AlexNet, VGG, GoogleNet and ResNet

Presented by Zhengxia Zou

18 Aug. 2018

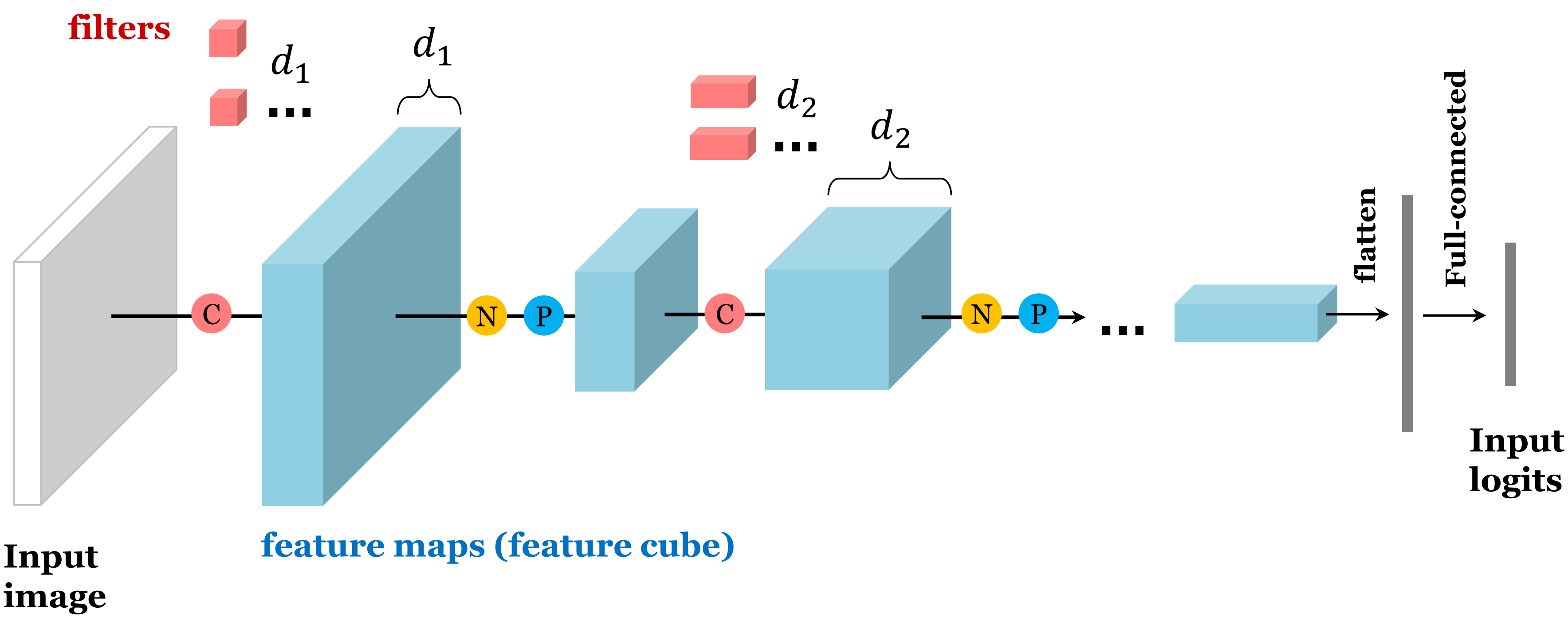
- **Basic components of CNN**
- **AlexNet**
- **VGG**
- **GoogleNet**
- **ResNet**

Basic components of CNN

C : Convolutional layer

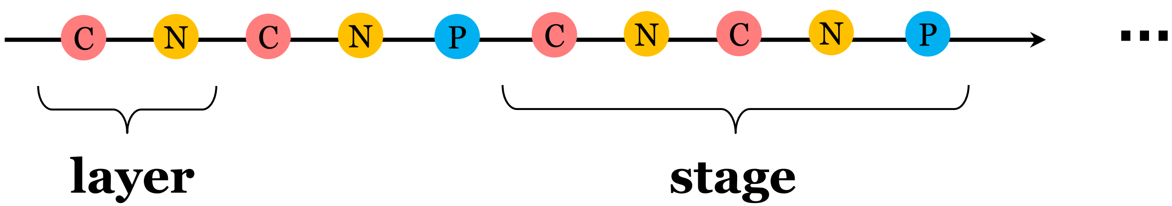
N : Nonlinear mapping layer

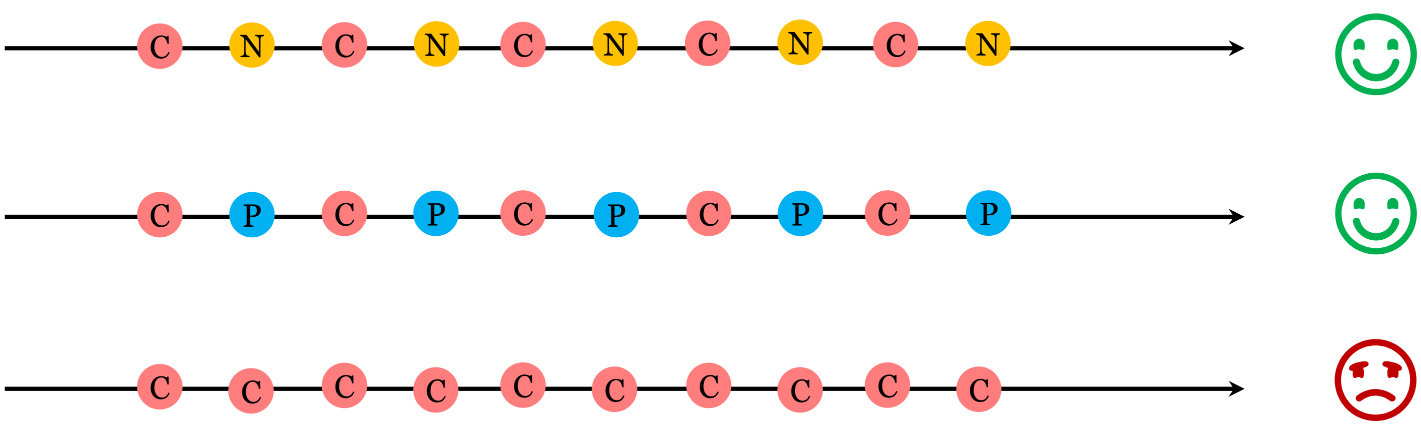
P : Pooling layer



Basic components of CNN

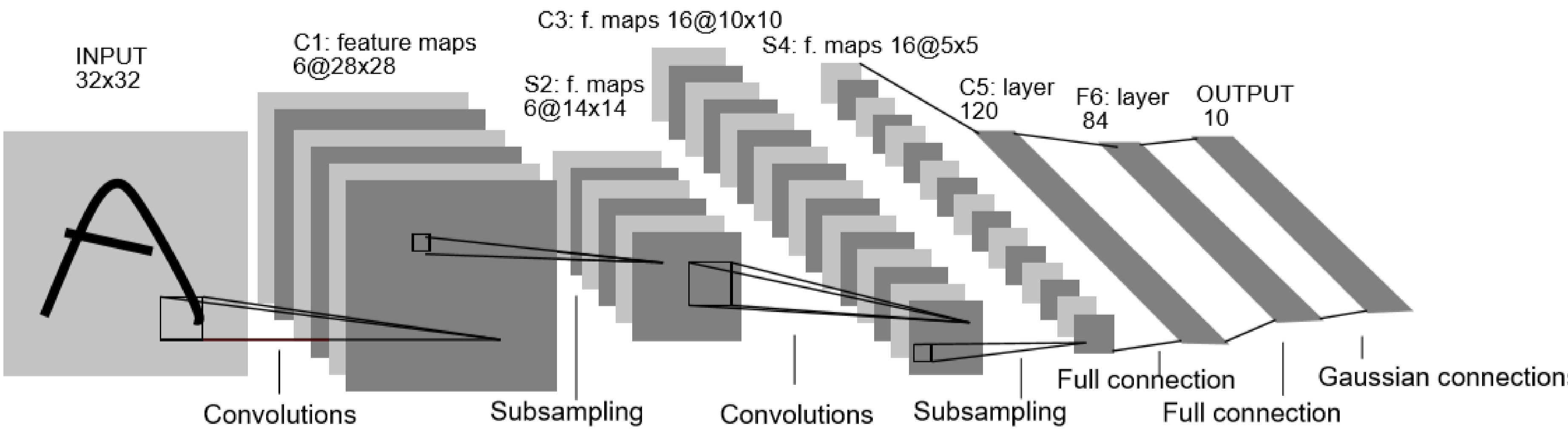
- C : Convolutional layer
- N : Nonlinear mapping layer
- P : Pooling layer

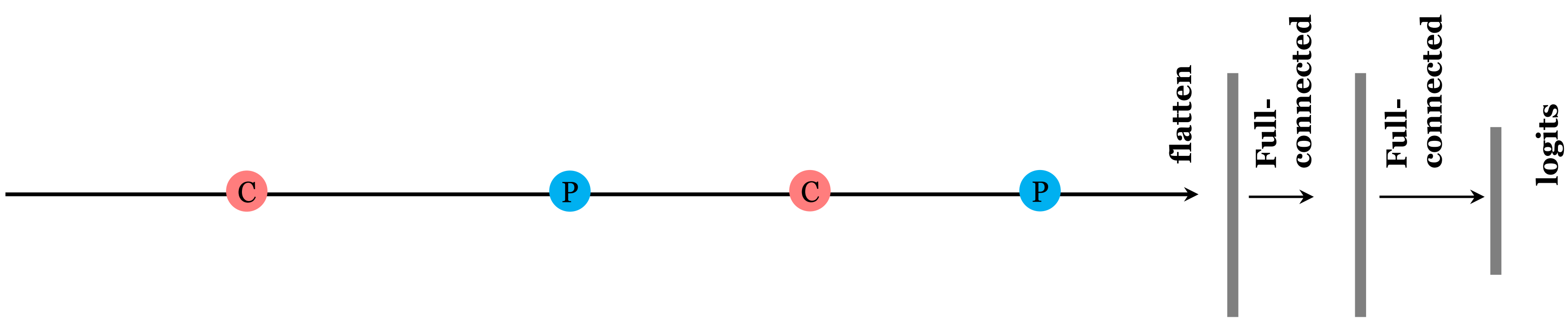




LeNet-5

Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 1998.

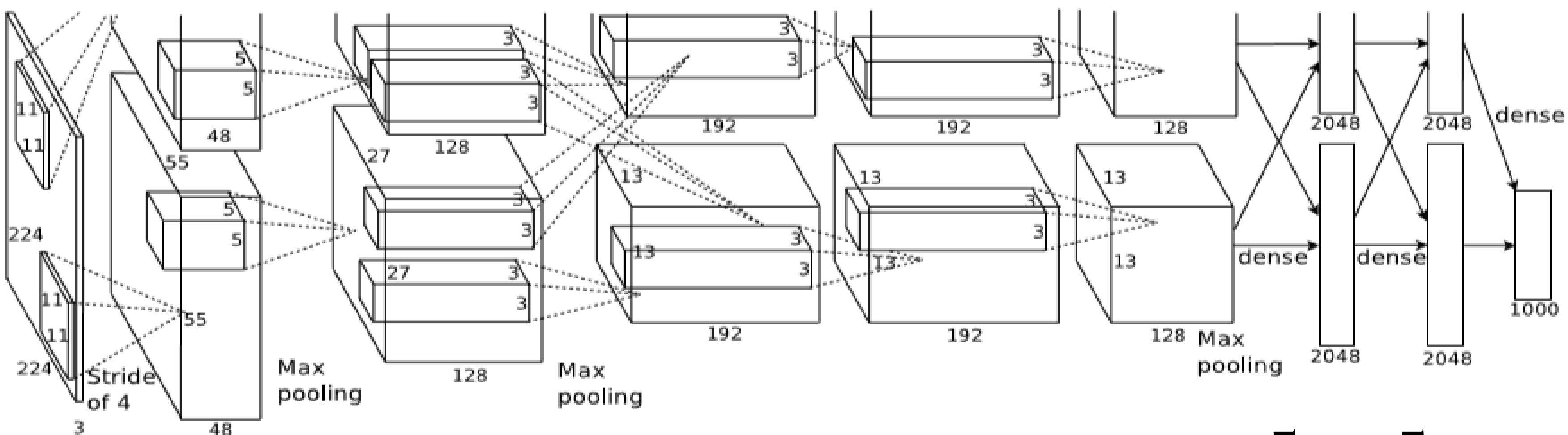


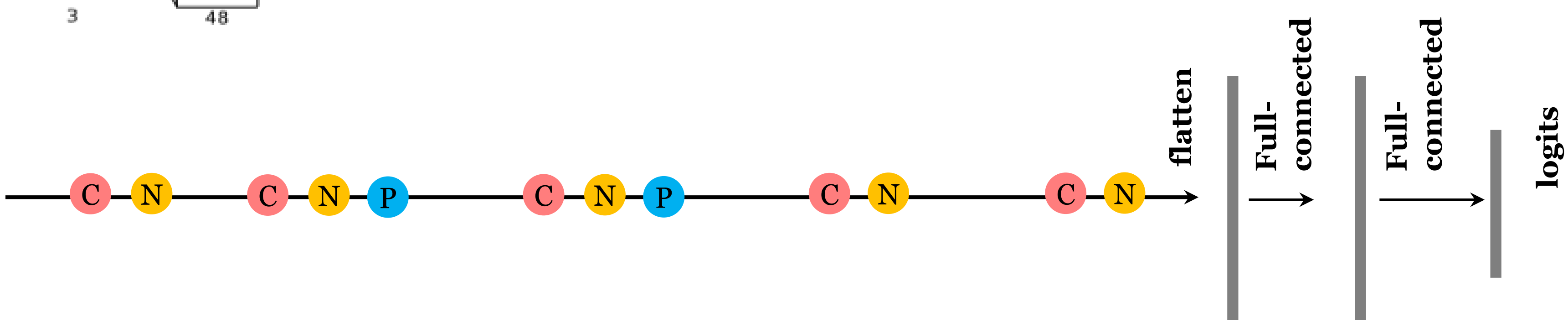


AlexNet

Krizhevsky, Alex, I. Sutskever, and G. E. Hinton. "ImageNet classification with deep convolutional neural networks." *NIPS*, 2012.

2012 ImageNet top-5 test error: 1st: 15.3%, 2nd: 26.2%

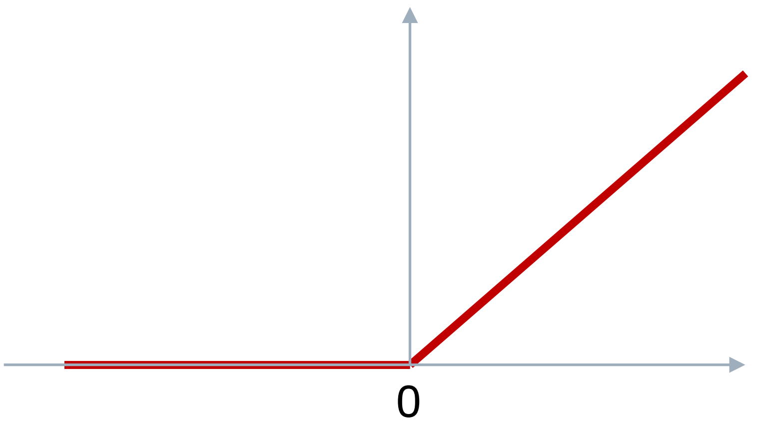




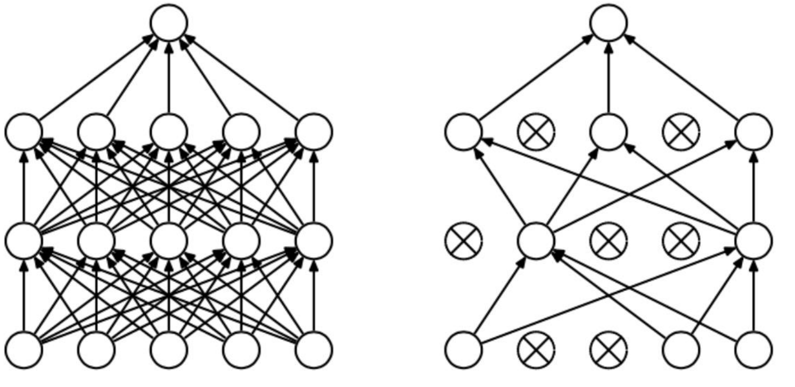
AlexNet

Key points of AlexNet

- Rectified Linear Unit (ReLU)
- Dropout
- Local Response Normalization (LRN)



ReLU

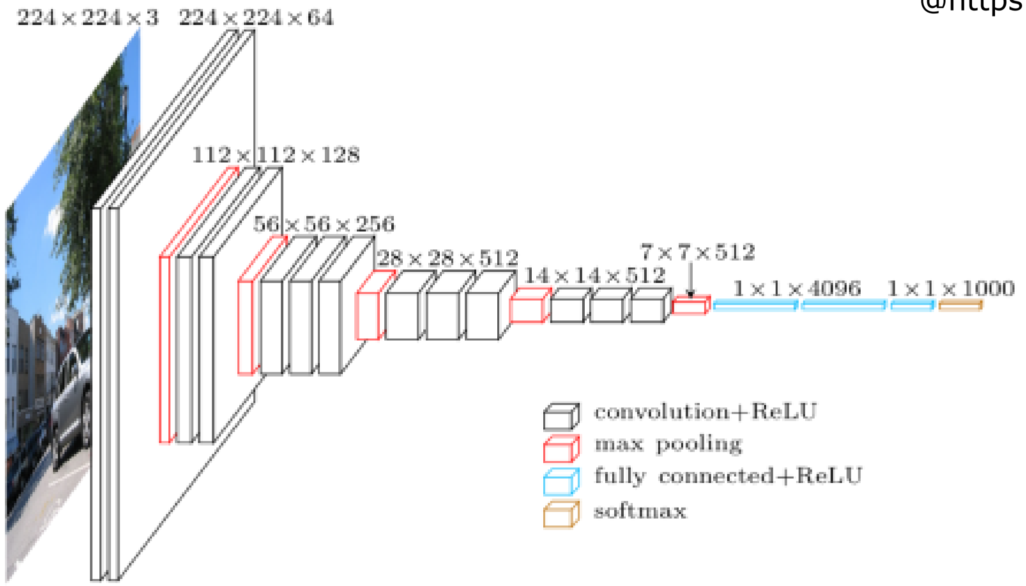


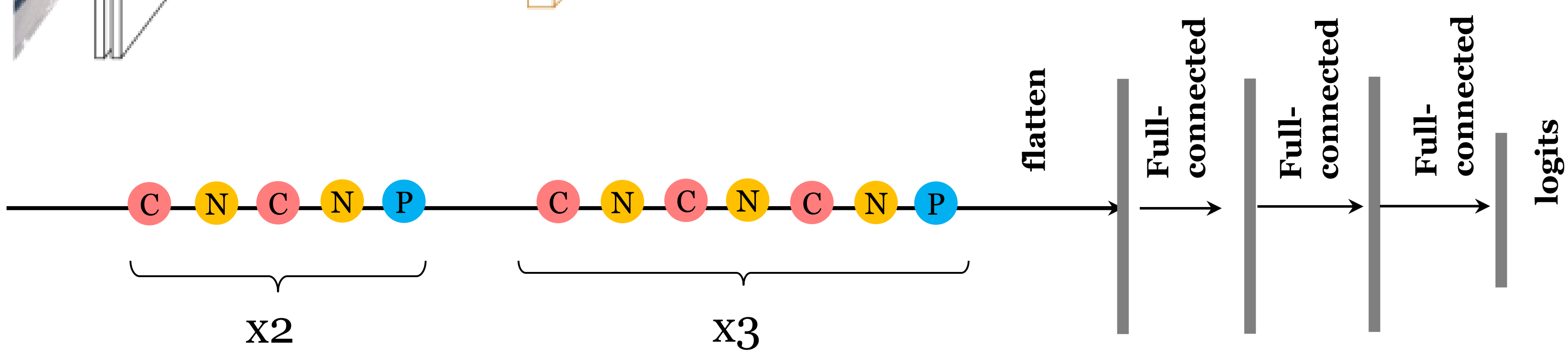
Dropout

VGG

Karen Simonyan, Andrew Zisserman. "Very Deep Convolutional Networks for Large-Scale Image Recognition." *ICLR*, 2015. (arXiv 2014)

@https://www.cs.toronto.edu/~frossard/post/vgg16/

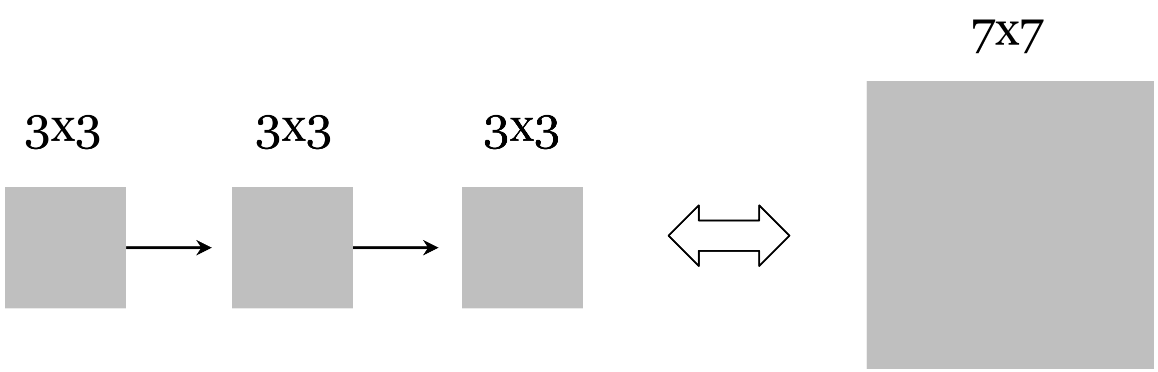




VGG

Key points of VGG

- Increasing depth (16-19 layers)
- Using very small (3x3) convolution filters (instead of 5x5 and 7x7)

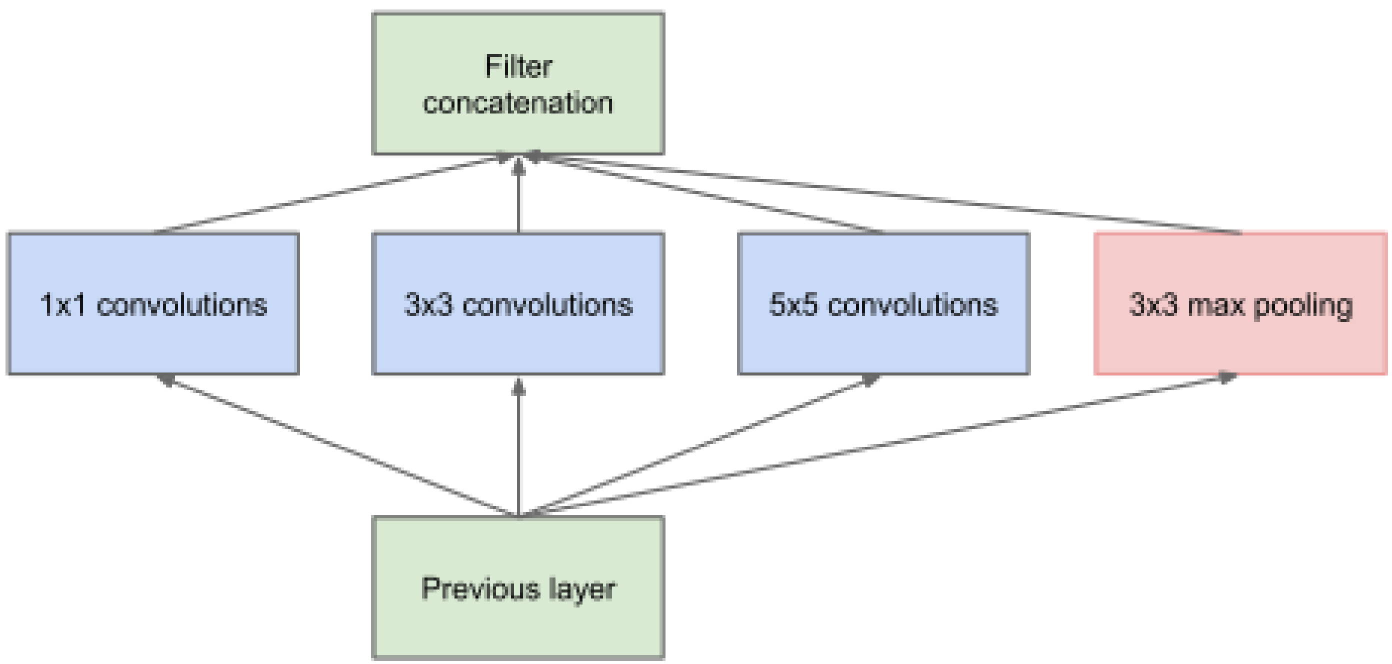


27 parameters

49 parameters:

GoogleNet

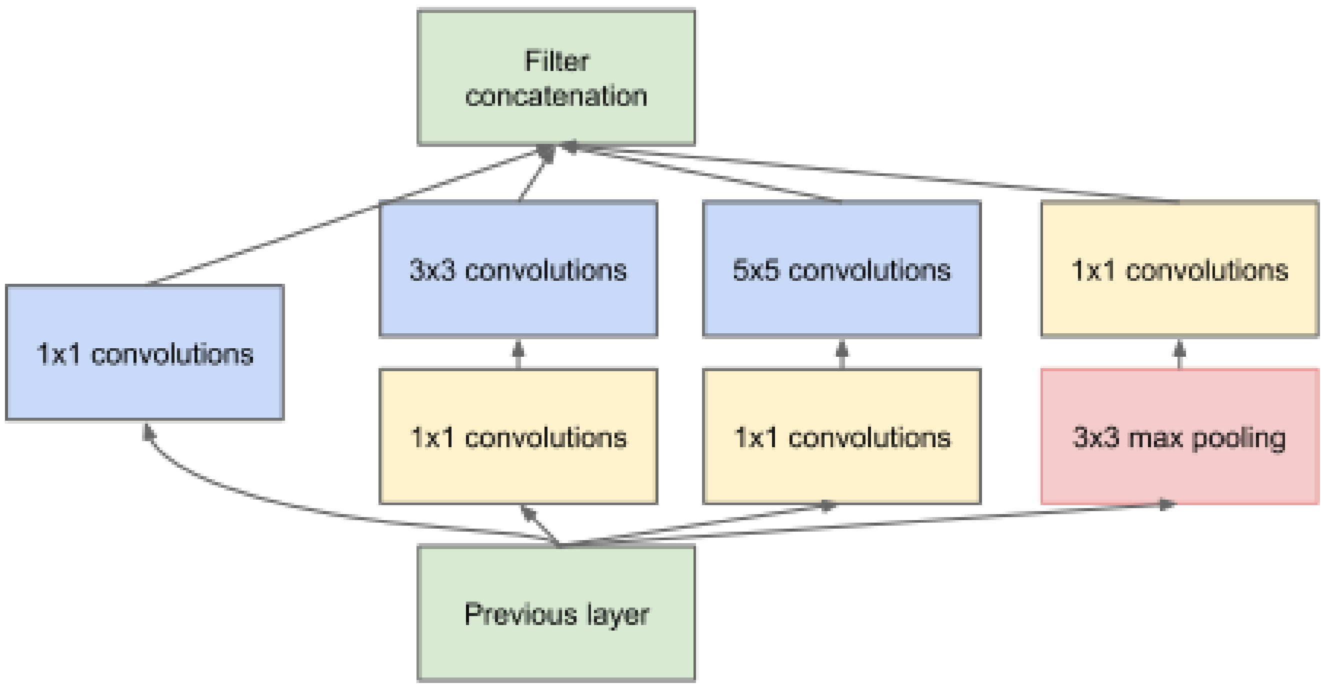
Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, et, al. “Going Deeper with Convolutions.” *CVPR*, 2015. (arXiv 2014)



(a) Inception module, naïve version

GoogleNet

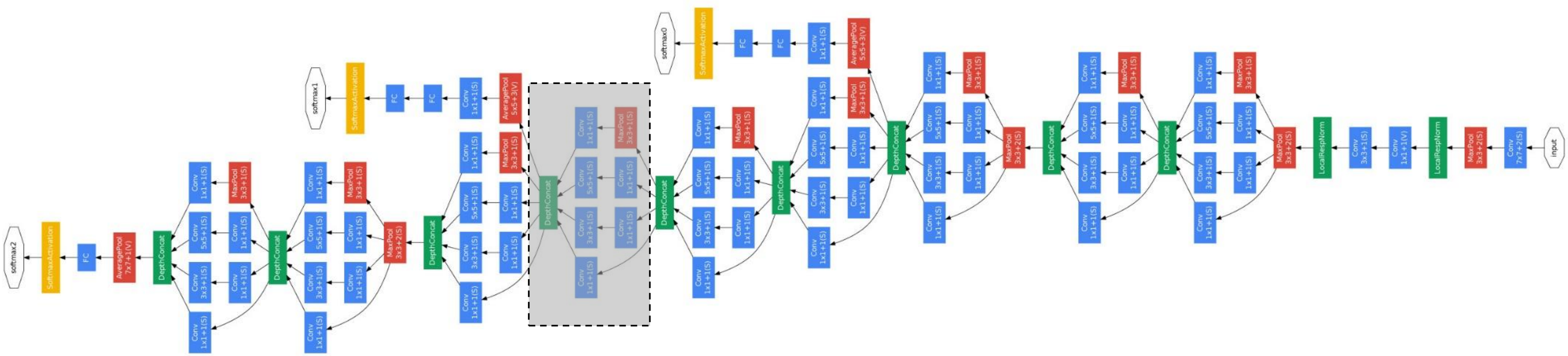
Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, et, al. “Going Deeper with Convolutions.” *CVPR*, 2015. (arXiv 2014)

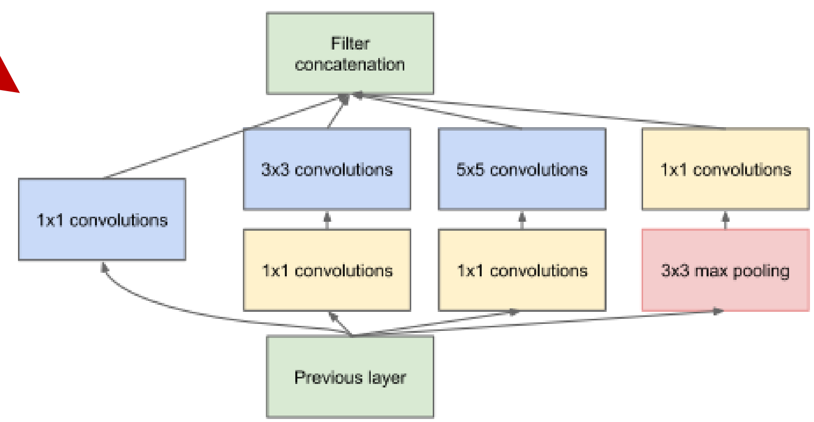


(b) Inception module with dimensionality reduction

GoogleNet

Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, et, al. “Going Deeper with Convolutions.” *CVPR*, 2015. (arXiv 2014)





(b) Inception module with dimensionality reduction

GoogleNet

Key points of GoogleNet (Inception v1-v4)

- Multiscale convolutions
- Increasing both of the width and depth (22 layers)
- Batch Normalization (BN)

$$\hat{x}^{(k)} = \frac{x^k - E[x^k]}{\sqrt{Var[x^k]}}$$

$$y^{(k)} = \gamma^k \hat{x}^{(k)} + \beta^{(k)}$$

Act as a pre-processing step:
shifting inputs of each layer to zero-mean and unit variance

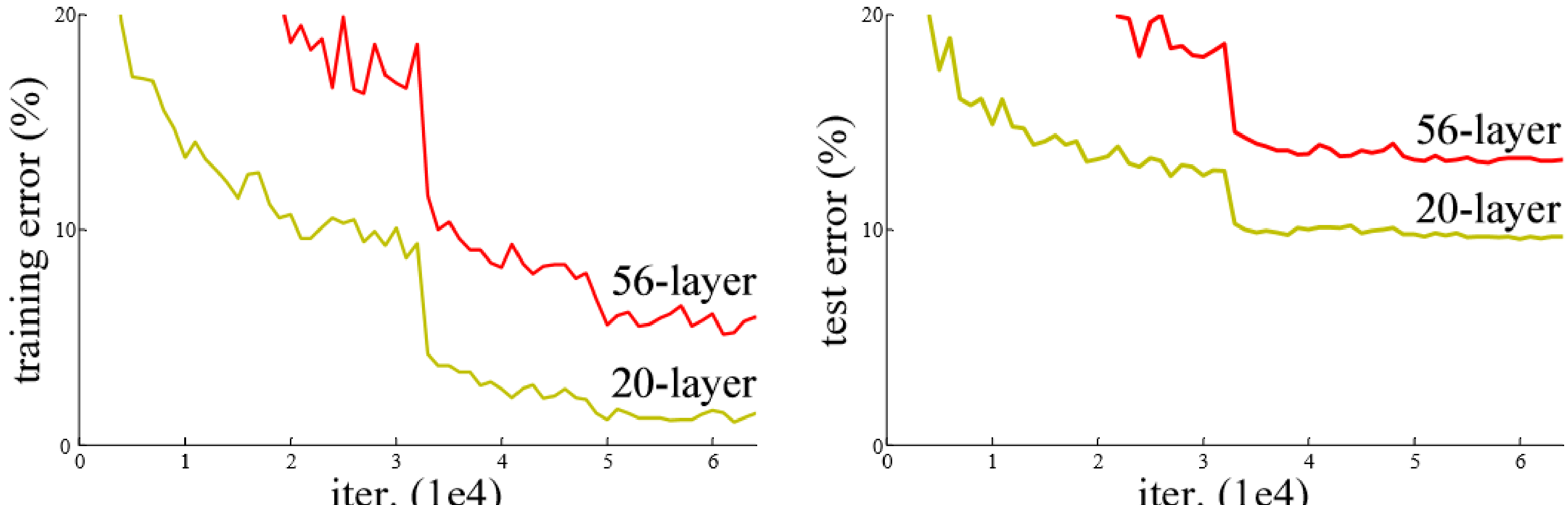
ResNet

Kaiming He, Xiangyu Zhang, Shaoqing Ren, Jian Sun. "Deep Residual Learning for Image Recognition." *CVPR*, 2016, best paper. (arXiv 2015)

Motivation

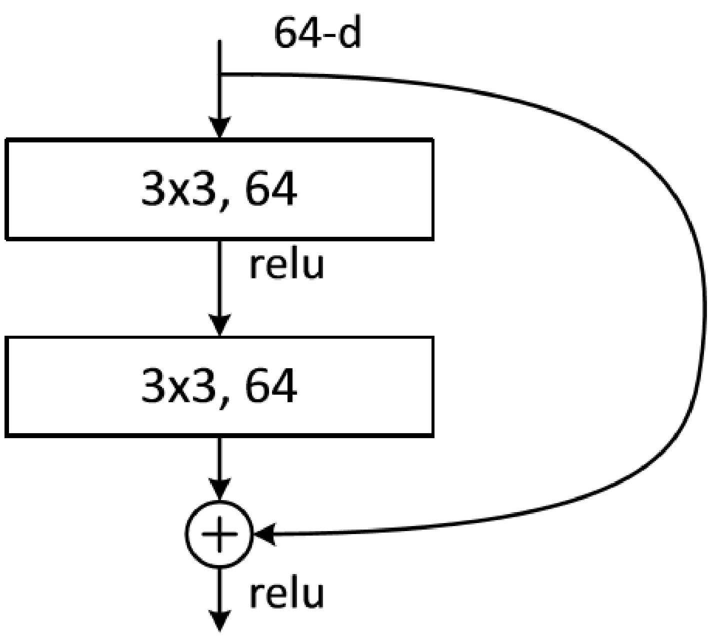
To ease the training of networks that are much deeper than those used previously.

Results on CIFAR10

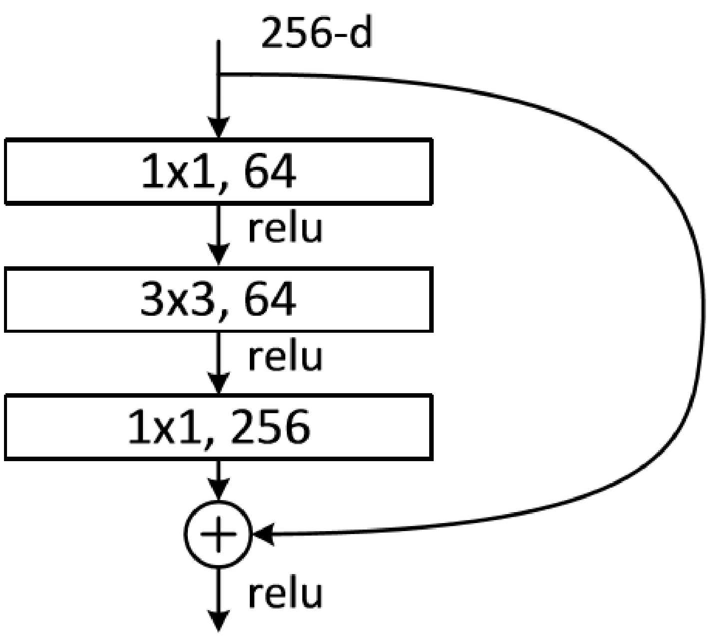


ResNet

Building Block of ResNet



naïve version



"bottleneck" version

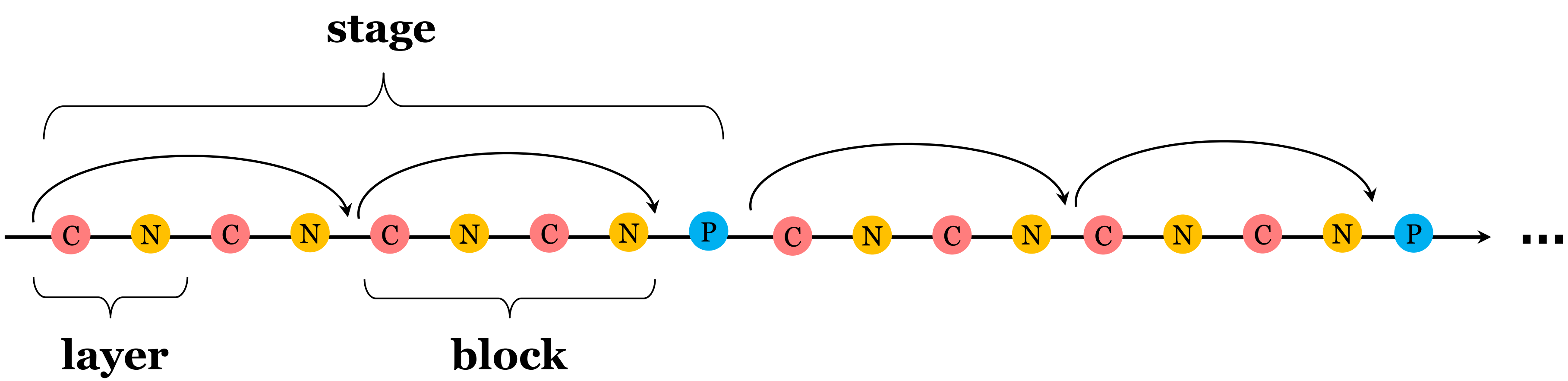
$$\mathbf{y} = \mathcal{F}(\mathbf{x}, \{W_i\}) + \mathbf{x}.$$

ResNet

C : Convolutional layer

N : Nonlinear mapping layer

P : Pooling layer



ResNet

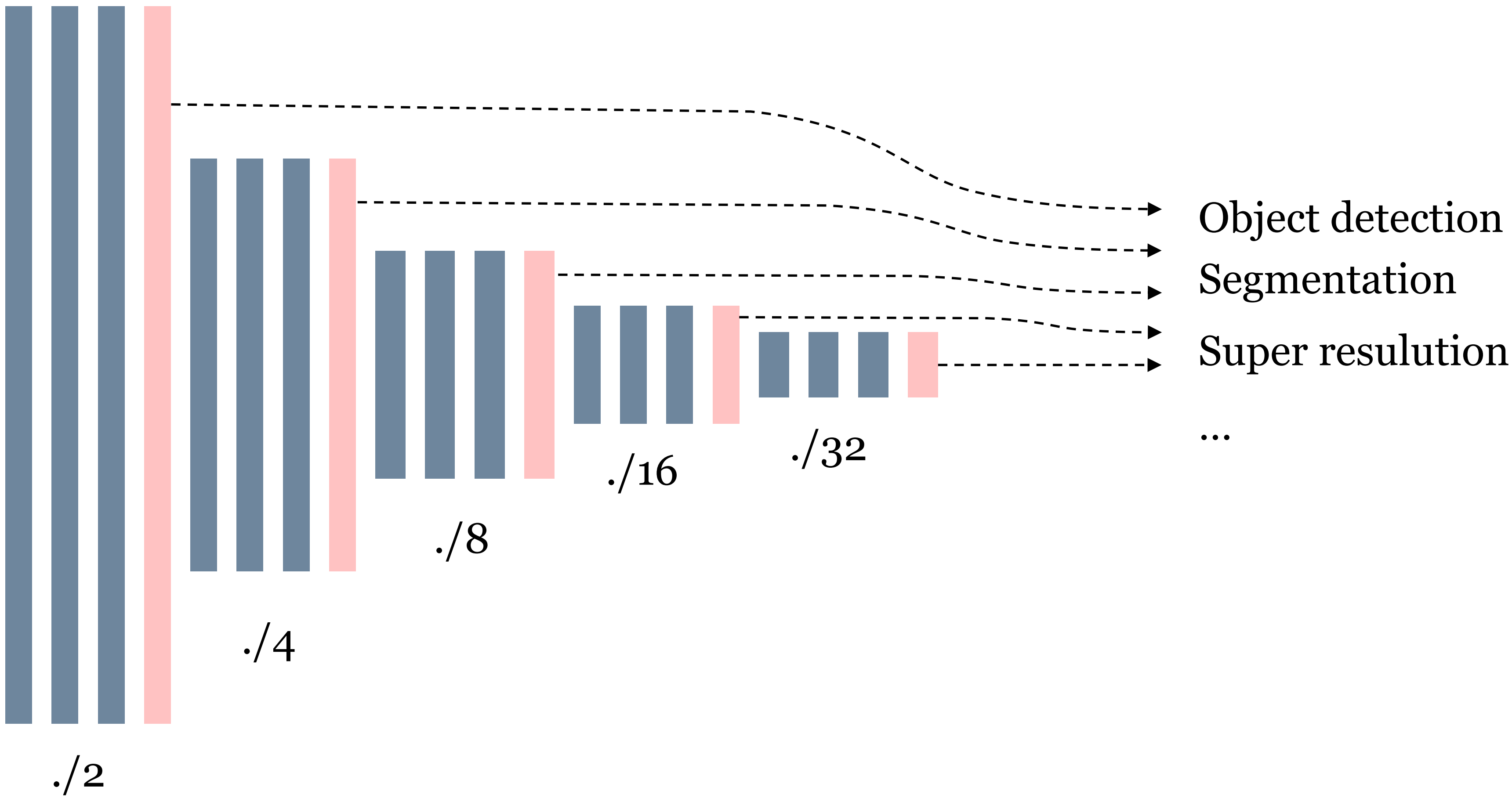
layer name	output size	18-layer	34-layer	50-layer	101-layer	152-layer
conv1	112×112	7×7, 64, stride 2				
conv2_x	56×56	3×3 max pool, stride 2				
		$\begin{bmatrix} 3\times3, 64 \\ 3\times3, 64 \end{bmatrix}\times2$	$\begin{bmatrix} 3\times3, 64 \\ 3\times3, 64 \end{bmatrix}\times3$	$\begin{bmatrix} 1\times1, 64 \\ 3\times3, 64 \\ 1\times1, 256 \end{bmatrix}\times3$	$\begin{bmatrix} 1\times1, 64 \\ 3\times3, 64 \\ 1\times1, 256 \end{bmatrix}\times3$	$\begin{bmatrix} 1\times1, 64 \\ 3\times3, 64 \\ 1\times1, 256 \end{bmatrix}\times3$
conv3_x	28×28	$\begin{bmatrix} 3\times3, 128 \\ 3\times3, 128 \end{bmatrix}\times2$	$\begin{bmatrix} 3\times3, 128 \\ 3\times3, 128 \end{bmatrix}\times4$	$\begin{bmatrix} 1\times1, 128 \\ 3\times3, 128 \\ 1\times1, 512 \end{bmatrix}\times4$	$\begin{bmatrix} 1\times1, 128 \\ 3\times3, 128 \\ 1\times1, 512 \end{bmatrix}\times4$	$\begin{bmatrix} 1\times1, 128 \\ 3\times3, 128 \\ 1\times1, 512 \end{bmatrix}\times8$
conv4_x	14×14	$\begin{bmatrix} 3\times3, 256 \\ 3\times3, 256 \end{bmatrix}\times2$	$\begin{bmatrix} 3\times3, 256 \\ 3\times3, 256 \end{bmatrix}\times6$	$\begin{bmatrix} 1\times1, 256 \\ 3\times3, 256 \\ 1\times1, 1024 \end{bmatrix}\times6$	$\begin{bmatrix} 1\times1, 256 \\ 3\times3, 256 \\ 1\times1, 1024 \end{bmatrix}\times23$	$\begin{bmatrix} 1\times1, 256 \\ 3\times3, 256 \\ 1\times1, 1024 \end{bmatrix}\times36$
conv5_x	7×7	$\begin{bmatrix} 3\times3, 512 \\ 3\times3, 512 \end{bmatrix}\times2$	$\begin{bmatrix} 3\times3, 512 \\ 3\times3, 512 \end{bmatrix}\times3$	$\begin{bmatrix} 1\times1, 512 \\ 3\times3, 512 \\ 1\times1, 2048 \end{bmatrix}\times3$	$\begin{bmatrix} 1\times1, 512 \\ 3\times3, 512 \\ 1\times1, 2048 \end{bmatrix}\times3$	$\begin{bmatrix} 1\times1, 512 \\ 3\times3, 512 \\ 1\times1, 2048 \end{bmatrix}\times3$
	1×1	average pool, 1000-d fc, softmax				
FLOPs		1.8×10^9	3.6×10^9	3.8×10^9	7.6×10^9	11.3×10^9

ResNet

Key points of ResNet

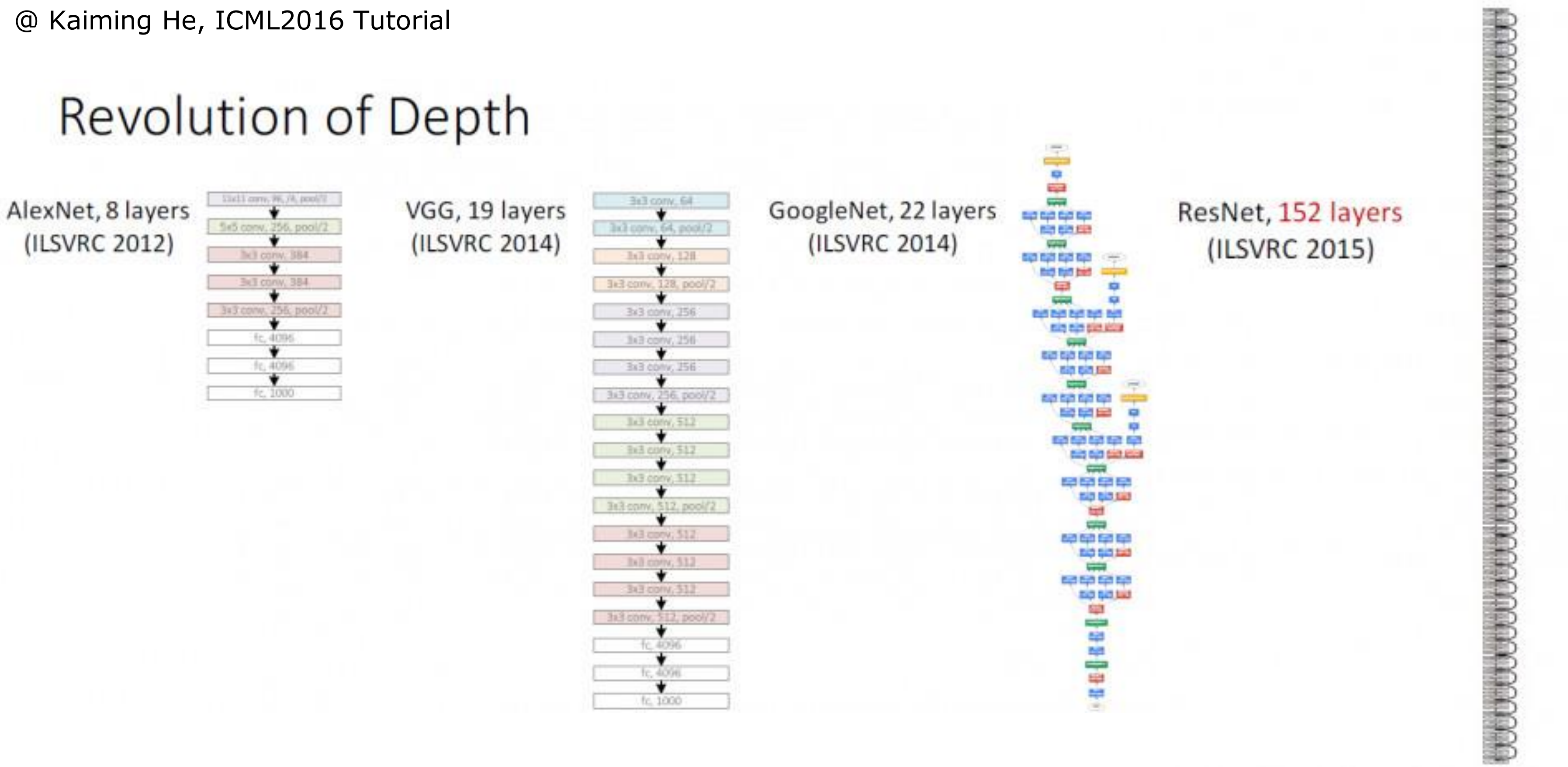
- Learning Residuals
- Much easier to optimize
- Gain accuracy from considerably increased depth (152 layers)

Apply to other tasks



Comparison

@ Kaiming He, ICML2016 Tutorial



ImageNet Cls. Top5-err:

15.3% 7.3% 6.7% 3.57%

Comparison

@ Alfredo Canziani, Adam Paszke, Eugenio Culurciello. An Analysis of Deep Neural Network Models for Practical Applications. arXiv 2016.