



Deep structured learning in the cybersecurity domain [AICS323]

CNN 핵심요소 (W03)

Prof. Mee Lan Han (aeternus1203@gmail.com)

고려대학교

인공지능사이버보안학과

CONTENTS

- CNN 핵심요소
- Time series CNN
- 1D-CNN & RNN & BiLSTM

Convolutional Neural Network (CNN)

■ CNN 핵심요소

- Convolution Layer (합성곱 계층)
- Pooling Layer (풀링 계층)

- Filter
- Channel
- Kernel
- Stride
- Padding
- Feature map

- Fully Connected Layer (평탄화)

Convolutional Neural Network (CNN)

■ CNN 핵심요소

(1) Convolution Layer (합성곱 계층)

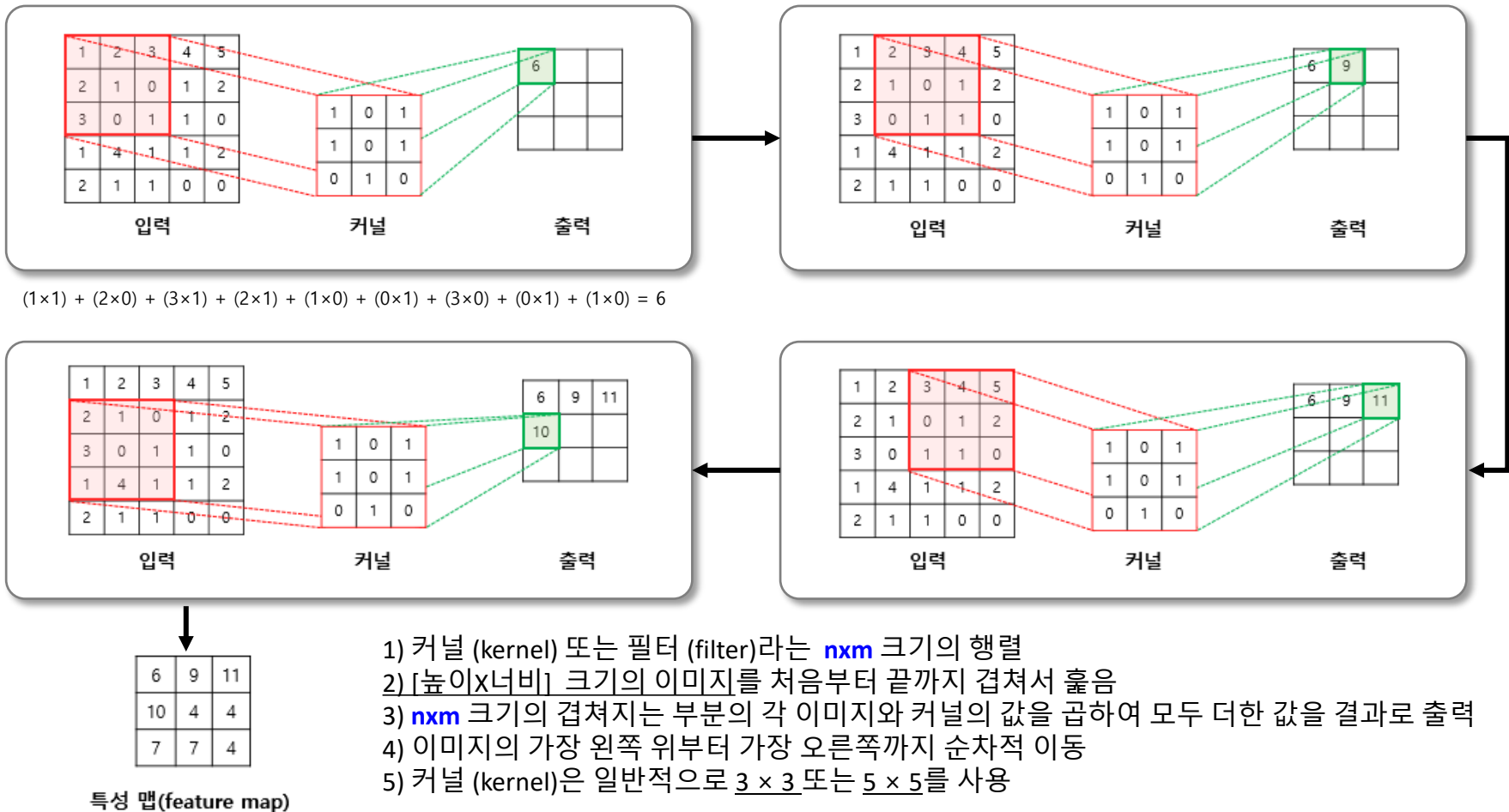
- 입력 이미지 (데이터)로부터 합성곱 연산을 통해 중요한 정보를 추출해 내는 필터 역할을 수행
- 새로운 이미지, 특징 맵 (**feature map**) 를 만들어내는 역할
- 어떤 합성곱 필터를 사용하느냐에 따라 합성곱 계층에서 추출해 내는 특징이 결정됨
- 합성곱 계층은 일반적인 신경망의 계층과는 다른 구조와 방식으로 작동함
 - 합성곱 계층의 노드
 - : 연결 가중치와 가중합의 개념이 아님
 - : 입력 이미지를 다른 이미지로 변환하는 필터 (커널, kernel)로 이미지 처리
- Convolution **Filter** or **Kernel** (합성곱 필터 or 커널)
 - 입력 이미지를 다른 이미지로 변환하는 필터
 - 합성곱 필터로 입력 이미지를 처리하면 특징 맵을 얻을 수 있음

Convolutional Neural Network (CNN)

■ CNN 핵심요소

(1) Convolution Layer (합성곱 계층)

- 5 × 5 이미지 행렬에 3 × 3 크기의 커널로 합성곱 연산을 수행하는 과정



Convolutional Neural Network (CNN)

■ CNN 핵심요소

(1) Convolution Layer (합성곱 계층)

- 5×5 이미지 행렬에 3×3 크기의 커널로 합성곱 연산을 수행하는 과정

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

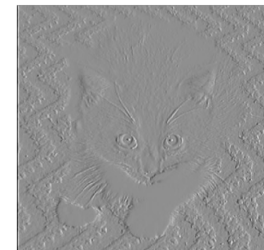
Image

4		

Convolved
Feature



Image



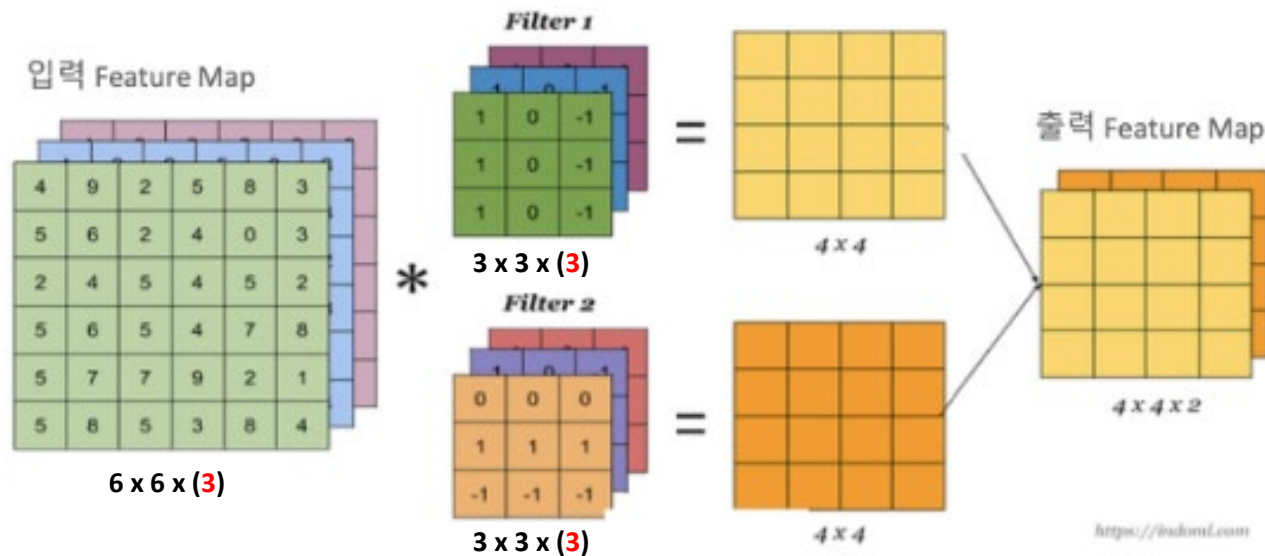
Convolved
Feature

Convolutional Neural Network (CNN)

■ CNN 핵심요소

(1) Convolution Layer (합성곱 계층)

- 6×6 이미지 행렬에 3×3 크기의 커널로 합성곱 연산을 수행하는 과정
- 하나의 단일 Filter는 여러 개의 커널을 가질 수 있고, 여러 개의 커널 수를 채널수라고도 함
 - Filter의 채널수가 Kernel 수 (Channel == Kernel)
- Convolution 연산 시 Filter의 채널수와 Feature map의 채널수는 같음



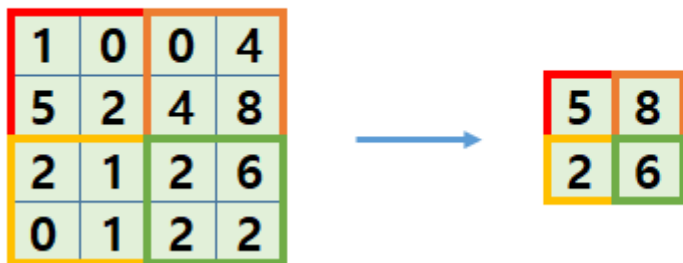
Convolutional Neural Network (CNN)

■ CNN 핵심요소

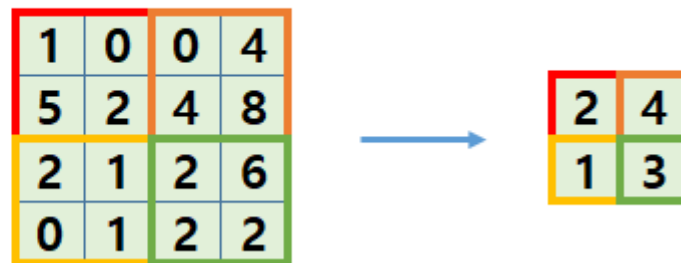
(2) Pooling Layer (풀링 계층)

- 입력 이미지의 특정 영역에 있는 픽셀을 묶어서 하나의 대표 픽셀로 축소
- 즉, 이미지의 차원을 축소하여 이미지의 크기를 줄이는 역할을 함
- CNN 내 앞 layer의 출력 feature map의 모든 data가 필요하지 않기 때문에 pooling layer만 사용 (추론을 위한 적당량의 데이터만 있어도 충분함!!) -> overfitting ↓ speedup ↑
- **Max pooling**: pooling window 내의 가장 큰 값을 선택하는 방법
- **Average pooling**: 평균 연산으로 인해 학습 결과가 좋지 않음

Max Pooling



Average Pooling

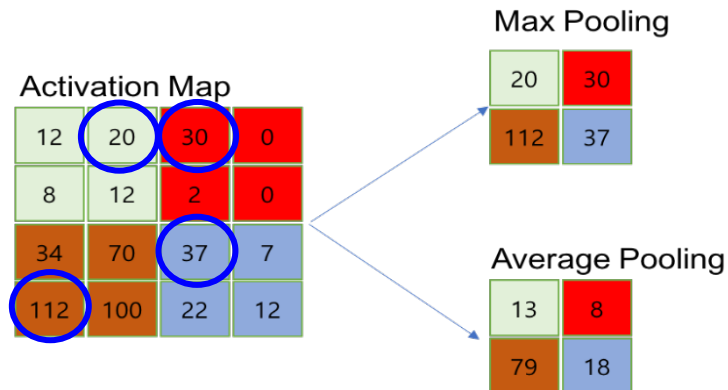


Convolutional Neural Network (CNN)

■ CNN 핵심요소

(2) Pooling Layer (풀링 계층)

- sub-sampling을 이용하여 feature map의 크기를 줄이고, 위치나 이동에 강한 특징을 추출하기 위한 방법
- **Max pooling**: pooling window 내의 가장 큰 값을 선택하는 방법, Overfitting 되는 단점
- **Average pooling**: 평균 연산으로 인해 학습 결과가 좋지 않음



- **Stochastic pooling**
 - 최대값 또는 평균값 대신 확률에 따라 적절한 activation을 선택함
 - 확률값은 특정 activation에 대해 전체의 activation의 합을 나누는 방식으로 계산됨

$$p_i = \frac{a_i}{\sum_{k \in R_j} a_k}$$

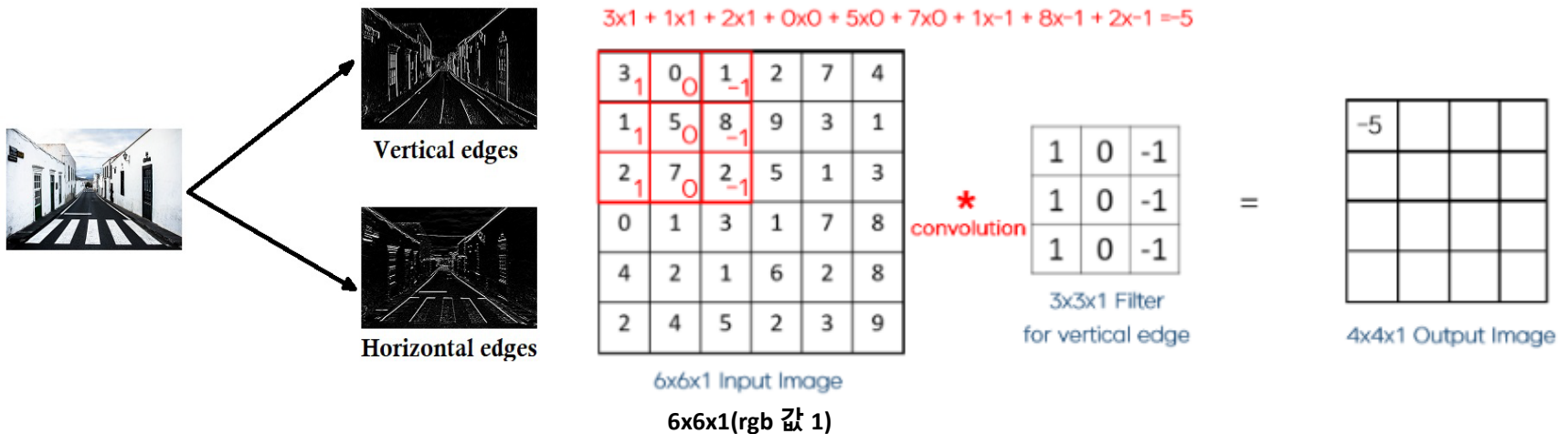
- Dropout과 같이 다양한 네트워크를 학습하는 듯한 model average 효과를 얻을 수 있음

Convolutional Neural Network (CNN)

■ CNN 핵심요소

(3-1) Filter

- 컴퓨터가 입력 이미지를 인식할 때, 가장 처음으로 하는 일이 vertical & Horizontal Edge를 인식하는 것
- CNN 알고리즘에서 Edge를 찾아내는 역할을 하는 것이 **Filter !!**
- Convolution operation** 즉, 합성곱 연산을 수행하여 엣지들을 찾아냄



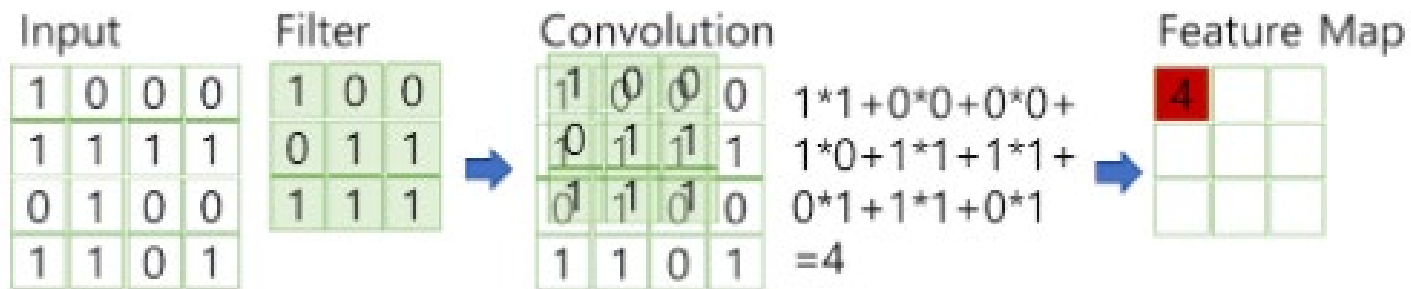
- 입력이미지가 주어졌을 때, Filter 를 직접 설정해줄 필요는 없음
- CNN에서 Filter 의 원소를 파라미터 설정한 뒤, **Backpropagation**을 이용해 스스로 학습하게 하여 최적의 filter을 찾아냄

Convolutional Neural Network (CNN)

■ CNN 핵심요소

(3-2) Filter

- 필터는 이미지의 특징을 찾아내기 위한 공유 파라미터
- 필터는 일반적으로 (3, 3) 이나 (5, 5) 과 같은 정사각 행렬로 정의
- CNN에서 학습의 대상은 필터 파라미터



- 입력 데이터를 지정된 간격으로 순회
- 채널별로 합성곱을 하고 모든 채널의 합성곱의 최종합을 Feature Map로 만들어 냄

Convolutional Neural Network (CNN)

■ CNN 핵심요소

(4-1) Channel (채널)

- 이미지는 높이 (height), 너비 (width), 채널 (color) 이라는 3Dimensional Tensor 구성
- 픽셀(Pixel): 작은 사각형으로 이루어진 흑백 이미지 (**gray_scale**)
- 높이와 너비 차원을 가짐 (높이 x 너비 = 이미지 크기)
- 각 픽셀은 숫자값으로 표시 (픽셀값)
- 픽셀값은 픽셀의 강도를 나타냄 (0~255)
 - 0에 가까울 수록 어두운 음영 / 255에 가까울 수록 밝거나 흰색의 음영



0	2	15	0	0	11	10	0	0	0	0	9	9	0	0	0
0	0	0	4	60	157	236	255	255	177	95	61	32	0	0	29
0	10	16	115	238	255	244	245	243	250	249	255	222	103	10	0
0	14	170	255	255	244	254	255	253	245	255	249	253	251	124	1
2	98	255	228	255	251	254	211	141	116	122	215	251	238	255	49
13	217	243	255	155	33	226	52	2	0	10	13	232	255	255	36
16	229	252	254	49	12	0	0	7	7	0	70	237	252	235	62
6	141	245	255	212	25	11	9	3	0	115	236	243	255	137	0
0	87	252	250	248	215	60	0	1	121	252	255	248	144	6	0
0	13	113	255	255	245	255	182	181	248	252	242	208	36	0	19
1	0	5	117	251	255	241	255	247	255	241	162	17	0	7	0
0	0	0	4	58	251	255	246	254	253	285	120	11	0	1	0
0	0	4	97	255	255	255	248	252	255	244	255	182	10	0	4
0	22	206	252	246	251	241	100	24	113	255	245	255	194	9	0
0	111	255	242	255	155	24	0	0	6	39	255	232	230	56	0
0	218	251	250	137	7	11	0	0	0	2	62	255	250	125	3
0	173	255	255	101	9	20	0	13	3	13	182	251	245	61	0
0	107	251	241	255	230	98	55	19	118	217	248	253	255	52	4
0	18	146	250	255	247	255	255	255	249	255	240	255	129	0	5
0	0	23	113	215	255	250	248	255	255	248	248	118	14	12	0
0	0	6	1	0	52	153	233	255	252	147	37	0	0	4	1
0	0	5	5	0	0	0	0	14	1	0	6	6	0	0	0

0	2	15	0	0	11	10	0	0	0	0	9	9	0	0	0
0	0	0	4	60	157	236	255	255	177	95	61	32	0	0	29
0	10	16	119	238	255	244	245	243	250	249	255	222	103	10	0
0	14	170	255	255	244	254	255	253	245	255	249	253	251	124	1
2	98	255	228	255	251	254	211	141	116	122	215	251	238	255	49
13	217	243	255	155	33	226	52	2	0	10	13	232	255	255	36
16	229	252	254	49	12	0	0	7	7	0	70	237	252	235	62
6	141	245	255	212	25	11	9	3	0	115	236	243	255	137	0
0	87	252	250	248	215	60	0	1	121	252	255	248	144	6	0
0	13	113	255	255	245	255	182	181	248	252	242	208	36	0	19
1	0	5	117	251	255	241	255	247	255	241	162	17	0	7	0
0	0	0	4	58	251	255	246	254	253	285	120	11	0	1	0
0	0	4	97	255	255	255	248	252	255	244	255	182	10	0	4
0	22	206	252	246	251	241	100	24	113	255	245	255	194	9	0
0	111	255	242	255	155	24	0	0	6	39	255	232	230	56	0
0	218	251	250	137	7	11	0	0	0	2	62	255	250	125	3
0	173	255	255	101	9	20	0	13	3	13	182	251	245	61	0
0	107	251	241	255	230	98	55	19	118	217	248	253	255	52	4
0	18	146	250	255	247	255	255	255	249	255	240	255	129	0	5
0	0	23	113	215	255	250	248	255	255	248	248	118	14	12	0
0	0	6	1	0	52	153	233	255	252	147	37	0	0	4	1
0	0	5	5	0	0	0	0	14	1	0	6	6	0	0	0

* 1채널 이미지

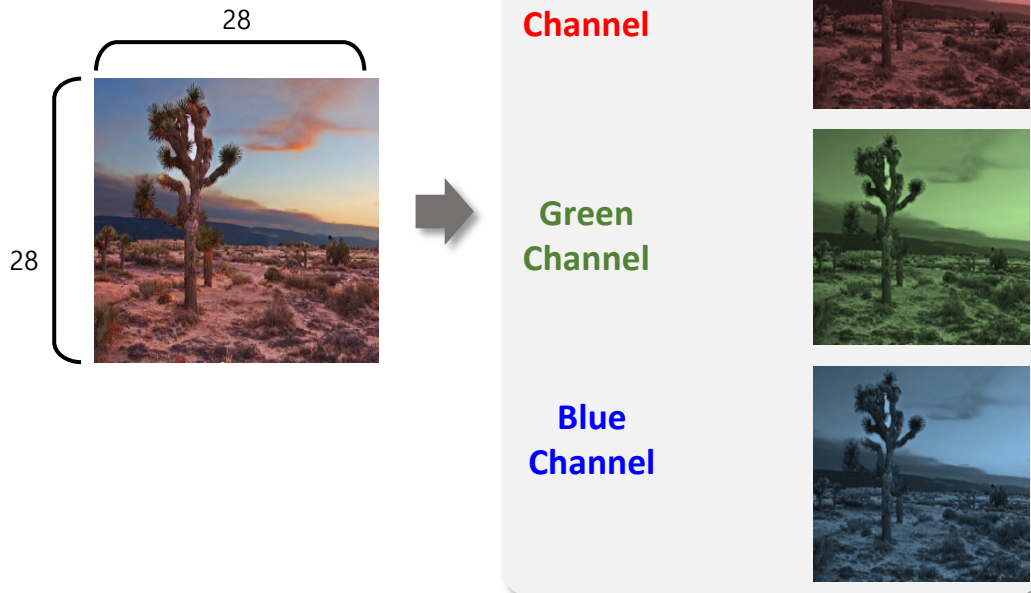
Convolutional Neural Network (CNN)

■ CNN 핵심요소

(4-2) Channel (채널)

- 컬러 사진 (다채널 이미지): 각 픽셀을 RGB 3개의 실수로 표현한 3차원 데이터
- 다채널 이미지는 높이 (height), 너비 (width), 채널 (color=depth) 이라는 3 차원 텐서로 구성됨
 - 높이: 이미지의 세로 방향 픽셀 수
 - 너비: 이미지의 가로 방향 픽셀 수
 - 채널: 최초의 이미지를 구성하는 색상 성분
 - 배치: 이미지 개수 (4차원 표현 시 포함되는 구성요소)

□ Ex) 다채널 이미지



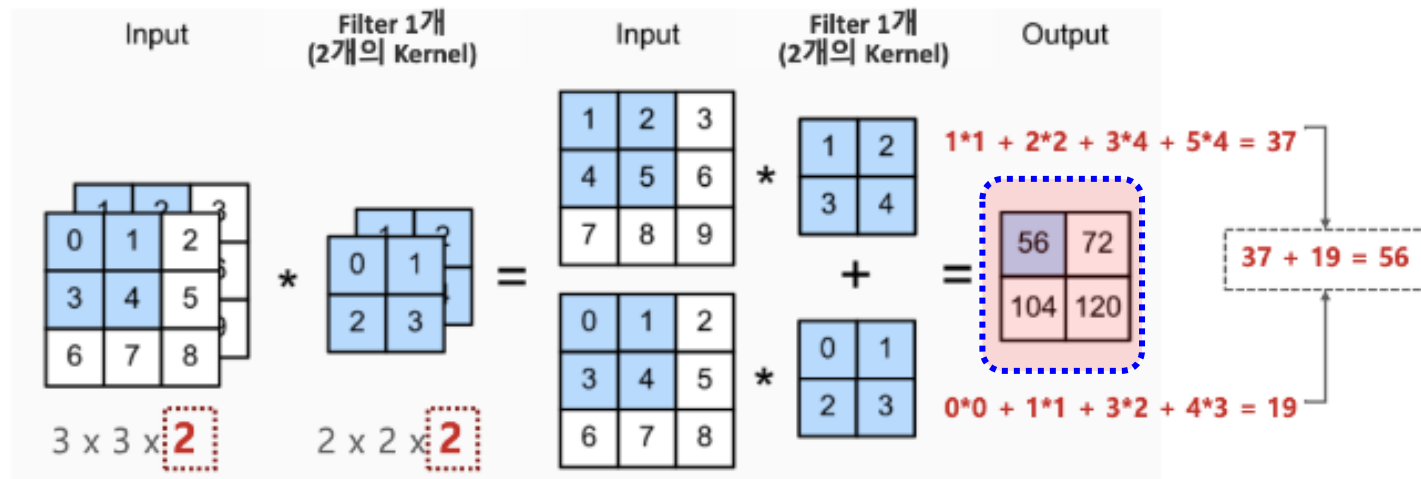
- 입력 이미지 크기 28 × 28
- 최초 입력 이미지 채널수 3개 (Red, Green, Blue)
- 생성된 feature map 3개
- Input shape = (28x28x3)

Convolutional Neural Network (CNN)

■ CNN 핵심요소

(4-3) Channel (채널)

- 다채널 Input의 **단일 Convolution Filter 메커니즘**
 - Input의 채널수와 Filter의 채널수는 무조건 같아야 계산이 수행됨
 - (하단 그림) **Filter가 1개**이기 때문에, **Feature map의 채널수는 1개** 생성
 - 출력 Feature map은 (2 x 2 x 1)의 크기를 가짐**

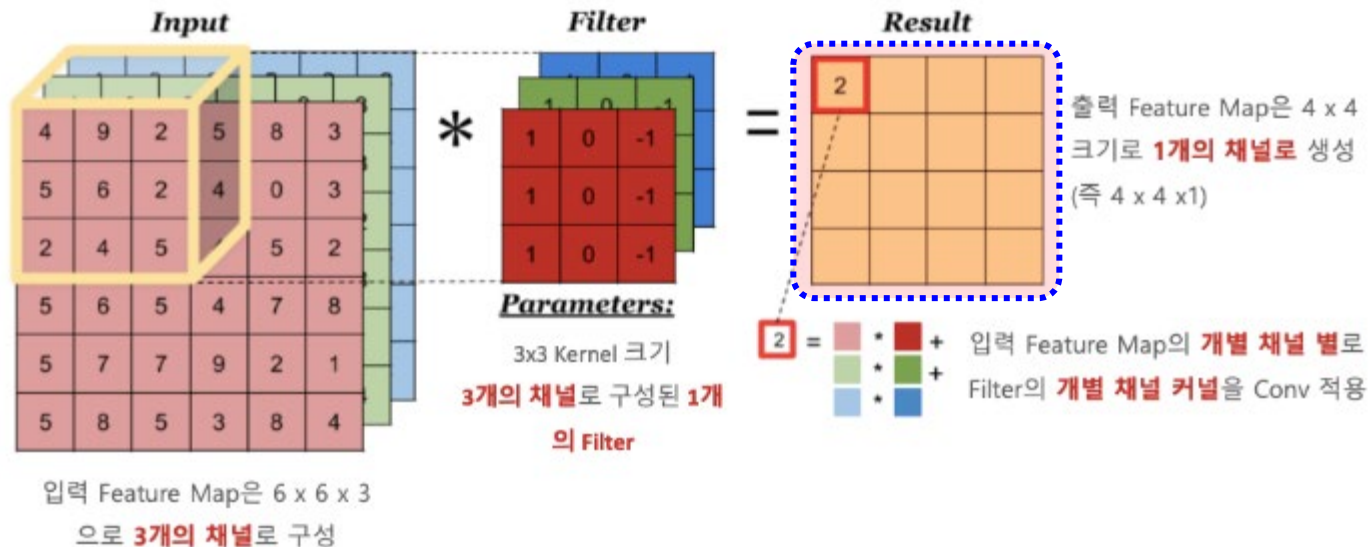


Convolutional Neural Network (CNN)

■ CNN 핵심요소

(4-4) Channel (채널)

- 다채널 Input의 **단일 Convolution Filter 메커니즘**
 - Input의 채널수와 Filter의 채널수는 무조건 같아야 계산이 수행됨
 - (하단 그림) **Filter가 1개**이기 때문에, **Feature map의 채널수는 1개** 생성
 - 출력 Feature map은 (4 x 4 x 1)의 크기를 가짐**

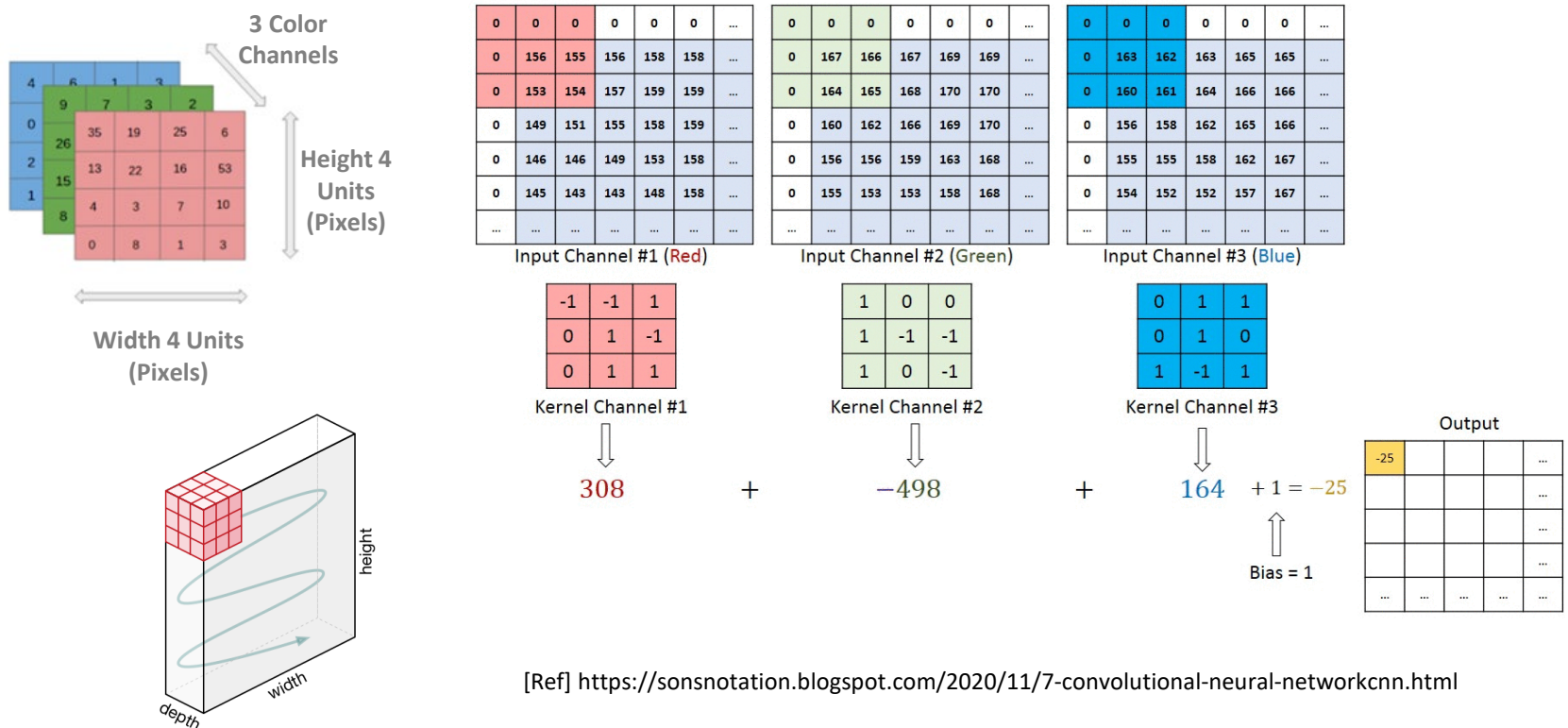


Convolutional Neural Network (CNN)

■ CNN 핵심요소

(4-5) Channel (채널)

- 다채널 Input의 **단일 Convolution Filter 메커니즘**
 - Input 의 채널수와 Filter의 채널수는 무조건 같아야 계산이 수행됨
 - (하단 그림) **Filter가 1개**이기 때문에, **Feature map의 채널수는 1개** 생성
 - 출력 Feature map은 (4 x 4 x 1)의 크기를 가짐**

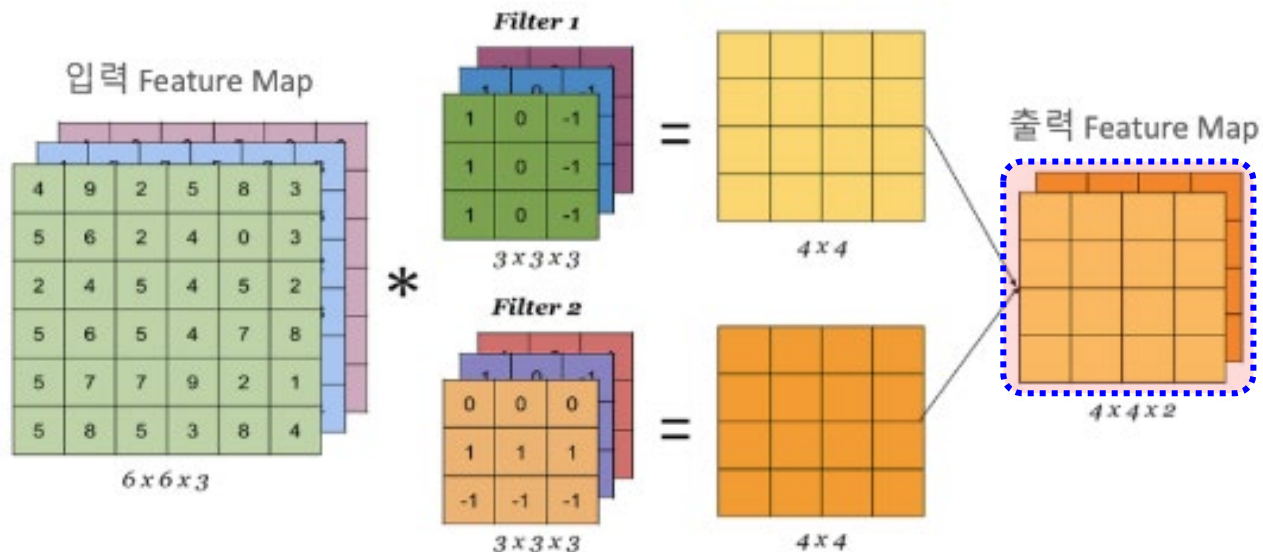


Convolutional Neural Network (CNN)

■ CNN 핵심요소

(4-6) Channel (채널)

- 다채널 Input의 여러 개 Convolution Filter 메커니즘
 - 출력 Feature Map의 채널수는 **Convolution** 을 적용한 Filter의 개수로 결정됨
 - (하단 그림) Filter가 2개이기 때문에, Feature map의 채널수는 2개 생성
 - 출력 Feature map은 (4 x 4 x 2)의 크기를 가짐**



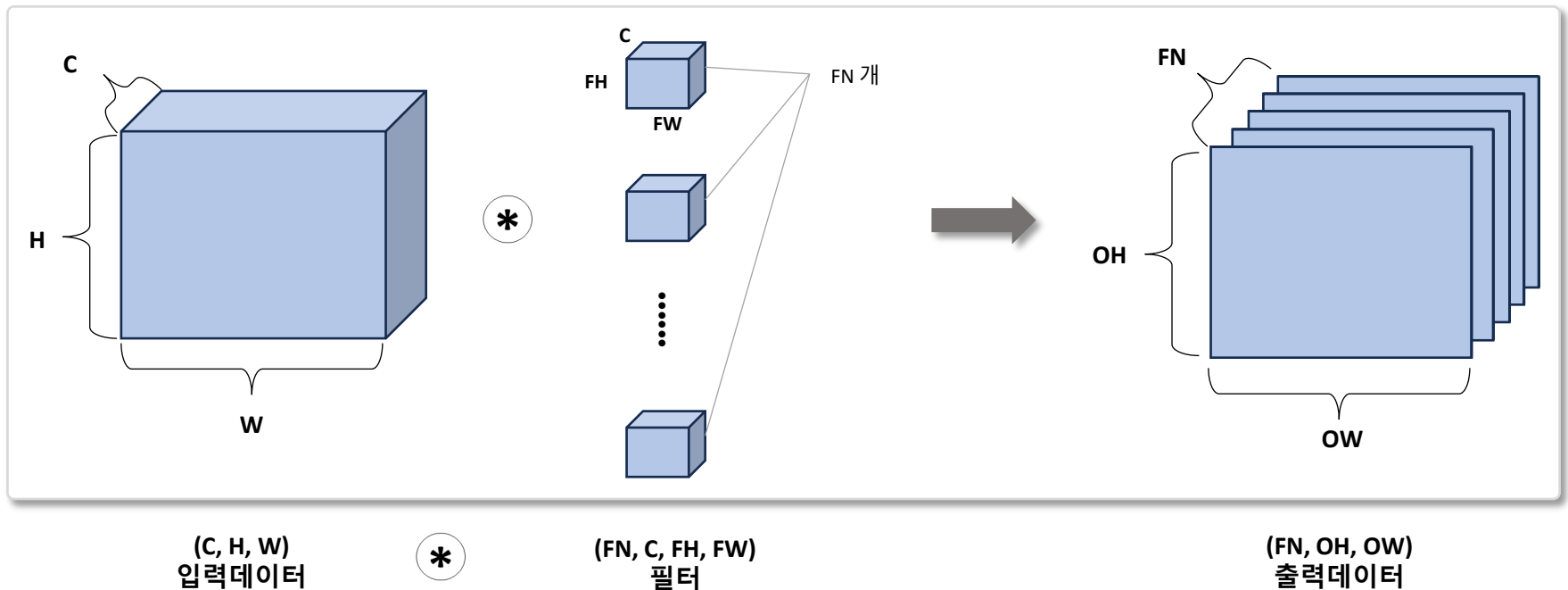
✓ Output, 즉, feature map의 채널을 늘리려면 어떻게 해야 할까???

Convolutional Neural Network (CNN)

■ CNN 핵심요소

(5-1) Kernel

- Convolution Filter가 input에 적용될 범위
- 개별 Kernel은 Filter 내 다른 값을 가질 수 있음
- **Kernel의 개수가 곧 channel의 개수**
 - 실생활에서 보는 이미지는 보통 다수의 채널을 갖고있음
 - 각 채널별로 필터를 훑은 후 나온 **n개의 Feature map**을 마지막으로 합쳐 최종 Feature map 을 생성



Convolutional Neural Network (CNN)

■ CNN 핵심요소

(5-2) Kernel

- Kernel Size는 면적 (가로x세로)를 의미, 일반적으로 동일한 크기로 일치시킴 (3x3, 5x5)
- Kernel의 크기가 클수록 입력값(원본이미지 또는 Feature map)으로부터 더 많은 Feature를 뽑을 수 있음
 - 다만, 너무 큰 사이즈일 경우, Convolution 연산 시 너무 많은 연산량과 파라미터 증가

```
from tensorflow.keras.layers import Input, Conv2D, MaxPooling2D
```

```
input_tensor = Input(shape=(28, 28, 1))
```

```
conv_out = Conv2D(filters=32, kernel_size=3, activation='relu', stride=1, padding='same')(input_tensor)
```

```
pool_out = MaxPooling2D(pool_size=2)(conv_out)
```

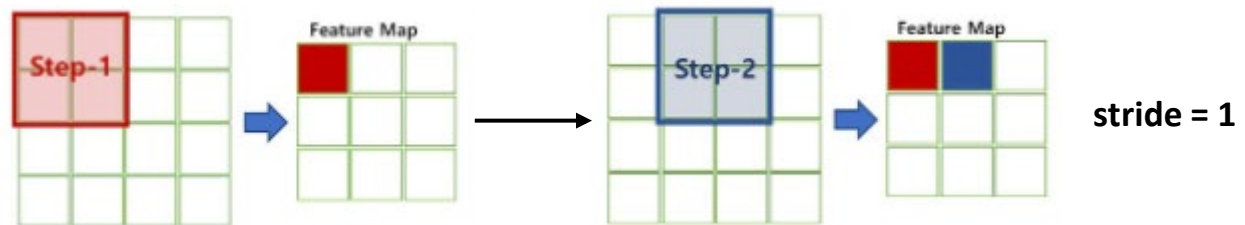
- 3x3인 정방행렬을 가지는 필터 32개를 input_tensor에 적용하겠다는 의미

Convolutional Neural Network (CNN)

■ CNN 핵심요소

(6) Stride

- 지정된 간격으로 필터를 순회하는 간격 (필터의 이동량)
- 필터는 입력 데이터를 지정한 간격으로 순회하면서 합성곱을 계산함

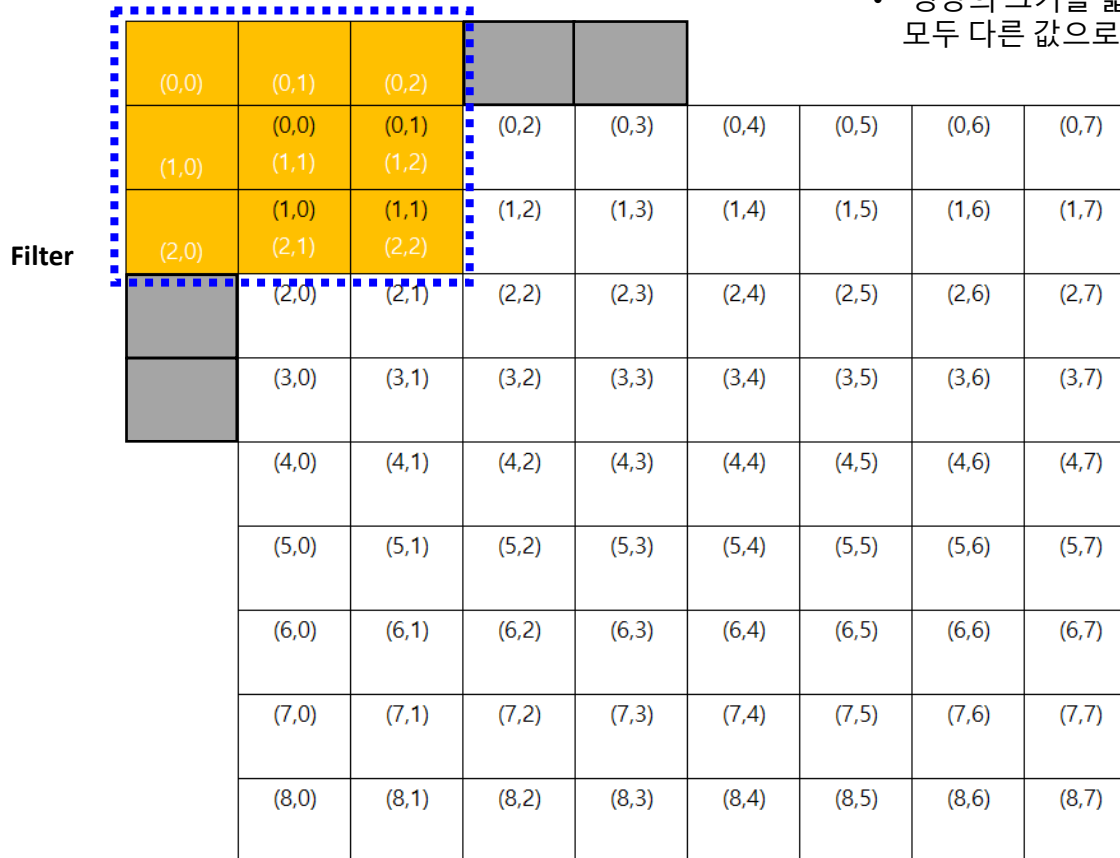


Convolutional Neural Network (CNN)

■ CNN 핵심요소

(7-1) Padding

- 필터를 영상에 적용시킬 때, 영상의 테두리에는 필터의 일부가 원본과 대응되지 않는 문제점 발생



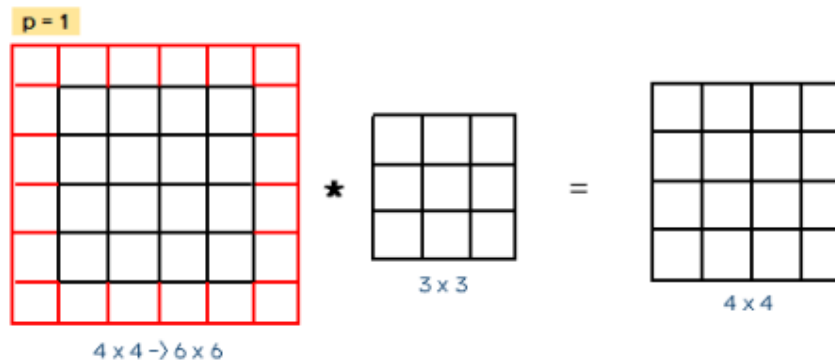
- 영상의 크기를 넓혀주기 위해 벗어난 범위를 모두 다른 값으로 채우는 padding 기법 사용함

Convolutional Neural Network (CNN)

■ CNN 핵심요소

(7-2) Padding

- 입력 데이터의 외각에 지정된 픽셀만큼 특정 값으로 채워 넣는 것을 의미
- Convolution 레이어에서 Filter와 Stride에 작용으로 Feature Map 크기는 입력데이터 보다 작음
- Convolution 레이어의 출력 데이터가 줄어드는 것을 방지
 - CNN과정에서 여러 번의 계산을 하게 될 때, 초반에 이미지가 너무 작아져 버려 더 깊게 학습할 데이터가 부족해지는 현상이 발생함
 - **Zero padding**: 보통 패딩 값으로 0으로 채움
 - **Same padding**: 이미지의 크기를 똑같이 유지해주는 padding
 - **Valid padding**: 실제 값만 사용하는 것, padding을 하지 않는 것을 의미함



p=1일때 (I=4, F=3, S=1), Output 이미지 크기 = ?

Convolutional Neural Network (CNN)

■ CNN 핵심요소

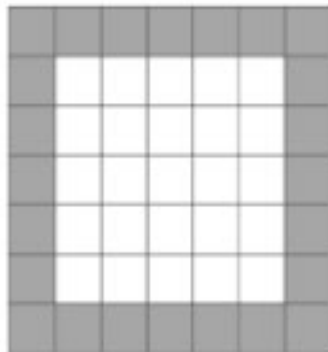
(8) Feature map

- 출력 Feature map 크기 계산
 - I: 입력 Feature map 의 크기
 - F: Filter의 크기 (Kernel size)
 - P: Padding
 - S: Strides

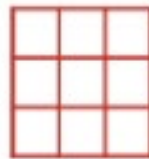
$$O = \frac{I - F + 2P}{S} + 1$$

$$O = \frac{I - F + 2P}{S} + 1 = \frac{5 - 3 + 2}{1} + 1 = 5$$

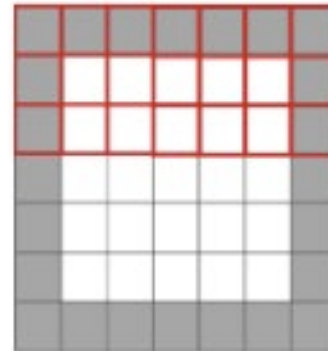
Stride가 1 일 때 Output이 input과 동일한 크기를 가지려면 P 는 (F-1)/2 로 설정



원본 5 x 5 Input
Padding 1 추가 후 7 x 7



3 x 3 Filter



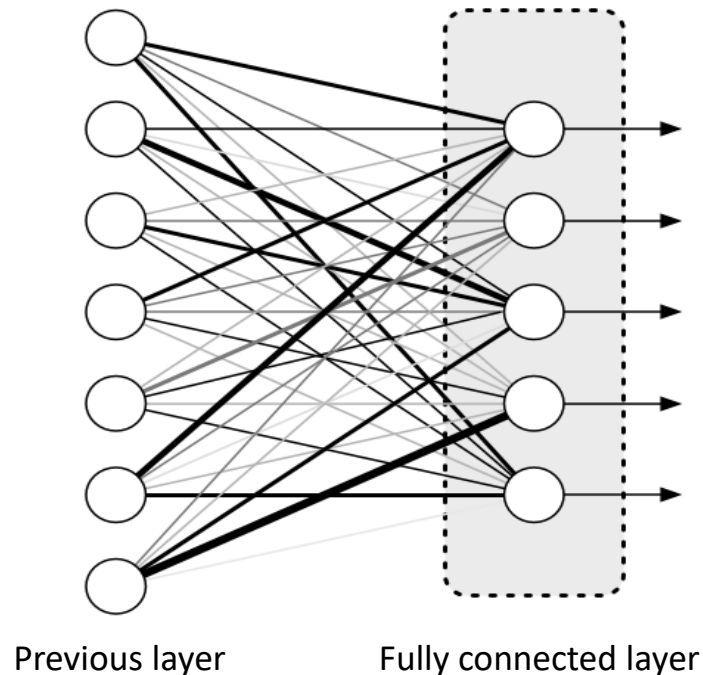
3 x 3 Filter가 5번 sliding 이
동하여 5 x 5 Output 생성

Convolutional Neural Network (CNN)

■ CNN 핵심요소

(9) Fully Connected Layer (FCN): 평탄화

- Convolution Layer과 Pooling Layer로부터 얻어진 특징 벡터들은 Fully Connected Layer의 입력으로 사용
 - **이전 레이어의 출력을 평탄화 (Flatten)하여 다음 스테이지의 입력이 될 수 있는 단일 벡터로 변환**
 - 비선형 공간에서의 분류를 수행하게 됨
- 모든 뉴런들이 전부 연결되는 형태를 갖고 있기 때문에 Fully Connected 라는 이름으로 불림



>> 1~3 과정을 Fully Connected Layers라고 정의함

- ① 2차원 배열 형태의 이미지를 1차원 배열로 평탄화
- ② 활성화 함수 (Relu, Tanh 등) 뉴런을 활성화
- ③ 분류기 (Softmax) 함수로 분류

Convolutional Neural Network (CNN)

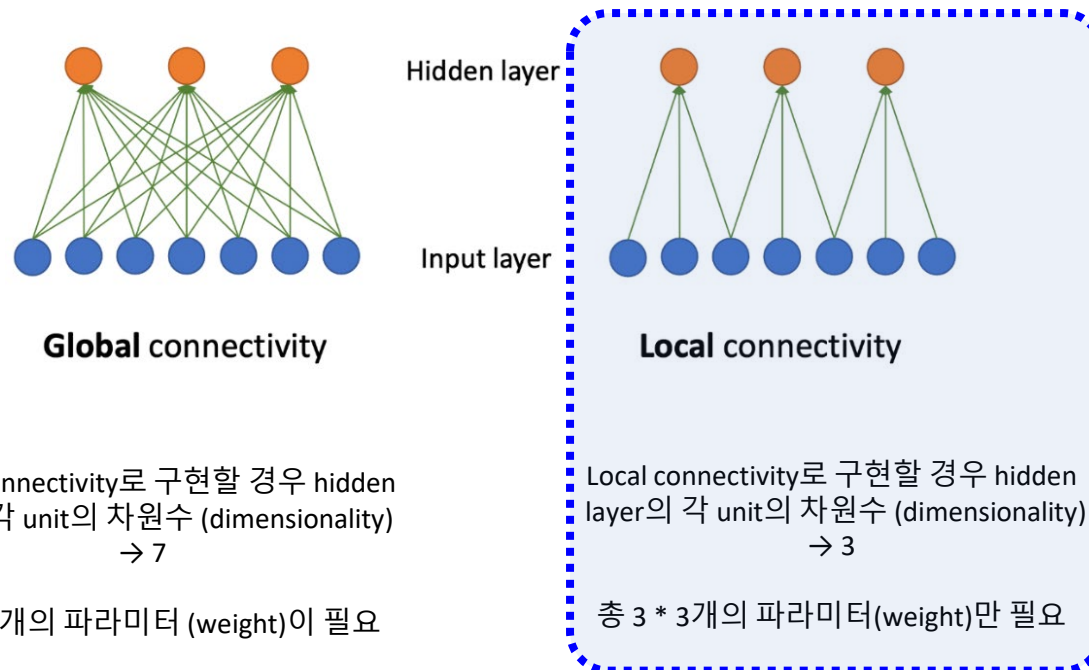
■ CNN 특징

□ CNN은 기존의 다층신경망에 비해 중요한 두 가지 특징 존재

* 비선형필터: 최소/최대값, 중간값필터

□ Locality (Local Connectivity)

- 공간적으로 인접한 신호들에 대한 correlation 관계를 비선형 필터를 적용하여 추출함 (local 정보활용)
- 모든 데이터에 대해 input 값으로 받아들이지 않고, **인접한 데이터에 대해서만 input 값**으로 받아들이م
- 각 output 값은 모든 input 값을 고려하지 않고 **적은 수의 input값에 의해서만 결정될 수 있음**



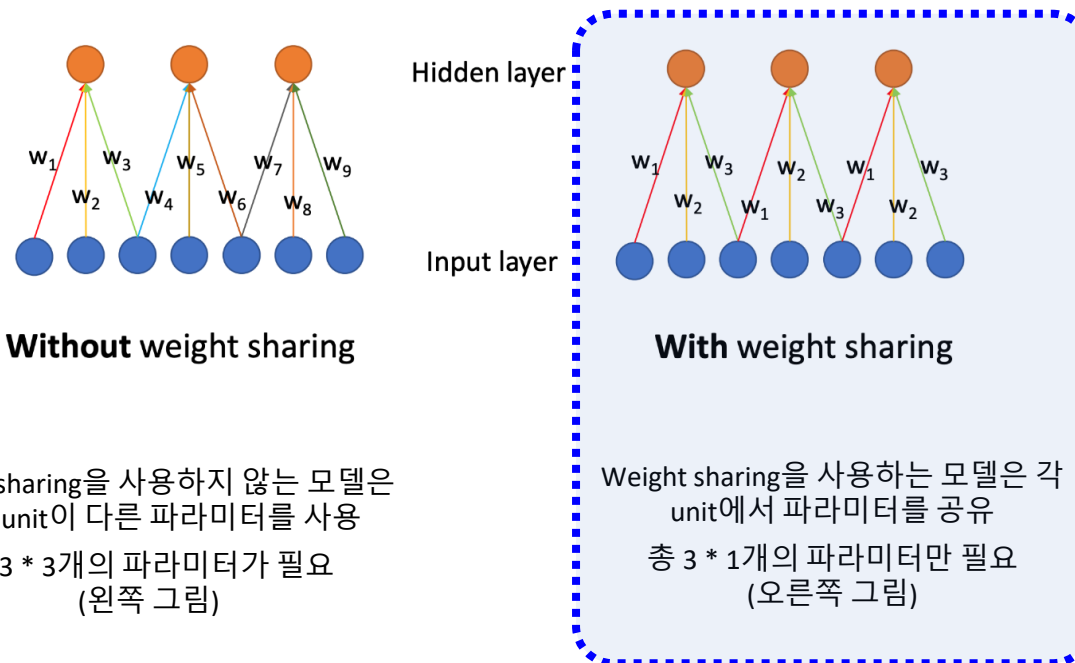
Convolutional Neural Network (CNN)

■ CNN 특징

□ CNN은 기존의 다층신경망에 비해 중요한 두 가지 특징 존재

□ Shared Weights

- 동일한 계수를 갖는 filter를 전체 영상에 반복적으로 적용함으로 변수의 수를 획기적으로 줄임
- Topology 변화에 무관하게 항상성 (Invariance) 을 얻을 수 있음



Thank you



KOREA
UNIVERSITY