NN의 역사

신경망

1957년 Frank Rosenblatt가 Mark I Perceptron machine을 개발

- 최초의 퍼셉트론 기계

1960년 Widrow와 Hoff가 Adaline and Madaline 개발

- 최초의 Multilayer Perceptron Network

1986년 Rumelhart

- 역전파 제안, 신경망 학습 시작

2006년 Geoff Hinton과 Ruslan Salakhutdinow

- DNN 학습가능성

2012년 Hintin lab

- 음성인식 NN 성능 좋음

- ImageNet Classification에서 최초로 NN 사용, 결과 좋았다 -> Alexnet

CNN

1950년 Hubel과 Wiesel의 뉴런 연구

- topographical mapping

- 뉴런의 계층구조

- Simple cells -> Complex cells -> hypercomplex cells

- 빛의 방향 -> 움직임 -> 끝 점

1980년 neocognitron

- simple/complex cell 사용한 최초의 NN

1998년 Yann LeCun

- NN 학습을 위한 역전파와 gradient-based learning

2012년 Alex Krizhevsky

- CNN의 유행

CNN으로 할 수 있는 것

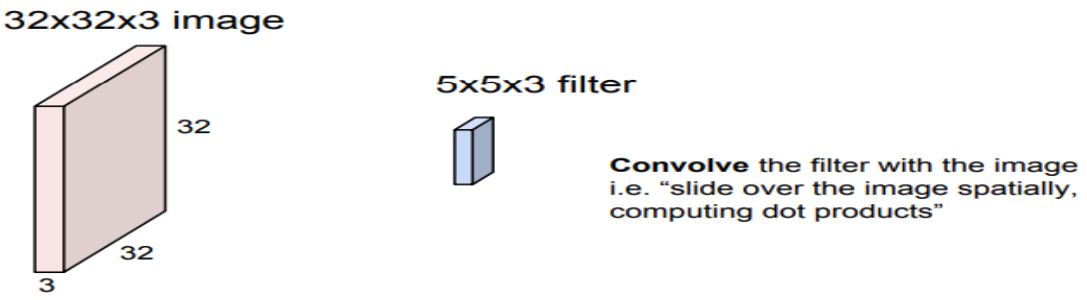
- 이미지 분류, Detection, Segmentation, 자율주행, 얼굴인식 사람추정, 자세 인식, 의학 진단, image captioning, 화풍 변경

CNN의 원리

Filter

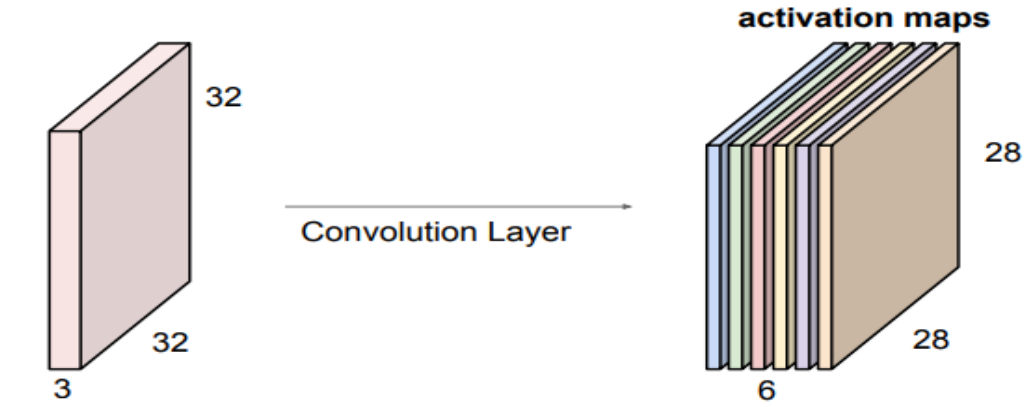
Fully Connected Layer : 벡터를 펴서 내적 연산을 하는 방식

Convolutional Layer : 기존의 structure을 보존하며 계산. 하늘색 필터가 이미지 내를 슬라이딩 하며 공간적인 내적을 함. 모든 depth에 대해 내적이 진행되어야 하므로, 필터의 depth는 input의 depth와 항상 같다.

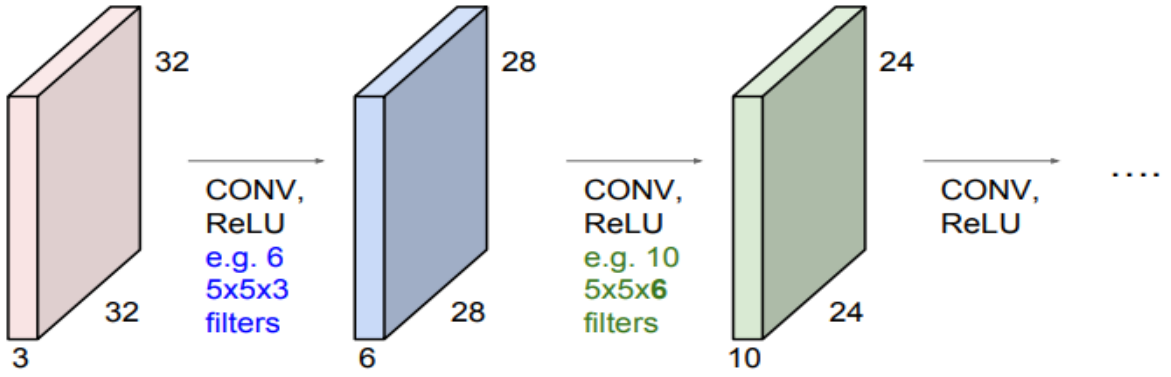


필터가 슬라이딩을 하면서 output을 뽑아내는 방법: 겹쳐놓고 내적하고, 슬라이딩해서 옆에서 계속 내적해서 output activation map의 해당 위치에 전달

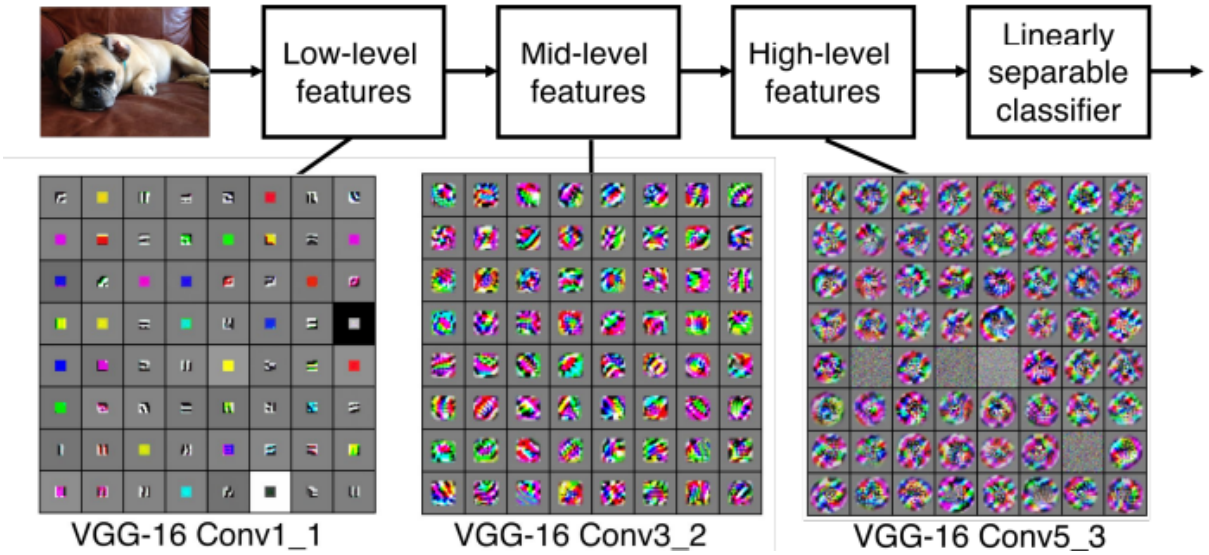
Activation map의 차원 : 입력은 32x32, 출력은 28x28. 원하는 만큼 필터를 사용할 수 있음.



사이사이에 activation, pooling 등이 들어감. layer는 여러개의 필터를 가지고 있고, 각 필터마다 각각의 출력 map을 만듦. 여러 layer들을 거치면서 각 필터들이 계층적으로 학습 가능.

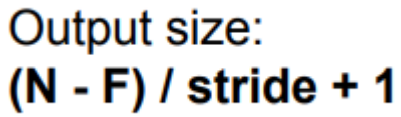


여러개의 convolution layer를 거치면서 단순한 구조에서 더 복잡한 구조로 찾아감. 각 grid는 하나의 뉴런(filter).



CNN은 input 이미지는 여러 layer들을 거치게 되고, 마지막에는 FC layer를 통해 score를 계산.

필터를 몇칸씩 움직일지를 stride로 정할 수 있음. 보통 input 사이즈와 슬라이딩 시 딱 맞아떨어지는 stride만을 이용.



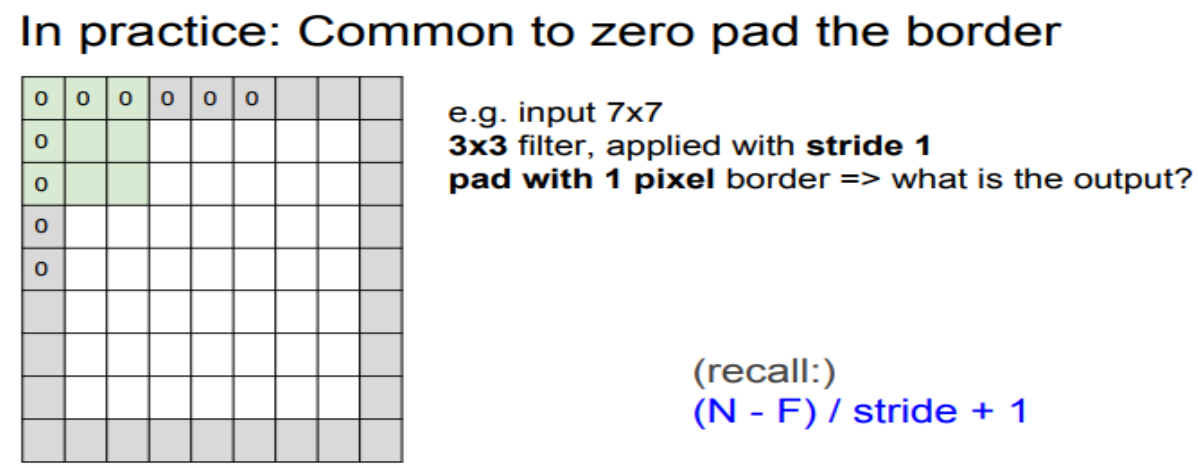
입력의 차원:N

필터 사이즈: F

stride를 설정해 줌으로써 pooling과 같이 다운샘플링할 수 있고, 더 좋은 성능을 가져다주기도 함. 이는 activation map의 사이즈를 줄이는 것이고, 나중에 FC layer의 파라미터의 수가 줄어들게 된다.

---

zero-padding

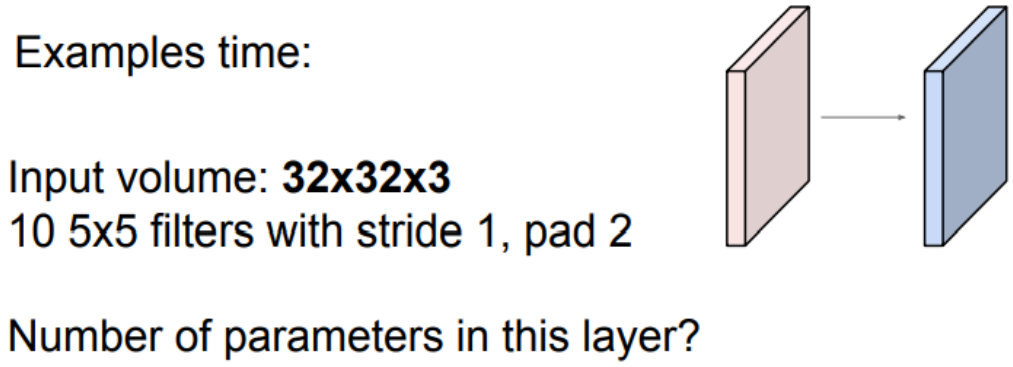


코너의 값들이 적게 연산되는 것을 막아주고, 레이어들을 거치면서 입력의 사이즈가 줄어드는 것을 막아준다. 깊은 네트워크에서는 Activation map이 엄청나게 작아지게 되고, => 정보 잃음. 항상 원본 이미지를 표현하기에 충분한 차원을 사용해야 한다.

위의 사진에서 출력은 7x7x(필터의 개수). 각 필터가 입력의 모든 depth에 대해 내적을 수행한다.

---

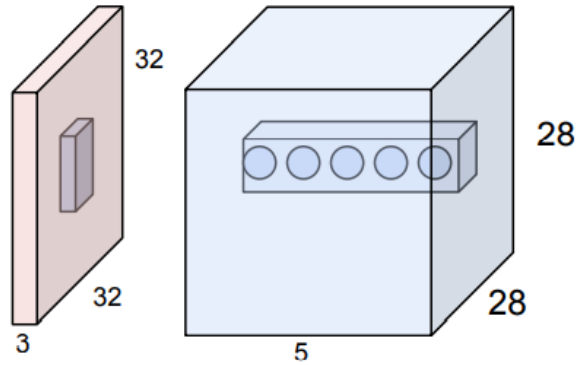
ex)



-> 필터당 5x5x3+1(bias)개의 파라미터 존재, 총 760개의 파라미터가 존재한다.

5x5 필터가 있다면 한 뉴런의 Receptive field가 5x5라고 한다. Receptive field란 한 뉴런이 한 번에 수용할 수 있는 영역을 의미한다.

만약 총 5개의 필터를 아래와 같이 거쳤다면, 한 점에서 depth 방향으로 바라보면, 이 5개는 정확하게 같은 지역에서 추출된 서로 다른 특징이다. 즉, 공간적 의미를 그대로 가져갈 수 있다.



Pooling, ReLU

CNN에 들어가는 다른 Layer

Pooling Layer

- Representation들을 더 작고 관리하게 쉽게 해줌

- DownSample

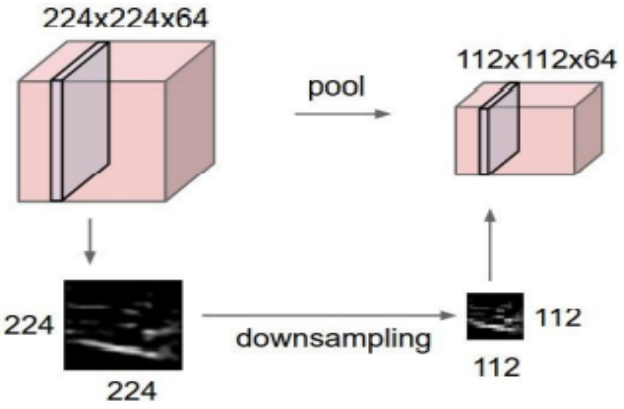
- 공간적인 invariance

- Depth는 그대로 둠

- 차원 계산은 (width-Filter)/Stride+1

- 보통 padding 안함(코너 값 계산 못하는 경우 없다.)

- 2x2, 3x3, stride=2 많이 씀

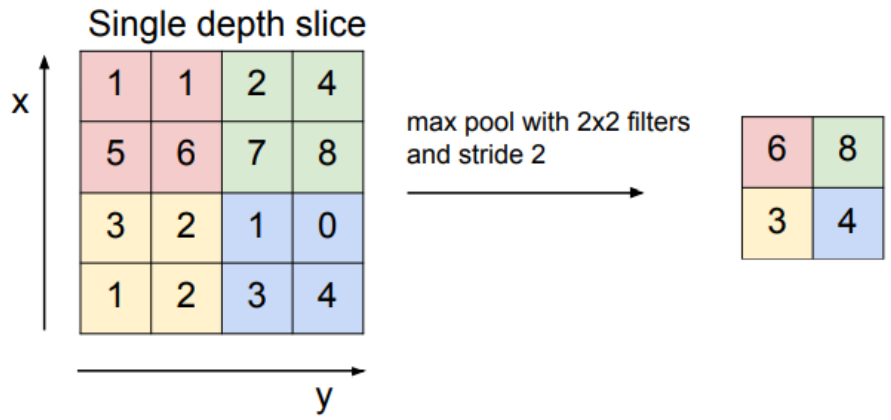


Max Pooling

- 필터 크기와 stride 정하면 됨

- 필터 안에 가장 큰 값 고름

- 겹치지 않게 풀링



ReLU Layer

- 실제 방식과 가장 유사한 비선형함수

- 활성화할지 비활성화 할지 결정

- 가장 많이 사용