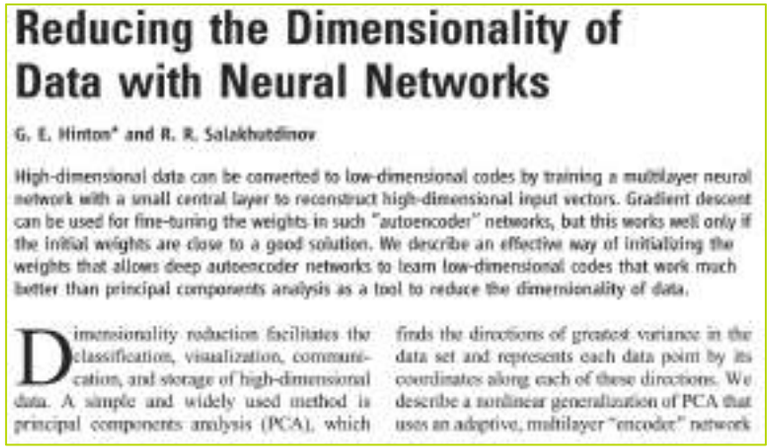
CNN Architecture Design: From Deeper to Wider

主讲人：微软亚洲研究院主管研究员 王井东

Deep learning in the past 10 years

- Reducing the dimensionality of data with neural networks, Science, 2006
  - Fast learning algorithms for Restricted Boltzmann machine

Not good as expected



Reducing the Dimensionality of Data with Neural Networks

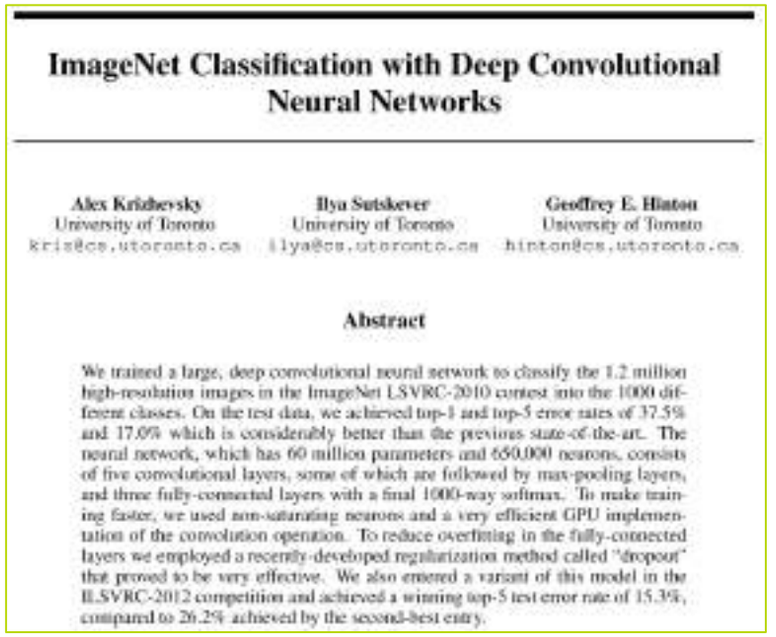
G. E. Hinton* and R. R. Salakhutdinov

High-dimensional data can be converted to low-dimensional codes by training a multilayer neural network with a small central layer to reconstruct high-dimensional input vectors. Gradient descent can be used for fine-tuning the weights in such "autoencoder" networks, but this works well only if the initial weights are close to a good solution. We describe an effective way of initializing the weights that allows deep autoencoder networks to learn low-dimensional codes that work much better than principal components analysis as a tool to reduce the dimensionality of data.

Dimensionality reduction facilitates the classification, visualization, communication, and storage of high-dimensional data. A simple and widely used method is principal components analysis (PCA), which finds the directions of greatest variance in the data set and represents each data point by its coordinates along each of these directions. We describe a nonlinear generalization of PCA that uses an adaptive, multilayer "encoder" network

- ImageNet Classification with deep convolutional neural networks, NIPS, 2012
  - Dramatic performance improvement
  - ImageNet, GPU

Win almost in all the applications



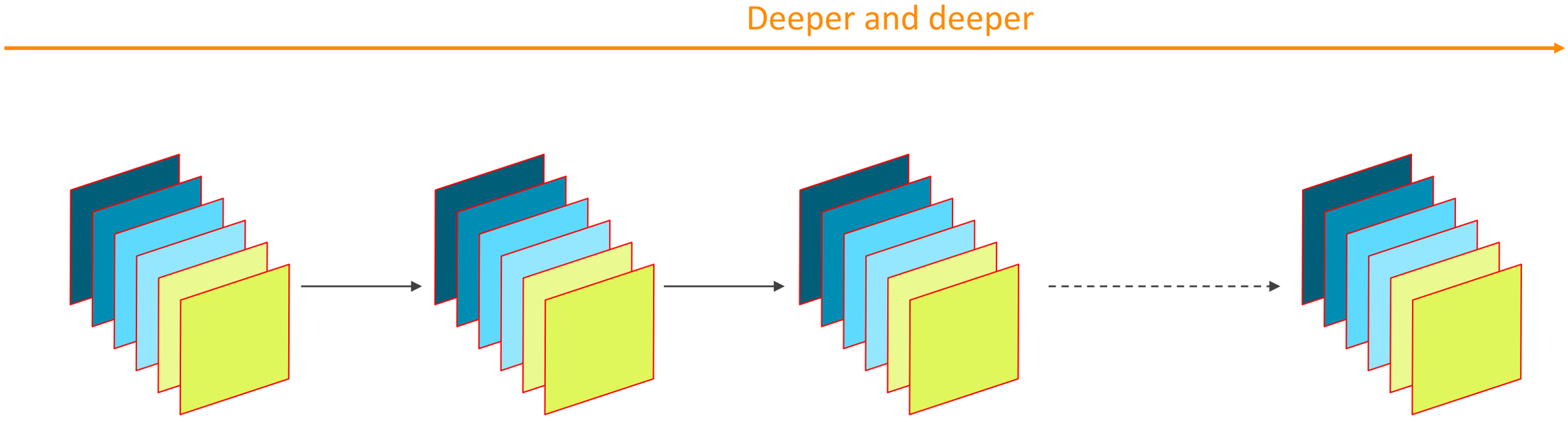
ImageNet Classification with Deep Convolutional Neural Networks

Alex Krizhevsky, University of Toronto, krizh@cs.utoronto.ca
Ilya Sutskever, University of Toronto, ilya@cs.utoronto.ca
Geoffrey E. Hinton, University of Toronto, hinton@cs.utoronto.ca

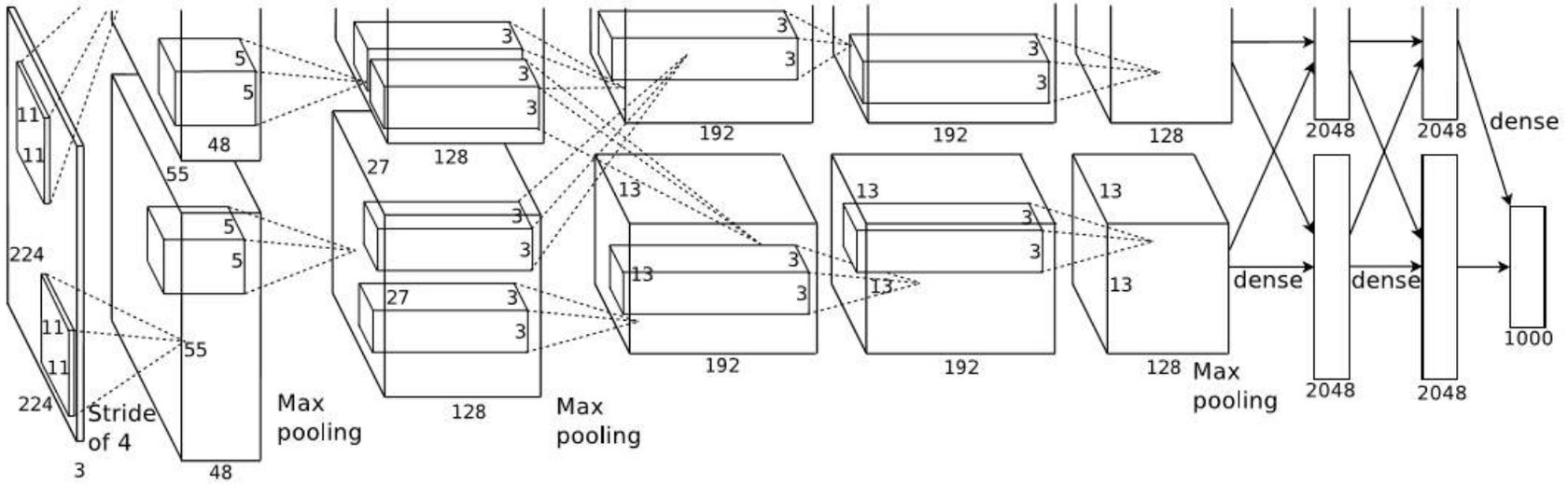
Abstract

We trained a large, deep convolutional neural network to classify the 1.2 million high-resolution images in the ImageNet LSVRC-2010 contest into the 1000 different classes. On the test data, we achieved top-1 and top-5 error rates of 37.5% and 17.0% which is considerably better than the previous state-of-the-art. The neural network, which has 60 million parameters and 650,000 neurons, consists of five convolutional layers, some of which are followed by max-pooling layers, and three fully-connected layers with a final 1000-way softmax. To make training faster, we used non-saturating neurons and a very efficient GPU implementation of the convolution operation. To reduce overfitting in the fully-connected layers we employed a recently-developed regularization method called "dropout" that proved to be very effective. We also entered a variant of this model in the ILSVRC-2012 competition and achieved a winning top-5 test error rate of 15.3%, compared to 26.2% achieved by the second-best entry.

Deeper and deeper

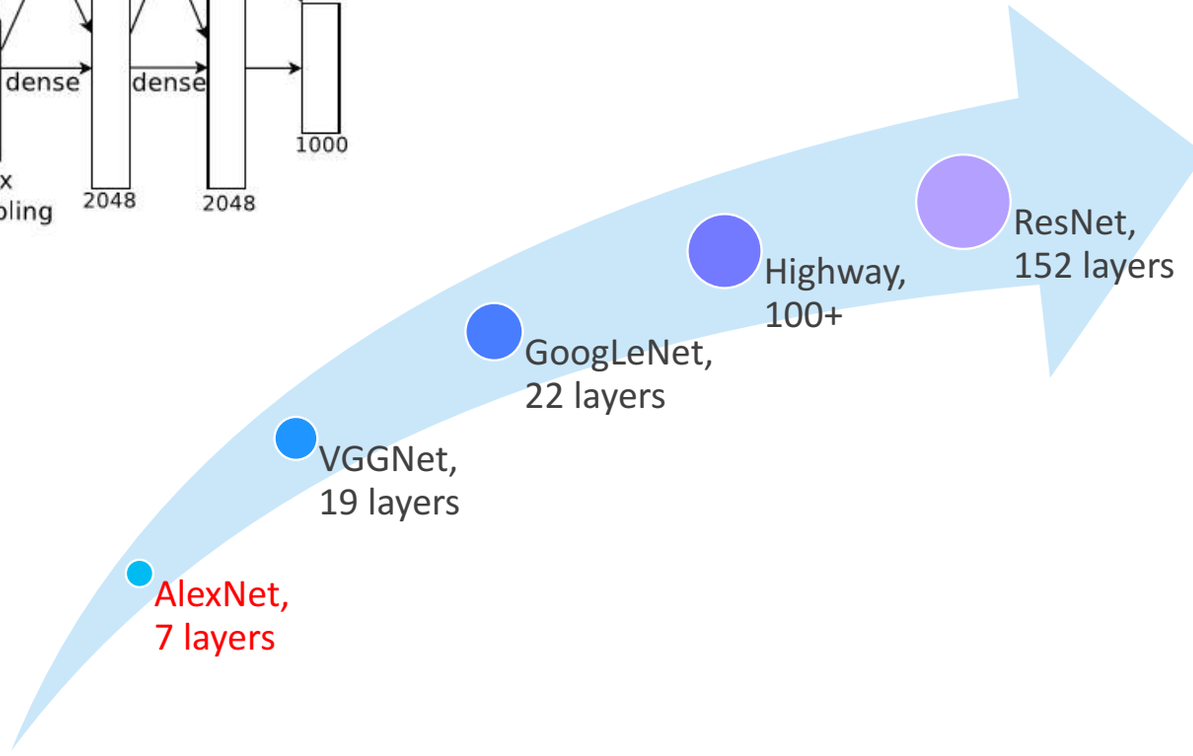


Deeper and deeper

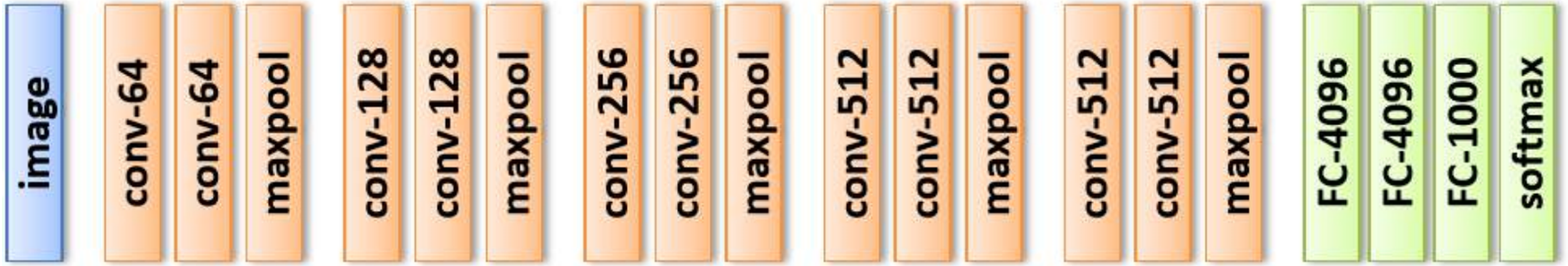


8 layers

AlexNet, 2012

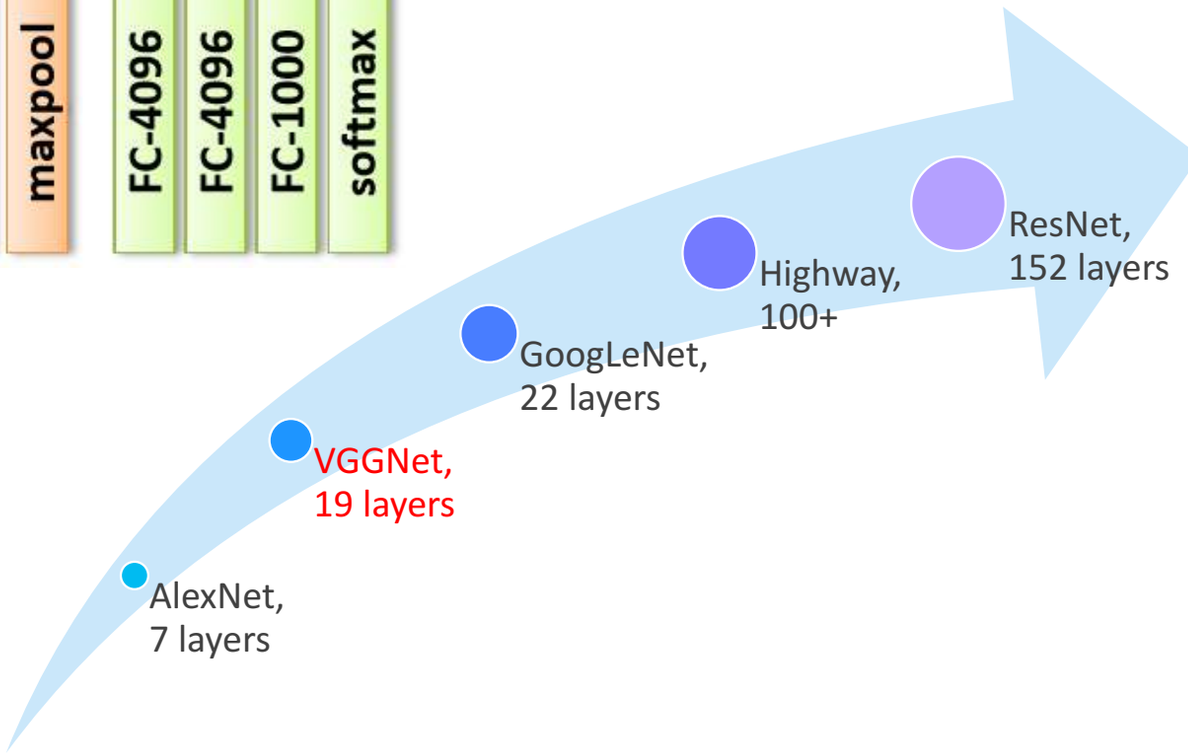


Deeper and deeper

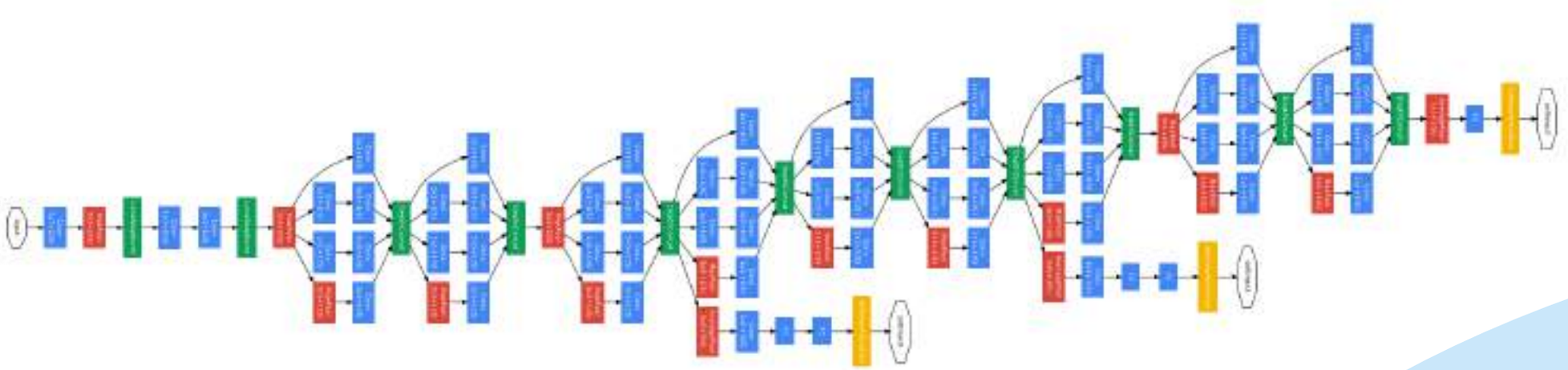


19 layers

VGGNet, 2014

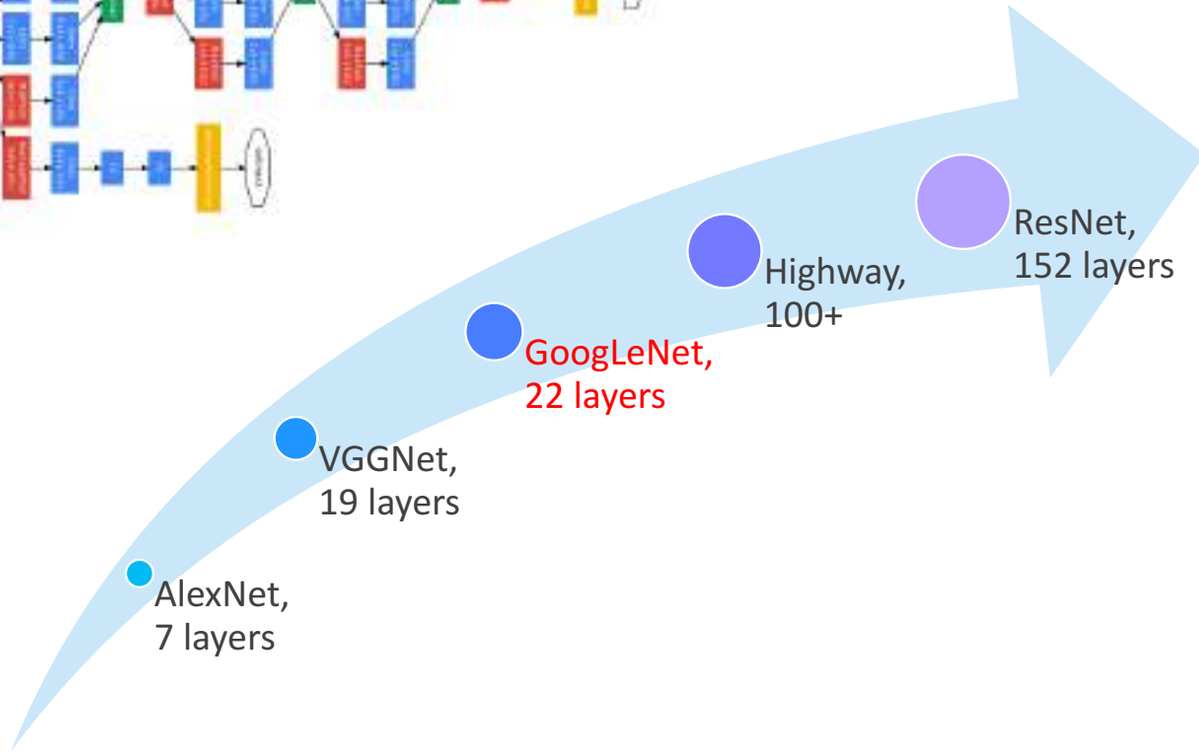


Deeper and deeper

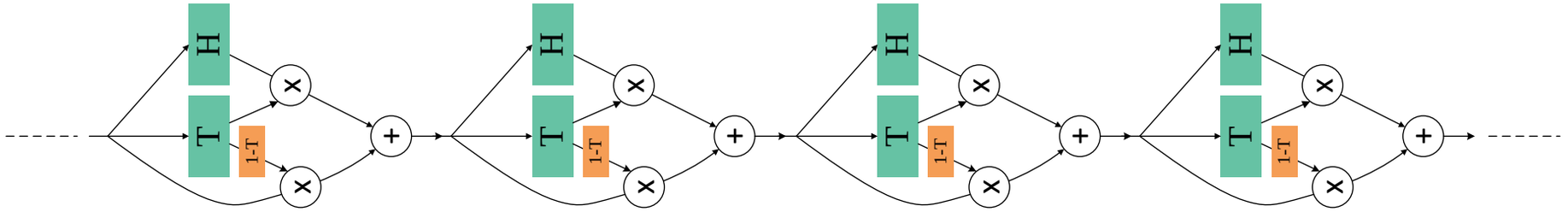


22 layers

GoogLeNet, 2014

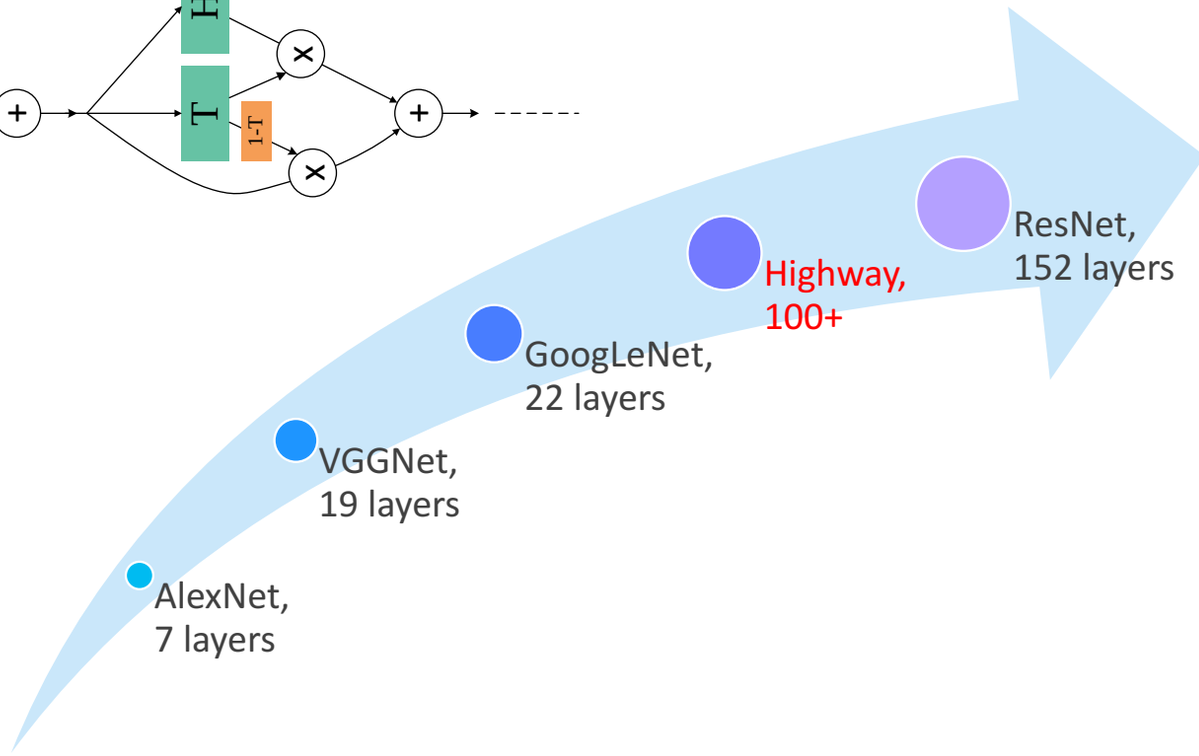


Deeper and deeper

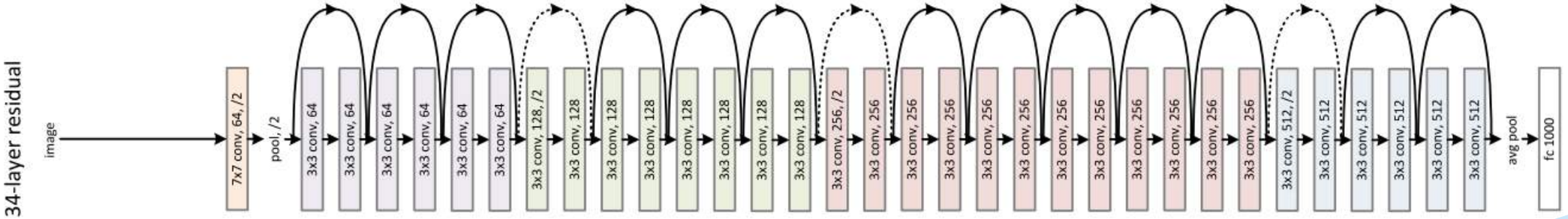


100+ layers

Highway, 2015

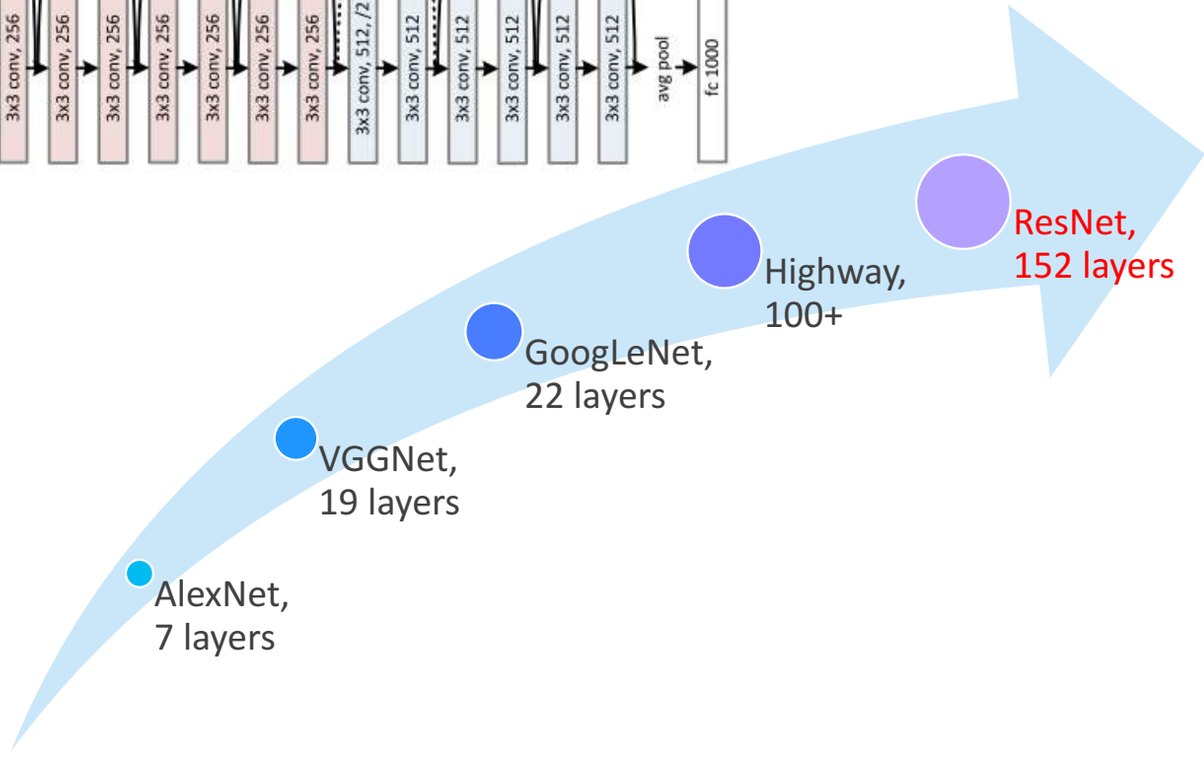


Deeper and deeper

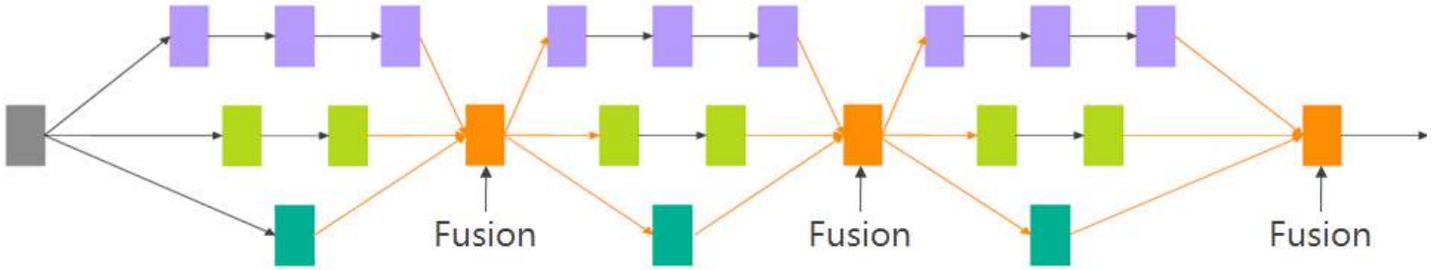


152 layers

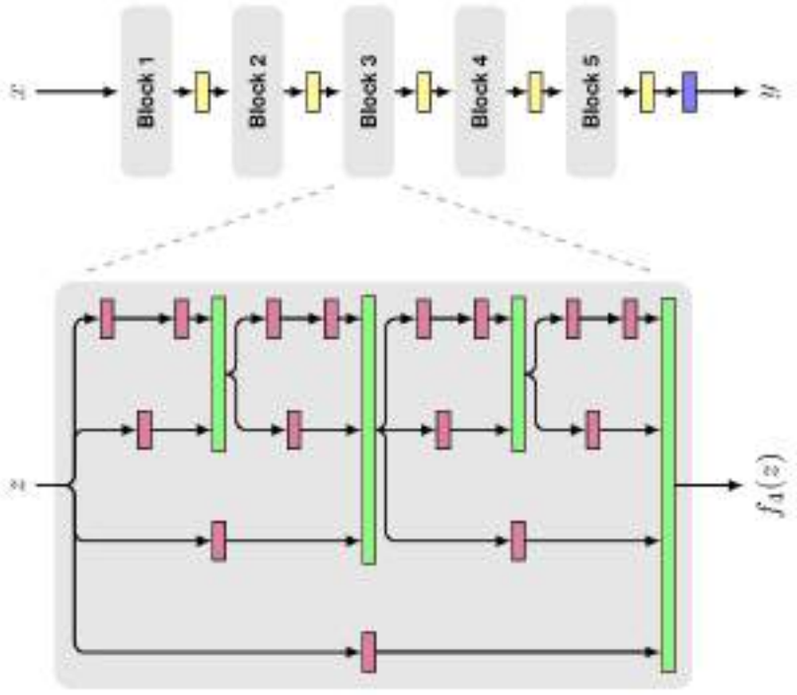
ResNet, 2015



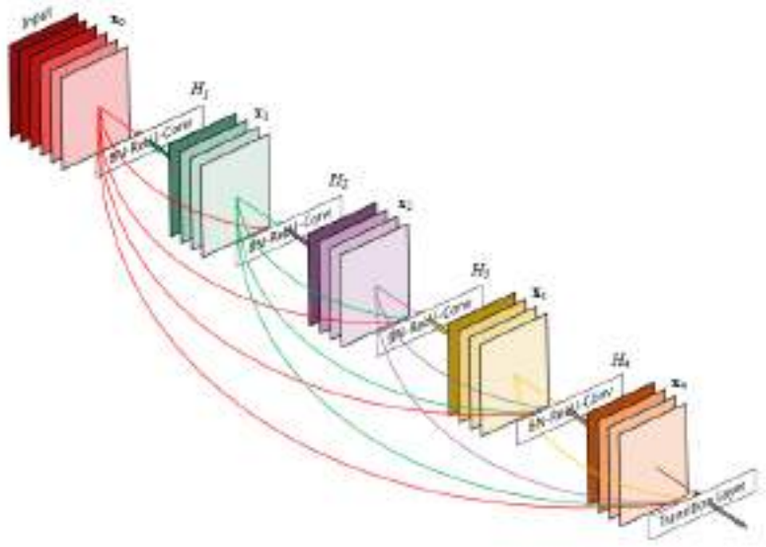
Other ultra-deep networks



Deeply-fused nets



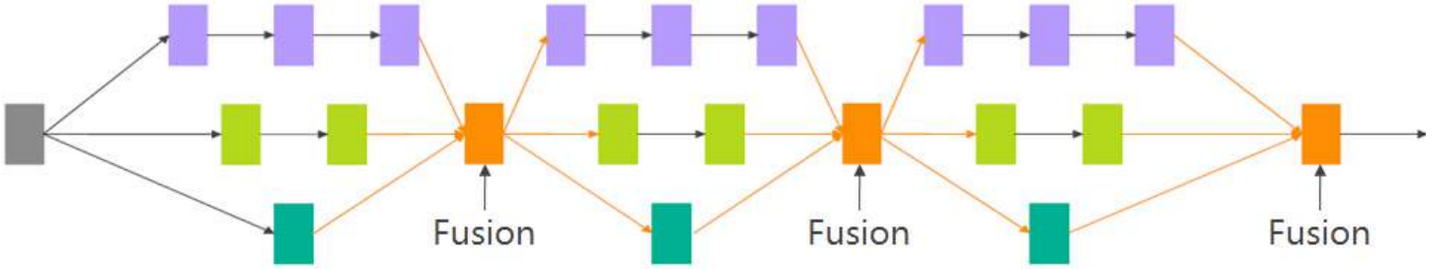
FractalNets



DenseNets

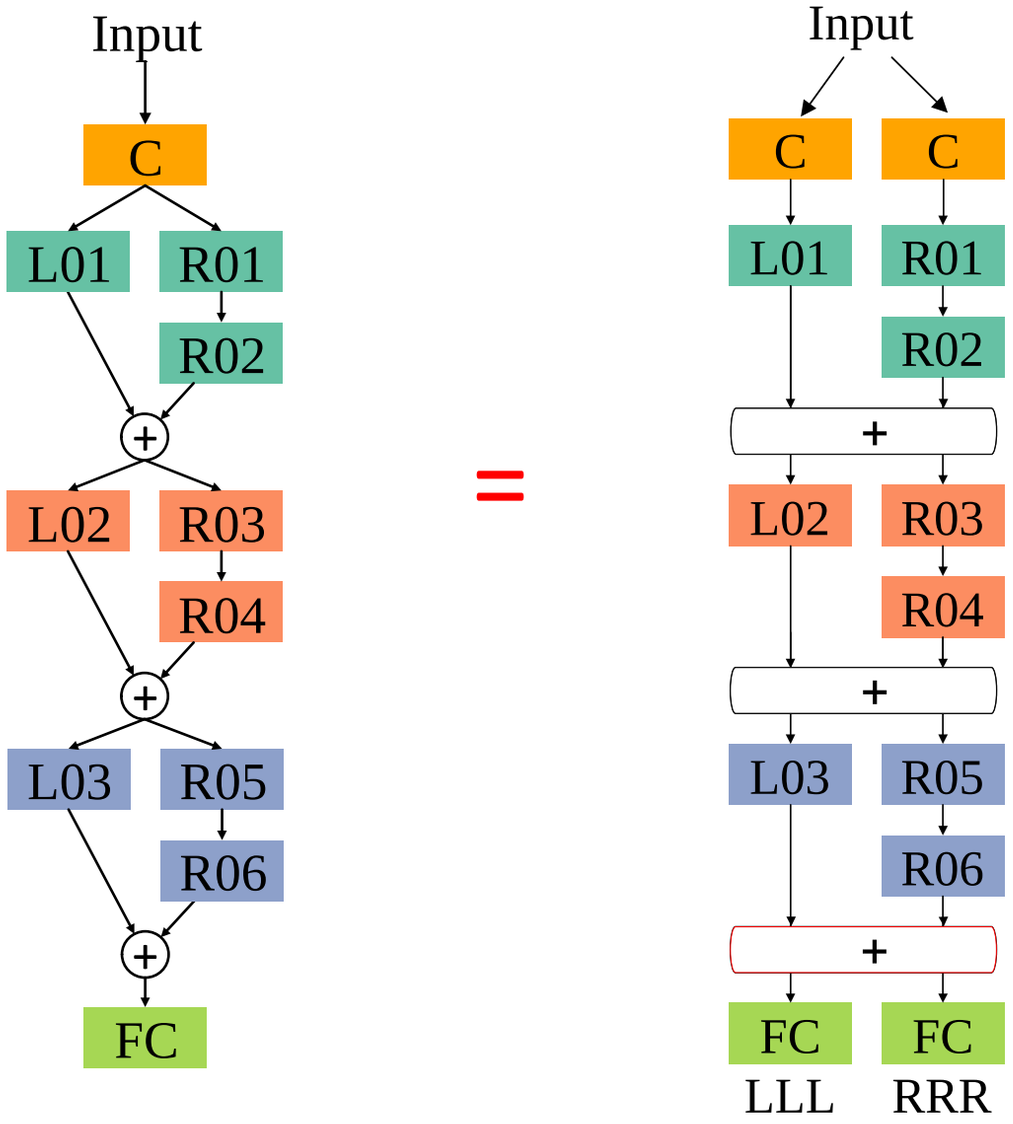
Deeply-Fused Nets

Unifying GoogLeNets, Highway, ResNets



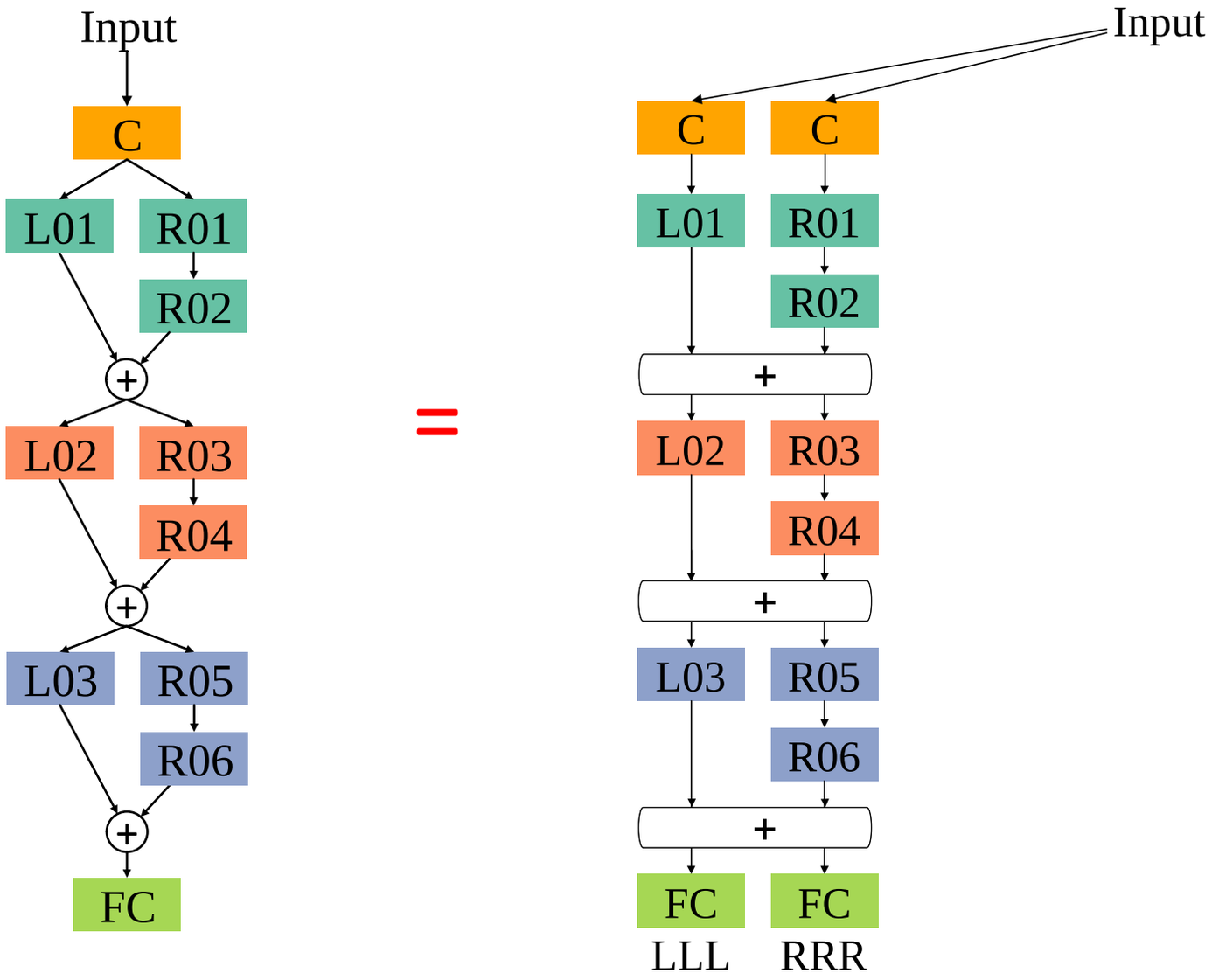
Jingdong Wang, Zhen Wei, Ting Zhang, Wenjun Zeng: Deeply-Fused Nets. CoRR abs/1605.07716 (2016)

Deeply-fused nets

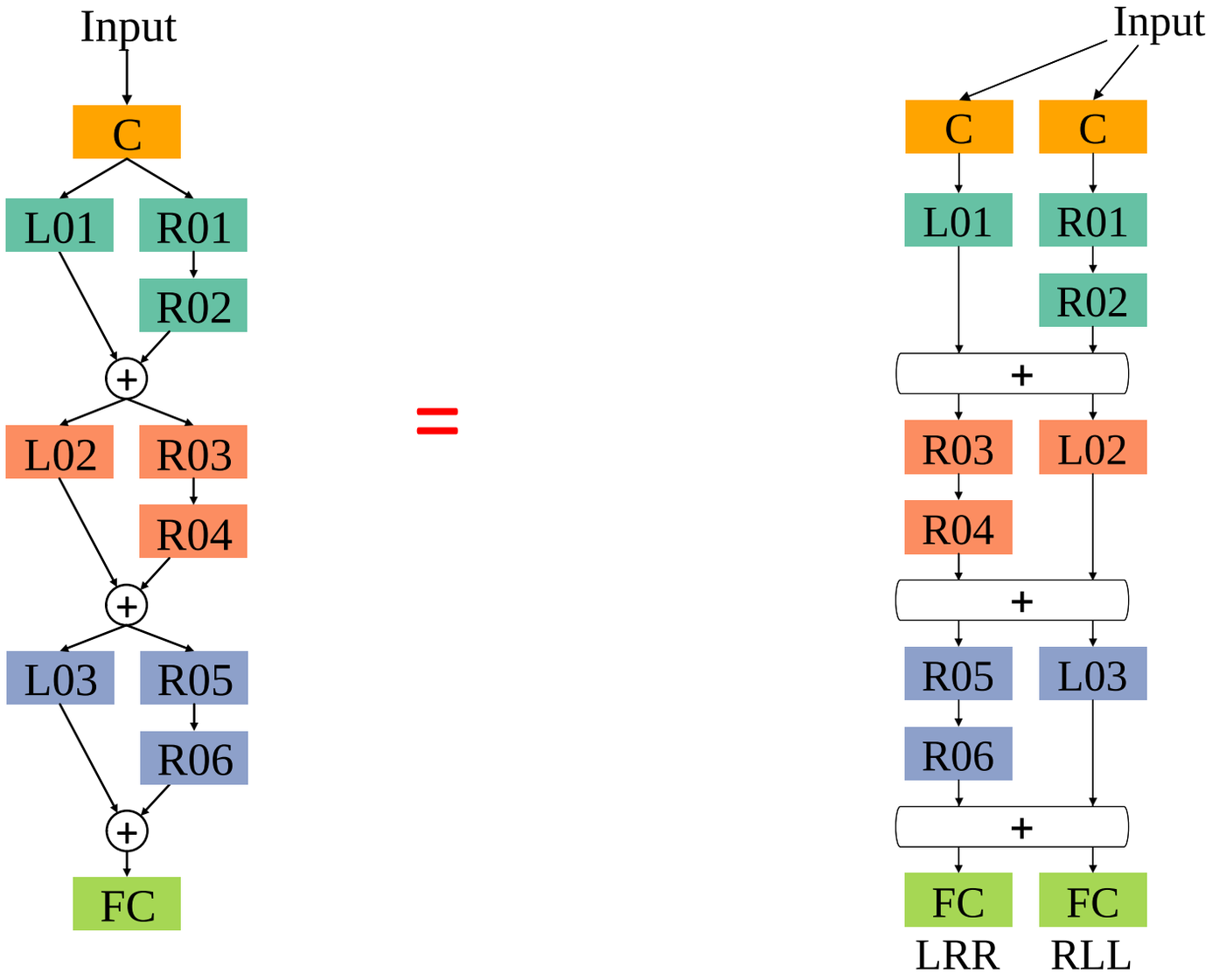


A deeply-fused net can be formed from many different base networks

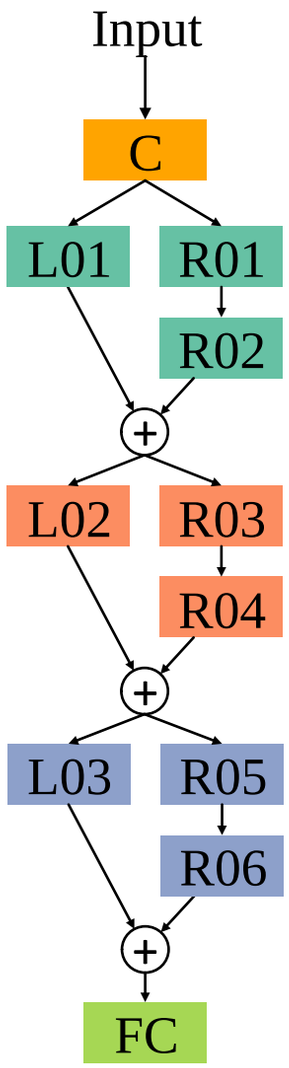
A deeply-fused net can be formed from many different base networks



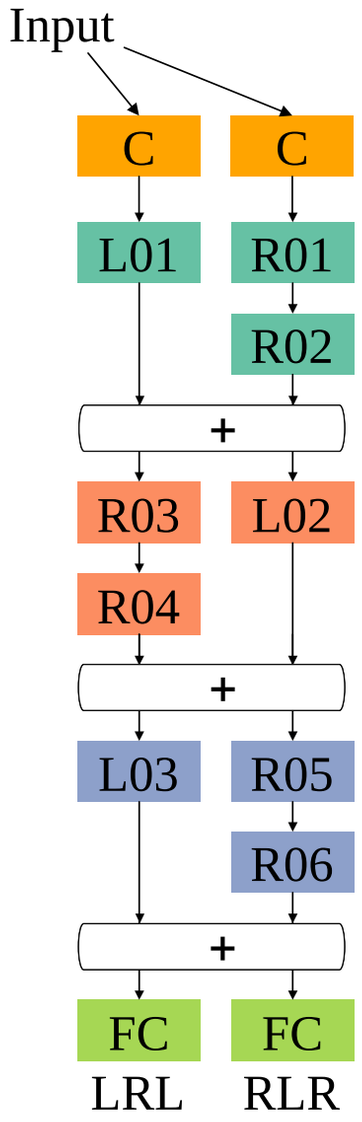
A deeply-fused net can be formed from many different base networks



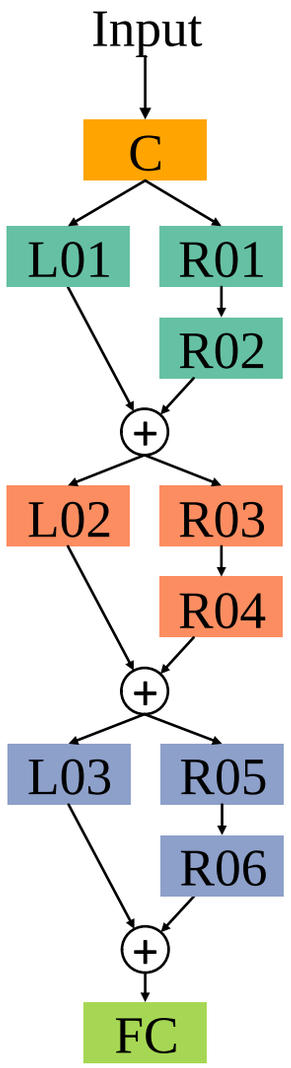
A deeply-fused net can be formed from many different base networks



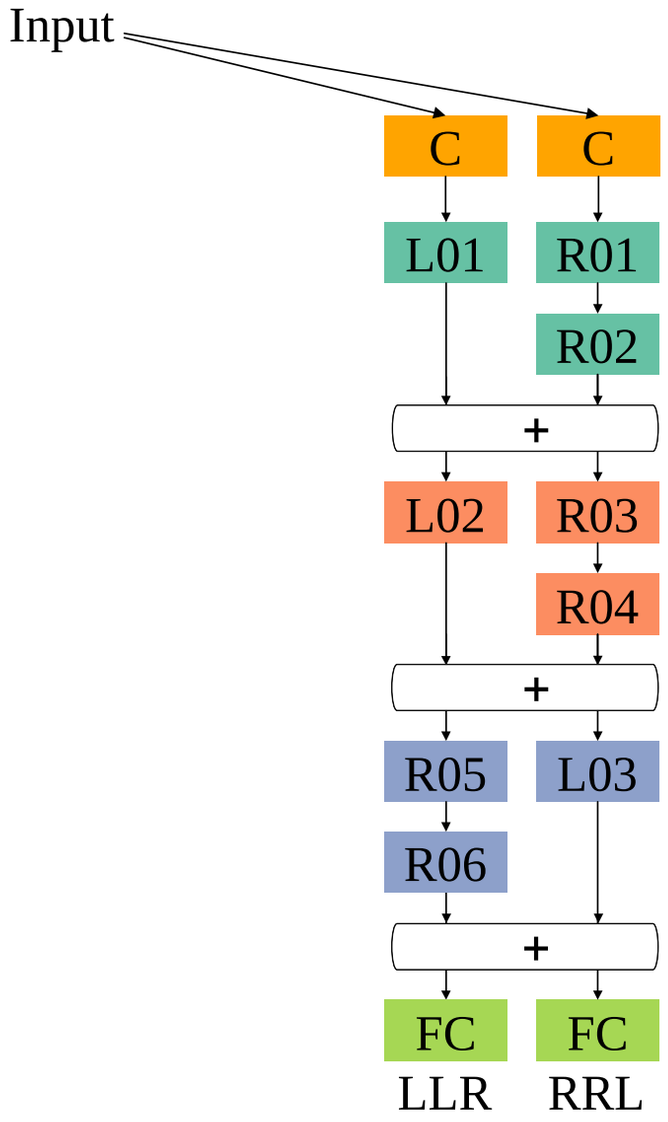
=



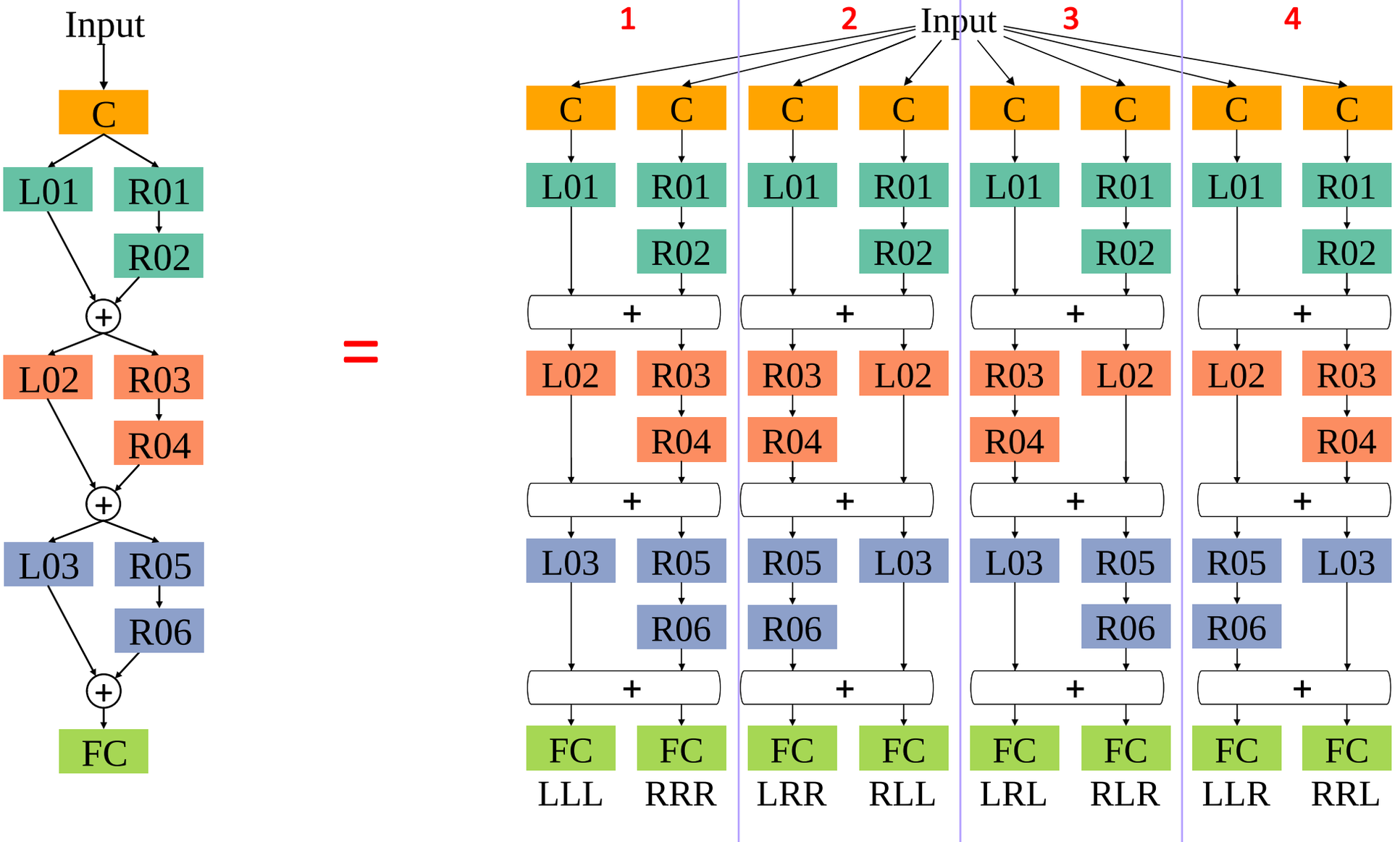
A deeply-fused net can be formed from many different base networks



=

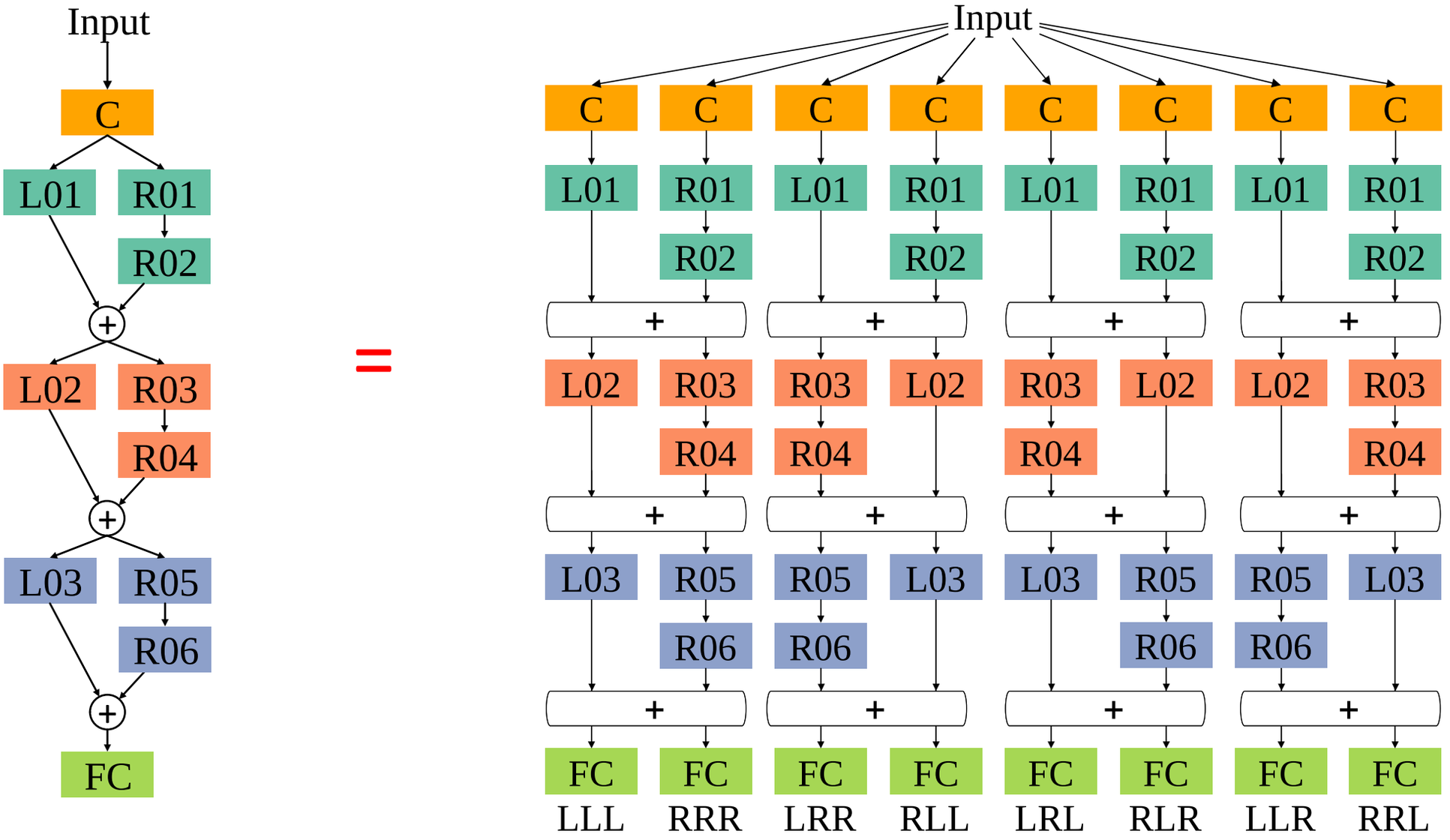


A deeply-fused net can be formed from many different base networks

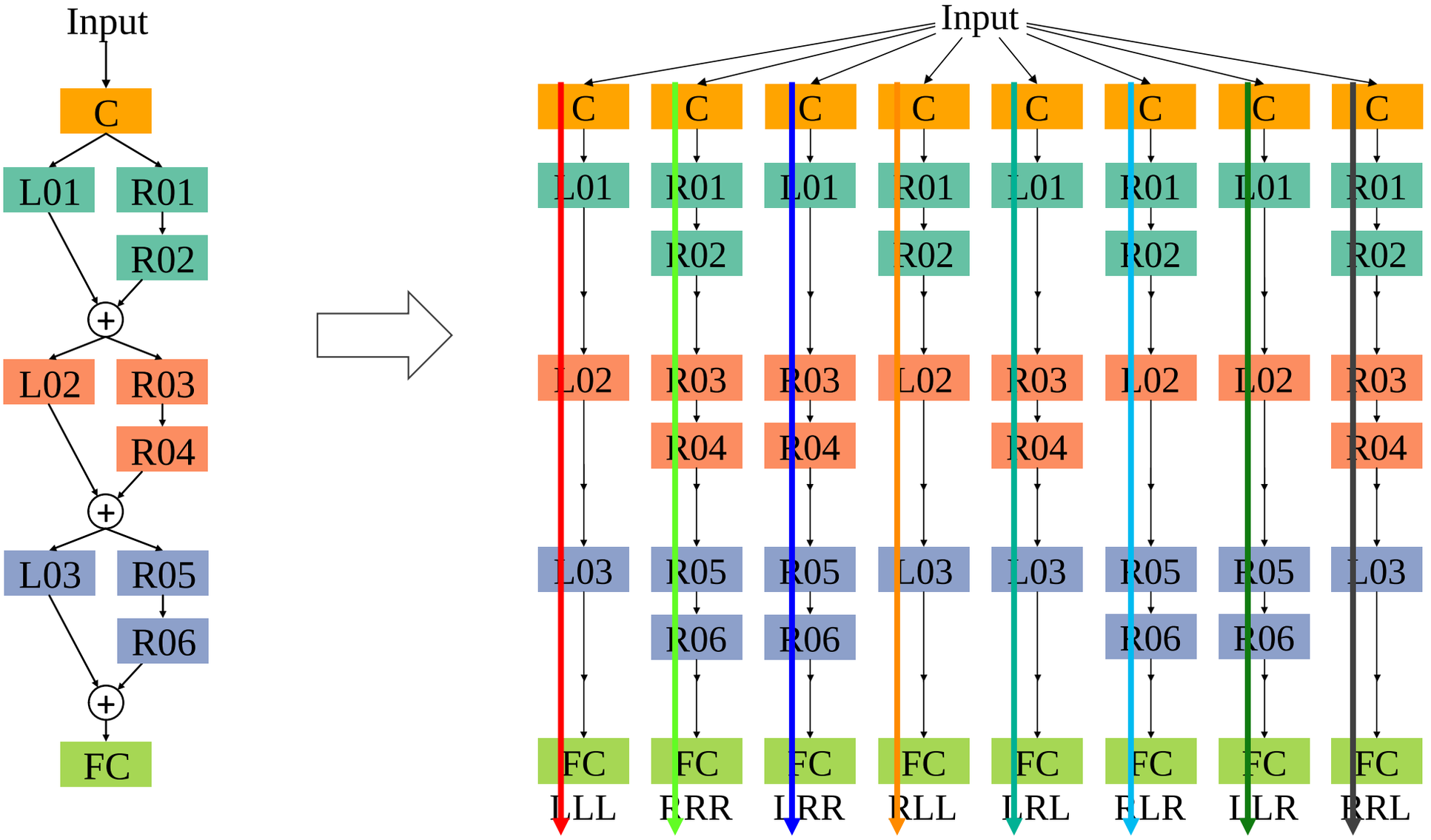


A deeply-fused net is a multi-path multi-scale network

A deeply-fused net is a multi-path multi-scale network

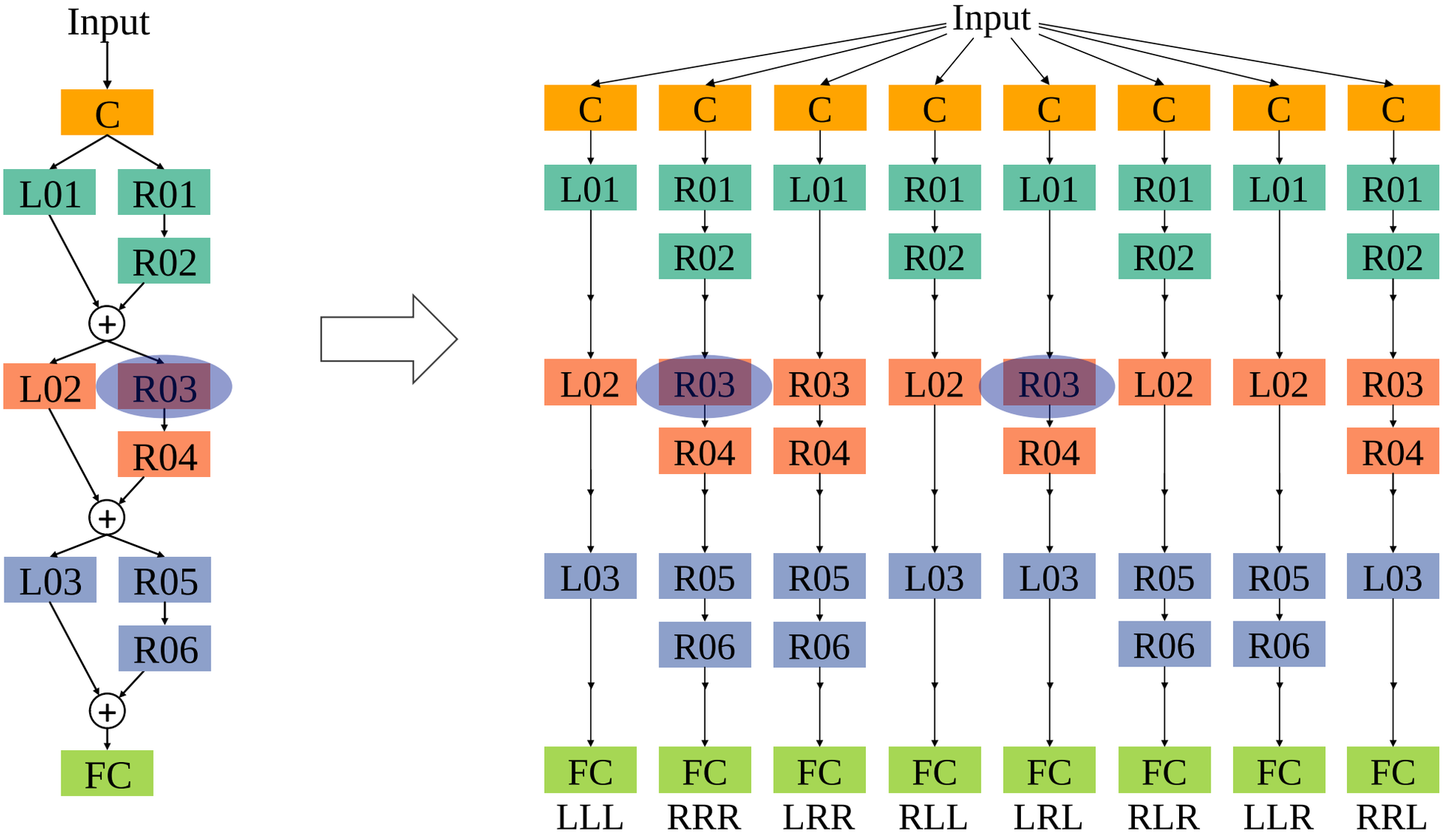


A deeply-fused net is a multi-path multi-scale network

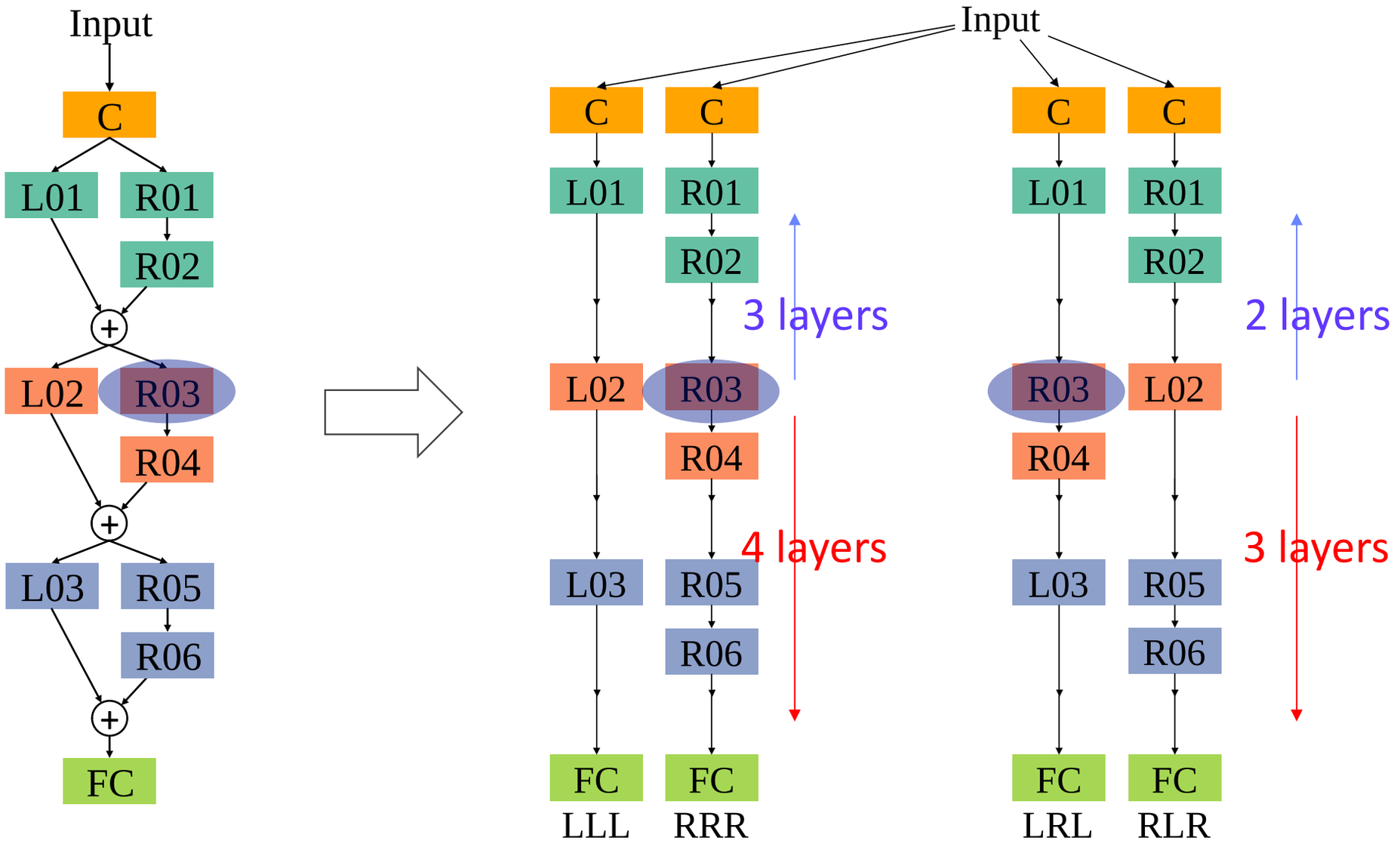


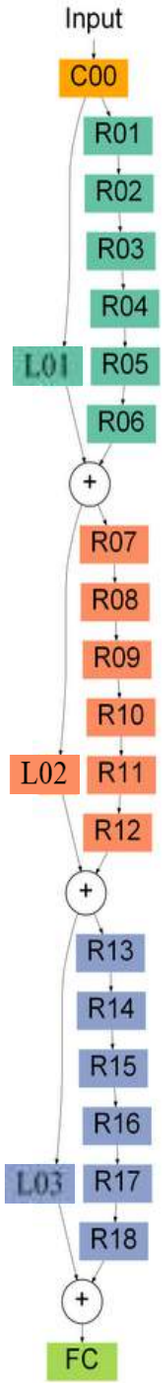
Each layer has an express way to input and output

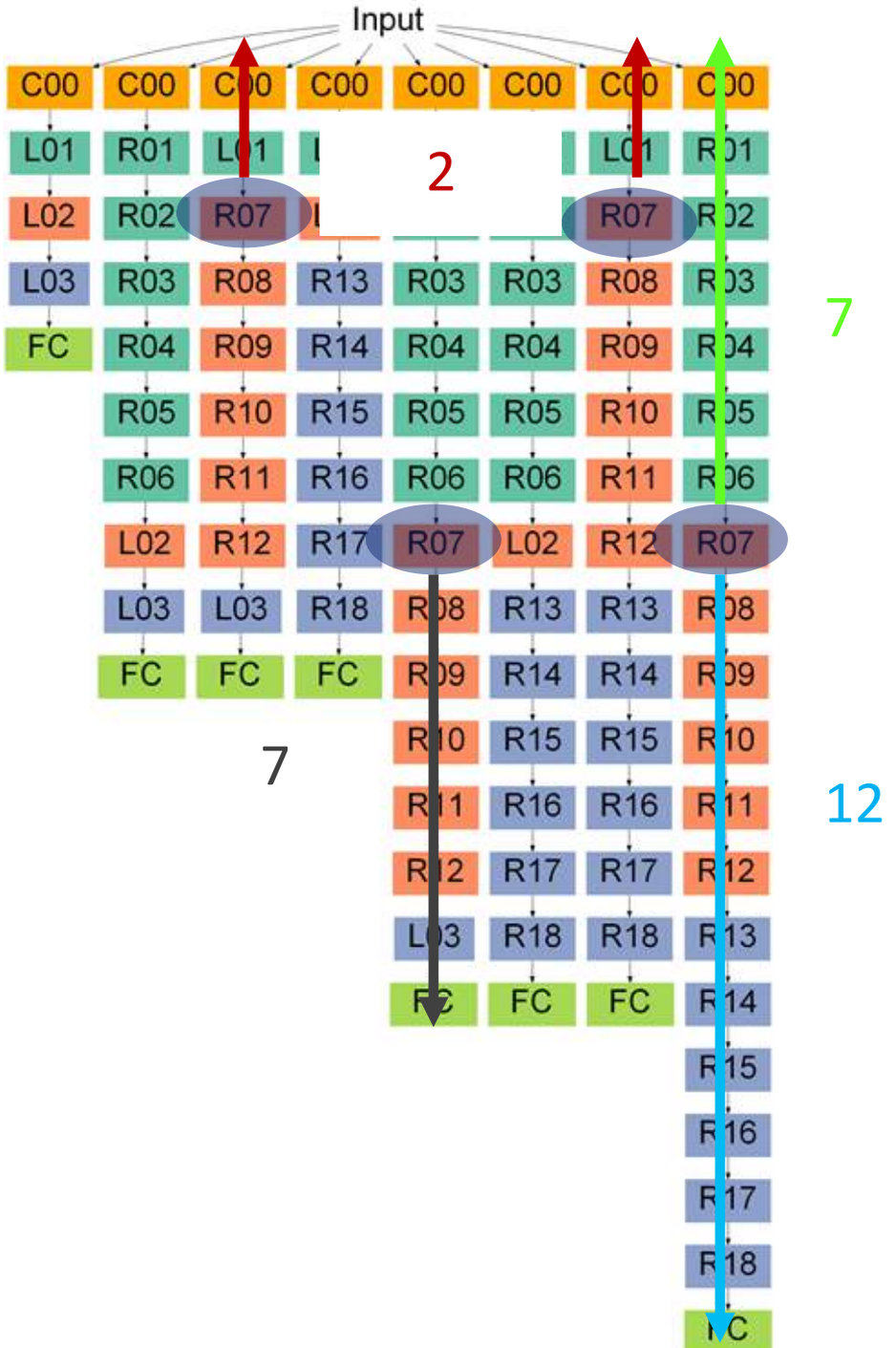
Each layer has an express way to input and output



Each layer has an express way to input and output




Input
C00
R01
R02
R03
R04
L01
R05
R06
+
R07
R08
R09
R10
L02
R11
R12
+
R13
R14
R15
R16
L03
R17
R18
+
FC


Input
C00
L01
R01
L02
R02
R07
L03
R03
R08
R13
FC
R04
R09
R14
R05
R10
R15
R06
R11
R16
L02
R12
R17
L03
L03
R18
FC
FC
FC
2
7
7
12

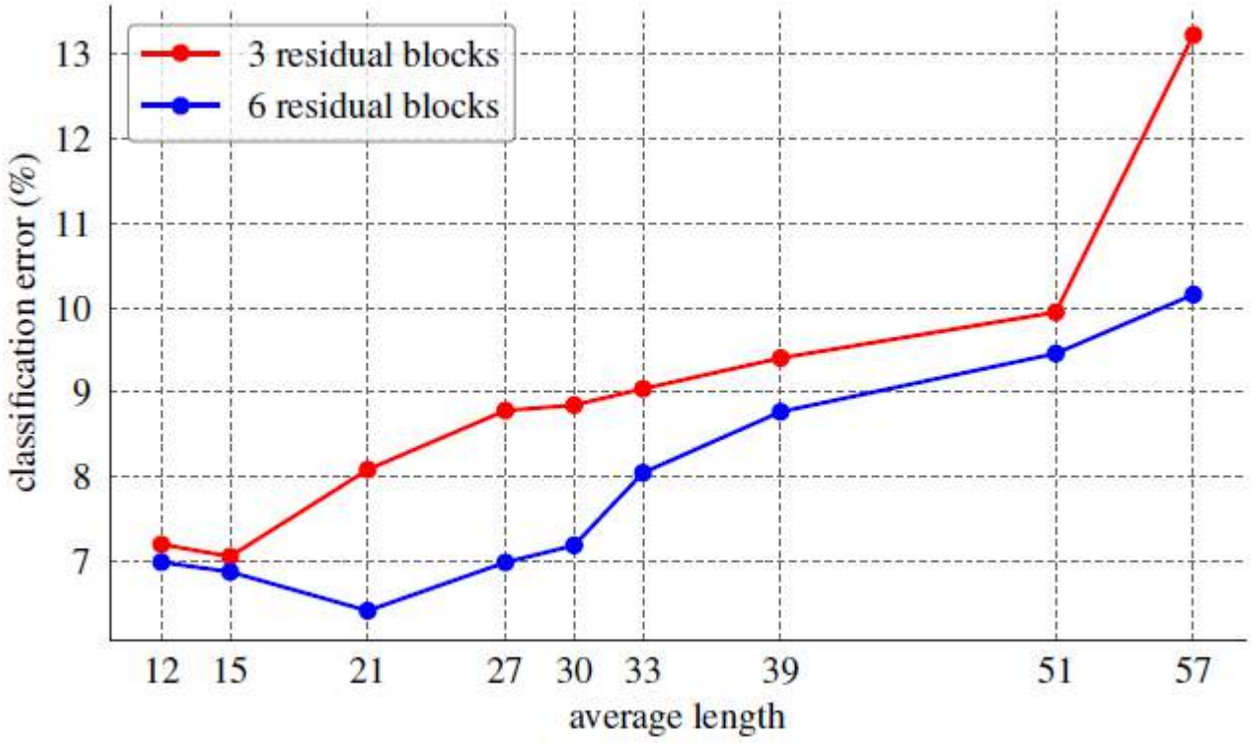
Unifying GoogLeNets, Highway, ResNets

Deep fusion

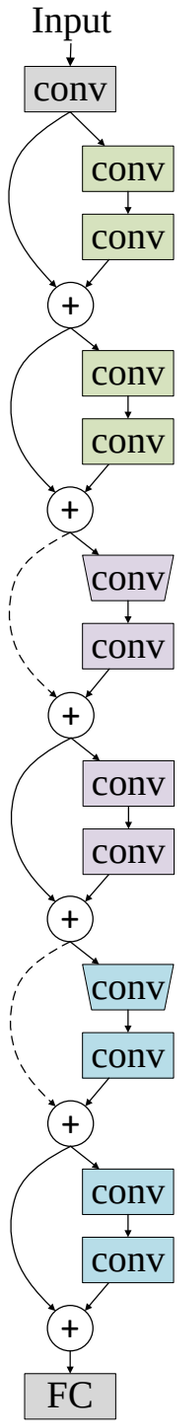
- Multiple paths
- Long and short
- Express way between layers
- Weight sharing

Are ultra-deep networks really necessary?

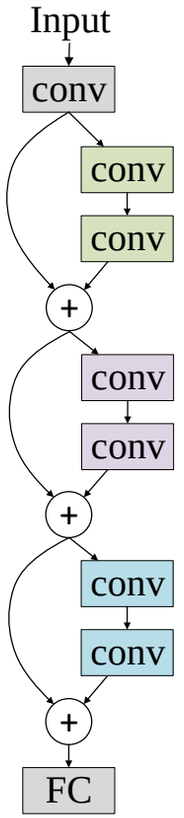
How depth affects the performance



CIFAR 10



6 blocks

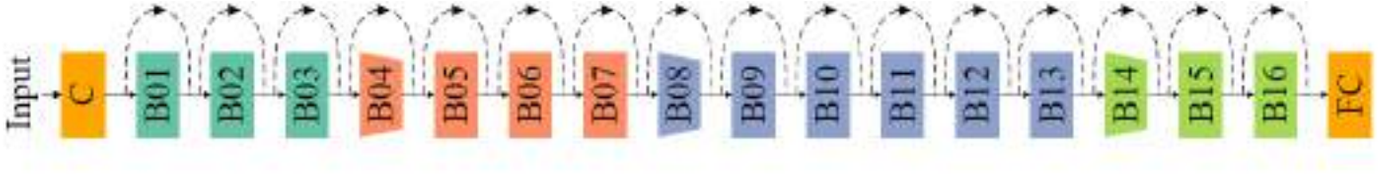


L layers

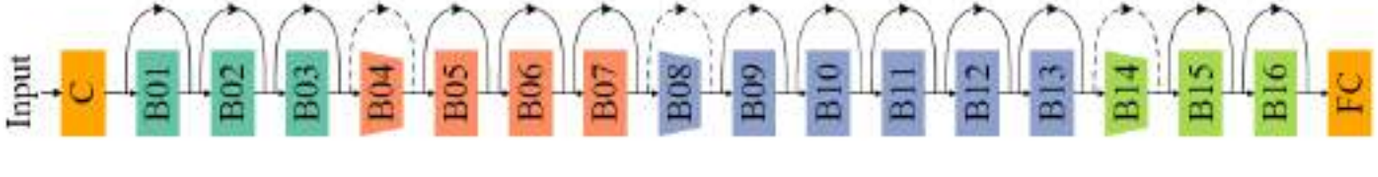
3 blocks

How depth affects the performance on ImageNet

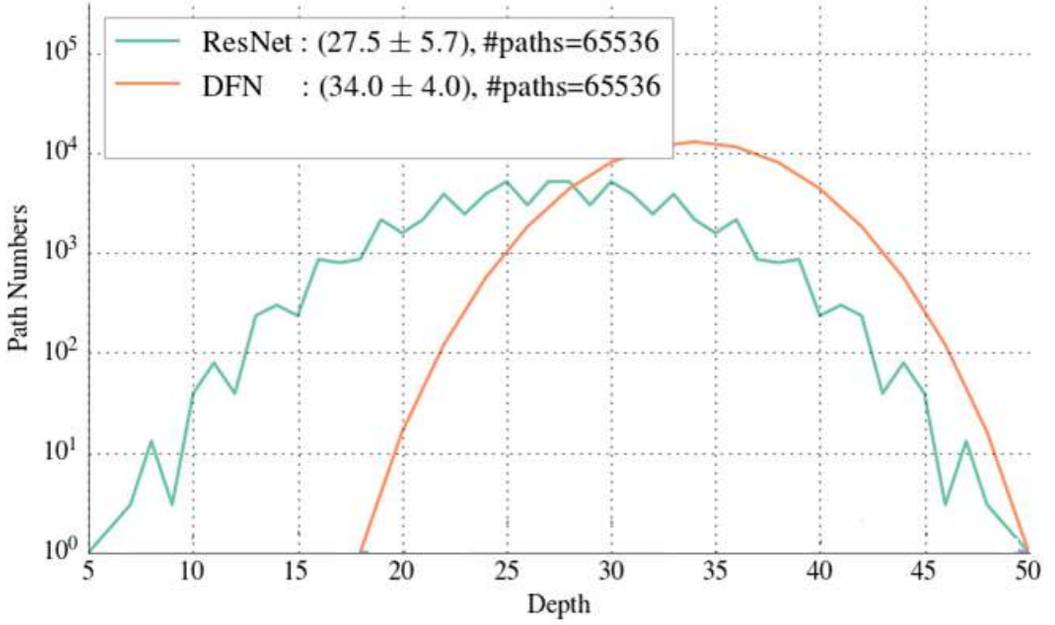
	ResNet	DFN
Top-1 validation error	24.94	25.10
Top-5 validation error	7.46	7.85



DFN

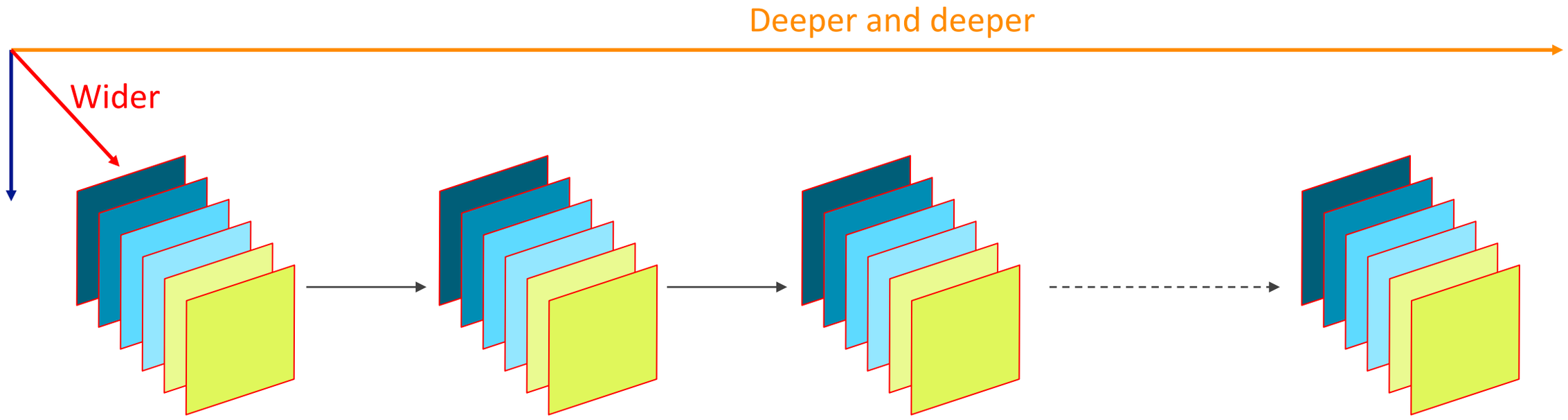


ResNet

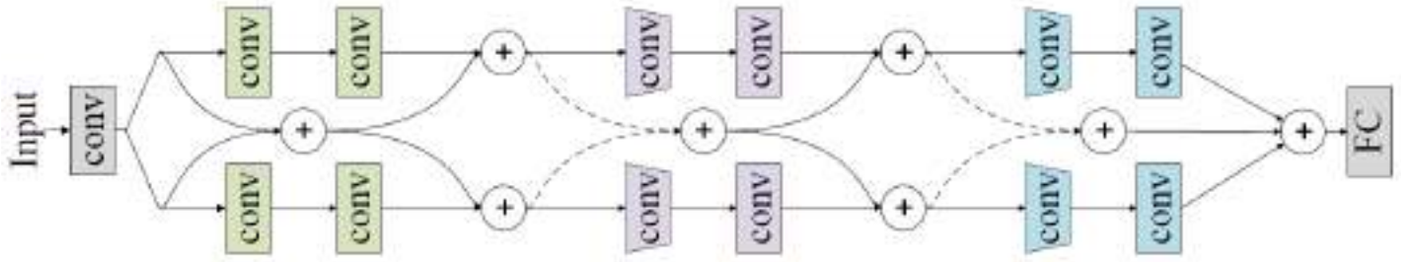


Ultra-deep is not necessary!

Go wider

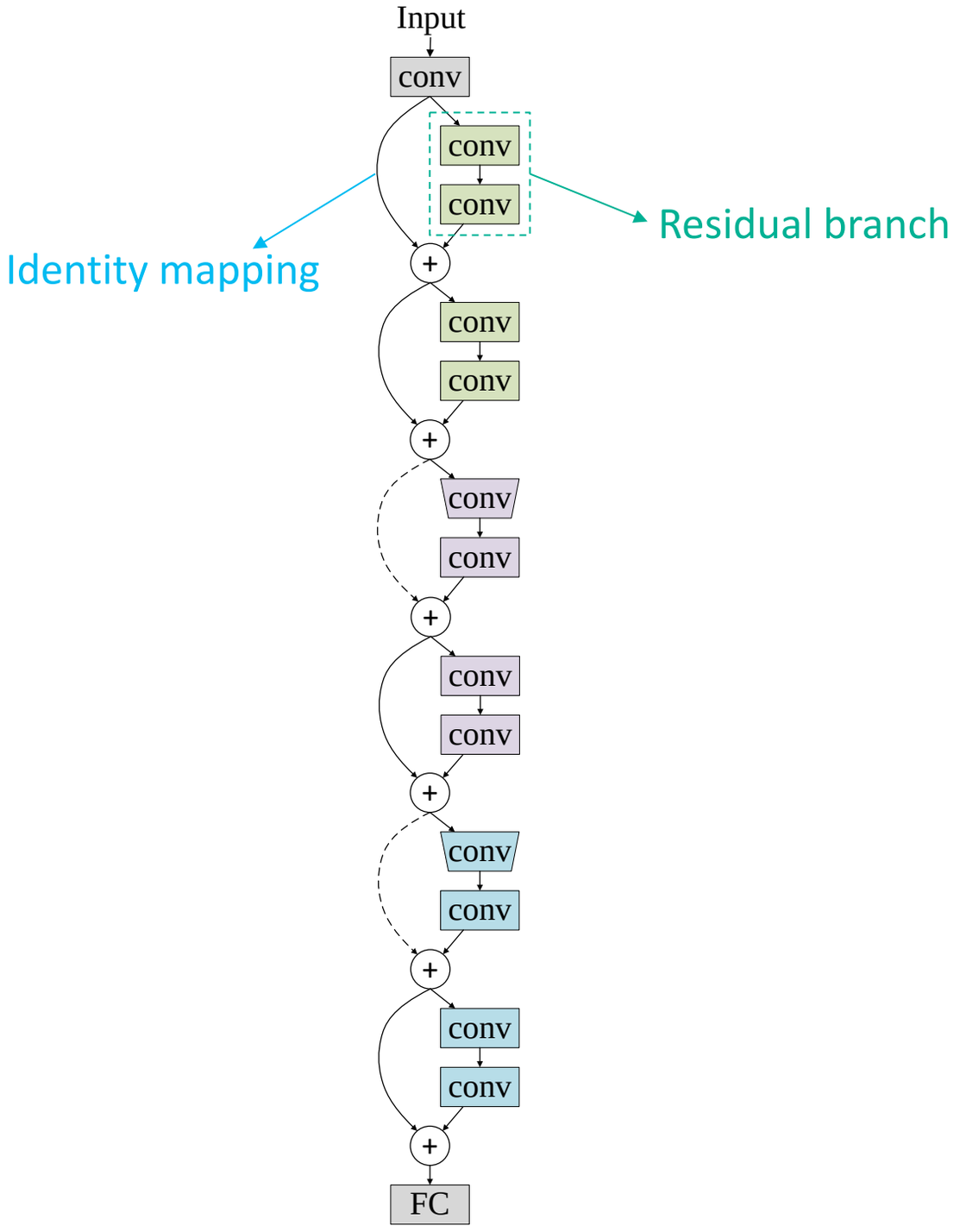


Going Less Deep but Wider:
Assembling branches in parallel with merge and run mappings

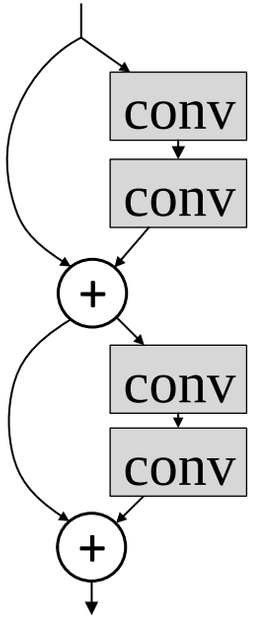


Liming Zhao, Jingdong Wang, Xi Li, Zhuowen Tu, Wenjun Zeng: Deep Convolutional Neural Networks with Merge-and-Run Mappings. (2017)

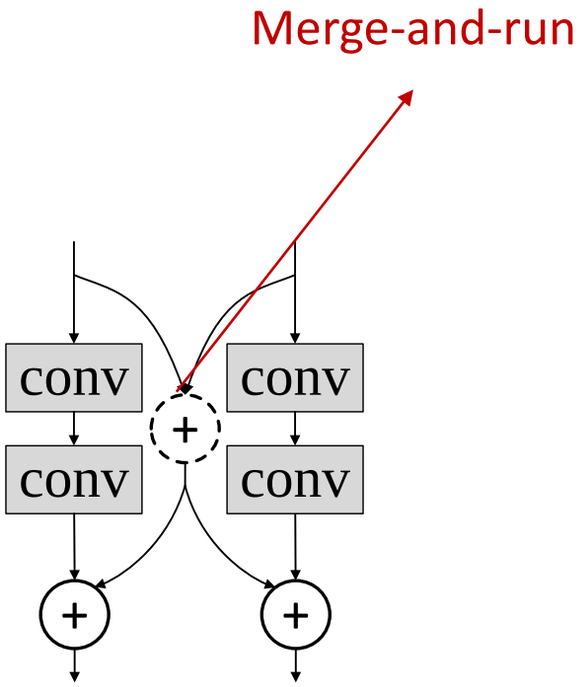
Blog: 深度神经网络中深度究竟带来了什么？ (http://www.msra.cn/zh-cn/news/blogs/2016/12/deep-neural-network-20161212.aspx)


Input
conv
conv
conv
Identity mapping
Residual branch
+
conv
conv
+
conv
conv
+
conv
conv
+
conv
conv
+
conv
conv
+
FC

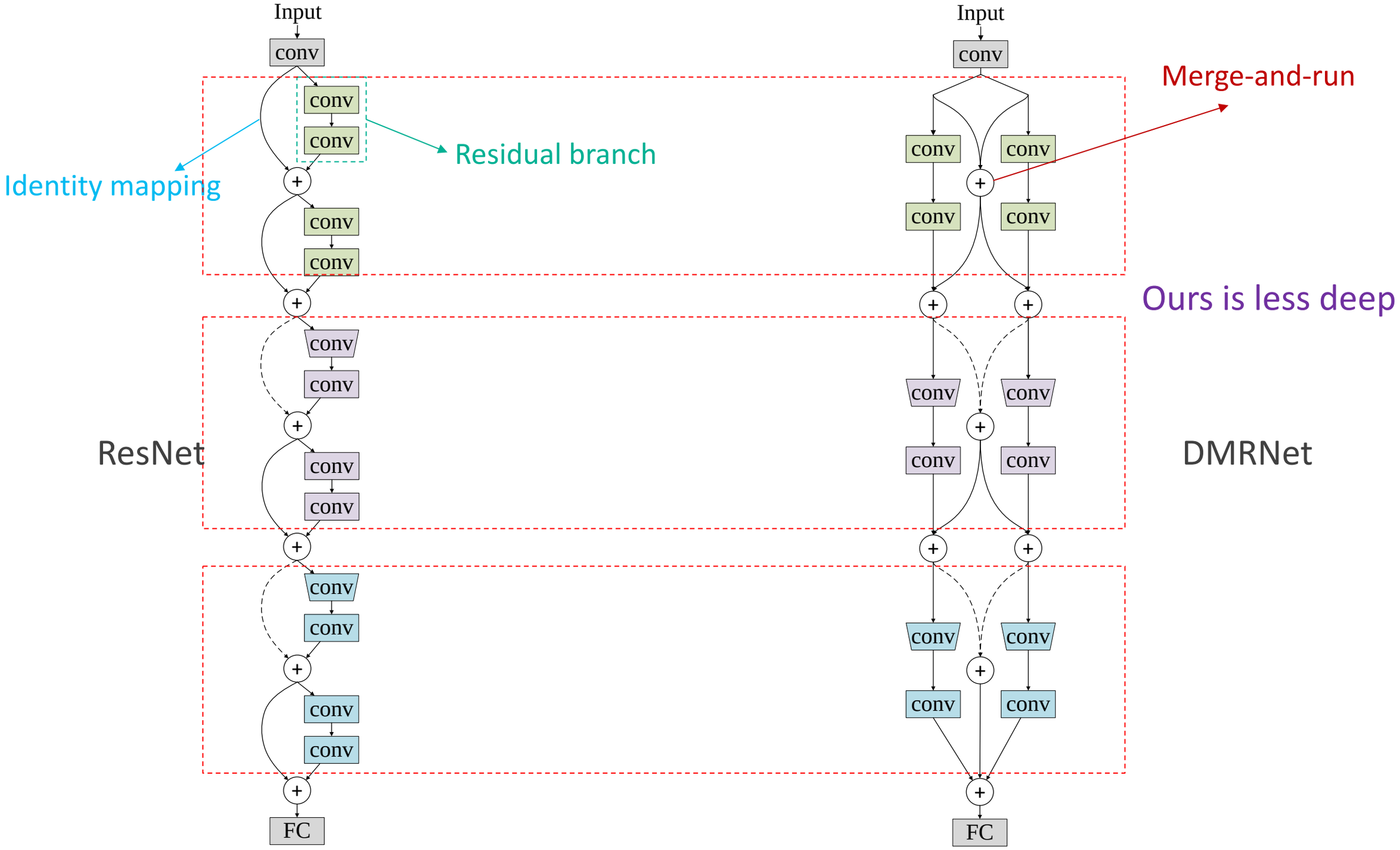
Assemble residual branches: from sequential to parallel



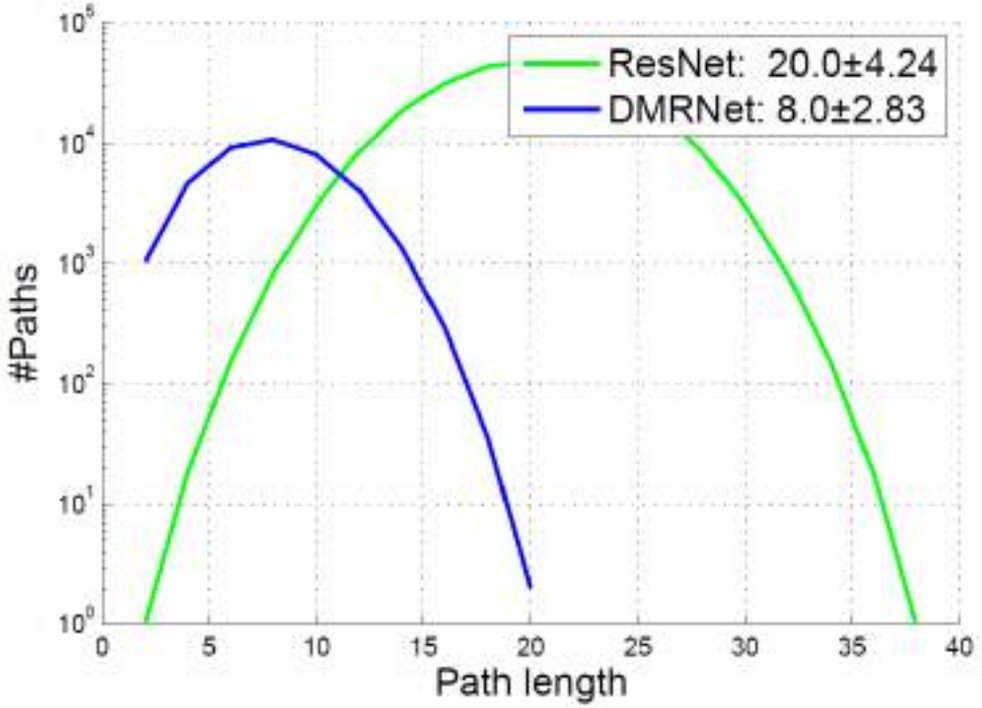
sequential



parallel

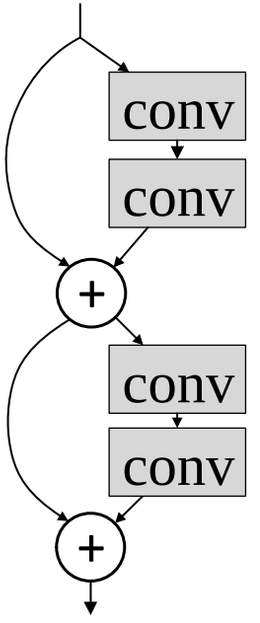

Input
conv
conv
conv
Identity mapping
Residual branch
conv
conv
conv
conv
conv
conv
conv
conv
conv
conv
conv
conv
FC
ResNet
Input
conv
Merge-and-run
conv
conv
conv
conv
Ours is less deep
conv
conv
conv
conv
conv
conv
conv
conv
conv
conv
conv
conv
FC
DMRNet

(1) Path length distribution

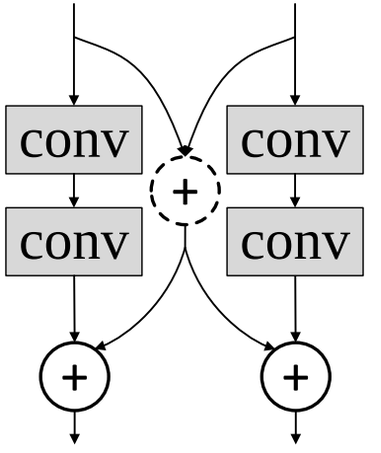


9 merge-and-run blocks

(2) Parallel assembly leads to wider networks

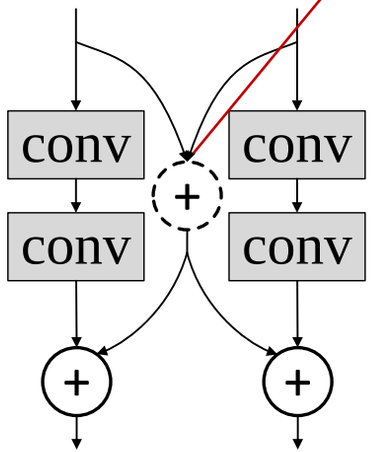


Width=d



Width=~2d

(3) Merge-and run mappings improve information flow



parallel

Merge-and-run

$$\begin{bmatrix} \mathbf{x}_{2(t+1)} \\ \mathbf{x}_{2(t+1)+1} \end{bmatrix} = \begin{bmatrix} H_{2t}(\mathbf{x}_{2t}) \\ H_{2t+1}(\mathbf{x}_{2t+1}) \end{bmatrix} + \frac{1}{2}\begin{bmatrix} \mathbf{I} & \mathbf{I} \\ \mathbf{I} & \mathbf{I} \end{bmatrix}\begin{bmatrix} \mathbf{x}_{2t} \\ \mathbf{x}_{2t+1} \end{bmatrix} = \mathbf{M}$$

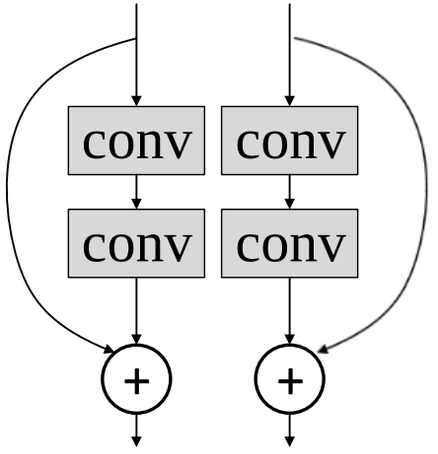
Idempotent mapping: $\mathbf{M}^n = \mathbf{M}$

$$\begin{bmatrix} \mathbf{x}_{2(t+1)} \\ \mathbf{x}_{2(t+1)+1} \end{bmatrix} = \begin{bmatrix} H_{2t}(\mathbf{x}_{2t}) \\ H_{2t+1}(\mathbf{x}_{2t+1}) \end{bmatrix} + \mathbf{M}\sum_{i=t'}^{t-1}\begin{bmatrix} H_{2t}(\mathbf{x}_{2i'}) \\ H_{2t+1}(\mathbf{x}_{2i'+1}) \end{bmatrix} + \mathbf{M}\begin{bmatrix} \mathbf{x}_{2t'} \\ \mathbf{x}_{2t'+1} \end{bmatrix}$$

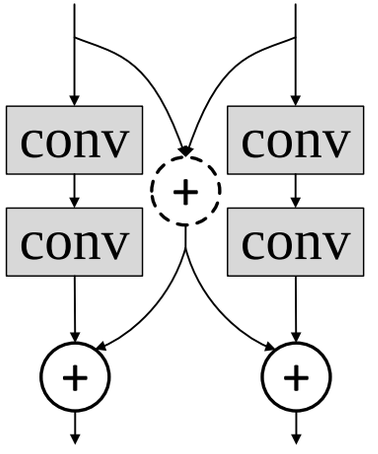
Quick path from *t'* to *t*

Merge-and-run mappings vs. Identity mappings

L	CIFAR-10		CIFAR-100	
	Identity	Merge-and-run	Identity	Merge-and-run
48	5.21	4.99	25.31	24.73
96	5.19	4.84	24.16	23.98



identity



Merge-and-run

Experimental results - datasets

	#(training images)	#(testing images)	#classes
CIFAR-10	50,000	10,000	10
CIFAR-100	50,000	10,000	100
SVHN	73,257 + 531,131	26,032	10

Comparison with state-of-the-arts

Method	Depth	#Params.	CIFAR-10	CIFAR-100	SVHN
DSN	-	-	7.97	34.57	1.92
FractalNet with DO/DP	21	38.6M	5.22	23.30	2.01
	21	38.6M	4.60	23.73	1.87
ResNet	110	1.7M	6.41	27.22	2.01
Multi ResNet	200	10.2M	4.35	20.42	-
Wide ResNet	16	11.0M	4.81	22.07	-
	28	36.5M	4.17	20.50	-
DenseNet	40	1.0M	5.24	24.42	1.79
	100	27.2M	3.74	19.25	1.59
DMRNet (ours)	56	1.7M	4.94	24.46	1.66
DMRNet-Wide (ours)	32	14.9M	3.94	19.25	**1.51**
DMRNet-Wide (ours)	50	24.8M	**3.57**	**19.00**	1.55

Comparison with state-of-the-arts

Method	Depth	#Params.	CIFAR-10	CIFAR-100	SVHN
DSN	-	-	7.97	34.57	1.92
FractalNet with DO/DP	21	38.6M	5.22	23.30	2.01
	21	38.6M	4.60	23.73	1.87
ResNet	110	1.7M	6.41	27.22	2.01
Multi ResNet	200	10.2M	4.35	20.42	-
Wide ResNet	16	11.0M	4.81	22.07	-
	28	36.5M	4.17	20.50	-
DenseNet	40	1.0M	5.24	24.42	1.79
	100	27.2M	3.74	19.25	1.59
DMRNet (ours)	56	1.7M	4.94	24.46	1.66
DMRNet-Wide (ours)	32	14.9M	3.94	19.25	**1.51**
DMRNet-Wide (ours)	50	24.8M	**3.57**	**19.00**	1.55

Comparison with state-of-the-arts

Method	Depth	#Params.	CIFAR-10	CIFAR-100	SVHN
ResNet	110	1.7M	6.41	27.22	2.01
DMRNet (ours)	56	1.7M	4.94	24.46	1.66

Comparison with ResNets

#parameters	L	ResNets	DMRNets
0.4M	12	6.62	6.48
0.6M	18	5.93	5.79
0.8M	24	5.60	5.47
1.0M	30	5.50	5.10
1.2M	36	5.35	5.18
1.5M	48	5.26	4.99
1.7M	54	5.24	4.96
3.1M	96	5.47	4.84

CIFAR-10 classification error, average over 5 runs

Comparison with ResNets

#parameters	L	ResNets	DMRNets
0.4M	12	29.69	29.62
0.6M	18	27.90	27.80
0.8M	24	27.03	26.76
1.0M	30	26.44	25.87
1.2M	36	26.00	25.41
1.5M	48	25.44	24.73
1.7M	54	24.56	24.41
3.1M	96	24.41	23.98

CIFAR-100 classification error, average over 5 runs