逢　甲　大　學

自 動 控 制 工 程 學 系

畢 業 專 題 論 文

基於 YOLO11 影像辨識與重量感測之自動結帳系統

An Automatic Checkout System Based on YOLO11 Object Detection and Weight Sensing

指導教授：陳志成

學生：孫愷、黃柏勳、王丞佑

中華民國一百一十四年十二月十七日

致謝

本研究能順利完成，首先要誠摯感謝指導教授陳志成老師。從專題構想的發想階段開始，老師以其豐富的經驗與嚴謹的態度，耐心地引導我們逐步釐清問題與研究方向。當我們對影像辨識與重量感測的結合方式感到迷惘時，老師總是以深入淺出的方式提出建議，提醒我們要同時兼顧系統的實用性與可行性，讓專題能在有限的時間內穩健地發展。老師在過程中不僅提供技術上的指導，更以開放心態鼓勵我們發揮創意、嘗試不同的解決方案，使我們在研究與實作的過程中都能有所成長。

在此，也要感謝專題組員們在整個開發過程中的努力與堅持。從拍攝商品影像、整理資料集、訓練YOLO模型，到整合電子秤與介面設計，每一個環節都需要不斷嘗試與修正。過程中雖然遭遇了許多挑戰，例如模型辨識不準、秤重延遲、資料同步困難等問題，但大家始終互相鼓勵、共同面對。每一次討論、每一次測試的改進，都是我們一起完成這套自動結帳系統的重要基石。這段經驗讓我們深刻體會到團隊合作的重要性，也學會了在壓力與不確定中找到解決問題的方向。

此外，感謝在專題進行期間曾提供協助的同學與朋友，協助我們進行商品拍攝與測試，讓我們能更有效率地蒐集資料、優化系統。也感謝家人一路以來的理解與支持，在我們為專題忙碌時給予最大的鼓勵與包容。

回顧整個研究歷程，這不僅是一個技術專題，更是一段成長的旅程。透過開發這套結合影像辨識與重量感測的自動結帳系統，我們不僅提升了對人工智慧與自動化應用的理解，也培養了規劃、協作與實踐的能力。這段寶貴的經驗，將成為我們未來邁向更高挑戰的重要基礎。

摘要

本研究旨在設計與實作一套結合影像辨識與重量感測之自動結帳系統，以提升零售環境中商品結帳的自動化與準確性。由於實際結帳流程中條碼常因遮擋、反光或磨損而導致掃描失敗，店員需改以人工輸入而增加時間成本並提高錯誤風險且部分商品外觀相似或同品項存在不同容量規格時亦容易造成辨識混淆，影響結帳效率與顧客體驗。為改善上述問題本研究採用深度學習影像辨識模型YOLO進行商品偵測與分類，並在影像辨識結果可能對應同一品項但存在不同大小或容量時，啟動Arduino搭配HX711重量模組進行二次驗證，以提升判斷一致性。系統整合部分以Python為主要開發平台，建立使用者互動介面，使顧客在結帳時可自動觸發影像辨識，即時顯示辨識結果與價格資訊，並依情況進行重量驗證。實驗結果顯示本系統能有效辨識多種常見商品，並透過重量輔助機制降低誤判情形，成功實現一套兼具穩定性與實用性的自動結帳流程。綜合而言本研究驗證影像辨識結合重量感測於小型零售場域之可行性，並可作為後續智慧結帳系統擴充與場域導入之參考基礎。

關鍵詞：YOLO11、影像辨識、重量感測、自動結帳系統、智慧零售

孫愷、王丞佑、黃柏勳謹誌

中華民國一百一十四年十二月

Abstract

This study proposes an automatic checkout system that integrates image recognition and weight sensing to improve the accuracy and efficiency of retail checkout. To overcome barcode scanning failures and recognition ambiguity caused by similar product appearances or different package sizes, a YOLO-based deep learning model is used for product detection and classification, while an Arduino with an HX711 weight module performs secondary verification when necessary. The system is implemented in Python with a user interface that enables real-time recognition, price display, and weight validation. Experimental results show that the proposed approach effectively reduces misclassification and demonstrates the feasibility and practicality of combining image recognition and weight sensing for small-scale retail applications.

Keywords: YOLO11, Image Recognition, Weight Sensing, Automatic Checkout System, Smart Retail

目錄

圖目錄

表目錄

第一章 緒論

1.1 研究背景及動機

隨著人工智慧與自動化技術的快速發展，智慧零售（Smart Retail）逐漸成為商業模式轉型的重要方向。近年來許多零售業者開始導入自助結帳、自動盤點及無人商店等技術，以降低人力成本並提升顧客體驗。在台灣7-Eleven與工研院合作的X-STORE亦展示結合多種感測與系統整合的「拿了就走」示範店，代表本地智慧零售落地的關鍵里程碑[1]；韓國也因少子化與人力成本上升，加速導入無人超商，無人門市由2019年208間增至3,310間，顯示多感測與即時辨識正成零售主流[2]。

在實際零售結帳流程中，條碼掃描雖已廣泛應用，然而在實務上仍常因條碼遮擋、反光、摺痕或磨損等因素，導致掃描失敗。當此情況發生時，店員往往需改以人工輸入商品編號進行結帳，不僅增加操作時間，也容易造成輸入錯誤，影響結帳效率與顧客體驗。

近年來智慧零售與自動結帳相關研究多以大型賣場或無人商店為主要應用場景，系統往往結合多組攝影機、RFID或高成本感測設備，雖可提升辨識準確率，但在設備成本與系統複雜度上，較不適合導入於小型零售或便利型商店。因此本研究以低成本、易部署為設計目標，提出一套結合影像辨識與重量感測之自動結帳系統，針對小型零售場景進行應用設計。系統在結帳時以影像辨識作為主要識別方式，並於辨識結果不確定時，透過重量資訊進行二次驗證，以減少條碼掃描失敗所造成的不便，提升結帳流程之效率與實用性。

1.2 研究目的

本研究的主要目的是設計並實現一套結合影像辨識與重量感測的自動結帳

系統，透過深度學習模型與感測技術，達成以下目標：

(1) 商品自動辨識：

以YOLO11模型為核心，建立自有商品資料集進行訓練與測試，使系統能自動辨識多種類型的商品。

(2) 重量輔助驗證：

使用Arduino電子秤模組即時量測商品重量，作為影像辨識結果的驗證依據，以減少誤判。

(3) 整合式操作介面：

以Python開發使用者操作介面，整合影像輸入與重量資訊，讓使用者能即時查看辨識結果與金額資訊。

(4) 提升準確與便利性：

建立一套具有穩定性與可擴充性的智慧結帳流程，作為未來無人商店技術的實際應用基礎。

1.3 研究範圍

本研究以小型零售場域之商品識別與結帳為主要應用範圍。研究內容涵蓋影像辨識模型訓練、重量感測資料擷取、資料整合與系統介面開發。商品種類以常見的包裝食品、飲料為主，系統環境設計為單一攝影機與電子秤模組構成之原型平台。

本研究聚焦於商品辨識與秤重驗證的技術整合，不涉及金流支付系統、雲端資料同步與後端庫存管理等延伸功能。研究成果主要透過系統原型之功能實作與測試分析進行展示，以驗證多感測資訊輔助商品辨識於小型零售應用情境下之可行性。

1.4 應用情境

本系統的應用情境以「小型無人商店」或「自助結帳櫃檯」為主要目標。顧客進行結帳時，只需將商品放置於平台上，系統即可自動啟動攝影機拍攝影像並由YOLO11模型進行辨識，接著由Arduino電子秤模組即時回傳重量資料，兩者比對後即能確認商品種類與數量。我們開發了一套智慧零售結帳系統，其前端介面由HTML、CSS與JavaScript負責視覺化呈現和使用者互動邏輯，後端核心則以Python驅動，實現商品名稱、價格與合計金額的即時同步顯示。

第二章 文獻探討

本章將針對數個方向做相關文獻之探討：一是有關於本篇論文做為物件辨識核心的YOLO物件辨識演算法相關文獻作探討；二是就資料面探討3D建模與渲染，並納入空標註負樣本、困難樣本以抑制誤警之效果；三是回顧感測器融合相關研究，重點放在於推論完成後，利用其他感測資訊進行一致性檢查的後期融合策略。此類方法透過不同感測來源之交叉驗證，以降低僅依賴單一感測來源所可能造成的誤判風險，並作為本研究後續重量驗證機制設計之理論依據。

2.1 卷積神經網路

整個卷積神經網路(CNN)的架構主要包含卷積層(Convolution layer)、池化層(Pooling layer)、全連接層(Fully Connected layer)三部分，透過局部感受野和權重共享在影像這類網格資料上高效萃取多層次特徵。

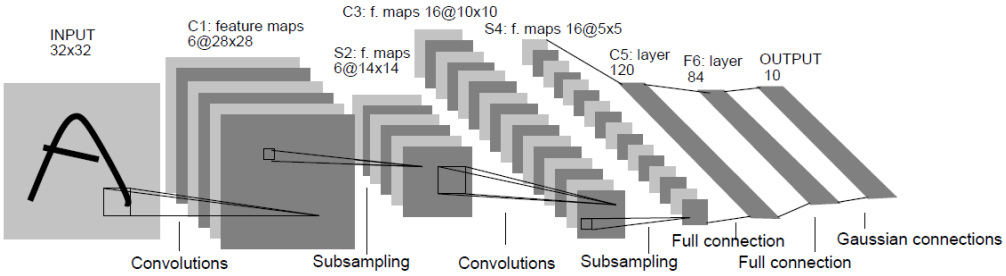


圖 2.1 Lecun 等人提出的 LeNet-5 的結構 [3]

2.1.1 卷積層：

卷積層以小型濾波器（kernel）在輸入上平移卷動，對局部區域做加權相加，產生特徵圖，能以極少參數學到低階邊緣、紋理，再透過多層堆疊形成高階語意表徵，並天然具平移等變性。此種結構在影像等網格資料上兼具參數效率與平移等變性，為現代視覺系統之關鍵基礎[3]。如圖 2.2所示，給定6×6影

像卷積核大小為3x3的卷積運算。

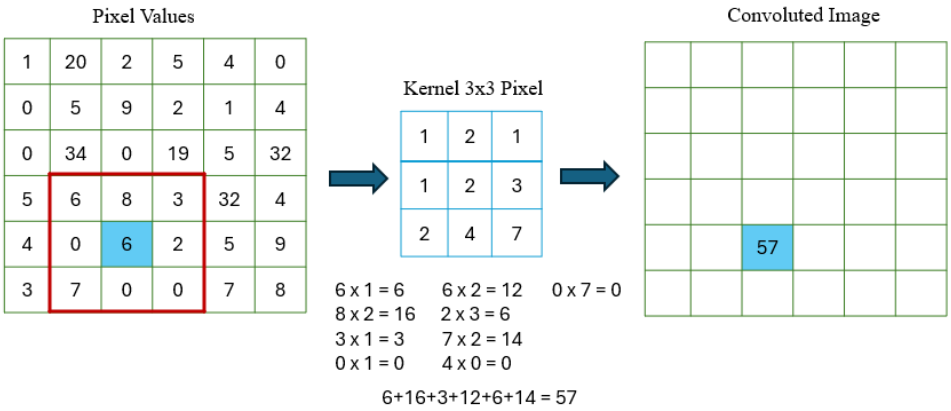


圖 2.2 卷積層示意圖

給定輸入 $X \in \mathbb{R}^{H\times W\times C_{\text{in}}}$與卷積核 $K \in \mathbb{R}^{k_h\times k_w\times C_{\text{in}}\times C_{\text{out}}}$，二維卷積之輸出特徵Y可表示為：

$$Y[i,j,C_{out}] = \sum_{u,v,Cin} X[i+u, j+v, C_{in}]K[u,v,C_{in},C_{out}] \quad (2.1)$$

i , j為輸出特徵圖Y的空間座標；u , v分別為卷積核K之列及行，用來滑過輸入的局部視窗； C_{in}為輸入通道索引，求和時會把所有輸入通道的貢獻加總； C_{out}為輸出通道索引，也就是「第幾個濾鏡或特徵圖」，每個C_{out}對應一組卷積核權重[4]。

在YOLO11中卷積層的設計對模型之特徵擷取能力、運算效率與複雜場景下的辨識穩定性具有關鍵影響，以3×3或1×1的Conv–BN–Act堆疊建立多層次表徵。這樣的卷積堆疊在多個面向同時受益：一、透過細緻的局部邊緣擷取與通道互動，整體表徵能力提升，能兼顧細節與語義；二、配合有序的下採樣與後續金字塔融合，模型對不同尺寸與姿態的目標更為穩定，形成良好的多尺度可辨識；三、計算路徑緊湊、延遲低，滿足即時性需求並利於邊緣部署。這樣的設計不僅支撐YOLO11的速度與精度，也為後續的特徵金字塔與輸出頭奠定穩固基底[5], [6]。

2.1.2 池化層：

池化層是卷積神經網路中用於空間降維的關鍵，將區域特徵聚合為更粗粒度的表示，以降低計算量、減輕過擬合，同時增強對平移與形變的穩定性。最常用的兩種作法為最大值池化層(Maximum Pooling)和平均值池化層(Average Pooling)。最大池化在特徵圖上以視窗滑動，對每個區塊僅保留數值最大的回應，特別有助於強化邊緣、輪廓與細節等高對比訊號。平均池化則是對每個區塊的回應取算術平均，屬於平滑化聚合，例如顏色分佈、紋理走向與形狀概貌。

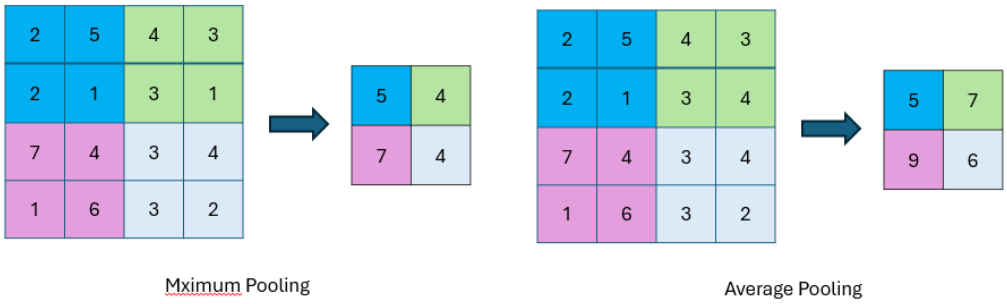


圖 2.3 最大池化和平均池化示意圖

對自助結帳場景而言，商品在鏡頭下可能出現微小位移、鏡面反光與姿態差異。Boureau等人指出最大池化能帶來良好的局部不變性，即使邊緣略有位移，區塊內的最大回應仍傾向被保留，對邊界與高對比細節更為穩健。平均池化維持整體顏色與紋理統計，但尖峰訊號會被稀釋造成細節對比下降，兩者形成穩定性和判別性之平衡[7]。

2.1.3 全連接層：

在卷積神經網路中，全連接層通常最為最終的分類器。將上一層特徵向量化後，對所有輸入與輸出節點做完全連結的仿射變換，常用來把高階特徵映射

成最終決策（如類別分數）。其優點是表達力強容易與常見激活損失結合；缺點是參數與計算成本高，資料不足時較易過擬合。對YOLO11而言，並不使用大型全連接層作為輸出頭，而是採用卷積式解耦頭（convolutional decoupled head）直接在多尺度特徵上回歸框與分類分數，以保留空間對齊並降低延遲。

2.2 YOLO11 架構介紹

Redmon等人於2016年首度提出YOLO（You Only Look Once），這是一類單階式目標偵測方法，將候選框生成、分類與回歸整合為單次前向傳遞，以即時性與低延遲為核心設計目標[8]。後續演進多採用典型的Backbone–Neck–Head架構。隨著主幹網路、錨點機制與資料擴增策略的持續優化，YOLO系列在精度與速度之間逐步取得更佳平衡；Ultralytics的官方文件亦指出，YOLOv8進一步採用錨點自由（anchor-free）與解耦頭（decoupled head），並以C2f與SPPF等模組提升效率，使訓練與推論更為靈活且高效[9]。

相較於YOLOv8，YOLO11在保持單階偵測路線的同時，在Backbone部分以C3k2取代C2f；Neck部分在SPPF後新增C2PSA，帶來更高的mAP與更低的參數與計算量。這些改動帶來更穩定的多尺度表徵與更好的速度精度折衷，對自助結帳情境更有利。

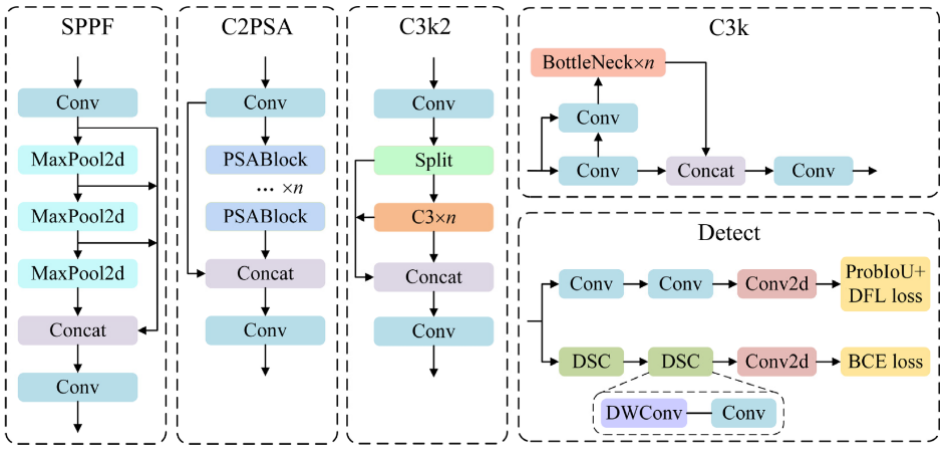


圖 2.4 Huang 等人所繪製原版 YOLO11 模型架構[10]

2.2.1 Backbone

骨幹（backbone）負責把輸入影像轉換為由淺至深的多尺度特徵圖，淺層保留邊緣與紋理，深層聚合語意與形狀。為了讓梯度更順暢、同時擴大對多尺度上下文的感知，當代骨幹常以殘差與跨層捷徑串接各層特徵，並透過金字塔彙整整合不同尺度的資訊。Lin等人就提出在金字塔上輸出P3、P4、P5等固定階層，作為後續特徵融合與檢測頭的輸入[11]。

在YOLO11的骨幹中，He等人提出的ResNet殘差機制被作為設計基礎，用以穩定深層網路的梯度傳遞並提升收斂表現。前端以Conv–BN–SiLU的前導模組串接3×3、步幅為2的卷積核作為採樣，以逐步擴張感受野與通道數[6]。為了在精度與計算量間取得平衡，Wang等人提出的CSPNet將特徵與梯度分流到兩條路徑再融合，被實作為C3k和C3k2等輕量殘差模組，以減少重複計算並強化特徵重用[12]。在骨幹尾端，He等人提出的SPP用多尺度池化彙聚上下文，以多層池化彙整多尺度資訊，從而提升尺度不變性並增強全局視野[13]。

2.2.2 Neck

在多尺度偵測中Neck的核心任務是把Backbone輸出的P3–P5特徵做跨層融合，讓細節資訊能雙向流動。Liu等人提出的PANet以自底向上路徑聚合補強傳統自頂向下金字塔，使深淺層之間的資訊往返更充分，從而同時提升小目標感知與定位精度；其關鍵操作即是在各金字塔層間反覆進行上採樣、拼接與局部卷積，以縮短資訊距離並增強特徵對齊能力[14]。在YOLO系列中Bochkovskiy等人提出的YOLOv4首先將PANet式路徑聚合完整落入YOLO架構，實證其在小物體、密集目標上的效果，為後續整個YOLO系列的頸部設計奠定先例[15]。

YOLO11的頸部延續這條路線，採用上採樣（upsample）、拼接（concat）、輕量卷積模組的往返聚合。為了在即時性與精度間取得更好的折衷，YOLO11

的頸部計算多由輕量殘差類模組(C2f變體)承載，目的和Liu等人提出的ASFF所強調的「跨層特徵對齊與自適應融合」一致，在保持多尺度一致性的同時，盡量減少多餘計算並強化有效特徵的流通[16]。以上設計讓YOLO11在相近或更低的計算量下，更能夠偵測小物體與密集場景的檢出穩定性、提升多尺度的一致性與定位精確度。

2.2.3 Head

Head負責產生最終的檢測結果。在YOLO11中Head採用解耦式、無錨點（ancho的設計，各金字塔層各自輸出一組預測，以配合不同尺寸目標的感受野與解析度需求，從而在單一模型中同時兼顧小、中、大物體的檢出穩定性與召回率。Tian等人提出的FCOS將一階偵測改為無錨點（anchor-free）並以每像素預測簡化頭部設計，降低先驗錨點帶來的配對與超參數負擔，為現代頭部設計提供了高效率的基礎路線[17]。Ge等人提出的YOLOX明確採用解耦式Head並配合更合理的樣本指派，使兩任務各自優化、減少互相牽制，實證可在同級模型下獲得更好的速度與精度的權衡[18]。2020年Li等人提出的Generalized Focal Loss（含分佈式定位DFL）讓回歸分支學習邊界框的位置分佈而非單一點估，進一步改善定位品質，並與解耦頭天然相容[19]；Zhang等人提出的一種自適應樣本指派方法ATSS(Adaptive Training Sample Selection) 透過統計式與機率式樣本指派，提升不同尺度下的學習穩定度[20]。

2.3 資料生成與增強

為克服真實資料蒐集成本高與涵蓋度不足的問題，本研究採用3D建模後渲染之合成資料生成流程，並採用幾何層面的領域隨機化(Domain Randomization)，在固定背景與光照下系統性生成不同視角的影像與對應YOLO邊界框標註。此作法的理論依據可追溯至Tobin等人提出的Domain Randomization，其核心即透過大

量隨機變化逼使模型學得與環境無關的穩定特徵，雖然原論文同時涵蓋外觀擾動，但其對幾何變動（視角/姿態）的強調與本研究相符[21]。Wong等人在2019年提出「Object-to-Model」的合成資料生成方法：以實體商品進行攝影測量（photogrammetry）獲得高擬真3D模型，批量渲染多視角影像並自動輸出像素級與邊界框標註，進而建立可控且可擴充的資料庫；文中以每類約60張實拍影像建模，渲染10萬張合成影像，在獨立真實測試集達95.8%分類準確率，並使用自動標註資料訓練一階偵測器（RetinaNet）以達到即時偵測，展示該流程在工業情境的實效與可遷移性[22]。以上幾何向DR使我們能以固定背景和光照快速生成覆蓋多視角的訓練資料，實作成本低及標註全自動的優勢，有助於提升小物體與大仰角與側視情境下的穩健度，同時維持資料製備與訓練流程的簡潔。

2.4 感測器融合之重量驗證

Hall與Llinas於其研究中系統性地探討多感測器資料融合之基本架構與設計原則，並指出單一感測器在實際應用中容易受到雜訊、不確定性與環境變化之影響，進而導致判斷誤差。該研究提出透過多個異質感測器所取得之資訊進行融合與交叉驗證，使系統能夠利用不同感測來源在資訊層面的互補特性，提升整體決策之可靠度與穩定性。研究結果顯示當多種感測資訊被合理整合後，系統對於異常狀況與非預期錯誤的辨識能力可明顯優於僅依賴單一感測來源之情況[23]。

Khaleghi等人針對多感測資料融合相關研究進行全面性的回顧與整理，分析不同感測器在資訊層級、特徵層級與決策層級之融合方式，並指出多感測融合特別適用於需要進行一致性檢查與錯誤抑制之應用情境。該研究強調當系統能夠利用不同感測器所提供之獨立資訊來源進行相互驗證時，可有效降低因單一感測器誤判所造成的風險，並提升系統於不確定環境下之穩定性[24]。

基於上述研究成果，本研究將多感測融合之概念應用於自助結帳系統中，於影像辨識完成後，針對可能存在外觀相似或不同容量規格之商品情境，進一步啟

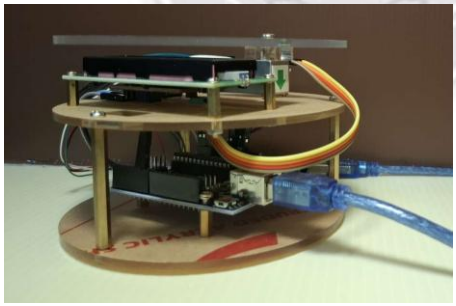
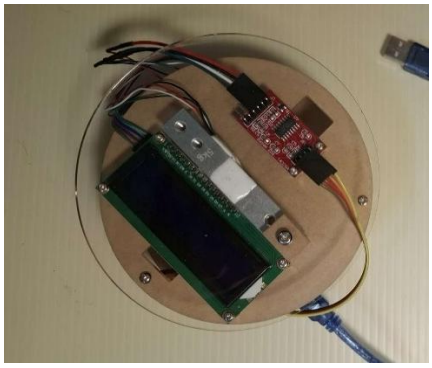
動電子秤模組進行重量量測，並將實測重量與 POS 資料庫中所對應之理論重量進行比對。透過影像辨識與重量感測之交叉驗證，可作為二次確認機制，以降低誤判與非預期錯誤的發生，並強化自助結帳流程中商品判斷之一致性與系統整體穩定性。

第三章 研究方法

3.1 實驗設備

本研究使用之硬體主要包括攝影機模組、電子秤感測模組、平板電腦以及後端執行的主控電腦等，各部分功能如下表所示。此型號之攝像頭無法調整焦距，因此我們將其固定在適當高度，並在底板處劃設界線，讓顧客能夠清楚知道鏡頭拍攝的區域。

表 3.1 實驗設備清單

設備	外觀圖	型號及功能
攝像頭		型號：羅技 c270i 解析度：720p/30 fps 功能：拍攝商品
電子秤感測模組	 	組成結構：Arduino Uno 開發板、HX711、重力感測元(load cell) 功能：當攝影機辨識有相同商品不同大小，需要再次重量驗證

平板電腦		型號：samsung tab s9 fe 功能：人機互動介面，顯示結帳金額、數量、掃描 Qrcode 印出 pdf 收據
主控電腦		型號：msi modern 14 B11M 功能：整體系統的運算中樞，負責影像辨識、跨模態資料整合與 POS 交易流程
瓦楞板		功能：結帳系統之硬體架構
水管		功能：架高攝影機作為支撐

3.2 實驗軟體

YOLO11程式：

本研究在Google Colab環境中使用YOLO11，主要採用yolo11n.pt 權重。先以pip指令安裝Ultralytics 8.3.30版。完成後在程式中載入yolo11n.pt時，Ultralytics會自動自官方模型庫下載YOLO11的預訓練權重並加以快取。鑑於雲端執行節點具暫時性與可能重置之特性，為確保實驗成果之持久保存與版本管理，本研究將Ultralytics輸出目錄預先指向Google Drive，並以時間戳命名每次實驗之彙整資料

夾，以便後續比對、重現與追蹤。

Polycam：

三維建模採用Polycam作為掃描與資料前處理軟體。Polycam支援以行動裝置與多視角影像進行重建，能快速產出含貼圖之網格模型，並提供遮罩與背景去除、網格簡化，以利後續資料集製作。

Blender：

完成商品3d建模後便能將Polycam產出的Gltf模型匯入到Blender進行圖像渲染，隨後在內建腳本進行領域隨機化，包含相機距離與方位、俯仰與滾轉角等，以模擬顧客以各種角度進行掃描。透過Python腳本擷取2D投影框座標，自動產生YOLO格式標註，並以固定亂數種子確保可重現，快速製作大量圖片資料集。

Roboflow：

本研究除合成影像外，亦納入少量真實商品影像與空標註影像。上述影像皆於Roboflow平台進行線上人工標註，並建立統一之類別與命名規範，完成後透過資料版本化功能進行train/val/test比例切分與歷程管理。為擴增資料多樣性，Roboflow之內建增強管線如水平翻轉、隨機縮放與裁切等被系統性套用，以在不改變標註語義的前提下倍增影像數量並提升對光照、反光與部分遮擋之穩定性。最終資料集以YOLO格式匯出並對應至絕對路徑，確保後續訓練與驗證流程之可重現性與可追溯性。

重量感測模組：

使用Arduino以C語言撰寫韌體程式，定期讀取HX711感測值並轉換為實際重量（公克），再透過USB串列埠傳送至Python。Python程式端利用pyserial函式庫接收並解析資料，使重量資訊能與影像辨識結果同步顯示。

3.3 實驗運作流程

本系統之運作流程如圖 3.1所示。顧客先將商品拿到攝影機下方，由

YOLO11進行即時影像辨識；僅當系統判定該品項在店內存在「同款不同容量或規格」的情形時，才啟動重量模組進行輔助比對，以區分同品項的不同大小。完成辨識後，系統將結果送入POS購物車並即時累計金額；結帳時輸出交易明細與總金額，並以QRcode方式產生PDF格式收據，模擬完整的自助結帳流程。

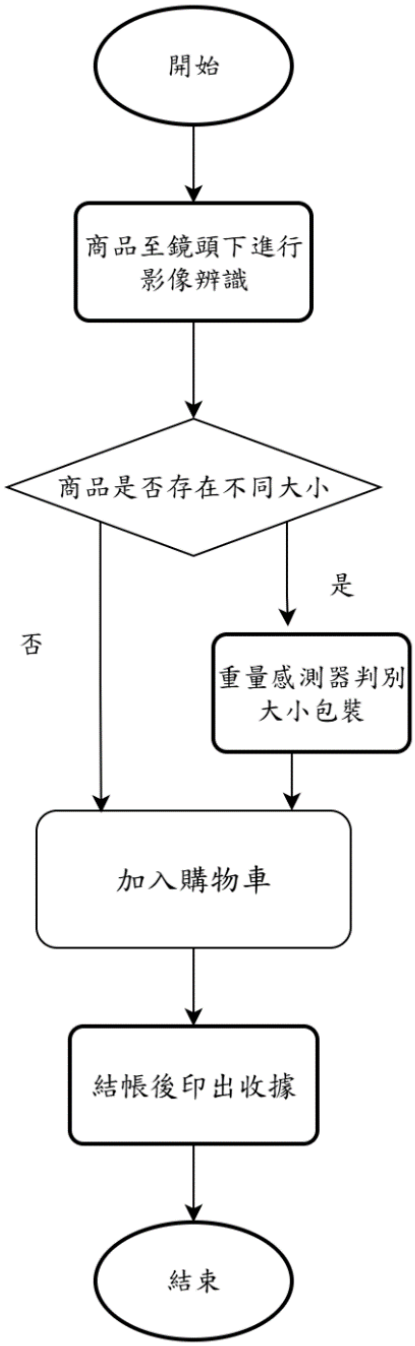


圖 3.1 流程圖

YOLO11資料庫建立如圖 3.2所示，採用合成影像為主、實拍補強之方式以商品三維模型為核心素材，透過Blender進行視角參數隨機化批次渲染，以產製高多樣性的影像，並以腳本自動生成對應之邊界框標註。納入少量實拍影像以縮減域差，並於Roboflow平台完成人工校正與一致性檢查，將兩路資料統一管理與版本化後匯出為YOLO格式。完成之資料集再於Google Colab環境以Ultralytics YOLOv11進行訓練與驗證，產出最佳化權重檔（best.pt），作為後續系統推論與評估之基礎。

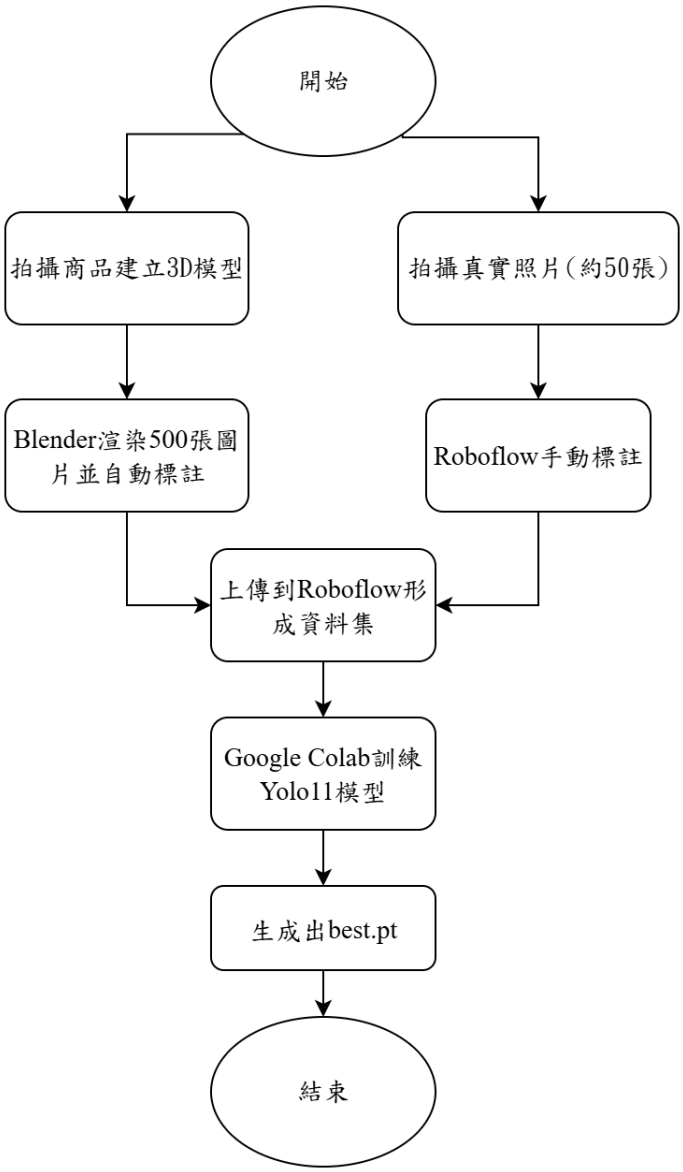


圖 3.2 資料庫建立流程圖

3.3.1 三維模型建立

我們採用 Polycam 的自動拍照模式蒐集物件影像。將物品放置於空曠平面，再以 360°環繞的方式使用自動拍照功能拍攝約 80 張照片。拍攝時與物體保持約 0.5 到 1 m 的固定距離，以平行視角繞行完成第一圈；隨後再以略高與略低視角各補一圈，以提升上緣與下緣的覆蓋率。若物件上表面或底面較複雜，另行由上方與低角度各補拍半圈，以確保重建幾何與貼圖的完整度，如圖 3.3 所示。AUTO 會在影像穩定且重疊度足夠時自動觸發快門，拍攝速度太快或間斷拍攝會導致破洞與接縫錯位，造成成品出現破損，如圖 3.4 所示，因間斷姓拍攝導致產品正反面融合。拍攝環境以均勻散射光為宜，背景選用具紋理的材質以利特徵追蹤，並盡量避免強反光、透明件或鏡面干擾。完成拍攝後，可在內部進行修圖去除地面或支撐物，再匯出 GLTF 檔。

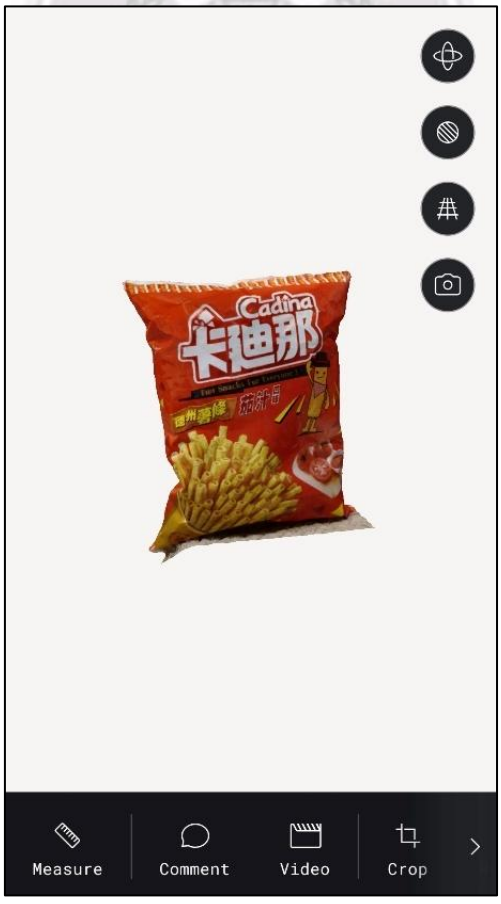


圖 3.3 完整三維模型圖

圖 3.4 間斷拍攝使產品兩面融合

3.3.2 生成合成影像

為快速擴充可訓練之影像資料，本研究以Blender建立參數化的批次渲染流程。匯入上述Polycam輸出的GLTF檔後，先指定目標模型、相機與輸出根目錄，並設定類別編號與輸出張數，本次研究以每樣商品500張作為資料集，如圖 3.5、圖 3.6所示。輸出解析度設為640×640，與YOLO常用的輸入尺寸對齊。研究場景採固定外觀條件以便單獨分析幾何因素，世界背景設定為中性白，曝光0.0，配置一盞定向光作為主光源；相機焦距固定為35 mm。整體設計不進行光照或材質的隨機化，僅在幾何層面進行變化，包括相機繞行視角、目標微傾與尺度抖動，以避免外觀變動干擾資料分布，並聚焦評估視角、姿態與尺度對偵測效果的影響。每張影像渲染後同步輸出YOLO格式標註，使影像與標註一一對應並可直接投入訓練。

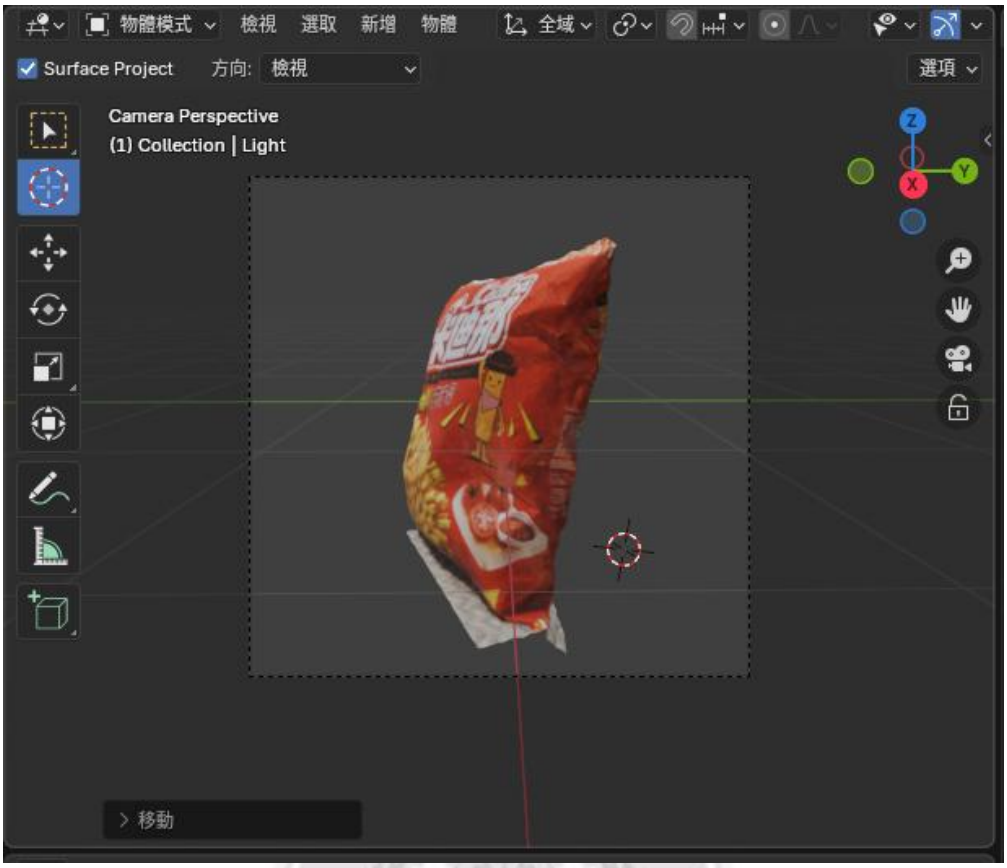


圖 3.5 Blender 相機視角下之商品 3D 模型

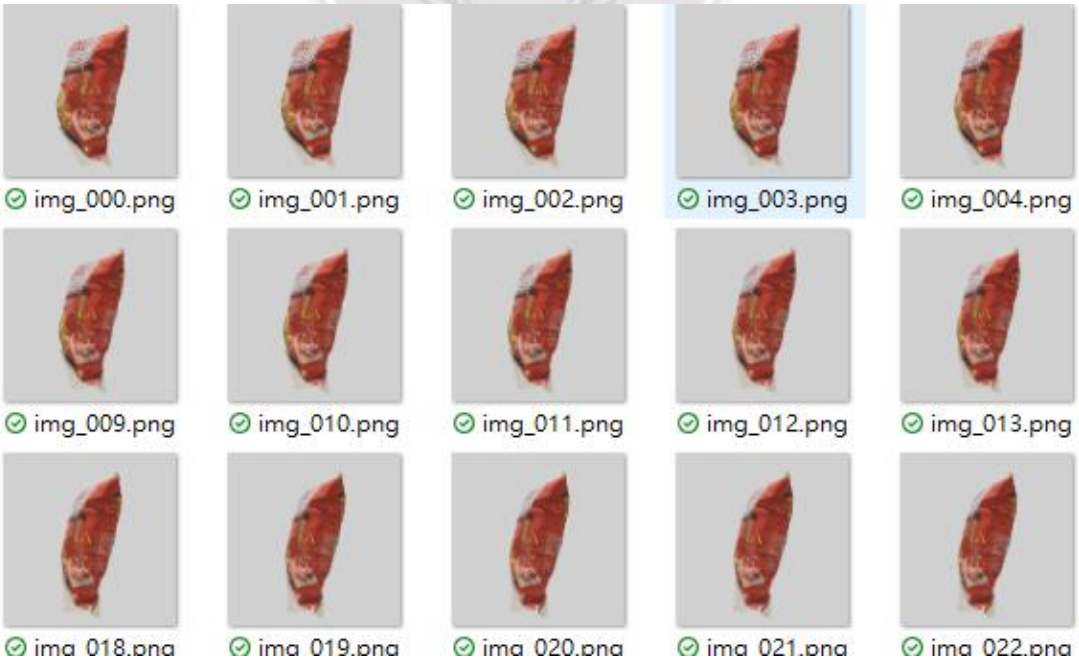


圖 3.6 批次渲染輸出影像範例

3.3.3 真實資料補強與手部負樣本

在本研究的資料管理與標註流程中，所有實拍補強影像與手部負樣本皆以Roboflow建立與維護。首先將Blender產生的合成影像與標註以及不同光源與背景下拍攝的真實照片一併上傳至同一專案，統一類別定義與標註格式。若為非資料集的商品，建立空標註做為負樣本，如圖 3.7所示；若手部分遮擋了商品，只框出可見的商品部分，不涵蓋被手遮住的區域，如圖 3.8所示；若僅有手而無商品，維持空標註，如圖 3.9所示。為避免分布漂移，我們僅啟用與部署情境一致的輕量增強，具體包含水平翻轉、縮放、旋轉±15°、飽和度-20%~20%、亮度-15%~15%、曝光-10%~10%與模糊上限2畫素；上述增強僅套用在訓練集，驗證與測試維持原樣。資料採8：1：1隨機分配方式區分訓練集、驗證集、測試集，並固定隨機種子以確保可重現。Roboflow的版本管理記錄每次新增的真實影像與負樣本比例，使我們能比較不同配方對誤檢率與小物體準確率的影響。最終以 YOLO格式匯出資料集並同步類別對照表，訓練腳本可直接讀取。

圖 3.7 非目標商品之負樣本影像

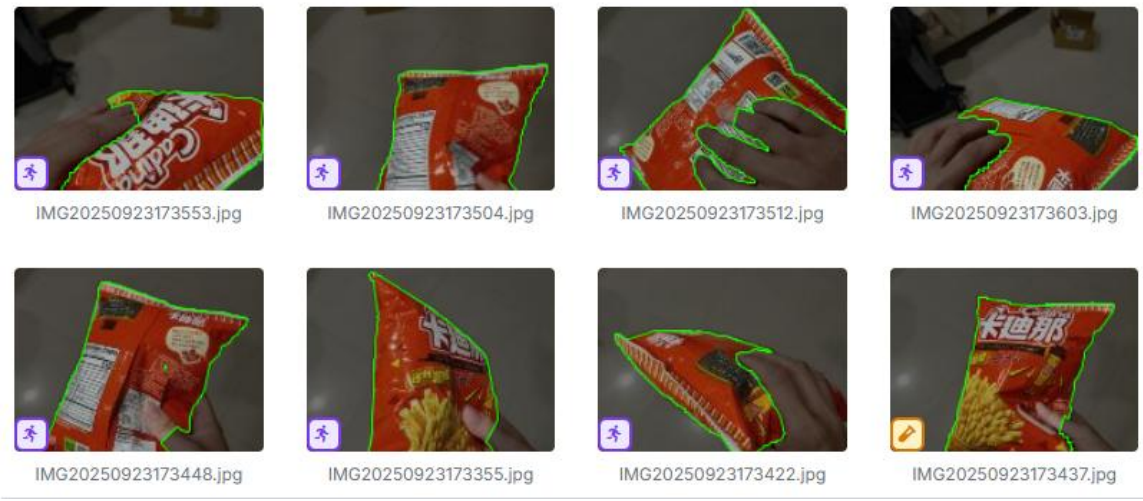


圖 3.8 部分遮擋情境之可見區域標註

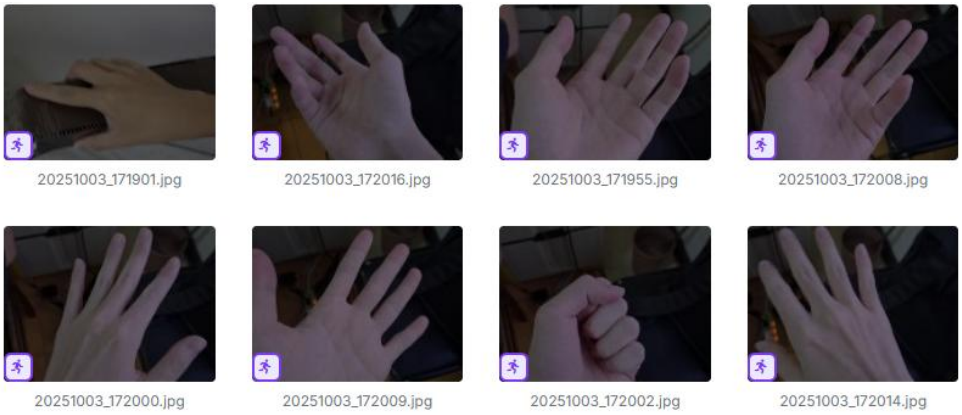


圖 3.9 手部空標註負樣本

3.3.4 模型訓練

本研究採用YOLO11進行即時影像辨識，適合在一般GPU上達成穩定的即時推論。訓練以官方yolo11n預訓練權重初始化，輸入影像640x640，批次大小(Batch)十六，80個訓練週期(epoch)，採用餘弦學習率排程並啟用自動混合精度與早停機制，早停耐心值二十。訓練過程自動產出學習曲線、精確率與召回率曲線與混淆矩陣，並保存於雲端目錄，另輸出最佳權重與最後權重以供比較。模型性能以mAP@0.5、Precision、Recall三項指標評估。訓練完成後，透過Python Ultralytics函式庫將YOLO11佈署到即時攝影機影像串流中進行推論，確保在

實務場域具備穩定的即時辨識能力。

3.3.5 模型訓練評估指標

(1) mAP@0.5

mAP@0.5指在交疊門檻IoU＝0.5下的「平均平均精度」。做法是：先對每一類別畫出精確率召回率曲線，計算其面積得到該類的Average Precision(AP)；IoU＝0.5表示預測框與真值框的交並比至少一半才算正確。接著把所有類別的AP取平均，就得到mAP@0.5。這個指標偏重「框到就算對」的情境，對框的位置要求較寬鬆。若要更嚴格的定位評估，通常會同時回報mAP@0.5:0.95。

(2) 精確度(Precision)

表示模型判為正確的預測中，實際正確的比例。在物件偵測中，它代表所有被框出的目標裡，有多少同時類別正確；精確率越高，誤檢越少。其計算公式如下：

$$\text{Precision} = \frac{\text{TP}}{\text{TP+FP}} \tag{3.1}$$

其中 TP 為真陽性，模型正確偵測到的目標數；FP 為假陽性，模型框出但其實不正確的數量。

(3) 召回率(recall)

表示在所有真實存在的目標中，模型成功找出的比例。召回越高代表漏檢越少，但通常需要和精確率一起權衡，因為降低置信度門檻雖可提升召回，卻可能增加誤檢。

$$\text{Recall} = \frac{\text{TP}}{\text{TP + FN}} \tag{3.2}$$

其中 TP 為真陽性，模型正確偵測到的目標數；FN（假陰性）：實際存在但被模型漏掉的目標數。

(4) F1 曲線(F1 Curve)

F1是精確率與召回率的調和平均，曲線形狀反映「少誤檢」與「少漏檢」的綜合權衡。當曲線越高、峰值越大，表示在當前設定下同時維持較好的精確率與召回率。P為精確率，R為召回率，計算公式如下：

$$F1 = \frac{2PR}{P+R} \quad (3.3)$$

(5) 混淆矩陣(Defusion Matrix)

混淆矩陣是用來比較真實類別與模型預測類別的表格，行通常表示真實標籤，列表示模型預測，對角線上的數量代表分類正確，非對角線的數量代表被混淆到其他類別，如圖 3.10所示。觀察矩陣可看出哪些類別最容易互相誤判、類別是否不平衡、以及每一類的精確率與召回率是否偏低；對角線越深或越大表示整體越準確，常見的做法是同時查看標準化版本以比較不同類別在相對比例上的表現，並據此調整資料增強、類別定義或決策閾值。混淆矩陣中的四個區塊定義如下：

1. True Positive（TP）：實際為正類，且模型亦正確預測為正類之樣本數。
2. True Negative（TN）：實際為負類，且模型亦正確預測為負類之樣本數。
3. False Positive（FP）：實際為負類，但模型誤判為正類之樣本數。
4. False Negative（FN）：實際為正類，但模型誤判為負類之樣本數。

透過上述四項指標，可進一步計算多種模型效能評估指標，以全面評估分類模型在不同錯誤類型下之表現。混淆矩陣不僅能反映整體分類正確率，亦能協助分析模型在正負樣本不平衡或特定錯誤類型下之行為特性。

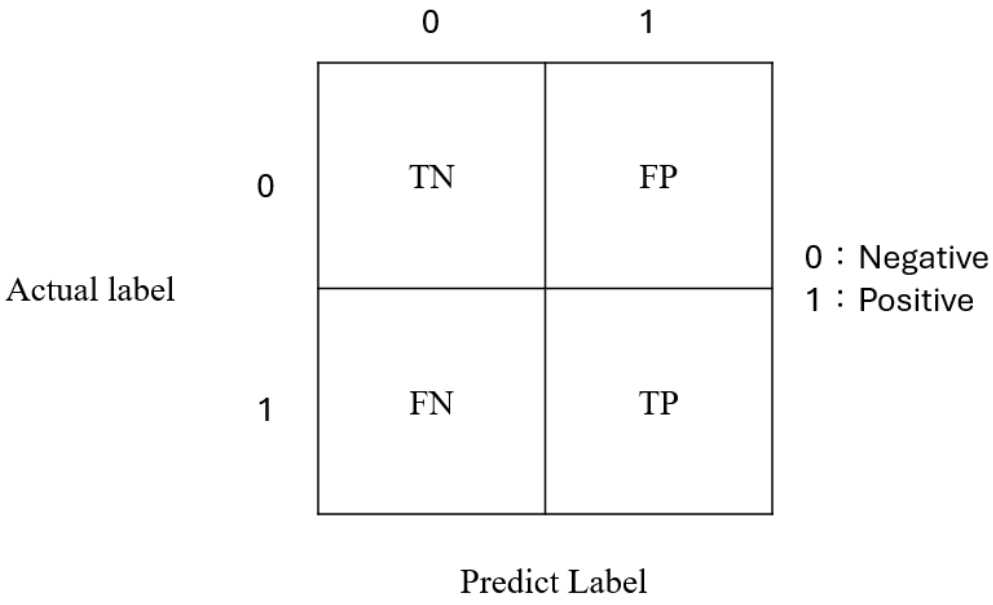


圖 3.10 混淆矩陣示意圖

第四章 實驗結果與討論

4.1 實驗架構

本研究之實驗架構如圖 4.1所示，系統主要由影像辨識模組、重量量測模組和POS機台三部分組成。影像辨識部分在結帳區域上方架設Logitech C270i攝影機，固定面向「影像辨識區域」，前端互動平板負責顯示操作介面與辨識結果，後端主控制電腦則接收攝影機影像並執行YOLO模型進行商品偵測與分類。重量量測部分由內建Load Cell之秤重平台、HX711重量感測模組與Arduino Uno開發板構成，當影像辨識判定為同一品項但可能為不同重量規格時，平板會跳出提示，要求顧客將商品放置於秤重平台進行重量感測，量測到的重量資料經由Arduino傳送至後端電腦，與資料庫中的標準重量區間比對，確認後加入購物車；若商品只有單一重量規格時則直接加入購物車中。在平板畫面會顯示商品清單與總金額，顧客按下「確認結帳」後即可完成交易，系統畫面會顯示QRcode連結提供連上商店WIFI的顧客下載收據。

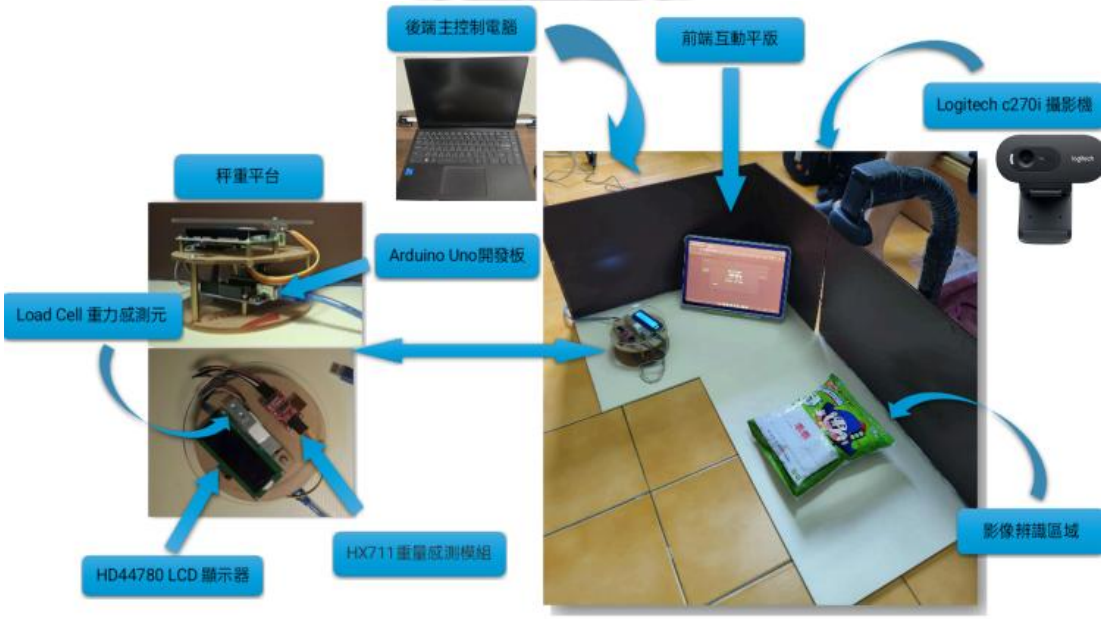


圖 4.1 實驗架構圖

4.2 模型訓練

4.2.1 訓練曲線

圖 4.2顯示yolo11n.pt訓練過程中所繪製出的曲線，共包含10條訓練與驗證評估曲線，本次訓練的最大週期數為80（epochs=80）。模型亦輸出F1-score曲線與正規化混淆矩陣，作為評估各項商品類別訓練成效與分類表現的依據。以下為各項指標之說明：

box loss可用來評估模型在邊界框迴歸上的表現，數值越低，代表模型繪製出的邊界框越接近標註結果。由圖 4.2可見，train/box_loss隨著訓練週期數穩定下降，並在後期逐漸趨於平緩；val/box_loss雖然在前期略有波動，但整體趨勢同樣持續下降且與訓練曲線相當接近，顯示模型在邊界框迴歸部分已達到良好收斂。

cls loss用來評估模型在判斷物件類別上的準確度。YOLO11會根據每一個預測框輸出各類別的機率分佈，cls loss則量測這個機率分佈與真實標籤之間的差異，因此數值越低代表模型越能正確區分各種商品類別。圖 4.2可觀察到，train/cls_loss 自訓練初期便快速下降，約在中後期收斂至較低且穩定的水平；val/cls_loss 雖然在前幾個epoch有明顯震盪，但隨著訓練進行同樣呈現整體下降趨勢，最終與訓練曲線相當接近。結果顯示模型在學習各商品類別特徵時能有效降低分類誤差並達到收斂。

dfl loss用來進一步優化框的位置預測精度。YOLO11透過dfl將每個邊界位置視為一個機率分佈，藉由學習正確分佈形狀，使模型能在像素層級上更精準地估計框的邊界位置，dfl loss越低，表示模型對物件邊界的細部預測越準確。train/dfl_loss隨訓練週期數穩定下降，呈現平滑且單調遞減的趨勢；val/dfl_loss雖在前期有較明顯的波動，但整體仍持續下降，並在中後期逐漸趨於穩定，與

訓練曲線的變化方向一致。顯示最終輸出的框不僅大致位置正確，其邊界也具有相當程度的精確度。

在整體效能指標部分，右側四條曲線分別為 Precision、Recall、mAP50與mAP50-95，用以評估模型在物件偵測上的整體表現。Precision代表在模型判定為有物件的預測中有多少比例是正確偵測，Recall則反映真實有物件時模型成功偵測出的比例；兩者皆在訓練前期迅速上升，於中後期趨於穩定，其中Precision維持在接近1、Recall約在0.9左右，顯示誤判與漏檢情形皆相對較少。mAP50與 mAP50-95則分別對應於IoU=0.5與多個IoU門檻下的平均準確率，曲線同樣呈現先快速提升、後期進入平臺期的收斂型態，整體而言顯示本研究訓練之YOLO11 模型在偵測正確性與邊界框定位精度上皆具有穩定且可靠的表現。

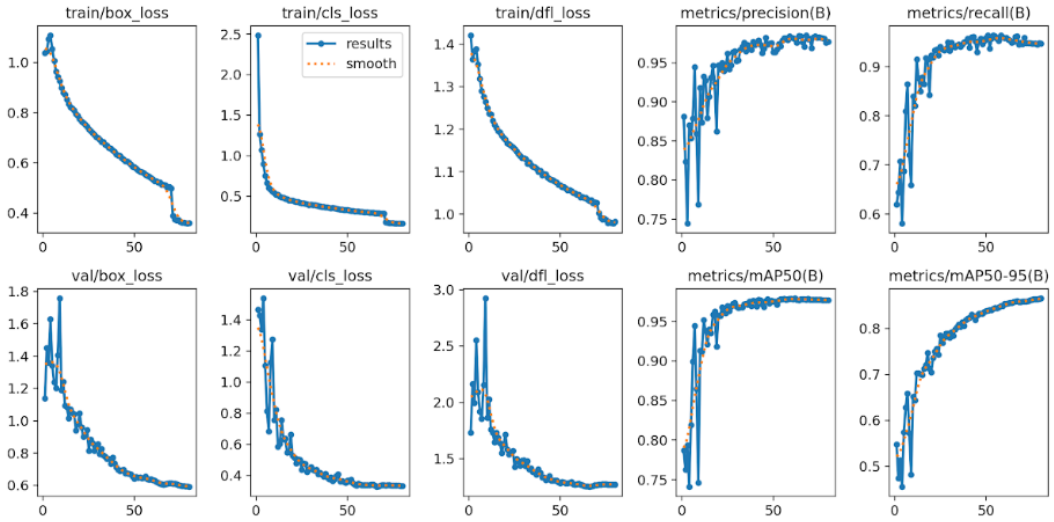


圖 4.2 yolo11n 訓練曲線

4.2.2 F1 評估指標

錯誤! 找不到參照來源。為本實驗模型之F1–Confidence曲線圖，橫軸為偵測時所設定的信心閾值（Confidence），縱軸為對應的F1值，用以同時評估Precision與Recall的整體表現。圖中各細線代表不同商品類別的F1曲線，粗藍線

則為全類別的平均結果，可見在信心閾值約0.1～0.8的區間內，多數類別之F1值皆維持在0.9以上，顯示模型在相當寬的門檻範圍內仍能兼顧誤判率與漏檢率，具備穩定的偵測效能。其中整體最佳表現在Confidence約0.67時達到平均F1約0.96，故本研究亦選擇此值作為後續推論的主要信心門檻；僅少數類別（如Real Leaf）之F1曲線略低，顯示該類別相對較難辨識，未來可透過增加實際影像或調整資料增強策略加以改善。

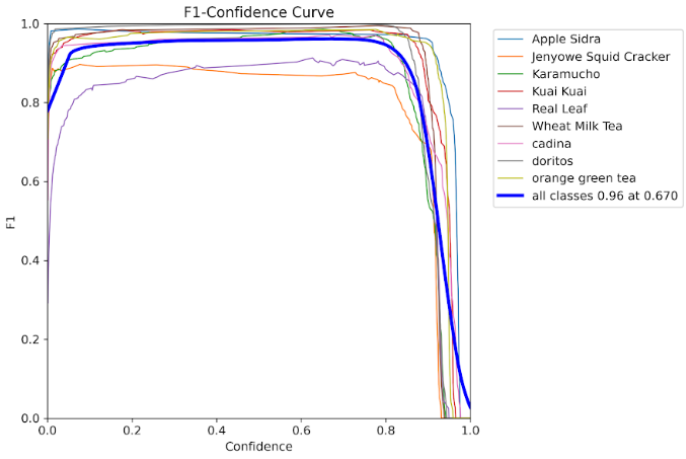


圖 4.3 F1 曲線

4.2.3 混淆矩陣

本研究之正規化混淆矩陣如圖 4.4所示，橫軸為真實類別，縱軸為模型預測類別，格子中的數值代表在某一真實類別下，被預測為各類別的比例，因此每一列加總約為1。整體而言各商品類別在對角線上的數值皆偏高，正確分類比例多落在0.95左右，僅Jenyowe Squid Cracker約0.82，顯示模型在大部分商品上具有良好辨識能力且商品彼此間的混淆較少。較顯著的誤判主要集中於商品被分類為 background的情形，其中Real Leaf被判為background的比例約0.41，說明在部分拍攝情境下仍存在漏檢問題，未來可透過增加該類別樣本數或調整資料增強策略來進一步改善。

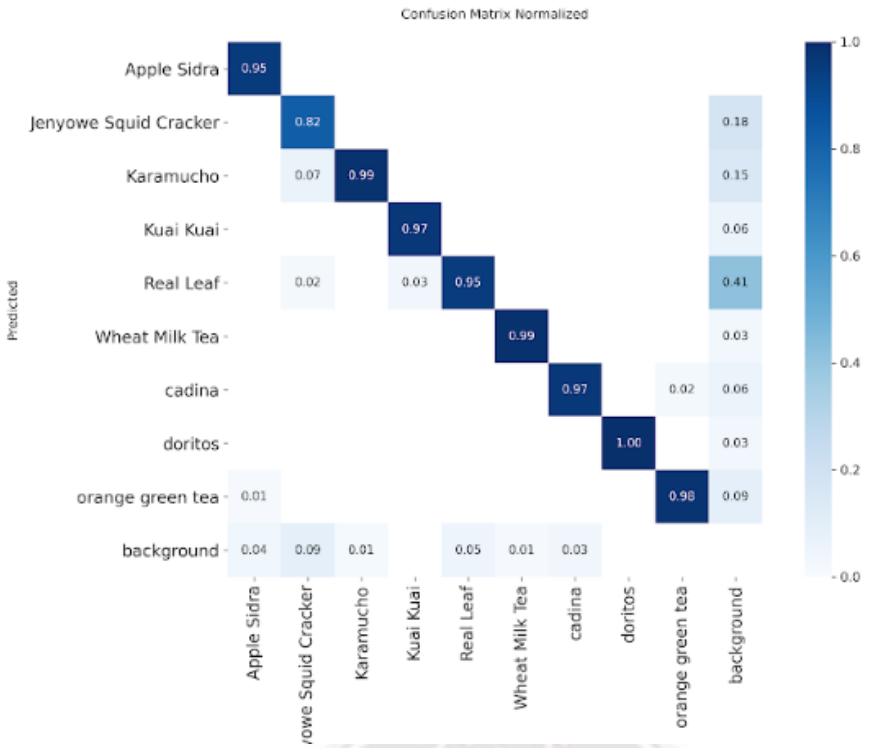


圖 4.4 正規化混淆矩陣

綜上所述，本研究訓練之YOLO11模型在本研究商品資料集上展現出良好的辨識效能。由正規化混淆矩陣可觀察到，各類別在對角線上的預測比例多介於0.95～1.00之間，僅在少數類別與background之間出現零星誤判，顯示模型對多數商品皆能準確區分。F1-Confidence曲線顯示，各類別在寬廣的信心區間內皆維持高F1值，整體最佳F1約為0.96、對應信心閾值約0.67，代表在此閾值下可取得良好的偵測平衡。從訓練與驗證曲線來看，模型整體收斂情形良好、指標表現趨於穩定，未見明顯過擬合現象。整體而言，該模型已具備在本系統場景中進行自動商品辨識與結帳應用之可行性。

4.3 系統整合

為了有效整合軟體邏輯與硬體訊號，系統實作上依據功能職責與資料流向，劃分為影像辨識與重量偵測端和操作介面二部分：。

4.3.1 影像辨識與重量偵測端

本系統的第一部分作為資料輸入端，主要由執行於運算單元上的核心偵測程式以及負責物理測量的重量感測模組共同組成。此部分的設計目標在於建立一個穩定且具備初步決策能力的感知系統，能夠將現實世界中非結構化的影像與重量訊號，轉換為後端可處理的結構化數位指令。本研究在實作多幀穩定鎖存機制與重量訊號穩定判決流程時，部分程式碼與數學式推導初稿透過生成式AI工具輔助產生，最終內容均由作者檢視、修正並確認其正確性。

4.3.1.1 影像辨識與多幀穩定演算法

在影像辨識方面，程式啟動時首先載入預先訓練好的YOLO11物件偵測模型(best.pt)，並透過OpenCV介面從攝影機擷取即時影像流，設定信心度門檻 (Confidence Threshold) 與重疊度門檻(IoU Threshold)進行初步過濾。然而，考量到實際零售場景中，手部晃動、環境光影變化或商品快速移動常導致辨識框產生閃爍或短暫消失，若直接將每一幀的辨識結果送出，極易導致購物車重複加入商品或誤判。為解決此問題，本研究在程式內部實作了一套「多幀穩定鎖存演算法 (Multi-frame Stability Latch Algorithm)」。該演算法透過維護一個固定長度的先進先出 (FIFO) 佇列，持續收集最近N幀影像的辨識結果並進行投票統計。僅當特定商品在視窗內的出現頻率超過設定的投票門檻，且其信心度佔比維持在穩定水準時，演算法才會判定該目標為有效並進入「鎖定狀態」。處於鎖定狀態時，系統會暫時忽略後續短

暫的訊號遺失，直到商品連續消失超過設定的冷卻幀數後才解除鎖定，此機制類似於電子電路中的去抖動處理，大幅提升了辨識的穩定性。本研究將該演算法之數學模型定義如下：設Ft為時間點t的影像幀，經由 YOLO 模型推論後得到的物件標籤為Lt（若無結果則為0）。系統維護一個長度為W的滑動視窗(Sliding Window)，其歷史狀態集合表示為$Ht = \{L_t, L_{t-1}, \dots, L_{t-W+1}\}$。針對視窗內的任意標籤$l$，定義其出現次數 $Count(l)$與有效樣本總數$Total$：

$$Count(l) = \sum_{i=0}^{W-1} 1(H_{t-i} = l), Total = \sum_{i=0}^{W-1} 1(H_{t-i} \neq 0) \quad (4.1)$$

系統判定是否鎖定目標l^*的條件為：

$$l^* = argmax\ Count(l) \text{ 且} (Count(l^*) \geq V_{min}) \wedge (\frac{Count(l^*)}{Total} \geq \theta_{ratio}) \quad (4.2)$$

其中V_{min}為最小票數門檻，θ_{ratio} 為信心比例門檻。此數學模型確保了系統在面對不穩定的影像輸入時，仍能輸出離散且可靠的觸發訊號。

4.3.1.2 邏輯分流與重量穩定演算法

除了穩定的視覺辨識，本程式亦具備邏輯分流的能力，實現了視覺與觸覺訊號的整合。當演算法鎖定一個有效商品標籤後，程式會即時查詢內部的商品設定檔 並依據商品屬性執行不同的處理流程。若辨識出的商品僅有單一規格，程式會執行API請求函式，將商品名稱與價格封裝為JSON格式，直接發送至後端伺服器接口，完成自動加入購物車的動作。反之若辨識出具備大小包變體的商品，程式則切換邏輯，判定該商品需要物理重量輔助驗證。此時，程式不執行加入購物車的動作，改為請求重量辨識，暫停視覺端的送單功能並啟動電子秤讀取流程，直到重量驗證完成。本系統在後端驅動層實作了「重量訊號穩定與判決演算法」。其數學模型定義如下：設ω_t為時間點t從感測器讀取的原始重量值。系統維護一個大小為K的採樣緩衝區$B_t = \{\omega_t, \omega_{t-1}, \omega_{t-K+1}\}$。為了判定當前重量是否已達穩定狀態，我們

定義兩項檢測指標：

1. 峰值變異檢測：

檢查緩衝區內最大值與最小值的差是否在容許誤差 ε_{tol} 內。

$$\triangle_{pp} = \max(B_t) - \min(B_t) \leq \varepsilon_{tol} \quad (4.3)$$

2. 有效負載檢測：

檢查平均重量 ω_t是否超過最小有效重量W_{min}，以過濾空秤時的零點漂移。

$$\omega_t = \frac{1}{K}\sum_{i=0}^{K-1}\omega_{t-i} \text{ ，} |\omega_t| \geq W_{min} \quad (4.4)$$

當上述條件連續滿足後，系統鎖定當前平均重量ω_{final}並與設定檔中的分界值S_{split}進行比對，執行最終的規格判決：

$$Variant = \begin{cases} Large\ size, & if\ \omega_{final} \geq S_{split} \\ Small\ size, & if\ \omega_{final} < S_{split} \end{cases} \quad (4.5)$$

這兩種協作模式解決了單純依賴視覺難以精確區分外觀相似但重量不同的商品。

4.3.2 操作介面

本研究所提出之自動結帳系統包含影像偵測介面、重量驗證介面、商品結帳頁面以及電子收據產出模組，整體介面以直覺、易操作為主要設計目標，使一般使用者能於無需專業知識的情況下完成商品辨識與自助結帳流程。本節將依照實際操作流程展示各介面功能。

4.3.2.1 影像偵測介面

此介面僅在測試過程中觀察模型的辨識狀態，並不會顯示於一般消費者的操作介面中。系統會持續接收攝影機影像並即時執行物件偵測，畫面上包含偵測框、辨識類別與信心度等資訊，以及左上角的 FPS（Frames Per

Second)，用以觀察系統處理效能，如圖 4.5 後端邊界框最終鎖定比例圖 4.5 所示。

圖 4.5 後端邊界框最終鎖定比例

系統設定基礎置信度閾值作為YOLO輸出端的初步過濾標準；凡置信度低於0.70之邊界框皆會在模型端被直接忽略，以避免低可信度的偵測結果進入後續流程。

在後處理階段，系統進一步要求單一幀之偵測結果必須至少達到0.65，方能被視為有效偵測並納入後續多幀穩定判斷流程。凡置信度低於0.65之邊界框將被視為該幀「未偵測到目標」，不會參與時序累積運算，也不會影響商品鎖定結果。此二次過濾設計可有效減少瞬間偵測噪聲或邊緣偵測框造成的誤判，並提升整體判斷於多幀序列中的一致性與穩定性，使最終商品辨識結果更加可信且符合實際零售環境之需求。實驗的過程中有部分商品較易辨識錯誤，本研究亦針對特定類別設定更嚴格的最低置信度要求(置信度0.88~0.90），只有當偵測結果超過該類別之專屬閾值時，方視為有效偵測。

最後為確保偵測結果在時間序列上的穩定性，本研究使用多幀穩定演算法，要求某商品在滑動視窗內(W＝10)之有效偵測比例需達到85%(0.85)

以上，才會最終被鎖定。透過此由單幀至多幀的分層式置信度過濾架構，系統得以兼具即時性與辨識穩定性，降低單幀誤判對整體結帳流程所造成的影響。

4.3.2.2 重量驗證介面

重量驗證介面是本系統中處理需秤重商品時的一個交互窗口，如圖4.6所示。系統的核心任務是將感測到的重量數據精確地劃分為預設的商品規格範圍，例如小包裝或大包裝。這個劃分過程根據商品重量配置表來確定判決依據。我們採用一個單一的分界值來決定商品的最終規格，而非固定區間。最終有效重量將直接與預先設定的分界值進行比較，從而判斷商品落在哪個規格範圍：

1.小包裝範圍：

如果最終鎖定的重量小於這個分界值，系統判決商品屬於小包裝。

2.大包裝範圍：

如果最終鎖定的重量等於或超過這個分界值，則系統判決商品屬於大包裝。

透過這種方式，系統便能自動完成商品的規格判斷，並將正確的變體名稱發送到購物車，解決了僅依賴影像辨識難以區分大小包裝的挑戰。

圖 4.6 重量驗證介面

4.3.2.3 商品結帳頁面及電子收據

結帳頁面是整個自動化交易流程的最終步驟，負責向用戶確認最終交易總額、處理結帳請求，並提供電子收據，如圖 4.7所示。此介面的數據完全依賴於後端狀態管理模組實時廣播的購物車資訊，確保了前端與後端狀態的高度一致性。

圖 4.7 POS 機結帳頁面

進入結帳頁面後，使用者可再次確認購物車內之商品內容與金額，確認無誤後按下「確認結帳」按鈕，此操作會觸發後端伺服器進入交易的最終處理階段，並同步生成電子收據。系統會依據購物車中所記錄的商品名稱、單價與總金額，透過文件建立一份標準格式之PDF收據，並於生成過程中嵌入中文字型，以確保中文品名與金額資訊能正確顯示。

當收據檔案生成完成後，後端伺服器會透過即時通訊機制通知前端介面，並提供該電子收據之存取連結。使用者介面接收到通知後，會立即於平板畫面上顯示對應之QR Code，如圖 4.8所示。使用者只需連接至店家所提供的WIFI，並使用手機掃描該QR Code，即可直接下載或預覽本次交易的電子收據，無需額外輸入任何交易資訊，如圖 4.9所示。

交易完成後，系統將啟動延遲清空機制，60秒後自動重置購物車資料並解除交易狀態標誌，使整體系統能在無需人工干預的情況下迅速恢復至待命狀態，準備服務下一位顧客。

圖 4.8 收據之 QR Code

圖 4.9 結帳電子收據

第五章 結論與未來展望

5.1 結論

綜合本研究的實作與測試結果，在小型零售且商品品項數相對有限的情境下，即使採用較低成本的硬體配置，仍能透過影像辨識、重量驗證與狀態管理的整合，完成一套可運作的自動結帳原型。相較於僅以單一模型的辨識率作為核心目標，本研究更強調系統層級的可用性，包含辨識結果如何被接收、如何在不確定狀態下避免錯誤觸發交易、以及如何將最終交易資訊以一致的方式呈現給使用者。從整體流程來看，本研究證實影像辨識與重量二次確認能在特定場域中互補其限制，影像辨識負責提供快速且直覺的品項判定，而重量驗證則在外觀一致的情況下提供額外依據，使交易流程更接近實務上所要求的可靠度。雖然目前仍受限於資料規模，且尚未建立以交易為單位的完整量化指標，如整體結帳正確率、誤收率與漏收率，因此對不同情境下的整體表現仍難以做出全面推論，但本研究已建立一個具備清楚架構與可擴充性的系統原型，並提供後續進行大規模驗證、效能優化與功能擴充的良好基礎。

5.2 未來展望

未來可朝提升商品多樣性與多商品同時辨識能力進行擴充。在商品多樣性方面，可逐步增加不同類別、包裝與規格之商品，使系統能因應實際零售環境中商品頻繁更新的需求。在多商品辨識方面系統可進一步優化對同一畫面中多個不同商品的偵測與判斷流程，以支援一次放置多樣商品的結帳情境。

本研究尚有數項可持續改善的方向。影像層面上可針對複雜場景進行強化，例如改善系統在商品遮擋、堆疊或反光情況下的辨識穩定性，以提升其在非理想拍攝條件下的可靠度。重量驗證模組方面，可導入更完善的量測策略，如多次取樣平均或自動校正機制，以降低感測誤差對判斷結果的影響。在系統流程設計上，

可進一步優化前後端狀態同步與錯誤處理機制，使系統在辨識失敗或感測異常時能提供更明確的回饋，避免影響使用者操作體驗。在實際應用層面，可進行更長時間與更大規模的場域測試，評估系統於長時間連續運行下的穩定性，並作為未來商用化與功能擴充的重要參考依據。

參考文獻

[1] 工業技術研究院（ITRI）,“創新科技「拿了就走」：工研院攜手 7-ELEVEN 打造 24 小時獨立智慧商店,” 工研院. [Online]. Available: https://www.itri.org.tw/ListStyle.aspx?DisplayStyle=01_content&MGID=11207191259577978 4&MmmID=1036276263153520257&SiteID=1. Accessed: Nov. 13, 2025.

[2] 新住民全球新聞網,“缺工解方！韓國智慧超商成趨勢，顛覆傳統購物模式,” 新住民全球新聞網. [Online]. Available: https://news.immigration.gov.tw/NewsSection/Detail/f6fcb32a-760f-404a-a449-410ddbe1810e?category=0&lang=TW. Accessed: Nov. 13, 2025.

[3] Y. Lecun, L. Bottou, Y. Bengio, and P. Haffner, “Gradient-based learning applied to document recognition,” *Proc. IEEE*, vol. 86, no. 11, pp. 2278–2324, Nov. 1998, doi: 10.1109/5.726791.

[4] V. Dumoulin and F. Visin, “A guide to convolution arithmetic for deep learning,” Jan. 12, 2018, *arXiv*: arXiv:1603.07285. doi: 10.48550/arXiv.1603.07285.

[5] T.-Y. Lin, P. Dollár, R. Girshick, K. He, B. Hariharan, and S. Belongie, “Feature Pyramid Networks for Object Detection,” in *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, July 2017, pp. 936–944. doi: 10.1109/CVPR.2017.106.

[6] K. He, X. Zhang, S. Ren, and J. Sun, “Deep Residual Learning for Image Recognition,” in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016, pp. 770–778. doi: 10.1109/CVPR.2016.90.

[7] Y.-L. Boureau, J. Ponce, and Y. LeCun, “A Theoretical Analysis of Feature Pooling in Visual Recognition”.

[8] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, “You Only Look Once: Unified, Real-Time Object Detection,” in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016, pp. 779–788. doi: 10.1109/CVPR.2016.91.

[9] Ultralytics, “YOLOv8.” Accessed: Nov. 13, 2025. [Online]. Available: https://docs.ultralytics.com/zh/models/yolov8

[10] J. Huang, K. Wang, Y. Hou, and J. Wang, “LW-YOLO11: A Lightweight Arbitrary-Oriented Ship Detection Method Based on Improved YOLO11,” *Sensors*, vol. 25, no. 1, p. 65, Jan. 2025, doi: 10.3390/s25010065.

[11] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollár, “Focal Loss for Dense Object Detection,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 42, no. 2, pp. 318–327, Feb. 2020, doi: 10.1109/TPAMI.2018.2858826.

[12] C.-Y. Wang, H.-Y. M. Liao, I.-H. Yeh, Y.-H. Wu, P.-Y. Chen, and J.-W. Hsieh, "CSPNet: A New Backbone that can Enhance Learning Capability of CNN," Nov. 28, 2019, *arXiv*: arXiv:1911.11929. doi: 10.48550/arXiv.1911.11929.

[13] K. He, X. Zhang, S. Ren, and J. Sun, "Spatial Pyramid Pooling in Deep Convolutional Networks for Visual Recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 37, no. 9, pp. 1904–1916, Sept. 2015, doi: 10.1109/TPAMI.2015.2389824.

[14] S. Liu, L. Qi, H. Qin, J. Shi, and J. Jia, "Path Aggregation Network for Instance Segmentation," in *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, June 2018, pp. 8759–8768. doi: 10.1109/CVPR.2018.00913.

[15] A. Bochkovskiy, C.-Y. Wang, and H.-Y. M. Liao, "YOLOv4: Optimal Speed and Accuracy of Object Detection," Apr. 24, 2020, *arXiv*: arXiv:2004.10934. doi: 10.48550/arXiv.2004.10934.

[16] S. Liu, D. Huang, and Y. Wang, "Learning Spatial Fusion for Single-Shot Object Detection," Nov. 26, 2019, *arXiv*: arXiv:1911.09516. doi: 10.48550/arXiv.1911.09516.

[17] Z. Tian, C. Shen, H. Chen, and T. He, "FCOS: Fully Convolutional One-Stage Object Detection," in *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, Oct. 2019, pp. 9626–9635. doi: 10.1109/ICCV.2019.00972.

[18] Z. Ge, S. Liu, F. Wang, Z. Li, and J. Sun, "YOLOX: Exceeding YOLO Series in 2021," Aug. 09, 2021, *arXiv*: arXiv:2107.08430. doi: 10.48550/arXiv.2107.08430.

[19] X. Li *et al.*, "Generalized Focal Loss: Learning Qualified and Distributed Bounding Boxes for Dense Object Detection," June 09, 2020, *arXiv*: arXiv:2006.04388. doi: 10.48550/arXiv.2006.04388.

[20] S. Zhang, C. Chi, Y. Yao, Z. Lei, and S. Z. Li, "Bridging the Gap Between Anchor-Based and Anchor-Free Detection via Adaptive Training Sample Selection," in *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020, pp. 9756–9765. doi: 10.1109/CVPR42600.2020.00978.

[21] J. Tobin, R. Fong, A. Ray, J. Schneider, W. Zaremba, and P. Abbeel, "Domain Randomization for Transferring Deep Neural Networks from Simulation to the Real World," Mar. 22, 2017, *arXiv*: arXiv:1703.06907. doi: 10.48550/arXiv.1703.06907.

[22] M. Z. Wong, K. Kunii, M. Baylis, W. H. Ong, P. Kroupa, and S. Koller, "Synthetic dataset generation for object-to-model deep learning in industrial applications," Sept. 25, 2019, *arXiv*: arXiv:1909.10976. doi: 10.48550/arXiv.1909.10976.

[23] D. L. Hall and J. Llinas, "An introduction to multisensor data fusion," *Proc. IEEE*, vol. 85, no. 1, pp. 6–23, Jan. 1997, doi: 10.1109/5.554205.

[24] B. Khaleghi, A. Khamis, F. O. Karray, and S. N. Razavi, "Multisensor data fusion:

A review of the state-of-the-art," *Inf. Fusion*, vol. 14, no. 1, pp. 28–44, Jan. 2013, doi: 10.1016/j.inffus.2011.08.001.