

시각지능 A – Object Detection

Object Detection 기초

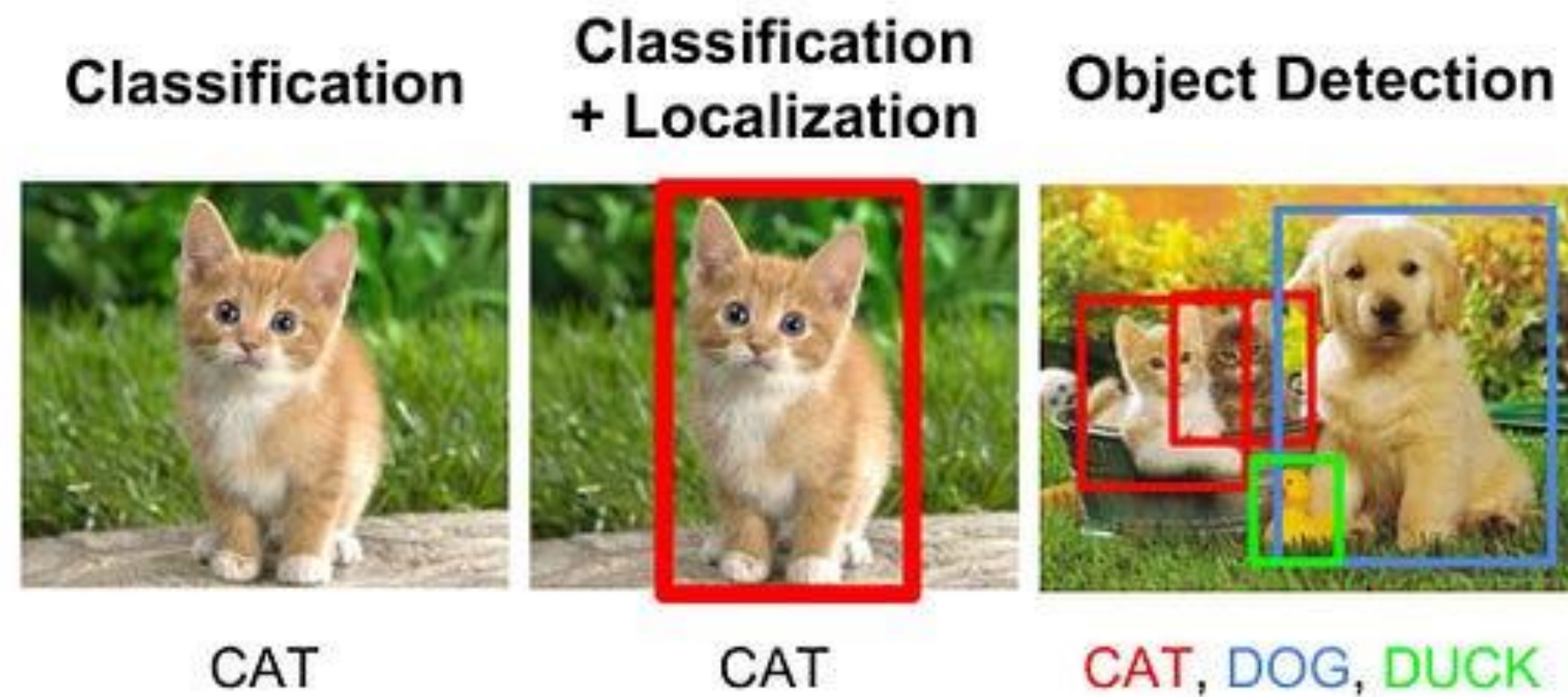
목차 객체 탐지 기초

1. 객체 탐지 개요
2. Image Localization
3. 객체 탐지 고전 알고리즘
4. 평가 지표

1. 객체 탐지 개요

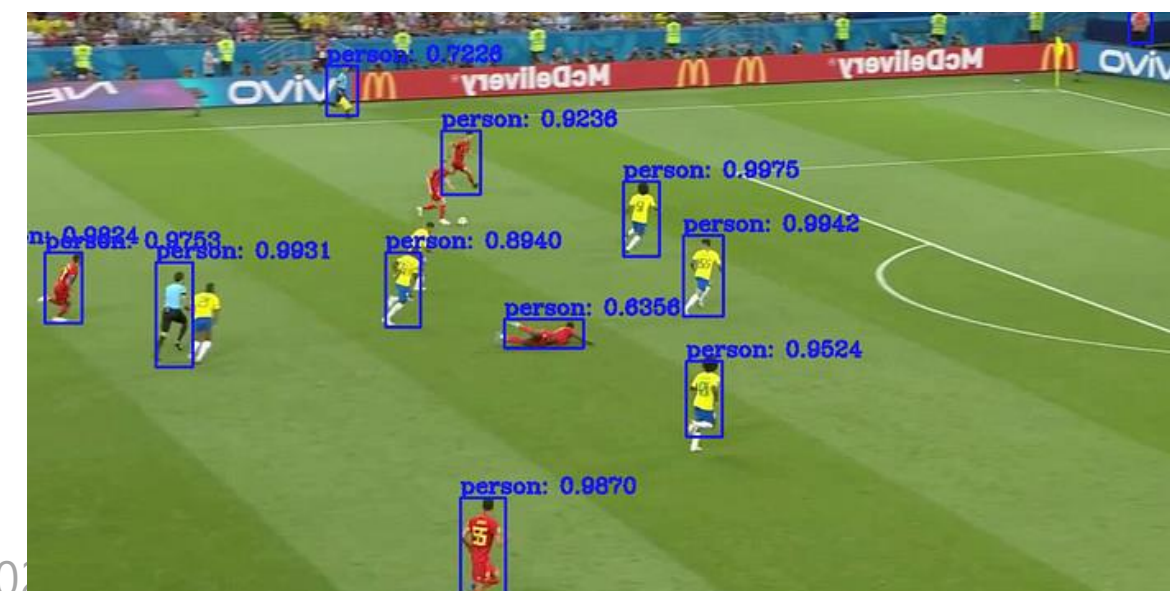
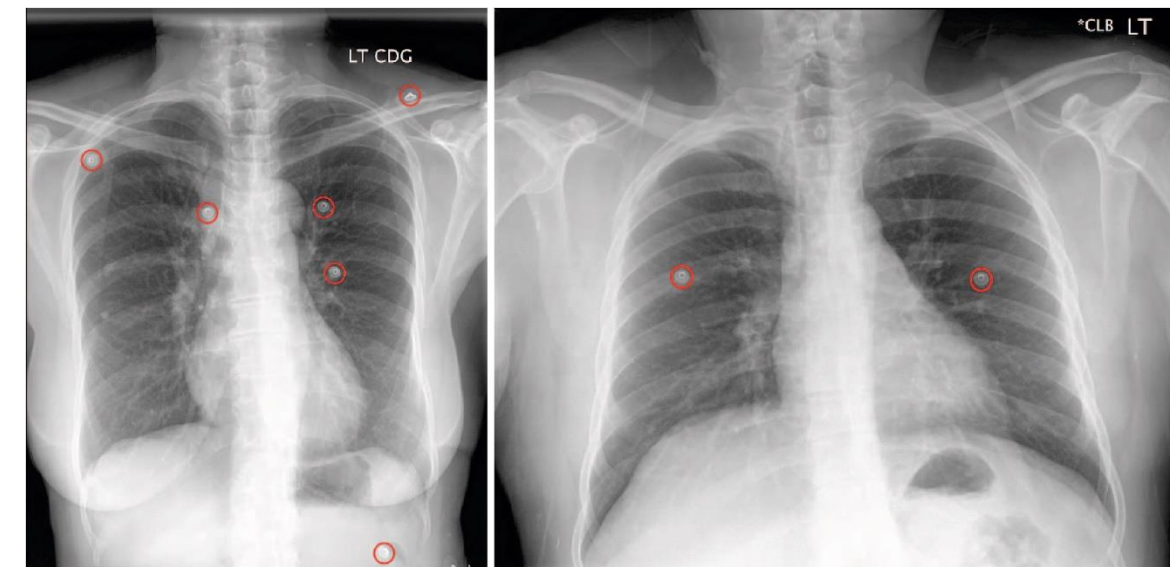
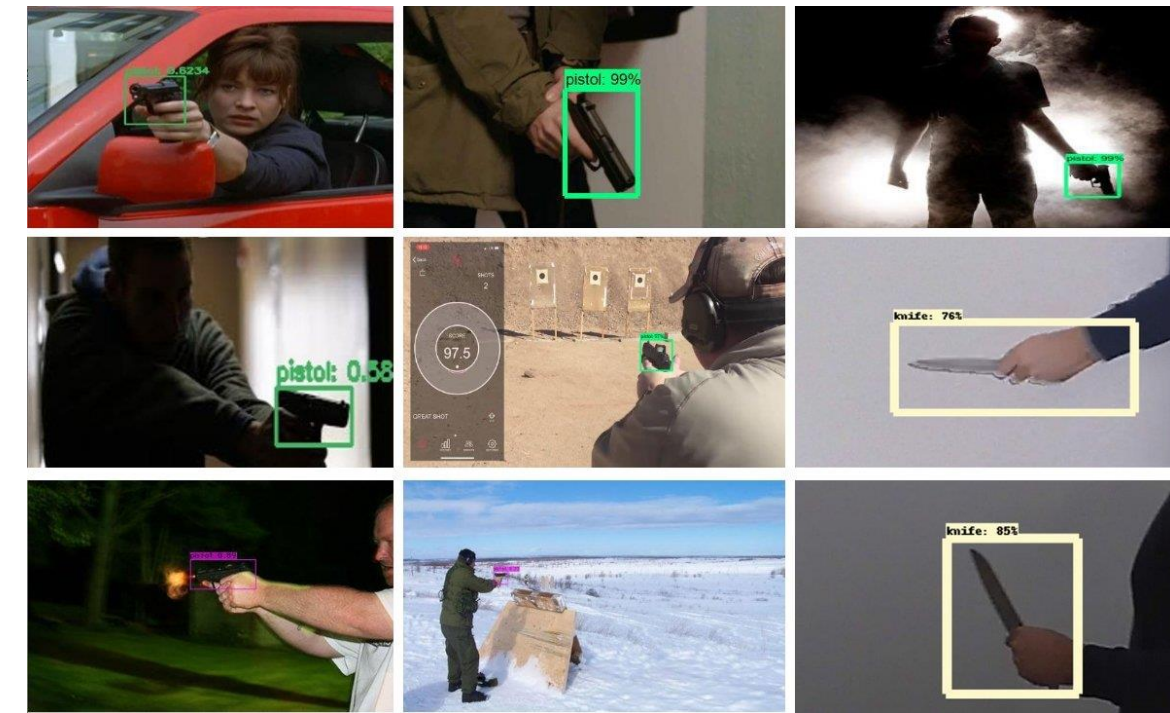
Object Detection

- Object Detection(객체 탐지)
 - 이미지나 영상에서 특정 클래스에 속하는 여러 객체를 인식하고
 - 그 객체들의 위치를 찾아내는 기술
 - 객체 분류(Classification): 대상 객체의 범주가 어느 것인지를 예측
 - 객체 위치 인식(Localization): 대상 객체의 위치와 이에 대한 신뢰도(Confidence)를 반환하는 회귀 문제



각 산업 분야에서 쓰이는 객체 탐지 기술

- 보안 시스템
 - CCTV, 인공위성 영상을 분석하여 대상의 위치와 신원을 추적
 - 또는 대상 간의 행위(범죄, 위협, 침입 등)를 분석하여 알림
- 질병 진단
 - 종양, 용종, 결핵, 기흉 등 병리 진단
- 멸종위기 생물 탐지 및 추적
- 소행성, 천체 관측 및 행성 탐지 및 기상 관측
- 스포츠 중계



객체 탐지 사용례

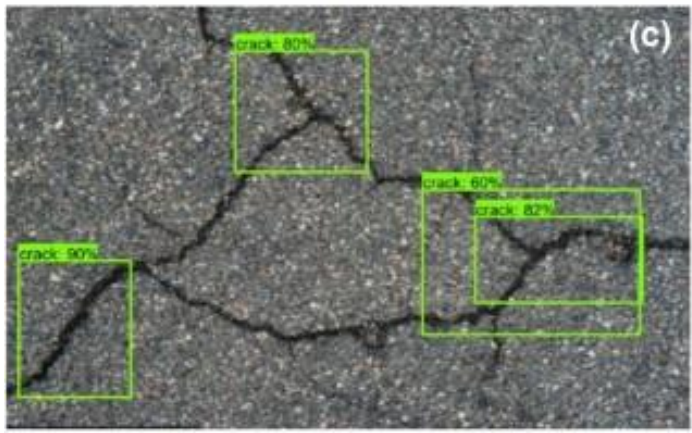
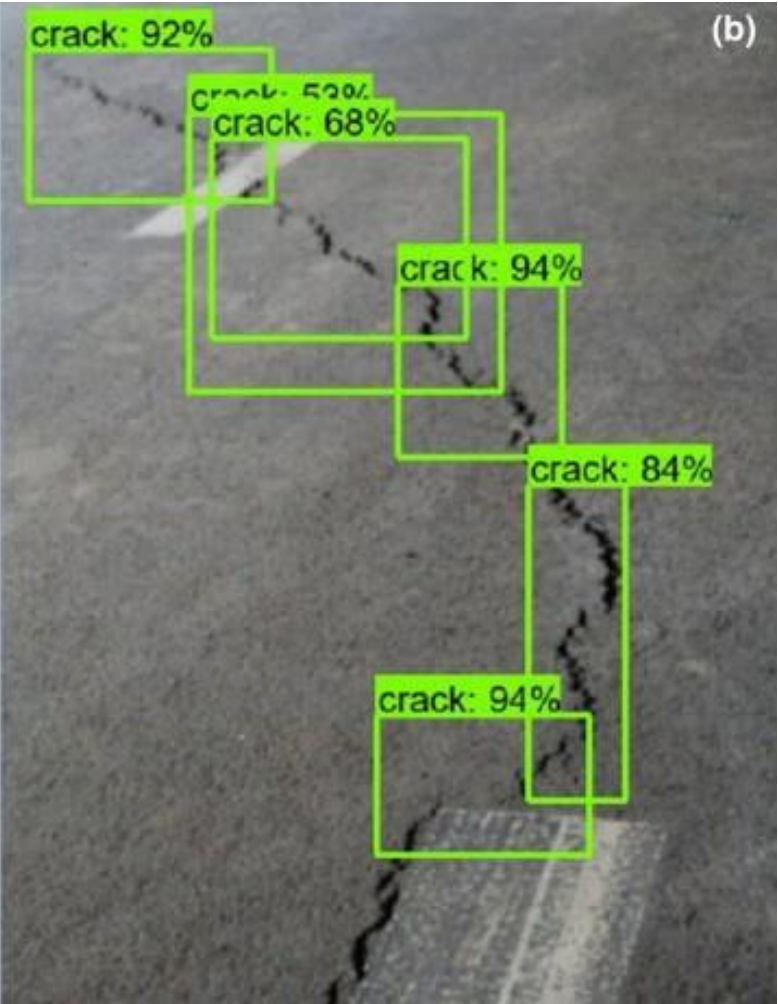
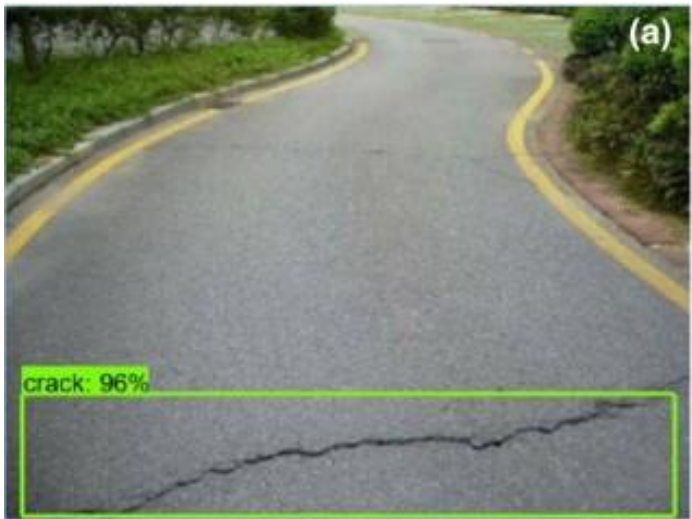
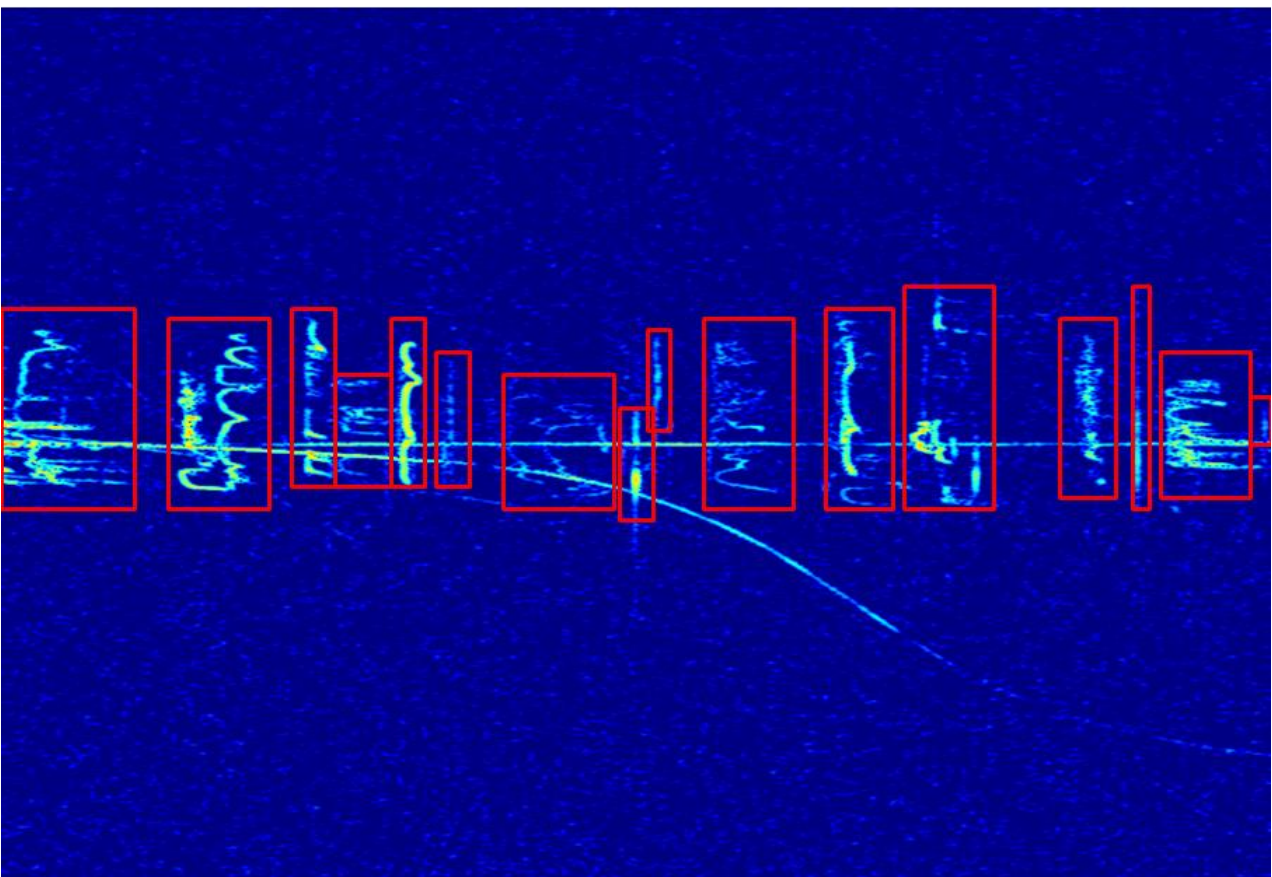
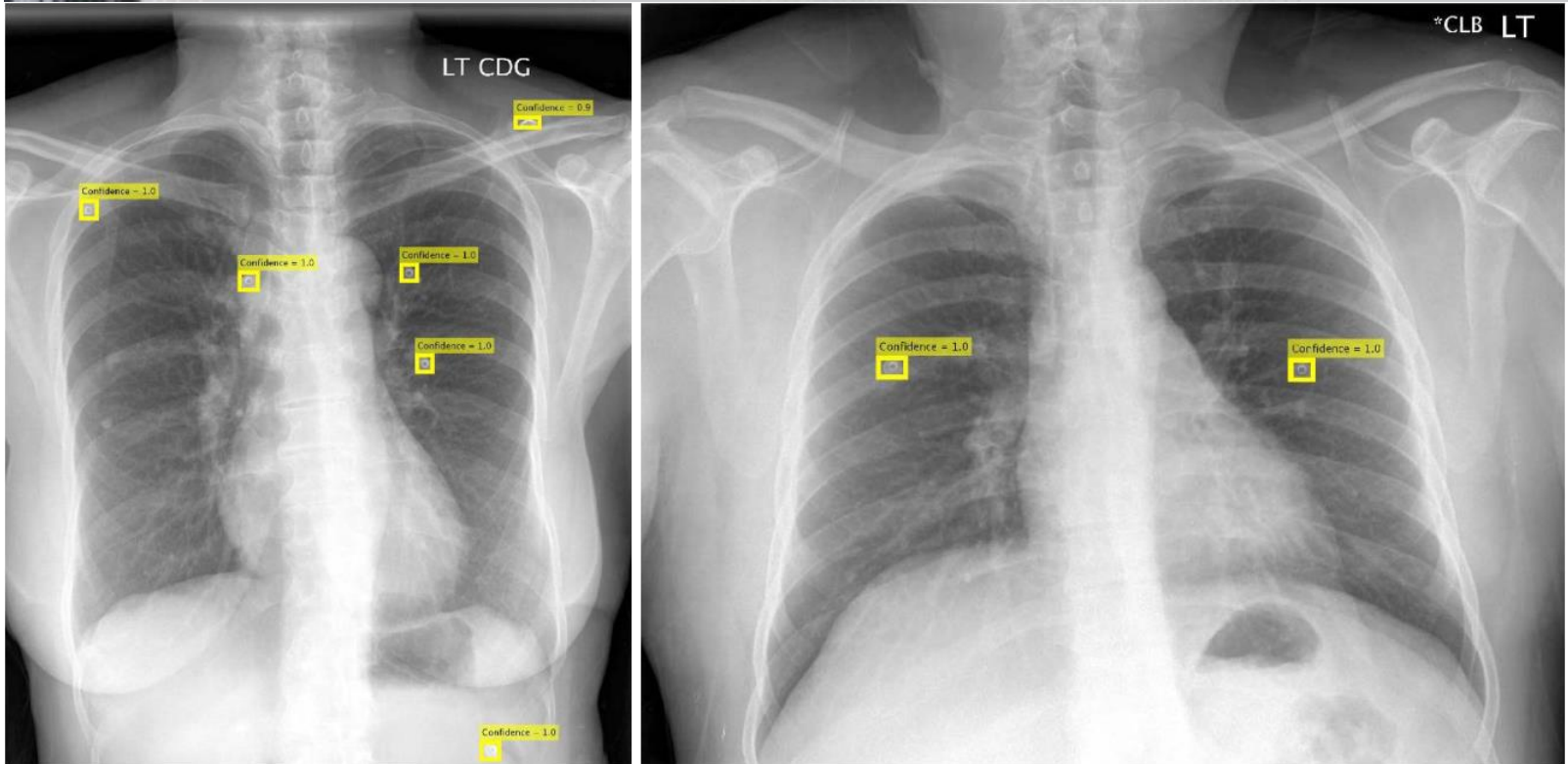
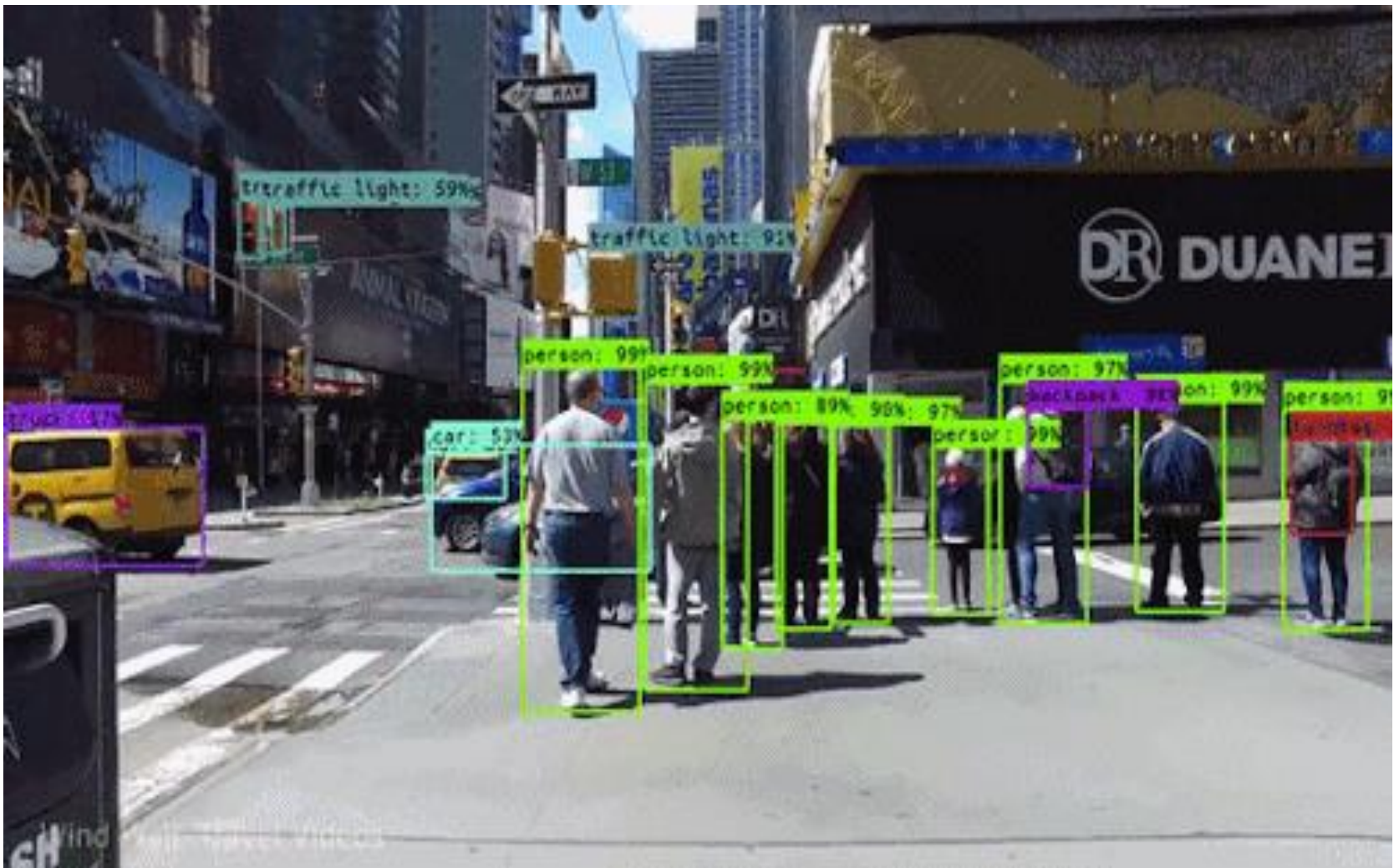
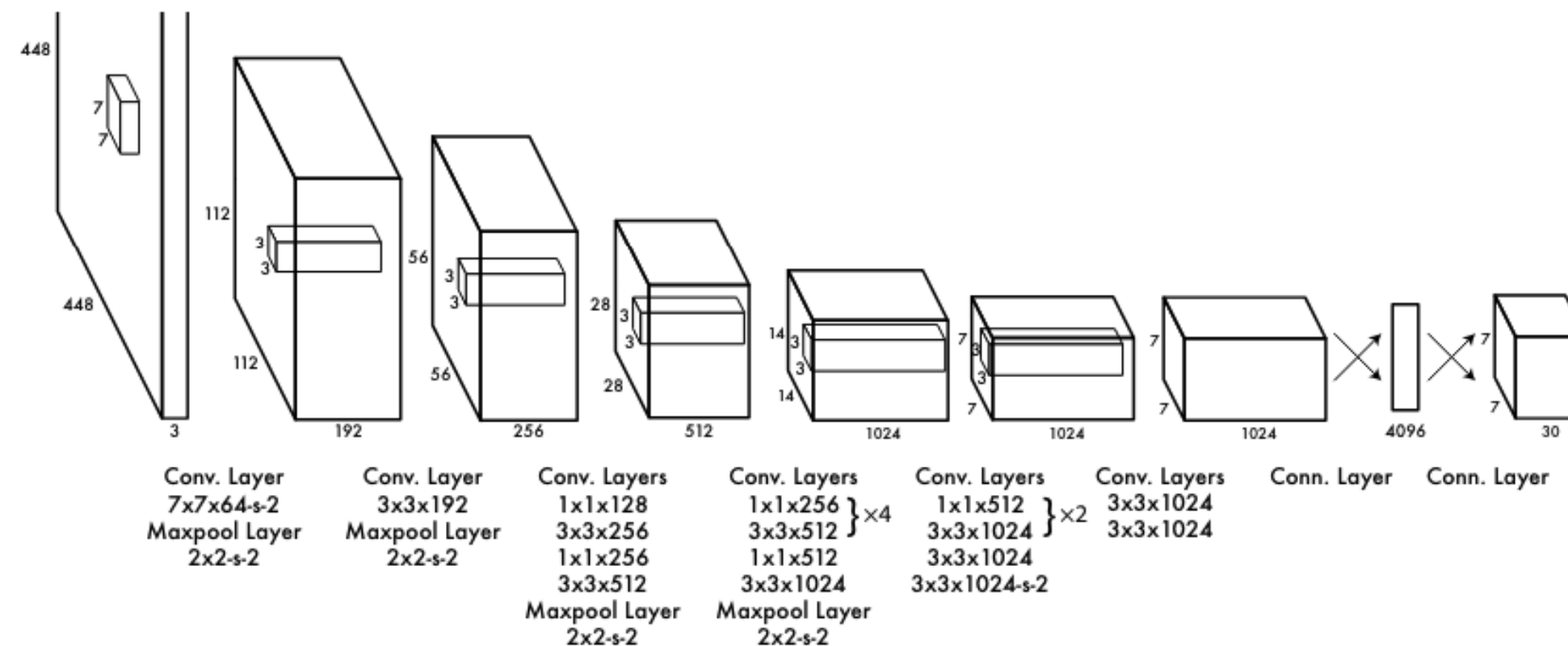


Fig. 2. Automatic crack detection (with confidence score) using the faster R-CNN (cf. Fig. 1).

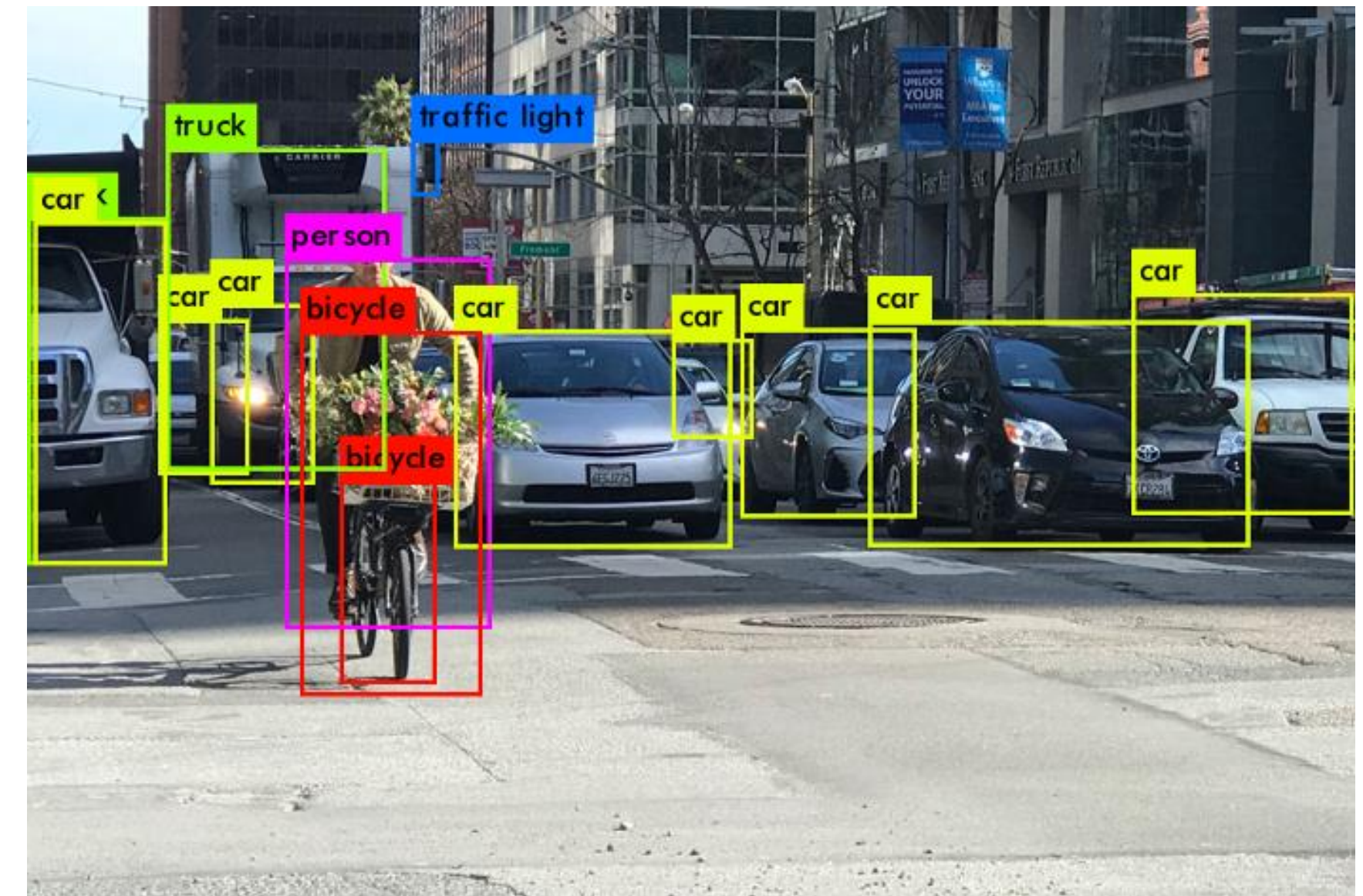
객체 탐지 개요

- Object detection task가 어려운 이유
 - 2가지 문제를 동시에 해결해야 함
 - 객체 위치 추정 + 객체 분류
 - Multi-output 구조
 - 시간이 오래 걸리며 두 가지 문제를 풀기 위한 두 가지 손실 함수 필요
 - 지도학습을 위한 데이터셋 또한 구축하기에 노동력이 많이 요구



객체 탐지 개요

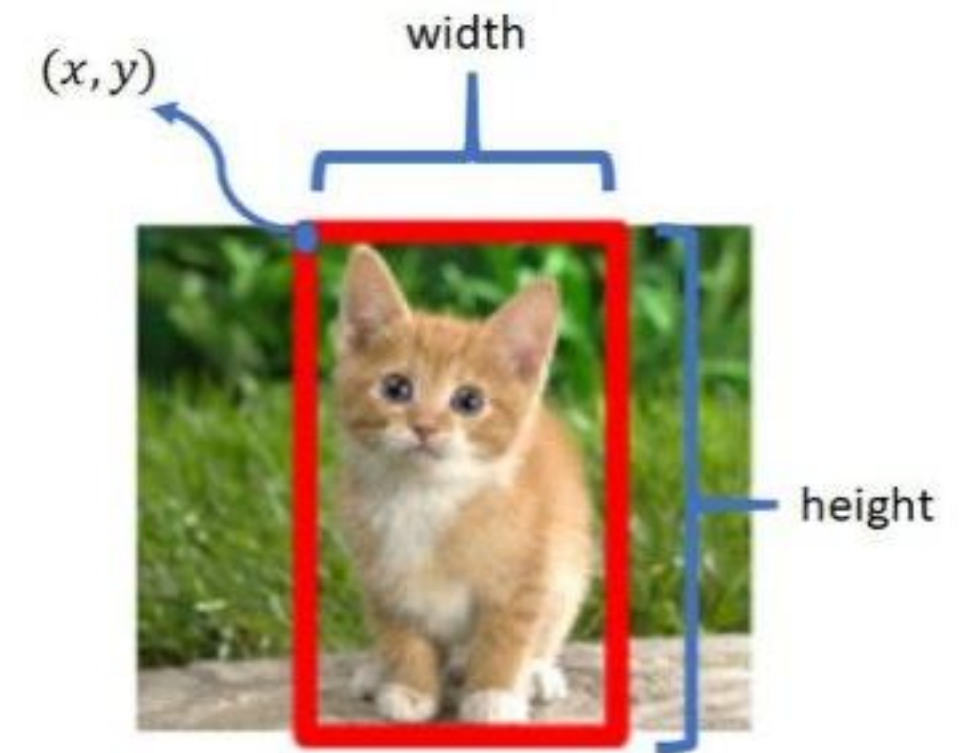
- Object Detection task가 어려운 이유
 - 이미지 분류에는 인식할 객체가 적음
 - 기존 분류 문제는 이미지 당 하나의 객체를 분류함
 - 또한 객체가 이미지 중앙에 위치하고, 이미지에서 큰 부분을 차지함
 - OD에서는 다중 객체를 인지해야 함
 - 한 이미지에서 많은 수의 객체를 인지
 - 객체의 종류가 매우 다양
 - 또한 크기와 위치도 제각각



2. Image Localization

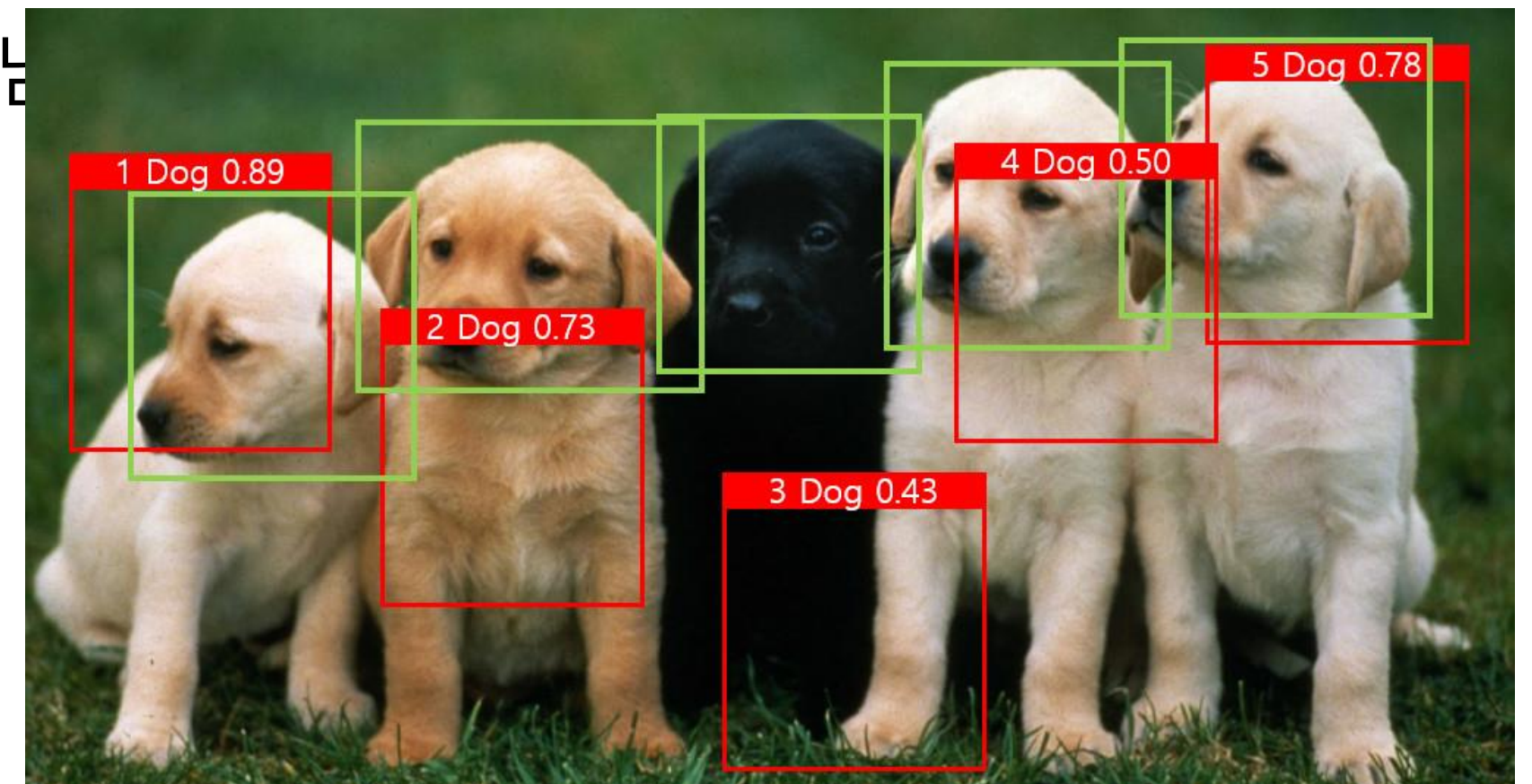
Image Localization

- Image Localization
 - 이미지에서 특정 객체를 인식하고, 그 위치를 추정하는 과정
 - 일반적으로 객체의 위치를 Bounding box로 표시
- 과정
 - 입력 데이터: 탐지하려는 이미지
 - 네트워크 출력: 이미지에서 객체가 있을 위치와 이에 대한 bbox
 - 손실: 두 bbox 사이의 불일치도



Bounding box

- Bounding box
 - 이미지나 영상 속 특정 객체를 감싸는 직사각형 영역
 - 모델이 추정한 객체의 위치와 범위, 신뢰도를 보여줌
 - 신뢰도(Confidence score): 해당 박스 위치에 객체가 존재하는지에 대한 확신 여부
 - 박스의 네 변이 이미지와 접할 때, 좋은 bbox라 함
 - 일반적으로 한 모델에서 특정 객체를 찾기 위하여 여러 개의 bbox를 생성
 - 이 중 잘못 예측한 대다수는 제거되고, 잘 예측된 값만 남
 - 크기가 맞지 않거나
 - 위치가 올바르지 않은 경우



Bounding box

- Bbox의 표현법
 - xyxy(KITTI format)
 - bbox의 좌상단 좌표(x_1, y_1)
 - 우하단 좌표(x_2, y_2)
 - xywh(COCO format)
 - bbox 좌상단/중심 좌표(x, y)
 - bbox 너비와 높이(w, h)
 - Normalized xywh(YOLO format)
 - xywh의 모든 값을 $[0, 1]$ 에서 표현
 - 이미지 크기와 무관하게 사용할 수 있음

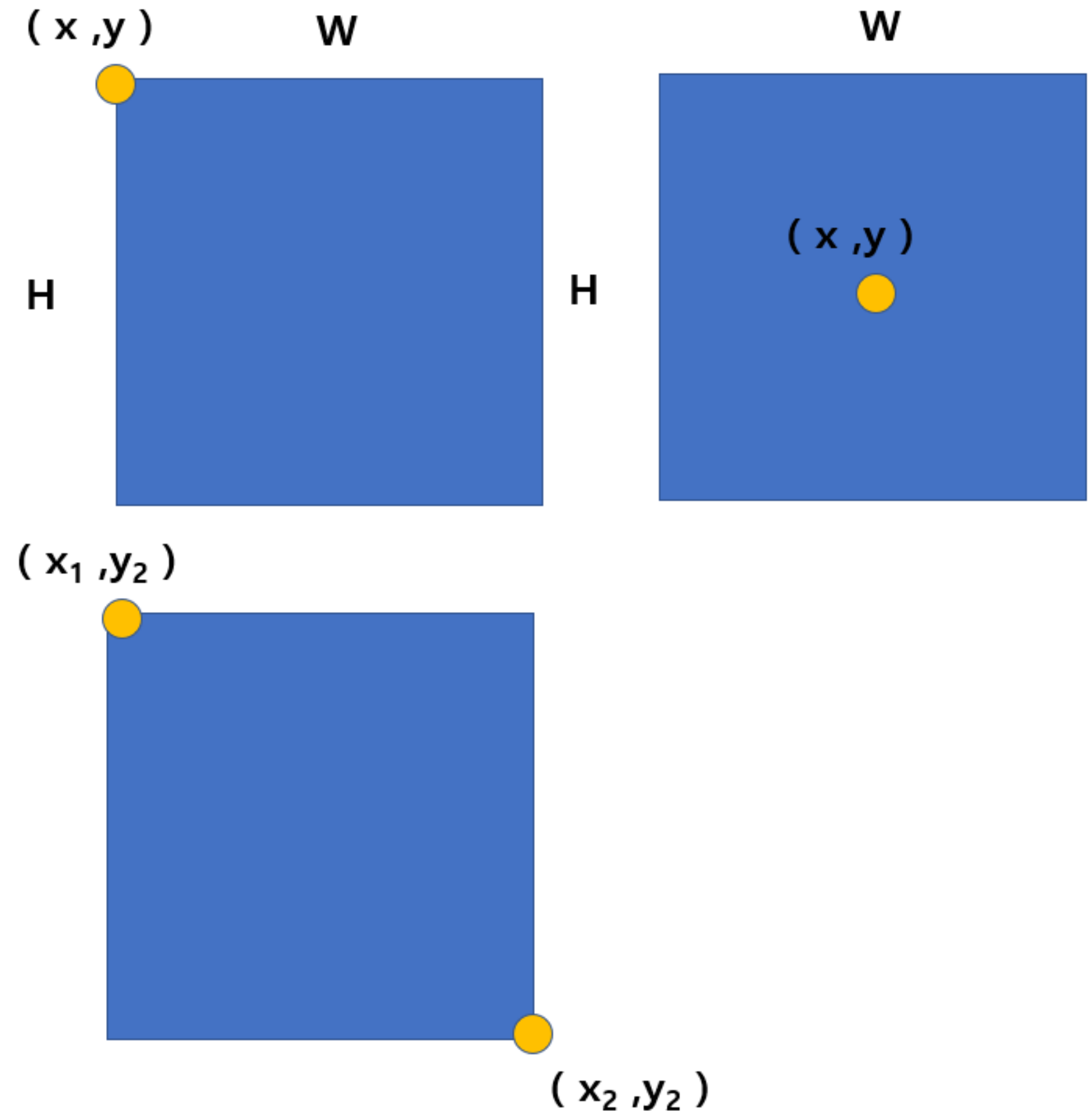


Image Localization

- 단일 Localization
 - x : 이미지(B, C, H, W)
 - y : bbox 좌표(x, y, w, h , Class probability)
- 다중 Localization
 - x : 이미지(B, C, H, W)
 - y_1 : bbox1 좌표(x, y, w, h , Confidence score, Class probability)
 - y_2 : bbox2 좌표(x, y, w, h , Confidence score, Class probability)
 - ...
- 다중 Localization task는 이미지 하나 당 서로 다른 수의 레이블이 붙음
 - 레이블 수 = 데이터 수

Detection Label

- Object detection 레이블은 주로 문서 파일로 제공됨
 - 이미지마다 검출되는 객체의 수가 다름:비정형 데이터
 - 그러나 추출하는 정보는 동일함: 정형 데이터
 - Bbox 좌표
 - Confidence score
 - 클래스 별 분류 확률
- 반정형 데이터를 사용하면 효율적인 표현이 가능
 - XML
 - JSON
 - YAML
 - csv, txt 등

Detection Label

- XML(Pascal VOC format)
 - Pascal VOC는 대표적인 Object detection 벤치마크 데이터셋
 - 이 데이터의 레이블은 XML 확장자로 작성됨

```
<annotation>
  <folder>images</folder>
  <filename>image1.jpg</filename>
  <object>
    <name>dog</name>
    <bndbox>
      <xmin>50</xmin>
      <ymin>100</ymin>
      <xmax>200</xmax>
      <ymax>300</ymax>
    </bndbox>
  </object>
</annotation>
```

Detection Label

- JSON(COCO format)
 - COCO dataset 또한 대표적인 Object detection 벤치마크 데이터셋
 - 파이썬 딕셔너리와 유사한 구조
 - LabelMe도 동일한 구조를 사용
 - 설정 파일 구성 시 자주 사용되는 포맷

```
{  
  "images": [{"id": 1, "file_name": "image1.jpg", "height": 600, "width": 800}],  
  "annotations": [{"image_id": 1, "category_id": 18, "bbox": [50, 100, 150, 200]},  
  "categories": [{"id": 18, "name": "dog"}]  
}
```


Detection Label

- YAML
 - JSON을 개선하여 만든 문서
 - 데이터셋 뿐만 아니라 모델 설정 등도 이에 저장

```
# 객체 탐지 레이블 예시

image: "path/to/image1.jpg" # 이미지 파일 경로

objects:
  - class: "dog" # 클래스 이름
    bbox: # 바운딩 박스 좌표
      xmin: 50 # 좌상단 x 좌표
      ymin: 100 # 좌상단 y 좌표
      xmax: 200 # 우하단 x 좌표
      ymax: 300 # 우하단 y 좌표

  - class: "cat" # 클래스 이름
    bbox: # 바운딩 박스 좌표
      xmin: 30
      ymin: 50
      xmax: 120
      ymax: 180

image_size:
  width: 800 # 이미지 너비
  height: 600 # 이미지 높이
```

```
# YOLO 모델 설정 파일 예시

# 입력 이미지 크기
input_size: 640

# 앵커 박스 (Anchor Boxes) 정의
anchors:
  - [10, 13, 16, 30, 33, 23]
  - [30, 61, 62, 45, 59, 119]
  - [116, 90, 156, 198, 373, 326]

# 백본 네트워크 (Backbone Network)
backbone:
  type: CSPDarknet53
  depth: 53
  width: 1.0

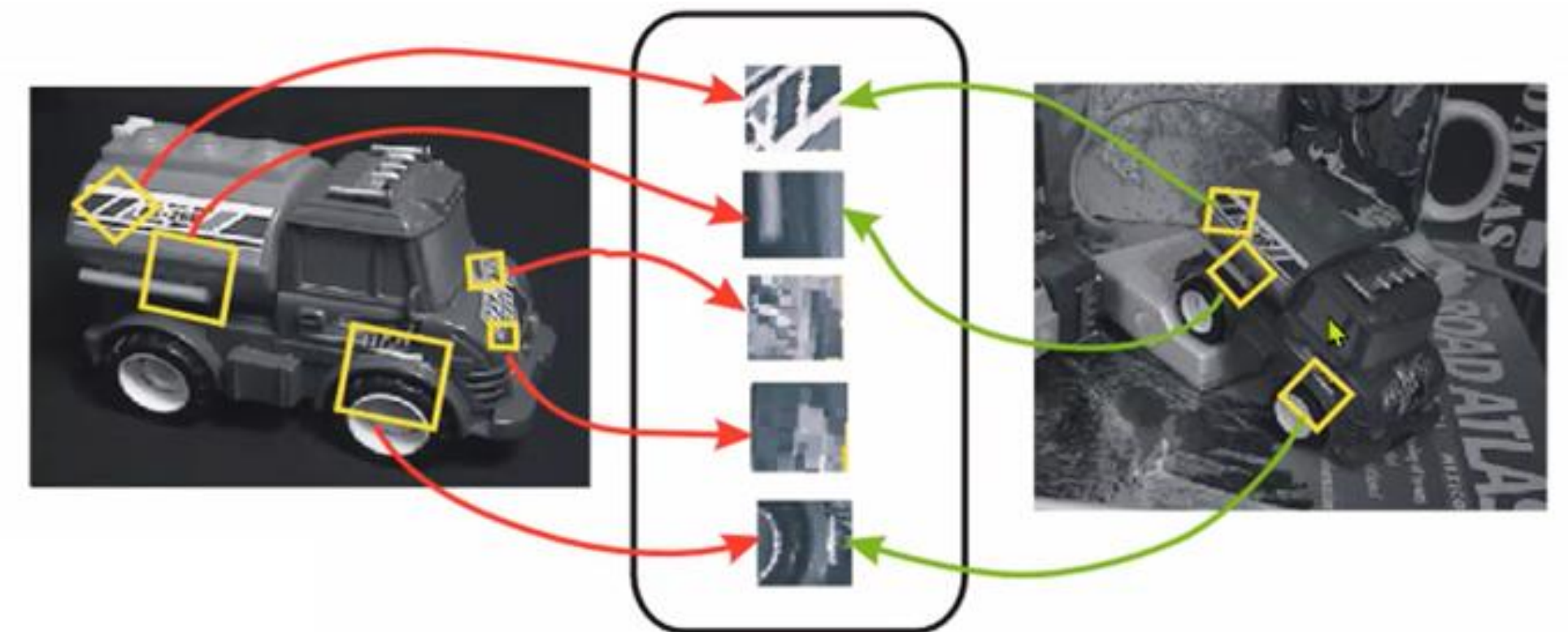
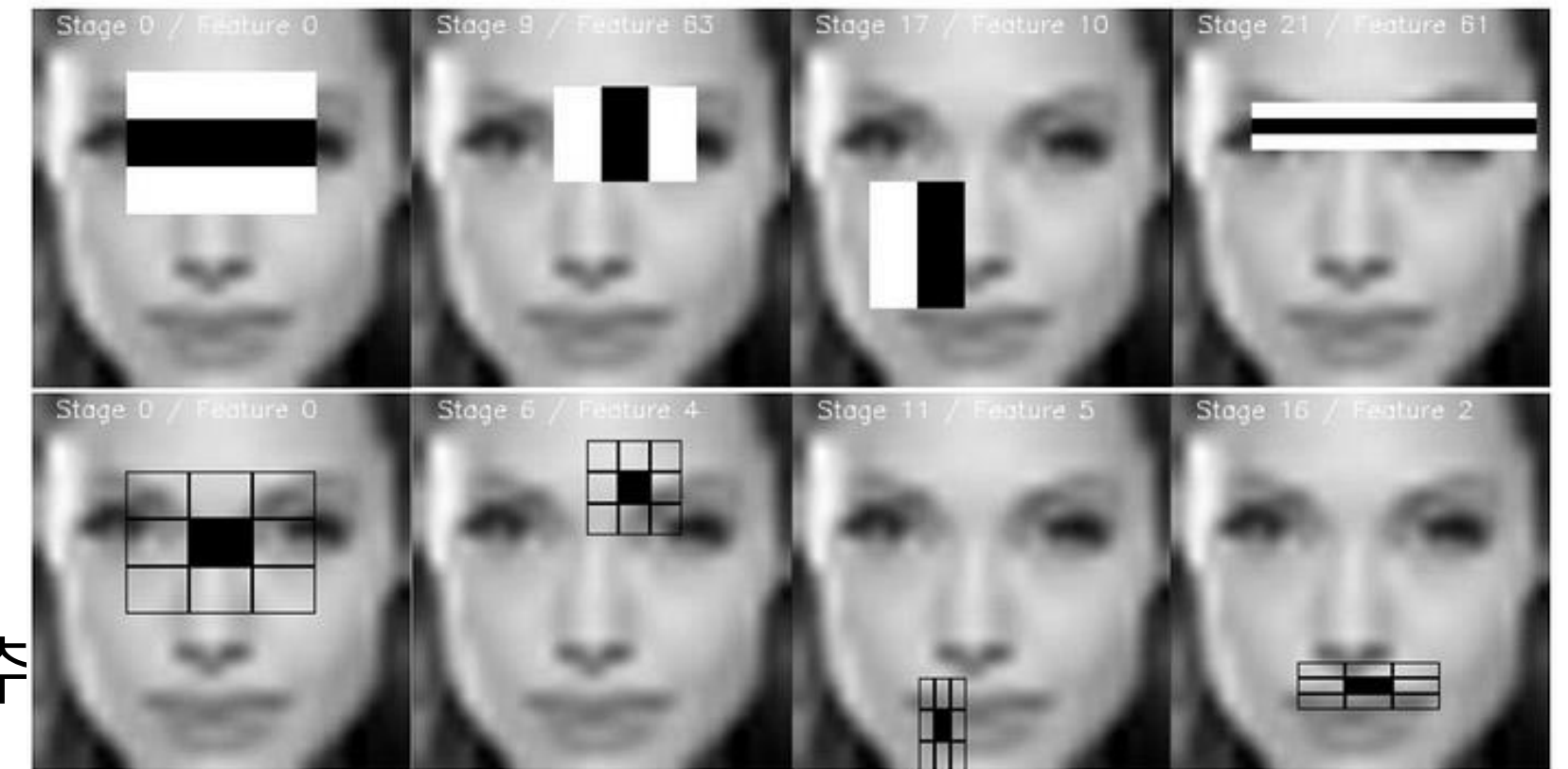
# 머리 부분 (Head)
head:
  type: YOLOHead
  num_anchors: 3
  num_classes: 80
```

3. Object Detection 고전 알고리즘

머신러닝 객체 탐지

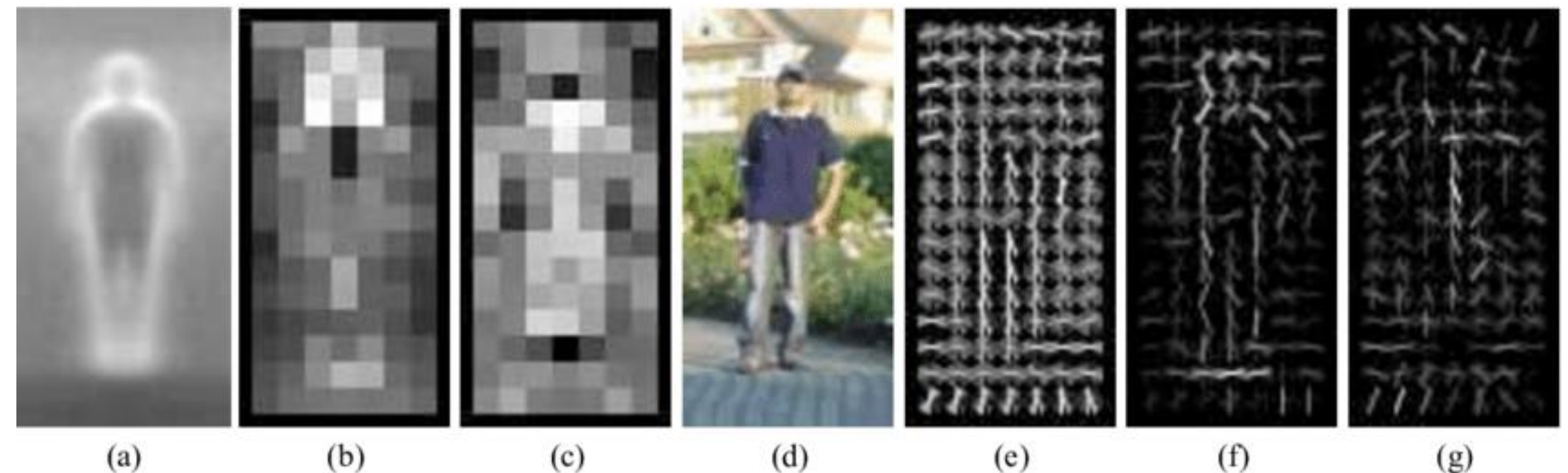
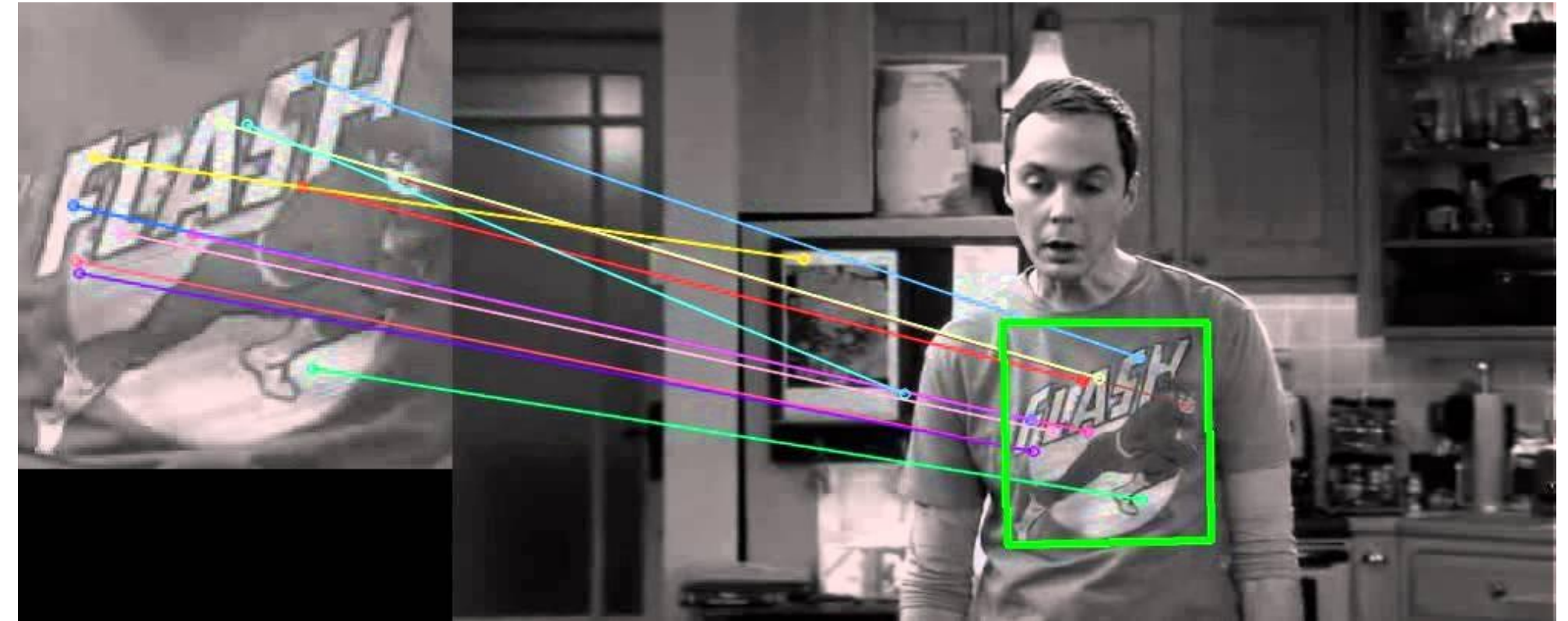
- OD 머신러닝 알고리즘

- 딥러닝이 등장하기 이전부터 연구되던 분야
- Viola-Jones Detector: AdaBoost 알고리즘을 사용한 안면
- SIFT(Scale-Invariant Feature Transform)
 - 이미지에서 키포인트를 검출하고, 각 키포인트 주변의 특징을 추
 - 크기와 회전에 불변한 특징을 생성하여 객체 인식



머신러닝 객체 탐지

- OD 머신러닝 알고리즘
 - SURF (Speeded-Up Robust Features):
 - SIFT보다 빠르게 동작하도록 개선된 알고리즘
 - 이미지에서 강한 특징을 추출하여 객체 인식
 - HOG (Histogram of Oriented Gradients):
 - 이미지의 기울기 방향 히스토그램을 계산하여 객체의 형태를 표현
 - 주로 사람 인식에 사용
- 현업에 사용하기엔 턱없는 성능



딥러닝 객체 탐지

- 일반적인 딥러닝 객체 탐지 모델의 절차
 - 이미지 입력
 - 이미지 전처리(Preprocess)
 - 변환(transformation), 정규화(normalization), 데이터 증강(augmentation)
 - 특징 추출(Feature extraction)
 - ROI searching: 객체 위치 후보 제안(Bounding box suggestion)
 - NMS(Non-Maximum Suppression, 비 최대 억제): 중복되는 바운딩 박스 제거
- 결과 출력
 - 객체 분류(Classification)
 - 바운딩박스 회귀(Regression)

Region Searching – Sliding Window

- Region proposal
 - 모델이 이미지 내 객체가 존재할 것으로 추정되는 후보군을 찾는 과정
- Sliding window searching
 - 이미지 내 모든 지역을 Sliding window 기법으로 탐색함
 - 불필요한 지역을 탐색
 - 지나치게 많은 Bbox가 만들어짐
 - 동일한 크기의 지역만을 탐색
 - 탐색 지역이 넓다면 지나치게 많은 객체가 탐지됨
 - 탐색 지역이 좁다면 큰 객체의 부분만을 인식함
 - 이미지 Pyramid 기법을 통해 탐색



Region Searching – Selective Search

- Selective search
 - 모든 이미지를 아날로그 방식으로 탐색하는 대신, 픽셀 값을 통해 위치를 추정
 - RoI(Region of Interest): 객체가 있을 것으로 추정되는 영역
- 방법
 - 각 픽셀의 값을 기준으로 유사한 픽셀을 군집화
 - 작은 군집들을 다시 모아 대표적인 큰 군집 형성
 - 군집의 수가 충분히 줄어들 때까지 이를 반복
 - 2000여개(R-CNN)
 - 이 또한 지나치게 많은 Bbox가 생성됨

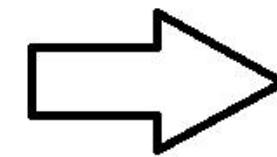


Region Searching – RPN

- RPN(Region Proposal Network)
 - 신경망을 이용하여 이미지 내에서 객체가 있을 가능성이 높은 영역(ROI)를 빠르게 제안
 - 전체 이미지를 픽셀 값으로만 판단하지 않고, 의미를 바탕으로 영역을 유추
 - 신경망이 이미지에서 Feature map을 추출 후, 여러 개의 Bbox를 제안
 - Bbox 생성 시
 - 많이 생성된 Bbox 중 쓸모 없는 것들을 제거하는 알고리즘 포함(NMS)
 - 마구잡이로 Bbox를 만들지 않고, 사전에 정의된 크기와 갯수 만큼의 Bbox를 제안(Anchor box)

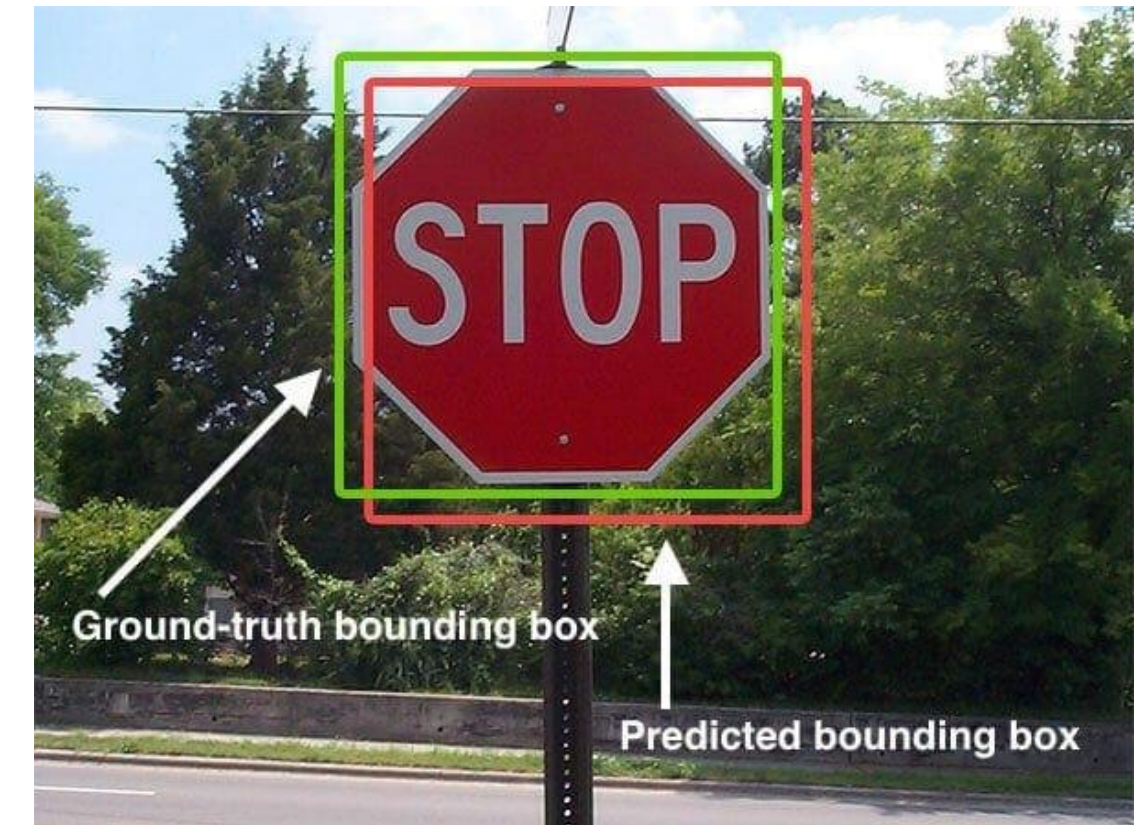
Region Searching – RPN

- NMS(Non-Maximum Suppression)
 - 겹치는 Bbox를 제거하여 최종적으로 하나의 객체에 하나의 Bbox만 남기기 위한 기법
 - 여러 Bbox가 동일한 객체를 감지할 때, 중복되는 박스를 제거하고 가장 정확한 박스를 선택하는 데 사용
- 방법
 - 여러 개의 Bbox를 Confidence score 순으로 정렬
 - 너무 값이 낮은 박스들은 제거
 - 가장 값이 큰 기준 Bbox 선택
 - 기준 박스와 다른 박스의 겹치는 면적을 계산
 - 겹치는 면적이 Threshold(0.5)를 넘지 못한다면 제거



Region Searching – RPN

- IoU(Intersect over Union)
 - 두 영역이 얼마나 겹치는지 평가하는 지표
 - 두 영역의 공통 면적을 이들의 전체 면적으로 나눈 것
 - 범위는 [0, 1] 구간에서 표현
 - 레이블의 박스 크기와 모델의 예측 값을 비교할 때 사용하거나
 - NMS에서 다른 제안된 Bbox들을 제한할 때 사용



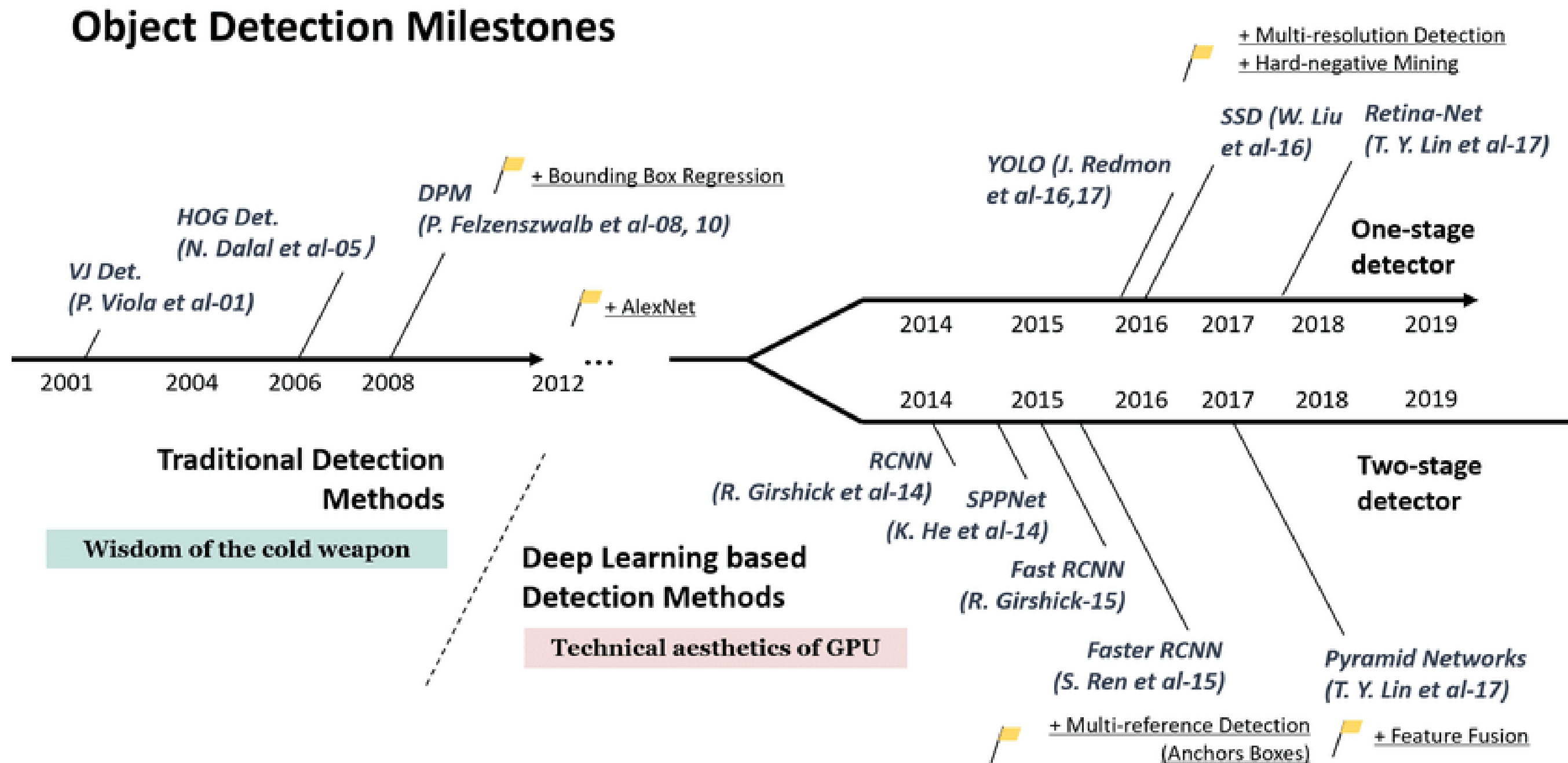
$$\text{IoU} = \frac{\text{Area of Overlap}}{\text{Area of Union}}$$

Region Searching – RPN

- Anchor box
 - 기존 Bbox 제안 방식의 문제점 중 다른 하나는 지나치게 많은 박스가 생성되는 것
 - 신경망이 이미지에서 객체를 탐지할 때에는 대상의 갯수를 고려하지 못하고 여러 박스를 생성
 - 고양이를 수염, 눈, 발톱으로 인식할 경우 객체 하나에서 여러 개의 Bbox가 제안됨
 - 또한 서로 다른 객체가 겹칠 때, 중심점이 가까울 경우 다른 대상의 Bbox를 제거할 수도 있음
 - 객체가 작거나 무리를 이룰 때(비둘기, 식물 등) 자주 발생
 - 특정 구역에서 생성될 수 있는 Bbox의 수와 형태를 사전에 지정하는 것
 - Bbox의 생성 수가 줄어들기에 속도 개선
 - 박스의 좌표와 크기 변화가 작아지므로 학습 쉬워짐



객체 탐지 분야의 갈래

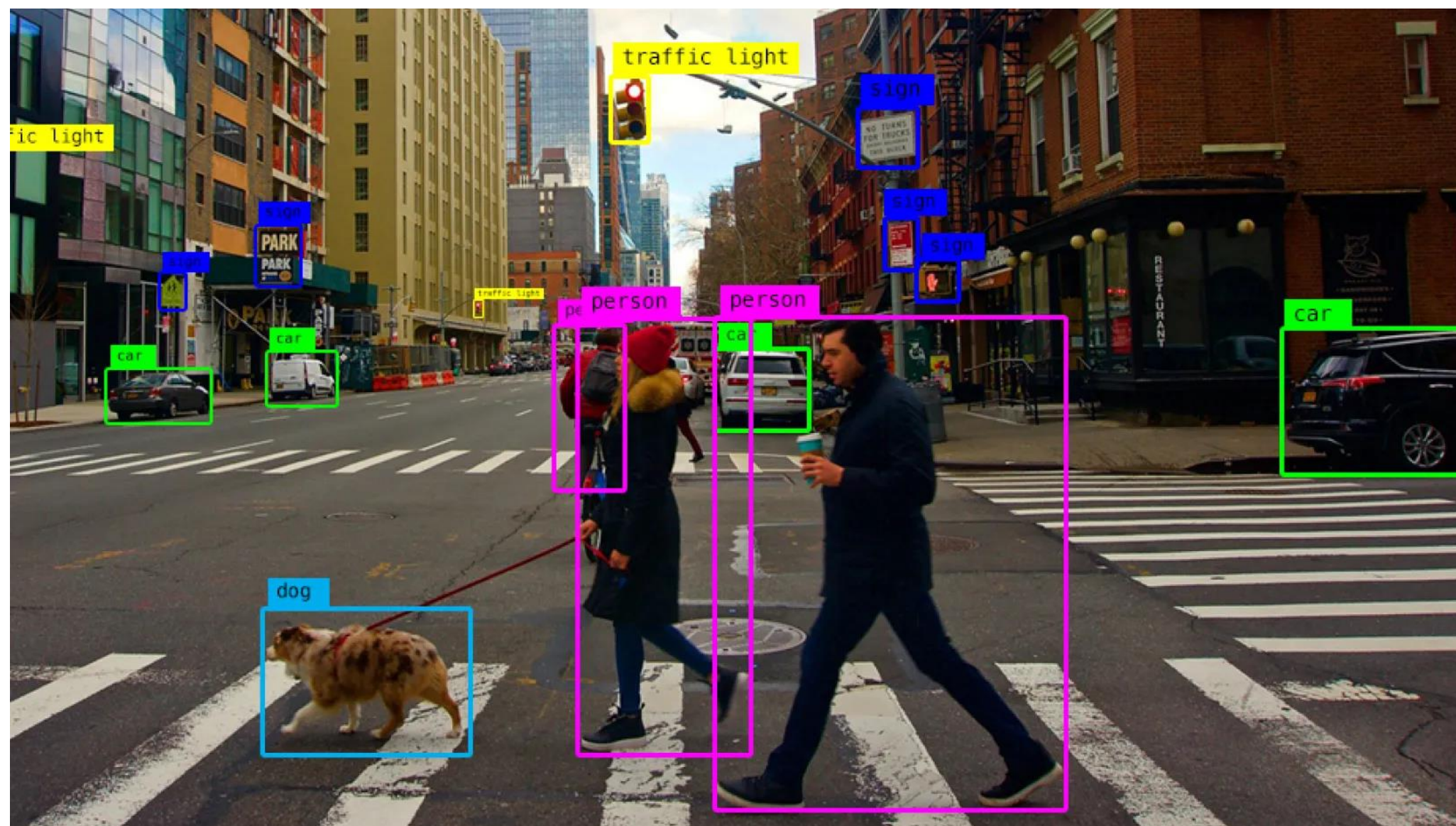


- 딥러닝 컴퓨터 비전 모델이 각광받으며, 객체 탐지 분야가 뒤이어 발전
- 1-Stage detector와 2-stage detector로 나뉘어 연구
- 두 분야의 구분 기준은 분류 문제와 Region proposal을 해결하는 방식에 따라 구분

4. 평가 지표

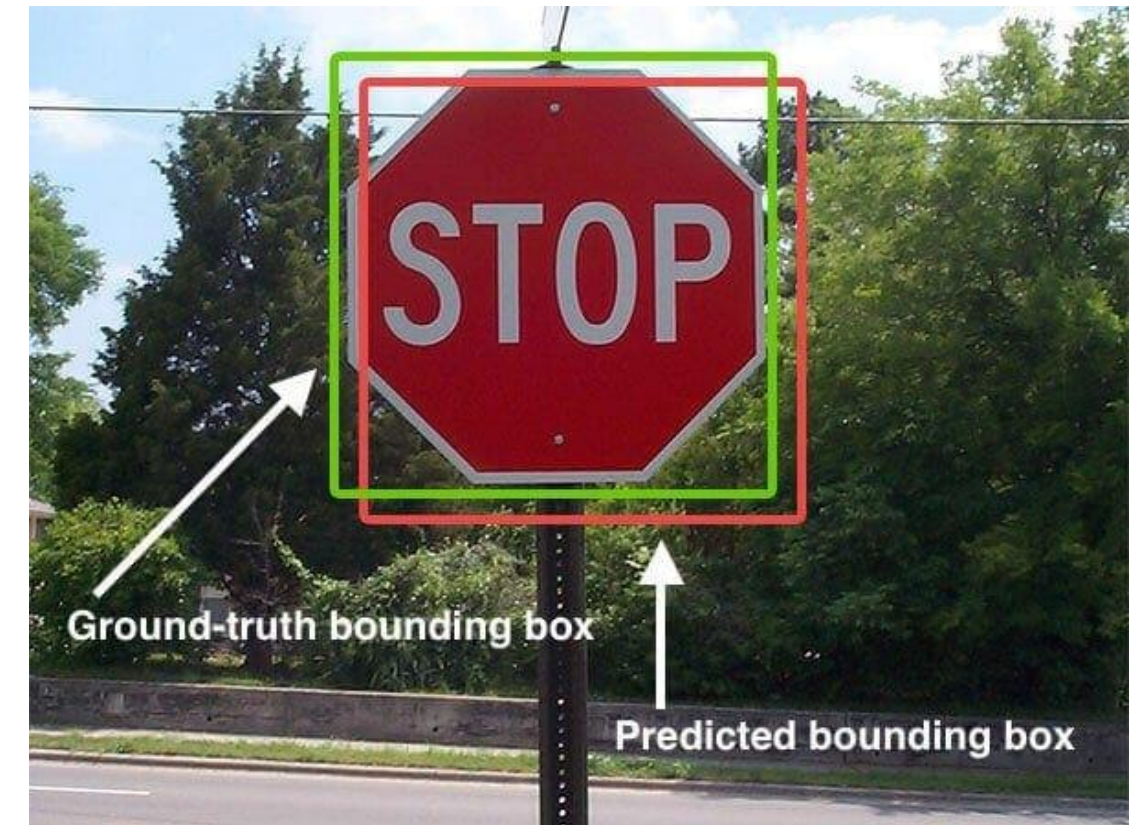
객체 탐지 평가 지표

- 객체 탐지는 분류와 회귀를 모두 해결해야 함
 - 분류 문제의 평가 지표(Confusion matrix)
 - 회귀 문제의 평가 지표(Bounding box regression)
- 그러나 일반적인 분류보다 훨씬 어려움



객체 탐지 평가 지표 – IoU

- IoU(Intersect over Union)
 - 두 영역이 얼마나 겹치는지 평가하는 지표
 - 두 영역의 공통 면적을 이들의 전체 면적으로 나눈 것
 - 범위는 [0, 1] 구간에서 표현
 - 레이블의 박스 크기와 모델의 예측 값을 비교
- mIoU(Mean IoU)
 - 전체 클래스 IoU 값의 평균



$$\text{IoU} = \frac{\text{Area of Overlap}}{\text{Area of Union}}$$

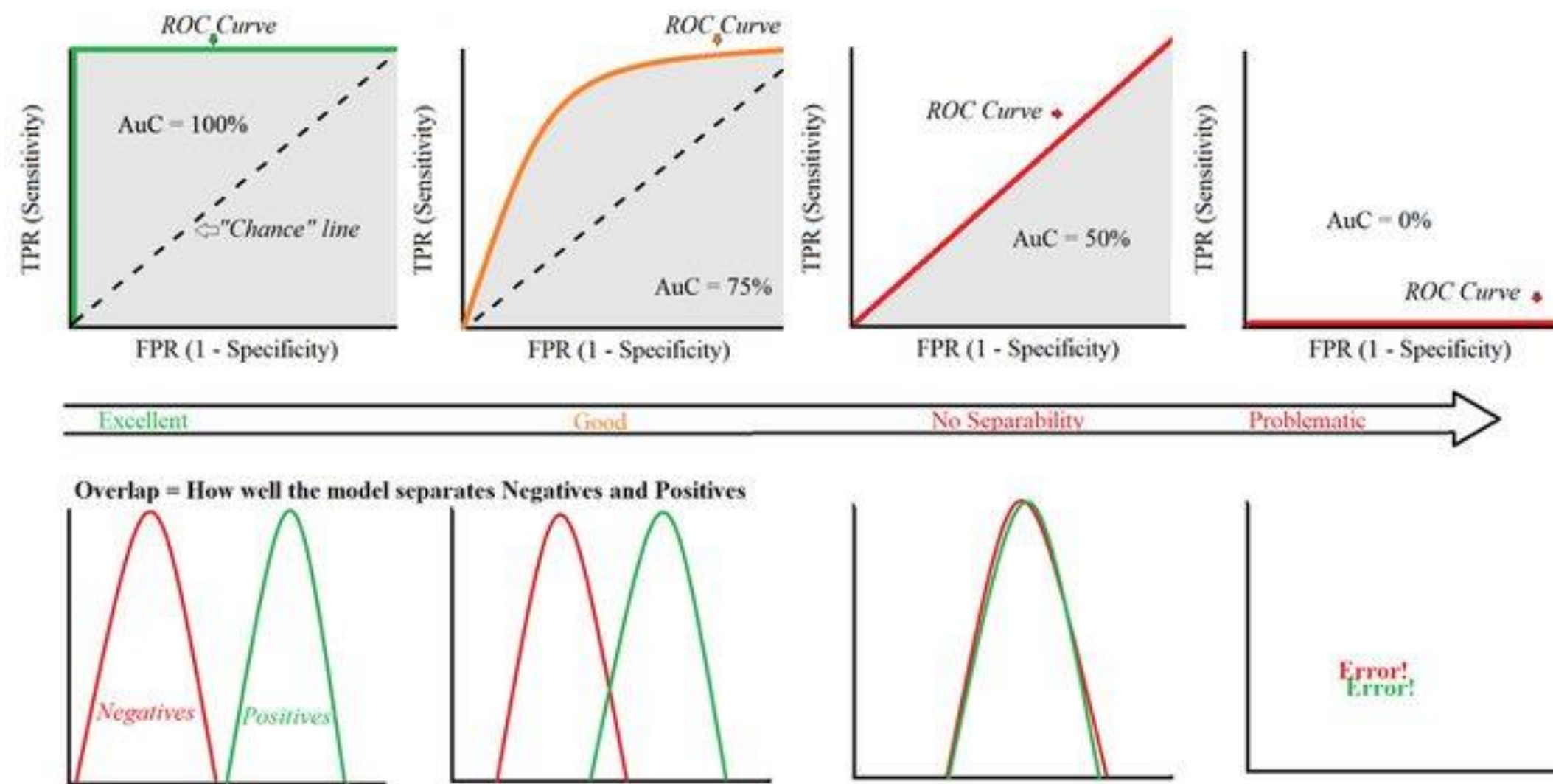
객체 탐지 문제 평가 지표 - AUC

- ROC Curve
 - 모델이 Positive/Negative 예측할 때의 Threshold를 조정하면 TP와 FP의 비율이 바뀜
 - Recall을 TPR(True Positive Rate)이라 부르기도 함
 - 반대로 FP의 비율을 구할 경우 FPR(False Positive Rate)이라 하며, Fallout이라 부르기도 함
 - $Fallout = \frac{FP}{TN+FP} = 1 - TNR$
 - ROC Curve(Receiver Operating Characteristic, 수신자 조작 특성 곡선)
 - Threshold 변화에 따른 TPR(True Positive Rate)과 FPR(False Positive Rate)의 관계를 표현한 곡선

객체 탐지 문제 평가 지표 - AUC

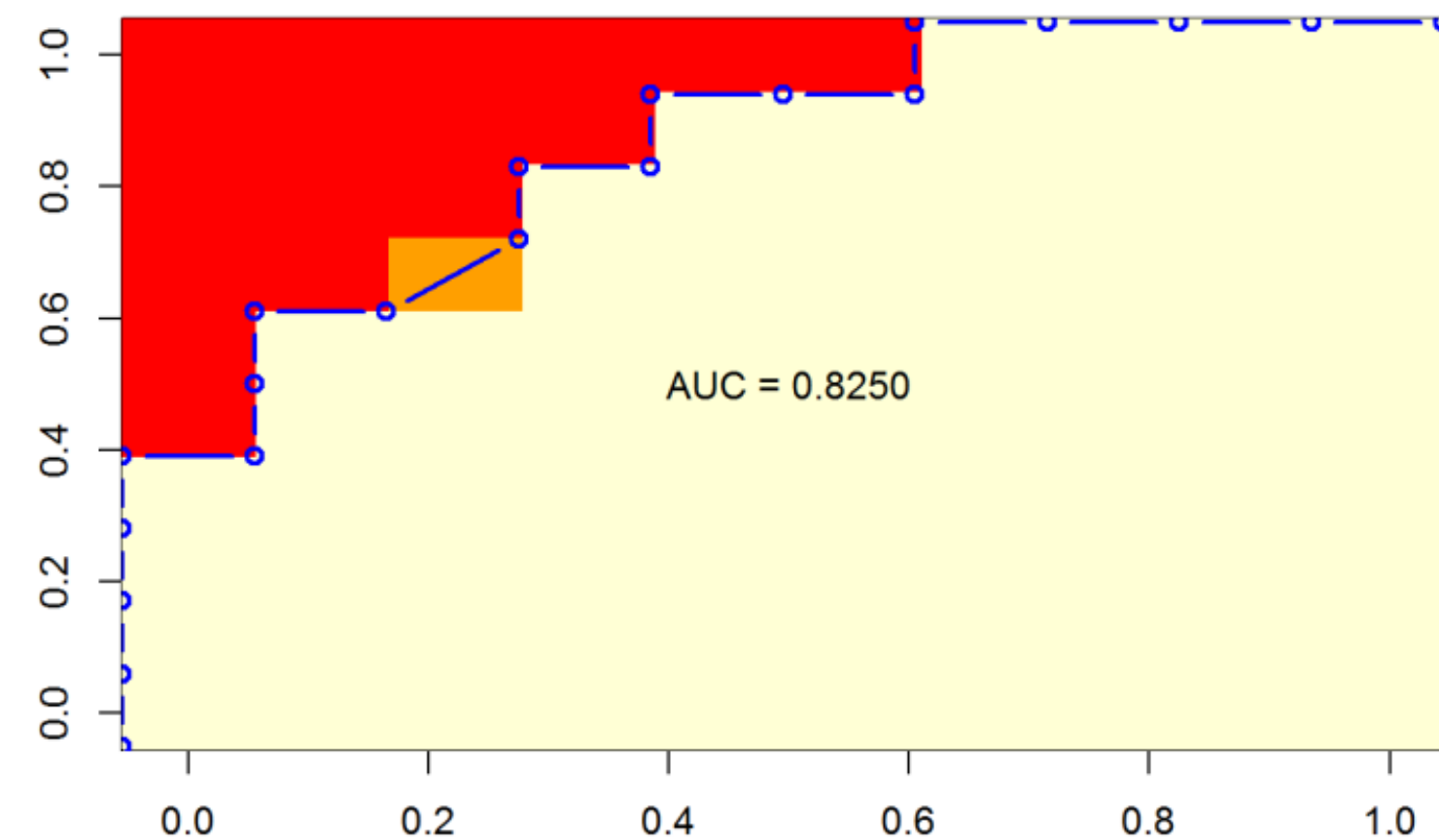
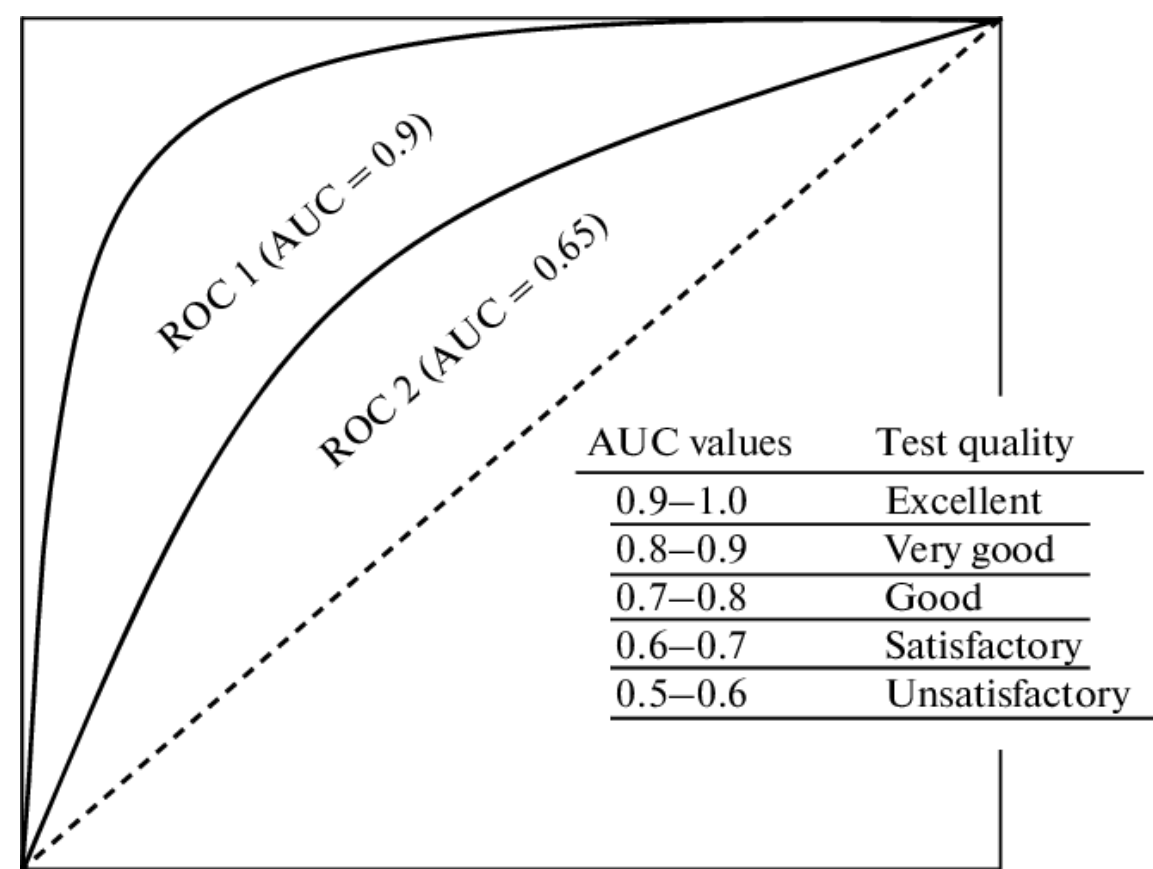
- ROC Curve

- 모델의 성능이 좋을 수록 곡선이 좌측 상단으로 휨
- 성능이 좋지 않을 수록, 우상향하는 직선의 형태에 가까워짐
- 이 형태는 모델이 각 범주를 얼마나 잘 분리하는지에 따라 결정



객체 탐지 문제 평가 지표 - AUC

- AUC(Area Under Curve)란 ROC Curve 아래의 면적을 의미하며 0-1사이 값으로 표현
 - 면적의 넓이 비율은 모델의 성능을 평가할 수 있는 지표
 - AUC=1: 분류 성능이 완벽
 - AUC=0.5: 모델이 무작위로 분류하는 것과 같음
 - AUC=0: 모델이 완벽하게 잘못 분류



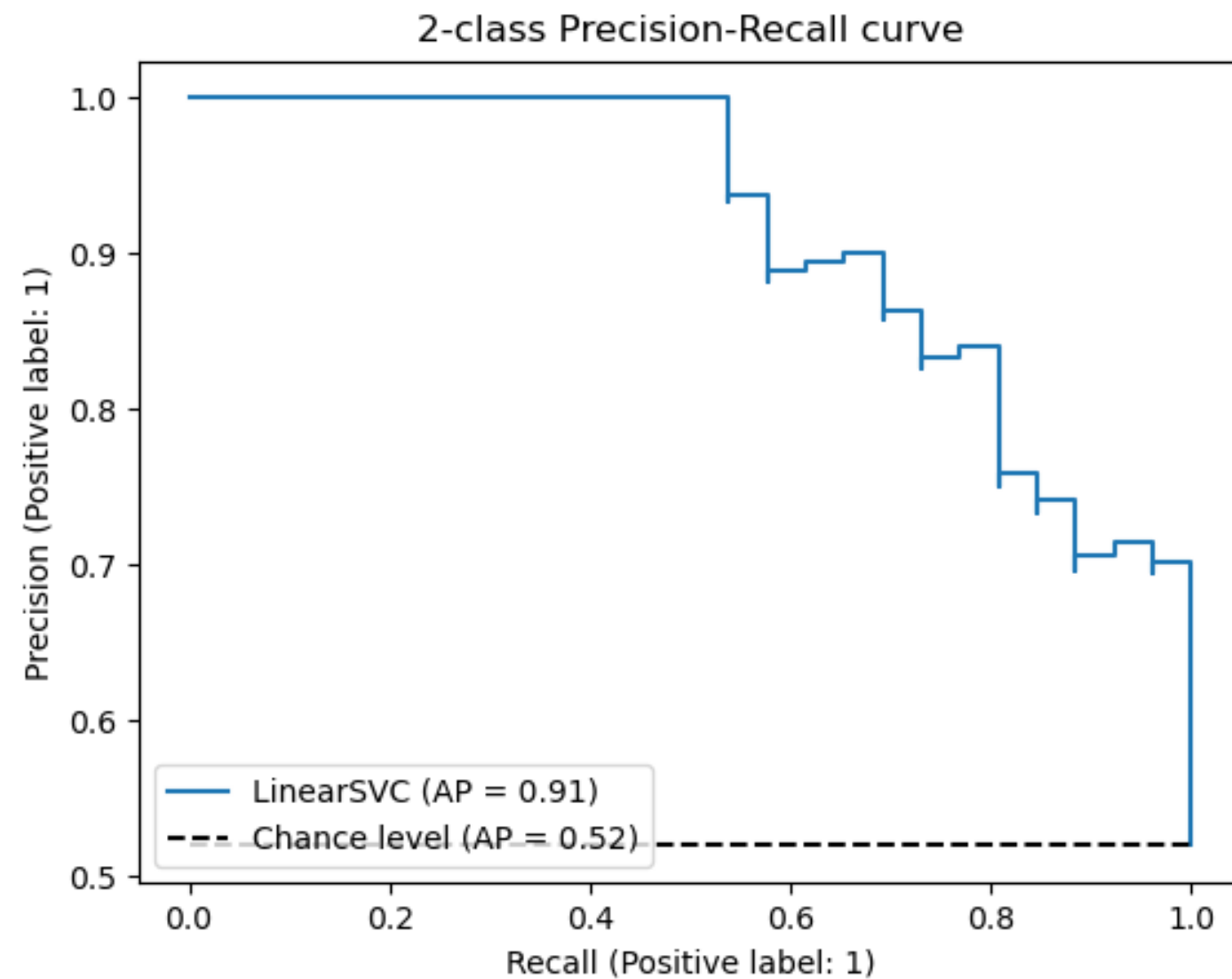
객체 탐지 문제 평가 지표 - mAP

- 정밀도(Precision)
 - 모델이 예측한 객체 중 실제로 올바르게 예측한 객체의 비율
 - $Precision = \frac{TP}{TP+FP}$
- 재현율(Recall)
 - 실존하는 객체 중 모델이 올바르게 탐지한 객체의 비율
 - $Recall = \frac{TP}{TP+FN}$
- 두 지표는 정확도의 단점을 보완하기에 좋으나, Threshold 기준 상쇄(Trade-off) 관계에 있음
 - 두 지표를 모두 사용하기 위한 방법: F1-score, ROC Curve, PR Curve

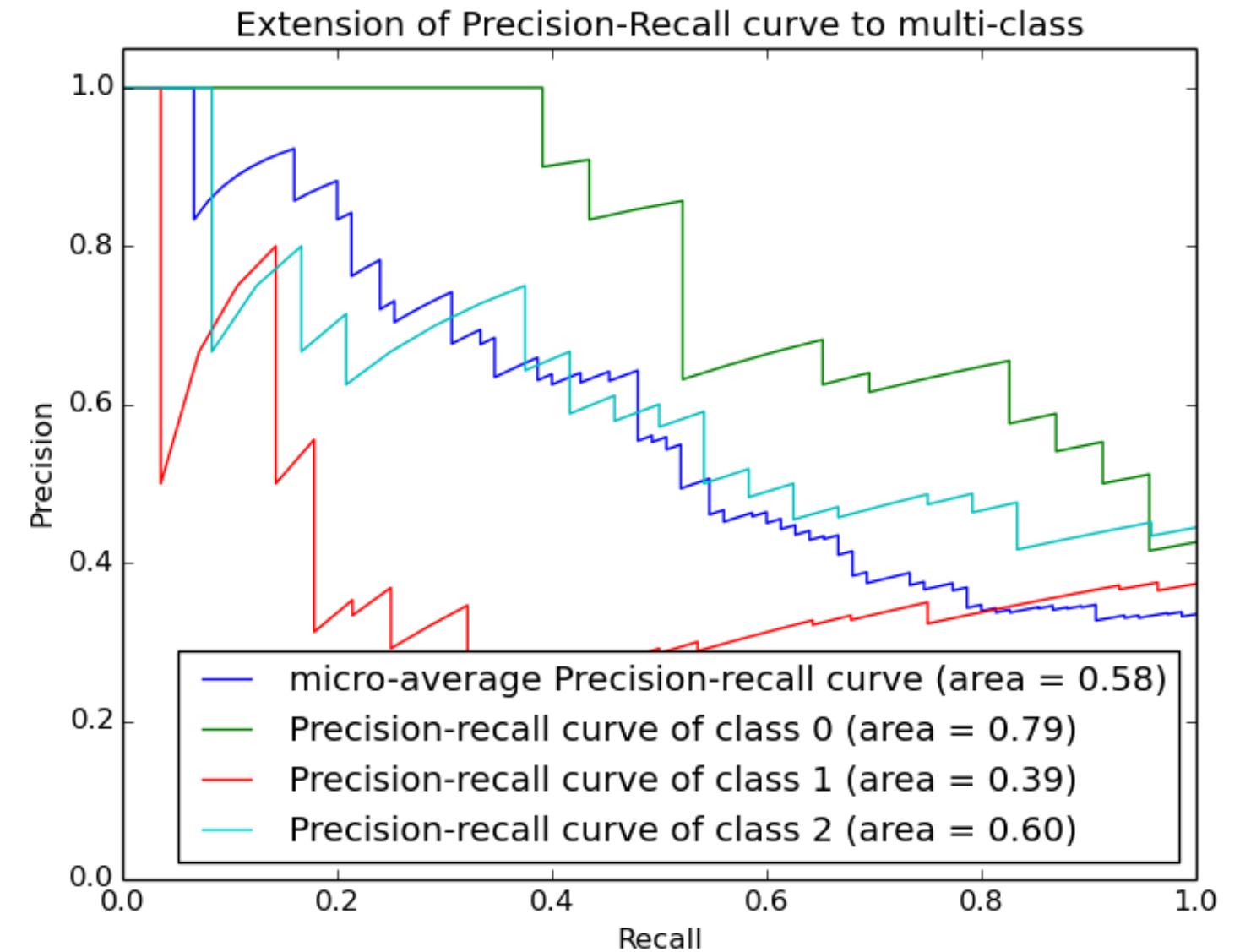
객체 탐지 문제 평가 지표 - mAP

- Precision – Recall은 Trade-off 관계이므로
 - 모델이 Positive/Negative 예측할 때의 Threshold를 조정하면
 - Confusion matrix의 결과가 달라짐
 - 즉 FP와 FN의 비율이 바뀜
- 임계값의 변화에 따른 Precision과 Recall의 관계를 그린 곡선을 Precision-Recall Curve라 함
 - Precision – Recall curve에서는 두 지표가 모두 높을수록 우수한 모델을 의미
 - 그래프가 우상향하는 경우에 해당

객체 탐지 문제 평가 지표 - mAP



이진 분류 시 PR-Curve의 시각화



다중 분류 시 PR-Curve의 시각화

- 파란색: 전체 평균
- 초록색: 0번 클래스
- 빨간색: 1번 클래스
- 하늘색: 2번 클래스

객체 탐지 문제 평가 지표 - mAP

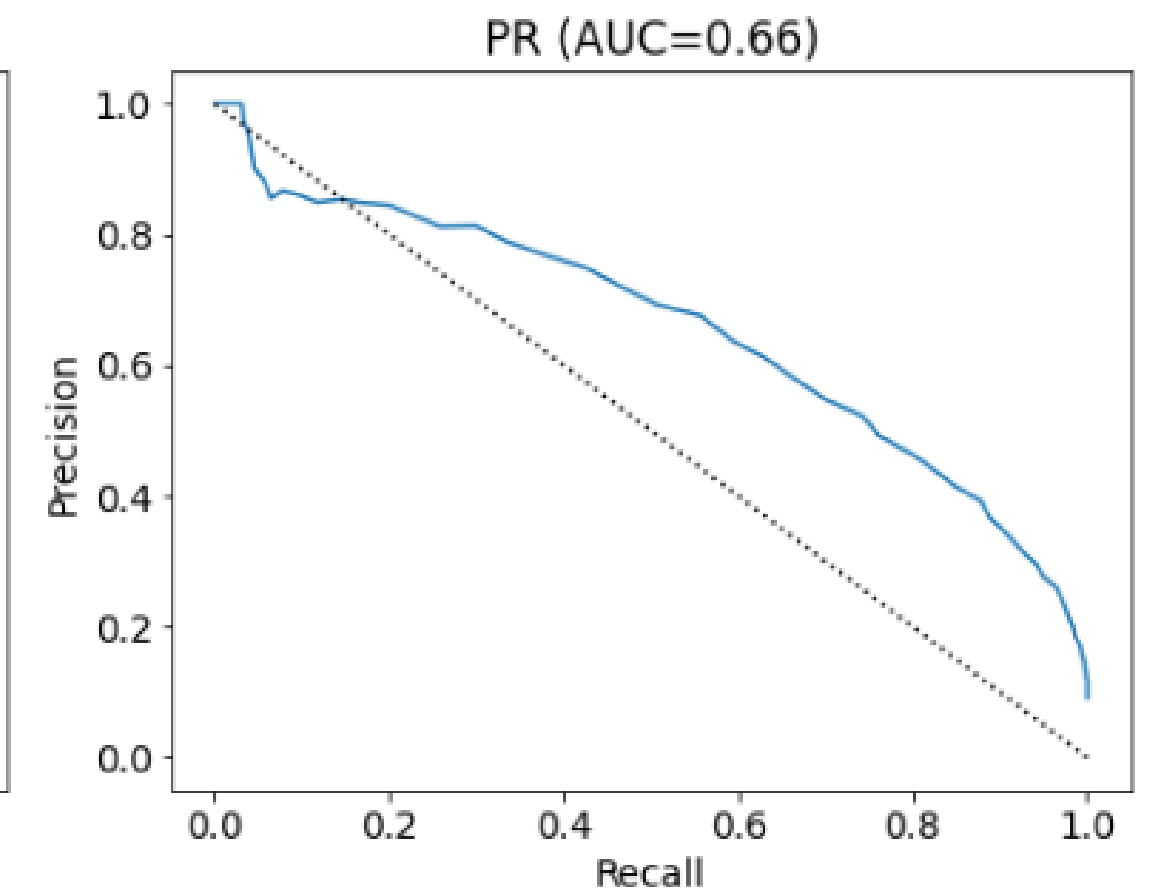
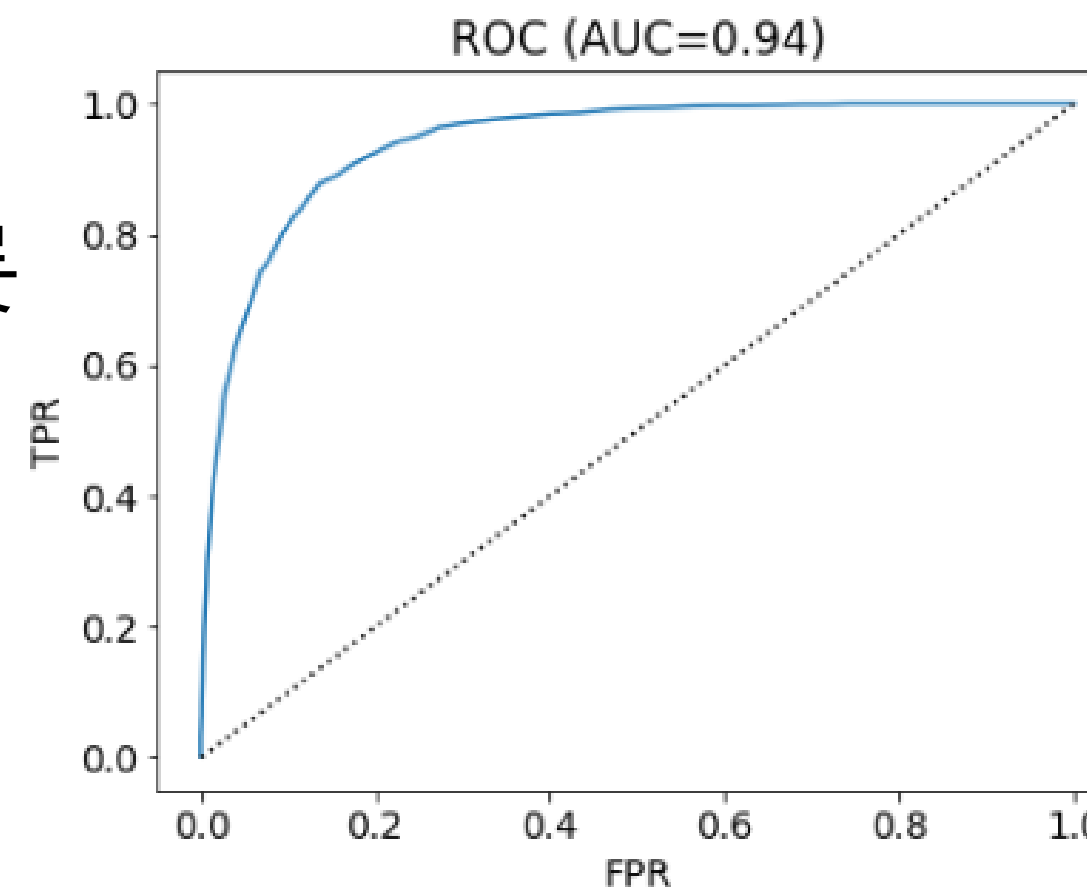
- 또는 PR-Curve의 아랫부분 면적을 구하면 모델의 성능을 평가할 수 있음
 - 이 면적을 AP(Average Precision)이라 부름
 - AP는 0-1 사이의 값으로 표현되며, 1에 가까울 수록 모델 성능이 우수함
- 다만, 객체 탐지 모델은 다중 클래스에 대한 분류를 수행하므로, 종합적 성능을 고려해야 함
 - mAP(Mean Average Precision): 다중 클래스에 대한 AP의 평균

객체 탐지 문제 평가 지표 - mAP

- mAP@K
 - 객체에 대하여 Bbox가 예측되었더라도, 항상 모든 값을 신뢰할 수는 없음
 - Bbox는 분류와 회귀 모두 올바르게 예측해야 함
 - 박스 내 객체의 존재 여부(Confidence score)와 클래스 분류(Class distribution) → mAP
 - 박스의 위치와 크기(xywh Regression) → IoU
 - Bbox 중 1차에서 IoU로 적합하지 않은 것을 제거한 후 mAP를 측정하는 방식이 mAP@K
 - K는 IoU의 threshold
 - mAP@50: IoU threshold=0.5로 제한한 뒤 남은 Bbox의 mAP를 계산한 결과(모델의 전반적인 성능)
 - mAP@90: IoU threshold=0.9로 제한한 뒤 남은 Bbox의 mAP를 계산한 결과(모델 정밀 진단)

객체 탐지 문제 평가 지표

- PR Curve v.s. ROC Curve
- PR curve는 Precision과 Recall을 바탕으로 계산
 - 즉 FP와 FN 모두 엄격하게 평가하므로, 불균형한 데이터셋에 민감하게 반응함
- ROC curve는 TP와 FP의 비율을 바탕으로 계산
 - 즉 FN을 고려하지 않기에 모델 성능을 낙관적으로 평가함
 - 일반적인 모델 성능 평가에 유용함
- 두 지표 모두 분류 문제 평가에서 비롯



객체 탐지 문제 평가 지표

- 객체 탐지에서의 분류는 매우 불균형한 데이터를 다룸
 - 데이터의 클래스 불균형
 - 사람 v.s. 버스
 - Bbox의 불균형
 - 모델에서 생성되는 Bbox는 갯수가 매우 많으나
 - 이들 중 대부분은 배경 Class를 지칭함
- 그렇기에 Class imbalance에 민감한 지표 사용
 - ROC Curve
 - PR Curve

