

CNN

CNN (Convolutional Neural Network) - 합성곱 신경망

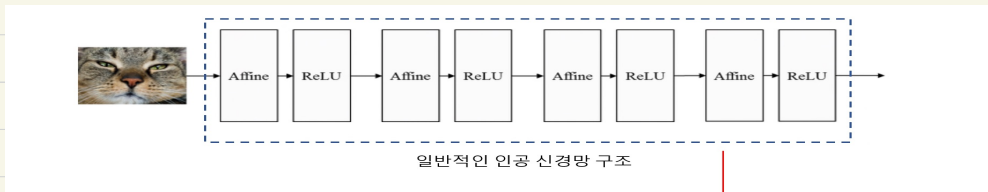
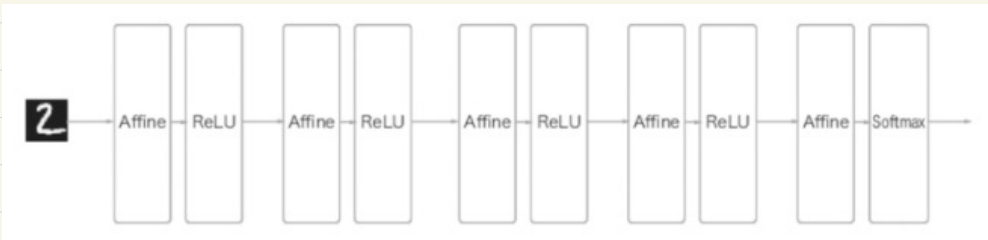
- 이미지를 분류하기 위해 패턴을 찾는 데 유용한 알고리즘
- 데이터에서 이미지로 직접 학습하고 패턴을 사용해 이미지를 분류
- 이미지의 공간 정보를 유지하며 학습
- "행렬로 표현된 픽셀의 각 row가 데이터 층에 적용하도록 자동으로 학습하게 해"

일반적인 신경망은 affine로 명시된 fully-connected 연산과 ReLU와 같은 비선형 활성화 함수 (nonlinear activation function)의 합성으로 정의된 여러 계층을 쌓은 구조이다.

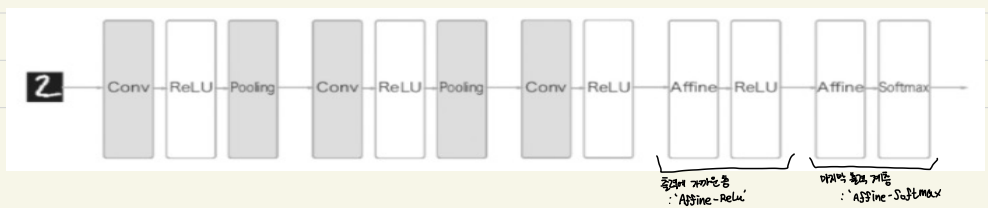
affine : 점, 직선, 평면을 보존하는 선형 매핑 방법

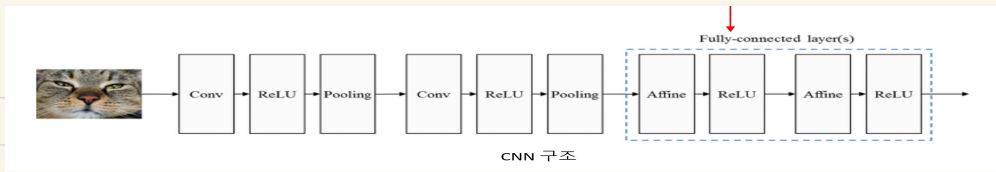
선형 매핑 : 어떤 벡터 공간에 대한 해의 공간으로써 함수 공간을 분해하는 성질

fully-connected 연산 : 완전 연결



CNN은 Feature를 추출하는 Convolution Layer와 추출된 Feature를 Sub-Sampling하는 Pooling Layer로 구성되어 있다.





Convolution Layer : 이미지에 필터링 계층 적용

pooling layer : 이미지의 좌측적인 부분들을 하나의 대표적인 스칼라 값으로 변환 \Rightarrow 이미지 크기를 줄여줌
다양한 계층 구성

[Convolutional Layer] : 합성곱

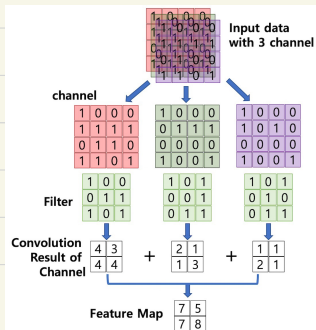
합성곱(合成-, convolution, 콘벌루션)은 하나의 함수와 또 다른 함수를 반전 이동한 값을 곱한 다음, 구간에 대해 적분하여 새로운 함수를 구하는 수학 연산자이다.

현재 위치의 출력 데이터는 인접한 pixel에 Convolution Filter를 곱해 얻은 값

단일 채널에 대한 합성곱 계층 동작

$$\begin{array}{c}
 \begin{bmatrix} 0 & 1 & 7 & 5 \\ 5 & 5 & 6 & 6 \\ 5 & 3 & 3 & 0 \\ 1 & 1 & 1 & 2 \end{bmatrix} * \begin{bmatrix} 1 & 0 & 1 \\ 1 & 2 & 0 \\ 3 & 0 & 1 \end{bmatrix} = \begin{bmatrix} 40 & \\ & \end{bmatrix} \\
 \begin{bmatrix} 0 & 1 & 7 & 5 \\ 5 & 5 & 6 & 6 \\ 5 & 3 & 3 & 0 \\ 1 & 1 & 1 & 2 \end{bmatrix} * \begin{bmatrix} 1 & 0 & 1 \\ 1 & 2 & 0 \\ 3 & 0 & 1 \end{bmatrix} = \begin{bmatrix} 40 & 32 \\ & \end{bmatrix} \\
 \begin{bmatrix} 0 & 1 & 7 & 5 \\ 5 & 5 & 6 & 6 \\ 5 & 3 & 3 & 0 \\ 1 & 1 & 1 & 2 \end{bmatrix} * \begin{bmatrix} 1 & 0 & 1 \\ 1 & 2 & 0 \\ 3 & 0 & 1 \end{bmatrix} = \begin{bmatrix} 40 & 32 \\ 26 & \end{bmatrix} \\
 \begin{bmatrix} 0 & 1 & 7 & 5 \\ 5 & 5 & 6 & 6 \\ 5 & 3 & 3 & 0 \\ 1 & 1 & 1 & 2 \end{bmatrix} * \begin{bmatrix} 1 & 0 & 1 \\ 1 & 2 & 0 \\ 3 & 0 & 1 \end{bmatrix} = \begin{bmatrix} 40 & 32 \\ 26 & 25 \end{bmatrix}
 \end{array}$$

멀티 채널 입력 데이터에 필터를 적용한 합성곱



합성곱 과정을 거치면서 이미지의 크기는 점점 작아짐 \Rightarrow 가장 작게 줄어든 픽셀들의 정보는 점점 사라짐

padding : 이미지의 가장자리에 특정값으로 생성된 픽셀을 추가 \Rightarrow 입력 이미지와 출력 이미지의 크기를 같게나 비슷하게 만듦

$$\begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 7 & 5 & 0 \\ 0 & 5 & 5 & 6 & 6 & 0 \\ 0 & 5 & 3 & 3 & 0 & 0 \\ 0 & 1 & 1 & 1 & 2 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} * \begin{bmatrix} 1 & 0 & 0 \\ 1 & 2 & 1 \\ 1 & 2 & 3 \end{bmatrix} = \begin{bmatrix} 26 & 42 & 55 & 35 \\ 34 & 41 & 33 & 28 \\ 18 & 25 & 23 & 14 \\ 3 & 9 & 8 & 8 \end{bmatrix}$$

0값을 갖는 픽셀을 추가하는 zero-padding 적용

CNN에서 zero padding 이용

[Pooling Layer] (Sub Sampling)

Pooling (sub sampling 라고도 부른다.) 레이어는 컨볼루션 레이어의 출력 데이터를 받아서 출력 데이터(Activation Map)의 크기를 줄이거나 특정 데이터를 강조하는 용도로 사용된다.

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

20	30
111	33

Max Pooling

13	8
66	18

Average Pooling

8	0
34	5

Min Pooling

Activation Map

일반적으로 pooling 과 stride를 동일하게 크게 설정하여 모든 원소에 한 번에 처리 되도록 함

Convolution Layer의 차이점

- 학습대상 파악이 더 많음
- Pooling Layer과를 통과하면 행렬의 크기 감소
- " 통해서 채널 수 변경 없음



요약하면 CNN은 입력 데이터에 대해서 필터를 통해 이미지의 특징을 추출하고 (Convolution Layer), 특징을 강화하고 이미지의 크기를 축소(Pooling Layer)한다. 이 과정을 반복하여 처리된 결과를 출력한다.