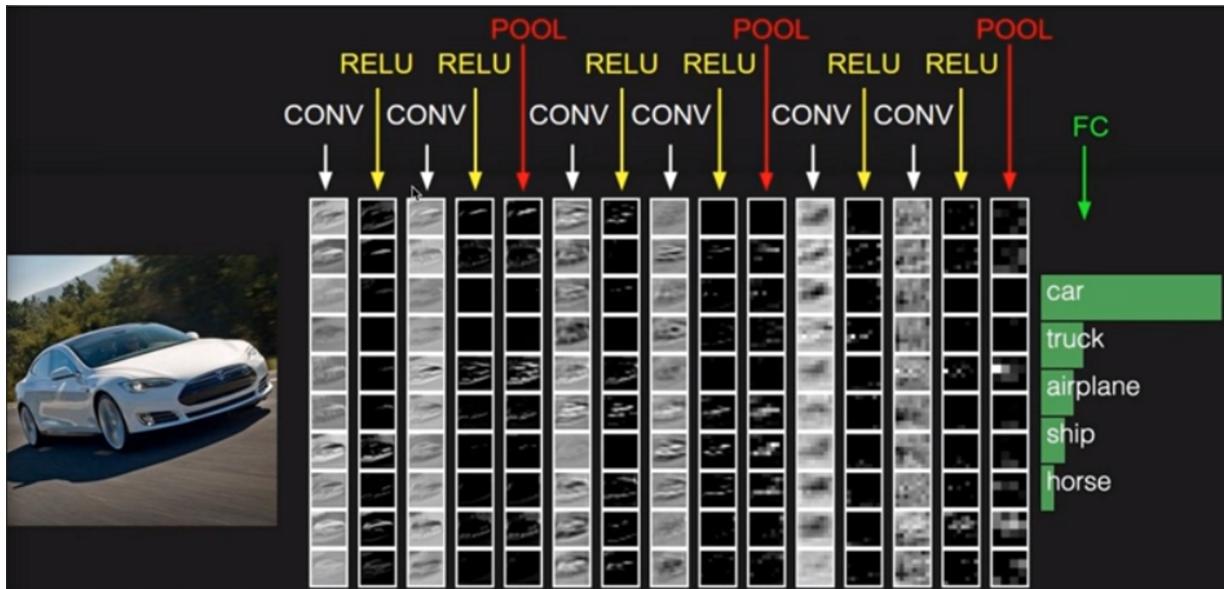


10. CNN (convolutional neural network)

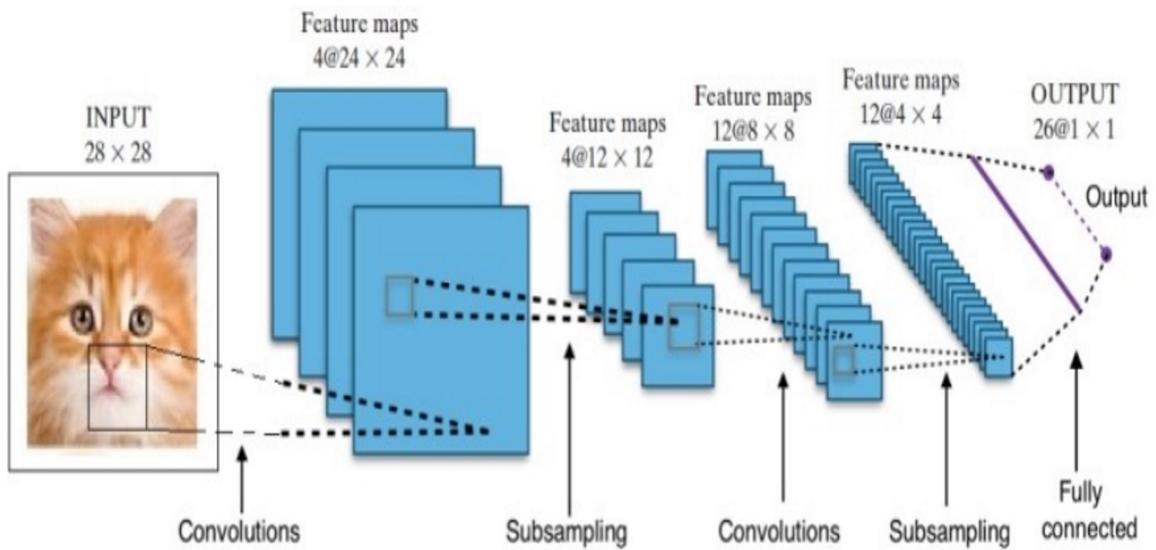
2019년 4월 8일 월요일 오후 5:18

1. Definition

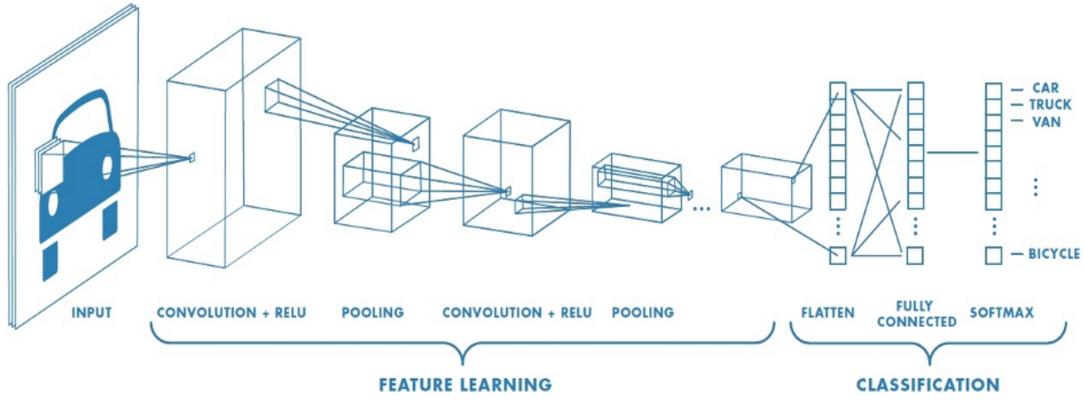
- 컨벌류션(합성곱) + 신경망 = CNN



- Input -> Conv -> Relu -> Conv -> Relu -> pool -> -> output
- 데이터량을 줄이고 왜곡시켜 신경망은 분류한다.



- $(X \times Y \times Z)$: X 행, Y 열, Z 채널
- 필터의 W 값을 임의로 변경해 convolution의 반복을 통해 여러 채널을 형성



- 필터(Filter) = 커널(Kernel)**

x1	x0	x1
x0	x1	x0
x1	x0	x1

필터는 **이미지의 특징을 찾아내기 위한 공용 파라미터**입니다.

Filter를 Kernel이라고 하기도 합니다.

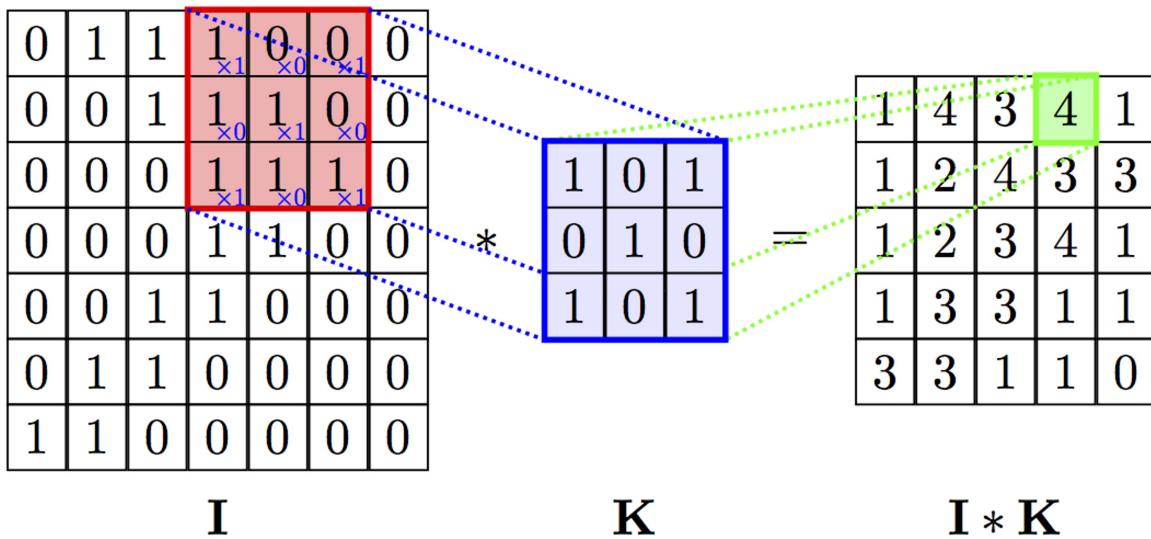
CNN에서 Filter와 Kernel은 같은 의미입니다.

필터는 일반적으로 (4, 4)이나 (3, 3)과 같은 **정사각 행렬**로 정의됩니다.

CNN에서 학습의 대상은 필터 파라미터 입니다.

사용자가 임의로 weight를 정한다.

- Convolution**



1	2	3	4	5	6	7
8	9	10	11	12	13	14
15	16	17	18	19	20	21
22	23	24	25	26	27	28
29	30	31	32	33	34	35
36	37	38	39	40	41	42
43	44	45	46	47	48	49

0.1	0.2	0.3
0.4	0.5	0.6
0.7	0.8	0.9

↓

$$\begin{aligned}
 &= 0.1 \times 10 + 0.2 \times 11 + 0.3 \times 12 \\
 &+ 0.4 \times 17 + 0.5 \times 18 + 0.6 \times 19 \\
 &+ 0.7 \times 24 + 0.8 \times 25 + 0.9 \times 26 \\
 &= 94.2
 \end{aligned}$$

1	1	1	0	0
0	1	1	1	0
0	0	1	1	1
0	0	1	1	0
0	1	1	0	0

Image

4		

Convolved Feature

1	1	1	0	0
0	1	1	1	0
0	0	1	1	1
0	0	1	1	0
0	1	1	0	0

Image

4	3	4
2	4	3
2	3	4

Convolved Feature

0	0	0	0	0	0	0
0	1	1	1	0	0	0
0	0	1	1	1	0	0
0	0	0	1	1	1	0
0	0	1	1	1	0	0
0	0	1	1	0	0	0
0	0	0	0	0	0	0

Input

x1	x0	x1
x0	x1	x0
x1	x0	x1

3x3 Window

2	2	3	1	1
1	4	3	4	1
2	2	4	3	3
1	2	3	4	1
1	2	3	1	1

Output

0	0	0
0	1	1
0	0	1

0	0	0
0	1	0
0	0	1

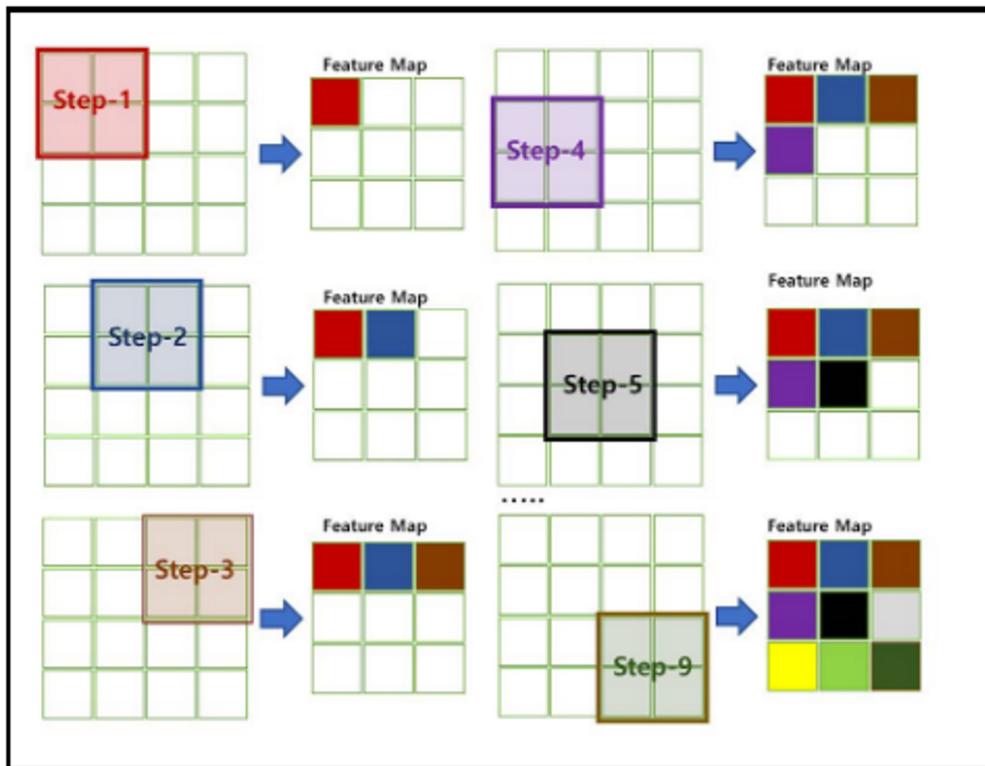
$$0+0+0+0+1+\dots+1=2$$

2	2
1	4

• 스트라이드(Stride)

필터는 입력 데이터를 지정한 간격으로 순회하면서 합성곱을 계산합니다.

여기서 지정된 간격으로 필터를 순회하는 간격을 Stride라고 합니다.

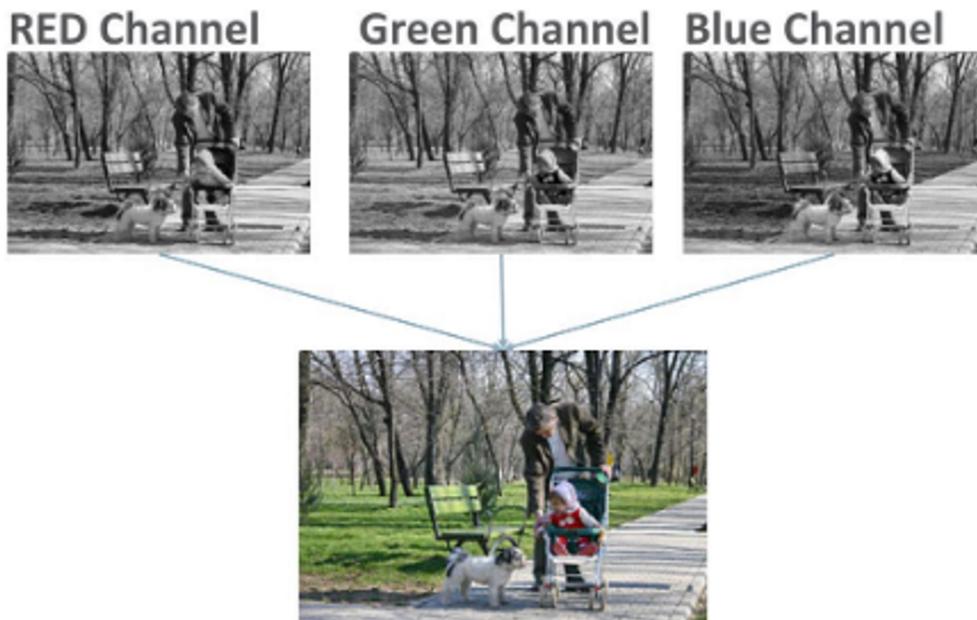


- 채널(Channel)

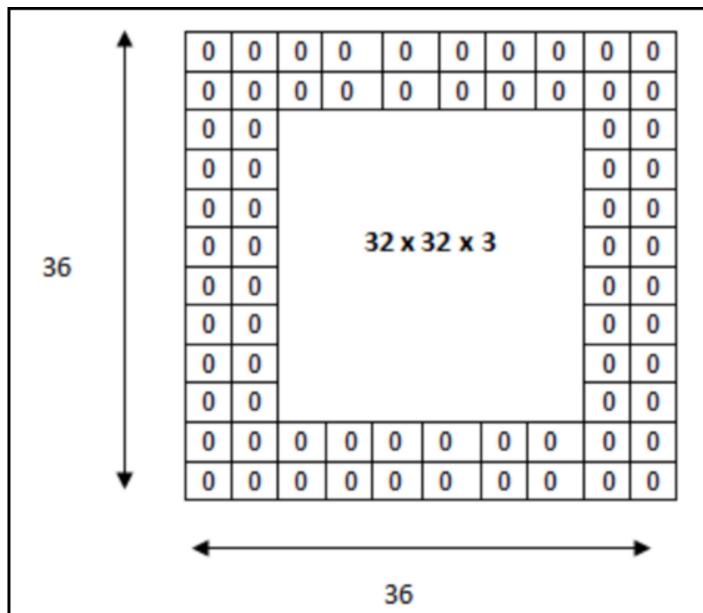
이미지 픽셀 하나하나는 실수입니다. 컬러 사진은 천연색을 표현하기 위해서, 각 픽셀을 RGB 3개의 실수로 표현한 3차원 데이터입니다. **컬러 이미지는 3개의 채널**로 구성됩니다.

반면에 흑백 명암만을 표현하는 **흑백 사진은 2차원 데이터로 1개 채널**로 구성됩니다.

높이 39 픽셀 **폭 31** 픽셀인 **컬러** 사진 데이터의 shape은 **(39, 31, 3)**으로 표현합니다. 반면에 **높이 39** 픽셀이고 **폭 31** 픽셀인 **흑백** 사진 데이터의 shape은 **(39, 31, 1)**입니다.



- 패딩(Padding)



(32, 32, 3) 데이터를 외각에 2 pixel을 추가하여 (36, 36, 3) 행렬을 만드는 예제입니다. Padding을 통해서 1. Convolution 레이어의 출력 데이터의 사이즈를 조절하는 기능이 외에, 외각을 "0"값으로 둘러싸는 특징으로 부터 2. 인공 신경망이 이미지의 외각을 인식하는 학습 효과도 있습니다.

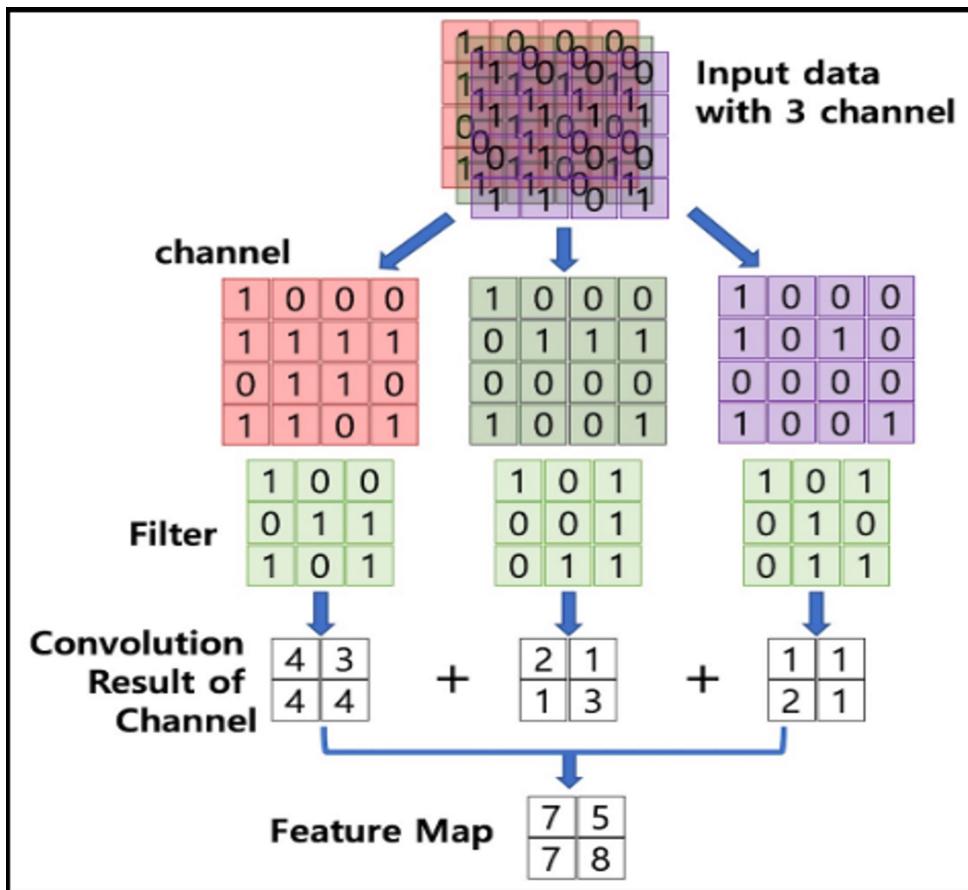
- **피처 맵(Feature Map) = 액티베이션 맵(Accivation Map)**

입력 1. 데이터가 여러 채널을 갖을 경우 필터는 각 2.채널을 순회하며 합성곱을 계산한 후, 3.채널별 피처 맵을 만듭니다. 그리고 4.각 채널의 피처 맵을 합산하여 5.최종 피처 맵으로 반환합니다.

1. 입력 데이터는 채널 수와 상관없이 2.필터 별로 1개의 피처 맵이 만들어 집니다.

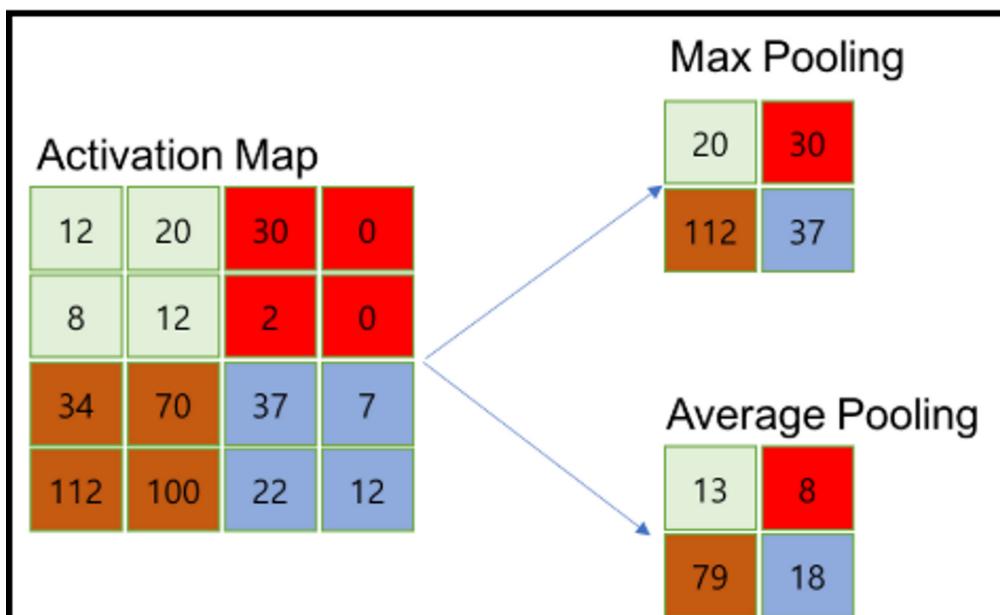
1.하나의 Convolution Layer에 2.크기가 같은 여러 개의 필터를 적용할 수 있습니다. 이 경우에 Feature Map에는 3.필터 갯수 만큼의 채널이 만들어집니다. 4.입력데이터에 적용한 필터의 개수는 5.출력 데이터인 Feature Map의 채널이 됩니다.

Convolution Layer의 입력 데이터를 필터가 순회하며 합성곱을 통해서 만든 출력을 Feature Map 또는 Activation Map이라고 합니다. Feature Map은 합성곱 계산으로 만들어진 행렬입니다. Activation Map은 Feature Map 행렬에 활성 함수를 적용한 결과입니다. 즉, Convolution 레이어의 최종 출력 결과가 Activation Map입니다.



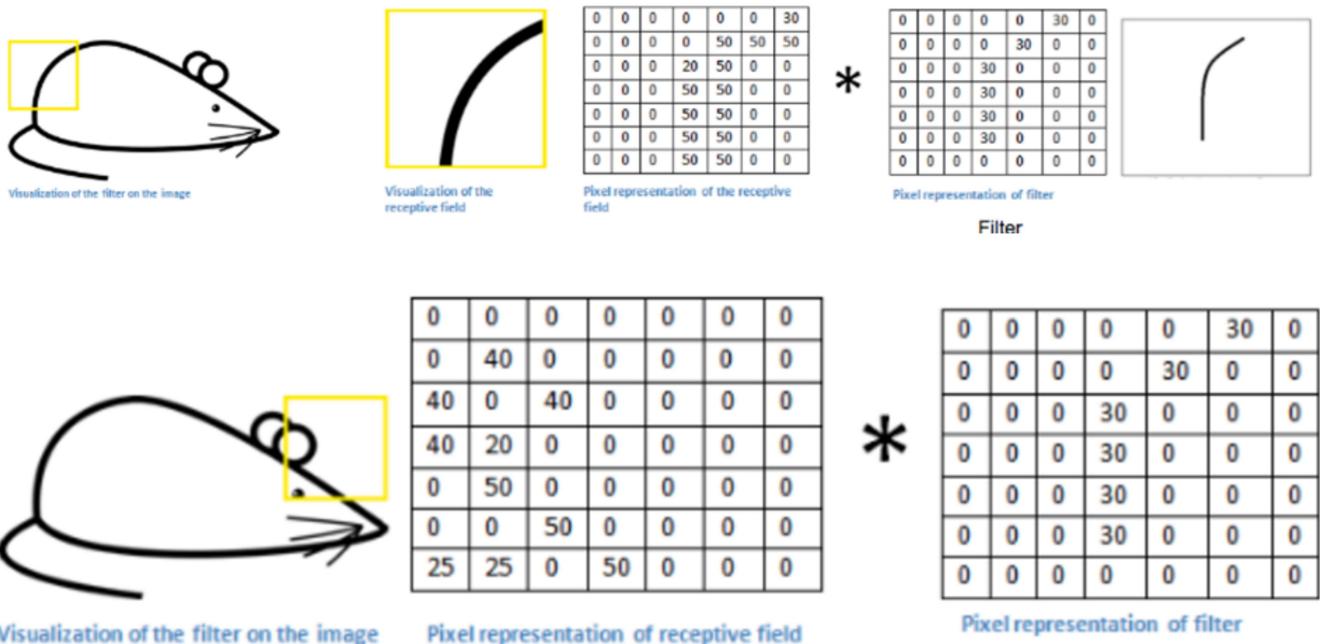
- 풀링(Pooling) 레이어

풀링 레이어는 컨볼류션 레이어의 **출력 데이터를 입력으로 받아서 출력 데이터(Activation Map)의 크기를 줄이거나 특정 데이터를 강조하는 용도로 사용됩니다.** 풀링 레이어를 처리하는 방법으로는 Max Pooling과 Average Pooling, Min Pooling이 있습니다. 정사각 행렬의 특정 영역 안에 값의 최댓값을 모으거나 특정 영역의 평균을 구하는 방식으로 동작합니다.



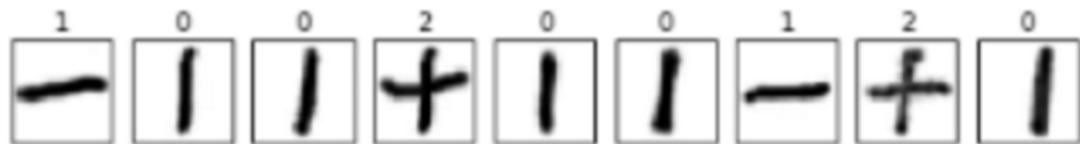
- Pooling 레이어는 Convolution 레이어와 비교하여 다음과 같은 특징이 있습니다.
 - 학습대상 파라미터가 없음 (필터X)
 - Pooling 레이어를 통과하면 행렬의 크기 감소

c. Pooling 레이어를 통해서 채널 수 변경 없음



다중 필터의 적용

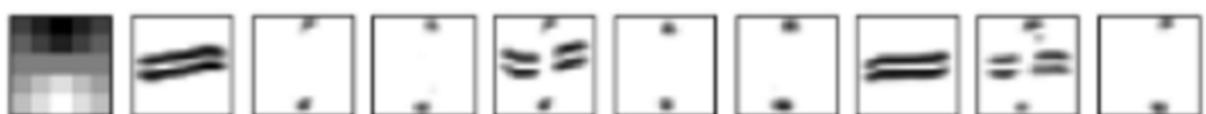
입력값에는 여러가지 특징이 있기 때문에 하나의 필터가 아닌 여러개의 다중 필터를 같이 적용하게 된다. 다음과 같이 |, +, - 모양을 가지고 있는 데이터가 있다고 하자



각 데이터가 |와 -의 패턴(특징을) 가지고 있는지를 파악하기 위해서 먼저 |(세로) 필터를 적용해보면 다음과 같은 결과가 나온다.



(맨 앞의 상자는 필터이다.) 두 번째 상자부터 원본 이미지에 세로선(|)이 없는 경우 결과 이미지에 출력이 없고, 세로선이 있는 경우에는 결과 이미지에 세로 선이 있는 것을 확인할 수 있다.
마찬가지로 가로선(-) 특징이 있는지 가로 선을 추출하는 필터를 적용해보면 다음과 같은 결과를 얻을 수 있다.



이렇게 각기 다른 특징을 추출하는 필터를 조합하여 네트워크에 적용하면, 원본 데이터가 어떤 형태의 특징을 가지고 있는지 없는지를 판단해 낼 수 있다. 다음은 하나의 입력 데이터에 앞서 적용한 세로와 가로선에 대한 필터를 동시에 적용한 네트워크의 모양이다.

- 2. Hypothesis**
- 3. Cost function**
- 4. Gradient descent algorithm**