

國立交通大學

電機與控制工程研究所

碩士論文

利用聲場特徵與 SVM 實現可結合輪式
機器人之避障與導航

Obstacle avoidance and docking of the combinational
robot using sound field characteristics and SVM

研究生：黃楷祥

指導教授：胡竹生 博士

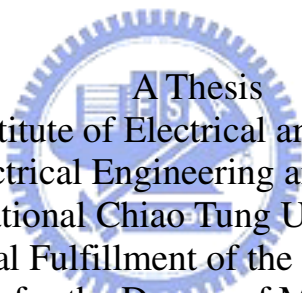
中華民國九十六年七月

利用聲場特徵與 SVM 實現可結合輪式
機器人之避障與導航

Obstacle avoidance and docking of the combinational robot
using sound field characteristics and SVM

研 究 生：黃 楷 祥 Student：Kai-Shiang Huang
指 導 教 授：胡 竹 生 博 士 Advisor：Jwu-Sheng Hu

國立交通大學
電機與控制工程學系
碩 士 論 文



A Thesis
Submitted to Institute of Electrical and Control Engineering
College of Electrical Engineering and Computer Science
National Chiao Tung University
in partial Fulfillment of the Requirements
for the Degree of Master
in

Electrical and Control Engineering

July 2007

Hsinchu, Taiwan, Republic of China

中 華 民 國 九 十 六 年 七 月

利用聲場特徵與 SVM 實現可結合輪式 機器人之避障與導航

研究生：黃 楷 祥

指導教授：胡 竹 生 博士

國立交通大學電機與控制工程研究所碩士班



本論文提出一套在室內環境中輪式機器人的導航方法與可結合式子母機器人系統。傳統機器人導航系統使用紅外線感測作為導航依據，當基地端與移動式機器人間有障礙物阻擋時，紅外線感測器導航系統將會失效，而本系統利用聲場特徵與 Support Vector Machine(SVM)分類，進行機器人障礙物偵測與導航，解決障礙物遮蔽影響。於移動平台接近基地端時，根據感測器資訊做子母機器人兩者上下的結合，使基地端成為能夠自由行動之機器人。在本論文中，亦透過嵌入式網路技術，建構以 802.11 為通訊協定之通訊介面，以傳遞基地端與移動平台間感測器的訊息。此外，並使用全向輪建構移動平台，使平台能夠有較高的運動靈活度，以利於結合運動的進行。

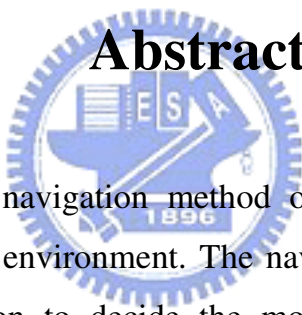
Obstacle avoidance and docking of the combinational robot using sound field characteristics and SVM

Student : Kai-Shiang Huang

Advisor : Prof. Jwu-Sheng Hu

Institute of Electrical and Control Engineering

Abstract



The thesis proposes a navigation method of the combinational wheels robot system in a complex indoor environment. The navigation system of traditional robot uses the infrared information to decide the moving method. But if the robot is sheltered by an obstacle from the base station, the infrared will not work effective. In order to solve this problem, the mobile robot can detect the obstacle and start navigation by using sound field characteristic and support vector machine(SVM) classifier in our system. When the mobile robot approach the base station, the robot and base station combine between the higher and lower levels according to the sensors information, and transfer the base station to a mobile station. Also in this thesis, the base station communicates with the mobile robot by the wireless LAN based on the embedded Ethernet technology. And we also propose the omni-directional platform offer a higher mobility in order to progress the combination of base station and mobile robot.

誌 謝

感謝上天讓我順利的完成此篇論文，在這過程中，首先要感謝我的指導老師胡竹生教授給予的指導與討論，為我解決許多疑難。也從老師身上了解了何謂研究的精神與態度，能夠從不同角度與思維對事情做詮釋與分析，大大的開拓我的視野與看待事物的方式，在此向老師獻上我由衷的謝意。在此也感謝口試委員宋開泰老師、黃育綸老師與林錫寬老師，在口試時給我各種建議與指教，讓我的論文能夠更為完整。

兩年的碩士生涯，有了一群最好的實驗室夥伴，讓我能夠在除了研究以外獲得其他的成長。劉大、興哥、宗敏、鏗元以及永融，實驗室在你們的帶領下，有了輕鬆嚴謹的研究氣氛。阿法、弘齡，能與你們一同度過這兩年日子，是我求學生涯中的榮幸。俊宇、正剛、治宏、啟揚、俊吉、明唐、瓊文，還有大學部學弟嘟嘟與 Lundy，實驗室有你們每天都充滿了活力，真的找不到比你們更優秀更好的學弟妹。

感謝我的家人，父親黃塗田、母親張素嬌、大哥黃棋偉、大姐黃美秀、二姐黃婉菱、三姐黃怡貞，謝謝你們讓我無後顧之憂的完成我的學業，家真的是我的充電站，在此獻上無比的感恩。謝謝女友依文，求學的過程有你一路相伴，與我分享生活上的點滴，彼此鼓勵，是我最幸福的時光。

要感謝的人很多，謝謝參與我這段日子的人，讓我有所成長，能夠面臨接下來更艱辛的挑戰。

目 錄

摘 要	i
Abstract	ii
誌 謝	iii
目 錄	iv
表 列	vi
圖 列	vii
第一章 緒論	1
1.1 研究動機	1
1.2 研究目標	3
1.3 論文架構	3
第二章 Support Vector Machine 統計學習原理	4
2.1 簡介	4
2.2 支持向量機分類法(SVC).....	5
2.2.1 最佳化超平面(Optimal Separating Hyperplane)	5
2.2.2 線性不可分(Linear non-separable case).....	7
2.3 SVM 實驗測試.....	9
第三章 聲源特徵偵測與運動判定	12
3.1 簡介	12
3.2 光流感測器定義與原理	12
3.3 聲源偵測判定法	14
3.3.1 聲源方位偵測.....	14
3.3.2 障礙物判定.....	16
3.4 輪式機器人運動路徑設定	18
3.4.1 光流偵測碰撞法.....	18
3.4.2 SVM 障礙物偵測與碰撞法.....	21
3.4.3 LOS 聲源偵測法.....	23
3.5 基地端(base)與輪式機器人結合動作	25

3.5.1 全方位運動平台分析.....	25
3.5.2 結合運動設計.....	30
第四章 系統軟硬體設計與實現.....	32
4.1 整體系統架構.....	32
4.2 實驗平台.....	33
4.2.1 軟體開發環境.....	35
4.2.2 程式開發流程.....	39
4.3 Base 與輪式機器人平台之架構.....	39
4.3.1 Base 與移動平台架構.....	40
4.3.2 電路介紹.....	41
第五章 實驗結果與展望.....	45
5.1 實驗結果.....	45
5.1.1 聲源方位判斷.....	45
5.1.2 SVM 障礙物判斷.....	47
5.2 未來展望.....	49
Reference:.....	50



表 列

表 1	SVM 分類結果表.....	17
表 2	OMAP 5912 OSK HARDWARE LIST	35
表 3	聲源方位判斷比較表(1).....	46
表 4	聲源方位判斷比較表(2).....	47
表 5	SVM 部分訓練資料(1).....	47
表 6	SVM 部分訓練資料(2).....	48
表 7	SVM 部分訓練資料(3).....	48
表 8	SVM 部分訓練資料(4).....	48
表 9	SVM 部分訓練資料(5).....	48
表 10	SVM 部分訓練資料(6).....	49



圖 列

圖 1 (A)感知器模型 (B)超平面(HYPERPLANE)將資料分為兩個區間	4
圖 2 最佳化超平面(OPTIMAL SEPARATING HYPERPLANE)	5
圖 3 標準超平面與支持向量示意圖	6
圖 4 輸入向量映射高維空間	8
圖 5 線性可分示意圖	10
圖 6 線性不可分示意圖	10
圖 7 光流感測器與平台的幾何關係圖	13
圖 8 麥克風與聲源關係圖	15
圖 9 訊號前段與中段之 TIME-DELAY(聲源於麥克風正前方).....	16
圖 10 標準差與位置關係圖	17
圖 11 平台行走示意圖(1).....	19
圖 12 碰撞行走流程圖	20
圖 13 平台行走示意圖(2).....	22
圖 14 SVM 碰撞行走流程圖	23
圖 15 LOS 聲源判斷流程圖	24
圖 16 不同型態的全相輪	25
圖 17 全向輪平台簡圖	26
圖 18 全向輪簡圖	26
圖 19 運動模式示意圖	29
圖 20 光敏感測器示意圖	30
圖 21 光敏感測器感測圖	31
圖 22 整體系統示意圖	32
圖 23 系統架構圖	33
圖 24 OMAP 5912 架構圖	34
圖 25 OMAP 5912 OSK 外觀	35
圖 26 SDRAM 記憶體映射	38
圖 27 BASE 與移動平台機構圖(A) (B).....	40
圖 28 BASE 端照片	40
圖 29 全方位移動式平台照片	41
圖 30 PIC ETHERNET 嵌入式網路板	41
圖 31 感測器控制板	42
圖 32 馬達控制板	42
圖 33 電源模組與電池	43
圖 34 光流感測器	43
圖 35 馬達組	43
圖 36 ACCESS POINT	44
圖 37 複頻聲音訊號	45

圖 38	無障礙物聲源判斷	45
圖 39	有障礙物聲源判斷	47



第一章 緒論

1.1 研究動機

機器人的應用從一般工業進入家庭環境，於室內複雜環境中的活動相較於工廠裡是一個重要的研究議題。為能夠有效控制機器人行為，機器人室內定位成為一個關鍵技術。使用各種不同的感測器可以達到定位效果，譬如：超音波感測器、雷射掃描器、攝影機、輪式編碼、RFID、Wireless LAN 等等[1-6]。使用上述之感測器確實都可以達到定位效果並進行導航，但是實際上卻都有相當的限制。雷射掃描器之成本過於昂貴，即使擁有高精準度卻不適合普及於家庭中；攝影機需要針對每個環境重新個別建構系統，且容易受光線變化影響；Wireless LAN 在室內環境之定位精度相對於其他系統則比較不好；而輪式編碼系統則會因為輪子的打滑而會有誤差的累積。

除了上述方法，使用麥克風陣列，利用聲場特徵來進行絕對位置估測的研究也已行之有年。以 Generalized cross correlation(GCC)為基礎作機器人定位的研究[7]亦討論多時。使用聲波進行機器人之絕對定位面臨最大困難在於聲波在室內環境中，反射、繞涉等現象較為明顯，且傳遞行為複雜，這些誤差對於位置的估計都會造成不確定性，因此本文將不使用絕對定位法而使用相對位置來進行導航。

以擁有自動充電功能之機器人為例，機器人若想要能夠快速精確的走回充電站，則須知道充電站之座標方能有效率的往充電站移動。但因為定位之困難，現今市售的智慧型吸塵機器人大多無使用座標定位系統，而直接使用區域性感測導引(如日本 sony 的 Aibo、台灣松騰實業的 RV10、美國 iRobot 公司的 Roomba 系列、德國 Kacher 的 RC3000)[8-10]，亦即使用紅外線引導、視覺辨別、超音波測距等方法。當發覺電力不足時，機器人以特定方式行走至充電站附近，在能夠接收紅外線引導範圍內進行導引。此方法有幾個缺點，如機器人與充電站相距過遠，超出引導範圍許多，或者充電站與機器人間有障礙物阻，都將會提高機器人返回基地的時間與難度。

在本文中將提出一簡易方法，解決上述問題。當一點聲源發聲，聲波到達麥

克風的方向與距離會直接反應在聲波訊號之相位差異(phase-difference)與大小比(magnitude ratio)上。在充電站設置喇叭，打出特定的複頻訊號，機器人設置雙聲道麥克風，在收到聲波訊號時，解析各個頻率之相位差異與大小比，取其中magnitude 最大的頻率之相位差，作為機器人下一步前進的依據，慢慢往充電站(聲源)移動，而並不需要知道機器人之絕對位置。另外系統還加入利用聲音判斷機器人是否面臨障礙物的阻礙。我們相信不同頻率在經過障礙物時的繞射現象會有所不同，也就是說麥克風收到的訊號裡每個頻率的 phase-difference 會與沒有障礙物(Non-line of sight)時有所不同。透過 Support vector machine(SVM)的訓練分類，可以將此兩種情況分辨出來，作為障礙物判別的依據。

本文中我們將實作一概念性結合式吸塵機器人，吸塵式機器人不單純只是一輪式機器人，而是在與充電站結合後，能帶著充電站隨意走動之載具。充電站可以是高整合系統之機器人或者單純為一個垃圾桶、櫃子、飲料台等各式功能。目的在於吸塵式機器人平時為機器人之一部分，在不需本體機器人之功能時，可以脫離本體進行吸塵的工作。



1.2 研究目標

本論文目標如下：

1. 實現聲場特徵導航與障礙物偵測系統。
2. 使用全向輪來建構一個擁有高運動自由度的全方向運動機器人平台，與可結合式機器人機構設計與方法。
3. 整合嵌入式網路技術，透過無線網路作為輪式機器人與 base 端的溝通方式，負責傳遞各個感測器(光敏電阻、紅外線感測器等)之訊息。

1.3 論文架構

整篇論文大致包含了下三個部分，分別為統計學習理論、聲場特徵與機器人運動方法與實驗平台的架構與硬體。於第二章介紹 Support vector machine 統計學習的原理與方法，第三章中介紹聲場特徵判斷方法以及機器人結合運動方法，最後第四章與第五章將演算法實現於使用的實驗平台與硬體介紹，並呈現實驗結果與未來展望。



第二章 Support Vector Machine 統計學習原理

2.1 簡介

統計學習理論[11-12]的演進過程，從第一個學習機器開始到近代的各種不同的統計學習方法，大致上可分為四個階段。1960 年由 F. Rosenblatt 首先提出了第一個名為感知器(perceptron)的學習機器模型，它是一個類似類神經模型的架構，用來解決資料分類的問題。基本概念如圖 1 所示。

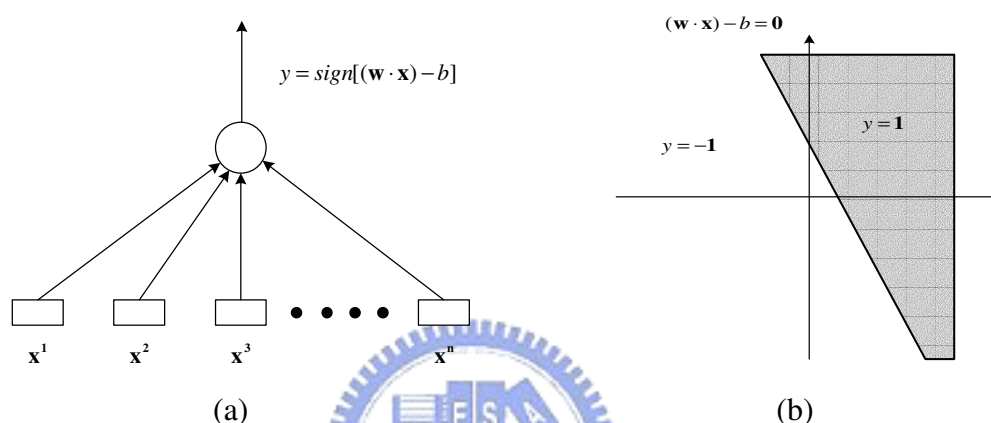


圖 1 (a)感知器模型 (b)超平面(hyperplane)將資料分為兩個區間

接下來的 1960~1970 年代，在這十年中基本的統計學習理論逐漸被建構成型，各種不同的學習機器慢慢被提出，如 B. Widrow 所提出的 Madaline、K. Steinbuch 所提的 learning matrices。到了 1980 年代，類神經網路(Neural Networks)成為了主流，1986 年 LeCun 及 Rumelhart, Hinton, Williams，等人同時提出了所謂的倒傳遞類神經網路法(back-propagation neural network method)，代替了原本由 McCulloch-Pitts 提出的模型。直至 1990 年到現今，類神經網路已經被稱為多層感知器，多數人將重心轉移至新的學習方法，諸如 radial basis functions method，並回歸到學習理論的基礎，發展新的演算法。

本文中所使用到支持向量機(Support Vector Machine)是眾多統計學習機器中的一種方法，其基礎理論由 Vapnik 所發展，目前較常拿來應用於分類。在有限個已知的兩群資料，透過 SVM 的訓練我們可以得到一個新的支持向量機，當有新的資料輸入時，SVM 就可以對其做分類。下面將詳細介紹 SVM 演算法。

2.2 支持向量機分類法(SVC)

在分類的方法中，主要可分為兩種，一種是將資料分類為數個類別，而另一個則是將資料一分為二，支持向量機的分類則是屬於後者。在兩群已知的資料中，可以使用一個函數將其區隔開來，這個函數我們稱他為分類器，理論上來看此種分類器將會有無限多個，但是擁有最大 margin(兩群中分別最靠近分類器的點的距離)的分類器只有一個，如圖 2 所示。

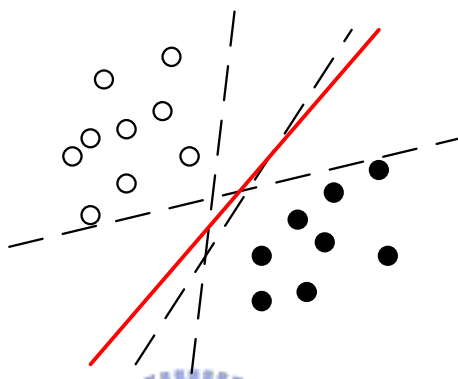


圖 2 最佳化超平面(optimal separating hyperplane)

我們稱此分類器為最佳化超平面，而 SVM 的訓練過程，也就是找出此超平面，再使用這個超平面來對新的、未知的資料做分類。

2.2.1 最佳化超平面(Optimal Separating Hyperplane)

假設我們有一群訓練資料

$$D = \{(x_1, y_1), \dots, (x_l, y_l)\} \quad (1)$$

可以被下式的超平面分為兩類

$$(w \cdot x_i) - b = 0 \quad (2)$$

當此超平面能夠完全無誤並且使最接近平面的向量離平面的距離為最大，我們就稱之為最佳化超平面。有了最佳化超平面限制了 w 與 b ，接著定義支持向量(support vector)與標準超平面(canonical hyperplane)如下：

$$\begin{cases} (w \cdot x_i) - b = 1 \\ (w \cdot x_i) - b = -1 \end{cases} \quad (3)$$

在我們的資料群裡符合式子(3)的 x_i ，稱為支持向量，而式子(3)即代表標準超平面，其幾何概念如圖 3 所示。

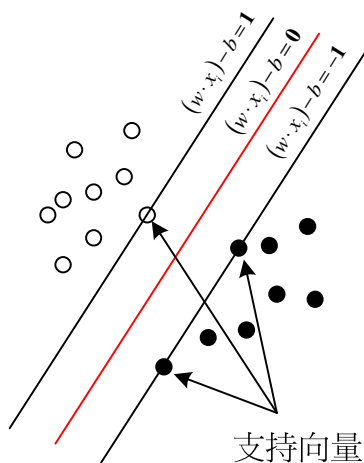


圖 3 標準超平面與支持向量示意圖

很容易的可以得到

$$\begin{aligned} (w \cdot x_i) - b &\geq 1 && \text{if } y_i = 1 \\ (w \cdot x_i) - b &\leq -1 && \text{if } y_i = -1 \end{aligned} \quad (4)$$

將式(4)重新整理導出式(5)

$$y_i [(w \cdot x_i) - b] \geq 1, \quad i = 1, \dots, l \quad (5)$$

從支持向量到最佳化超平面的距離可以表示為式(6)

$$d = \frac{|(w \cdot x_i) - b|}{\|w\|} \quad (6)$$

根據式(3)與式(6)，margin ρ 就等於

$$\begin{aligned} \rho(w, b) &= 2 \times d \\ &= \frac{2 \times |(w \cdot x_i) - b|}{\|w\|} \\ &= \frac{2}{\|w\|} \end{aligned} \quad (7)$$

所以如果想要得到最大的 margin，就必須對 $\|w\|$ 做最小化處理，此為二次項規劃

最優化問題，也就是說將 $\|w\|$ 最小化等於對 $\frac{1}{2} \|w\|^2$ 做最小化處理。因此在式(5)的限

制條件下，對 $\frac{1}{2}\|w\|^2$ 最小化處理就可以得到最佳化超平面。

引進拉格朗日乘數(Lagrange multipliers)來解決這個極值問題，可以列出式(8)

$$L(w, b, \alpha) = \frac{1}{2}(w \cdot w) - \sum_{i=1}^l \alpha_i \{[(w \cdot x_i) - b]y_i - 1\} \quad (8)$$

其中 α_i 為拉格朗日乘數。在極值點將會有一組最佳解 w_0 、 b_0 、 α^0 滿足下列式子

$$\frac{\partial L(w_0, b_0, \alpha^0)}{\partial b} = 0 \quad \rightarrow \quad \sum_{i=1}^l \alpha_i^0 y_i = 0, \quad \alpha_i^0 \geq 0, \quad i = 1, \dots, l \quad (9)$$

$$\frac{\partial L(w_0, b_0, \alpha^0)}{\partial w} = 0 \quad \rightarrow \quad w_0 = \sum_{i=1}^l y_i \alpha_i^0 x_i, \quad \alpha_i^0 \geq 0, \quad i = 1, \dots, l \quad (10)$$

將式(9)、式(10)代入式(8)，可以得到新的拉格朗日函數

$$W(\alpha) = \sum_{i=1}^l \alpha_i - \frac{1}{2} \sum_{i=1}^l \sum_{j=1}^l \alpha_i \alpha_j y_i y_j (x_i \cdot x_j) \quad (11)$$

在式(9)的限制條件下，求式(11)的解，就可以得到一組 $\alpha_0 = (\alpha_1^0, \alpha_2^0, \dots, \alpha_l^0)$ ，而根據式(10)，此組拉格朗日乘數變決定了最佳化超平面，如下列式子

$$w_0 = \sum_{i=1}^l y_i \alpha_i^0 x_i$$

$$b_0 = \frac{1}{2} [(w_0 \cdot x^*(1)) + (w_0 \cdot x^*(-1))] \quad (12)$$

其中 $x^*(1)$ 代表第一個類別裡的任一個支持向量，而 $x^*(-1)$ 代表另一個類別裡的任一個支持向量。

得到了 w_0 與 b_0 便決定了我們的最佳化超平面為 $(w_0 \cdot x) - b_0 = 0$ ，當我們獲得新的一筆資料，對他做下列的運算

$$f(x) = \text{sign}[(w_0 \cdot x) - b_0] \quad (13)$$

便可以知道此筆新資料屬於哪個類別。

2.2.2 線性不可分(Linear non-separable case)

在實際情況中，資料通常為線性不可分的，也就是說使用線性分類，沒有辦法把資料完全無誤的分為兩類。為了解決這個問題，SVM 使用非線性函數，將低維度輸入向量空間 x ，映射到一個高維度空間 z ，接著於 z 空間中尋找相對應的

最佳化超平面，而此非線性函數稱為 Kernel Function。如圖 4 所示：

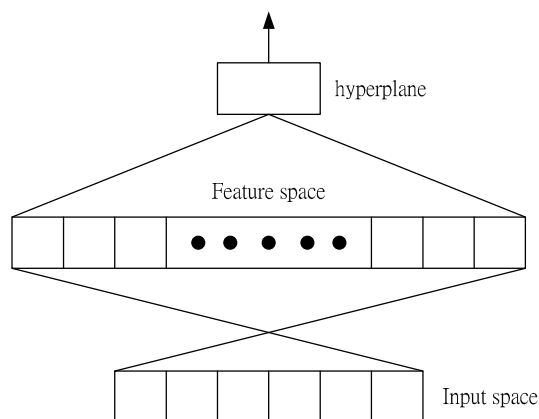


圖 4 輸入向量映射高維空間

將式(11)經過 Kernel Function 的映射，求得新的拉格朗日乘數

$$W(\alpha) = \sum_{i=1}^l \alpha_i - \frac{1}{2} \sum_{i=1}^l \sum_{j=1}^l \alpha_i \alpha_j y_i y_j K(x_i, x_j) \quad (14)$$

求得新的拉格朗日乘數後，便決定了 z 空間中的最佳化超平面

$$\begin{aligned} (w_0 \cdot x) &= \sum_{i=1}^l \alpha_i y_i K(x_i, x) \\ b_0 &= \frac{1}{2} [(w_0, x_1^*) + (w_0, x_{-1}^*)] = \frac{1}{2} \sum_{i=1}^l \alpha_i y_i [K(x_i, x_1^*) + K(x_i, x_{-1}^*)] \end{aligned} \quad (15)$$

於是式(13)經過映射後，變為下式

$$f(x) = \text{sign} \left[\sum_{i=1}^l \alpha_i y_i K(x_i, x) - b_0 \right] \quad (16)$$

根據式(16)，便能夠對新輸入的資料做分類。

底下我們將介紹常見的非線性映射函數。

1. 多項式(Polynomial)

$$K(x, x') = (x \cdot x' + 1)^d \quad (17)$$

其中 d 為多項式級數。

2. 高斯徑向基底函數(Gaussian Radial Basis Function, RBF)

$$K(x, x') = \exp \left(-\frac{\|x - x'\|^2}{2\sigma^2} \right) \quad (18)$$

σ 為 RBF 之半徑長度參數。

3. 指數徑向基底函數(Exponential Radial Basis Function, eRBF)

$$K(x, x') = \exp\left(-\frac{\|x - x'\|}{2\sigma^2}\right) \quad (19)$$

同 2.， σ 為 RBF 之半徑長度參數。

4. 傅立葉級數(Fourier Series)

$$K(x, x') = \frac{\sin(N + \frac{1}{2})(x - x')}{\sin\left(\frac{1}{2}(x - x')\right)} \quad (20)$$

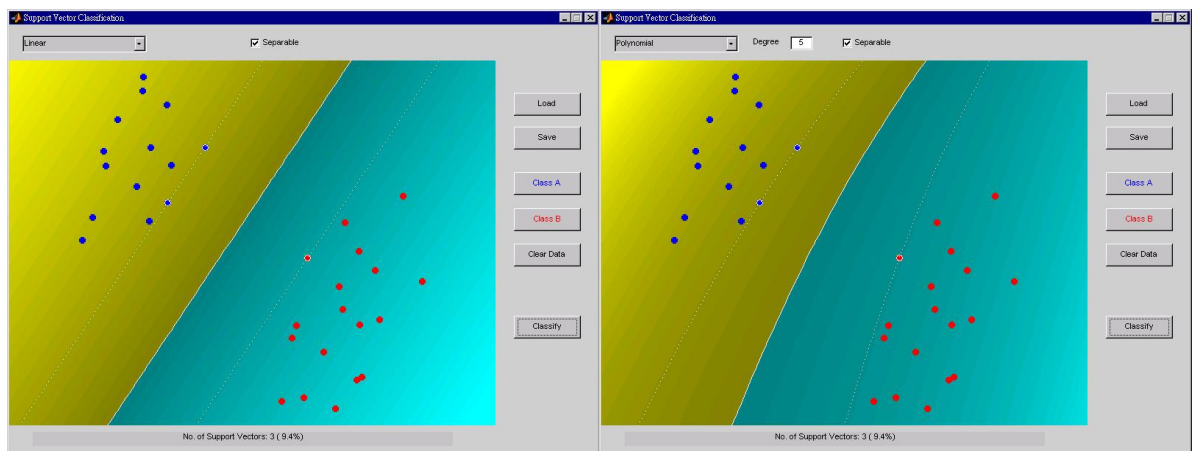
傅立葉級數可視為將空間映射至 $2N+1$ 維的函數。

2.3 SVM 實驗測試

SVM 訓練工具的發展已經有段時間，現在有各種不同版本，如：C++、C、Matlab、線上訓練機，等不同的訓練工具，本文使用英國南開普敦大學 ISIS 所發展的 Matlab toolbox 作為我們的訓練機[13]，先使用 Matlab 求出已知資料的最佳化超平面，再用嵌入式系統實現即時資料分類。

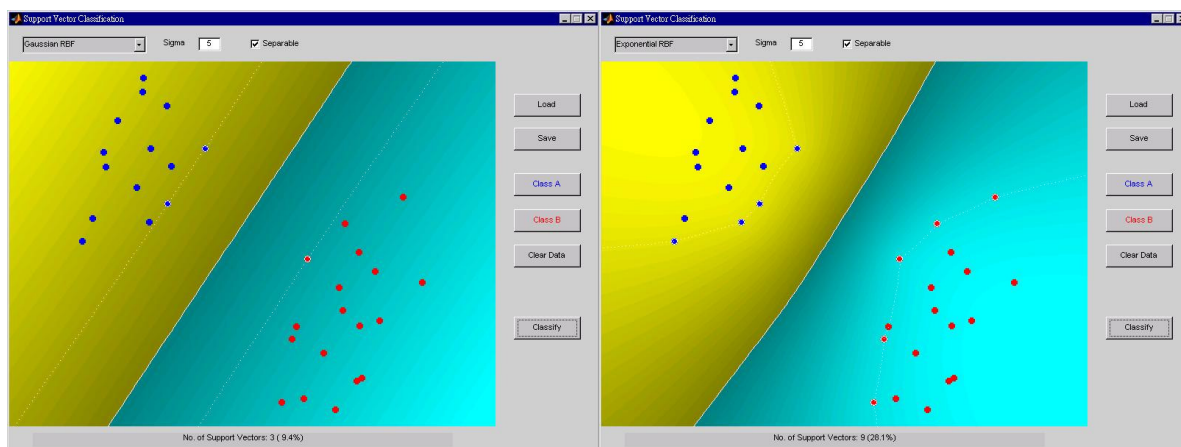
以下為簡單的二維空間線性可分與線性不可分的兩種資料，使用不同分類方式的情況：

1. 線性可分



(a)Linear

(b)Polynomial with degree 5

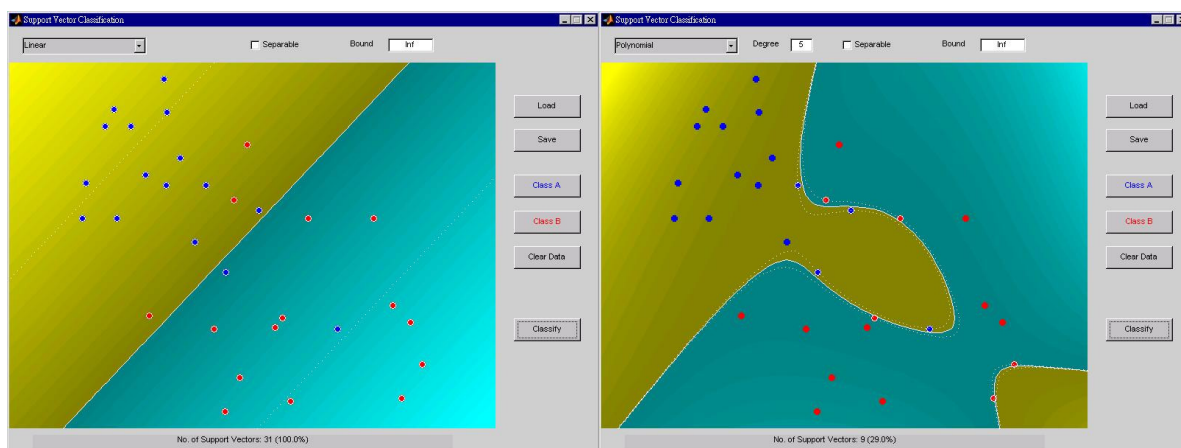


(c)RBF with radius 5

(d)eRBF with radius 5

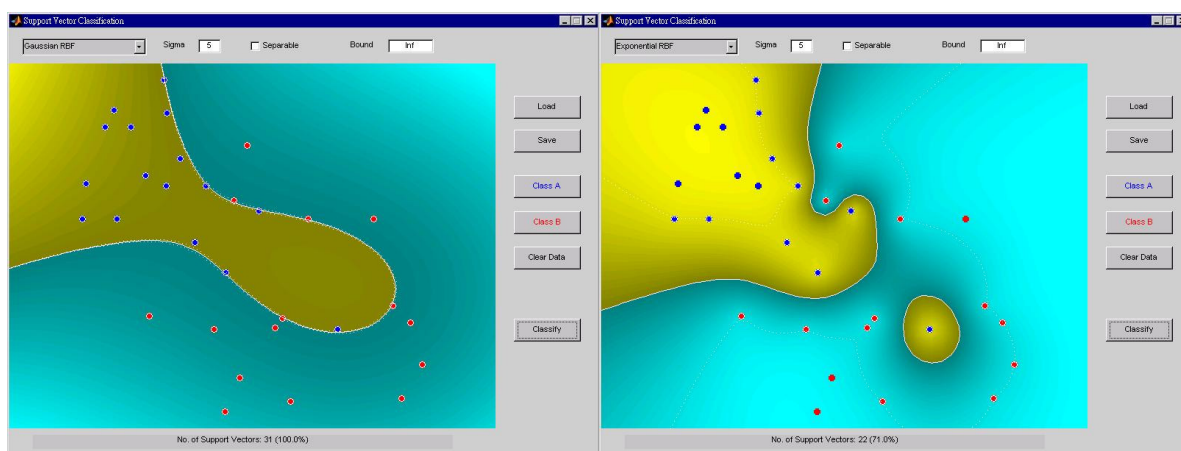
圖 5 線性可分示意圖

2. 線性不可分



(a)Linear

(b)Polynomial with degree 5



(c)RBF with radius 5

(d)eRBF with radius 5

圖 6 線性不可分示意圖

如圖 5 所示，我們可以發現在線性可分的情況下，各種不同的分類方法，效果都差不多，都能夠成功無誤的將資料分類。而在線性不可分的情況，很明顯的 Linear 無法將資料成功的分類，需透過其他非線性映射，才能較有效果的分開資料，其中又以 eRBF 的效果最好。



第三章 聲源特徵偵測與運動判定

3.1 簡介

理論上來說，聲音的複雜傳遞行為可以反應在麥克風陣列所收到訊號的相位差異與大小比上，根據相位差異(Time-delay)可以容易的判斷出聲音相對於麥克風陣列的方位。且聲源相對於麥克風在 line of sight 與 non-line of sight 的情況下，各個頻率的相位差異將會有所變動，利用這個特性，我們使用 SVM 來作為機器人的分類器，作為障礙物偵測的依據。並且由相位差異所得到之資訊，可以作為機器人行走的依據，搭配光流感測器的碰撞偵測，順利的到達 base 端進行結合的動作。

3.2 光流感測器定義與原理

光流(Optical Flow)[14]是指，對於兩張以上連續影像中的物體，預測其運動方向及大小。透過比較兩張影像之間的影像強度來獲得資訊，通常在兩張影像中必須要擁有大部分相同的內容，才能夠確保這個方法的正確。使用光流感測器來作碰撞偵測的工具，是因為根據機器人的不正常移動就可以判斷是否面臨障礙物，相較於其他感測器，如：紅外線、超音波、微動開關等，都會有死角的問題。

使用光流感測器，我們可以得到其位移資訊。根據其與機器人運動平台的幾何關係，我們可以得到下列的運動學方程式：

$$\begin{aligned} V_x &= V_{rx} - w_r \cdot r_y \\ V_y &= V_{ry} + w_r \cdot r_x \end{aligned} \quad (21)$$

其中 V_{rx} 和 V_{ry} 是機器人平台中心的移動速度， w_r 是平台的旋轉角速度， V_x 和 V_y 為光流感測器所量出的移動速度，而 $r = r_x + r_y$ 為感測器與平台中的的距離向量。如圖 7 所示。

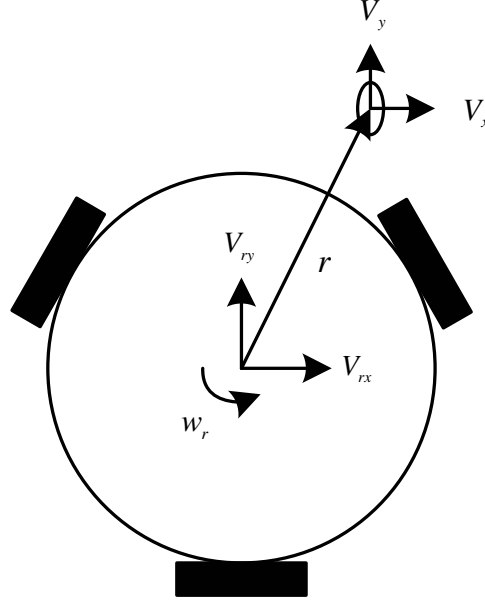


圖 7 光流感測器與平台的幾何關係圖

將兩個光流感測器放置於平台上，經過整理可以得到下列的方程式：

$$\begin{aligned}
 V_{rx} &= V_{1x} + w_r \cdot r_{1y} \\
 V_{ry} &= V_{1y} - w_r \cdot r_{1x} \\
 V_{rx} &= V_{2x} + w_r \cdot r_{2y} \\
 V_{ry} &= V_{2y} - w_r \cdot r_{2x}
 \end{aligned} \tag{22}$$

為求平台的速度與角速度，將上述式子改寫成矩陣向量的形式，整理成為下面的樣子：

$$\begin{bmatrix} 1 & 0 & -r_{1y} \\ 0 & 1 & r_{1x} \\ 1 & 0 & -r_{2y} \\ 0 & 1 & r_{2x} \end{bmatrix} \cdot \begin{bmatrix} V_{rx} \\ V_{ry} \\ w_r \end{bmatrix} = \begin{bmatrix} V_{1x} \\ V_{1y} \\ V_{2x} \\ V_{2y} \end{bmatrix} \tag{23}$$

可以知道(23)式是一個過度決定(over-determine)的系統，因此使用最小平方誤差法(least square error method)來求得此方程式的最佳近似解，也就是求得平台的速度與角速度 V_{rx} 、 V_{ry} 、 w_r ：

$$\begin{bmatrix} V_{rx} \\ V_{ry} \\ w_r \end{bmatrix} = \left\{ \begin{bmatrix} 1 & 0 & -r_{1y} \\ 0 & 1 & r_{1x} \\ 1 & 0 & -r_{2y} \\ 0 & 1 & r_{2x} \end{bmatrix}^T \begin{bmatrix} 1 & 0 & -r_{1y} \\ 0 & 1 & r_{1x} \\ 1 & 0 & -r_{2y} \\ 0 & 1 & r_{2x} \end{bmatrix} \right\}^{-1} \begin{bmatrix} 1 & 0 & -r_{1y} \\ 0 & 1 & r_{1x} \\ 1 & 0 & -r_{2y} \\ 0 & 1 & r_{2x} \end{bmatrix}^T \begin{bmatrix} V_{1x} \\ V_{1y} \\ V_{2x} \\ V_{2y} \end{bmatrix} \quad (24)$$

得到了平台的速度與角速度，再透過積分的運算，就能夠得到平台的移動資訊與旋轉量：

$$\begin{aligned} \theta_r &= \int (w_r) dt \\ X_r &= \int (V_{rx} \cos \theta_r - V_{ry} \sin \theta_r) dt \\ Y_r &= \int (V_{rx} \sin \theta_r + V_{ry} \cos \theta_r) dt \end{aligned} \quad (25)$$

透過機器人平台的移動與旋轉資訊，我們就可以判斷平台是否被障礙物阻擋。在控制馬達等速運動的情況下，平台在單位時間內的位移量應為定值 S ，若由光流感測器所偵測出位移量小於 S ，即代表平台遭遇阻礙。

3.3 聲源偵測判定法

3.3.1 聲源方位偵測

在做 DOA(Direction of arrival)的判斷，主要分為兩類技術，第一類為 TDE(Time Delay Estimation)，第二類是 Eigenspace Method。兩類方法主要差別在於 TDE 為針對一個聲音來源作方位的辨別，無法在同一時間內辨別出多個聲源的位置。而 Eigenspace Method 就可以對多個聲源作方位的辨別。而根據不同的應用可以選用不同的方法。在本篇論文中，因為聲音來源只有一個(由 base 端所發出)，且為了簡化硬體成本與符合即時應用，所以我們使用第一類方法 TDE，就可以達到我們所需要的目標。

TDE 法通常使用兩個麥克風即可。聲源從不同的方向傳遞到兩個麥克風，會造成不同的響應，TDE 即是利用聲音在空間中傳遞時所造成的時間差特性來做方位的估計。如圖 8 所示：

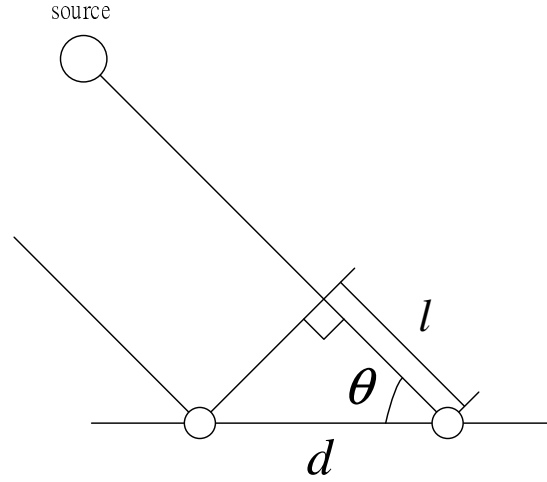


圖 8 麥克風與聲源關係圖

根據關係圖，我們可以很清楚表示出式(26)：

$$\theta = \cos^{-1} \frac{l}{d} \quad (26)$$

其中 θ 為聲源相對於麥克風的角度， d 為兩個麥克風間的距離為一定值， l 是聲源在兩個麥克風的時間差中所走的距離，如此機器人便能根據 θ 往聲源方向移動。

根據 GCC 方法，可以求出時間差 t ，所以 $l = t \times c$ ，其中 c 為聲速。GCC 能針對聲音的特定頻帶作運算，作比較精準的時間差估算，但在我們的系統中嘗試了另一個較為直覺且簡單的方法來估測聲源的方位。由 base 端經由喇叭打出一包含六個頻率的複頻音源，由機器人上的兩顆麥克風收到之後，經由 FFT 我們可以得到六個頻率的 phase 與 magnitude。計算兩個聲道的 phase-difference，根據此 phase-difference 即可以求出時間差 t ，如式(27)。

$$S_{delay} = \frac{phase_{diff}}{2\pi} \times T_f \times S_{rate} \quad (27)$$

$$t = \frac{S_{delay}}{S_{rate}}$$

其中 S_{delay} 為兩個聲音訊號之間的 sample delay 個數， $phase_{diff}$ 為訊號的 phase difference， T_f 為六個不同頻率各自的週期， S_{rate} 為系統對麥克風的 sampling rate。理想的情況下，我們可以得到六個相同的 S_{delay} ，但是在反射、繞射等環境影響，

S_{delay} 並不會如預期的一樣相同，會有所變動，所以我們取六個頻率中 magnitude 最大的那個頻率當作基準，以他的 phase-difference 作為聲源方位判斷的依據。為了減少反射等影響，我們取訊號的最前端 512 點來做 FFT 運算，讓 phase 的判斷更為準確。如圖 9 所示：

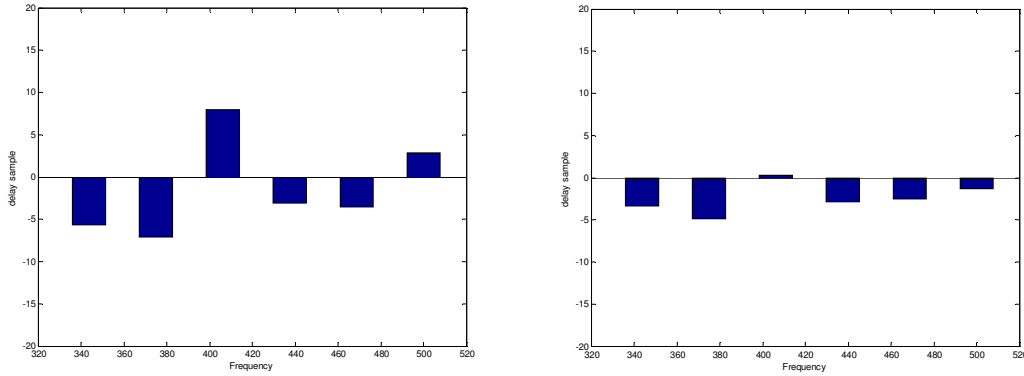
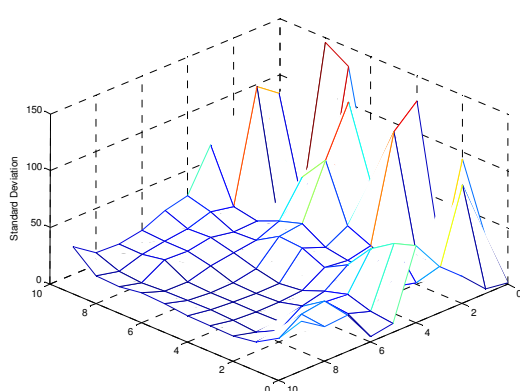


圖 9 訊號前段與中段之 time-delay(聲源於麥克風正前方)

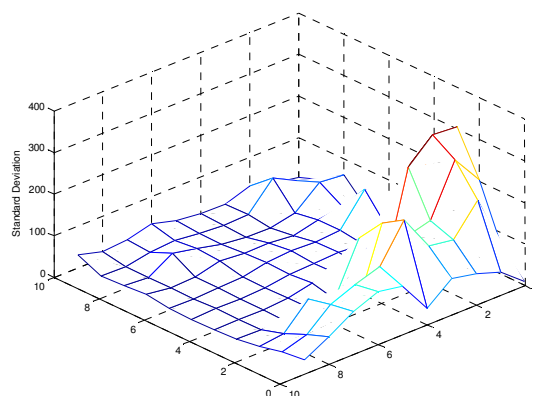
3.3.2 障礙物判定

得到六個頻率的 time delay 後，我們可以進行 SVM 的分類，來辨別移動平台與 base 端之間為 line of sight(LOS)或是 non-line of sight(NLOS)。在做 SVM 的分類之前，須先對 SVM 分類器作訓練，訓練方法如下：

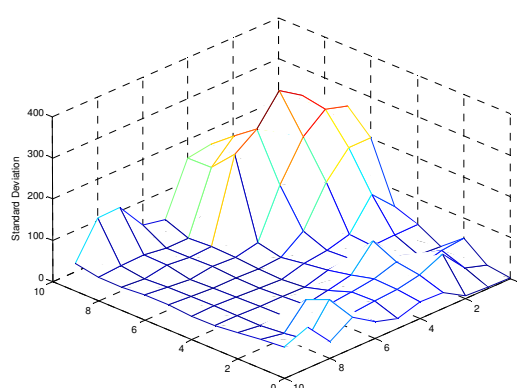
1. 固定聲源位置，使移動平台與聲源間為 LOS，使平台正對聲源、聲源位於平台左方 30° 、聲源位於平台右方 30° ，分別錄製 30 個點，得到 90 筆 LOS 的訓練資料 $\mathbf{a}_i = (x_1, x_2, x_3, x_4, x_5, x_6)$ 。
2. 將障礙物放置於場地中，於平台與聲源位置間為 NLOS 的地方，將其切割為 10×11 、邊長為 10 公分的小方格，使平台正對聲源、聲源位於平台左方 30° 、聲源位於平台右方 30° ，分別錄製 110 個點，得到 330 筆 NLOS 的訓練資料 $\mathbf{b}_i = (x_1, x_2, x_3, x_4, x_5, x_6)$ 。
3. 將得到的所有訓練資料作分析，分別取 \mathbf{a}_i 和 \mathbf{b}_i 的標準差 S_a 與 S_b ，很合理的發現，因為聲源與平台為 LOS， S_a 都比較小，而在靠近障礙物物的 6×11 的方格裡， S_b 有較明顯的變化，所以我們只取這 66×3 筆資料，作為 NLOS 的訓練資料。如圖 10 所示：



平台正對聲源



聲源位於平台右方



聲源位於平台左方

圖 10 標準差與位置關係圖

4. 最後在將 90 筆 a_i 與 198 筆 b_i 作為 SVM 分類器的訓練資料，找出其最佳化超平面。

使用不同的 kernel function，會有不一樣的分類效果，嘗試以 Polynomial、Gaussian Radial Basis Function、Exponential Radial Basis Function 三種 kernel 作分類訓練，對 42 筆 NLOS 資料進行分類，結果如表 1 所示：

表 1 SVM 分類結果表

	錯誤判斷	正確判斷	正確率
Poly	5	37	88.1%
rbf	5	37	88.1%
e-rbf	1	41	97.6%

因此我們使用 Exponential Radial Basis Function 作為訓練的 kernel。

3.4 輪式機器人運動路徑設定

3.4.1 光流偵測碰撞法

利用頻譜裡 magnitude 最大的頻率的 phase-difference 作為行走依據並加入碰撞偵測，行走規則如下：

1. 通知 base 端發出聲音，根據 phase-difference 得到聲源相對於機器人之角度 θ ，旋轉 θ 角後往前行進 l ， $l = 5 \times$ 連續未碰撞次數 $l \leq 30$ 。前行時如未碰撞，重複 step 1。碰撞發生時，機器人往後退 10 公分，進入 step 2。
2. 通知 base 端發出聲音，得到新的聲源對於機器人之角度 θ_1 ，機器人旋轉 θ_1 角，往前行進 $l = 30$ 公分，紀錄旋轉方向 α (左或右)。前行時如發生碰撞，進入 step 3，未碰撞，進入 step 4。
3. 根據 step 2 紀錄之旋轉方向 α ，機器人碰撞後退後 10 公分，往 α 方向旋轉 10 度，在往前行 $l = 30$ 公分。如發生碰撞，重複 step 3，直到碰撞未發生進入 step 4。
4. 通知 base 端發出聲音，得到聲源對於機器人之角度 θ_2 ，機器人旋轉 θ_2 角，往前行進 $l = 30$ 公分並紀錄旋轉方向 β 。如發生碰撞，機器人往後退 10 公分，往 β 之反方向轉回 θ_2 度，在往前行 $l = 30$ 公分，前行過程如有碰撞，則在往 β 之反方向旋轉 10 度在往前行，直到無碰撞發生，然後重複 step 4。如沒發生碰撞回到 step 1。

本方法利用機器人偵測聲源對於機器人之角度 θ ，使機器人朝聲源方向前進，前進過程如有遇到障礙物，機器人則以嘗試碰撞的方式，走出障礙物的阻礙。方法如上面所述。以下圖 11 表示過程：

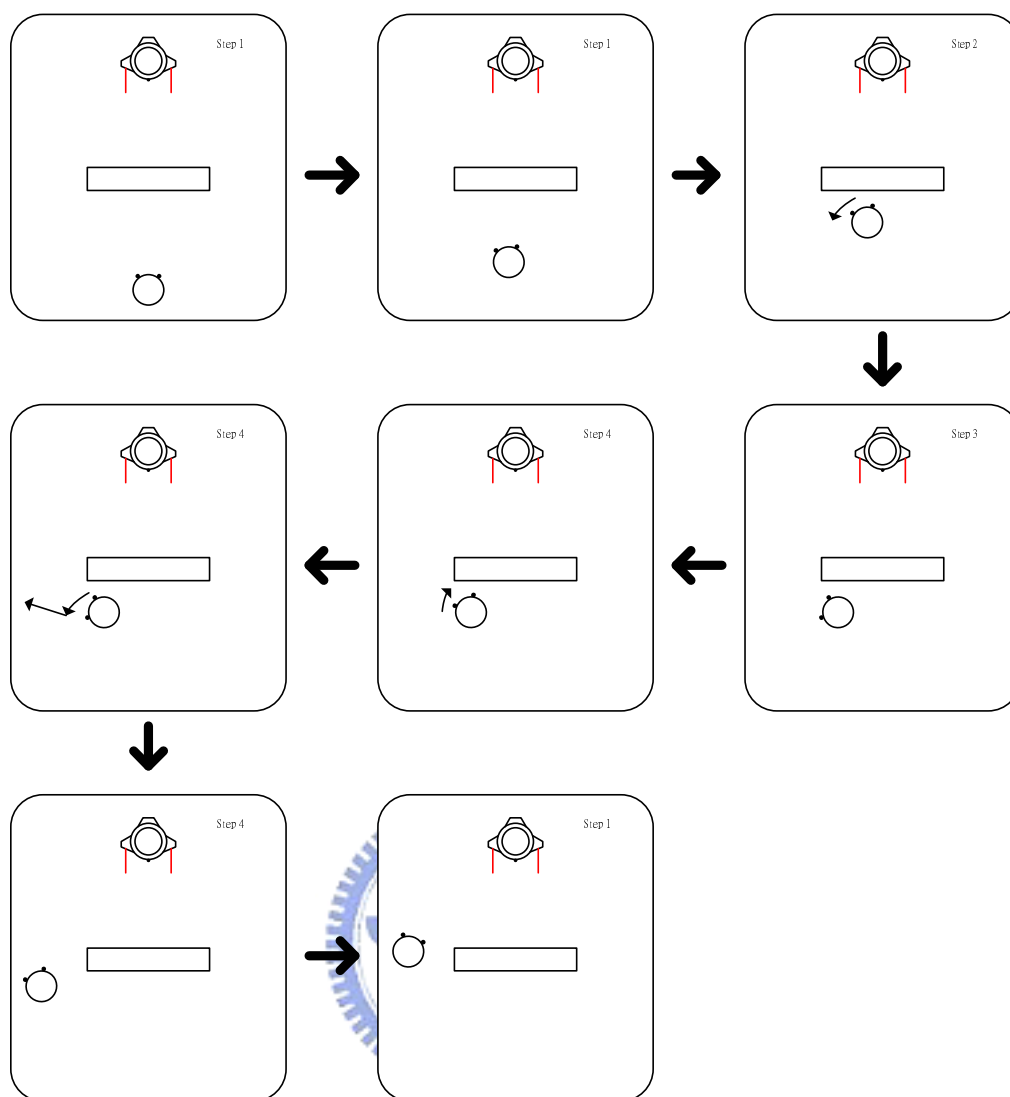


圖 11 平台行走示意圖(1)

碰撞行走流程圖：

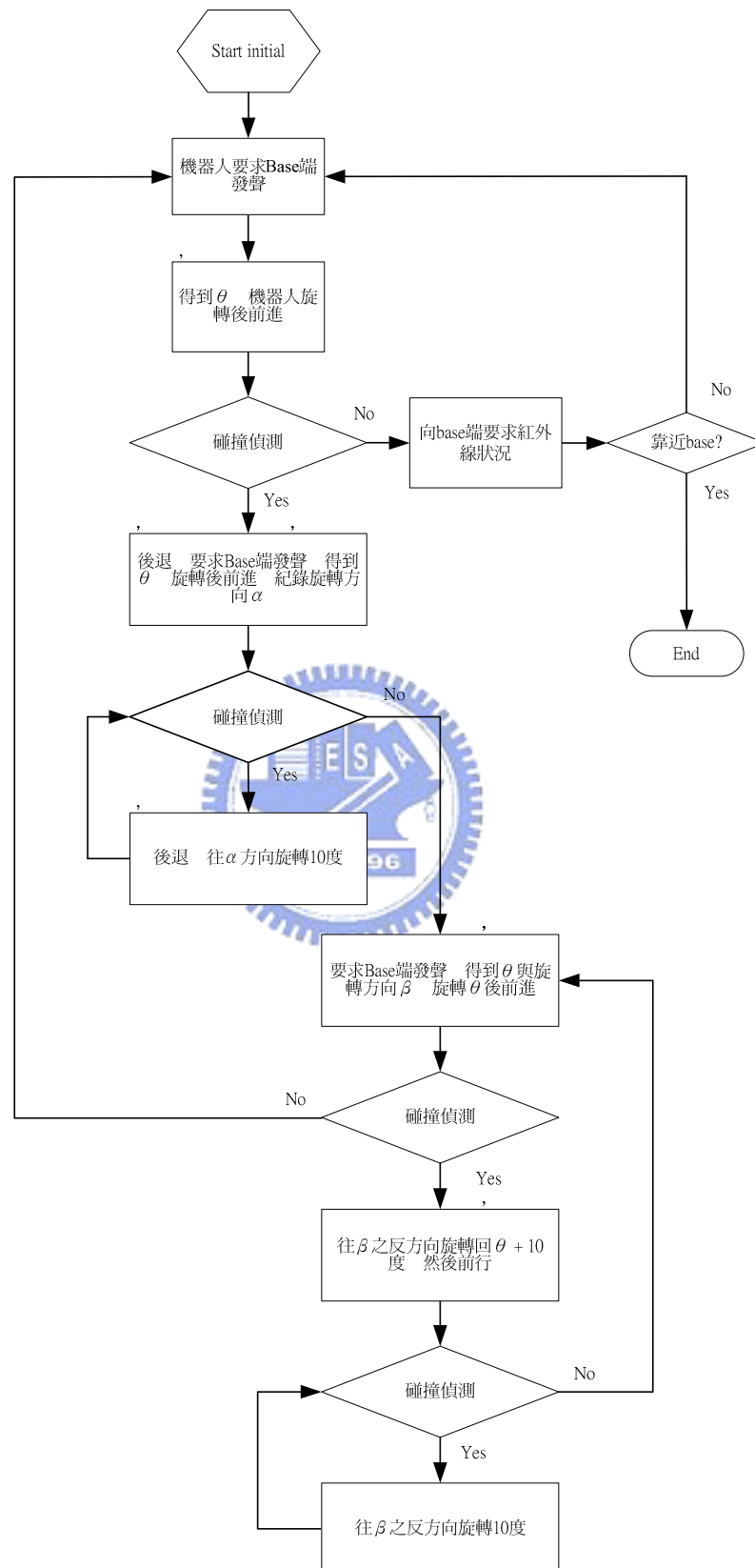


圖 12 碰撞行走流程圖

3.4.2 SVM 障礙物偵測與碰撞法

使用 SVM 判斷障礙物偵測並根據實驗 1 之方法作為行走角度之依據(有碰撞偵測)，規則如下：

1. 通知 base 端發出聲音，得到六個頻率之 phase-difference 參數，將其丟入 SVM 計算，判斷機器人對於聲源為 LOS 或是 NLOS。如為 LOS，機器人根據聲源相對於機器人之角度 θ 旋轉，並前行如過程有碰撞，則往後退，並進入 step 2，無碰撞則判斷是否到達 base 端，未到重複 step 1。如為 NLOS，進入 step 2。
2. 根據 θ 角旋轉，並紀錄旋轉方向 α ，再次通知 base 端發聲，得到 phase-difference 在由 SVM 判斷 LOS 或 NLOS。如為 LOS，則代表前次判斷可能誤判，進入 step 3，如為 NLOS，則確定確實有障礙物在前方，進入 step 4。
3. 機器人根據 LOS 判斷出之角度 θ_1 旋轉 θ_1 角，並前行。回到 step 1。
4. 根據旋轉方向 α 旋轉 10 度，在往前行 $l=30$ 公分，如有碰撞，則後退在旋轉 α 方向 10 度，在前行，紀錄次數 count，直到無碰撞發生，進入 step 5。
5. 依旋轉方向 α 之反方向旋轉回 $10 \times \text{count}$ ，回到 step 1。

為能夠改善實驗 1 需要經過碰撞之後才能知道障礙物的存在，加入 SVM 的判斷，使機器人能夠在撞到障礙物之前即能知道已經面對障礙物。當機器人判斷不在障礙物附近，即使機器人直接根據 phase-difference 所算出角度方向前進。當機器人在障礙物附近，根據實驗 1 的行走方法。行走過程如圖 13：

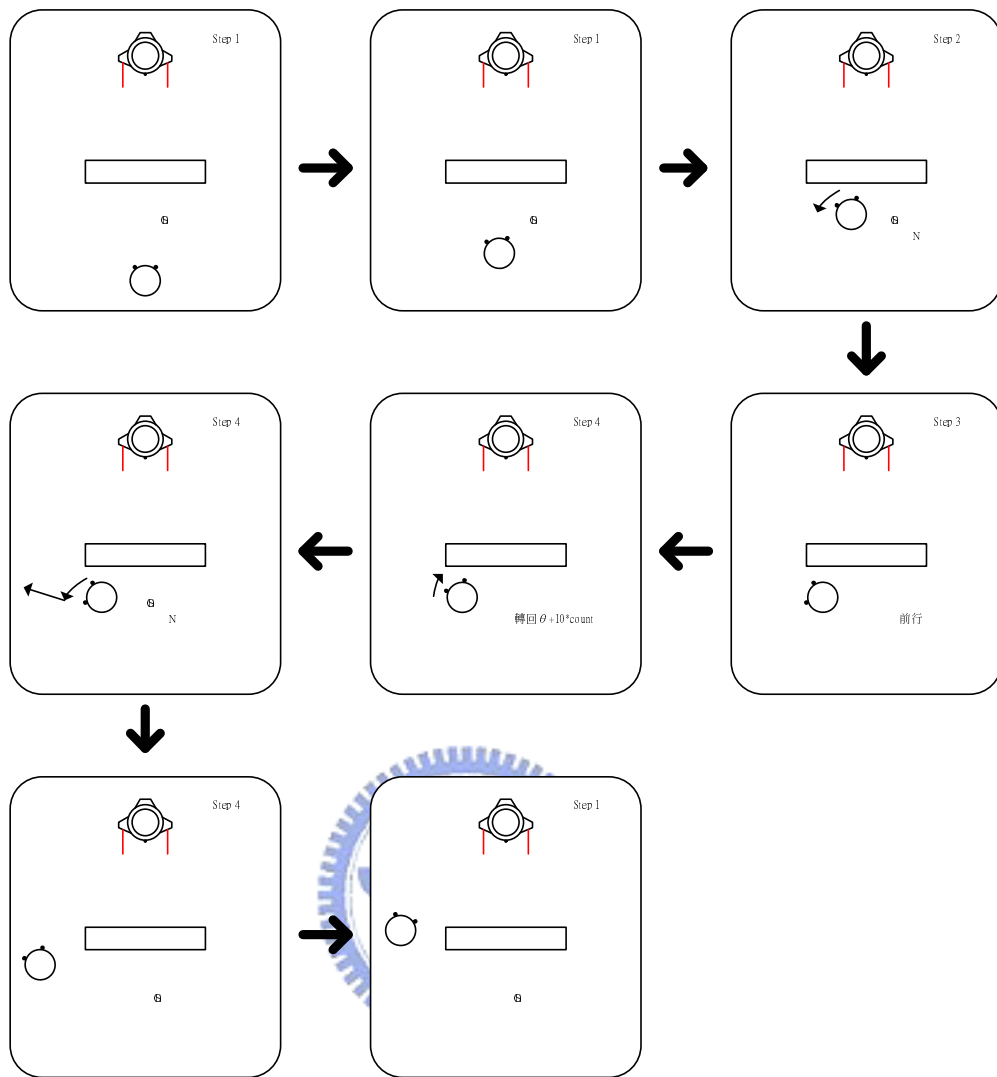


圖 13 平台行走示意圖(2)

SVM 碰撞行走流程圖：

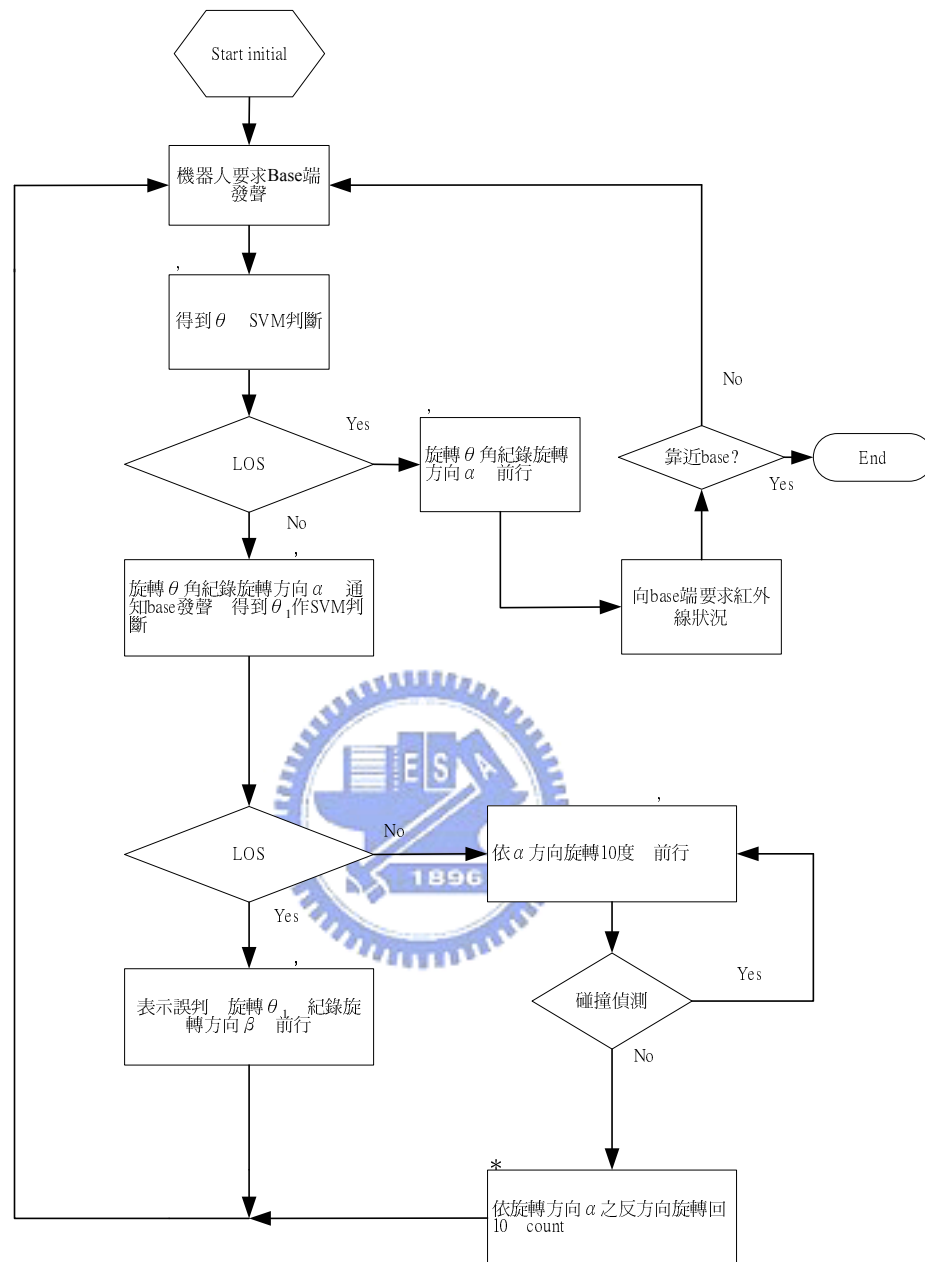


圖 14 SVM 碰撞行走流程圖

3.4.3 LOS 聲源偵測法

由於移動平台偵測聲源方位的角度範圍有限，當聲源位於移動平台後方時，便不能正確的偵測出聲源位置。因此我們多做一次聲源方位的判斷，來確認是否找到真正的聲源方位。過程如下，第一次計算出聲源方位，移動平台根據此角度

θ 旋轉，然後再要求聲源發聲一次，計算第二次聲源位置，如果此次角度在 90 ± 10 以內，那代表我們正確的找到聲源了，如果不是則代表初始時移動平台可能背對聲源或其他情況，此時就使移動平台往右旋轉 $90 \pm \theta$ ，再重新判斷一次，直至找到聲源為止。規則如下：

1. 通知聲源發聲，並計算出相對角度 $\pm \theta$ (正代表平台往右轉，負代表平台往左轉)，平台根據 θ 旋轉。
2. 再通知聲源發聲一次，計算出第二次相對角度，如果此時角度在 ± 10 度以內，代表找到聲源，就往前進，回到 step 1。
3. 如角度超越了 ± 10 度，代表第一次旋轉有問題，未能讓移動平台正對聲源，此時讓動平台往右旋轉 $90 \pm \theta$ ，然後再回到 step 1.

重複上面的動作，就能夠讓移動平台在 LOS 的行況下，往聲源移動。流程圖如圖 15 所示：

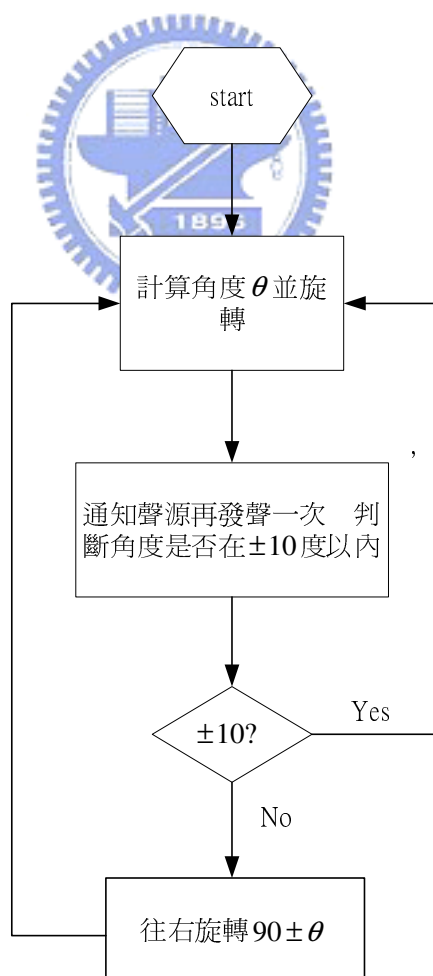


圖 15 LOS 聲源判斷流程圖

3.5 基地端(base)與輪式機器人結合動作

3.5.1 全方位運動平台分析

全方位運動平台使用特殊結構的輪子，一般稱為全向輪(omni-directional wheel or omni-wheel)；它可以將地面作用於輪子上的摩擦力之中平行於輪軸的部分，藉由滾子的轉動而分散掉；因而，造就了全向輪平台所擁有的獨特運動模式。

實際上，全向輪的構想早在 1910 年就發表於美國的專利文件中，截至目前，為了提高效率及改善震動的缺點，已有許多改良的全向輪被發明使用，而現今也已被使用在搬運車、輪椅或輪式足球機器人等方面。圖 16 中共列出三種不同型式的全向輪，(c)是目前我們正在使用的全向輪。從平行於輪軸的方向看過去，其形狀就像是一般傳統的輪子。

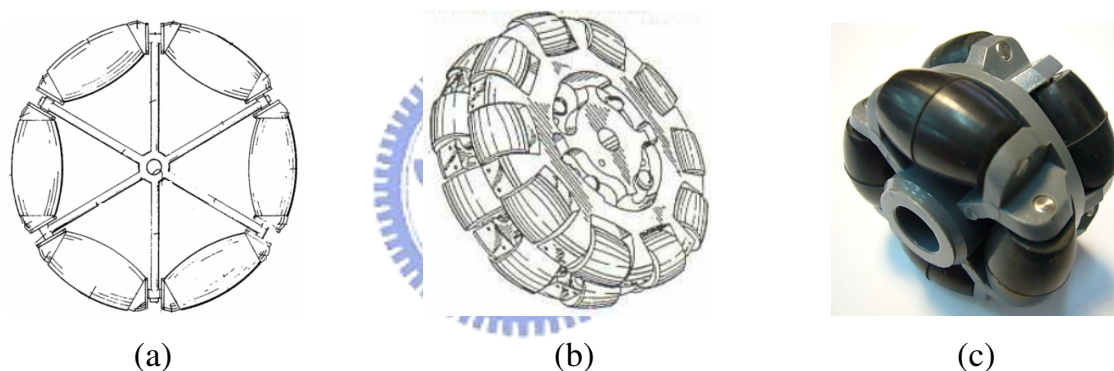


圖 16 不同型態的全相輪

我們的機器人平台共使用三顆全向輪，較使用傳統輪子的二輪或四輪式的機器人擁有較高的運動自由度；經由三顆輪子所產生的多種合力組合，共可以產生多達五種運動模式，包括：原地旋轉、對頭直行、對頭差速轉彎、平移、平移且自轉。

在此平台中使用三顆全向輪，如圖 17 所示，三輪之間各夾 120 度角，為這個移動平台最大的一個特點，這種構造使得這個平台得以呈現較一般輪式平台更多樣化的運動模式，在此將詳細探討這種構造的運動學理論[15-18]。

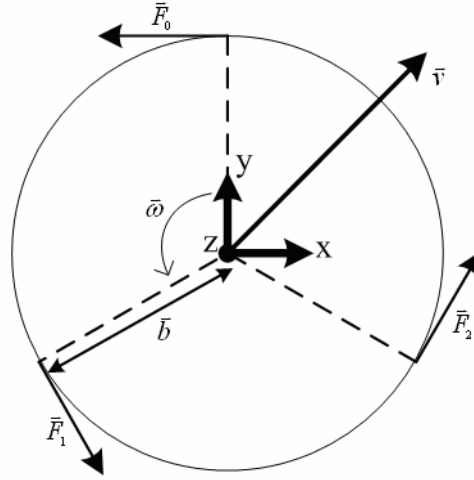


圖 17 全向輪平台簡圖

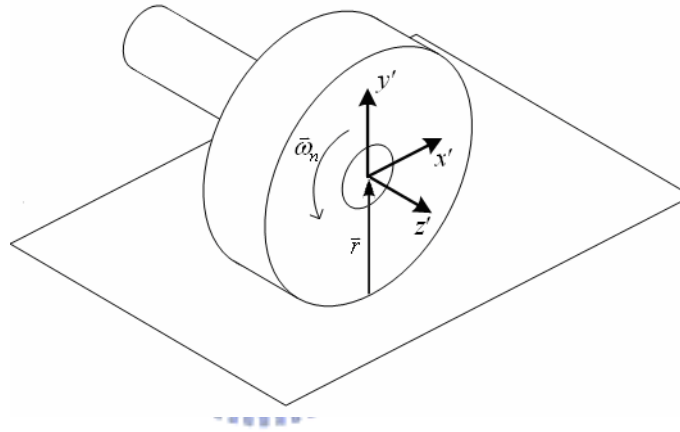


圖 18 全向輪簡圖

在圖 17 及圖 18 中，各項符號的意義表示如下：

- (a) x 、 y 、 z ：定義在平台上的座標系，下面推導中的向量皆相對於本座標系。 x 、 y 如圖所示，而 z 軸為指出紙面為正。
- (b) \bar{v} ：希望平台移動之速度向量。
- (c) $\bar{\omega}$ ：希望平台轉動之角速度向量，且定義沿著反時針方向旋轉為正。
- (d) \bar{F}_n ：垂直於各輪軸之單位向量， $n = 1, 2, 3$ 。

$$\bar{F}_0 = (-1, 0, 0)$$

$$\bar{F}_1 = \left(\frac{1}{2}, -\frac{\sqrt{3}}{2}, 0 \right) \quad (28)$$

$$\bar{F}_2 = \left(\frac{1}{2}, \frac{\sqrt{3}}{2}, 0 \right)$$

(e) \bar{b}_n ：從 n 號輪軸心端點到平台中心的距離向量。

(f) \bar{P}_n ：代表 n 號輪軸心相對於平台座標的速度向量。

令建立於各輪之上的座標系為 $x' y' z'$ ，如圖 18 所示(圖 18 中皆表其正向)：

(g) \bar{r} ：從輪子接觸地面之點到輪子軸心的距離向量。

(h) $\bar{\omega}_n$ ：各輪相對於其軸心之轉動角速度向量，令逆時針方向旋轉為正(沿正 z' 軸方向)。

(i) \bar{Y}_n ：各輪之軸心在輪沿地面滾動時的速度向量於 x' 軸方向上的分量。

在考慮 $x y z$ 座標系與 $x' y' z'$ 座標系之間的轉換，可知 x' 軸方向與 \bar{F}_n 方向永遠一致，因此在 $x y z$ 座標系中看 x' 軸上的向量，為：

$$(x' \text{ 軸上正向大小}) \cdot \bar{F}_n$$

考慮此平台一個最廣泛的運動模式：平台本體具有一個自旋轉的角速度 $\bar{\omega}$ ，並且以速度向量 \bar{v} 移動；如何求得每一伺服機所需運作的角速度 $\bar{\omega}_n$ ，其運動學推導如下：

在 $x y z$ 座標上，各輪子之軸心端點的速度等於平台中心點的速度加上平台自旋所產生的角速度：

$$\bar{P}_n = \bar{v} + (\bar{b} \times \bar{\omega}) \quad (29)$$

(29)式也可以寫成

$$\bar{P}_n = \bar{v} + (b \cdot \omega) \cdot \bar{F}_n \quad (30)$$

而在 $x' y' z'$ 座標上，各輪子之軸心端點速度為：

$$\bar{Y}_n = \bar{r} \times \bar{\omega}_n \quad (31)$$

(31)式也可以寫成

$$Y_n = r \cdot \omega_n \quad (32)$$

在 $x' y' z'$ 座標中的 x' 軸正向即為 xyz 座標中的 \bar{F}_n 單位向量的方向，因此我

們可知，同是表示各輪軸心端點速度的 \vec{Y}_n 跟 \vec{P}_n ，其關係如下：

$$Y_n = \vec{P}_n \cdot \vec{F}_n \quad (33)$$

綜合(32)、(33)式，可得

$$r \cdot \omega_n = \vec{P}_n \cdot \vec{F}_n \quad (34)$$

將(30)式代入(34)式可得：

$$r \cdot \omega_n = [\vec{v} + (b \cdot \omega) \cdot \vec{F}_n] \cdot \vec{F}_n \quad (35)$$

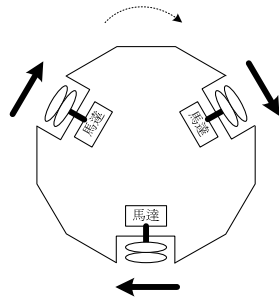
$$\omega_n = [\vec{v} + (b \cdot \omega) \cdot \vec{F}_n] \cdot \vec{F}_n / r \quad (36)$$

由此可知，如果要讓此平台以角速度 ω （逆時針為正）作自轉、加上以速度向量 \vec{v} 作平移，將 ω 以及 \vec{v} 代入(36)式，即可求出各輪所需要轉動的角速度 ω_n ，亦即各伺服機所需要提供之角速度。

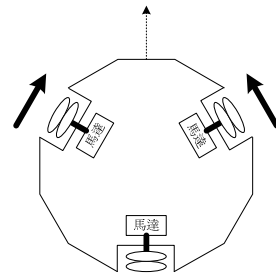
經過前一小節的分析，可以知道此一移動平台有能力呈現較一般兩輪平台更為複雜、多樣化的運動模式，共有以下五種：

- (a). 原地旋轉：三個輪子的角速度保持一定且相等、轉向相等，則平台可作原地等角速度旋轉。
- (b). 對頭直行：當三輪中其中一輪不運轉，其他兩輪以相同的角速度旋轉但轉向相反時，則本平台會朝不運轉的輪子方向(一個「頭」)前進。
- (c). 對頭差速轉彎：以對頭直行模式為基礎，轉動的兩輪若轉速不同，則本平台會進行轉彎的動作，此時「頭」的指向會改變。這種運動模式與一般兩輪平台差速轉彎的道理類似。
- (d). 平移：所謂平移即是本體指向不改變的移動，而且移動方向可以任意選擇。這種運動方式正是普通輪式平台無法呈現的運動模式。
- (e). 平移且自轉：這是本平台所能呈現最複雜的一種運動模式，一邊朝著固定方向作平移，一邊又以固定角速度作自轉，這也是普通輪式平台無法呈現的運動模式。

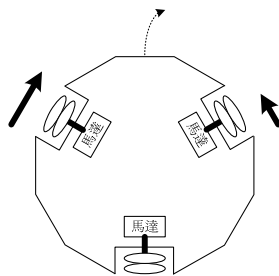
圖 19 為上述幾種運動模式的示意圖。(a)為「原地旋轉」，(b)為「對頭直行」，(c)為「對頭差速轉彎」，(d)為「平移」，(e)為「平移且自轉」。



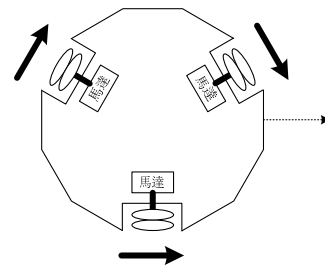
(a) Rotation



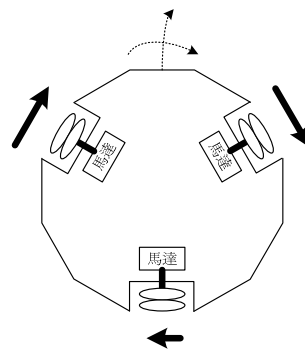
(b) Go ahead



(c) Differential Turning



(d) Translation



(e) Translation and rotation

圖 19 運動模式示意圖

3.5.2 結合運動設計

在平台根據聲音判斷順利的靠近 base 端，即進行平台與 base 的結合。為了能夠讓平台在結合後能夠載動 base 端行走，因此我們設計平台使用上下結合的方式，並分別在兩邊的結合面上，佈上一圈的磁鐵，讓他們在結合後可以更為穩固。由上一節所提到的全向輪運動方式，將會使用在結合過程中。結合的過程主要分為兩個部分，第一部分為移動平台進入 base 前與 base 端的對準，第二部分為移動平台在進入 base 後，兩方上下位置的對準，在進行提升結合動作結合。

為了偵測移動平台是否到達 base 端附近，我們於 base 門口的左右邊各放置一個紅外線感測器，當平台從 base 端的左邊靠近被偵測到時，base 將通知平台往右邊行走一個弧線，直到平台走到 base 端的中間為止。此時可以確定平台在 base 前面，但是不能確定平台是否正對著門口。於是我們在 base 端正中間加上一個光敏感測器，平台正前方放置發光二極體，做面對門口的校正。當平台畫完弧線以後，在往左旋轉 40° ，然後開始往右邊旋轉掃描光敏感測器是否有變化，如偵測到變化，即停止旋轉。

接下來進行結合的部分。為了讓平台與 base 能夠大概的上下對齊，我們於 base 中央部份佈了一圈光敏感測器(8 個)，移動平台正中間放置發光二極體，利用光的特性來做對正。如圖 20 所示：

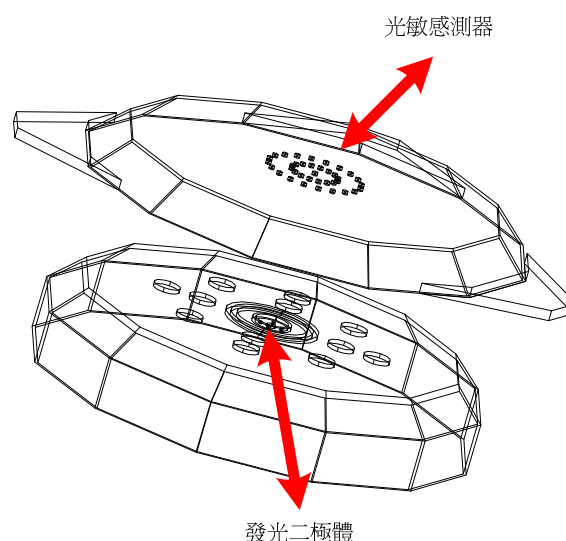
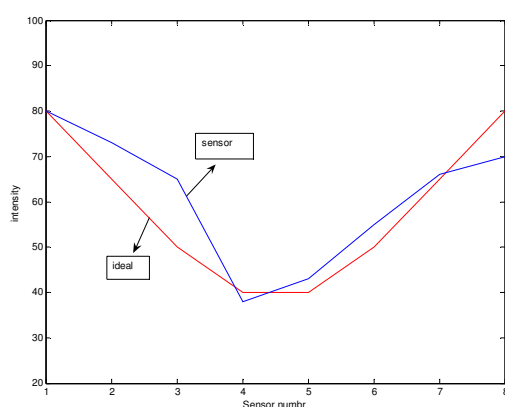
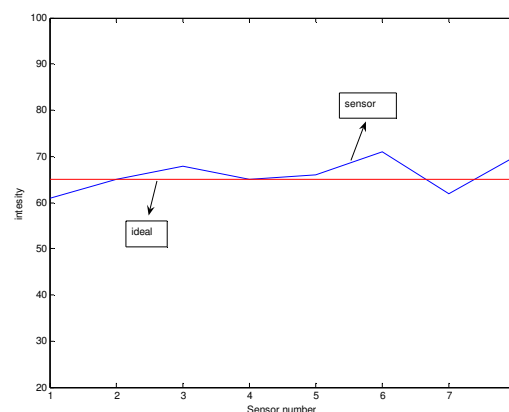


圖 20 光敏感測器示意圖

當平台進入 base 端，位於 base 端的光敏感測器會有如圖 21(a)之變化，此時平台便利用全向輪特有的移動方式，平移來做微調的動作，使光敏感測器得到較為平均的受光量，如圖 21(b)。如此可以確保機器人上下兩端中心位置已經大約對準，接著移動平台便將自己的外殼升起，與 base 端結合。



(a)



(b)

圖 21 光敏感測器感測圖



第四章 系統軟硬體設計與實現

4.1 整體系統架構

系統大致分為三個部分，由一台 PC、全方位移動平台、base，所組成，如圖 22 所示：

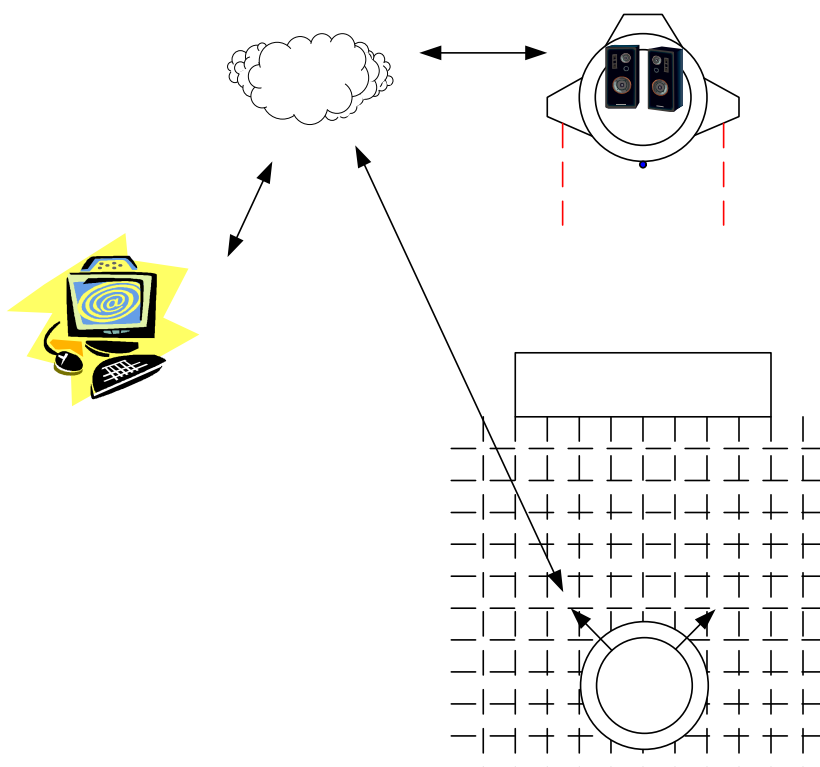


圖 22 整體系統示意圖

PC 端主要功能為撥放複頻聲音訊號，移動平台上有由 TI 所發行的嵌入式系統實驗板 OMAP 5912 OSK，主要演算皆在此實現，於 base 端上設置有 Access Point，用以連接三方的通訊，互相傳遞控制訊號與感測器資訊。詳細系統架構圖如圖 23 所示：

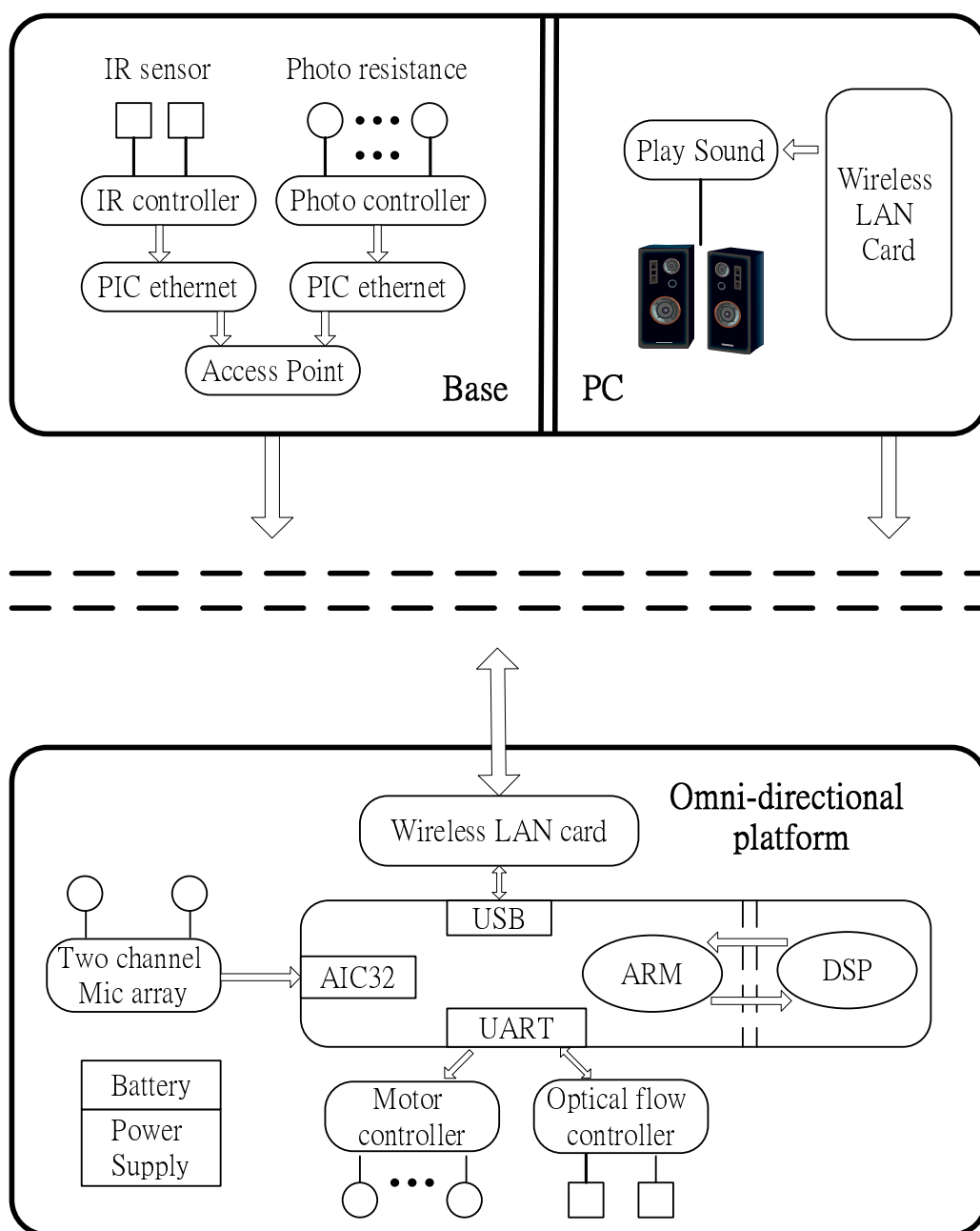


圖 23 系統架構圖

4.2 實驗平台

整體系統主要以 TI 所發行的 OMAP 5912 OSK 為實驗核心平台。OMAP 5912 為 Dual-Core 處理器，包含了一個 GPP(General Purpose Processor): ARM926EJ-S，與一個 DSP(Digital Signal Processor): TMS320C55x。所謂 DSP 與一般的 Processor 架構上有所不同。訊號處理上，常需要有大量的乘法運算或者重複的運算過程一直發生，使用 GPP 做運算會浪費許多時間，因此造成了 DSP 的產生。DSP 使用

硬體設計實現單週期乘法運算，相對於 GPP 的多週期乘法快上許多。因此有了 Dual-Core Processor 的整合。圖 24 為 OMAP5912 之硬體架構。

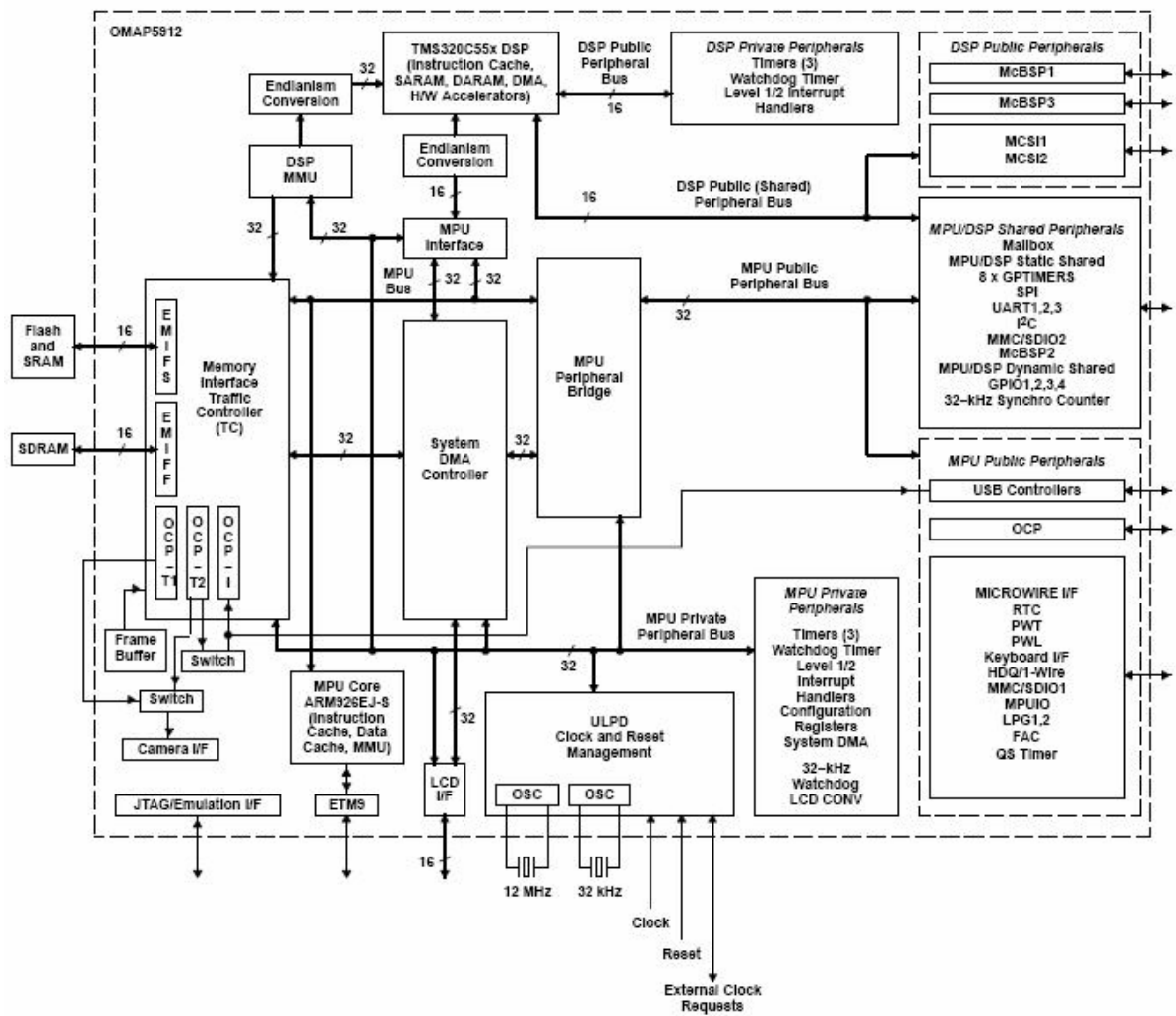


圖 24 OMAP5912 架構圖

OMAP5912 對 ARM926 與 TMS320C55x 做了高度的整合，使兩個處理器能夠做彈性的分工，擁有高處理效能與低耗電的特性。

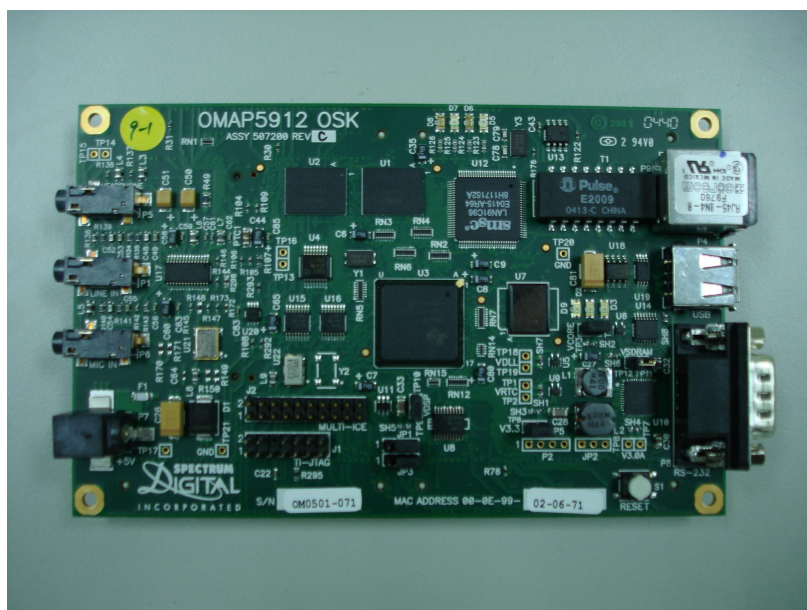


圖 25 OMAP 5912 OSK 外觀

表 2 OMAP 5912 OSK Hardware list

- ARM9 core operating at 192 Mhz
- DSP core operating at 192 Mhz
- 32 Mbyte DDR SDRAM
- 32 Mbyte Flash ROM
- RS-232 Serial Port
- 10 MBPS Ethernet port
- USB Host Port

4.2.1 軟體開發環境

為了使程式能夠順利於 OMAP 端執行，需要以下幾個工具，為其建立開發環境：

1. PC 端 Linux 作業系統
2. Cross-compiler[19]
3. DSP-Gateway[20]
4. Code Composer Studio(CCS)

◆ Linux 作業系統

一般來說嵌入式系統受限於 processor 速度與記憶體大小，程式開發者不可能直接於嵌入式系統板上直接開發程式，因此我們需要一個平台作為 OMAP ARM

端程式的開發環境。程式開發者有兩種選擇為自己準備開發環境，若手邊有多餘的 PC，則直接於 PC 上架設一 Linux 系統即可。若無多餘的電腦，使用 VMware 虛擬機器軟體是另一個選擇。VMware 可以模擬 x86 架構的電腦系統，除了執行 x86 指令外，還可以模擬其他週邊設備，如：網路卡、序列埠、串列埠，等等。也就是說使用 VMware 就可以在作業系統中，模擬另一個 PC，如此就可以在同一台電腦上擁有多個不同的作業系統。

在此我們直接架設一台 PC 作為開發 OMAP ARM 端程式所用，使用 Fedora Core 5.0 Linux 作為作業系統。

◆ Cross-compiler

如上所提，嵌入式平台不會有多餘的記憶體來直接編譯程式，因此需要由其他機器產生嵌入式系統端的程式執行檔，在放進嵌入式系統上執行。在這裡我們的發展平台為 x86 架構的 PC，與目的端 OMAP 5912 的 ARM 處理器使用不同的指令集，並不能直接使用發展平台裡的 GCC 作為 compiler。因此需要一套可以編譯出 ARM 端指令集的 Cross-compiler 來為我們做編譯程式的動作。

我們使用 arm-linux-gcc-3.3.2 工具組作為我們的 cross-compiler。

◆ DSP-Gateway

如同一開始所提，OMAP 5912 為一 Dual-Core processor，包含了 ARM926 處理器與 TMS32055x 數位訊號處理器。而 DSP-Gateway 則是讓在 ARM 端執行的 Linux 能與 DSP 溝通的一套軟體。

ARM 與 DSP 的溝通主要透過下列三種形式：

1. Mailbox

ARM 與 DSP 可以透過 Mailbox 的中斷來做資料的交換，當 ARM 把資料寫進 DSP 時，DSP 會收到一個 Mailbox 中斷，DSP 借由中斷即可拿取 ARM 寫進來的資料。因為透過這個方法資料流量不能太大，不適合做大量的資料傳輸，通常拿來作為通訊協定、傳送命令、參數等。

2. MPU Interface(MPUI)

MPUI 可以讓 ARM 存取整個 DSP 的記憶體空間與 DSP 的周邊匯流排，因此 ARM 可以擁有 DSP 輸出輸入空間的完整存取權限。ARM 便是利用 MPUI 來將 DSP 端程式載入到 DSP 的程式記憶體中。

3. Share Memory

透過 OMAP 上的 Traffic Controller，ARM 與 DSP 可以存取相同的記憶體空間位址(如：SRAM、SD-ram 或其他記憶體媒介)。ARM 可以控制 DSP 來存取這些記憶體媒介。適合 ARM 與 DSP 端有大量資料流通使用。

DSP-Gateway 就是架構在這三種溝通方式上完成的一套軟體，包含兩個部分，ARM 上所執行的 Linux 裝置驅動程式與 DSP 上的函式庫。Linux 裝置驅動程式提供程式開發者可以於 ARM 端的程式裡透過 read()、write() 等系統呼叫來使用 DSP，而 DSP 函式庫提供函式讓 DSP task 可以被 ARM 透過驅動程式使用。於 Linux 端產生的裝置，有下面五個：

- DSP task devices

這個裝置為 Linux 提供一個管道來與 DSP 端的 task 溝通，程式設計者透過 read()、write()，等呼叫存取這個裝置，就代表著接收與傳送資料到 DSP 端的 task。

透過 dspctl utility，device file 將自動的產生於/dev 目錄下，以及一個 symbolic link 於/dev/dsptask 目錄下。

- DSP control device

此裝置提供 Linux 對 DSP 端的控制 API，透過呼叫 ioctl()，Linux 端程式可以對 DSP 做 reset、run、read DSP configuration，等控制。

Device file 將產生於/dev/dspctl/ctl。

- DSP memory device

此裝置提供 Linux 的 DSP program loader 能夠存取 DSP 端的記憶體空間，以便能夠將 DSP 程式載入 DSP 的程式記憶體(SARAM and DARAM)裡。除此之外透過 ioctl()，Linux 可以將外部 SDRAM 記憶體空

間映射進 DSP 端的記憶體空間，經過映射後，DSP 端即可以存取外部記憶體。

Device file 將產生於/dev/dspctl/mem。

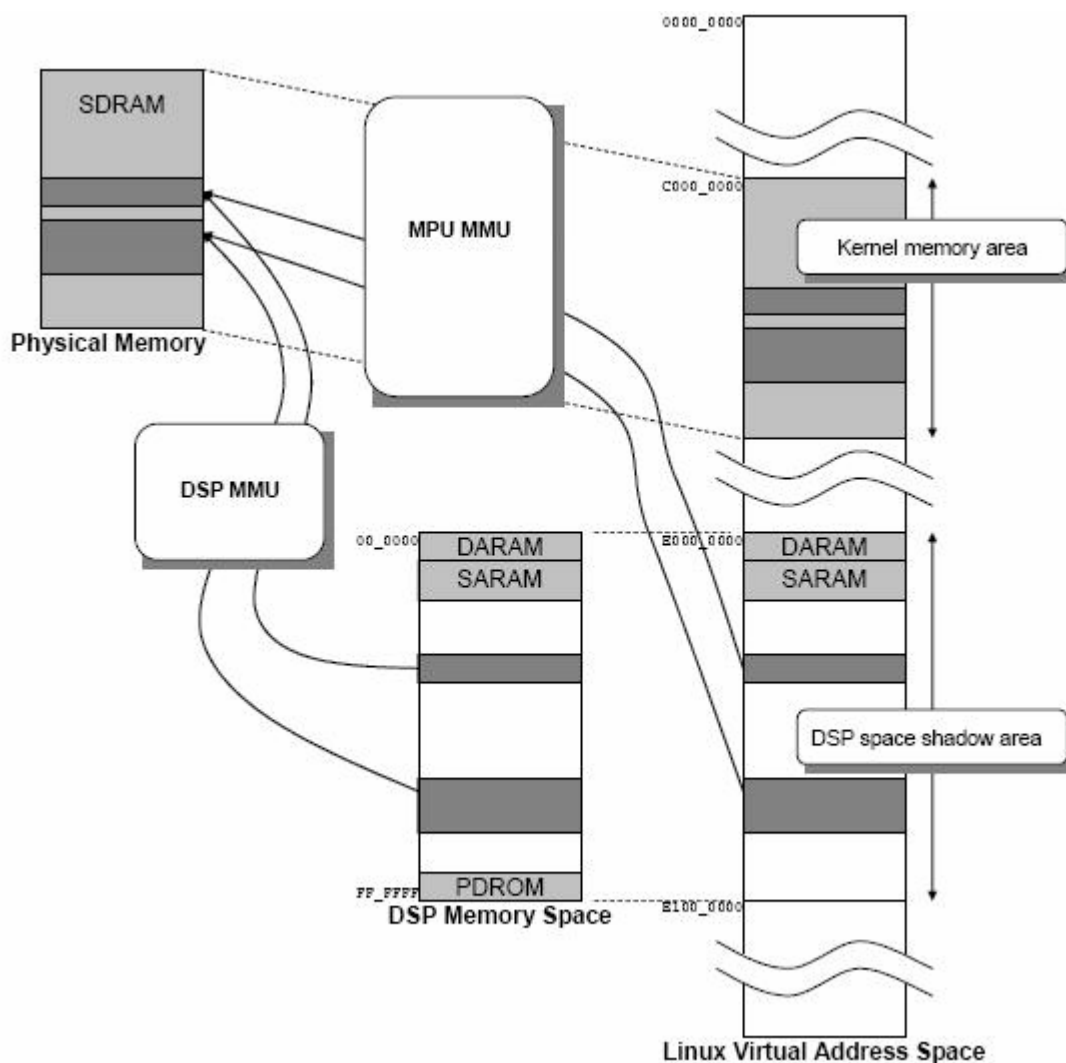


圖 26 SDRAM 記憶體映射

- DSP task watch device

此裝置提供 Linux 能夠知道 DSP 端 task 的情況，哪些 task 沒在使用，哪些正在使用，哪些正在載入中等資訊。

Device file 將產生於/dev/dspctl/twch。

- DSP error detection device

此裝置用來偵測 DSP watchdog timer 是否逾時，他只提供 read()呼

叫，一般情況使用 read()時是不會返回，直到 DSP 有錯誤發生如：DSP MMU fault、WDT expiration，read()呼叫才會返回。

Device file 將產生於/dev/dspctl/err。

◆ Code Composer Studio(CCS)

CCS 是一套整合式開發環境的發展軟體，提供 C 語言的編譯器與組合語言的組譯器。於一般情況下可以直接程式下載入 DSP 並執行，並且可以做簡單的除錯工作，但是在 OMAP 上搭配 DSP Gateway 使用，CCS 並不能直接將程式載入到 DSP 中，因此 CCS 用來編譯出 DSP 端的程式執行檔，在透過 DSP Gateway 將執行檔 load 進 DSP 中執行。

4.2.2 程式開發流程

程式分為兩個部分，一為 ARM 端程式，另一部分為 DSP 端程式。ARM 端程式負責控制移動平台、光流感測器的資訊與 SVM 計算，並且將接收到聲音資料傳給 DSP 端。DSP 端程式負責將接收到聲音資料做相位差與大小比的演算。兩支程式的編譯在不同的平台上，透過 Linux 系統的 cross-compiler 來做 ARM 端程式的編譯，以及使用 windows 作業系統上的 CCS 來做 DSP 程式的開發與編譯。

程式編譯完成後，利用 TFTP 將兩支程式透過網路下載進 OMAP 5912 OSK 的 Embedded Linux 系統上，分別於 ARM 端與 DSP 端執行。

4.3 Base 與輪式機器人平台之架構

此節將接紹硬體與電路架構，圖 26 為機構之 CAD 圖。

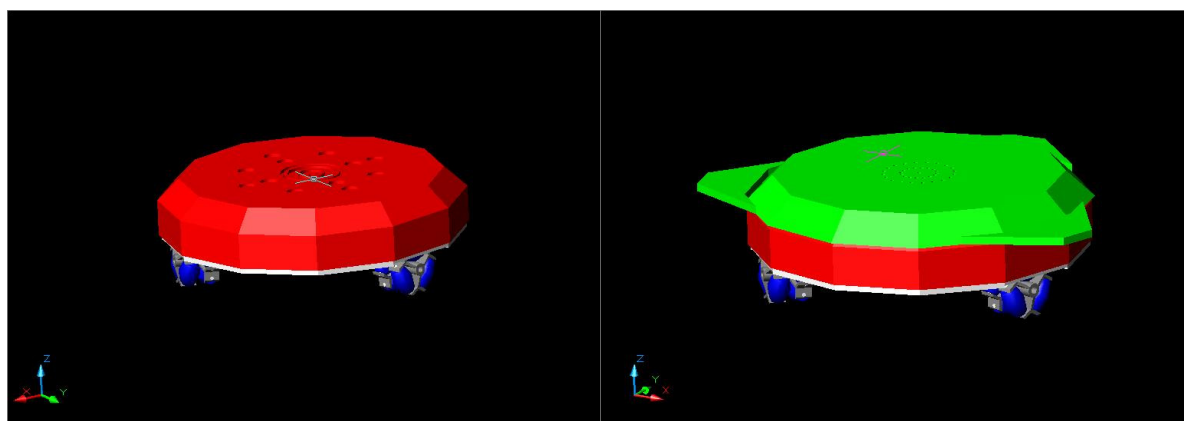


圖 27 Base 與移動平台機構圖(a) (b)

4.3.1 Base 與移動平台架構

◆ Base

基地端為正 12 邊形向內凹的蓋子，上面裝置了一台無線網路基地台(Access Point 含 Switch hub 功能)、2 組 PIC Ethernet 系統板、2 組感測器控制板、兩個紅外線感測器、一圈(8 個)光敏感測器、以及 JS-J707 喇叭一組。如圖 28 所示。

Access Point 建構整個系統的無線網路通訊路徑，而 base 上的紅外線感測器與光敏感測器資訊經過控制板擷取後，透過 PIC Ethernet 與 Access Point 傳送回給 OMAP 端。

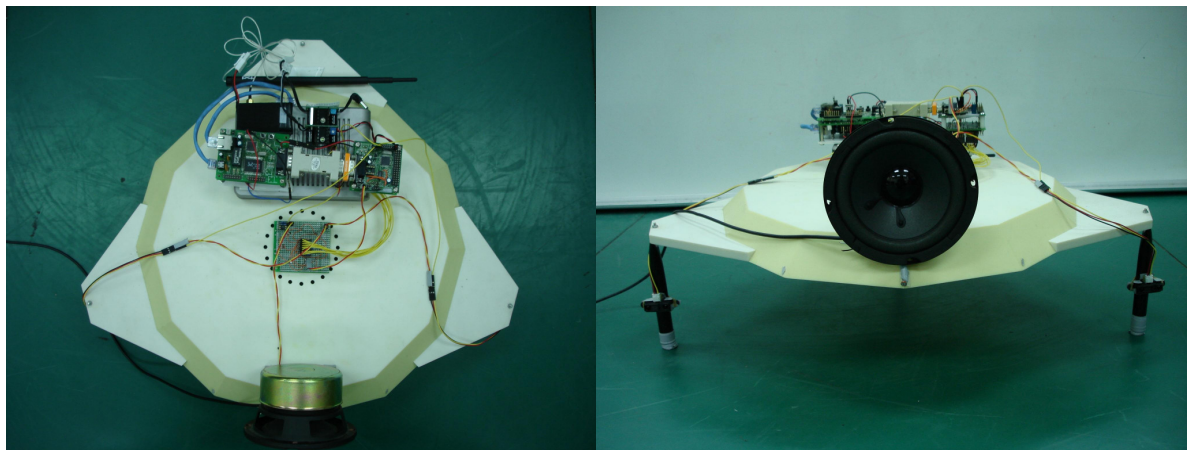


圖 28 Base 端照片

◆ 全方位移動式平台

平台為一正 12 邊形之輪式機器人，由上蓋與底盤兩個部分組成，且上蓋可以經由控制馬達自由的升降。移動平台上裝置 1 組 OMAP 5912 OSK 嵌入式系統發展板與 USB 無線網路卡、雙聲道麥克風陣列、2 組光流感測器、4 組 DC 直流馬達、1 組馬達控制器與 1 組感測器控制板，如圖 29 所示。

系統控制中心為 OMAP 嵌入式系統板，透過 UART 可以傳送馬達控制指令給馬達控制器，控制四科馬達的旋轉速度與方向，以及向光流感測器控制板拿取光流目前的資料，計算移動平台目前的移動速度與方向。透過 USB 無線網路卡，可以與 Base 互相通訊，讀取 Base 端各個感測器的資訊。為了能夠與 Base 端做上

下的結合，經過設計使上蓋能夠透過中心馬達旋轉方式，慢慢向上提升。且於平台正中間上方與平台前方有數個發光二極體，作為結合時所需要的依據。

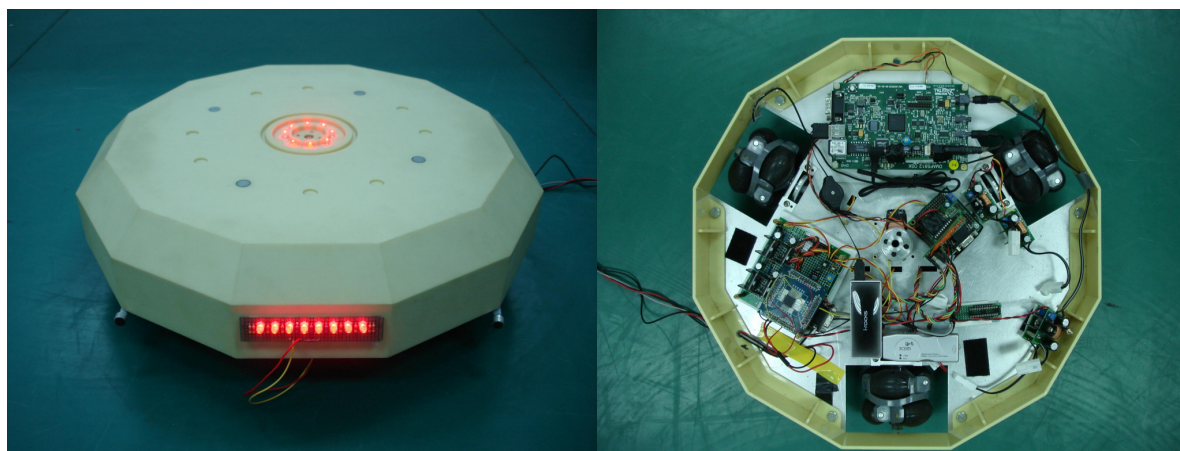


圖 29 全方位移動式平台照片

4.3.2 電路介紹

下面將詳細介紹系統中各個區塊的電路功能，可參考圖 23 的詳細系統架構圖。

◆ PIC Ethernet 系統板

如圖 30 所示，此為 Microchip 所發行之嵌入式網路版，他可以把網路訊息封包解開之後，將資料內容從 UART 送出，也能夠從 UART 接收資料，將資料做網路封包的包裝，在透過網路傳送出去。



圖 30 PIC Ethernet 嵌入式網路板

◆ 感測器控制板

如圖 31 所示，此為本實驗室自行製作之單晶片微控制器實習板，有基本的單晶片功能，包含 8 個 S/HAD 轉換器、1 組 UART 與 16 個 I/O。透過此感測器控制

板，我們可以讀取各種感測器的 AD 值，與解讀光流感測器之訊號，在將資料透過 RS-232 送出。

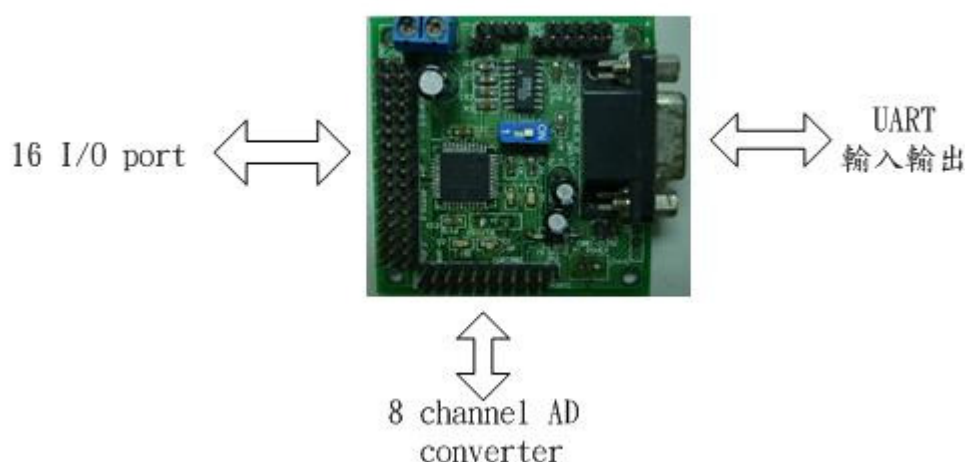


圖 31 感測器控制板

◆ 馬達控制板

如圖 32 所示，此為移動平台之馬達控制器，上面包含 4 組馬達驅動電路與 1 個單晶片微控制器。單晶片透過 UART 接收馬達控制指令，在對 4 組馬達給予相對應的 PWM 控制訊號，控制馬達的轉向與速度。

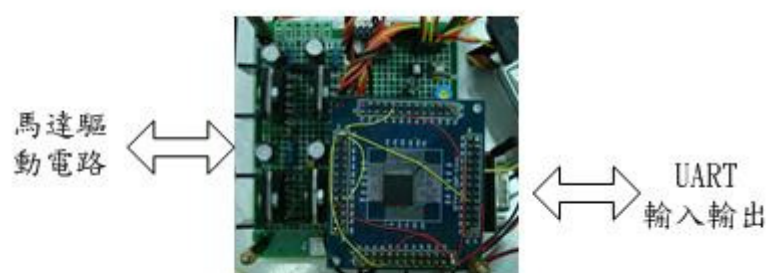


圖 32 馬達控制板

◆ 電源模組與電池

如圖 33 所示，使用 Step-down 穩壓模組，相較於一般 78 系列的穩壓 IC，能夠提供更有效率之能量轉換與電流。電池組選用鋰聚合電池，比一般鎳氫、鎳鎘電池更為持久與壽命較長。

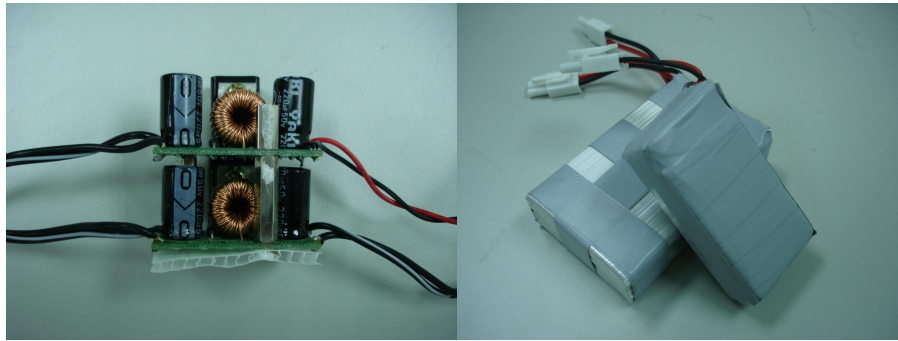


圖 33 電源模組與電池

◆ 光流感測器

如圖 34 所示，我們使用雷射滑鼠作為光流感測器，此感測器相較與普通滑鼠有幾項優點，解析度較高、可在透明物質上使用、且不需與地面接觸。

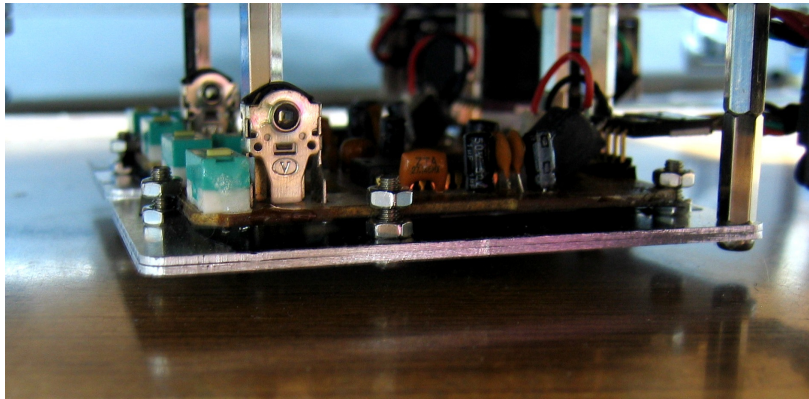


圖 34 光流感測器

◆ 馬達組

如圖 35 所示，我們將雷虎科技所出的 DS 1313 伺服機改裝後，在自行加上直流馬達驅動電路，完成我們需要的馬達組。

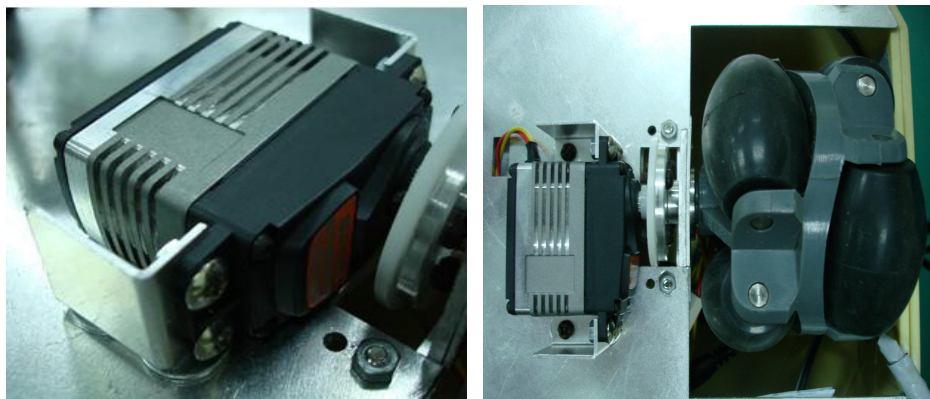


圖 35 馬達組

◆ Access Point

如圖 36 所示，Access Point 為 ASUS 所出的 WL-520g，最大傳輸率為 54Mbps，且符合 IEEE 802.11g 的規格。



圖 36 Access Point



第五章 實驗結果與展望

5.1 實驗結果

5.1.1 聲源方位判斷

本實驗為聲源相對於移動式平台方位的判斷，實驗環境中有一固定位置的喇叭放置於基地端，由移動平台要求喇叭發出一個 0.25 秒由六個頻率(343.75、375、406.25、437.5、468.75、500Hz)所組成的聲波，如圖 37 所示，經由移動式平台上的雙聲道麥克風接收後計算其相位差與大小比，取其中大小比最大之頻率的相位差作為方位判斷的依據。



圖 37 複頻聲音訊號

◆ 無障礙物之聲源方位判斷(LOS)

如圖 38 所示，於各點上移動平台的麥克風裝置皆面向正 y 方向，且距離基地端 2 公尺，而角度以海星號座標系統為準，由左而右從 0 度遞增為 180 度。

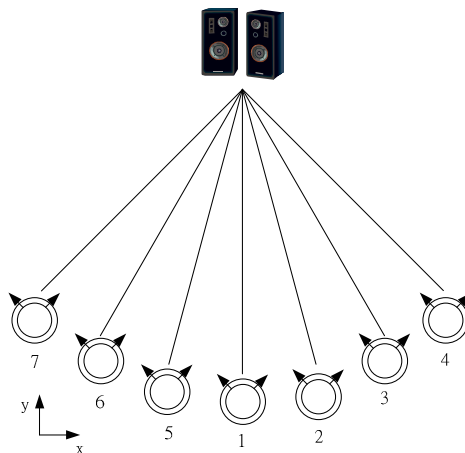


圖 38 無障礙物聲源判斷

我們分別於每一個位置計算五次相對方位的角度，在取其平均值，得到實驗結果如表 3。

表 3 聲源方位判斷比較表(1)

平台位置	相對真實角度	平均角度	誤差
位置 7	135	120.7196	10.578%
位置 6	120	101.0345	15.804%
位置 5	105	102.874	2.025%
位置 1	90	90.6911	0.768%
位置 2	75	78.0892	4.119%
位置 3	60	72.3738	20.623%
位置 4	45	63.4263	40.947%

觀察實驗結果不難發覺，當聲源與移動平台之間的相對夾角越大時，所判斷出來的角度會有越大的誤差，許多因素都會影響結果，如，喇叭並不是一完美的點聲源、聲速於不同的溫度會有所改變，與我們實驗計算所代入的 340m/sec 會有些許差異、受限於機構的影響，兩個麥克風接收同一方向來的聲波，其角度不能太大等問題。整體來說當聲源在移動平台正前方正負 30 度裡，結果都可以接受。

◆ 有障礙物之聲源方位判斷(NLOS)

如圖 39 所示，我們於環境中加入一個障礙物，使聲源與移動平台間為 NLOS。

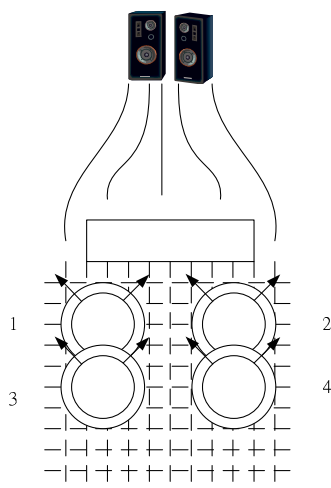


圖 39 有障礙物聲源判斷

一樣我們分別於四個點各取五次計算值，再做其平均，得到實驗結果如表 4 所示。

表 4 聲源方位判斷比較表(2)

平台位置	平均角度	與移動平台夾角
位置 1	41.3232	48.6768
位置 2	129.2134	39.2134
位置 3	70.8285	19.1715
位置 4	103.2532	13.2532

根據聲音的特性，可以合理的發現，角度在移動平台越靠近障礙物時，會變大，而在離障礙物較遠時，會變小。且平台位於障礙物右邊時，將會判斷聲源從右邊而來，位於障礙物左邊時，聲源從左邊而來(以移動平台座標為準)，例用這個特性作為當移動平台遭遇障礙物時，該往左邊旋轉或是往右邊旋轉且應該旋轉幾度的依據。



5.1.2 SVM 障礙物判斷

如 3.3.2 小節所述，聲波在經過障礙物時各個頻率的相位差將會被障礙物影響，透過這個性質我們就可以使用 SVM 做無障礙物與有障礙物的分類。下面各表列出 SVM 訓練資料部分的 Time-Delay，分為無障礙情況，移動平台正對聲源、面向左邊與面向右邊，與有障礙物情況，移動平台正對聲源、面向左邊與面向右邊等情形。

表 5 SVM 部分訓練資料(1)

無障礙物，正面對聲源					
343.75 Hz	375 Hz	406.25 Hz	437.5 Hz	468.75 Hz	500 Hz
0.9874	1.9056	-0.6107	-2.6210	-2.3923	-1.0558
2.4496	4.6186	1.4559	2.0173	2.9202	-1.3500
-1.4080	3.7006	-0.2018	0.3360	-1.3007	0.0698
-2.0330	-1.6572	-2.3299	-0.9830	0.0172	-0.4639

表 6 SVM 部分訓練資料(2)

無障礙物，相對於聲源面向右					
343.75 Hz	375 Hz	406.25 Hz	437.5 Hz	468.75 Hz	500 Hz
14.4548	5.8251	11.2482	8.7178	7.7048	6.5466
13.8857	7.6232	7.8759	5.9847	5.5569	6.8381
11.6379	7.7656	8.9118	5.7475	5.1322	4.7858
13.8554	6.7265	5.0417	4.9285	6.1641	6.5525

表 7 SVM 部分訓練資料(3)

無障礙物，相對於聲源面向左					
343.75 Hz	375 Hz	406.25 Hz	437.5 Hz	468.75 Hz	500 Hz
-7.5415	-8.6338	-4.9863	-7.3628	-6.6807	-7.9686
-9.5273	-6.6877	-7.4528	-7.7632	-8.3501	-7.9273
-6.2943	-9.1943	-10.4075	-10.4825	-7.5241	-6.3070
-17.7045	-7.5876	-7.0942	-10.9675	-7.0430	-7.7731

表 8 SVM 部分訓練資料(4)

有障礙物，正面對聲源					
343.75 Hz	375 Hz	406.25 Hz	437.5 Hz	468.75 Hz	500 Hz
-8.2780	-1.5710	5.5553	0.8414	8.3310	2.9949
-7.2128	-7.7908	12.1135	-7.3668	10.4116	4.4061
-6.4258	-11.9954	-11.8331	-5.2844	-14.4090	-2.9287
-5.6418	-10.9596	-6.8353	1.6410	-8.5969	-4.7880

表 9 SVM 部分訓練資料(5)

有障礙物，相對於聲源面向右					
343.75 Hz	375 Hz	406.25 Hz	437.5 Hz	468.75 Hz	500 Hz
-8.1009	-2.6886	1.4795	1.0887	13.6318	1.3304
-8.7547	-0.6977	4.4465	-5.1807	-12.6419	3.0378
-10.4636	1.9489	-2.6907	-8.6015	-4.9038	3.5847
14.5260	1.9102	-0.6031	17.0035	-2.2865	1.9544

表 10 SVM 部分訓練資料(6)

有障礙物，相對於聲源面向左					
343.75 Hz	375 Hz	406.25 Hz	437.5 Hz	468.75 Hz	500 Hz
-19.1788	-17.9150	15.1478	-17.3959	5.3624	-5.4933
-17.7148	-18.1472	18.3165	-15.9885	11.7731	-15.1064
-16.3082	-18.0972	-17.5491	-11.7094	14.0306	-13.8169
-15.8751	-14.3287	-8.9911	-5.9017	15.5264	-13.8252

觀察上列各表，可以發覺在無障礙物時，六個頻率之 Time-Delay 呈現較均勻的分佈，變動性較小，相反的再有障礙物的情況下，Time-Delay 呈現較不穩定的變動，其標準差比較大，根據這個特性，就可以利用 SVM 將這兩個不同的情況分類出來。

5.2 未來展望

本文中我們已發展一概念性可結合輪式機器人，系統主要分為兩部分，一為由嵌入式網路與感測器構成的基地端，另一部份則是建構於嵌入式系統上，擁有光流感測器、麥克風與全向輪之全方位運動平台，經由結合可將基地端變為可移動式機器人。透過聲場特徵、SVM 的分類與光流感測資訊，進行移動平台與基地間的導航與避障，並建構一完整的通訊系統，控制移動平台與感測器資料的傳遞。

目前於基地端配置有嵌入式無線網路與感測器等主動式元件，未來可利用別的方式進行結合使能夠移除基地端的嵌入式網路版與感測器，降低系統成本與複雜度。並且我們現在為使用人工移動方式進行 SVM 訓練資料的收集，未來可發展為利用攝影機與感測器，進行移動平台的自我訓練。另外可著手路徑規劃的研究，藉由多次訓練可以做更好的路徑規劃。

Reference:

- [1] Haruhiko Niwa, Tetsuya Ogata, Kazunori Komatani, and Okuno G. Hiroshi, "Distance Estimation of Hidden Objects Based on Acoustical Holography by applying Acoustic Diffraction of Audible Sound," *IEEE Int. Conf. on Robotics and Automation*, pp.423-428, 2007.
- [2] Nissanka B. Priyantha, Anit Chakraborty, and Hari Balakrishnan, "The Cricket Location-Support System," *ACM Int. Conf. on Mobile Computing and Networking*, August, 2000.
- [3] Tomonao Okubo and Kohei Yamamoto, "Simple prediction method for sound propagation behind edge-modified barriers," *Acoust. Sci. & Tech*, pp.7-15, 2007.
- [4] A. M. Ladd, K. E. Bekris, A. P. Rudys, D. S. Wallach, and L. E. Kavraki, "On the feasibility of using wireless ethernet for indoor localization," *IEEE Trans. Robotics and Automation*, vol. 20, no. 3, pp. 555-559, June 2004.
- [5] K. H. Kim, H. D. Choi, S. Yoon, K. W. Lee, H. S. Ryu, C. K. Woo, and Y. K. Kwak, "Development of Docking System for Mobile Robots Using Cheap Infrared Sensors," *1st Int. Conf. on Sensing Technology*, pp.287-291, 2005.
- [6] I. Ohya, A. Kosaka, and A. Kak, "Vision-based navigation by a mobile robot with obstacle avoidance using single-camera vision and ultrasonic sensing," *IEEE Trans. Robotics and Automation*, vol. 14, no. 6, pp. 969-978, Dec 1998.
- [7] C. H. Knapp, and G. C. Carter, "The generalized correlation method for estimation of time delay," *IEEE Trans. Acoustic, Speech, Signal Processing*, vol. 24, pp. 320-327, Aug 1976.
- [8] 大澤洋, 細沼直泰, "移動機器人用之充電系統, 搜索充電站之方法, 移動機器人, 連接器, 及電氣性連接構造", 中華民國專利, 申請案號 089122588, 2000 年 10 月 26 日。
- [9] 林文勛, 邱頂殷, "供自走機器用之自動充電座", 中華民國專利, 申請案號 093213754, 2005 年 6 月 11 日。
- [10] 燕成祥, "自走式電子裝置自動充電系統及其方法", 中華民國專利, 申請案號 093110957, 2006 年 7 月 11 日

- [11] V. Vapnik. “*The Nature of Statistical Learning Theory.*” Springer-Verlag, New York, 1995.
- [12] Nello Cristianini, and John Shawe-Taylor, “*An Introduction to Support Vector Machines and other kernel-based learning methods.*” Cambridge University Press, 2000.
- [13] Steve R. Gunn, “Support Vector Machine for Classification and Regression.” University of Southampton, 1998.
- [14] S. Lee and J.B. Song, “Robust mobile robot localization using optical flow sensors and encoders,” *IEEE Int. Conf. on Robotics and Automation*, pp.1039 – 1044, April, 2004
- [15] 蔡寬騰， “全方向輪式機器人之建構及其利用立體視覺與超音波資訊之導航與避障”，國立成功大學，碩士論文，民國 94 年。
- [16] P.F. Muir, C.P. Neuman, “Kinematic modeling of wheeled mobile robots,” *J. Robotic Systems*, 4(2):281-340, 1987.
- [17] S. M. Killough, F.G. Pin, “Design of an Omnidirectional and Holonomic Wheeled Platform Prototype, ” *IEEE Int. Conf. on Robotics and Automation*, pp. 84-90, May, 1992.
- [18] Y.P. Leow, K.H. Low and W.K. Loh, “Kinematic Modelling and Analysis of Mobile Robots with Omni-Directional Wheels,” *7th Int. Conf. on Control, Automation, Robotics and Vision (ICARCV’02)*, Dec, 2002.
- [19] GCC, the GNU Compiler Collection, <http://gcc.gnu.org>.
- [20] DSP Gateway, <http://dspgateway.sourceforge.net/pub/index.php>.