

12/15

- DNN

이미지 학습이 어려운데!

- ✓ ① 이미지의 개수를 많이 필요!
- ✓ ② 차원변경 ( $4\text{차원} \rightarrow 2\text{차원}$ )에 오는 data(데이터)의 무게

어떻게 하면 학습이 가능할까요?

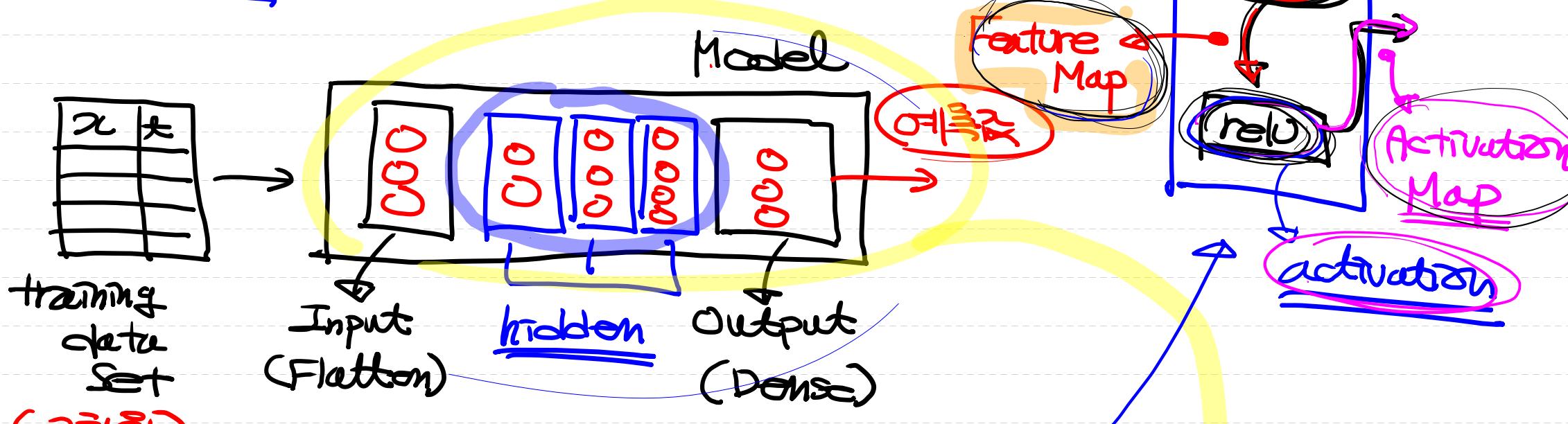
→ CNN (convolutional Neural Network)

이미지의 pixel 그 자체를 학습하는데 아니라

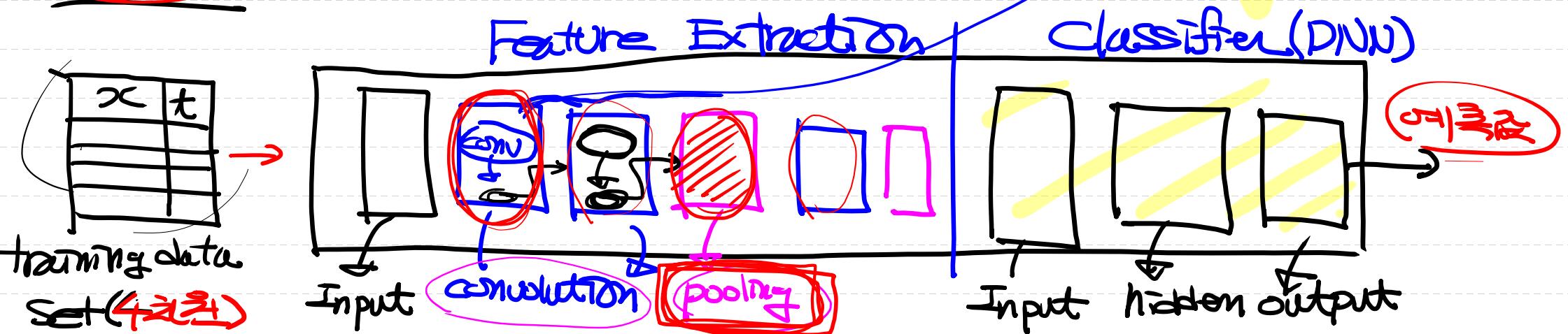
이미지의 특징을 추출해서 이를 pixel data로 바꾸면 ① data를  
학습.

# CNN Architecture

- DNN (Deep Neural Network)



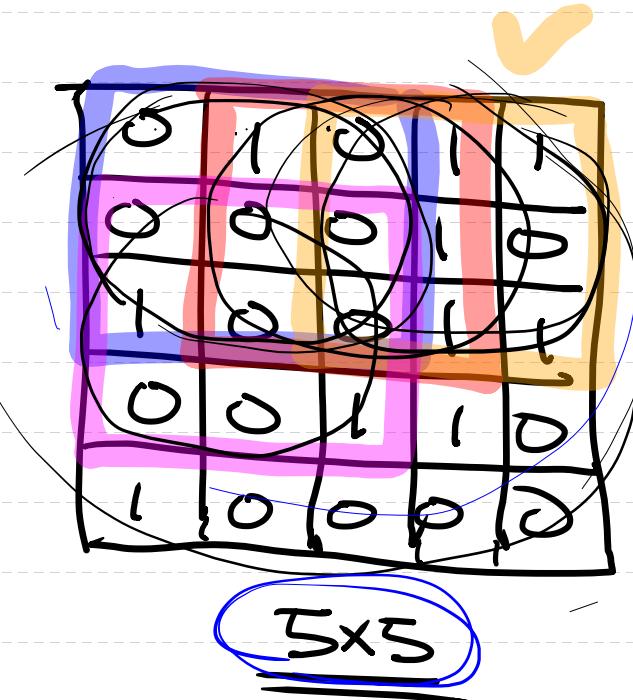
CNN (convolutional Neural Network)



이미지를 둘러싸고 있는  
입력으로

그리고 이미지의 개수  
둘다 상황을 가정

이미지  
입력



Conv 연산

Strides → 1

만약 Strides → 2

(2x2)

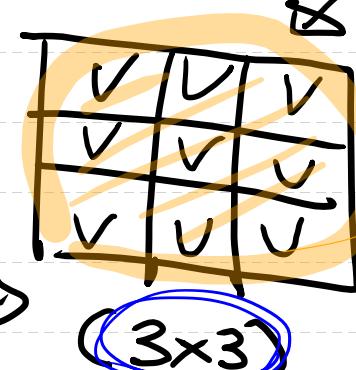
Strides  
기본값은 1

Filter (일반적 3x3)

초기화  
다른 방식으로  
초기화 가능

필터가 유통  
Feature Map

결과 Feature Map



& Feature  
Map.

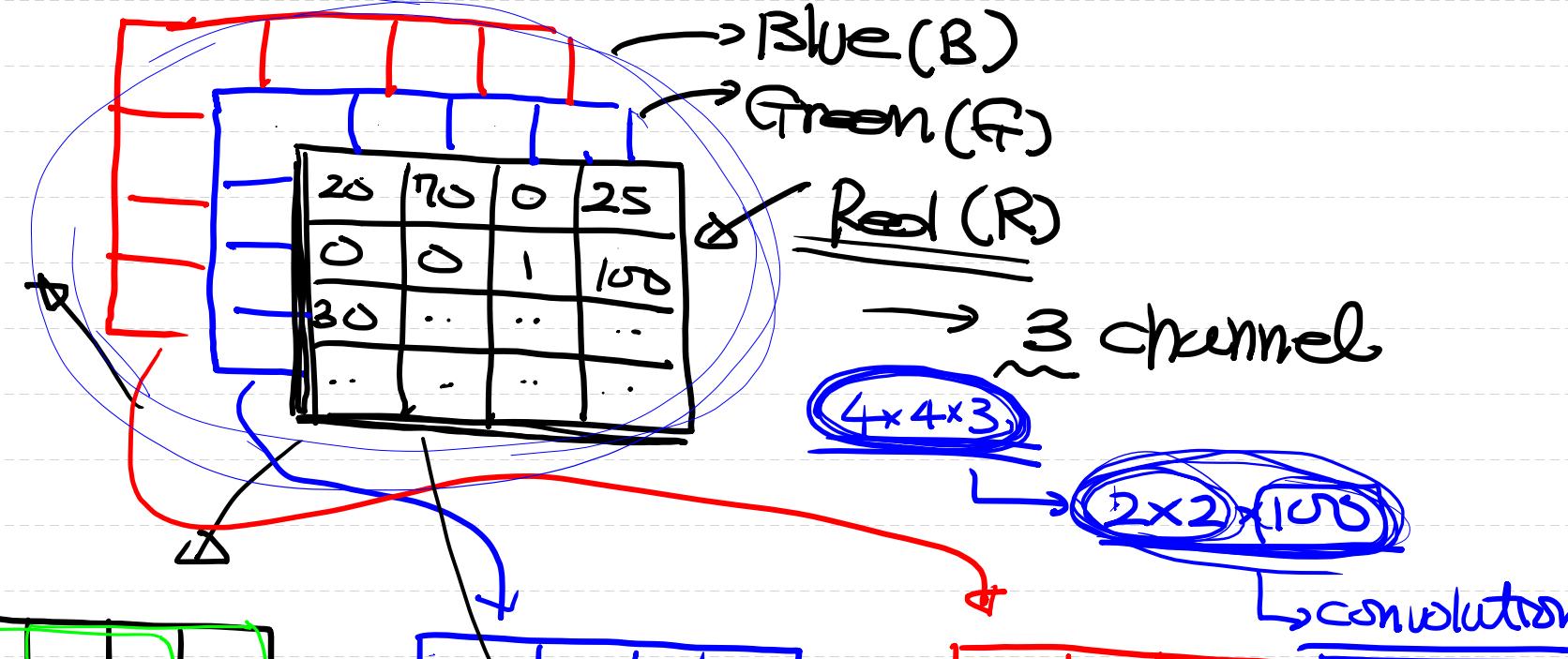
입력 이미지

3 채널 인풋

(오른쪽 인풋)

convolution 연산을

어떻게 해나요



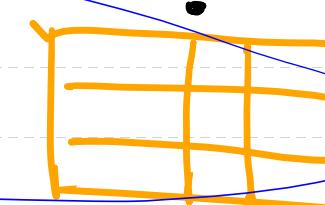
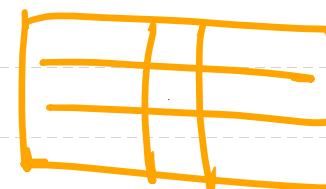
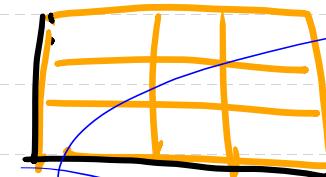
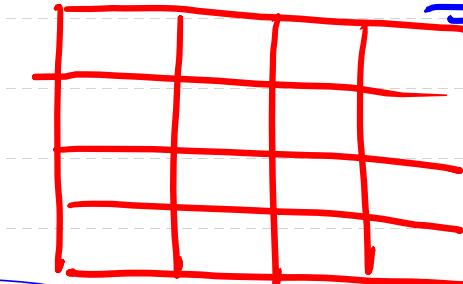
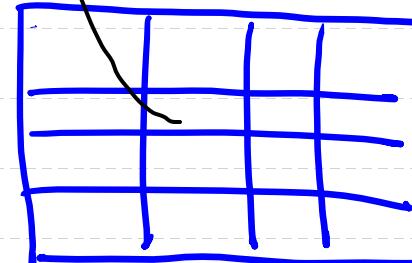
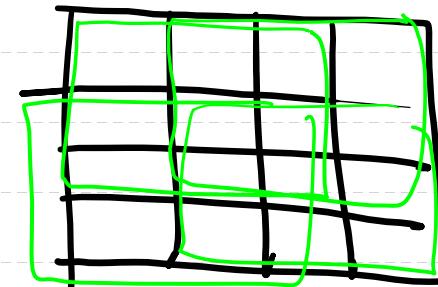
(channel #3)

Filter

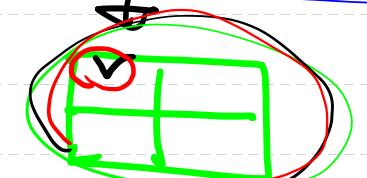
(3x3x3)

(3,3,3)

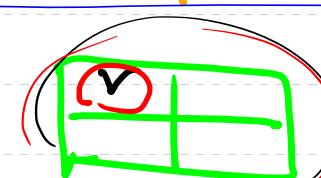
strides = 1 흔적



ORI Filter  
여러개 사용



+

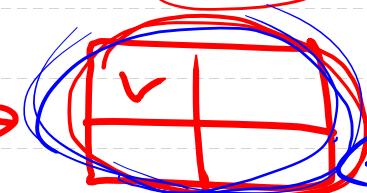


+



Feature Map (2x2x1)

Filter 1(4x3)



(2x2x100)

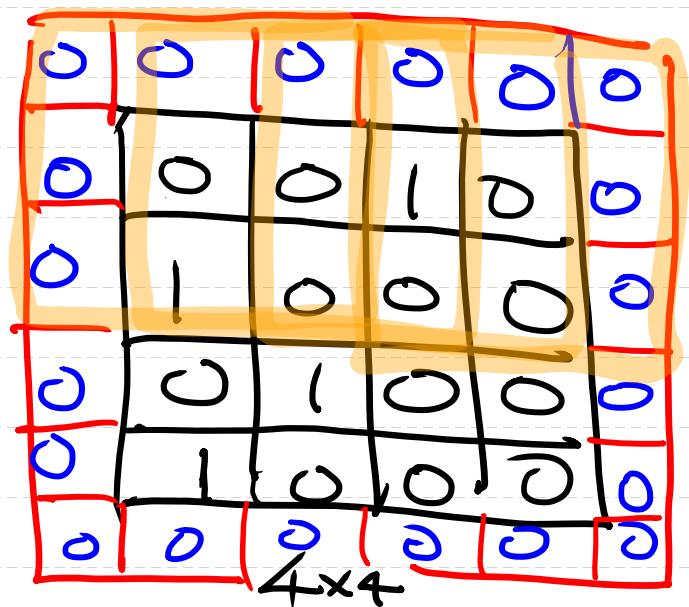
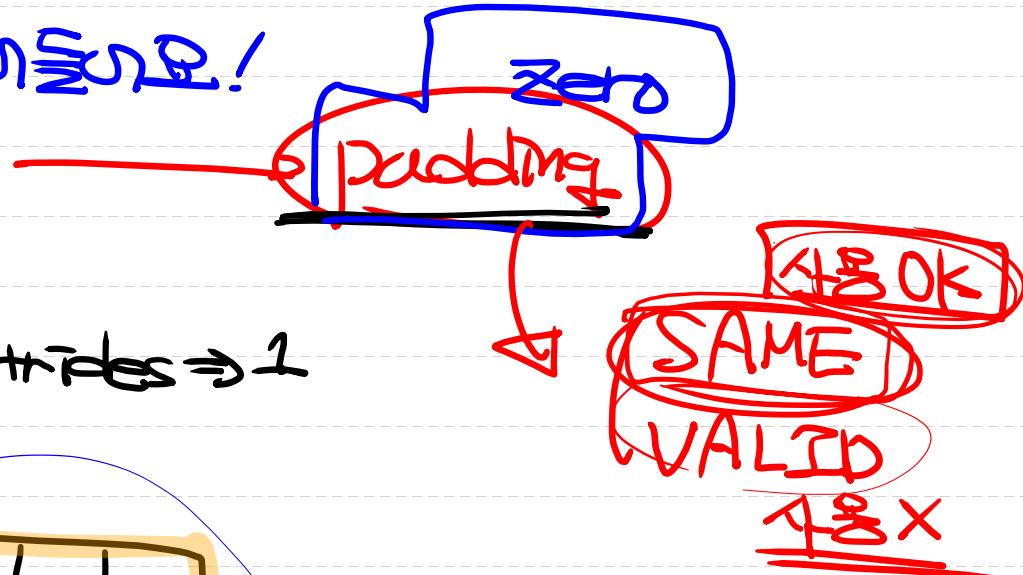
2 채널 Feature Map (3x3)

- Convolution 연산을 진행하면

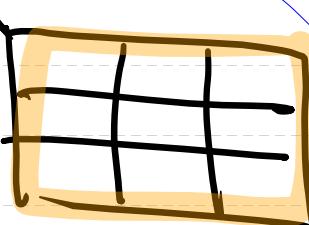
기본적으로 Filter의 크기와 Strides의 흐름에 따라

필터의 크기는 줄어들어요!

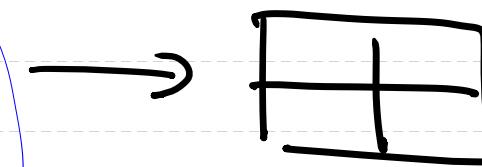
그러나 이를 허락해 주는게 있어요



Strides  $\Rightarrow 1$



(Filter)  
 $3 \times 3$



Feature Map  
 $(2 \times 2)$

padding을 이용하면

필터의 크기를 줄여도 있고 convolution 연산 횟수가 줄어드는 경우가 있음.

\* Filter로 적용하는 Feature Map을 만드는 거

어떤 의미로 일컬어 그걸로 학습해 보아요~



이미지 1장  
(회색)  
3channel

50x50

Convolution 연산 수행

Filter 1개  $\rightarrow$  2차원 Feature Map

Filter 여러 개  $\rightarrow$  여러 개  
여러 개

48x48  $\times 10$

Padding을 하지 않으면 stride가 약간 됩니!

(Filter 크기 & Strides)

12th Convolution 연산을 수행하면

Filter 개수와 따라 전체적인 data 수가 크게 증가.

~~Max~~  
Pooling

Input의 양도 줄이면서 특징을 조금 더 명확히하기 위한 부분

# • Pooling

- (1) MAX Pooling
- (2) Average pooling
- (3) Min Pooling

Convolution \*

영상을  $\rightarrow$  Feature Map

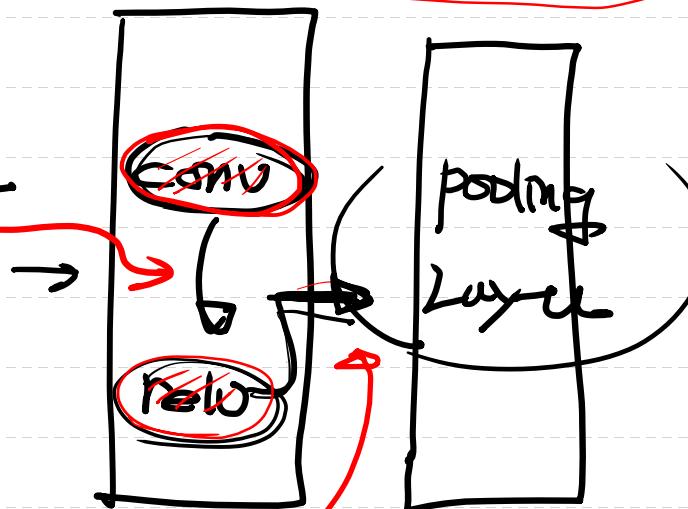
→

"ReLU" activation을 통한

\*

activation Map

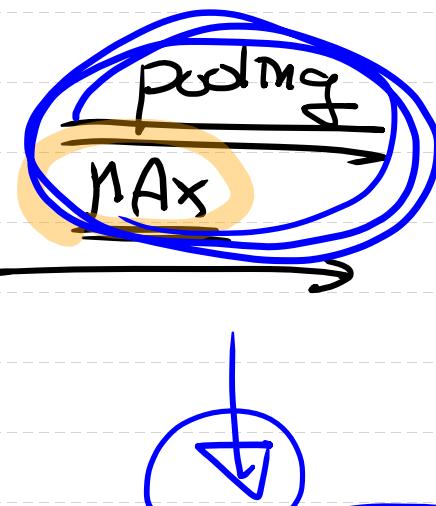
convolution Layer



activation Map

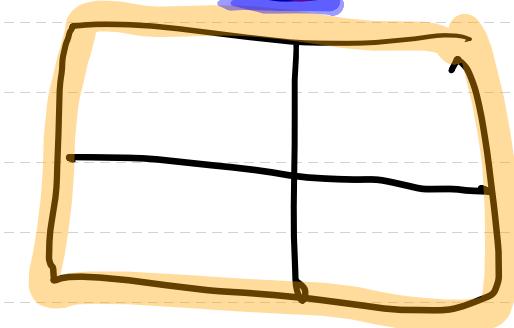
$2 \times 2$  (4x4)

13	20	30	0
8	12	3	0
34	70	33	5
11	80	10	23



(filter) → (kernel)

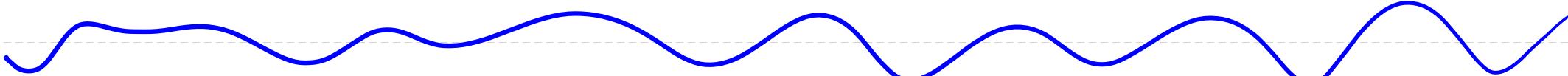
$2 \times 2$   
Stride → 2



$4 \times 4$

$2 \times 2$

20	30
11	33



(feature  
提取  
Extraction)

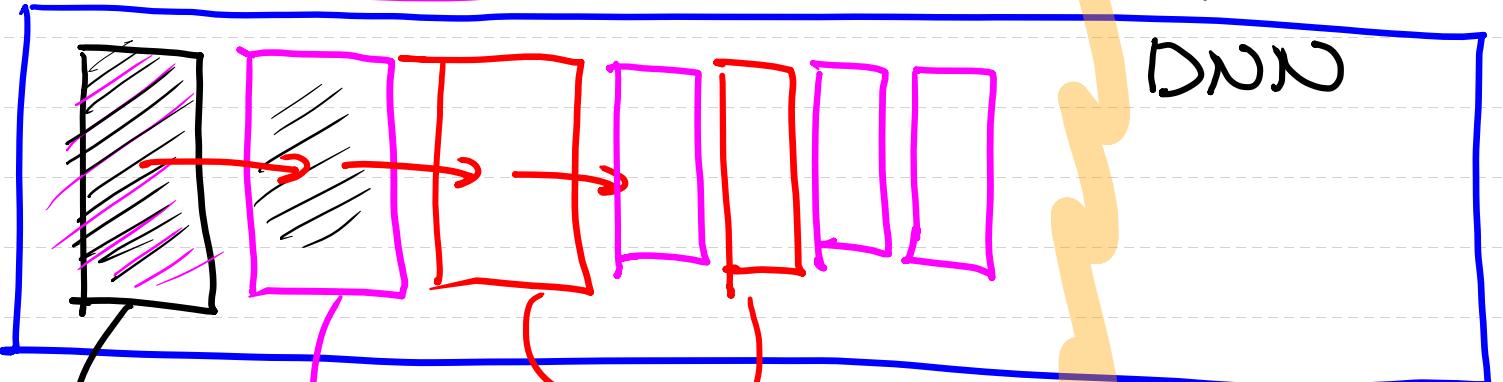
Model

DNN

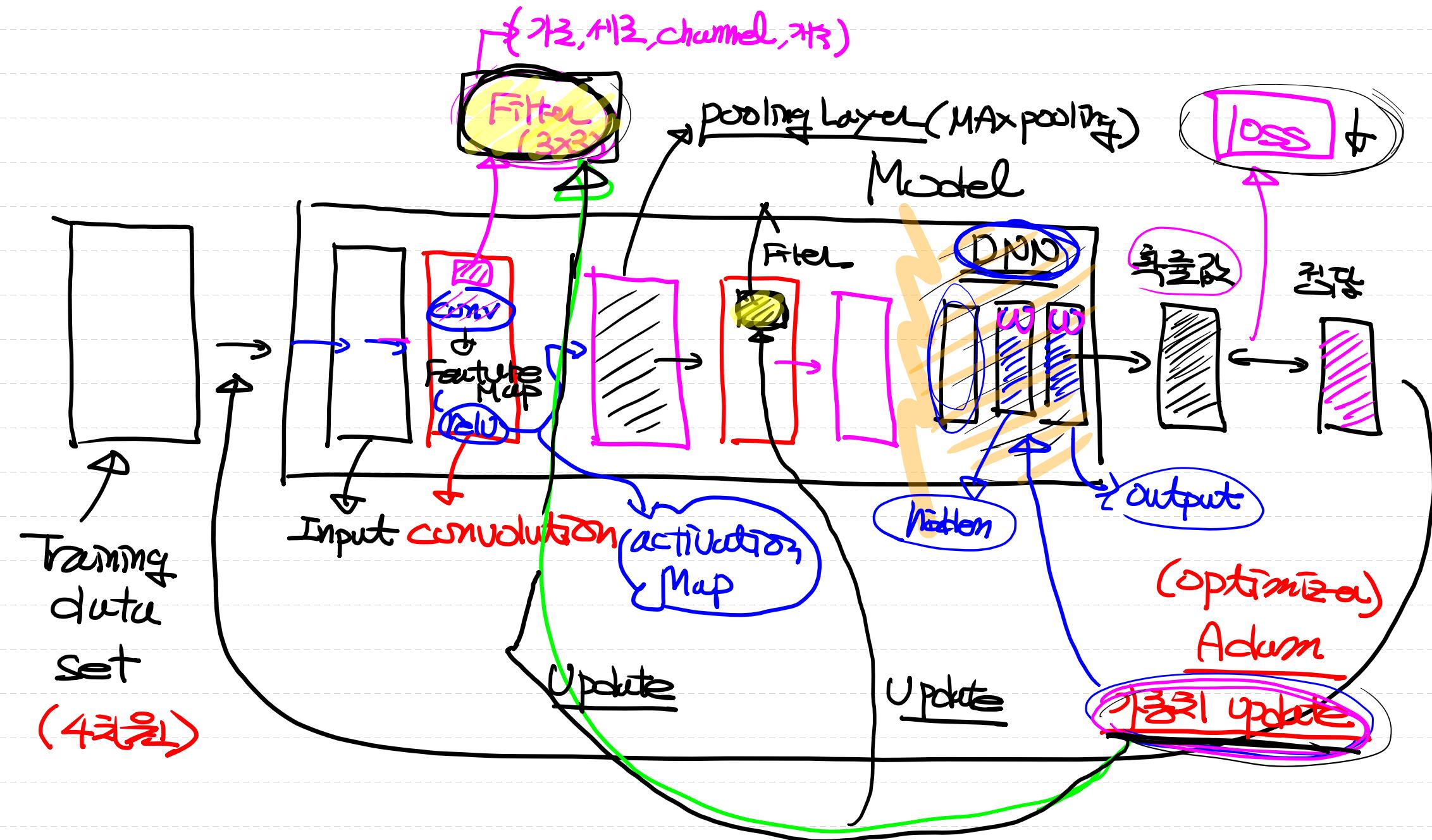
x  
Input  
Layer

Convolution  
Layer

Pooling  
Layer



# ② CNN model



## \* CNN의 Mechanism

MNIST는

(작고간단한 이미지)

Image로 머신러닝이 가능하다고 2가지 보여주는  
용도)

실제 실무에서 이용하면

(시작도 오래걸리고

중복도도 높지 않아요 (CNN을 이용해도)

실무에서 이용하면

어떻게 해야하는지 → 그러면 중복도를 ↑ 어떻게 해야하는지

개인적 도움을 봄에서 찾기!