Deep Residual Learning

Workshop
on
Intro to Deep Neural Networks
26th to 27th August 2016

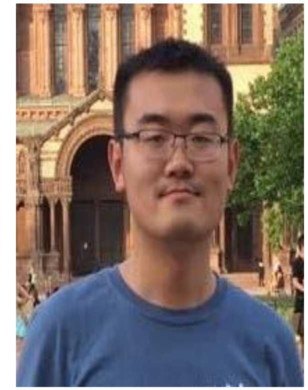
Presented by:
Noorul Wahab
(PhD Student)
Supervised by:
Dr. Asifullah Khan
DCIS, PIEAS

Pattern Recognition Lab
Department of Computer Science & Information Sciences
Pakistan Institute of Engineering & Applied Sciences

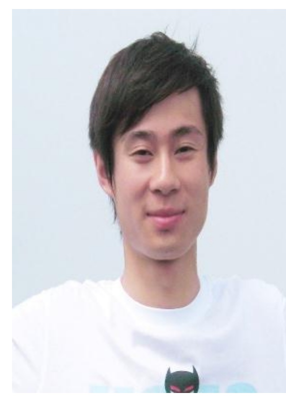
Authors

- Worked for Microsoft Research Asia (MSRA)
- Currently He is a Research Scientist at Facebook AI Research (FAIR).
- Ref [1] He, Kaiming, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. "Deep residual learning for image recognition." *arXiv preprint arXiv:1512.03385* (2015)

Kaiming He



Xiangyu Zhang



Shaoqing Ren

Jian Sun

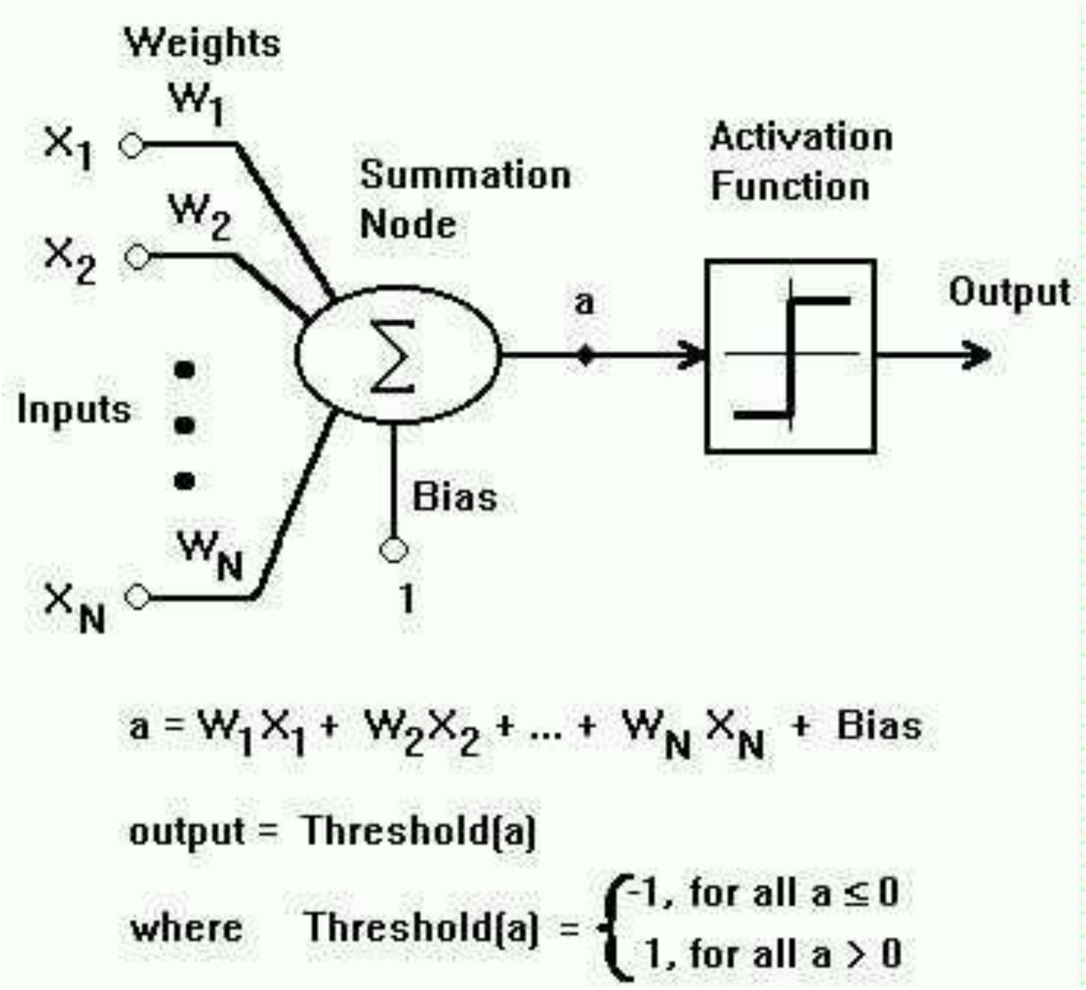
MSRA team

Abstract

- Deep features are important for visual recognition tasks, but deep nets suffer from vanishing/exploding gradients.
- Also adding more layers results in higher training error (as reported by the results of the experiments in this paper) .
- The proposed ResNet: learn residual functions instead of unreferenced functions.

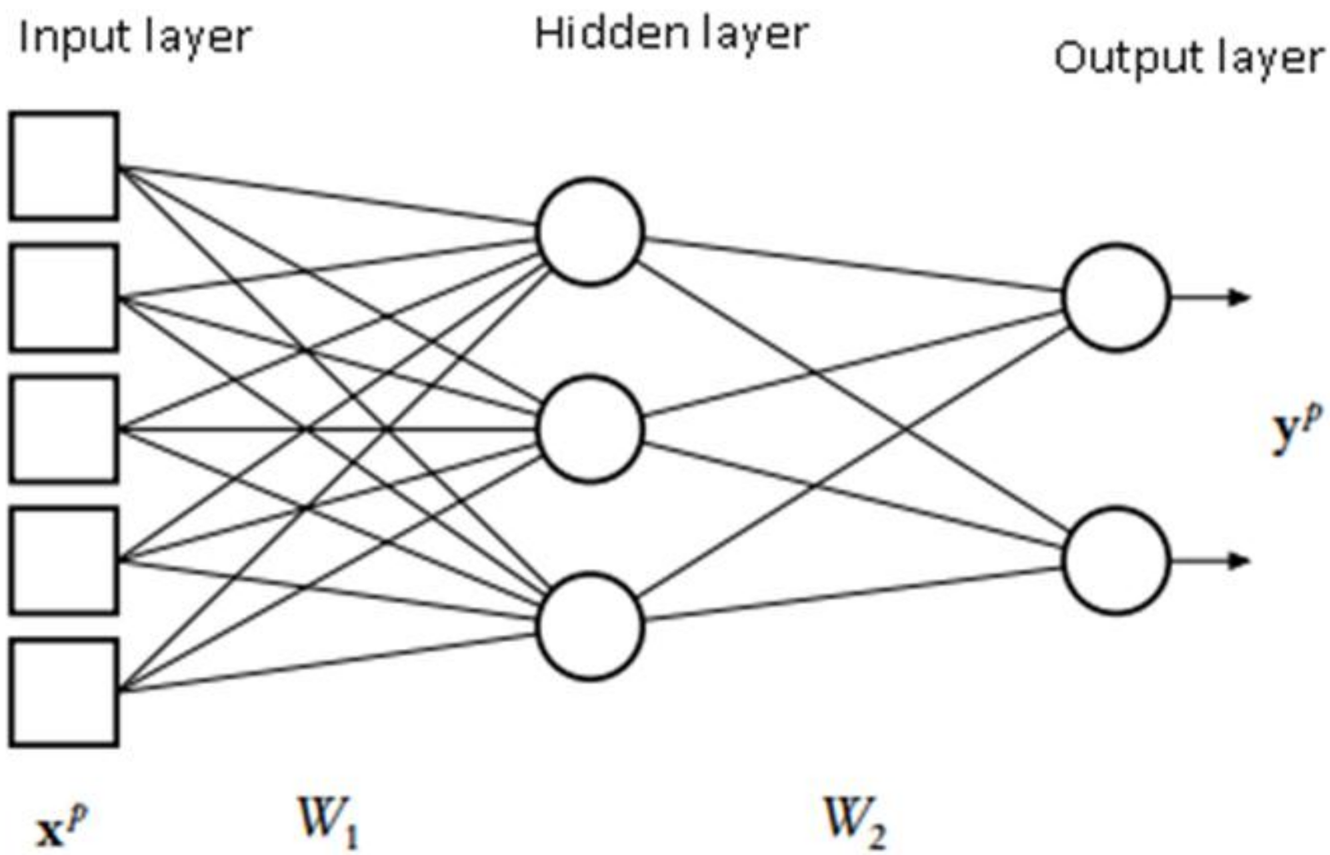
Background

Single layer perceptron



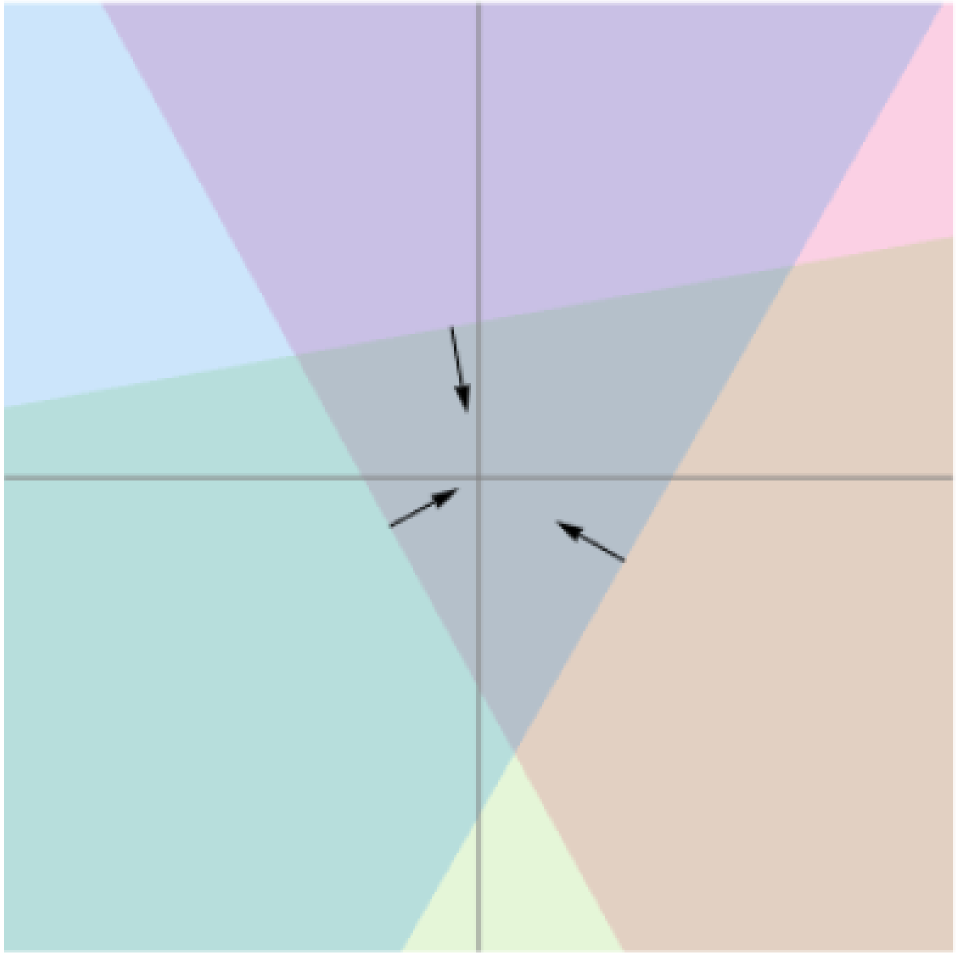
Background

A three-layer neural network



Background

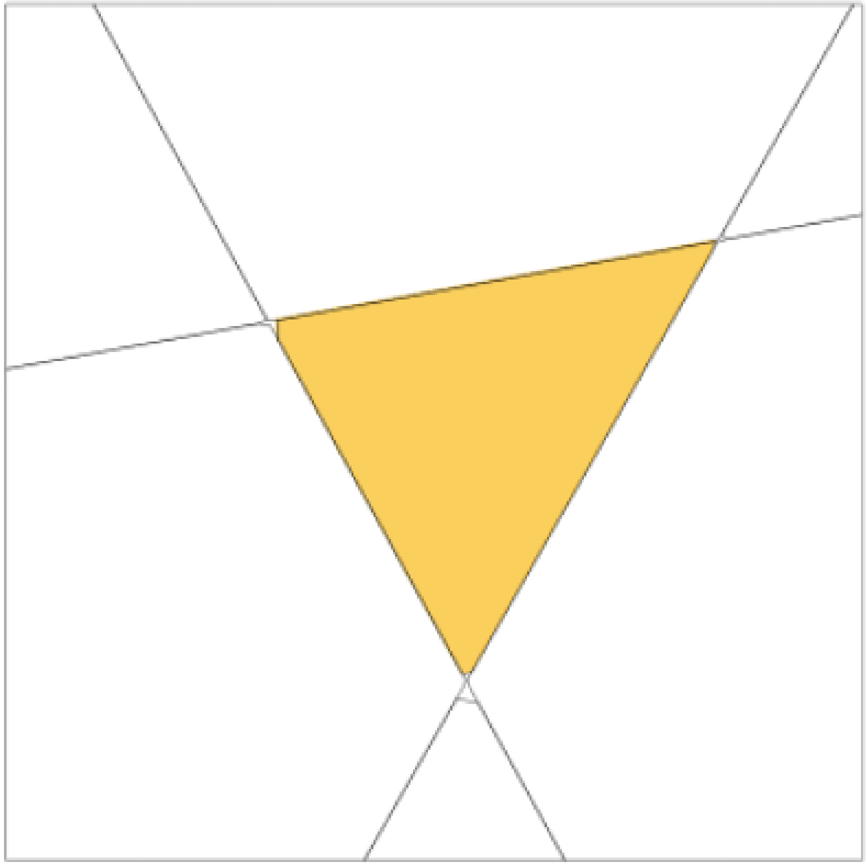
Working of multi-layer neural nets



Decision boundaries of three hidden neurons

Background

Working of multi-layer neural nets



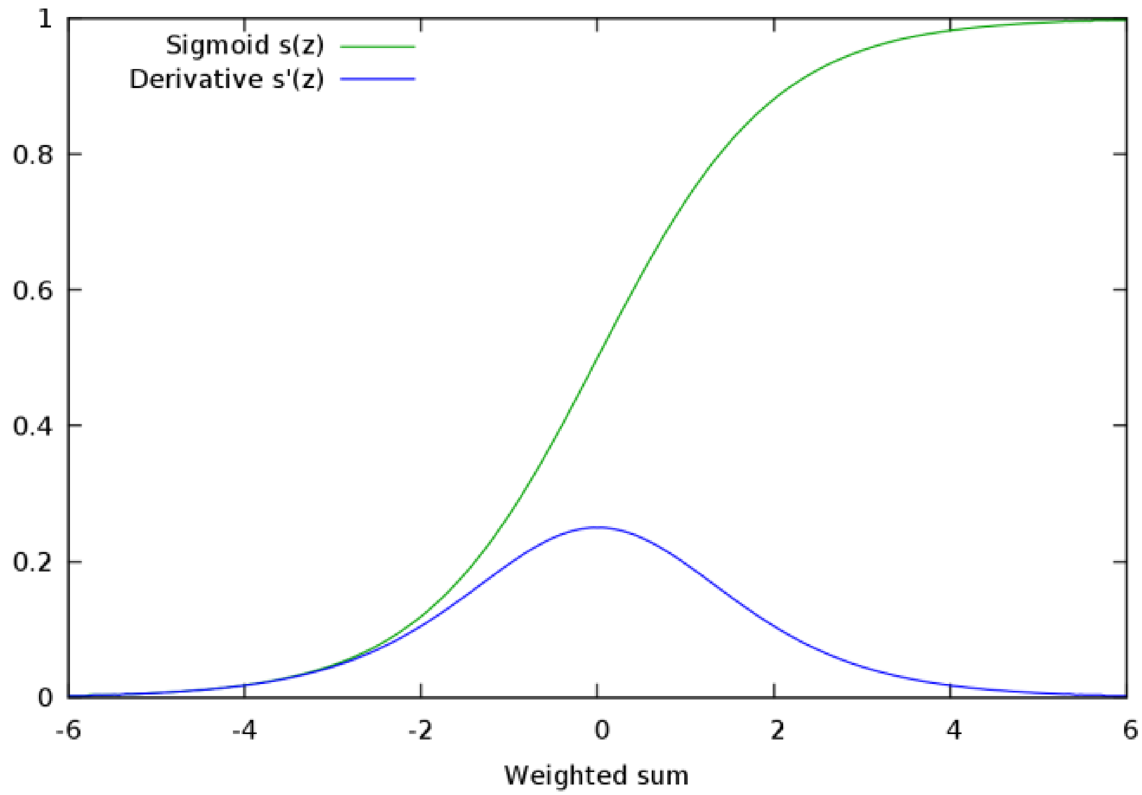
Demo

Decision boundary of the output neuron based on the decision boundaries of three hidden neurons

Background

Challenges in training deep nets

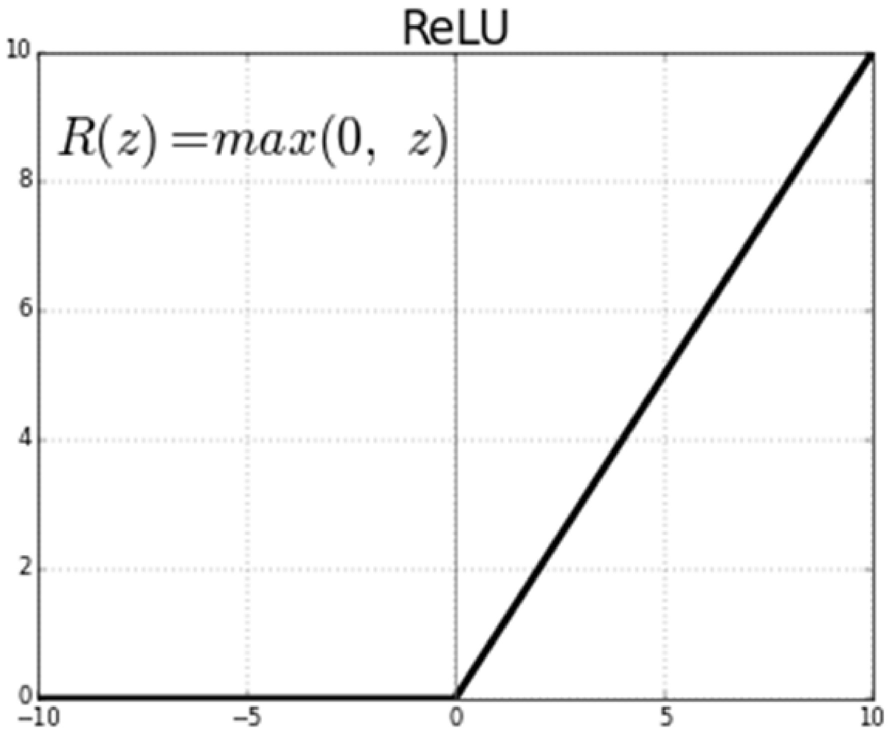
- Sigmoid neurons stop learning when they saturate (i-e when their output is either 0 or 1)



Background

Challenges in training deep nets

- ReLU reduces likelihood of vanishing gradients
- But stops learning entirely when the input to a rectified linear unit is negative.



Background

Challenges in training deep nets

- The point-wise derivative for ReLU is

$$\frac{dy}{dx} = \begin{cases} 1 & x > \epsilon \\ 0 & x \leq \epsilon \end{cases}$$

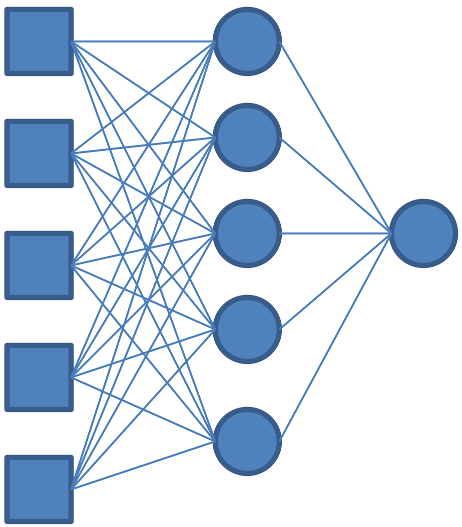
- A Leaky ReLU can help fix the “dying ReLU” problem

$$\frac{dy}{dx} = \begin{cases} 1 & x > 0 \\ 0.01 & x \leq 0 \end{cases}$$

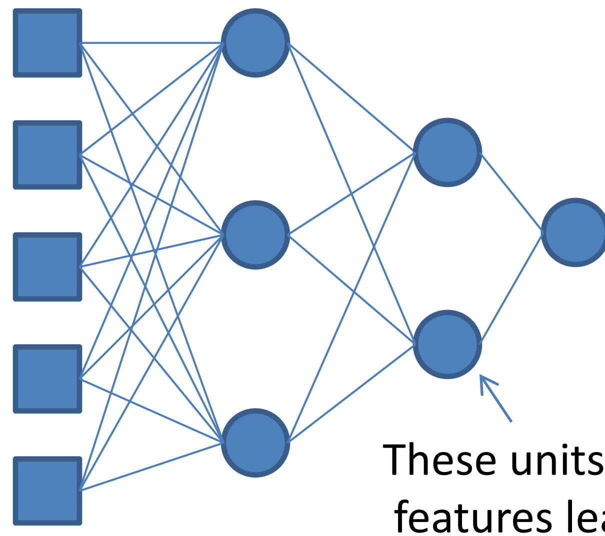
Background

Why multi-layer hierarchy?

- Shallow architectures are inefficient at representing deep functions
- Deep net, deep (enriched) features



Params: 5x5+5=30

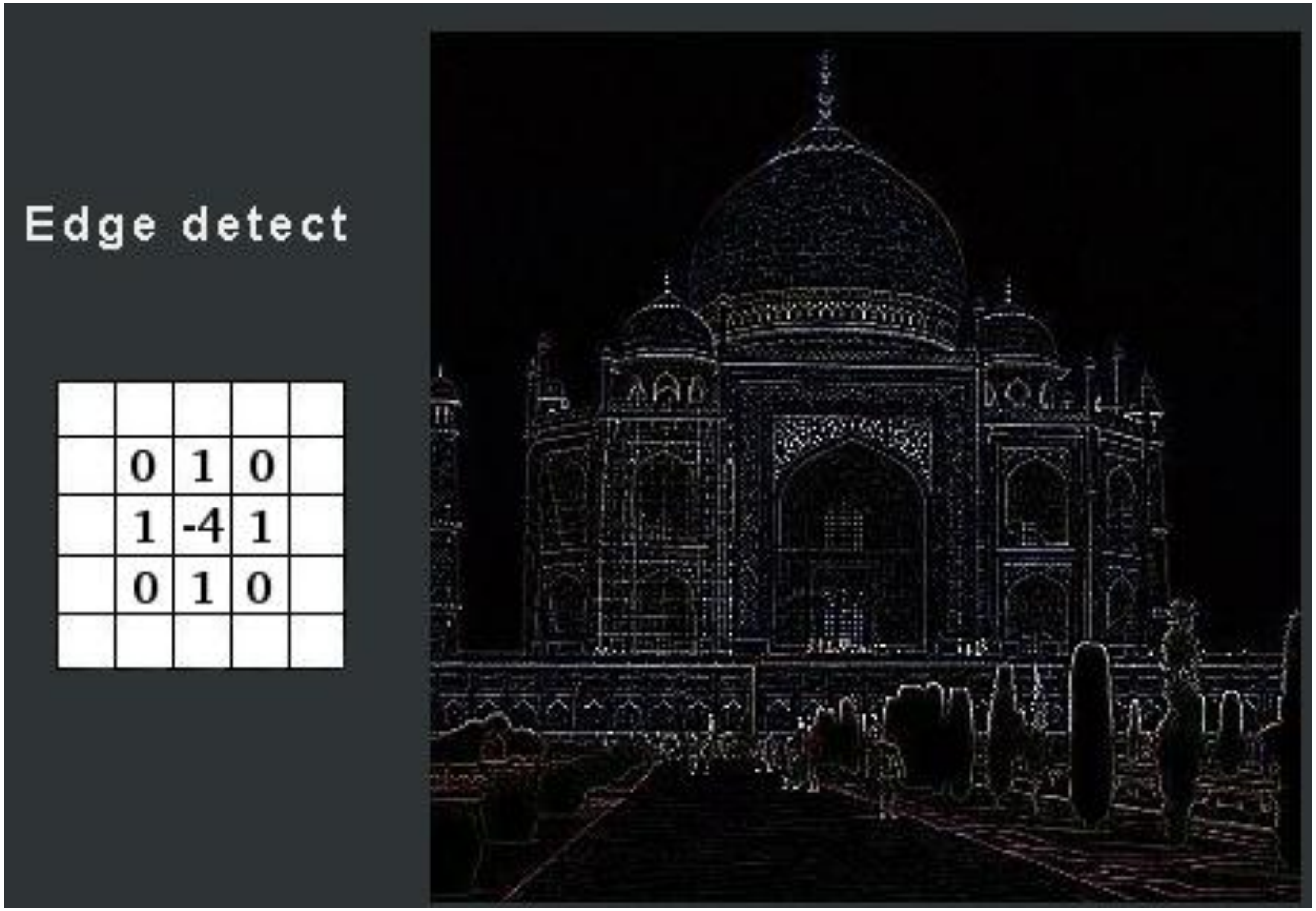


These units fine-tune the features learned by those in the previous layer

Params: 5x3+6+2=23

Background

Filters/kernels/features



Impossible
CanStock

Motivation behind ResNet

Training deep nets

- Is learning better networks as easy as stacking more layers?
- An obstacle to answering this question was the notorious problem of vanishing/exploding gradients
- Normalized initialization, intermediate normalization layers and ReLU addresses this problem to some extent

Motivation behind ResNet

Training deep nets

- Adding more layers to a suitably deep model leads to higher training error.
- Unexpectedly, the degradation problem in deeper networks is not caused by overfitting
- The degradation (of training accuracy) indicates that not all systems are similarly easy to optimize

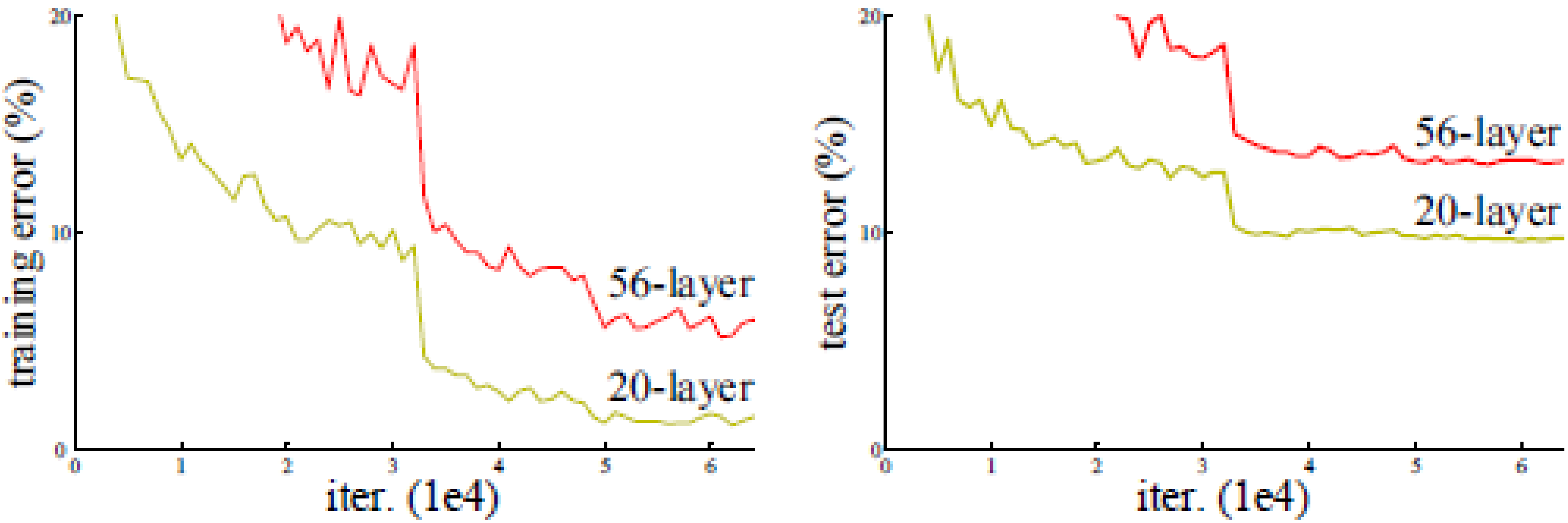
Motivation behind ResNet

Datasets

- CIFAR-10 dataset consists of 60000 32x32 color images in 10 classes, with 6000 images per class. There are 50000 training images and 10000 test images.
- ImageNet 2012: The training data contain 1000 categories and 1.2 million images.100k test images. 482x415 pixels average resolution.

Motivation behind ResNet

Training deep nets



on CIFAR-10

Conjecture: deep plain nets may have exponentially low convergence rates. Not studied in this work.

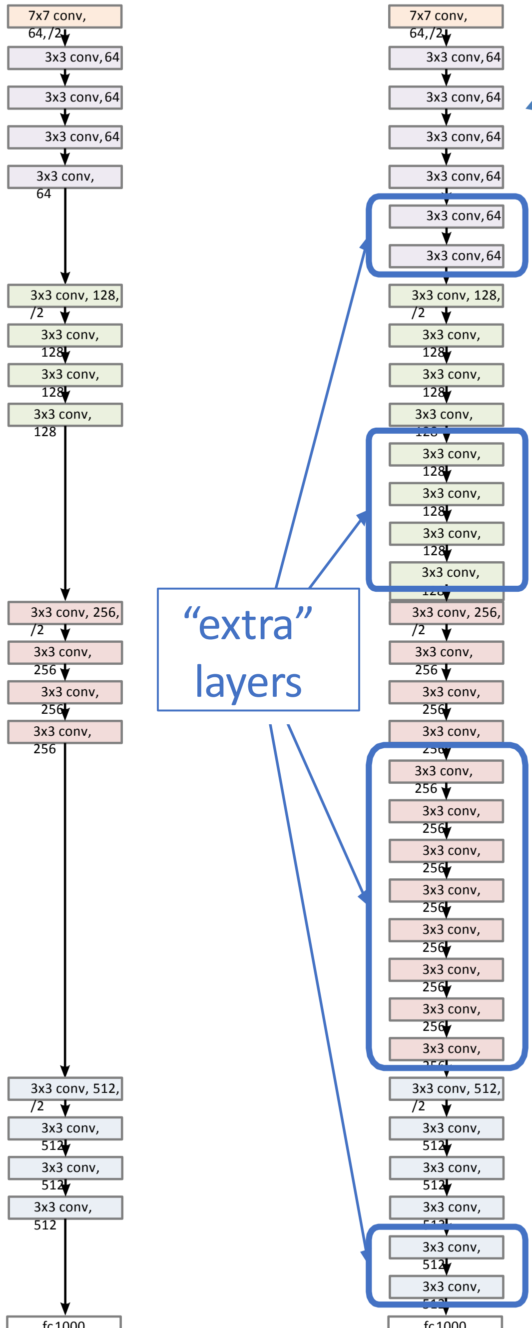
Motivation behind ResNet

Training deep nets

Solution by construction to the deeper model:

- Consider a shallower architecture and its deeper counterpart that adds more layers onto it.
- There exists a solution by construction to the deeper model: the added layers are identity mapping, and the other layers are copied from the learned shallower model.
- The existence of this constructed solution indicates that a deeper model should produce no higher training error than its shallower counterpart.

a shallower model (18 layers)



a deeper counterpart (34 layers)

- 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…

Motivation behind ResNet

Training deep nets

- But experiments show that our current solvers on hand are unable to find solutions that are comparably good or better than the constructed solution.
- The degradation problem suggests that the solvers might have difficulties in approximating identity mappings by multiple nonlinear layers.

Motivation behind ResNet

Referenced learning

- E.g.: classifying samples into k classes
- Unreferenced way: train a multi-class classifier based on samples' features
- Referenced way: k-mean clustering
- The k-means are the reference points w.r.t which the sample are classified

Motivation behind ResNet

Referenced learning

- In image recognition, Vector of Locally Aggregated Descriptors (VLAD) is a representation that encodes by the residual vectors with respect to a dictionary
- It is a powerful shallow representations for image retrieval and classification

ResNet

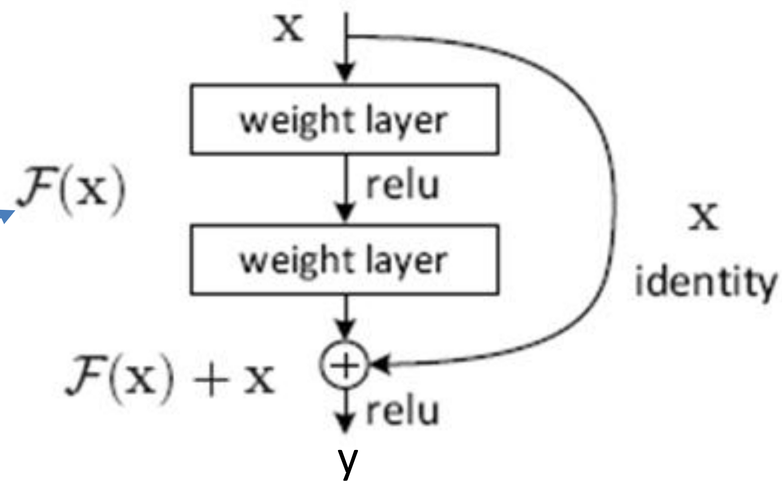
Residual learning

- Hypothesis: It is easier to optimize the residual mapping than to optimize the original, unreferenced mapping.
- Let us consider $H(x)$ as an underlying mapping to be fit
- Let a residual function $F(x) := H(x) - x$
- The original function thus becomes $F(x)+x$

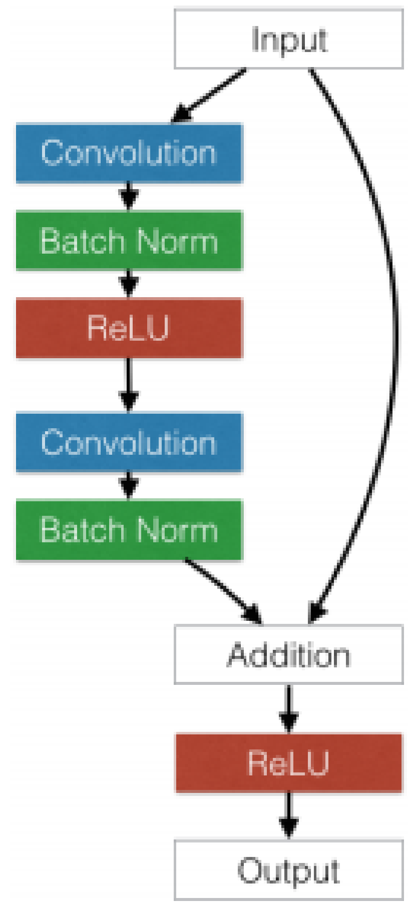
ResNet

Residual learning

Residual mapping to be learned



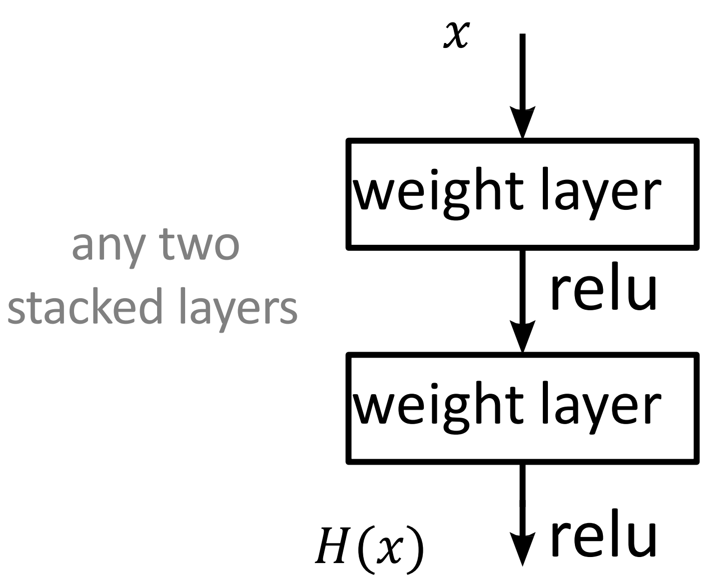
Residual learning: a building block



ResNet

Residual learning

- Plaint net



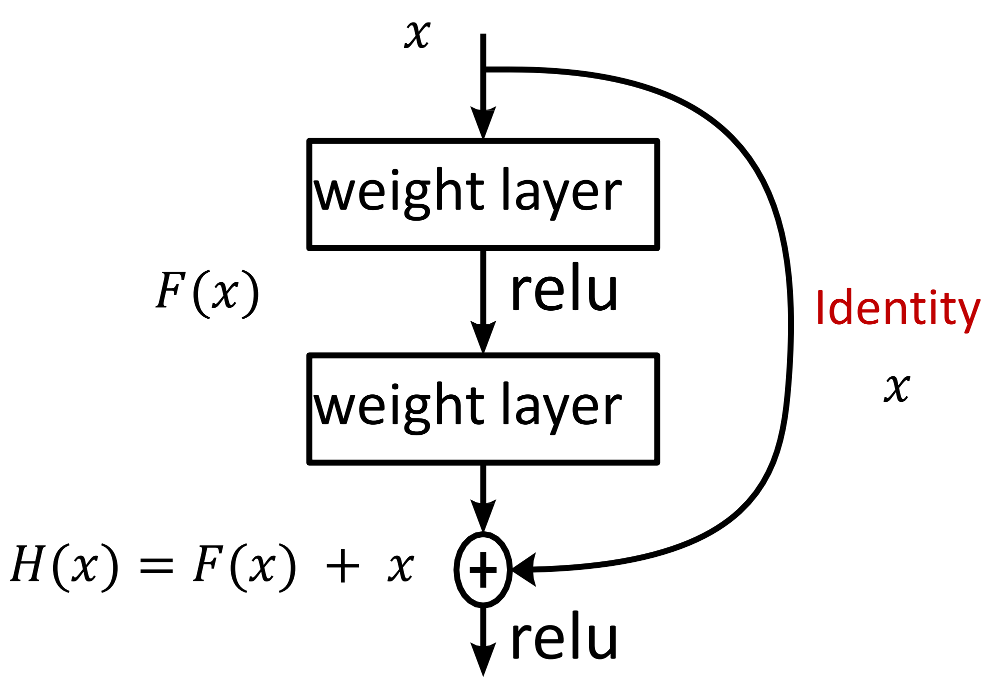
$H(x)$ is any desired mapping,

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

ResNet

Residual learning

- Residual net



$H(x)$ is any desired mapping,

~~Hope the two weight layers fit $H(x)$~~

Hope the two weight layers fit $F(x)$

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

Residual function:

$F(x) = H(x) - x$

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

ResNet

Residual learning

- Each subsequent layer is only responsible for, in effect, fine tuning the output from a previous layer by just adding a learned "residual" to the input.
- This differs from a more traditional approach where each layer has to generate the whole desired output

Ref: [2]

ResNet

Preconditioning

- With the residual learning reformulation, if identity mappings are optimal, the solvers may simply drive the weights of the multiple nonlinear layers toward zero to approach identity mappings.
- In real cases, it is unlikely that identity mappings are optimal, but the reformulation may help to precondition the problem.

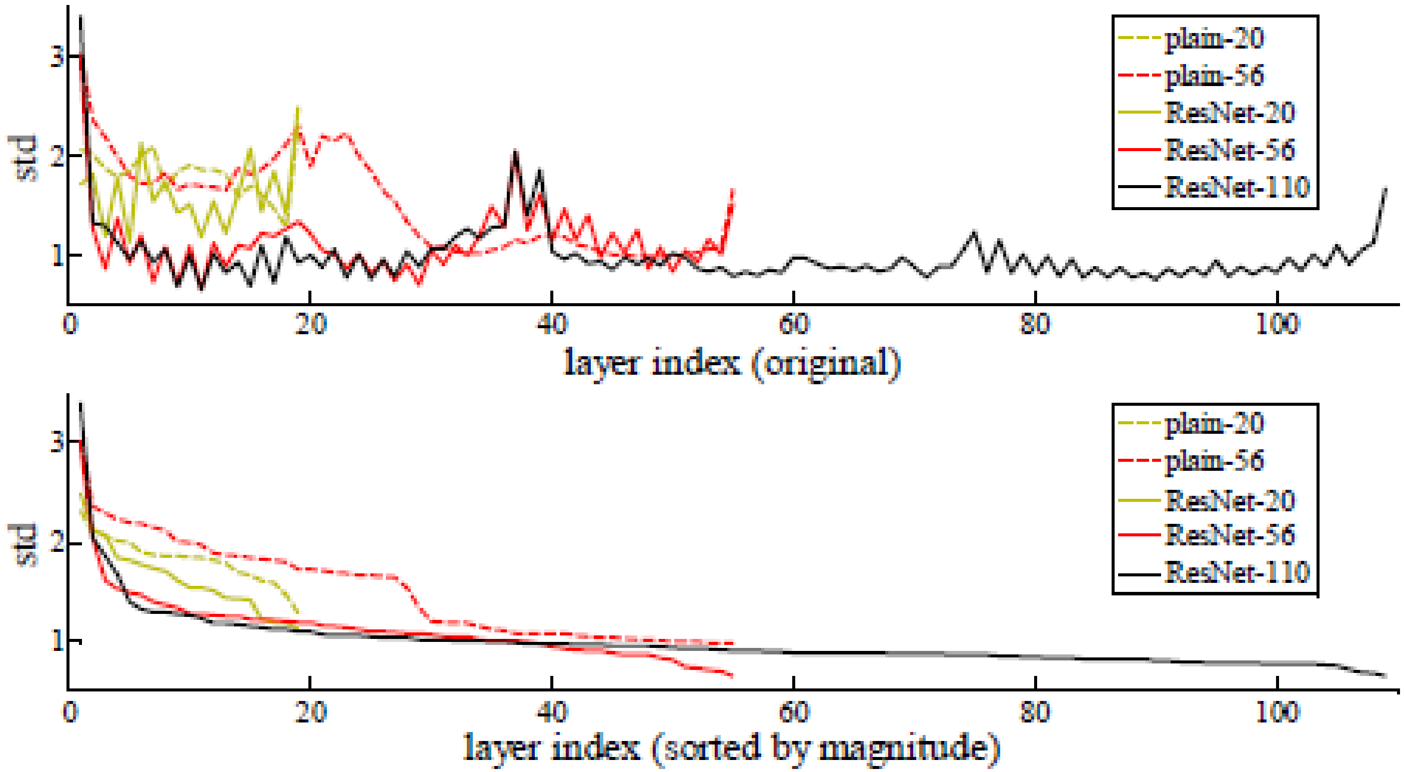
ResNet

Preconditioning

- If the optimal function is closer to an identity mapping than to a zero mapping, it should be easier for the solver to find the perturbations with reference to an identity mapping, than to learn the function as a new one.

ResNet

Preconditioning



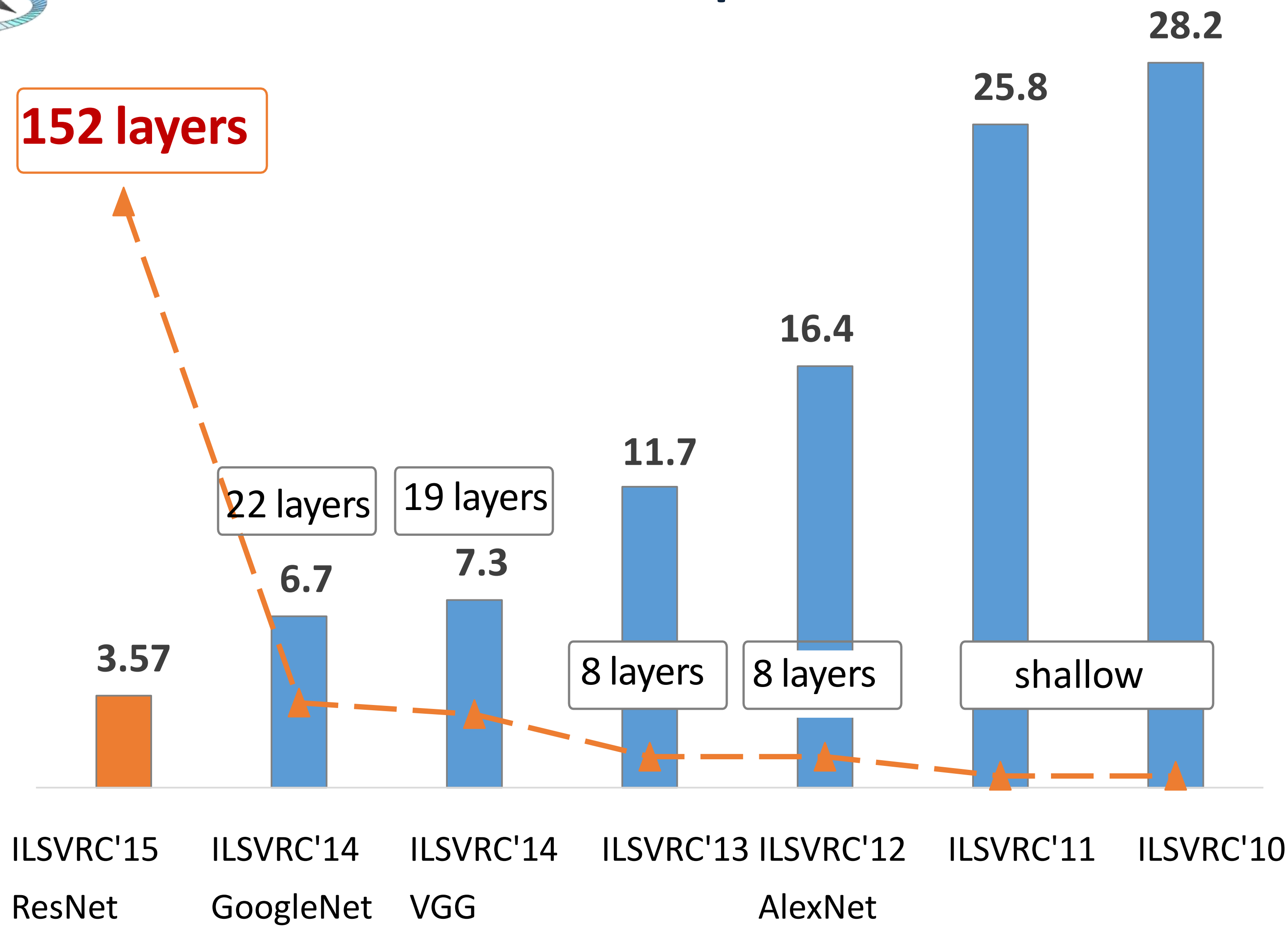
Standard deviations (std) of layer responses on CIFAR-10.
The responses are the outputs of each 3x3 layer.
Top: the layers are shown in their original order.
Bottom: the responses are ranked in descending order.

MSRA @ ILSVRC & COCO 2015 Competition

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

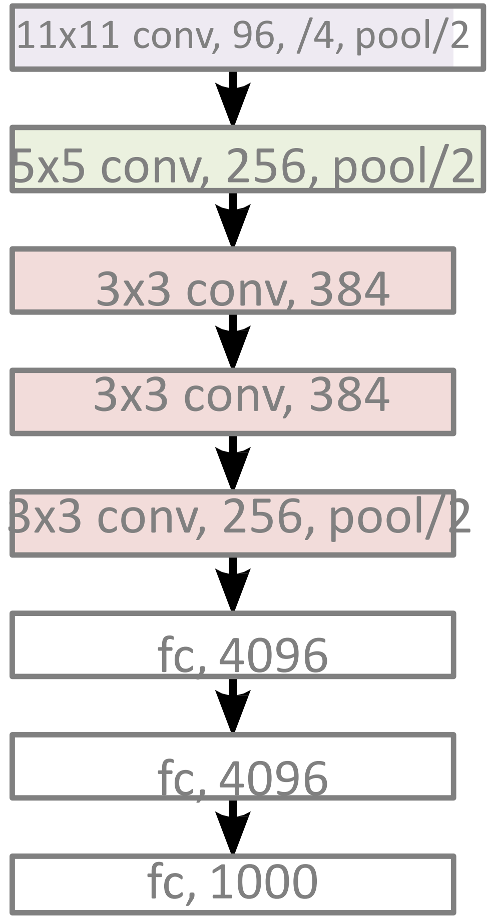
ILSVRC: Imagenet Large Scale Visual Recognition Challenge

Revolution of Depth



ImageNet Classification top-5 error (%)

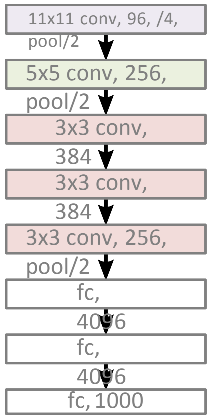
Revolution of Depth



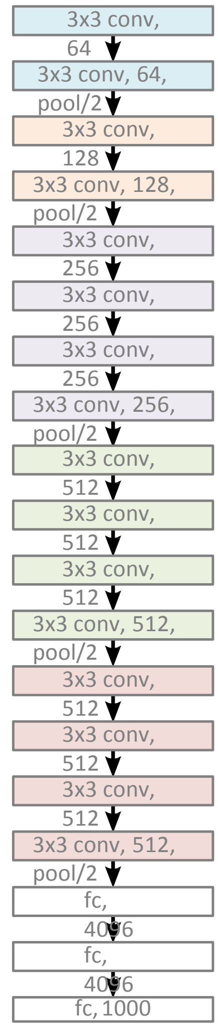
AlexNet, 8 layers
(ILSVRC 2012)

Revolution of Depth

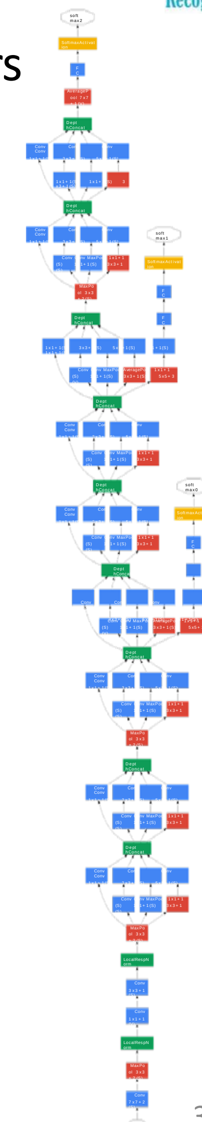
AlexNet, 8 layers
(ILSVRC 2012)



VGG, 19 layers
(ILSVRC 2014)



GoogleNet, 22 layers
(ILSVRC 2014)



ResNet, 152 layers
(ILSVRC 2015)

Experiments

ImageNet classification

- Model is evaluated on the ImageNet 2012 classification dataset that consists of 1000 classes.
- The models are trained on the 1.28 million training images, and evaluated on the 50k validation images.
- Obtain a final result on the 100k test images, reported by the test server.
- Evaluate both top-1 and top-5 error rates

Experiments

Plain vs Res nets

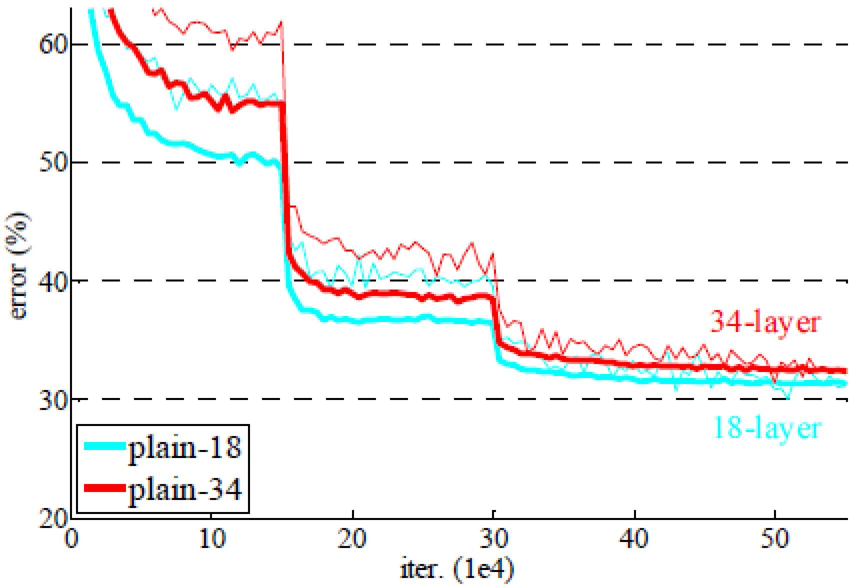
- The baseline architectures are the same as the above plain nets, expect that a shortcut connection is added to each pair of 3X3 filters
- When the net is "not overly deep" (18 layers here), the current SGD solver is still able to find good solutions to the plain net.
- In this case, the ResNet eases the optimization by providing faster convergence at the early stage

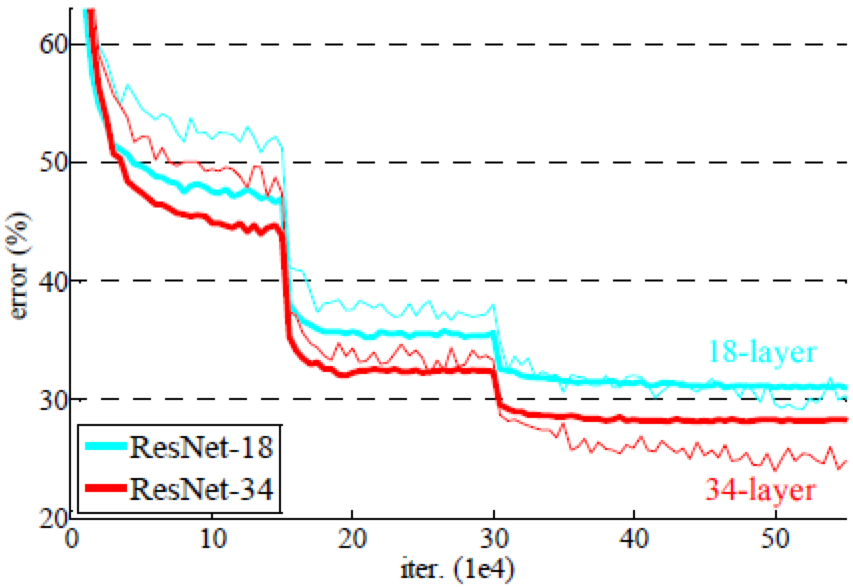
	plain	ResNet
18 layers	27.94	27.88
34 layers	28.54	**25.03**

Top-1 error (%) on ImageNet validation

Experiments

ImageNet classification





Thin curves denote training error, and bold curves denote validation error

Experiments

ImageNet classification

method	top-1 err.	top-5 err.
VGG [41] (ILSVRC'14)	-	8.43†
GoogLeNet [44] (ILSVRC'14)	-	7.89
VGG [41] (v5)	24.4	7.1
PReLU-net [13]	21.59	5.71
BN-inception [16]	21.99	5.81
ResNet-34 B	21.84	5.71
ResNet-34 C	21.53	5.60
ResNet-50	20.74	5.25
ResNet-101	19.87	4.60
ResNet-152	**19.38**	**4.49**

Error rates (%) of single-model results on the ImageNet validation set (except the 1st, reported on the test set).

Experiments

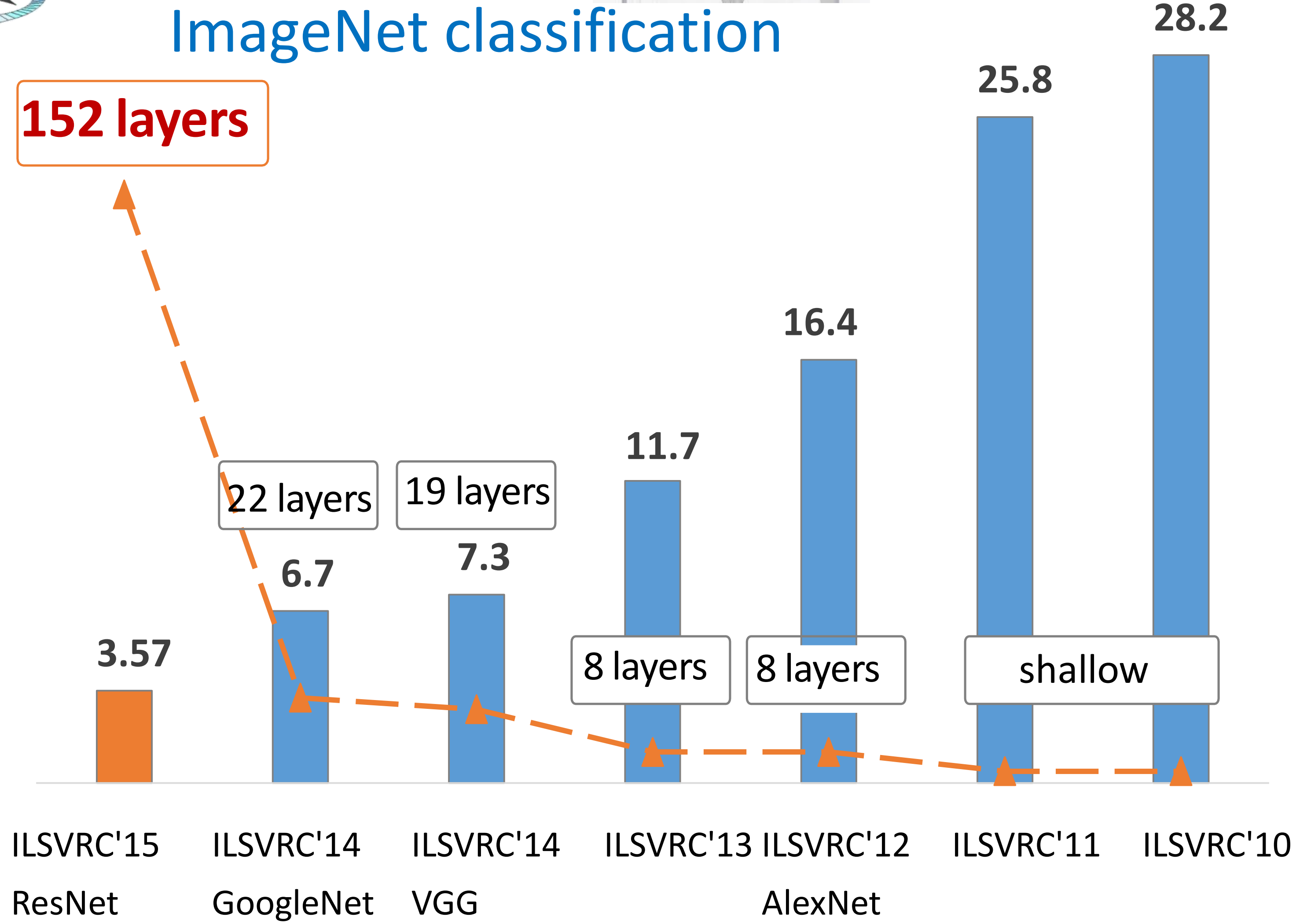
ImageNet classification

method	top-5 err. (test)
VGG [41] (ILSVRC'14)	7.32
GoogLeNet [44] (ILSVRC'14)	6.66
VGG [41] (v5)	6.8
PReLU-net [13]	4.94
BN-inception [16]	4.82
ResNet (ILSVRC'15)	**3.57**

Error rates (%) of ensembles. The top-5 error is on the test set of ImageNet and reported by the test server.

Experiments

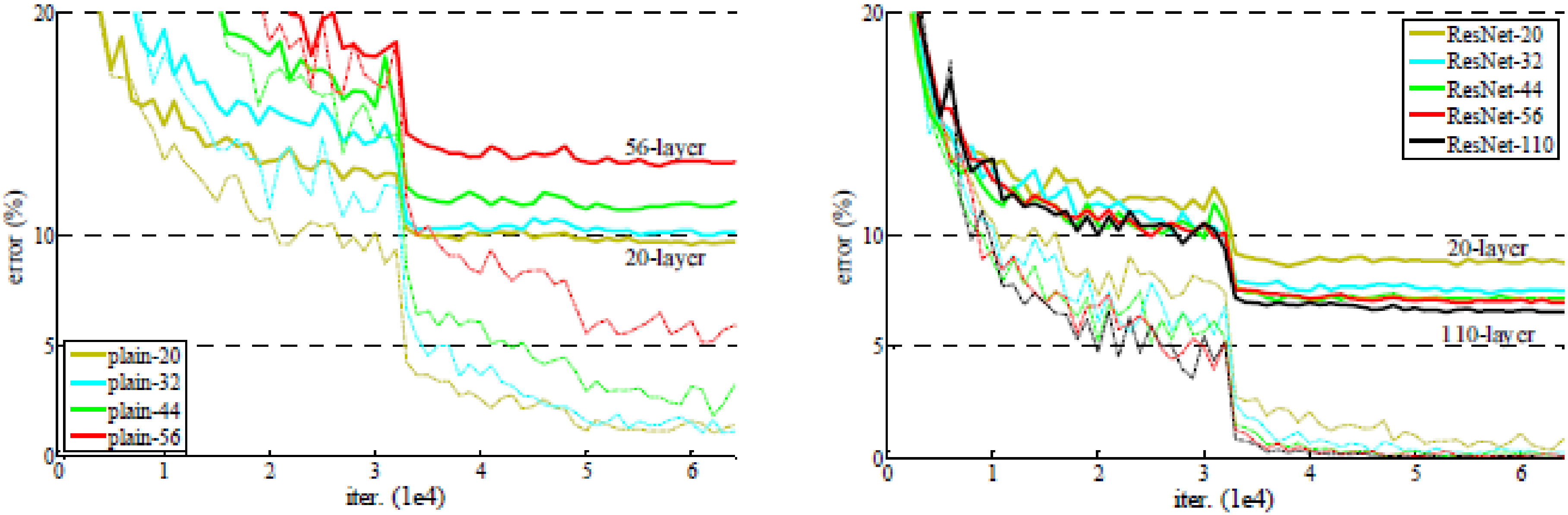
ImageNet classification



ImageNet Classification top-5 error (%)

Experiments

CIFAR-10

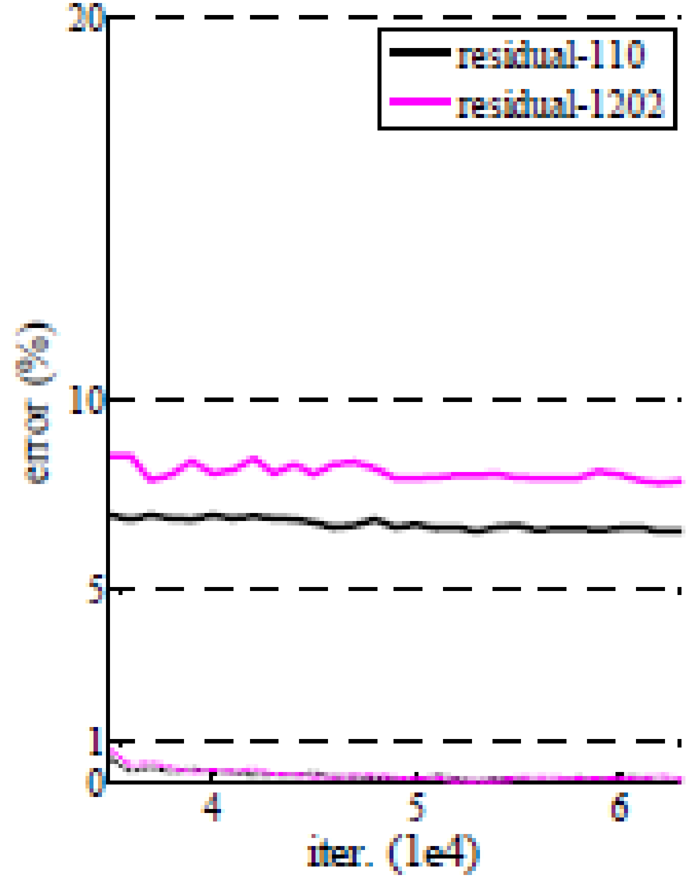


Training on CIFAR-10. Dashed lines denote training error, and bold lines denote testing error

Experiments

CIFAR-10

- But there are still open problems on such aggressively deep models. ResN-1202 have shown effects of overfitting due to overkill.



Experiments

CIFAR-10

method	# layers	# params	error (%)
Maxout [10]			9.38
NIN [25]			8.81
DSN [24]			8.22
FitNet [35]	19	2.5M	8.39
Highway [42, 43]	19	2.3M	7.54 (7.72±0.16)
Highway [42, 43]	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

Classification error on the CIFAR-10 test set

References

- [1] Kaiming He, Xiangyu Zhang, Shaoqing Ren, & Jian Sun. “Deep Residual Learning for Image Recognition”.
- [2] https://www.quora.com/How-does-deep-residual-learning-work

Questions?

Thank you!

- Thank you all for coming.
- Thanks also goes to Mr. Sajjad Jamil (MPhil student) for helping in slides preparation.