

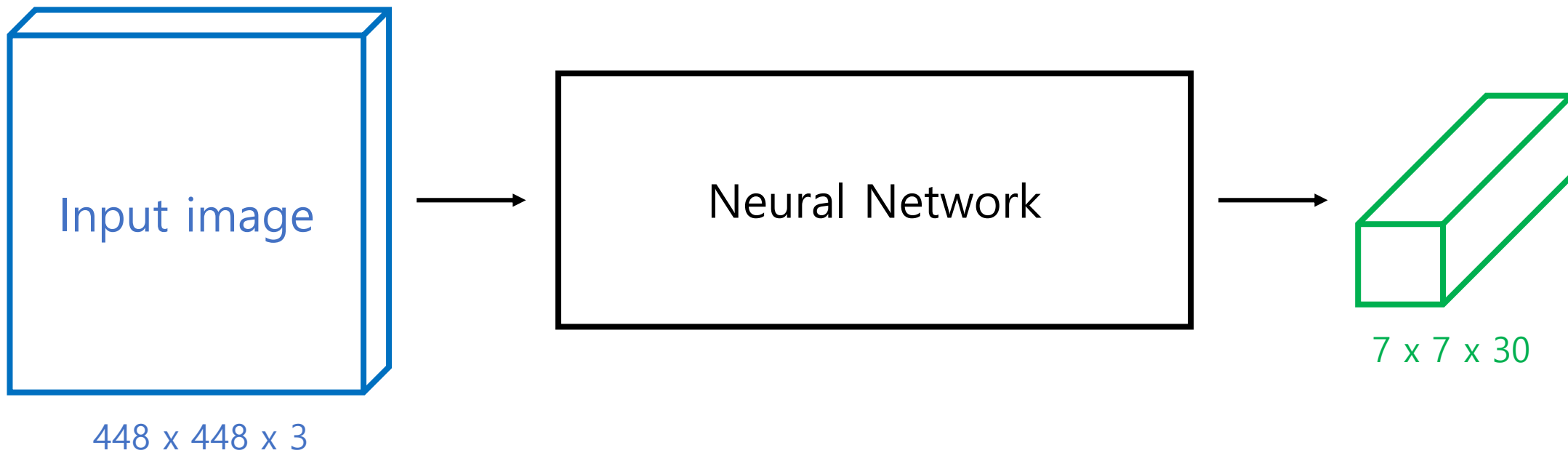
# **You Only Look Once: Unified, Read-Time Object Detection**

**박태우**

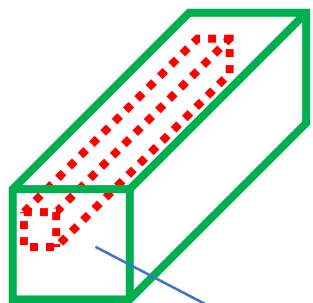
# YOLO란

- 속도를 극대화한 Object detector.
- 다른 object detector의 경우 sliding window 또는 region proposal 사용.
- YOLO의 경우  $S \times S$  grid 셀을 이용함.
- 핵심은 전체 이미지를 한번만 보고 detection을 수행함.
- Regression : bounding-box의 위치를 결정.
- Classification : 해당 object가 어떤 종류인지 결정.
- YOLO는 regression과 classification을 한번에 학습함.

# 동작 알고리즘



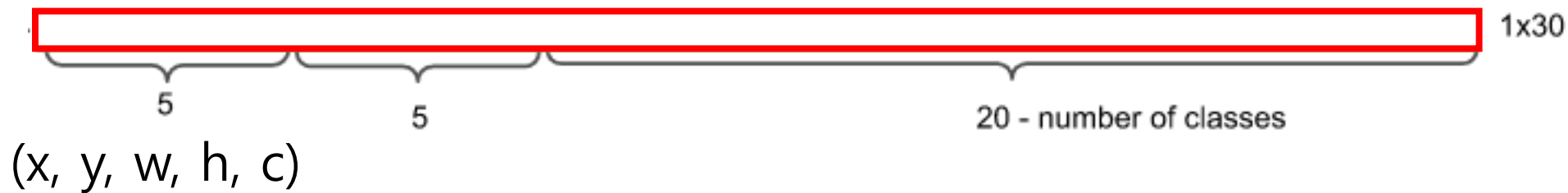
# 동작 알고리즘



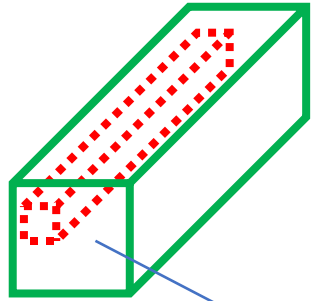
7 x 7 x 30

$$30 = 5 \times B + C$$

논문에선 B를 2, C를 20으로 설정



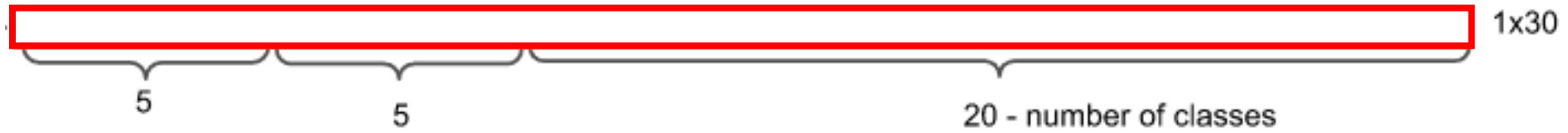
# 동작 알고리즘



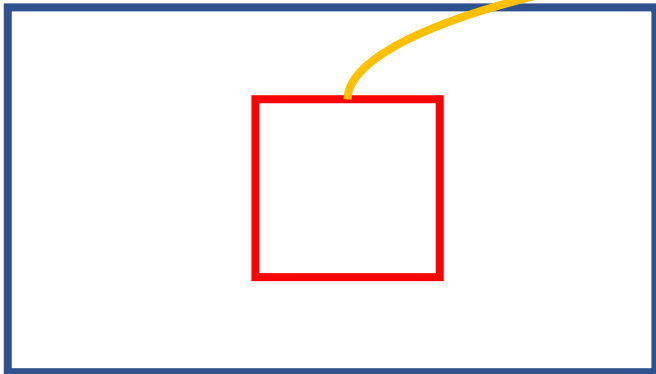
7 x 7 x 30

$$30 = 5 \times B + C$$

논문에선 B를 2, C를 20으로 설정

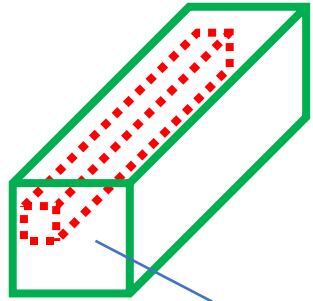


(x, y, w, h, c)



x = 0.5  
y = 0.5

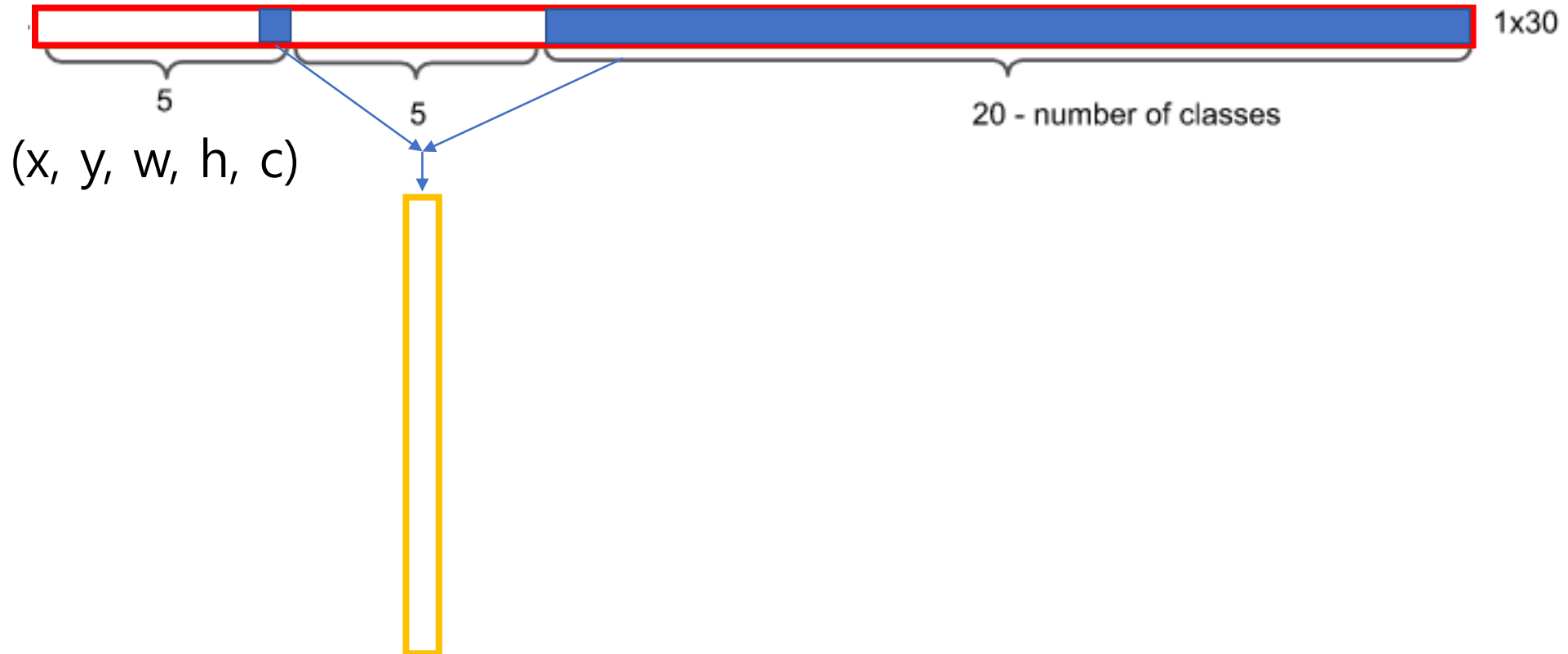
# 동작 알고리즘



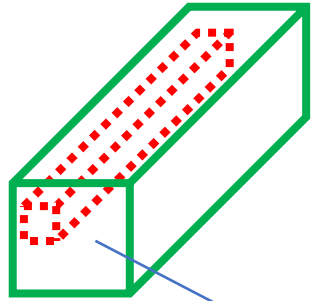
7 x 7 x 30

$$30 = 5 \times B + C$$

논문에선 B를 2, C를 20으로 설정



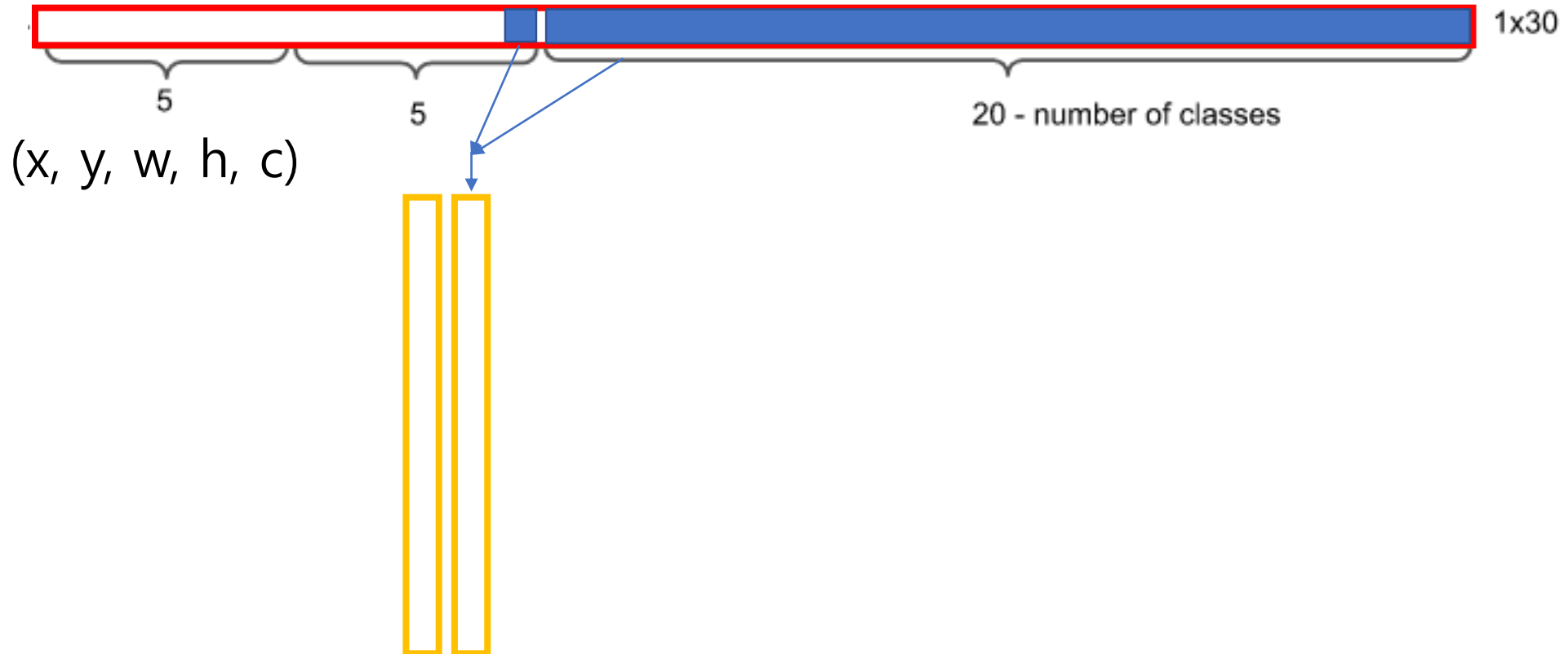
# 동작 알고리즘



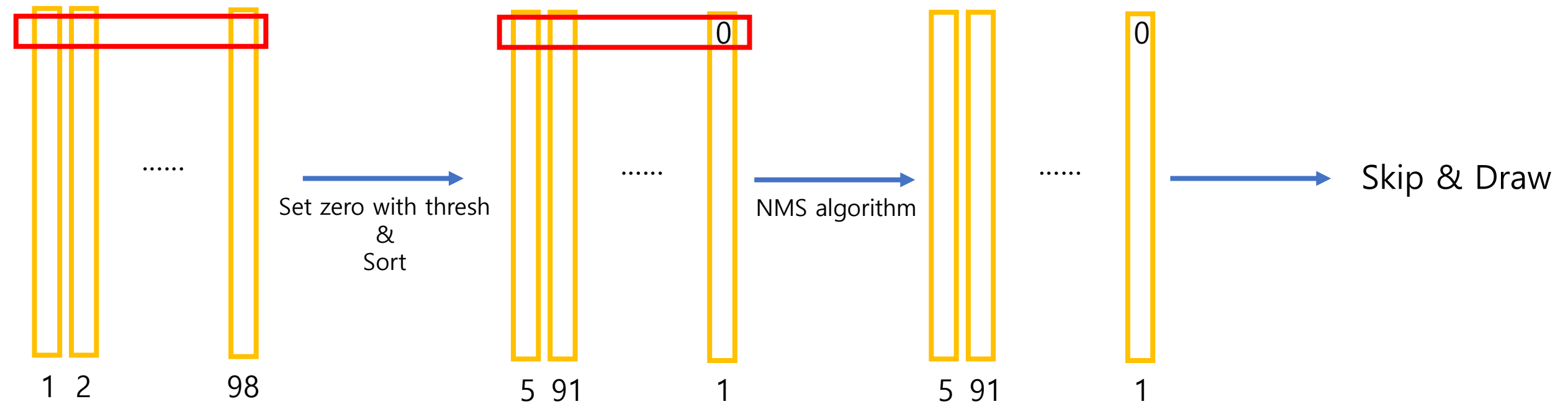
7 x 7 x 30

$$30 = 5 \times B + C$$

논문에선 B를 2, C를 20으로 설정

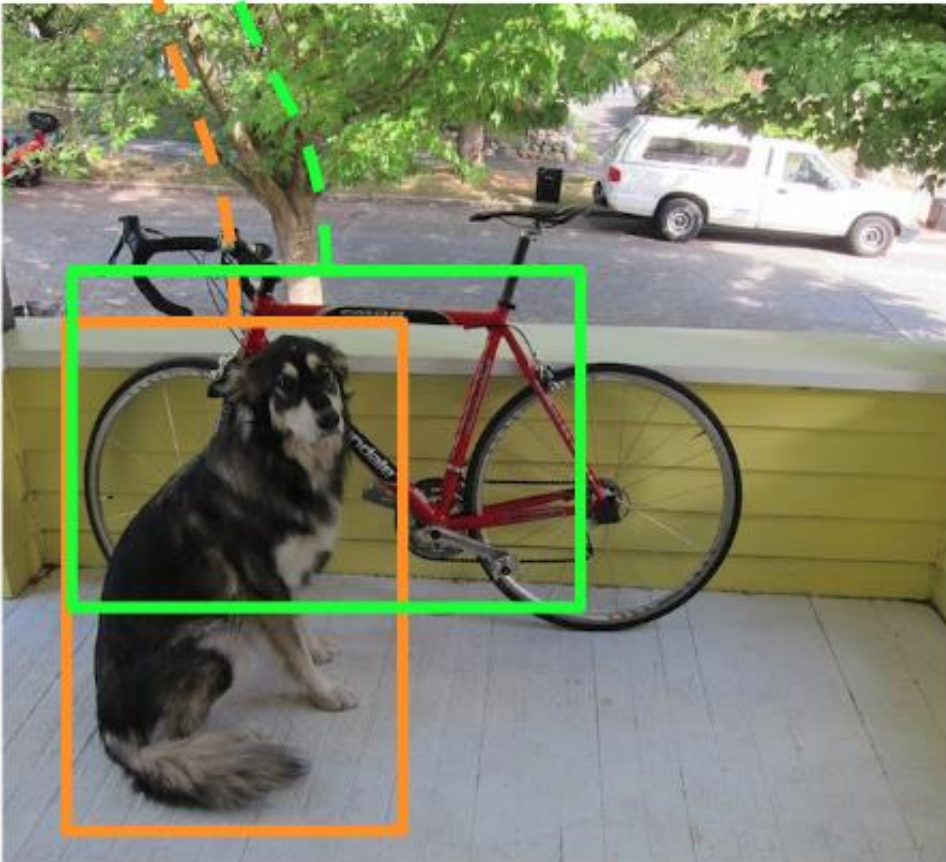


# 동작 알고리즘





# NMS algorithm

[illegible]

If  $\text{IoU}(\text{bbox\_max}, \text{bbox\_cur}) > 0.5$  then set 0 score to  $\text{bbox\_cur}$ .

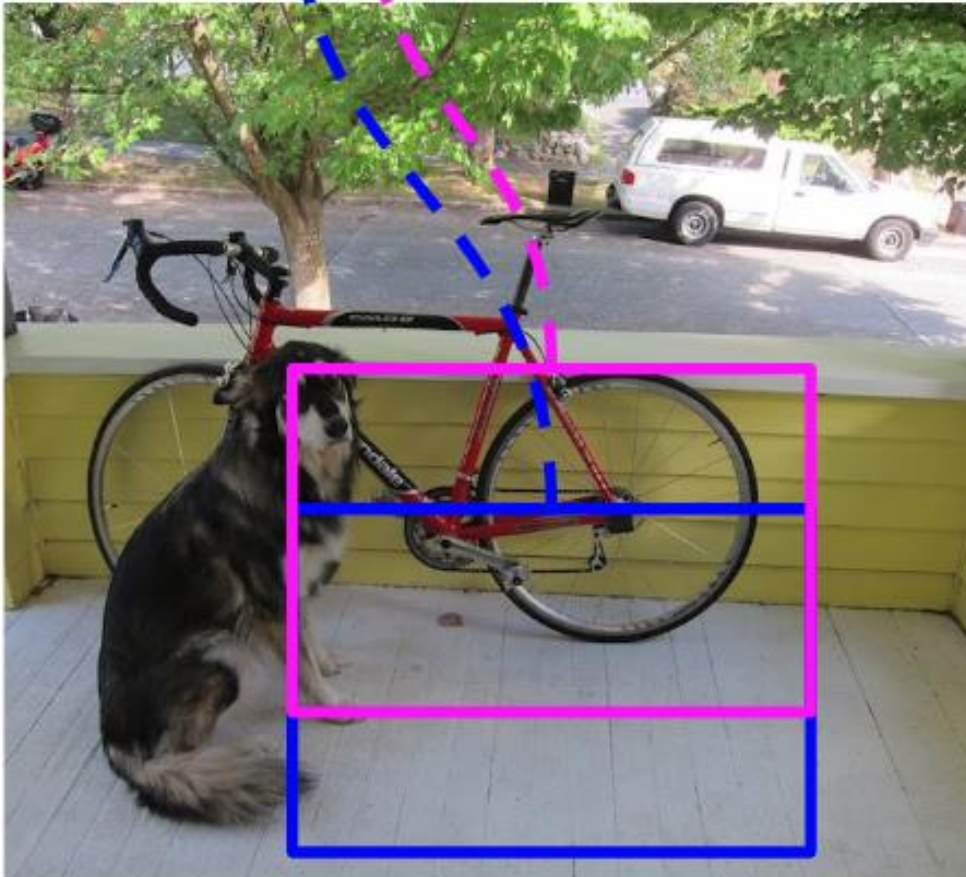
Max 값과 다른 모든 값을 비교해 검사한다.

# NMS algorithm

class (dog) scores for each bbox

class (dog) scores for each bbox																bb1	bb4	bb8	bb98
bb47	bb20	bb15	bb7																
0.5	0	0.2	0.1													0	0	0	0

1x98



Go to next **bbox\_cur**.

If  $\text{IoU}(\text{bbox\_max}, \text{bbox\_cur}) > 0.5$  then set 0 score to **bbox\_cur**.

그 다음 Max 값과 다른 모든 값을 비교해 검사한다.

# 정리

## 장점

- 전체 이미지를 input으로 한번만 받음.
- Regression과 Classification을 동시에 처리하기 때문에 빠르다.

## 단점

- 전체 이미지를 SxS grid cell로 나누어서 object를 찾기 때문에 작은 물체에 대해 약하다.
- 1개 grid에 2개 Bbox, 1개 class를 예측하기 때문에 밀집되어 있는 경우 성능이 약하다.