

손글씨 심층 CNN 구현(VGG)

그림 8-1 손글씨 숫자를 인식하는 심층 CNN



합성곱 계층의 필터 : 모두 3×3

층이 깊어질 수록 채널 수 증가 : 16-16-32-32-64-64

풀링으로 공간 크기 점차 줄임

마지막 완전 연결 계층에선 Dropout 계층 사용

Optimizer = Adam

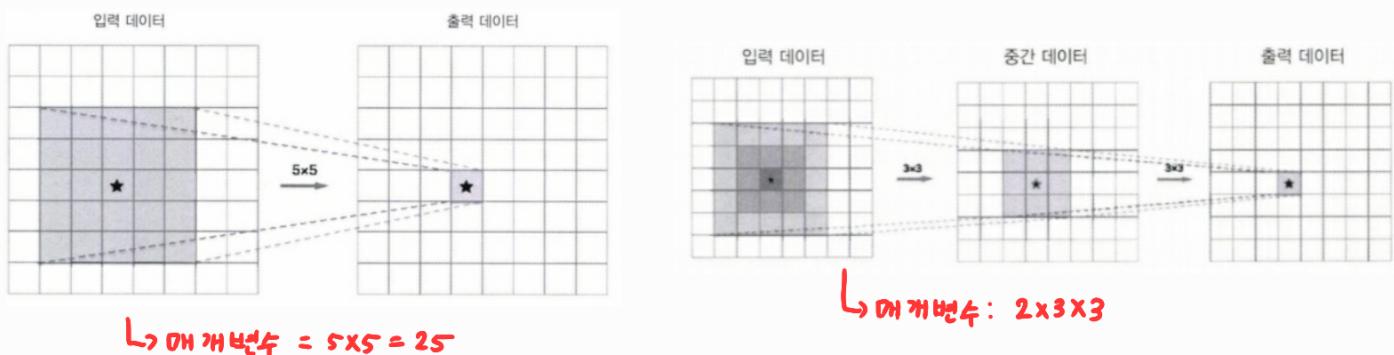
init_weights = 'He' 초기값

정확도를 높이기 위한 기법

- 양상을 학습, 학습률 감소, 데이터 확장 등
- 데이터 확장 : 훈련 이미지를 이용하여 인위적으로 확장
 - ex. Crop : 이미지 일부 잘라냄, Flip : 좌우를 뒤집음

층을 깊게 하는 이유

1. 매개변수 수를 줄일 수 있음!



작은 필터를 걸쳐 신경망을 깊게 할 때 장점

⇒ 매개변수 수를 줄여 넓은 수용영역 소화 가능

↳ 누군에 변화를 일으키는 국소적인 공간 영역

⇒ 흥을 거둘하여 활성화 합수를 합성을 사이에 끼워 신경망 표현력을 개선

⇒ 선형성↑ → 높은 표현도 가능

2. 학습의 효율성을 높일 수 있다.

-> 문제를 계층적으로 분해 가능(처음 층에서 옛지의 패턴을 학습)

-> 풀기 쉬운 문제로 분해 가능

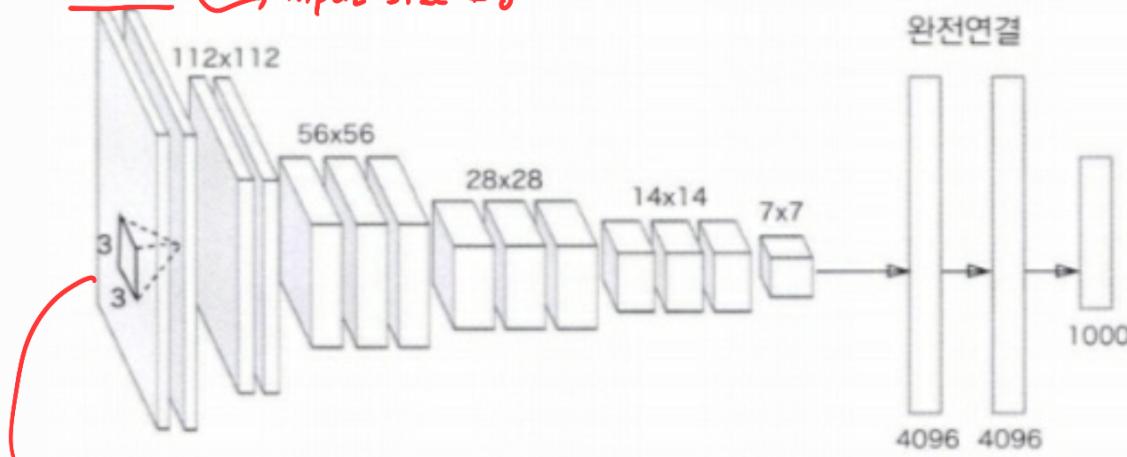
VGGNet

합성곱과 풀링 계층으로 구성되는 기본적인 CNN

-> Convolution Layer와 FC(Fully Connected) Layer를 16 또는 19 층으로 심화한 특징(층의 깊이에 따라 VGG16, VGG19로 구분)

그림 8-9 VGG^[22]

224x224 → input size 고정

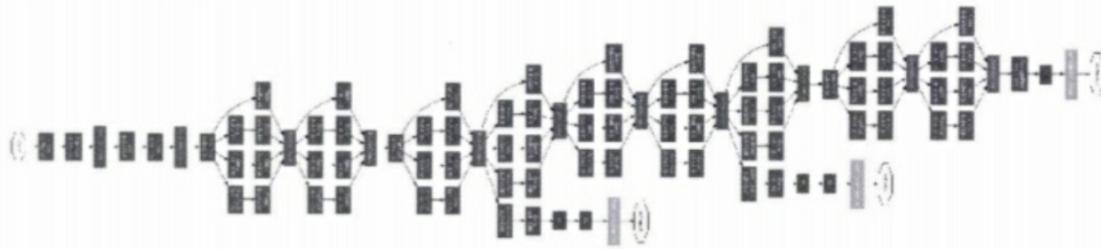


→ 3x3의 작은 필터 사용하여 합성곱 계층을 연속적으로 거친

2~4회 Pooling 계층 거쳐서 크기 절반으로 줄임

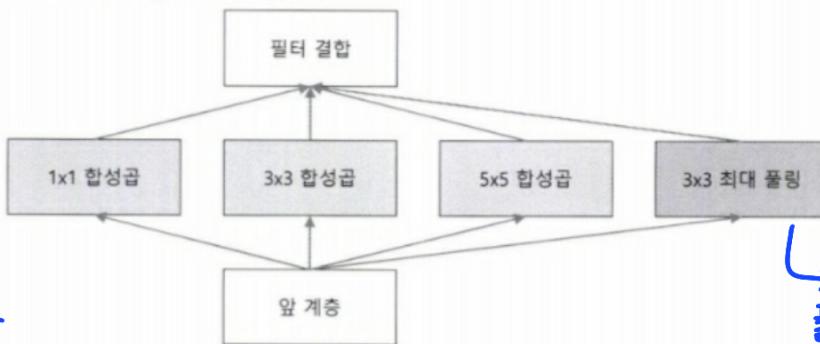
GoogLeNet

그림 8-10 GoogLeNet^[23]



세로 뿐만 아니라 가로 방향도 있다. 가로 방향에 '꽃'이 있는데 이를 인셉션 구조라고 함

그림 8-11 GoogLeNet의 인셉션 구조^[23]

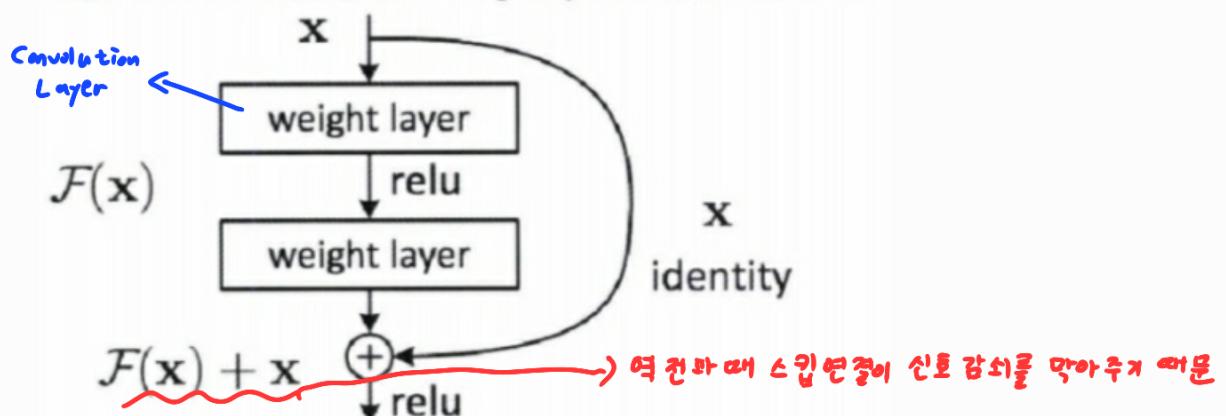


이 인셉션 구조를 하나의 블록(블록(구성요소))로 사용

ResNet(Residual Network)

- 스킵 연결(skip connection) 도입 : 층이 지나치게 깊어 학습이 잘 되지 않고 오히려 성능을 떨어지는 것을 방지하기 위해
=> 층의 깊이에 비례해 성능을 향상시킬 수 있다.

그림 8-12 ResNet의 구성요소^[24] : 'weight layer'는 합성곱 계층을 말한다.



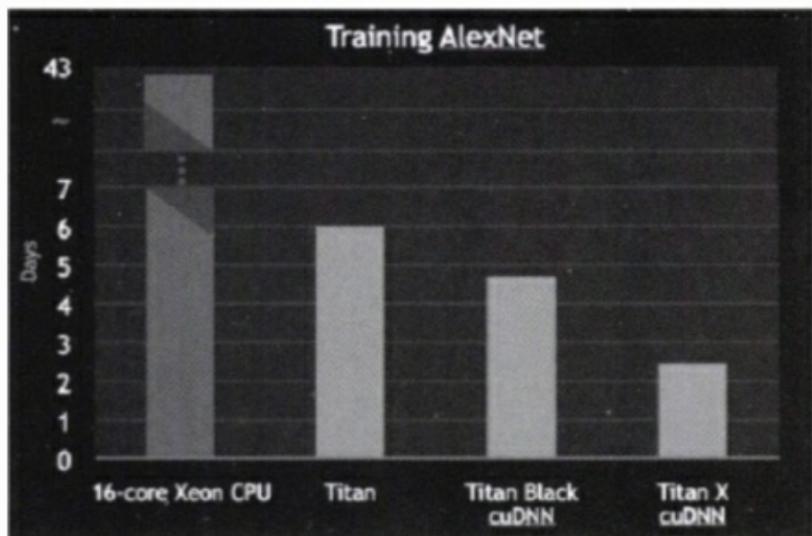
- skip connection : 입력 데이터를 합성곱 계층을 건너 뛰어 출력에 바로 더하는 구조
- VGG 기반 + 스킵 연결 = 층을 깊게 만듦
- 2개 층마다 건너뛰며 층을 깊게 함

전이학습

- 학습된 가중치를 다른 신경망에 복사한 다음 그 상태로 재학습을 수행
- 미리 학습된 가중치를 초기값으로 설정한 후 새로운 데이터셋을 대상으로 재학습(fine Tuning) 수행

GPU(Graphics Processing Unit)을 이용한 딥러닝 고속화

- GPU 장점 : 병렬 수치 연산을 고속으로 가능하게 함 = GPU Computing



⇒ CUDNN을 이용하여
더욱 빠른 학습속도 구현

분산 학습

- : 딥러닝 학습을 수평 확장 하자는 아이디어
- : 다수의 GPU 를 활용하여 고속화

연산 정밀도와 비트 줄이기

- : OOM(Out of Memory)를 방지하기 위해 데이터의 비트를 최소화해야됨
- : 딥러닝은 16비트 까지 사용해도 무방.(보통 32비트로 줄임)