UNIVERSITY OF SCIENCE & TECHNOLOGY

Korea Institute of Science and Technology Information

2021 딥러닝기술 및 응용 - Paper Review

# Deep Residual Learning for Image Recognition (CVPR 2016)

**Kaiming He, Xiangyu Zhang, Shaoqing Ren, Jian Sun**
**(Microsoft Research)**

KISTI-UST
Gunho Lee

Some slides are borrowed from https://github.com/ndb796

2021.04.30.FRI.

# DEEP NETWORK의 문제점

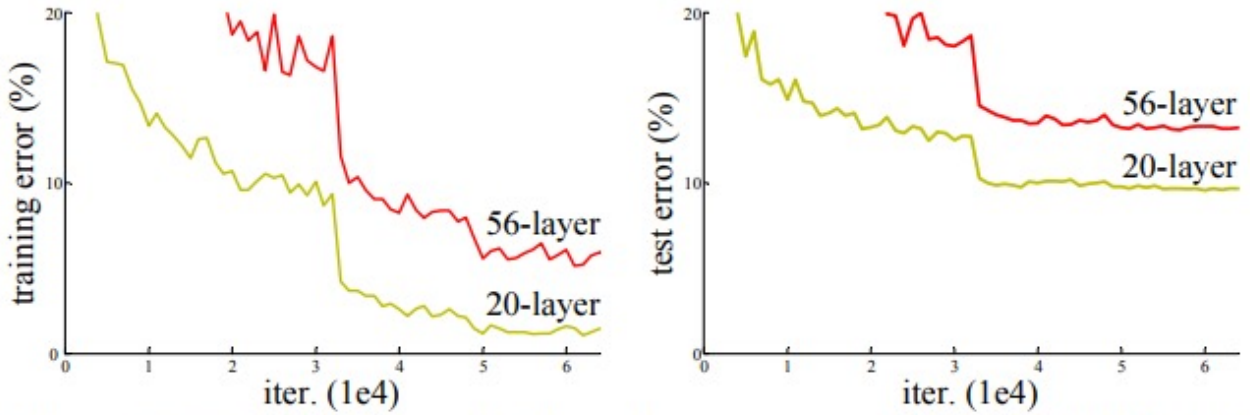



Figure 1. Training error (left) and test error (right) on CIFAR-10 with 20-layer and 56-layer "plain" networks. The deeper network has higher training error, and thus test error. Similar phenomena on ImageNet is presented in Fig. 4.

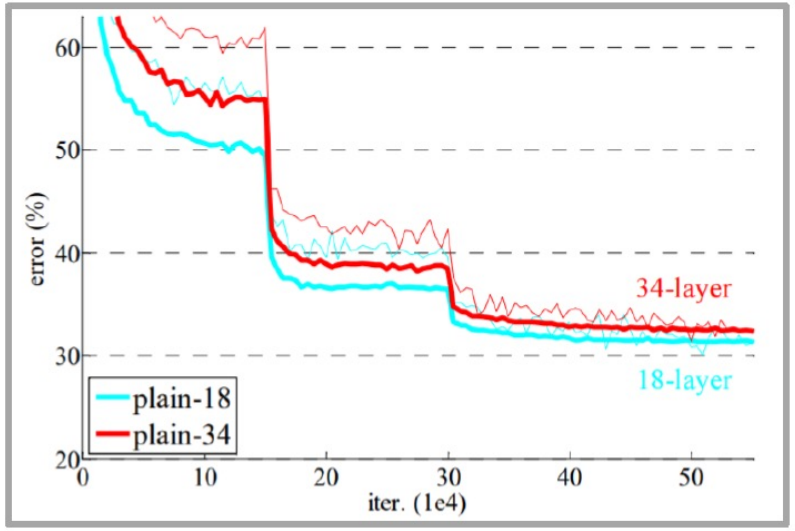
- 깊은 뉴럴 네트워크는 degradation 문제때문에 학습하기 어렵다는 단점이 있다.

- 그림을 확인해보면 기본적인 컨벌루션 뉴럴 네트워크에서 레이어 수만 늘리는 것은 트레이닝과 테스트 에러를 증가시킨다.

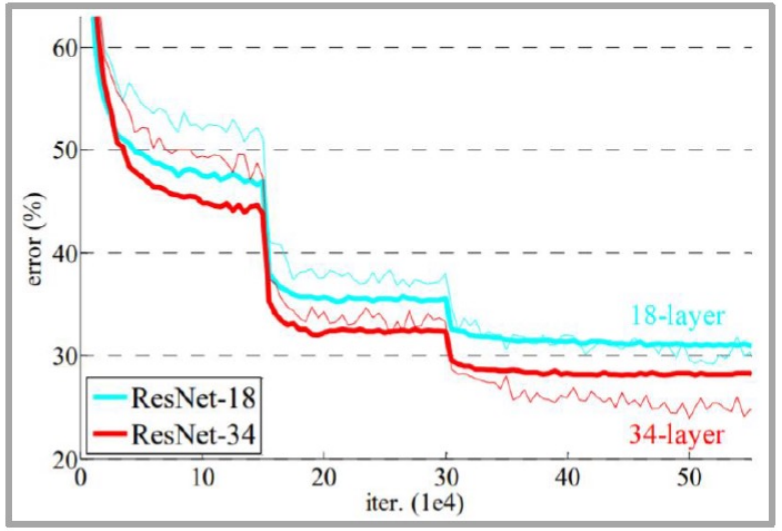
# DEEP RESIDUAL LEARNING FOR IMAGE RECOGNITION

- 본 논문에서는 **깊은 네트워크**를 학습시키기 위한 방법으로 **잔여 학습**(residual learning)을 제안합니다.

< ImageNet top-1 training error >



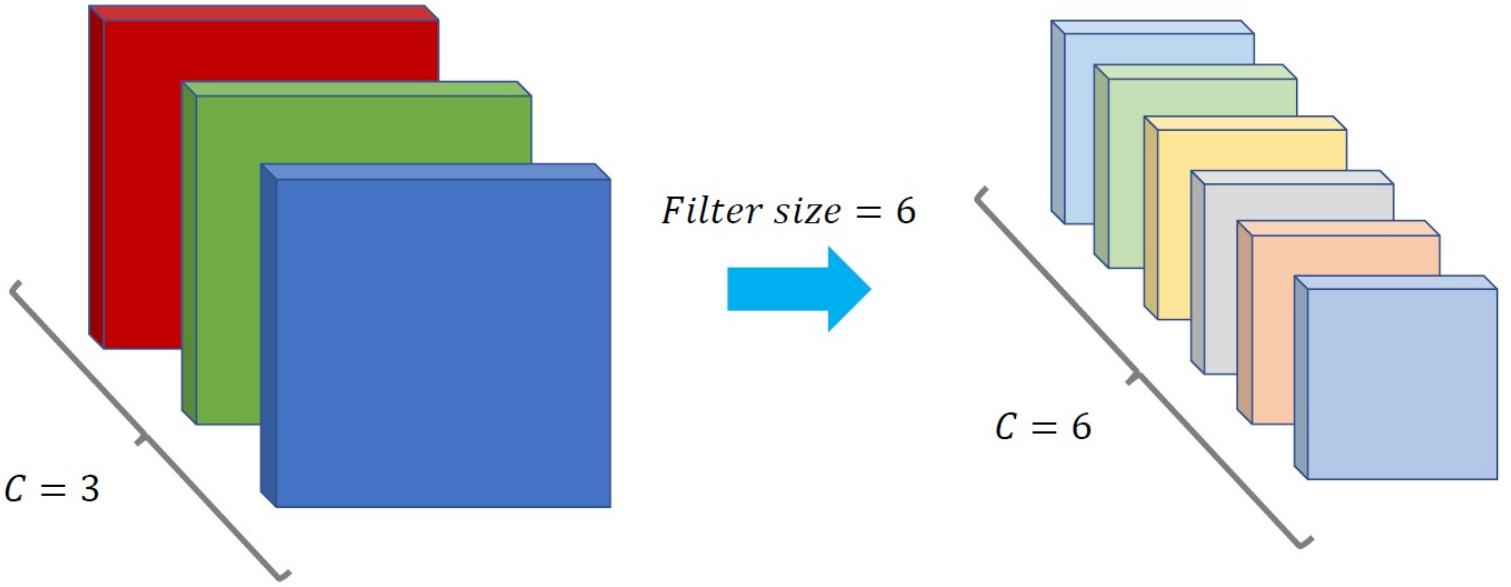

일반적인 CNN




잔여 학습을 적용한 CNN

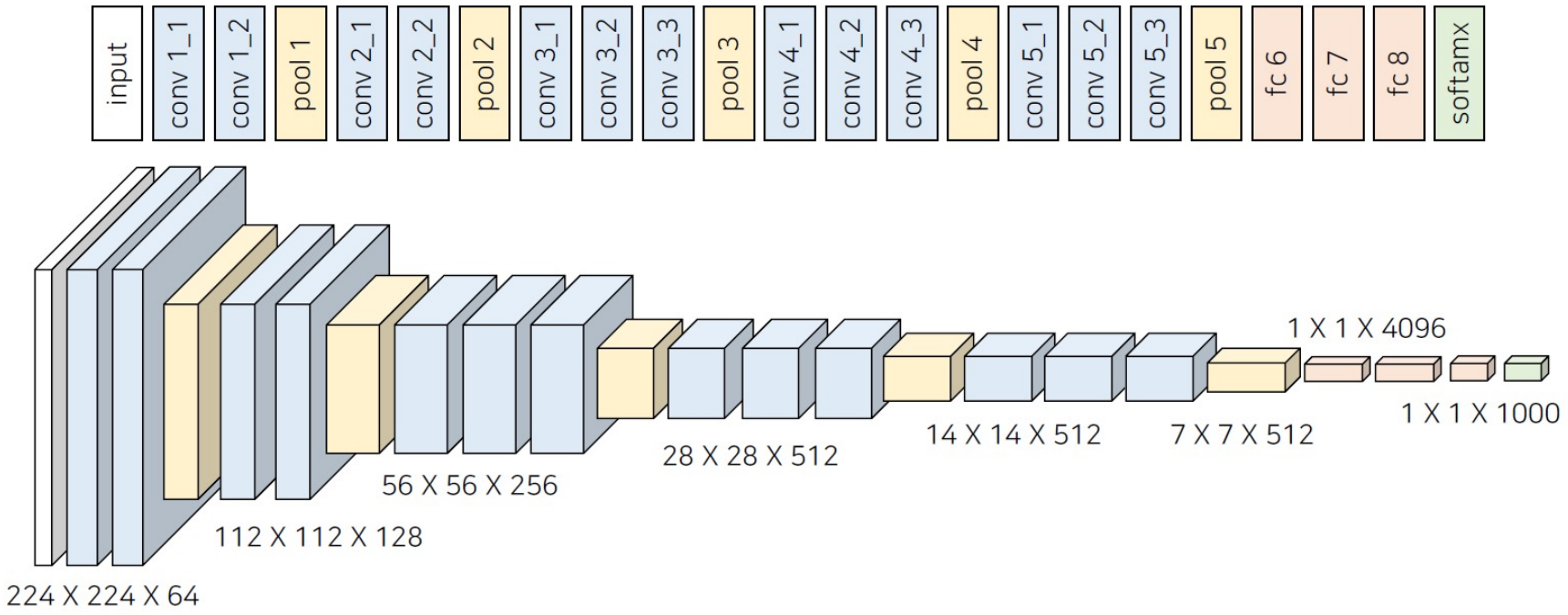
# CNN모델의 특징 맵 (FEATURE MAP)

- 일반적으로 CNN에서 레이어가 깊어질수록 채널의 수가 많아지고 너비와 높이는 줄어듭니다.
- 컨볼루션 레이어의 **서로 다른 필터들은** 각각 적절한 **특징(feature)** 값을 추출하도록 학습됩니다.



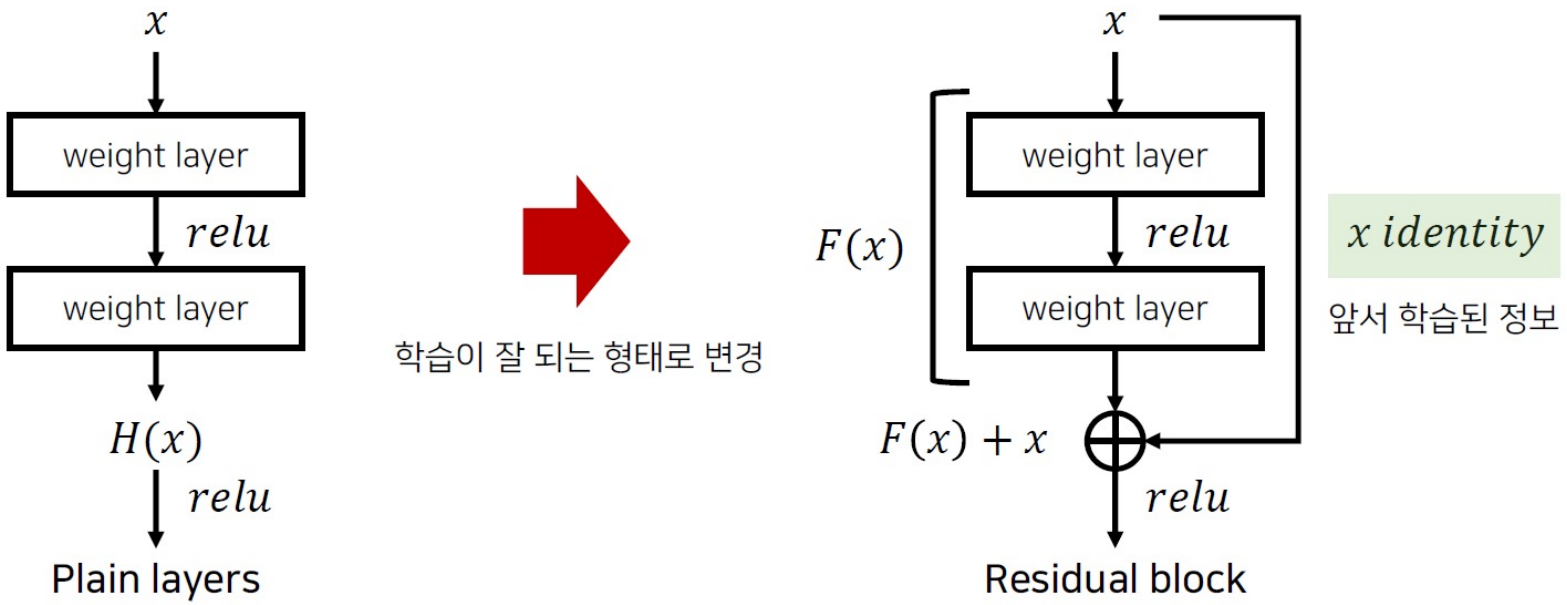
# VERY DEEP CONVOLUTIONAL NETWORKS FOR LARGE SCALE IMAGE RECOGNITION (ICLR 2015)

- **VGG 네트워크**는 작은 크기의 **3x3 컨볼루션 필터(filter)**를 이용해 **레이어의 깊이**를 늘려 우수한 성능을 보입니다.



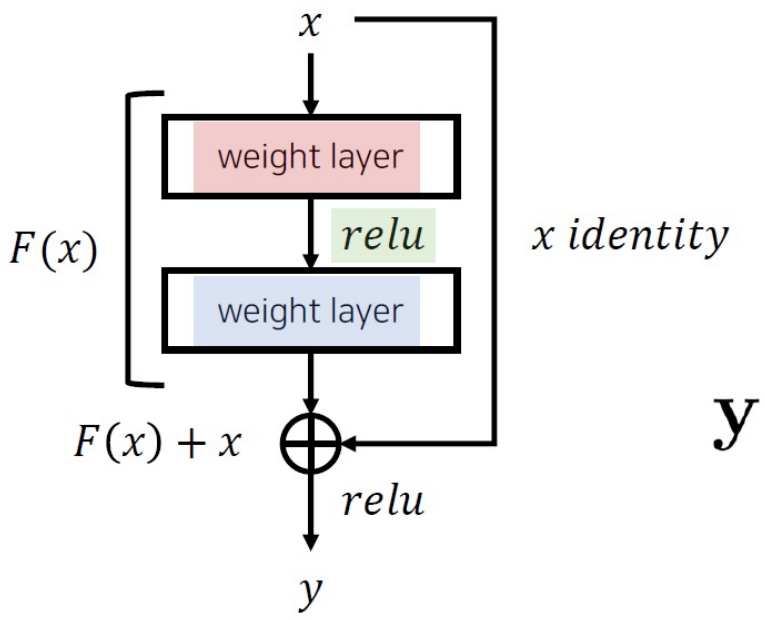
# 본논문의 핵심 아이디어 : 잔여 블록 (RESIDUAL BLOCK)

- **잔여 블록(residual block)**을 이용해 네트워크의 **최적화(optimization) 난이도**를 낮춥니다.
  - 실제로 내재한 mapping인 $H(x)$를 곧바로 학습하는 것은 어려우므로 대신 $F(x) = H(x) - x$를 학습합니다.



# 본논문의 핵심 아이디어 : 잔여 블록 (RESIDUAL BLOCK)

- **잔여 블록(residual block)**을 이용해 네트워크의 **최적화(optimization) 난이도**를 낮춥니다.




$$\mathcal{F} = W_2\sigma(W_1\mathbf{x})$$

일반적인 형태

$$\mathbf{y} = \underbrace{\mathcal{F}(\mathbf{x}, \{W_i\})}_{\text{multiple convolutional layers}} + \underbrace{W_s\mathbf{x}}_{\text{shortcut}}$$

# IMAGENET 2012 CLASSIFICATION DATASET

- 이미지넷(ImageNet)은 대표적인 대규모(large-scale) 데이터셋
- 데이터셋은 1,000개의 클래스로 구성되며 총 백만 개가 넘는 데이터를 포함
- 약 120만 개는 학습(training)에 쓰고, 5만개는 검증(validation)에 쓰임

# IMAGENET에서의 테스트 결과 분석

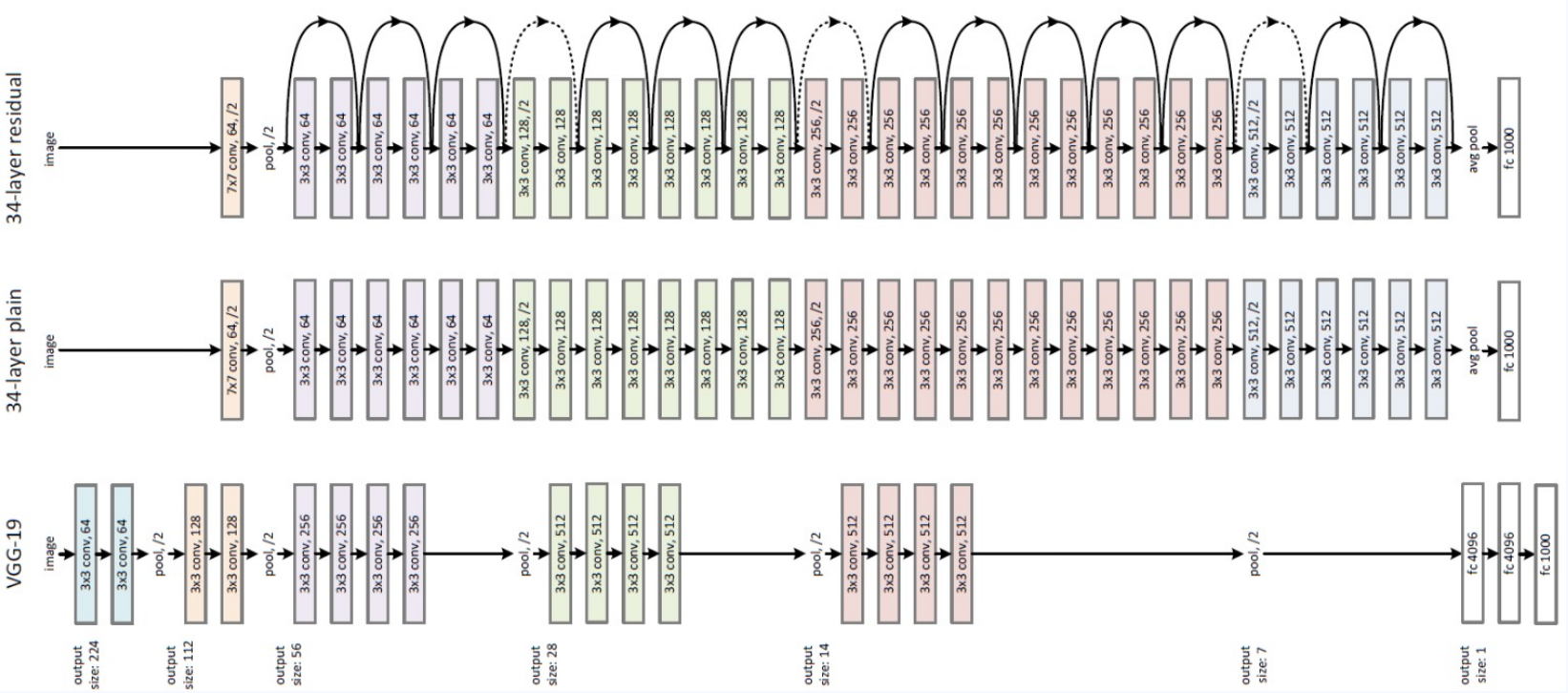



Figure 3. Example network architectures for ImageNet. Left: the VGG-19 model [40] (19.6 billion FLOPs) as a reference. Middle: a plain network with 34 parameter layers (3.6 billion FLOPs). Right: a residual network with 34 parameter layers (3.6 billion FLOPs). The dotted shortcuts increase dimensions. Table 1 shows more details and other variants.

# IMAGENET에서의 테스트 결과 분석

plain 네트워크에서 깊은 네트워크를 쌓는 것은 오히려 얕은 네트워크보다 에러율이 높아짐.

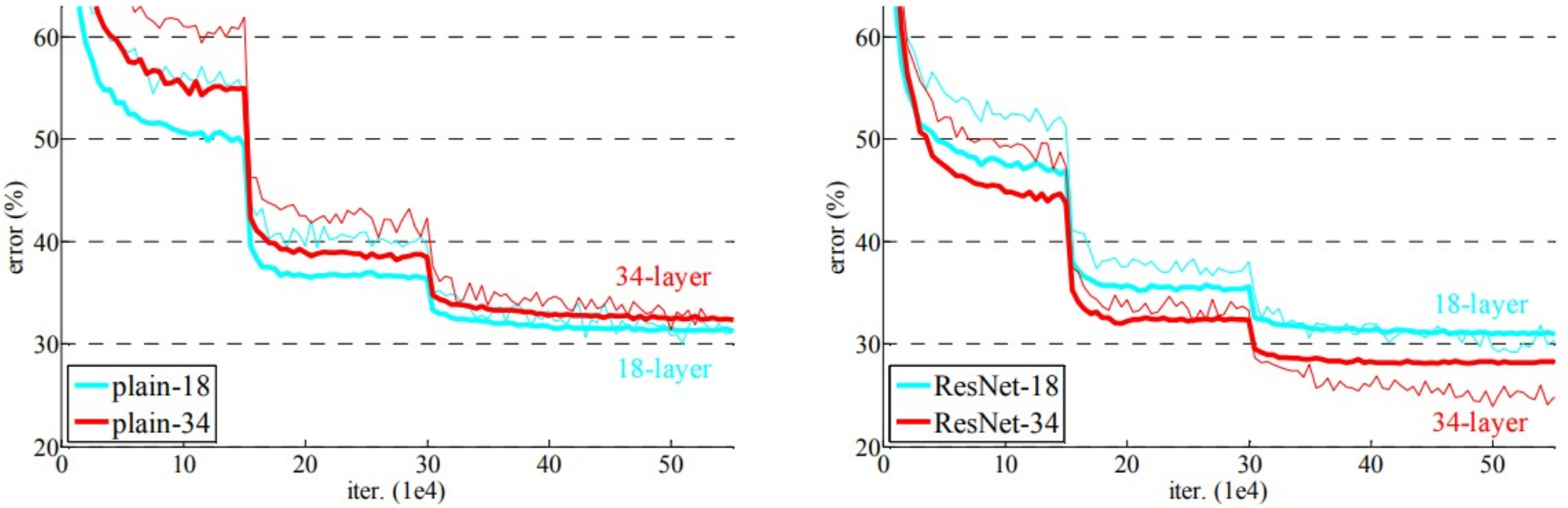



Figure 4. Training on **ImageNet**. Thin curves denote training error, and bold curves denote validation error of the center crops. Left: plain networks of 18 and 34 layers. Right: ResNets of 18 and 34 layers. In this plot, the residual networks have no extra parameter compared to their plain counterparts.

# IMAGENET에서의 테스트 결과 분석

- Plain 네트워크와  비교했을때 달라진것은 shortcut connection이 더해진거 밖에 없음에도 불구하고 훨씬 성능이 개선됨.
- Resnet은 깊은 레이어가 얕은 레이어보다 트레이닝 에러도 줄고 일반화성능도 높다.

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

Table 2. Top-1 error (%, 10-crop testing) on ImageNet validation. Here the ResNets have no extra parameter compared to their plain counterparts. Fig. 4 shows the training procedures.

# IMAGENET에서의 테스트 결과 분석

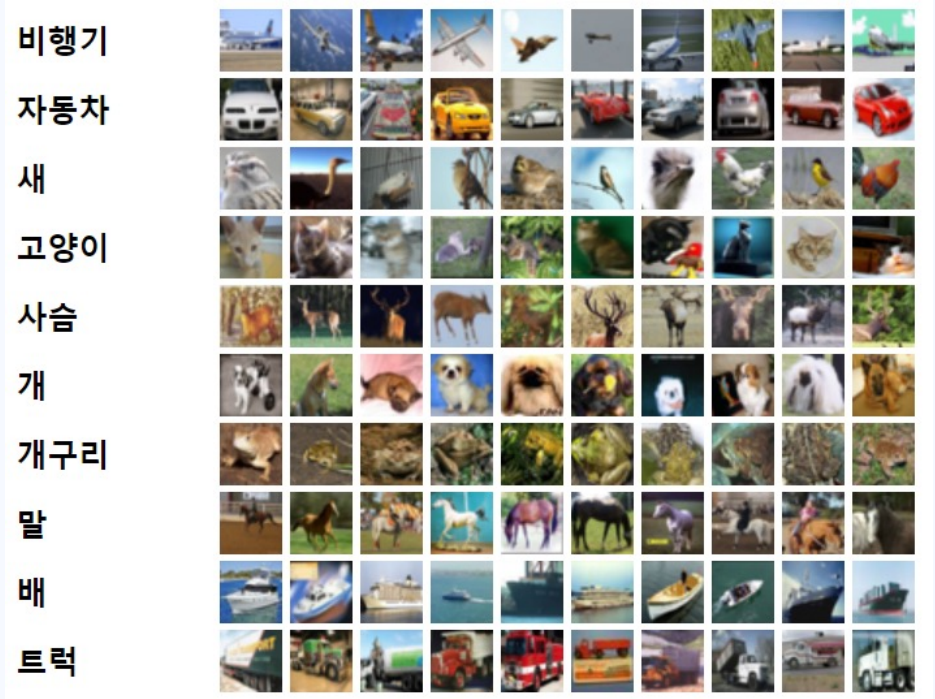
| model | top-1 err. | top-5 err. |
|---|---|---|
| VGG-16 [40] | 28.07 | 9.33 |
| GoogLeNet [43] | - | 9.15 |
| PReLU-net [12] | 24.27 | 7.38 |
| plain-34 | 28.54 | 10.02 |
| ResNet-34 A | 25.03 | 7.76 |
| ResNet-34 B | 24.52 | 7.46 |
| ResNet-34 C | 24.19 | 7.40 |
| ResNet-50 | 22.85 | 6.71 |
| ResNet-101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

Table 3. Error rates (%, **10-crop** testing) on ImageNet validation. VGG-16 is based on our test. ResNet-50/101/152 are of option B that only uses projections for increasing dimensions.

Shortcut connection을 위해서 identity mapping을 사용할지 projection을 사용할지 성능차이를 보여줌.

(A) zero-padding shortcuts are used for increasing dimensions, and all shortcuts are parameter free
(B) projection shortcuts are used for increasing dimensions, and other shortcuts are identity
(C) all shortcuts are projections

# CIFAR-10 DATASET




- 10개의 클래스로 구분된 32 x 32 사물 사진을 모은 데이터셋 (학습용: 50,000개, 테스트용: 10,000개)

# CIFAR-10에서의 테스트 결과 분석

| 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 |

Table 6. Classification error on the **CIFAR-10** test set. All methods are with data augmentation. For ResNet-110, we run it 5 times and show "best (mean±std)" as in [42].

- 이미지넷과 비교했을때 아키텍처가 다르긴 하지만 유사한 형태를 가지고 있음.
- 기존의 다른 네트워크와 비교했을때 파라미터수는 적지만 성능은 좋음.
- 이미지넷과 마찬가지로 레이어가 깊어질수록 성능 좋아짐.
- 레이어가 너무 불필요하게 깊어지면 오히려 성능이 떨어짐. 작은데이터셋에 너무 깊은 레이어면 오버피팅발생.

# OBJECT DETECTION

| training data | 07+12 | 07++12 |
|---|---|---|
| test data | VOC 07 test | VOC 12 test |
| VGG-16 | 73.2 | 70.4 |
| ResNet-101 | **76.4** | **73.8** |

Table 7. Object detection mAP (%) on the PASCAL VOC 2007/2012 test sets using **baseline** Faster R-CNN. See also appendix for better results.

| metric | mAP@.5 | mAP@[.5, .95] |
|---|---|---|
| VGG-16 | 41.5 | 21.2 |
| ResNet-101 | **48.4** | **27.2** |

Table 8. Object detection mAP (%) on the COCO validation set using **baseline** Faster R-CNN. See also appendix for better results.

- Mean average precision을 비교한 결과, VGG 모델보다 성능이 향상된 것을 볼 수 있음.
- 이미지 분류 뿐만 아니라 object detection분야에도 좋은 성능을 보임.

# SUMMARY

- 깊은 뉴럴네트워크는 학습하기 어렵다는 단점이 있다.
- Residual Learning을 사용하여 훨씬 깊은 네트워크를 학습할 수 있다.
- VGG 네트워크에 비해서 더 깊지만 복잡더는 더 낮으며 성능은 개선되었다.
- 이미지 분류 뿐만 아니라 object detection이나 Semantic segmentation 분야에도 좋은 성능을 보인다.

# REFERENCE


- 1. https://github.com/ndb796/Deep-Learning-Paper-Review-and-Practice
- 2. https://gruuuuu.github.io/machine-learning/cifar10-cnn/#

# Thank You