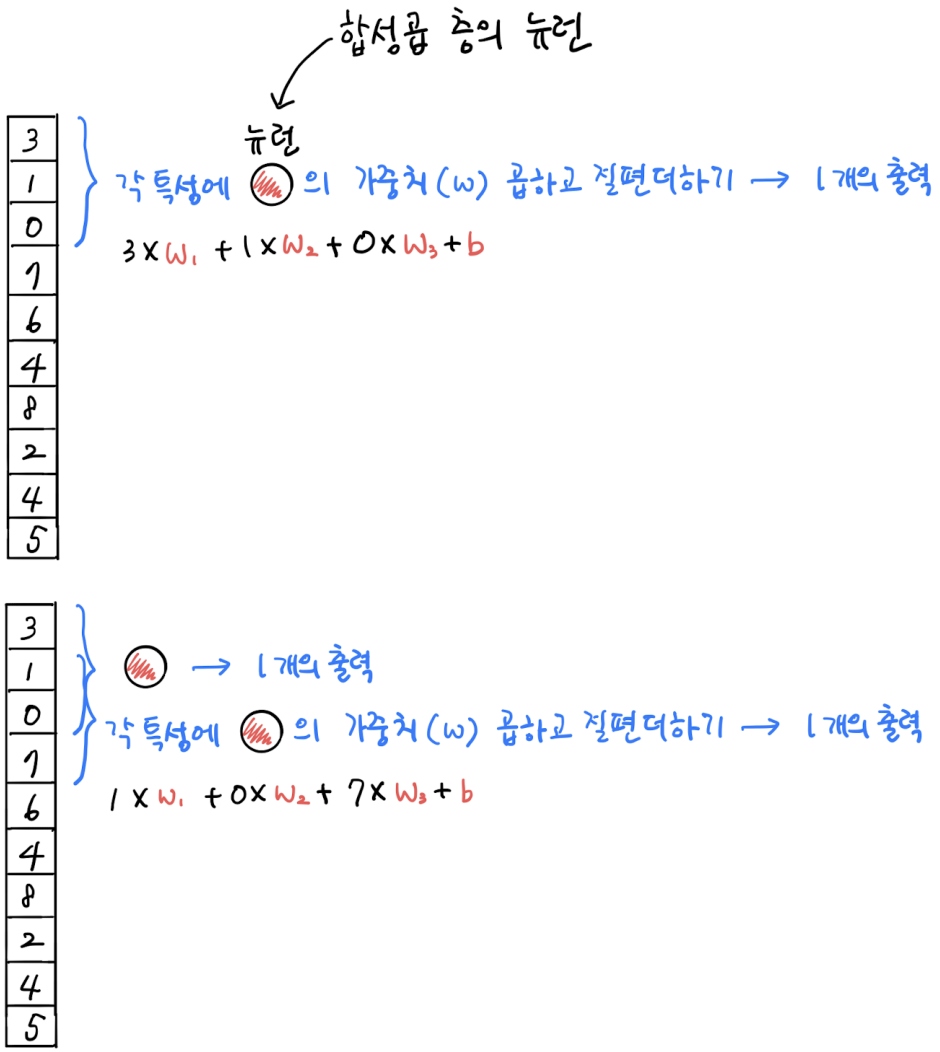
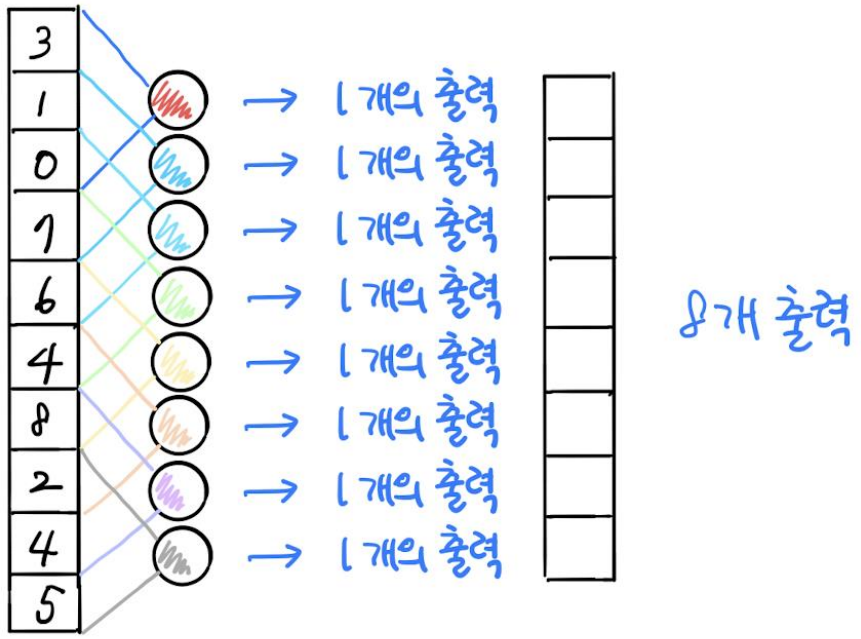
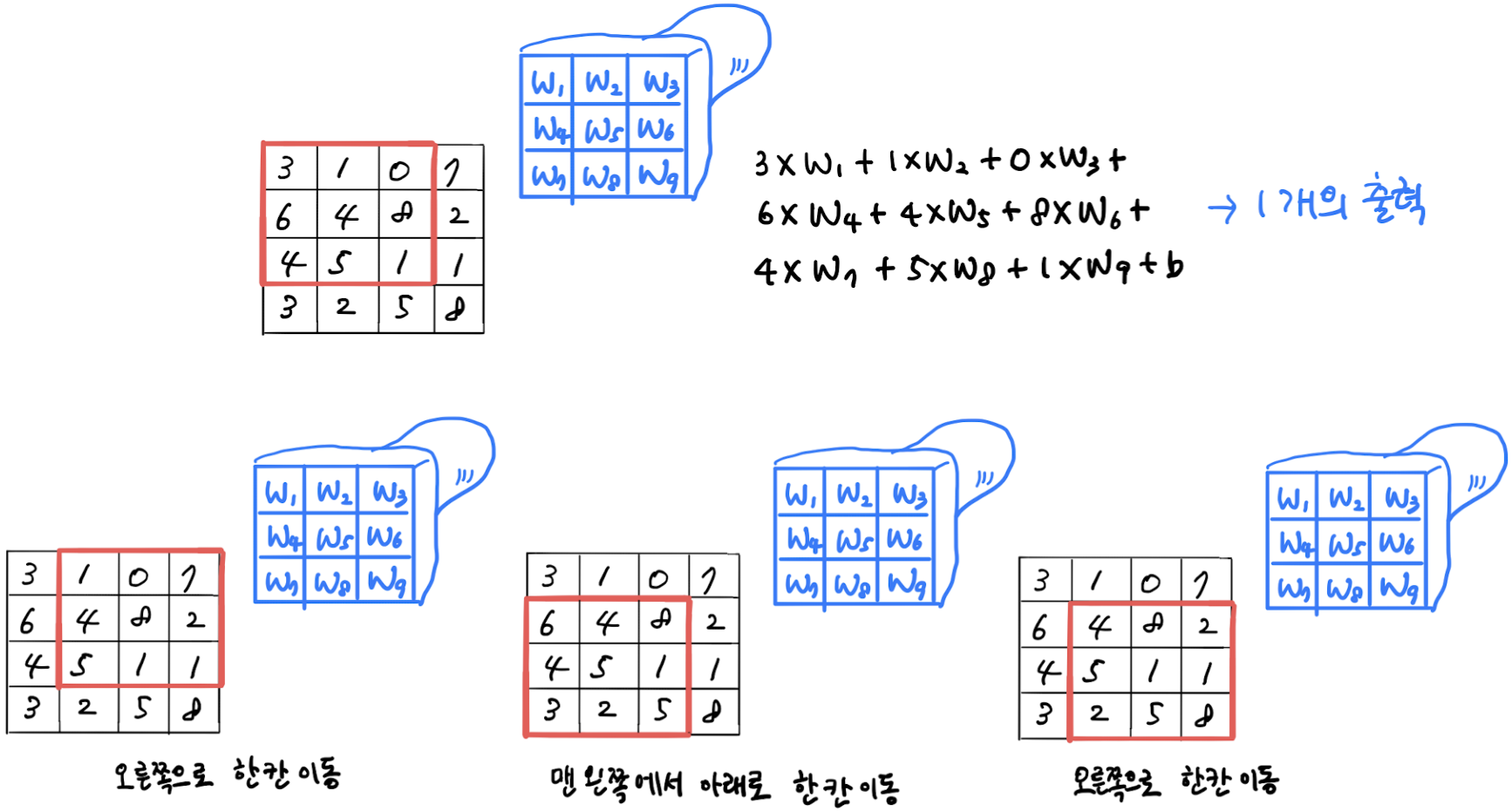
합성곱

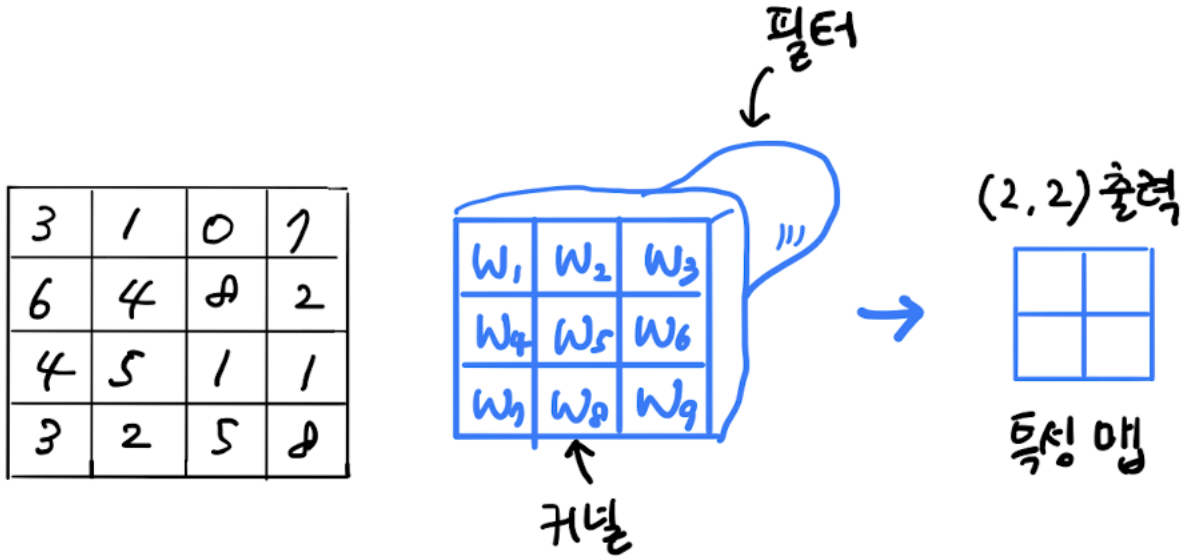

합성곱 층의 뉴런
뉴런
3
1
0
1
6
4
8
2
4
5
각 특성에 의 가중치(w) 곱하고 절편더하기 → 1개의 출력
3×w₁ + 1×w₂ + 0×w₃ + b
→ 1개의 출력
각 특성에 의 가중치(w) 곱하고 절편더하기 → 1개의 출력
1×w₁ + 0×w₂ + 7×w₃ + b


3
1
0
1
6
4
8
2
4
5
→ 1개의 출력
8개 출력

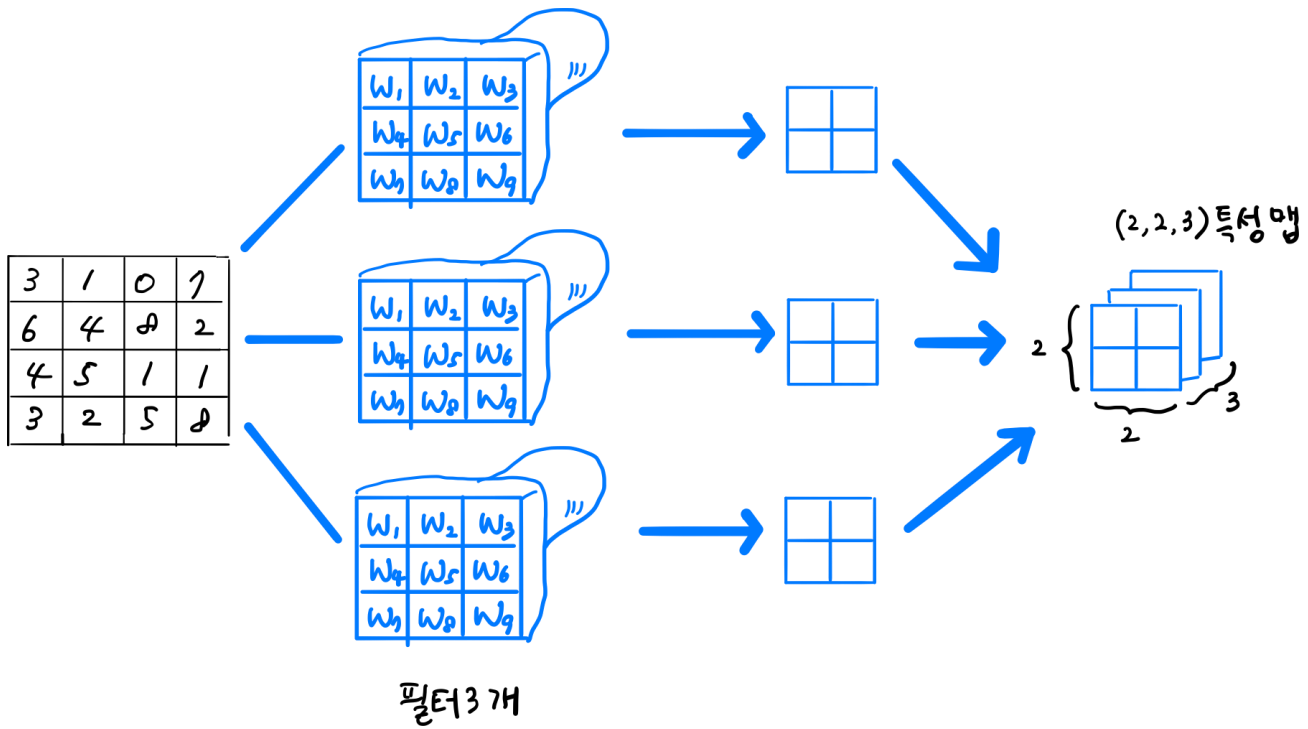
2차원 합성곱


3 1 0 1
6 4 8 2
4 5 1 1
3 2 5 8
W1 W2 W3
W4 W5 W6
W7 W8 W9
3×W1 + 1×W2 + 0×W3 +
6×W4 + 4×W5 + 8×W6 +
4×W7 + 5×W8 + 1×W9 + b
→ 1개의 출력
오른쪽으로 한칸 이동
맨 왼쪽에서 아래로 한칸 이동
오른쪽으로 한칸 이동

특성 맵



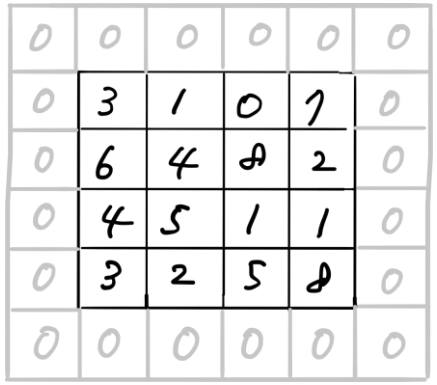
여러 개의 필터

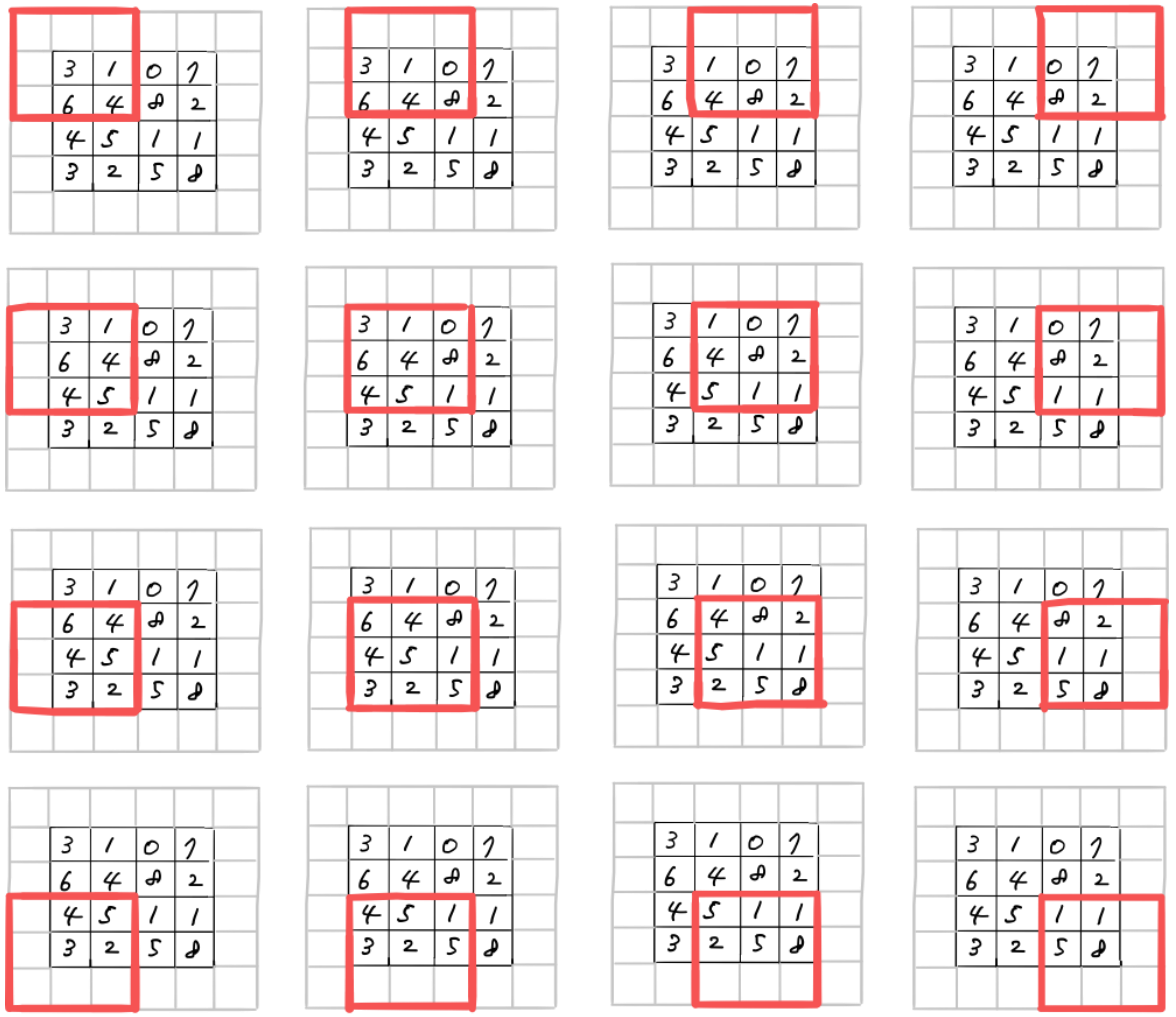


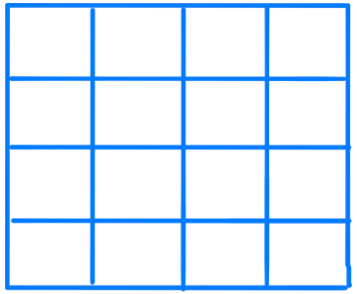
케라스 합성곱 층

```
from tensorflow import keras
keras.layers.Conv2D(10, kernel_size=(3, 3), activation='relu')
```

패딩

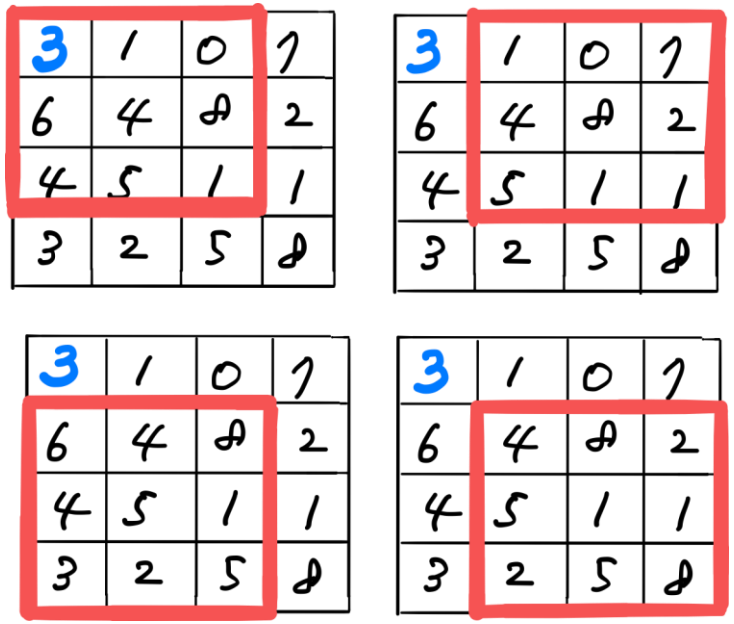


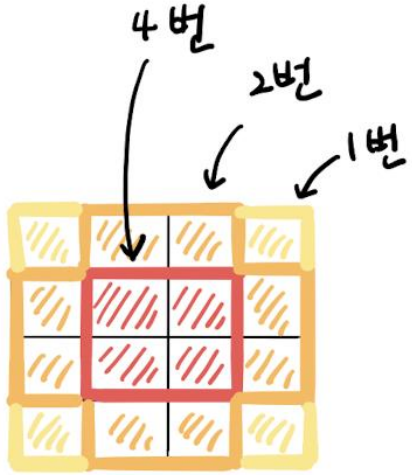


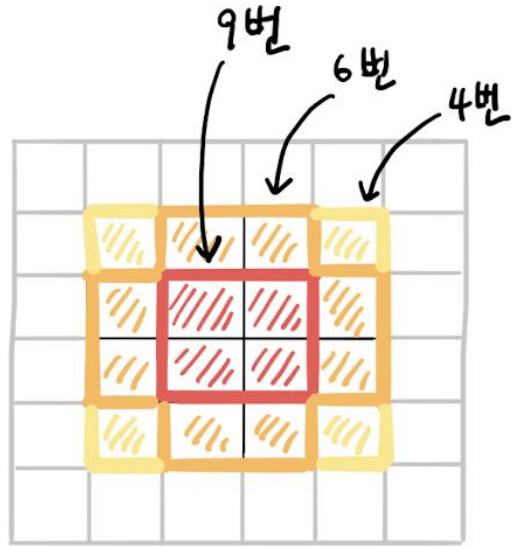


(4,4) 특성맵

패딩의 목적


3 1 0 1
6 4 8 2
4 5 1 1
3 2 5 8
3 1 0 1
6 4 8 2
4 5 1 1
3 2 5 8
3 1 0 1
6 4 8 2
4 5 1 1
3 2 5 8
3 1 0 1
6 4 8 2
4 5 1 1
3 2 5 8

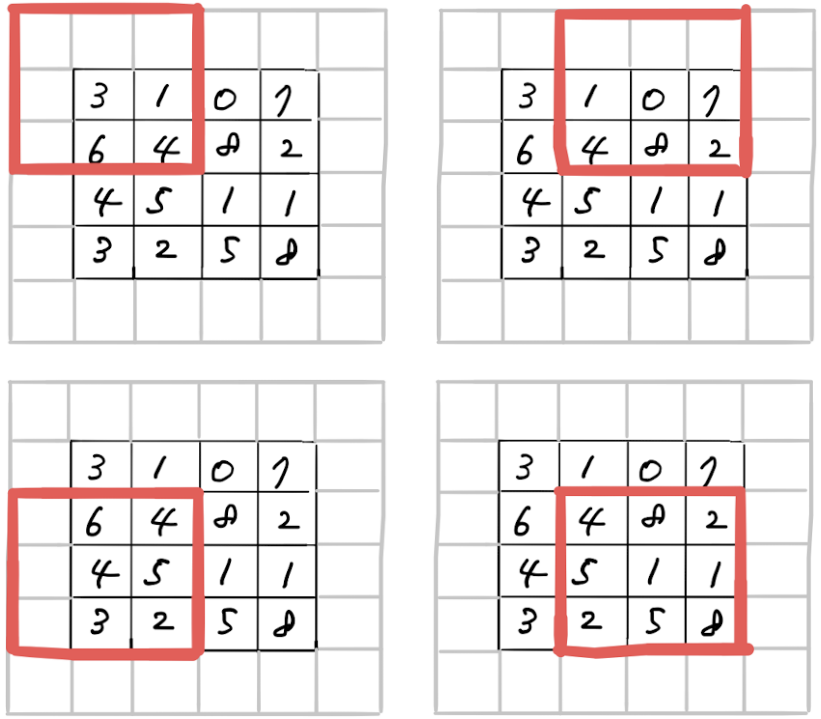

4번
2번
1번


9번
6번
4번

케라스의 패딩 설정

```
keras.layers.Conv2D(10, kernel_size=(3, 3), activation='relu', padding='same')
```

스트라이드



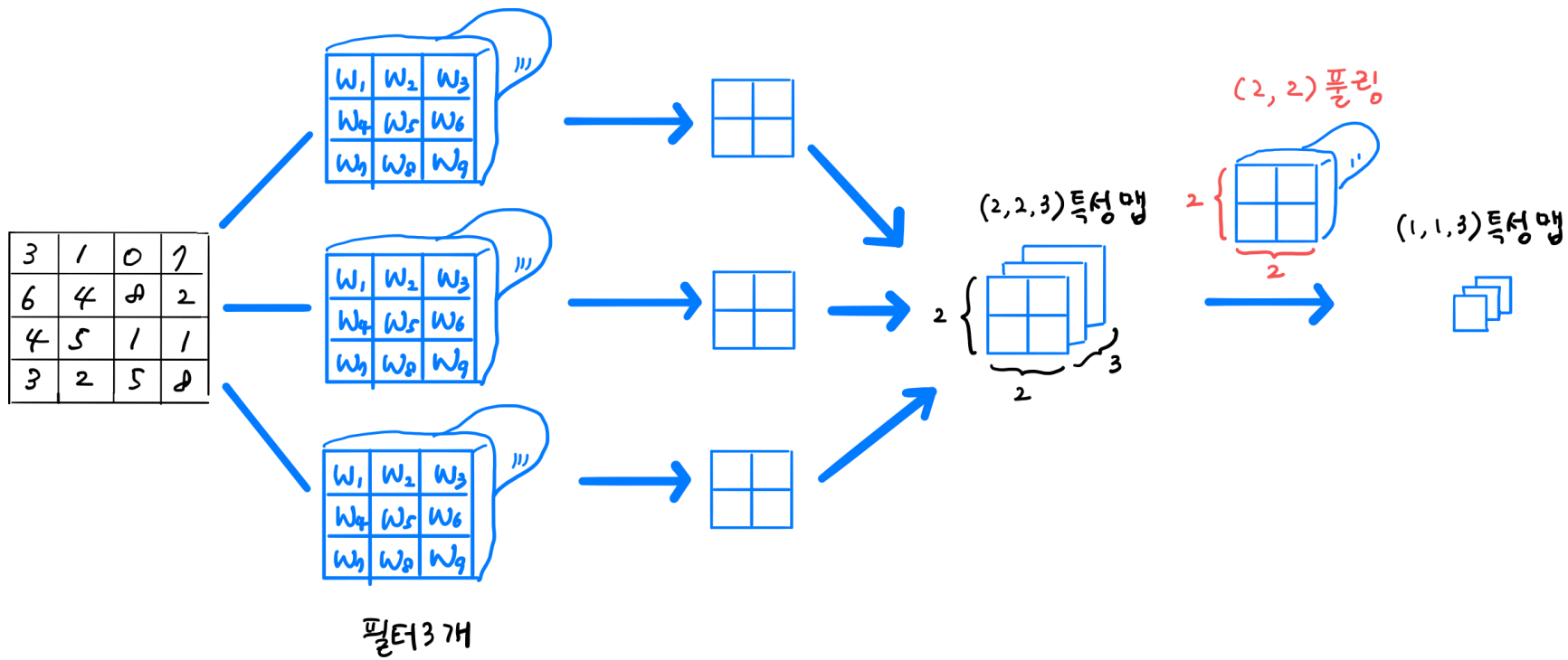
(2,2) 출력

특성 맵

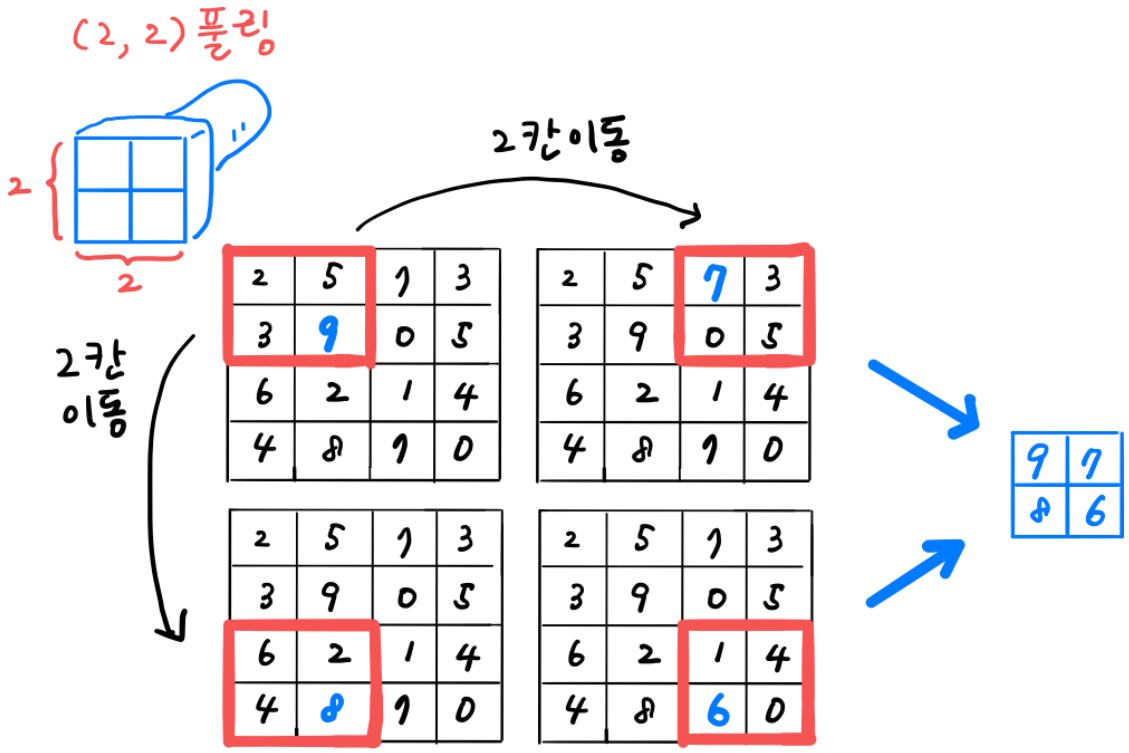
케라스의 스트라이드 설정

```
keras.layers.Conv2D(10, kernel_size=(3, 3), activation='relu',
                    padding='same', strides=1)
```

풀링


3 1 0 1
6 4 0 2
4 5 1 1
3 2 5 0
W1 W2 W3
W4 W5 W6
W7 W8 W9
필터3개
(2,2,3)특성맵
2
2
3
(2, 2) 풀링
2
2
(1,1,3)특성맵

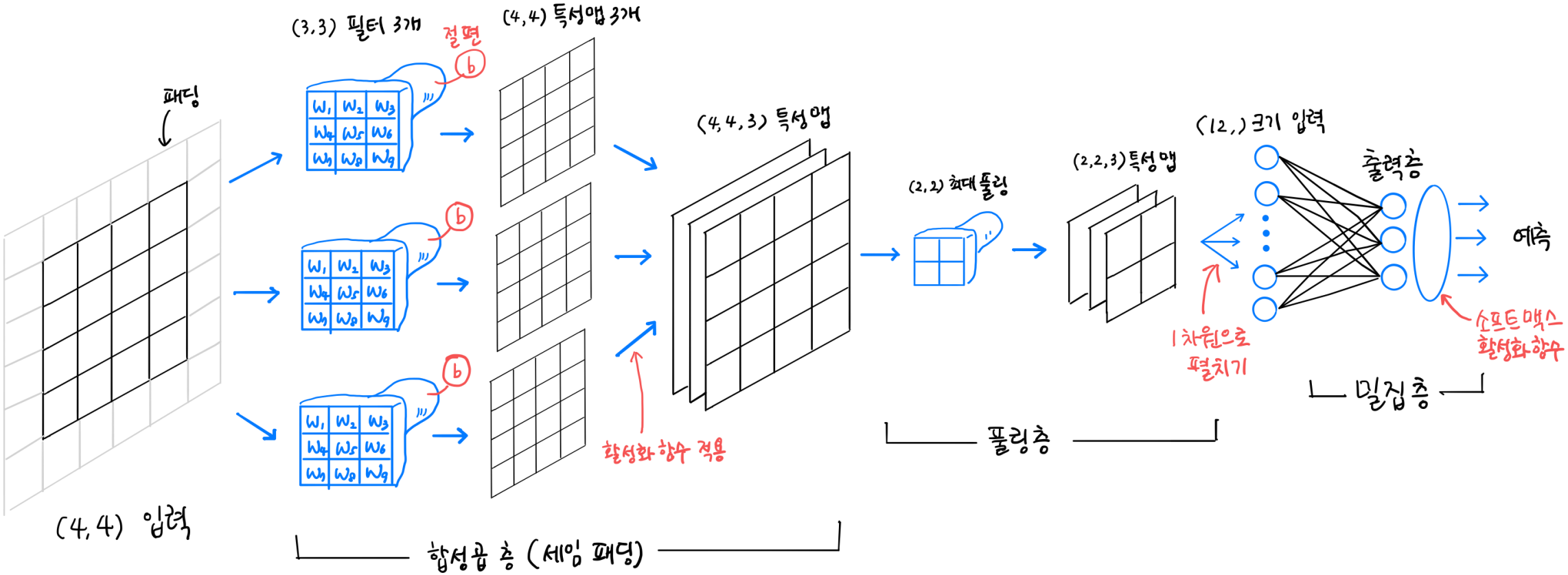
최대 풀링



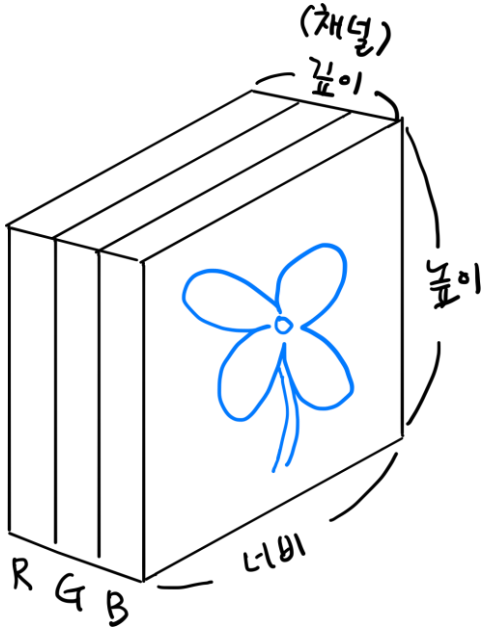
케라스의 풀링 층

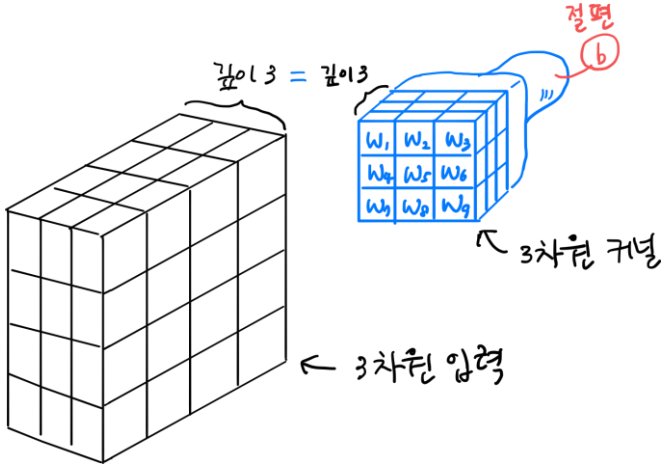
```
keras.layers.MaxPooling2D(2)

keras.layers.MaxPooling2D(2, strides=2, padding='valid')
```

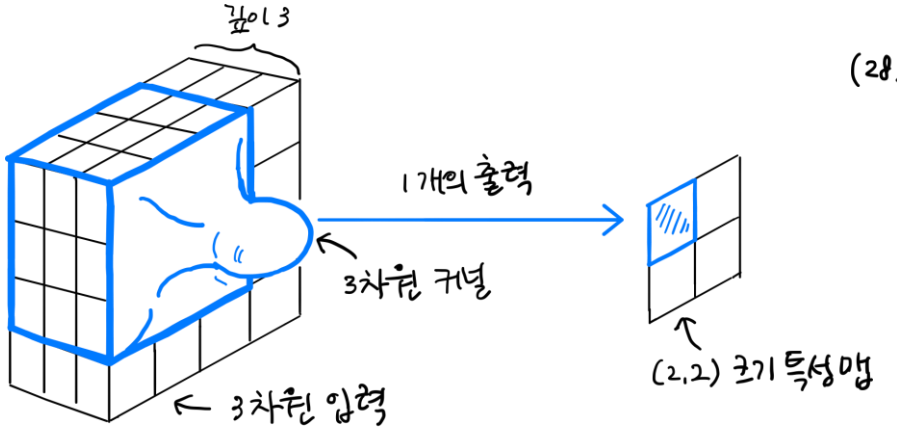
합성곱 신경망


패딩
(3,3) 필터 3개
절편
b
(4,4) 특성맵 3개
(4,4,3) 특성맵
(2,2) 최대 풀링
(2,2,3) 특성맵
(12,) 크기 입력
출력층
예측
1차원으로 펼치기
소프트맥스 활성화 함수
(4,4) 입력
활성화 함수 적용
합성곱 층 (세임 패딩)
풀링층
밀집층

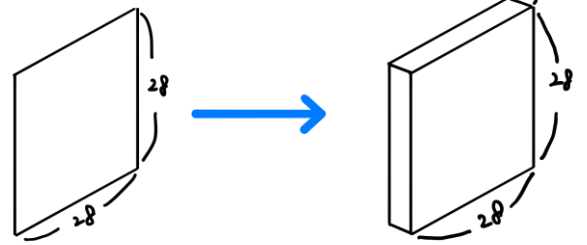
3차원 합성곱





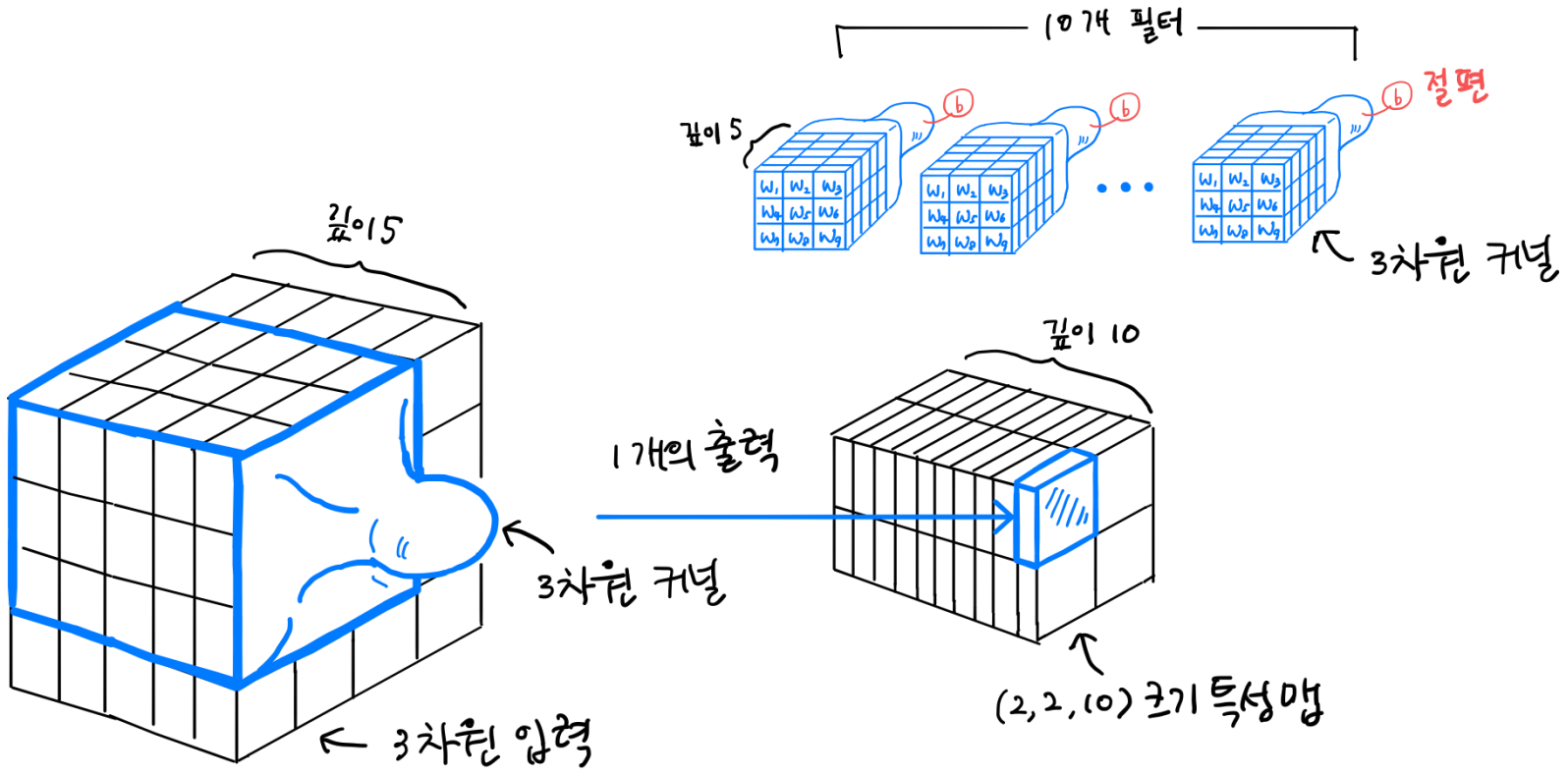


(4,4,3) 크기 입력과 (3,3,3) 크기 커널의 합성곱



(28,28) 크기의 2차원 배열 (28,28,1) 크기의 3차원 배열

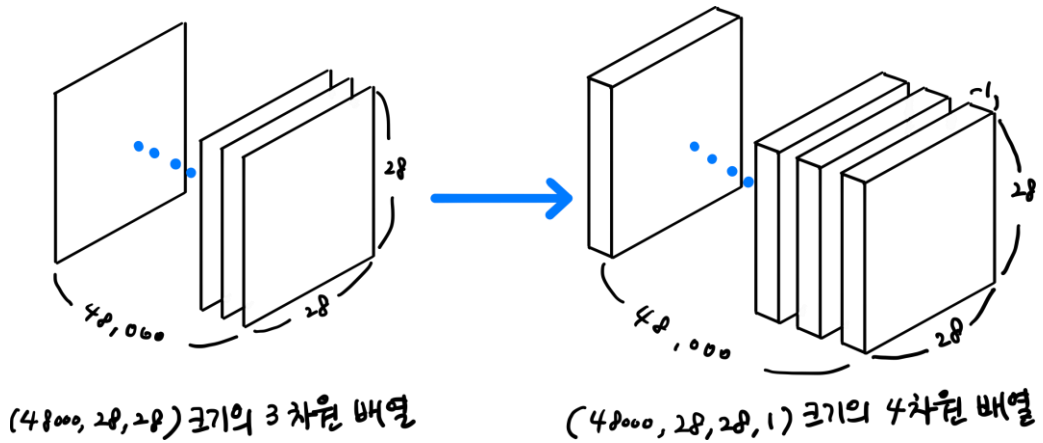
여러 개의 필터가 있는 3차원 합성곱



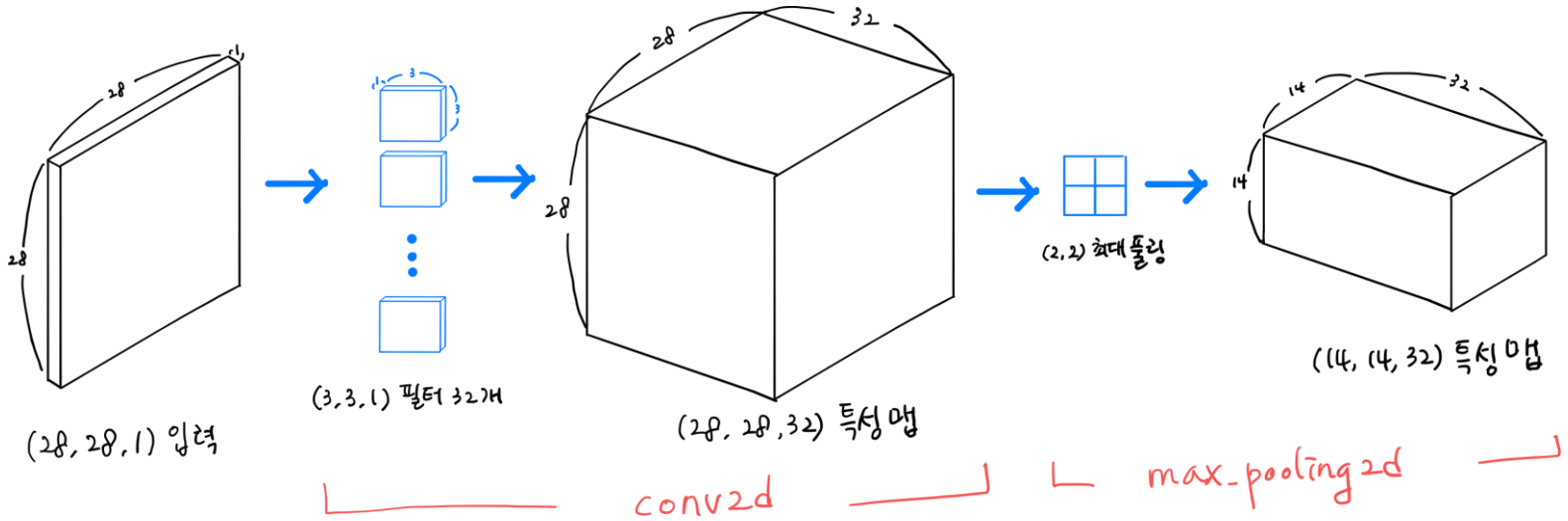
(4,4,5) 크기 입력과 (3,3,5) 크기 커널의 합성곱

패션 MNIST 데이터

```
(train_input, train_target), (test_input, test_target) =
keras.datasets.fashion_mnist.load_data()

train_scaled = train_input.reshape(-1, 28, 28, 1) / 255.0

train_scaled, val_scaled, train_target, val_target = train_test_split(
    train_scaled, train_target, test_size=0.2, random_state=42)
```

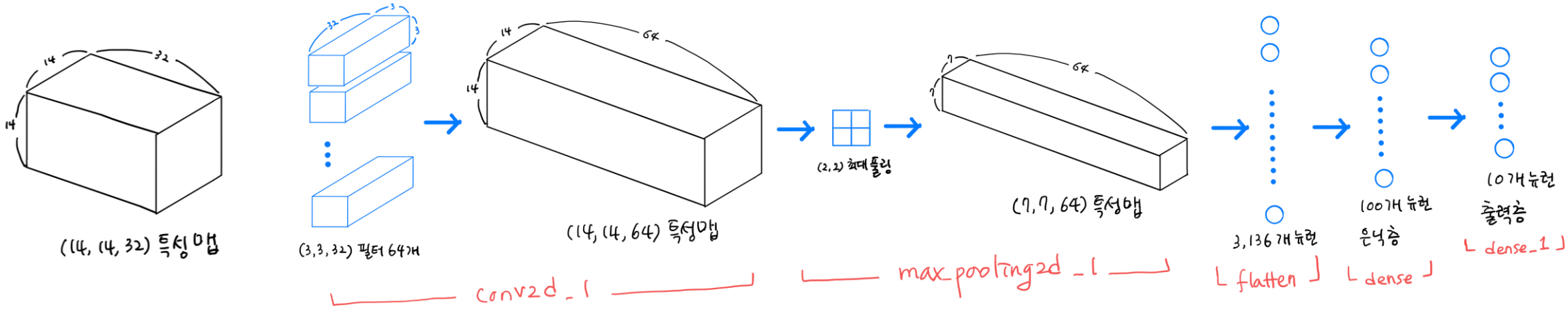


첫 번째 합성곱 층




```
model = keras.Sequential()

model.add(keras.layers.Conv2D(32, kernel_size=3, activation='relu',
          padding='same', input_shape=(28,28,1)))

model.add(keras.layers.MaxPooling2D(2))
```

두 번째 합성곱 층+완전 연결 층

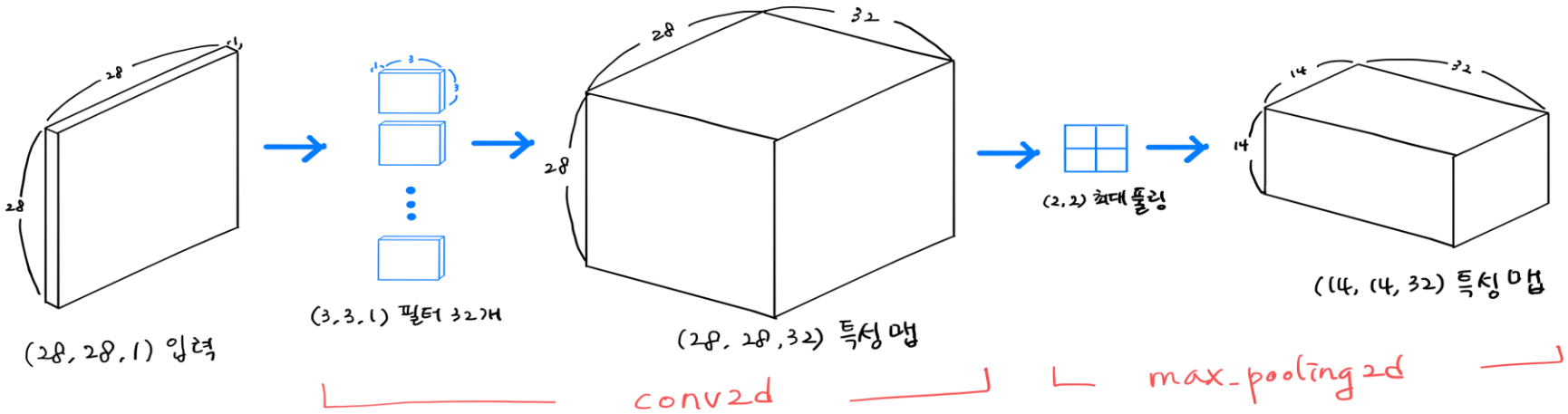


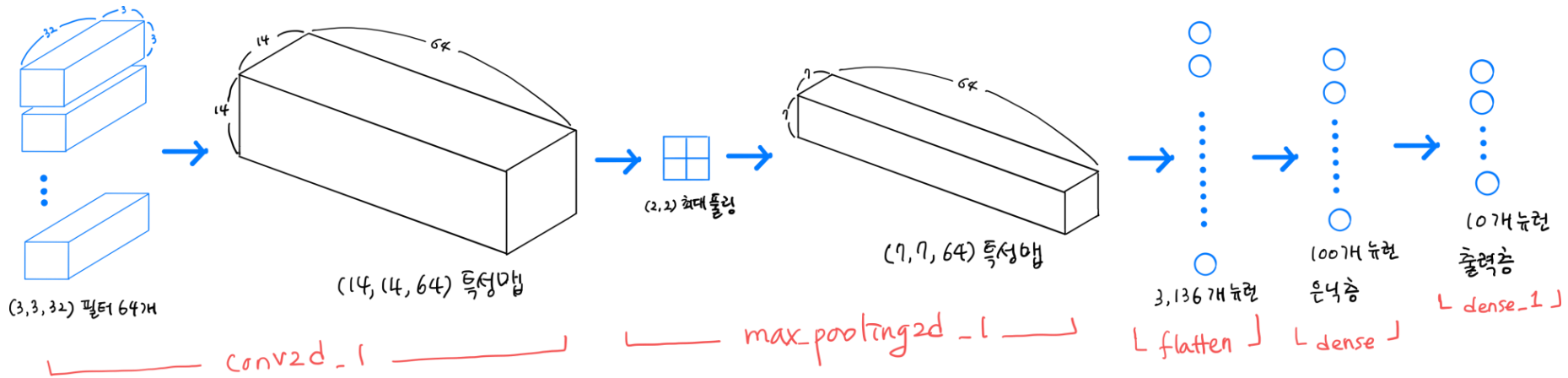

```
model.add(keras.layers.Conv2D(64, kernel_size=(3,3), activation='relu ' , padding='same’))
model.add(keras.layers.MaxPooling2D(2))

model.add(keras.layers.Flatten())
model.add(keras.layers.Dense(100, activation='relu’))
model.add(keras.layers.Dropout(0.4))
model.add(keras.layers.Dense(10, activation='softmax'))
```

모델 요약

```
Model: "sequential"
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
conv2d (Conv2D)              (None, 28, 28, 32)        320
_________________________________________________________________
max_pooling2d (MaxPooling2D) (None, 14, 14, 32)        0
_________________________________________________________________
conv2d_1 (Conv2D)            (None, 14, 14, 64)        18496
_________________________________________________________________
max_pooling2d_1 (MaxPooling2 (None, 7, 7, 64)          0
_________________________________________________________________
flatten (Flatten)            (None, 3136)              0
_________________________________________________________________
dense (Dense)                (None, 100)               313700
_________________________________________________________________
dropout (Dropout)            (None, 100)               0
_________________________________________________________________
dense_1 (Dense)              (None, 10)                1010
=================================================================
Total params: 333,526
Trainable params: 333,526
Non-trainable params: 0
_________________________________________________________________
```

모델 요약


28
28
1
(28, 28, 1) 입력
3
3
1
32
(3, 3, 1) 필터 32개
28
28
32
(28, 28, 32) 특성 맵
(2, 2) 최대풀링
14
14
32
(14, 14, 32) 특성 맵
conv2d
max_pooling2d


32
3
3
(3, 3, 32) 필터 64개
14
14
64
(14, 14, 64) 특성맵
(2, 2) 최대풀링
7
7
64
(7, 7, 64) 특성맵
3,136개 뉴런
flatten
100개 뉴런
은닉층
dense
10개 뉴런
출력층
dense_1
conv2d_1
max_pooling2d_1