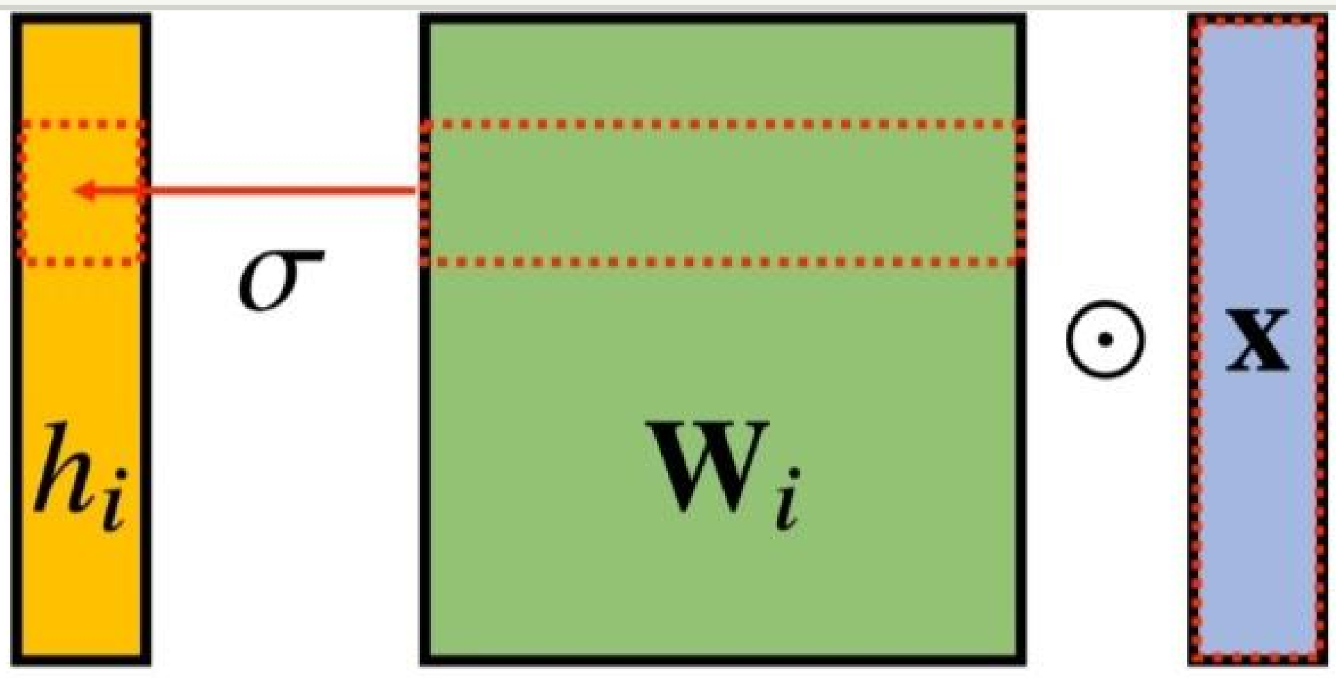
# CNN 첫 걸음.

· 지금까지 배운 다층신경망(MLP)는 각 뉴런들이 선형모델과 활성함수로 모두 연결된 구조였음.

이 때 각 성분 $h_i$에 대응하는 가중치행 $W_i$가 필요하기 때문에 가중치 행렬의 구조의 사이즈가 커진다는 문제가 있었음

$$h_i = \sigma\left(\sum_{j=1}^{p} W_{ij}x_j\right)$$

활성함수 가중치 행렬



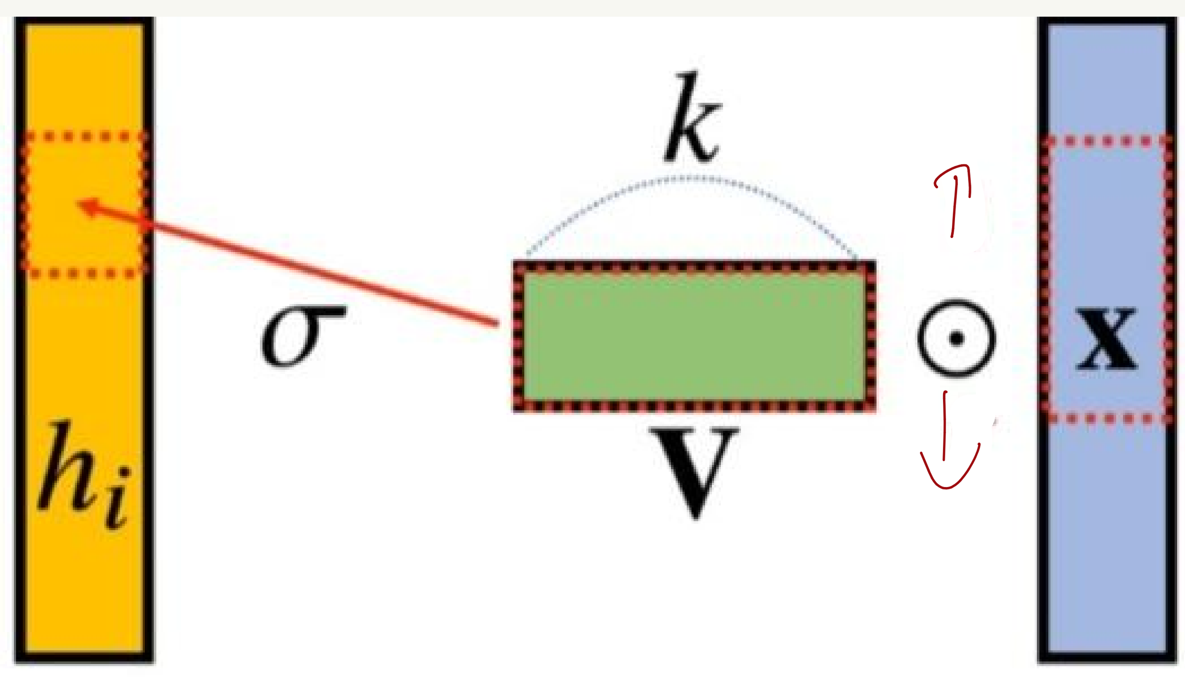

각 성분 $h_i$ 에 대응하는 가중치 행 $\mathbf{W}_i$ 이 필요하다

· CNN : 기존 MLP와 달리 커널(kernel)을 입력 벡터 상에서 움직여가면서 선형모델과 합성함수가 적용되는 구조.

고정치

$$h_i = \sigma\left(\sum_{j=1}^{k} V_j x_{i+j-1}\right)$$

커널 사이즈 활성함수 가중치 행렬




모든 $i$ 에 대해 적용되는 커널은 $\mathbf{V}$ 로 같고 커널의 사이즈만큼 $\mathbf{x}$ 상에서 이동하면서 적용합니다

=> parameter 사이즈를 많이 줄일 수 있음

고정치 local

· CNN의 수학적 의미: 신호(signal)를 커널을 이용하여 국소적으로 증폭 또는 감소시켜서 정보를 추출 또는 필터링하는 것.

continuous 연속 $$[f * g](x) = \int_{\mathbb{R}^d} f(z)g(x-z)\mathrm{d}z = \int_{\mathbb{R}^d} f(x-z)g(z)\mathrm{d}z = [g * f](x)$$

적분

discrete 이산 $$[f * g](i) = \sum_{a \in \mathbb{Z}^d} f(a)g(i-a) = \sum_{a \in \mathbb{Z}^d} f(i-a)g(a) = [g * f](i)$$

급수

· 다양한 차원에서의 Convolution

1D-conv $$[f * g](i) = \sum_{p=1}^{d} \boxed{f(p)} g(i+p)$$

$i, j, k$ 가 바뀌어도 커널 $f$ 의 값은 바뀌지 않습니다

2D-conv $$[f * g](i, j) = \sum_{p,q} \boxed{f(p, q)} g(i+p, j+q)$$

데이터 성격에 따라 사용되는 커널이 달라짐.

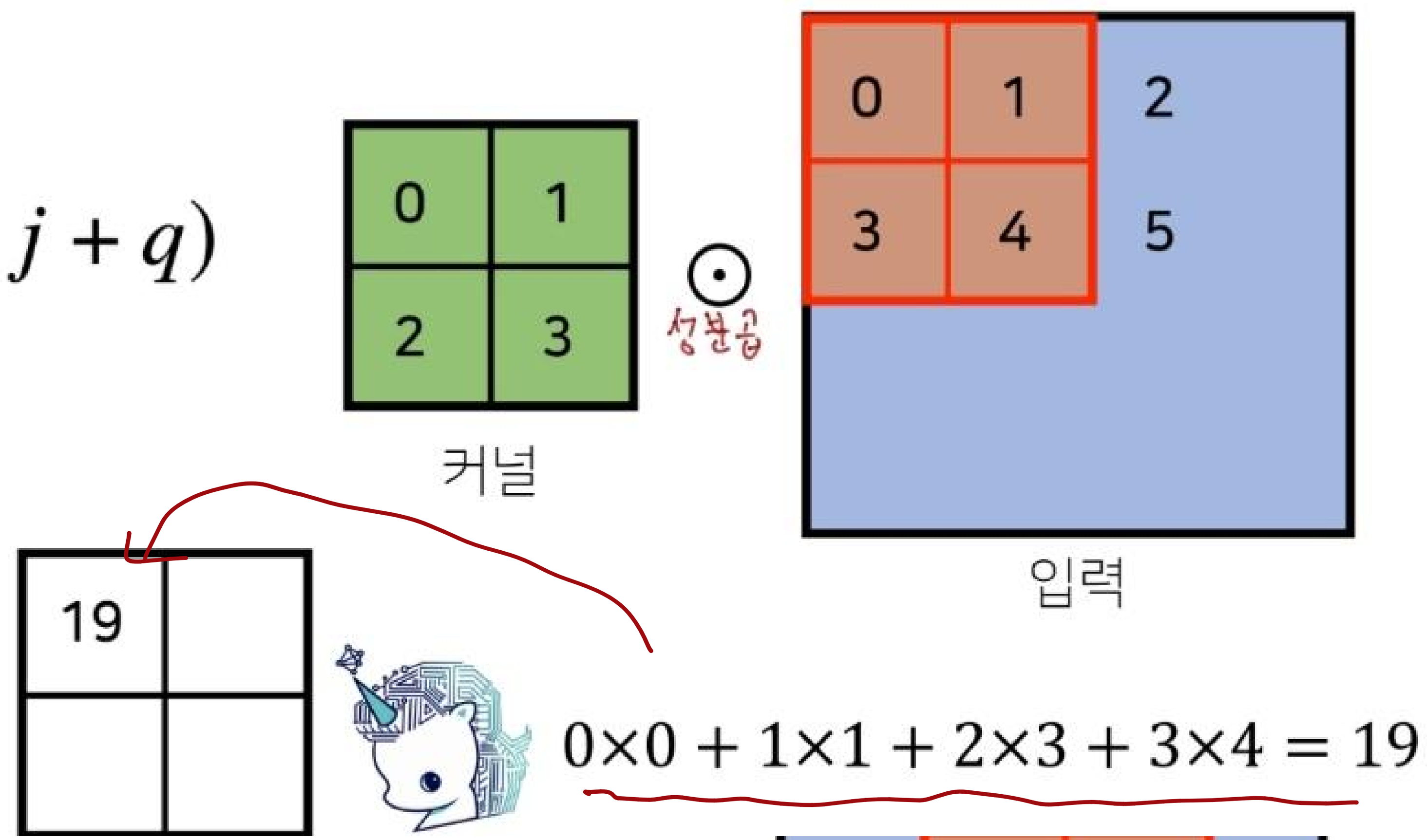
3D-conv $$[f * g](i, j, k) = \sum_{p,q,r} \boxed{f(p, q, r)} g(i+p, j+q, k+r)$$

- 2차원 Convolution 이해하기.

$$[f * g](i, j) = \sum_{p,q} f(p, q) g(i+p, j+q)$$

커널: $f(p, q)$

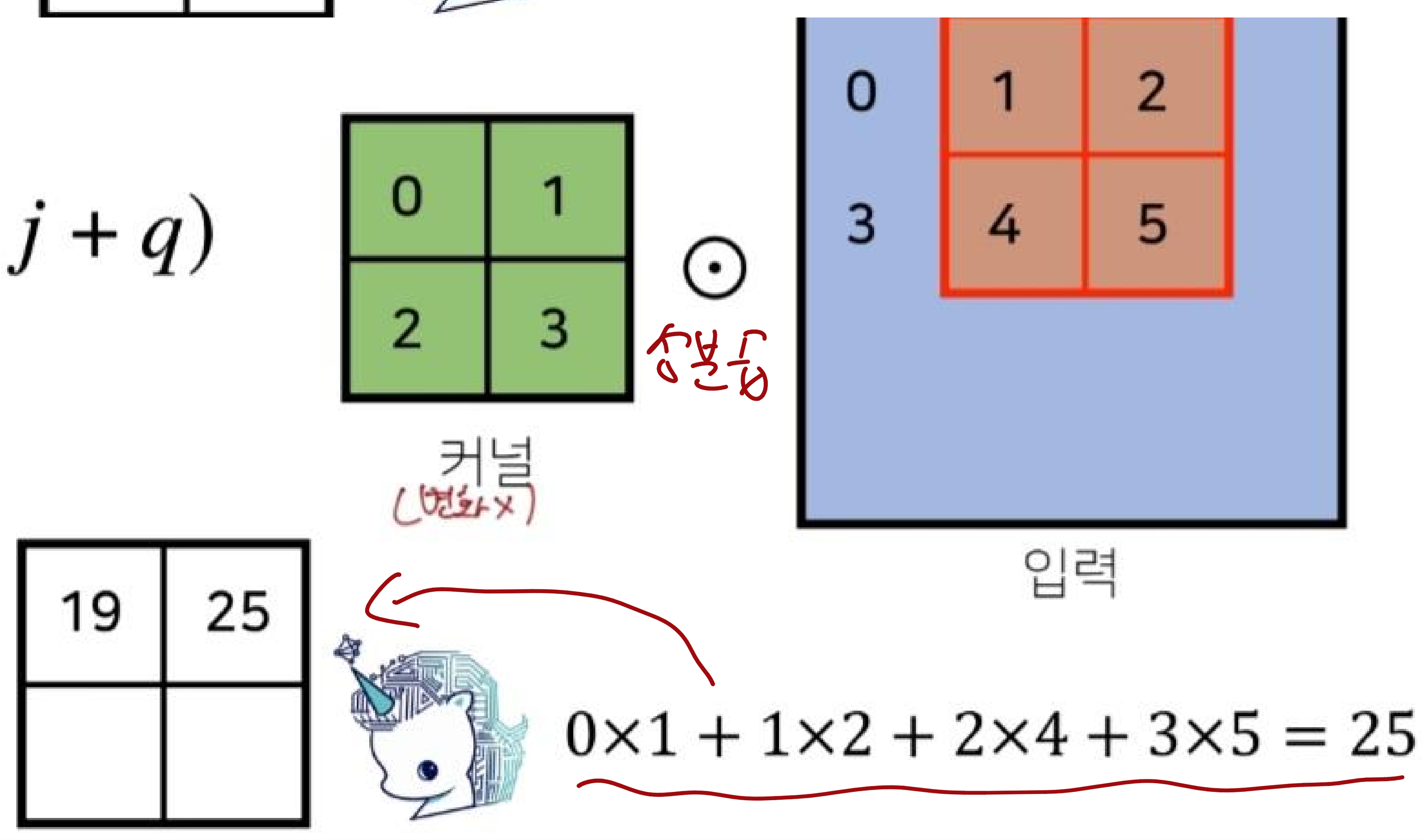
입력: $g(i+p, j+q)$




$$[f * g](i, j) = \sum_{p,q} f(p, q) g(i+p, j+q)$$

커널: $f(p, q)$

입력: $g(i+p, j+q)$




· 입력크기를 $(H, W)$, 커널 크기를 $(K_H, K_W)$, 출력크기를 $(O_H, O_W)$라 하면, 출력 크기는 다음과 같음.

$$\begin{cases} O_H = H - K_H + 1 \\ O_W = W - K_W + 1 \end{cases}$$

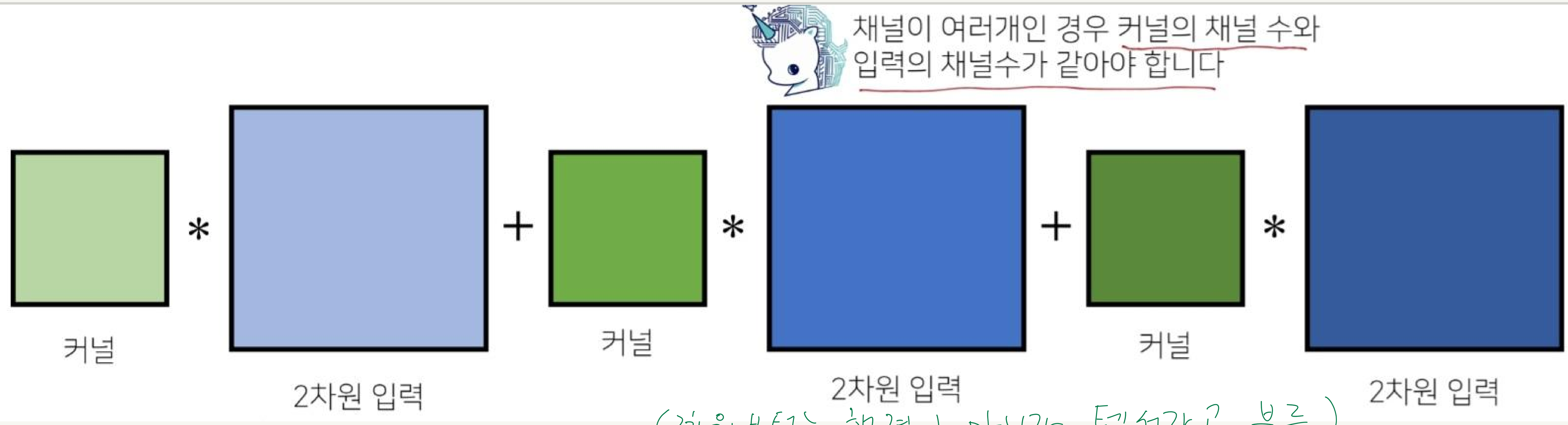
· ex) 28×28 입력을 3×3 커널로 2D-conv 연산을 하면,

$O_H = 28 - 3 + 1 = 26$

$O_W = 28 - 3 + 1 = 26$

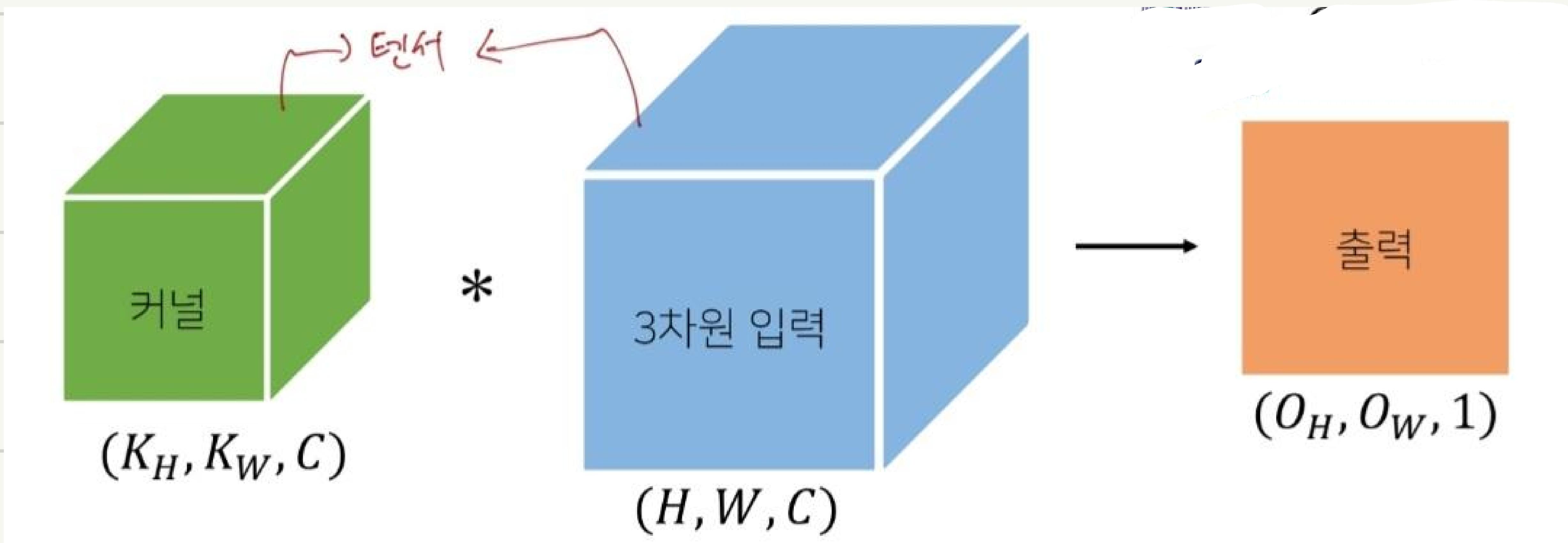
$\Rightarrow 26 \times 26$

· 채널이 여러개인 2차원 입력의 경우, 2차원 Convolution을 채널 개수만큼 적용하면 됨.

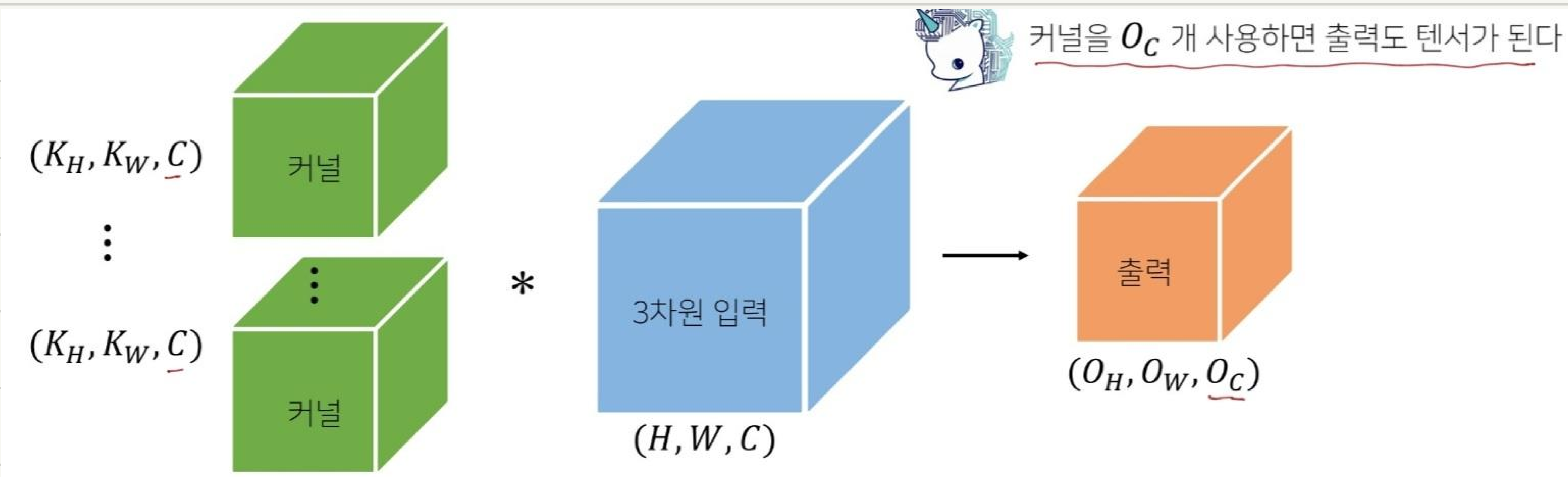



(3차원부터는 행렬이 아니라 텐서라고 부름)

||




· 출력을 텐서로 하고싶으면?



· Convolution 연산의 역전파 이해하기

- Convolution 연산은 커널이 모든 입력데이터에 공통으로 적용되기 때문에, 역전파를 계산할 때에도 convolution 연산이 나오게 됨.

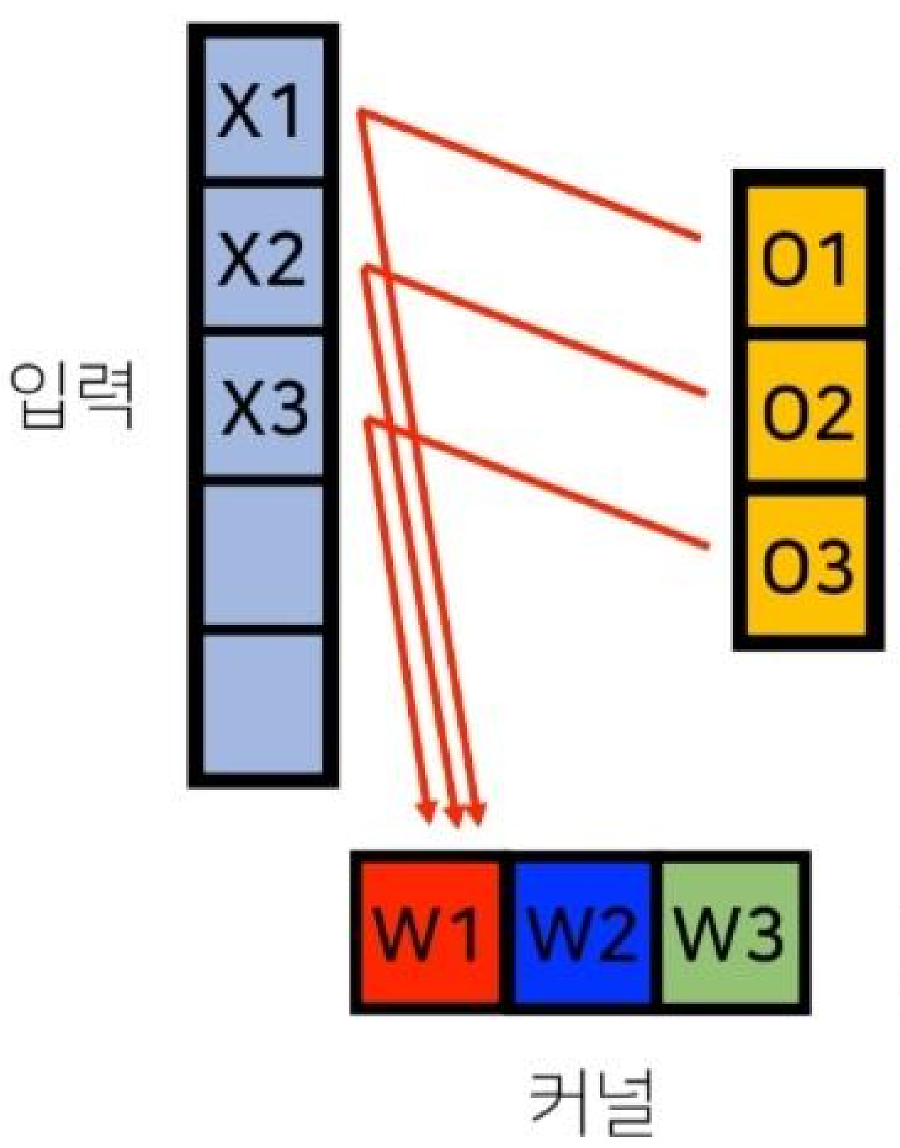
(Convolution 연산에 미분을 해도 Convolution이 그대로 나옴.)

$$
\begin{aligned}
\frac{\partial}{\partial x}[f * g](x) &= \frac{\partial}{\partial x}\int_{\mathbb{R}^d} f(y)g(x-y)\mathrm{d}y \\
&= \int_{\mathbb{R}^d} f(y)\frac{\partial g}{\partial x}(x-y)\mathrm{d}y \\
&= [f * g'](x)
\end{aligned}
$$

⇒ Convolut
Convoluti

Discrete 일 때도 마찬가지로 성립한다

⇓




$$\delta_1 \quad \delta_2 \quad \delta_3$$

$$\frac{\partial \mathcal{L}}{\partial w_i} = \sum_j \delta_j x_{i+j-1}$$

각 커널에 들어오는 모든 그레디언트를 더하면 결국 convolution 연산과 같다

$$\frac{\partial \mathcal{L}}{\partial w_1} = \delta_1 x_1 + \delta_2 x_2 + \delta_3 x_3$$