

國立清華大學 電機工程學系

實作專題研究成果摘要

Integration of gmapping and  
robot control

機器人控制和建圖技術整合

專題領域：系統組

組別：A61

指導教授：邱偉育

組員：陳佳柔、陳玄評、王彥雅、劉世瑄、吳季陽

研究期間：108 年 07 月 01 日至 109 年 5 月底止，計 11 個月

## **Abstract**

Intelligent service robot which can solve plenty of tasks by itself will definitely play an important role in our future life. Among all sorts of smart robots, Slam robot is the one with the most refined technique in recent technology. Though whether can an robot solve actual problems has always being highly debated in academia, developing navigation and autonomy positioning system in SLAM technique is thought to be the breakthrough .

In this paper, we will demonstrate a slam robot which has the ability to create a virtual map by scanning the actual environment. Therefore the robot will become familiar to where it is and recognize the situation. On this basis, when the robot receive a message , it will navigate itself the best path to reach the goal.

We utilize the ros system and Arduino control the moving part of the SLAM robot. The raspberry pi is the platform for receive and send the message from different devices and packages ,which then evaluate and transfer the message to make it available for different device to communicate with each other. The rviz will receive the message from the lens and built an virtual map for the robot to realize where it is.

By sending the velocity of each wheel, the robot can move around and establish the environment it detects. The details of the message and the moving condition will be mention in the later part of the paper.

The testing of our robot is accurate and stable while walking around, it has showed that the remote control and the navigation of the SLAM technique is feasible and outstanding.

## 摘要

智能服務機器人發展出了很多不同的功能，被預期會在未來的生活中扮演重要角色。在許多的智慧機器人中，SLAM 機器人是一個相當完整的技術。雖然時常可以聽到究竟機器人能否解決現實生活中的實際問題的爭論，但 SLAM 機器人自主定位和導航這樣的強大功能將打破這樣的迷思，成為一個絕佳的突破口。

在文章中，我們要展示如何讓機器人走動，以及有能力在掃描實際環境後在螢幕上進行虛擬的建圖。如此一來，SLAM 機器人就有能力可以自己接收訊息並進行最佳路徑分析，以達到所輸入的指定位置。

我們主要是利用 ros 系統以及 Arduino 來控制機器人的移動部分。而 raspberry pi 是提供一個平台來進行接收和傳送訊息給不同的裝置，並可以進行運算來轉換所收到的不同種類訊息，讓不同裝置可以相互溝通。建圖的部分則是用 rviz 來進行，它接收到鏡頭偵測到的資料後，就可以建立出進行虛擬的地圖。

在傳送速度指令給每個輪子後，SLAM 機器人就可以原地轉圈進行環境的偵測及建圖。訊息的細節以及轉動情形會在原理分析的部分進行更詳細的解說。

經過不斷的測試和優化後，我們的機器人可以相當穩定且準確的進行圓周運動來建圖。這也看出 SLAM 技術的遠端控制、建圖和導航能力不但相當可行，更相當傑出。



Figure 1: 機器人身體

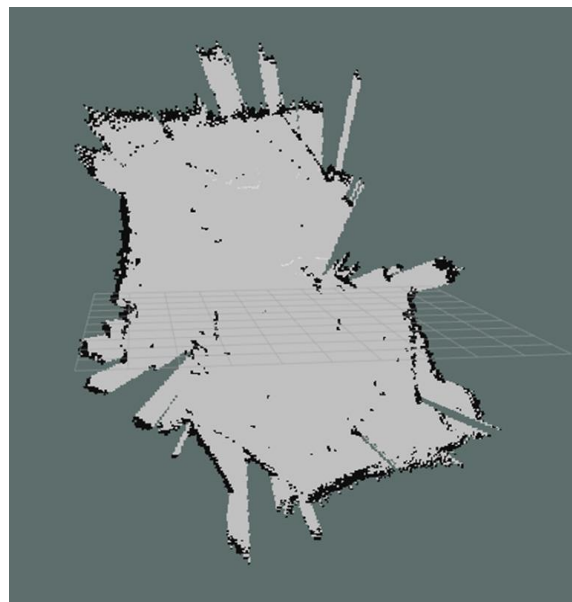


Figure 2: occupancy grid map

## 內容

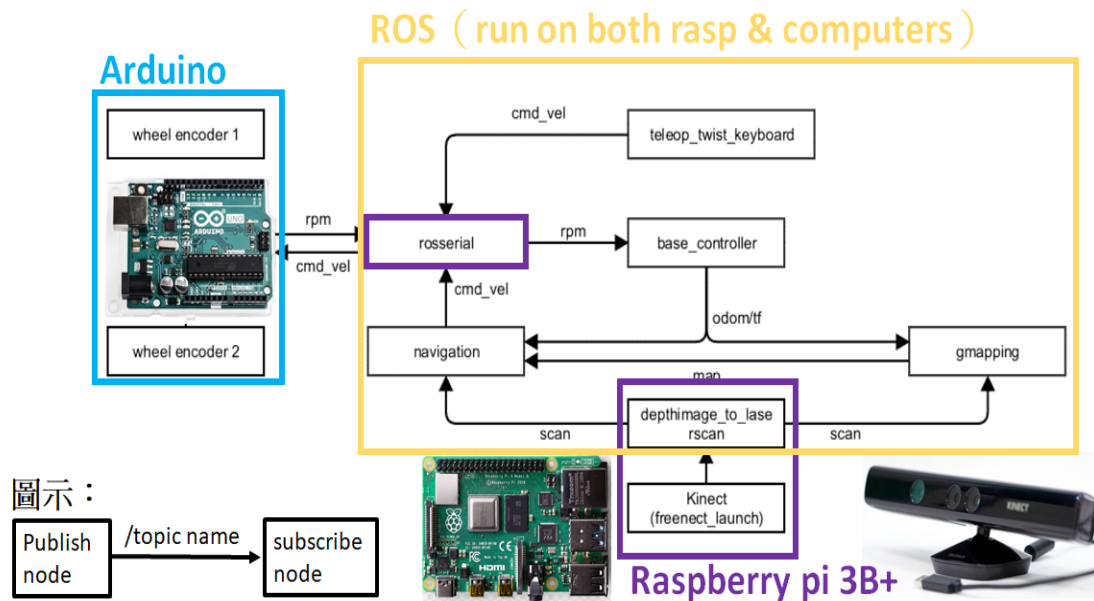


Figure 3: SLAM 機器人架構圖

- **硬體：**

鏡頭：Xbox One Kinect v1

開發板：raspberry pi 3b+、Arduino

底盤：三輪全向輪、optical rotary encoder

- **軟體：**

作業系統：Ubuntu MATE 16.04、ROS (Robot Operating System)

關於 raspberrry pi 作業系統配置的部分，我們選擇了 light 版的 Ubuntu：Ubuntu MATE 16.04 來做為作業系統，而不是 raspberrry pi 的官方作業系統 Raspbian，因為 Ubuntu 的 ARM 與 ROS 更適配。但是礙於 Ubuntu 系列並沒有正式支援 raspberrry pi，所以不是每一代的樹莓派都有適合的 Ubuntu 官方作業系統，以我們所用的 raspberrry pi 3b+ 而言，就必須要用 custom image 才能將作業

系統安裝成功。

在實作方面，為了將效能發揮到最好，我們在測試過後將鏡頭、深度資料模擬雷射資料套件以及與 Arduino 溝通的功能交給 raspberry pi 3b+，再利用 ROS 系統可在多部機器同步運行的特性，在電腦端接收雷射資料以及底盤的速度訊息，再去做建圖的計算，而電腦端所發布的命令訊息，一樣可以透過在 raspberry pi 上的訂閱節點傳達給馬達。

以下為各個模組的功能介紹、實作時面臨的問題及解決方法：

## 1. teleop\_twist\_keyboard

讓機器人可以藉由鍵盤控制，首先先從 ROS 網頁中下載 teleop\_twist\_keyboard 的套件，了解其 python code 的運作方式。我們主要做的改變是將此 python 檔傳的 cmd\_vel 指令寫入 Arduino，讓他可以將得到的 cmd\_vel 轉入 roserial 中。主要的指令有讓機器人前後左右移動以及轉變角速度。另外，也可以將速度加快或降低 10%，起始速度是用終端機輸入的。

## 2. roserial

是一個適用於 Arduino UART 的 ROS 通訊協議，可以將 Arduino 轉換為 ROS node，並直接 publish/ subscribe ROS topic。透過 roserial 來接收 teleop\_twist\_keyboard 鍵盤發出的 Vx/Vy/Vth message，並 publish 目前機器人的 Vx/Vy/Vth message。roserial 安裝於 raspberry pi 3 b+上。

## 3. Arduino

由於 raspberry pi 已經安裝 roserial，在 Arduino 中我們加入訂閱 cmd\_vel 的 subscriber，以及發布 rpm 的 publisher。接收來自使用者發布的速度訊息後，以轉換矩陣將座標軸速度轉換為輪子的期望轉速，讓機器人以指定的速度行駛。

## 4. base\_controller

包含兩個可運行的節點，接收來自底盤的速度訊息 (topic: /rpm)，進行運算轉換出里程計訊息 (odometry message) 並發布到 /odom 主題上，另一個節點則負責發布坐標系之間的轉換關係 (base\_link 坐標系及 odometry

坐標系)。

## 5. gmapping 與 depthimage\_to\_laserscan

Gmapping 是 ROS 系統常用的 2D SLAM 算法，需要訂閱/tf 與/scan 這兩個主題，其中/tf 定義了兩個剛體之間的旋轉與平移矩陣，而/scan 則是雷射雷達數據。我們使用的是 kinect 深度相機而非雷射感測器，所以使用 depthimage\_to\_laserscan 套件將深度數據轉換成雷射雷達掃描數據供 gmapping 使用。tf 資料需要由里程計判斷，編寫節點讀取里程計資訊並提供 gmapping 所需的 base\_link、odom 與 tf 的關係。滿足所有需要訂閱的資料即可運行 gmapping 建圖發布/map，並使用 Rviz 查看建圖成果。

## 6. navigation

導航功能的建構，使用 ROS 官方提供的 navigation package，內含 amcl 以及 move\_base 兩種套件，再提供 move\_base 套件所需的輸入，使 move\_base 能夠做局部路徑規劃 (local\_planner) 以及全局路徑規劃 (global\_planner)，輸出命令訊息至 cmd\_vel 主題。由圖 8 可知，move\_base 所需的輸入有：/map、/LaserScan、/tf、/odom 以及導航的目的地/goal。這些輸入由架構圖內的各模組提供。

## 心得

在本專題中，我們的實作方式偏向於 bottom-up，每個人平行式地建構不同的功能，再一起整合到開發板上。其中遭遇的問題除了各模組間的整合以外，還有電腦與開發板(raspberry)環境不同，導致在整合時遇到許多困難。但是透過本專題，我們初步了解了如何一步一步控制自己的機器人，認識了 ROS 系統的強大，還有學習如何與他人合作完成一項專案。在資料檢索方面的能力也有所提升，學習到如何選用關鍵字並過濾可用的資料，再應用到實作上。另一個困難的點是在硬體的架設和整理方面。因為我們的硬體是學長姊傳下來的，所以有些東西已經有些老舊，接線的地方也十分的脆弱，常常需要拿焊槍將原本的線接回去。除此之外，硬體常常在意想不到的地方發生問題，此時我們就要用三用電錶一個一個檢查 node 確認是否有電壓或電流通過。從一開始的束手無策，到後來越來越知道怎麼解決硬體問題。