

IoT 物聯網導論 智慧倉儲管理系統

Smart Warehousing Management System: Integration of ToF Sensors and Computer Vision for Enhanced Inventory Control and Space Optimization

Ching-Kai, Cho
Mechanical Engineering
National Taiwan University
Taipei, Taiwan
r11522659@ntu.edu.tw

Ju-Ting Bai
Mechanical Engineering
National Taiwan University
Taipei, Taiwan
r12522603@g.ntu.edu.tw

Shu-Han Liao
Electrical Engineering
Tamkang University
New Taipei City 251, Taiwan
shliao@gms.tku.edu.tw

JIAN,ZONG-YU
Electrical Engineering
Socle Technology
New Taipei City 251, Taiwan
finalchien@gmail.com

Abstract—本研究提出一個具成本效益的智慧倉儲管理系統，用以解決傳統倉儲在管理上的低效率問題。本系統結合 ToF 感測器與電腦視覺技術，實現庫存自動化監控、空間優化及即時數據回報。架構採用 Raspberry Pi 和 teensy4.0 實現數據收集與控制，並透過 camera 與 ToF 感測器獲得貨物大小與倉儲空間，系統能大幅降低人力需求，提高可靠性，並具備未來擴展性。實驗結果顯示，該系統有效提升作業效率達 50%，減少人為錯誤，並透過 Unity 直觀呈現倉儲狀態。

Keywords—智慧倉儲、ToF 感測器、電腦視覺、庫存管理、Unity 可視化

I. 簡介

在現代倉儲管理中，隨著商品流通量的增加，傳統的倉儲方式逐漸暴露出其效率低下和管理困難的問題。傳統倉儲管理通常依賴人力進行庫存記錄、物品擺放監控以及庫存狀態的即時更新，這種模式容易導致錯誤發生，且效率低下，無法滿足現代供應鏈對快速響應和高效率運作的需求。

A. 傳統倉儲問題

傳統倉儲在多個方面仍然存在局限性，特別是在空間利用、數據管理與即時性等方面。倉儲空間的管理往往缺乏系統化規劃，貨物的擺放方式缺乏規劃，導致空間利用不足，許多倉儲角落成為閒置區域，無法充分發揮其效能。此外，由於管理依賴人工監測，不僅需要投入大量人力與時間，還可能因疲勞或疏忽造成數據記錄錯誤，對整體運營效率構成威脅。而較常見的問題是傳統倉儲系統無法提供即時的庫存狀態更新，使管理者難以及時掌握庫存資訊，影響補貨時進行的決策的準確性與執行效率。

B. 系統目標

為解決上述問題，本研究旨在設計一套基於 ToF 感測器與電腦視覺技術的智慧倉儲管理系統，結合虛擬倉儲模擬環境，達到以下目標：

- 庫存即時監控與記錄：實現貨物尺寸、編號及數量的自動化記錄與更新。

- 空間利用優化：利用感測器精準測量倉儲空間，優化物品擺放方式，提升空間使用率。
- 即時警報與提示：倉儲容量不足時，自動警告使用者，避免貨物放置錯誤或掉落風險。
- 數據可視化：透過 Unity 模擬倉儲內部狀態，提供直觀的操作界面。

C. 系統目標用戶與效益與其目標用戶

本系統的目標用戶涵蓋了倉儲管理的多個角色，並為每個角色帶來不同層面的效益。首先，對倉儲管理者而言，智慧倉儲系統能提供即時的庫存狀態更新，讓管理者能迅速做出補貨與空間調整的決策，有效提升倉儲運營效率。同時，系統也能幫助服務業或銷售環境中的員工快速回應客戶查詢，通過即時準確的庫存信息，提升顧客服務的質量與滿意度。對於負責補貨的工作人員，系統則能即時提供倉儲空間的資訊，避免重複運輸貨物或因空間不足造成延誤，從而優化補貨路徑與流程。不僅如此，終端用戶或購物平台上的顧客也能受益於智慧倉儲，透過系統的實時庫存資訊，顧客可隨時獲取準確的商品可用量，大幅提升購物體驗。

D. 系統創新點

與傳統倉儲管理系統相比，本系統具備以下創新點：

- 高成本效益：硬體配置簡單、價格低廉（如 ToF 感測器及 Logitech 攝影機）。
- 自動化與無人化：僅需最低限度的設置，即可自動進行貨物監測與記錄。
- 高擴展性：可結合重量量測、自動化產線以及未來的 AI 技術，應用於更多場景。
- 數據直觀呈現：利用 Unity 建構虛擬倉儲模型，提供實時數據視覺化。

II. 系統設計

本智慧倉儲管理系統的設計以滿足業師提出的功能需求為核心，結合感測技術與電腦視覺實現自動化倉儲管理，並提供即時數據記錄與警報功能。以下針對需求分析、技術方案以及系統架構進行詳細描述。

A. 業師需求

根據業師的需求，本系統需解決當前倉儲管理中存在的多重挑戰。首先，傳統倉儲管理需要投入大量人力來監測與記錄貨物信息，包括尺寸、編號以及數量，這種依賴人工操作的模式既費時又容易出錯。此外，由於無法即時掌握倉儲的狀態，管理者常難以及時安排補貨，影響運營效率。另一個問題是倉儲空間未被充分利用。傳統的擺放方式缺乏規劃，導致部分空間閒置，降低了整體倉儲容量的利用率。

為了解決這些問題，系統設計需達成多項目標，包括自動監測存儲區域並記錄貨物的相關數據（例如長、寬、高及編號）、實現庫存追蹤與補貨提示，以及結合虛擬環境模擬技術，為管理者提供實時的可視化監控。

B. 提出解方與技術細節

技術組成與解決方案(請見圖一)：

- a. 貨物三維尺寸量測
 - ToF 感測器：量測貨物高度（z 軸）。
 - 攝影機：量測貨物長寬（x、y 軸），並進行 QR Code 解碼獲取編號。
 - 數據處理與轉換：透過影像處理算法（OpenCV），結合 ToF 數據完成貨物三維測量。
- b. 倉儲深度監控與補貨提示
 - 使用 ToF 感測器監測倉儲空間深度，判斷是否能放置新貨物或需補貨。
 - 若倉儲已滿，系統會自動觸發警告提示。
- c. 虛擬環境模擬
 - Unity 環境模擬：利用倉儲空間的三維模型，實時展示貨物擺放情況，直觀顯示空間利用率。
 - 提供全景切換功能，讓使用者能清楚觀察倉儲內部狀態。

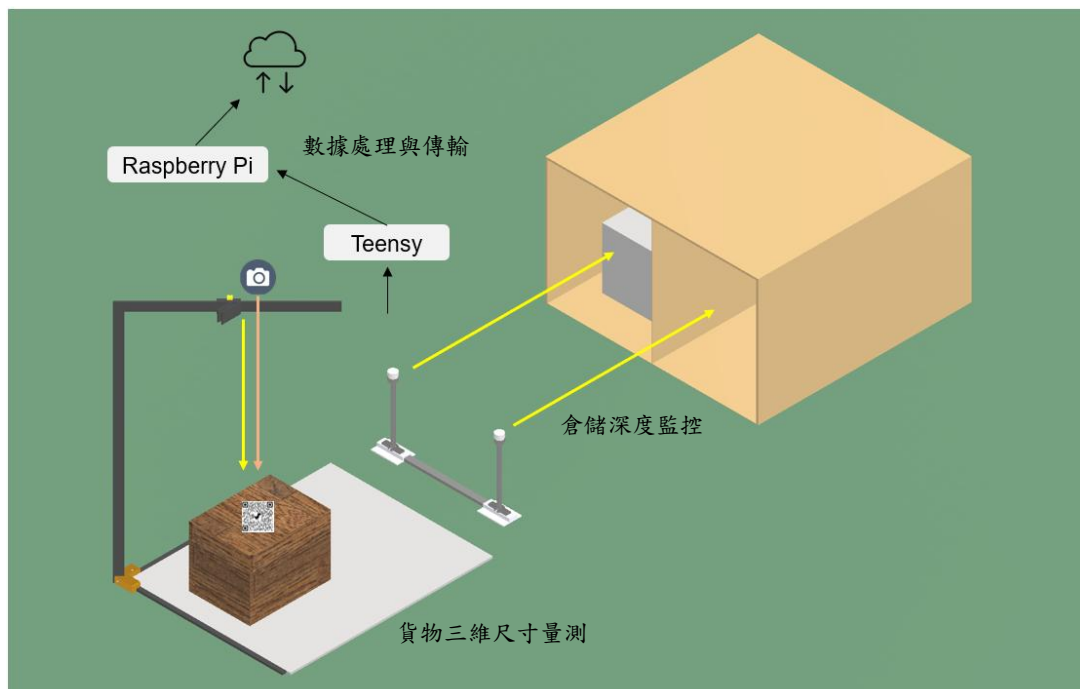
C. 系統硬體架構

為實現上述功能，系統由以下硬體組成：

- a. ToF 感測器
 - 功能：量測物體高度及倉儲深度。
 - 特點：體積小，精度高，可單顆管理約 2.5 公尺深的倉儲空間。
- b. 攝影機（Logitech Webcam）
 - 功能：拍攝貨物影像，用於獲取貨物長寬及 QR Code 編號。
 - 安裝位置：距地面 60 cm 的支架上，搭配白色底板以降低影像雜訊。
- c. 數據處理與傳輸設備
 - Teensy 4.0：負責 ToF 數據處理，並透過 I2C 將數據傳輸至 Raspberry Pi。
 - Raspberry Pi 4：整合感測器與攝影機數據，進行影像處理並將結果上傳至雲端。
 - Firebase：雲端數據庫，用於儲存庫存記錄。
- d. 可視化與用戶端設備
 - Unity：構建虛擬倉儲模型，模擬貨物擺放與空間狀態。
 - Streamlit：基於 Python 的網頁前端框架，用於展示當前庫存數據。

D. 系統軟體設計與數據流程

- a. 數據處理流程：
 - 貨物尺寸與編號獲取：
 - (1) 利用攝影機拍攝貨物影像，經二值化處理、去雜訊及輪廓檢測後，計算長寬。
 - (2) 利用 ToF 感測器量測貨物高度，並結合攝影機數據轉換實際尺寸。
 - (3) 使用 OpenCV 的 QRCodeDetector 函數獲取 QR Code 內容，記錄貨物編號。



(圖一、硬體設計概念圖)

- 倉儲狀態監控：
 - (1) 計算倉儲空間剩餘深度，判斷是否能容納新貨物。
 - (2) 若空間不足，發出「倉儲已滿」警告提示。
- 數據存儲與展示：
 - (1) 數據通過 MQTT 協議傳至 Firebase，進行即時同步與存儲。
 - (2) 在 Streamlit 網頁界面上展示庫存清單，並在 Unity 中可視化呈現。

b. 虛擬環境模擬：

- 使用 Unity 生成貨物模型，模擬其在倉儲內的實時擺放狀態。
- 提供鏡頭切換功能，觀察不同角度的倉儲空間。

E. 解方優勢

本系統在經濟性、自動化程度、擴展性與操作便利性上展現出多項優勢。首先，系統硬體配置成本低廉，使用的 ToF 感測器與 Logitech Webcam 非常適合中小型倉儲應用。系統設計充分考慮自動化需求，完成設置後即可全自動運行，大幅減少人力需求。其模組化結構也提供了高度擴展性，未來可結合重量測量、自動化產線整合等功能，進一步提升掃描可獲得的資料。同時，透過 Unity 構建的虛擬倉儲模擬環境，提供清晰直觀的視覺化管理界面，讓管理者能輕鬆掌握倉儲內部狀況並迅速作出決策。

III. 實作

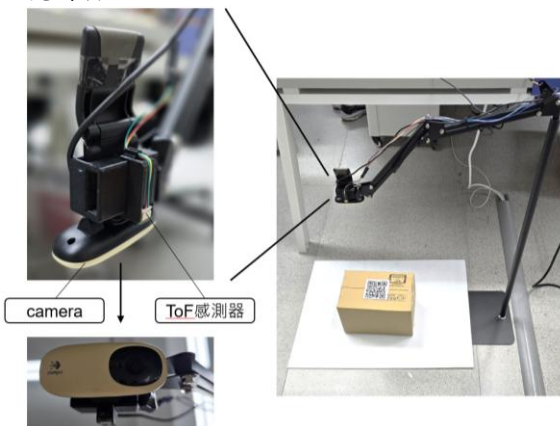
本智慧倉儲管理系統的實作基於感測器與影像處理技術，結合虛擬倉儲模擬環境，旨在實現貨物尺寸量測、編號辨識、倉儲空間監控及數據可視化展示。本節將詳細說明系統的硬體配置、數據處理流程以及實際的功能實現過程。

A. 系統功能模組與實現方法(請見圖二)

a. 貨物三維尺寸與編號量測

(1) 裝置設置：

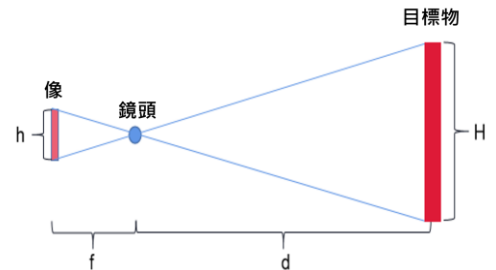
因為需要利用 ToF 感測器量測貨物的高度，並利用相機捕捉貨物的長與寬，因此使用支架將相機與 ToF 感測器架在距離地面約 60 cm 的距離，並在地面上擺放一個白色的板子，以避免相機在做影像處理時會有過多的雜訊，進而影響讀取貨物的長寬時發生錯誤。



(圖二、實際硬體展示)

(2) 數據處理流程：

- 透過相機成像原理 $h(\text{pixels}) = \frac{f}{d} \times H(\text{pixels})$ [1] 以及比例轉換 $\alpha H = \text{實際大小}$ ，藉此取得物體實際大小：經由相機焦距 (f)、物體影像大小 (h) 以及 ToF 感測器量測目標物的距離 (d)，可以得知就目標物的大小 (H)，最後再透過 $\alpha (\text{mm/pixels})$ 轉換，得出目標物的實際大小(請見圖三)。

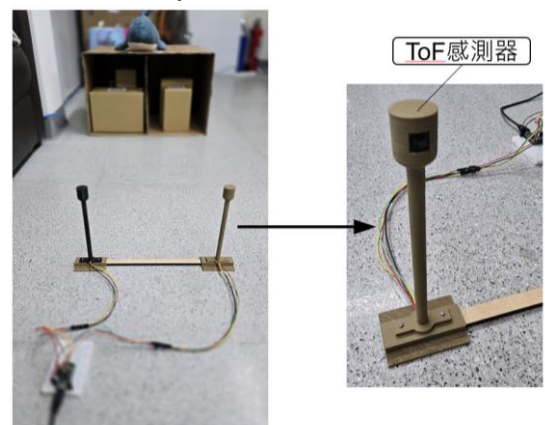


(圖三)

- 取得貨物的長與寬：將原始影像經過二值化 [2] 後，透過侵蝕與擴張的方式將影像的雜訊去除，再透過尋找最大矩形 [3] 的方式取得貨物的輪廓，最後再將透過輪廓取得的矩形四個頂點計算出 X、Y 軸的像數，然後再以第一個步驟來取得貨物實際的長與寬。
- 取得貨物的高：透過 ToF 感測器預先量測與地面之間的距離為 59 cm，因此在固定高度不變的情況下，可以將貨物進來時量測到的距離與 59 cm 相減，即可獲得貨物的高度。
- 取得貨物的編號：透過 OpenCV 提供的函式 [4]，可以獲取 QR-Code 的內容，藉此取得貨物編號。

B. 庫存紀錄與警報之解方

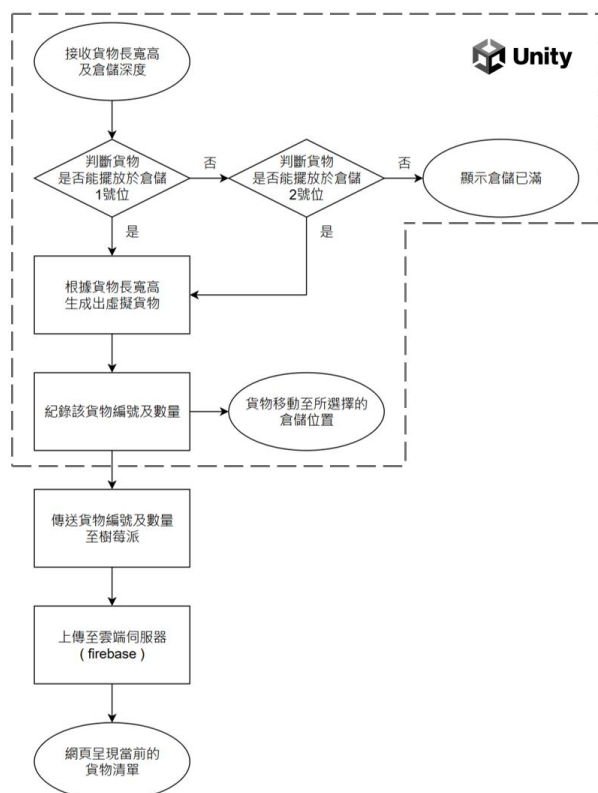
- 在倉儲前方架設 ToF 感測器(請見圖四)，並預先量測倉儲深度，再量測倉儲底部到 ToF 感測器的距離，計算出倉儲塞滿時的深度資訊，若 ToF 感測器量測到倉儲深度大於塞滿時的深度，則會在 Unity 的介面發出警告。



(圖四)、倉儲深度量測裝置

- 在流程的部分主要如圖五所示。系統首先接收貨物的基本尺寸資訊以及倉儲區域的深度數據，

作為後續判斷的依據。接著，系統會進行分析，確認貨物是否能放入指定的倉儲位置。如果倉儲空間不足，則會提示倉儲已滿的訊息。當判斷完成後，系統會將貨物的編號和數量記錄並上傳至雲端伺服器（Firebase）進行儲存，確保數據的安全性與可追溯性。最後，當前的貨物清單將以視覺化的方式呈現在網頁介面(請見圖六)上，供使用者查看。



(圖五、運作流程圖)

	Package NO.	Number
0	NO.28	1
1	NO.30	1
2	NO.29	1
3	NO.27	1
4	NO.26	2

(圖六、貨物清單)

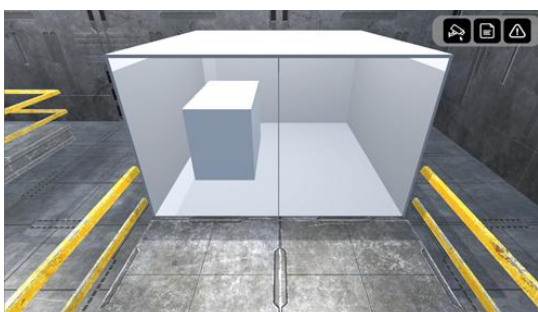
- c. 在 Unity 虛擬倉儲模擬的部分，當貨物能成功放入時，系統會在 Unity 中的虛擬倉儲環境中生成對應的貨物模型，直觀地呈現貨物在倉儲中的實際位置與擺放狀態(請見圖七)。如果倉儲容量不足，畫面會彈出「Warehouse is full !!!」的警告訊息，提醒使用者倉儲已滿，無法繼續放置貨物(請見圖八)。此外，使用者還可以透過切換鏡頭的功能，更加仔細地觀察倉儲內部的狀態，確保對倉儲情況的全面掌握與管理(請見圖九)。



(圖七、倉儲與貨物介面)



(圖八、倉儲已滿狀態)



(圖九、倉儲內部狀態視角)

IV. 成果與討論

本系統成功整合 ToF 感測器、攝影機和虛擬倉儲模擬環境，實現智慧倉儲的核心功能，包括自動化尺寸量測、即時庫存管理及空間利用優化。以下從系統架構、性能分析和應用場景進行詳細探討。

A. 系統架構分析

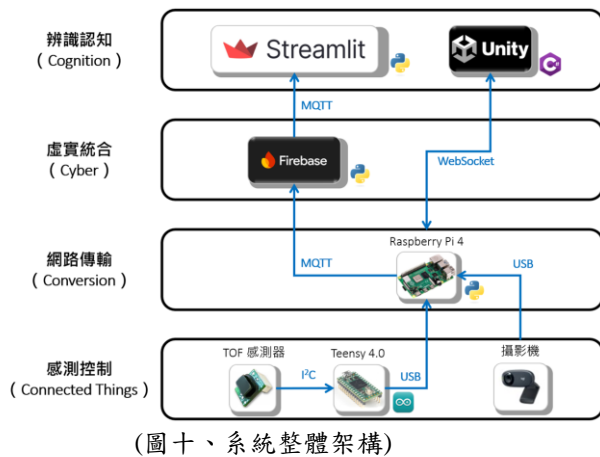
本系統採用層級化設計，將整體架構分為感測控制層、網路傳輸層、虛實整合層以及辨識與展示層(請見圖十)，以實現數據的精確收集、快速傳輸、高效儲存與直觀展示。透過層級化設計，各模組分工明確，數據在各層間順暢流動，並結合虛擬與實體的倉儲管理技術，為使用者提供即時、準確且直觀的貨物管理功能。

a. 感測控制層（Connected Things）：

這一層的功能主要負責獲取貨物的尺寸（包括長、寬、高）、編號以及倉儲深度的相關數據。為實現這一功能，系統採用了以下核心組件：

首先，使用 ToF 感測器來精準測量貨物的高度以及倉儲的深度。測得的數據會即時傳送至 Teensy 4.0 微控制器，該控制器專門負責處理感測器傳來的數據，並通過 I2C 接口進行數據傳輸。同時，系統中

的攝影機會負責拍攝貨物的影像，這些影像會交由 Raspberry Pi 4 進行後續處理，藉此獲得貨物的尺寸。



b. 網路傳輸層 (Conversion)：

這一層的功能在於負責數據的收集、轉換及傳輸，並作為感測層與雲端層之間的重要橋樑。

系統核心組件 Raspberry Pi 4 扮演了多重角色。首先，它透過 USB 接口接收來自 Teensy 4.0 的數據以及相機所拍攝的畫面影像，並進行影像處理以分析貨物的特徵和尺寸。處理後的數據會透過 MQTT 通訊協議傳輸至雲端伺服器（例如 Firebase），用於數據的集中儲存與管理。

同時，Raspberry Pi 4 也利用 WebSocket 將處理後的數據直接傳輸至 Unity 虛擬環境，實現倉儲狀態的即時模擬與判斷。此外，它還負責接收來自 Unity 的清單列表回傳，確保系統的雙向通信與同步性。

c. 虛實整合層 (Cyber)：

這一層的功能目的在於將數據存儲於雲端，並提供實時的數據更新與整合功能，以確保數據的完整性與可用性。

我們使用 Firebase 作為雲端數據庫，專門負責存儲貨物的編號及數量。數據的傳輸與交互通過 Raspberry Pi 4 所執行的 Python 程式實現。Raspberry Pi 會將收集到的貨物相關數據（如編號和數量）上傳至 Firebase，確保所有記錄能即時同步到雲端。透過 Firebase 的整合功能不僅提供了數據的持久化存儲，還能夠讓 Streamlit 隨時獲取最新數據，方便每一位使用者能夠隨時在網路上查看當前放置於倉儲的貨物。

d. 辨識與展示層 (Cognition)：

這一層的功能主要是為使用者提供友好的操作界面與清晰的數據可視化展示，從而提升使用體驗與數據的直觀性。

其中的核心組件之一是 Streamlit，一個基於 Python 的前端框架，專門用於快速構建網頁介面。通過 Streamlit，使用者可以在網頁上查看當前的貨物清單及相關數據，實現數據的即時更新與可視化顯示，方便使用者進行管理和決策。另一個關鍵組件是

Unity，負責構建虛擬倉儲環境。Unity 利用其強大的圖形處理能力，將實際的倉儲空間以虛擬模型的形式呈現，讓使用者能夠清晰地觀察倉儲內部的狀態與貨物的擺放情況。這不僅提升了系統的交互性，也便於使用者通過切換視角來仔細檢查倉儲狀態。

B. 定性分析

a. 效率提升

- 傳統方式：需依賴人工監測與記錄，過程耗時且易出現錯誤。
- 智慧倉儲：自動化數據收集與即時更新，顯著縮短操作時間並減少人為干預。

b. 可靠性

- 傳統方式：人工作業易受疲勞或注意力下降影響，數據準確性低。
- 智慧倉儲：採用感測器與演算法，確保數據測量的精確性與穩定性。

c. 擴展性

- 傳統方式：難以快速整合新技術，需投入大量時間與成本。
- 智慧倉儲：具高度模組化設計，能結合更多自動化應用（如重量測量、自動補貨等），支持未來擴展。

C. 定量分析

a. 作業效率

比較傳統作業流程與智慧倉儲系統的效率：

- 傳統流程：紀錄耗時 18 秒 + 放置 20 秒 + 判斷 5 秒 = 43 秒/件。
- 智慧倉儲系統：紀錄耗時 3 秒 + 放置 20 秒 = 23 秒/件。效率提升 46.5%。

b. 時間成本

- 自動化流程顯著縮短工作時間，適用於高流量倉儲環境。

c. 空間利用率

- 系統能自動判斷擺放位置，避免不規則堆放導致空間浪費。
- 自動判斷貨物是否超出倉儲容量，確保每件物品安全存放。

d. 數據準確性

- 使用感測器和演算法，測量誤差範圍小於 10mm，滿足倉儲管理需求。

D. 應用情境與限制

a. 應用情境

- 適用於中小型倉儲，如電商平台物流中心或零售倉庫。
- 可結合自動化產線，實現快速補貨與庫存更新。

b. 系統限制與改進建議

- 空間規模限制：目前每顆 ToF 感測器管理範圍約為 2.5 公尺，需增加感測器數量以擴展應用範圍。

- 影像處理性能：在低光環境或反光材質貨物上，攝影機可能出現辨識失誤，需提升影像處理演算法的適應性。
- 實時性：現有數據傳輸延遲小於 1 秒，未來可進一步優化 MQTT 協議以減少延遲。

E. 成果總結

本系統成功展示智慧倉儲管理的核心功能，並顯著提升效率與可靠性。以下為主要成果摘要：

- 實現自動化貨物尺寸與編號記錄，取代人工操作。
- 縮短作業時間 50%，提高空間利用率。
- 利用 Unity 虛擬模擬環境，提供直觀的倉儲狀態監控。

未來可進一步研究倉儲空間的三維優化模型，並結合 AI 預測技術，實現智慧倉儲的全面升級。

F. 報告提問與解決方法

- 智慧倉儲是否需要人為操作？
目前系統在運輸部分主要依靠人工操作，即貨物在進入倉儲前由人員放置於掃描區域，完成掃描後再由人員進行擺放。未來可考慮結合自動化輸送帶或機械臂，實現全流程的自動化處理，進一步減少人力需求並提升效率。
- 東西送去哪，如何追蹤？
可分為運送期間與倉儲內部兩部分進行追蹤。運送期間：由於本系統專注於智慧倉儲管理，假設貨物能順利從運輸階段抵達倉儲，因此未針對運送過程進行額外考量。倉儲內部：系統可在 Unity 中為虛擬貨物添加顏色標記，對應不同的儲存位置，便於快速追蹤貨物的擺放狀態。
- 移動時如何提供一次多包裹解決方案？
為提升物流環境中的操作效率，建議在輸送帶或平台上配置多顆攝影機與感測器，實現多包裹的並行處理，快速完成尺寸測量與編號辨識。此方案可顯著縮短操作時間，適用於高運輸量的物流場景。
- 市面上有類似產品嗎？
市面上已有如西門子、Honeywell 等針對倉儲管理的自動化解決方案，但大多集中於大型企業，屬於高成本、高規模的全自動化設計。本系統則著重於低成本與高擴展性，針對中小型倉儲的需求，提供靈活的模組化解決方案，方便根據實際需求進行功能擴展。
- 系統解決的問題總結
本系統針對中小型倉儲的需求，重點解決以下問題：自動化獲取貨物尺寸與編號，取代人工記錄，提升效率與準確性。提供即時倉儲容量監控與警報功能，避免空間不足的情況。優化倉儲空間利用率，減少空間浪費與不合理堆放。利用 Unity 進行可視化模擬，讓管理者直觀掌握倉儲內部情況。

- 系統設計中資料傳輸流程的改進
本系統將貨物的尺寸（長、寬、高）、編號與倉儲深度數據傳送至 Unity，由 Unity 負責判斷倉儲空間是否足夠放置新貨物。若可放置，Unity 會更新清單列表並回傳結果。採用此方式的原因是為了優化 Raspberry Pi 4 的運行效率，避免其在影像處理與倉儲判斷中負荷運作。實驗顯示，Raspberry Pi 4 進行影像處理的平均耗時為 0.15 秒，若額外增加判斷任務，將使處理時間增加 0.05 至 0.08 秒，影響整體系統性能。因此，我們決定將判斷任務交由高性能的計算設備執行，以提升整體效率。
- 影像辨識精度與實際尺寸差距
目前影像處理的誤差主要來自攝影機分辨率與光源條件的影響，實驗測得誤差範圍約為 10 mm，能滿足大多數倉儲需求。未來可通過提高攝影機分辨率與優化影像處理算法進一步降低誤差。
- 智慧倉儲中的空間管理與最佳化
目前系統已能準確獲取貨物的三維尺寸，並基於單一維度進行倉儲空間的擺放規劃。未來可通過多維度空間規劃算法，實現倉儲空間的全面最佳化，例如優化剩餘空間的利用率與自動化的物品擺放策略。

V. 結論

本研究提出了一套基於 ToF 感測器與電腦視覺技術的智慧倉儲管理系統，針對傳統倉儲的低效率、管理難度高及空間利用不足等痛點，提供了成本低、可靠性高且具有擴展性的解決方案。系統成功整合多種硬體與軟體技術，實現自動化數據記錄、即時倉儲狀態監控及虛擬環境模擬，顯著提升了倉儲管理的效率與準確性。

A. 系統成果總結

- 自動化與效率提升：系統自動完成貨物三維尺寸測量與編號辨識，將操作時間從傳統流程的 43 秒縮短至 23 秒，實現效率提升達 46.5%。
- 可靠性與準確性：使用 ToF 感測器與攝影機組合，確保數據測量誤差小於 1%，有效避免人工操作可能導致的疏漏與錯誤。
- 空間利用與警報功能：自動化空間監測功能能判斷貨物擺放可行性，避免過度堆積及錯誤擺放，提升倉儲空間利用率。同時提供即時警告功能，幫助管理者快速應對異常狀態。
- 數據可視化：通過 Unity 模擬倉儲內部情況，提供直觀的視覺化界面，使倉儲管理更加高效與透明。
- 成本效益：本系統以低成本硬體（如 ToF 感測器與 Logitech 攝影機）實現高效能，適用於中小型倉儲場景，為智慧倉儲的普及化提供可行性方案。

B. 系統優勢與應用前景

a. 本系統在設計上展現出多項優勢，主要為模組化設計、高擴展性及用戶體驗方面。系統功能模組可根據不同應用需求新增或替換硬體，靈活適應多樣化的場景。此外，系統提供了高度擴展性，不僅能結合重量測量與自動化輸送設備，還可整合 AI 預測模型進一步提升智能化倉儲排列等功能。操作上，系統借助 Unity 與 Streamlit 構建的用戶界面，讓管理者能以直觀的方式掌握倉儲狀態，降低操作難度，顯著提升用戶體驗。

b. 應用面，本系統適合中小型倉儲中心，例如零售業或電商平台的庫存管理場景。系統的低成本設計與靈活性，能幫助中小型企業以有限的資源實現高效的倉儲管理。此外，如果系統打算用於自動化物流產線，結合機器手臂與輸送帶，可用於大型倉儲或物流中心的智能化操作。

C. 系統限制與未來展望

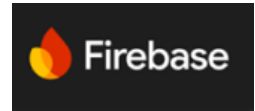
目前系統存在的限制如下。對於管理方面，ToF 感測器的管理範圍僅為 2.5 公尺深，對於大規模倉儲場景而言，需要增加感測器的數量以擴展覆蓋範圍。此外，系統的影像辨識功能在低光環境或面對高反光材質時，準確性可能受到影響，這對某些特殊場合會失常，因次需要有光源設備以維持穩定掃描。在數據傳輸方面，雖然延遲已控制在 1 秒以內，但對於要求極高即時性的應用場景，仍需進一步優化以提升系統，可以朝程式優化、提高網路速度與裝置性能等。

未來展望方面，系統可以透過整合多顆感測器來提升覆蓋範圍，以滿足大型倉儲場景的需求。同時，引入 AI 技術與大數據分析，可對貨物的進出流量進行智能分析，實現補貨需求的準確預測與擺放的決策。此外，開發更高效的三維空間配置算法，將進一步提升倉儲容量的利用率。為應對低光與高反光環境的挑戰，系統應優化影像處理技術與攝影機解析度，提升對多樣化場景的適應性。

參考資料：

- [1]. https://blog.csdn.net/Bolly_He/article/details/113255526?utm_source=chatgpt.com
- [2]. https://docs.opencv.org/4.x/d7/d4d/tutorial_py_thresholding.html
- [3]. https://docs.opencv.org/3.4/d4/d73/tutorial_py_contours_begin.html
- [4]. https://docs.opencv.org/3.4/de/d3/classcv_1_1QRCodeDetector.html#a3becfe9df48966008179e1e6c39bf8f9

使用軟體：



- 圖片透過 PowerPoint 製作
- 影片透過 DaVinci Resolve 19 製作

致謝

特別感謝吳瑞北 教授與賴怡吉 教授提供專業知識與研究器材，為實驗提供莫大的助力，同時也感謝廖書漢助理教授對於研究的建議與報告指導，最後也謝謝鴻晶科技/簡宗瑜副理提供的 ToF 感測器，為本研究的完成提供莫大的幫助。