



D&A

Deep Session 4차시

# CNN 기초.

2022 / 03 / 31  
D&A 운영진 이예진



2022 빅데이터 분석 학회 D&A

# CONTENTS.

## 01 CNN

- # CNN
- # 매개변수 / 인자
- # CNN vs MLP

## 02 CNN구조

- # Convolution Layer
- # Pooling Layer
- # Fully Connected Layer

## 03 Data Augmentation

내용을 입력하세요.

## 04 실습

# CIFAR-10 이용



# 01. CNN

## 컴퓨터 비전 task

### object detection 객체검출

입력 이미지에 존재하는 객체를 검출해내는 것

검출 목적 별 분석 방법

Classification 분류 + localization 지역화 => object detection 객체검출

### object recognition 객체 인식

객체의 존재 유무 뿐만 아니라 어떤 것인지를 구분해내는 것

### object segmentation 객체 분할

입력 이미지(pixel)에서 객체가 존재하는 영역을 분할해내는 것을 의미



# 01. CNN

## ImageNet

이미지 분류 모델을 측정하기 위한 데이터

가장 많이 사용하는 데이터셋

학습 데이터만 총 138GB, 총 2만개 이상의 클래스 & 약 1,400만장의 이미지로 구성

## ILSVRC

ImageNet Large Scale Visual Recognition Challenge

2015년을 기점으로 인간의 성능을 뛰어넘음



# 01. CNN

## Region Feature (지역정보)

$x_{11}$	$x_{12}$	$x_{13}$
$x_{21}$	$x_{22}$	$x_{23}$



$x_{11}$	$x_{11}$	$x_{13}$	...	$x_{21}$	$x_{22}$	$x_{23}$
----------	----------	----------	-----	----------	----------	----------

가까운 위치라는 정보 반영 X

$x_{11}$	$x_{12}$	$x_{13}$
$x_{21}$	$x_{22}$	$x_{23}$



지역 정보를 학습할 수 있는 신경망 구조 : CNN

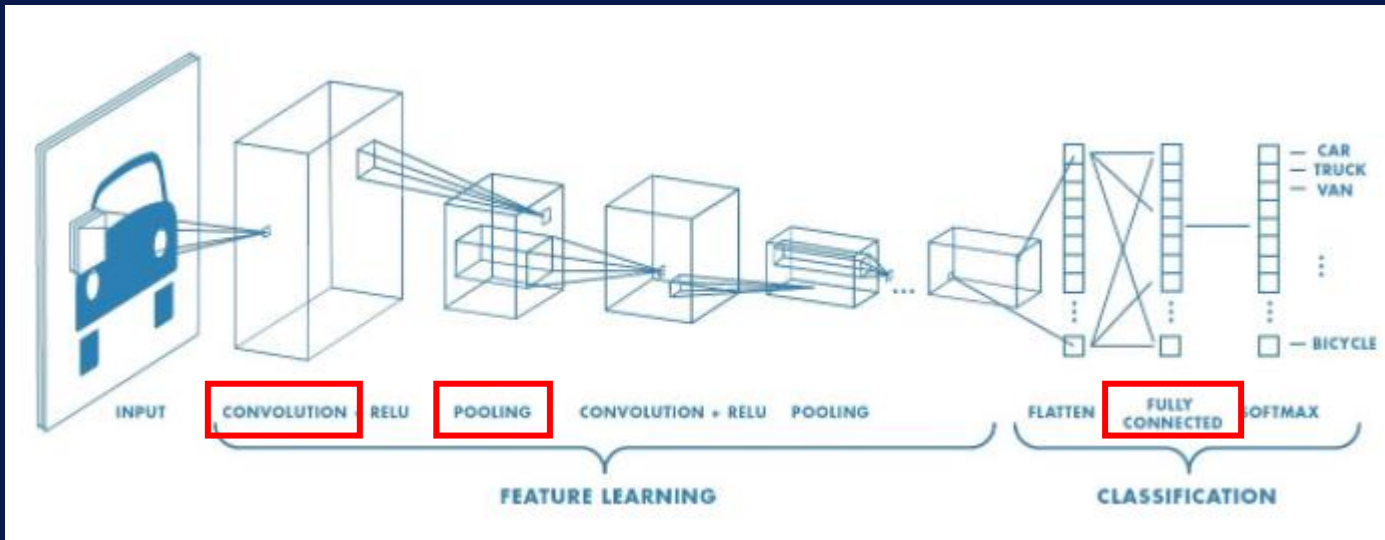


# 01. CNN

## CNN이란?

Convolutional Neural Network의 약자

- 지역 정보(Region Feature, Graphical Feature)를 학습시키기 위한 신경망 모형
- Yann LeCun 교수가 1998년에 제안 (손글씨를 분류하는 CNN모형)
- 당시 학습시간이 너무 오래걸려 주목받지 못함
- 2010년대 알고리즘 & 하드웨어의 발전으로 이미지 처리하는 신경망 모형으로 급부상



기본적인 CNN 구조

**Convolution Layer**

: 지역 정보를 뽑아내기 위해

**Pooling Layer**

: Feature의 차원을 줄이기 위해

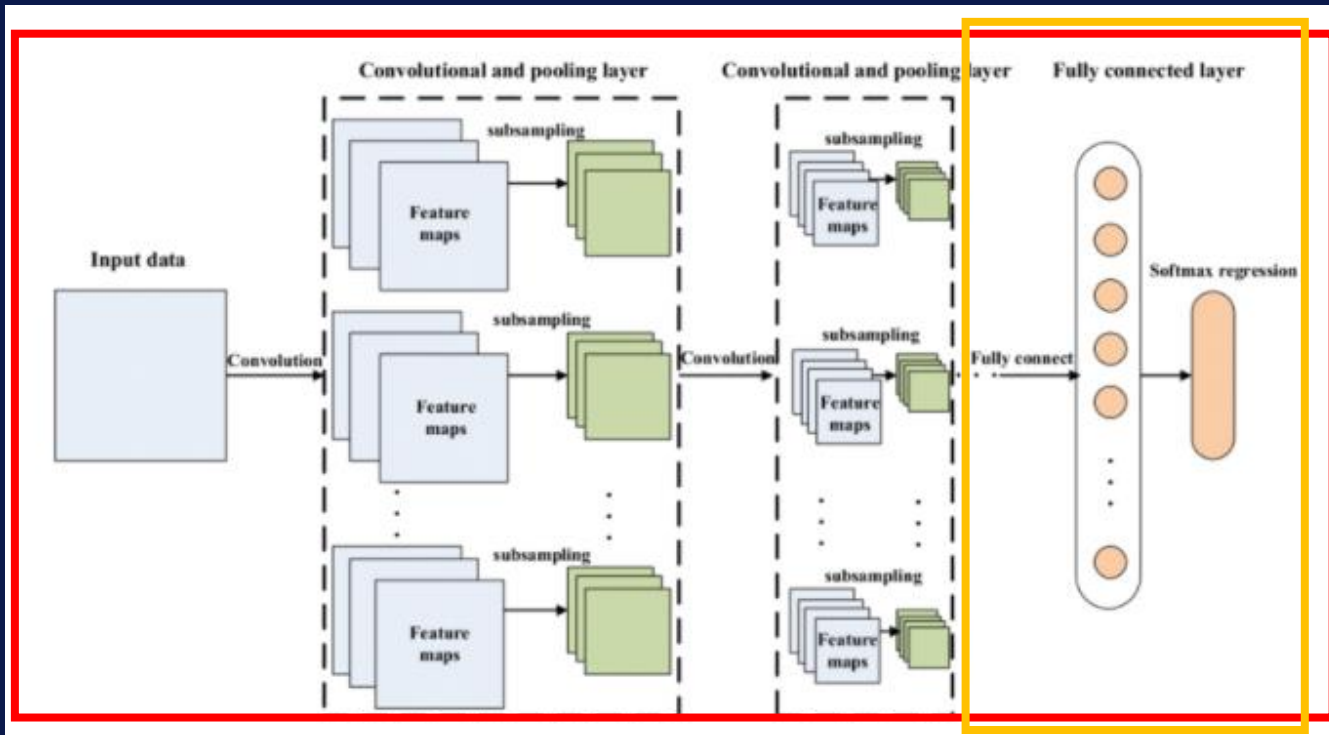
**Fully Connected Layer**

: 최종적인 분류를 위해  
(일반적인 MLP구조를 지님)

# 01. CNN

## CNN vs MLP

가장 큰 차이점 : 이미지의 Feature를 어떻게 추출하는 가



MLP

: 이미지의 픽셀 값을 바로 Network의 Input으로 사용

CNN

: 이미지의 Region Feature를 Convolution Layer와 Pooling Layer를 이용해 추출 -> 이 Feature를 MLP의 Input으로 사용

CNN이 이미지 처리와 관련된 Vision분야에서 성능이 좋은 이유 : Region Feature 추출

## 02. CNN 구조

### Convolution Layer

Receptive Field를 정의해 입력층의 이미지의 Feature를 추출하는 역할  
Filter(kernel)을 일정간격(stride)만큼 이동시키며 이미지와 겹쳐지는 부분과 연산하여 출력  
이미지 픽셀값과 Receptive Field의 Weight가 선형결합하여 값이 나옴  
(Weight는 고정된 값이 아닌 신경망 모형을 이용해 학습하면서 바뀌는 값)  
추출한 Feature를 Feature Map이라 부름

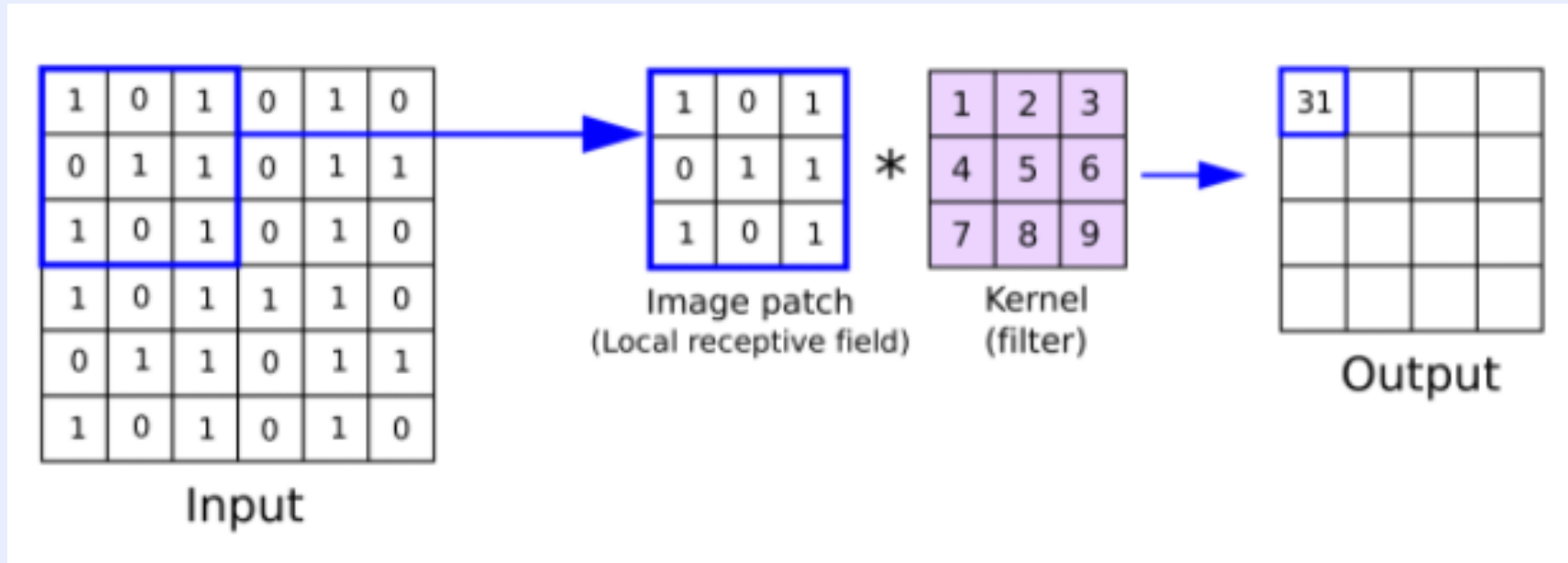
### Weight Sharing

필터를 이동하면서 연산을 하게 되면 겹치는 부분이 발생  
필터 값들을 공유한다고 해서 CNN은 Weight Sharing 특성 존재  
weight sharing을 할 경우 파라미터 수를 최대한 줄일 수 있음  
(Receptive Field를 움직여 Feature Map을 추출할 때마다 다른 weight를 사용할 경우 많은 파라미터 학습이 필요)



## 02. CNN 구조

### Convolution Layer



## 02. CNN 구조

### Channel

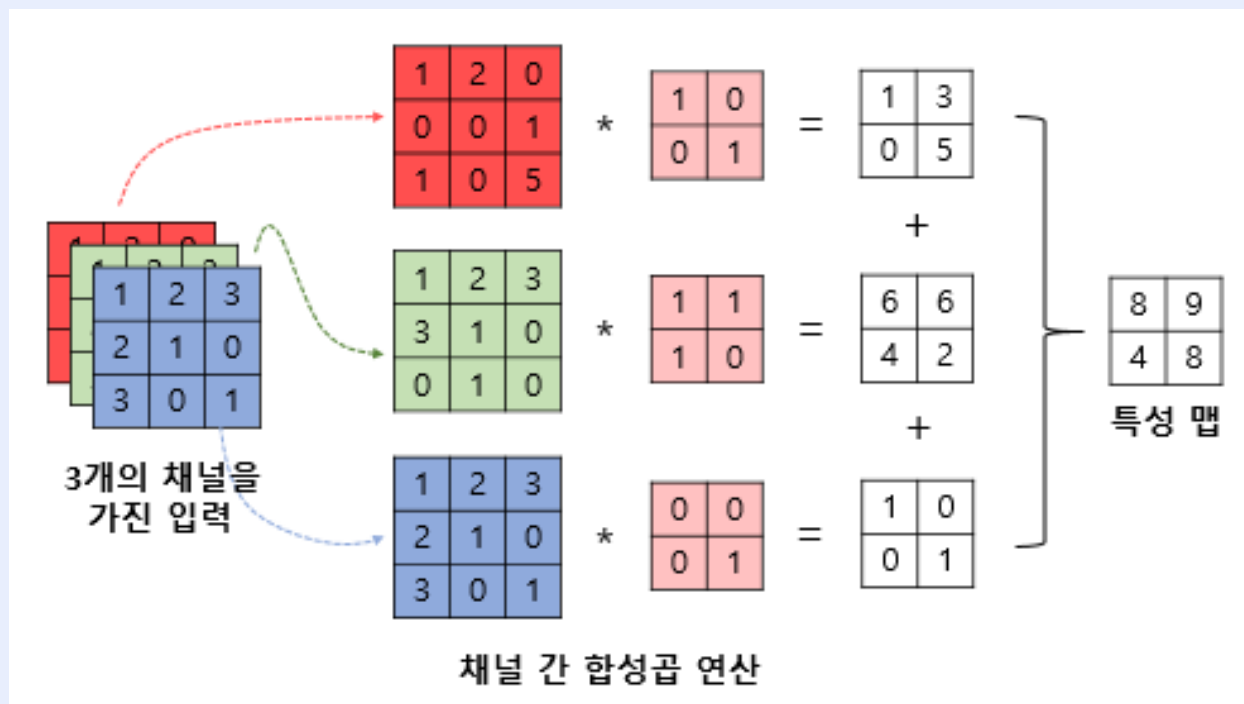
이미지 픽셀 하나하나 실수  
컬러 사진은 각 픽셀을 RGB 3개의 실수로 표현한 3차원 데이터  
따라서 이미지는 3개의 채널로 구성

### Filter (=Kernel)

이미지의 특징을 찾아내기 위한 공용 파라미터  
일반적으로 (4,4) or (3,3)과 같은 정사각 행렬로 정의  
CNN에서 학습의 대상은 필터 파라미터  
필터가 입력 데이터를 지정된 간격으로 순회하며 채널별로 합성곱 진행, 모든 채널의 합성곱의 합 Feature Map을 만듦

# 02. CNN 구조

## Convolution Layer



# 02. CNN 구조



## Filter

Receptive Field가 이미지를 돌면서 Feature를 뽑을 때 이동하는 칸 수 의미

목적 : Weight sharing을 하게 될 경우 중복된 값이 계속 나오는 것을 방지하기 위해

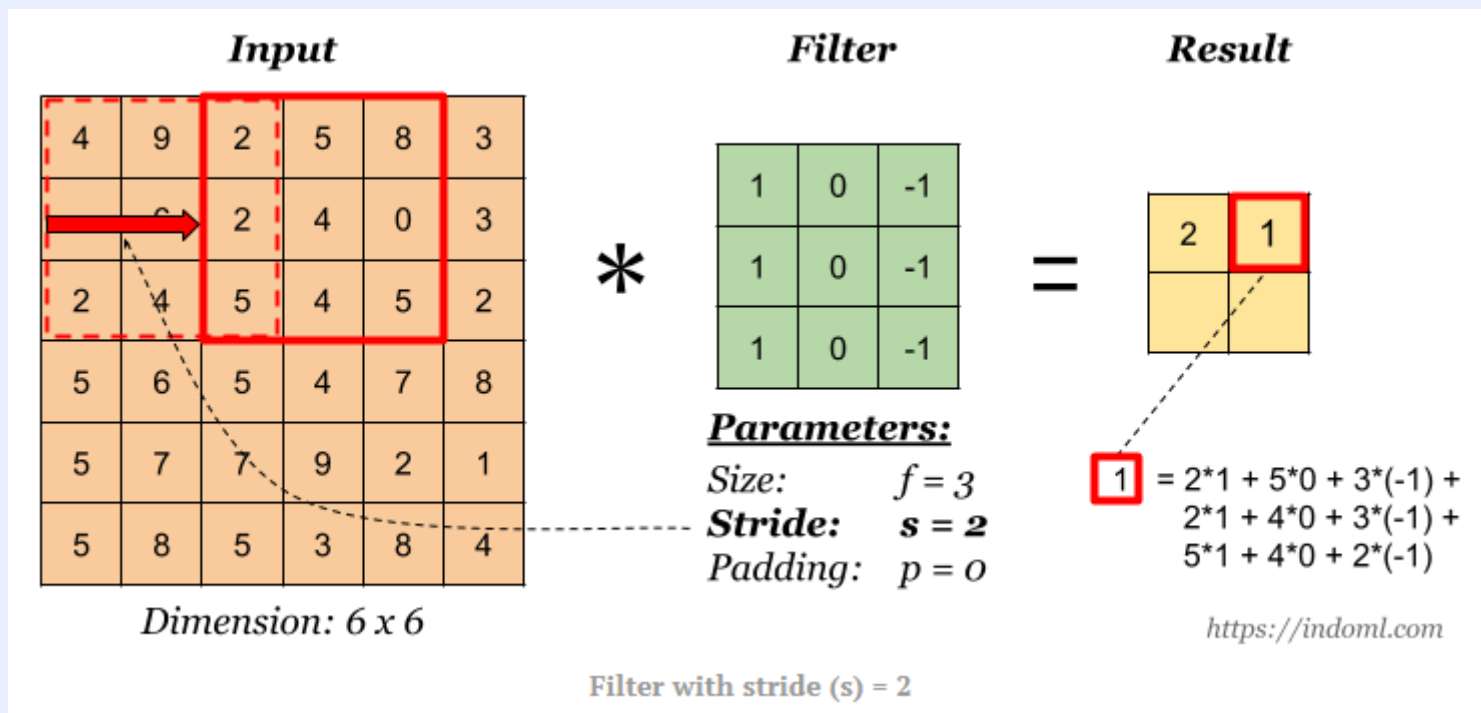


# 02. CNN 구조

## Stride

Receptive Field가 이미지를 돌면서 Feature를 뽑을 때 이동하는 칸 수 의미

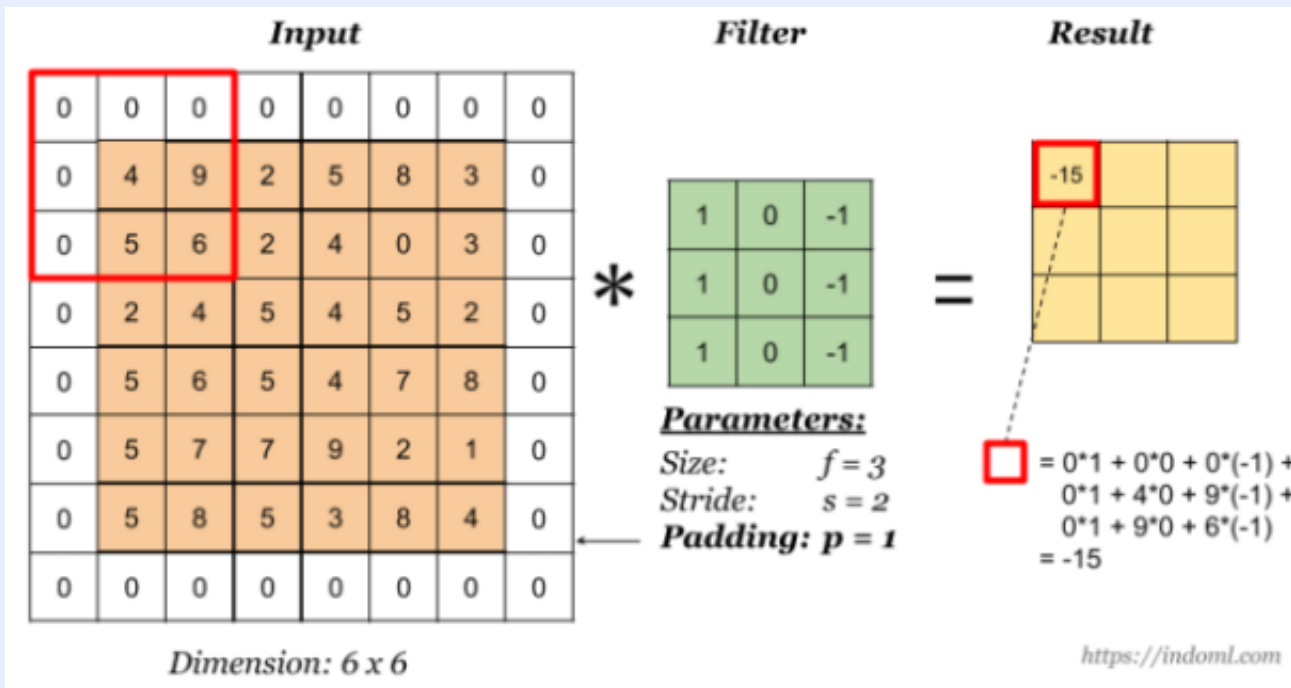
목적 : Weight sharing을 하게 될 경우 중복된 값이 계속 나오는 것을 방지하기 위해



# 02. CNN 구조

## Padding

일반적으로 Convolution을 적용하면 Feature(Image)의 size 감소  
가장자리 픽셀 값은 안쪽 픽셀값보다 적게 convolution됨  
역할 : 입력 데이터의 사이즈를 늘려줌



## Full Padding

모든 요소들이 같은 비율로 연산에 참여하도록

## Same Padding

output크기를 input크기와 동일하게

## Valid Padding

패딩을 안 하는 것

## Zero Padding

패딩에 들어가는 값을 0으로

패딩에 들어가는 값에 다른 임의의 값을 넣기도 가능

## 02. CNN 구조

### Output 크기 계산

$$output\ size = \frac{input\ size - filter\ size + (2 \times padding)}{stride} + 1$$

#### 예제 1

input size = 227 X 227

filter size = 11 X 11

stride = 4

padding = 0

#### 예제 2

input size = 32 X 32

filter size = 5 X 5

stride = 1

padding = 2

#### 예제 3

input size = 64 X 32

filter size = 3 X 3

stride = 1

padding = 1

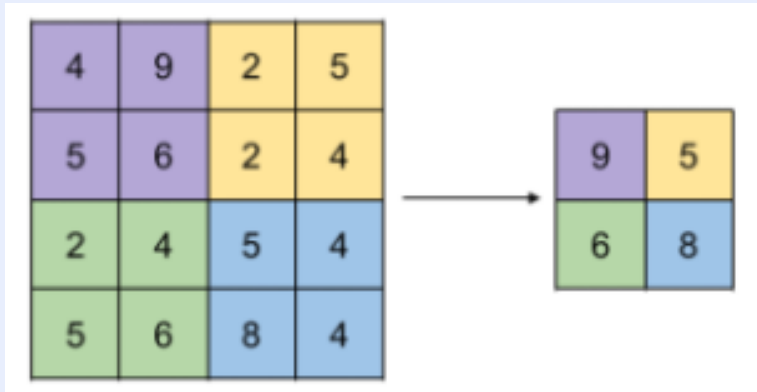


# 02. CNN 구조

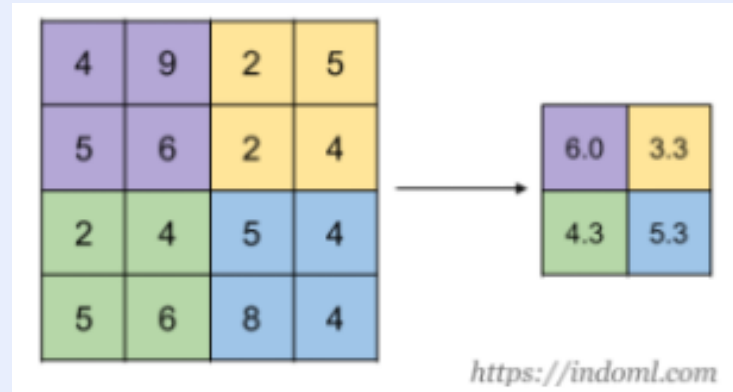
## Pooling Layer

CNN의 학습 속도를 향상시키기 위해 Feature의 Dimension을 줄이는 것  
일반적으로 Convolution을 거치고 Pooling Layer를 거치게 됨

Max Pooling



Average Pooling



Cross Channel Pooling





# 02. CNN 구조

## Fully Connected Layer

일반적인 MLP 구조와 동일

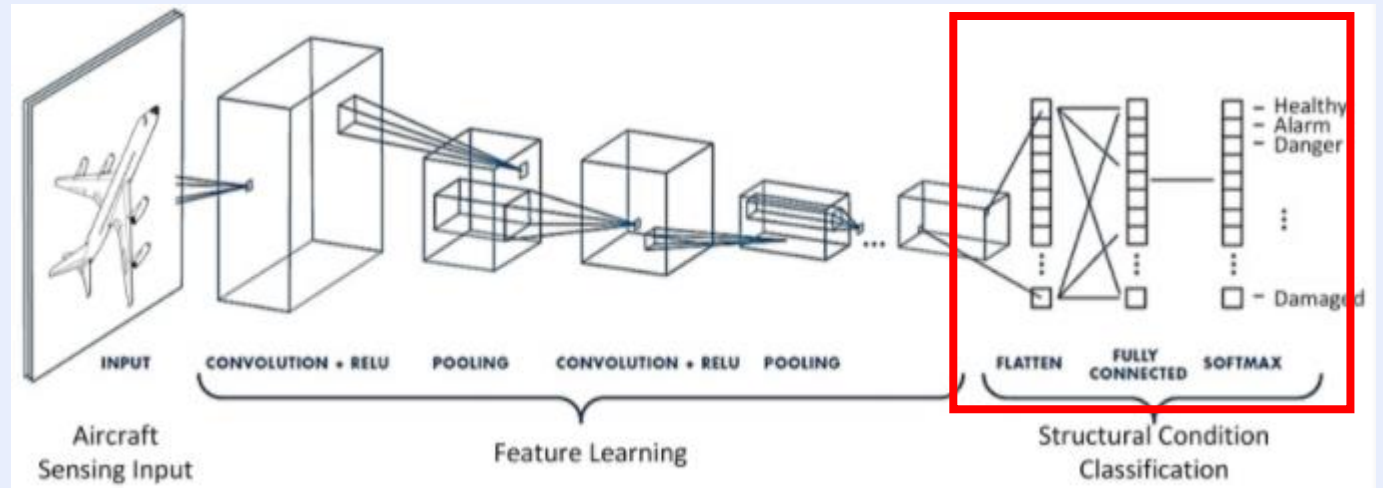
Pooling Layer에서 나온 Feature를 Flatten시켜 MLP의 Input으로 놓고 학습 진행

목적 : (단순 분류 문제경우) Convolution/Pooling에서의 결과를 취해 이미지를 정의된 라벨로 분류하는데 사용하는 것

한층의 모든 뉴런이 다음층의 모든 뉴런과 연결된 상태

2차원의 배열 형태 이미지를 1차원으로 Flatten시켜 이미지를 분류하는데 사용되는 계층

- 2차원 배열 형태의 이미지를 1차원 배열로 Flatten
- 활성화함수(Relu, Tanh등) 뉴런 활성화
- 분류기(softmax) 함수로 분류하는 과정



# 03. Data Augmentation

## Data Augmentation이란?

데이터를 임의로 변형해 데이터의 수를 늘려 다양한 Feature를 뽑는 방법

Random Flip : 이미지를 랜덤으로 좌우/상하 반전

Rotation : 이미지 회전

Crop : 이미지의 일정 부분 자르기

Scaling : 이미지 확대 or 축소



원본



Random Flip



Rotation



Crop



Scaling

# 03. Data Augmentation

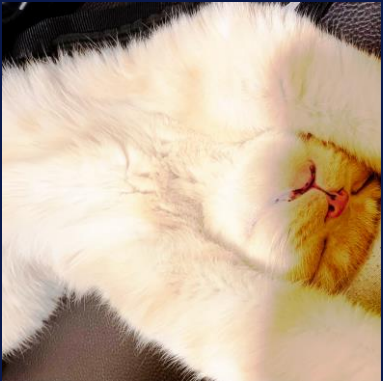
Color Shifting : RGB를 이용한 색변환

Cutout : 이미지의 일부를 검은색 사각형 모양으로 칠하는 방법 (0을 채움. Dropout효과)

Cutmix : 여러 이미지를 합쳐 놓고, 이미지의 label을 각 이미지가 차지하는 비율로 설정

Mixup : 두 데이터와 라벨을 일정 비율로 섞어 생성

Color Shifting



Cutout



Cutmix



[1.0, 0.0]  
고양이 강아지



[0.0, 1.0]  
고양이 강아지

Mixup



[0.7, 0.3]  
고양이 강아지



# 04. 실습

---

CIFAR-10 이용



# 참부자료 출처

## 01. CNN

CNN 이미지 출처

: <https://towardsdatascience.com/a-comprehensive-guide-to-convolutional-neural-networks-the-eli5-way-3bd2b1164a53>

CNN vs MLP 이미지 출처

: <https://medium.com/analytics-vidhya/cnn-convolutional-neural-network-8d0a292b4498>

## 02. CNN 구조

Convolution Layer 이미지 출처

: <https://anhreynolds.com/blogs/cnn.html>

Convolution Layer 이미지 출처

: <https://wikidocs.net/62306>

Stride 이미지 출처

: <https://indoml.com/2018/03/07/student-notes-convolutional-neural-networks-cnn-introduction/>

Padding 이미지 출처

: <https://indoml.com/2018/03/07/student-notes-convolutional-neural-networks-cnn-introduction/>

Pooling Layer 이미지 출처

: <https://indoml.com/2018/03/07/student-notes-convolutional-neural-networks-cnn-introduction/>

Pooling Layer 이미지 출처

: <https://light-tree.tistory.com/147>

Fully Connected Layer 이미지 출처

: <https://www.mdpi.com/1424-8220/19/22/4933/htm>





D&A

Deep Session 4차시 CNN 기초

Thank You.

2022 / 03 / 31  
D&A 운영진 이예진



2022 빅데이터 분석 학회 D&A