

# 2020 年臺灣國際科學展覽會 優勝作品專輯

作品編號 100024

參展科別 工程學

作品名稱 三槽式微生物燃料電池及不同尿液中微生物產電效率影響之探討

得獎獎項 大會獎：三等獎  
出國正選代表

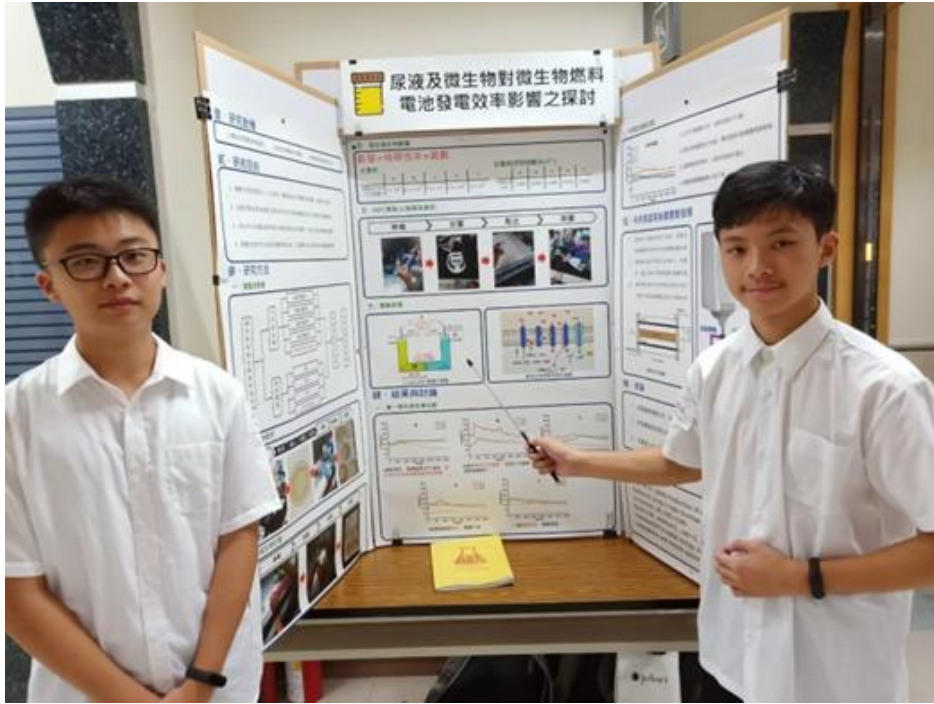
就讀學校 臺北市立麗山高級中學

指導教師 周楚洋、林獻升

作者姓名 蘇詩元、路峰棋

關鍵詞 SLAM、YOLO、機器人

## 作者簡介



路峰棋(左)：我是路峰棋，自高中以來，我一直在臺北市立麗山高中學習許多科學方法。就因為我和我的夥伴喜歡探索新事物，所以就一起進行了微生物燃料電池的研究。小時候，我想參加科展，但沒有機會，直到上了高中，跟著老師們的腳步，我才能圓夢。我非常感謝指導老師們能夠不辭辛勞的協助我們，給了我們這麼難得的機會來參加國際科展。

蘇詩元(右)：從小我熱就愛學習新知，尤其是自然科學以及最新科技的新知，因此我喜歡看科學性期刊或節目，如 Discovery, Discovery science, National Geography。國中時，我夢寐以求的就是進入實驗室做實驗，可是在國中畢業前一直都無法實現，因此，做專題，實現了我長年以來的夢想。在專題當中我不僅學到了如何進行研究，還學到了當遇到問題時如何解決。我們專題之前已通過青少年培育計畫以及中學生獎助計畫(因需與青培擇一放棄)，希望能藉此比賽，繼續累積經驗，與國內外一流高手競爭，提升自己的實力，發現自己的不足。

## 摘要

本研究以可辨識、定位、記憶空間物品的眼鏡，建構可於非固定環境與人協同作業的搬運機器人。眼鏡結合 SLAM 即時定位與地圖構建及 YOLO 即時物體偵測，透過 SLAM 平面偵測建構空間中平面資訊，作為物品座標及物理資訊的依據參考，再經過深度學習訓練後的 YOLO 模型達到辨識物品，結合兩者建立空間與物品的依存關係。透過網路即時傳輸環境資訊與指令，使機器人可以知道使用者移動的物品位置，並自主於環境中搬運物品，實現更有效的人機協同作業。本作品運用範圍，包括工地建材搬運、居家物品整理等，使人與機器人能並肩工作，提升人與機器人整合互動。

## Abstract

Our study constructs a moving robot that can work with people in a non-fixed environment with glasses that can identify, locate, and memorize space objects. The space object memory device combined with SLAM that used for map constructing and locating and YOLO object detection, through the SLAM plane detection to construct the spatial information, as a reference for the object coordinates and physical information, and then through the deep learning training YOLO model to achieve identification, combined both establishing a relationship between space and objects. Instantly transmitting environmental information and instructions through the network, so that the robot can know the location of the objects moved by the user and carry the objects in the environment independently, so as to achieve more effective human-machine collaborative work. The scope of application of this work, including the moving of building materials and housekeeping homework, so that people and robots can work altogether, enhance the integration of people and robots.

# 一、前言（含研究動機、目的）

## （一）、研究動機

現代社會生活步調快速，事情多且繁雜，再加上人口高齡化，老化引發的記憶力衰退，健忘症與失智症人口都在不斷的增加。如果我們能將所有看過的物品資訊都記錄在電腦中，要尋找物品時，可以呼叫機器人去物品位置尋找，並搬運物品到我們面前，這樣對於工作繁忙的人或老人都很有幫助。

希望以眼鏡作為使用者平時攜帶的裝置，眼鏡裝置可以即時且方便的紀錄環境與使用者的操作等相關資訊，並記憶空間中物品的種類與位置，將資訊與使用者命令傳給搬運機器人，讓機器人可以正確到達物品位置，搬運物品到達指定位置。

記憶物品方式比較市面上現有解決辦法，將物品裝上藍芽發送裝置，用裝置取得物品位置，或外裝多台攝影機交叉比對影像定位的方式，與本作品的方法相比之下，本作品在記憶、運用的方式上，更符合人性化的使用，應用上更具有實用性。裝置與搬運機器人的傳輸方式，可以使機器人在非固定路線或環境的情況下，也能夠找到物品並搬運，擴展人與機器人更有效的互動的方式，使機器人可以處理更多事情，對其他領域能夠使工作效率提升。

## （二）、研究目的

- 1.研究運用 SLAM 系統建構三維空間的原理，並用於建構環境座標系，與定位使用者當下的位置與視角。
- 2.研究物體偵測技術，運用 YOLO 方法，達成準確辨識物品種類、得知物品在圖片中的位置等資訊。
- 3.結合三維空間建模與物體偵測，以達到在座標系中定位與記錄物品資訊。
- 4.傳遞物品資訊與指令給機器人，使機器人搬運物品到達指定位置。

## 二、研究方法或過程

表1. 研究設備及器材		
名稱	版本及型號	用途
(AR 眼鏡)	因市面上 AR 眼鏡價格昂貴，且開發功能難以使用，所以使用手機和 VR 眼鏡代替 AR 眼鏡的功能	
iPhone 手機	iPhone XR      iOS 系統 13.1.3 iPhone 6s Plus    iOS 系統 13.1.3	APP 安裝與執行的設備
VR 頭套	VRBOX 5PLUS	將手機固定在頭上，達到使用手機取代 AR 眼鏡
Mac 電腦	macOS 系統 10.14.2 Beta 版	iOS 系統手機的開發和測試
Xcode 開發軟體	10.1 版	Swift 程式語言的整合開發環境
Swift 程式語言	4.2 版	用於撰寫 APP
ARKit 框架	2 版	撰寫 AR 處理功能
SceneKit 物理引擎	1 版	撰寫 3D 物件與碰撞環境
Core ML 框架	2 版	影像辨識模型能在手機上使用
Darknet 框架	YOLOV2	官方辨識基礎框架
Python 程式語言	3.6.6 版	Python 程式語言
WSL	1709 版	於 Windows 系統中安裝 Linux
麥克納姆輪	Mecanum Wheel 100mm * 4	機器人移動輪子
底盤	ReMi SYS	機器人裝載物品
MyRIO	1900 主機	機器人控制裝置
編碼直流馬達	36mm DC Motor * 4	機器人輪子轉動控制

## (一)、資料探討

### 1.人類的記憶

#### (1).記憶事情

人類記憶事情主要分成三個不同階段，從感官（如：視覺）接收到訊息，傳入下視丘形成短期記憶（持續約三十秒）的編碼過程；到海馬迴將部分短期記憶轉換為長期記憶（持續數天以上），在大腦皮層進行紀錄的儲存過程；以及提取儲存內容的檢索過程。因此，只要其中一部分未完成，就可能無法記住事情。

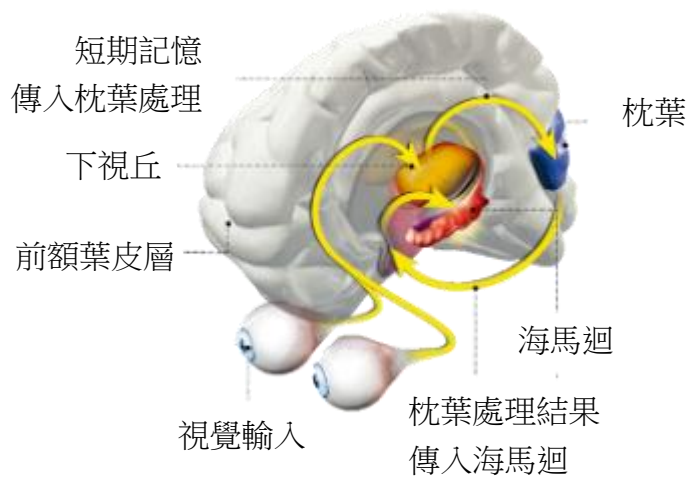


圖1. 大腦的記憶流程

#### (2).記憶空間與認知位置

大腦記憶環境空間的方式，是網格細胞（Grid Cells）在腦中，以六角形網格的方式做到建立座標系統、建構認知地圖。透過不同區域網格細胞的反應，使位置細胞能瞭解自身在認知地圖中的位置。人透過移動雙眼得知不同邊界的前後距離，使網格細胞建構認知地圖，運用位置細胞確定要如何移動。



圖2. 大腦對於環境的記憶

位置細胞場域

網格細胞場域

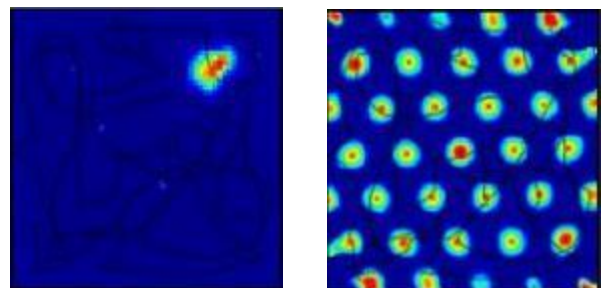


圖3. 位置細胞與網格細胞

## 2.AR 眼鏡（AR Glasses）

AR 最廣泛的定義為一九九七年由羅納德·阿祖瑪（Ronald Azuma）提出，主要說明 AR 的功能包含：結合虛擬與現實、即時互動與 3D 定位。也就是將電腦的資訊疊合到現實世界，讓使用者看現實空間時，能看到虛擬物件的資訊或物件。

### （1）.AR 眼鏡介紹

AR 眼鏡的發展早在 1968 年，科學家 Ivan Sutherland 開發了第一套頭戴式 AR 顯示裝置，當時的裝置只能在現實中顯示單純的框線圖。多年之後隨著硬體設備與計算能力的提升，2013 年 Google 公司推出 AR 頭戴式顯示器 Google Glass，用於顯示各種即時資訊給使用者得知。現今 AR 最常應用於手機上，因為手機的研發、作業系統、以及生態系等，相較於 AR 眼鏡更完整，但手機對於 AR 可能只是一個過渡期，未來 AR 眼鏡很有可能取代手機成為 AR 常用的攜帶式裝置。根據 IDC 的 AR 市場營收分析，可見未來 AR 在市場中是很有發展價值的一種技術。



圖4. Google Glass 示意圖

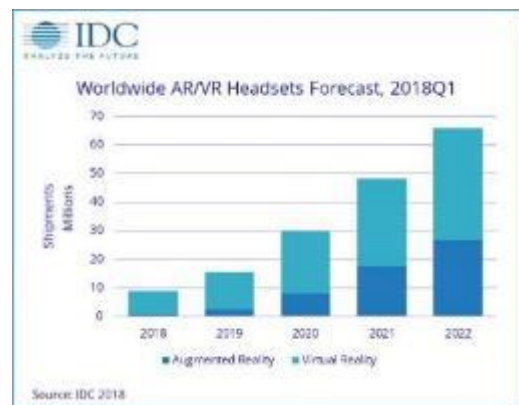


圖5. IDC AR/VR 分析圖

### （2）.AR 眼鏡功能

近年來許多公司積極研發 AR 眼鏡，因為 AR 眼鏡攜帶與顯示的方式，改善了使用裝置的便利性；AR 眼鏡上的感測元件與使用者的位置與視角同步，能夠取得使用者的頭部動作與視角；配合處理元件，計算與分析資訊，讓使用者透過鏡片看現實景象時，能看到虛擬資訊與現實影像結合後的結果，達到融合現實提供更準確有用的資訊。

### (3) .搬運機器人

搬運機器人如：自動導引車(AGV)，主要功能為移動至指定位置幫助人搬運大型、重型或特殊貨物的機器人，軟體需要紀錄物品位置、感知環境及移動路徑規劃，硬體需要移動裝置與搬運平台。機器人搬運可達到提高搬運效率、完成人無法搬運的工作及減少人工成本。

### (二)、作品架構

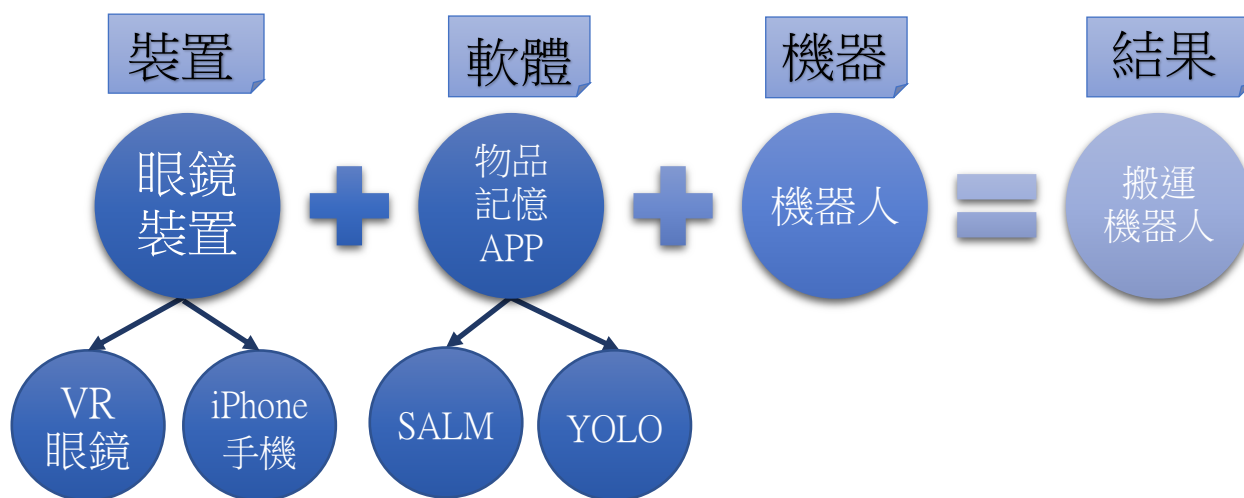


圖6. 作品軟硬體架構圖

#### 1.硬體設備

##### (1) .AR 眼鏡的現有問題

現在市售的 AR 眼鏡價格普遍偏高，售價（三萬到十萬台幣不等），且硬體規格與作業系統不統一，造成開發者開發 AR 眼鏡 APP 較為困難，因此 AR 眼鏡的附加功能較少，整體性價比不如手機，市場普及率低。

##### (2) .AR 眼鏡替代方案

本研究 AR 眼鏡設備採用 VR 眼鏡及 iPhone 手機替代，達到符合 AR 眼鏡攜帶、顯示、運算的功能，價格比起市售 AR 眼鏡，可降到 1000 元以下。且市面上可執行 AR 程式功能的手機，相當普遍，開發資源也較為廣泛。



圖7. 使用者攜帶樣式



### (3) .替代方案存在的問題

- a.視野大小：使用時畫面只顯示視角六分之一的影像。
- b.重量：重量超過四百公克，以眼鏡的方式攜帶在頭上，難以長時間舒適的使用。
- c.3D 暈眩：視角快速改變，虛擬影像與真實影像未能即時同步，會造成暈眩問題。

### (4) .搬運機器人

以 MyRIO 作為機器人的控制裝置，操控麥克納姆輪依公式轉動輪子方向與速度，達到車身不用旋轉即可到達指定位置，以 PWM 方式控制機器人直流減速編碼馬達，使機器人可以較好的加減速，正確到達物品定位的位置。

## 2.軟體開發

使用蘋果公司的 Mac 電腦，在作業系統 macOS 中安裝 Anaconda 與 Xcode 開發環境，Anaconda 下載 Python 程式語言執行 Tensorflow 套件，將輸入的訓練資料經過機器學習 (Machine Learning) 訓練出影像辨識模型，用 Xcode 撰寫 Swift 程式語言，使用 ARKit、SceneKit、Core ML 套件，並加入影像辨識模型，以開發 APP (application 應用程式) 的方式來製作作品的軟體部分，此方法在開發上相較於 AR 眼鏡的開發環境，能有更多的學習資源，在相容上也較完整。機器人運用網頁接收 APP 資訊，以 LabVIEW 裝 Web service 套件接收指令，傳入 myRIO 執行機器人移動。

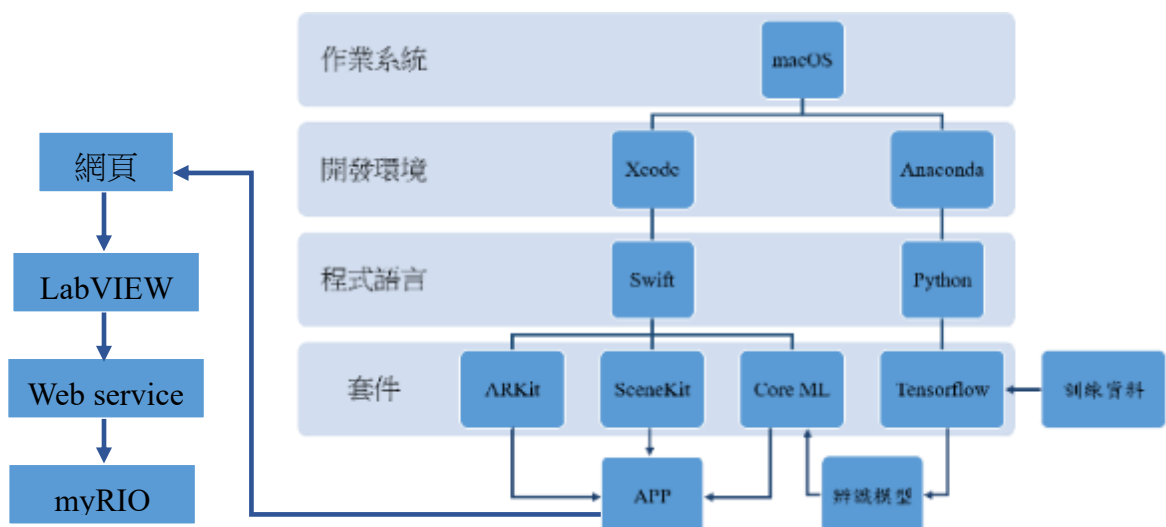


圖8. 軟體開發架構

- (1) .辨識：使用者平時攜帶此作品在環境中移動時，就能因為視角與位置的轉變，給

予辨識環境所需要的資訊，進而辨識出使用者在空間中的資訊、空間平面資訊與圖中物品資訊。

- (2) 記錄：將辨識到的平面資訊與圖中物品資訊統整並以座標化的方式記錄於裝置中，以便於使用者搜尋物品時，能取得空間中物品位置與資訊以及最後的影像。
- (3) 搜尋：使用者操作耳機的音量按鍵選擇要搜尋的物品種類，就能檢索記錄中種類符合的資訊，並將訊息顯示在 AR 眼鏡上，讓使用者可以呼叫機器人去搬運所選的物品，或是依據其他條件提醒機器人去搬運。
- (4) 導航：機器人取得物品資料後，分析記錄中物品的空間位置，根據機器人當下位置及角度，移動到達物品位置，並回傳資訊顯示於眼鏡上，讓使用者能知道機器人與物品位置。
- (5) 搬運：機器人到達物品位置後，能夠搬運物品到達指定位置或使用者當下的位置，搬運過程確保物品與人的安全。

(三)、系統裝置設計及規劃

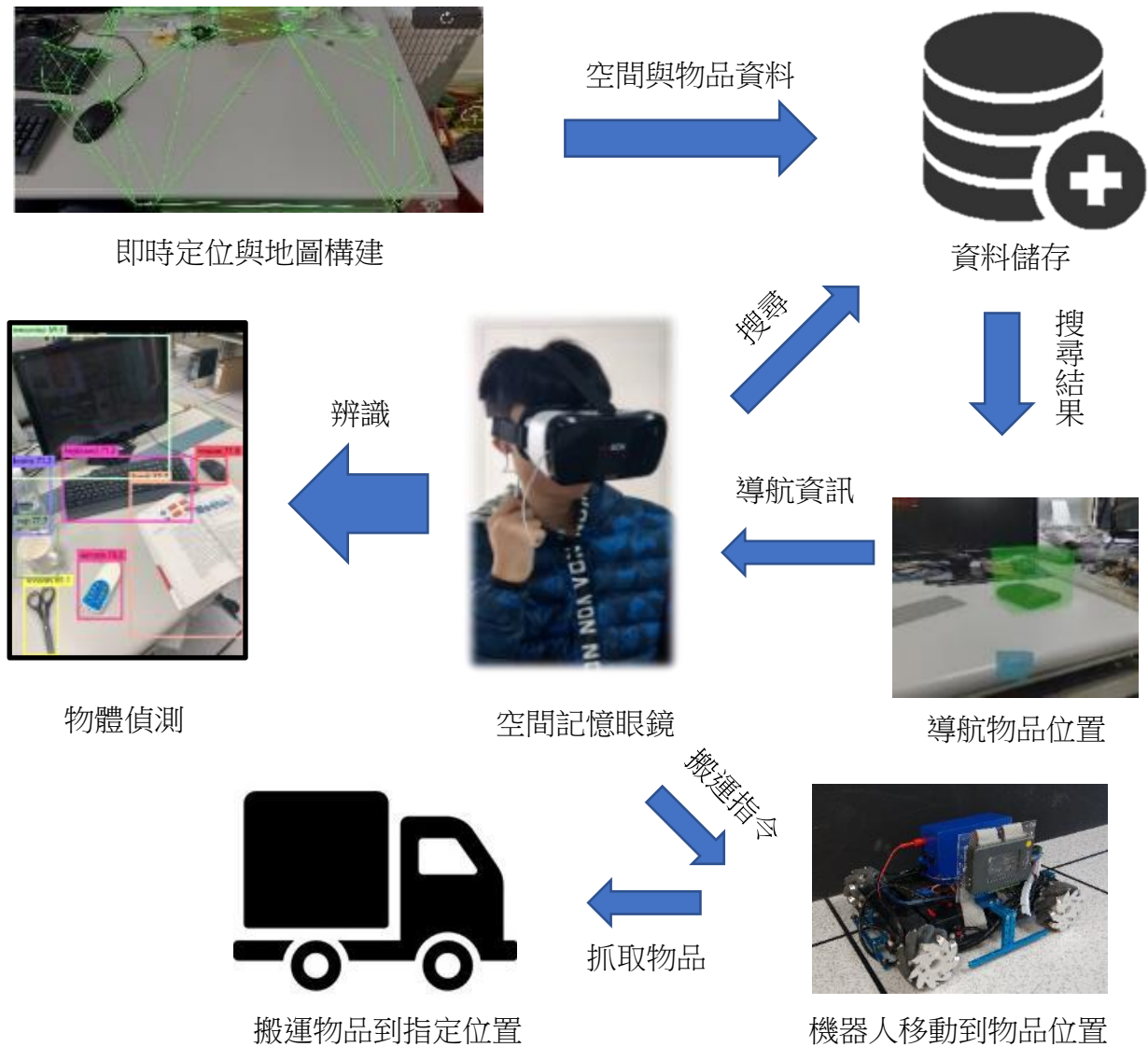


圖9. 空間記憶眼鏡與機器人搬運運作流程圖

系統的運作方式模仿大腦記憶事物的方式，依循以辨識、記錄、搜尋、導航及機器人搬運五大流程，達到大腦記憶事物的編碼、處存、檢索的功能，機器人了解空間與物品位置，搬運物品至指定位置的能力，針對運用的各項技術及演算法說明如下。

1. 辨識：辨識包含兩個主要方向，「辨識空間」與「辨識物品」，流程如圖 11 所示。

辨識空間以 SLAM（即時定位與地圖構建 Simultaneous localization and mapping）系統當作辨識方法，SLAM 系統的感測器分為用 ToF（飛時測距 Time-of-Flight）的雷射感測方式與用視覺的單目（單鏡頭相機）或雙目（多鏡頭相機）感測方式，比較資料如表 1。

辨識物品資訊是運用影像辨識，根據對物品的聚焦程度與實際需要，依序分為三個層次：從確定圖像內容的分類（Classification），將圖片依事先確定好的類別做分類；到框出圖像物品位置的物體偵測（Object Detection），依序對不同位置做分類；以及能畫出物品輪廓的圖像分割（Image Segmentation），增加對邊緣與前後景的處理，如圖 12。

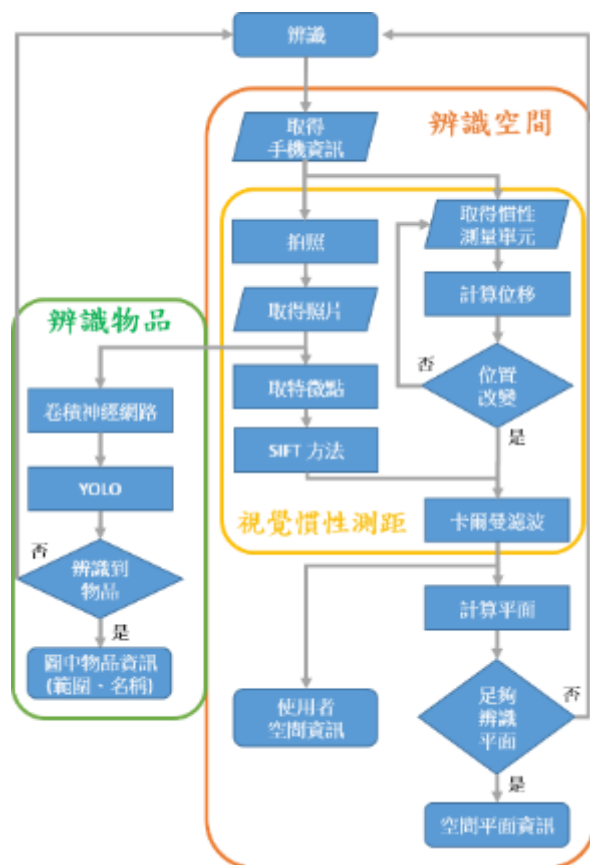
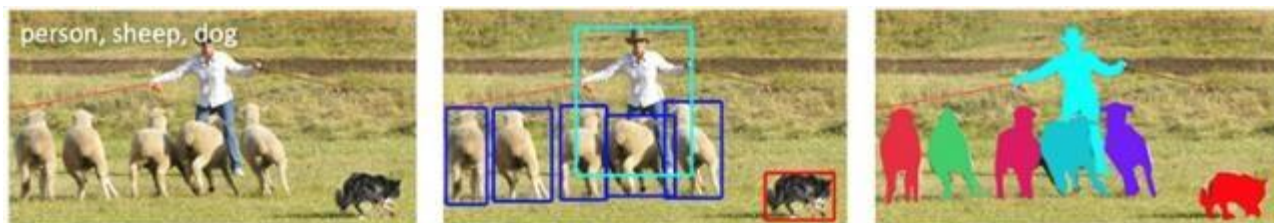


圖10. 辨識流程圖

表2. SLAM 系統感測器比較表 來源：組員整理

感測器類型	雷射	視覺	
		單目（單鏡頭）	雙目（多鏡頭）
相機	飛時測距 ToF	單目（單鏡頭）	雙目（多鏡頭）
原理	以自身與環境障礙物間的距離作為感測依據。	以多張影像與移動訊息作為感測依據。	多個相機的單張影像與相機之間的距離作為感測依據。
優點	精準度高，環境影響小，速度快，計算量小。	結構簡單，成本低，技術資源多。	易取得深度信息，識別環境大。
缺點	價格昂貴，缺少影像資訊，技術較少。	易受環境影響，誤差大，缺少深度資訊。	高計算資源，速度慢，體積大。

由表 1 得知，視覺 SLAM 包含影像資訊，可直接用於物體偵測，以手機的單鏡頭與內建 IMU（慣性測量單元 Inertial measurement unit）設備，就能做到單目 SLAM 系統，不須額外加裝其他設備，且單目 SLAM 系統只需較少的處理器計算資源，就能達到辨識空間資訊，符合小型設備應用。



分類 (Classification)      物體偵測 (Object Detection)      圖像分割 (Image Segmentation)  
圖11.辨識物品資訊三個層次示意圖

取得相機拍攝的影像後，以手機處理器達到即時辨識物品，且需要有圖像中物品的位置做到定位物品在空間中的位置，考量處理速度與實際需求，選擇以物體偵測作為辨識物品的方式。

### (1).辨識空間

與人類的空間感知模式相似，使用影像與位移達到辨識空間、定位自身位置，但本研究單目 SLAM 系統地圖架構以單鏡頭拍攝影像並加入 IMU 數據建構環境地圖，運用 VIO（視覺慣性測距 Visual Inertial Odometry）技術測得自身方向與位移量，並以座標系的方式紀錄環境及自身的位置。簡要說明如下：

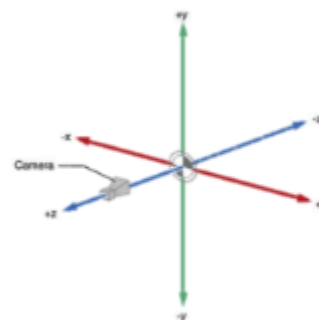


圖12.座標系圖示

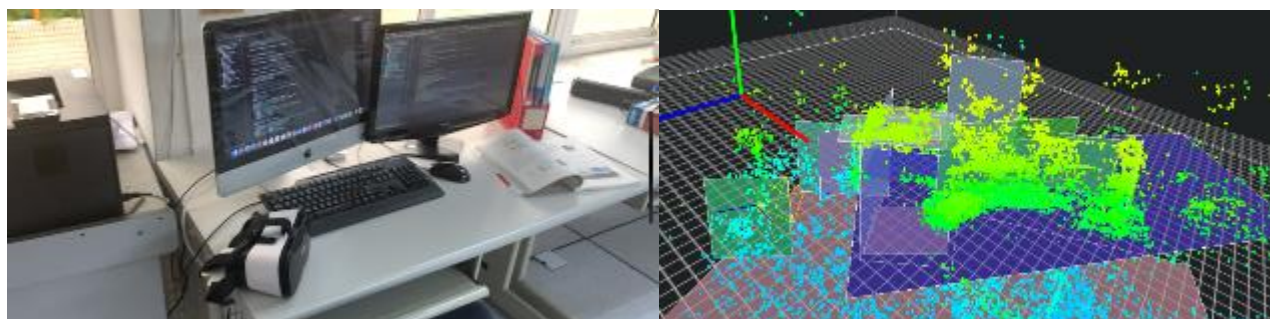


圖13.單目 SLAM 系統地圖的特徵點分布圖

a.單目 SLAM 系統:SLAM 系統主要功能為確定自身的位置與建構環境地圖。單目 SLAM 運用 VIO 技術以單鏡頭透過 SIFT (尺度不變特徵轉換 Scale-Invariant Feature Transform) 取得影像位移預估值，並與 IMU 的位移預估值經過卡爾曼濾波 (Kalman Filter) 互相校正以取得較精準的設備運動資訊。將空間資訊匯入 SceneKit 物理引擎的座標系中(圖 13)，以右手座標系繪製環境地圖。當使用者開啟 APP，眼鏡就會將當下的位置設為地圖世界座標系的原點，移動時持續繪製環境地圖並定位。

b.VIO 技術：VIO 技術是整合影像資訊與 IMU 資訊，做到測量自身移動量，VIO 技術有兩種整合方式，緊耦合 (tightly-coupled) 與鬆耦合 (loosely-coupled)，兩種方式都使用影像與 IMU 的位移預估值資訊，兩種方法分析如 (表 2)，因為資料穩定度低且需要快速計算，採用鬆耦合方式取得位移。

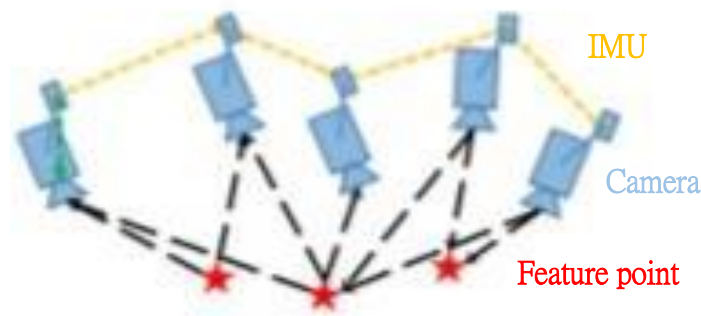


圖14.VIO 方法示意圖

表3. VIO 技術位移預估方法 來源：組員整理

方法	緊耦合 (tightly-coupled)	鬆耦合 (loosely-coupled)
原理	同時優化影像與 IMU 的位移預估，建立統一位移預估值。	將影像的位移預估值與 IMU 的位移預估作為單獨模組，最後統整出位移預估值。
計算複雜度	高	中
優點	精度高	演算法簡單、誤差容忍度大

鬆耦合 VIO 是將鏡頭拍攝到的每幀畫面經過 SIFT 方法，根據特徵點 (feature point) 在每幀畫面間位置的變化與 IMU 等資訊，進行卡爾曼濾波比較與運算，最終將兩種不準確的預測結果，統整出較為準確的設備運動資訊。

c.SIFT 方法：SIFT 方法的用途是從兩張影像取得自身的影像位移預估，先將圖片灰階後，取圖元（pixel）點值與周圍差距較大的或邊緣的交會點作為特徵點，之後匹配兩張圖片的特徵點，以取得多個點的向量，整合多個向量並去除誤差，最後計算出預估位移結果。

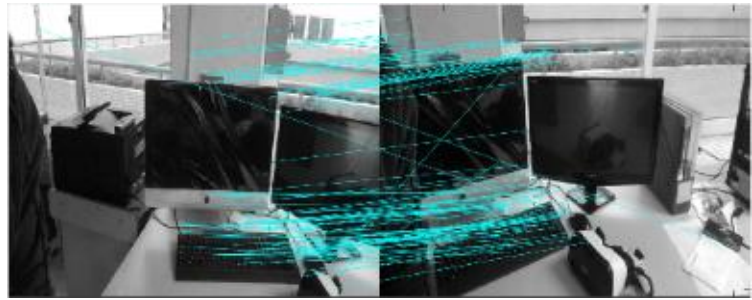


圖15. SIFT 方法 示意圖

d.卡爾曼濾波：卡爾曼濾波

是一種遞歸濾波器，能從一系列不完整且包含雜訊的測量結果，估計出動態系統的狀態。卡爾曼濾波主要用於將影像的預估位移結果，與 IMU 的預估位移結果經過公式運算 如圖 16，最終得出較精準的設備運動資訊。

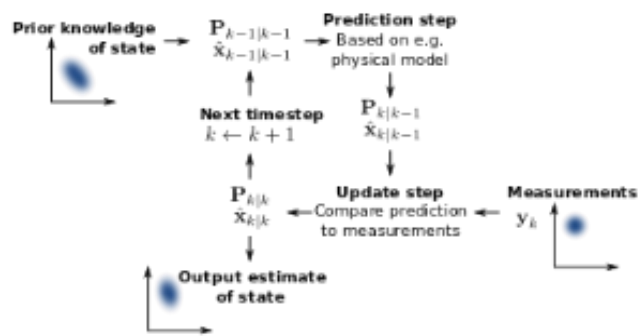


圖16.卡爾曼濾波流程與公式圖

e.空間平面資訊：根據尋找出的多個特徵點，再依據尺度、方向和位置建構平面，並判斷符合的結果，就可以得出多個空間平面的位置資訊。

## (2) .辨識物品

加入物體偵測模型，用以辨識物品在圖中的位置，現有的物體偵測方法是用 CNN（卷積神經網路 Convolutional Neural Networks）演算法辨識影像做到分類，並搭配深度學習模型來確定物品位置。

a.CNN：CNN 的過程分為卷積層 (Convolution)、池化層 (Pooling)、平坦層 (Flatten)、全連接層(Fully Connected Layer)、輸出層 (Output)這五層，流程如下：

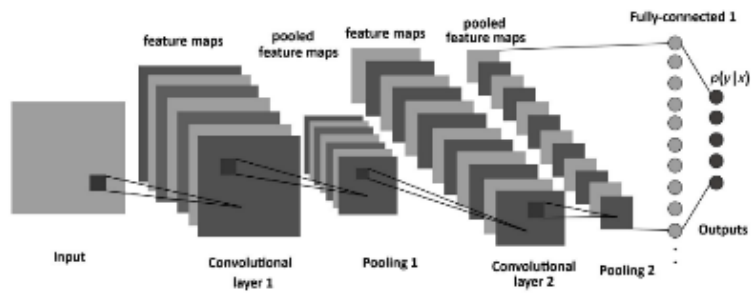


圖17. CNN 架構示意圖

(a). 卷積層：先將圖片二值化，為了保持圖片輸出與原有大小相同，在圖片的最外圍補上 0，之後依序取輸入與權重大小相同的矩陣，進行矩陣相乘運算，輸出特徵圖。特徵圖中將保留輸入圖片明顯的特徵。

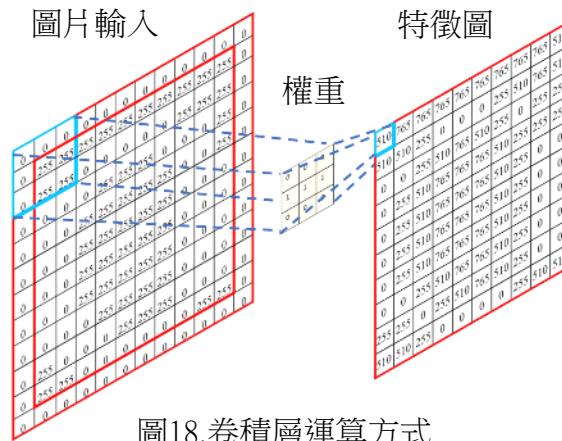


圖18. 卷積層運算方式

(b). 池化層：池化目的在於減少資料量、保留明顯特徵、減少過擬合 (overfitting) 的發生，將圖片分成多個小塊，保留每塊的最大值輸出，輸出將使特徵圖降維。

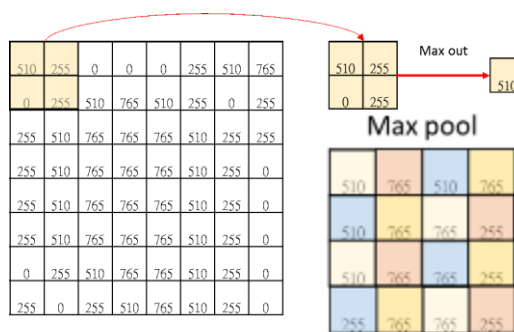


圖19. 池化層運算方式

(c). 平坦層：為了連接全連接層，將特徵圖轉為一維陣列。



(d).全連接層與輸出層：將平坦後的結果連接至神經網路，進行運算並輸出可以得到辨識的權重。

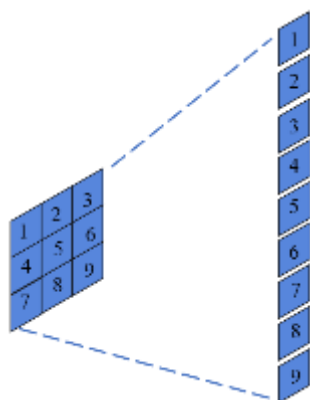


圖20. 平坦層運算方

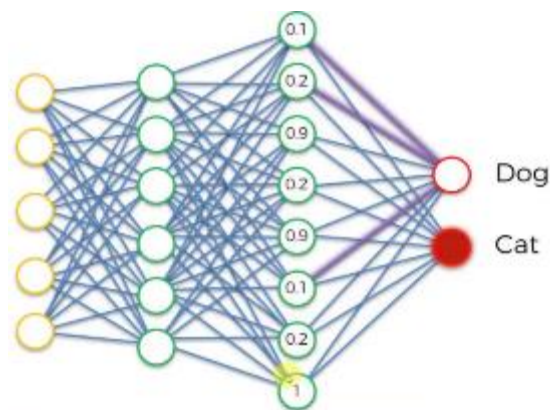


圖21. 全連接層運算方式

b.深度學習模型：常見的深度學習模型大致上分為兩種，運用預先篩選（Region Proposal）的 RCNN（Regions with CNN features）與運用迴歸分析（Regression Analysis）的 YOLO（You only look once），經過表 3 比較兩種類型的模型，最終採用 YOLO 模型，因 YOLO 比起 RCNN 辨識更快速且辨識準確度並不差，使用的效能較少，因此較適合使用 YOLO 模型進行辨識。

表4. RCNN 與 YOLO 比較 來源：組員整理

比較項目	RCNN	YOLO
速度	慢	快
準確率	中	中
訓練時間	中	多
同時偵測物品的數量	多	中
效能使用	多	少
演算法原理	分割出幾千塊可能是物品的範圍，每個範圍進行多次的 CNN，最後透過線性回歸的方式確定範圍。	將物體分類與定位的預測框直接輸出，運用機率的方式篩選可能的區塊，最後以 NMS 與 IOU 框出物品位置。

c.YOLO 模型：在 YOLO 模型中只有卷積層與池化層，移出了全連接層，能夠預訓練不同大小的圖片。先是將圖片分成 S 乘 S 方格，判斷結果包含類別、位置與概率。計算邊界框對於每一個類別出現的可能性，機率高的方格會以中心每小格用 5 個不同尺寸的 anchors box 進行預測。

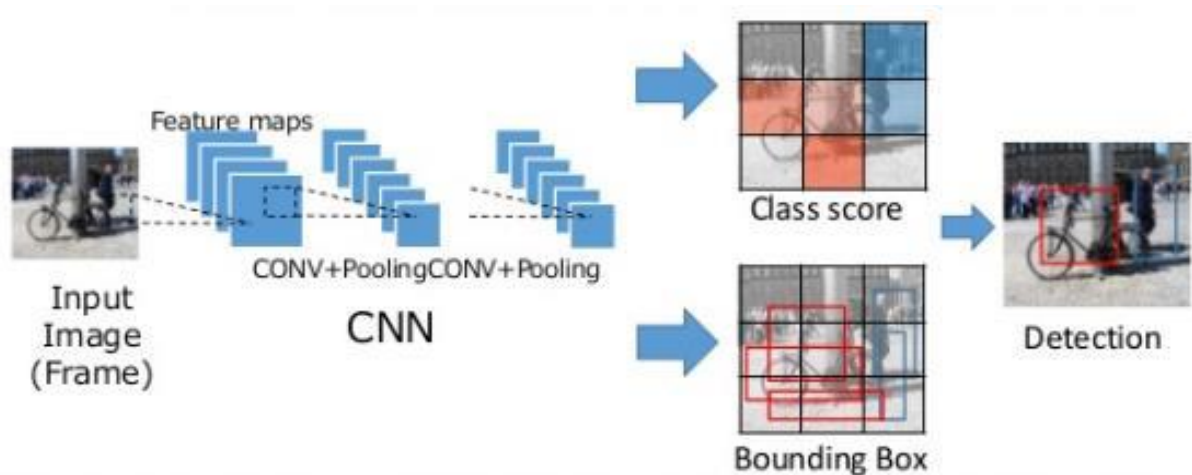


圖22. YOLO 方法示意圖

d.預測框：對應格子 cell，左上角的距離為 (Cx, Cy)，σ 定義為 sigmoid 啟動函數，處理後，將函數值約束在 0~1 的範圍，讓預預測中心在 cell 中，不會偏離 cell。預測框 anchor 的寬高為 (Pw, Ph)，預測位置(bw,bh)是相對於 anchor 的寬高，並乘上係數。

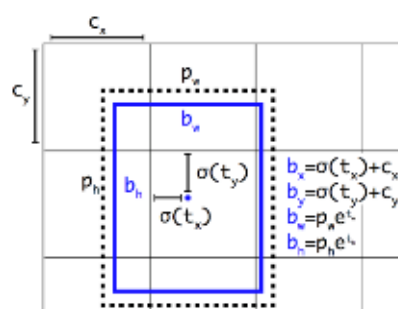


圖23. YOLO 預測框計算

e.訓練模型的方式：將幾百張物品經過標記的圖片，丟給電腦去學習不同物品的特徵，讓電腦以減少誤差為目標去修改權重，達到更準確的辨識物品。誤差主要分析的是邊界框的內容，包含五個預測值，中心座標 (x,y)、長寬 (w,h) 和準確度 IoU (檢測評價函數 Intersection over Union)。

$$\begin{aligned}
 loss_t = & \sum_{l=0}^W \sum_{j=0}^H \sum_{k=0}^A 1_{Max\ IOU < Thresh} \lambda_{noobj} * (-b_{ijk}^o)^2 \\
 & + 1_{t < 12800} \lambda_{prior} * \sum_{r \in (x,y,w,h)} (prior_k^r - b_{ijk}^r)^2 \\
 & + 1_k^{truth} (\lambda_{coord} * \sum_{r \in (x,y,w,h)} (truth^r - b_{ijk}^r)^2 \\
 & \quad + \lambda_{obj} * (IOU_{truth}^k - b_{ijk}^o)^2 \\
 & \quad + \lambda_{class} * (\sum_{c=1}^C (truth^c - b_{ijk}^c)^2)
 \end{aligned}$$

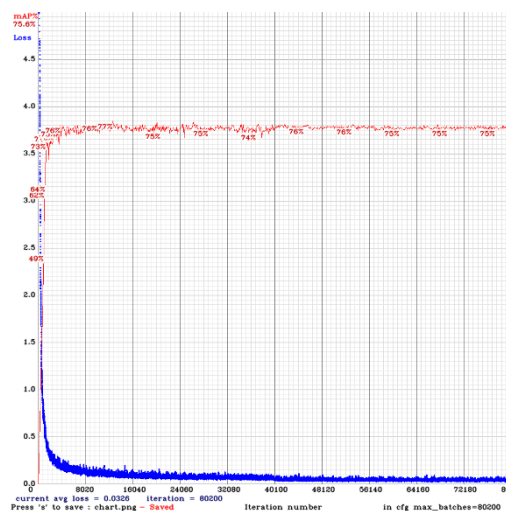
公式1. YOLO 辨識誤差數學式

$$IoU = \frac{\text{預測範圍} \cap \text{標記範圍}}{\text{預測範圍} \cup \text{標記範圍}}$$

公式2. IoU 計算

下表是在做模型訓練時，誤差與訓練筆數的關係，可知在訓練時隨著訓練筆數增加，辨識誤差會逐漸下降，辨識也會越精準。

表 5 YOLO 訓練損失與準確度



## 2.記錄

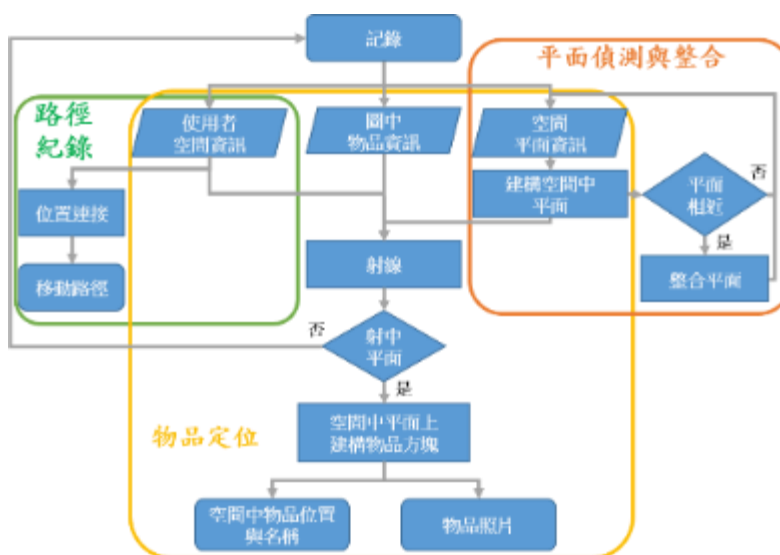


圖24. 記錄流程圖

### (1) .平面偵測與整合

使用辨識所得到的空間平面資訊，將其建構於 SLAM 系統的地圖中，因此地圖中會有多個高低大小不一的平面，藉由判斷新辨識到的平面位置與已建構平面位置相距小於 5 公分，將兩平面以新平面為基準，合併為一個大平面，如果平面位置差距較大則會保留，以便區分出高低不同的桌面、地板或櫃子的平面。

## (2) .路徑記錄

將辨識到的使用者空間資訊，先後位置點連接，用以完整記錄使用者在空間中的移動軌跡。

## (3) .物品定位

(1) 射線：將圖片中辨識到的物品範圍，依最低位置的中央為射線的起點，眼鏡的旋轉角度為方向，在虛擬空間中射出一條射線，當射線碰到已建構出的平面，就將此位置定為物品於空間中的位置。

(2) 空間中平面上建構物品方塊：以物品於空間中的位置為方塊底面中心，在虛擬空間建構一個十立方公分的方塊，以此表示物品的範圍。

(3) 新增：當第一次看見這個物品時，經過射線與空間中平面上建構物品方塊後，添加一個虛擬方塊，虛擬方塊內包含物品的名稱狀態等資訊。

(4) 更新：當第二次看到此物品時，再新增一個新的虛擬方塊，因為物品位置不變，所以兩次虛擬方塊添加的座標系位置都會相同，因此當名稱相同的虛擬方塊在座標系添加後發生重疊碰撞，就表示第二次看到此物品時位置還是與第一次位置相同，不論第二次看物品的視角是否與第一次的視角相同，都能因為添加虛擬方塊在座標系上的位置相同而確定物品位置。

## 3. 搜尋

### (1) .主動搜尋

主動選定物品種類，並指定機器人要搬運到的位置，或是到達自身位置。

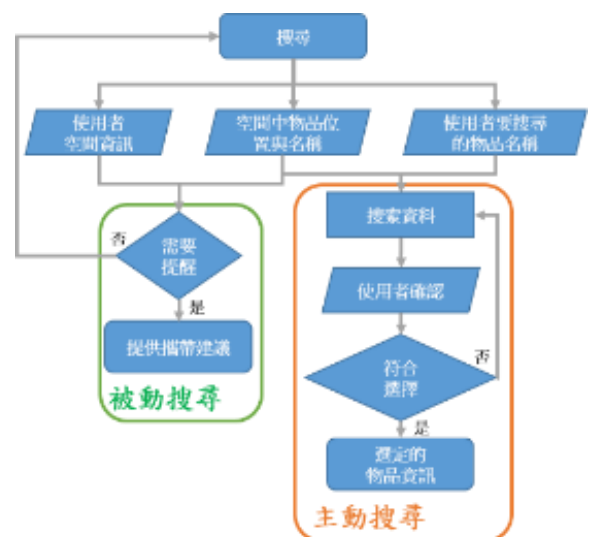


圖25. 搜尋流程圖

## 4. 導航

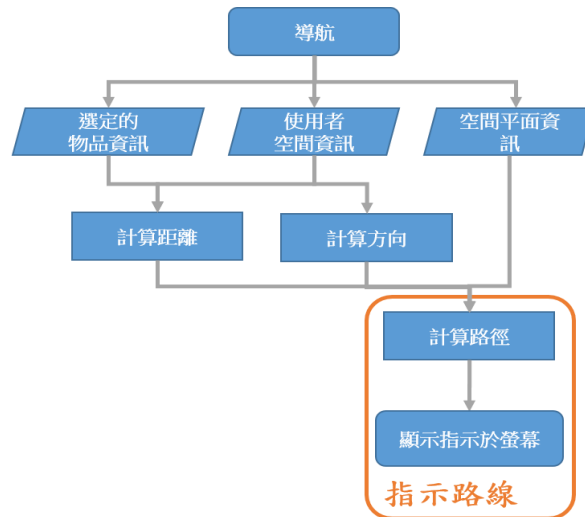
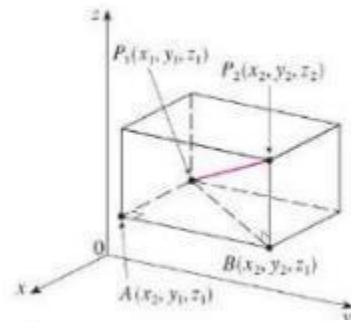


圖26. 導航流程圖

### (1) .計算距離

依具虛擬方塊與使用者在座標系中的兩點，用畢氏定理計算兩點在空間中的距離，並在視角上方即時顯示使用者與物品的三維距離。



$$|P_1P_2| = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2 + (z_2 - z_1)^2}$$

圖27. 計算距離方法

### (2) .計算方向

用三角函數求得座標系中物品相對於使用者的角度，加上使用者的偏轉視角計算出指示箭頭的旋轉角度。

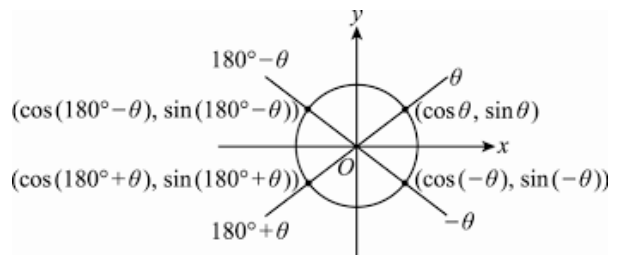


圖28. 計算方向方法

### (3) .路線

從已選定要尋找的物品資料中，計算且判斷最快能到達的位置，將直線距離與旋轉的指令傳給機器人，讓機器人到達物品位置。

## 5.搬運

到達物品所在位置後，能夠搬運物品移動到指定位置，透過 PID 加減速控制理論減少物品晃動與提高移動準確性。

### (1) .PID 加減速控制理論

比例控制器 P：基本的等比加減速控制。

積分控制器 I：重複加減控制直到穩定。

微分控制器 D：差距較大則比例增大。

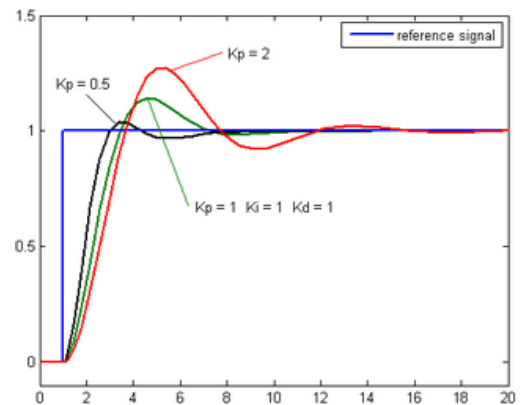


圖29. PID 方法示意圖

### (2) .PWM 信號控制馬達轉速

PWM 是將 PID 的類比訊號轉換為脈波訊號的一種技術，透過脈波訊號輸出給編碼直流馬達，使馬達旋轉操控機器人移動。

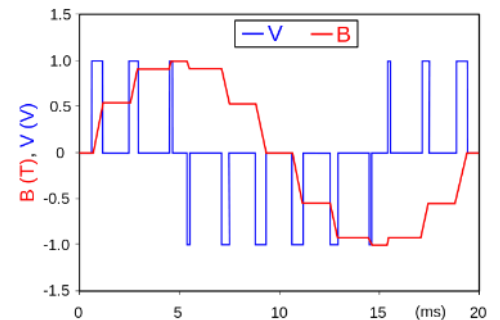


圖30. PWM 方向示意圖

### 三、研究結果與討論

#### (一)、研究結果

##### 1.SLAM 空間辨識

###### (1) .特徵點取得

取得影像中灰階後顏色相差較大的點或明顯邊緣點作為特徵點。如：圖 31

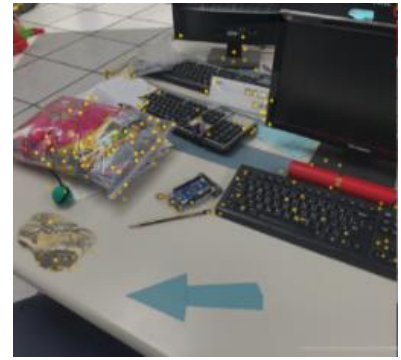


圖31.特徵點(黃色圓點)與辨識平面

###### (2) .平面紀錄與擴充

依據特徵點的分布繪製平面，並整合多個相近平面，達到擴充與彙整空間資訊。如：圖 32



之前



之後



圖32.平面擴充過程

##### 2.YOLO 物體偵測

###### (1) .辨識物品種類

將拍攝到的影像經過 YOLO 物體偵測模型辨識物品種類與可信度，只顯示可信度高於 50 的物品。如：圖 33

###### (2) .框出辨識物範圍

以模型辨識到的兩點位置，繪製辨識方框，以確定物品於視角中的範圍。如：圖 33

##### 3.結合空間與影像辨識

###### (1) .建構地圖

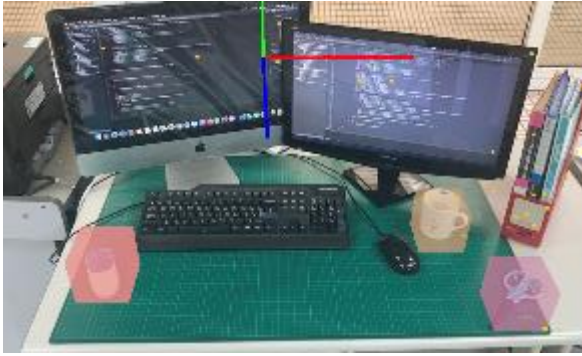
能夠依據特徵點在空間中的分布，建構空間地圖，如：圖 34

###### (2) .標記物品位置

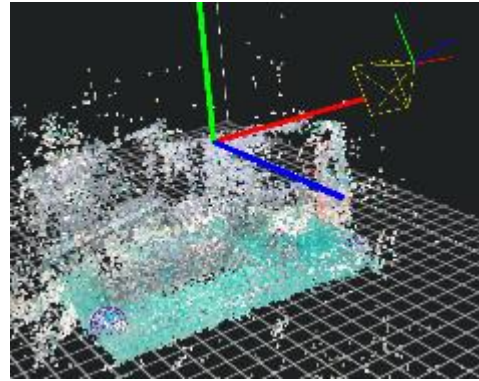
定位辨識到的物品於空間中的位置，標記於地圖座標位置上，以紀錄物品的位置。如：圖 34



圖33.YOLO 辨識結果



視角所見影像



特徵點建構空間地圖

圖36.空間辨識地圖

#### 4. 機器人移動

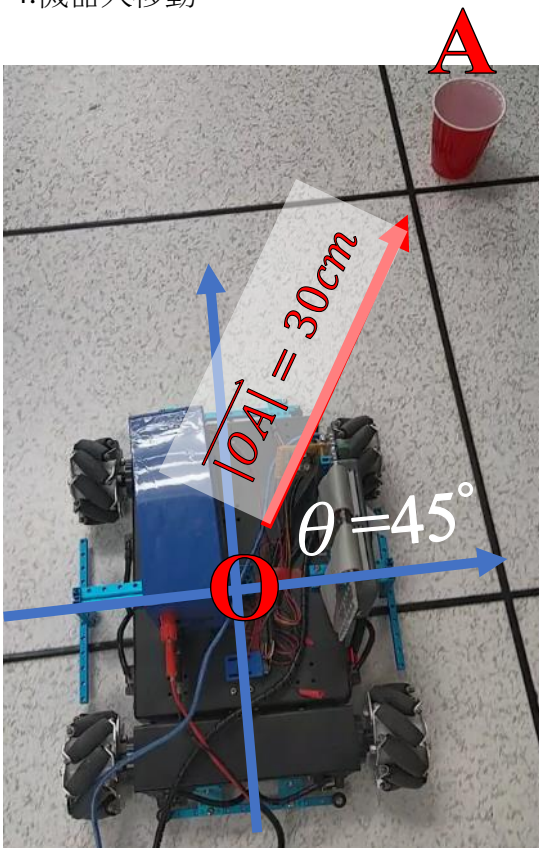


圖35. 機器人移動方向

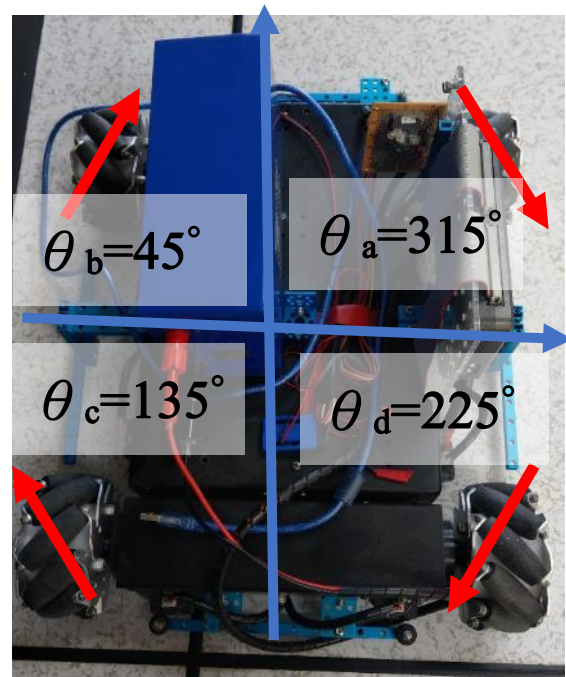


圖34. 輪子移動方向速度

$$|\overline{OA}| \times \cos(\theta_{\text{輪子角度}} - \theta_{\text{車體移動方向}}) = \text{輪子方向速度 cm/s}$$

表 2. 移動計算 (以向 45°, 30cm/s 為例)

麥克納姆輪	輪子速度
a 輪	$30 \times \cos(315^\circ - 45^\circ) = 0$
b 輪	$30 \times \cos(45^\circ - 45^\circ) = 30$
c 輪	$30 \times \cos(135^\circ - 45^\circ) = 0$
d 輪	$30 \times \cos(225^\circ - 45^\circ) = -30$



## 5.製作 AR 空間記憶眼鏡

### (1) .製作 APP

以手機 APP 的方式撰寫程式。

如：圖 37

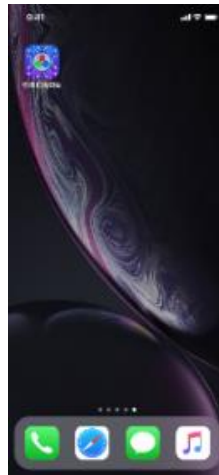


圖37. 安裝 APP



圖38. 使用者攜帶樣式

### (2) .自製 AR 空間記憶眼鏡

結合手機與 VR 眼鏡模擬 AR 眼鏡，使用相機影像與 IMU 資訊達到紀錄空間、定位自身，以螢幕顯示資訊來達成眼鏡功能。如：圖 38

### (3) .物品搜尋選擇

使用音量鈕選擇不同物品名稱，搜尋選擇的物品位置並給予導航。如：圖 39



(箭頭指向藍色方塊)  
搜尋剪刀

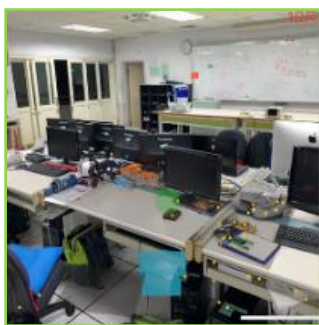


(箭頭指向綠色方塊)  
搜尋遙控器

圖39. 指示過程

### (4) .導航物品位置

依據自身與物品的位置，箭頭與距離標示都可以正確指引使用者到達物品位置。  
如：圖 40



近距離面對物品



中距離背對物品

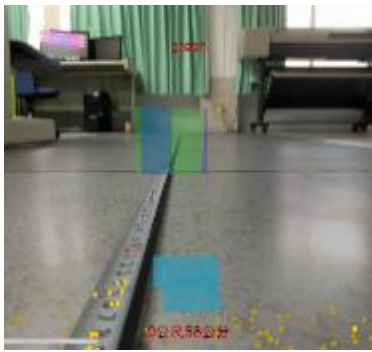


遠距離面對物品

圖40. 導航過程

實驗一 實際距離與預估距離誤差

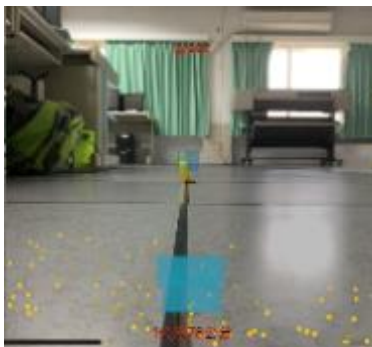
實際 60 公分  
測得 58 公分



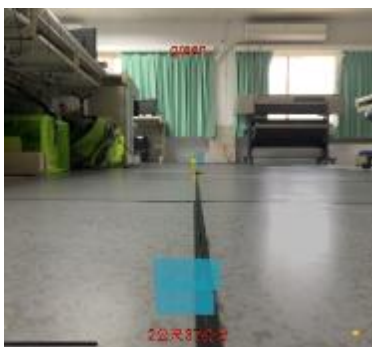
實際 120 公分  
測得 118 公分



實際 180 公分  
測得 178 公分



實際 240 公分  
測得 237 公分



實際 300 公分  
測得 295 公分

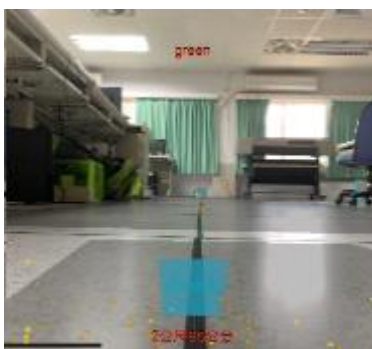
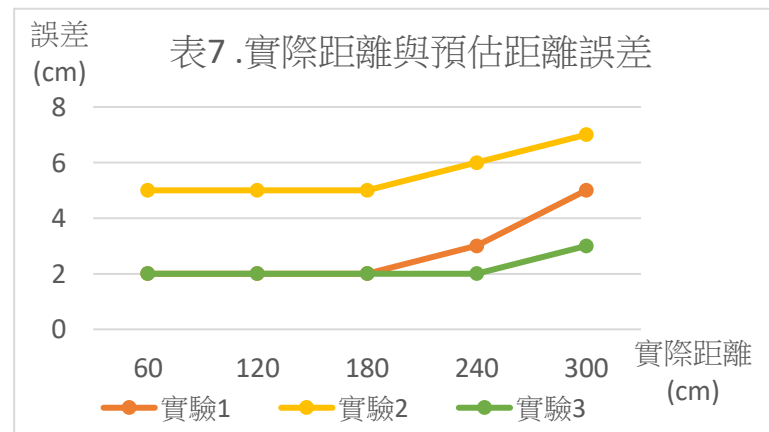


圖41. 不同距離測試圖

表6. 移動與距離誤差

實際距離 (公分)	預估距離 (公分)	差距 (公分)	誤差增量 (公分)	總誤差 (公分)
實驗 1				
60	58	2	0	0
120	118	2	0	0
180	178	2	0	0
240	237	3	1	1
300	295	5	2	3
實驗 2				
60	55	5	0	0
120	115	5	0	0
180	175	5	1	0
240	234	6	1	1
300	293	7	1	2
實驗 3				
60	58	2	0	0
120	118	2	0	0
180	178	2	0	0
240	238	2	0	0
300	297	3	1	1



實驗過程與結果

以每塊 60 公分的地板為基準，測量真實距離與程式預估距離的誤差，距離 200 公分內都保持相同誤差，此誤差是因為定位點與測量點的偏差，但到了 200 公分後，誤差明顯有增加，表示超過 200 公分後系統才會產生大於 1 公分的測量誤差。

實驗二 移動時物品辨識能力

表8. 物品不同方向  
最大可辨識距離

方向(度)	距離(公分)
0	165
45	187
90	173
135	130
180	150
225	162
270	157
315	163
辨識：可信度高於 50 物品離地 70(公分) 相機離地 150(公分)	

表9. 辨識距離相等  
不同方向的可信度

距離(公分)	方向(度)	可信度(%)
50	0	86.1
	45	92.0
	90	89.2
	135	86.8
	180	87.4
100	0	71.4
	45	66.8
	90	75.1
	135	66.9
	180	74.7



距離：1 公尺  
辨識：是



距離：1.5 公尺  
辨識：是



距離：2 公尺  
辨識：否

圖42.不同距離辨識

分別測試不同方向與距離，不同物品的辨識準確度，經過實驗得知在不同方向辨識且距離 1.5 公尺以內，多數情況下辨識可信度皆高於 50。

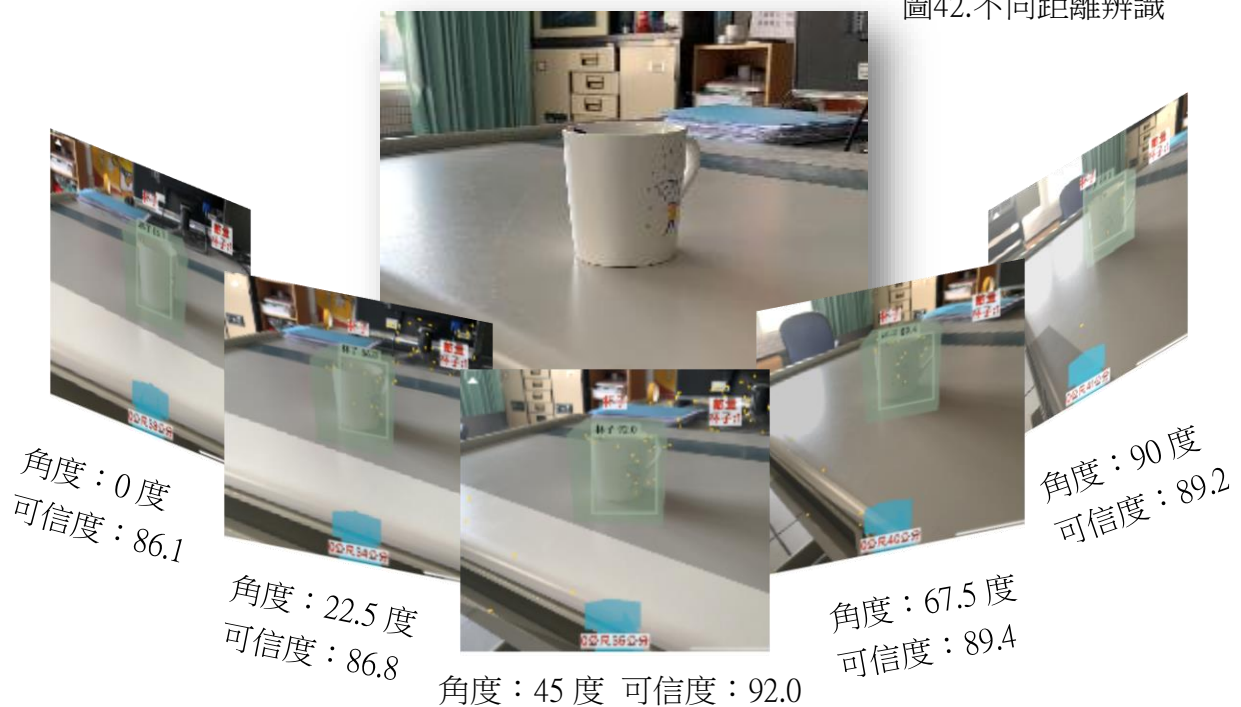


圖43.不同方向辨識

實驗三 測試遠離再回到定位點的偏移誤差

表11.移動過程紀錄

表10. 移動遠近與定位誤差

直線長度 (公分)	誤差距離 (公分)	每公分誤差 百分比(%)
480	2	0.42
	3	0.63
	2	0.42
	1	0.21
	1	0.21
	3	0.63
	2	0.42
	1	0.21
	3	0.63
總平均誤差	1.9	0.33
600	2	0.50
	3	0.33
	2	0.50
	3	0.50
	3	0.17
	1	0.17
	1	0.33
	2	0.33
	2	0.17
總平均誤差	2	0.63



圖44. 移動前與移動後定位誤差演示圖

實驗過程與結果

測試遠離再回到定位點後，定位點偏移的距離，以距離定位點 480 公分與 600 公分，分別測試 10 次，平均結果得出，約產生 0.35%的誤差。

#### 實驗四 機器人移動準確率

表11. 相同距離多次

最高速度(cm/s)	時間(s)	距離(cm)	實際(cm)	誤差(%)
80	4.06	200	204	2
80	3.89	200	203	1.5
80	4.13	200	203.5	1.75
80	4.02	200	202.5	1.25
80	4.04	200	202	1
80	4.11	200	202.5	1.25
80	4.06	200	203	1.5
80	4	200	203	1.5

表12. 不同最高速度

最高速度(cm/s)	時間(s)	距離(cm)	實際(cm)	誤差(%)
30	7.70	200	204	2
40	6	200	205	2.5
50	4.87	200	204	2
60	4.40	200	203.5	1.75
70	4.16	200	203	1.5
80	3.99	200	203	1.5

表13. 不同距離

最高速度(cm/s)	時間(s)	距離(cm)	實際(cm)	誤差(%)
80	1.92	50	50.5	1
80	2.7	100	102	2
80	3.39	150	153	2
80	3.76	200	203	1.5
80	4.52	250	255	2
80	5.19	300	307	2.3

#### 實驗過程與結果

可知以 80 最高速度，移動 200 公分誤差約 1.5%，表示物品定位的位置，機器人可以正確到達，且移動時間約為 4 秒。

## （二）、討論

- 1.以 SLAM 系統辨識環境，YOLO 方法辨識物品達到紀錄物品位置的成果，雖然準確度與穩定性還有改善空間，但確實能達到移動時定位物品位置。
- 2.使用者能透過程式，方便的記錄與得知物品的擺放位置，機器能根據指令到達物品放置的位置。
- 3.現有方法在眼鏡上及微電腦上執行，因為效能達不到即時且大量資訊處理的能力。

## 四、結論與應用

### （一）、結論

本作品利用 SLAM、YOLO 等技術，建構一個可做到環境建構、記憶物品位置的記憶眼鏡，並整合搬運機器人，達到記憶物品使人與機器能具有更好協同作業，完成的功能如下：

- 1.運用 SLAM 達到辨識環境、YOLO 達到物品辨識，完成空間中定位物品的位置。
- 2.使用者能透過指令，使機器人到達物品所在位置並搬運物品。
- 3.使機器人在非固定環境能夠搬運物品並到達指定位置。

未來展望：近期目標是優化影像辨識，達到更方便且實用，遠期目標將整理不同使用者的辨識數據，達到使用者共用辨識資料，實現物品空間資訊的互通與傳遞，目標規劃如下：

- 1.近期：提升機器人路徑規劃，完善使用者指令傳輸方式。
- 2.遠期：將優化機器人現有功能，擴增更多功能使機器處理更多狀況。

### （二）、應用

發展與應用空間：此作品可運用的範圍除了記憶物品位置外，加入了可移動的搬運機器人，可用於建築工地紀錄建材位置，搬運建材至工人旁，加快材料運輸、減少搬運傷害及危險，另外居家物品清潔整理，也可有效整理被使用過的物品，另外工廠、辦公室的物品處理等，也能夠有幫助，提升人與機器人的協同作業效率與方式，提升工作效率。

## 五、參考文獻

### (一)、論文書籍：

- 江翊璋 (2018)。基於 SLAM 之擴增實境技術於史蹟導覽之應用與創作。國立臺北科技大學互動設計系碩士論文，臺北市。取自 <https://hdl.handle.net/11296/76v76n>
- 高翔,張濤 (2017)。視覺 SLAM 十四講：從理論到實踐。出版商：電子工業出版社
- 陳玉琨,湯曉鷗 (主編) (2018)。人工智慧基礎 (高中版)。出版商：華東師範大學出版社
- 葉暘、何政勳 (2017)。用於機器人空間建模的仿生認知系統。臺灣國際科展
- David G. Lowe, **Distinctive image features from scale-invariant keypoints**. In IJCV, 2004.
- Ian Goodfellow, Yoshua Bengio, & Aaron Courville (2016) . **Deep Learning**. MIT Press.
- J. Redmon, A. Farhadi. **YOLO9000: Better, Faster, Stronger**. In CVPR, 2016.
- J. Redmon, S. K. Divvala, R. B. Girshick, and A. Farhadi. **You only look once: Unified, real-time object detection**. In CVPR, 2016.
- Ole Kiehn & Hans Forssberg (2014) , **The Brain’ s Navigational Place and Grid Cell System**.
- Sudeep Pillai and John J. Leonard (2015) ,**Monocular SLAM Supported Object Recognition**.

### (二)、網路資源：

- Apple Develop Document, **ARKit**, 引用自 <https://developer.apple.com/documentation/arkit>
- Apple Develop Document, **Working with Core ML Models**,  
引用自 <https://developer.apple.com/machine-learning/build-run-models/>
- Steven Shen, **YOLO with CoreML**,  
引用自 <https://medium.com/@syshen/yolo-with-coreml-819799789c11>
- Vlas Voloshin,**Visualizing ARKit Sessions**,  
引用自 <https://www.ittybittyapps.com/blog/posts/2018/12/visualizing-arkit-sessions/>

(三)、圖片來源：

圖 1：《How It Works 知識大圖解》 2015 年 9 月號第 12 期

圖 2：<https://case.ntu.edu.tw/blog/?p=19357>

圖 3：[http://scimonth.blogspot.com/2014/12/blog-post\\_9.html](http://scimonth.blogspot.com/2014/12/blog-post_9.html)

圖 4：<https://zh.wikipedia.org/wiki/Google%E7%9C%BC%E9%95%9C>

圖 5：<https://www.idc.com/getdoc.jsp?containerId=prUS44001618>

圖 11：<https://code.fb.com/ml-applications/segmenting-and-refining-images-with-sharpmask>

圖 12：<https://www.jianshu.com/p/7faa4a3af589>

圖 14：<https://arxiv.org/pdf/1708.03852.pdf>

圖 16：<https://goo.gl/TRBqpQ>

圖 17：<https://tinyurl.com/ybshnf2j>

圖 18、19、20、21：<https://medium.com/@chih.sheng.huang821>

圖 22：<https://tinyurl.com/y3jxhkom>

圖 23、公式 1：YOLO 論文

圖 27：<https://goo.gl/TNCAUG>

圖 28：<https://goo.gl/4Vg7WT>

圖 29：<https://tinyurl.com/y6sn82b7>

圖 30：<https://tinyurl.com/y2dgxp9>



## 【評語】 100024

本作品是利用尿液經由微生物來產生氫離子，造成與另一鹽水的氫離子構成濃度差，來構成燃料電池之作用。研究者在微生物之培養選擇和電池槽之設計組合也有一定之研究成果。