樹莓派採用機器人作業系統之效能分析以 AIoT 智慧機器人競速賽為例

陳志遠

南亞技術學院 資訊工程系

摘要

隨著智慧化時代正式來臨，人工智慧與物聯網技術已經被廣泛應用於各個領域，人工智慧更經常被應用於智慧機器人，國內外也出現很多智慧機器人的實作競賽。其中智慧機器人競速一直是國內外經常比賽的項目之一，常見的比賽類型有以視覺感測器辨識賽道及號誌使智慧機器人完成賽道。這類的競賽通常會使用樹莓派當作智慧機器人的運算平台並且進行機器視覺辨識處理。本論文以 AIoT 智慧機器人競速賽為例，分析在樹莓派平台中有採用機器人作業系統與沒有採用機器人作業系統的競速效能與辨識穩定度差異。經過 AIoT 智慧機器人競速賽的實驗測試後，在樹莓派平台中有採用機器人作業系統比沒有採用機器人作業系統的競速效能提升 70%以上及辨識穩定度提升 160%以上。

關鍵字：樹莓派、機器人作業系統、智慧機器人、人工智慧

Email: cychen@nanya.edu.tw　TEL: (03) 4361070 ext. 5301　FAX: (03) 4373959

THE EFFICIENCY ANALYSIS OF THE RASPBERRY PI USING THE ROBOT OPERATING SYSTEM TAKES THE AIOT SMARTS ROBOT RACE AS AN EXAMPLE

Chih-Yuan Chen

Department of Computer Science and Information Engineering,
Nanya Institute of Technology

Abstract

With the coming of the era of intelligence, Artificial Intelligence and Internet of Things technologies have been widely used in various fields. Artificial Intelligence is more often applied to smart robots, and there are many smart robot implementation competitions. Among them, smart robot racing has always been one of the frequent competitions. Common types of competitions include visual sensors to identify the track and signs to make the smart robot complete the track. This type of competition usually uses the Raspberry Pi as a computing platform for smart robots and performs machine vision recognition processing. Taking the AIoT smart robot racing competition as an example, the paper analyzes the difference in racing performance and recognition stability between the robot operating system and the robot operating system not used in the Raspberry Pi platform. After the experimental test of the AIoT smart robot racing competition, the robot operating system on the Raspberry Pi platform has increased by more than 70% and the recognition stability increased by more than 160% compared to the racing performance without the robot operating system.

Keywords：Raspberry Pi, ROS, Smart Robot, Artificial Intelligence

Email: cychen@nanya.edu.tw　TEL: (03) 4361070 ext. 5301　FAX: (03) 4373959

壹、前言

為配合政府產業政策願景，致力五加二產業創新，推動智慧機械、亞洲・矽谷、生技醫藥、綠能科技、國防及循環經濟等，強化產業關鍵技術自主與多元應用能力，培育我國經濟發展新動能。加強扶植中小企業，善用數位、循環及體驗經濟，活化在地發展[中華民國經濟部，2016]。其中推動智慧機械與亞洲・矽谷都與人工智慧(Artificial Intelligence, AI)、雲端計算、大數據、物聯網(Internet of Things, IoT)、智慧機器人等技術相關。推動智慧機械的願景與使命為：以精密機械導入智慧技術，透過智慧化產線進行智慧製造，並以國內產業為練兵對象，進而整廠整線輸出國外，建構智慧機械產業之生態體系[智慧機械推動辦公室，2017]。亞洲・矽谷的願景為：為連結全球先進科技研發能量，搶進下一世代產業，國家發展委員會、科技部、經濟部等部會共同推動「亞洲・矽谷推動方案」，經105年9月8日行政院第3514次院會通過，將建立一個以研發為本的創新創業生態系，由「推動物聯網產業創新研發」及「健全創新創業生態系」2大主軸，輔以連結國際、連結未來及連結在地的3大連結，透過4大策略來推動，希望以物聯網促進產業轉型升級，並以創新創業驅動經濟成長[亞洲・矽谷，2016]。因此產業非常需要人工智慧、雲端計算、大數據、物聯網、智慧機器人等人才。

隨著智慧化時代正式來臨，人工智慧、雲端計算、大數據、物聯網、智慧機器人等技術已經被廣泛應用於各個領域。為了培育這些人才，舉辦競賽來培育人才是一種有效的方法，特別是針對技職體系的學生[陳志遠、羅裕群，2014]。近幾年，國內外出現很多智慧機器人的實作競賽，例如TIRT全能機器人國際邀請賽[TIRT國際新創機器人節，2018]，其中智慧機器人競速一直是國內外經常比賽的項目之一，常見的比賽類型有以視覺感測器辨識賽道及號誌使智慧機器人完成賽道，例如視覺無人車競賽[TIRT國際新創機器人節，2018]、AIoT智慧機器人競速賽[城市盃智慧型自走車競賽，2020]。這類的競賽通常會使用樹莓派[Raspberry Pi Foundation，2009]當作智慧機器人的運算平台並且進行機器視覺辨識處理。本論文以AIoT智慧機器人競速賽為例，分析在樹莓派平台中有採用機器人作業系統(Robot Operating System, ROS) [機器人作業系統官方網站，2007]與沒有採用機器人作業系統的競速效能與辨識穩定度差異。

本文共分為五節，其餘各節概述如下：第貳節為AIoT智慧機器人競速賽介紹。第參節針對智慧機器人軟體與硬體設計說明，先說明智慧機器人硬體的設計，接著說明智慧機器人軟體架構與演算法設計。第肆節將針對第參節所設計的智慧機

器人分為軟體採用ROS與不採用ROS進行測試及競賽，並且分析與討論測試與競賽結果。第伍節為結論與未來研究方向。

貳、AIoT智慧機器人競速賽

智慧機器人競速賽最常見的競賽為智慧型自走車循跡競速賽及智慧型自走車避障競速賽[陳志遠、宋守正、羅裕群，2013]、[TIRT 國際新創機器人節，2018]，近幾年，人工智慧應用發展成熟，以視覺感測器辨識賽道及號誌使智慧機器人完成賽道的競賽類型有增加的趨勢。本論文將以 AIoT 智慧機器人競速賽為例，分析比較智慧機器人分採用 ROS 與不採用 ROS 的競速效能與辨識穩定度差異。AIoT 智慧機器人競速賽競賽規則如下[城市盃智慧型自走車競賽，2020]：

(一)競賽場地：

1. 競賽場地：白底黑線，黑線外有美工圖。黑線圖參考如圖 1，跑道材質為油性防水相紙單面霧膜。整體競賽場域僅於比賽當天現場公布。
2. 整體競賽場域：210cm x 150cm；黑色跑道區域大小：寬 165+/-5cm 高 80+/-5cm。
3. 道路寬：18+/-2cm；線寬 4+/-2cm。跑道黑線外側至少保留有 2cm 之白底不施以美工。
4. 路上標誌牌：圖案外接矩形大小為 6.5+/-0.6cm 間之白底黑色的單純幾何圖樣，如圖 2 所示。

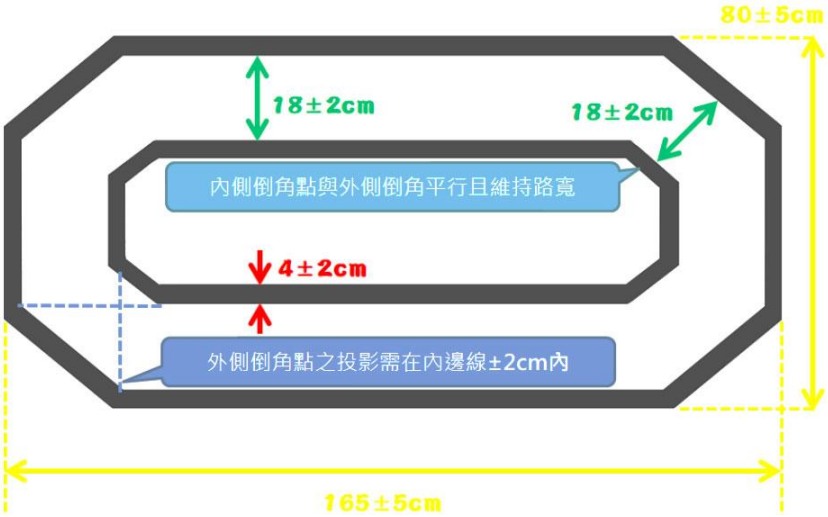

圖 1：AIoT 智慧機器人競速賽競賽場地尺寸說明圖

圖 2：辨識圖形依序為三角形、正方形、五邊形、圓形、十字形、右轉彎

(二)競賽規則：

1. 每隊只有一次出賽機會(或當天由裁判長決定次數)。
2. 凡經唱名 3 次未到者，即視同比賽棄權。
3. 經唱名後，選手先向裁判抽取標誌牌兩個（或是固定組合中的一組，一組為兩張）後自行放置好，再將智慧機器人放置競賽指定起點，放置動作需於 20 秒內完成，期間不得藉故再對智慧機器人所有組件進行調整、設定或置換(含程式、電池及電路等)，亦不得請求暫停。待裁判哨音後開始計時。
4. 參賽機器人需在雙黑線賽道上順時針行駛，在長邊尾端附近放置標誌牌圖樣（6 取 2）。
5. 當機器人看到標誌牌到下一個長邊前,機器人需語音報出該圖樣的名稱，若正確則計乙次。
6. 當兩圈完成時停止計時，並將該時間與正確次數做為成績。
7. 若有多報或誤報或報太小聲聽不到，該次區域辨識視為無效，機器人仍需繼續前進。
8. 過程中機器人若離開跑道(機器人正投影離開黑色邊線)，裁判吹哨請選手將機器人放回離開跑道時之跑道中心位置。每次放置後成績須加 10 秒，最多兩次機會，第三次則算未完賽，記錄辨識成功次數作為成績。
9. 競賽時間暫定 3 分鐘，過程中不會中斷或延長計時。超過 3 分鐘停止競賽，紀錄辨識成功次數作為成績。
10. 成績以正確次數為優先，若同次(0~4 次)數者，以時間短者為勝。
11. 競賽過程中，參賽選手及自走車不得破壞比賽場地，若裁判發現有此項行為，得宣告該選手及自走車退場，並喪失比賽資格。

參、軟體與硬體設計

依據第貳節說明的AIoT智慧機器人競速賽場地與規則，本節將說明智慧機器

人的硬體設計與軟體設計，其中軟體設計分為採用ROS與不採用ROS的版本，ROS我們採用ROS 2.0版。接下來將分五大部分說明智慧機器人軟體與硬體的設計，第一部分介紹使用的單晶片電腦硬體樹莓派；第二部分介紹使用的軟體ROS 2.0；第三部分說明智慧機器人整體的硬體設計；第四部份說明不採用ROS的程式設計流程；第五部分說明採用ROS 2.0的程式設計流程。

一、樹莓派

樹莓派（Raspberry Pi）為樹莓派基金會[Raspberry Pi Foundation，2009]所開發的一款於Linux上運作的單晶片電腦，如圖3為Raspberry Pi 4 Model B。開發樹莓派的目的是以低價硬體及自由軟體促進學校的基本電腦科學教育。根據樹莓派基金會，截至2015年2月，已經售出超過500萬台樹莓派，成為最暢銷的英國電腦。截至2016年11月，累計售出1100萬台樹莓派。截至2017年3月，累計售出1250萬台樹莓派，成為最暢銷的通用電腦。截至2017年7月，累計售出1500萬台樹莓派。截至2018年3月，累計售出1900萬台樹莓派。截至2019年12月，累計售出3000萬台樹莓派[Raspberry Pi Foundation，2009]。我們的智慧機器人使用的單晶片電腦硬體為Raspberry Pi 4 Model B 4GB版本，Raspberry Pi 4 Model B發布於2019年6月底，處理器升級為1.5GHz的博通BCM2711（四核Cortex-A72），增大了板載記憶體容量，為1/2/4GB，藍牙升級為5.0，擁有2個USB2.0介面，2個USB3.0介面，電源也採用了較新的USB-C介面。

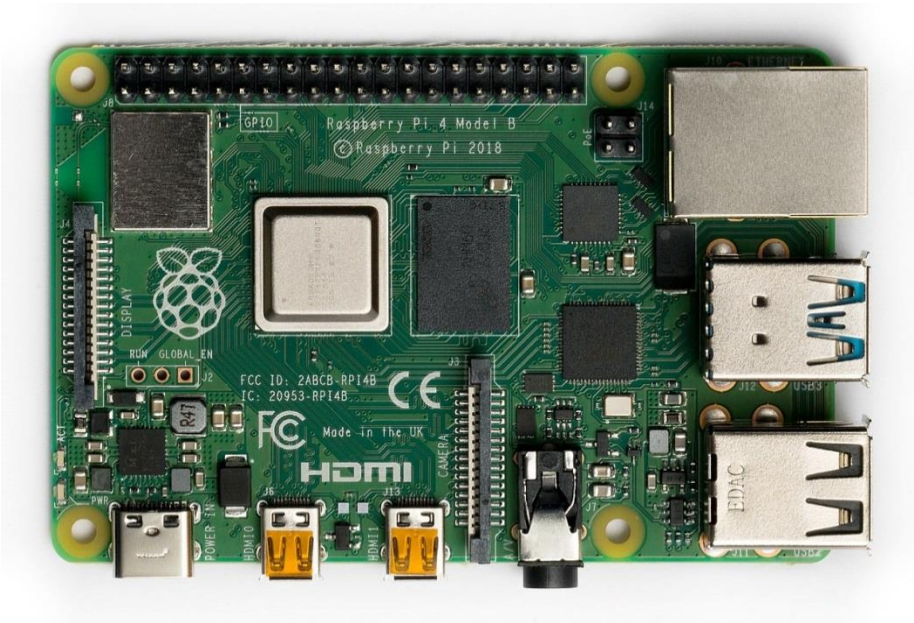

圖3：Raspberry Pi 4 Model B板

二、ROS 2.0

ROS是專門為機器人軟體開發時所設計出來的一套電腦作業系統架構。ROS由Stanford Artificial Intelligence Laboratory新創的Willow Garage公司在2007年所開發，開發至今已經13年，並且於2010年初推出ROS 1.0版。ROS在2008年至2013年期間主要由Willow Garage公司管理維護，從2013年開始ROS的開發和維護管理工作移交給新成立的Open Source Robotics Foundation [Open Source Robotics Foundation官方網站，2013]。

根據Dirk Thomas等人在ROSCon 2014的演講所描述[Dirk Thomas, William Woodall, Esteve Fernandez, 2014]，ROS 2.0的開發是為了解決ROS 1.0所遇到的一些問題。另外，根據Open Source Robotics Foundation總裁Brian Gerkey [Open Source Robotics Foundation官方網站，2013]針對ROS 1.0與ROS 2.0的差異說明如下：

1. 相較於ROS 1.0在ROS 2.0做了許多的改良，它是建立在產業標準上所打造的平台。包括在通訊方式、QoS通訊服務品質、安全性上都有十足進展，能支援多平台以及更小型的系統設計。
2. 在ROS 1.0只有一種傳送訊息的方式，且在安全防護上的做法沒有太多選擇，而在ROS 2.0上可提供QoS優先順位的組態設定及訊息的安全防護，此外以無人車的應用來說，ROS 2.0已可與即時控制系統相容。
3. 微軟2018年也宣布Windows 10將支援ROS 1.0與ROS 2.0，ROS 1.0長久以來是安裝在Linux平台上，而ROS 2.0則可安裝在Linux、Mac OS及Windows平台，大幅簡化開發人員的作業程序。
4. ROS 2.0可支援做到更小的系統設計，如可在小型嵌入式微處理器上執行，這常見於自動駕駛輔助系統。
5. 整體來說，ROS 2.0大幅節省企業開發成本，不需要所有環節都靠自己開發因而縮短產品上市時間。且在ROS 2.0上開發商用產品，如無人車，從最初設計到完成所有程序皆可通過驗證，是可真正上路的運輸工具，不像 ROS 1.0所開發的較類似技術論證的原型車。

另外，Dirk Thomas等人在ROSCon 2014的演講“ROS 2.0: Developer preview” [Dirk Thomas, William Woodall, Esteve Fernandez, 2014]中也說明ROS 2.0其主要設計理念為中介層的抽換，將原先過時中心化發布訂閱機構(Publish Subscribe Middleware)換成比較新穎的技術例如：DDS (Data Distribution Service)、Zeroconf (Zero-configuration networking)、ZeroMQ等。詳細的ROS 2.0與DDS整合架構或未

來可能應用可參考機械工業雜誌424期的「開源機器人作業系統第二代的介紹與未來應用」這篇文章[林浩銘、蔡宗廷，2018]。

三、 智慧機器人的硬體設計

本論文的智慧機器人影像處理系統與語音處理系統硬體使用Raspberry Pi 4 Model B 4GB版本當作主要的運算系統，搭配攝影鏡頭進行影像擷取與迷你喇叭進行語音輸出。移動處理系統硬體使用Arduino UNO板當作控制系統，搭配2顆連續旋轉伺服馬達進行智慧機器人移動控制。智慧機器人的硬體系統架構如圖4所示。

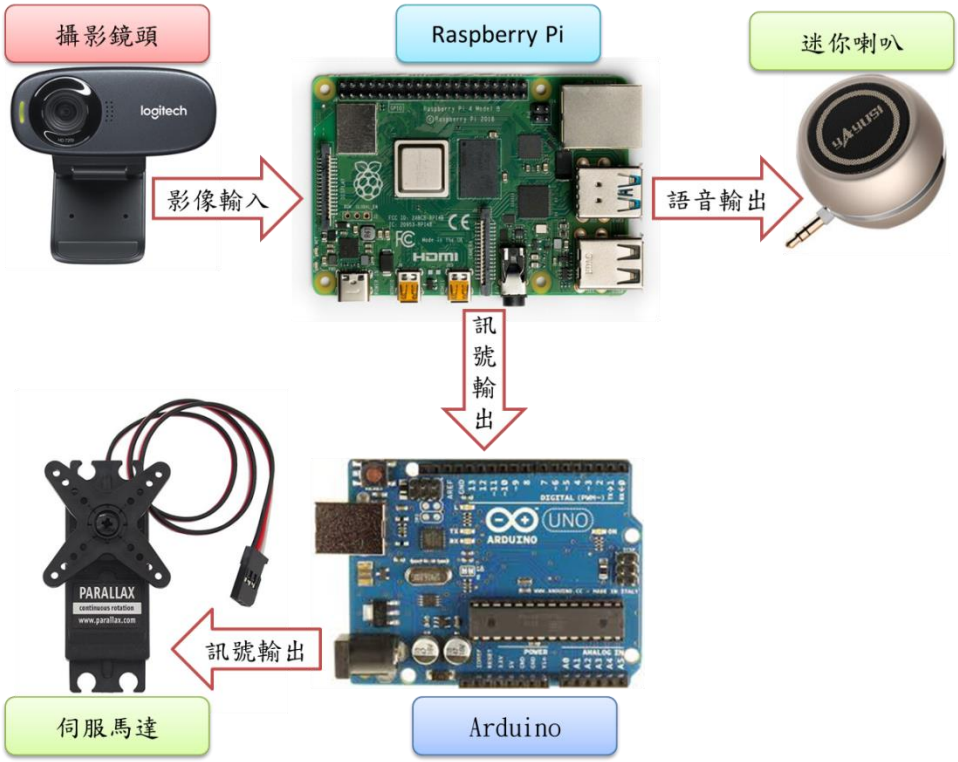

圖4：智慧機器人的硬體系統架構設計

四、不採用ROS的程式設計流程

在前一部分我們已經說明智慧機器人的硬體系統架構，本部分將說明不採用ROS的程式設計流程。程式設計流程圖如圖5所示，程式有7大步驟，影像擷取、二值處理、辨識處理、移動辨識、圖形確認、語音輸出這6大步驟都是在樹莓派中依序執行，並且使用Python程式語言撰寫程式；只有馬達控制這個步驟是在Arduino中執行，並且使用C程式語言撰寫程式。

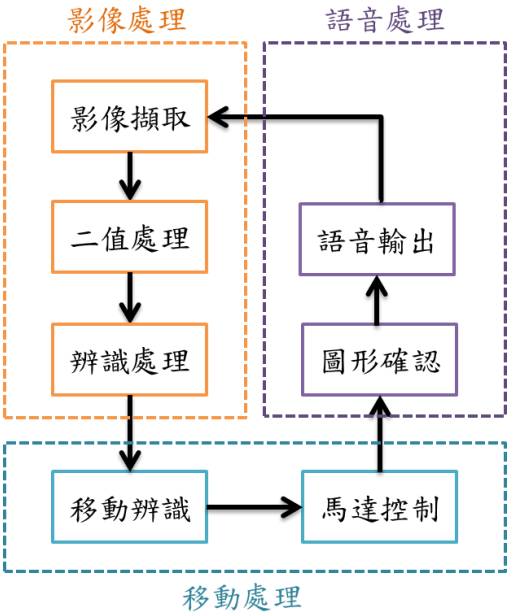

圖5：不採用ROS的程式設計流程圖

五、 採用ROS 2.0的程式設計流程

本部分所使用的智慧機器人的硬體系統架構與第四部分不採用ROS的程式設計流程所使用的智慧機器人的硬體系統架構完全相同。本部分將說明採用ROS的程式設計流程，我們使用ROS 2.0並且利用ROS 2.0節點(Node)和訊息交換的特性來設計程式。程式設計流程圖如圖6所示，我們將不採用ROS的程式設計流程的7大步驟分為3個節點，並且在語音處理節點增加計時運算步驟，與都不採用ROS的程式設計流程一樣，影像擷取、二值處理、辨識處理、移動辨識、圖形確認、語音輸出、計時運算這7大步驟都是在樹莓派中執行並且使用Python程式語言撰寫程式；只有馬達控制這個步驟是在Arduino中執行，並且使用C程式語言撰寫程式，與不採用ROS的程式設計流程最大的差別是影像處理Node、移動處理Node、語音處理Node這3個節點是平行執行。

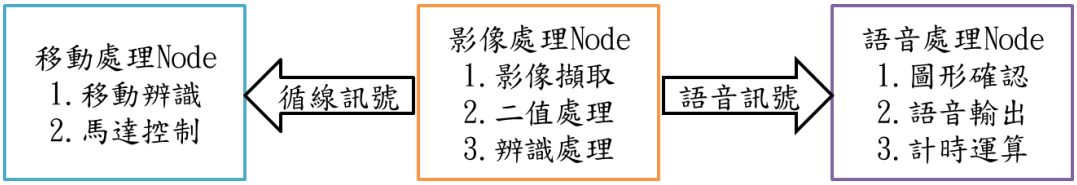

圖6：採用ROS 2.0的程式設計流程圖

肆、測試結果分析與討論

本節主要是針對第參節所設計的智慧機器人進行測試與比較採用ROS 2.0與不採用ROS的競速效能與辨識穩定度差異。我們測試的場地有兩種，圖7為無3D立體圖案的測試場地；圖8為2020城市盃智慧型自走車競賽AIoT智慧機器人競速賽項目的3D立體圖案競賽場地[城市盃智慧型自走車競賽，2020]。

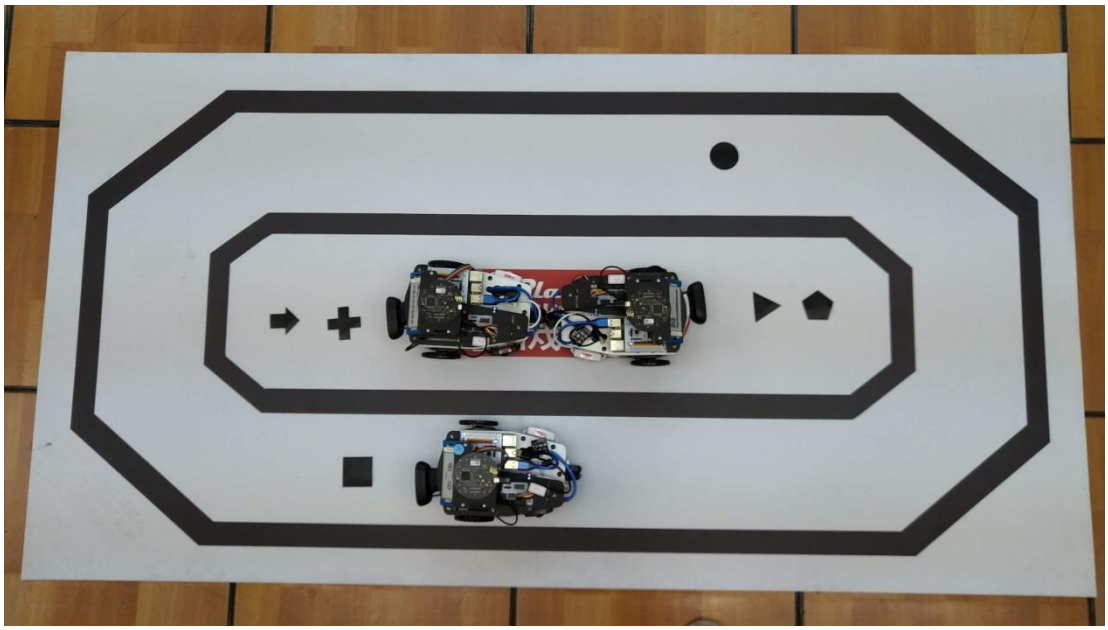

圖7：無3D立體圖案的測試場地

圖8：2020城市盃智慧型自走車競賽AIoT智慧機器人競速賽場地

一、圖7場地的測試結果與分析

我們使用圖 7 的場地進行測試，智慧機器人的硬體設計採用 ROS 與不採用 ROS 完全相同，接下來會改變的為智慧機器人馬達控制的轉速。測試的結果如表 1 所示，分別是採用 ROS 與不採用 ROS 完成圖 7 賽道 10 圈最佳狀態的馬達控制參數、完成時間及辨識圖形成功次數。由表 1 的完成時間分析在樹莓派平台中有採用 ROS 比沒有採用 ROS 的競速效能提升 71.6% [(405-236)/236*100%]；由表 1 的辨識圖形成功次數分析在樹莓派平台中有採用 ROS 比沒有採用 ROS 的辨識穩定度提升 166.6% [20/12*100%]。

表 1：採用 ROS 與不採用 ROS 完成圖 7 賽道 10 圈的最佳狀態相關數值

程式流程	馬達控制參數	完成時間	辨識圖形成功次數
採用 ROS	直行：左輪(1700)、右輪(1300) 左轉：左輪(1500)、右輪(1390) 右轉：左輪(1610)、右輪(1500)	236 秒	20 次
不採用 ROS	直行：左輪(1560)、右輪(1440) 左轉：左輪(1500)、右輪(1440) 右轉：左輪(1560)、右輪(1500)	405 秒	12 次

二、圖8場地的AIoT智慧機器人競速賽結果與分析

我們使用 2020 年 11 月 14 日的 2020 城市盃智慧型自走車競賽 AIoT 智慧機器人競速賽項目[城市盃智慧型自走車競賽,2020]來分析在樹莓派平台中有採用 ROS 比沒有採用 ROS 的競速效能與辨識穩定度。AIoT 智慧機器人競速賽的場地如圖 8，參加競賽的智慧機器人全都使用樹莓派平台當作運算系統，全部參加競賽的智慧機器人有 22 台，有 5 台在樹莓派平台中採用 ROS，其他 17 台沒有採用 ROS，競賽的規則如第貳節所說明。競賽的結果如表 2 所示，分別是採用 ROS 與不採用 ROS 完成圖 8 賽道 2 圈的完成賽道數量、最佳完成時間、走完兩圈的平均時間及辨識圖形成功平均次數。由表 2 的完成賽道數量分析在樹莓派平台中有採用 ROS 的完成率為 100%，沒有採用 ROS 的完成率只有 58.8%；由表 2 的最佳完成時間分析在樹莓派平台中有採用 ROS 比沒有採用 ROS 的競速效能提升 76.4% [(80.8-45.8)/45.8*100%]；由表 2 的走完兩圈的平均時間分析在樹莓派平台中有採用 ROS 比沒有採用 ROS 的競速效能提升 103.3% [(96.6-47.5)/47.5*100%]；由表 2

的辨識圖形成功平均次數分析在樹莓派平台中有採用 ROS 比沒有採用 ROS 的辨識穩定度提升 200.2% [4/1.8*100%]。當然這次競賽的前 3 名都由在樹莓派平台中有採用 ROS 2.0 的智慧機器人獲得。

表 2：採用 ROS 與不採用 ROS 完成圖 8 賽道 2 圈的競賽結果

程式流程	參加競賽數量	完成賽道數量	最佳完成時間	走完兩圈的平均時間	辨識圖形成功平均次數
採用 ROS	5 台	5 台	45.8 秒	47.5 秒	4 次
不採用 ROS	17 台	10 台	80.8 秒	96.6 秒	1.8 次

伍、結論

本論文以AIoT智慧機器人競速賽為例，分析在樹莓派平台中有採用ROS與沒有採用ROS的競速效能與辨識穩定度差異。由第肆節的圖7場地的測試結果與分析及圖8場地的AIoT智慧機器人競速賽競賽結果與分析，在樹莓派平台中有採用ROS比沒有採用ROS的競速效能提升70%以上及辨識穩定度提升160%以上。所以智慧機器人使用ROS 2.0能大幅提升智慧機器人的各項效能，這也表示參加智慧機器人競賽如果沒有採用ROS將很難與採用ROS的智慧機器人競爭。

經過AIoT智慧機器人競速賽的實驗測試與競賽後，未來在各項智慧機器人的競賽中可以考慮採用ROS 2.0，例如TIRT全能機器人國際邀請賽的無人車競速賽項目[TIRT國際新創機器人節，2018]就可以採用ROS 2.0並且進一步研究採用ROS 2.0是否提升競速效能與辨識穩定度。

參考文獻

中華民國經濟部，2016，產業創新研發推動五加二產業創新計畫。https://www.moea.gov.tw/Mns/populace/Policy/Policy.aspx?menu_id=32800&policy_id=4，取用日期：2020 年 11 月 1 日。

林浩鋕、蔡宗廷，2018，開源機器人作業系統第二代的介紹與未來應用。機械工業雜誌，424 期：pp17-26 。

亞洲・矽谷，2016，亞洲・矽谷願景，https://www.asvda.org/chi/index.aspx，取用日期：2020 年 11 月 1 日。

城市盃智慧型自走車競賽，2020，2020 城市盃智慧型自走車競賽。http://www.match.tpcu.edu.tw/bin/home.php，取用日期：2020 年 11 月 1 日。

陳志遠、宋守正、羅裕群，2013，智慧型避障自走車演算法設計與分析。2013 數位與科技生活創新應用學術研討會，桃園創新技術學院台灣，2013 年 12 月 27 日，pp163-167。

陳志遠、羅裕群，2014，參加校外競賽對學生學習成效影響之分析-以桃園創新技術學院資訊工程系為例。 2014 強化學生學習成效學術研討會，桃園創新技術學院台灣，2014 年 11 月 20 日，pp93-102 。

智慧機械推動辦公室，2017，智慧機械願景與使命。http://www.smartmachinery.tw/index.aspx，取用日期：2020 年 11 月 1 日。

機器人作業系統官方網站，2007，機器人作業系統。https://www.ros.org/，取用日期：2020 年 11 月 1 日。

Dirk Thomas, William Woodall, Esteve Fernandez, 2014, “Next-generation ROS: Building on DDS”, 2014 ROS Developer Conference, Chicago, Illinois, USA, 2014/9/12-13.

Dirk Thomas, William Woodall, Esteve Fernandez, 2014, “ROS 2.0: Developer preview”, 2014 ROS Developer Conference, Chicago, Illinois, USA, 2014/9/12-13.

Open Source Robotics Foundation 官方網站，2013，Open Robotics。https://www.openrobotics.org/，取用日期：2020 年 11 月 1 日。

Raspberry Pi Foundation，2009，Raspberry Pi。https://www.raspberrypi.org/，取用日期：2020 年 11 月 1 日。

TIRT 國際新創機器人節，2018，TIRT 全能機器人國際邀請賽。https://www.tirtpointsrace.org/，取用日期：2020 年 11 月 1 日。