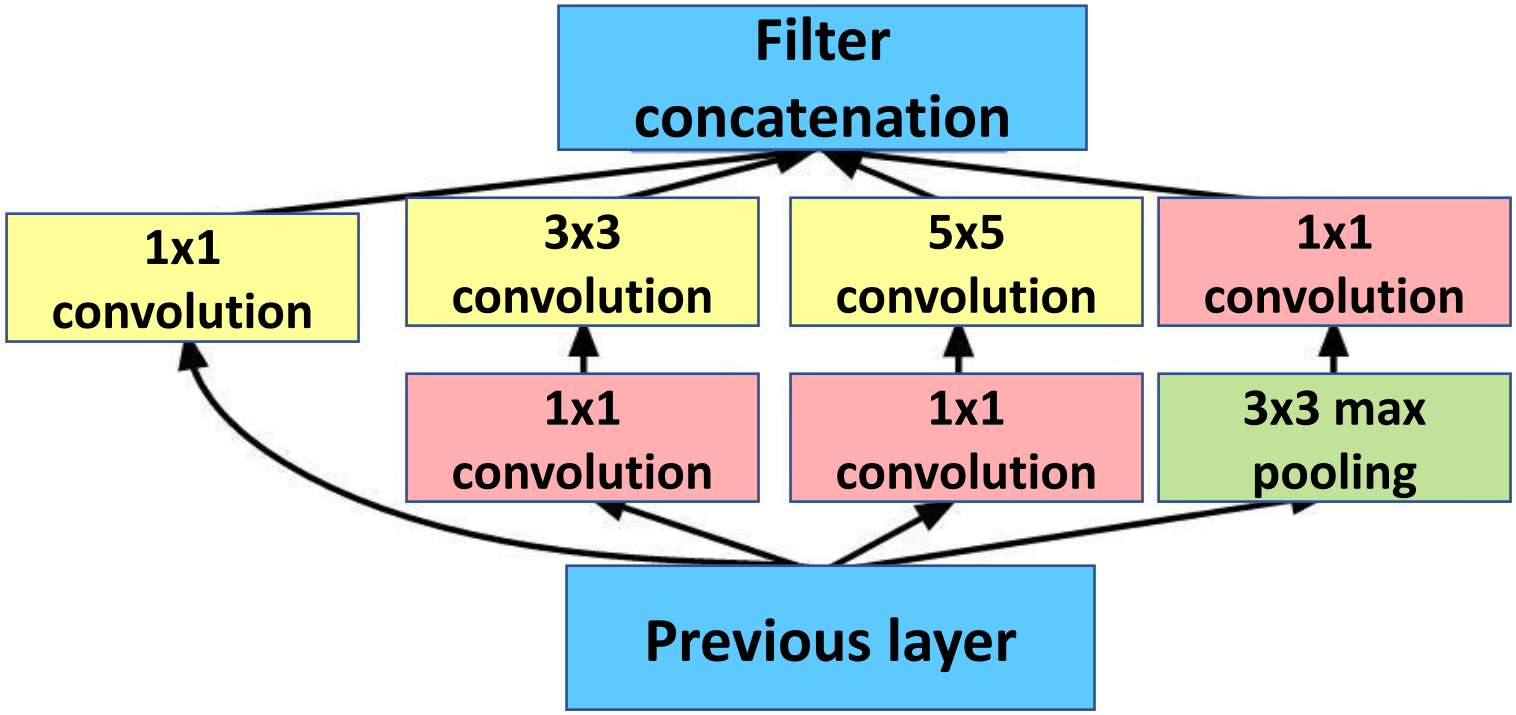
- Number of computations:
- 1x1 conv, 128: 28x28x128x1x1x256
- 3x3 conv, 192: 28x28x192x3x3x256
- 5x5 conv, 96: 28x28x96x5x5x256
- Total: 854M ops ==> **Heavy load!**



[Szegedy et al., 2014]

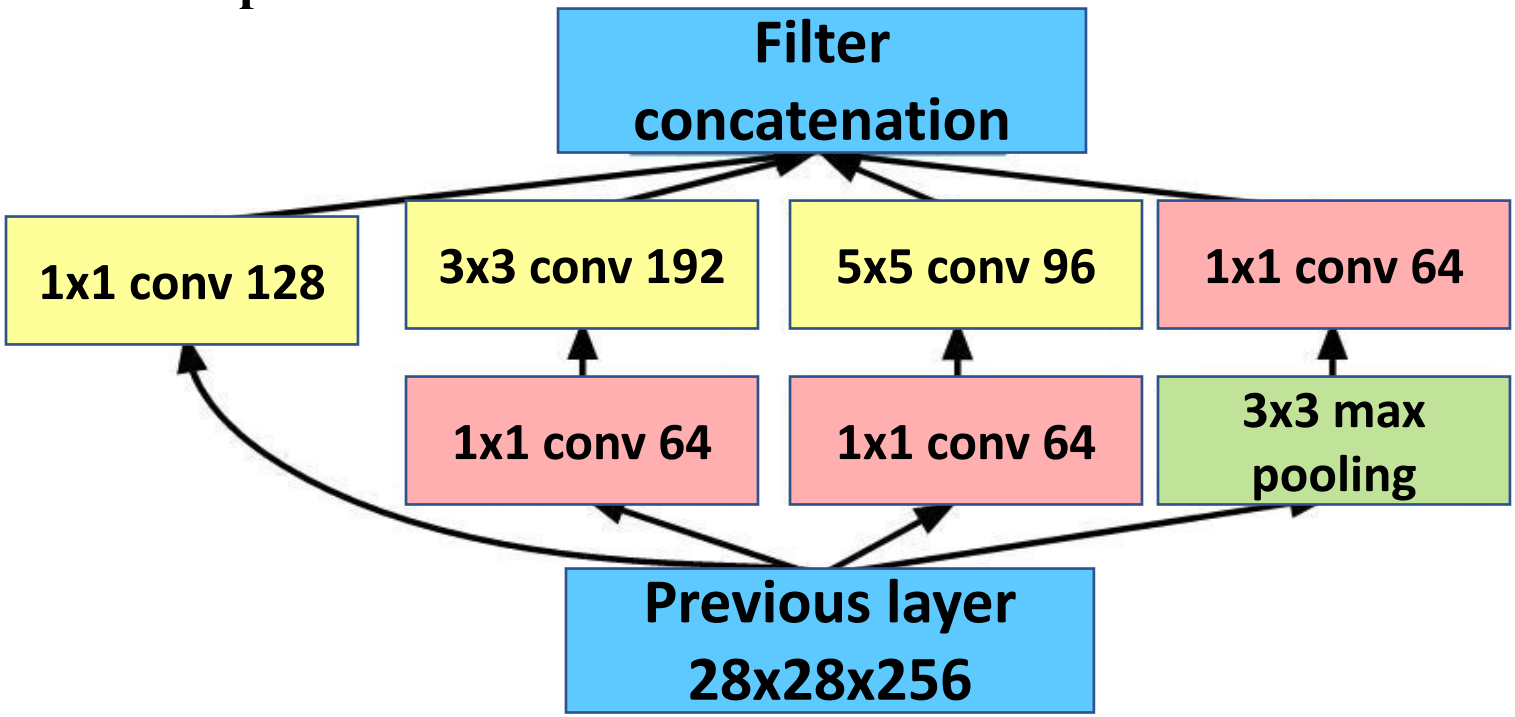
GoogleNet

- Solution: use "bottleneck" conv 1x1 to reduce the block depth.



[Szegedy et al., 2014]

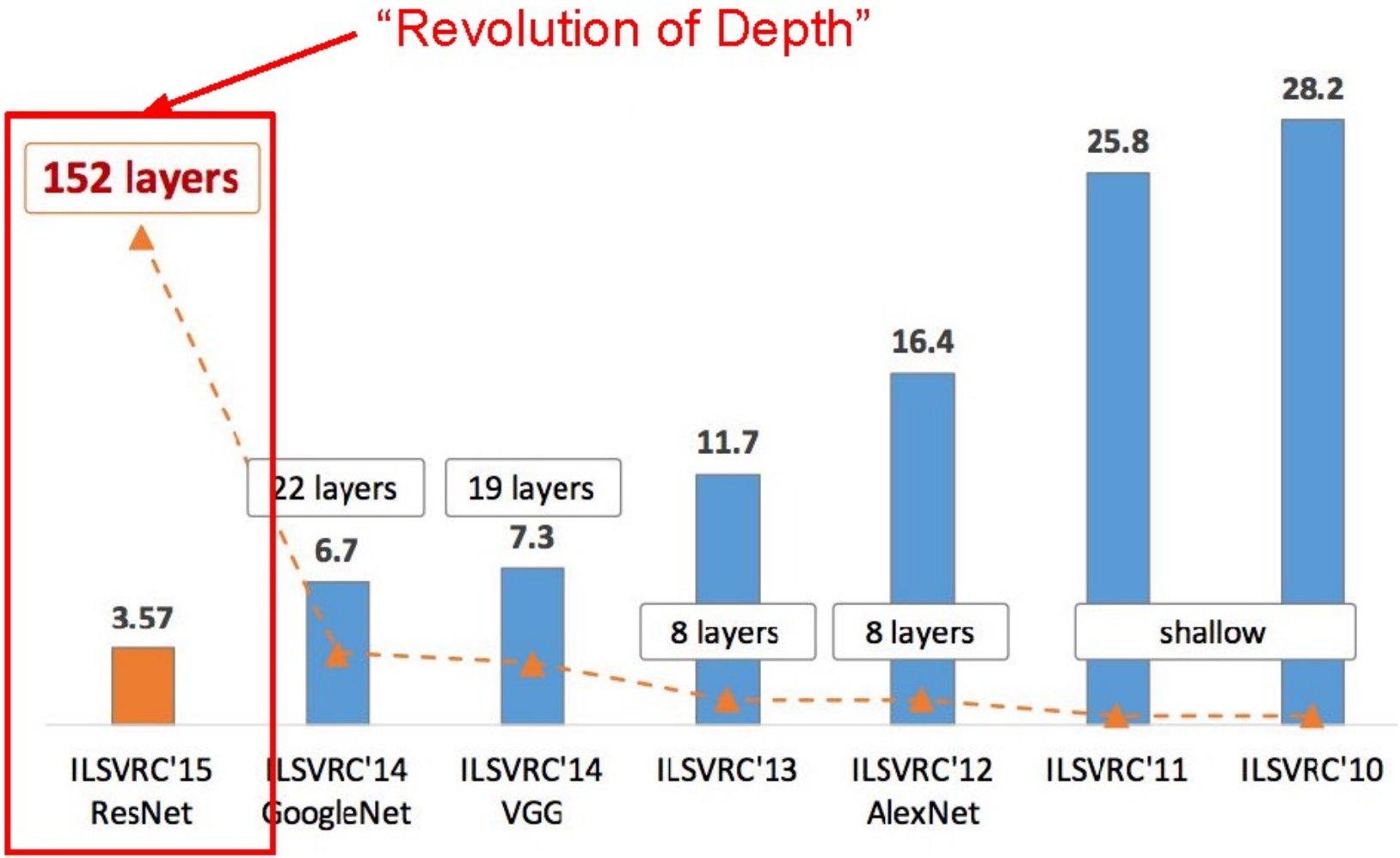
- **Number of computations:**
  1x1 conv, 64: 28x28x64x1x1x256
  1x1 conv, 64: 28x28x64x1x1x256
  1x1 conv, 128: 28x28x128x1x1x256
  3x3 conv, 192: 28x28x192x3x3x64
  5x5 conv, 96: 28x28x96x5x5x64
  1x1 conv, 64: 28x28x64x1x1x256
  Total: 353M ops



- Compared to 854M ops with a regular inception block

[Szegedy et al., 2014]

ImageNet Large Scale Visual Recognition Challenge (ILSVRC) winners

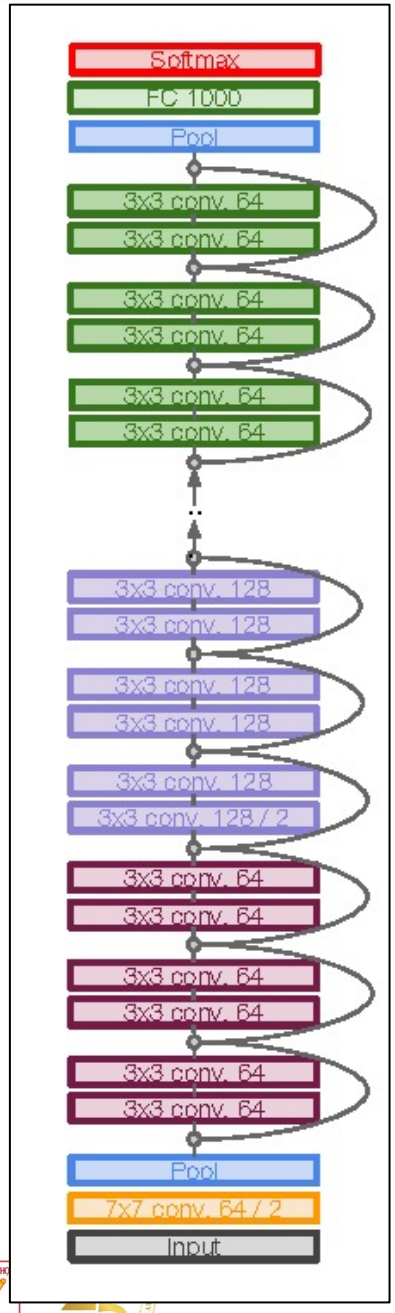


Slide taken from Fei-Fei & Justin Johnson & Serena Yeung. Lecture 9.

ResNet

- *Deep Residual Learning for Image Recognition* - Kaiming He, Xiangyu Zhang, Shaoqing Ren, Jian Sun; 2015.
- The network is very deep, up to 152 layers.
- The deeper the network, the harder it is to train.
- The deeper the network, the more affected the gradient explosion and vanishing problem.
- ResNet proposes a residual learning method that allows to effectively train networks much deeper than the ones that came before.

[He et al., 2015]



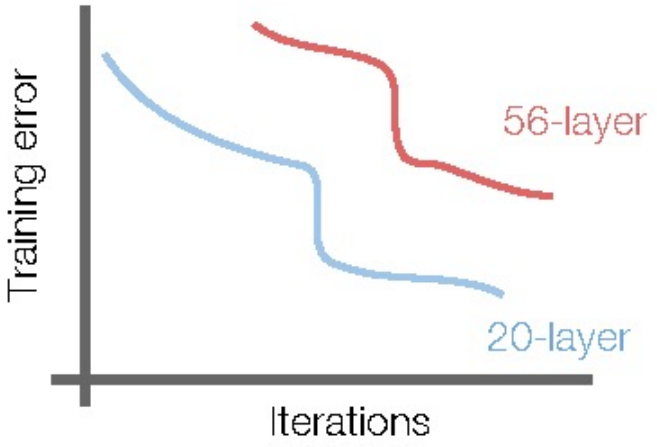
ResNet

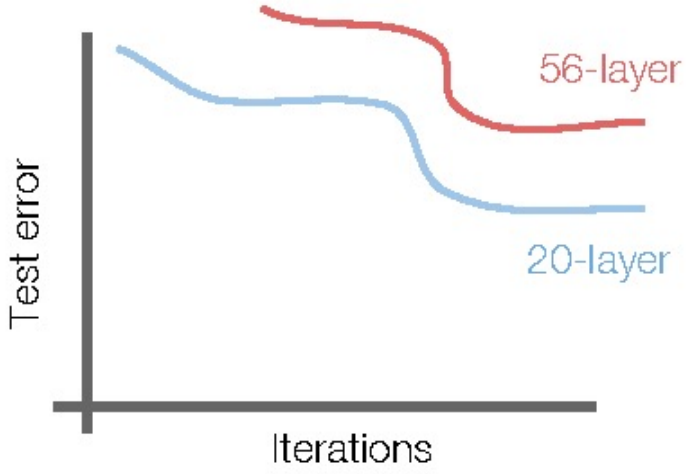
- Champion of the ILSVRC'15 classification task (3.57% top 5 error, while human error is about 5.1%)
- Sweep all image classification contests at ILSVRC'15 and COCO'15!

[He et al., 2015]

ResNet

- What happens when we increase the neural network depth?
- The 56-layer network performs worse on both the training and test sets (not caused by overfitting)
- Degeneracy phenomenon of deep networks

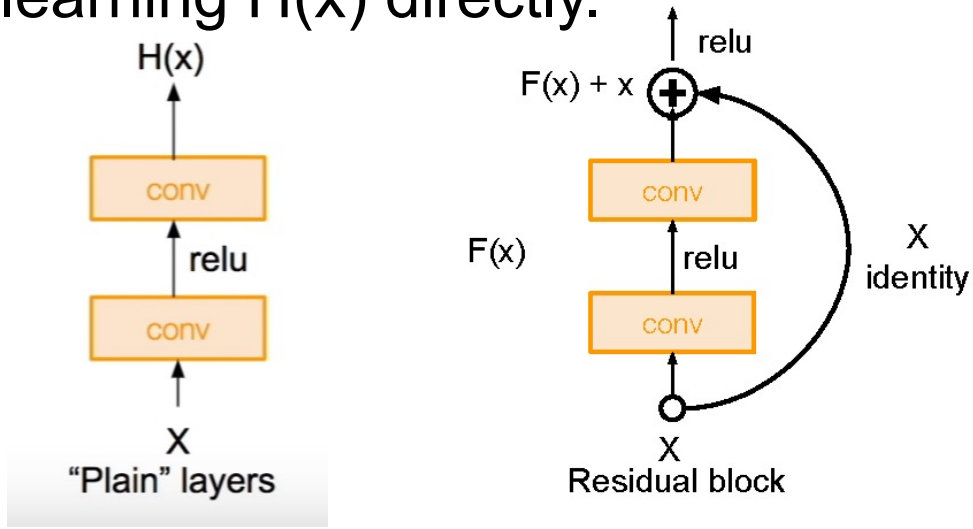




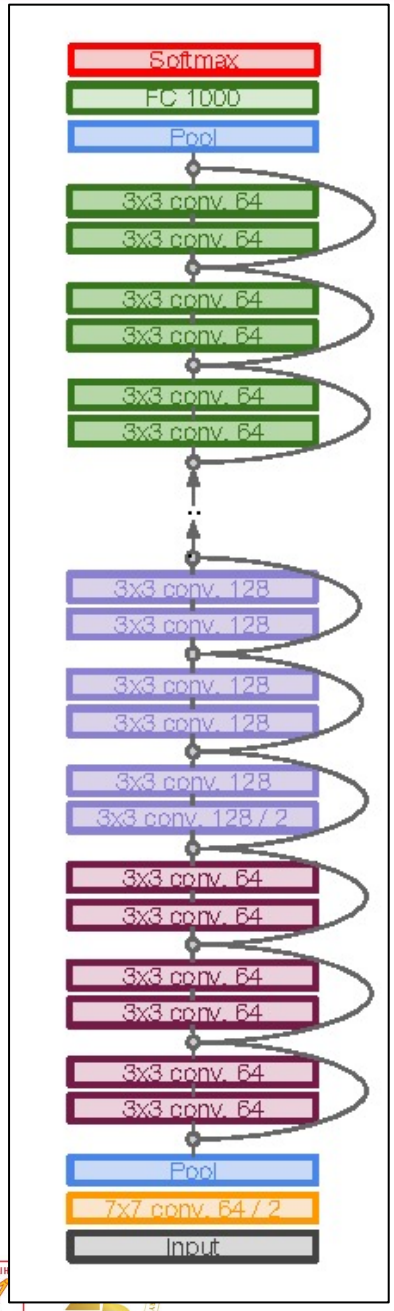
[He et al., 2015]

ResNet

- Assumption: The problem lies in the optimization problem. Very deep networks are harder to optimize.
- Solution: Let the network layers to learn residual representation (difference between output and input) instead of directly learning the output as before.
- Learn the residual representation F(x) = H(x) – x instead of learning H(x) directly.



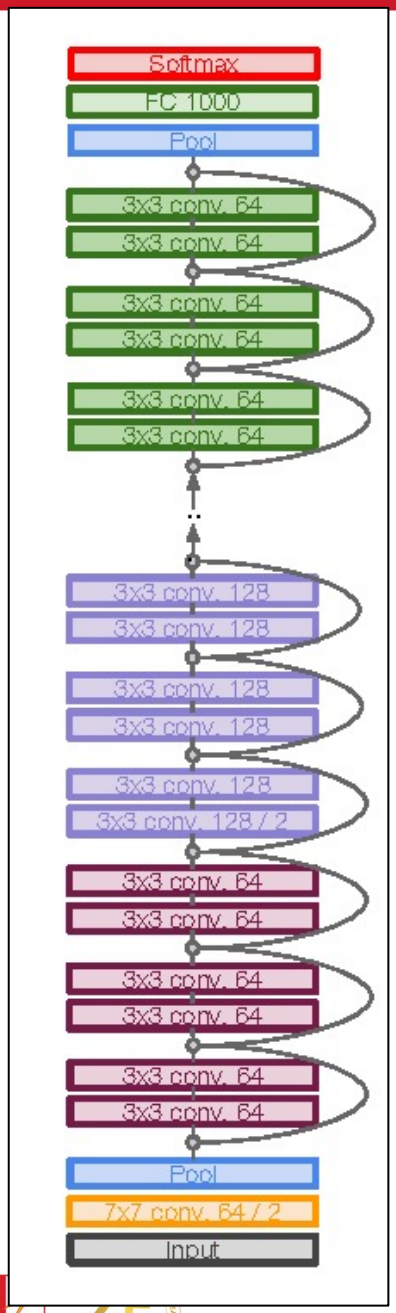
[He et al., 2015]



ResNet

- Full ResNet Architecture:
- Stack residual blocks
- Each block has two 3x3 conv layers
- Periodically double the number of convs and decrease the resolution by conv stride 2
- Sub-layer conv at the top of the network
- No FC layer at the end (only FC 1000 layer to output 1000-class classification result)

[He et al., 2015]

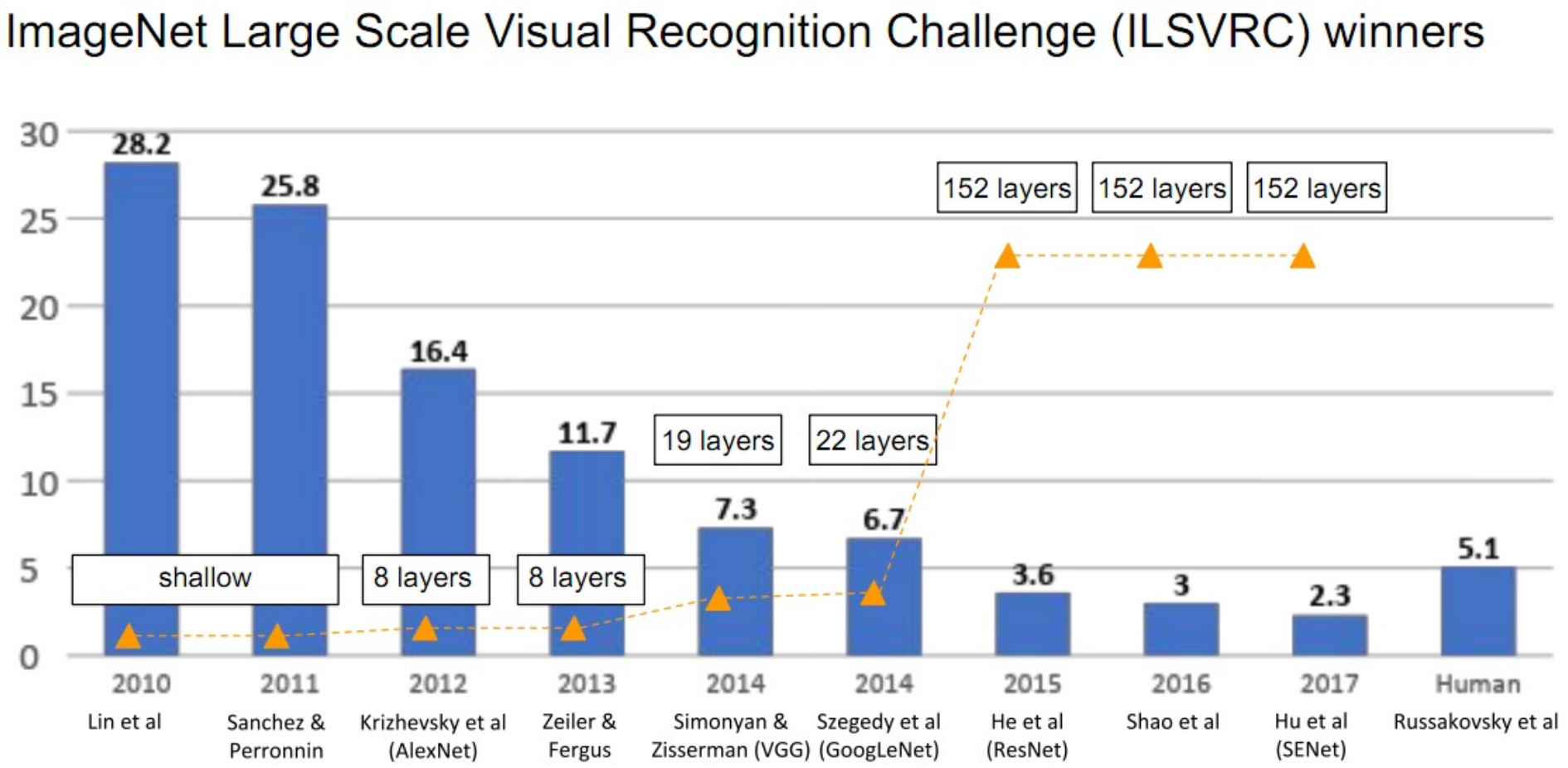


ResNet

- Network depths when participating in ImageNet competition: 34, 50, 101, 152
- For deep networks (ResNet-50+), the author uses the "bottleneck" layer to increase efficiency (similar to GoogLeNet).

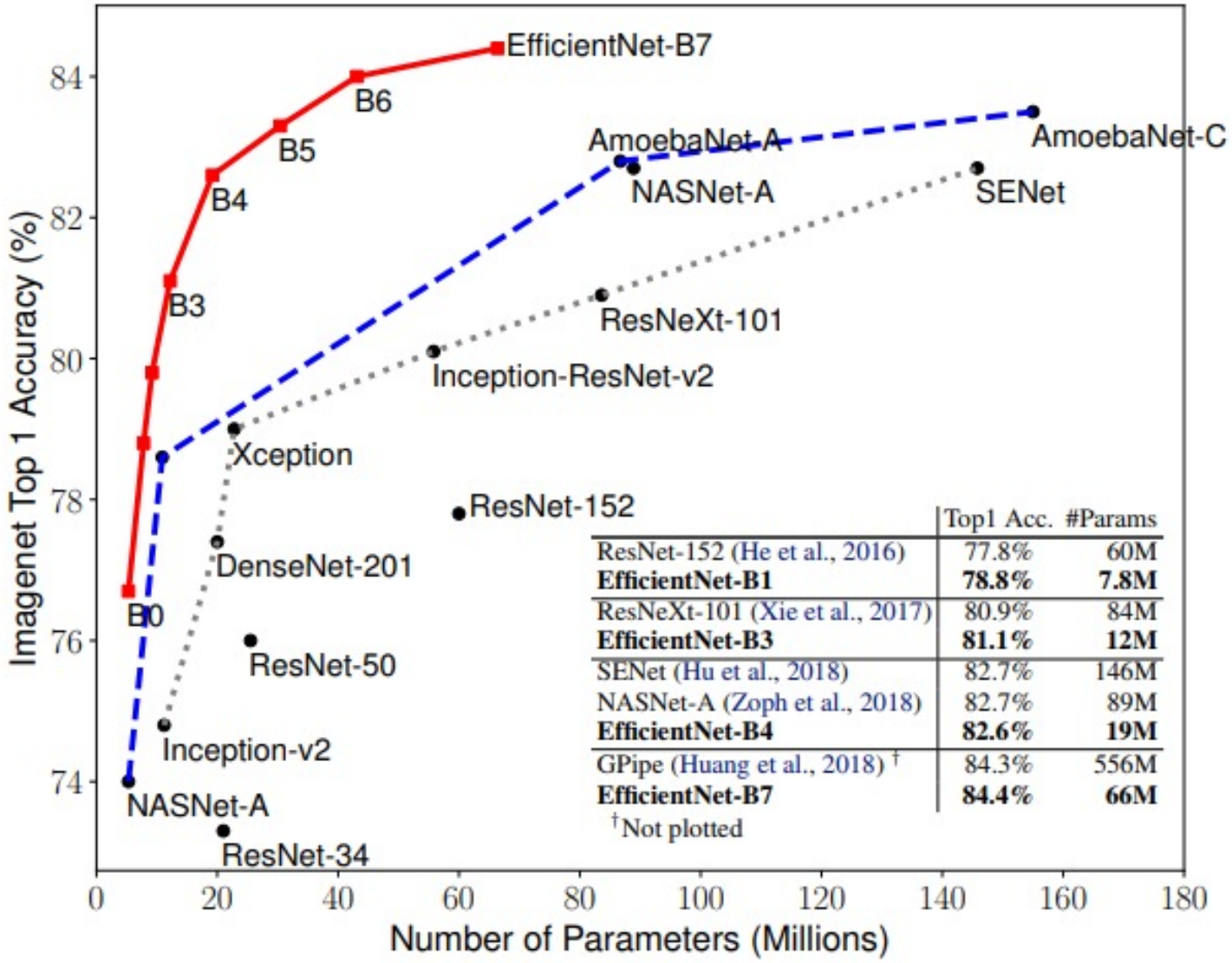
[He et al., 2015]

Recent SOTA



Slide taken from Fei-Fei & Justin Johnson & Serena Yeung. Lecture 9.

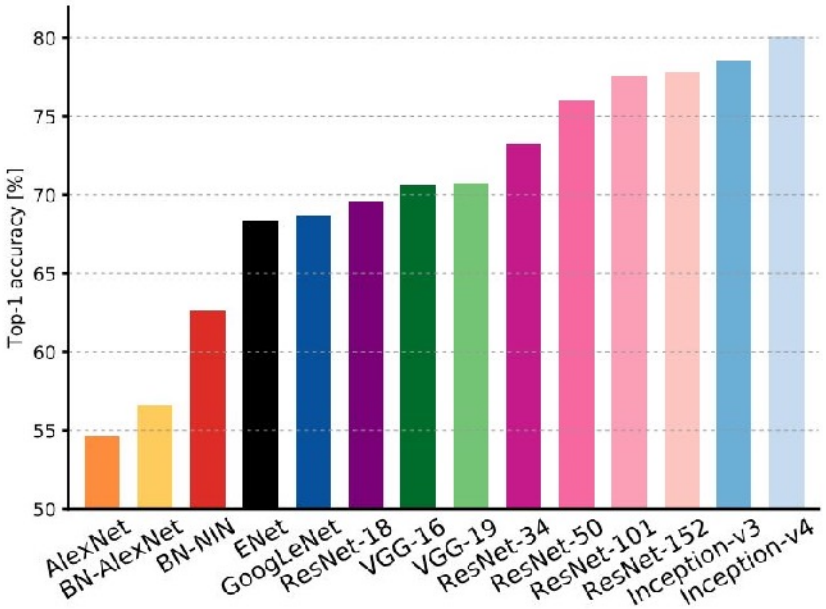
Recent SOTA

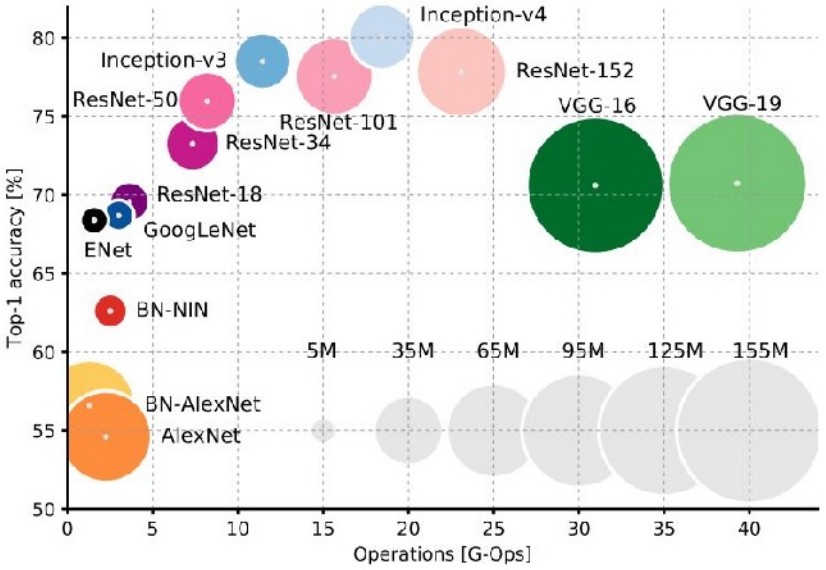


	Top1 Acc.	#Params
ResNet-152 (He et al., 2016)	77.8%	60M
EfficientNet-B1	**78.8%**	**7.8M**
ResNeXt-101 (Xie et al., 2017)	80.9%	84M
EfficientNet-B3	**81.1%**	**12M**
SENet (Hu et al., 2018)	82.7%	146M
NASNet-A (Zoph et al., 2018)	82.7%	89M
EfficientNet-B4	**82.6%**	**19M**
GPipe (Huang et al., 2018) †	84.3%	556M
EfficientNet-B7	**84.4%**	**66M**

†Not plotted

Accuracy comparison





References

1. http://introtodeeplearning.com/
2. http://cs231n.stanford.edu/

ĐẠI HỌC BÁCH KHOA

25 YEARS ANNIVERSARY SOICT

VIỆN CÔNG NGHỆ THÔNG TIN VÀ TRUYỀN THÔNG

SCHOOL OF INFORMATION AND COMMUNICATION TECHNOLOGY

Thank you for your attention!!!