

Pooling Layer

작성일/작성자: 2025.04.02 / 김소현

CNN 기존 구조



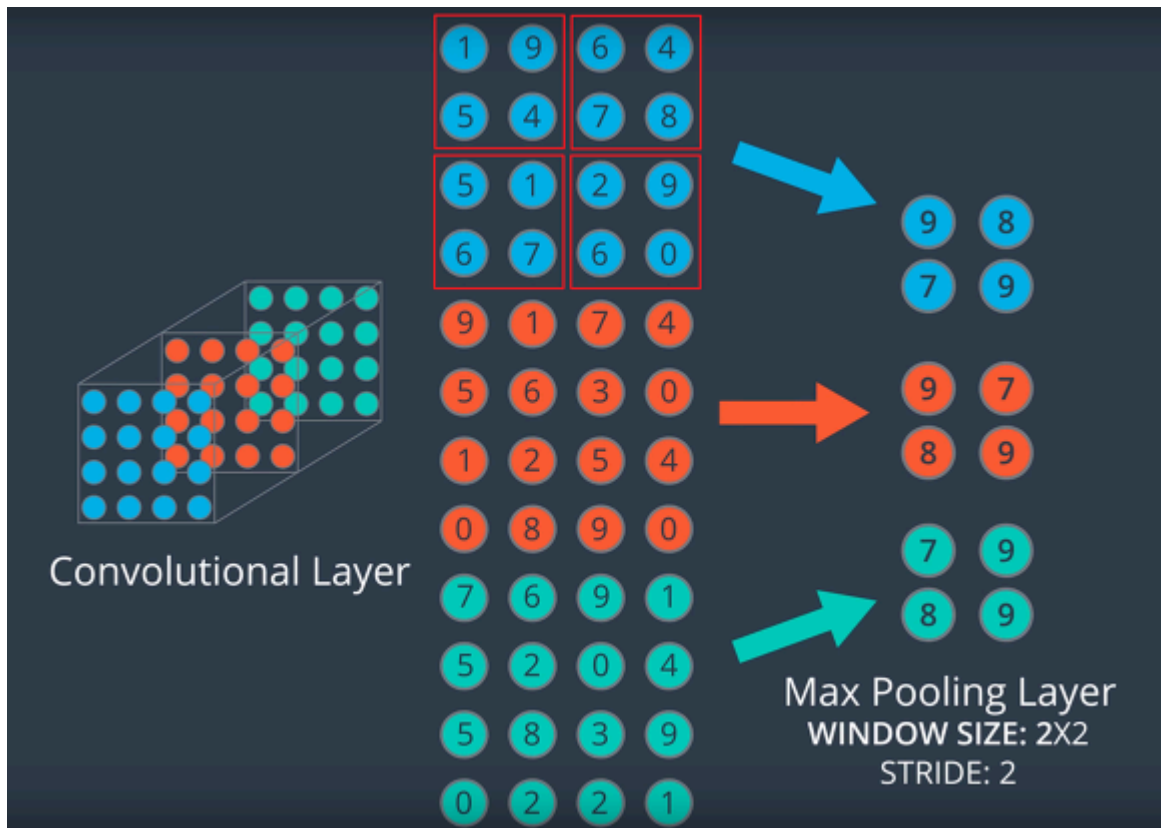
- **Input:** 원본 이미지 들어옴
 - **CONV (Convolution Layer):** Input에 대한 Feature 추출
 - 각 필터당 하나의 Feature map 형성되고, 그 feature map을 Stack한 것
 - 고차원(Higher dimensionality)을 구현하려면
그에 상응하는 더 많은 수의 파라미터들 필요, Over-fitting을 초래할 수 있음
- ⇒ 이때 필요한 것이 **Pooling Layer**
- **POOL (Pooling Layer):** 크기를 줄이고 중요 정보만 유지

>> 이 과정을 여러 번 반복해서 패턴을 인식, 복잡한 특징 학습하도록 구성

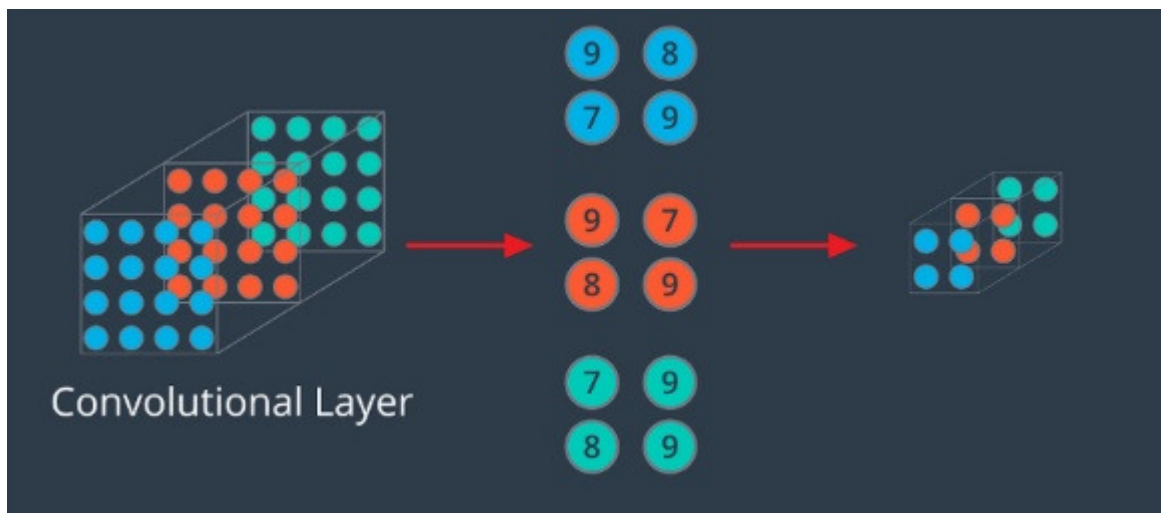
Pooling Layer

(1) Max Pooling Layer

- 각 feature map에서 가장 큰 값(최댓값) 하나만 뽑아내는 방식
- window size와 stride 지정 필요



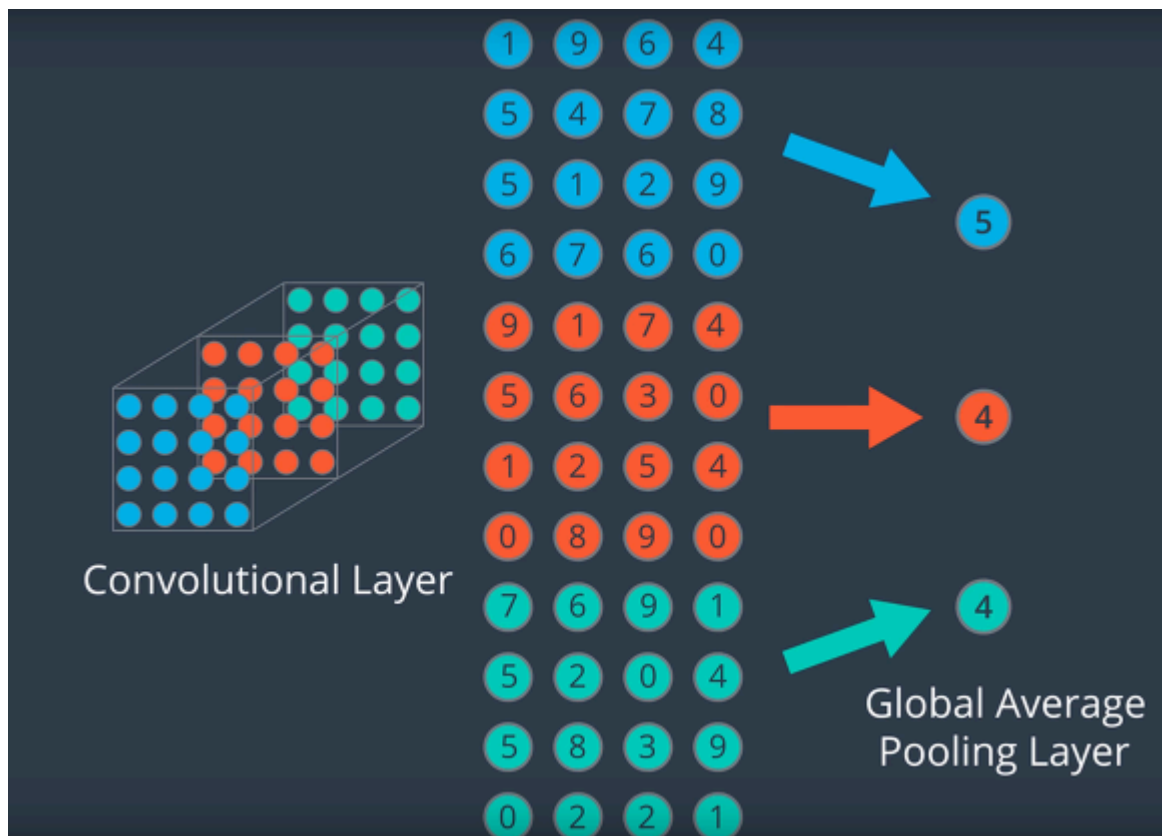
- **Window size** : 2 by 2 / **Stride** : 2 / 좌측 상단부터 시작
- window상에서 포함하고있는 픽셀들 중 최대값을 뽑아냄



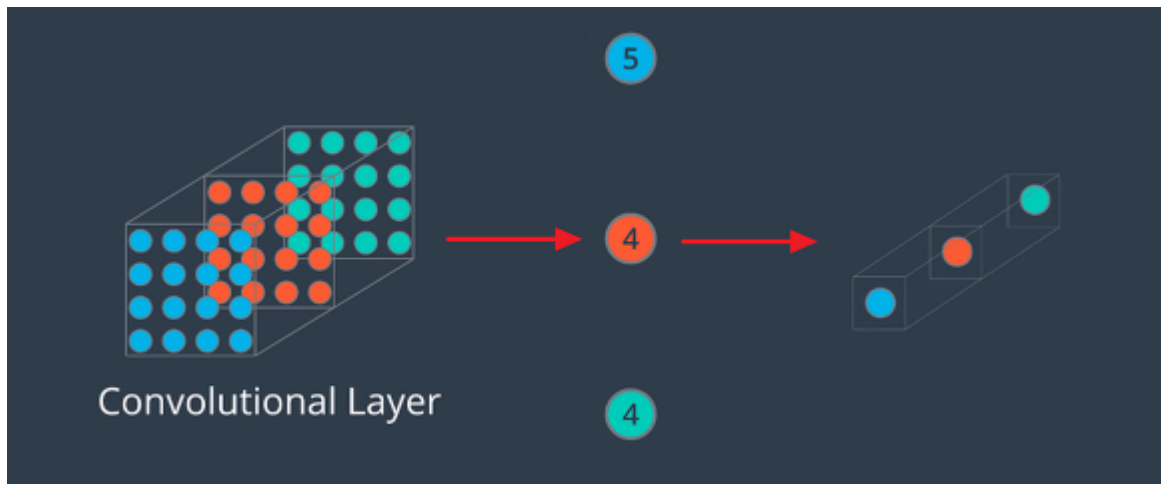
- **output**
 - feature map의 개수와 동일한 개수의 컬렉션을 리턴
 - 이전 input보다 너비와 높이가 감소됨
- ⇒ 최댓값들로 이루어진 축소된 feature map

(2) Global average pooling layer

- 각 feature map의 평균값을 뽑아서 벡터로 요약하는 방식
- window size나 stride 지정 X
- 모델의 마지막 부분에서 사용

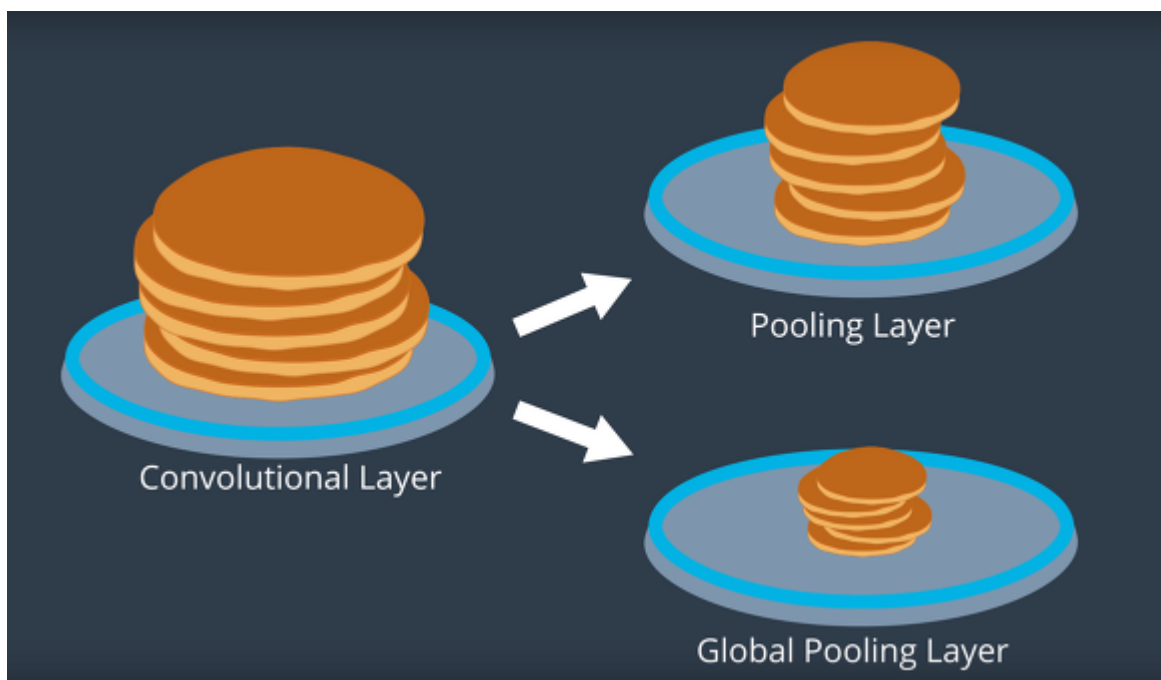


- 각 feature map 상의 노드값들의 평균을 뽑아냄
ex) 파랑: feature map 총 합 = 80, node 전체 개수 = 16 \Rightarrow 평균은 5



- **output**
 - single value로 크기가 감소된 feature map
 - 벡터 ex) [5, 4, 4]

요약



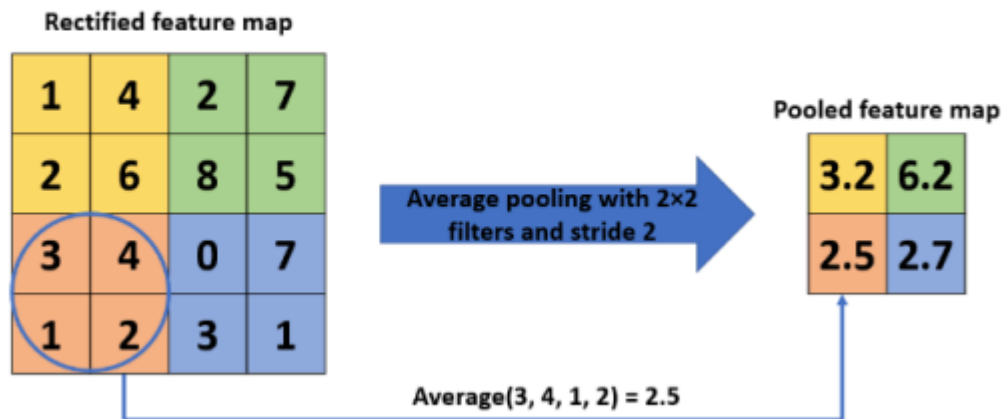
- 두 pooling layer 모두 input의 차원을 줄여줌
- **Max pooling layer**
 - 영역에서 제일 큰 값만 가져옴 ⇒ 가장 눈에 띄는 특징만 유지
 - 적절하게(지정해준 window와 stride의 값에 상당한) 차원을 줄여줌

- **Average pooling**

- 평균값을 가져옴 ⇒ 전체적으로 부드러운 정보 유지
- output의 차원을 single value로 급격하게 줄여줌

(3) Average Pooling layer

- window와 stride를 사용해 부분 평균을 사용하는 방식
- 중간 Pooling layer에서 사용



- **Output**

- feature map의 개수와 동일한 개수의 컬렉션을 리턴
- 이전 input보다 너비와 높이가 감소됨

⇒ 평균값들로 이루어진 축소된 feature map

- **Average Pooling vs. Global Average Pooling**

- **Average Pooling:** 이미지의 크기를 줄이기 위한 단계적인 압축
- **Global Average Pooling:** 최종적으로 feature map 전체를 하나의 값으로 요약

참고 자료

(1) Max Pooling vs. Global Average Pooling

<https://kevinthegrey.tistory.com/142>

(2) Average Pooling <https://blog.paperspace.com/a-comprehensive-exploration-of-pooling-in-neural-networks/>