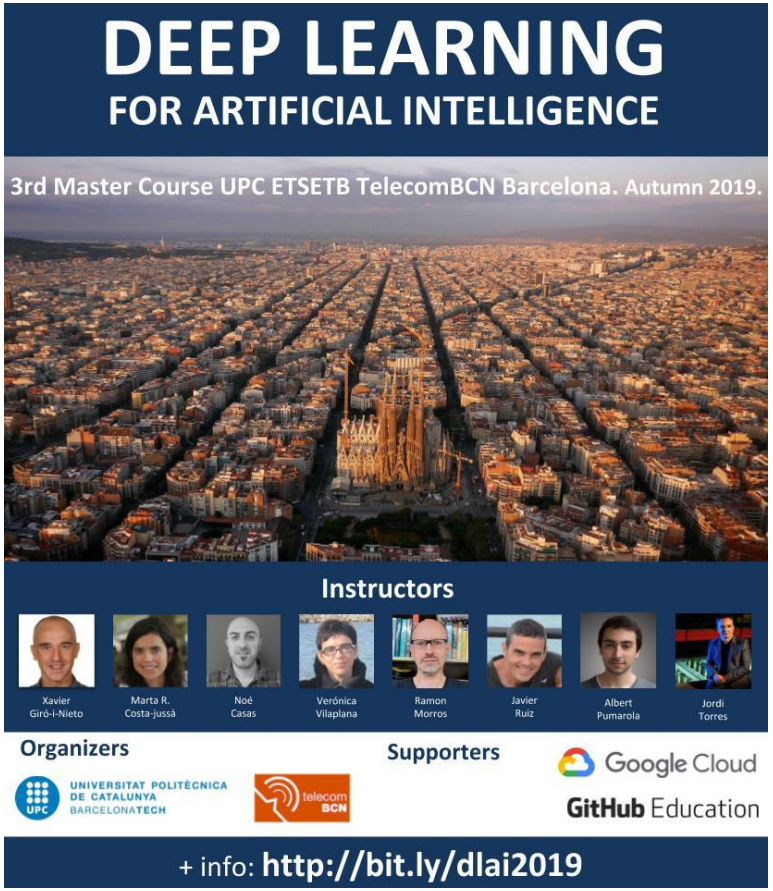


[course site]

#DLBCN

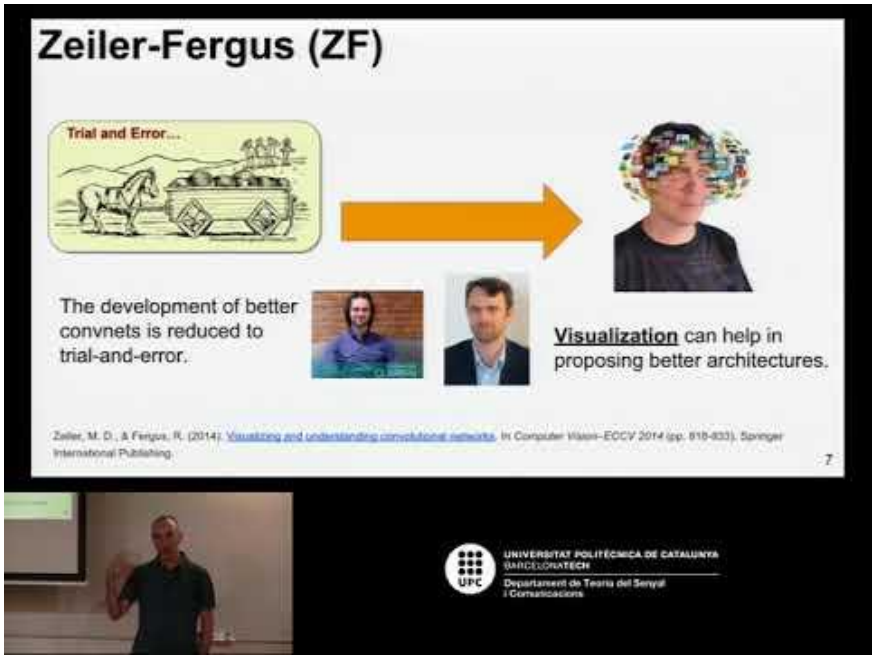
Day 8 Lecture 1

Advanced Neural Architectures

Xavier Giro-i-Nieto
xavier.giro@upc.edu

Associate Professor
Universitat Politècnica de Catalunya
Technical University of Catalonia

Previous Videolectures

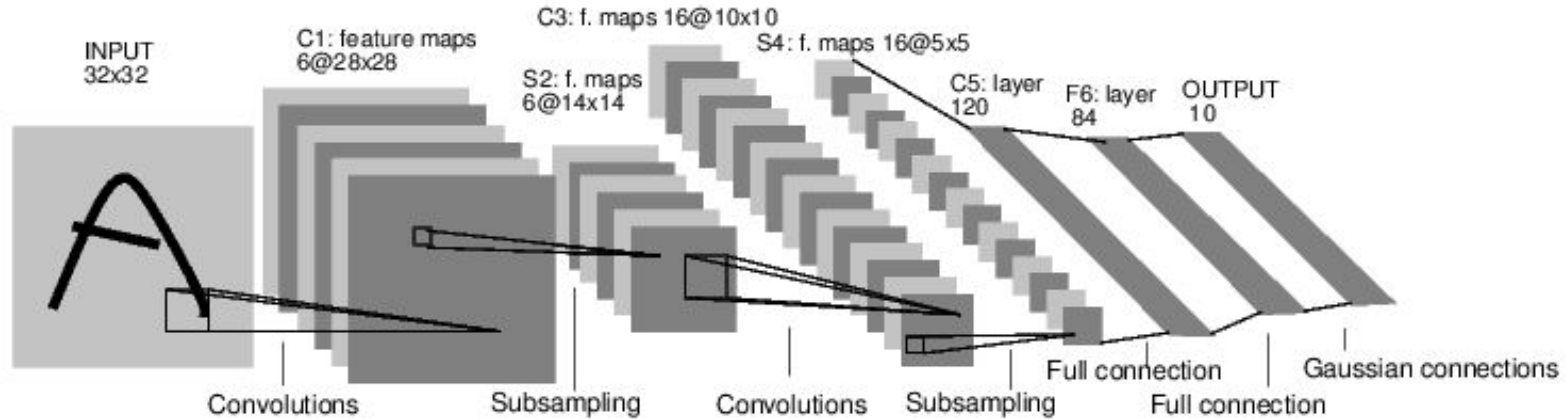


Xavier Giro-i-Nieto (DLCV 2017)

Kevin McGuinness (DLCV 2018)

Convolutional Neural Networks for Vision

LeNet-5: Several convolutional layers, combined with pooling layers, and followed by a small number of fully connected layers

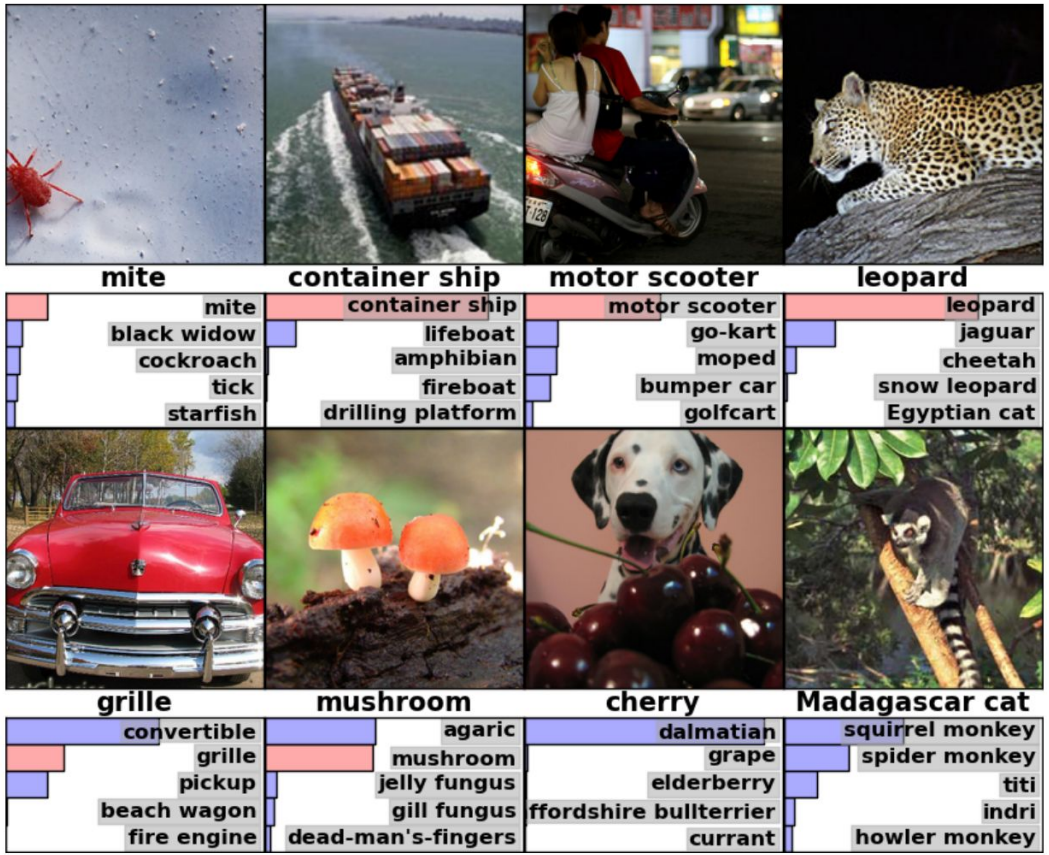


#LeNet-5 LeCun, Y., Bottou, L., Bengio, Y., & Haffner, P. (1998). Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, *86*(11), 2278-2324.

ImageNet Challenge

IMAGENET

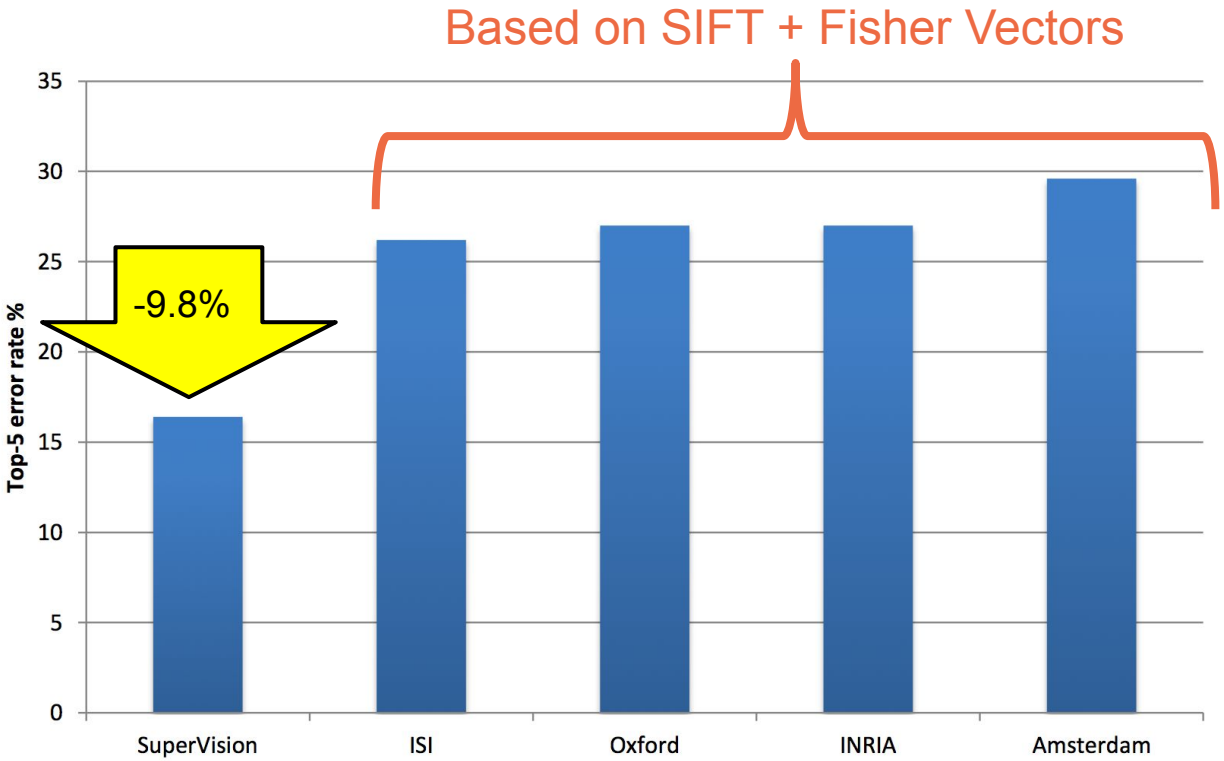
- 1,000 object classes (categories).
- Images:
  - 1.2 M train
  - 100k test.



ImageNet Challenge

Russakovsky, Olga, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang et al. "Imagenet large scale visual recognition challenge." International Journal of Computer Vision 115, no. 3 (2015): 211-252. [web]

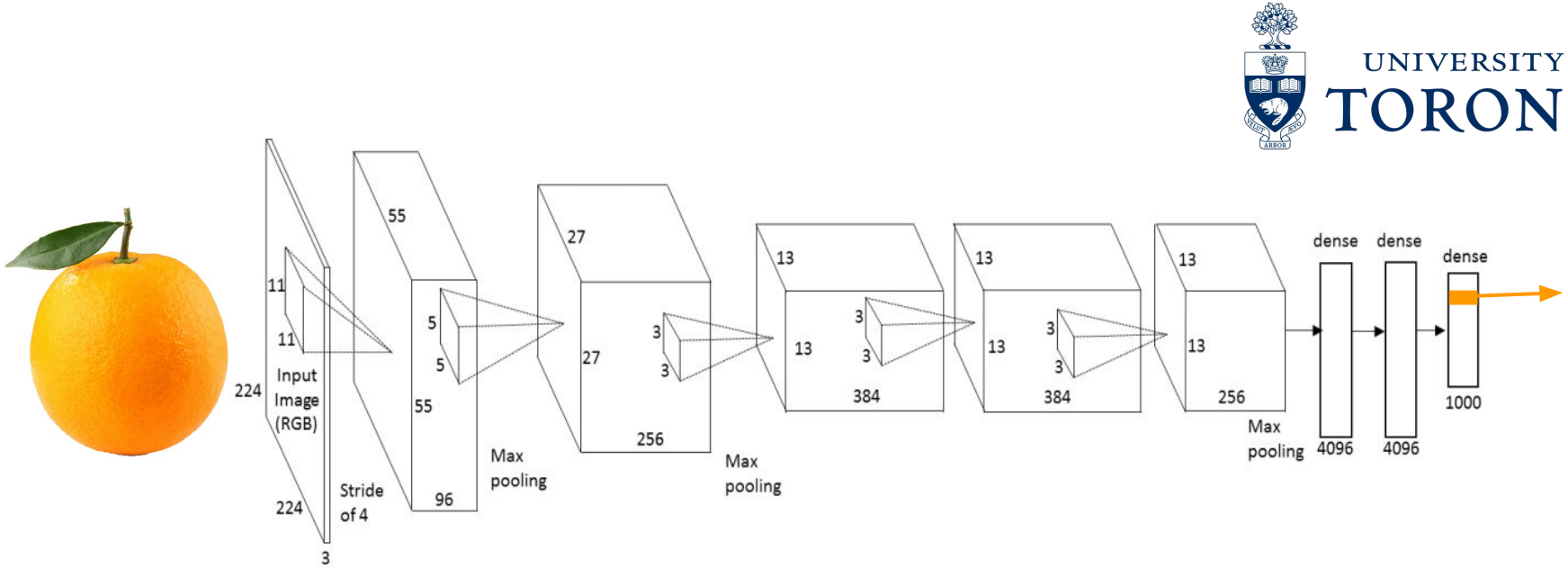
ImageNet Challenge: 2012



Slide credit:
Rob Fergus (NYU)

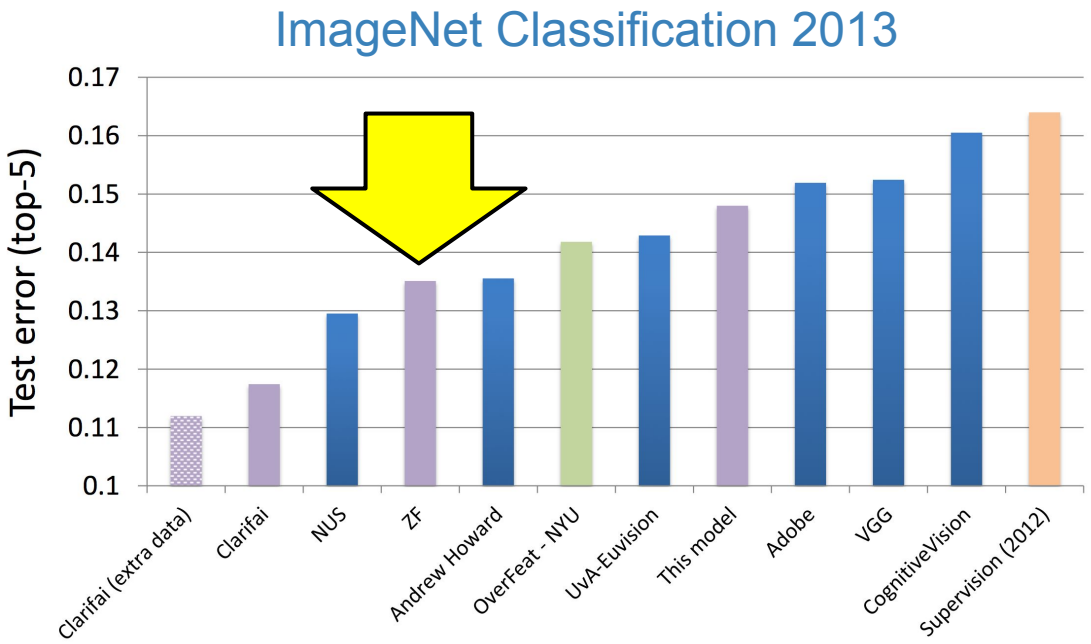
Russakovsky, Olga, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang et al. "Imagenet large scale visual recognition challenge." International Journal of Computer Vision 115, no. 3 (2015): 211-252. [web]

AlexNet (SuperVision)



#AlexNet Krizhevsky, Alex, Ilya Sutskever, and Geoffrey E. Hinton. "Imagenet classification with deep convolutional neural networks." NIPS **2012**

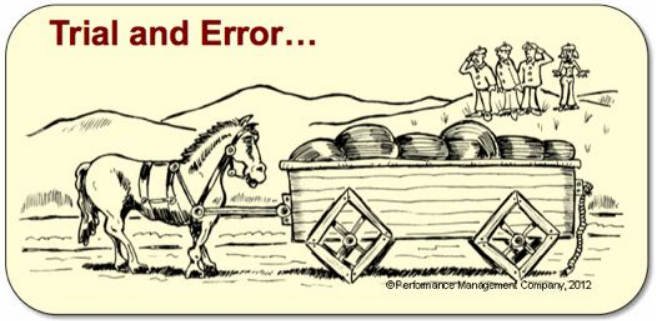
ImageNet Challenge: 2013



Slide credit:
Rob Fergus (NYU)

Russakovsky, Olga, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang et al. "Imagenet large scale visual recognition challenge." International Journal of Computer Vision 115, no. 3 (2015): 211-252. [web]

Zeiler-Fergus (ZF)

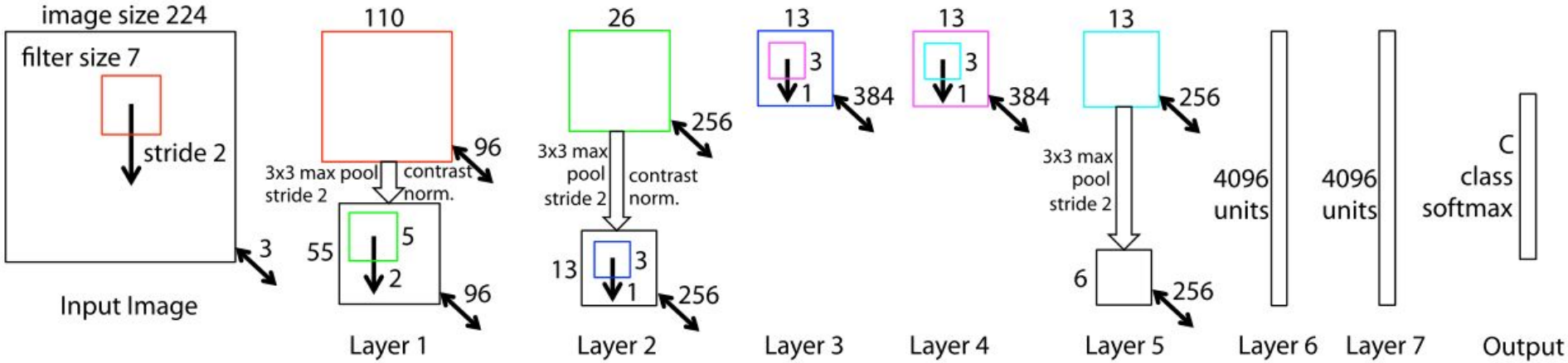


The development of better convnets is reduced to trial-and-error.

Visualization can help in proposing better architectures.

Zeiler, M. D., & Fergus, R. . Visualizing and understanding convolutional networks. ECCV 2014

Zeiler-Fergus (ZF)

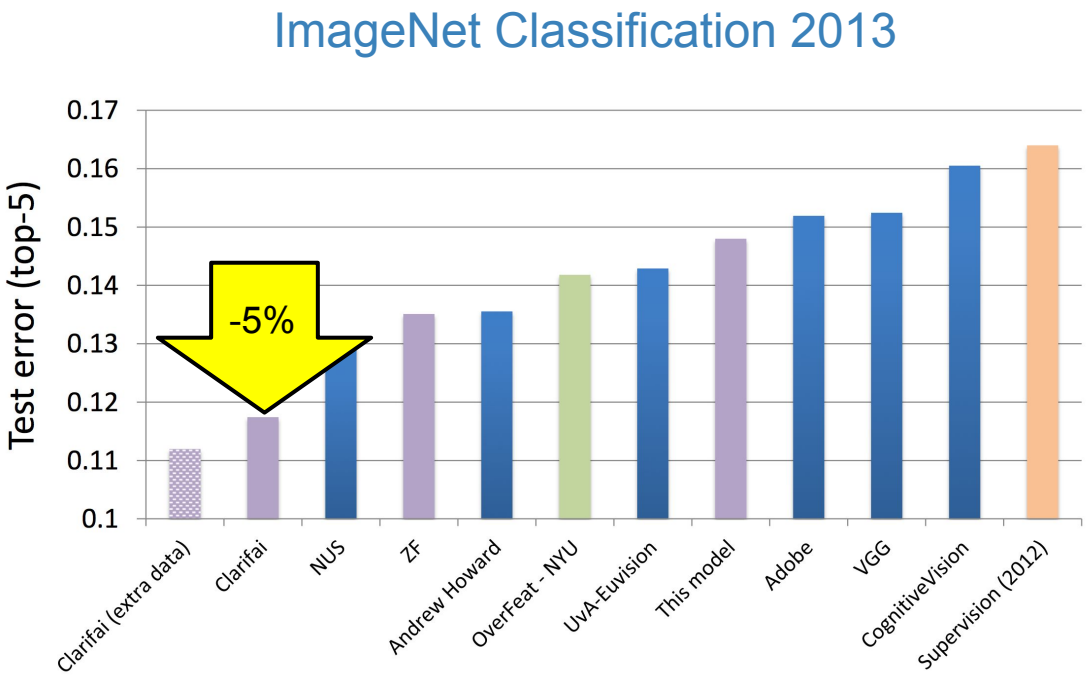


Zeiler, M. D., & Fergus, R. . Visualizing and understanding convolutional networks. ECCV 2014

ImageNet Challenge: 2013

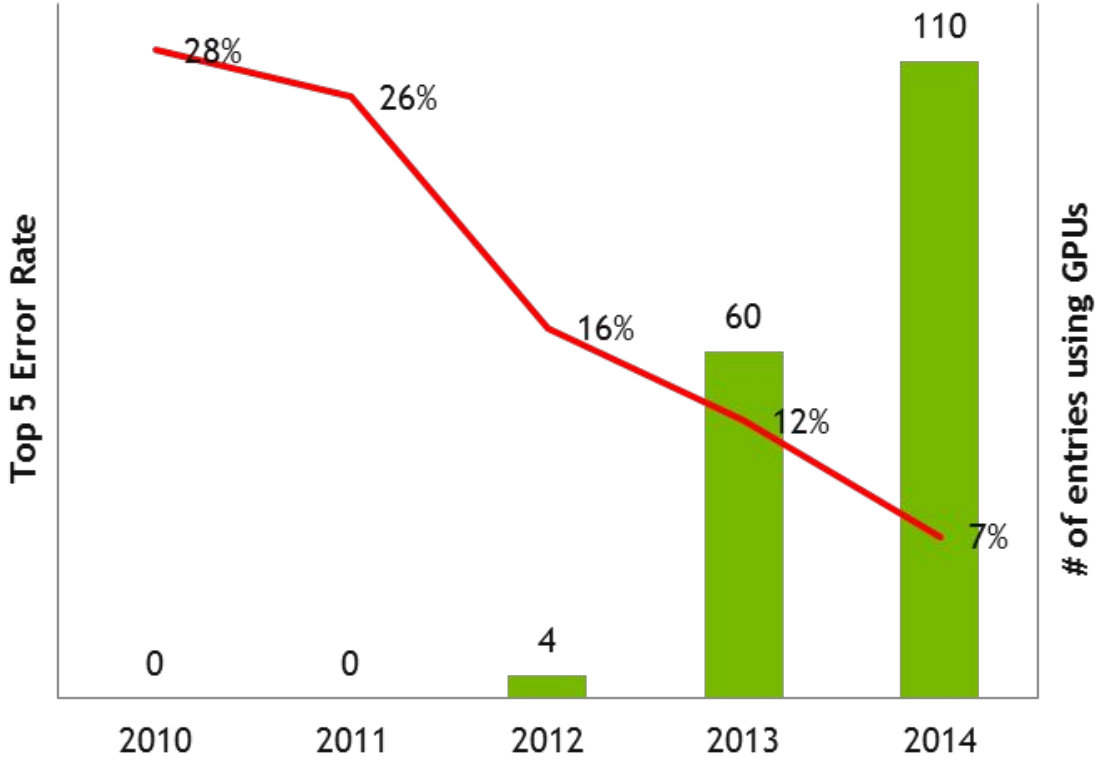
IMAGENET

clarifai



Russakovsky, Olga, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang et al. "Imagenet large scale visual recognition challenge." IJCV 2015. [web]

ImageNet Challenge: 2014

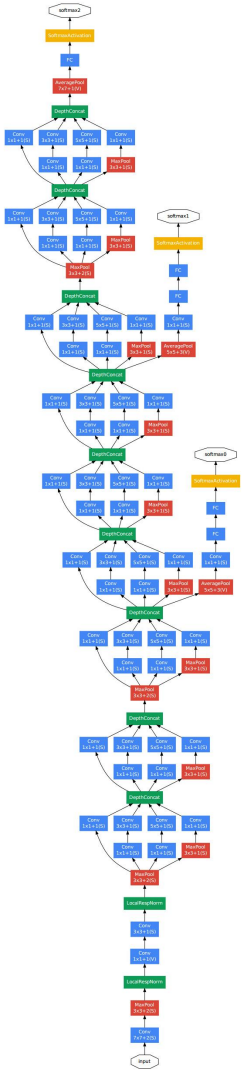


NVIDIA, [“NVIDIA and IBM CLoud Support ImageNet Large Scale Visual Recognition Challenge”]() (2015)

ImageNet Challenge: 2014

AlexNet

- image
- conv-64
- conv-192
- conv-384
- conv-256
- conv-256
- FC-4096
- FC-4096
- FC-1000



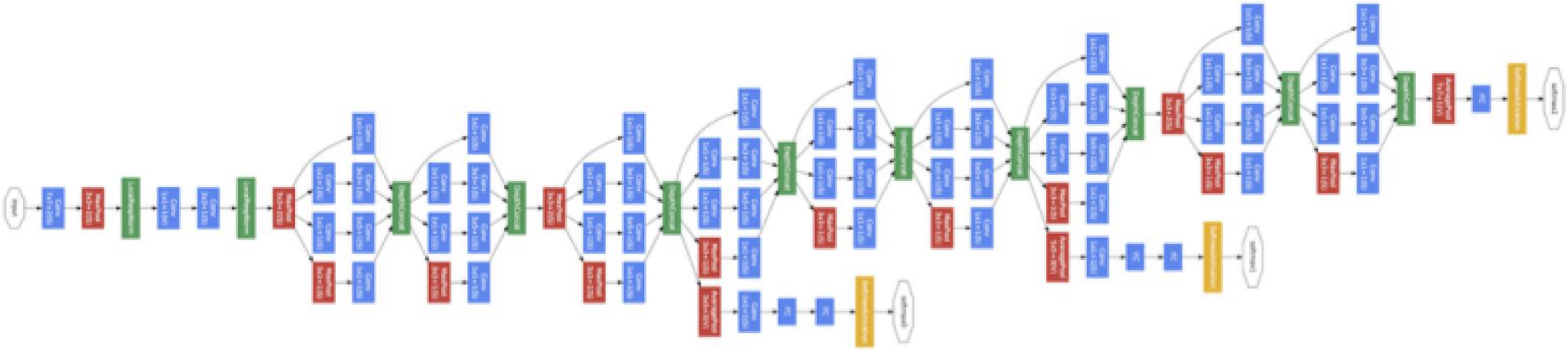
- image
- conv-64
- conv-64
- maxpool
- conv-128
- conv-128
- maxpool
- conv-256
- conv-256
- conv-256
- conv-256
- maxpool
- conv-512
- conv-512
- conv-512
- conv-512
- maxpool
- conv-512
- conv-512
- conv-512
- conv-512
- maxpool
- FC-4096
- FC-4096
- FC-1000
- softmax

GoogleNet (Inception)

Movie: Inception (2010)

GoogleNet (Inception)

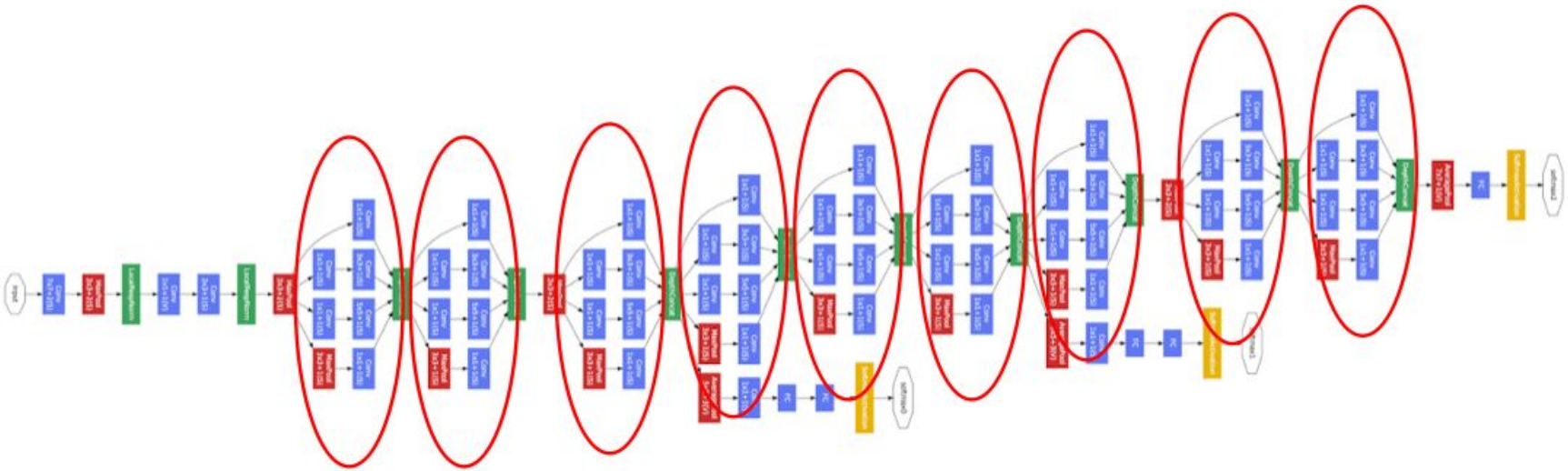
22 layers !



Convolution
Pooling
Softmax
Other

#Inception Szegedy, Christian, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich. "Going deeper with convolutions." CVPR 2015. [video] [slides] [poster]

GoogleNet (Inception)

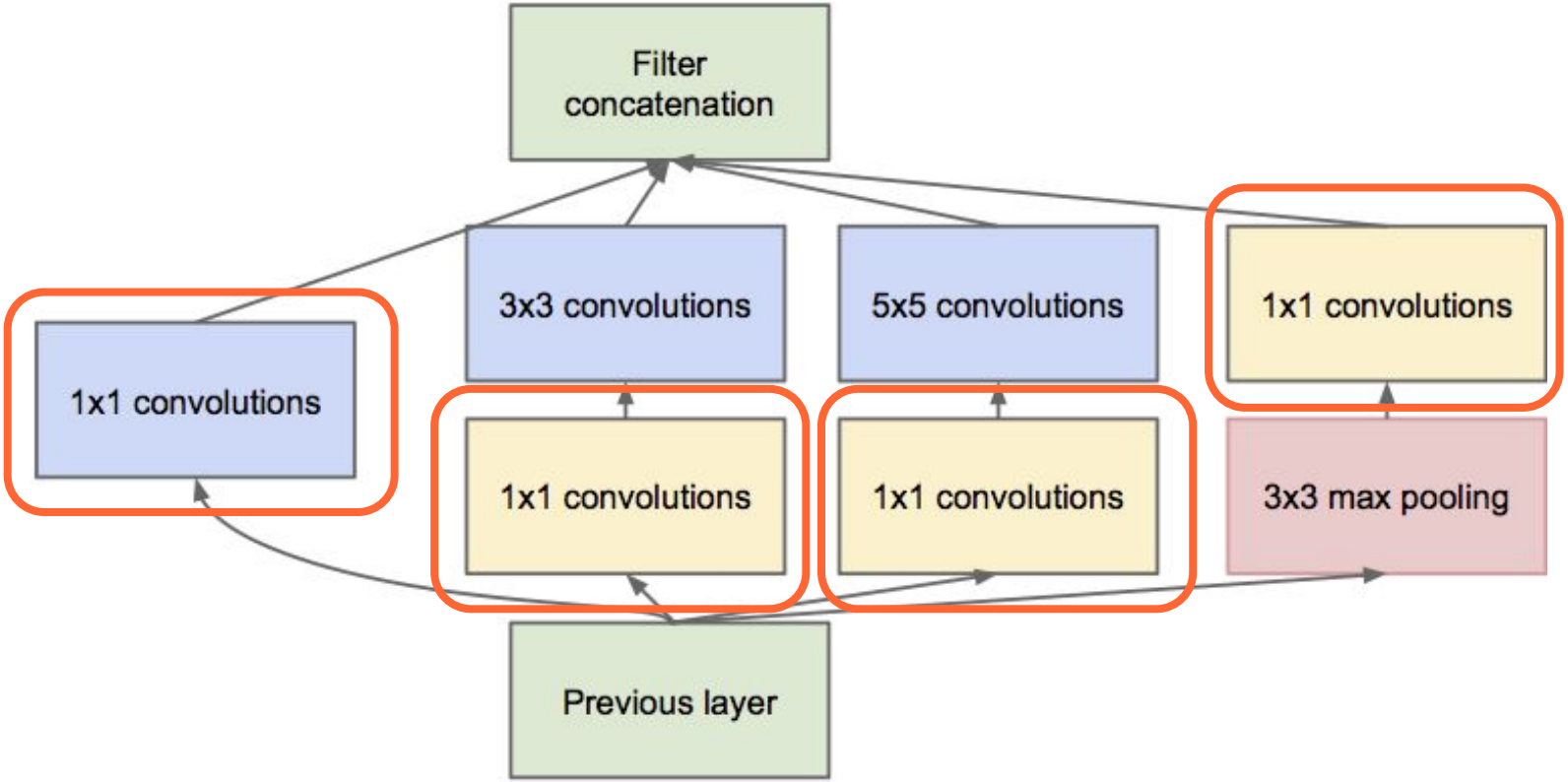


9 **Inception** modules

Network in a network in a network...

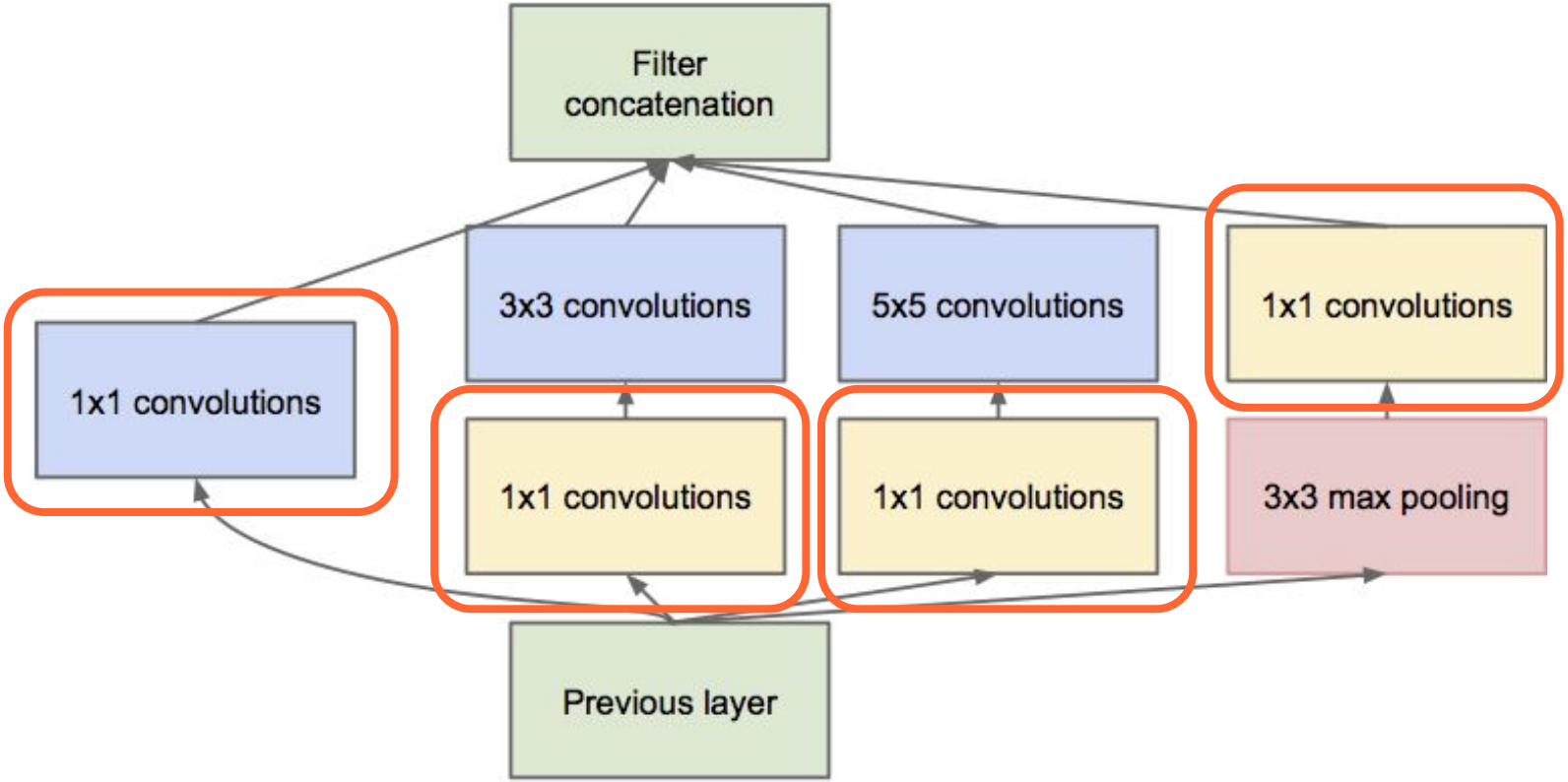
Convolution
Pooling
Softmax
Other

Multiple convolutional filter sizes (receptive field)



#NiN Lin, Min, Qiang Chen, and Shuicheng Yan. "Network in network." *ICLR 2014.*

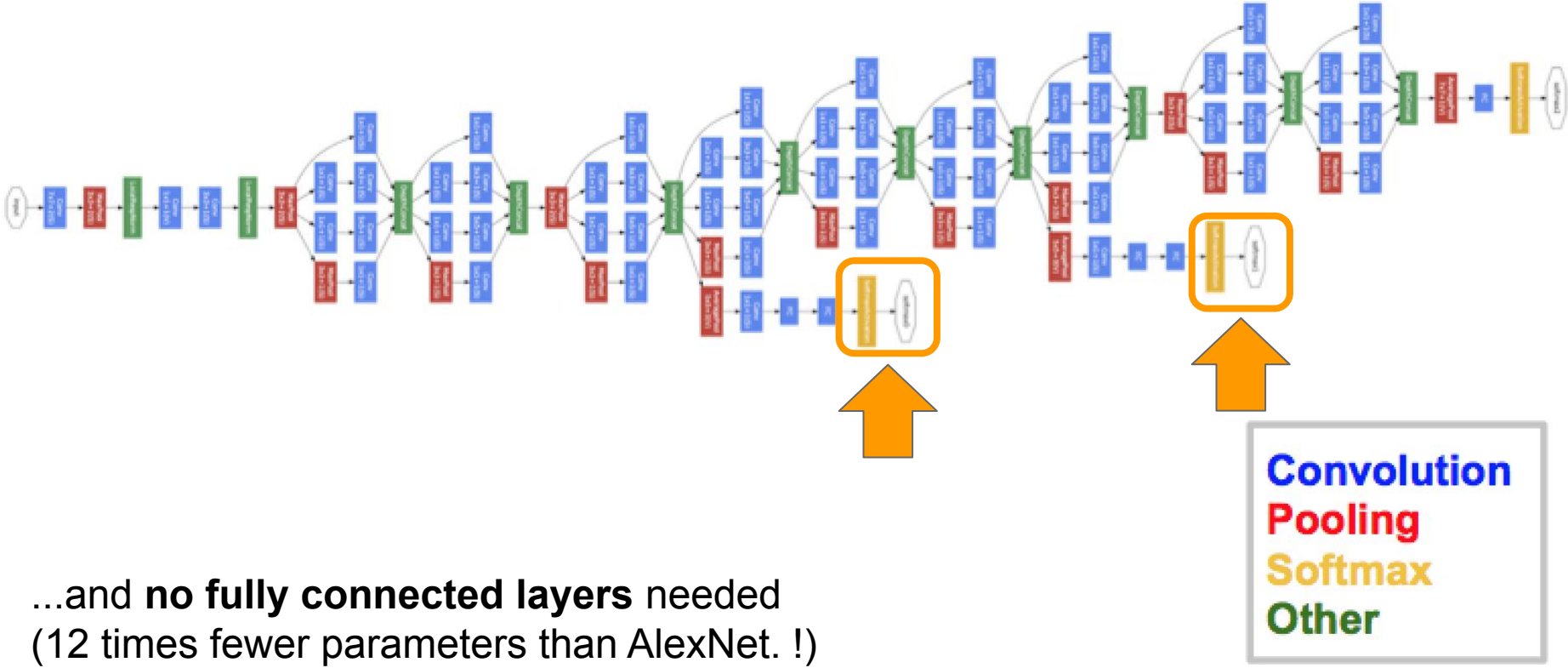
Pointise convolutions (bottlenecks)



#NiN Lin, Min, Qiang Chen, and Shuicheng Yan. "Network in network." *ICLR 2014.*

GoogleNet (Inception)

Two Softmax classifiers at intermediate layers combat the vanishing gradient while providing regularization at training time.



...and **no fully connected layers** needed
(12 times fewer parameters than AlexNet. !)

GoogleNet (Inception)

#Inception Szegedy, Christian, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich. "Going deeper with convolutions." CVPR 2015. [video] [slides] [poster]

Xception

Depth-wise (in channels) separable convolutions.

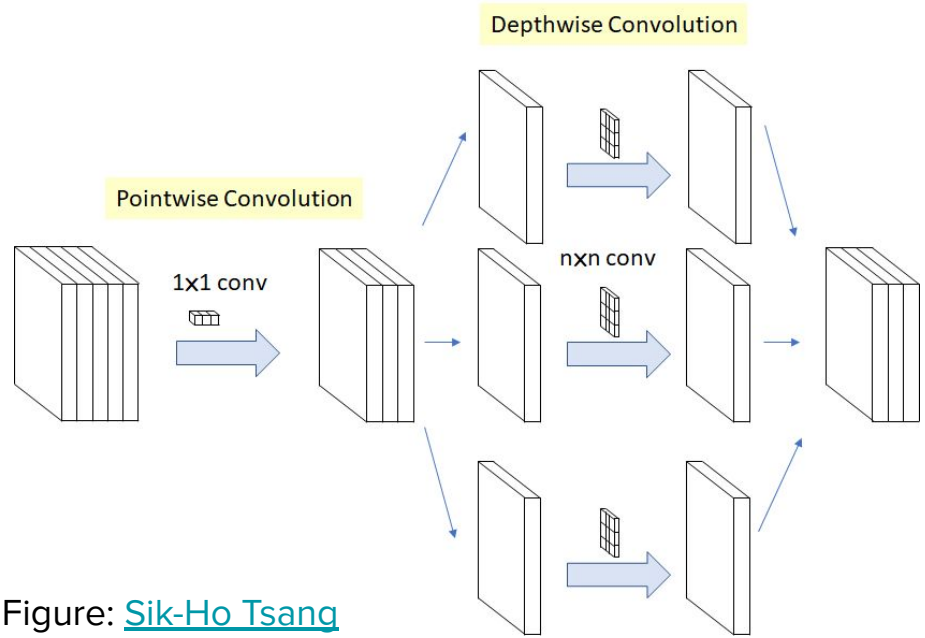
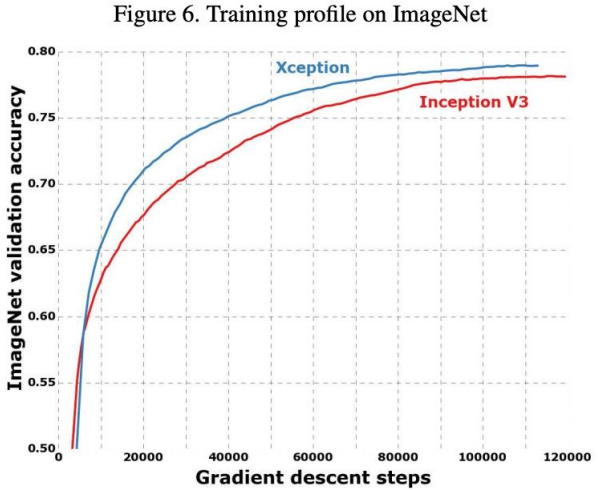


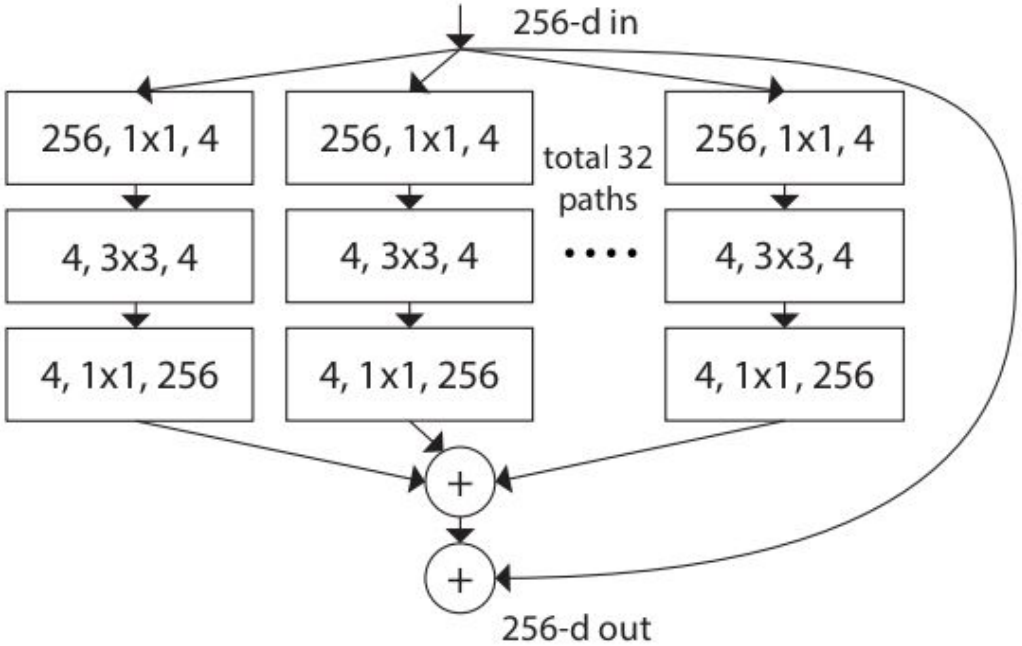
Figure: Sik-Ho Tsang



#XCeption Chollet, François. "Xception: Deep learning with depthwise separable convolutions." CVPR 2017.

ResNext

Depth-wise separable convolutions



facebook research

#ResNext Xie, Saining, Ross Girshick, Piotr Dollár, Zhuowen Tu, and Kaiming He. "Aggregated residual transformations for deep neural networks." CVPR 2017 [code]

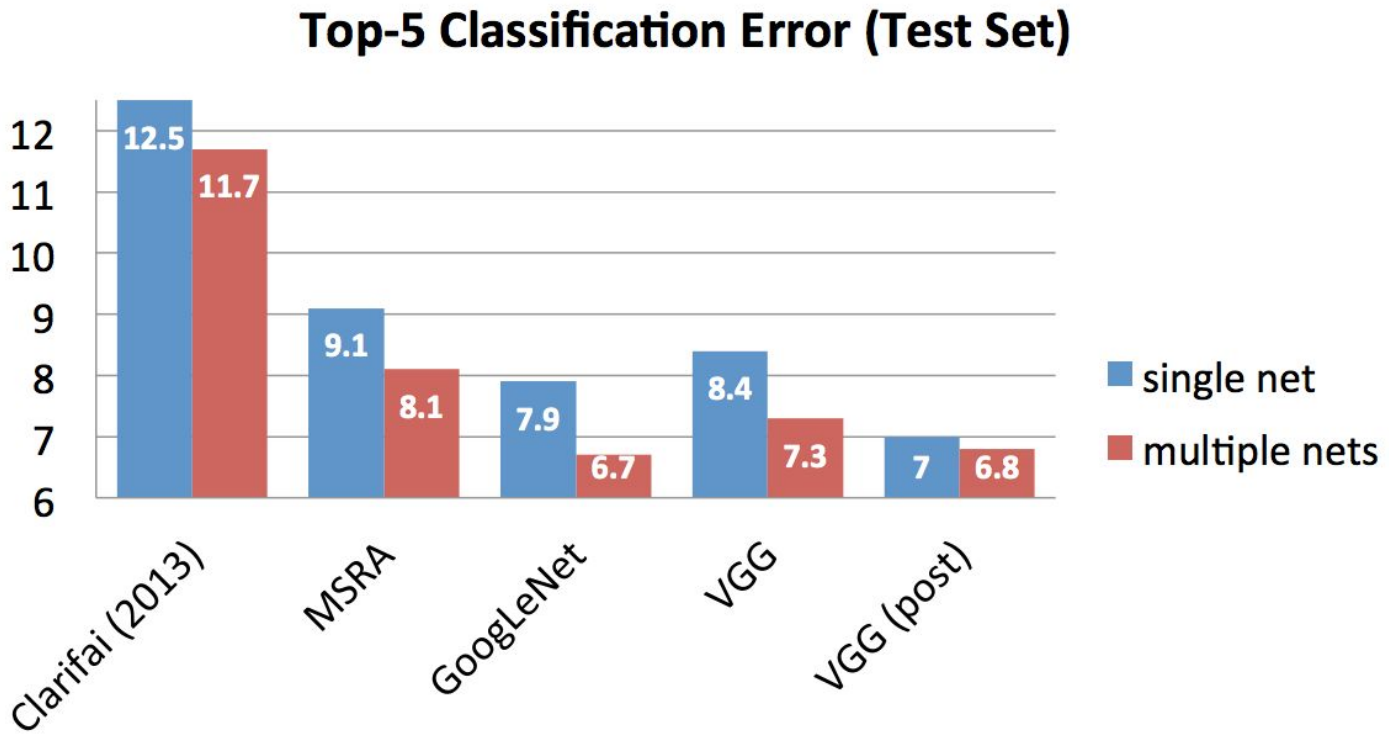
VGG

UNIVERSITY OF OXFORD

Google DeepMind

#VGG Simonyan, Karen, and Andrew Zisserman. "Very deep convolutional networks for large-scale image recognition." *ICLR 2015*. [video] [slides] [project]

VGG

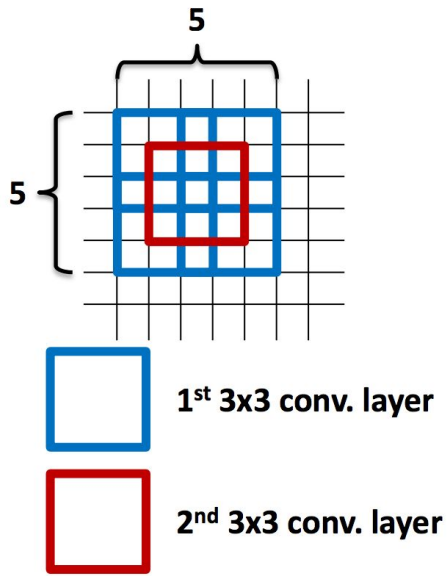


#VGG Simonyan, Karen, and Andrew Zisserman. "Very deep convolutional networks for large-scale image recognition." *ICLR 2015*. [video] [slides] [project]

VGG: Stacked 3x3 convolutions

Why 3x3 layers?

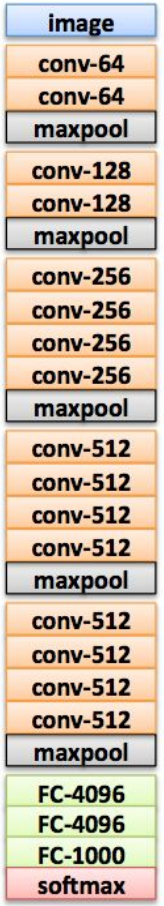
- Stacked conv. layers have a large receptive field
  - two 3x3 layers – 5x5 receptive field
  - three 3x3 layers – 7x7 receptive field
- More non-linearity
- Less parameters to learn
  - ~140M per net



#VGG Simonyan, Karen, and Andrew Zisserman. "Very deep convolutional networks for large-scale image recognition." *ICLR 2015*. [video] [slides] [project]

VGG: Other details

- No poolings between some convolutional layers.
- Convolution strides of 1 (no skipping).



Simonyan, Karen, and Andrew Zisserman. "Very deep convolutional networks for large-scale image recognition." *ICLR 2015*. [video] [slides] [project]

ImageNet Challenge: 2015

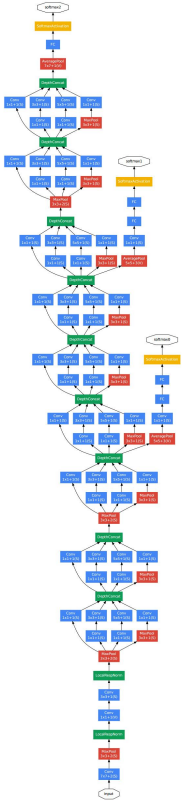
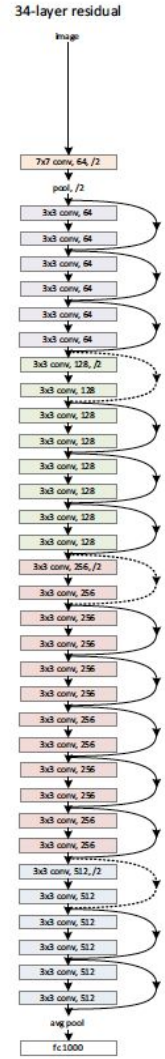
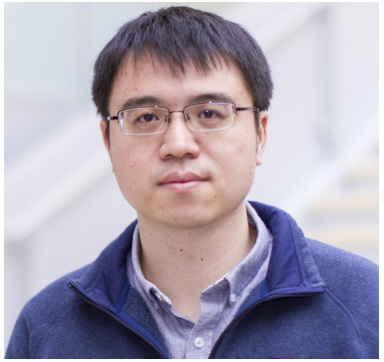


image
conv-64
conv-64
maxpool
conv-128
conv-128
maxpool
conv-256
conv-256
conv-256
conv-256
maxpool
conv-512
conv-512
conv-512
conv-512
maxpool
conv-512
conv-512
conv-512
conv-512
maxpool
FC-4096
FC-4096
FC-1000
softmax

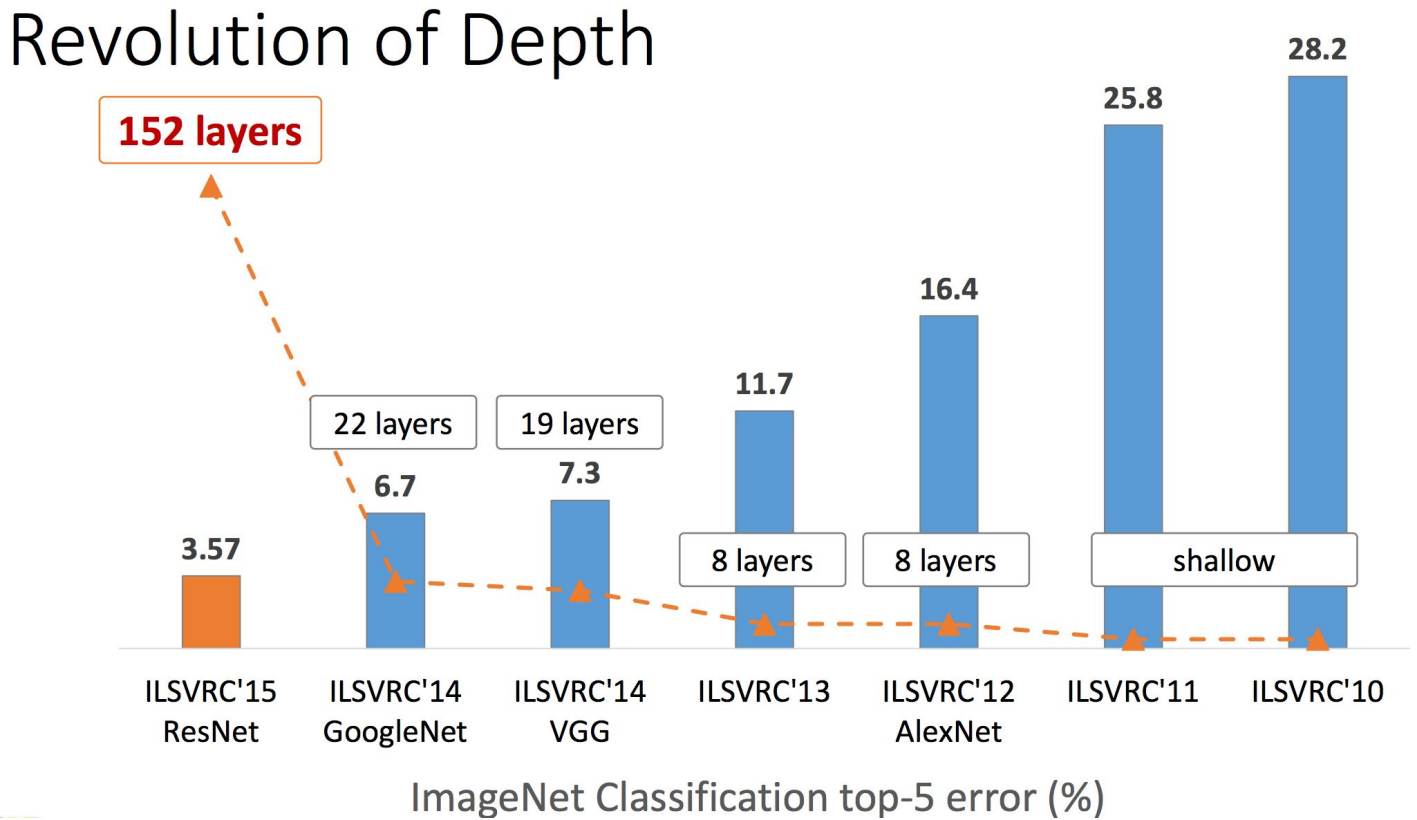




Microsoft Research

3.6% top 5 error…
with 152 layers !!

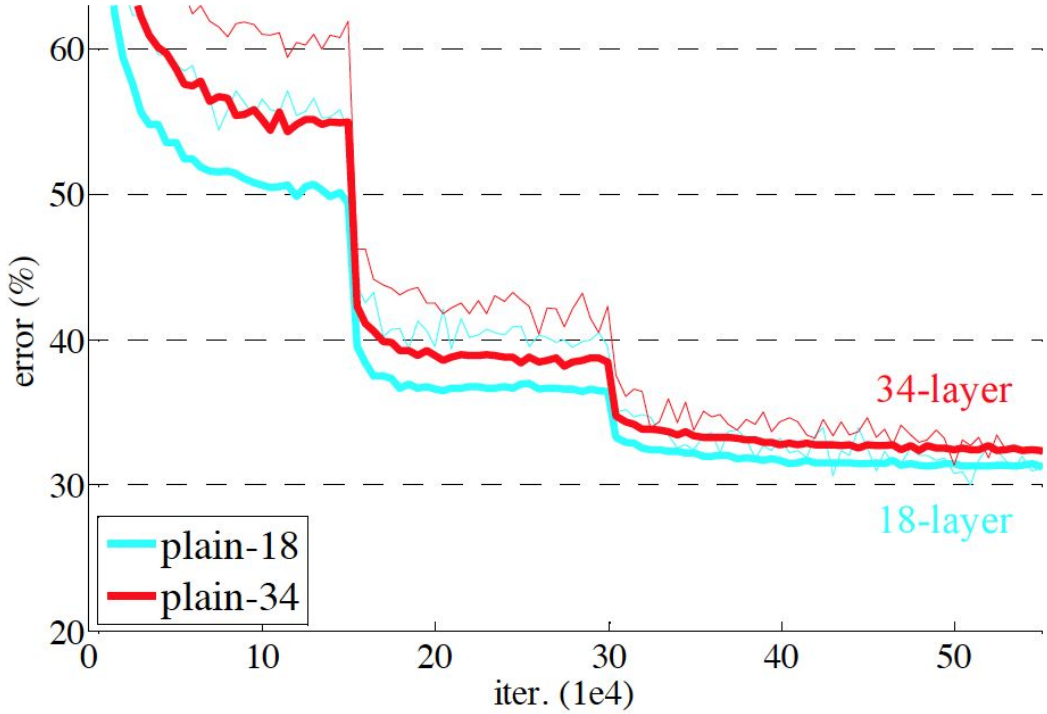
ResNet



He, Kaiming, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. "Deep residual learning for image recognition." CVPR 2016. [slides]

ResNet

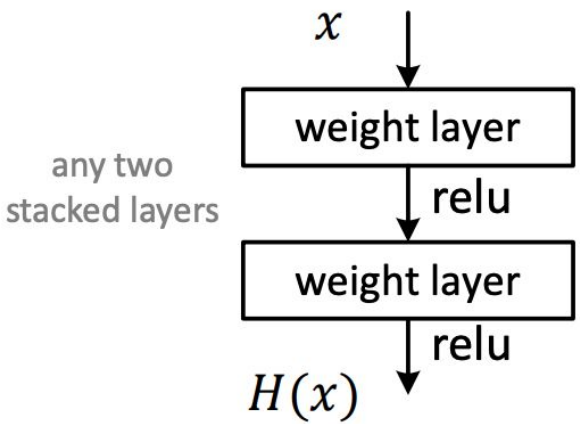
Challenge: Deeper networks (34 is deeper than 18) are more difficult to train.



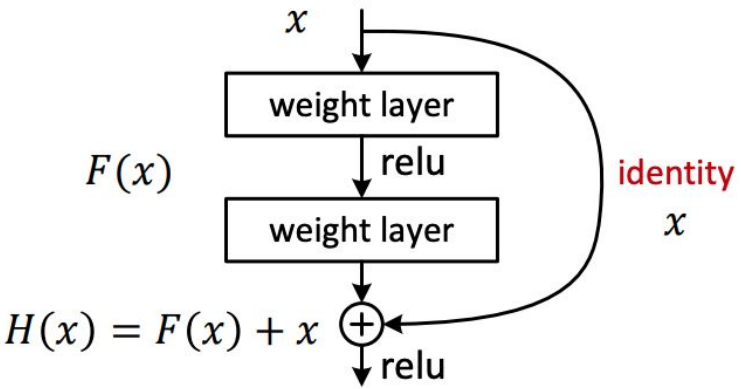
He, Kaiming, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. "Deep residual learning for image recognition." CVPR 2016. [slides]

ResNet

Plain Net



Residual Net



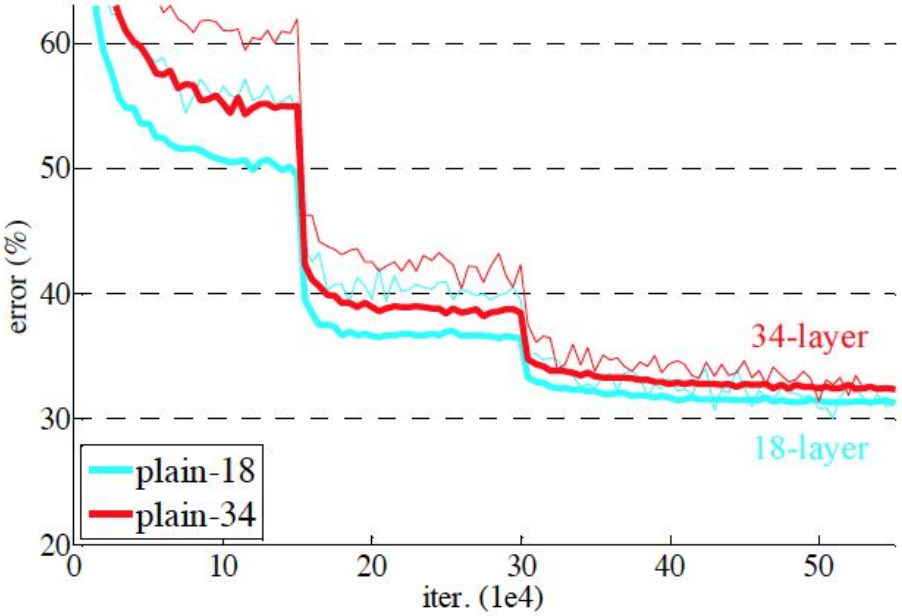
Residual learning: reformulate the layers as learning residual functions with respect to the identity F(x), instead of learning unreferenced functions H(x)

He, Kaiming, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. "Deep residual learning for image recognition." CVPR 2016. [slides]

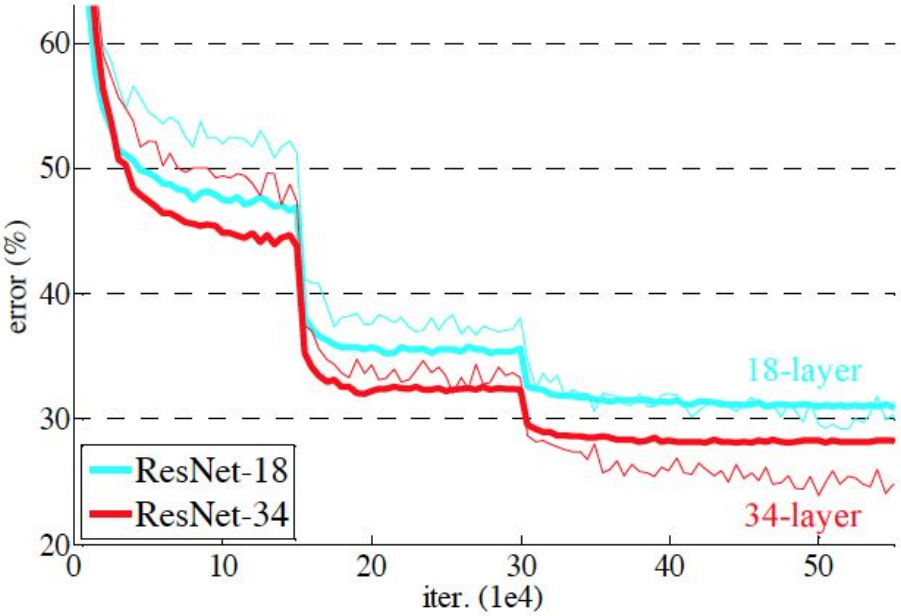
ResNet

Solution: Replace plain layer for residual layers.

Non-Residual connections



Residual connections

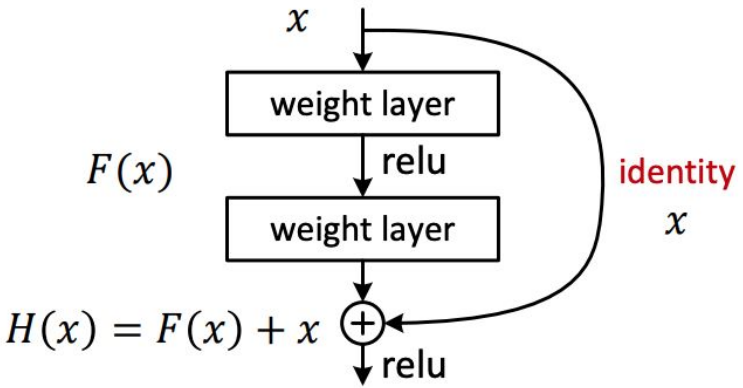


He, Kaiming, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. "Deep residual learning for image recognition." CVPR 2016. [slides]

ResNet

Why ResNets train better ?:

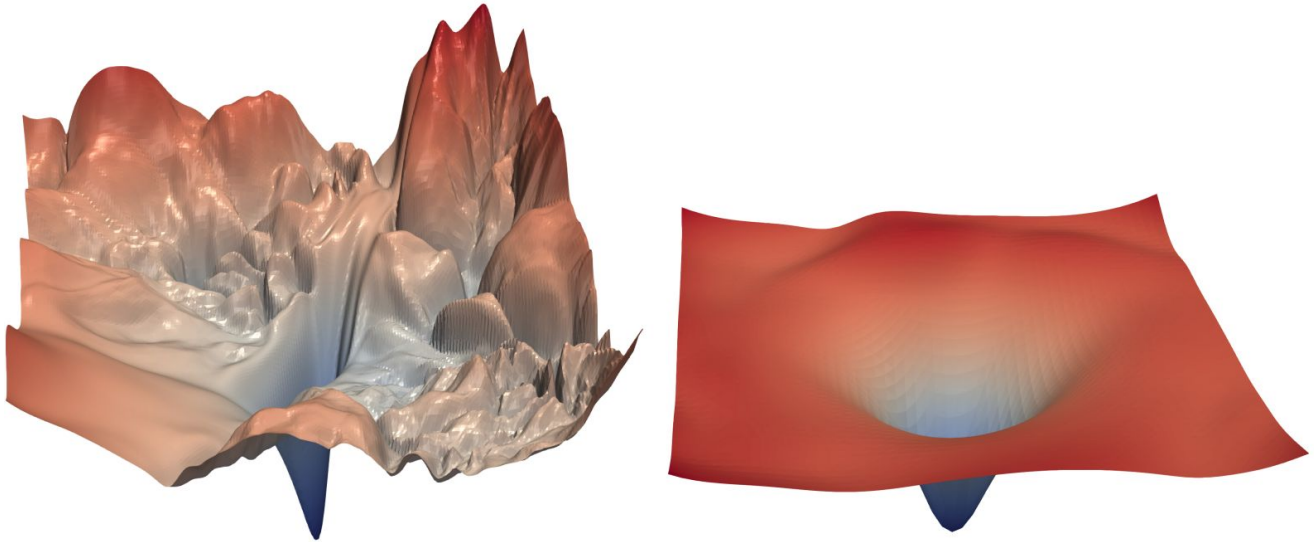
“We hypothesize that it is easier to optimize the residual mapping than to optimize the original, unreferenced mapping. To the extreme, if an identity mapping were optimal (given x return x), it would be easier to push F(x) to zero than to fit an identity mapping by a stack of nonlinear layers”



He, Kaiming, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. "Deep residual learning for image recognition." CVPR 2016. [slides]

ResNet

The loss surfaces of ResNet-56 with/without skip connections



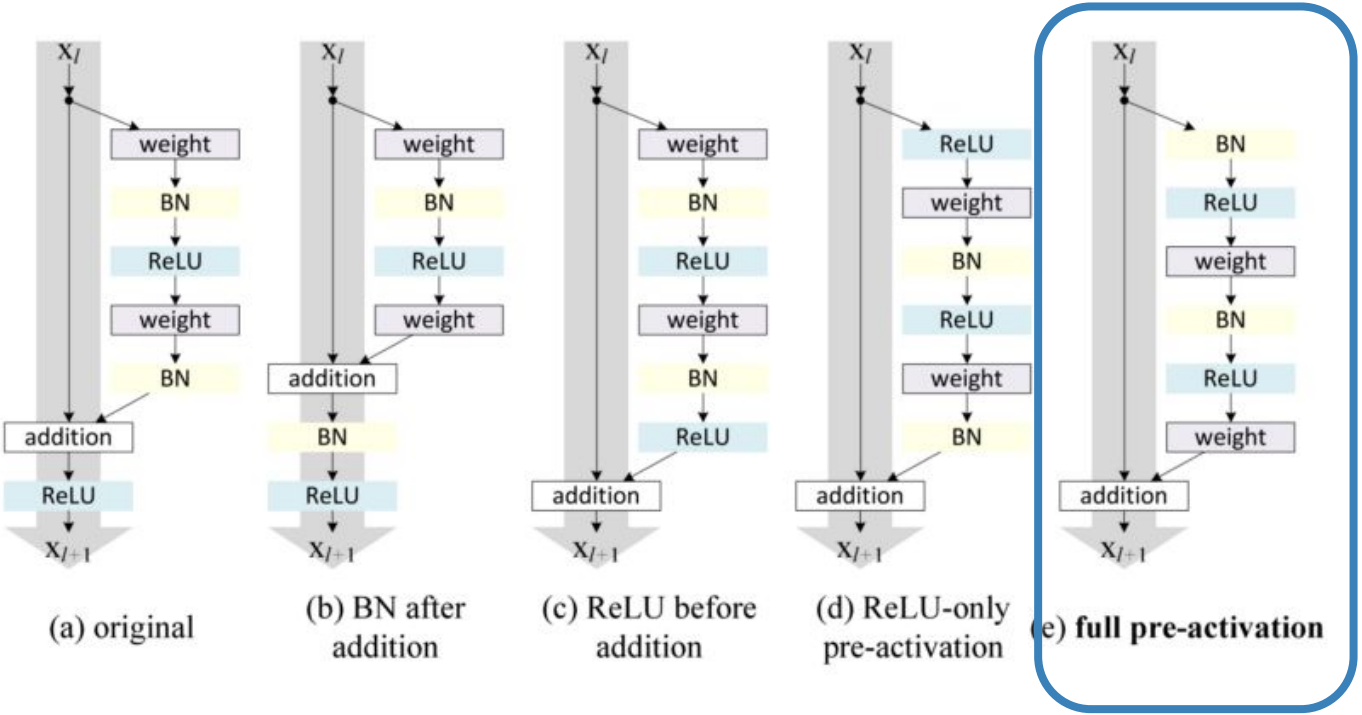
(a) without skip connections

(b) with skip connections

Hao Li, Zheng Xu, Gavin Taylor, Christoph Studer and Tom Goldstein. Visualizing the Loss Landscape of Neural Nets. NeurIPS, 2018. [code] [poster]

ResNet

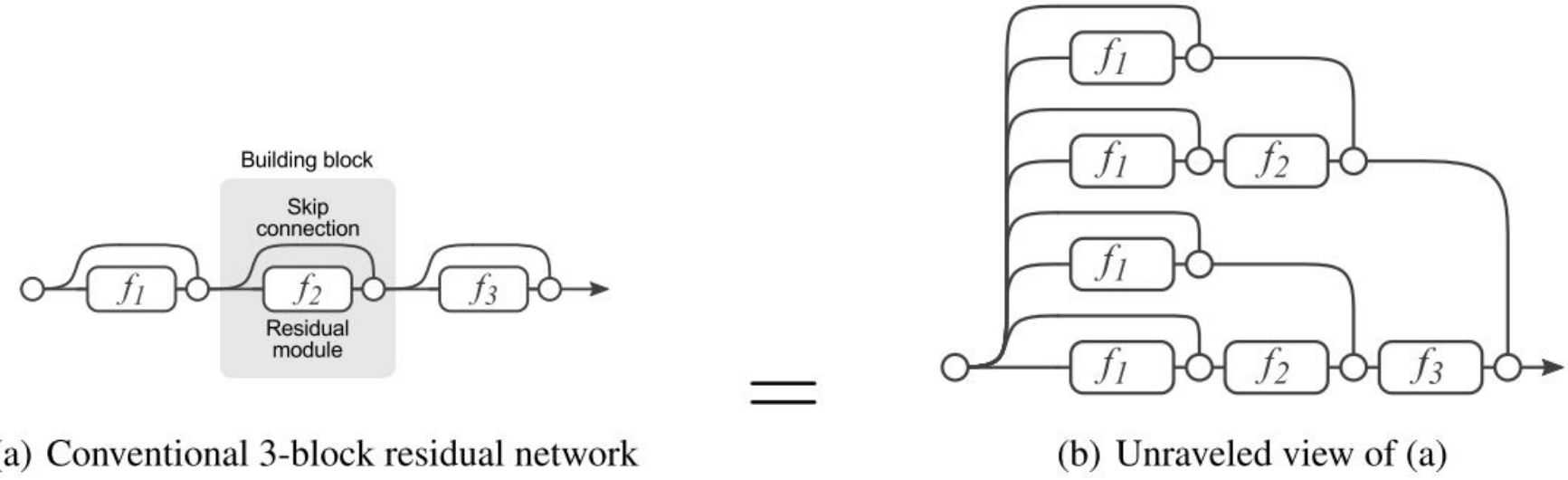
It has been observed that pre-activations with batch normalizations give the best results in general



He, K., Zhang, X., Ren, S., & Sun, J. "Identity mappings in deep residual networks". ECCV 2016.

ResNet

After we unroll the network architecture, it is quite clear that a ResNet architecture with i residual blocks has 2 ** i different paths (because each residual block provides two independent paths).



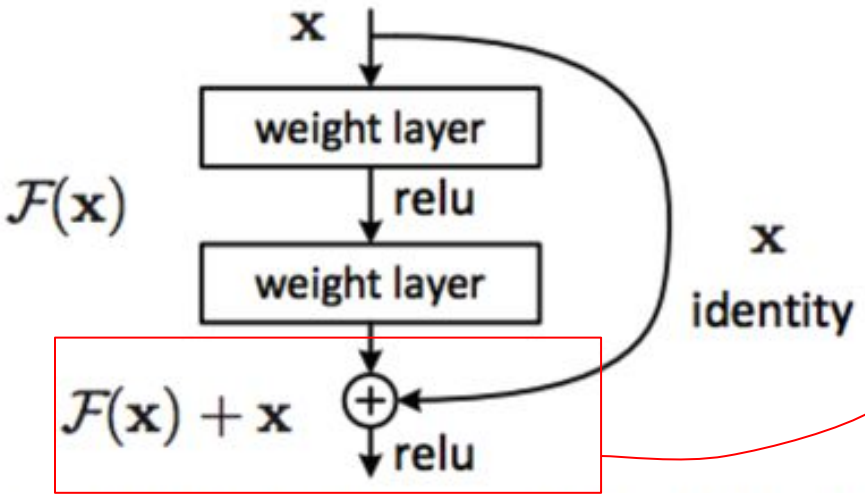
(a) Conventional 3-block residual network

(b) Unraveled view of (a)

Veit, Andreas, Michael J. Wilber, and Serge Belongie. "Residual networks behave like ensembles of relatively shallow networks." NIPS 2016.

ResNet

PyTorch

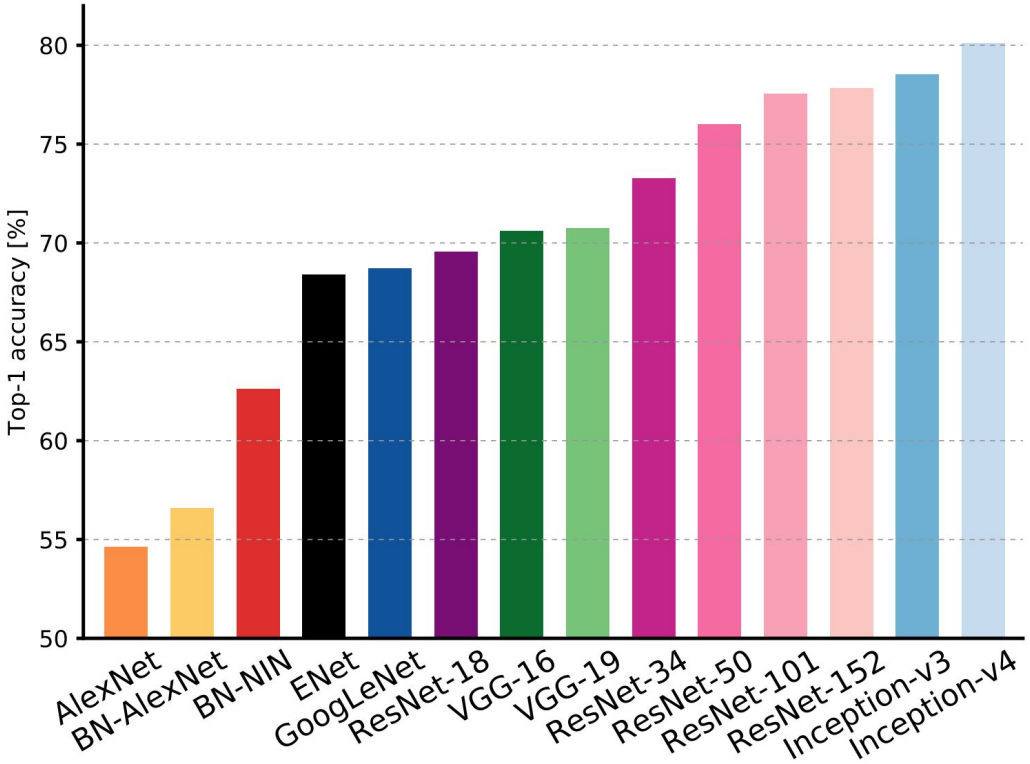



```python
class ResLayer(nn.Module):

  def __init__(self, num_inputs):
    super().__init__()
    self.num_inputs = num_inputs
    num_outputs = num_inputs
    self.num_outputs = num_outputs
    self.conv1 = nn.Sequential(
        nn.Conv2d(num_inputs, num_outputs, 3, padding=1),
        nn.BatchNorm2d(num_outputs),
        nn.ReLU(inplace=True)
    )
    self.conv2 = nn.Sequential(
        nn.Conv2d(num_outputs, num_outputs, 3, padding=1),
        nn.BatchNorm2d(num_outputs),
        nn.ReLU(inplace=True)
    )
    self.out_relu = nn.ReLU(inplace=True)

  def forward(self, x):
    # non-linear processing trunk
    conv1_h = self.conv1(x)
    conv2_h = self.conv2(conv1_h)
    # output is result of res connection + non-linear processing
    y = self.out_relu(x + conv2_h)
    return y

x = torch.randn(1, 64, 100, 100)
print('Input tensor x size: ', x.size())
reslayer = ResLayer(64)
y = reslayer(x)
print('Output tensor y size: ', y.size())
```

```
Input tensor x size:  torch.Size([1, 64, 100, 100])
Output tensor y size:  torch.Size([1, 64, 100, 100])
```

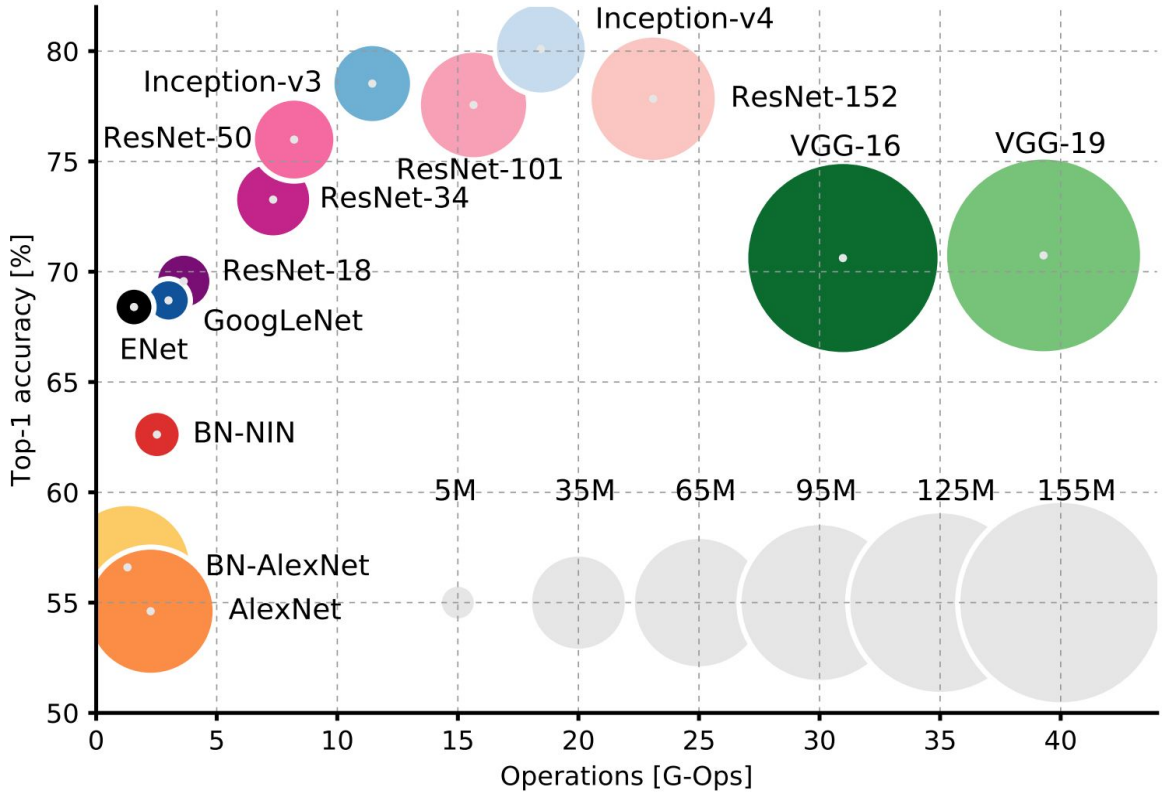

Slide credit: Santiago Pascual (UPC School 2019)

ResNet



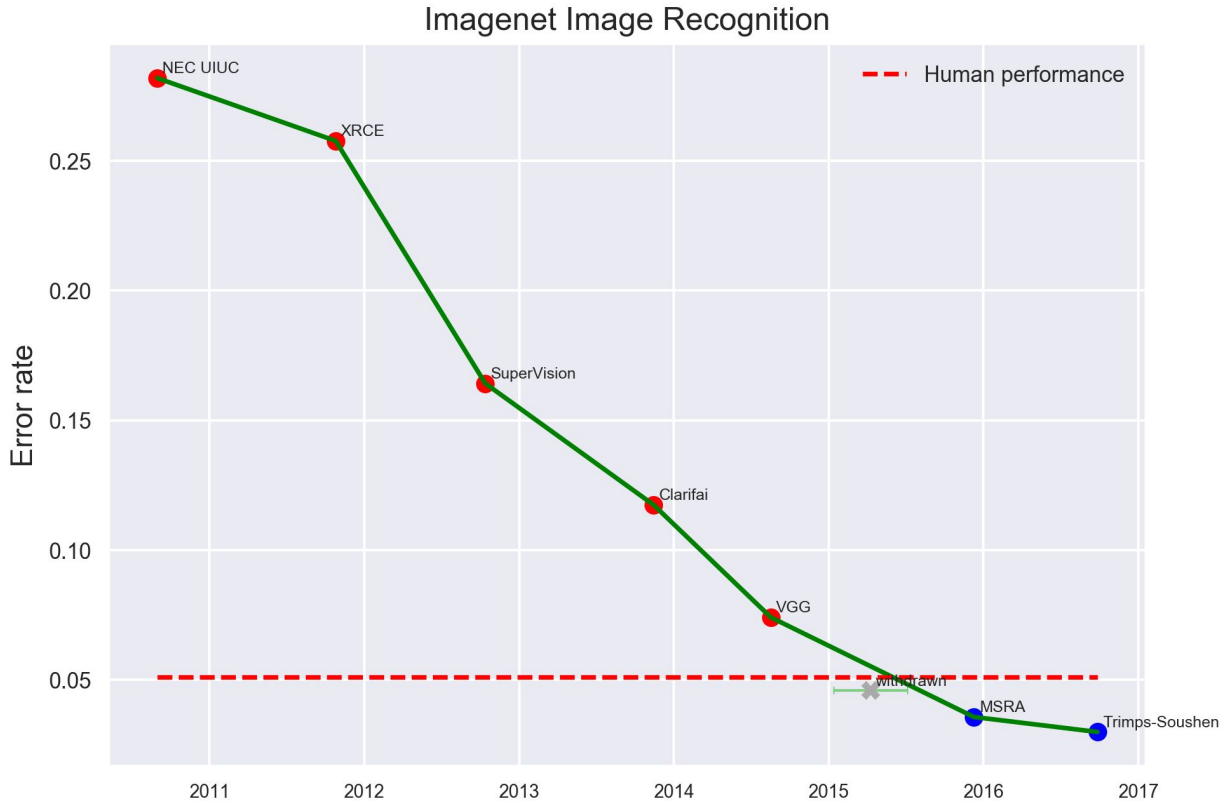
Canziani, Alfredo, Adam Paszke, and Eugenio Culurciello. "An analysis of deep neural network models for practical applications." arXiv preprint arXiv:1605.07678 (2016).

ResNet



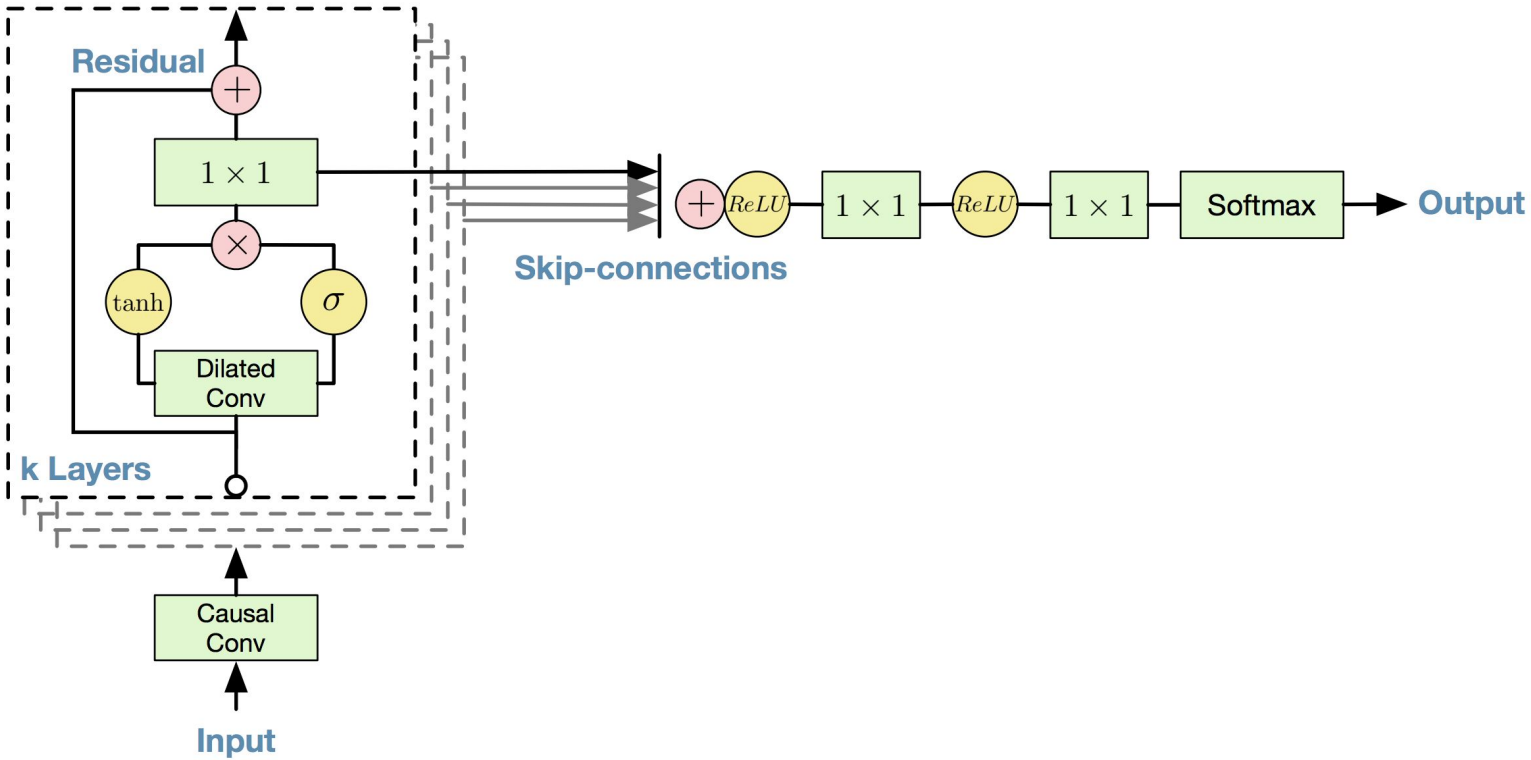
Canziani, Alfredo, Adam Paszke, and Eugenio Culurciello. "An analysis of deep neural network models for practical applications." arXiv preprint arXiv:1605.07678 (2016).

ResNet



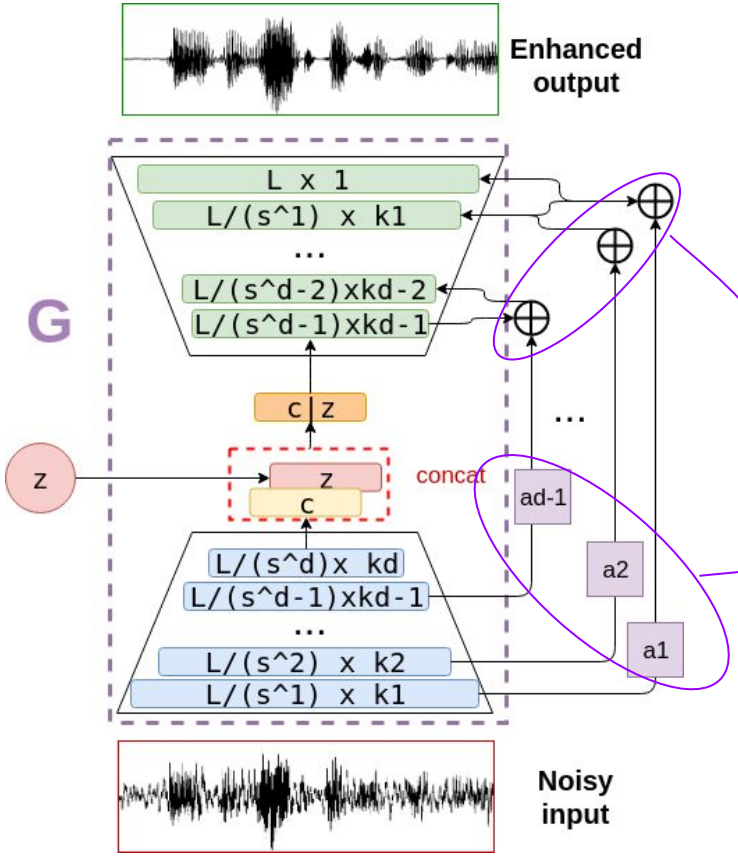
Electronic Frontier Foundation: [“Measuring the Progress of AI Research”](#) (2017)

Skip connections (speech synthesis)



#WaveNet Van Den Oord, Aaron, Sander Dieleman, Heiga Zen, Karen Simonyan, Oriol Vinyals, Alex Graves, Nal Kalchbrenner, Andrew Senior, and Koray Kavukcuoglu. "Wavenet: A generative model for raw audio." Inetrspeech 2016.

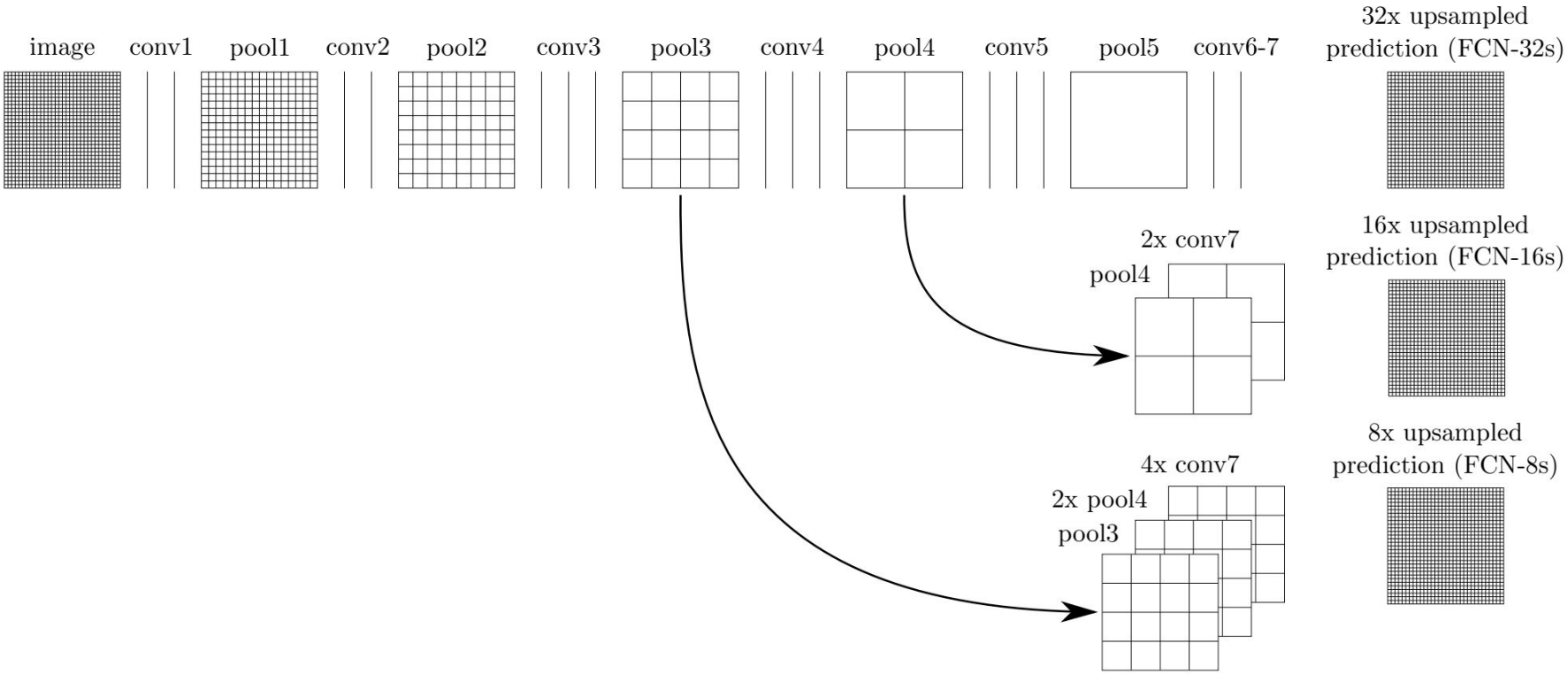
Skip connections (speech denoising)




```python
class Conv1dGenerator(nn.Module):

  def __init__(self, enc_fmaps=[64, 128, 256, 512], kwidth=31,
               pooling=4):
    super().__init__()
    self.enc = nn.ModuleList()
    ninp = 1
    for enc_fmap in enc_fmaps:
      self.enc.append(DownConv1dBlock(ninp, enc_fmap, kwidth, pooling))
      ninp = enc_fmap

    self.dec = nn.ModuleList()
    # revert encoder feature maps
    dec_fmaps = enc_fmaps[::-1][1:] + [1]
    act = True
    for di, dec_fmap in enumerate(dec_fmaps, start=1):
      if di >= len(dec_fmaps):
        # last decoder layer has no activation
        act = False
      self.dec.append(UpConv1dBlock(ninp, dec_fmap, kwidth, pooling, act=act))
      ninp = dec_fmap

  def forward(self, x):
    skips = []
    h = x
    for ei, enc_layer in enumerate(self.enc, start=1):
      h = enc_layer(h)
      if ei < len(self.enc):
        skips.append(h)
    # now decode

    for di, dec_layer in enumerate(self.dec, start=1):
      if di > 1:
        # sum skip connection
        skip_h = skips.pop(-1)
        h = h + skip_h
      h = dec_layer(h)
    y = h
    return y

G = Conv1dGenerator()
x = torch.randn(1, 1, 8192)
print('Input tensor x size: ', x.size())
y = G(x)
print('Output tensor y size: ', y.size())
```

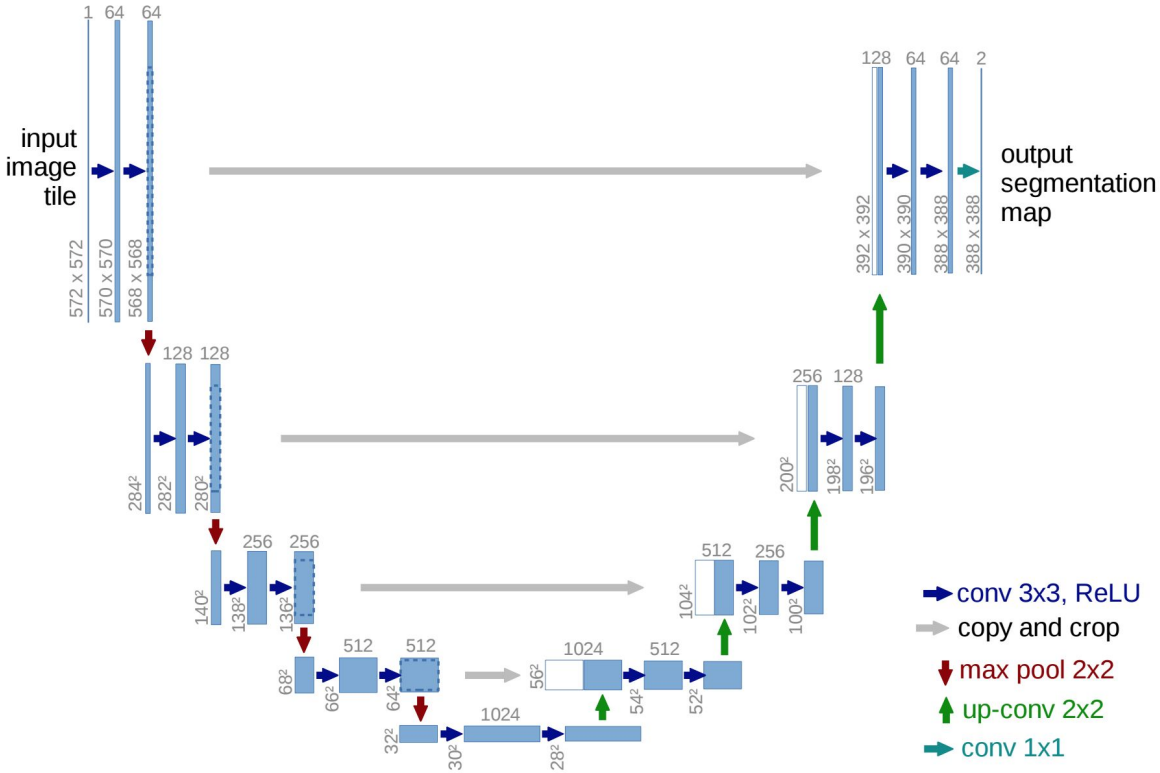
Pascual, S., Bonafonte, A., & Serrà, J. "SEGAN: Speech Enhancement Generative Adversarial Network". Interspeech 2017. [code]

Skip connections (image segmentation)



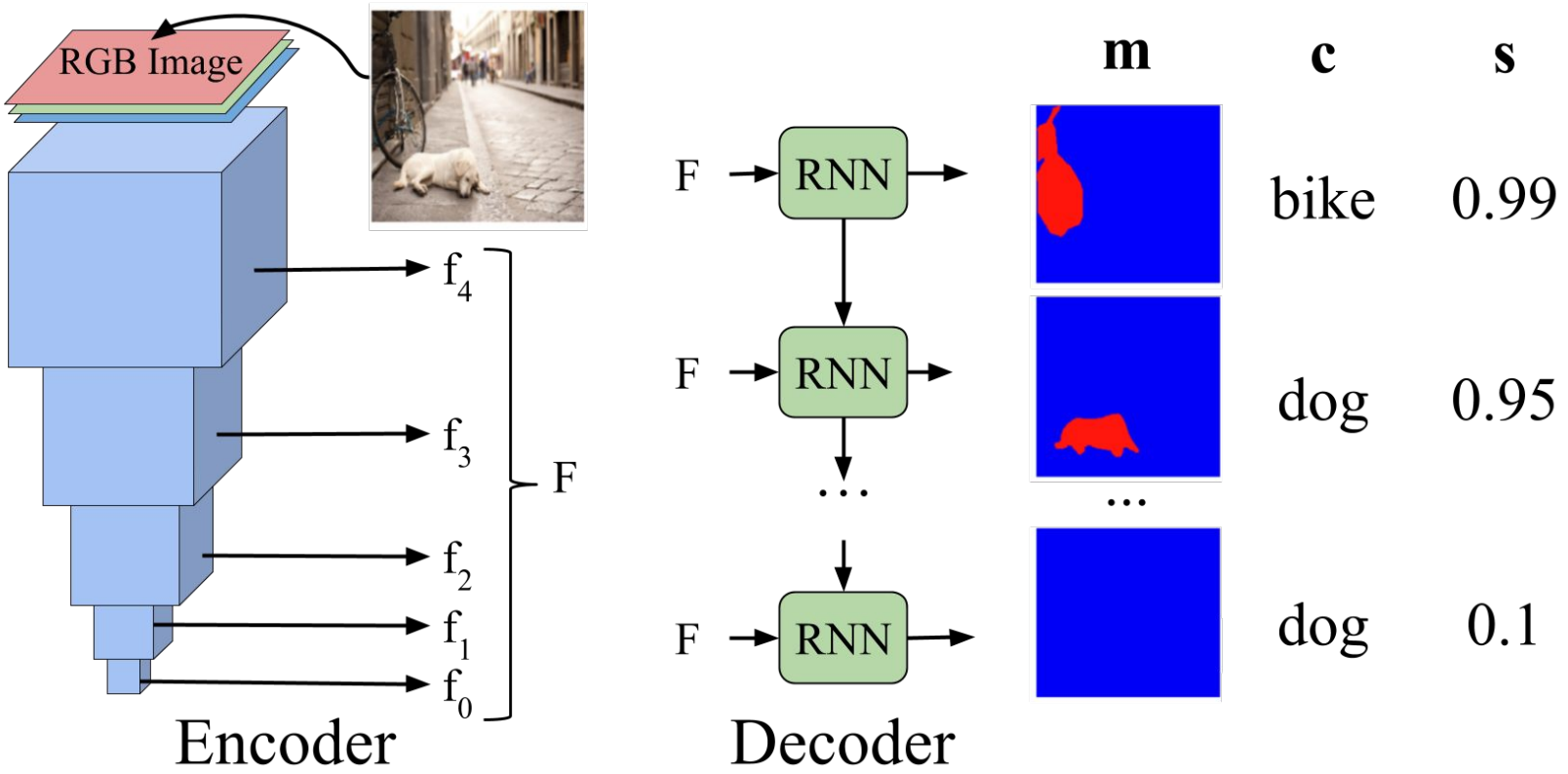
#FCN Long, Jonathan, Evan Shelhamer, and Trevor Darrell. "Fully convolutional networks for semantic segmentation." CVPR 2015. [code]

Skip connections to intermediate layers (image segmentation)



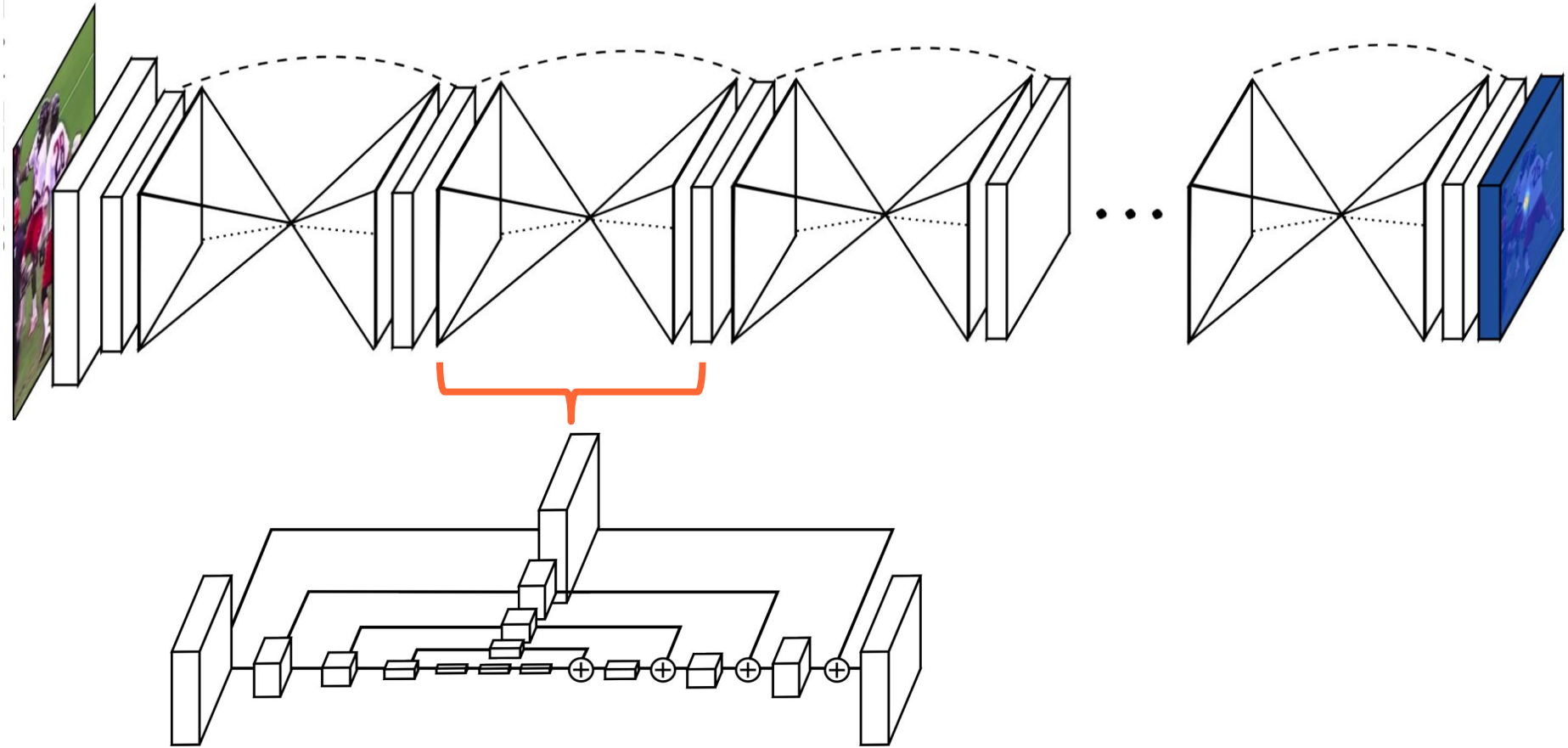
#U-Net Ronneberger, Olaf, Philipp Fischer, and Thomas Brox. "U-net: Convolutional networks for biomedical image segmentation." MICCAI 2015

Skip connections to intermediate layers (image segmentation)



#RSIS Amaia Salvador, Míriam Bellver, Víctor Campos, Manel Baradad, Ferran Marqués, Jordi Torres, Xavier Giró-i-Nieto, "Recurrent Neural Networks for Semantic Instance Segmentation". NIPS Women in Machine Learning 2017.

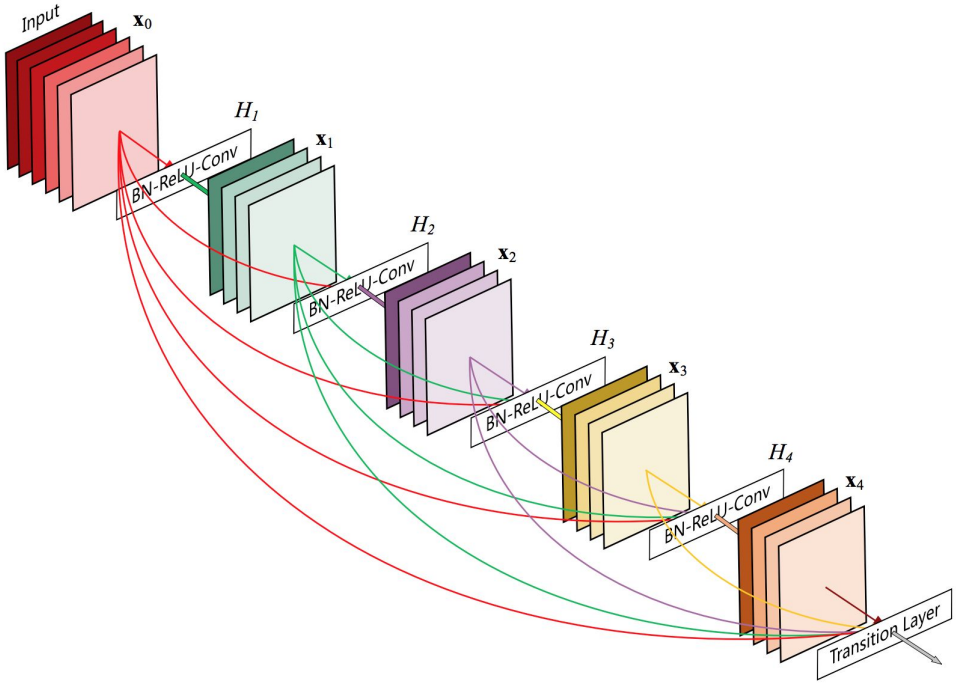
Stacked Hourglass Networks



Newell, Alejandro, Kaiyu Yang, and Jia Deng. "Stacked hourglass networks for human pose estimation." ECCV 2016.

Dense Blocks

The input of each layer consists of the feature maps of all earlier layer, and its output is passed to each subsequent layer.

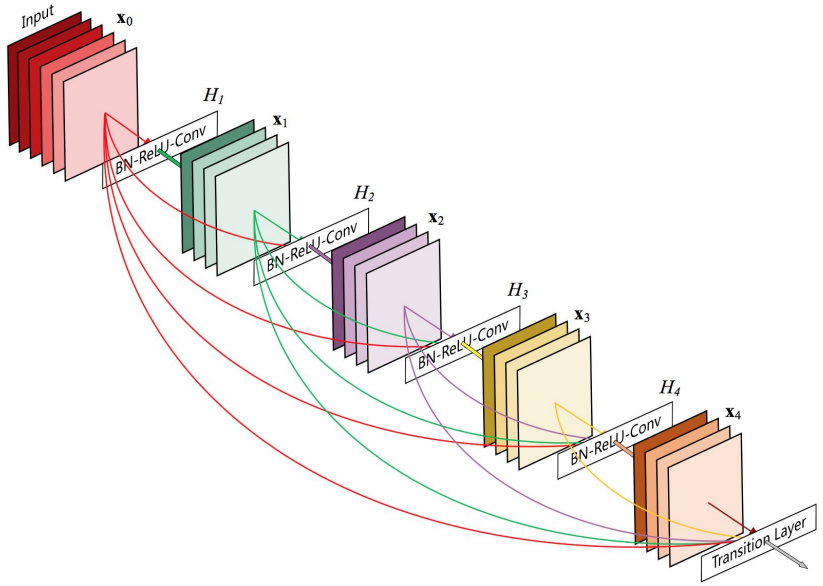


#DenseNet Huang, Gao, Zhuang Liu, Kilian Q. Weinberger, and Laurens van der Maaten. "Densely connected convolutional networks." CVPR 2017. [code]

Dense Blocks

Growth rate (k): If each function H_l produces k feature maps, it follows that the l^{th} layer has $k_0 + k \times (l-1)$ input feature-maps, where k_0 is the number of channels in the input layer.

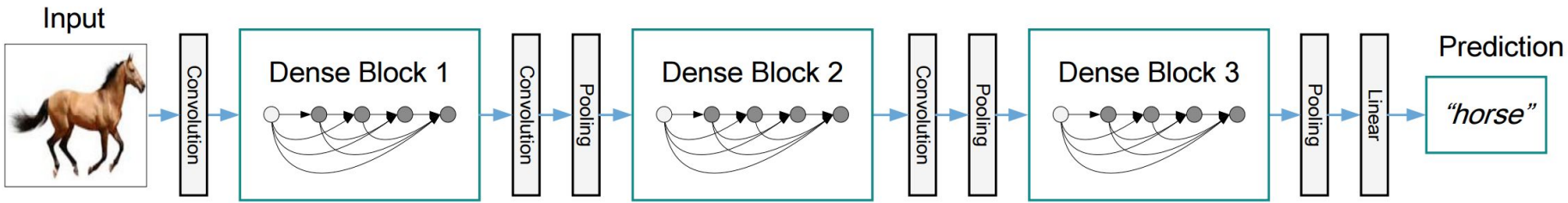
Layer (l)	Growth rate (k)	# input f. maps
0	k_0=5	-
1	4	5
2	4	9
3	4	13
4	4	17



#DenseNet Huang, Gao, Zhuang Liu, Kilian Q. Weinberger, and Laurens van der Maaten. "Densely connected convolutional networks." CVPR 2017. [code]

Dense Blocks

DenseNet is divided into multiple densely connected dense blocks to facilitate spatial **pooling**.



#DenseNet Huang, Gao, Zhuang Liu, Kilian Q. Weinberger, and Laurens van der Maaten. "Densely connected convolutional networks." CVPR 2017. [code]

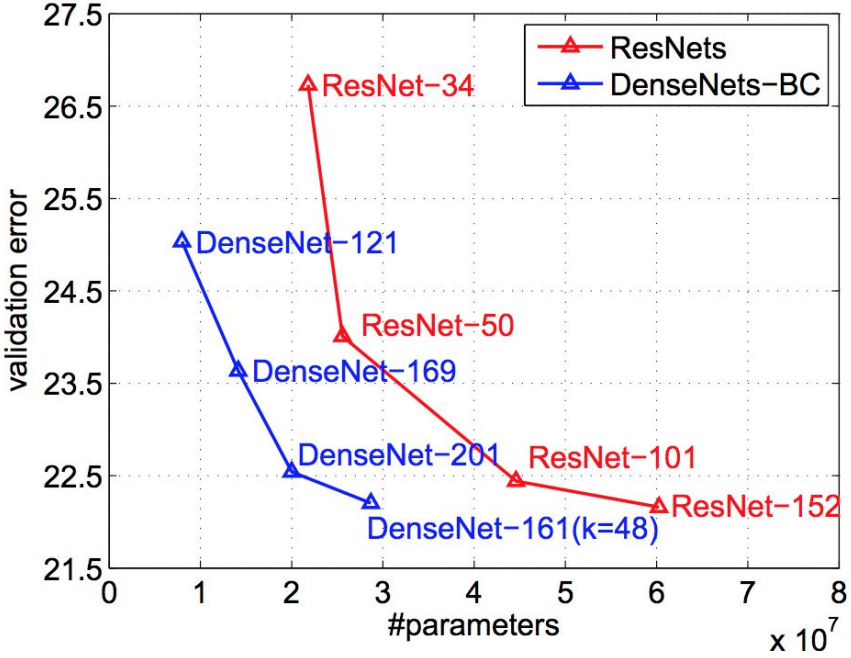
DenseNet

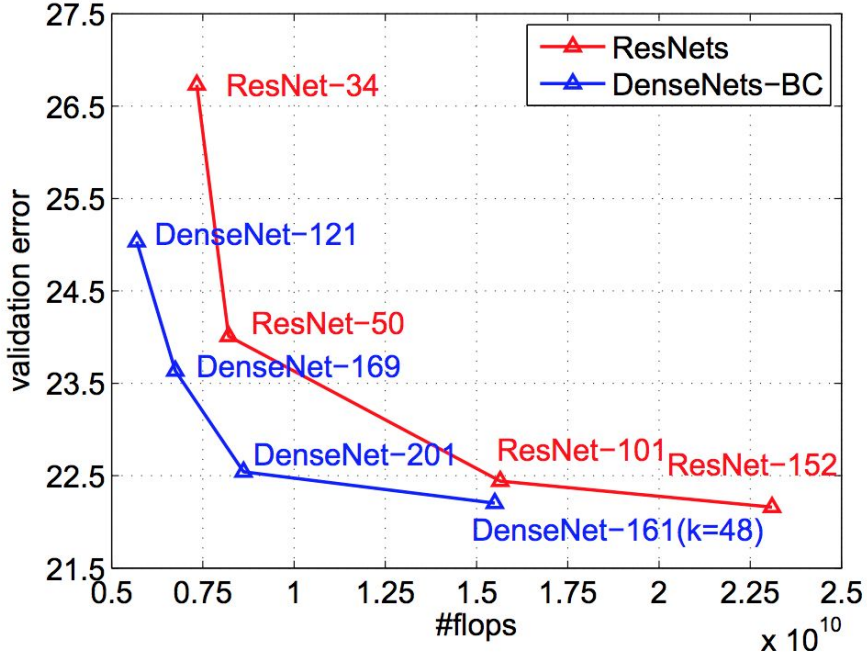
Layers	Output Size	DenseNet-121	DenseNet-169	DenseNet-201	DenseNet-264
Convolution	112 × 112	7 × 7 conv, stride 2			
Pooling	56 × 56	3 × 3 max pool, stride 2			
Dense Block (1)	56 × 56	$\begin{bmatrix} 1 \times 1 \text{ conv} \\ 3 \times 3 \text{ conv} \end{bmatrix} \times 6$	$\begin{bmatrix} 1 \times 1 \text{ conv} \\ 3 \times 3 \text{ conv} \end{bmatrix} \times 6$	$\begin{bmatrix} 1 \times 1 \text{ conv} \\ 3 \times 3 \text{ conv} \end{bmatrix} \times 6$	$\begin{bmatrix} 1 \times 1 \text{ conv} \\ 3 \times 3 \text{ conv} \end{bmatrix} \times 6$
Transition Layer (1)	56 × 56	1 × 1 conv			
	28 × 28	2 × 2 average pool, stride 2			
Dense Block (2)	28 × 28	$\begin{bmatrix} 1 \times 1 \text{ conv} \\ 3 \times 3 \text{ conv} \end{bmatrix} \times 12$	$\begin{bmatrix} 1 \times 1 \text{ conv} \\ 3 \times 3 \text{ conv} \end{bmatrix} \times 12$	$\begin{bmatrix} 1 \times 1 \text{ conv} \\ 3 \times 3 \text{ conv} \end{bmatrix} \times 12$	$\begin{bmatrix} 1 \times 1 \text{ conv} \\ 3 \times 3 \text{ conv} \end{bmatrix} \times 12$
Transition Layer (2)	28 × 28	1 × 1 conv			
	14 × 14	2 × 2 average pool, stride 2			
Dense Block (3)	14 × 14	$\begin{bmatrix} 1 \times 1 \text{ conv} \\ 3 \times 3 \text{ conv} \end{bmatrix} \times 24$	$\begin{bmatrix} 1 \times 1 \text{ conv} \\ 3 \times 3 \text{ conv} \end{bmatrix} \times 32$	$\begin{bmatrix} 1 \times 1 \text{ conv} \\ 3 \times 3 \text{ conv} \end{bmatrix} \times 48$	$\begin{bmatrix} 1 \times 1 \text{ conv} \\ 3 \times 3 \text{ conv} \end{bmatrix} \times 64$
Transition Layer (3)	14 × 14	1 × 1 conv			
	7 × 7	2 × 2 average pool, stride 2			
Dense Block (4)	7 × 7	$\begin{bmatrix} 1 \times 1 \text{ conv} \\ 3 \times 3 \text{ conv} \end{bmatrix} \times 16$	$\begin{bmatrix} 1 \times 1 \text{ conv} \\ 3 \times 3 \text{ conv} \end{bmatrix} \times 32$	$\begin{bmatrix} 1 \times 1 \text{ conv} \\ 3 \times 3 \text{ conv} \end{bmatrix} \times 32$	$\begin{bmatrix} 1 \times 1 \text{ conv} \\ 3 \times 3 \text{ conv} \end{bmatrix} \times 48$
Classification Layer	1 × 1	7 × 7 global average pool			
		1000D fully-connected, softmax			

Table 1: DenseNet architectures for ImageNet. The growth rate for all the networks is $k = 32$. Note that each "conv" layer shown in the table corresponds the sequence BN-ReLU-Conv.

#DenseNet Huang, Gao, Zhuang Liu, Kilian Q. Weinberger, and Laurens van der Maaten. "Densely connected convolutional networks." CVPR 2017. [code]

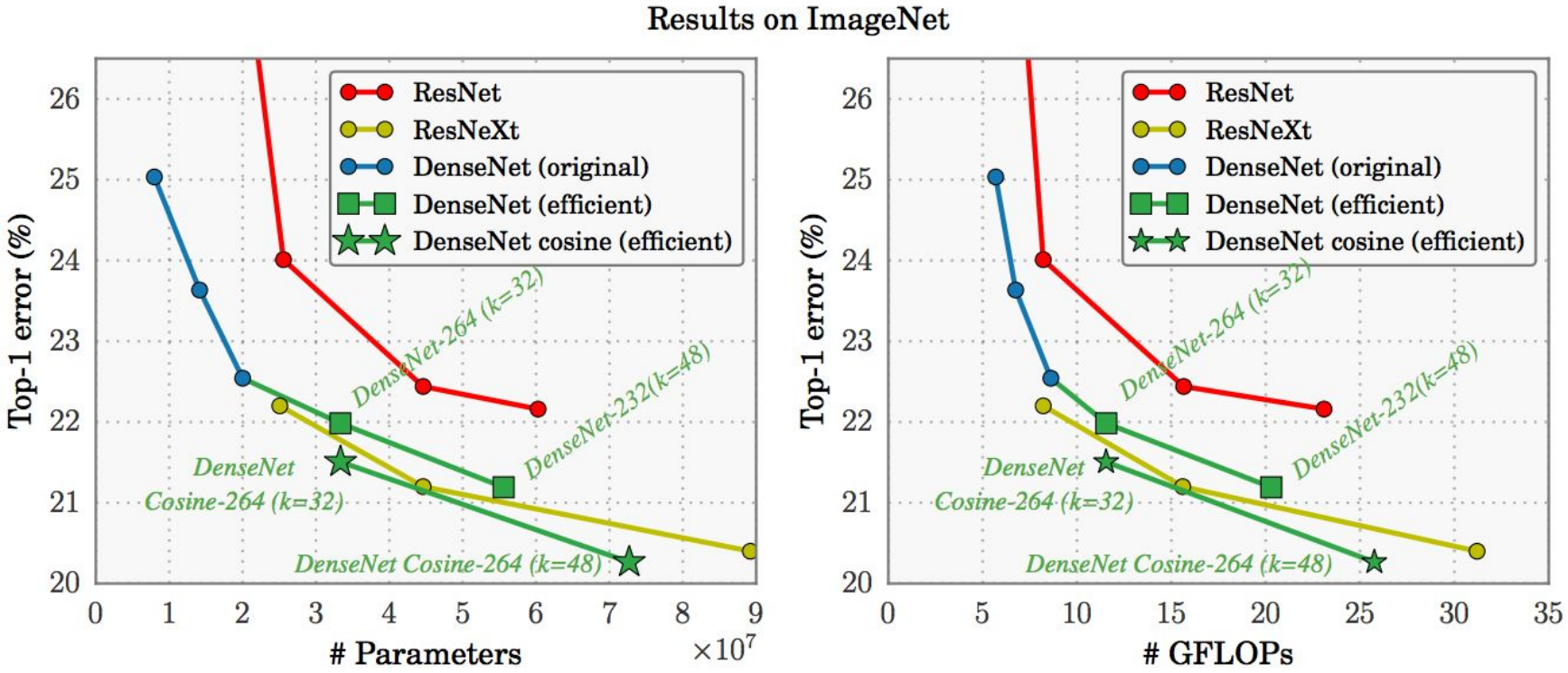
DenseNet





#DenseNet Huang, Gao, Zhuang Liu, Kilian Q. Weinberger, and Laurens van der Maaten. "Densely connected convolutional networks." CVPR 2017. [code]

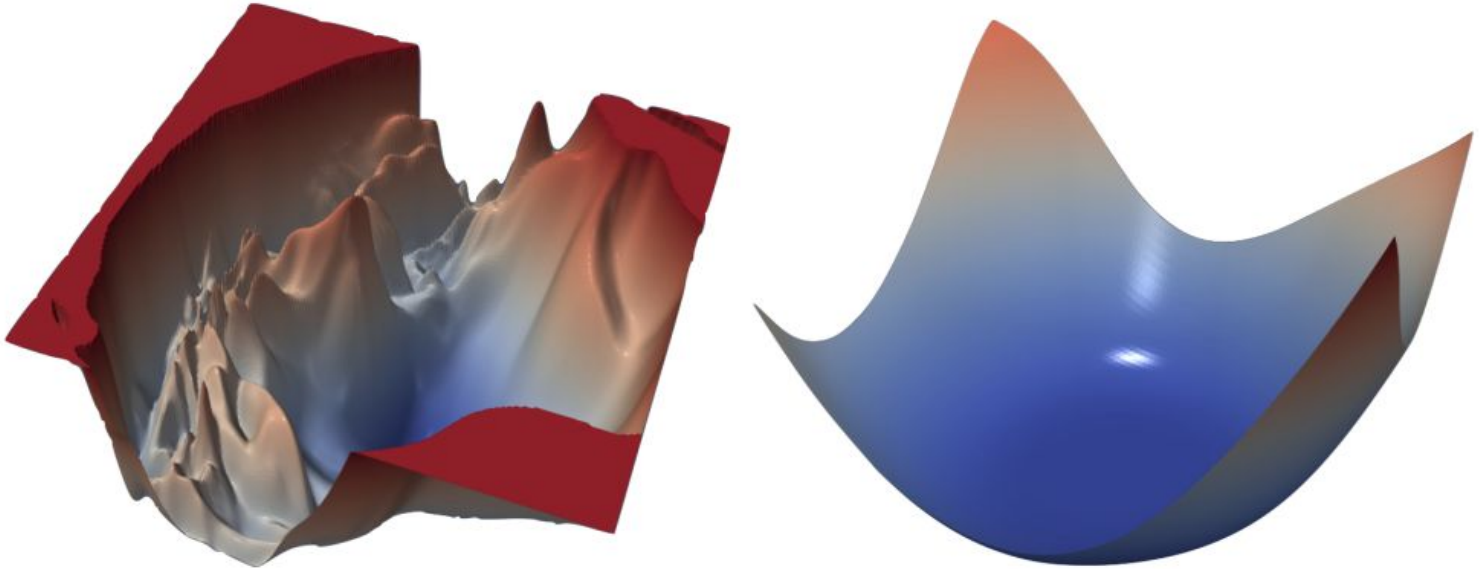
DenseNet



#DenseNet Huang, Gao, Zhuang Liu, Kilian Q. Weinberger, and Laurens van der Maaten. "Densely connected convolutional networks." CVPR 2017. [code]

DenseNets

The loss surfaces of ResNet-110-noshort and DenseNet for CIFAR-10.

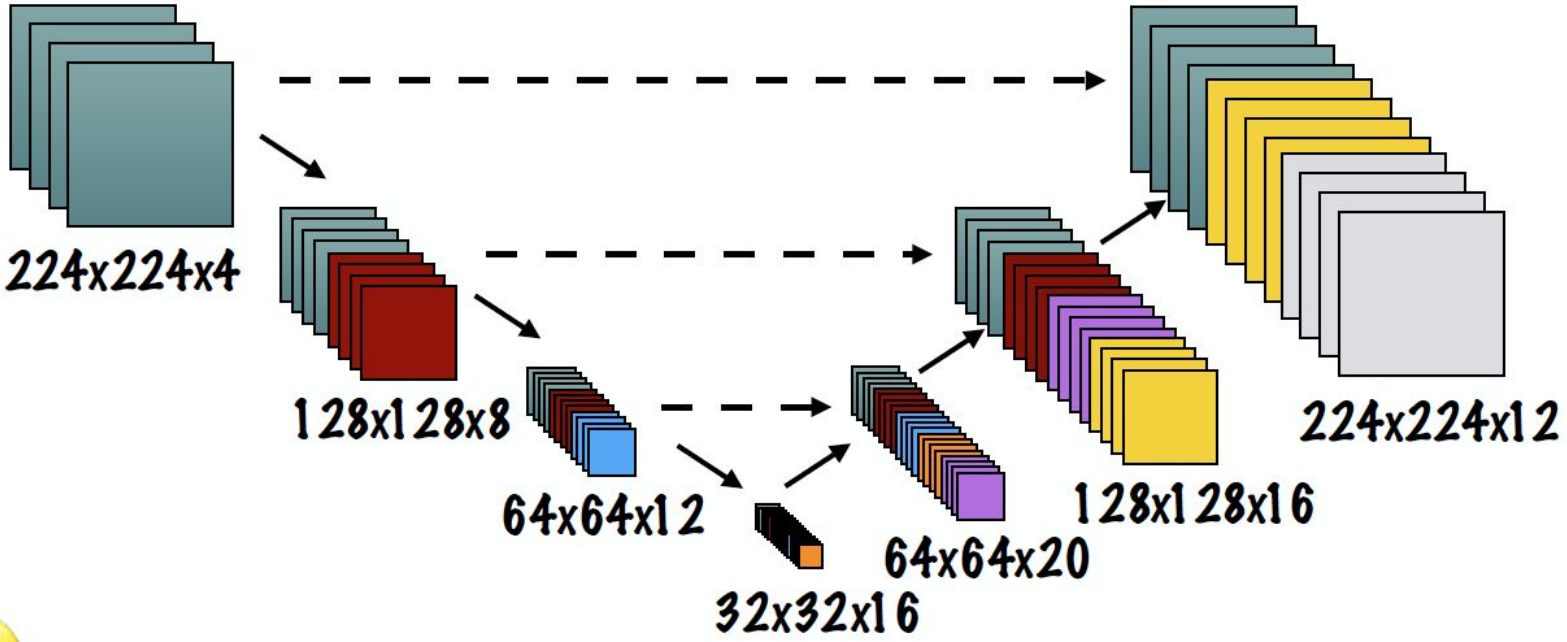


(a) ResNet-110, no skip connections

(b) DenseNet, 121 layers

Hao Li, Zheng Xu, Gavin Taylor, Christoph Studer and Tom Goldstein. Visualizing the Loss Landscape of Neural Nets. NeurIPS, 2018. [code] [poster]

DenseNets for Image Segmentation



Jégou, Simon, Michal Drozdzal, David Vazquez, Adriana Romero, and Yoshua Bengio. "The One Hundred Layers Tiramisu: Fully Convolutional DenseNets for Semantic Segmentation." CVPRW 2017 [code] [slides]

Learn more on Residual Networks

Summary

Vincent Fung, “An Overview of ResNet and its Variants” (2017)

Papers

Zagoruyko, S., & Komodakis, N. (2016). Wide residual networks. arXiv preprint arXiv:1605.07146.

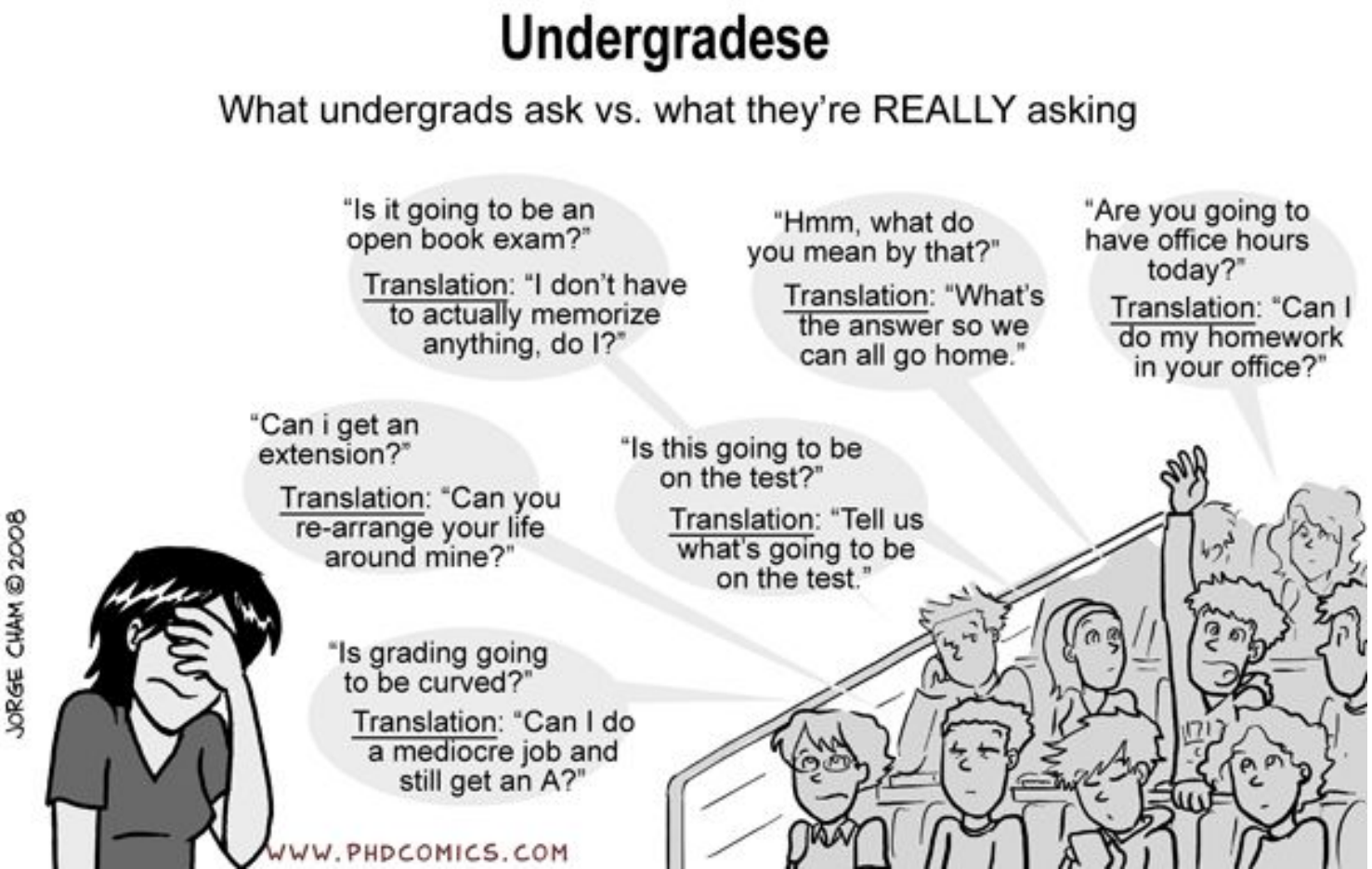
Xie, Saining, Ross Girshick, Piotr Dollár, Zhuowen Tu, and Kaiming He. "Aggregated residual transformations for deep neural networks." CVPR 2017.

Han, Dongyoon, Jiwhan Kim, and Junmo Kim. "Deep pyramidal residual networks." CVPR 2017.

Yu, Fisher, Vladlen Koltun, and Thomas Funkhouser. "Dilated residual networks." CVPR 2017.

Wu, Zifeng, Chunhua Shen, and Anton Van Den Hengel. "Wider or deeper: Revisiting the resnet model for visual recognition." Pattern Recognition 90 (2019): 119-133.

Questions


Undergradese
What undergrads ask vs. what they're REALLY asking
"Is it going to be an open book exam?"
Translation: "I don't have to actually memorize anything, do I?"
"Hmm, what do you mean by that?"
Translation: "What's the answer so we can all go home."
"Are you going to have office hours today?"
Translation: "Can I do my homework in your office?"
"Can i get an extension?"
Translation: "Can you re-arrange your life around mine?"
"Is this going to be on the test?"
Translation: "Tell us what's going to be on the test."
"Is grading going to be curved?"
Translation: "Can I do a mediocre job and still get an A?"
JORGE CHAM © 2008
WWW.PHDCOMICS.COM