Object Detection -- YOLO

Joseph
May 24, 2019

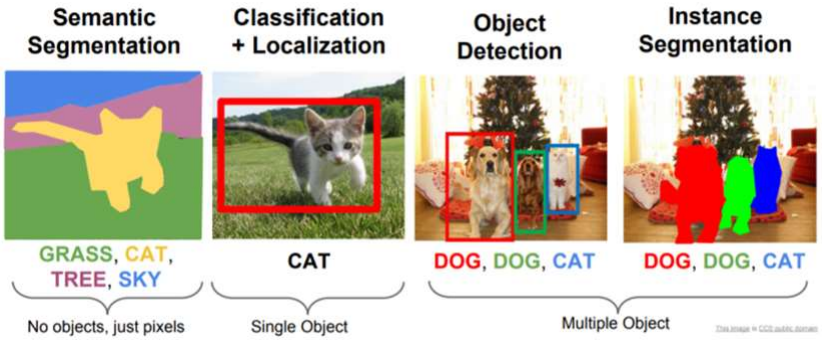
ITTRAINING 艾鍗學院 - 嵌入式軟韌體訓練中心

www.ittraining.com.tw

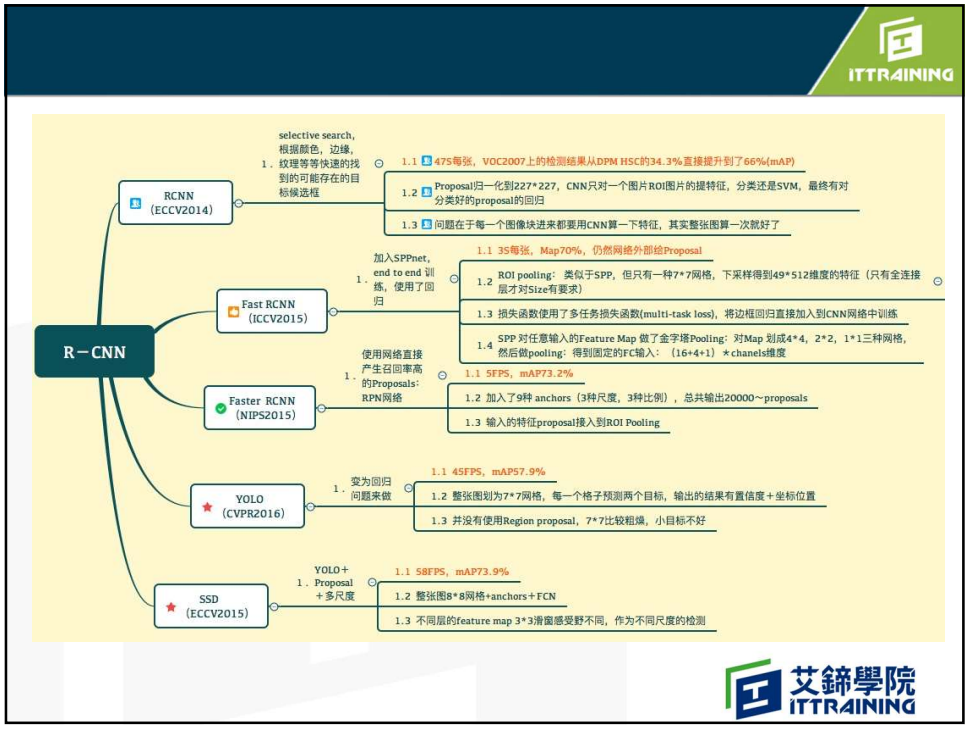
Objection Detection

ITTRAINING

- **Objection Detection: 標註多個物體(Multiple Object)所在的位置及大小**



艾鍗學院 ITTRAINING



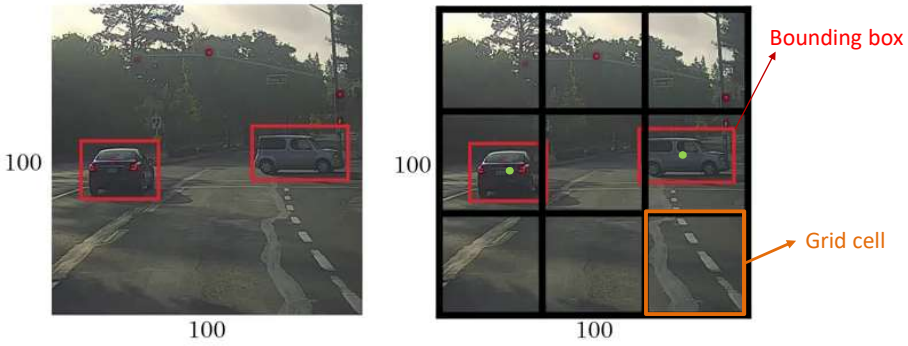
YOLO

- **Object detection** 上則為 **You only look once**，意思是說 **YOLO** 模型的特性原始圖像輸入到CNN網路中，直接輸出圖像中所有目標的位置和目標的類別，大大提升辨識速度。
- **YOLO** 的好處是單一網路設計，判斷的結果會包含 **bounding box** 位置，以及每個 **bounding box** 所屬類別及機率。整個網路設計是 **end-to-end** 的，容易訓練，而且速度快。
- **YOLOv1(TinyYOLO)**是 **416x416x3**
- **YOLOv2** 可以依所需的辨識率不同，用不同的 **input shape size** ，可以是 **608x608x3** 。

15

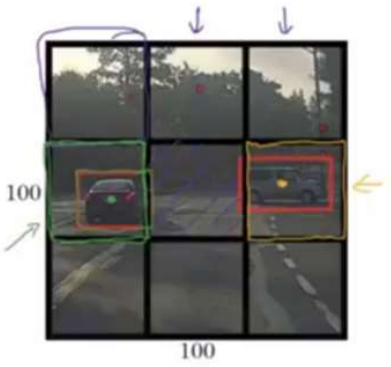
Bounding box 找出物件和分類

(100x100x3) === > (3x3x8)



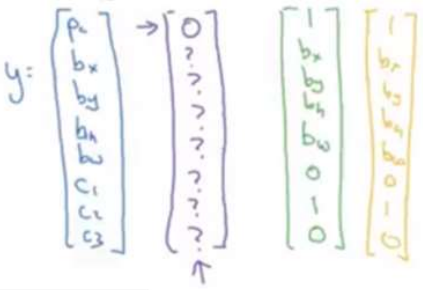
for an image of 100x100, divide into 3x3 grid, for the training data, each grid rectangle need to be labeled as "y" of 8 variables.

For now, each grid can only identify one object. Even if an object spans out to more than one grid, it will only be assigned to a single grid in which its mid-point is located.



Labels for training

For each grid cell:



How to Encode Bounding Boxes?

(0,0)

target vector.

(1,1)

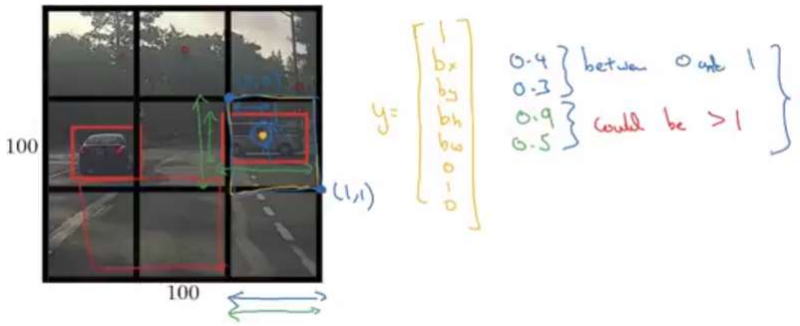
y =	
	0
	?
	?
	?
	?
	?
	?
	?

y =	
	1
	bx
	by
	bh
	bw
	0
	1
	0

y =	
	1
	0.4
	0.3
	0.9
	0.5
	0
	1
	0

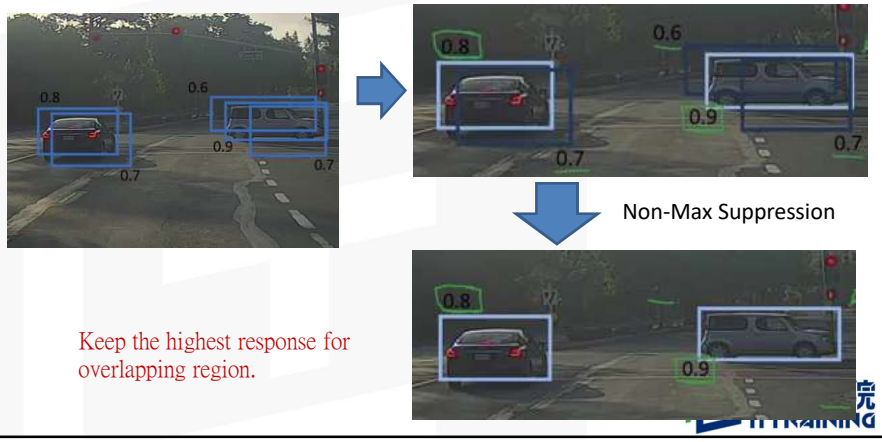
Notice here that b_x and b_y will always range between 0 and 1 as the midpoint will always lie within the grid. Whereas b_h and b_w can be more than 1 in case the dimensions of the bounding box are more than the dimension of the grid.

- Encode b_x, b_y, b_h, b_w information. By the fraction of it over the regional box.



Object might detect it multiple times!

- One object can be detected multiple times. Need to be clean up for the final result.
- Do the IoU and Non-Max Suppression to avoid selecting overlapping boxes



Keep the highest response for overlapping region.

About Non-Max Suppression

The boxes with maximum probability and suppressing the close-by boxes with non-max probabilities.

1. Discard all the boxes having probabilities less than or equal to a pre-defined threshold (say, 0.5)
2. For the remaining boxes:
   1. Pick the box with the highest probability and take that as the output prediction
   2. Discard any other box which has IoU greater than the threshold with the output box from the above step
3. Repeat step 2 until all the boxes are either taken as the output prediction or discarded

The algorithm

19× 19

Each output prediction is: p_c, b_x, b_y, b_h, b_w

Discard all boxes with $p_c \leq 0.6$

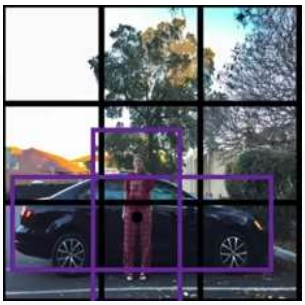
While there are any remaining boxes:

- Pick the box with the largest p_c Output that as a prediction.
- Discard any remaining box with IoU ≥ 0.5 with the box output in the previous step

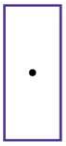
Andrew Ng

Anchor box

- What about overlapping objects ? How can recognize multiple things in the same grid box?
- Predefine anchor boxes, each box for a type of shape, replicated the same 8 variable label structure into “y” more.

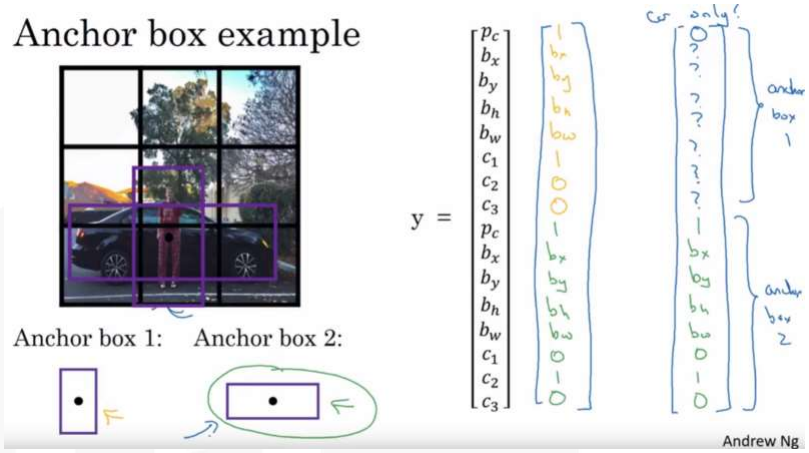


Anchor box 1:



Anchor box 2:

Add the "Anchor box" to y

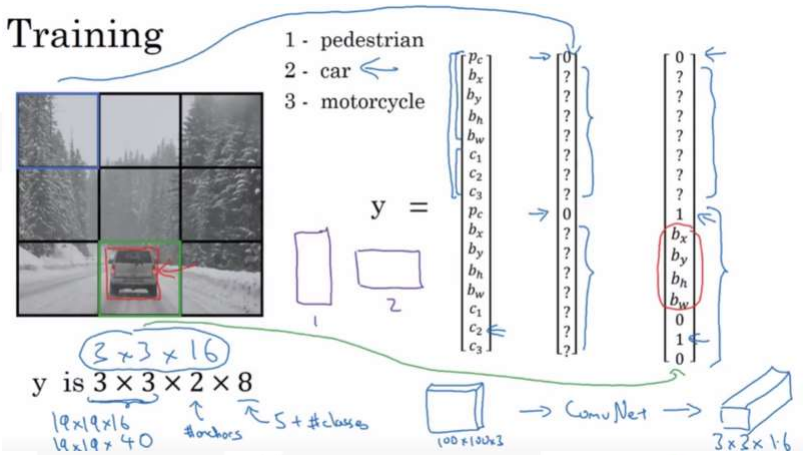


3 X 3 X 8 (using a 3 X 3 grid and 3 classes), will be 3 X 3 X 16 (since we are using 2 anchors)

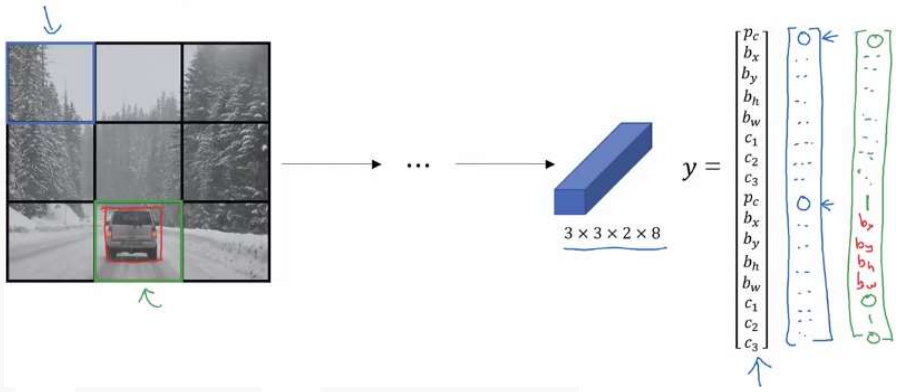
So, for each grid, we can detect two or more objects based on the number of anchors.

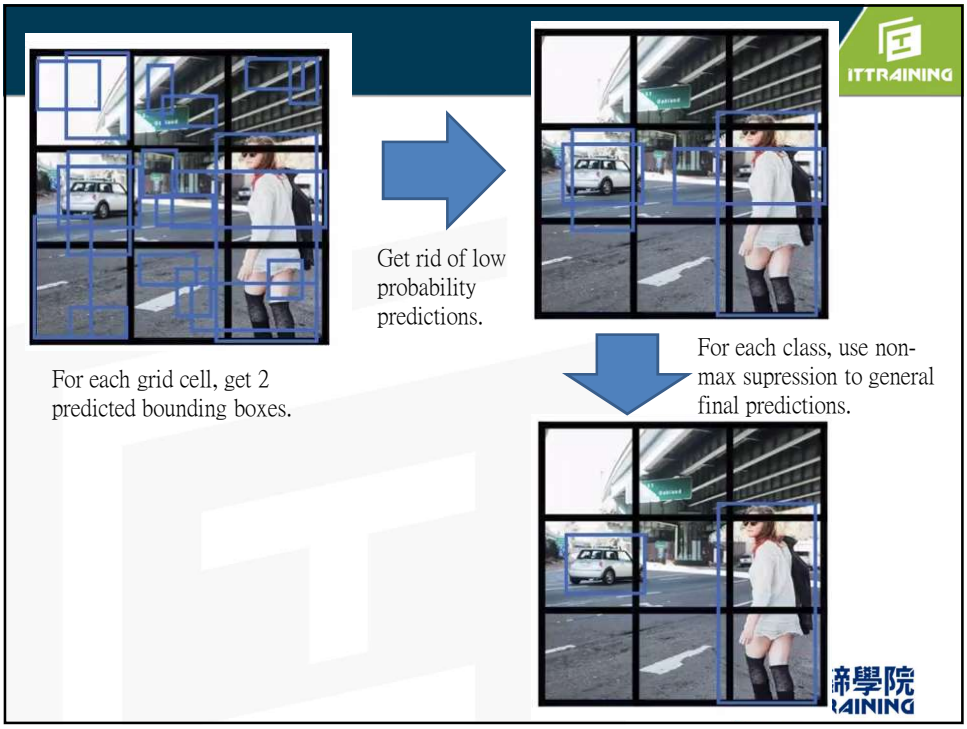
YOLO algorithm

- Training data definition for YOLO algorithm.



Making the prediction

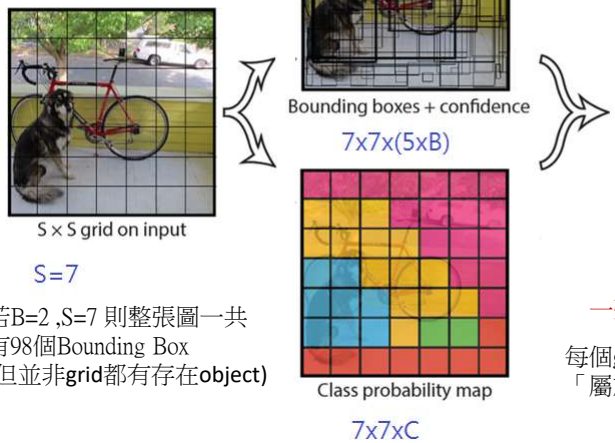




Summary

- YOLO is a state-of-the-art object detection algorithm that is incredibly fast and accurate
- We send an input image to a CNN which outputs a 19 X 19 X 5 X 85 dimension volume.
- Here, the grid size is 19 X 19 and each grid contains 5 boxes
- We filter through all the boxes using Non-Max Suppression, keep only the accurate boxes, and also eliminate overlapping boxes

Overall of YOLO Framework



用一堆框框去掃整張圖
但每一個grid只留下B個框框的結果(某些grid的框框可能只是背景而不是object, 可再用confidence加以判斷是否有object

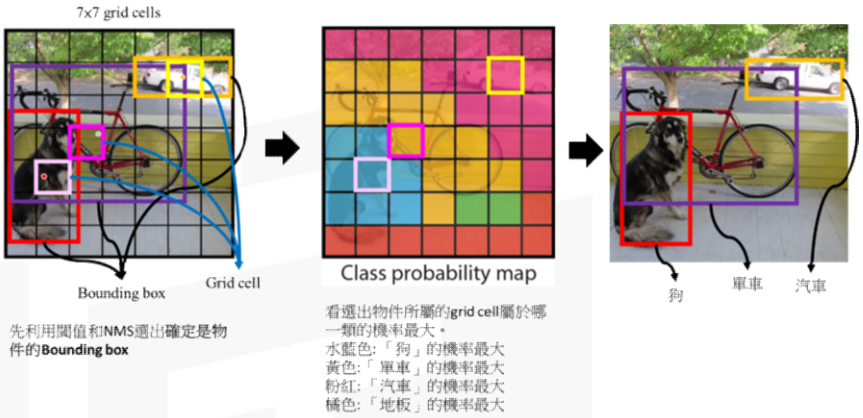
S=7

若B=2 ,S=7 則整張圖一共有98個Bounding Box
(但並非grid都有存在object)

7x7x(5xB+C)
一張圖最後輸出的format

每個grid cell還需要負責預測「屬於每個類別的機率

grid 屬於那一類



YOLO的每個grid cell最終只預測一組類別的條件機率,每個grid cell雖然有B個anchor box，但實際上只從B個bounding box中預測一個物件和這個物件屬於哪一類的機率。

30

References

- https://medium.com/datatype/deeplearning-ai-cnn-week-3-object-detection-ce86e6c648e
- https://www.analyticsvidhya.com/blog/2018/12/practical-guide-object-detection-yolo-framewor-python/

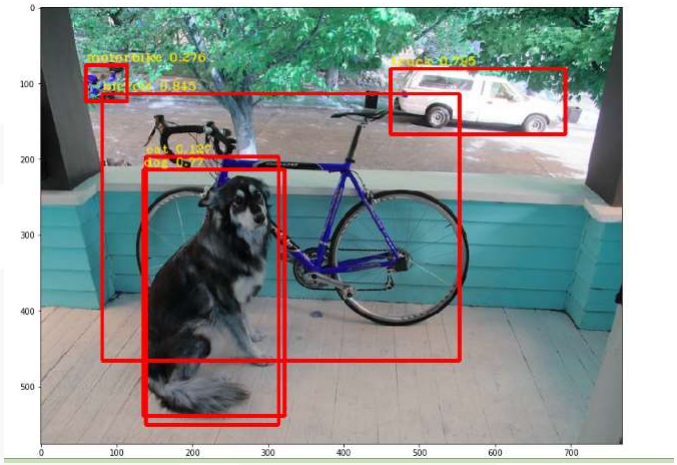
31

YOLO 實作

ITTRAINING 艾鍗學院 - 嵌入式軟韌體訓練中心

www.ittraining.com.tw

YoLo demo

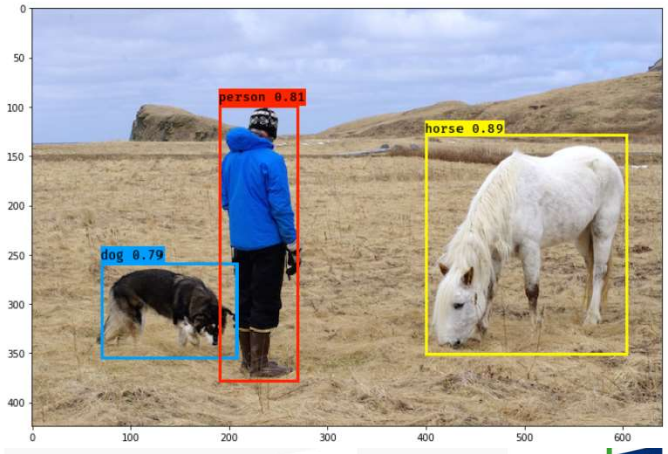


艾鍗學院 ITTRAINING

YOLOv2 Project-Object Detection

```
dog 0.79 (70, 258) (209, 356)
person 0.81 (190, 98) (271, 379)
horse 0.89 (399, 128) (605, 352)

<matplotlib.image.AxesImage at 0x14d09f28>
```



34

艾鍗學院 ITTRAINING

安裝darkflow

- https://github.com/thtrieu/darkflow
- https://github.com/pjreddie/darknet/wiki/YOLO:-Real-Time-Object-Detection

艾鍗學院 ITTRAINING