

CNN

(Convolutional Neural Network)

<https://youtu.be/Ex7rEXvIOps>

CNN이란

CNN Layer

CNN 실습

CNN이란?

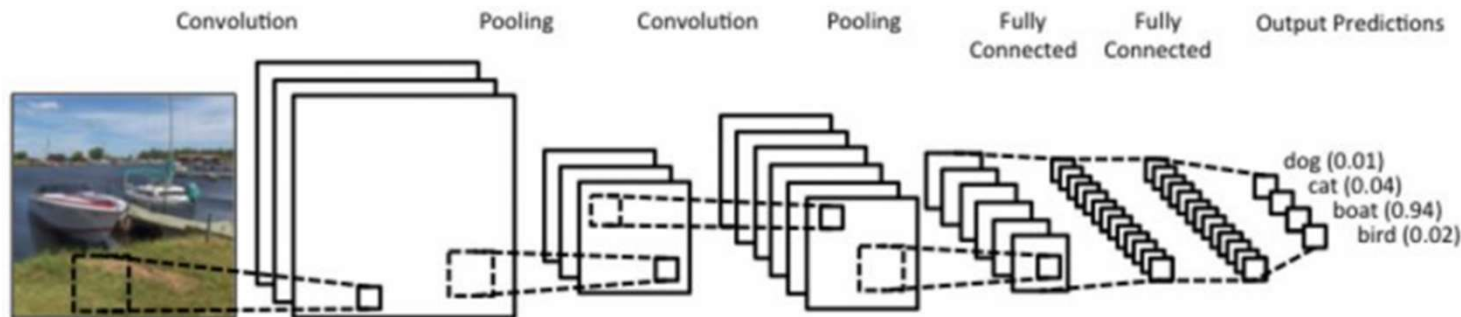
Convolutional Neural Network(CNN, ConvNet)

이미지의 공간 정보를 유지하면서 인접 이미지와의 특징을 효과적으로 인식

- 기존 신경망의 3차원 데이터를 1차원으로 변환하는 과정에서 공간정보 손실

Convolution Layer와 Pooling Layer를 통해 이미지의 특징을 추출

마지막 단의 Fully Connected Layer(Dense Layer)를 통해 이미지의 클래스를 분류



CNN이란?

Convolutional Neural Network(CNN, ConvNet)

잘 정리된 데이터와 달리 이미지에서는 연구자가 스스로 특징을 추출해야 한다.

시각적 이미지를 연구하는 분야에서는 이미지에서 특징을 추출하기 위한 다양한 알고리즘이 개발되었다.

그림 6.1

외곽선 검출 알고리즘 중 하나인 Canny Edge Detection을 이용한 결과 이미지¹




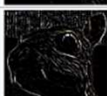



$$\frac{1}{159} \begin{bmatrix} 2 & 4 & 5 & 4 & 2 \\ 4 & 9 & 12 & 9 & 4 \\ 5 & 12 & 15 & 12 & 5 \\ 4 & 9 & 12 & 9 & 4 \\ 2 & 4 & 5 & 4 & 2 \end{bmatrix}$$

CNN이란?

Convolutional Neural Network(CNN, ConvNet)

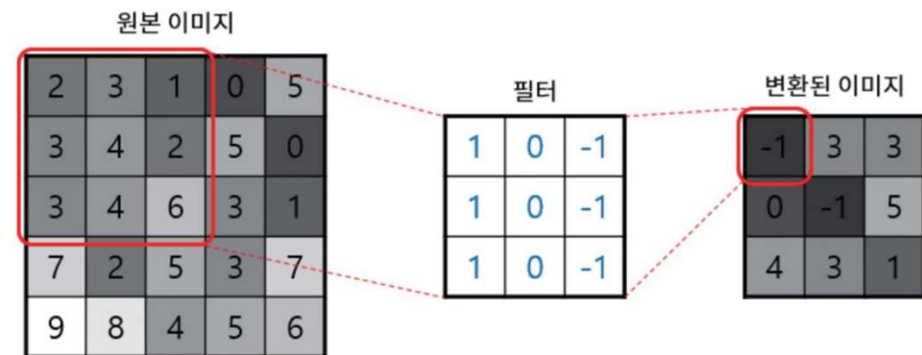
합성곱 신경망이라고도 불리는 이것은 필터링 기법을 인공 신경망에 적용하여 이미지를 더욱 더 효과적으로 처리하기 위해 **컨볼루션 네트워크** 논문(LeCun, 1989)에서 처음 소개 되었다.

Operation	Filter	Filtered image
Identity	$\begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}$	
Edge detection	$\begin{bmatrix} 1 & 0 & -1 \\ 0 & 0 & 0 \\ -1 & 0 & 1 \end{bmatrix}$	
	$\begin{bmatrix} 0 & 1 & 0 \\ 1 & -4 & 1 \\ 0 & 1 & 0 \end{bmatrix}$	
	$\begin{bmatrix} -1 & -1 & -1 \\ -1 & 8 & -1 \\ -1 & -1 & -1 \end{bmatrix}$	
Sharpen	$\begin{bmatrix} 0 & -1 & 0 \\ -1 & 5 & -1 \\ 0 & -1 & 0 \end{bmatrix}$	
Box blur (normalized)	$\frac{1}{9} \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix}$	

컨볼루션 연산은 원본 이미지와 필터 행렬의 합성곱

그림 6.2

컨볼루션 연산은 원본 이미지와 필터 행렬의 합성곱입니다.



$$2 \times 1 + 3 \times 0 + 1 \times (-1) + 3 \times 1 + 4 \times 0 + 2 \times (-1) + 3 \times 1 + 4 \times 0 + 6 \times (-1) = -1$$

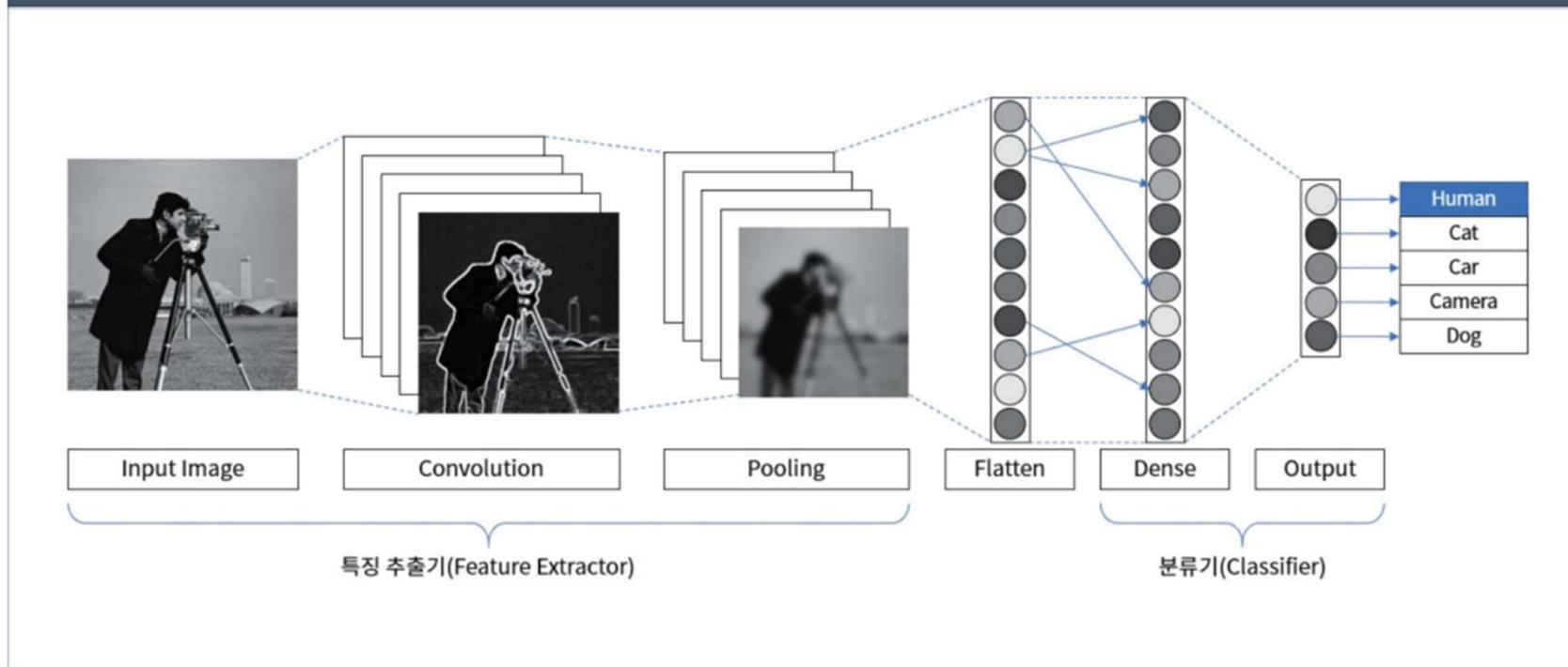
CNN Layer

CNN Layer

이미지 분류를 위한 컨볼루션 신경망은 **특징 추출기**(Feature Extractor)와 **분류기**(Classifier)가 합쳐진 상태

그림 6.5

이미지 분류에 사용되는 컨볼루션 신경망의 구조⁶



CNN Layer

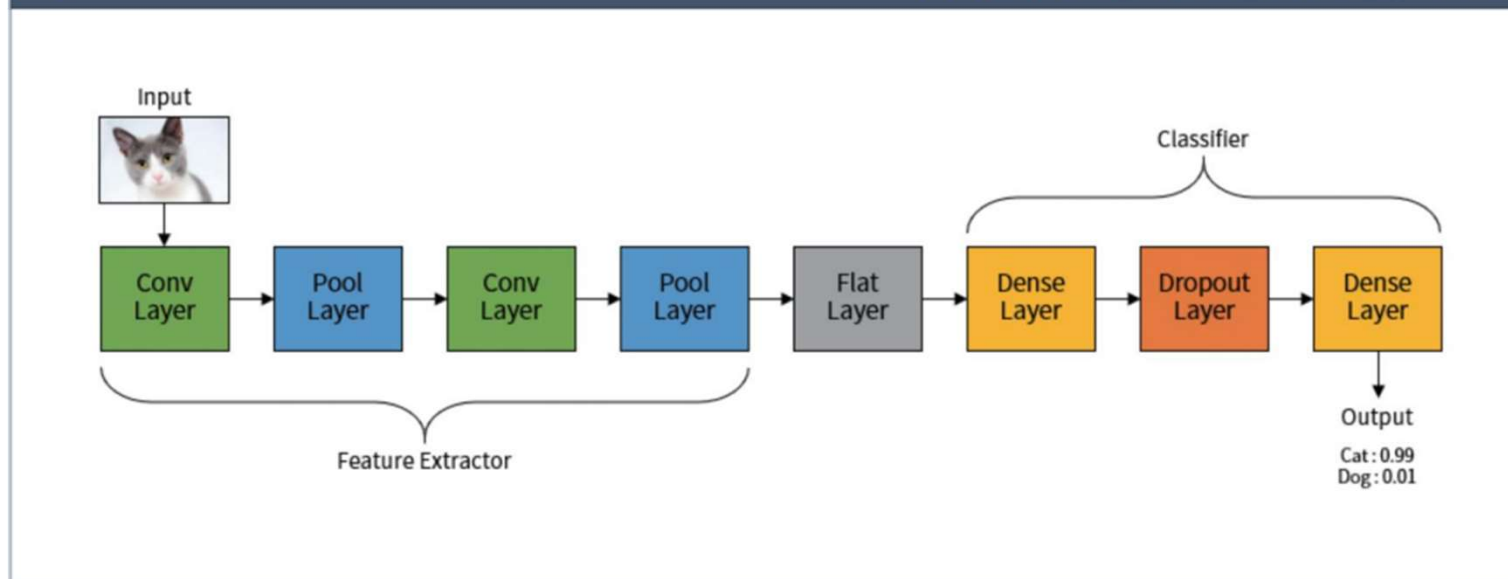
CNN Layer

특징 추출기(Feature Extractor)에는 컨볼루션 레이어와 풀링 레이어가 교차되며 배치된다.

분류기(Classifier)에는 Dense 레이어가 배치되고 과적합을 막기 위한 드롭아웃 레이어가 Dense 레이어 사이에 배치

그림 6.6

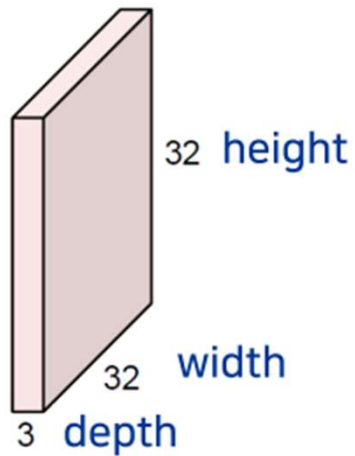
이미지 분류에 사용되는 컨볼루션 신경망의 구조



CNN Layer

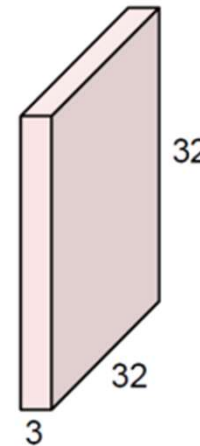
Input Layer

32 x 32 x 3 image



Convolution Layer

32 x 32 x 3 image



5 x 5 x 3 filter

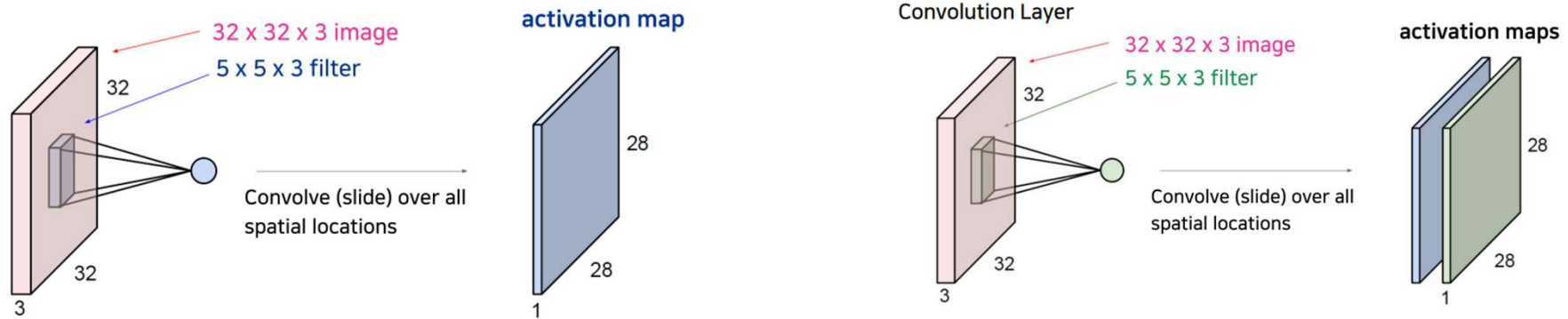


Filter의 depth는 항상 Input의 depth와 같아야 함.

이 층에서 학습한 파라미터 수는 $5 \times 5 \times 3 + 1$

CNN Layer

Convolution Layer



→ 계속 쌓다보면 이미지의 정보 손실. Padding 기법을 적용.

CNN Layer

Convolution Layer

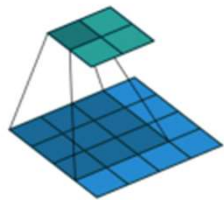
Padding

Activation map이 input 이미지와 동일한 크기를 유지하기 위해 사용하는 값

Conv Layer 수행 후 output 크기가 줄어든다. → 원본 input 데이터에 대한 손실 방지 위해 Padding 적용

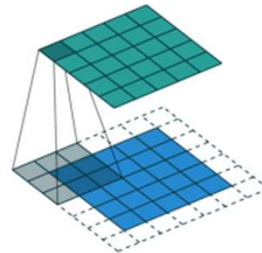
원본의 크기를 유지할 수 있도록 일정 숫자로 채움 → 일반적으로 zero padding

(filter size - 1)/2로 패딩을 한다 → Stride가 1일 경우



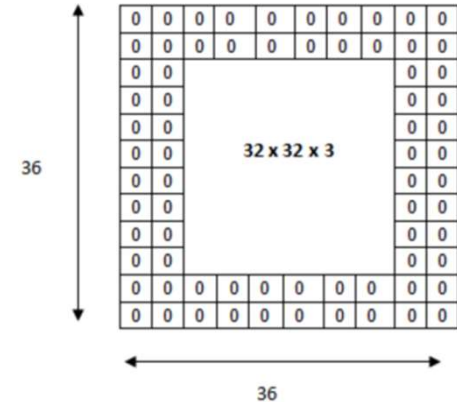
valid

- Input: 4×4
- Conv. Filter: 3×3
- Output: 2×2
- No zero-padding
- No strides



same

- Input: 5×5
- Conv. Filter: 3×3
- Output: 5×5
- Zero-padding: 1
- No strides



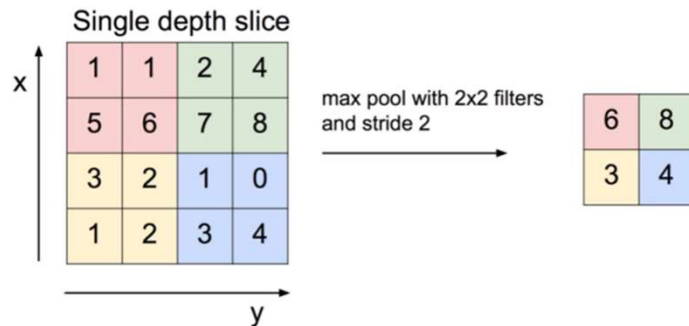
CNN Layer

Pooling

Resize 또는 Sampling을 의미 즉, 사이즈 변환을 의미함

→ 블록에서 최댓값(max pooling) 또는 평균값(average pooling) 방식

Pooling → 작은 변화에 민감하지 않게 됨, 파라미터를 줄이고 Overfitting 억제



일반적으로 CNN에서 Max Pooling 방식을 사용

Resize 되는 이미지의 크기는 filter와 stride와 관계가 있다.

2 x 2 filter에 stride 2를 적용하는 max pooling의 경우 2 x 2로 resize 됨

CNN Layer

Fully Connected Layer

CNN은 전처리 과정을 통해 이미지의 특징을 추출, **Fully connected** 과정을 통해 이미지를 분류

Conv. Layer나 pooling Layer의 경우

이전 layer의 작은 부분과 연결되어 연산을 수행. **공간적으로 떨어진 부분과 무관**

Fully connect layer의 경우

모든 node가 모든 output에 연결되어 있어야 함.

fully connect 레이어는 가장 나중에 연결하며, 레이어 단수의 제한은 x

→ 연산량이 많이 필요하므로 일반적으로 1~2 단의 layer만을 구성

Q & A