Deep Residual Learning for Image Recognition

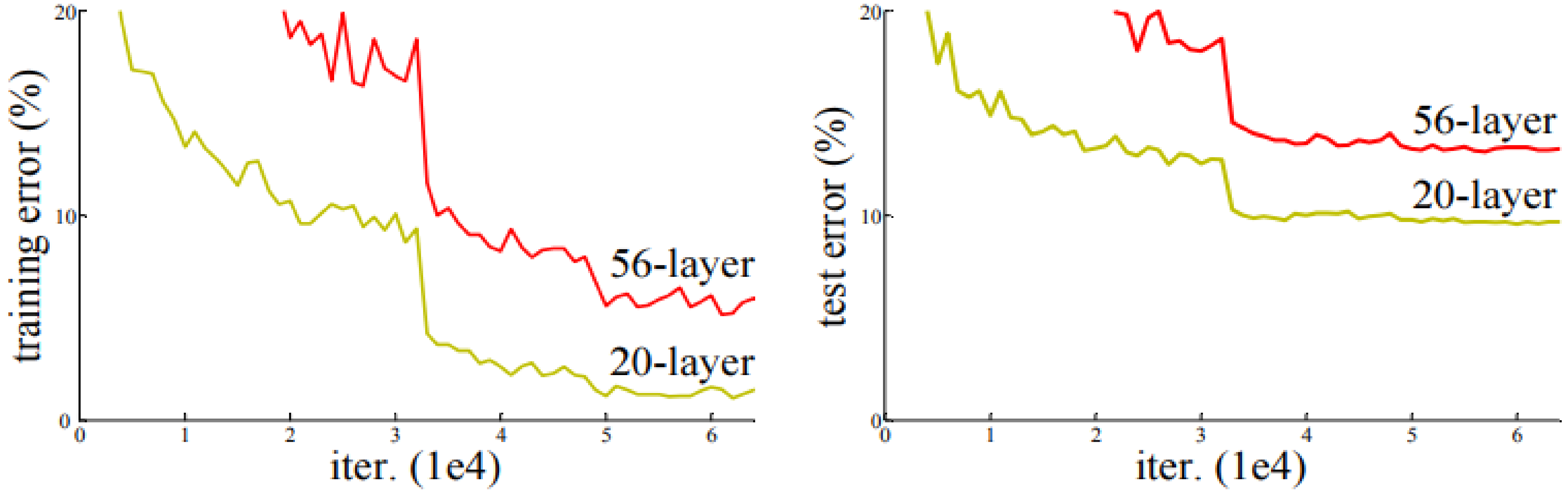
CONTENTS

01 Introduction

the **depth of representations** is of central importance for many visual recognition tasks

breakthroughs

- levels of features can be enriched by the number of stacked layers
- gradient vianising/exploding problem
  → nomalized initailization
- degradation problem (not overfitting, higher training error)



we address the degradation problem by introducing a deep residual learning framework.

01 Introduction

Will show

- extremely deep residual nets are easy to optimize, but the counterpart "plain" nets exhibit higher training error when the depth increases
- deep residual nets can easily enjoy accuracy gains from greatly increased depth

02 ResNet

residual mapping

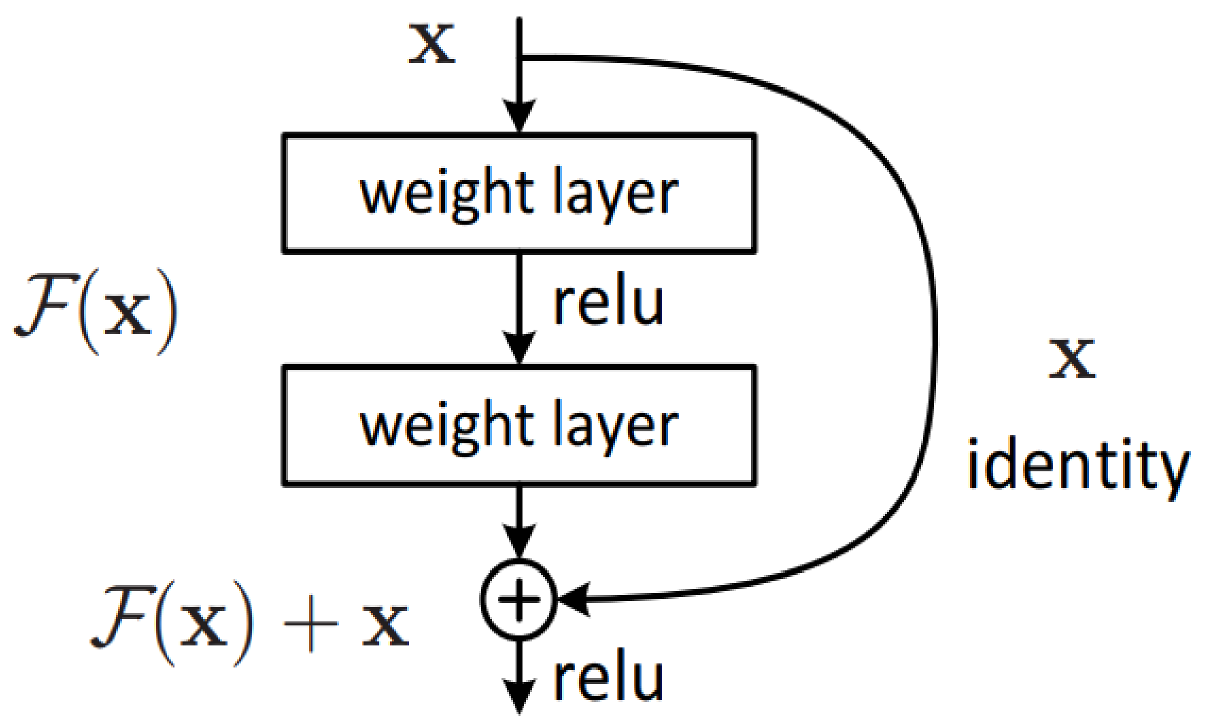
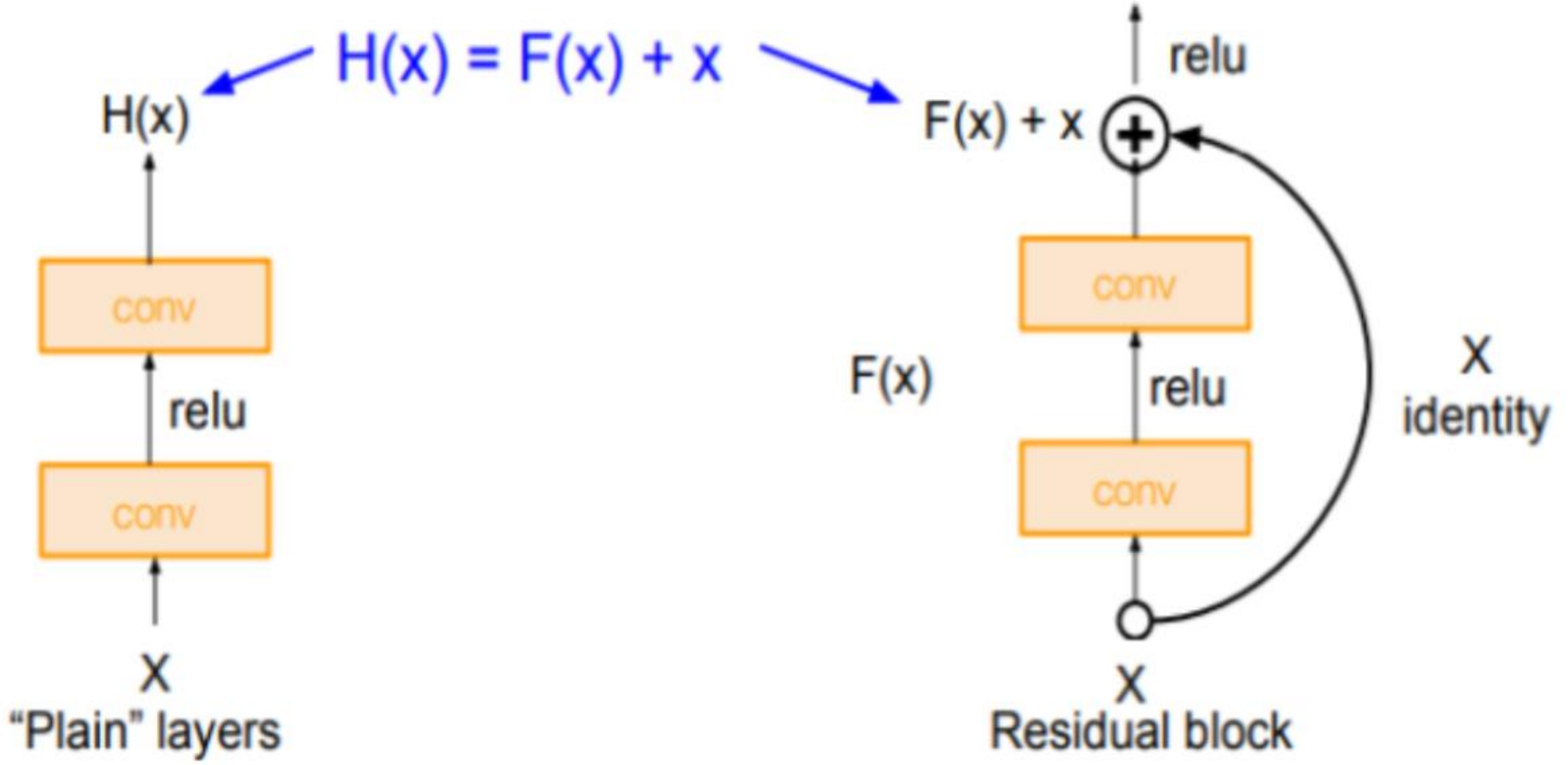


Figure 2. Residual learning: a building block.



H(x) : a desired underlying mapping
F(x) : residual mapping
F(x) + x : shortcut connection

It would be easier to push the residual to zero than to fit an identity mapping by a stack of nonlinear layers

02 ResNet

Building block

$$\mathbf{y} = \underbrace{\mathcal{F}(\mathbf{x}, \{W_i\})}_{\text{Residual mapping}} + \mathbf{x}. \qquad \mathcal{F} = W_2\sigma(W_1\mathbf{x})$$

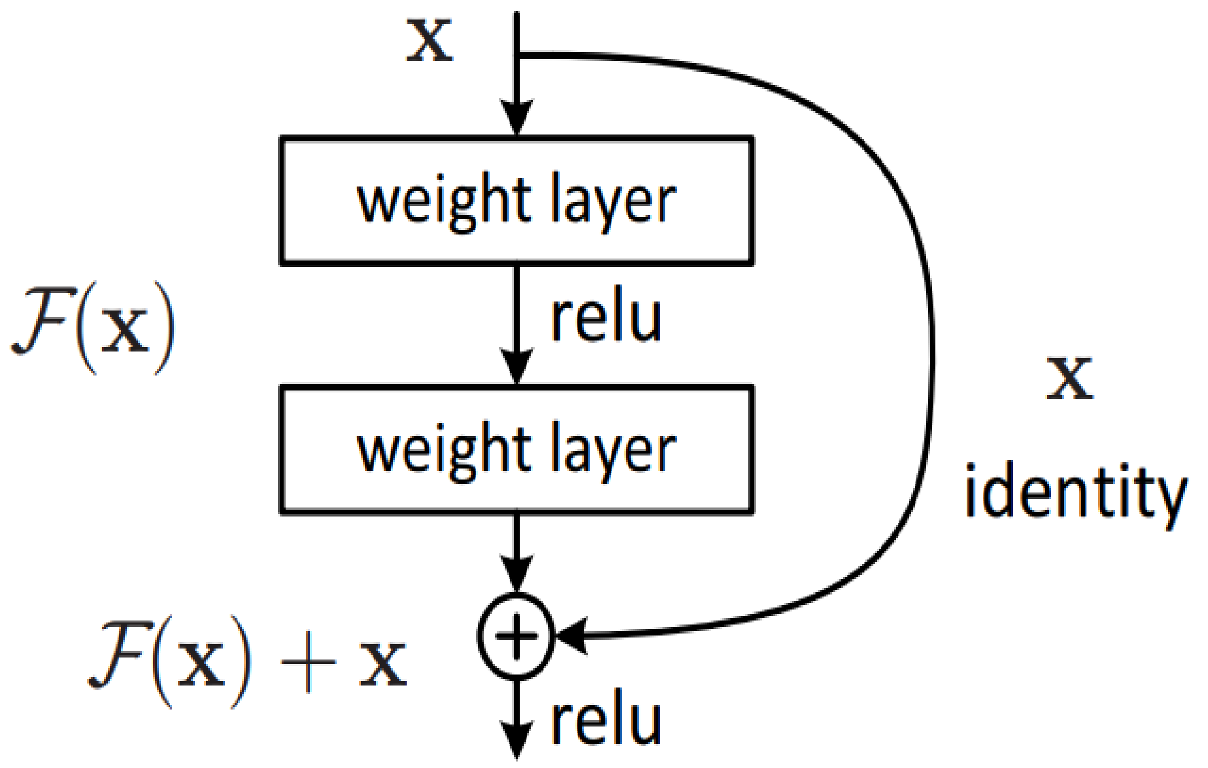


Figure 2. Residual learning: a building block.

If dimensions of x and F is not equal, can perform a linear projection Ws by the shortcut connections to match the dimensions.

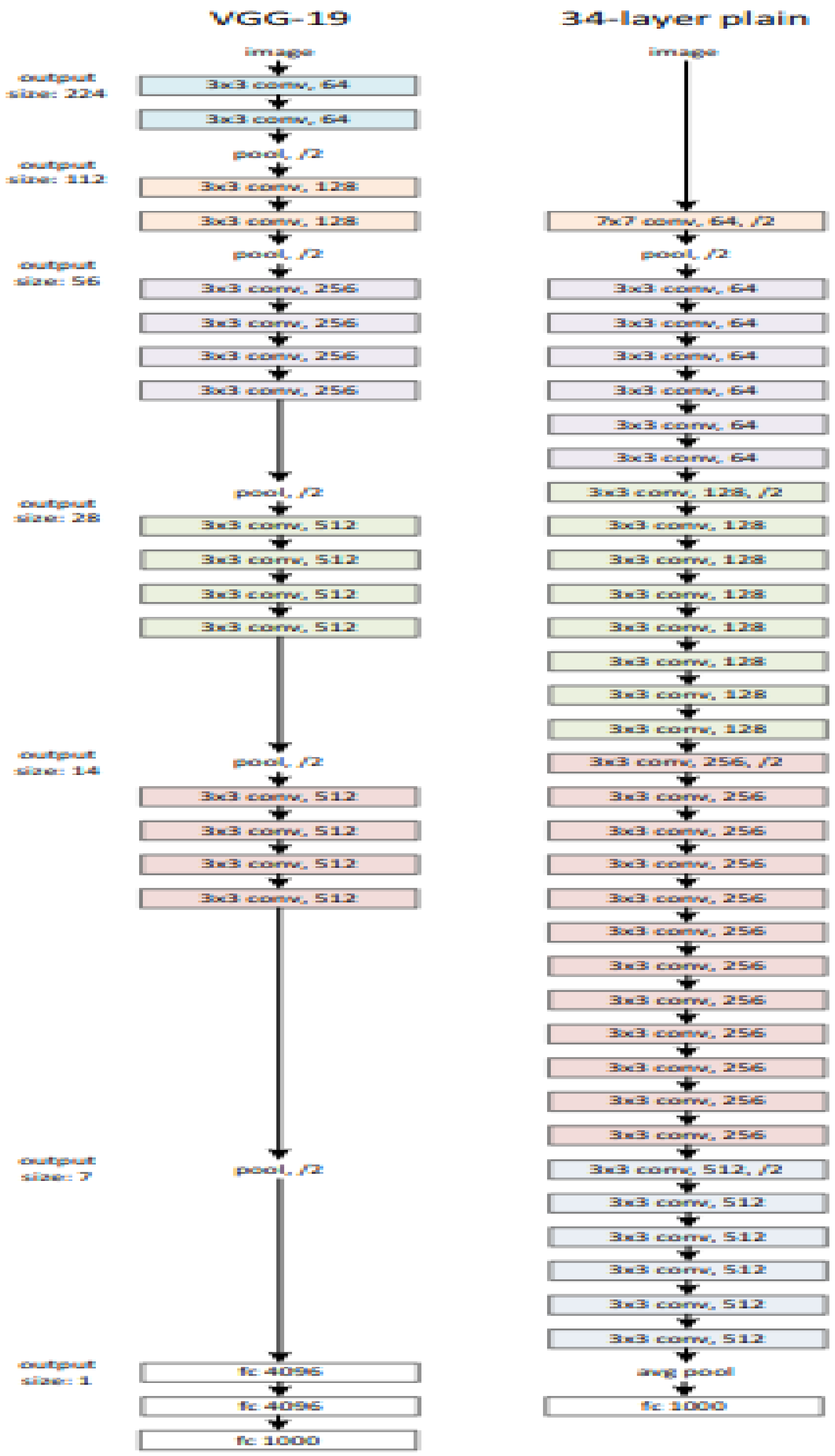
$$\mathbf{y} = \mathcal{F}(\mathbf{x}, \{W_i\}) + W_s\mathbf{x}.$$

02 ResNet

Network Architectures

Plain Network

- Inspired by the philosophy of VGGnets
- Convolutional layers mostly have 3x3 filters
- For the same output feature map size, the layers have the same number of filters
- If the feature map size is halved, the number of filters is doubled so as to preserve the time complexity per layer.



02 ResNet

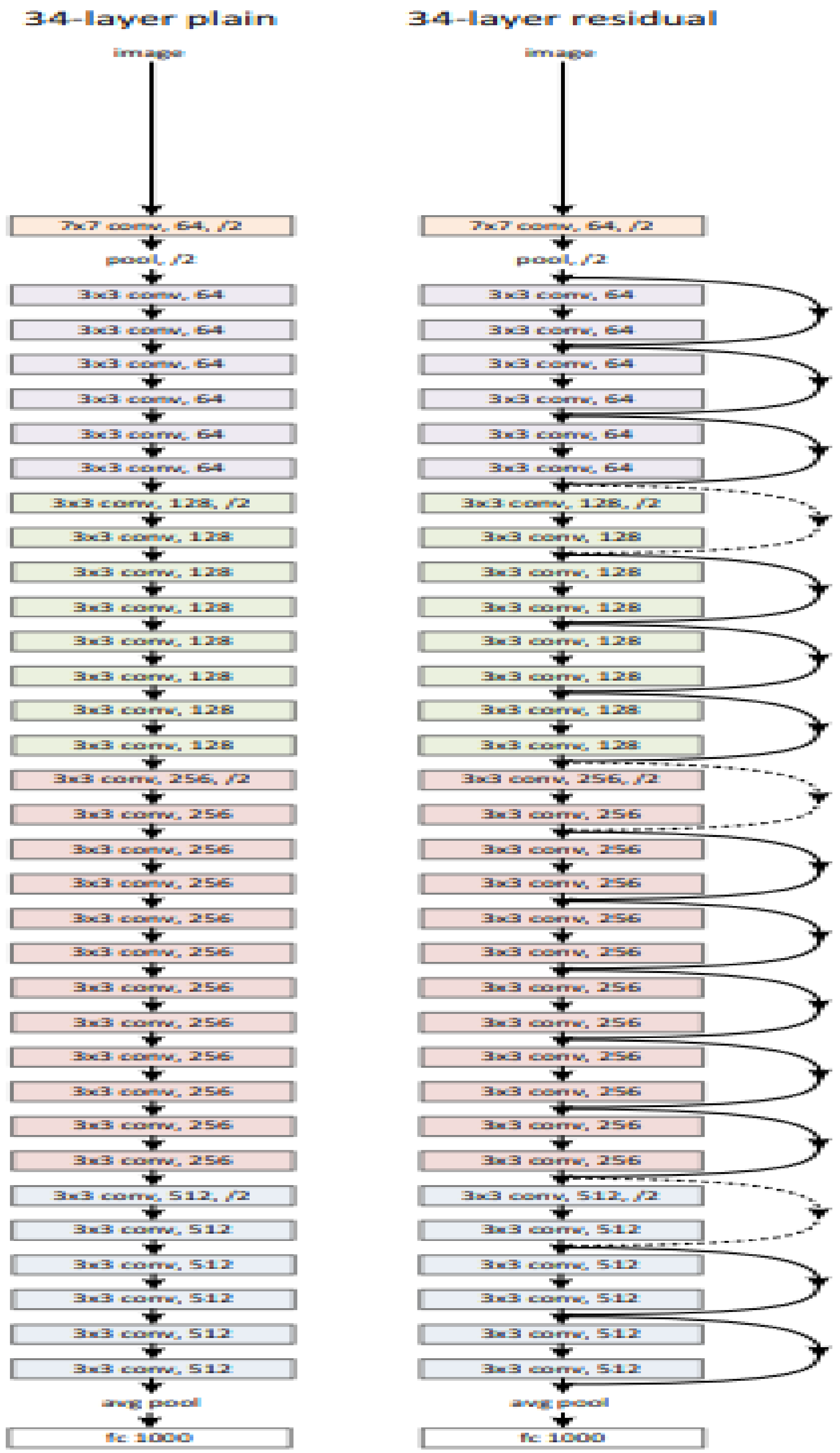
Network Architectures

Plain Network

- Inspired by the philosophy of VGGnets
- Convolutional layers mostly have 3x3 filters
- For the same output feature map size, the layers have the same number of filters
- If the feature map size is halved, the number of filters is doubled so as to preserve the time complexity per layer.

Residual Network

- Based on the above plain network, insert shortcut connections
- When dimensions increase, 1) shortcut still perform, 2) projection shortcut



03 Experiments

ImageNet Classification

1. Resize with randomly sampled in [256, 480]
2. 224 x 224 crop randomly (+ horizontal flip version)

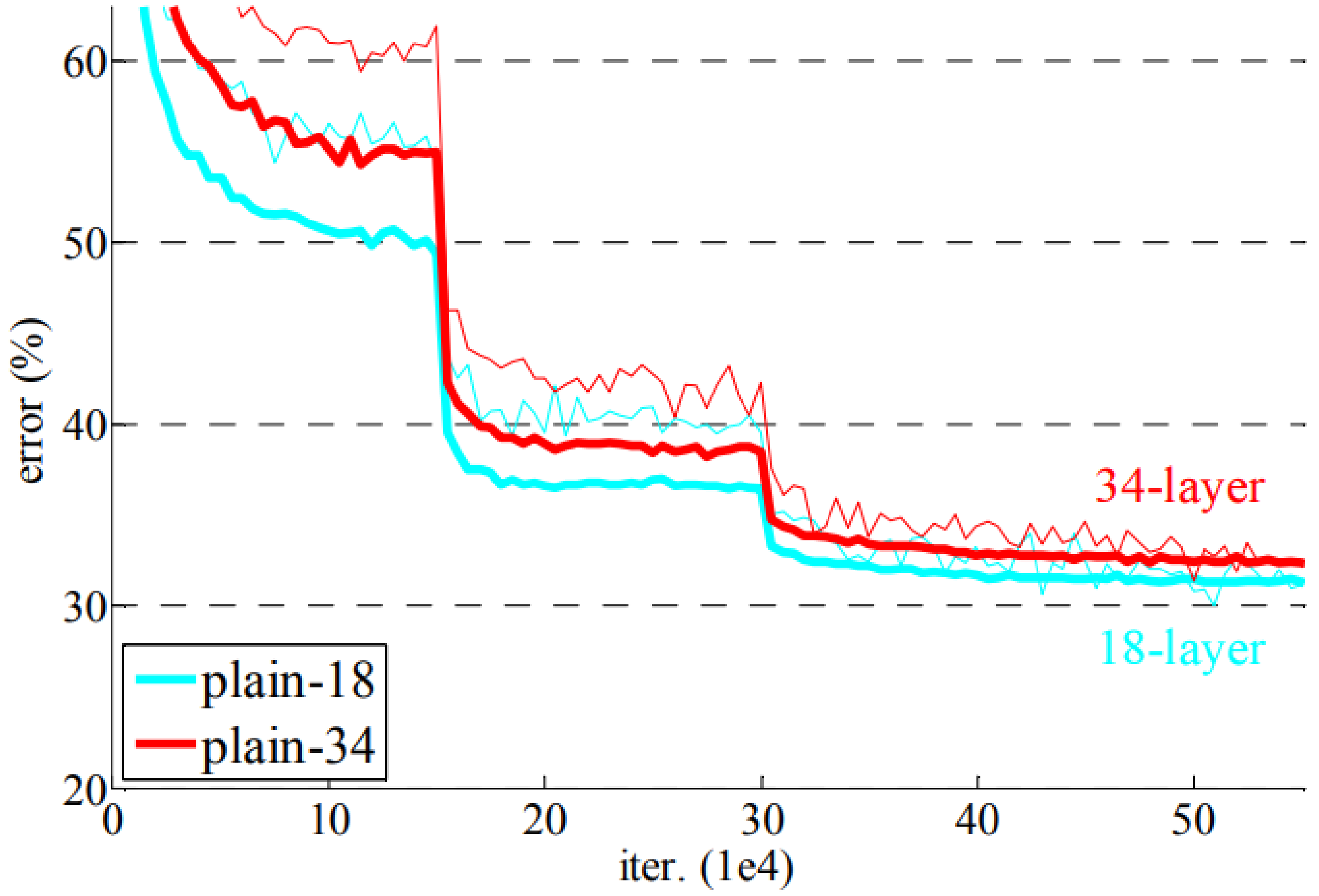
ImageNet 2021 classification dataset

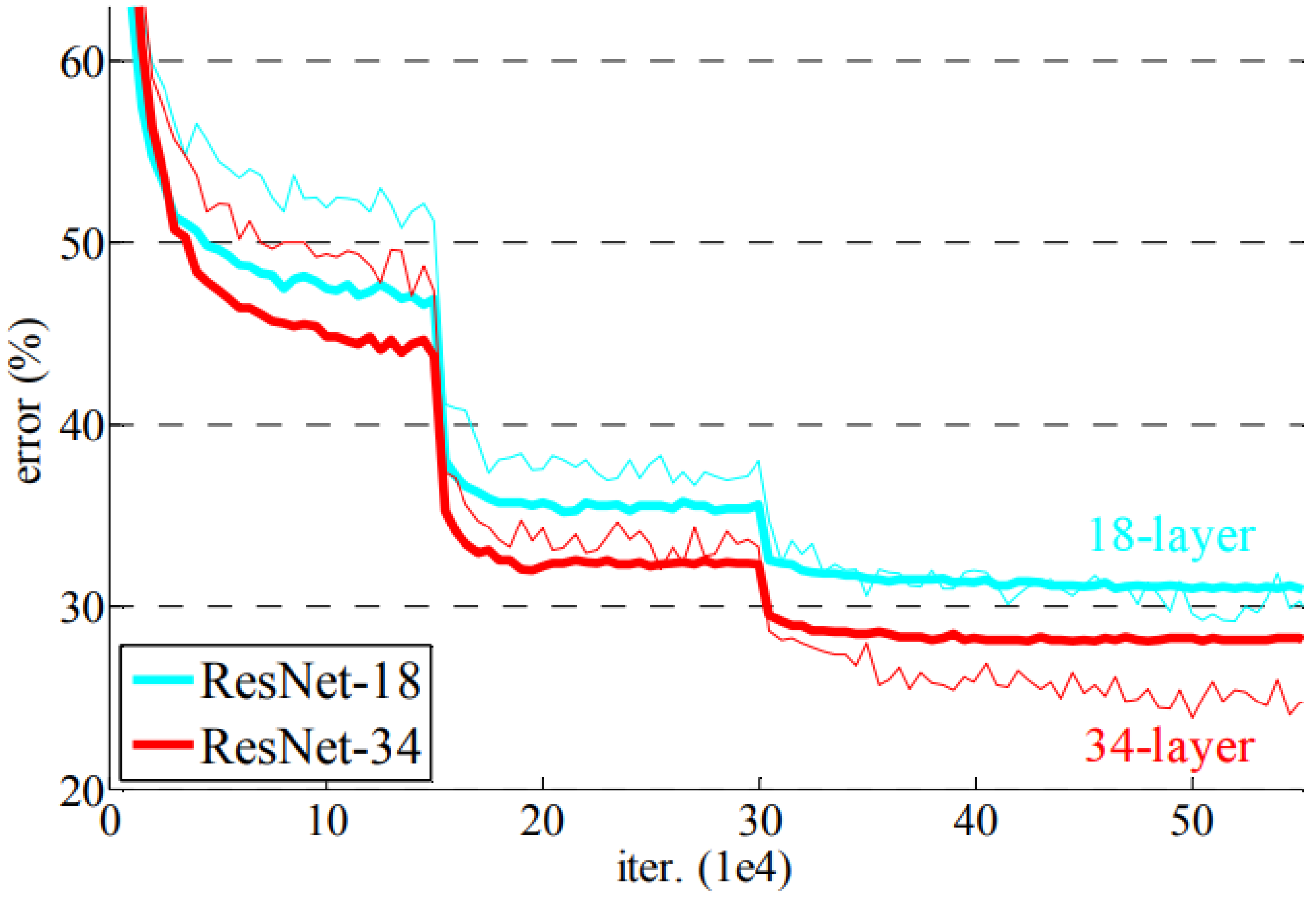
- 1.28 training images
- 50k validation images
- 100k test images

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

03 Experiments

ImageNet Classification – validation error





	plain	ResNet
18 layers	27.94	27.88
34 layers	28.54	**25.03**

This indicates that the degradation problem is well addressed in resnet and we manage to obtain accuracy gains from increased depth

03 Experiments

ImageNet Classification

model	top-1 err.	top-5 err.
VGG-16 [40]	28.07	9.33
GoogLeNet [43]	-	9.15
PReLU-net [12]	24.27	7.38
plain-34	28.54	10.02
ResNet-34 A	25.03	7.76
ResNet-34 B	24.52	7.46
ResNet-34 C	24.19	7.40
ResNet-50	22.85	6.71
ResNet-101	21.75	6.05
ResNet-152	**21.43**	**5.71**

A – zero-padding shortcuts are used for increasing dimensions
B – projection shortcuts are used for increasing dimensions
C – all shortcuts are projection

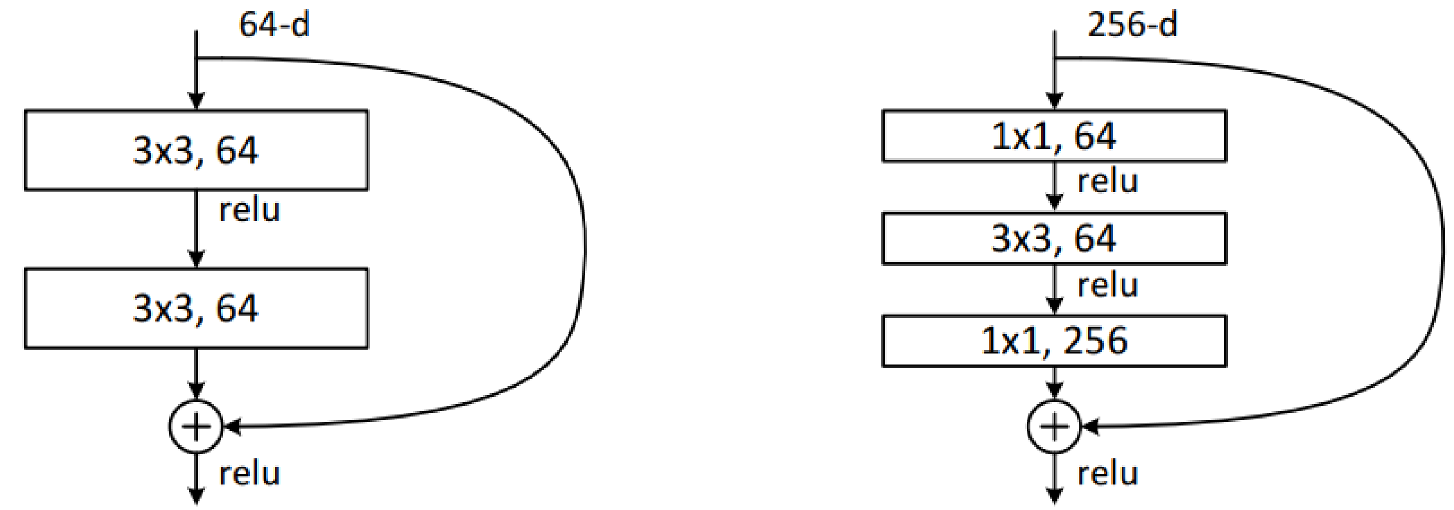
Projection shortcuts are not essential for addressing the degradation problem.

→ not use C

03 Experiments

ImageNet Classification – Bottle neck (with option B)

Both designs have similar time complexity



plain-34	28.54	10.02
ResNet-34 A	25.03	7.76
ResNet-34 B	24.52	7.46
ResNet-34 C	24.19	7.40
ResNet-50	22.85	6.71
ResNet-101	21.75	6.05
ResNet-152	**21.43**	**5.71**

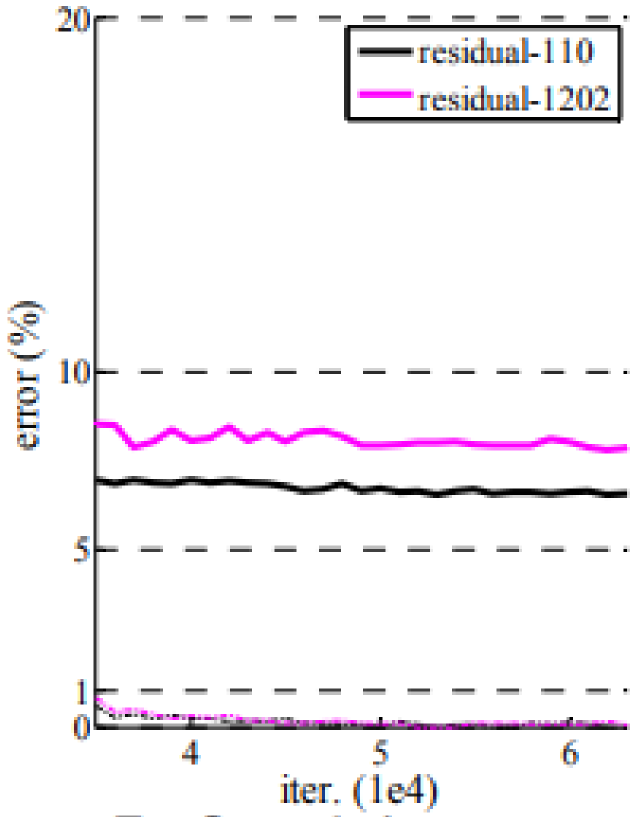
50/101/152-layer ResNets

50-layer	101-layer	152-layer
7×7, 64, stride 2		
3×3 max pool, stride 2		
$\left[\begin{array}{c}1\times1, 64\\3\times3, 64\\1\times1, 256\end{array}\right]\times3$	$\left[\begin{array}{c}1\times1, 64\\3\times3, 64\\1\times1, 256\end{array}\right]\times3$	$\left[\begin{array}{c}1\times1, 64\\3\times3, 64\\1\times1, 256\end{array}\right]\times3$
$\left[\begin{array}{c}1\times1, 128\\3\times3, 128\\1\times1, 512\end{array}\right]\times4$	$\left[\begin{array}{c}1\times1, 128\\3\times3, 128\\1\times1, 512\end{array}\right]\times4$	$\left[\begin{array}{c}1\times1, 128\\3\times3, 128\\1\times1, 512\end{array}\right]\times8$
$\left[\begin{array}{c}1\times1, 256\\3\times3, 256\\1\times1, 1024\end{array}\right]\times6$	$\left[\begin{array}{c}1\times1, 256\\3\times3, 256\\1\times1, 1024\end{array}\right]\times23$	$\left[\begin{array}{c}1\times1, 256\\3\times3, 256\\1\times1, 1024\end{array}\right]\times36$
$\left[\begin{array}{c}1\times1, 512\\3\times3, 512\\1\times1, 2048\end{array}\right]\times3$	$\left[\begin{array}{c}1\times1, 512\\3\times3, 512\\1\times1, 2048\end{array}\right]\times3$	$\left[\begin{array}{c}1\times1, 512\\3\times3, 512\\1\times1, 2048\end{array}\right]\times3$
erage pool, 1000-d fc, softmax		
3.8×10^9	7.6×10^9	11.3×10^9

03 Experiments

Training on CIFAR-10, Exploring over 1000 layers

method			error (%)
Maxout [9]			9.38
NIN [25]			8.81
DSN [24]			8.22
	# layers	# params	
FitNet [34]	19	2.5M	8.39
Highway [41, 42]	19	2.3M	7.54 (7.72±0.16)
Highway [41, 42]	32	1.25M	8.80
ResNet	20	0.27M	8.75
ResNet	32	0.46M	7.51
ResNet	44	0.66M	7.17
ResNet	56	0.85M	6.97
ResNet	110	1.7M	**6.43** (6.61±0.16)
ResNet	1202	19.4M	7.93



The 1202-layer network may be unnecessarily large (19.4M) for this small dataset. (CIFAR-10)
→ **because of overfitting.**

combining with stronger regularization may improve result.