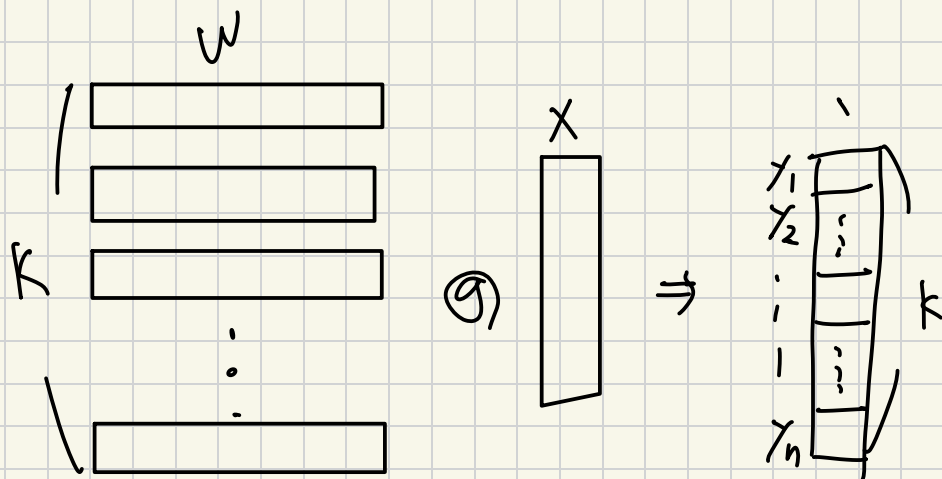
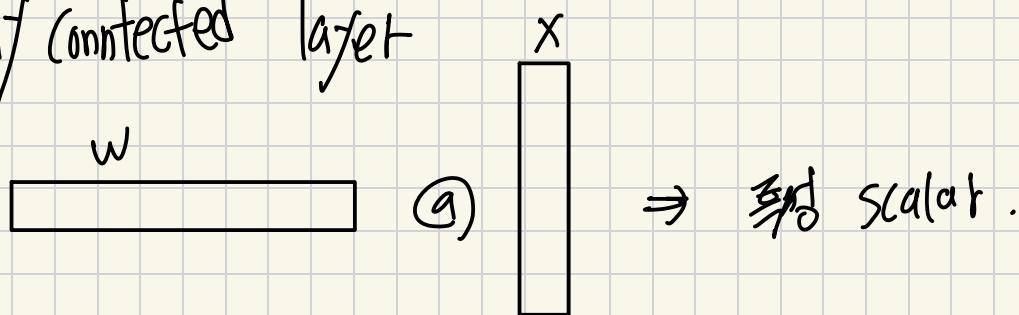
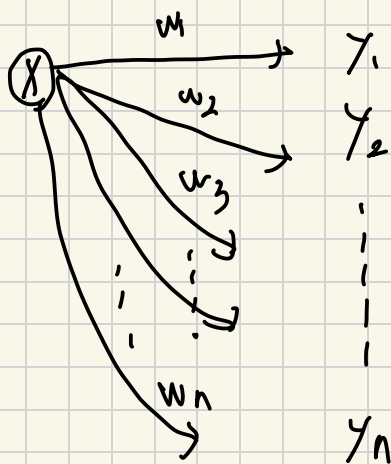


fully connected layer



이것 시각화하면



이미지는 1D가 아니라 2D 이 존재한다.

\Rightarrow 모든 이미지 pixel을 flatten 해서 받고 특징 크기의 squares mapping
시켜서 pixel 개수 x, w 를 계산하라 (수학적으로는 동일)

flatten 하면 의미적 (공간적) 정보가 훼손된다.

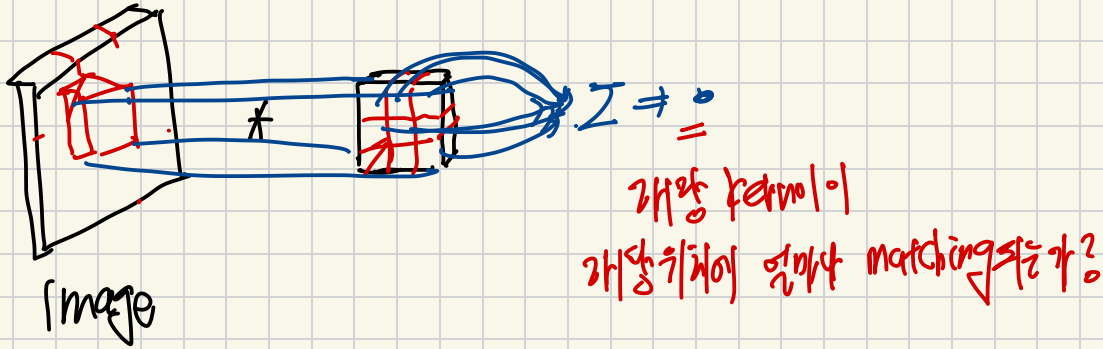
특정 conv 가 특정 pattern을 학습한다! \Rightarrow 특정 pattern이 많으면 해당 score가 증가한다.

"그게 domain knowledge 관련도 중요하다"

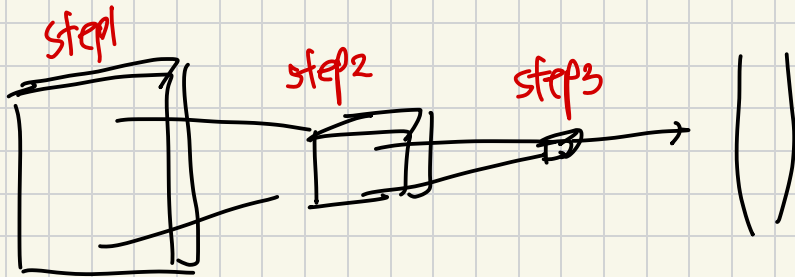
"spatial locality" = 근처만 보면 대상 object의 pattern 가능

"positional invariance" = object가 이미지 어디에 존재하든 특징을 보장한다.
이 2개의 개념을 learnable parameter가 풀이한다.

<Convolutional Layer>



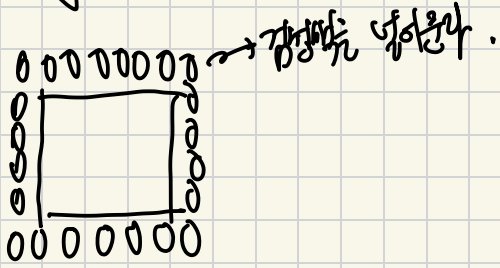
kernel의 channel = Image channel인 이유는 결국 커널과 입력의 matching/치환을 하기 위해서. (만일 channel도 크게 늘리고 conv를 layers를 사용)



\therefore Step 2의 특징은 classification을 할 수 있게 아나 step3을 개발하는 것이다!

Stride \Rightarrow kernel을 얼마나 move 시킬지.

padding \Rightarrow feature map이 작아지지 않도록 도와준다.



즉, stride와 padding으로 출력 image size 조절 가능.

모든 filter는 각 1개의 bias를 존재한다.

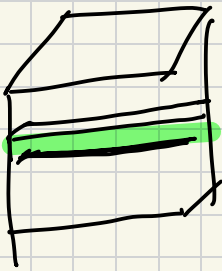
그래서 $5 \times 5 \times 3$ filter가 10개 존재하는 경우

$10((5 \times 5 \times 3) + 1)$ 개의 learnable parameter를 존재한다.

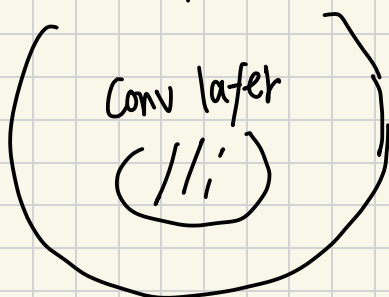
< 1차 convolution >

\Rightarrow 동일하게 \Rightarrow channel만큼 쓰여준다. (주변공간 보지 않는다)

+ channel이 두를 곱해주는 역할.

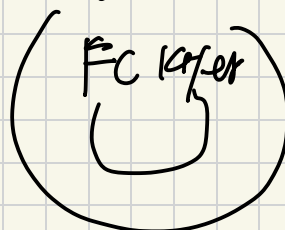


FC Layer



이렇게

(conv의 필터크기가 이미지 크기와 같을 경우.)



< pooling Layer >

각 층 X 1/2 down Sampling을 위한 역할 수행.