딥러닝의 통계적이해

8강. 합성곱 신경망의 응용

1. 전이학습
2. 합성곱 신경망의 응용

한국방송통신대 이긍희 교수

오늘의 학습목표

1. 전이학습을 이해한다.
2. 합성곱 신경망의 활용에 대해 이해한다.

1. 전이학습

딥러닝 모형의 작성

- 높은 정확도의 딥러닝 모형
- 딥러닝 모형의 설계 및 하이퍼파라미터의 조정
  → 대량의 학습데이터, 고성능 컴퓨팅, 긴 학습 시간

전이학습

- 전이학습(transfer learning) :

  이미 훈련된 신경망으로 신경망을 학습

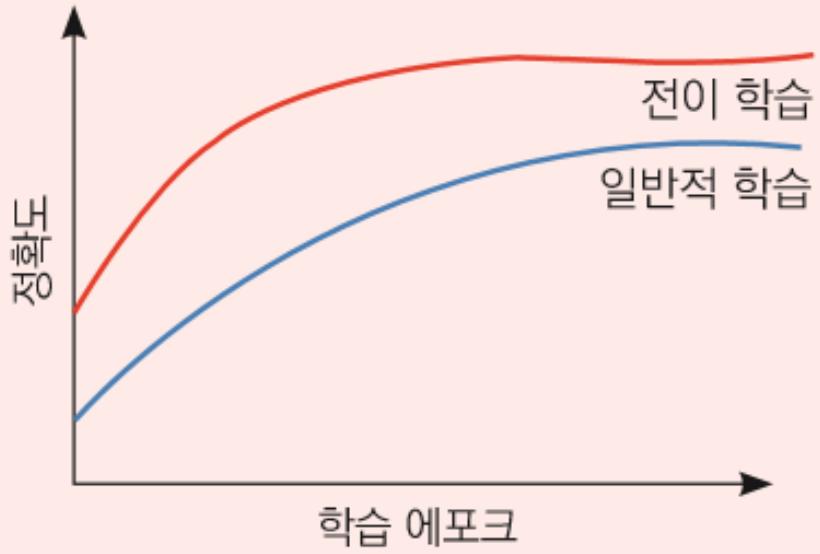
  → 데이터와 컴퓨팅 환경이 충분하지 않다면 이미지넷 경진대회에서 우승했던 모형들의 공개된 가중치 그대로 이용

전이학습

- 식별하고 싶은 이미지가 이미지넷 경진대회의 1,000개 범주
  → 별도 학습 없이 우승 모형 그대로 이용
- 식별하고 싶은 이미지가 이미지넷 1,000개 범주에 없다면
  → 신규 데이터 추가 학습 이용

전이학습

- 전이학습을 이용한 신경망 모형 학습 : 좋은 초깃값



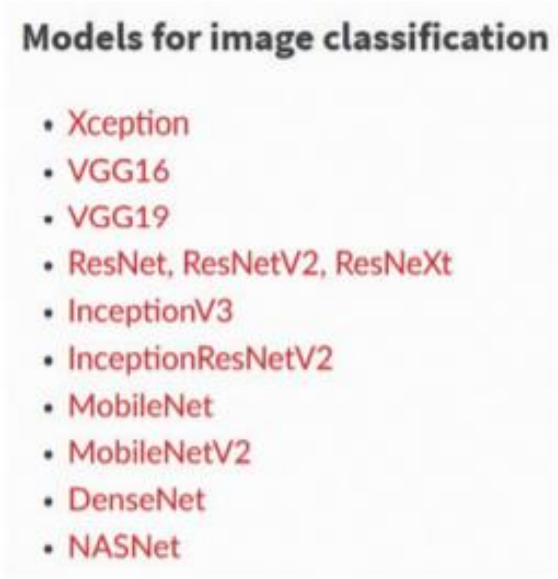
전이학습

◆ 데이터 크기와 유사성에 따른 전이학습 방안

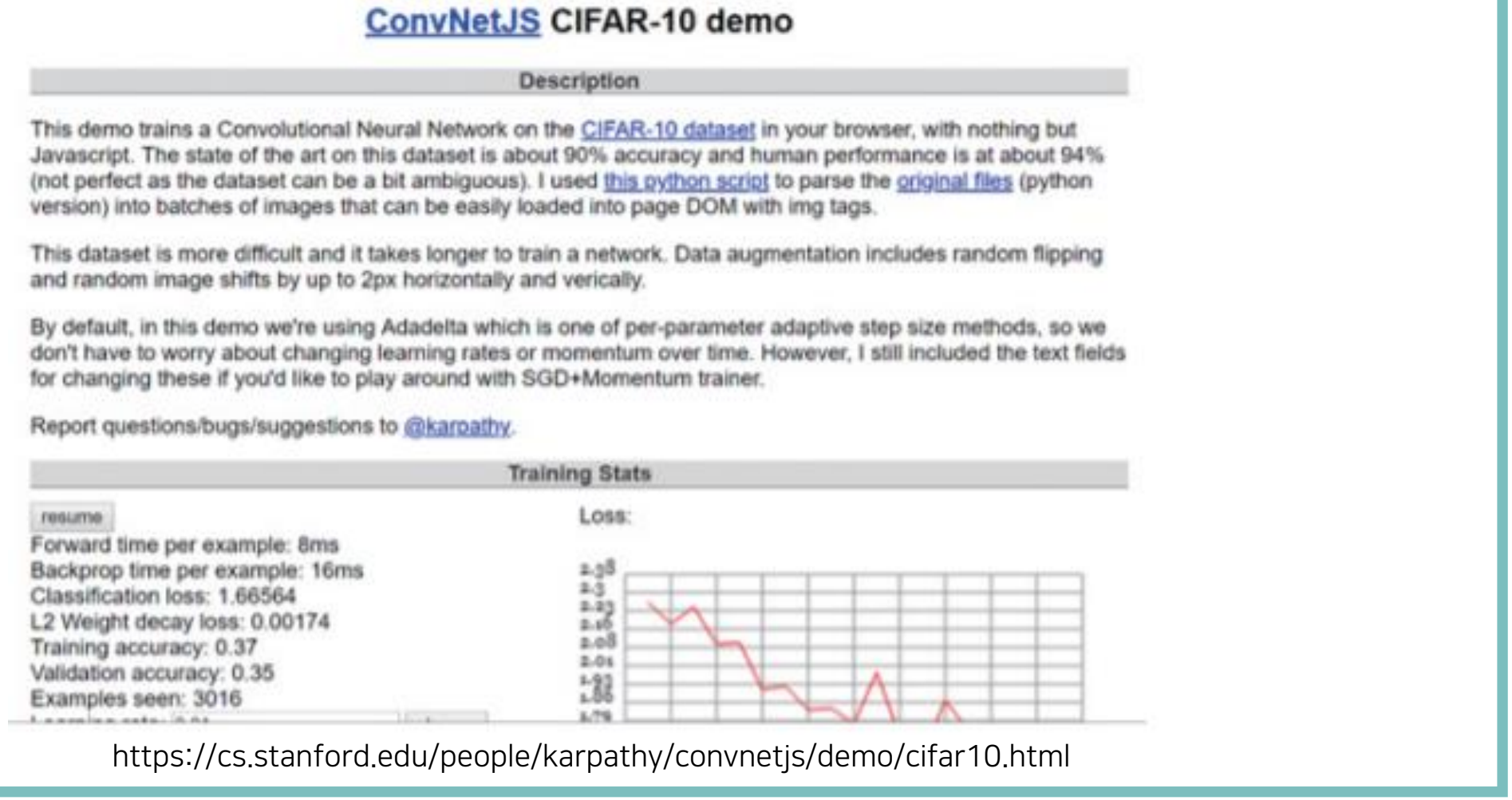
		데이터의 유사성	
		크다	작다
데이터	소규모	기존 모형	학습 어려움
	대규모	일부 층 조정	많은 층 조정

전이학습

- 사전학습된 모형 : Keras나 Pytorch에서 찾을 수 있음
  → TensorFlow Hub에 다양한 모형을 찾을 수 있음

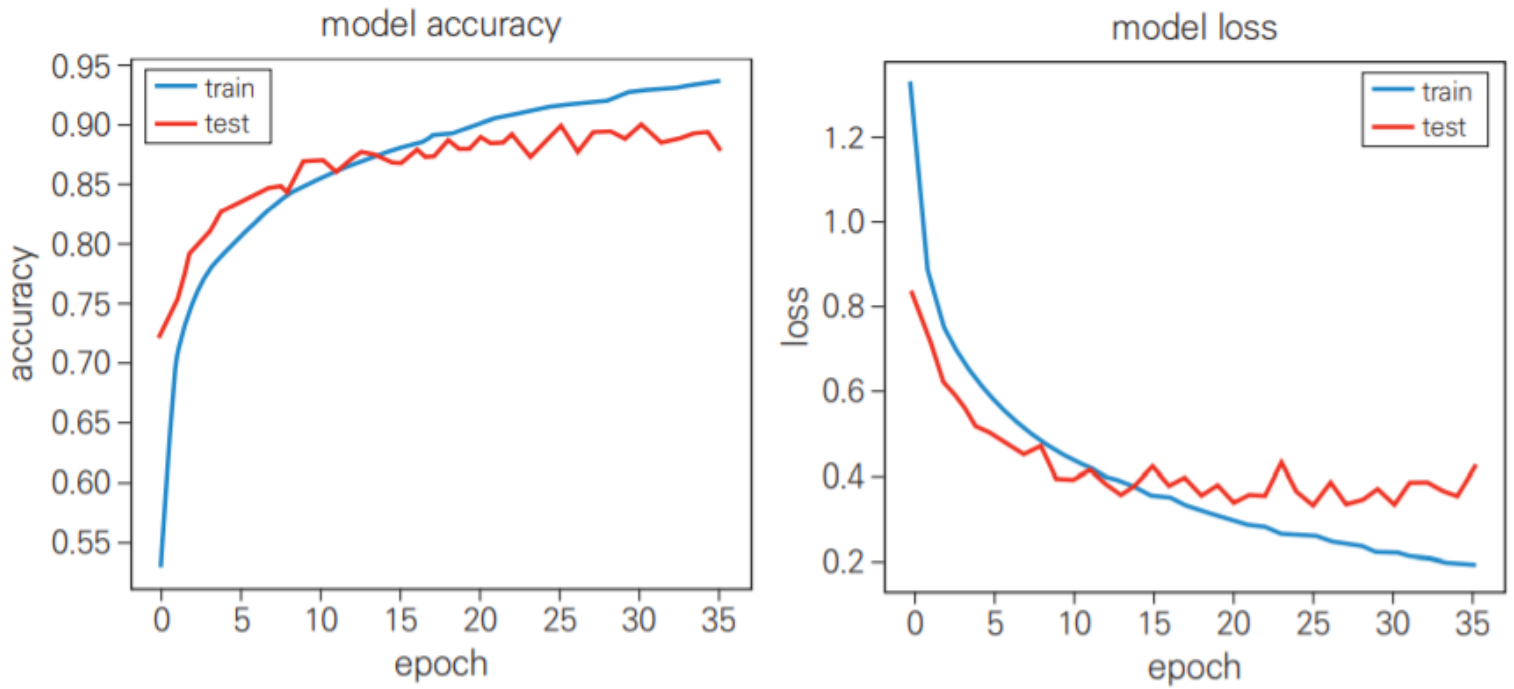


합성곱 신경망 실험

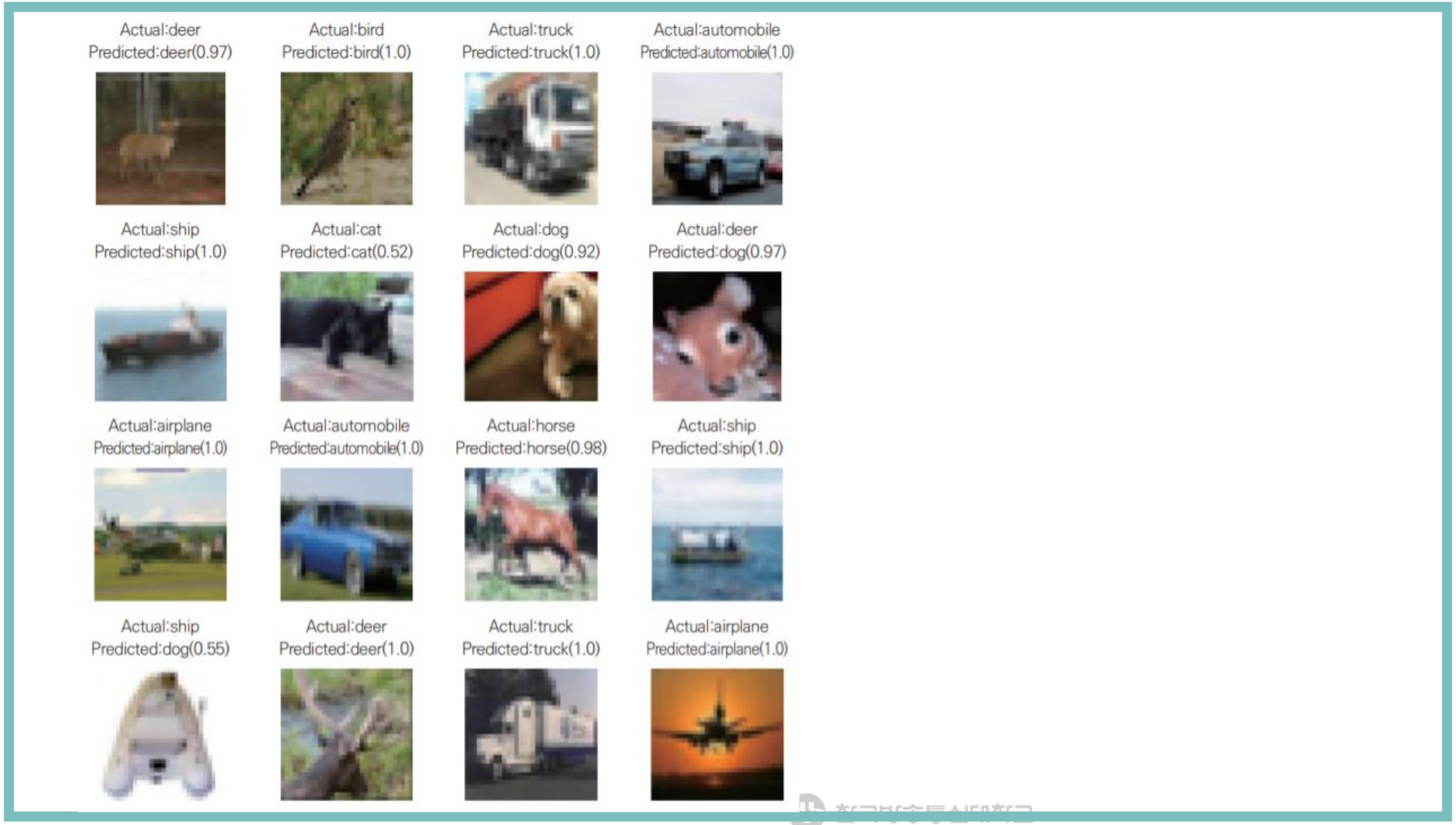


https://cs.stanford.edu/people/karpathy/convnetjs/demo/cifar10.html

VGG16을 이용한 전이학습 결과



VGG16을 이용한 전이학습 결과



2. 합성곱 신경망의 응용

객체 검출

- 객체 검출
  - 이미지 식별 + 이미지 위치 찾는 작업 → 지도학습

    → 지역화(localization)
    : 박스를 쳐서 위치 찾는 것

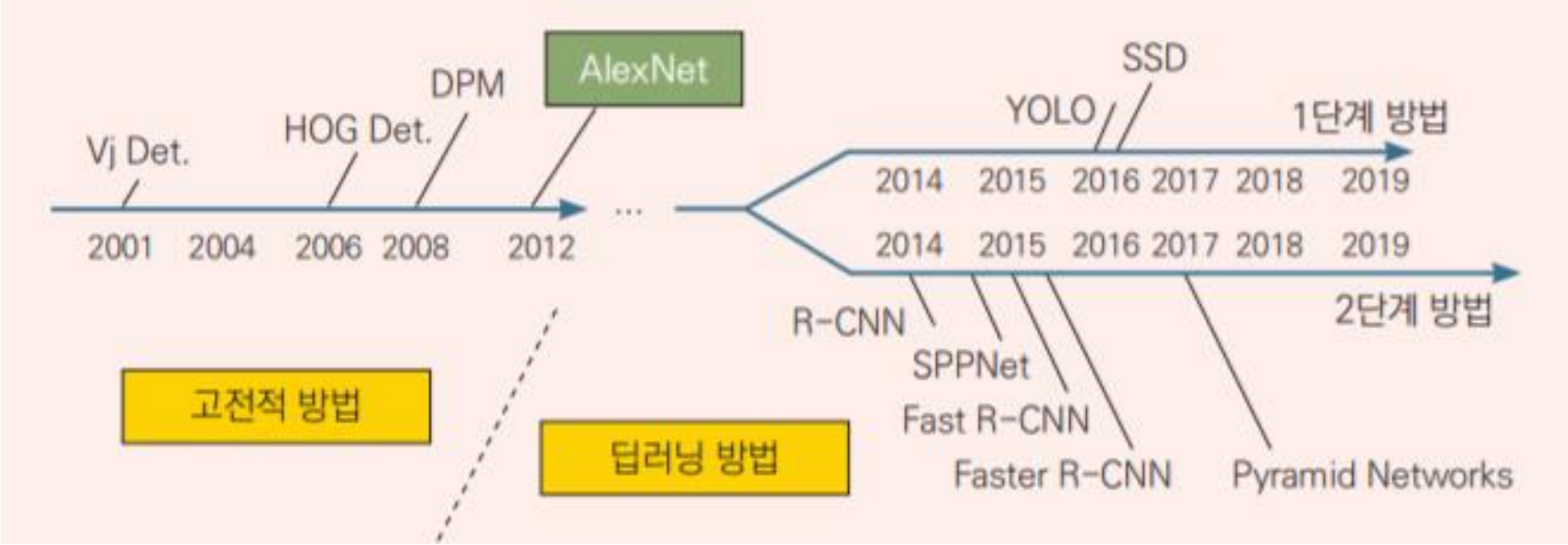
    → 사례 분할(instance segmentation)
    : 화면을 픽셀단위로 구분, 비슷한 것들끼리 모아서 객체를 형태대로 분할·식별

객체 검출 데이터셋

- 이미지넷(ImageNet) 데이터베이스
  - 200개 객체 검출할 수 있는 박스 있는 이미지 데이터
    → 1개 이미지 당 1.1개 객체 밖에 없는 제약
- PASCAL VOC 데이터베이스
  - 분류 범주 20개, 이미지 당 2.4개의 객체
    → 객체 검출에서 이미지넷보다 유용
- 마이크로소프트(MS) COCO 데이터베이스
  - 80개 이상의 분류와 12만개 이상 이미지와 90만개 객체로 구성

객체 검출 방법

- 2012년 AlexNet 이전 : 특성맵을 구하거나 템플릿 이용
  → 2012년 이후 합성곱 신경망 기반 모형이 이용



출처 : Zou et al.(2019).

분류와 위치 파악

- 분류 : 객체 구분, 각각 확률을 구하는 것
- 지역화(localization) : 객체의 경계상자(bounding box) 찾는 것
  → 경계 상자 : 중심점(b_x, b_y), 상자의 너비 b_w와 높이 b_h

분류와 위치 파악

- 합성곱 신경망에서 객체분류와 지역화
  - 객체 분류 : 마지막 층에 소프트맥스 함수를 적용
  - 경계상자 4개 데이터 → 회귀 통해 구함
    → 경계상자 거리 기반 손실함수를 최소화, 많은 계산이 필요

분류와 위치 파악

- 객체분류 : AlexNet이나 VGGNet의 이미지 식별과 같은 방식으로 진행
  → 경계상자 : 마지막 합성곱층에 2개의 완전연결층과 회귀층을 추가한 신경망으로 구함

지역 제안

- 이동창 객체 검출(sliding window object detection) :
  여러 크기의 창을 이동하면서 이미지의 객체 영역을 찾는 것
- 영역제안(Region Proposal) :
  비슷한 색이나 비슷한 문양들을 추려내서 경계상자를 찾는 방식
  - 경계상자가 객체 영역이 될 가능성이 높다고 판단
    → 해당 경계상자에 합성곱 신경망을 적용

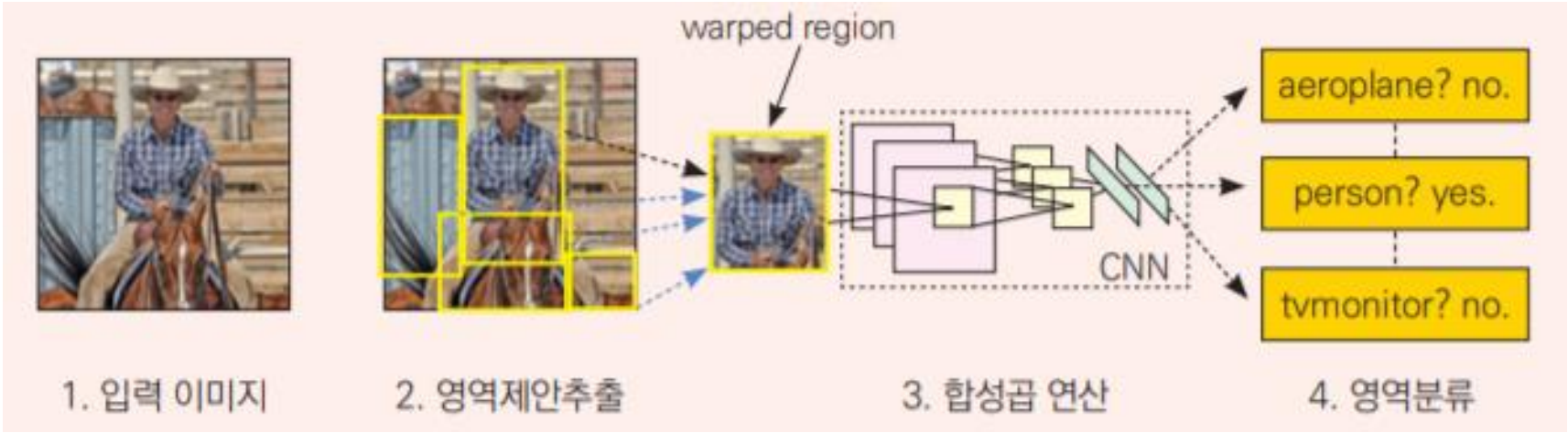
R-CNN

- 영역제안을 바탕으로 합성곱신경망 적용
  - 다른 크기, 위치의 제안된 영역 선택적 검색 : 2,000개
  - 선택적 영역에 VGGNet, GoogLeNet, ResNet 등 적용, pool5의 특성값 저장

    → 이 값 바탕으로 영역별 SVM으로 분류
- 분류 후 제안 상자가 정답 상자와 가까운 지 평가, 그 정확도가 증가하도록 보정

R-CNN

- R-CNN : 선택적 검색을 통해 구한 영역 + CNN 적용
  → 그 결과에 회귀와 SVM 동시 진행
  → 중간 저장과정, 종단 학습할 수 없음



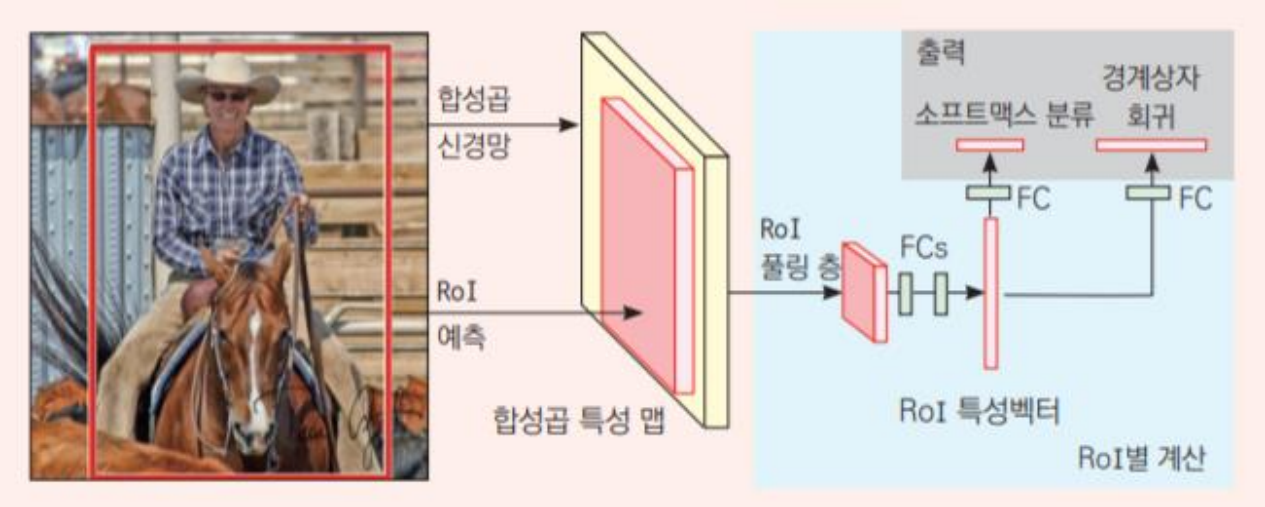
출처 : Girshick et al. (2014)

Fast R-CNN

- R-CNN과 구조에는 큰 차이는 없지만 속도가 빠름
  - R-CNN : 모든 영역에 제안마다 합성곱 신경망 적용
  - Fast R-CNN : 입력 이미지에 합성곱 신경망 우선 적용

    → 관심 영역(RoI)에 맞춰 풀링으로 영역 추출

    → 완전연결망 적용하여 객체 분류, 영역 찾음



출처 : Girshick (2015)

Fast R-CNN

- Fast R-CNN의 학습 소요 시간 : R-CNN보다 8.8배 빨랐고, 시험 데이터에서는 약 146배 빠르게 계산
  → Fast R-CNN은 종단 학습이지만, 실생활에 쓸 수 있을 만큼 속도가 빠르지 못함

Faster R-CNN

- Fast R-CNN과 같은 구조, 영역 제안시 RPN(Region Proposal Network)이라는 작은 합성곱 신경망 추가
  - 특정 위치 앵커(상자)들을 9개 정도 정하고, 위치별 분류 확률을 구하고 이를 모아서 최종 분류
  - Faster R-CNN은 시험 이미지 0.2초 만 식별
    → Fast R-CNN보다 11.5배 빠른 결과

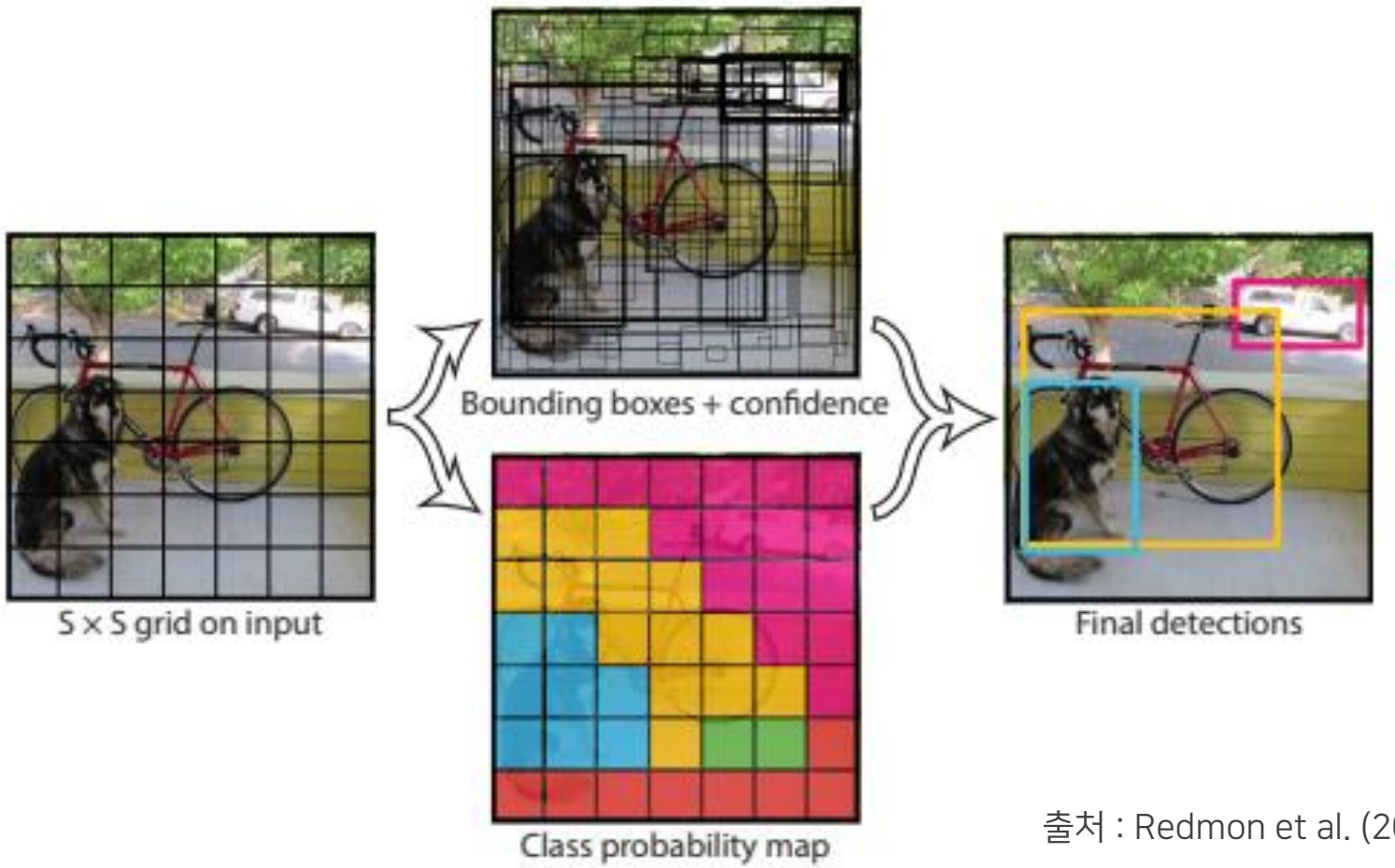
Yolo

- Yolo(You Only Look Once) : 분류와 영역 찾기를 동일한 합성곱 신경망으로 수행
  - 이미지 분할, 이에 대해 경계상자를 작성
    → 객체가 포함될 확률과 IoU를 곱해서 신뢰값을 구함
    → 어떤 객체인지 분류

출처 : Redmon et al. (2015)

한국방송통신대학교

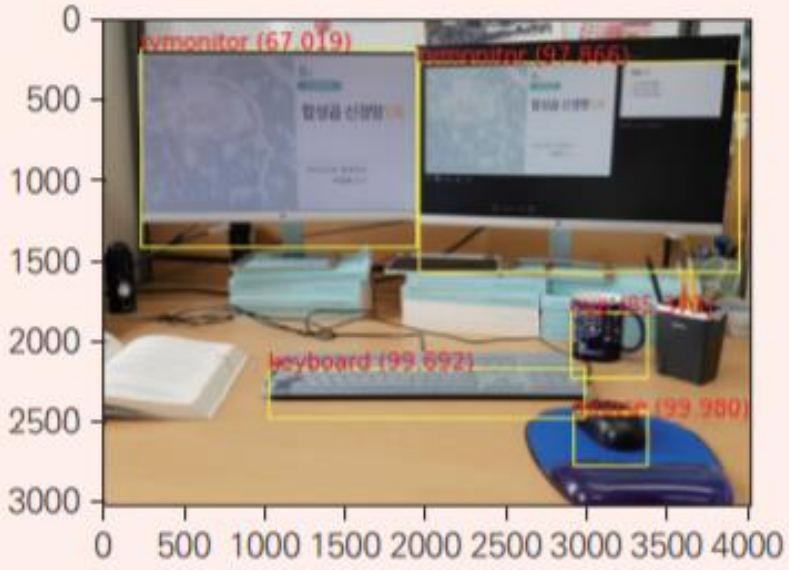
Yolo



출처 : Redmon et al. (2015)

Yolo

tvmonitor 67.01919436454773
tvmonitor 97.86608815193176
keyboard 99.6916651725769
cup 85.3725016117096
mouse 99.97950196266174



한국방송통신대학교

Yolo

- Yolo는 Yolo2, Yolo3, Yolo4로 발전
  - Yolo는 R-CNN 계열보다 분류 성능 자체는 조금 낮지만 반응 속도가 빨랐음
  - Yolo3 이후 버전 성능을 보면 Yolo보다 반응 속도가 매우 빠르고, 분류 성능도 높은 것으로 나타났음

Yolo

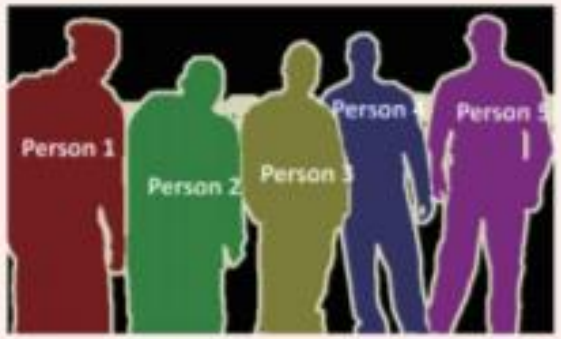
출처 : https://www.youtube.com/watch?v=Df0Tu6IOE44

영상분할

- 이미지의 모든 픽셀에 레이블을 할당, 배경과 객체 분리
  → 의미 분할(semantic segmentation)
  → 인스턴스 분할(instance sgmentation)

객체검출

의미분할



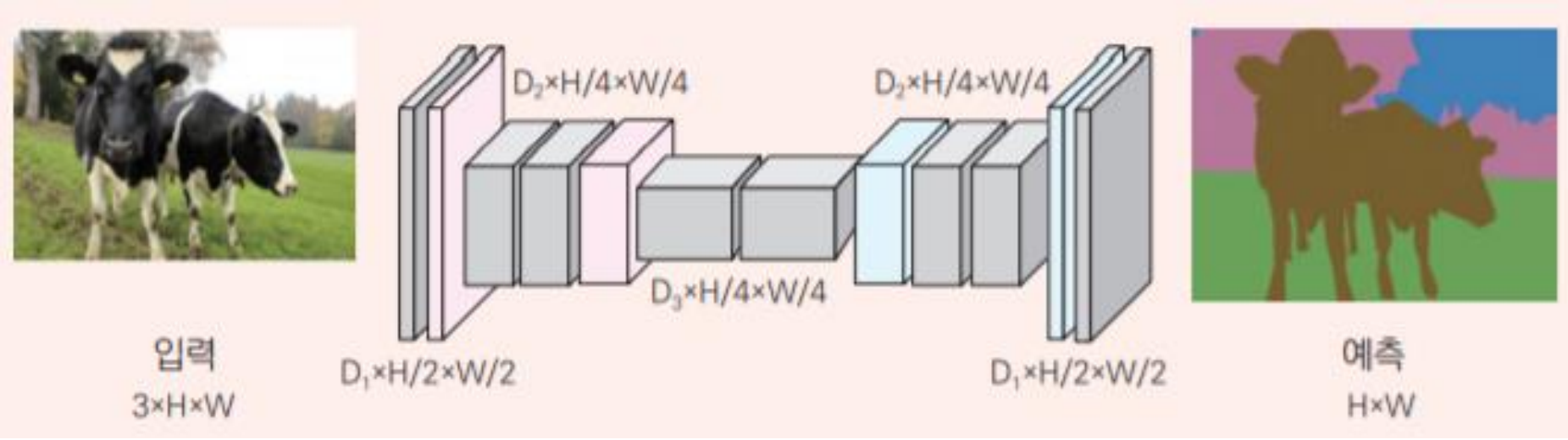
인스턴스 분할

출처 : http://kaiminghe.com/iccv17tutorial/maskrcnn_iccv2017_tutorial_kaiminghe.pdf

의미 분할

- 완전 합성곱 신경망(Fully Convolutional Network, FCN)

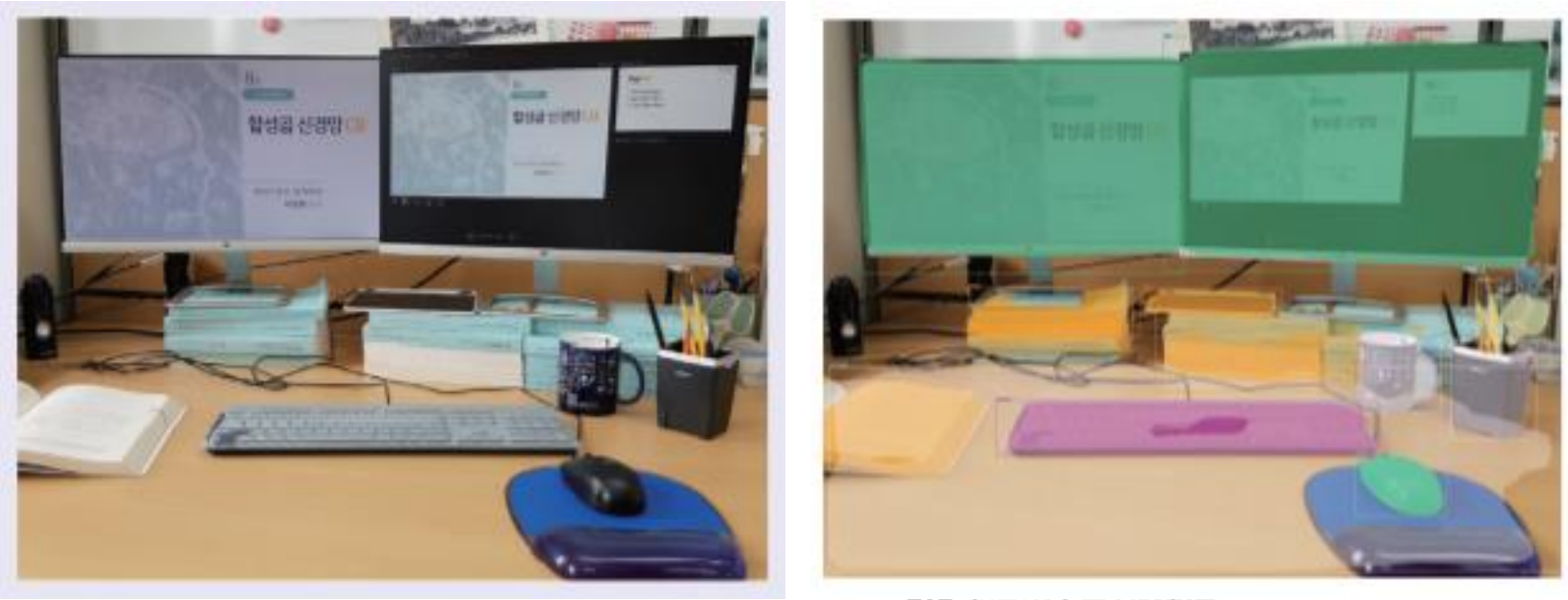
  → 완전 합성곱 연결망으로 구성된 인코더와 디코더를 이용



출처 : http://cs231n.Stanford.edu

Mask R-CNN

- 인스턴스 분할의 대표적 방법
  → Faster R-CNN에 완전 합성곱 신경망과 관심영역(RoI)을 추가하여 Mask를 예측



얼굴 인식

- 얼굴 인식

  : 얼굴 검증(Face Verification)과 얼굴식별(Face Identification)로 구분

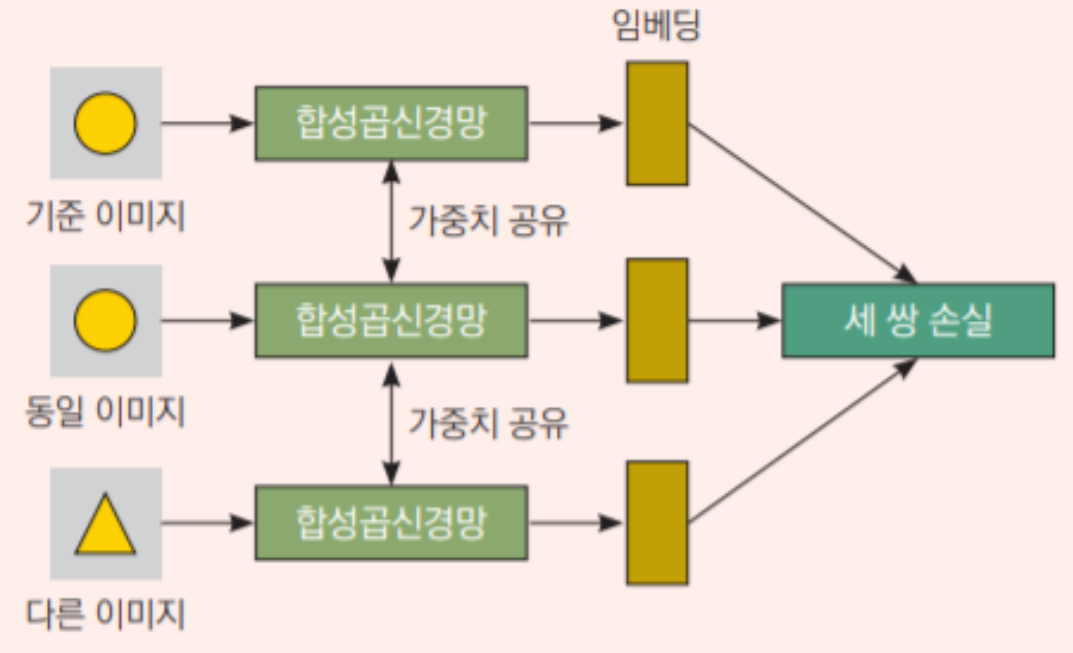
얼굴 인식

- 얼굴인식에서 딥러닝을 적용할 때 데이터 사전처리가 필수적
  - 다양한 각도에서 찍은 사진의 사전조정이 필요
- 두 이미지가 같은 사람인지 파악
  - 두 이미지의 거리 : 합성곱 신경망의 특성맵을 비교해서 구함
    → 거리가 매우 작다면 두 사람은 같은 사람,
    아니면 다른 사람

얼굴 식별

- 얼굴을 식별하는 신경망 : 샴 네트워크(Siamese Network)

  → 합성곱 신경망을 적용한 두 이미지의 인코더를 기반으로 유사도를 비교



얼굴 식별

- 손실함수로는 세 쌍 손실(Triplet Loss) 함수가 이용
  - 기준 이미지(anchor image) : A, 같은사람 : P, 다른사람 : N
  - 손실함수 $L(A, P, N)$:
    - ✓ 이들 사이의 거리 기반
    - ✓ 같은 사람은 더 가깝게, 다른 사람은 더 멀어지게 하는 함수
    - ✓ α : 마진 역할

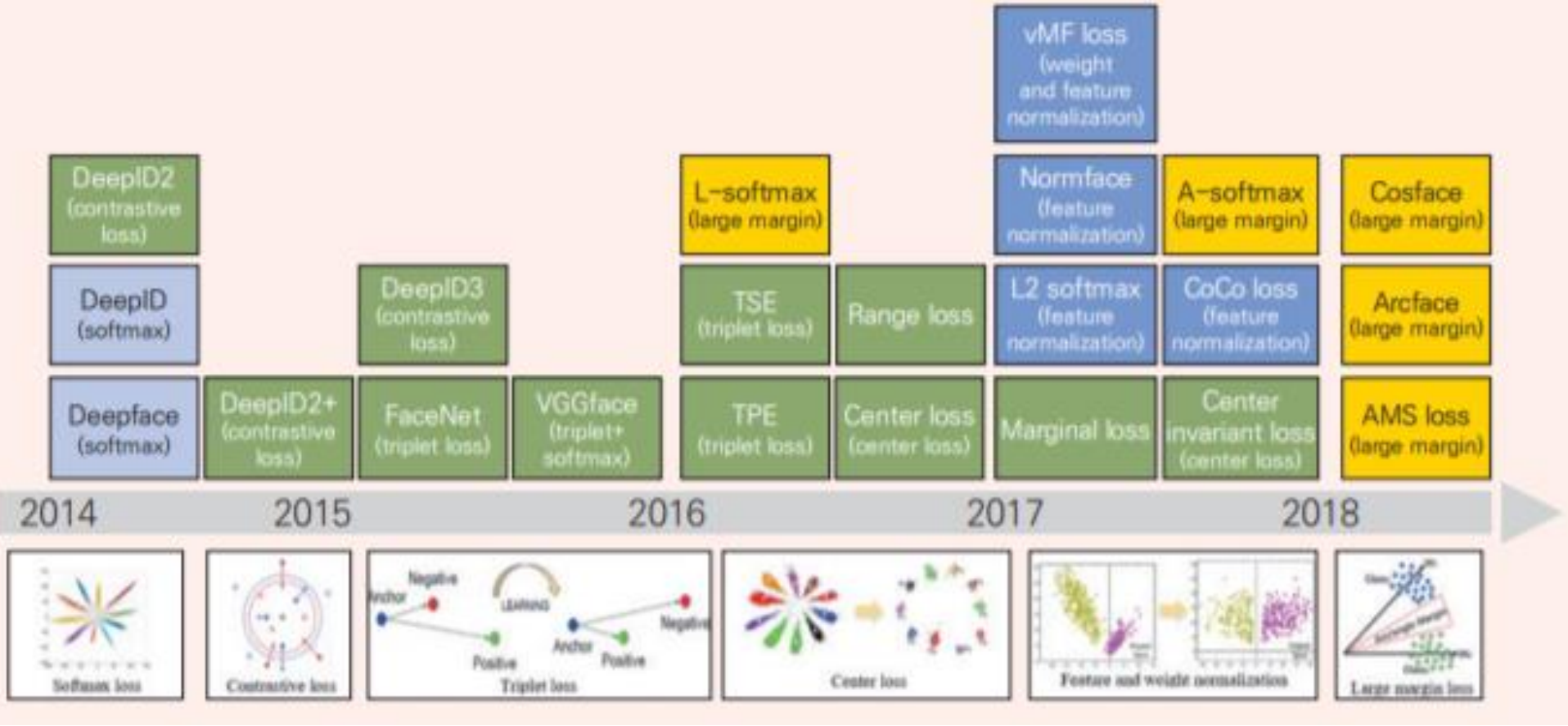
$$L(A, P, N) = ||f(A) - f(P)||^2 - ||f(A) - f(N)||^2 + \alpha$$

얼굴 검증

- 얼굴 검증 : 합성곱 신경망 적용 → 완전연결망에 시그모이드 함수를 적용하여 같은지 여부 이항 분류

$$\hat{y} = \alpha\left(\sum \left\| f\left(x_i^{(1)}\right) - f\left(x_i^{(2)}\right)\right\|^2\right)$$

딥러닝 기반 얼굴인식 방법의 발전



출처 : Wang and Deng (2019)

한국방송통신대학교

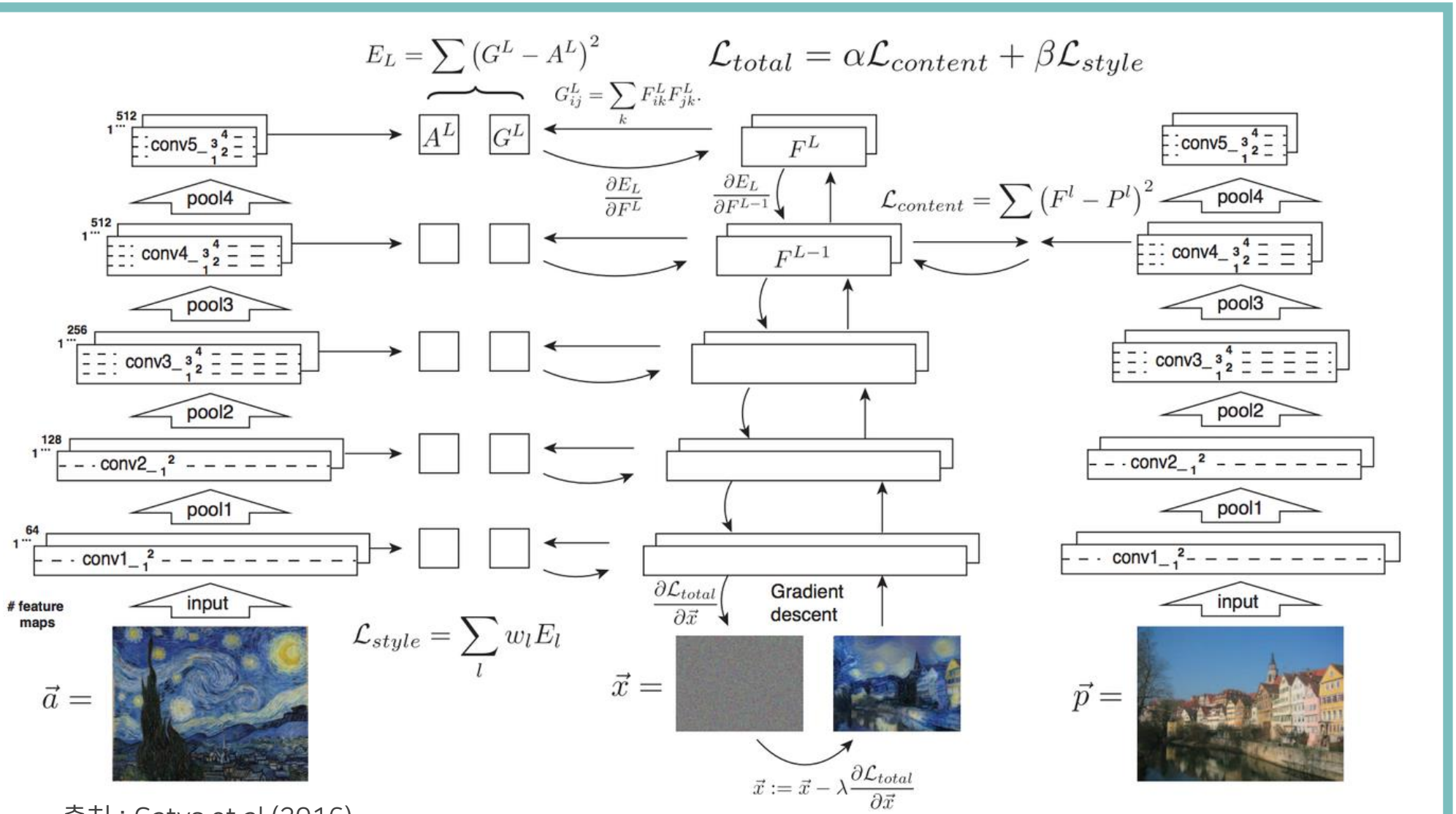
스타일 변환과 합성곱 신경망

- 합성곱 신경망은 층 별 특성을 추출해내는 역할이 다름
  - 첫 번째 층 : 아주 기초적인 것들을 볼 수 있고 층이 깊어질수록 복잡한 모양
  - 합성곱 신경망을 이용하여 이미지의 내용 부분과 스타일 부분을 구분하여 결합 → 이미지의 스타일을 전환

스타일 변환

- 스타일 이미지 S, 가지고 있는 사진 C
  - S로부터는 특유의 스타일을, 가지고 있는 사진 C에서는 내용을 보존해서 새로운 이미지 G를 생성
  - 손실함수 : L

스타일 변환



출처 : Gatys et al.(2016)

스타일 변환

- 임의의 그림 G가 손실함수를 최소화 방향으로 오차역전파법을 이용하여 가중치를 갱신 → 이미지 각각의 픽셀을 갱신
  - 스타일 : 채널 간 상관관계 즉, 두 값을 곱해서 더한 공분산의 형태로 표현
  - 스타일 부분은 전 층에 대해 진행, 콘텐츠 부분은 특정 층의 정보를 이용하여 손실함수를 정함

스타일 변환

스타일 변환

한국방송통신대학교

학습정리

- ✓ 딥러닝 모형은 직접 학습하여 구축하는 것보다 학습된 모형의 결과를 바탕으로 구축하는 것이 보다 효율적인데 이를 전이학습이라 한다.
- ✓ 객체 검출은 이미지에서 객체의 분류와 위치파악을 동시에 진행하는 것으로 R-CNN, Yolo 등이 있다.

학습정리

- ✓ 얼굴인식은 딥러닝 기반으로 이루어지고 있으며 얼굴검증과 얼굴식별이 있다.
- ✓ 이미지 스타일 변환은 딥러닝의 특성을 이용하여 콘텐츠의 특성을 보존하면서 스타일의 특성을 바꾸는 것이다.

딥러닝의 통계적이해

다음시간안내

9강. 오토인코더와 GAN(1)