

國立臺灣大學生物資源暨農學院生物機電工程學系

碩士論文



Department of Biomechatronics Engineering

College of Bioresources and Agriculture

National Taiwan University

Master Thesis

自主巡航無人機系統應用於溫室洋香瓜之定位及量測

Application of an Autonomous Drone Navigation System on  
Greenhouse Muskmelon Localization and Measurement

汪軍諺

Jun-Yan Wang

指導教授：林達德 博士

Advisor: Ta-Te Lin, Ph.D.

中華民國 112 年 07 月

July 2023

# 致謝



剛進實驗室的我知道自己能力不夠，接踵而至的壓力讓我窒息，感覺我似乎永遠無法追上身旁的同儕與老師對我的期望，挫敗感充斥在剛入學的階段。面對無數挑戰，我不斷地調適心態，過程的辛勞雖不能用數據量化，但實力的增進是有目共睹，從不會用程式的 function，到此刻無人機能在溫室自主飛行。在這裡想對愛我的家人宏順、雅鴻、珍瑋、軍叡與女友筠婷說聲抱歉，這兩年讓你們操心了，也謝謝你們陪我走過每個苦澀的夜晚，若沒有你們就真的沒有現在的我。

特別感謝我的指導教授林達德老師，在研究的每個航點上為我指點迷津，教導我思考實驗設計以及數據分析的方式。以往的我，面對新的挑戰經常害怕失敗，但老師總是以誨人不倦的方式鼓勵且相信我能做到，這兩年的教誨有如春風化雨薰陶了我。還記得在口試前最煎熬的時刻，老師對我說：「論文最後階段一向是最辛苦的，也是重要的鍛練過程，要對自己有信心，也能堅持到底。畢業後你再回首就會感受到過程的意義與甘甜。加油！」。而此刻的我回首一望，過往的煎熬全成了滋養我茁壯的養分，所有的汗水與眼淚，都已化為閃亮的鑽石。

口試完的當下，深深體認到要感謝的人有太多。感謝口試委員楊江益老師，細心的審閱論文，幫學生找到論文的弱點。感謝口試委員顏炳郎老師，提供許多寶貴的意見，讓無人機的系統在日後得以持續精進。感謝大鍋提供有強大 GPU 的 Toolmen 讓我們訓練模型，讓研究順利產出。感謝林裕彬老師與鄭鎌杰助理提供智慧溫室，能在校內有一個溫室場域使用真的太幸福了。感謝璟寬與少政一起在知武館 405 打拼，一起品味研究過程的酸甜苦辣，有你們一起嘴砲一起玩，使平淡的日子也充滿歡笑。感謝桂芝協助所有報帳的事務，讓我們能順利參與 ISMAB 與 ASABE、外縣市溫室出差、採購研究資源。感謝 Lab405 的所有夥伴，在這兩年間的陪伴，能與你們一起學習、吃飯、出遊，是我永生難忘的時光。感謝陪我打遊戲、衝浪的朋友們，讓我偶爾能從忙碌的日子裡抽離，喘口氣再繼續努力。最後感謝敬恆把無人機的題目交給了我，這個題目真的不簡單，常常遇到各種惱人的問題（筆電無人機過熱、溫室光照變化、收果實資料...），但你總在我遇到難題時，適時地給予建議，幫助我很多，接下來無人機就交給妳啦徐滋！

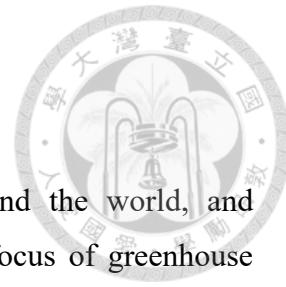


## 摘要

目前全世界存在大量規模龐大的溫室，而溫室的作物生長狀況監測為溫室管理的一大焦點。傳統的人工巡查監測方式耗時且需要投入大量人力資源，使種植者無法快速、即時地了解溫室當前的整體情況。然而，自動化的無人機導航監控系統能夠解決此難題，基於純視覺定位的無人機不需安裝昂貴的傳感器，僅需搭載 RGB 相機即可執行導航任務。視覺定位無人機自主導航系統成為低成本的溫室自動監測之核心，從技術上改變了智慧農業的樣貌。本研究的無人機自主巡航系統，使用了加入 ArUco Marker 的 ORB-SLAM2，稱為 Enhanced ORB-SLAM2。ArUco Marker 是一種標誌，具有特定的幾何模式，通常被用作計算機視覺中的參考點，能夠幫助系統進行位置定位和追蹤。ORB-SLAM2 則是一種視覺定位與地圖構建演算法，能夠使用相機實現同時定位和地圖構建。實驗驗證了 Enhanced ORB-SLAM2 在環境存在陰影特徵的定位結果優於原始的 ORB-SLAM2。此無人機導航系統可於溫室中自動執行各式飛行任務，且飛行軌跡均方根誤差範圍在 30 公分以下。此外，地圖校正使用仿設轉換算法，可以使地圖的 MapAruco 與人工量測的 ArUco Marker 位置完全貼合。果實偵測使用 YOLOv4 深度學習模型訓練，果實偵測模型之 mAP 達到 0.96，DeepSORT 基於此果實偵測模型運行果實追蹤任務。將 DeepSORT 的追蹤結果經過三步驟的資料清理後，使假果實實驗中的 ID switch 數量由平均 5.83 顆，下降至 0 顆，達到準確追蹤之目標。溫室果實定位算法基於清理後的果實追蹤結果，並使用三角測量算法計算果實位置，計算求得之果實位置再分別使用地圖校正以及迭代 ArUco Marker 的校正常數做位置校正，校正後假果實位置的均方根誤差由 2.758 公尺下降至 0.223 公尺，此果實追蹤與定位算法也已驗證可應用於真實果實之追蹤與定位。

關鍵詞：自動化、無人機、自主導航、ORB-SLAM2、果實追蹤、果實定位

# Abstract



There are a large number of large-scale greenhouses around the world, and monitoring the growth of crops in these greenhouses is a major focus of greenhouse management. Traditional manual inspection and monitoring methods are time-consuming and require a significant amount of human resources, preventing growers from quickly and instantly understanding the overall situation within the greenhouse. However, automated unmanned aerial vehicle (UAV) navigation and monitoring systems can address this challenge. UAVs equipped with pure visual positioning, without the need for expensive sensors, can perform navigation tasks with an RGB camera. In this study, an autonomous UAV navigation system was developed using Enhanced ORB-SLAM2, which incorporates ArUco Markers. ArUco Markers are specific geometric patterns used as reference points in computer vision, aiding in location estimation and tracking. ORB-SLAM2 is a visual positioning and mapping algorithm that achieves simultaneous localization and mapping using a camera. The experiment demonstrated that Enhanced ORB-SLAM2 outperforms the original ORB-SLAM2 in positioning results when dealing with environments containing shadow features. This UAV navigation system can perform various flight missions within a greenhouse autonomously, with a root mean square error of the flight trajectory within 30 centimeters. Additionally, map calibration using a similarity transformation algorithm ensures the alignment of the MapAruco with manually measured ArUco Marker positions. Fruit detection employs the YOLOv4 deep learning model, achieving a fruit detection model mAP of 0.96. DeepSORT utilizes this fruit detection model for fruit tracking. After three steps of data cleaning on DeepSORT's tracking results, the average ID switch count in false fruit experiments decreased from 5.83 to 0, achieving accurate tracking. The greenhouse fruit localization algorithm utilizes the cleaned fruit tracking results. Using triangulation, fruit positions are calculated and subsequently corrected using map calibration and iterative ArUco Marker calibration. The root mean square error of false fruit position decreased from 2.758 meters to 0.223 meters after correction. This fruit tracking and localization algorithm has also been validated for real fruit tracking and localization applications..

**Keywords:** automation, unmanned aerial vehicle, autonomous navigation, ORB-SLAM2, fruit tracking, fruit localization

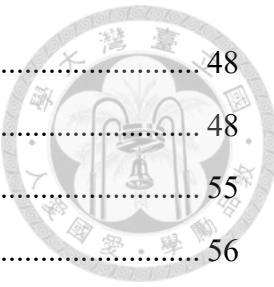
# 目錄



致謝 .....	1
摘要 .....	ii
Abstract.....	iii
圖目錄 .....	vii
表目錄 .....	x
第一章 緒論 .....	1
1.1 前言 .....	1
1.2 研究目的 .....	3
第二章 文獻探討 .....	4
2.1 無人機與智慧溫室 .....	4
2.2 無人機於室內室外之應用 .....	5
2.2.1 室外無人機之應用與發展 .....	5
2.2.2 室內無人機之應用與發展 .....	5
2.2.3 室內無人機的困難點與案例 .....	6
2.3 室內無人機巡航與 SLAM .....	6
2.3.1 溫室無人機巡航 .....	6
2.3.2 即時定位與地圖構建 .....	6
2.4 作物表型分析 .....	9
2.4.1 作物表型分析技術 .....	9
2.4.2 作物產量預估 .....	11
2.4.3 作物定位與量測 .....	12
第三章 研究方法 .....	13
3.1 自主導航系統架構 .....	13
3.1.1 硬體架構 .....	13



3.1.2 飛行控制軟體架構 .....	16
3.2 ORB-SLAM2 .....	17
3.2.1 Enhanced ORB-SLAM2 .....	17
3.2.1 視覺地圖建立 .....	17
3.2.2 Enhanced ORB-SLAM2 地圖校正 .....	18
3.3 飛行控制系統 .....	19
3.3.1 無人機 3D 位姿估計 .....	19
3.3.2 導航路徑設置與抵達航點之判斷方法 .....	20
3.3.3 無人機控制系統與控制閥設計 .....	21
3.4 果實資訊分析 .....	22
3.4.1 果實追蹤演算法 .....	22
3.4.2 果實定位演算法 .....	25
3.4.3 果實大小演算法 .....	29
3.5 實驗設置 .....	30
3.5.1 實驗場域 .....	30
3.5.2 ArUco Marker 之規格與製作方法 .....	31
3.5.3 實驗設計與目的 .....	31
3.6 程式檔案與操作流程 .....	34
第四章 結果與討論 .....	37
4.1 Enhanced ORB-SLAM2 定位分析 .....	37
4.1.1 ORB-SLAM2 定位效果分析 .....	37
4.1.2 Enhanced ORB-SLAM2 定位效果分析 .....	39
4.1.3 ORB-SLAM2 與 Enhanced ORB-SLAM2 定位效果比較 .....	40
4.2 地圖校正 .....	40
4.2.1 ICP 方法之地圖校正 .....	40
4.2.2 仿射轉換方法之地圖校正 .....	41
4.2.3 校正地圖之定位效果分析 .....	43
4.2.4 SLAM 建圖可視化結果 .....	45



4.3 飛行控制分析 .....	48
4.3.1 SLAM 系統與飛行軌跡分析 .....	48
4.3.2 轉彎 PID 參數調整 .....	55
4.3.3 不同 SLAM 系統之差異比較 .....	56
4.4 果實算法分析 .....	57
4.4.1 果實偵測模型效能分析 .....	57
4.4.2 假果實追蹤結果分析 .....	58
4.4.3 假果實定位結果分析 .....	63
4.4.4 假果實大小結果分析 .....	74
4.4.5 真實果實追蹤結果分析 .....	75
4.4.6 真實果實定位結果分析 .....	76
4.4.7 真實果實大小結果分析 .....	80
第五章 結論與建議 .....	82
5.1 結論 .....	82
5.2 建議 .....	84
參考文獻 .....	85

# 圖目錄



圖 1-1 自主巡航無人機系統應用於溫室洋香瓜之整體研究架構圖 .....	3
圖 3-1 自主巡航無人機系統架構示意圖 .....	13
圖 3-2 DJI Tello 無人機 .....	14
圖 3-3 Intel RealSense D435 深度攝影機：(a)攝影機實體圖；(b) rqt_graph 中 rostopic 訂閱關係 .....	15
圖 3-4 自主巡航無人機軟體架構圖 .....	16
圖 3-5 Pygem 仿射轉換之官方範例：(a)原始點之空間分布圖； (b)原始控制點與變形控制點；(c)變形點之空間分布圖 .....	19
圖 3-6 使用手動設定之航點經由系統自動生成多個細部航點 .....	20
圖 3-7 ORB-SLAM2 位置更新與無人機控制訊號之控制閥架構 .....	22
圖 3-8 YOLOv4 果實偵測模型之 RGB 訓練影像 .....	23
圖 3-9 果實追蹤結果儲存之欄位與意義示意圖 .....	24
圖 3-10 相機內參使用之棋盤格標定板：(a)9x7 棋盤格；(b)棋盤格角點檢測結果	25
圖 3-11 ArUco 校正常數自動迭代求導之範例圖 .....	28
圖 3-12 國立臺灣大學農業試驗場：(a)溫室上視簡圖；(b)夏系 2 號日本阿露斯洋 香瓜；(c)溫室走道實體圖 .....	30
圖 3-13 ArUco 樣式與尺寸：(a)ArUco Marker 尺寸；(b)ArUco Marker 板尺寸 .....	31
圖 3-14 數字照度計（型號: HA-5010A） .....	32
圖 3-15 捲尺：(a)量測果實位置之鋼捲尺；(b) 量測果實圓周之軟捲尺 .....	33



圖 4-1 ICP 校正溫室 ArUco Marker 位置，紅色方塊為地圖中的 MapAruco 位置，藍色方塊為校正後的 ArUco 位置，黑色方塊為實際在溫室量測的 ArUco 位置，(a)ICP 在全局最佳解；(b)ICP 在局部最佳解.....	41
圖 4-2 仿射轉換校正將地圖之 MapAruco 校正至溫室的真實 ArUco Marker 位置	42
圖 4-3 原始地圖與校正地圖之計算方法與三種飛行軌跡.....	43
圖 4-4 原始地圖軌跡與校正地圖軌跡之比較圖 .....	44
圖 4-5 五種飛行實驗之軌跡：(a)直線；(b)倒退繞圈；(c)轉彎繞圈；(d)全溫室繞圈；(e)全溫室 S 型 .....	49
圖 4-6 直線飛行之 3D 空間軌跡圖與誤差圖 .....	52
圖 4-7 倒退繞圈之 3D 空間軌跡圖與誤差圖 .....	53
圖 4-8 轉彎繞圈之 3D 空間軌跡圖與誤差圖 .....	54
圖 4-9 P 值過大導致轉彎產生擺動之角度圖與角度誤差圖 .....	55
圖 4-10 無人機轉彎定位效果分析：(a)在無 aruco 區域使用 ORB-SLAM2；(b)在有 aruco 區域使用 ORB-SLAM2；(c)Enhanced ORB-SLAM2 .....	57
圖 4-11 假果實追蹤結果之不同 ID switch 分析.....	60
圖 4-12 Exp. III-1 之 ArUco marker 定位結果分析 ArUco Marker 常數校正之差異： (a)常數校正前；(b)常數校正後 .....	64
圖 4-13 Exp. IV-1 之果實定位結果分析 ArUco Marker 常數校正之差異：(a)常數校正前；(b)常數校正後 .....	67
圖 4-14 Exp. IV-1 之果實位置誤差圖：(a)原始三角測量結果；(b)仿射轉換地圖校正結果；(c)仿射轉換地圖校正與 ArUco Marker 常數校正結果 .....	71
圖 4-15 Exp. IV-2~Exp. IV-6 之果實位置誤差圖 .....	72



圖 4-16 三角測量結果與無人機使用斜率判斷果實位置：(a)果實於無人機右側之三角測量點於空間分布圖；(b)果實於無人機左側之三角測量點於空間分布圖；  
(c)果實於無人機右側之 2D 上視圖與斜率分析；(d)果實於無人機左側之 2D 上視圖與斜率分析 ..... 73

圖 4-17 真實果實個別果實位置誤差圖：(a) Exp.1；(b) Exp.2；(c) Exp.3 ..... 78

圖 4-18 真實果實空間位置圖：(a) Exp.1；(b) Exp.2；(c) Exp.3 ..... 79



# 表目錄

表 2-1 LIDAR SLAM & VSLAM 比較表.....	7
表 3-1 DJI Tello 無人機技術規格表 .....	14
表 3-2 Robot Localization 以 15 維向量表示機器人運動狀態 .....	20
表 3-3 Upper and lower bounds 迭代運行調整邏輯.....	28
表 3-4 無人機自主導航 ROS 系統之程式說明 .....	35
表 3-5 果實追蹤、定位、大小演算法之程式說明 .....	36
表 4-1 不同光照環境之 ORB-SLAM2 定位效果 .....	38
表 4-2 不同光照環境之 Enhanced ORB-SLAM2 定位效果.....	39
表 4-3 原始地圖與校正地圖之平均特徵點數量與追蹤率 .....	44
表 4-4 ORB-SLAM2 建立之三種地圖 .....	46
表 4-5 Enhanced ORB-SLAM2 建立之三種地圖.....	47
表 4-6 九種無人機飛行實驗控制變因設定 .....	48
表 4-7 九種無人機飛行實驗之平均均方根誤差（空間、角度）與平均使用幀數.	51
表 4-8 YOLOv4 果實偵測模型之訓練結果 .....	58
表 4-9 假果實追蹤結果使用三步驟資料清理前後之分析.....	59
表 4-10 調整 Size Threshold 與 Frame Threshold 之假果實追蹤 ID switch 數量差異 .....	62
表 4-11 假果實追蹤結果做第三步驟資料清理之有效性驗證分析 .....	63
表 4-12 三角測量 ArUco Marker 位置經常數校正前後之均方根誤差結果.....	65



表 4-13 Exp. III 六個實驗使用 ArUco Marker 自動迭代之校正常數 .....	66
表 4-14 假果實位置使用地圖校正與 ArUco marker 常數校正前後之均方根誤差 ..	69
表 4-15 假果實實驗之果實大小計算結果 .....	74
表 4-16 真實果實追蹤使用不同閾值之結果 .....	75
表 4-17 真實果實追蹤結果使用三步驟資料清理前後之分析 .....	76
表 4-18 真實果實位置使用地圖校正與 ArUco marker 常數校正前後之均方根誤差 .....	77
表 4-19 真實果實實驗之果實大小計算結果 .....	80



# 第一章 緒論

## 1.1 前言

農業一直是人類生存和發展的基石，傳統農業需要大量的農民勤勞耕耘才能獲得豐盈的收成。但近年來，由於產業結構的快速變化，鄉村人口紛紛湧向城市，導致從事農業的人口急劇下降 (Lobao and Meyer, 2001)。另外，根據 Hoornweg 等人(2017)的研究顯示，到了 2100 年，全球人口可能會達到 131 億。農業工作者的減少加上世界人口持續增加，讓農業自動化變得相當重要且緊迫，溫室農業因為不受天氣影響，且能穩定產量並維持品質，因此成為了一個非常受歡迎的新興產業。在傳統農業中，為了保持作物健康和良好的生長狀態，需投入大量人力資源定期進行作物環境巡查和管理，其中包括耕作、施肥、除草、除蟲等各種工作，這也使得栽種成本隨之上升。此外，傳統農業需要農民多年的工作經驗，才能掌握栽培技巧和經驗，並獲得高品質的產量。然而，對於栽種經驗不足的農民來說是難以克服的障礙。因此，智慧農業應運而生，隨著科技的進步，利用各種感測器在溫室中蒐集溫濕度、光照、影像等數據，再透過數據分析的方式，即可提供農民即時的溫室狀況，以便隨時調整環境控制設備，讓作物得以在最佳環境中生長。例如，利用物件追蹤與深度學習技術，可分析溫室中的作物數量、大小、成熟度、品質等，農民可以更加有效地控管作物生長狀態，提高作物產量和品質。智慧農業技術不僅減少人力需求，也降低農業入行門檻，為年輕農民提供更好的技術支持，使得農業更具發展潛力。

智慧農業中的自動蒐集影像技術，常使用自動化機器人收集表型數據。這些機器人可分為軌道式和非軌道式 (Tangarife and Díaz, 2017)；軌道式機器人只需感測器偵測作物位置，無需考慮定位問題。如 Zhao 等人 (2016) 開發的雙三軸機械手臂機器人可協助採收番茄，提高生產效率；非軌道式機器人則利用感測器在溫室中自主巡航，例如 Vakilian 等人 (2012) 使用履帶式機器人自主導航，在表型分析方面比人類視覺更為敏銳，可提早兩天發現缺氮的黃瓜植株。

農業自動化與作物檢測緊密相關，作物檢測需確定位置並分析表型，以指導機器人採摘。然而，多變的光照及枝葉遮蔽限制了準確性。Gongal 等人(2016)以大型傳感器平台分析不同角度的蘋果影像，將準確度由 58% 提升至 82%。Mehta 和 Burks(2014)利用單目相機估計橘子深度，手臂採摘誤差在 15 cm 以內。Li 等人(2011)認為對於基於機器視覺的收割機而言，作物定位與檢測是最基本且重要的資訊，提升定位與檢測的效能，有助於提高作物產量估計 (crop-load estimation) 的準確性。產量估計對農業生產者極為重要，可幫助有效規劃種植、收穫、供應管理，確保時機成熟投放市場，提升銷售和利潤，提升生產效率和經濟效益。

雖智慧農業技術能有效解決農民在作物管理上的困難，但機器也有其限制，例如在不同溫室使用時，需要調整機器人的尺寸、感知器的位置和數量等，才能符合不同作物和溫室環境的需求。Fonteijn 等人(2021)指出，使用機器人在溫室中記錄植物表型需要更多攝影機以實現更全面的拍攝角度，但增加感知器的數量也意味著成本上升。相對地，使用室內無人機能夠在溫室的任何高度進行植物表型的記錄，有效地解決了這個問題。

以往無人機在大眾印象中的特徵是龐大的機身和高昂的售價，但由於近年來無人機產業蓬勃發展，相關技術迅速變革，技術的成熟降低了機身大小和售價，這使得研究者開始將無人機應用於溫室中。Roldán 等人 (2015) 對四旋翼無人機的空氣動力學進行了分析，確定了在無人機上安裝感知器的最佳位置為四旋翼頂部的正中心，並實際安裝了多種感測器以得到空氣溫溼度、光亮度和二氫化碳濃度。而無人機的應用不僅侷限於表型分析，Shi 等人 (2019) 試圖利用無人機在溫室中為花卉授粉，在無人機上安裝導流板以增加風速。無人機自主導航須先建立完整的點雲圖，使 Simultaneous Localization and Mapping (SLAM) 系統計算無人機的位置，Roldán 等人 (2016) 將無人機與地面機器人組合成一組異構機器人，透過實體線路連接，手動操控無人機搭配自主導航的地面機器人，以結合無人機的敏捷性與機器人的穩健性，使室內點雲圖快速建立，以進行導航任務。Krul 等人 (2021) 驗證了小型無人機在溫室和牧場中使用 Visual Simultaneous Localization and Mapping (VSLAM) 定位的可行性，使無人機自主飛行於農業環境。

近年來將無人機應用於智慧溫室的研究文獻日益增多，也逐漸證實了其技術的可行性和價值。目前，在溫室中使用無人機仍存在諸多問題，例如地形和障礙物、氣流擾動、自主巡航、電力續航、數據傳輸等問題。學者們正逐一解決這些問題，使無人機能穩定飛行在溫室中，並精準分析數據提供農民參考。未來，無人機將成為智慧溫室中無可取代的一部分，推動智慧農業的改革，實現農業自動化的願景。

## 1.2 研究目的

本研究的目的為使用 VSLAM 及小型無人機，開發溫室無人機自主巡航果實檢測系統。研究採用 ORB-SLAM2 作為視覺定位基礎演算法，並新增 ArUco Marker 以提高地圖的穩定性，同時使用 ArUco Marker 校正點雲地圖。無人機將拍攝影像回傳至地面站，接著通過果實定位演算法計算出溫室果實的位置與大小，以幫助溫室管理人員管控當前的溫室狀況。

本研究主要研究目標條列如下：

1. 建立溫室無人機自主巡航系統
2. 建立果實追蹤演算法
3. 開發果實定位與大小演算法
4. 利用果實位置分析溫室果實之數量與分布

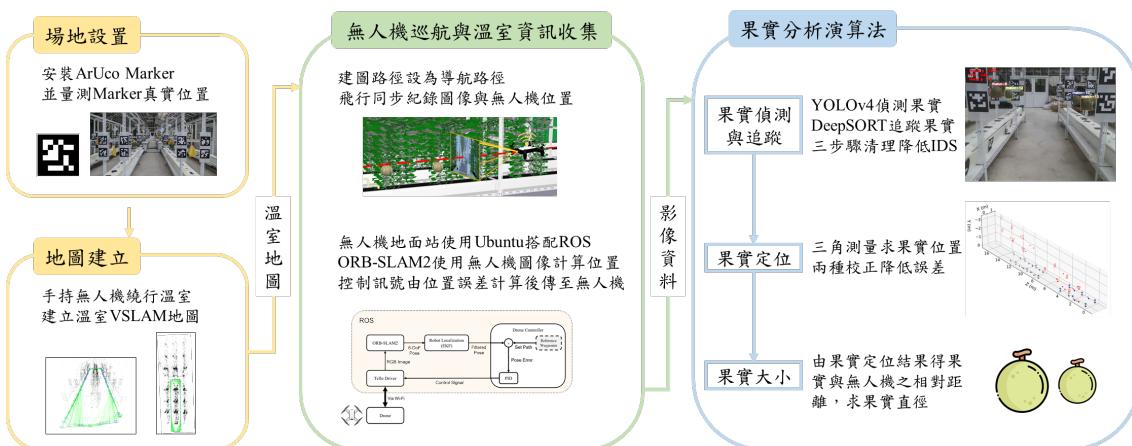


圖 1-1 自主巡航無人機系統應用於溫室洋香瓜之整體研究架構圖



## 第二章 文獻探討

### 2.1 無人機與智慧溫室

在溫室中農民可以有效的掌控環境因素，優化作物的生長環境，同時降低病蟲害所帶來的農業損失。二氧化碳、水分、光照度與溫濕度是溫室作物生長的重要因素 (Iddio et al., 2020)，透過安裝感測器的方式，能使智慧溫室透過感測器的資訊行更精準的控制環境。

Liyang 等人 (2018)比較傳統溫室與智慧溫室的不同，智慧溫室建立開花期模型以及結果期模型，預測作物的生長狀況並適時地給予水分與養分，改善小黃瓜的成長條件。其研究結果顯示，智慧溫室的栽種方式可大幅降低溫室的勞動力需求，並且使小黃瓜產量提升 13~20%，與此同時降低了 22% 的用水量。Abbasi-Kesbi 等人 (2020)透過無線感測器網路紀錄溫室中的溫濕度、CO、CO<sub>2</sub>、光照度，並結合灌溉系統直接管理溫室，實驗結果表明，在番茄產量提升 30% 的狀況下，甲烷氣體消耗量、用水量、用電量分別減少了 30%、24%、10%。

上述在溫室中使用的感測器皆採定點設置，雖穩定性高，但需要依照溫室的大小與需求安裝不同數量的感測器，而更多的感測器意味著成本的提升。因此，輕巧且具備高機動性的無人機成為備受矚目的替代方案，無人機上可安裝不同的感測器，在飛行時紀錄溫室的資訊，當溫室管理者需要不同高度的溫溼度資訊，僅需調整飛行路徑即可完成資訊蒐集，不需再逐一安裝固定式感測器，減少了安裝的程序與成本。Roldán 等人 (2015)在無人機上安裝了多種感測器以得到空氣溫溼度、光照度、二氣化碳濃度。

目前無人機的主要困難點在於飛行的時間受電池電量與工作的複雜度所影響，需要紀錄的資訊越多，也代表無人機必須搭載更多的感測器，其乘載的重量越大，電池的消耗速度越快。無人機在農業中被應用在許多地方，包含農作物的分類與偵測、施肥、乾旱檢測、產量估算、害蟲檢測、雜草檢測、疾病檢測。農用無人機的相關研究從 2011 的 30 篇到 2020 年的 1275 篇，可以明顯看出無人機在農業

中廣受關注 (Rejeb et al., 2022)。根據國際無人駕駛車輛系統協會 (AUVSI) 的數據所示，在未來將有 80%的無人機被應用在農業上。



## 2.2 無人機於室內室外之應用

### 2.2.1 室外無人機之應用與發展

近年來許多研究者投身開發農用無人機，而比較室外與室內無人機的文獻數量可以發現，室外無人機的文獻占多數，研究主題如：農作物監測、雜草檢測、噴灑農藥與肥料等。如 López-Granados 等人 (2016) 使用空拍的多光譜影像分析玉米田的雜草位置，能精確的將除草劑噴灑在雜草密集的區域，節省 85~96%的除草劑使用量；Kalischuk 等人 (2019) 使用無人機所拍攝的多光譜影像分析西瓜田的病害，其早期疾病檢測能力比傳統方法提升 20%以上。室外無人機為研究大宗的主要因素為無人機的原始設定是應用在戶外，會搭載抗風系統穩定無人機飛行，透過氣壓計估算飛行高度，而其龐大的機身導致不易於室內飛行。室外無人機可以使用 GPS 訊號定位，GPS 訊號的精度大約在 1~20 英尺，此飛行誤差對於拍攝高空影像是容易被接受的。此外，室外無人機飛行可得到高分辨率且低成本的影像，相較昂貴又不易取得的衛星影像，無人機更受喜愛。

### 2.2.2 室內無人機之應用與發展

由於近年來無人機產業的興起，無人機的技術迅速成長，機身逐漸縮小，使室內無人機的相關研究與日俱增。事實上，溫室環境非常適合無人機自主巡航，溫室內可不受天候影響的工作，且其機動性高，能測量溫室中任意位置的數據。溫室無人機可以設定特定時間（每週、每日、每小時）連續監測作物，並監測作物個體生長變化，搭配與植物生長相關的各項指標，即可了解作物的健康狀況。除此之外，無人機還有許多應用，單就影像為例產量估計、疾病檢測等，可見其經濟價值極高，乃實現精準農業不可或缺的一部分。



### 2.2.3 室內無人機的困難點與案例

由於建築物的遮擋，在室內無法使用 GPS 訊號定位，因此需使用室內定位技術。除了定位無人機在溫室中還有其他的困難，如：動態與靜態障礙物、光照變化、狹小走道的控制精準度等，這些問題提升無人機在溫室中的飛行難度。Aslan 等人 (2022)認為無人機在室內應用會比室外環境更加困難，其中定位是最主要的問題，也因其困難性，溫室中的無人機研究數量較少，且若要在溫室中精確地定位，必須在溫室內使用即時定位與地圖建構協助無人機自動飛行。Roldán 等人 (2016)同時使用 Unmanned Ground Vehicle (UGV)以及 Unmanned Aerial Vehicle (UAV)，結合了兩者的優勢，當 UGV 遇到障礙物時，使用 UAV 蒐集更全面的資訊，使此系統能蒐集更多角度的溫室資訊，由於此系統的 UAV 採用實體連接 UGV，因此不需要背負沉重的電池，僅需透過 UGV 供電即可。溫室中的無人機也不僅僅侷限於作物的表型分析或是溫室的溫溼度紀錄，無人機也被使用在人工授粉以及消滅害蟲，如 Amador 等人 (2017)使用電場將馬匹的毛垂直豎立於接觸表面，再透過凝膠固定，可成功沾取溫室花朵上的花粉；荷蘭的無人機開發商 PATS 推出一款消除溫室害蟲的無人機，使用無人機的懸翼殺滅飛行中的害蟲，以維護溫室環境。

## 2.3 室內無人機巡航與 SLAM

### 2.3.1 溫室無人機巡航

由於大部分的溫室任務皆具備重複性，因此使用自動化的機器人如 UAV、UGV 取代人力，是有效率且具有經濟價值的技術。為了達成室內的巡航工作，在溫室中定位出無人機的位置至關重要，而 SLAM 技術可以應用在室內定位。

### 2.3.2 即時定位與地圖構建

多數的室內無人機研究皆會使用即時定位與地圖建構演算法 (Bailey and Durrant-Whyte, 2006; Durrant-Whyte and Bailey, 2006)。SLAM 的基礎概念是使機器人在未知的環境中移動，在連續的運動過程中持續定位，並同步記錄環境中的特

徵建立地圖，完成建圖後即可令機器人移動至使用者設定的位置，以執行特定任務。而 SLAM 主要分為兩種，分別為使用光達 (light detection and ranging, LIDAR) 的 LIDAR SLAM 以及使用攝影機的 VSLAM，兩者的差別列舉如表 2-1 所示。



表 2-1 LIDAR SLAM & VSLAM 比較表

	LIDAR SLAM	VSLAM
Cost	High	Low
Affected by external light sources	No	Yes
Map accuracy	High	Low
Cumulative error	No	Yes
Map language information	No	Yes
Usability	High	Low
Stability	High	Low

雖然 LIDAR SLAM 的技術相對成熟，且穩定度與易用性都略勝一籌，但由於其應用成本過高，在此實驗不被考量。而 VLSAM 具有成本低的優勢，僅需使用 RGB 相機即可運行，其中 VSLAM 由相機的種類被分為兩種，第一種為 RGBD 深度相機，如 Kinect；第二種為 RGB 的單目相機、雙目相機、魚眼相機等。RGBD 深度相機與 LIDAR SLAM 的技術相似，皆透過飛時測距 (time of flight, ToF)，以確認該物體與相機的距離。

Lachlan 等人 (2018) 在無人機上安裝 2D LIDAR 並使用 LIDAR SLAM 在室內建立點雲圖，其地圖的精度約為 2cm，但建議後續的實驗採用 3D LIDAR 以應付更複雜的環境。Qin 等人 (2019) 先使用搭載 3D LIDAR 的 UGV 在地面上建立一個



粗略的點雲圖，再將此地圖傳至 UAV 供其使用，UAV 基於此地圖建立更加精確的點雲圖，此系統利用 UAV 的機動性與 UGV 較強的計算資源，可快速地建立未知空間的 3D 點雲圖。

VSLAM 中分成了直接方法與特徵方法 (Huletski et al., 2015)。直接方法如 Jakob 等人 (2014) 所提出的 LSD-SLAM (large-scale direct monocular SLAM)，直接方法顧名思義不須提取特徵，直接使用影像建立地圖，這也代表著比需要提取特徵的 ORB-SLAM、MonoSLAM、PTAM 等更加穩健且詳細，而其缺點為計算的成本相對特徵方法高，但仍然可以在 CPU 上實時運行。另外 ORB-SLAM 為一種特徵方法，由 Mur-Artal 等人 (2015) 在 2015 年提出，是基於 ORB 特徵 (Oriented FAST 和 Rotated Brief) 的演算法，ORB 特徵由 Brief 的快速二進制描述符所改良，其計算速度比 SIFT 快兩個數量級 (Rubleev et al., 2011)，而 ORB 特徵具有旋轉和尺度不變性，且不易受雜訊影響，因此計算與匹配特徵的速度非常快，不須使用 GPU 即可實時運行。而 2017 由 Mur-Artal 等人 (2017) 所發布的 ORB-SLAM2 可適用於三種類型的相機（單目相機、雙目相機和 RGBD 相機），其中包括地圖重利用、閉環檢測和重新定位的能力。此外，在 Krul 等人 (2021) 的研究指出，基於特徵方法的 ORB-SLAM 比直接方法的 LSD-SLAM 有更小的建圖誤差，而溫室中的走道狹窄，誤差需要被放大檢視，以維持其飛行穩定性。



## 2.4 作物表型分析

表型一詞最早由一位丹麥的植物學家與遺傳學家，威廉·路德維格·約翰森 (Wilhelm Ludvig Johannsen) 所提出，在 1905 年他以相同大小的豆子種植出大大小小的豆子，證明了不同表型的作物，仍然保留了相同的基因型。表型可由不同生物體中觀察，如：植物的高度、大小、顏色、形狀等，皆為表型的一種。表型受基因型以及外部環境的交互作用所影響 (Peirson, 2013)。

### 2.4.1 作物表型分析技術

作物表型分析以取像尺度分成三種，顯微表型、地面表型、空中表型，不同尺度在表型分析上有不同的應用 (Song et al., 2021)。

顯微表型分析近年來廣受人們的關注，其發展方向朝快速、非侵入式的測量作物表型。如 Hughes 等人 (2017) 使用 X-ray μCT 以非破壞性、非侵入性的方式準確地測量小麥麥穗的表型。Xiong 等人 (2019) 提出一種基於 3D 表型分析的新式 CT 影像處理流程，該方法可以準確的提取穀物組織的表型資訊，且實驗證明其準確度優於人類的判斷。另外，該研究定義一組新的小麥 3D 表型，包含縱橫比 (aspect ratio)、孔隙率 (porosity)、穀物間距離 (grain-to-grain distance) 和穀物角度 (grain angle)，並且發現縱橫比、孔隙率、穀物間距離三者與環境因素有關，而穀物角度則是與穀物的體積相關。此研究證明了作物的表型隨特定條件變化，因此，種植者可依據其需求，改變種植環境，以得到預期的作物表型。

地面表型分析可分為三種，便攜式 (portable)、固定式 (stationary)、移動式 (movable)。便攜式表型分析的特點為易於操作、方便攜帶、低成本，如 Aquino 等人 (2018) 開發手機的 APP，直接使用智慧型手機的 RGB 相機，即可計算葡萄串的果實數量，該系統分析 144 張影像所需的計算時間在 3.14 秒到 8.40 秒之間。而不僅僅是 RGB 相機被應用在便攜式的系統，目前紅外線熱像儀、多光譜相機等，皆逐漸被導入 (Araus and Kefauver, 2018)，便攜式雖然方便，但其相對限制了檢測規模與高通量表型分析中的可行性。因此，可攜帶多種感知器的固定式表型分析平台應運而生，在該平台上可以安裝 RGB 相機、多光譜相機 (multispectral)、高

光譜相機 (hyperspectral)、熱像儀 (thermal)、螢光偵測器 (fluorescent)、RGB-D 相機 (3D imaging sensors) 以增強表型分析的精準度。如 Virlet 等人 (2016) 開發了一個全自動田間表型數據蒐集平台，該平台安裝高分辨率的 RGB 相機、葉綠素螢光感知器、紅外線熱像儀、兩個高光譜相機、兩個 3D LiDAR，並且能夠在三維空間中準確地定位，可持續蒐集高通量表型數據，並記錄作物的生長變化。移動式表型分析平台即在農用車輛上添加感測器，其移動性使一組感測器不再侷限於特定區域，而是在空間中自由移動，以更加全面的方式蒐集目標場域的表型數據。如 Jimenez-Berni 等人 (2018) 手動控制電動的地面機器人在室外耕地上行走，並透過搭載的 LIDAR 精準地測量植株高度 (canopy height)、地被植物 (ground cover)、地面生物量 (above-ground biomass)。且亦可搭配 GPS 定位使機器人依循預定的導航路線行走，以自主巡航的方式完成表型資訊的蒐集 (Shafiekhani et al., 2017)。

空中表型分析主要分為衛星影像、無人機影像，兩者已被廣泛運用於田間作物表型分析，與地面表型分析不同的是，航空的表型平台通常被使用在大規模的種植場域，可以分析數千個耕地的冠層溫度與結構 (canopy temperature and structure)、葉綠素含量 (chlorophyll content)、氮含量 (nitrogen content)、植株高度 (plant height) 與地面生物量 (above-ground biomass) (Sankaran et al., 2015)。而與衛星影像相比，無人機在較低的成本下，提供更高的空間解析度，因而更受喜愛 (Yang et al., 2020)。也因此無人機在農業的作物表型應用在過去幾年間以指數的方式飛快的成長 (Barbedo, 2019)。無人機通常會搭載 RGB 相機、紅外線熱像儀、LIDAR、多光譜相機、高光譜傳感器等。RGB 相機最常被使用，其高解析度的 2D 影像可以用來估算作物產量，如 Elazab 等人 (2016) 利用 RGB 影像預測玉米田的產量，並認為 RGB 影像具有足夠的資訊量，不須使用更昂貴的多光譜相機。Zhou 等人 (2017) 則是使用多光譜影像預測水稻產量。Trachsel 等人 (2019) 運用高光譜影像預測穀物產量。



## 2.4.2 作物產量預估

產量估算對農業生產者來說極為重要，生物產量 (biological yield) 直接關係到最終的利潤，現今的精準農業能協助農民分析生物產量，透過表型分析得到作物在生長期間累積的有機物總量。而希望提升作物產量，可在實驗室中設計不同的生長條件，使相同作物處於不同的生長環境，由實驗可知作物在何種生長條件下可以有較佳的表型，最終擴大應用於田間或溫室。相關應用如 He 等人 (2017) 開發多視圖立體成像系統，該系統用以蒐集草莓 360 度的表型數據，能計算草莓高度、長度、寬度、體積、花萼大小、顏色，其結果顯示，系統分析結果與人工評估結果有良好的一致性。Das Choudhury 等人 (2018) 從側面拍攝玉米的 RGB 影像，基於電腦視覺檢測葉子，計算葉子總數與長度，以監測葉子的表型變化。由上述研究結果所示，當前技術可使用電腦視覺分析作物影像，得到的表型資訊可以被用來找尋適合作物生長的環境。產量預估在農業中的重要性，使近幾年的作物產量估算系統逐漸被開發，而機器學習與深度學習皆被應用其中 (Zheng et al., 2021)。

傳統的機器學習方法，如支持向量機 (support vector machine, SVM)、人工神經網絡 (artificial neural network, ANN) 和隨機森林 (random forest, RF)，這類方法需從圖像或數據集中自行提取關鍵特徵，而特徵提取對分類或預測的效果影響甚遠 (Miao and Niu, 2016)。使用機器學習的方法如 Liu 等人 (2015) 先透過影像前處理的方式增強影像特徵，再經由支持向量機預估兩種葡萄的產量，其結果顯示分類的準確率為 87%、召回率為 90%。Maskey 等人 (2019) 使用三種機器學習的方法，預測主成分回歸 (PPCR)、神經網絡 (NN) 和隨機森林 (RF)，搭配與天氣數據預測草莓產量，而與統計方法相比，機器學習有更佳的預測效果。

除了傳統的機器學習之外，深度學習也經常被應用，不同的是深度學習不須提取特徵，即可直接訓練模型，而其缺點為需要更大量的訓練數據集、計算能力以及訓練時間。如 Olenskyj 等人 (2022) 將 150 張影像分至訓練集、驗證集和測試集，輸入 YOLOv5 訓練，該模型可成功偵測葡萄，再計算模型的邊界框，用於產量估算。Kalantar 等人 (2020) 基於 RetinaNet 深度卷積神經網路，使用無人機空拍影像預測甜瓜個體體積並計算其重量，其平均精度 (average precision score) 為 0.92，F1 Score 在 0.9 以上。



### 2.4.3 作物定位與量測

農作物自動化採收與作物的檢測密不可分，作物自動檢測需要先找到作物的位置，而偵測位置後亦須分析作物的表型，用以判斷機器人採摘的標準。Li 等人 (2011) 認為對於基於機器視覺的收割機而言，作物定位與檢測是最基本且重要的資訊，提升定位與檢測的效能，有助於提高作物產量估計 (crop-load estimation) 的準確性。

在作物偵測的研究上學者們面對各式問題，如作物檢測的準確性與穩定性受限於多變的室外光照條件與複雜交錯的枝葉遮蔽問題。Gongal 等人 (2016) 的研究中針對蘋果的自動檢測與採收，開發一種大型傳感器平台應用於取得蘋果樹兩側枝不同角度的影像，用以分析蘋果於 3D 空間的位置，研究證實結合不同角度的圖像資料能將果實計數準確度由 58 % 提升至 82 %。Mehta and Burks (2014) 的研究可透過機器手臂上的單目相機估計橘子的深度，透過果實的像素坐標、大小和相機的內參計算水果的歐氏位置，再透過相機外參將相機坐標轉換到機器人底座座標，其機器手臂採摘的誤差 15 cm 以內。

作物偵測除了使用相對較低成本的 RGB 單目相機，也能使用光達或是雙目立體相機。Feng 等人 (2012) 使用 3D 光達在實驗室中分析假蘋果的數量與大小，從生成的深度圖像中可以分析出水果、樹葉和樹枝的幾何特徵以及相互之間的層次關係，提供了更豐富的信息分析作物表型。另外，將雙目立體相機應用於作物檢測的研究中，分析了作物的大小、距離，平均距離誤差為 5%，在梨子和蘋果的大小估計上，平均直徑誤差在 2% 至 6% 之間 (Font et al., 2014)。

# 第三章 研究方法



## 3.1 自主導航系統架構

本研究之目的為建立一溫室無人機自主巡航系統，利用無人機蒐集的資訊計算溫室中洋香瓜的數量與座標。硬體部分包含 DJI Tello、Intel RealSense D435 深度攝影機、無人機地面站。在無人機自主導航時，ROS 系統自動錄製 Rosbag 紀錄所有資訊，完成飛行後再使用果實分析算法計算洋香瓜數量與座標。

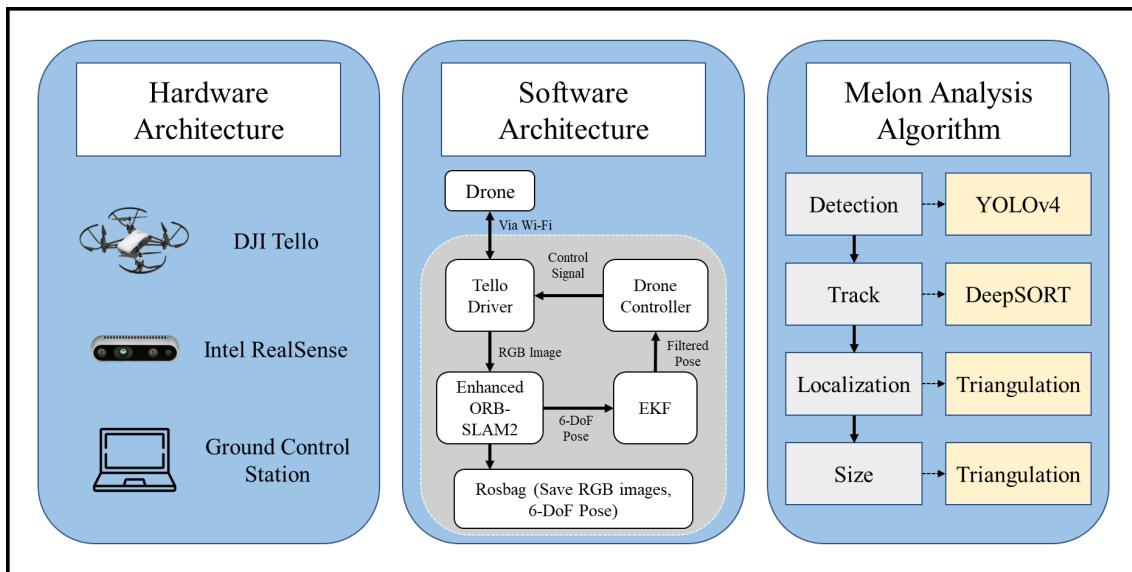


圖 3-1 自主巡航無人機系統架構示意圖

### 3.1.1 硬體架構

本研究選用 DJI Tello 作為實驗中的目標無人機，如圖 3-2 所示。Tello 是一款市售價格低廉的無人機，大疆官方網站上的市售價格為台幣 3500 元，全機含一顆電池，機體搭載 HD720p 30fps 的 RGB 相機。該無人機的重量輕，且機身尺寸小巧，非常適合飛行在狹小的溫室走道。無人機的 ROS Package 可以在 ROS Wiki 上找到 ([http://wiki.ros.org/tello\\_driver](http://wiki.ros.org/tello_driver))，該 ROS Package 基於非官方的 TelloPy 庫建立。相較於官方的 Tello SDK，非官方的 TelloPy 庫提供了更多的功能。此套件可發送指令來控制起飛、降落、遙控飛行操作，或者是收集無人機的 RGB 影像、里程計、IMU 等資訊。



圖 3-2 DJI Tello 無人機

表 3-1 DJI Tello 無人機技術規格表

DJI Tello	
Video quality	HD720P 30fps
Weight (including battery)	80 g
Body size (including propellers)	9.8 cm × 9.25 cm × 4.1 cm
Maximum flight distance	100 m
Maximum flight speed	8 m/s
Maximum flight time	13 min
Field of view	82.6°
Visual odometry (precision hovering)	YES
Stabilizer	NO

Intel RealSense D435 深度攝影機 (9 cm x 2.5 cm x 2.5 cm)，該攝影機的有效測量範圍介於 0.11 至 10 公尺之間。如圖 3-3 (a) 所示，相機由左至右分成四個部分，分別為左紅線相機、紅外線點陣頭影儀、右紅外線相機和 RGB 相機。其中，紅外線點陣頭影儀和紅外線相機被用於紀錄深度，深度計算採用飛時測距 (ToF) 技術。相機解析度方面，RGB 相機最高畫質為 1920 x 720 像素，最高幀數為 60FPS，深度相機最高畫質為 1280 x 720 像素，最高幀數為 300FPS，但相機無法同時選擇最高畫質和最高幀數，需在之間找取平衡。深度相機和 RGB 相機具有不同的視野大

小和角度，但可以通過影像處理使其一致。Intel RealSense 相機官方開發了一個開源的 ROS 套件 (<https://github.com/IntelRealSense/realsense-ros>)，該套件提供深度相機與 RGB 相機的校正功能，可以將校正後的影像傳送至 rostopic，開發者可以直接使用 realsense 相機的影像。在我們的系統中，RealSense 相機的影像用於製作預建構地圖。如圖 3-3 (b) 所示，ORB-SLAM2 的節點 (/orb\_slam2\_rgbd) 只需訂閱兩個 rostopic：/camera/color/image\_raw 和 /camera/aligned\_depth\_to\_color/image\_raw，就可以直接應用於 ORB-SLAM2 演算法。

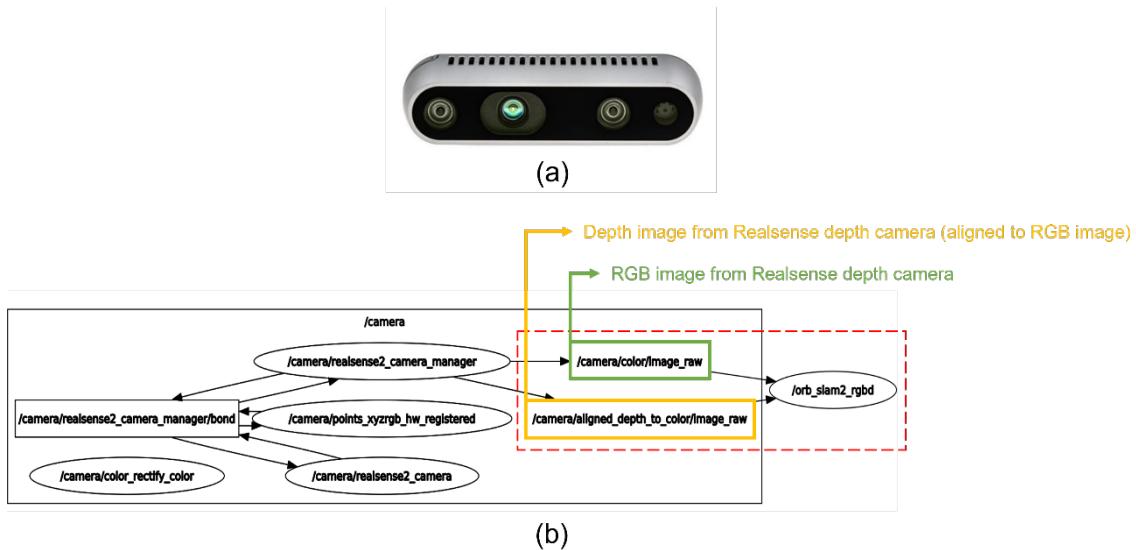


圖 3-3 Intel RealSense D435 深度攝影機：(a)攝影機實體圖；

(b) rqt\_graph 中 rostopic 訂閱關係

無人機地面站選用微星筆記型電腦（型號：MSI GP65 Leopard 9SE-018）作為控制無人機的核心裝置。該筆記型電腦搭載了 Intel i7-9750H 處理器和 8GB DDR4 記憶體、1TB+256GB PCIe SSD 硬碟組合、NVIDIA GeForce RTX 2060 6G。上述硬體規格的組合可確保無人機地面站能有效地控制無人機，處理飛行任務和數據。此外，在筆電上額外安裝 TP-Link Wi-Fi dongle（型號：TL-WN722N V4），以增強地面站與無人機訊號傳輸之穩定度。



### 3.1.2 飛行控制軟體架構

無人機地面站的作業系統為 Ubuntu18.04 LTS 版本，並搭配對應的 Robot Operating System (ROS) Melodic 版本運行。ROS 是一個廣泛使用的開源機器人軟體平台，提供了分佈式架構，使不同模組可以相互通信和協作，包括感知、運動控制、導航和任務規劃等。其模組化設計降低了開發和除錯的難度，成為機器人研究和應用開發的主流選擇。

無人機與地面站溝通流程如圖 3-4 所示，在導航任務運行中，無人機主動透過 Wi-Fi 將即時的 H.264 壓縮影像傳送至地面站。在 ROS 系統下，Tello Driver 解壓縮影像後，將影像發送至 ORB-SLAM2 做無人機定位。ORB-SLAM2 先將 RGB 影像轉換成灰階影像，計算影像中的 ORB 特徵。同時，ORB-SLAM2 啟用重定位模式與預建構地圖進行匹配，一旦重定位成功，將無人機每個 Frame 的位置轉換矩陣 (6-DoF Pose) 發送至 Robot Localization (EKF) 經過 Extended Kalman filter (EKF) 處理。Drone Controller 使用 EKF 濾波後的位置 (filtered pose) 與預先設定的導航路徑計算出位置誤差 (pose error)，最後，PID 控制器計算控制訊號，Tello Driver 將其發送給飛行中的無人機，完成閉迴路控制。此溝通流程允許無人機與地面站之間進行即時的影像傳輸和位置信息的回饋。藉由 ORB-SLAM2 的視覺定位和 EKF 的濾波處理，無人機可以實現更準確的導航和控制。

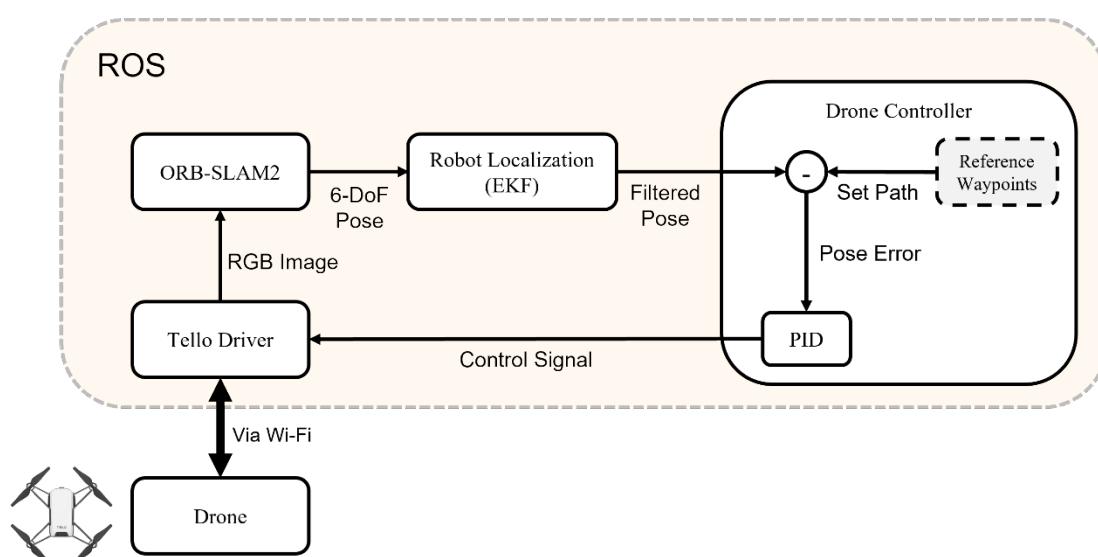


圖 3-4 自主巡航無人機軟體架構圖



## 3.2 ORB-SLAM2

### 3.2.1 Enhanced ORB-SLAM2

本研究中基於 GitHub 作者 CarminLiu 改良的 ORB-SLAM2 做延伸，其專案在 GitHub 上的名稱為 "ORB\_SLAM2\_aruco"。該作者在 ORB-SLAM2 加入了 ArUco Marker 的功能，這使得 ORB-SLAM2 不僅可以使用 Keypoints 進行 SLAM，亦可使用 ArUco Marker 進行 SLAM。作者將 MapAruco 合併至 KeyFrames，初始化地圖可使用 Keypoints 或 ArUco Marker 來完成，也能追蹤和校正 MapPoint 和 MapAruco，並更新全局束平差 (global bundle adjustment)。

我們基於 CarminLiu 的研究，新增了儲存與載入地圖的功能，儲存地圖對於自主飛行至關重要，此功能關乎能否運用地圖做溫室自主巡航。本研究選擇一個 ORB-SLAM2 開源的地圖儲存函式進行修改，由於原始函式僅提供 MapPoint 與 KeyFrames 的儲存，我們將其修改成可以存取 MapAruco，並整合進 CarminLiu 的 ORB\_SLAM2\_aruco 中。完成後，本系統能在 SLAM 建圖後儲存地圖，並在下一次運行時載入地圖，我們稱此系統為 Enhanced ORB-SLAM2。

### 3.2.1 視覺地圖建立

本研究主要運用兩個系統來比較，原始的 ORB-SLAM2 與我們修改的 Enhanced ORB-SLAM2。在預建構地圖的部分，兩個系統略有不同，ORB-SLAM2 系統使用了 Intel RealSense D435 深度攝影機，而 Enhanced ORB-SLAM2 系統則使用了 DJI Tello 的 RGB 相機。SLAM 建圖使用人工手持相機的方式，在溫室走道上行走並錄製影像，該影像使用 ROS 框架下的 rosbag 工具進行儲存。在系統建構地圖時，將錄製的影像以 0.5 倍速度重播.bag 檔案來建立溫室地圖（指令: rosbag play xxx.bag -r 0.5）。透過這種方法，可以降低播放的 FPS，使 SLAM 演算法能建構出更完整的地圖。



### 3.2.2 Enhanced ORB-SLAM2 地圖校正

ORB-SLAM2 在建立地圖的過程中具有一定的隨機性，代表即使使用相同的影像建地圖，也無法獲得完全相同的地圖。這主要是因為系統初始化地圖時需要找到連續兩個符合初始化影像條件的幀 (frame)，而每次計算可能得到不同的初始化幀。而這兩個初始化幀決定了地圖的世界座標，因此世界座標的初始方向無法由使用者自行定義，而是在每次初始化時自動確定。因此，地圖的方向可能與使用者預期的方向不同，需要進行校正。

原始 ORB-SLAM2 的地圖由 MapAruco 和 KeyFrames 組成，本研究的 Enhanced ORB-SLAM2 由 MapPoint、MapAruco、KeyFrames 三者組成。MapAruco 在校正地圖起了關鍵的作用。首先，我們使用捲尺量測溫室內所有 ArUco Marker 的真實位置，接著我們使用這些真實位置校正地圖中有誤差的 MapAruco 位置。

在地圖校正的過程中，我們採用了兩種不同的方法。第一種方法是使用迭代最近點 (iterative closest point, ICP) 算法，使用 Python 的 open3d 函式庫中的剛性 ICP 函數。然而，此方法的缺點是無法利用已知的 ArUco 點對進行匹配，僅通過迭代的方式尋找最低誤差，因此可能只能獲得局部最優解 (local minimum)。

第二種方法是使用仿射轉換 (affine transformation) 來進行地圖校正。在本研究中使用 Python Geometrical Morphing (PyGeM) 函式庫，PyGeM 主要用於幾何變形，它提供了各種常見的幾何變形技術，如形狀插值、形狀曲線和曲面的變形，以及形狀優化算法。其中我們選擇 Radial Basis Function (RBF) 的方法來實現仿射轉換。RBF 是一種在機器學習和計算機圖形學中常用的工具，具有良好的插值特性。RBF 的核心概念是對於輸入空間中的每一點，其輸出值應該是其鄰近點的函數。圖 3-5 為 PyGeM 的官方使用範例，僅需設置原始控制點與變形控制點，程式會計算出映射矩陣，即可用來完成仿射變形。我們將地圖中的 MapAruco 設為原始控制點，而實際量測的 Ground truth ArUco 設為變形控制點，使用映射矩陣校正整張地圖 (MapPoint, MapAruco, KeyFrames)，完成仿射轉換的地圖校正。

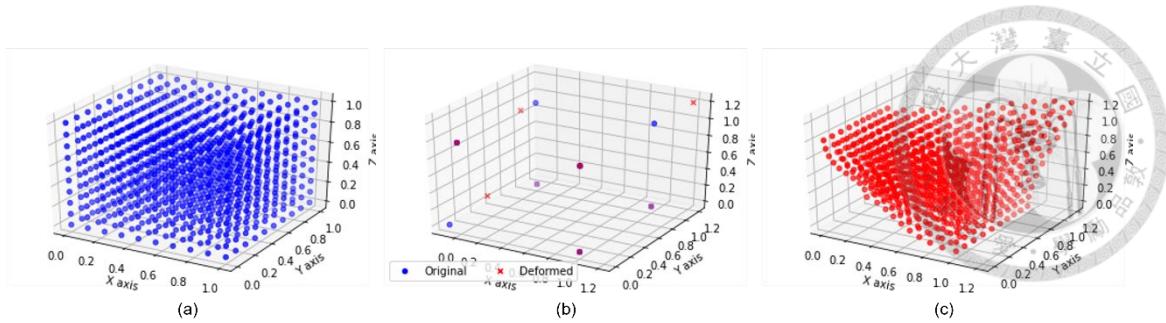


圖 3-5 Pygem 仿射轉換之官方範例：(a)原始點之空間分布圖；

(b)原始控制點與變形控制點；(c)變形點之空間分布圖

### 3.3 飛行控制系統

#### 3.3.1 無人機 3D 位姿估計

本研究使用由 Charles River Analytics, Inc. 開發的 Robot Localization 機器人多傳感器位置估計工具定位無人機位置，Robot Localization 為 ROS 開源套件且被廣泛應用於無人水下航行器 (Unmanned Underwater Vehicles, UUV)、無人地面車輛 (UGV) 和無人飛行器 (UAV)。Robot Localization 基於卡爾曼濾波器 (kalman filter) 的算法所開發，在 ROS 上被完善的維護，除了本研究使用的 ROS1 之外，在 ROS2 架構也有發布對應的版本支援。該套件可於 ROS Wiki 上找到 ([http://wiki.ros.org/robot\\_localization](http://wiki.ros.org/robot_localization))。Robot Localization 主要包含兩個節點 (node) 提供開發者使用，分別為 `ekf_localization_node` 和 `ukf_localization_node`。`ekf_localization_node` 使用擴展卡爾曼濾波器(EKF)；而 `ukf_localization_node` 使用無跡卡爾曼濾波器 (UKF)，兩者皆使用相同的全向運動模型來預測系統的狀態。由於 UKF 需要計算和處理更多的 sigma 點，因此在計算上相對繁重。相比之下，本研究選用相對輕量的 EKF 算法。

EKF 使用全向運動模型預測無人機位置，使用過去的資料預測機器人位姿，並使用持續收到的感測器數據校正模型預測的位置。Robot Localization 提供多傳感器位置估計算法，基於純視覺定位，我們僅使用 ORB-SLAM2 回傳的 6DoF pose 作為 Robot Localization 的輸入資訊，用以降低 SLAM 位姿計算的雜訊。

表 3-2 Robot Localization 以 15 維向量表示機器人運動狀態

State	units
Coordinates in the x-y-z system (robot position)	$x, y, z$
Angles around the x-y-z axes (robot orientation)	$\alpha, \beta, \gamma$
Velocity along the x-y-z axes	$\dot{x}, \dot{y}, \dot{z}$
Angular velocity around the x-y-z axes	$\dot{\alpha}, \dot{\beta}, \dot{\gamma}$
Acceleration along the x-y-z axes	$\ddot{x}, \ddot{y}, \ddot{z}$

### 3.3.2 導航路徑設置與抵達航點之判斷方法

在導航路徑設置中，可以直接設定連續的路徑點 (waypoint)，路徑點包含空間座標 ( $x, y, z$ ) 與角度 (degree)。為了提高控制的準確性，算法會自動對指定的空間座標進行空間內插處理，如圖 3-6 所示。如此可將原本距離較遠的路徑點分割成多個路徑點，如此有助於無人機貼近導航路徑，確認抵達每一個路徑點才往下一個移動，也可以避免 PID 控制過程中指定點距離過遠，導致控制訊號過大的情況，使控制更為穩定。在導航過程中，我們使用位置閥值與角度閥值檢測無人機是否已到達指定路徑點，當兩者皆滿足條件時，算法判定無人機已經到達目標點，並將目標點更新成下一個預定的目標點，以完成自主導航。

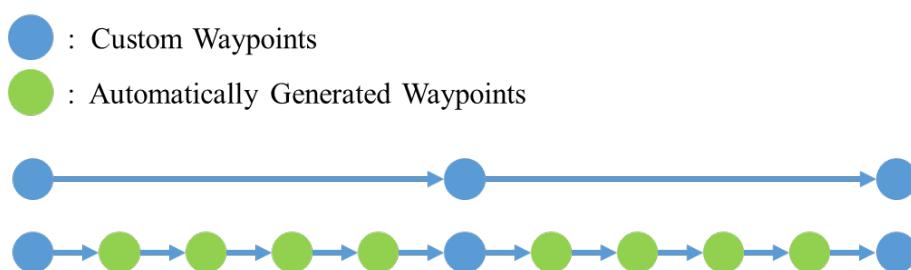


圖 3-6 使用手動設定之航點經由系統自動生成多個細部航點

為了計算機器人目前位置與目標位置之間的距離，我們計算三維空間中目前位置與目標位置之間的歐幾里德距離。利用目前位置和目標位置的 X、Y 和 Z 坐



標，根據歐幾里德距離的公式計算出距離值。歐幾里德距離是計算兩個點之間直線距離的常用方法，它通過對兩個點在每個坐標軸上的差的平方進行平方根運算來獲得距離值。當計算出來的距離小於設定的距離閾值時，判定位置以滿足條件。

除了計算位置的誤差，也需透過角度閾值確認無人機是否與目標方向對齊，系統內所儲存的角度使用四元數 (quaternions) 表示，我們使用 ros 中的 tf 套件 (`tf.transformations.euler_from_quaternion`) 將四元數轉換成歐拉角 (euler angles)，歐拉角是一種常用的姿態表示方法，用於描述物體在三個旋轉軸上的旋轉角度。通過獲取目前位置和目標位置的歐拉角，我們計算它們在 Z 軸上的差異，即偏航角 (yaw)。偏航角表示物體繞著垂直於其平面的軸旋轉的角度。最後，我們將計算得到的偏航角差異與預先設定的 `angle_threshold` 進行比較。如果偏航角差異小於 `angle_threshold`，表示方向已經對齊。

當滿足位置與角度的條件時，即無人機的位置與目標位置之間的距離小於預設位置閾值，且無人機的方向與目標方向之間的差異小於預設角度閾值時，算法將判定無人機已經到達目標點。此時，目標點將被更新為下一個預定的目標點，以實現自主導航。

### 3.3.3 無人機控制系統與控制閥設計

除了導航路徑設置外，PID 控制也是飛行控制系統的重要部分。PID 控制是一種常用的閉迴路控制方法，用於根據無人機當前狀態與目標狀態之間的誤差計算出適當的控制指令。在本研究中，我們使用 ROS 的開源 PID 控制套件，控制無人機的三個角度 (roll, pitch, yaw) 與高度 (altitude)，透過 PID Node (節點) 訂閱無人機的 EKF 位置以及事先設定的導航路徑點。根據位置誤差計算出相應的控制指令，從而實現閉迴路控制。

使用 EKF 是為了降低 ORB-SLAM2 在位置估計上的雜訊，使控制上能更加穩定。在 ORB-SLAM2 失去追蹤時，ORB-SLAM2 即停止更新 6DoF pose，但 EKF 會使用過去的 6DoF pose 資料持續估計無人機的位置，這使得控制變得不穩定。因此我們在這個部分加入了一個控制閥，如圖 3-7 所示，橢圓框代表節點 (ros

node)，方框代表主題(ros topic)，orb\_slam2\_mono 發布位姿資訊，由 orb\_check 訂閱，在 orb\_check 這個 node 中會確認位置是否有更新，若 ORB-SLAM2 有更新位置，發布布林值 True，反之發布 False。Waypoints\_mission 發布的是 PID 計算後的控制訊號，會將此訊號經由 tello/cmd\_vel\_relay 傳給 cmd\_relay。最終 cmd\_relay 會接收到 ORB-SLAM2 有無更新的布林值以及控制訊號，當布林值為 True 時，發送控制訊號給/tello/cmd\_vel，若布林值為 False 則不發送控制訊號。

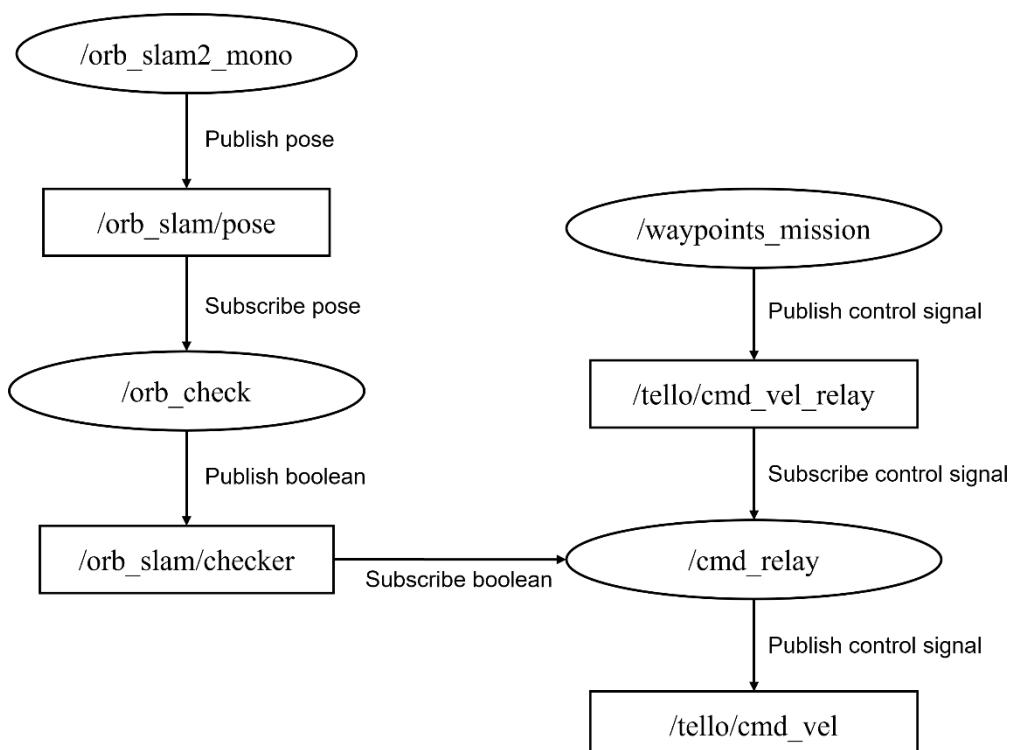


圖 3-7 ORB-SLAM2 位置更新與無人機控制訊號之控制閥架構

## 3.4 果實資訊分析

### 3.4.1 果實追蹤演算法

本研究使用名為"yolov4-deepsort" 的開源專案，該專案由 "theAIGuysCode" 團隊在 GitHub 上發布。此專案的主要貢獻在於將 YOLOv4 的權重從 Darknet 架構轉換為 TensorFlow 架構，以便在 DeepSORT 上執行。本研究基於此專案修改，儲存每個 frame 的 ID、果實框位置與果實 ID，並調整原始輸入資料格式。原始專案使



用的輸入格式是 MP4 影片，而我們將其修改為連續的 PNG 圖像序列。此修改方便直接使用 ORB-SLAM2 輸出的圖像資料進行追蹤，同時每張影像都可以進行分析和檢查，有助於除錯工作。

執行果實追蹤演算法除了輸入連續影像，也需要準備果實偵測的 YOLO 模型。研究人員手持無人機在溫室走道上行走，收集了共計 211 張圖像，這些圖像依照 7:2:1 的比例分為訓練集、驗證集和測試集，並使用 LabelImg 標記圖像中的果實框。



圖 3-8 YOLOv4 果實偵測模型之 RGB 訓練影像

在果實追蹤演算法執行後，將果實追蹤結果儲存在 txt 文字檔，如圖 3-9 所示。該文字檔每個 row 代表一個果實框的資訊，包括 frame\_ID、x1、y1、x2、y2 和 melon\_ID。其中，frame\_ID 表示該果實在哪個 frame 被偵測到， $(x_1, y_1)$  和  $(x_2, y_2)$  分別代表果實框的左上角和右下角座標，melon\_ID 則是該果實的追蹤 ID。

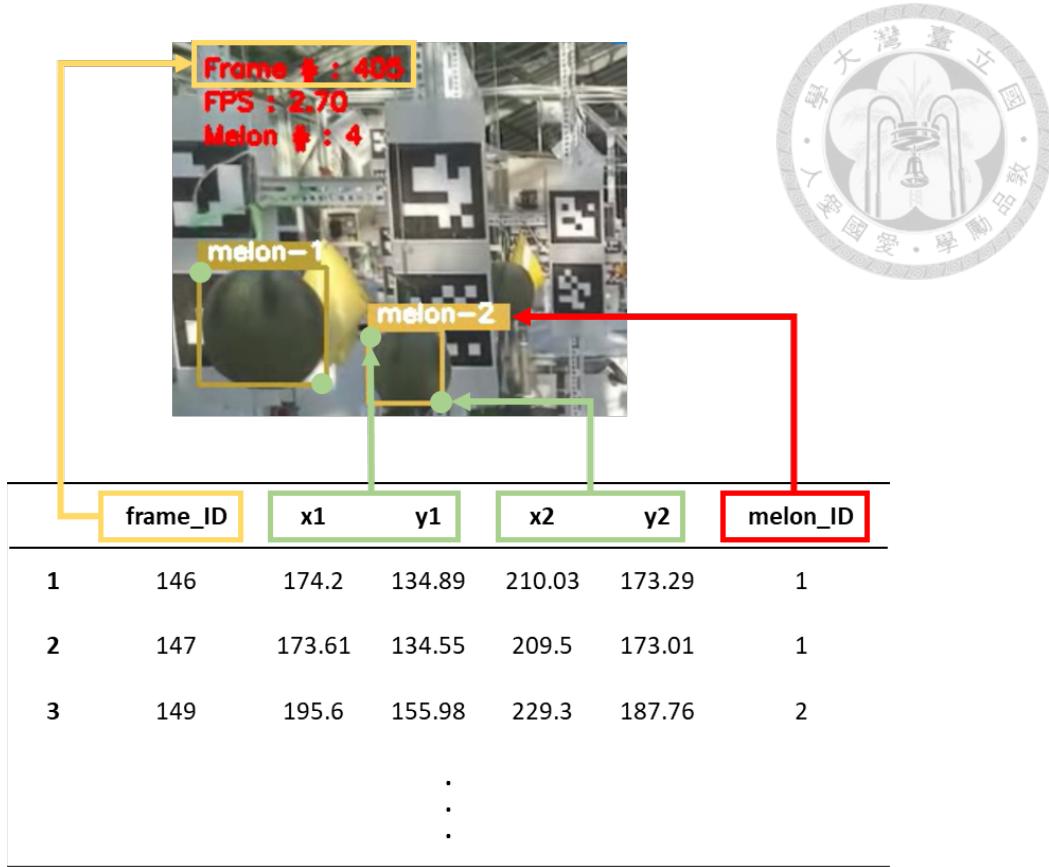


圖 3-9 果實追蹤結果儲存之欄位與意義示意圖

我們將此追蹤結果分三個步驟依序做資料清理，第一步，刪除果實框長或寬小於閾值 (size threshold)的該"筆"資料（刪除單一 row），降低果實距離相機較遠時造成誤判的可能性。第二步，刪除果實 ID 出現次數少於閾值 (frame threshold) 的該"顆"果實所有資料（melon\_ID 為該顆果實的所有 row 都刪除），降低少量的錯誤偵測的影響。第三步，此步驟基於果實不會重複出現所設計，找出 ID switch 的方法如下，當該果實 ID 已碰觸到影像左右邊界後，該果實 ID 又重複出現在後續的影像中，且果實位置不在邊界，即判定為 ID switch，並給予重複出現的果實新的 ID。透過資料清理，可以篩選出追蹤結果較可靠和準確的果實資訊，有助於提升果實追蹤演算法結果的可靠性，使得後續的果實定位更具應用價值。



### 3.4.2 果實定位演算法

果實定位演算法使用三角測量，三角測量利用三角形的幾何關係來確定物體的位置，是一種常用於測量空間中距離、角度和高度的技術。使用三角測量的果實定位算法，需要無人機的相機內參、Enhanced ORB-SLAM2 回傳的相機外參（旋轉矩陣、位移矩陣）以及果實追蹤結果。

相機內參如公式 3-1 所示，包括相機焦距 (focal length)、主點坐標 (principal point)，相機內參可由相機標定技術取得。本研究使用 9x7 的黑白棋盤格作為標定板，將棋盤格列印成 A4 大小，棋盤格的邊長皆為 2.65 cm，如圖 3-10 (a) 所示。相機拍攝標定板時，使棋盤格填滿相機畫面，並以不同方向與角度拍攝。取得多張棋盤格影像後，對每張圖像使用角點檢測算法 (`cv2.findChessboardCorners()`) 檢測標定板上的角點，接著利用檢測到的角點坐標和標定板的實際尺寸，使用相機標定算法 (`cv2.calibrateCamera()`)，即可得到相機內參，圖 3-10 (b) 為檢測到角點的結果圖。

$$K = \begin{bmatrix} f_x & 0 & u_0 \\ 0 & f_y & v_0 \\ 0 & 0 & 1 \end{bmatrix} \quad (3-1)$$

其中  $f_x$  與  $f_y$  分別水平方向與垂直方向的焦距像素尺度， $u_0$  與  $v_0$  分別代表圖像平面上主點的水平和垂直像素坐標。

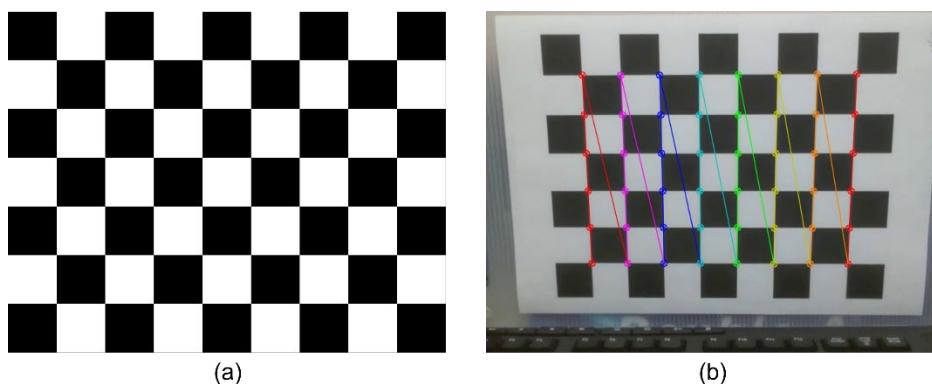


圖 3-10 相機內參使用之棋盤格標定板：(a)9x7 棋盤格；(b)棋盤格角點檢測結果



相機外參由旋轉矩陣與位移矩陣組成，用於描述相機在世界坐標系中的姿態與位置，Enhanced ORB-SLAM2 運行時，會即時回傳每個幀的相機外參，此相機外參在本節用來計算果實位置。將每張影像的相機外參以齊次矩陣 (homogeneous matrix) 儲存至 txt 文字檔，如式 3-2。

$$T = \begin{bmatrix} r_{11} & r_{12} & r_{13} & t_1 \\ r_{21} & r_{22} & r_{23} & t_2 \\ r_{31} & r_{32} & r_{33} & t_3 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (3-2)$$

3.5.1 節已詳述果實追蹤方法，本節使用追蹤結果作為三角測量的輸入，可由追蹤結果得到每顆果實在影像中的果實框 ( $x_1, y_1, x_2, y_2$ )，從果實框的資訊亦可求得果實中心點 ( $x_{center}, y_{center}$ ) 如下式：

$$x_{center} = \frac{x_1 + x_2}{2} \quad (3-3)$$

$$y_{center} = \frac{y_1 + y_2}{2} \quad (3-4)$$

以相機內參、相機外參與果實中心點作為輸入，使用 OpenCV 的三角測量函式 (cv2.triangulatePoints()) 計算果實位置。選用連續兩張影像做三角測量（如 frame 1 與 frame 2），當該果實在連續影像中出現在  $i$  個 frame 中，算法會運行  $i-1$  次三角測量，最後由  $i-1$  組果實位置取中位數，得到中位數的果實位置後，使用 3.3.2 使用 ArUco Marker 做仿射轉換地圖校正求得的轉換矩陣，校正此果實位置。

上述說明果實在溫室的座標位置可由三角測量後再使用轉換矩陣求得，但分析後發現三角測量的果實位置與真實果實位置存在常數誤差，為了解決此問題，我們使用 ArUco Marker 尋找常數誤差。我們在運行 Enhanced ORB-SLAM2 時，將偵測到的 ArUco Marker 儲存至 txt 文字檔 (frame\_ID, x1, y1, x2, y2, aruco\_ID)，此文字檔的格式與 DeepSORT 儲存的果實框格式相同，差別僅在於第五個 column 由 melon\_ID 變成 aruco\_ID。儲存相同格式的優點是，可以直接使用果實追蹤的資料清理算法，同樣設置資料清理第一步驟 size threshold 與第二步驟 frame threshold，可以濾除較遠的 ArUco Marker，但這裡不須執行第三步驟的資料清理，因為



ArUco Marker 的 ID 是可以確定的，不會有 ID switch 的問題產生。果實框與 ArUco Marker 框同樣做資料清理可以使三角測量的資料更為相近，由此 ArUco Marker 框計算出的校正常數才能將果實位置校正得更準確。

校正方法如下，使用 ArUco Marker 位置框的中心點做三角測量，使用三角測量計算的 ArUco Marker 位置與 ArUco Marker 真實位置，分別計算出空間中三個方向的位置均方根誤差 (RMSE)，分別為  $(\delta_x, \delta_y, \delta_z)$ ，初次計算的誤差用來計算初始迭代參數的範圍 (upper and lower bounds)，計算公式如下。

$$Upper bound = 2 \times |\delta| \quad (3-5)$$

$$Lower bound = -Upper bound \quad (3-6)$$

從第一次的三角測量結果與真實位置可求得  $\delta$ ，再使用公式 3-5 與 3-6 可得到 Upper bound 與 Lower bound，由公式 3-7、3-8、3-9 求得 Midpoint、Lower Mid、Mid Upper 三個參數後後，將 Lower Mid、Midpoint、Mid Upper 設為校正常數，依次帶入計算後可求得三組 RMSE。依照三個 RMSE 的結果，找最小的 RMSE 迭代限縮 Upper bound 與 Lower bound，三種 RMSE 結果的調整方式如表格 3-2 所示，自動尋找校正常數之範例如圖 3-11 所示。

$$Midpoint = \frac{Upper bound + Lower bound}{2} \quad (3-7)$$

$$Lower Mid = \frac{Lower bound + Midpoint}{2} \quad (3-8)$$

$$Mid Upper = \frac{Midpoint + Upper bound}{2} \quad (3-9)$$

表 3-3 Upper and lower bounds 迭代運行調整邏輯

Parameter for obtaining the minimum RMSE	Upper and lower bounds adjustment method
Lower Mid	Lower bound remains unchanged. New Upper bound is set to the old Midpoint.
Midpoint	New Lower bound is set to the old Lower Mid. New Upper bound is set to the old Mid Upper.
Mid Upper	New Lower bound is set to the old Midpoint. Upper bound remains unchanged.

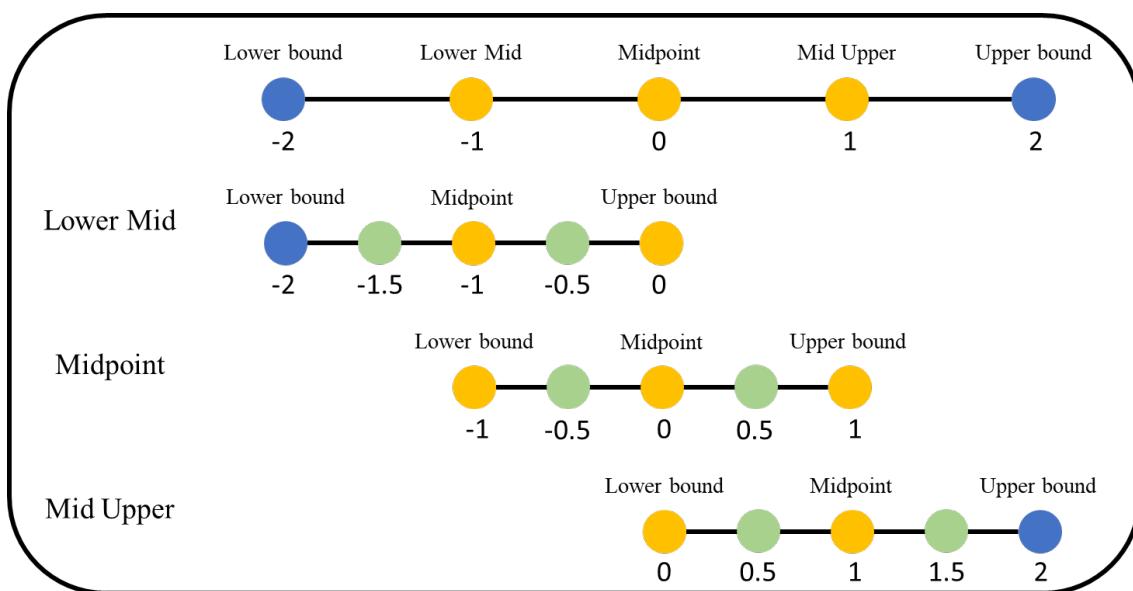


圖 3-11 ArUco 校正常數自動迭代求導之範例圖

使用該方法迭代 10 次，以獲得每個軸向的校正常數。在得到三個軸向的校正常數後，我們可以使用常數校正三角測量果實位置的結果。其中 x 方向的校正較為特殊，需判斷果實位於無人機的右側或左側，如果果實位於無人機的右側，則我們將果實位置加上 x 方向的校正常數；如果果實位於無人機的左側，則我們將果實位置減去 x 方向的校正常數。判斷果實在左右側的方法由三角測量結果可求得，詳細方法敘述於 4.4.2 的果實定位章節。



### 3.4.2 果實大小演算法

在三角測量果實位置時，算法會同步計算果實的大小。我們使用三個資訊計算果實在真實世界中的大小，包含：相機內參、果實與無人機的相對距離（由三角測量果實定位求得）、影像中果實方框大小。使用以下公式將影像中的寬度 (w) 與高度 (h)，換算成真實世界的寬度 ( $W_{real}$ ) 與高度 ( $H_{real}$ )。

公式 3-10 與 3-11 中，使用相機內參的  $f_x$  與  $f_y$ ，以及像素大小參數  $size_{pixel}$ ，來換算影像中的果實方框寬度和高度。

$$W_m = \frac{w * size_{pixel}}{f_x} \quad (3-10)$$

$$W_h = \frac{h * size_{pixel}}{f_y} \quad (3-11)$$

接著，我們需要考慮影像中的透視效果。為此，我們計算了物體和相機之間的距離 D，透過公式 3-12、3-13 可推導出了水平視場角 ( $FOV_x$ ) 和垂直視場角 ( $FOV_y$ )，單位均為弧度。

$$FOV_x = 2 * \arctan\left(\frac{W_m}{2 * D}\right) \quad (3-12)$$

$$FOV_y = 2 * \arctan\left(\frac{H_m}{2 * D}\right) \quad (3-13)$$

最後，利用上述計算得到的水平和垂直視場角，使用公式 3-14 與 3-15 可求得果實的真實寬度和高度。

$$W_{real} = 2 * D * \tan\left(\frac{FOV_x}{2}\right) \quad (3-14)$$

$$H_{real} = 2 * D * \tan\left(\frac{FOV_y}{2}\right) \quad (3-15)$$

三角測量會得到多組果實位置，在果實大小亦相同。每顆果實得到的寬度與高度大小會存在 list 當中，最後對寬度與高度分別取中位數，再將果實的寬度與高度相加除以 2，視為果實的直徑。



## 3.5 實驗設置

### 3.5.1 實驗場域

本研究在國立臺灣大學農業試驗場的玻璃溫室進行，如圖 3-12 (a)所示，該溫室的場域實際尺寸為長 30.15 公尺、寬 8.15 公尺，溫室內設有 12 組角鋼架，每組長 5.4 公尺、寬 0.75 公尺、高 2.1 公尺。溫室內種植洋香瓜，品種為夏系 2 號日本阿露斯，一組角鋼架可種植 5 個植株，每季度（三個月）的最大產量為 60 顆果實。此外，兩組角鋼架之間的走道寬度為 1.16 公尺，寬敞的走道適合作為無人機飛行實驗場域，如圖 3-12 (c)所示。

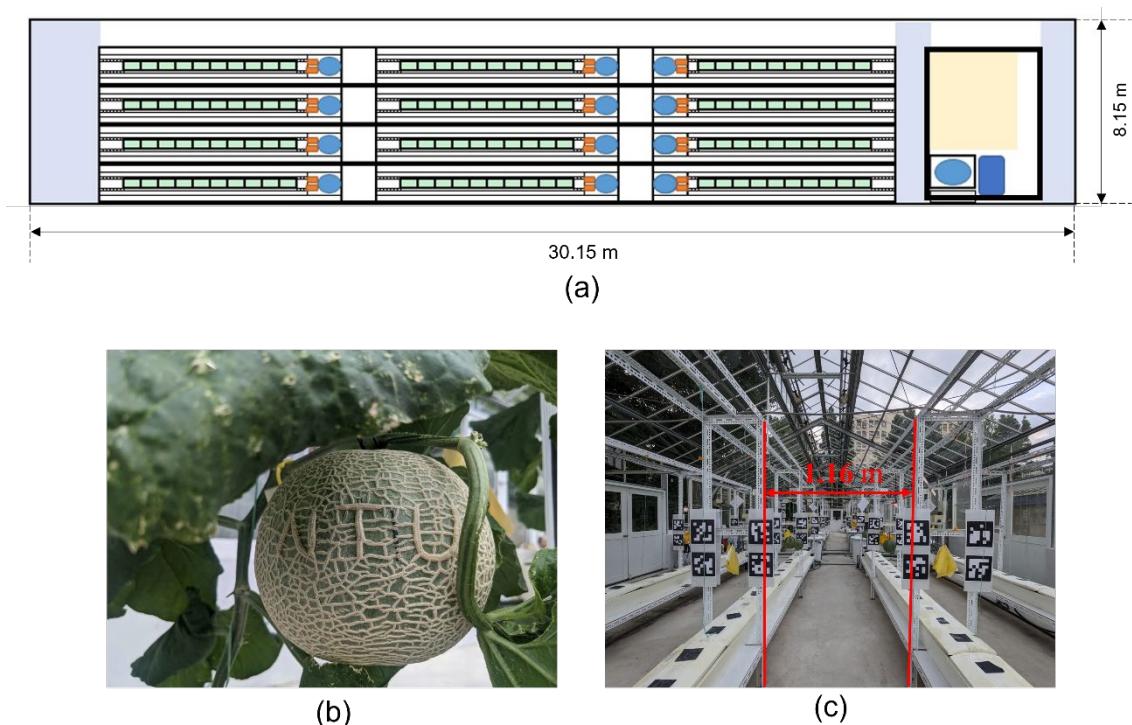


圖 3-12 國立臺灣大學農業試驗場：(a)溫室上視簡圖；  
(b)夏系 2 號日本阿露斯洋香瓜；(c)溫室走道實體圖



### 3.5.2 ArUco Marker 之規格與製作方法

本研究的 Enhanced ORB-SLAM2 演算法使用 ArUco Marker 加強定位效果，我們選擇了 aruco\_mip\_36h12 此標記型式，總共有 250 個 Marker，每個 Marker 由 8x8 的黑白方塊組成，並且具有唯一的 Marker ID。

如圖 3-13 (a)，我們將 Marker 列印在 A4 紙上，並裁切成 19.5 公分 x 19.5 公分的正方形，留下約 0.5 公分的白邊。如圖 3-13 (b)我們使用透明瓦楞板 (20 cm x 60 cm)，將兩個 Marker 用透明膠帶固定在瓦楞板上，並在瓦楞板的上下排各鑽三個直徑為 0.4 公分的孔洞。孔洞是為了將瓦楞板固定在溫室的角剛上，透明束帶使用兩種尺寸，第一種是 14 公分 x 0.25 公分的束帶，用來將瓦楞板安裝到角鋼上，第二種是 25 公分 x 0.25 公分的束帶，用在角落點，使瓦楞板與角鋼貼齊。

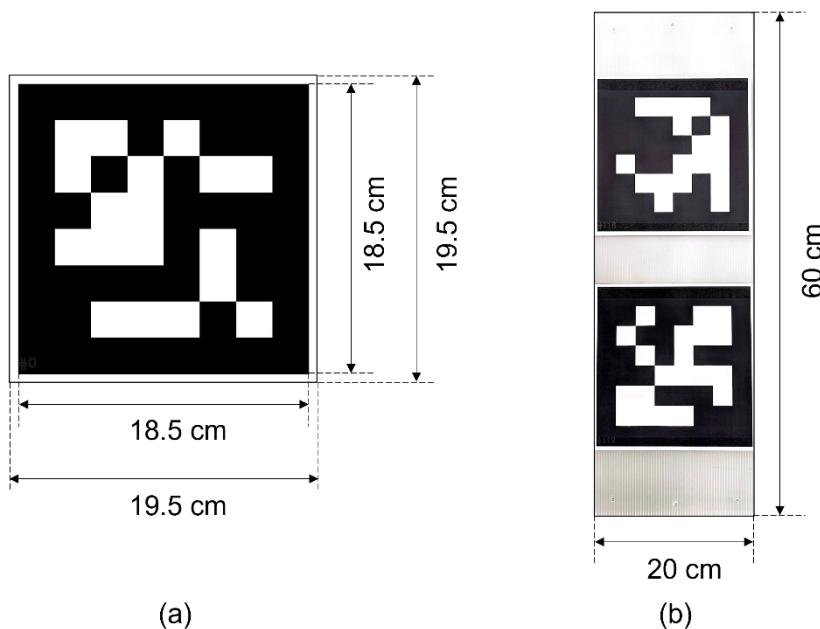


圖 3-13 ArUco 樣式與尺寸：(a)ArUco Marker 尺寸；(b)ArUco Marker 板尺寸

### 3.5.3 實驗設計與目的

本研究的實驗可拆分成四大主軸，分別為：溫室環境光照量對 SLAM 算法之影響、地圖校正結果分析、飛行控制實驗、果實算法準確度分析（追蹤、定位、大小）。

在 4.1 節為溫室環境光照量對 VSLAM 算法影響之實驗，在不同光照量的溫室中蒐集影像資料，供 VSLAM 系統之定位效果作分析。圖 3-14 為實驗中使用之 HA-5010A 數字照度計，其測量範圍於 0 到 200,000 Lux 之間，並且可量測光照之最大值與最小值。實驗中我們手持照度計，以照度計量測之最大值視為本實驗的光照條件。此外，實驗又分成兩種條件，陰天光照均勻之溫室、晴天存在陰影特徵之溫室。本實驗可解析光照度與陰影分別對於 VSLAM 定位穩定性之影響。



圖 3-14 數字照度計（型號: HA-5010A）

在 4.2 節中我們將依序分析，兩種地圖校正之結果（ICP 與仿射轉換）、原始地圖與校正地圖之差異性，最終呈現 ORB-SLAM2 以及 Enhanced ORB-SLAM2 所建立之地圖。

無人機控制實驗於 4.3 節分析，針對 SLAM 系統與飛行軌跡皆分成三種作分析，因此有九種組合。SLAM 系統先分為 ORB-SLAM2 與 Enhanced ORB-SLAM2，在 ORB-SLAM2 又依溫室有無架設 ArUco Marker 作比較，目的為了比較三種情況：第一，在使用無 ArUco Marker 這類強烈的人工特徵下，使用純 ORB 特徵定位的 ORB-SLAM2 的定位效果；第二，加入 ArUco Marker，使用純 ORB 特徵定位的 ORB-SLAM2 的定位效果；最終，加入 ArUco Marker 人工特徵，使用 Enhanced ORB-SLAM2（可偵測 ArUco Marker 之系統）。由一與二可比較 ArUco Marker 的影響，另外從二與三的結果，比較能偵測 ArUco Marker 是否能增加定位穩定性。

最後是本論文的果實算法分析實驗，從 YOLO 果實偵測模型效能開始分析，接著講述使用此果實偵測模型做果實追蹤之結果，並比較三步驟資料清理對於追蹤結果的影響。找出在第一步驟與第二步驟中，能將 ID switch 數量降至最低的兩個閾值，分析第三步驟有效性的實驗，確認了此方法能有效降低新果實被判定為舊果實 ID 的狀況。果實定位實驗基於果實追蹤結果延伸，使用三角測量計算出果實相對於相機的位置後，將果實位置從相機坐標系轉換至世界座標系，即得到果實位置，接著分析使用 ArUco Marker 常數校正與地圖校正後的果實位置如何讓果實位置誤差逐漸下降，提升定位準確度。在果實大小中應用了果實定位的結果，因此已知果實位置與相機位置的條件下，依照果實框在影像中的大小，即可求出果實的大小，以果實框長寬的平均，視為果實的直徑。如圖 3-15 所示，果實的位置以及直徑大小的真實數值皆由人工使用捲尺量測，而因果實直徑難以直接使用捲尺量測，因此我們量測果實周長後，再將其換算成直徑作為參考數值。



圖 3-15 捲尺：(a)量測果實位置之鋼捲尺；(b)量測果實圓周之軟捲尺



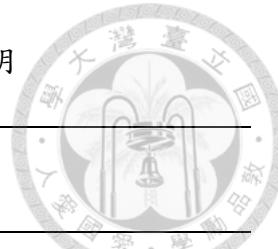
### 3.6 程式檔案與操作流程

本研究以 ROS 為基礎，融合無人機控制和建圖技術，成功實現了在溫室環境中進行自主導航的功能。這項研究旨在開發一個系統，使無人機能夠在受限的溫室環境中精確導航，透過整合多種程式檔案和相關操作步驟，達成自主導航的目標。表 3-4 為自主導航系統在 ROS 架構下撰寫的 Python 檔案與 launch 檔案，表 3-5 為果實演算法與軌跡可視化之 Python 檔案。

表 3-4 無人機自主導航 ROS 系統之程式說明

程式檔名(軟體功能)	運行說明與備註
<b><i>rosrun tello_driver 20230614_Tcw_ekf_wps.py</i></b>	運行 rosbag file 可以儲存無人機飛行的 tcw (SLAM 得到的位置)、EKF (過濾後的位置)、wps (航點位置)，將三個資訊儲存至 txt 文字檔，再使用 <b>3D 空間軌跡圖與誤差圖.py</b> 分析飛行軌跡結果。
<b><i>roslaunch orb_aruco_addArucoMap mono_tello_aruco_706_534_buildmap.launch</i></b>	建 Enhanced ORB-SLAM2 之地圖，運行開啟一張新地圖，結束後將地圖儲存於 .ros 資料夾內，地圖名稱為 Map_ORB_aruco.bin
<b><i>roslaunch orb_aruco_addArucoMap mono_tello_aruco_706_534.launch</i></b>	Enhanced ORB-SLAM2 載入 .ros 資料夾中的 Map_ORB_aruco.bin，使用地圖開啟，啟用定位模式，結束後不儲存地圖。
<b><i>rosrun tello_driver 20230729_show_xyz_xyztw.py</i></b>  給 <b>tcw_print</b> 出加入航點的格式.py	在建圖後為了設定無人機之航點，重複播放建圖的 rosbag 可在 terminal 中顯示當前無人機空間座標與四元數 (旋轉)。
<b><i>rosrun drone_controller moma_mission.py</i></b>	使用 <b>20230729_show_xyz_xyztw.py</b> 儲存所有航點後，使用此程式將空間位置與四元數轉換成 <b>moma_mission.py</b> 需要的格式。
<b><i>rosrun tello_driver orb_check.py</i></b>	運行前在此程式內加入航點 (waypoint)，使用 <b>給 tcw_print 出加入航點的格式.py</b> 將航點設定完成。
<b><i>rosrun tello_driver cmd_relay.py</i></b>	修改抵達航點的判斷依據：原始僅判斷 x 與 y 距離是否小於距離閥值，更新後會判斷 x, y, z 距離以及旋轉角度都小於閥值，才會更新下一個航點。
<b><i>rosrun teleop_twist_keyboard teleop_twist_keyboard.py</i></b>	確認 SLAM 是否成功定位，發送布林值，若有成功定位發送 1，反之發送 0。
<b><i>rosrun tello_driver savevideo.py</i></b>	在 tello 連接狀態下，可即時顯示影像並在關閉程式時儲存影片。

表 3-5 果實追蹤、定位、大小演算法之程式說明



檔名(軟體功能)	使用說明與備註
<b>1_Maparuco 排序.py</b>	<p>功能: Enhanced ORB-SLAM2 取出的 maparuco 的 3D 位置會儲存於 txt 文字檔，依照 ArUco Marker ID 重新排序此 txt 檔案，並儲存至新的 txt 文字檔。</p>
<b>2_迭代找 aruco marker 校正常數.py</b>	<p>功能: 找出計算果實位置所需的 ArUco Marker 校正常數 方法: 從 Enhanced ORB-SLAM2 取出 maparuco 在所有 2D 影像中的 xy 像素座標中心位置，使用與計算果實位置相同的三角測量方法，計算出 ArUco Marker 的位置，再透過迭代找出一組參數(x, y, z)，使三角測量位置與真實 aruco marker 位置之 RMSE 最小化，此參數被應用於 <b>3_果實位置.py</b> 與 <b>4_果實大小.py</b></p>
<b>3_果實位置.py</b>	<p>使用三角測量計算出果實位置後，再使用常數校正與地圖校正降低果實位置誤差。 備註: 建立環境要選擇 Conda，因 pygem 套件無法由 pychram 的 Virtualenv 建立，本研究使用的 pygem 版本為 2.0.0。</p>
<b>4_果實大小.py</b>	<p>(由 <b>3_果實位置.py</b> 修改) 修改 func_find_melon_pos，使其能輸出果實大小。</p>
<b>5_資料清除後果實追蹤結果可視化.py</b>	<p>(由 <b>3_果實位置.py</b> 修改) 功能: 輸出經濾除後之果實追蹤結果，每個果實框使用不同顏色表示(方便人工判斷有無 IDS)，果實追蹤 ID 顯示於果實框左上角，將執行結果儲存至 avi 影片檔。 備註: 若需計算果實位置誤差，需先執行此程式，並用人工方式記錄所有果實被追蹤的 ID，才依照 ID 配對真實果實位置，計算果實位置誤差。(因每次追蹤結果得到的果實 ID 會不同)</p>
<b>3D 空間軌跡圖與誤差圖.py</b>	用於製作圖 4-6 之 3D 空間軌跡圖與誤差圖
<b>追蹤分析.py</b>	分析 ORB-SLAM2 與 Enhanced ORB-SLAM2 紀錄飛行狀態之 txt 文字檔，分析 Keypoints per frame 與 Tracking rate 如表 4-1 所示。



## 第四章 結果與討論

### 4.1 Enhanced ORB-SLAM2 定位分析

在本節中，我們將分析 ORB-SLAM2 和 Enhanced ORB-SLAM2 在不同溫室光源下的定位效果。本實驗蒐集溫室一天的資料作分析，由於溫室的東邊有高架橋遮擋陽光，在九點半前溫室會同時存在陰影以及非陰影區，因此實驗選擇在早上九點半至下午五點半日落後這段時間蒐集資料，每隔半小時以手持無人機方式蒐集，並同步紀錄當前溫室之最大光度。資料蒐集當天早上是晴天，在無雲遮擋陽光的情況下，溫室具備大量陰影特徵，而下午由於夏季午後熱對流旺盛，逐漸有雲朵遮蔽太陽光，使陰影消失，因此本實驗使用溫室一整天之資料，分析在不同光度與陰影條件下之兩種 VSLAM 之定位效果。而定位效果的判定以三個指標為依據，Keypoints per frame（每個幀平均特徵點追蹤數量）、Tracking rate（追蹤率）以及 Frames required for localization（定位所需幀數），Frames required for localization 是無人機能否成功並執行導航任務的關鍵，若無法在起飛後及時成功定位，將無法執行後續的導航任務。

#### 4.1.1 ORB-SLAM2 定位效果分析

表 4-1 呈現了 ORB-SLAM2 在溫室不同光度下的定位表現，早上 9:30 ~ 10:30 這三筆資料的光度都超過 11 萬，並且存在強烈地陰影，Keypoints per frame 約莫落在 300 至 350 之間，Tracking rate 最低至 92.21 %，而 Frames required for localization 分別需要 78, 90, 387 個 frame，代表在地圖初始化重新定位的效能差，無人機起飛後會無法定位成功定位，將無法執行導航任務。

在 11:00 ~ 12:00 時溫室陰影消失，環境光線對比下降，光度在 5~10 萬之間，Keypoints per frame 從原本的 300 提升至 500，Tracking rate 皆在 99.7 % 以上，且都僅需兩個 frame 即可成功重新定位，可見陰影對於純 ORB-SLAM2 的影像甚大。綜觀表 4-1 的結果，從 Shadow 與 Keypoints per frame 這兩個列去分析，可以觀察到當陰影出現時，追蹤的特徵點數量就會明顯地跟著下降。

表 4-1 不同光照環境之 ORB-SLAM2 定位效果

Time	Lux (k)	Shadow	Keypoints per frame	Tracking rate (%)	Frames required for localization
09:30	113.1	YES	340.98	98.7	78
10:00	144.7	YES	320.22	98.4	90
10:30	160.6	YES	314.76	92.21	387
11:00	48.2	NO	519.21	99.98	2
11:30	62.9	NO	488.08	99.79	2
12:00	95.0	NO	555.34	99.97	2
12:30	112.9	YES	528.08	99.98	2
13:00	25.8	NO	600.85	99.98	2
13:30	39.9	NO	605.38	99.98	2
14:00	30.4	NO	636.10	99.98	2
14:30	117.6	YES	473.21	99.95	3
15:00	95.4	YES	475.62	99.9	2
15:30	79.0	YES	471.20	99.98	2
16:00	48.1	YES	548.61	99.98	2
16:30	4.8	NO	672.57	99.98	2
17:00	1.8	NO	663.49	99.98	2
17:30	1.0	NO	754.65	99.98	2
Mean			397	97.82	111.80
Std. Dev.			88	2.87	142.45



### 4.1.2 Enhanced ORB-SLAM2 定位效果分析

表 4-1 與表 4-2 使用的資料皆相同，差別在於使用不同的 VSLAM 系統，表 4-2 分析的是 Enhanced ORB-SLAM2 在溫室不同光照度下的定位表現，實驗中的結果與表 4-1 呈現相同趨勢，唯一不同的在於 Frames required for localization，當光照度達 16 萬且存在陰影時，Frames required for localization 需要 24 個 frame 成功重新定位，其他的光照條件下僅需 3 個 frame 以內即可定位成功，這說明了 Enhanced ORB-SLAM2 在不同光照條件下有相對穩定的效果。

表 4-2 不同光照環境之 Enhanced ORB-SLAM2 定位效果

Time	Lux (k)	Shadow	Keypoints per frame	Tracking rate (%)	Frames required for localization
09:30	113.1	YES	322.78	99.96	3
10:00	144.7	YES	315.35	99.95	3
10:30	160.6	YES	324.57	99.42	24
11:00	48.2	NO	562.94	99.95	3
11:30	62.9	NO	539.65	99.55	3
12:00	95.0	NO	589.65	99.9	3
12:30	112.9	YES	566.40	99.94	3
13:00	25.8	NO	624.82	99.97	2
13:30	39.9	NO	627.47	99.97	2
14:00	30.4	NO	658.54	99.97	2
14:30	117.6	YES	494.86	99.94	3
15:00	95.4	YES	500.56	99.94	3
15:30	79.0	YES	485.57	99.95	3
16:00	48.1	YES	574.77	99.97	2
16:30	4.8	NO	681.91	99.97	2
17:00	1.8	NO	667.48	99.98	2
17:30	1.0	NO	755.26	99.98	2
Mean			413	99.77	7.20
Std. Dev.			113	0.23	8.40

### 4.1.3 ORB-SLAM2 與 Enhanced ORB-SLAM2 定位效果比較

在 4.1.1 與 4.1.2 中分析了兩個系統在不同光照條件下的定位穩定度，證實了新增偵測 ArUco Marker 的 Enhanced ORB-SLAM2 的定位穩定性會優於純 ORB 特徵的 ORB-SLAM2。

從表 4-1 與 4-2 可分析出當溫室存在陰影時，兩個系統的可供作光照度範圍。ORB-SLAM2 之可工作光照度範圍約落在 1 k~100 k lux 之間，而 Enhanced ORB-SLAM2 由於加入了偵測 ArUco Marker 之功能，受環境光照條件的影響相對較小，可工作光照度範圍約落在 1 k~150 k lux。

## 4.2 地圖校正

### 4.2.1 ICP 方法之地圖校正

Iterative Closest Point (ICP) 是一種常用的點雲匹配演算法，用於將兩個或多個點雲資料集對齊。ICP 在電腦視覺、機器人領域以及三維重建等應用中廣泛使用。ICP 校正是透過迭代的方式將一個點雲圖轉換到另一個點雲圖的座標系中，使兩個點雲圖之間的距離最小化。而我們使用此概念將地圖中的 MapAruco 轉換到實際量測的 Ground truth Aruco，我們使用 Python 中的 open3d 函式庫提供的剛性 ICP 算法 (`o3d.pipelines.registration.registration_icp()`)，計算完成後可得到一組 4x4 的齊次坐標（旋轉矩陣與位移矩陣），使用此矩陣轉換 MapAruco 即可得到校正後的 Aruco 位置。使用 ICP 校正結果如圖 4-1 (a) 所示，圖中校正前的紅點，移動至校正後的藍點，與 Ground truth 的黑點更為接近。校正前的 RMSE 為 1.073 m，而校正後的 RMSE 下降至 0.145 m。但使用 ICP 校正兩組點雲圖，僅使用迭代方式找到最佳解，沒有利用已知的匹配資訊，因此容易落入局部最佳解 (local minima) 如，圖 4-1 (b) 所示，當點雲圖往不同方向移動時，皆無法得到更低的誤差，迭代停止在局部最佳解。此外，由於本研究使用剛性 ICP 轉換，因此無法將 ArUco 點對之間的誤差校正至無誤差，下一節將討論如何應用仿設變形轉換校正地圖。

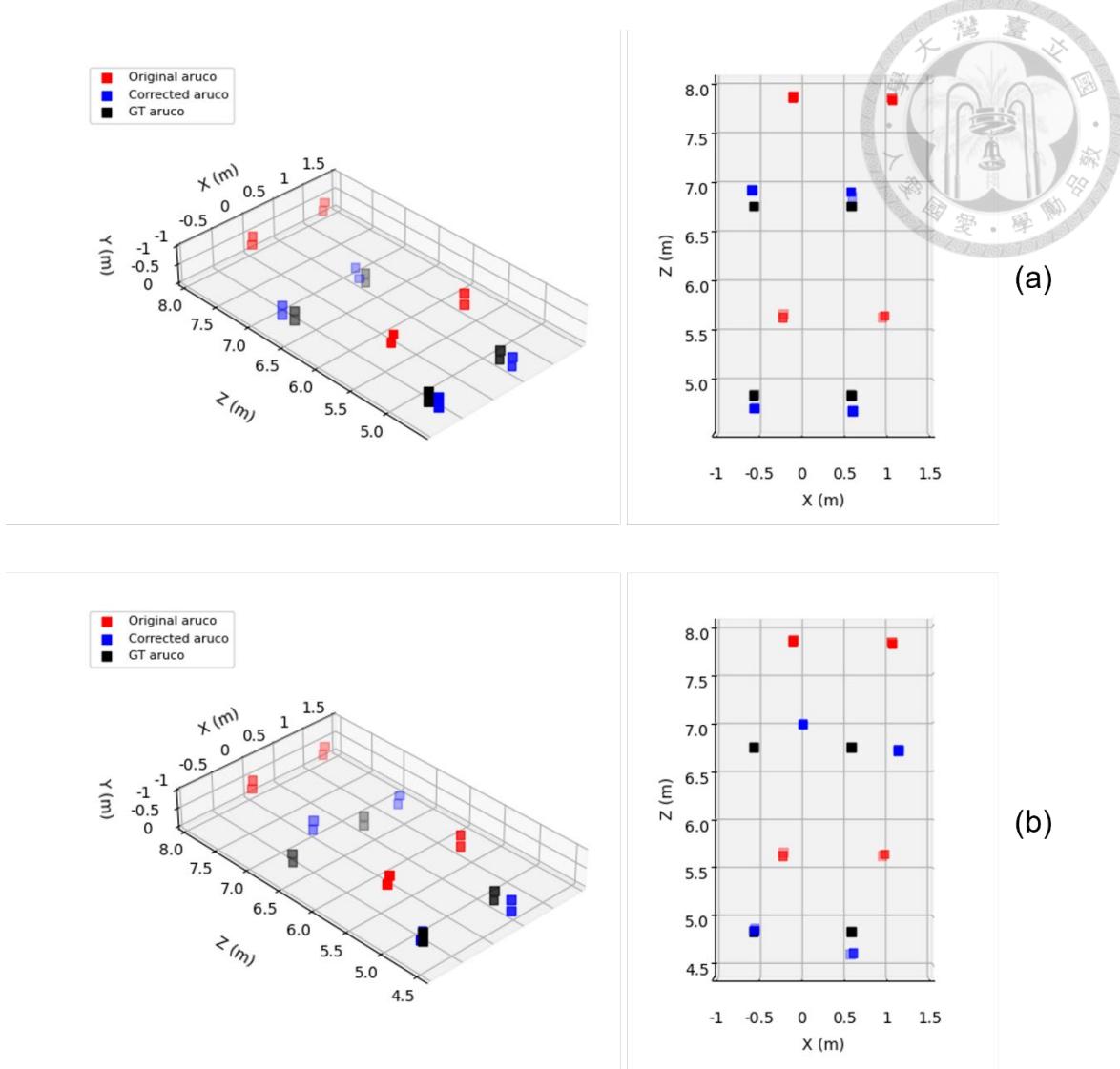


圖 4-1 ICP 校正溫室 ArUco Marker 位置，紅色方塊為地圖中的 MapAruco 位置，藍色方塊為校正後的 ArUco 位置，黑色方塊為實際在溫室量測的 ArUco 位置，  
 (a)ICP 在全局最佳解；(b)ICP 在局部最佳解

#### 4.2.2 仿射轉換方法之地圖校正

仿射轉換實驗的輸入為 Enhanced ORB-SLAM2 地圖的 MapAruco 與人工實際在溫室量測的 ArUco Marker 位置。本實驗中使用 23 個 ArUco Marker 進行校正，透過 PyGeM 函式庫中的 rbf 函示可求得  $(23 + 1 + 3) \times 3$  的變換矩陣，此矩陣用於校正 Enhanced ORB-SLAM2 地圖的所有資訊。如圖 4-2 所示，使用變換矩陣將 Original aruco（紅色方框）校正至 Corrected aruco（藍色方框）。校正前 MapAruco 與 Ground truth aruco 的均方根誤差分別為：x 方向 0.262 m、y 方向 1.884 m、z 方向 2.592 m，而校正後 MapAruco 與 Ground truth aruco 的均方根誤差在三



個軸的誤差皆為 0，可與人工實際量測的位置完全貼合。如圖所示，Corrected aruco（藍色方框）與 GT aruco（黑色十字）在 3d 空間中重疊，表示仿射轉換的地圖校正精度高，可將地圖中有誤差的 MapAruco 校正至溫室中 Aruco marker 的真實位置。

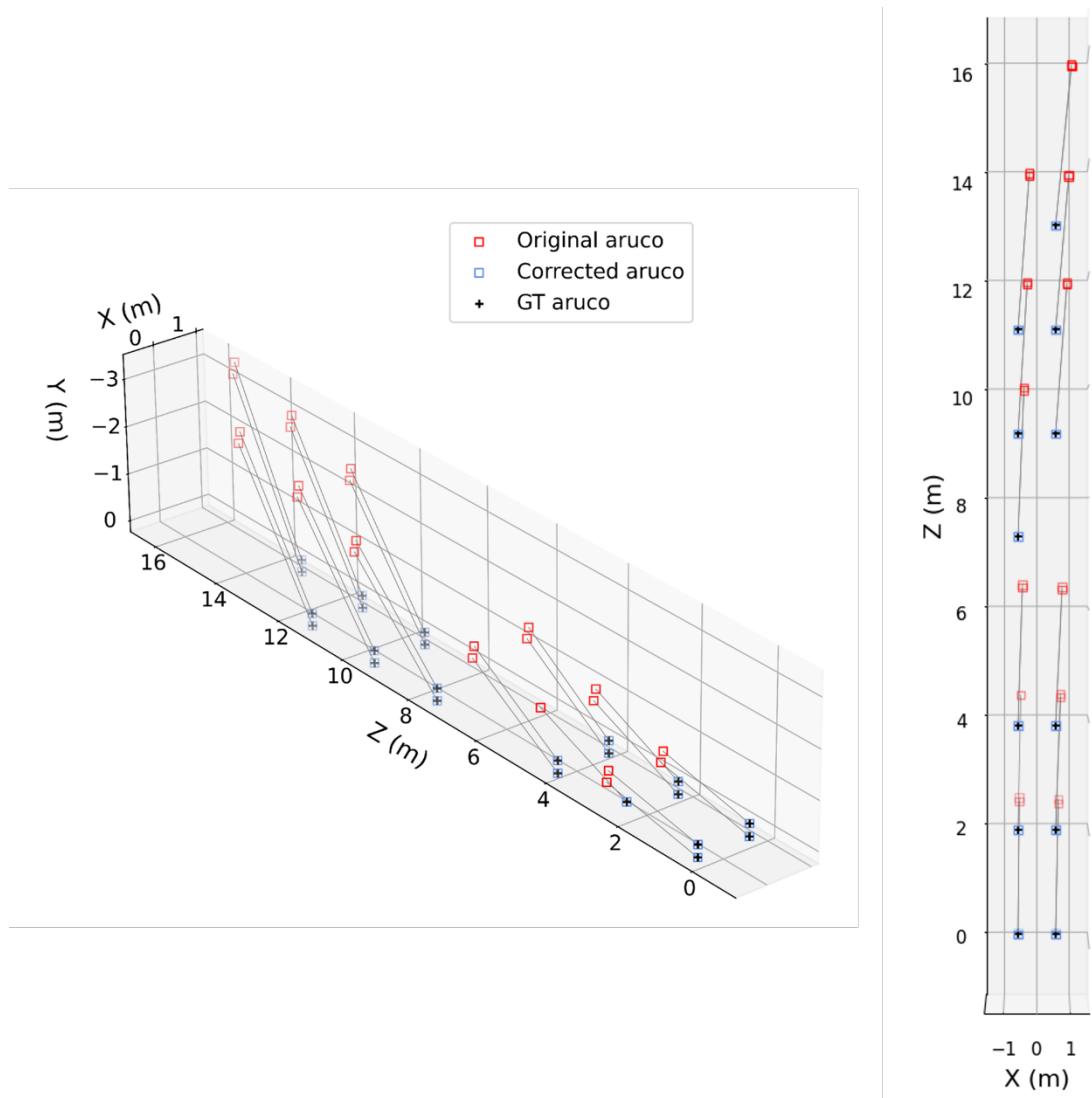


圖 4-2 仿射轉換校正將地圖之 MapAruco 校正至溫室的真實 ArUco Marker 位置

Enhanced ORB-SLAM2 的地圖使用二進制的檔案儲存 (.bin)，由於地圖使用二進制存取，因此需了解地圖儲存的格式與順序，才能順利的完成讀取。我們依照存取的順序，將原始地圖使用 python 的 struct 函式庫取出，再將 MapAruco、MapAruco 與 KeyFrames 三者的位置資訊使用 PyGem 計算的變換矩陣校正位置，最終寫入一個新的地圖檔，完成地圖校正。

### 4.2.3 校正地圖之定位效果分析

本節分析 Enhanced ORB-SLAM2 使用原始地圖與校正地圖之間的差異，比較的方法與光照度實驗相同，計算使用原始地圖與校正地圖下，平均每每個 frame 偵測到的 Keypoints 數量 (keypoints per frame) 以及追蹤率 (tracking rate)。

如圖 4-3 所示，原始地圖的建立是使用一組 rosbag 所儲存的影片，而校正地圖則是使用論文中 3.3.2 節所提及的仿射轉換算法，將原始地圖經由 MapAruco 校正到真實位置。在分析兩種地圖的定位效果上，我們使用建立原始地圖的 rosbag 影片做為輸入，比較兩種地圖在 Enhanced ORB-SLAM2 算法輸出的差異。實驗結果如表 4-3 所示，原始地圖平均每個 frame 的特徵點數量為 683，而使用校正地圖則下降至 385 個，得到相對較差的結果，在追蹤度上也下降約 1.5 %。

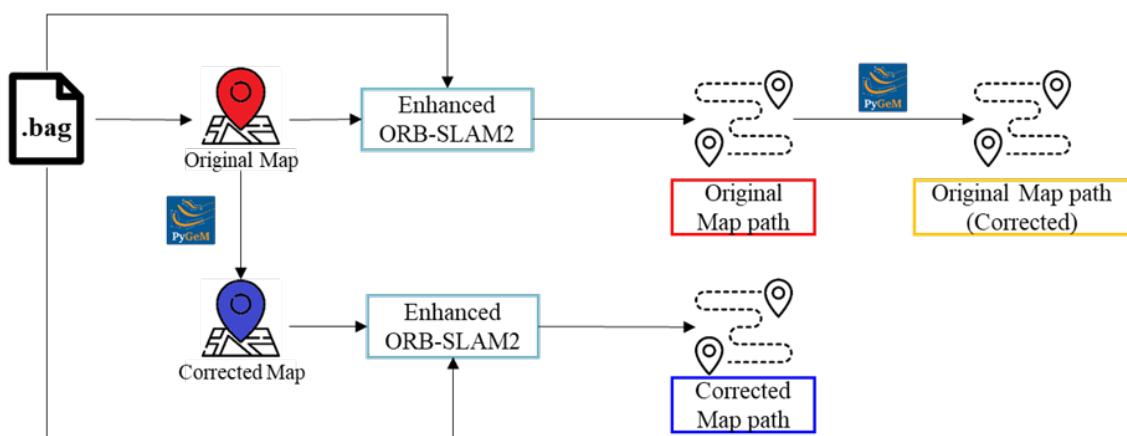


圖 4-3 原始地圖與校正地圖之計算方法與三種飛行軌跡

表 4-3 原始地圖與校正地圖之平均特徵點數量與追蹤率

	Keypoints per frame	Tracking rate (%)
Original Map	683	99.92
Corrected Map	385	98.39



經由 Enhanced ORB-SLAM2 系統運行的視覺化視窗可以觀察到，使用原始地圖的追蹤畫面較為穩定，而校正地圖的在相機位置的估計上有較大的誤差，相機位姿在點雲圖中不穩定的移動。如圖 4-4 所示，使用原始地圖可得到紅色軌跡，校正地圖為藍色軌跡，從圖中可看出校正地圖的定位效果相對不穩定，有許多藍色的線段發生不連續的位移，這是因為位姿估計的錯誤計算。而使用原始地圖則沒有大幅度的位姿計算錯誤，可以說相對較為穩定。

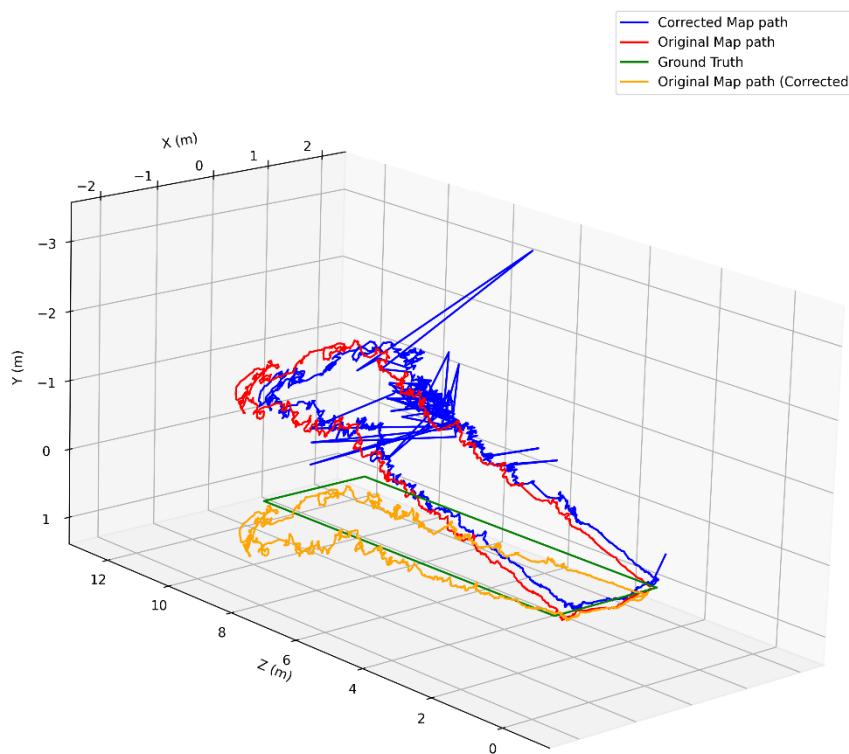


圖 4-4 原始地圖軌跡與校正地圖軌跡之比較圖



在本研究中，我們嘗試使用仿射轉換對 Enhanced ORB-SLAM2 的原始地圖進行校正，然而我們觀察到計算相機位置的結果出現了不穩定性。這種不穩定性可以歸因於以下兩個因素：

拓撲關係 (topology relationship)，當應用仿射轉換時，地圖中部分特徵點的位置和相互關係可能會發生改變，這可能導致拓撲關係的變化，即地圖中特徵點之間的空間關係。原始地圖中可能存在著特定的拓撲結構，例如特徵點以直線或曲線方式連接。然而，當我們對地圖應用仿射轉換進行校正時，這些原始的拓撲關係可能會被破壞或改變，進而導致關鍵信息的丟失。相機位置的估計通常基於地圖中特徵點的觀測和匹配，而如果特徵點的位置和相互關係發生變化，計算相機位置的準確性和穩定性可能會受到影響。

變形效應，仿射轉換僅能處理簡單的幾何變換，如平移、旋轉和縮放。然而，地圖中可能存在較大的變形，例如非線性扭曲或透視變換，這些變形超出了仿射轉換的處理範圍。因此，當我們嘗試使用仿射轉換校正這些地圖時，無法準確地處理這些變形，進而導致計算相機位置的不穩定性。

因此，在後續的實驗中，我們將直接使用原始地圖計算位姿，再對位姿結果進行校正，以獲得更準確且穩定的結果。如圖 4-4 所示，黃色軌跡是將紅色軌跡透過 PyGem 的變換矩陣所求得，通過對原始地圖的位姿進行校正，我們能夠獲得更接近綠色真實軌跡且穩定的結果。

#### 4.2.4 SLAM 建圖可視化結果

本研究使用的 SLAM 系統分成 ORB-SLAM2 與 Enhanced ORB-SLAM2 兩種，兩種 SLAM 的地圖形式也有所不同。ORB-SLAM2 的地圖保存了 MapPoint 與 KeyFrame 兩個資訊，在可視化地圖中 MapPoint 以黑色點代表，而 KeyFrame 由深藍色方框內畫上叉代表。Enhanced ORB-SLAM2 比原始的 ORB-SLAM2 多了偵測 ArUco Marker 的功能，而儲存於地圖中稱為 MapAruco，在可視化地圖中可以看到 MapAruco 以綠色方框代表。

表 4-4 ORB-SLAM2 建立之三種地圖

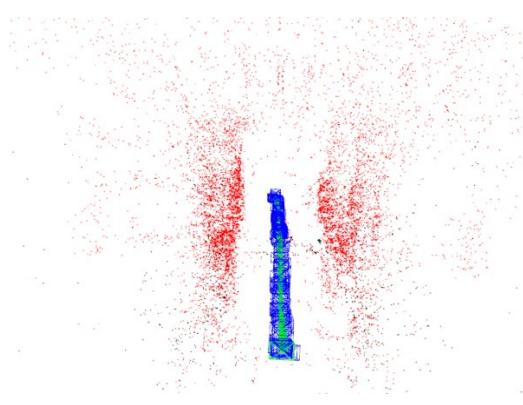
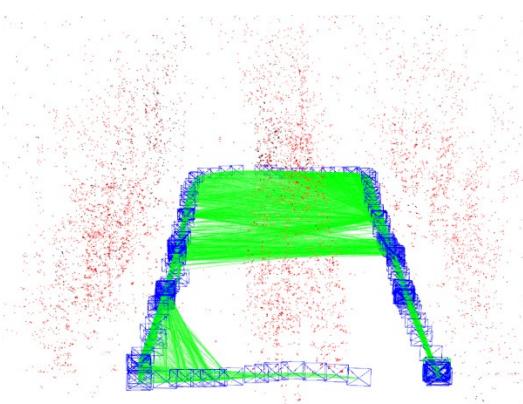
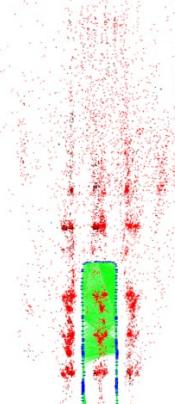
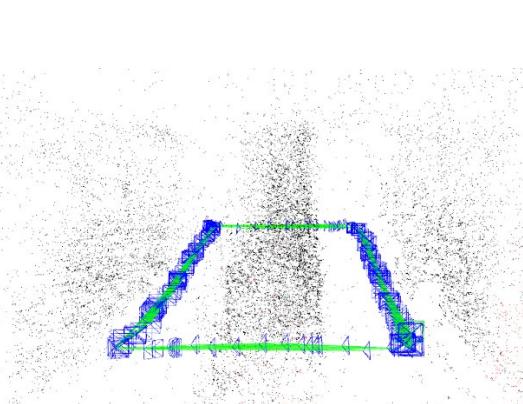
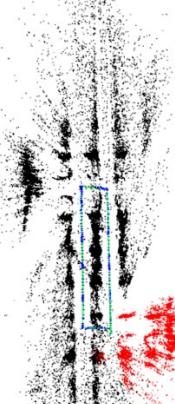
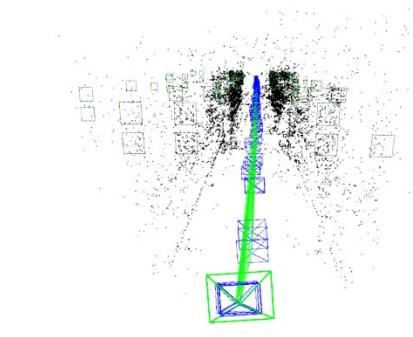
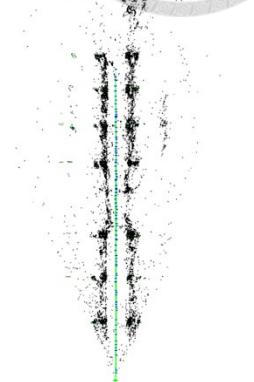
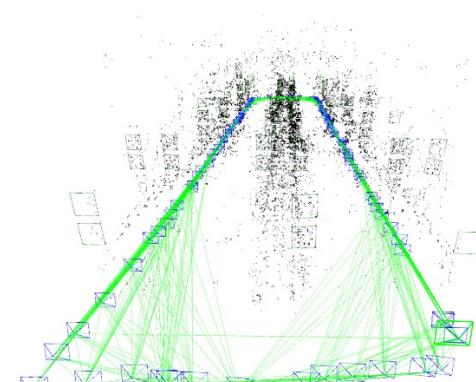
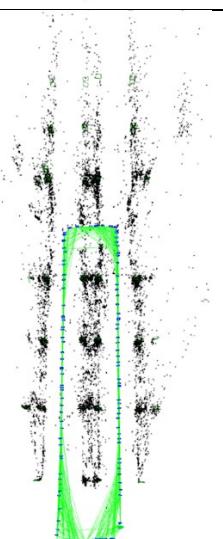
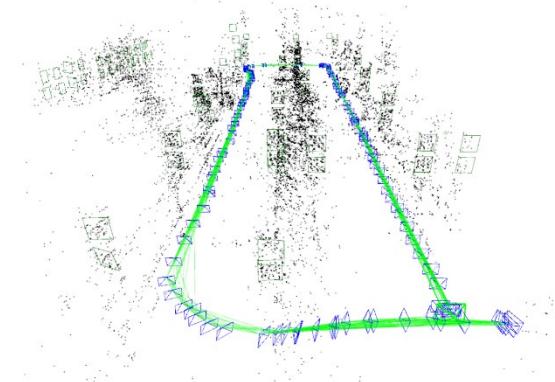
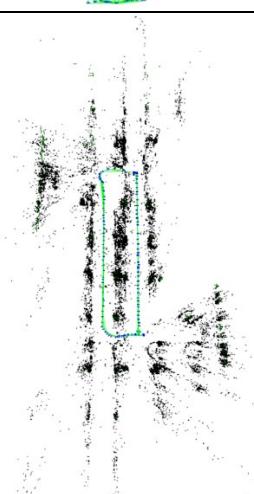
	Front view	Top view
Straight Line		
Closed-loop without turning		
Closed-loop with turning		

表 4-5 Enhanced ORB-SLAM2 建立之三種地圖

	Front view	Top view
Straight Line	 A front view of a SLAM map showing a straight horizontal trajectory. The trajectory is highlighted with a green line and blue markers. The map includes a grid of buildings and a small green rectangle at the bottom.	 A top-down view of the same SLAM map, showing a vertical trajectory highlighted with a green line and blue markers. The map features a grid of buildings and a small green rectangle at the bottom.
Closed-loop without turning	 A front view of a SLAM map showing a closed-loop trajectory that does not involve turns. The trajectory is highlighted with a green line and blue markers, forming a roughly triangular loop. The map includes a grid of buildings and a small green rectangle at the bottom.	 A top-down view of the same SLAM map, showing a vertical trajectory highlighted with a green line and blue markers. A green rectangle highlights a specific section of the trajectory.
Closed-loop with turning	 A front view of a SLAM map showing a closed-loop trajectory that includes turns. The trajectory is highlighted with a green line and blue markers, forming a loop with a curved segment. The map includes a grid of buildings and a small green rectangle at the bottom.	 A top-down view of the same SLAM map, showing a vertical trajectory highlighted with a green line and blue markers. A green rectangle highlights a specific section of the trajectory.



## 4.3 飛行控制分析

### 4.3.1 SLAM 系統與飛行軌跡分析

本節依照不同的 ORB-SLAM 系統與飛行方式設計了 15 種實驗，並分析 PID 控制下的無人機飛行軌跡誤差，誤差包括空間座標的 xyz 誤差和 yaw 方向的角度誤差。實驗中考慮的控制變因如表 4-6 所示，分別使用了原始的 ORB-SLAM2 和本研究改進後的 Enhanced ORB-SLAM2 系統。此外，原始的 ORB-SLAM2 系統根據場域是否架設 ArUco Marker 進行比較。另外，飛行軌跡分為 5 種，直線、倒退繞圈、轉彎繞圈、全溫室繞圈、全溫室 S 型，如圖 4-5 所示。15 種實驗最終都重複進行了三次，以確保實驗結果的可重複性和獲得平均值。

表 4-6 九種無人機飛行實驗控制變因設定

SLAM System	ORB-SLAM2	ORB-SLAM2	Enhanced ORB-SLAM2
Presence of ArUco Markers in the Flight Area	No	Yes	Yes
Straight Line	Exp. 1-1	Exp. 1-2	Exp. 1-3
Closed-loop without turning	Exp. 2-1	Exp. 2-2	Exp. 2-3
Closed-loop with turning	Exp. 3-1	Exp. 3-2	Exp. 3-3
Full Greenhouse Loop	Exp. 4-1	Exp. 4-2	Exp. 4-3
Full Greenhouse S-shape	Exp. 5-1	Exp. 5-2	Exp. 5-3

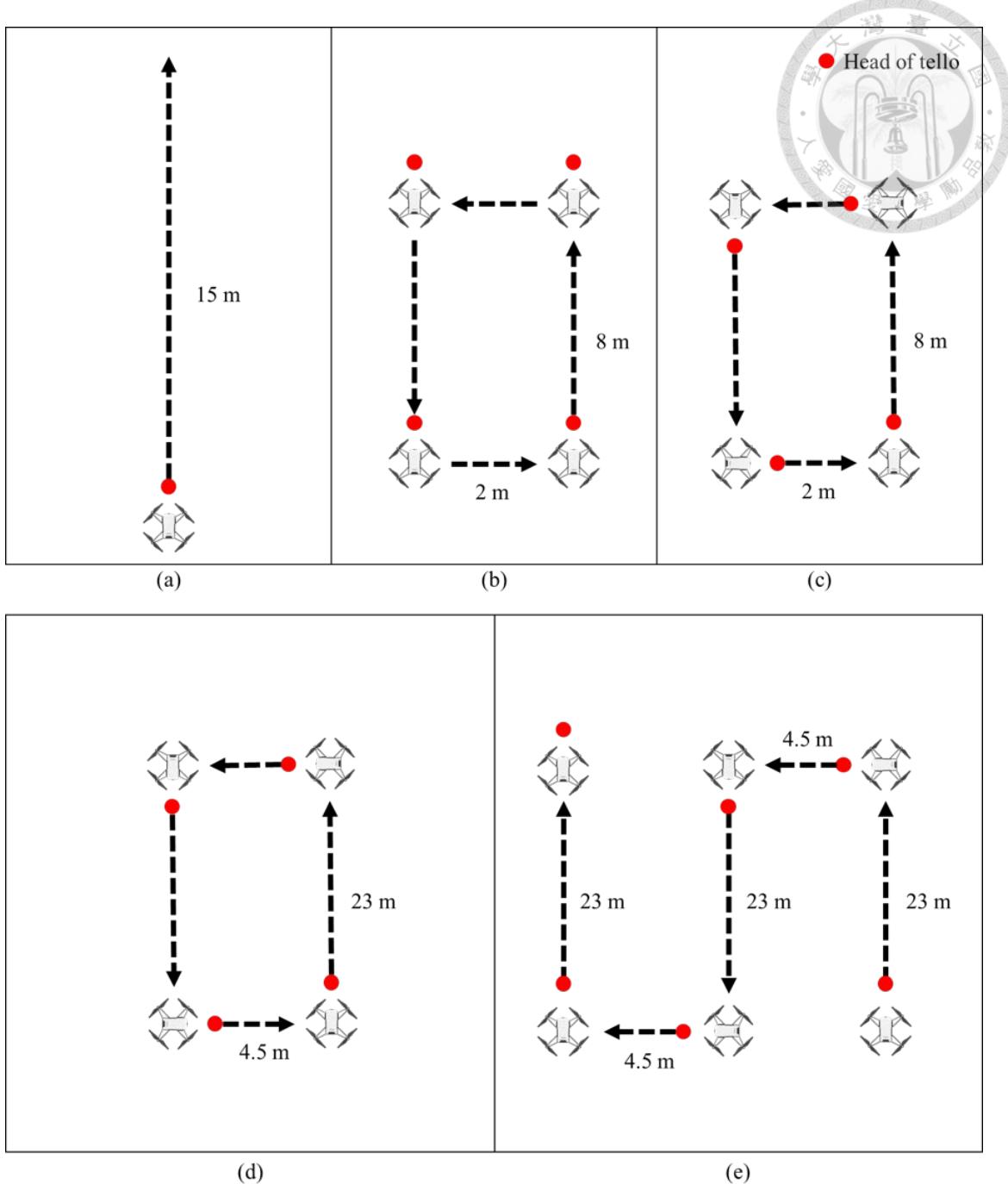


圖 4-5 五種飛行實驗之軌跡：(a)直線；(b)倒退繞圈；(c)轉彎繞圈；  
(d)全溫室繞圈；(e)全溫室 S 型

實驗結果分析如表 4-7 所示，從平均的空間座標誤差來看 (mean of x,y,z error)，實驗一到實驗三的誤差非常相近，空間誤差範圍在 0.22 公尺到 0.29 公尺之間，而實驗四、五因為距離較遠，在深度方向的誤差累積較多，因此誤差相對較大。

首先分析實驗一到實驗三的結果，從三種飛行軌跡來分析結果，觀察 x RMSE 可以得知，倒退繞圈的 x RMSE 範圍在 0.24 公尺到 0.3 公尺之間，此誤差比直線與轉彎繞圈來得更大，主要原因是倒退繞圈在轉彎處有較大的橫向移動，這是其他兩種飛行軌跡不需要的。觀察完成飛行任務所花費的 frame 數量 (number of frame)，可以發現平均的 frame 數量由小到大分別為直線、倒退繞圈、轉彎繞圈，這代表著任務的難度越高使得無人機導航系統需要花費更多的時間完成任務。在轉彎繞圈的任務中，無人機在四個角落點皆須轉 90 度，角度的誤差落在 33~35 度之間，會有如此大的誤差主要原因是我們將控制 yaw 轉角的 P 調整的較小，在轉彎時無人機是緩慢的達到指定的角度，如圖 4-7 角度圖所示，綠色線為目標的轉角，橘黃色線為 EKF 估算的無人機當前轉角，EKF 位置會逐漸調整至目標角度。將 P 調小目的是為了讓無人機能緩慢地轉動，這有助於減少 VSLAM 在轉彎時畫面變化太快所導致的追蹤失敗問題。而實驗四、五的平均飛行軌跡誤差較大，分析可能的原因為飛行距離較長，當無人機判定抵達航點後，會指派下一個航點，期間尚未到達的位置誤差也累積較多，隨著飛行距離加長，誤差也相對提升。另外實驗四與實驗五在轉彎處，無人機皆會旋轉 90 度，而其角度誤差僅在 10 ~ 15 度之間，相比實驗三之角度誤差 (30 度以上) 較小，這是因為飛行的路徑較長，期間直線飛行的部分分攤掉了角度的誤差，因此在實驗四、五中的平均較度均方根誤差會相對實驗三的結果來得小一些。

接著我們由不同 SLAM 系統的角度來解析實驗結果，首先最明顯可以看到的為 Enhanced ORB-SLAM2 系統在所有的飛行任務中，需要的 frame 數量為三者之間最少的，在直線、倒退繞圈、轉彎繞圈、全溫室繞圈、全溫室 S 型分別使用了 873、1006、1625、2437、3683 個 frame，亦代表執行時間上得到三種 SLAM 系統中最佳的結果。

表 4-7 九種無人機飛行實驗之平均均方根誤差（空間、角度）與平均使用幀數

Flight Trajectory	Exp .	RMSE of x (m)	RMSE of y (m)	RMSE of z (m)	Mean of x,y,z RMSE (m)	Angle RMSE (degree)	Number of Frame
Straight Line	1-1	0.18	0.05	0.55	0.26	3.27	1261
	1-2	0.11	0.10	0.50	0.24	1.11	1154
	1-3	0.12	0.07	0.50	0.23	1.01	873
Closed-loop without turning	2-1	0.25	0.05	0.48	0.26	3.17	1328
	2-2	0.24	0.18	0.42	0.28	2.44	1883
	2-3	0.30	0.13	0.44	0.29	1.55	1006
Closed-loop with turning	3-1	0.19	0.11	0.40	0.23	34.38	1825
	3-2	0.18	0.08	0.41	0.22	33.86	1818
	3-3	0.20	0.11	0.39	0.23	34.94	1625
Full Greenhouse Loop	4-1	0.48	0.27	0.96	0.57	13.7	3482
	4-2	0.43	0.25	1.02	0.57	12.2	3408
	4-3	0.44	0.21	0.82	0.49	9.80	2437
Full Greenhouse S-shape	5-1	0.39	0.25	1.12	0.59	14.09	4355
	5-2	0.38	0.23	1.10	0.57	15.70	4832
	5-3	0.39	0.24	1.06	0.56	13.39	3683

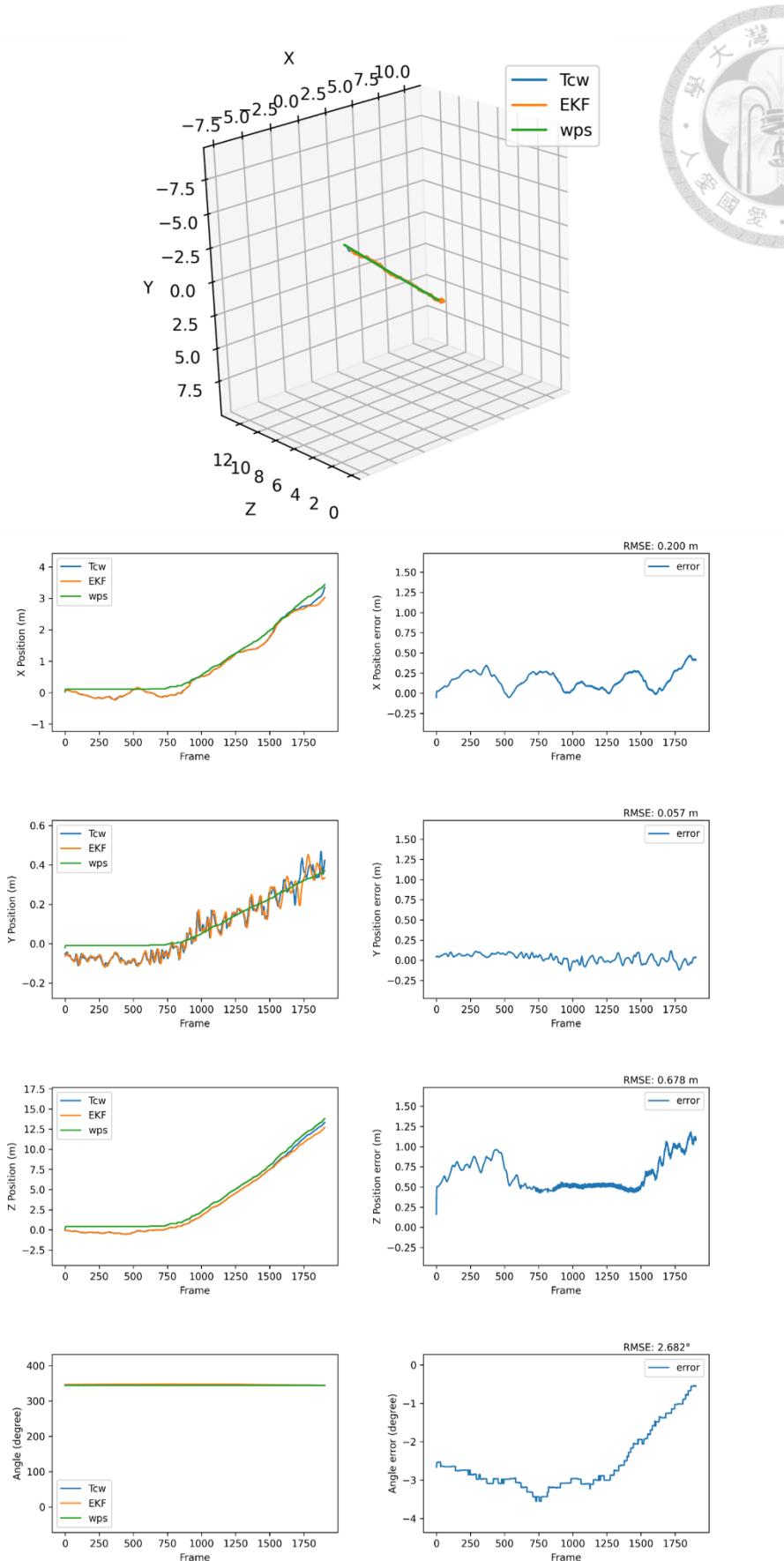


圖 4-6 直線飛行之 3D 空間軌跡圖與誤差圖

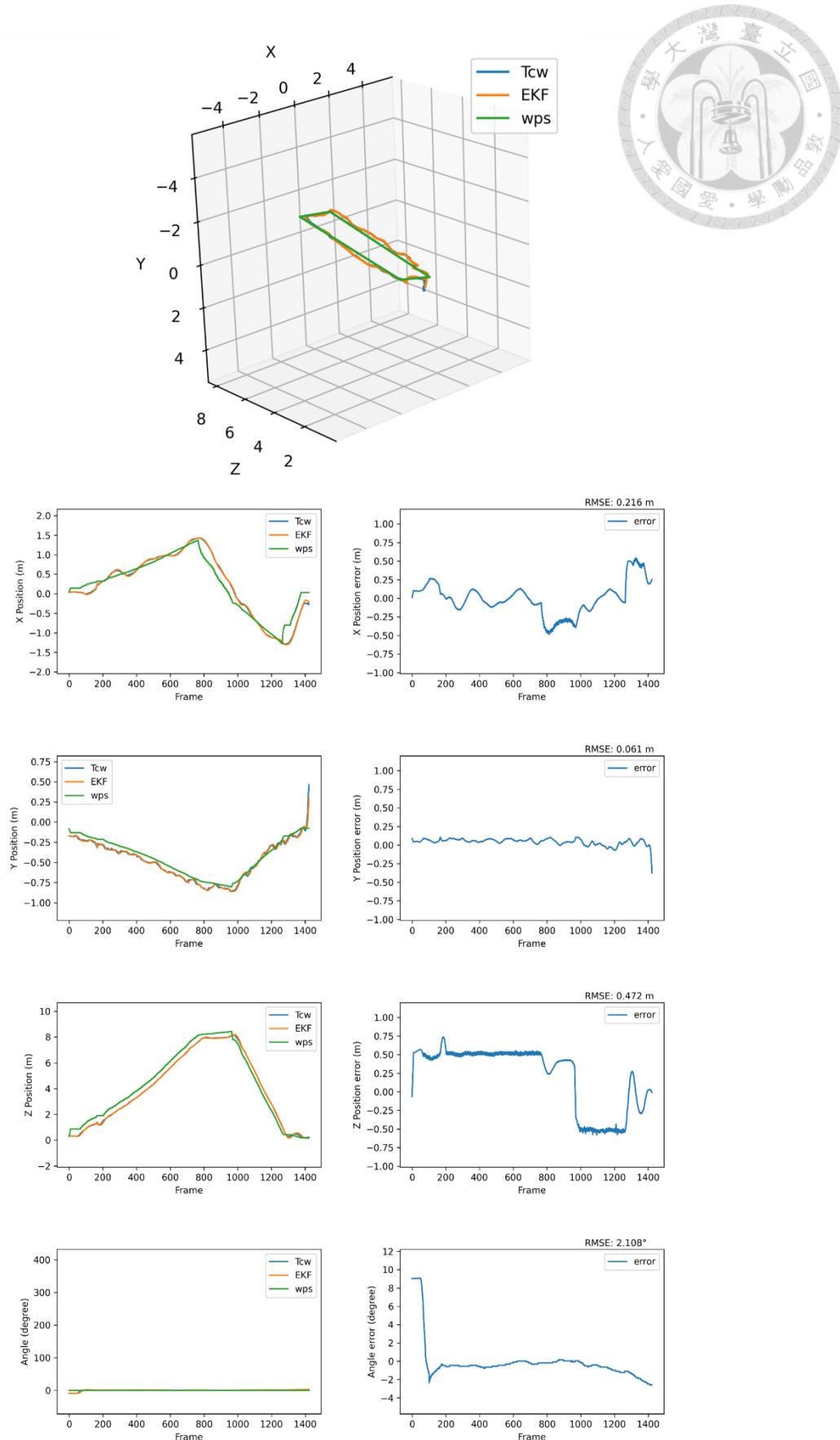


圖 4-7 倒退繞圈之 3D 空間軌跡圖與誤差圖

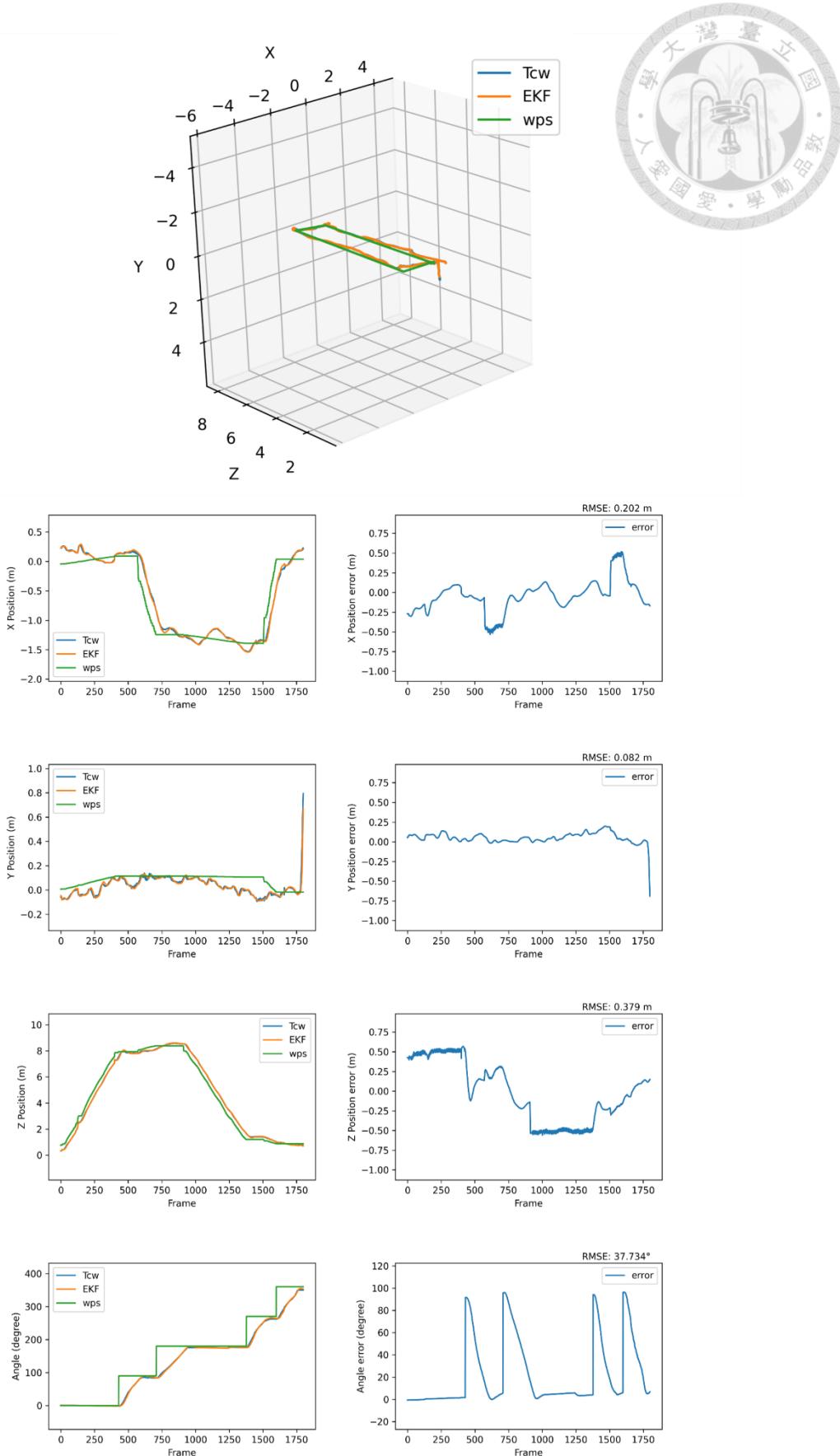


圖 4-8 轉彎繞圈之 3D 空間軌跡圖與誤差圖



### 4.3.2 轉彎 PID 參數調整

在轉彎繞圈的飛行中，我們發現無人機在經過轉彎後，直線飛行會有持續轉動的問題，如圖 4-9 (a)與 (b)所示，圖 a 中，約 1000 到 1600 的 frame 中有明顯的左右轉動問題，其具有規律性震盪經分析可能是轉彎後角度存在誤差，且使用的 P 值過大，導致控制上無法停止在目標點。如圖 4-9 (c)與(d)為 P 值調降後的結果，在相同的區域中轉角不再震盪，穩定的控制在目標角度。

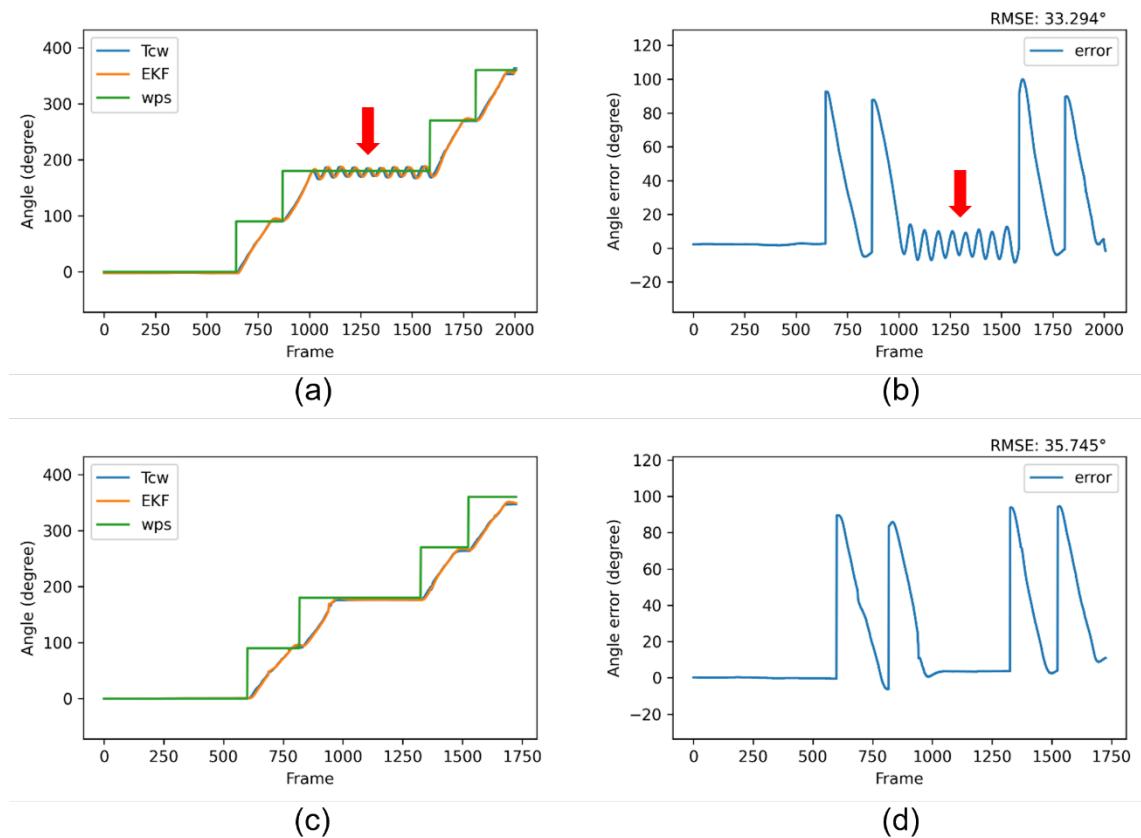


圖 4-9 P 值過大導致轉彎產生擺動之角度圖與角度誤差圖



### 4.3.3 不同 SLAM 系統之差異比較

在執行表 4-4 的九種實驗時，我們發現 ORB-SLAM2 的系統，在溫室光影有變化時（包含光照度與陰影），無法在起飛後直接定位成功，需先使用手動遙控無人機，使其飛行一段距離，SLAM 算法才可重新定位，定位成功後才可執行導航任務，因此 ORB-SLAM2 的系統飛行實驗選擇在陰天光照度均勻且沒有陰影的情況下，使無人機可以在起飛後立即定位成功。

圖 4-10 分析了三種系統在轉彎時的定位效果。圖 4-10 (a)展示了在沒有 aruco 標記的場地上使用 ORB-SLAM2 進行追蹤的結果。在溫室中的這個轉角，特徵主要聚集在控制箱按鈕的黑白特徵上。然而，當無人機逐漸轉向另一個方向時，透明玻璃上沒有明顯的特徵，這導致追蹤的特徵點數量從約 600 個降至約 100 個。特徵點的大幅減少會影響系統的定位穩定度。因此，如果要使用 ORB-SLAM2 進行轉彎任務，需要在轉角提供足夠的特徵點以進行定位。圖 4-10 (b)同樣使用 ORB-SLAM2 作為定位系統，但不同的是在帶有 ArUco 標記的區域進行飛行。ArUco 標記能夠提供 ORB-SLAM2 大量穩定的特徵，使得無人機在轉彎時能夠維持追蹤的特徵點數量在 400 以上，進而實現相對穩定的定位效果。Enhanced ORB-SLAM2 的結果如圖 4-10 (c)，此系統可以偵測 ArUco Marker，並使用藍色方框將其標記出來，同時在 ArUco Marker 的右下方顯示其 ID。從結果來看，在有架設 ArUco Marker 的區域轉彎，皆能使 SLAM 系統穩定的在溫室中定位。

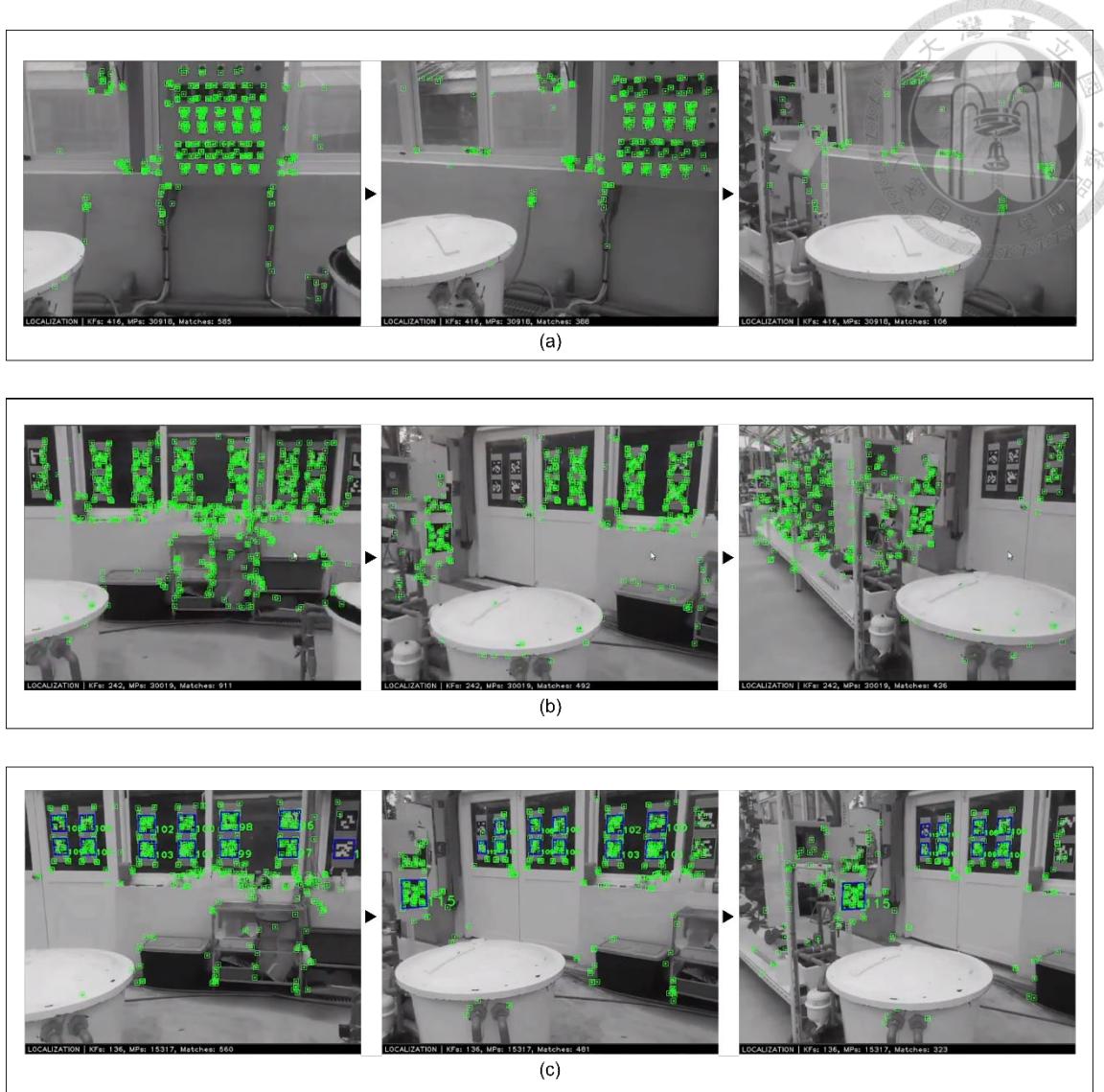


圖 4-10 無人機轉彎定位效果分析：(a)在無 aruco 區域使用 ORB-SLAM2；  
 (b)在有 aruco 區域使用 ORB-SLAM2；(c)Enhanced ORB-SLAM2

## 4.4 果實算法分析

### 4.4.1 果實偵測模型效能分析

本研究在國立臺灣大學農業試驗場的玻璃溫室蒐集 211 張洋香瓜 RGB 影像，其影像解析度為 960 x 720 pixels，使用 LabelImg 標記果實位置後用於訓練 YOLOv4 果實偵測模型。211 張影像依照 7：2：1 分成訓練集、驗證集與測試集，模型訓練使用 0.0013 的學習率共 4000 次 iteration，結果如表 4-8 所示，mAP 達到

0.96，在 24 張影像中正確的偵測到 63 顆果實，而有 8 顆不是果實的部分被模型誤認為果實，最後有 4 顆果實沒有被正確地偵測到。

表 4-8 YOLOv4 果實偵測模型之訓練結果

Metric	Value
mAP	0.96
F <sub>1</sub> -score	0.91
Precision	0.89
Recall	0.94
TP	63 melons
FP	8 melons
FN	4 melons
Average IoU	72.5 %

#### 4.4.2 假果實追蹤結果分析

為了分析果實追蹤的結果，本實驗吊掛 15 顆直徑為 14cm 的假果實在溫室的角鋼架，並使用手持無人機方式，錄製六組 RGB 影片進行果實追蹤分析。

實驗一的結果如表 4-9 所示，我們對實驗一的每一組分別進行了果實追蹤算法的原始結果 (before)和三步驟資料清理後 (after)的差異比較，此實驗中使用三步驟的資料清理方式已詳述於 3.5.1 節，將第一步驟的 size threshold 與第二步驟的 frame threshold 皆設為 50。在六組實驗中，所有未做資料清理的結果都發生 ID switch (before)，其中 Exp. I -1 的 ID switch 數量最少，僅發生一次，而 Exp. I -6 的 ID switch 數量最多，達到 12 次。相反地，透過進行三步驟資料清理 (after)，我們成功地排除所有 ID switch 的情況，六組實驗中的追蹤數量皆為 15 顆，誤差皆為 0。這表明使用三步驟資料清理方法，能降低果實追蹤發生 ID switch 的狀況，準確地追蹤果實並得到正確的數量。綜合所有實驗的平均數據 (mean)，資料清理前平均每組實驗的 ID switch 數量為 5.83 顆，平均追蹤數量誤差為 2.5 顆，而資料清理後的 ID switch 數量與平均追蹤數量誤差皆下降至 0 顆。

表 4-9 假果實追蹤結果使用三步驟資料清理前後之分析

		ID switch	Real Number	Tracking Number	Error
Exp. I-1	Before	1		15	0
	After	0		15	0
Exp. I-2	Before	3		16	1
	After	0		15	0
Exp. I-3	Before	7		18	3
	After	0		15	0
Exp. I-4	Before	4		15	1
	After	0		15	0
Exp. I-5	Before	8		21	6
	After	0		15	0
Exp. I-6	Before	12		19	4
	After	0		15	0
Mean	Before	5.83		17.5	2.5
	After	0		15	0

在資料清理前的果實追蹤結果有發生 ID switch，而 ID switch 的計數方式如下說明。圖 4-11 (a)與 (b)為 Exp. I-1 的追蹤結果，圖 4-11 (a)的 11 號果實在圖 4-11 (b)被誤判成 8 號果實，果實 ID 的變換被記為一次 ID switch。另一種情況如圖 4-11 (c)與(d)所示，這兩張圖像為 Exp. I-6 的追蹤影片截圖，果實 ID 8 與果實 ID 9 經過 36 個 frame 後 ID 相互轉換，此情況下記為兩次 ID switch。圖 4-11 (e)與 (f)為 Exp. I-4 的追蹤結果，圖 (e)最右側的果實已經被標記為 ID 4，經過 96 個 frame 後，一顆新的果實被誤判成 ID 4，此狀況記為一次 ID switch。圖 4-11 (a)、(b)、(c)、(d)的 ID switch 為果實 ID 變換，但圖 4-11 (e)、(f)的 ID switch 是新果實誤判成舊果實，兩者的發生原因與解決方法不同，將在下一段落分析與說明。

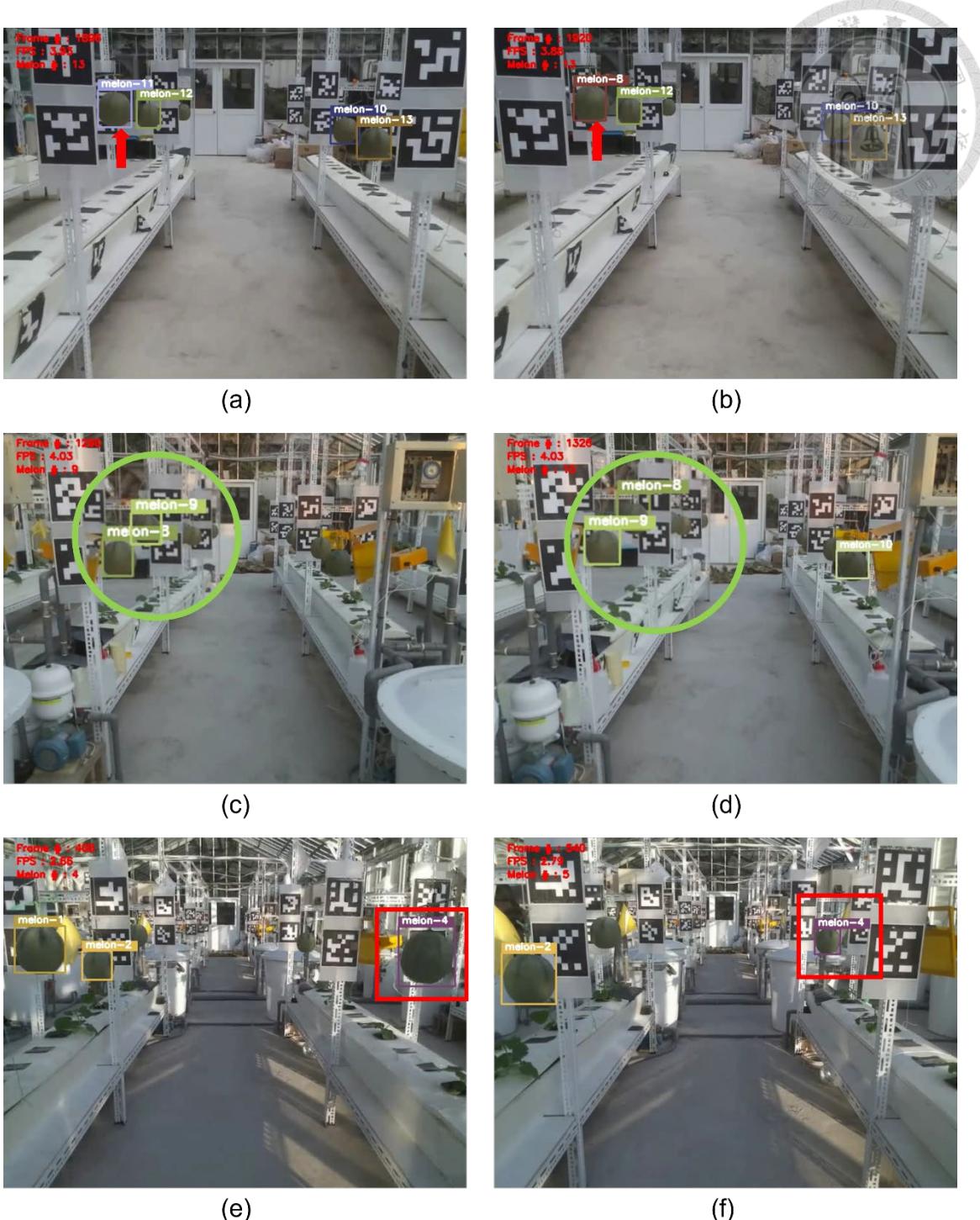


圖 4-11 假果實追蹤結果之不同 ID switch 分析

在 Exp. I 中我們直接將 size threshold 與 frame threshold 皆設為 50，本節將分析不同的 size threshold 與 frame threshold 會如何影響果實追蹤結果。本實驗使用第六個影片做分析，因其 ID switch 發生次數最高，在調整參數時，容易比較不同參數之間的差異性。如表 4-10 的所示，增加 Size Threshold 代表濾除較遠的果實框，隨著閥值的提升，Total ID switch 的數量隨之下降，可以說明果實距離相機較遠時，容易發生 ID switch，將閥值提升至 50 後可有效地將 Total ID switch 從 12 下降至 2。此外，Frame Threshold 的設置是防止出現次數太少的果實，降低果實追蹤的雜訊，Frame Threshold 由 0 提升至 70，Total ID switch 由 12 下降至 8，代表 Frame Threshold 對於 Total ID switch 並沒有明顯的下降，但仍然濾除部分短暫出現的錯誤果實。當 Size Threshold 與 Frame Threshold 兩個閥值皆提升至 70 時，算法僅追蹤到 3 顆果實，是因為將 Size 小於 70 個 pixel 大小的果實框都濾除後，剩餘的 frame 數量會很低，此時再用 Frame Threshold 70 濾除，會濾除大部分的果實數據，導致最後的追蹤結果僅得到 3 顆果實。

表 4-10 將 Total ID switch 分成兩種做分析，第一種為果實 ID 變換，第二種為新果實誤判成舊果實，提升與 Size Threshold 與 Frame Threshold 可有效地降低果實 ID 變換（第一種 ID switch），而新果實誤判成舊果實（第二種 ID switch）無法經由閥值濾除，如表中所示，可有效濾除第一種 ID switch 的 Size Threshold，由 50 提升至 70 無法降低第二種 ID switch。需使用第三步驟的資料清理方法濾除第二種的 ID switch。

由結果可以看出，兩種閥值皆可降低 Total ID switch，且 Size Threshold 的效果更加明顯，Size Threshold 由 0 提升至 70 的結果所示，將 Size Threshold 設置在 50 足以濾除第一種 ID switch，因此將 Size Threshold 設置在 50。而從 Frame Threshold 由 0 提升至 70 的結果可看出，雖效果不如 Size Threshold 明顯，但仍然可降低 Total ID switch。將兩個閥值都設為 50 可以完全濾除第一種 ID switch，因此最終選擇兩個閥值都設定為 50。

表 4-10 調整 Size Threshold 與 Frame Threshold 之假果實追蹤 ID switch 數量差異

Threshold		Total ID switch	First Type of ID Switch	Second Type of ID Switch	Real Number	Tracking Number	Error
Size	Frame						
0	0	12	10	2		19	4
30	0	10	10	0		19	4
50	0	2	0	2		13	-2
70	0	2	0	2		13	-2
0	30	9	6	3	15	15	0
0	50	8	3	5		13	-2
0	70	8	3	5		13	-2
30	30	8	6	2		15	0
50	50	2	0	2		13	-2
70	70	2	0	2		3	-12

如表 4-11 所示，表格中的 Before 代表僅使用第一、二步驟的資料清理，size threshold 與 frame threshold 皆設為 50。After 代表加入第三步驟的資料清理。Exp. II 證明了第三步驟可有效濾除第二種的 ID switch。在六組實驗中，Before 的平均 ID switch 數量為 1.17 顆，平均追蹤數量為 14 顆，其 ID switch 發生的情況是新的果實被誤判為舊果實，沒有給予新的 ID 所致，最後導致追蹤的果實數量相對真實的果實數量少，平均缺少一顆。經過資料清理第三步驟濾除 ID switch 後，ID switch 數量與誤差下降至 0 顆，可充分證明此濾除方法之有效性。

表 4-11 假果實追蹤結果做第三步驟資料清理之有效性驗證分析

		Second Type of ID Switch	Real Number	Tracking Number	Error
Exp. II-1	Before	1		15	0
	After	0		15	0
Exp. II-2	Before	1		14	-1
	After	0		15	0
Exp. II-3	Before	1		14	-1
	After	0		15	0
Exp. II-4	Before	1	15	14	-1
	After	0		15	0
Exp. II-5	Before	1		14	-1
	After	0		15	0
Exp. II-6	Before	2		13	-2
	After	0		15	0
Mean	Before	1.17		14	-1
	After	0		15	0

#### 4.4.3 假果實定位結果分析

在果實定位算法中，我們對三角測量的果實位置進行兩次位置校正，第一次使用仿射轉換地圖校正的轉換矩陣，第二次校正則是透過使用 ArUco Marker 進行三角測量迭代求解得到的常數校正，本節將依序分析 ArUco Marker 校正常數的求解結果以及兩次果實位置校正的效果。

首先說明 ArUco Marker 校正常數的 Exp. III，此實驗中共使用六組 rosbag 影像建立六張不同的 SLAM 地圖，並分成六個獨立的實驗驗證結果。每組實驗的進行方式如下說明，使用建好的地圖與 rosbag 的影片運行 Enhanced ORB-SLAM2，將 SLAM 中偵測到的 ArUco Marker 位置框儲存至 txt 文字檔，並使用三角測量計算 ArUco Marker 位置。如圖 4-12 (a)所示，黃色方格點代表三角測量得到的 ArUco Marker 位置，第一步使用仿射轉換地圖校正的轉移矩陣，將黃色方格點校正至綠色方格點。由圖中的紅色線段可明顯看出，經過地圖校正的三角測量 ArUco Marker 位置（綠色方格點）與 ArUco Marker 真實位置（黑色十字）存在誤差，

XYZ 座標均方根誤差分別為 0.266 m、0.060 m、0.868 m，ArUco Marker 三角測量的左右方向位置（x 軸）皆往相機方向位移，深度方向位置（z 軸）也相對更靠近原點，觀察到此結果後，我們針對 ArUco Marker 三角測量結果做常數校正（常數求得的方法已詳述於 3.5.2 節），使用迭代的方法求得該地圖的校正常數後，可使用此常數降低三角測量結果的誤差。圖 4-12 (b)為使用校正常數後的結果，圖中的綠色方格點與黑色十字更加靠近，XYZ 軸的座標均方根誤差下降至 0.064 m、0.059 m、0.240 m。

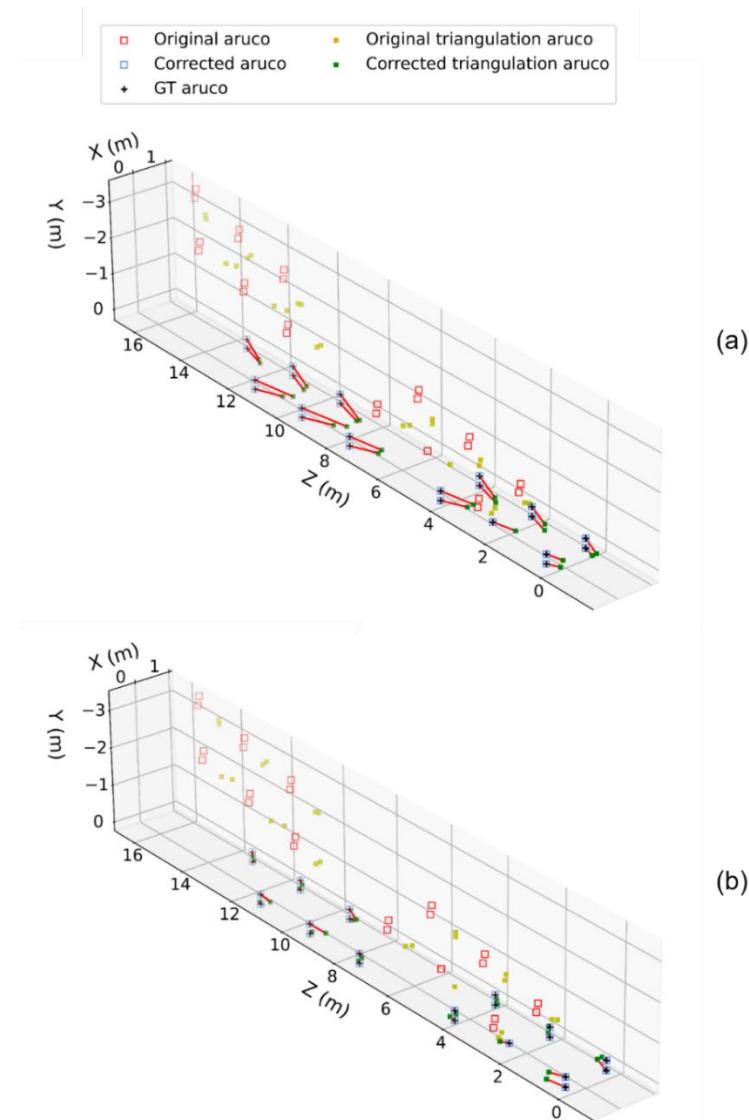


圖 4-12 Exp. III-1 之 ArUco marker 定位結果分析 ArUco  
Marker 常數校正之差異：(a)常數校正前；(b)常數校正後

表 4-12 為 Exp. III 的六組實驗對 ArUco Marker 三角測量結果做常數校正的均方根誤差比較。從平均結果來看，使用常數校正在 X 與 Z 方向的均方根誤差分別從 0.261 m 與 0.880 m 下降至 0.067 m 與 0.278 m，有顯著的下降；而常數校正對於 Y 方向的均方根誤差並無明顯的影響，校正前後的平均均方根誤差皆為 0.059 m。總均方根誤差由 0.920 m 下降至 0.293 m，由此可證實三角測量結果的誤差可經由 ArUco Marker 迭代的常數降低。

表 4-12 三角測量 ArUco Marker 位置經常數校正前後之均方根誤差結果

	Calibration	X (m)	Y (m)	Z (m)	Distance (m)
Exp. III-1	Original	0.266	0.060	0.868	0.910
	Constant Calibration	0.064	0.059	0.240	0.255
Exp. III-2	Original	0.305	0.073	1.022	1.069
	Constant Calibration	0.043	0.072	0.203	0.220
Exp. III-3	Original	0.308	0.069	1.006	1.054
	Constant Calibration	0.063	0.069	0.263	0.279
Exp. III-4	Original	0.264	0.057	0.934	0.972
	Constant Calibration	0.081	0.057	0.354	0.368
Exp. III-5	Original	0.192	0.045	0.626	0.656
	Constant Calibration	0.055	0.043	0.208	0.219
Exp. III-6	Original	0.230	0.052	0.823	0.856
	Constant Calibration	0.095	0.053	0.399	0.414
Mean	Original	0.261	0.059	0.880	0.920
	Constant Calibration	0.067	0.059	0.278	0.293
Std. Dev.	Original	0.041	0.010	0.133	0.139
	Constant Calibration	0.017	0.010	0.074	0.074

表 4-13 Exp. III 六個實驗使用 ArUco Marker 自動迭代之校正常數

	X (m)	Y (m)	Z (m)
Exp. III-1	0.278	-0.014	0.834
Exp. III-2	0.314	-0.013	1.003
Exp. III-3	0.315	-0.001	0.971
Exp. III-4	0.273	0.001	0.865
Exp. III-5	0.198	0.012	0.592
Exp. III-6	0.215	0.011	0.720
Mean	0.265	-0.0006	0.831
Std. Dev.	0.042	0.009	0.131

Exp. III 每組地圖求得的校正常數如表 4-13 所示，將此校正常數除了應用於校正應用 ArUco Marker 三角測量結果，亦可用在校正三角測量果實位置，以降低果實位置誤差。Exp. IV 將分析果實經由第一次地圖校正與第二次 ArUco Marker 常數校正的均方根誤差結果。圖 4-13 為 Exp. IV-1 的結果，使用常數做第二次位置校正前的結果如圖 4-13 (a) 所示，三角測量的果實位置為紅色圓點，將其使用第一次校正（仿射轉換地圖校正）調整後，果實位置會移動至藍色圓點，從圖中可看見校正後的果實位置（藍色圓點）與果實真實位置（黑色圓點）存在紅色線段的誤差，XYZ 軸的座標均方根誤差分別為 0.274 m、0.058 m、0.763 m。

校正常數使用表 4-13 在 Exp. III-1 求得的 0.278 m、-0.014 m、0.834 m，我們對第一次地圖校正的結果，再做第二次的常數校正，兩次校正的結果如圖 4-13 (b) 所示，兩次校正後的果實位置（藍色圓點）與果實真實位置（黑色圓點）更加靠近，紅色線段距離縮短，XYZ 軸的座標均方根誤差下降至 0.055 m、0.071 m、0.167 m。

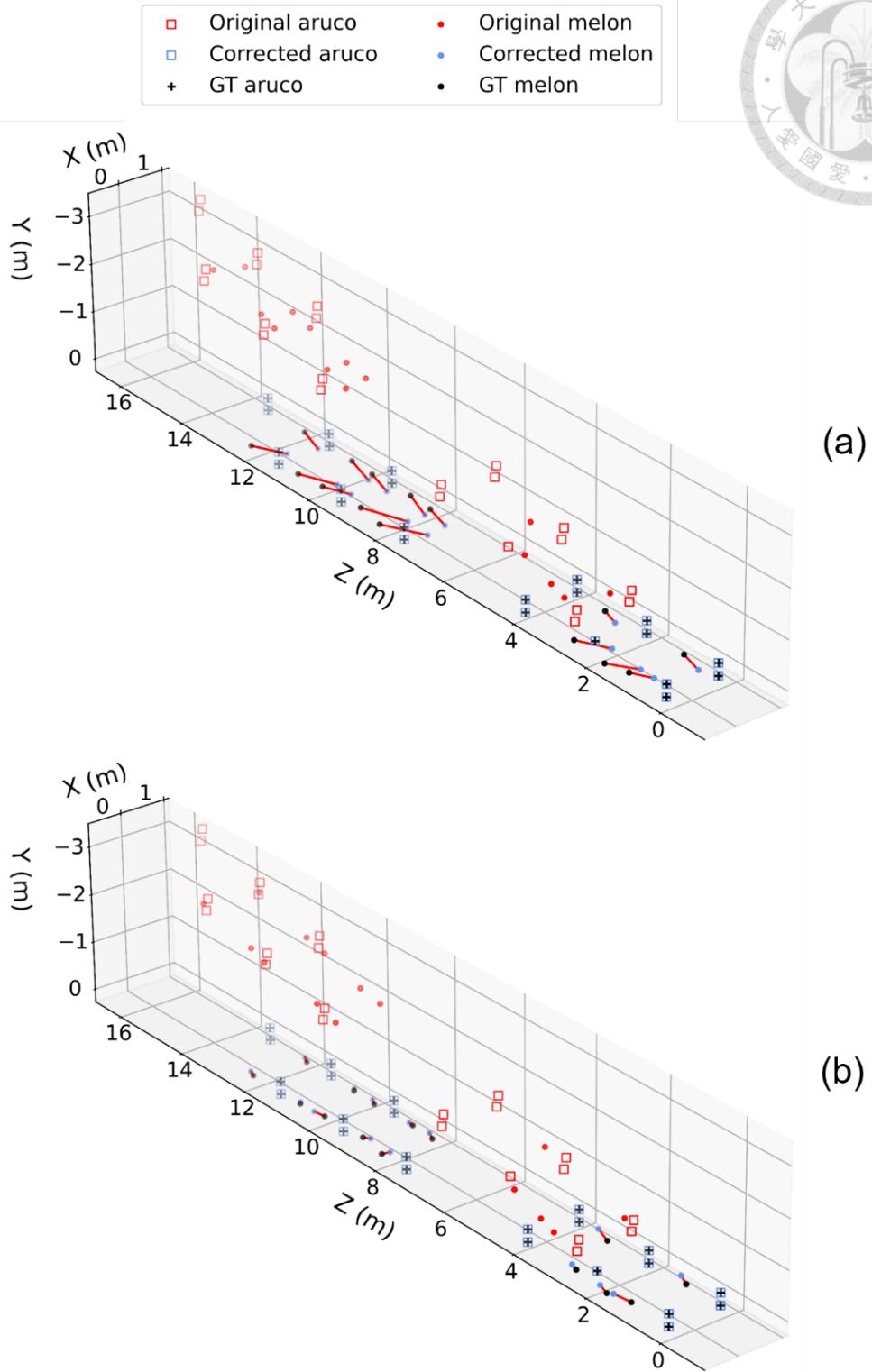


圖 4-13 Exp. IV-1 之果實定位結果分析 ArUco Marker 常數

校正之差異：(a)常數校正前；(b)常數校正後



Exp. IV的六組實驗結果分析如表 4-14，從平均均方根誤差分析，無校正的果實位置結果存在大量的誤差，總平均均方根誤差達到 2.758 公尺，而經過地圖校正後，可大幅度地下降至 0.76 公尺，證實地圖校正對於果實定位的重要性。

而常數校正方面，從平均均方根誤差來看，使用常數校正使 X 與 Z 方向的均方根誤差分別從 0.255 m 與 0.713 m 下降至 0.056 m 與 0.205 m，誤差皆有明顯下降；而 Y 方向的均方根誤差經過常數校正後，並無明顯的影響，校正前後的誤差僅相差 0.001 m，其主要原因可能為 Y 方向的原始誤差就相對較低，不需要再做常數較正。Distance 的平均均方根誤差由 0.760 m 下降至 0.223 m，證實了可透過 ArUco Marker 迭代的常數有效地降低果實三角測量的誤差。

表 4-14 假果實位置使用地圖校正與 ArUco marker 常數校正前後之均方根誤差

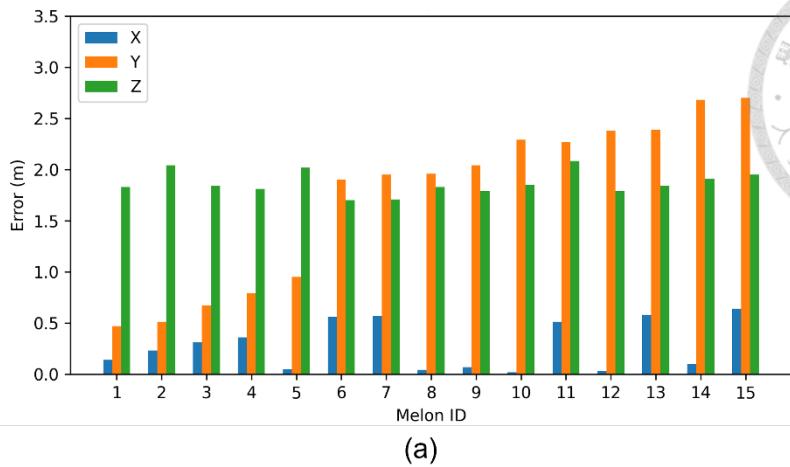
			X (m)	Y (m)	Z (m)	Distance (m)
Exp. IV-1	Original	0.362	1.899	1.87	2.69	
	Map Calibration	0.274	0.058	0.763	0.813	
	Map Calibration + Constant Calibration	0.055	0.071	0.167	0.190	
Exp. IV-2	Original	0.459	1.806	1.846	2.623	
	Map Calibration	0.287	0.059	0.798	0.850	
	Map Calibration + Constant Calibration	0.072	0.071	0.314	0.330	
Exp. IV-3	Original	0.298	1.668	1.814	2.482	
	Map Calibration	0.283	0.073	0.814	0.865	
	Map Calibration + Constant Calibration	0.058	0.075	0.241	0.259	
Exp. IV-4	Original	0.27	2.343	1.828	2.984	
	Map Calibration	0.257	0.062	0.722	0.769	
	Map Calibration + Constant Calibration	0.047	0.060	0.206	0.220	
Exp. IV-5	Original	0.395	2.084	2.132	3.007	
	Map Calibration	0.199	0.058	0.543	0.581	
	Map Calibration + Constant Calibration	0.047	0.046	0.138	0.153	
Exp. IV-6	Original	0.466	1.876	1.972	2.761	
	Map Calibration	0.230	0.063	0.637	0.680	
	Map Calibration + Constant Calibration	0.055	0.055	0.166	0.183	
Mean	Original	0.375	1.946	1.910	2.758	
	Map Calibration	0.255	0.062	0.713	0.760	
	Map Calibration + Constant Calibration	0.056	0.063	0.205	0.223	
Std. Dev.	Original	0.074	0.216	0.112	0.188	
	Map Calibration	0.031	0.005	0.095	0.100	
	Map Calibration + Constant Calibration	0.008	0.010	0.059	0.058	

圖 4-14 比較 Exp. IV-1 之果實位置誤差結果，分析使用兩種校正方法後的果實誤差結果。從圖 4-14(a)可以看出，橘色和綠色線段的誤差最大，分別為 Y 方向和 Z 方向的誤差，根據圖 4-13 的結果所示該地圖往上偏移，因此地圖校正前的果實位置在高度上逐漸向上攀爬，這對應到圖 4-14(a)中橘色的 Y 方向誤差也有逐漸增加的趨勢。然而，綠色的 Z 方向誤差保持在 1.5 公尺至 2.5 公尺之間，沒有明顯的趨勢。

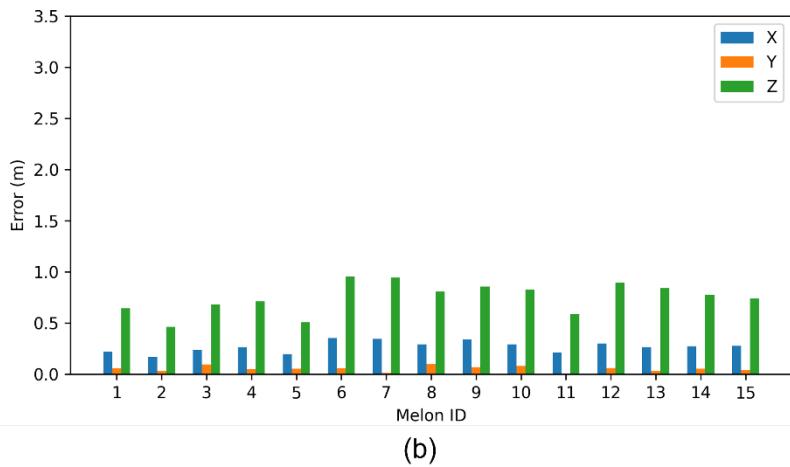
圖 4-14(b)是使用地圖校正後的果實位置誤差結果，與校正前的結果相比，15 顆果實的橘黃色高度誤差皆下降至 0.2 公尺以下，代表地圖校正可以有效地降低果實在 Y 方向的誤差。綠色的 Z 方向誤差也有明顯的下降，平均降低一公尺的果實位置誤差。而左右的 X 方向誤差則沒有明顯的變化。

圖 4-14(c)使用兩種校正方法，分別為地圖校正以及 ArUco Marker 常數校正。圖中 X 方向與 Z 方向的誤差皆大幅度的下降，Z 方向的誤差剩下三顆果實在 0.2 公尺以上，X 與 Y 方向的誤差皆下降至 0.2 公尺以下。

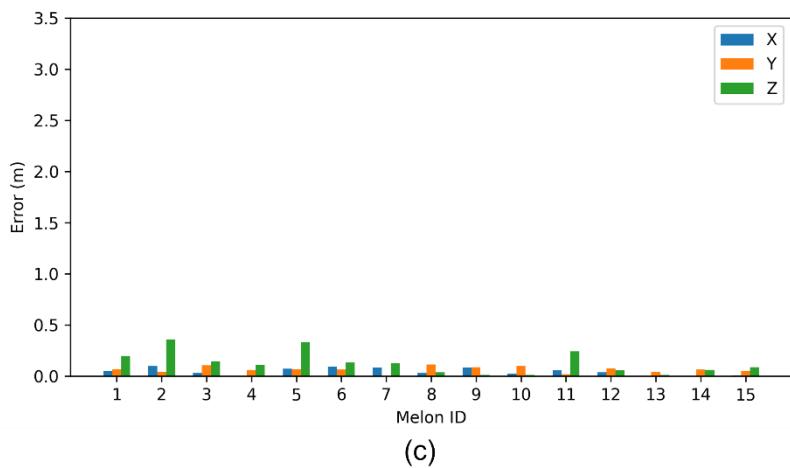
圖 4-15 為剩餘五組實驗的誤差經過兩種校正方法後的結果，與 Exp. IV-1 的結果呈現相同的趨勢。



(a)



(b)



(c)

圖 4-14 Exp. IV-1 之果實位置誤差圖：(a)原始三角測量結果；(b)仿射轉換地圖校正結果；(c)仿射轉換地圖校正與 ArUco Marker 常數校正結果

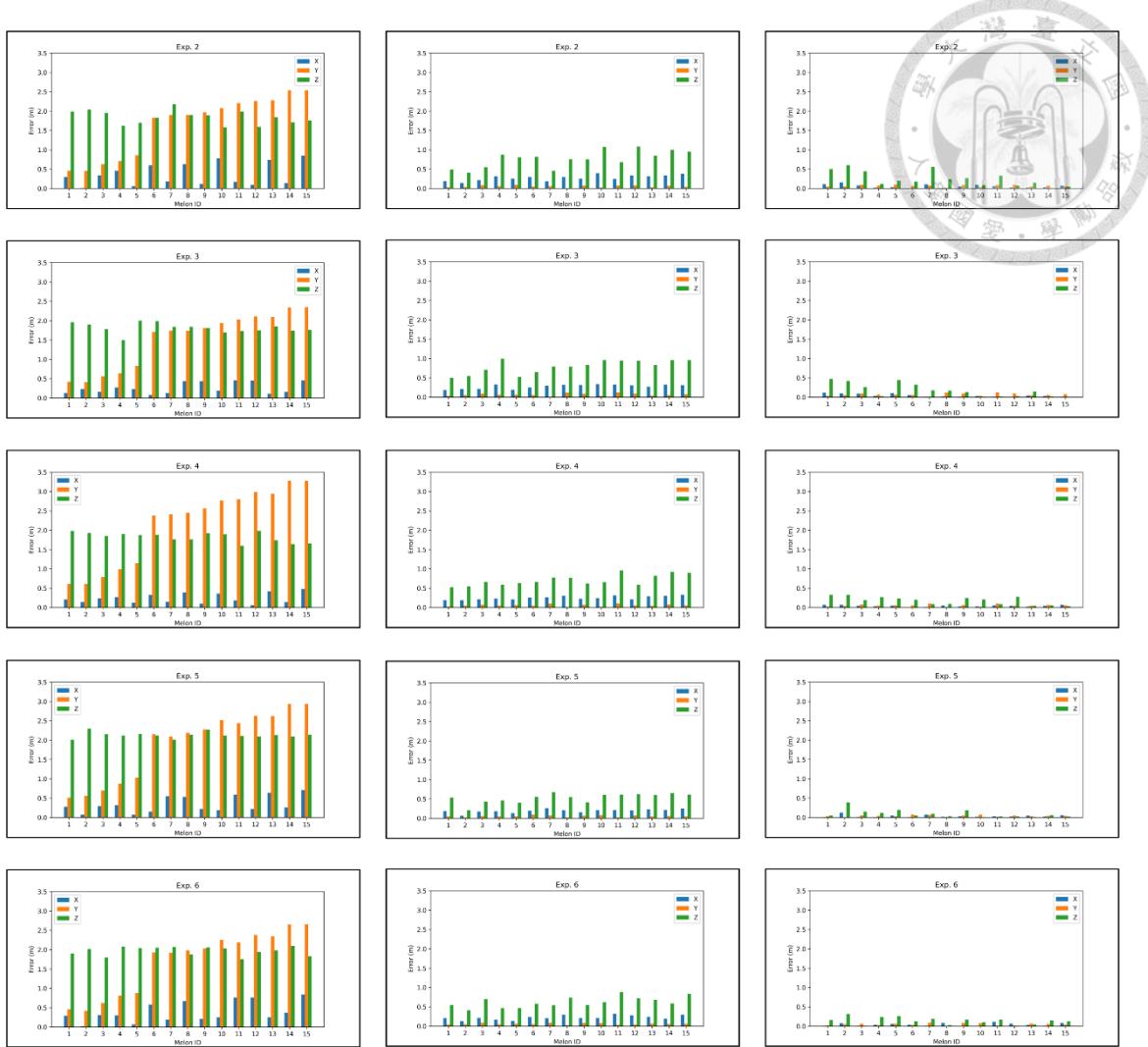


圖 4-15 Exp. IV-2~Exp. IV-6 之果實位置誤差圖

本節最後解釋如何判斷物體（ArUco 標記和果實）在無人機的右側或左側。在進行 ArUco Marker 與果實位置的 X 方向校正之前，須先判斷物體位置在無人機的右側或是左側，如果物體在右側，則需要加上 X 方向的校正常數；相反地，如果物體在左側，則需要減去 X 方向的校正常數。我們可以利用三角測量的結果來判斷物體的位置是在無人機的左側還是右側。

如圖 4-16 (a)與 (b)所示，在空間圖中，果實 ID 1 與 ID 2 的三角測量結果呈現線性關係。將這些果實的三角測量結果繪製在 2D 圖上，如圖 4-16 (c)和 (d)，可以清楚地看到在 x 和 z 方向上存在著線性關係。我們可以使用曲線擬合 (curve fitting) 的方法來求得這個線性關係的斜率。根據斜率的值，我們可以判斷果實是位於無人機的左側還是右側。當斜率大於 0 時，表示果實位於無人機的右側；反之，當

斜率小於 0 時，表示果實位於無人機的左側。如圖 4-16(c)果實 ID 1 的結果，紅色的 fitting curve 斜率為 2.35，代表果實 ID 1 位於無人機右側，在常數校正果實位置時，應加上 x 方向常數。圖 4-16(d)果實 ID 2 的結果，其中的紅色的 fitting curve 斜率為 -2.96，代表果實 ID 2 位於無人機左側，在常數校正果實位置時，應減去 x 方向常數。

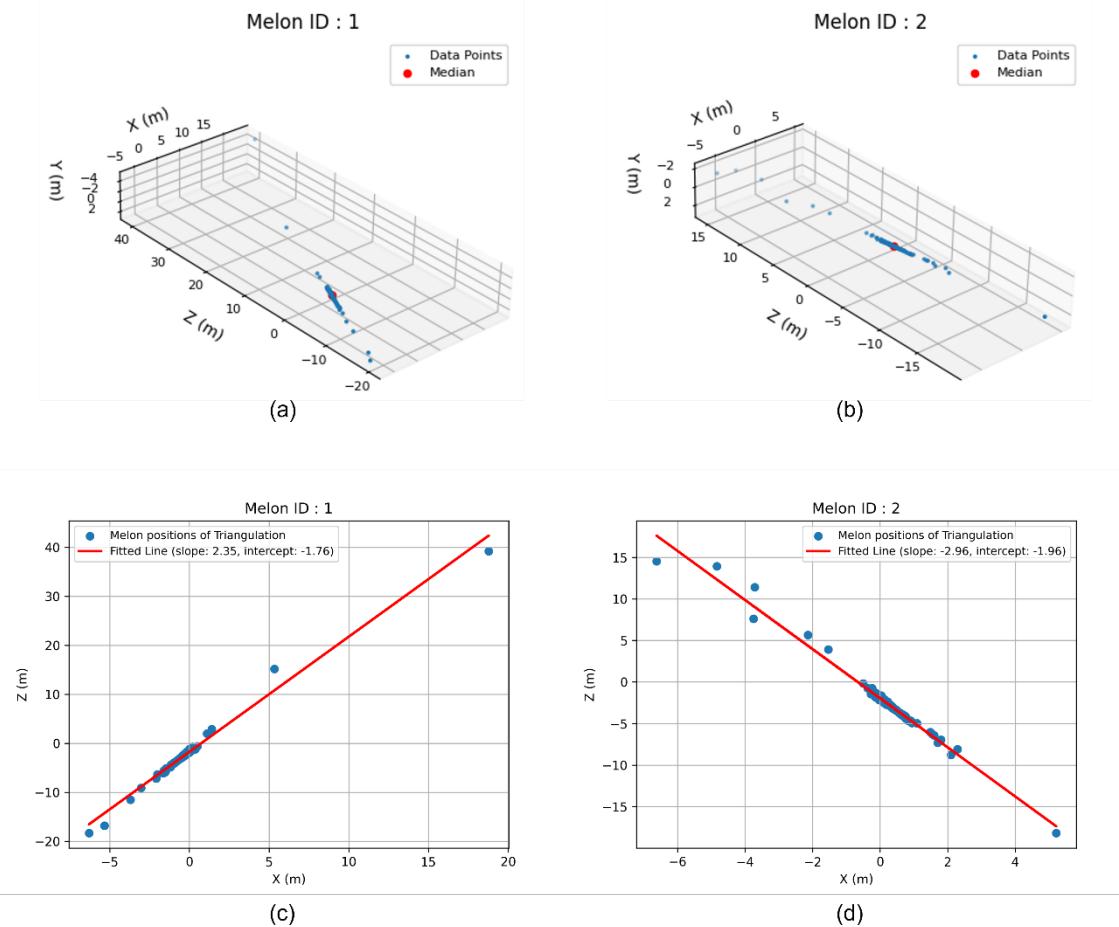


圖 4-16 三角測量結果與無人機使用斜率判斷果實位置：(a)果實於無人機右側之三角測量點於空間分布圖；(b)果實於無人機左側之三角測量點於空間分布圖；(c)果實於無人機右側之 2D 上視圖與斜率分析；(d)果實於無人機左側之 2D 上視圖與斜率分析

#### 斜率分析



#### 4.4.4 假果實大小結果分析

果實大小的實驗中，同樣使用六組假果實實驗的資料做分析，六組實驗中分別有 15 顆假果實，所有假果實的直徑皆為 14 cm。果實大小的實驗結果如表 4-15 所示，由果實大小演算法求得的果實直徑之平均均方根誤差與個別實驗中的標準差皆在 1 公分以內，驗證了果實大小算法的可靠性。

在表 4-15 中，我們也分析了不同實驗中，同一顆果實直徑的標準差，第 13 號果實得到最大的標準差為 0.84 cm，代表在不同實驗中計算到該顆果實的大小誤差較大，而在 3 號果實與 6 號果實的標準差為 0.34 cm，相對其他顆果實有更高的準確度。

表 4-15 假果實實驗之果實大小計算結果

Melon_ID	Calculated Melon Diameter (cm)						Std. Dev. (each melon)
	Exp. 1	Exp. 2	Exp. 3	Exp. 4	Exp. 5	Exp. 6	
1	15.2	13.4	13.6	14	13.1	13.3	0.70
2	15.2	14.3	13.9	13.5	13.2	13.6	0.66
3	14.3	14.1	13.3	14.3	13.9	13.9	0.34
4	13.9	13.5	14.7	15.3	15.2	14.3	0.65
5	14.4	13.9	15.3	14.2	15.1	13.8	0.57
6	13.6	12.9	13.3	13.5	13.5	12.7	0.34
7	15	15.5	14.4	13.3	14.1	13.9	0.72
8	15.5	14.3	14.4	14	13.8	13.8	0.58
9	15.3	14.6	15.3	15	14.7	14	0.45
10	15.8	15.6	14.6	14.4	14.6	15.2	0.53
11	13.8	14.5	13.4	14.4	13.7	14	0.39
12	14.7	12.8	13.1	13.6	13.2	13.3	0.61
13	12.5	15	12.9	12.9	13.8	12.9	0.84
14	13	12.7	12.9	13.7	13.3	13.9	0.43
15	13.7	14.2	13.3	14.4	13.9	14.3	0.38
Mean	14.4	14.1	13.9	14.0	13.9	13.8	0.55
RMSE	1	0.9	0.8	0.6	0.7	0.6	
Std. Dev. (each Exp.)	0.9	0.9	0.8	0.6	0.7	0.6	



#### 4.4.5 真實果實追蹤結果分析

本節使用無人機自主導航系統，使無人機於溫室自動飛行，蒐集溫室所有走道的影像資料，用於分析洋香瓜果實位置與大小。本實驗使用無人機於溫室自動飛行三次所記錄的 rosbag 檔案做分析，總飛行區域面積約為 100 平方公尺，總飛行距離約為 72 公尺，平均花費約 7 至 8 分鐘完成飛行任務。

在真實果實追蹤分析中，嘗試使用在 4.4.2 節中假果實使用的 size threshold 與 frame threshold（皆為 50），但追蹤結果不如預期，如表 4-16 所示，溫室內實際果實數量為 35 顆 (Ground Truth)，將兩個閾值皆調整為 50 時，導致果實數量大幅度下降至約 10 顆，因此在本實驗中，我們將兩個閾值皆調整至 30，追蹤數量最為接近真實果實數量。使用的閾值與假果實實驗不同的主要原因有兩個，第一點，因為真實果實周圍有葉片遮蔽的問題，導致偵測的果實框較小，因此若將 Size Threshold 調整過大，會使得大部分的果實皆被濾除在外；而第二點，由於假果實實驗中我們手持無人機取像，能在與果實最接近的高度中取像，相對地果實框會比較大，相反地，使用無人機自動飛行，會有控制上的誤差，例如飛行高度較高時，果實距離無人機的位置較遠，影像中的果實框也相對較小，此時再使用較大的閾值濾除，會導致大多數果實無法被正常地偵測到。

表 4-16 真實果實追蹤使用不同閾值之結果

Threshold	Size	Frame	Size	Frame	Size	Frame	Ground Truth
	50	50	30	30	0	0	
Exp.1		10		31		39	
Exp.2		9		32		46	35
Exp.3		13		34		53	

表 4-17 分析三步驟資料清理真實果實追蹤之結果，三組實驗中原始的 ID switch 數量平均為 11.33 次，而在經過完整的資料清理後，ID switch 次數平均下降了 9 次，大幅度地降低錯誤偵測發生的機率，最終三組實驗平均計算到的果實數量為 32.33 顆與真實數量 35 顆非常接近，也驗證了我們的追蹤算法能將無人機自主蒐集的影像資訊，應用於計算溫室洋香瓜果實數量。

表 4-17 真實果實追蹤結果使用三步驟資料清理前後之分析

		ID switch	Real Number	Tracking Number	Error
Exp.1	Before	10		39	4
	After	1		31	4
Exp.2	Before	13		46	11
	After	3	35	32	3
Exp.3	Before	11		53	18
	After	3		34	1
Mean	Before	11.33		46	11
	After	2.33		32.33	2.66

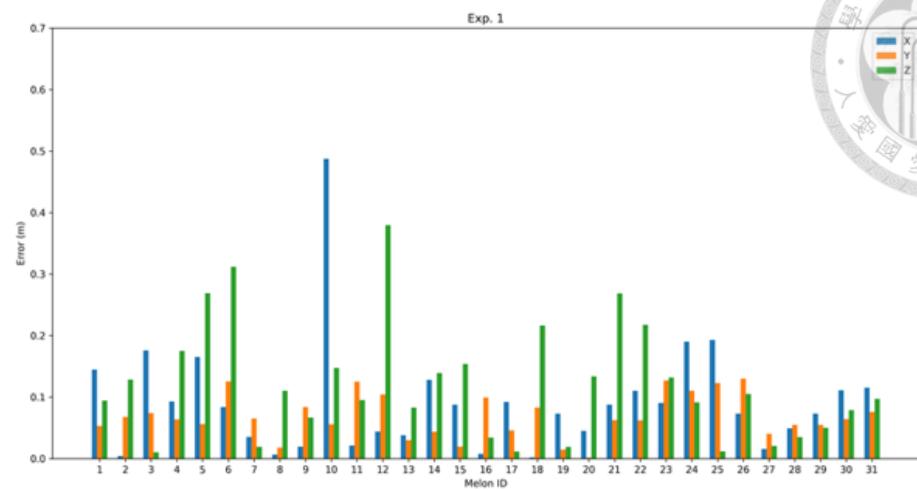
#### 4.4.6 真實果實定位結果分析

前一節中我們使用無人機自主飛行蒐集的資料計算出果實數量，本節將分析全溫室的真實果實位置計算誤差，並依照原始結果、地圖校正、地圖校正與常數校正列表分析，真實果實位置均方根誤差結果如表 4-18 所示，xyz 三方向的誤差從三個實驗中都顯示相同結果，經過地圖校正與常數校正後的位置誤差最小，最終從平均結果來看，z 方向（深度）的位置誤差最大為 20 公分，y 方向（高度）的位置誤差最小為 7 公分，而真實果實位置平均總均方根誤差為 25 公分，在 4.4.3 中分析的假果實位置誤差為 22.3 公分，與真實果實的結果相近。

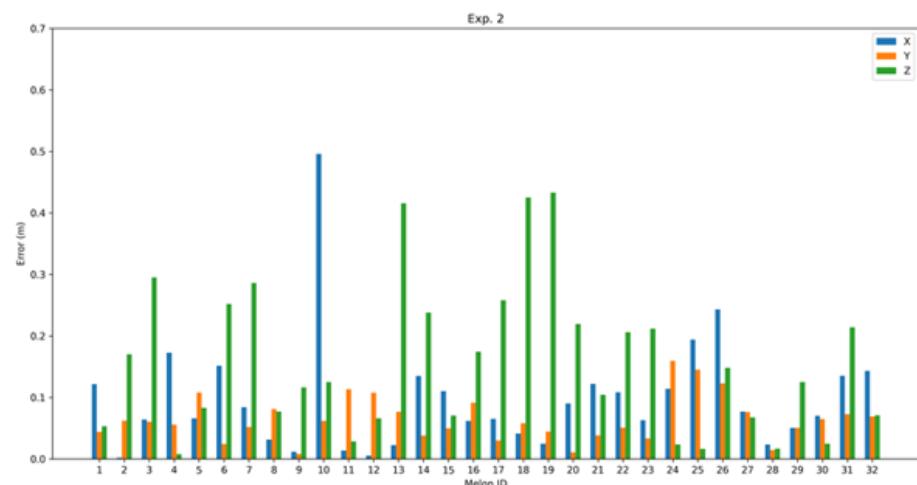
表 4-18 真實果實位置使用地圖校正與 ArUco marker 常數校正前後之均方根誤差

			X (m)	Y (m)	Z (m)	Distance (m)
Exp.1	Original		6.96	1.21	3.90	8.07
	Map Calibration		0.18	0.17	0.38	0.45
	Map Calibration + Constant Calibration		0.13	0.08	0.15	0.21
Exp.2	Original		7.05	1.23	3.87	8.14
	Map Calibration		0.19	0.17	0.41	0.49
	Map Calibration + Constant Calibration		0.13	0.07	0.20	0.25
Exp.3	Original		7.21	1.23	3.84	8.26
	Map Calibration		0.18	0.15	0.45	0.51
	Map Calibration + Constant Calibration		0.12	0.07	0.26	0.29
Mean	Original		7.07	1.22	3.87	8.15
	Map Calibration		0.18	0.16	0.41	0.48
	Map Calibration + Constant Calibration		0.13	0.07	0.20	0.25
Std. Dev.	Original		0.10	0.01	0.02	0.08
	Map Calibration		0.00	0.01	0.03	0.02
	Map Calibration + Constant Calibration		0.01	0.01	0.04	0.03

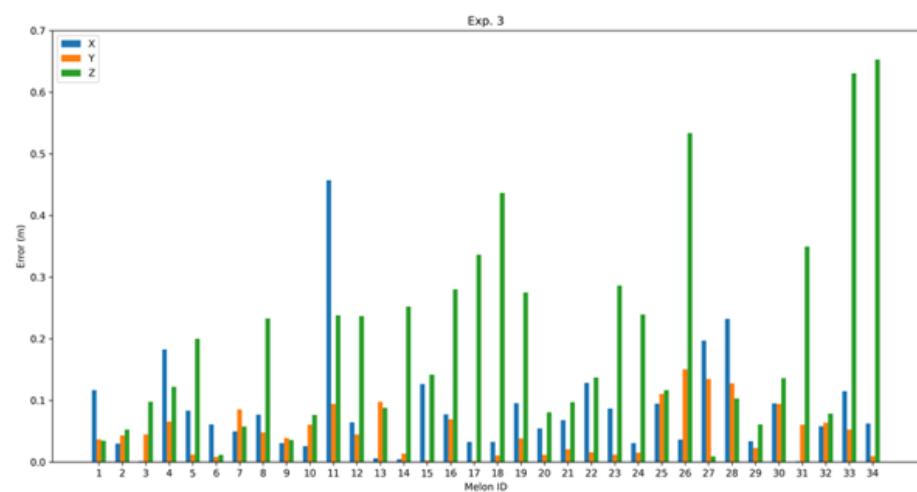
真實果實個別位置誤差圖如圖 4-17 所示，經過地圖校正與常數校正後，最大誤差在 70 公分以內，從圖中能明顯看出，代表 z 方向（深度）的綠色誤差相對其他兩個方向更大，誤差最小的則是黃色的 y 方向，結果與表 4-18 有相同的趨勢。



(a)

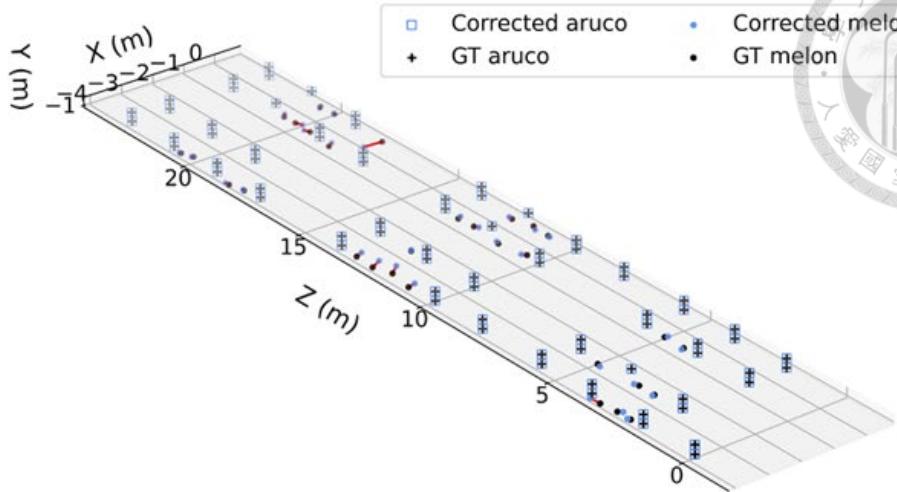
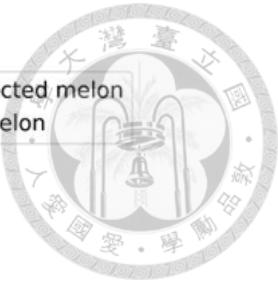


(b)

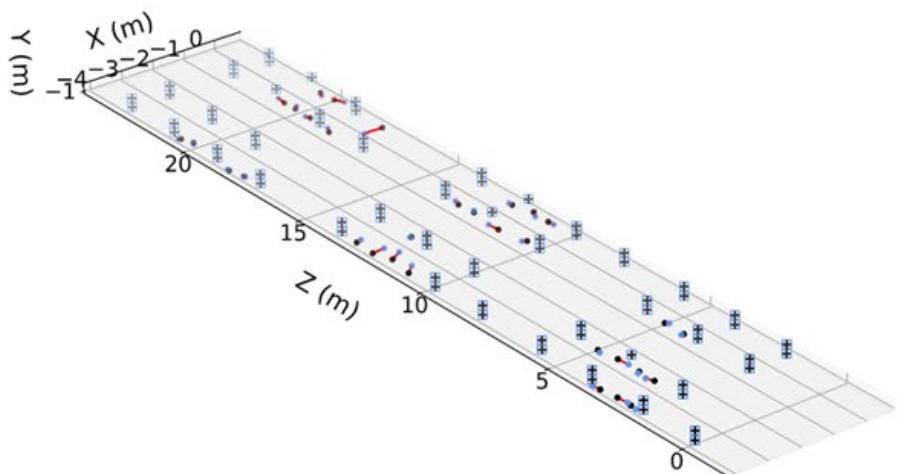


(c)

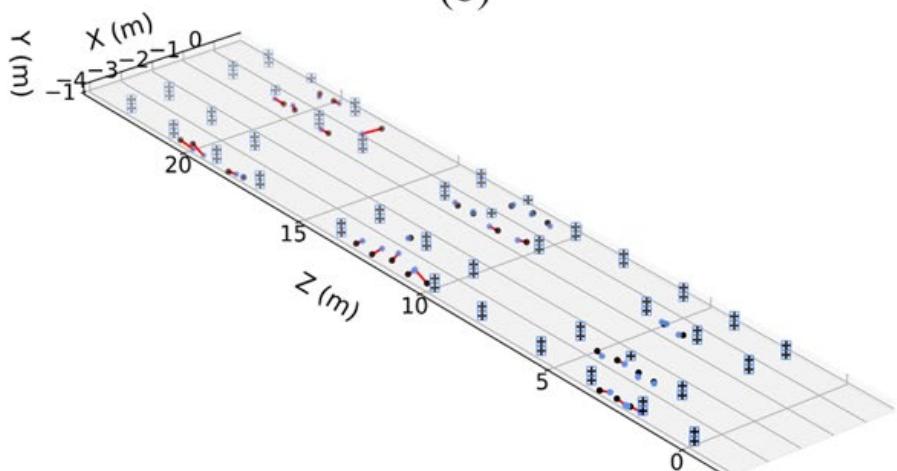
圖 4-17 真實果實個別果實位置誤差圖：(a) Exp.1；(b) Exp.2；(c) Exp.3



(a)



(b)



(c)

圖 4-18 真實果實空間位置圖：(a) Exp.1；(b) Exp.2；(c) Exp.3



#### 4.4.7 真實果實大小結果分析

真實果實實驗中，溫室共有 35 顆真實果實，在三角測量計算果實位置時，也一同求出果實的大小，三組實驗的果實大小結果如表 4-19 所示。其中有部分結果顯示 Not Tracked 代表在果實追蹤時，就沒有偵測到，果實大小算法需基於追蹤結果才能計算，因此將無法測量果實大小。從平均的結果來看，三組實驗的果實直徑平均誤差約在 3 cm，相比假果實實驗的 1 cm 誤差，真實果實的結果相對較差，其中原因與 4.4.5 中果實濾除閾值需降低的原因相同，與果實受葉片遮蔽以及無人機飛行控制高度有關。

表 4-19 真實果實實驗之果實大小計算結果

Melon ID	Ground Truth (cm)	Exp.1		Exp.2		Exp.3	
		Calculated Diameter (cm)	Error (cm)	Calculated Diameter (cm)	Error (cm)	Calculated Diameter (cm)	Error (cm)
1	0.16	0.12	0.04	0.10	0.06	0.12	0.04
2	0.16	0.11	0.05	0.10	0.06	0.12	0.04
3	0.15	0.13	0.03	0.14	0.01	0.13	0.02
4	0.16	0.12	0.04	0.12	0.04	0.11	0.04
5	0.16	0.12	0.04	0.11	0.04	0.12	0.04
6	0.16	0.13	0.03	0.13	0.03	0.12	0.04
7	0.11	0.12	0.01	Not Tracked		Not Tracked	
8	0.13	0.12	0.01	0.13	0.00	0.12	0.01
9	0.12	0.10	0.02	0.09	0.04	0.11	0.01
10	0.15	0.12	0.04	0.12	0.04	0.12	0.04
11	0.14	0.12	0.02	0.10	0.03	0.11	0.03
12	0.15	0.12	0.04	0.10	0.05	0.11	0.05
13	0.14	0.14	0.01	0.11	0.03	0.11	0.03
14	0.14	0.12	0.02	0.10	0.04	0.13	0.02
15	0.13	0.19	0.06	0.13	0.01	Not Tracked	
16	0.14	0.11	0.03	0.09	0.05	0.11	0.04
17	0.14	0.12	0.02	0.12	0.02	0.12	0.02
18	0.16	Not Tracked		Not Tracked		Not Tracked	



19	0.15	0.12	0.03	0.10	0.05	0.11	0.04
20	0.15	0.12	0.03	0.11	0.04	0.12	0.03
21	0.15	Not Tracked		0.11	0.04	0.13	0.02
22	0.14	0.11	0.04	0.10	0.04	0.11	0.04
23	0.14	0.13	0.01	0.13	0.01	0.15	0.01
24	0.14	0.13	0.01	0.09	0.06	0.16	0.02
25	0.14	0.10	0.04	0.10	0.05	0.11	0.04
26	0.14	0.13	0.02	0.15	0.00	0.10	0.05
27	0.14	Not Tracked		Not Tracked		0.12	0.01
28	0.14	0.10	0.04	0.09	0.05	0.10	0.04
29	0.14	0.12	0.02	0.11	0.04	0.12	0.03
30	0.13	0.12	0.01	0.11	0.03	0.10	0.03
31	0.13	0.10	0.03	0.11	0.02	0.11	0.02
32	0.14	0.10	0.03	0.10	0.04	0.09	0.05
33	0.15	0.11	0.04	0.11	0.04	0.10	0.05
34	0.16	0.11	0.05	0.11	0.06	0.12	0.04
35	0.15	0.11	0.04	0.11	0.05	0.10	0.05
Mean		0.12	0.03	0.11	0.04	0.12	0.03
Std. Dev.		0.02	0.01	0.01	0.02	0.01	0.01

# 第五章 結論與建議



## 5.1 結論

本研究成功建立自主巡航無人機系統，並應用無人機飛行紀錄的資訊分析溫室中洋香瓜的果實數量與位置。具體達成目標條列如下：

### 無人機自主巡航系統：

1. 新增存取地圖的功能至 Enhanced ORB-SLAM2
2. 證實 Enhanced ORB-SLAM2 的定位效果優於 ORB-SLAM2，當溫室存在陰影時，ORB-SLAM2 之可工作光照度範圍約落在  $1\text{ k} \sim 100\text{ k lux}$ ，而 Enhanced ORB-SLAM2 之可工作光照度範圍約落在  $1\text{ k} \sim 150\text{ k lux}$
3. 驗證無人機導航系統可於溫室中以直線、倒退繞圈、轉彎繞圈、全溫室繞圈、全溫室 S 型等方式飛行，所有飛行軌跡之均方根誤差範圍在 0.6 公尺以內

### 地圖校正：

1. 仿設變換地圖校正後，MapAruco 與 Ground truth aruco 的均方根誤差在三個軸的誤差皆為 0，此校正方法使地圖的 MapAruco 與人工量測的位置完全貼合
2. 原始地圖平均每個 frame 的特徵點數量為 683，而使用校正地圖則下降至 385，相對不穩定，選擇不直接校正地圖，而是使用求得的變換矩陣校正原始地圖得到的軌跡

### 果實演算法分析：

1. 以 RGB 影像訓練 YOLOv4 洋香瓜果實偵測模型，模型之 mAP 為 0.96
2. 建立兩步驟的閾值篩選（果實框大小閾值、果實 ID 出現數量閾值）以及第三步驟的新果實誤判成舊果實的偵測方法，處理 DeepSORT 的追蹤結果，使假果實實驗中之 ID switch 數量由平均 5.83 次，下降至 0 次，真實果實之 ID switch 數量則是由平均 11.33 次，下降至 2.33 次



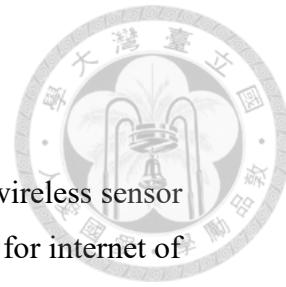
3. 透過三角測量算法，以及 DeepSORT 的果實框、ORB-SLAM2 的相機外參、棋盤格的相機內參等資訊，開發果實定位算法
4. 使用地圖校正的變換矩陣以及迭代 ArUco Marker 求得的校正常數，調整三角測量的果實定位結果，經過兩種果實位置校正後，假果實實驗中的位置均方根誤差從 2.758 公尺下降至 0.223 公尺；真實果實實驗的位置均方根誤差由 8.15 公尺下降至 0.25 公尺
5. 果實大小演算法在假果實求得之直徑均方根誤差在 1 cm 以內，而真實果實之直徑均方根誤差約 3 cm，原因與作物遮蔽果實以及無人機控制精準度有關



## 5.2 建議

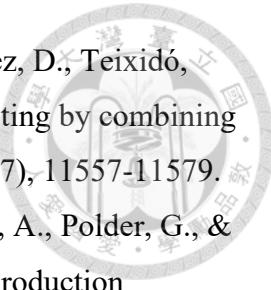
本研究之飛行系統與果實算法已建構完成，並初步測試不同飛行環境、方法，以及分析果實追蹤與定位之結果。對未來可能之發展條列如下：

1. 本研究在溫室放置的 ArUco Marker 數量較為密集，使用的 ArUco Marker 大小為 19.5 公分 x 19.5 公分的正方形，最適合 SLAM 的 ArUco Marker 數量與尺寸可再做實驗分析比較
2. ArUco Marker 雖可強化定位效果，但容易受葉片遮蔽，可能降低無人機定位穩定度，或影響需使用 ArUco Marker 做校正的果實位置
3. 溫室的光影變化使 ORB-SLAM2 系統的追蹤度下降，若要使用 ORB-SLAM2，可以嘗試做影像增強的方式降低光照度的影響，陰影的強烈特徵可開發算法濾除，但仍需考量即時運算之需求
4. 小型無人機可能受溫室內之風扇氣流影響，可針對外部因素做分析
5. 需再評估無人機的 Wi-Fi 通訊距離，DJI Tello 的通訊距離為 100 m，但在智慧溫室中有各式訊號干擾，可考慮加入 Wi-Fi 強波器，以增強通訊穩定性
6. 目前導航路徑是由人工設置，此部分可使用設計紀錄特定的地圖點，自動產出導航路徑，降低人工介入，降低使用難度
7. 果實定位仍存在誤差，可嘗試使用深度學習方法取代三角測量計算果實位置
8. 現行系統中之地圖校正僅能以 Enhanced ORB-SLAM2 實現，可嘗試用建圖的軌跡，以及真實世界的建圖軌跡去做地圖校正，如此能實現在沒有 ArUco Marker 的環境中做地圖校正
9. 本研究僅使用單一無人機進行溫室飛行，未來可考量加入無人機群飛系統 (swarm drone)，提升果實分析之效率
10. 無人機於溫室飛行時，可能會遇到溫室的工作人員，未來可加入一偵測模型，在偵測到工作人員時，使無人機在空中懸停
11. 倘若未來要建立一個溫室供純視覺定位之無人機自動飛行檢測作物，要避免晴天的太陽光直射溫室，在溫室出現陰影特徵會大幅降低定位穩定度

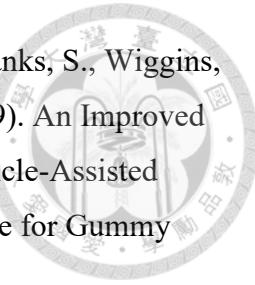


## 參考文獻

- Abbasi-Kesbi, R., Nikfarjam, A., & Nemati, M. (2020). Developed wireless sensor network to supervise the essential parameters in greenhouses for internet of things applications. *Iet Circuits Devices & Systems*, 14(8), 1258-1264.
- Amador, G. J., & Hu, D. L. (2017). Sticky Solution Provides Grip for the First Robotic Pollinator. *Chem*, 2(2), 162-164.
- Aquino, A., Barrio, I., Diago, M.-P., Millan, B., & Tardaguila, J. (2018). vitisBerry: An Android-smartphone application to early evaluate the number of grapevine berries by means of image analysis. *Computers and Electronics in Agriculture*, 148, 19-28.
- Araus, J. L., & Kefauver, S. C. (2018). Breeding to adapt agriculture to climate change: affordable phenotyping solutions. *Current opinion in plant biology*, 45, 237-247.
- Aslan, M. F., Durdu, A., Sabancı, K., Ropelewska, E., & Gueltekin, S. S. (2022). A Comprehensive Survey of the Recent Studies with UAV for Precision Agriculture in Open Fields and Greenhouses. *Applied Sciences-Basel*, 12(3), Article 1047.
- Bailey, T., & Durrant-Whyte, H. (2006). Simultaneous localization and mapping (SLAM): part II. *IEEE Robotics & Automation Magazine*, 13(3), 108-117.
- Barbedo, J. G. A. (2019). A review on the use of unmanned aerial vehicles and imaging sensors for monitoring and assessing plant stresses. *Drones*, 3(2), 40.
- Das Choudhury, S., Bashyam, S., Qiu, Y., Samal, A., & Awada, T. (2018). Holistic and component plant phenotyping using temporal image sequence. *Plant Methods*, 14(1), 1-21.
- Durrant-Whyte, H., & Bailey, T. (2006). Simultaneous localization and mapping: part I. *IEEE Robotics & Automation Magazine*, 13(2), 99-110.
- Elazab, A., Ordóñez, R. A., Savin, R., Slafer, G. A., & Araus, J. L. (2016). Detecting interactive effects of N fertilization and heat stress on maize productivity by remote sensing techniques. *European Journal of Agronomy*, 73, 11-24.
- Feng, J., Liu, G., Wang, S., Zeng, L., & Ren, W. (2012). A novel 3D laser vision system for robotic apple harvesting. 2012 Dallas, Texas, July 29-August 1, 2012.



- Font, D., Pallejà, T., Tresanchez, M., Runcan, D., Moreno, J., Martínez, D., Teixidó, M., & Palacín, J. (2014). A proposal for automatic fruit harvesting by combining a low cost stereovision camera and a robotic arm. *Sensors*, 14(7), 11557-11579.
- Fonteijn, H., Afonso, M., Lensink, D., Mooij, M., Faber, N., Vroegop, A., Polder, G., & Wehrens, R. (2021). Automatic Phenotyping of Tomatoes in Production Greenhouses Using Robotics and Computer Vision: From Theory to Practice. *Agronomy-Basel*, 11(8), Article 1599.
- Gongal, A., Silwal, A., Amatya, S., Karkee, M., Zhang, Q., & Lewis, K. (2016). Apple crop-load estimation with over-the-row machine vision system. *Computers and Electronics in Agriculture*, 120, 26-35.
- He, J. Q., Harrison, R. J., & Li, B. (2017). A novel 3D imaging system for strawberry phenotyping. *Plant Methods*, 13(1), 1-8.
- Hoornweg, D., & Pope, K. (2017). Population predictions for the world's largest cities in the 21st century. *Environment and Urbanization*, 29(1), 195-216.
- Hughes, A., Askew, K., Scotson, C. P., Williams, K., Sauze, C., Corke, F., Doonan, J. H., & Nibau, C. (2017). Non-destructive, high-content analysis of wheat grain traits using X-ray micro computed tomography. *Plant Methods*, 13(1), 76.
- Huletski, A., Kartashov, D., & Krinkin, K. (2015, 9-14 Nov. 2015). Evaluation of the modern visual SLAM methods. 2015 Artificial Intelligence and Natural Language and Information Extraction, Social Media and Web Search FRUCT Conference (AINL-ISMW FRUCT).
- Iddio, E., Wang, L., Thomas, Y., McMorrow, G., & Denzer, A. (2020). Energy efficient operation and modeling for greenhouses: A literature review. *Renewable and Sustainable Energy Reviews*, 117, 109480.
- Jakob Engel, T. S. o., and Daniel Cremers. (2014). LSD-SLAM: Large-Scale Direct Monocular SLAM.
- Jimenez-Berni, J. A., Deery, D. M., Rozas-Larraondo, P., Condon, A. G., Rebetzke, G. J., James, R. A., Bovill, W. D., Furbank, R. T., & Sirault, X. R. (2018). High throughput determination of plant height, ground cover, and above-ground biomass in wheat with LiDAR. *Frontiers in Plant Science*, 9, 237.
- Kalantar, A., Edan, Y., Gur, A., & Klapp, I. (2020). A deep learning system for single and overall weight estimation of melons using unmanned aerial vehicle images. *Computers and Electronics in Agriculture*, 178, Article 105748.



- Kalischuk, M., Paret, M. L., Freeman, J. H., Raj, D., Da Silva, S., Eubanks, S., Wiggins, D. J., Lollar, M., Marois, J. J., Mellinger, H. C., & Das, J. (2019). An Improved Crop Scouting Technique Incorporating Unmanned Aerial Vehicle-Assisted Multispectral Crop Imaging into Conventional Scouting Practice for Gummy Stem Blight in Watermelon. *Plant Dis*, 103(7), 1642-1650.
- Krul, S., Pantos, C., Frangulea, M., & Valente, J. (2021). Visual SLAM for Indoor Livestock and Farming Using a Small Drone with a Monocular Camera: A Feasibility Study. *Drones*, 5(2), Article 41.
- López-Granados, F., Torres-Sánchez, J., De Castro, A.-I., Serrano-Pérez, A., Mesas-Carrascosa, F.-J., & Peña, J.-M. (2016). Object-based early monitoring of a grass weed in a grass crop using high resolution UAV imagery. *Agronomy for Sustainable Development*, 36(4), 67.
- Lachlan Dowling, T. P., Isaac Hook, Hao Tang, Ying Tan,, & Will Glenn, R. R. U. (2018). Accurate indoor mapping using an autonomous unmanned aerial vehicle (UAV).
- Li, P., Lee, S.-h., & Hsu, H.-Y. (2011). Review on fruit harvesting method for potential use of automatic fruit harvesting systems. *Procedia Engineering*, 23, 351-366.
- Liu, S., Whitty, M., & Cossell, S. (2015). Automatic grape bunch detection in vineyards for precise yield estimation. 2015 14th IAPR International Conference on Machine Vision Applications (MVA).
- Liyang, Liu, P., Li, B., & Yu, X. (2018). Intelligent Control System of Cucumber Production in the Greenhouse Based on Internet of Things. In *Cloud Computing and Security* (pp. 395-406).
- Lobao, L., & Meyer, K. (2001). The great agricultural transition: Crisis, change, and social consequences of twentieth century US farming. *Annual Review of Sociology*, 27, 103-124.
- Maskey, M. L., Pathak, T. B., & Dara, S. K. (2019). Weather Based Strawberry Yield Forecasts at Field Scale Using Statistical and Machine Learning Models. *Atmosphere*, 10(7), 378.
- Mehta, S. S., & Burks, T. F. (2014). Vision-based control of robotic manipulator for citrus harvesting. *Computers and Electronics in Agriculture*, 102, 146-158.
- Miao, J., & Niu, L. (2016). A survey on feature selection. *Procedia Computer Science*, 91, 919-926.

- Mur-Artal, R., Montiel, J. M. M., & Tardos, J. D. (2015). ORB-SLAM: A Versatile and Accurate Monocular SLAM System. *Ieee Transactions on Robotics*, 31(5), 1147-1163.
- Mur-Artal, R., & Tardos, J. D. (2017). ORB-SLAM2: An Open-Source SLAM System for Monocular, Stereo, and RGB-D Cameras. *Ieee Transactions on Robotics*, 33(5), 1255-1262.
- Olenskyj, A. G., Sams, B. S., Fei, Z. H., Singh, V., Raja, P. V., Bornhorst, G. M., & Earles, J. M. (2022). End-to-end deep learning for directly estimating grape yield from ground-based imagery. *Computers and Electronics in Agriculture*, 198, Article 107081.
- Peirson, B. (2013). Wilhelm Ludvig Johannsen (1857-1927). *Embryo Project Encyclopedia*.
- Qin, H., Meng, Z., Meng, W., Chen, X., Sun, H., Lin, F., & Ang, M. H. (2019). Autonomous Exploration and Mapping System Using Heterogeneous UAVs and UGVs in GPS-Denied Environments. *IEEE Transactions on Vehicular Technology*, 68(2), 1339-1350.
- Rejeb, A., Abdollahi, A., Rejeb, K., & Treiblmaier, H. (2022). Drones in agriculture: A review and bibliometric analysis. *Computers and Electronics in Agriculture*, 198, Article 107017.
- Roldán, JoosSEN, Sanz, Cerro, & Barrientos. (2015). Mini-UAV Based Sensory System for Measuring Environmental Variables in Greenhouses. *Sensors*, 15(2).
- Roldán, J. J., Garcia-Aunon, P., Garzón, M., De León, J., Del Cerro, J., & Barrientos, A. (2016). Heterogeneous Multi-Robot System for Mapping Environmental Variables of Greenhouses. *Sensors*, 16(7).
- Rublee, E., Rabaud, V., Konolige, K., & Bradski, G. (2011, 6-13 Nov. 2011). ORB: An efficient alternative to SIFT or SURF. 2011 International Conference on Computer Vision.
- Sankaran, S., Khot, L. R., Espinoza, C. Z., Jarolmasjed, S., Sathuvalli, V. R., Vandemark, G. J., Miklas, P. N., Carter, A. H., Pumphrey, M. O., & Knowles, N. R. (2015). Low-altitude, high-resolution aerial imaging systems for row and field crop phenotyping: A review. *European Journal of Agronomy*, 70, 112-123.
- Shafeikhani, A., Kadam, S., Fritschi, F. B., & DeSouza, G. N. (2017). Vinobot and vinoculer: Two robotic platforms for high-throughput field phenotyping. *Sensors*, 17(1), 214.

- Shi, Q., Liu, D., Mao, H., Shen, B., Liu, X., & Ou, M. (2019). *Study on Assistant Pollination of Facility Tomato by UAV* 2019 ASABE Annual International Meeting.
- Song, P., Wang, J. L., Guo, X. Y., Yang, W. N., & Zhao, C. J. (2021). High-throughput phenotyping: Breaking through the bottleneck in future crop breeding. *Crop Journal*, 9(3), 633-645.
- Tangarife, H. I., & Díaz, A. E. (2017, 18-20 Oct. 2017). Robotic applications in the automation of agricultural production under greenhouse: A review. 2017 IEEE 3rd Colombian Conference on Automatic Control (CCAC).
- Trachsel, S., Dhliwayo, T., Gonzalez Perez, L., Mendoza Lugo, J. A., & Trachsel, M. (2019). Estimation of physiological genomic estimated breeding values (PGEBV) combining full hyperspectral and marker data across environments for grain yield under combined heat and drought stress in tropical maize (*Zea mays* L.). *PLoS One*, 14(3), e0212200.
- Vakilian, K. A., & Massah, J. (2012). Design, development and performance evaluation of a robot to early detection of nitrogen deficiency in greenhouse cucumber (*Cucumis sativus*) with machine vision. *Int. J. Agric. Res. Rev*, 2, 448-454.
- Virlet, N., Sabermanesh, K., Sadeghi-Tehran, P., & Hawkesford, M. J. (2016). Field Scanalyzer: An automated robotic field phenotyping platform for detailed crop monitoring. *Functional Plant Biology*, 44(1), 143-153.
- Xiong, B., Wang, B., Xiong, S., Lin, C., & Yuan, X. (2019). 3D morphological processing for wheat spike phenotypes using computed tomography images. *Remote Sensing*, 11(9), 1110.
- Yang, W., Feng, H., Zhang, X., Zhang, J., Doonan, J. H., Batchelor, W. D., Xiong, L., & Yan, J. (2020). Crop phenomics and high-throughput phenotyping: past decades, current challenges, and future perspectives. *Molecular Plant*, 13(2), 187-214.
- Zhao, Y., Gong, L., Liu, C., & Huang, Y. (2016). Dual-arm Robot Design and Testing for Harvesting Tomato in Greenhouse. *IFAC-PapersOnLine*, 49(16), 161-165.
- Zheng, C., Abd-Elrahman, A., & Whitaker, V. (2021). Remote Sensing and Machine Learning in Crop Phenotyping and Management, with an Emphasis on Applications in Strawberry Farming. *Remote Sensing*, 13(3), 531.
- Zhou, X., Zheng, H., Xu, X., He, J., Ge, X., Yao, X., Cheng, T., Zhu, Y., Cao, W., & Tian, Y. (2017). Predicting grain yield in rice using multi-temporal vegetation

indices from UAV-based multispectral and digital imagery. *ISPRS Journal of Photogrammetry and Remote Sensing*, 130, 246-255.

