

逢 甲 大 學

資訊工程學系

專題研究報告

拯救食物大作戰-整合擴充實境與食物  
效期自動辨識之多人食材管理系統

指導教授：林明言 老師

學生：莊舒媛 D0845278

鍾侑庭 D0886019

蘇展偉 D0879612

中 華 民 國 一 百 一 十 一 年 十 二 月

# 摘要

冰箱是個被遺忘的時光機，放進去的食物很多都會被遺忘，再次打開時會突然發現一罐過期的牛奶、醬料，當想下廚時發現食材發霉/發芽。合租冰箱的食物突然不見等等。為減少食物浪費及室友間的和諧，本專題計畫為發展出一個完善的多人食材管理系統為主，並利用自動化功能計算食品到期日、減少繁瑣輸入資料的步驟、使用影像辨識及 AR 技術分辨食物的持有人、食物種類等，以有別於市面上的方式，讓使用者追求更佳完善的使用者體驗。

關鍵字: 手機跨平台 App、影像處理、食材管理

# 致謝詞

首先要感謝我們的指導教授：林明言老師，老師都額外犧牲私人時間及心力，每週都跟我們討論專題的內容、提供研究方向、指導及建議等等，萬分感激教授在專題的各個方面對我們的協助，讓我們的專題能夠順利完成。

另外也要感謝曾經教導過我們的老師，因為在你們熱心的教導及栽培下學習，現在才可以使用所學的知識完成專題。

# 目錄

摘要 .....	I
致謝詞 .....	II
目錄 .....	III
圖目錄 .....	V
表目錄 .....	VI
第一章 緒論 .....	1
1.1 研究背景 .....	1
1.2 研究動機 .....	1
1.3 研究目的 .....	2
1.4 研究方法 .....	3
1.5 工作分配 .....	4
第二章 相關文件回顧 .....	6
2.1 食物辨識及解析 .....	6
2.2 套件介紹 .....	6
2.2.1 辨識端 .....	6
2.2.2 應用端：Dart Package .....	7
第三章 相關文件回顧研究方法 .....	8
3.1 研究架構 .....	8
3.2.1 影像辨識 .....	8
3.2.1 模型訓練流程 .....	11

第四章 實作結果 .....	15
4.1 系統概述 .....	15
4.2 系統特色 .....	15
4.3 系統流程 .....	15
4.3.1 首頁 .....	15
4.3.2 食材庫存 .....	16
4.3.3 發票食材登入 .....	18
4.3.4 食材辨識登入 .....	19
4.3.5 實景辨識 .....	20
第五章 結論 .....	21
5.1 遇到的困難與解決方法 .....	21
5.2 未來展望 .....	21
參考文獻 .....	23

# 圖目錄

圖 1 系統架構圖 .....	3
圖 2 以 COCO 作為測試資料集測試 YOLOV3 V4 性能 [2].....	10
圖 3 YOLOV3 V4 V5 版本下在 COCO 測試資料集比對 [10].....	10
圖 4 TENSORFLOW 訓練後 MAP 數據 .....	12
圖 5 由 TENSORFLOW 轉換為 TFLITE 檔後 MAP 數據 .....	13
圖 6 實體辨識模型流程圖 .....	13
圖 7 APP 首頁系統介面圖 .....	16
圖 8 APP 首頁系統流程圖 .....	16
圖 9 APP 食物庫存系統介面圖 .....	17
圖 10 APP 庫存系統流程圖 .....	17
圖 11 發票登入流程圖 .....	18
圖 12 資料庫端的資訊 .....	19
圖 13 APP 食材辨識系統流程圖 .....	20

# 表目錄

表 1 工作分配表 .....	5
表 2 模型 MAP .....	12

# 第一章 緒論

## 1.1 研究背景

在生活中過程中，我們會因人生階段不同，而會有住在不同的地方，從一開始的跟家人共住，上學時期有可能要跟同學外宿，畢業後再之回到跟家人共住或是搬出家裡住等等。無論在哪個階段我們都需要進食，不管日常多麼繁忙，每天攝取食物是人類日常活動也是生存本能。但太過日常的活動也讓民眾容易遺忘還放在冰箱或是櫥櫃的食物，一些尚未食用或啃食過的食物，被使用者放置到過了有效期限造成食物的浪費。在此時你就會發現原來食材管理是一件很繁瑣的事。許多人在超市、菜市場買的食物原材料回家後，誤以為可以一直放在冰箱裡就可以存放好長一段時間，因此常年累積許多食物。當下次想要拿出來使用時食物已經腐壞，從而造成食物浪費。另外有些民眾在購買牛奶、豆漿、雞蛋等食品時不會留意包裝上的到期日，加上食物一買多就會不小心忘記吃掉，並放到到期，同時也把保存期限、有效期限、賞味期限搞混，有些食物雖然過了賞味期限，但是其實該食材尚未壞掉，只是喪失了最佳賞味期的口感，但食物一旦過了有效期限就不建議使用者繼續食用造成身體的危害。又或是在準備食材時才發現缺少了某項食材，而需要再次出門購買，相信這是每個人都有的經驗。

## 1.2 研究動機

鑒於對於食材管理的需求，我們希望可以完善的食材管理軟體來處理生活上與食材有關的大小問題，我們認為手機是一個來達成這件事情相當好的終端裝置，現代人手機不離身，加上手機的運算能力也越來越好，手機作為生活貼身助理的定位越來越符合。加上食材管理方面，需要計算零碎的食材到期日，每件食材的時間還不盡相同，需要定時通知採買清單，需要記憶食材的消



耗狀況，還要處理多人共用冰箱所衍生的問題，如：分不清購買的食材等。適合使用應用程式來處理這些瑣碎的大小事，為現代人節省下在這方面所花費的時間。

雖然現今手機應用有諸如「食材 7 號 Lite」、「冰箱小管家」、「冰箱庫存管理」、「雲冰箱管家」、「你的冰箱裡有什麼」、「No Waste」等食材管理軟體，但多數 Apps 在多個食材登入部分，都是採用食物條碼一一列出清單的方式，使用者必須手動輸入各個食物品項數量，在大量食物登入時會減少使用者登入食材的耐心及準確性。為了讓使用者減少輸入資料數量，也沒有對於多人共享冰箱的解決方案。我們希望可以完成一個較為易用，更加簡易完成目標，且具有多人共用儲存資源功能的應用程式。

### 1.3 研究目的

一個食材管理的應用程式，由食材輸入開始，以多種登入方式，讓使用者可以選擇最適合自身的方式，快速的輸入購買資訊。如採用電子發票掃描 QR Code 將電子發票內的產品明細過濾，將食材的明細儲存下來，以這類能一次輸入多種食材的輸入方式取代原先的模式，減少使用者需要清單式手動輸入資訊，也能在使用者輸入完畢後可以按照食物類別預估出購買食材的有效日期等資訊，大幅度減少繁瑣的登入過程，讓使用者不會對登入食物部分產生厭煩感，增加使用者使用頻率。

本研究能從使用者購物時藉由家中所缺食材而產生出購物清單開始選購，而購買完的食物可藉由各類的輸入方式，更有效率的列入品項，系統也會一併記下食材的儲藏地點及有效日期，能讓使用者藉由本系統高效率的管理家中的各項食材。而本研究同時也針對多人同住者做出相應功能，藉由系統便利的方

式分辨出所屬食材。我們期盼透過本研究讓使用者能更有效率的享受生活，並順勢的減少食物的浪費亦可省下不小費用。

1.4 研究方法

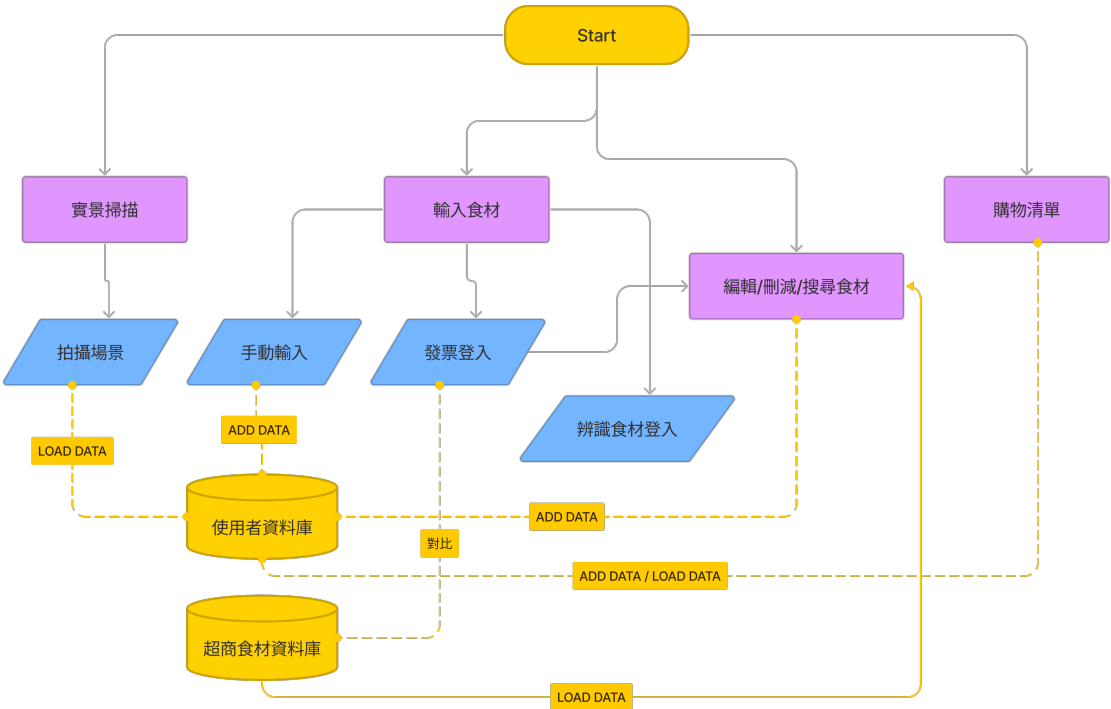


圖 1 系統架構圖

本系統架構如圖一，整個應用系統包括食材管理系統、食材登入系統、食材辨識系統、食材登入刪減系統。在 Server 端的「超商食材資料庫」中將儲存大量的食材資訊，系統在 Offline 時將透過網頁爬蟲出訊息根據食物的品項以及類別，做出食材的分類及統整，並將食材所屬類別的資訊，如：有效期限、食材類別等資訊儲存在「超商食材資料庫」中。使用者可以透過「輸入食材」功能以“手動輸入”、“發票登入”、“辨識食材登入”的形式輸入購買資料，使用者可以手動將食材資訊登入至系統中。如果使用者使用發票登入，模組會將使用者輸入的食材資訊與「超商食材資料庫」進行比對與整合，在使用者微調後將資訊儲存至「使用者資料庫」。而如果購買的食材沒有發票可以做登入，使用者可以透過拍照的方式，讓系統透過辨識模型讀出該項食材，輸出相關資

訊，再讓使用者登入還缺少的資訊後登入儲存至「使用者資料庫」。另外，使用者可透過「實景掃描」模組由相機直接辨別收納場景，讀取資訊，並再介面上列印出這圖中的食材擁有者及資訊。

## 1.5 工作分配

姓名	工作內容
莊舒媛	<ul style="list-style-type: none"><li>● 文件編撰</li><li>● Debug Program</li><li>● 程式運作環境設置</li><li>● Mongo DB 建置及串接</li><li>● UI Design</li><li>● User Story 編寫</li><li>● Flutter 前端介面開發</li><li>● Firebase 資料串接</li><li>● 整合</li></ul>
鍾侑庭	<ul style="list-style-type: none"><li>● 文件編撰</li><li>● Debug Program</li><li>● 程式運作環境設置</li><li>● 商場品項爬蟲</li><li>● Mongo DB 串接</li><li>● 模型訓練、串接</li><li>● 掃描系統</li><li>● 整合</li></ul>
蘇展偉	<ul style="list-style-type: none"><li>● 文件編撰</li></ul>

	<ul style="list-style-type: none"> <li>● Debug Program</li> <li>● 程式運作環境設置</li> <li>● Tflite 串接</li> <li>● 影像辨識模型訓練</li> </ul>
--	--

表 1 工作分配表

## 第二章 相關文件回顧

### 2.1 食物辨識及解析

在[4]中，曾成訓練自己的 yolo 模型。使用 Labellmg 標記加入所需要的食物圖片辨識，包括食物的包裝、未包裝圖、多重標記等。

在[5]中，可得知食品保存期限、食品有效期、賞味期限等的分別，知道在被丟棄的良好食物中有 90% 的人會常會因為不了解食材日期標示之含義，而丟棄仍可食用的物品。

在[9]中，曾永翔把混沌理論結合字詞爬蟲程式開發出用以收集不同網站的資料的程式。當中的文字探勘技術及資料庫擷取，可以更進一步加深對於食材資料庫的拓展。

在[8]中，施登騰提及 AR 擴增實境的多個應用，當中包括了博物館把展覽物品的資料、歷史及其他資料以 AR 的形式展示等研究。

總合市面上功能比較重面的食材管理軟體，包括:「冰箱庫存管理」、「雲冰箱管家」、「No Waste」。以下會以這三款軟體的食材登錄、多人管理、做分析。

在食材登錄方面，「冰箱庫存管理」只有手動輸入資料的登錄方式，「No Waste」則包括掃描條碼以及手動輸入，但是因使用美國資料庫的原因，有可能會在描條碼後搜尋不到對應食品，最後「雲冰箱管家」只有語入輸入的方式。

上述的應用程式中，「No Waste」及「雲冰箱管家」都支援多人使用，「No Waste」針對的使用群族是西方國家，是唯一個有家庭成員功能的應該程式，多個使用者登入同一帳號(電郵)，再在帳號裡建立不同的成員身份，以方便集中管理。「雲冰箱管家」則是給亞洲人使用，軟體結合了微信的朋友圈的社群網路功能，透過每筆資料的更動發佈到動態上，從而知道誰變動了冰箱的食品。

我們希望結合以上文獻應用、優點以及[10]的格式統一特性，建立一個以 3D 形式與實體空間有所連動的多人食材管理應用程式，用以改善食品過期食品浪費的問題。

### 2.2 套件介紹

#### 2.2.1 辨識端

- YOLOv4、YOLOv4 Tiny & OPENCV & Darknet

因需要進行大量的圖片特徵處理，所以會在影像辨識中使用 OpenCV 把圖片加載並進行縮放、清除雜訊等功能，然後使用 OpenCV 中的 Dcarknet 套件讀入要處理的圖片及 YOLOv4 的權重文件進行模型訓練。

- TFlite Package

把從 YOLO 訓練好的 .h5 檔模型封裝轉換成 .Tflite 檔後，可以直接在 Android、iOS 等設備上使用。

### 2.2.2 應用端：Dart Package

- cloud\_firestore & firebase\_auth & firebase\_core

用此套件與 Cloud Firestore API & Firebase Authentication API & Firebase Core API 做串接，做用於 App 端使用 Firebase 的資料庫作為使用者登入及紀錄相關資訊的用途。

- qr\_code\_scanner

用來進行二維條碼掃描使用，可將條碼解碼為文字。

- mongo\_dart

用純 Dart 實現的 MongoDB 的服務器端驅動庫。用此插件幫助我們與 MongoDB Atlas 做資料傳輸、編輯、查詢等的功能。

- json\_annotation & json\_serializable

此插件用於處理應用端輸入資料至 MongoDB 前轉換輸入成 Json 文件至 MongoDB 方可處理相關資料，以及從 MongoDB 取出資料至應用端時的資料解析方可應用。

## 第三章 相關文件回顧研究方法

### 3.1 研究架構

本系統的主要架構為圖一系統架構圖。首先，使用者由相機掃描電子發票上面的 QR code 將發票明細內容讀入，轉換成能呈現在介面上的資訊，讓使用者將非食材的內容刪除及現有食材的資訊做編輯，而食材則以食材名稱、購買金額、購買數量、保存期限等的紀錄資訊，再由輸入食材模組賦予擁有者 id 後，存放入使用者資料庫中。

接下來，受到紀錄的食材就會由食材管理模組提供互動，使用者在消耗食材時，則可以手動紀錄消耗狀況，並在即將耗盡時獲得通知方便使用者加入首頁購物清單，也可於購物清單中手動建立品項。食材庫存管理模組中也會在使用者使用完畢該項食材時匯入耗盡食材進入購物清單。

實體辨識的部分分作登入使用與顯是實體資訊時使用，兩者可共用模型，登入時，可將食材類別進行辨識，輸入其餘資訊用做無發票時的登入方式。

實體資訊辨識則為辨識類別過後，進入資料庫尋找相符的類別品項，在介面端利用辨識座標生成虛擬按鈕，實現 2D AR 的場景互動。

### 3.2 方法步驟

本次的研究方法主要分為以下步驟，我們會以數個小節的方式做說明。

#### 3.2.1 影像辨識

- 卷積神經網路簡介

卷積神經網路(CNN)目前主要應用於圖片處理及數據預測，其特點是能夠有效保留圖片的特徵及把圖片數據化，原理是通過以下三層來實現，第一層是卷積層(Convolution Layer)提取特徵，保留圖片中的結構，並從中提取出其特徵儲存。第二層是池化層(Pooling Layer)，作用是把卷積層提取的數據降維減少參

數以減少計算成本，池化為更少像素並具有原本特徵的圖片。最後是通過全連接層(Fully Connected Layer)以之前圖片特徵數具轉化為數據並以投票(權重)的方式輸出結果。

- YOLO

YOLO (You Only Look Once) 是一個 [3] one-stage 的物件偵測演算法，於 2015 年 6 月由 Joseph Redmon 推出，隨後分別在 2016 年及 2018 年推出 YOLOv2 和 YOLOv3，可惜在 2020 年作者宣布退出電腦視覺這領域，同年份由 Alexey Bochkovskiy、台灣中研院資訊所的王建堯博士和廖弘源所長合力推出新研發的 YOLOv4 [2]。

原理是將整個影像輸入當作一個整體，透過 CNN 模型可以辨識多個圖中物件的類別及對目標物位置進行預測，特點是速度比 R-CNN 等 two-stage 演算法的辨識速度快、經過優化及改良令到 YOLOv4 能夠實現實時偵測並維持高準確度。

在 YOLO 版本選擇方面我們做出了以下的對比，如下圖?所示，YOLOv4 在精度方面升了 10%,FPS 提升了 12%。



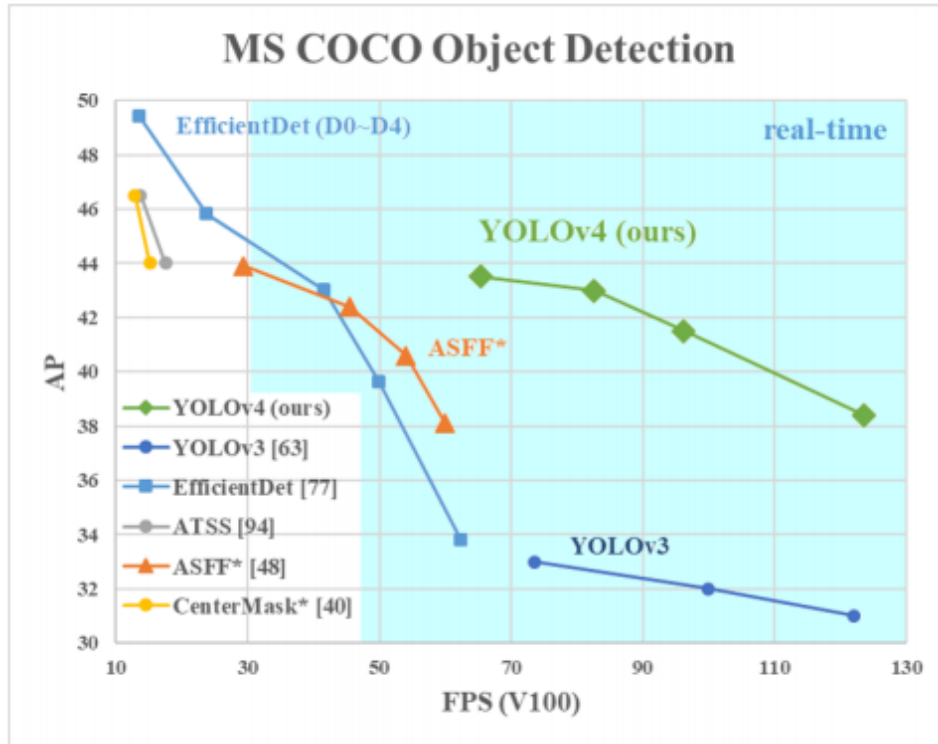


圖 2 以 COCO 作為測試資料集測試 YOLOv3 v4 性能 [2]

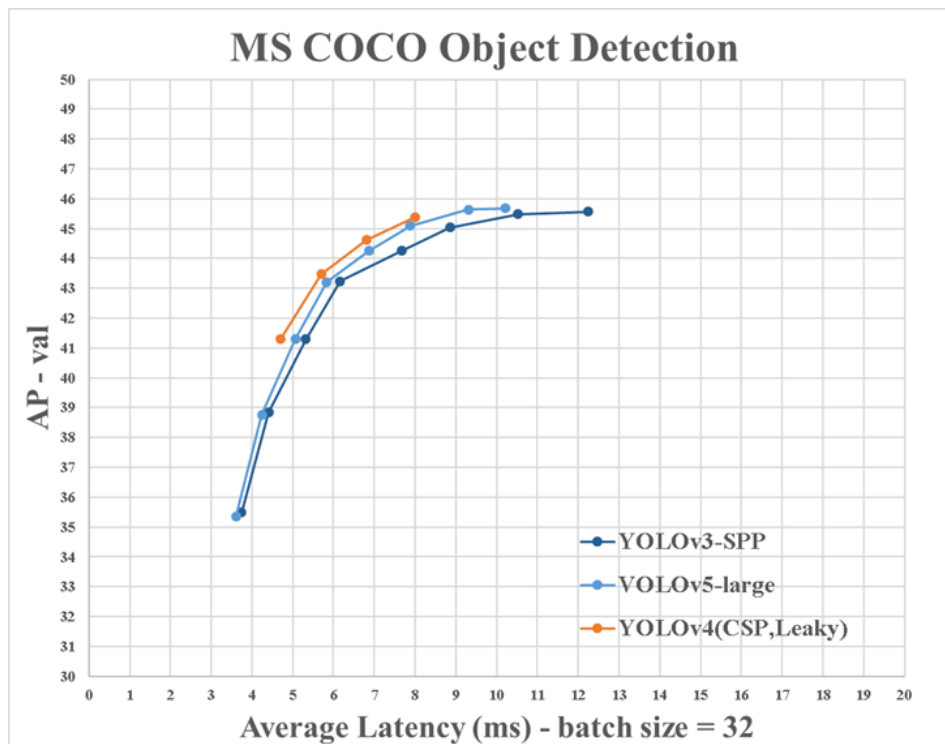


圖 3 YOLOv3 v4 v5 版本下在 COCO 測試資料集比對 [10]

通過圖 2 比較 YOLOv3、v4、v5 對比，YOLOv4 在整體表現最好，YOLOv5 性能上比 YOLOv4 差一點點，但是靈活性較強，具四種模型可以選擇。總合性能來說 YOLOv4 優於 YOLOv3。而在速度相同時 YOLOv5 的表現優於 YOLOv3。因此我們將以 YOLOv4 實作影像辨識模型。

- TensorFlow

TensorFlow 是除了 YOLO 以外有名的深度學習框架，2015 年由 Google 開發，模型可轉存為.h5、.hdf5、.ckpt、.pb 檔，加上與 TensorFlow Lite 同源，在轉換過程中相容性較高及損失較低的 mAP。

- TensorFlow Lite

TensorFlow Lite 是 Google 針對移動手機端設計的神經網絡計算框架，支持手機硬體加速的功能，可以同時在 Android、iOS 上做跨平台運行，模型體積對比 TensorFlow 及 Pytorch 等檔案小。我們會把 YOLO 訓練出的模型.h5 檔轉換成.tflite 檔放進系統作最後的影像辨識使用。

### 3.2.1 模型訓練流程

由於網路上不存在與本次專題相符的相關資料集，因此需要自己收集資料集圖片及進行標記再利用 YOLO 演算法進行模型訓練並生成所需的權重檔(.weights)，相關流程圖如下：

過程中，已成功將模型通過 yolov4-tiny、yolov2-tiny、TensorFlow 訓練完成轉成 TFlite 格式後 mAP 如下：

模型	mAP(mean average precision)
Yolov4-tiny	24.13%
Yolov2-tiny	53.59%
TensorFlow	51.32%

表 2 模型 mAP


	<code>model.evaluate(val_data)</code>
1/1 [=====] - 6s 6s/step	
	<pre>{   'AP' : 0.5228951,   'AP50' : 0.7501319,   'AP75' : 0.5424295,   'APs' : -1.0,   'APm' : 0.0,   'APl' : 0.5282184,   'ARmax1' : 0.35319903,   'ARmax10' : 0.6026038,   'ARmax100' : 0.6432377,   'ARs' : -1.0,   'ARm' : 0.0,   'ARl' : 0.64694136,   'AP_/apple' : 0.6390971,   'AP_/banana' : 0.38871288,   'AP_/bell_pepper' : 0.7802137,   'AP_/bottled_drinks' : 0.07139426,   'AP_/boxed_drinks' : 0.7342763,   'AP_/cabbage' : 0.07268977,   'AP_/carrot' : 0.9,   'AP_/cauliflower' : 0.9,   'AP_/kiwi' : 0.5757426,   'AP_/lemon' : 0.21299261,   'AP_/orange' : 0.63781995,   'AP_/tomato' : 0.36180168}</pre>

圖 4 TensorFlow 訓練後 mAP 數據

```

model.evaluate_tflite('food.tflite', val_data)

13/13 [=====] - 31s 2s/step

{'AP': 0.513285,
 'AP50': 0.70826924,
 'AP75': 0.53154343,
 'APs': -1.0,
 'APm': 0.0,
 'APl': 0.5193814,
 'ARmaxl': 0.35676026,
 'ARmaxl0': 0.5702976,
 'ARmaxl00': 0.5822848,
 'ARs': -1.0,
 'ARm': 0.0,
 'ARl': 0.5839515,
 'AP_/apple': 0.5445545,
 'AP_/banana': 0.3990099,
 'AP_/bell_pepper': 0.7580858,
 'AP_/bottled_drinks': 0.064851485,
 'AP_/boxed_drinks': 0.7326733,
 'AP_/cabbage': 0.06732673,
 'AP_/carrot': 1.0,
 'AP_/cauliflower': 0.9,
 'AP_/kiwi': 0.5757426,
 'AP_/lemon': 0.21906576,
 'AP_/orange': 0.5963696,
 'AP_/tomato': 0.30174005}

```

圖 5 由 TensorFlow 轉換為 tflite 檔後 mAP 數據

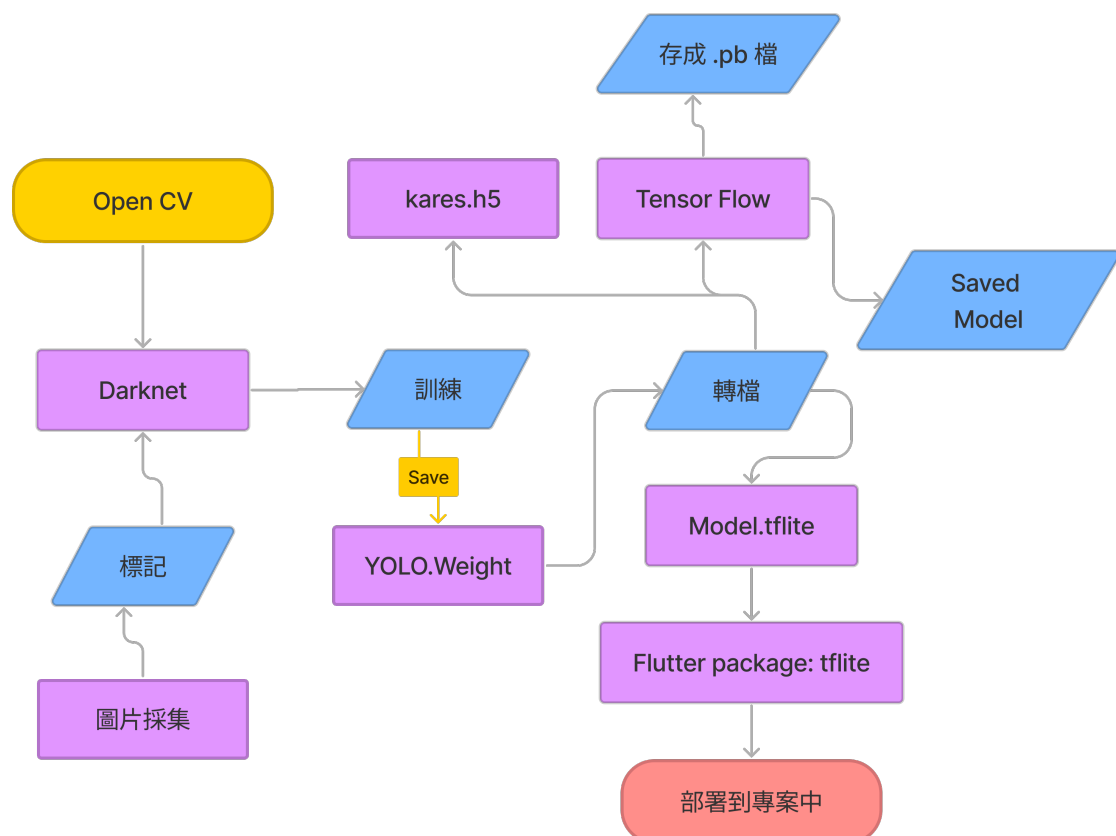


圖 6 實體辨識模型流程圖

不過由於轉檔所用 tensorflow 與 keras 版本更新頻繁以及語法替換快速，又不向前相容，flutter 的 packages-tflite 無法與其進行整合，以至於部署至 flutter app 尚沒成功，這對我們是一件艱鉅的挑戰。

## 第四章 實作結果

### 4.1 系統概述

此系統是希望藉著一般用戶的日常使用習慣讓使用者能夠以有效率且簡易的方式如：食材辨識、發票登入的方式紀錄並能管理所擁有的食材，且能夠在使用者的收納空間呈現收納的食材，方便使用者查找。

### 4.2 系統特色

在目前的世面上的食材管理的系統中，大部分以更加便利的手機 App 作為工具，然而現今市面上已上架的食材管理軟體已有不少，但我們為了給用戶更增加便利性，針對輸入食材的部分增添了額外的輸入方式，我們添加了多種的食材登入模式如：食材辨識或發票登入等...，以增加使用者登入食材的使用性，也以 AR 系統呈現食材收納資訊畫面方便使用者管理，本系統針對想要管理食材的使用者作為目標客群，以下是統整出來本系統的特色：

1. 操作簡易：減少其他相似系統較為繁瑣的流程，大量減少使用者的操作時間，增加使用的次數。
2. 能快速登入食材：除了手動輸入外，增加了發票系統及食物辨識系統，在沒有發票的商店購買也能透過相機辨識輸入商品品項，減少使用者手動輸入的機會。

### 4.3 系統流程

本系統以 App 為系統運行方式，以下根據 App 操作流程做說明：

#### 4.3.1 首頁

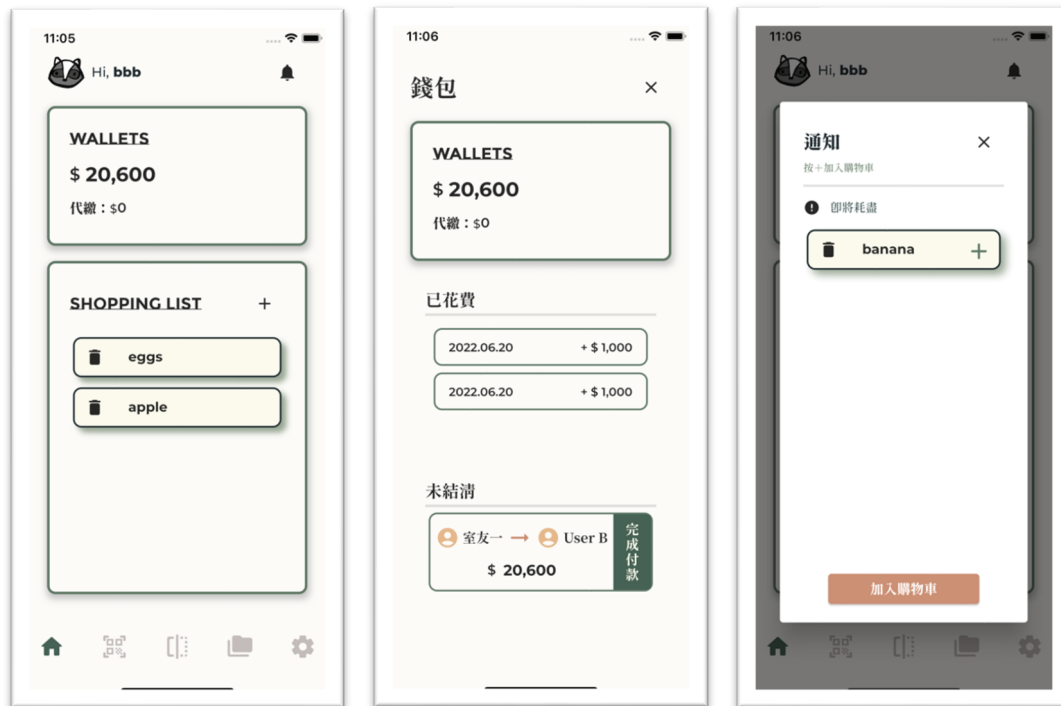


圖 7 App 首頁系統介面圖

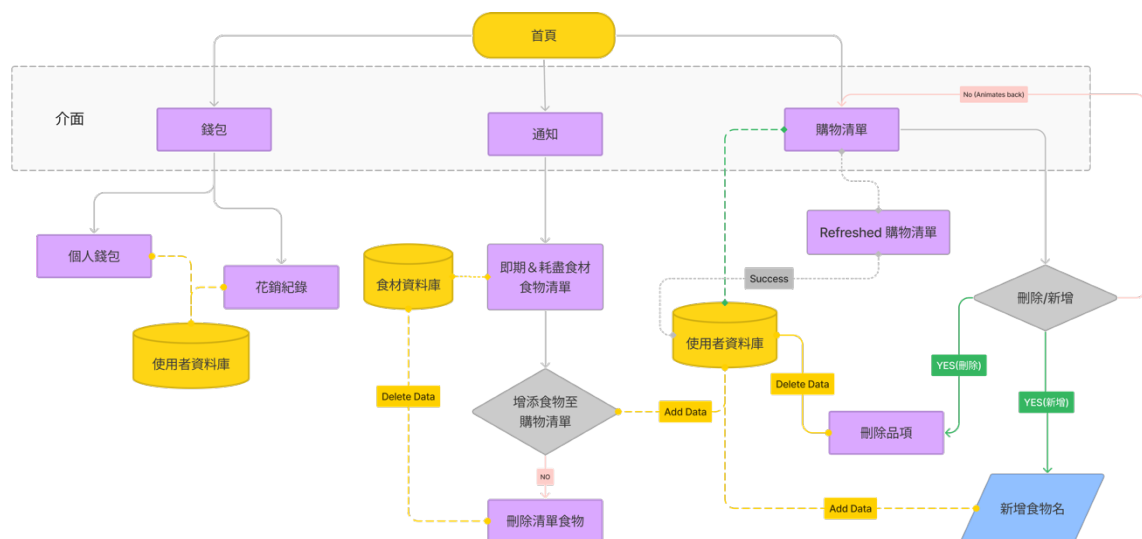


圖 8 App 首頁系統流程圖

本 App 在首頁區呈現了使用者錢包以及即將過期與已消耗完畢的通知區，使用者可以將食物透過通知區加入首頁購物清單的中，錢包部分也能根據每次輸入食材的明細紀錄每個使用者的花費的金錢，如果有多位使用者也能分別紀錄各自購買的金額作為分帳紀錄。

#### 4.3.2 食材庫存



圖 9 App 食物庫存系統介面圖

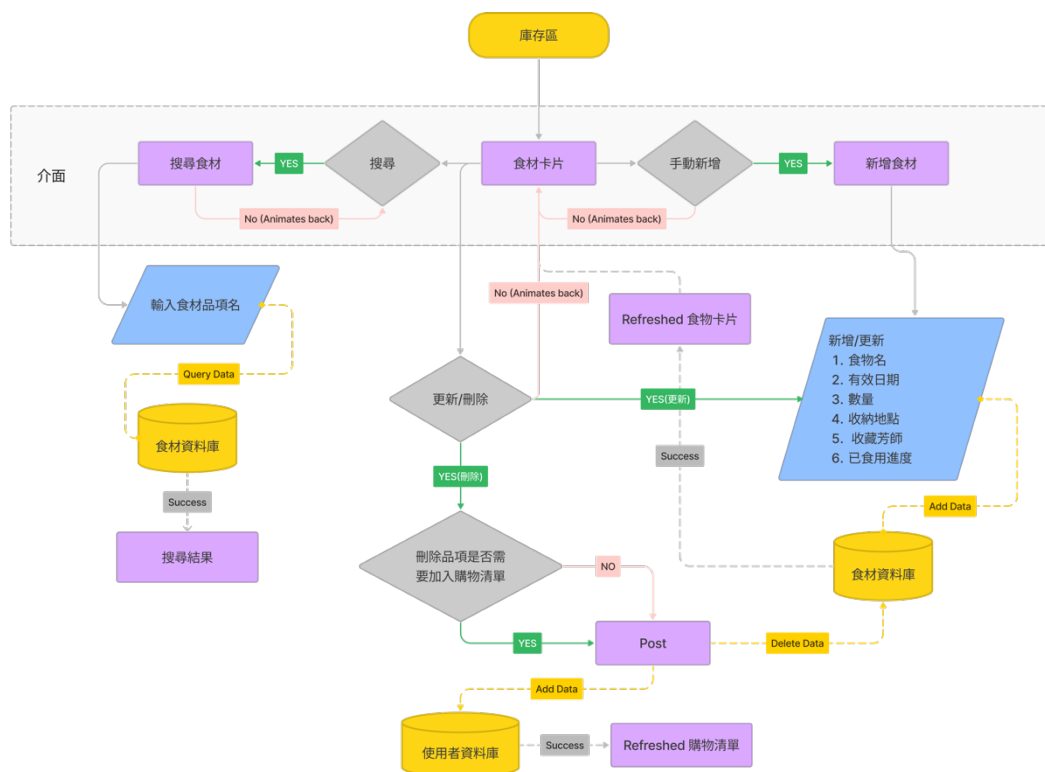


圖 10 App 庫存系統流程圖



本 App 在庫存區可以提供使用者查看及搜索自身擁有的食材卡片，快速查看剩餘日期，及收納地點。使用者可以對擁有的食材做食材編輯及刪減方便做紀錄及使用者也能在卡片區快速查看將要過期食物以利於提醒使用者盡早消耗掉食材，當食材耗盡時或作刪減時也能夠幫使用者匯入首頁的購買清單。

#### 4.3.3 發票食材登入

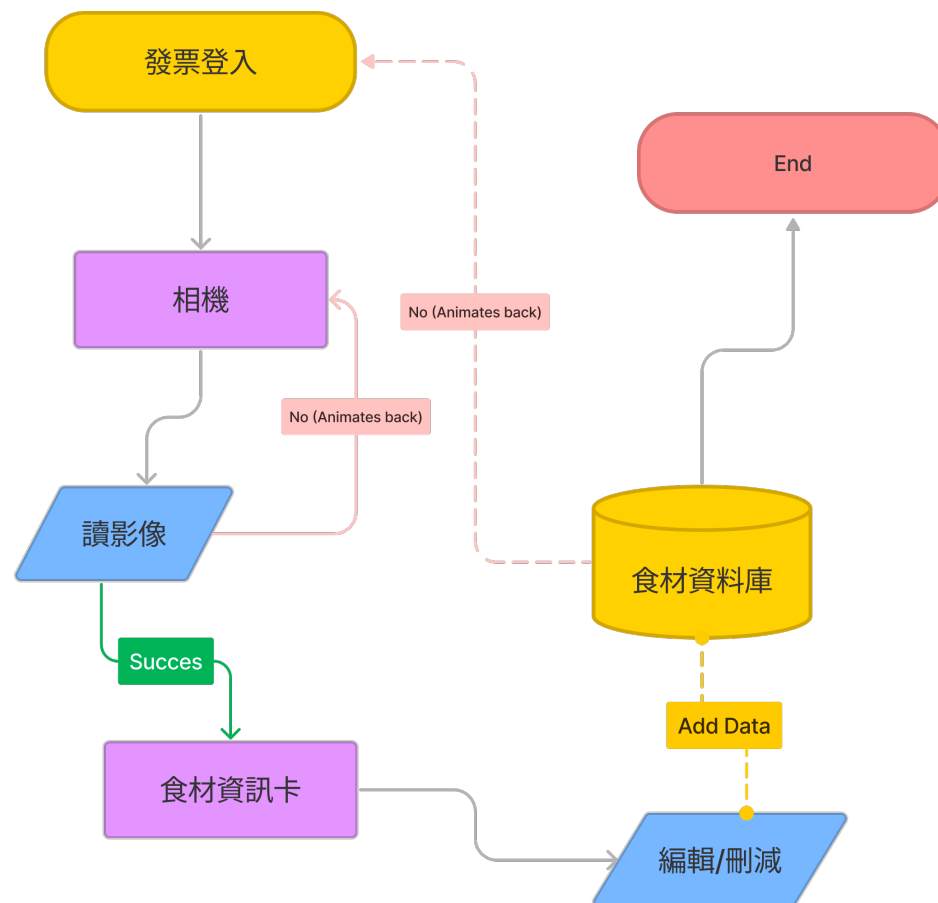


圖 11 發票登入流程圖

```
_id: ObjectId('63879ab22ff64a55b4ad8966')
name: "生可可波堤"
count: "1"
cost: "50"
```

```
_id: ObjectId('63879ab22ff64a55b4ad8967')
name: "焗烤雙起司鮭魚三明治"
count: "1"
cost: "49"
```

```
_id: ObjectId('63879ab22ff64a55b4ad8968')
name: "肉鬆玉子雙手卷"
count: "1"
cost: "49"
```

圖 12 資料庫端的資訊

掃描 QRcode 讀取資訊，參考財政部電子發票二維條碼相關規則，建立食材登入模組進行相關解碼，可以讀入電子發票中的兩個二維條碼中的明細內容，內含有品名、價格、數量，並將明細內容傳入資料庫。

#### 4.3.4 食材辨識登入

為了應對沒有電子發票的狀況，利用物體辨識模型將食材類別登入，再手動修改登入列表的詳細內容，以簡化手動輸入困擾。

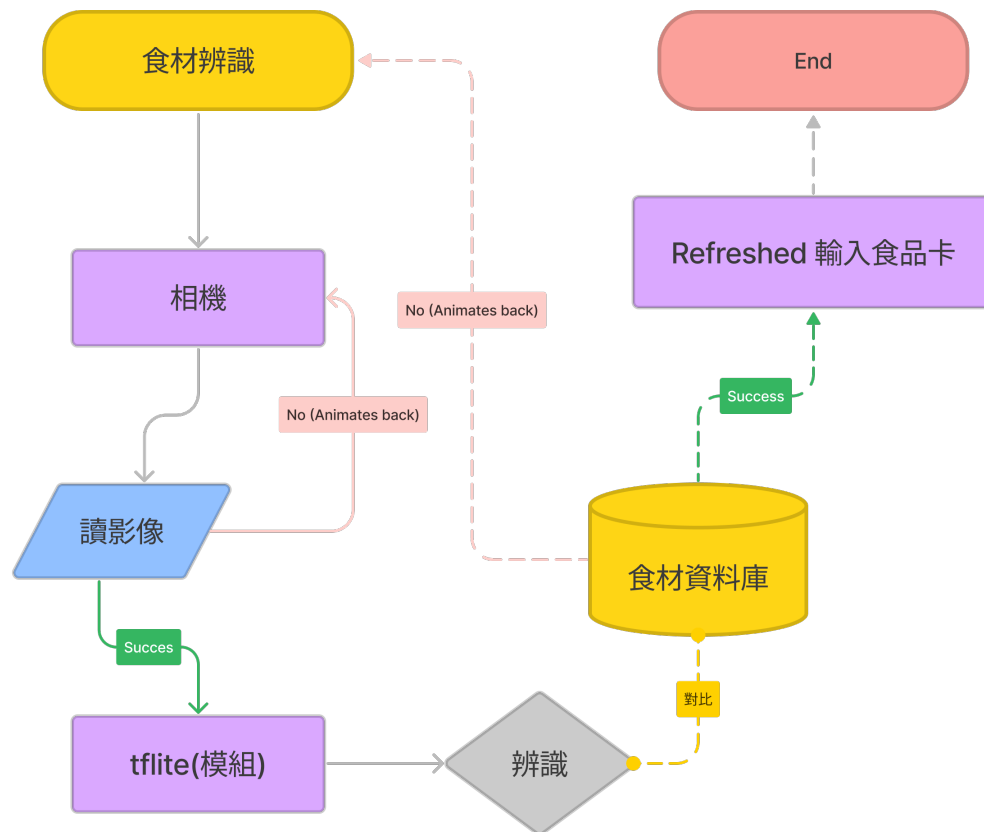


圖 13 App 食材辨識系統流程圖

#### 4.3.5 實景辨識

在 App 打開辨識功能後，系統會從攝相頭讀取影像截圖傳給 Tflite 模型進行辨識，並把辨識結果(label 與座標內容)回傳到辨識介面上。生成虛擬按鈕同時會跟資料庫做對比，找出有相同名稱的食品卡輸出 AR 資訊實現場景互動。

## 第五章 結論

當初打算想要開發這個系統的初衷是，想要增加我們紀錄的習慣減少減少不小心放太久過期無法食用的情況，所以嘗試了許多 App 去做使用，我們發現如果不有效的增加使用者登入食材的效率，使用者很難有下次打開的機會，所以萌生了能幫助我們快速紀錄的系統，為了達成這目標除了加入發票明細登入外，也同時利用了手機的鏡頭幫助我們完成想做的事情，加入了食物辨識的系統，為此系統添加更多多樣性。

### 5.1 遇到的困難與解決方法

- 研發功能與應用平台整合衝突：把影像辨識模型整合到 flutter 遇到很大的問題，當中因為 flutter 各個插件更新速度很快，插件之間的版本會互相影響，甚至到後期因為模型的參數出現問題，不能在 flutter 裡運行，後來在網上及在討論區發問，發現是從 yolo 訓練出來的.h5 檔案轉換去 tflite 檔的過程中出現問題。
- 跨平台的版本問題：因為本系統是以 Android 和 iOS 系統同步進行開發的，在技術上會運用到一些需使用插件，及辨識模型也有可運行的版本限制，因初期並未一一列出各種所需插件已固定各個環境開發的版本設置，在開發中發生了許多衝突，花費了一段時間去做解決。
- 使用者體驗測試尚未完善：由於進度整體系統開發上花費許多時間，尚未有機會進行完善的使用者體驗測試，無法妥善的納入專題，僅透過開發人員進行流程確認，未來希望能對此能更完善處理。

### 5.2 未來展望

- 結合食譜網站或應用程式做合作：能夠將根據食譜的材料需求，推測出使用者依據目前食材可以烹煮出的食物，更有效且最大化的運用已擁有食材。
- 與剩食餐廳去做相關合作：EX.明日餐廳，延續本系統想要減少浪費的理念，如果與這類餐廳合作能夠提高雙方更多的使用者應用程度及曝光，能夠擴大彼此相同的理念，更大的減少食物的浪費。
- 結合食物倉庫區給使用者健康建議：計算日朝使用者常食用的食物整理出用戶各類營養攝取的份量，已提供使用者有更完善的飲食攝取且能在冰箱同存在特定食物的組合的會提醒我們在同時食用時會有食物中毒的風險，幫助使用者更加健康的飲食。

## 參考文獻

- [1] “進貨退出或折讓證明單證明聯格式壹、電子發票,”附件 pp.1-9 頁,  
<https://www.ntbt.gov.tw/download/4674a3e26f2747cbbee8ef2d58d5f58d>
- [2] Institute of Information Science Academia Sinica, Taiwan : YOLOv4: Optimal Speed and Accuracy of Object Detection
- [3] Joseph Redmon, Santosh Divvala, Ross Girshick, Ali Farhadi. “You Only Look Once: Unified, Real-Time Object Detection,”*Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* , 2016, pp. 779-788.
- [4] CH. Tseng. “【採果辨識】建立自己的 YOLO 影像辨識模型 — 以柑橘為例,” Sept. 2018, [https://www.superlab.com.tw/food\\_expiry/](https://www.superlab.com.tw/food_expiry/).
- [5] “有效日期與食品安全，各國效期標示方式,”台美檢驗,  
[https://www.superlab.com.tw/food\\_expiry/](https://www.superlab.com.tw/food_expiry/).
- [6] 財政部電子發票 API. 財政部，2022,  
<https://www.einvoice.nat.gov.tw/APCONSUMER/BTC605W/>.
- [7] 馬偉雲，劉興寰，蔡瑜方，戴嘉宏，白明弘，范嘉仁，謝佑明，李朋軒，楊慕.“財政部電子發票 API,”CKIP Lab,  
<https://www.einvoice.nat.gov.tw/APCONSUMER/BTC605W/>.
- [8] 施登騰,“數位轉譯/博物館科技系列：從偽互動（pseudo-Interaction）談「AR 實境策展」技術實踐,” *Medium*, Mar. 2019.
- [9] 曾永翔。「Python 應用於字詞爬蟲與資料庫擷取之研究」。碩士論文，龍華科技大學資訊管理系碩士班，2021  
<https://hdl.handle.net/11296/c8qzd7>
- [10] Joseph Nelson, Jacob Solawetz, Responding to the Controversy about YOLOv5

<https://blog.roboflow.com/yolov4-versus-yolov5/>