YOLO 說明

1. Metric
   1. IOU:
   2. mAP
      * [mAP@.5](mailto:mAP@.5)

IoU 取一個 threshold = 0.5 去計算 mAP。

* + - [mAP@[.5:.95](mailto:mAP@[.5:.95)]

IoU 取一個 threshold (如 .5) 可以計算 mAP，因此可以取 0.5 到 0.95 的每 0.05 一個間隔設定 threshold 都算一次 mAP 後將所有數值平均，即為 mAP@[.5:.95]。但 Yolo 的作者覺得太高的 IoU 人類也分不出來，在應用場景上可能不那麼實用。

1. YOLOv5的loss

總損失 = box loss + obj loss + cls loss

* 1. 定位損失box\_loss(box loss)：預測框與標定框之間的誤差。
  2. 置信度損失obj\_loss(object loss)：計算網路的置信度。
  3. 分類損失cls\_loss(class loss)：計算錨框與對應的標定分類是否正確。

1. continue

**yolov7-main folder**

**|-- cfg**

**| |-- baseline**

**| |-- deploy**

**| |-- training**

**| | |-- fruits.yaml**

**|-- data**

**| |-- fruits.yaml**

**| |-- hyp.scratch.p5.paml**

**| |-- hyp.scratch.p6.paml**

**|-- datasets(EX: fruits)**

**| |-- images**

**| | |-- train**

**| | |-- \*.jpg**

**| | |-- \*.jpg**

**| | | … …**

**| | | -- val**

**| | |-- \*.jpg**

**| | |-- \*.jpg**

**| | | … …**

**| |-- labels**

**| | |-- train**

**| | |-- \*.txt**

**| | |-- \*.txt**

**| | | … …**

**| | |-- val**

**| | |-- \*.txt**

**| | |-- \*.txt**

**| | | … …**

**| |-- train.txt**

**| |-- val.txt**

**|-- runs**

**| |-- detect**

**| |-- train**

**|-- detect.py**

**|-- train.py**

**|-- test.py**

1. Command for detect