Wide Residual Networks

Beyond ResNet

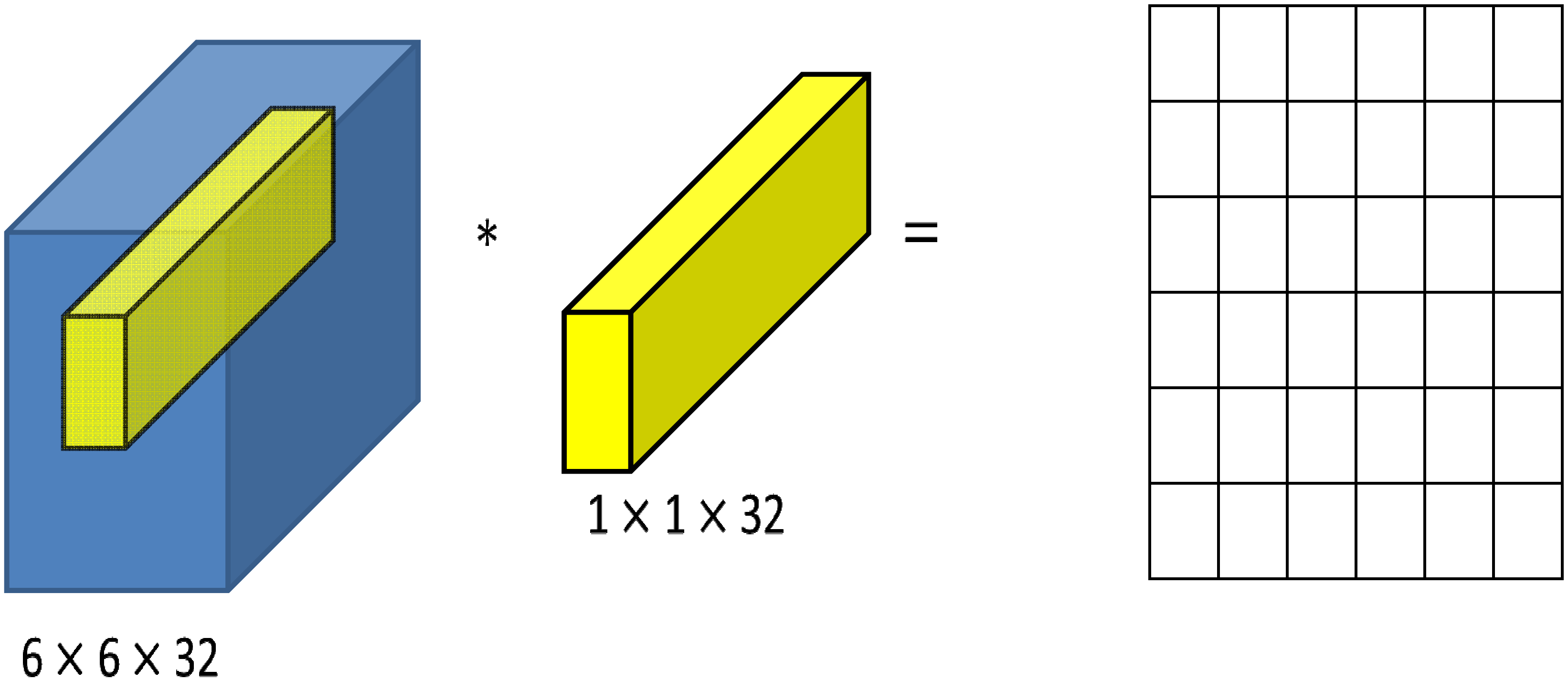
Some slides were adated/taken from various sources, including Andrew Ng's Coursera Lectures, CS231n: Convolutional Neural Networks for Visual Recognition lectures, Stanford University CS Waterloo Canada lectures, Aykut Erdem, et.al. tutorial on Deep Learning in Computer Vision, Ismini Lourentzou's lecture slide on "Introduction to Deep Learning", Ramprasaath's lecture slides, and many more. We thankfully acknowledge them. Students are requested to use this material for their study only and NOT to distribute it.

Topics

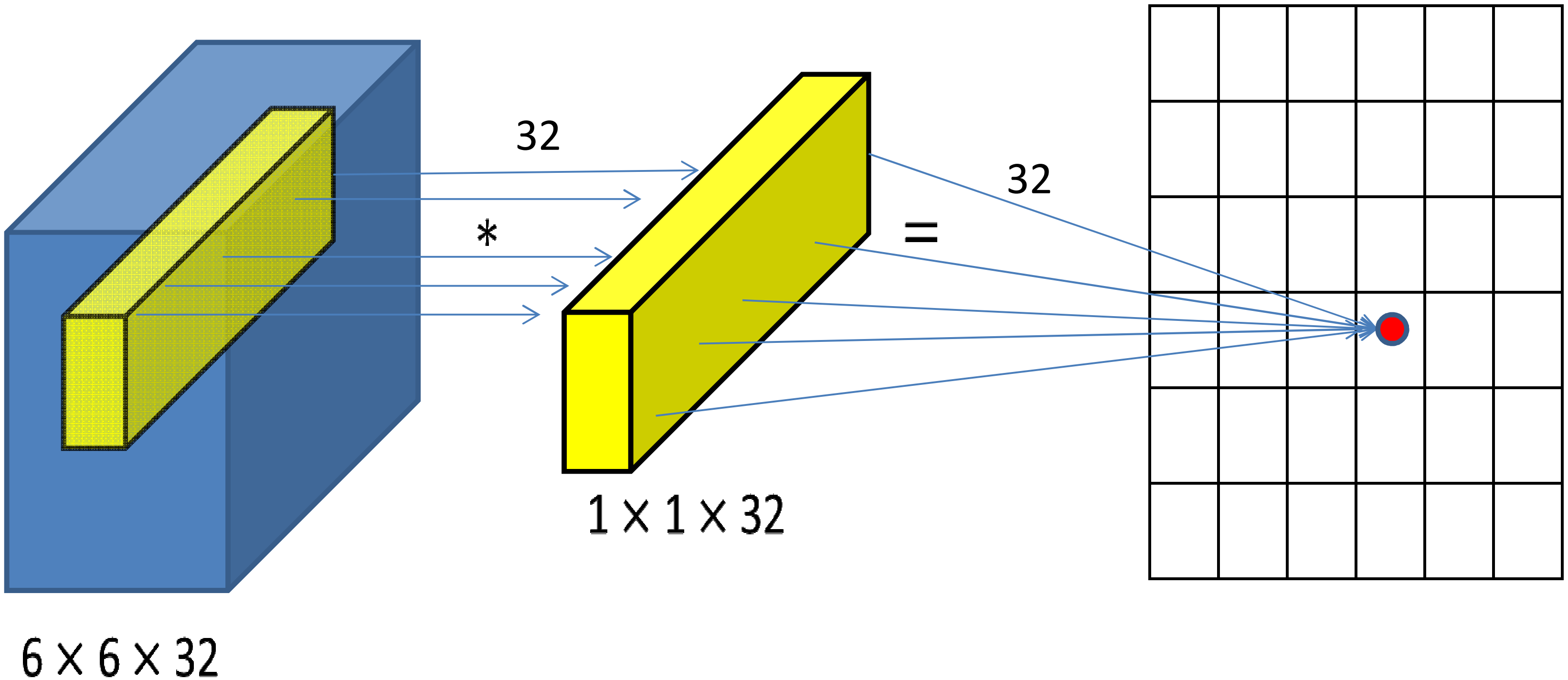
- Network in Network
- Inception Network
- Examples
  - Network in Network (NiN) 2014
  - Wide Residual Networks (2016)
  - Aggregated Residual Transformations for Deep Neural Networks (ResNeXt) 2017
  - FractalNet: Ultra-Deep Neural Networks without Residuals
  - Densely Connected Convolutional Networks
  - SqueezeNet: AlexNet-level Accuracy With 50x Fewer Parameters and <0.5Mb Model Size

What does 1x1 convolution do?

1X1 Convolutions



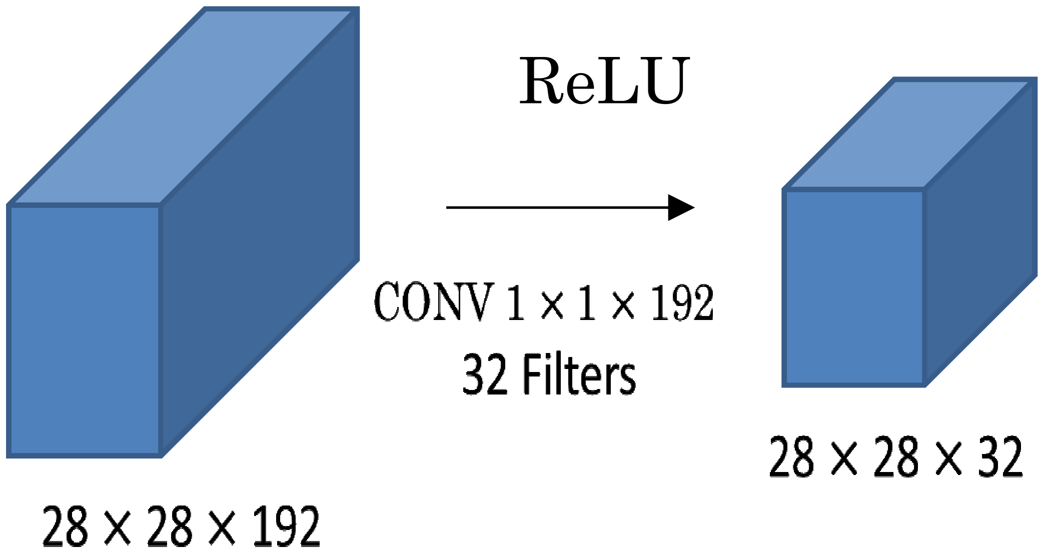
1X1 Convolutions



Pixel values from each of the 32 channels are multiplied with corresponding convolution kernel coefficients and are series summed to get one pixel of response matrix.

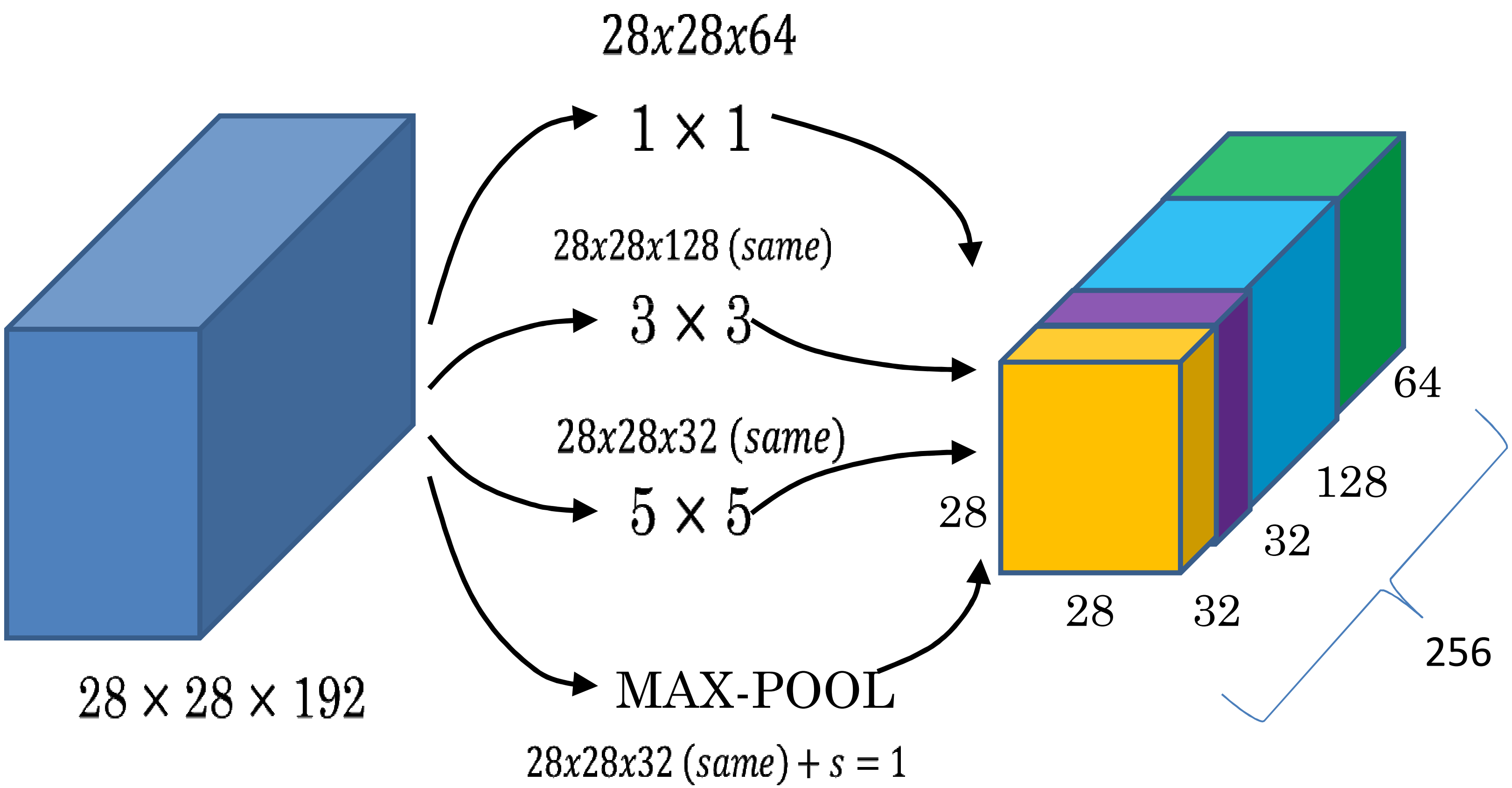
1X1 Convolutions

1X1 filters are often used to reduce the dimensionality of a layer



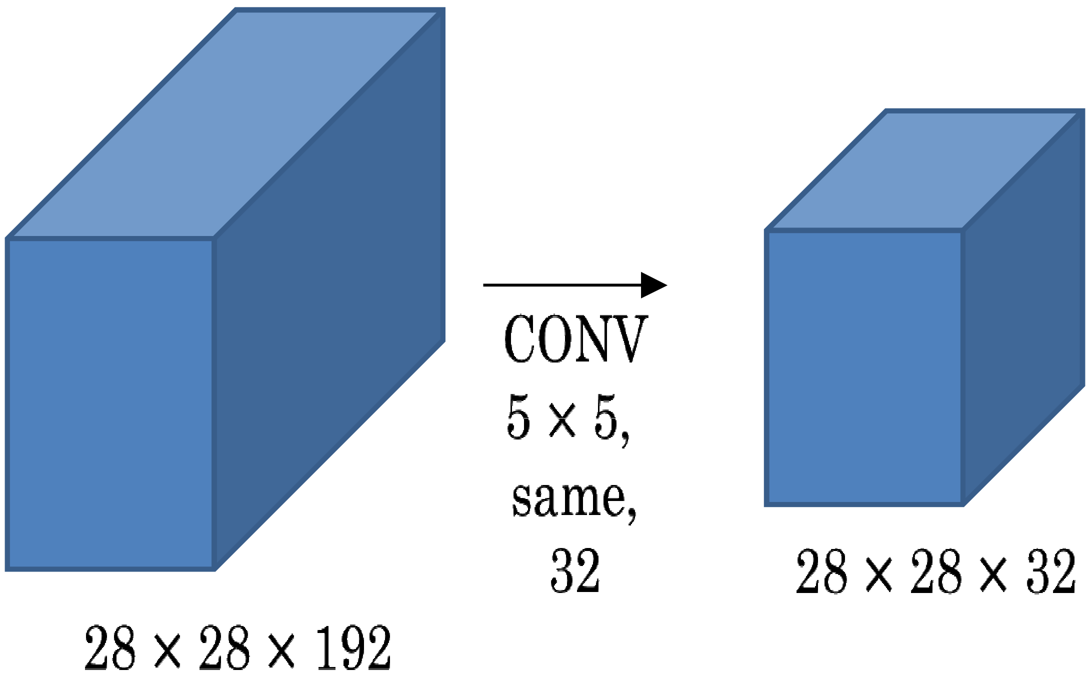
[Lin et al., 2013. Network in network]

Inception Network



[Szegedy et al. 2014. Going deeper with convolutions]

Inception Networks: Computational Cost

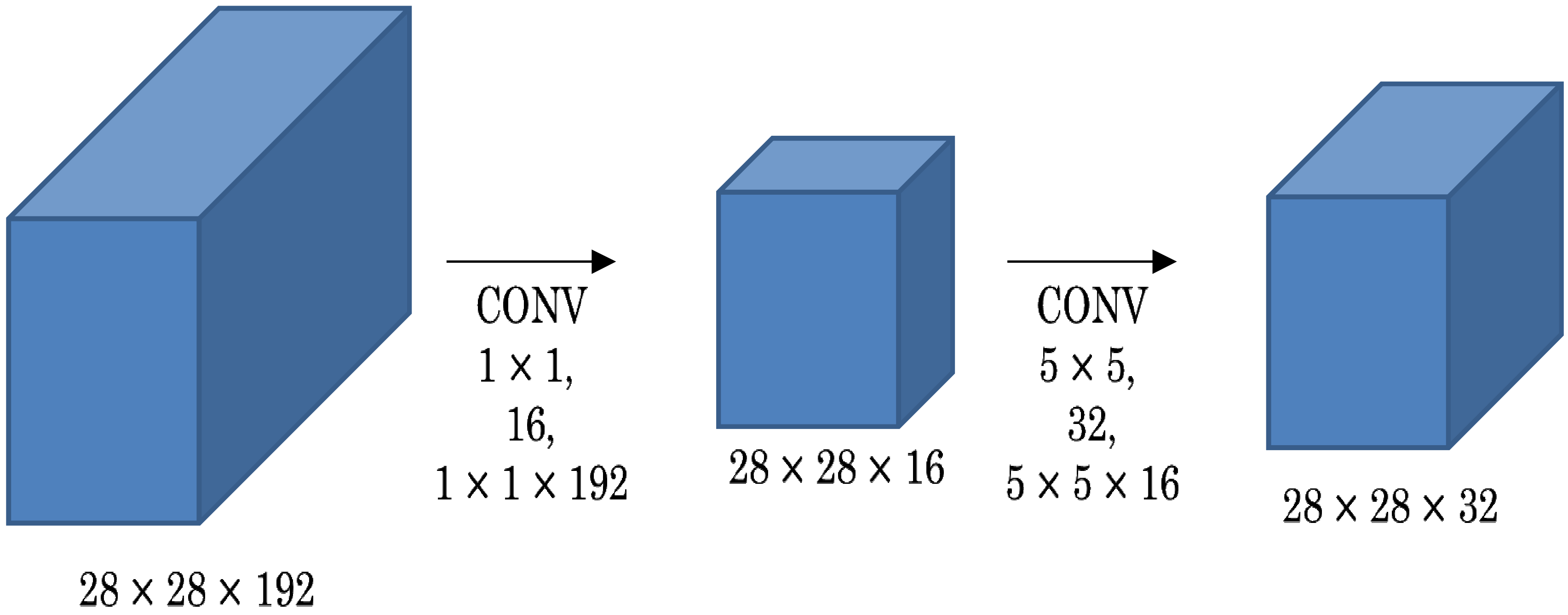


One output pixel requires 5x5x192 multiplications

There are total 28x28x32 output pixels

Total # of multiplications required = (5x5x192) x (28x28x32) **= 120M**

How Computational cost can be reduced with 1X1 Convolutions

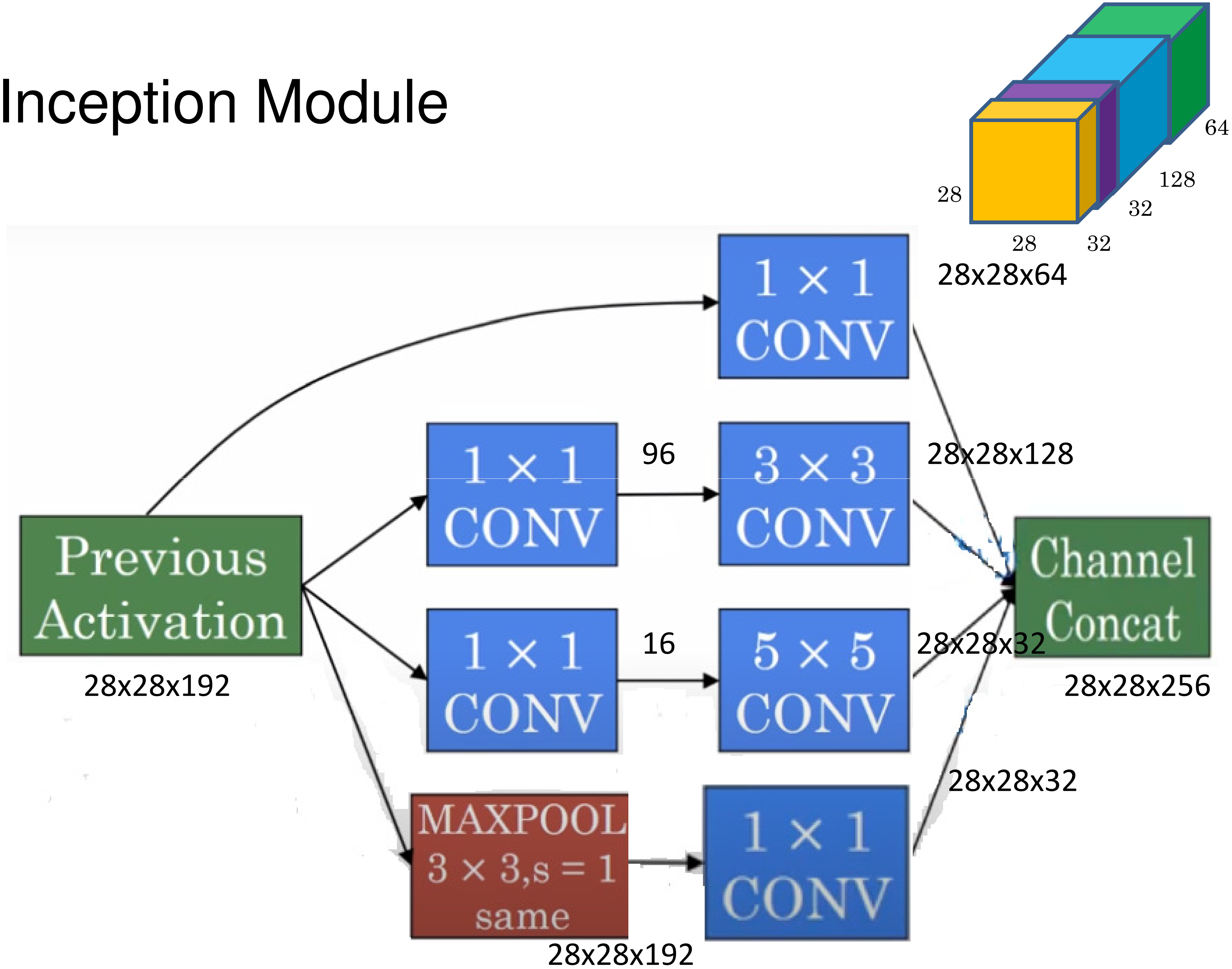


Multiplications for first layer
= 28x28x6x 192 = 2.4 M

Multiplications for second layer
= (28x28x32) x (5x5x16) = 10 M

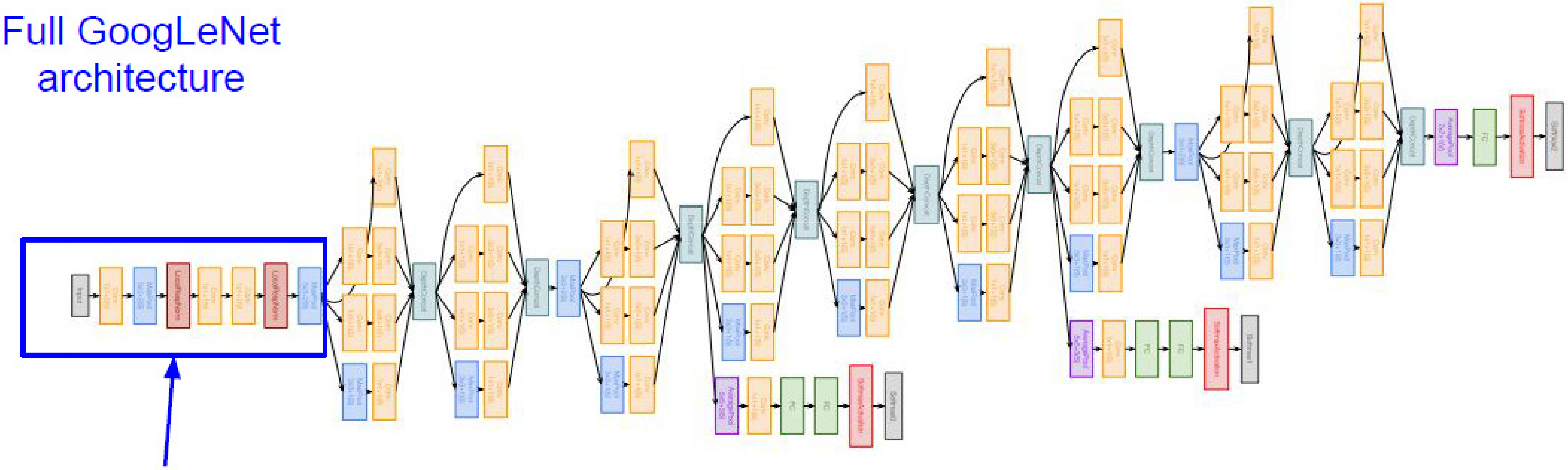
Total # of Multiplications = 2.4 + 10 =12.4 M which is approximately one tenth of the previous inception model (120 M)

Inception Module



Inception Network: Google Net

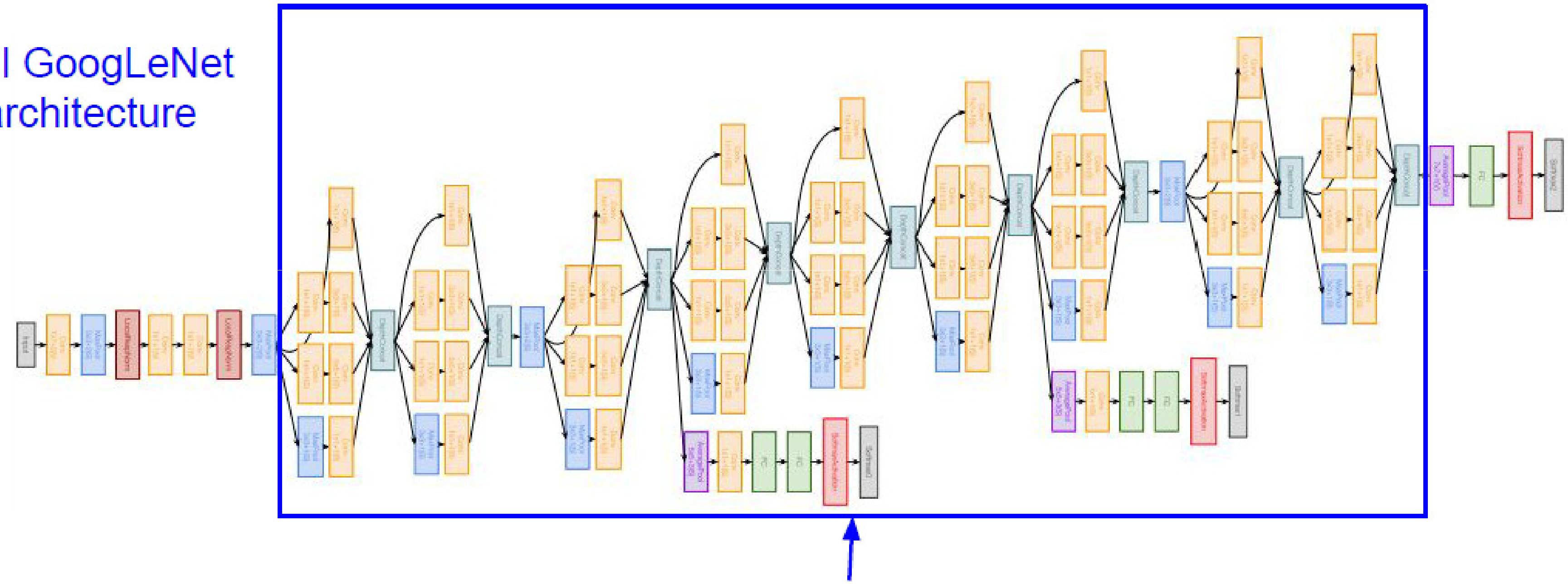
Full GoogLeNet architecture



Stem Network:
Conv-Pool-
2x Conv-Pool

Inception Network: Google Net

Full GoogLeNet architecture

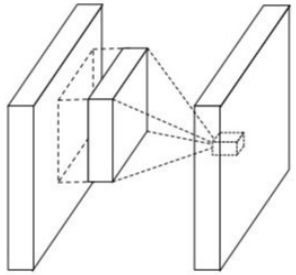


Stacked Inception Modules

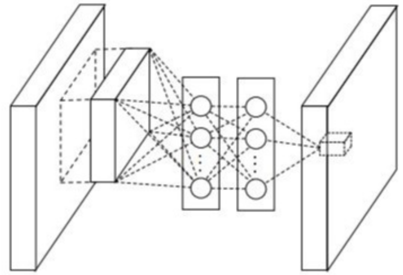
Network in Network (NiN)

[Lin et al. 2014]

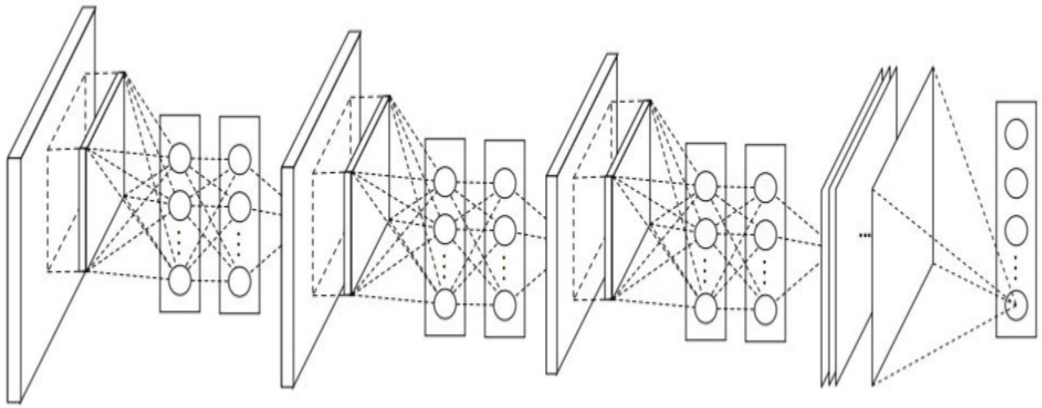
- Mlpconv layer with "micronetwork" within each conv layer to compute more abstract features for local patches
- Micronetwork uses multilayer perceptron (FC, i.e. 1x1 conv layers)
- Precursor to GoogLeNet and ResNet "bottleneck" layers
- Philosophical inspiration for GoogLeNet



(a) Linear convolution layer



(b) Mlpconv layer

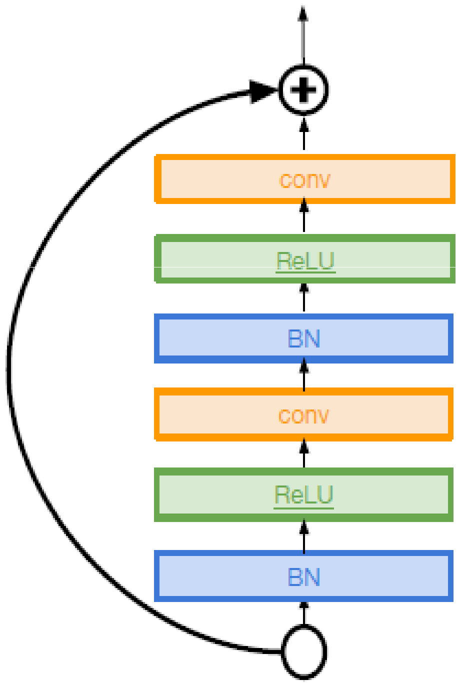


Improving ResNets...

Identity Mappings in Deep Residual Networks

[He et al. 2016]

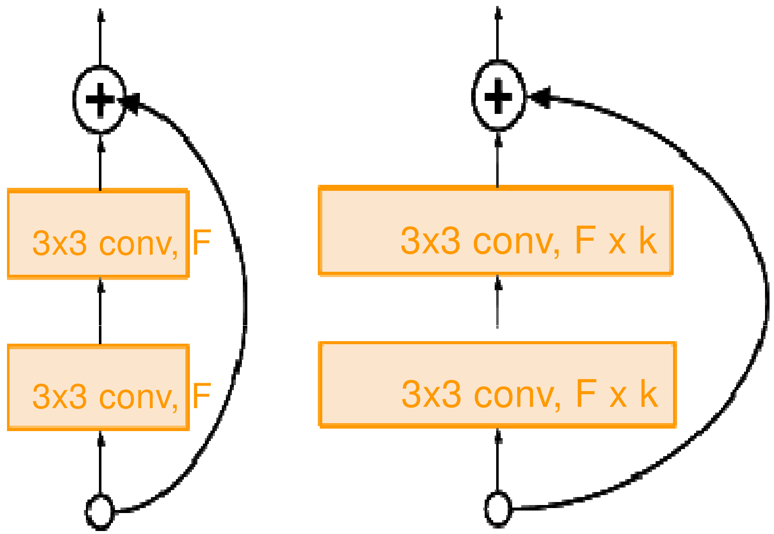
- Improved ResNet block design from creators of ResNet
- Creates a more direct path for propagating information throughout network (moves activation to residual mapping pathway)
- Gives better performance



Wide Residual Networks

[Zagoruyko et al. 2016]

- Argues that residuals are the important factor, not depth
- User wider residual blocks (F x k filters instead of F filters in each layer)
- 50-layer wide ResNet outperforms 152-layer original ResNet
- Increasing width instead of depth more computationally efficient (parallelizable)



Basic residual block

Wide residual block

Improving ResNets...
Aggregated Residual Transformations for Deep Neural Networks (ResNeXt)

[Xie et al. 2016]

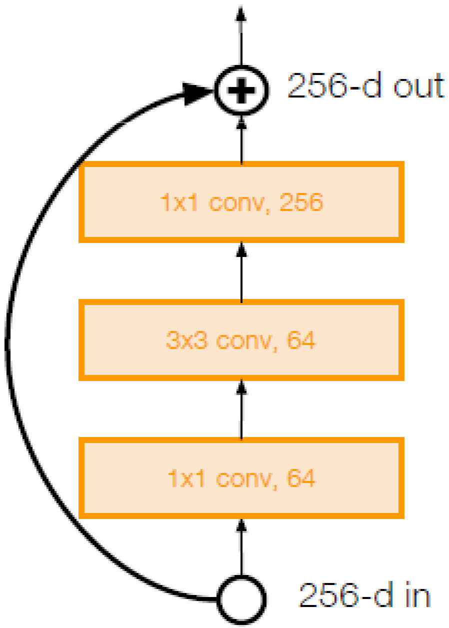
- Also from creators of ResNet

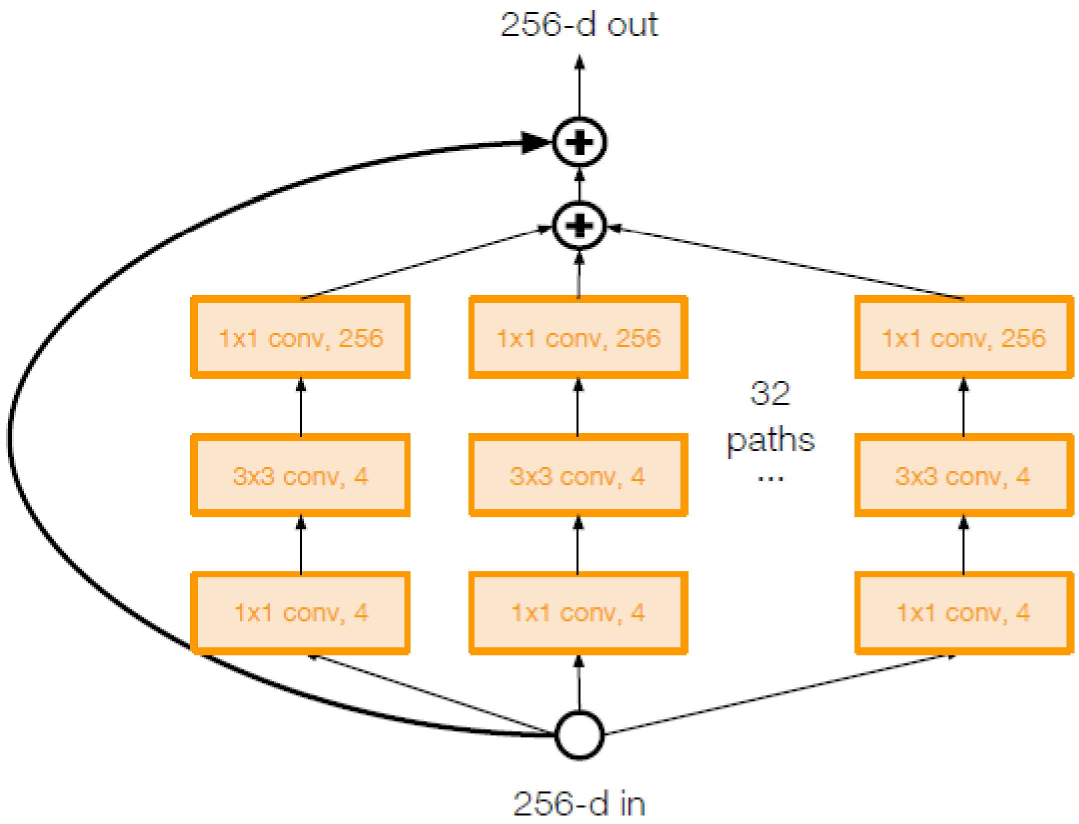
- Increases width of residual block through multiple parallel pathways (“cardinality”)

- Parallel pathways similar in spirit to Inception module

Improving ResNets...
Aggregated Residual Transformations for Deep Neural Networks (ResNeXt)

[Xie et al. 2016]





ResNeXt Networks

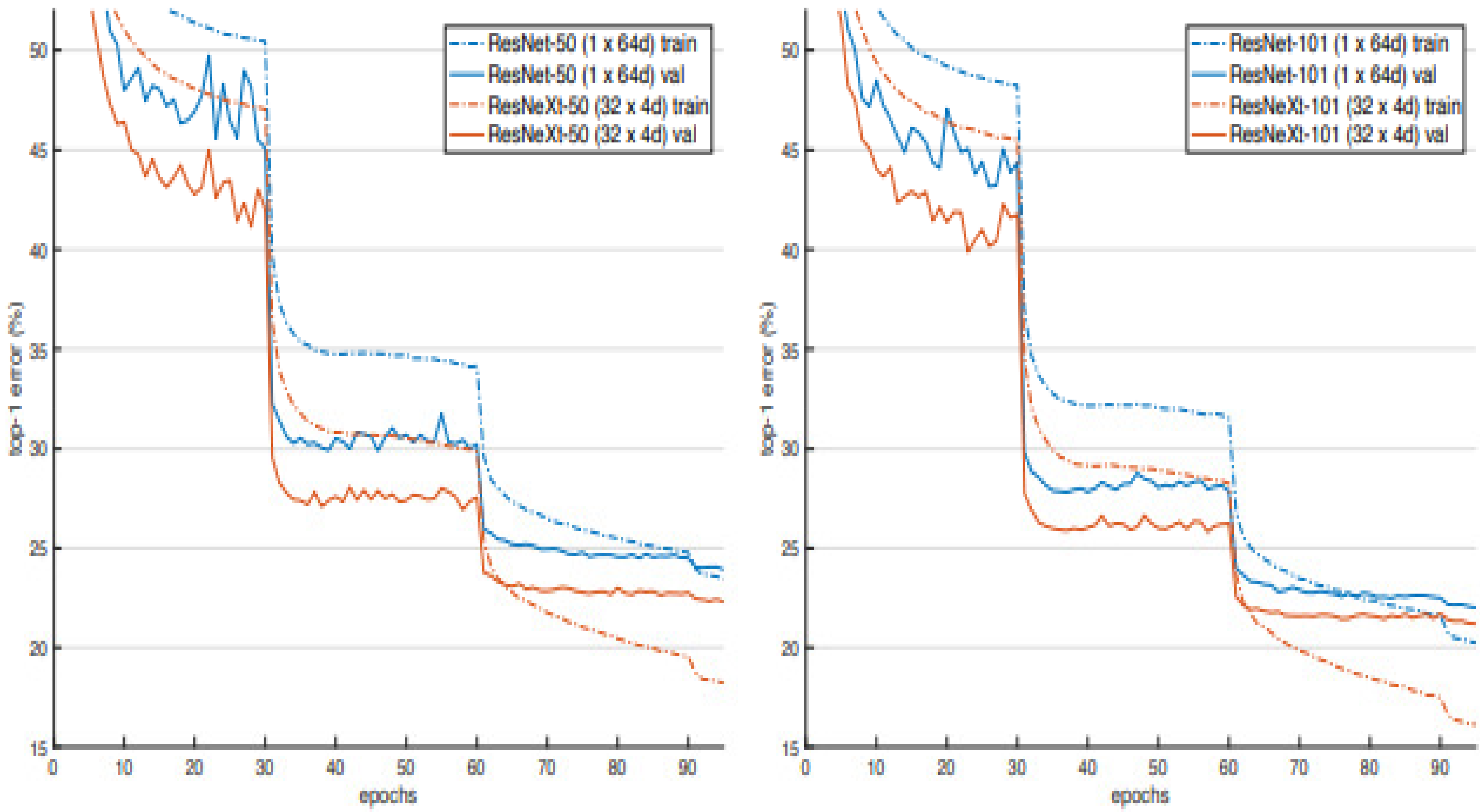


Figure 5. Training curves on ImageNet-1K. (**Left**): ResNet/ResNeXt-50 with preserved complexity (~4.1 billion FLOPs, ~25 million parameters); (**Right**): ResNet/ResNeXt-101 with preserved complexity (~7.8 billion FLOPs, ~44 million parameters).

ResNeXt Networks

Test error rates vs. model sizes:
The increasing cardinality is more effective than increasing width

This graph shows the results and model sizes, comparing with the Wide ResNet which is the best published record (observed on ImageNet-1K).

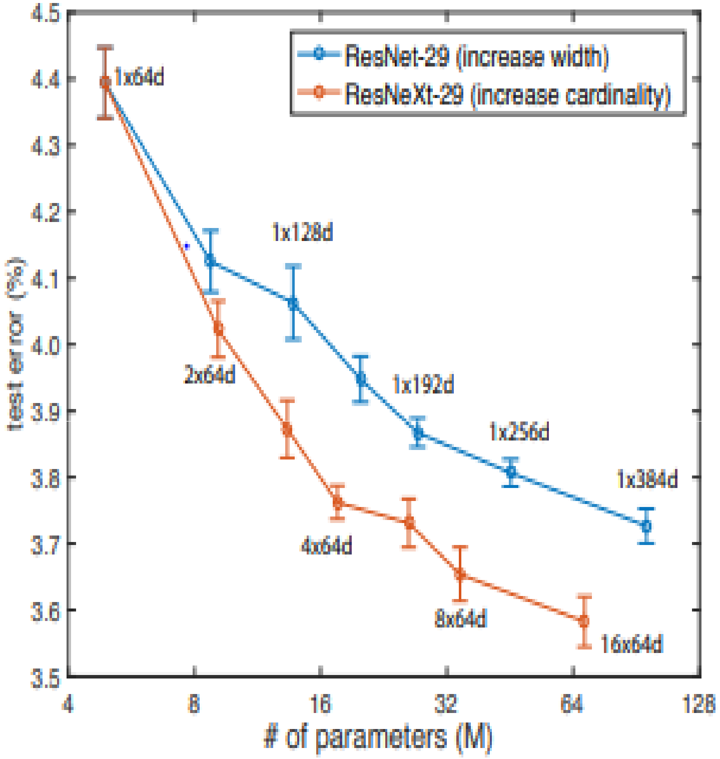


Figure 7. Test error *vs.* model size on CIFAR-10. The results are computed with 10 runs, shown with standard error bars. The labels show the settings of the templates.

Deep Networks with Stochastic Depth

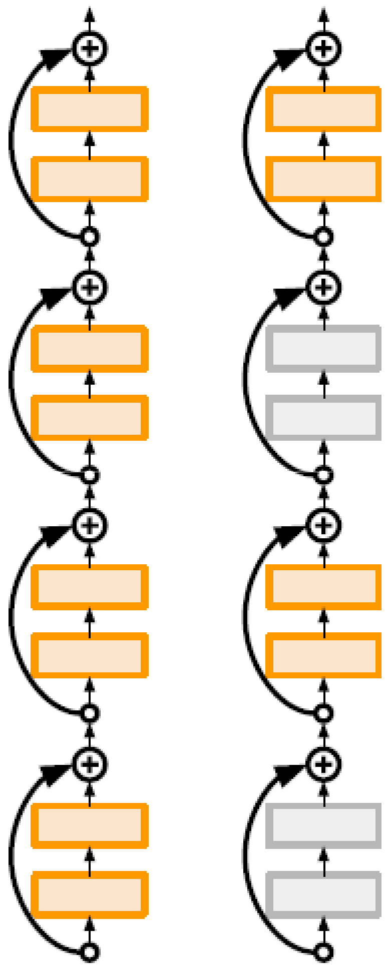
Huang et. al. 2016

Motivation: reduce vanishing gradients and training time through short networks during training

- Randomly drop a subset of layers during each training pass

- Bypass with identity function

- Use full deep network at test time



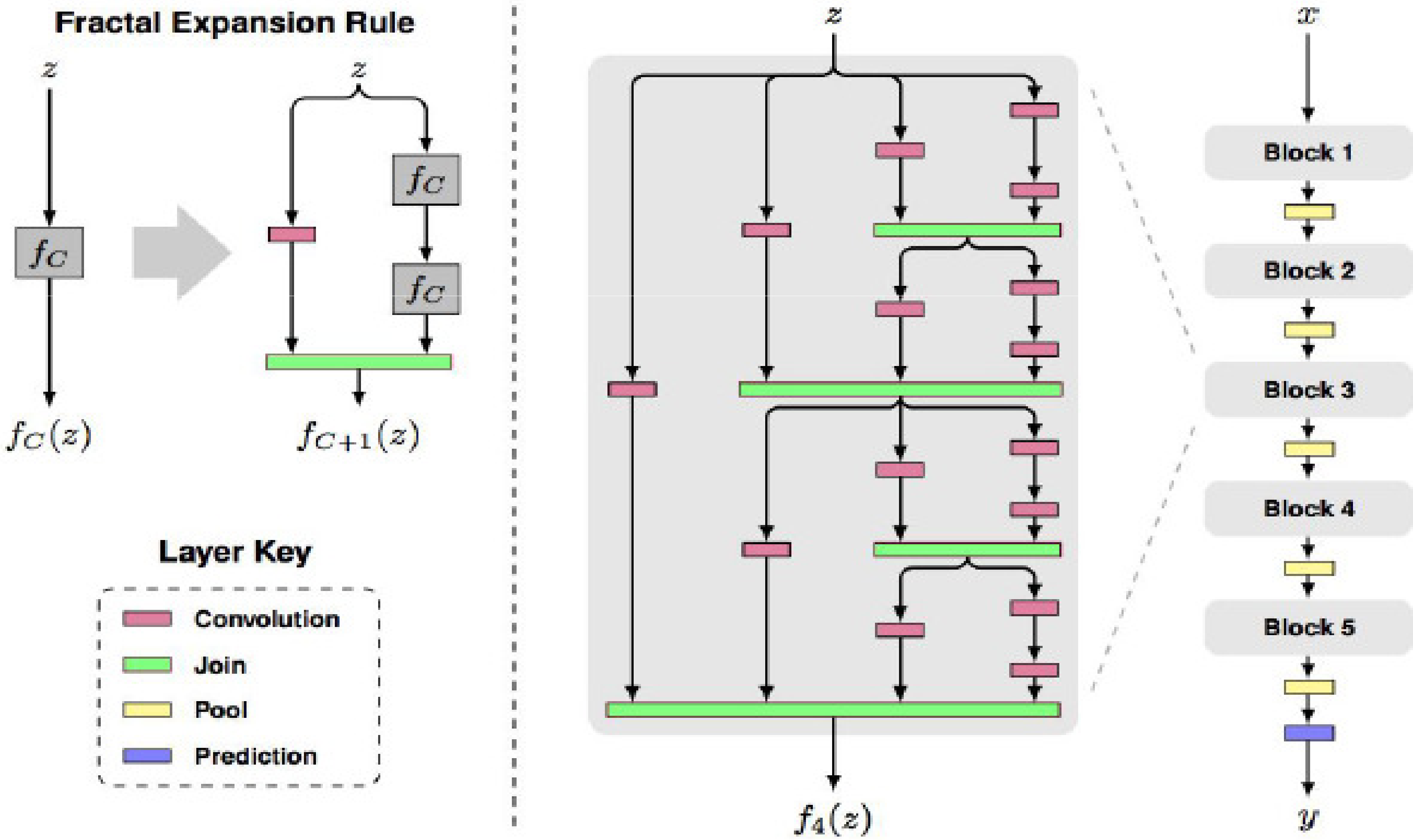
FractalNet: Ultra-Deep Neural Networks without Residuals

Larsson et al. 2017

- Argues that key is transitioning effectively from shallow to deep and residual representations are not necessary
- Fractal architecture with both shallow and deep paths to output
- Trained with dropping out sub-paths
- Full network at test time

FractalNet: Ultra-Deep Neural Networks without Residuals

Larsson et al. 2017



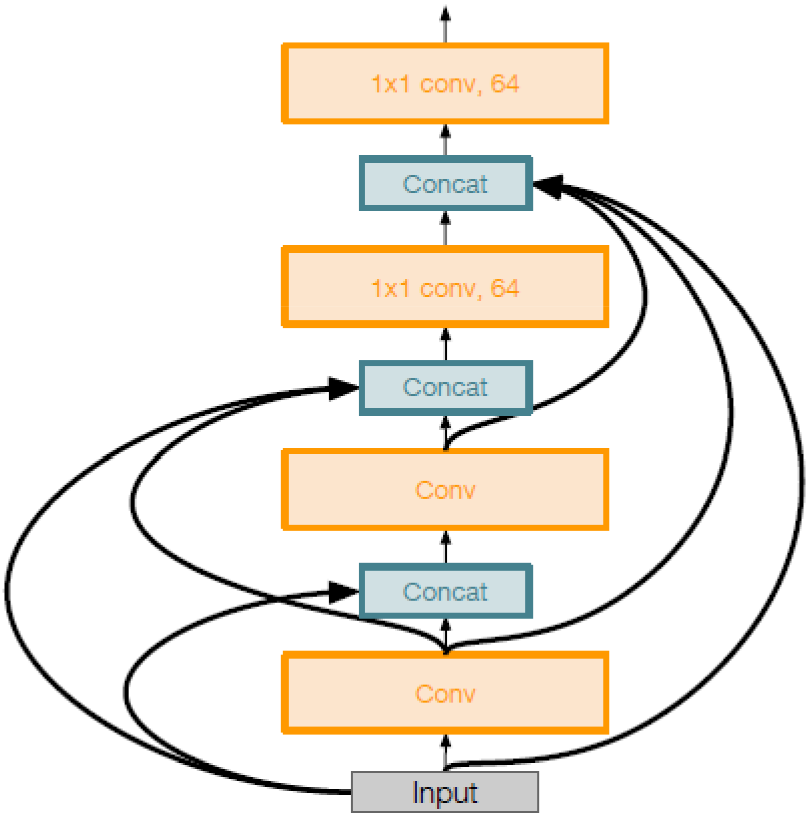
Densely Connected Convolutional Networks
Huang et al. 2017

- Dense blocks where each layer is connected to every other layer in feed-forward fashion

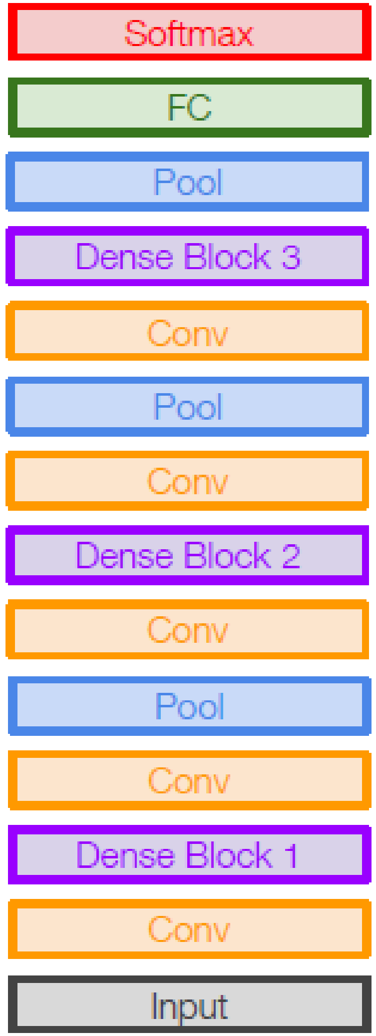
- Alleviates vanishing gradient, strengthens feature propagation, encourages feature reuse

Densely Connected Convolutional Networks

Huang et al. 2017



Dense Block



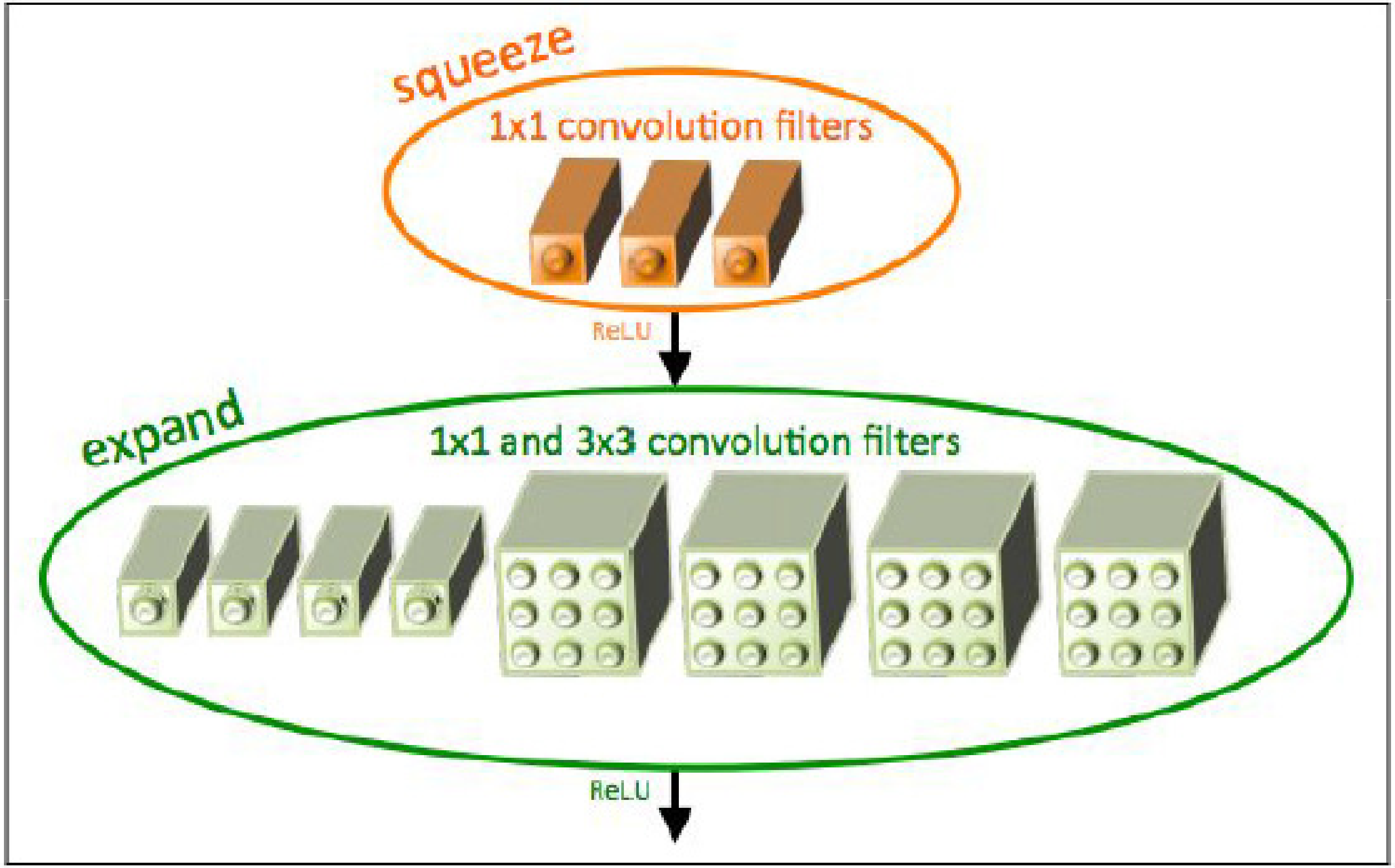
SqueezeNet: AlexNet-level Accuracy With 50x Fewer Parameters and <0.5Mb Model Size

Iandola et al. 2017

- Fire modules consisting of a ‘squeeze’ layer with 1x1 filters feeding an ‘expand’ layer with 1x1 and 3x3 filters
- AlexNet level accuracy on ImageNet with 50x fewer parameters
- Can compress to 510x smaller than AlexNet (0.5Mb)

SqueezeNet: AlexNet-level Accuracy With 50x Fewer Parameters and <0.5Mb Model Size

Iandola et al. 2017



to continue…