

# 國立交通大學

電機與控制工程研究所

碩士論文

Multi-Agent 全天駕駛輔助系統



The Multi-Agent All-Day Driving Assistance System

研究生：李尚玲

指導教授：徐保羅 博士

中華民國九十四年六月

# Multi-Agent 全天駕駛輔助系統

研究生：李尚玲

指導教授：徐保羅 博士

國立交通大學電機與控制工程學系

## 摘要

駕駛輔助系統的發展是 ITS 的主要計畫之一，經過多年的發展以及許多方面的考量，包括環境、成本等，感測器與車輛的結合似乎是最佳的組合；單一的感測器雖能藉由所處理的訊息，加強某方面的安全性，但其感測的資訊有限，仍無法達到完整的輔助功能，於是，結合多種感測器，互相截長補短以強健輔助系統的能力將是較佳的解決方法。為此，本論文提出以多代理人架構整合之，利用代理人間的溝通特性達到感測器間互補不足的目的。

本論文之輔助系統是架構於多代理人系統上，以感測器為真實環境中的接收器，資料處理程式及多代理人系統是以 VC++ 撰寫，並實現於 PC 系統上。目前，本系統具有 CCD 及雷射掃瞄器二個感測器，由 CCD 所接受的影像經處理後，將防偏、防撞訊息交給影像代理人作為警示依據，而由雷射掃瞄器所接收的距離資訊經處理後，將防撞訊息交給雷射掃瞄器代理人做為警示依據。由本論文列舉出的 10 個範例，可看出經由代理人間溝通後，可以解決單一感測器無法克服的問題。

本影像輔助系統已在高速公路上驗證其偵測能力，而其多代理人駕駛輔助系統，也已成功的架設於實驗用車(高爾夫球車)上，實現於交大校園環境中。

# The Multi-Agent All-Day Driving Assistance System

Student : Shang-Ling Lee

Advisor : Dr. Pau-Lo Hsu

Department of Electrical and Control Engineering

National Chiao-Tung University

## ABSTRACT

By combining the image sensor and the laser scanner, a driving assistance system is proposed in this thesis operated under different light and road conditions. Since the sensing data from all individual sensors are still limited to provide enough driving assistance functions, a feasible driving assistance system combining multiple sensors with exchanged information is proposed in this study to improve driving safety. Different types of sensors can be adopted and integrated as a multi-agent system so that communication with each other overcome the limitations of individual sensors and it leads to a more robust and capable driving assistance system.

Two sensors are adopted in the proposed system, the CCD and the laser scanner. After processing the image data under different light conditions, the developed image agent obtains information of the road lanes and the front vehicles; different algorithms for day and night determined by an Otsu criterion are processed separately. Therefore, the functions of lane keeping and collision avoidance can be achieved for driving under all day. Moreover, after processing the distance data from the laser scanner, the laser-agent obtains all information of the front vehicles by applying the Kalman filter accurately.

Finally, algorithms of all signal processing and multi-agent system are implemented using the VC++ . Ten typical models of driving conditions are discussed in this thesis and they indicate that through the communication of agents, difficult road conditions can be properly identified and the shortage of the single sensor is thus overcome. The detection capability of the imaging assistance system without laser radar has been proven on the high way; and furthermore, the multi-agent driving assistance system has also been realized on a golf car in the campus of National Chiao-Tung University.

## 誌謝

回想起這二年的研究生涯，首先要感謝的是指導教授徐保羅博士對我的熱心指導以及照顧，且讓我有充分的發展空間，使我能順利的完成碩士學位，在此表達我最深誠的敬意與感謝。另外感謝口試委員吳炳飛博士與黃育綸博士對於本論文的指正，並給予寶貴的建議，使得本論文更加完善。

感謝實驗室 816 的夥伴們：博士班謝鎮洲學長、幸琮政學長、王安平學長、張昭琳學長和已畢業的李俊賢學長，平時給我的意見、指導與幫助，以及實驗室同學景文、議寬、學弟元銘、興漢、宗翰、瑞原、孝麟，在學業上的相互切磋與指教，並讓實驗室裡增添了不少的歡樂與笑聲，還有新進實驗室來的學弟們，在最後口試階段的幫忙。除此之外，特別對琮政學長、瑞原學弟、以及已畢業的政衍學長在研究與實驗上的幫助，獻上誠摯地感謝。

並感謝今年一起從交大電控所畢業的男友大漢，有了你的陪伴豐富了我的研究生活。最後，將此論文獻給我最敬愛的父母與家人，因為有你們在身旁的支持與關懷，使我能夠心無旁騖地完成學業，並且順利地克服我人生中的每一個重大考驗。

感謝所有曾經幫助過我與默默祝福我的朋友，謝謝你們。

# 目錄

中文摘要	i
英文摘要	ii
誌謝	iii
目錄	iv
表目錄	vii
圖目錄	viii
第一章 緒論	1
1.1 研究動機與目的	1
1.2 研究背景與發展概況	3
1.3 問題界定	6
1.4 研究方法與步驟	7
1.5 論文貢獻	7
1.6 論文架構	9
第二章 全天影像偵測系統	10
2.1 白天影像偵測系統	10
2.1.1 以對稱特性偵測前方車輛之方法	10
2.1.2 IPM based 之車道線偵測方法	15
2.1.3 整合型車道線偵測方法	19
2.2 夜間前方車輛偵測方法	22
2.2.1 前處理方法	22
2.2.2 車輛判定方法	25

2.3	夜間前方車輛追蹤方法	27
2.4	夜間車道線偵測方法	32
2.5	白天與夜間影像偵測系統之整合	34
	2.5.1 Otsu 演算法	34
	2.5.2 日夜演算法及實驗結果	37
2.6	影像代理人	40
	2.6.1 影像代理人警告訊息	40
	2.6.2 影像代理人程式介面	44
第三章 雷射掃瞄器防撞偵測系統		46
3.1	車輛前方之雷射掃瞄器於接近前車時的實驗	46
3.2	Kalman Filter 於車輛行駛時的速度估測	49
3.3	雷射掃瞄器代理人	53
	3.3.1 雷射掃瞄器代理人警告訊息	53
	3.3.2 雷射掃瞄器代理人程式介面	57
	3.3.3 與影像代理人於防撞方面的比較	59
第四章 全天駕駛輔助系統		61
4.1	代理人的介紹	61
4.2	系統整合	68
	4.2.1 以代理人(Agent)整合影像與雷射掃瞄器	68
	4.2.2 多代理人溝通管理平台	70
4.3	整合能力展現	74
	4.3.1 雷射代理人輔助影像代理人	75
	4.3.2 影像代理人輔助雷射代理人	79

4.4	系統架構與軟體處理流程	81
4.4.1	系統架構	81
4.4.2	軟體處理流程	82
第五章 結論與未來發展		85
5.1	結論	85
5.2	未來發展	86
參考文獻		88



## 表目錄

表 2.1	色度座標轉換 RGB→YCrCb	23
表 4.1	代理人間的傳遞訊息表	73
表 4.2	代理人間的傳遞訊息表	74





## 圖目錄

圖 2.1	特定點的分布	12
圖 2.2	二值化後的影像	12
圖 2.3	(a)水平邊抓取後的影像 (b)垂直邊抓取後的影像 (c)車輛輪廓抓取 (d)選取對稱值最高其對稱軸	13
圖 2.4	偵測前方影像的車輛位置座標	14
圖 2.5	以對稱特性偵測前方車輛之方法其程式流程	14
圖 2.6	以對稱特性偵測前方車輛之方法其偵測結果	15
圖 2.7	兩個座標系( $W$ and $I$ )的關係	16
圖 2.8	$xy$ plan in the $W$	16
圖 2.9	$z\eta$ plane in the $W$	16
圖 2.10	(a) 原始圖 (b) IPM 後的結果	17
圖 2.11	細化處理後的影像	18
圖 2.12	(a) 所找出的車道線 (b) 將車道線對應回原始圖上	18
圖 2.13	IPM based 之車道線偵測方法其程式流程	19
圖 2.14	所找出的偵測視窗	20
圖 2.15	動態視窗之車道線偵測方法其程式流程	20
圖 2.16	動態視窗之車道線偵測方法所定義出的車道線影像	21
圖 2.17	整合型車道線偵測方法其程式流程	21
圖 2.18	彩色影像經灰階、二值化(閾值=200)處理後的結果	23

圖 2.19	彩色影像經色度座標轉換、條件式二值化 (閾值為 $Y=145$ , $Cr=130$ )處理後的結果	24
圖 2.20	經“尋找單一光點”後的結果	25
圖 2.21	“尋找成對光點”步驟圖解以及實際執行結果	26
	(a) 前一刻影像所得單一光點質心位置	
	(b) 與後一刻影像比對區塊內質心移動趨勢	
	(c) 二個光點合併為同一區塊	
	(d) 實際影像執行結果	
圖 2.22	相關(correlation)的圖解	28
圖 2.23	追蹤步驟圖解	29
	(a) 尾燈樣本沿著 X 方向在區域內做相關性運算	
	(b) 找到 X 方向最大相關後，再延 Y 方向尋找	
	(c) 找到區域內的最大相關位置，完成樣本比對處理	
圖 2.24	追蹤處理展示(frame_27 ~ frame_2208)	30
圖 2.25	夜間前方車輛偵測與追蹤流程	31
圖 2.26	在不同的環境偵測及追蹤前方車輛之結果	31
圖 2.27	不同的閾值所得之車道線偵測結果	32
圖 2.28	各種路況之車道線偵測結果	33
	(a)無車 (b)有車 (c)彎道 (d)有車	
	(e)彎道 (f)路面上有路標 (g)黃昏 (h)雨後路面	
圖 2.29	Otsu 演算法實驗結果	37
	(a) 輸入影像	
	(b) 輸入影像之直方圖分析結果	
	(c) 由 Otsu 演算法所得之最佳閾值 k	
	(d) 亮區與暗區的區域極值	

圖 2.30	不同日夜狀況的 Otsu 明亮度及其直方圖分析	38
圖 2.31	根據二種明亮度指標之偵測結果	39
圖 2.32	圖像經 IPM 轉換後的示意圖	41
圖 2.33	車道線方向角	41
圖 2.34	位置左右對稱性	41
圖 2.35	車道線角度差及距離差的計算結果範例	42
圖 2.36	前車相對距離的計算範例	43
圖 2.37	夜間前方車輛錯誤偵測	43
圖 2.38	影像代理人程式介面	44
圖 3.1	(a) LMS 外型	46
	(b) 雷射光點直徑與相對距離	
圖 3.2	實驗用高爾夫電動車	47
圖 3.3	裝置在車輛前方的雷射掃瞄器	47
圖 3.4	(狀況 1)透過雷射掃瞄器所得到的相對距離與相對速度	48
圖 3.5	(狀況 2)透過雷射掃瞄器所得到的相對距離與相對速度	48
圖 3.6	透過雷射掃瞄器得到相對速度統一單位(m/s)	50
圖 3.7	Kalman filter 對圖 3.6 的速度估測 (速度變化較劇烈)	51
	$(p(1) = 1, \sigma_w^2 = 0.03, \sigma_\gamma^2 = 0.1)$	
圖 3.8	Kalman filter 對左圖的速度估測(速度變化較緩和)	52
	$(p(1) = 1, \sigma_w^2 = 0.03, \sigma_\gamma^2 = 0.1)$	
圖 3.9	相對距離與相對速度	53
圖 3.10	相對距離除以相對速(D/V)	53
	(原始結果，未經過 Kalman filter 處理)	
圖 3.11	圖 3.10 相對距離除以相對速度(D/V)放大刻度	54
圖 3.12	相對距離除以經過 Kalman filter 估測的速度	55

圖 3.13	相對距離與相對速度	55
圖 3.14	相對距離除以相對速度	55
圖 3.15	(a) 相對距離除以相對速度放大刻度 (b) 相對距離除以估測的相對速度	56
圖 3.16	真實與虛擬的雷射瞄器配置	56
圖 3.17	設定雷射掃瞄器參數之程式畫面	57
圖 3.18	雷射掃瞄器與溝通管理伺服程式連線設定，及所獲得其它代理人所發出的訊息顯示之程式畫面	57
圖 3.19	雷射掃瞄器代理人程式主畫面 (a) 文字模式畫面 (b) 前方物體模式畫面 (c) 繪圖模式畫面	58
圖 3.20	前方車輛測距實驗場景圖	59
圖 3.21	雷射掃瞄器代理人測得資料 (a) 相對距離與相對速度 (b) D/V 曲線圖	59
圖 3.22	影像代理人測得資料 (a) 相對距離與相對速度 (b) D/V 曲線圖	60
圖 4.1	Gilbert 對 Agent 的分類	62
圖 4.2	單代理人系統的架構	64
圖 4.3	MARS 模型概念圖	65
圖 4.4	”multi-world”示意圖	66
圖 4.5	集中式程式設計	67
圖 4.6	分散式程式設計	67

圖 4.7	FIPA 所提的代理人系統架構	70
圖 4.8	本論文之多代理人其溝通、管理平台及架構	71
圖 4.9	本論文提出的代理人溝通語法格式	72
圖 4.10	影像及雷射掃瞄器代理人溝通管理伺服程式	72
圖 4.11	訊息傳遞示意圖	73
	(a)廣播式	
	(b)目標式	
圖 4.12	影像與雷射代理人合作 Model_1 之示意圖	75
圖 4.13	影像與雷射代理人合作 Model_2 之示意圖	76
圖 4.14	影像與雷射代理人合作 Model_3 之示意圖	76
圖 4.15	影像與雷射代理人合作 Model_4 之示意圖	77
圖 4.16	影像與雷射代理人合作 Model_5 之示意圖	77
圖 4.17	影像與雷射代理人合作 Model_6 之示意圖	78
圖 4.18	影像與雷射代理人合作 Model_7 之示意圖	79
圖 4.19	影像與雷射代理人合作 Model_9 之示意圖	80
圖 4.20	影像與雷射代理人合作 Model_10 之示意圖	80
圖 4.21	USB 埠 CCD	82
圖 4.22	UPS 和 HUB	82
圖 4.23	系統設備的配置圖	82
圖 4.24	系統程式流程	83
圖 4.25	多感測器多代理人系統示意圖	84

# 第一章 緒論

## 1.1 研究動機與目的

智慧型運輸系統(intelligent transportation system ; ITS)係藉由先進之電腦、資訊、電子、通訊與感測等科技的應用，透過所提供即時資訊的溝通與連結，以改善人、車、路等運輸次系統間的互動關係，進而增進運輸系統之安全、效率與舒適，同時減少交通環境衝擊之有效整合型運輸系統。其包含的範圍相當廣泛，且早在 1930 年之後就已陸續被提出來[1-3]，但礙於當時技術面不成熟而未受到人們重視，一直到 1986 年，歐洲 19 個國家率先聯合成立了 Prometheus 計畫(program for European traffic with highest efficiency an unprecedented safety project)才改變局勢，尤其自 90 年代以後，汽車已成為陸地上最重要的交通工具，也由於車子的普及率越來越高，使得交通日漸壅塞，隨之而來的就是交通意外事件頻繁，汽車的安全性受到了嚴苛的考驗。因此，各先進國家(歐洲、美國、日本)都投注了相當大的心力，積極發展先進車輛控制及安全系統(advanced vehicle control and safety system ; AVCSS)，結合感測器、電腦、通訊、電機及控制技術，應用於車輛及道路設施上，協助駕駛人駕駛、提高行車安全、增加道路容量、減少交通擁擠，以改善行車安全。

各先進國家經過 20 餘年的研究，先進車輛控制及安全系統的架構與技術日趨完善，不少的團隊也已進入實車實驗。然而我們發現大多的研究仍侷限於白天晴朗的天候，對於夜間或其他不理想的天候狀況著墨甚少，更甚者是將不同的晝夜狀況獨立分開發展，方法系統林立，難以統整成一套完整系統。如此的情況與我們理想的 AVCSS 相距甚遠，因此，本研究將屏除困難，致力發展出全天的 AVCSS。

根據內政部警政署統計，歷年來台灣地區每年因交通事故而死亡的人數皆維持在三千人左右，分析其肇事原因，有百分之 70-90%的交通事故都是起因於駕駛人操作不當所致。以 90 年為例，前三項為未依規定讓車 16.76%、轉彎(向)不當(含左、右、迴轉)14.47%及未注意車前狀況占 13.71%。而統計其事故發生時間，夜間件數為日間之 1.10 倍。而一天中以 18-20 時件數最多，而 2-4 時、8-14 時件數較少。分析其原因，通常是入夜後車流量漸小、空間加大及車速加快，或因精神恍惚，意識不清楚、造成判斷力及反應力遲緩，致衝撞停滯於車道之車輛，釀成慘重傷亡事故[4-5]。有鑒於此，ITS 的首要目標應是利用資訊及控制等技術來輔助駕駛人行車、增進交通安全，且夜間輔助系統的發展實應與白天具有同等的重要性。

在目前駕駛人輔助系統上，夜間的演算法跟白天的演算法一樣可分為防撞與防偏警告兩方面，主要是針對駕駛人周圍的障礙物做出預警以及協助駕駛人在正確的路線上行進。其輔助系統常用的感測器主要分為主動式和被動式[3][7]，主動式感測器包括毫米波(millimeter wave，也就是微波)和光學式雷射掃瞄器(laser scanner)，其中毫米波感測器因為價格昂貴及體積龐大，已鮮少被使用；雷射掃瞄器具有精確的距離偵測，因此多應用於防撞系統。被動式感測器，也就是指機器視覺(machine vision)，包括了 CCD、CMOS 及紅外線(infrared)攝影機，其中，CCD 和 CMOS 攝影機所拍攝的是與人類肉眼所見相同之黑白或彩色影像，而紅外線攝影機則為熱源影像。在防偏系統的發展上，雷射掃瞄器容易受到距離的限制或必須有路旁欄杆來作為反射訊號用，又因影像畢竟是比較接近人類視覺感官，比較容易為大部分人所接受，故一般還是以影像為基礎來實現這類系統。

代理人(Agent)的概念始於 MIT (Massachusetts institute of technology)，實屬於人工智慧(artificial intelligence；AI)的一類。它具有許多仿人的特性：自主性、互動性、智慧性等等，其應用非常廣泛，如商業應用中的資訊管理、電子商務和商業流程管理；醫療應用方面的病人監視、健康監視；娛樂方面的遊戲和互動式電影，而在工業應用中的製造與控制系統[18-21]、核電廠控制[22]，都有引入代

理人系統的案例。分散式人工智慧(distributed artificial intelligence ; DAI)系統更是現今熱門的話題，其優點是當 DAI 系統遇到較複雜的問題時，會集結不同的觀點，整合解決方法、降低不確定性，並且其平行處理的方式增進了效能。有學者看中 DAI 所具的優點將其架構用於動態環境的控制問題上，因而提出了以多代理人(Multi-Agent)為基礎的運輸系統[6]。儘管大部分的研究還在模擬階段，但可觀察出其所擁有的性質是與 ITS 系統相輔相成的，因此此二系統的結合必將是未來的趨勢。

因此，在本論文中，結合 CCD 影像、雷射掃瞄器與多代理人架構，建構一套穩健的全天輔助系統。影像方面，經由處理獲得的夜間影像定義出車道線及前方車輛，並將車道線與車輛之位置突顯出來，接著利用並延伸 Otsu 演算法[23]來整合白天演算法，使其能依據當時的日夜狀況準確、順暢的切換狀態；雷射掃瞄器方面，透過安裝在實驗車輛前方的雷射掃瞄器，發展出利用所測量與前方車輛的距離，得到雷射掃瞄器與前車間的距離資訊，經過進一步資料處理分析後可以得知兩車間相對速度；多代理人方面，經由集結影像和雷射二方代理人提供的資訊，判斷出目前車前方正確的狀況，以此作為產生警示訊息的依據。最後，進而完成一全天駕駛人輔助系統。

## 1.2 研究背景與發展概況

防偏與防撞輔助系統在汽車安全上一直是專家學者積極研究與發展的領域。90 年代以後，世界各國投入了大量的人力和物力在此領域的研究與實驗，以期為保障汽車行駛安全、降低肇事率作出貢獻。在美國、日本甚至其它歐美先進國家，都已經在防偏與防撞輔助系統上有許多研究成果出來了[3][7]，例如歐洲的 VaMoRs、VITA II、ARGO 系統、美國的 Navlab 系統、日本的 VICS 系統等等。我們發現，其系統接收感測器多是 CCD，原因是 CCD 所擷取的影像相



近於人的視覺，因此，相對的，影像處理的方法也會變得出於直覺，易於研究的進行。然而當所拍攝的影像變成夜間時，影像明亮度、對比度皆大大的降低，內容物變得不清晰而難以辨識，造成原本用於白天的影像處理方式無法使用或辨識率大幅降低的情形，因而鮮少看到全天的輔助系統。

全天系統方面，特別是針對夜間駕駛，Betke et al.[8]的系統僅使用單眼 CCD 及 PC-based 架構進行全天的偵測，其夜間的演算法僅針對前方車輛的防撞，利用偵測成對且對稱的紅光，也就是車尾燈來確定車體，實驗結果是對於正前方車輛的偵測有不錯的效果，而對於其他車道則容易偵測錯誤。Gangyi et al.[9]的系統也是利用 CCD 擷取畫面，將其灰階影像經過濾波器留下所需資訊，再定義出車道線及前方車輛，其中車輛的偵測也是經由尋找車尾燈。Okamoto et al.[10]則是在已知車道線資訊的情況下，只尋找駕駛人車道線內的前方車輛，首先將彩色影像進行色度座標轉換，由 RGB 座標轉至 YCrCb 座標以突顯紅色圖像，再進行成對光點的搜尋以確定前方車輛的存在，並經由觀察影像的明亮度，簡易的將天候區分為白天與夜晚，以進行演算法的切換，因此實驗結果在夜間的平均準確度是 59.8%。

此外，也有不少團隊單獨的發展夜間輔助系統，如 Tsuji et al.[11]的系統是使用二台紅外線(infrared)攝影機擷取左右二張影像，經影像處理器取得前方資訊(車道線、車輛和行人)，對於危機狀況可發出警示訊息，並可透過位於擋風玻璃底部的 HUD(Head-up Display)提醒駕駛人前方行人的位置，完成一完整的夜間防撞系統。Chern et al.[12]是針對台灣夜間高速公路所做的研究，利用單眼 CCD 感測器，透過尋找影像中路面光點及近似二次方程曲線的方式定義出車道線，而車輛的定義則是將彩色影像經由不同的 R、G、B 閾值濾出可能的車尾燈，再經過成對且平行的法則而得。

另外，值得注意的是，在定點路況監控的研究中，也有許多夜間偵測的技術被發展出來，甚至已發展至全天偵測的完整系統。Kashiwagi et al.[13]利用擷取紅外線影像，配合二種演算法，分別是特徵視窗(eigenwindow)法及向量量化

(vector-quantization)法，實現於 IMAP 平行影像處理板上，加上事先樣本影像的學習，精確的達到全天辨識車輛的效果，正確率並有 90%。Cucchiara et al.[14]的感測器是用單眼 CCD，他們提出將監控系統分為低階與高階模組，低階也就是影像處理模組，執行影像分析取出車輛；高階也就是推理式模組，藉由一組啟發式法則再次的確認並防止錯誤的發生，高低模組共同合作的結果達到穩健的車輛追蹤，並使系統增添了擴充性。

關於代理人(Agent)的理論及架構發展早已行之有年，且日趨成熟，儘管大家對這個名詞還是有點陌生，但隨著微處理器及嵌入式元件的蓬勃發展，隨時隨地皆可”運算(computing)”的結果，使得其應用早已深入我們的日常生活中。例如：運輸系統-GPS，家用玩具-機器狗，遠端控制方面-行動式機器人和無人的衛星工作站，軍事上...等等。甚至在不久的未來，我們生活週遭的物品說不定皆可感測、行動、思考、或是通訊，這都是因為”內建代理人(Agent inside)”的關係[15]。Lekkas et al.[6]提出適用於動態環境中的代理人架構及發展方法，他表示在動態環境中，傳統具知識庫(knowledge-based)的專家系統無法與人類或其它系統互動，因而提出結合 DAI 的觀念，也就是多代理人的架構來改善此種狀況，利用廣集多位代理人的觀點找出最佳解決方案，具有錯誤容忍、降低不確定性、和增加處理效能的優點。目前，結合代理人與運輸系統的研究多屬於電腦模擬，例如，交通狀況監控的應用上，利用電腦模擬號誌燈行為、車輛行為等等(Li et al.[16])，Papp et al.[17]進一步的試著將代理人與運輸系統上的感測器結合，提出將影像攝影機與雷射掃瞄器等感測器以元件的方式加入 MARS(multi-agent real-time simulator)架構中，目前只進行感測器模擬階段以評估 MARS 架構效能。

可以發現，由代理人觀點出發的運輸系統，其注重架構化是其優點，此將使得系統具有擴充性，且理論上對於動態環境具有適應性，然而由於太著重理論發展的結果，使其目前的研究進度多在電腦模擬的階段，尚未進行至實際環境的具體驗證。

### 1.3 問題界定

- 1、由於白天的防偏與防撞系統已發展成熟，於是我們進一步的要發展兼具夜間的防偏與防撞系統。為了系統的完整性，感測器仍使用 CCD 以及雷射掃瞄器，並且以 VC++ 語言，也就是 PC based 來架構防偏與車前防撞系統。
- 2、在我們的系統是以道路上的車道線與車輛做為判斷的資訊，故除了車道線與車輛之外，其他所有的影像皆可視為雜訊。在影像方面，由於夜間光線不足，導致所擷取影像明亮度及對比度差，所以我們必須找出一些影像加強的過程來突顯目標物，以及一些基本的二值化過程來有效的濾除雜訊，以確實的偵測出車道線與車輛，並以此為依據來建立防偏與車前防撞警告。
- 3、為了達到全天的輔助警告系統，全天的影像監控是必要的，也就是要整合白天和夜間的演算法，影像系統要能針對不同的日夜狀況，自動且準確的切換狀態。
- 4、在此論文中，利用雷射雷達量測與前車之間的相對距離，得到相對速度，理論上，可用 Kalman filter 來估算兩車間的相對速度，根據實驗得到數據來看車輛前方防撞的問題點，提出發生碰撞前事先警告駕駛人剎車的方式，發展一個可靠的車輛前方防撞警告。
- 5、由於我們最終的目的是，建立一穩健且具擴充性的全天防偏與車前防撞警告系統，我們引入多代理人觀念及架構，藉由整合影像代理人以及雷射掃瞄器代理人的資訊，來判斷是否發出警示訊息，並有相對的人機介面來顯示目前偵測狀況，使用者(駕駛人)也可透過此介面對影像及雷射掃瞄器代理人下達指令。

## 1.4 研究方法與步驟

本論文研究的方法與步驟如下所述：

- 1、針對影像方面，我們將 CCD 架在實驗用車輛(在此為高爾夫球電動車)上，透過 USB 與 PC 端連接，首先我們將實際在車道上行駛的狀況先錄成影片，或儲存為單張圖片，再利用其資料作影像上的分析，進一步發展出車道線與前方車輛擷取的方法，最後，再實際驗證其法則的正確，作細部的調整，或法則的修改，達到能在實際狀況下，正確的抓取車道線與前方車輛資訊，發展防偏與前方防撞準則。
- 2、在前方防撞的實驗中，我們安裝雷射雷達在實驗用車輛的前方，透過 PC 端串列埠對雷射雷達取得每一筆距離及兩段資料間的時間差，將得到相對距離與時間數據抓取下來後，在 PC 端上來做相對速度的分析處理，以便作為進一步發展防撞準則的判斷條件。
- 3、多代理人方面，其架構架設於自行開發的溝通管理平台，溝通訊息也是採用自訂的格式。在此我們將建立二個代理人：影像代理人與雷射掃瞄器代理人，此二代理人皆可經由人機介面得知目前彼此訊息處理狀況，以及調整其系統參數以符合所需。此二代理人皆可發出與接收訊息，並透過溝通管理平台傳遞訊息，進而建立一個更佳穩健的防偏與車前防撞警告系統。

## 1.5 論文貢獻

本論文之貢獻如下所述：

- 1、在影像處理方面：

我們微調白天之車道線偵測方法使其適用於夜間影像進行偵測，並且利用

CCD 所擷取之彩色影像，經色度座標轉換(color-space transform)後偵測車輛紅色的車尾燈，以平行等高、成對、長寬比等法則定義出車輛位置。而這兩種方法在 PC based 架構下，能具有即時性與較高的辨識率或較能適應適應各種環境的變化。

(1) 在車道線偵測方面：

承接白天之整合型車道線偵測方法，由於夜間影像中，車道線與路面的色差較小，因此我們調整其中濾波器的閾值以適應夜間路面環境，再來並以車道線的角度差與其位置差為防偏警告的依據。除了承接 IPM 偵測法之準確性外，也可達到程式演算法精簡及可重複性。

(2) 在前方車輛偵測方面：

程式主要分為“偵測”與“追蹤”二步驟，提出將影像進行色度座標轉換 RGB → YCbCr，透過條件式的二值化處理可輕易地將車尾燈以外的雜訊(包含車道線)剔除，再經過圖點搜尋、平行等高、成對、長寬比驗證等法則後可得偵測車輛。之後還會進行追蹤的步驟，利用圖形比對方法，也就是執行區域性的相關(correlation)處理達到車輛的追蹤，不用一再的重覆進行偵測步驟，可節省處理速度與增進效能，經實驗結果，追蹤步驟也較偵測步驟抗雜訊，提供一在 PC based 架構下，即時地搜尋前方車輛，並依所得的車輛位置，再與雷射掃瞄器的資料相互配合，作為前方防撞警告的依據。

## 2、系統整合方面


(1) 影像系統的整合：

以 Otsu 演算法[23]為基礎，提出日夜決策法則，透過 Otsu 明亮度和平均明亮度二指標，使影像監控系統能自動的偵測當時天候狀況，準確的切換白天或夜間的演算法，讓駕駛人擁有一全天的影像輔助系統。

## (2)多感測器系統的整合：

在此方面我們引進多代理人溝通合作觀念，主要因為由於單一個別獨立的感測器，其能力有限，所以，我們利用多代理人溝通合作觀念，利用相互間的資訊傳遞，互補各感測器的不足，如此可以減少一些不必要的錯誤，及處理流程，提升系統的穩定與效能，並在程式設計方面，由於包含了分散式的概念，在程式撰寫上會相對的比較簡單，因只需考慮單一感測器，再考慮溝通與配合的撰寫即可，且利用網路作溝通合作，其擴充性及修改性也較好，因為只需了解其相互間的傳輸模式，就可加入此系統，即可擷取或提供所需的資訊，即使需要修改相互間的程式，也只需作部份的修改即可完成，讓系統有更大的彈性。

## 1.6 論文架構



本論文共分五章，首先第一章中旨在說明研究的動機與目的、研究的背景與發展、界定出所要研究的問題、研究方法與步驟以及本論文之貢獻。第二章為說明在夜間影像中如何偵測前方車輛以及車道線，將所使用的方法做一詳細的介紹，以及如何的整合白天與夜間的偵測系統，之後引入影像代理人的觀念。第三章介紹透過安裝在實驗車輛前方的雷射掃瞄器偵測前方障礙物，利用一維 Kalman Filter 改善偵測品質，並根據相對速度與相對距離得到 D/V(distance/velocity)曲線，作為雷射掃瞄器代理人的防撞警示標準。第四章描述有關於以代理人為基礎之防偏與車前防撞警告系統其架構與實現，並且說明影像與雷射代理人如何藉由溝通、協調以互補不足，進而完成一全天的駕駛輔助系統。在最後第五章中會對本論文做一個結論，並提出未來可再改進發展之方向。

## 第二章 全天影像偵測系統

本章說明在白天和夜間影像中如何偵測前方車輛以及車道線，將所使用的方法做一詳細的介紹，以及如何的整合白天與夜間的偵測系統，最後引入影像代理人的觀念於本系統中。

### 2.1 白天影像偵測系統

以下簡單介紹在白天的影像中如何的偵測前方車輛與車道線。此部份是由吳政衍學長完成[24]。

#### 2.1.1 以對稱特性偵測前方車輛之方法

在偵測前方車輛方面主要利用車輛外形的特徵，來作偵測前方車輛的依據，此處理方法、步驟較簡單，所以處理速度較快，較可以達到 PC based 下即時的要求，但辨識率不會非常好，因此，我們引入代理人的觀念來改善其辨識率(第四章將會討論)。

一般而言，前方車輛其外形都成左右對稱，因此我們利用此對稱特性來找出前方車輛，在對稱方面除了一般常提到的灰階對稱外，我們還考慮邊的對稱、水平邊的對稱和垂直邊的對稱，來增加偵測上的準確性，並再利用車輛輪廓特徵作進一步確認。

##### a. 在對稱性量度方面

利用 contrast function ( intensity function ) 來度量對稱程度的大小。一般而言，任何方程式都可以化為奇函數與偶函數的和，如(2-1)式所示，而在量度對稱性方面，也是利用將灰階值函數化為奇函數與偶函數，再考慮對稱寬度參數，形

成偶函數 (symmetry function)和奇函數 (anti-symmetric function)，如(2-2)式與(2-3)式所示。

$$\begin{aligned}
 G(u) &= G_e(u) + G_o(u) \\
 G_e(u) &= \frac{G(u) + G(-u)}{2} \quad \text{偶函數} \\
 G_o(u) &= \frac{G(u) - G(-u)}{2} \quad \text{奇函數}
 \end{aligned} \tag{2-1}$$

$$E(u, x_s, w) := \begin{cases} \frac{1}{2}(G(x_s+u)+G(x_s-u)), & \text{if } -\frac{w}{2} \leq u \leq \frac{w}{2} \\ 0 & \text{otherwise} \end{cases} \tag{2-2}$$

$$O(u, x_s, w) := \begin{cases} \frac{1}{2}(G(x_s+u)-G(x_s-u)), & \text{if } -\frac{w}{2} \leq u \leq \frac{w}{2} \\ 0 & \text{otherwise} \end{cases} \tag{2-3}$$

$W$ ：所設定之對稱寬度     $u$ ：寬度變數     $x_s$ ：對稱中心

再利用偶函數和奇函數計算其相對應的能量值(Energy[f(x)]= $\int f(x)^2 dx$ )，並由所得的能量值獲得 contrast function ( intensity function ) ( $=\frac{E[even]-E[odd]}{E[even]+E[odd]}$ )，但由於(2-3)式之均值(mean value)為零，而(2-2)式之均值為某一正數，其基準值不同，所以要將(2-2)的均值調整為零，以求一致性，如(2-4)式所示，而最後所獲得的 contrast function ( intensity function )如(2-5)式所示。

$$E_n(u, x_s, w) := E(u, x_s, w) - \frac{1}{w} \int_{-w/2}^{w/2} E(v, x_s, w) dv \tag{2-4}$$

$$S(x_s, w) = \frac{\int E_n(u, x_s, w)^2 du - \int O(u, x_s, w)^2 du}{\int E_n(u, x_s, w)^2 du + \int O(u, x_s, w)^2 du} \tag{2-5}$$

所獲得的 contrast function ( intensity function ) 其值在-1 與+1 之間 ( $-1 \leq S(x_s, w) \leq 1$ )，而當  $S=1$  是理想的對稱性， $S=0$  是完全沒有對稱性， $S=-1$  是理想的反對稱性(anti-symmetry)。



在前方車輛偵測方面，但若將圖上所有的圖點都作對稱值的運算，其處理時間會相當的長，不符實際需求，所以我們只設定在特定的偵測區域內，特定點作對稱值的運算，如圖 2.1 所示其特定點分布。

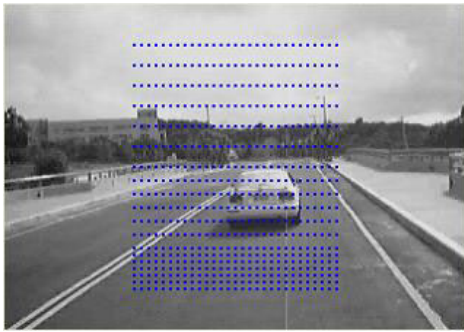


圖 2.1 特定點的分布

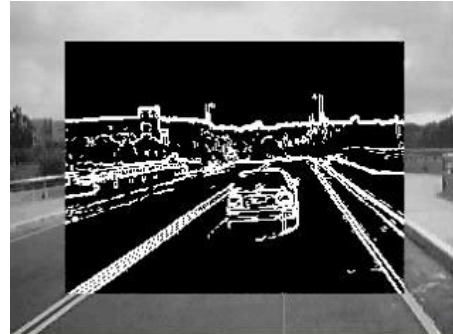
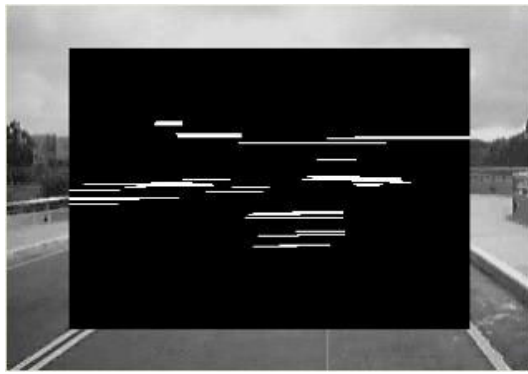


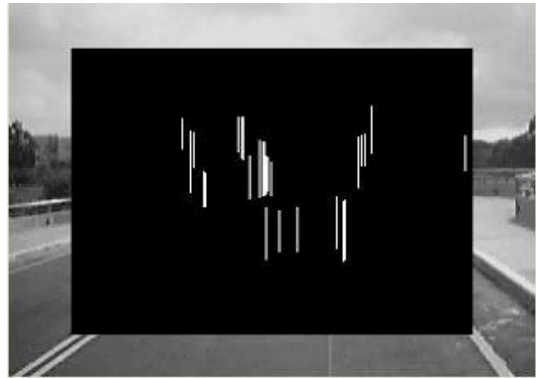
圖 2.2 二值化後的影像

#### b. 偵測處理步驟方面

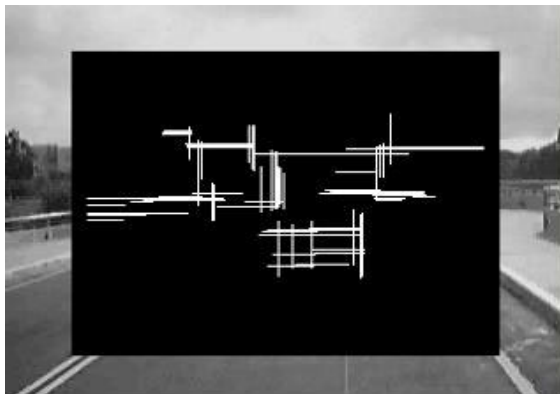
首先將原始圖作灰階處理後，再使用微分法抓取邊界(Sobel 遮罩)、再作二值化其影像如圖 2.2 所示，並由於一般車輛的外形在水平邊有『|\_|』形狀的特徵，垂直邊有『|\_| or |\_|』形狀的特徵，利用此特徵將水平邊及垂直邊抽取出來，可得到車輛大約的輪廓(如圖 2.3(a)(b)(c)所示)。之後，將所得的灰階影像、邊的影像、水平邊的影像及垂直邊的影像，由 contrast function ( intensity function ) 找出各特定點灰階、邊、水平邊及垂直邊其對稱值總和，選取對稱值最高其對稱軸，及其對稱寬度，如圖 2.3(d)所示。



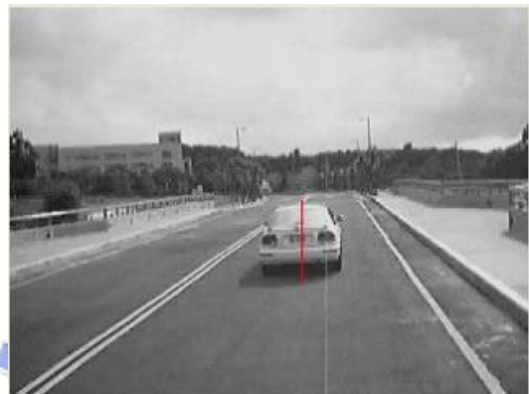
(a)水平邊抓取後的影像



(b)垂直邊抓取後的影像



(c)車輛輪廓抓取



(d)選取對稱值最高其對稱軸

圖 2.3

由所得的對稱軸、對稱寬度及車輛輪廓，從對稱軸所在位置的周圍，左右搜尋車輛輪廓，是否符合此『 $\square$ 』形狀特徵，若符合即為前方車輛，如圖 2.4 所示，若不符合再選取次高對稱值其對稱軸，再作另一次比對。但一般為了保險起見，在對稱軸的選取時，會由所獲得的車道線資訊，提供選取對稱軸依據，並會找出數個可能的前方車輛，再由車道線資訊，來選取最有可能的前方車輛。



圖 2.4 偵測前方影像的車輛位置座標

在獲得前方車輛後，以此位置向外小幅擴張，定義一偵測範圍，為下張影像偵測區域，可大幅減少處理速度，而當無法偵測或偵測錯誤時，再將偵測範圍放大，作較廣域的偵測。其程式流程如圖 2.5 所示，其各偵測結果如圖 2.6 所示。

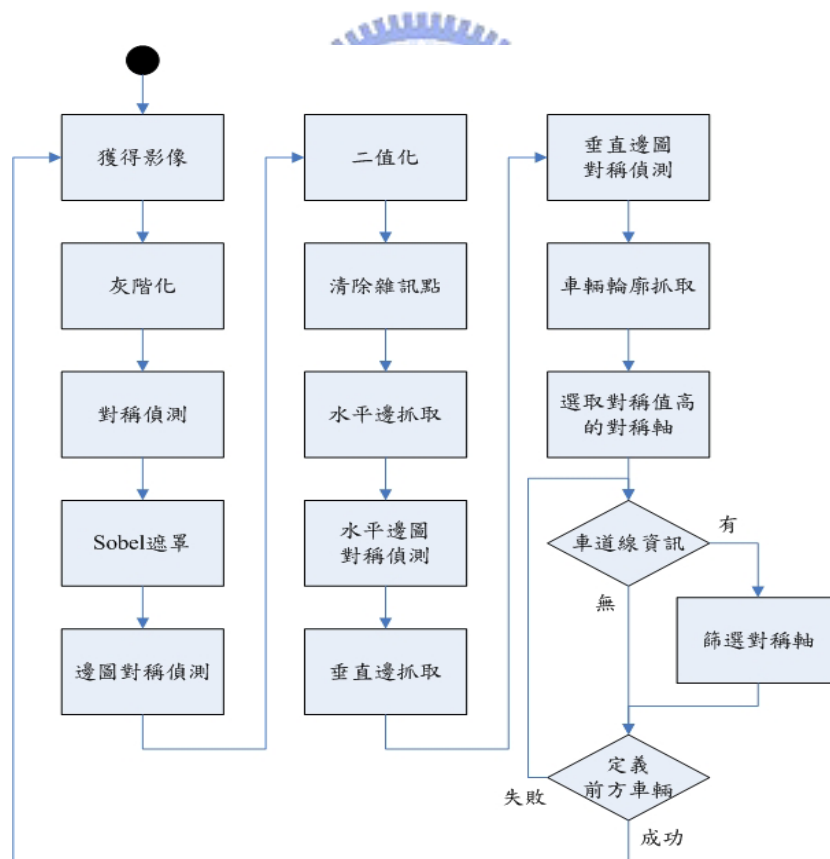


圖 2.5 以對稱特性偵測前方車輛之方法其程式流程

以對稱特性偵測前方車輛之方法，其處理速度約  $70ms \sim 500ms$  (以  $320*240$  pixel 在 AMD CPU 1.1GHZ 為例) 依 detection box 大小而定，其辨識率仍可接受，但仍有改善的空間，因此引入代理人的觀念，利用透過雷射掃瞄作輔助提高其辨識率，降低處理速度。而此方法也提供在 PC based 架構下，即時偵測前方車輛其相對應的解決方案。



圖 2.6 以對稱特性偵測前方車輛之方法其偵測結果

### 2.1.2 IPM based 之車道線偵測方法

由於透視效果( perspective effect )關係，一般從攝影機所獲取的道路影像，其車道線寬度會隨著與攝影機的距離而有所改變，而每個像素所代表的實際大小也不同，因此造成原本平行的車道線，在圖像上會形成非平行而相交在遠處兩直線，造成不容易從圖片中找出車道線，使用 IPM(inverse perspective mapping)方法，將透視效果的影響從圖片中移除，形成等比例的俯視圖( top view )，讓之後車道線偵測更容易，先定義所需物理參數如下：

$W = \{(x, y, z)\} \in E^3$ ，表示 3-D world space。

$I = \{(u, v)\} \in E^2$ ，表示 2-D image space(如圖 2.7 所示)。

(1) Viewpoint : 攝影機的位置  $C = (l, d, h) \in W$  。

(2) Viewing direction : optical axis  $\hat{o}$  由下述兩個角度來決定

$\bar{\gamma}$  : 為 optical axis  $\hat{o}$  投影到  $z=0$  平面(定義為向量  $\hat{\eta}$ )，與  $x$  軸之夾角(如圖 2.8 所示)。

$\bar{\theta}$  : 為 optical axis  $\hat{o}$  與向量  $\hat{\eta}$  之夾角(如圖 2.9 所示)。

(3) Aperture : camera angular aperture is  $2\alpha$  。

(4) Resolution : camera resolution is  $n \times n$  。

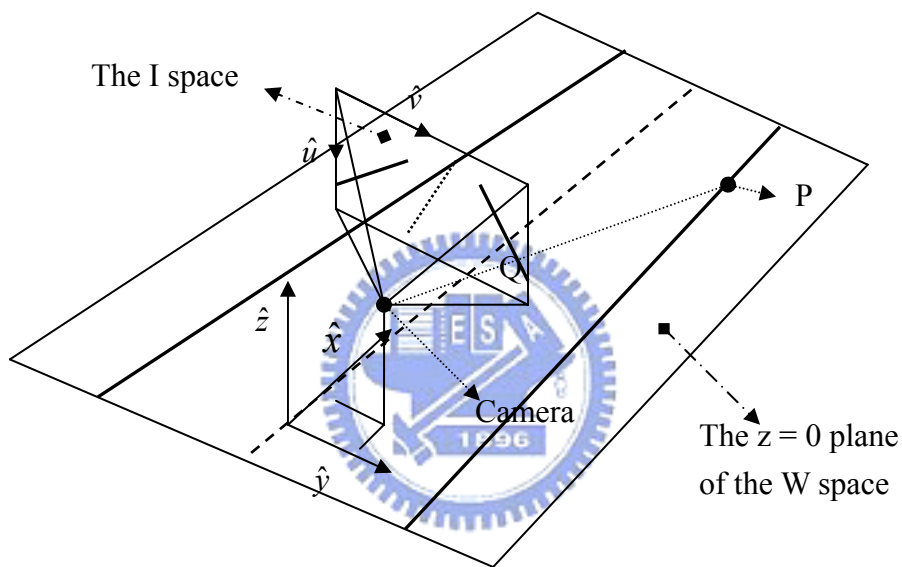


圖 2.7 兩個座標系( $W$  and  $I$ )的關係

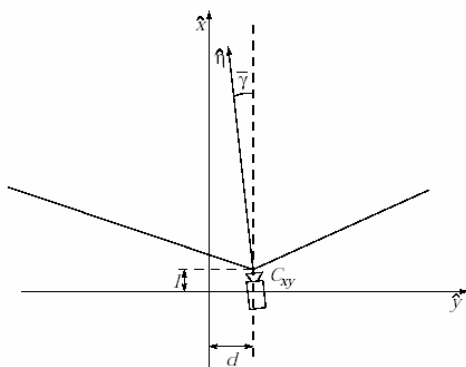


圖 2.8  $xy$  plan in the  $W$

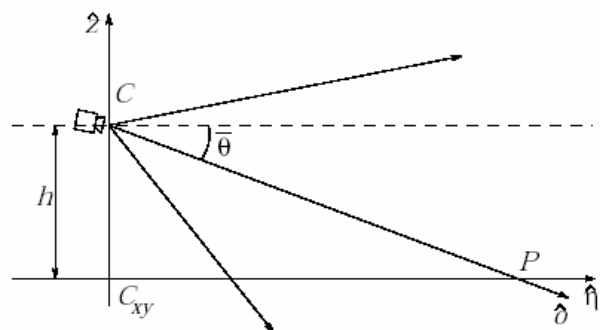


圖 2.9  $z\eta$  plane in the  $W$

利用(2-6)式及(2-7)式( $z=0$ )對應到世界座標上實際的位置，再依世界座標上實際的位置，由俯視方向等比例重新應對回圖像座標，重新分配其對應的灰階值，將透視影響移除，其結果如圖 2.10 所示。

$$x(u, v) = h \times \cot[(\bar{\theta} - \alpha) + u \frac{2\alpha}{n-1}] \times \cos[(\bar{r} - \alpha) + v \frac{2\alpha}{n-1}] + l \quad (2-6)$$

$$y(u, v) = h \times \cot[(\bar{\theta} - \alpha) + u \frac{2\alpha}{n-1}] \times \sin[(\bar{r} - \alpha) + v \frac{2\alpha}{n-1}] + d \quad (2-7)$$

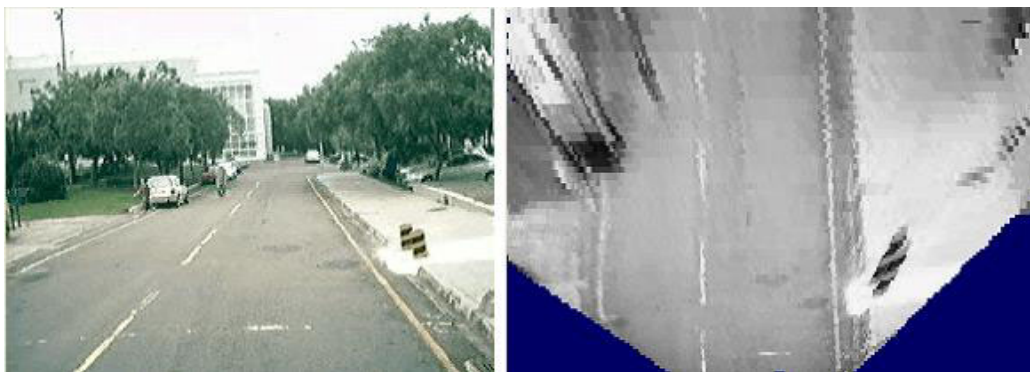


圖 2.10 (a)原始圖 (b) IPM 後的結果

此方法除了可以將透視影響移除，形成俯視圖，並可以將部份不屬於車道的影像移除，讓影像集中在車道上，減少在車道線擷取時受到其周圍影像影響。接著將影像利用灰階值差異將車道線特徵擷取出來，再採用 GMD(geodesic morphological dilation)方法加強車道線的影像，最後局部二值化和線條細化處理後得一黑白影像完成車道線截取動作(圖 2.11)。在定義車道線方面我們採用的方法是利用影像中 CCD 位置與車道線的相對關係、車道線寬度限制等得到車道線位置，最後可再利用(2-8)式將找出的車道線重新對應回原始圖上(如圖 2.12)。

$$u(x, y, 0) = \frac{\arctan\left[\frac{h \sin \gamma(x, y, 0)}{y - d}\right] - (\bar{\theta} - \alpha)}{\frac{2\alpha}{n - 1}}$$

$$v(x, y, 0) = \frac{\arctan\left[\frac{y - d}{x - l}\right] - (\bar{\gamma} - \alpha)}{\frac{2\alpha}{n - 1}} \quad (2-8)$$

$$\gamma(x, y, 0) = \arctan\left(\frac{y - d}{x - l}\right)$$



圖 2.11 細化處理後的影像

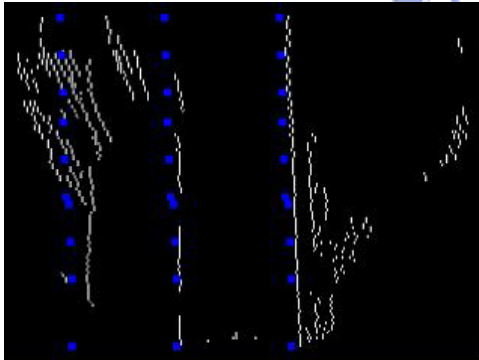


圖 2.12(a) 所找出的車道線



(b) 將車道線對應回原始圖上

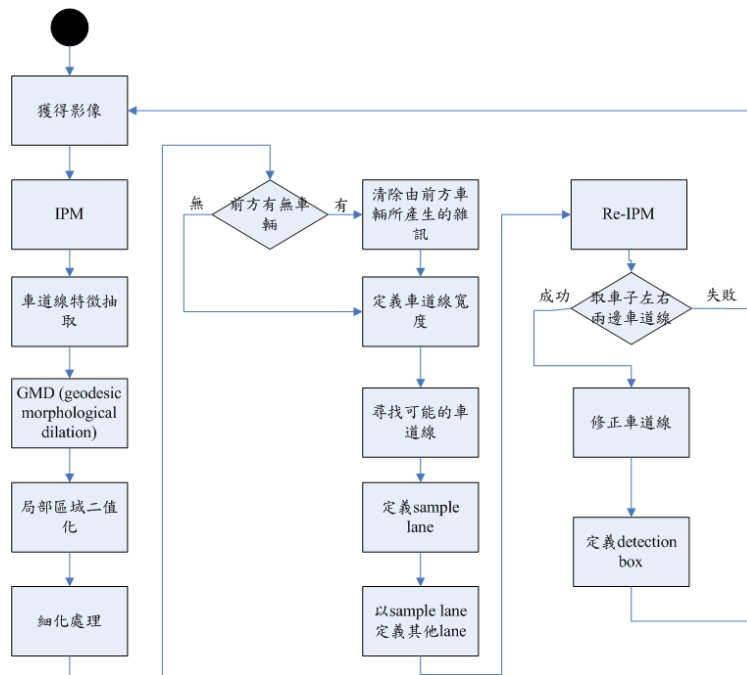


圖 2.13 IPM based 之車道線偵測方法其程式流程

其程式流程如圖 2.13 所示。IPM based 之車道線偵測方法，定義偵測視窗雖可達到不錯的效果，但所花的時間較長約需 500 ~ 600 ms(以 320\*240 pixel 在 AMD CPU 1.1GHZ 為例)，若以 PC based 來說是無法達到即時的要求，需與其它方法配合改善之，因此提出動態視窗之車道線偵測方法與之配合。

### 2.1.3 整合型車道線偵測方法

透過 IPM based 之車道線偵測方法可得偵測視窗，定義出此四個小視窗(如圖 2.14)後可進入”動態視窗之車道線偵測方法”，此法不僅去除了絕大多數我們不需要的影像，且減少了處理的資料量。其流程圖如圖 2.15 所示，在動態視窗上進行車道線擷取與定義處理，經灰階化、車道線特徵抽取、二值化、型態擴張、細化後，接著定義出車道線直線方程式，最後繪於原始圖上並微調其動態視窗位置以提供下一張影像利用(如圖 2.16)。

此法的處理量少，經測試結果其處理速度大約 60 ms 以下 (以 AMD CPU 1.1GHZ 為例)，可達到即時的要求，而其辨識率在略差的影像下還有不錯的效果，但若太過模糊時仍無法辨識，且其動態的偵測視窗在車子變動太大時，會無



法跟上，造成無法辨識的結果，因此我們結合 IPM based 之車道線偵測方法以互補雙方的不足，也就是採用”整合型車道線偵測方法”。

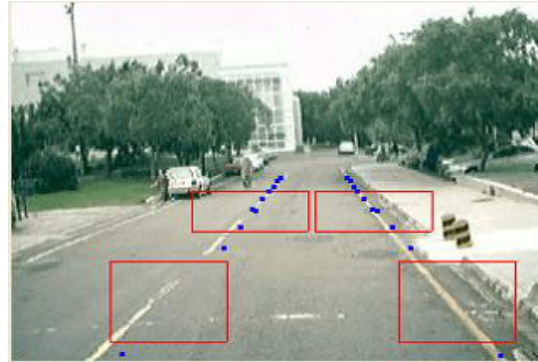


圖 2.14 所找出的偵測視窗

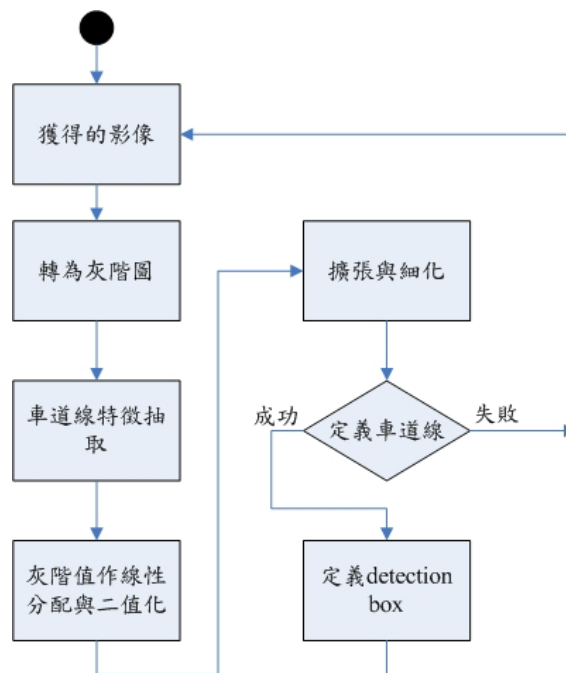


圖 2.15 動態視窗之車道線偵測方法其程式流程



圖 2.16 動態視窗之車道線偵測方法所定義出的車道線影像

整合型車道線偵測方法，主要是先利用 IPM based 之車道線偵測方法，找出車道線偵測視窗，再由動態視窗之車道線偵測方法，從所獲得偵測視窗對車道線作追跡動作，但若一直無法抓取到車道線，再切換回 IPM based 之車道線偵測方法。此作法是在取 IPM based 之車道線偵測方法其辨識率高，並較能適應多種狀況之優點，與動態視窗之車道線偵測方法其處理速度快的優點，來相互配合，達到我們的需求。其程式流程如圖 2.17 所示。

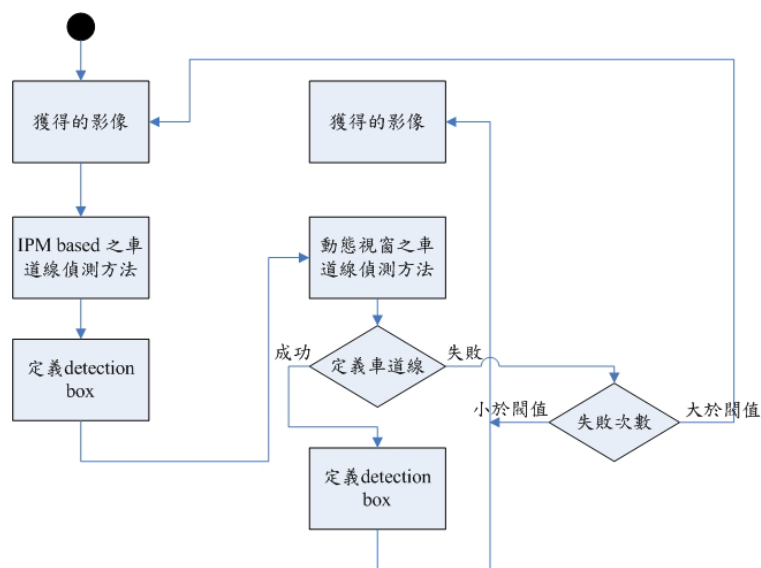


圖 2.17 整合型車道線偵測方法其程式流程

整合型車道線偵測方法，從我們所使用的狀況來看，已有不錯的辨識率及即時性(平均處理速度約 80 ms -- 以 AMD CPU 1.1GHZ 為例)，可達到我們的需求。一般在 PC based 架構下，在影像辨識方面要有較高的辨識率，通常就需要較長的處理時間，而要有較快的處理速度，一般而言其辨識率會大幅下降，因此整合型車道線偵測方法在這方面提供了一相對應的解決方案，讓在 PC based 下也可以擁有較高的辨識率與即時性。

## 2.2 夜間前方車輛偵測方法

在夜間車輛偵測方面，由於夜間 CCD 影像品質差，所以一直沒有一套有效的、適應於各種環境的方法提出。本論文提出一套較實用簡潔的偵測法則：透過尋找影像中的車尾燈訊息以偵測前方車輛，此外，並嘗試著驗證此法則於各種路況環境中的表現。此偵測法則包含二部份：首先是將輸入影像做前處理，以去除大部分的雜訊；再來是運用一系列特別發展的車輛判定(detection)的方法，以確定出前方車輛的所在位置。

### 2.2.1 前處理方法

當黑夜來臨，大地一切事物皆籠罩在黑暗中，對人眼或 CCD 影像來說，發光物及反射係數強的色彩，也就是車尾燈、車道線、路燈和道路標示牌等，會變的相對明顯。根據我們的目的-尋找前車車尾燈，故影像中非車尾燈之物皆屬於雜訊。於是，在此步驟中，我們必須使用一些影像處理的方法將雜訊去除。

一般去除雜訊的方式多用二值化方法，也就是先將彩色影像取灰階，再取一固定閾值進行二值化處理，此二值化處理會將閾值以下的圖素設為黑色，閾值以上的設為白色，最後得到的是一黑白影像。如圖 2.18 所示，二值化閾值皆取 200 的情況下，雖然已經能將大部份雜訊濾除，但對於亮度高的車道線雜訊卻不盡理想，一再的調高二值化閾值並不是好的解決方法，這是由於灰階影像是明亮度的

表現，當雜訊的明亮度相等於車尾燈的明亮度時，此種方法便會失效，因此，我們必須更改去除雜訊的程序。

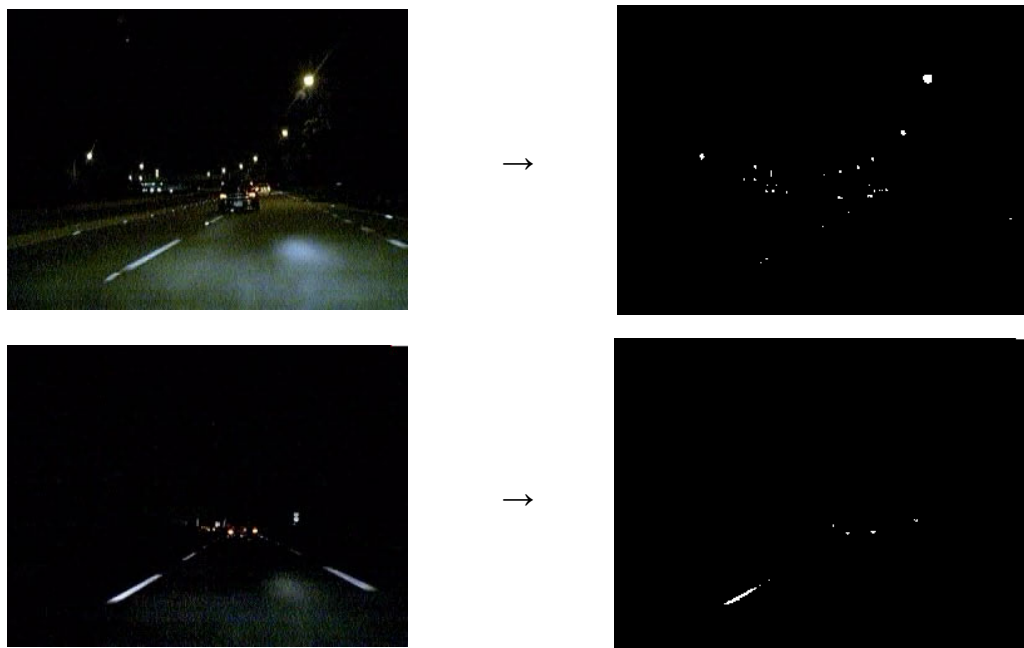


圖 2.18 彩色影像經灰階、二值化(閾值=200)處理後的結果

前處理的程序更改為色度座標轉換(color-space transform)以及條件式二值化。在此我們所進行的色度座標轉換是將原彩色影像 RGB 座標依(2-9)的式子轉換至 YCrCb 座標。

$$\begin{bmatrix} Y \\ Cr \\ Cb \end{bmatrix} = \begin{bmatrix} 0.257 & 0.504 & 0.098 \\ 0.439 & -0.368 & -0.071 \\ -0.148 & -0.291 & 0.439 \end{bmatrix} \begin{bmatrix} R \\ G \\ B \end{bmatrix} + \begin{bmatrix} 16 \\ 128 \\ 128 \end{bmatrix} \quad (2-9)$$

表 2.1 色度座標轉換 RGB→YCrCb

顏色名稱	(R G B)	→	(Y Cr Cb)
紅(red)	(191 27 75)		(86 196 125)
白(white)	(255 255 255)		(235 128 128)
黑(black)	(0 0 0)		(16 128 128)

表 2.1 顯示數個我們較關心的代表色之轉換結果。明顯地，相對於白色與黑

色，紅色具有較高的 Cr 值，因此，我們可單就 Cr 值進行二值化的處理，以期將雜訊(白色車道線)濾除。經由實驗結果，由於車尾燈並非只是單純的紅色圖素，而是摻有亮度的光點，經轉換後發現 Y 值會偏高，所以如再將 Y 值加入二值化條件效果會更好。最後，我們採用的條件式二值化如(2-10)式所描述，並由實驗中調整其閾值為  $Y=145$ ， $Cr=130$ 。

$$r(i, j) = \begin{cases} 255, & \text{if } r_Y(i, j) \geq 145 \wedge r_{Cr}(i, j) \geq 130 \\ 0, & \text{otherwise} \end{cases} \quad (2-10)$$

經由色度座標轉換以及條件式二值化處理後的結果如圖 2.19 所示。由圖中可清楚發現，地面上的車道線被完全的濾除，只留下車尾燈的光點圖像，效果非常不錯。另外說明的是，因為我們對路況的假設是平坦的，所以影像的上半部份是不可能存在車輛的，所以之後的輸入影像皆會事先用黑色圖像覆蓋上半部。



圖 2.19 彩色影像經色度座標轉換、條件式二值化

(閾值為  $Y=145$ ， $Cr=130$ )處理後的結果

### 2.2.2 車輛判定方法

至此所得影像為一黑白影像，而其中白色圖像有非常大的可能就是車體所在，於是接下來需要做的就是車輛判定(detection)。車輛判定的方法有二：首先是尋找單一光點，將影像中聚集的光點分類，並計算其質心所在；第二是尋找成對光點，此時要利用下一刻的輸入影像，比較同區域光點質心的移動方式，具有相同高度及移動方式的光點即為同一群組，也就是同一車體。於是，經由以上判定方式找到車體所在，並且經由監控畫面標示出來。

#### a.尋找單一光點：

將所有白色區域找出後，再細部分類成許多區塊，同時計算區塊內光點質心，以備下個步驟使用。圖 2.20 表示經由此步驟，找出了三個光點區塊(底線標示處)。



圖 2.20 經“尋找單一光點”後的結果

#### b.尋找成對光點

取得下一刻輸入影像後，對於此兩張連續影像的資訊，針對同質心高度的區塊，比較其區塊內光點(質心)移動的趨勢，藉以確定那些區塊為同一群組，因而可以做到車輛的判定(如圖 2.21 所示)。(a)表示的是在前一步驟(尋找單一光點)中所得的影像，假設已經找到二個光點區塊並計算好各光點質心位置。(b)再輸入下一刻影像，淺色是上一張光點位置，深色是此張的光點位置。同樣的，計算出其質心位置，並且計算出與上一張質心的位移量 $\Delta x1$ ,  $\Delta y1$ ,  $\Delta x2$ ,  $\Delta y2$ 。(c)如

果 $(\Delta x1-\Delta x2) + (\Delta y1-\Delta y2) < \text{閾值}$ ，我們可確定區塊 1(block1) 和 區塊 2(block2) 是成對的。(d)根據圖 2.3 所得資訊，實作於真實影像中，判定出前方有一輛車存在，並經由底線標示出。

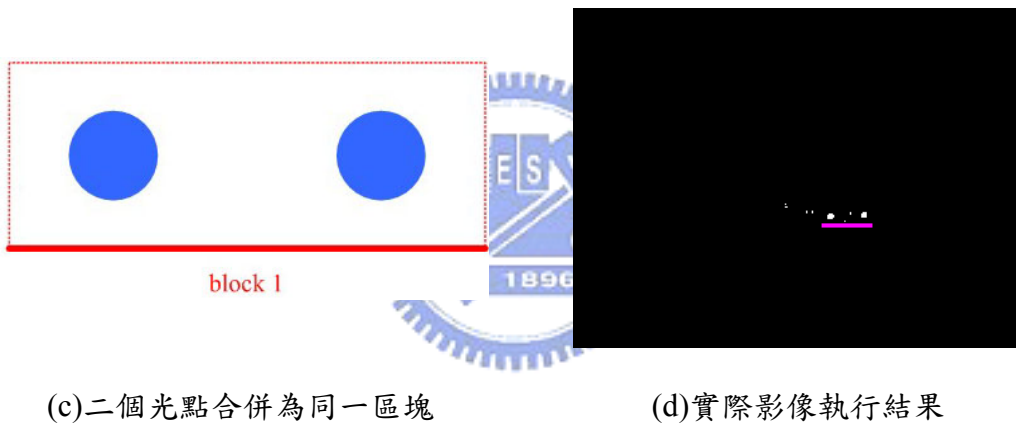
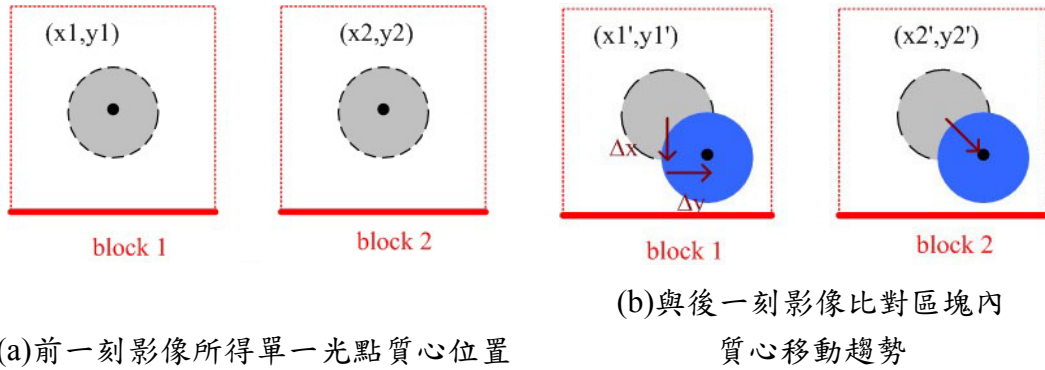


圖 2.21 “尋找成對光點”步驟圖解以及實際執行結果

綜和以上過程得知，要判定車輛的存在最少要處理二張影像，如圖 2.21(a)、(b)，根據實驗結果，對於事先錄製的前方有車之影像檔，判定一輛車平均需要 140ms(以 320×240 的影像大小在 P4 CPU 2.4GHz 為例)。然而對於已經找到的車輛，其實不必一再地重新搜尋，只要微調其影像中的相對位置即可。關於這個想法，於是我們在之後加入了追蹤的處理，只需一張輸入影像便可達到，不僅可以縮短將近三分之一的程式處理時間，也可減少一些不必要錯誤的產生。

## 2.3 夜間前方車輛追蹤方法

此處我們採用樣本比對(pattern matching)來進行車輛的追蹤(tracking)。首先大致介紹樣本比對的觀念，透過計算影像間的相關性(correlation)做樣本比對是很普遍的作法，雖然準確性高，然而有運算量大，費時等缺點。在此本研究改進的方法是在局部區域進行比對以減少運算時間，而局部區域即是指在車輛偵測步驟中找到的車輛位置。

以下是相關性(correlation)理論的介紹[26]：

為了方便說明，假設二個連續函數  $f(x)$  和  $g(x)$ ，其相關性計為  $f(x) \circ g(x)$ ，並利用(2-11)式的相關性定義。

$$f(x) \circ g(x) = \int_{-\infty}^{\infty} f^*(\alpha)g(x+\alpha)d\alpha \quad (2-11)$$

式中的符號 \* 表示複數共軛，且由於我們實際處理的是影像函數，所以  $f^*(x) = f(x)$ 。圖 2.22(a)~(f)圖解此一程序，其中陰影區域表示乘積不為零的區域。

因此，在我們的情況中，使用的式子是如(2-12)所示之離散相關式， $f(x,y)$  即為樣本圖像，大小是  $M \times N$ ； $g(x,y)$  則為輸入影像圖像。

$$f(x,y) \circ g(x,y) = \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} f(m,n)g(x+m,y+n) \quad (2-12)$$



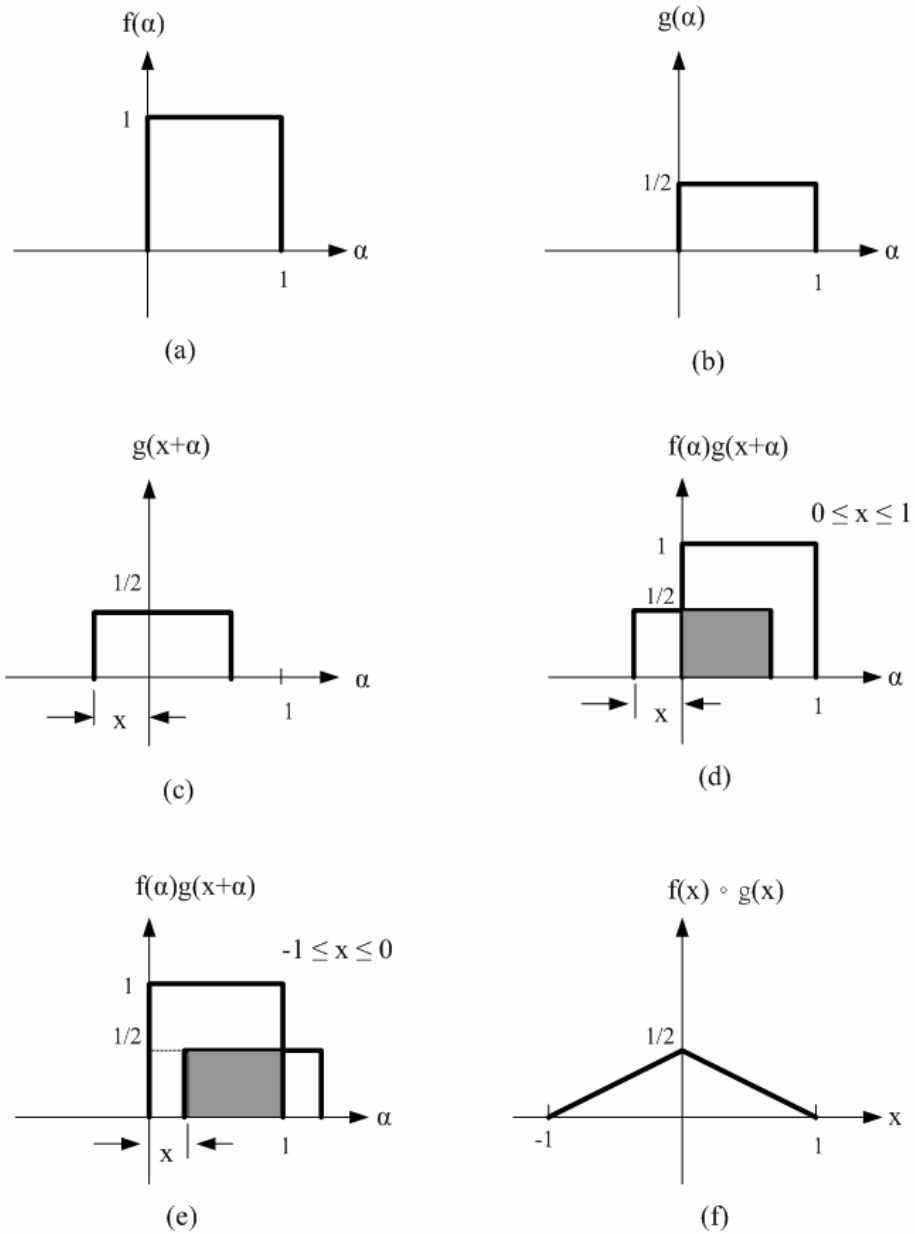
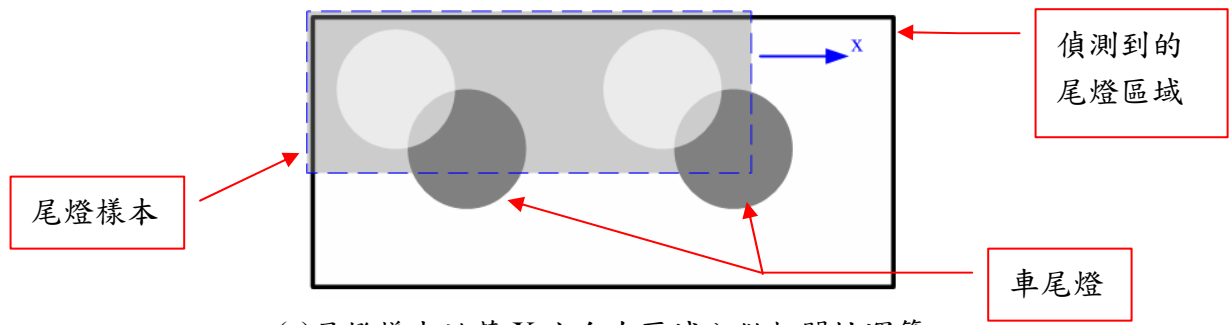


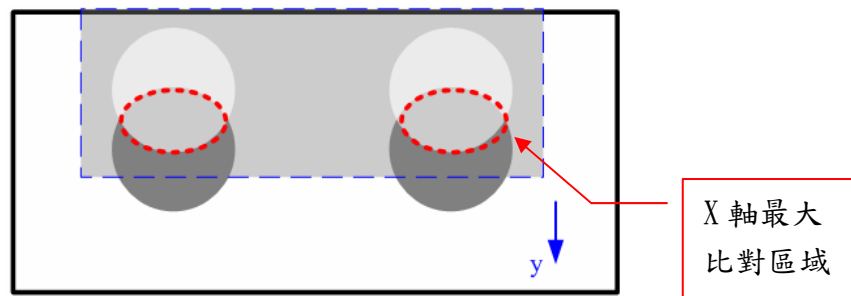
圖 2.22 相關(correlation)的圖解

實作於真實的影像中時，我們將前一刻所尋得的車輛圖像複製下來作為樣本，在下一刻輸入影像的局部區域做樣本比對計算，具有最大相關性之處即為此張影像中同一部車輛的新位置。

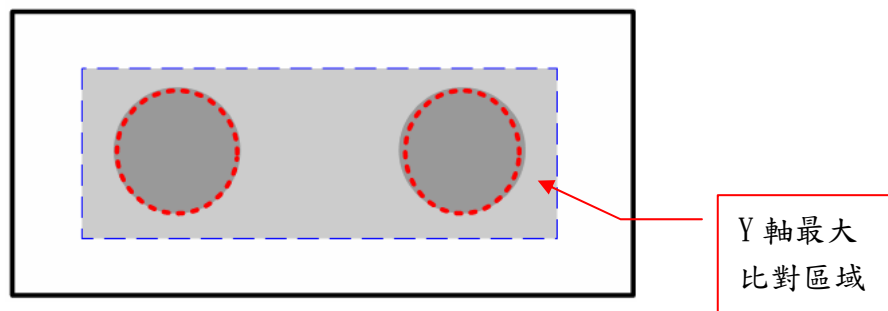
車輛的樣本比對程序如圖 2.23 所示，圖(a)：將前一張影像所得的尾燈樣本置於局部區域的左上角，沿著方向做比對。圖(b)：得 X 軸方向上最大的相關值後，在此 X 值上延著 Y 方向找最大相關值。圖(c)：得此區域之最大相關處，結束追蹤程序，藉以完成樣本的確定，以及追蹤位置的微調。



(a)尾燈樣本沿著 X 方向在區域內做相關性運算



(b)找到 X 方向最大相關後，再延 Y 方向尋找



(c)找到區域內的最大相關位置，完成樣本比對處理

圖 2.23 追蹤步驟圖解

圖 2.24 顯示程式執行在一事先錄好的影像檔上，除了展現出持續追蹤同一部車輛的能力外，程式並且可以搜尋新出現的車輛(如圖(c)和(f))，一起加入追蹤的行列中。依追蹤車輛數目的不同，其追蹤一次的速度約 90ms~100ms(以 320×240 的影像大小在 P4 CPU 2.4GHz 為例)，採用第 27、470、667、1250、1953、2208 張影像。

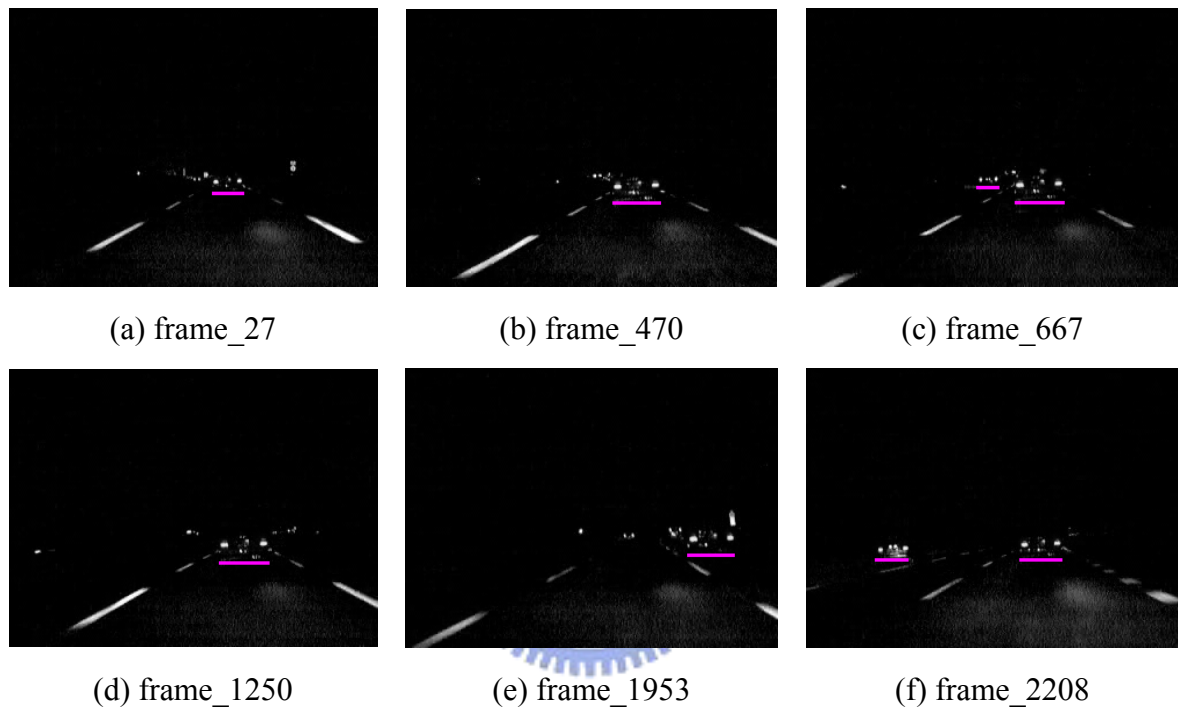


圖 2.24 追蹤處理展示(frame\_27 ~ frame\_2208)

於是，前方車輛的程式流程可整理如圖 2.25 所示：用來執行車輛判定 (detection) 的輸入影像皆會經過“前處理”的步驟，而車輛追蹤(tracking)的步驟用的輸入影像則只單單經灰階處理，原因是為了確保樣本比對的品質。追蹤步驟執行數次後，會再次重頭進行車輛偵測的步驟，尋找影像中新出現的車尾燈訊息，以保持輔助系統的品質。目前程式設計可同時追蹤五輛前方車輛。圖 2.26 顯示在不同路況的偵測結果。

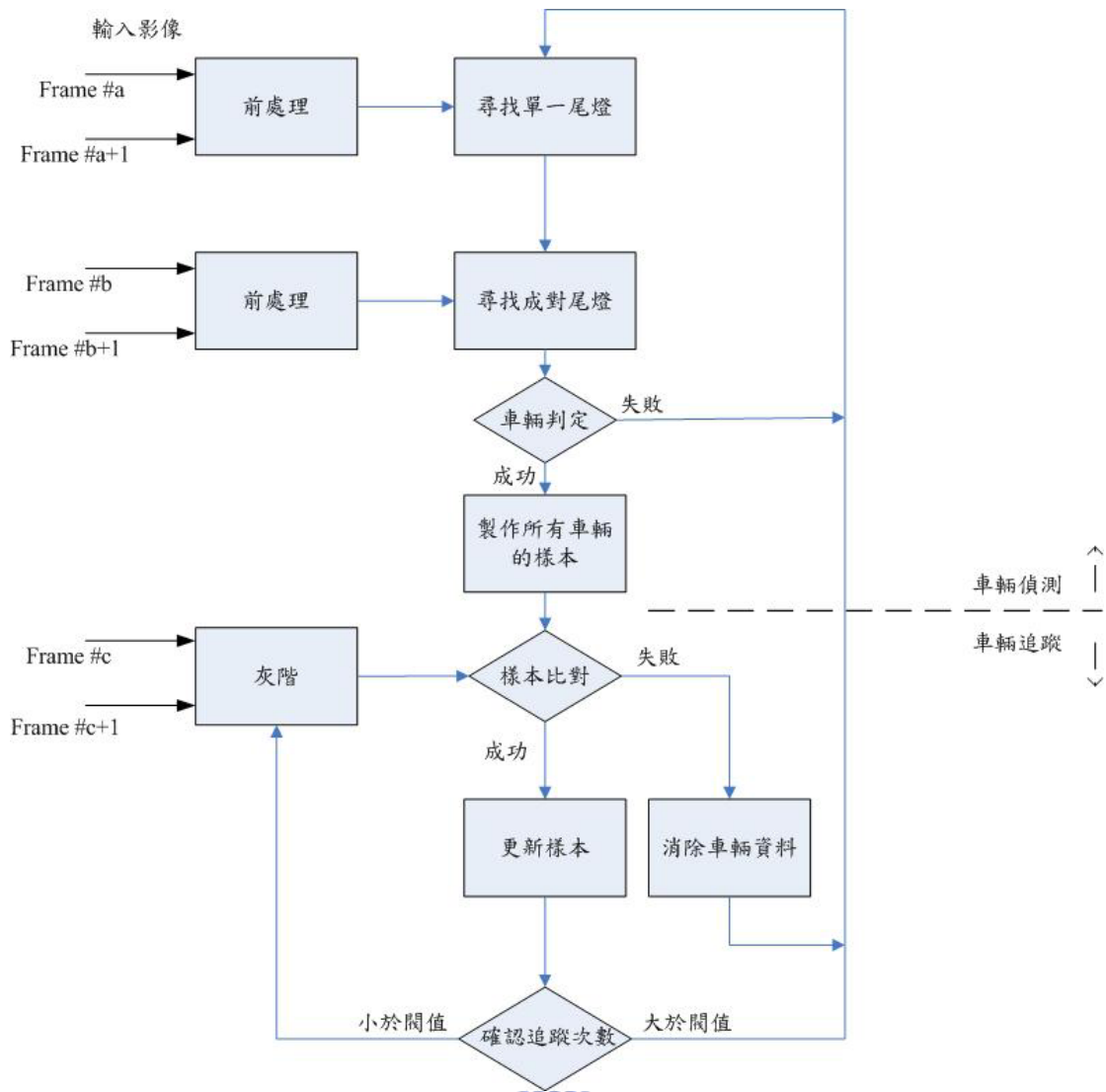


圖 2.25 夜間前方車輛偵測與追蹤流程



圖 2.26 在不同的環境偵測及追蹤前方車輛之結果

## 2.4 夜間車道線偵測方法

在夜間車道線偵測方面，我們主要還是採用以 IPM 為基礎的整合型車道線偵測法。然而，由於影像明亮度與白天有差，所以我們必須作些微的修改以適用於夜間情況。在車道線截取步驟中，我們增加特徵抽取的閾值，這是因為夜間影像中，路面與車道線顏色較接近，且路面因路燈、車前燈等的照射下造成顏色不均，此時必須將閾值調大才能去除路面雜訊而留下正確的車道線(如圖 2.27 所示)。因此程式中，白天的閾值為 3，而夜間的閾值調為 6。

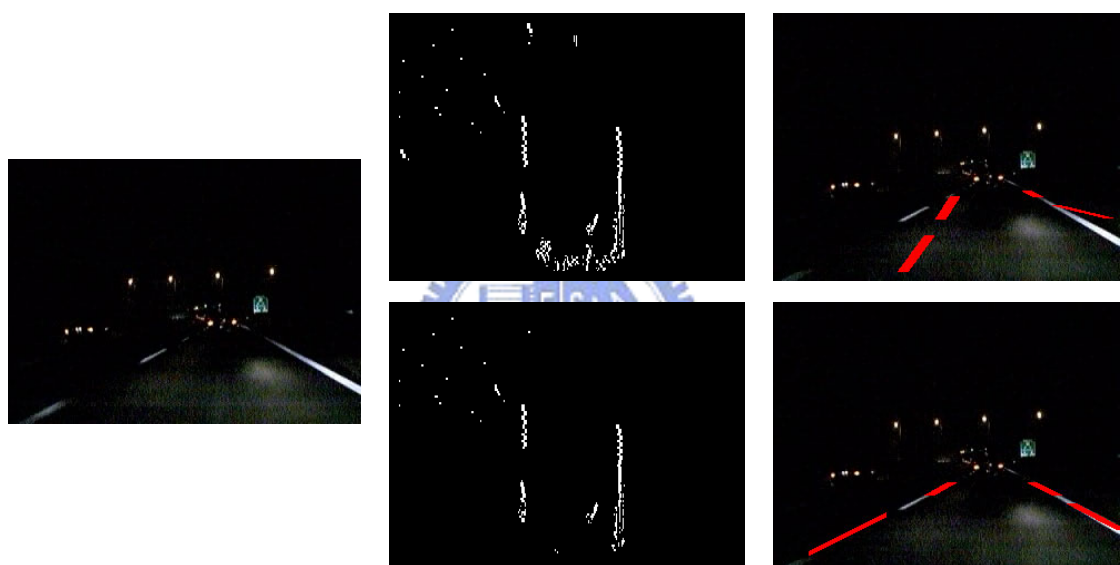


圖 2.27 不同的閾值所得之車道線偵測結果

圖 2.28 顯示一些日夜和不同天候狀況影像之車道線偵測結果，在一般白天的影像中，對於彎曲的車道線也很不錯的效果，如圖(c)所示，實驗發現，IPM based 的方法雖不能很準確的抓到車道線，但只要車體不搖晃過大，可透過動態視窗的偵測方法微調後抓到。在一般夜間影像中，只要車道線標示明確，即使有對方來車車頭燈的影響，皆有不錯的偵測結果，如圖(d)(e)，而對於圖(f)來說，路面上的標示，不論是白天或夜晚，仍是需待解決的狀況。其他天候影像中，我們發現黃昏狀況(圖(g))的偵測最容易受到駕駛人車頭燈的影響，當天色還不夠黑，車道線特徵擷取的閾值太小會造成錯誤的偵測，此時本系統透過

多代理人間的合作來解決之，詳細說明在 4.3.1 小節；雨天路況，如圖(h)所示，也是我們嘗試狀況之一，當路面的雨水過多，其水漬的反光容易造成偵測失敗，而目前還沒有一個好的解決方式，故此偵測法則仍有改善的空間。

一般白天影像：



(a)無車



(b)有車



(c)彎道

一般夜間影像：



(d)有車



(e)彎道



(f)路面上有路標

其他天候影像：



(g)黃昏



(h)雨後路面

圖 2.28 各種路況之車道線偵測結果

## 2.5 白天與夜間影像偵測系統之整合

我們個別完成了白天與夜間的駕駛輔助系統，如今更進一步要做的是如何順暢並準確的切換二種演算法。根據本系統白天與夜間前方車輛的偵測法則，白天方面是使用對稱法則，其輸入影像必須清楚的顯示車輛外型；而夜間方面是尋找車輛尾燈，其輸入影像必須偏暗，如此車尾燈才會相對明顯。因此，切換的法則必須依據的是影像明亮度。

對於影像明亮度有許多的定義，最簡單的便是將整張影像做一灰階值的平均，也就是計算影像的平均明亮度(average luminance)，然而只憑單一數據就下判斷容易造成錯誤的發生，可是嚴謹又複雜的判斷法則無法達到即時的要求，不適用於我們的系統。於是，在此提出利用 Otsu 演算法為基礎的判斷法則。

### 2.5.1 Otsu 演算法[23]

Otsu 當初提出此方法是為了要將影像中的前景與背景分離，利用的是統計學上的判別分析法(discriminant analysis)。假設一張灰階影像有  $L$  個灰階色(graylevels)，灰階色  $i$  共有  $n_i$  個 pixels，此張影像共有  $N$  個 pixels。則其灰階直方圖(graylevel histogram)可如(2-13)式所示表示成一機率分布方程式。

$$p_i = n_i / N, \quad p_i \geq 0, \quad \sum_{i=1}^L p_i = 1 \quad (2-13)$$

在我們的例子中主要要將影像分為二類，因此必須先定義以下參數，類別事件機率(probabilities of class occurrence)可寫成(2-14)式、(2-15)式的類別期望值等級(class mean levels)，其中  $\omega(k)$  為零階動差、 $\mu(k)$  為一階動差、 $\mu_T$  為總期望值等級(total mean level)，定義如(2-16)式。

$$\omega_0 = \Pr(C_0) = \sum_{i=1}^k p_i = \omega(k) \quad (2-14)$$

$$\omega_1 = \Pr(C_1) = \sum_{i=k+1}^L p_i = 1 - \omega(k)$$

$$\mu_0 = \sum_{i=1}^k i \cdot \Pr(i | C_0) = \sum_{i=1}^k i \cdot p_i / \omega_0 = \mu(k) / \omega(k) \quad (2-15)$$

$$\mu_1 = \sum_{i=k+1}^L i \cdot \Pr(i | C_1) = \sum_{i=k+1}^L i \cdot p_i / \omega_1 = \frac{\mu_T - \mu(k)}{1 - \omega(k)}$$

$$\omega(k) = \sum_{i=1}^k p_i, \quad \mu(k) = \sum_{i=1}^k i \cdot p_i, \quad \mu_T = \mu(L) = \sum_{i=1}^L i \cdot p_i \quad (2-16)$$

為了評估何值為”好的”閾值，也就是 k 值，我們利用在判別分析法中的 discriminant criterion measures (或稱 measure of class separability)，此法可透過  $\lambda$ 、 $\kappa$ 、和  $\eta$  三種指標找出最佳閾值，其指標定義如(2-17)式所示。

$$\lambda = \sigma_B^2 / \sigma_W^2, \quad \kappa = \sigma_T^2 / \sigma_W^2, \quad \eta = \sigma_B^2 / \sigma_T^2 \quad (2-17)$$

其中  $\sigma_W^2$ 、 $\sigma_B^2$ 、和  $\sigma_T^2$  的物理意義是 within-class variance、between-class variance、以及 the total variance of levels，如(2-18)式。

$$\begin{aligned} \sigma_W^2 &= \omega_0 \sigma_0^2 + \omega_1 \sigma_1^2 \\ \sigma_B^2 &= \omega_0 (\mu_0 - \mu_T)^2 + \omega_1 (\mu_1 - \mu_T)^2 \\ &= \omega_0 \omega_1 (\mu_1 - \mu_0)^2 \\ \sigma_T^2 &= \sum_{i=1}^L (i - \mu_T)^2 \cdot p_i \end{aligned} \quad (2-18)$$

由(2-12)和(2-13)式可發現，在  $\sigma_T^2$  是已知數的情況下， $\sigma_W^2$  式是有關於 class variance 的二階統計方程，而  $\sigma_B^2$  是有關於類別期望值(class means)的一階統計方程，所以透過指標  $\eta$  來尋找最佳閾值 k 是最簡單的方法。因此，經由尋找  $\eta$  的最大值，也就是如(2.19)式所寫的，使得  $\sigma_B^2$  最大的 k 值便是最佳值。



$$\sigma_B^2(k) = \frac{[\mu_T \cdot \omega(k) - \mu(k)]^2}{\omega(k)[1 - \omega(k)]}$$

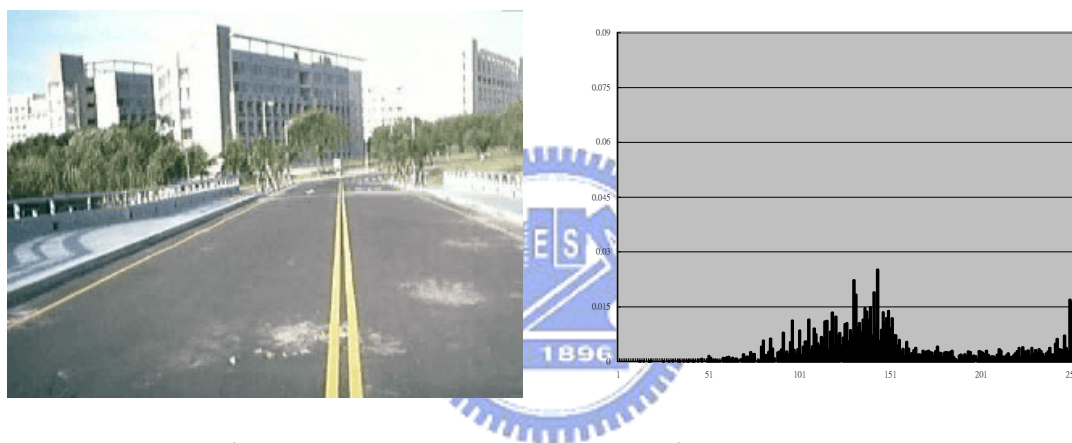
$$\sigma_B^2(k^*) = \max_{1 \leq k < L} \sigma_B^2(k) \quad (2.19)$$

當然，此種演算法還可將影像的灰階直方圖分做三類，甚至多類，但不在本研究的應用中，故在此便不再多加討論。



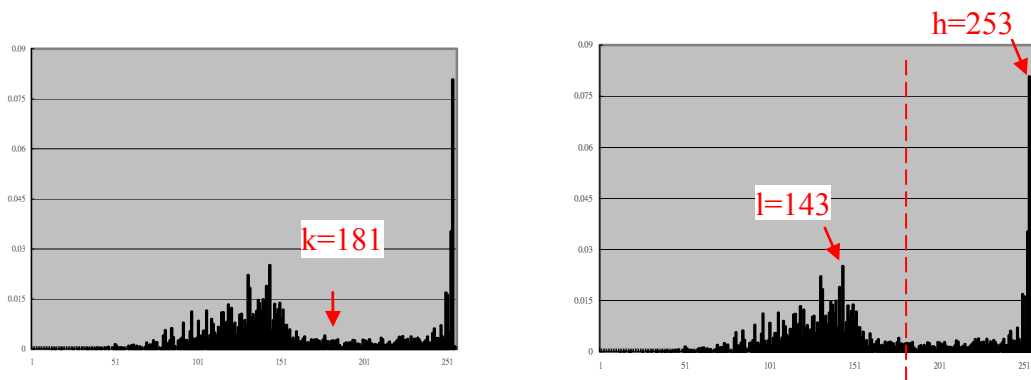
## 2.5.2 日夜演算法及實驗結果

將 Otsu 演算法應用於日夜決策部份，處理步驟如圖 2.29 所示，先將輸入影像(圖(a))灰階化後進行直方圖分析，所得如圖(b)所示，再將其數值做判別分析法取得一最佳閾值  $k$  (如(2.19)式)，如圖(c)箭頭所示， $k=181$ 。接著以灰階值等於 181 為分界，將直方圖分為二類：亮區(右)與暗區(左)，並在各區中尋找區域最大值： $h$  與  $l$ ，圖示於(d)圖， $h=253$ 、 $l=143$ 。故亮區與暗區的差異量可用  $h-l$  計算之，其值為 110。此差異量便是我們判斷白天與夜晚的依據，稱之為 Otsu 明亮度。



(a) 輸入影像

(b) 輸入影像之直方圖分析結果



(c) 由 Otsu 演算法所得之最佳閾值  $k$

(d) 亮區與暗區的區域極值

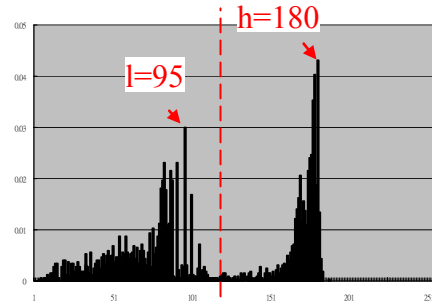
圖 2.29 Otsu 演算法實驗結果

接著我們必須定出 Otsu 明亮度的閾值以準確的判斷日夜狀況，經由圖 2.30

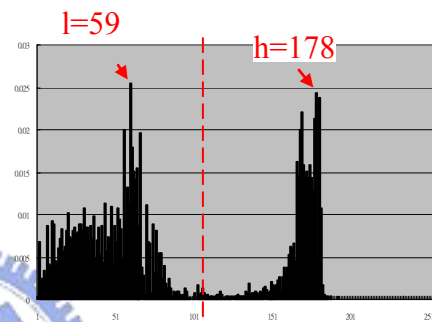
一連串的實驗結果，將閾值定為 80，大於此閾值為白天影像，否則為夜間影像。



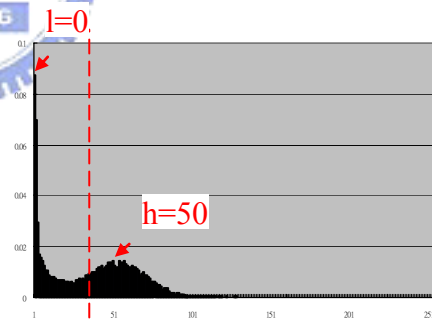
白天影像  
Otsu 明亮度=85



白天影像  
Otsu 明亮度=119



夜間影像  
Otsu 明亮度=50



夜間影像  
Otsu 明亮度=45

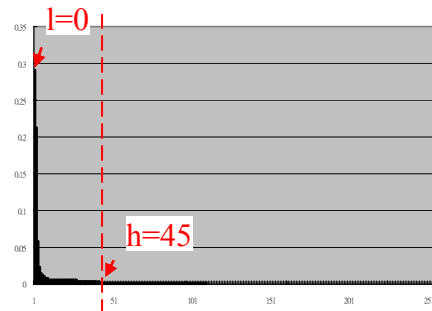


圖 2.30 不同日夜狀況的 Otsu 明亮度及其直方圖分析

白天的 Otsu 明亮度大，夜間的小是非常自然的現象，然而，我們發現夜間的影像並非全都是遵照此現象，如圖 2.31 所示，當影像中有特別明亮的光點，如路燈、對方來車之頭燈、以及過於明亮的車道線等等皆會造成 Otsu 明亮度過大，而導致系統誤判。解決方法是除了判斷 Otsu 明亮度外再加上平均明亮度的依據。經由實驗結果發現，Otsu 明亮度對於白天影像具有較佳的準確率，而平均明亮度則較適於夜間影像的判斷。在此，本系統將平均明亮度閾值設為 80，Otsu 明亮度閾值設為 80，並且每 10 分鐘便會進行一次偵測，以確保掌握即時的天候狀態。

儘管此法則對於白天與夜間影像的判斷已經具有相當準確的辨識率，然而黃昏及清晨的影像仍是極需加強的時段，這不僅嚴重的考驗閾值的選擇，對於偵測系統的演算法也極具挑戰性。之後，會針對日夜切換法利用代理人的溝通進行補救，將會在第 4.3 節說明。



圖 2.31 根據二種明亮度指標之偵測結果

## 2.6 影像代理人

本研究之輔助系統將引用代理人(Agent)的觀念，用以加強輔助系統的功能，而有關於代理人詳細的定義請看第四章。整合後的白天與夜間程式我們稱之為影像代理人，它具有以下基本的功能：偵測日夜狀況以轉換演算法狀態，由所擷取的影像偵測出車道線和前方車輛，以及針對偏離車道線和距離前方車輛過近的情況發出警示訊息。以下是對影像代理人的介紹。

### 2.6.1 影像代理人警告訊息

#### a.防偏

假設，我們一開始車輛是在車道中央，且 CCD 是架在車輛的大約中心線上，若車輛行駛在直線路段，且沒有偏離的情況時，車道影像在螢幕上應該是左右對稱的。在這樣的前提假設下，若車道線在影像上不對稱時，依實際車輛行駛的情況來說，有兩種可能的情況：其一為車輛行駛的方向與車道方向有較大的差異，這種情形大多會發生在彎曲的路段或變換車道時；另一種情況是車輛與車道方向沒有太大的差別，而車輛是慢慢地偏向某一邊，最後偏離車道，這可能是駕駛員在打瞌睡或其他情況，使得駕駛員沒有自覺地使車輛逐漸偏離了車道。

根據這兩種情況，我們採用兩旁車道線的角度差及車道線與特定點(為攝影機所在的位置，如圖 2.32 (S 點)的位置差，來判定車輛是否會偏離車道。在角度的判斷上我們分別求取兩邊視窗內車道線的方向角，並計算其兩邊的角度差，如圖 2.33 所示，來判斷車輛是否有偏離車道，其角度計算如(2-20)所示。以位置對稱特性測量車輛是否偏移車道上，我們取視窗內車道線中心位置，與特定點 S 為軸的距離差(a-b)，如圖 2.34 所示，檢查是否在我們設定的範圍內，來判斷車輛是否會偏離車道。其計算結果如圖 2.35 所示。

$$\theta = \tan^{-1}(\text{直線斜率}) \times 180 \div \pi \quad (2-20)$$

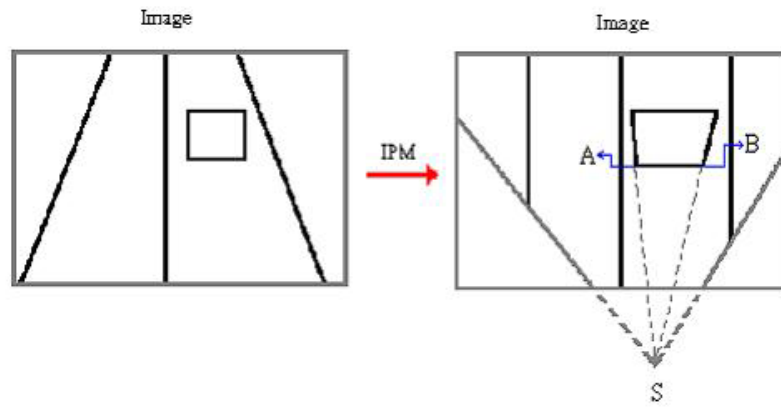


圖 2.32 圖像經 IPM 轉換後的示意圖

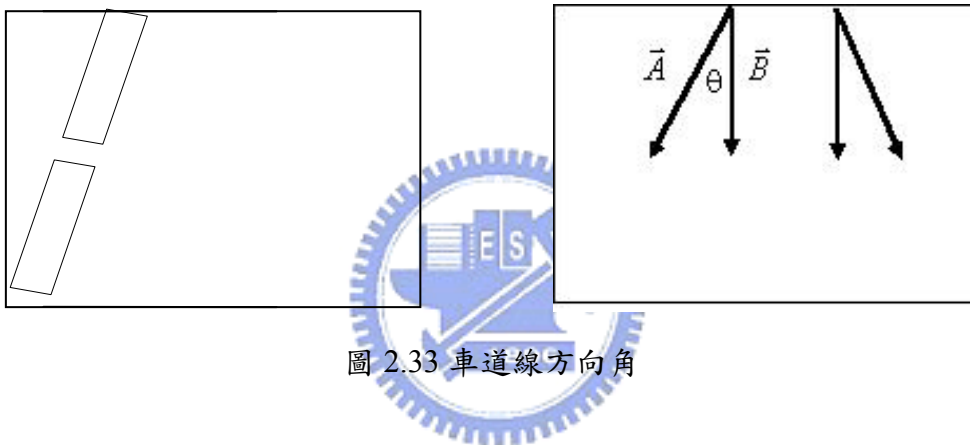


圖 2.33 車道線方向角

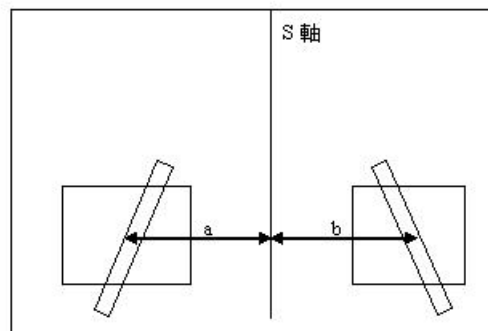


圖 2.34 位置左右對稱性



(a)  $a=100, b=120$  (pixel)

左邊車道線角度=45.376 度

右邊車道線角度=49.514 度

角度差=4.138 度、距離差=20(pixel)



(b)  $a=40, b=130$  (pixel)

左邊車道線角度=31.943 度

右邊車道線角度=60.461 度

角度差=28.518 度、距離差=90 (pixel)

圖 2.35 車道線角度差及距離差的計算結果範例

故在此我們的判斷法則有兩個部分：

1. 角度差：即兩邊車道線方向角的差值，在車子沒有偏離的情況下，兩邊的車道線方向角應該是差不多的。
2. 位置差：主要是根據車子在路中央行駛且沒有任何偏掉的情況時，其車道線在影像上應該是對稱的前提下，若車子沒有偏掉的話，車道線和特定點 S 的距離應該是差不多的，故若兩邊車道線到影像中心距離的差不大則表車子沒有偏掉。

而在車前防撞方面，我們將依所偵測的前方車輛位置來判斷，當其位置接近到一定距離時，是否有愈來愈接近並有可能產生碰撞的趨勢，則將再與雷射掃瞄器的資料相互配合比對，來確定警告產生。

### b. 防撞

經由前方車輛偵測可得前車相對距離，其方法是透過公式(2-8)式，將影像中的位置換算為世界座標中的位置，而得相對距離，如圖 2.36 所示。本系統只根據相對距離來做防撞警示，詳細原因會在之後的 3.3.3 小節中說明。



車輛位置：  
 影像座標(I.C.)：(198, 148)  
 單位：圖像

世界座標(W.C)：(-21, 2017)  
 單位：公分

圖 2.36 前車相對距離的計算範例

前方車輛的位置是經由前車偵測演算法而得，在此，夜間的偵測法則就出現了一個問題，在夜間的偵測方法中，因為依循的法則是尋找成對光點，當前方有車輛並行或尾燈能量相近時，會造成搜尋的錯誤(如圖 2.37(a))；或者在追蹤方法中，樣本比對的過程對路面光點太過敏感，導致樣本尺寸變大而造成錯誤發生(如圖 2.37(b))。這些錯誤皆會使測得的相對距離不正確，以至於發出錯誤的警示訊息，改進的方法是利用與雷射掃瞄器代理人的溝通，進行前方車輛位置的校正，進一步的說明請看 4.3 節。



(a)成對光點方法偵測錯誤



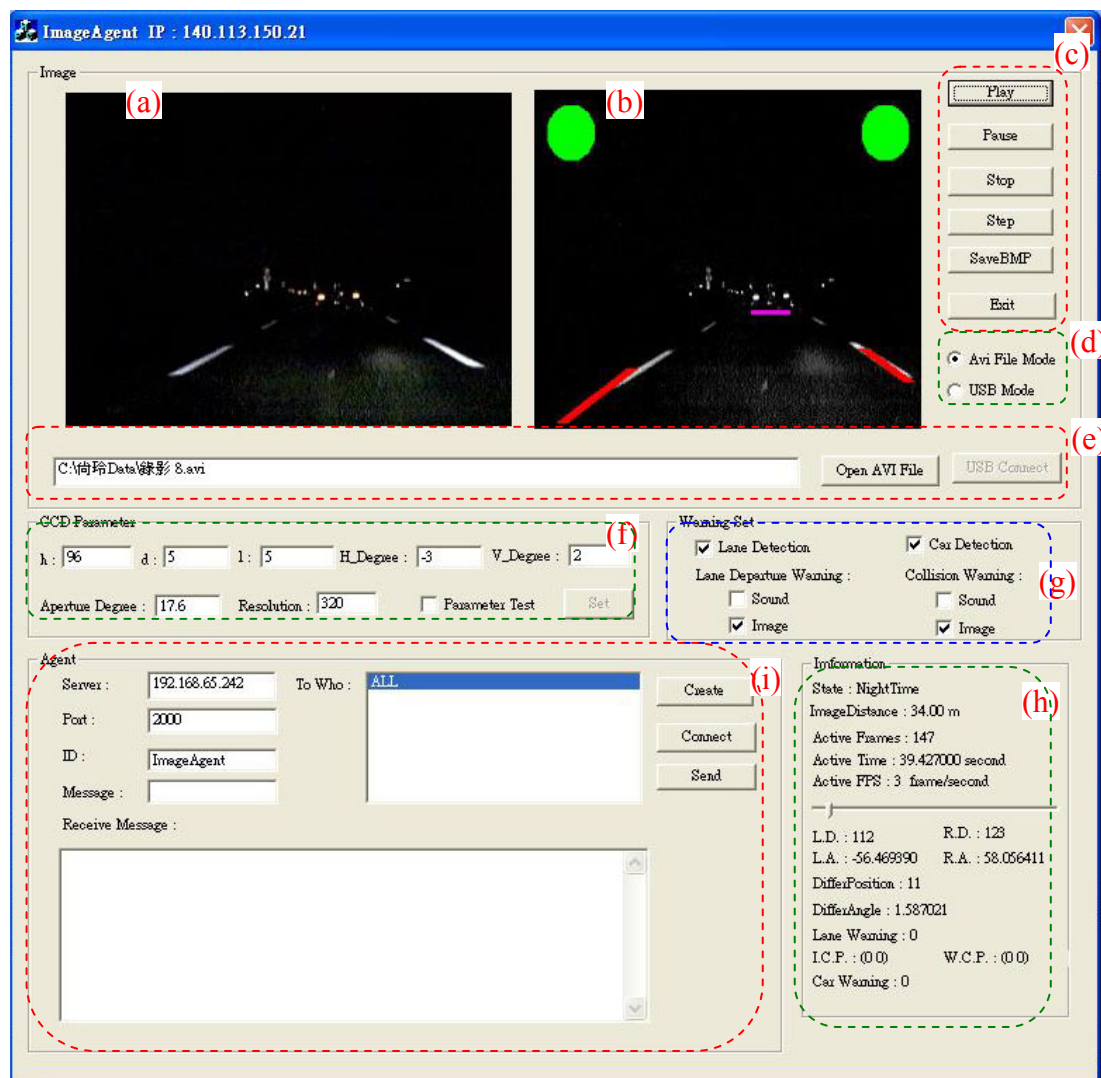
(b)追蹤方法因路面光點造成偵測錯誤

圖 2.37 夜間前方車輛錯誤偵測



## 2.6.2 影像代理人程式介面

圖 2.38 為影像代理人程式介面，其正在顯示的是影像檔(錄影 8.avi)測試結果。



- (a):來自 CCD 所接收的或影像檔的影像
- (b):偵測後的影像
- (c):相關操作
- (d):兩種影像選擇模式
- (e):不同模式的相關操作
- (f): IPM 相關物理參數的輸入
- (g):警告訊息的開關設定
- (h):偵測後的相關結果
- (i):影像代理人與其他代理人溝通介面

圖 2.38 影像代理人程式介面

此介面清楚的顯示駕駛人所需的資訊以及可操作的功能，駕駛人經由圖中的(b)顯示目前車前的偵測狀況(車道線被偵測出及一輛前方車輛正在被追蹤)，

右上角及左上角兩圖示分別為防偏、防撞的警告圖像；(g)為影像代理人的警告功能開關，包括車道線和前方車輛偵測功能、圖像警示、聲音警示，可依使用者需要自行調整；因此透過此介面，駕駛人不僅可即時的得知系統的偵測狀況，而且操作方式也非常的簡易，這便是我們系統特別所要求的互動關係。

為了方便設計者進程式演算法的測試，此介面還保留了某些功能鍵，(a)是原始的輸入影像，可用以與(b)作比對；(c)程式演算法的操作按鈕，有關於執行(Play)、暫停(Pause)...等功能；(d)選擇影像輸入模式，影像檔(avi 檔)或 CCD 畫面；(e)開啟某一影像檔或是連接 CCD；(f)車道線偵測法則(IPM)的相關物理參數的輸入，因為目前還未能做到參數的自動調整，須靠人工輸入；(h)可即時提供設計者有關目前偵測狀況的文字資訊；(i)顯示的是有關影像代理人與其他代理人的連線、溝通情況。



### 第三章 雷射掃瞄器防撞偵測系統

本章透過安裝在實驗車輛前方的雷射掃瞄器偵測前方障礙物，利用一維 Kalman Filter 改善偵測品質，並根據相對速度與相對距離得到 D/V 曲線，作為雷射掃瞄器代理人的防撞警示標準，最後與影像代理人比較後再次驗證其在防撞方面的優勢。

#### 3.1 車輛前方之雷射掃瞄器於接近前車時的實驗

首先對雷射掃瞄器做一介紹：本實驗採用 SICK 的雷射掃瞄器，其型號為 LMS 221，光學式雷射掃瞄器，為紅外線雷射光，經過美國國家標準協會(American national standards institute ; ANSI) 認證安全等級為 Class 1。這裡安全等級 Class 1 是指所發射的雷射光束對人眼及身體不會造成危險，雷射掃瞄器外型與特性如圖 3.1(a) (b) 所示，LMS 為 laser measurement system 的縮寫。

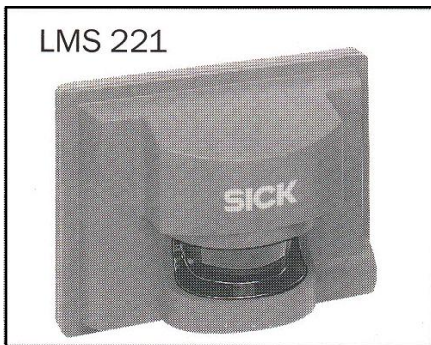
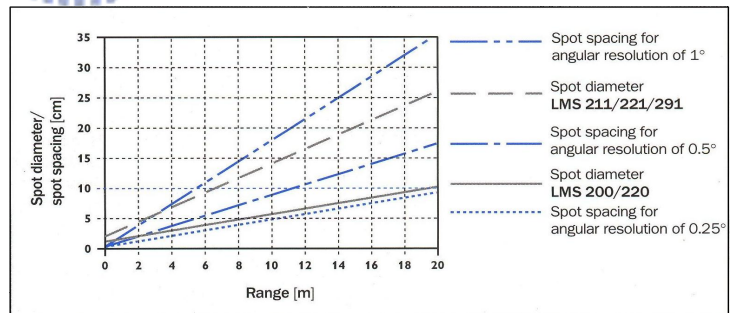


圖 3.1 (a) LMS 外型



(b)雷射光點直徑與相對距離

利用發射一脈衝為 3ns 的紅外線雷射光，發射與接收的雷射光以接近光速 ( $3 \cdot 10^8 m/s$ ) 在空間中傳輸，因此不會造成都普勒效應(Doppler effect)，或許移動中光的頻率、波長會發生變化，但是光速卻是不變的，利用 time - of - flight measurement 測得與障礙物間的距離，且其距離的精確度在 2cm 內。發射的光點

與相對應距離由圖 3.1 (b)所示，可以看出相對距離越遠光點直徑越大，因此在反射接收上較不容易造成漏接。

此雷射掃瞄器有其自訂的通訊介面，並透過 PC 串列介面與雷射掃瞄器做溝通，以連續下達 request for measured values telegram 命令，來得到雷射掃瞄器所測的距離資料，同時也紀錄每兩筆資料時間間隔，作為下一步量測到相對距離後相對速度的分析。

有關車前雷射掃瞄器的防撞實驗，安裝雷射掃瞄器在實驗用車輛前方，將車子向一面牆壁行駛，作為模擬兩車間的相對距離與相對速度實驗，雷射掃瞄器安裝在車輛前方如圖 3.2 所示，實驗用高爾夫電動車如圖 3.3 所示：



圖 3.2 實驗用高爾夫電動車



圖 3.3 裝置在車輛前方的雷射掃瞄器

在實驗方面，首先在 PC 端利用串列埠對雷射掃瞄器做資料存取，將相對距離與每兩筆資料的時間差紀錄下來，利用 Matlab 分析車輛行進時與前車的相對距離與相對速度變化，分別在兩種不同加速前進的情況下紀錄下來，如圖 3.4 的狀況 1 與圖 3.5 狀況 2。

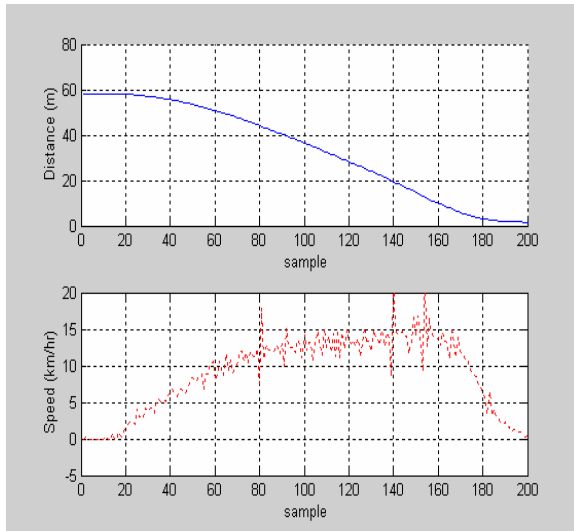


圖 3.4 (狀況 1)透過雷射掃瞄器所得到的相對距離與相對速度

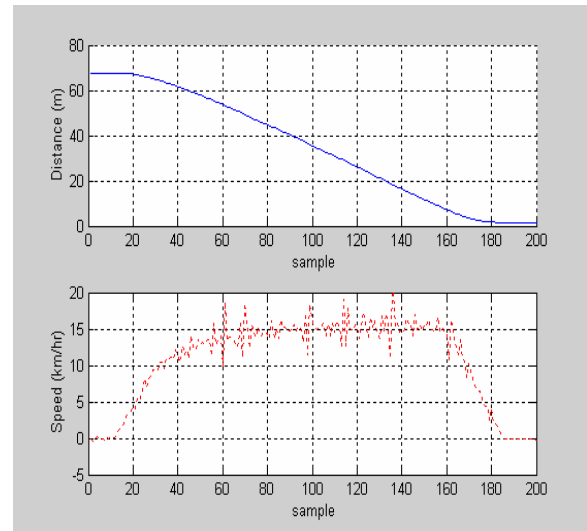


圖 3.5 (狀況 2)透過雷射掃瞄器所得到的相對距離與相對速度

圖 3.4 之實驗為一開始距離牆壁 58m 處，緩慢往前行駛至一定速度，當距離牆壁太近時，開始做剎車動作減速以避免碰撞；圖 3.5 之實驗為一開始距離牆壁 67m 處，以較高的速度加速到一固定速度，直到覺得距離牆壁太近而剎車做減速。相對距離的數值是透過雷射掃瞄器直接輸出(因為誤差不大，所以在相對距離上不作處理)，相對速度的取得是透過兩筆距離資料的差分( $\Delta v = \frac{\Delta d}{\Delta t}$ )。不過由圖 3.4 與 3.5 相對速度曲線，可以看到有片段且不連續的速度產生，其有可能造成在防撞系統中，因為瞬間速度突然加大(例如圖 3.4 與圖 3.5 中尖峰值)，造成系統的誤判而發出碰撞警示，這對兩車相對速度在防撞上的條件判斷是相當不利的。

仔細的分析這個問題，我們希望整體相對速度的曲線是平緩的，不應有相對速度在特定的時間點突然變大的情況發生，因此在相對速度的處理上，由於平均速度變化不大，所以最理想的曲線就是平滑曲線，故本論文提出以一維的 Kalman filter 來對所得相對速度資料做處理。

常見曲線平滑法(curve smooth)有平均、加權平均法與最小平方誤差等。不論是平均法或加權平均法，都是直覺式且未特別考慮曲線本身動態趨勢，最小平方

誤差則是假設曲線本身具有某種數學特性，而 Kalman filter 的理論根據可以讓此曲線的速度估測達到一平緩且平方誤差較小的基礎，相關式子推導請見[24]中的 3.2 節。

### 3.2 Kalman Filter 於車輛行駛時的速度估測

當兩車前後地在同一車道上行駛時，通常會以相近的速度來保持一固定安全車距，也就是跟車的狀況，因此對前一節的實驗(圖 3.4-5)而言，雖然安裝雷射掃瞄器在車輛前方，以模擬相對速度與相對距離間的關係，但是在很短的時間間隔內仍然可以視為等速前進，且因為雷射掃瞄器的距離量測誤差很小，所以距離測量狀態在速度估測上並不需做為要估測的狀態。相較其他速度估測不同處是以一維的想法來看此問題，假設量測系統狀態如下所示：


$$v(k) = v(k-1) + w(k-1) \quad (3-1)$$

$$y(k) = v(k) + \gamma(k) \quad (3-2)$$

其中  $v(k-1)$  為前一個時刻車子的速度狀態。

$v(k)$  為現在這個時刻車子的速度狀態。

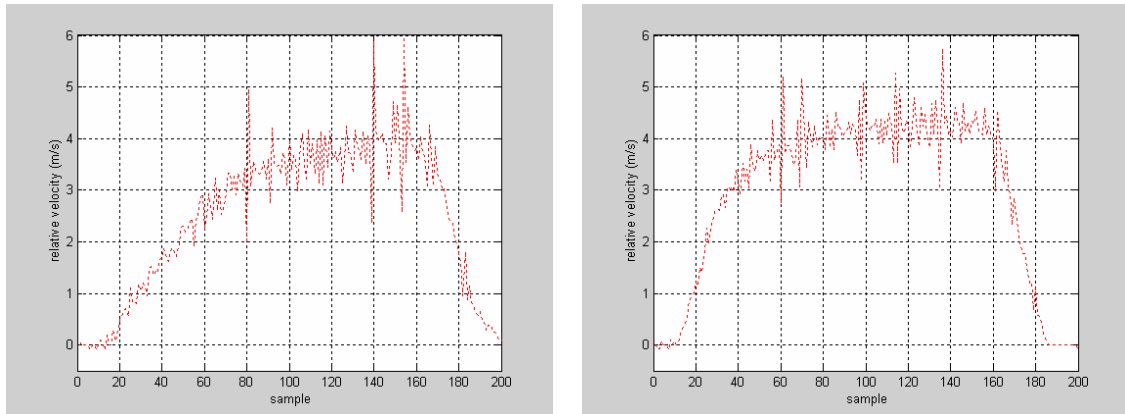
$y(k)$  為目前車子所觀測到的速度狀態，也就是兩筆距離資料的差分值。

$w(k)$  視為在輸入時的 white noise。

$\gamma(k)$  視為在輸出觀測時的 white noise。

在輸入、輸出系統狀態中，對於所存在的 white noise 可能造成的原因：在輸入狀態方面，其 white noise 是由於車輛動態行駛時，對輸入的速度狀態有一隨機的改變，駕駛人根據跟車狀況判斷踩下油門、或踩下剎車；而在輸出狀態方面，

其 white noise 可能是因為車子在行進間本身就有震動的問題，或速度的量測計算上直接採用差分，而造成相對距離與相對速度的量測受到干擾，使得相對速度曲線偶有尖峰值的狀態。而在這裡要處理的相對速度雜訊干擾問題，是把焦點集中在相對速度曲線的估測，於是分別將圖 3.4 與圖 3.5 的相對速度曲線取出，並且在單位上統一以 m/s 來做計算，如圖 3.6 (a)，(b)。



(a)

(b)

圖 3.6 透過雷射掃瞄器得到相對速度統一單位(m/s)

運用 Kalman Filter 對圖 3.6 (a)做速度的估測，估測狀態如下所示：

$$\hat{v}(k) = a\hat{v}(k-1) + b(k)[y(k) - ac\hat{v}(k-1)] \quad (3-3)$$

其中  $\hat{v}(k-1)$  為前一個時刻車子的速度估測狀態。

$\hat{v}(k)$  為現在這個時刻車子的速度估測狀態。

第一點的速度估測直接用量測到的值，也就是  $\hat{v}(1) = y(1)$ 。

此速度估測的系統狀態，其空間參數  $a = c = 1$ ，要小心  $cb(k) \neq 1$ ，相對均方誤差  $p(k)$ 、時變增益  $b(k)$  也如(3-4)式、(3-5)式所示：

$$b(k) = \frac{c[a^2 p(k-1) + \sigma_w^2]}{\sigma_\gamma^2 + c^2[a^2 p(k-1) + \sigma_w^2]} = \frac{p(k-1) + \sigma_w^2}{p(k-1) + \sigma_\gamma^2 + \sigma_w^2} \quad (3-4)$$

$$p(k) = \frac{1}{c} b(k) \sigma_\gamma^2 = b(k) \sigma_\gamma^2 \quad (3-5)$$

$w(k)$  為輸入狀態的 white noise，其變異數  $\sigma_w^2 = E[w^2(k-1)]$ ， $\gamma(k)$  為輸出狀態的 white noise，其變異數  $\sigma_\gamma^2 = E[r^2(k-1)]$ 。

在 Kalman filter 中，有幾個參數是未知的，包括  $w(k)$ 、 $\gamma(k)$  與第一個均方誤差  $p(1)$ 。因此在考慮實際狀況及經過實驗調整後，得  $p(1) = 1, \sigma_w^2 = 0.03, \sigma_\gamma^2 = 0.1$  這一組參數，利用其來處理圖 3.6 後，其結果如圖 3.7 所示。可以看出在 noise 干擾之下，最大的振幅干擾可以降低到 1/4 左右，達到理想的估測速度狀態，且估測狀態變化量不大，在均方誤差上也是很小。

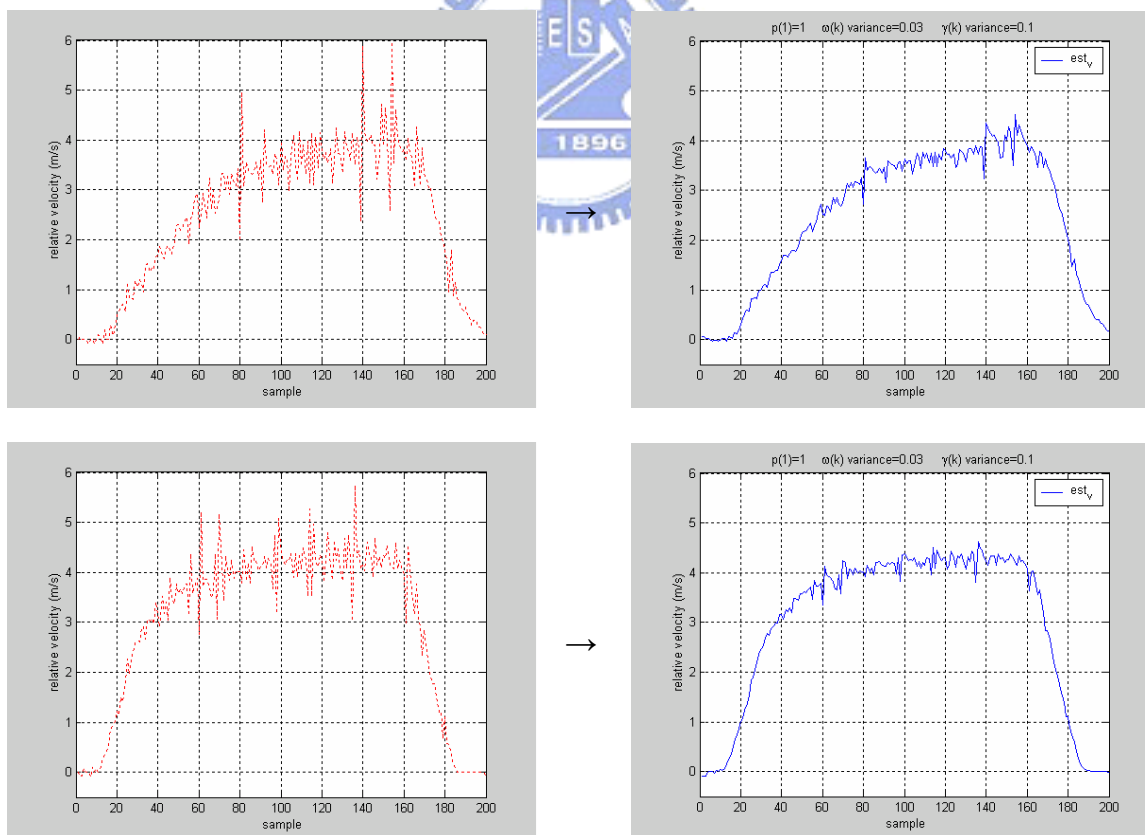


圖 3.7 Kalman filter 對圖 3.6 的速度估測 (速度變化較劇烈)

$$(p(1) = 1, \sigma_w^2 = 0.03, \sigma_\gamma^2 = 0.1)$$



再以同樣的參數  $p(1)=1, \sigma_w^2=0.03, \sigma_\gamma^2=0.1$ ，用 Kalman filter 來處理誤差變化不這麼劇烈的相對速度曲線，均方誤差值在自動修正效果上也很不錯，如圖 3.8。

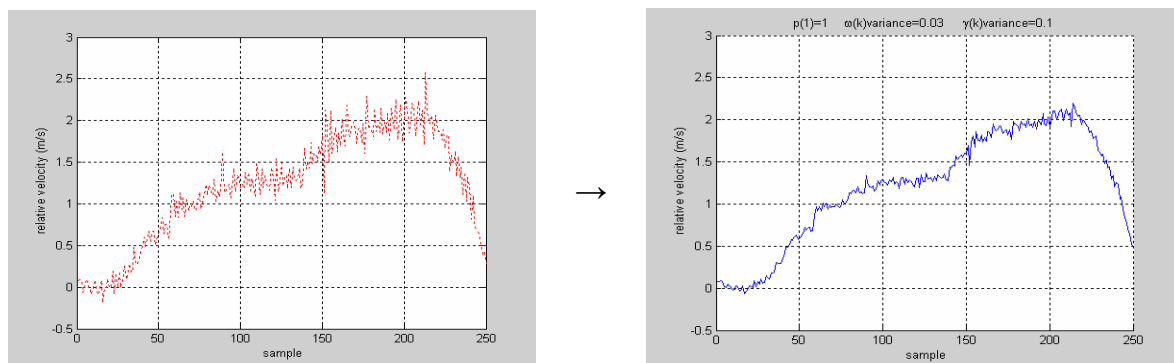


圖 3.8 Kalman filter 對左圖的速度估測(速度變化較緩和) ( $p(1)=1, \sigma_w^2=0.03, \sigma_\gamma^2=0.1$ )

在此提出以一維的 Kalman filter 經過實驗數據調整輸入、輸出系統狀態的 white noise 變異量，不僅在運算上方便且容易實現，並且對於不同情況下的車速駕駛都有很好的速度估測曲線。將所得到的速度估測值做為車輛前方防撞判斷的條件之一，可以大大降低系統誤判動作，使得駕駛人的行車安全更有保障。

### 3.3 雷射掃瞄器代理人

#### 3.3.1 雷射掃瞄器代理人警告訊息

在上節中，針對從雷射掃瞄器得到具有雜訊干擾的相對速度，透過 Kalman filter 做估測，得到一較平滑且均方誤差值小的速度估測值，而這節中就是要透過相對距離及相對速度的估測來討論車輛前方的防撞。

當兩車在同一車道上跟車，正常情況下當駕駛人覺得與前車距離太近，或是相對速度太大，都會採取放開油門、踩剎車等動作來拉大距離或是降低兩車間的相對速度，以避免發生碰撞。簡單來說，追撞事故的造成原因就是兩車相對距離與相對速度間的關係，相對速度太大或是相對距離太小都可能發生碰撞。從圖 3.9 可以看出，相對速度的變化在 sample 140~160 開始有下降趨勢，這說明了一個訊息，當駕駛人覺得與前車間相對速度太大以致於可能發生危險時，駕駛人會開始踩剎車減速，來降低兩車間的相對速度，相對的如果在 sample 140~160 沒有開始作剎車動作，那就會發生碰撞。

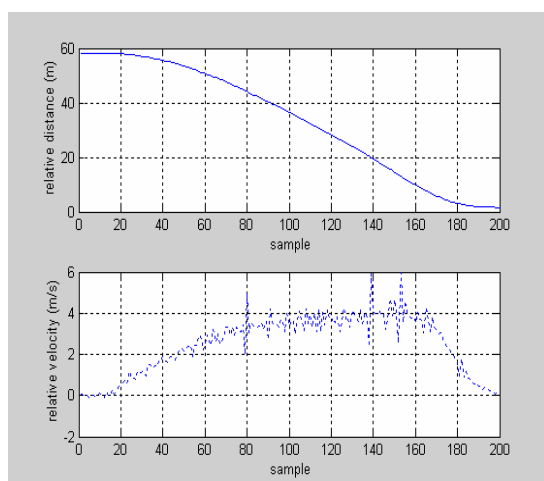


圖 3.9 相對距離與相對速度

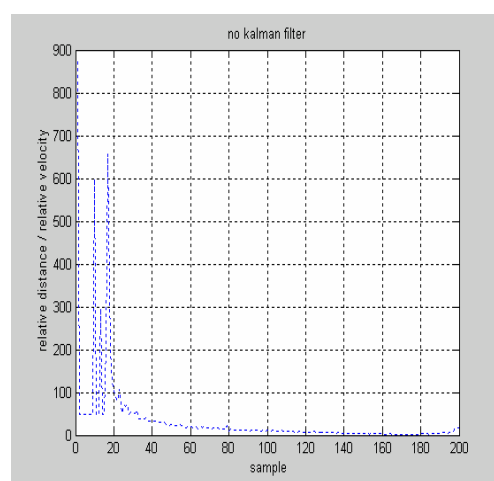


圖 3.10 相對距離除以相對速(D/V)  
(原始結果，未經過 Kalman filter 處理)

因此根據實驗，提出碰撞之前必須先警告駕駛人的預警時間，即為兩車相對距離與相對速度間的關係，可以如下表示：

$$P = D/V \quad (3-6)$$

僅就實驗所得到的相對距離除以相對速度與估測速度看其結果(在相對速度為 0 或是小於 0 時，將  $P$  設定為 50)，如圖 3.10 所示，並特別在 sample 140~160 放大刻度來看，如圖 3.11 所示：

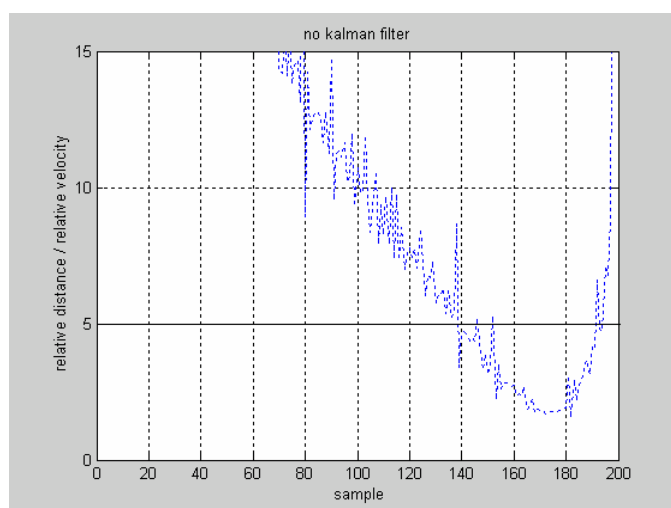


圖 3.11 圖 3.10 相對距離除以相對速度(D/V)放大刻度

在這個例子裡，若是在 sample 140 之前就事先警告駕駛人，讓駕駛人有一個緩衝時間，就是對預警時間  $P = D/V$  設定一個數值，這裡訂定為 5(其意義為 5 秒後將發生碰撞)，但從圖 3.11 可以發現在相對速度未經過 Kalman filter 處理前，其結果很有可能會發生誤判的動作，因為相對速度受到干擾，其預警時間也存在很大的擾動。來看看經過 Kalman filter 相對速度估測後的  $P = D/V$  曲線，在 sample 140~160 區段：

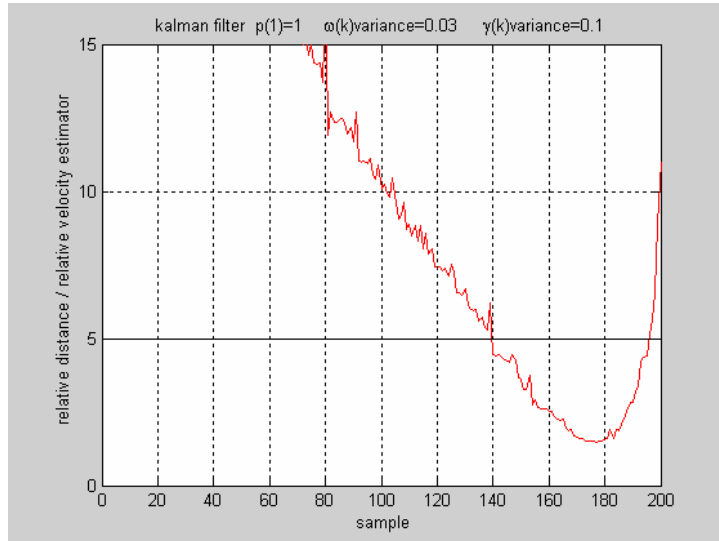


圖 3.12 相對距離除以經過 Kalman filter 估測的速度

可以看出經過 Kalman Filter 處理後的速度估測值，作為防撞的判斷條件可以減少系統的誤判動作。另外，對於相對距離較短且前進速度變化不大情況下也有不錯的效果，如圖 3.13，圖 3.14 所示：

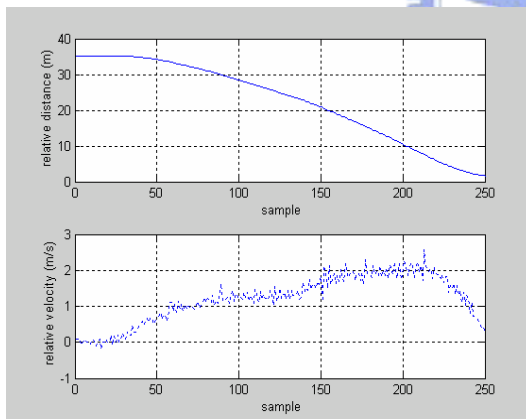


圖 3.13 相對距離與相對速度

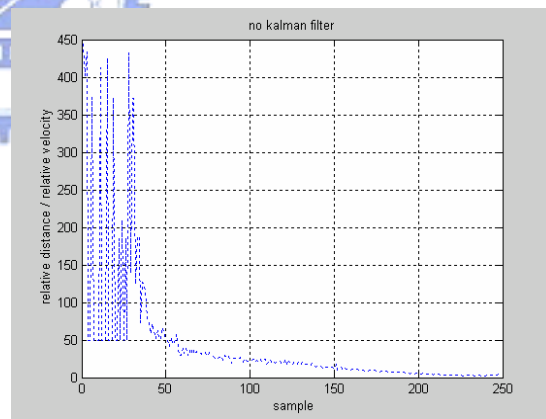
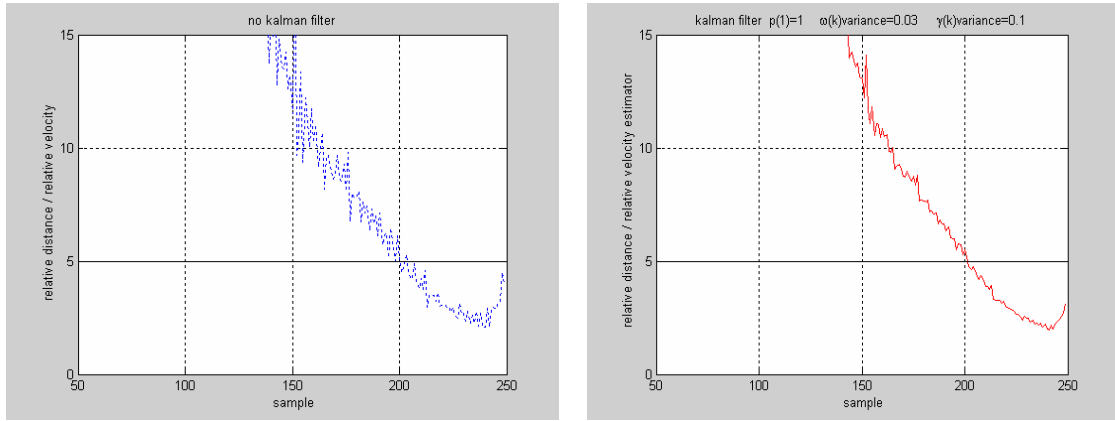


圖 3.14 相對距離除以相對速度

由圖 3.13 可以看出在 sample 200 左右速度開始減慢，雖然駕駛人行駛速度較慢，但相對距離太近可能碰撞，因此減速。將刻度放大，特別在 sample 200 觀察，未經 Kalman filter 處理的預警時間圖 3.15 (a)，經 Kalman filter 處理後的相對速度估測其預警時間圖 3.15(b)：



(a)相對距離除以相對速度放大刻度 (b)相對距離除以估測的相對速度

圖 3.15

最後附帶一提，此方法只適用於其物體是以一定角度向雷射掃瞄器接近，且由實測結果也指出，若不以一定角度向雷射掃瞄器原點接近，其警告偵測將失敗，所以我們透過所獲得的距離資訊，作簡單的原點變換，在左右以每 25cm 的間隔建立 6 個虛擬的雷射掃瞄器，如圖 3.16，可讓警告偵測的範圍更大，更加安全，並由於雷射掃瞄器掃瞄器範圍太大，所獲得的距離資訊變化太多，所以我們設定一特定範圍，超過此範圍時，其距離設為此範圍邊緣值，進入此範圍時，才起動防撞警告偵測，可減少錯誤警告的發生。

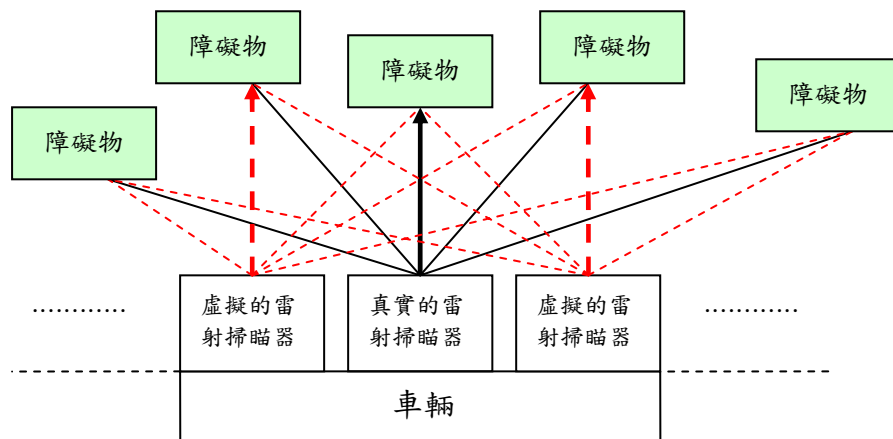


圖 3.16 真實與虛擬的雷射瞄器配置

### 3.3.2 雷射掃瞄器代理人程式介面

雷射掃瞄器代理人程式介面介紹：透過圖 3.17 可設定雷射掃瞄器的硬體參數；經由圖 3.18 的介面可與溝通管理伺服程式連線設定，並可顯示出來自其它代理人發出的訊息；以及程式的主畫面(如圖 3.19 所示)，其主要任務為緊急的防撞警告，可依使用者的習慣轉換不同的接收模式，圖(a)文字模式含有所接收到的距離、前方物體位置及防撞警告資訊，圖(b)前方物體模式其為繪製前方物位置及寬度，圖(c)繪圖模式其為依前方障礙物距離資訊所繪製的圖像。

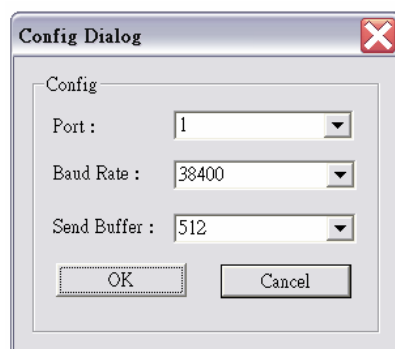


圖 3.17 設定雷射掃瞄器參數之程式畫面

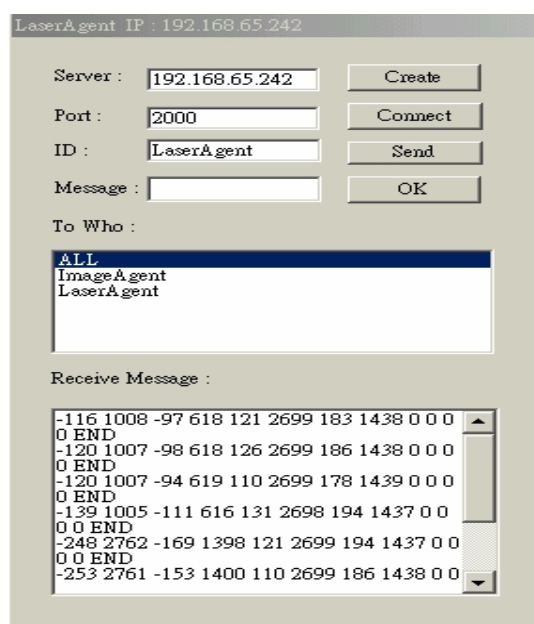
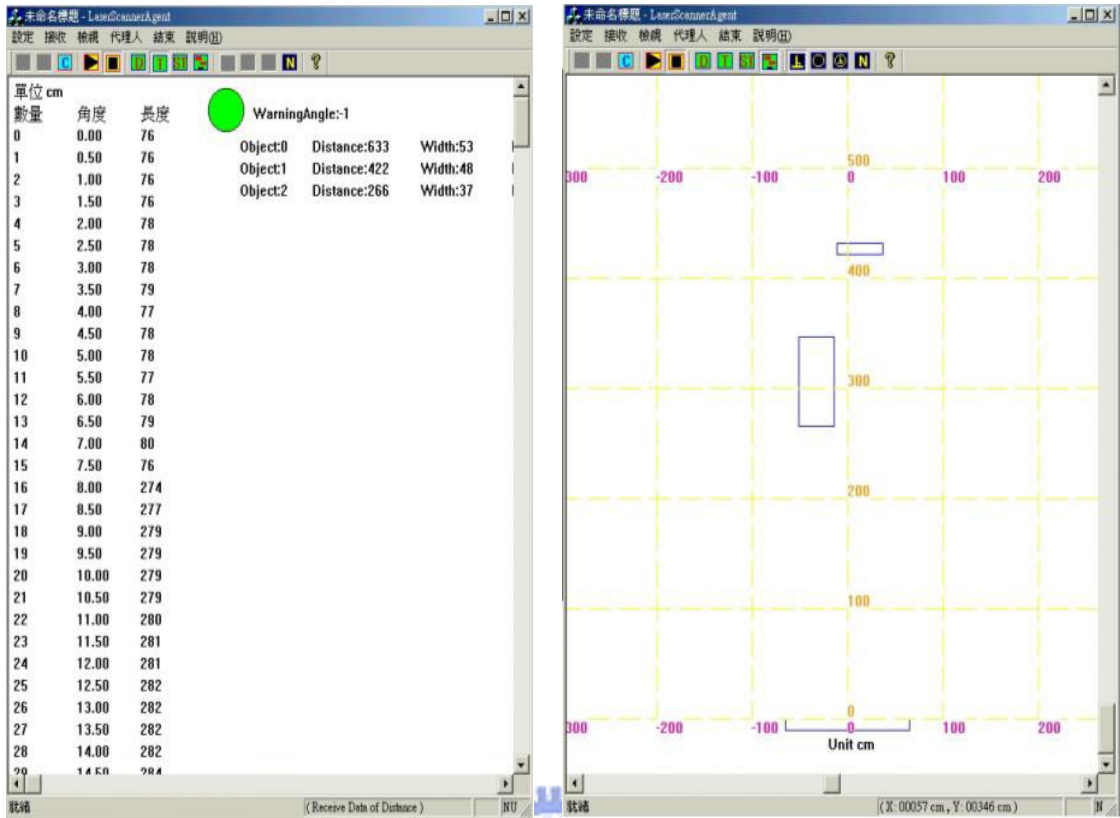
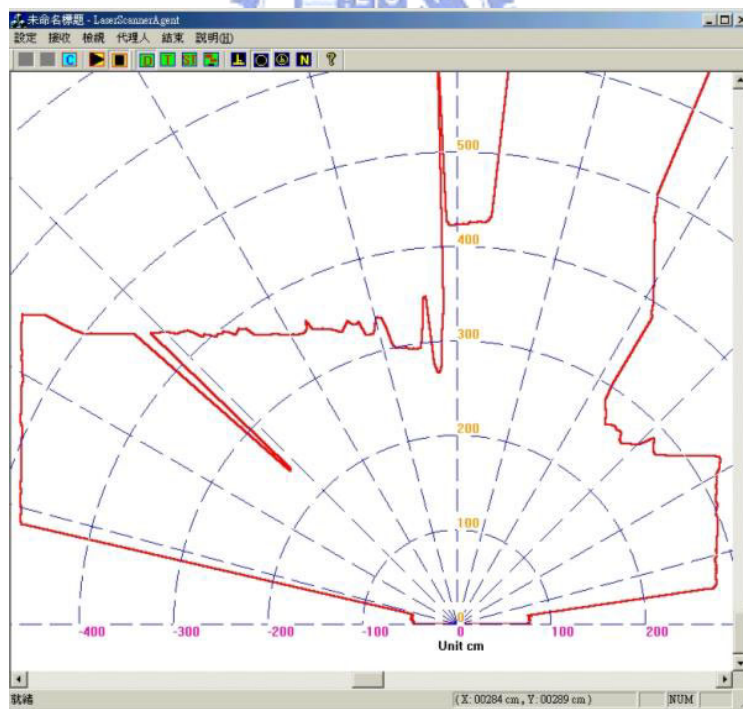


圖 3.18 雷射掃瞄器與溝通管理伺服程式連線設定，及所獲得其它代理人所發出的訊息顯示之程式畫面



(a)文字模式畫面

(b)前方物體模式畫面



(c)繪圖模式畫面

圖 3.19 雷射掃瞄器代理人程式主畫面

### 3.3.3 與影像代理人於防撞方面的比較

在 2.5 節中有提過，影像代理人具有防撞的警示系統，這表示其程式具有測量駕駛人與前車距離的能力，那何須雷射掃瞄器存在呢?在此節中，我們針對防撞能力將此二代理人的優缺做一比較。

為了公平得到二方的數據，我們為此設計一實驗，其場景如圖 3.20 所示，一輛裝有 CCD 與雷射掃瞄器的車輛(即為高爾夫球車)，停於 50 公尺處，前方有一輛車位於 0 公尺處以接近等速度往高爾夫球車接近，兩個感測器同時接收資料：雷射掃瞄器經由掃瞄前方障礙物得到相對距離、相對速度、與 D/V 曲線等資料(如圖 3.21);影像代理人經由偵測前方車輛位置，利用(2-7)式得其相對距離，之後再用 Excel 計算出相對速度與 D/V 曲線(如圖 3.22)。附帶一提，此實驗是在雷射掃瞄器測得的相對距離是準確的假設下進行。

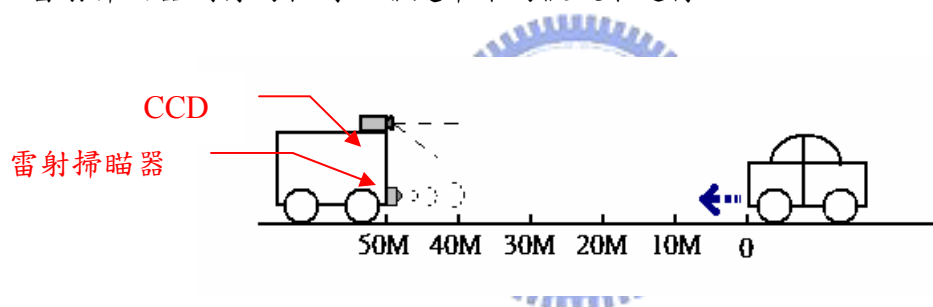
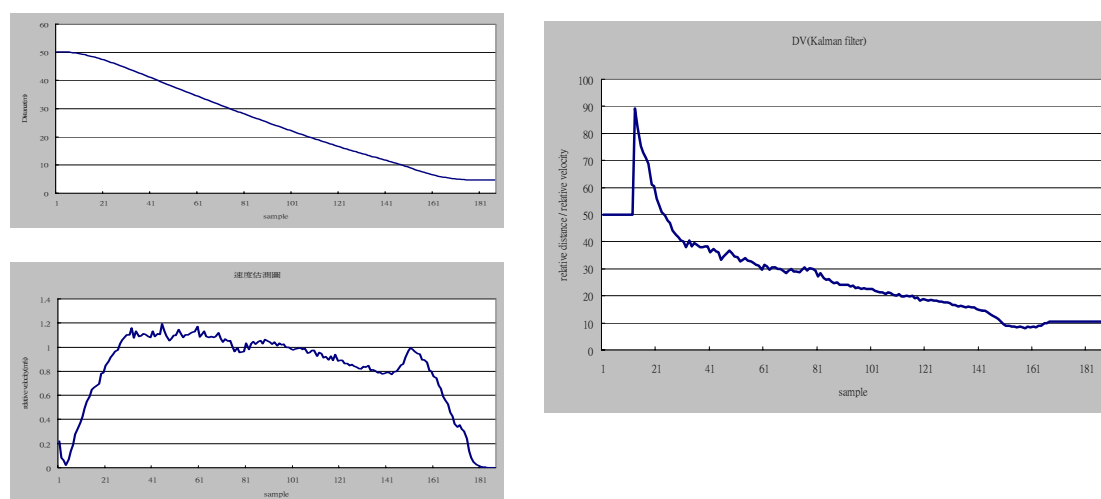


圖 3.20 前方車輛測距實驗場景圖

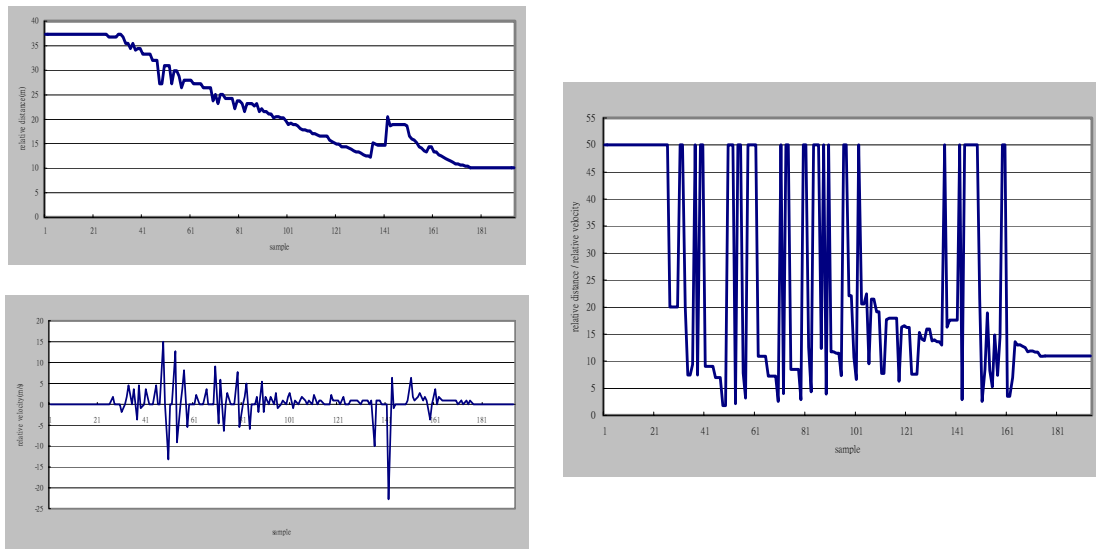


(a) 相對距離與相對速度

(b) D/V 曲線圖

圖 3.21 雷射掃瞄器代理人測得資料





(a)相對距離與相對速度

(b) D/V 曲線圖

圖 3.22 影像代理人測得資料

相較之下，雷射掃瞄器所測得相對距離非常精準，精確度可到五公尺以下，而影像代理人因為偵測演算法的關係，在近距離的偵測因跳動過大而不準確，且由圖中看出 10 公尺已經是最低的底限，然而其底限會因 CCD 俯角而略有差異，根據實驗所得，在偵測狀態穩定與環境單純時，影像測得距離與真實世界的距離誤差為 2~3 公尺；至於相對速度與 D/V 曲線，雷射掃瞄器因為引用前幾節所介紹的方法而有相當好的可靠度，而影像代理人則因為不平滑的相對距離曲線造成沒有意義的相對速度與 D/V 曲線。

綜合以上，我們驗證了在防撞方面上，雷射掃瞄器代理人具有相當穩健的偵測能力，而影像代理人則無法使用相對速度及 D/V 曲線作為警示依據。因此，在之後的系統整合上，影像代理人的防撞警示除了依據其所得相對距離外，還須由雷射掃瞄器代理人提供相關訊息，以確保輔助系統的可靠性。詳細介紹請見第四章的 4.3.1 小節。

## 第四章 全天駕駛輔助系統

本章將描述有關以代理人為基礎之車道線防偏與車前防撞警告系統、架構與實現，並且說明影像與雷射代理人如何藉由溝通、協調以互補不足，進而完成一天的駕駛輔助系統。

### 4.1 代理人的介紹

代理人這一個概念的呈現並非是最近才興起的，它可以推回到早期人工智慧的發展。代理人的概念於 1950 年中期由 John McCathy 所提出，而『Agent』這一個字眼則由 MIT(Massachusetts institute of technology)的 Oliver G. Selfridge 所創造。Nwana (A.D. 1996 年) 將有關代理人技術的研究進展分為兩個主要階段：第一階段起自 1977 年，主要是應用在分散式人工智慧；第二階段則起自 1990 年，與傳統的人工智慧領域不同的是，在這個階段比較不注重思慮和推理等能力，所專注的方向是 doing 和遠端行動。這一波風潮的帶動主要是因為有兩個力量在拉扯，第一個是因應網路時代的來臨，資訊的大量成長已經不是光靠人就可以解決，因此需要代理人幫忙我們進行一些工作，像搜尋資訊、過濾資訊等等；另外一點就是針對個人化需求的來臨，讓使用者或軟體應用發展者能夠快速的發展個人化、多樣化的應用。

關於代理人的定義，到目前為止並沒有一個統一的說法，在此便不多加贅述，詳細的整理可在[24]的第四章中看到。代理人的分類與特點，可參考 Gilbert 等人(1995, IBM)所描述的：將代理行為(agency)，智慧(intelligent)和移動性(mobility)分別代表三個維度，透過這三個維度形成如圖 4.1 所示的空間概念以分類代理人。

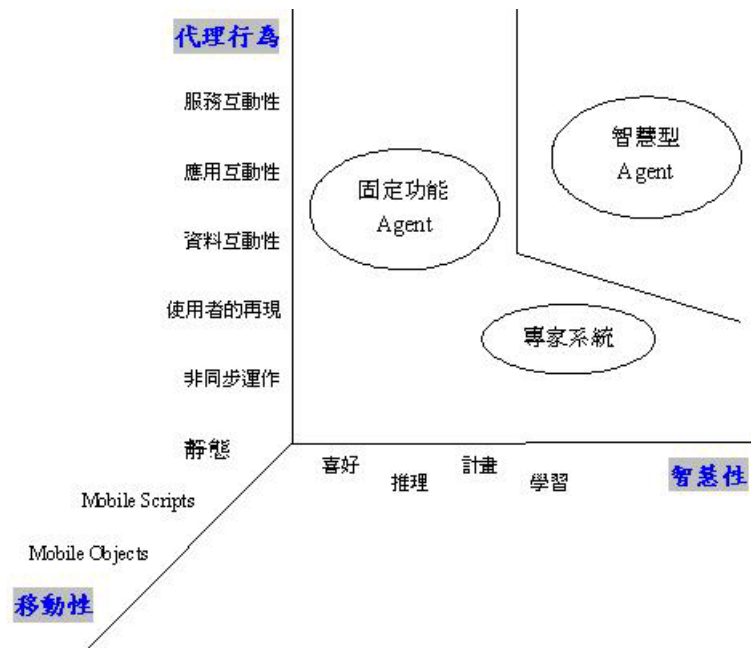


圖 4.1 Gilbert 對 Agent 的分類

- 智慧性(intelligence)：表示的是推理和學習的能力，屬於傳統人工智慧的範疇，可以分為喜好(preferences)、推理(reasoning)、計畫(planning)和學習(learning)。
- 代理行為(agency)：表示代理人是運作在一個什麼樣的機制下，基本的可能只是代理人和代理人之間簡單的非同步協調運作，更深一點的可代表使用者的某些行為，甚至是一個完整的服務。可將其劃分為幾個程度：非同步運作(asynchrony)、使用者的再現(representation of user)、資料互動(data interactivity)、應用互動(application interactivity)、服務互動(service interactivity)。
- 移動性(mobility)：表示代理人在網路上不同節點的移動能力，其想法來源比較偏屬於分散式計算，可將其劃分為靜態(static)代理人、mobile scripts 和 mobile objects。

由 Gilbert 等人所劃分的技術維度來看，我們可以看到傳統的應用，如專家系統、智慧型代理人等等，其多主要是在代理行為和智慧性兩個維度的設計，而移動性這個維度的提出主要是因應現在的系統越來越複雜，已經不是單一平台能夠處理，因此移動性這個功能將能夠分擔計算上的負擔、以及一些其他的優點。而從 Gilbert 等人對代理人的劃分仍然無法讓我們具體的了解到一個代理人該具有那些重要屬性，此外，根據其他文獻的討論，代理人還有其他的內涵屬性，然而代理人並不需要完全符合所有的屬性，而要視代理人的應用和目標而論，來增加或減少他的功能和屬性。

代理人系統，特別是多代理人系統，是分散式人工智慧(distributed artificial intelligence ; DAI)研究當中的一個子領域，已經在人工智慧領域中被提起二十多年。分散式人工智慧又成分散式問題解決(distributed problem solving ; DPS)和多代理人系統(multi-agent systems)兩個部分被廣泛討論，下面將就代理人系統分成單代理人系統和多代理人系統兩種來為大家介紹。

單代理人系統比較簡單，也比較傾向解決自己內部的問題，在整個外界環境中，自主性的透過偵測器(sensors)蒐集外部環境資訊，並在內部則使用自己的領域知識(domain knowledge)做出符合自己目標的處理、決策並直接反應(effectors)到外界環境上，期許結果能夠達到符合自己的目標。圖 4.2 表示了一個單代理人系統的架構圖。

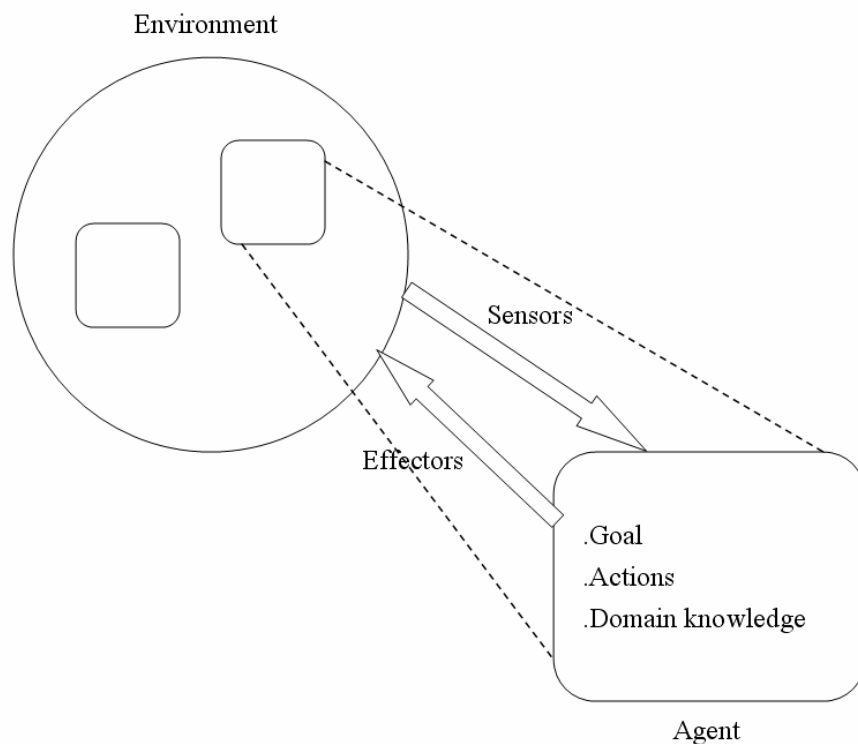
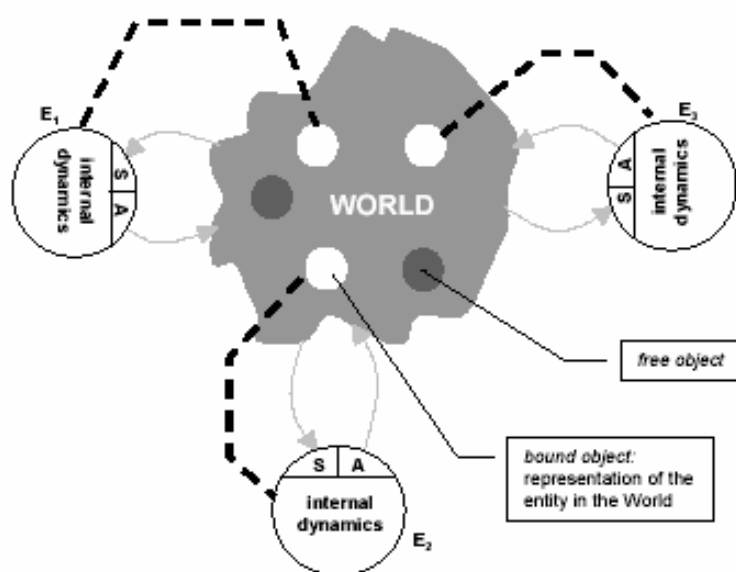


圖 4.2 單代理人系統的架構

多代理人(Multi-Agent)系統是近年來在人工智慧的研究中，引起相當多討論的一種概念，其概念為利用電腦模擬人類的社會行為，例如互相合作、競爭資源等以達成某一目的。多代理人之研究著重於一群分散於網路中的問題解決者，如何透過之間的行為互動、本身的能力以及對於周遭環境的認知來解決問題。而這些問題解決者通常稱之為代理人。因此，代理人與代理人之間的互動就會是多代理人系統研究領域中的研究重點，如：合作(cooperation)、交涉協商(negotiation)、溝通(communication)與協調整合(coordination)。

應用於本研究的多代理人系統著重在分散式的觀念，首先讓我們來看看 Z. Papp 等人[17,25]針對智慧型車輛系統發展的相對應模擬環境，圖 4.3 為其提出的 MARS(Multi-Agent real-time simulator)概念圖。*World* 代表駕駛人週遭的模擬環境，是一組 Objects 的集合，駕駛人可經由 Sensors 去感測 *World* 並產生相對應的 Entities，為了讓此模擬系統更真實，他們加入了 Actuators，其可即時的修改 *World* 中的 Objects。



- World : 代表周圍環境。
- Sensors (*S*) : 收集 *World* 的真實狀態資訊。
- Actuators (*A*) : 產生或摧毀 Objects，以及修正 Object 屬性。
- Objects : 用以描述 *World*。
- Entities( $E_i$ ) : 經由 Sensors 或是 Actuators 產生。

圖 4.3 MARS 模型概念圖

Z. Papp 等人更進一步清楚的說明所要模擬的”multi-world”環境(如圖 4.4)。

舉例來說，在 *World* 中有一 Object( $O_2$ )表示一輛前方車輛，此時系統中有二種 Sensor，影像( $S_1$ )和雷射掃瞄器( $S_2$ )，分別針對 *World* 有自己的詮釋( $W^1$  和  $W^2$ )，所以也有相對應的  $O_2^1$ (此車輛的外型、顏色等等)和  $O_2^2$ (此車輛的距離、車速等等)，因此，結合二種 Sensor 分別對於  $O_2$  的詮釋後產生了  $E_1$ (例如：帶有外型和距離的資訊)，當然，在此圖中沒表示出來的，針對模擬的需要，還可透過 Actuators 改變模擬情境，例如側方來車、前方車輛遠離等等。

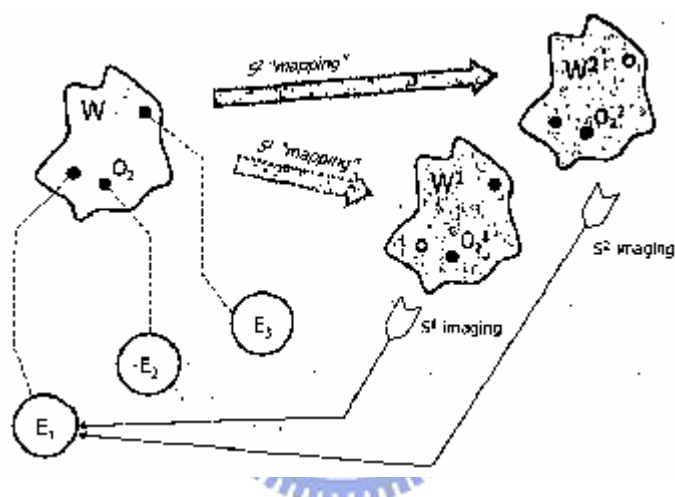


圖 4.4 ”multi-world”示意圖

本研究在多代理人的應用層面與 Z. Papp 等人所提出的非常類似：環境資訊擷取的模式，以及同樣有影像和雷射掃瞄器二種感測器；不同的是，第一，我們是實作於真實環境中。第二，影像與雷射掃瞄器並非抽象，並且可分別產生 Entity。第三，利用 Entities 產生警示訊息以輔助駕駛人的行車。要是實際結合硬體則其程式設計圖將如圖 4.5 所示，是一集中式的設計方式，也就是將 CCD 與雷射掃瞄器的接收處理程式寫在一起，在同一電腦上做計算，如此的結果會增加程式設計上的複雜度，如要再加上即時性的考慮的話，撰寫上非常的不容易，執行效能也會被降低。

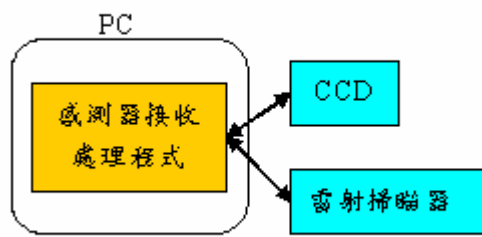


圖 4.5 集中式程式設計

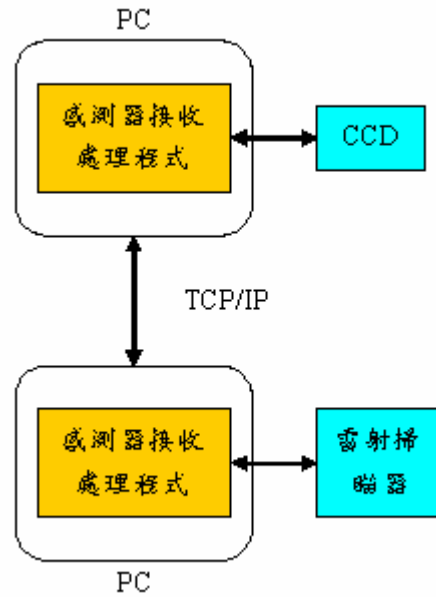


圖 4.6 分散式程式設計

因此，為了增加各感測器接收處理程式的擴充性及修改性，我們改為如圖 4.6 的程式設計方式，將分散式的概念也套用在感測器上，分別撰寫各別感應器的接收處理程式，並置於各別的平台，再考慮溝通傳輸的介面，採取這樣的方法，在每個代理人的程式撰寫上就只需考慮單一感測器，所以較為簡單，並其程式置於各別的平台(可跨平台)，執行效能相對較高，且利用網路作溝通合作，得到擴充性及修改性的優點，例如再加入一感測器時，只需了解其相互間的傳輸模式，就可加入此系統，獲得所需的資訊，即使需要修改相互間的程式，也只需作部份的修改，即可完成，讓系統有更大的彈性。但由於分散式的關係，相對於集中式的，其成本上的支出相對提高，整體操作上也較為複雜。

於是，目前在本系統中有二個代理人，分別是影像代理人和雷射掃瞄器代理人，個別的代理人皆已具有基本的輔助功能，進一步的可經過二方代理人合作協調後互補不足，提高行車安全性。



## 4.2 系統整合

### 4.2.1 以代理人(Agent)整合影像與雷射掃瞄器

一般在使用感測器時，大多都是每個感測器擁有各自的運算邏輯，只會針對各自所負責的任務，進行其任務，但每個感測器能力有限，其運算邏輯也並不完美，當所針對的環境較為單純時其仍足以應付，但若所針對的環境較為複雜時，將會造成各感測器正確性、穩定性的降低，或者產生其無法應付的問題。

例如：我們所使用的雷射掃瞄器及 CCD，在雷射掃瞄器方面，主要負責監控前方 180 度範圍中，是否有物體快速往我們所在車輛異常接近的趨勢，而發出警告以提醒駕駛者注意，但是在正常行駛時，時常會有對方來車會車的情形，有時當我們的車輛行駛較靠近車道中線時，很容易將其誤判，或是在轉彎時也很容易誤判另一車道上的來車。而在 CCD 感測器方面，我們由所接收的影像來判斷車道線及前方車輛，但偵測前方車輛時，如何確定前方是否有車輛、是否要啟動前方車輛偵測，並不容易，因其所處的環境有太多的狀況，很容易造成誤判，以致系統的處理速度降低等，這些問題由各別感測器自行來解決其問題並不容易，或是要花上大量的處理時間。因此，我們引進了代理人的觀念，來解決上述的問題。

由於代理人的定義眾說紛紜，依切入點的不同，而有不同的定義，到目前為止，並沒有一統一的說法，而我們所採取的定義為 Wooldridge 在 2000 年所提出的：『代理人能自主性的進行運作，主動察覺環境的變化並採取相對應的動作，本身並擁有特定的技能來，執行使用者所賦予他的任務，而所謂的智慧可以是簡單固定的程序或物件邏輯，也可以複雜到具有合作、推理和學習能力』。

以我們實驗為例，我們在 CCD 與雷射掃瞄器各安排一個代理人，為影像代理人 and 雷射掃瞄器代理人，其特色為：

- 1、「能主動察覺環境的變化並採取相對應的動作」，例如：影像代理人由 CCD 擷取前方影像，從影像中獲得車道線與前方車輛，監控車道線與前方車輛資訊，當有偏離車道線或與前車過近的狀況時則提出相對應的警告。而雷射掃瞄器代理人由掃瞄器獲得距離資訊，並經程式處理後獲得相對速度與 D/V 曲線，監控各方物體是否有快速異常接近趨勢，而發出相對應的警告。
- 2、「本身並擁有特定的技能，來執行使用者所賦予他的任務」，例如：影像代理人擁有車道線與前方車輛偵測及防偏、防撞警告的技能，而雷射掃瞄器代理人，則擁有相對速度計算及修正與防撞警告的技能，來執行使用者所要求的危險警告之任務。
- 3、「簡單固定的程序或物件邏輯，或具有合作、推理和學習能力之智慧」，例如：影像代理人 and 雷射掃瞄器代理人，除了各擁有其偵測法則、程序及警告判斷外，並各有其優點、缺點，我們更將在其間建立區域網路，能彼此相互溝通、合作、訊息交換。

由上述可知，我們所設計之影像代理人和雷射掃瞄器代理人，符合 Wooldridge 在 2000 年所提出的定義，故可稱之為代理人。在本論文中，我們主要是引用代理人溝通與合作的觀念，來整合 CCD 與雷射掃瞄器，並改善上述在複雜環境中，單一個別獨立的感測器可能產生的問題。在我們實驗中，影像代理人從影像中可獲得較多形態上的資訊，如車道線、前方車輛等，而雷射掃瞄器代理人擁有正確的距離資訊，可以說兩者互有優點但仍有不足，因此，以溝通與合作觀念來整合此系統，主要使影像代理人與雷射掃瞄器代理人互補其不足，不僅可提升系統可靠度，並增加系統的正確性、處理速度及穩定性。

#### 4.2.2 多代理人溝通管理平台

在 4.1 節的最後提到本研究的分散式程式設計理念，各代理人間是利用網路作為溝通的傳輸介面，而為了讓此類溝通介面具有統一格式而有 FIPA 規格(FIPA standard)的出現。FIPA(Foundation for Intelligent Physical Agents)[27,28]成立於 1996 年，是一個非營利性組織，主要目的是規範異質性代理人交互運作的標準。FIPA 組織發表許多 FIPA 規格書，其內容定義異質性代理人之間的互通與代理人提供服務的方法。FIPA 規格書的範圍極廣，例如：抽象架構、代理人管理、溝通協定、溝通語言、個人旅遊助理、行動代理人管理、視聽娛樂與廣播、服務品質以及其他服務規格等。FIPA 提出的代理人平台(如圖 4.7 所示)包含有：(1)代理人目錄：提供代理人命名、位址與狀態管理；(2)服務目錄：提供代理人服務型態、位址的查詢；(3)訊息傳遞：代理人與代理人之間訊息的傳遞、接收與聯繫等工作；(4)代理人溝通語言：提供代理人與代理人之間互相溝通時使用的語言，包括：本體論、語法與文法。目前已經有許多開發工具是遵循 FIPA 規格書進行實作，例如:ZEUS、FIPA-OS、JADE 等。其中 JADE(Java Agent Development Framework)[29]是最為廣泛應用的平台，使用高階的 FIPA-ACL 作為代理人溝通語言。

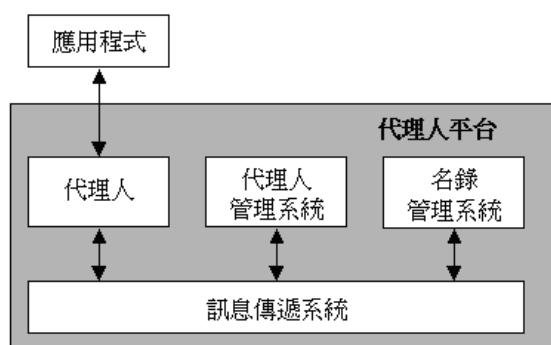


圖 4.7 FIPA 所提的代理人系統架構

關於代理人溝通語言(agent communication language ; ACL)有低階與高階之分，低階是指一般人所熟習的 WAP(用於 wireless)、TCP/IP(用於 internet)、XML/HTTP(用於 web)；而高階則有 KQML、FIPA-ACL、FIPA-KIF、KIF 等。低階的 ACL 強調的是「界面」，可以促進代理人系統的效率。高階的 ACL 強調的是傳輸內的「內容」，或許可以稱之為「語意」，使用高階的 ACL，代理人只需透過一些簡單的傳輸機制，便可傳達任何可能的意思，則其可以促進代理人系統的威力。

在我們應用中，代理人不需要與網際網路上的代理人做互動，而且代理人間的溝通單純，因此，我們自行以 Visual C++ (MFC)撰寫簡單伺服平台，此平台只負責簡單的管理與溝通，其類似聊天室一般，只需輸入 IP、port 和 ID 就能連上平台，並獲得相關資訊。其架構如圖 4.8 所示。

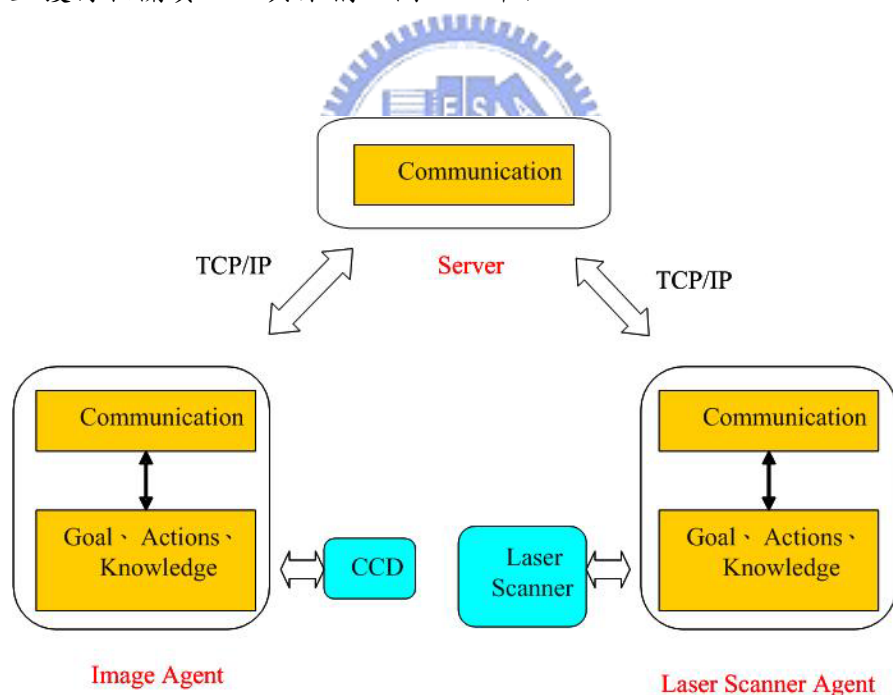


圖 4.8 本論文之多代理人溝通管理平台及架構

另外也自行定義了代理人的溝通語法，能被伺服平台、影像代理人、以及雷射掃瞄器代理人所認知，透過此語法的交流可讓其他的代理人得到對方的資訊，

其訊息格式如圖 4.9 所示。To Who 與 From Who 記載代理人的名稱，目前在我們的系統中有二個名稱可使用，分別是影像代理人(ImageAgent)與雷射掃瞄器代理人(LaserAgent)；Function 有二種，分別是要求訊息(:RequestData:)或回覆訊息(=>)；Data 則是記載著同步時間、前方車輛的位置、車道線等資訊。其程式介面如圖 4.10 所示。

To Who	From Who	Function	Data
--------	----------	----------	------

- To Who：指出此訊息將傳送給何代理人。
- From Who：指出此訊息由何代理人發送。
- Function：指出此訊息是何種功能。
- Data：傳送給 To Who 的資料。

圖 4.9 本論文提出的代理人溝通語法格式

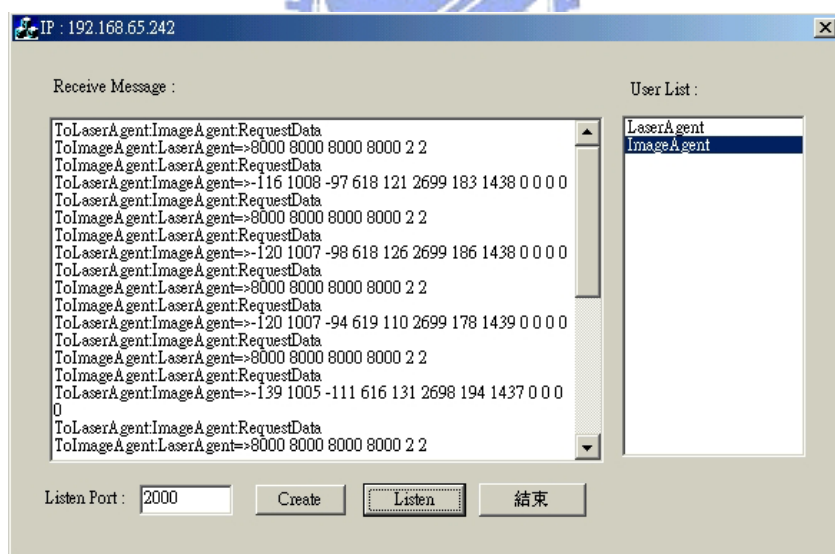


圖 4.10 影像及雷射掃瞄器代理人溝通管理伺服器程式

此外，在伺服器管理平台方面，我們也做了些微的修改。之前訊息傳遞方式為廣播式，如圖 4.11(a)所示，當 Agent 1 欲透過 Server 傳遞訊息給 Agent 2 時，Server 的作法是將此訊息以廣播的方式傳給所有連線的代理人，這種方式會使 Agent 2

以外的代理人平白地多接收並處理無用的訊息，同時也會造成整個網路的負擔，當之後的系統再加幾個代理人時，訊息的混亂和網路延遲的現象會更嚴重。因此，我們在 Server 端加上判斷的功能，如圖 4.11(b)所示，使訊息準確的送至目標代理人。表 4.1 條列出目前代理人間會傳遞的訊息，其中因為雷射掃瞄器代理人(LaserAgent)掃描速度較快，並且對於前方障礙物具有較穩健的表現，故會將最近的 20 筆資料儲存起來，以供影像代理人(ImageAgent)要求同步訊息時搜尋用，而箭頭方向代表要求訊息的流向。

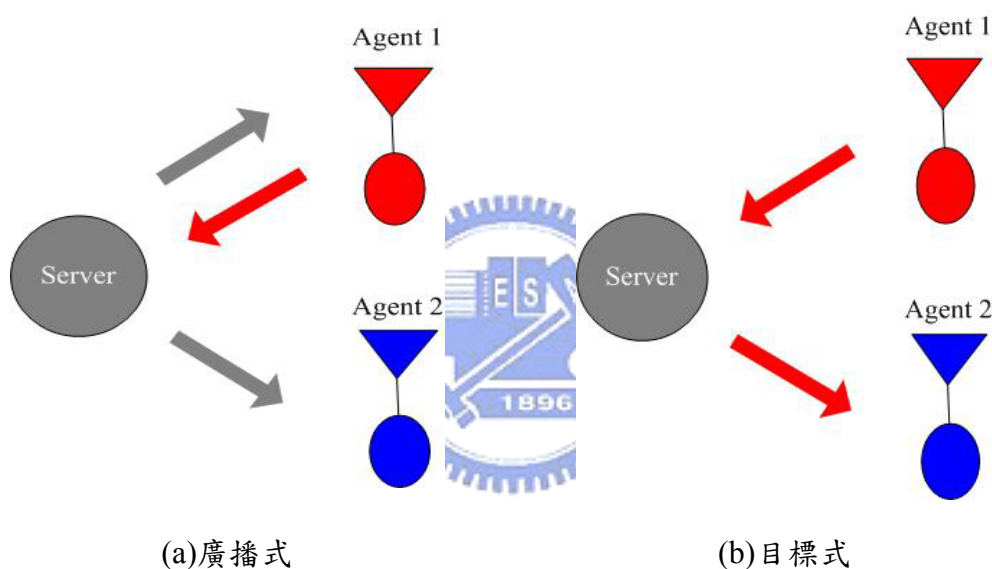


圖 4.11 訊息傳遞示意圖

表 4.1 代理人間的傳遞訊息表

ImageAgent		LaserAgent
每 50 張影像要求一次訊息	→	回傳最新訊息
要求訊息	→	回傳最新訊息
要求同步的訊息	→	回傳同步訊息
回傳最新訊息	←	每 40 筆資訊要求一次訊息
由駕駛人發出訊息	→	(未定義)
(未定義)	←	由駕駛人發出訊息

### 4.3 整合能力展現

如今在我們的系統已有影像代理人、雷射掃瞄器代理人、與溝通協定，並且各代理人已具有基本的輔助功能；影像代理人具有防撞與防偏警示，而雷射掃瞄器代理人具有防撞警示。然而本系統進一步的要整合此二代理人，接下來的小節會舉出數個在行車時常會發生的情況，說明當只有單一代理人做輔助時會發生的問題，以及經由與其他代理人溝通合作後獲得的改善。以下雷射掃瞄器代理人簡稱雷射代理人。

為了方便說明，再次回顧本系統代理人間的傳遞訊息表，並分別給予每個訊息代號(A)~(F)，如表 4.2。

表 4.2 代理人間的傳遞訊息表

	ImageAgent		LaserAgent
(A)	每 50 張影像要求一次訊息	→	回傳最新訊息
(B)	要求訊息	→	回傳最新訊息
(C)	要求同步的訊息	→	回傳同步訊息
(D)	回傳最新訊息	←	每 40 筆資訊要求一次訊息
(E)	由駕駛人發出訊息	→	(未定義)
(F)	(未定義)	←	由駕駛人發出訊息

### 4.3.1 雷射代理人輔助影像代理人

Model\_1~ Model\_6 是針對路況所作的輔助，Model\_7~ Model\_8 是針對日夜決策方面所作的輔助。

Model\_1: 問題: 前方若無車輛時，仍讓影像代理人去執行前方車輛偵測，可能會造成誤判的情形發生。

輔助: 訊息類別(A)。

由雷射代理人適時提供前方是否有物體的資訊，依此決定是否啟動車輛偵測，這可減少影像偵測系統的運算量，並減少誤判的發生，提升系統的穩健性。

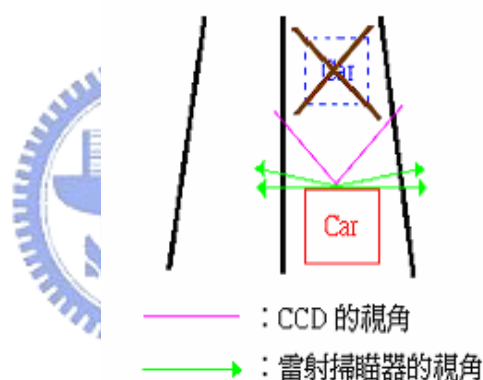


圖 4.12 影像與雷射代理人合作 Model\_1 之示意圖

Model\_2: 問題: 因前方車輛太過接近時，物體容易佈滿 CCD 的視角，並遮蔽車道線，易造成車道線及車輛偵測上判斷的錯誤。

輔助: 訊息類別(A)。

由雷射代理人適時提供前方物體是否太過接近的訊息，進而停止影像代理人的偵測動作，以減少系統誤判的發生。



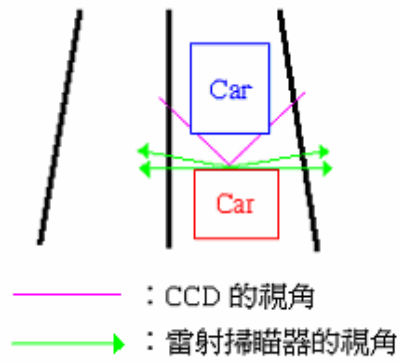


圖 4.13 影像與雷射代理人合作 Model\_2 之示意圖

Model\_3: 問題: 影像代理人在執行前方車輛偵測時，是採取動態的偵測視窗，但常因環境的變化(例如震動)而不能很正確又持續地抓住車輛，有時必須將偵測視窗放大再去抓取可能的車輛，這樣將會增加運算量，且容易產生錯誤。

輔助: 訊息類別(A)。由雷射代理人適時提供前方距離資訊，利用此資訊調整偵測視窗的大小，減少運算時間並預防錯誤的發生。

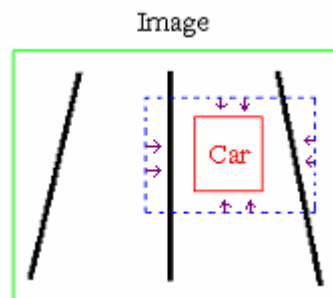


圖 4.14 影像與雷射代理人合作 Model\_3 之示意圖

Model\_4: 問題: 影像代理人是依所得的車輛位置之變化，來提出車輛過於接近的警告訊息，但由於車輛偵測的方法並不能很穩定抓取前方車輛，有時會有劇烈的變化，而造成誤判的可能。

輔助: 訊息類別(C)。

由雷射代理人適時提供前方的距離變化資訊，驗證防撞警告的正確性，減少錯誤警告的發生，以提升系統的穩健性。

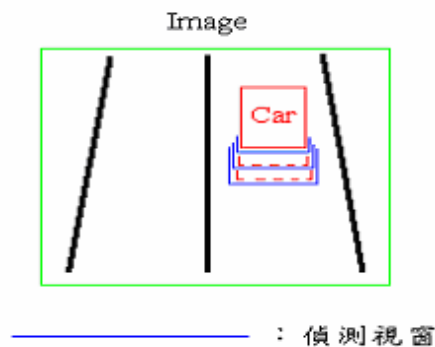


圖 4.15 影像與雷射代理人合作 Model\_4 之示意圖

Model\_5: 問題: 影像代理人在偵測車道線時，若有車輛遮蔽住車道線，會造成部份車道線判斷錯誤，影響之後的防偏警告及運算速度。

輔助: 訊息類別(A)。

由雷射代理人提供車道線上是否有遮蔽物之資訊，利用此資訊減少車道線偵測錯誤的發生。

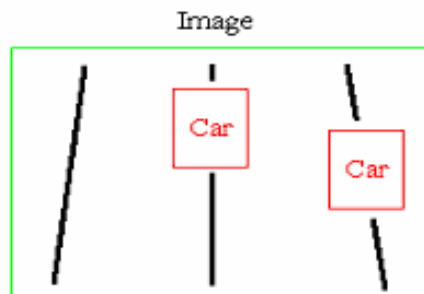


圖 4.16 影像與雷射代理人合作 Model\_5 之示意圖

Model\_6: 問題: 影像代理人在前方車輛切換車道後，無法直接的變換偵測視窗到另一車輛，須放大偵測視窗作抓取車輛的動作，因而增加運算量及錯誤的發生。

輔助: 訊息類別(A)。

由雷射代理人適時提供駕駛人所在的車道上前方車輛大小與距離的資訊，此資訊可以輔助影像代理人在前車變換車道時，能順利的變換所抓取的前方車輛之偵測視窗，可以減少相關的運算量及抓取減少錯誤。

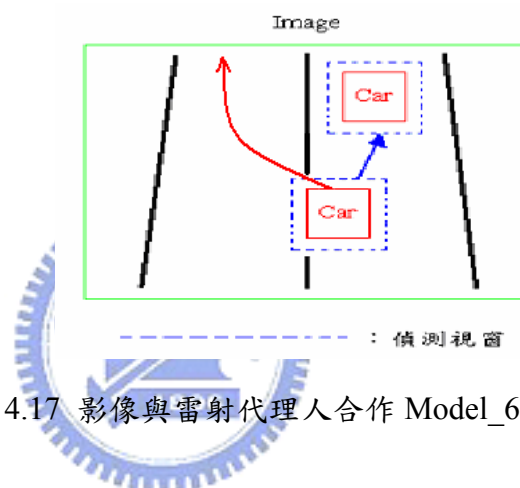


圖 4.17 影像與雷射代理人合作 Model\_6 之示意圖

Model\_7: 問題: 延續 Model\_3。影像代理人無法在雷射代理人提供的偵測視窗中找到前方車輛，導致無法對前方路況做有效監控，防撞警示無法發揮功能。

輔助: 訊息類別(B)。

當區域視窗搜尋失敗後，影像代理人會馬上再向雷射代理人要求最新訊息，並再做偵測視窗搜尋，仍然失敗後，我們判斷此時處於黃昏或清晨等昏暗不明的狀況，於是切換日夜模式。

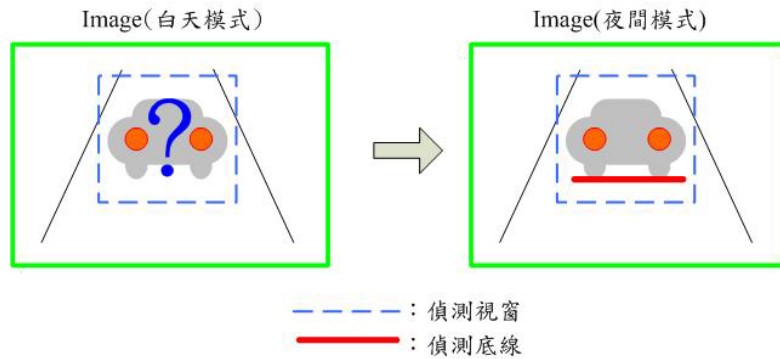


圖 4.18 影像與雷射代理人合作 Model\_7 之示意圖

Model\_8: 問題: 延續 Model\_7，切換日夜狀況後影像代理人仍無法在雷射代理人提供的偵測視窗偵測到前方車輛，此時會一直重複 Model\_7 的輔助方式，不斷的切換日夜演算法，這不僅無法對前方路況做有效監控，也增加系統誤判的機會。

輔助: 訊息類別(B)。

我們判斷此時為白天與夜間演算法的盲點時刻，於是系統會暫時停掉影像代理人的車輛偵測功能，並增加雷射代理人的警示比重，以其負責防撞警示。

#### 4.3.2 影像代理人輔助雷射代理人

Model\_9: 問題: 雷射代理人在有前方物體異常快速接近時會發出警告，但有時是與另一車道上的車輛會車或是在轉彎處的物體，有可能造成錯誤的警告訊息。

輔助: 訊息類別(D)。

由影像代理人提供車道線的資訊，利用此資訊來輔助判斷是否要提出警告訊息，減少錯誤的發生。

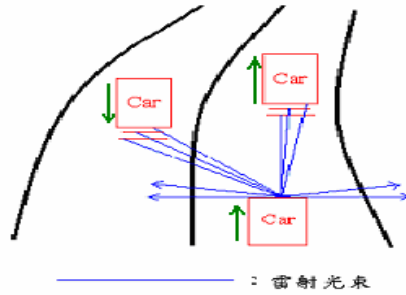


圖 4.19 影像與雷射代理人合作 Model\_9 之示意圖

Model\_10: 問題: 雷射代理人利用 D/V 曲線來提出碰撞警告，但由於雷射掃瞄器對水平的要求較嚴苛，當有稍大幅度的振動，容易極大或極小的相對速度值而導致產生錯誤的碰撞警告。

輔助: 訊息類別(D)。

由影像代理人適時提供前方車輛在影像上是否有接近的趨勢，以輔助雷射代理人作出正確的判斷。

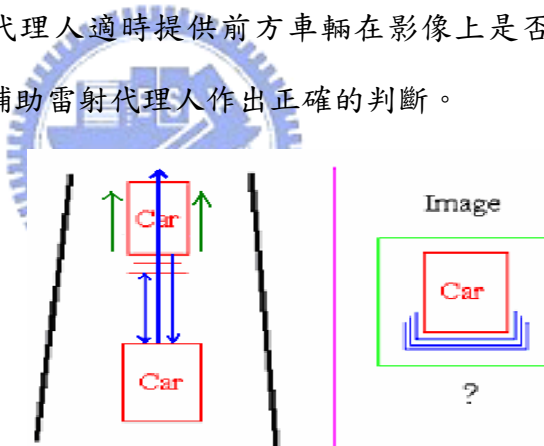


圖 4.20 影像與雷射代理人合作 Model\_10 之示意圖


由上述幾個應用例子來看，利用代理人間的相互溝通可避免一些不必要的錯誤、減少部份不須要的程式流程，因而增加系統的穩健性及運算速度，使其具有更好的可靠性及效率。另外我們觀察到，在合作的 10 個 Model 中，有八個是屬於雷射代理人輔助影像代理人，僅有二個是影像代理人輔助雷射代理人，這是由於對於雷射掃瞄器而言，其所擷取的資訊是屬於單純(只有相對距離)且準確，可是能達到的輔助功能是有限的；CCD 所擷取的影像資訊是屬於豐富且雜亂，而

且經過程式處理後所得結果往往具有誤差，可是其所能達到的輔助功能卻是多樣的，因此，針對好的感測器(如 CCD)，如有雷射掃瞄器此種準確的感測器來輔助之，則會有事半功倍的效果出現。

#### 4.4 系統架構與軟體處理流程

就一個防偏、防撞系統而言，因我們必須將其實現在車輛行駛中的狀況下，即時地處理，並在必要時適當地提出警告。在即時處理與判斷方面，我們並沒有以硬體的方式來實現，而是以軟體來做處理與判斷，故在本節中概述我們的系統架構及軟體處理流程。

##### 4.4.1 系統架構



整個系統所需的硬體包括一台 USB 埠 CCD、二部筆記型電腦、一台雷射掃瞄器、一台 HUB 及一台 UPS。我們所使用的影像擷取設備是一台簡單的視訊會議 USB 埠 CCD，如圖 4.21 所示，將其連接到 PC 上後，完成 PC-based 之影像處理系統。而這部 CCD 其解析度最高為 640×480，擷取速度最高可達每秒 30 個張影像(frame)。而雷射掃瞄器如圖 3.1(a)所示，為紅外線雷射光，其使用串列埠 (RS232、RS422)與 PC 連接，以 RS232 的最快傳輸速度為 38400 baud-rate，以 RS422 的最快傳輸速度為 500k baud-rate。由於兩台 PC 要以區域網路作溝通，因此需要一台 HUB，利用網路線與兩台 PC 作連接，而因為雷射掃瞄器與 HUB 皆需供應電源，所以要準備一台 UPS，如圖 4.22 所示。



圖 4.21 USB 埠 CCD



圖 4.22 UPS 和 HUB

CCD 與雷射掃瞄器是配置在八人座的高爾夫球車上，其配置示意圖如圖 4.23 所示，將各裝置架設好後，由於 IPM 方法需要一些物理參數，其量測方式如 2.3.1 節所述。

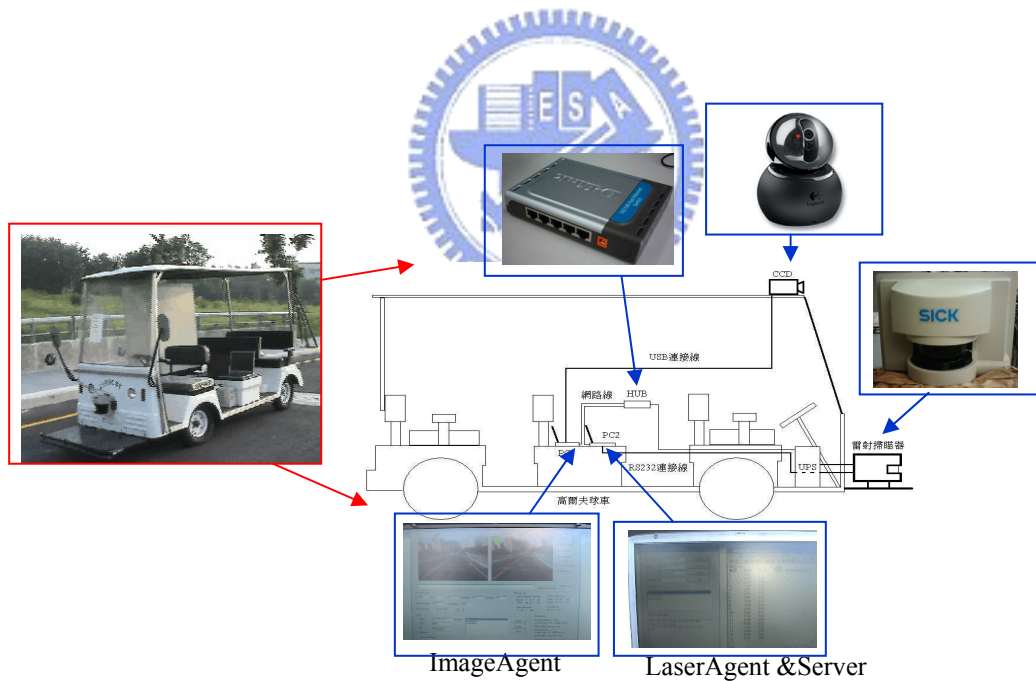


圖 4.23 系統設備的配置圖

#### 4.4.2 軟體處理流程

在軟體方面，使用 Visual C++ 中的 MFC 撰寫相關的視窗程式，其包含影像代理人程式、雷射掃瞄器代理人程式及簡單的溝通管理伺服程式。

影像代理人程式以及雷射掃瞄器代理人程式分別已於 2.5 節、3.3 節介紹，而溝通管理伺服程式也在 4.2.2 小節詳細說明過，而經由 4.3 節的說明，展現出利用多代理人溝通合作觀念，來改善系統穩健性及處理效能。於是，整個全天輔助系統的程式流程可由圖 4.24 表示，並可進一步的簡易成如圖 4.25 之概念圖。此程式流程再次的驗證此架構的可擴充性，在未來的發展裡，如要增加感測器或是代理人是容易達成的，只要其代理人可與伺服平台溝通、其語言能被其他代理人解讀即可，不用擔心感測器內部處理程式與外界的相容性。

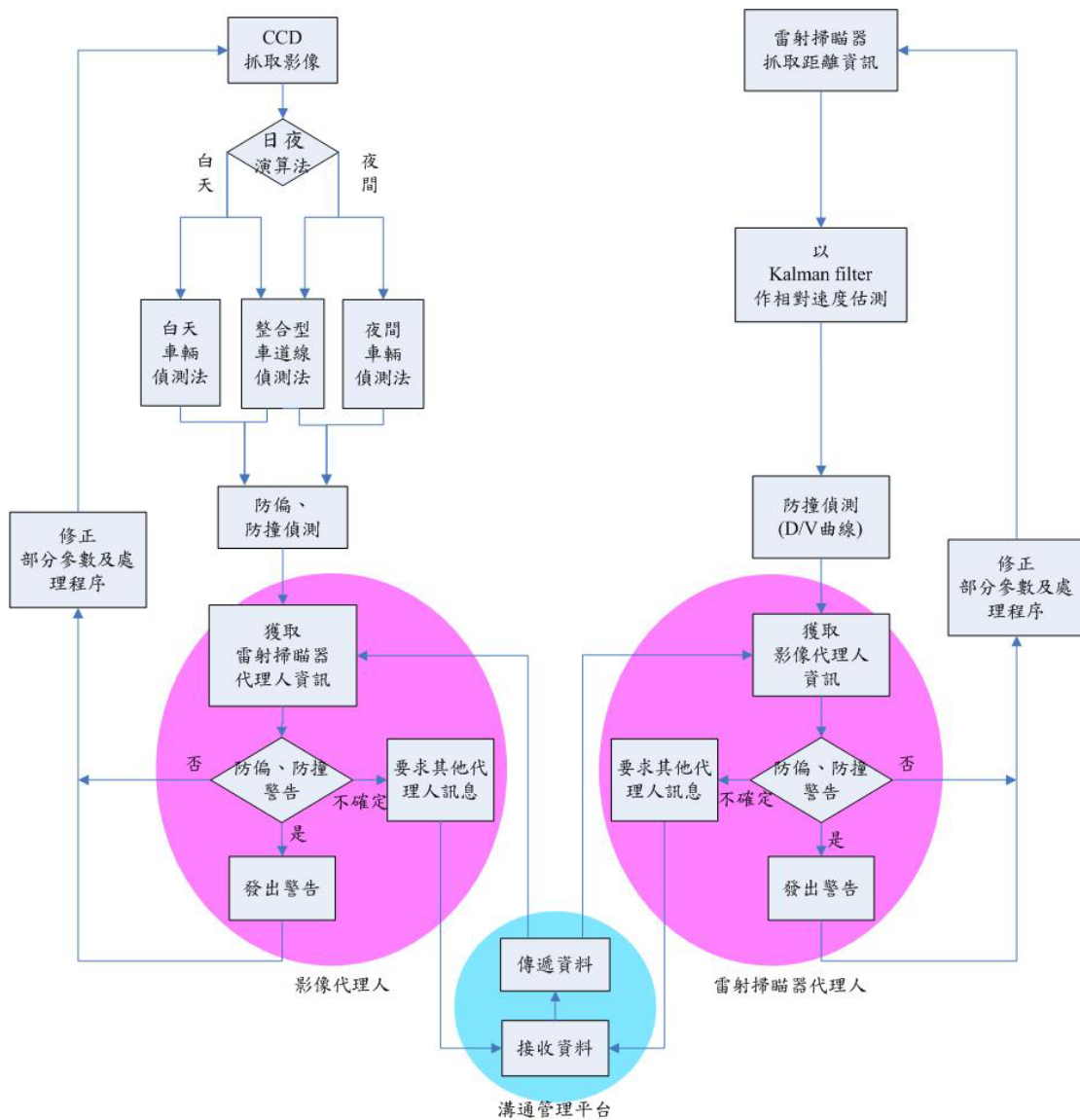


圖 4.24 系統程式流程



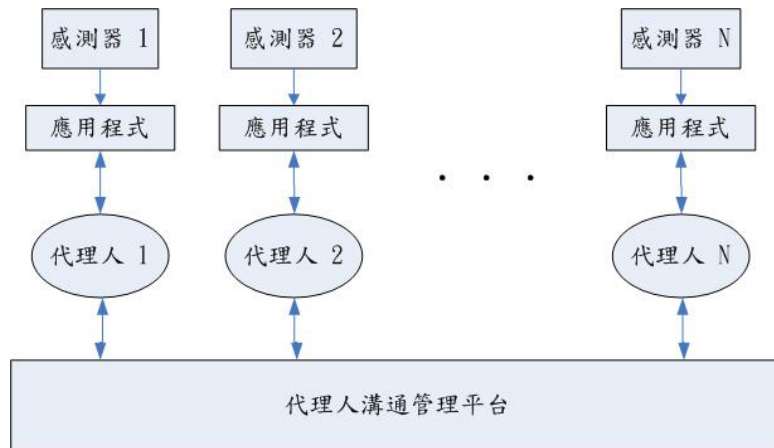


圖 4.25 多感測器多代理人系統示意圖

整個系統的處理速度，CCD 方面，以 Celeron 1.5 GHz 的筆記型電腦以軟體方式處理，每秒約可以處理 3~5 張圖；而雷射掃瞄器方面，以 PC 串列介面 (RS232)、Baud Rate = 38400 的方式接收，每秒約可以處理 3~4 筆訊息。若以整體處理速度大約 0.30 秒來看，應足適時的提出警告，但仍需視車速與希望預防多少意外的發生。一般來說，若預先在 0.5 秒前發警報則可預防 30%~60% 的意外，若在 1.0 秒前發警報則可預防 60%~90% 意外[30]。若車輛有危險的情況時，且以等速度前進時將在行駛 20 公尺後會有意外發生，若要降低至少 30%~60% 的意外，則其車子的速限為時速 120 公里，若要降低至少 60%~90% 的意外，則其車子的速限為時速 65 公里，當然若要預防更短距離的意外時，其速限自然更低。若將來我們以更高等級的電腦來做處理，或以硬體的方式實現將可更快且處理速度更穩定。

## 第五章 結論與未來發展

### 5.1 結論

本論文整合了影像與雷射掃瞄器，架構於多代理人(MAS)之上，有效利用其溝通合作之觀念，藉由車道線與車輛的偵測，實現防偏及前方防撞警示系統，提供駕駛人一個全天的輔助系統，以減少因偏離車道線和前方追撞的事故發生。本論文中，在影像處理與多代理人系統整合二方面之貢獻如下：

一、影像系統方面提出了：

1.一套有效地夜間車輛偵測方法，針對前方車輛共有的特徵-車尾燈-抓出其所在位置。其包含：

(1) 基於夜間影像資訊少且易受光源影響，在前處理步驟中提出 RGB→YCbCr 色度座標轉換，再利用條件式二值化去除大部分非車尾燈的雜訊，使之後的車尾燈辨識步驟更加容易，而得到較高的辨識率。

(2) 利用局部性樣本比對達到車輛追蹤處理，對於已尋得的車輛具有較佳的辨識率，且因為不用一再地重新搜尋而避免掉不必要的錯誤偵測，以及較佳的處理速度。

2.不用替換感測器的全天影像偵測方法，並以 Otsu 演算法為基礎，利用 Otsu 明亮度和平均明亮度為指標的日夜決策法則，整合白天與夜間演算法使其能準確、順暢的切換。

## 二、雷射掃瞄器的應用方面：

提出以一維 Kalman filter 得到穩定的相對距離、相對速度及 D/V 值，並利用簡單的原點變換，建立虛擬雷射掃瞄器來擴展監控範圍，建立一穩健的防撞偵測系統。

## 三、整合影像與雷射之多代理人系統整合方面

### 1. 管理溝通平台：

(1) 明確定義自定的代理人溝通語言(ACL)，有助於多代理人系統的發展及管理。

(2) 訊息傳遞目標化，免除代理人接收不必要的訊息，更有助於降低網路流量負載及因訊息碰撞而遺失的現象。

2. 確實減少如本論文 4.3 節所提到的，單一的影像或雷射掃瞄器，無法解決單獨的問題 (Model\_1~Model\_10)，互補各兩者間不足的地方，並由於管理溝通平台的建立，更容易增加其它代理程式，使得系統更具有彈性。

## 5.2 未來發展

1、目前在影像與雷射掃瞄器的處理方面，我們是以 PC based 來實現，但在未來發展可以將其移植到其微電腦處理器上，如、DSP 等，這更能更確保其即時性，也較節省成本且較實用，另外亦能解決在 Windows 下，因系統有時使用部份資源去執行其它程式，造成處理速度並不是都維持一定。

2、目前輔助系統只有二個感測器，雖已能解決不少路況情形，但限於類似高速公路的單純環境，對於複雜的城市交通仍顯不足，例如行人、號誌燈、複雜的車道線等等，因此仍需其他類型感測器(聲納感測器、紅外線感測器等等)

的加入，以及更周詳的影像偵測技術。同樣的，感測器多或溝通訊息種類增加的情況下，在代理人溝通合作方面，其伺服平台便需建立更良好的溝通方式及管理環境，使得輔助系統不會因增加複雜度而導致效能降低。

- 3、由所得的車道線與前方車輛的資訊，再加入相關車輛控制，以達到自動駕駛的目的。



## 參考文獻

- [1] M. Tomizuka, “Advanced Vehicle Control Systems (AVCS) Research for Automated Highway Systems in California Path”, *Proceedings of the IEEE Vehicle Navigation and Information Systems*, pp. 41-45, 1994.
- [2] S. E. Shladover, “Research and Development Needs for Advanced Vehicle Control Systems”, *IEEE Micro*, Vol. 13, pp. 11-19, Feb., 1993.
- [3] I. Masaki, “Machine-Vision Systems for Intelligent Transportation Systems”, *IEEE Intelligent systems*, pp. 24-31, 1998.
- [4] 電子公共安全，交通運輸管理，全年交通事故肇事原因統計表  
[http://www.motc.gov.tw/business\\_5\\_1a3.htm](http://www.motc.gov.tw/business_5_1a3.htm)
- [5] 內政部警政署全球資訊網，道路交通事故狀況分析  
<http://www.npa.gov.tw/maindefault.asp>
- [6] G.P. Lekkas, N.M Avouris, G.K. Papakonstantinou, “Development of Distributed Problem Solving Systems for Dynamic Environments,” *IEEE Trans. on Systems, Man, and Cybernetics*, Vol. 25, pp. 400-414, March 1995.
- [7] Bertozzi, M.; Broggi, A.; Cellario, M.; Fascioli, A.; Lombardi, P.; Porta, M.; “Artificial vision in road vehicles” *Proceedings of the IEEE*, Volume 90, Issue 7, pp. 1258 – 1271, July 2002.
- [8] M. Betke, E. Haritaoglu, L.S. Davis, “Real-time multiple vehicle detection and tracking from a moving vehicle”, *Machine Vision and Applications*, pp. 69-83, 2000.
- [9] J. Gangyi, Y. Mei, H. Sailong, S. Shoudong, L. Xiao, “A New Method for Obstacle Detection in ITS”, *Proceedings of the Conference on IEEE*, pp. 1770-1773, 2002.

- [10] S. Nagumo, H. Hasegawa, N. Okamoto, "Extraction of Forward Vehicle by Front-Mounted Camera using Brightness Information", *Proceedings of the Conference on IEEE*, pp. 1243-1246, 2003.
- [11] T. Tsuji, H. Hattori, M. Watanabe, N. Nagaoka, "Development of Night-Vision System", *IEEE Transac. On Intelligent Transportation System*, Vol. 3, No. 3, September 2002.
- [12] M.Y. Chern, P.C. Hou, "The Lane Recognition and Vehicle Detection at Night for A Camera-Assisted Car on Highway", *Proceedings of the Conference on IEEE*, pp. 14-19, 2003.
- [13] M. Kagesawa, S. Ueno, K. Ikeuchi, H. Kashiwagi, "Recognizing Vehicles in Infrared Images Using IMAP Parallel Vision Board", *IEEE Transac. On Intelligent Transportation System*, Vol. 2, No. 1, March 2001.
- [14] R. Cucchiara, M. Piccardi, P. Mello. "Image Analysis and Rule-Based Reasoning for a Traffic Monitoring System", *IEEE Transac. On Intelligent Transportation System*, Vol. 1, No. 2, June 2000.
- [15] C. Thompson, "Everything Is Alive", *IEEE Internet Computing*, pp. 83-86, February 2004.
- [16] Y. Li, S. Ma, W. Li, H. Wang, "Microscopic Urban Traffic Simulation with Multi-Agent System", *Proceedings of the Conference on IEEE*, pp. 1835-1839, December 2003.
- [17] Z. Papp, A.H.C. Thean, M.G. van Elk, M. Dorrepaal, "Multi-Agent Based Simulator with High Fidelity Virtual Sensors", *Proceedings of the Conference on IEEE*, pp. 14-19, May 2003.
- [18] W. Shen, "Distributed Manufacturing Scheduling using Intelligent Agents", *IEEE Intelligent Systems*, Vol 17, pp. 88-94, Jan.-Feb. 2002.

- [19] L. Wang, S. Balasubramanian, D. H. Norrie and R. W. Brennan, "Agent-Based Control System for Next Generation Manufacturing", *Proceedings of the IEEE International Symposium on Intelligent Control*, pp. 78-83, 14-17 Sept. 1998.
- [20] J. C. Fraile, C. J. J. Paredis, C. H. Wang and P. K. Khosla, "Agent-Based Planning and Control of a Multi-Manipulator", *IEEE International Conference on Robotics and Automation*, Vol. 2, pp. 1219-1225, 10-15 May 1999.
- [21] N. R. Jennings and S. Bussmann, "Agent-Based Control Systems", *IEEE Control Systems Magazine*, Vol. 23, pp. 61-73, June 2003.
- [22] H. Wang and C. Wang, "Intelligent Agents in the Nuclear Industry", *Computer*, Vol. 30, pp. 28-31, Nov. 1997.
- [23] N. Otsu, "A Threshold Selection Method from Gray-Level Histograms", *IEEE Trans. Systems, Man, Cybern.*, vol. 9, pp. 62-66, Jan. 1979.
- [24] 吳政衍, Multi-Agent 駕駛輔助系統, 國立交通大學 電機與控制工程學系 碩士論文, 2004.
- [25] Z. Papp et al, "Distributed Hardware-In-the-Loop Simulator for Autonomous Continuous Dynamical Systems with Spatially Constrained Interactions", *Proc. of the International Parallel and Distributed Real-Time System*, Nice, France, June 2003
- [26] R.C.Gonzalez, R.E.Woods著, 吳成柯等譯,"數位影像處理",儒林圖書, 90年 10月.
- [27] FIPA.FIPA Interaction Protocol Library Specification. 2001  
<http://www.fipa.org/specs/fipa00025/XC00025D.html>[J/OL]
- [28] 楊豐兆等,"Jini社群環境下FIPA代理人的服務品質衡量機制之研究與設計",ITIS產業論壇,  
<http://www.if.itri.org.tw/content07/04if019a.htm>

[29] JADE, Java Agent DEvelopment Framework.

<http://jade.tilab.com/>

[30] 歐文雄、歐家駿,“工業電子學”,全華圖書,四月,1992

