

## 7. 합성곱 신경망

### (Convolutional Neural Network: CNN)

---

1. CNN 개요
2. 컨볼루션 연산. 풀링. 패딩
3. 필터를 통해 데이터 특징을 추출하는 원리

# 1. CNN 개요

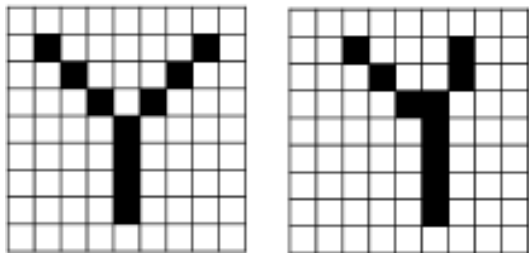
## ❖ 합성곱 신경망(Convolutional Neural Network)?

- 이미지 처리에 탁월한 성능을 보이는 신경망
- 합성곱층(Convolution layer)과 풀링층(Pooling layer)으로 구성

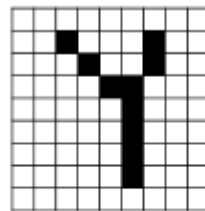


# 1. CNN 개요

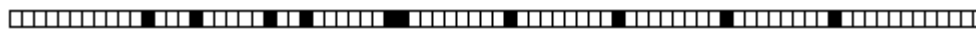
## ❖ 1. CNN 개요



y를 정자로 쓴 손글씨와 휘갈겨  
쓴 손글씨를 2차원 텐서인 행렬  
로 표현한 것



↓ 변환



이미지를 1차원 텐서인 벡터로 변환하고 다층 퍼셉트론의 입력층으로 사용

두 이미지를 기계가 같은 글자(y)로 인식하는가?    공간적 정보를 잃은 1차원 벡터로 된 정보로 기계나 사람이 y로 인식하는가?

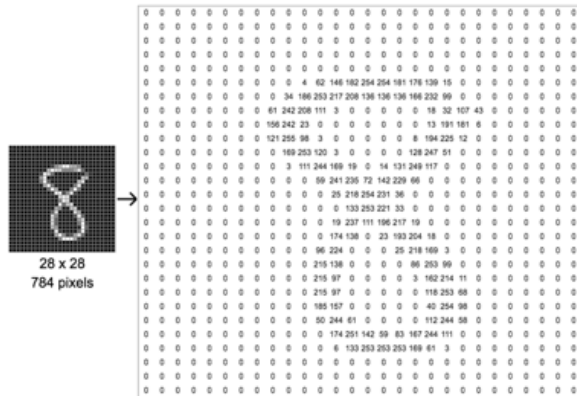
**이미지의 공간적인 구조 정보를 보존하면서 학습할 수 있는 방법이 필요하며, 이를 위해 사용하는 것이 합성곱 신경망 임**

공간적 정보란 : 거리가 가까운 어떤 픽셀들끼리는 어떤 연관이 있고, 어떤 픽셀들끼리는 값이 비슷하거나 등을 포함

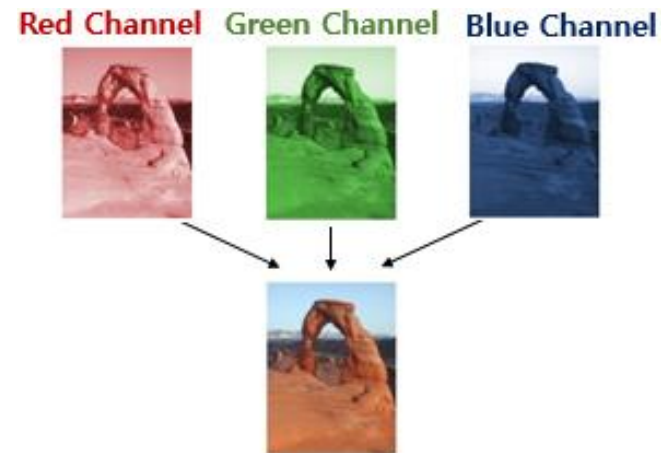
# 1. CNN 개요

## ❖ 채널(Channel)

- 기계는 글자나 이미지보다 숫자(텐서)를 더 잘 처리함
- 이미지는 (높이, 너비, 채널)이라는 3차원 텐서
- 높이: 이미지의 세로 방향 픽셀 수, 너비: 이미지의 가로 방향 픽셀 수, 채널: 색 성분을 의미
- 흑백 이미지는 채널 수가 1이며, 각 픽셀은 0부터 255 사이의 값을 가집니다. 아래는  $28 \times 28$  픽셀의 손글씨 데이터를 보여줌



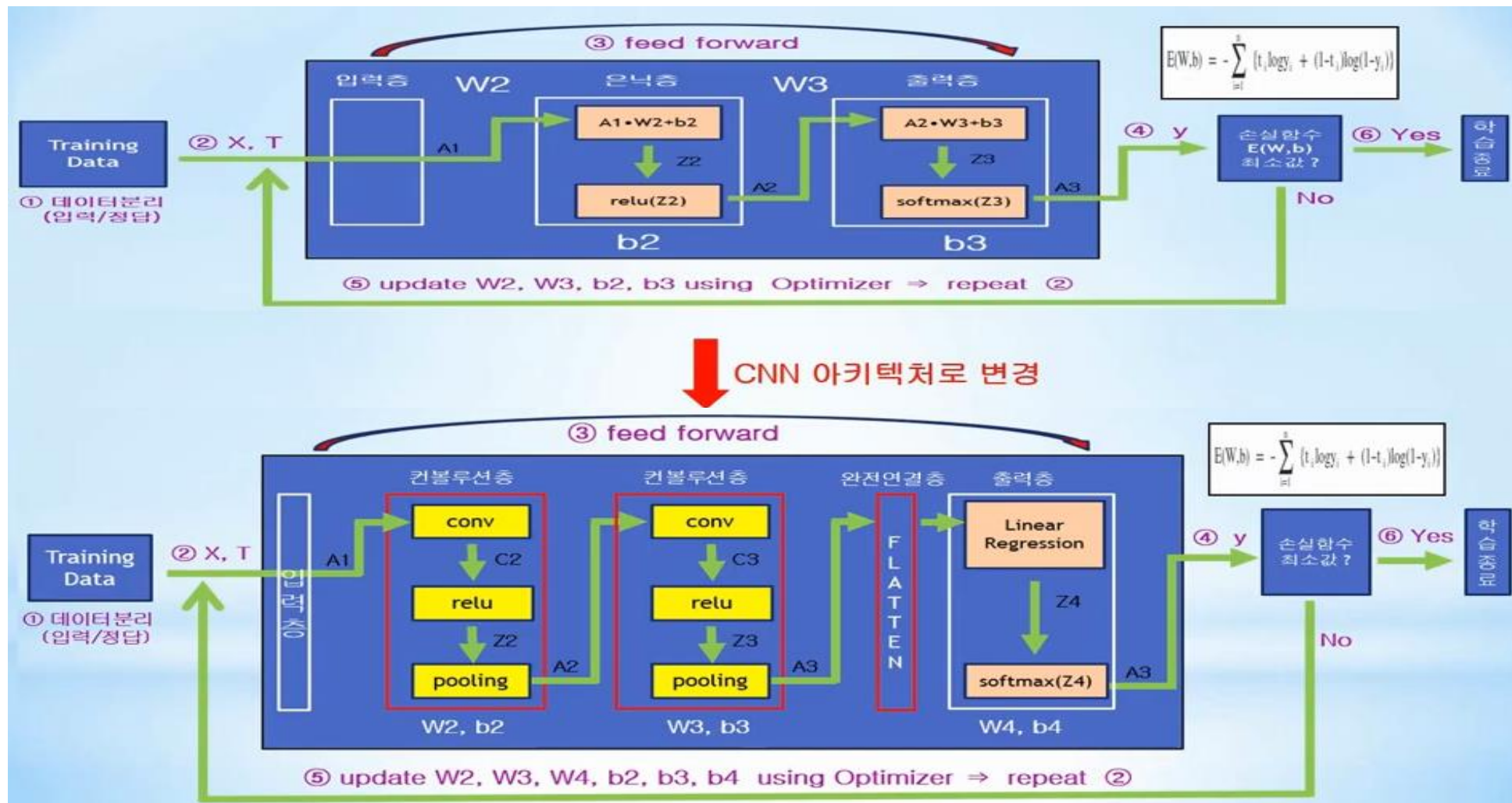
( $28 \times 28 \times 1$ )의 크기를 가지는 3차원 텐서.



( $28 \times 28 \times 3$ )의 크기를 가지는 3차원 텐서

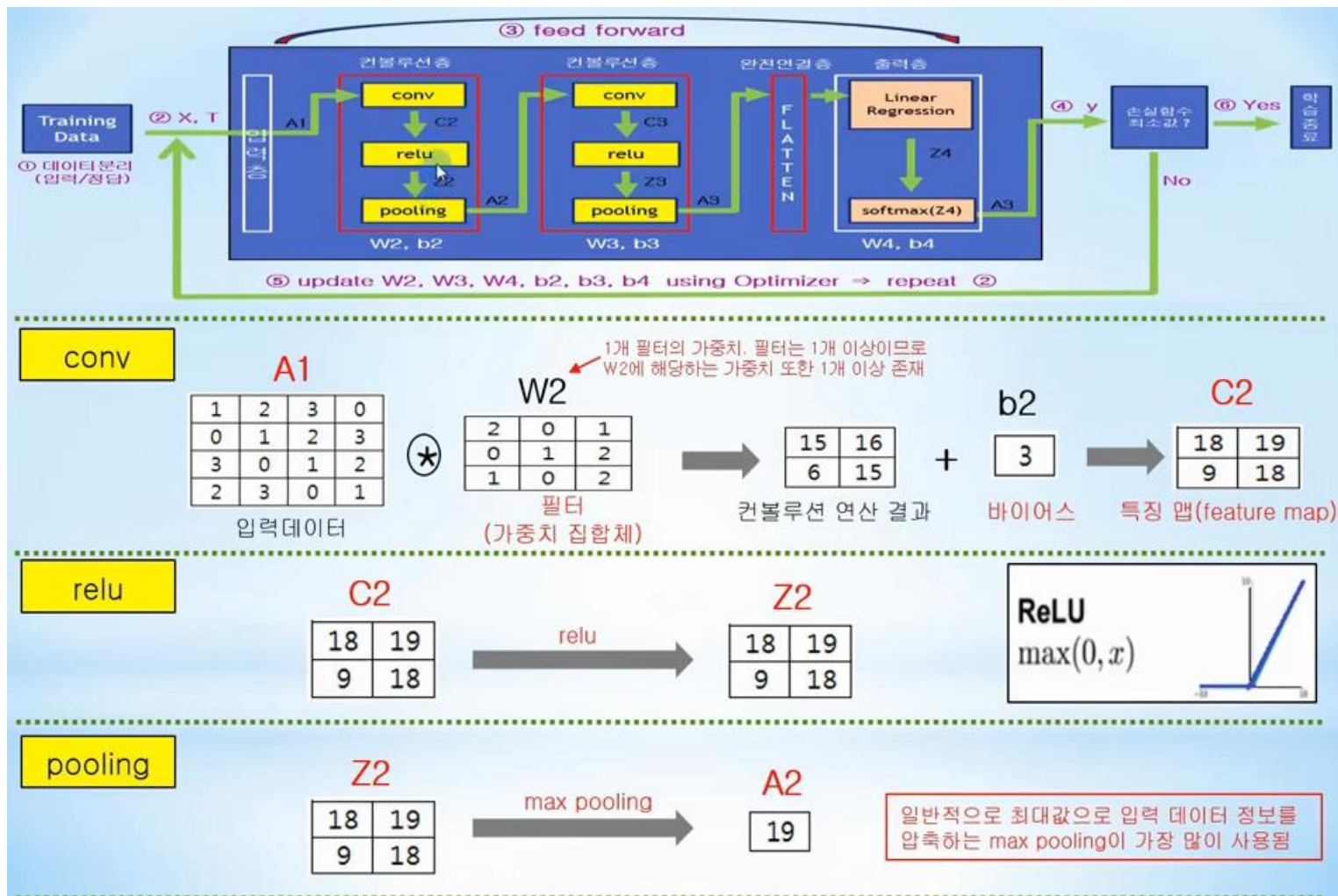
# 1. CNN 개요

## ❖ 아키텍처 비교(ANN vs. CNN)



## 2. 컨보루션 연산. 풀링. 패딩

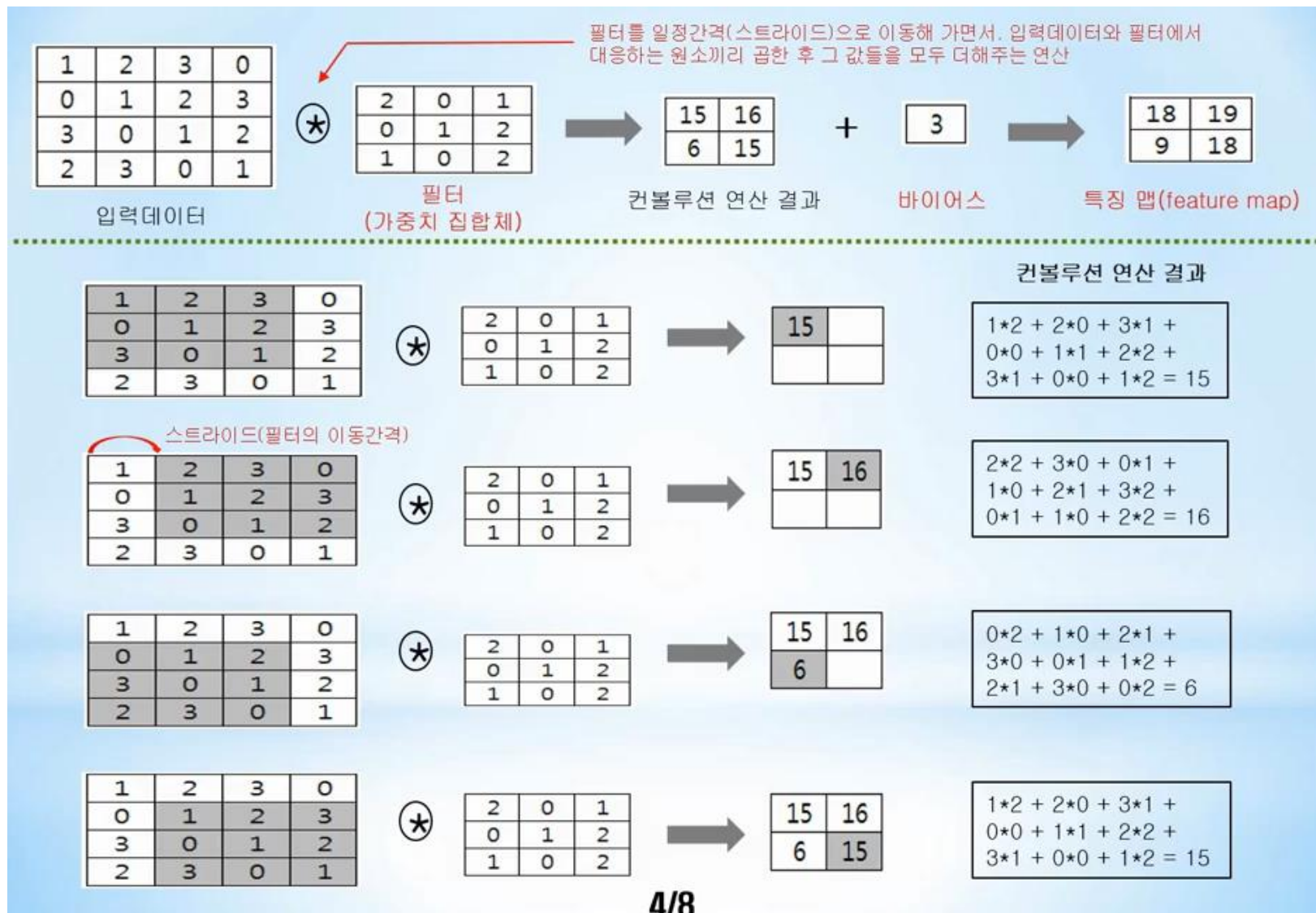
### ❖ 합성곱/풀링 연산(Convolution/Pooling operation)





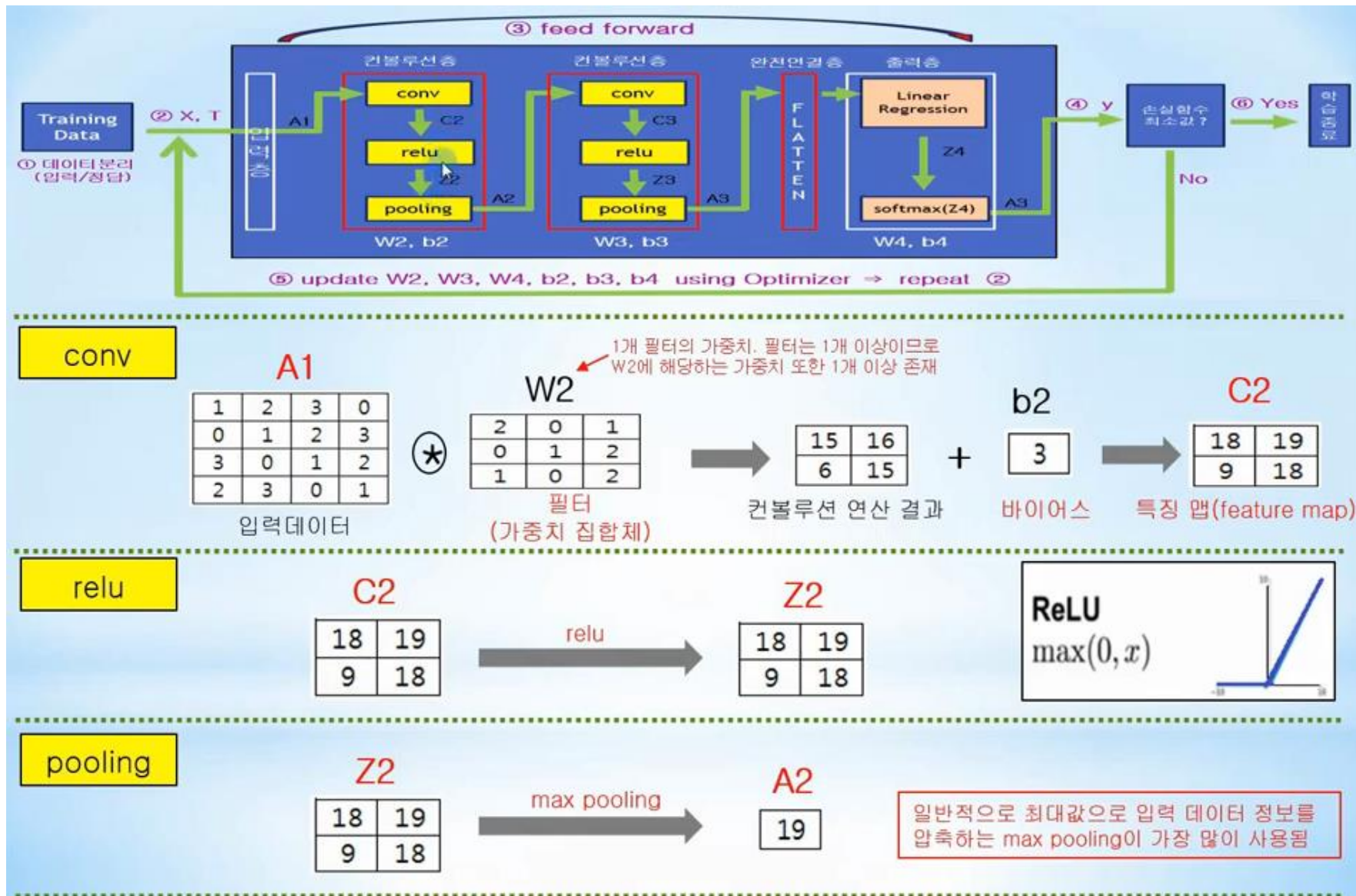
## 2. 컨볼루션 연산. 풀링. 패딩

### ❖ 합성곱 연산



## 2. 컨볼루션 연산. 풀링. 패딩

### ❖ ReLU/Pooling 연산

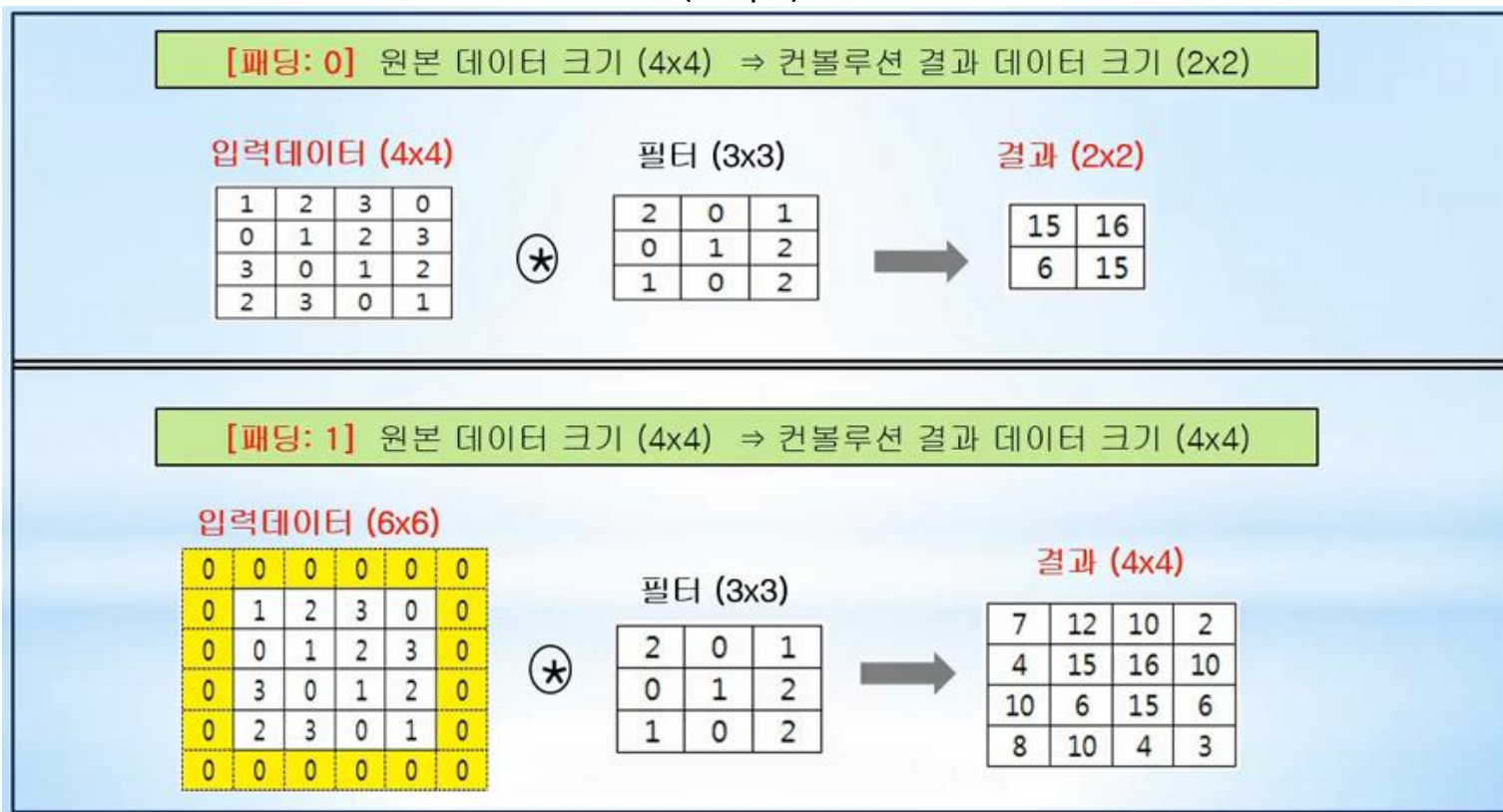




## 2. 컨볼루션 연산. 풀링. 패딩

### ❖ 패딩(Padding)

- 컨볼루션 연산을 수행하기 전에 **입력 데이터 주변을 특정 값(예를 들면 0)으로 채우는 것**, 컨볼루션 연산에서 자주 이용되는 방법
- 컨볼루션 연산을 수행하면 데이터 크기(shape)이 줄어드는 단점을 방지



## 2. 컨볼루션 연산. 폴링. 패딩

### ❖ 컨볼루션 연산을 통한 출력 데이터 크기 계산

- 입력 데이터 크기(H,W), 필터 크기(FH,FW), 패딩 P, 스트라이드 S일 때 **출력 데이터 크기 (OH, OW)**

$$OH = \frac{H + 2P - FH}{S} + 1$$

$$OW = \frac{W + 2P - FW}{S} + 1$$

[예1] 입력 (4, 4), 필터 (3, 3), 패딩 1, 스트라이드 1 ⇒ 출력 (4, 4)

$$OH = \frac{4 + 2*1 - 3}{1} + 1 = 4$$

$$OW = \frac{4 + 2*1 - 3}{1} + 1 = 4$$

[예2] 입력 (7, 7), 필터 (3, 3), 패딩 0, 스트라이드 2 ⇒ 출력 (3, 3)

$$OH = \frac{7 + 2*0 - 3}{2} + 1 = 3$$

$$OW = \frac{7 + 2*0 - 3}{2} + 1 = 3$$

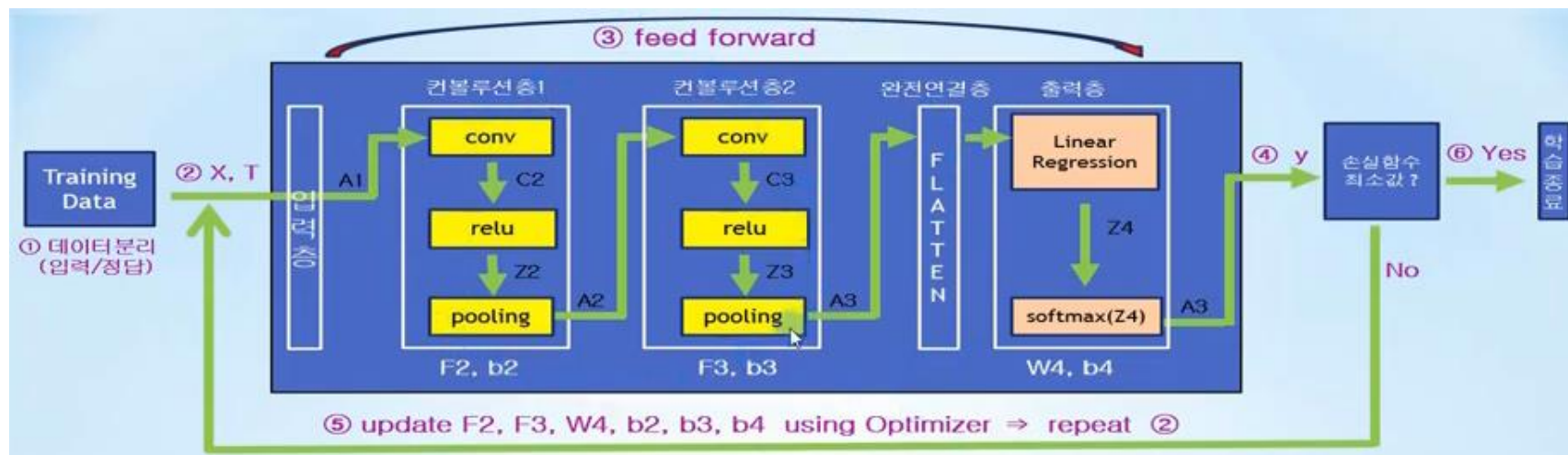
[예3] 입력 (28, 31), 필터 (5, 5), 패딩 2, 스트라이드 3 ⇒ 출력 (10, 11)

$$OH = \frac{28 + 2*2 - 5}{3} + 1 = 10$$

$$OW = \frac{31 + 2*2 - 5}{3} + 1 = 11$$

### 3. 필터를 통해 데이터 특징을 추출하는 원리

#### ❖ 컨볼루션 층(convolution layer) 역할 => 입력 데이터의 특징 추출



	conv 출력	relu 출력	pooling 출력	컨볼루션 층 역할
컨볼루션층1	$A1 \otimes F2 + b2 = C2$ 입력 데이터      필터 (가중치 집합체)      바이어스      특징 맵 (feature map)	$C2 > 0, C2 \leq 0, 0$ $Z2$	$A2$ max pooling	입력데이터 A1과 가중치들의 집합체인 1개 이상의 필터 F2와 컨볼루션 연산을 통해 입력데이터 A1의 특징(feature)을 추출하는 역할을 수행함
컨볼루션층2	$A2 \otimes F3 + b3 = C3$ 입력 데이터      필터 (가중치 집합체)      바이어스      특징 맵 (feature map)	$C3 > 0, C3 \leq 0, 0$ $Z3$	$A3$ max pooling	입력데이터 A2과 가중치들의 집합체인 1개 이상의 필터 F3와 컨볼루션 연산을 통해 입력데이터 A2의 특징(feature)을 추출하는 역할을 수행함

### 3. 필터를 통해 데이터 특징을 추출하는 원리

---

#### ❖ 컨볼루션 층(convolution layer) 역할

- ① 입력데이터와 1개 이상의 필터들과 컨볼루션 연산을 통해서
- ② 입력데이터 특징(feature)을 추출하여 특징맵(feature map)을 만들고
- ③ 특징맵에서 초대 값을 뽑아내서 다음 층으로 전달

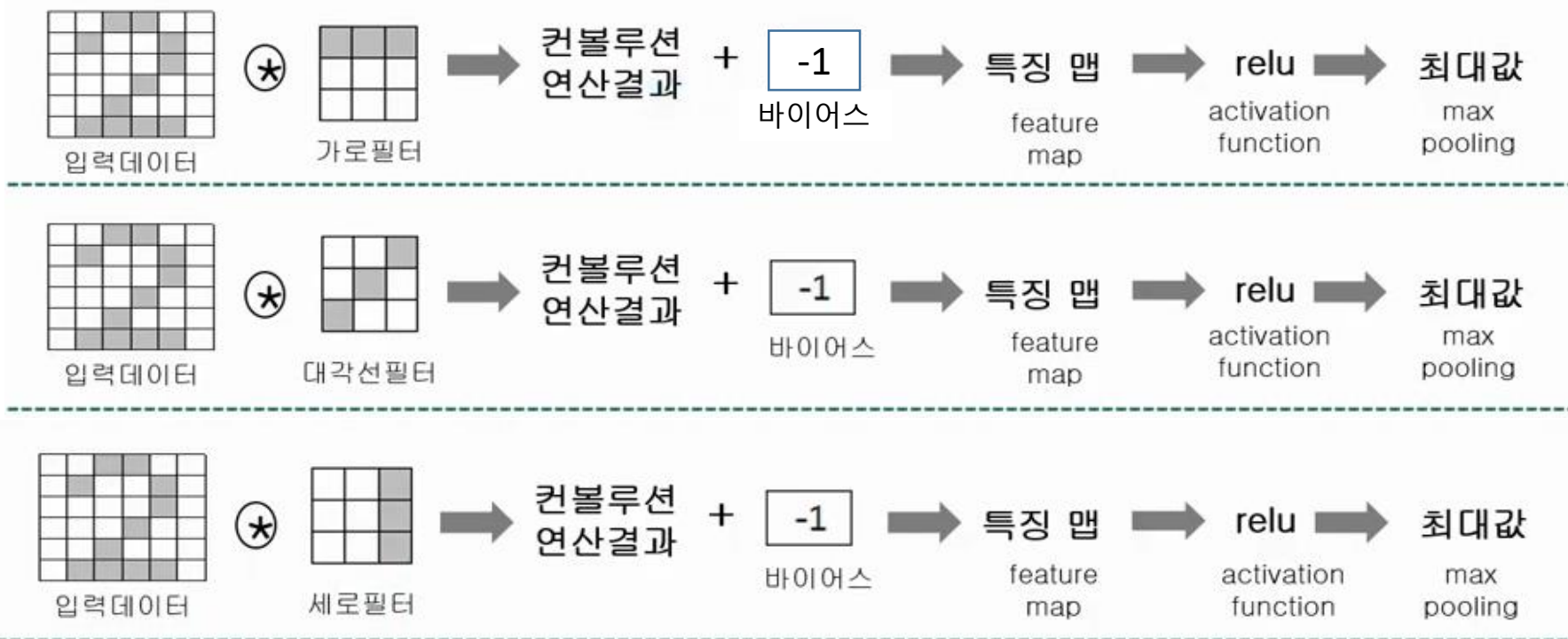


필터를 통해 데이터 특징을 추출 ?

### 3. 필터를 통해 데이터 특징을 추출하는 원리

#### ❖ 특징추출 과정

➤ 입력데이터 1개 (숫자 2)에 필터 3개 (가로, 대각선, 세로 필터) 적용 (계산 편의를 위해 패딩 적용하지 않음)



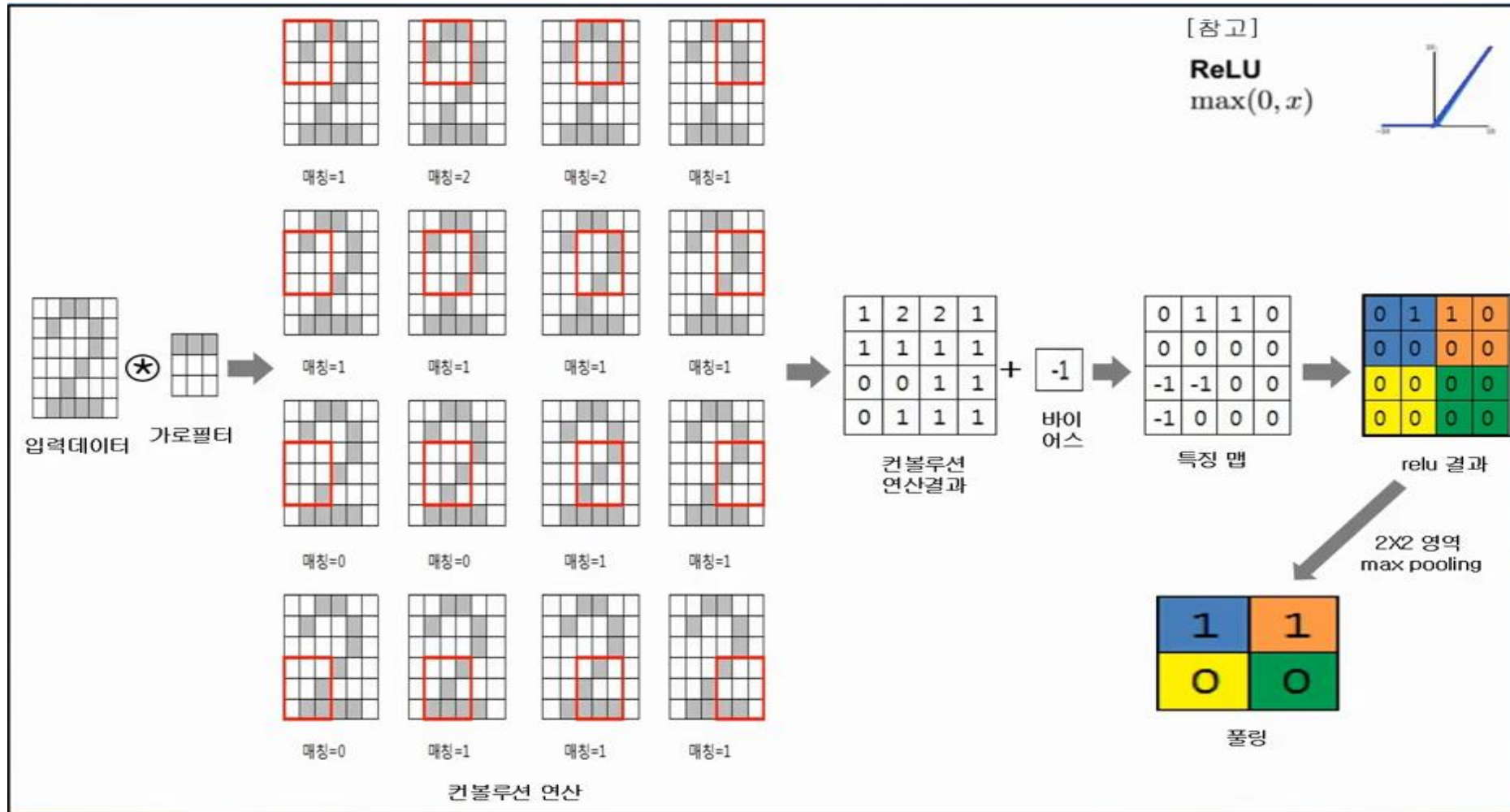
[참고] 입력데이터와 필터





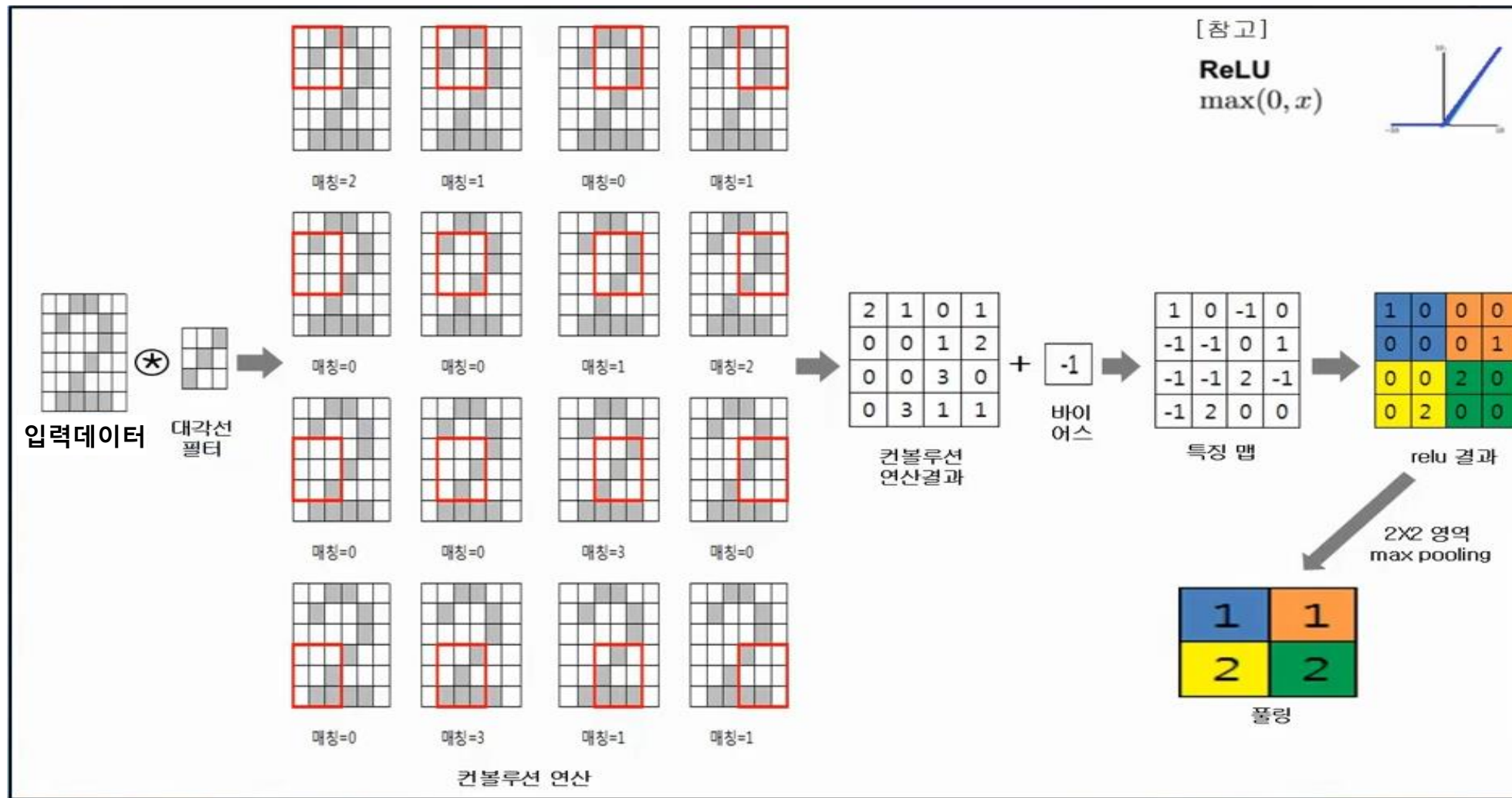
### 3. 필터를 통해 데이터 특징을 추출하는 원리

#### ❖ 가로 필터를 통한 입력 데이터 특징 추출(스트라이트 1, 패딩 없음)



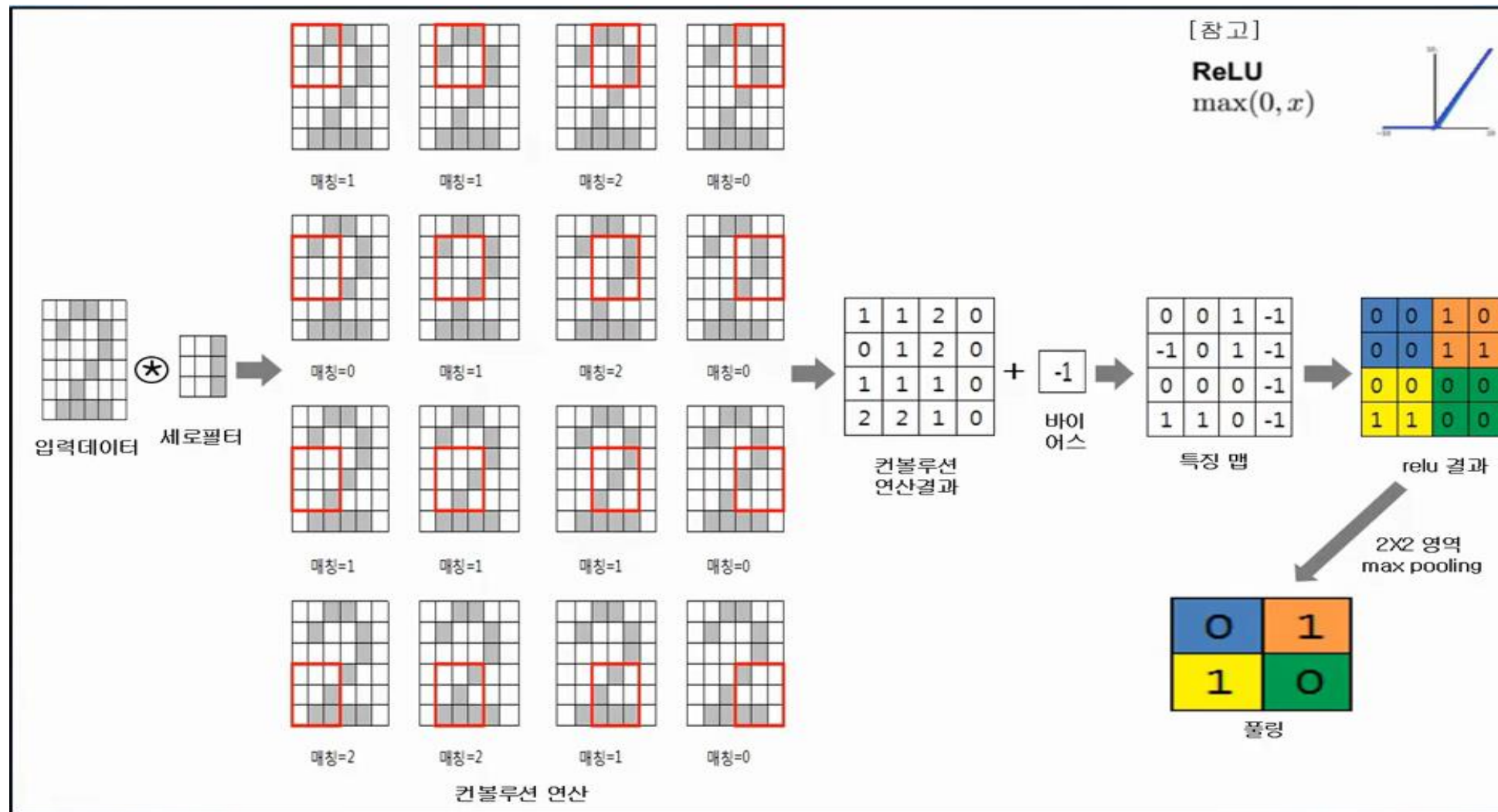
### 3. 필터를 통해 데이터 특징을 추출하는 원리

#### ❖ 대각선필터를 통한 데이터 특징 추출



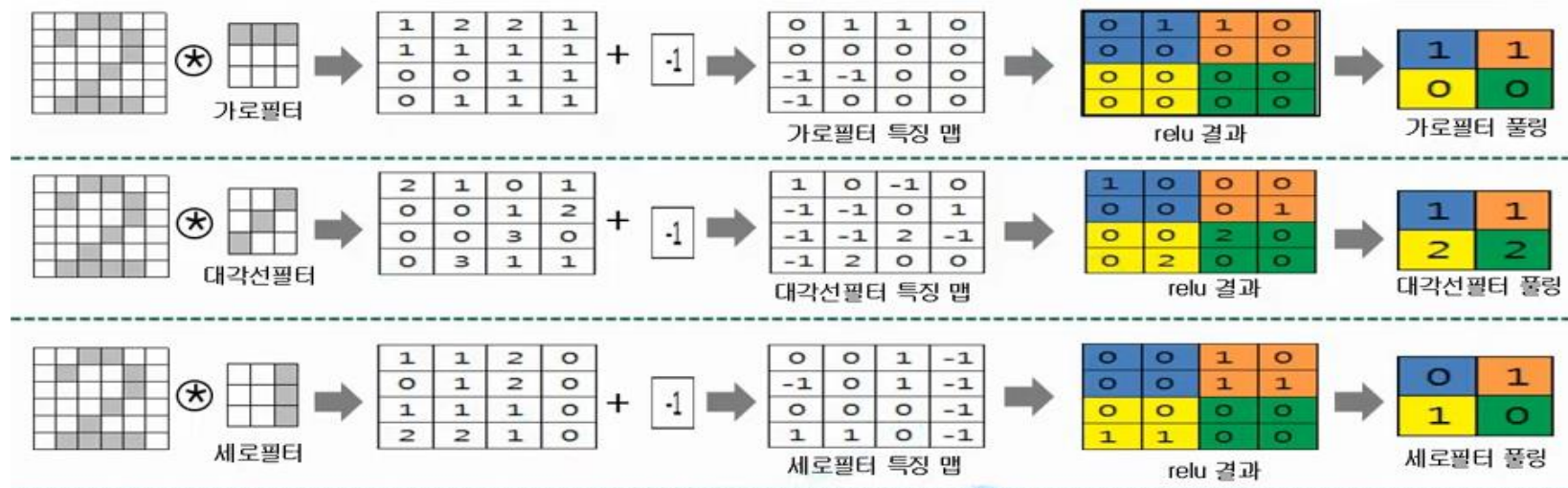
### 3. 필터를 통해 데이터 특징을 추출하는 원리

#### ❖ 새로 필터를 통한 입력 데이터 특징 추출



### 3. 필터를 통해 데이터 특징을 추출하는 원리

#### ❖ 필터를 통한 입력데이터 특징 추출원리-특징 맵이 압축된 풀링 값

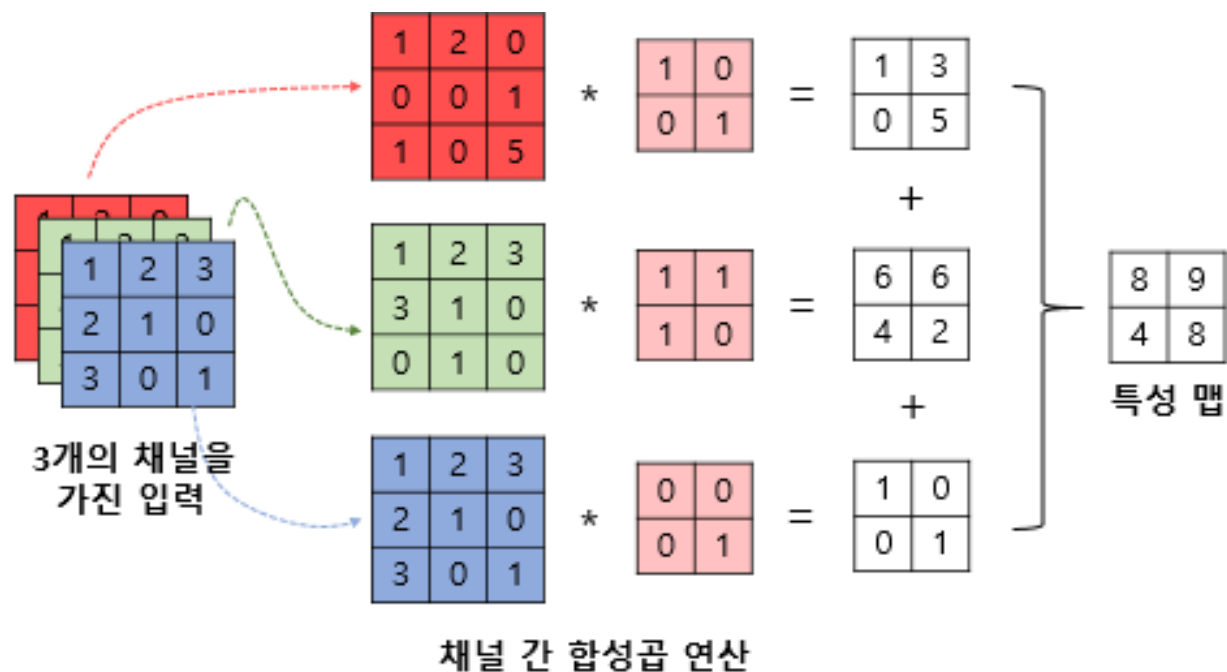


➤ 컨볼루션 연산 결과인 특징 맵(feature map) 값을 압축하고 있는 풀링 값을 보면,

- 대각선 필터에 대한 풀링 값이 가로와 세로필터의 풀링 값 보다 큰 값으로 구성되어 있는데,
- 풀링 값이 크다는 것은, 데이터 안에 해당 필터의 특징(성분)이 많이 포함되어 있는 것을 의미함. 즉, 특징 맵 값이 압축되어 있는 풀링 결과 값을 통해 데이터의 특징(성분)을 추출 할 수 있음
- 위의 예제를 보면, 입력 데이터 '2' 는 대각선 특징이 가로나 세로 특징보다 더욱 많이 포함되어 있으며 이러한 특징을 추출하는데 대각선 필터가 가로나 세로보다 유용하다는 것을 알 수 있음

### 3. 필터를 통해 데이터 특징을 추출하는 원리

#### ❖ 다수의 채널을 가질 경우의 합성곱 연산(3차원 텐서의 합성곱 연산)



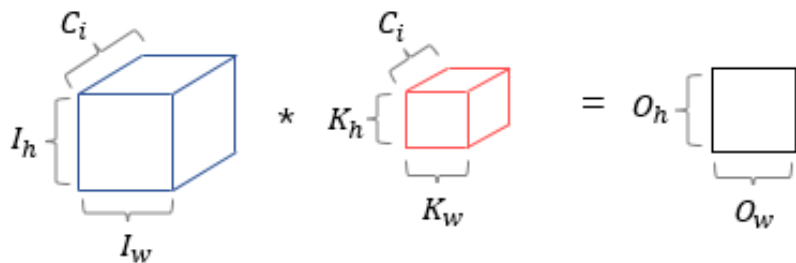


### 3. 필터를 통해 데이터 특징을 추출하는 원리

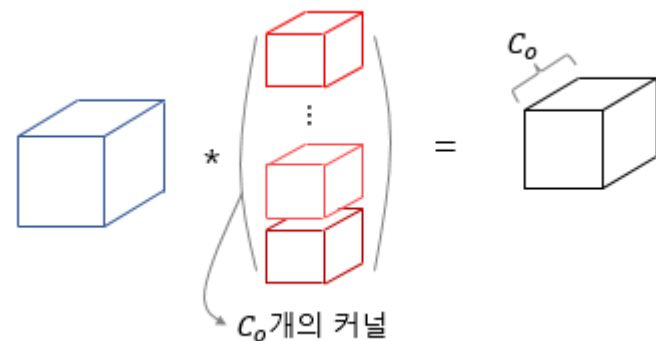
#### ❖ 3차원 텐서의 합성곱 연산

- 일반화를 위해 사용하는 각 변수가 의미하는 바는 다음과 같다

- $I_h$ : 입력의 높이
- $I_w$ : 입력의 너비
- $K_h$ : 커널의 높이
- $K_w$ : 커널의 너비
- $O_h$ : 특성 맵의 높이
- $O_w$ : 특성 맵의 너비
- $C_i$ : 입력 데이터의 채널



1개 커널 3차원 텐서의 합성곱 연산



다수 커널 사용 3차원 텐서의 합성곱 연산