

**110 年度第二學期 智慧自動化工程科**

**機器視覺與實習 期末報告**

**班級：智動四**

**姓名：鄭秉豐 李家碩**

**座號：1072b0010 1072B0015**

**中華民國111年6月26日星期天**

**目次**

**一、 摘要** ………………………………………………………………2

* 1. **SSD(物件偵測類神經網路演算法)**
  2. **PyQt**

**二、 方法與流程圖** ……………………………………………………3

**三、 實驗結果** …………………………………………………………4

**3.1 介面概覽**

**3.2 自動標籤流程**

**3.3 開始自動標籤與中途暫停**

**3.4 存檔與更改**

**四、 貢獻說明** …………………………………………………………5

**五、 參考文獻** …………………………………………………………5

1. **摘要**

在機器視覺之深度學習時，我們需要準備好固定格式與標註好的資料，提供給訓練模型學習。所需之資料集現有之影像標註工具層出不窮，然而在使用這些工具時，往往需要消耗大量時間與精力在框選物件及填寫資料上面。因此，此專案將Yolo (物件偵測Object Detection類神經網路演算法)與Pyqt5 (GUI編程)整合，並建立一個介面，給予需要建立訓練資料集的使用者更方便的體驗，減少在標籤物件時所需要消耗時間。

* 1. **SSD(物件偵測類神經網路演算法)**

本專題所選用的SSD演算法是基於YOLO v3所編寫的。近幾年十分熱門的YOLO演算法是個準確且快速的物件辨識模型，YOLO是「You Only Look Once」的簡寫。過往的物件辨識採用兩階段式的分工，也就是使用兩個神經網路，第一個神經網路負責找出圖片中的關鍵物件，利用候選框標示後再給下一個模型進行分類。但是，這種作法在速度上無法滿足即時性的要求，因此，YOLO提出只用一個網路同時做物件辨識及分類判斷。

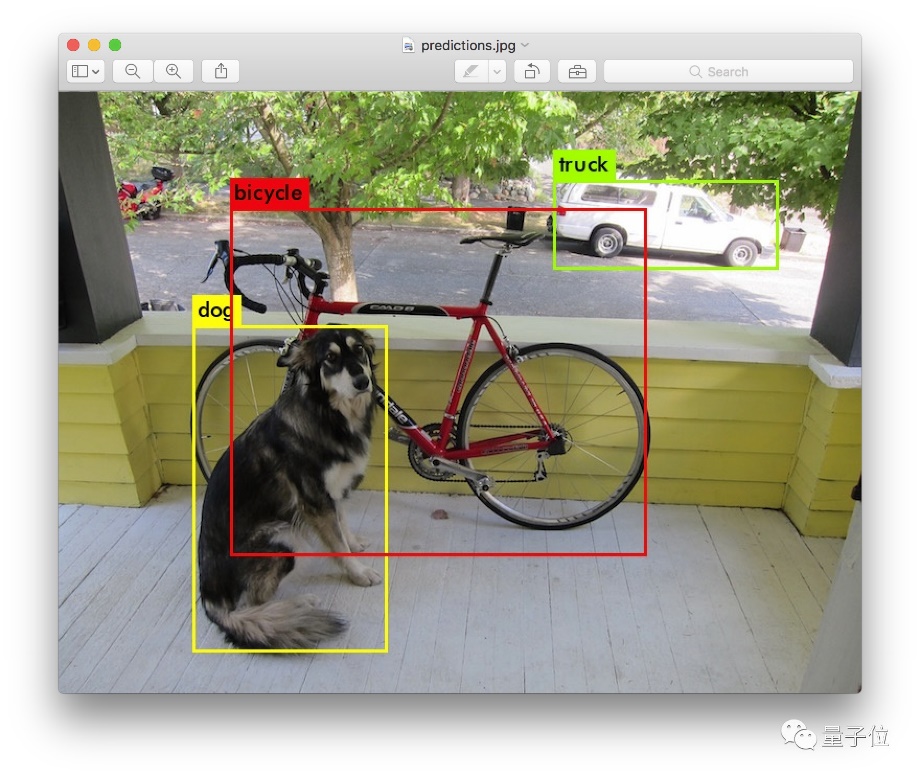


圖 1 YOLO演算法之輸出視窗

將輸入的圖片切成多個網格（grid），每個網格會產生類別機率、候選框資訊及信心度結合NMS產生結果。(圖1)

**候選框資訊**:當電腦尋找圖片哪些地方可能包含所需物件時，候選框的格子中所產出的訊息。

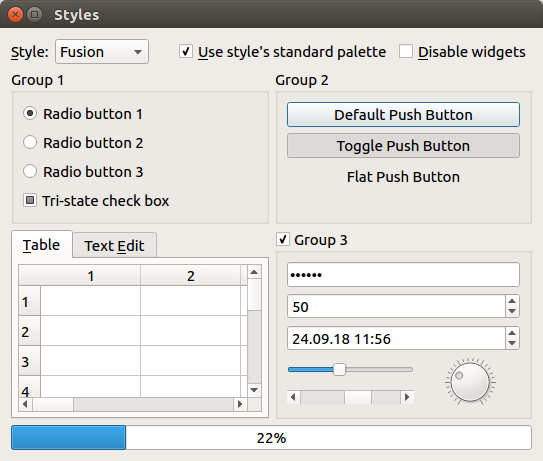
**信心度**:當電腦標示出有可能符合物件的候選框時，必須要篩選出真正符合物件的候選框並保留，每個候選框內包含物件的信心程度多寡就是信心度，若信心度較低的候選框就會被排除掉。

**NMS**:為了消除多餘的候選框，NMS考慮候選框重疊程度，保留信心度最高的候選框，再進行下一步將框分類。

由於電腦在更新參數時，會考慮到候選框損失函數、格子內是否有物件的信心值，以及每個格子類別的機率等因素，並與實際值對照並進行參數更新，以幫助模型學習。YOLO將物件辨識分類跟框預測問題轉成迴歸預測效果，也讓物件辨識的效率提升。且在不同尺寸下，每一個格子（gird）都會預測三個候選框，一張圖片在電腦中會被分成三種不同尺寸的格子，每個格子都預測三個候選框資訊，一張照片可能會產生一萬多個候選框。接著，電腦會計算並挑選出那些框是有物件且保留。YOLO的計算量很大，但也因為這樣才能更精準地找出物件。

* 1. **PyQt**

PyQt是一個建立Python GUI應用程式的工具包，將Qt的功能用於Python開發的一個Qt的Python包裝器。



PyQt的整個程式開發框架，主要包括如下部分(圖2)：

**圖形介面編輯的工具**：Qt Designer

**不同部分資訊交換機制**：訊號和槽

**介面操作的事件及捕獲機制**

**控制介面顯示和資料儲存分離以及對映的機制**：Model/View架構

圖 2 Pyqt之介面

透過這些工具和框架機制，開發人員可以設計對應的GUI圖形化介面、定義不同部件的操作及響應、捕獲部件或應用的訊息以及實現介面顯示元件和資料儲存元件的聯動，從而構造完整的應用程式框架。本專案將二者整合，提供給使用者更方便的影像標註工具。

1. **方法與流程圖**

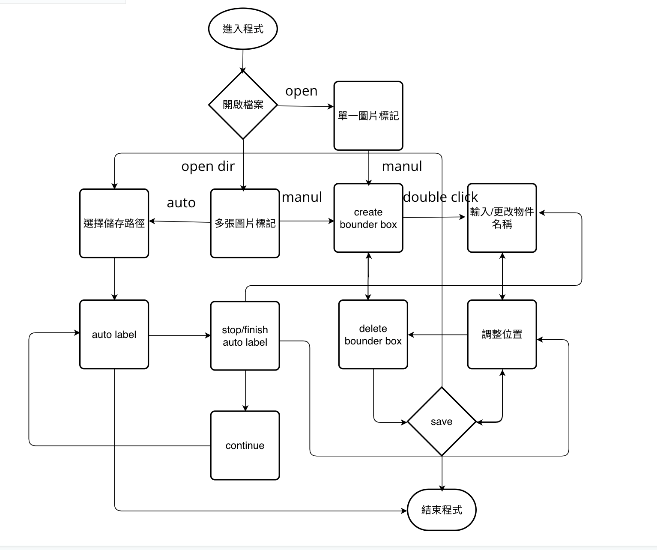


圖 4 程式編寫流程圖

圖 3 使用者介面流程圖

首先是構思整個使用過程所需要的結果(圖3)，包括需要出現的選項，點擊選項之後會進到什麼介面等。在構思好使用者整個介面與使用流程之後。我們開始著手規劃編寫程式的流程，包括收集資料、介面設計、結合SSD演算法去實現自動化標註的功能。(圖4)

1. **實驗結果**

**3.1 介面概覽**



圖 5 介面概覽圖

介面中由上至下分別列出「打開檔案」、「儲存」、「建立方框」等按鍵，仿照現有之物件標籤工具介面，讓使用者可以無縫接軌地使用此專案開發之工具。

**3.2 自動標籤流程**



圖 6 選取原圖資料夾

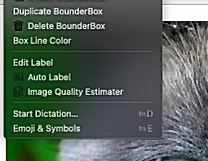
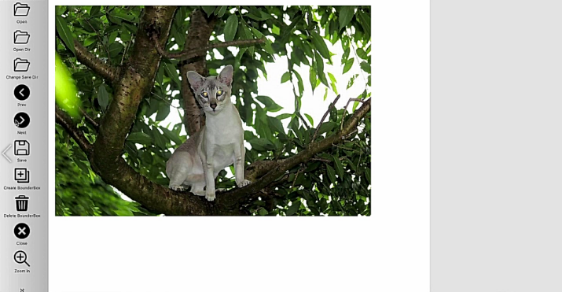
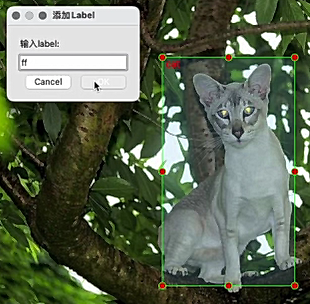
選取打開資料夾之按鍵，可以選取存有需要進行標籤之原始圖檔之資料夾，選取完成之後會在介面出現先前選取之圖檔。此時圖檔還未一一進行標籤和轉檔成資料集之格式 (XML檔)。

圖 9 自動標籤

圖 7 圖檔顯示於介面

圖 8 點選自動標籤

圖 10 更改標籤名稱

點選Auto Label 之選項，選取資料集之檔案夾之後，此工具會自動將每張圖標好方框範圍與物件標籤名稱，如圖。在標籤完成之後，若使用者發現方框之範圍不如預期的話，可透過介面左方的 Create Bounderbox、Delete Bounderbox等選項，修改方框範圍與標籤名稱。(圖7-10)

**3.3 開始自動標籤與中途暫停**



圖 11 自動標籤過程

在自動標籤過程之前，系統會跳出確認開始視窗，在執行自動標籤之後，若需要停下更改方框範圍或是調整標籤資訊，可點選 Stop 按鈕暫停。反之點選 Continue 即可繼續進行自動標籤。(圖11)

**3.4 存檔與更改**

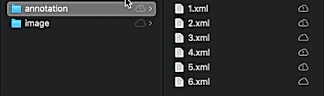
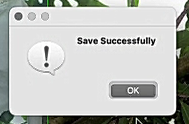


圖 12 存檔過程

在完成自動標籤之執行之後，標籤好之資料將會自動以XML格式存檔於先前所選取之資料夾。點選存檔選項也可以更改原有之標籤資料。(圖12)

1. **貢獻說明**

此專案將現有的Single Shot Multibox Detector物件檢測演算法與自己通過Pyqt5套件建立GUI應用見面結合。改善先前在標籤物件所需繁雜與耗時的過程，利用Single Shot Multibox Detector影像辨識模組為影像進行自動化標註，並提供了完善的調整資料、存取與檔案整理之功能，大大提升了製作資料集之效率。

1. **參考文獻**

**《[LabelImg 影像標註工具使用教學，製作深度學習用的資料集](https://blog.gtwang.org/useful-tools/labelimg-graphical-image-annotation-tool-tutorial/" \o "LabelImg 影像標註工具使用教學，製作深度學習用的資料集)》**。2017/11/29取自<https://blog.gtwang.org/useful-tools/labelimg-graphical-image-annotation-tool-tutorial/>

**《PyQt - 快速指南》**。2021/10/21取自<https://iowiki.com/pyqt/pyqt_quick_guide.html>

**《YOLO演算法詳細解析》**。2020/06/09取自<https://www.getit01.com/p20190608048530654/>

**《Read And Write Pascal Voc XML Annotations In Python》**，[**Faizan Amin**](https://faizanamin.web.app/)。2022/02/07取自<https://mlhive.com/2022/02/read-and-write-pascal-voc-xml-annotations-in-python#:~:text=Pascal%20VOC(Visual%20Object%20Classes,and%20train%20Machine%20Learning%20models>.