

# Deep Learning Study











---

2020/10/06

Byungchan Lee



# 딥러닝 4단계 : 합성곱 신경망 네트워크(CNN)

탐지 알고리즘	
	물체 로컬리제이션 업데이트 : 2020.07.14   ♥ 7
	특징점 탐지 업데이트 : 2020.07.14   ♥ 6
	물체 인식 (Object Detection) 업데이트 : 2020.07.14   ♥ 5
	합성곱으로 슬라이딩 윈도우 구현하기 업데이트 : 2020.07.14   ♥ 5
	경계 상자 (Bounding Boxes) 예측하기 업데이트 : 2020.07.14   ♥ 5
	합집합 위의 교집합 (IOU) 업데이트 : 2020.07.15   ♥ 3
	비-최대값 억제 (Nonmax Suppression) 업데이트 : 2020.07.15   ♥ 5
	앵커 박스 (Anchor Boxes) 업데이트 : 2020.07.15   ♥ 4
	YOLO 알고리즘 업데이트 : 2020.07.15   ♥ 6
	지역 제안 알고리즘 (Optional) 업데이트 : 2020.07.15   ♥ 5

# (last week) Object localization

## Defining the target label $y$

- 1 - pedestrian
- 2 - car ←
- 3 - motorcycle
- 4 - background ←

Need to output  $b_x, b_y, b_h, b_w$ , class label (1-4)

$$y = \begin{bmatrix} p_c \\ b_x \\ b_y \\ b_h \\ b_w \\ c_1 \\ c_2 \\ c_3 \end{bmatrix}$$

is there any  
object?

$x =$



$$\begin{bmatrix} 1 \\ b_x \\ b_y \\ b_h \\ b_w \\ 0 \\ 0 \\ 0 \end{bmatrix}$$



$$\begin{bmatrix} 0 \\ \sim \\ \sim \\ \sim \\ \sim \\ \sim \\ \sim \\ \sim \end{bmatrix}$$

← "don't  
care"

Andrew Ng

# Landmark Detection



$$\begin{bmatrix} 1 \\ y_{l1} \\ y_{l2} \\ \vdots \\ \vdots \end{bmatrix}$$



# Object Detection

## Car detection example



Use closely cropped images  
as training examples

Andrew Ng

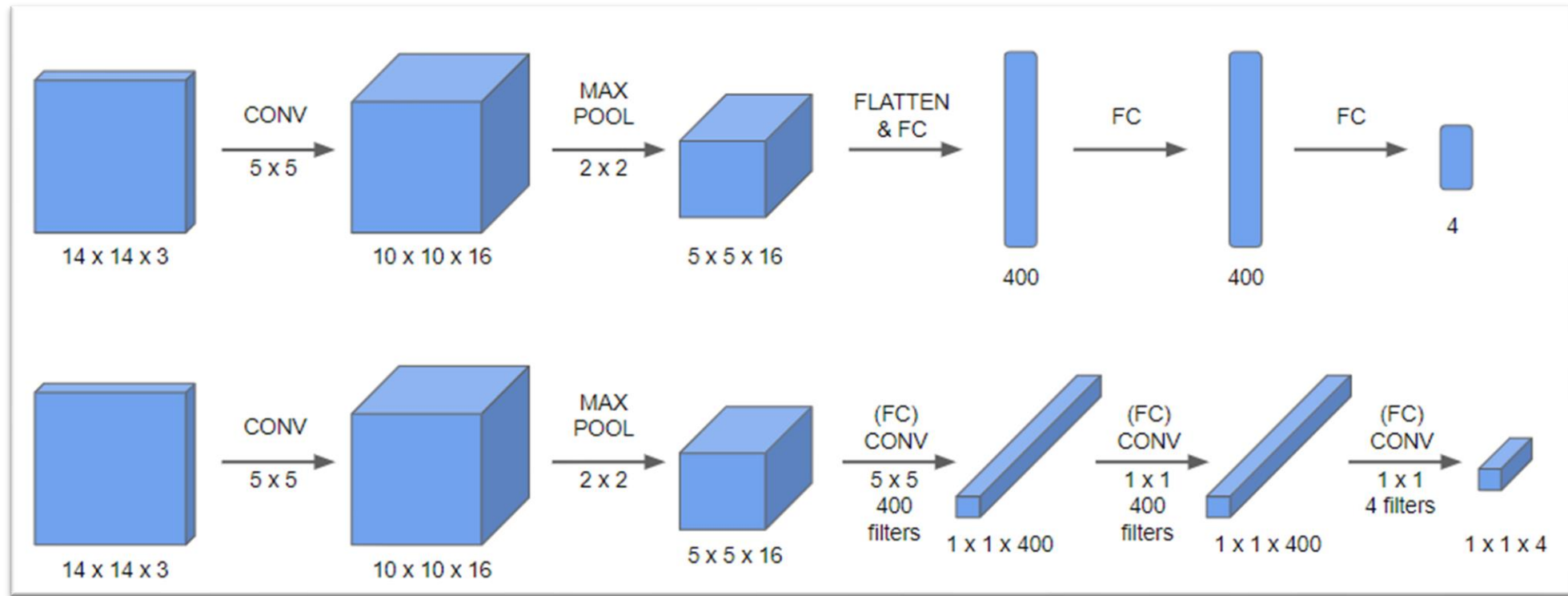
# Sliding windows detection



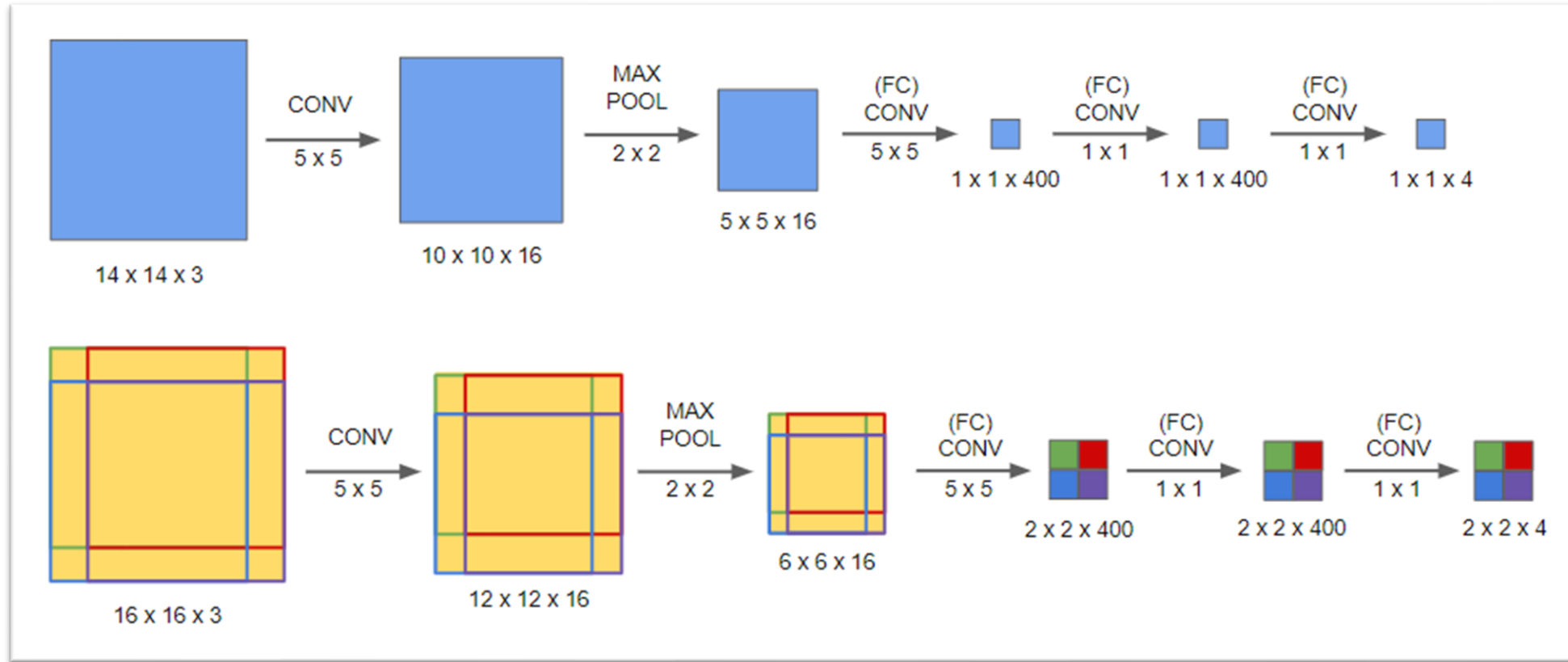
- Disadvantage : High computational cost (slow..)
- People used to use much simpler classifier; Linear classifier



# Sliding windows detection

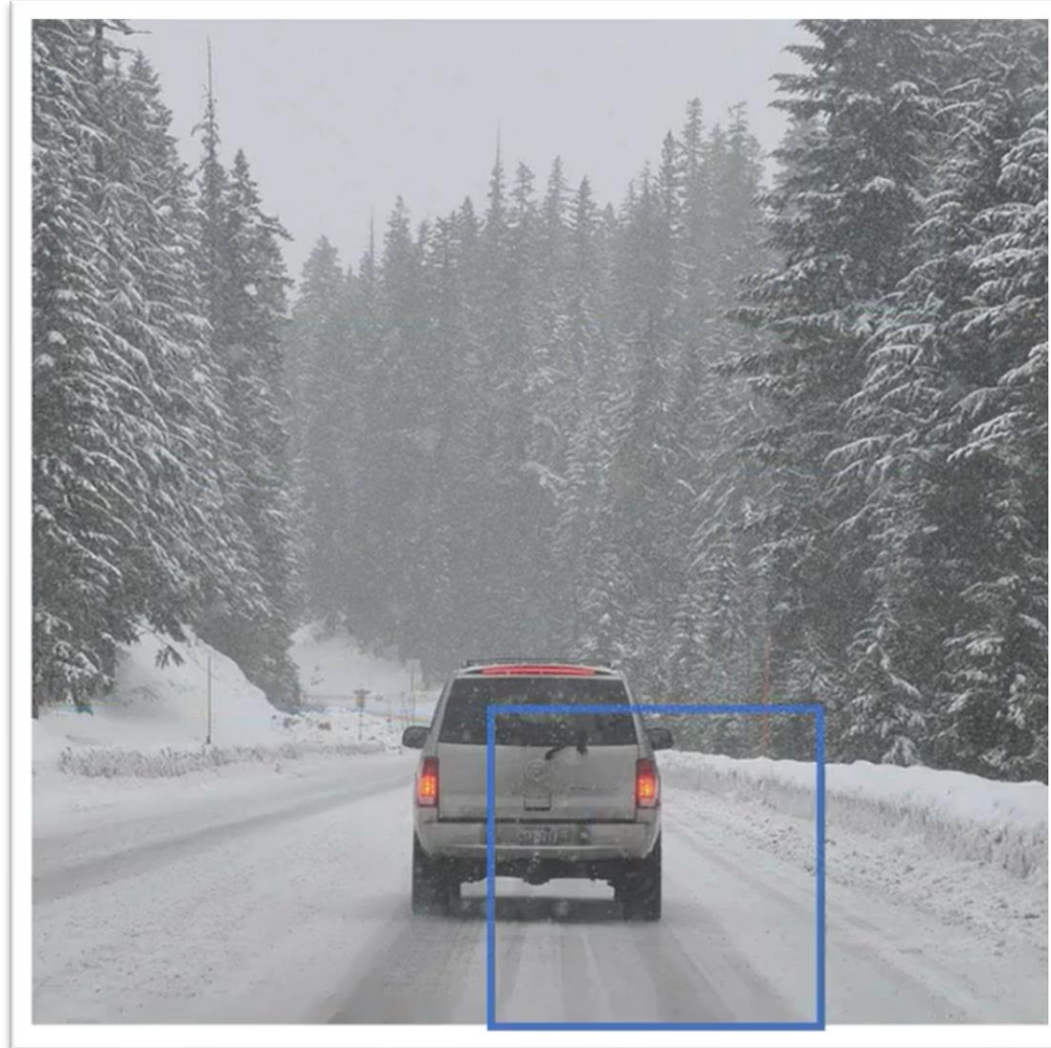


# Sliding windows detection



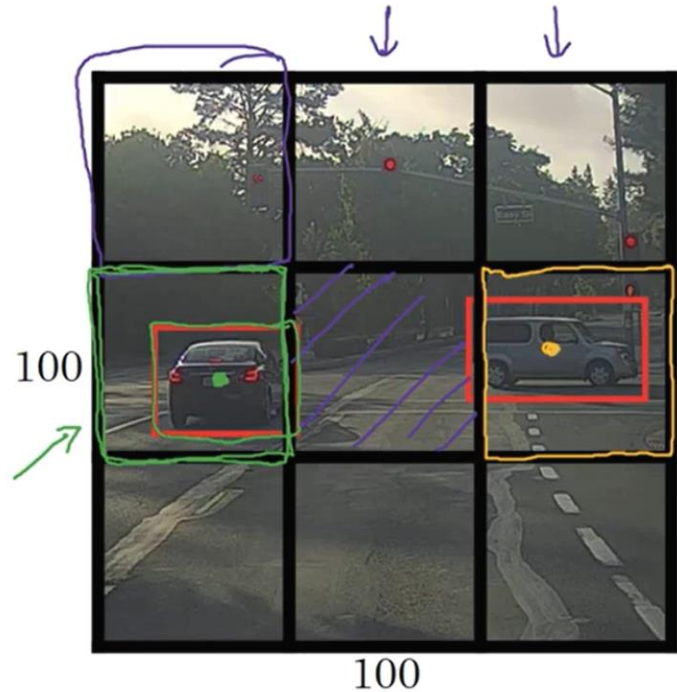


# Bounding boxes



# YOLO (You Only Look Once)

## YOLO algorithm



Labels for training  
For each grid cell:

$$y = \begin{bmatrix} p_c \\ b_x \\ b_y \\ b_h \\ b_w \\ c_1 \\ c_2 \\ c_3 \end{bmatrix} \rightarrow \begin{bmatrix} 0 \\ ? \\ ? \\ ? \\ ? \\ ? \\ ? \\ ? \end{bmatrix}$$

$$\begin{bmatrix} 1 \\ b_x \\ b_y \\ b_h \\ b_w \\ 0 \\ 1 \\ 0 \end{bmatrix} \quad \begin{bmatrix} 1 \\ b_x \\ b_y \\ b_h \\ b_w \\ 0 \\ 1 \\ 0 \end{bmatrix}$$

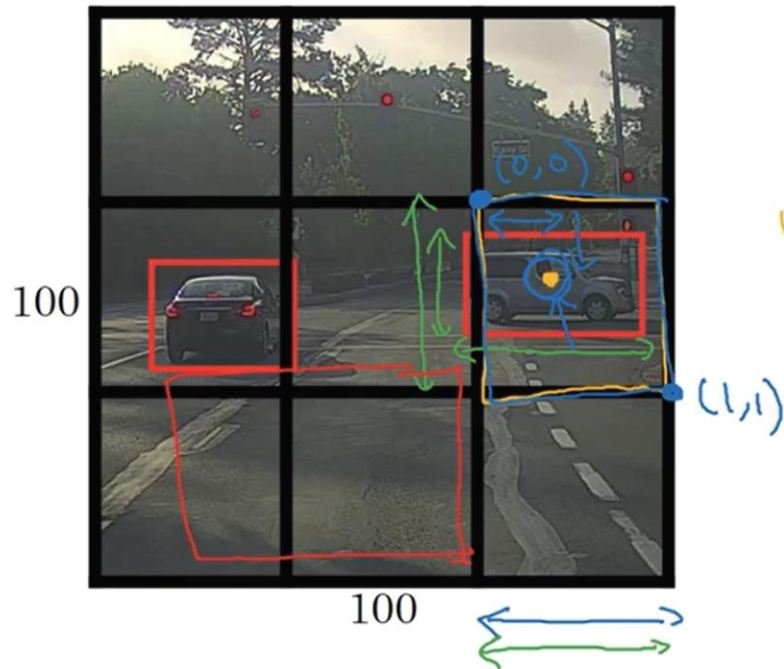
[1506.02640] You Only Look Once: Unified, Real-Time Object Detection

[Redmon et al., 2015, You Only Look Once: Unified real-time object detection]

Andrew Ng

# YOLO (You Only Look Once)

## Specify the bounding boxes



$$y = \begin{bmatrix} 1 \\ b_x \\ b_y \\ b_h \\ b_w \\ 0 \\ 1 \\ 0 \end{bmatrix}$$

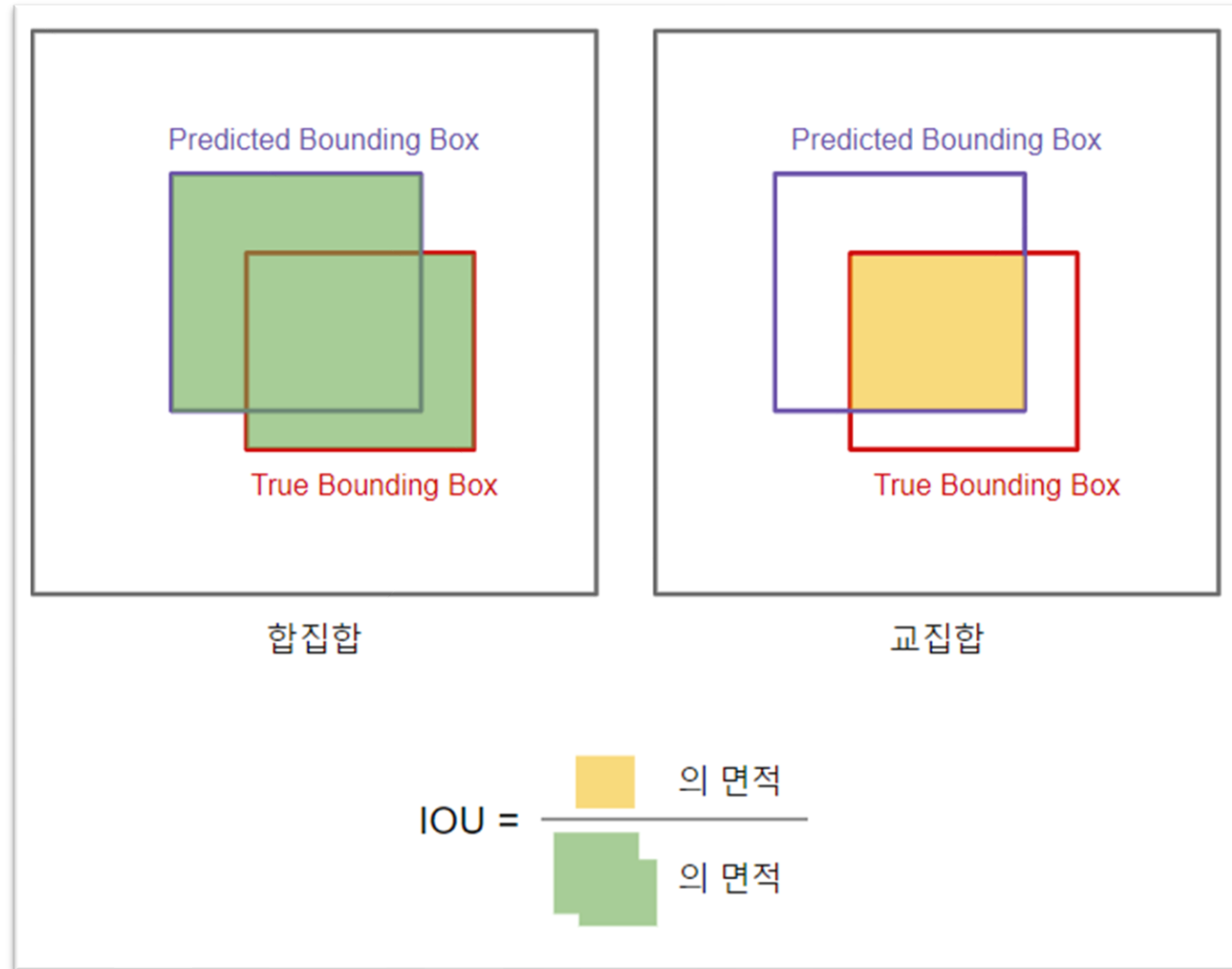
0.4 } between 0 and 1  
0.3 }  
0.9 } could be > 1  
0.5 }

- One reasonable convention..
- There are other parametrizations that work better

[Redmon et al., 2015, You Only Look Once: Unified real-time object detection]

Andrew Ng

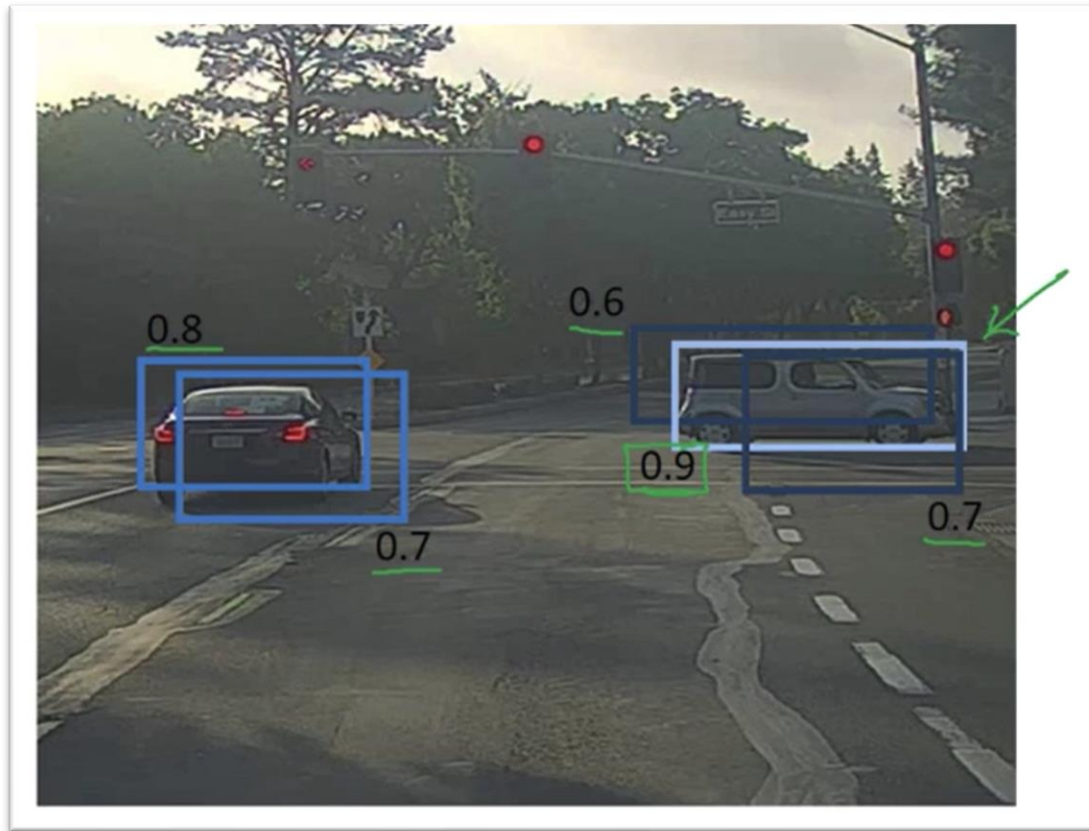
# IOU (Intersection Over Union)



Convention

- $\text{IOU} \geq 0.5 \rightarrow \text{Correct!}$

# Non-max suppression



- Discard all boxes with  $p_c \leq 0.6$
- While there are any remaining boxes
  - Pick the box with the largest  $p_c$
  - Discard boxes with  $\text{IoU} \geq 0.5$  with box output in the previous step