

國立交通大學

電機與控制工程學系

碩士論文

運用雷射測距儀之機器人定位設計

Mobile Robot Localization Using
A Laser Range Finder



研 究 生：鍾鎮謙

指導教授：宋開泰 博士

中 華 民 國 九 十 五 年 九 月

運用雷射測距儀之機器人定位設計

Mobile Robot Localization Using A Laser Range Finder

研 究 生：鍾鎮謙

Student: Chen-Chien Chung

指導教授：宋開泰 博士

Advisor: Dr. Kai-Tai Song

國立交通大學

電機與控制工程學系

碩士論文

A Thesis

Submitted to Department of Electrical and Control Engineering

College of Electrical and Computer Engineering

National Chiao Tung University

in Partial Fulfillment of the Requirements

for the Degree of Master

in

Electrical and Control Engineering

September 2006

Hsinchu, Taiwan, Republic of China

中華民國九十五年九月

運用雷射測距儀之機器人定位設計

學生:鍾鎮謙

指導教授:宋開泰 博士

國立交通大學電機與控制工程學系

摘要

智慧型機器人必須有在移動環境中自我定位的能力，以達成各種實際應用。本論文的目的在於設計一機器人之自我定位系統來估測機器人在環境中的姿態。本論文利用粒子濾波器設計位置估測的方法，藉由里程計的資訊先對機器人的位置做預測，再經由從距離感測器對環境的量測結果得到的機器人位置資訊來對預測的機器人位置做修正，最後由粒子濾波器估測出機器人的位置。本論文並以將感測資訊與環境地圖做比對的方式估測朝向角，將雷射掃描儀量測的環境距離資料以最小平方方法擬合成包含若干線段的線段組，再結合機器人的位置資訊與內建的环境地圖做比對，藉由比對結果可以估測出機器人的朝向角。本論文使用的距離感測器為雷射掃描儀，其具有相當高的精確度與可靠度，利用它來得到用來擬合成線段的環境距離資料。最後經由機器人繞行環境實驗的驗證，結果顯示所設計的機器人之定位系統確實有預期的功效。

Mobile Robot Localization Using A Laser Range Finder

Student: Chen-Chien Chung

Advisor: Dr. Kai-Tai Song

Department of Electrical and Control Engineering
National Chiao Tung University

ABSTRACT

Mobile robots should have the ability of self-localization for navigation in an environment. The objective of this thesis is to design a mobile robot localization system using a laser scanner. A particle filter is designed to estimate robot's position in this study. The particle filter predicts robot's position from odometer data, and updates the predicted robot's position by using the distance measurements of the environment. The localization system estimates robot's orientation by comparing the information from laser range finder with the environmental map. We can estimate the robot's orientation by first fitting the distance data from laser range finder to a line set, then combining the line set with robot's position information to match with the environment map. The system has been demonstrated using an experimental mobile robot. Several navigation experiments in an indoor environment have been carried out. Satisfactory experimental results are presented in this thesis.

誌謝

謹向我的指導老師宋開泰教授致上感謝之意，感謝他二年來的指導，以他豐富的學識與經驗，使本論文得以順利完成。感謝口試委員胡竹生教授、陳傳生教授與王傑智教授的指導及建議，讓本論文能更加的完整。

感謝學長任詔、奇謚、孟儒、嘉豪和奕龍在理論與實作上的指點，給予我許多寶貴的經驗。感謝與我一起奮鬥的同學晉懷、忠憲、宏宜及富全的相互鼓勵與提攜。感謝學弟富聖、俊瑋、振暘、濬尉、志昇與裕宏在生活上帶來的樂趣，及對於完成此論文所提供的協助。感謝我的朋友們，在我覺得沒力時，給我鼓勵與打氣。感謝所有在交大幫助過我的人，讓我能更順利地完成學業。

感謝我的兩位哥哥，在我的成長過程中各方面的關照。最後，特別感謝我的父母，由於他們的辛苦栽培，在生活上給予我細心地關懷與照料，使我才能夠順利完成此論文，在此我願以此論文獻給我最感激的父母親。



目錄

中文摘要	i
英文摘要	ii
誌謝	iii
目錄	iv
圖例	vi
表格	viii
第一章 緒論	1
1.1 研究動機	1
1.2 相關研究回顧	1
1.3 問題描述	6
1.4 章節說明	6
第二章 粒子濾波器	7
2.1 貝氏濾波器	7
2.2 粒子濾波器原理	11
第三章 自走式機器人之定位設計	16
3.1 設計概念	16
3.2 位置估測方法	19
3.2.1 預測(Prediction)	19
3.2.2 更新(Update)	22
3.3 朝向角估測方法	24
3.3.1 線段擷取	25
3.3.2 線段比對	29
第四章 模擬與實驗結果	32
4.1 機器人定位系統硬體架構機器人系統	32

4.1.1 機器人系統.....	32
4.1.2 雷射掃描儀.....	33
4.2 模擬結果.....	37
4.3 實驗結果.....	43
4.3.1 繞行實驗(一).....	43
4.3.2 繞行實驗(二).....	48
4.3.3 繞行實驗(三).....	50
第五章 結論與未來展望.....	53
5.1 結論.....	53
5.2 未來展望.....	53
參考文獻.....	55



圖例

圖 1-1 機器人姿態定義.....	2
圖 1-2 基於 Bayesian filter 之定位演算法架構圖[9].....	5
圖 2-1 動態貝氏網路示意圖.....	8
圖 3-1 機器人對其位置的不確定性範圍示意圖.....	16
圖 3-2 自走式機器人定位系統架構圖.....	18
圖 3-3 擷取線段座標轉換關係圖.....	22
圖 3-4 線段方程式參數定義示意圖.....	25
圖 3-5 判斷資料點是否包含於線段之示意圖.....	27
圖 3-6 線段擷取流程圖.....	28
圖 3-7 機器人在環境中擷取線段.....	28
圖 4-1 機器人硬體裝置架構圖.....	32
圖 4-2 機器人系統控制架構圖.....	33
圖 4-3 雷射掃描儀.....	34
圖 4-4 雷射掃描儀工作原理[11].....	34
圖 4-5 雷射掃描儀角度掃描範圍[11].....	34
圖 4-6 雷射掃描儀量測結果.....	37
圖 4-7 模擬環境與機器人繞行軌跡.....	37
圖 4-8 模擬一之機器人定位系統姿態估測誤差.....	39
圖 4-9 模擬二之機器人定位系統姿態估測誤差.....	41
圖 4-10 模擬三之機器人定位系統姿態估測誤差.....	42
圖 4-11 繞行實驗(一)之環境示意圖.....	43
圖 4-12 機器人與目標點相對位置示意圖.....	44
圖 4-13 繞行實驗(一)之實驗過程.....	46
圖 4-14 繞行實驗(一)之結果.....	47

圖 4-15 繞行實驗(二)之環境示意圖.....	48
圖 4-16 繞行實驗(二)之結果.....	49
圖 4-17 繞行實驗(三)之環境示意圖.....	50
圖 4-18 繞行實驗(三)之結果.....	51



表格

表 3-1 機器人在圖 3-7 環境中擷取的線段資料.....	29
表 4-1 要求回傳量測值指令格式[11].....	35
表 4-2 要求回傳量測值命令之反應[11].....	36
表 4-3 模擬一之機器人姿態估測絕對誤差.....	39
表 4-4 模擬二未加雜訊部分之機器人姿態估測絕對誤差.....	41
表 4-5 模擬二加雜訊部分之機器人姿態估測絕對誤差.....	41
表 4-6 模擬三之機器人姿態估測絕對誤差.....	42



第一章 緒論

1.1 研究動機

對於一個多功能的智慧型機器人來說，自我定位是其必備的功能之一。智慧型機器人能夠自我定位，即它能知道自己目前在所處環境中的位置，機器人就可以在一個平面空間裡自由移動。有了自我定位的能力，機器人可以執行的任務將增加許多，並且更加能夠展現其智慧型行為的能力。例如，人可以命令機器人到指定的地方去取物品、可以引領人到平面空間中的任何人們想要到達的地方。拓展了機器人的應用面。

未來可將有定位能力的機器人應用於保全工作中，藉由機器人的自主移動，讓機器人巡邏整個環境區域。由於機器人能夠自我定位，機器人可以持續的巡邏而不會迷失自己的方位。另外，具定位能力之機器人亦可應用於家庭看護中，對於行動不便的老年人，機器人可以輔助其移動或移動到家庭中的某一地點去取得老年人指定的物品。而機器人要能夠移動到家庭中的某個地點，首先它要知道自己目前所處的位置，亦即要有自我定位功能的機器人才能夠來完成這項任務。除此之外，亦可將有定位能力的機器人應用於博物館中，作為導覽機器人，可以帶領遊客到博物館內的各個區域參觀，當遊客迷失方位的時候，也可藉由機器人的帶領，而到達遊客想參觀的區域。

以上的應用皆須藉著機器人的自我定位與對於周遭環境的了解能力才能達成，即機器人能夠知道自己目前的位置且對於周遭的環境能探索了解而不會迷失自己的方位。所以這個研究主題是相當有趣且重要的，值得去做進一步的探討。

1.2 相關研究回顧

自走式機器人定位問題是要求得機器人在移動環境中的姿態(Pose)，即機器人的位置與朝向角， (x, y, θ) 值，其定義如圖 1-1 所示。其中朝向角定義為機器

人正前方與座標系 x 軸的夾角。

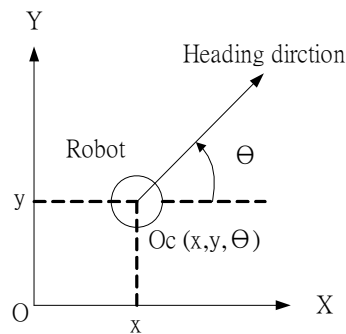


圖 1-1 機器人姿態定義

使用里程計(odometer)的資訊來達成機器人自我定位的方法稱作 dead-reckoning。由輪型機器人兩個輪子個別轉動的圈數，並可將其轉換成機器人左右輪個別的線性移動距離，加上已知的前一時刻的位置與朝向角資訊，便可以估測出機器人目前所在的位置。不過，里程計會受到輪子打滑與路面不平的影響，使得從里程計得到的機器人移動距離與實際上機器人的移動情形不一致，並且由此所造成的機器人位置估測誤差是不斷地累積的，所以 dead-reckoning 只適合在短距離移動的機器人應用中。為使機器人在更長距離的移動中能持續地自我定位，加入其它感測器的使用，以及一套有效的絕對位置定位演算法是必要的。

在以地圖為基礎的定位方法中，以內建的全域地圖與外部環境感測器的測量資料來決定機器人的姿態。機器人將由感測器得到的局部地圖與全域地圖做比對 (Map-matching)，以全域地圖為基準來修正目前機器人的姿態。在相關文獻中，外部環境感測器可以是影像感測器(Image sensor)[1]、超音波感測器(Ultrasonic sensor)[2]或雷射掃描儀(Laser scanner)等。在這些感測器中，雷射掃描儀因有高精確度的優點，所以廣泛的被運用。使用雷射掃描儀為感測器而產生的定位演算法可以依用來比對的資料型態不同而分成 point-based 定位演算法[3]與 feature-based 定位演算法[4]。Point-based 定位演算法比較連續兩筆掃描資料，運用最小平方法得到點對點(Point-to-point)的最佳對應關係，並以此推算機器人的

姿態。另一方面，feature-based 定位演算法則從掃描的距離資料中擷取環境特徵，如牆壁或牆角，將擷取出的特徵和全域地圖比對來得到機器人的姿態。

本論文設計的定位系統將使用 feature-based 的地圖比對方法，以環境中的牆面作為環境特徵，從雷射掃描的距離資料擷取出代表牆面的線段並與環境地圖比對並由比對結果來估測機器人的朝向角。與 point-based 的方法比較，feature-based 方法的比對結果其精確度較高，這是因為 feature-based 方法是以掃描資料整體的形狀(如代表牆面的形狀)作為地圖比對的依據，而 point-based 方法則是個別的去比對每一筆距離資料，又個別的距離資料會受到量測誤差的影響，降低了比對結果的正確性，所以 point-based 方法的比對結果並不如 feature-based 方法來得精確。不過，point-based 方法可以使用在非多邊形(Non-polygonal)為主的環境中，而本論文使用的 feature-based 方法則只能運用在多邊形(Polygonal)為主的環境中。

另外有些研究以在環境中設置地標的方式，透過感測器對地標的量測來得到機器人在環境中的姿態資訊。因此必須要先記錄所設置之地標在環境中的位置，則機器人對於地標的量測資料才能成為有用的定位資訊。文獻[5]採用影像處理的方式，在環境中設置特定顏色圖案的地標，讓機器人上的影像感測器能在環境中辨認出此地標，並對地標做量測來得到機器人的姿態資訊。文獻[6]則以圓柱狀的物體作為地標設置在環境中，並使用雷射掃描儀作為感測器來辨認圓柱狀地標的幾何形狀。辨認出地標後便可得到機器人的姿態資訊。

許多的研究以機率理論來解決機器人定位問題。目前以機率理論發展出來的機器人定位演算法中，主要有卡曼濾波器(Kalman Filter)演算法[7]和 MCL(Monte Carlo Localization)演算法[8]。因為機器人在對其所在位置做估測時存在著不確定性，為了要將此不確定性以數學來表示，所以需要以機率理論來作為定位演算法的理論基礎。另外，機器人在估測其所在位置時，必需將裝在機器人上的感測器所得到資訊作為輸入，經過定位演算法處理後才能估測出機器人之所在位置。不過，感測器在做測量時必定存在著量測誤差，而量測誤差雜訊經常是複雜的且不

容易將它對測量資訊所造成的影響消除。在機率理論中將雜訊以隨機變數(Random Variable)來表示，所謂的隨機變數即是它在每一時刻的數值皆是隨機的，並以機率密度函數(Probability Density Function)來表示，所以以量測雜訊的機率密度函數來當作感測器的感測誤差模型，便可以處理感測誤差所造成的影響。而這是以機率理論來建構機器人定位演算法的另一個主要原因。由以上的這些原因，不難看出為何以機率統計的方法來解決機器人定位問題會如此重要了。

卡曼濾波器(Kalman Filter)演算法利用了目前機器人的感測器量得的資訊以及機器人目前的位置來推測機器人在下一時刻的位置，以達成機器人的自我定位。所以我們可以發現在使用卡曼濾波器演算法來解決機器人自我定位問題時，我們必須要有機器人的運動模型(Motion Model)以及感測器的感測誤差模型(Measurement Model)才能對機器人的位置進行估測。在卡曼濾波器演算法中假設所使用的機率模型為高斯分佈(Gaussian Distribution)，亦即常態分佈(Normal Distribution)，而後經過遞迴的運算來估測出機器人的位置狀態。卡曼濾波器定位演算法其優點為精確度高且收斂速度快，而缺點為使用的機率模型必須是高斯機率分佈，若不是高斯分佈，則其估測結果的準確度將會降低。

MCL 演算法，亦即以粒子濾波器(Particle Filter)理論為基礎產生的機器人定位演算法，是以取樣(Sampling)的方式(Sample-based)來表示機器人姿態的機率密度函數，並給予每一個取樣點(Sample)或稱為粒子(Particle)一個機率值，隨後再加入感測器的資訊對每一個粒子給予不同的權重(Weight)，最後再對所有的取樣點重新取樣(Resampling)，而重新取樣後得到的新的一組取樣點將成為下一個循環的開始。所以 MCL 的定位演算法以不斷地取樣、給與取樣點權重以及重新取樣的方式來對機器人的位置狀態做估測與修正。MLC 定位演算法的優點在於能夠表示具有多種或不同的機率密度函數，這是因為 MLC 演算法是以取樣的方式來表示機器人姿態的機率密度函數。

將卡曼濾波器與 MCL 這兩種定位演算法做比較，可以發現卡曼濾波器定位演算法只適用於機率模型為高斯分佈(Gaussian distribution)或接近於高斯分佈的

系統，而 MCL 演算法則可以適用與具有多種或不同機率模型之系統中。不過，從另一方面來看，只要卡曼濾波器演算法其使用的機率模型接近高斯分佈，則其所得到的位置狀態(State)估測結果將會比 MLC 演算法要來得精確，而且機率模型愈接近高斯分佈，使用卡曼濾波器演算法得到的位置狀態估測結果將會愈精確。雖然卡曼濾波器演算法比 MCL 演算法還要早被提出，但是兩種用來定位的演算法各有其長處，所以個別皆有不少的專家、學者在進行研究與運用來解決機器人的自我定位問題，甚至持續地提出許多改良的法方。

不論是粒子濾波器演算法或卡曼濾波器演算法，皆是以 Bayesian filter 為基本架構發展而來的。在 Bayesian filter 演算法中，主要分成兩個步驟，分別是預測(Predict)和更新(Update)。因此，在以粒子濾波器演算法或卡曼濾波器演算法來解決機器人定位問題的過程中，也可被分成預測階段(Prediction)與感測器資訊更新階段(Measurement update)，如圖 1-2 所示。

在 Prediction 階段，定位演算法利用 odometer 的資料來對機器人的位置資訊做預測。不過以 odometer 的資料來估測機器人的位置，會因為路面不平或輪子打滑的影響而產生誤差，並且誤差會隨著機器人的移動而持續累積。所以在 Update 階段時，必須依靠其它感測器得到的資訊來對在 Prediction 階段預測出的機器人位置做修正，如此才能正確地估測出機器人的位置。而在 Update 階段所使用的感測器一般來說大部分是影像感測器、超音波感測器或雷射掃描儀等。

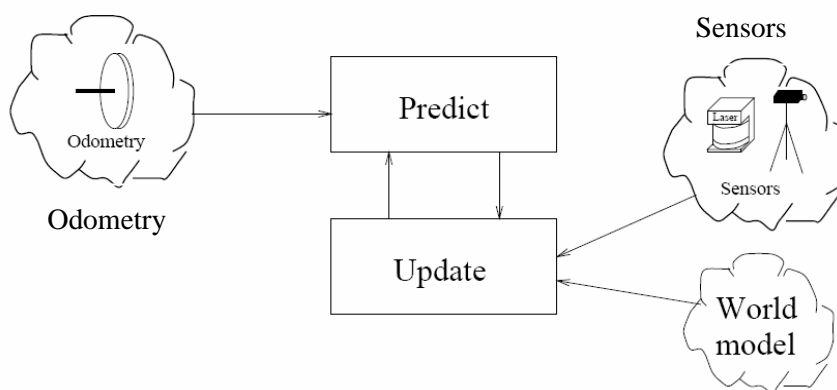


圖 1-2：基於 Bayesian filter 之定位演算法架構圖[9]

1.3 問題描述

為了讓自走式機器人能夠完成更多樣的任務與應用，自走式機器人必須要有在環境中自我定位的能力。因此，如何能讓機器人得到它所處的環境資訊與在所處的環境區域中能夠自我定位，即機器人能知道自己目前所處的位置及朝向角，是這個研究主題所要達成的目標。

以 Dead-reckoning 來估測機器人的姿態，會有累積誤差的問題，造成所估測的機器人姿態誤差愈來愈大。因此，在定位系統中需加入其它感測器的使用以及有效的定位演算法，才能在機器人長距離的移動過程中不使估測誤差累積，而仍可有效的估測機器人姿態。

文獻[5][6]在環境中設置地標，藉由感測器對地標的量測，來得到機器人在環境中的姿態參考資訊。此法雖然實用，不過當機器人移動到新的環境時，就需重新設置地標，使機器人定位系統的適用環境不夠彈性。若能從感測器的資訊擷取出環境中原有的特徵，並將此環境特徵當作修正機器人姿態的參考資訊，便可不用設立特定地標。在一般室內環境中，牆壁為主要的結構，而牆壁可以簡單的線段來表示，所以採用牆壁構成的線段當作環境特徵。因此，本論文希望以粒子濾波器演算法為基礎結合 feature-based 的地圖比對方式，來設計機器人在已知初始姿態與環境地圖資訊的情況下的自我定位系統。

1.4 章節說明

本論文共分為五章。第一章為緒論，介紹研究動機與相關的研究背景。第二章介紹粒子濾波器的理論背景，並包含貝氏濾波器的介紹。第三章介紹以粒子濾波器為基礎的機器人定位演算法設計與機器人朝向角的估測方法。第四章介紹模擬與實驗結果，以驗證定位系統的設計。第五章為結論與未來工作。

第二章 粒子濾波器(Particle Filter)

本章之重點在於介紹粒子濾波器的理論基礎。粒子濾波器演算法是由貝氏濾波器(Bayesian filter)演算法衍生而來的，故先介紹貝氏濾波器演算法，說明貝氏濾波器演算法如何的以遞迴(Recursive)的方式來對系統狀態做估測。隨後才是粒子濾波器演算法理論的介紹。

2.1 貝氏濾波器(Bayesian Filter)

在介紹貝氏濾波器演算法[12]之前，必須先說明幾個貝氏濾波器應用在自走式機器人領域的重要名詞，以便後續的介紹：

(1) 狀態轉移機率(State transition probability)：

以條件機率函數 $p(x_t | x_{t-1}, u_t)$ 來表示，其中 x_t 與 x_{t-1} 分別表示在 t 和 $t-1$ 時刻的系統狀態，在機器人定位系統中即是機器人在 t 與 $t-1$ 時刻的姿態(Pose)，而 u_t 則是在 t 時刻的機器人控制命令。此機率函數說明了在已知目前控制命令與前一時刻機器人狀態的條件下，目前機器人狀態的機率分佈情形。

(2) 量測機率(Measurement probability)：

以條件機率函數 $p(z_t | x_t)$ 來表示，其中 z_t 為系統在 t 時刻的量測資訊，在機器人定位系統中即是機器人在時刻 t 由感測器測得的感測資訊。量測機率函數說明了機器人在目前的狀態下(x_t)，從感測器量得的環境資訊的機率分佈 z_t 。

(3) 相信度函數(Belief)：

以 $bel(x_t)$ 符號表示狀態 x_t 的相信度函數，並可寫成

$$bel(x_t) = p(x_t | z_{1:t}, u_{1:t}) \quad (2-1)$$

其中 $p(x_t | z_{1:t}, u_{1:t})$ 通常稱作 Posterior，是系統狀態 x_t 在已知所有過去的感測資訊 $z_{1:t}$ 與控制命令 $u_{1:t}$ 的情況下的機率分佈函數。在未包含 t 時刻的量測資訊 z_t 前的 Posterior 通常稱做 Prior，而未包含 z_t 的相信度函數以 $\overline{bel}(x_t)$ 符號表示，可寫成如下的式子：

$$\overline{bel}(x_t) = p(x_t | z_{1:t-1}, u_{1:t}) \quad (2-2)$$

在貝氏濾波器演算法中 $\overline{bel}(x_t)$ 是 Prediction 階段之後得到的，而在加入 t 時刻的量測資訊 z_t 後，便可從 $\overline{bel}(x_t)$ 算出 $bel(x_t)$ ，這個過程稱作 Measurement update。

圖 2-1 說明了在貝氏濾波器中系統狀態、控制命令與量測資訊之間隨時間的演化過程，在時間 t 的系統狀態是根據同時刻的控制命令 u_t 與前一時刻的系統狀態而來，而 t 時刻的量測資訊 z_t 則是根據系統在狀態為 x_t 時的量測而來。如此的演化模式稱作動態貝氏網路(Dynamic Bayes network,DBN)[12]。

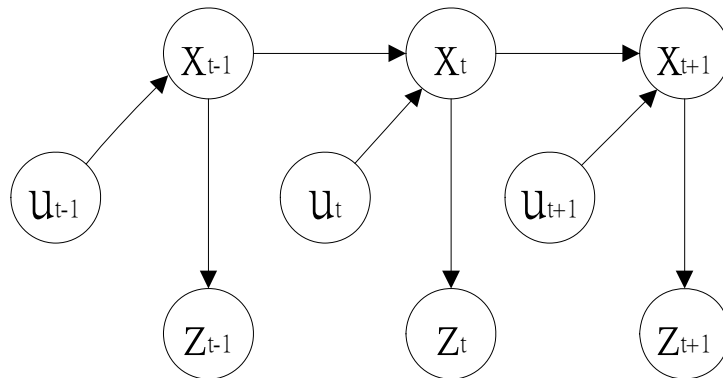


圖 2-1 動態貝氏網路示意圖

貝氏濾波器演算法的目的是要求得系統狀態的相信度函數 $bel(x_t)$ ，並且以遞

迴的方式從感測資訊與控制命令來求得 $bel(x_t)$ ，所以貝氏濾波器的輸入是 $t-1$ 時刻的相信度函數 $bel(x_{t-1})$ 與 t 時刻的感測資訊與控制命令，而輸出即是 t 時刻的相信度函數 $bel(x_t)$ 。

貝氏濾波器演算法主要分成兩個步驟來求得 $bel(x_t)$ ，分別是 Prediction 與 Update：

1. 預測階段(Prediction)：

$$\overline{bel}(x_t) = \int p(x_t | u_t, x_{t-1}) bel(x_{t-1}) dx_{t-1} \quad (2-3)$$

2. 更新階段(Update)：

$$bel(x_t) = \eta p(z_t | x_t) \overline{bel}(x_t) \quad (2-4)$$

在 Prediction 階段，貝氏濾波器利用了前一時刻的相信度函數 $bel(x_{t-1})$ 與控制命令 u_t ，並對兩個機率函數，狀態轉移函數 $p(x_t | x_{t-1}, u_t)$ 與相信度函數 $bel(x_{t-1})$ 相乘的結果做積分，從積分的結果我們可以得到 $\overline{bel}(x_t)$ 。而在 Update 階段，貝氏濾波器則是將前一步驟得到的 $\overline{bel}(x_t)$ 乘上量測機率 $p(z_t | x_t)$ ，相乘後的結果可以得到最後的結果相信度函數 $bel(x_t)$ 。因為兩個機率函數相乘的結果不一定仍為一機率函數，即將兩機率函數相乘的結果積分不一定為 1，所以在(2-4)式中必須要乘上一係數 η 使得 $\overline{bel}(x_t)$ 為正常的機率函數。

接著是數學的推導，將證明時刻 t 的相信度函數 $bel(x_t)$ 確實可由前一時刻的相信度函數 $bel(x_{t-1})$ 經由貝氏濾波器演算法的計算求出。以下的推導必須有兩個假設為前提：

1. 假設系統狀態 x_t 包含了所有過去時間的量測資訊與控制命令的資訊(Markov Assumption)。

2. 假設控制命令是隨機產生的並且已知。

有了假設之後，首先我們將 $bel(x_t)$ ，即 target posterior $p(x_t | z_{1:t}, u_{1:t})$ 以貝氏定理展開成：

$$\begin{aligned} p(x_t | z_{1:t}, u_{1:t}) &= \frac{p(z_t | x_t, z_{1:t-1}, u_{1:t}) p(x_t | z_{1:t-1} | u_{1:t})}{p(z_t | z_{1:t}, u_{1:t})} \\ &= \eta p(z_t | x_t, z_{1:t-1}, u_{1:t}) p(x_t | z_{1:t}, u_{1:t}) \end{aligned} \quad (2-5)$$

接著利用假設 1，如果我們已知系統狀態 x_t ，而且我們想要得到量測資料 z_t 的資訊，則過去時間的所有控制命令 $u_{1:t-1}$ 與量測資料 $z_{1:t-1}$ 皆無關於系統對於 z_t 的量測結果。以數學表式可得到：

$$p(z_t | x_t, z_{1:t-1}, u_{1:t}) = p(z_t | x_t) \quad (2-6)$$

所以我們便可以將(2-5)式簡化成：

$$p(x_t | z_{1:t}, u_{1:t}) = \eta p(z_t | x_t) p(x_t | z_{1:t-1}, u_{1:t}) \quad (2-7)$$

可以得到：

$$bel(x_t) = \eta p(z_t | x_t) \overline{bel}(x_t) \quad (2-8)$$

(2-8)式即是貝式濾波器在 Update 階段所使用的方程式。

另外，我們利用 Theorem of total probability 將 $\overline{bel}(x_t)$ 展開可以得到：

$$\begin{aligned} \overline{bel}(x_t) &= p(x_t | z_{1:t-1}, u_{1:t}) \\ &= \int p(x_t | x_{t-1}, z_{t-1}, u_{1:t}) p(x_{t-1} | z_{1:t-1}, u_{1:t}) dx_{t-1} \end{aligned} \quad (2-9)$$

再一次利用假設 1，若已知系統狀態 x_{t-1} ，則我們可以不用考慮 t-1 時刻之前的控

制命令與量測資訊，但不包括 u_t ，可以得到：

$$p(x_t | x_{t-1}, z_{1:t-1}, u_{1:t}) = p(x_t | x_{t-1}, u_t) \quad (2-10)$$

最後利用假設 2，使得 u_t 可以從機率函數 $p(x_{t-1} | z_{1:t-1}, u_{1:t})$ 中去除，所以我們可以從 (2-9) 式得到：

$$\overline{bel}(x_t) = \int p(x_t | x_{t-1}, u_t) p(x_{t-1} | z_{1:t-1}, u_{1:t-1}) dx_{t-1} \quad (2-11)$$

(2-11) 式即是使用在貝式濾波器中 Prediction 階段的式子。所以經由上面的證明可以得到貝氏濾波器確實可以正確地由前一時刻的相信度函數 $bel(x_{t-1})$ 加上控制命令與量測資訊，得到目前時刻的 $bel(x_t)$ 。

2.2 粒子濾波器原理



粒子濾波器演算法是基於貝氏濾波器發展而來的[12]，它的特色是將我們想要的 Posterior density function $bel(x_t)$ 以一組隨機的取樣點(Sample)或稱為粒子(Particle)來表示，並且每一個 Sample 都有其相對應的權重(Weight)，再利用這一組取樣點與其對應的權重來估測系統狀態。所以粒子濾波器能夠表示更多樣的機率函數。

通常可以將有 M 個粒子的粒子組(Particle set) X_t 表示成：

$$X_t := x_t^{[1]}, x_t^{[2]}, \dots, x_t^{[M]} \quad (2-12)$$

每一個 Particle $x_t^{[m]}$ ($1 < m < M$) 都代表著在 t 時刻系統的某一個狀態。故粒子濾波器演算法的概念是以這一組 Particle X_t ，來近似 Posterior density $bel(x_t)$ 。既然是用來近似 Posterior density $bel(x_t)$ ，每一個 Particle 都應該與 $bel(x_t)$ 成一定的比例關係：

$$x_t^{[m]} \sim p(x_t | z_{1:t}, u_{1:t}) \quad (2-13)$$

從(2-12)式可以得知，若在系統狀態空間中某個狀態附近有 particle 分佈得特別密集，則此狀態很可能就是真實的系統狀態。

如同貝氏濾波器演算法，粒子濾波器演算法亦是遞迴(Recursive)演算的架構，從前一時刻的 Posterior $bel(x_{t-1})$ 求得目前時刻的 Posterior $bel(x_t)$ 。但是粒子濾波器是以 Particle Set 來表示 $bel(x_t)$ ，故粒子濾波器是遞迴地從前一時刻的 Particle set X_{t-1} 經由演算法的計算而求得目前的 Particle set X_t 。所以粒子濾波器演算法的輸入就是前一時刻的 Particle set X_{t-1} 與目前的控制命令 u_t 以及量測資訊 z_t ，先將 X_{t-1} 裡的每一個 particle $x_{t-1}^{[m]}$ 處理後，變成一組用來表示 Prior 相信度函數 $\overline{bel}(x_t)$ 的 Particle Set \overline{X}_t ，隨後再將 \overline{X}_t 轉換成用來近似 Posterior 相信度函數 $bel(x_t)$ 的 Particle Set X_t 。

粒子濾波器演算法的流程主要可以分成三個步驟，分別是預測(Prediction)、設定權重(Assign weight)與重新取樣(Resampling)：

1. 預測(Prediction)：

在這個步驟中，從前一時刻的 Particle Set X_{t-1} 與控制命令 u_t 產生另一組 Particle Set \overline{X}_t ，而 \overline{X}_t 即是用來近似 \overline{bel}_t 的 Particle Set。產生的方式為對狀態轉移機率函數 $p(x_t | u_t, x_{t-1})$ 做取樣。即：

$$x_t^{[m]} \sim p(x_t | u_t, x_{t-1}^{[m]}) \quad (2-14)$$

2. 設定權重(Assign weight)：

這個步驟將產生每一個 Particle $x_t^{[m]}$ 其相對應的權重 $w_t^{[m]}$ 。計算權重 $w_t^{[m]}$ 的

目的是要將量測資訊 z_t 加入到 Particle Set 中，計算的方式為：

$$w_t^{[m]} = p(z_t | x_t^{[m]}) \quad (2-15)$$

因為權重 $w_t^{[m]}$ 的加入使得原來的 Particle Set 從用來表示 $\overline{bel}(x_t)$ ，變成用來表示 $bel(x_t)$ ，所以下式也可代表權重 $w_t^{[m]}$ 的涵意：

$$w_t^{[m]} = \frac{\text{target distribution}}{\text{proposal distribution}} \quad (2-16)$$

其中 target distribution 與 proposal distribution 分別對應 $bel(x_t)$ 與 $\overline{bel}(x_t)$ 。

3. 重新取樣(Resampling)：

這個步驟根據前一步驟產生的權重 $w_t^{[m]}$ ，將原來的 Particle Set 轉換成另一組數目相同的 Particle Set，轉換的方式是在權重大的 Particle 附近取更多的 Particle，反之將權重小的 Particle 減少或直接消去，如此 Particle 會集中在權重大的區域附近，使得 Particle Set 愈接近真實的系統狀態。所以在 Resampling 步驟之前，Particle Set 是根據 Prior $\overline{bel}(x_t)$ 而分佈的，而在 Resampling 步驟之後，Particle Set 的分佈則是近似於 Posterior $bel(x_t)$ 。

粒子濾波器的這三個主要步驟，第 1 步驟 Prediction 與貝氏濾波器相同，而步驟 2 和 3，Assign weight 與 Resampling，則是對應貝氏濾波器的 Update 階段。

接著是數學的推導，將證明(2-15)的 target distribution $bel(x_t)$ 的確可經由權重 $w_t^{[m]}$ 與 proposal distribution 的運算來求得。在此我們將考慮 state sequence：

$$x_{0:t} = x_0, x_1, \dots, x_t \quad (2-17)$$

而非先前的單一個 state x_t 。所以考慮整個 state sequence 的 Posterior $bel(x_{0:t})$ 可寫成：

$$bel(x_{0:t}) = p(x_{0:t} | u_{1:t}, z_{1:t}) \quad (2-18)$$

並且可進一步化簡成：

$$\begin{aligned} & p(x_{0:t} | u_{1:t}, z_{1:t}) \\ &= \eta p(z_t | x_{0:t}, z_{1:t-1}, u_{1:t}) p(x_{0:t} | z_{1:t-1}, u_{1:t}) \\ &= \eta p(z_t | x_t) p(x_{0:t} | z_{1:t-1}, u_{1:t}) \\ &= \eta p(z_t | x_t) p(x_t | x_{0:t-1}, z_{1:t-1}, u_{1:t}) p(x_{0:t-1} | z_{1:t-1}, u_{1:t}) \\ &= \eta p(z_t | x_t) p(x_t | x_{t-1}, u_t) p(x_{0:t-1} | z_{1:t-1}, u_{1:t-1}) \end{aligned} \quad (2-19)$$

(2-18)中的第 2 與第 4 個等號使用了 Markov assumption 來化簡。此外，我們假設 Proposal distribution 為：

$$p(x_t | x_{t-1}, u_t) bel(x_{0:t-1}) = p(x_t | x_{t-1}, u_t) p(x_{0:t-1} | z_{1:t-1}, u_{1:t-1}) \quad (2-20)$$

所以由(2-18)和(2-19)式，我們可以將(2-15)式展開成：

$$\begin{aligned} w_t^{[m]} &= \frac{\text{target distribution}}{\text{proposal distribution}} \\ &= \frac{\eta p(z_t | x_t) p(x_t | x_{t-1}, u_t) p(x_{0:t-1} | z_{1:t-1}, u_{1:t-1})}{p(x_t | x_{t-1}, u_t) p(x_{0:t-1} | z_{1:t-1}, u_{1:t-1})} \\ &= \eta p(z_t | x_t) \end{aligned} \quad (2-21)$$

將(2-20)式的結果與(2-14)式做比較，兩式相一個差係數 η ，而在 Resampling 時根據的是每一個 Particle 的權重大小之間的相互比例關係，並非權重的數值，所以係數 η 是不重要的，(2-20)式仍和(2-14)式相符合。

因此粒子濾波器演算法在 Resampling 步驟中根據權重大小重新產生的

Particle Set，其分佈情況的確是與 Proposal distribution 和權重兩者相乘的結果符合，亦即：

$$\eta w_t^{[m]} p(x_t | x_{t-1}, u_t) p(x_{0:t-1} | z_{0:t-1}, u_{0:t-1}) = bel(x_{0:t}) \quad (2-22)$$

所以 target distribution $bel(x_t)$ 確實可經由權重 $w_t^{[m]}$ 與 proposal distribution 經由粒子濾波器演算法的運算得到。

由上面的介紹以粒子濾波器來做機器人定位演算法設計的步驟為：

Step1：以粒子組 $X_t = \{(x_t^{[m]}, w_t^{[m]}), m=1, \dots, M\}$ 代表機器人的姿態。

Step2：利用控制命令 u_t 與時刻 $t-1$ 的粒子組 X_{t-1} 產生新的粒子組 \bar{X}_t 來代表預測的機器人姿態。產生方式為： $x_t^{[m]} \sim p(x_t | u_t, x_{t-1}^{[m]})$ 。

Step3：根據外部環境感測器的資訊來修正前一步驟預測的機器人姿態，根據量測資訊 z_t 更新每個粒子的權重。

$$w_t^{[m]} = p(z_t | x_t^{[m]})$$

Step4：將權重正規化，使 $\sum_m w_t^{[m]} = 1$ 。

Step5：設定權重臨界值 w_{th} 。

Step6：對每一個粒子，若 $w_t^{[m]} > w_{th}$ ，則保留此粒子，否則消去此粒子。

Step7：從保留的粒子產生新的粒子直到總粒子數為 M 。

接著重覆 Step2 到 Step7。

第三章 自走式機器人定位設計

本章將介紹機器人自我定位演算法之設計，說明以粒子濾波器演算法結合地圖比對的方法，來設計自走式機器人定位系統。一般來說，機器人的定位問題就是要求得機器人在環境中的姿態(Pose)，也就是位置與朝向角 (x, y, θ) 。在本論文所設計的機器人定位系統中，位置的估測是經由粒子濾波器的處理求得，而機器人的朝向角估測則是經由感測器得到的環境資訊與機器人已知的環境地圖作比對來求得。

3.1 設計概念

一般在自走式機器人的移動過程中，可以使用里程計(Odometer)的回傳資訊以及已知的機器人初始位置與朝向角來估測機器人目前的位置與朝向角，但是使用這種定位方法最常遇到的問題就是累積誤差(Accumulate Error)。也就是說，機器人在每次移動之後，其對本身位置資訊的不確定性範圍就會不斷變大。如圖 3-1 所示，當機器人依序經過 $A \rightarrow B \rightarrow C \rightarrow D \rightarrow E$ 時，我們可以發現其對本身

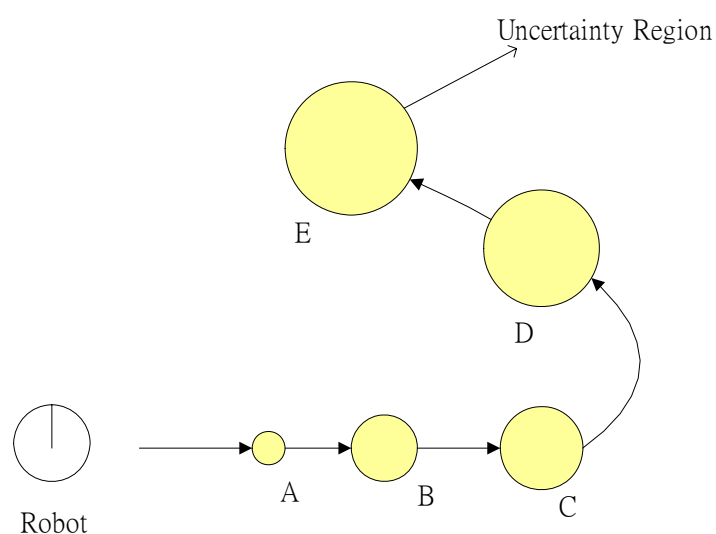


圖 3-1 機器人對其位置的不確定性範圍示意圖

位置的不確定性將會愈來愈高，這種情況在自走式機器人的應用當中是不被允許的，因為這容易造成機器人產生錯誤的後續動作。因此在機器人定位系統中還必須加上其它感測器的使用與一套有效的定位演算法，才能確實解決自走式機器人定位問題。

本論文以粒子濾波器演算法結合地圖比對的方式設計機器人定位系統，並使用雷射掃描儀(Laser Range Finder)作為用來得到環境資訊的感測器。在機器人朝向角估測方面，將雷射掃描儀量測的機器人與環境周圍的相對距離，以最小平方方法做線段擬合可以得到用來表示機器人周圍環境的線段組合，而這些線段組合是代表機器人在目前的位置與朝向角所面對的局部環境地圖。所以我們將此線段組合與機器人已知的內建地圖做比對，進行比對的時候須加上機器人前一時刻的位置與朝向角資訊，如此可以減少比對結果的多種可能性(Ambiguity)，而從最後的比對結果我們便可以估測出機器人的朝向角。

因此要由線段比對來估測機器人朝向角，必須要有機器人的位置資訊當作輸入，位置資訊愈準確則得到的朝向角也會愈準確。所以在設計上，我們以粒子濾波器來進行位置的估測，希望能估測出精確的位置資訊，進而使線段比對的朝向角估測方法也能精確地估測朝向角。

首先，將雷射掃描儀所量測的機器人與環境周圍的相對距離，以最小平方方法產生用來表示機器人周圍環境的線段組合，然後將此環境線段組合的線段表示法從以機器人為座標中心的區域座標系轉換成以內建地圖座標原點為中心的全域座標系，如此便可得到環境線段組合與內建地圖的對應關係，再由機器人與環境線段組合中線段的相對距離可以得到機器人在環境中的位置資訊，最後經由粒子濾波器對機器人的位置做估測。

根據上面陳述的概念，本論文所設計的自走式機器人定位系統的架構圖，如圖 3-2 所示，系統包含了兩個感測器，分別是里程計(軸編碼器)與雷射掃描儀。

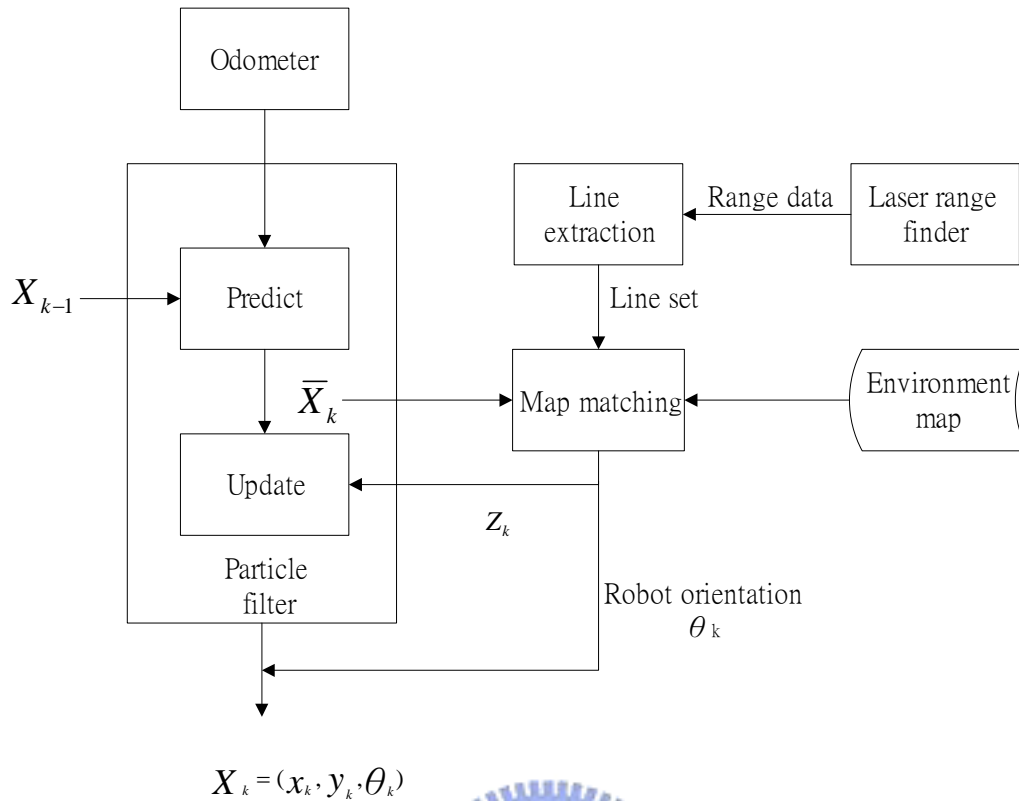


圖 3-2 自走式機器人定位系統架構圖



機器人定位系統的輸入為兩個感測器的量測資料以及前一時刻的機器人姿態

$X_{t-1} = (x_{t-1}, y_{t-1}, \theta_{t-1})$ ，輸出為機器人目前的姿態 $X_t = (x_t, y_t, \theta_t)$ 。

一開始里程計的資訊與前一時刻的機器人姿態 X_{t-1} 輸入到系統的粒子濾波器中，由粒子濾波器對機器人目前的姿態做預測得到 \bar{X}_t ，接著利用雷射掃描儀的量測資料來產生環境線段組合，計算機器人與線段組的相對距離並將結果輸入到粒子濾波器中，經由濾波器中 Update 階段的處理，可以得到機器人的位置資訊。另外，由雷射掃描儀對環境的距離量測，經由線段擷取(Line extraction)可得到一組線段組合，再將此線段組合與環境地圖做比對，可得到機器人的朝向角資訊。最後將位置和朝向角資訊結合，就可得到系統的輸出，即機器人目前的姿態 X_t 。以上所描述的便是設計的機器人定位系統的運作流程。

綜合以上的討論，本論文所設計的機器人定位系統運作之前必須有兩個已知條件：

- 1) 機器人在環境中的初始位置與朝向角。
- 2) 機器人所處環境的地圖。

3.2 位置估測方法

本節介紹以粒子濾波器理論為基礎而設計的機器人位置估測方法，其中包含了粒子濾波器演算法的兩個主要步驟，預測階段(Prediction)與更新(Update)階段。說明如何利用感測器的資訊與前一時刻機器人的姿態，經由粒子濾波器的處理來估測機器人的位置資訊。而機器人朝向角的部分，將在下一節以線段擷取與比對的方式加入此節產生的位置資訊當作輸入來求得，並未包含在粒子濾波器的設計中。



3.2.1 預測(Prediction)

將粒子濾波器應用在機器人位置估測問題中，粒子在環境中散佈的位置就是粒子濾波器對於機器人位置的估測結果。如下式所示：

$$s_t^m = (x^m, y^m) \quad (3-1)$$

其中 s_t^m 代表在時間 t 的第 m 個粒子

一開始因為我們已知機器人的初始位置，所以將粒子濾波器的所有粒子(Particles)均勻散佈在初始位置的周圍，並且我們將每一個粒子的權重(Weight)皆設定為 $\frac{1}{M}$ ，如下式所示：

$$w_t^m = \frac{1}{M} \quad (3-2)$$

其中 w_t^m 代表粒子 s_t^m 的權重

M 為粒子濾波器的粒子數目

接著利用里程計的資訊，來得到機器人在一個取樣周期內的移動距離與朝向角改變的角度， $(dx, dy, d\theta)$ ，將此資訊加上前一時刻的機器人的位置及朝向角， $X_{t-1} = (x_{t-1}, y_{t-1}, \theta_{t-1})$ ，我們可以對機器人目前的位置與朝向角做預測而得到預測結果，以 $\bar{X}_t = (\bar{x}_t, \bar{y}_t, \bar{\theta}_t)$ 表示。

一般來說， $(dx, dy, d\theta)$ 的計算方式是經由機器人輪子驅動馬達的軸編碼器 (Encoder) 回傳的脈波 (Pulse) 數來得到機器人左右兩輪各自移動的距離，再由兩輪移動距離推算出 $(dx, dy, d\theta)$ 。數學式子如下[20]：

$$ds(t) = \frac{1}{2}(ds_r(t) + ds_l(t)) \quad (3-3)$$

$$d\theta(t) = \frac{1}{W}(ds_r(t) - ds_l(t)) \quad (3-4)$$

$$dx(t) = ds(t) \times \cos\left(\frac{\theta_e(t) + \theta_e(t-1)}{2}\right) \quad (3-5)$$

$$dy(t) = ds(t) \times \sin\left(\frac{\theta_e(t) + \theta_e(t-1)}{2}\right) \quad (3-6)$$

其中 $ds_l(t)$ 與 $ds_r(t)$ 分別為左右輪在 t-1 至 t 時間內的移動距離

$ds(t)$ 為機器人在 t-1 至 t 時間內的移動距離

$\theta_e(t)$ 為里程計對機器人朝向的估測結果

W 為機器人的兩輪間距

從(3-5)式與(3-6)式我們可以發現，式子中利用了里程計對機器人朝向的估測結果 $\theta_e(t)$ 來求得 $dx(t)$ 與 $dy(t)$ ，但由前面的討論可知，用里程計來估測機器人的位置及朝向角會有誤差累積的問題，所以使用 $\theta_e(t)$ 來計算 $dx(t)$ 與 $dy(t)$ ，會產生不正確的結果，使得預測的機器人姿態有著更大的誤差。為了改善這個問題，我們將 $\theta_e(t)$ 以所設計的定位系統估測的朝向角 $\theta_{est}(t)$ 來代替，如此可以得到較精確的 $dx(t)$ 與 $dy(t)$ 。不過，可以如此替換的前提是 $\theta_{est}(t)$ 的正確性要比 $\theta_e(t)$ 高，而這點可以從下一節的討論結果得到。所以我們可以將(3-5)式與(3-6)式改寫成：

$$dx(t) = ds(t) \times \cos\left(\frac{\theta_{est}(t) + \theta_{est}(t-1)}{2}\right) \quad (3-7)$$

$$dy(t) = ds(t) \times \sin\left(\frac{\theta_{est}(t) + \theta_{est}(t-1)}{2}\right) \quad (3-8)$$

可以正確的得到 $(dx, dy, d\theta)$ 之後，接著在實際做法上，因為粒子所在的位置就是代表機器人可能存在的位置，所以將每一個粒子所表示的位置都加上 (dx, dy) ，以此來完成預測機器人位置的步驟。而加上 (dx, dy) 後的粒子分佈，即用來代表預測的機器人位置 (\bar{x}_t, \bar{y}_t) 。綜合以上，這個步驟可以下式來表示：

$$s_t^m \sim p(x_t | u_t, s_{t-1}^m), \quad m=1 \sim M \quad (3-9)$$

其中 u_t 在此是代表里程計的資訊

意義為讓所有粒子的分佈符合機率函數 $p(x_t | u_t, s_{t-1}^m)$ ，而 $p(x_t | u_t, s_{t-1}^m)$ 表示在已知里程計資訊與前一時刻粒子分佈的情況下，機器人位置的機率分佈。因此，將每一個粒子加上 (dx, dy) 即可達成(3-9)式。

3.2.2 更新(Update)

在這個階段中，我們使用預測階段得到的粒子組(Particle Set)與雷射掃描儀的量測資訊來估測機器人的位置。首先將雷射掃描儀量測的距離資訊以最小平方產生一表示環境的線段組，並以極座標的直線方程式參數 (α, ρ) 來表示線段(詳述於 3.3 節)。其中 ρ 為機器人與線段的垂直距離，另外以 d_{RL} 代表。為了得到機器人的位置參考資訊，我們必須知道由量測資訊產生的環境線段組與內建地圖的對應關係。所以將環境線段組的直線方程式參數由以機器人為中心的區域座標系轉換到以內建地圖原點為中心的全域座標系來表示，轉換公式如下：

$$\alpha_G = \alpha_L + (\theta - \frac{\pi}{2}) \quad (3-10)$$

$$\rho_G = \rho_L + x \cos \alpha_G + y \sin \alpha_G \quad (3-11)$$

其中 (x, y, θ) 為機器人的姿態，而 (α_L, ρ_L) 與 (α_G, ρ_G) 分別為區域座標系與全域座標系的直線方程式參數，如圖 3-3 所示。所以我們可以得知 d_{RL} 是機器人與內建地圖中的那一條線段之間的相對距離，藉此可以得到機器人在環境的位置參考資訊。接著我們就可以對預測階段產生的粒子組做權重給予(Assign weight)與更新(Update)的步驟。

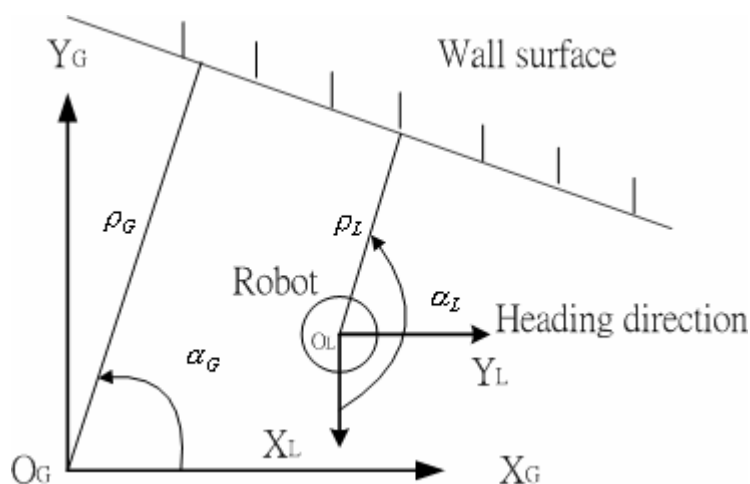


圖 3-3 擷取線段座標轉換關係圖

在粒子濾波器演算法中，每一個粒子的權重是根據下列式子來決定：

$$w_t^m = p(z_t | x_t^m), m=1 \sim M \quad (3-12)$$

其中 z_t 為量測資訊

所以決定了機率分佈函數 $p(z_t | x_t^m)$ 就可以決定權重。而機率函數 $p(z_t | x_t^m)$ 代表的意義為在已知粒子的位置情況下，量測資訊 z_t 的機率分佈情形。

在我們的設計中，量測資訊 z_t 為機器人與環境線段之間的相對距離 d_{RL} 。另外，對每一個粒子再求出其與環境線段之間的相對距離 d_{PL}^m ，因為我們已知環境線段在全域座標系的直線方程式參數 (α_G, ρ_G) 以及每個粒子的位置座標

(x_t^m, y_t^m) ，所以可以經由距離公式求得 d_{PL}^m 。接著對於每一個粒子我們比較 d_{RL} 與 d_{PL}^m ，當 d_{RL} 與 d_{PL}^m 之間的差距愈小代表此粒子的位置與機器人在環境中的真實位置愈符合。相反的，若 d_{RL} 與 d_{PL}^m 之間的差距愈大，則表示此粒子的位置與機器人的真實位置相距愈大。所以我們可以根據 d_{RL} 與 d_{PL}^m 間的差值來給予每個粒子合理的權重，差值愈小則權重愈大，反之，當差值愈大則權重愈小，如下式：

$$w_t^m = p(z_t | x_t^m) \leftarrow \frac{1}{|d_{RL} - d_{PL}^m|} \quad (3-13)$$

接著還必須對每個權重做正規化(Normalize)的動作，即將每個權重 w_t^m 除以所有

權重的總合 $\sum_{m=1}^M w_t^m$ ，藉此得到正規化後的粒子權重。

求得每一個粒子的權重後，須將權重太小的粒子從粒子組中消去，在我們的設計中是將權重小於 $\frac{1}{M}$ 的粒子消去，其中 M 為粒子組包含的粒子數目，而其餘的粒子則持續存在於粒子組中。另外，為了使粒子組的粒子數目在每個循環都保持相同，我們須產生與被消去的粒子數目一樣多的新的粒子。在我們的設計中，是從存留下來的粒子去產生新的粒子。這是因為粒子的位置代表著機器人在環境

中的位置，而存留下來的粒子其位置能夠符合於機器人在環境中的真實位置，所以存留下來的粒子來產生新的粒子，會使得新的粒子組的位置分佈亦集中在真實機器人位置的周圍。而實際做法是先設定一個距離範圍，然後在存留下來粒子的位置的設定之距離範圍內隨機地產生新的粒子。如此新的粒子的位置也會集中在實際機器人的位置周圍。不過所設定的距離範圍不可過大，若過大會造成新的粒子與存留粒子相距太大，使新的粒子組的位置分佈過於分散而與機器人的實際位置不符合。當新的粒子組產生後，再將所有粒子的權重都設為 $\frac{1}{M}$ ，接著新的粒子組便進入下一個循環。以上的處理過程，即是粒子濾波器演算法中的重新取樣 (Resampling) 步驟。

重新取樣後，可得一組新的粒子組，並將粒子組中每個粒子的位置座標取平均值，而得到平均位置的座標即是粒子濾波器對機器人位置的估測結果，如下式所示：



$$x_t = \frac{\sum_m x_t^m}{M} \quad (3-14)$$

$$y_t = \frac{\sum_m y_t^m}{M} \quad (3-15)$$

所以經由(3-14)與(3-15)式，我們可以得到估測的機器人位置座標 (x_t, y_t) 。

3.3 朝向角估測方法

本節將說明如何以最小平方法將由雷射掃描儀得到的環境資訊擷取成線段組合，並將此線段組合結合粒子濾波器預測的機器人姿態 $\overline{X_t}$ 的資訊與已知的環境地圖做比對，最後從比對的結果來估測機器人的朝向角。

3.3.1 線段擷取

從雷射掃描儀我們可以得到以點為基礎的資料，為了將這些資料與環境地圖做線段比對，所以必須將這些用來表示距離的點以線段的方式來描述。所使用的雷射掃描儀的角度掃描範圍(Angular range)為 0 到 180 度，若以 1 度為間隔來量取距離資料，可以得到 181 筆距離資料。每一筆距離資料可以以 (r_i, ϕ_i) 來表示，其中 ϕ_i 代表掃描的角度，而 r_i 則代表對應於 ϕ_i 角度的距離值。

接著使用最小平方誤差法(Least Square Error Method)將 181 筆距離資料 (r_i, ϕ_i) 以線段來表示。一般的直線方程式如下式所示：

$$y = ax + b \quad (3-16)$$

其中 a 代表直線的斜率， b 代表直線與 Y 軸的截距，不過以這種方程式來表示線段會遇到斜率為無窮大的情況，所以我們改用極座標的線段表示方式：

$$x \cos \alpha + y \sin \alpha = \rho \quad (3-17)$$

其中 ρ 代表 XY 平面原點到直線的垂直距離，而 α 則是垂直距離 ρ 代表的直線與 X 軸間的夾角。在我們的應用當中， ρ 代表機器人與牆面之間的垂直距離，而 α 則是 ρ 代表的直線與機器人 0 度方向的夾角，如圖 3-4 所示。

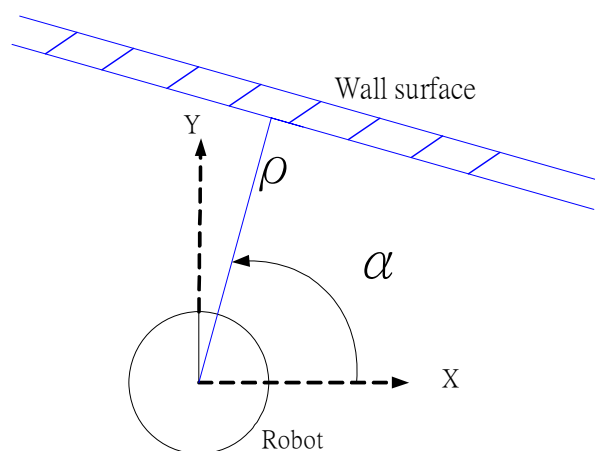


圖 3-4 線段方程式參數定義示意圖

另外，資料點 (r_i, ϕ_i) 到直線 (ρ, α) 的距離 d_i 可由下式得到：

$$d_i = r_i \cos(\phi_i - \alpha) - \rho \quad (3-18)$$

接著最小平方法即是以使所有 d_i 平方的總和為最小值的方式來求得最適合所有資料點的直線參數 (ρ, α) ，若有 n 個資料點，則平方總和 S 可由下式求得：

$$s = \sum_{i=0}^{n-1} (r_i \cos(\phi_i - \alpha) - \rho)^2 \quad (3-19)$$

然後將平方和 S 分別對 ρ 及 α 作偏微分：

$$\frac{\partial \sum_{i=0}^{n-1} (r_i \cos(\phi_i - \alpha) - \rho)^2}{\partial \rho} = 0 \quad (3-20)$$

$$\frac{\partial \sum_{i=0}^{n-1} (r_i \cos(\phi_i - \alpha) - \rho)^2}{\partial \alpha} = 0 \quad (3-21)$$

整理(3-20)式與(3-21)式，可以得到：

$$\rho = \frac{1}{n} \sum_{i=0}^{n-1} r_i \cos(\phi_i - \alpha) \quad (3-22)$$

$$\tan(2\alpha) = \frac{n \sum_{i=0}^{n-1} r_i^2 \sin 2\phi_i - \sum_{i=0}^{n-1} \sum_{j=0}^{n-1} r_i r_j \sin(\phi_i + \phi_j)}{n \sum_{i=0}^{n-1} r_i^2 \cos 2\phi_i - \sum_{i=0}^{n-1} \sum_{j=0}^{n-1} r_i r_j \cos(\phi_i + \phi_j)} \quad (3-23)$$

所以最佳解 (ρ, α) 可以由(3-20)式及(3-21)式求得。

可以求得資料點的最佳線段後，接下來就是要將由雷射掃描儀讀入的 181 筆資料點擬合成線段，但因為機器人所在的環境中通常不會只有一面牆，讀入的資料點通常會被擬合成多條線段來代表他們所描述的環境特徵(牆面)，稱為線段組。所以我們要將所有的資料點做分組使得符合同一條線段的資料點分成同一組後再進行線段擷取，步驟如下[13]：

1. 首先將前兩筆資料點 D_0 及 D_1 (0 度與 1 度) 以最小平方誤差法求得線段 L_1 ,
 (ρ_1, α_1) 。

2. 接著判斷第三筆資料點 D_2 是否也屬於前一步驟求得的線段 L_1 , 判斷方式是去計算資料點 D_2 到直線 L_1 的距離 d , 如圖 3-5 所示, 若得到的距離值小於一預先設定的臨界值 dis_{th} , 則我們可以將 D_2 包含於 L_1 內, 並回到步驟 1 將 $D_0 \sim D_2$ 以最小平方誤差法求得新的線段。但若資料點 D_2 到直線 L_1 的距離 d 大於 dis_{th} , 表示 D_2 不包含於 L_1 內, 則 L_1 即為擷取出的線段, 並將 D_2 與下一資料點 D_3 代入步驟 1, 進行下一條線段的擷取。

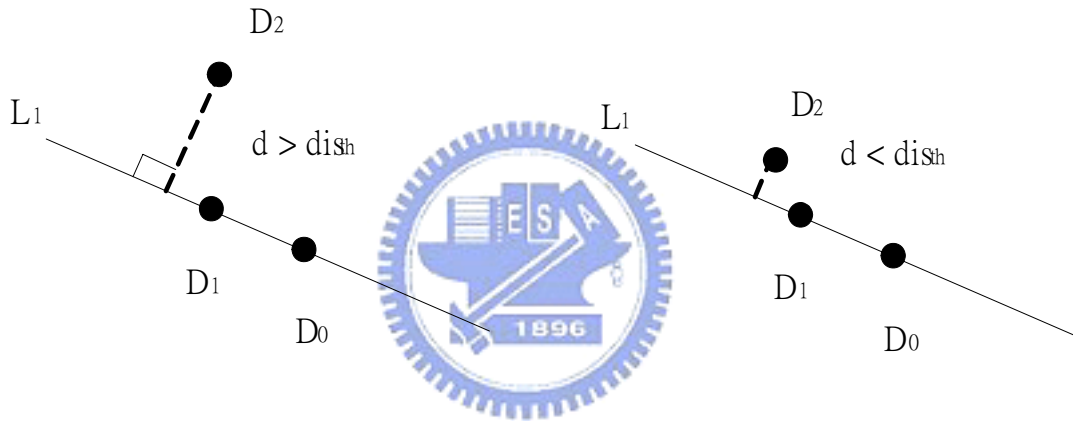


圖 3-5 判斷資料點是否包含於線段之示意圖

如此重覆的進行步驟 1、2 直到最後一筆資料點結束, 我們可以得到若干線段的線段組:

$$L = \{L_i \mid i = 1 \dots N\} \quad (3-24)$$

其中 $L_i = (\alpha_i, \rho_i) \quad (3-25)$

而線段擷取的流程如圖 3-6 所示。

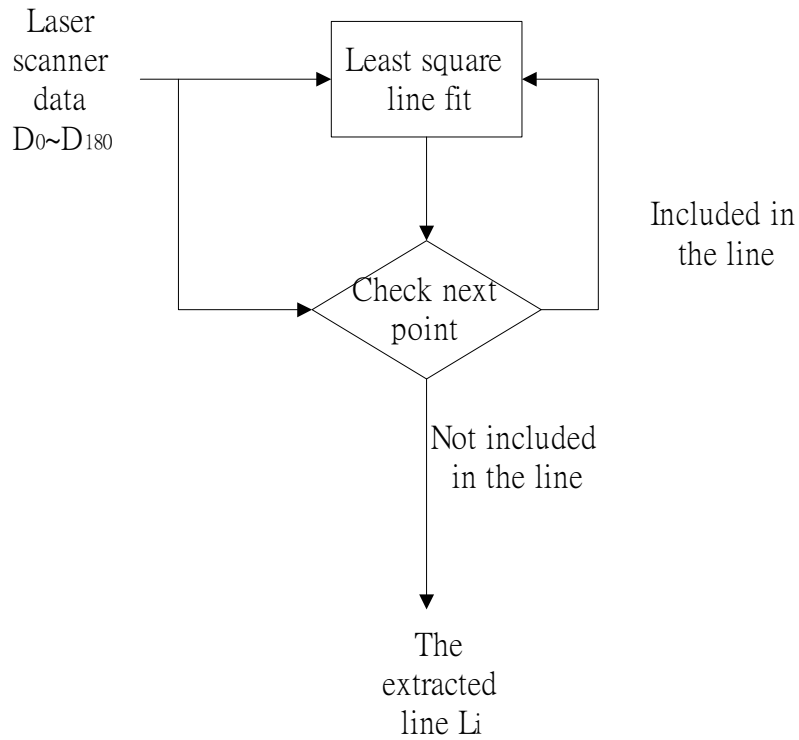


圖 3-6 線段擷取流程圖

假設機器人在如圖 3-7 的環境裡，機器人的座標為(100,200)朝向角為負 90 度(X 軸方向為 0 度)。使用上面陳述的線段擷取方法，我們可以得到三條線段如圖中所示的 Line1、Line2 及 Line3，其線段參數(ρ, α)如表 3-1 所列，其中第三欄為每條線段所包含的資料點數目。

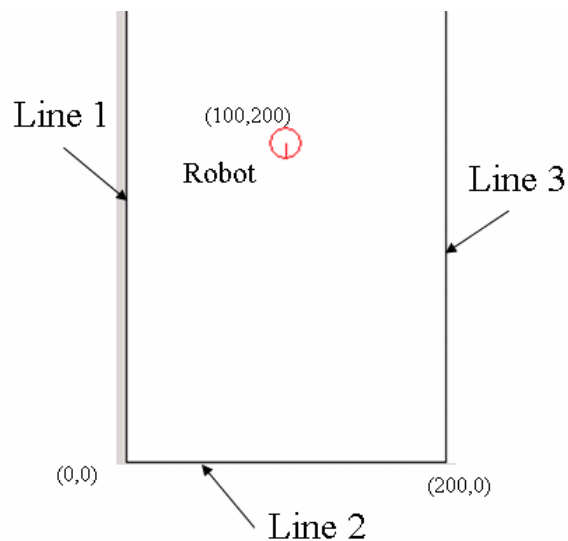


圖 3-7 機器人在環境中擷取線段

表 3-1 機器人在圖 3-7 環境中擷取的線段資料

	ρ	α (deg)	Data number
Line 1	99.66	0.16	65
Line 2	201.51	89.89	53
Line 3	101.53	179.9	63

3.3.2 線段比對

線段比對的目的是要求出機器人的朝向角。因為我們已知以線段表示的環境地圖以及粒子濾波器預測的機器人姿態 $\bar{X}_t = (\bar{x}_t, \bar{y}_t, \bar{\theta}_t)$ 的資訊，所以可以經由模擬得到機器人在環境地圖中的 (\bar{x}_t, \bar{y}_t) 位置而朝向角為 $\bar{\theta}_t$ 時雷射掃描儀掃描的距離資料，並經由線段擷取可得到一組線段組：

$$L^M = \{L_k^M \mid k = 1 \dots N_M\} \quad (3-26)$$

其中 $L_k^M = (\alpha_k^M, \rho_k^M) \quad (3-27)$

另外，將機器人上的雷射掃描儀量到的實際環境距離資料以線段擷取步驟處理，同樣地我們可以得到另一組線段組：

$$L^L = \{L_i^L \mid i = 1 \dots N_L\} \quad (3-28)$$

其中 $L_i^L = (\alpha_i^L, \rho_i^L) \quad (3-29)$

接著便可以將 L^L 與 L^M 兩線段組進行比對，而比對的方式為分別比較 L^L 中每一條

線段 L_i^L 與 L^M 中的每一條線段 L_k^M 的直線方程式參數，當相互比較的兩條線段的參數符合下面兩式時，則將這兩條線段視為相互符合。

$$|\alpha_i^L - \alpha_k^M| \leq \alpha_{th} \quad (3-30)$$

$$|\rho_i^L - \rho_k^M| \leq \rho_{th} \quad (3-31)$$

其中 α_{th} 與 ρ_{th} 為預先設定的臨界值。

當 L^L 與 L^M 中各別的每一條線段都相互比較過之後，我們可以得知 L^L 與 L^M 兩線段組的互相符合程度，並且由此我們可以判定預測的機器人朝向角 $\bar{\theta}_t$ 與機器人實際的朝向角是否符合。通常 $\bar{\theta}_t$ 與機器人實際朝向角之間的誤差不大，這是因為 $\bar{\theta}_t$ 是由前一時刻定位系統估測的機器人朝向角 θ_{t-1} 加上里程計的資訊得來，所以我們認定機器人實際的朝向角會落在以 $\bar{\theta}_t$ 為中心向正負方向展開的角度範圍內。

在我們的設計中，是取以 $\bar{\theta}_t$ 角度為中心正負 15 度的範圍。因此我們可以將機器人的朝向角設定成此角度範圍內的角度，再經由模擬可得到對應每個朝向角角度的線段組 L^M ，共會產生 31 組。最後，我們將這 31 組線段組分別與由實際雷射掃描的量測資料點產生的線段組 L^L 做比對，接著從比對的結果我們可以得到 31 組線段組中那一組與 L^L 最符合，而最符合的線段組所對應的朝向角即是估測的機器人朝向角 θ_t 。

不論是位置估測方法或朝向角估測方法都需要用到環境中牆面的資訊，因此環境中牆面的多寡將會影響定位系統所估測的機器人姿態。當環境中只有一面牆時，因為朝向角估測方法是去比對代表牆面的線段參數 (α, ρ) 來估測朝向角，故在只有一面牆的情況下仍可估測出朝向角。另一方面，在位置估測方法上，是使

用機器人與牆面的垂直距離作為量測資訊來進行位置估測，而機器人位置 (x, y) 是二維資料，需要環境中至少有兩面牆來提供量測資訊，位置估測方法才能有效的估測出機器人的位置，所以環境中只有一面牆是不夠的。因此機器人必需處於有兩面或兩面以上牆壁的室內環境中，所設計的機器人定位系統才能正確地估測機器人的姿態。



第四章 模擬與實驗結果

本章的目的是為了驗證第三章所設計的自走式機器人定位系統。先介紹用來做為實驗平台的機器人系統與雷射掃描儀，接著呈現模擬與實驗結果。在實驗部分，第一個實驗讓機器人繞行環境，藉以驗證機器人在繞行的過程中是否能夠持續地定位。第二個實驗則增長了繞行的環境，以此測試機器人在更大的環境中是否同樣具有定位的能力。第三個實驗則將繞行環境改變成 L 型走廊來驗證所設計的定位系統。

4.1 機器人定位系統硬體架構

4.1.1 機器人系統

本論文以實驗室自行開發的機器人做為實驗平台，如圖 4-1 所示。其系統核心為工業電腦，負責處理由各個感測器傳來的資訊並估測機器人的姿態，以及決定機器人的運動控制命令並透過 RS232 介面傳送到 DSP 馬達控制系統。在感測器方面，雷射掃儀是透過 RS422 介面將量測資訊傳送到工業電腦，同時亦接收由工業電腦傳來的指令。圖 4-2 為機器人系統控制架構。

方塊說明：

1. 雷射掃描儀

2. 工業電腦

3. DSP 馬達控制卡

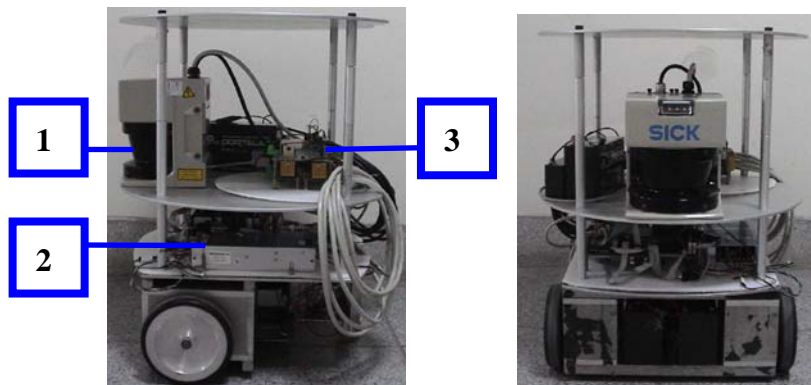


圖 4-1 機器人硬體裝置架構圖

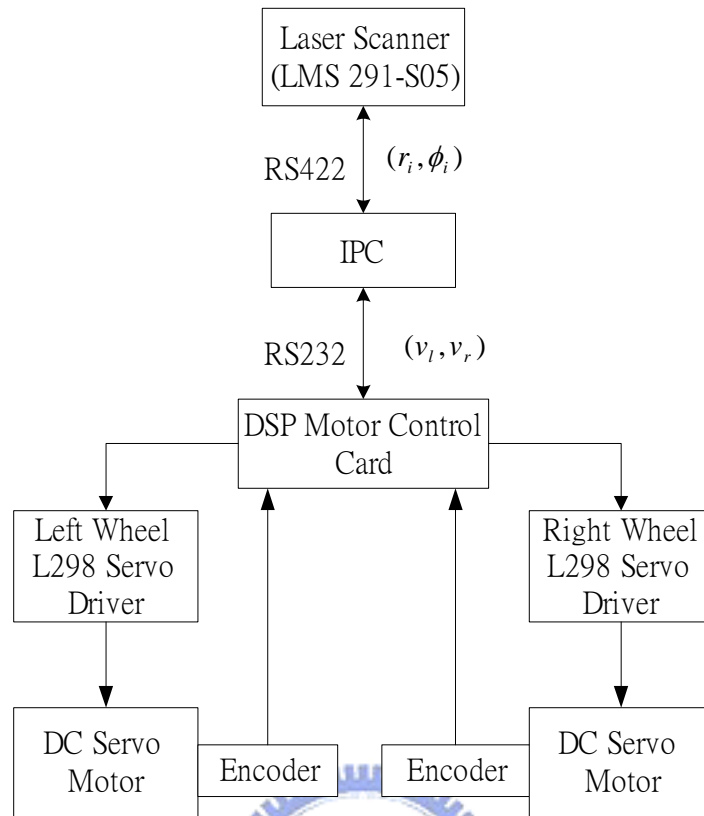


圖 4-2 機器人系統控制架構圖

在機器人之馬達控制系統部分採用實驗室學長所開發完成之 DSP 馬達控制卡，再加上兩塊 L298 驅動電路，來做為機器人的兩輪移動控制(速度回授控制)，其與工業電腦端之訊息傳遞透過 RS232 介面來實現，並且資料傳輸速率為 19200bps。另外，兩個馬達各附有一組軸編碼器(Encoder)，經由軸編碼器的回傳資料可以得知兩輪的移動情形。

4.1.2 雷射掃描儀

本論文所使用的雷射掃描儀為 SICK 公司的 LMS291-S05，如圖 4-3，其規格如下：

- 最大量測距離：80 公尺

- 角度量測範圍：0 ~ 180 度
- 角度解析度：0.25 / 0.5 / 1 度
- 量測距離解析度：10 公釐
- 資料傳輸速率：9.6 / 19.2 / 38.4 / 500 k baud

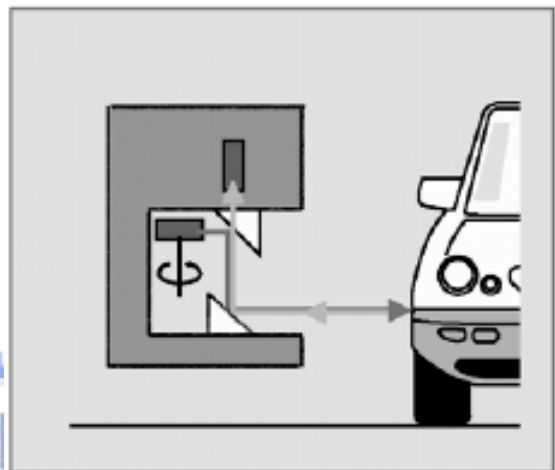


圖 4-3 雷射掃描儀

圖 4-4 雷射掃描儀工作原理[11]

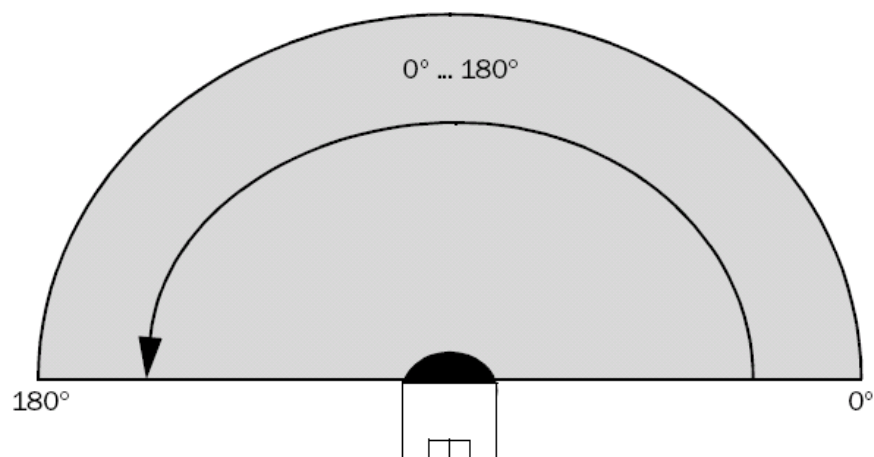


圖 4-5 雷射掃描儀角度掃描範圍[11]

在我們的使用上是將傳輸速率設成 38400，並使用 1 公分的距離解析度，0.5 度的角度解析度，與 180 度的角度掃描範圍(如圖 4-5)，所以在每一次掃描之後雷

射掃描儀可得到 0~180 度的 361 筆距離資料。

在雷射掃描儀的內部結構中，主要包含三個部分，分別是不斷旋轉的鏡面、光學折射裝置與接收反射回來雷射光的接收器，如圖 4-4 所示。其工作原理為當雷射光從雷射掃描儀發射出去後，經由光學折射裝置射向前方，待雷射光碰到物體反射回雷射掃描儀，再經由折射裝置將反射回來的雷射光送到雷射光的接收器，雷射掃描器便利用此雷射光往返的時間(time-of-flight)去計算出雷射掃描器與障礙物間的距離。而不斷旋轉的鏡面則將雷射光打至 0~180 度之間不同的角度，如此便可計算出每個角度與環境障礙物之間的距離。

另外，我們可以對 LMS 291-S05 雷射掃描儀下達不同的指令參數來對掃描儀做不同的要求，但當我們給予的指令不是雷射掃描儀所能明白的格式，這時掃描儀的輸出將不會有變化，也就是不會有訊號產生，而當我們所輸入的命令是雷射掃描所能明白時，它會先輸出一個確認訊號(ACK, 06H)表示雷射掃描器已經收到命令，緊接在確認性號後的則是對應所下達命令的反應(Response)。

在 LMS 291-S05 雷射掃描儀的指令當中最常用到的是要求回傳量測值指令(Requesting measured value, 30H)，其指令格式如下表所示：

表 4-1 要求回傳量測值指令格式[11]

Description	STX	Address	Length		Command	Data	Checksum	
Byte position	1	2	3	4	5	6	7	8
Hex. value	02	00	02	00	30	01	31	18

其中 STX 代表同步檢查碼，固定為 02H；Address 代表在使用多個雷射掃描儀的系統中要對第幾個雷射掃描儀下命令(從 0 開始)；Length 代表粗黑框框中 Command 與 Data 的長度，第 3 個 Byte 表示為低位元，第 4 個 Byte 表示為高位元，在本例中 Command 與 Data 的總長度為兩個 Byte，因此第三個 Byte 為 02H，第四個 Byte 為 00H；Command 代表所欲送出命令的代碼，30H 是代表要求雷射

掃描儀回傳量測值；Data 中的 01H 為 Command 30H 的參數，Checksum 表示檢查碼。關於 Checksum 的部份，SICK 公司有其自己產生檢查碼的方式，詳細說明可以參考[11]中第 104~105 頁所附之程式碼。

在送出上列指令後，LMS 會先回傳一個確認訊號 ACK(06H)，然後再回傳如下格式的反應：

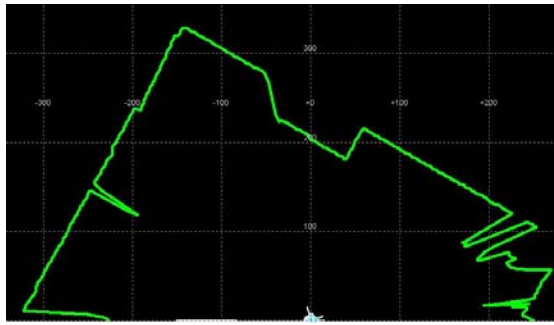
表 4-2 要求回傳量測值命令之反應[11]

Description	STX	Address	Length		Response	Data		Checksum	
						Data	LMS status		
Byte position	1	2	3	4	5	6 to 729	730	731	732
Hex. value	02	80	D6	02	B0	724 bytes	10	15	D4

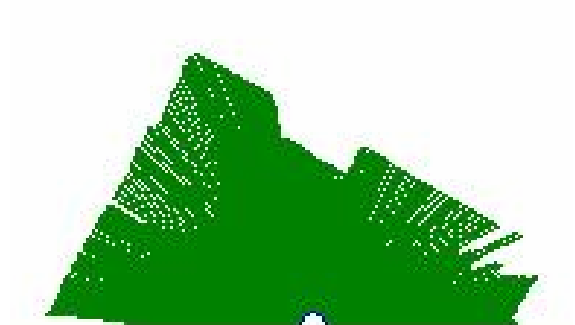
由上表可知在要求回傳量測值命令反應中，STX 不變；Address 變為 80H；Length 代表 Response 與 Data 的總長度；而 Response 裡的命令代號通常為所下達的命令代號在加上 80H，所以 Response B0H 為 30H 增加 80H 後的結果；Data 欄內的資訊即是我們所欲得的量測資料，稍後將進一步說明；LMS status 則代表在反應裏 Data 的結尾，以 10H 表示(固定為 10H)；而 Checksum 同樣表示檢查碼。

在 Data 欄中(第 6 到第 729 個 Byte)共有 724 個 Byte，這是由於在原始設定下雷射掃描儀的解析度預設為 0.5 度，因此在 0~180 度的範圍內將共有 361 組量測資料，而每組量測資料都有兩個 Byte，因此便有 722 個 Byte，再加上一開始的 2 個 Byte(在此例為 9H 與 61H，即十進制的 361，代表隨後有 361 組量測資訊)便成為表 4-2 中 Data 欄內的 724 個 Byte。

接著使用上述的要求回傳量測值命令下達給雷射掃描儀來得到量測距離資訊。為了要測試得到的量測距離資料是否正確，所以我們將得到的距離資料與經由 SICK 公司提供的軟體(LMS-LMI400 user software V 5.11)取得的距離資料做比對。經由這兩種方式得到同一環境的距離量訊資訊如下圖所示：



(a)



(b)

圖 4-6 雷射掃描儀量測結果(a)經由 LMS-LMI400 user software V 5.11

(b)控制程式實際測試值

圖 4-6 中的(a)(b)兩圖所得到的環境距離量測結果是相同的，所以可以知道經由直接給予要求量測值命令得到的距離資訊是正確的。

4.2 模擬結果



為了驗證所設計的定位系統是否可行，我們模擬機器人在長 16 公尺，寬 4 公尺的長方形環境中繞行，設定的繞行軌跡如圖 4-7 所示。

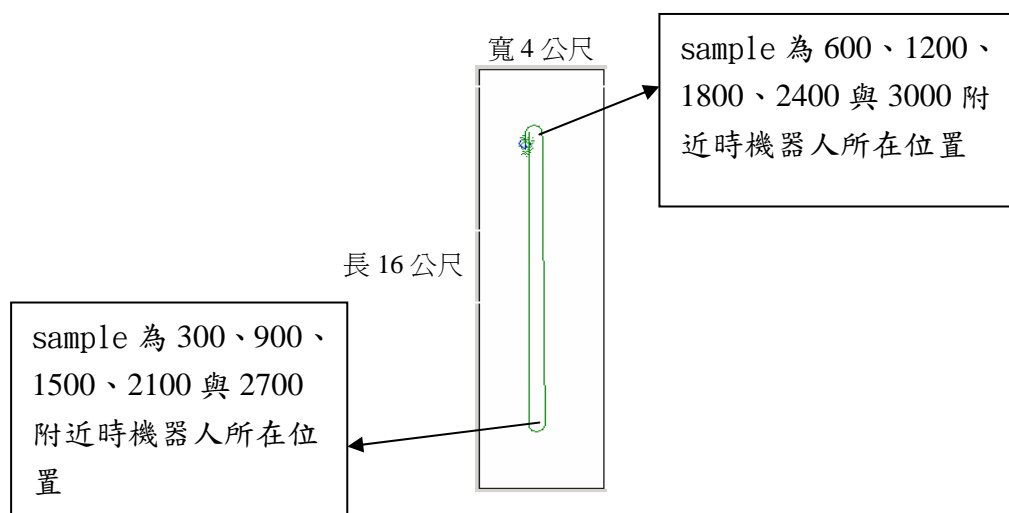
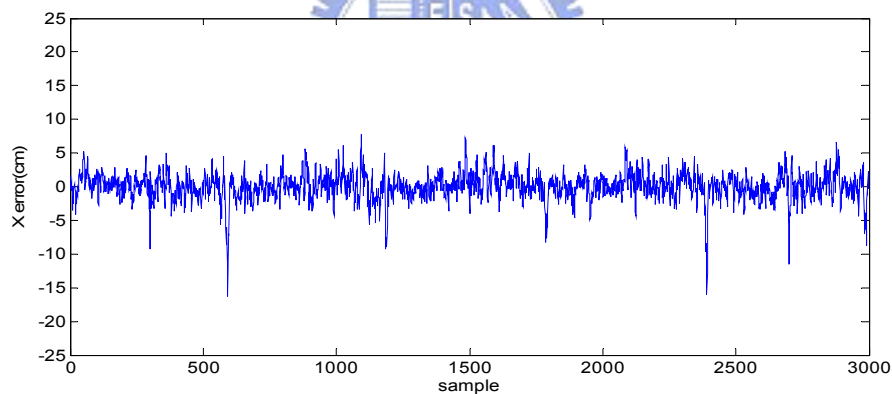


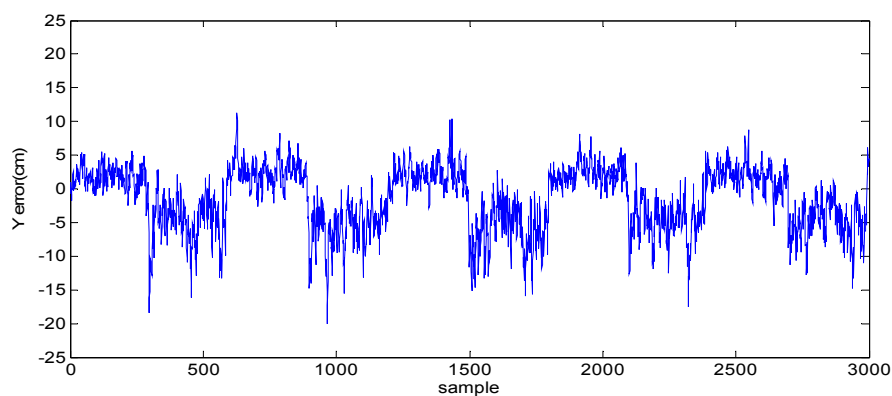
圖 4-7 模擬環境與機器人繞行軌跡

粒子濾波器所使用的粒子數目為 100 個。一開始機器人已知初始姿態，所以將粒子散佈於以機器人位置為中心的區域內。另外，機器人內建的環境地圖包含了構成長方形環境的四條線段(L1~L4)及其極座標的直線方程式表示參數 (α, ρ) ，在繞行過程中定位系統即是將由程式中模擬的雷射掃描儀產生的環境線段組經由座標轉換與內建地圖的四個線段做對應，藉此來得到機器人在環境中的位置參考資訊。

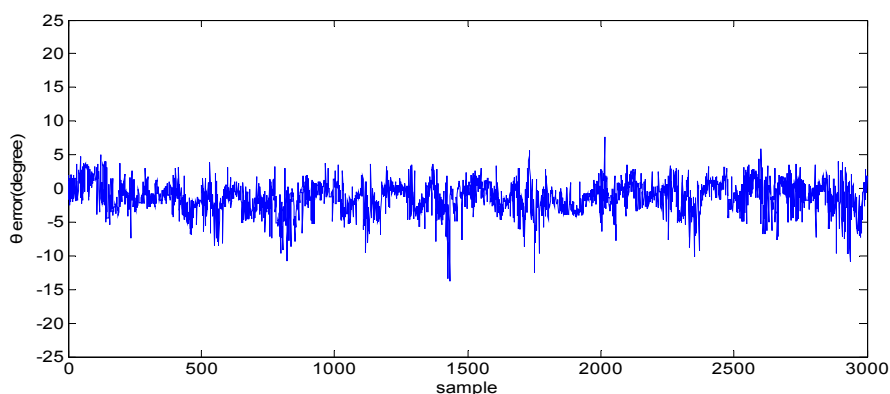
設定機器人以每個 sample 16cm 的速度持續繞行設定的軌跡 5 圈，繞行一圈會經過 600 個 sample，並且分別在機器人的 x 方向與 y 方向加上平均值為 0，標準差為 1.6 公分的高斯雜訊 $N(0, 1.6^2)$ ，以此模擬在實際情況下機器人運動過程中會產生的誤差。模擬結果如圖 4-8 所示，為定位系統對於在繞行過程中機器人姿態 (x, y, θ) 隨時間的估測誤差。



(a)



(b)



(c)

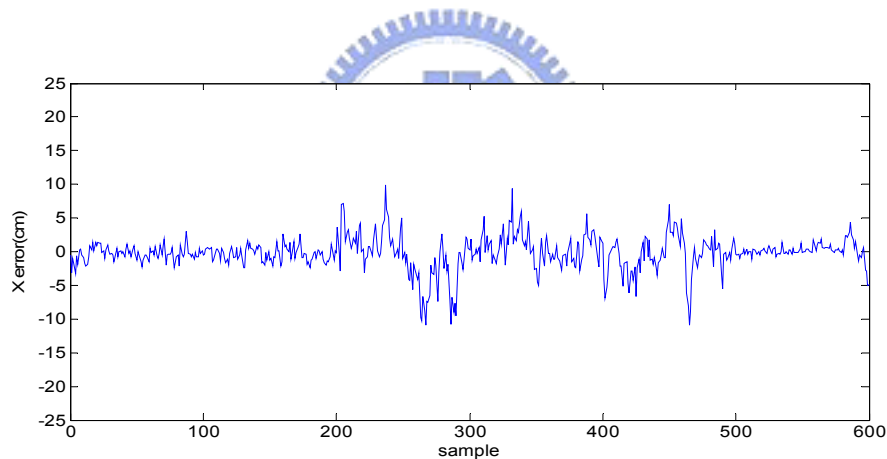
圖 4-8 模擬一之機器人定位系統姿態估測誤差

觀察圖 4-8(a)與 4-8(b)的模擬結果可以發現機器人定位系統所估測的 y 座標誤差比 x 座標的誤差大，這是因為機器人在繞行的過程中大部分是 y 方向的移動，只有在軌跡的上面與下面轉彎的部分機器人才有 x 方向的移動，故造成 y 座標有較大的估測誤差。另外圖 4-8(a)的 x 座標估測誤差中，在 sample 為 300、600、900、1200、1500、1800、2100、2400 與 2700 附近有突然變大的誤差產生，造成的原因是機器人在這些 sample 附近時正在轉彎，其移動的情況從只有 y 方向部分變成增加了 x 方向的移動，使 x 座標的估測誤差突然變大。在朝向角估測誤差部分，定位系統在估測機器人朝向角時需要使用到粒子濾波器估測的機器人位置資訊，所以若位置估測的結果有比較大的誤差產生，會使得朝向角的估測也產生對應的誤差。整個模擬的姿態估測平均的絕對誤差如表 4-3 所示，由平均誤差的大小可以發現定位系統估測的機器人姿態與實際的情況相當接近，由此我們可以驗證所計設的機器人定位系統是有效的。

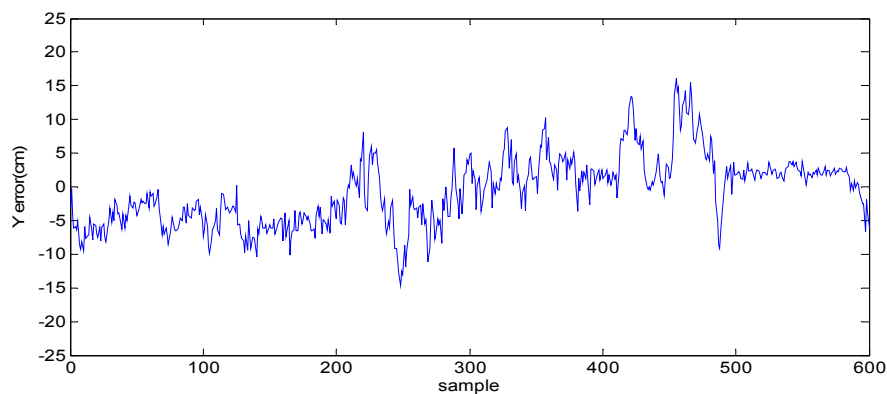
表 4-3 模擬一之機器人姿態估測絕對誤差

	X(cm)	Y(cm)	Θ (deg)
Mean	1.17	3.27	2.06
Max	16.3	19.98	13.84

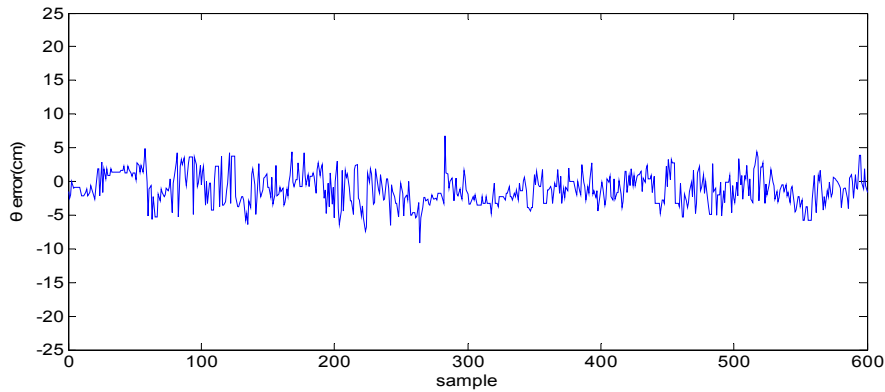
接著進行第二個模擬目的在於討論所加入的誤差雜訊對於定位系統的影響，故在機器人繞行環境時只在某一時段區間內在機器人的 x 座標與 y 座標加入高斯雜訊，並觀察定位系統所估測的機器人姿態誤差的結果。在模擬的過程中讓機器人同樣繞行圖 4-7 的軌跡，並且只在 sample 為 200 到 500 的時段內在機器人的 x 座標與 y 座標加入平均為 0，標準差為 3.2 公分的高斯雜訊 $N(0, 3.2^2)$ 。模擬結果如圖 4-9 所示。從模擬結果可以發現 x 座標與 y 座標的估測誤差在 sample 為 200 至 500 之間有變大的情形，其中 x 座標的部分較為明顯。不過誤差的大小仍小於一定的範圍內。而圖 4-9(a) 中 x 座標的估測誤差在 sample 為 300 與 600 處有突然高起的情形，造成的原因與模擬一相同，是機器人運動方向變成 x 方向所產生的。這個模擬中未加雜訊部分與加上雜訊部分的統計絕對誤差分別如表 4-4 與表 4-5 所示。



(a)



(b)



(c)

圖 4-9 模擬二之機器人定位系統姿態估測誤差

表 4-4 模擬二未加雜訊部分之機器人姿態估測絕對誤差

	X(cm)	Y(cm)	Θ (deg)
Mean	0.75	3.08	1.99
Max	5.04	10.36	6.36

表 4-5 模擬二加雜訊部分之機器人姿態估測絕對誤差

	X(cm)	Y(cm)	Θ (deg)
Mean	2.25	4.28	2.26
Max	10.9	16.2	9.1

最後進行第三個模擬，這個模擬中除了在機器人的 x 座標與 y 座標加入與模擬一相同的高斯雜訊之外，還在模擬程式中的雷射掃描儀部分加入量測雜訊，以模擬實際的雷射掃描儀的量測誤差。所加入的量測雜訊是平均為 0，標準差為 4 公分的高斯雜訊 $N(0, 4^2)$ 。模擬過程中讓機器人繞行圖 4-7 的軌跡 5 圈，完成一圈的繞行需經過 600 個 sample。模擬結果如圖 4-10 所示，為定位系統對於機器人姿態的估測誤差，而表 4-6 為統計的絕對誤差，與模擬一的結果比較，可發現模擬三的估測誤差比較大，但仍小於一定的範圍內，由此我們可以驗證在增加了

量測雜訊後所設計的定位系統仍是有效的。

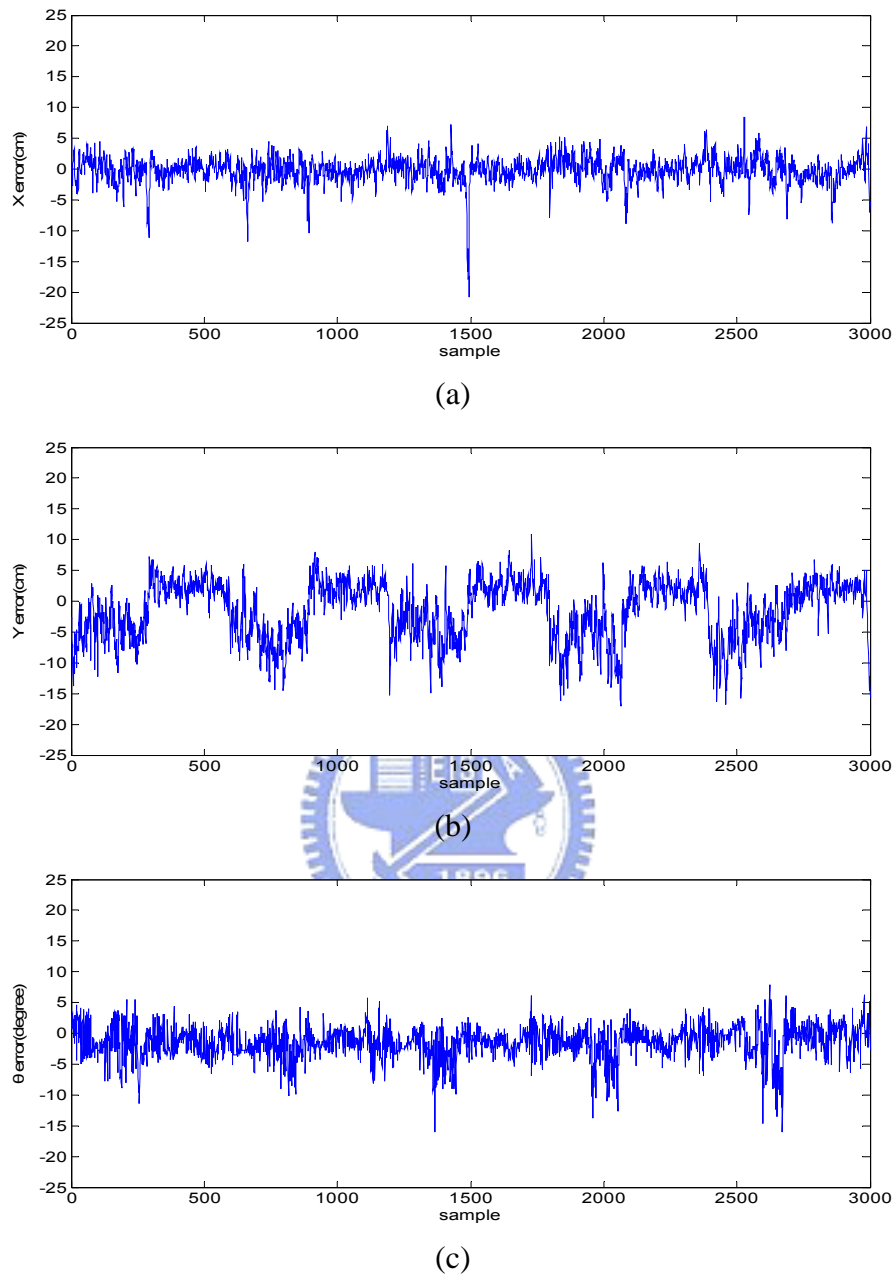


圖 4-10 模擬三之機器人定位系統姿態估測誤差

表 4-6 模擬三之機器人姿態估測絕對誤差

	X(cm)	Y(cm)	Θ (deg)
Mean	1.51	3.92	2.27
Max	20.78	17.04	16.07

4.3 實驗結果

4.3.1 繞行實驗(一)

在這個實驗中，我們讓機器人在長方形的走廊環境中繞行，如圖 4-11，環境的長為 7.2 公尺，寬為 2.58 公尺。並在內建的環境地圖中將構成走廊環境的四條線段以 L1~L4 表示，其極座標的直線方程式參數(α, ρ)分別為，L1($90^\circ, 0$)、L2($180^\circ, 138$)、L3($90^\circ, 720$)及 L4($0^\circ, 120$)。機器人繞行的路徑設定成長 3.6 公尺，寬 1.2 公尺的軌跡，其四個頂點座標為(40,200)、(40,560)、(-80,560)及(-80,200)，機器人的起始位置座標為(40,200)朝向角為 90 度，並讓機器人以逆時針方向繞行。

機器人的繞行控制方式是將設定的軌跡的四個頂點當成四個目標點，讓機器人依序朝向各個目標點前進，順序為(40,200)→(40,560)→(-80,560)→(-80,200)。而機器人的運動控制是根據 l_{Goal} 與 θ_{Goal} 這兩個參數來決定的，其中 l_{Goal} 是機器人與目標點的相對距離， θ_{Goal} 是機器人正前方延長線與目標點的夾角(逆時針為正，

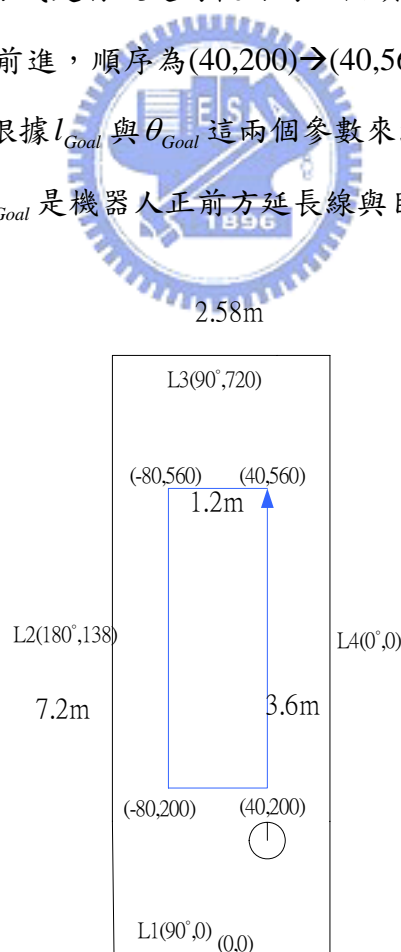


圖 4-11 繞行實驗(一)之環境示意圖

順時針為負)，如圖 4-12 所示，這兩個參數可由定位系統估測的機器人姿態與已知的目標點座標經運算得到。

機器人在運動的過程中會不斷地計算 l_{Goal} 與 θ_{Goal} ，並判斷 l_{Goal} 與 θ_{Goal} 的數值大小作為運動控制的依據，當 θ_{Goal} 值超出一預定的閾值 θ_{Goal_th} 所構成的範圍之外，即 $\theta_{Goal} > \theta_{Goal_th}$ 或 $\theta_{Goal} < -\theta_{Goal_th}$ ，則設定機器人的線速度 v 為零並改變角速度值 w 使得 θ_{Goal} 值落在設定範圍內，即 $-\theta_{Goal_th} < \theta_{Goal} < \theta_{Goal_th}$ ；當 θ_{Goal} 值被修正回設定範圍內時，就將 w 設為零， v 設為原來給定的線速度值。此外，當 l_{Goal} 值小於一預定的閾值 l_{Goal_th} ，即 $l_{Goal} < l_{Goal_th}$ ，就改變目標點的座標，使機器人能朝下一個目標點前進。

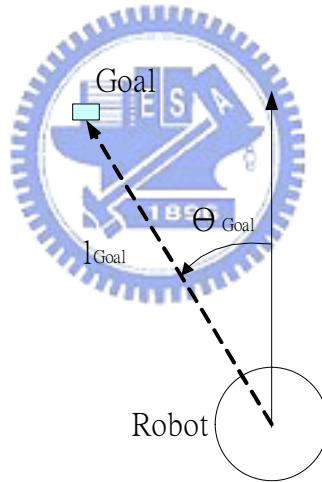


圖 4-12 機器人與目標點相對位置示意圖

所以一開始機器人從(40,200)出發，這時機器人的目標點被設定為(40,560)，故機器人會根據上述的運動控制方式朝(40, 560)的方向前進。當 $l_{Goal} < l_{Goal_th}$ 時，則將機器人的目標點改變成(-80,560)，接著機器人會改變前進方向朝(-80,560)前進。實驗中便是以這樣的方式讓機器人持續地繞行四個目標點所構成的長方形軌跡。

機器人要能確切地接近各個目標點，必須依靠其本身的自我定位系統所估測

出來的機器人姿態是否與實際環境中機器人的姿態相符合，若兩者符合的話，機器人所計算出的 l_{Goal} 與 θ_{Goal} 的數值才是有效的，並藉此來控制機器人確切地朝各個目標點前進。

實驗中設定 $v=10\text{cm/sec}$ ， $w=0.15\text{ rad/sec}$ ， $l_{Goal_th}=10\text{cm}$ ， $\theta_{Goal_th}=10^\circ$ 。一開始機器人的初始位置為右下角的(40,200)，朝向角為 90 度，以 10cm/sec 的速度朝右上角的目標點(40,560)前進，透過定位系統所估測的姿態使機器人知到目前的位置與方向並由此決定朝目標點移動的方向。實驗過程中機器人共繞了長方形軌跡五圈，移動過程如圖 4-13，其中第一列(a)~(d)為機器人依序繞過長方形軌跡的四個頂點的情形，而第二列(e)~(h)到第五列(q)~(t)則分別是機器人在繞行第二到第五圈時經過長方形軌跡的四個頂點的情形。從圖中可以得知機器人一直繞行到了第五圈，仍然移動經過四個設定的目標點，而其行走的路徑也仍在長方形軌跡上。

所以由實驗結果可以知道在機器人的移動過程當中，定位系統確實將雷射掃描儀與里程計的量測資訊讀入，經由粒子濾波器對機器人的位置做估測，以及經由線段擷取與比對來得到機器人的朝向角，最後估測出與實際相符合的機器人姿態。

圖 4-14 為實驗結果，在圖 4-14 (a)中虛線部分為 Dead-reckoning 所估測的機器人軌跡，可以發現從第三圈開始估測軌跡有明顯偏移設定軌跡的情形，其偏移設定軌跡的最大值在 x 座標為 65.2 公分， y 座標為 34.5 公分，所以若機器人依據 Dead-reckoning 所估測的機器人姿態來繞行設軌跡，會因繞行圈數增加而逐漸偏離設定軌跡。實線部分為定位系統所估測出的軌跡，其結果大致符合所設定的長方形軌跡，並不會因繞行圈數增加而逐漸偏離設定的軌跡。另外，定位系統估測的機器人軌跡與設定軌跡之間的絕對誤差平均值為 x 座標：7.84 公分； y 座標：8.09 公分；朝向角：7.33 度。軌跡誤差的原因除了定位系統的姿態估測誤差外，還包含了運動控制的部分：即當機器人朝目標點前進時，可能其運動軌跡已

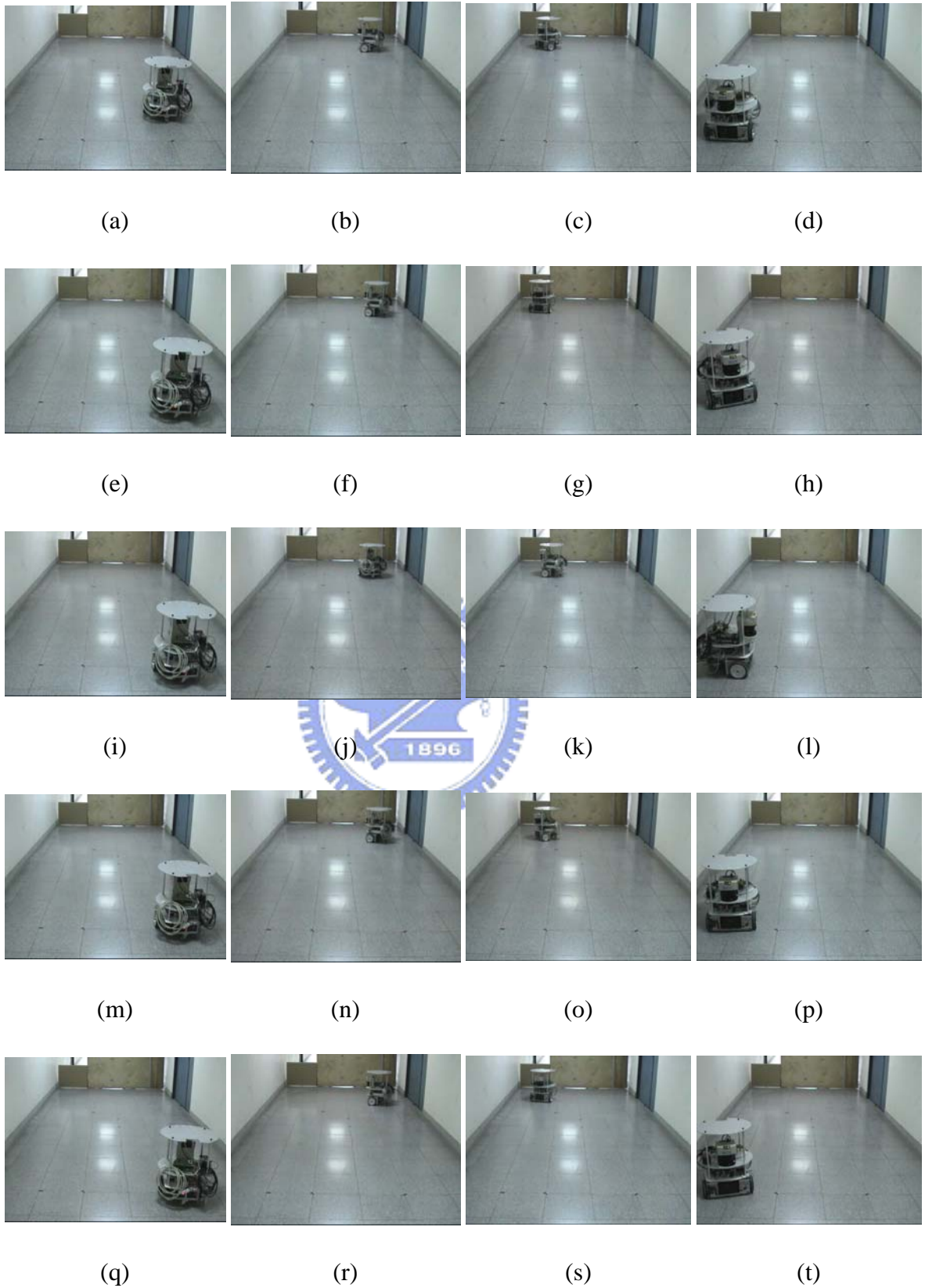
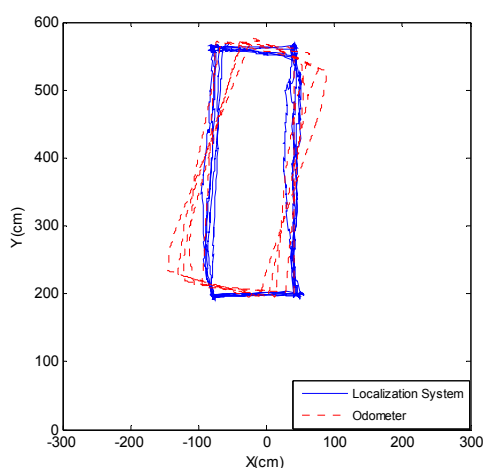
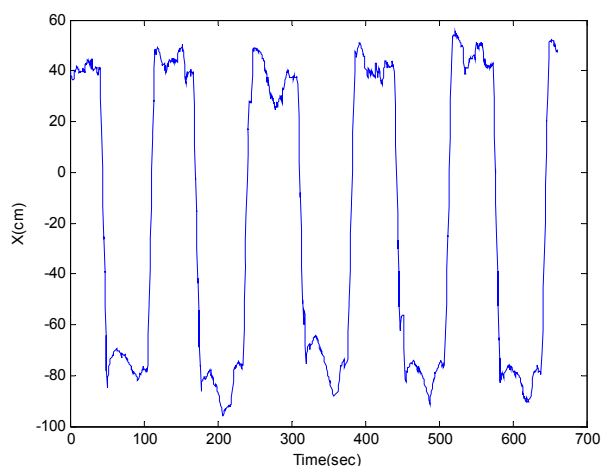


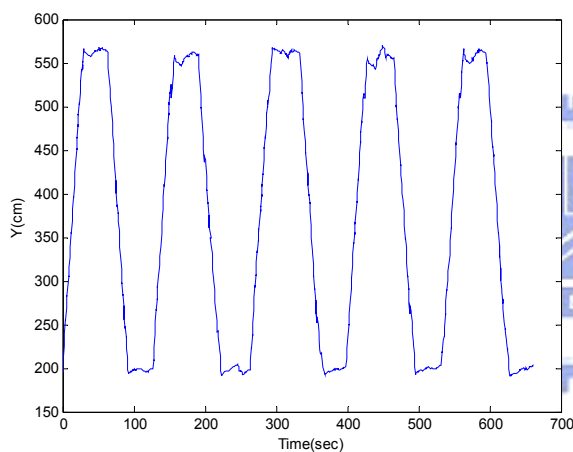
圖 4-13 繞行實驗(一)之實驗過程



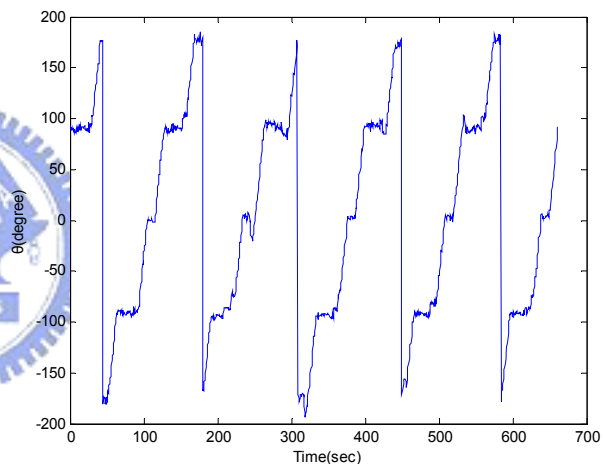
(a)



(b)



(c)



(d)

圖 4-14 繞行實驗(一)之結果：(a)機器人移動軌跡 (b)X 座標與時間關係圖

(c)Y 座標與時間關係圖 (d) θ 與時間關係圖

偏離了設定軌跡，但因 θ_{Goal} 仍在預設範圍內， $-\theta_{Goal_th} < \theta_{Goal} < \theta_{Goal_th}$ ，所以機器人只會依目前的朝向角直線前進，直到 θ_{Goal} 超出了預設範圍，機器人才會修正其朝向角，使其運動路徑接近設定軌跡，因此這是造成估測軌跡與設定軌跡之間誤差的另一因素。圖 4-14 (b)(c)(d) 為機器人姿態 (x, y, θ) 與時間關係圖，在設定軌跡附近(x 為 40 與 -80， y 為 200 與 560， θ 為 0、 ± 90 與 ± 180) 的值有些會有明顯的

凸起或凹下部分，這是因為機器人在朝目標點前進時會有誤差產生，當誤差過大時則修正機器人的朝向角使機器人再朝向目標點前進，造成圖中的凸起或凹下部分。

4.3.2 繞行實驗(二)

在這個實驗中，我們增加走廊環境的長度為 10 公尺，如圖 4-15，同樣在環境地圖中將構成走廊環境的四條線段以 L1~L4 表示，其直線方程式參數(α, ρ)分別為，L1($90^\circ, 0$)、L2($180^\circ, 138$)、L3($90^\circ, 1000$)及 L4($0^\circ, 120$)。機器人繞行的路徑設定成長 6 公尺，寬 1.2 公尺的長方形軌跡，其四個頂點座標為(40,200)、(40,800)、(-80,800)及(-80,200)，機器人的起始位置座標為(40,200)朝向角為 90 度，並讓機器人以逆時針方向繞行。另外，設定線速度 $v=10\text{cm/sec}$ ，角速度 $w=0.15\text{ rad/sec}$ ， $l_{Goal_th}=10\text{cm}$ ， $\theta_{Goal_th}=10^\circ$ 。

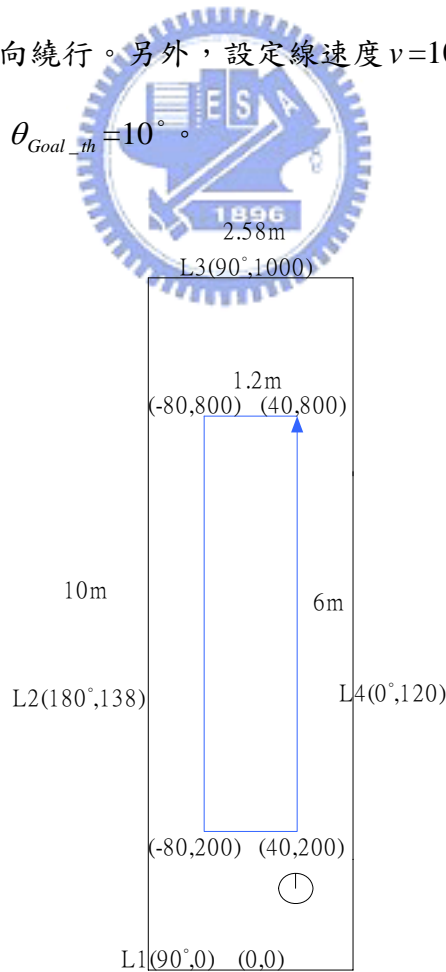


圖 4-15 繞行實驗(二)之環境示意圖

實驗結果如圖 4-16，在圖 4-16(a)中可看出在五圈的繞行過程當中，由定位系統估測的實線軌跡與設定的長方形軌跡相符合，所以可知在長度增加到 10 公尺的環境裡，定位系統仍可有效地估測機器人的姿態，而虛線為 dead-reckoning 估測的軌跡，可看出繞行圈數增加時，偏移情形愈明顯，其偏移的最大值在 x 座標為 115.5 公分， y 座標為 54.8 公分。定位系統估測的機器人軌跡與設定軌跡之間的絕對誤差平均值為 x 座標：8.33 公分； y 座標：9.74 公分；朝向角：7.56 度，其誤差同樣包含了運動控制造成的部分。圖 4-16(b)(c)(d)分別為機器人姿態 (x, y, θ) 與時間關係圖，在設定軌跡附近(x 為 40 與 -80， y 為 200 與 800， θ 為 0、 ± 90 與 ± 180)的值有些會有明顯的凸起或凹下部分，造成的原因與實驗一相同為機器人在繞行過程中修正其朝向角使其朝向目標點前進而產生。

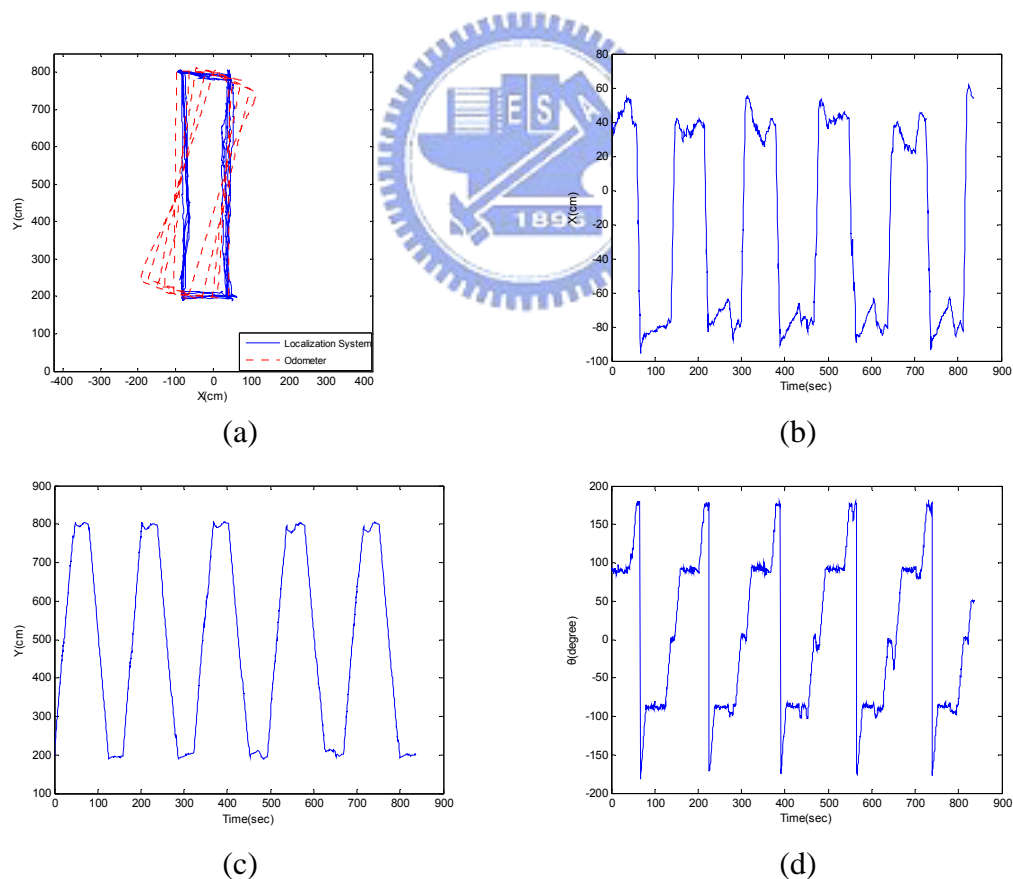


圖 4-16 繞行實驗(二)之結果：(a)機器人移動軌跡 (b) X 座標與時間關係圖

(c) Y 座標與時間關係圖 (d) θ 與時間關係圖

4.3.3 繞行實驗(三)

在這個實驗中，我們將繞行環境改變成如圖 4-17 的 L 型走廊，同樣在環境地圖中將構成走廊環境的八條線段以 L1~L8 表示，其直線方程式參數 (α, ρ) 分別為，L1($90^\circ, 136$)、L2($180^\circ, 105$)、L3($90^\circ, 0$)、L4($180^\circ, 825$)、L5($90^\circ, 446$)、L6($180^\circ, 58$)、L7($90^\circ, 1136$)及 L8($0^\circ, 200$)。機器人繞行的路徑設定成 L 型軌跡，其六個頂點座標為(0,936)、(0,376)、(-345,376)、(-345,216)、(120,216)及(120,936)，機器人的起始位置座標為(0,936)朝向角為 -90° 度，讓機器人以逆時針方向繞行設定的 L 型路徑兩圈。此外，設定線速度 $v=15\text{cm/sec}$ ，角速度 $w=0.15\text{ rad/sec}$ ， $l_{Goal_th}=15\text{cm}$ ， $\theta_{Goal_th}=10^\circ$ 。

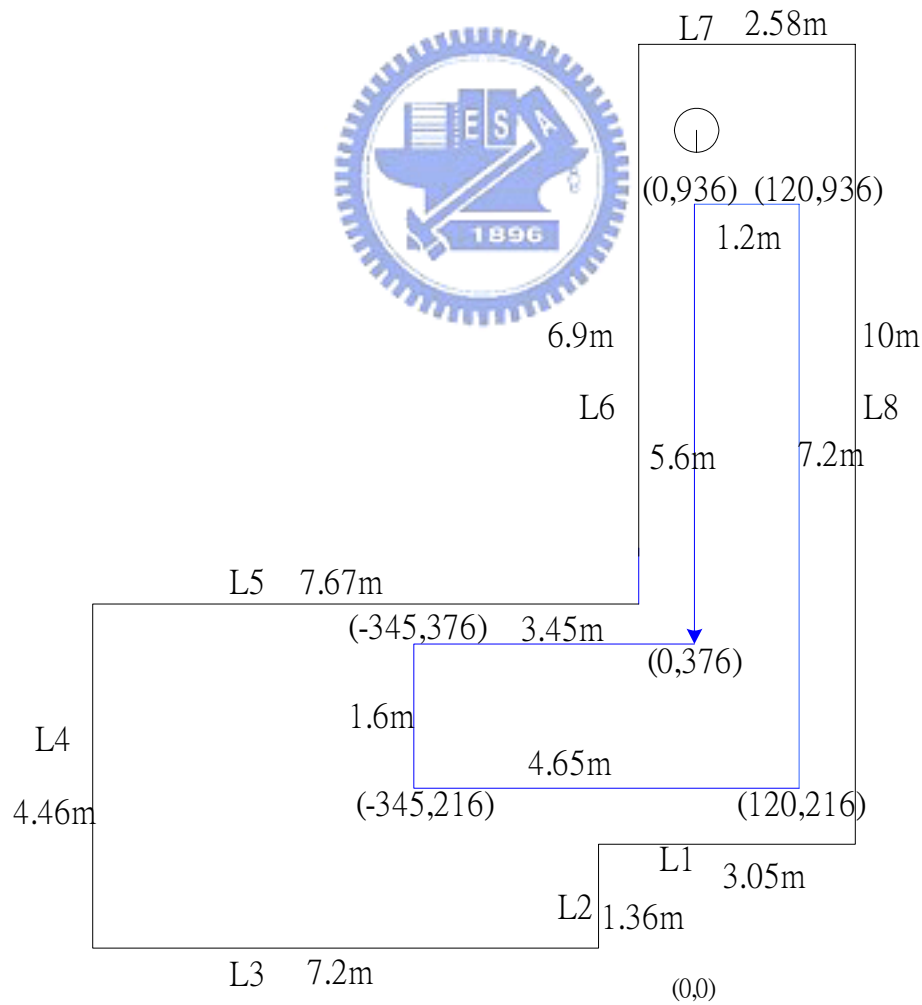
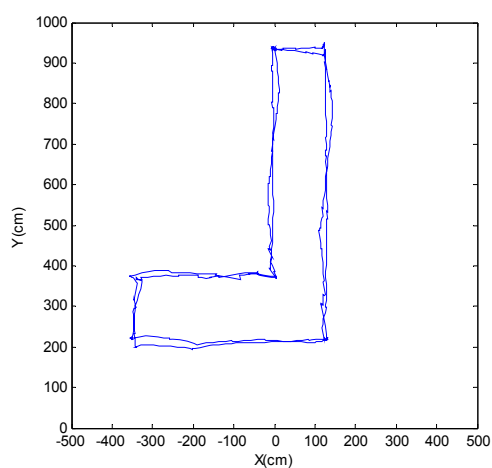
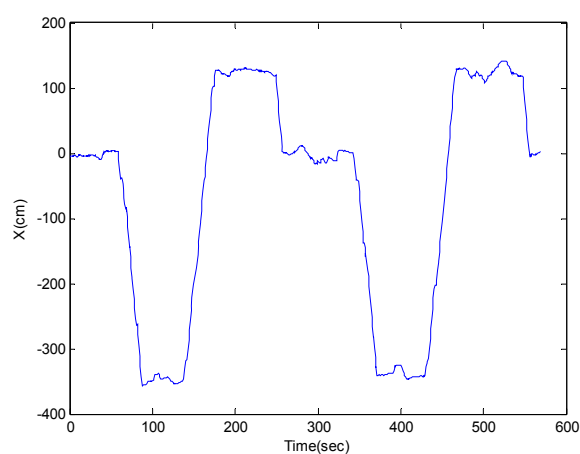


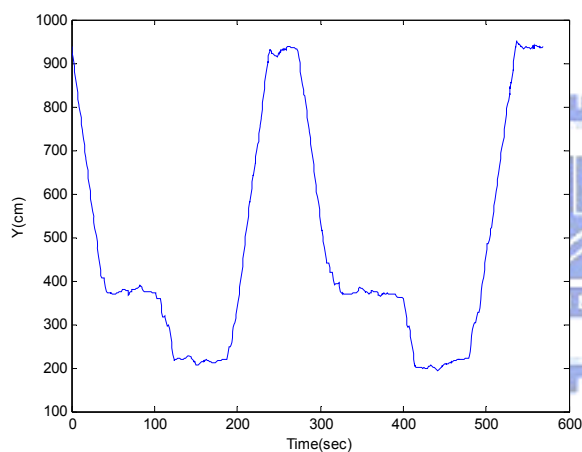
圖 4-17 繞行實驗(三)之環境示意圖



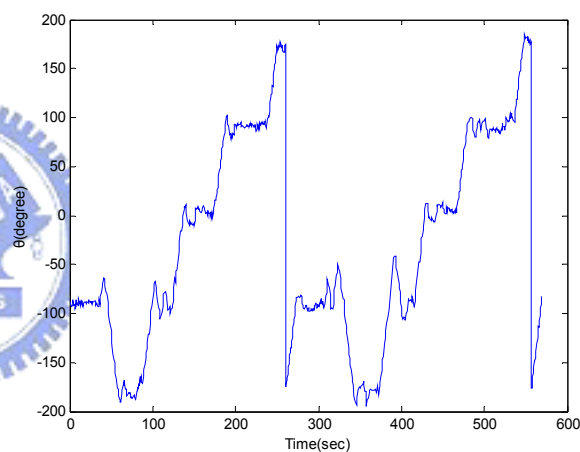
(a)



(b)



(c)



(d)

圖 4-18 繞行實驗(三)之結果：(a)機器人移動軌跡 (b)X 座標與時間關係圖

(c)Y 座標與時間關係圖 (d) θ 與時間關係圖

實驗結果如圖 4-18，在圖 4-18(a)中可看出在繞行過程當中，由定位系統估測的實線軌跡與設定的 L 型軌跡相符合，所以可知機器人在此環境裡繞行，定位系統仍可有效地估測機器人的姿態。另外，定位系統所記錄的機器人繞行軌跡與設定軌跡之間的絕對誤差平均值為 x 座標：9.4 公分； y 座標：9.89 公分；朝向角：8.69 度，同樣地包含了運動控制造成的誤差在內。圖 4-17(b)(c)(d)分別為

機器人姿態 (x, y, θ) 與時間關係圖，在設定軌跡附近(x 為 120、0 與-345， y 為 216、376 與 936， θ 為 0、 ± 90 與 ± 180)的值有些會有明顯的凸起或凹下部分，造成的原因為機器人在繞行過程中修正其朝向角使其朝向目標點前進而產生。



第五章 結論與未來展望

5.1 結論

本論文以粒子濾波器為基礎設計估測機器人位置的方法，並結合以環境比對方式估測機器人朝向角的方法，發展出一個有效的機器人自我定位系統。在位置估測方面，由里程計的資訊可以先對機器人的位置做預測，接著再加入由雷射掃描儀的距離資料產生的環境線段組的資訊，並藉由粒子濾波器計算出估測的機器人位置座標。在朝向角估測上，將雷射掃描儀所得到的環境距離資料以線段擷取的方法擷取成線段組，再將此線段組結合機器人位置座標的資訊與機器人內建的环境地圖做比對，根據比對的結果來估測機器人的朝向角。從繞行環境的實驗結果可以得知，機器人在繞行過程當中其行走路徑皆與所設定的軌跡相符合，而機器人要能依照設定的軌跡移動，必須根據定位系統對其目前姿態的估測結果來決定行進的方向，如果定位系統所估測出來的機器人姿態接近實際的情況，則機器人便能確實的依照設定軌跡而繞行。所以從實驗結果顯示的機器人移動路徑，我們可以驗證設計的定位系統是有效的。

5.2 未來展望

目前所設計的定位系統仍有許多值得進一步探討的地方：

- 1.在環境特徵擷取方面目前是以牆面作為環境特徵來進行地圖比對，未來若能加入其它的環境特徵，如牆角、門等，將可使地圖比對的結果更為精確，提高定位系統的準確度。
- 2.在機器人運動控制方面，因為目前所使用的控制方式是判斷機器人與目標點之間的夾角，當夾角大於設定範圍時才修正機器人前進的朝向角，如此會造成機

器人在偏離設定軌跡時無法即時修正返回設定軌跡的情況產生。所以未來如果能對於機器人系統設計一個運動控制器，使得當機器人根據定位系統的姿態估計得知其已偏離設定軌跡時便能即時修正其運動方向，將可讓機器人行走的軌跡與設定的軌跡更加符合。

- 3.目前的定位系統在加上運動控制的改善後，將可運用於在室內以牆面為主要結構環境中的巡邏機器人的應用上。



參考文獻

- [1] G. Dudek and C. Zhang, "Vision-based robot localization without explicit object models," in *Proceedings of IEEE International Conference on Robotics and Automation*, Minneapolis, Minnesota, USA, 1996, pp. 76-81.
- [2] A. Elfes, "Sonar-based real-world mapping and navigation," *IEEE Journal of Robotics and Automation*, vol. RA-3, no. 3, pp. 249-265, 1987
- [3] I. J. Cox, "Blanche-an experiment in guidance and navigation of an autonomous robot vehicle," *IEEE Transactions on Robotics and Automation: a publication of the IEEE Robotics and Automation Society*, vol. 7, no. 2, pp. 193-204, 1991
- [4] F. Lu and E. Milios, "Robot pose estimation in unknown environments by matching 2D range scans," *Journal of Intelligent & Robotic Systems*, vol. 18, no. 3, pp. 249-275, 1997
- [5] Gijeong Jang, Sungho Lee, Inso Kweon, "Color landmark based self-localization for indoor mobile robots," in *Proceedings of IEEE International Conference on Robotics and Automation*, Washington, DC, USA, 2002, pp. 1037-1042.
- [6] J. Ryde, H. Hu, "Fast Circular Landmark Detection for Cooperative Localisation and Mapping," in *Proceedings of IEEE International Conference on Robotics and Automation*, Barcelona, Spain, 2005, pp. 2745-2750.
- [7] L. Iocchi, D. Mastrantuono, and D. Nardi. "A probabilistic approach to Hough localization," In *Proceedings of IEEE International Conference on Robotics and Automation*, Seoul, Korea, 2001, pp. 4250-4255.
- [8] D. Fox, W. Burgard, Frank Dellaert, and S. Thrun, "Monte Carlo Localisation for Mobile Robots," in *Proceedings of IEEE International Conference on Robotics and Automation*, Detroit, Michigan, 1999, pp. 1322-1328.
- [9] P. Jensfelt, "Approaches to Mobile Robot Localization in Indoor Environments," Doctoral Thesis, Royal Institute of Technology, Sweden, 2001.
- [10] M. Sanjeev Arulampalam, Simon Maskell, Neil Gordon, and Tim Clapp, "A Tutorial on Particle Filters for Online Nonlinear/Non-Gaussian Bayesian Tracking", *IEEE Transactions on Signal Processing*, vol. 50, no. 2 February 2002.
- [11] SICK, "Telegrams for Operating/Configuring the LMS 2xx Laser Measurement Systems," 2003.

- [12] S. Thrun , W. Burgard, and D. Fox, *Probabilistic Robotics*, MIT Press, Cambridge, MA, 2005.
- [13] Hee Jin Sohn, and Byung Kook Kim, “A Robust Localization Algorithm for Mobile Robots with Laser Range Finders,”in *Proceedings of IEEE International Conference on Robotics and Automation*, Barcelona, Spain, 2005, pp. 3545-3550.
- [14] W. Burgard, A.B. Cremers, D. Fox, D. Hähnel, G. Lakemeyer, D. Schulz, W. Steiner, and S. Thrun., “Experiences with an interactive museum tour-guide robot,” *Artificial Intelligence*, vol. 114 , Issue 1-2:1-55, 1999.
- [15] A. Doucet, J. F. G. de Freitas, and N. J. Gordon, “Sequential Monte Carlo Methods in Practice,” New York: Springer-Verlag, 2001.
- [16]J.S. Liu and R. Chen, “Sequential Monte Carlo Methods for Dynamical Systems,” *Journal of the American Statistical Association*, vol. 93, pp. 1032-1044, 1998.
- [17] P. Jensfelt and H. I. Christensen, “Pose Tracking Using Laser Scanning and Minimalistic Environmental Models,” In *Proceedings of IEEE International Conference on Robotics and Automation*, Seoul, Korea, 2001, pp. 138-147.
- [18] Dongheui Lee, Woojin Chung and Munsang Kim, “Probabilistic Localization of the Service Robot by Map Matching Algorithm,” *International Conference on Control, Automation and Systems*, Jeonbuk, Korea, 2002.
- [19] 邱黃正憲, “多機器人合作感測定位與環境地圖建立,” 國立交通大學電機與控制工程研究所碩士論文, 2003.
- [20] 林鎮源, “移動式機器人之行為融合控制器設計,” 國立交通大學電機與控制工程研究所碩士論文, 2005.
- [21] 孫柏秋, “基於行為模式之多機器人協調控制,” 國立交通大學電機與控制工程研究所碩士論文, 2005.
- [22] Ching-Chih Tsai, Yu-Sheng Huang, Yu-Chiao Wang, Ssu-Min Hu, “Design and Implementation of a Nursing-Care Walking Assistant for the Elderly,” *Proceedings of 2005 CACS Automatic Control Conference*, Tainan, Taiwan, Nov 18-19, 2005.