

國立交通大學

電信工程研究所

碩士論文

單視域之遞迴式深度估測補償

Monoscopic Depth Map Compensation Using Recursive
Method

研究生：羅至中

指導教授：張文鐘 博士

中華民國 一百零一年 九月

單視域之遞迴式深度估測補償

Monoscopic Depth Map Compensation Using Recursive Method

研究生：羅至中

Student：Chih-Chung Lo

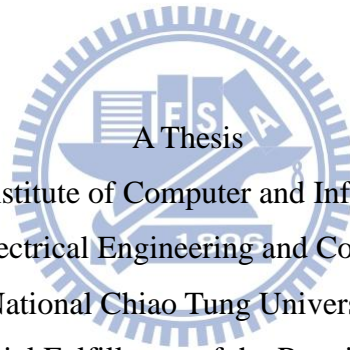
指導教授：張文鐘 博士

Advisor：Dr. Wen-Thong Chang

國立交通大學

電信工程研究所

碩士論文



Submitted to Institute of Computer and Information Science
College of Electrical Engineering and Computer Science

National Chiao Tung University

in partial Fulfillment of the Requirements

for the Degree of

Master

in

Computer and Information Science

September 2012

Hsinchu, Taiwan, Republic of China

中華民國 一百零一年 九月

單視域之遞迴式深度估測補償

研究生：羅至中

指導教授：張文鐘 博士

國立交通大學

電信工程研究所

中文摘要

本論文提出了一個針對單視域影像的深度估測方法，為了能從單張影像中獲得深度資訊，我們使用了一些深度線索來幫助估測，例如影像中的花紋等。此外，移動向量這個深度線索常在連續兩張影像中使用，大部分對於單視域影像的深度估測都是將這些深度線索做線性加成而得到初始的深度估測圖，然而將各個深度線索線性加成所使用的權重分配並沒有一套有系統的逼近方法，因此我們提出了一個利用遞迴式的方法來做深度估測的補償。在我們利用線性加成得到初步估測的深度圖之後，便可利用 DIBR(Depth Image Based Rendering)的技術合成出左右眼影像對，接著我們把這左右眼影像對利用立體視覺測量法計算深度圖，在本論文使用的立體視覺匹配演算法是 Dynamic Programming(DP)，由於 DP 考慮到了影像像素點之間的關連性，這是初始深度圖估測所沒有的，因此可以得到一個較平滑的深度圖也解決了前景物體內部的深度不正確區塊，在獲得這個新的深度圖之後，我們利用它來補償初步估測的深度圖以得到更好的深度估測效果。實驗中證實本論文提出的方法可有效改進原始的估測深度圖，此外對合成左右眼影像對並利用 3D 立體顯示器觀賞時也將對人眼更為舒適。

關鍵字：單視域影像，深度圖，深度線索，立體視覺

Monoscopic Depth Map Compensation Using Recursive Method

Student: Chih-Chung Lo

Advisor: Dr. Wen-Thong Chang

Institute of Communications Engineering,

National Chiao Tung University

Hsinchu, Taiwan

Abstract

This paper presents a method for depth map estimation from monoscopic video. To estimate depth from single image, some depth cues such as texture is used for initial guess. However motion vectors between current frame and previous frame is more often used. Most methods combine these cues linearly to get initial depth map estimation. However no systematic approach is available for weight estimation. Thus a recursive method is proposed for depth map estimation. After we got initial depth map, we can compute a left/right image pair by DIBR(depth image based rendering). Then this image pair can be used to compute disparity map by using DP(dynamic programming) based on stereo vision. DP takes the depth relation between pixels into account while the initial depth map estimation does not. So this disparity map get smoother depth estimation and fix the problem inside foreground objects. With this new depth map, recursive depth map compensation is used to derive a better depth estimation. Experiment results show our method can improve the original depth map and further more derive the stereo images with better visual comfort.

Keywords : monoscopic video, depth map, depth cues, stereo vision.

誌謝

時光荏苒，兩年的時光匆匆流逝，碩士生涯至此也算完美結束，對於我的指導教授張文鐘博士，心中盡是感謝，不論在學業以及生活上都得到老師的指導與教誨，這些在往後的日子裡都將銘記在心，同時也要感謝口試委員莊仁輝教授、王聖智教授、陳永昇教授，各位口試委員都以精闢的角度和專業知識來給與論文上的建議，使我知道自己不足的地方，並得以改進以及修正使論文更加完整。

感謝 Lab821 的所有成員陪伴我兩年的成長，博班學長家豪；98 級學長姊維哲、耀駿、舒評、信妤；99 級同學任恆帶領我見識交大排球隊的熱血，也在我運動受傷時給予幫助與建議、奕廷總是能想出許多有趣話題在氣氛低迷時給予緩和，也因為他認識到了許多交大球友、詩倩則是會時常分享美食資訊給大家，在論文衝刺階段也最常一起待實驗室的好夥伴。另外要感謝室友超哥讓我開始對打羽球產生興趣，也能固定在每周都有運動的一天、GY 從大學就跟你當同學到現在，你無厘頭的發言到現在還是令我印象深刻，也因為你才能夠有固定的籃球以及遊戲隊友，得到適當的壓力紓解、兼原常常來實驗室與我們分享生活點滴，也在最後幾個月不吝惜的跟我平分冷氣支出。還要感謝時常幫我加油打氣的朋友們，有你們關心的話語才能讓我走到現在，也謝謝瑜珈老師在這一年的教學，讓我學到許多紓壓放鬆的方式。

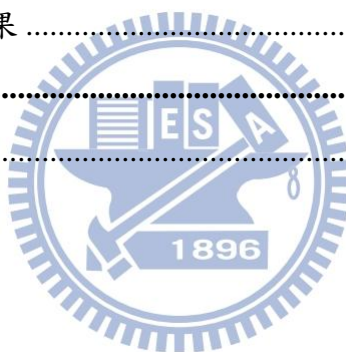
最後也是最重要的，我要感謝我的家人，不論在何時總是成為我背後的支柱，雖然忙碌時不常回家，但你們也能體諒我而不責怪，姊姊也時常關心我的生活狀況，從以前到現在各種需求的物品也都會不吝嗇的贈與，這對還沒有經濟能力的我真是莫大的幫助，而對於我的想法也都給與滿滿的支持與建議；還要感謝交大土地公的保佑，讓我這兩年過的平安健康。靠著大家的支持才能走到這一步，成就現在的我，由衷的希望大家在未來都能健康順利，平平安安。

誌於 2012.夏 新竹。交大
至中

目錄

中文摘要	i
Abstract.....	ii
誌謝	iii
表目錄	vi
圖目錄	vii
第 1 章 緒論.....	1
1.1 研究動機與方向	1
1.2 章節概要	3
第 2 章 3D 量測技術.....	4
2.1 TOF.....	4
2.2 結構光投影	6
2.2.1 XBOX360 Kinect.....	6
2.2.2 Gocator	11
2.3 手機三角測距	13
2.3.1 AOA 法	13
2.3.2 RSS 法.....	13
2.3.3 TOA 法.....	14
2.3.4 TDOA 法.....	15
第 3 章 立體視覺.....	18
3.1 相機模型	18
3.2 立體視覺成像	20
3.3 極線幾何	22
3.4 匹配問題	23
3.4.1 DP.....	24
3.4.2 限制條件	24
3.4.3 花費函數.....	26
第 4 章 單視域立體視訊轉換技術.....	29

4.1	深度線索	29
4.1.1	動態視差(motion parallax).....	29
4.1.2	材質漸層(texture gradient).....	31
4.1.3	大氣透視(atmospheric perspective)	31
4.2	深度圖修正	32
4.2.1	相鄰單元標記法(connected component labeling)	32
4.2.2	空洞填補(hole filling).....	34
4.2.3	前景背景深度計算	35
4.2.4	左右影像對合成法	37
4.3	3D 立體顯示器	40
第 5 章	實驗模擬與結果分析.....	42
5.1	遞迴式補償架構	42
5.2	參數設置與模擬結果	43
第 6 章	結論與未來展望.....	60
參考文獻	61



表目錄

表 4.1	相鄰編號表範例。	33
表 5.1	改變量百分比與遞迴次數關係。	53
表 5.2	不同深度線索得到的初步深度圖與 ground truth 的相似度比較。	54



圖目錄

圖 2.1	Time-of-flight 原理。	5
圖 2.2	結構光測距原理示意圖。	6
圖 2.3	XBOX360 Kinect 外觀。	7
圖 2.4	Kinect 紅外線光點移動變化實驗。	8
圖 2.5	Kinect 紅外線光點移動標記實驗。	9
圖 2.6	Kinect 紅外線光點變化示意圖。	10
圖 2.7	Kinect 投影的紅外線光點圖案。	10
圖 2.8	azt.tm 拼湊出的 Kinect 完整投影圖案。	11
圖 2.9	Gocator 2300 系列。	12
圖 2.10	雷射光點反射成像。	12
圖 2.11	使用三個基地台做 TOA 定位示意圖。	14
圖 3.1	透視投影。	18
圖 3.2	坐標系統轉換示意圖。	20
圖 3.3	立體視覺成像。	21
圖 3.4	極線幾何。	22
圖 3.5	校正(Rectification)。	23
圖 3.6	左右影像匹配路徑示意圖。	24
圖 3.7	限制 disparity 運算範圍的 DP。	25
圖 3.8	左右成像點關係。	26
圖 3.9	DP 計算程序。	28
圖 4.1	Block matching。	30
圖 4.2	Sobel operator。	31
圖 4.3	大氣透視。	32
圖 4.4	前景相鄰單元標記。	34
圖 4.5	空洞填補與前景移除。	34
圖 4.6	空洞內插深度值。	35
圖 4.7	高斯濾波器。	36
圖 4.8	使用高斯平化過後的前景深度圖。	36

圖 4.9	背景深度圖。	37
圖 4.10	修正過後的初步深度圖。	37
圖 4.11	影像合成規則。	39
圖 4.12	HTC EVO 3D 手機外觀。	40
圖 4.13	視差屏障原理。	40
圖 5.1	遞迴式深度補償架構。	42
圖 5.2	深度線索計算結果。	44
圖 5.3	初步估測的深度圖。	45
圖 5.4	前景區域修正結果。	45
圖 5.5	內插過後的前景深度圖。	45
圖 5.6	最後估測深度圖。	46
圖 5.7	左右影像對合成與 disparity map 計算結果。	46
圖 5.8	深度圖補償。	47
圖 5.9	Scanline 150 與 Scanline 230 的深度分布情況。	47
圖 5.10	違反單調次序性。	48
圖 5.11	違反唯一性。	48
圖 5.12	移除限制條件下的 DP 運算結果。	49
圖 5.13	遞迴補償深度圖。	50
圖 5.14	前景物體內部區域補償。	51
圖 5.15	物體邊界破壞。	51
圖 5.16	前景、邊界、背景分割。	52
圖 5.17	輸入深度圖與 disparity map 的差異。	52
圖 5.18	隨著遞迴次數改變的差異圖。	52
圖 5.19	深度區間劃分。	54
圖 5.20	遞迴深度圖與文獻深度圖的比較結果。	56
圖 5.21	“Ballet” 測試影像模擬結果。	58
圖 5.22	“Akko and Kayo” 測試影像模擬結果。	59

1.1 研究動機與方向

近年來 3D 電影不斷的充斥著消費市場，從最早的阿凡達到現在幾乎每部電影都標榜著 3D 觀賞效果，目前市面上的 3D 立體拍攝系統主要分為陣列式攝影機與深度攝影機，前者使用多架平行擺設的攝影機同步拍攝，後者使用一般傳統的攝影機加上一台深度攝影機做深度的計算，然而目前深度攝影機尚未普及，因此難為 3D 內容找出一個好的解決方案。雖然現在已經擁有 3D 畫面的拍攝技術與裝置，但對於一般使用者來說還是最希望看到貼近日常生活的一些照片影片等，這些都仍然是傳統的 2D 視訊，因此為了解決此問題所需要的便是一套 2D 轉 3D 技術，也就是必須得到 2D 影像的深度資訊，目前的深度估測方式主要可分為人工指派、半自動深度估測法與全自動深度估測法三類，人工指派是使用者依據數個主觀的 3D 線索以及拍攝技巧制定規則，對每張畫面去做手動的分析與評估，雖然得到的品質最好，但這是最耗資源成本的方式，因此本論文的方向是朝著全自動深度估測的方式去實行，也就是透過計算單視域各種深度線索得到場景的深度圖，雖然全自動深度估測法至今仍未找到一套非常好的轉換技術，但我們的目標是期望對估測的深度圖做一個補償改善，以得到較好的 3D 內容品質。

單視域影像的深度估測仰賴於深度線索資訊，在文獻中有使用單一深度線索做為深度估測的依據[1][2]，有使用影像分割的方法來輔助場景深度估測[3]，也有混合使用的方法加強深度估測的正確性[4][5]，由於僅憑單一深度線索就決定場景的深度分布情況並不太恰當，因此本論文使用多種深度線索來綜合輔助判斷場景深度。雖然使用多種深度線索可改善深度的估測，但是影像內容千變萬化，要找出一組適用於所有場景的深度線索組合是不太容易的，所以本論文目的並不在於去找出這麼一組最佳解，而是在於如何利用已估測出的深度圖，將其不完整性以其他方法做為彌補，因此本論文提出一種遞迴式的深度估測補償方法，結合了單視域影像深度估測與立體視覺(stereo vision)的深度計算方式，經由不斷的循環疊代來改善深度圖的估測，由於從單視域影像中估測深度圖所擁有的資訊有限，並且此估測方法是以個別區塊去猜測深度，沒有考慮到 pixel 間的深度關連性，而我們解決此方法的手段就是利用立體視覺的 DP 演算法，DP 將 pixel 間的關連性做為深度估測的一項因素，以全域最佳化原則重新計算出一張深度圖並用來改

善原本前景物體內部所產生的深度不連續性。

在立體視覺中最困難的部分即是要解決匹配問題，一旦匹配問題解決，利用影像間的對應關係即可求出深度。立體視覺的匹配演算法可大致分成兩類，一類是區域(local)的演算法，另一類是全域(global)的演算法[6]。區域演算法通常根據 window 範圍內的 pixel 亮度值來決定對應的 pixel 點，利用 MAD(mean absolute difference)或 MSE(mean squared error)等匹配函數決定匹配的花費(matching cost)，接著將 window 內的花費做加總找出最小值，最小值即表示此 pixel 在另一畫面中最適合的對應點，算是一種區域的 winner-take-all(WTA)的演算法，此演算法沒有對 disparity 做出一些限制條件，較容易匹配到錯誤的對應點，因此 disparity map 將會有許多不平滑的現象發生。而全域演算法如 DP(dynamic programming)、SO(scanline optimization)、GC(graph cuts)、SA(simulated annealing)則在 disparity map 平滑上有所限制，全域演算法要求的是一個全域的最佳化，也就是說它要最小化的是一個全域的花費函數，disparity 的 smooth 與否會影響此花費函數，若是兩張圖片對應點間只有水平方向的位移(經過校正的圖片)，那麼全域的範圍指的是每條在處理的掃描線(scanline)即圖片中的每個列(row)。由於區域的匹配演算法會造成一些不正確的深度不連續性，這些深度不連續對於估測的深度圖以及合成立體影像對時都將造成不好的影響，為此本論文使用全域的匹配演算法，而在眾多的全域演算法中，DP 是最符合直覺且易懂的演算法，加上其運算花費時間短的特性，因此 DP 成為我們深度計算演算法的選擇。

為了得到較正確的單視域深度估測，本論文結合了深度線索估測、立體影像對合成法與立體視覺法形成一有系統的深度估測補償架構，在深度線索估測中我們使用的深度線索有動態視差、材質漸層與大氣透視，將這些深度線索以線性加成的方式計算出初步的深度圖。在立體影像對合成中，本論文使用的是所謂 DIBR(Depth Image Based Rendering)的技術，也就是根據估測出的深度圖將原始單視域影像偏移(shift)而獲得有視差的左右眼影像，由於垂直方向上的偏移將對人眼造成不適的影響，因此本論文的左右影像對是將原影像進行水平方向上的位移得到。而在立體視覺法中，我們使用的匹配演算法是 DP，由於立體影像對的合成是進行水平方向上的偏移，因此我們不需使用影像校正(image rectification)的技術做匹配前的處理，可直接對影像的每個 scanline 做獨立的匹配運算。我們的深度補償架構首先利用各深度線索估測影像的深度資訊，並利用線性加成的方式合併得到初步估測深度圖，有了初步的深度圖即可利用 DIBR 的立體影像合成技術得到左右眼影像對，接著將此左右眼影像對利用立體視覺的方式重新計算出一張

disparity map，最後以固定的權重比例將初步估測的深度圖與立體視覺計算出的 disparity map 做線性合併補償，此合併過後的補償深度圖將做為下一次遞迴運算的輸入深度圖使用，藉由不斷的遞迴運算將對深度圖的正確性得到改善的效果，此外，改善過後的深度圖也將合成出使人眼觀賞較為舒適的立體影像對，本論文利用 HTC EVO 3D 這款智慧型手機做為觀賞立體影像之用。

1.2 章節概要

本論文在第二章將會介紹各種 3D 測量技術，包含 TOF、結構光投影以及手機定位技術。第三章則詳細介紹立體視覺，從相機的成像原理、透視投影到相機內外部參數矩陣，以及相機間的極線幾何關係，最後說明困難的匹配問題以及介紹本論文使用的 DP 演算法流程及其相關的限制條件。第四章先簡介本論文使用的各項深度線索，接著說明如何對深度圖做適當的修正與處理，包括相鄰單元標記、空洞填補等，最後是前景背景的深度計算方式以及合成立體影像對的方法。第五章為本論文的系統架構以及模擬的實驗結果。最後第六章為結論以及未來展望。



第2章 3D 量測技術

在現實生活中我們所看到的東西或照片內容皆為 2D 圖像，如何利用已知的 2D 資訊來推測失去的第三維資訊(深度資訊)將是接下來要探討的 3D 量測技術。3D 測量技術根據光源性質的不同分為主動式(active)與被動式(passive)兩類，前者如 TOF(Time-Of-Flight)與 Structured light 都是藉由投射光線至待測物體上，根據物體反射光線的訊息來測量深度資訊，後者為運用立體視覺(Stereo Vision)的技術，不主動使用光線照射物體，而是利用多台相機根據不同的角度拍攝物體並根據三角測量法來求得物體的深度，立體視覺法將在下一章節做詳細說明。

2.1 TOF

TOF(Time-Of-Flight)是一種 3D 測距的技術，其測量距離的方式是利用已知的光速來換算物體距離相機的遠近，在半導體發展快速的這個時代，TOF 相機的深度精密度已經可以達到約 1cm 左右，而影像的解析度大多為 320*240 pixels，其運算速度可高達 100 frame/sec，非常適於 real-time 的應用[7]。

3DV Systems 公司推出的 ZCam 就是利用 TOF 這項技術[8][9]，ZCam 的發射器投射近紅外線光照射物體表面，計算出的每個 pixel 的深度值都對應到照射物體上的某一點與相機的距離。ZCam 屬於一種閘門式(range-gated)的測距系統，利用高速的驅動電路來控制產生光脈衝以及閘門開關，因此需要調變器(modulator)的技術使用，使用閘門式的測量技術最關鍵的在於偵測器的調變器必須運作的非常快(<1ns)且必須與發射器的控制同步，如此才能準確的測量光反射的時間。ZCam 的深度解析度是 1-2cm，適合的量測距離為 0.5-2.5m，由其解析度(假設為 1.5cm)與光速($3 \times 10^8 \text{ m/s}$)可計算如下式：

$$\frac{2 \times 1.5 \times 10^{-2}(\text{m})}{3 \times 10^8(\text{m/s})} = 100 \times 10^{-12}(\text{s}) = 100 (\text{ps}) \quad (1)$$

表示 ZCam 的調變運作快到可分辨 100ps 的時間差，換算為頻率約為 10 GHz，ps 為 picosencond(= $10^{-12}(\text{s})$)，時鐘週期(clock cycle)愈小對於量測深度資訊的精確性將會提高。

TOF 相機利用高速的切換器(switch)來控制閘門，產生的光脈衝經由透鏡聚集在感

光平面上，原理如下圖 2.1：

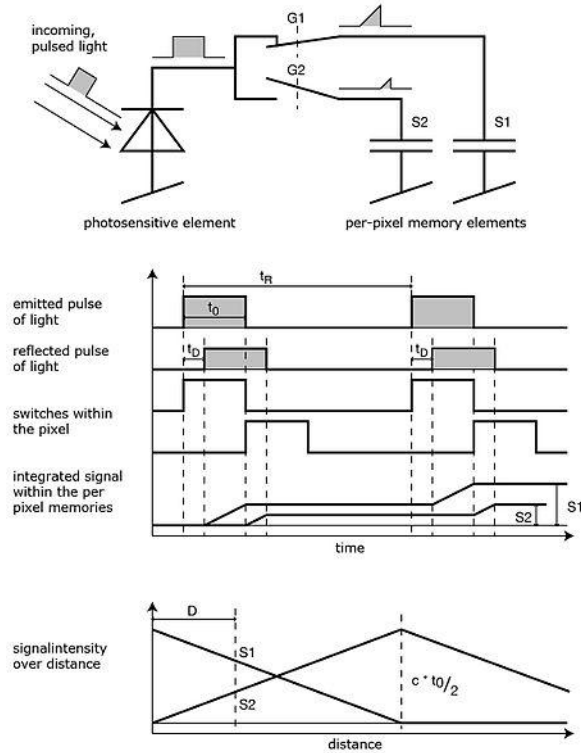


圖 2.1 Time-of-flight 原理。

如圖 2.1 上所示，每個 pixel 都由感光元件組成，將入射光轉換成電流，經由 switch G1、G2 流至儲存原件 S1、S2，圖 2.1 中標示了光脈衝的產生與閘門開關的時間，G1 switch 的開關與產生光脈衝的控制是同時的，開關時間等於光脈衝的寬度，而 G2 switch 則延遲了一個光脈衝的寬度才開啟，因此在 G1、G2 開啟的時間內分別有不同量的光線進入。而相機可以測量的最遠距離(D_{max})是由光脈衝的寬度(t_0)決定，如圖 2.1 下， D_{max} 計算如下式：

$$D_{max} = \frac{1}{2} c t_0 \quad (2)$$

其中 c 為光速。

ZCam 的最大量測距離為 2.5m，由上式可計算得知 ZCam 的光脈衝寬度約為

$$t_0 = \frac{2 \times 2.5}{3 \times 10^8} = 1.67 \times 10^{-8} \text{ (sec)} \quad (3)$$

在已知產生的光脈衝的寬度(t_0)之下，物體與相機的距離(D)可經由下式計算出來：

$$D = \frac{1}{2} c t_0 \frac{S_2}{S_1 + S_2} \quad (4)$$

2.2 結構光投影

結構光(Structured light)測量技術使用一台投影機(projector)與一台照相機來測量物體深度，利用投影機投影一個已知的圖案(pattern)於場景中，接著用相機將被投影的場景照片拍下，如果場景中只有一個平坦的面而無任何表面起伏，那麼照片拍下的畫面則會與投影的圖案相同，反之，如果場景中有非平面的物體，物體表面的幾何形狀會扭曲投影的光線，根據照片中光線的扭曲程度與投影的圖案做比較可推算出物體的深度資訊 [10]。

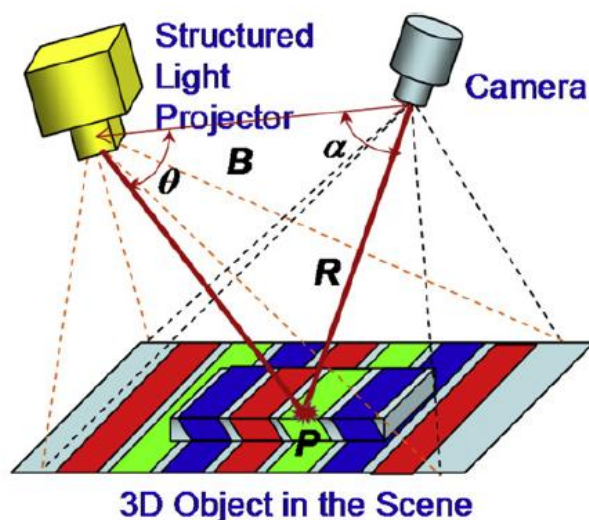


圖 2.2 結構光測距原理示意圖。

上圖 2.2 顯示了投影的結構光 pattern 被場景中非平坦物體表面影響的扭曲情形，其中 P 為畫面中的一點，B 為投影機與相機間的距離(baseline)，R 為 P 點到相機間的距離(深度)， θ 為 P 點到投影機與 baseline(B)的夾角， α 為 P 點到相機與 baseline(B)的夾角，由三角形定理可推算深度 R 如下式：

$$R = B \frac{\sin(\theta)}{\sin(\alpha + \theta)} \quad (5)$$

2.2.1 XBOX360 Kinect

Kinect 是由微軟開發應用於 XBOX360 主機的周邊設備，他不需要任何控制器來做操作，主打「身體就是控制器」的應用服務。下圖 2.3 為 Kinect 的外觀架構

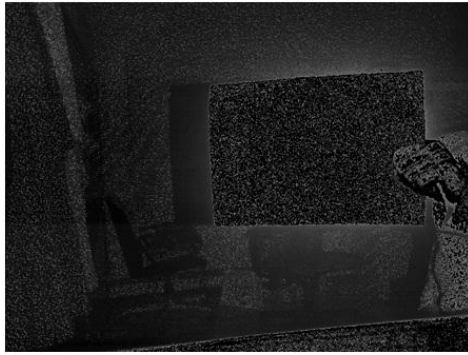


圖 2.3 XBOX360 Kinect 外觀。

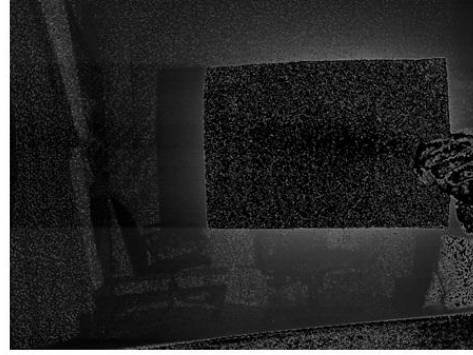
由正面看來有三個光學鏡頭，由左到右分別是紅外線光投影機、RGB 相機與紅外線光攝影機，紅外線投影機與紅外線攝影機的用途是計算景物的深度資訊，而 RGB 相機則只用來拍攝一般的照片。

Kinect 的深度計算方式外界有各種不同的說法，但最後經由微軟官方公布 Kinect 是採用微軟收購的以色列公司 PrimeSense 所開發的光編碼技術(Light Coding)來做深度資訊的計算。光編碼(Light Coding)實際上也是結構光技術的一種，利用投影已知的近紅外線光散斑圖案來計算空間中物體的深度資訊，此散斑圖利用偽隨機的方式布置光點[11]，實際上 Kinect 在出廠時就已經經過校正而知道每一個投影的紅外線點將會成像在一指定距離(校正距離)處的哪個地方，也就是說每台 Kinect 內部存放著一張已知距離的紅外線點投影畫面[12]，當空間中任何物體出現在比校正距離近的地方時，物體上的紅外線光點將會往一個方向移動，反之若物體是存在於比校正距離遠的地方時，該物體上的紅外線光點將會往另一個方向移動[13]。由於 Kinect 是已經經過校正的，所以它可以預先知道每個紅外線光點應該出現的位置，當空間中存在著不只有校正距離的物體時，Kinect 便可利用物體上紅外線光點偏移的距離來估算物體距離 Kinect 的深度。但若是畫面中有被遮蔽的物體，其產生出的陰影區域是紅外線光點無法照射到的，這些區域 Kinect 則無法得知其深度，另外若有反射率太高或太低的物體存在(鏡子或玻璃等)也會造成該處的深度估測錯誤。

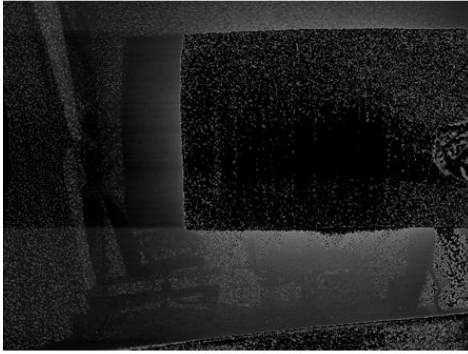
為了驗證上述[13]說法，我們實驗用手將一張 A4 大小的紙由遠而近移動至 Kinect 鏡頭前方，觀察到紙上的光點變化如下圖 2.4：



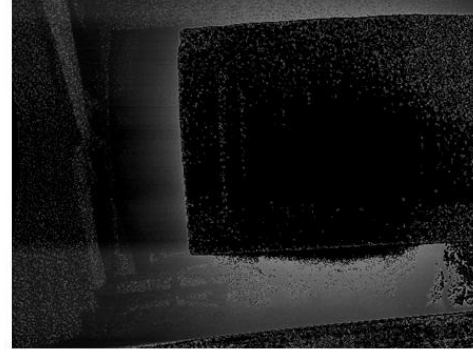
Frame 0



Frame 46



Frame 58



Frame 75

圖 2.4 Kinect 紅外線光點移動變化實驗。

由圖 2.4 的幾張 frame 變化可觀察到，在 A4 紙上的 Kinect 紅外線光點似乎是由紙上的一點為中心，當 A4 紙朝 Kinect 接近時，紙上的光點會以此中心朝四周做放射狀的移動，反之，遠離 Kinect 時則會朝此中心移動。因為紅外線光點數量非常多，且有些光點會時而閃爍不定，用肉眼不易分辨其移動，因此我們從 frame 80 開始手動將較分散易分辨的紅外線光點群用紅黃藍綠四種不同顏色的圓圈框起來，如下圖 2.5：

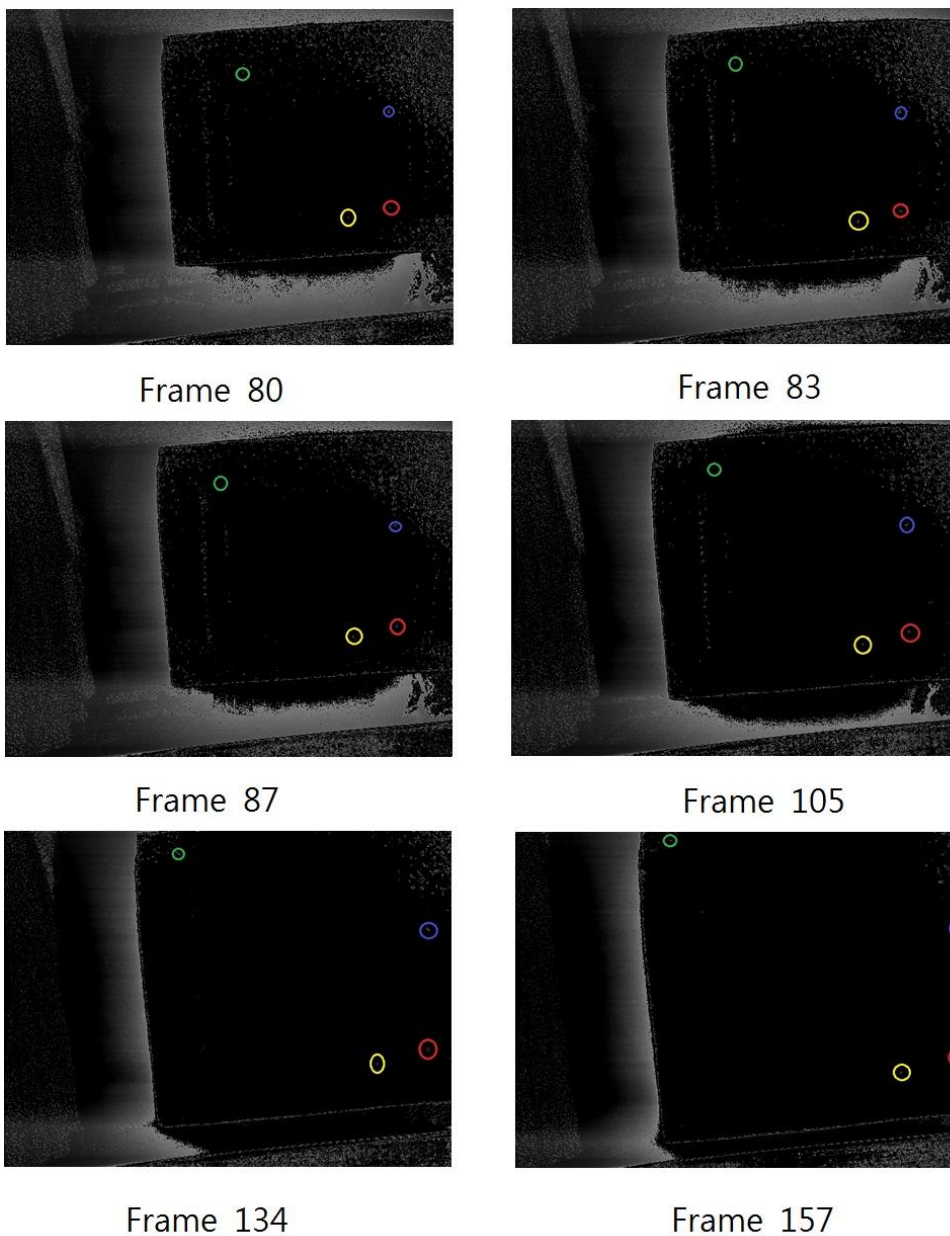


圖 2.5 Kinect 紅外線光點移動標記實驗。

藉由觀察圖 2.5 這四個圓圈的移動可以確定當一物體距離 Kinect 變近時，物體上的紅外線光點是如我們所推測的以一中心點向外放射狀移動。據觀察，光點的放射狀中心位於畫面中心點偏右的位置，恰巧與 Kinect 兩個相機擺放的位置有關，紅外線影像光點的放射狀移動中心似乎就是紅外線投影機的中心投射線在畫面中的成像位置。如下圖 2.6：

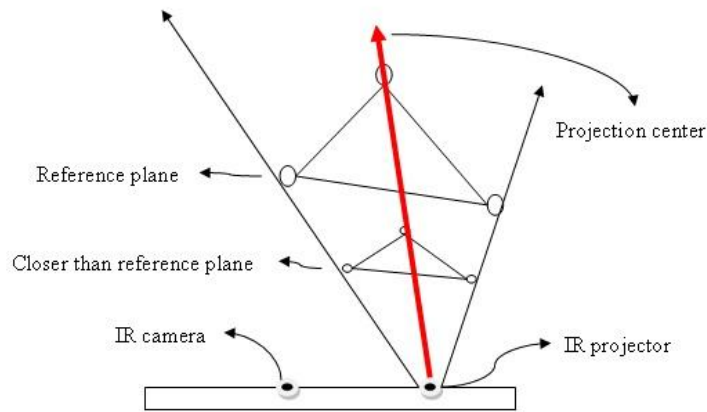


圖 2.6 Kinect 紅外線光點變化示意圖。

由於 Kinect 的紅外線投影機是投射出一束狀的紅外線光點圖，由圖 2.6 可看出當物體存在於比校正平面近的位置時，我們所觀察到的光點相對位置會較小，也就是往投射中心移動，這與我們在 IR 相機中觀察到的情況不同，推測其理由是，當物體接近 Kinect 時，由於相機的成像原理會造成物體在影像中也是放大的情況，而此相機的放大比例大於投影光點圖的縮小比例，因此在物體上的紅外線光點在紅外線影像中看起來相對是朝遠離投射中心的方向移動。



圖 2.7 Kinect 投影的紅外線光點圖案。

上圖 2.7 是 Kinect 投影出來的紅外線光點圖案，圖中可看出投影圖案大致分割成 9 塊矩形的區域，每塊矩形中間都有一個特別亮的點。根據 azt.tm 在網路上的分析[14]，他將 Kinect 的投影圖案投射至牆上，並將此圖案另外用紅外線相機拍攝下來，利用 photoshop

將拍攝到的圖片做一些裁剪與減少光點的扭曲，經由實驗他發現這些光點在水平方向上每 211 個點會重複，也剛好是整張圖案寬的 $\frac{1}{3}$ 處，在垂直方向上每 165 個點會重複，也是在圖片高 $\frac{1}{3}$ 處，這說明了 Kinect 的點狀投影圖案是由 211*165 大小的子圖案 (subpattern)重覆排列成 3*3 九宮格大小的完整投影圖案，總共 633*495 個點也相近於其規格的 VGA 解析度(640*480)，最後他拼湊出了完整的投影圖案，如下圖 2.8：



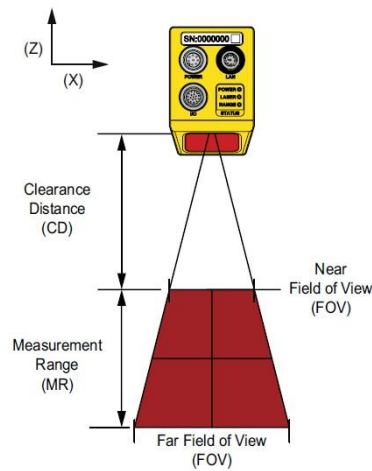
圖 2.8 azt.tm 拼湊出的 Kinect 完整投影圖案。

經過他的分析，他把子圖案的亮點做計算，得到在平均每 3*3 的區域內就會有一個是亮點。他拼湊出的圖案外側其實還有一些點，但那些都是暗的點，此外沒有 9 個亮點是連接在一起的。此投影圖案的設計是可以上下顛倒而不改變的，將子點狀圖排列成九宮格形狀對計算深度的演算法似乎沒有什麼益處，反之在計算 disparity 時必須將範圍限制在 $\frac{1}{3}$ 的投影圖案內。

2.2.2 Gocator

Gocator2300 感測器是由 LMI Technology 公司所開發，他將所有掃描測距系統做在一個測量器裡，無需額外的使用器材做輔助，且可透過網路瀏覽器做設定與控制，擁有百萬像素的解析度與量測至微米等級的精密度，常用於自動儀器的檢測及工業製程(輪胎、木材切割等)的輔助工具。下圖 2.9 是 Gocator2300 系列的系統規格：

Gocator 2300 Family



Model	2330	2340	2350	2370	2380
Data Points / Profile	1280	1280	1280	1280	1280
Linearity Z (% of MR)	0.01	0.01	0.01	0.01	0.01
Resolution Z (mm)	0.006 - 0.014	0.013 - 0.037	0.019 - 0.060	0.053 - 0.173	0.071 - 0.413
Resolution X (mm)	0.044 - 0.075	0.095 - 0.170	0.150 - 0.300	0.275 - 0.550	0.375 - 1.100
Clearance Distance (CD) (mm)	90	190	300	400	350
Measurement Range (MR) (mm)	80	210	400	500	800
Field of View (FOV) (mm)	47 - 85	96 - 194	158 - 365	308 - 687	390 - 1260
Standard Laser Class	2M	3R	3R	3B	3B
Available Laser Classes*	3R, 3B	2M, 3B	2M, 3B	2M, 3R	2M, 3R
Dimensions (mm)	49x75x142	49x75x197	49x75x272	49x75x272	49x75x272
Weight (kg)	1	1.15	1.45	1.45	1.45

*Pricing will vary

All 2300 Family Models	
Scan Rate	Approx. 170Hz to 5000 Hz
Interface	Gigabit Ethernet
Inputs	Differential Encoder, Laser Safety Enable, Trigger
Outputs	2x Digital output, RS-485 Serial (115 kbaud), 1x Analog Output (4 - 20 mA)
Input Voltage (Power)	+24 to +48 VDC (13 Watts); RIPPLE +/- 10%
Housing	Gasketed aluminum enclosure, IP67
Operating Temp.	0 to 50°C
Storage Temp.	-30 to 70°C

NOTE: Specifications stated reflect standard laser classes, values will vary for other laser classes.

圖 2.9 Gocator 2300 系列。

圖 2.9 列出了 Gocator2300 系列的詳細資料，觀看 Gocator2330 的深度解析度為 0.006-0.014(mm)，表示 Gocator 感測器的確可達到微米(μm)等級的量測精確度，而根據不同的使用性質，不同系列的 Gocator 感測器也使用不同的雷射光(2M、3R、3B)來達到最佳的測量效果。

根據官方說法，Gocator 的 3D 測量技術是雷射光三角測量法(laser triangulation)[15]，LMI Technology 一直以來也都是使用雷射光技術來做 3D 測量，也擁有許多相關專利。下圖 2.10 為利用雷射光點反射成像示意圖

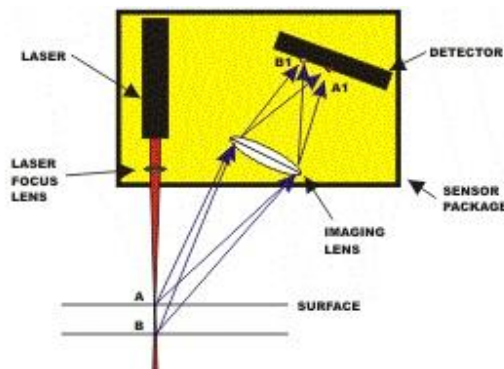


圖 2.10 雷射光點反射成像。

圖 2.10 中若物體表面在 A 的位置，光線反射經過透鏡後會成像在 A1，同理 B 會成像在 B1，因此距離測量器遠近不同的物體將會有不同的成像位置。雷射光三角測量法即

利用雷射光投射位置、物體位置與相機位置形成的三角關係來測量物體的深度。雷射光投射器與相機的位置是已知(baseline)，雷射光投射器在三角形中的角度(θ)也是已知，而雷射光點在相機畫面中的位置可決定相機在三角形中的角度(α)，由這些條件可決定一個三角形並利用式(5)計算物體與相機間的距離。

2.3 手機三角測距

在手持行動裝置中常配備有 GPS 做為定位導航的工具，但若不具備 GPS 功能的手機用戶想要定位，則必須仰賴基地台的無線定位技術，其屬於地面定位系統，利用蜂巢式行動電話系統訊號傳遞時量測到的一些相關參數來做定位，常使用的量測參數包括 AOA(Angle of Arrival)、RSS(Received Signal Strength)、TOA(Time of Arrival)、TDOA(Time Difference of Arrival)等[16][17]。

2.3.1 AOA 法

AOA 定位法是利用接收機的天線陣列測量電磁波的相位差來計算出訊號的入射角度，一旦從兩個基地台得知其訊號入射角後，經由計算這兩個方向線的交點可以得知手機用戶的位置，為此接收機必須使用方向性強的陣列天線才能精確得知訊號入射角。使用此方法定位只需要兩個基地台即可，而兩條直線只有一個交點，不會有定位的模糊性，且不需要與基地台做時間同步的動作。

2.3.2 RSS 法

RSS 是透過接收信號的訊號強度和已知的頻道衰弱模型來估測收發兩端的距離，根據測量周圍多個基地台的距離值即可推算手機的位置，一般透過 3 個基地台就可確定手機位置。首先透過一個基地台的測量值可以將手機位置範圍定在一個以該基地台為中心的圓上，透過第二個基地台的測量值可以將手機位置縮小至兩個定點(圓與圓的相交點)，接著可透過第 3 個以上的測量值來找出最佳解求出手機端位置。使用此方法定位應用上較不方便，因為需要事先知道頻道的衰弱模型，且由於 multi-path 對訊號強度造成的干擾影響，使得其精確度較差。

2.3.3 TOA 法

TOA 定位法與 RSS 類似，差別在於 TOA 量測的不是訊號強度，而是訊號到達的時間，無線訊號傳遞速度接近於光速，因此其與 TOF 相機的測距原理類似，當然也受到相似的精確度限制。TOA 法的前提是手機端(MS)與基地台(BS)必須做好精確的時間同步(許多現在的無線通訊系統標準只能做到讓 BS 間的時間同步)，如此才能準確計算出近似實際的距離，但通常 MS 會有幾個微秒的偏移量，這將造成估計距離上的誤差。

MS 與BS_i的距離可表示如下式：

$$r_i = (t_i - t^0)c \quad (6)$$

其中 c 為光速(3×10^8 m/s), t^0 是基地台發射訊號的時間, t_i 為BS_i訊號到達MS的時間。利用三個測量出的基地台距離半徑 r_1 、 r_2 、 r_3 ，可用下列式子(7)-(9)估算 MS 的位置 (x_m, y_m) ：

$$r_1^2 = x_m^2 + y_m^2 \quad (7)$$

$$r_2^2 = (x_2 - x_m)^2 + (y_2 - y_m)^2 \quad (8)$$

$$r_3^2 = (x_3 - x_m)^2 + (y_3 - y_m)^2 \quad (9)$$

在不失一般性的情況下可以假設 $r_1 < r_2 < r_3$ ，定位示意圖如下圖 2.11：

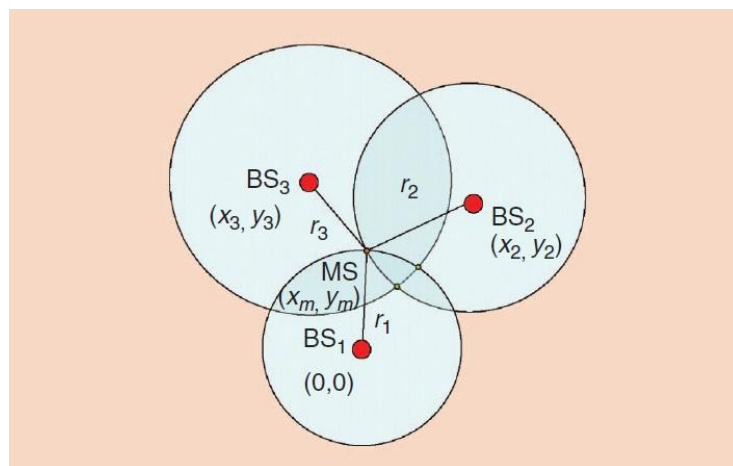


圖 2.11 使用三個基地台做 TOA 定位示意圖。

要解出所給定的方程式多過於未知解的非線性系統，又要使得三個以上的基地台量測資訊能夠提高精確度與減少誤差，可將此問題使用最小平方法(least square method)求

解，將式(8)減去式(7)可得到下式：

$$r_2^2 - r_1^2 = x_2^2 - 2x_2x_m + y_2^2 - 2y_2y_m \quad (10)$$

同理，將式(9) 減去式(7)可得：

$$r_3^2 - r_1^2 = x_3^2 - 2x_3x_m + y_3^2 - 2y_3y_m \quad (11)$$

將上面兩個方程式寫成矩陣型式如下：

$$\begin{bmatrix} x_2 & y_2 \\ x_3 & y_3 \end{bmatrix} \begin{bmatrix} x_m \\ y_m \end{bmatrix} = \frac{1}{2} \begin{bmatrix} k_2^2 - r_2^2 + r_1^2 \\ k_3^2 - r_3^2 + r_1^2 \end{bmatrix} \quad (12)$$

其中 $k_i^2 = x_i^2 + y_i^2$ ，式(12)可改寫成下式：

$$Hx = b \quad (13)$$

其中 H、x、b 分別為 $H = \begin{bmatrix} x_2 & y_2 \\ x_3 & y_3 \end{bmatrix}$ ， $x = \begin{bmatrix} x_m \\ y_m \end{bmatrix}$ ， $b = \frac{1}{2} \begin{bmatrix} k_2^2 - r_2^2 + r_1^2 \\ k_3^2 - r_3^2 + r_1^2 \end{bmatrix}$ 。

當量測值超過三個以上時可將算式寫成：

$$H = \begin{bmatrix} x_2 & y_2 \\ x_3 & y_3 \\ x_4 & y_4 \\ \vdots & \vdots \end{bmatrix}, b = \begin{bmatrix} k_2^2 - r_2^2 + r_1^2 \\ k_3^2 - r_3^2 + r_1^2 \\ k_4^2 - r_4^2 + r_1^2 \\ \vdots \end{bmatrix}$$

此種情況下式(13)需改寫如下：

$$\hat{x} = (H^T H)^{-1} H^T b \quad (14)$$

2.3.4 TDOA 法

TDOA 定位法解決了 TOA 定位法的限制，即 TOA 定位法需要 MS 與 BS 有精確的時間同步，而 TDOA 因為是計算各個訊號到達的時間差值，因此 MS 不需要與 BS 精確的同步，提升了定位精確度。

假設我們分別量測到 MS 到 BS_i 與 BS_1 的 TOA 值，將其相減可得 $t_i - t_1$ 的時間差值，換算成距離差則如下算式：

$$\begin{aligned} r_{i1} &\triangleq r_i - r_1 \\ &= (t_i - t^0)c - (t_1 - t^0)c = (t_i - t_1)c \end{aligned} \quad (15)$$

上式中， t^0 在相減中消去，表示 TDOA 定位法可以排除 t^0 所產生的時間同步誤差，因此

MS 不需要與 BS 做時間同步。將式(8)代入 TDOA 量測值 r_{21} 可以改寫成下式：

$$(r_{21} + r_1)^2 = k_2^2 - 2x_2x_m - 2y_2y_m + r_1^2 \quad (16)$$

將上式展開後如下：

$$-x_2x_m - y_2y_m = r_{21}r_1 + \frac{1}{2}(r_{21}^2 - k_2^2) \quad (17)$$

同理，式(9)可整理如下：

$$-x_3x_m - y_3y_m = r_{31}r_1 + \frac{1}{2}(r_{31}^2 - k_3^2) \quad (18)$$

將以上式子寫成矩陣形式如下：

$$Hx = r_1c + d \quad (19)$$

$$\text{其中 } c = \begin{bmatrix} -r_{21} \\ -r_{31} \end{bmatrix}, d = \frac{1}{2} \begin{bmatrix} k_2^2 - r_{21}^2 \\ k_3^2 - r_{31}^2 \end{bmatrix}。$$

將式(19)乘上 H^{-1} 可以得到一個利用未知數 r_1 求解 x 的算式如下：

$$x = r_1H^{-1}c + H^{-1}d \quad (20)$$

為了解出未知數 r_1 ，需將式(20)改寫成 r_1 的一次線性解如下式：

$$\begin{bmatrix} x_m \\ y_m \end{bmatrix} = r_1 \begin{bmatrix} a_1 \\ a_2 \end{bmatrix} + \begin{bmatrix} b_1 \\ b_2 \end{bmatrix} \quad (21)$$

$$\text{其中 } \begin{bmatrix} a_1 \\ a_2 \end{bmatrix} = H^{-1}c, \begin{bmatrix} b_1 \\ b_2 \end{bmatrix} = H^{-1}d。$$

將式(21)平方後代入式(7)可得到 r_1 的解方程式如下：

$$ar_1^2 + br_1 + c = 0 \quad (22)$$

$$\text{其中 } a = a_1^2 + a_2^2 - 1, \quad b = 2(a_1b_1 + a_2b_2), \quad c = b_1^2 + b_2^2。$$

式(21)中的正根解即為 r_1 解，解出 r_1 後代入式(21)可得到 (x_m, y_m) ，此即MS的估測位置。

當量測值超過3個以上時，可類似於上面 TOA 的方式改寫參數如下：

$$H = \begin{bmatrix} x_2 & y_2 \\ x_3 & y_3 \\ x_4 & y_4 \\ \vdots & \vdots \end{bmatrix}, c = \begin{bmatrix} -r_{21} \\ -r_{31} \\ -r_{41} \\ \vdots \end{bmatrix}, d = \frac{1}{2} \begin{bmatrix} k_2^2 - r_{21}^2 \\ k_3^2 - r_{31}^2 \\ k_4^2 - r_{41}^2 \\ \vdots \end{bmatrix}$$

最終MS的位置可由下式計算得到

$$\hat{x} = (H^T H)^{-1} H^T (r_1 c + d) \quad (23)$$



第3章

立體視覺

立體視覺(Stereo Vision)，利用兩台或兩台以上的相機從不同的位置拍攝物體，模擬人眼的視覺系統，人兩眼所看到的畫面不同，這些微小的差異經由人腦分析可以判斷出物體的遠近距離。同理，使用兩台相機相隔一段距離(baseline)，藉由比對物體在兩張照片中的偏移距離(disparity)，將可推算出物體的深度。

3.1 相機模型

相機是模擬人類眼球成像原理的一個裝置，將相機當作一個針孔(pinhole)，光線經由針孔後再投影至成像平面(retinal plane)，將空間中的三維坐標轉換至影像的二維坐標利用的是所謂透視投影(perspective projection)的轉換，如下圖 3.1 所示：

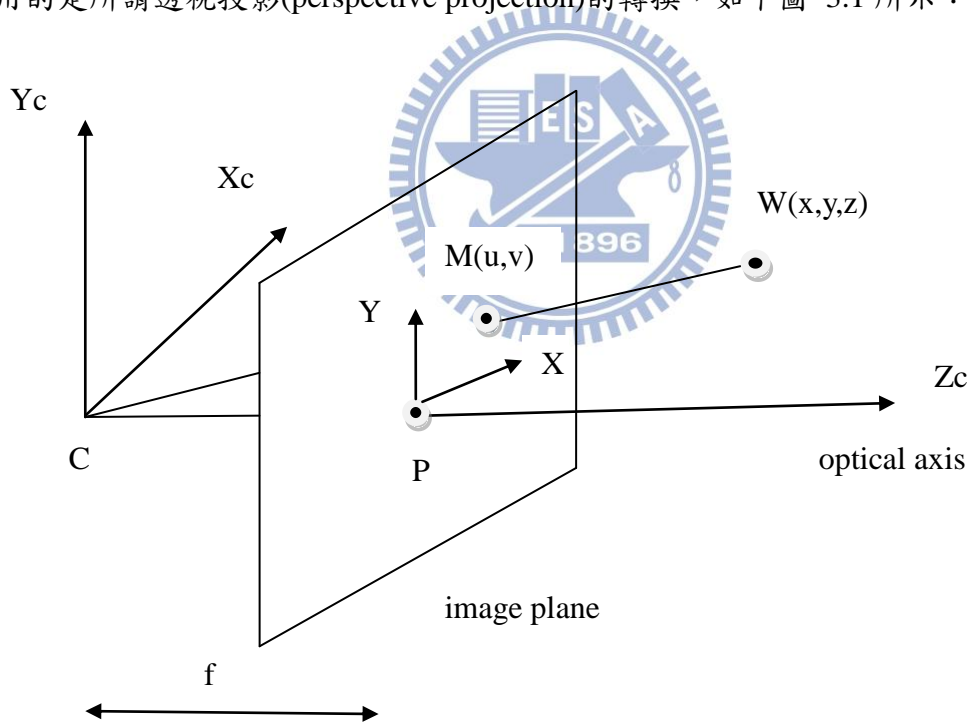


圖 3.1 透視投影。

圖 3.1 中 W 為空間中的一點坐標 (x, y, z) ， M 為 W 投影至影像平面上的成像點， M 在相機坐標系統中的坐標為 (u, v, f) ， C 是相機中心也稱作光學中心(optical center)，光軸(optical axis)為包含 C 且垂直於成像平面的向量， P 是基準點(principle point)也就是影像的中心點，為光軸與成像平面的交點，相機中心到成像平面的距離稱為焦距，以 f 表示也就是

\overline{CP} ， X_c 、 Y_c 、 Z_c 為相機的坐標軸， X 、 Y 為影像的坐標軸，由相似三角形比例關係可得到下式：

$$\frac{u}{x} = \frac{f}{z}, \quad \frac{v}{y} = \frac{f}{z} \quad (24)$$

經過移項整理得到

$$u = \frac{f}{z}x, \quad v = \frac{f}{z}y \quad (25)$$

所以 $M = (u, v) = (\frac{f}{z}x, \frac{f}{z}y)$ ，將其表示為三維的相機坐標則為 (u, v, f) ，因此可得到下式：

$$(x, y, z) = s(u, v, f) \quad (26)$$

其中 s 為一比例因子(scale factor)，隨著影像的解析度大小而異。將 M 用齊次坐標(homogeneous coordinates)可表示如下式：

$$\tilde{M} = \begin{bmatrix} fx \\ fy \\ z \end{bmatrix} = \begin{bmatrix} f & 0 & 0 & 0 \\ 0 & f & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} \quad (27)$$

一般來說在不考慮外在的因素下，影像的中心點位置會在光軸上，如圖 3.1 所示，但是受到透鏡輻射的影響可能會偏離光軸位置，此外一般數位相機的影像 xy 軸比例並不相等，所以將 f_x 、 f_y 分別代表 x 軸與 y 軸的焦距，因此 M 可改寫為下式：

$$M = (u, v) = \left(\frac{f_x}{z}x + c_x, \frac{f_y}{z}y + c_y \right) \quad (28)$$

其中 (c_x, c_y) 為成像中心(image center)。用齊次坐標表示如下式：

$$\tilde{M} = \begin{bmatrix} f_x x + c_x z \\ f_y y + c_y z \\ z \end{bmatrix} = \begin{bmatrix} f_x & 0 & c_x & 0 \\ 0 & f_y & c_y & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} \quad (29)$$

若是影像的 xy 軸不是正交，則會有一歪斜係數 γ 用來做校正，因此完整的相機內部參數(intrinsic parameter)矩陣為：

$$A = \begin{bmatrix} f_x & \gamma & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{bmatrix} \quad (30)$$

內部參數矩陣是將相機坐標空間轉換至二維影像空間，因此不會隨著外在環境的變化而改變，只要是使用同一台相機，且相機實體焦距不改變，則此內部參數矩陣可繼續使用不需重新校正。外部參數(extrinsic parameter)矩陣則是將世界坐標系統(world coordinate system)轉換為相機坐標系統(camera coordinate system)，如下圖 3.2 所示：

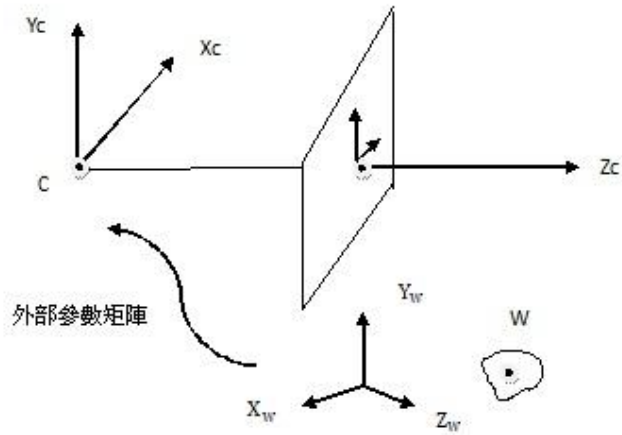


圖 3.2 坐標系統轉換示意圖。

外部參數矩陣由一個旋轉矩陣 R 與一個平移矩陣 t 組成如下：

$$[R|t] = \begin{bmatrix} r_{11} & r_{12} & r_{13} & t_1 \\ r_{21} & r_{22} & r_{23} & t_2 \\ r_{31} & r_{32} & r_{33} & t_3 \end{bmatrix} \quad (31)$$

旋轉矩陣是要將世界坐標系統的三軸轉成與相機坐標系統的三軸平行，平移矩陣則用來將世界坐標系統的原點平移至相機坐標系統的原點 C 。

透視投影就是利用相機的內外部參數來將空間中的物體坐標 W 轉換至影像平面坐標上 M ，假設透視投影矩陣為 P ，以齊次坐標表示則如下式：

$$\tilde{M} \cong P\tilde{W} \quad (32)$$

其中 $\tilde{M} = [u \ v \ 1]^T$ ， $\tilde{W} = [x \ y \ z \ 1]^T$ ， $P = A[R|t]$ ， \cong 表示可任意乘上一個比例常數，將上式展開如下：

$$s \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \begin{bmatrix} f_x & \gamma & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} r_{11} & r_{12} & r_{13} & t_1 \\ r_{21} & r_{22} & r_{23} & t_2 \\ r_{31} & r_{32} & r_{33} & t_3 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} \quad (33)$$

s 為一比例因子。

3.2 立體視覺成像

將空間中的一點利用透視投影使其成像在影像平面，這種轉換失去了一個維度(深度)，也就是說無法單由一個成像點反推回原本物體在空間中的位置，但是如果利用兩個以上的成像點加上相機間的幾何關係即可反推回物體在空間中的位置，此過程稱為立

體視覺成像。假設在空間中有一點 $w=(x,y,z)$ ，兩台相機利用透視投影將其成像於左右成像平面，兩台相機的光學中心分別為 C_l 與 C_r ，焦距皆為 f ，成像點坐標分別為 (x_l, y_l) 與 (x_r, y_r) ，將世界坐標原點定為左相機的光學中心，水平方向為 x 軸方向，垂直方向為 z 軸方向， y 軸為垂直於 $x-z$ 平面(紙面)的方向，兩相機間的距離為 b (baseline)，概念如下圖 3.3 所示：

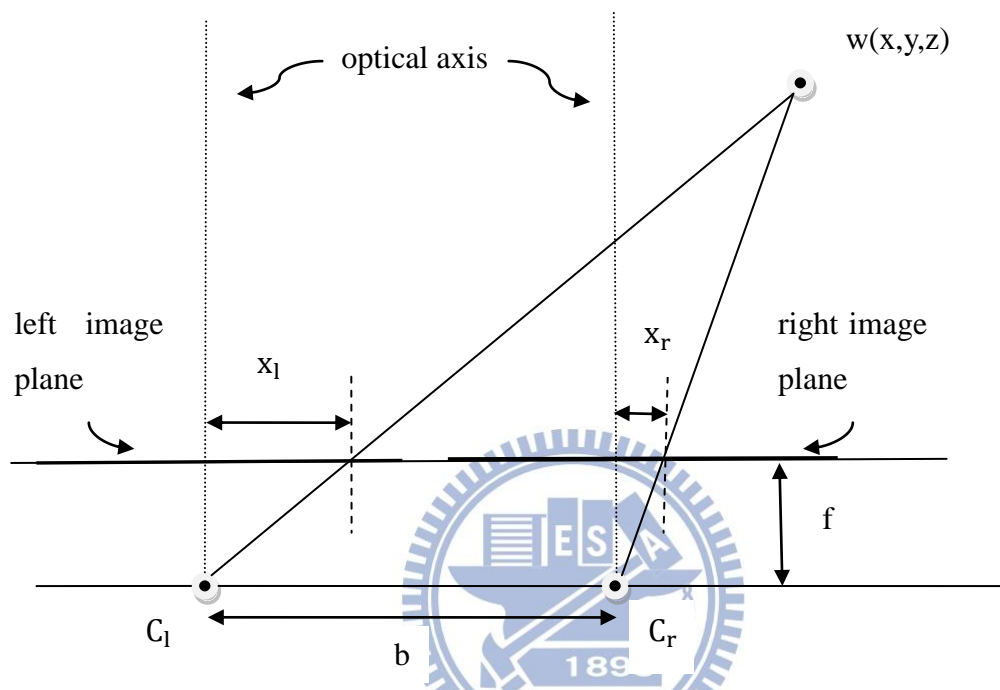


圖 3.3 立體視覺成像。

由圖 3.3 物體的深度可由相似三角形原理求出，如下式：

$$\frac{x}{x_l} = \frac{z}{f}, \quad \frac{x-b}{x_r} = \frac{z}{f} \quad (34)$$

將上式合併得到下式

$$z = \frac{bf}{x_l - x_r} \quad (35)$$

$x_l - x_r$ 定義為 disparity，可用 d 來表示。將上式改寫如下

$$z = \frac{bf}{d} \quad (36)$$

由上式可知，假設我們找到空間中物體分別在左右影像的投影點，再經由相機校正得知相機焦距與其基準線的長度，即可由上式推算出物體的深度。光軸的平行與否會影響計算出的物體坐標深度，除非用精密的儀器校正否則很難達到光軸的平行。

3.3 極線幾何

極線幾何(Epipolar geometry)是用來描述兩相機間的對應關係，藉由兩個相機的原點 C_1 與 C_2 和物件點 W 在空間中產生的一個平面，此平面稱為極線平面(epipolar plane)，相機原點間的連線稱為基準線(baseline)，物體在兩相機分別成像於點 M_1 、 M_2 ，基準線與兩成像平面的交點稱為極點(epipole)，分別為 E_1 與 E_2 ，極線(epipolar line)是極線平面與成像平面的相交線，也是極點與成像點的連接線，即 $\overrightarrow{E_1M_1}$ 與 $\overrightarrow{E_2M_2}$ ，如下圖 3.4 所示：

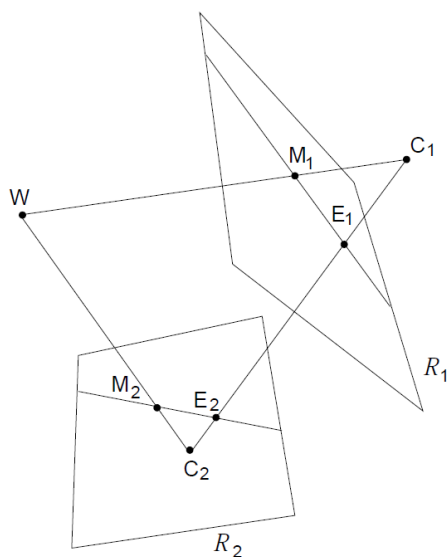


圖 3.4 極線幾何。

極限幾何對於成像點的位置有所限制(epipolar constraint)，假設物件點 W 在左畫面中成像於 M_1 ，那麼在右畫面中成像的位置 M_2 一定會在相對應的極線上，且影像中的所有極線都會通過極點。有一種特別的情況是，當基準線與兩成像平面都平行時，極點會產生在無窮遠，此時兩成像平面中的所有極線皆會是平行線，如下圖 3.5 所示：

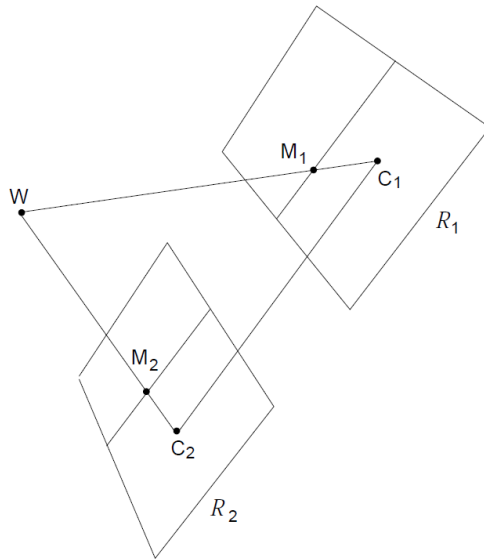


圖 3.5 校正(Rectification)。

當圖 3.5 的情況發生時，假設我們已知在左畫面中的一個成像點，那麼它在右畫面中對應的成像點將會是在同一個水平線上，在數位影像中也就是同一個列，將原本的左右影像轉換到如圖 3.5 的關係，此過程就稱為校正(Rectification)。

校正(Rectification)的原理是將原本的相機做旋轉，使得基準線平行左右相機新的 x 軸，這樣極線才會是水平線，因此左右影像的對應點將會是在相同的垂直坐標上，要達到此目的兩相機的內部參數必須是相同的，也就是焦距相同且兩成像平面共平面[18]。透視投影矩陣 P 在校正前後光學中心都是相同的，只有方向改變了，且校正完後的兩個新透視投影矩陣其內部參數皆相同，唯一不同的只有光學中心的位置，可以想像是同一台相機延著 x 軸移動。校正的目的是為了方便匹配演算法的計算，極線幾何限制將匹配搜尋範圍從二維影像限制在一維的極線上，校正則將極線限制在同一水平方向上。

3.4 匹配問題

利用立體視覺計算深度資訊最困難的問題便是如何在兩張照片中找到物件的對應關係，亦即在左畫面中的每個物件位置要如何在右畫面中找到相同的物件位置，這個被稱作是匹配問題。在本論文中使用全域的匹配演算法，其中因為 DP 的概念較符合直覺且易懂，加上其快速的運算時間，所以本論文採用 DP 解決匹配問題。

3.4.1 DP

Dynamic Programming(DP)這個詞首次被提出是在 1940 年代由 Richard Bellman 所使用[19]，當時他是將它用來描述一個需要被逐次的找出最佳解的程序，將一個複雜的問題分解成好幾個簡單的子問題來解決。DP 屬於全域演算法的一類，假設左右影像只有水平方向的位移，則可利用它來找出每對掃描線花費最小的路徑，由於影像中每個列將是獨立處理運算的，造成鄰近兩個列在垂直方向上可能會有深度不連續的情況發生，也就是 disparity map 將會出現一些水平方向上的橫條紋情況，這是由於 DP 的演算法所造成的，可利用一些事後平滑處理來解決，在本論文中使用 median filter 對計算出的 disparity map 做後製處理。

3.4.2 限制條件

在使用 DP 演算法計算最小花費路徑時常會利用一些限制條件[20]以符合實際情況，例如：

1. 唯一性(uniqueness)：在左右影像中的特徵匹配只能是一對一的，不能有兩個以上的對應關係。
2. 單調次序性(monotonic ordering)：假設左影像的 P_i 對應到右影像的 P_j ，那麼左影像的 P_{i+1} 只能對應到右影像的 P_{j+k} ， $k > 0$ ， P 代表一個 pixel 點。

有了上述的約束條件，匹配問題就近似轉化成了一個找出從左到右從上到下的一條最短路徑的問題，如下圖 3.6 所示：

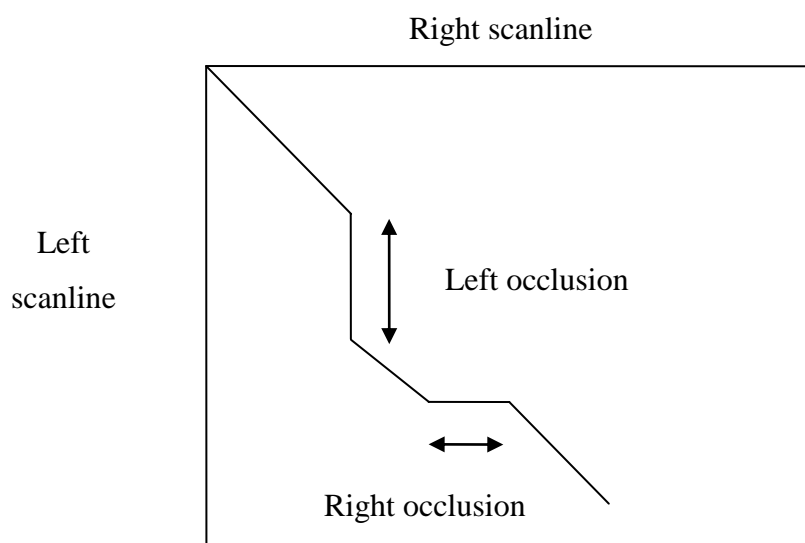


圖 3.6 左右影像匹配路徑示意圖。

圖 3.6 中畫出了一條可能的匹配路徑，橫坐標 right scanline 代表的是右影像 scanline 的每個 pixel 點，縱坐標 left scanline 則是左影像 scanline 的每個 pixel 點，由於單調次序性的條件限制，匹配路徑只能是斜率小於等於零的直線或是斜率無窮大。從 left scanline 的觀點來看，垂直路徑代表著多對一的情況，即多個左影像 pixel 點都對到右影像的同一個 pixel 點，意味著左遮蔽，而水平路徑則剛好相反，代表一對多的情況，即一個左影像 pixel 點對應到多個右影像的 pixel 點，意味著右遮蔽，這些都違反了唯一性的條件限制，因此在匹配演算過程中，這些路徑將給予額外的遮蔽花費。斜線路徑的部分則是滿足了上述的限制條件，是我們所樂見的，斜率與此區域內的 disparity 遞增量有關。

匹配路徑的路途長短可看成是匹配時的花費，根據匹配花費選擇最小的匹配路徑解，最後只要對每條左右影像的 scanline 都做相同的計算，即可求出整張影像的 disparity map。由於 DP 為了求出全域的最佳解必須把 left scanline 中的每個 pixel 都與 right scanline 的每個 pixel 做匹配計算，這個計算量是非常龐大的，因此為了提高運算速度通常會對 disparity 的範圍做一個限制，此限制不但可以大大減少運算量也能有效減少很明顯的匹配錯誤(例如 disparity 過大)，如下圖 3.7：

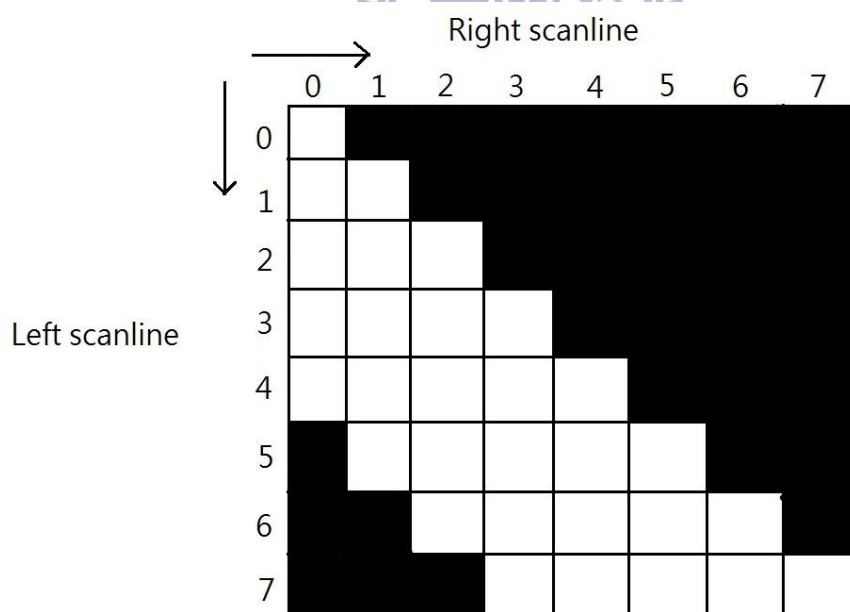


圖 3.7 限制 disparity 運算範圍的 DP。

圖 3.7 中顯示了限制運算範圍的 DP，disparity 的範圍限制在 0 到 4，白色方格是需要被計算的區域，灰色方格因超出設定的 disparity 範圍所以不需考慮計算。物件不管擺放在空間中哪個位置，在左影像的成像點水平坐標一定會大於等於右影像的成像點水平坐標

(假設坐標原點為影像的左上角)，下圖 3.8 證明了此一特性

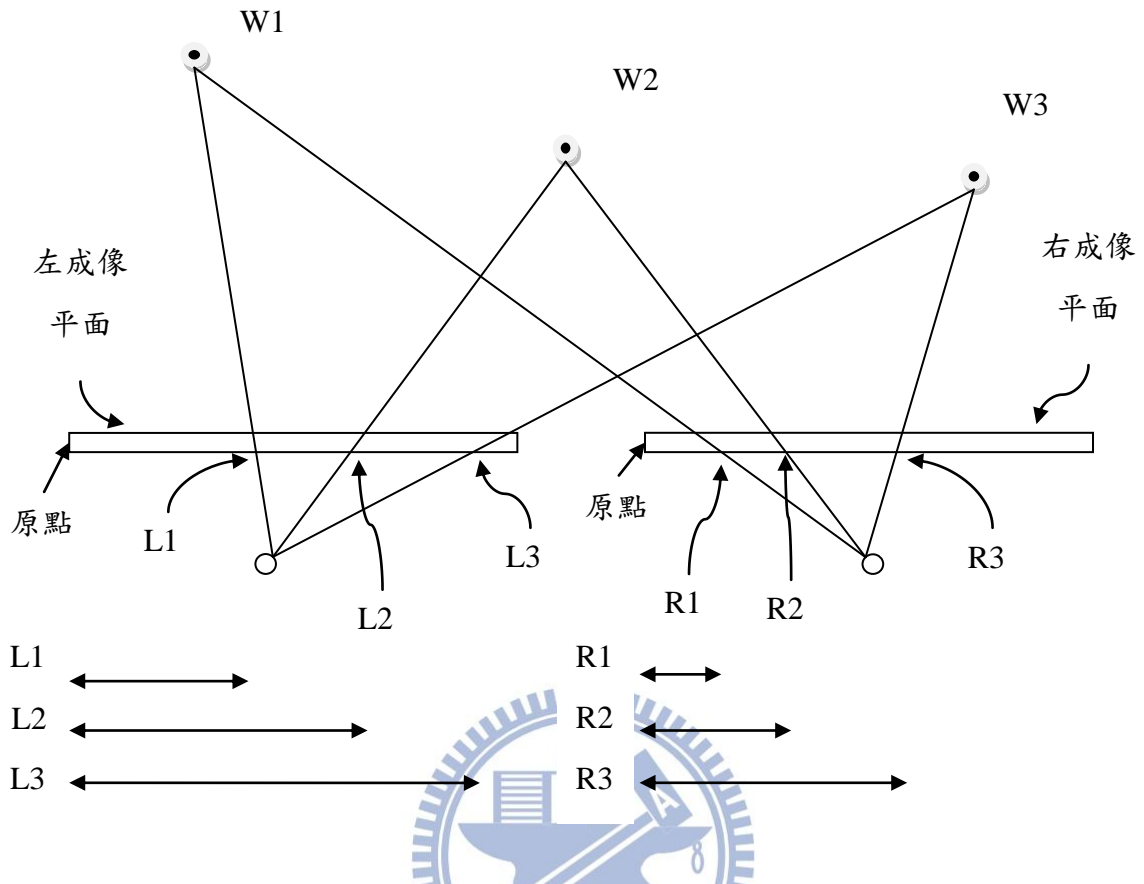


圖 3.8 左右成像點關係。

圖 3.8 中顯示了三個在不同位置的物件 W1、W2、W3 分別成像於左右影像平面，在左成像平面中的成像點水平坐標值為 L1、L2、L3，在右成像平面中的成像點水平坐標值為 R1、R2、R3，假設兩成像平面原點都在影像的左上角，在圖中可看成是成像平面中最左邊的位置，圖中可看出 $L1 \geq R1$ ， $L2 \geq R2$ ， $L3 \geq R3$ ，因此說明了不論物體在何處，左影像成像點的水平坐標必定大於等於右影像成像點的水平坐標，所以限制 disparity 運算範圍的 DP 才會如圖 3.7 白色區塊一樣是階梯狀的運算範圍。

3.4.3 花費函數

如果我們將 DP 比喻成找出一條最短路徑的演算法，那麼花費函數(cost function)就代表著一條匹配路徑的路途長短，而路徑中每段路途的花費則可用匹配時的相似度來代表，兩個匹配點相似度愈小則此路途愈長。本論文根據文獻[21]利用 NSSD(Normalized Sum of Squared Difference)函數計算匹配點間的相似度來當作匹配花費(matching cost)，

如下式：

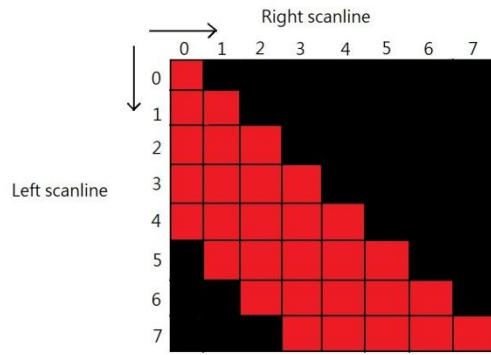
$$M(l,r) = \frac{1}{2} \frac{\sum_{i=0}^{N_1} \sum_{j=0}^{N_2} [(I_L(l_x + i, l_y + j) - \bar{I}_L) - (I_R(r_x + i, r_y + j) - \bar{I}_R)]^2}{\sum_{i=0}^{N_1} \sum_{j=0}^{N_2} (I_L(l_x + i, l_y + j) - \bar{I}_L)^2 + \sum_{i=0}^{N_1} \sum_{j=0}^{N_2} (I_R(r_x + i, r_y + j) - \bar{I}_R)^2} \quad (37)$$

其中 l 與 r 分別為左右影像的對應像素點，坐標分別為 (l_x, l_y) 與 (r_x, r_y) ， I_L 與 I_R 分別為左右影像的 pixel 亮度值，匹配視窗(matching window)的大小為 $N_1 \times N_2$ ， \bar{I}_L 與 \bar{I}_R 分別為左右影像匹配視窗內的平均 pixel 亮度值。

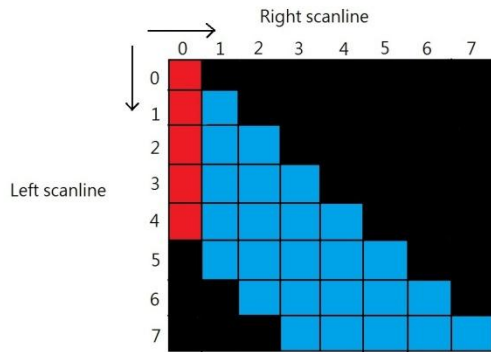
計算完匹配花費之後，我們將從圖 3.7 左上方開始針對每個需計算的區域(圖 3.7 中的白色方格)根據其前一點可能的路線方向(前一點可能從左方、上方或左上方來)計算其累積花費，如下式：

$$C(l,r) = \min \begin{cases} C(l-1,r) + K_{occ} \\ C(l-1,r-1) + M(l,r) \\ C(l,r-1) + K_{occ} \end{cases} \quad (38)$$

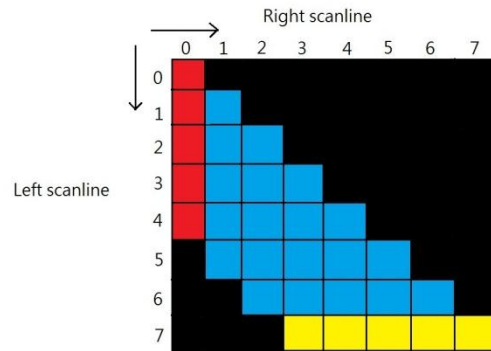
其中 K_{occ} 為遮蔽花費，在本論文中為一常數。由上式計算出了每個白色區塊的累積花費 C 之後(第一欄的累積花費等於本身的匹配花費)，也順便將其前一個匹配點的位置記錄下來以供最後尋找最佳匹配路線時使用。最佳的匹配選擇也就是根據累積花費選擇出最短路徑，選擇的方式是從最後一列可能的匹配選擇中找出最小匹配花費的配對，整個計算程序舉例如下圖 3.9



(a)



(b)



(c)

圖 3.9 DP 計算程序。

圖 3.9(a)為針對需考慮的部分計算匹配花費 M ，以紅色方格表示，(b)為計算累積花費 C 所得到的數值，以藍色方格表示，注意第一欄仍為紅色，表示累積值是從第二欄之後的方格才開始計算，(c)為將最後一列匹配選擇以黃色方格表示，最佳的匹配路徑就是從黃色方格中找出最小的花費數值的那一格做為最後匹配方格，因為在計算累積花費時(藍色方格)，每個方格額外記錄了他的前一個匹配方格，因此我們可從最後匹配方格往回找出匹配路徑，並計算出 disparity map。

由立體視覺成像原理中我們已知無法僅由一個成像點來推算物體在空間中原本的位置，如同在單視域中假設我們有的只是單張影像的資訊，想從其中的一些深度線索完整的推測出畫面中的景物深度關係略顯困難，因此我們考量影像在時間軸上的關係，利用前後兩張影像間的移動變化做為另一項重要的深度線索，藉由這些深度線索來「猜測」畫面中物體可能的深度關係，

4.1 深度線索

深度線索是當做幫助我們估測畫面深度圖的依據，在本論文中僅憑單一影像我們能使用的深度線索有材質漸層與大氣透視，而利用前後影像間的關係則是利用動態視差這項深度線索，以下小節將分別介紹我們使用的各項深度線索。在視訊影像中由於編碼壓縮方面的考量，因此將影像切割成一塊塊固定大小的 Macroblock(MB)，做為其處理的每一個元件(element)，而非對每一個 pixel 做處理，在本論文的各項深度線索計算也將以 MB 做為最小處理單元。

4.1.1 動態視差(motion parallax)

在連續的影像中人眼對於物體移動的變化最為敏感，也就是物體的動態視差愈大距離我們可能愈近。動態視差的計算方法是計算連續影像中每個 Macroblock(MB)的移動向量，Block matching 在移動向量的估測上是常被使用的方式，因為其硬體實做複雜度低，為了提高運算速度也發展出了許多快速的 block matching 演算法[22]，block matching 示意圖如下圖 4.1：

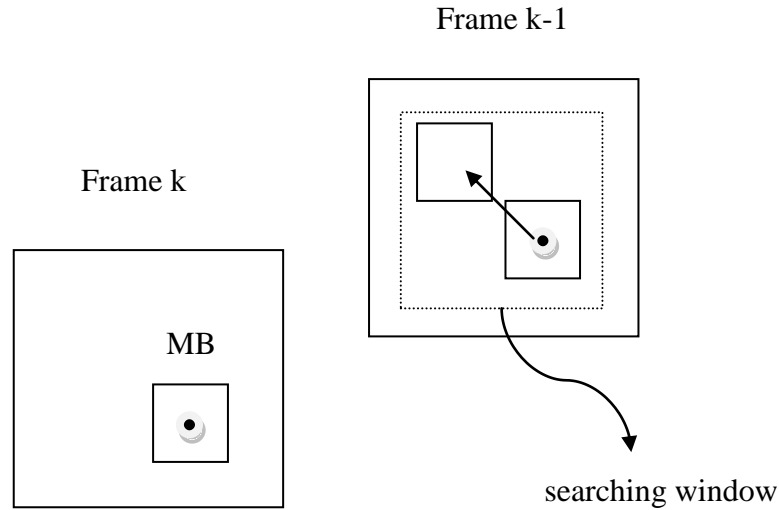


圖 4.1 Block matching。

圖 4.1 中每個拿來 matching 的 element 是一個 MB，黑點為 macroblock 的中心點，通常限制一個搜索範圍(searching window)做搜尋，搜索範圍的大小可根據影像性質自由做設定。匹配函數通常使用 MAD(mean absolutely difference)或 MSE(mean squared error)如下式(40)(41)：

$$MAD(u, v) = \frac{1}{N_1 N_2} \sum_{n_1=0}^{N_1-1} \sum_{n_2=0}^{N_2-1} |I(n_1, n_2, k) - I(n_1 + u, n_2 + v, k - 1)| \quad (39)$$

$$MSE(u, v) = \frac{1}{N_1 N_2} \sum_{n_1=0}^{N_1-1} \sum_{n_2=0}^{N_2-1} (I(n_1, n_2, k) - I(n_1 + u, n_2 + v, k - 1))^2 \quad (40)$$

其中 MB 的大小為 $N_1 \times N_2$ ， (u, v) 為一組候選的移動向量， (n_1, n_2) 為 MB 裡的每一個 pixel， I 為 pixel 的亮度值。移動向量的選擇將是在搜索範圍中產生最小花費的一組 (u, v) 解，如下式(42)(43)：

$$(u, v) = \arg \min_{(u, v)} MAD(u, v) \quad (41)$$

$$(u, v) = \arg \min_{(u, v)} MSE(u, v) \quad (42)$$

在有立體感的 2D 影像對中，垂直方向的差異(disparity)會造成人眼觀看的不適以及增加影像融合的困難度，反之，一範圍內的水平差異是人眼可以接受的並且產生立體感，為此在單視域影像轉換為立體視域影像時垂直方向上差異的消除將是必行且重要的，消

除的方式是取移動向量的大小(norm)做為該 MB 的水平移動量，如下式：

$$d_B = \sqrt{u^2 + v^2} \quad (43)$$

其中 u, v 為 MB 的移動向量，用 d_B 代表此 MB 的移動量，移動量愈大表示物體深度愈小。

4.1.2 材質漸層(texture gradient)

相同的紋理由遠而近會產生一種漸層感，我們將每個 MB 的 edge 密度當作其深度線索，密度愈大深度則愈近，edge 則利用 Sobel operator 做偵測如下圖 4.2：

-1	-2	-1
0	0	0
1	2	1

(a)

-1	0	1
-2	0	2
-1	0	1

(b)

圖 4.2 Sobel operator。

圖 4.2 為 Sobel operator 常用來做邊緣偵測的運算子，(a) 可用來偵測水平方向上的 edge，(b) 則用來偵測垂直方向上的 edge。Edge 密度則由下式計算：

$$D = \frac{\sum_{i \in MB} E(i)}{S_{MB}} \quad (44)$$

其中 S_{MB} 代表 MB 的大小， E 為 edge 點判斷函數，如果是 edge， E 為 1，反之則為 0。

4.1.3 大氣透視(atmospheric perspective)

因為空氣中微小粒子與霧氣的影響，造成遠方的景物較模糊近的景物較清晰，此現象稱為大氣透視[23]，如下圖 4.3

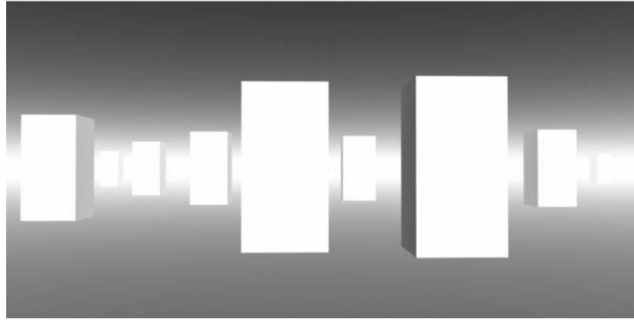


圖 4.3 大氣透視。

我們利用每個 MB 的變異性(variance)和對比度(contrast)來代表此 block 的清晰度，值愈大，物體距離愈近。變異性和對比度計算如下列式子：

$$\text{Var}(B) = \frac{1}{N} \sum_{i \in B} (x_i - \bar{x})^2 \quad (45)$$

$$\text{Con}(B) = \frac{x_{\max} - x_{\min}}{x_{\max} + x_{\min}} \quad (46)$$

其中，B 代表某個 MB，x 為 pixel 的亮度值， \bar{x} 為該 MB 內像素的平均亮度，N 為 MB 內的像素數量，而 x_{\max} 與 x_{\min} 分別為該 MB 內的最大與最小 pixel 亮度值。

4.2 深度圖修正

由於初步猜測的深度圖將會有一些錯誤估測的部分，例如過於平滑的前景物體內部將會產生空洞(hole)，或是由於線性加成的關係讓背景區塊誤判為是前景物體，因此必須對深度圖做一些修正處理，首先為了切割前景與背景，我們將初步深度圖做二值化處理，也就是設一門檻值，小於此門檻值則被當作是背景部分，大於的則屬於前景部分，由於門檻值過大會造成前景物體切割不正確，過小則會將過多的背景部分當做是前景，經由不同的門檻值模擬結果，門檻值設在 26 到 43 之間可以得到較佳的深度圖修正結果，本論文中將此二值化門檻值設為 40，經過此步驟後會得到一個初步估測的二值化前景與背景圖，接著我們利用相鄰單元標記法(connected component labeling)將相鄰的區塊部分加上同樣的標記，並計算出區塊的面積大小，並對前景物體內部的空洞做填補的動作(hole filling)，也移除因為誤判而產生的面積過小的前景區塊。

4.2.1 相鄰單元標記法(connected component labeling)

本論文使用簡單易懂的 two-pass 法來實做相鄰單元標記法[24]，做相鄰單位標記之前需要的是一張二值化的影像，此二值化影像區分出前景物體與背景，演算法將給予前景部分標記，背景部分則會被標記為同一個數值(本論文中將背景的 pixel 都標記為 0)。相鄰的定義可以是周圍四個方向或是八個方向當成是相鄰的 pixel，在本論文中使用的是四個方向的相鄰定義，也就是以每個 pixel 而言在其上、下、左、右四個方向的 pixel 才算是相鄰的。

First pass :

假設我們從影像的左上角開始處理，逐欄逐列的尋找前景 pixel 做為處理對象，將此 pixel 上方與左方的前景與背景情況取出，根據下列判斷式做不同處理：

1. 上方與左方都是背景：給此 pixel 新的編號。
2. 其中一方是前景：給此 pixel 跟相鄰的前景編號一樣的編號。
3. 上方與左方都是前景：給此 pixel 其中較小的編號，並將這兩個算是相鄰的編號記錄下來。

Second pass :

根據相鄰編號的紀錄表重新給予每個 pixel 編號，規則是取相鄰編號中最小的編號為新編號，也就是所有被記錄成同一區塊的編號的 pixel 都將改成此編號，舉例如表 4.1

編號	First Pass(相鄰編號紀錄)	Second Pass
1	{1,2}	1
2		1
3	{3,4,5}	3
4		3
5		3
6	{6,7}	6
7		6

表 4.1 相鄰編號表範例。

在表 4.1 中，First pass 過後把相鄰編號情況記錄在第二欄中，編號 1 與 2 在 First pass 被記錄成是相鄰的編號，而編號 3 到 5 被記錄成相鄰的編號，編號 6 與 7 也被記錄成相鄰的編號。在演算法的 Second pass 之後，編號 2 的 pixel 將會被改成編號 1，編號 4 與 5 的 pixel 將會被改成編號 3，編號 7 的 pixel 則會被改成編號 6。

經由相鄰單元編號演算法我們可將二值化的前景與背景圖依據其相鄰情況做標記，

被標記為同一個數字的 pixel 將是相鄰的，我們可進一步算出相鄰 pixel 區塊的面積，這些資訊可做為後續使用。下圖 4.4 為前景相鄰區塊標記的結果

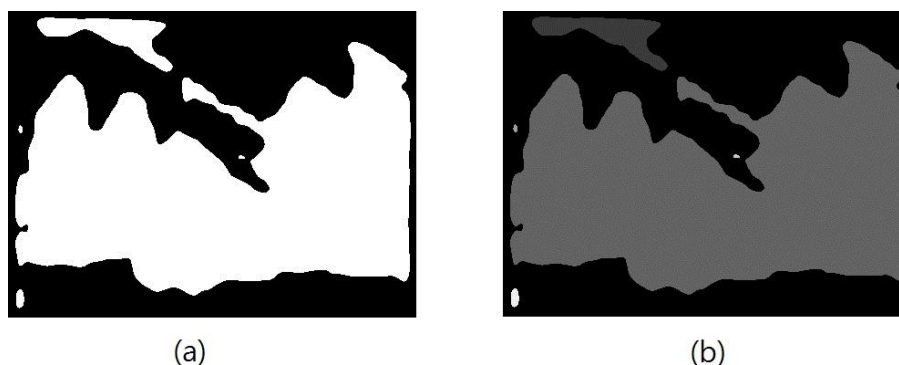


圖 4.4 前景相鄰單元標記。

圖 4.4 (a)為二值化過後的圖像，白色部分為前景，黑色部分為背景，(b)為利用相鄰單元標記法將相連的區塊做編號，我們將編號過後的每一塊前景區塊用不同的亮度表示，圖中依照亮度可看出標記了 5 塊不同的前景區塊。

4.2.2 空洞填補(hole filling)

將前景物體與背景做了編號之後，我們將面積過小的前景區塊與背景區塊當成是錯誤的而將其修正，面積過小的背景區塊是存在於前景區塊的內部(hole)，因為物體內部存在平滑部分導致的錯誤，而面積過小的前景區塊則是因為初步深度圖由四項深度線索線性加成，造成某些小區域背景被判定為是前景，我們將把這些小區域移除，此外，在一般視訊影像中前景物體通常位於影像的下半部，因此把前景區塊底部線位於影像上半部的區塊移除。下圖 4.5 為空洞填補與前景移除的結果

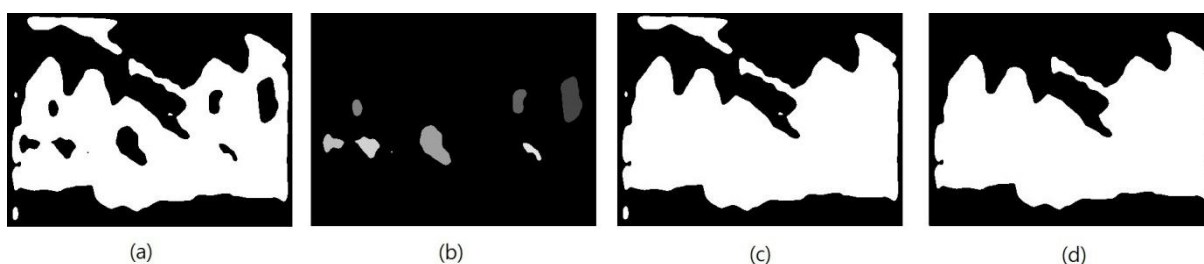


圖 4.5 空洞填補與前景移除。

圖 4.5(a)為二值化過後的前景背景圖，白色區塊為前景，黑色為背景，(b)為利用相鄰單元標記法標記出的空洞區塊(利用不同的亮度值顯示各個區塊)，(c)為將前景物體內部的空洞(b)填補後的結果，(d)為將面積過小或區塊底部線位於影像上半部的前景區塊移除

過後的結果。

移除過後的物體內部空洞是沒有給定深度值的，因此我們將這些空洞利用其周邊有值的部分做內插，本論文將每個屬於空洞的 pixel 為中心，取一矩形範圍內的最大深度值當做是內插值使用，下圖 4.6 為內插前後的深度圖

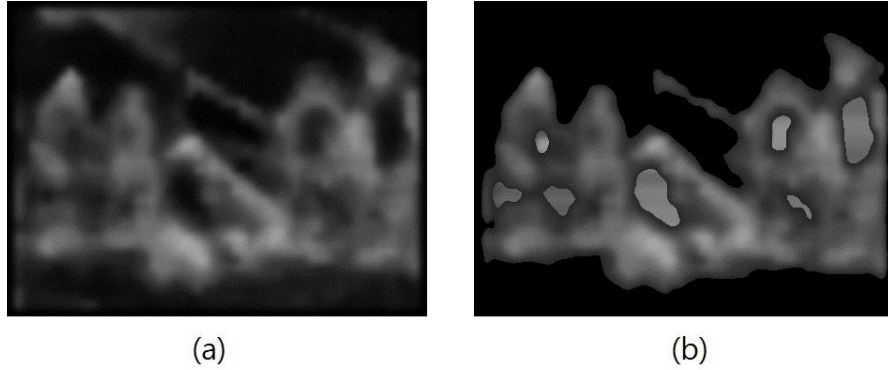


圖 4.6 空洞內插深度值。

圖 4.6 (a)為初步估計的深度圖，(b)為空洞填補以及移除小區塊前景之後對空洞區域內插深度值的結果。

4.2.3 前景背景深度計算

由於初步估測的深度圖在經過空洞填補、小面積前景區塊移除以及內插深度值的修正後，空洞區塊的深度值將會與周遭其他前景 pixel 有不平滑的現象(因為是取周遭的深度峰值做內插)，如圖 4.6(b)，因此最後的前景深度圖我們將對其使用高斯濾波器做平滑效果，也就是通過一個 FIR 濾波器，高斯函數在二維空間的方程式如下：

$$G(x,y) = \frac{1}{2\pi\sigma^2} e^{-\frac{x^2+y^2}{2\sigma^2}} \quad (47)$$

其中 σ 為標準差。假設平均值為 0，標準差為 1 的情況下，離散型態的高斯濾波器使用整數來逼近高斯函數，如下圖 4.7：

$\frac{1}{16} \times$	1	2	1
	2	4	2
	1	2	1

$\frac{1}{273} \times$	1	4	7	4	1
	4	16	26	16	4
	7	26	41	26	7
	4	16	26	16	4
	1	4	7	4	1

(a)
(b)

圖 4.7 高斯濾波器。

圖 4.7(a)為離散型態的 3*3 高斯濾波器，(b) 為離散型態的 5*5 高斯濾波器。由於我們估測的初步深度圖是以 MB 為單位(本論文中 MB 大小為 8*8 pixel)，即每 8*8 pixel 為相同的深度值，因此要做到平滑深度圖的動作要使用大於 8*8 大小的高斯濾波器才能有效果，在本論文中是使用 23*23 大小的高斯濾波器做平滑深度圖的動作，其結果如下圖 4.8

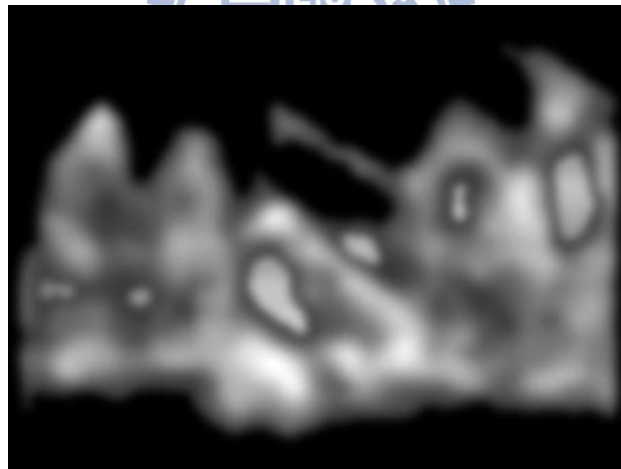


圖 4.8 使用高斯平滑化過後的前景深度圖。

圖 4.8 顯示了經過高斯濾波器之後的前景深度圖，空洞填補的部分已經有較好的平滑效果，但空洞周圍仍會產生一些黑色環形區域(深度值較小的部分)，這是因為在做二值化前景時取的門檻值的關係，門檻值若過大將無法有效區分出前景與背景區塊，反之，門檻值過小則會造成物體內部空洞區塊過小，而此黑色環形區域也將會愈大。

背景深度值則是根據消失點(vanishing point)為基準，消失點的選擇本論文則簡單的先找出最遠前景物體的底部線，以其為基準往上移動影像高度的十分之一做為消失點，消失點以下的背景給定梯度深度值如下式：

$$D(x,y) = \frac{y - y_{vp}}{H - y_{vp}} * 255 \quad (48)$$

消失點以上($y < y_{vp}$)的背景則給定深度值為 0，其中 y_{vp} 為消失點的 y 坐標， H 為影像高度。下圖 4.9 為背景深度的計算結果：



圖 4.9 背景深度圖。

有了修正過後的前景深度圖與背景深度圖之後，我們將其合併得到修正後的深度圖如下圖 4.10

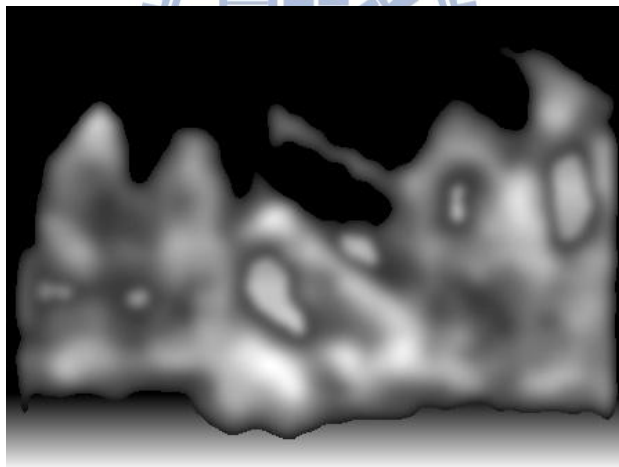


圖 4.10 修正過後的初步深度圖。

4.2.4 左右影像對合成法

有了原始影像以及估測出的深度圖之後，我們可利用 DIBR 的技術合成出左右眼影像，換句話說就是把深度當作位移量的大小，由於過大的位移量將無法得到好的立體效果與觀看舒適度，而估測出的深度圖範圍值是 0-255，我們簡單的將深度值除以 15 以 scale 到 0-17 的範圍值當做位移量，經實驗結果此位移量是能夠得到好的觀看效果。DIBR

即是利用原始影像每個 pixel 相對應的深度值做為合成左右眼影像的基礎，在本論文中原始影像被當作是中間影像，也就是說將原影像右移可產生左影像，左移可產生右影像，運算式如下：

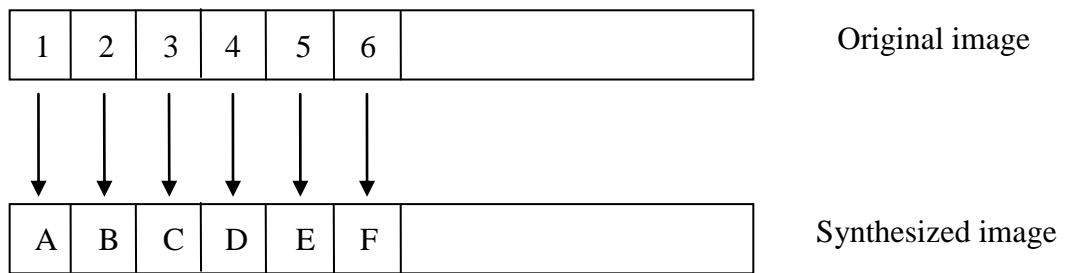
$$x_l = x_c + \frac{\text{disparity}}{2} \quad (49)$$

$$x_r = x_c - \frac{\text{disparity}}{2} \quad (50)$$

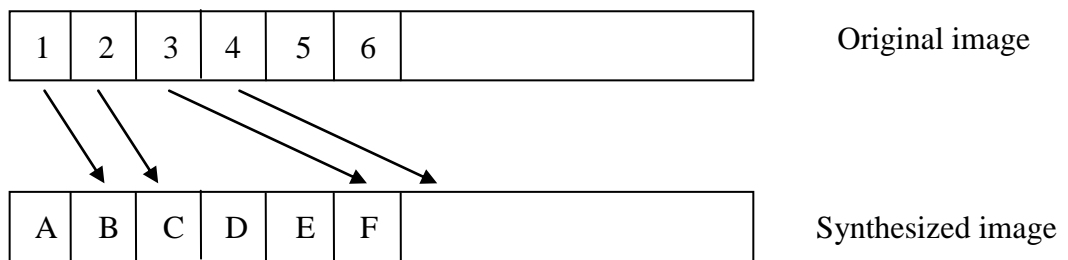
其中 x_c 為原始影像的 pixel 位置， x_l 與 x_r 分別為產生的左影像與右影像的 pixel 位置，將其相減則得到此左右影像的 disparity，也就是說我們當初拿來合成左右影像的偏移量即為 $\frac{\text{disparity}}{2}$ 。將原始影像偏移而得到新影像的過程中將會有一些情況發生，以合成左影像為例，我們按照下圖 4.11 的規則來實做



Case 1



Case 2



Case 3

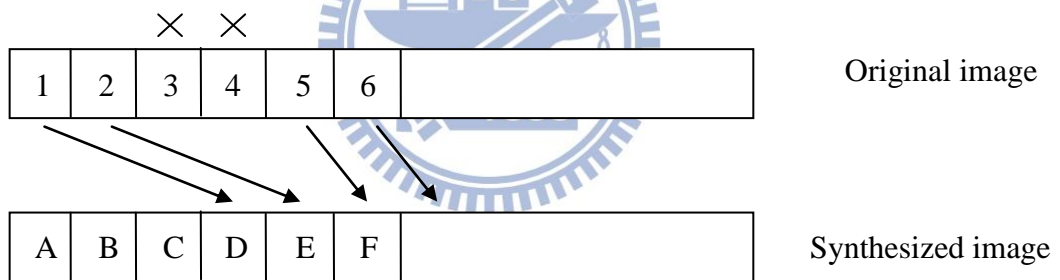


圖 4.11 影像合成規則。

Case 1 中，原始影像的每個 pixel 偏移量都為 0，因此合成出的新影像不會有任何問題，Case 2 中，位置 1、2 的 pixel 偏移量皆為 1，位置 3 到 6 的 pixel 偏移量皆為 3，這在合成出的新影像中會產生空洞(位置 D、E)，我們將取出這些空洞左側有正確位移值的 pixel(位置 2)來填補空洞，因此合成出的新影像中位置 D、E 將會是原始影像位置 2 的值。Case 3 中，位置 1、2 的 pixel 偏移量皆為 3，位置 3 到 6 的 pixel 偏移量皆為 1，這在合成出的新影像中位置 D、E 的地方將會有重複的位移 pixel 來源，由於位移量大的是屬於較靠近的前景部分，位移量小的將是較靠近背景的部分，且理論上背景會被前景所遮蔽，因此新影像中位置 D、E 的 pixel 值將是由位置 1、2 的原始影像而來，而位置 3、4 的 pixel 將不會出現在合成出的新影像中。

4.3 3D 立體顯示器

本論文用來觀看 3D 影像的顯示器是 HTC 生產的 EVO 3D 這款智慧型手機，3D 影片或照片以往都需要配戴偏光式鏡片才能讓眼睛看到個別差異之影像，進而產生出 3D 的立體感，而 HTC EVO 3D 這隻手機主打裸視 3D 技術，觀看者不用再配帶麻煩的 3D 眼鏡，而是可以直接觀看有立體感的相片或影片，下圖 4.12 為手機的外觀



圖 4.12 HTC EVO 3D 手機外觀。

圖 4.12 為 HTC EVO 3D 手機外觀，配備了兩顆水平方向的光學鏡頭，其裸視 3D 是使用視差屏障式 (parallax barriers) 技術，利用螢幕上方的液晶薄膜排列成與雙眼垂直的光學柵欄，這些光學柵欄遮蔽了部分螢幕光線，剩餘的光線由柵欄間隙分別進入左右眼，讓左右眼分別看到不同的畫面產生視差造成立體感，下圖 4.13 為視差屏障原理

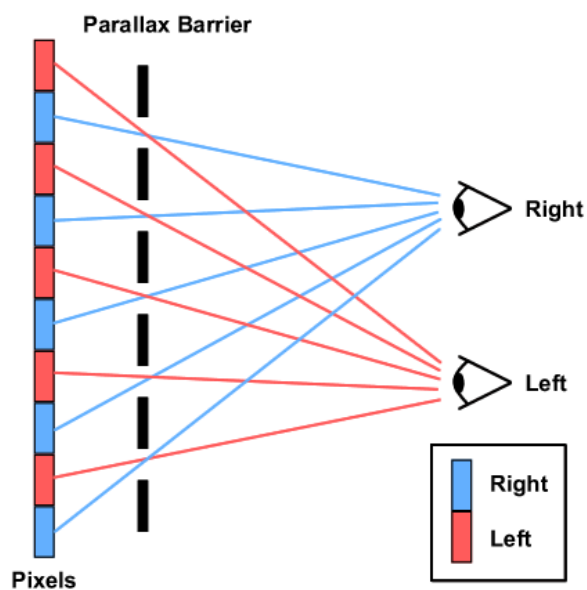


圖 4.13 視差屏障原理。

圖 4.13 為視差屏障原理的說明圖[25]，由於光學柵欄的關係，圖中藍色的 pixel 只進入右眼，紅色的 pixel 只進入左眼，利用此設計讓左右眼分別看到不同的影像產生視差，但因光學柵欄遮蔽了部分的光線，因此會大幅降低 3D 立體顯示器的亮度，且利用此技術的顯示器觀看者必須離顯示器一段適當的距離，才能看出立體感的影像而不致頭暈目眩。



在第三章節與第四章節中已分別介紹了立體視覺與單視域連續影像轉換的技術，接下來將介紹本論文提出的改進深度圖估測概念，說明如何將兩樣技術利用遞迴方法做結合，以得到更佳深度估測結果。首先在 5.1 章節中我們將說明本論文使用的遞迴式深度補償架構流程，接著在 5.2 章節中展示模擬結果與各項參數的設定。

5.1 遞迴式補償架構

本論文中提出了一種遞迴式的方法來補償單視域連續影像的深度估測，下圖為系統流程架構圖：

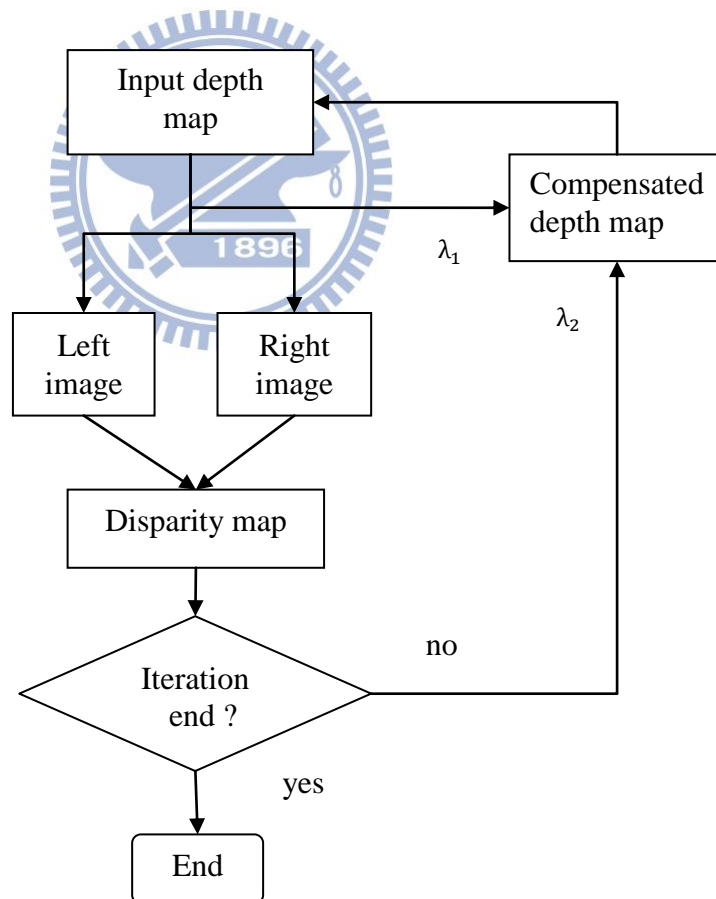


圖 5.1 遞迴式深度補償架構。

圖 5.1 流程中，首先將 Input depth map 利用 DIBR 的技術將原始單視域影像合成出左右眼影像，接著將左右眼影像利用立體視覺演算法計算出 Disparity map，最後把 Input depth

map 與 Disparity map 合併得到補償過後(Compensated depth map)的深度圖，然後進入遞迴運算中。第一張的 Input depth map 是利用各種深度線索做線性加成得到的初步估測深度圖，而之後的 Input depth map 即為進入遞迴運算之後得到的補償深度圖(Compensated depth map)，補償深度圖是由 Input depth map 與 Disparity map 做線性加成得到，即 $\lambda_1 * \text{Input depth map} + \lambda_2 * \text{Disparity map}$ ，在本論文中權重值 λ_1 與 λ_2 分別設定為 0.7 與 0.3。

5.2 參數設置與模擬結果

在本論文中使用的 MB 大小為 8*8 pixel，block matching 所使用的 searching window 大小為 20*20 pixel，原始影像使用 Microsoft 提供的 break dancer 影像，下圖 5.2 為原始影像與各深度線索計算出的深度圖結果



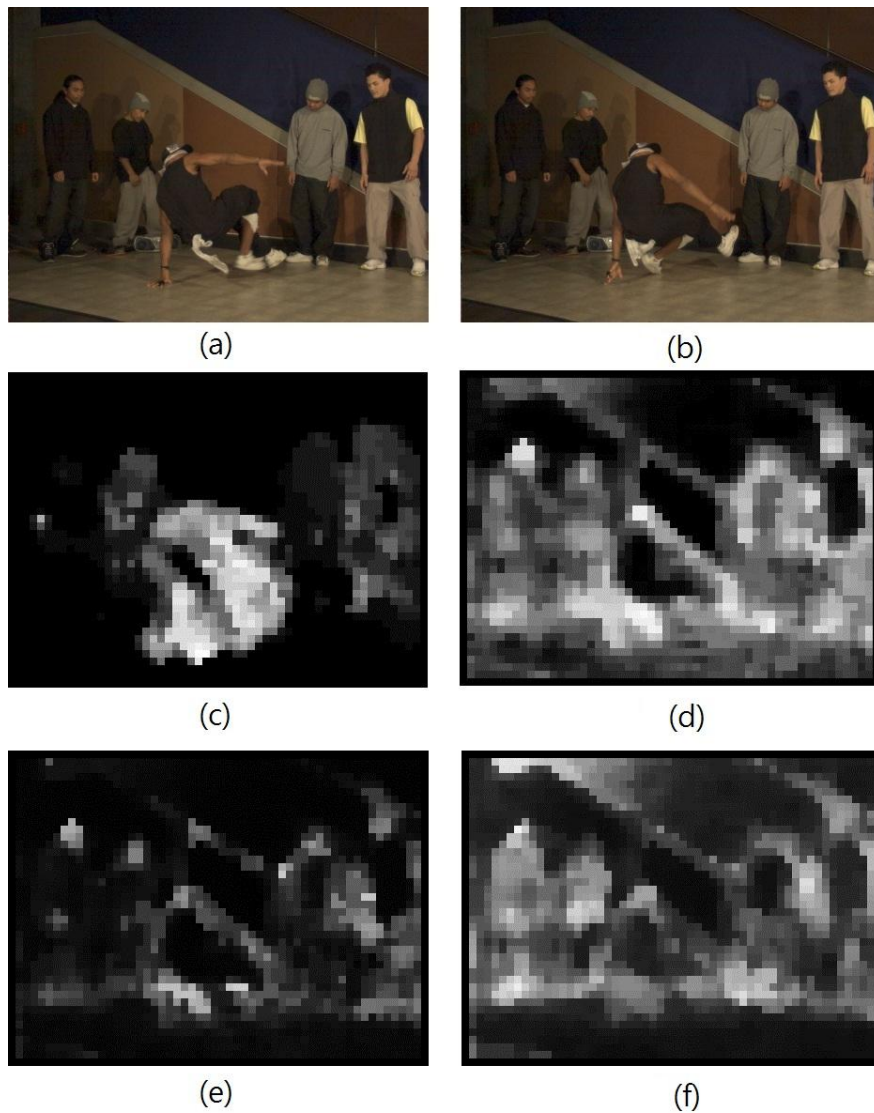


圖 5.2 深度線索計算結果。

圖 5.2 (a)為前一時刻影像，(b)為現在時刻影像，(c)為移動向量計算深度的結果，(d)為材質漸層計算深度的結果，(e)為變異性計算深度的結果，(f)為對比度計算深度的結果，以上結果皆已使用 3×3 MB based(=24*24 pixel based)中值濾波器做濾波。

初步估測的深度圖是利用各深度線索估測出的深度圖分別以 0.25 的權重值線性加成得到，深度圖修正方面則使用相鄰單元標記法做空洞填補以及移除不適當的前景區塊部分，空洞的判定是根據背景區塊的面積大小，本論文中將背景區塊面積小於影像大小的五十分之一的($T_{\text{hole}} = \frac{\text{Width} \times \text{Height}}{50}$)部分當做是存在於物體內部的空洞，將其填補成為前景區塊，而前景區塊面積小於影像大小的百分之一($T_{\text{remove}} = \frac{\text{Width} \times \text{Height}}{100}$)的部分則當作是不適當的前景區塊，將其移除成為背景。初步估測的深度圖如下圖 5.3：

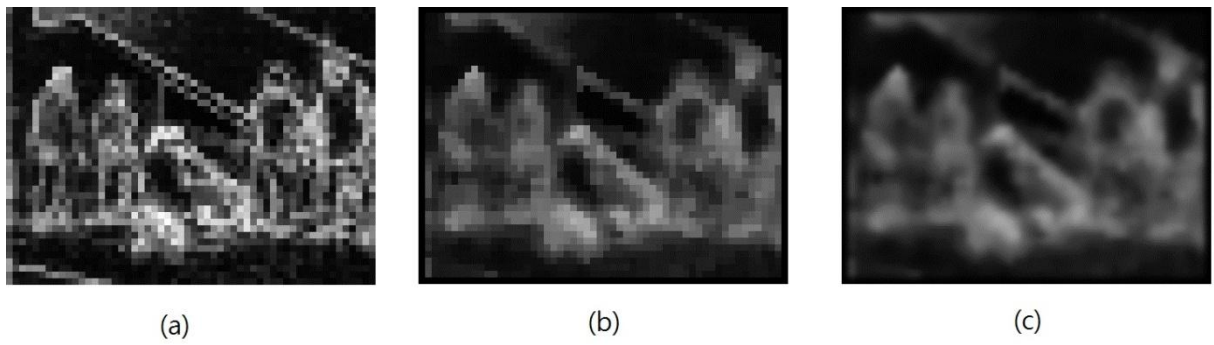


圖 5.3 初步估測的深度圖。

圖 5.3(a)為四樣深度線索線性加成的深度圖，(b)為經過 3×3 MB based 中值濾波器之後的深度圖，(c)為經過 23×23 pixel based 高斯濾波器之後的深度圖。下圖 5.4 為空洞填補以及移除不適當前景區塊的結果

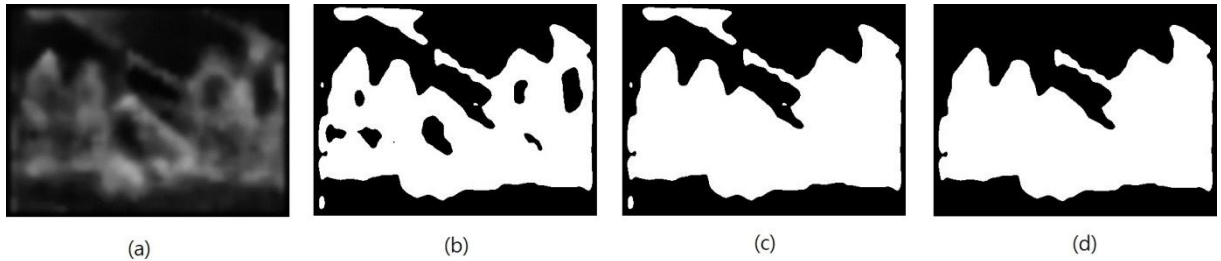


圖 5.4 前景區域修正結果。

圖 5.4(a)為高斯濾波過後的初步深度圖，(b)為使用門檻值 40 做二值化深度圖的結果，白色部分為前景，黑色為背景，(c)為空洞填補的結果，(d)為移除不適當前景區塊的結果。經過前景區域修正之後，我們將針對空洞區塊內插深度值，結果如下圖 5.5

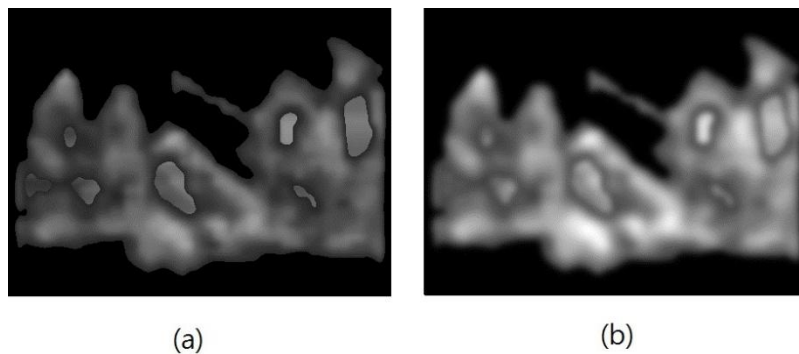


圖 5.5 內插過後的前景深度圖。

圖 5.5(a)為將填補過後的空洞區塊內插深度值的結果，內插方式以該 pixel 為中心取一矩形範圍內的最大深度值做為內插值，矩形範圍的長寬皆為 $2 \times \sqrt{T_{\text{hole}}}$ ，(b)為將內插過後的前景深度圖經過 23×23 pixel based 高斯濾波器之後的結果。將此前景深度圖與計算

出的背景深度圖合併則得到最後的估測深度圖，如下圖 5.6

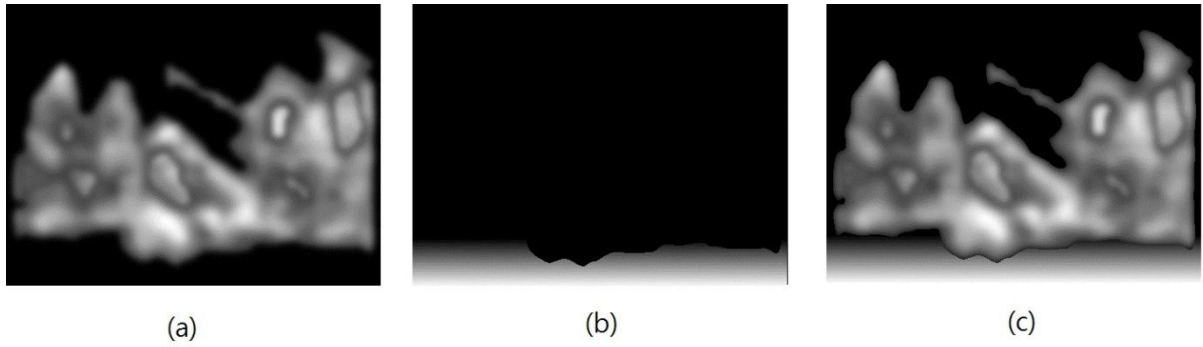


圖 5.6 最後估測深度圖。

圖 5.6(a)為修正過後的前景深度圖，(b)為背景深度圖，(c)為把前景背景合併後得到的最後估測深度圖，此圖將做為本論文深度補償架構中第一次的 Input depth map 使用。

估算好的深度圖可利用 DIBR 的方式合成出左右眼影像對，此左右影像對可接著使用 DP 計算 disparity map，結果如下圖 5.7

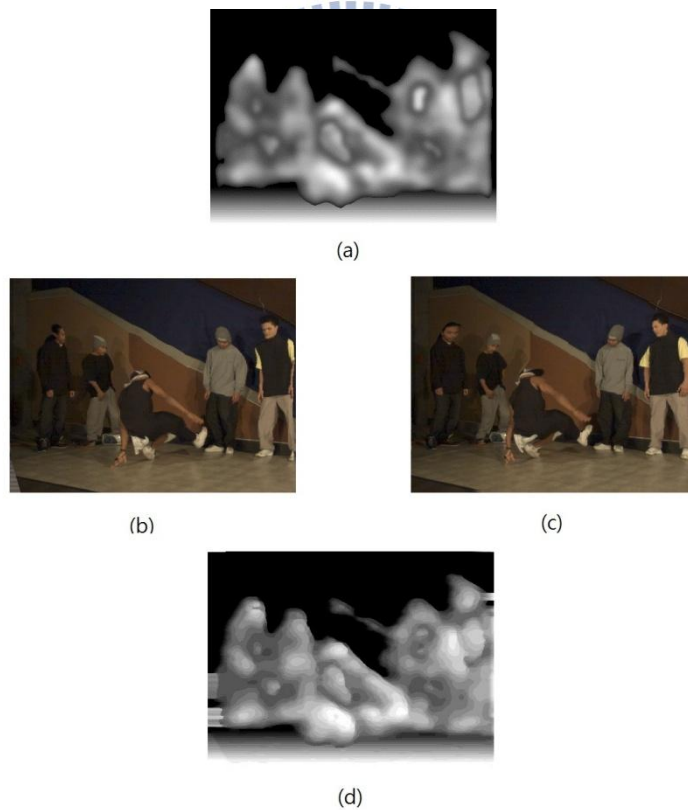


圖 5.7 左右影像對合成與 disparity map 計算結果。

圖 5.7(a)為估測出的深度圖，(b)(c)為利用 DIBR 合成出的左右眼影像，(d)為利用左右眼影像以 DP 計算出的 disparity map，已通過 7×7 中值濾波器做平滑化。此深度圖與 disparity map 將以 λ_1 與 λ_2 的權重值做線性加成，得到一補償過後的深度圖，並做為下一次遞迴運

算的輸入深度圖使用，下圖為補償深度圖的計算結果

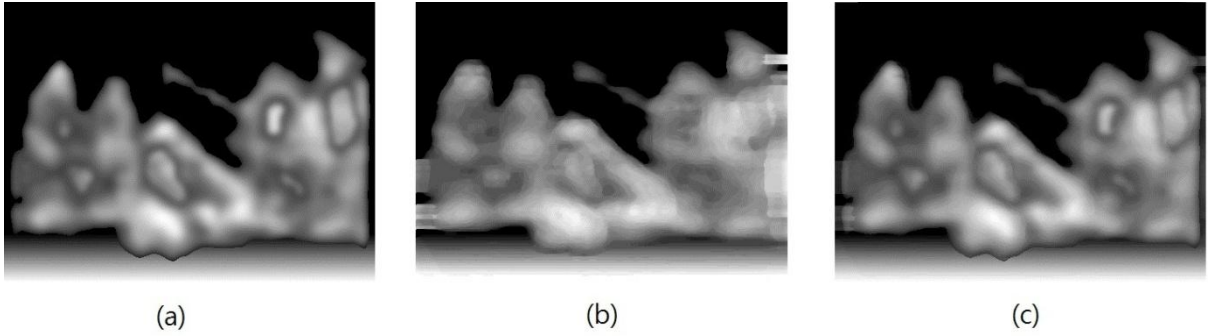


圖 5.8 深度圖補償。

圖 5.8 (a)為輸入深度圖，(b)為利用左右影像算出的 disparity map，(c)為將(a)(b)以 λ_1 與 λ_2 的權重值做線性加成得到的補償深度圖，補償深度圖將做為下一次遞迴運算的輸入深度圖。由於 DP 是以每條 scanline 分別獨立去做運算，為了描述初始估測出的深度圖與用 DP 重新計算得到的 disparity map 的差異，我們挑選幾條 scanline 對深度變化趨勢做細部的觀察，下圖 5.9 為第 150 條 scanline 與第 230 條 scanline 的深度變化趨勢

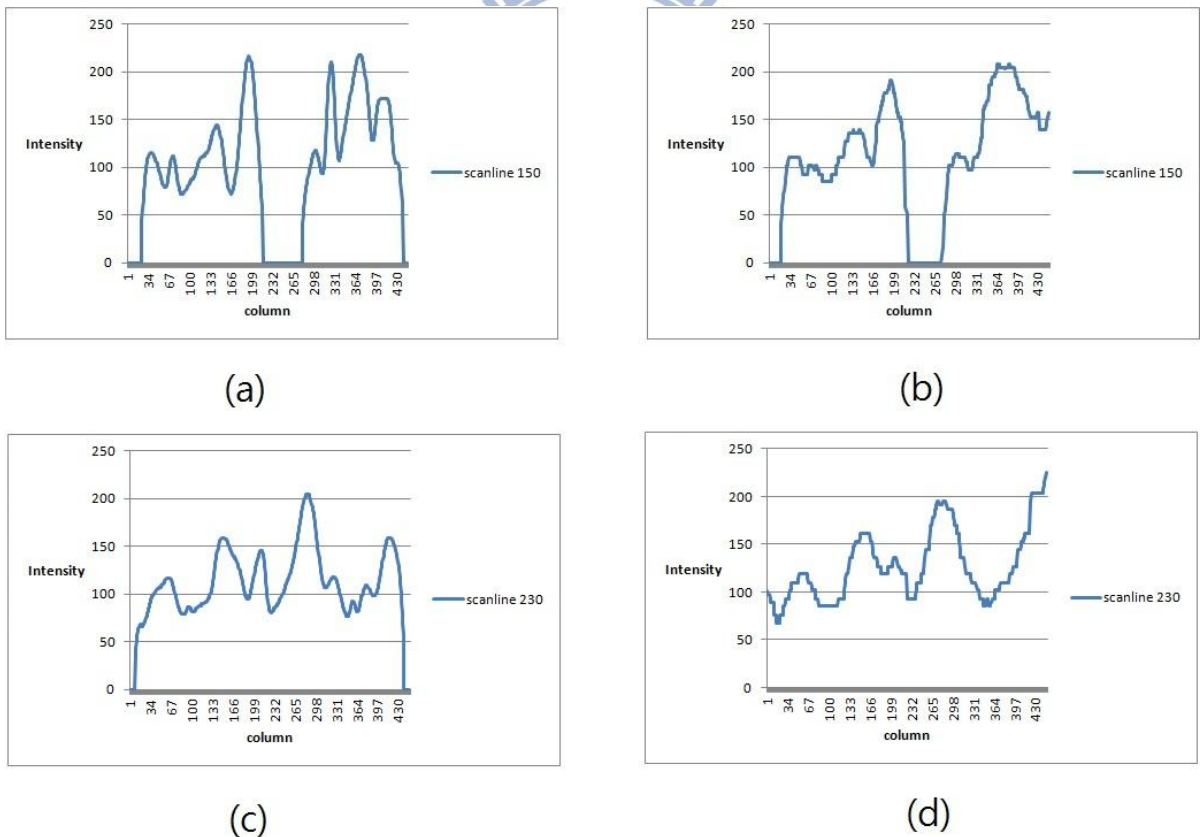


圖 5.9 Scanline 150 與 Scanline 230 的深度分布情況。

圖 5.9 (a)(c)分別為初始估測出的深度圖第 150 條 scanline 與第 230 條 scanline 的分布情

況，(b)(d)分別為 DP 重新計算得到的深度圖第 150 條 scanline 與第 230 條 scanline 的深度分布，其中橫坐標為影像的每一欄，縱坐標為 scanline 對應每一欄的 pixel 深度值，深度值範圍為 0 到 255，由圖中可看出初始估測出的深度圖因為沒有考慮 pixel 點之間的關連性，會產生較多的波峰波谷，也就是較多的深度變化性，因此會使得整體的深度圖起伏較大，較不平滑，而經由 DP 重新計算得到的深度圖因為利用單調次序性(monotonic ordering)與唯一性(uniqueness)兩個限制條件加強了 pixel 之間的關連性，單調次序性限制了匹配路徑只能是一條單調遞減曲線，如果沒有單調次序性的限制將會造成多重對應點的情況發生並產生深度的不連續性，如下圖 5.10

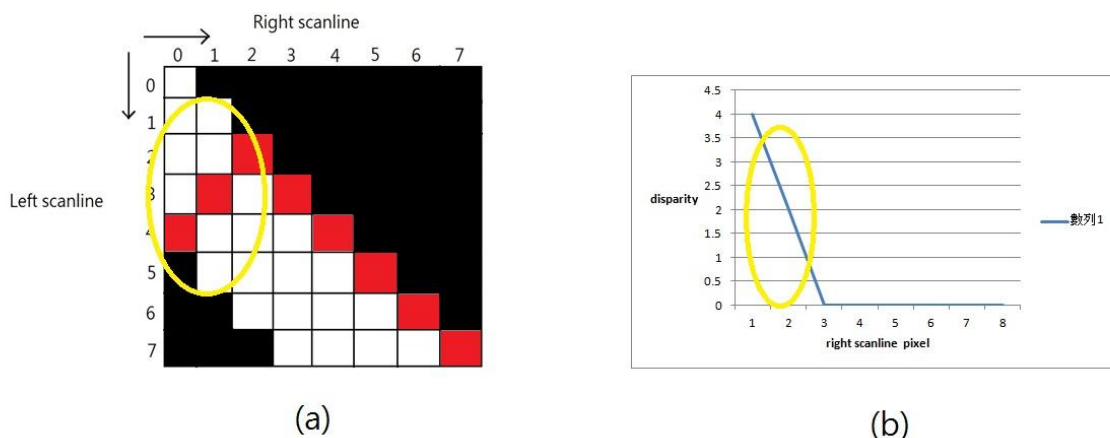


圖 5.10 違反單調次序性。

圖 5.10 (a)紅色方格為違反單調次序性的匹配路徑，(b)為相對應的 disparity 變化，圖中可看出因為違反了單調次序性，pixel 的對應發生 1 對 2 的情況，如圖中黃色圓圈部分，這是不合理的現象，而這種現象也造成了深度的落差變化。同樣的，唯一性限制了 pixel 的對應關係只能是一對一的，一對多或多對一的情況常發生在遮蔽處附近且造成深度的落差，如下圖 5.11 所示

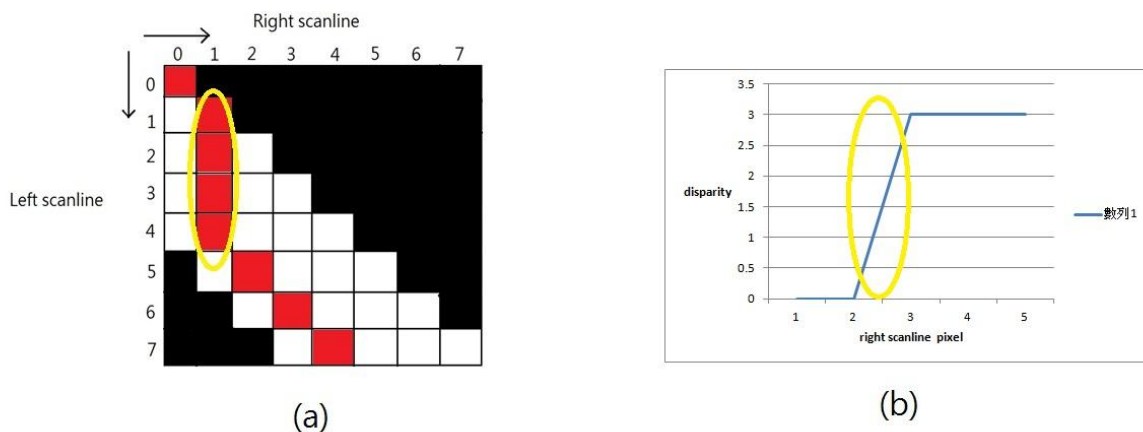


圖 5.11 違反唯一性。

圖 5.11 (a)紅色方格為匹配路徑，(b)為相對應的 disparity 變化，圖中可看出若是因為遮蔽而造成匹配路徑的選擇為水平或垂直方向，這在相對應的 disparity 會產生一個深度落差，如圖中黃色圓圈部分，為此我們將給予遮蔽路徑額外的花費，使得最佳的匹配路徑較容易選擇到斜 45 度方向的路線，而不是水平或垂直路線，這樣便可減少因為非遮蔽情況而引起的深度落差，使得同一前景區塊物體內部得到較為平滑的深度分布，所以由 DP 重新計算過後的深度圖分布是較為平緩的，也就是減少了部分不必要的波峰波谷產生，使其得到較為平滑的深度分布圖，因此我們將利用 DP 計算出的新深度圖做為原始深度圖的修正補償來源，以線性合併的方式做遞迴式的運算。

接下來我們將逐步移除使用的 3 種限制條件，觀察沒有限制條件下與有限制條件下的 DP 運算結果差異為何，如下圖 5.12

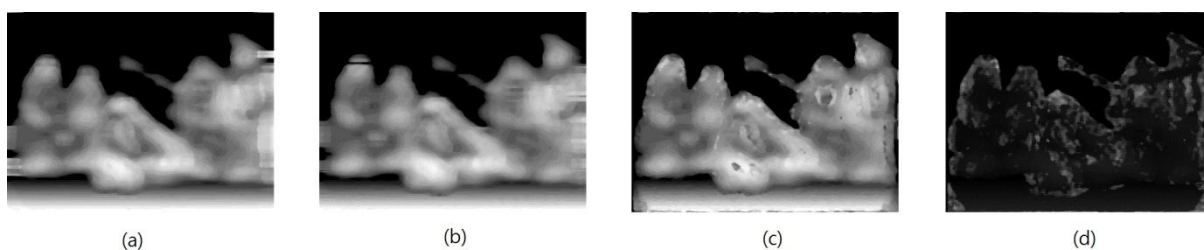
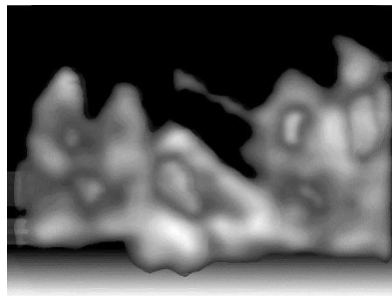


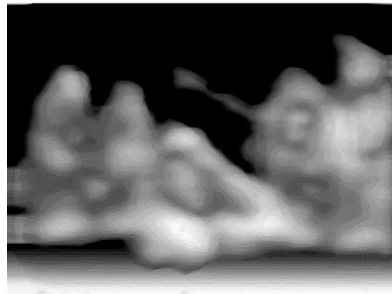
圖 5.12 移除限制條件下的 DP 運算結果。

圖 5.12 (a)為加入了唯一性、單調次序性以及最大 disparity 範圍三種限制條件下 DP 運算的結果，(b)為只有單調次序性以及最大 disparity 範圍兩種限制條件下的運算結果，(c)為只有最大 disparity 範圍一種限制條件的運算結果，(d)為完全不設限制條件的運算結果，由圖(b)中可看出少了唯一性限制條件會造成 DP 的匹配路徑容易在非遮蔽處選擇到水平或垂直的路徑，因此產生較多的深度不連續性，整體深度圖較不平滑，而圖(c)中可看出少了唯一性與單調次序性這兩樣限制條件後，基本上產生了很多交錯匹配點以及非一對一的匹配對應情況，導致深度圖因為深度不連續在同一個前景區塊內產生像是斑塊狀的深度缺陷，最後圖(d)是移除了所有限制條件後的結果，與原本的深度圖已經產生過大差距無法做為比較之用，因此由圖 5.12 的結果可看出本論文所設置的三種限制條件是必要的，不論對於深度的重新估算或是深度平滑上的限制都達到顯著的影響。

下圖 5.13 為遞迴運算的部分結果



(a)



(b)



(c)



(d)

圖 5.13 遞迴補償深度圖。

圖 5.13 (a)(b)(c)(d)分別為遞迴運算 0、3、7、10 次後的補償深度圖。圖中可看出經由遞迴補償的方式，深度圖(a)中因為空洞填補不完全而產生在前景物體內部的黑色環形區塊將得到彌補改善，如下圖 5.14 左紅色線條部分所示

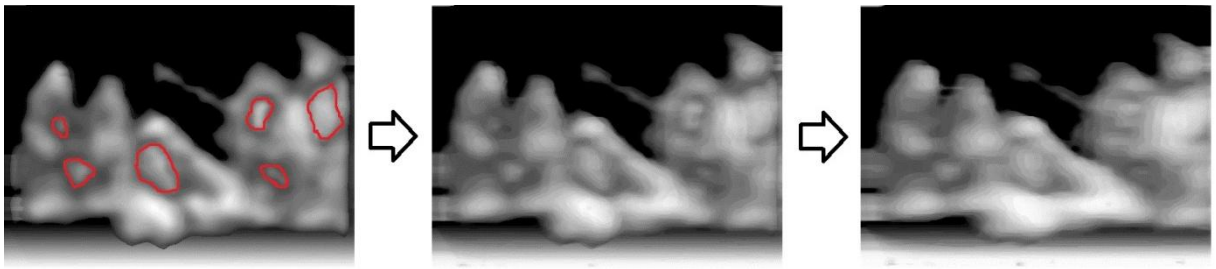


圖 5.14 前景物體內部區域補償。

除了前景內部的補償改善之外，整體的深度圖因為不斷遞迴補償的關係，將逐漸趨於平滑，這在利用 DIBR 合成左右影像並放置到 3D 立體顯示器觀賞時也會得到較舒適的結果。

隨著遞迴次數的增加，我們的遞迴補償法並不會持續的改善深度圖，原因是因為 DP 雖然可將 pixel 間的深度關連性重建，但因為 global optimum 的關係，會使深度圖趨於平滑的狀態，因此在物體邊界(boundary)地帶的深度估測將會是一種破壞的效應，從而繼續的影響到背景部分，如下圖 5.15 黃色圓圈部分

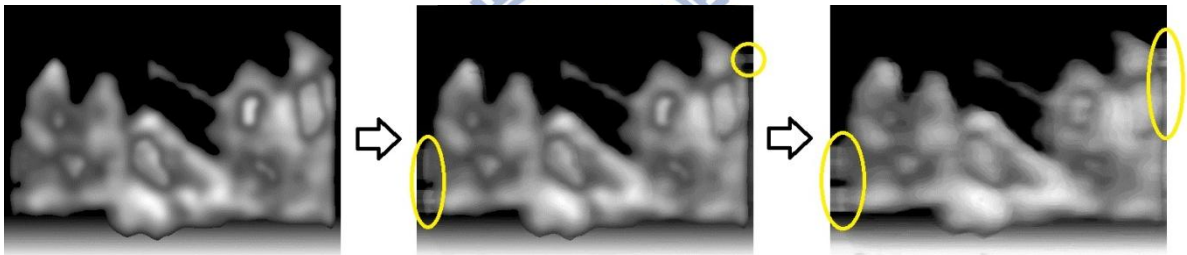


圖 5.15 物體邊界破壞。

所以在遞迴次數少的情況下，前景物體內部的深度補償多於物體邊界以及背景的深度破壞，得到的整體深度圖是改善的，反之，若是遞迴次數持續增多，邊界以及背景的破壞效應大於前景物體內部的補償效果，將得到變糟的深度圖。我們把前景、邊界與背景的分割如下圖 5.16

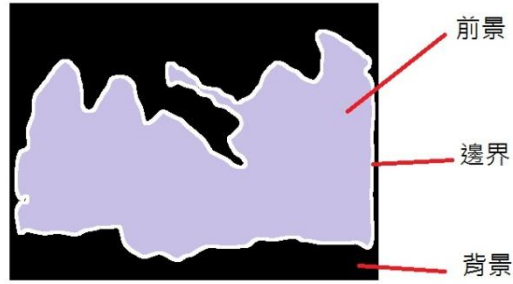


圖 5.16 前景、邊界、背景分割。

上圖 5.16 是以修正過後的前景區塊來做分割，也就是將圖 5.4(d)的邊界部分另外切割出來，以白色部分表示邊界地帶。分割好前景、邊界與背景之後，我們將輸入深度圖(input depth map)與用 DP 計算得到的 disparity map 做影像相減的動作，大於一門檻值則判定此 pixel 經由 DP 重新計算過後的深度是有改變的並以白點表示如下圖 5.17 右

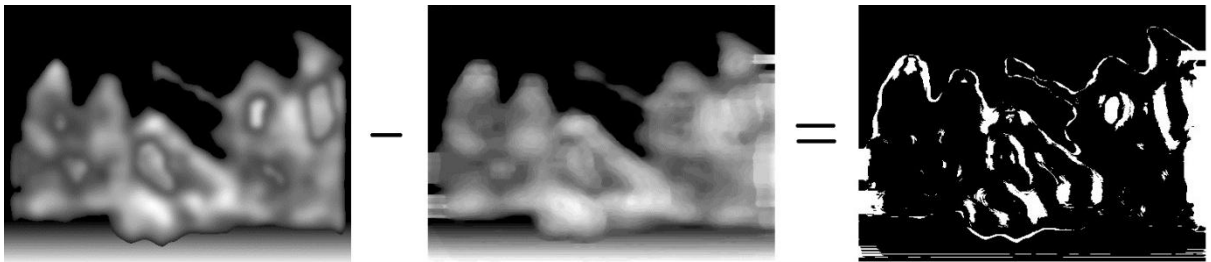


圖 5.17 輸入深度圖與 disparity map 的差異。

將每次遞迴的差異都計算出來如下圖 5.18

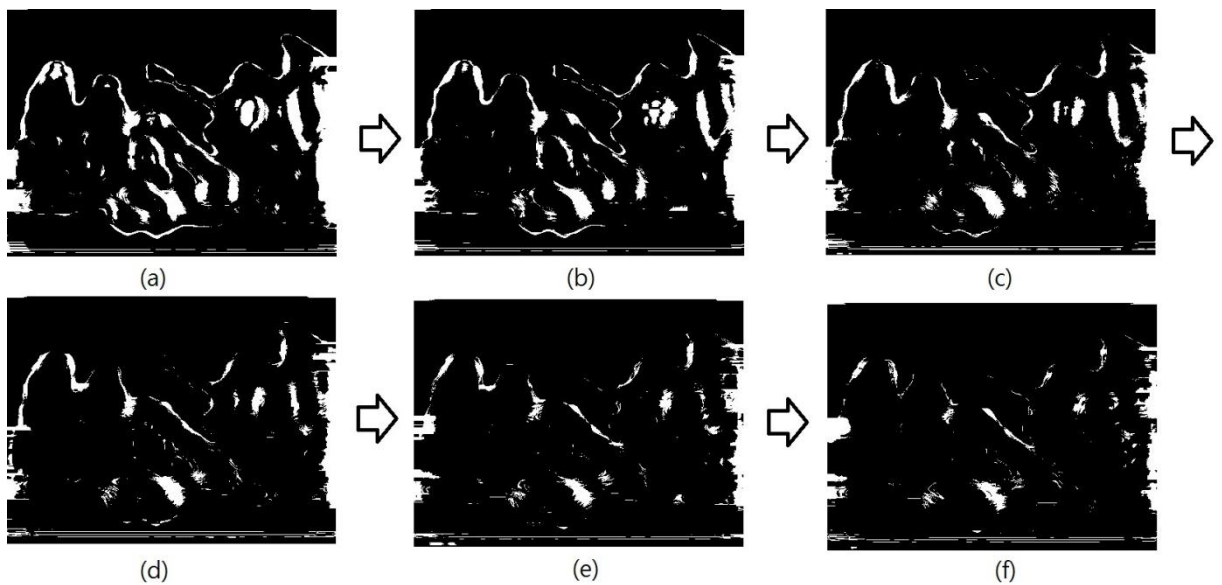


圖 5.18 隨著遞迴次數改變的差異圖。

圖 5.18 的箭頭方向代表遞迴順序，也就是(a)-(f)分別代表遞迴第 0 次的差異圖到遞迴第 5 次的差異圖，由圖中可大致看出差異的部分隨著遞迴次數增加漸漸的在轉移，我們把這些差異點 pixel 位於前景、邊界或者是背景的數量統計出來，以百分比表示如下表 5.1

遞迴次數	改變數量在各部分所佔比重(%)		
	前景	邊界	背景
0	60.0994	22.6251	17.2755
1	53.7477	24.1947	22.0576
2	43.7171	25.0552	31.2276
3	37.3728	25.5718	37.0554
4	35.005	23.6406	41.3544
5	33.8876	20.7641	45.3484

表 5.1 改變量百分比與遞迴次數關係。

從表 5.1 中可看出在前景部分的改變比重隨著遞迴次數上升而減少，這表示經由遞迴補償過後前景深度值是漸趨於利用 DP 做深度重新計算的數值，也就是將圖 5.14 指出的前景區域內部逐漸補償起來，所以之後遞迴可改善的前景部分是愈來愈少的，而邊界部分是隨著遞迴次數上升先增加後減少，這是因為前幾次(表中數據是到第三次前)遞迴中由於 DP 的深度重新估測會在邊界部分造成改變，隨著不斷的改變破壞最後趨於一深度值，因此邊界部分在破壞改變且趨於定值後將造成之後此部分的改變比重減少，至於背景部分本論文是以粗略的方式做估測，消失點以上的背景深度值都給定為零，因此在 DP 的深度平滑化效應下會將部分前景深度值融合進背景區域，造成背景深度值持續的被破壞改變，所以背景的改變比重是隨著遞迴次數上升不斷的增加的。經由上述實驗數據顯示，利用本論文的遞迴方法重新對深度圖做估測補償必須在限制的遞迴次數下才能得到較好的改善結果，否則盲目的增加遞迴次數只會對背景造成更多的破壞效應，在我們的實驗參數設定下，遞迴補償次數設定在不超過 5 次的情況下皆可改進深度圖估測。

我們把文獻[26]利用立體視覺方式計算出的深度圖做為比較對象(ground truth)，設定不同初始深度線索的權重值(α_1 、 α_2 、 α_3 與 α_4)組合得到不同的初步深度圖，並且計算其與 ground truth 的差異如下表 5.2

移動向量 (α_1)	Edge 密度 (α_2)	變異性(α_3)	對比度(α_4)	MSE(mean squared error)

0.25	0.25	0.25	0.25	4110.49
0.3	0.3	0.2	0.2	4064.37
0.4	0.2	0.2	0.2	4375.43
0.5	0.2	0.2	0.1	3871.92
0.6	0.2	0.1	0.1	3918.11
0.7	0.1	0.1	0.1	6394.05
0.3	0.4	0.2	0.1	4226.78
0.2	0.5	0.2	0.1	4598.8
0.2	0.1	0.4	0.3	4507.01
0.1	0.2	0.3	0.4	4271.48
0.1	0.1	0.2	0.6	4027.21
0.2	0.1	0.6	0.1	5936.5

表 5.2 不同深度線索得到的初步深度圖與 ground truth 的相似度比較。

由表 5.2 可看出不同的深度線索權重組合將對初始深度圖造成不同的影響，實在很難決定出所謂一組最佳的權重組合，但是根據我們使用的 data source 特性，應該分配移動向量(α_1)較高的權重，而由表中也可看出我們分配的權重組合中 MSE 最小的一組為 $\alpha_1 = 0.5$ 、 $\alpha_2 = 0.2$ 、 $\alpha_3 = 0.2$ 與 $\alpha_4 = 0.1$ 。

為了凸顯本論文遞迴補償架構的深度圖改善效益，我們依據 ground truth 其不同的深度區間劃分為 6 個深度 section 如下圖 5.19

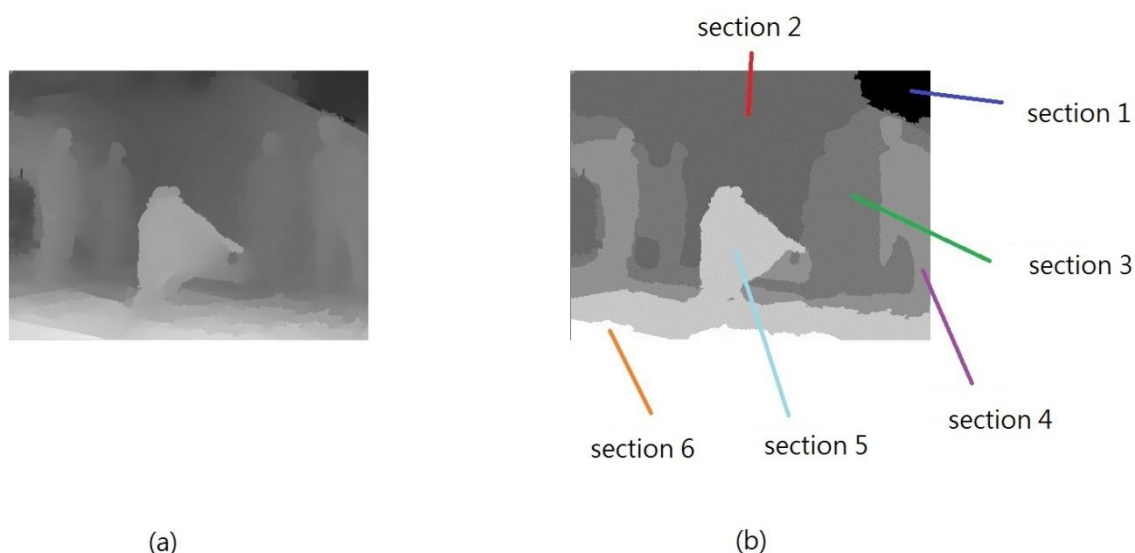
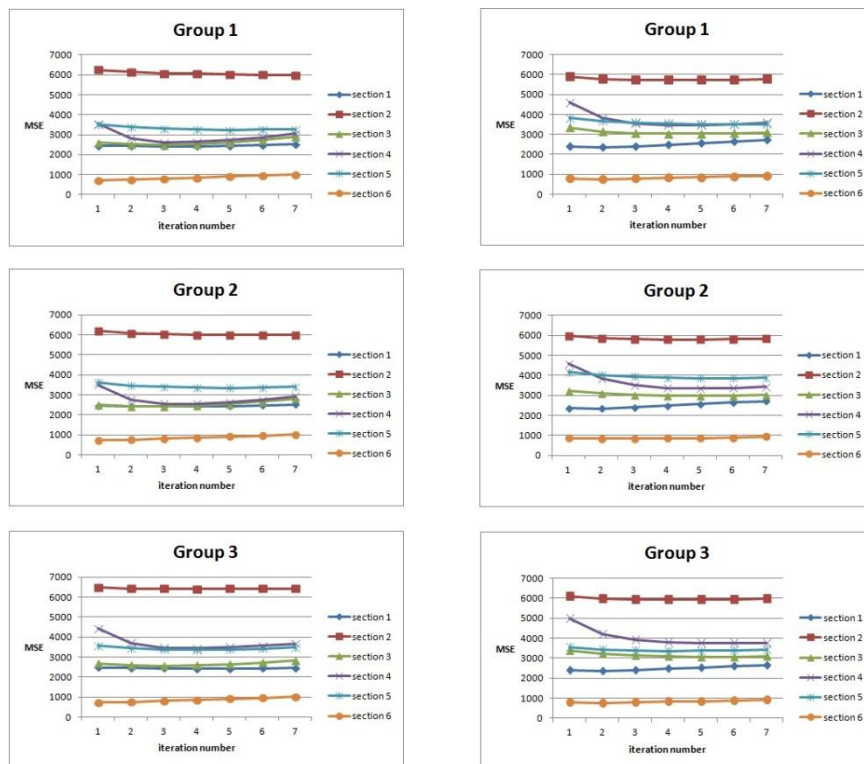


圖 5.19 深度區間劃分。

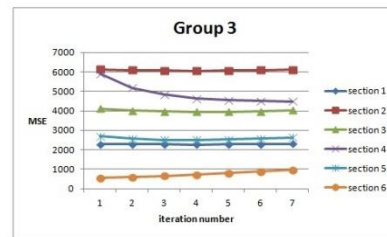
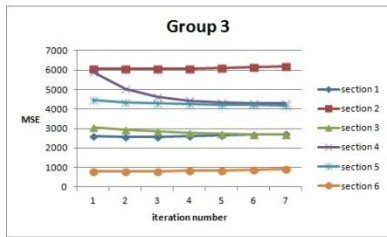
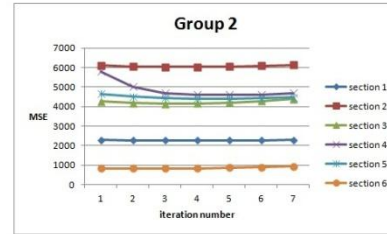
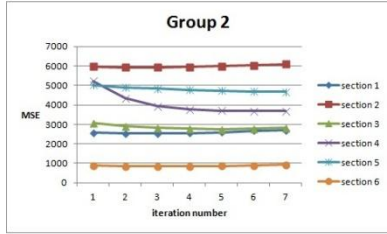
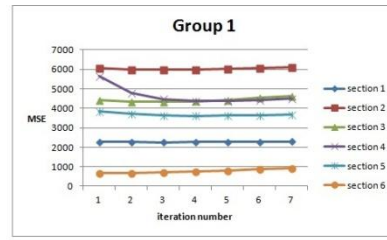
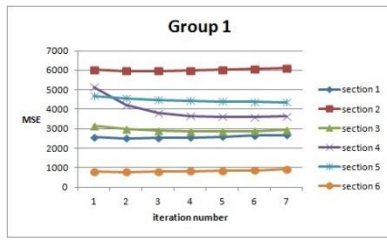
圖 5.19 (a)為文獻中拿來做比較的深度圖，(b)為將(a)依據不同的深度區間劃分成 6 個

section, section 1 為深度值 0 到 60, section 2 為深度值 60 到 105, section 3 為深度值 105 到 120, section 4 為深度值 120 到 145, section 5 為深度值 145 到 200, section 6 為深度值 200 到 255。在此我們使用 MSE(Mean Squared Error)來計算我們的遞迴補償深度圖與文獻中深度圖的相似度,並且使用了 3 種不同的深度線索權重線性組合得到 3 種不同的初步深度圖作為比較用,假設 α_1 、 α_2 、 α_3 與 α_4 分別為移動向量、edge 密度、變異性與對比度的權重值,那麼我們設計的第一種組合(Group 1)為 $\alpha_1 = \alpha_2 = \alpha_3 = \alpha_4 = 0.25$,第二種組合(Group 2)為 $\alpha_1 = \alpha_2 = 0.3$, $\alpha_3 = \alpha_4 = 0.2$,第三種組合(Group 3)為 $\alpha_1 = 0.4$, $\alpha_2 = \alpha_3 = \alpha_4 = 0.2$,下圖 5.20 為本論文的遞迴深度圖與文獻 ground truth 的比較結果



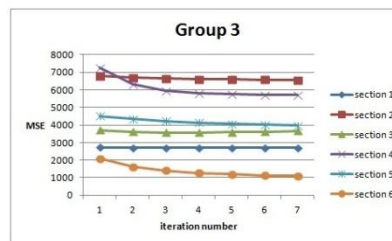
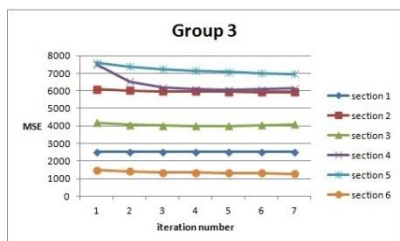
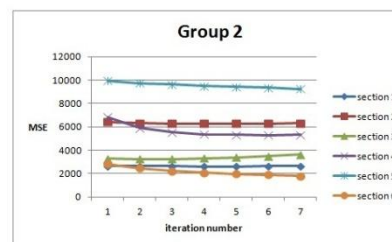
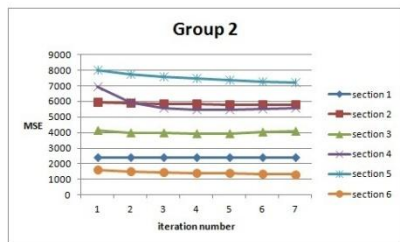
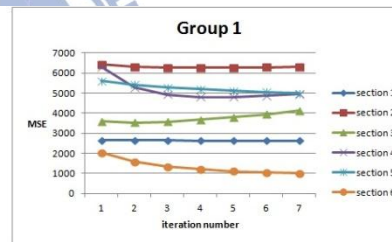
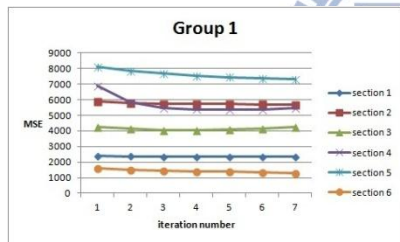
(a)

(b)



(c)

(d)



(e)

(f)

圖 5.20 遞迴深度圖與文獻深度圖的比較結果。

圖 5.20 中，橫坐標疊代次數(iteration number)的第一次(iteration number 1)為初步深度圖

與 ground truth 的 MSE，第二次之後才是論文架構的補償深度圖(compensated depth map) 與 ground truth 的 MSE。其中(a)-(f)分別為 data sequence 中不同張 Frame 的計算結果，(a) 為 Frame 1、(b)為 Frame 4、(c)為 Frame 7、(d)為 Frame 10、(e)為 Frame 15、(f)為 Frame 20。由圖 5.20 可看出，由於每張 Frame 中的景象不相同，造成 MSE 與遞迴次數的關係曲線也不盡相同，不過從中可以觀察到的是經由本論文的遞迴補償方式，不論初始深度線索的不同或是畫面不同(不同張 Frame)，位於中間深度的區塊(section 3、4、5)大部分其 MSE 都隨著遞迴次數增加而獲得改善(MSE 呈現下降曲線)，尤其是深度 section 4 改善的部分最多。Section 1、2 為較靠近背景的部分，由於本論文對於背景方面的估測只是簡單的使用猜測出的消失點給予梯度深度值，消失點以上的深度值都設為 0，而消失點選取為最遠物體的底部附近，因此 section 1、2 幾乎都在深度 0 的範圍，所以這些部分基本上使用遞迴方法也無法獲得明顯的改善。Section 6 為離我們最近的區塊，基本上大部分是地板部分，這區域是本論文給定背景梯度深度值的地方，經由遞迴運算後基本上變化也不大，因此較無參考價值。Section 3、4、5 為最主要前景物體的區塊，由上面實驗數據結果顯示本論文的遞迴補償方法是可以改善前景區塊部分的深度估測的，並在對其他區塊影響小的情況下將深度估測結果逼近 ground truth。

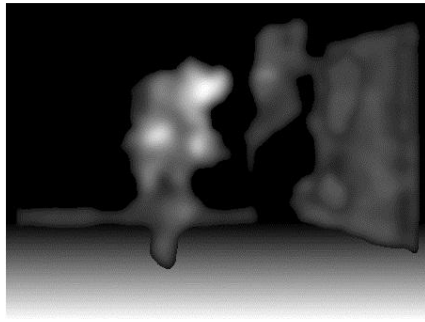
接下來將展示其他測試影像的模擬結果，下圖 5.21 為 Ballet 測試影像的實驗結果



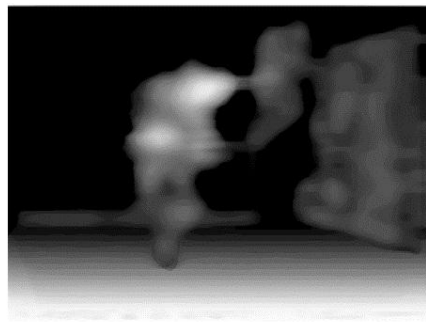
(a)



(b)



(c)



(d)



(e)



(f)

圖 5.21 “Ballet” 測試影像模擬結果。

圖 5.21 (a)為前一時刻影像，(b)為現在時刻影像，(c)為初步估測出的深度圖，(d)為經過 3 次遞迴補償之後的補償深度圖，(e)(f)為利用 DIBR 合成出的左右眼影像，由於此測試影像主要的 motion 只有在跳舞的人，距離畫面較近的人在聽音樂，只有頭部附近輕微的晃動，因此聽音樂的人下半身並未計算成前景區塊，此外由於畫面右方的牆壁有許多 edge 資訊在，所以被誤判為前景區域而沒被消除，關於這個測試影像可能得加入其他深度線索估算才能得到較準確的初步深度圖，不過由圖(d)依然可看出本論文的遞迴補償法依然能夠改善前景物體內部的深度不連續性。下圖 5.22 為測試影像 Akko and Kayo 的模擬結果

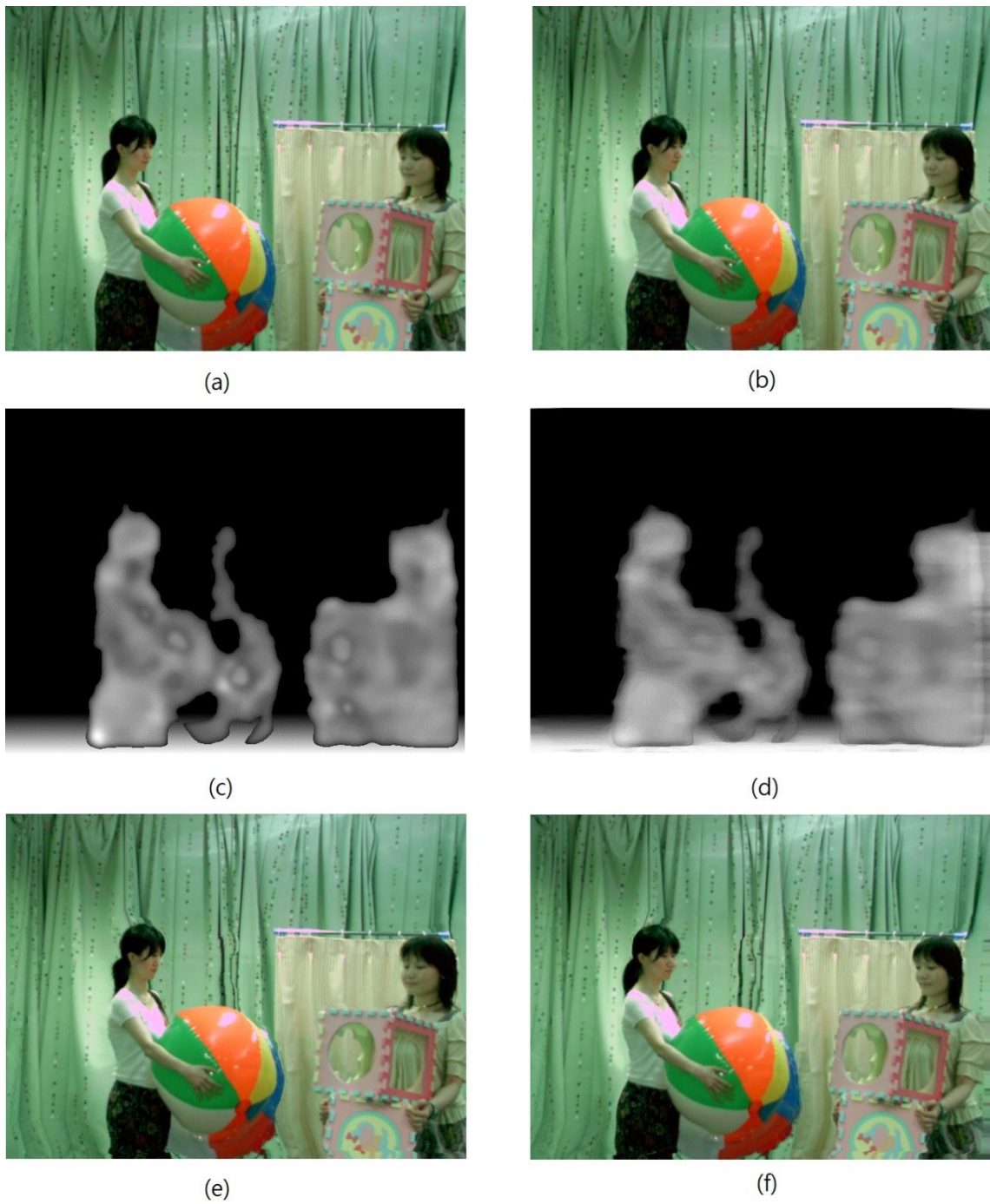


圖 5.22 “Akko and Kayo” 測試影像模擬結果。

圖 5.22 (a)為前一時刻影像，(b)為現在時刻影像，(c)為初步估測出的深度圖，(d)為經過 3 次遞迴補償之後的補償深度圖，(e)(f)為利用 DIBR 合成出的左右眼影像，由圖(d)可看出經過遞迴補償之後的深度圖較初步估測出的深度圖(圖(c))為平滑，用 DIBR 合成左右眼影像時也將得到較好的 3D 立體效果。

單視域影像的深度估測方法存在其困難性，原因歸咎於有限的深度線索，為了提高其正確性，計算連續兩張影像間的移動資訊將是重要的深度線索之一，然而大部分的單視域深度估測方法是將這些深度線索做線性加成，且權重的選擇將是一困難的問題，本論文提出的遞迴式補償方法目的在於不論初始深度線索權重組合為何，都能利用 DP 演算法重新將像素點間的關連性建立起來，改善因為過於平滑的物體內部的深度不連續性分布，並利用遞迴式的方法不斷對深度圖的估測做補償。模擬實驗結果顯示本論文提出的遞迴式補償方法確實能夠改善前景物體深度估測值，雖然不同的深度權重線性組合將得到不同的初始深度圖，但是經由不斷的疊代補償都將得到更好的深度估測圖，對於合成出的 3D 立體影像對也將得到較適宜的觀賞效果。

由於本論文著重在前景物體深度估測上，背景深度的計算並沒有使用較多的處理步驟，而是只給予簡單的深度值指派，因此在背景深度的改進上將有所限制，之後將針對背景深度的部分做較縝密的估測，改進整體的深度圖，此外，深度線索的增加也將有助於改善初步的深度圖估測，未來將期許完成這些不足的部分，並做為之後的整體系統效能改進。



參考文獻

- [1] Man-Bae Kim, Mun-Sup Song, Do-Kyoon Kim, and Kwang-Chul Choi, “Stereoscopic conversion of monoscopic video by the transformation of vertical-to-horizontal disparity,” in Proc. *SPIE*, vol. 3295, pp. 65-75, 1998.
- [2] I. Ideses, L. P. Yaroslavsky, and B. Fishbain, “Real-time 2D to 3D video conversion” , *Journal of Real-Time Image Processing*, vol. 2, no.1, pp. 3-9, 2007.
- [3] Yu-Lin Chang, Chih-Ying Fang, Li-Fu Ding, Shao-Yi Chen and Liang-Gee Chen, “Depth Map Generation for 2D-to-3D Conversion by Short-Term Motion Assisted Color Segmentation,” in Proc. *IEEE International Conference on Multimedia and Expo*, pp.1958 –1961, 2007.
- [4] Cheng-Ying Yeh, “Stereoscopic Conversion Technique Based on Monocular View Source”, National Chung Cheng University, master thesis, 2008.
- [5] S. Battiato, A. capra, S. Curti, and M. L. Cascia, “3D stereoscopic pairs by depth-map generation,” Proceeding of the 2nd *International Symposium on 3D Data Processing, Visualization, and Transmission*, pp. 124-131, 2004.
- [6] D. Scharstein and R. Szeliski, “A taxonomy and evaluation of dense two-frame stereo correspondence algorithms” , *IJCV*, vol. 47, no. 1-3, pp.7-42, 2002.
- [7] http://en.wikipedia.org/wiki/Time-of-flight_camera
- [8] <http://en.wikipedia.org/wiki/ZCam>
- [9] <http://ntuzhchen.blogspot.tw/2011/04/3dv-systems-zcam-depth-camera.html>
- [10] J.Geng, “Structured-light 3D surface imaging: a tutorial,” *Advances in Optics and Photonics*, Vol. 3, Issue 2, pp. 128-160, 2011.
- [11] http://campar.in.tum.de/twiki/pub/Chair/TeachingSs11Kinect/2011-DSensors_LabCourse_Kinect.pdf
- [12] http://www.cse.iitb.ac.in/~niketbagwe/mtech_project/seminar.pdf
- [13] http://www.google.com.tw/books?hl=zh-TW&lr=&id=G8Ua-YTU4CoC&oi=fnd&pg=PR7&dq=kinect+infrared+dots&ots=WsP8sfCO8G&sig=6GuSmRgtdCPpJEMGqiN-jKaXdtY&redir_esc=y#v=onepage&q=kinect%20infrared%20dots&f=false
- [14] <http://azttm.wordpress.com/2011/04/03/kinect-pattern-uncovered/>
- [15] <http://www.stemmer-imaging.co.uk/en/news/5247-LMI-Gocator-3D-sensor-technology-as-strategic-element>
- [16] 謝定龔, “CDMA 手機定位研究與實作”, 國立中央大學資訊工程研究所碩士論

文，2006 年。

- [17] A.H. Sayed, A. Tarighat, and N. Khajehnouri, "Network-based wireless location," *IEEE Signal Processing Magazine*, pp 24-40, July 2005.
- [18] A. Fusiello, E. Trucco, and A. Verri, "A compact algorithm for rectification of stereo pairs," *Machine Vision and Applications*, vol. 12, no. 1, pp. 16–22, 2000.
- [19] http://en.wikipedia.org/wiki/Dynamic_programming
- [20] I. J. Cox, S. L. Hingorani, S. B. Rao, and B. Maggs, "A maximum likelihood stereo algorithm", *Computer Vision and Image Understanding*, vol.63, no 3, pp. 542–567, 1996.
- [21] A. Criminisi, J. Shotton, A. Blake, C. Rother, and P. H. S. Torr. "Efficient dense stereo with occlusions for new view-synthesis by four-state dynamic programming," *IJCV*, vol. 71, no. 1, pp.89–110, Jan 2007.
- [22] H. Gharavi and M. Mills, "Block-matching motion estimation algorithms: New results," *IEEE Trans. on Circ. and Syst.*, Vol. 37, pp. 649-651, 1990.
- [23] 陳明民，數位立體影像之理論探討與創作實驗之研究，國立嘉義大學碩士論文，2005 年。
- [24] http://en.wikipedia.org/wiki/Connected-component_labeling
- [25] <http://m.eprice.com.tw/mobile/talk/4205/4665996/>
- [26] C.L. Zitnick, S.B. Kang, M. Uyttendaele, S. Winder, and R. Szeliski, "High-quality video view interpolation using a layered representation," *ACM SIGGRAPH and ACM Trans. on Graphics*, Los Angeles, CA, pp. 600-608, Aug. 2004.
- [27] <http://www.tanimoto.nuee.nagoya-u.ac.jp/~fukushima/mpegftv/Akko.htm>