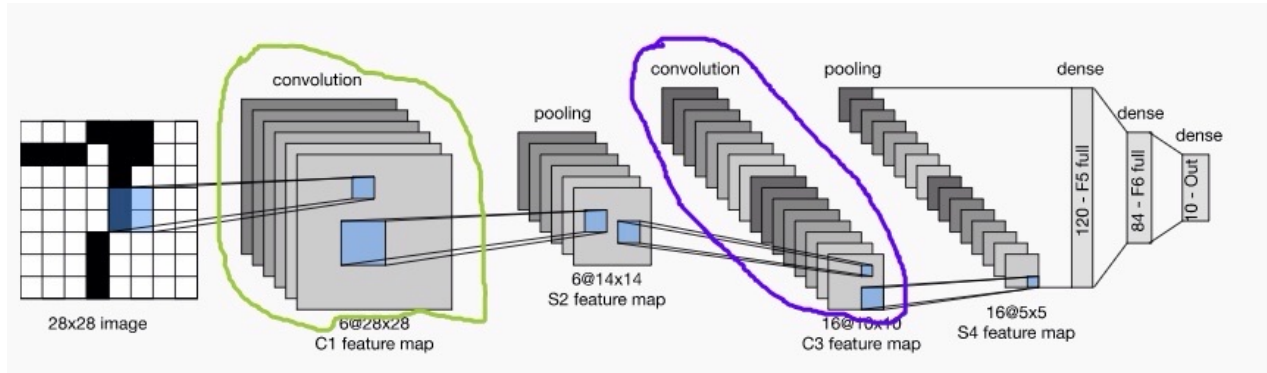


CNN

* Convolution Neural Network

* Convolution이란?



CNN구조

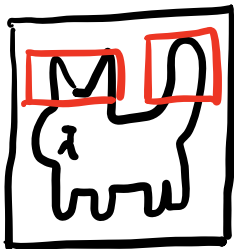
Convolution / sampling이 반복되자 fully connected layer가 적용되는 구조,

Convolution & subsampling

⇒ feature extraction

- 최전트 -

Ex>



feature extraction

⇒ 특징을 얻음!!



fully connected ⇒ 물체가 "어떤것"인가 판단,

CNN이 잘되는 이유.

① local invariance (구조적으로 차이가 없는 것)

→ loosely speaking.

convolution filter, sliding

② Compositimality

→ 이미지
input

conv & sub-sampling

관찰

→ 계층구조 쌓음

→ Compositionality

③ Conv 과정-1

Image

1 ^{x1}	1 ^{x0}	1 ^{x1}		
0 ^{x0}	1 ^{x1}	1 ^{x0}		
0 ^{x1}	0 ^{x0}	1 ^{x1}		

$$1 \times 1 + 1 \times 1 + 1 \times 1 \\ + 1 \times 1 + 0 \times 1 = \cancel{\times}$$

→

$\cancel{\times}$		

convolution
Feature

[3] Conv 과정-2

Conv결과 \rightarrow feature

해당픽셀에재입.

만약 유사하다면 response가 높지나올 것이다.

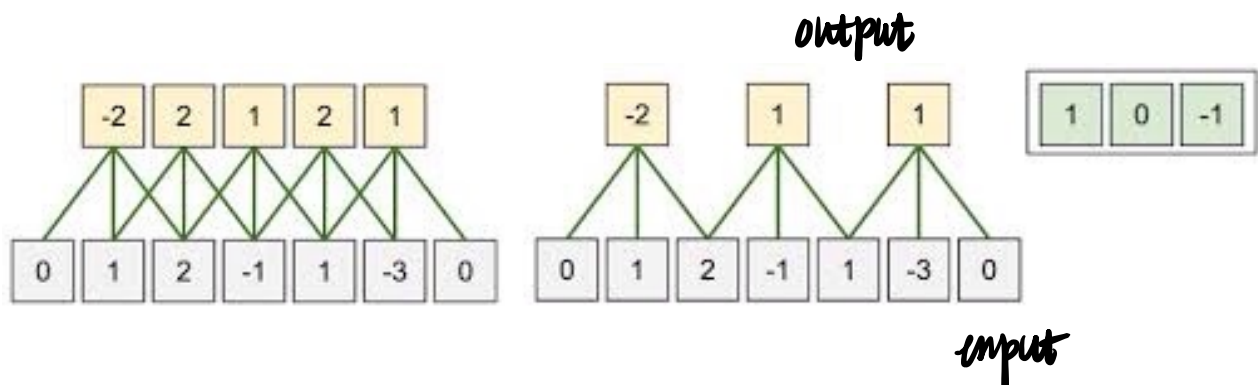
?! 우리가 학습시키는 것은 이 필터의 모양이다.

?! 동일한 이미지가 전체를 돌아다니며 conv한다는 개념이 중요,

\Rightarrow zero padding / stride / channel

개념은 확실히 알아두어야 한다.

[4] zero-padding-1



④ zero-padding-2

conv를 가장자리에서 할 경우

→ 2에서 적용한다고 하면 칸이 부족...

그래서 좌우 양을 늘리는 것.

질문!!...

input size :

output size :

filter size :

zero padding size :

filter가 5로 변환하면 zero padding → ?

⑤

$$N_{out} = N_{in} + N_{padding} + N_{filter} + 1.$$

⑤ stride

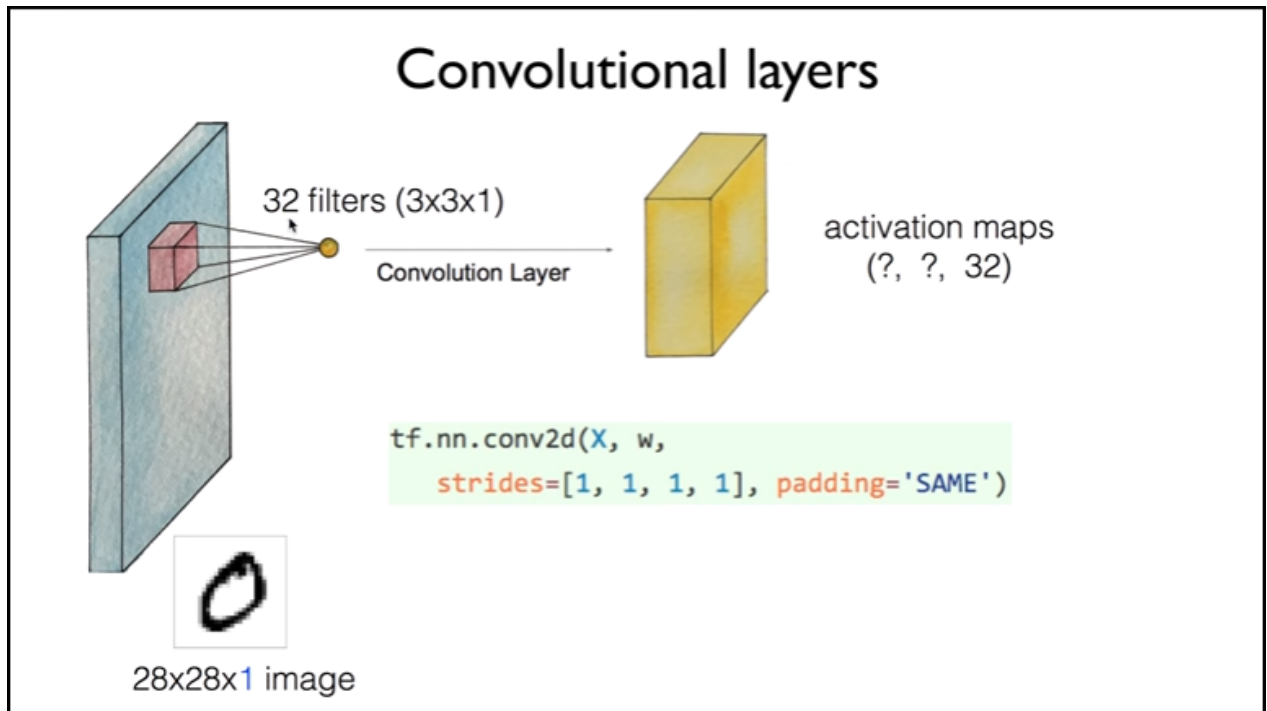
영어로 → 스텝 크기.

5x5가 있다고 하고 stride : 2라고 하자.

conv → $5/2 = 2.5$ 가 됨 2.3 → 3 ; output = 3

만약 stride size가 filter 사이즈와 동일하다면 overlapping ×

16) 코드로 확인

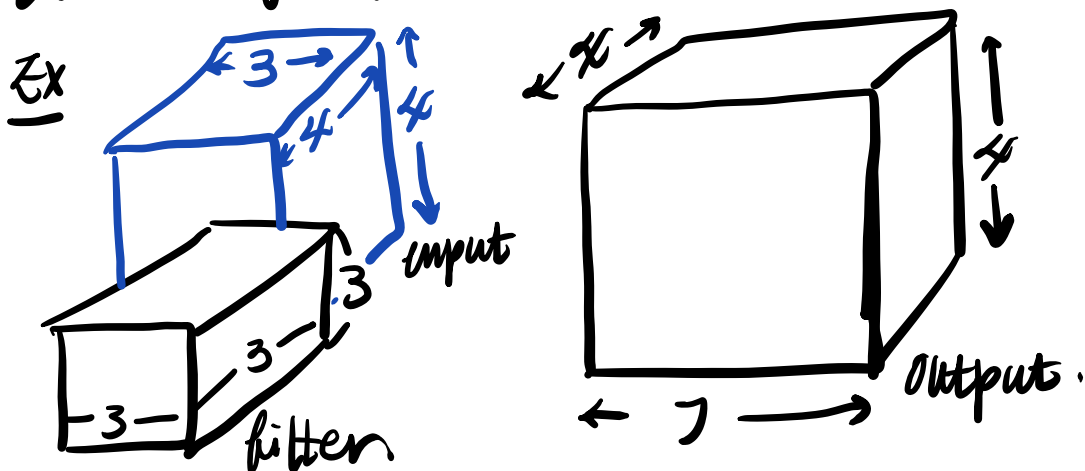


입력

[batch, in-height, in-width, in-channel]

출력

[filter-height, filter-width, in-channels, out-channels]



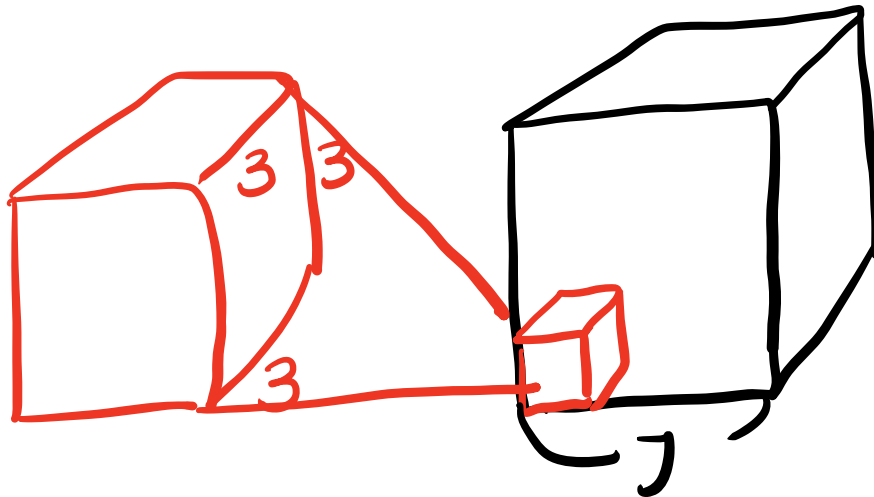
[batch, in-height=~~4~~, in-width=~~4~~, in-channel=3]

input

[filter-height=3, filter-width=3, in-channels=3,

out-channels=1]

output



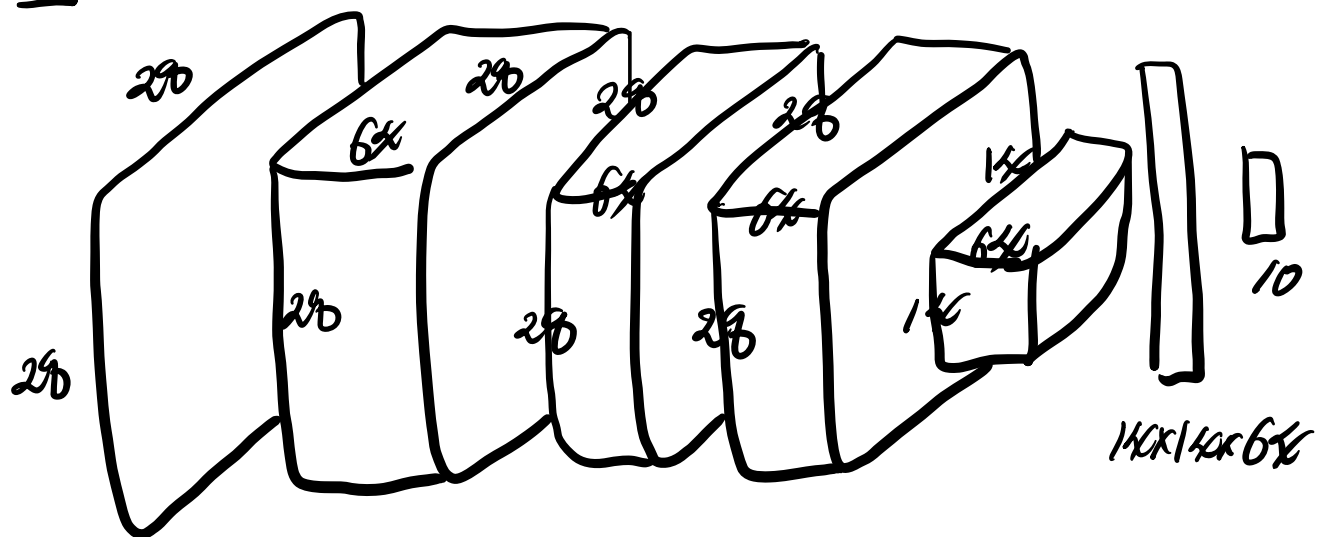
파라미터 $\rightarrow 3 \times 3 \times 3 \times 1$

?! 파라미터의 수를 적으면 적을수록 좋고

접레이어는 쌓이면 쌓일수록 좋다.

\rightarrow 다양한 스킴들 존재,

Ex



$6 \times 6 \rightarrow 64$ 개의 필터 존재.

Ex

