14.9 객체 탐지

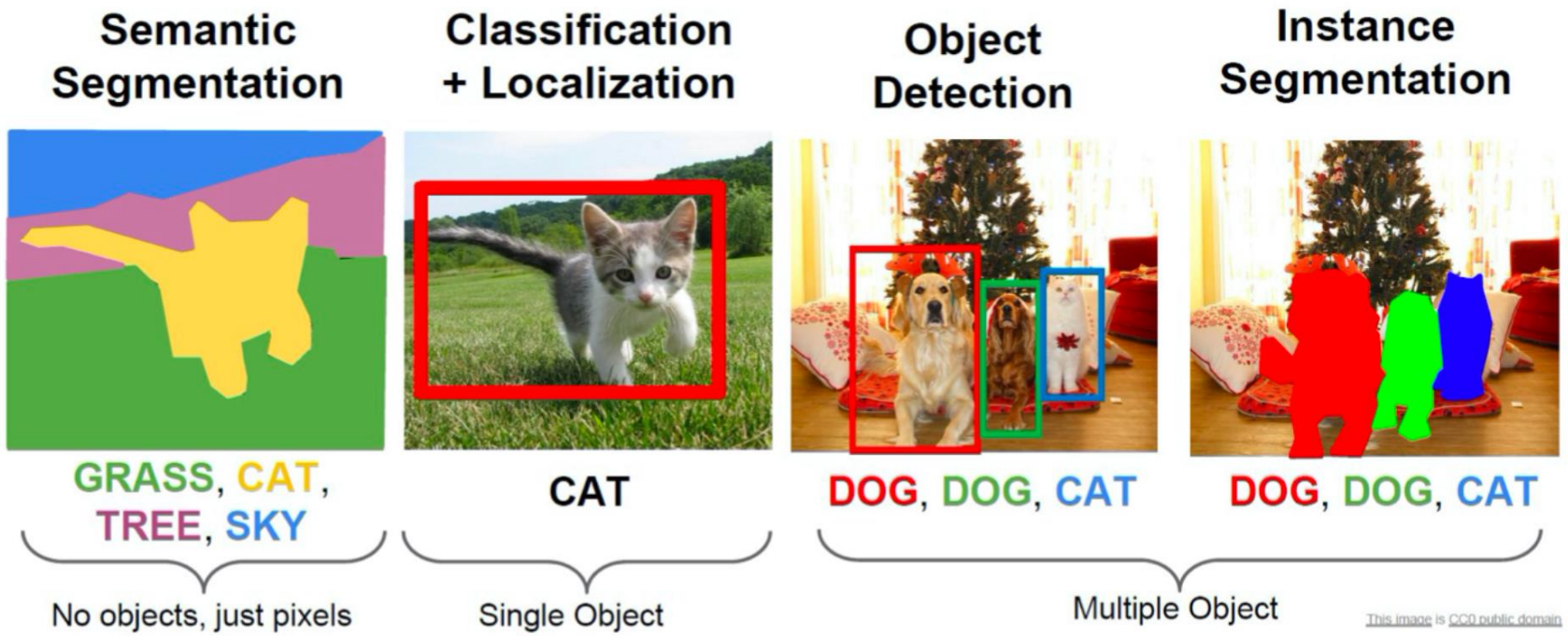
2021-04-25 백관구

14.9. 객체 탐지(Object detection)

- 정의: 하나의 이미지에서 여러 물체를 분류하고 위치를 추정하는 작업

  ➔ 즉, **어떤 객체(label)**가 **어디(x, y)**에 **어느 크기(w, h)**로 존재하는지 찾는 작업

- 종류: NMS (Non-Maximum Suppression), FCN (Fully Convolutional Network), YOLO (You Only Look Once), SSD (Single Shot multibox Detector), R-CNN (Regional CNN)



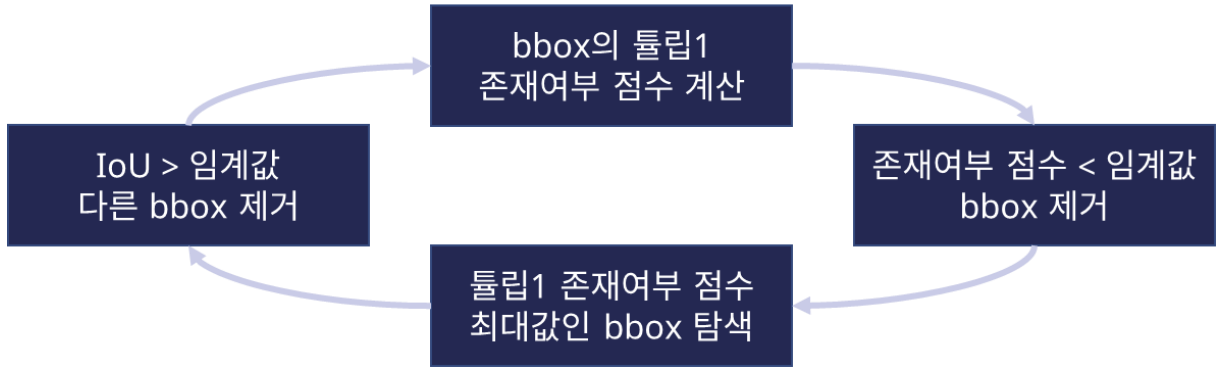
https://mj-lahong.tistory.com/84

14.9. 비최대값 억제 알고리즘(Non-Maximum Suppression; NMS)

- 목적: 다른 위치에서 동일한 물체를 여러 번 감지하는 경우, 불필요한 바운딩 박스(bbox)를 제거
- 원리: **가장 스코어가 높은 bbox만 남기고 나머지를 제거**
- 과정
  1. 존재여부 점수 출력: 객체가 이미지에 존재하는지 확률을 추정(Sigmoid 사용)
  2. 존재여부 점수가 임계값 이하인 bbox를 제거
  3. 존재여부 점수가 가장 높은 bbox 탐색
  4. 위 bbox와 많이 중첩된(IoU) 다른 bbox를 제거
  5. 제거할 bbox가 없을 때까지 반복
- 단점: 객체마다 반복 계산해야 하므로 속도가 **느림**

  → 대신, 비교적 빠른 완전 합성곱 신경망(FCN)을 사용

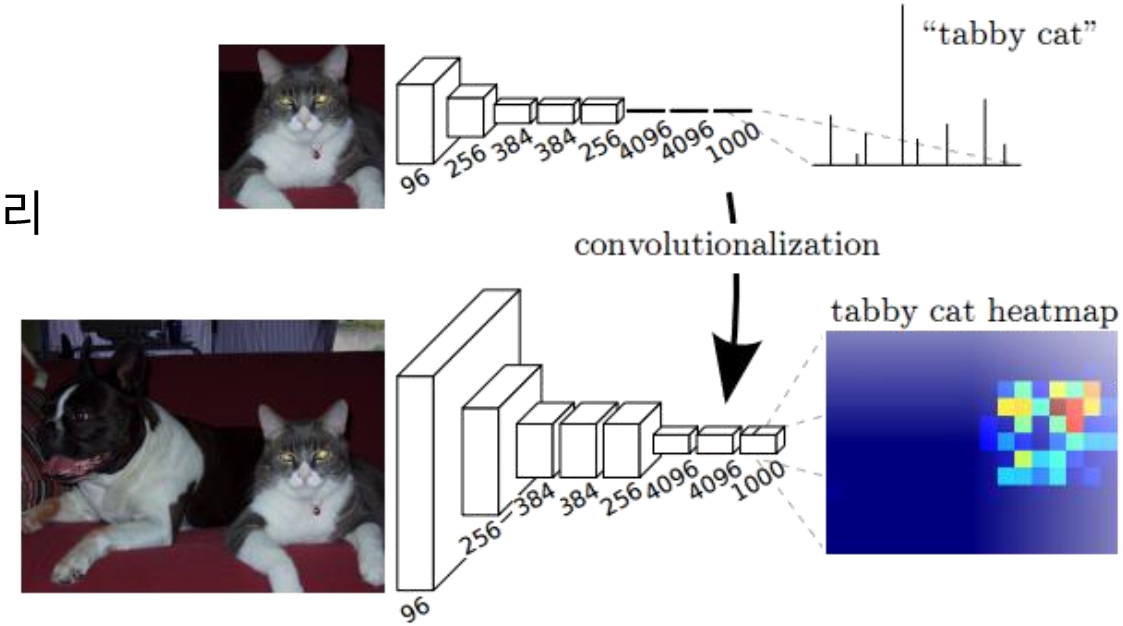
<tèulip1 탐지 과정>



14.9.1. 완전 합성곱 신경망(Fully Convolutional Network; FCN)

- Fully Convolutional Networks for Semantic Segmentation (Jonathan Long et al., 2015)
- 특징
  1. 기존 CNN 구조에서는 맨 위의 층이 밀집 층(Dense layer)였지만, 이 또한 합성곱 층(Convolutional layer)으로 바꿔 오직 합성곱 층으로만 구성
  2. 입력 이미지를 패치 단위로 분석하지 않고, 한번에 처리
  3. NMS에 대한 연산이 줄어들어 학습 속도가 빠름
- 합성곱 층의 장점
  1. **어떤 크기의 이미지**도 처리할 수 있음
  2. **위치 정보를 유지**할 수 있음

<기존 CNN>



<FCN>

14.9.2. YOLO (You Only Look Once)

- You Only Look Once: Unified, Real-Time Object Detection (Joseph Redmon et al., 2016)
- 이후 2016년에 YOLOv2 → 2018년에 YOLOv3
- **매우 빠른 객체 탐지** 알고리즘

  ➔ 영상에도 실시간으로 적용 가능

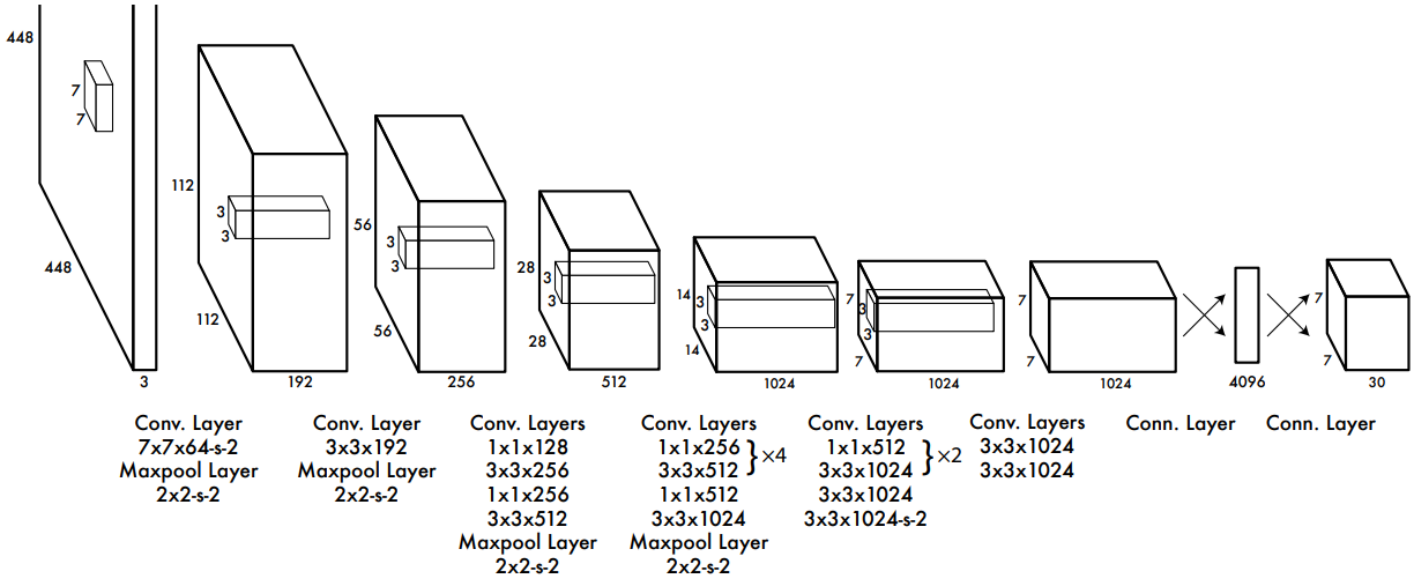


Figure 3: The Architecture. Our detection network has 24 convolutional layers followed by 2 fully connected layers. Alternating 1×1 convolutional layers reduce the features space from preceding layers. We pretrain the convolutional layers on the ImageNet classification task at half the resolution (224×224 input image) and then double the resolution for detection.

https://arxiv.org/pdf/1506.02640.pdf

14.9.2. YOLOv3 (You Only Look Once version 3)

- 특징

(0, 0)

(1, 1)

1. bbox 중심의 상대 좌표를 예측: (0, 0)은 왼쪽 위, (1, 1)은 오른쪽 아래를 의미
2. 앵커 박스(Anchor box): 신경망 훈련 전, 대표적인 bbox 크기를 탐색

   ➔ k-평균 알고리즘을 사용해 bbox의 높이와 너비의 전형적인 비율을 추적

   예시) 훈련 이미지에 많은 보행자가 있다면, 앵커 박스들 중 하나는 전형적인 보행자의 크기를 반영할 것임(높이 100픽셀, 너비 50픽셀)

   → bbox의 높이와 너비를 보행자 크기의 비율에 맞게 스케일을 조정
3. 다양한 크기의 이미지 사용: 신경망이 학습하는 동안 랜덤하게 새로운 이미지 크기를 선택(330X330에서 608X608 픽셀까지)

   ➔ 다양한 스케일의 객체를 감지하도록 학습

- 트레이드-오프(Trade-off): 스케일 v.s. 정확도
- **https://github.com/zzh8829/yolov3-tf2** 에서 YOLOv3를 TensorFlow로 구현

추가. 객체 탐지의 평가 지표 mAP

- Mean Average Precision (mAP)
- 정밀도(precision)/재현율(recall)은 일반적으로 트레이드-오프 관계
- **평균 정밀도(Average Precision; AP)**

  ➔ 오른쪽 아래 그림의 빨간색 선 아래의 넓이
- 두 개 이상의 클래스가 있는 경우, 각 클래스에 대해 AP를 계산한 다음 평균 AP를 산출 ➔ mAP!

실제 상황 (ground truth)	예측 결과 (predict result)	
	Positive	Negative
Positive	TP(true Positive) 옳은 검출	FN(false negative) 검출되어야 할 것이 검출되지 않음
Negative	FP(false positive) 틀린 검출	TN(true negative) 검출되지 말아야할 것이 검출되지 않음

$$Precision = \frac{TP}{TP + FP} = \frac{TP}{All\ Detections} \qquad Recall = \frac{TP}{TP + FN} = \frac{TP}{All\ Ground\ truths}$$