

國立交通大學

電機與控制工程學系

碩士論文

影像追蹤系統及其在車輛安全之應用

Visual Tracking System and Its Applications to
Vehicle Safety



研究生：楊智傑

指導教授：宋開泰 博士

中華民國九十三年七月

影像追蹤系統及其在車輛安全之應用

Visual Tracking System and Its Applications to Vehicle Safety

研究生：楊智傑

Student: Chih-Chieh Yang

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 Engineering and Computer Science

National Chiao Tung University

in Partial Fulfillment of the Requirements

for the Degree of Master

in

Electrical and Control Engineering

July 2004

Hsinchu, Taiwan, Republic of China

中華民國九十三年七月

影像追蹤系統及其在車輛安全之應用

學生:楊智傑

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

本論文發展一套應用於車輛安全之影像追蹤系統。文中提出一基於特徵追蹤法並結合擴散焦點(FOE)及場景分析之影像追蹤系統，此系統可自動偵測並同時追蹤多個動態物體，包括汽車及機車，並可在複雜的環境中有效地分離背景與動態物體，以提供碰撞偵測之資訊。本論文採用 CMOS 影像感測器及 Nios embedded processor 將此系統發展為一獨立式系統，使之能在車輛上獨立運作。我們將此獨立式影像追蹤系統置於車中進行實際測試。測試之結果顯示本系統具有追蹤車輛並提供碰撞偵測之資訊。

Visual Tracking System and Its Applications to Vehicle Safety

Student: Chih-Chieh Yang

Advisor: Dr. Kai-Tai Song

Department of Electrical and Control Engineering
National Chiao Tung University



ABSTRACT

In the thesis, a visual tracking system for vehicle safety has been developed. We proposed a feature-based tracking system, which is combined with the concept of FOE and scenario analysis. This system can automatically detect and track multiple moving objects, including cars and motorcycles. The developed system can segment features of moving objects from moving background simultaneously and provide useful information of collision warning. Furthermore, we have realized the system in a stand-alone system, which can operate independently in the vehicle using a CMOS image sensor and Nios embedded processor architecture. The stand-alone visual tracking system has been validated in actual road environment. Experimental results show that the proposed methods successfully tracks front vehicles and provides information of collision warning.

誌謝

謹向我的指導教授宋開泰博士致上感謝之意，感謝他兩年來在專業上的指導，以他豐富的學識與經驗，配合理論的應用，使得本論文得以順利完成。

感謝與我共同奮鬥的同學嘉豪、志豪、耀慶及栢瀚的相互鼓勵及提攜，以及學弟柏秋、恩捷、松峙、維祥、崇民及鎮源在生活上帶來的樂趣，同時感謝學長任詔、奇謚及孟儒的指導，另外感謝茂綸公司周志峰先生、周怡慧小姐及國家晶片系統設計中心吳錫澤先生的幫忙。

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，使得我得以順利完成此論文，在此我願以此論文獻給我
最感激的父母親。



目錄

摘要	I
ABSTRACT	II
誌謝	III
目錄	IV
圖例	VI
表格	X
第一章 緒論	1
1.1 研究動機	1
1.2 相關研究回顧	2
1.3 問題描述	5
1.4 章節說明	5
第二章 影像追蹤系統	7
2.1 場景分析	7
2.1.1. 影像流與特徵流	7
2.1.2. 擴散焦點(Focus of Expansion).....	9
2.1.3. 觀察者與移動中物體的相對運動	9
2.1.4. 道路場景分析	10
2.2 系統架構	10
2.3 Harris 角點偵測	13
2.4 角點比對	16
2.4.1. 特徵點運動預測(Feature motion prediction)	16
2.4.2. 角點比對	17
2.5 特徵流分析	18
2.6 FOE 估測	20
2.7 相關性追蹤	21
2.8 碰撞偵測	23
第三章 獨立式影像追蹤系統設計	24
3.1 影像追蹤系統之硬體架構	24
3.2 CMOS 影像感測器介紹	26
3.3 Nios 發展介面設計	27

3.4	影像擷取介面設計	27
3.5	Nios 之規劃與設計	32
3.6	Harris 角點偵測硬體設計	36
3.7	Nios 之軟體設計	43
3.7.1.	內部中斷向量設定與記憶體存取	43
3.7.2.	角點比對	43
3.7.3.	影像流分析	43
3.7.4.	相關性追蹤與碰撞偵測	44
第四章	實驗結果	45
4.1	Matlab 模擬影像追蹤系統結果	45
4.2	獨立式系統驗證	48
4.2.1.	CMOS 影像感測器擷取實驗結果	48
4.2.2.	Harris 角點偵測硬體之實驗結果	49
4.3	實車測試結果	50
4.4	實驗討論	58
第五章	結論與未來展望	59
5.1	結論	59
5.2	未來展望	60
	參考文獻	61
	附錄 A NIOS DEVELOPMENT KIT, STRATIX EDITION 規格	64
	附錄 B ICM 205B 之規格	66

圖例

圖 1-1 (a)擴張焦點(FOE, Focus of expansion) (b)Inner FOE window 與 Outer FOE window[2].....	4
圖 2-1 三維空間之物體投影於影像平面之示意圖	8
圖 2-2(a)觀察者之注視方向(line of sight)與移動方向相同之示意圖(motion direction) (b)此時觀察者在影像平面(image plane)所看到的影像流。	8
圖 2-3 四種基本運動型式	10
圖 2-4 道路場景分析(側視圖).....	11
圖 2-5 道路場景分析(上視圖).....	11
圖 2-6 車前所見之影像.....	11
圖 2-7 系統架構圖	13
圖 2-8 局部區塊於影像中的位置。	14
圖 2-9 特徵點運動預測示意圖	17
圖 2-10 GVM 角點比對示意圖	18
圖 2-11 移動中物體與道路平面重疊示意圖	19
圖 2-12 道路之特徵流，垂直分量必向下	20
圖 2-13 道路之特徵流，取 I,IV 各一個求 FOE.....	21
圖 2-14 兩道路特徵流之交點即為 FOE 的位置	21
圖 2-15 相關係數之求法	22
圖 2-16 碰撞偵測範圍	23
圖 3-1 影像追蹤系統之硬體架構圖	25
圖 3-2 ICM205B 之外觀.....	26
圖 3-3 Nios development kit, Stratix edition	28
圖 3-4 輸出影像格式[21].....	28
圖 3-5 ICM205B 輸出腳位圖[21].....	29
圖 3-6 ICM205B 之時序圖(1)[21].....	29
圖 3-7 ICM205B 之時序圖(2)[21].....	30
圖 3-8 ICM205B 之時序圖(3)[21].....	30
圖 3-9 Dual clock FIFO.....	31
圖 3-10 中斷訊號 ff_irq 電路圖	32

圖 3-11 CMOS 取像時序圖	33
圖 3-12 SOPC 之操作介面	34
圖 3-13 DMA 運作示意圖	34
圖 3-14 DMA 內部設定	35
圖 3-15 School Boy 研發電路板之外觀	36
圖 3-16 Harris 角點偵測之腳位	36
圖 3-17 Harris 內部電路	39
圖 3-18 角點與水平、垂直灰階梯度之 FIFO	40
圖 3-19 高斯平滑遮罩(1)	40
圖 3-20 高斯平滑遮罩(2)	40
圖 3-21 3x3 mask generator 內部電路	41
圖 3-22 7x7 mask generator	42
圖 4-1 影像追蹤系統之外觀	45
圖 4-2 影像追蹤結果(1)	46
圖 4-3 影像追蹤結果(2)	46
圖 4-4 影像追蹤結果(3)	46
圖 4-5 影像追蹤結果(4)	46
圖 4-6 影像追蹤結果(5)	47
圖 4-7 影像追蹤結果(6)	47
圖 4-8 影像追蹤結果(7)	47
圖 4-9 影像追蹤結果(8)	47
圖 4-10 影像追蹤結果(9)	47
圖 4-11 影像追蹤結果(10)	47
圖 4-12 影像追蹤結果(11)	48
圖 4-13 影像追蹤結果(12)	48
圖 4-14 CMOS 道路取像結果	48
圖 4-15 CMOS 室內取像結果	48
圖 4-16 原始影像	49
圖 4-17 角點偵測結果(1)	49
圖 4-18 角點偵測結果(2)	49
圖 4-19 角點偵測結果(3)	50
圖 4-20 角點偵測結果(4)	50

圖 4-21 角點偵測結果(5)	50
圖 4-22 角點偵測結果(6)	50
圖 4-23 角點偵測結果(7)	50
圖 4-24 角點偵測結果(8)	50
圖 4-25 影像追蹤系統置於實車內之情況	51
圖 4-26 與前車距離遠	52
圖 4-27 前車接近中	52
圖 4-28 前車接近中	52
圖 4-29 前車接近中	52
圖 4-30 前車接近中	52
圖 4-31 系統發出碰撞警示燈	52
圖 4-32 系統發出碰撞警示燈	53
圖 4-33 系統發出碰撞警示燈	53
圖 4-34 系統發出碰撞警示燈	53
圖 4-35 系統發出碰撞警示燈	53
圖 4-36 前車開始遠離	53
圖 4-37 前車遠離	53
圖 4-38 前車遠離	54
圖 4-39 前車接近，系統警示	54
圖 4-40 系統發出碰撞警示燈	54
圖 4-41 系統發出碰撞警示燈	54
圖 4-42 前車遠離	54
圖 4-43 前車遠離	54
圖 4-44 前車接近，系統警示	55
圖 4-45 前車接近，系統警示	55
圖 4-46 與前車近，系統警示	55
圖 4-47 與前車近，系統警示	55
圖 4-48 與前車近，系統警示	56
圖 4-49 與前車近，系統警示	56
圖 4-50 前車遠離，但仍在警示範圍	56
圖 4-51 前車遠離，警示燈熄滅	56
圖 4-52 前車遠離	56



圖 4-53 前車遠離	56
圖 4-54 前車遠離	57
圖 4-55 前車遠離	57
圖 4-56 前車遠離	57
圖 4-57 前車接近中	57
圖 4-58 前車接近中	57
圖 4-59 前車進入警示範圍	57
圖 4-60 前車進入警示範圍	58
圖 4-61 前車進入警示範圍	58
圖 4-62 前車進入警示範圍	58
圖 4-63 前車進入警示範圍	58



表格

表 1 本論文所用之符號定義	7
表 2 Nios 系統所規畫的元件	35
表 3 本系統之處理效能	59
表 4 Stratix EP1S10F780C6ES 之規格	65



第一章 緒論

1.1 研究動機

在智慧型運輸系統(Intelligent Transportation System)的研究項目，先進車輛控制與安全系統(Advanced Vehicle Control and Safety System, AVCSS)的許多研究重點大都致力於如何避免行車意外的發生[1]。各個國家也投入很多的資金與人力來改進道路安全與增加現在道路的效率[2]。如美、日兩國的 Intelligent Vehicle Highway System(IVHS)、歐洲的 Road Transport Informatics(RTI)等。這些系統可接收車子周圍的多種感測器所感測到的資訊以及與其它車輛或道路上的交通監測系統來交換的資訊，將這些資訊做一統整性的判斷。而先進車輛控制與安全系統可以是駕駛人的行車輔助系統(Driver assistant system)或是自主式車輛系統(Autonomous vehicle system)。行車輔助系統能偵測行車周圍的其它車輛及障礙物、道路狀況以及駕駛人的精神狀況等等。當一有可能的危險發生時，如其它車輛或障礙物的接近、路面不平整或駕駛人的精神不濟，行車輔助系統會適時的提醒駕駛人，達到安全行車的目的。自主式車輛系統更可不需經過駕駛人的操控，維持安全的行車，避免危險的發生。

自主式車輛具有兩大主要功能。第一，能自主式地行駛於道路上，此主要為一側邊控制的問題。第二，能與前方車輛(前導車或其它車輛)保持安全距離，或將前方車輛作為引導車成為跟車系統。此為一縱向控制問題。其中縱向控制問題裡一個關鍵的議題是如何有效地感測周圍的環境，相關的研究包括障礙物之偵測、障礙物範圍估測和整體場景之描述。當有效地感測到周圍的環境後，才能針對不同的環境做適當的反應。而我們需要解決的問題在於如何從一移動中的觀察者(自主式車輛)來偵測與其它車輛的相對位置？即為在動態背景中如何即時地偵測動態物體，且能同時追蹤多個動態物體？

目前市面上採用置於車上的測距雷達可解決跟前車保持距離的問題[3][4]，然而，價格較昂貴(通常取決於雷達感測器)及其本身性能上的限制(偵測範圍窄和側向解析度低)成為打入市場的阻礙。近幾年來電腦系統的功能愈來愈強大及影像擷取晶片的性能高且價格低廉，使得影像感測器更

趨普及且重要[5]。超音波、雷達、雷射為主動感測器，量測發出的訊號與接收訊號的時間差。然而為採用相同主動感測器會有數量上的限制，才不會使訊號互相干擾，且反射率會因材質而有很大的變化。影像感測器屬於被動感測器，且只有視覺系統可不需在加工的道路上(如在道路裡埋磁感應式感測器)偵測出標線及交通標秩[6]。影像感測器的另一個優點在於可同時追蹤多個不同的物體。綜合以上觀點，本論文將運用影像感測器作為行車狀況之感測介面，發展一影像追蹤系統，可在動態背景中偵測動態物體，且能同時追蹤多個動態物體。

1.2 相關研究回顧

藉由影像感測器追蹤動態物體的問題在文獻中一般分為兩個步驟[7]:

1. 移動估測 (Motion estimation):

由影像感測器所擷取到的二維影像中，估測物體在此二維影像中移動的影像流。

2. 移動詮釋 (Motion interpretation):

將移動估測所得到的影像流還原成在三維空間中的移動量。

為了解決移動估測的問題，在文獻中許多的方法被提出來，可大略分成三種方法:基於光流法之追蹤(Optic flow based tracking)、基於特徵法之追蹤(Feature based tracking)、基於模型之追蹤(Model based tracking)。

基於光流法之追蹤的原理為假設影像的亮度變化平滑且無空間上的不連續，每點亮度在一微小時間間隔內移動一微小位移，假設其亮度為一定值，而導出光流限制式(Image brightness constraint equation)，大部分提出的方法還會結合其他的限制法則，如邊緣資訊[8]、彩色資訊[9]或 Kalman Snakes[10]，利用移動之物體與背景移動的速度的不同，把移動物體從背景中分離出來。光流法的一個優點是可以把速度場很快速的求出。另一項優點為利用一些較多的資料可以使此法更強健。然而，其缺點在於此法需要每秒較多的圖像數與較高的影像解析度來確保結果的正確。另外，Giachetti, et al 等人利用 multi-scale/multi-window 的概念[11]，不需高的時間與空間的解析度即可求出速度場，然而其結果未能達到 real-time。

在基於特徵法之追蹤的研究上，Beymer, et al[12]提出解決車輛重疊問題。以往的做法是追蹤整輛車身，當兩輛車重疊時，因為特徵未能完全比對，導致追蹤失敗。若改以在車身上找出幾個特徵點(此法以角點作為特徵)，即使車輛重疊時，仍可依現有的部分特徵點做追蹤。另一個衍生的問題是如何把這些特徵點作分類呢？意思是如何將這些特徵群組起來說這是同一部車上的特徵而不會與其它的車的特徵混淆。作者依在同一物體上的特徵會有相同的運動行為作為分類。利用多張連續影像的觀察，即使兩輛等速行駛的車輛也不會一直維持相同速度，終究可以分類出特徵。但此法應用在交通監測上，所使用的攝影機為靜止不動的，若考慮移動中的攝影機，則要考慮場景的變化。Roberts 提出裝設在移動中的車輛的攝影機所擷取的到的影像中，偵測且追蹤動態車輛的方法[2]。他認為以高階的影像感知來選取所想要追蹤的區域(Region of interest)，回傳給低階的影像處理(特徵抽取)，做特定區域的特徵抽取，而不用整個影像全盤特徵抽取，使得整個處理加快許多。另外作者也對動態背景的影像流做詳細的分析，即擴張焦點(Focus-of-5xpansion, FOE)的概念，利用 Inner FOE window 和 Outer FOE window 的方法，將影像分割成三個部分，如圖 1-1 所示。Inner FOE window 內的解析度低，並不處理，如圖 1-1(a)。只在兩 window 之間偵測角點，而在 Outer FOE window 外做持續追蹤。Smith 提出 ASSET-2 的架構[13]，特徵(角點)的抽取使用 SUSAN corner detector，特徵的比對從原來的兩張連續影像改為多張連續影像，特徵的分類也從單一影像移動量變為有考慮線性影像流的變化。對於兩輛車重疊時，也有做很好的預測。以上特徵點追蹤法的挑戰在於如何分辨背景與移動物體的影像流的不同。

基於模型之追蹤(Model based tracking)法，這方面的學者希望能把移動中的物體一般化。通常要看應用面，如要用到追蹤車輛，則需要建立一般化的車輛模型，愈能描述多種車型愈好。Leuven, et al 提出用三個 refinements :Kalman Prediction, Multiple Hypotheses, Dynamic Updating，讓此法能在模糊的影像也能分辨。然而此模型只適用於一般的轎車。

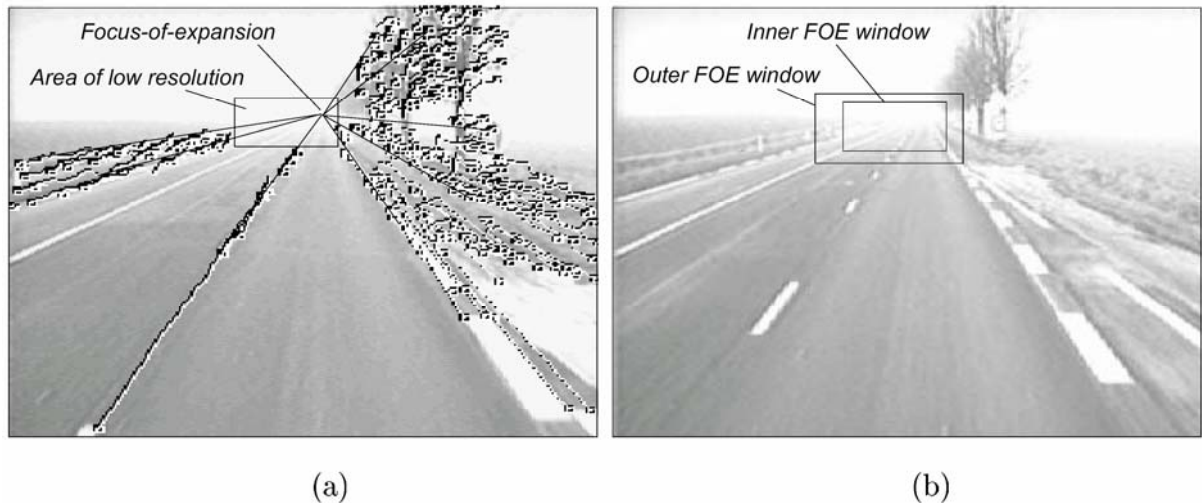


圖 1-1 (a)擴張焦點(FOE, Focus of expansion) (b)Inner FOE window 與 Outer FOE window[2]

林俊宏提出讓一輛車單一的模型分成四個區域作小塊、多個的模型比對[1]。一開始提供使用者能快速選取追蹤目標車輛的功能，以使其更符合實用的需求。前導車追蹤模組使用前一個模組選取的前導車影像作為初始的樣板(Template)，並使用特徵匹配法追蹤目標的位置。另外，利用偵測前車車牌在影像中的大小，來推算兩車的相對距離和相對速度。最後採用了 Fuzzy $\alpha - \beta$ filter 做濾波並預測前車的位置，讓系統更加實用。然而需要手動選取目標車輛，且只有一台，無法達到自動分割多個動態物體。如果車身為白色或亮色，將無法有效地從中找出車牌位置。

綜觀以上的方法，整理各方法之特性：

光流法：

優點：

利用一些較多的資料使方法穩健。

缺點：

1. 由於是對時間與空間的微分，所以抗雜訊低。
2. 對於影像時間與空間的解析度要高。
3. 物體的移動必需是平滑且連續。

特徵法：

優點：由於只找出影像中的特徵點，處理的資料量小。

缺點：

1. 特徵點的挑選必需要明顯且在時間與空間上要穩定呈現。
2. 物體的移動不應包含翻轉的動作，而使特徵點消失。
3. 特徵點的匹配與預測法則需對應用面做不同的設計。

模型比對法：

優點：

1. 即使物體在靜止時也能依所存在的模型找到物體。
2. 抗雜訊能力強，即使在模糊的環境也能比對。

缺點：

模型的建立無法全面性，且車子的形式不可能完全一樣，還有其它如機車、行人等要對每種物體作各別模型是不實際的做法。

1.3 問題描述

綜合以上的方法，有鑑於在台灣的交通，機車占了很重要的部分，為了能同時偵測機車和汽車，建立多個模型來比對較不適合。另一方面，希望能即時處理影像資料，因此選擇特徵點追蹤法作為本論文的主軸。然而特徵點追蹤法所面臨的問題是如何利用這些特徵點在連續圖像間的移動(在此稱為特徵流)來分離移動中物體與背景的特徵點。雖然 Robert[2]提出雙 FOE window 的作法，但對於複雜的背景而言，遠方背景的特徵仍容易與移動中物體的特徵混淆。意思是說離觀察者愈遠的背景的特徵點，從觀察者的角度來看，其特徵點的移動量不大。另一方面，若移動中的物體的行進速度和方向與觀察者相似的話，從觀察者的角度來看，其特徵點的移動量也不大，所以此兩種特徵點並不能只靠移動量來區分。故本論文的目標為發展一能在動態背景中簡單且有效地偵測出動態物體並持續追蹤其特徵點之影像追蹤系統。另外，為了達到易於與自主式車輛結合的目的，本論文將此影像追蹤系統實現於獨立式系統(Stand alone system)，使之能在車輛上獨立運作。

1.4 章節說明

本論文共分五章，第一章為介紹相關研究背景，並說明研究目標。第二章介紹影像追蹤系統，包括場景的分析、特徵點的偵測與其追蹤法則。第三章介紹獨立式影像追蹤系統設計，包括硬體架構、影像擷取介面設計、

角點偵測電路設計、追蹤法則設計。第四章為實驗結果，使用所發展的影像處理系統進行測試。第五章為結論與未來展望。



第二章 影像追蹤系統

在此先將本論文所用的符號定義整理如表 1:

表 1 本論文所用之符號定義

Symbol: Explanation
* : 線性迴旋積分
\cap : 交集運算子
$E_{x,y}$: 灰階值變化量
$w_{u,v}$: 以 (u,v) 為中心點的局部區塊
$I_{u,v}$: 影像中點 (u,v) 的灰階值
x,y : 分別為在 x 方向和 y 方向的移動量
σ : 標準差 (standard deviation)
ρ : 相關係數 (Correlation coefficient)
I_{ij} : 模板影像 I 中第 (i,j) 個像素 (pixel) 之灰階值
M_{ij} : 映射矩陣 M 中第 (i,j) 個像素 (pixel) 之灰階值

2.1 場景分析

在說明所要分析的場景之前，我們需要先了解影像流、特徵流以及觀察者、移動中的物體與背景之間可能的相對運動型式和擴張焦點 (Focus of expansion) 的概念。

2.1.1. 影像流與特徵流

影像流 (Image flow) 是用三維空間中物體在影像平面上的投影點的二維速度來表示三維空間中相對應投射點的速度，以圖 2-1 來說明，一個剛體 B 相對於影像平面作任意方向的運動，而在剛體上有一點 P 具有速度 S ，其世界座標為 (X, Y, Z) ，原點固定於 O_w 。影像平面是以 O_w 作為投影中心 (即觀點)，同時在影像平面上有影像座標系統 (x,y) ，原點 O_I 被世界座標的 Z 軸所通過。則假如影像平面上的一點 p 是空間中點 P 的投影點，則 p 的速度向量 $V=(u, v)$ 就是空間中點 P 速度向量 S 的投影，所以經由前面的定義，速度向量 V 就是 P 點速度向量 S 投影在 p 點的影像流速度向量，並且可用

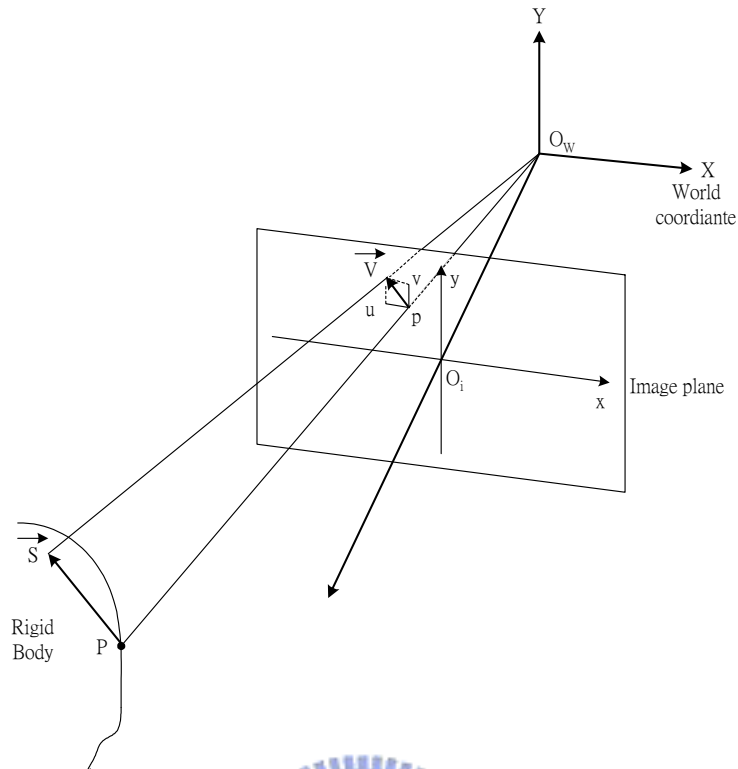


圖 2-1 三維空間之物體投影於影像平面之示意圖

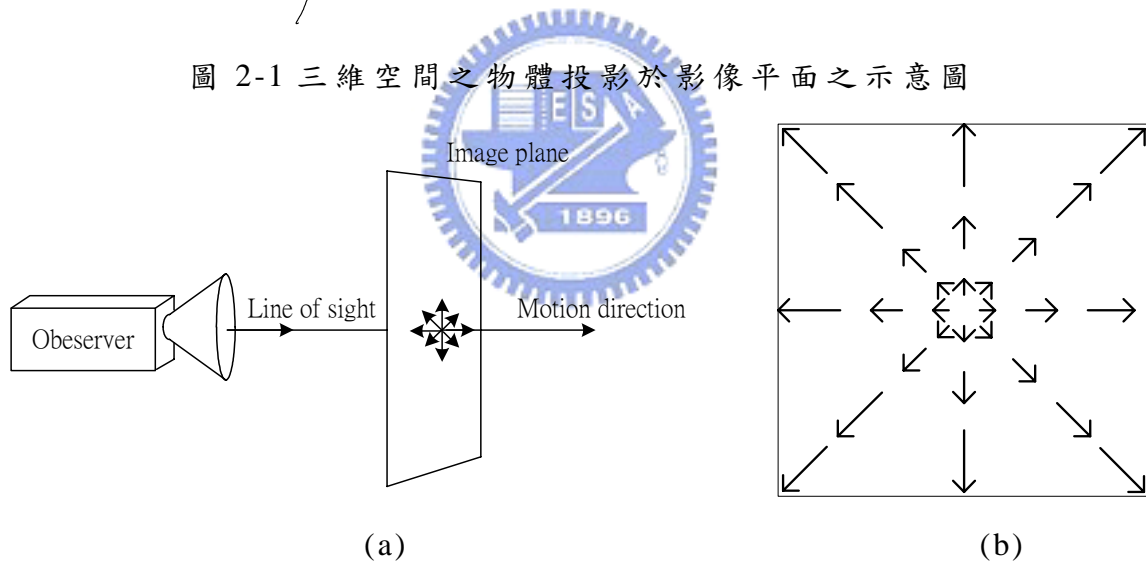


圖 2-2(a)觀察者之注視方向(line of sight)與移動方向相同之示意圖(motion direction) (b)此時觀察者在影像平面(image plane)所看到的影像流。

它來代替描述點 P 在空間中之運動。而像這種影像流速度向量在整個影像平面上的分布情形就構成了所謂的影像流場(Image-flow field)[14]。

特徵流(Feature flow)則是在連續影像中把每張影像的特徵點擷取出來，並在影像間比對其特徵。因此可看作是這些特徵點會在影像間運動，而我們稱這些特徵點在整個影像平面上移動分佈的情形稱為影像流場

(Feature-flow field)。

2.1.2. 擴散焦點(Focus of Expansion)

首先我們先考慮觀察者與背景之間的相對運動，在此定義背景是在空間中靜止不動的物體。當我們移動經過一個在空間中是靜止的物體時，投影在眼睛視網膜上的世界看起來就好像在流動一般。事實上在移動方向(Motion direction)及注視方向(Line of sight)已知的情況下，視網膜或稱影像平面(Image plane)上所看到世界中的物體就好像從視網膜上某特定點流出來一樣，而這個特定點就是所謂的擴張焦點，簡稱 FOE。當我們注視方向與移動方向相同時，如圖 2-2(a)，則會看到如圖 2-2(b)的景像。每個運動與注視方向者會產生唯一的 FOE。假如運動方向平行於視網膜(影像)平面則 FOE 可能位於視網膜(影像)平面範圍以外無窮遠的位置上[15]。

2.1.3. 觀察者與移動中物體的相對運動

一般出現在動態影像中的運動通常由四個基本運動型式所構成，如圖 2-3[14]，這四個基本運動型式分別是：

型式一：與觀察者保持固定距離的移動(Translation)，移動的方向與視軸方向垂直。觀察者看到的影像如圖 2-3(a)，圖中物體是相對觀察者向右移動。

型式二：與觀察者的距離是呈遠近變化的移動，移動方向與視軸方向平行。觀察者看到的影像如圖 2-3(b)，圖中物體正接近觀察者。

型式三：與視軸保持固定距離的轉動(Rotation)。觀察者看到的影像如圖 2-3(c)，圖中物體正向左旋轉。

型式四：與視軸保持固定距離的轉動(Rotation)，但旋轉軸線垂直於視軸。觀察者看到的影像如圖 2-3(d)，圖中物體正逆時針翻轉。

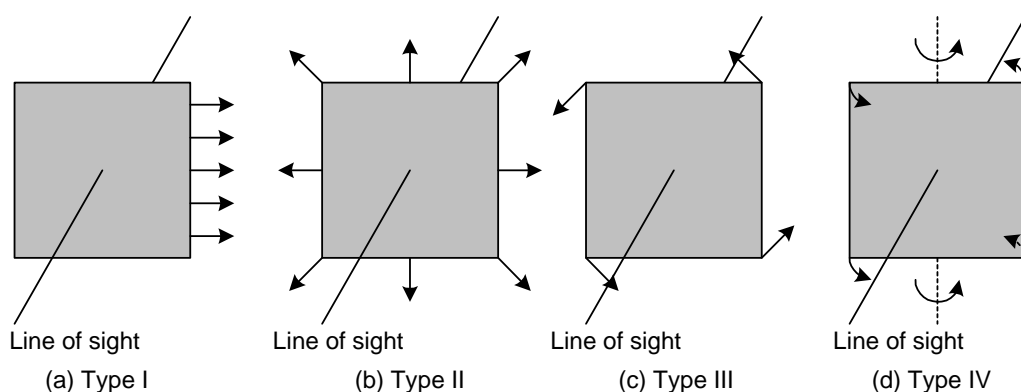


圖 2-3 四種基本運動型式

根據這四個基本運動型式，由於我們所要追蹤的物體是前方行駛中的車輛，幾乎為型式一與二的組合。至於型式三、四則是因為路面的高低起伏而產生的，因為遠小於前兩者，故在此不多做討論。

2.1.4. 道路場景分析

綜合前兩小節，本論文所要研究的道路影像為向前直行於平面道路上的車(簡稱本車)中所見到的前方連續影像，如圖 2-4，影像中可看到行駛於本車前方的車輛(簡稱前車)，其行駛方向與本車相同，如圖 2-5。因為移動方向(Motion direction)與 CMOS 影像感測器之注視方向相同，則影像中必會出現 FOE，而其位置約位於影像中心點，如圖 2-6 所示。就運動型式一而言，前車在影像中會呈現左右的移動，而無上下之移動。其大小會隨運動型式二的移動而改變，愈接近本車，形體愈大;反之愈小。

2.2 系統架構

本論文主要是基於特徵法之追蹤的原理在利用追蹤的物體所表現出一些特殊、可信的特徵做為追蹤的依據，然後在每張影像中找出其相關的位置[2]。這些特徵通常為邊緣、角點或紋理等可以很清楚地在每張影像中被確認出來。Barnard and Thompson [16]提出對好的特徵的三個重要的準則：

1. 特徵必須可明確地定義出來，而且是離散的

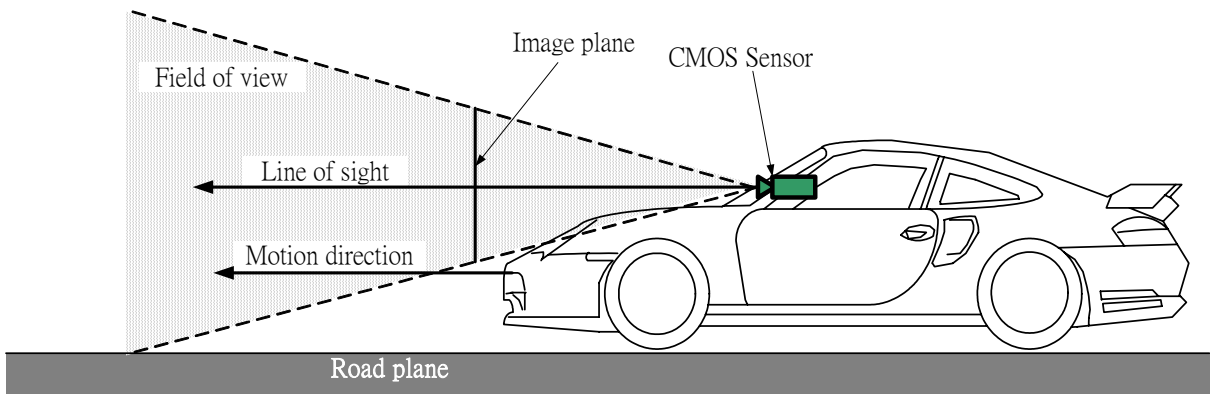


圖 2-4 道路場景分析(側視圖)

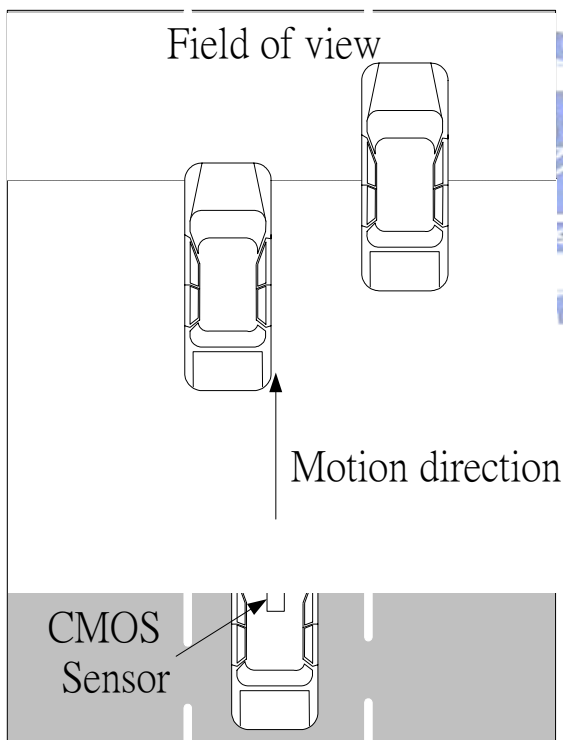


圖 2-5 道路場景分析(上視圖)



圖 2-6 車前所見之影像

2. 特徵必需在時間上是穩定，即相同的特徵點經過一段時間還保有相似的特性。
3. 對於相似的特徵在匹配上必需一致。

從特徵點找出特徵流包括兩個步驟：

1. 將特徵點從兩張或更多張的連續影像中抽取出來。抽取出來的特徵要準確地位於該影像平面上，因為特徵位置的誤差會影響後序整個處理過程的準確度。
2. 這些特徵點必需在各個影像間被追蹤與比對。

在邊緣、角點和紋理三種特徵中，因為角點屬於二維特徵點，經由追蹤比對後，可得到明確的移動資訊，故本論文採用角點作為特徵點。圖 2-7 顯示本論文提出之影像追蹤系統，其設計主要分為六部分：特徵擷取、特徵比對、特徵流分析、FOE 估算、相關性追蹤及碰撞偵測。一開始輸入連續灰階影像經由特徵擷取 (Feature detector) 將每張影像圖像 (Image frame) 所偵測到的特徵點 (即角點) 位置、及其屬性 (灰階值、水平和垂直灰階梯度) 形成一特徵點清單 (Feature list)。再由特徵比對 (Feature matching) 將鄰近兩張圖像所得到的特徵點清單比對，得到特徵點在連續影像中移動的向量，在此稱特徵流 (Feature flow)。用特徵分析器 (Flow analyser) 來分析特徵流，找出屬於背景特徵的特徵流，經過 FOE 估算 (FOE estimation) 定義出偵測範圍 (Detect region) 回傳給特徵分析器。找出在偵測範圍內符合移動中物體的特徵流，將其特徵點位置輸出給相關性追蹤 (Correlation tracking) 在原始影像中擷取出以其特徵點為中心的樣板 (Template)。最後依其樣板在連續影像中持續追蹤。碰撞偵測則是定義出與前車的安全距離，若偵測到移動中物體的特徵點在安全距離內，則發出警告訊號提醒駕駛人。以下則分別解說每一個區塊。

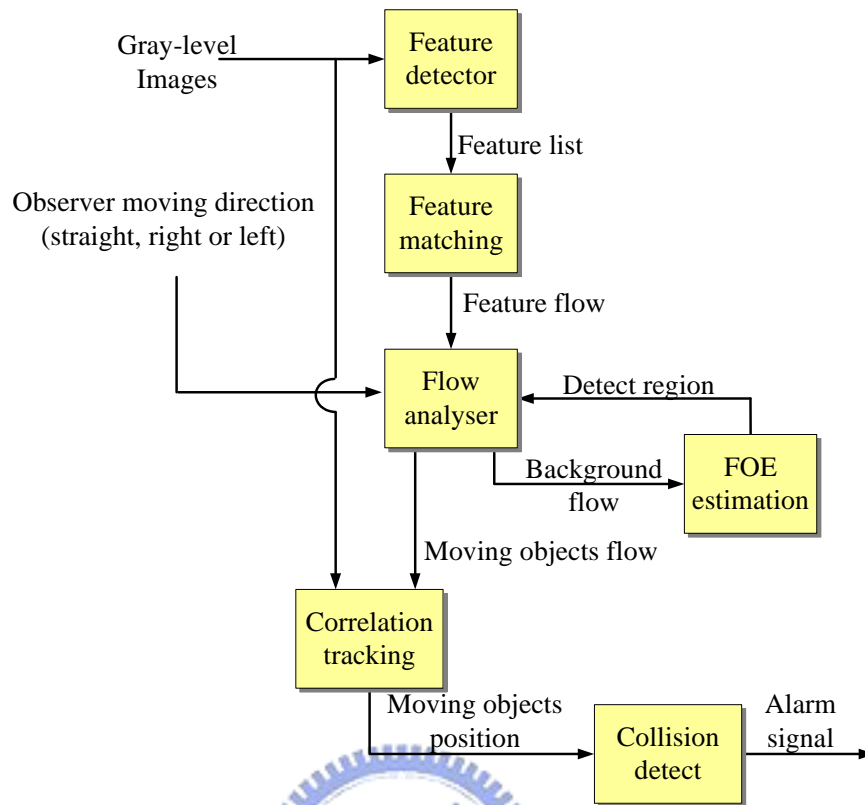


圖 2-7 系統架構圖

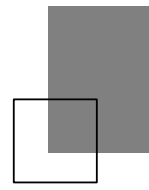
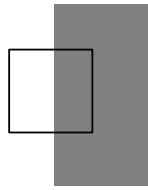
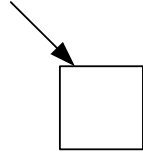
2.3 Harris 角點偵測

本論文所使用的角點偵測法則是由 Harris [17]於 1988 年所提出。其主要的觀念是從 Moravec[18]的角點偵測所延伸出來的。故我們先了解 Moravec 角點偵測的原理。

Moravec 考慮在影像中一局部的區塊(Local window)，經由將此區塊在各方向做小幅的移動，偵測其灰階值平均改變的情況。主要分為以下三種情況：

1. 若此區塊在灰階值平坦的影像區域中移動(即灰階值變化近似一定值)，則各方向的移動所得的灰階值變化均很小，如圖 2-8(a)。

A local window



(a) 位於平坦的影像中 (b) 位於邊緣的影像中 (c) 位於角點的影像中

圖 2-8 局部區塊於影像中的位置。

2. 若此區塊在含有邊緣的影像區域中移動，則區塊沿著邊緣的移動所得的灰階值變化很小。但垂直於邊緣方向的移動所得的灰階值變化則有大幅的變化，如圖 2-8(b)。
3. 若此區塊在含有一角點(Corner)或一獨立的點(Isolated point)的影像區域中移動，則此區塊在各方向上移動所得的灰階值變化均很大，如圖 2-8(c)。

故我們可以將區塊藉由各方向小幅的移動，均得到明顯的灰階值變化來偵測角點。

由以上的分析，我們可以用式(2-1) [17]來表示。

$$E_{x,y} = \sum_{u,v} w_{u,v} |I_{x+u,y+v} - I_{u,v}|^2 \quad (2-1)$$

其中 $w_{u,v}$ 為中心點是 (u,v) 的局部區塊，在此定義為一固定大小的矩形區塊，若影像中的點(Pixel)位於此區塊中， $w_{u,v}=1$ ，否則 $w_{u,v}=0$ 。 $I_{u,v}$ 為影像中點 (u,v) 的灰階值。 x,y 分別為在 x 方向和 y 方向的移動量。 $E_{x,y}$ 則是計算後的灰階值變化量。因此 Moravec 認為只要找出 $E_{x,y}$ 的局部最大值且大於某於閾值 $\min\{E\}$ 即為角點。

然而 Harris 提出三點 Moravec 會遇到的問題

1. 對於數位影像，我們只需考慮每隔 45° 方向的小幅移動，將式(2-1)泰勒展開得式(2-2) [17]

$$E_{x,y} = \sum_{u,v} w_{u,v} [xX + yY + O(x^2, y^2)]^2 \quad (2-2)$$

其中 $O(x^2, y^2)$ 為泰勒展開式的二次以上的多項式

一階導數 X, Y 可由式(2-3)、(2-4) [17] 近似

$$X = I * (-1, 0, 1) = \partial I / \partial x \quad (2-3)$$

$$Y = I * (-1, 0, 1)^T = \partial I / \partial y \quad (2-4)$$

對於 x, y 分量有極小的移動量，式(2-2)忽略較高次項(即忽略 $O(x^2, y^2)$) 得式(2-5) [17]

$$E_{x,y} = Ax^2 + 2Cxy + By^2 \quad (2-5)$$

其中 $A = X^2 * w, B = Y^2 * w, C = (XY) * w$

2. $E_{x,y}$ 對影像的雜訊很敏感，因為 $w_{u,v}$ 定義為以 (u, v) 為中心的局部矩形區塊，若影像中的點(Pixel)位於此區塊中， $w_{u,v} = 1$ ，否則 $w_{u,v} = 0$ 。Harris 改用 Gaussian 函數產生一平滑變化的圓形區塊(Smooth circular window)，如式(2-6) [17]

$$w_{u,v} = e^{-(u^2+v^2)/2\sigma^2} \quad (2-6)$$

3. 式(2-1)只考慮 E 的最小值($\min\{E\}$)，致使式(2-1)對邊緣的反應變化太大。將式(2-1)改寫為式(2-7) [17]

$$E(x, y) = (x, y)M(x, y)^T \quad (2-7)$$

其中 M 為一 2×2 的對稱矩陣(symmetric matrix)，如式(2-8) [17]

$$M = \begin{bmatrix} A & C \\ C & B \end{bmatrix} \quad (2-7)$$

Harris 分析出當 M 的特徵值 α, β 都很大時，即為角點。並提出角點/邊緣響應方程式(Corner/edge response function)，如式(2-9) [17]

$$R = \text{Det}(M) - k \cdot [\text{Tr}(M)]^2 \quad (2-9)$$

其中 $\text{Tr}(M) = \alpha + \beta = A + B$, $\text{Det}(M) = \alpha\beta = AB - C^2$, k 為常數，通常設為 0.04。

找出 R 的局部最大值且大於某一閾值即為角點。

2.4 角點比對

追蹤角點的流程是對連續影像中的每一張圖像擷取出角點，再從該角點的初始圖像與隨後的圖像間做比對。角點比對包含兩個步驟：首先對該角點的移動做預測，稱為特徵點運動預測(Feature motion prediction)，定義出角點的候選名單，再針對這些角點做比對。以下分別對特徵運動預測(Feature motion prediction)與 GVM(Gradient Vector Matcher)[2]角點比對說明。

2.4.1. 特徵點運動預測(Feature motion prediction)

由於影像的雜訊、移動物體的任意移動及攝影機本身的位移，對特徵點運動的預測不可能完全準確地預測其位置。因此我們必須定義出影像平面中一個以預測後位置為中心的區域，在此稱為 Region of Interest(ROI)[19][2]。如圖 2-9 所示，當我們找出時間為 $t-2$ 的圖像(以後簡稱 frame($t-2$))裡的 A 點，對應到時間為 $t-1$ 的圖像(以後簡稱 frame($t-1$))裡的 A' 點。設 A 到 A' 的距離($\|A - A'\|$)= L ，則我們可以預測 A' 在 frame(t)時可能的位置為 A'' 。 A'' 的估測位置為 (x'', y'') ，如式(2-10)、(2-11)。

$$x'' = (x' - x) + x' = 2x' - x \quad (2-10)$$

$$y'' = (y' - y) + y' = 2y' - y \quad (2-11)$$

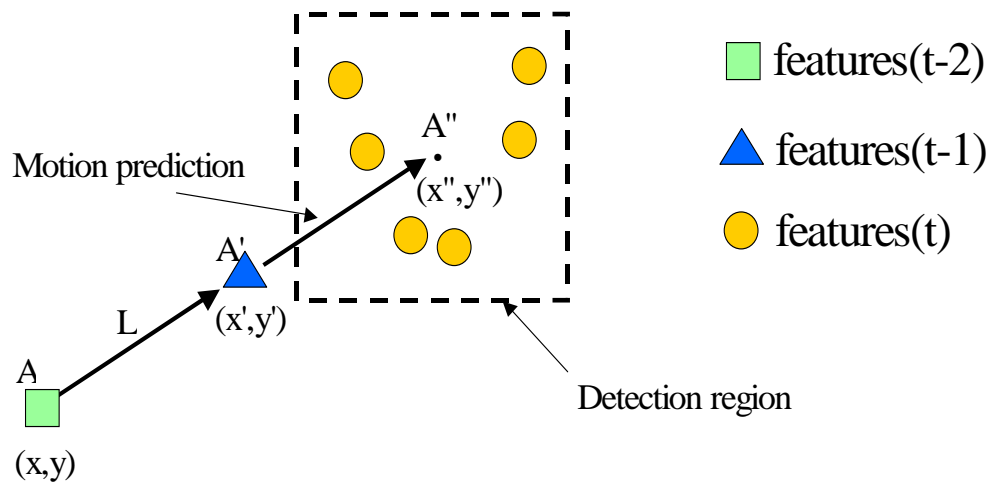


圖 2-9 特徵點運動預測示意圖

本論文設 ROI 的大小為 7×7 pixels。接著找出落於 ROI 中的特徵點(如圖 2-9 所示之圓形特徵點)與 A'' 比對。

2.4.2. 角點比對

本論文主要用 GVM(Gradient Vector Matcher) [2][19]作為角點比對的方法，並做部分的更改。以下對本論文作法做詳細說明。

當我們用 Harris 偵測出角點時，同時將該角點的性質:灰階值(以下簡稱 I)、水平灰階梯度(以下簡稱 dx)及垂直灰階梯度(以下簡稱 dy)作為比對的項目。由於一般的 CMOS 影像感測器會有自動調整影像灰階度的功能(即自動白平衡,Auto white balance)，以保持影像不會過度曝光或太暗。故我們不能直接用 I, dx, dy 來比對角點。因此我們將設一個三維向量由 I, dx, dy 三個分量所組成，假設原角點之向量為 v ，候選角點之向量為 w ，如圖 2-10 所示。

將 v, w 用式(2-12)比對，其中 $m(v, w)$ 為比對可信值(Match confidence value)。若 m 值愈小，則兩角點愈相似，在此設一閾值 $mth=0.0007$ 。原 GVM 的作法是將該角點周圍 3×3 的 $(dx)^2$ 相加再開根號取代 dx ，同理，該角點周圍 3×3 的 $(dy)^2$ 相加再開根號取代 dy 。但這樣會把向量 v, w 原有的四個象限縮為一個象限，減少比對的可信度。然而此法仍有比對錯誤的時候，但經

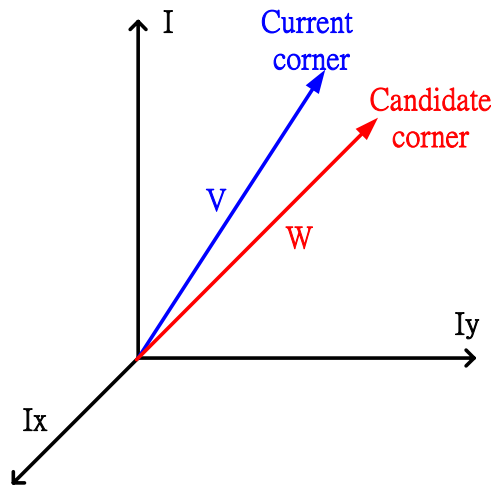


圖 2-10 GVM 角點比對示意圖

過多個連續圖像的比對後，自然而然這些比對錯誤的角點會被濾除，這也是使用多張圖像處理的好處之一。

$$m(v, w) = \frac{|v - w|}{\sqrt{|v||w|}}$$

(2-12)

2.5 特徵流分析

若我們在連續的影像中成功地比對角點，則可得到特徵點在連續影像中的移動資訊，即為特徵流。而特徵流的分析則是要與現實的場景結合，才能分割出移動中的物體。首先我們先定義角點被成功地比對為此角點在連續的十張圖像中都能夠被比對。

一般來說，靜態背景的特徵流必是由 FOE 輻射狀向外擴散，離 FOE 愈遠，擴散速度愈快。然而，靜態背景的特徵流移動的速度並不只與 FOE 的距離有關，還跟本車(即觀察者)與該背景的距離有關。離本車愈遠的靜態背景的特徵點擴散速度愈慢(在此稱遠方背景特徵點)。

由 2.1 場景分析可知我們所要追蹤的物體為行駛於道路上的前方車輛(簡稱前車)。設前車之車速為 V_c ，本車之車速為 V_o ，以運動型式二來分，前車在影像平面中有三種表現：

1. $V_c > V_o$ ，前車將超前本車，即超車。其特徵流必朝向 FOE。
2. $V_c \cong V_o$ ，前車與本車約保持固定距離。其特徵流移動量甚小。
3. $V_c < V_o$ ，前車逼近本車。前車之特徵流會遠離 FOE。

第 1 項因為與背景特徵流之方向完全相反，故可輕易地與背景特徵流區分。第 2 項，看似也可由移動特性來區分。但遠方之背景特徵流也是移動緩慢，故並不能用移動特性區別。第 3 項，理論上，雖然也是同背景特徵流一樣遠離 FOE 運動，由於前車屬於高速行駛狀態，故向外擴散的速度遠小於背景特徵流。但它的問題同第 2 項，相似於遠方背景的特徵流。

本論文提出將針對第 2、3 項之前車與遠方之背景的特徵流之區分提出解決方法。由於我們所要追蹤之前車為行駛於道路上之車輛。故由影像平面來看，前車之特徵點必與道路區域有所重疊，以式(2-13)表示，如圖 2-11 所示。遠方之背景特徵並沒有與道路區域重疊，故可將兩者分開。

$$(moving_objects) \cap (road_plane) > 0 \quad (2-13)$$

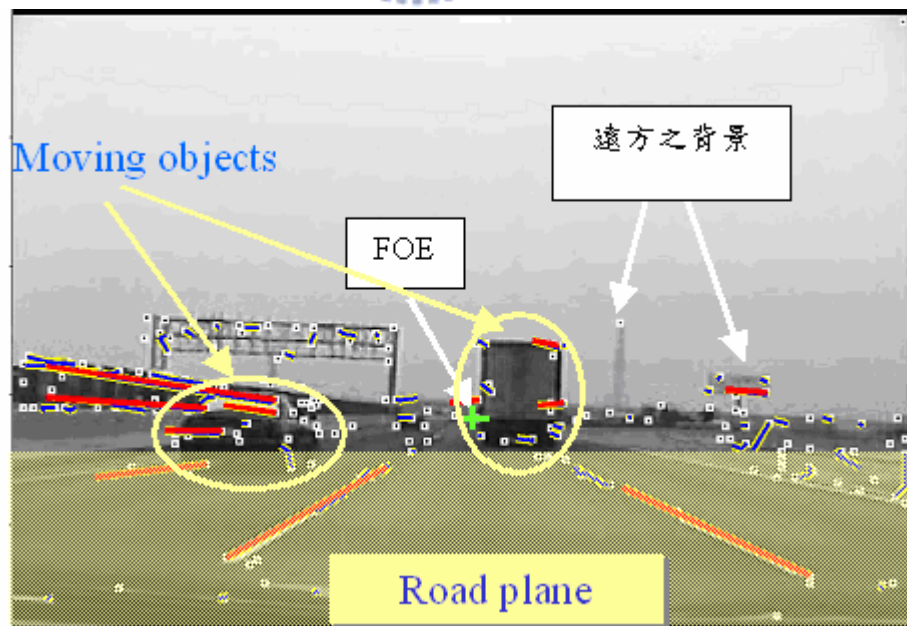


圖 2-11 移動中物體與道路平面重疊示意圖

問題是我們如何找出在影像中道路的平面區域？經由觀察，不難發現道路平面必在 FOE 之下方。所以我們可以用 FOE 的位置估測道路平面的區域，由圖 1-1(a)知接近 FOE 的區域，其解析度低，並不適合特徵流的分析，故我們將設道路平面設為 FOE 若干距離以下的區域。2.6 將介紹 FOE 之估測。

2.6 FOE 估測

本論文提出一個藉由找出道路特徵流來估測 FOE 之法則，以下將對其詳細說明。FOE 產生是由於觀察者本身的移動，造成靜態物體的特徵流會從該點向外擴散。由此可知，我們必須找出屬於背景的特徵流來估測 FOE。在影像中諸多的特徵流，我們發現道路的特徵流其垂直分量向下，消失於影像下緣部分，如圖 2-12。根據此特性，我們可以找出兩條斜率分別在 I、IV 象限的道路特徵流，如圖 2-13。延伸此兩條特徵流，其交點即為 FOE，如圖 2-14。

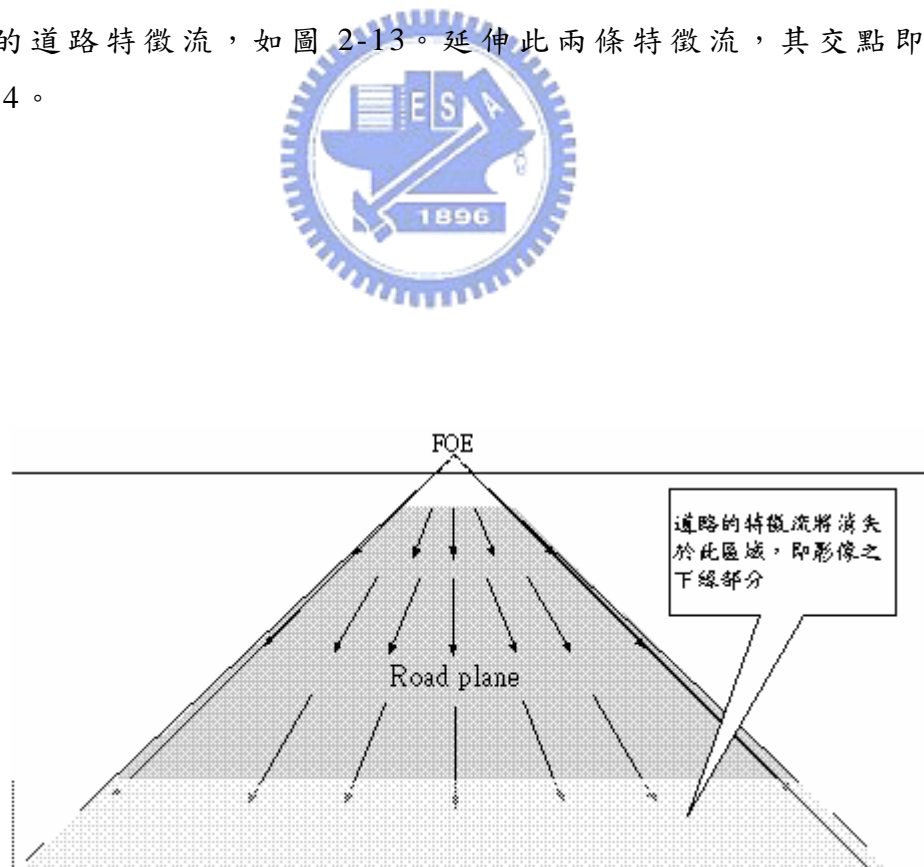


圖 2-12 道路之特徵流，垂直分量必向下

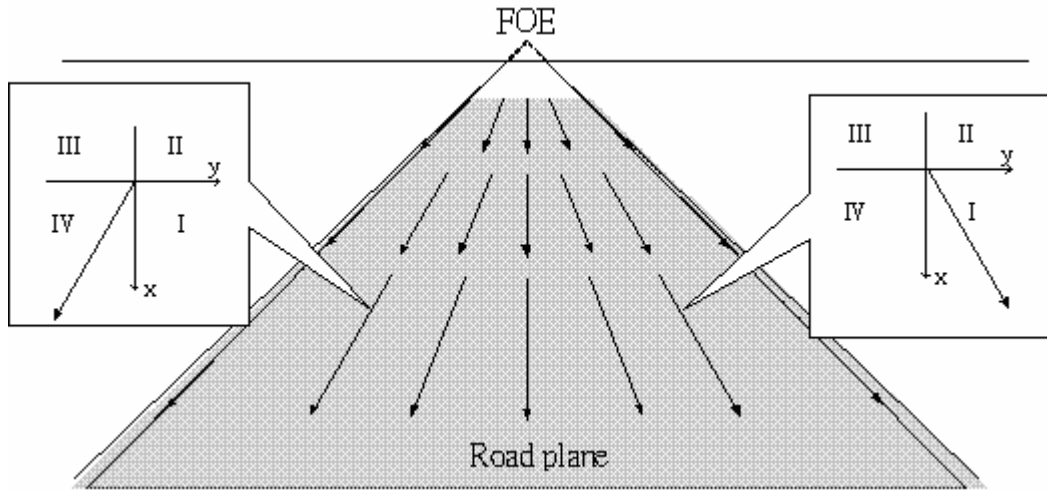


圖 2-13 道路之特徵流，取 I,IV 各一個求 FOE

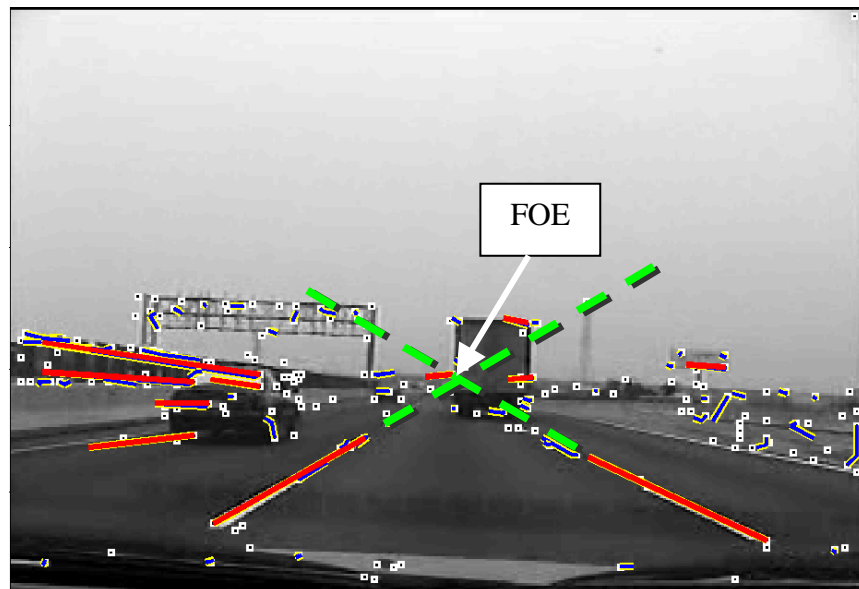


圖 2-14 兩道路特徵流之交點即為 FOE 的位置

2.7 相關性追蹤

經由特徵流分析與 FOE 估測，我們得到移動物體的特徵點。但因為影像受到外界光變化的影響或訊號本身的雜訊，角點比對無法在連續影像持續很久。所以我們用相關係數(Correlation coefficient)[20][1]的方法將特徵點持續追蹤。

如圖 2-15 所示，設點 A 為在圖像(t)偵測出移動物體的特徵點，位置為 (x_0, y_0) ，I 是以 A 為中心，大小為 $m \times n$ 之模板影像(Template image)。M 則為圖像(t+1)中，取一大小為 $m \times n$ 的映射矩陣(Mapping matrix)，即我們要比對是否與 I 模板影像相似。

用式(2-14)求出相關係數(Correlation coefficient) ρ ，在此設一閾值為 0.7[1]。 ρ 愈大表 M 與 I 愈相似。其中 I_{ij} 為模板影像 I 中第(i,j)個像素(Pixel)之灰階值， M_{ij} 為映射矩陣 M 中第(i,j)個像素(Pixel)之灰階值。若 M(x,y) 有 ρ 之最大值且大於 0.7，該 M 之中心點位置為 $A'(x,y)$ ，則將 A 的位置更新為(x,y)，完成整個比對的過程。

$$\rho = \frac{mn \sum_{i=1}^m \sum_{j=1}^n (I_{ij} \cdot M_{ij}) - \sum_{i=1}^m \sum_{j=1}^n I_{ij} \sum_{i=1}^m \sum_{j=1}^n M_{ij}}{\sqrt{\left[mn \sum_{i=1}^m \sum_{j=1}^n (I_{ij})^2 - \left(\sum_{i=1}^m \sum_{j=1}^n I_{ij} \right)^2 \right] \cdot \left[mn \sum_{i=1}^m \sum_{j=1}^n (M_{ij})^2 - \left(\sum_{i=1}^m \sum_{j=1}^n M_{ij} \right)^2 \right]}} \quad (2-14)$$

本論文所使用之模板影像 I 的大小為 7×7 像素。映射矩陣 M 則為以該特徵點為中心之 11×11 像素之間任一點作為 M 之映射矩陣之中心點，如圖 2-15。

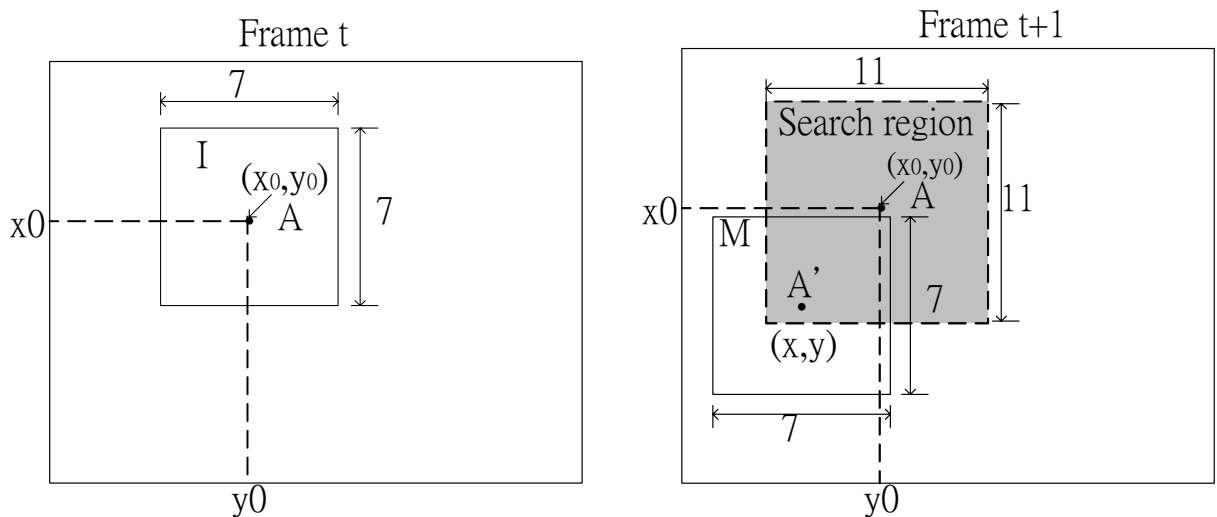


圖 2-15 相關係數之求法

2.8 碰撞偵測

當我們持續地利用相關係數法來追蹤前車時，我們可以在影像中定義出一固定大小及位置之碰撞偵測區域(Collision detect region)，如圖 2-16 所示，其位置為 $179 < x < 200$, $150 < y < 230$ 。一旦被追蹤的特徵點進入此碰撞偵測區域時，系統則發出警告訊號，以提醒駕駛者。

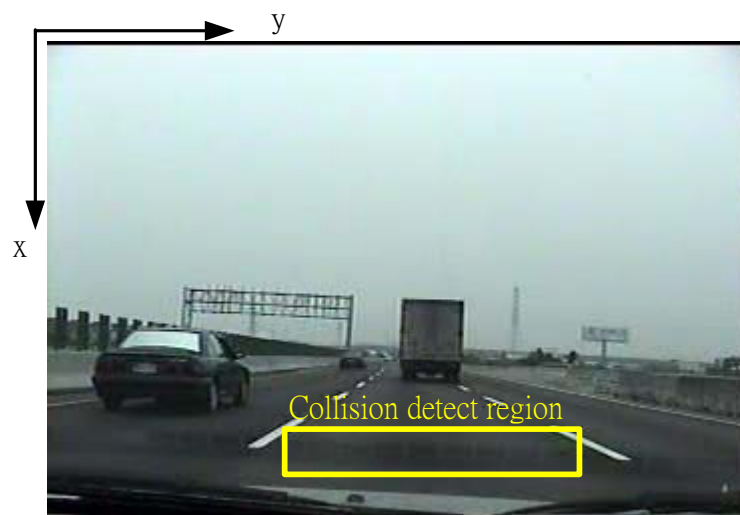


圖 2-16 碰撞偵測範圍

第三章 獨立式影像追蹤系統設計

本論文之系統架構設計為發展一獨立式系統(Stand alone system)。即此系統可獨立運作，即時擷取影像並追蹤前方車輛。若偵測到與前車之距離過近，則即時發出提醒資訊，以供駕駛者參考。此獨立式系統的優點在於可輕易地放置在車中與車輛結合，若技術成熟則可進一步發展成 SoC(System on chip)晶片化而達到價格低廉，大量生產且易安裝使用的好處。然而，如何設計一精簡的獨立式系統又可即時地處理影像資訊則是一項挑戰[6]。因此本論文依系統內每個模組的功能將系統分成軟體與硬體來設計。其中硬體設計有 CMOS 影像感測器之介面電路、Harris 即時角點偵測電路及 Nios 內嵌式處理器。軟體設計則為 Nios 處理器內部中斷向量、記憶體存取、角點比對、影像流分析、相關性追蹤。

3.1 影像追蹤系統之硬體架構

本系統主要分為三個部分。第一部分是 CMOS 影像感測器之介面電路。第二部為 Harris 角點偵測之硬體電路。第三部分則是 Nios 內嵌式處理器。本系統架構圖如圖 3-1 所示。CMOS 影像感測器採用 IC MEDIA 公司所提供的 ICM205B CMOS image sensor module; Harris 角點偵測電路則是實現於茂綸公司所生產的 Cyclone School Boy 研發電路板; 主體 Nios 內嵌式處理器及周邊電路實現於 Altera 公司所製造的 Nios development kit, Stratix edition。我們可以使用 PC 透過 UART 下達命令給 Nios，並可將影像藉由乙太網路傳至 PC，或發出碰撞警示訊號使 LED 燈發亮。

一開始 I²C 控制器先設定 CMOS 感測器之初始狀態，本系統設影像大小為 320×240 之 16 位元 4:2:2 YCbCr，取其 8 位元之 Y 當灰階影像。將灰階影像傳給 FIFO 及 Harris 角點偵測電路，Harris 角點偵測電路則輸出角點位置，而 Sobel 輸出水平、垂直灰階梯度。灰階影像存入 Image FIFO，水平、垂直灰階梯度存入 dx,dy FIFO，角點位置存入 Corner FIFO。原本 Harris 角點偵測電路可輸出水平、垂直灰階梯度，因為考量外部接線之雜訊干擾，故將 Harris 內部之 Sobel operator 同樣在 Stratix 晶片內做一次產生水平、垂直灰階梯度。Nios 處理器則將 FIFO 內的資料寫入 SDRAM 以供 Nios 做後級的影像處理。接下來說明各部分之設計流程。

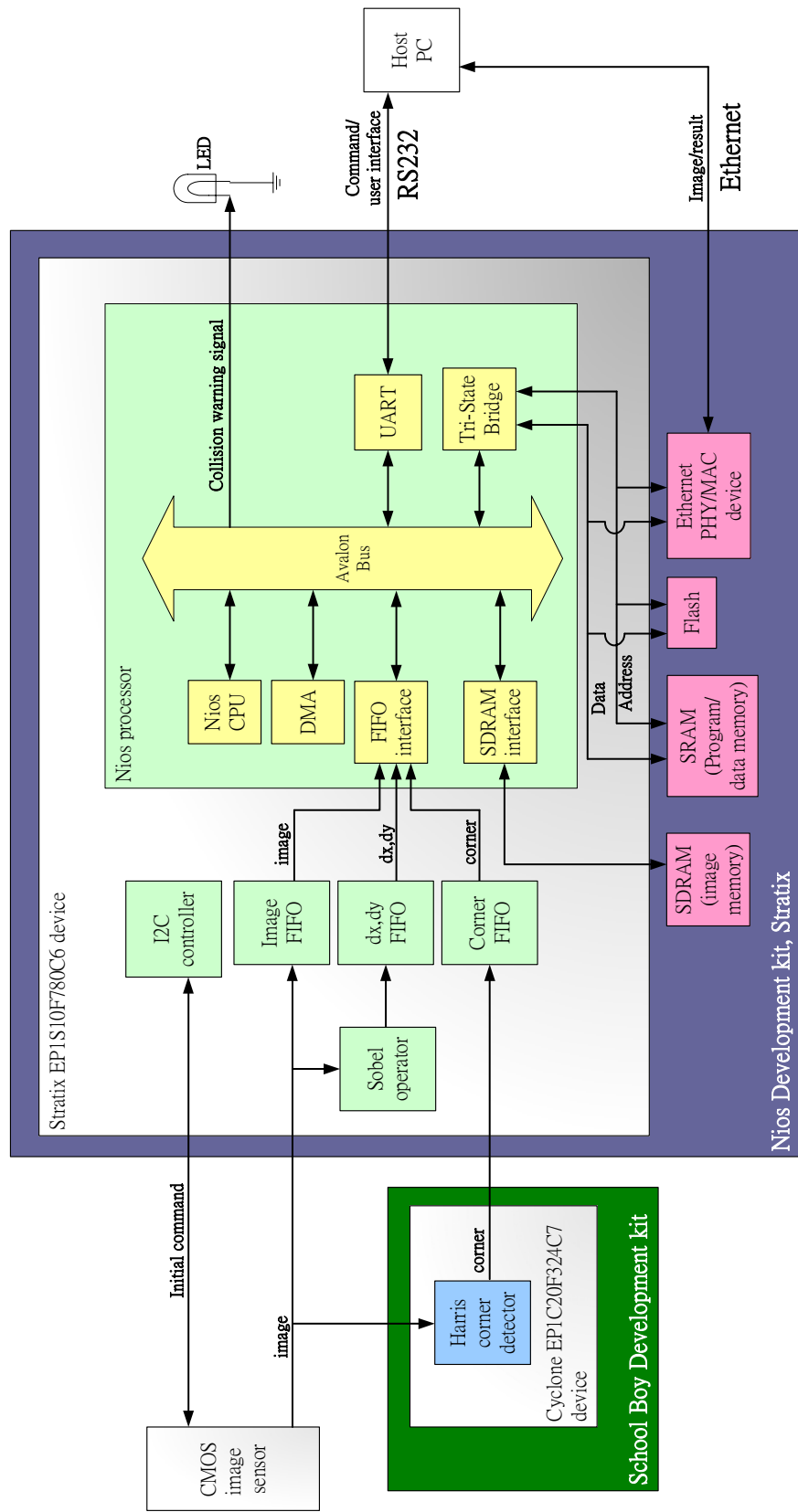


圖 3-1 影像追蹤系統之硬體架構圖

3.2 CMOS 影像感測器介紹

本論文所使用的 CMOS 影像感測器為 IC MEDIA 公司所出產的 ICM205B VGA/QVGA CMOS 影像感測器，如圖 3-2。以下將介紹 ICM205B 之特性。

ICM205B 為一數位彩色影像單晶片[21]。內部有 640×480 之影像感測陣列，以漸進式方式(Progressive manner)輸出，每秒最高可輸出 30 張圖像(Frame)。每個像素(Pixel)會經由一個彩色濾波器形成所謂的 Bayer pattern。在此稱這種資料格式為 Raw data。

隨著外在景色亮度的變化，Raw data 可經由數位增益(Digital gain)調整所有的像素(Pixels)或分別調整四個 Bayer pattern 的像素。我們可用內建的自動曝光控制電路(Built-in automatic exposure control circuit)來控制或是外部 CPU 經由串列匯流排(Serial bus)來控制。

接著資料會經由彩色插補模組使得每個像素可得到完整的 RGB 數值。另外，可用內建的自動白平衡電路(Built-in white balance circuit)或是外部 CPU 經由串列匯流排(Serial bus)來校正或強化色彩。在最後輸出級的輸出格式包括 8/16 位元 YCbCr, 24 位元 RGB 或 8 位元 Raw data。以下列出 ICM205B 之特性[21]:

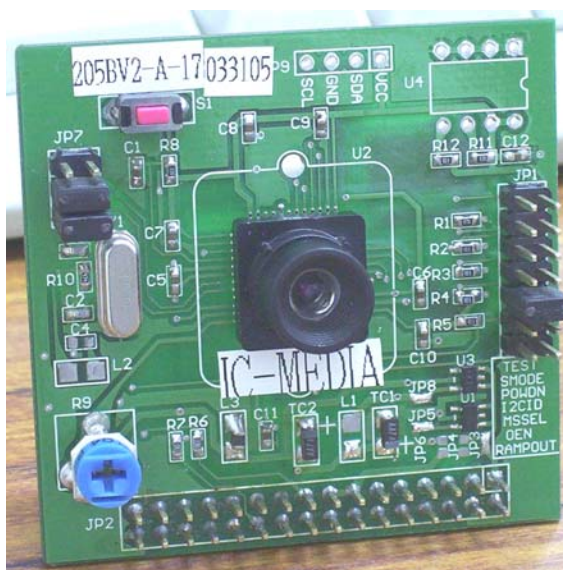


圖 3-2 ICM205B 之外觀

3.3 Nios 發展介面設計

本論文主要使用 Altera 公司針對 (System On a Programmable Chip)SOPC 設計所提供的 Nios 發展板(Stratix 版本)，以 Nios 嵌入式處理器為主體。SOPC 為一種軟硬體整合設計技術，讓我們可以很快速地將硬體系統(包括微處理器、記憶體、週邊介面電路及使用邏輯電路)以及軟體設計，都放至單一個可規劃晶片中，以達到系統的積體設計。Nios 則是 Altera 對 SOPC 設計提供的解決方案之一。Nios 為一軟核心(Soft Core)嵌入式處理器，可有彈性地整合記憶體、週邊介面電路、使用者邏輯電路及其它矽智產(Intellectual property, IP)。這種可組成(Configurable)，一般目的(General-purpose) 精簡指令集計算機 RISC 處理器可輕易地結合使用者邏輯電路，並且規劃至 PLD 晶片。Nios CPU 可以被規劃成 16 或 32-位元的處理器。由於考量到在影像處理的資料量龐大，故本設計規劃運算邏輯單元(Arithmetic and Logic Unit, ALU)、匯流排及暫存器的位元數均為 32 位元。

本論文所使用的發展板 Nios development kit, Stratix edition 之外觀如圖 3-3。



Stratix EP1S10F780C6ES 為可規劃之 FPGA，可將我們的硬體設計檔燒錄於其中。若將 Nios 燒錄於其中，則可快速地與發展板之周邊電路溝通。

3.4 影像擷取介面設計

本論文所使用的輸出模式為 QVGA，即大小為 320×240 像素之影像。輸出格式為 16 位元 4:2:2 YCbCr。因本論文只處理灰階影像，故只取 Y 值，大小為 8 位元，如圖 3-4。影像感測器輸出腳位如圖 3-5，影像感測器介面時序圖如圖 3-6~3-8 所示，其中圖 3-6 為訊號 vsync、hsync 與 blank 的對應關係，圖 3-7 為訊號 hsync、blank 與 pclk 的對應關係。圖 3-8 主要為訊號 sen_clk、pclk 與 sen_data 的對應關係。

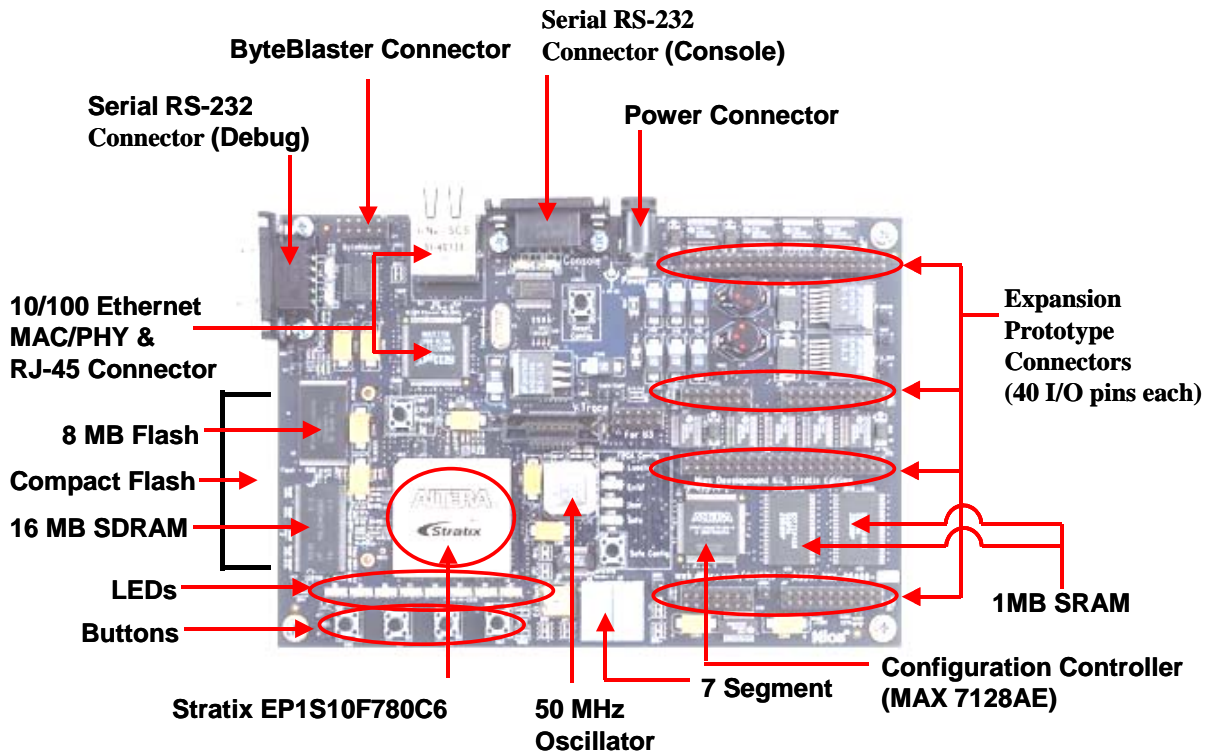


圖 3-3 Nios development kit, Stratix edition

Data out [7:0]														
Y1	Y2	Y3	Y4	Y5	Y6	Y7	Y8	Y317	Y318	Y319	Y320
Data out [15:8]														
Cb1	Cr2	Cb3	Cr4	Cb5	Cr6	Cb7	Cr8	Cb317	Cr318	Cb319	Cr320

圖 3-4 輸出影像格式[21]

腳位說明：

- VSYNC: 垂直同步訊號。其週期為輸出一張圖像的時間，換言之，輸出頻率即為每秒輸出之圖像數(frame/second)，本論文設定每秒三十張圖像。
- HSYNC: 水平同步訊號，其週期為圖像中輸出一水平掃描線之時間。
- BLANK: 當 BLANK=1 時，輸出之資料才有意義。因此 BLANK 可

視為輸出致能訊號。

- DOUT0~DOUT15: 輸出之資料線，因為只取 Y，故只取 DOUT0~DOUT7。
- PCLK: Pixel clock，為輸出同步訊號。當 PCLK 正緣時，輸出一個像素的資料，如圖 3-8。因為輸出為 16 位元 4:2:2YCbCr,QVGA 格式，故 PCLK 之頻率為 3MHz。

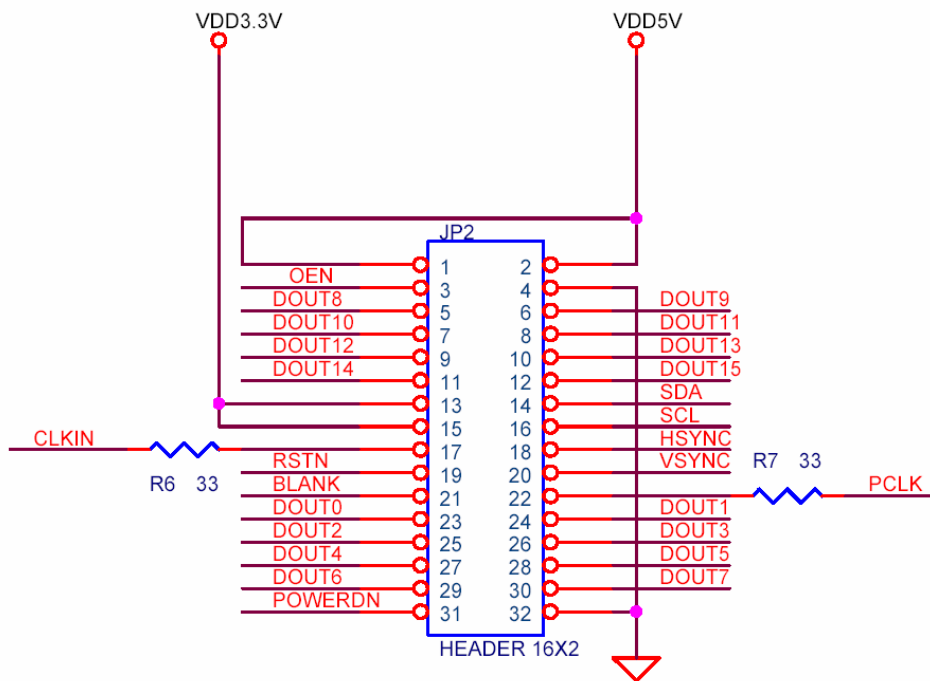


圖 3-5 ICM205B 輸出腳位圖 [21]

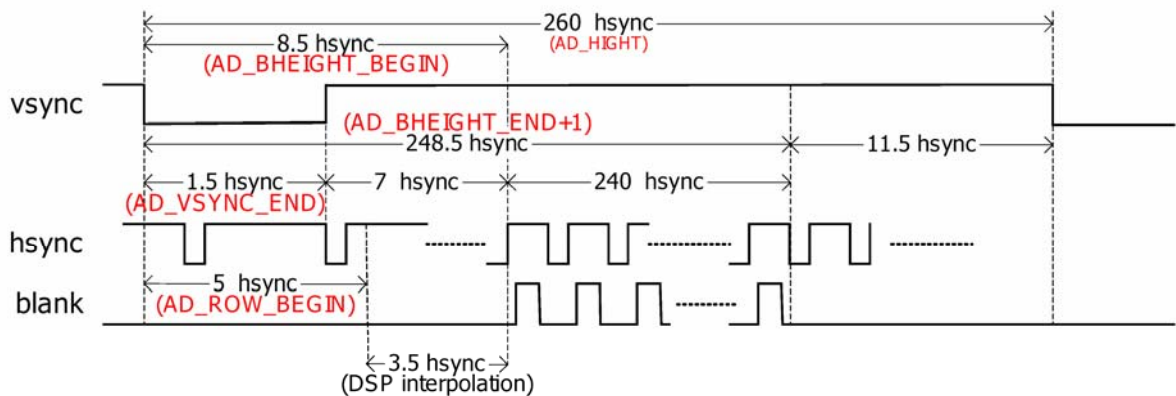


圖 3-6 ICM205B 之時序圖(1)[21]

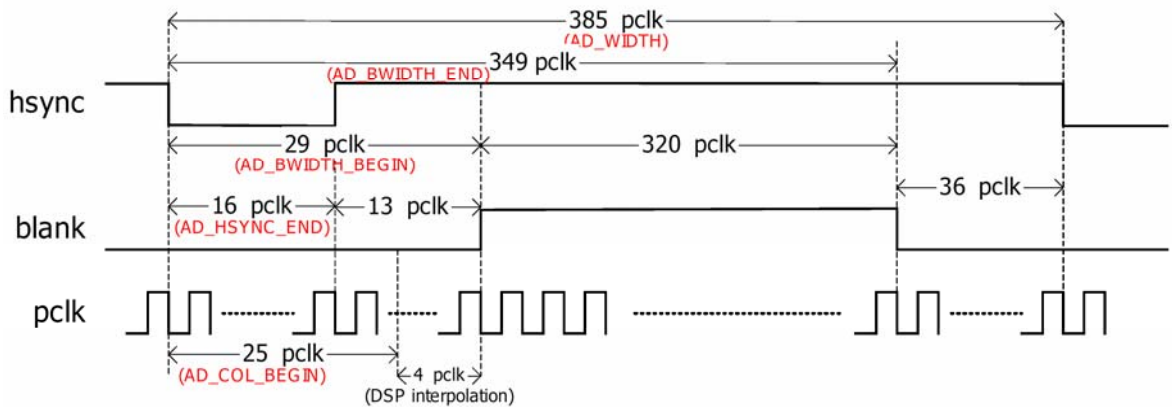


圖 3-7 ICM205B 之時序圖(2)[21]

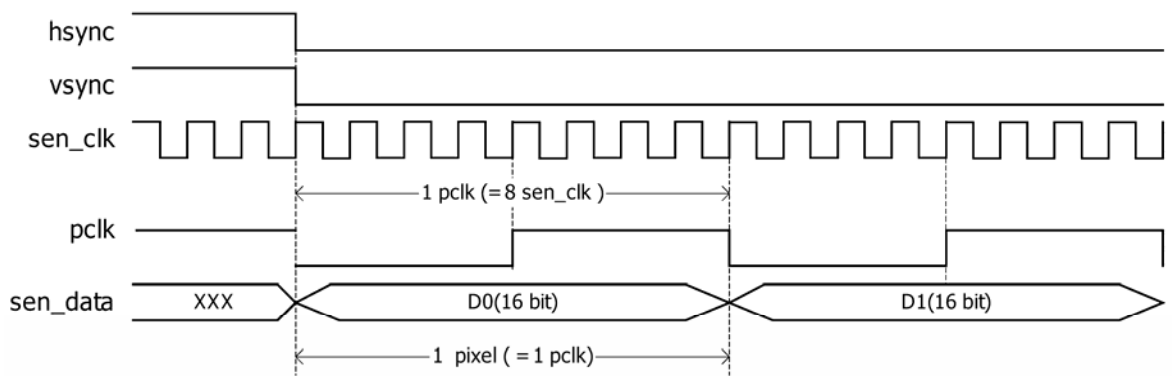


圖 3-8 ICM205B 之時序圖(3)[21]

在了解 CMOS 影像感測器之輸出訊號後，因為 CMOS 寫入之時脈頻率 (3MHz)遠小於 Nios CPU 之存取時脈(50MHz)，故我們利用 Quartus II 之 Mega wizard plug-in manager 設計一雙時脈 FIFO(Dual Clock FIFO)，大小為 8192 個位元組，如圖 3-9 所示作為暫存影像資料的記憶體，可同時讀寫資料以彌補兩讀寫時脈的差距。同時設置一中斷訊號產生電路，如圖 3-10，作為發出一中斷訊號通知 Nios CPU 去讀取 FIFO 的資料。FIFO 腳位說明如下[23]:

■ 輸入腳位

- data[7..0]:輸入的資料，接 DOUT[7..0]。
- wrreq: write request，寫入致能訊號。與 BLANK 連接。
- wrclk: write clock，寫入時脈，接 PCLK。
- rdreq: read request，讀取致能訊號，接 Nios 之系統讀取致能

訊號 read_enable。

- rdclk: read clock，讀取時脈，接 Nios 之系統時脈 sysclk。
- aclr: all clear，清除 FIFO 內部所有資料。每張圖像要開始寫入 FIFO 前，希望把 FIFO 內部資料清除，以確保資料的同步。故使用/VSYNC 將 FIFO 清除。

■ 輸出腳位

- usedw[12..0]: 記錄 FIFO 目前有多少筆資料已寫入而未讀出。可以當作 Nios CPU 偵測 FIFO 是否快滿的訊號。
- q[7..0]: 輸出之資料。

因為 QVGA 之單張影像資料量為 76800Bytes，然而 mega wizard 所能提供 FIFO 的最大深度為 8192Bytes，故我們將原始影像分成十次填入所設計之 my_FIFO，如圖 3-11，因我們設定每秒輸出 30 張圖像，故一張圖像輸出時間為 33.3 ms。而在一個圖像輸出的時間內，即 VSYNC 一週期的時間內，ff_irq 發出 10 次中斷，每次的中斷 Nios 都會命令 DMA 去抓取 FIFO 內 7680 筆資料。因為讀取時脈 50MHz 遠大於寫入時脈 3MHz，待資料存滿 7680 個位元組，仍有 512Bytes 當緩衝空間，故可順利同時寫入和讀取資料。而中斷的週期為 2.9232ms，DMA 一次抓取 7680 筆資料的時間為 156.4 μ s。VSYNC 為 0 時，會把 FIFO 內資料清除，故 DMA 必須在 VSYNC 為 0 之前把最後一次的 7680 筆資料取完，由圖 3-11 右下圖可知，DMA 在 1.316ms 前將資料讀取完成，故本設計可成功地連續讀取每秒 30 張 320 \times 240 之 8 位元灰階圖像。

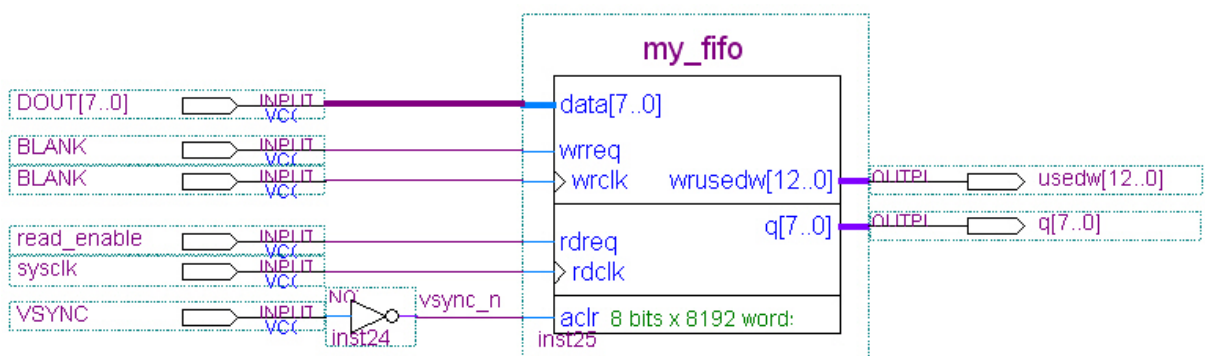


圖 3-9 Dual clock FIFO

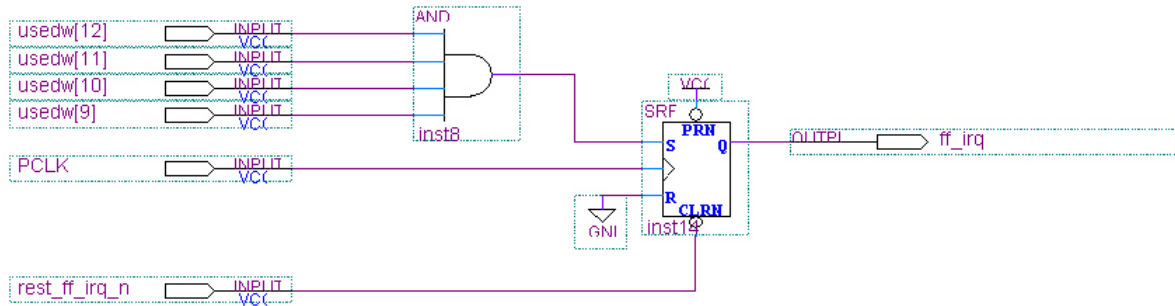
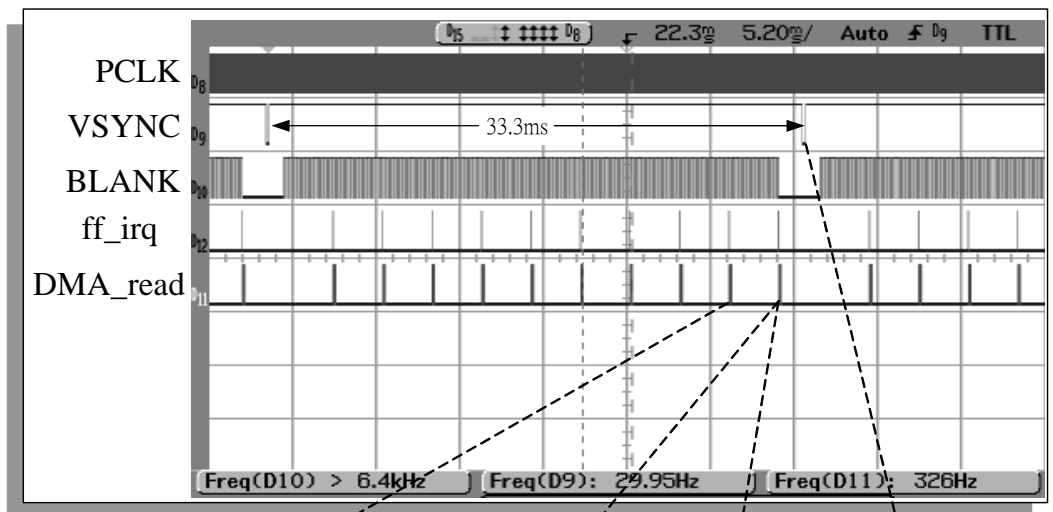


圖 3-10 中斷訊號 ff_irq 電路圖

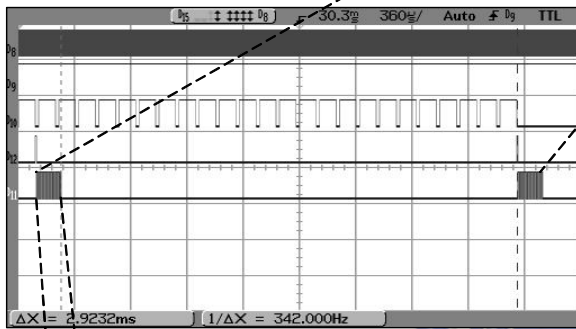
3.5 Nios 之規劃與設計

由於考量到影像處理的資料量龐大，故本設計規劃運算邏輯單元 (Arithmetic and Logic Unit, ALU)、匯流排及暫存器的位元數均為 32 位元。規劃 Nios 之介面是由 Quartus II 發展軟體外掛之 SOPC Builder 所建構的，其規劃介面如圖 3-12。

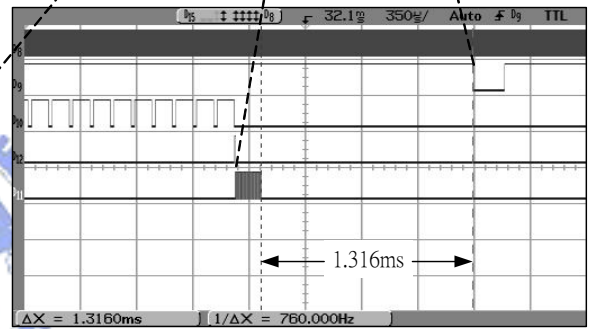
DMA 主要功能是可以於記憶體或週邊電路之間快速地存取資料，而不需花 CPU 的時間。好處是 CPU 可以在這個時候，執行其它的工作。如圖 3-13 所示，當 DMA 在存取 Data memory 時，CPU 可同時存取 Program memory 或其它 I/O 資料。故像從 FIFO 將大量的影像資料搬移至 SDRAM 的工作就適合 DMA 來執行，CPU 只須設定 DMA 內部參數即可 [24]。如圖 3-14，因為 DMA 一次要存取 7680×4 Bytes 的資料，另兩個 FIFO 各存 16 位元的水平與垂直灰階梯度及 8 位元的角點資訊，故設定 DMA 的長度記憶體的寬度 (Width of the DMA length register) 為 15 位元，如圖 3-14



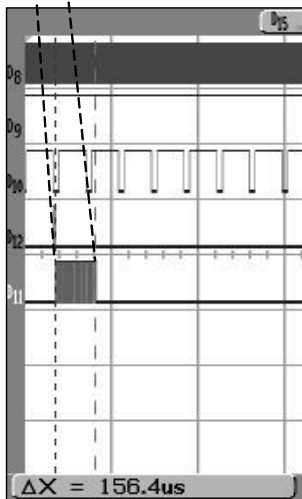
(1)



(2)



(3)



(4)

- (1) VSYNC週期為33.3ms。
- (2)DMA讀取週期為2.9ms。
- (3)DMA在下一個圖像輸出前的1.316ms讀完影像。
- (4)DMA讀取7680個位元組資料之時間為156.4 μ s

圖 3-11 CMOS 取像時序圖

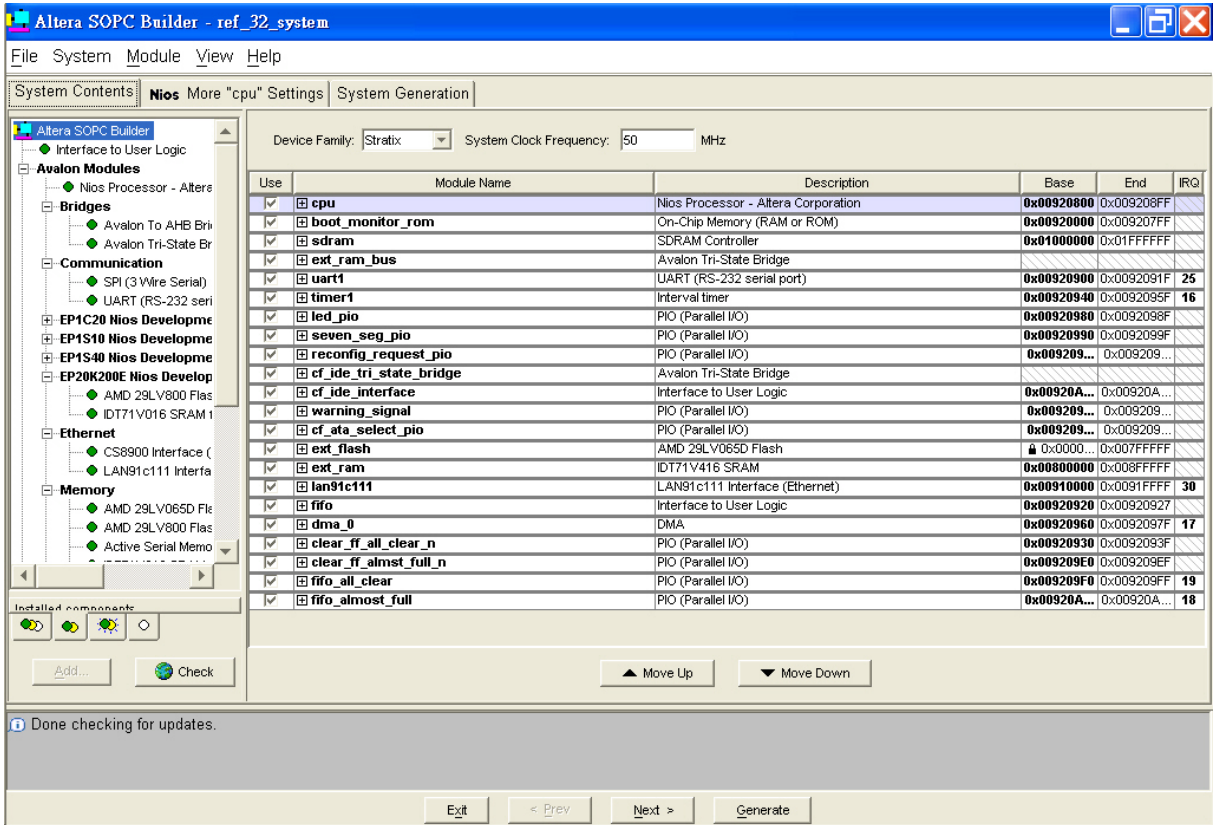


圖 3-12 SOPC 之操作介面

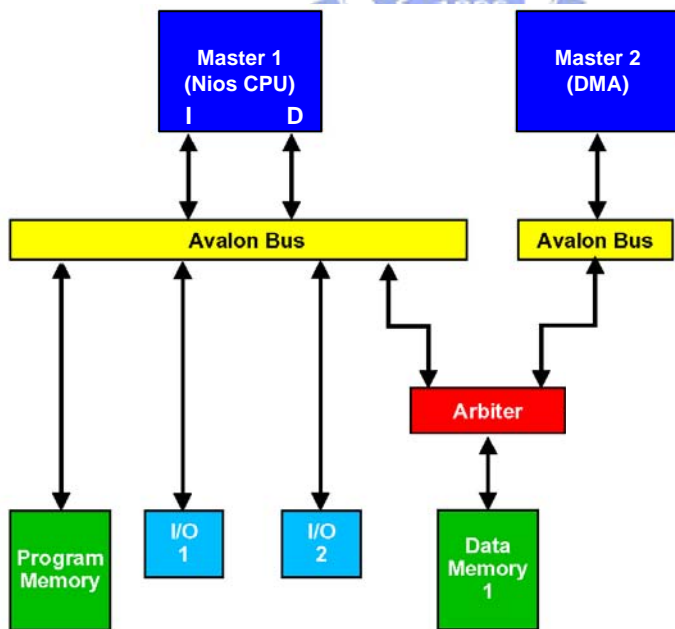


圖 3-13 DMA 運作示意圖

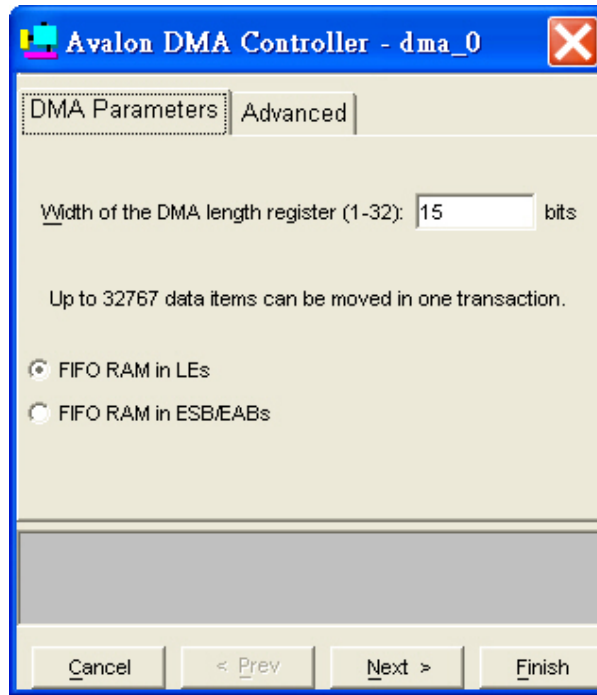


圖 3-14DMA 內部設定

本論文主要用到的硬體如表 2 所示

表 2 Nios 系統所規畫的元件

元件名稱	說明
CPU	規劃成 32 位元
SDRAM	16MB，將外部影像、角點、梯度資料暫存於此
SRAM	1MB, data/program memory
UART	與 pc 端之溝通介面
Ethernet	負責乙太網路之資料傳輸，可將影像傳送至 PC 端
FIFO 控制介面	存取外部 FIFO 資料的介面，使用 interface to user logic 構成
DMA	負責存取外部 FIFO 的資料，減輕 CPU 的工作量。
PIO(Parallel input/output)	規劃一些外部控制訊號，或規劃成輸入訊號，成為中斷訊號。

3.6 Harris 角點偵測硬體設計

Harris 角點偵測電路是實現於茂綸公司所生產的 School Boy 研發電路板，燒錄於 Cyclone EP1C20F324C7 Device，其外觀如圖 3-15，採用 Altera Cyclone 系列 Fine Line BGA 324 pin 包裝設計，分為母板與子板。在母板部分，除了電源部分已處理完成，所有的 I/O 接腳也接至連接器以方便使用者接線或測試。而在子板部分，主要是 Cyclone EP1C20F324C7 Device 與其所需的 In-system programmability, ISP 電路所構成，提供 20,060 Logic Elements 與 294,912 bits RAM。

Harris 演算法之輸出/入腳如圖 3-16 所示，輸入訊號 Clk_in 接 PCLK，data_in[7..0] 接 DOUT[7..0]，W_en 接 BLANK，Reset 接 VSYNC 之反向訊號。則輸出之訊號則是如同 CMOS 輸出之時序。Clk_out 為輸出同步訊號，Enable_out 為輸出致能訊號，Corner 則為角點資訊，當 Corner=1 代表該點為角點，若 Corner=0 則不為角點。



圖 3-15 School Boy 研發電路板之外觀

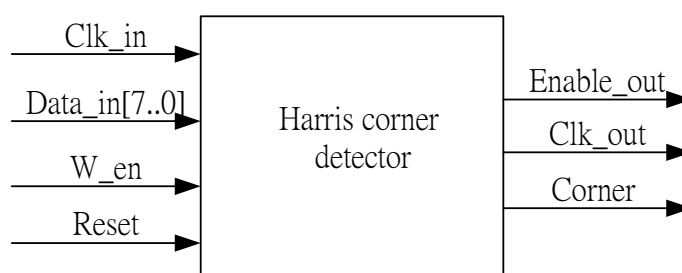


圖 3-16 Harris 角點偵測之腳位

最後我們將所偵測到的角點及水平與灰階梯度傳送到另兩個 FIFO，如圖 3-18。因為 Harris 角點偵測電路只比原影像訊號延遲兩個 PCLK 週期，故我們設單一個中斷同時抓取原影像資料、角點、水平與灰階梯度。

Harris 內部電路設計如圖 3-17 所示。圖 3-17 中，輸入的資料即為 CMOS 影像感測器輸出的影像資料，3x3 mask generator 是將灰階影像資料暫存於 FIFO 與 D flip-flop 的組合電路中，如圖 3-21。如此可將先由左而右再由上而下之影像輸出格式轉換為將 3x3 遮罩內之 9 個圖素資料同時輸出。接著將遮罩內每一個資料與其對應的 v_mask 內的值相乘。如圖 3-21 內輸出的為 $q1\sim q9$ ，則與 v_mask 相乘後又再乘 $1/8$ 即得到 Dx ，如式(3-1)。同理 Dy 求法如式(3-2)

$$Dx = \begin{bmatrix} q1 \times 1 + q2 \times 0 + q3 \times (-1) + \\ q4 \times 2 + q5 \times 0 + q6 \times (-2) + \\ q7 \times 1 + q8 \times 0 + q9 \times (-1) \end{bmatrix} \times \frac{1}{8} \quad (3-1)$$

$$Dy = \begin{bmatrix} q1 \times 1 + q2 \times 2 + q3 \times 1 + \\ q4 \times 0 + q5 \times 0 + q6 \times 0 + \\ q7 \times (-1) + q8 \times (-2) + q9 \times (-1) \end{bmatrix} \times \frac{1}{8} \quad (3-2)$$

然後再將 Dx 與 Dx 、 Dx 與 Dy 、 Dy 與 Dy 相乘分別得到 Dxx 、 Dxy 、 Dyy 。接下來則是要通過高斯平滑濾波器，此為簡化的電路，如圖 3-19。圖 3-20 則為將圖 3-19 之位置為其水平面，垂直高度則為其值。如同之前的 3x3 mask generator 電路，將大小改為 7x7 的 mask generator 電路，如圖 3-22。經過 7x7 mask generator 及高斯遮罩後，即為高斯平滑化後的 $Dxxg, Dxyg, Dyyg$ ，目的在將影像中的雜訊所造成的影響濾除。接著將設兩個閾值 $th1$ 、 $th2$ 以選取我們所要的角點的品質，如式(3-3)、(3-4)。

$$\begin{aligned} \text{if } (Dxx + Dyy) > 100 &\Rightarrow th1 = 1 \\ \text{else,} &\Rightarrow th1 = 0 \end{aligned} \quad (3-3)$$

$$\begin{aligned} \text{if } (Dxxg + Dyyg) > 0 &\Rightarrow th2 = 1 \\ \text{else,} &\Rightarrow th2 = 0 \end{aligned} \quad (3-4)$$

將所得的 $D_{xxg}, D_{xyg}, D_{yyg}$ 代入 Corner response function，式(3-5)， Res 值愈大表示為角點的可能性愈高。

$$Res = D_{xxg} \cdot D_{yyg} - (D_{xyg})^2 - 0.04 \cdot (D_{xxg} + D_{yyg})^2 \quad (3-5)$$

最後將式(3-3)~(3-5)相乘如式(3-6)得 $Res2$

$$Res2 = Res \cdot th1 \cdot th2 \quad (3-6)$$

為了要得到該角點較精確的位置，我們把 $Res2$ 輸入 5x5 mask generator 得到遮罩內所有的值，若遮罩內中心的值都比遮罩內其他的值大的話，則說該位置即為角點。



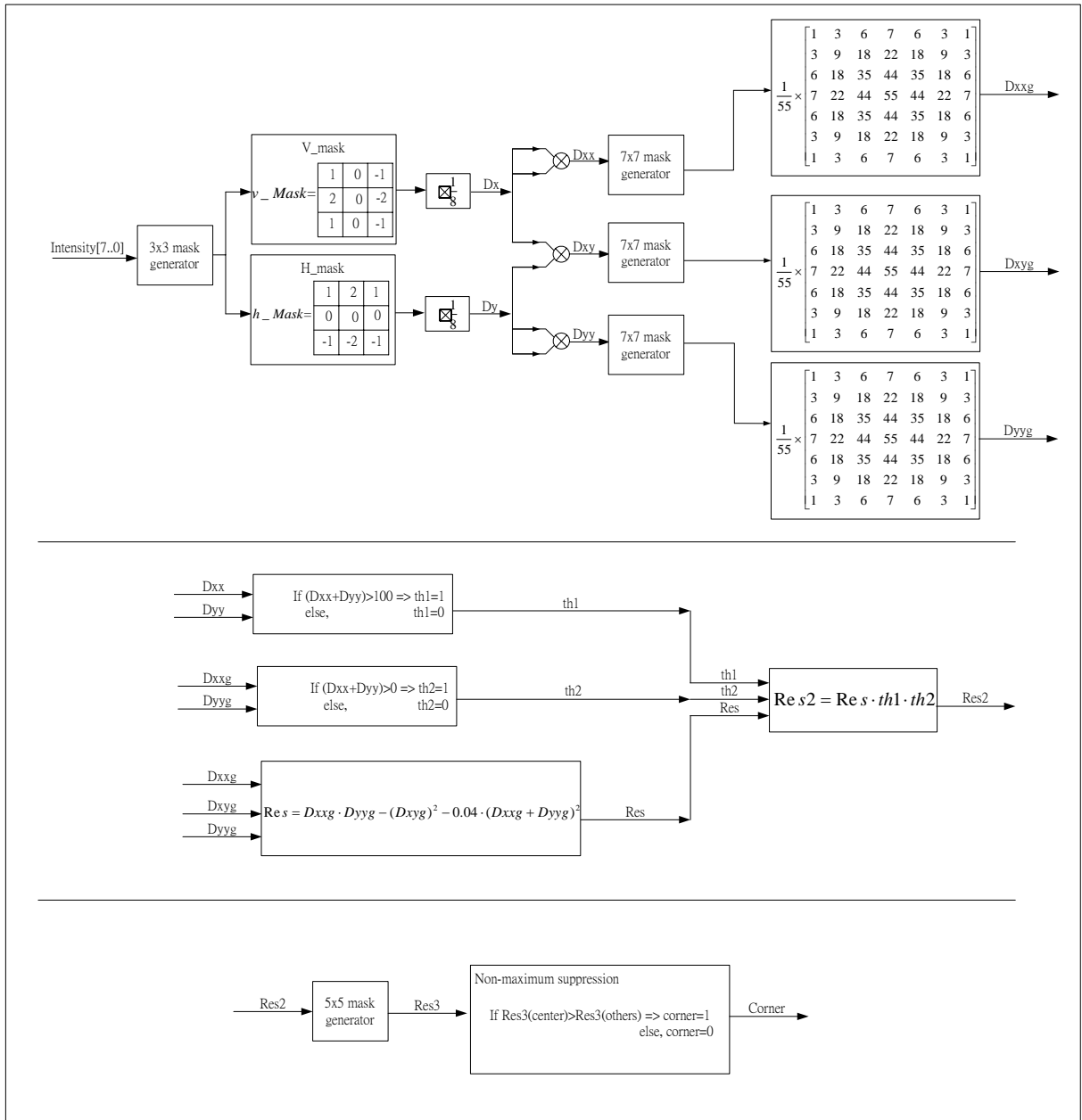


圖 3-17 Harris 內部電路

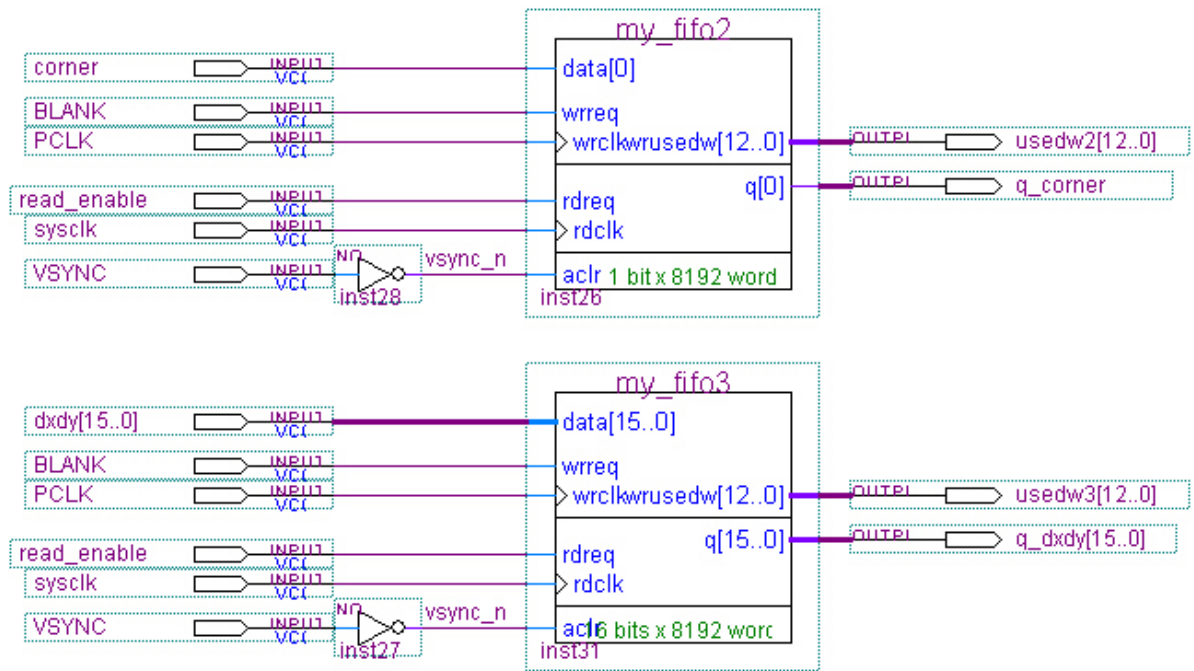


圖 3-18 角點與水平、垂直灰階梯度之 FIFO

$$\frac{1}{55} \times \begin{bmatrix} 1 & 3 & 6 & 7 & 6 & 3 & 1 \\ 3 & 9 & 18 & 22 & 18 & 9 & 3 \\ 6 & 18 & 35 & 44 & 35 & 18 & 6 \\ 7 & 22 & 44 & 55 & 44 & 22 & 7 \\ 6 & 18 & 35 & 44 & 35 & 18 & 6 \\ 3 & 9 & 18 & 22 & 18 & 9 & 3 \\ 1 & 3 & 6 & 7 & 6 & 3 & 1 \end{bmatrix}$$

圖 3-19 高斯平滑遮罩(1)

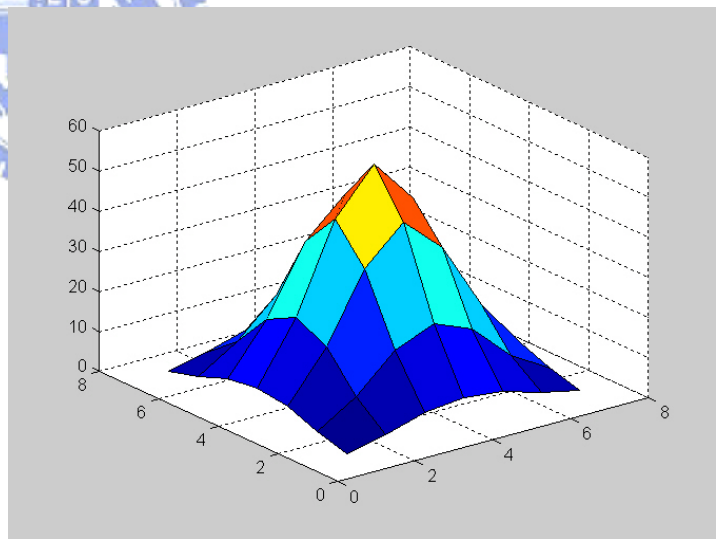


圖 3-20 高斯平滑遮罩(2)

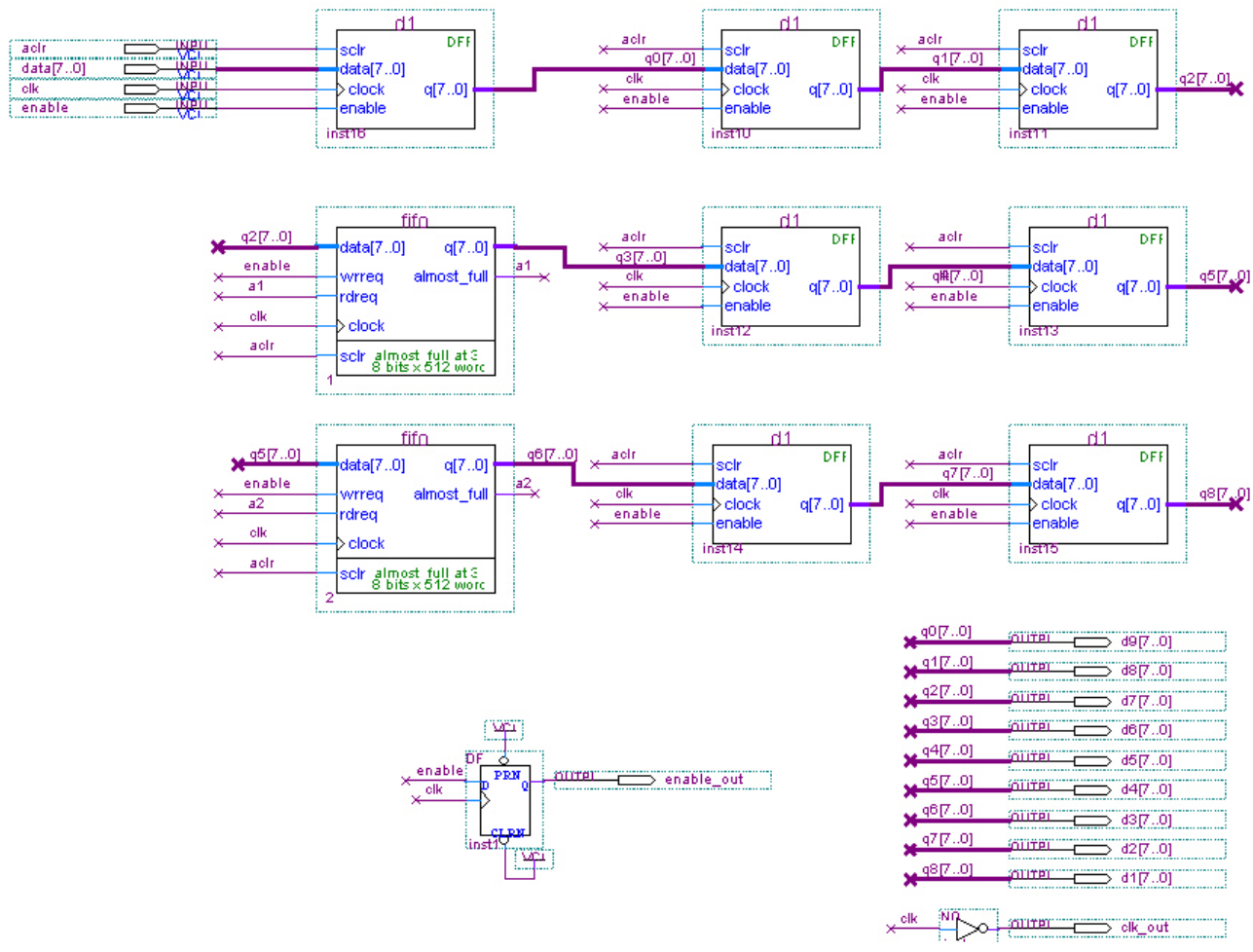


圖 3-21 3x3 mask generator 內部電路

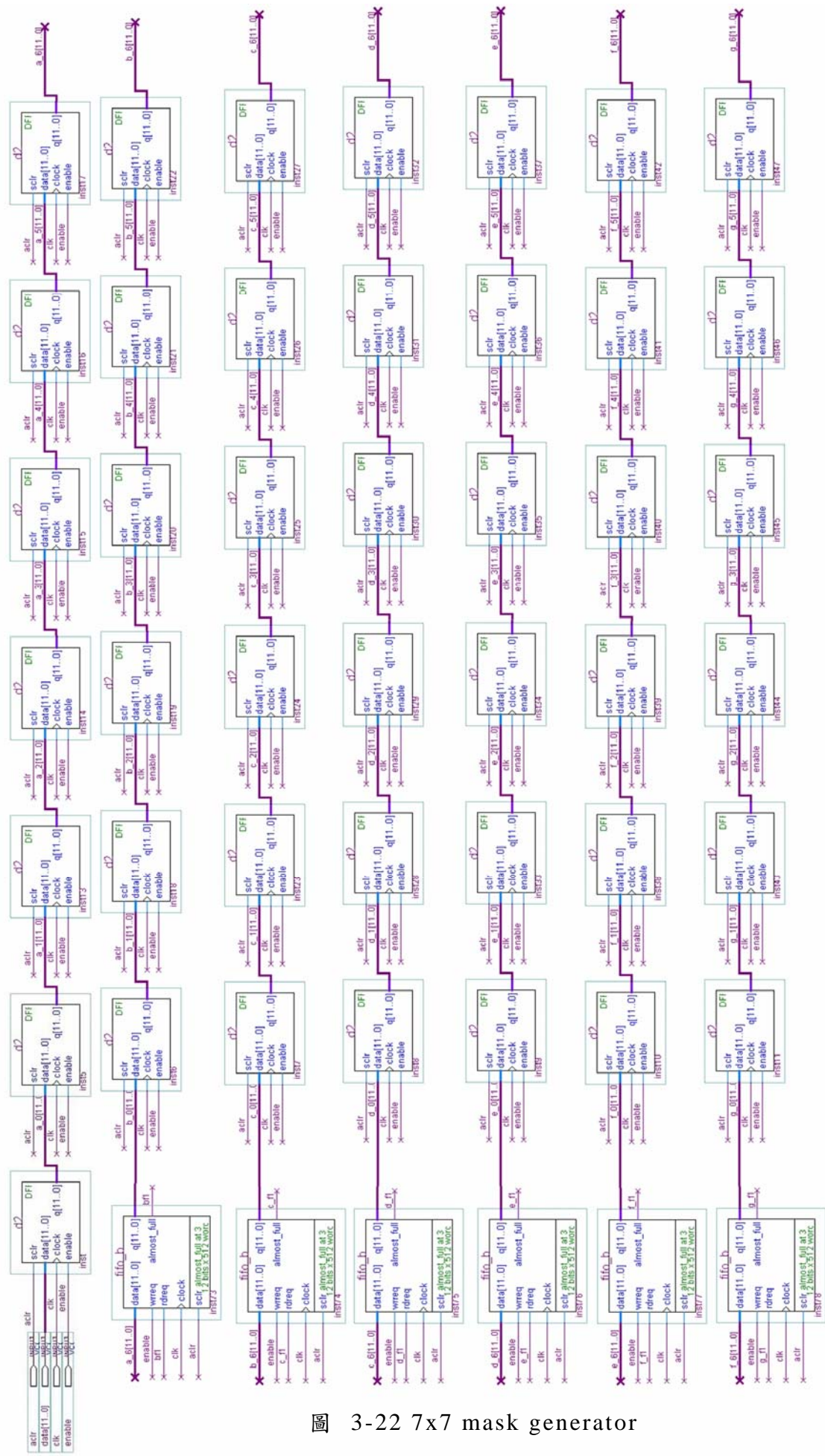


圖 3-22 7x7 mask generator

3.7 Nios 之軟體設計

Nios 之軟體設計為內部中斷向量設定、記憶體存取、角點比對、影像流分析、相關性追蹤。

3.7.1. 內部中斷向量設定與記憶體存取

為了能即時將外部資料寫入系統之記憶體中，本論文採用設定中斷向量的方式來觸發 CPU 去設定 DMA 存取外部資料。首先，我們設定 VSYNC 為每張圖像的同步觸發訊號。當 VSYNC 為 0 時，清除外部記憶體(即 FIFO)，同時將內部記憶體存放一張圖像的 Offset 設為 0，如此下次抓取圖像資料時，則可同步。另一中斷向量則是 FIFO 發出其內部資料已滿，即 7680 筆資料，此時 CPU 設定 DMA 之內部參數後隨即回到主程式中，讓 DMA 來代理抓取 FIFO 內的資料。然而為了避免後級影像處理與 DMA 的存取同時使用同一內部記憶體(在此指 SDRAM)的位置，於是將 SDRAM 設計為環形緩衝存儲器 Circular Buffer，深度為 40 張圖像資料。當後級影像處理完一張圖像時，則將 DMA 剛寫好完整的一張圖像作為下一個要處理的圖像。



3.7.2. 角點比對

由 2.4.2 節之式(2-12)知 GVM 需要三次向量的長度及一次 2 次方根，為了簡化式(2-12)，將更改為式(3-7)，其中 $v=(v_I, v_{dx}, v_{dy})$, $w=(w_I, w_{dx}, w_{dy})$ 。設一閾值為 0.2。

$$m(v, w) = \frac{|v_I - w_I| + |v_{dx} - w_{dx}| + |v_{dy} - w_{dy}|}{(|v_I| + |v_{dx}| + |v_{dy}|) + (|w_I| + |w_{dx}| + |w_{dy}|)} \quad (3-7)$$

3.7.3. 影像流分析

當角點成功地被比對在連續 10 張圖以上，則視此角點為可信賴的角點，判斷其 10 張圖像間移動特性，假設此角點由位置 (x_1, y_1) 經 10 張圖像後移動到 (x_2, y_2) 。由 2.5 節可知，前車有三種可能的運動模式，第一種以

式(3-8)來判斷，第二、三種則由式(3-9)判斷。

$$\text{if } |y_2 - y_f| + 10 < |y_1 - y_f| \cap |x_1 - x_2| < 5 \Rightarrow \text{overtaking car} \quad (3-8)$$

$$\text{if } (|y_1 - y_2| < 5) \cap (|x_1 - x_2| < 5) \cap [x_2 > (x_f + 10)] \Rightarrow \text{moving car} \quad (3-9)$$

3.7.4. 相關性追蹤與碰撞偵測

由 2.7 節知在成功地偵測到前車之特徵點時，隨即對該點取樣板 (Template)，在此設樣板大小為 5×5 pixels，由於式(2-14)計算較複雜，故改以式(3-10)實現。

$$\rho = \sum_{i=1}^m \sum_{j=1}^n |I_{ij} - M_{ij}| \quad (3-10)$$

碰撞偵測之區域則如同 2.8 節為 $179 < x < 200, 150 < y < 230$ 。

