

# 딥러닝 입문 (CNN 입문)

# Histroy

Date	Ver	내용
2020.12.27	v01	CNN 입문

# 목차

- ▶ 01 합성곱 연산
- ▶ 02 풀링 연산(pooling)
- ▶ 03 데이터 증식(Data Augmentation)
- ▶ 04 사전 훈련된 네트워크
- ▶ 05 기타 사전 훈련 네트워크
- ▶ 06 사전 훈련된 네트워크를 사용하는 두가지 방법

# 01 합성곱 연산

- 완전 연결 층과 합성곱 층 사이의 차이는 다음과 같습니다.
  - (1) Dense 층은 입력 특성 공간에 있는 모든 픽셀에 대한 패턴학습
  - (2) 합성곱 층은 지역 패턴을 학습.

# 01 합성곱 연산

- 합성곱 연산은 입력이 3D 텐서에 적용됩니다.

(1) (150, 150, 3) => (높이, 너비, 깊이)

RGB 이미지일 경우, 3개의 컬러 채널을 가지므로 깊이는 3입니다.

- 합성곱 연산은 필터의 수만큼 특성 맵을 만들어냅니다.

# 01 합성곱 연산

- 합성곱 연산은 핵심적인 2개의 파라미터로 정의

- (1) 입력으로부터 하나의 합성곱 연산을 할 필터 :  $3 \times 3$ ,  $5 \times 5$

- (1) 합성곱으로 계산할 **필터의 수**

# 01 합성곱 연산 - 스트라이드

- 3D 입력 특성 맵 위를  $3 \times 3$  또는  $5 \times 5$  크기의 윈도우가 슬라이딩(sliding)하면서
- 해당 위치에서 **이미지의 영역의 값과 필터의 가중치의 합성곱을 통한 하나의 값을** 출력합니다.
- 이것들이 반복적으로 이루어져 하나의 필터가 하나의 특성맵을 만들어냅니다.
- 이때 윈도우가 슬라이딩하는 이동의 크기를 **스트라이드**라고 합니다.

스트라이드는 기본적으로 1입니다. 만약 2를 사용하면 특성 맵의 높이와 너비가 2의 배수로 다운샘플링 된 것을 의미합니다.

# 01 합성곱 연산 - 패딩

...

- 이때 윈도우가 슬라이딩하는 이동의 크기를 스트라이드(stride)라고 합니다.
- 이에 대한 반복으로 생성되는 특성맵의 이미지의 크기를 그대로 유지하려면 우리는 **패딩**이라는 것을 이용하여 이미지의 크기를 유지할 수 있습니다.
- Keras의 Conv2D층에서 패딩은 **padding 매개변수**로 설정이 가능합니다.  
**valid** : 패딩 사용 안함. **same** : 입력과 동일한 높이와 너비를 가진 출력.



## 02 풀링(Pooling) 연산

입력 특성 맵에서 윈도우에서 조건에 맞는 하나의 값을 추출한다.

### (1) 최대 풀링 연산(Max Pooling)

해당 윈도우에서 가장 최대의 값을 추출한다. 2x2라면 해당 영역안에서 가장 높은 값을 취한다.

### (2) 평균 풀링 연산(average Pooling)

해당 윈도우에서 평균값을 추출한다.

## 03 Data Augmentaion(데이터 증식)

- (1) 컴퓨터 비전에서 과대 적합을 줄이기 위한 강력한 방법.
- (2) 데이터 증식은 기존 훈련 샘플로부터 더 많은 훈련 데이터를 생성하는 방법입니다.
- (3) 그럴듯한 이미지를 생성하도록 여러가지 랜덤 변환(각도, 좌우변환 등)을 적용하여 샘플을 늘린다.
- (4) 케라스에서는 ImageDataGenerator 클래스를 사용.

## 04 사전 훈련된 네트워크

- (1) 작은 이미지 데이터 셋에서 딥러닝을 적용하는 일반적이고 매우 효과적인 방법.
- (2) 사전 훈련된 네트워크(pretrained network)는 일반적으로 대규모 이미지 분류 문제를 위해 대량의 데이터셋에서 미리 훈련되어 저장된 네트워크
- (3) 1400만개의 레이블된 이미지와 1000개의 클래스로 이루어진 데이터 셋(ImageNet)

## 04 사전 훈련된 네트워크

- (1) 작은 이미지 데이터 셋에서 딥러닝을 적용하는 일반적이고 매우 효과적인 방법.
- (2) 사전 훈련된 네트워크(pretrained network)는 일반적으로 대규모 이미지 분류 문제를 위해 대량의 데이터셋에서 미리 훈련되어 저장된 네트워크
- (3) 1400만개의 레이블된 이미지와 1000개의 클래스로 이루어진 데이터 셋(ImageNet)

## 05 기타 사전 훈련 네트워크

(1) VGG

(2) ResNet

(3) Inception-ResNet

(4) Xception

\* Keras에서는 `keras.applications` 모듈에서 임포트가 가능.

## 06 사전 훈련된 네트워크를 사용하는 두가지 방법

### (1) 특성 추출(feature extraction)

사전에 학습된 네트워크의 표현을 사용하여 새로운 샘플에서 흥미로운 특성을 뽑아낸다.

즉, 합성곱 기반층은 사전 훈련 네트워크 가중치를 이용하고 분류기 부분을 훈련 시킨다.

### (2) 미세 조정(fine tuning)

미세 조정은 특성 추출에 사용했던 동결 모델의 상위 층 몇 개를 동결에서 해제하고 모델에 새로 추가한 층과 함께 훈련하는 것.