
大綱

- Github repo link
- 動機與介紹
- 相關研究
- 方法
- 實驗
- 結論
- Reference

Github repo link :

<https://github.com/liuyy3364/Introduntion-to-Artificial-Intelligence-Final-Project>

動機與介紹：

動機：

最近疫情頻傳，為了防止疫情擴散，減少人與人之間的接觸是一大重點，但有些場所又不適合禁止人員進入，因此人流管控的重要性就被突顯出來，而在這種情況下，採用人工計算顯然不是明智的方法，所以我們決定做一個可以進行人流管控的系統框架，讓其他使用者可以透過這個框架簡單的做出符合他們要求的人流管控系統。

介紹：

我們的框架是利用結合 Object Detection 與 Multiple Object Tracking (MOT) 功能，做出可以偵測並追蹤人員的框架，並提供簡單的出入計算與 web server，讓使用者可以方便根據其需求更改系統。

相關研究：

要達成我們的目標，有兩個重點，一是要可以偵測出人來，二是可以持續追蹤被偵測出來的人，這兩點分別可以找到相關研究：Object Detection 與 Multiple Object Tracking (MOT)。

Object Detection：這裡探討兩種方式：

Haar-like feature 與 YOLOv5。

Haar-like feature 的部分是四種長方形的特徵：

two-rectangle feature type (horizontal/vertical)

three-rectangle feature type

four-rectangle feature type

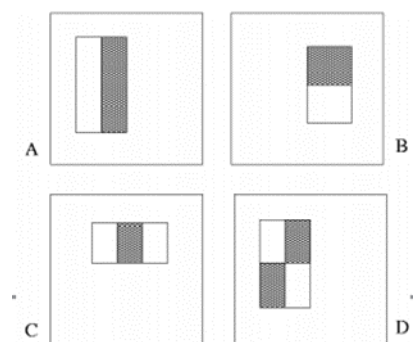


Figure 1. Haar-like feature

而 YOLO 簡單的概念為：

- Divide an image into $S \times S$ grids.
- Each grid has B bounding boxes.
- For each bounding box of a grid, predict if it has an object and the best position & size.
 - 5 information: x, y of the center, w, h , confidence
- For each grid, predict which object it has (C class probabilities) (only one object).

Figure 2. YOLO_intro

Multiple Object Tracking (MOT)：這裡介紹三種方式 SORT、DeepSORT、JDE。

SORT 由 Kalman Filter, Hungarian Algorithm 組成，Kalman Filter 可以大概預測物體下個 frame 的位置，而 Hungarian Algorithm 則是一個任務分配的組合最佳化演算法。

而 DeepSORT 就是 SORT 再加入了外觀的訊息來匹配前後 frame 的 object。

最後 JDE 就是將 object detect 和 appearance embedding 融合在同一個網路裡一起訓練，直接算出 detection location, class, embedding feature。

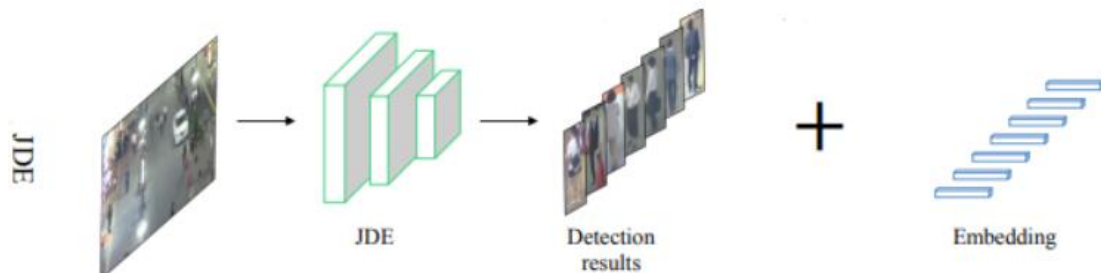


Figure 3. JDE intro

方法：

在主要框架中我們實作了四個功能，分別是 Detect、Track、Display、Optimize。

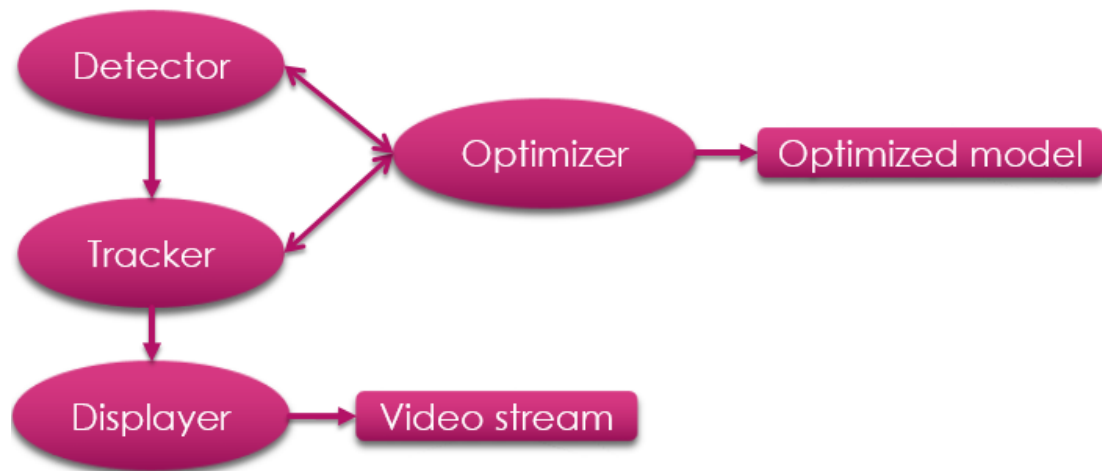


Figure 4. Arhcitecture

Detect 部份我們採用了上述 Haar-like feature 與 YOLOv5 兩種方法，Track 的部分使用了 SORT，Display 的部分採用了 OpenCV 和 Flask，Optimize 的部分採用了 Netadapt。

實驗：

我們利用我們自己做的 framework 做一個人流計算器。

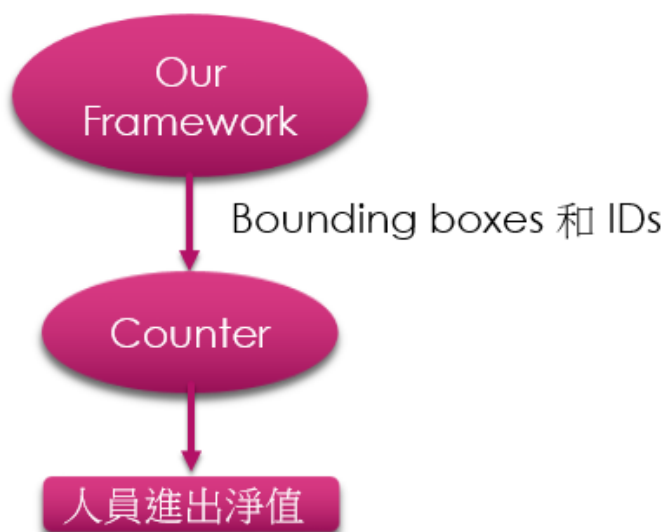


Figure 5. Arhcitectur

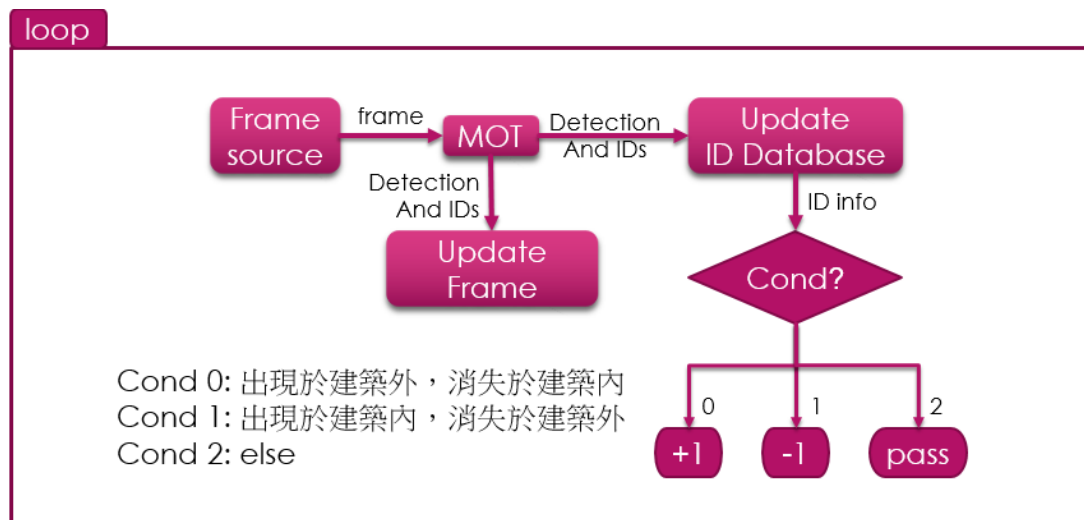


Figure 6. Flow Chart of counter

在實驗中我們分別測試兩種 Detector：用 Haar-like feature 與用 YOLOv5，運行之差異。

結果：

結論：

References：

Haar-like feature：

https://www.researchgate.net/profile/Michael-Jones-66/publication/3940582_Rapid_Object_Detection_using_a_Boosted_Cascade_of_Simple_Features/links/0f31753b419c639337000000/Rapid-Object-Detection-using-a-Boosted-Cascade-of-Simple-Features.pdf

YOLOv5：<https://github.com/ultralytics/yolov5>

SORT：<https://github.com/abewley/sort>

Netadapt：