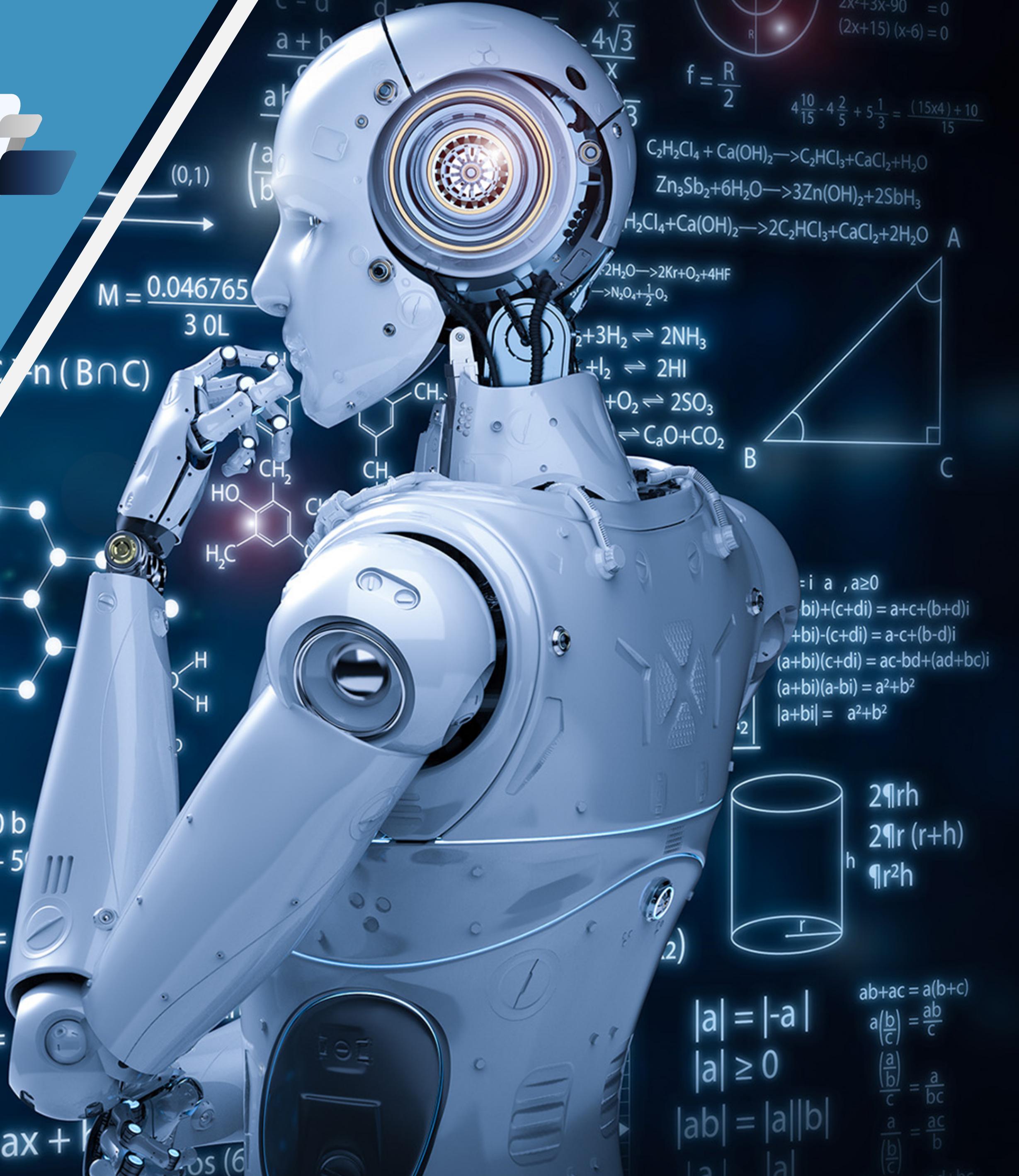


# Day 33

## 深度學習與電腦視覺 學習馬拉松

cupay 陪跑專家：陳穗碧



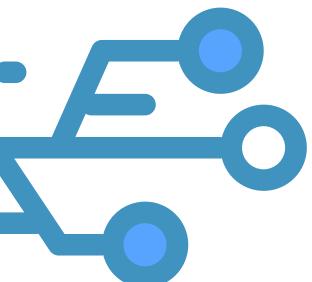


# YOLO 細節理解 - 網路輸出的後處理

# 重要知識點



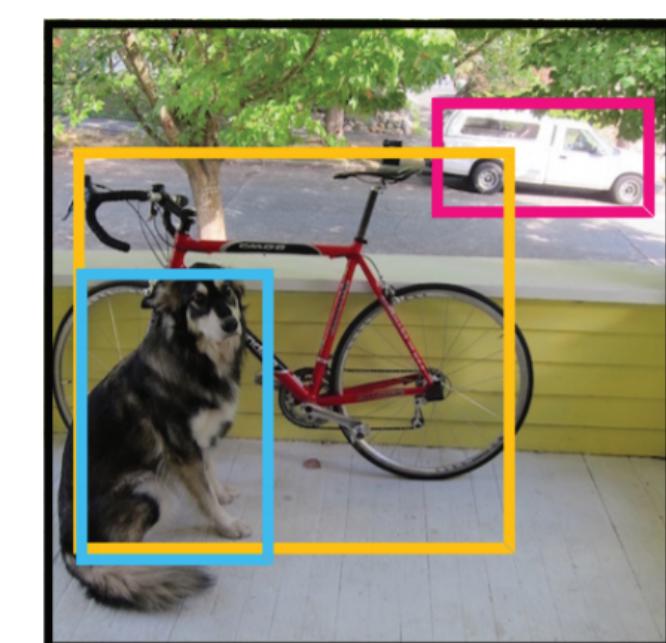
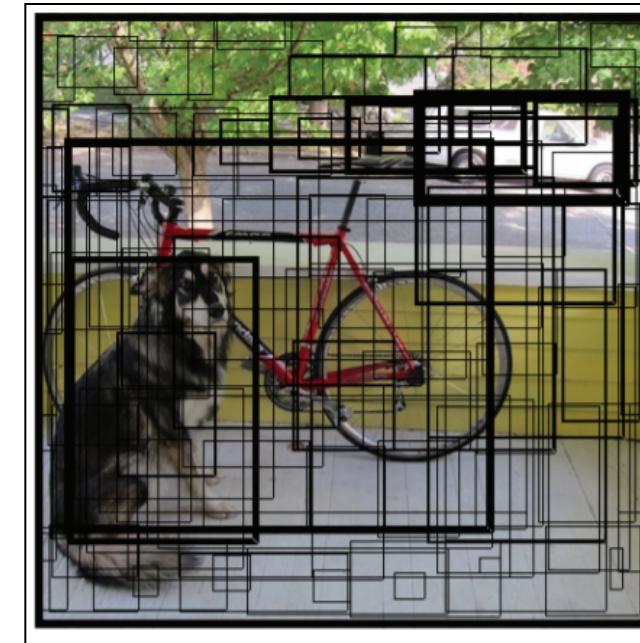
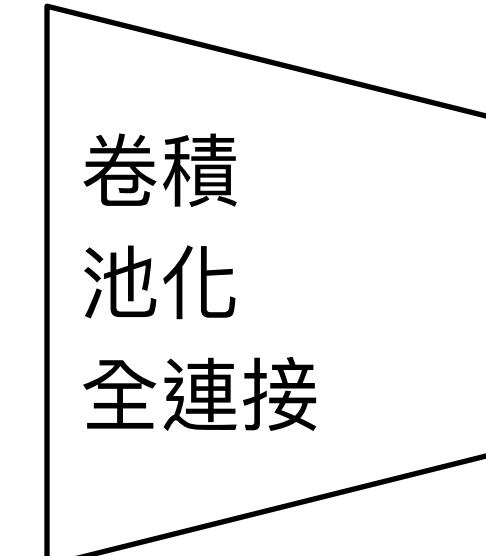
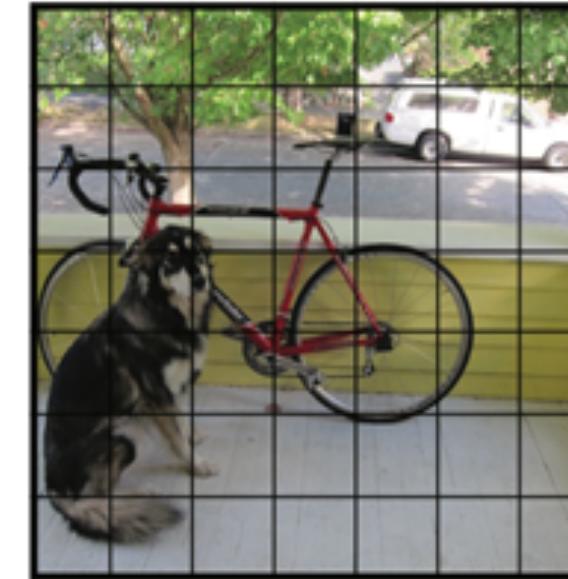
- YOLO 輸出層轉換成 bbox 資訊的過程
- NMS 在 YOLO 的實際運作
- NMS 在 YOLO 中採用的信心度為何



# YOLO 模型偵測流程

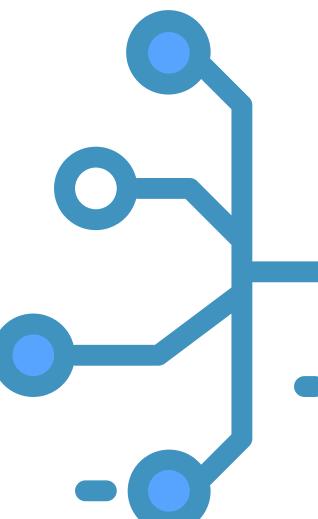


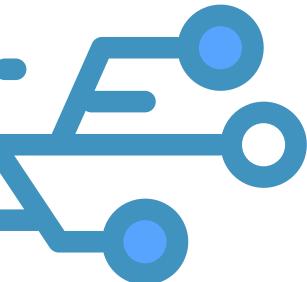
YOLO 模型資料流包含以下 7 個步驟，



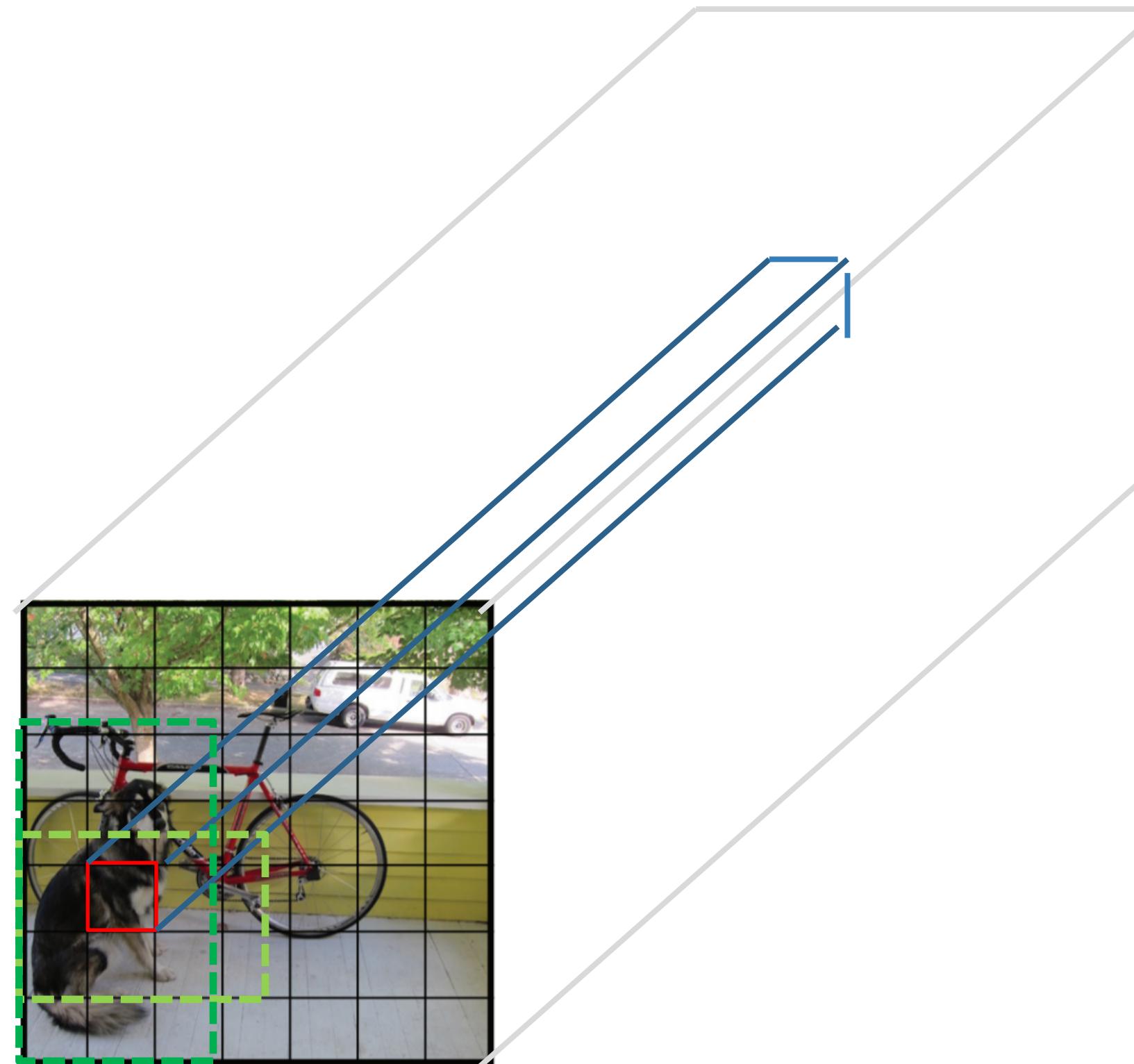
YOLO 特殊輸出層的設計，能讓一個 grid cell 預測一個物體

然而，會遇到一個物體有多個預測框的情形，例如輸出層中有很多框包圍住狗，需透過 NMS 演算法來挑選出最佳物體預測框（藍色框）。





# 回顧 YOLOv1 輸出層

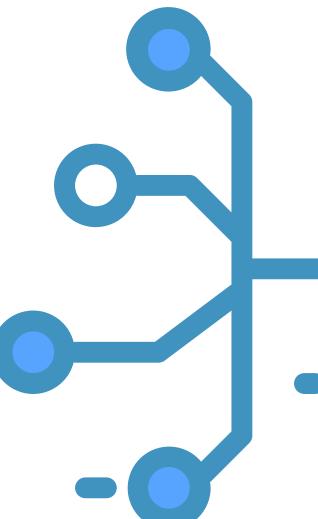


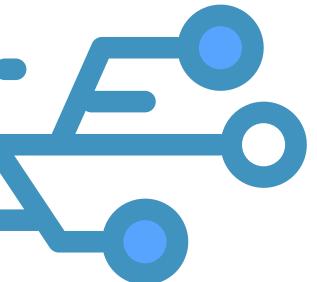
圖片切成  $7 \times 7$  格，以  $i=(5,2)$  位置為例，  
模型預測 20 個類別，每格有兩個預測框  
(bbox)，其輸出層大小  $T_i$  為  $(2 \times 5 + 20)$ ，  
包含兩個預測框的參數  $(bbox_1, bbox_2)$  以  
及預測是屬於哪一個類別的機率  $p_i(c)$  所需  
的資訊，輸出層輸出資訊如下：

$$T_i = [T_{i,bbox1}, T_{i,bbox2}, T_{i,p_i(c)}]$$

$$T_{i,bbox1} = [T_{i,b1,x}, T_{i,b1,y}, T_{i,b1,w}, T_{i,b1,h}, T_{i,c}]$$

這樣的輸出層設計，限制一個 grid cell 預測一個物體





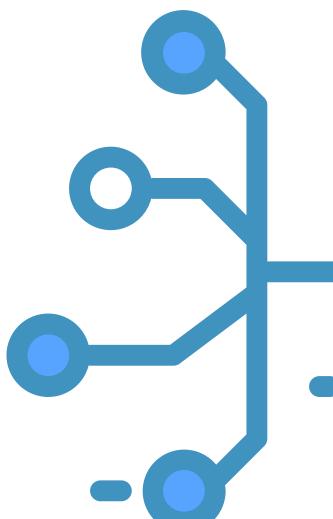
# YOLO 輸出層轉換成 bbox 資訊

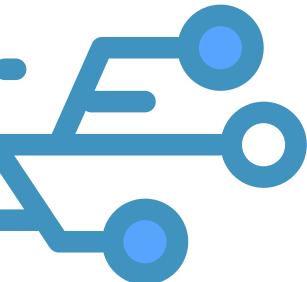


我們所需要的每一個格點 i 包含兩個 bbox 的資訊如下：

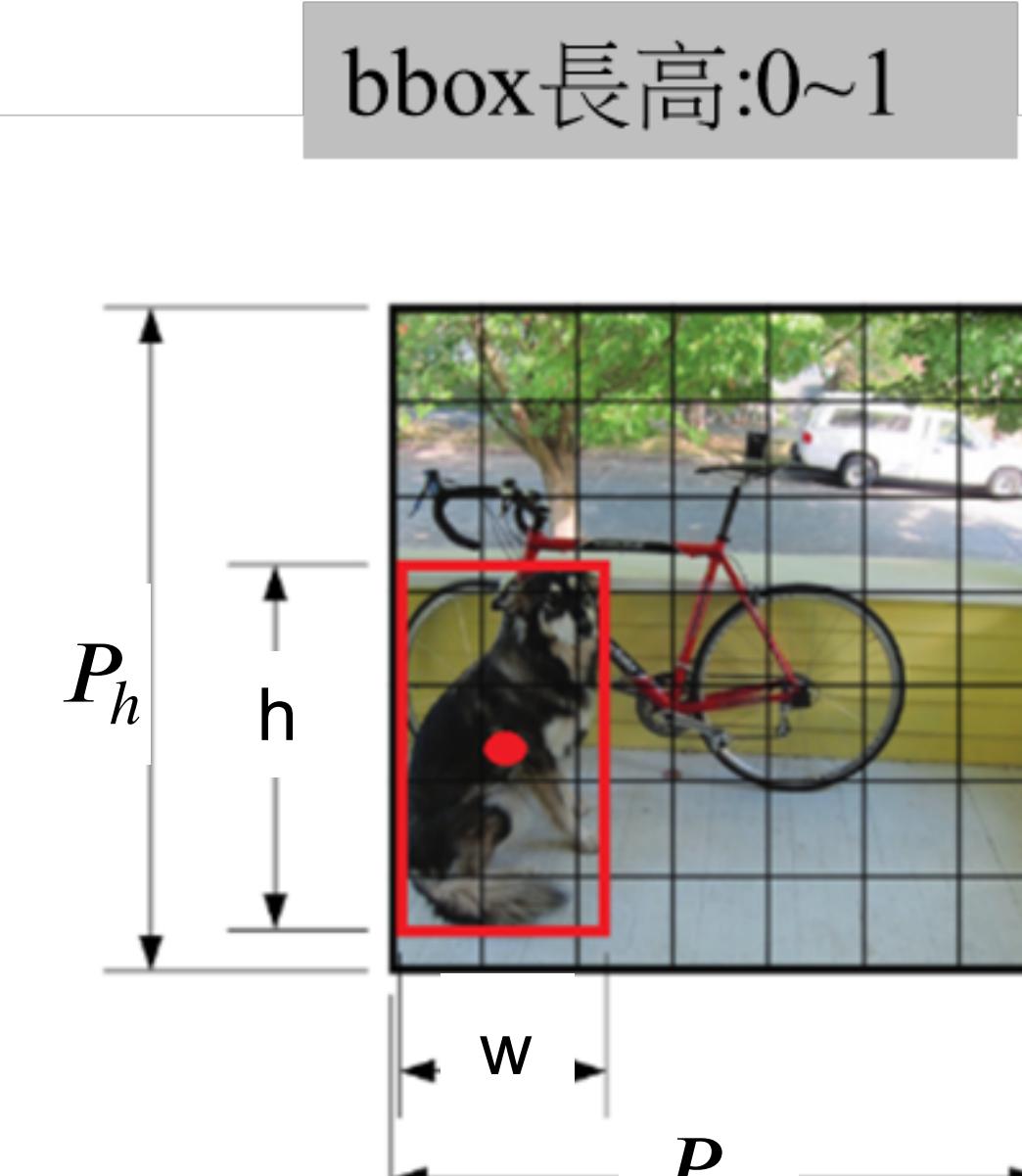
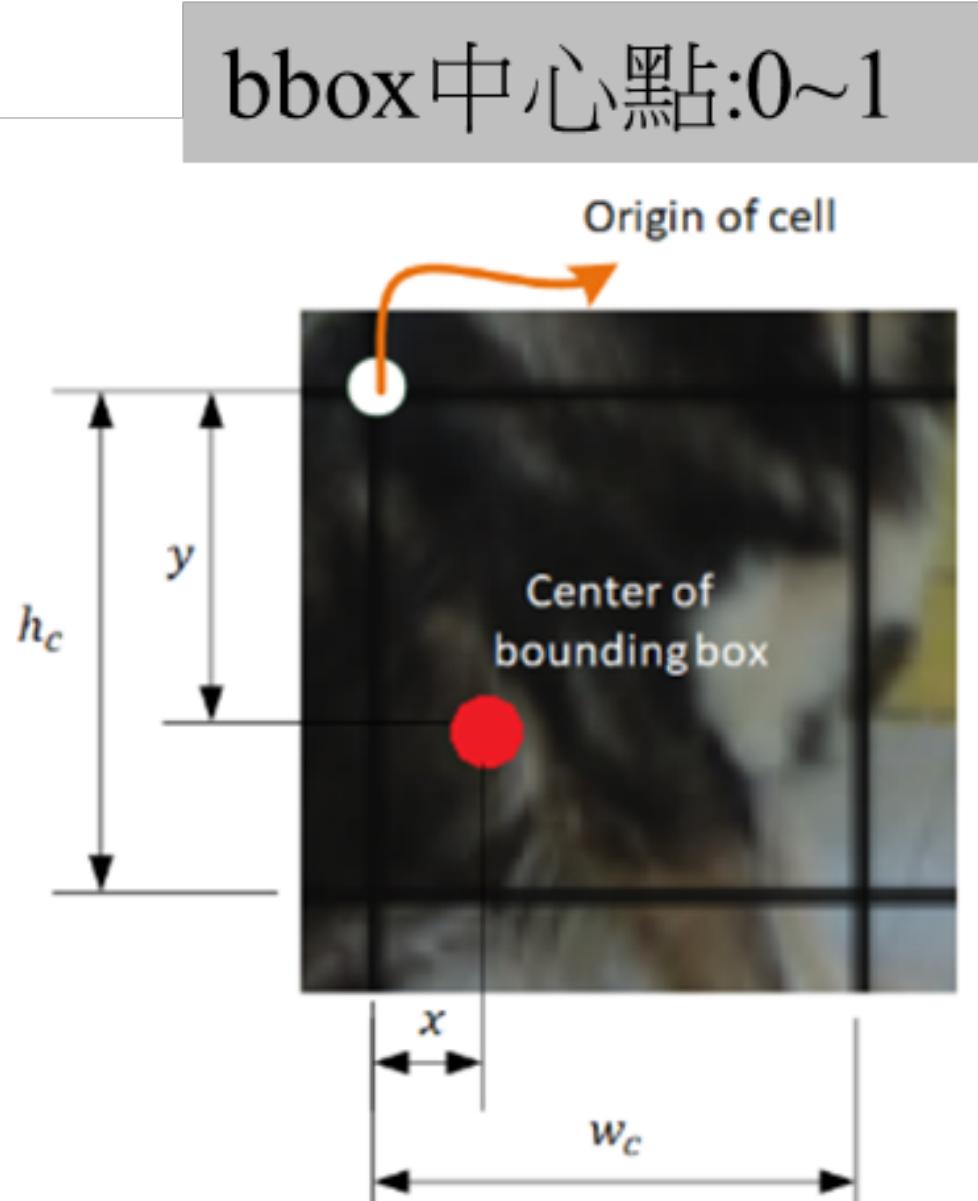
x	y	w	h		x	y	w	h		
bbox1 中心x	bbox1 中心y	寬	高	包含物體 信心度	bbox2 中心x	bbox2 中心y	寬	高	包含物體 信心度	類別機率

怎麼透過輸出層轉換出來？

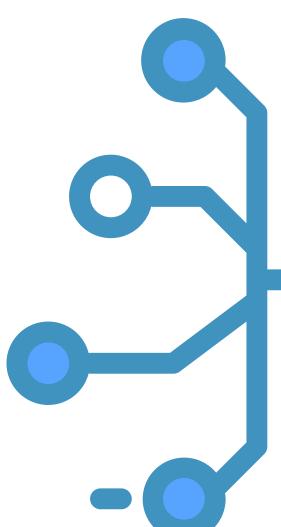


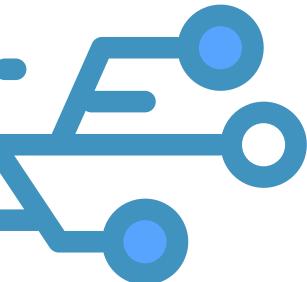


# YOLO 輸出層轉換成 bbox 資訊

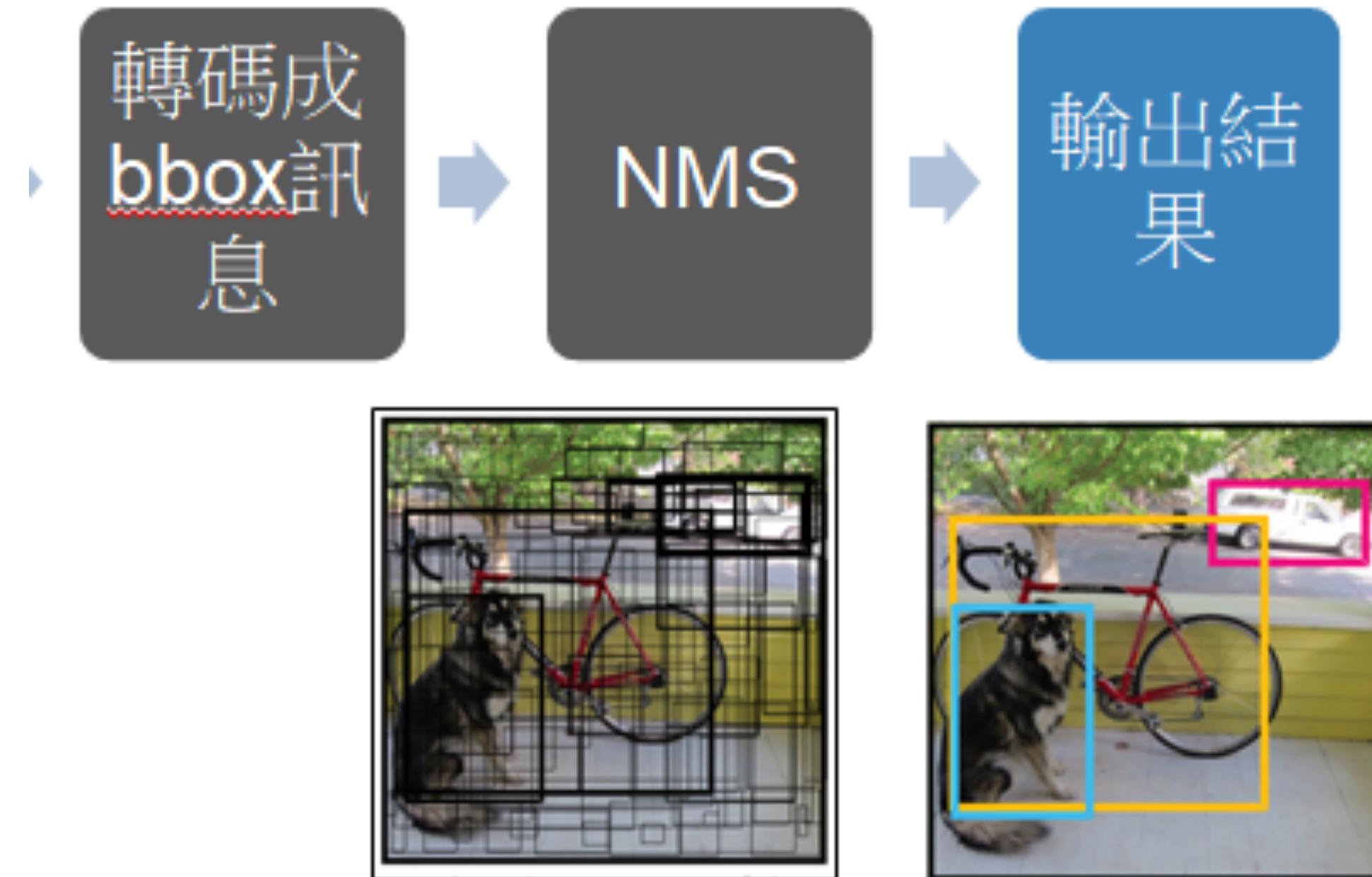


- 圖片切成  $S \times S$  格，格點i的左上方的位置為  $(C_x, C_y)$ ，圖片輸入寬高為  $P_w, P_h$
- $w_c$  為每一個格欄的寬  $= \frac{P_w}{S}$
- $h_c$  為每一個格欄的高  $= \frac{P_h}{S}$
- $x = (\text{sigmoid}(T_{i,b1,x}) + C_x) \times w_c$
- $y = (\text{sigmoid}(T_{i,b1,y}) + C_y) \times h_c$
- $w = (\text{sigmoid}(T_{i,b1,w})) \times P_w$
- $h = (\text{sigmoid}(T_{i,b1,h})) \times P_h$
- $c = \text{sigmoid}(T_{i,b1,c})$
- $\text{class} = \text{softmax}(T_{i,p_i(c)})$



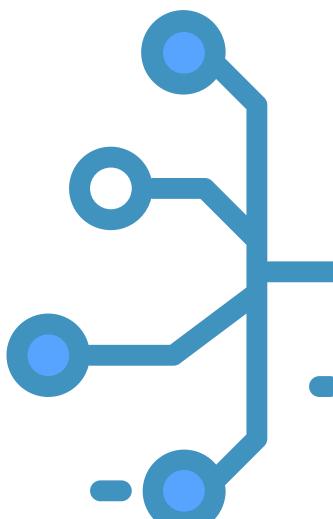


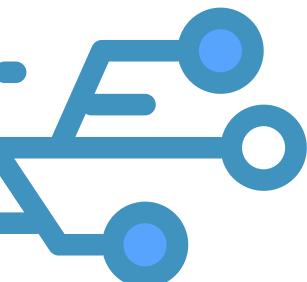
# 轉碼後得出很多 bbox 後的下一步-NMS



YOLO 特殊輸出層的設計，能讓一個 grid cell 預測一個物體。

然而，會遇到一個物體有多個預測框的情形，例如輸出層中有很多框包圍住狗，需透過 NMS 演算法來挑選出最佳物體預測框（藍色框）。





# NMS 回顧



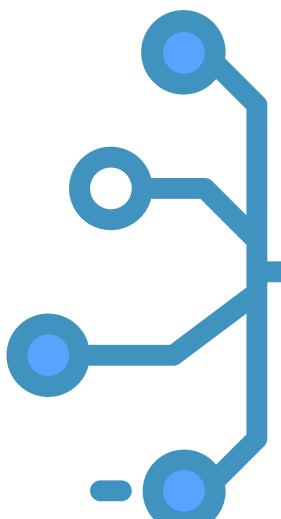
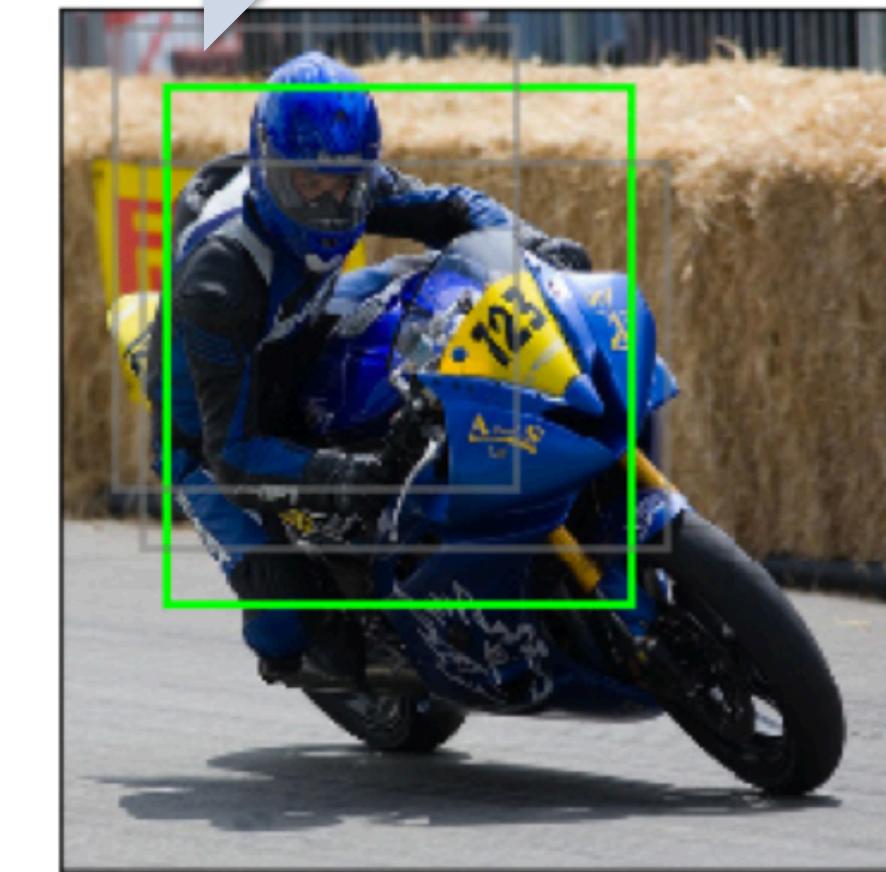
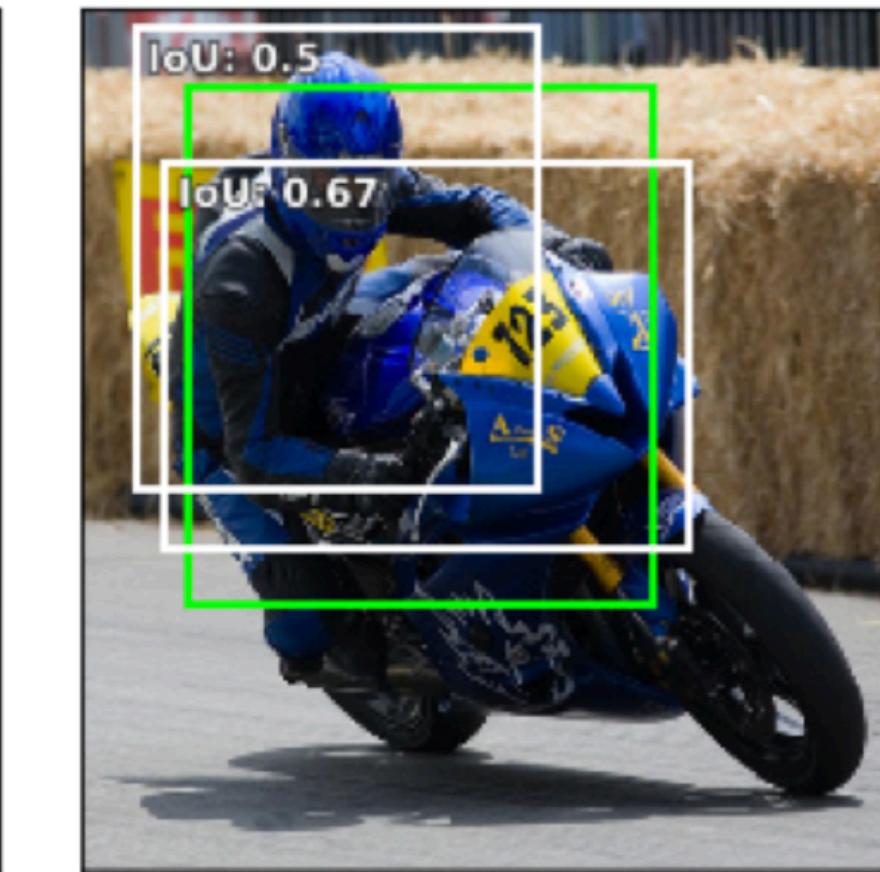
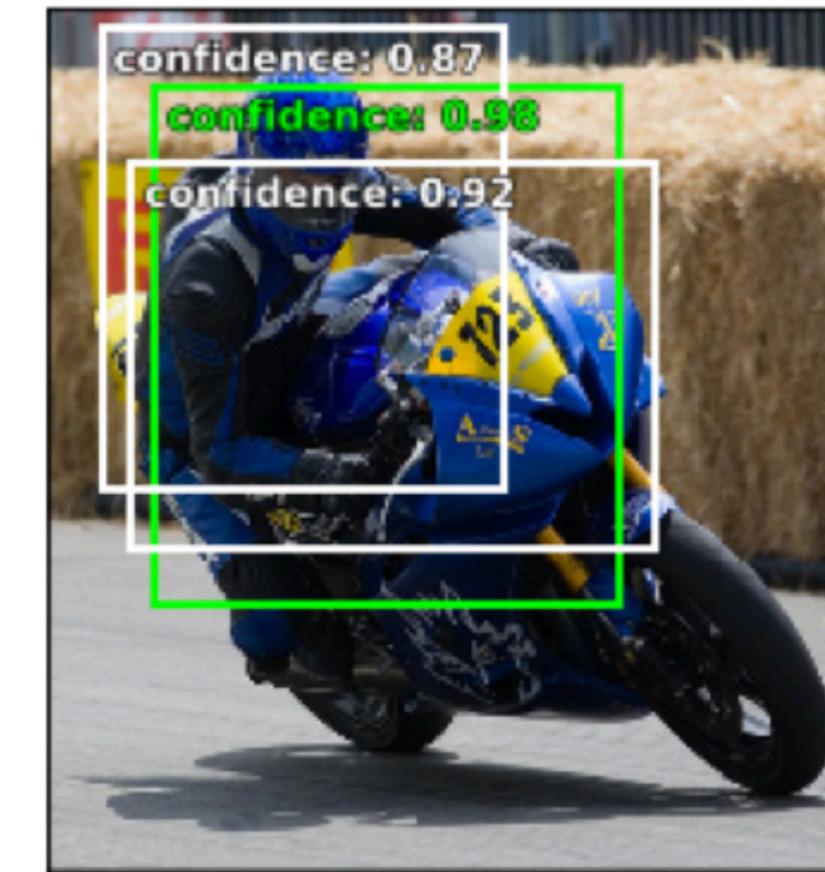
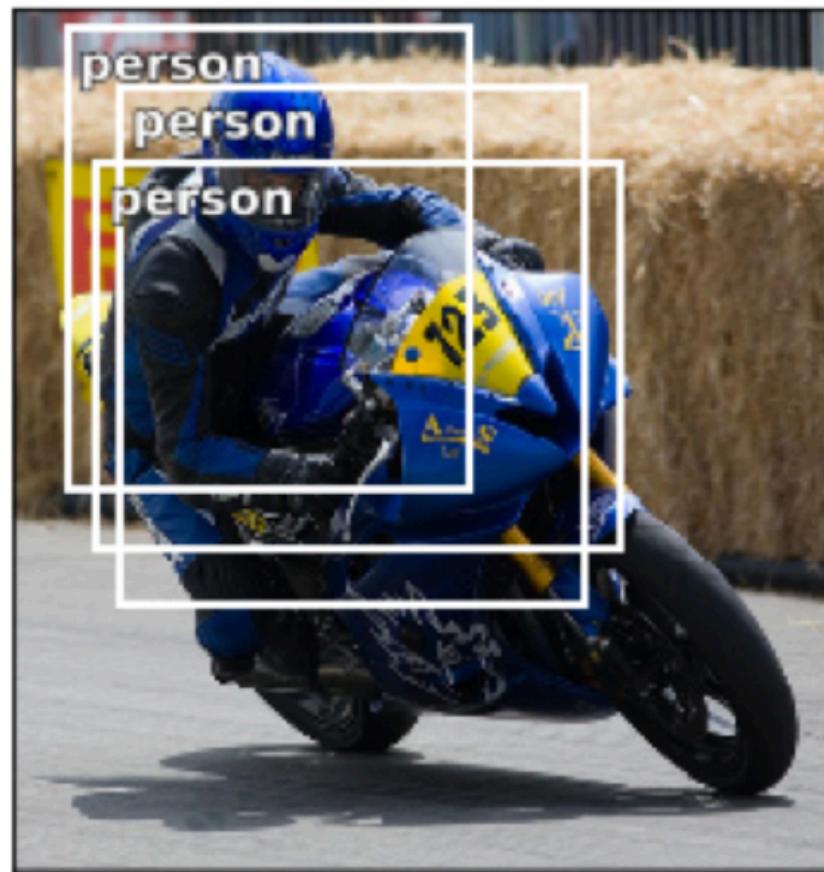
- Day28天學習了NMS演算法，包含四個步驟

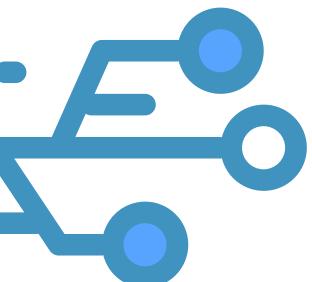
過濾掉**信心度**較低的預測框

選定**信心程度**最高的預測框，並計算其他預測框與選定預測框的**重疊率(IoU)**

當其他預測框與選定預測框重疊率(IoU)大於一定值(**nms\_threshold**)時，將此框移除。

得到第一個選定框，此時再**重複第一步**(選定框不在計算內)，直到無法再重複為止。





# NMS 回顧



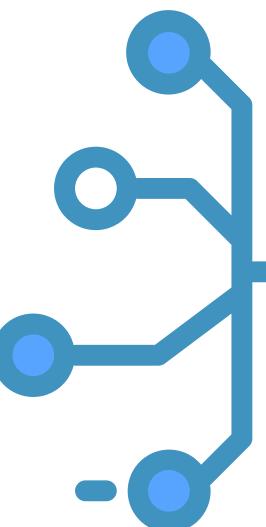
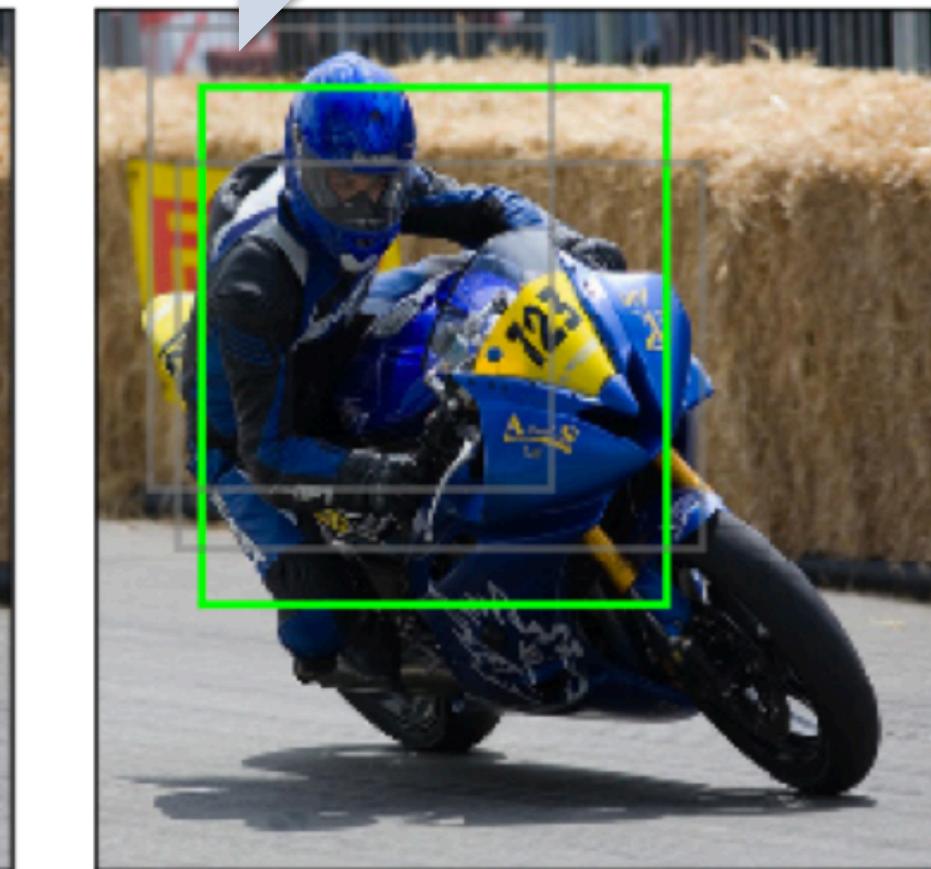
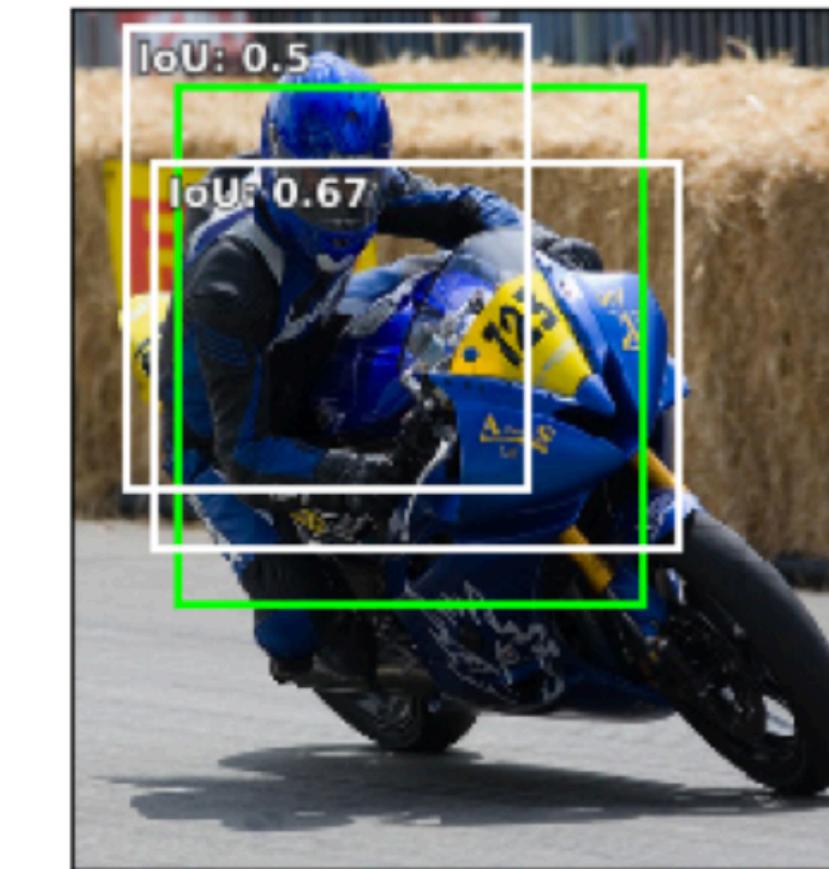
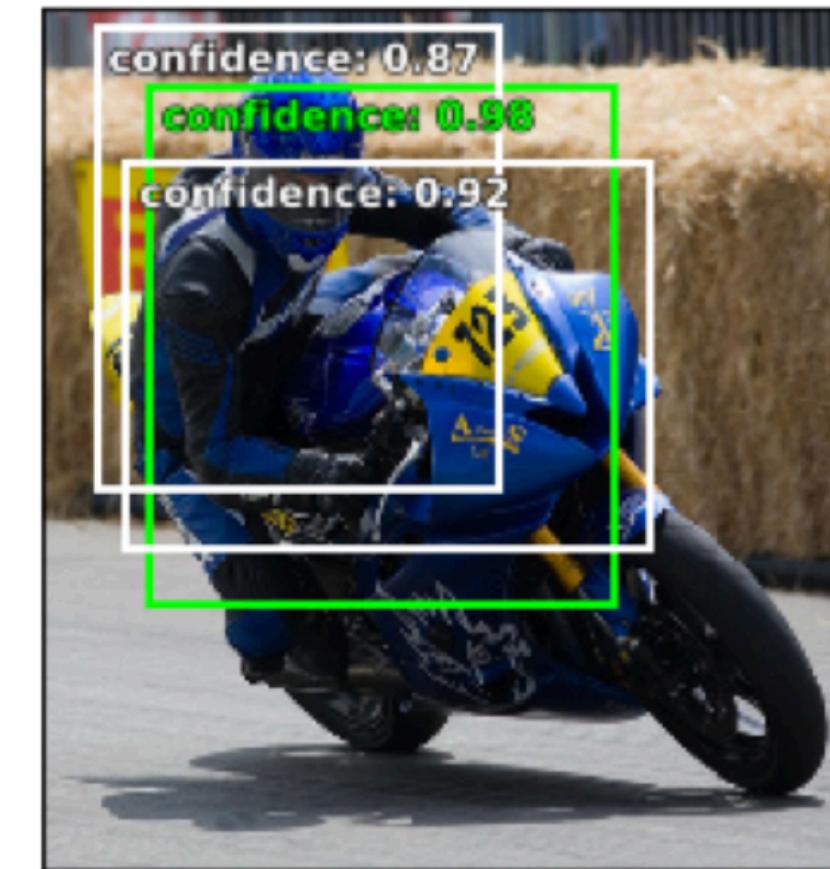
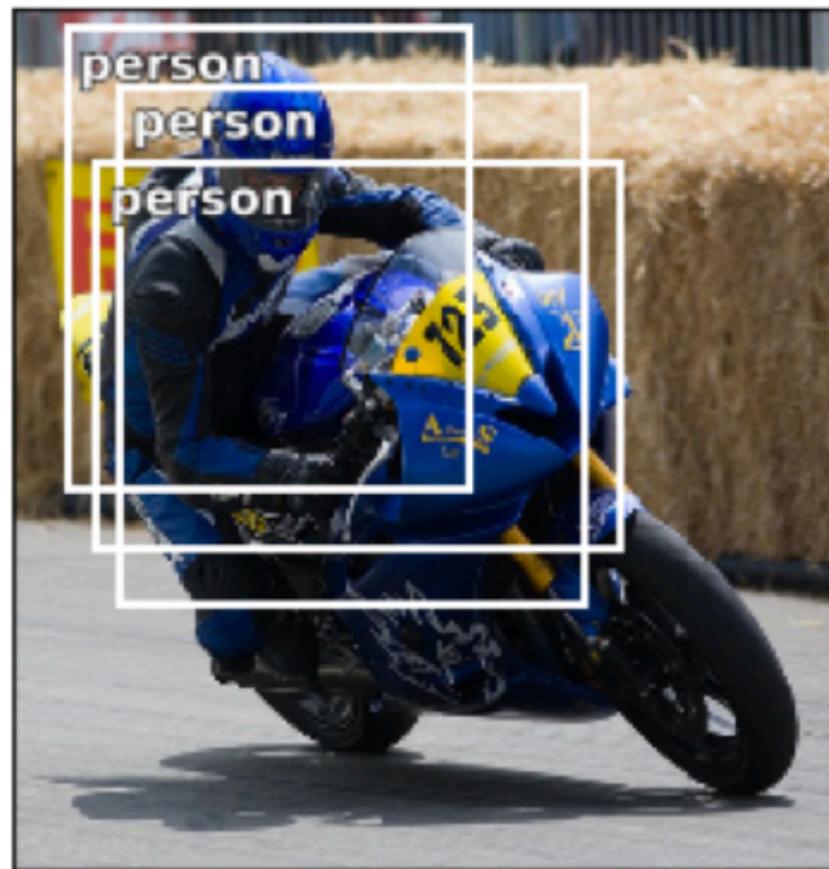
Day28天學習了NMS演算法，包含四個步驟

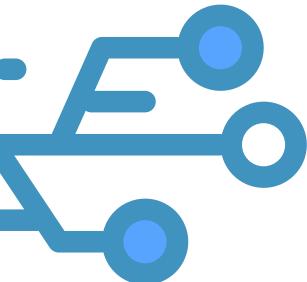
過濾掉**信心度**較低的預測框

選定**信心程度最高**的預測框，並計算其他預測框與選定預測框的**重疊率(IoU)**

當其他預測框與選定預測框重疊率(IoU)大於一定值(**nms\_threshold**)時，將此框移除。

得到第一個選定框，此時再**重複第一步**(選定框不在計算內)，直到無法再重複為止。





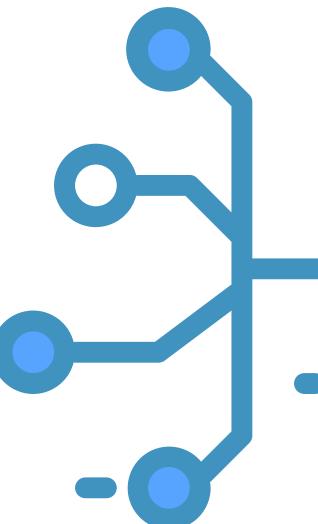
# YOLO 於 NMS 中使用的信心度

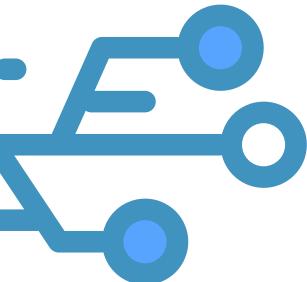


YOLO於NMS中採用的信心度為「每個bbox包含各類別的信心度」，這些信心分數同時量化此各個類別物件存在於bbox的可能性以及預測的bbox大小是否適合該物件。

- 每個bbox包含各類別的信心度
- $$p_i(Class_j) \times IOU_{truth} = p(Class_j | object) \times P(object) \times IOU_{truth}$$
$$= p(Class_j | object) \times C_i$$

因為你得確保 cell中有物體(confidence大)，計算類別機率才有意義。





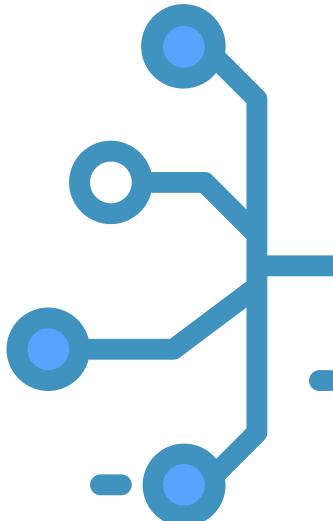
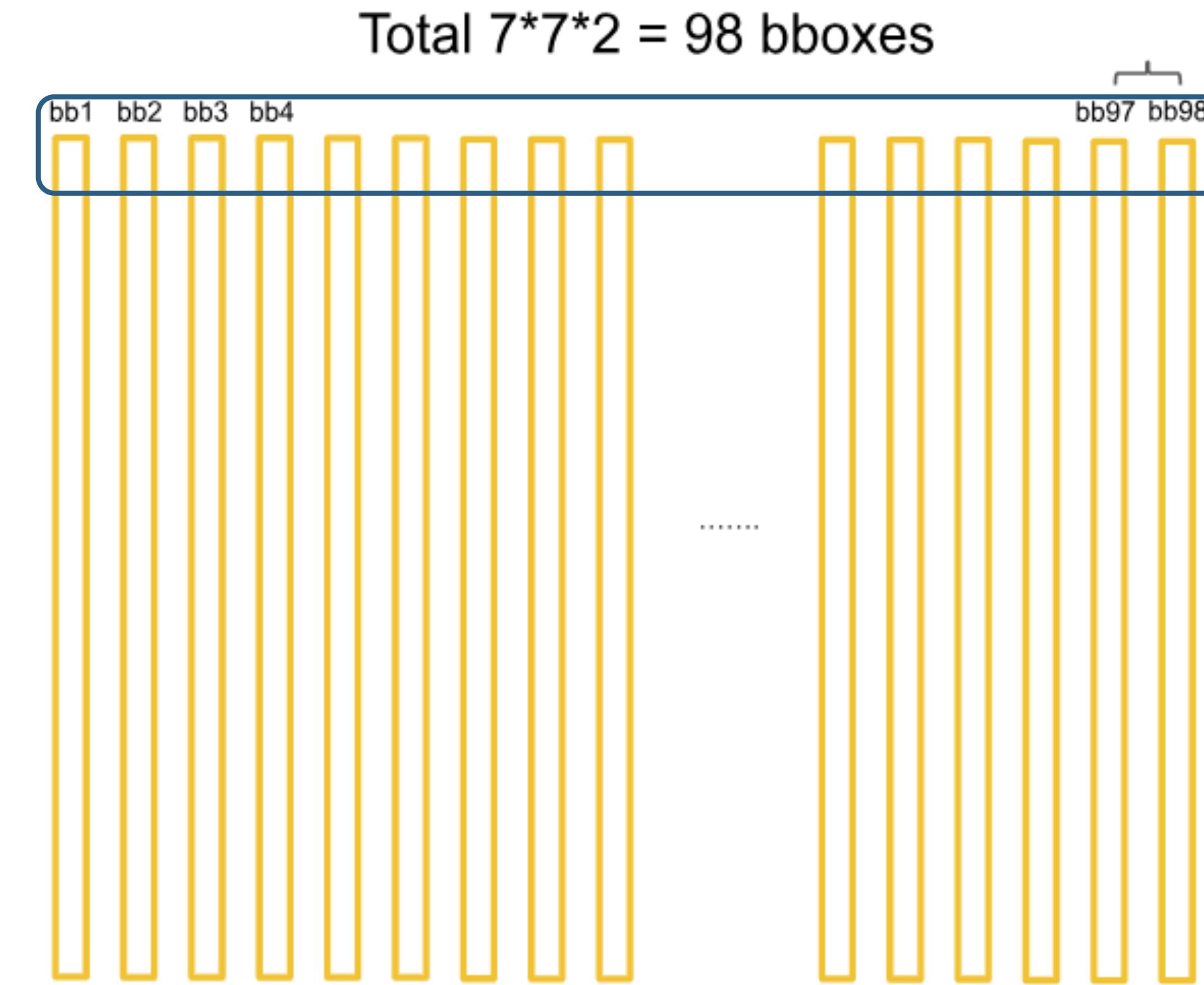
# NMS 應用於 YOLO 具體流程

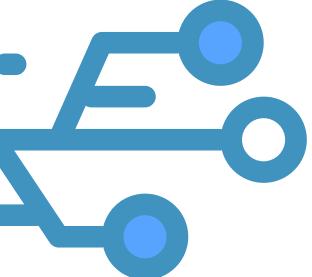


每一個類別皆執行一次NMS流程

$$p_i(\text{Class}_1) \times \text{IOU}_{truth}$$

$$p_i(\text{Class}_{20}) \times \text{IOU}_{truth}$$





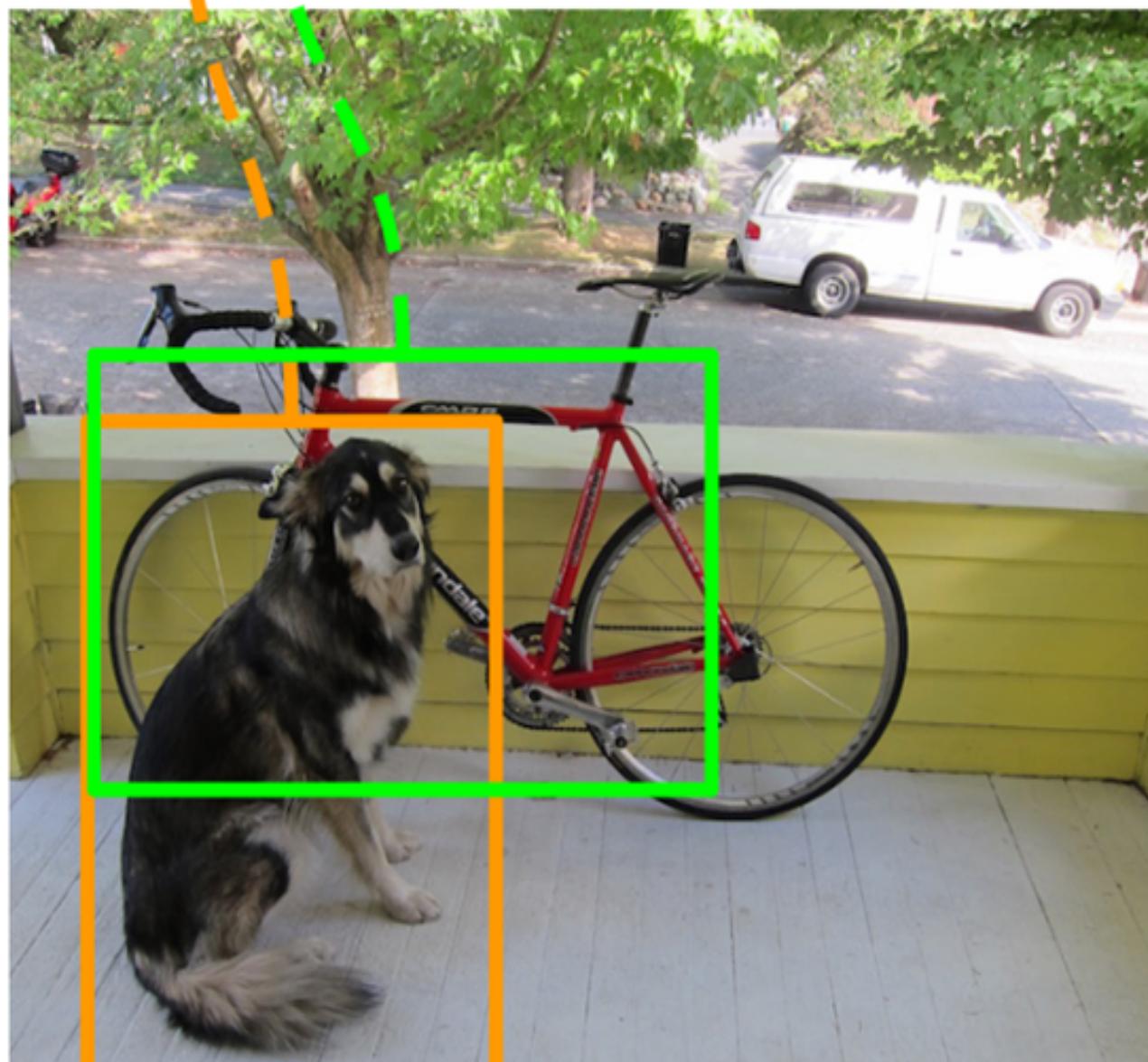
# 第一個類別（狗）執行 NMS - 第一步



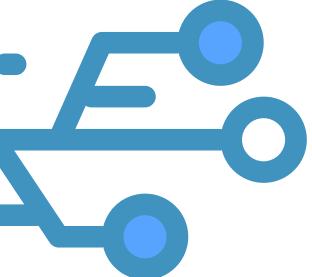
# 第一步：訓練的模型得到預測輸出

- 設定 threshold
  - 當 bbox 的信心度小於 threshold 時，代表 bbox 沒有此類別物體，因此將此 bbox 的類別物體信心度設為 0。
  - 依 bbox 的信心度排序

	bb47	bb20	bb15	bb7											bb1	bb4	bb8	bb98
class: dog	0.5	0.3	0.2	0.1											0	0	0	0



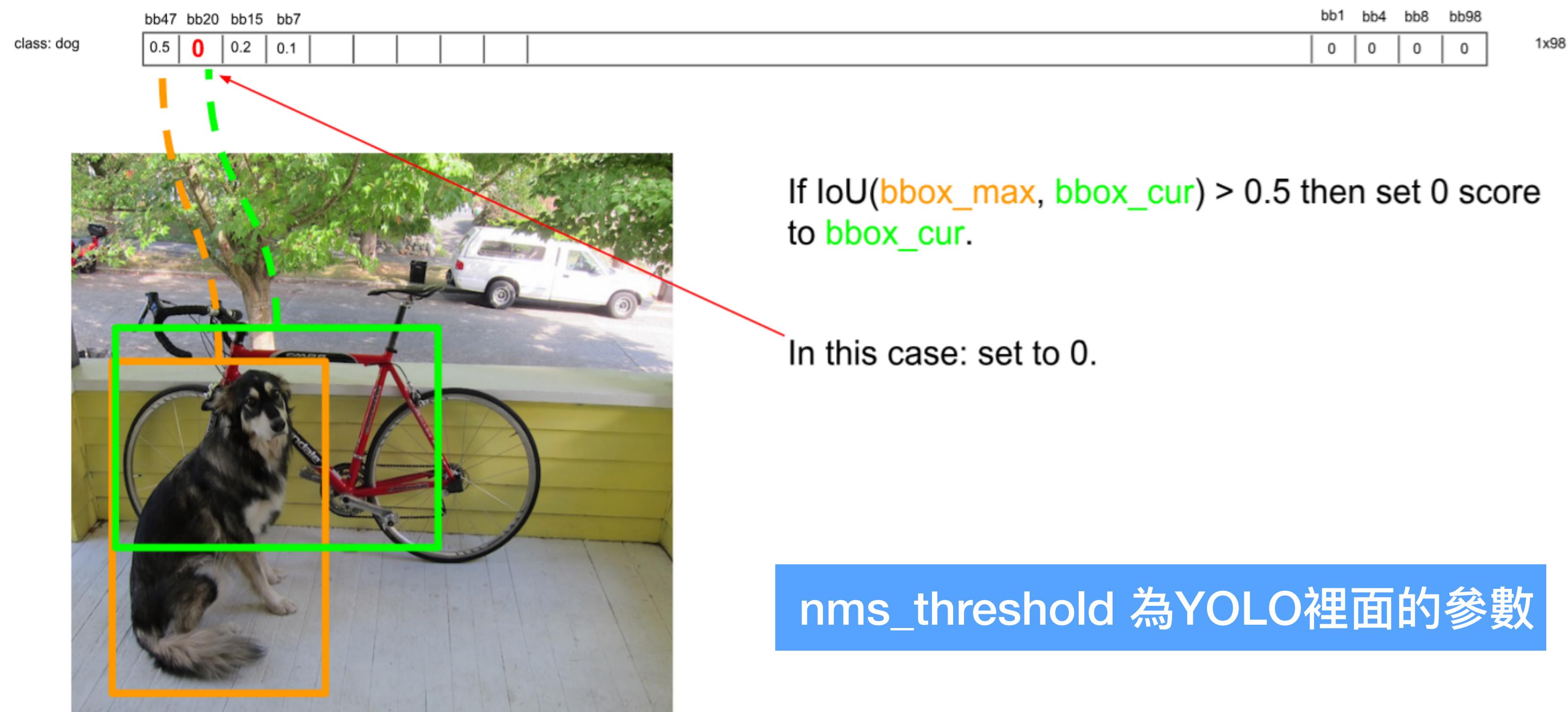
Compare “bbox\_max” with others less score (non-zero!) bboxes. Let’s denote it “bbox\_cur”

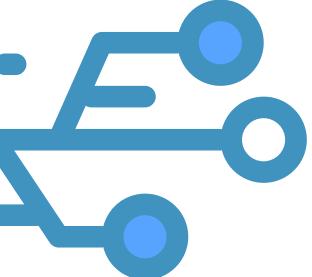


# 第一個類別（狗）執行 NMS - 第二步



第二步：信心程度最高為選定預測框，並計算其他預測框與選定預測框的 IoU，當  $\text{IoU} > \text{nms\_threshold}$ ，代表兩者框住同一個物體，信心度較小的 bbox，信心度為 0。

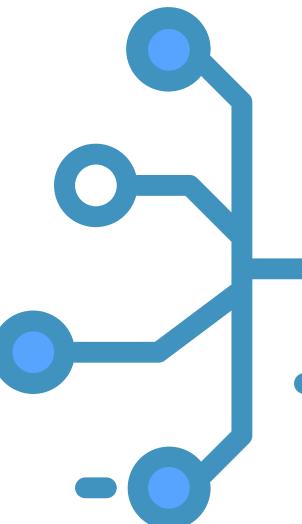
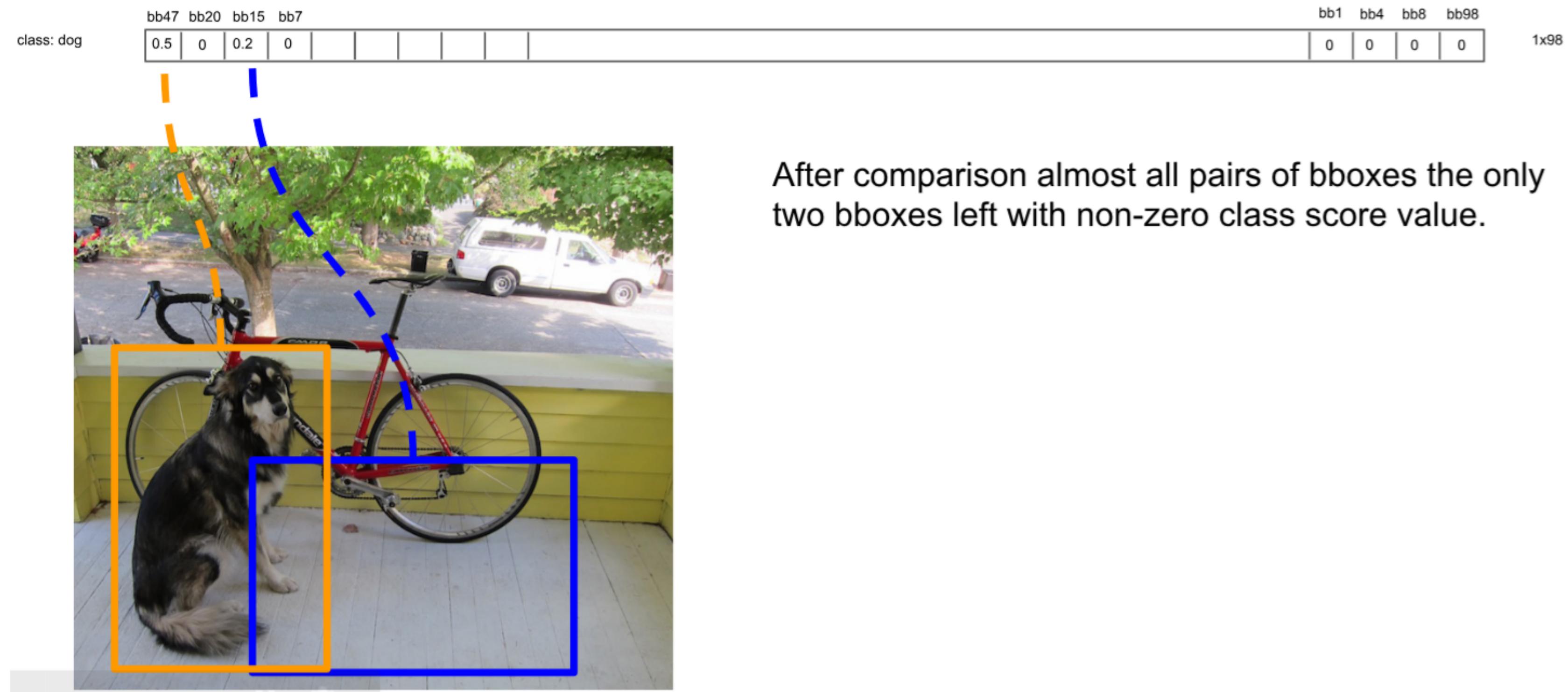


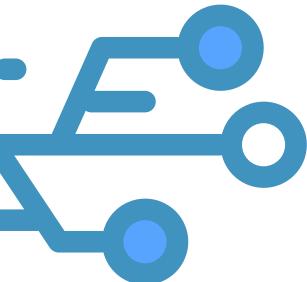


# 第一個類別 (狗) 執行 NMS - 第三步



第三步：此類別只留下信心度大於 obj\_threshold 的 bbox，然後執行下一個類別，直到全部類別都執行。

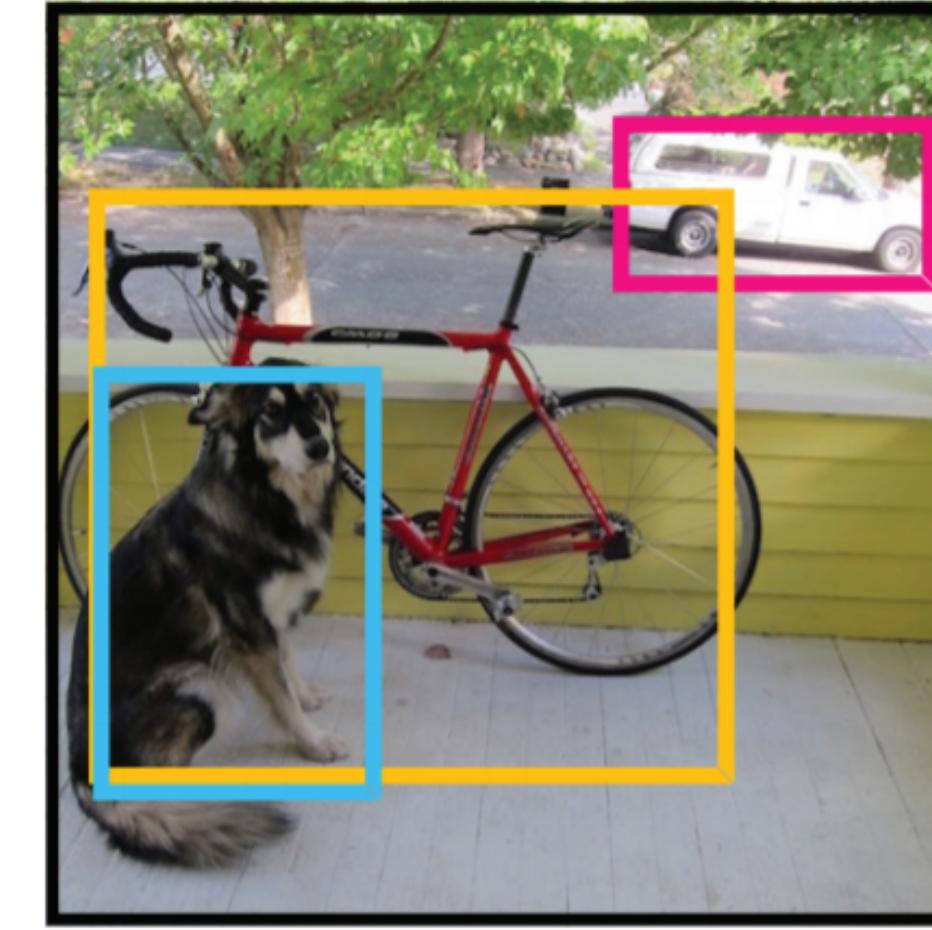
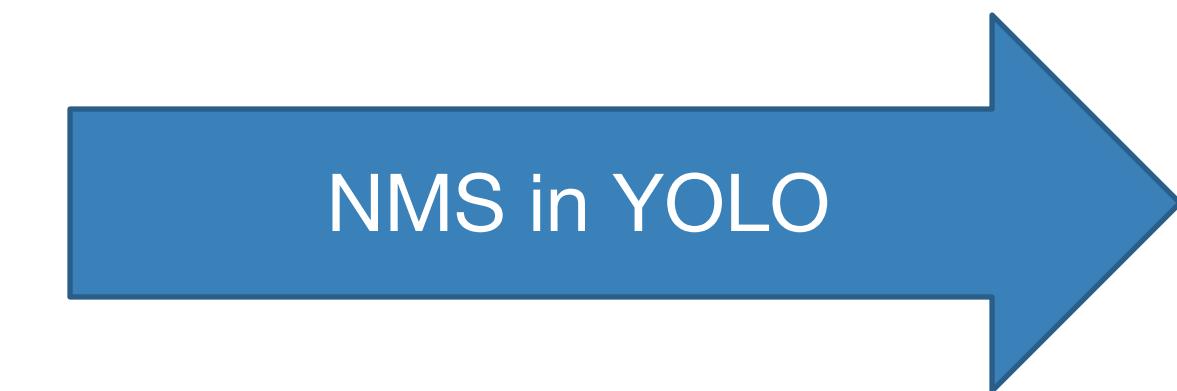




# 全部類別重複執行 NMS



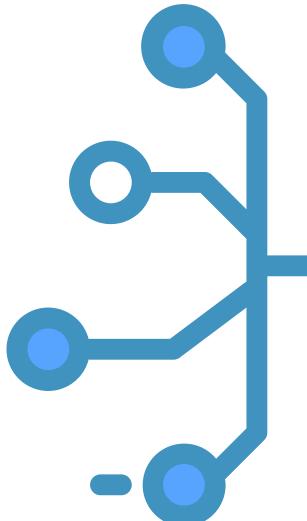
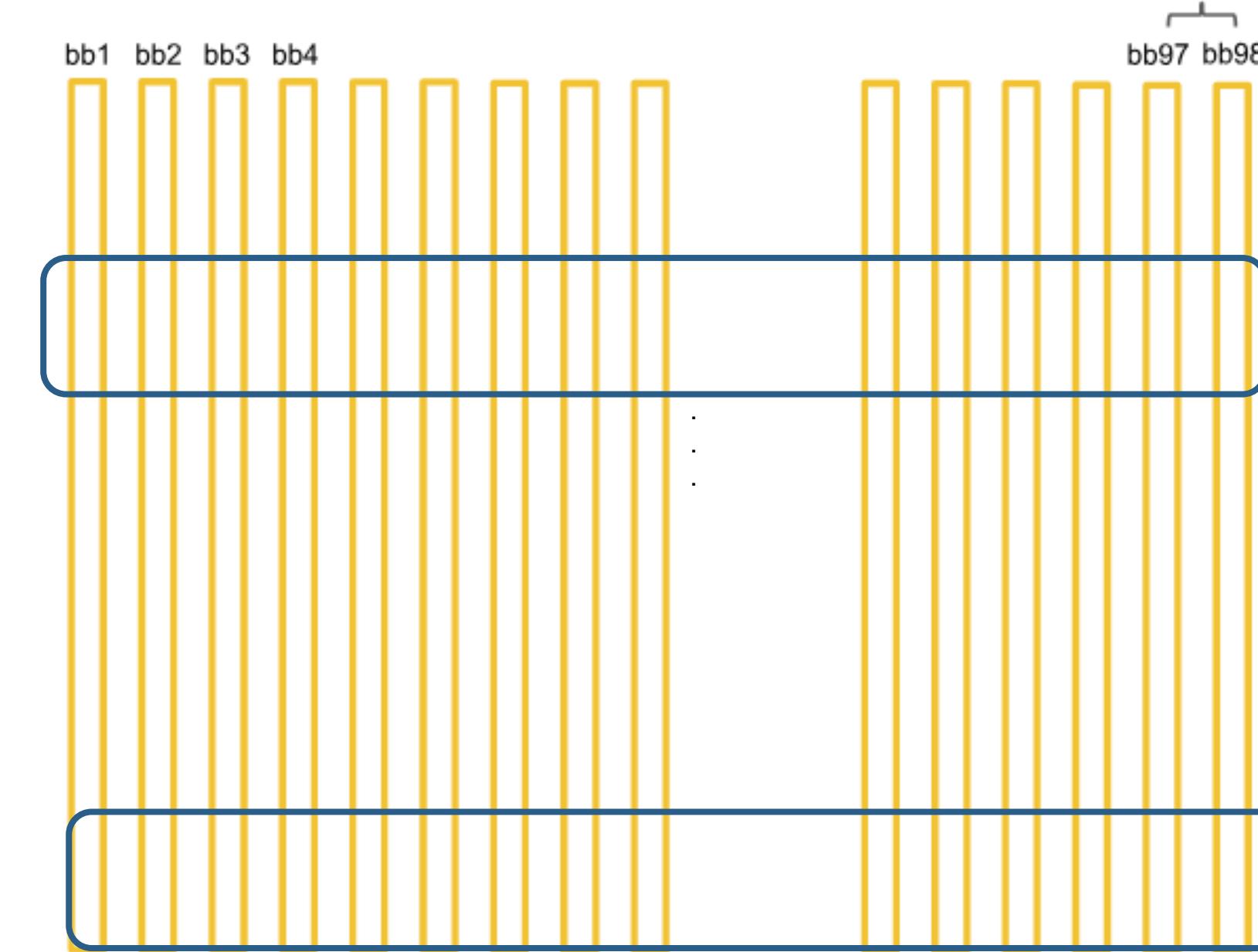
CUPOY



$$p_i(\text{Class}_1) \times C_i$$

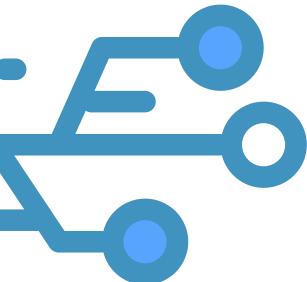
$$p_i(\text{Class}_{20}) \times C_i$$

Total  $7 \times 7 \times 2 = 98$  bboxes



# 知識點 回顧

- 理解 YOLO 輸出層轉換成 bbox 資訊的過程
- NMS 在 YOLO 的實際運作以每一個類別為主，各別執行 NMS。
- YOLO 在 NMS 中採用的信心度為「每個 bbox 包含各類別的信心度」



# 參考資料



- DeepLearning.ai- C4W1L01  
吳恩達課程  
[NMS Youtube連結](#)
- 其他NMS介紹  
[Soft NMS](#)

Non-max Suppression



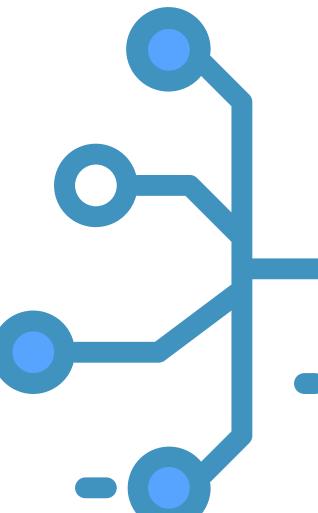
[Convolutional Neural Networks](#)

deeplearning.ai

★★★★★ 4.9 (27,370 个评分) | 200K 名学生已注册

课程 4 (共 5 门, 深度学习 专项课程)

免费注册



# 解題時間 Let's Crack It



請跳出 PDF 至官網 Sample Code & 作業開始解題