DEEP RESIDUAL NETWORKS AND OTHER ADVANCED CNN ARCHITECTURES

THE ROADMAP

- INTRODUCTION
- DEEP RESIDUAL NETWORKS
- WHY DO DEEP RESIDUAL NETWORKS WORK?
- SURVEY OF ADVANCED CNN ARCHITECTURES
- CONCLUSION

THE ROADMAP

- **INTRODUCTION**
- DEEP RESIDUAL NETWORKS
- WHY DO DEEP RESIDUAL NETWORKS WORK?
- SURVEY OF ADVANCED CNN ARCHITECTURES
- CONCLUSION

Introduction

Deep Residual Networks (ResNets)

- "Deep Residual Learning for Image Recognition". CVPR 2016 (next week)
- A simple and clean framework of training "very" deep nets
- State-of-the-art performance for
  - Image classification
  - Object detection
  - Semantic segmentation
  - and more…

Kaiming He, Xiangyu Zhang, Shaoqing Ren, & Jian Sun. "Deep Residual Learning for Image Recognition". CVPR 2016.

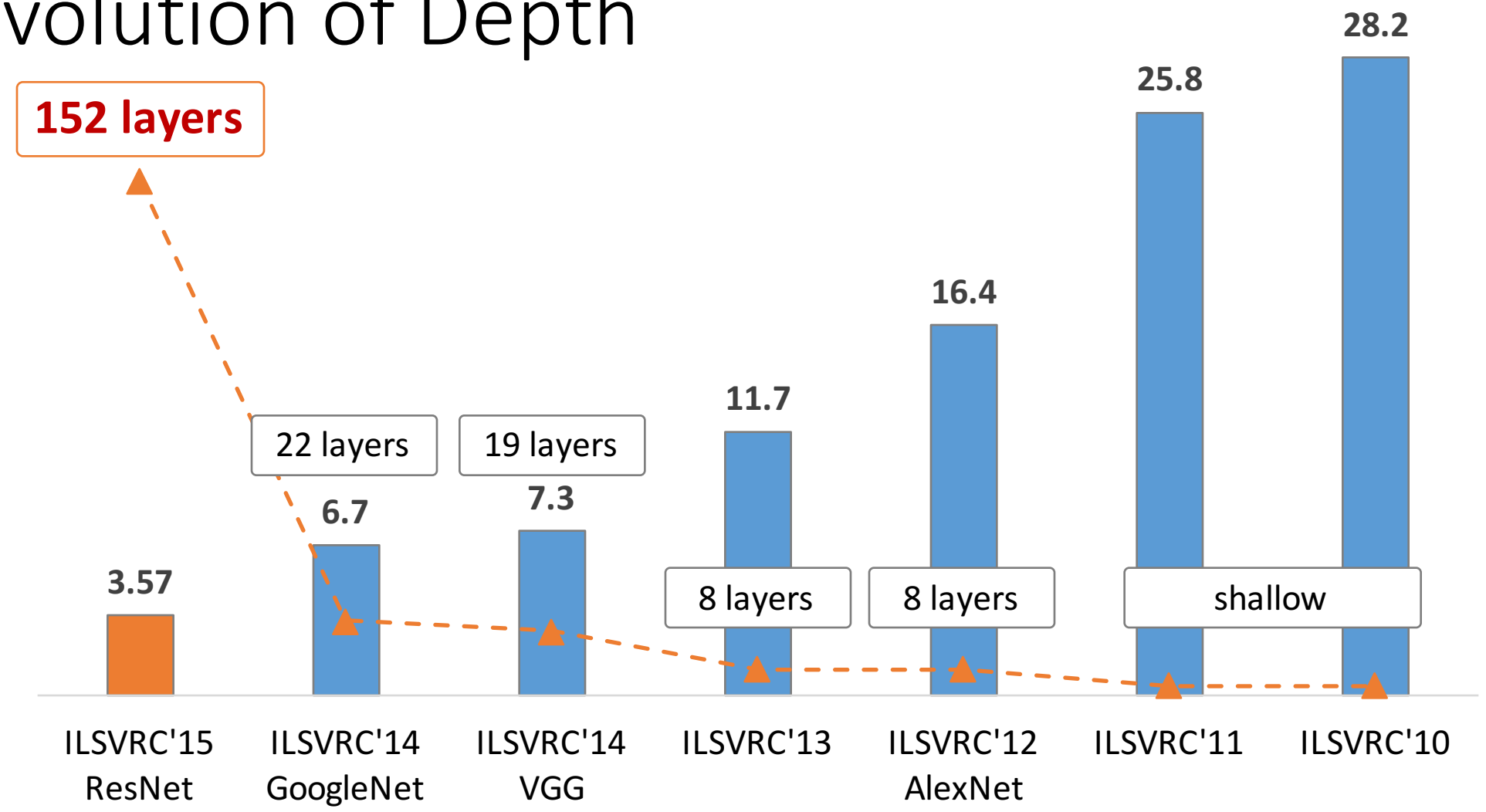
ResNets @ ILSVRC & COCO 2015 Competitions

- **1st places in all five main tracks**
  - ImageNet Classification: "*Ultra-deep*" 152-layer nets
  - ImageNet Detection: 16% better than 2nd
  - ImageNet Localization: 27% better than 2nd
  - COCO Detection: 11% better than 2nd
  - COCO Segmentation: 12% better than 2nd

*improvements are relative numbers

Kaiming He, Xiangyu Zhang, Shaoqing Ren, & Jian Sun. "Deep Residual Learning for Image Recognition". CVPR 2016.

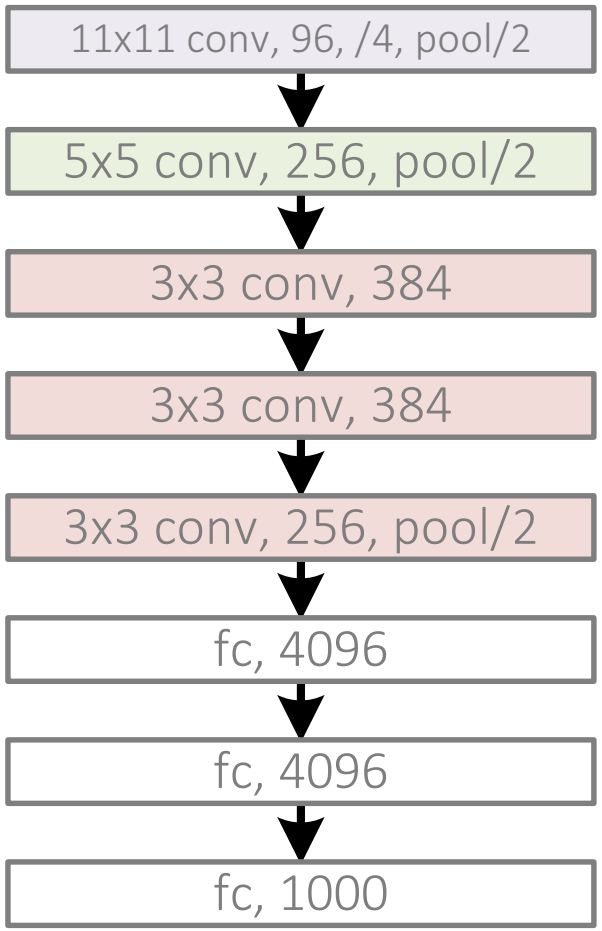
Revolution of Depth



Kaiming He, Xiangyu Zhang, Shaoqing Ren, & Jian Sun. "Deep Residual Learning for Image Recognition". CVPR 2016.

Revolution of Depth

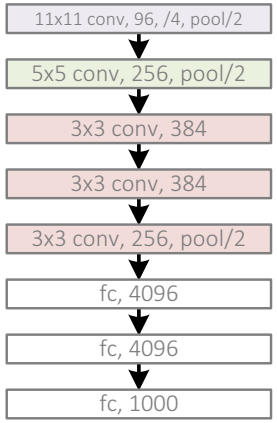
AlexNet, 8 layers
(ILSVRC 2012)



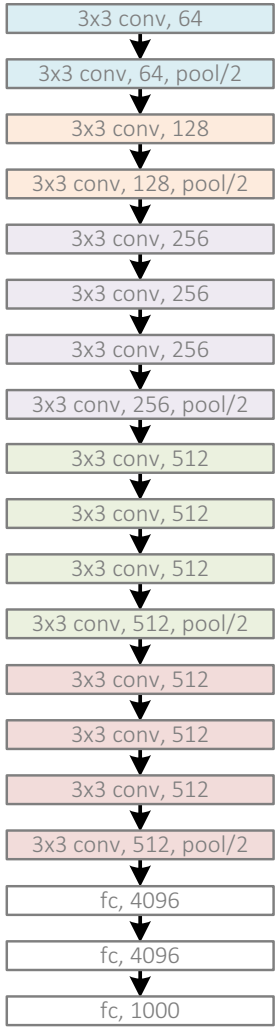
Kaiming He, Xiangyu Zhang, Shaoqing Ren, & Jian Sun. "Deep Residual Learning for Image Recognition". CVPR 2016.

Revolution of Depth

AlexNet, 8 layers
(ILSVRC 2012)



VGG, 19 layers
(ILSVRC 2014)



GoogleNet, 22 layers
(ILSVRC 2014)

Kaiming He, Xiangyu Zhang, Shaoqing Ren, & Jian Sun. "Deep Residual Learning for Image Recognition". CVPR 2016.

Revolution of Depth

AlexNet, 8 layers
(ILSVRC 2012)

VGG, 19 layers
(ILSVRC 2014)

ResNet, 152 layers
(ILSVRC 2015)

Kaiming He, Xiangyu Zhang, Shaoqing Ren, & Jian Sun. "Deep Residual Learning for Image Recognition". CVPR 2016.

THE ROADMAP

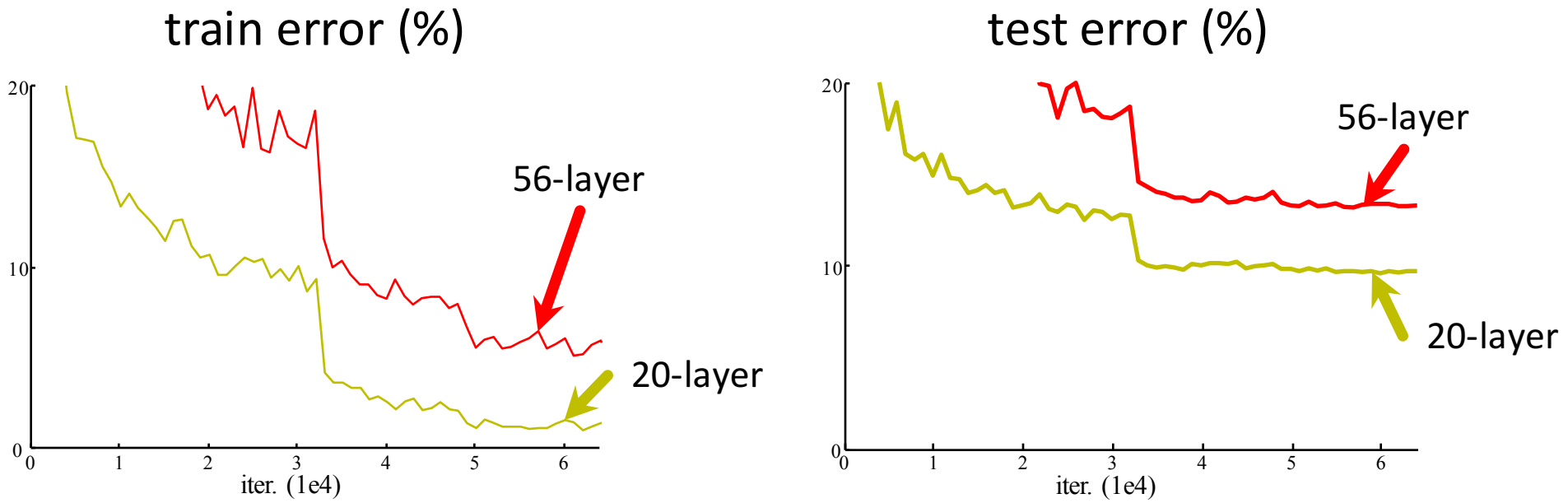
- INTRODUCTION
- **DEEP RESIDUAL NETWORKS**
- WHY DO DEEP RESIDUAL NETWORKS WORK?
- SURVEY OF ADVANCED CNN ARCHITECTURES
- CONCLUSION

Deep Residual Networks

From 10 layers to 100 layers

Simply stacking layers?

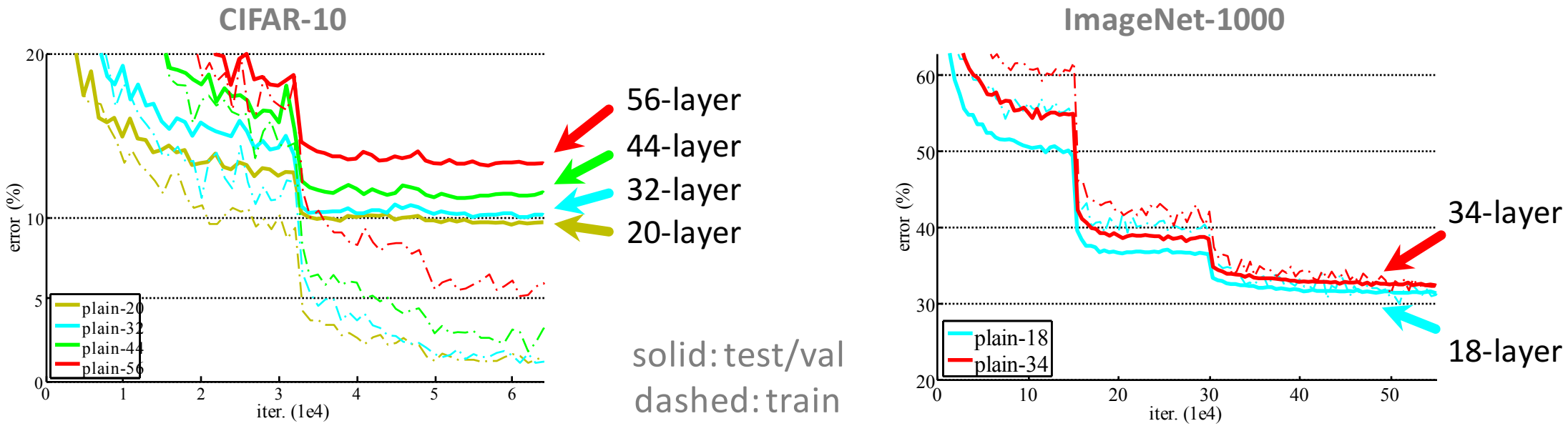
CIFAR-10



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

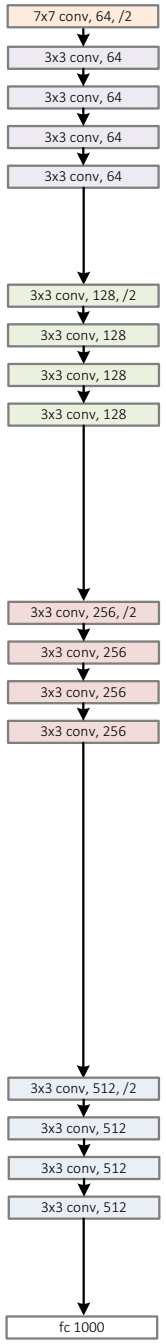
Kaiming He, Xiangyu Zhang, Shaoqing Ren, & Jian Sun. "Deep Residual Learning for Image Recognition". CVPR 2016.

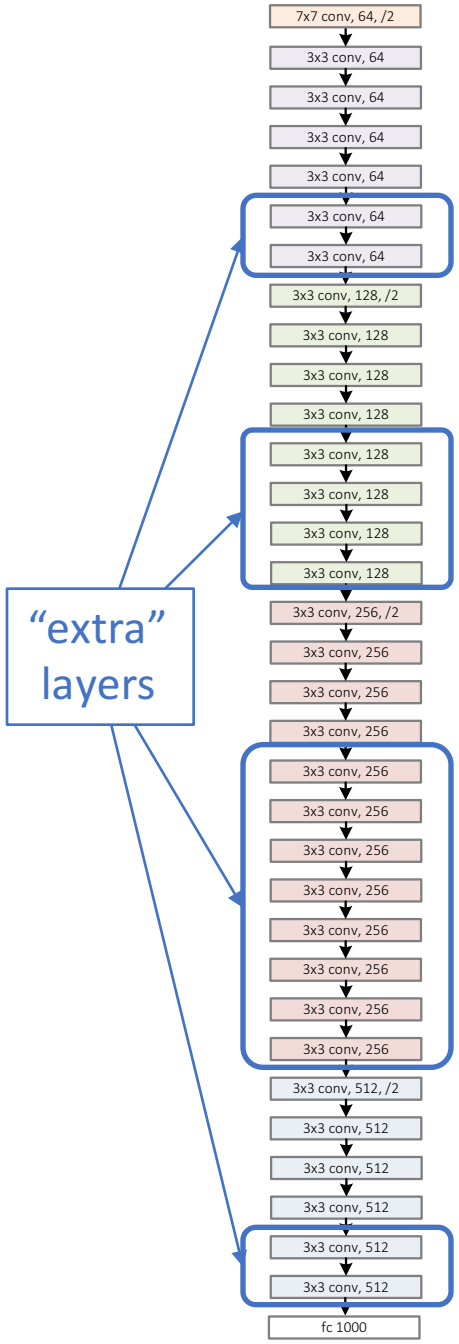
Simply stacking layers?



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

a shallower model (18 layers)





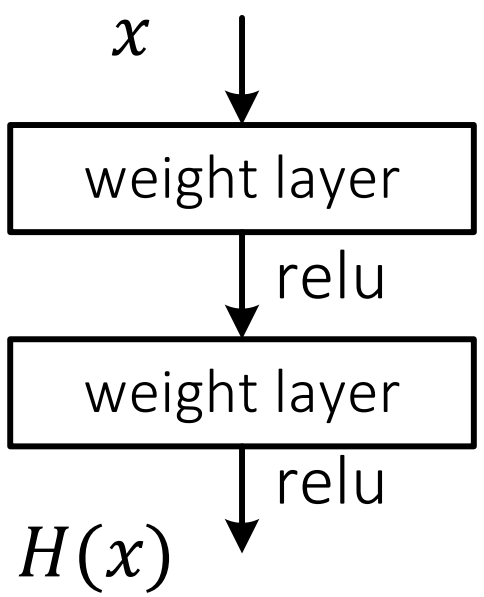
a deeper counterpart (34 layers)

- Richer solution space
- A deeper model should not have **higher training error**
- A solution *by construction*:
  - original layers: copied from a learned shallower model
  - extra layers: set as identity
  - at least the same training error
- Optimization difficulties: solvers cannot find the solution when going deeper…

Kaiming He, Xiangyu Zhang, Shaoqing Ren, & Jian Sun. "Deep Residual Learning for Image Recognition". CVPR 2016.

Deep Residual Learning

- Plaint net



any two
stacked layers

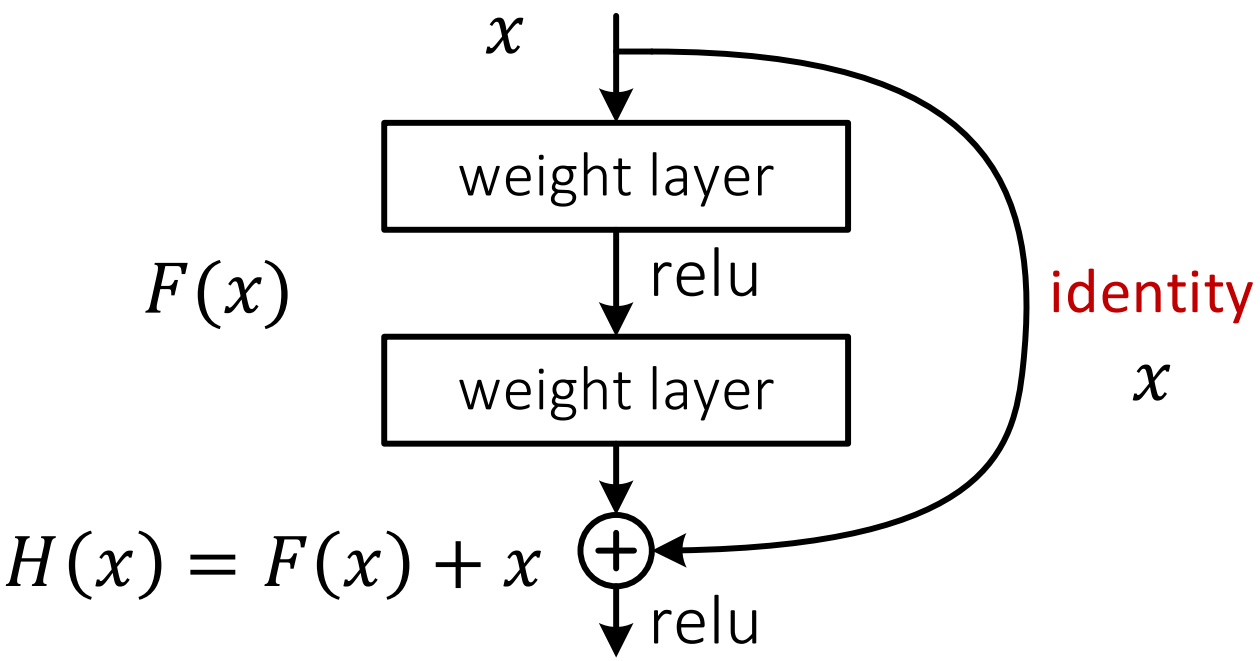
$H(x)$ is any desired mapping,

hope the 2 weight layers fit $H(x)$

Kaiming He, Xiangyu Zhang, Shaoqing Ren, & Jian Sun. "Deep Residual Learning for Image Recognition". CVPR 2016.

Deep Residual Learning

- Residual net



$H(x)$ is any desired mapping,

~~hope the 2 weight layers fit $H(x)$~~

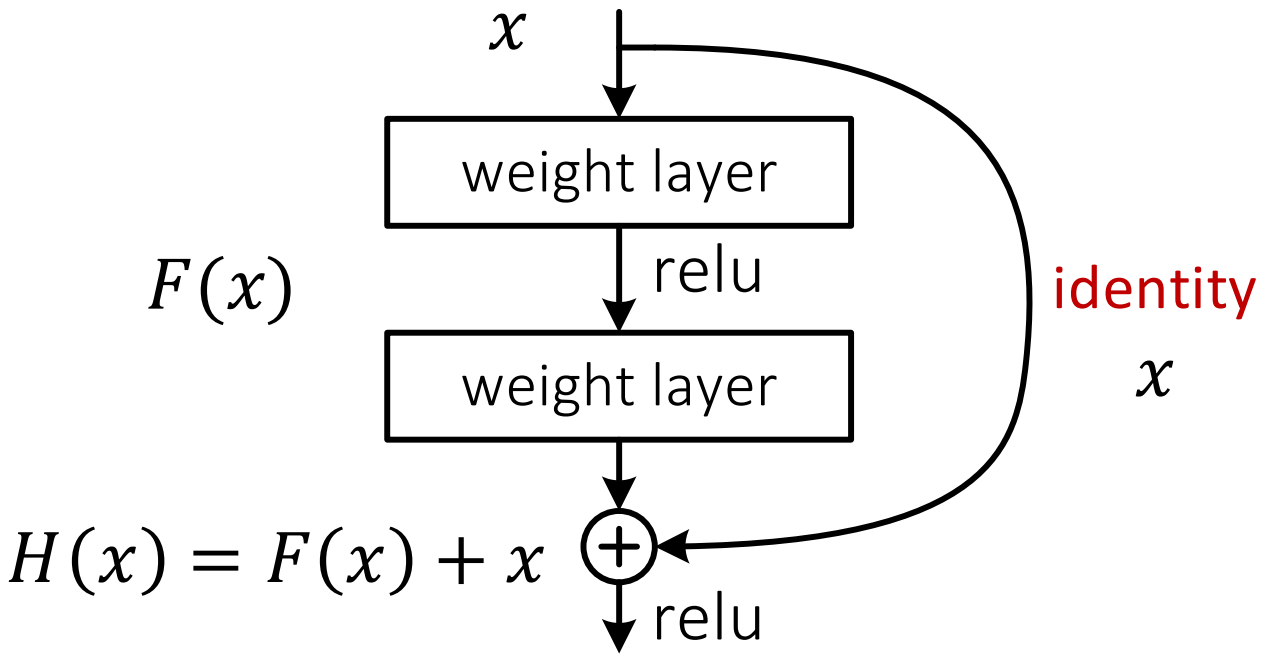
hope the 2 weight layers fit $F(x)$

let $H(x) = F(x) + x$

Kaiming He, Xiangyu Zhang, Shaoqing Ren, & Jian Sun. "Deep Residual Learning for Image Recognition". CVPR 2016.

Deep Residual Learning

- $F(x)$ is a residual mapping w.r.t. identity

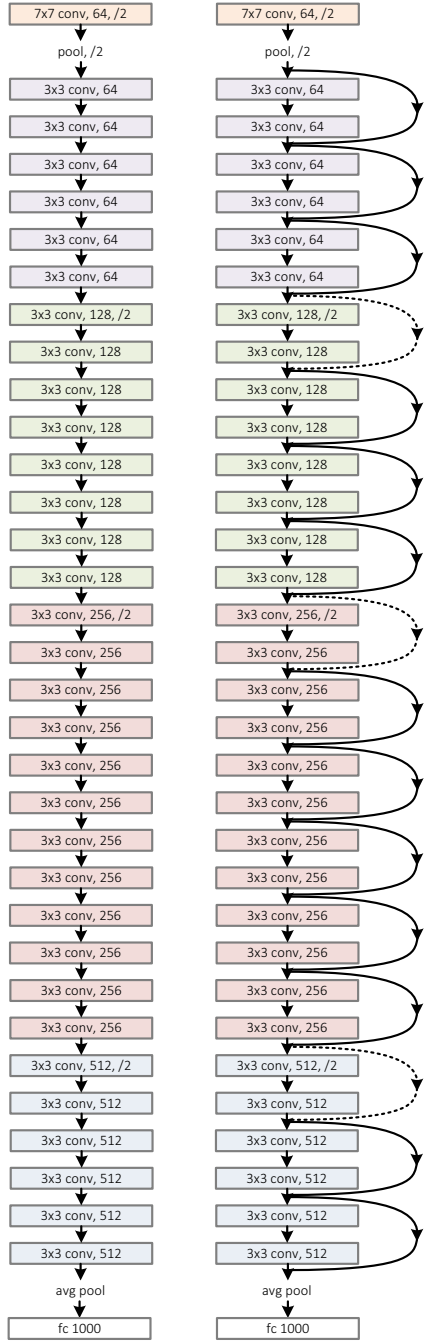


- If identity were optimal, easy to set weights as 0
- If optimal mapping is closer to identity, easier to find small fluctuations

Kaiming He, Xiangyu Zhang, Shaoqing Ren, & Jian Sun. "Deep Residual Learning for Image Recognition". CVPR 2016.

Network “Design”

plain net



ResNet

- Keep it simple
- Our basic design (VGG-style)
  - all 3x3 conv (almost)
  - spatial size /2 => # filters x2 (~same complexity per layer)
  - Simple design; just deep!
- Other remarks:
  - no hidden fc
  - no dropout

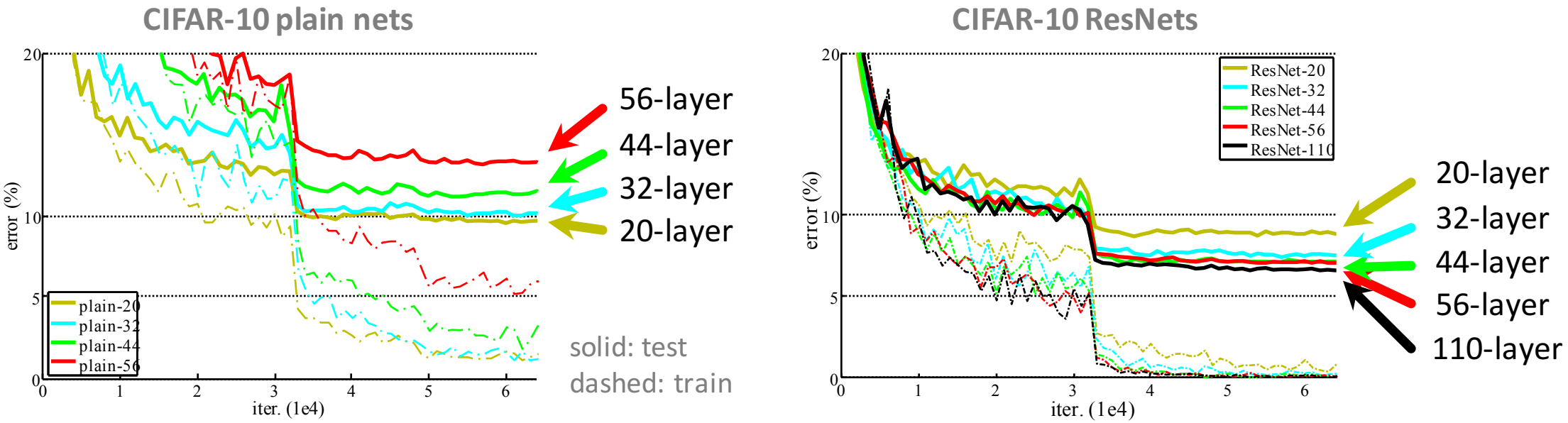
Kaiming He, Xiangyu Zhang, Shaoqing Ren, & Jian Sun. “Deep Residual Learning for Image Recognition”. CVPR 2016.

Training

- All plain/residual nets are trained from scratch
- All plain/residual nets use Batch Normalization
- Standard hyper-parameters & augmentation

Kaiming He, Xiangyu Zhang, Shaoqing Ren, & Jian Sun. “Deep Residual Learning for Image Recognition”. CVPR 2016.

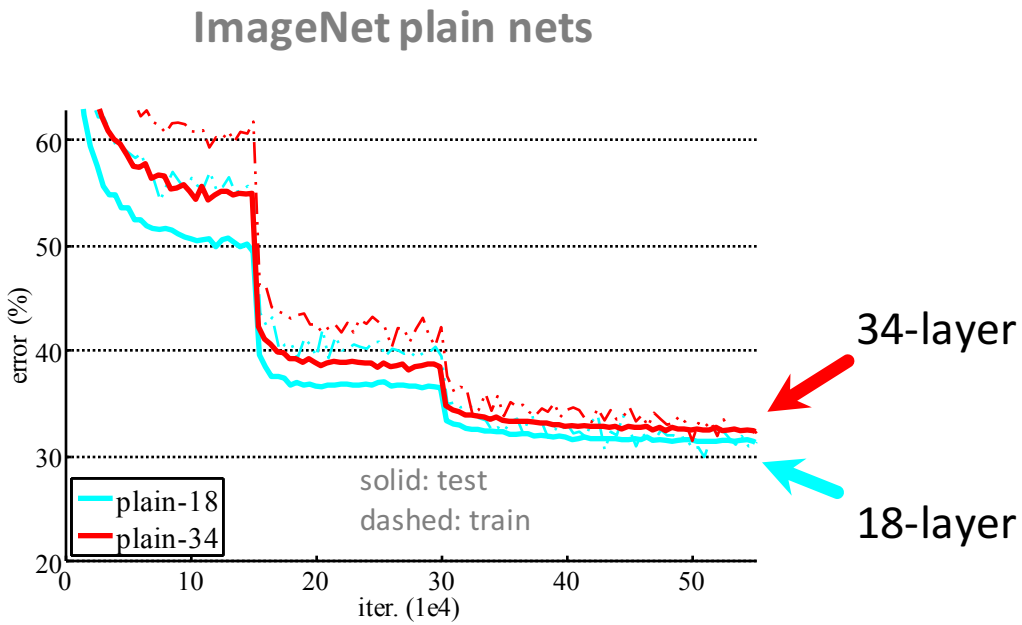
CIFAR-10 experiments

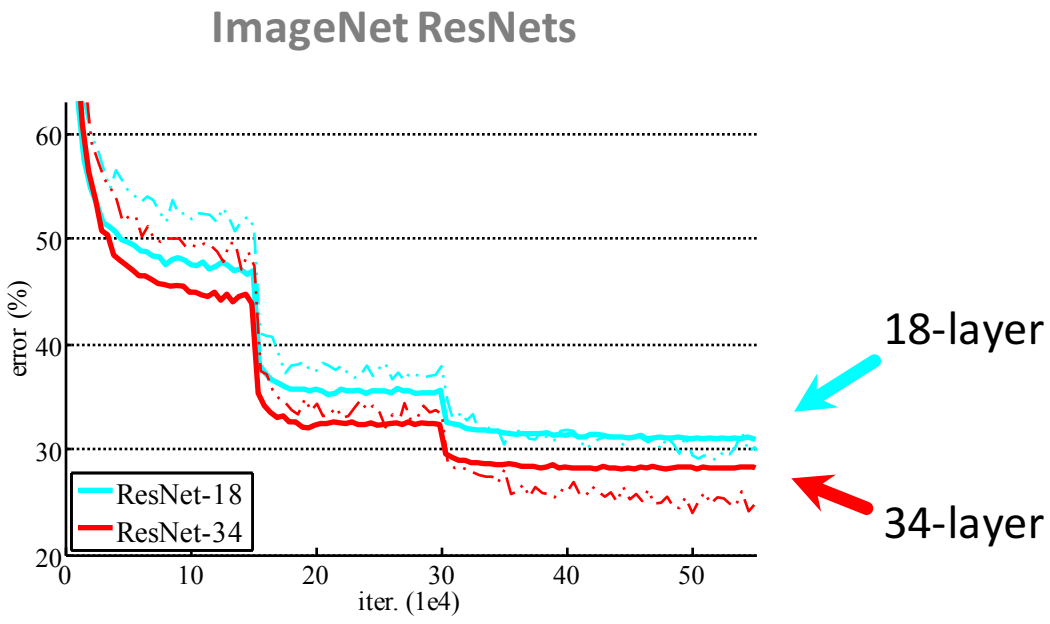


- Deep ResNets can be trained without difficulties
- Deeper ResNets have **lower training error**, and also lower test error

Kaiming He, Xiangyu Zhang, Shaoqing Ren, & Jian Sun. "Deep Residual Learning for Image Recognition". CVPR 2016.

ImageNet experiments



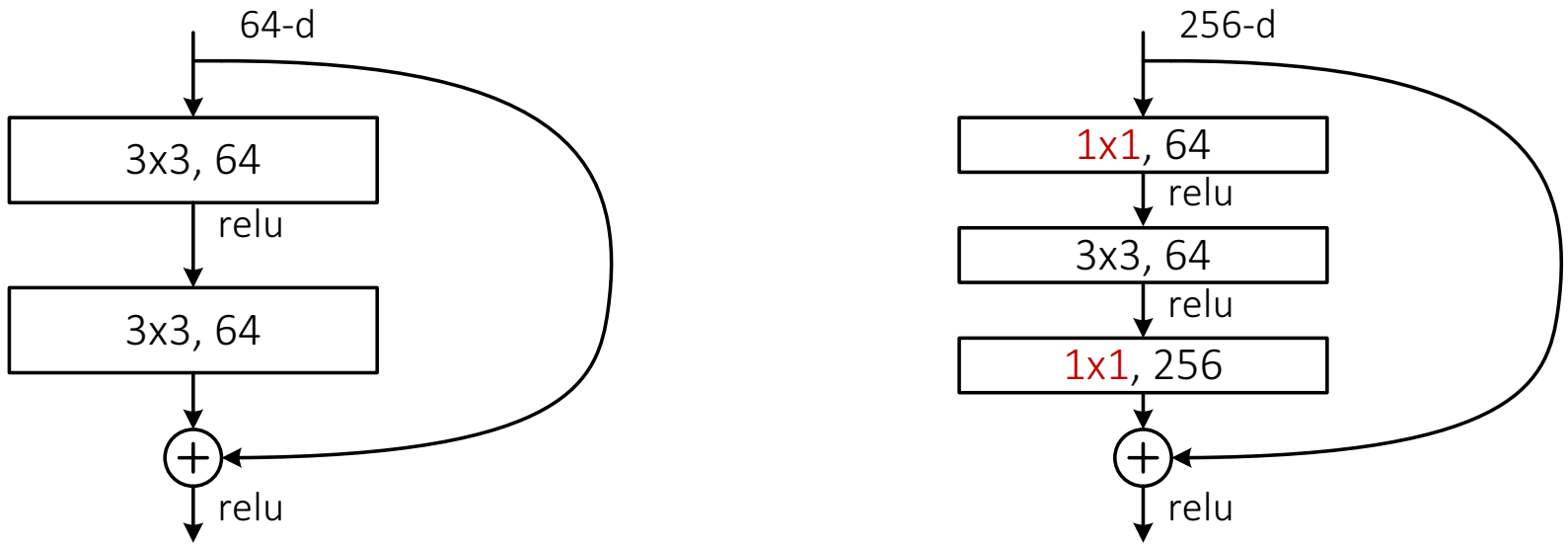


- Deep ResNets can be trained without difficulties
- Deeper ResNets have **lower training error**, and also lower test error

Kaiming He, Xiangyu Zhang, Shaoqing Ren, & Jian Sun. "Deep Residual Learning for Image Recognition". CVPR 2016.

ImageNet experiments

- A practical design of going deeper



all-3x3

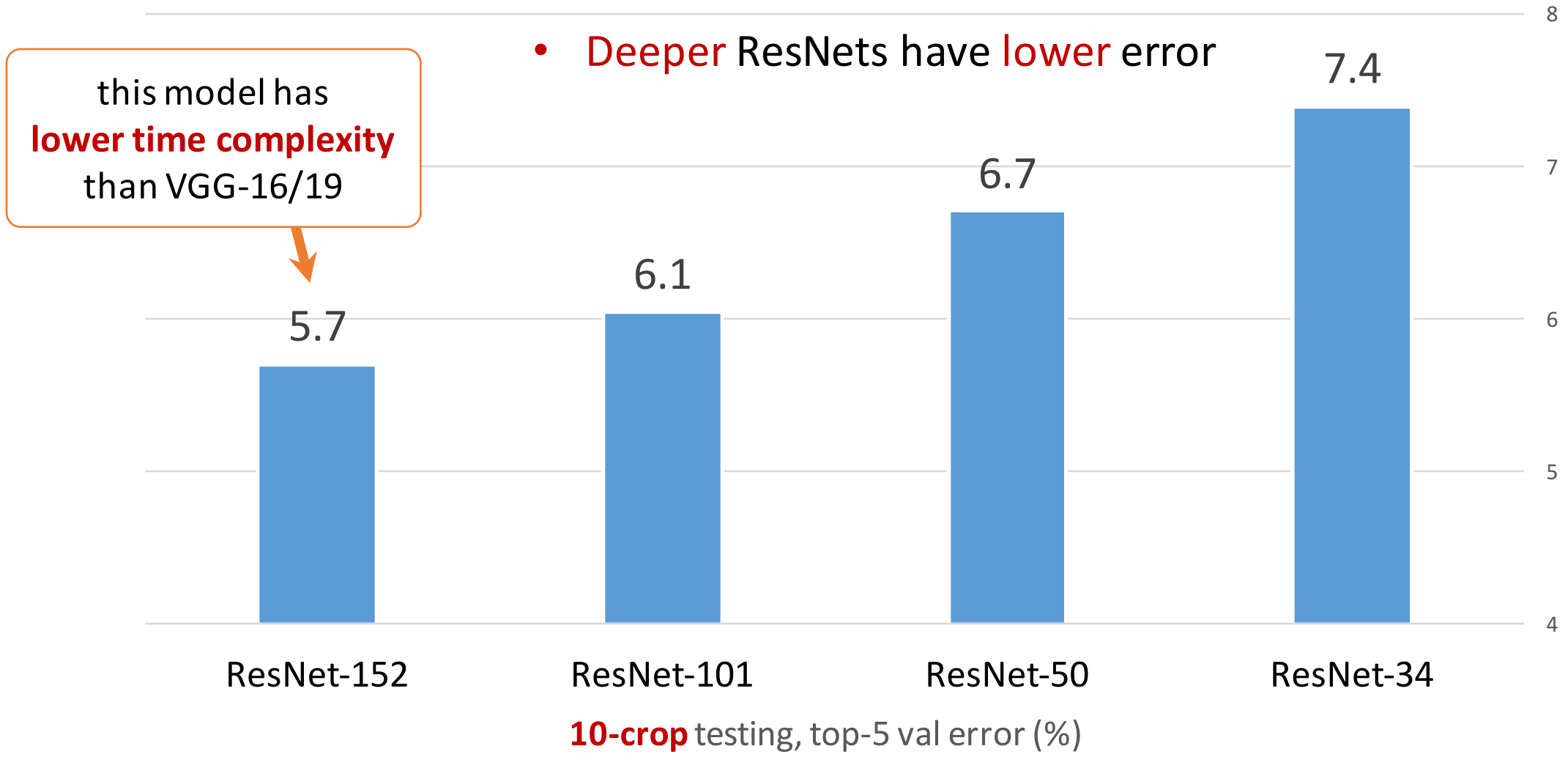
similar complexity

bottleneck

(for ResNet-50/101/152)

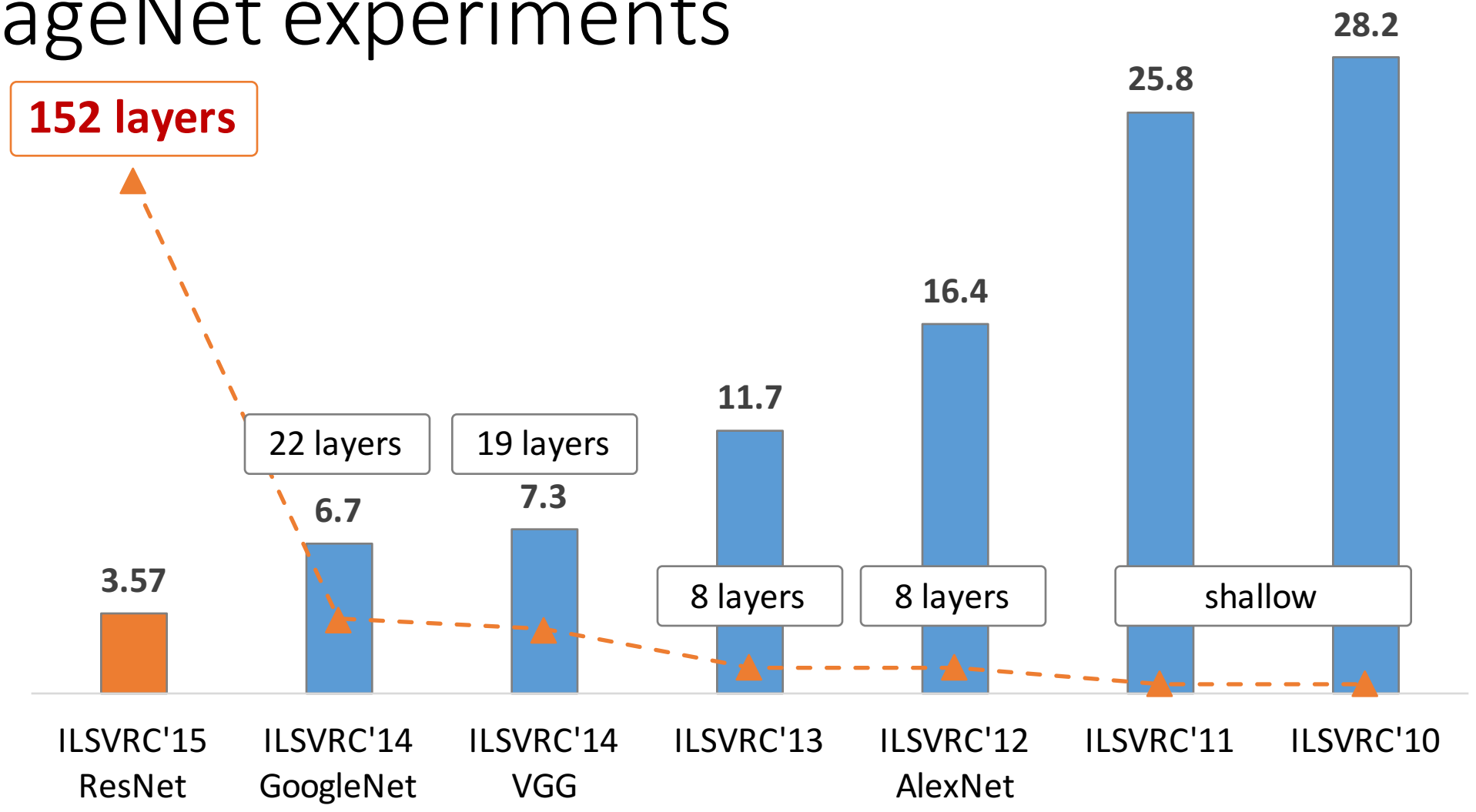
Kaiming He, Xiangyu Zhang, Shaoqing Ren, & Jian Sun. "Deep Residual Learning for Image Recognition". CVPR 2016.

ImageNet experiments



Kaiming He, Xiangyu Zhang, Shaoqing Ren, & Jian Sun. "Deep Residual Learning for Image Recognition". CVPR 2016.

ImageNet experiments

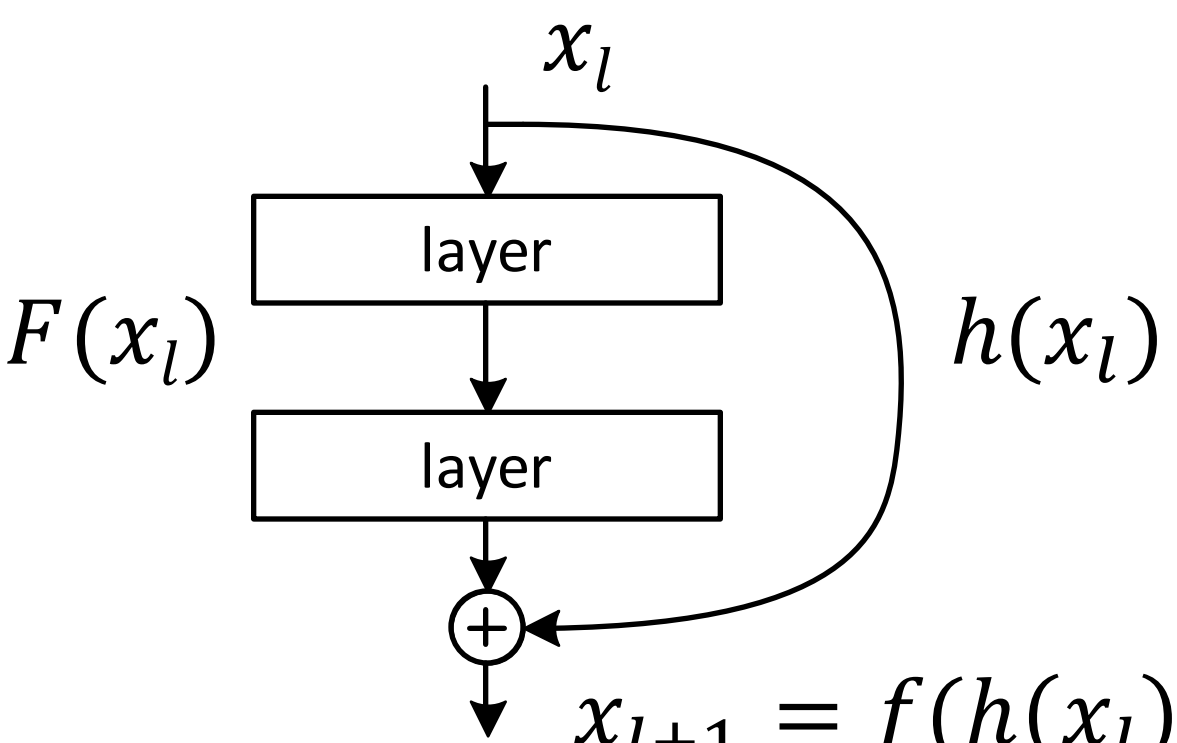


Kaiming He, Xiangyu Zhang, Shaoqing Ren, & Jian Sun. "Deep Residual Learning for Image Recognition". CVPR 2016.

On the Importance of Identity Mapping

From 100 layers to 1000 layers

On identity mappings for **optimization**

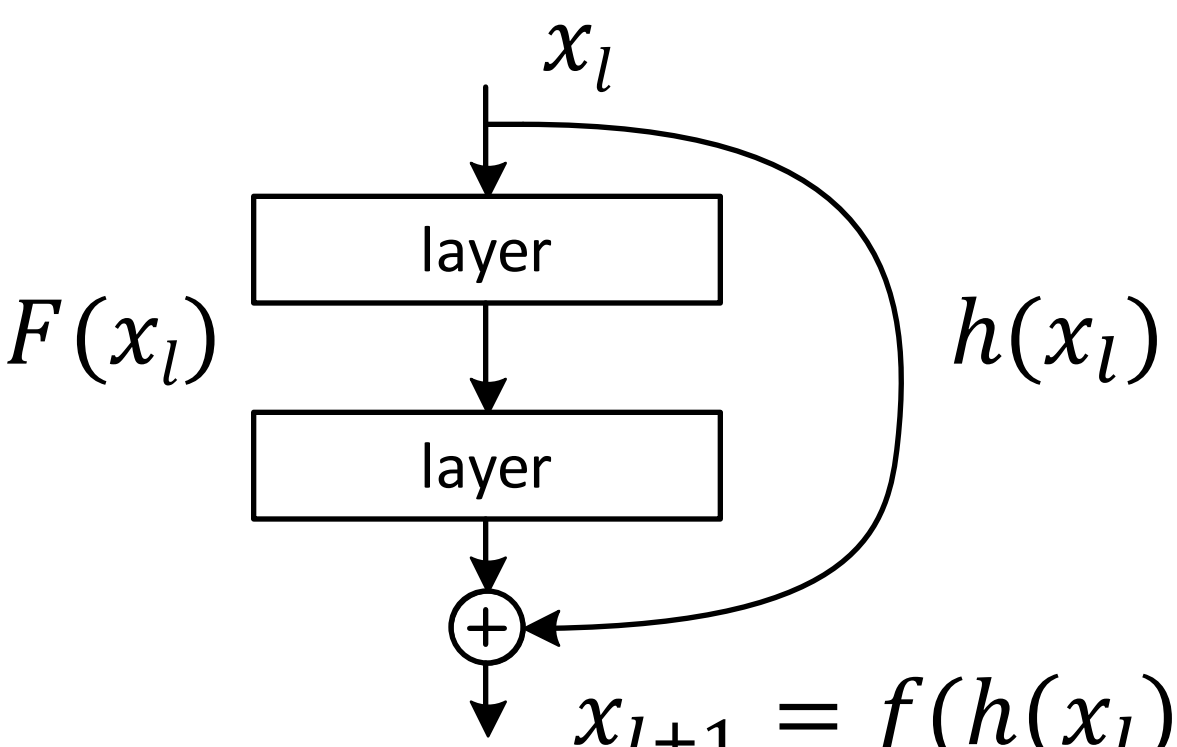


- shortcut mapping: h = identity
- after-add mapping: f = ReLU

$$x_{l+1} = f(h(x_l) + F(x_l))$$

Kaiming He, Xiangyu Zhang, Shaoqing Ren, & Jian Sun. "Identity Mappings in Deep Residual Networks". arXiv 2016.

On identity mappings for **optimization**

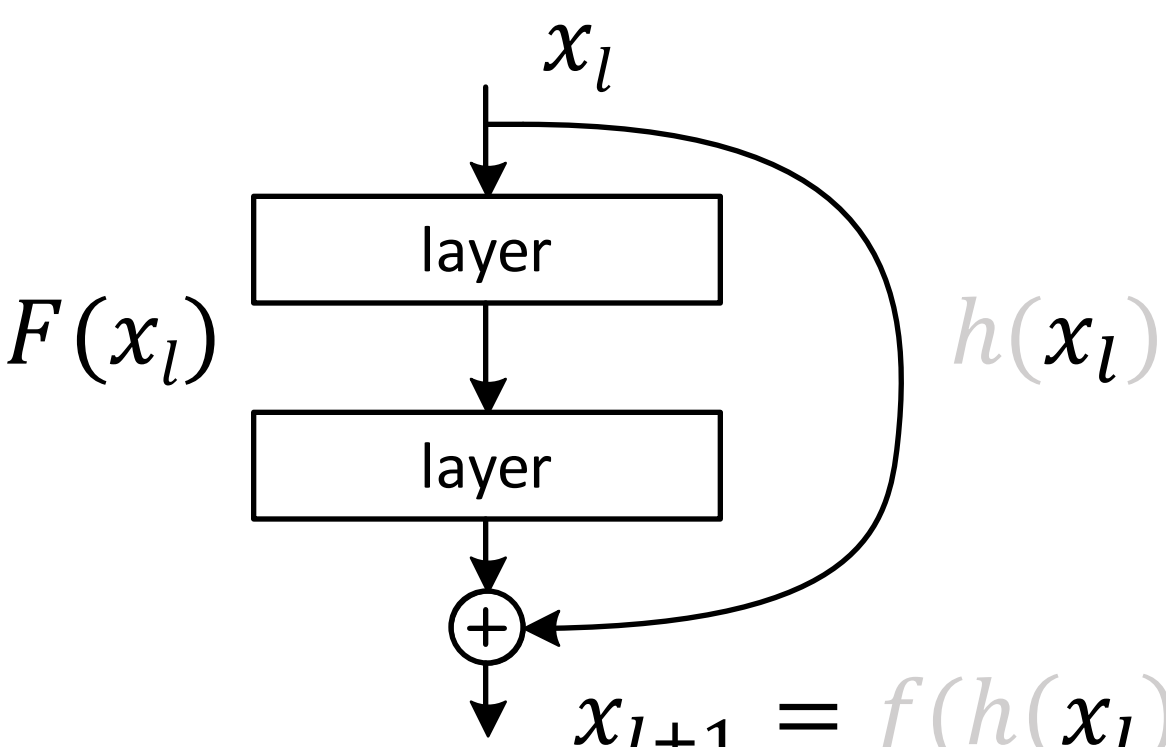


- shortcut mapping: h = identity
- ~~after-add mapping: f = ReLU~~
- What if f = identity?

$$x_{l+1} = f(h(x_l) + F(x_l))$$

Kaiming He, Xiangyu Zhang, Shaoqing Ren, & Jian Sun. "Identity Mappings in Deep Residual Networks". arXiv 2016.

On identity mappings for **optimization**



- shortcut mapping: h = identity
- ~~after-add mapping: f = ReLU~~
- What if f = identity?

$$x_{l+1} = f(h(x_l) + F(x_l))$$

Kaiming He, Xiangyu Zhang, Shaoqing Ren, & Jian Sun. “Identity Mappings in Deep Residual Networks”. arXiv 2016.

Very smooth forward propagation

$$x_{l+1} = x_l + F(x_l)$$

$$x_{l+2} = x_{l+1} + F(x_{l+1})$$

Kaiming He, Xiangyu Zhang, Shaoqing Ren, & Jian Sun. “Identity Mappings in Deep Residual Networks”. arXiv 2016.

Very smooth forward propagation

$$x_{l+1} = x_l + F(x_l)$$

$$x_{l+2} = x_{l+1} + F(x_{l+1})$$

$$x_{l+2} = x_l + F(x_l) + F(x_{l+1})$$

Kaiming He, Xiangyu Zhang, Shaoqing Ren, & Jian Sun. “Identity Mappings in Deep Residual Networks”. arXiv 2016.

Very smooth forward propagation

$$x_{l+1} = x_l + F(x_l)$$

$$x_{l+2} = x_{l+1} + F(x_{l+1})$$

$$x_{l+2} = x_l + F(x_l) + F(x_{l+1})$$

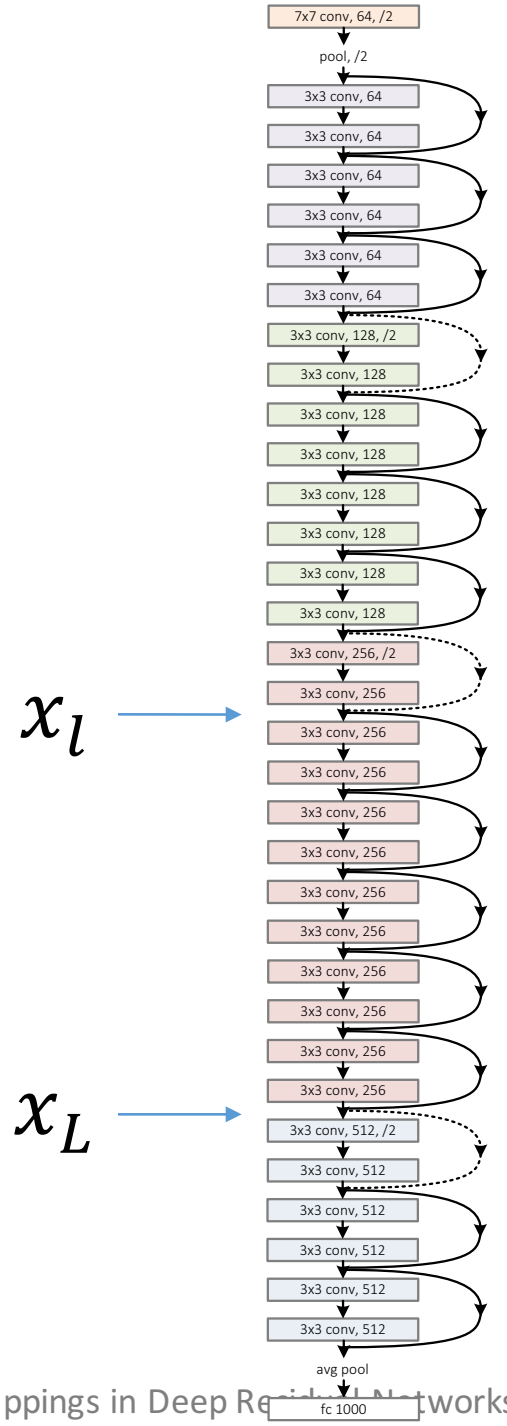
$$x_L = x_l + \sum_{i=l}^{L-1} F(x_i)$$

Kaiming He, Xiangyu Zhang, Shaoqing Ren, & Jian Sun. “Identity Mappings in Deep Residual Networks”. arXiv 2016.

Very smooth forward propagation

$$x_L = x_l + \sum_{i=l}^{L-1} F(x_i)$$

- Any x_l is directly forward-prop to any x_L, plus residual.
- Any x_L is an additive outcome.
  - in contrast to multiplicative: $x_L = \prod_{i=l}^{L-1} W_i x_l$

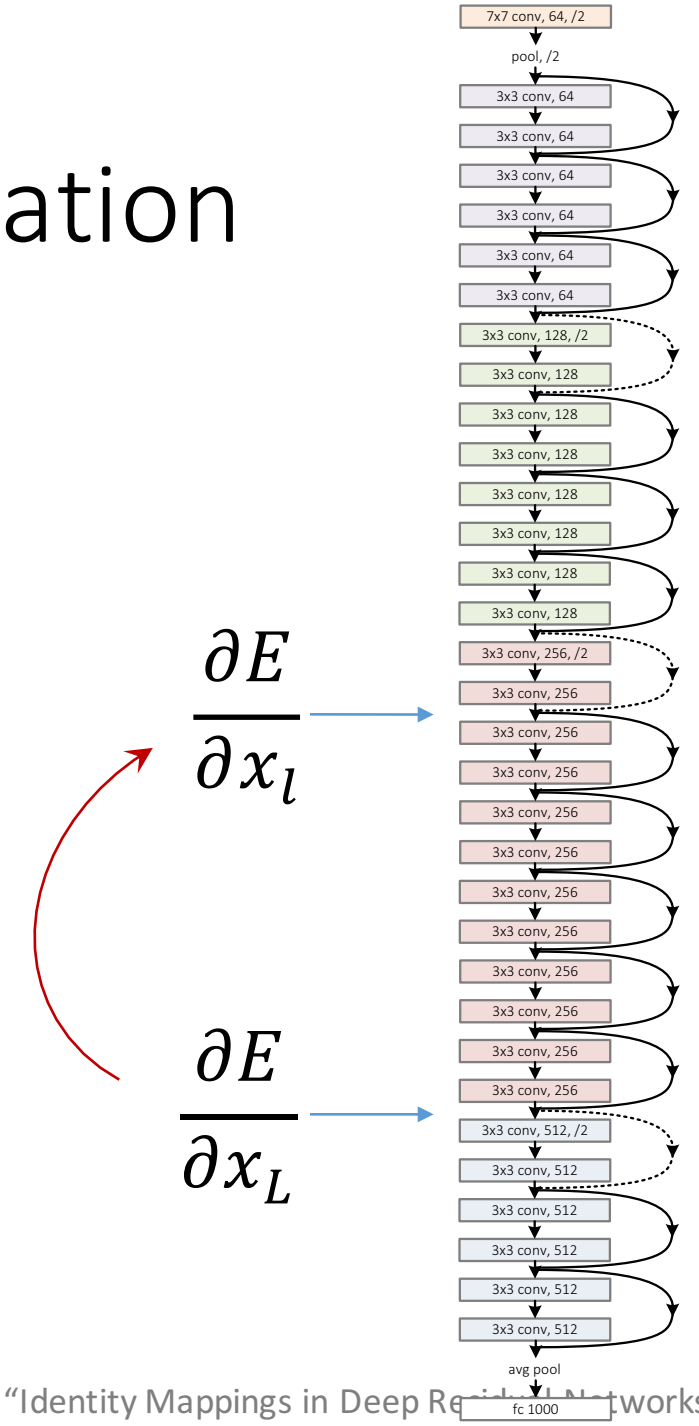


Kaiming He, Xiangyu Zhang, Shaoqing Ren, & Jian Sun. "Identity Mappings in Deep Residual Networks". arXiv 2016.

Very smooth backward propagation

$$x_L = x_l + \sum_{i=l}^{L-1} F(x_i)$$

$$\frac{\partial E}{\partial x_l} = \frac{\partial E}{\partial x_L}\frac{\partial x_L}{\partial x_l} = \frac{\partial E}{\partial x_L}\left(1 + \frac{\partial}{\partial x_l}\sum_{i=1}^{L-1} F(x_i)\right)$$

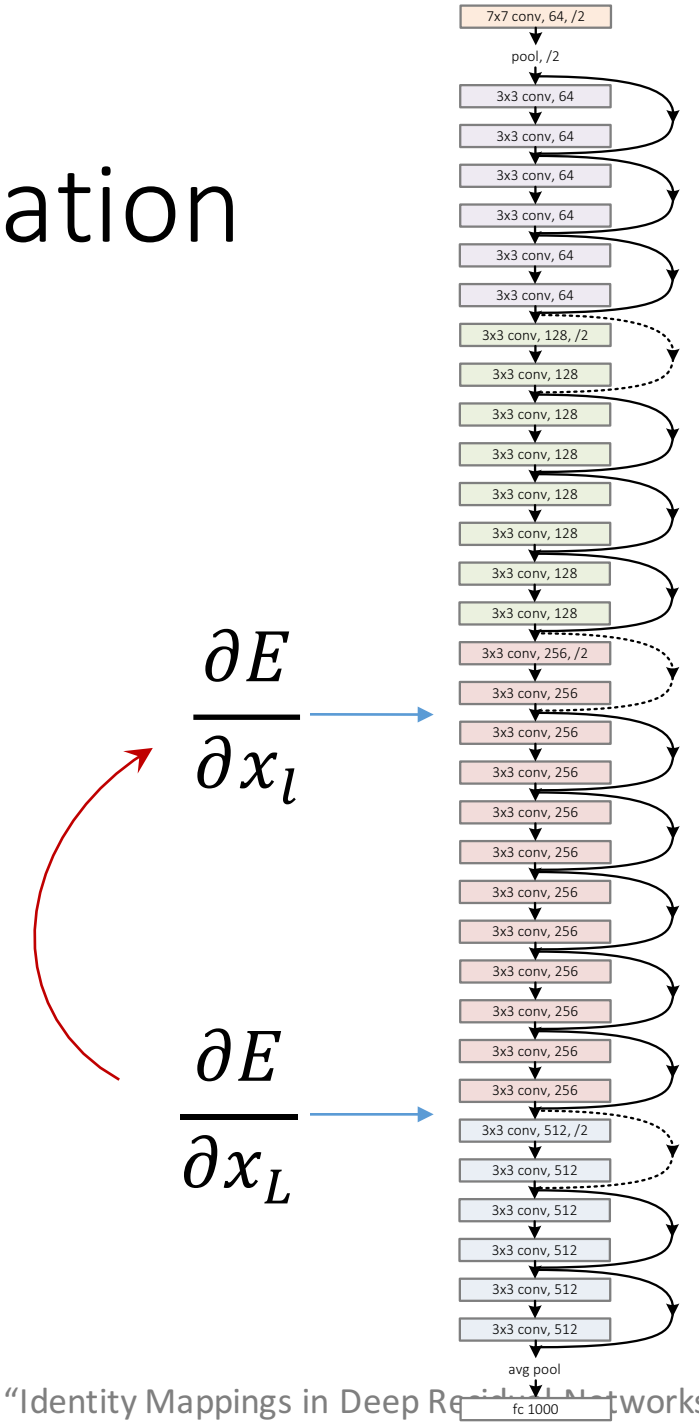


Kaiming He, Xiangyu Zhang, Shaoqing Ren, & Jian Sun. "Identity Mappings in Deep Residual Networks". arXiv 2016.

Very smooth backward propagation

$$\frac{\partial E}{\partial x_l} = \frac{\partial E}{\partial x_L}\left(1 + \frac{\partial}{\partial x_l}\sum_{i=1}^{L-1} F(x_i)\right)$$

- Any $\frac{\partial E}{\partial x_L}$ is directly back-prop to any $\frac{\partial E}{\partial x_l}$, plus residual.
- Any $\frac{\partial E}{\partial x_l}$ is additive; unlikely to vanish
  - in contrast to multiplicative: $\frac{\partial E}{\partial x_l} = \prod_{i=l}^{L-1} W_i \frac{\partial E}{\partial x_L}$



Kaiming He, Xiangyu Zhang, Shaoqing Ren, & Jian Sun. "Identity Mappings in Deep Residual Networks". arXiv 2016.

Residual for every layer

forward: $x_L = x_l + \sum_{i=l}^{L-1} F(x_i)$

Enabled by:

- shortcut mapping: h = identity
- after-add mapping: f = identity

backward: $\frac{\partial E}{\partial x_l} = \frac{\partial E}{\partial x_L}\left(1 + \frac{\partial}{\partial x_l}\sum_{i=1}^{L-1} F(x_i)\right)$

Kaiming He, Xiangyu Zhang, Shaoqing Ren, & Jian Sun. “Identity Mappings in Deep Residual Networks”. arXiv 2016.

Experiments

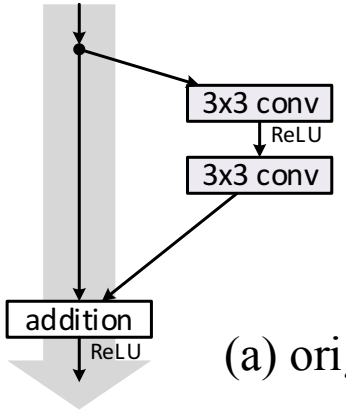
- Set 1: what if shortcut mapping $h \neq$ identity
- Set 2: what if after-add mapping f is identity
- Experiments on ResNets with more than 100 layers
  - deeper models suffer more from optimization difficulty

Experiment Set 1:
what if shortcut mapping h ≠ identity?

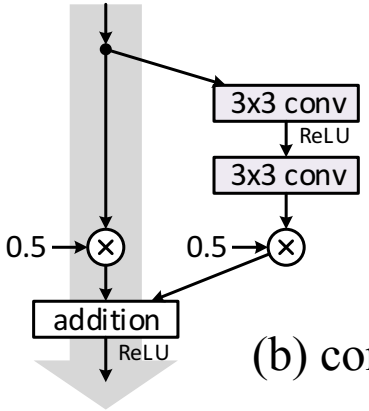
* ResNet-110 on CIFAR-10

$h(x) = x$

error: 6.6%



(a) original



(b) constant scaling

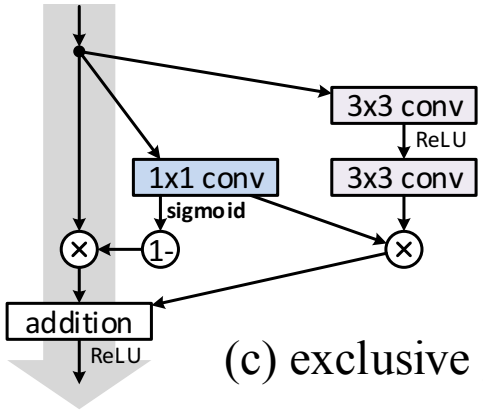
$h(x) = 0.5x$

error: 12.4%

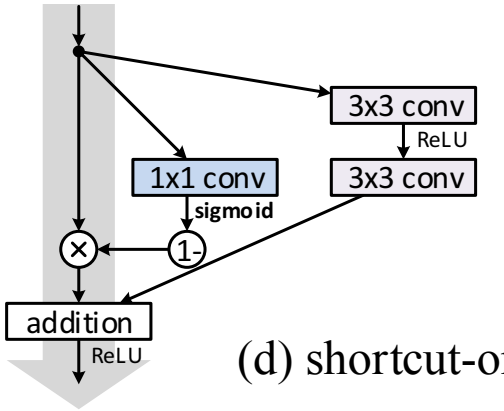
$h(x) = \text{gate} \cdot x$

error: 8.7%

*similar to "Highway Network"



(c) exclusive gating



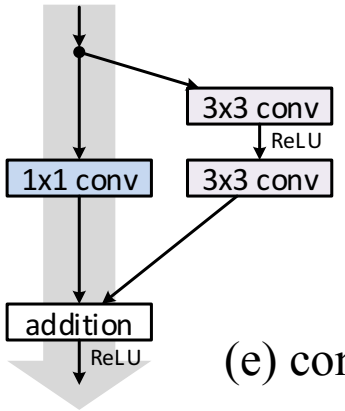
(d) shortcut-only gating

$h(x) = \text{gate} \cdot x$

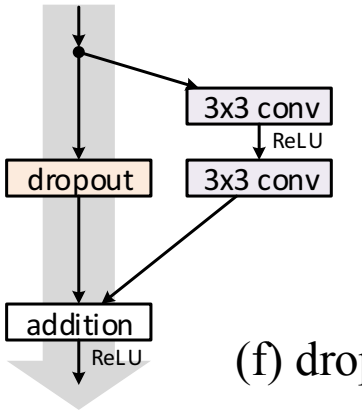
error: 12.9%

$h(x) = \text{conv}(x)$

error: 12.2%



(e) conv shortcut



(f) dropout shortcut

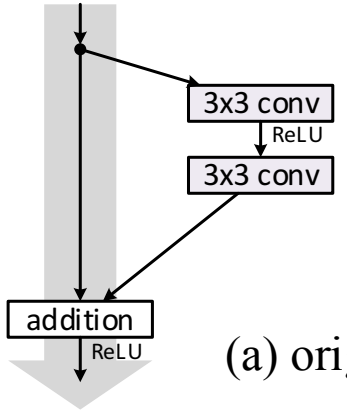
$h(x) = \text{dropout}(x)$

error: > 20%

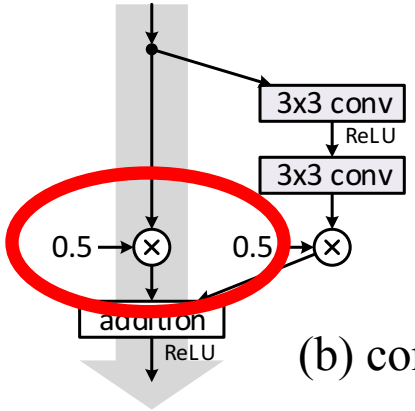
Kaiming He, Xiangyu Zhang, Shaoqing Ren, & Jian Sun. "Identity Mappings in Deep Residual Networks". arXiv 2016.

* ResNet-110 on CIFAR-10

$h(x) = x$
error: 6.6%



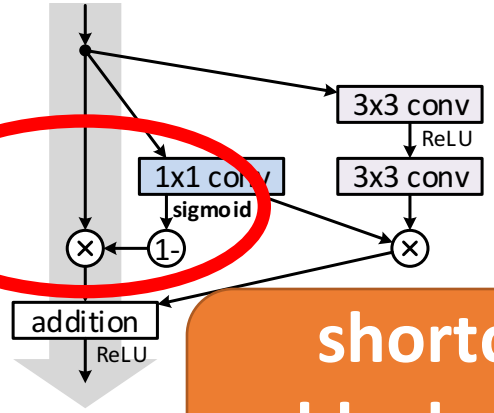
(a) original

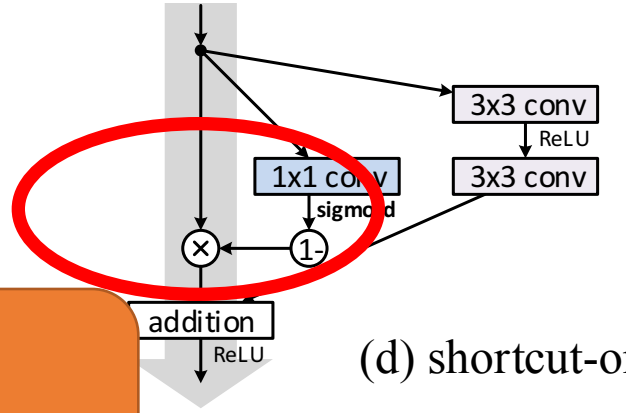


(b) constant scaling

$h(x) = 0.5x$
error: 12.4%

$h(x) = \text{gate} \cdot x$
error: 8.7%



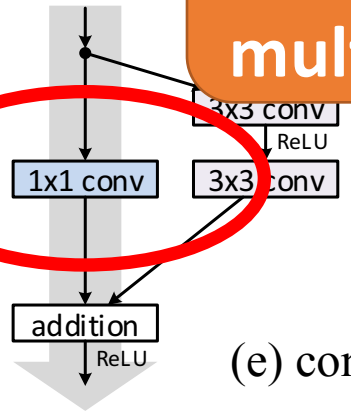


(d) shortcut-only gating

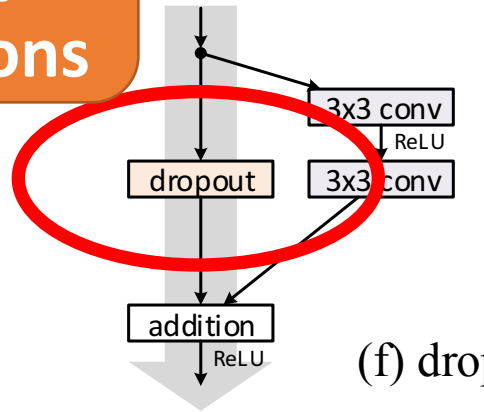
$h(x) = \text{gate} \cdot x$
error: 12.9%

shortcuts blocked by multiplications

$h(x) = \text{conv}(x)$
error: 12.2%



(e) conv shortcut



(f) dropout shortcut

$h(x) = \text{dropout}(x)$
error: > 20%

Kaiming He, Xiangyu Zhang, Shaoqing Ren, & Jian Sun. "Identity Mappings in Deep Residual Networks". arXiv 2016.

If h is multiplicative, e.g. $h(x) = \lambda x$

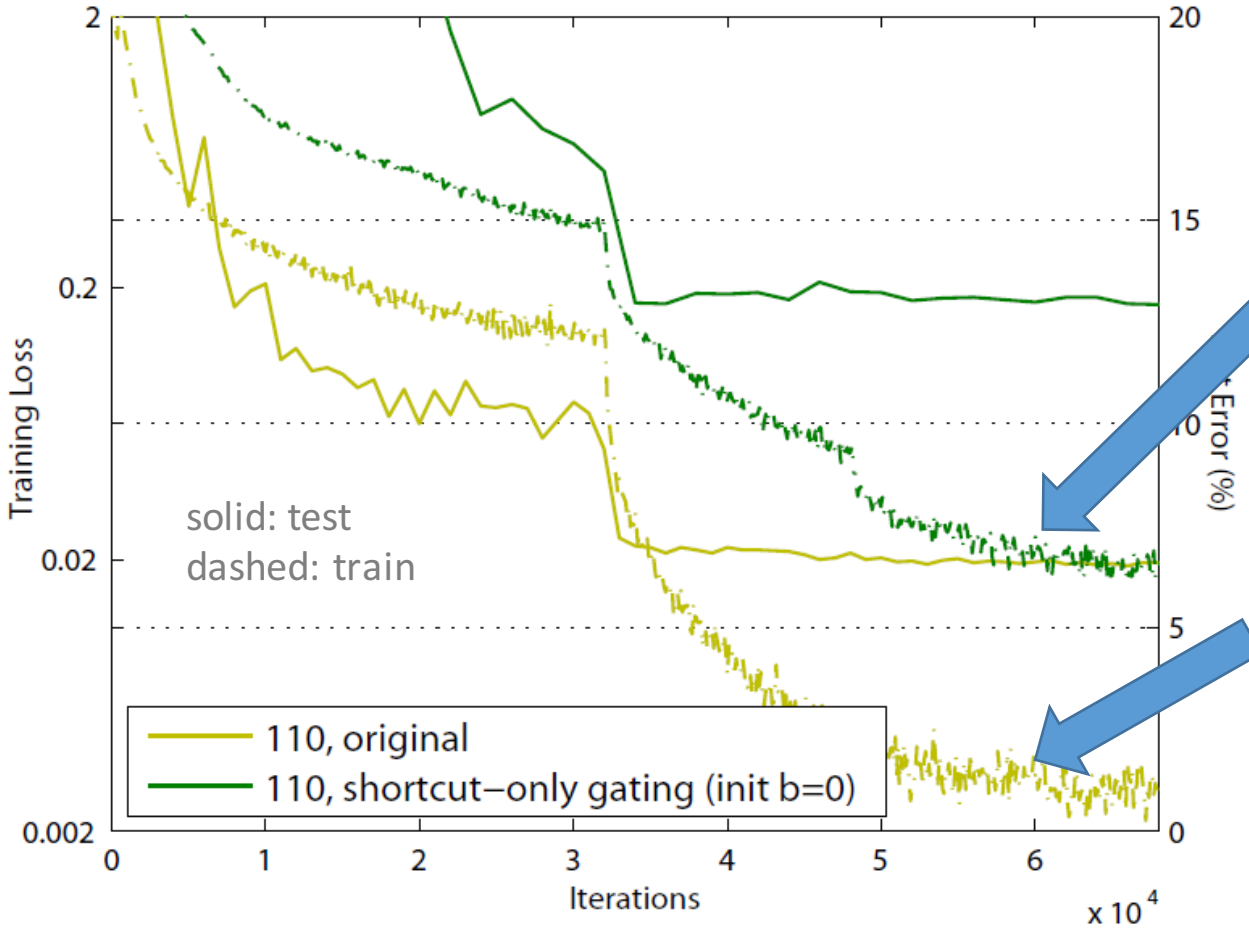
forward: $x_L = \lambda^{L-l} x_l + \sum_{i=l}^{L-1} \hat{F}(x_i)$

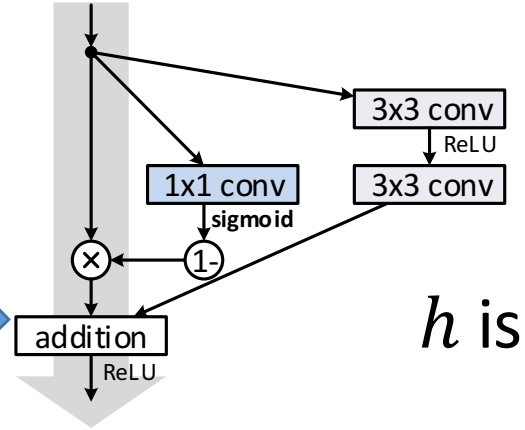
- if h is multiplicative, shortcuts are blocked
- direct propagation is decayed

backward: $\frac{\partial E}{\partial x_l} = \frac{\partial E}{\partial x_L}\left(\lambda^{L-l} + \frac{\partial}{\partial x_l} \sum_{i=1}^{L-1} \hat{F}(x_i)\right)$

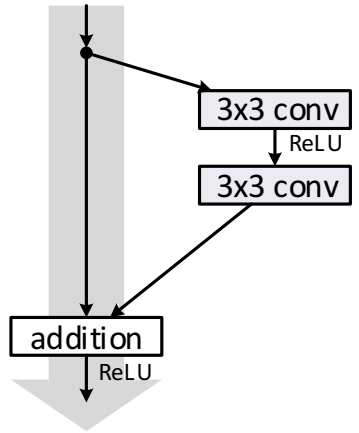
*assuming f = identity

Kaiming He, Xiangyu Zhang, Shaoqing Ren, & Jian Sun. “Identity Mappings in Deep Residual Networks”. arXiv 2016.





h is gating

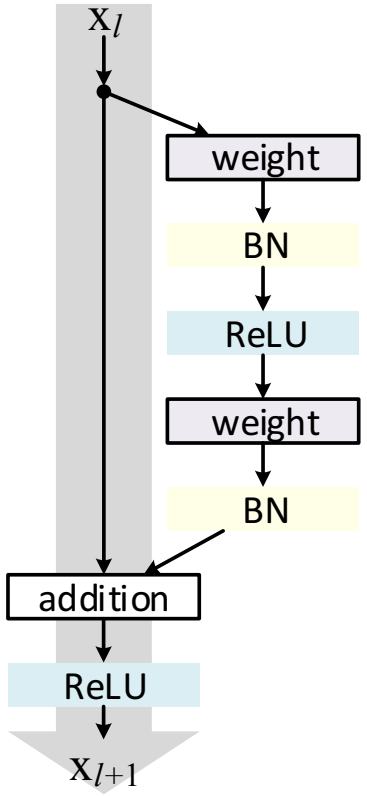


h is identity

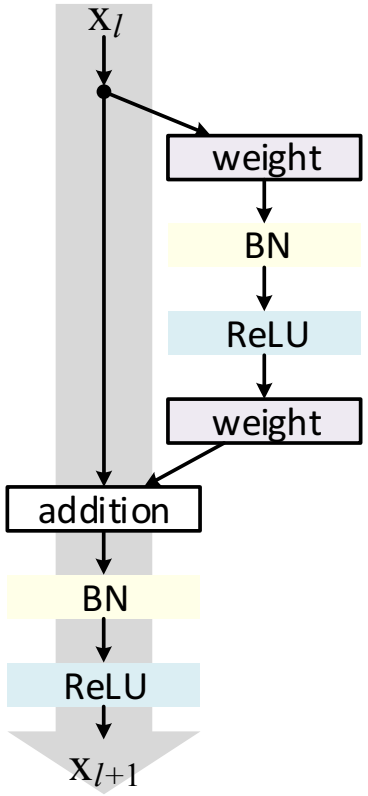
- gating should have better representation ability (identity is a special case), but
- optimization difficulty dominates results

Kaiming He, Xiangyu Zhang, Shaoqing Ren, & Jian Sun. “Identity Mappings in Deep Residual Networks”. arXiv 2016.

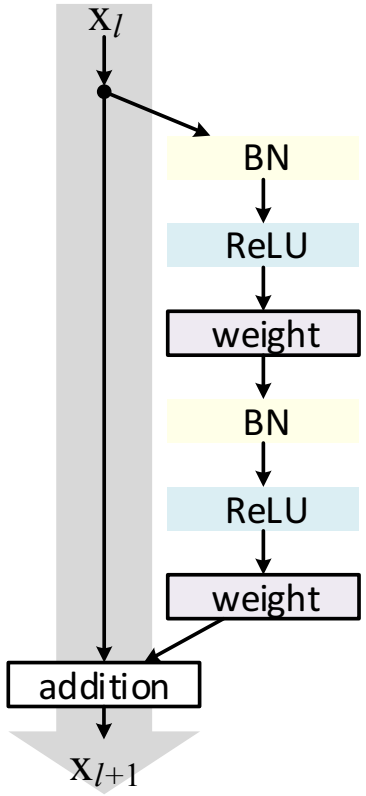
Experiment Set 2:
what if after-add mapping f is identity



f is ReLU
(original ResNet)

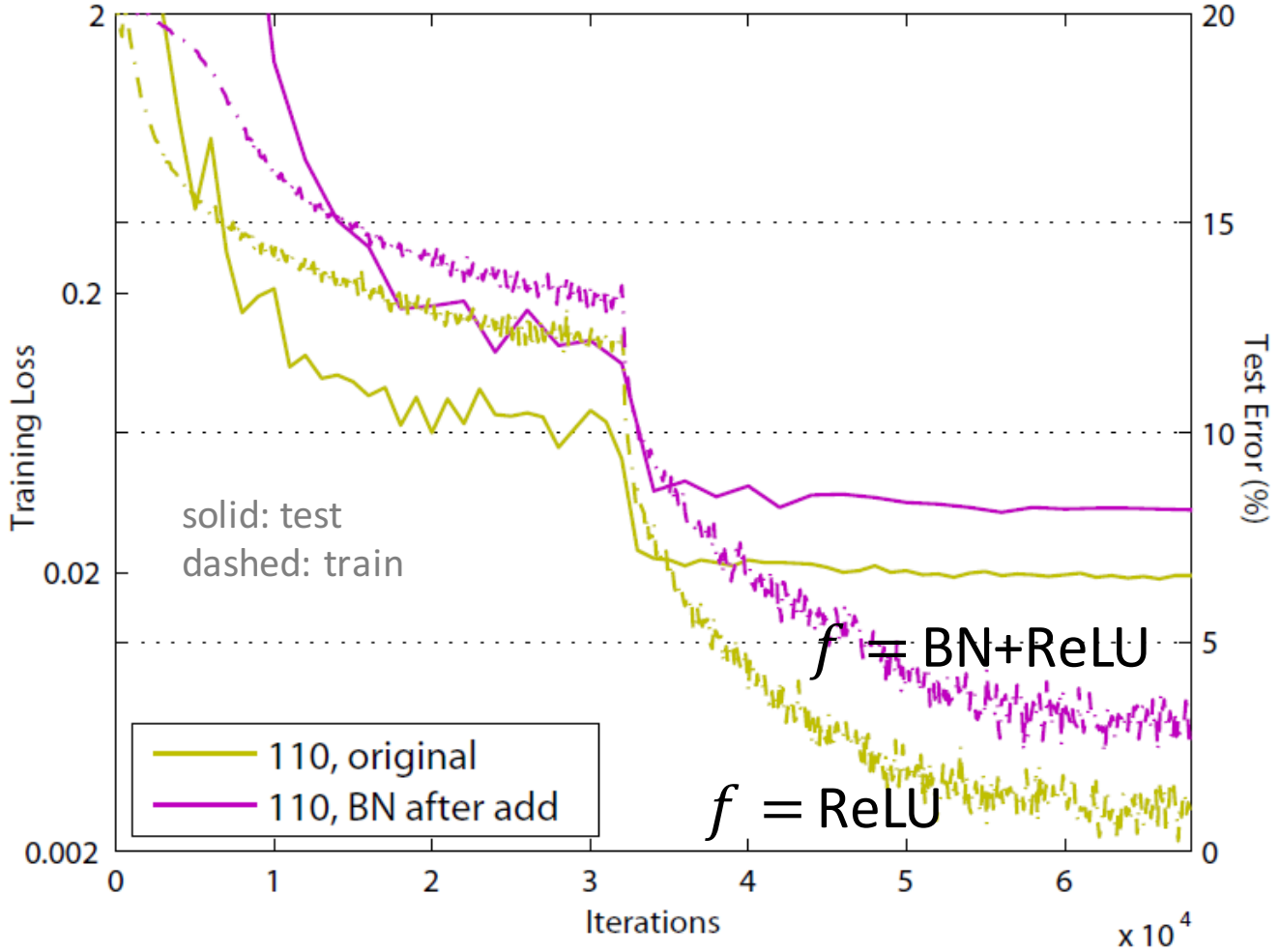


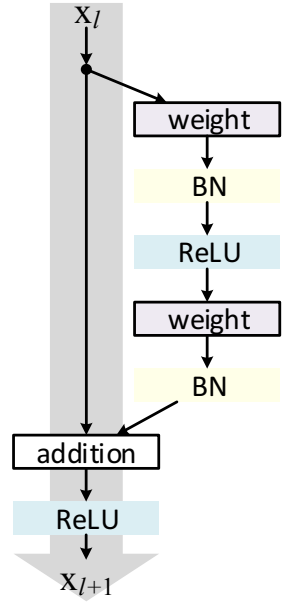
f is BN+ReLU

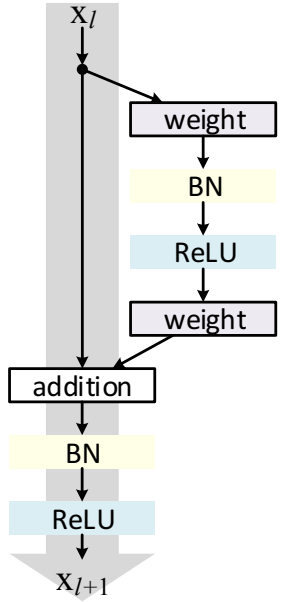


f is identity
(**pre-activation** ResNet)

Kaiming He, Xiangyu Zhang, Shaoqing Ren, & Jian Sun. "Identity Mappings in Deep Residual Networks". arXiv 2016.





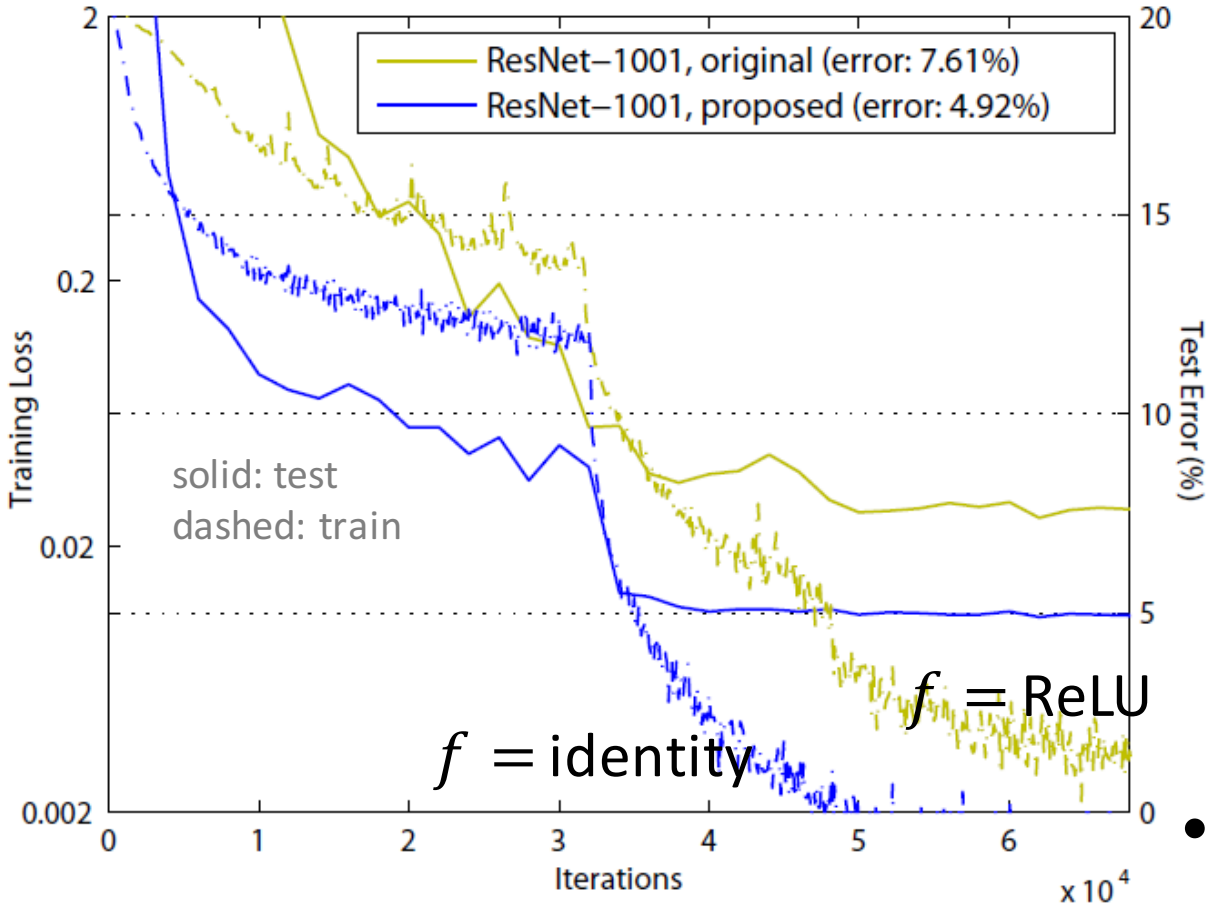


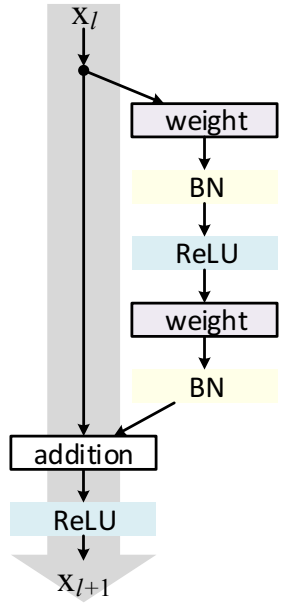
f = ReLU f = BN+ReLU

- BN could block prop
- Keep the shortest pass as smooth as possible

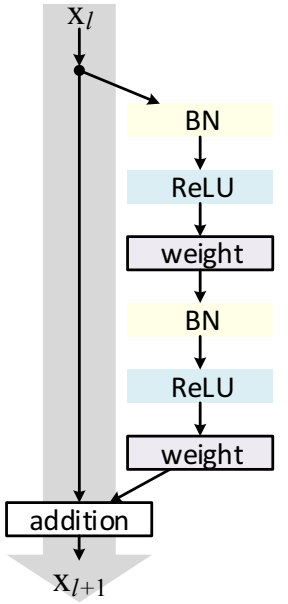
Kaiming He, Xiangyu Zhang, Shaoqing Ren, & Jian Sun. "Identity Mappings in Deep Residual Networks". arXiv 2016.

1001-layer ResNets on CIFAR-10





f = ReLU



f = identity

- ReLU could block prop when there are 1000 layers
- pre-activation design eases optimization (and improves generalization; see paper)

Kaiming He, Xiangyu Zhang, Shaoqing Ren, & Jian Sun. "Identity Mappings in Deep Residual Networks". arXiv 2016.

Comparisons on CIFAR-10/100

CIFAR-10

method	error (%)
NIN	8.81
DSN	8.22
FitNet	8.39
Highway	7.72
ResNet-110 (1.7M)	6.61
ResNet-1202 (19.4M)	7.93
ResNet-164, pre-activation (1.7M)	5.46
ResNet-1001, pre-activation (10.2M)	**4.92** (4.89±0.14)

CIFAR-100

method	error (%)
NIN	35.68
DSN	34.57
FitNet	35.04
Highway	32.39
ResNet-164 (1.7M)	25.16
ResNet-1001 (10.2M)	27.82
ResNet-164, pre-activation (1.7M)	24.33
ResNet-1001, pre-activation (10.2M)	**22.71** (22.68±0.22)

*all based on moderate augmentation

Kaiming He, Xiangyu Zhang, Shaoqing Ren, & Jian Sun. "Identity Mappings in Deep Residual Networks". arXiv 2016.

ImageNet Experiments

ImageNet single-crop (320x320) val error

method	data augmentation	top-1 error (%)	top-5 error (%)
ResNet-152, original	scale	21.3	5.5
ResNet-152, pre-activation	scale	21.1	5.5
ResNet-200, original	scale	21.8	6.0
ResNet-200, pre-activation	scale	**20.7**	**5.3**
ResNet-200, pre-activation	scale + aspect ratio	**20.1***	**4.8***

*independently reproduced by:

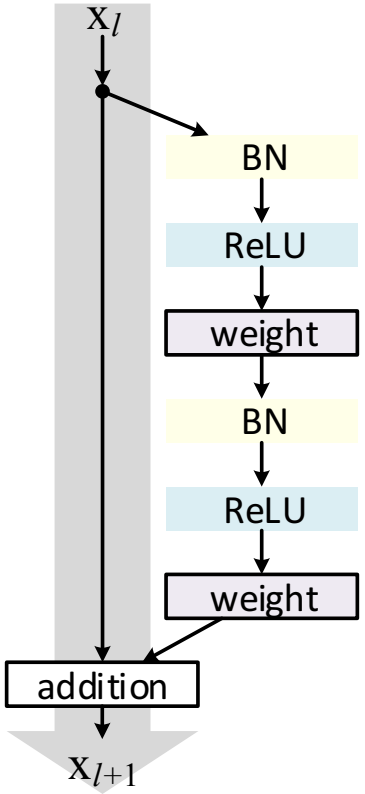
https://github.com/facebook/fb.resnet.torch/tree/master/pretrained#notes

training code and models available.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, & Jian Sun. "Identity Mappings in Deep Residual Networks". arXiv 2016.

Summary of observations

- Keep the shortest path as smooth as possible
  - by making h and f identity
  - forward/backward signals directly flow through this path
- Features of any layers are additive outcomes
- 1000-layer ResNets can be easily trained and have better accuracy



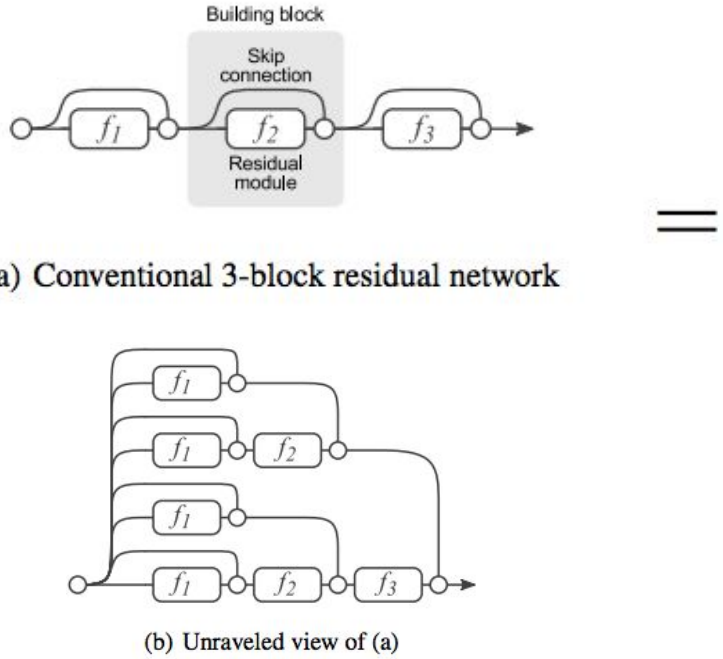
Kaiming He, Xiangyu Zhang, Shaoqing Ren, & Jian Sun. "Identity Mappings in Deep Residual Networks". arXiv 2016.

THE ROADMAP

- INTRODUCTION
- DEEP RESIDUAL NETWORKS
- **WHY DO DEEP RESIDUAL NETWORKS WORK?**
- SURVEY OF ADVANCED CNN ARCHITECTURES
- CONCLUSION

Why do ResNets work? Some ideas:

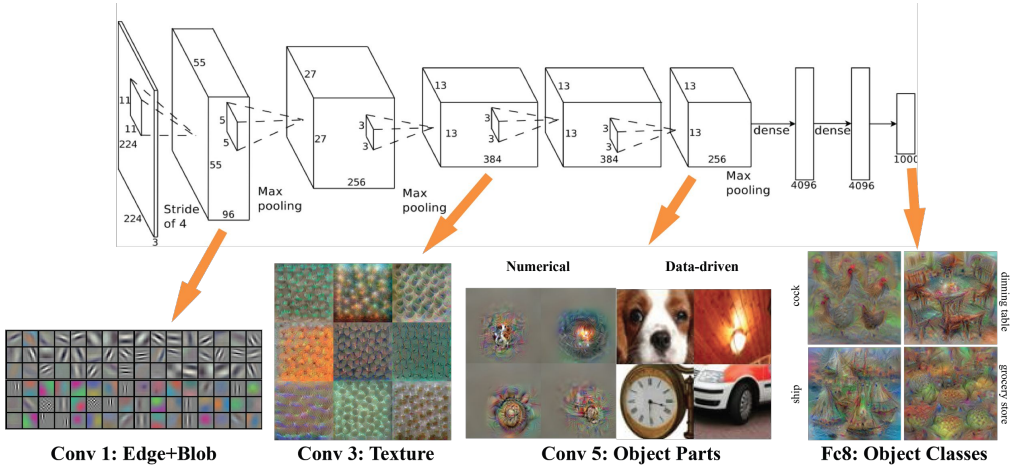
- **They can be seen as implicitly ensembling shallower networks**
- They are able to learn unrolled iterative refinements
- Can model recurrent computations necessary for recognition



Andreas Veit, Michael Wilber, Serge Belongie, Residual Networks Behave Like Ensembles of Relatively Shallow Networks, arxiv 2016

Why do ResNets work? Some ideas:

- They can be seen as implicitly ensembling shallower networks
- **They are able to learn unrolled iterative estimation**
- Can model recurrent computations necessary for recognition

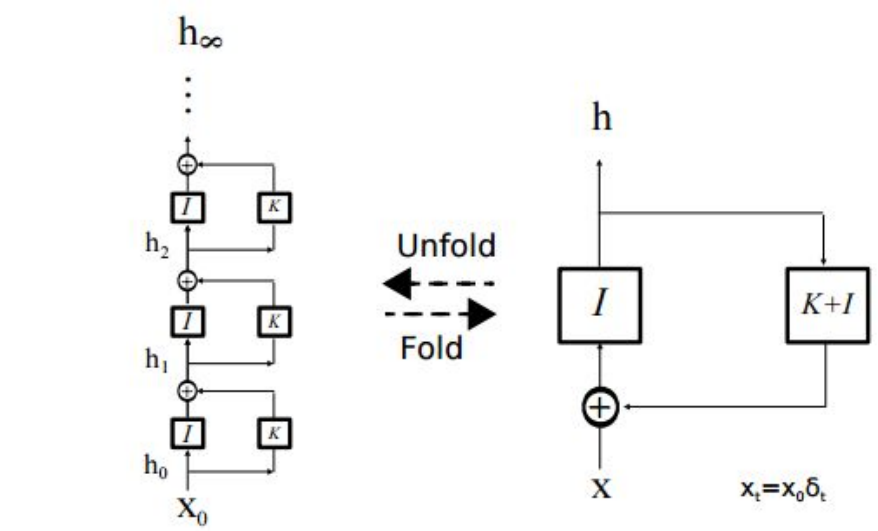


Challenges the “representation view”

Image source: http://vision03.csail.mit.edu/cnn_art/

Why do ResNets work? Some ideas:

- They can be seen as implicitly ensembling shallower networks
- They are able to learn unrolled iterative estimation
- **Can model recurrent computations useful for recognition**



Qianli Liao, Tomaso Poggio,Bridging the Gaps Between Residual Learning, Recurrent Neural Networks and Visual Cortex,

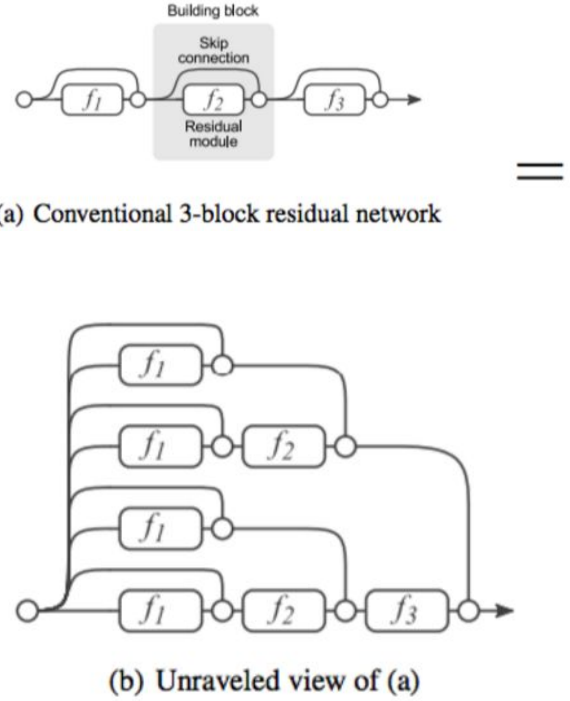
ResNets as Ensembles

- **Can think of ResNets as ensembling subsets of residual modules**
- With L residual modules there are 2^L possible subsets of modules
- If one of modules is removed, there are still 2^{L-1} possible subsets of modules

- For each residual module, we can choose whether we include it
- There are 2 options per module (include/exclude) for L modules
- Total of 2^L modules in the implicit ensemble

ResNets as Ensembles

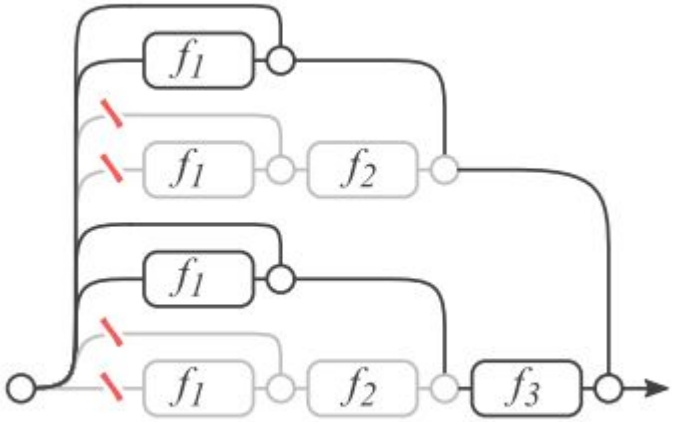
- Can think of ResNets as ensembling subsets of residual modules
- **With L residual modules there are 2^L possible subsets of modules**
- If one of modules is removed, there are still 2^{L-1} possible subsets of modules



Andreas Veit, Michael Wilber, Serge Belongie, Residual Networks Behave Like Ensembles of Relatively Shallow Networks, arxiv 2016

ResNets as Ensembles

- Can think of ResNets as ensembling subsets of residual modules
- With L residual modules there are 2^L possible subsets of modules
- **If one of modules is removed, there are still 2^{L-1} possible subsets of modules**

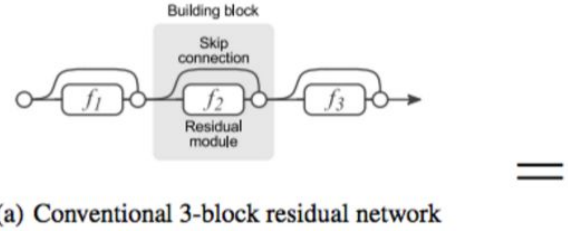


(a) Deleting f_2 from unraveled view

Andreas Veit, Michael Wilber, Serge Belongie, Residual Networks Behave Like Ensembles of Relatively Shallow Networks, arxiv 2016

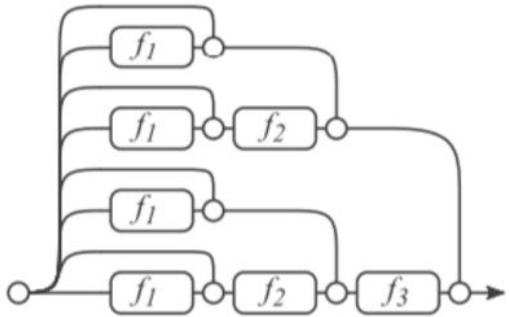
ResNets as Ensembles

- **Wanted to test this explanation**
- Tried dropping layers
- Tried reordering layers
- Found effective paths during training are relatively shallow



(a) Conventional 3-block residual network

=



(b) Unraveled view of (a)

Andreas Veit, Michael Wilber, Serge Belongie, Residual Networks Behave Like Ensembles of Relatively Shallow Networks, arxiv 2016

ResNets as Ensembles

- Wanted to test this explanation
- **Tried dropping layers**
- Tried reordering layers
- Found effective paths during training are relatively shallow

Dropping layers on VGG-Net is disastorous...

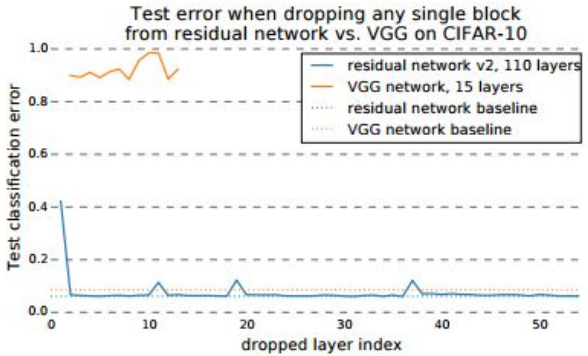


Figure 3: Deleting individual layers from VGG and a residual network on CIFAR-10. VGG performance drops to random chance when any one of its layers is deleted, but deleting individual modules from residual networks has a minimal impact on performance. Removing downsampling modules has a slightly higher impact.

Andreas Veit, Michael Wilber, Serge Belongie, Residual Networks Behave Like Ensembles of Relatively Shallow Networks, arxiv 2016

ResNets as Ensembles

- Wanted to test this explanation
- **Tried dropping layers**
- Tried reordering layers
- Found effective paths during training are relatively shallow

Dropping layers on ResNet is no big deal

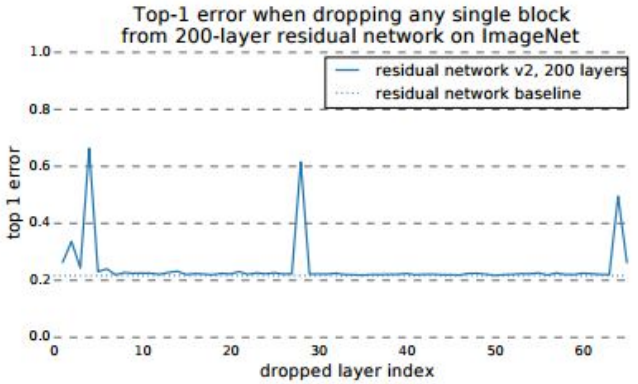
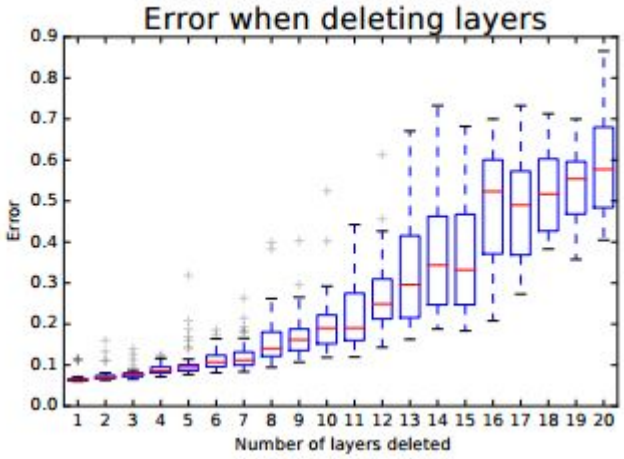


Figure 4: Results when dropping individual blocks from residual networks trained on ImageNet are similar to CIFAR results. However, downsampling layers tend to have more impact on ImageNet.

Andreas Veit, Michael Wilber, Serge Belongie, Residual Networks Behave Like Ensembles of Relatively Shallow Networks, arxiv 2016

ResNets as Ensembles

- Wanted to test this explanation
- **Tried dropping layers**
- Tried reordering layers
- Found effective paths during training are relatively shallow

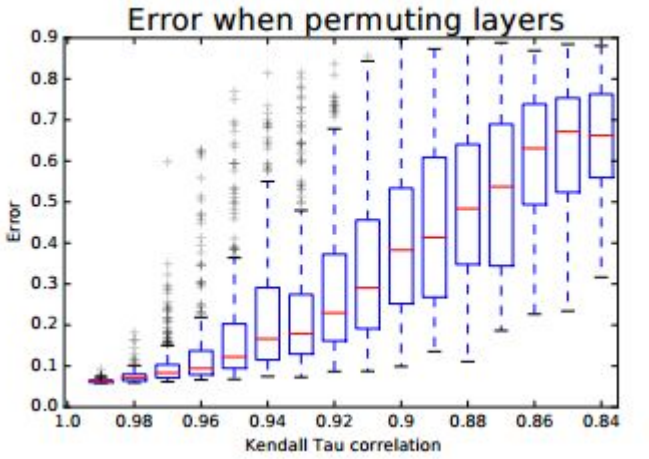


Performance degrades smoothly as layers are removed

Andreas Veit, Michael Wilber, Serge Belongie, Residual Networks Behave Like Ensembles of Relatively Shallow Networks, arxiv 2016

ResNets as Ensembles

- Wanted to test this explanation
- Tried dropping layers
- **Tried reordering layers**
- Found effective paths during training are relatively shallow

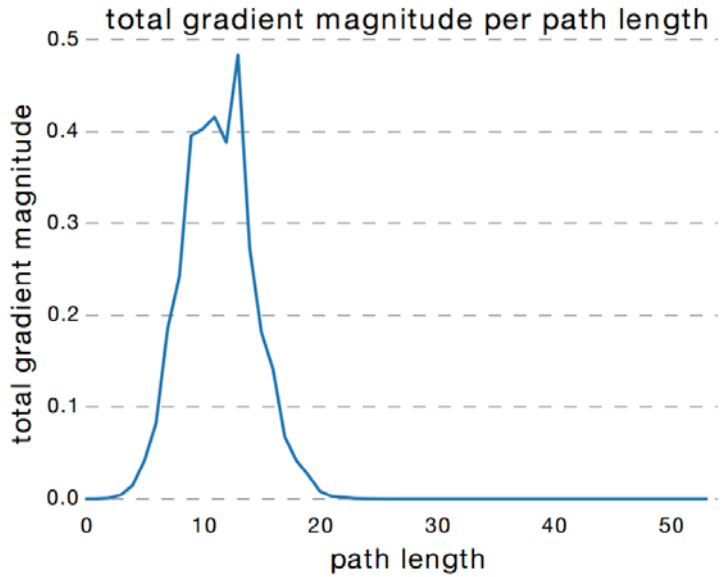


The Kendall Tau correlation coefficient measures the degree of reordering

Andreas Veit, Michael Wilber, Serge Belongie, Residual Networks Behave Like Ensembles of Relatively Shallow Networks, arxiv 2016

ResNets as Ensembles

- Wanted to test this explanation
- Tried dropping layers
- Tried reordering layers
- **Found effective paths during training are relatively shallow**



The gradient updates during training come from paths between 5 and 17 modules long.

Summary

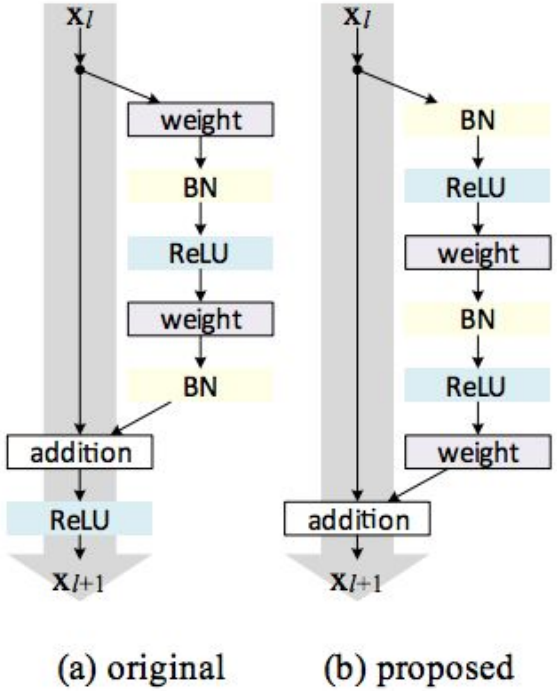
- ResNets seem to work because they facilitate the training of deeper networks
- Are suprisingly robust to layers being dropped or reordered
- Seem to be function approximations using iterative refinement

THE ROADMAP

- INTRODUCTION
- DEEP RESIDUAL NETWORKS
- WHY DO DEEP RESIDUAL NETWORKS WORK?
- **SURVEY OF ADVANCED CNN ARCHITECTURES**
- CONCLUSION

What are the current trends?

- **Some make minor modifications to ResNets**
- Biggest trend is to split of into several branches, and merge through summation
- A couple architectures go crazy with branch & merge, without explicit identity connections



Identity Mappings in Deep Residual Networks
Kaiming He, Xiangyu Zhang, Shaoqing Ren, Jian Sun

What are the current trends?

- Some make minor modifications to ResNets
- **Biggest trend is to split of into several branches, and merge through summation**
- A couple architectures go crazy with branch & merge, without explicit identity connections

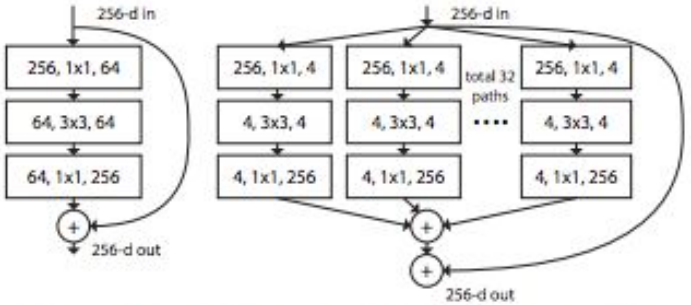
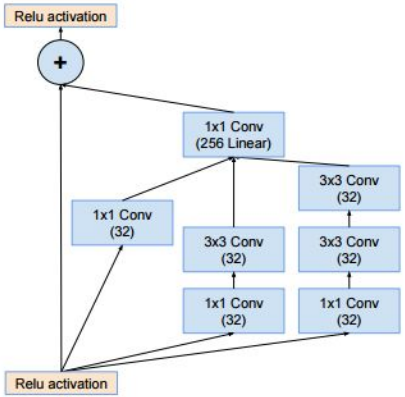


Figure 1. **Left**: A block of ResNet [13]. **Right**: A block of ResNeXt with cardinality = 32, with roughly the same complexity. A layer is shown as (# in channels, filter size, # out channels).

Saining Xie, Ross Girshick, Piotr Dollár, Zhuowen Tu, Kaiming He.
Aggregated Residual Transformations for Deep Neural Networks

What are the current trends?



Inception ResNet

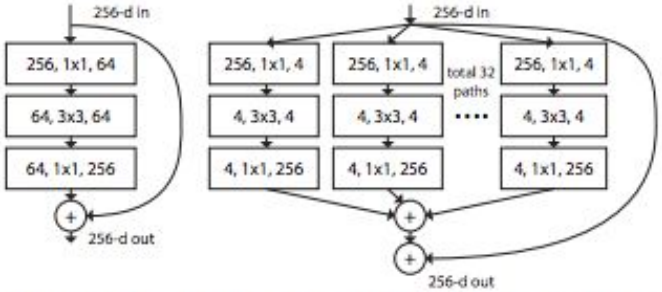


Figure 1. **Left**: A block of ResNet [13]. **Right**: A block of ResNeXt with cardinality = 32, with roughly the same complexity. A layer is shown as (# in channels, filter size, # out channels).

ResNeXt

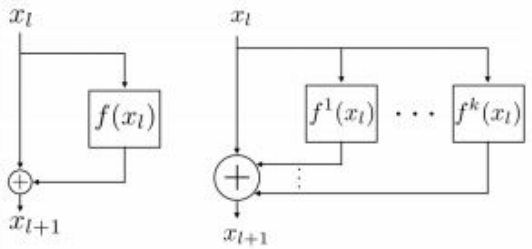


Figure 2: A residual block (left) versus a multi-residual block (right).

MultiResNet

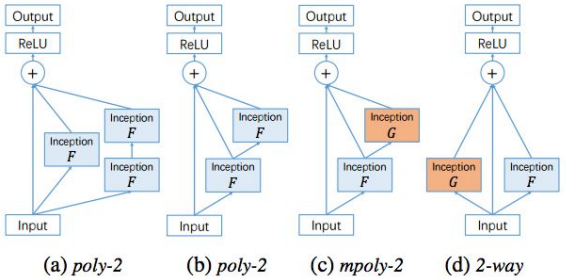


Figure 4: Examples of PolyInception structures.

PolyNet

Saining Xie, Ross Girshick, Piotr Dollár, Zhuowen Tu, Kaiming He. **Aggregated Residual Transformations for Deep Neural Networks**
Masoud Abdi, Saeid Nahavandi. Multi-Residual Networks: **Improving the Speed and Accuracy of Residual Networks**
Xingcheng Zhang, Zhizhong Li, Chen Change Loy, Dahua Lin. **PolyNet: A Pursuit of Structural Diversity in Very Deep Networks**
C. Szegedy et al., **Inception-v4, Inception-ResNet and the Impact of Residual Connections on Learning**

What are the current trends?

- Some make minor modifications to ResNets
- Biggest trend is to split of into several branches, and merge through summation
- **A couple architectures go crazy with branch & merge, without explicit identity connections**

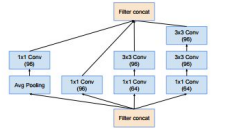


Figure 4. The schema for 35 × 35 grid modules of the pure Inception-v4 network. This is the Inception-A block of Figure 9.

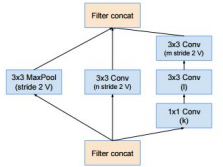


Figure 7. The schema for 35 × 35 to 17 × 17 reduction module. Different variants of this blocks (with various number of filters) are used in Figure 9, and 15 in each of the new Inception(-v4, ResNet-v1, -ResNet-v2) variants presented in this paper. The k, l, m, n numbers represent filter bank sizes which can be looked up in Table 1.

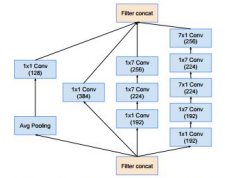


Figure 5. The schema for 17 × 17 grid modules of the pure Inception-v4 network. This is the Inception-B block of Figure 9.

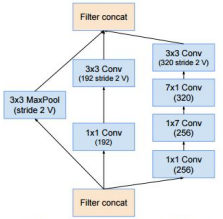


Figure 8. The schema for 17 × 17 to 8 × 8 grid-reduction module. This is the reduction module used by the pure Inception-v4 network in Figure 9.

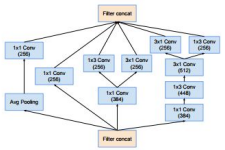
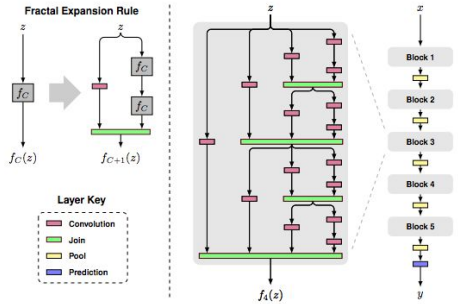


Figure 6. The schema for 8 × 8 grid modules of the pure Inception-v4 network. This is the Inception-C block of Figure 9.

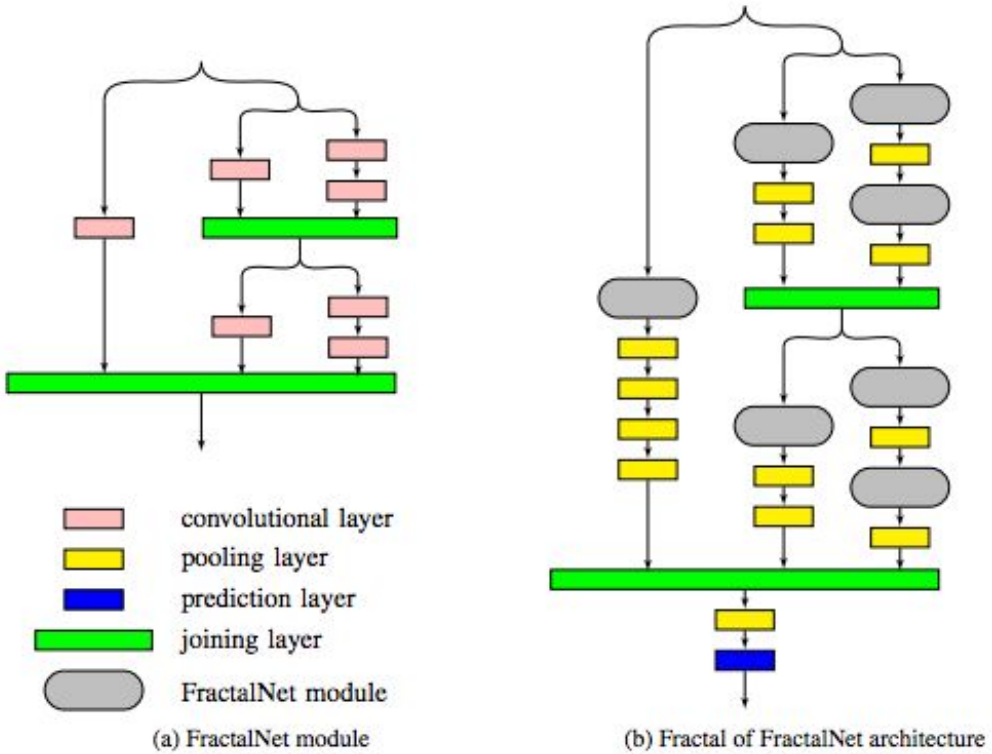


Gustav Larsson, Michael Maire, Gregory Shakhnarovich
FractalNet: Ultra-Deep Neural Networks without Residuals

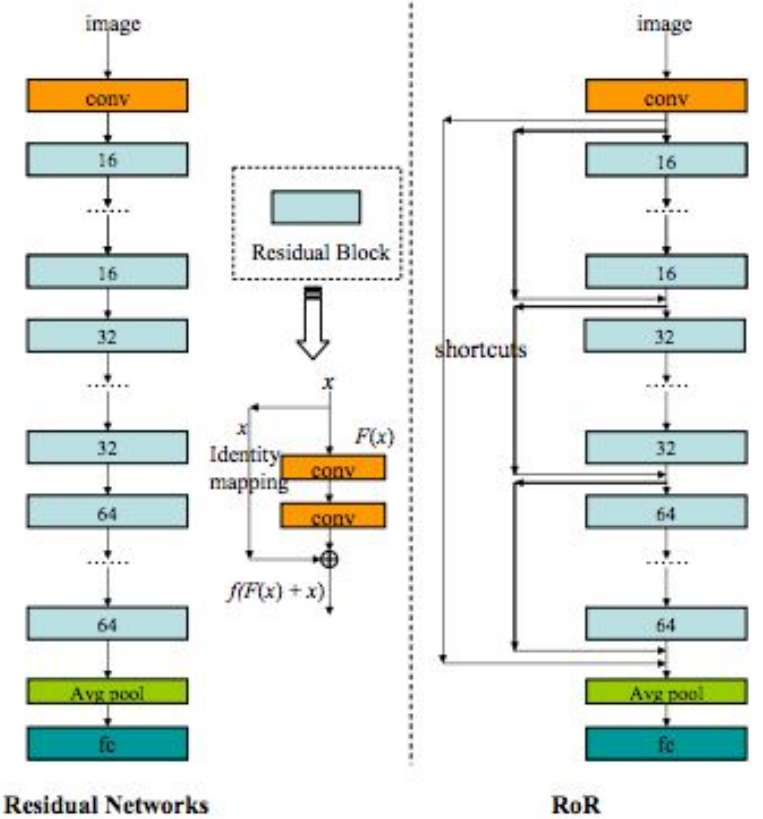
C. Szegedy et al., Inception-v4, Inception-ResNet and the Impact of Residual Connections on Learning

Some try going meta..

Fractal of Fractals



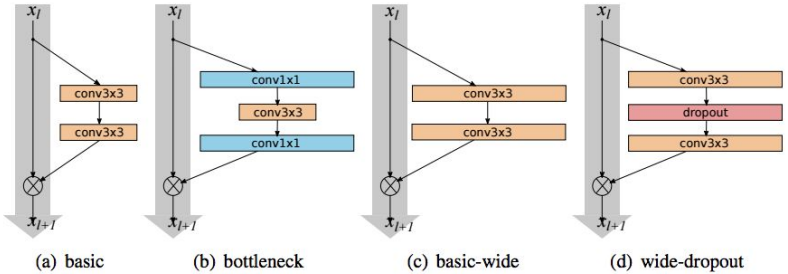
Residuals of Residuals



Leslie N. Smith, Nicholay Topin, Deep Convolutional Neural Network Design Patterns

ResNet tweaks: Wide ResNets

- **Use pre-activation ResNet's basic block with more feature maps**
- Used parameter "k" to encode width
- Investigated relationship between width and depth to find a good tradeoff



Sergey Zagoruyko, Nikos Komodakis
Wide Residual Networks

ResNet tweaks: Wide ResNets

- Use pre-activation ResNet's basic block with more feature maps
- **Used parameter "k" to encode width**
- Investigated relationship between width and depth to find a good tradeoff

group name	output size	block type = $B(3,3)$
conv1	32×32	$[3\times3,\ 16]$
conv2	32×32	$\begin{bmatrix} 3\times3,\ 16\times k \\ 3\times3,\ 16\times k \end{bmatrix} \times N$
conv3	16×16	$\begin{bmatrix} 3\times3,\ 32\times k \\ 3\times3,\ 32\times k \end{bmatrix} \times N$
conv4	8×8	$\begin{bmatrix} 3\times3,\ 64\times k \\ 3\times3,\ 64\times k \end{bmatrix} \times N$
avg-pool	1×1	$[8 \times 8]$

Sergey Zagoruyko, Nikos Komodakis
Wide Residual Networks

ResNet tweaks: Wide ResNets

- Use pre-activation ResNet's basic block with more feature maps
- Used parameter "k" to encode width
- **Investigated relationship between width and depth to find a good tradeoff**

depth	k	# params	CIFAR-10	CIFAR-100
40	1	0.6M	6.85	30.89
40	2	2.2M	5.33	26.04
40	4	8.9M	4.97	22.89
40	8	35.7M	4.66	-
28	10	36.5M	**4.17**	20.50
28	12	52.5M	4.33	**20.43**
22	8	17.2M	4.38	21.22
22	10	26.8M	4.44	20.75
16	8	11.0M	4.81	22.07
16	10	17.1M	4.56	21.59

Sergey Zagoruyko, Nikos Komodakis
Wide Residual Networks

ResNet tweaks: Wide ResNets

- Use pre-activation ResNet's basic block with more feature maps
- Used parameter "k" to encode width
- **Investigated relationship between width and depth to find a good tradeoff**

depth	*k*	# params	CIFAR-10	CIFAR-100
40	1	0.6M	6.85	30.89
40	2	2.2M	5.33	26.04
40	4	8.9M	4.97	22.89
40	8	35.7M	4.66	-
28	10	36.5M	**4.17**	20.50
28	12	52.5M	4.33	**20.43**
22	8	17.2M	4.38	21.22
22	10	26.8M	4.44	20.75
16	8	11.0M	4.81	22.07
16	10	17.1M	4.56	21.59

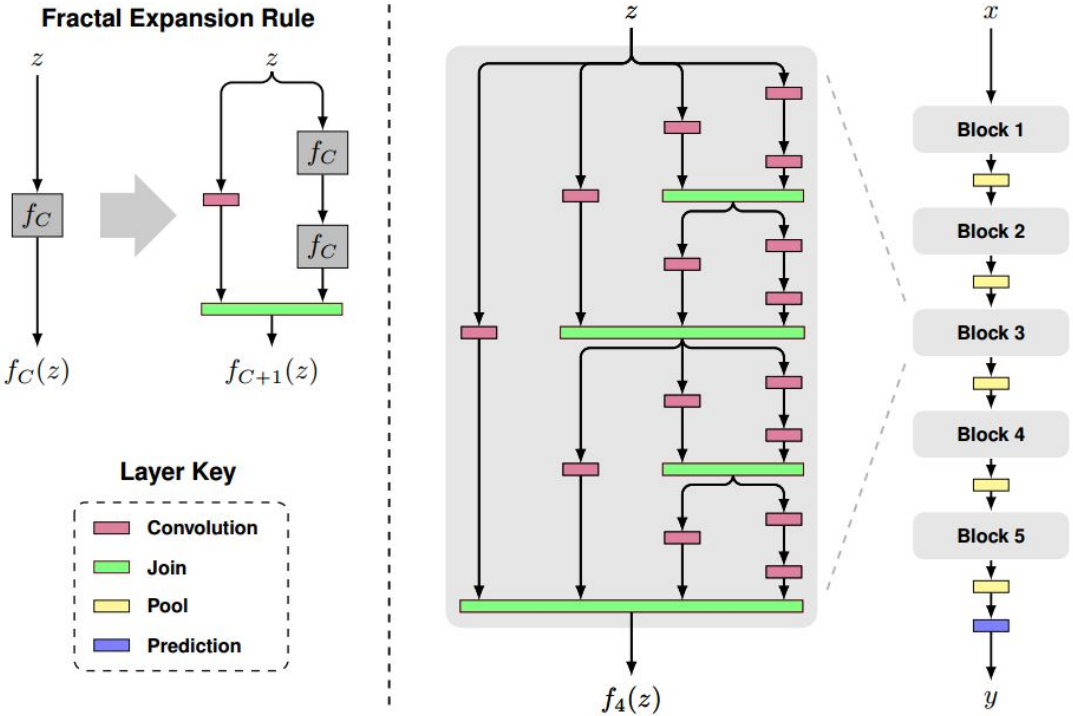
Sergey Zagoruyko, Nikos Komodakis
Wide Residual Networks

Aside from ResNets

FractalNet and DenseNet

FractalNet

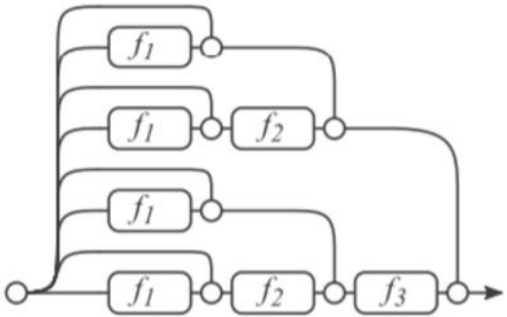
- A competitive extremely deep architecture that does not rely on residuals



Gustav Larsson, Michael Maire, Gregory Shakhnarovich
FractalNet: Ultra-Deep Neural Networks without Residuals

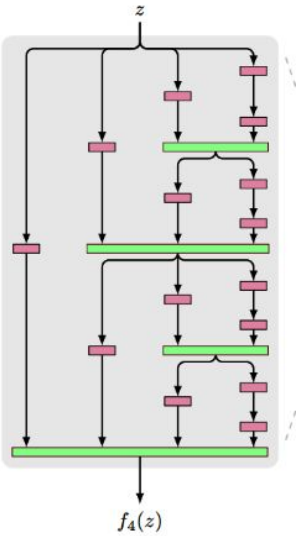
FractalNet

- A competitive extremely deep architecture that does not rely on residuals
- Interestingly, its architecture is similar to an unfolded ResNet



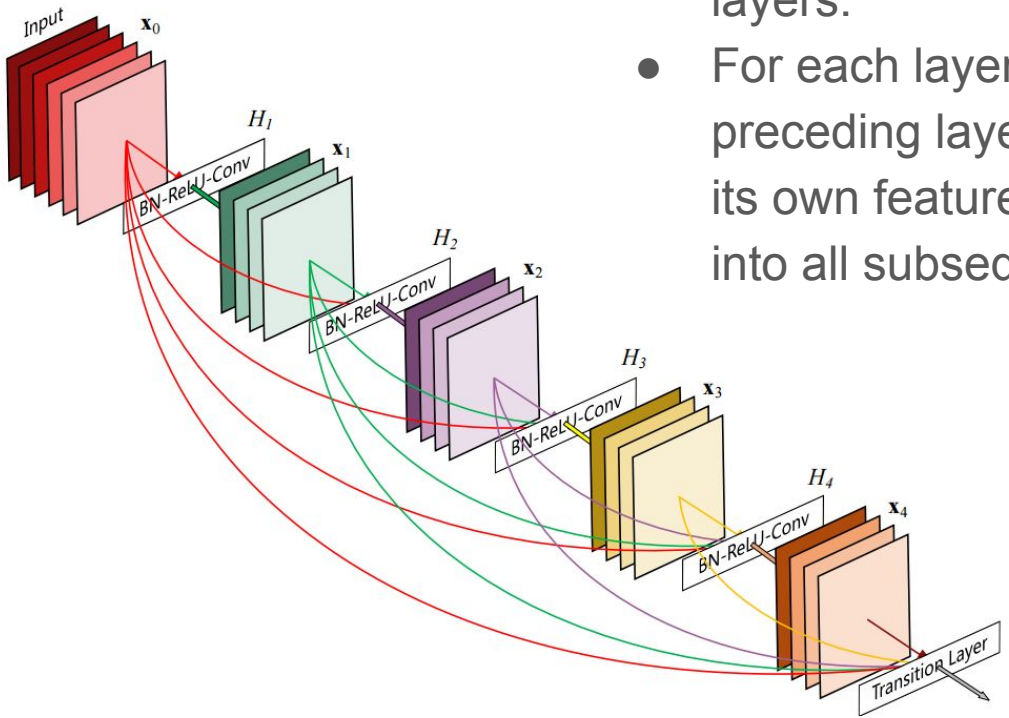
(b) Unraveled view of (a)

Andreas Veit, Michael Wilber, Serge Belongie, Residual Networks Behave Like Ensembles of Relatively Shallow Networks, arxiv 2016



Gustav Larsson, Michael Maire, Gregory Shakhnarovich FractalNet: Ultra-Deep Neural Networks without Residuals

DenseNet (Within a DenseBlock)



- Every layer is connected to all other layers.
- For each layer, the feature-maps of all preceding layers are used as inputs, and its own feature-maps are used as inputs into all subsequent layers.

DenseNet

- Alleviate the vanishing-gradient problem.
- Strengthen feature propagation.
- Encourage feature reuse.
- **Substantially** reduce the number of parameters.

THE ROADMAP

- INTRODUCTION
- DEEP RESIDUAL NETWORKS
- WHY DO DEEP RESIDUAL NETWORKS WORK?
- SURVEY OF ADVANCED CNN ARCHITECTURES
- **CONCLUSION**

CONCLUSION

- A highway/short path: the signal could be directly propagated from one unit to any other units, in both forward and backward passes.
- ResNET can be thought as ensembles of relatively shallow networks.
- Increasing the path in each Resnet block can improve the performance.

Thank you!