

逢甲大學學生報告 ePaper

報告題名：

安全是唯一的路

Safety is The Only Way

作者：蔡宜瑾、鄭日翔、陳昀

系級：電子三乙、電子三甲、電子三乙

學號：D0343172、D0343984、D0386791

開課老師：陳冠宏 老師

課程名稱：專題研究(一)

開課系所：電子工程學系

開課學年：105 學年度 第二學期



中文摘要

世界衛生組織發佈的《2015 年全球道路安全現狀報告》表示，雖然行車安全已有所提升，每年仍有約 125 萬人死於道路交通事故，其中交通事故死者 46% 為步行者、自行車、兩輪機動車的使用者。有鑑於提昇行車安全的必要性，近年各大車廠已開發許多不同類型的 ADAS 系統，包含自動煞車、盲點警示偵測、前車防撞以及車道偏移偵測等等。本專題利用異質多核心嵌入式系統，以及多鏡頭的影像資訊，結合多項技術，感知車輛前方的障礙物及距離，成果方面在雙鏡頭測距方面平均誤差為 3.1%，亦即精準度為 96.9%；而障礙物偵測準確率則可達到 93.6%，並提供安全的行車建議動作。

關鍵字：雙鏡頭深度視覺、單鏡頭測距、車道線偵測、障礙物偵測

Abstract

According to the “Global Status Report on Road Safety 2015” published by the World Health Organization (WHO), more than 1.2 million people die due to traffic accidents each year though driving safety has been improved. The report indicates that 46% of the deaths were pedestrians, cyclists and motorcyclists. In light of the need to improve driving safety, a significant number of Advanced Driver Assistance Systems (ADAS) have been developed recently, including adaptive cruise control, blind-spot monitoring, forward collision warning, automatic emergency braking, lane departure warning, etc. This paper used Heterogeneous Multi-core and image data of multi-camera to combine with many technologies. We can detect the different environments and chase objects, then plan the safety route. If there is an emergency situation and the driver can't do anything, the car will go along the route to avoid obstacles automatically.

Keyword: Stereo Vision, Monocular Camera Distance Measurement, Lane Detection, Obstacle Detection

目 次

一、	摘要.....	4
二、	作品功能與構想.....	4
2-1	研究動機與目標.....	4
2-2	現有技術分析.....	5
2-3	市場調查與競爭力評估.....	5
2-4	設計流程.....	7
三、	系統實作內容.....	7
3-1	軟硬體使用.....	7
3-2	系統架構與流程.....	8
3-3	所使用演算法探討與文獻回顧.....	10
3-3-1	單鏡頭測距.....	10
3-3-2	雙鏡頭測距.....	12
3-3-3	ZED Camera 校正.....	15
3-3-4	障礙物偵測.....	17
3-3-5	車道線偵測.....	20
3-3-6	俯視環境重建與物件移動路徑預測.....	20
3-3-7	風險指數估計與路徑規畫最佳化.....	21
3-4	設計規格.....	22
四、	實作結果與功能.....	23
五、	創意設計構想.....	24
六、	結論與未來展望.....	24
七、	Demo 影片.....	24

一、摘要

道路交通事故造成財產損失也威脅生命，不論任何車種、生物或其他物體，皆可能是造成事故的潛在原因，而 ADAS 系統已是現代車輛的基本配備，但功能分散單一且大多僅止於輔助駕駛避免個人疏忽造成意外，或者遇到狀況時緊急煞車而非主動避障。本專題利用異質多核心嵌入式系統，以及多鏡頭的影像資訊，結合多項技術，感知車輛前方的障礙物及距離，並提供安全的行車建議動作，成果方面在雙鏡頭測距方面平均誤差為 3.1%，亦即精準度為 96.9%；而障礙物偵測準確率則可達到 93.6%，並提供安全的行車建議動作。

二、作品功能與構想

2-1 研究動機與目標

如何預防事故發生仍然是一個重要課題，在早期當駕駛的車輛發生碰撞意外時，是靠著被動安全配備，像是使用安全氣囊或自動緊縮式安全帶來保護車內的駕駛與乘客。如今隨著不斷提升的汽車科技，除了車體結構本身的安全性之外，讓行車更為安全的主動式輔助配備逐漸備受重視，包含循跡防滑系統、ABS 等等。而車用先進駕駛輔助系統 ADAS 包含自動煞車、盲點警示偵測、前車防撞以及車道偏移偵測等等，都讓人們開車在發生危險之前得到保護。

世界衛生組織發佈的《2015 年全球道路安全現狀報告》表示，雖然行車安全已有所提升，每年仍有約 125 萬人死於道路交通事故，其中交通事故死者 46% 為步行者、自行車、兩輪機動車的使用者。肇事原因以駕駛人因素為主，未依法規行車的案例更最為居多，其次為未注意行車狀況所導致的交通事故。由於許多意外是由於人為疏失，例如：對向車道有車輛因疲勞駕駛而偏離車道，朝我們逆向而來，這時煞車可以減少迎面衝撞的力道，不過某些情況下可藉由主動閃避來完全避免車禍，但這需要依靠駕駛的反應速度。

為了再提升行車的安全性，有整合相關技術的必要，因此本專題整合多鏡頭的影像資訊，感知車輛周遭的環境及追蹤物件，以此規劃出安全的行車路徑，並運用體積小且效能高的嵌入式系統，使其能在車輛上使用。如果緊急狀況發生而駕駛反應不及，讓車輛能沿著規劃出的路徑自動閃避，以避免發生碰撞。

2-2 現有技術分析

對於車用測距，依感測方式可分為雷達式和影像式，其中雷達式可再分為超音波(Acoustic Wave)、紅外線(Infrared)、毫米波(Millimeter wave)、光波雷達(LiDAR)，紅外線及超音波因測量距離相對較短，所以主要應用於倒車輔助和夜視系統；毫米波測量距離遠，可穿透沙塵及煙霧，較不受外在天候影響，可測量車輛間的距離及相對速度，但視場範圍較小，多應用於前車防撞系統、主動巡航控制系統及緩解撞擊剎車系統等。上述幾種雷達屬於單點測距，如果要對環境進行建模，則需對偵測範圍內連續掃描，LiDAR 就是利用其機械旋轉結構，讓雷達可大範圍掃描，產生高精度的環境建模，如 Google、百度所開發的無人駕駛車輛均有配備，是實現無人駕駛的一個重要設備。

綜上所述，以環境重建的需求上 LiDAR 是最好的選擇，不過 Google 無人駕駛項目顧問 Brad Templeton 在文獻[1]中提到，LiDAR 雖可呈現三維點雲空間，但高解析代表雷射發射器的線束越多，也就意味著造價更加昂貴，Google 無人車所使用的 Velodyne HDL-64E，其價格就高達 70,000 美元，也不可避免的有些缺點：本身體積較大且須放在車頂；機械式的 LiDAR 在掃描相對速度快的物件時，會有類似相機捲簾快門的果凍效應；LiDAR 可能會偵測到不可見的氣體如汽車的排氣；無法判別同距離下的不同物件。

影像式則可分成單鏡頭與多鏡頭，實現測距的方法並不太一樣，大多數車廠僅只在近距離如倒車時使用，或者用來輔助雷達，Tesla Model S 就是以多個不同方向的雷達+單鏡頭的方式來實現自動駕駛，但在報導[14]中提到，其自動駕駛系統 Autopilot 無法辨別道路縮減的圍欄而發生擦撞，究其原因是使用單鏡頭作測距，需要事先對要檢測的物體作影像辨識的建模，而這次事故的圍欄就不是屬於 Autopilot 可偵測的物體。而多鏡頭的影像，經由立體視覺演算法，得到緻密的三維空間資訊，不須事先建模就能達成對所有可視障礙物的偵測，也可對畫面中的物件做進一步辨識，成本也比光達或者其他雷達來的低，因此本專題著重在多鏡頭影像式的測距。

2-3 市場調查與競爭力評估

影像辨識應用廣泛，目前全球已經發展得相當多元化(圖 2-3-1)。車用方面(圖 2-3-2)，主動式安全系統已經成為了各家車廠重要的配備，包含了循跡防滑系統、ABS 等等，而近年來車用先進駕駛輔助系統 ADAS 也出現在各家車廠的配備清單當中，例如：M-BENZ Intelligent Drive、LEXUS LSS+、BMW ConnectDrive。先進駕駛輔助系統方面涵蓋許多使用鏡頭相關的方法，其中之

安全是唯一的路

一為障礙物偵測，主要目的為提升駕駛安全且有效的防止碰撞。以智慧科技發展來說，運用在智慧機器上，可以讓機器得知周遭環境，且做出相對應的對應方式，提升機器的反應速度。

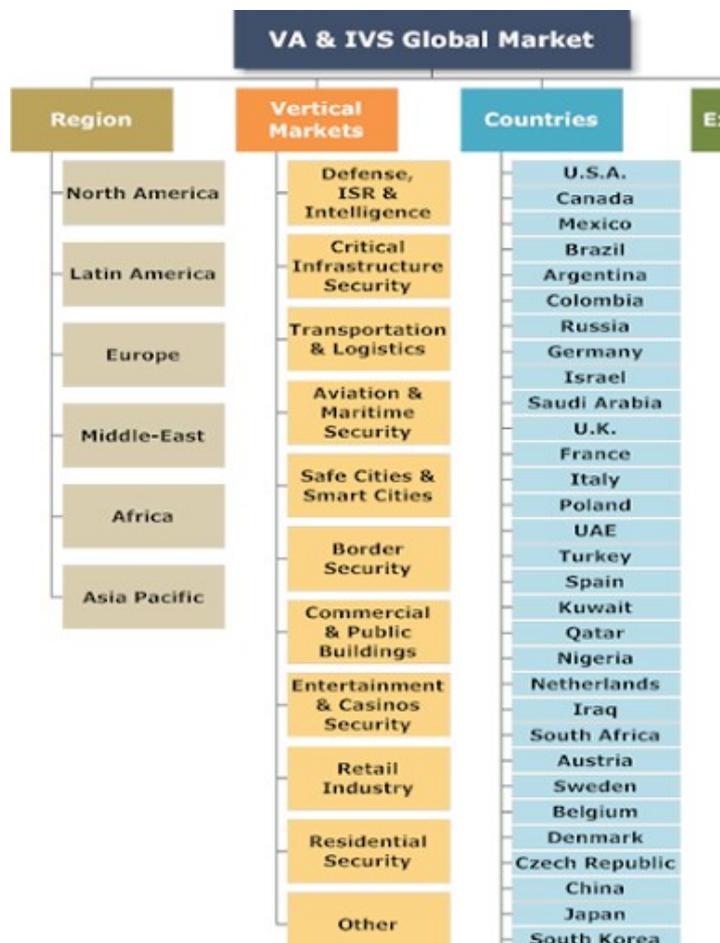
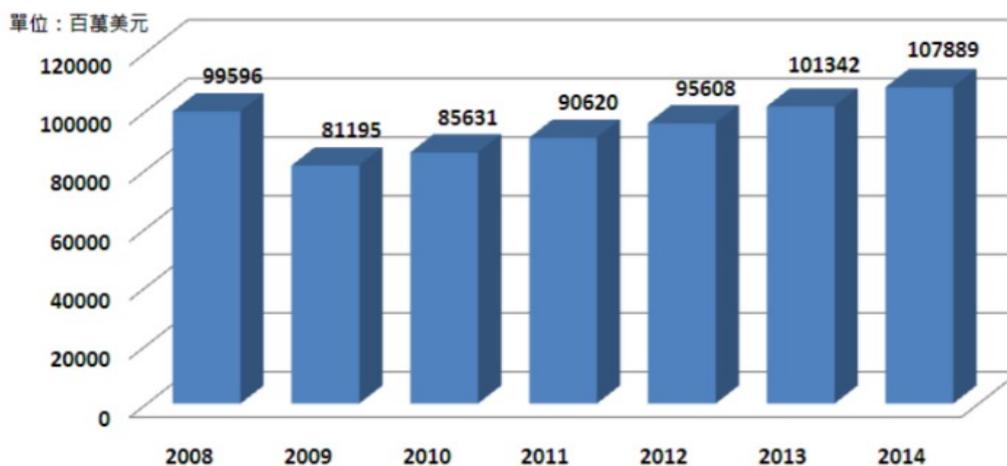


圖 2-3-1、全球智慧影像辨識應用各國及產業領域拓撲圖



資料來源：iSuppli、車輛研究測試中心(ARTC)整理

圖 2-3-2、全球車電市場趨勢圖

2-4 設計流程

本專題以影像視覺為系統核心，在思考所需功能並決定系統架構後，開始搜尋文獻資料並研究其原理，使用 OpenCV Library 進行演算法測試與驗證，先實現其基礎功能如雙鏡頭深度視覺、單鏡頭測距模型等，再決定整體系統架構，並從中加入自己的創意構想做結合。有了每個功能後，開始做全面性的整合，並驗證系統功能完整度，最後調整細部參數來優化其效能。

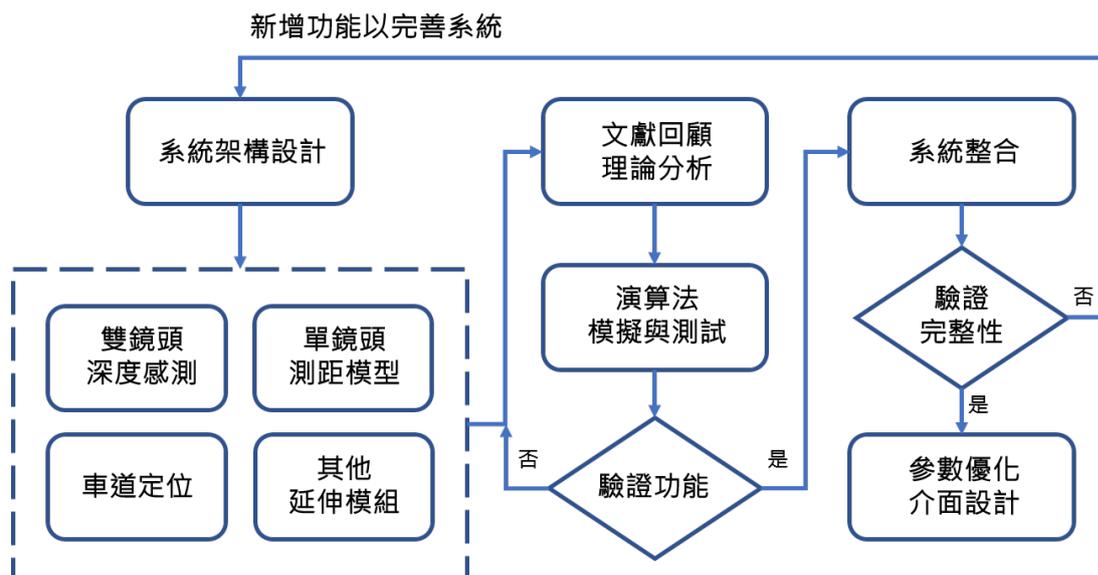


圖 2-4-1、系統設計流程圖

三、系統實作內容

3-1 軟硬體使用

軟體: CUDA、OpenCV、MATLAB

硬體: NVIDIA Jetson TX1 or PC with NVIDIA Graphics Cards、
ZED Stereo Camera

3-2 系統架構與流程

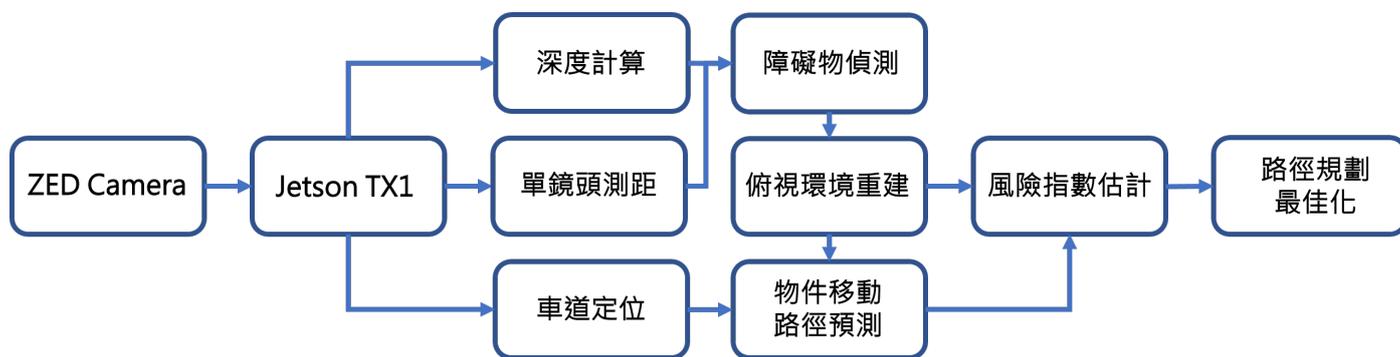


圖 3-2-1、系統架構流程

圖 3-2-1 是系統整體抽象層面上的架構與流程圖，以目標來說，本專題目標是實時規劃安全行車路徑，需要感知車輛周遭環境，追蹤和預測物件是否會與自己碰撞，並規劃出最安全的行車路徑。以下為各部份做介紹，而 3-3 節將詳細介紹演算法流程。

1. ZED Camera 及 Jetson TX1

ZED Camera 是一個雙鏡頭攝影機，可搭配 NVIDIA Jetson TK1、TX1 或者 NVIDIA Graphics Cards 使用 CUDA 來生成緻密深度圖，原廠提供的 SDK 本只可量測到最遠 20 公尺且有誤差，在自行校正後最遠可達 70 公尺的距離並保持 10% 以下的最大誤差；而 Jetson TX1 優點在於擁有一般家用 PC 顯示卡的效能但整體功耗在 20W 以內，非常適合在車輛上使用。

2. 雙鏡頭深度計算、單鏡頭測距與障礙物偵測

利用雙鏡頭產生如圖 3-2-2 的深度圖，藉由單鏡頭測距，回推實際地面距離在畫面中的位置，再參考以上兩者資訊，找出真正的障礙物。



圖 3-2-2、深度計算與障礙物偵測
(左上：實際畫面、左下：深度圖、
右下：障礙物偵測)

3. 俯視環境重建

大多數自動駕駛解決方案，都是使用 LiDAR 或者多鏡頭生成三維點雲空間，其運算量十分龐大，分析上也較為困難，我們認為路面行駛車輛只須注意前後左右的方向，而俯瞰圖可以減少資訊量，駕駛或者系統不需要了解前方障礙物是什麼，只需知道不能撞上任何物體，也簡化分析上的困難。利用障礙物偵分離出車輛周遭應該注意的物件，並以其深度值從相機模型回推物件的世界座標，重建車輛的俯瞰畫面。並將車道定位的結果標示在俯瞰圖上。圖 3-2-3 的俯視圖中，灰色為被障礙物遮擋的視線，而綠色為可行車的空間。



圖 3-2-3、前向鏡頭偵測結果與俯視環境示意圖

4. 車道定位、物件移動路徑預測、風險指數估計與路徑規劃最佳化

以車道定位為輔，追蹤所有分離物件的距離及偏移變化，以提前預判其移動路徑是否會與本車的前進路徑相交會，再評估其風險指數、是否可能發生碰撞，最後依照情形決定進行避障或者煞車。

圖 3-2-4 為範例圖，圖中中間車輛為本車，紅色方塊為障礙物偵測結果，紅色與綠色箭頭長度為物件預測路徑與速度，黑色箭頭為本車建議行車路徑與速度。左圖中，前車因不明原因減速，左車道為逆向車道，右車道有車輛快速接近，因此建議繼續直行並減速；右圖中，有車輛從左前方逆向而來，而右車道無物體，因此建議往右車道移動，避免和逆向車發生碰撞。

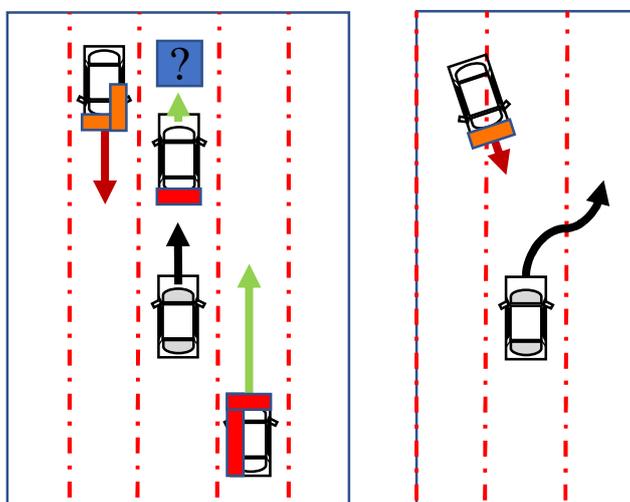


圖 3-2-4、物件預測路徑和建議行車路徑示意圖

3-3 所使用演算法探討與文獻回顧

3-3-1 單鏡頭測距

單鏡頭測距大致可分為兩種，第一種是先建模再測距，如文獻[2]透過相機模型的內部參數、外部參數、畸變係數，求得世界坐標與影像坐標的幾何關係，便可以得到相機與目標物體的真實距離；第二種則先測距再建模，如文獻[3]透過量測畫面中像素(Pixel)所對應的真實距離，使用數據回歸模型求得影像坐標與距離的函數關係。

有一簡單模型只須相機高度、焦距以及與地面水平夾角，再利用其模型幾何關係，將物體呈現在影像座標上的像素換算成實際的距離，來達成測距的效果，或將想知道的距離回推到影像座標上。

1. 單鏡頭測距模型幾何關係推導：

- a. 參考圖 3-3-1，先假設一攝影機其透鏡光軸延伸至地面距離為 d ，攝影機高度為 h ，光軸與地面夾角 θ ，而成像水平線在影像座標上為 y_0 ，我們假設 y_0 為影像中心水平線。其關係式為

$$d = \frac{h}{\tan\theta}$$

- b. 現在想得知距離 d' 位在影像座標的何處，其夾角為 θ' ，則

$$d' = \frac{h}{\tan\theta'} , \quad \alpha = \theta - \theta'$$

- c. 再假設攝影機焦距 f ， y' 為所測距離的影像座標，可以得到

$$\alpha = -\arctan\left(\frac{y' - y_0}{f}\right) , \quad y' > y_0$$

- d. 即可求得

$$d' = \frac{h}{\tan\left[\theta + \arctan\left(\frac{y' - y_0}{f}\right)\right]} \quad \text{or} \quad y' = \tan\left[\arctan\left(\frac{h}{d'}\right) - \theta\right] \times f + y_0$$

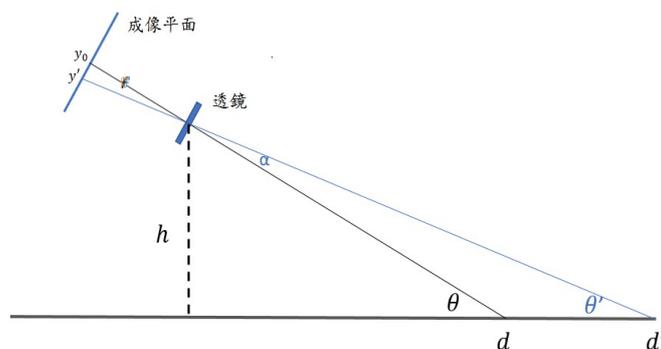


圖 3-3-1、單鏡頭測距幾何模型

2. 實際驗證

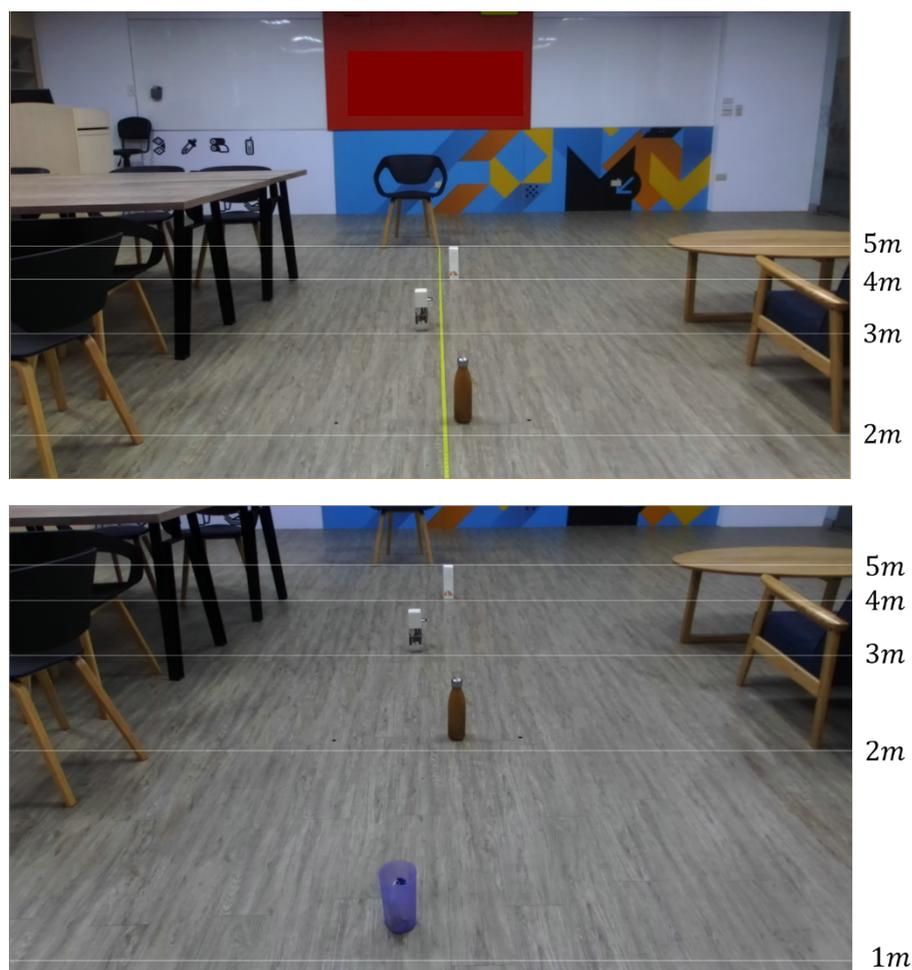


圖 3-3-2、驗證結果

在圖 3-3-2 實際距離中，每公尺放置一標記物，而白線為幾何模型推算後的位置，可以看出，因為忽略相機成像以及透鏡所產生的失真，導致近距離部分有較明顯的誤差，但這簡單模型仍有其精準度。

就如 2-2 小節現有技術分析中所提到，單鏡頭在偵測物體時，需要事先決定要判斷何種物體，才能進行影像辨識的建模，測距的精準度在於影像辨識的 ROI 是否可精確地框住，也要確保能完全偵測到被測物，且單鏡頭測距需假設鏡頭與地面夾角不會改變，導致數據偶有錯誤或者遺漏，較適合用在近距離且只偵測少數被測物的情況。

綜上所述，我們並不直接使用此方法來做物體的測距，而是只去找出畫面中平面道路的位置，並結合雙鏡頭深度視覺來達到障礙物偵測的效果。

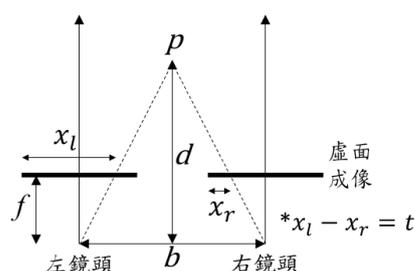
3-3-2 雙鏡頭測距

雙鏡頭的深度計算大致可分為三大步驟：雙鏡頭影像校正、比對代價估算及相機模型三角測量：

1. 相機模型(Camera Model)與三角測量

如圖 3-3-3，假設已知被測物體 p 在左右鏡頭中的視差距離 t (pixel)，相機焦距為 f (pixel/mm)，兩鏡頭光心之間的距離 b (mm)，即可計算被測物體與鏡頭距離為 $d = bf/t$ (mm)。

圖 3-3-3、三角量測



不過在找出視差之前，需先完成相機模型。有三個坐標系統：世界座標系統(World Coordinate System, WCS)、相機座標系統(Camera Coordinate System, CCS)、影像座標系統(Image Coordinate System, ICS)。

將 WCS 轉為 CCS 的矩陣稱為外部參數矩陣(Extrinsic Parameters)，由三維旋轉矩陣 R 和平移矩陣 T 所組成，通常會把 WCS 設為其中一台相機的 CCS，便只需要求另一個相機的外參矩陣，而將 CCS 轉為 ICS 的矩陣稱為內部參數矩陣(Intrinsic Parameters)，由 x 和 y 方向焦距上，每單位距離的像素數 α_x, α_y 、光心位置在 IS 上的座標 u_0, v_0 。式 3-1 是 WCS 與 ICS 的轉換矩陣。

$$\begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \begin{bmatrix} \alpha_x & 0 & u_0 \\ 0 & \alpha_y & v_0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} r_{11} & r_{12} & r_{13} & t_1 \\ r_{21} & r_{22} & r_{23} & t_2 \\ r_{31} & r_{32} & r_{33} & t_3 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} \quad (\text{式 3-1})$$

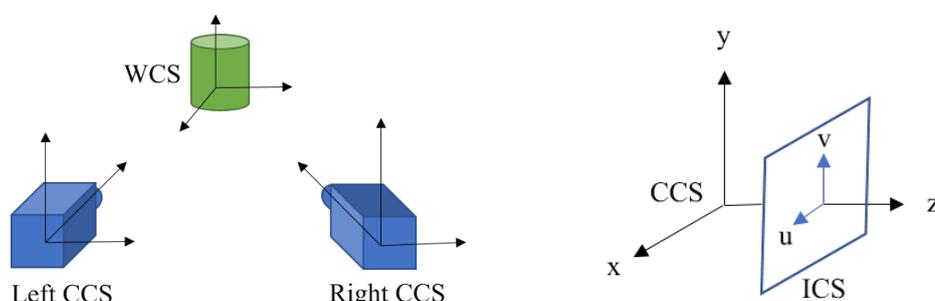


圖 3-3-4、三種座標系統

2. 比對代價(Matching Cost)估算

比對代價估算是去搜尋被測物體在雙鏡頭中對應的位置，計算兩像素在影像座標中的差距，便是被測物體的視差。代價(Cost)依類型可分為：

- (a) pixel-based [4]: Absolute Intensity Differences(AD), Squared Intensity Differences (SD), Color Distances(CD), Sampling-Insensitive Absolute Differences(Birchfield and Tomasi, BT) [5]
- (b) window-based: Sum of Absolute Differences (SAD), Sum of Squared Differences (SSD), Normalized Cross correlation(NCC), Rank and Census Transforms
- (c) Mutual Information(MI)

圖 3-3-5 是測試圖以及模擬深度圖，圖 3-3-6 是計算的深度圖，其中，BT 是 AD 的優化版，藉由線性插值函數，來減少因影像像素的離散性，而造成比對代價估算時的誤差。SAD/SSD 是 AD/SD 的視窗版本，藉由累加視窗範圍內的代價，提高信噪比來減少大部分的錯誤匹配。



圖 3-3-5、tsukuba(左)、groundtruth(右)

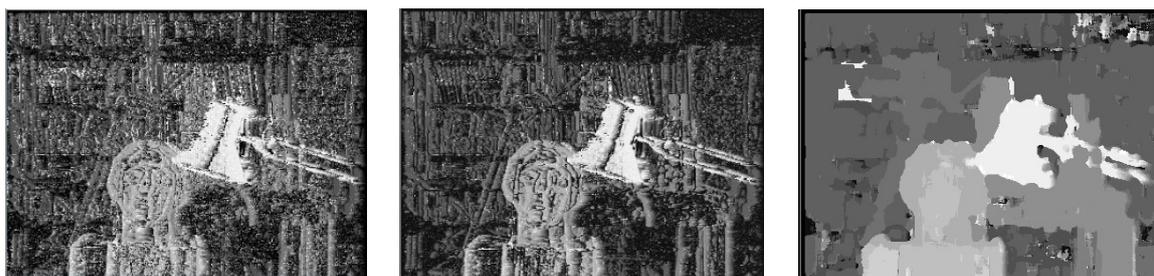


圖 3-3-6、AD(左)、BT(中)、SAD(右)

而比對演算法也分成三大類：區域(Local)、全域(Global)、半全域(Semi-Global)，區域演算法最常用的是 Winner-Take-All(WTA)，尋找極線上範圍內的代價極值為匹配點，可用在所有代價上，但對於低紋理區域的匹配錯誤率較高，如圖 3-3-6 SAD 的右上角；全域演算法則去考慮整張影像，定義一能量函數並基於像素求其全局最優解，可解決 WTA 對影像中低紋理區域的匹配問題，常見的演算法有 Belief Propagation(BP)、Graph Cuts(GC)、

Dynamic Programming(DP)；半全域演算法利用了區域 WTA 及全域 DP 的觀念，以左影像中一條水平像素為基準，向右影像中相同水平位置的像素進行比對，再透過多個一維方向做 DP 影像匹配，模擬二維影像的全域匹配最佳化。

文獻[6]-[7]對目前立體視覺演算法做了比較與評估，[6]主要使用 pixel-based 和 window-based，區域和全域演算法並加入了代價優化前景膨脹效應濾除，和後處理如相似點濾除、孤立點濾除；[7]測試了 window-based 和 MI，增加測試半全域演算法，並加入前處理如 Laplacian of Gaussian (LoG)、Rank、Mean filter。

在做文獻回顧並實際編寫程式去生成深度圖後，我們了解到雙鏡頭深度視覺的運算量龐大，且生成高質量的稠密深度圖原理複雜，並非一蹴可幾，短時間內無法達成，因此我們決定使用 ZED Camera 以及 TX1 配合 CUDA 來取得較精確的深度圖以及較低運算時間。

3. 雙鏡頭影像校正

影像需要校正的部分，有攝影機裡透鏡(Lens)所產生的徑向畸變(Radial Distortion)與切向畸變(Tangent Distortion)，以及雙鏡頭的極線(Epipolar line)對齊。

徑向畸變是由於透鏡的形狀，使得光線在通過透鏡時，因離透鏡中心不同距離，會有不一樣的折射角度，產生桶狀(Barrel)、枕狀(Pincushion)等不同的畸變。切向畸變則是由於鏡頭模組組裝的誤差，使透鏡和感光元件並非完全平行對齊所造成的。徑向畸變模型如式 3-2、3-3，切向畸變模型如式 3-4、3-5。其中， x, y 為 CCS 相機座標， $x_{distorted}, y_{distorted}$ 為失真後的影像座標， k_1, k_2, k_3 為徑向畸變係數， p_1, p_2 為切向畸變係數， $r^2 = x^2 + y^2$ 。

$$x_{distorted} = x(1 + k_1r^2 + k_3r^4 + k_3r^6) \quad (式 3-2)$$

$$y_{distorted} = y(1 + k_1r^2 + k_3r^4 + k_3r^6) \quad (式 3-3)$$

$$x_{distorted} = x + [2p_1xy + p_2(r^2 + 2x^2)] \quad (式 3-4)$$

$$y_{distorted} = y + [p_1(r^2 + 2y^2) + 2p_2xy] \quad (式 3-5)$$

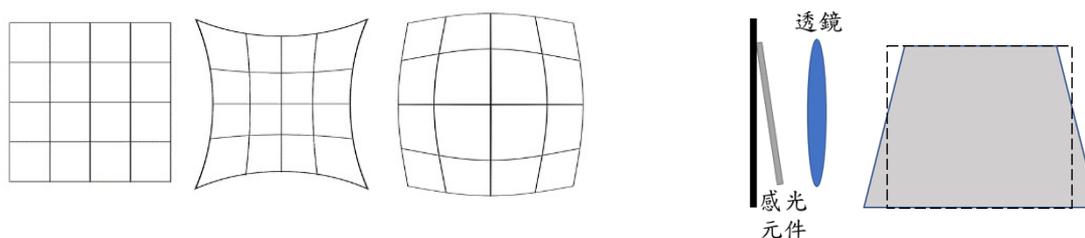


圖 3-3-7、徑向畸變與切向畸變

極線對齊則是為了減少視差估算的比對時間，在二維影像上去比對所有像素極為耗時，為了減少比對範圍，可以利用極線約束的特性，也就是雙鏡頭中對應的特徵點會出現在同一條極線上，藉由對齊雙鏡頭的極線，使得雙鏡頭像素的比對範圍由二維降為一維水平線上，節約時間並可排除許多錯誤匹配。

文獻[8]利用一標定物如棋盤格，找出相機的內部參數、外部參數、畸變係數，便得以校正雙鏡頭的畫面。圖 3-3-8 為實際校正結果。

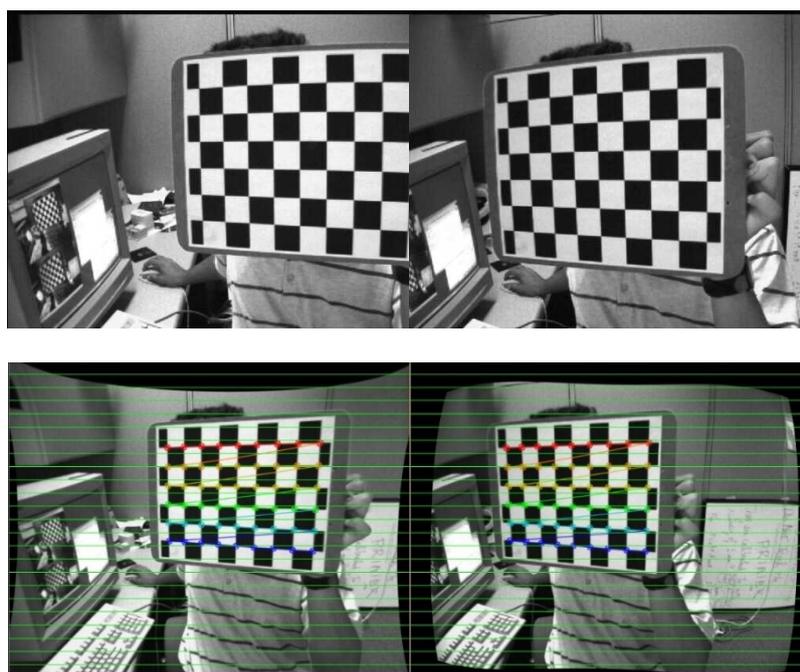


圖 3-3-8、校正前(上圖)、校正後(下圖)的影像

3-3-3 ZED Camera 校正

從原廠 SDK 所取得的深度圖，最遠範圍只有 20 公尺且有誤差，並與距離成正比，在圖 3-3-9 中，量測值為 SDK 所取得深度數值，誤差值為實際距離減量測值，誤差值比例為誤差值除以實際距離。圖中可看到校正前量測值在 20 公尺時誤差比例已達 50%，但呈現曲線分佈的形式，我們分析真實距離和量測值數據取得其一元二次漸近線方程式，將量測值拉回真實距離，再重新測試並分析。最終統計發現，依實際距離分成三部分：0~2、2~20、20~70 公尺，使用不同的漸近線方程，如式 3-6，可達到最低的誤差。圖 3-3-9 為校正前後的實際、量測與誤差數據，其原始平均誤差值比例為 46.04%、最大為 76.8%，而校正後平均誤差值比例則降至 3.11%，最大為 10.74%。

$$\begin{aligned} 0.0012X^2 - 1.8684X + 1697.3, & \quad \text{when } X \leq 2000 \\ 0.0012X^2 - 0.0258X + 1888.5, & \quad \text{when } 2000 < X < 20000 \\ 0.0012X^2 - 22.841X + 128912, & \quad \text{when } X > 20000 \end{aligned} \quad (\text{式 3-6})$$

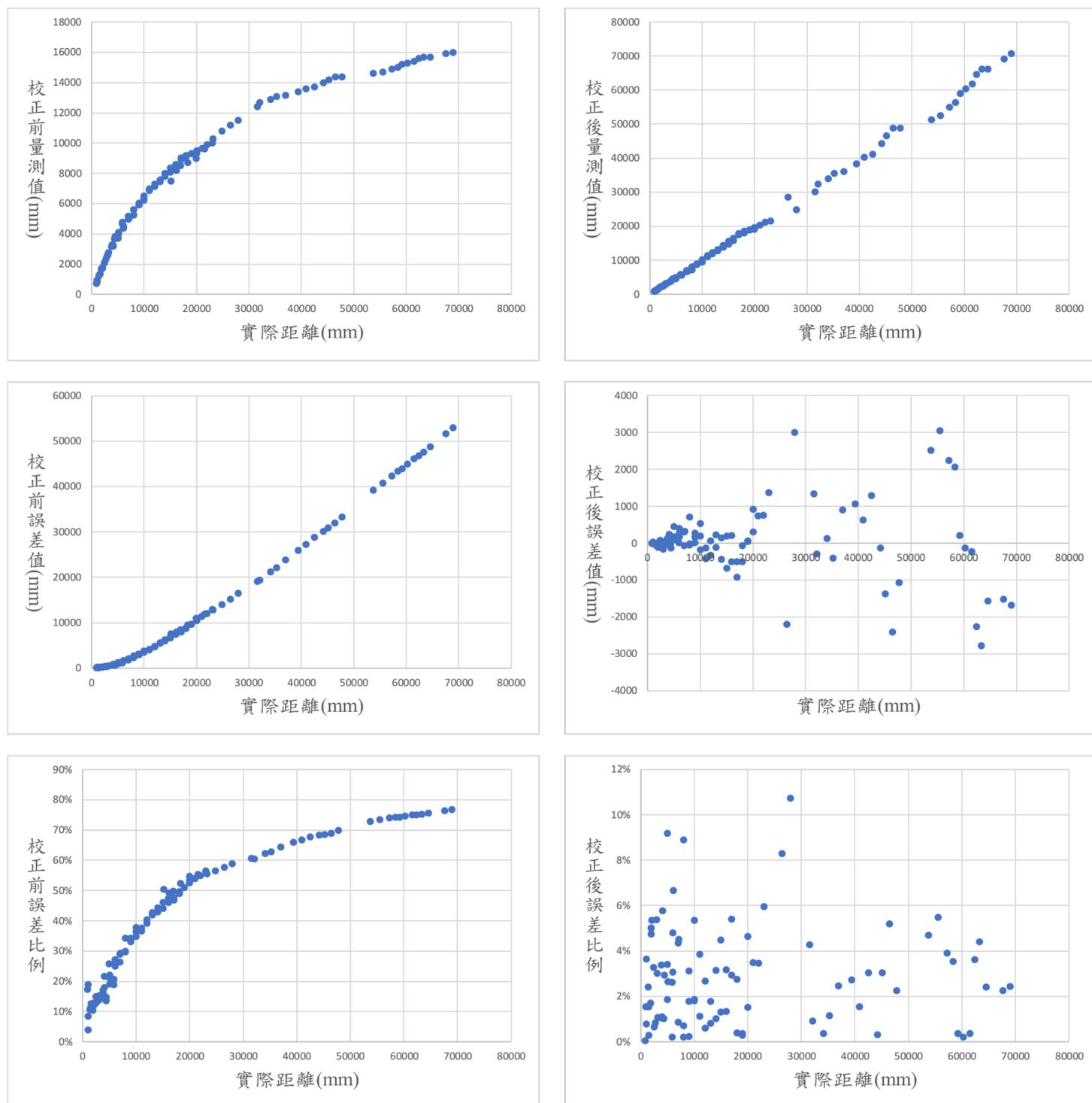


圖 3-3-9、校正前後的數據

3-3-4 障礙物偵測

在這裡我們利用雙鏡頭所生成的視差圖以及單鏡頭測距模型來達成，圖 3-3-9 為前向鏡頭畫面示意圖，假設每一公尺實際距離所在影像座標畫面畫一紅線，中心水平線為地平線，藍色長方體為障礙物，當我將每條紅色線之間的地面深度值數值設為各設為與底部紅色線對應距離後，其深度圖如 3-3-11 示意圖，而障礙物會位在 2~3 公尺之間，其深度值都為 2 公尺，此時如果已知 3 公尺水平線位在何處，便可往上找出非地面區域，如圖 3-3-12 橘色框處為障礙物。

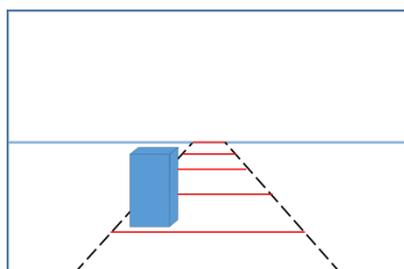


圖 3-3-10、前向鏡頭示意圖

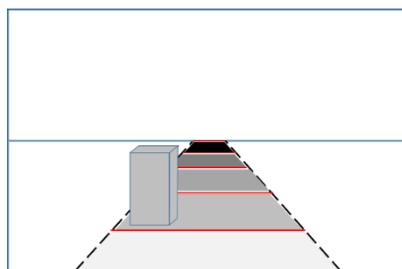


圖 3-3-11、深度圖示意圖

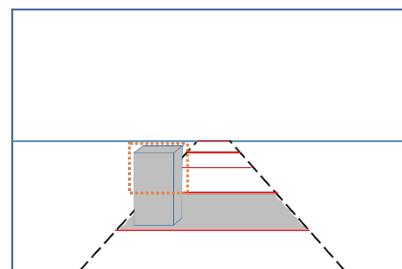


圖 3-3-12、障礙物偵測示意圖

為了增加視差圖中各個物件的連續性，首先如圖 3-3-13 將取得視差圖的位元深度(Bit Depth)從 8 bits 降為 4 bits，讓整張視差圖可直接分出 16 層，再來將每層抽出，並利用單鏡頭測距模型，推算圖 3-3-14 中所在層紅線對應的影像座標位置，便可往上找出障礙物，至於搜尋框的寬度，則可根據偵測精度的要求做調整，如圖 3-3-15 寬度為 15 pixel 的實際偵測結果。

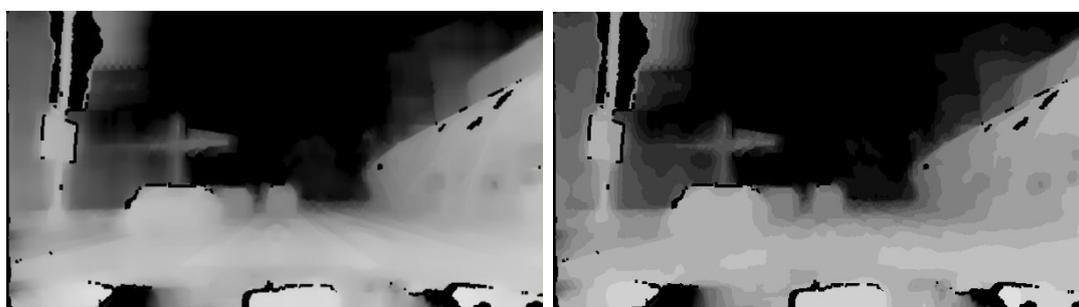


圖 3-3-13、減少位元深度

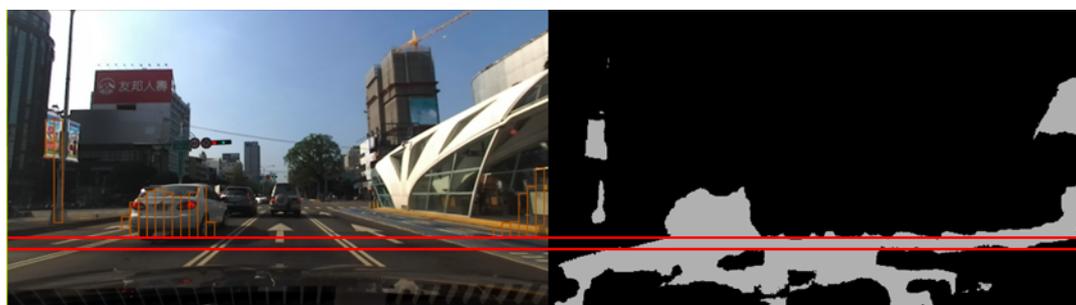


圖 3-3-14、找出地面範圍



圖 3-3-15、障礙物偵測最終結果

此偵測的方法目標是找出畫面橫軸中所有離車輛最近的障礙物，因此準確率的計算方式為：統計正確找出離車輛最近的障礙物的框，去除以總框數。我們收集了各種行車環境下的測試影片，並挑出每個影片的一部分影像作紀錄，如圖 3-3-16 是各個環境下的障礙物偵測結果，其搜尋框的寬度為 5 pixel。表一中統計 50 張具有障礙物畫面的偵測結果，我們發現準確率可達到 93.6%。

TP	FP	FN
23329	683	1588
Recall		Precision
93.6%		97.2%

表一、準確率分析，其中 TP(True Positives)正確辨識為障礙物，FP(False Positives)不是障礙物卻被認為是，FN(False Negatives)是障礙物卻沒有被辨識到。

至於 6.4%的錯誤，主要來自高度較低的障礙物，如圖 3-3-16(b)左下角處的安全島。因車輛在行駛時，不可避免地會使相機輕微的上下晃動，使相機與地面角度是有小幅度變化的，而目前的單鏡頭測距模型是根據固定的角度做計算，所以為了避免出現過多誤判，實際在演算法中有加入最小障礙物影像高度，以此過濾大部分的誤判。因此像安全島在畫面中可能所佔影像面積過小，而沒有被成功辨識成障礙物。



圖 3-3-16(a)、高速公路



圖 3-3-16(b)、市區大道



圖 3-3-16(c)、小巷道

3-3-5 車道線偵測

在偵測之前通常會做些前處理，常見的做法是先對影像設定車道的 ROI 感興趣區間做逆透視矩陣轉換，得到俯視的車道圖，再經由灰階、二值化、Sobel 邊緣檢測，測出車道線邊緣之後，進行下一步車道線偵測。文獻[9]利用 Hough 轉換找出車道線，實際效果如圖 3-3-17，對於彎曲的車道線效果不彰。文獻[10]提出車道線可以由一拋物線的數學形式表示，帶入車道線上三個點，利用高斯消去法求得所需的參數，便可畫出符合目前畫面中車道的曲線，實際效果如圖 3-3-18。文獻[11]除了形態學處理和 Hough 轉換外，加入遺傳算法(Genetic Algorithm)和區域成長(Region Growing)來減少誤判。文獻[12]利用 Adaptive Randomized Hough Transform、模板匹配(Template-Matching)達到更有效率的偵測方法。文獻[13]整理各種方法和各自可應用的基本條件及優缺點。



圖 3-3-17、直線車道線



圖 3-3-18、曲線車道線

3-3-6 俯視環境重建與物件移動路徑預測

從 3-3-2 小節中我們已可從視差圖求出物體在世界座標中 Z 軸深度為 $d = bf/t$ ，而世界座標的 X、Y 軸如式 3-7，其中 u, v 為影像座標點， u_0, v_0 為光心所在的影像座標， a_x, a_y 為每單位距離的像素數。

$$x = (u - u_0) \frac{d}{a_x}, y = (v - v_0) \frac{d}{a_y} \quad (\text{式 3-7})$$

而我們只需要俯瞰圖，計算 d 和 x 即可，搭配障礙物偵測與車道線偵測，將所有可能的障礙物藉由取得的 d 和 x 映射出一張俯瞰圖，並以此來追蹤物件及計算其預測路徑。



圖 3-3-19、前向鏡頭偵測結果與俯視環境示意圖

3-3-7 風險指數估計與路徑規劃最佳化

舉前車相對距離為例，我們錄製一段測試影片並記錄影片中前車與本車的距離，其總幀數為 11134 張，如圖 3-3-20，先測試影片中我們認為需要煞車的部分做標記與擷取，如圖 3-3-21、22 為第 382~584 張的部分。如式 3-8，我們定義一煞車指標為 $\Delta d/d$ ，以及判斷是否需要煞車的閾值，其中 d 為前車距離， Δd 為每單位時間 d 與前一個 d 的差值，其理念在於前車越遠，可容許越大的前車距離減少範圍，反之亦然。

$$\frac{\Delta d}{d} \geq threshold, Warning \quad (式 3-8)$$

$$\frac{\Delta d}{d} < threshold, Safe$$

以本測試影片來說，單位時間為錄影時的幀數的倒數，而閾值則是通過分析觀察所取得，在統計每一幀與前一幀的差值，再去除以距離，並觀察其數值，如圖 3-3-23，我們將 0.01 做為初始閾值。接下來放進影片測試預測結果，如果出現不符合預期結果的部分，則再調整閾值，直到符合預期為止，比如說在測試影片中不需要煞車的地方但程式判斷需要煞車，則需要上修閾值，反之亦然。在統計前車距離、障礙物所佔畫面的面積等等量化數值後，再設定各部份權重並觀察分析及調整，以此估算整體風險指數分析，另外再細分為前車、左右車道等等，以此來決定車輛的安全行車路徑。

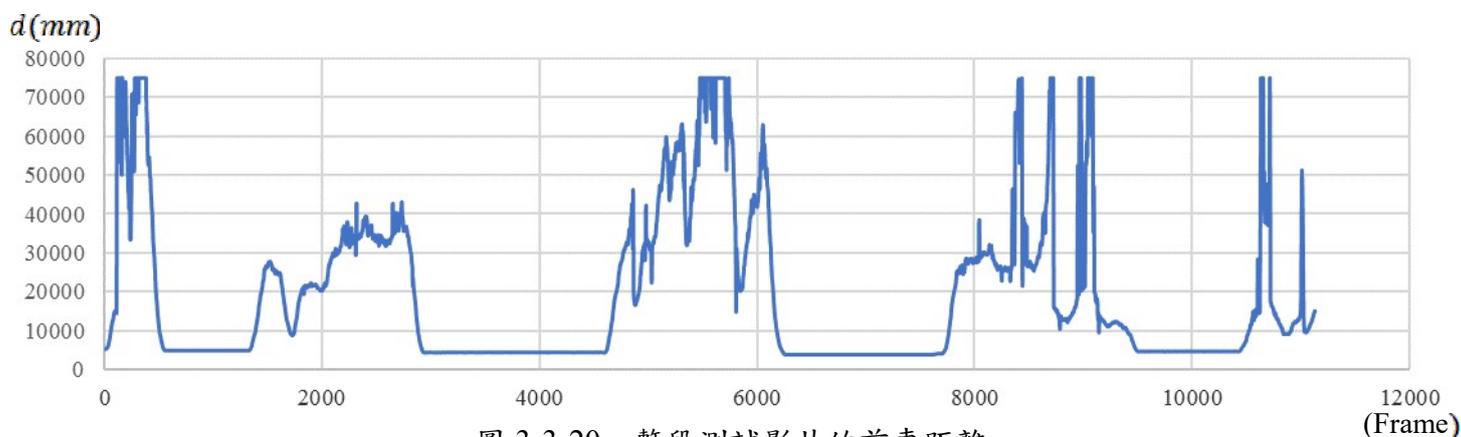


圖 3-3-20、整段測試影片的前車距離

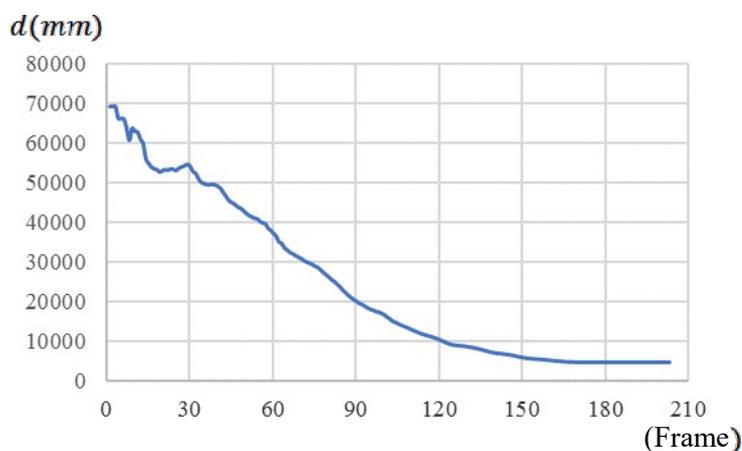


圖 3-3-21、第 382~585 張每一幀的前車距離

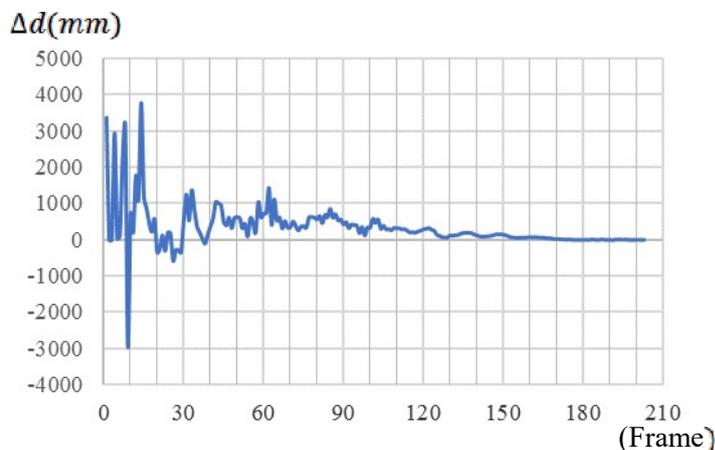


圖 3-3-22、每一幀與前一幀的差值

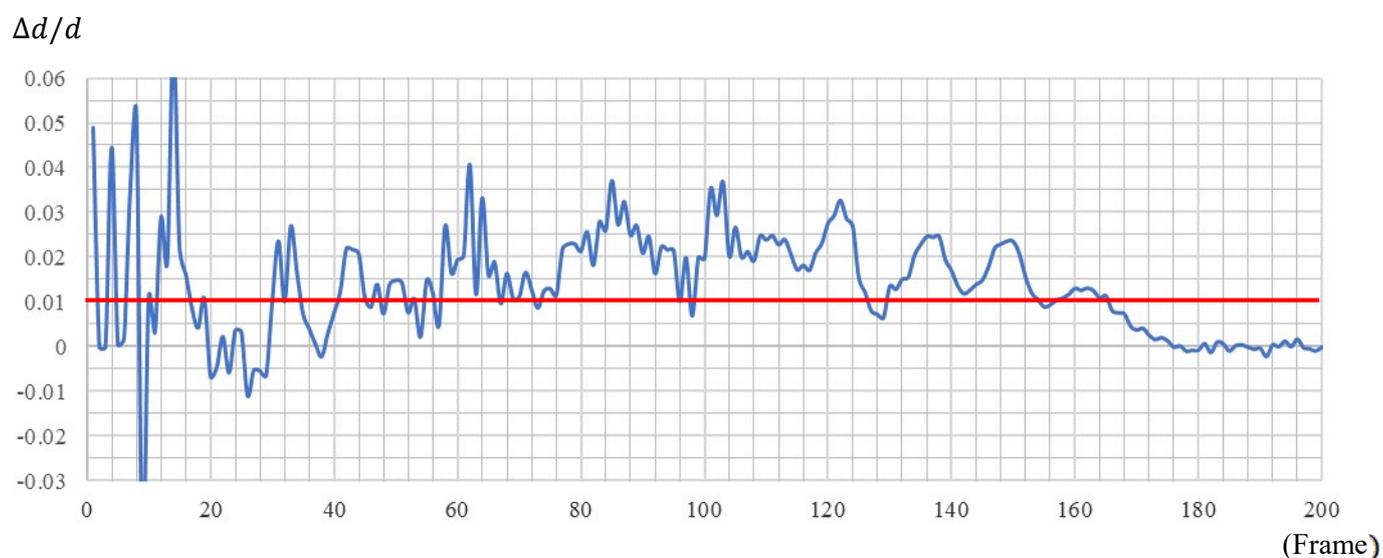


圖 3-3-23、剎車指標

3-4 設計規格

Item	Specifications
Technology	Stereo Match、Monocular Camera Distance Estimation、Lane Detection、Obstacle Detection、Environment Reconstruction、Estimation of Collision Risk
Maximum Distance	70m
Distance Error	Avg: 3.1%, Max:10.7%
Camera Resolution	1280x720 x 2
FPS	15~30
Field of View	110 degrees
Embedded System	NVIDIA Jetson TX1
Power consumption	5~20 W

四、實作結果與功能

本專題已完成並整合深度計算、單鏡頭測距、障礙物偵測及部分的風險指數分析並提供行車建議動作。在 Jetson TX1 上運行 Linux For Tegra (L4T) R24.2.1、CUDA 8.0、OpenCV4Tegra 2.4.13、ZED SDK 1.2，雙鏡頭 720P 解析度下，同時使用 GPU 計算深度圖，CPU 處理其他後續部份，其運算速度為 15~30 FPS，其運作成效已拍攝至 Demo 影片中。目前可偵測車輛前的障礙物及測距，並且追蹤前方行車路徑的物體，以提供安全的行車建議動作。

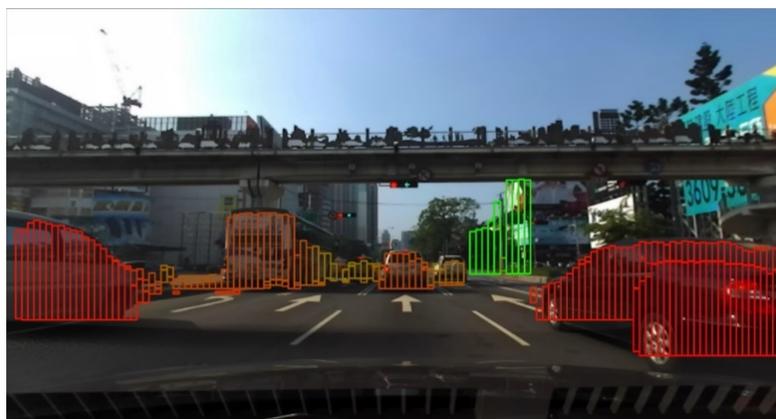


圖 4-1、障礙物偵測



圖 4-2、前方車輛接近
而本車速度過快
顯示 Brake 警示



圖 4-3、前方有車切入
顯示 Warning 警示

五、 創意設計構想

1. 思考如何達到更安全的行車環境，我們認為單純的主動煞車仍有不足，應進一步感知車輛環境，追蹤及分析來提供最佳的安全行車路徑。
2. 提出探討並整合多項技術的安全行車解決方案。
3. 結合雙鏡頭深度圖以及單鏡頭測距模型兩種不同方式，達到強健度高、效果佳的障礙物偵測。
4. 使用雙鏡頭及 Jetson TX1 開發板，以較低的成本得到與 LiDAR 相似的效果。

六、 結論與未來展望

在都市中交通紛亂，各種大小型車、機車、腳踏車與行人相互夾雜，又或者在公路上高速行駛，駕駛需全程注意車輛周圍情況，但人無法像機器一般，不可能隨時保持良好的精神狀況與注意力，而我們的目標就是盡可能地讓乘客與車輛保持安全。

本專題結合多項技術，提供一個有效的解決方案，利用雙鏡頭深度感測來達到障礙物偵測與距離定位，即時偵測前方是否危險並警示，我們使用成本較低的光學鏡頭，並提供足夠精確的環境建模。在雙鏡頭測距方面平均誤差為 3.1%，亦即精準度為 96.9%；而障礙物偵測準確率則可達到 93.6%，並提供安全的行車建議動作。

此次是我們踏出的第一步，而最困難的是找到合適的解決方案，因此先只偵測車輛前向的部分，待未來系統架構完善後，再將偵測範圍延伸到車輛四周所有區域，但不可避免的會遇到效能瓶頸的問題，Jetson TX1 目前系統運算單個雙鏡頭 720P 的深度圖平均有 25 FPS，而要達到環車至少需要四個方向，也就是四個雙鏡頭深度圖，因此無法達到實時運作。但現在無人自動駕駛技術正是目前的研究熱點，相信不久的未來會有更合適的硬體或者更好的演算法來達成。

七、 Demo 影片

網址: <https://youtu.be/yCawzKU7R9A>

參考文獻

- [1] "Cameras or Lasers?", Templetons.com, 2013. [Online]. Available: <http://www.templetons.com/brad/robocars/cameras-lasers.html>. [Accessed: 20-Jan- 2017].
- [2] Q. Zhang and X. Mei, "Study on the Method of Measuring the Preceding Vehicle Distance Based on Trilinear Method", 2010 Second International Conference on Computer Modeling and Simulation, 2010.
- [3] Z. Shen and X. Huang, "Monocular vision distance detection algorithm based on data regression modeling" 2007 CEA, 43(24), pp. 15-18, 2007.
- [4] S. Birchfield and C. Tomasi, "Depth Discontinuities by Pixel-to-Pixel Stereo", International Journal of Computer Vision, vol. 35, no. 3, pp. 269-293, 1999.
- [5] S. Birchfield and C. Tomasi, "A pixel dissimilarity measure that is insensitive to image sampling", IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 20, no. 4, pp. 401-406, 1998.
- [6] 陳育菘、廖育昇、徐子建, "立體視覺特徵點比對演算法分析與實現", 中華民國第十六屆車輛工程學術研討會, 2011.
- [7] H. Hirschmuller and D. Scharstein, "Evaluation of Cost Functions for Stereo Matching", 2007 IEEE Conference on Computer Vision and Pattern Recognition, 2007.
- [8] Z. Zhang, "A flexible new technique for camera calibration", IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 22, no. 11, pp. 1330-1334, 2000.
- [9] J. McDonald, "Detecting and tracking road markings using the Hough transform", in Proc. Irish Machine Vision and Image Processing Conf., pp. 1-9, 2001.
- [10] K. Kluge, "Extracting road curvature and orientation from image edge points without perceptual grouping into features", Proceedings of the Intelligent Vehicles '94 Symposium, 1994.
- [11] Y. Guo, Z. Bai, Y. Li and Y. Liu, "Genetic Algorithm and Region Growing Based Road Detection in SAR Images", Third International Conference on Natural Computation (ICNC 2007), 2007.
- [12] Q. Li, N. Zheng and H. Cheng, "Springrobot: A Prototype Autonomous Vehicle and Its Algorithms for Lane Detection", IEEE Transactions on Intelligent Transportation Systems, vol. 5, no. 4, pp. 300-308, 2004.
- [13] M. Bertozzi, A. Broggi, M. Cellario, A. Fascioli, P. Lombardi and M. Porta, "Artificial vision in road vehicles", Proceedings of the IEEE, vol. 90, no. 7, pp. 1258-1271, 2002.

安全是唯一的路

- [14] J. GORDON, "Dashboard camera footage captures crash of Tesla autopilot", Mail Online, 2017. [Online]. Available: <http://www.dailymail.co.uk/news/article-4276810/Dashboard-camera-footage-captures-crash-Tesla-autopilot.html>. [Accessed: 23- Mar- 2017].