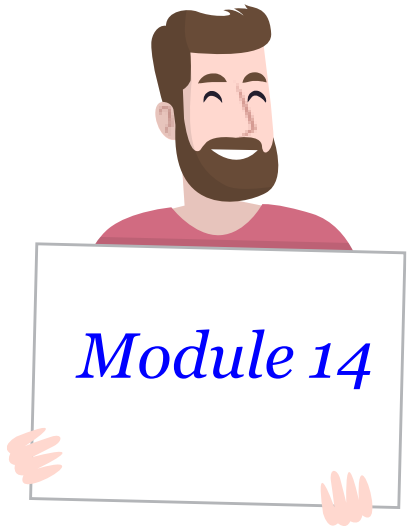
CNN物件偵測原理



Estimated time:
45 min.

學習目標

- **14-1: 物件偵測原理**
- **14-2: Yolo介紹**
- **14-3: Labeling資料**

14-1: 物件偵測原理

- 物件偵測原理
- 傳統物件偵測作法
- 深度學習作法

物件偵測原理

- 物件偵測原理為給予一張照片，電腦要能將所有認識的物件bounding box框出來，並辨識該bounding box的類別
  - 每個bounding box均有自己的(x, y, w, h)以及對應的類別

Cat: (x, y, w, h)

Cat: (x, y, w, h)
Dog: (x, y, w, h)

傳統物件偵測作法

- 傳統物件偵測作法
  - 使用一個滑動窗口去掃描整張圖
  - 針對每個當下掃描到的圖去偵測是否有認識的物件

背景：YES
狗： NO
貓：NO

背景： NO
狗： NO
貓： YES

傳統物件偵測作法

- 傳統物件偵測作法缺點
  - 由於需要不停地去掃描整張圖片，所以非常花時間
  - 此外，傳統做法每掃到圖片的一的地方，都需要做一次卷積，運算資源也花費非常大

背景：YES
狗： NO
貓：NO

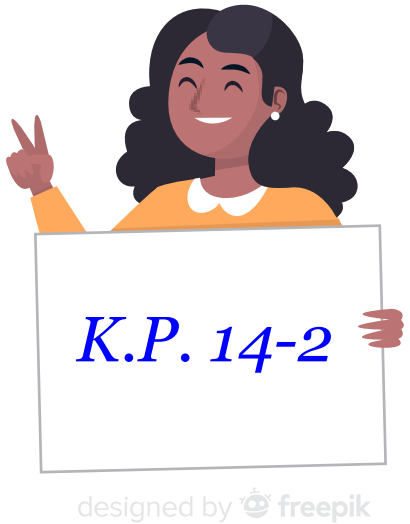
背景： NO
狗： NO
貓： YES

深度學習作法

- 近期大家比較偏好以深度學習來實作物件偵測
- 常見深度學習物件偵測方法有兩種
  - two stage的方法
  - one stage的方法

14-2:Yolo介紹

- One stage及Two stage物件偵測介紹
- Yolo原理
- 衡量物件偵測模型

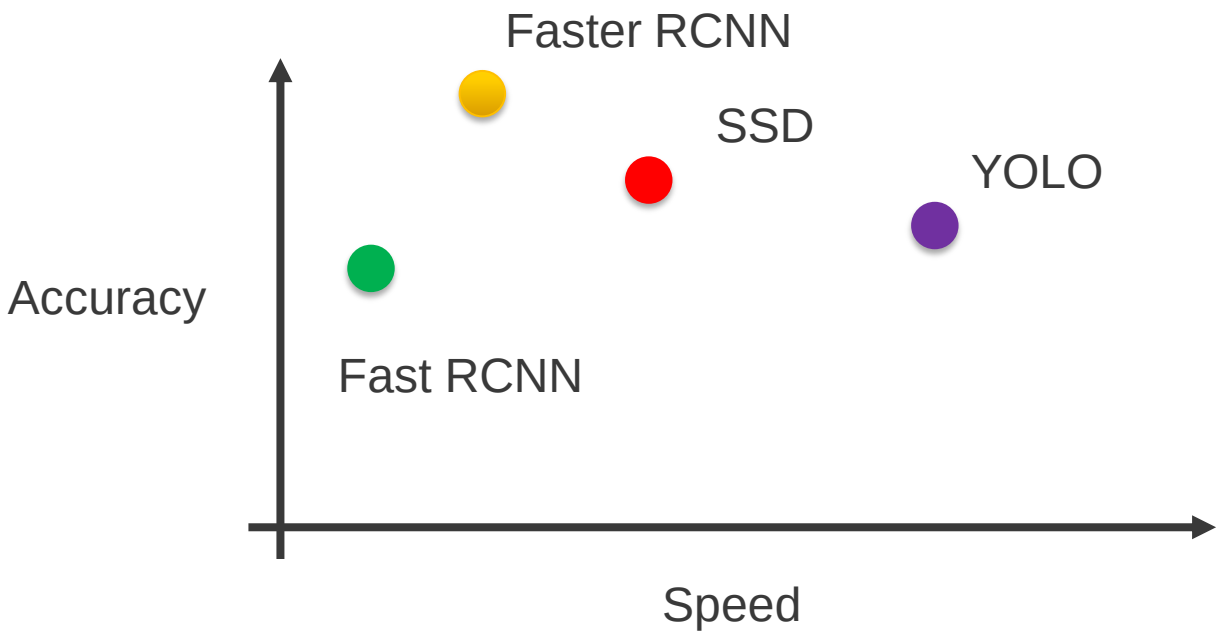


One stage及Two stage物件偵測介紹

- 所謂的one stage以及two stage的差異在於網路是否分成兩部分
  - 一部分物件選取，另一部分是物件辨識
- Two stage物件偵測的方法
  - 準確度比較高、但辨識速度比較慢
- One stage
  - 準確度比較低、但辨識速度比較快

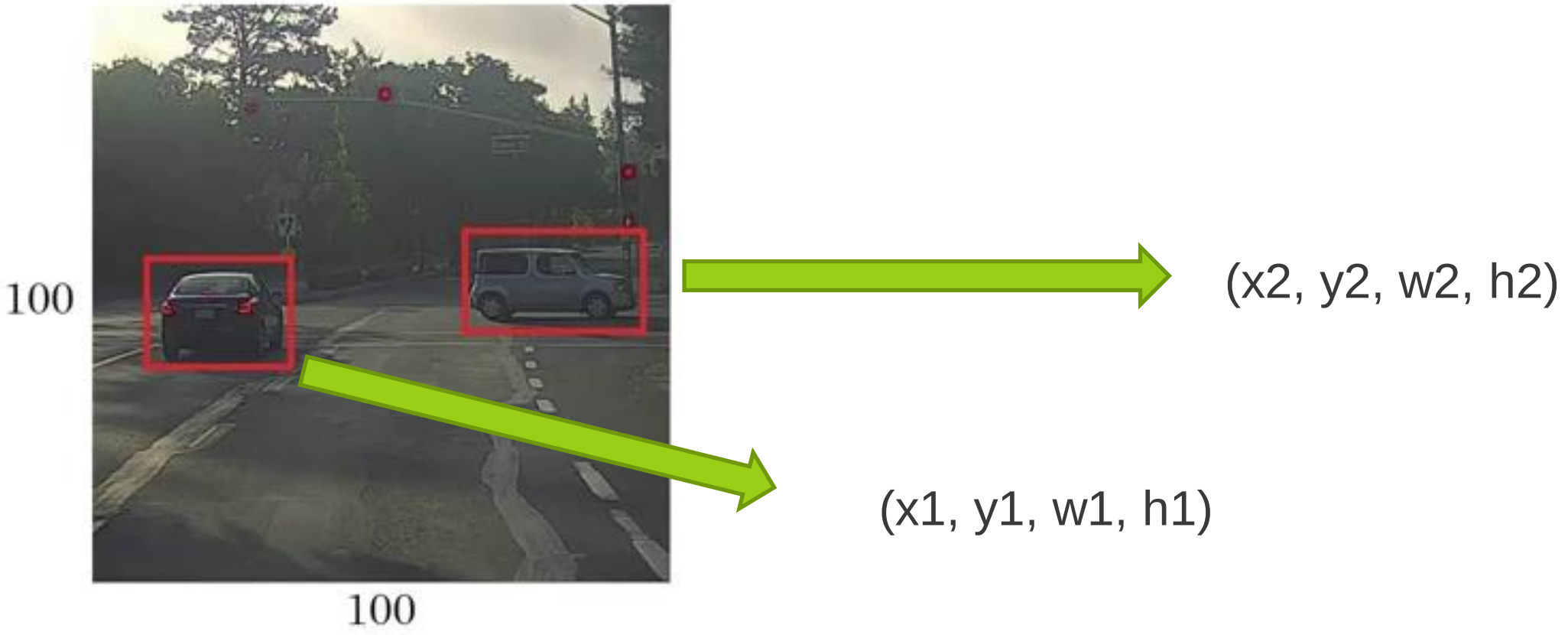
One stage及Two stage物件偵測介紹

- one stage以及two stage速度及準度效能上的差異圖如下
  - 在某些應用需要real-time情況下，YOLO會比較受歡迎
- 常見two sage演算法有RCNN、Fast-RCNN、 Faster-RCNN
- 常見one sage演算法有Yolo



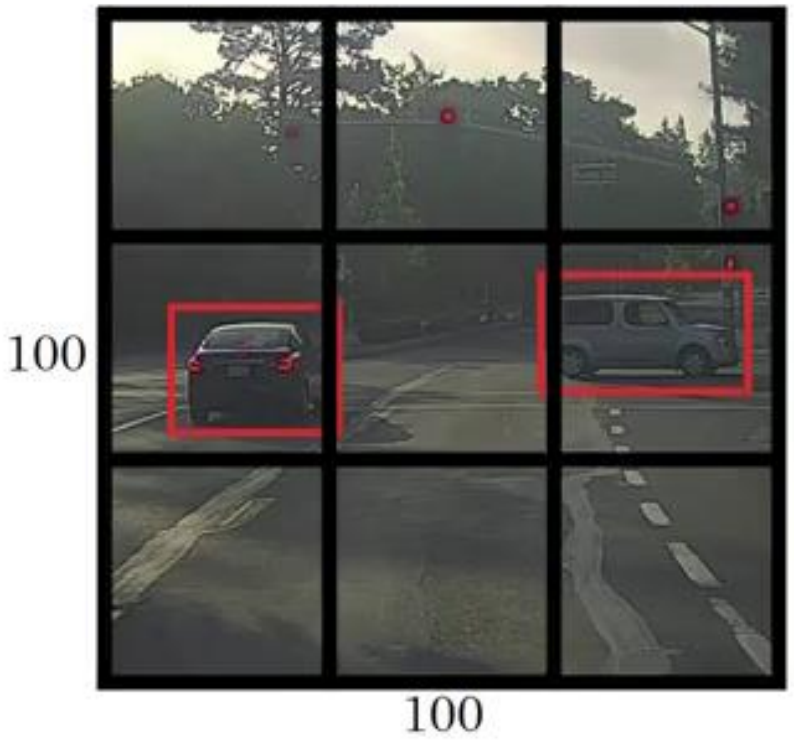
Yolo原理

- **Yolo是一個one stage物件偵測演算法**
  - **其用一個神經網路來預測照片內每個物件在哪以及對應的類別**



Yolo原理

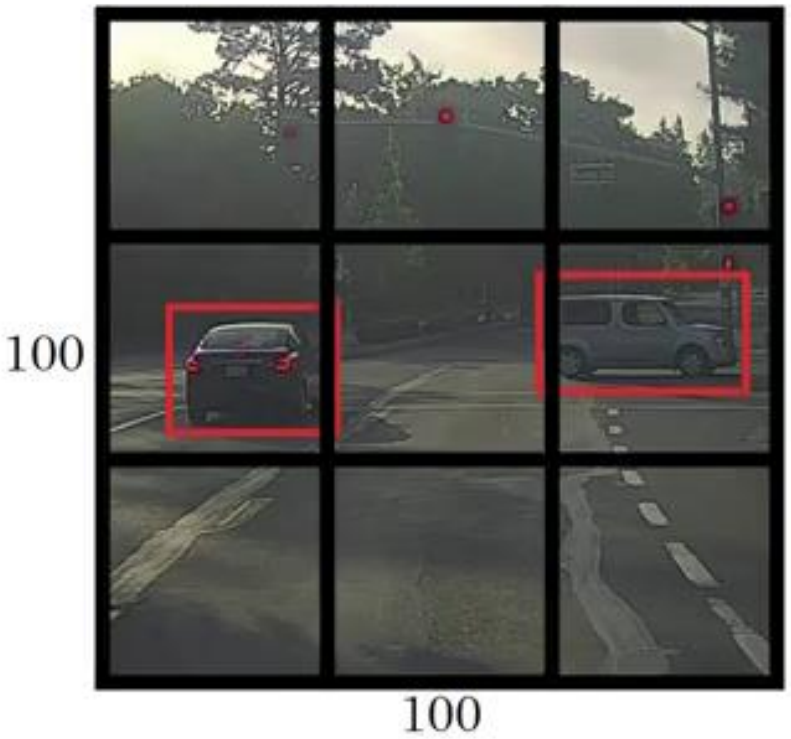
- 首先Yolo會將照片都分成一個一個的格子
  - 格子要切多細可以自己設定



將照片分成3*3的格子grid

Yolo原理

- **針對每個格子，需要給他一組標籤**
  - **這裡的標籤包含是否有物件、物件的bounding box、該物件屬於哪個類別(如下右圖)**



針對每個格子我們給他一組

$$y = \begin{bmatrix} pc \\ bx \\ by \\ bh \\ bw \\ c1 \\ c2 \\ c3 \end{bmatrix}$$

是否有物體(1=是 0=否)

是否是類別1(1=是 0=否)

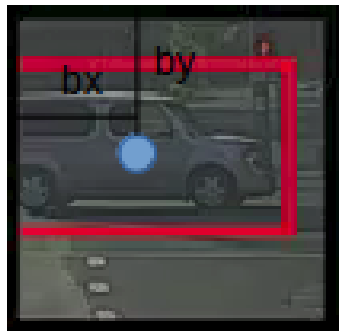
是否是類別2(1=是 0=否)

是否是類別3(1=是 0=否)

Yolo原理

- 以下為部分格子所對應標籤之範例
  - 如果已經確定該格沒有物件，除了第一個欄位為0外，其他欄位數值不重要

(0,0)



(1,1)

y =

0
?
?
?
?
?
?
?
?

y =

1
0.4
0.3
0.9
0.5
0
1
0

Yolo原理

- 根據剛剛所有格子的標籤，可以建構一個神經網路，直接將整張照片輸入，其輸出為每個格子的標籤

100*100*3

3*3*8

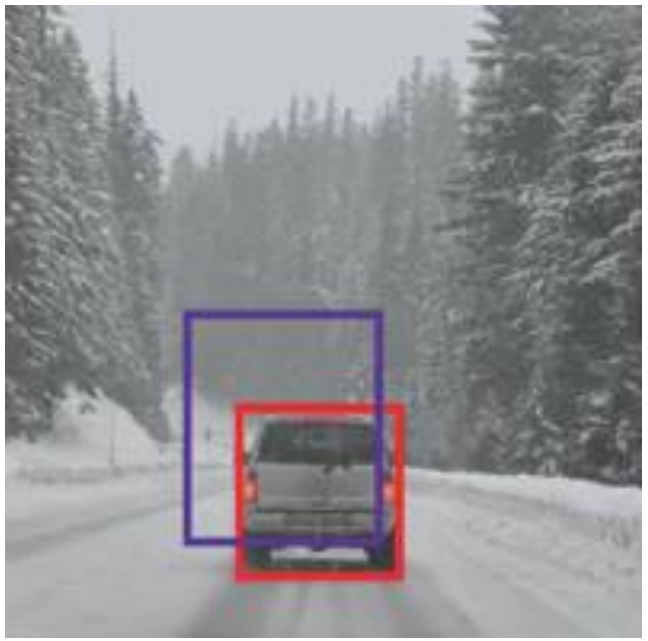
pc
bx
by
bh
bw
c1
c2
c3

Yolo原理

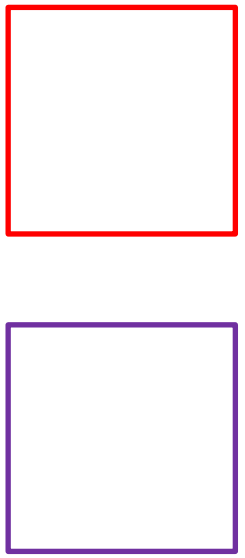
- 常見Yolo開源程式碼
  - https://github.com/thtrieu/darkflow
  - 通常都會有附預訓練好的模型，或是使用者可以增加自己的類別再重新訓練
  - 詳細內容可以參考open source說明

衡量物件偵測模型

- 如何衡量物件偵測的bounding box是否框的正確
  - 常見的衡量方法為使用IOU這個指標

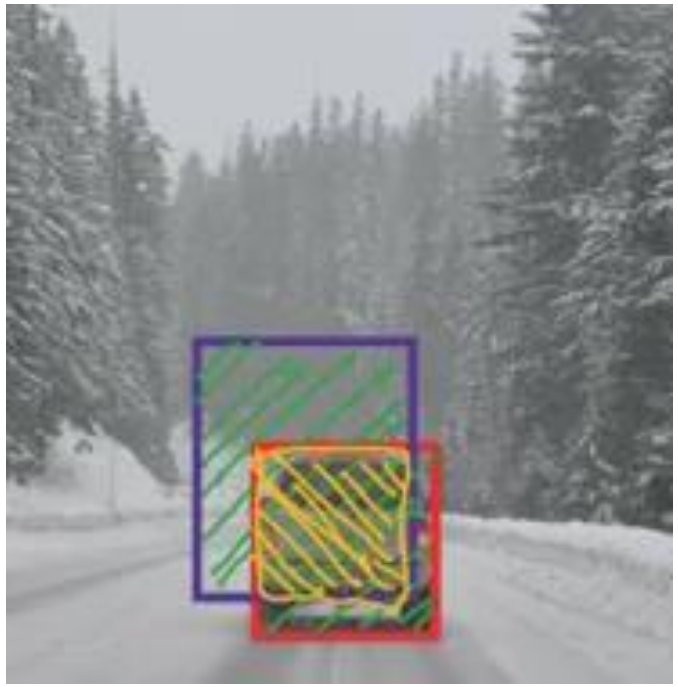


如何衡量模型好壞



衡量物件偵測模型

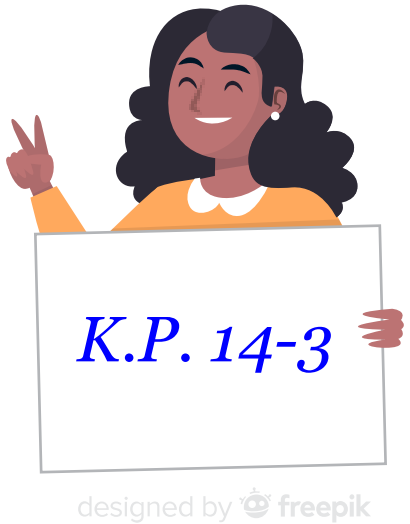
- **IOU是用來衡量bounding box是否框的好的指標**
  - **一般來說，IOU大於0.5表示預測的bounding box夠好**
  - **其公式如右下**



$$IoU = \frac{\text{兩個框框的交集面積}}{\text{兩個框框聯集面積}} = \frac{\text{黃色框框面積}}{\text{綠色框框面積}}$$

14-3:Labeling資料

- Labeling tool介紹
- 群眾外包

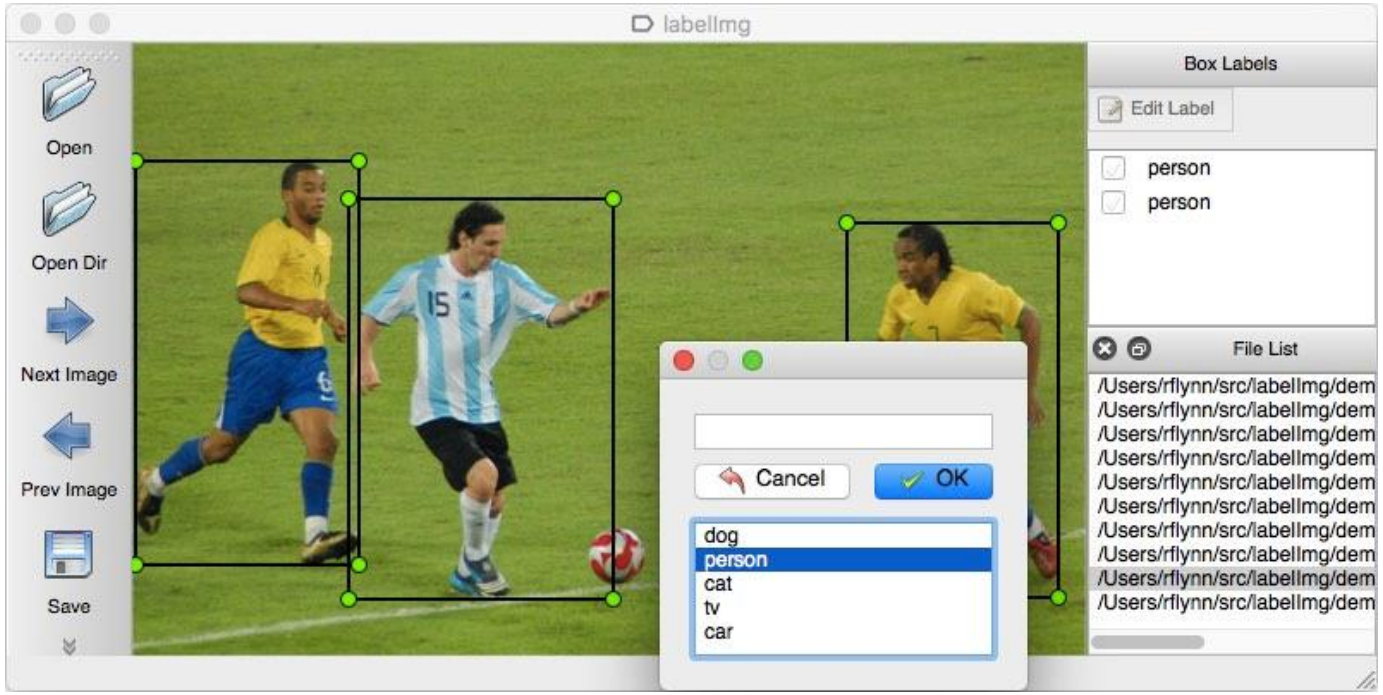


Labeling tool介紹

- 在增加辨識類別並重新訓練Yolo的時候，常常需要自己標註蒐集好的資料
- 可以使用常見的Labeling tool自己標註
  - 現有不少labeling tool可以使用

Labeling tool介紹

- **LabelImg是免費開源的Labeling tool**
  - 內建許多快捷鍵以及協助使用者標註資料
  - https://github.com/tzutalin/labelImg



群眾外包

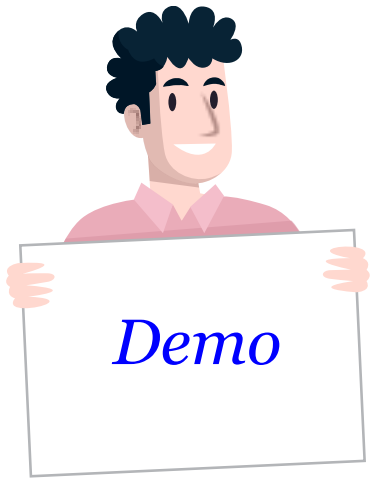
- 群眾外包是另一種labeling資料的方法
  - 可以使用如Amazon Mechanical Turk (MTurk)這類的服務，會有人協助標記資料
  - https://www.mturk.com/
  - 另外還有一些專業的labeling公司可以協助

群眾外包

- 雖然可以藉由群眾外包平台協助標註資料，但許多專業領域資料標註還是仰賴專業人員
  - 例如醫療影像、工廠機台參數設定等.....

Demo 14-3

- Yolo相關套件安裝
- 準備資料
- 執行Yolo程式



線上Corelab

- 題目1： 使用yolo偵測指定照片
  - 將dog.jpg照片輸入讓yolo偵測
- 題目2：使用yolo將照片讀入並標準化
- 題目3：使用yolo模型辨識指定照片

本章重點精華回顧

- 物件偵測的原理
- YOLO
- 衡量物件偵測模型
- 資料標註

Lab:Yolo模型使用

- **Lab01:Yolo相關套件安裝**
- **Lab02:準備資料**
- **Lab03:執行Yolo程式**

Estimated time:
20 minutes


THANK YOU