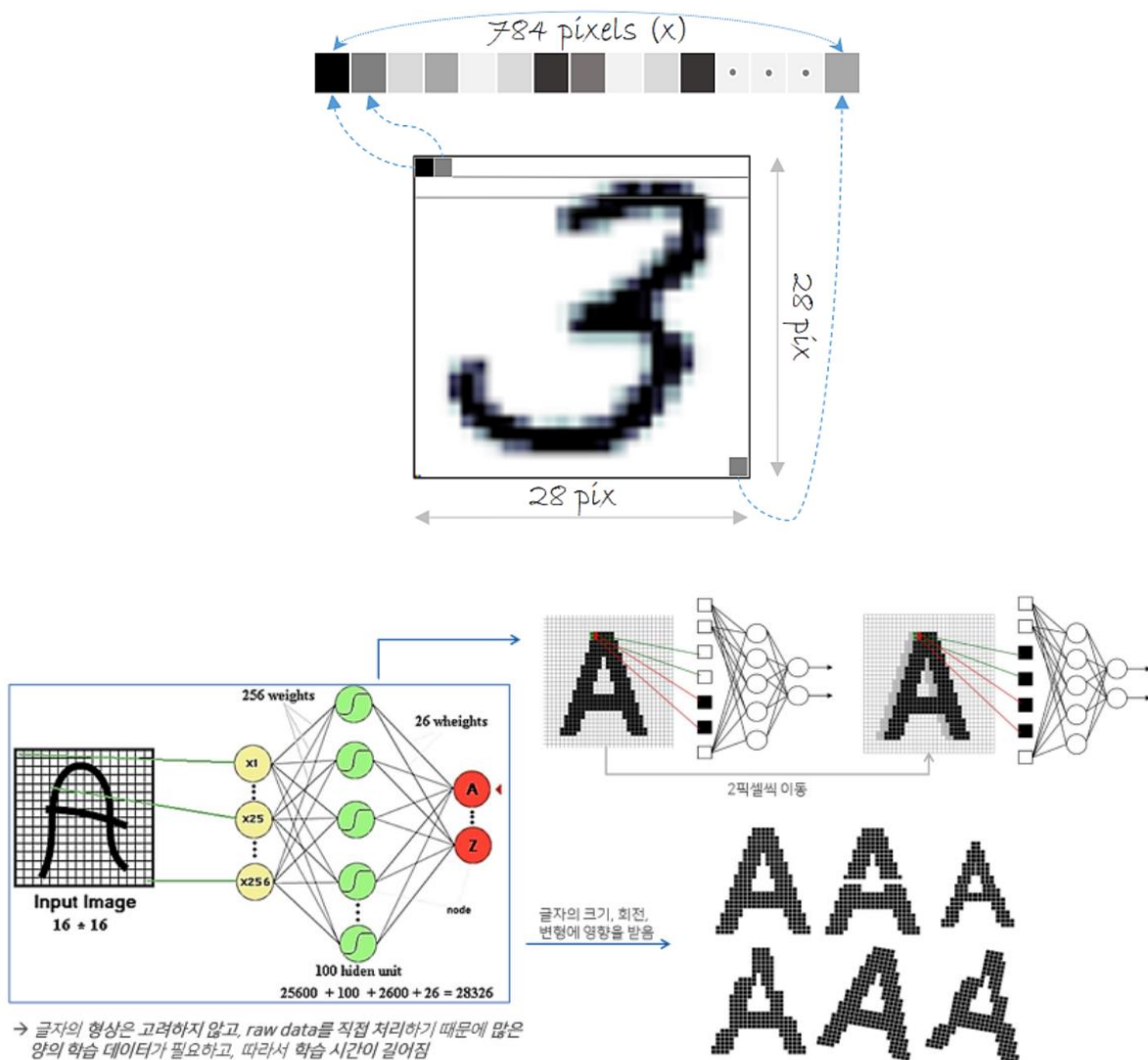


CNN

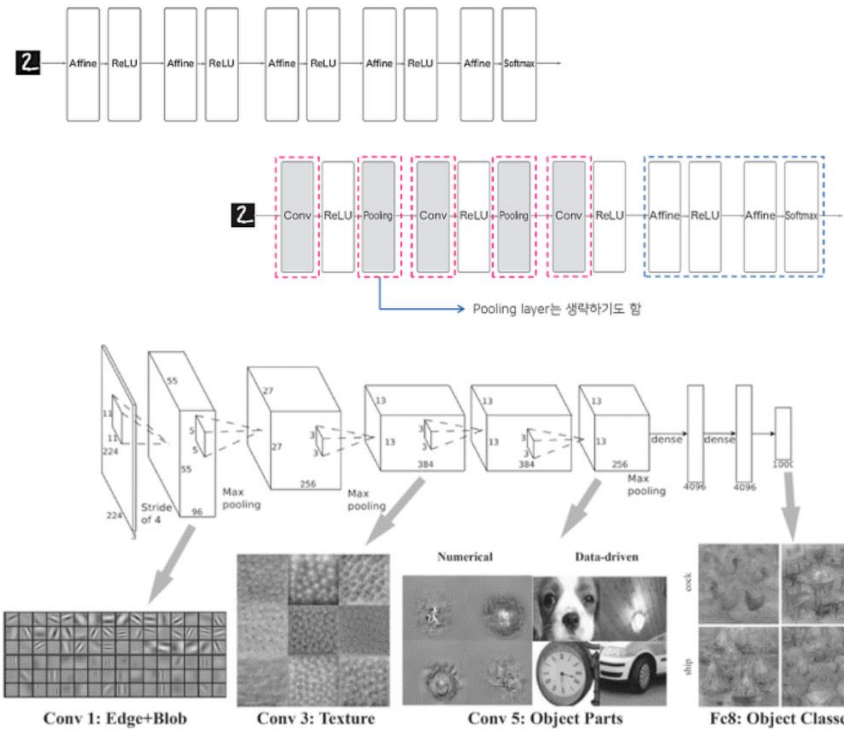
1. CNN이 나오게 된 배경

기존 완전연결 계층(fully connected layer)을 이용해 모델학습을 진행하기 위해서는 3차원의 이미지를 1차원으로 변형한 후 사용해야하는 문제점 → 데이터의 형상(공간적 구조)이 무시됨, 파라미터의 개수가 굉장히 많음



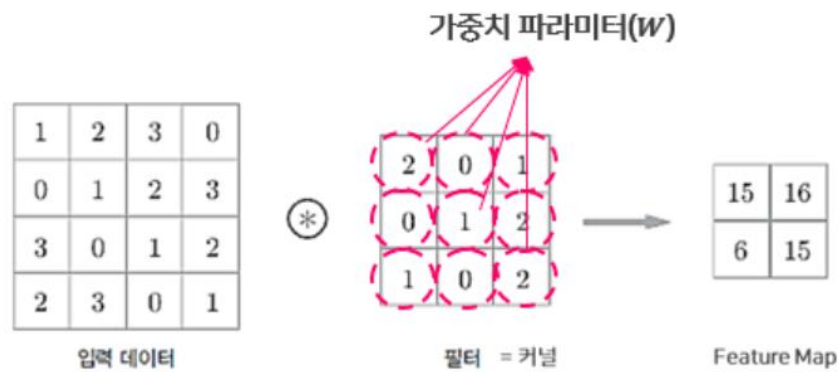
[그림 2] MLNN의 문제점

2. CNN



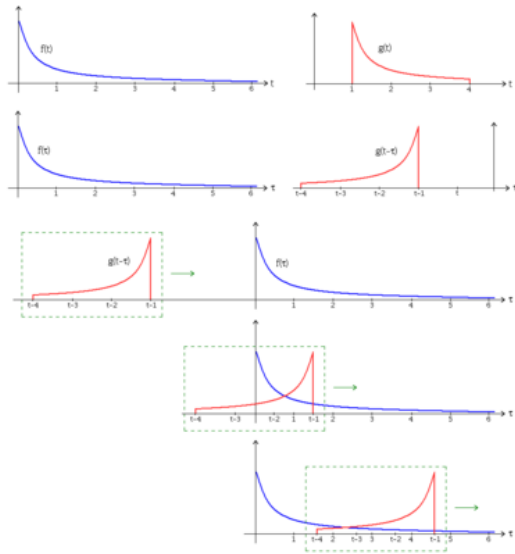
필터를 활용하여 합성곱을 진행하여 특징맵을 추출함 → 공간적 구조를 고려하며 학습이 가능

3. 필터



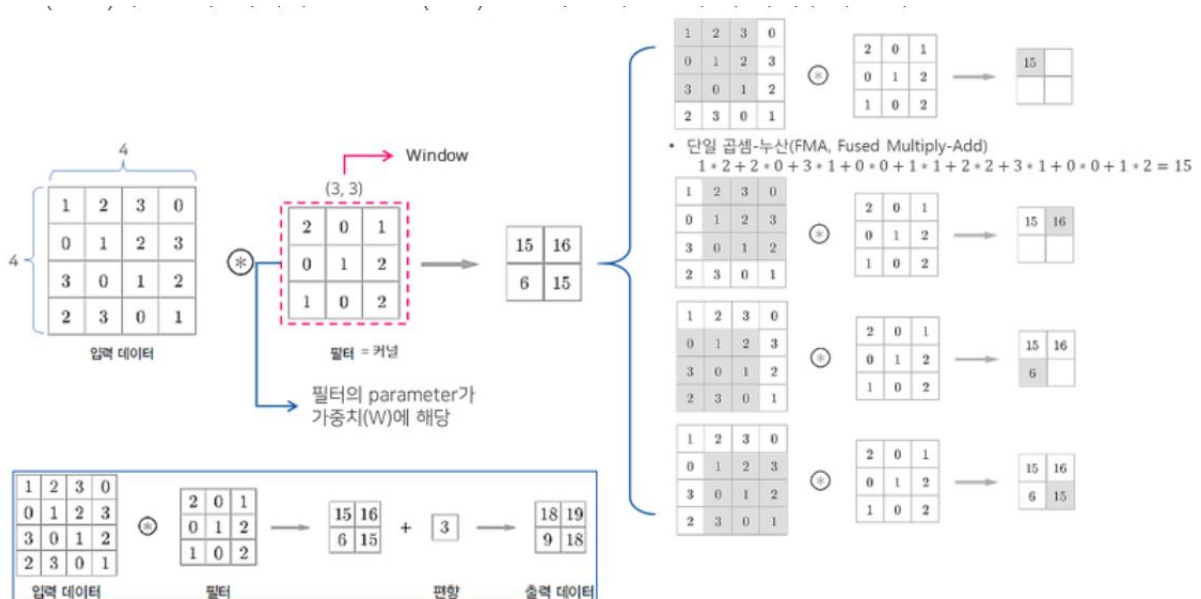
필터를 이용하여 합성곱 진행, 특징맵 생성 (필터 값들을 조정하는 과정 = 모델 학습)

4. 합성곱



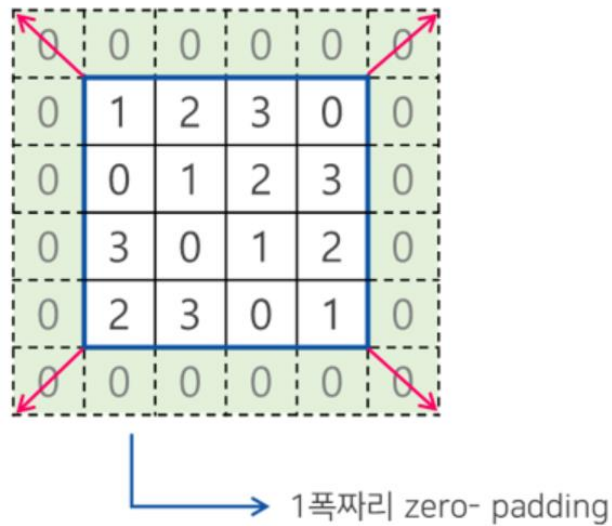
$$(f * g)(t) = \int_{-\infty}^{\infty} f(\tau)g(t - \tau)d\tau$$

사전 말은 어려워 보이는데 입력 데이터와 필터의 원소 별 곱이라고 보면 될 거 같다.



4. 패딩, 스트라이드

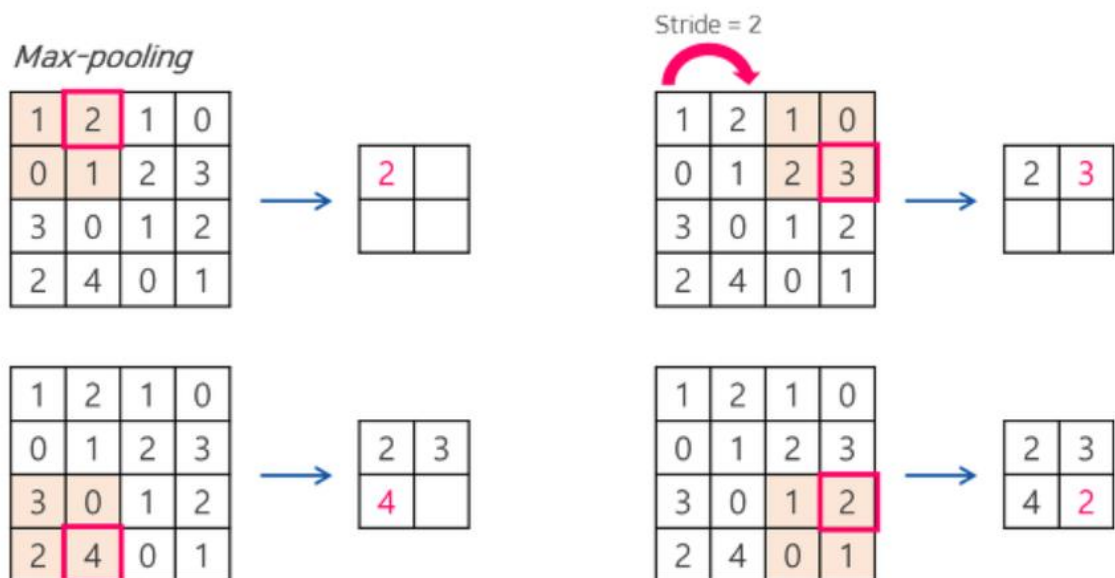
합성곱 과정을 거치게 되면 사이즈가 작아지는 경우가 발생해 가장자리의 정보가 소실되는 문제가 발생할 수도 있는데 패딩이라는 과정을 거침으로써 이를 방지해준다.

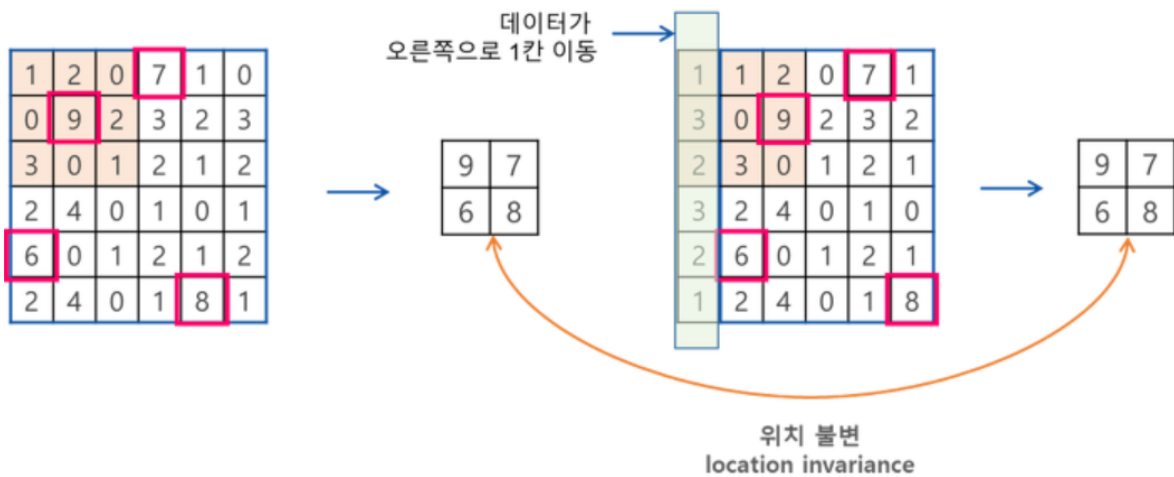


스트라이드는 필터가 이동하는 간격을 나타내는 것으로 이에 따라 특징맵의 크기가 조정된다.

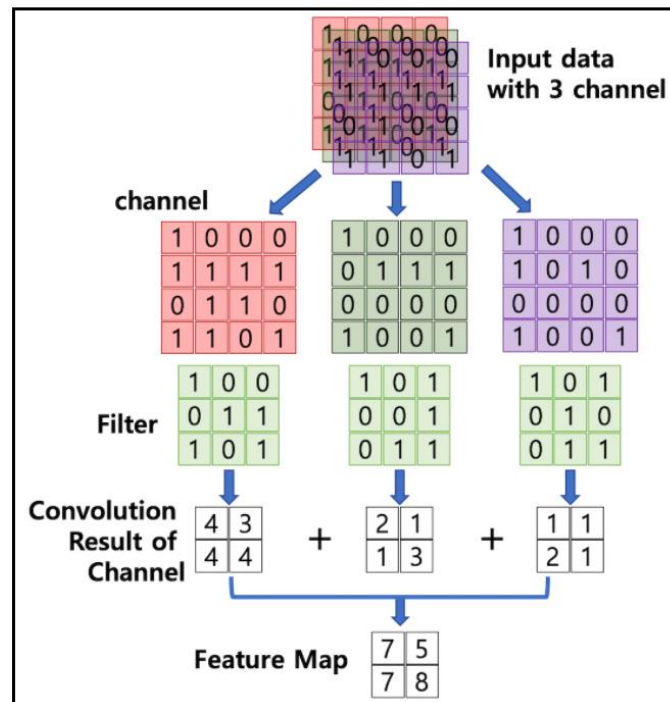
5. 풀링층

사이즈가 너무 큰 경우나 어느 특징만을 좀 더 추출하고 싶은 경우 합성곱 이후 풀링층을 활용
최댓값, 평균 방법 활용. 공간적 변화를 극복하기 위해서도 사용





6. 채널이 여러 개일 경우



7. 사진 및 자료 출처

<http://taewan.kim/post/cnn/>

<https://excelsior-cjh.tistory.com/79?category=1013831>

<https://excelsior-cjh.tistory.com/180>

https://tykimos.github.io/2017/01/27/CNN_Layer_Talk/