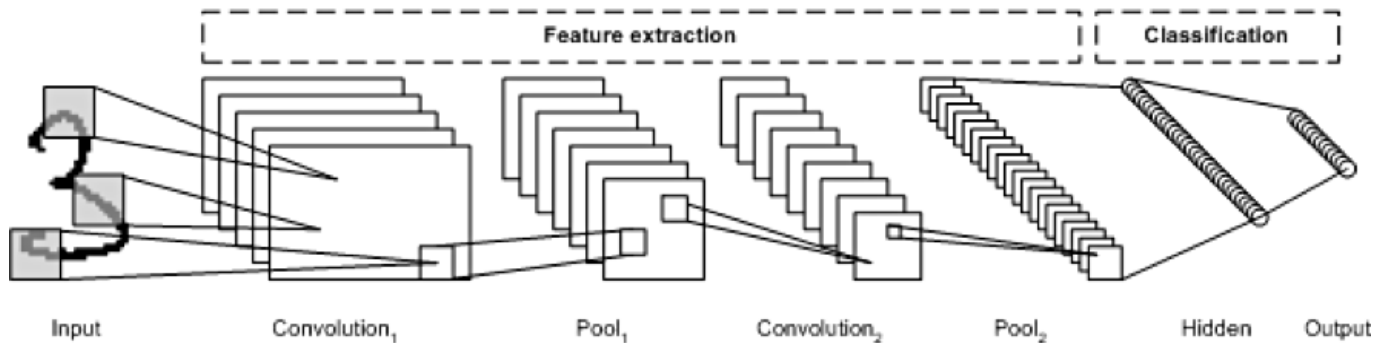


CNN 요약



fully connected Layer 만으로 구성된 인공 신경망의 입력 데이터는 1차원(배열) 형태로 한정됩니다. 한 장의 컬러 사진은 3차원 데이터입니다. 배치모드에 사용되는 여러장의 사진은 4차원 데이터입니다. 사진 데이터로 전 연결(FC, Fully connected)신경망을 학습시켜야 할 경우에, **3차원 사진 데이터를 1차원으로 평면화시켜야** 합니다. 사진 데이터를 평면화 시키는 과정에서 공간 정보가 손실될 수밖에 없습니다. 결과적으로 이미지 공간 정보 유실로 인한 정보 부족으로 인공 신경망이 특징을 추출 및 학습이 비효율적이고 정화도를 높이는데 한계가 있습니다. **이미지 공간 정보를 유지한 상태로 학습이 가능한 모델이 바로 CNN입니다.**

CNN은 기존 Fully Connected Neural Network 와 비교하여 다음과 같은 차별성을 가집니다.

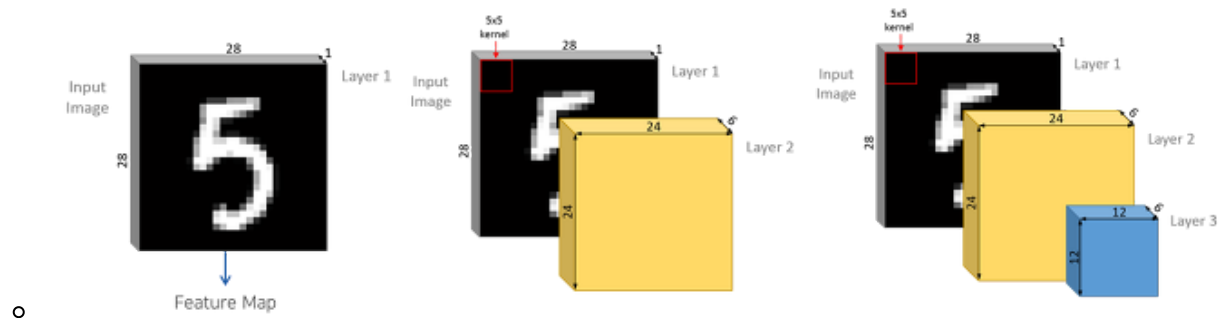
- 각 레이어의 입출력 데이터의 형상유지
- 이미지 공간 정보를 유지하면서 인접 이미지와의 특징을 효과적으로 인식
- 복수의 필터로 이미지의 특징 추출 및 학습
- 추출한 이미지의 특징을 모으고 가오하하는 Pooling레이어
- 필터를 공유 파라미터로 사용하기 때문에, 일반 인공 신경망과 비교하여 학습 파라미터가 매우 적음

CNN은 위 이미지와 같이 이미지의 특징을 추출하는 부분과 클래스를 분류하는 부분으로 나눌 수 있습니다. 특징 추출 영역은 Convolution Layer와 Pooling Layer를 여러 겹 쌓는 형태로 구성됩니다. Convolution Layer는 **입력 데이터에 필터를 적용 후 활성화 함수를 반영하는 필수 요소**입니다. Convolution Layer 다음에 위치하는 Pooling Layer는 선택적인 레이어입니다. CNN 마지막 부분에는 이미지 분류를 위한. Fully connected레이어가 추가됩니다. 이미지는 특징을 추출하는 부분과 이미지를 분류하는 부분 사이에 이미지 형태의 데이터를 배열 형태로 만드는 Flatten레이어가 위치합니다.

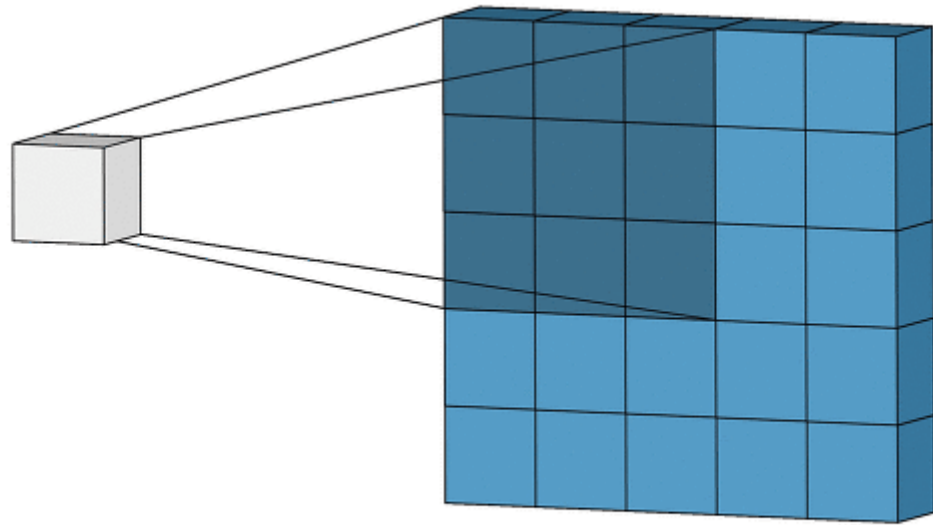
CNN은 이미지 특징 추출을 위하여 입력데이터를 필터가 순회하며 합성곱을 계산하고 그 계산 결과를 이용하여 Feature map을 만듭니다. Convolution Layer는 Filter 크기 Stride Padding 적용 여부, MAX Pooling 크기에 따라서 출력 데이터의 Shape이 변경됩니다.

CNN주요용어정리

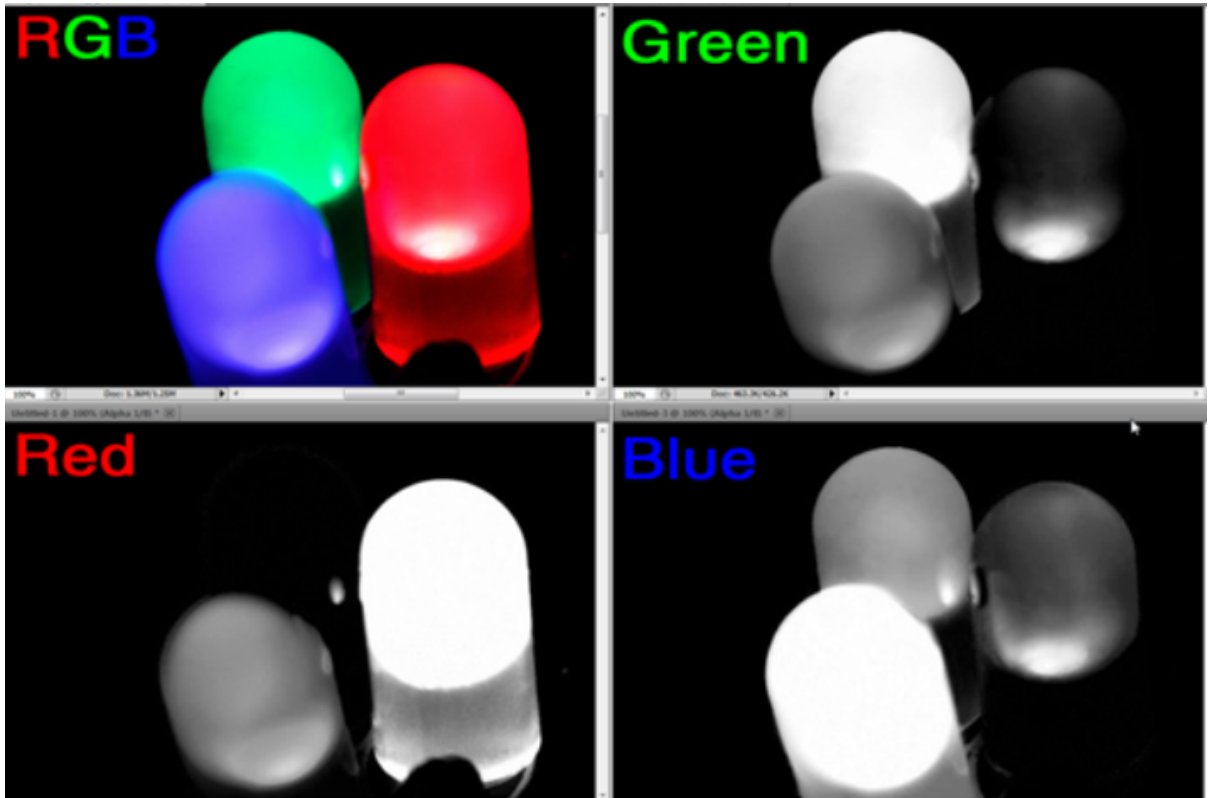
- **합성곱(Convolution)**
 - 합성곱층은 CNN에서 가장 중요한 구성요소이며 완전연결 계층과는 달리 합성곱층(convolutional layer)은 아래 그림과 같이 입력 데이터의 형상을 유지한다. 3차원 이미지 그대로 입력층에 입력받으며, 출력 또한 3차원 데이터로 출력하여 다음 계층으로 전달하기 때문에 CNN에서는 이미지 데이터처럼 형상을 가지는 데이터를 제대로 학습할 가능성이 높다고 할 수 있다.



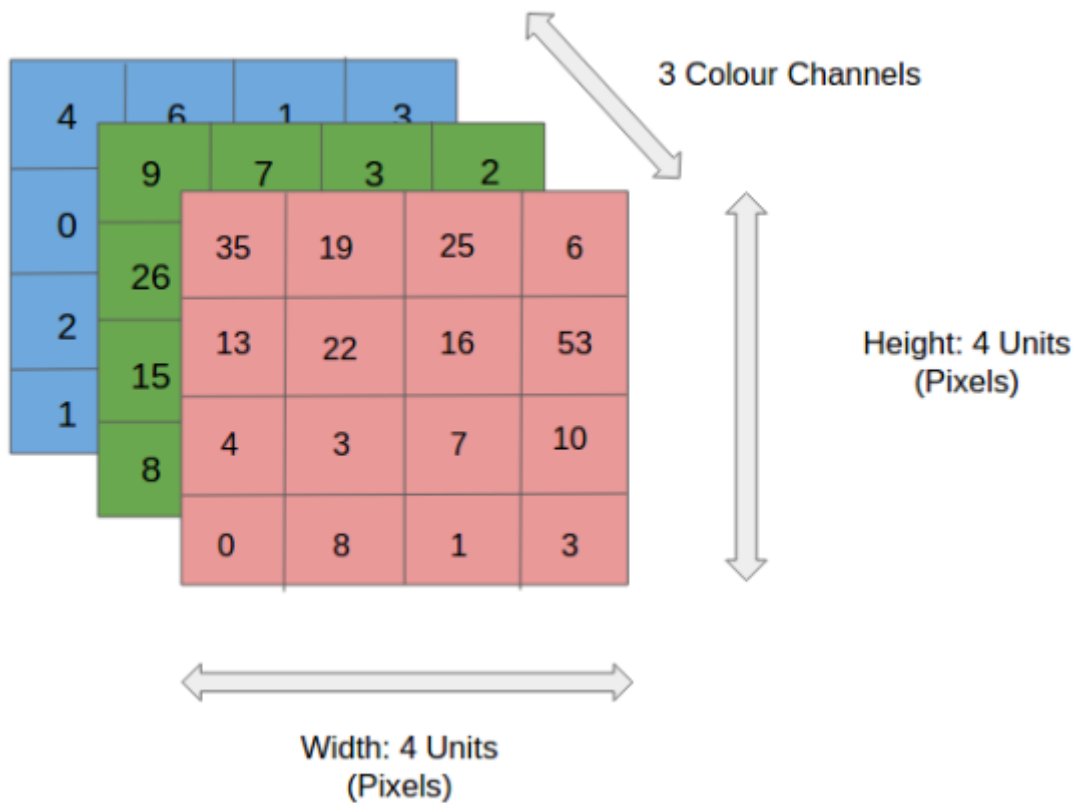
-
- 합성곱층의 뉴런은 아래의 그림처럼 입력 이미지의 모든 픽셀에 연결되는 것이 아니라 합성곱층 뉴런의 수용영역 안에 있는 픽셀에만 연결이 되기 때문에 앞의 합성곱층에서는 저수준 특성에 집중하고 그 다음 합성곱층에서는 고수준 특성으로 조합해 나가도록 해준다.



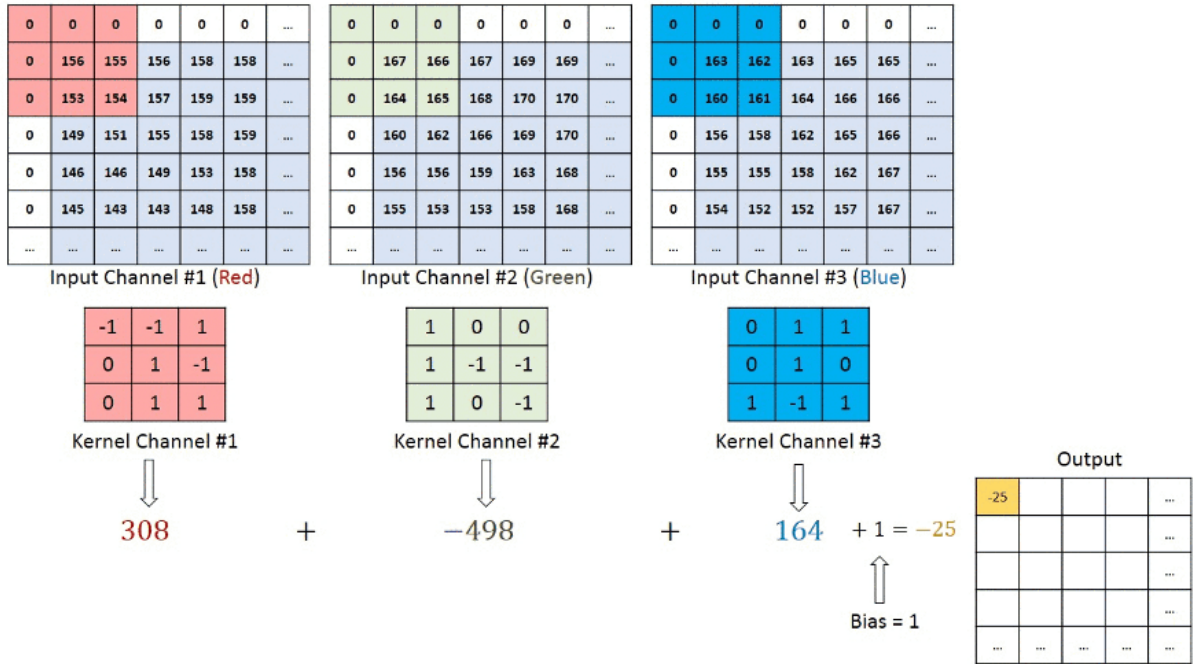
-
- 합성곱 처리결과로부터 Feature Map을 만든다.
- 채널



-
- 보통 우리가 알고있는 color 이미지는 RGB3개로 표현된 3차원 데이터이다.
- 즉 color 이미지는 red, blue, green channel로 이루어져있다.
- 추가로 흑백 이미지의 경우 하나의 channel로 이루어져있다.



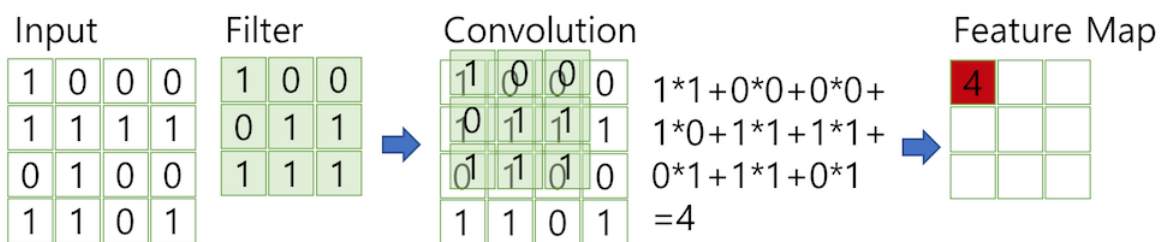
○



- 그래서 하나의 input 이미지를 3개의 channel로 분리시켜서 convolution을 진행한다.
- 이때 3개의 channel은 위에서 언급했던 red, blue, green channel이다.
- Convolution layer에 유입되는 입력 데이터에는 한개 이상의 필터가 적용된다.
- 1개 필터는 Feature map의 채널이 된다. Convolution layer에 n개의 필터가 적용된다면 출력 데이터는 n개의 채널을 갖게 된다.

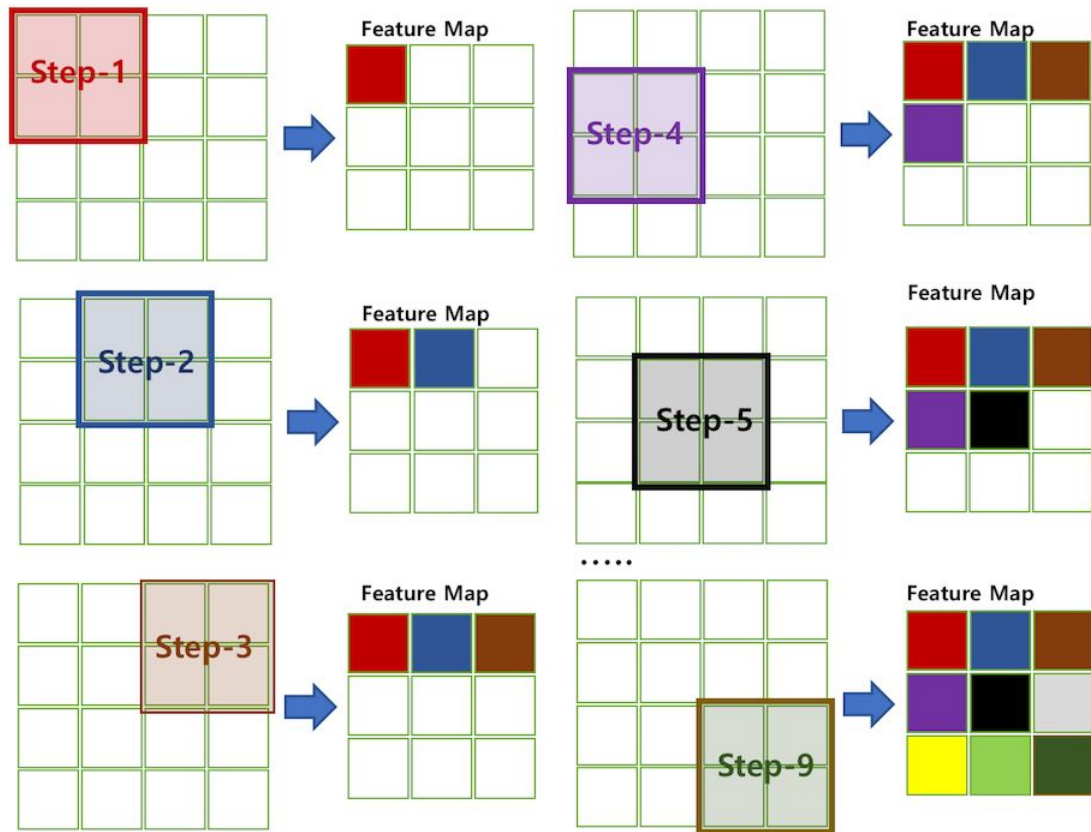
필터(filter) & stride

- 필터는 이미지의 특징을 찾아내기 위한 공용 파라미터입니다. filter를 kernel이라고 하기도 합니다.
- 필터는 일반적으로 (4,4)이나(3,3)과 같은 정사각 행렬로 정의한다
- CNN에서 학습의 대상은 필터 파라미터 입니다. 입력 데이터를 지정된 간격으로 순회하며 채널별로 합성곱을 하고 모든 채널(컬러의 경우 3개)의 합성곱의 합을 Feature Map로 만듭니다. 필터는 지정된 간격으로 이동하면서 전체 입력데이터와 합성곱하여 Feature Map을 만듭니다.

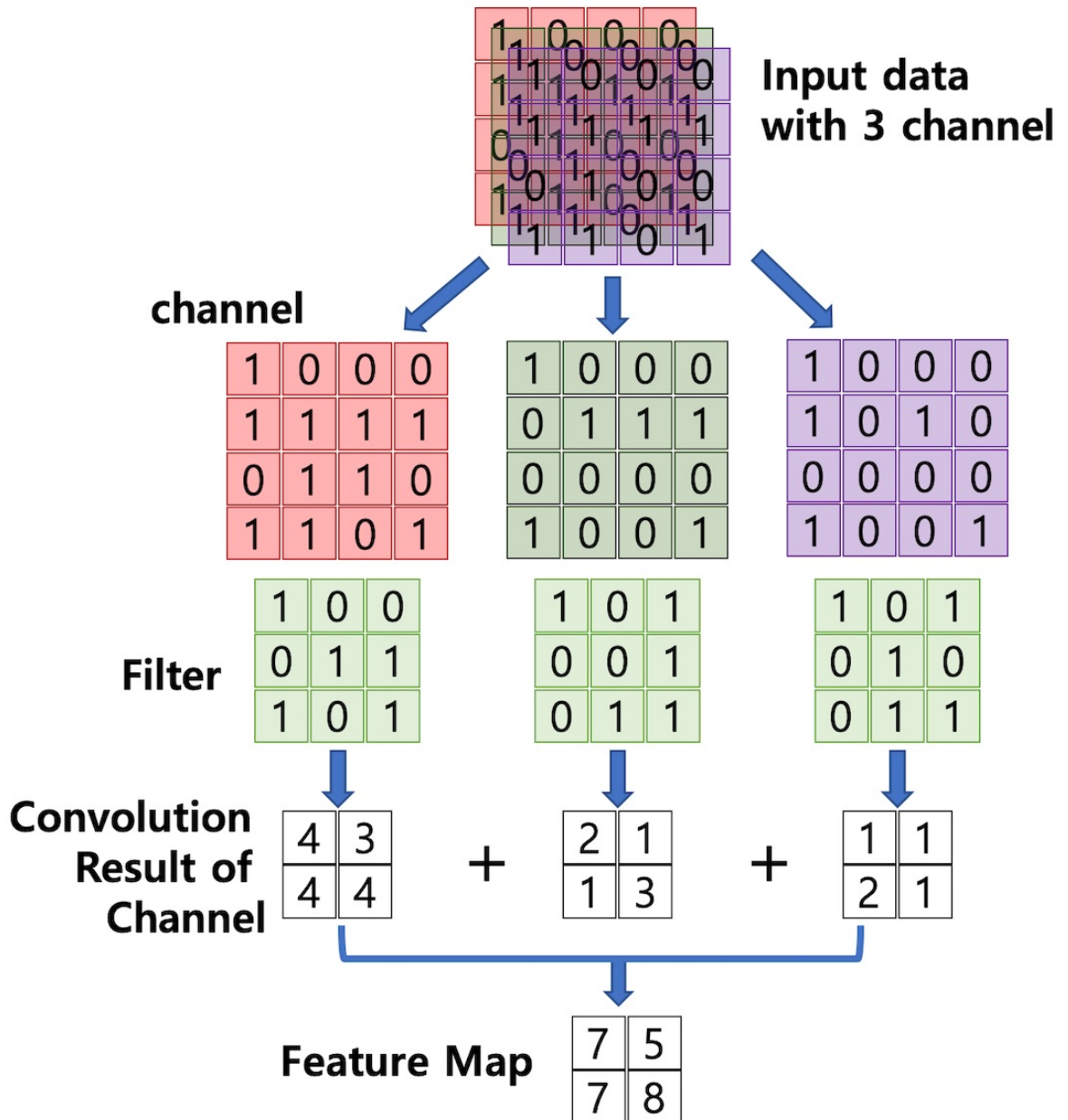


<http://taewan.kim>

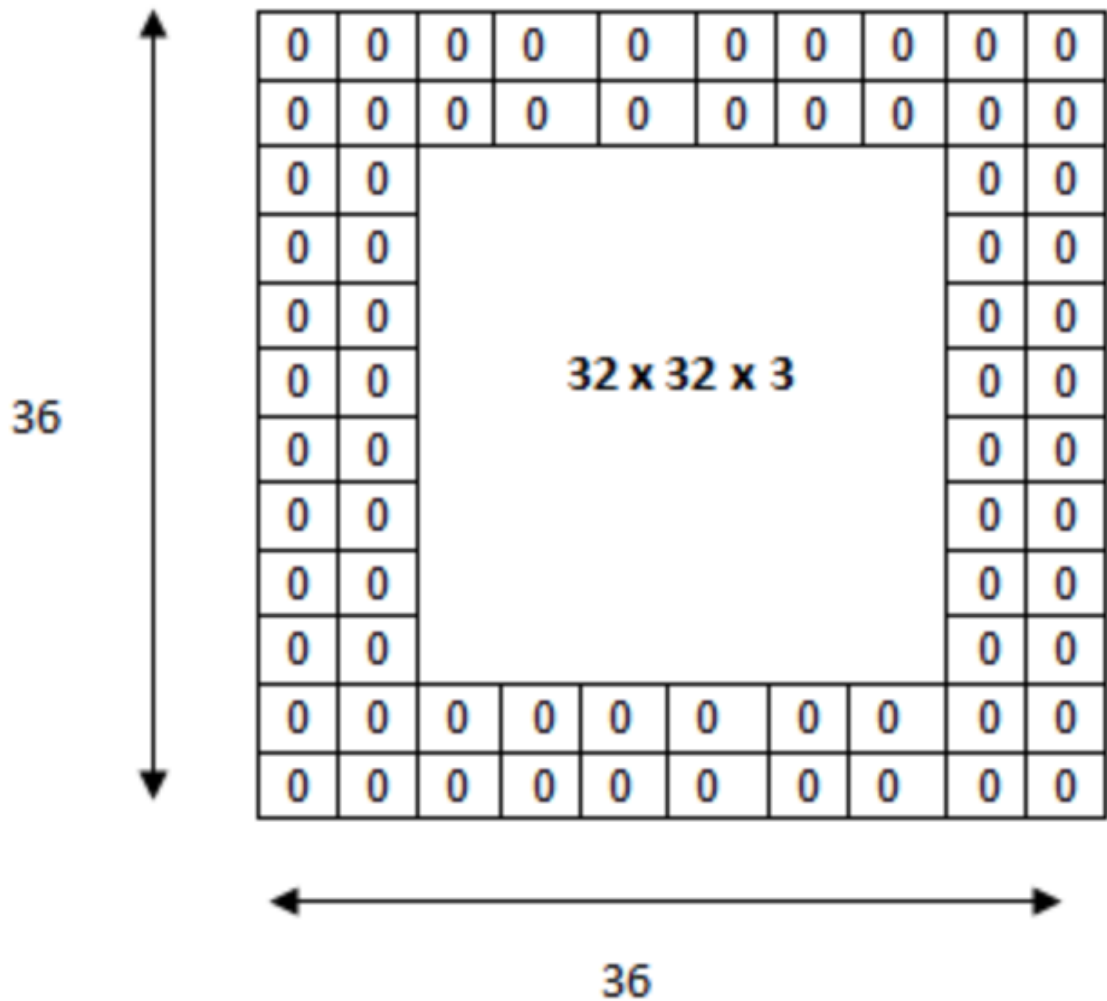
- 필터는 입력 데이터를 지정한 간격으로 순회하면서 합성곱을 계산합니다. 여기서 **지정된 간격으로 필터를 순회하는 간격을 Stride**라고 합니다.



-
- stride가 2가 되면 2칸씩 이동한다.
- 애플리케이션이 여러 채널을 갖는 경우 필터는 각 채널을 순회하며 합성곱을 계산한 후 채널별 피쳐 맵을 만듭니다. 그리고 각 채널의 피쳐 맵을 합산하여 최종 피쳐 맵으로 반환합니다.
- 입력 데이터는 채널 수와 상관없이 필터 별로 1개의 피쳐맵이 만들어 집니다.



- 하나의 Convolution layer에 크기가 같은 여러 개의 필터를 적용할 수 있습니다. 이 경우에 Feature Map에는 필터 갯수 만큼의 채널이 만들어집니다. 입력데이터에 적용한 필터의 개수는 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 활성화 함수를 적용한 결과입니다. 즉 Convolution 레이어의 최종 출력 결과가 Activation Map입니다.
- **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.



-
- **pooling**
 - 이미지의 크기를 계속 유지한 채 FC(Fully Connected)레이어로 가게 된다면 연산량이 기하급수적으로 늘것입니다.
 - 적당히 크기도 줄이고, 특정 feature를 강조할 수 있어야 하는데 그 역할을 Pooling 레이어에서 하게 됩니다.

13	20	30	0
8	12	3	0
34	70	33	5
111	80	10	23

Activation Map

20	30
111	33

Max Pooling

13	8
66	18

Average Pooling

8	0
34	5

Min Pooling

-
- 그림과 같이 3가지 방법이 있다.

레이어별 출력 데이터 산정

Convolution 레이어와 pooling 레이어의 출력 데이터 크기를 계산하는 방법

- **Convolution**레이어 출력 데이터 크기 산정

- 입력 데이터에 대한 필터의 크기와 Stride 크기에 따라서 Feature Map 크기가 결정됩니다. 공식은 다음과 같습니다.
- 입력 데이터 높이: H
- 입력 데이터 폭: W
- 필터 높이: FH
- 필터 폭: FW
- Strid 크기: S
- 패딩 사이즈: P

◦ 식 1 :

$$\text{outputHeight} = OH = \frac{(H + 2P - FH)}{S}$$

$$\text{outputWeight} = OW = \frac{(W + 2P - FW)}{S}$$

- <식1>의 결과는 자연수가 되어야 합니다. 또한 Convolution레이어 다음에 Pooling레이어가 온다면 Feature Map의 행과 열 크기를 Pooling크기의 배수여야 합니다. 만약 Pooling사이즈가 (3,3)이라면 위 식의 결과는 자연수이고 3의 배수여야 합니다. 이 조건을 만족하도록 Filter의 크기 Stride의 간격 Pooling크기 및 패딩 크기를 조절해야 합니다.

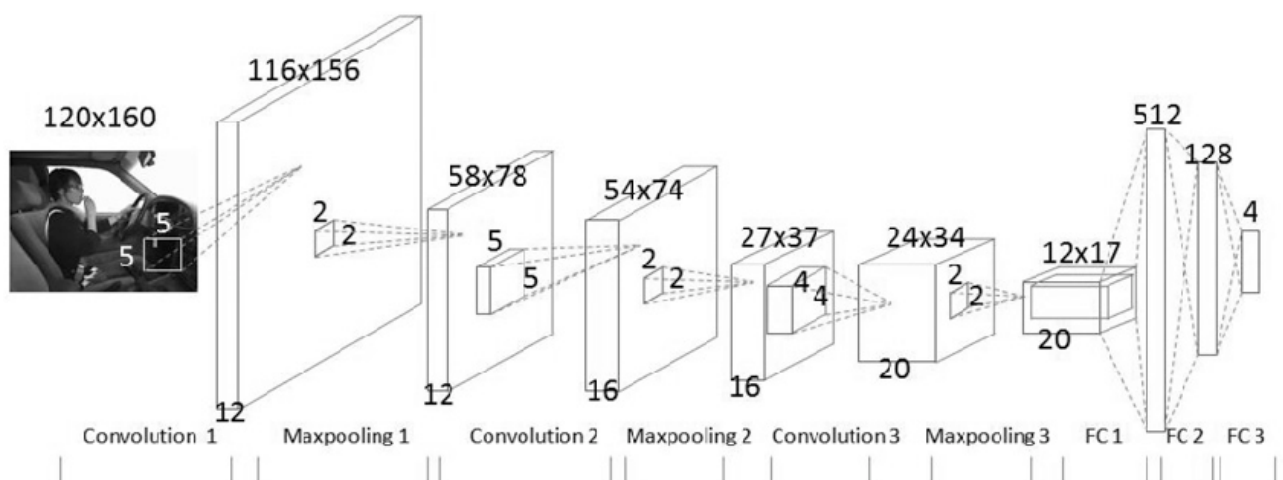
• Pooling레이어 출력 데이터 크기 산정

- Pooling 레이어에서 일반적인 Pooling 사이즈를 정사각형입니다. Pooling 사이즈를 Stride 같은 크기로 만들어서, 모든 요소가 한번씩 Pooling되도록 만듭니다. 입력 데이터의 행 크기와 열 크기는 Pooling 사이즈의 배수(나누어 떨어지는 수)여야 합니다. 결과적으로 Pooling 레이어의 출력 데이터의 크기는 행과 열의 크기를 Pooling 사이즈로 나눈 몫입니다. Pooling 크기가 (2, 2) 라면 출력 데이터 크기는 입력 데이터의 행과 열 크기를 2로 나눈 몫입니다. pooling 크기가 (3, 3)이라면 입력데이터의 행과 크기를 3으로 나눈 몫이 됩니다.

$$\text{OutputRowSize} = \frac{\text{InputRowsize}}{\text{PoolingSize}}$$

$$\text{OutputColumnSize} = \frac{\text{InputColumnsize}}{\text{PoolingSize}}$$

CNN구성



- 전형적인 CNN 구성입니다. CNN은 Convolution Layer와 Max Pooling 레이어를 반복적으로 stack을 쌓는 특징 추출(Feature Extraction) 부분과 Fully Connected Layer를 구성하고 마지막 출력층에 Softmax를 적용한 분류 부분으로 나뉩니다
- CNN을 구성하면서 Filter, Stride, Padding을 조절하여 특징 추출(Feature Extraction) 부분의 입력과 출력 크기를 계산하고 맞추는 작업이 중요합니다.

CNN 입출력, 파라미터 계산

- 입력데이터 Shape: (39, 31, 1)
- 분류 클래스: 100

layer	Input Channel	Filter	Output Channel	Stride	Max Pooling	activation function
Convolution Layer 1	1	(4, 4)	20	1	X	relu
Max Pooling Lyaer 1	20	X	20	2	(2, 2)	X
Convolution Layer 2	20	(3, 3)	40	1	X	relu
Max Pooling Lyaer 2	40	X	40	2	(2, 2)	X
Convolution Layer 3	40	(2, 2)	60	1	1	relu
Max Pooling Lyaer 3	60	X	60	2	(2, 2)	X
Convolution Layer 4	60	(2, 2)	80	1	1	relu
Flatten	X	X	X	X	X	X
fully connected Layer	X	X	X	X	X	softmax

- 컨볼루션 레이어의 학습 파라미터 수는 “입력채널수X필터폭X필터높이X출력채널수”로 계산됩니다.

• Convolution Layer 1

- 입력 데이터 Shape = (39, 31, 1)
- 입력 채널=1
- 필터=(4, 4)
- 출력 채널=20
- tride = 1

- 입력 이미지에 Shape이 (4, 4)인 필터 20개를 적용할 경우에, 출력 데이터(Activation Map)의 Shape을 계산하는 과정은 <식 3>과 같습니다.

$$\text{RowSize} = \frac{N-F}{\text{strid}} + 1 = \frac{39 - 4}{1} + 1 = 36$$

$$\text{ColumnSize} = \frac{N-F}{\text{strid}} + 1 = \frac{31 - 4}{1} + 1 = 28$$

- 계산된 출력 데이터(Activation Map)의 Shape은 (36, 28, 20) 입니다. Convolution Layer 1에서 학습 시킬 대상은 입력 채널 1, 필터 사이즈 (4, 4), 출력 채널 20개 입니다. 따라서 이 레이어의 학습 파라미터는 320개 (4X4X20) 입니다.
- 입력 채널: 1
- 출력 데이터(Activation Map) Shape: (36, 28, 20)
- 학습 파라미터: 320개 (1 X 4 X 4 X 20)

• Max Pooling Layer 1

- Max Pooling Layer 1의 입력 데이터의 Shape은 (36, 28, 20)입니다. Max Pooling 크기가 (2, 2)이기 때문에 출력 데이터 크기는 밑의 식과 같이 계산된다

$$\text{RowSize} = \frac{36}{2} = 18$$

$$\text{ColumnSize} = \frac{28}{2} = 14$$

- 계산된 출력 데이터의 Shape은 (18, 14, 20) 입니다. Max Pooling Layer에서 학습 파라미터가 없습니다.
- 입력 채널: 20
- 출력 데이터 Shape: (18, 14, 20)
- 학습 파라미터: 0

- 이런 방법으로 4번 레이어까지 끝낸다

- Flatten Layer의 Shape

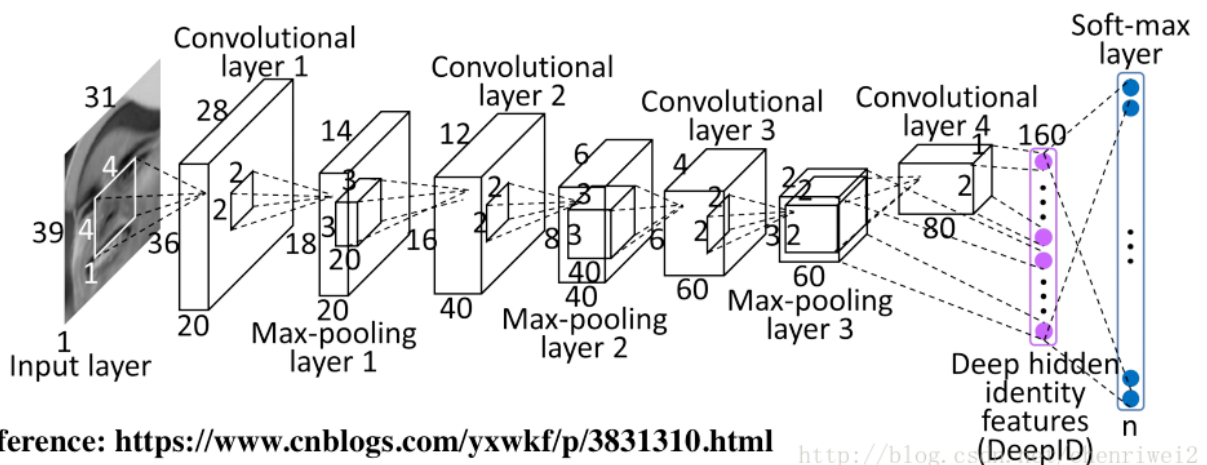
- Flatten Layer는 CNN의 데이터 타입을 Fully Connected Neural Network의 형태로 변경하는 레이어입니다. Flatten 레이어에는 파라미터가 존재하지 않고, 입력 데이터의 Shape 변경만 수행합니다.
- 입력 데이터 Shape =(2, 1, 80)
- 출력 데이터 Shape =(160, 1)

- Softmax Layer

- 이 레이어의 입력 데이터 Shape은 (160, 1)입니다. 이 네트워크의 분류 클래스가 100개이기 때문에 최종 데이터의 Shape은 (100, 1)입니다.
- 입력 데이터의 shape: (160, 1)
- 출력 데이터의 shape: (100, 1)
- 이때 Weight Shape은 (100, 160)입니다. Softmax 레이어의 파라미터는 160,000개 (100X160)입니다.

- 체 파라미터 수와 레이어별 Input/Output 요약

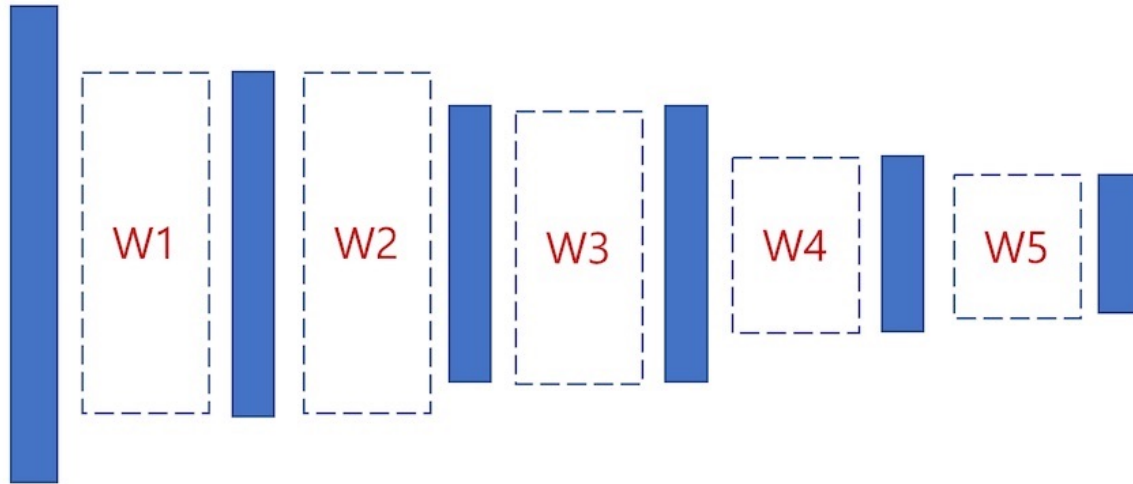
layer	input channel	Filter	output channel	Stride	Pooling	활성함 수	Input Shape	Output Shape	파라미터 수
Convolution Layer 1	1	(4, 4)	20	1	X	relu	(39, 31, 1)	(36, 28, 20)	320
Max Pooling Layer 1	20	X	20	2	(2, 2)	X	(36, 28, 20)	(18, 14, 20)	0
Convolution Layer 2	20	(3, 3)	40	1	X	relu	(18, 14, 20)	(16, 12, 40)	7,200
Max Pooling Layer 2	40	X	40	2	(2, 2)	X	(16, 12, 40)	(8, 6, 40)	0
Convolution Layer 3	40	(2, 2)	60	1	1	relu	(8, 6, 40)	(6, 4, 60)	21,600
Max Pooling Layer 3	60	X	60	(2, 2)	60	X	(6, 4, 60)	(3, 2, 60)	0
Convolution Layer 4	60	(2, 2)	80	1	1	relu	(3, 2, 60)	(2, 1, 80)	19,200
Flatten	X	X	X	X	X	X	(2, 1, 80)	(160, 1)	0
fully connected Layer	X	X	X	X	X	softmax	(160, 1)	(100, 1)	160,000
합계	X	X	X	X	X	softmax	(160, 1)	(100, 1)	208,320



CNN과 FC Neural Network 파라미터 비교

앞에서 다룬 CNN의 총 파라미터 수는 208,320개입니다. 이 CNN과 유사한 4개의 은닉 레이어를 갖는 FC(Fully Connected) Neural Network는 다음과 같습니다.

Input layer Layer 1 Layer 2 Layer 3 Layer 4 Output Layer
 (1209, 1) (600, 1) (300, 1) (300, 1) (150, 1) (100, 1)



<http://taewan.kim>

레이어	입력 노드	출력 노드	Weight Shape	파라미터 수
Layer 1	1209	600	(1209,600)	725,400
Layer 2	600	300	(600,300)	180,000
Layer 3	300	300	(300,300)	90,000
Layer 4	300	150	(300,150)	45,000
Output	150	100	(150,100)	15,000
합계				1,055,400

인공 신경

망의 총 파라미터는 백만 개가 넘습니다. 예제 CNN 파라미터와 비교하면 10배 이상의 학습 파라미터를 갖습니다. 은닉층을 더 깊게 만들 경우 Fully Connected Neural Network과 CNN과의 학습 파라미터의 차이는 더 급격하게 늘어납니다. CNN은 Fully Connected Neural Network과 비교하여 다음과 같은 특징을 갖습니다.

- CNN은 학습 파라미터 수가 매우
- 학습 파라미터가 작고, 학습이 쉽고 네트워크 처리 속도가 빠름

참조

- <http://taewan.kim/post/cnn/>
- <https://excelsior-cjh.tistory.com/180>
- <https://all-young.tistory.com/43>
- <https://gruuuuu.github.io/machine-learning/cnn-doc/>