



Google Developer Student Clubs
Hanyang

ML/DL 스터디

basic



김남호, 이지환, 김찬원
GDSC Hanyang

8주차 Kaggle - Yolo를 활용한 선수 tracking

```
lookup.KeyValue  
f.constant(['em  
=tf.constant([G  
.lookup.StaticV  
_buckets=5)
```

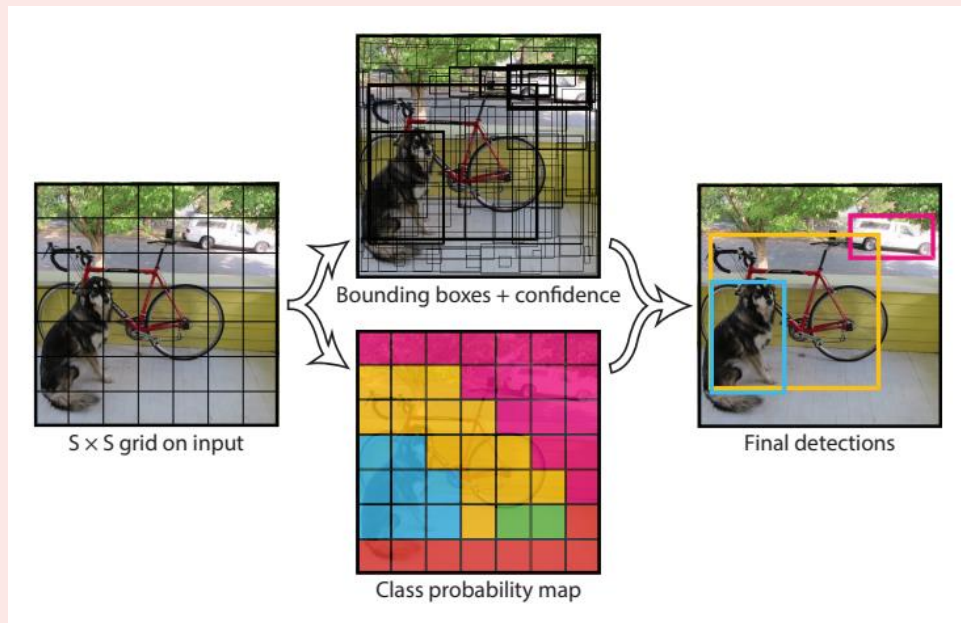
목차

1. YOLO 이론 설명
2. 바이트트랙 이론 설명
3. 모델 활용 아이디어

YOLO 이론 설명

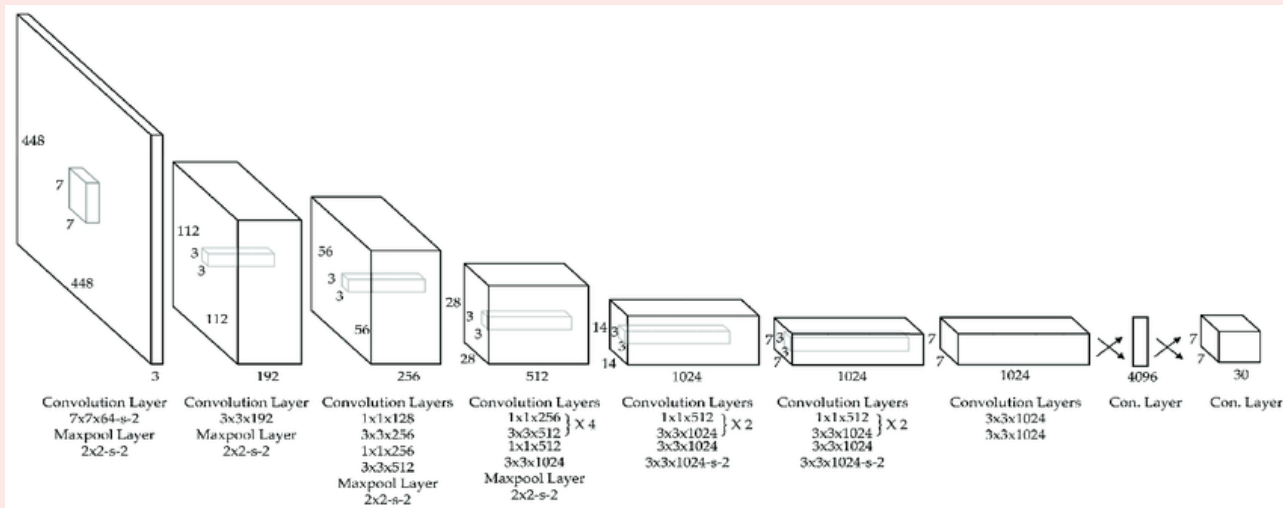
You Only Look Once (YOLO)

1. 이미지를 patches 들로 나누고
2. 각 patch 마다 bounding box를 예측한다. (ex. X좌표, y좌표, width, height, confidence score = 각 정보별로 bounding box 두개로 하면 10 features)



YOLO Architecture

여러 convolutional layer 과 pooling을 지나 7by 7by 30의 3D tensor형태의 정보가 나옴
각각의 좌표마다 x, y, w, h, confidence에 2 bounding box -> 10 variables
나머지 $30 - 10 = 20$ variable 들은 20개의 각각의 class들로 들어갈 확률을 나타냄



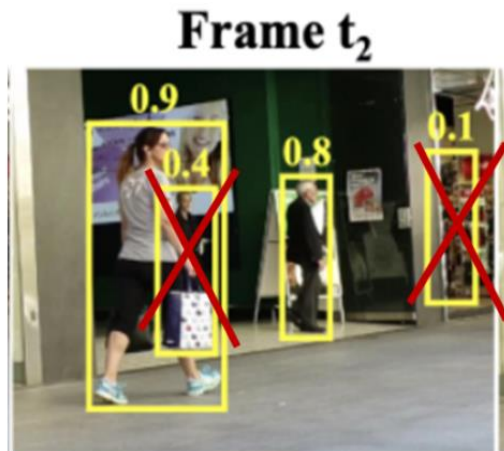
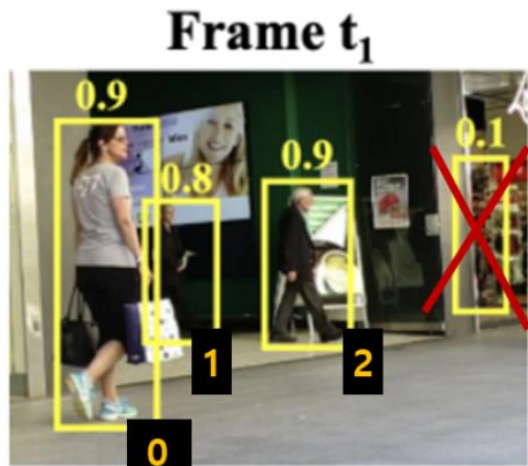
ByteTrack 이론 설명

Bytetrack

1.confidence score 0.5이상만 tracking
대상에 고려 및 box들에 ID 부여

2. t2 시각의 새로운 프레임에서
confidence score에 따른 필터링 후
t1과 IOU 가 가장 높은 box들끼리 매칭


3. 여기서 box 1은 매칭이 안됨



IOU

Intersection over union

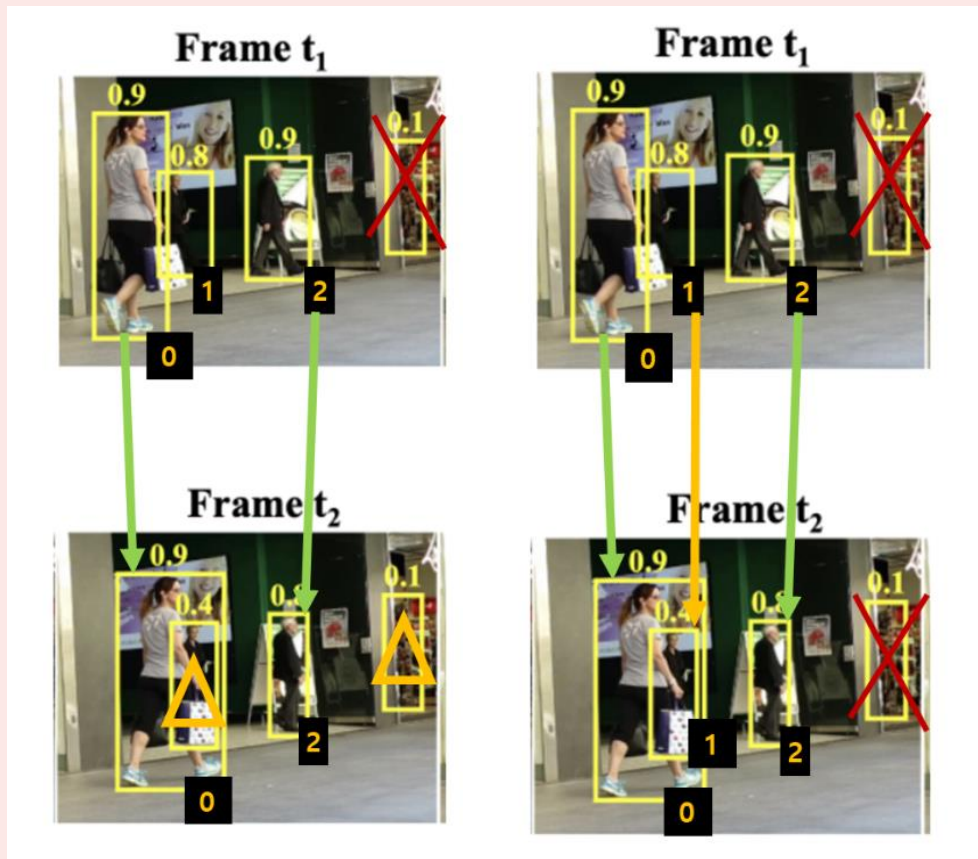


$$\text{Score} = \frac{\text{Area of overlap}}{\text{Area of union}}$$


Bytetrack

4. 매칭 할 때 Confidence Score가
적당히 낮은 것들도 일단은 살려 둬
(오른쪽 그림에서 세모)

5. 최종적으로 Occlusion이 발생한
왼쪽에서 두번째 사람 또한 매칭이 됨



모델 발전 아이디어

To-Do List

1. 인덱싱을 해서 return하는 것
2. 이전단위 프레임에 대한 객체의 detection box에 대한 좌표를 return하는 것
3. 그 좌표 값에 대한 점을 찍는 것을 누적해서 계속 어떤 리스트형태로 계속 관리(최신화)
4. 우리가 보는 현재 프레임에 그 점을 넣어주는 것



Problems

1. class_id vs tracked_id

class_id: 심판, 선수 등을 나누는 id -> 특정 선수를 인덱싱하지 않음

tracker_id: ByteTrack에서 가져올 수 있는 id -> 특정 선수를 인덱싱함

=> ByteTrack으로 tracking이 선행되어야만 tracker_id를 활용할 수 있음

2. 반복 예측 시 고정되지 않는 인덱스

예측 시마다 index 값이 23 또는 24씩 증가, index를 고정시킬 방안?

=> 바이트트래킹이 indexing에 관여하는 부분 추가 공부

3. 구도의 움직임에 따른 점 위치 수정

단순히 이미지 위에 점을 찍으면 구도에 따라 점을 이동시킬 수가 없음

=> 영상에서 고정되어있는 물체를 이용해서 카메라의 속도를 파악해서 점 위치를 빼는 방법

```
child: Column(  
  crossAxisAlignment: CrossAxisAlignment.  
  children: [  
    /*2*/  
    Conta  
    pad  
    chi  
    '  
    s  
    )  
  ),  
),  
Text(  
  'Ka  
  sty  
  c  
  ),  
),  
),
```

발표를 들어 주셔서 감사합니다.