

The background features a close-up of a woman's face, looking slightly to the right. Overlaid on this are various digital and technological elements: a grid of binary code (0s and 1s) in white and blue, a collage of small images showing cityscapes, a globe, a rocket, and other tech-related scenes, and a faint, glowing blue grid pattern.

딥러닝

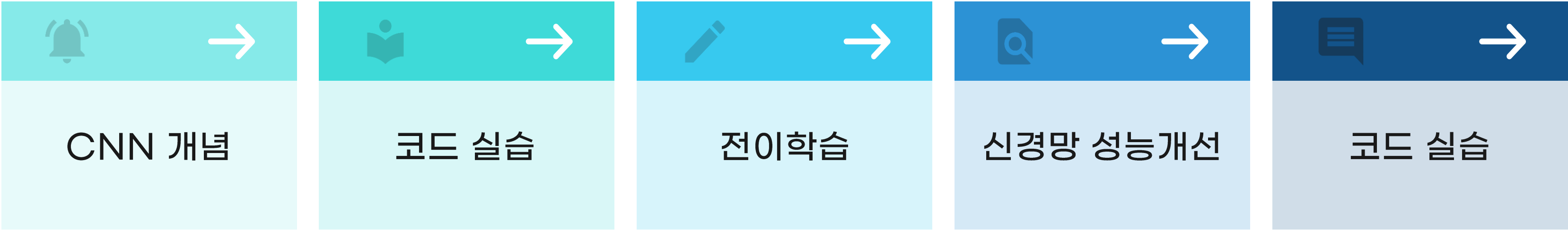
(Deep Learning)

최성우

학습목표

- 합성곱신경망(CNN)에 대해 알 수 있다.
- keras를 활용해 CNN을 구현할 수 있다.
- 신경망 성능 개선을 위한 방법들을 알 수 있다.





합성곱신경망 (CNN)

데이터의 특징들을 추출 해내는 신경망!

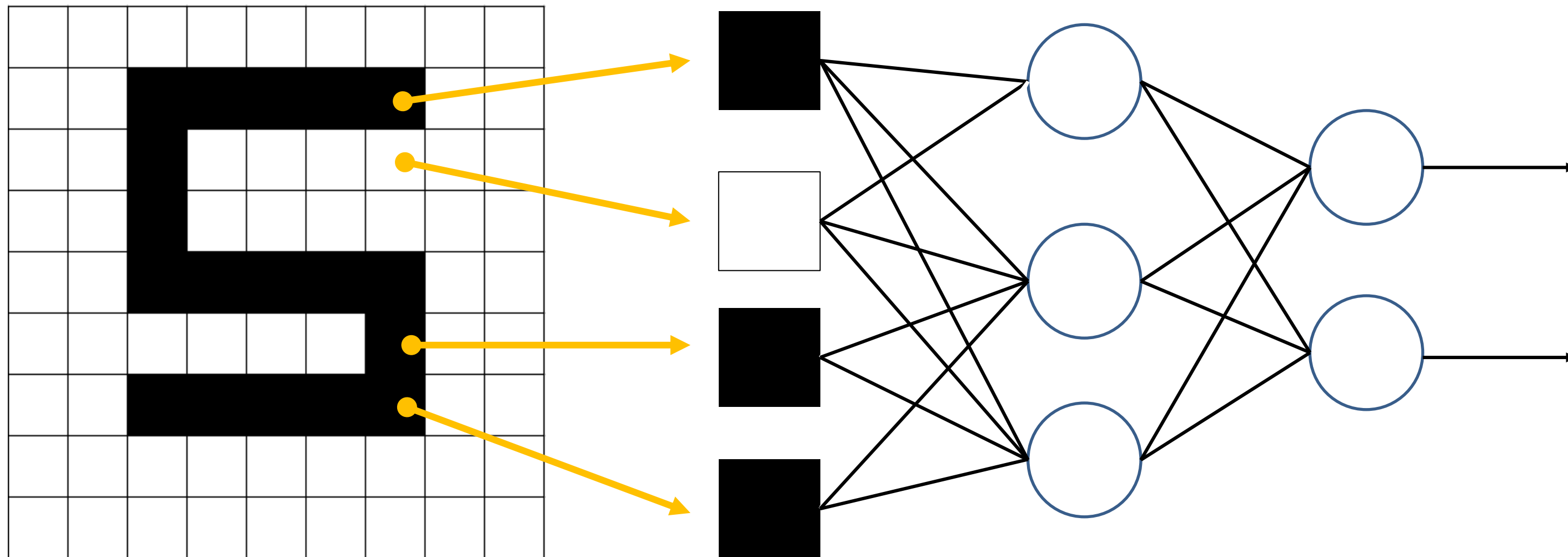


MLP의 이미지 분석

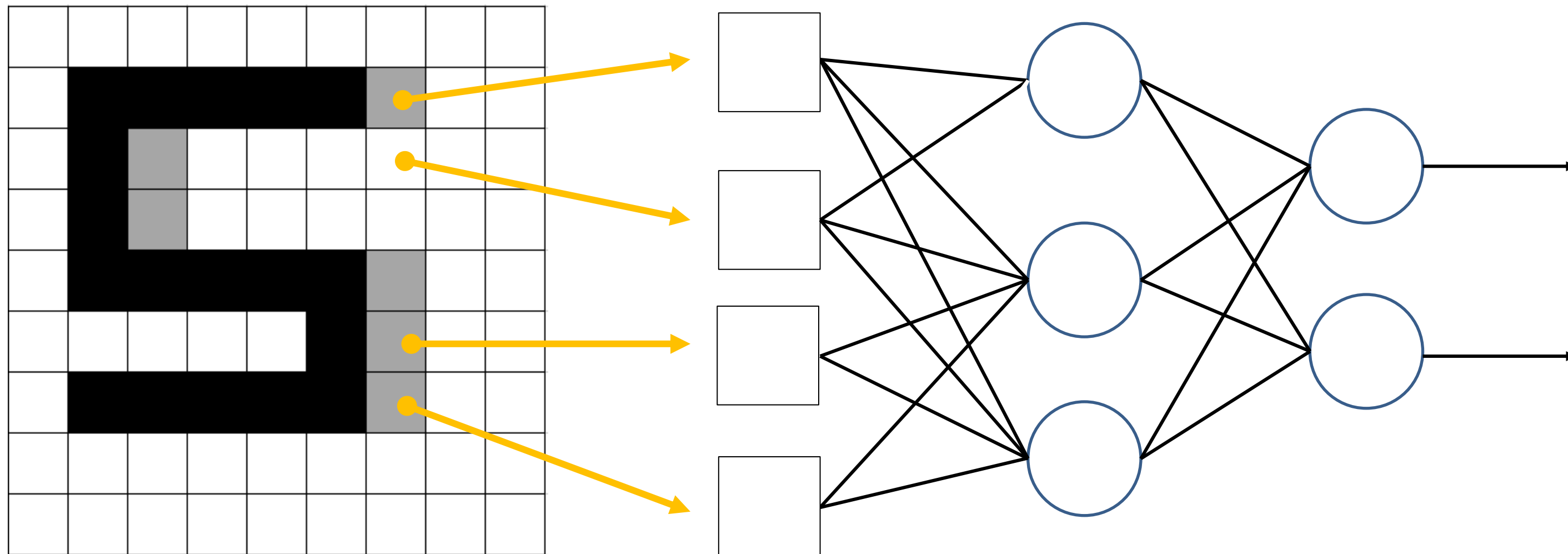
- MLP 신경망을 이미지 처리에 사용한다면 **이미지의 위치에 민감**하게 동작하며 위치에 종속적인 결과를 얻게 됨(모든 픽셀을 연산하기 때문)
- MLP는 아래 세 개의 7은 패턴이 다르다고 판단
- MLP로 이러한 숫자 인식을 하려면 숫자의 크기와 위치를 비슷하게 맞춰야 함



CNN(Convolutional Neural Network)

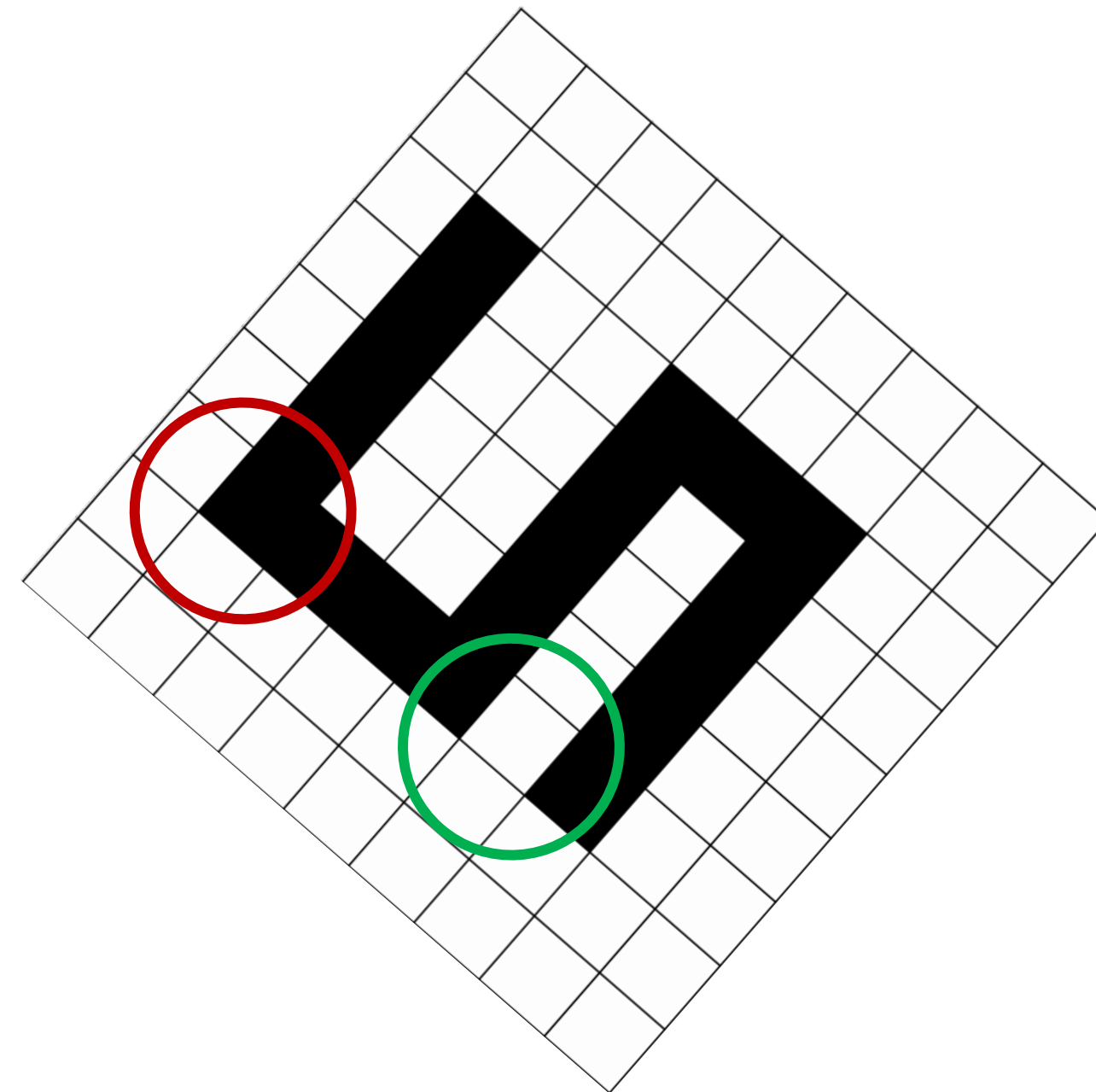
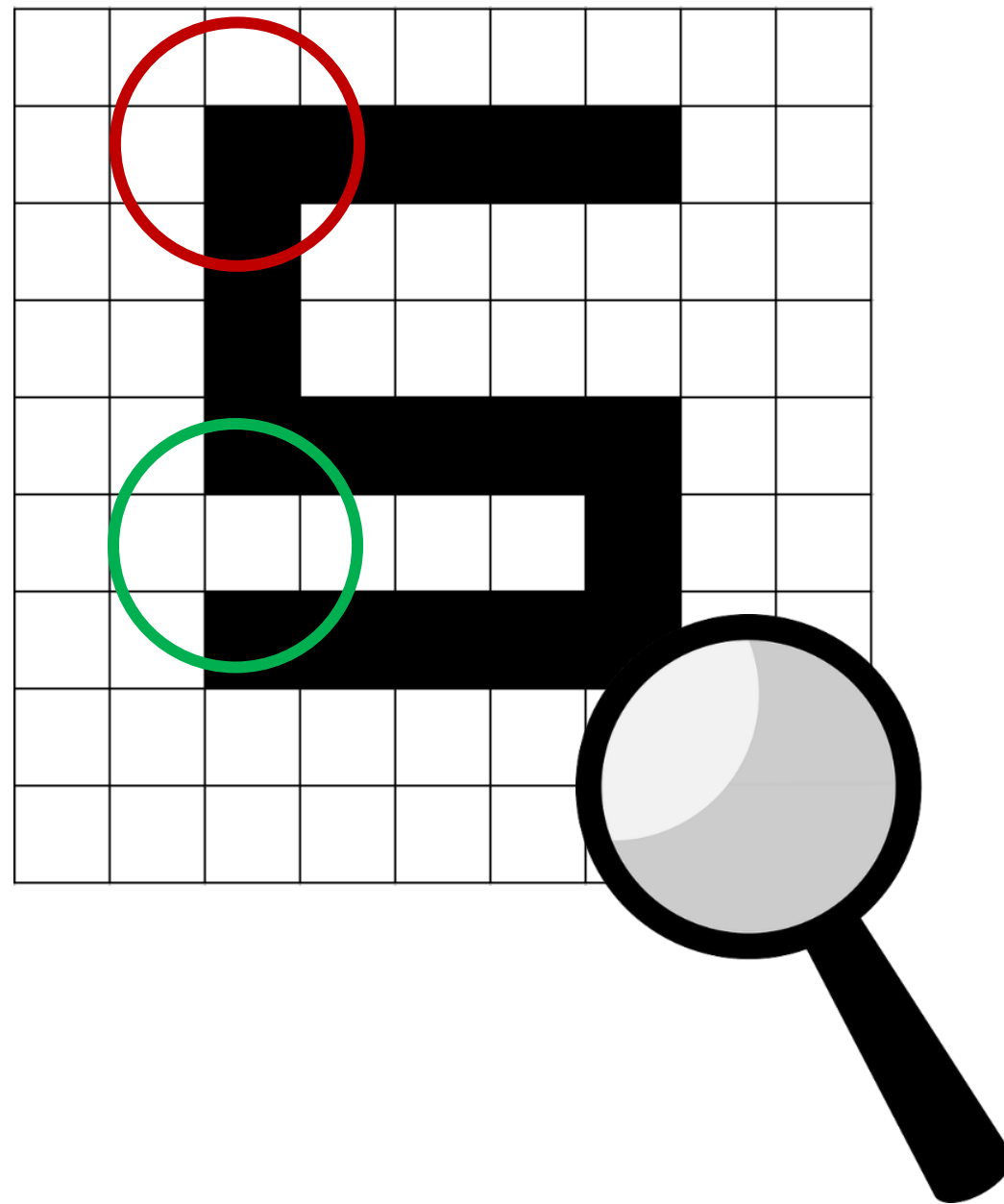


CNN(Convolutional Neural Network)



CNN(Convolutional Neural Network)

특징을 추출해서 비교해보자!

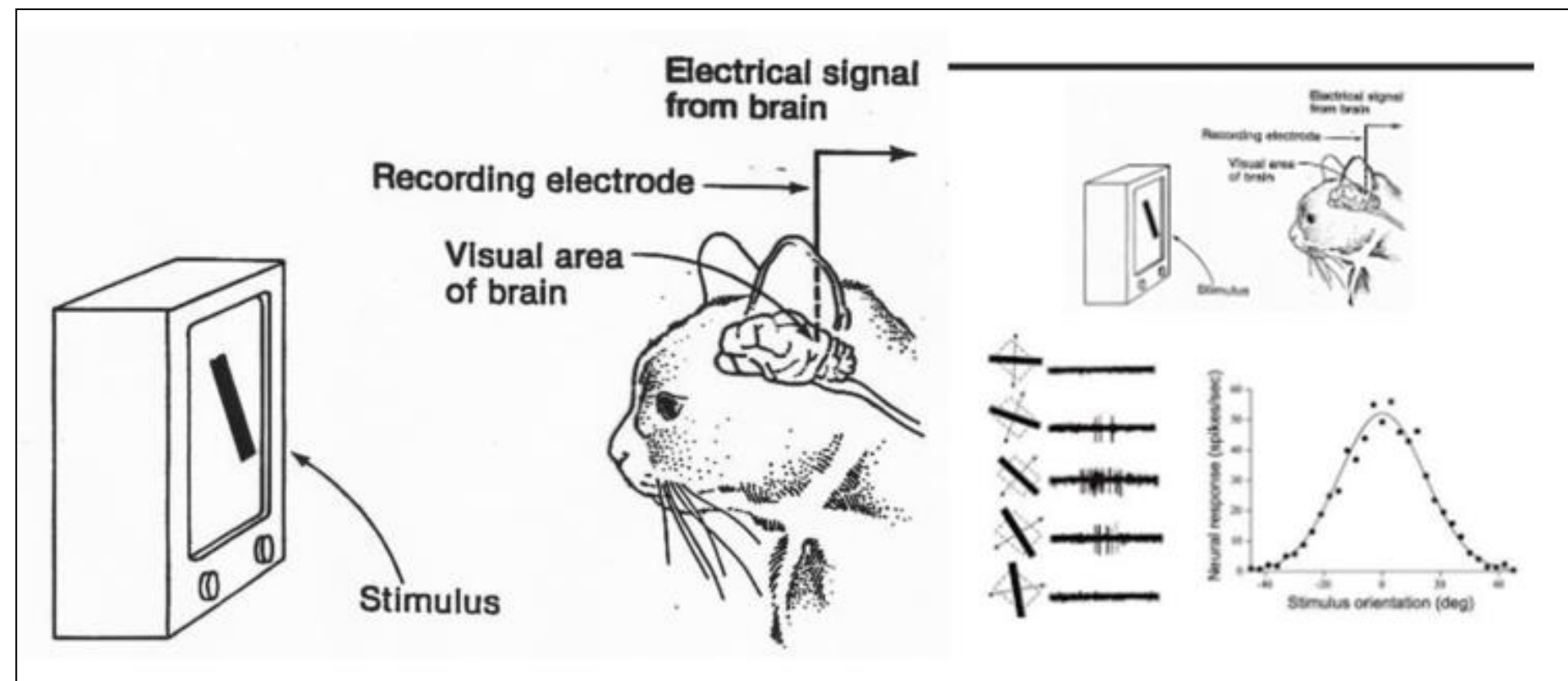


CNN(Convolutional Neural Network)



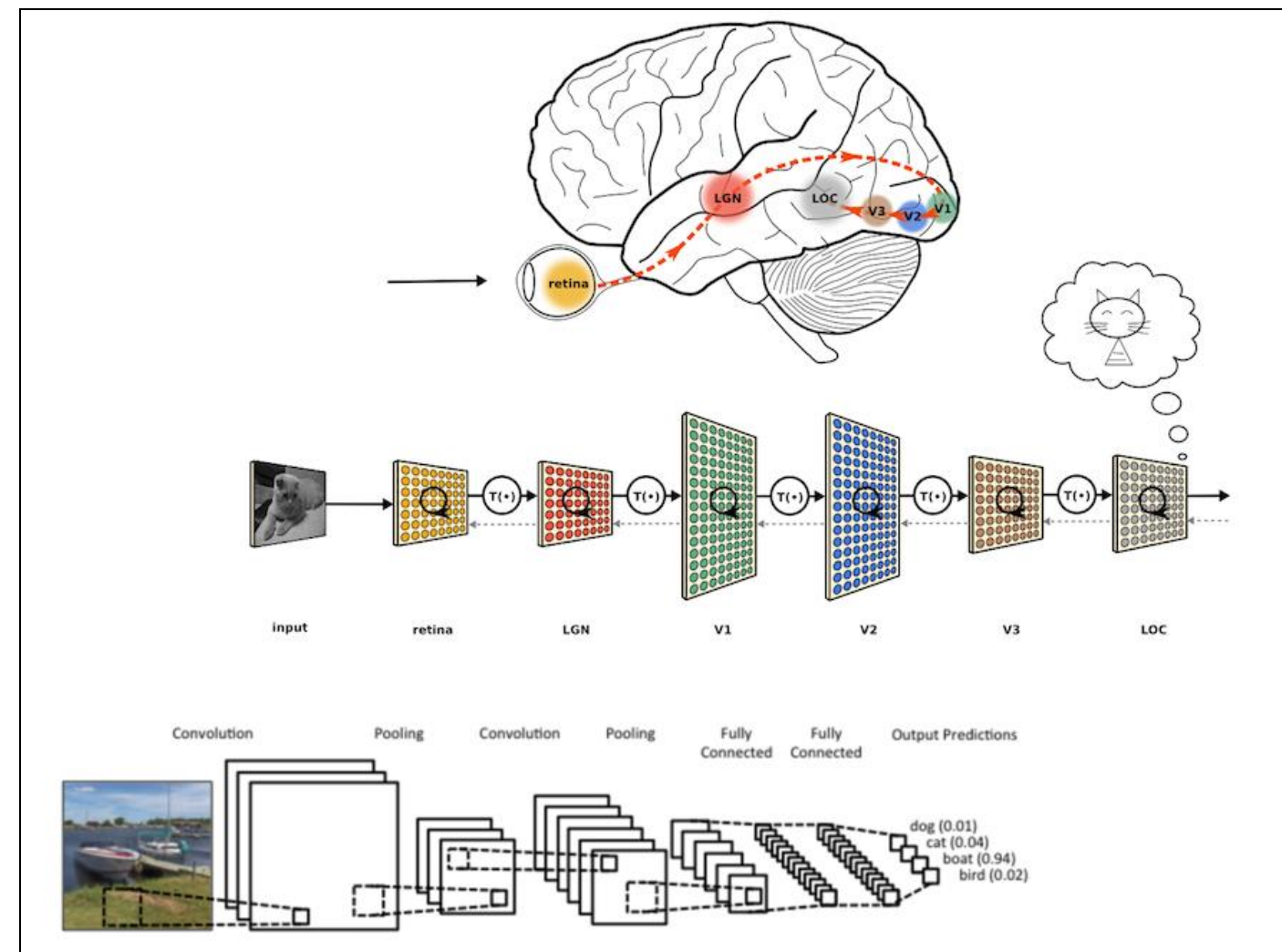
CNN 제안 배경

- 1950년대 수행된 고양이 뇌파 실험에 영감을 얻은 Yann Lecun 교수에 의해 1998년 이미지 인식을 획기적으로 개선할 수 있는 CNN이 제안됨



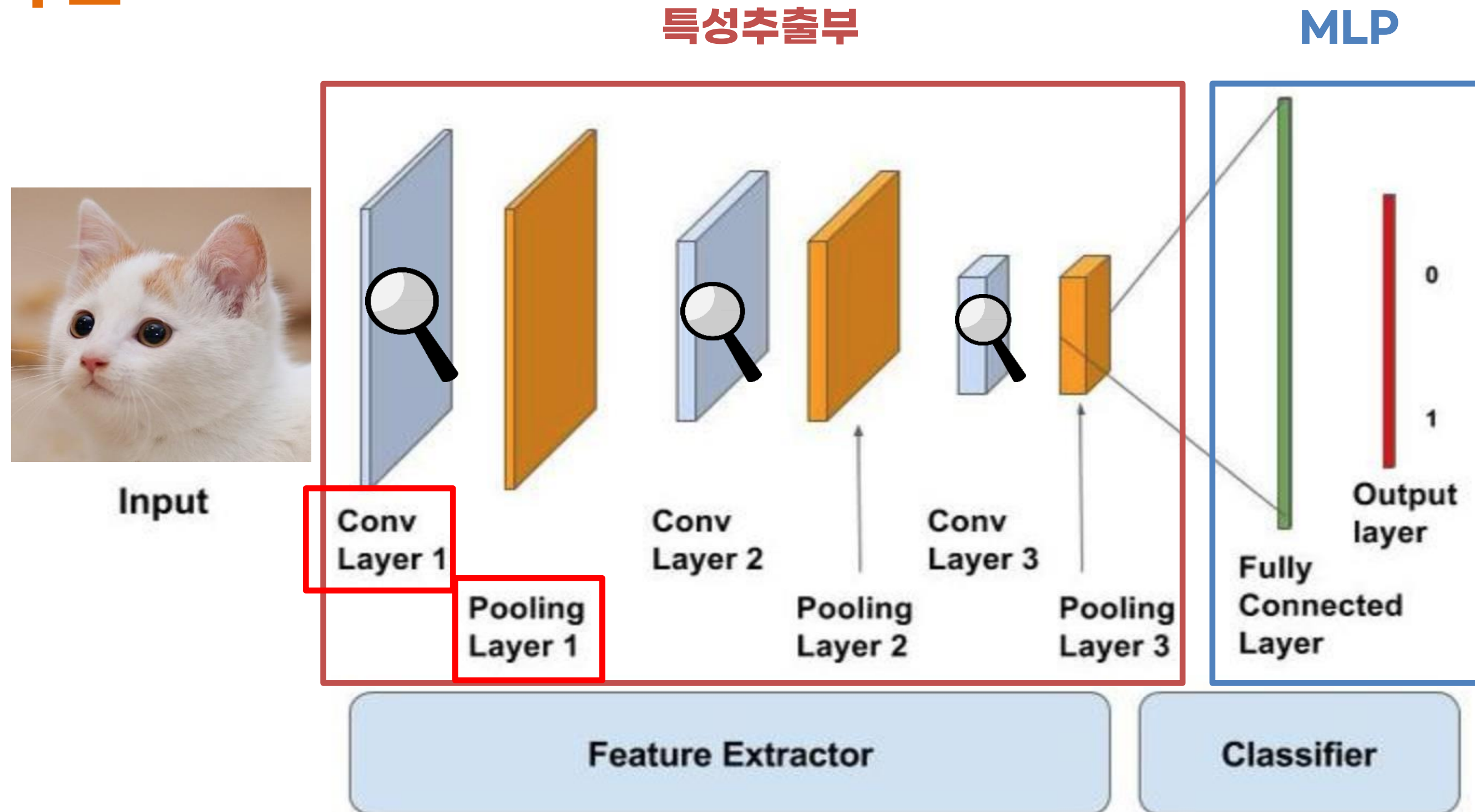
CNN 제안 배경

- 고양이의 눈으로 보는 사물의 형태에 따라 뇌의 특정영역(특정뉴런)만 활성화
- 2010 ~ 2013년도 CNN의 획기적인 발전과 다수의 논문이 출시됨



CNN(Convolutional Neural Network)

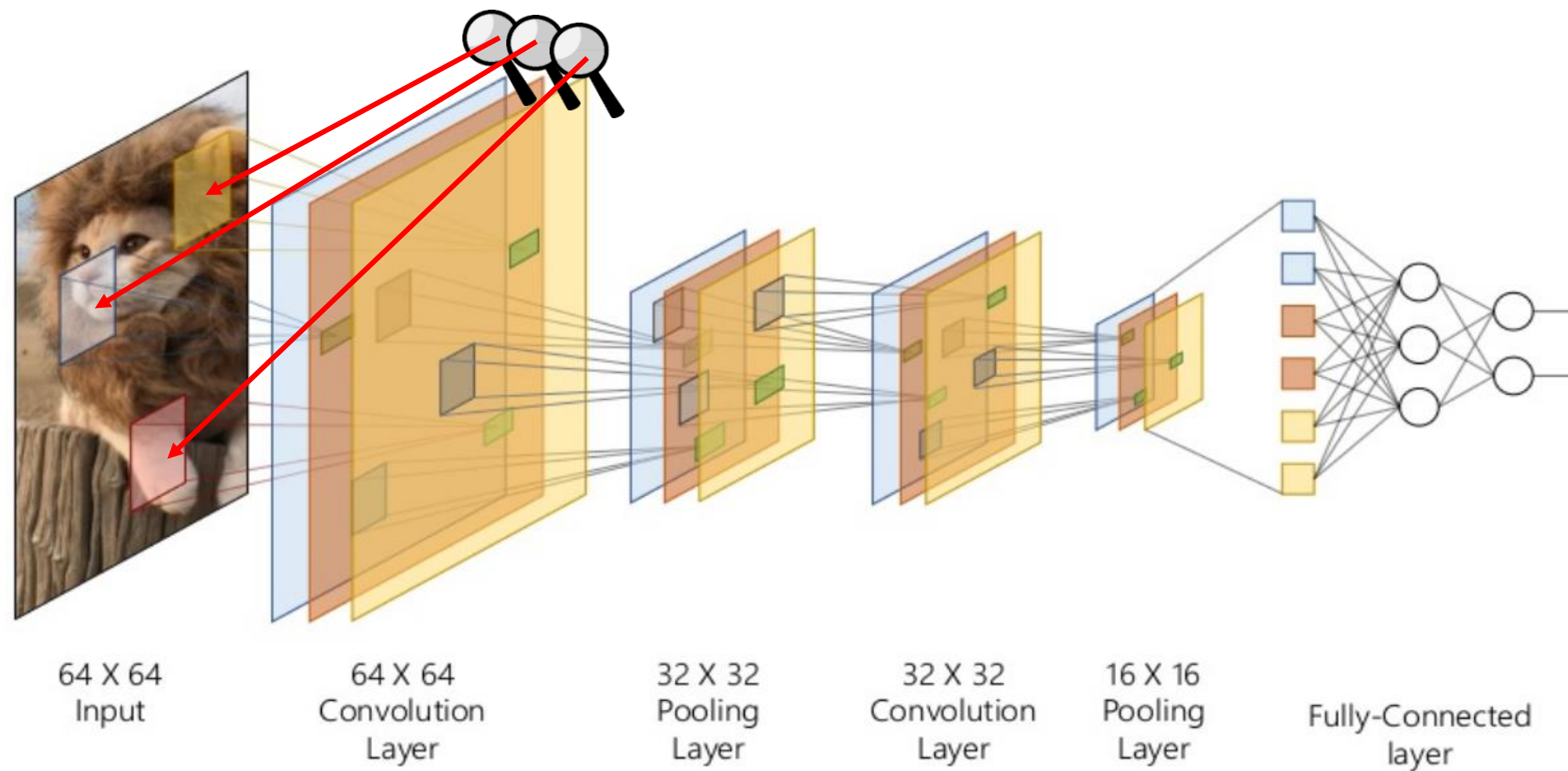
CNN 구조



CNN(Convolutional Neural Network)

CNN 구조

CNN 기본 구조

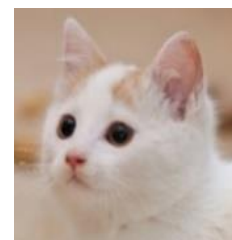
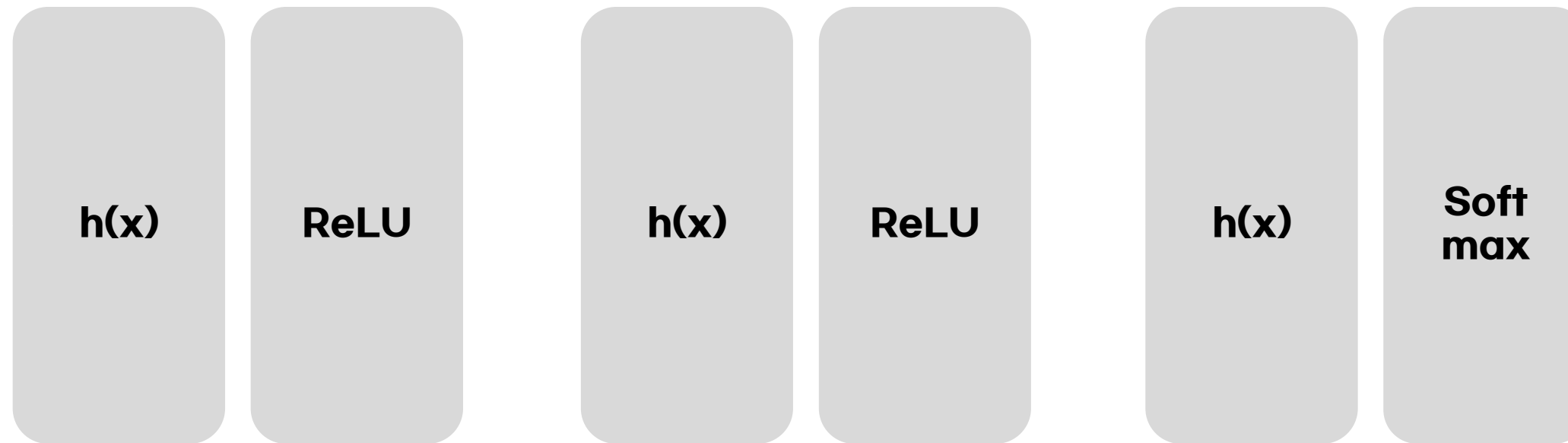


CNN(Convolutional Neural Network)

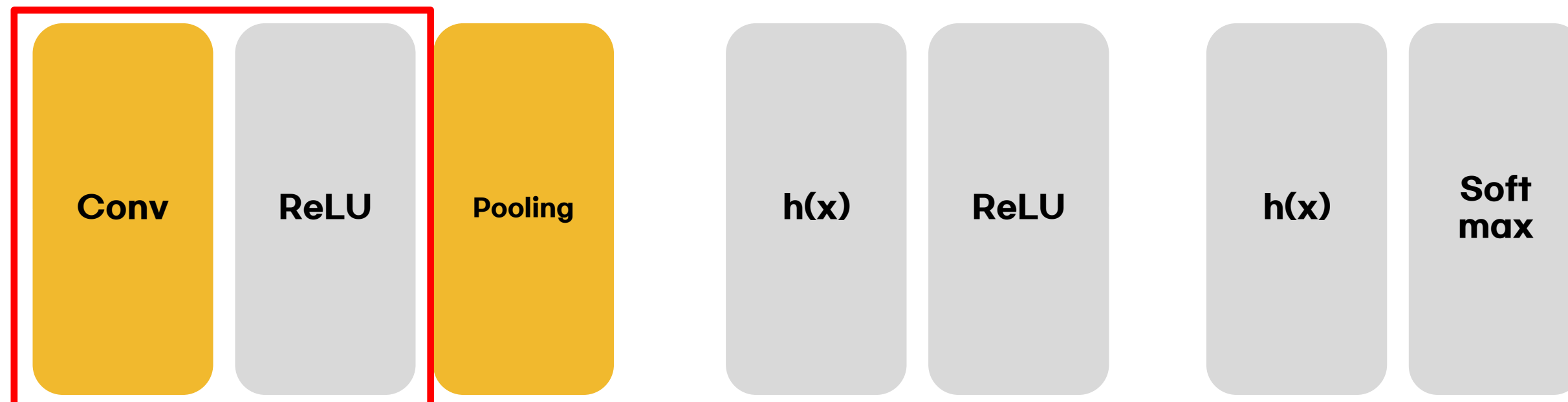
MLP, CNN 구조 비교



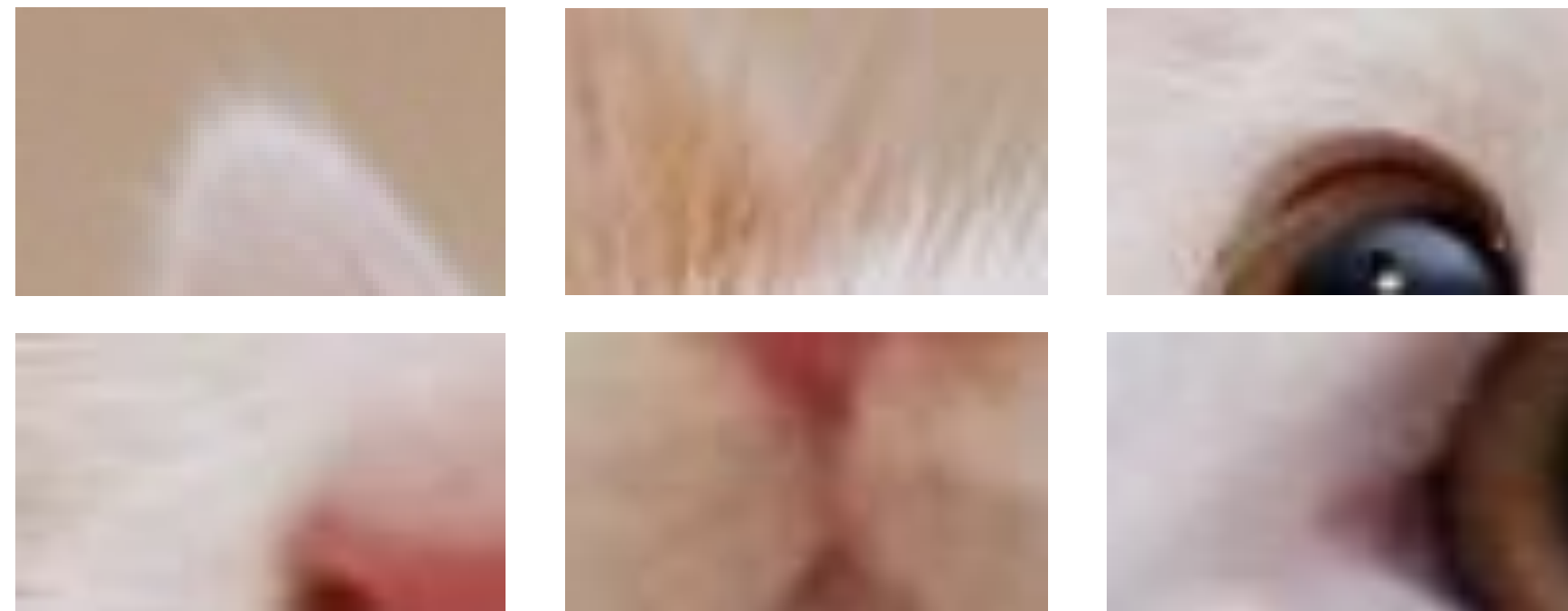
MLP



CNN

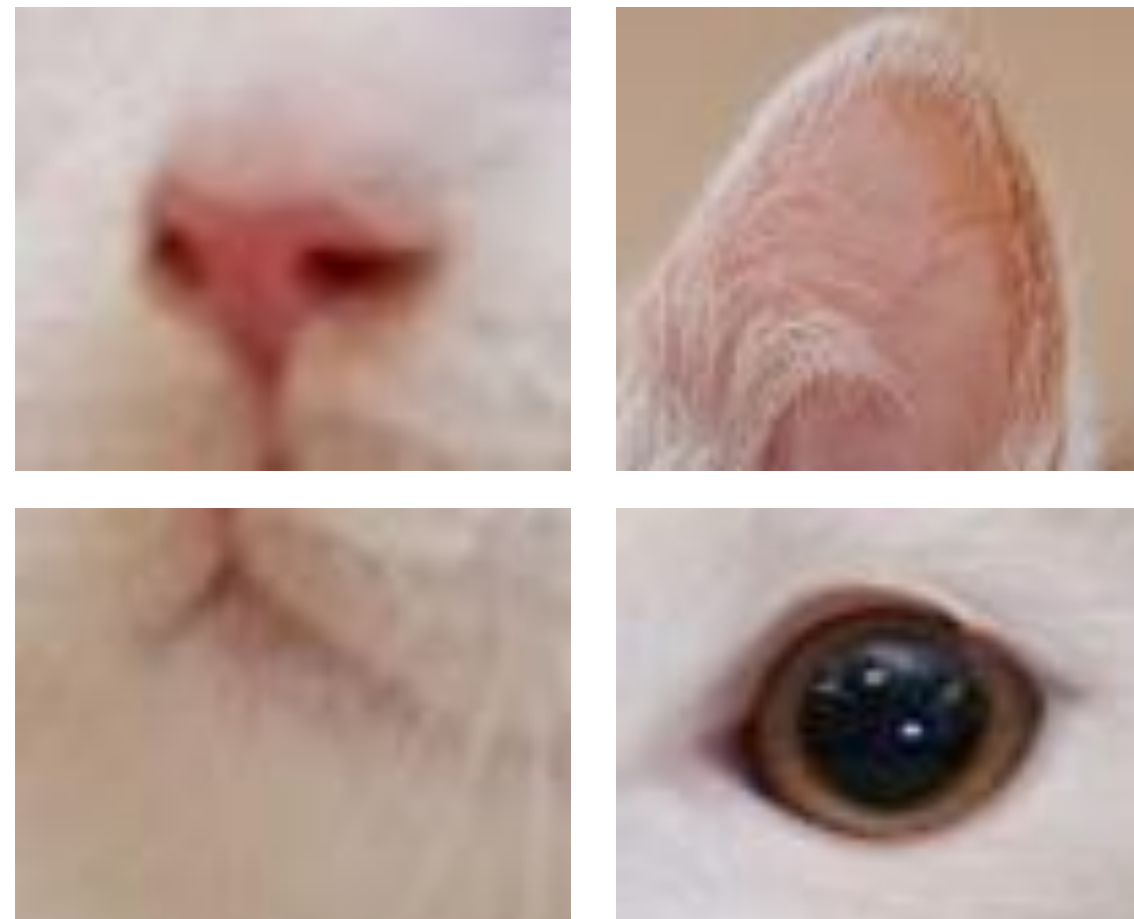


특성추출부의 초반 층들은 데이터의 심플한 특징들을 잡아낸다



1단계 : 세모, 동그라미

특성추출부의 층들이 깊어지면서 조금 더 디테일한 특징들을 잡아낸다



2단계 : 코, 귀, 입, 눈

이런 특징들이 모여서 최종적으로 고양이라고 판단하게 된다



3단계 : 고양이

CNN은 어떤 방식으로 특징을 추출할까?

- CNN은 입력된 이미지에서 특징을 추출할 수 있는 **필터**의 개념을 도입
- 이미지 전체 영역(전체 픽셀)에 대해 서로 동일한 연관성(중요도)으로 처리하는 대신 **특정 범위에 한정해 처리**한다면 훨씬 효과적일 것이라는 아이디어 에서 착안

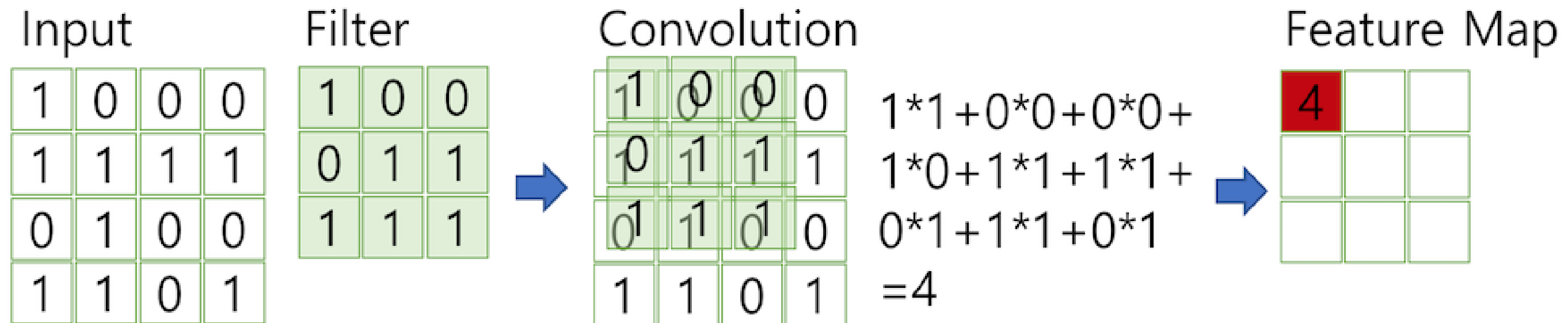
1	0	1	0
1	0	1	0
1	1	1	1
0	0	1	0

이미지

1	0
0	1

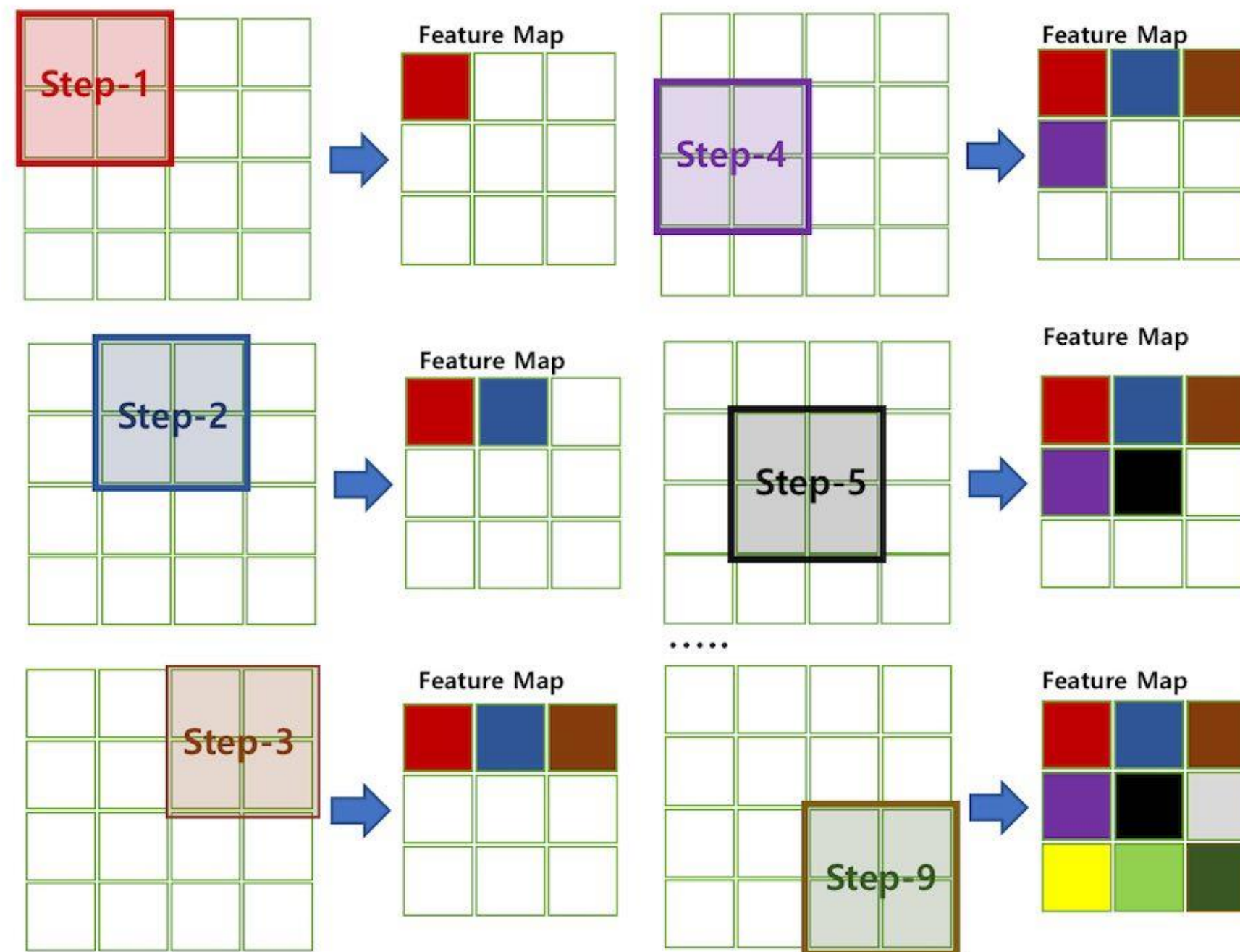
필터
(=kernel)

CNN은 어떤 방식으로 특징을 추출할까?

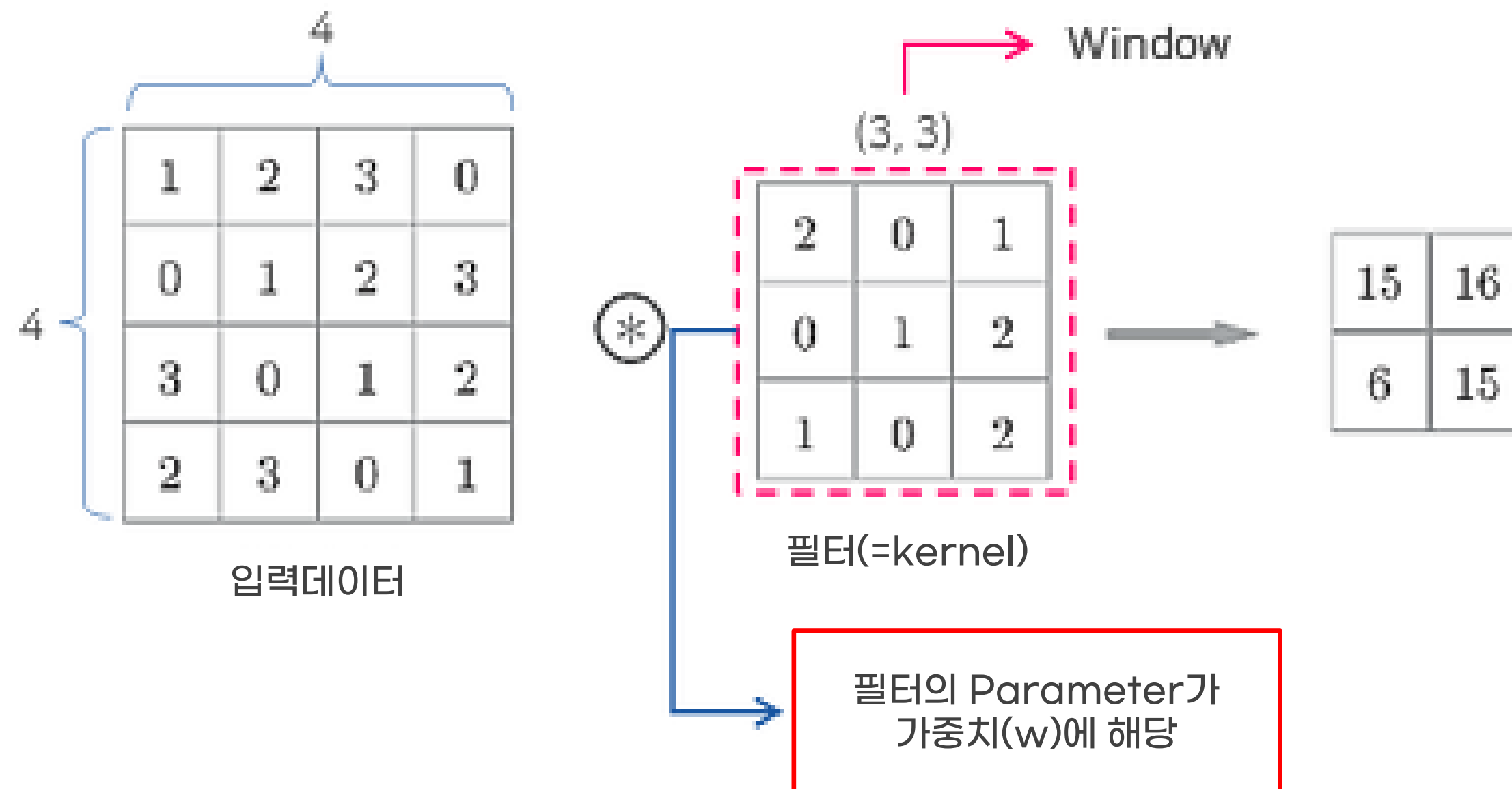


Convolution 연산

CNN은 어떤 방식으로 특징을 추출할까?



CNN은 어떤 방식으로 특징을 추출할까?

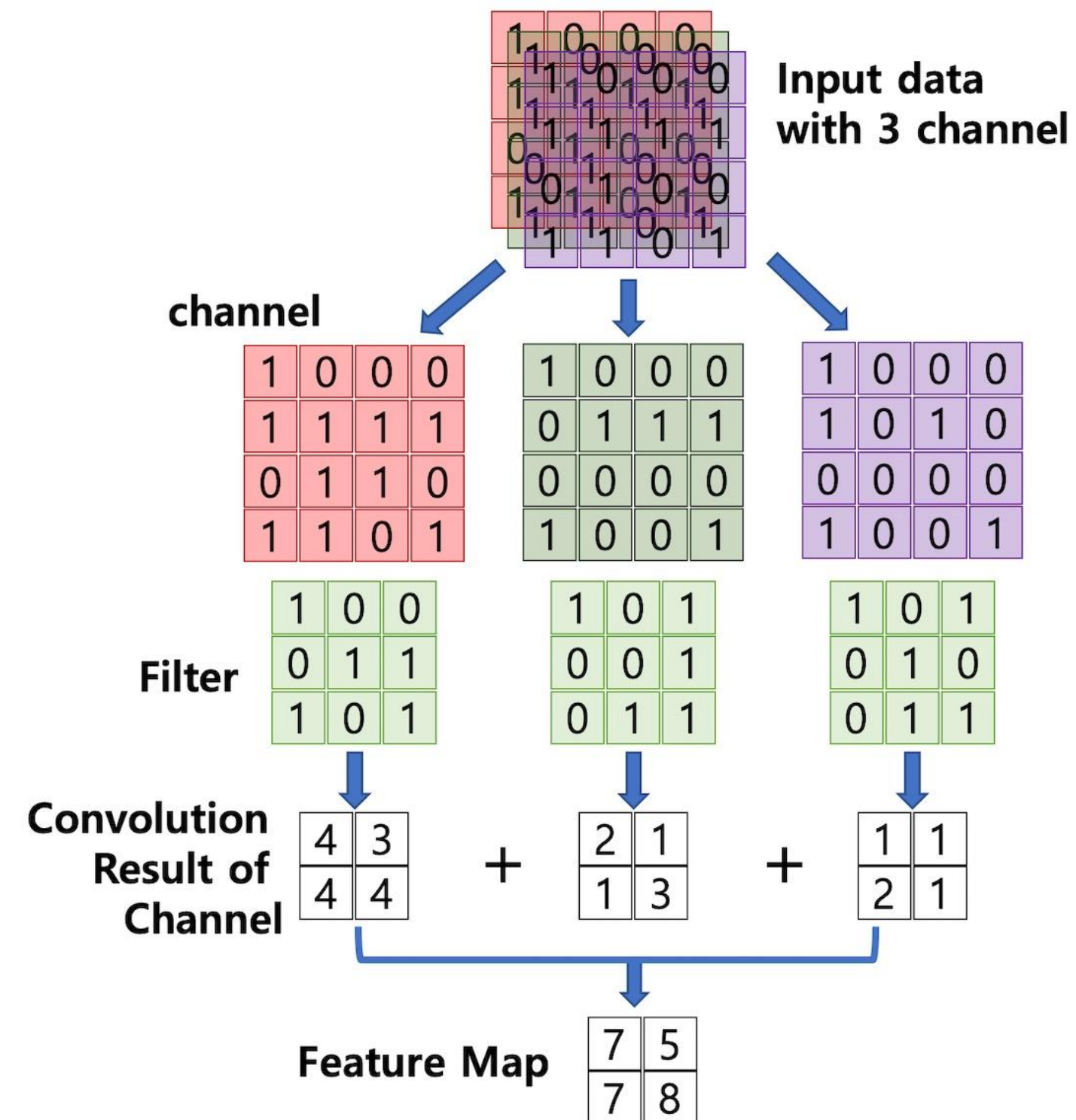


이미지 데이터에서 색상의 개념

- 합성곱 계층에서 이미지의 색상 정보를 **채널**이라고 부름
- **흑백**으로 코딩 된 경우(ex 손글씨 이미지) 흑백의 그레이 스케일 (0:검은색, 255:흰색)만 나타내면 되므로 채널은 **1**이 됨
- 입력신호가 **RGB** 신호로 코딩된 경우, 채널은 세가지 색을 각각 나타내는 **3**이 됨(데이터의 색상 정보를 유지 할 수 있음)



컬러 이미지 Convolution 연산



패딩(Padding)

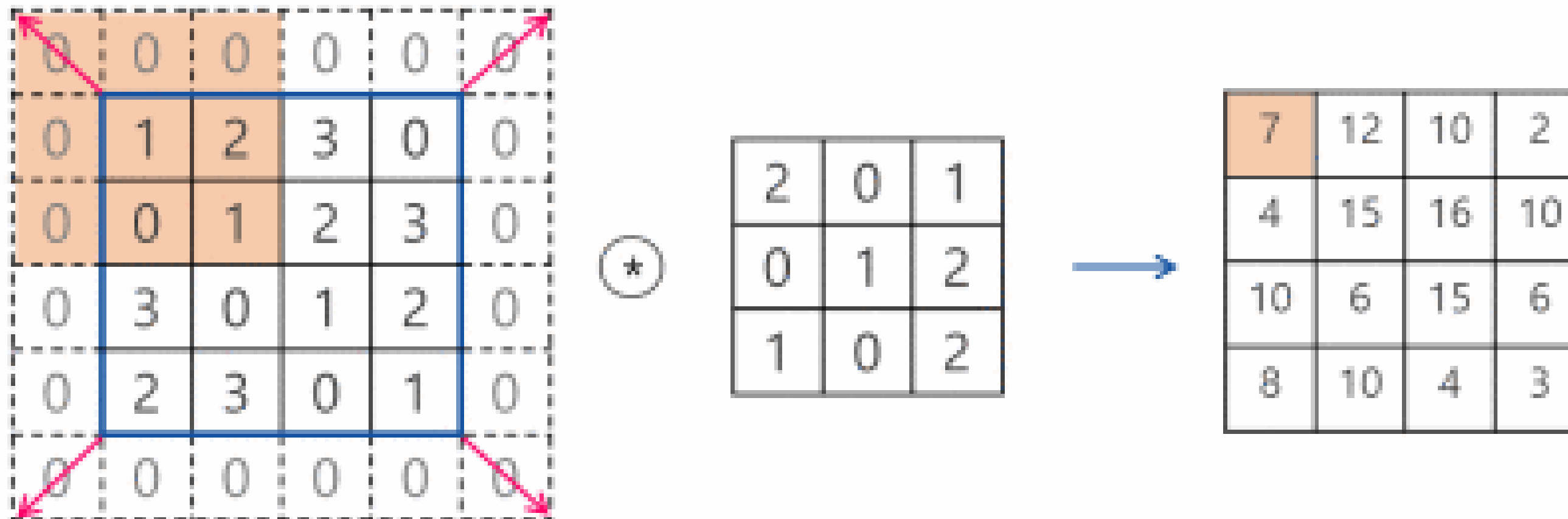
- 필터의 크기로 인해 가장자리 부분의 데이터가 부족해서 입력과 출력 이미지의 크기가 달라지게 되는데 이를 보완하기 위해서 **입력데이터의 가장자리 부분에 0**을 미리 채워 넣는 것을 패딩(padding) 이라고 함
- 패딩을 사용하면 특정 층의 입력과 출력의 크기를 같게 맞춰줄 수 있음
- 즉, 층이 깊어지면서 **이미지의 크기가 줄어드는 것을 방지**

Conv2D 계층에서는 padding 명령을 사용해 패딩을 지정할 수 있음

- ▶ **same**으로 지정하면 출력과 입력이 같아지게 적절한 수의 패딩을 자동으로 입력
- ▶ **valid**로 설정하면 패딩을 사용하지 말라는 뜻

CNN(Convolutional Neural Network)

패딩(Padding)



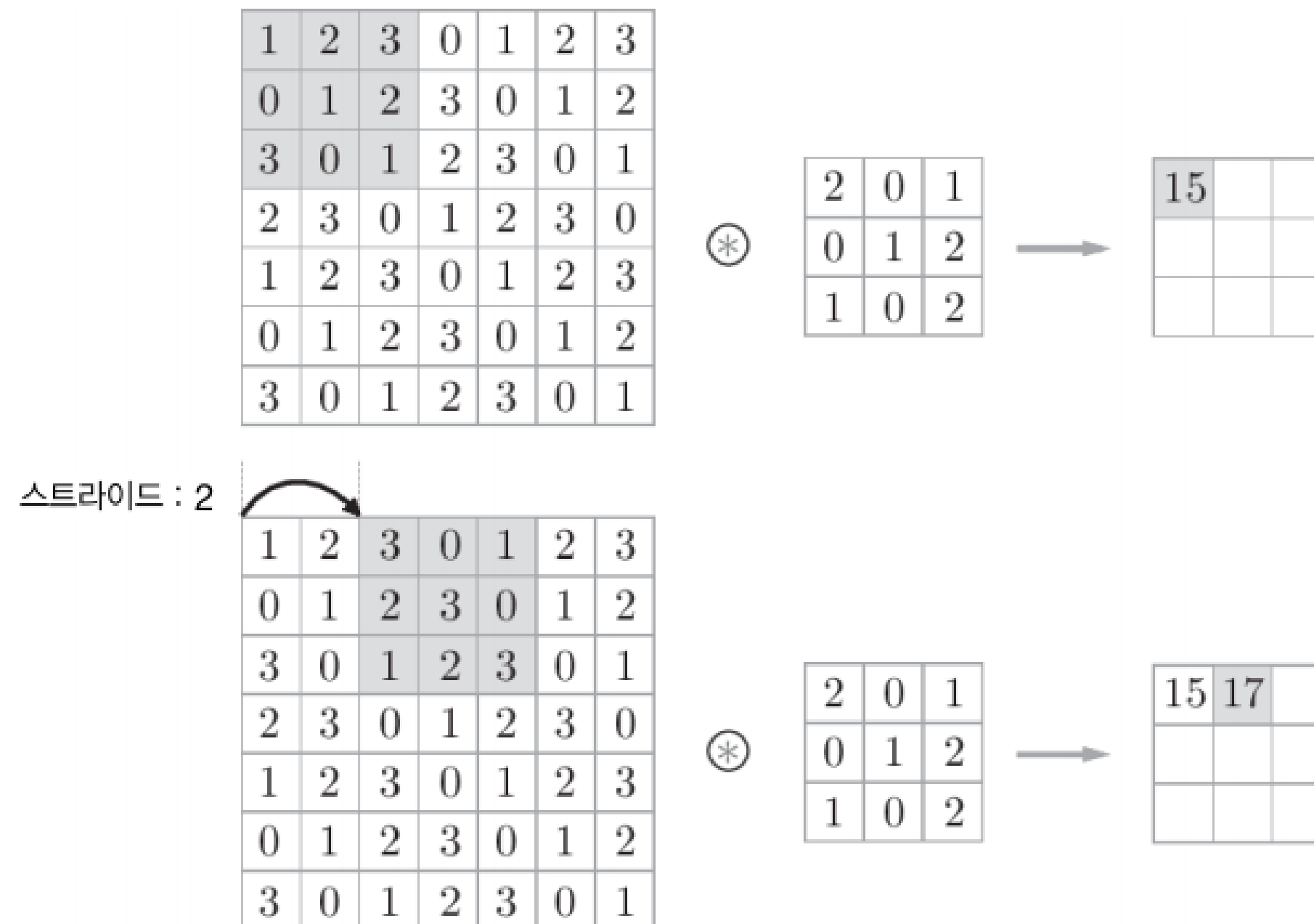
축소샘플링

- 합성곱을 수행한 결과 신호를 다음 계층으로 전달할 때, **모든 정보를 전달하지 않고 일부만 샘플링** 하여 넘겨주는 작업을 축소 샘플링(subsampling)이라고 함
- 축소 샘플링을 하는 이유는 **좀 더 가치 있는 정보**만을 다음 단계로 넘겨주기 위함
- 이미지 처리에서 원하는 결과를 얻기 위해서는 가치 있는 정보를 줄여가야 하며, 결국 핵심 정보만 다음 계층으로 전달하는 장치가 필요함
- 축소 샘플링에는 크게 **스트라이드(stride)**와 **풀링(pooling)** 두 가지 기법이

스트라이드(Stride)

- 스트라이드는 합성곱 연산을 수행할 때 필터를 한 픽셀씩 옆으로 이동하면서 출력을 얻는게 아니라, 2 혹은 3 픽셀씩 건너 뛰면서 합성곱 연산을 수행하는 방법
- 이를 스트라이드 2 또는 스트라이드 3이라고 하는데, 이렇게 하면 출력 데이터 (특성맵)의 크기를 $1/4$ 또는 $1/9$ 로 줄일 수 있음

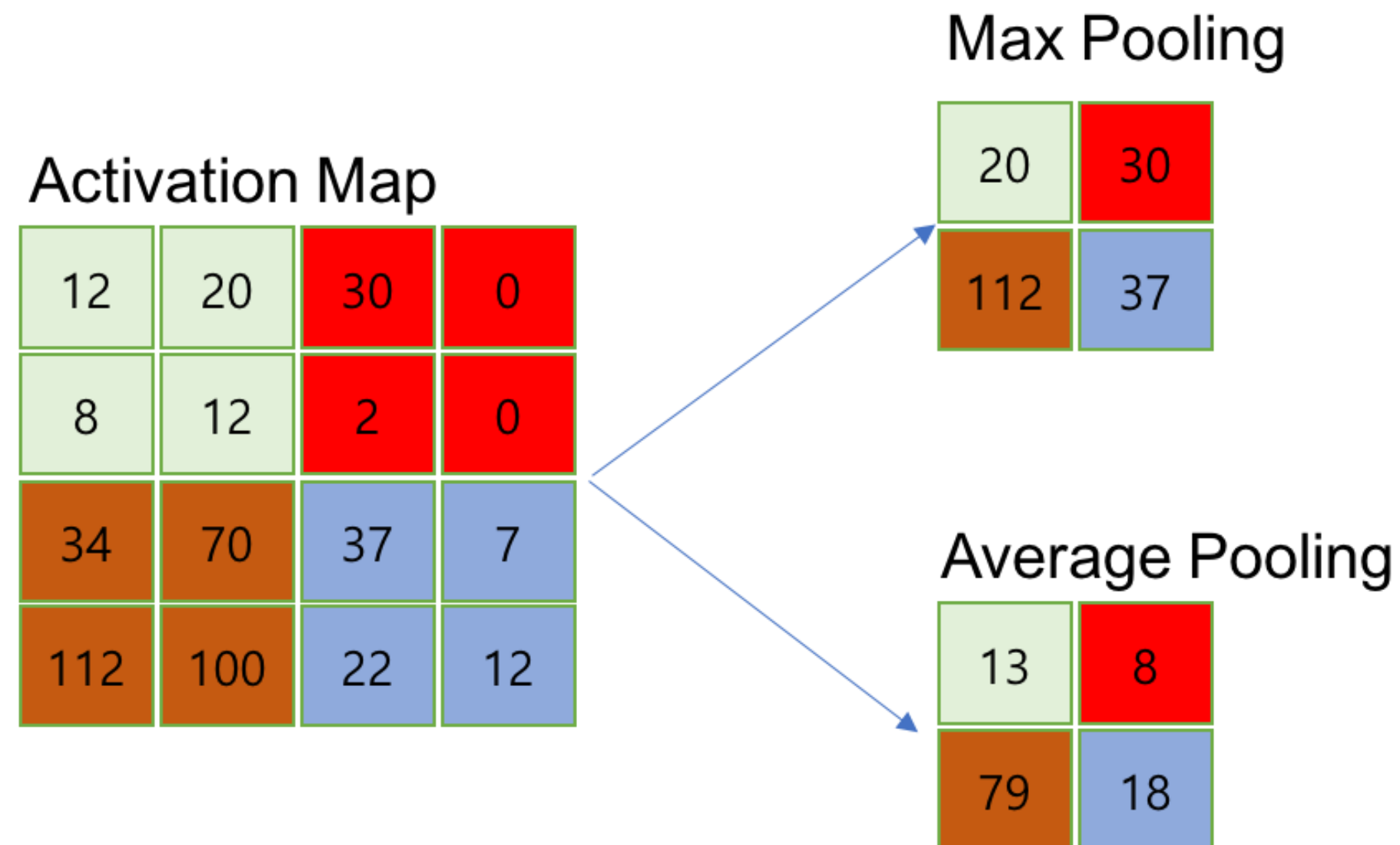
스트라이드(Stride)



풀링(Pooling)

- 풀링이란 CNN에서 합성곱 수행 결과를 다음 계층으로 모두 넘기지 않고, 일정 범위 내(가로,세로)에서 가장 큰 값을 하나만 선택하여 넘기는 방법
- 이렇게 지역내 최대 값만 선택하는 풀링을 max pooling이라 함
- max pooling을 하면 작은 지역 공간의 대표 정보만 남기고 나머지 신호들을 제거하는 효과를 얻을 수 있음
- 지역내 평균 값을 선택하는 average pooling도 있음

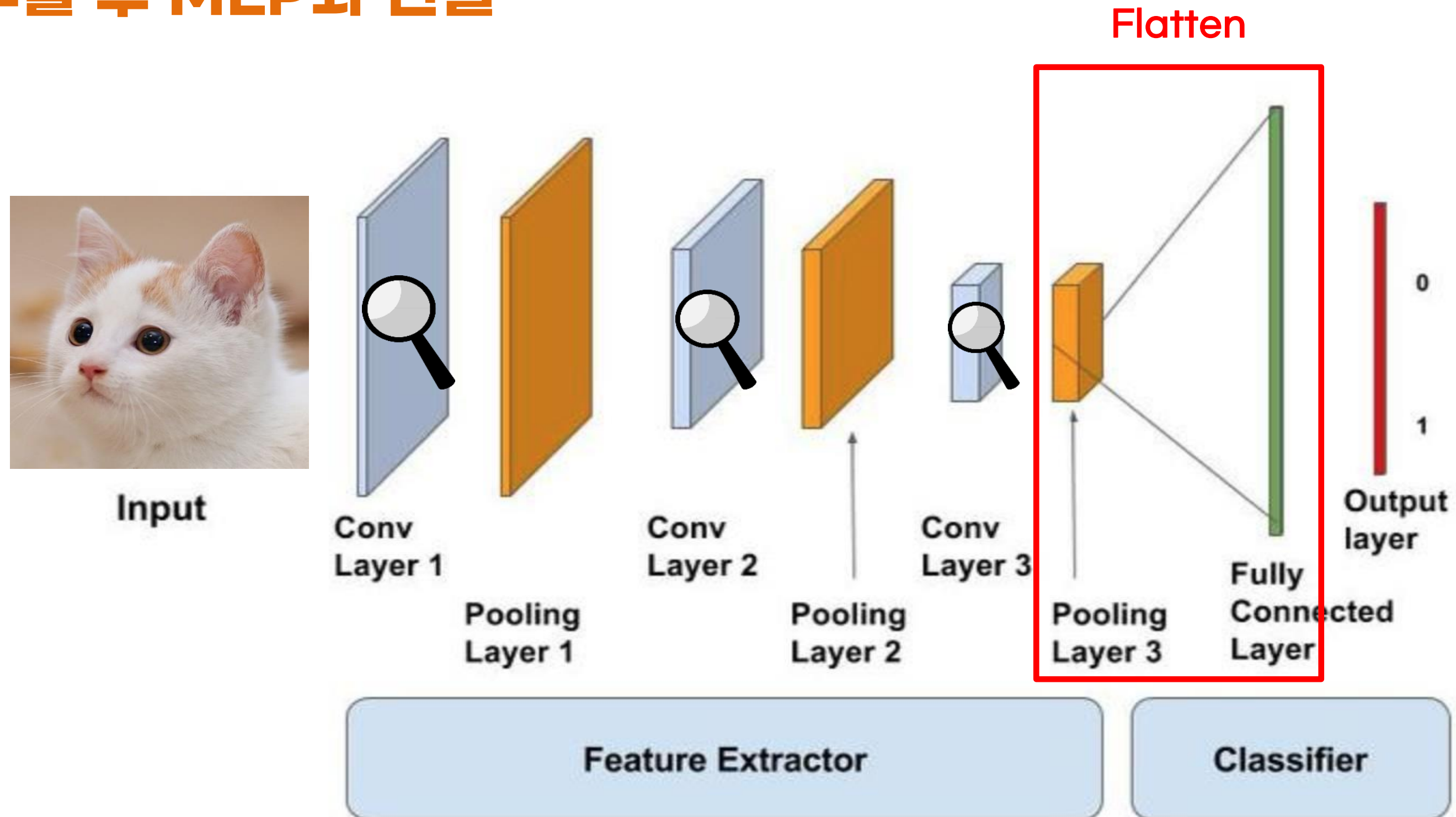
풀링(Pooling)



Pooling size가 2x2인 경우

CNN(Convolutional Neural Network)

특징 추출 후 MLP와 연결



웹 크롤링과 keras를 활용한 3종 동물 이미지 분류

전이학습 (Transfer Learning)

| 이미 잘 학습된 모델을 우리도 사용할 수 있을까?



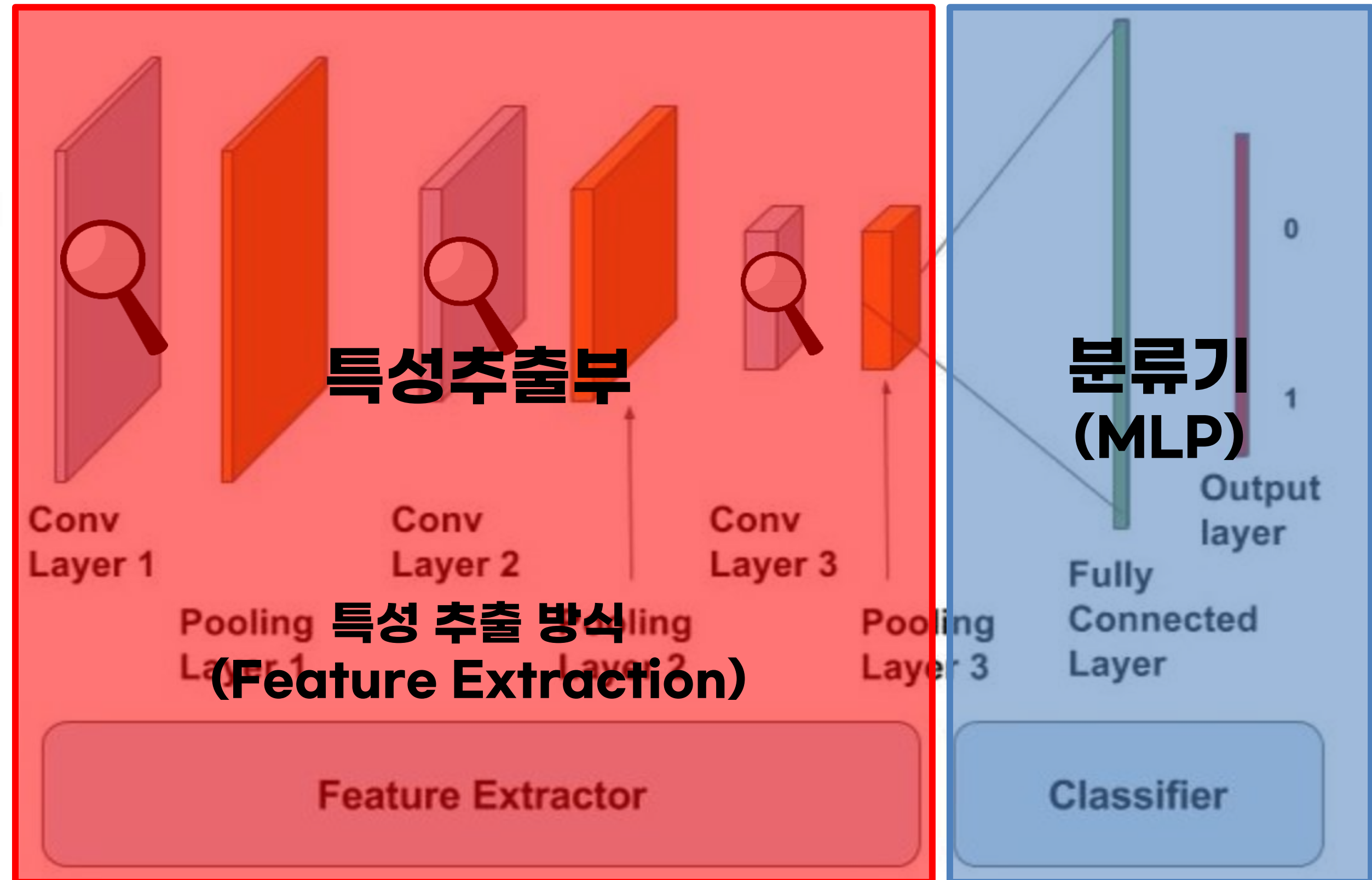
전이학습

- 전이학습이란 다른 데이터 셋으로 이미 학습한 모델을 유사한 다른 데이터를 인식하는데 사용하는 기법
- 사람으로 치면 잘 학습된 사람의 지식을 그대로 가져와서 문제 해결에 사용하는 식
- 이 방법은 특히 새로 훈련시킬 데이터가 충분히 확보되지 못한 경우에 높은 학습 효율을 높여줌
- 전이학습 모델을 적절히 이용하는 방법은 특성 추출(feature extraction)방식과 미세 조정(fine-tuning)방식이 있음

전이학습(Transfer Learning)



Input



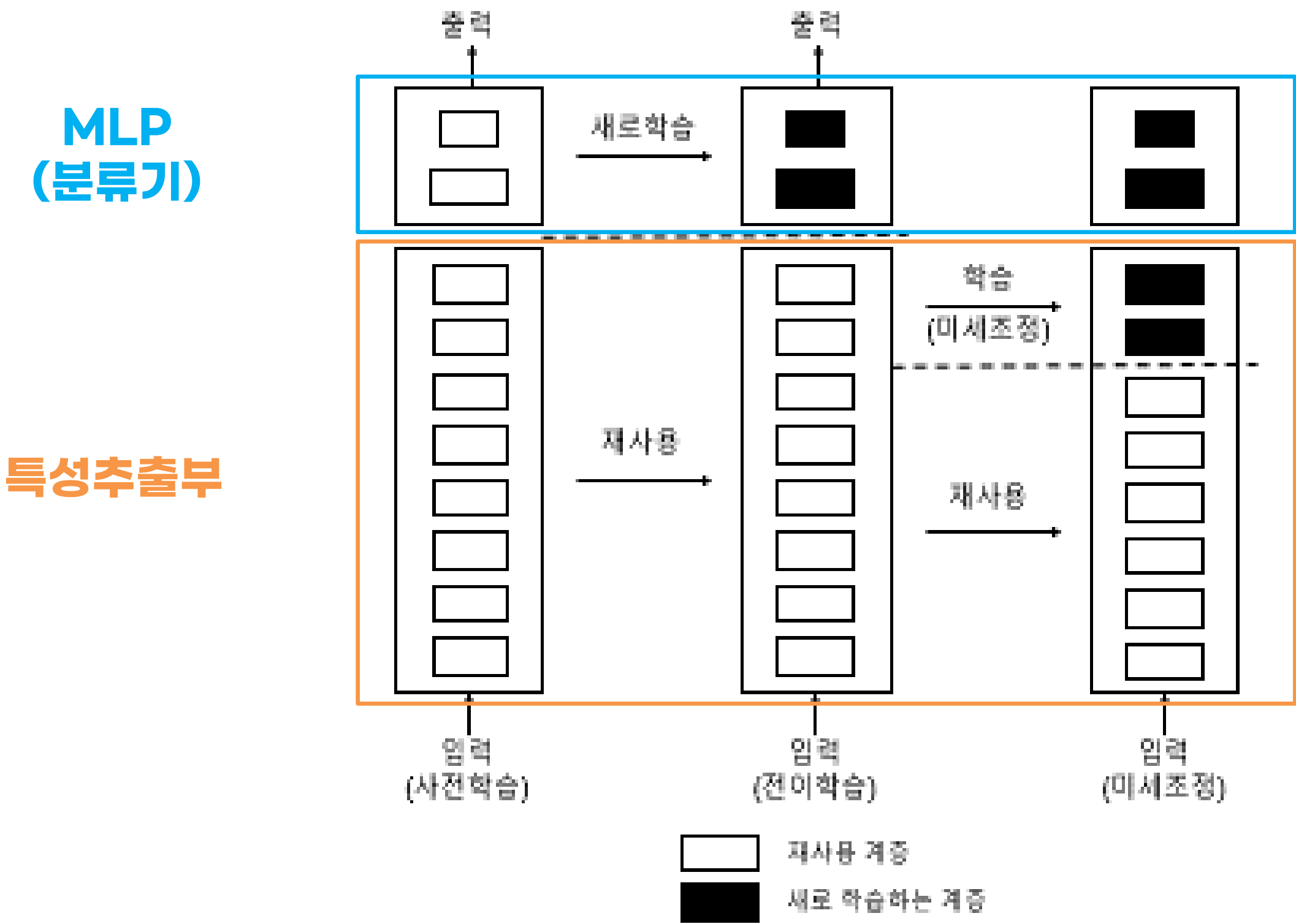
전이학습 - 특성추출방식

- CNN 층에서 특성추출부만 가져와서 사용하는 방식
- 특성추출부 부분만 사용하는 이유는 분류기(MLP)의 경우 우리가 해결 하고자 하는 문제에 맞게 새로 설정해줘야 하기 때문
- 단, 새롭게 분류할 클래스의 종류가 사전 학습에 사용된 데이터와 특징이 매우 다르다면, 특성추출부 전체를 재사용해서는 안되고 앞 단의 일부 계층만을 재사용해야 함(심플한 특징들만 추출해내기 위해)

전이학습 - 미세조정방식

- '사전 학습된 모델의 가중치' 를 목적에 맞게 전체 또는 일부를 재학습시키는 방식
- 특성추출부의 층들 중 하단부 몇 개의 계층은 **분류기(MLP)**와 함께 **새로 학습**시킴
- 처음부터 특성추출부 계층들과 분류기(MLP)를 같이 훈련시키면 새롭게 만든 분류기에서 발생하는 큰 에러 값으로 인해, 특성추출부에서 **사전 학습된 가중치가 많이 손실**될 수 있음
- 처음에는 분류기(MLP)의 파라미터가 랜덤하게 초기화 되어 있으므로 컨볼루션 베이스 중 **앞 단 계층들을 고정(동결)**하고 뒷 단의 일부 계층만 학습이 가능하게 설정한 후, MLP와 같이 학습시켜 파라미터(w, b) 들을 적당하게 잡아 줌

전이학습(Transfer Learning)



VGG-16
Architectures
ImageNet Challenge
2014 준우승 모델

ConvNet Configuration					
A	A-LRN	B	C	D	E
11 weight layers	11 weight layers	13 weight layers	16 weight layers	16 weight layers	19 weight layers
input (224 × 224 RGB image)					
conv3-64	conv3-64 LRN	conv3-64 conv3-64	conv3-64 conv3-64	conv3-64 conv3-64	conv3-64 conv3-64
maxpool					
conv3-128	conv3-128	conv3-128 conv3-128	conv3-128 conv3-128	conv3-128 conv3-128	conv3-128 conv3-128
maxpool					
conv3-256 conv3-256	conv3-256 conv3-256	conv3-256 conv3-256	conv3-256 conv3-256 conv1-256	conv3-256 conv3-256 conv3-256	conv3-256 conv3-256 conv3-256 conv3-256
maxpool					
conv3-512 conv3-512	conv3-512 conv3-512	conv3-512 conv3-512	conv3-512 conv3-512 conv1-512	conv3-512 conv3-512 conv3-512	conv3-512 conv3-512 conv3-512 conv3-512
maxpool					
conv3-512 conv3-512	conv3-512 conv3-512	conv3-512 conv3-512	conv3-512 conv3-512 conv1-512	conv3-512 conv3-512 conv3-512	conv3-512 conv3-512 conv3-512 conv3-512
maxpool					
FC-4096					
FC-4096					
FC-1000					
soft-max					

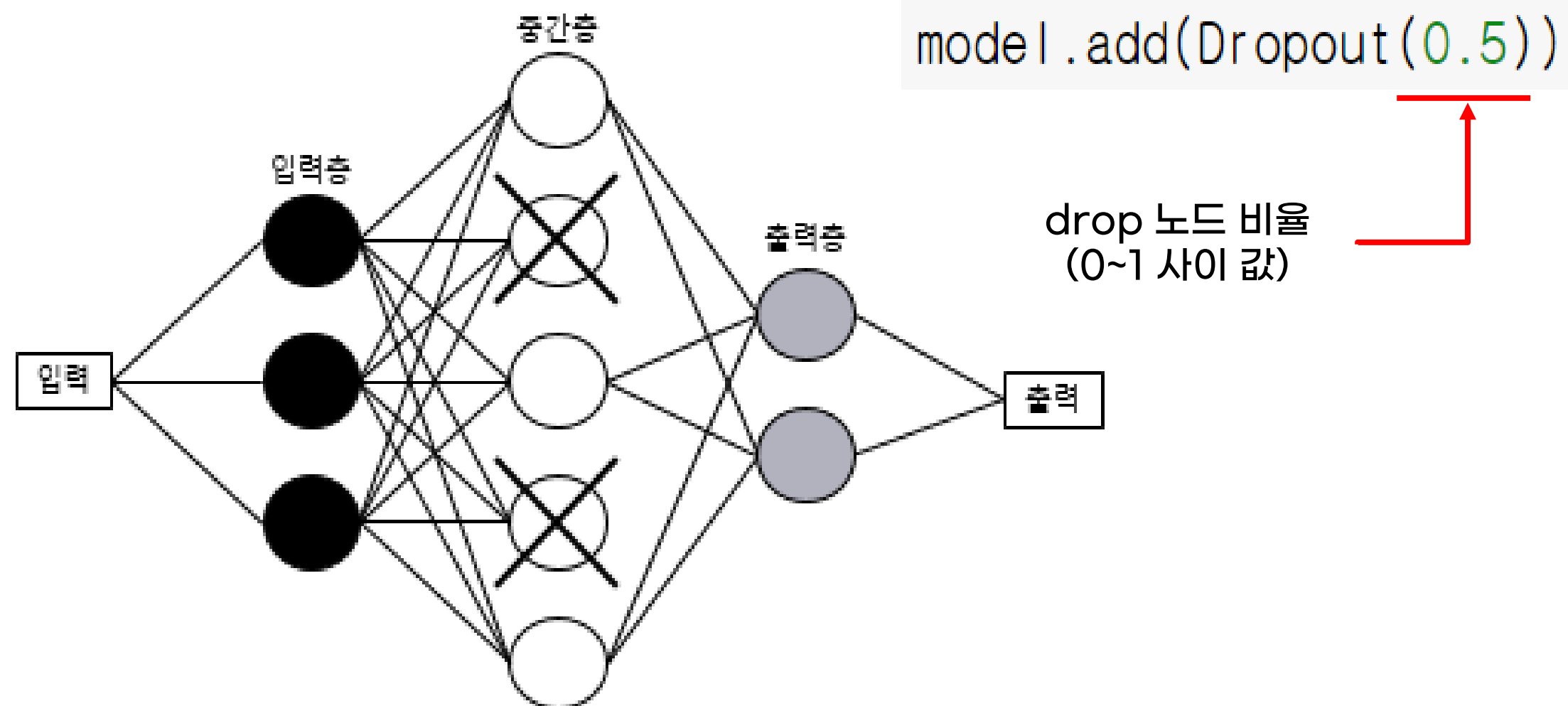
웹 크롤링과 keras를 활용한 3종 동물 이미지 분류 (전이학습 적용)

과대적합을 피하는 방법

- 학습조기중단 (early stopping)
과대적합이 되기 전까지만 모델 학습
- 드롭아웃(dropout)
신경망 중간층 뉴런 일부를 비 활성화시켜 과대적합을 방지
- 데이터 증강(data augmentation) - 이미지 증식
원본과 유사한 데이터를 생성하여 폭넓은 학습에 도움

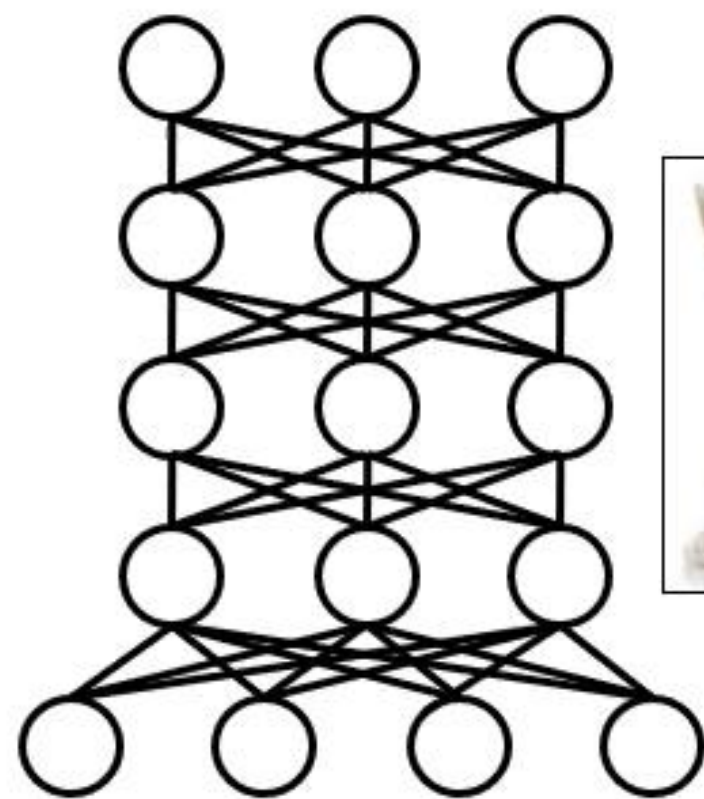
과대적합을 피하는 방법 - drop out

- 드롭아웃은 일정한 비율만큼 랜덤으로 중간층의 뉴런을 비활성화시켜, 신경망이 과도하게 학습되는 현상을 방지

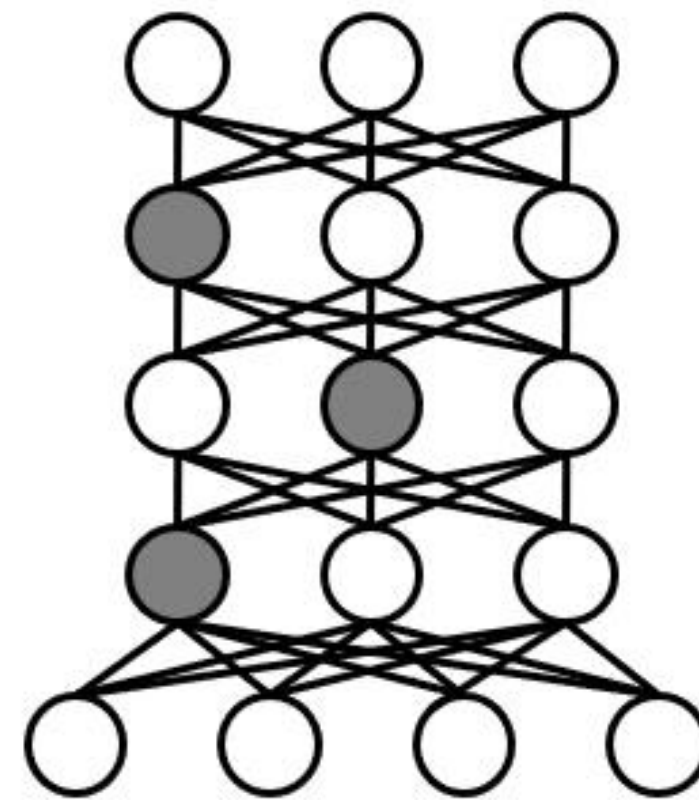


과대적합을 피하는 방법 - drop out

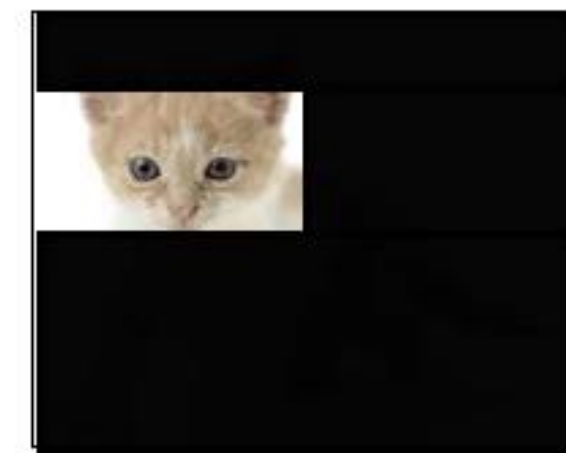
- 드롭아웃은 **학습(역전파)**을 하는 동안에만 적용되고 학습이 종료된 후 예측을 하는 단계에서는 모든 유닛을 사용하여 예측함



예측(순전파)



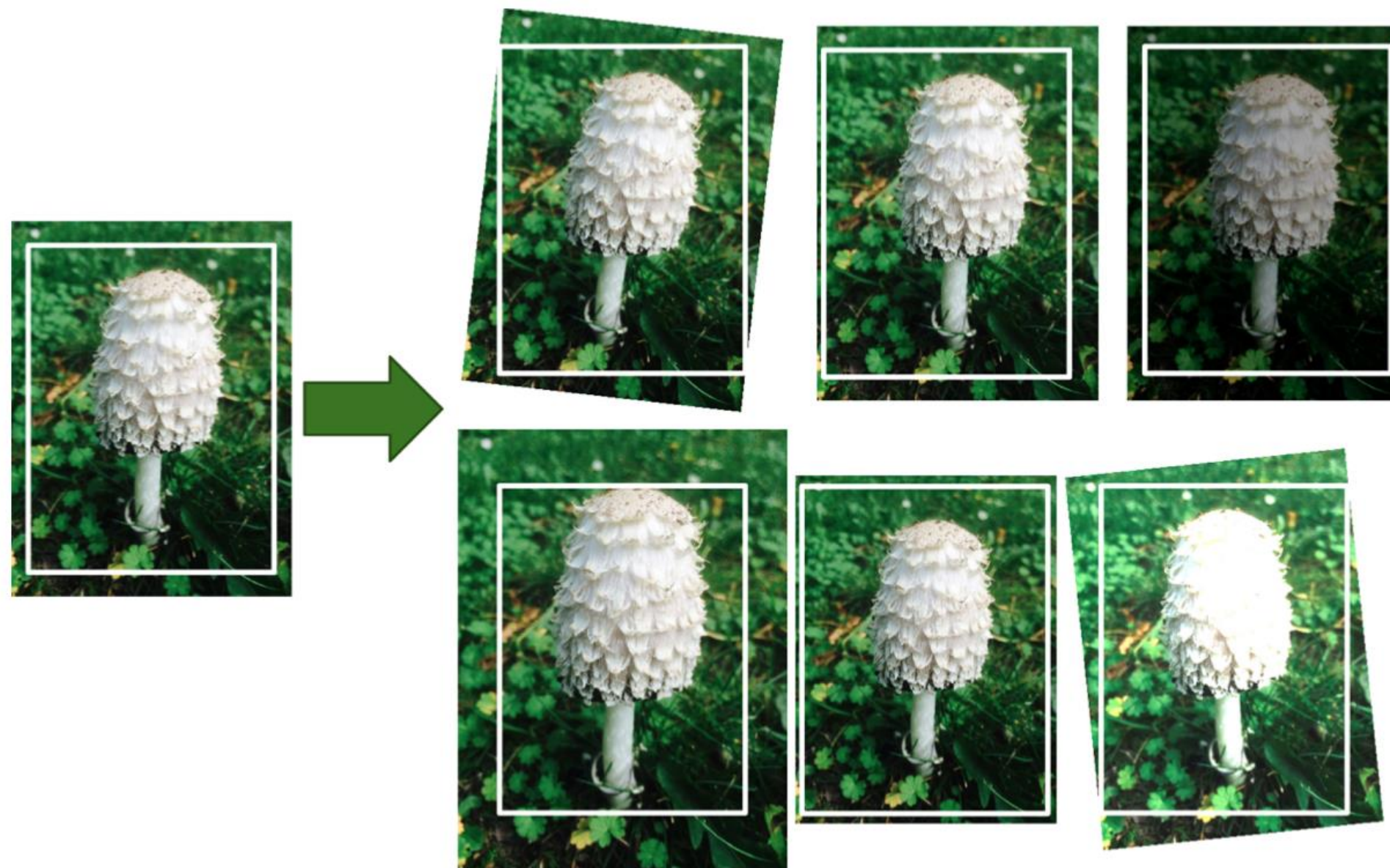
학습(역전파)



과대적합을 피하는 방법 - Data Augmentation(데이터 증강)

- 과대적합이 일어나는 이유 중 하나는 훈련데이터가 부족하기 때문이며 데이터가 충분히 많다면 과대적합을 줄일 수 있음
- Data Augmentation 이란 **훈련 데이터를 유사하고 다양하게 변형**하여 새로운 훈련 데이터처럼 추가적으로 사용함으로써 마치 훈련 데이터 수가 늘어난 효과를 얻는 것

과대적합을 피하는 방법 - Data Augmentation(데이터 증강)



과대적합을 피하는 방법 - Data Augmentation(데이터 증강)



Image
Augmentation



과대적합을 피하는 방법 - Data Augmentation(데이터 증강)

- rotation_range = 360 → 0° 에서 360° 사이에서 회전
- width_shift_range = 0.1 → 전체에서 10% 내외 수평이동
- height_shift_range = 0.1 → 전체에서 10% 내외 수직이동
- shear_range = 0.5 → 0.5라디안 내외 전단방향으로 변형
- zoom_range = 0.3 → 0.7~1.3배로 축소/확대
- horizontal_flip = True → 수평방향으로 뒤집기
- vertical_flip = True → 수직방향으로 뒤집기

웹 크롤링과 keras를 활용한 3종 동물 이미지 분류 (전이학습+데이터증강 적용)



다음시간에

복습 없이 전부를 이해하려는 것은 정신병 초기 증상이다