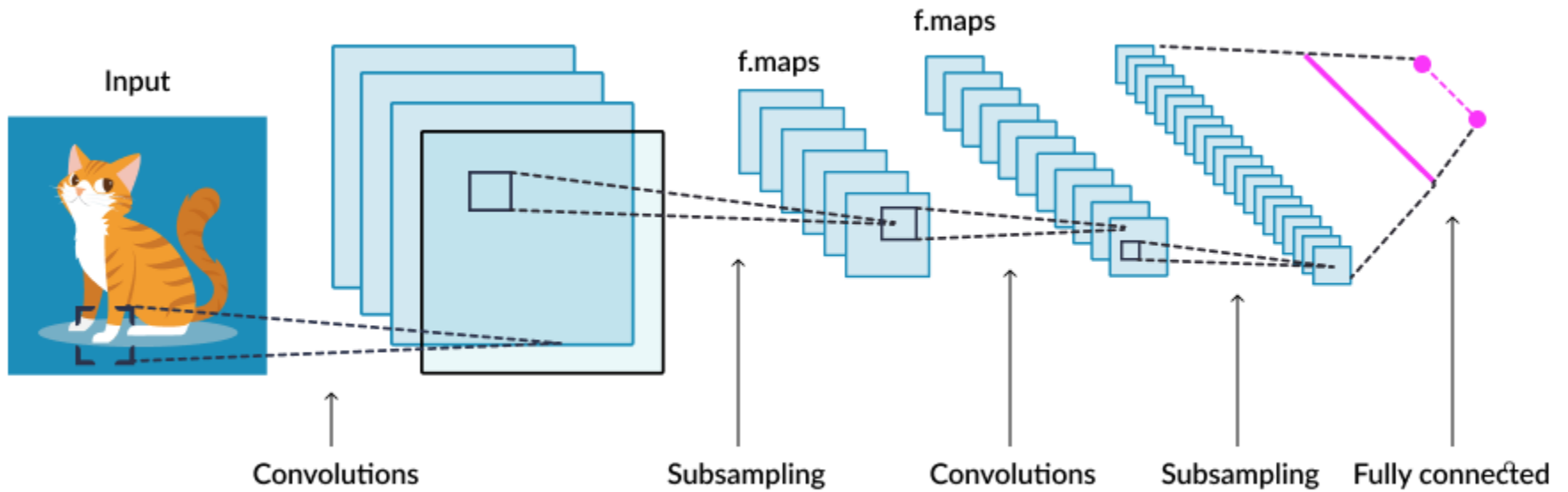
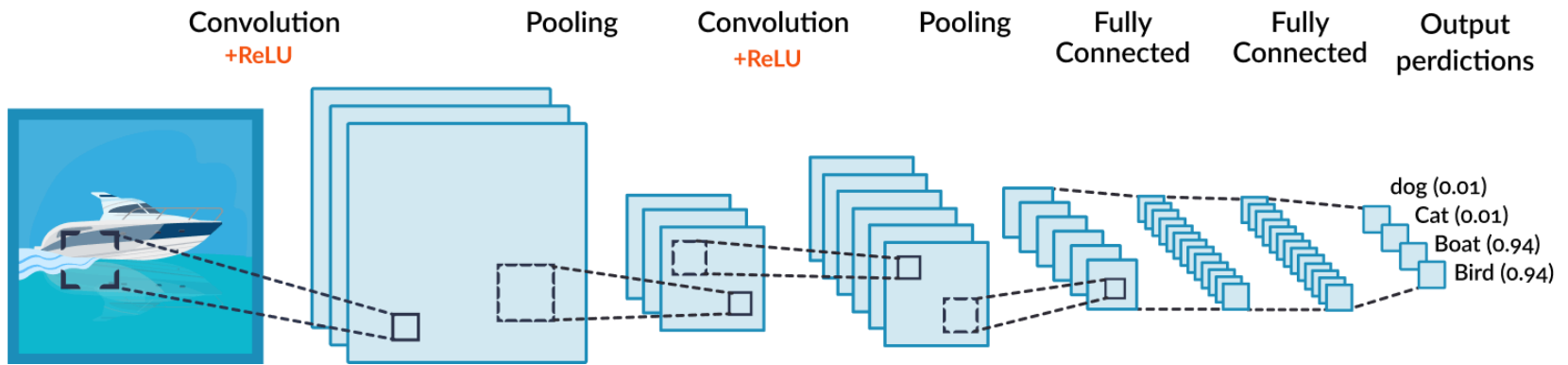


Convolutional Neural Network

기본 개념



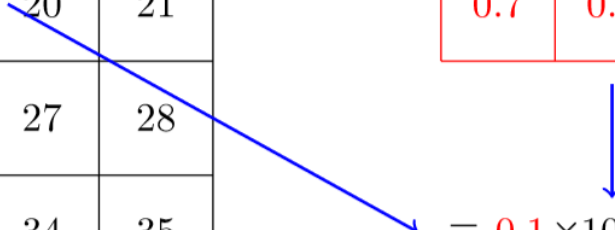
CNN의 주요 용어 정리

- Convolution(합성곱)
- 채널(Channel)
- 필터(Filter)
- 스트라이드(Strid)
- 피쳐 맵(Feature Map)
- 액티베이션 맵(Activation Map)
- 패딩(Padding)
- 풀링(Pooling) 레이어

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


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

Smoothing Spatial filters

$$\frac{1}{9} \times$$

1	1	1
1	1	1
1	1	1

Box filter

$$\frac{1}{16} \times$$

1	2	1
2	4	2
1	2	1

Weighted average

$$\frac{1}{273}$$

1	4	7	4	1
4	16	26	16	4
7	26	41	26	7
4	16	26	16	4
1	4	7	4	1

gaussian smoothing filter



$\frac{1}{9}$

1	1	1
1	1	1
1	1	1



Convolution의 작동 원리

- 입력 데이터 5X5, 필터 크기 3X3
- Stride = 1

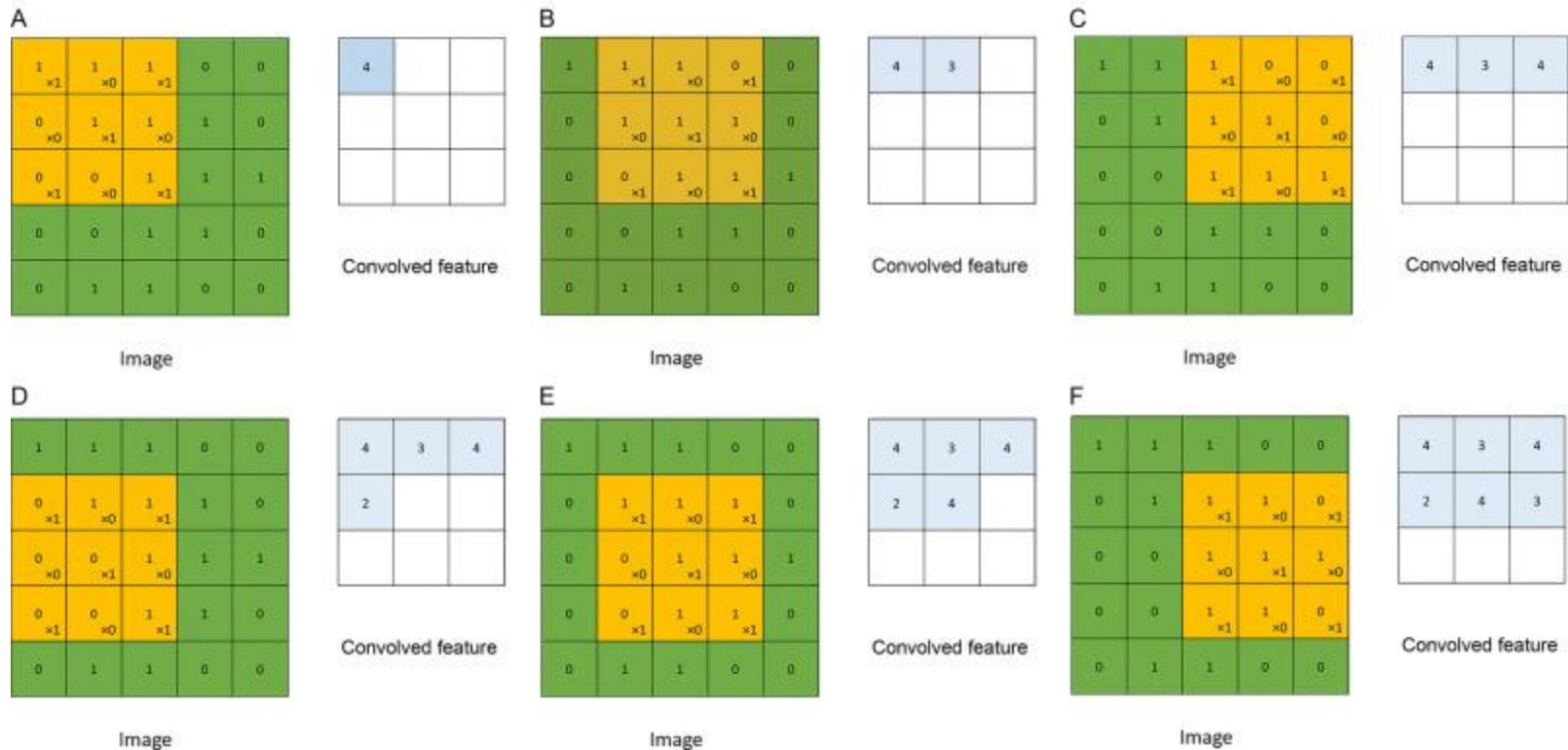
1 _{x1}	1 _{x0}	1 _{x1}	0	0
0 _{x0}	1 _{x1}	1 _{x0}	1	0
0 _{x1}	0 _{x0}	1 _{x1}	1	1
0	0	1	1	0
0	1	1	0	0

Image

4		

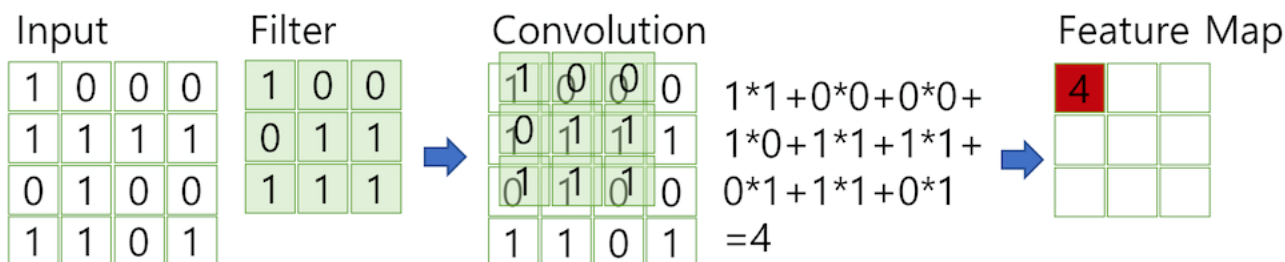
Convolved
Feature

Convolution의 작동 원리



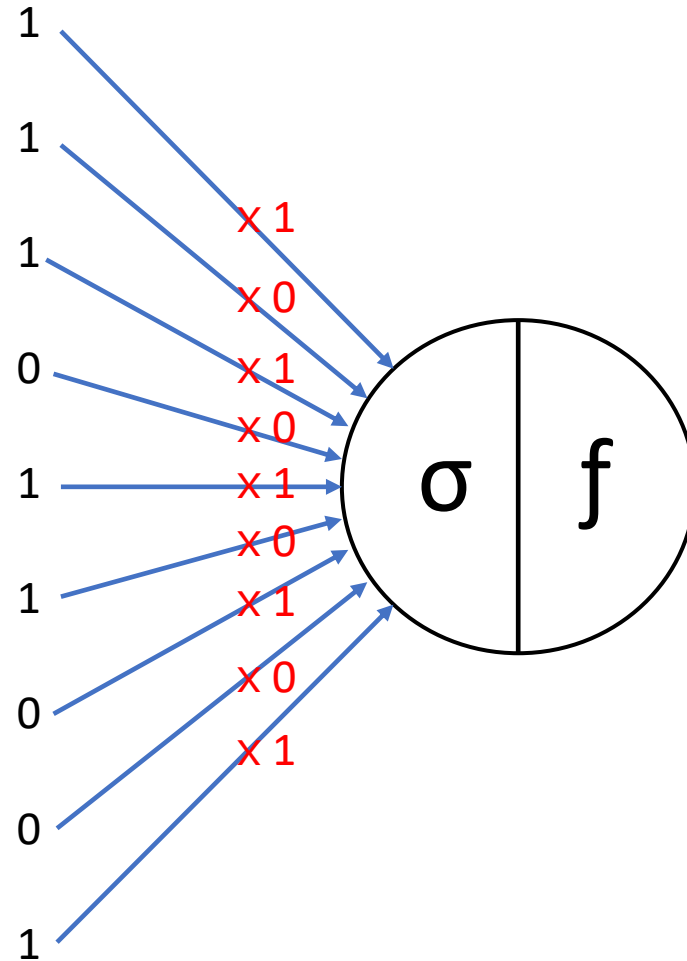
필터(Filter) & Stride

- 필터는 이미지의 특징을 찾아내기 위한 공용 파라미터입니다.
- 필터는 일반적으로 (3, 3)과 같은 정사각 행렬로 정의됩니다.
- CNN에서 학습의 대상은 필터 파라미터 입니다.



CNN에서 학습의 대상은 필터 파라미터

1 _{x1}	1 _{x0}	1 _{x1}	0	0
0 _{x0}	1 _{x1}	1 _{x0}	1	0
0 _{x1}	0 _{x0}	1 _{x1}	1	1
0	0	1	1	0
0	1	1	0	0



채널, Channel

Keras data_format: A string, one of `channels_last` (default) or `channels_first`. The ordering of the dimensions in the inputs. `channels_last` corresponds to inputs with shape `(batch_size, height, width, channels)`

RED Channel



Green Channel

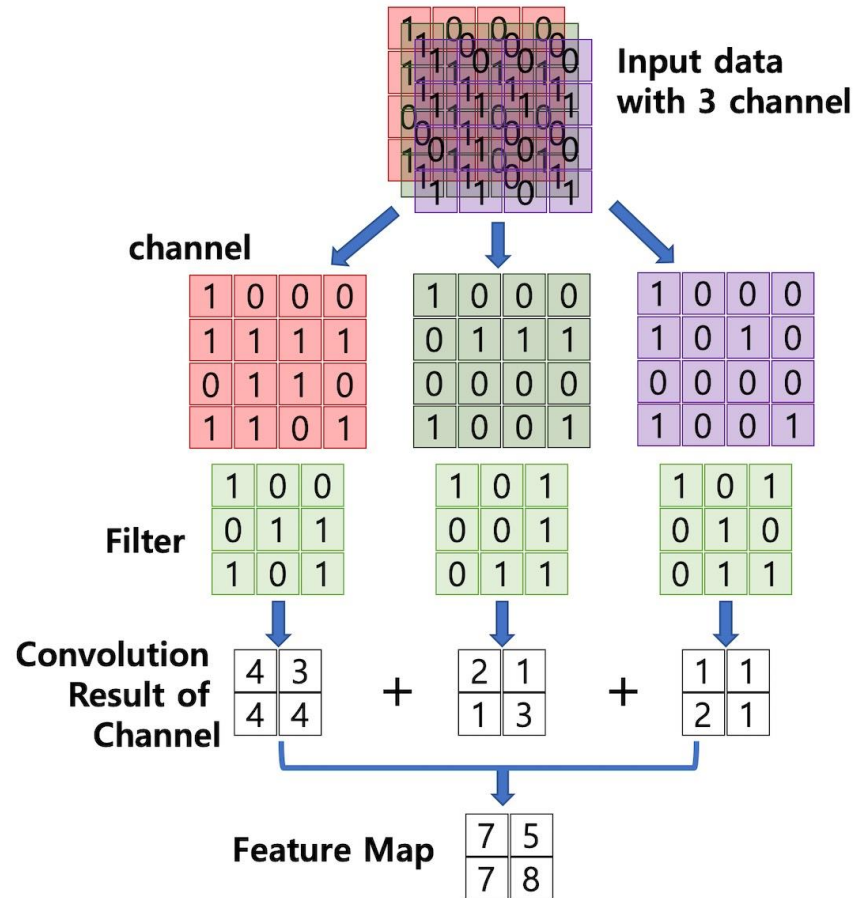


Blue Channel



이미지 출처: [https://en.wikipedia.org/wiki/Channel_\(digital_image\)](https://en.wikipedia.org/wiki/Channel_(digital_image))

컬러 영상 (3-channel) 에서 Feature Map 생성 과정



패딩(Padding)

- Convolution 레이어에서 Filter와 Stride에 작용으로 Feature Map 크기는 입력데이터 보다 작습니다.
- Convolution 레이어의 출력 데이터가 줄어드는 것을 방지하는 방법이 패딩입니다
- 패딩은 입력 데이터의 외각에 지정된 픽셀만큼 특정 값으로 채워 넣는 것을 의미합니다. 보통 패딩 값으로 0으로 채워 넣습니다.

패딩(Padding)

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

0	0	0	0	0	0	0
0	60	113	56	139	85	0
0	73	121	54	84	128	0
0	131	99	70	129	127	0
0	80	57	115	69	134	0
0	104	126	123	95	130	0
0	0	0	0	0	0	0

Kernel

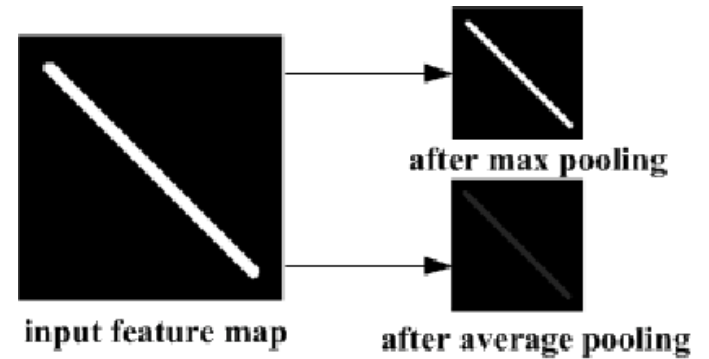
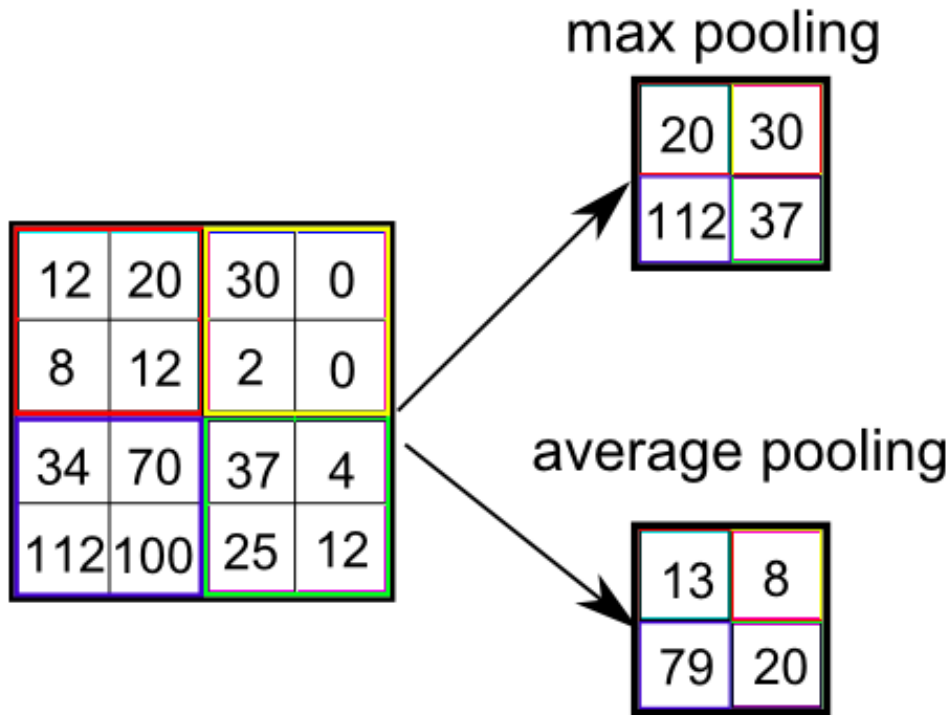
0	-1	0
-1	5	-1
0	-1	0

114				

Pooling 레이어

- 풀링 레이어는 컨볼루션 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.

Pooling 레이어



Conv2D layer

Conv2D class

```
tf.keras.layers.Conv2D(  
    filters,  
    kernel_size,  
    strides=(1, 1),  
    padding="valid",  
    data_format=None,  
    dilation_rate=(1, 1),  
    groups=1,  
    activation=None,  
    use_bias=True,  
    kernel_initializer="glorot_uniform",  
    bias_initializer="zeros",  
    kernel_regularizer=None,  
    bias_regularizer=None,  
    activity_regularizer=None,  
    kernel_constraint=None,  
    bias_constraint=None,  
    **kwargs  
)
```


Conv2D layer

Arguments

- **filters:** Integer, the dimensionality of the output space (i.e. the number of output filters in the convolution).
- **kernel_size:** An integer or tuple/list of 2 integers, specifying the height and width of the 2D convolution window. Can be a single integer to specify the same value for all spatial dimensions.
- **strides:** An integer or tuple/list of 2 integers, specifying the strides of the convolution along the height and width. Can be a single integer to specify the same value for all spatial dimensions. Specifying any stride value $\neq 1$ is incompatible with specifying any `dilation_rate` value $\neq 1$.
- **padding:** one of "valid" or "same" (case-insensitive). "valid" means no padding. "same" results in padding evenly to the left/right or up/down of the input such that output has the same height/width dimension as the input.
- **data_format:** A string, one of `channels_last` (default) or `channels_first`. The ordering of the dimensions in the inputs. `channels_last` corresponds to inputs with shape `(batch_size, height, width, channels)` while `channels_first` corresponds to inputs with shape `(batch_size, channels, height, width)`. It defaults to the `image_data_format` value found in your Keras config file at `~/.keras/keras.json`. If you never set it, then it will be `channels_last`.

Conv2D layer

Examples

```
>>> # The inputs are 28x28 RGB images with `channels_last` and the batch
>>> # size is 4.
>>> input_shape = (4, 28, 28, 3)
>>> x = tf.random.normal(input_shape)
>>> y = tf.keras.layers.Conv2D(
... 2, 3, activation='relu', input_shape=input_shape[1:])(x)
>>> print(y.shape)
(4, 26, 26, 2)
```

CNN 의 특징 정리

- 각 레이어의 입출력 데이터의 형상 유지
- 이미지의 공간 정보를 유지하면서 인접 이미지와의 특징을 효과적으로 인식
- 복수의 필터로 이미지의 특징 추출 및 학습
- 추출한 이미지의 특징을 모으고 강화할 수 있는 Pooling 레이어를 사용
- 필터를 공유 파라미터로 사용하기 때문에, 일반 인공 신경망과 비교하여 학습 파라미터가 매우 적음