CS60010: Deep Learning

Sudeshna Sarkar

Spring 2018

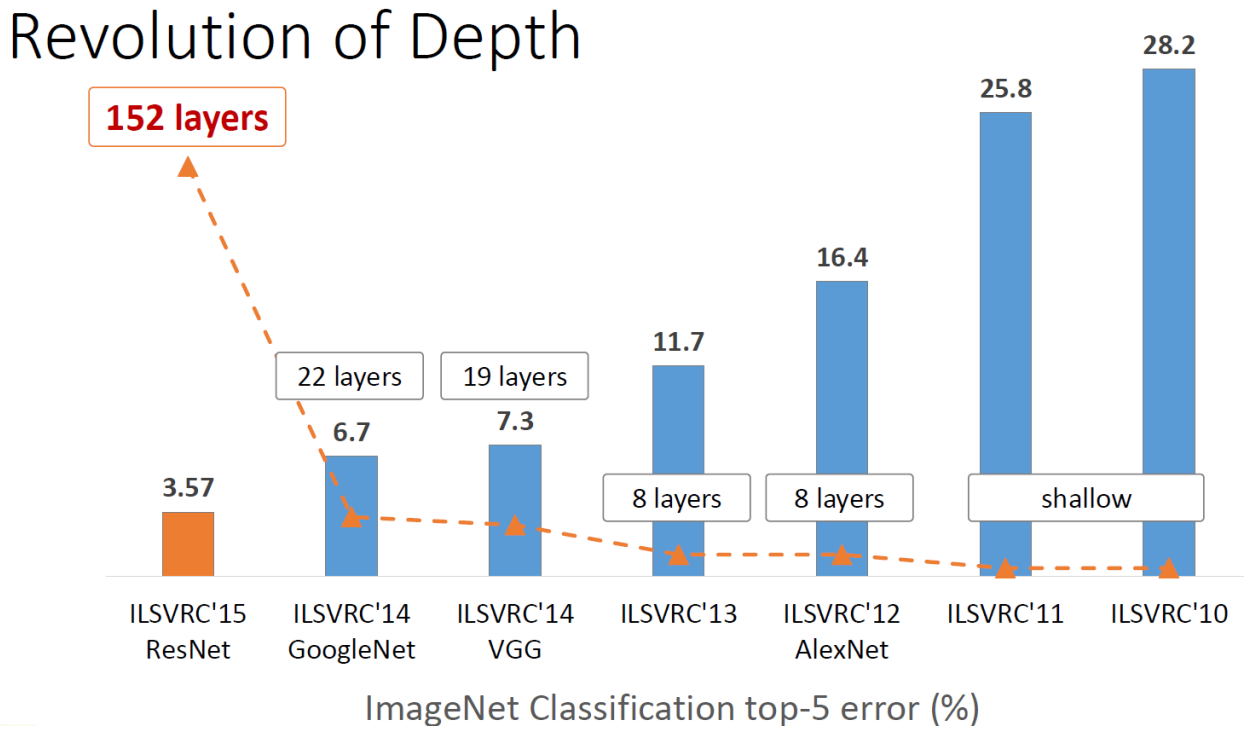
22 Jan 2018

Residual Net

- Deep Residual Learning for Image Recognition, Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun (CVPR 2016)

The deeper, the better?

- The deeper network can cover more complex problems
  - Receptive field size ↑
  - Non-linearity ↑
- However, training the deeper network is more difficult because of vanishing/exploding gradients problem

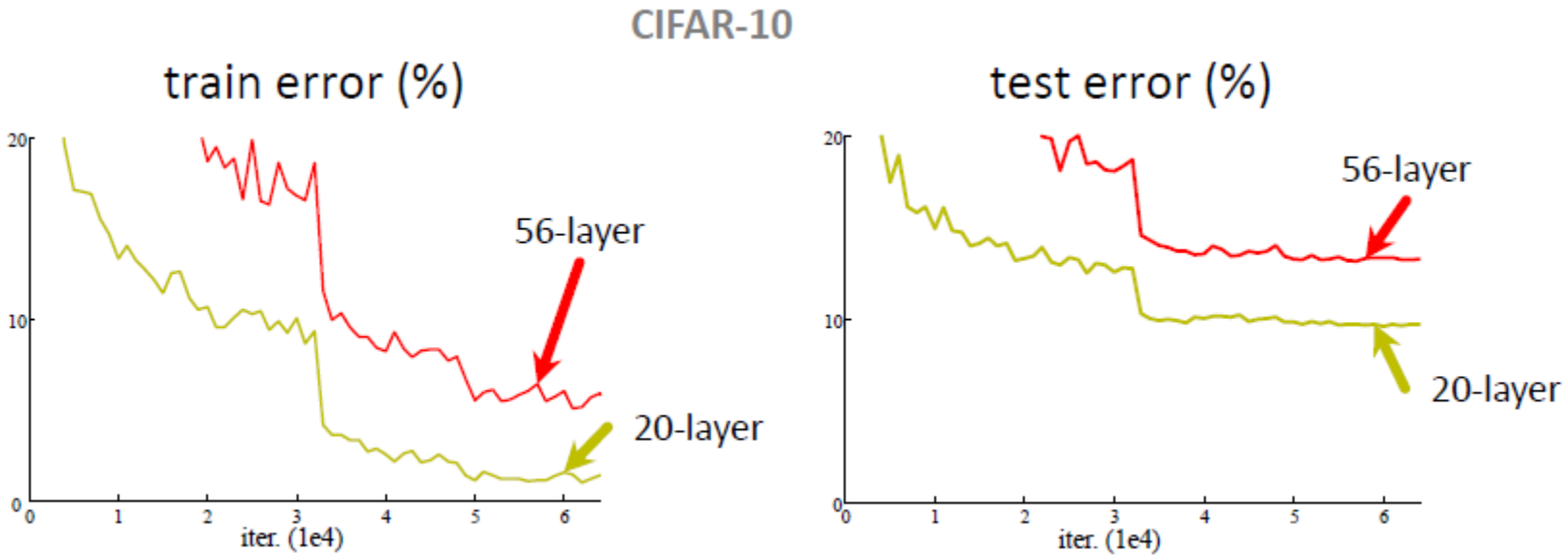


Deep NN

- Escape from few layers
  - ReLU for solving gradient vanishing problem
  - Dropout …
- Escape from 10 layers
  - Normalized initialization
  - Intermediate normalization layers
- Escape from 100 layers
  - Residual network

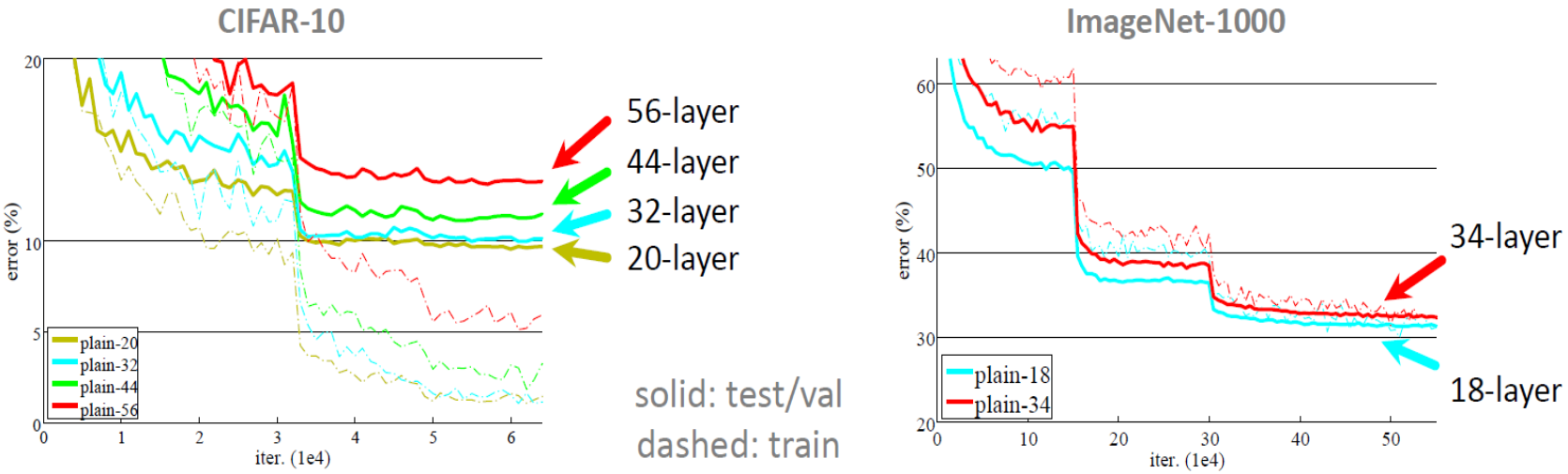
Plain Network

- Plain nets: stacking 3x3 conv layers
- 56-layer net has higher training error and test error than 20-layers net



Plain Network

- "Overly deep" plain nets have higher training error
- A general phenomenon, observed in many datasets

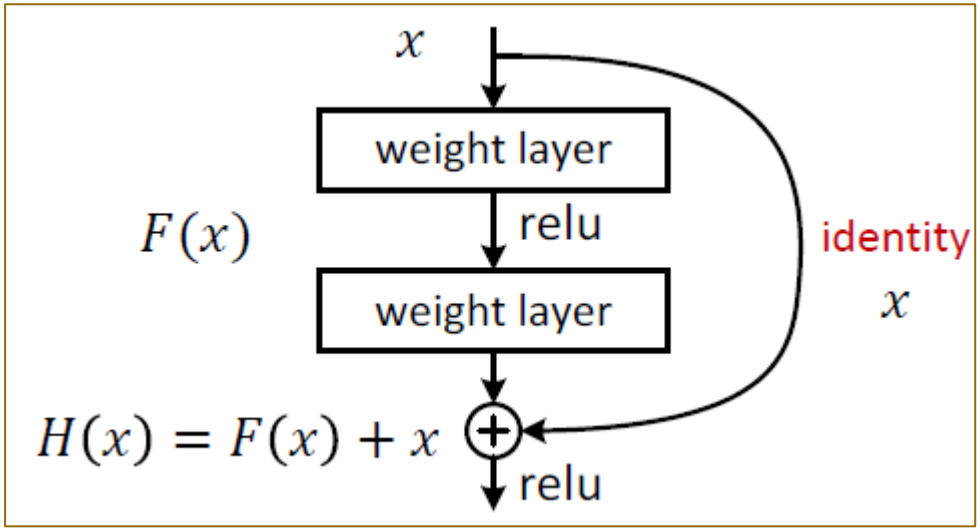


Residual Network

The residual module

Introduce *skip* or *shortcut* connections

Make it easy for network layers to represent the identity mapping



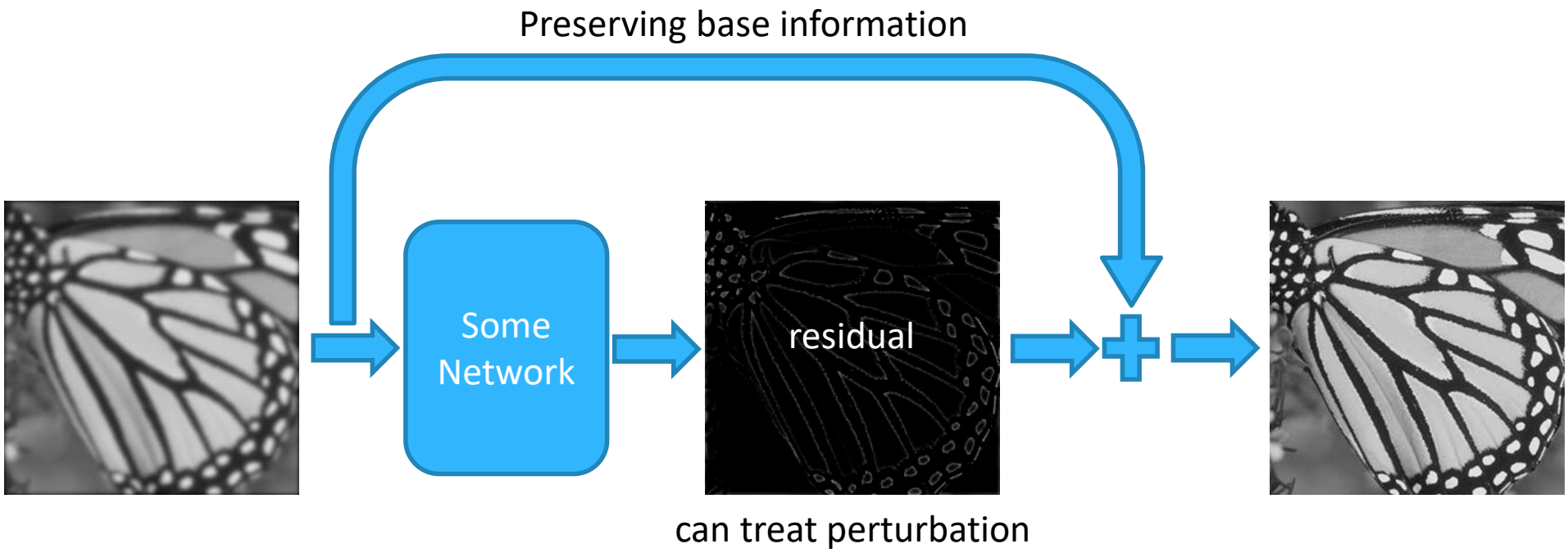
Residual block

- If identity were optimal, easy to set weights as 0
- If optimal mapping is closer to identity, easier to find small fluctuations

-> Appropriate for treating perturbation as keeping a base information

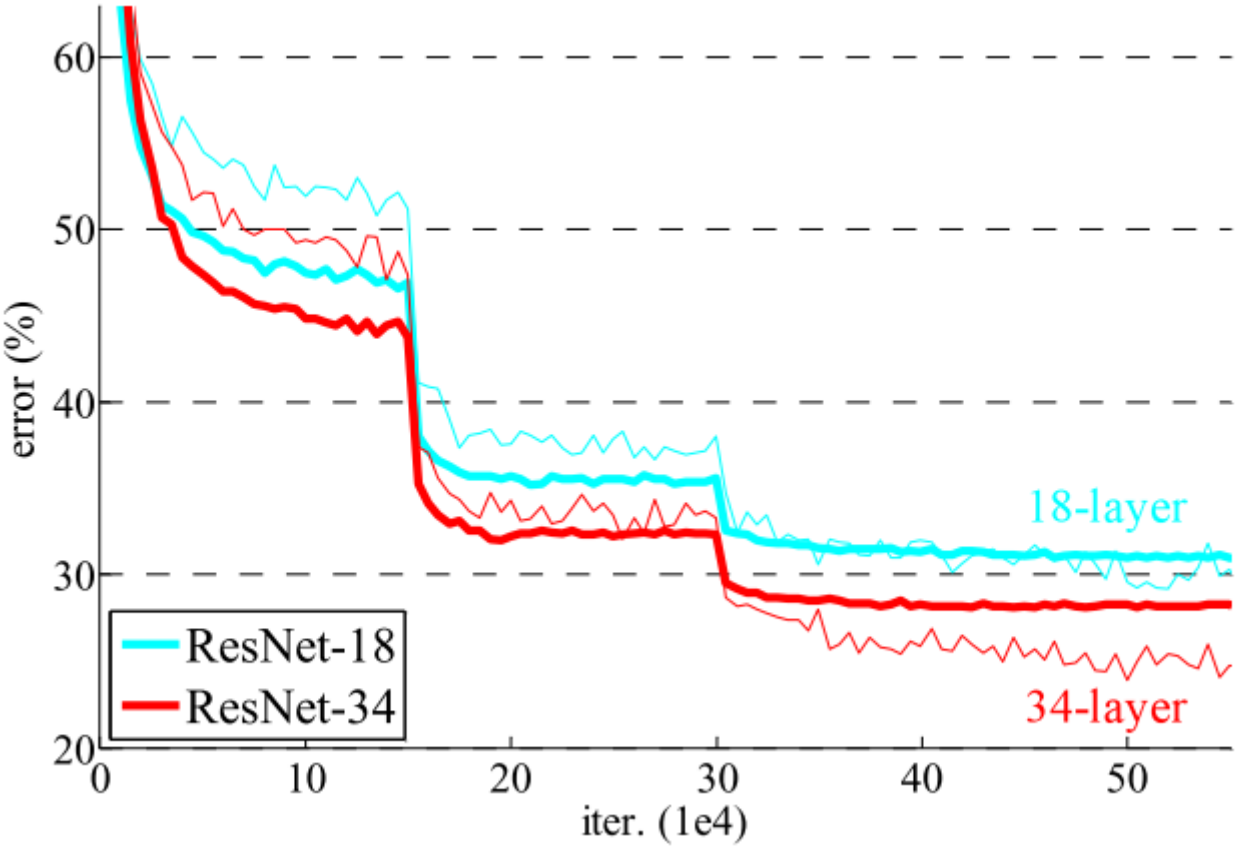
Residual Network

- Difference between an original image and a changed image



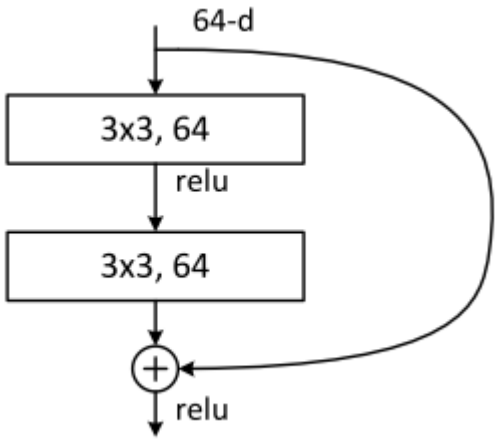
Residual Network

- Deeper ResNets have lower training error

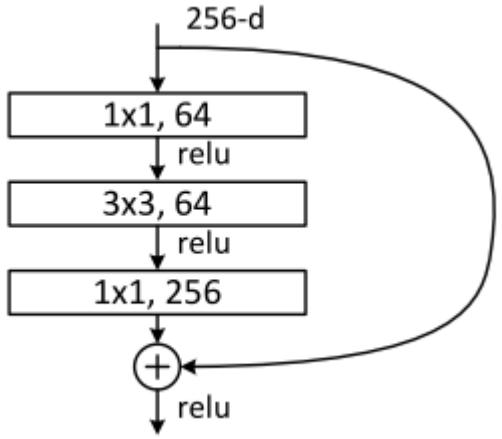


Residual Network

- Residual block
  - Very simple
  - Parameter-free



A naïve residual block



"bottleneck" residual block

Result

- Performances increase absolutely

task	2nd-place winner	MSRA	margin (relative)
ImageNet Localization (top-5 error)	12.0	9.0	**27%**
ImageNet Detection (mAP@.5)	53.6	62.1	**16%**
COCO Detection (mAP@.5:.95)	33.5	37.3	**11%**
COCO Segmentation (mAP@.5:.95)	25.1	28.2	**12%**

absolute 8.5% better!

- Based on ResNet-101
- Existing techniques can use residual networks or features from it

Summary: ILSVRC 2012-2015

Team	Year	Place	Error (top-5)	External data
SuperVision – Toronto (AlexNet, 7 layers)	2012	-	16.4%	no
SuperVision	2012	1st	15.3%	ImageNet 22k
Clarifai – NYU (7 layers)	2013	-	11.7%	no
Clarifai	2013	1st	11.2%	ImageNet 22k
VGG – Oxford (16 layers)	2014	2nd	7.32%	no
GoogLeNet (19 layers)	2014	1st	6.67%	no
ResNet (152 layers)	2015	1st	3.57%	
Human expert*			5.1%	

http://karpathy.github.io/2014/09/02/what-i-learned-from-competing-against-a-convnet-on-imagenet/

Design principles

- Reduce filter sizes (except possibly at the lowest layer), factorize filters aggressively
- Use 1x1 convolutions to reduce and expand the number of feature maps judiciously
- Use skip connections and/or create multiple paths through the network

Reading list

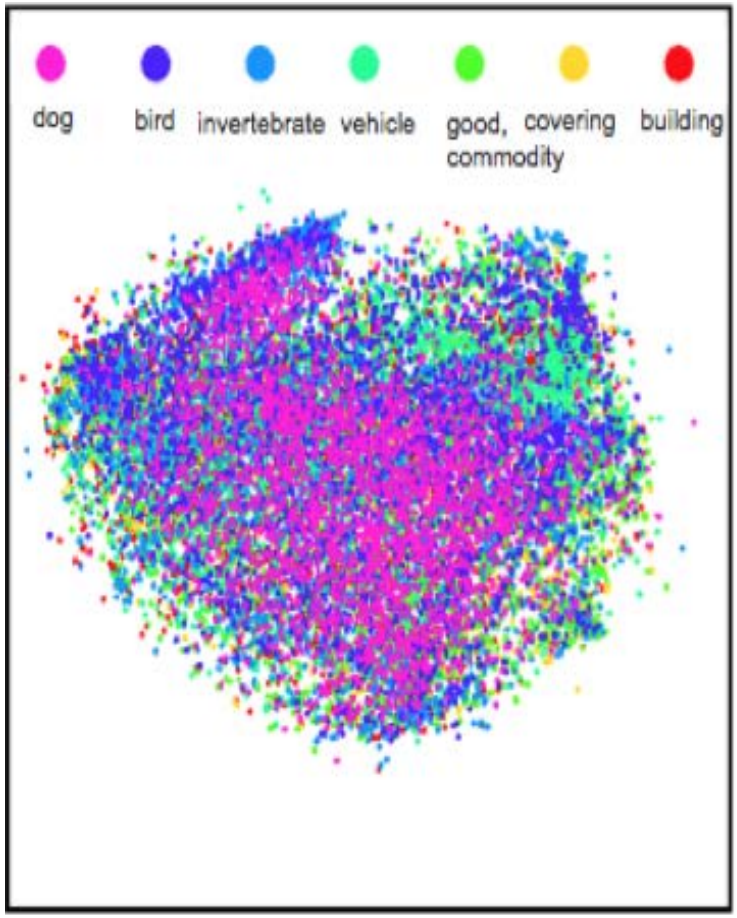
- https://culurciello.github.io/tech/2016/06/04/nets.html
- Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, Gradient-based learning applied to document recognition, Proc. IEEE 86(11): 2278–2324, 1998.
- A. Krizhevsky, I. Sutskever, and G. Hinton, ImageNet Classification with Deep Convolutional Neural Networks, NIPS 2012
- M. Zeiler and R. Fergus, Visualizing and Understanding Convolutional Networks, ECCV 2014
- K. Simonyan and A. Zisserman, Very Deep Convolutional Networks for Large-Scale Image Recognition, ICLR 2015
- M. Lin, Q. Chen, and S. Yan, Network in network, ICLR 2014
- C. Szegedy et al., Going deeper with convolutions, CVPR 2015
- C. Szegedy et al., Rethinking the inception architecture for computer vision, CVPR 2016
- K. He, X. Zhang, S. Ren, and J. Sun, Deep Residual Learning for Image Recognition, CVPR 2016

"You need a lot of a data if you want to train/use CNNs"

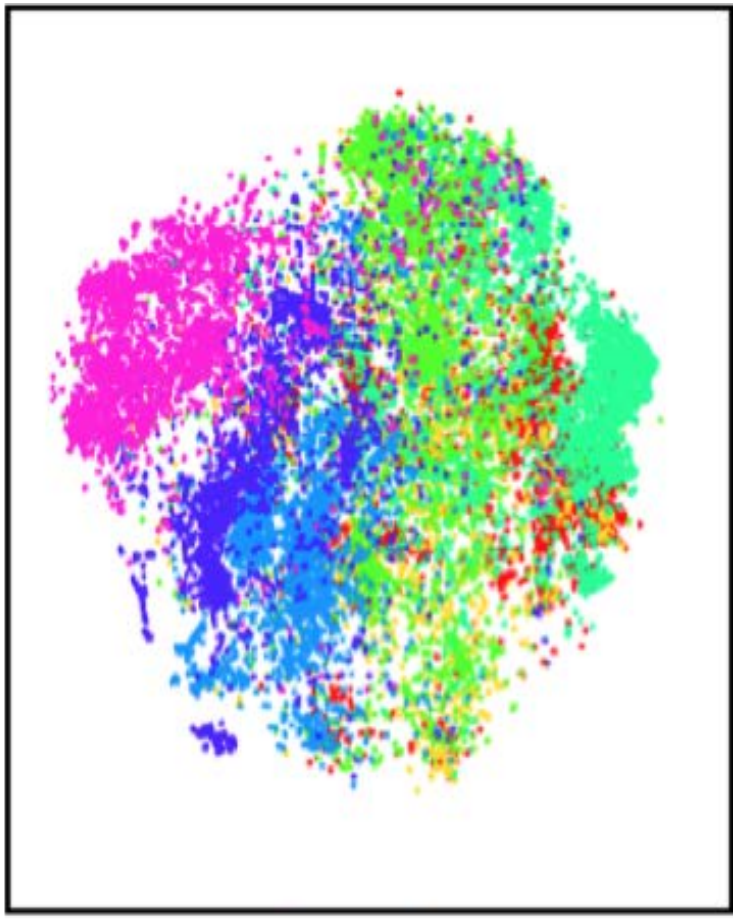
Transfer Learning

"You need a lot of a data if you want to train/use CNNs"

The Unreasonable Effectiveness of Deep Features



Low-level: $Pool_1$

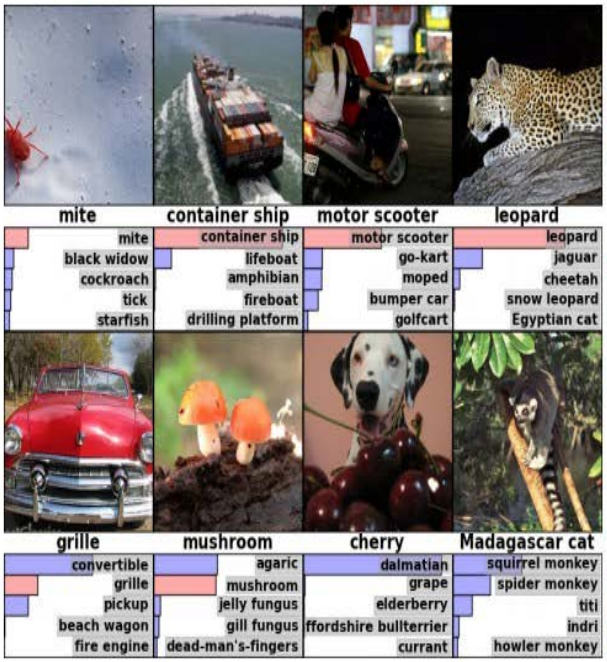


High-level: FC_6

Classes separate in the deep representations and transfer to many tasks.
[DeCAF] [Zeiler-Fergus]

Can be used as a generic feature

(“CNN code” = 4096-D vector before classifier)



query image

nearest neighbors in the “code” space

Can be used as a generic feature

("CNN code" = 4096-D vector before classifier)

query image

nearest neighbors in the "code" space

Transfer Learning with CNNs

1. Train on Imagenet

Transfer Learning with CNNs

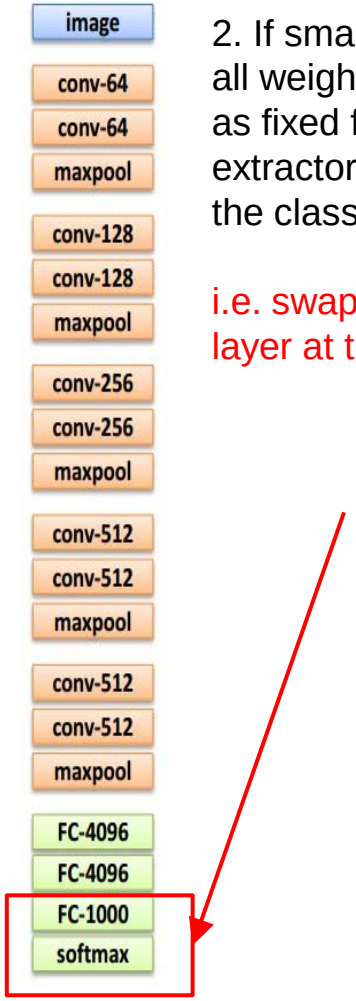
1. Train on Imagenet

2. If small dataset: fix all weights (treat CNN as fixed feature extractor), retrain only the classifier

i.e. swap the Softmax layer at the end

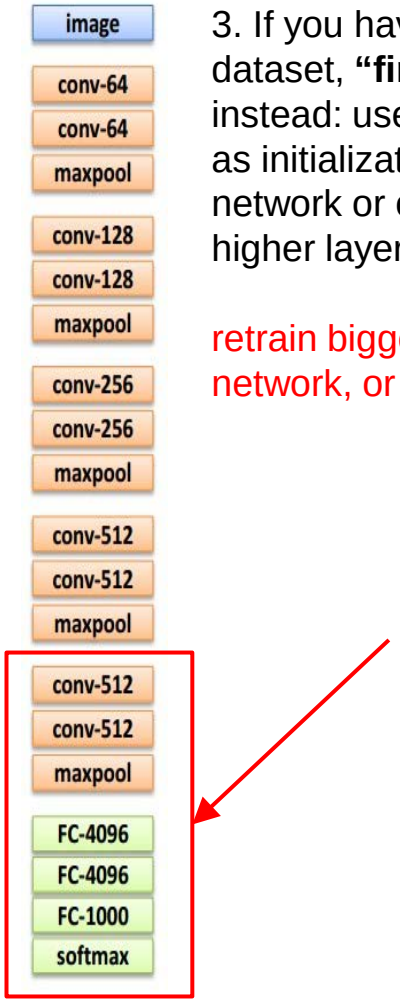
Transfer Learning with CNNs

1. Train on Imagenet



2. If small dataset: fix all weights (treat CNN as fixed feature extractor), retrain only the classifier

i.e. swap the Softmax layer at the end



3. If you have medium sized dataset, **"finetune"** instead: use the old weights as initialization, train the full network or only some of the higher layers

retrain bigger portion of the network, or even all of it.

Object Detection: PASCAL VOC mean Average Precision (mAP)

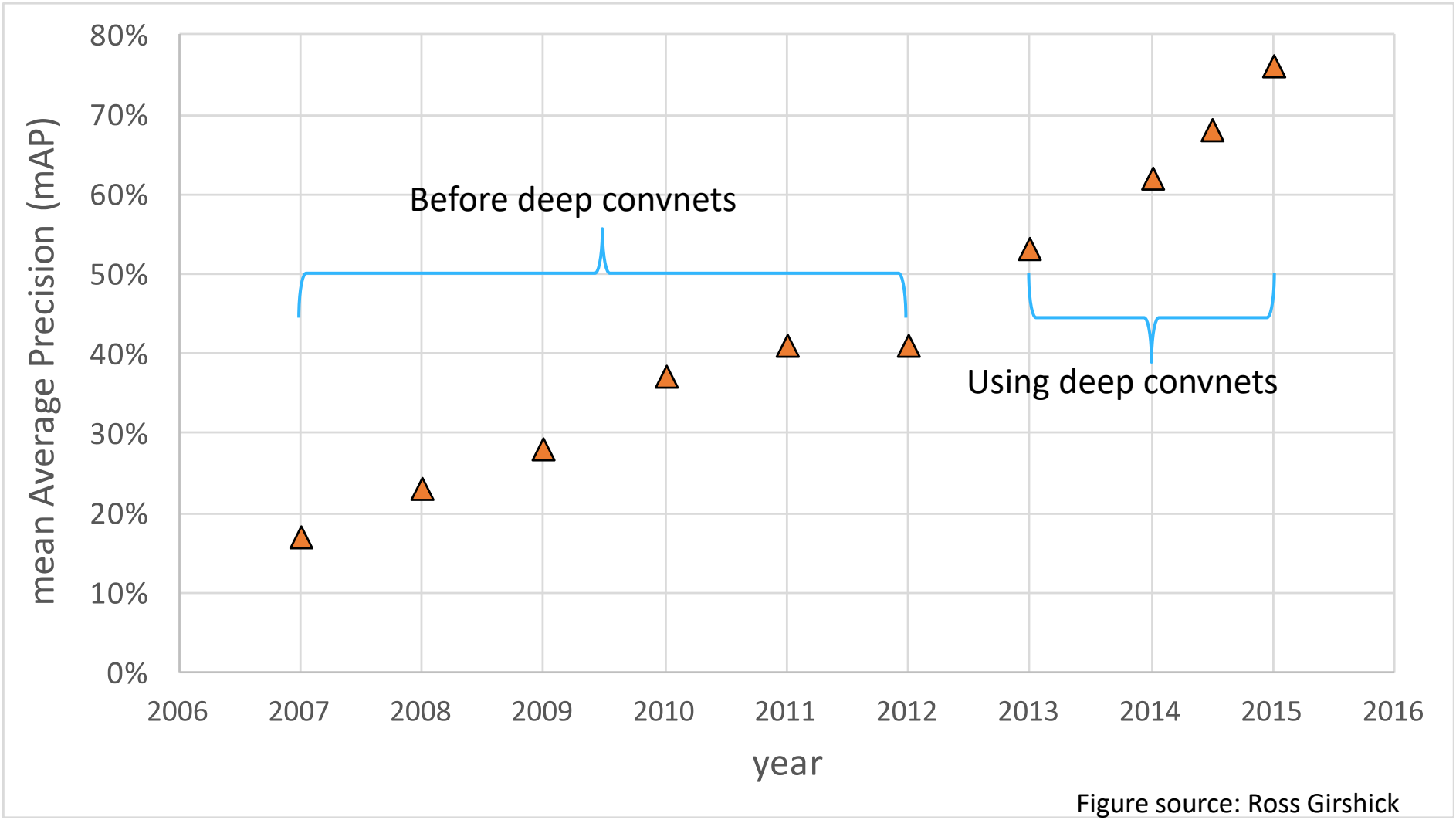


Figure source: Ross Girshick