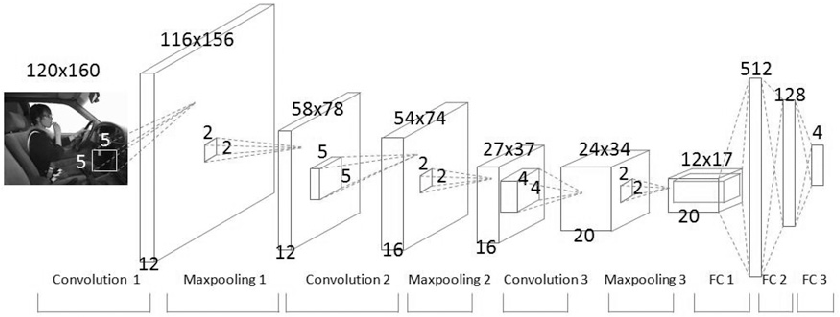
CNN 공부

* 기존 Multi-Layer Neural Network의 단점: 이미지 데이터를 처리할 때 정보를 1차원으로 받아야 하는데 이때 이미지의 상하관계의 2차원 정보가 손실된다. 이미지의 대상이 조금 움직였을 때에 대해서도 한 픽셀씩 이동한 새로운 데이터로 인식을 하여 학습이 오래 걸린다. 또 해당 이미지 대상의 크기, 회전, 변형에 대해서도 모두 학습하여야 하기 때문에 처리량이 너무 많아진다. 이러한 단점들을 Convolutional Layer를 통해 극복할 수 있다.

합성 곱 계층(Convolutional Layer)

* 기존 MLNN에서는 이미지의 3차원 데이터(채널,세로,가로)를 1차원 데이터로 바꿔 입력하였으나 Convolutional Layer에서는 3차원 데이터 그대로 입력하고 출력하여 형상을 유지할 수 있다.
* 방법 : 입력 데이터에 대해서 n\*n 필터(filter, kernel이라고도 함)를 통해 합성 곱 연산을 하여 feature map을 만든다. 적절한 필터를 통해 해당 부분과 그 주변의 데이터를 포함하는 feature map이 생성되고 이를 통하여 이미지 정보를 학습할 수 있다. 이 과정에서 연산을 도와주는 몇 가지 기능이 있다. 패딩(Padding), 스트라이드(Stride), 풀링(Pooling)이다.
* 패딩(Padding) : 합성 곱 연산을 하기 전에 데이터의 가장자리에 특정 값을 채우는 것이다. 패딩 없이 n\*n커널로 합성 곱 연산을 진행하면 가장자리의 데이터는 보존되지 않는다.
* 스트라이드(Stride) : 합성 곱 연산을 할 때 커널을 일정 간격으로 띄어가며 계산하는 것이다. Stride가 1이면 가장 원본과 가까운 데이터를 가져올 수 있으나 연산이 그만큼 느리므로 적절히 조절해준다.
* 풀링 계층(Pooling layer) : 스트라이드처럼 데이터 크기를 줄이는데 사용하는 계층이다. n\*n의 윈도우 크기로 해당 윈도우에서 Max값 또는 average값을 추출한다. 여기에도 스트라이드를 적용하여 일정 간격을 건너뛸 수 있다.

최종 CNN(Convolutional Neural Network) 구성

**그림 8**: 전형적인 CNN, 출처: <https://www.researchgate.net/figure/Architecture-of-our-unsupervised-CNN-Network-contains-three-stages-each-of-which_283433254>

다음과 같이 Convolutional Layer를 통해 원본 이미지에서 특징을 추출한 작은 feature map을 구성하고 이 데이터에서 fully connected layer를 통하여 학습을 하게 된다. 이러한 과정을 통해서 훨씬 적은 파라미터로 빠르게 이미지 학습이 가능하다.