

國立交通大學

電子工程學系 電子研究所碩士班

碩 士 論 文

基於正逆向深度映射演算法下的

稀疏多相機虛擬視點合成

Sparse Multi-Camera Virtual View Synthesis Using
Forward and Backward Depth Warping Algorithms

研 究 生：李讀修

指導教授：杭學鳴 教授

中 華 民 國 一 〇 一 年 七 月

基於正逆向深度映射演算法下的稀疏多相機虛擬視點合成
**Sparse Multi-Camera Virtual View Synthesis Using Forward and
Backward Depth Warping Algorithms**

研究生: 李讀修

Student: Du-Hsiu Li

指導教授: 杭學鳴 教授

Advisor: Prof. Hsueh-Ming Hang

國立交通大學

電子工程學系 電子研究所碩士班

碩士論文

A Thesis

Submitted to Department of Electronics Engineering and

Institute of Electronics

College of Electrical and Computer Engineering

National Chiao Tung University

In Partial Fulfillment of the Requirements

for the Degree of

Master of Science

in

Electronics Engineering

March 2012

Hsinchu, Taiwan, Republic of China

中華民國一〇一年七月

基於正逆向深度映射演算法下的稀疏多相機虛擬視點合成

研究生：李讀修

指導教授：杭學鳴 教授

國立交通大學

電子工程學系 電子研究所碩士班

摘要

近年來，互動式多視角視頻 (Interactive Multiple View Video) 在多媒體領域上已經是一個熱門的研究方向之一。因為擁有廣泛應用的潛力、以及大量的市場需求，國際標準會議遂制訂立體視訊編碼 (Three-Dimension Video Coding, 簡稱 3DVC) 相關的標準競賽。

然而目前的立體視訊編碼只針對於密集且平行相機陣列系統，其相鄰相機間的距離不會超過 10 公分。本篇論文係打破此一距離上的限制，以稀疏且非平行相機陣列為重心，發展一套稀疏多相機虛擬視點合成系統。本篇論文的目標在於藉由已知的稀疏影像，合成出虛擬視角。一開始會先分析目前現有合成系統中的正向景深映射，並探討映射後所產生的各種瑕疵現象成因，而提出一套景深修復方法來做改進。

接著我們根據現有的自由視點架構，一共改進四個地方，分別是**景深圖升取樣方法**、**逆向景深圖映射演算法**、**金字塔型影像修補技術**、以及**瑕疵抑制方法**。景深圖對於升取樣中，我們經由實驗發現複本插值法的效果比一般插值法還佳。而目前現有的正向景深圖映射演算法會產生大量的瑕疵現象，利用一些幾何關係，我們發展一套**逆向景深圖映射演算法**，可大幅降低映射後的瑕疵現象。再來是**虛擬視點合成後所產生的遮蔽區域**，傳統上所使用的影像修補技術對於雜訊過於敏感，因此我們提出**金字塔型影像修補**

技術，以彌補其缺陷。最後本論文則提供一個簡單的演算法，抑制合成影像的瑕疵，使人眼主觀感受上能更加生動自然。

以上演算法皆會利用三台稀疏陣列相機所拍攝的影像來做測試。實驗結果展示出每項技術對於最後的虛擬視點合成，其主觀品質皆有不錯的改善。



Sparse Multi-Camera Virtual View Synthesis Using Forward and Backward Depth Warping Algorithms

Student: Du-Hsiu Li

Advisor: Prof. Hsueh-Ming Hang

Department of Electrical Engineering &
Institute of Electronics
National Chiao Tung University

Abstract

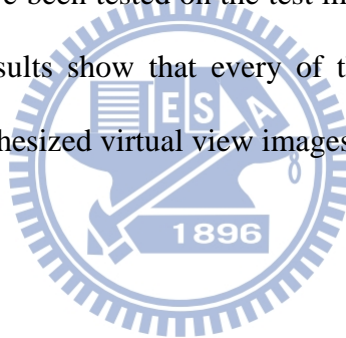
Recently, multiview video technology receives a lot of attention. Because of its potential wide applications and large market, the so-called free-view system or virtual view system becomes a standardization item (3DVC) of the international MPEG committee.

The current focus of MPEG 3DVC project is on the parallel and dense camera array system. The distance between two nearby cameras is about of less than 10 cm. In contrast, our focus in this study is on the so-called sparse multi camera systems, of which the cameras are located farther away. Our target is to synthesize a virtual view based on the recorded sparse camera pictures. We first study the causes of the depth map artifacts produced in the process of forward warping. And then to reduce these artifacts and the artifacts in the texture image synthesized stage, we propose a few refinement tools described below.

Four techniques have been developed and presented in this thesis. They are **depth map**

up-sampling, backward depth map warping, pyramid-based hole filling, and post artifact reduction. At the up-sampling stage, we find empirically that the duplicated interpolation produces better depth map as compared to the other interpolation methods. The ordinary forward warping produces many types of artifacts. We propose an instrumental backward warping algorithm on the depth map. It is able to reduce most artifacts in depth warping due to the more accurate geometric relations. The occlusion regions can be eliminated by using our proposed pyramid-based hole filling method. One of its nice features is that it can suppress the noise when filling in the occlusion area. Finally, with the aid of artifact reduction techniques, the synthesized virtual view is more vivid and natural.

All the above techniques have been tested on the test images captured by a set of sparsely located 3-camera array. The results show that every of them can prove visible subjective quality improvement on the synthesized virtual view images.



誌謝

在研究所生涯中，首先我想感謝指導老師杭學鳴教授。老師從不會給予我太大的壓力，在這兩年的努力中，老師就像一盞明燈，讓我放手實現想法之餘，在旁給予適時的建議，使我研究可以順利發展下去；並且老師會以鼓勵代替責備，對於研究上的問題切入點也是多方且簡明扼要，不僅讓我在挫折中找回自信，也給了我其它想法之契機。在生活機能上老師也會像家人一樣地關切我；不僅如此，我也從老師身上學到很多做人處事的謙卑態度，以及同濟間互助合作的精神，而不單只一頭栽進學問中。我非常慶幸能當老師的指導學生。

接著我想感謝彰哲學長和鴻志學長，雖然在實驗室的相處時間不多，但在研究上也給了我一些實用的知識和建議，而且在求職上也給了不少寶貴的建議。我也很感謝朝雄學長、峻利學長、和宸銜學長，他們不僅在專業知識給予我寶貴的意見，實驗室很多瑣碎的雜事與管理也都是他們一手包辦，使我可以專心於研究上。也很感謝學長姊們：兆軒、書緯、偵源、復凱、曉盈、俊言、威宇、智凱、卓翰、頌文、郁婷、怡茹、琬瑜、Chandan；以及同學和學弟們：凱翔、義文、維哲、士傑、晉源、基峰、政憲、長廷、鈞凱、哲偉、...等，使我研究生涯上不孤單，生活過得多采多姿。

最後我非常感謝父母親、哥哥、和姊姊，在背後不斷地陪伴與支持我，而得以踏上交大，遇上很多指導過我的師長們、以及一群優秀的同儕。尤其是父母親，辛苦地將我養育成人，也從小培育我很多做人處事的道理。在致謝之餘，也希望家人們身體健康、事事順心，如是我才能無畏地勇往直前。

讀修 謹誌於

國立交通大學電子研究所

中華民國一〇一年七月

目錄

摘要	i
Abstract.....	iii
誌謝	v
目錄	vi
圖目錄	xi
表目錄	xvi
Chapter 1 緒論 (Introduction)	1
1.1 研究背景 (Background).....	1
1.2 研究動機及貢獻 (Motivations and Contributions)	2
1.3 論文大綱 (Organization of the Thesis)	3
Chapter 2 立體影像系統基礎知識 (Stereo Image System Basics).....	4
2.1 立體成像系統導論 (Introduction of Stereo Image System).....	4
2.1.1 投影空間 (Projective Space)	4
2.1.2 針孔相機模型 (Pinhole Camera Model).....	6
2.1.3 極平面幾何概述 (Overview of Epipolar Geometry)	9
2.2 相機校正 (Camera Calibration)	11
2.3 色彩校正 (Color Calibration)	13
2.4 幾何校正 (Rectification).....	15
2.5 景深圖估計 (Depth Estimation)	18
2.5.1 視差與景深介紹 (Introduction of Disparity and Depth).....	18
2.5.2 景深圖量化 (Quantized Depth Map).....	20

2.5.3	景深估計參考軟體介紹 (Depth Estimation Reference Software) ...	21
2.6	紋理映射 (Texture Warping).....	25
2.7	線性疊合 (Linear Blending)	27
2.8	影像修補 (Image Inpainting).....	29
Chapter 3 正向景深映射誤差分析 (Forward Depth Warping Error Analysis) 32		
3.1	平行映射下的誤差分析 (Parallel Warping Error Analysis).....	32
3.1.1	正向景深映射演算法簡介 (Overview of Forward Depth Warping) 32	
3.1.2	映射誤差分析 (Warping Error Analysis)	34
3.1.3	競爭條件 (Race Condition).....	36
3.2	平行景深映射下的誤差修復 (Parallel Depth Warping Refinement)	41
3.2.1	可信賴模型建構 (Reliability Model Construction)	41
3.2.2	平行映射後景深修復演算法 (Parallel Warped Depth Refinement) 44	
3.2.3	實驗結果與討論 (Simulation Results and Discussion)	46
3.3	廣義映射下的誤差分析 (General Warping Error Analysis).....	51
Chapter 4 稀疏多相機虛擬視點合成設計與改進(Design and Improvement of Sparse Multi-Camera Virtual View Synthesis).....52		
4.1	傳統自由視點合成系統架構 (System Flow Chart of Traditional Free Viewpoint Television)	52
4.1.1	事前處理 (Pre-Processing).....	53
4.1.2	景深估計 (Depth Estimation).....	54
4.1.3	虛擬視點合成 (Virtual View Synthesis)	55
4.1.4	事後處理 (Post-Processing).....	56
4.2	稀疏視點合成系統架構 (System Flow Chart of Sparse View Synthesis).57	

4.3	景深圖升取樣方法 (Up-Sampling Method in Depth Map)	60
4.3.1	各類升取樣方法介紹 (Introduction of Various Up-Sampling Methods)	60
4.3.2	升取樣對映射後的景深圖影響 (Effects after Depth Warping)	62
4.4	逆向景深圖映射演算法 (Backward Depth Warping Algorithm).....	65
4.4.1	平面下的單應映射 (Homography Induced by a plane).....	65
4.4.2	256 張平面下的逆向景深圖映射 (Backward Depth Warping Induced by 256 planes)	66
4.4.3	使用適應性中值濾波器消除殘餘的量化誤差 (Survival Quantization Error Reduction using Adaptive Median Filter).....	69
4.5	金字塔型影像修補技術 (Pyramid-based Hole Filling)	72
4.6	瑕疵抑制演算法 (Artifact Reduction).....	77
4.6.1	形態學影像處理簡介 (Introduction of Morphological Image Processing) 77	
4.6.2	可信賴性模型建構 (Reliability Model Construction)	79
4.6.3	使用中值濾波器來抑制瑕疵現象 (Artifact Reduction Using Median Filter)	80
Chapter 5 實驗結果與討論 (Simulation Results and Discussions).....		83
5.1	質量評估方法 (Quality Assessment)	83
5.1.1	峰值信噪比 (Peak Signal to Noise Ratio , or PSNR).....	83
5.1.2	歸一化交叉相關 (Normalized Cross Correlation , or CORR).....	84
5.2	測試影像介紹 (Introduction of Test Sequences)	85
5.3	環境設定 (Environmental Setting)	87
5.4	平行景深圖映射後修復演算法對合成品質影響 (Parallel Warped Depth	

Refinement versus Quality of Synthetic Virtual Images)	89
5.4.1 環境設定 (Environmental Setting)	89
5.4.2 實驗結果 (Simulation Results)	90
5.4.3 實驗結果討論 (Discussions)	93
5.4.4 可信賴模型之門檻參數決策 (Threshold Decision about Reliability Model)	
96	
5.5 景深圖各種升取樣方法對合成品質影響 (Up-Sampling of Depth versus Quality of Synthetic Virtual Images)	100
5.5.1 環境設定 (Environmental Setting)	100
5.5.2 實驗結果 (Simulation Results)	101
5.5.3 實驗結果討論 (Discussions)	104
5.6 景深圖正/逆向映射對合成品質影響 (Forward/Backward Depth Warping Algorithm versus Quality of Synthetic Virtual Images)	106
5.6.1 環境設定 (Environmental Setting)	106
5.6.2 實驗結果 (Simulation Results)	107
5.6.3 實驗結果討論 (Discussions)	114
5.7 金字塔型影像修補技術對合成品值影響 (Pyramid-based Hole Filling versus Quality of Synthetic Virtual Images)	116
5.7.1 環境設定 (Environmental Setting)	116
5.7.2 實驗結果 (Simulation Results)	117
5.7.3 實驗結果討論 (Discussions)	121
5.8 瑕疵抑制方法對合成品值影響 (Artifact Reduction versus Quality of Synthetic Virtual Images)	122

5.8.1	環境設定 (Environmental Setting)	122
5.8.2	實驗結果 (Simulation Results)	123
5.8.3	實驗結果討論 (Discussions)	127
5.9	總結 (Summarize)	129
Chapter 6	結論與未來工作 (Conclusions and Future Work)	131
6.1	結論 (Conclusions)	131
6.2	未來工作 (Future Work)	133
	參考文獻	135
	自傳	139



圖目錄

圖 1	針孔相機模型示意圖 [19]	6
圖 2	針孔成像模型至 CCD 示意圖 [20]	7
圖 3	點對應的極平面幾何關係圖。 [19]	9
圖 4	校正板在不同位置下的示意圖	12
圖 5	校正板特徵點擷取	12
圖 6	色彩校正結果	14
圖 7	校正前的極平面幾何關係圖 [23]	15
圖 8	校正後的極平面幾何關係圖 [23]	16
圖 9	校正前的極線與特徵點關係圖	17
圖 10	校正後的極線與特徵點關係圖	17
圖 11	雙眼視差示意圖 [20]	18
圖 12	視差與景深示意圖 [10]	19
圖 13	光學孔徑成像系統 [24]	20
圖 14	DERS 分群模式流程圖 [12]	22
圖 15	圖切法中的 $\alpha-\beta$ 交換法和 α -膨脹示意圖 [10]	23
圖 16	已校正的左右兩影像和對應的景深圖	24
圖 17	正向與逆向紋理映射設示意圖	26
圖 18	兩種線性疊合方法示意圖。(a) 為(23)式，(b) 為(25)式。	28
圖 19	影像修補原理 [30]	29
圖 20	影像修補示意圖	31
圖 21	景深映射示意圖	34

圖 22	L2RMI (左) 與 L2RMD (右) 示意圖	37
圖 23	L2RBF (左) 與 L2RFB (右) 示意圖	38
圖 24	量化誤差型態 I (Quantization Error: Type I) 示意圖	40
圖 25	量化誤差型態 II (Quantization Error: Type II) 示意圖	40
圖 26	遮蔽區域 (Occlusion Region) 示意圖	40
圖 27	連續圖 與 下降圖 的範例 (白色區域代表標示為 1)	43
圖 28	景深圖映射後修復示意圖	45
圖 29	原始參考影像、景深圖、與映射後的景深圖	46
圖 30	連續圖門檻對於平行映射後景深修復演算法的影響	47
圖 31	下降圖門檻對於平行映射後景深修復演算法的影響 (一)	48
圖 32	下降圖門檻對於平行映射後景深修復演算法的影響 (二)	49
圖 33	廣義映射後的景深圖	51
圖 34	傳統自由視點合成演算法架構圖	52
圖 35	事前處理系統流程圖 (Pre-Processing)	53
圖 36	景深估計系統流程圖 (Depth Estimation)	54
圖 37	虛擬視點系統流程圖 (Virtual View Synthesis)	55
圖 38	事後處理系統流程圖 (Post-Processing)	56
圖 39	景深圖映射方案 (Depth Warping Scheme)	58
圖 40	稀疏視點合成系統架構圖	59
圖 41	各類升取樣方法對景深圖映射後的影響	63
圖 42	景深圖映射後 ROI 編號位置示意圖	64
圖 43	各類升取樣方法對景深圖映射後的放大部分示意圖	64

圖 44	平面下的單應映射 (HIP) 示意圖 [17]	65
圖 45	量化階層 5 的景深圖與 BDWIP-5 的結果示意圖	67
圖 46	BDWIP-5 各層結構圖 (黑色區塊代表不屬於此平面的區域)	68
圖 47	正/逆向景深映射演算法比較	69
圖 48	添加雜訊與使用(適應性)中值濾波器後示意圖	71
圖 49	景深圖經由(適應性)中值濾波器結果	72
圖 50	PBHF_level_0 (左邊) 與 PBHF_level_1 (右邊) 示意圖	73
圖 51	PBHF_level_2 示意圖	74
圖 52	引進雙三次插值法至景深圖後產生雜訊，並用 PBHF 補洞示意圖	75
圖 53	金字塔影像修補前 ROI 編號位置示意圖	76
圖 54	PBHF 補洞放大部分示意圖	76
圖 55	形態學梯度演算法架構，以半徑 4 的圓盤大小為結構元素	78
圖 56	AR Map 系統流程圖	79
圖 57	合成後瑕疵抑制方法與結果示意圖	81
圖 58	瑕疵抑制演算法前 ROI 編號位置示意圖	82
圖 59	合成後瑕疵抑制方法與結果部分放大示意圖	82
圖 60	實拍真實影像之相機陣列	86
圖 61	左視點平行景深圖映射與針對不同修復演算法之示意圖 (Kendo Sequence)90	
圖 62	右視點平行景深圖映射與針對不同修復演算法之示意圖 (Kendo Sequence)91	
圖 63	平行虛擬視點合成針對不同景深修復演算法之示意圖 (Kendo Sequence)92	
圖 64	Kendo Sequence 虛擬視點合成針對景深修復演算法之部分放大示意圖 ..95	
圖 65	不同下降圖門檻下，連續圖門檻對於合成影像 PSNR 之曲線圖	98

圖 66	連續圖門檻 2% 下，下降圖門檻對於合成影像 PSNR 之曲線圖.....	99
圖 67	虛擬視點合成針對不同升取樣方法之示意圖 (Test_01 Sequence).....	101
圖 68	虛擬視點合成針對不同升取樣方法之示意圖 (Test_02 Sequence).....	102
圖 69	虛擬視點合成針對不同升取樣方法之示意圖 (Test_03 Sequence).....	103
圖 70	虛擬視點合成針對不同升取樣方法之示意圖 (Test_04 Sequence).....	104
圖 71	Test_01 虛擬視點合成針對景深圖升取樣之部分放大示意圖.....	105
圖 72	各左視點測試影像針對正/逆向景深圖映射示意圖	108
圖 73	各左視點測試影像針對正/逆向景深圖映射後修補示意圖	109
圖 74	虛擬視點合成針對正/逆向景深映射演算法之示意圖 (Test_01 Sequence)	110
圖 75	虛擬視點合成針對正/逆向景深映射演算法之示意圖 (Test_02 Sequence)	111
圖 76	虛擬視點合成針對正/逆向景深映射演算法之示意圖 (Test_03 Sequence)	112
圖 77	虛擬視點合成針對正/逆向景深映射演算法之示意圖 (Test_04 Sequence)	113
圖 78	正/逆向景深映射對於線性疊合後的影響.....	114
圖 79	Test_01 虛擬視點合成針對正逆向景深映射之部分放大示意圖.....	115
圖 80	虛擬視點合成針對不同金字塔型影像修補階層之示意圖 (Test_01 Sequence)	117
圖 81	虛擬視點合成針對不同金字塔型影像修補階層之示意圖 (Test_02 Sequence)	118
圖 82	虛擬視點合成針對不同金字塔型影像修補階層之示意圖 (Test_03 Sequence)	119
圖 83	虛擬視點合成針對不同金字塔型影像修補階層之示意圖 (Test_04 Sequence)	120
圖 84	Test_01 虛擬視點合成針對金字塔型影像修補階層之部分放大示意圖...	121
圖 85	虛擬視點合成針對瑕疵抑制方法之示意圖 (Test_01 Sequence).....	123
圖 86	虛擬視點合成針對瑕疵抑制方法之示意圖 (Test_02 Sequence).....	124
圖 87	虛擬視點合成針對瑕疵抑制方法之示意圖 (Test_03 Sequence).....	125

圖 88 虛擬視點合成針對瑕疵抑制方法之示意圖 (Test_04 Sequence).....126

圖 89 Test_01 虛擬視點合成針對瑕疵抑制方法之部分放大示意圖.....128



表目錄

表 1	左至右方向的景深圖資訊	36
表 2	右至左方向的景深圖資訊	36
表 3	景深資訊映射後可能誤差分析表	39
表 4	深資訊映射後各種瑕疵現象	39
表 5	Kendo Sequence (Nagoya Univ.)	85
表 6	真實測試影像	86
表 7	真實測試影像的前置規格表	87
表 8	合成參數示意表	88
表 9	環境設定：平行景深圖映射後修復方法對合成品值影響	89
表 10	Kendo Sequence 虛擬視點合成針對各類方法之質量評估表 (PSNR)	93
表 11	Kendo Sequence 虛擬視點合成針對各類方法之質量評估表 (CORR).....	93
表 12	環境設定：平行景深圖映射後修復方法中門檻參數對合成品值影響	96
表 13	不同下降圖門檻下，所對應的最佳連續圖門檻與最佳合成影像 PSNR	98
表 14	環境設定：景深圖各種升取樣方法對合成品值影響	100
表 15	各類測試影像的虛擬視點合成針對不同升取樣方法之質量評估表 (CORR)	104
表 16	環境設定：景深圖正/逆向映射對合成品質影響	106
表 17	各類測試影像的虛擬視點合成針對正/逆向景深映射之質量評估表 (CORR)	114
表 18	環境設定：金字塔型影像修補技術對合成品值影響	116
表 19	各類測試影像的虛擬視點合成針對金字塔型影像修補之質量評估表 (CORR)	121
表 20	環境設定：瑕疵抑制方法對合成品值影響	122

表 21	各類測試影像的虛擬視點合成針對瑕疵抑制有無之質量評估表 (CORR)	127
表 22	平行映射下的最佳稀疏虛擬視點合成系統參數一覽表	129
表 23	廣義映射下的最佳稀疏虛擬視點合成系統參數一覽表	129



Chapter 1 緒論 (Introduction)

1.1 研究背景 (Background)

近年來，隨著軟硬體不斷的進步，立體影像技術不斷推陳出新，例如**立體電視系統 (3DTV)**[1][1]。而 MPEG 國際標準會議 (ISO/IEC Moving Picture Expert Group) 也在近幾年制定了 3DAV (3D audio-visual)[3] 文件，其中有一項是**互動式多視角視頻 (Interactive Multiple View Video)**，也稱做**自由視點視頻 (Free Viewpoint Video)**，簡稱 **FVV**)，目的是希望利用新的視點合成技術，達到即時的**互動視訊**效果。

要達到自由視點視頻的效果，現階段而言具有非常大的挑戰性。目前的架構是希望可利用若干台攝影機拍攝場景，若給定吾人想要看見相機之外的視角畫面(包括位置和方向)，則設法利用已知視訊做合成。因此在前置處理上，除了要確保視訊訊號間在時序上有無同步(Synchronization)，尚須考慮攝影機與影像校正(Calibration) 等問題。

目前在**自由視點成像系統 (Free Viewpoint Television)**，簡稱 **FTV** [4][5][6]) 上也有了初步的架構，包括**影像校正 (Camera Correction)**[7][8][9]、**影像深度估計 (Depth Estimation)**[10][11]、**多視點壓縮 (Multiview Video Coding)**，簡稱 **MVC**[12][13]、**多視點影像合成 (Multiple View Synthesis)**[14]、以及**多視點 2D/3D 顯像技術 (2D/3D Multiview Display)**。本篇論文會以影像合成為目的加以探討。

現階段而言，影像合成主要可分為兩大區塊，分別是**模型式合成 (Model-Based Rendering)**，簡稱 **MBR**) 和 **影像式合成 (Image-Based Rendering)**，簡稱 **IBR**[1]。前者主要將空間中的場景或物體做重建 (Reconstruction)，再投影到虛擬影像上。而重建方式除了可利用雷射掃描外 (稱做主動式方法，或 Active methods)；也可利用現有的影像資訊，找尋物體的特徵資訊，如邊緣與稜角，再設法重建立體資訊 (稱做被動式方法，或 Passive methods)。一些比較熱門的應用像是**擴增實境 (Augmented Reality)**[15]。

而在影像式合成[16]中，它會直接利用影像間的相對關係，將虛擬影像中的色彩空間做內插 (View Interpolation) 的動作。常見的方法如**光場式合成 (Light Field**

Rendering)[17]、或**景深圖式合成 (Depth Image-Based Rendering, 簡稱 DIBR)**[1][18]。光場式合成係利用全視函數 (Plenoptic Function) 的建構與分析，直接做虛擬視點的內插，代價是資料量過於龐大，尚有很多待改進的地方。至於景深圖式合成，它是利用景深圖資訊來分析出影像資訊在空間中的相對位置，並且做適當的映射至虛擬視角。由於景深圖式合成只需二維影像和景深圖資訊，資料量相對上減少許多，目前國際標準會議也是以此架構制定相關可行的標準；但缺點在於景深圖無法得知**遮蔽區域 (Occlusion)**的資訊，於合成過程中會有**破洞 (Hole)**現象產生，因此有不少文獻致力於探討如何修補這些瑕疵。本篇論文也會以景深圖式合成加以探討與分析。

1.2 研究動機及貢獻 (Motivations and Contributions)

目前為止，絕大多數的文獻論文所探討的景深圖式影像合成中，都是基於密集排列的一維攝影機陣列 (Dense Camera) 來實現，而在兩兩攝影機間得到的影像，其對應點的水平位移量 (Disparity) 往往不超過 4~50 個像素 (pixels)。以此假設下所遇到的遮蔽區域，占整張欲合成的影像並不多。

此外，一般的視點合成系統架構，也都會考慮影像已經做過**幾何校正 (Rectification)**，在此假設下，對後續的處理有很大的幫助。但對真實情況而言，攝影機之間不會完全平行，而在不平行排列下所獲取的影像資訊，其遮蔽區域相較之下會變多、物體間的相對位置也比較複雜。雖有一些文獻論文會考慮到未校正的情形[1]，不過還是脫離不了密集相機陣列的架構。

本篇論文會以**稀疏非平行陣列相機 (Sparse Non-parallel Camera Array)**為考量，實現自由視點合成系統。在此過程中，我們根據景深映射後所產生的各種瑕疵現象，做一連串的分析，並且提出一套完善的**景深映射後修復演算法**，以改進現有的平行自由視點合成系統。此外，針對稀疏非平行陣列合成系統下所遇到的各種問題，一共改進了四個地方而提出適當的演算法，分別是**景深圖升取樣方法**、**逆向景深圖映射演算法**、**金字塔型影像修補技術**、以及**瑕疵抑制演算法**。

在**景深圖升取樣方法**中，我們觀察到景深在做升取樣時，需有別於一般的色彩影像做升取樣的方法，否則會影響到後續的合成品質。而**逆向景深圖映射演算法**則是改進現有的正向景深圖映射演算法，以降低大量的瑕疵現象。至於現有的影像修補技術，因無法抵抗雜訊所產生的瑕疵現象，本篇論文也提供一套**金字塔型影像修補技術**來做改善。最後則是提出**瑕疵抑制演算法**，讓最後的合成品質可以更佳。詳細流程會在後面的章節闡述。

1.3 論文大綱 (Organization of the Thesis)

在本篇論文中，**第二章**會先從投影空間的理論基礎講起，然後加以介紹整個相機系統的基本成像原理。並且根據現有的視點合成系統，簡介每個系統所需用到的各種工具。**第三章**則會根據景深圖映射所產生的各種瑕疵現象，加以分析並且提出一套景深映射後修補方法，以降低瑕疵區域。**第四章**是結合傳統的自由視點合成系統、加上本篇論文提出的各種改善方法，而提出稀疏多視點的整個系統。改善方法包括景深圖升取樣方法、金字塔型影像修補技術、瑕疵抑制演算法。並且發展一套逆向景深映射演算法，以改善傳統的正向景深映射演算法。**第五章**會是利用自己拍攝的真實影像，合成出任意的虛擬視角影像，並且加以討論與分析。最後**第六章**的部分則是對本篇論文的總結與未來研究方向的期許。

Chapter 2 立體影像系統基礎知識 (Stereo Image System Basics)

2.1 立體成像系統導論 (Introduction of Stereo Image System)

當我們要分析影像中的資訊，較有系統的方式是建立一個數學模型，來描述空間中的物體是如何經由相機透鏡、感光耦合元件 (Charge Coupled Device, 簡稱 CCD)，以至於成像。本節 2.1.1 會簡介成像系統中的投影空間，並且如何用它來建構影像和物體的座標。本節 2.1.2 則是介紹相機成像最常使用的針孔相機模型。最後本節 2.1.3 會介紹兩相異影像間的幾何關係。



2.1.1 投影空間 (Projective Space)

若想有系統地描述一張影像的資訊，我們必須先建構出點和線的資訊。假設影像中的某一條直線方程式為 $ax+by+c=0$ ，則可以用向量 $\ell=[a,b,c]^T$ 來表示該線段，不難得知對任意非 0 常數 k ，其 $[ka, kb, kc]^T$ 皆代表同一條直線，在數學上可將 $[ka, kb, kc]^T$ 視為一等價類 (Equivalence Class)¹。因此將所有屬於 $\mathbb{R}^3 - [0,0,0]^T$ 的等價類集合起來，即構成二維的**投影空間** IP^2 (**2D Projective Space**)。簡單說， IP^2 包含二維資料量，而每個資料 (等價類) 就代表一條直線。

¹ 等價類 (Equivalence Class) 在數學上為一個集合，代表著裡頭的每個元素對於某些規則下會互相等價。

例如考慮集合 $[\ell]=\{\ell \mid \ell \in [ka, kb, kc]^T, k \neq 0\}$ ，對所有在 $[\ell]$ 裡頭的元素，皆代表著一條直線方程式 $ax+by+c=0$ ，因此我們稱集合 $[\ell]$ 為一等價類。

點的描述方式也可類比線段，因為不論在平面空間中的點座標、或是平面向量表示法，皆為 $[x, y]^T$ 。由前面的直線方程式假設得知 $[x, y, 1]^T \ell = 0$ ，因此我們令 $p = [x, y, 1]^T$ 代表平面中的一點 $q = [x, y]^T$ ，並且落在直線 ℓ 上。我們稱 p 為 q 的齊次座標 (Homogeneous Coordinate)，並且不難得知 $[p] \in \mathbb{P}^2$ ，其中 $[p]$ 代表與 p 點等價的所有元素集合。

利用上面的描述，我們可推導出在齊次座標下，任兩條相異直線 ℓ 和 ℓ' ，其交點為 $\ell \times \ell'$ (兩直線外積)；兩相異點 p 和 p' 被直線 $p \times p'$ 通過(兩點外積)。此兩條簡單又有力的公式，可突顯出投影空間對於點和線是互通的，並且打下了良好的基石。

一個最好的例子就是將無窮遠的概念合理化。對一個無窮遠處的點而言，可以想成是兩條平行直線的交點，利用前面所述外積的想法，不難推得其結果會型如 $[x, y, 0]^T$ ，而對應的平面座標則是 $\left[\begin{matrix} x \\ y \\ 0 \end{matrix} \right]^T$ 。雖然此一平面座標表示法在數學上沒有意義，不仿想成兩平行線會交在無窮遠點處，也就是該平面座標點。我們稱此點為理想點 (Ideal Point)。

相對的，於一條無窮遠處的線而言，可想成是這些理想點的宇集所構成的一條線，會型如 $\ell_\infty = [0, 0, 1]^T$ ，我們稱 ℓ_∞ 為線在無窮遠處 (Line at Infinity)。並且可以利用外積的概念，得知任一條直線 ℓ 和 ℓ_∞ 的交點，會是某一個理想點，而該理想點的齊次座標會和 ℓ 的方向有關。可想成是任一條直線 ℓ 其實是封閉曲線，亦即 ℓ 的”兩端點”會交於一個理想點。而 ℓ_∞ 正好將所有封閉曲線 ℓ 串聯在一起。

仿造 \mathbb{P}^2 投影空間，我們也能利用相同的想法建構 \mathbb{P}^3 ，例如齊次座標下的 $[a, b, c, 1]^T$ 代表空間中的一個平面方程式，也可以代表空間中的一個點。而平面在無窮遠處 (Plane at Infinity) 的概念以及其它細節，就不多加詳述。

此外，若投影空間中的點經由簡單的線性轉換至新的投影空間中，由線性轉換所構成的矩陣我們一般稱作是**投影矩陣 (Projective Matrix)**。以下 2.1.2 小節會提到針孔相機模型其實就是一種投影矩陣關係。

2.1.2 針孔相機模型 (Pinhole Camera Model)

針孔相機模型是一種描述相機投影的簡化模型，並且廣為使用。顧名思義，將相機透鏡視為一個針孔，因此整個相機投影模型可以估計成簡單的針孔成像模型，對之後的分析與複雜度將大幅簡化許多。如圖 1 所示：

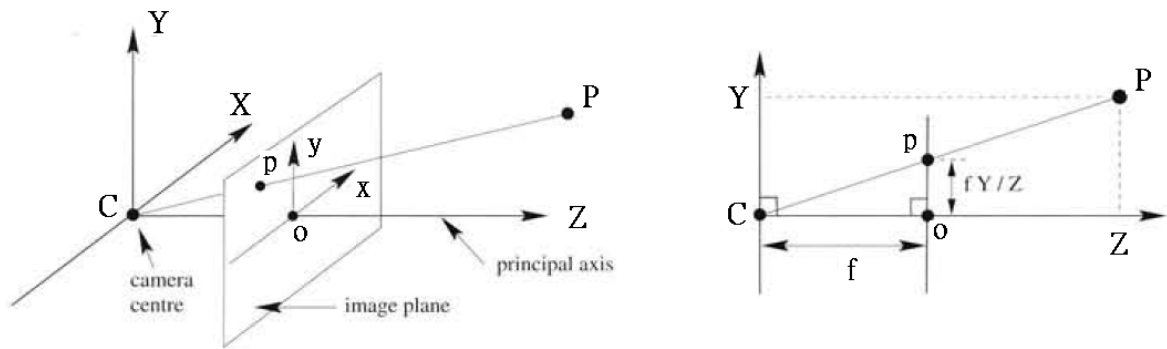


圖 1 針孔相機模型示意圖 [19]

圖中的 C 代表相機透鏡位置、o 代表影像的主軸點、f 代表焦距 (單位一般訂為公尺)、大寫 P 代表空間中的一個點座標，小寫 p 則是經由相機 C 投影後的影像座標。若 $P_{cam} = [X, Y, Z]^T$ ，則經由簡單的相似三角形性質可推得如下關係：

$$p = (u, v) = \left(\frac{fX}{Z}, \frac{fY}{Z} \right) \quad (1)$$

若寫成矩陣型態，則可改寫如下：

$$z \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \begin{bmatrix} f & 0 & 0 & 0 \\ 0 & f & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix} = K \begin{bmatrix} I_{3 \times 3} & \vec{0} \\ \hline P_{cam} \\ 1 \end{bmatrix} \quad (2)$$

其中 $K = \begin{bmatrix} f & 0 & 0 \\ 0 & f & 0 \\ 0 & 0 & 1 \end{bmatrix}$ ， z 為一非 0 常數。不過由於透鏡的主軸點不一定會落在影像中心座標，所以投影後的點 p 需要加上位移量 (c_x, c_y) 。而且 CCD 在真實製造中並非是單純的正方形，所以長寬比會有所差異，且直角座標會有所歪斜。因此 K 矩陣可被重新改寫為：

$$K = \begin{bmatrix} f_x & s & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{bmatrix} \quad (3)$$

其中 (f_x, f_y) 為焦距分別被 CCD 的水平與垂直長度正交化後的量，單位為像素 (pixel)。而 s 為一歪斜常數，跟歪斜角度 θ 和 f_x 有關，如下圖所示：

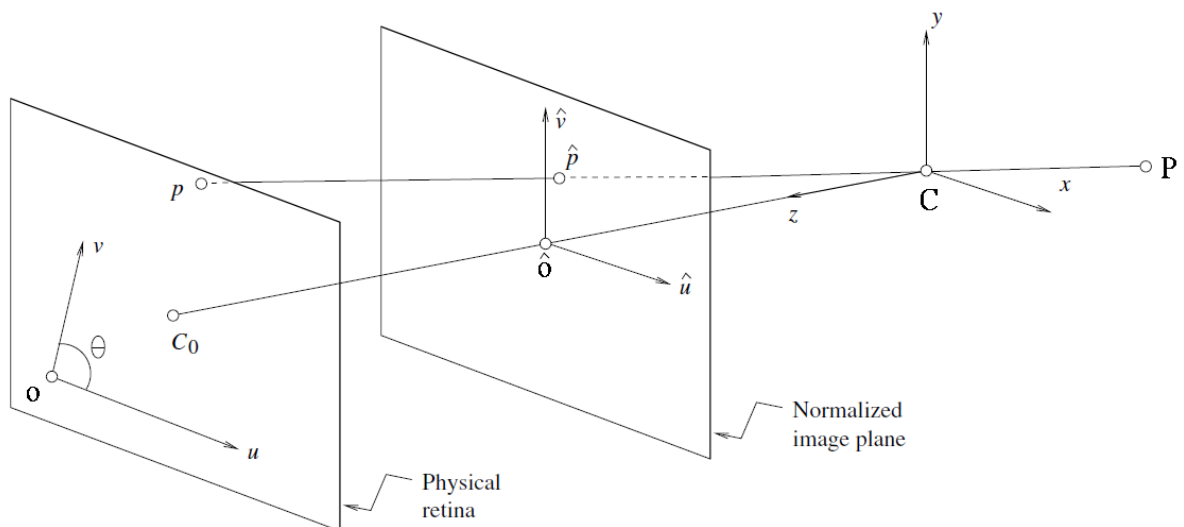


圖 2 針孔成像模型至 CCD 示意圖 [20]

此外，世界座標中心點可以不需要訂在透鏡中心位置 C ，可以透過簡單的平移旋轉量來定義世界座標和相機座標之間的關係：

$$P_{cam} = RP_w + t = \begin{bmatrix} R & t \end{bmatrix} \begin{bmatrix} P_w \\ 1 \end{bmatrix} \quad (4)$$

其中 P_w 為點 P 在世界座標 W 上的座標、 R 為一 3×3 旋轉矩陣、 t 為一 3×1 平移向量。綜合 (3)(4) 式，(2)式可以進一步改寫為：

$$z \begin{bmatrix} p \\ 1 \end{bmatrix} = K \begin{bmatrix} R & t \end{bmatrix} \begin{bmatrix} P_w \\ 1 \end{bmatrix} = KR \begin{bmatrix} I & -c \end{bmatrix} \begin{bmatrix} P_w \\ 1 \end{bmatrix} = M \begin{bmatrix} P_w \\ 1 \end{bmatrix} \quad (5)$$

其中 c 代表相機中心在世界座標上的座標值。

(5)式中的 M 矩陣正是 2.1.1 小節所提到的投影矩陣，因為它可以將世界座標中的一點 P_w ，經過齊次座標表示法，線性投射到影像中的點 p 。而這樣的 3×4 矩陣我們一般也稱作**相機參數 (Camera Parameters)**。由前面的討論可知相機參數可以分成兩部分，分別是**內部參數 (Intrinsic Parameter) K** ，和**外部參數 (Extrinsic Parameter) $\begin{bmatrix} R & t \end{bmatrix}$** 。內部參數用來描述相機的焦距、CCD 的規格等等；而外部參數則是定義出相機中心和世界座標間的關係。

不過由 (5)式中我們會發現 M 矩陣左邊的 3×3 矩陣 KR 會是非奇異矩陣，但是對於投影矩陣而言不需要有此限制。因此我們有時候會嚴格地把型如 $K \begin{bmatrix} R & t \end{bmatrix}$ 的相機參數稱作是**有限投影矩陣 (Finite Projection Matrix)**；而用來轉換投影空間中的任意 3×4 矩陣，我們會稱作**廣義投影矩陣 (General Projection Matrix)**。一般而言，我們會比較喜歡由幾何關係所建構出來的有限投影矩陣。

有了相機參數，我們便能透過幾何關係分析影像中的資訊，並且了解在世界座標中的物體大小、位置、以及相對關係、...等等。只是對真實的相機而言，其透鏡不若針孔成像原理那麼單純，尚須考慮到透鏡的製程曲率、大小等等，因此需要額外考慮到相機透鏡的失真率 (Lens Distortion)，而透鏡失真率常數也算是內部參數的其中一部份。但它牽扯到很複雜的非線性轉換，在此就不詳細介紹，可參考電腦視覺相關書籍 [19][20]。

2.1.3 極平面幾何概述 (Overview of Epipolar Geometry)

若兩台相機拍攝同一個物體，我們可以利用幾何上的關係，從兩張影像間找出可用的線索與資訊，並且加以處理，因此多張影像資訊的幾何分析會是一個很重要的學問與研究基礎。如圖 3 所示，圖 3-a 代表空間中一點 X 分別經由兩台相機 C 和 C' 投影至影像平面 x 和 x' 。我們把經過點 X 、 C 和 C' 的平面稱作極平面 (Epipolar Plane)，而兩相機中心的連線線段稱作基線 (Baseline)，至於 C' 投影至 C 上的影像平面 e 則稱做極點 (Epipole)。我們可觀察出當兩相機的位置與角度皆固定，則兩影像間的極點也會跟著固定，所以當找出對應的極點，可有助於我們了解相機的擺放位置。

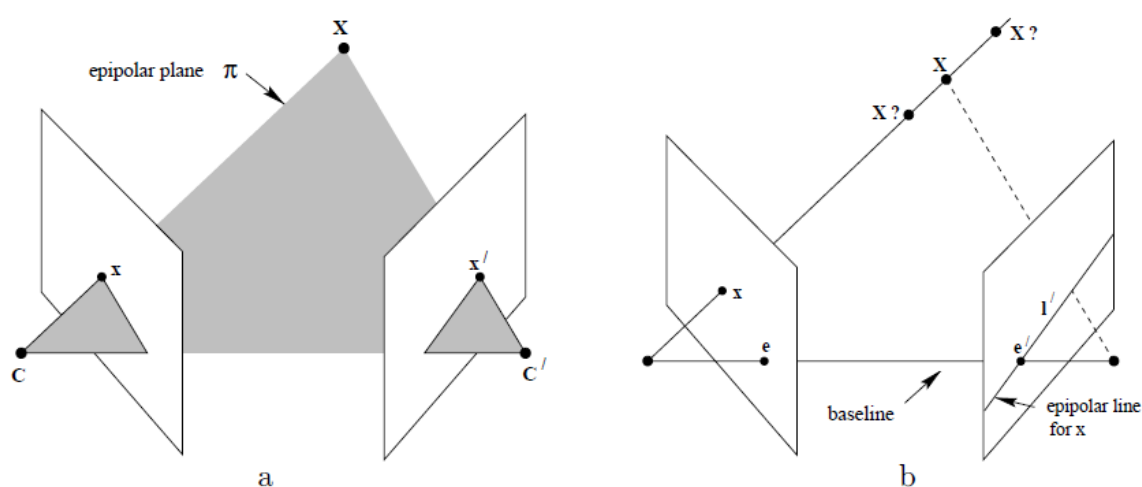


圖 3 點對應的極平面幾何關係圖。 [19]

一個古老卻又很難的問題是：我們要如何知道相機 C 中的影像點 x ，對應到相機 C' 中的哪個影像點？如圖 3-b 所示，我們會發現由點 C 、 X 所形成的射線，其射線上的點對於相機 C ，皆會投影至影像點 x ；但是對於相機 C' ，卻會投影至不同的影像點 x' 。若我們將這些點 x' 集合起來，在影像平面上會構成一條直線 l' ，我們把 l' 稱作是影像點 x 的**極線 (Epipolar Line)**。不難觀察出極線一定會通過相對應的極點。因此利用極線的關係，不僅可以找出極點位置，對於 x 的**對應點 x' (Point Correspondence)** 尋找也有相當的幫助。

此外，任意的對應點在齊次座標下的表示式 x 和 x' 必然會存在以下關係：

$$x'^T F x = 0 \quad (6)$$

其中 F 是一個 3×3 矩陣，稱作**基本矩陣 (Fundamental Matrix)**。該基本矩陣不僅連結兩對應點的關係，並且也直接透露出相對應的極線向量為何。更進一步，若已知兩相機參數分別為 M_L 和 M_R ，則可以推導出基本矩陣為：

$$F = [e']_x M_R M_L^+ \quad (7)$$

其中 $[e']_x$ 是極點 e' 的矩陣表示法，意思是若對任意 3×1 向量 v ，其 $[e']_x v$ 等價於 e' 和 v 做外積。而 M_L^+ 則是對 M_L 做虛擬逆轉換 (Pseudo-Inverse)。

2.2 相機校正 (Camera Calibration)

相機校正的目的用於找出每台相機的相機參數為何，最具經典代表莫過於 Zhang [7][21] 所提出來的校正方法。他的想法是利用校正板中的點，透過空間中的平面關係簡化問題，推導出相機參數的精確解，並且在用幾何關係的限制，來修正代數解所造成的誤差。其大致演算法如下：

<步驟 1> 拍攝校正板，並且給定該特徵點在世界座標中的座標數值。

<步驟 2> 對校正板做適當的旋轉與移動，拍攝新的照片，並且重複做 <步驟 1>。

如圖 4 所示。

<步驟 3> 偵測每張照片中的特徵點位置，如圖 5 所示。並且利用最小化代數距離 (Algebraic Distance) 的解，估算出每張照片的內部參數與外部參數。

<步驟 4> 利用最大化似然估計 (Maximum Likelihood Estimation, 簡稱 ML) 所找到的解，也就是最小化幾何距離 (Geometric Distance)，來改善每張照片估出來的相機參數，包括透鏡失真率。而這裡的幾何距離定義如下：

$$\sum_{i=1}^n \sum_{j=1}^m \left\| p_{ij} - \hat{p}(K, R_i, t_i, P_j) \right\|^2 \quad (8)$$

其中 i 代表第 i 個相機的引數、 j 代表特徵點的引數、大寫 P 和小寫 p 分別代表世界座標值和對應的影像特徵點座標值、而 \hat{p} 則是 P 經由估計完的相機參數後，所投射的座標值。不過最小化 (8) 式並非是單純的線性問題，因此可使用 Levenberg-Marquardt 演算法 (簡稱 LM Algorithm) 來求解。在此就不加詳述。

以下圖 4 是利用目前網路上所提供的免費相機校正 Toolbox [22]，其原理大致跟 Zhang 所提供的方法差不多，只是擁有更完善的 GUI 介面方便使用者操作。而圖 5 則是利用該軟體所擷取出來的特徵點。



圖 4 校正板在不同位置下的示意圖

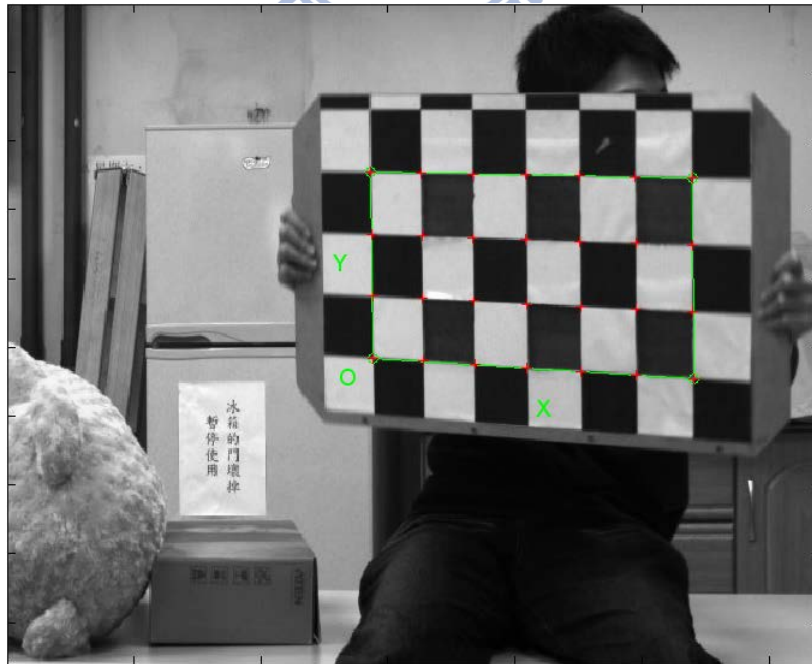


圖 5 校正板特徵點擷取

2.3 色彩校正 (Color Calibration)

當若干台相機拍攝同一個場景，會因為光線折射、散射、與接收度問題等等，使得色彩會有所差異性，而色彩校正的目的就是要最小化不同影像之間色彩差異。目前常見的方法有統計圖式校正法 (Histogram-based Color Calibration)[9]、及模擬退火式校正法 (Simulated Annealing Color Calibration)[15]。以下只介紹最簡單的統計圖式校正法。

統計圖式校正法簡單說是將一張欲校正的色彩統計圖，經由設計一個等化器，使得校正完後的色彩統計圖和目標色彩統計圖一致。而對一張 bmp 格式的彩色圖，我們可以分別擷取 R、G、B 等三個通道的色彩資訊，並且對 0~255 強度值做色彩統計。

若以機率模型的角度上來看，可把正交化後的色彩統計圖視為顏色強度的機率函數 (Probability Mass Function, 簡稱 PMF) $P[n]$ ，代表在顏色強度 n 在關心的區域上所發生的機率為 $P[n]$ 。接著計算出累計質量函數 (Cumulative Mass Function, 簡稱 CMF) $C[n]$ ，並且利用累計質量函數的單調遞增特性，來設計出適當的反函數。有了反函數後，我們便能設計等化器如(9)式所示：

$$n_{new} = C_s^{-1}[C_t[n_{old}]] \quad (9)$$

其中下標的 t 和 s 分別代表欲校正的影像和參考影像。利用(9)式，我們便能將欲校正的顏色強度 n_{old} ，轉到顏色強度 n_{new} ，使得色彩統計圖變得一致。如圖 6 所示。



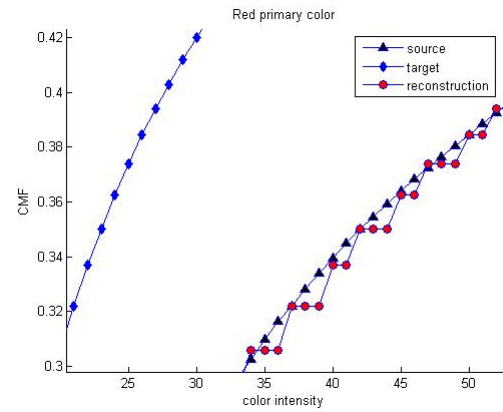
(a) 參考影像



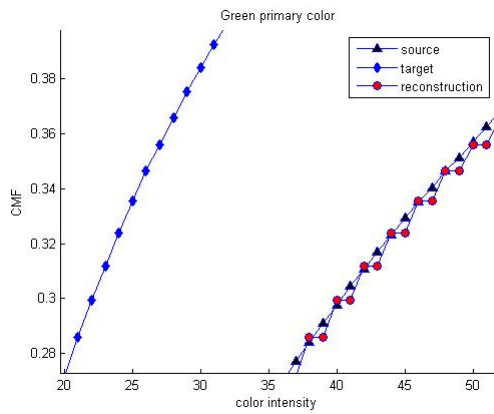
(b) 欲校正的影像



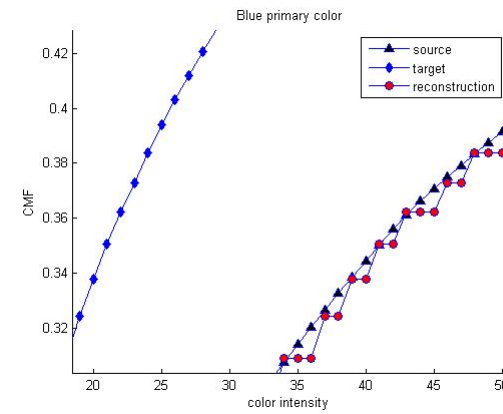
(c) 色彩校正後的影像



(d) 紅色主色彩累計分布圖



(e) 綠色主色彩累計分布圖



(f) 藍色主色彩累計分布圖

圖 6 色彩校正結果

2.4 幾何校正 (Rectification)

在 2.1.3 小節中有介紹關於極平面的幾何性質，其中的極線可以用來幫助尋找對應點，不過真的要實現尋找對應點的演算法還是會有所困難，因為一條歪斜的極線對於以像素點為單位的量化上，尋找對應點容易會有數值上的誤差，並且計算量龐大。因此一個簡單的想法是可以將兩張影像做一下適當的處理，使得所有的極線皆為水平線，如此一來對應點的垂直座標量會跟對應點獨立，對於對應點的尋找上也相對方便許多。而幾何校正正是在做如何讓對應的極線落在同一條水平線上。

幾何校正的方法基本上可以用簡單的幾何關係推導得出 [23]。如圖 7 所示， C_1 和 C_2 分別代表兩台相機的中心點、 W 為世界座標中的一個點座標、 M_1 和 M_2 分別為兩相機的投影點、 E_1 和 E_2 則是對應的極點。我們的想法是讓兩影像分別經過一個轉換後，使得兩影像可以共平面。如此一來轉換後的影像，其極線的對應關係就會是同一條水平線上，如圖 8 所示：

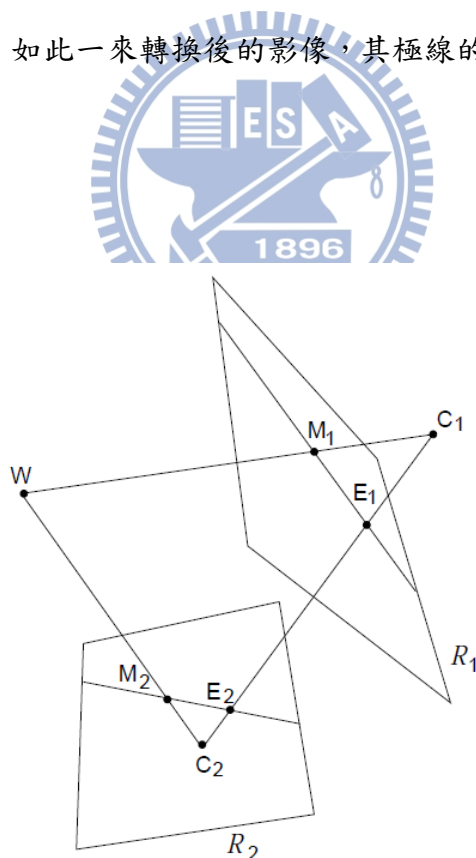


圖 7 校正前的極平面幾何關係圖 [23]

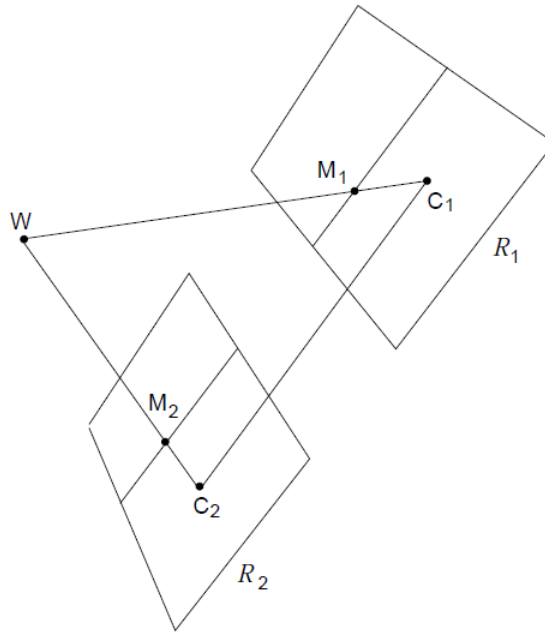


圖 8 校正後的極平面幾何關係圖 [23]

可觀察出其轉換關係就只是以 C_1 和 C_2 為中心，做一下適當的旋轉。若兩原始相機參數為 $K_L R_L [I | -c_L]$ 和 $K_R R_R [I | -c_R]$ ，經由推導可以得知兩影像校正後的相機參數可寫成 $KR [I | -c_L]$ 和 $KR [I | -c_R]$ ，其中：

$$K = \frac{1}{2}(K_L + K_R) \quad (10)$$

$$R = [r_1 | r_2 | r_3]^T \quad (11)$$

(11)式的 r_1 需跟基線平行，所以 $r_1 = \pm \frac{c_L - c_R}{\|c_L - c_R\|}$ 。而 r_2 只要跟新的相機坐標系中的 z 軸方向和 r_1 垂直即可，可以有很多種定義方式。最後 r_3 在直接計算 r_1 和 r_2 的外積即可 (因為旋轉矩陣為正交矩陣)。至於 (10) 式中我們也會將歪斜參數設為 0，以降低複雜度。此外，我們也能調整校正後的內部參數中，其相機主軸點在影像中心的座標值 (c_x, c_y) ，使得可以擷取更多校正後的資訊。以下圖 9 和圖 10 分別是利用標準照片所測試校正前與校正後的結果，並且標上特徵點與對應的極線：

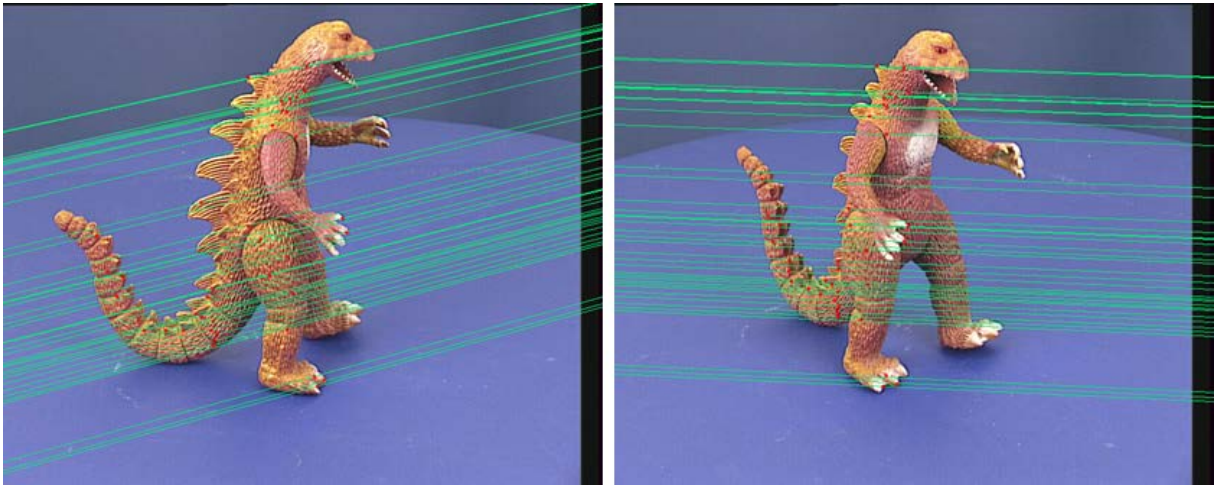


圖 9 校正前的極線與特徵點關係圖

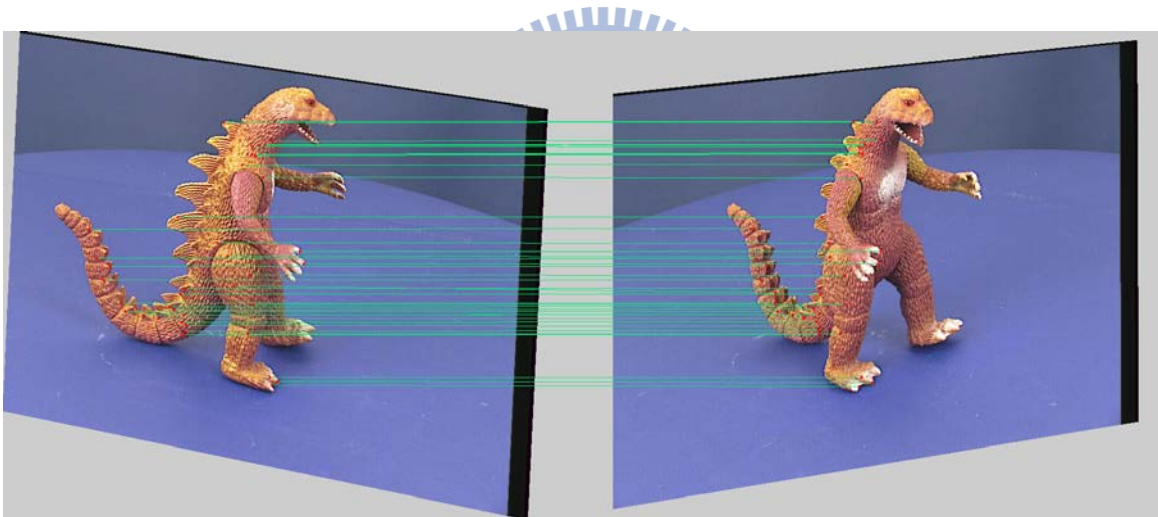


圖 10 校正後的極線與特徵點關係圖

幾何校正對於影像處理相關有很大的幫助。若考慮更多張影像的校正，可參考 Kang *et al.* [8]。

2.5 景深圖估計 (Depth Estimation)

2.5.1 視差與景深介紹 (Introduction of Disparity and Depth)

人在看一個物體會感到有立體感，主要是因為人的雙眼所接收到的訊息有所差異，而這差異再經由大腦整合後，便可辨識出哪些資訊是近景、那些是遠景，並且加以學習而善加利用。經由研究發現，可供人眼視覺有遠近感的其中一個重要線索就是**雙眼視差 (Binocular Disparity)**。如圖 11 所示：

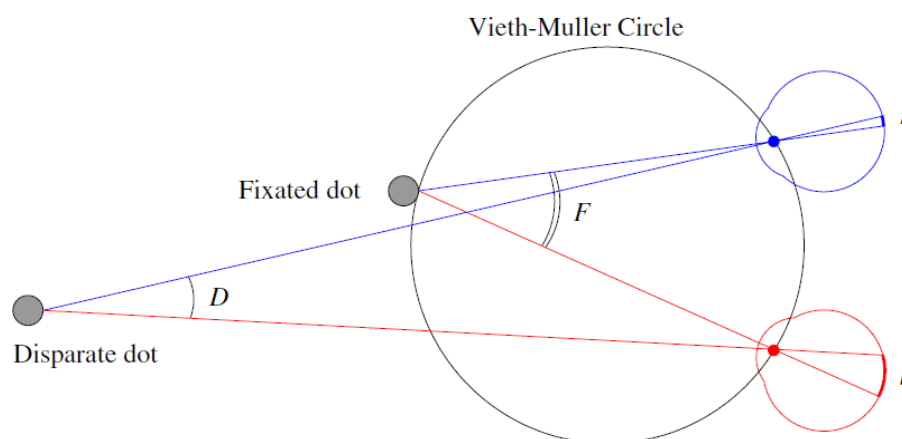


圖 11 雙眼視差示意圖 [20]

圖 11 中，雙眼視差定義為 $l-r$ ，也就是兩個視角的差異。大腦可以藉由視角差的差異性，加上肉眼的旋轉角度等等，即可辨別出物體的相對遠近程度。而對於兩台平行相機所拍攝出來的影像，也能藉由尋找視差 (Disparity) 的資訊，來估測影像中的物體遠近程度。而物體相對於相機的遠近程度，就稱為**景深 (Depth)**。景深值一般的明確定義是物體對於相機主軸方向的距離，如圖 12 所示：

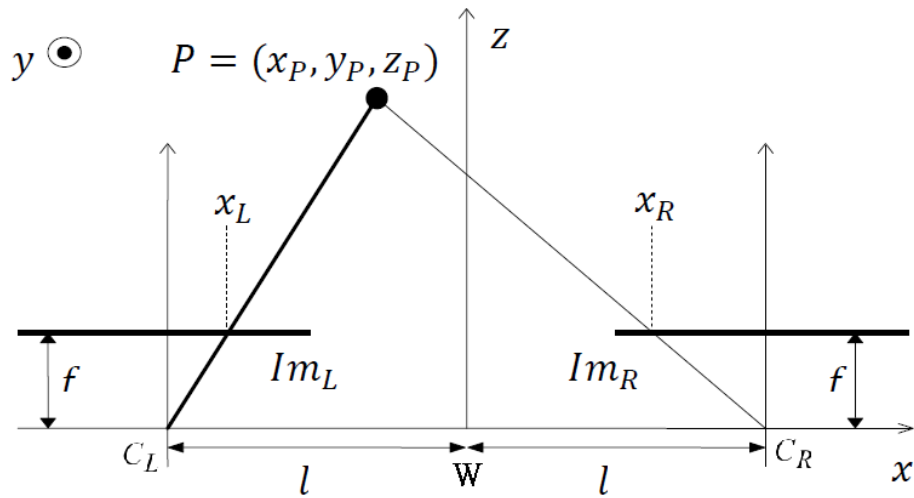


圖 12 視差與景深示意圖 [10]

假設空間中一點 P 在 W 座標系下為 (x_p, y_p, z_p) ，則對於兩台平行相機（或更精確點描述為已做過幾何校正的影像） C_L 和 C_R ，其景深值 (Depth) 就定義為 z_p 。若分別投影至兩影像平面，其水平座標分別為 x_L 和 x_R ，則 P 對於兩影像間的視差 (Disparity) 定義為 $|x_L - x_R|$ 。利用相似三角形可推得景深值 z_p 和視差 d 有如下關係：

$$d = |x_L - x_R| = f \left(\frac{x_p + l}{z_p} - \frac{x_p - l}{z_p} \right) = \frac{f \cdot (2l)}{z_p} = \frac{f \cdot B}{z_p} \quad (12)$$

其中 f 為兩相機的焦距(假設兩相機的焦距相同)、而 $B = 2l$ 為兩相機的距離，也就是基線線段長度。因此若從兩張校正後的影像中，找出每個對應點的視差值，則我們可以利用 (12) 式簡單的轉換關係，推得影像資訊中每個對應點的景深值。有了景深資訊，不論要做影像辨識、分割、重建、合成、... 等等，都會有莫大的助益。

2.5.2 景深圖量化 (Quantized Depth Map)

由於景深資訊在立體成像系統方面扮演很重要的角色，我們會很希望能達到資料即時傳輸效果，因此有很多人致力於景深資訊的壓縮工作上 [12] [13]。不過景深壓縮並非是本篇論文的重點，我們只會介紹目前國際標準會議所使用的景深資訊量化方法。

目前公認的想法是，將景深資訊用 8 位元(0~255) 共 256 個灰階數值，來表示一張圖的景深資訊，稱做**景深圖 (Depth Map)**。最早期的做法是有人將景深資訊根據最大和最小值做線性量化，不過目前大家所使用的量化方式是根據微軟 (Microsoft) 所提供的量化公式 (13)：

$$z_q = \text{round} \left\{ (N-1) \cdot \frac{z^{-1} - z_{far}^{-1}}{z_{near}^{-1} - z_{far}^{-1}} \right\} \quad (13)$$

其中 z_{near} 和 z_{far} 分別是整張圖的最小和最大景深值、 z 和 z_q 分別為原始景深值和量化後的景深值、而 N 代表量化的階層數 (用 $\lceil \log_2 N \rceil$ 位元數來表示)、 $\text{round}\{\}$ 則是四捨五入的算符。該公式的想法是來自於 Chai *et al.* [24] 利用分析全視函數 (Plenoptic Function)，對視差做最佳化取樣所得到的結果。如圖 13 所示：

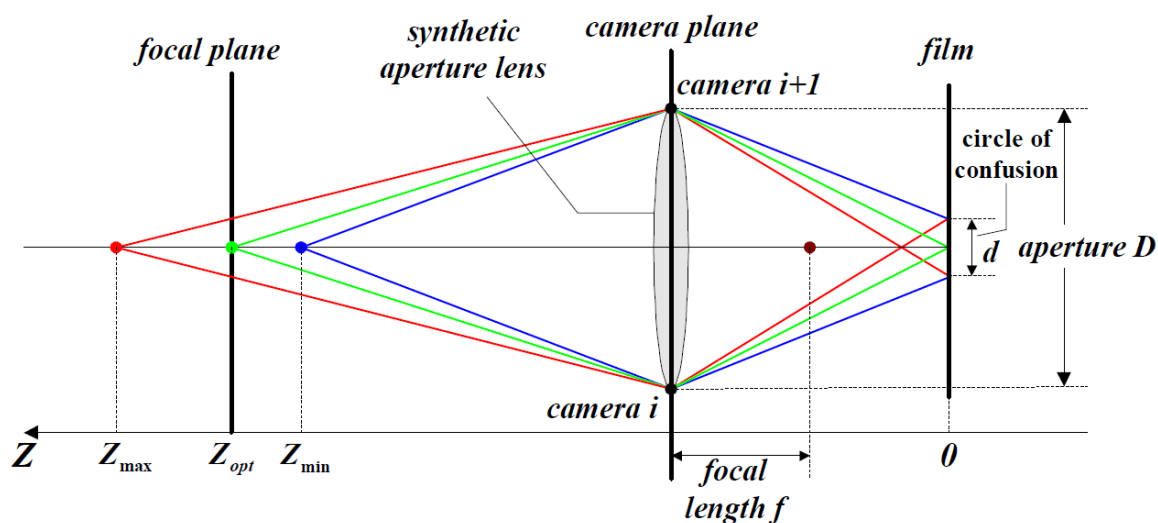


圖 13 光學孔径成像系統 [24]

至於要將量化後的景深資訊還原回真實的景深資訊，則可以對 (13)式做反轉換：

$$z_{est} = \frac{1}{\frac{z_q}{N-1} \cdot (z_{near}^{-1} - z_{far}^{-1}) + z_{far}^{-1}} \quad (14)$$

其中 z_{est} 是還原後的景深資訊。下標的 est 代表估計的意思，目的是為了要和原始景深資訊 z 做區別。

2.5.3 景深估計參考軟體介紹 (Depth Estimation Reference Software)

景深估計參考軟體 (Depth Estimation Reference Software，簡稱 DERS) 是由日本名古屋大學 (Nagoya University) 的研究團隊所開發出來的參考軟體，在國際標準會議上被用來在立體視訊和自由視點合成的其它演算法開發上。DERS 它是根據目前現有各式的演算法與改進而整合出來的一套景深估計軟體，因此會不斷地更新與改進。

景深估計一般會有三大假設性的出發點：<1>像素色彩強度相似性。<2>像素唯一性。<3>平滑和連續性。不過由於兩影像間難免會存在遮蔽區域，使得第<2>點相違背。而且也沒有一個比較好的方法來解決遮蔽區域的景深問題，因此目前的景深估計演算法會以最小化遮蔽區域誤差為著手點。

而以 DERS 的目前架構，共可分為三類。分別是 <1>全自動模式 (Automatic Mode)。<2>分群模式 (Segmentation Mode)。<3>半自動模式 (Semi-Automatic Mode)。本章節只會介紹分群模式的系統流程圖。如圖 14 所示：

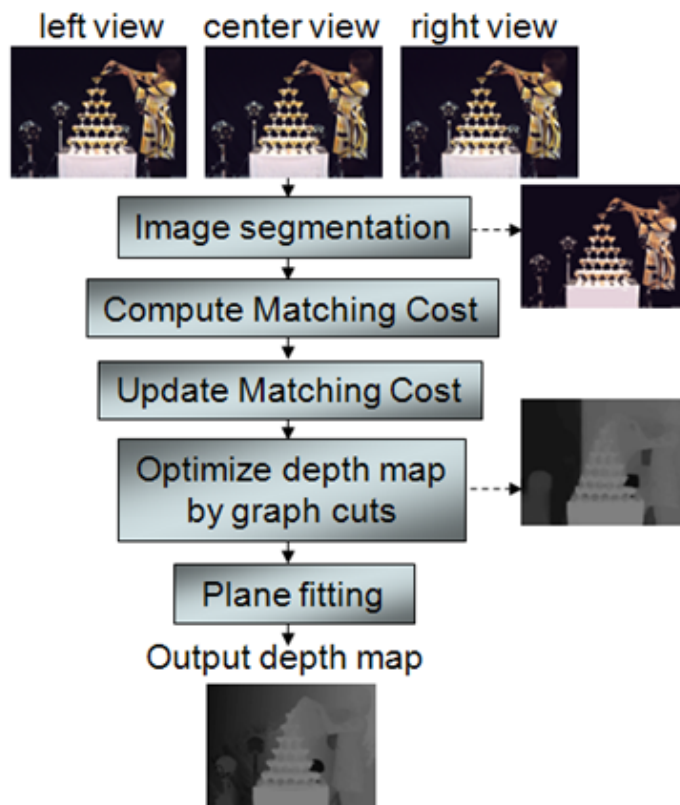


圖 14 DERS 分群模式流程圖 [12]

若我們只有考慮左右兩張影像資訊圖，一開始會先分別做影像分群 (Segmentation)。而分群演算法可以使用均值移動演算法 (Mean Shift Algorithm)、或是 K 類均值群聚演算法 (K-mean Clustering)。也就是將影像中有相似色彩強度且位置相近的像素點視為一群。分群的目的是在於降低點對應尋找演算法的複雜度。

接著在根據每個像素點間的色彩強度關係，建構出視差空間圖像 (Disparity Space Image，簡稱 DSI)，而其建構方式如(15)所示：

$$C(x, y, d) = f(I_L(x, y), I_R(x - d, y)) \quad (15)$$

其中 $I(x, y)$ 代表色彩強度。而函數 $f(I_L, I_R)$ 則是用來量測兩像素點的颜色相似程度，在機率角度上一般會扮演著對數似然函數 (Log-likelihood Function) 的腳色，一般的定義方式有很多種，包括 SAD、SWA、GRAD、... 等等。等建構完後，在建立一成本函數，稱作能量函數 (Energy Function)，它的型態如(16)所示：

$$E(d) = E_{data}(d) + \lambda \cdot E_{smooth}(d) \quad (16)$$

$$E_{data}(d) = \sum_{(x,y) \in ROI} C(x, y, d(x, y)) \quad (17)$$

$$E_{smooth}(d) = \sum_{(x,y) \in ROI} \rho\left(\frac{\partial d(x, y)}{\partial x}, \frac{\partial I(x, y)}{\partial x}\right) + \rho\left(\frac{\partial d(x, y)}{\partial y}, \frac{\partial I(x, y)}{\partial y}\right) \quad (18)$$

其中 $d(x, y)$ 代表左影像至右影像的視差圖。 $E_{data}(d)$ 代表著兩區域的色彩強度越相似，則該值會越小。不過因為邊緣與遮蔽區域部分會出問題，因此需要考慮到(18)，而其中的 $\rho(d, I)$ 可定義為對 d 為一遞增函數、對 I 為一遞減函數。

有了能量函數的定義，接著再使用廣義的方法來解(16)，進而找出視差圖 $d(x, y)$ 。而 DERS 採用圖切法 (Graph Cuts) 中的 α - β 交換法 (α - β Swap) 和 α -膨脹 (α -expansion)。如圖 15 所示。圖切法可利用前面所做的分群法來作加速與近似。

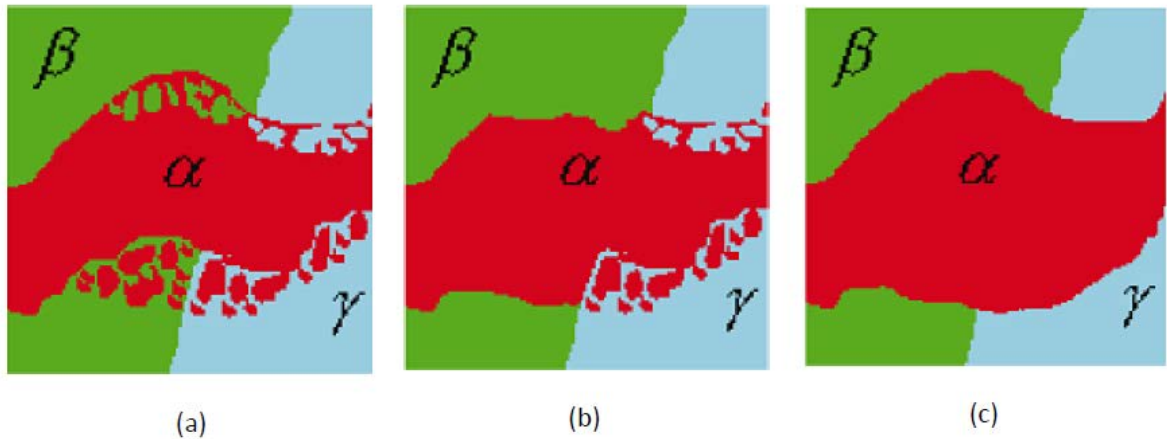


圖 15 圖切法中的 α - β 交換法和 α -膨脹示意圖 [10]

等利用圖切法得到視差圖，我們再利用平面擬合法 (Plane Fitting) 來做視差圖的修正，它是利用視差圖屬於同一群的點，用不同的平面方程式來近似每群在空間中的位置。不過由於使用最小平方誤差法來作近似，很容易因受到離群點 (Outlier) 的影響，因此會先使用隨機一致性演算法 (Random Consecutive and Sample，簡稱 RANSAC) 來排除離群點，再進而使用平面擬合法來修復視差圖。此外，視差圖修復完後可能還是

會存在一些雜訊，因此最後會在使用**中值濾波器 (Median Filter)** 來濾掉高頻雜訊。

只是對於 DERS，它所使用到的資訊是三張影像，而非傳統的兩張影像資訊，因此在視差空間圖像的建構上會修正如下(19)所示：

$$C(x, y, d) = \min\{|I_L(x+d, y) - I_C(x, y)|, |I_C(x, y) - I_R(x-d, y)|\} \quad (19)$$

此外，若輸入資料為視訊，而非單純的影像，DERS 還會加上**時間穩定性分析 (Temporal Stability)** 來做些微的修正，不過這裡就不多加詳談，可以參考[10]。

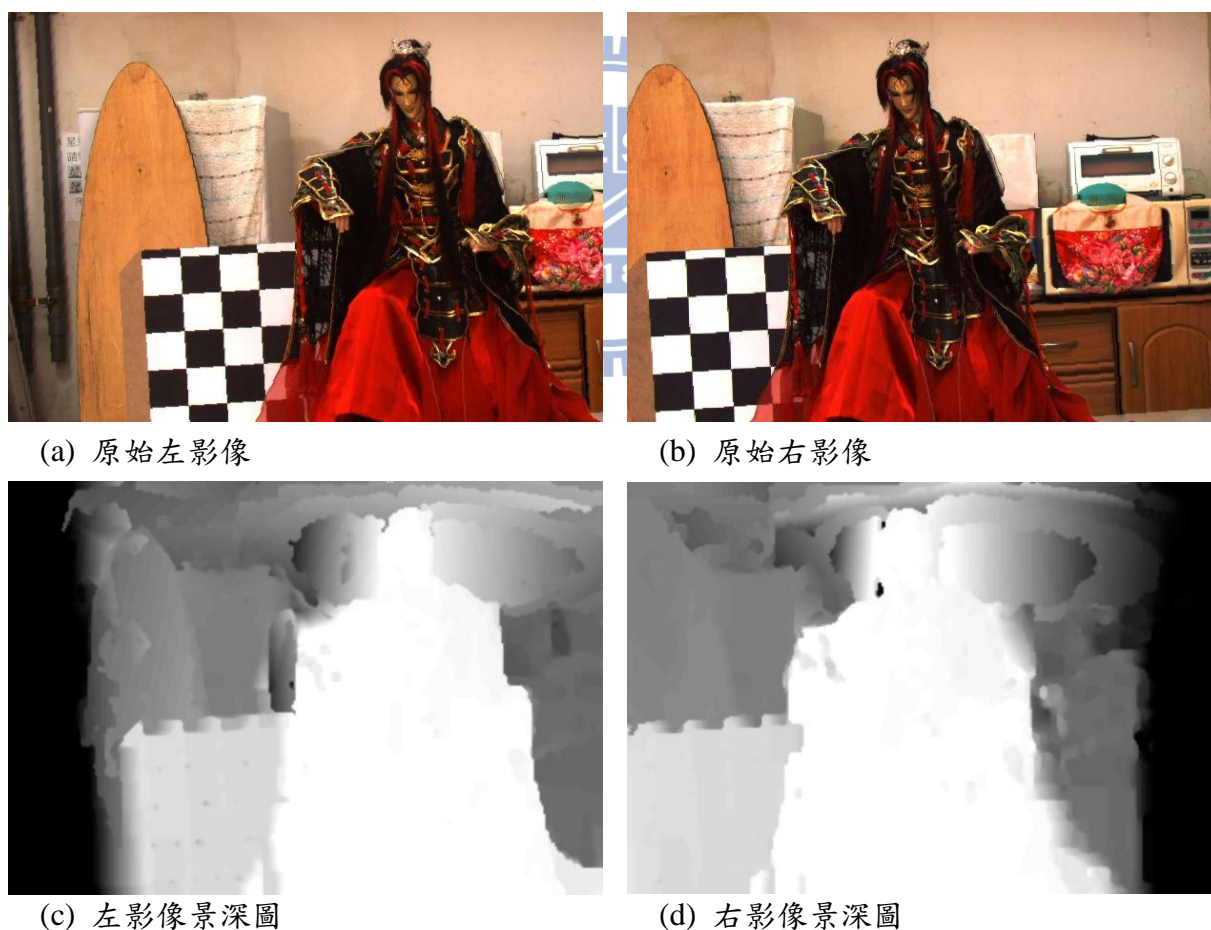


圖 16 已校正的左右兩影像和對應的景深圖

2.6 紋理映射 (Texture Warping)

紋理映射是影像式合成 (Image Based Rendering) [16] 的一個重要技術之一。早期的做法是利用 3×3 轉換矩陣來達到紋理映射[25]，但是卻只能假設物體在空間中的位置為平面。而至近期發展了景深圖式合成技術 (Depth Image Based Rendering, 簡稱 DIBR) [1][6][14][26]，它是利用景深資訊，將影像資訊重建立體模型，再投影至想要的視點上，如 (20)(21)所示：

$$\begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = z_r Q_r^{-1} \begin{bmatrix} u_r \\ v_r \\ 1 \end{bmatrix} + c_r \quad (20)$$

$$z_v \begin{bmatrix} u_v \\ v_v \\ 1 \end{bmatrix} = Q_v \left(\begin{bmatrix} X \\ Y \\ Z \end{bmatrix} - c_v \right) \quad (21)$$

其中下標的 r 和 v 分別代表參考資訊和虛擬資訊、 $Q[I|-c]$ 為相機參數、而 (u, v) 和 (X, Y, Z) 分別代表影像座標和對應的世界座標。若把 (20)(21) 式合併，則會變成：

$$z_v \begin{bmatrix} u_v \\ v_v \\ 1 \end{bmatrix} = z_r A \begin{bmatrix} u_r \\ v_r \\ 1 \end{bmatrix} + b \quad , \quad A = Q_v Q_r^{-1} \quad , \quad b = Q_v(c_r - c_v) \quad (22)$$

因此我們可以利用 (22)式，將參考影像位置 (u_r, v_r) 中的色彩資訊，直接映射到虛擬影像位置 (u_v, v_v) 。不過我們需要有景深資訊才能完成上述動作，依據景深資訊差異，會有兩種不同的映射方式。一個是利用參考影像的景深資訊，直接做紋理映射，稱做**正向紋理映射 (Forward Texture Warping)**；另一個則是利用虛擬影像的景深資訊，回頭擷取參考影像相對應座標上的色彩資訊，稱做**逆向紋理映射 (Backward Texture Warping)**。

一般來說，逆向紋理映射會比正向紋理映射還要來的好，因為正向映射會因為像素點量化的對應，而產生細長的破洞，並可能造成前景參雜背景雜訊等瑕疵現象。不過逆向紋理映射所需的虛擬視點景深圖相對上會比較難獲得，一般是直接利用參考景深圖來做估計；而且逆向紋理映射很容易造成前景重複出現的瑕疵現象，只是此現象可以利用遮蔽區域的辨識加以消除。如圖 17 所示：



圖 17 正向與逆向紋理映射設示意圖

2.7 線性疊合 (Linear Blending)

線性疊合是一種將若干張有共同區域的影像，根據前後景的關係做適當的疊合。在影像式合成中被常來使用的地方像是 **影像樞圖技術 (Image Matting)**[27]、**同心拼圖 (Concentric Mosaics)**[16]。影像疊合技術最大的困難點在於如何將影像中的前後景擷取出來，以其合成出不錯的效果。

而在 DIBR 領域下，因為有了景深圖資訊，裡頭已經包含了相對深度與遮蔽區域等資訊，使得線性疊合變得更加容易，如(23)式所示：

$$I_c(p) = \begin{cases} (1-\alpha) \cdot I_{L2C}(p) + \alpha \cdot I_{R2C}(p) & \text{if } occ_{L2C}(p) = 0 \text{ and } occ_{R2C}(p) = 0 \\ I_{L2C}(p) & \text{if } occ_{L2C}(p) = 0 \text{ and } occ_{R2C}(p) = 1 \\ I_{R2C}(p) & \text{if } occ_{L2C}(p) = 1 \text{ and } occ_{R2C}(p) = 0 \\ 0 & \text{if } occ_{L2C}(p) = 1 \text{ and } occ_{R2C}(p) = 1 \end{cases} \quad (23)$$

其中 $occ_{L2C}(p)$ 代表左圖對虛擬視點圖而言是否為遮蔽區域，是的話標示成 1，反之則 0 ($occ_{R2C}(p)$ 也是)。而 α 則是一個比例常數，若需線性疊合的視點和左右兩視點的相機位置是呈現水平且畫面共面的情況下， α 的選取會如 (24)式所示，也就是根據相機距離的遠近來做適當的比例疊合。

$$\alpha = \frac{|c_C - c_L|}{|c_C - c_L| + |c_C - c_R|} \quad (24)$$

不過在真實情況下，因紋理映射的位置有誤差，直接用(24)式做疊合容易造成影像模糊的現象。故我們會採用另外一種疊合方式：

$$I_c(p) = \begin{cases} I_{L2C}(p) & \text{if } depth_{L2C}(p) < depth_{R2C}(p) \text{ or } occ_{R2C}(p) = 1 \\ I_{R2C}(p) & \text{if } depth_{L2C}(p) > depth_{R2C}(p) \text{ or } occ_{L2C}(p) = 1 \\ 0 & \text{if } occ_{L2C}(p) = 1 \text{ and } occ_{R2C}(p) = 1 \end{cases} \quad (25)$$

(25)式代表著若左右兩張影像圖區域皆非遮蔽區域，則我們選取景深資訊離我們較近的紋理來替代。圖 18 展示出在紋理映射有誤差的情況下，分別使用(23)式 (如圖 18-(a))和(25)式 (如圖 18-(b))的結果。

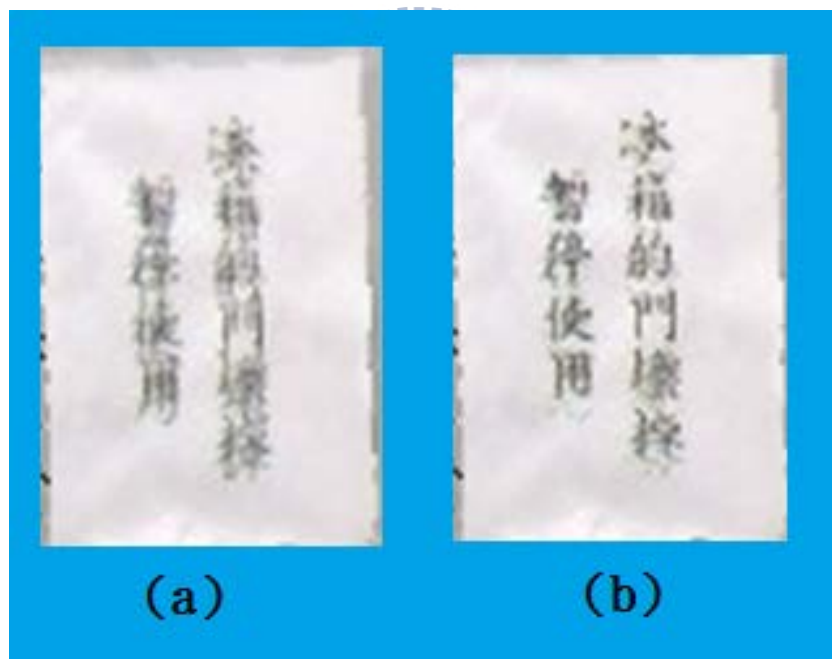


圖 18 兩種線性疊合方法示意圖。(a) 為(23)式，(b) 為(25)式。

2.8 影像修補 (Image Inpainting)

影像修補是影像處理中的一個技術，它的目的在於修復影像中已被破壞的小區域。而對於影像合成方面，因為會存在**遮蔽區域 (Occlusion Region)**，我們只能藉由周圍已知區域來估計被遮蔽的區域，因此影像修補技術會是一個不錯的解決方案之一。在自由視點合成技術上，影像修補技術也被廣泛的利用在資訊修復上 [6][14][28][29]。

影像修補最經典的作法是 Telea [30] 利用**快速行進演算法 (Fast Marching Method, 簡稱 FMM)**，在相對於更複雜的影像修補技術下可以有相似的修補成果，且運算速度很快。首先他是利用周圍已知區域，用一階泰勒展開近似來逼近未知區域：

$$I(p) = \frac{\sum_{q \in B_\varepsilon(p)} \omega(p, q) \cdot [I(q) + \nabla I(q)(p - q)]}{\sum_{q \in B_\varepsilon(p)} \omega(p, q)} \quad (26)$$

其中 p 為欲修補的區域座標、 $I(p)$ 為 p 上色彩強度值、 $B_\varepsilon(p)$ 為 p 點鄰域中已知色彩強度的點集合、 $\nabla I(q)$ 則是計算 q 點上的色彩強度梯度方向。至於 $\omega(p, q)$ 代表 p 和 q 點間的權重，越大代表越重要。所以這裡會衍生兩個問題。一個是待修復區域中，那些位置要先更新；另一個則是權重 $\omega(p, q)$ 如何定義。

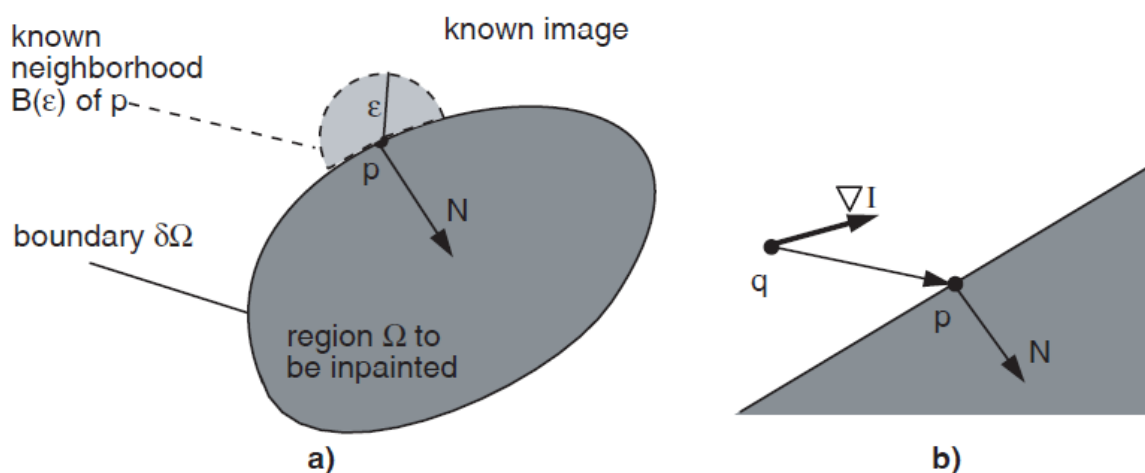


圖 19 影像修補原理 [30]

Telea 他的想法是利用物理上波動方程式的**光程函數方程 (Eikonal Equation)** 來模擬影像修補的先後次序，如 (27)式所示。程函方程是幾何光學上，光程和折射率間的關係。

$$|\nabla T(x)| = f(x) \quad \text{subject to } T(x)|_{x \in \partial\Omega} = 0 \quad (27)$$

(27)式中的 x 代表位置、 $T(x)$ 代表光程、而 $f(x)$ 代表折射率。意思是光從 $\partial\Omega$ 區域出發，穿過折射率為 $f(x)$ 的介質層而抵達 x 點位置，所需時間為 $T(x)$ 。簡單說可以想成是當油脂滴到水面上，其油脂在 $T(x)$ 時間點後，會擴散至 x ；而每一個時間點油脂所增加的表面積區域，就是影像需先修補的區域。若 $f(x)=1$ ，則 $T(x)$ 數值會等於 Ω 和邊界 $\partial\Omega$ 的距離，這時我們可以利用快速行進演算法 (FMM) 來近似 (27)式的解。因此只要先用**距離轉換 (Distance Transform, 簡稱 DT)**，將二值化的圖轉成距離圖 $T(x)$ ，再利用(26)式，從 $T(x)$ 最小的 x 處不斷地修補 $\partial\Omega$ ，且不斷地更新 $T(x)$ ，即可完成 FMM 演算法。

至於權重的定義，Telea 他共考慮三個因子，分別是**方向成分 (Directional Component)** $dir(p,q)$ 、**幾何距離 (Geometric Distance)** $dst(p,q)$ 、**階層成分 (Level Set Component)** $lev(p,q)$ 。方向成分代表亮度梯度值越接近法向量的方向，則權重越大；幾何距離代表已知亮度值越接近待修補區域，貢獻越大；而階層成分代表離邊界 $\partial\Omega$ 越近，貢獻越大。

$$dir(p,q) = \frac{p-q}{\|p-q\|} \cdot N(p) \quad (28)$$

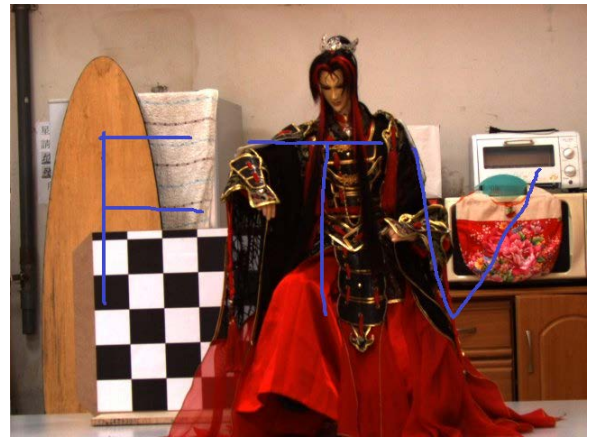
$$dst(p,q) = \frac{d_0^2}{\|p-q\|^2} \quad (29)$$

$$lev(p,q) = \frac{T_0}{1+|T(p)-T(q)|} \quad (30)$$

$$\omega(p,q) = dir(p,q) \cdot dst(p,q) \cdot lev(p,q) \quad (31)$$



(a) 原始影像



(b) 待修補的影像



(c) 影像修補結果



(d) 白色區域為被破壞的地方

圖 20 影像修補示意圖

Chapter 3 正向景深映射誤差分析 (Forward Depth Warping Error Analysis)

在自由視點合成下，景深資訊扮演著很重要的腳色，相對的，若景深資訊沒有善加處理，容易影響到最後的合成品質。雖然有相關論文探討紋理映射完的誤差分析[31][32][33][34]，但多半並沒有正確地將瑕疵區域補上、或只使用一些常見的後製濾波器，例如**中值濾波器 (Median Filter)**、或**雙邊濾波器 (Bilateral Filter)**，而造成邊緣的景深資訊被破壞掉。有些文獻論文中對於景深圖映射完後，更只有使用比中值濾波器還要差的其它簡單的濾波器，如**極值濾波器**[35]。因此本章會針對景深映射演算法所產生的各種誤差加以探討，並發展一套演算法來修正這些瑕疵現象。

3.1 平行映射下的誤差分析 (Parallel Warping Error Analysis)

3.1.1 正向景深映射演算法簡介 (Overview of Forward Depth Warping)

在第二章的 2.6 小節中有介紹紋理映射的一般性公式，如下所示：

$$z_v \begin{bmatrix} u_v \\ v_v \\ 1 \end{bmatrix} = z_r A \begin{bmatrix} u_r \\ v_r \\ 1 \end{bmatrix} + b \quad , \quad A = Q_v Q_r^{-1} \quad , \quad b = Q_v (c_r - c_v) \quad (32)$$

若只考慮映射方向跟兩相機的基線方向相同，且為平行映射，則可利用針孔相機模型的假設，將 (32)式簡化為如下(33)式：

$$x_v = \frac{\alpha_v}{\alpha_r} \cdot x_r + \frac{y_r - c_y}{\alpha_r \cdot \beta} \cdot \begin{vmatrix} \alpha_r & \alpha_v \\ s_r & s_v \end{vmatrix} + \frac{1}{\alpha_r} \cdot \begin{vmatrix} \alpha_r & \alpha_v \\ c_{x,r} & c_{x,v} \end{vmatrix} + \frac{\alpha_v \cdot (t_{x,v} - t_{x,r})}{z_r} \quad (33)$$

其中 參考相機參數為

$$\begin{bmatrix} \alpha_r & s_r & c_{x,r} \\ 0 & \beta & c_y \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} r_x^T & t_{x,r} \\ r_y^T & t_y \\ r_z^T & t_z \end{bmatrix} \quad (34)$$

而虛擬相機參數為

$$\begin{bmatrix} \alpha_v & s_v & c_{x,v} \\ 0 & \beta & c_y \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} r_x^T & t_{x,v} \\ r_y^T & t_y \\ r_z^T & t_z \end{bmatrix} \quad (35)$$

假設進一步讓兩相機內部參數中的焦距 ($\alpha_r = \alpha_v$) 與歪斜常數 ($s_r = s_v$) 一樣，則 (33)式可再度被簡化為：

$$x_v = x_r + (c_{x,v} - c_{x,r}) + \frac{\alpha \cdot (t_{x,v} - t_{x,r})}{z_r} = x_r + d(z_r) \quad (36)$$

(36)式正好為本篇論文的 2.5.1 小節所介紹的視差定義 (12)式，差別在於內部參數中，主軸點投影完後的水平座標相對定義量。一般的景深圖式影像合成技術也會以 (36)式為出發點來做紋理映射[14]。若不滿足該假設，我們也可利用幾何校正的概念，讓校正完後的內部參數一致。

而在真實情況下，影像是以像素為單位做量化切割，因此視差值會用四捨五入的運算子來操作，亦即 (36)式可近似成：

$$\tilde{x}_v = x_r + \text{round}\{d(z_r)\} \quad (37)$$

此外，若有超過一個像素點的景深，映射到同一個虛擬影像的位置，則我們會選取景深值最小的景深值來取代，亦即在空間中選取離影像平面最近的物體點，如 (38)式所示。

$$z_v(u, v) = \min_x \left\{ z_r \left(u - \text{round} \left\{ d \left(z_r(x, v) \right) \right\}, v \right) \right\} \quad (38)$$

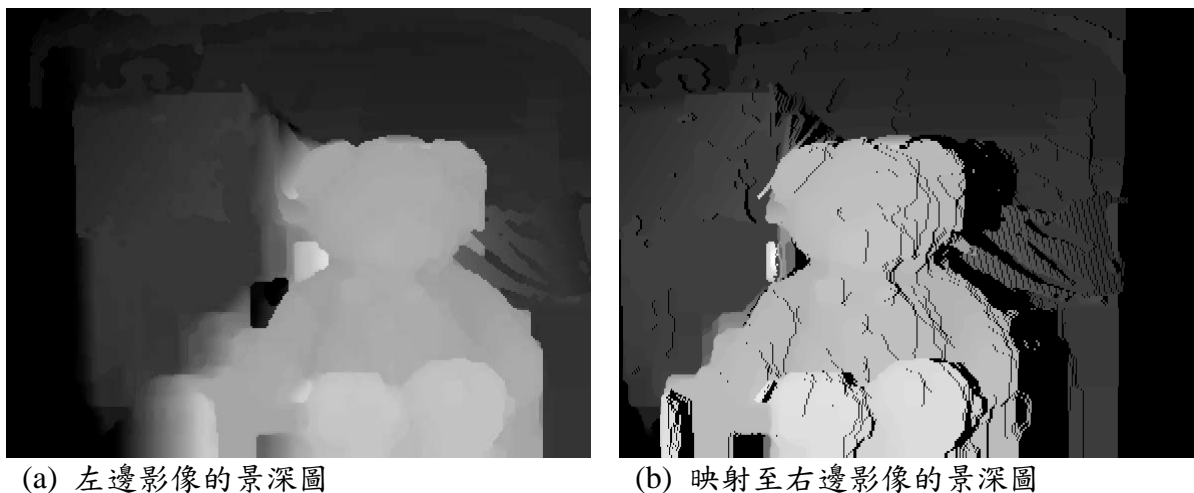


圖 21 景深映射示意圖

3.1.2 映射誤差分析 (Warping Error Analysis)

由圖 21 可觀察到景深圖映射完後會產生不規則形的大小破洞，這是因不連續的景深資訊所造成的現象，我們把這種破洞現象稱做**競爭條件 (Race Condition)**。就跟數位電路中訊號會有競爭問題而產生資料危害 (Data Hazard) 的產生原因是一樣，只是對影像合成而言，這種競爭問題所產生的後果會是**破洞**。

要了解破洞的成因與組成前，須先分析映射公式所造成的影響。在理想情況下，由(36)式可看出當 $x_r, z_r > 0$ ， x_v 是 x_r 與 z_r 的函數，且因 x_r 也可以視為 z_r 的函數，所以我們根據 $x_r(z_r)$ 的連續性，來分析 x_v 的連續性。若 $x_r(z_r)$ 為連續函數，則 $x_v(x_r, z_r)$ 亦為連續函數；反之若 $x_r(z_r)$ 不為連續函數，則 $x_v(x_r, z_r)$ 非連續函數。這意味著一個連續物體若沒有被前景給遮蔽住，則紋理映射在理想狀況下理應會保持連續性；若有物體發生不連續的現象 (例如：邊緣與前後景的關係)，則映射後的紋理會產生不連續的現象。此一特性跟直覺是相符的。

不過根據本篇論文的 2.5.2 小節可知，我們接收到的訊息是量化後的景深 $z_{q,r}$ ，而非真正的景深 z_r 。因此利用平行映射公式(37)、與量化公式(14)，可結合出系統運算時的真正景深映射公式：

$$\tilde{x}_v = x_r + \text{round}\{d(z_{q,r})\} \quad (39)$$

$$d(z_{q,r}) = \text{scaling} \cdot z_{q,r} + \text{shift} \quad (40)$$

其中
$$\text{scaling} = \frac{\alpha}{N-1} \cdot (t_{x,v} - t_{x,r}) \cdot (z_{near}^{-1} - z_{far}^{-1}) \quad (41)$$

$$\text{shift} = (c_{x,v} - c_{x,r}) + \alpha \cdot (t_{x,v} - t_{x,r}) \cdot (z_{far}^{-1}) \quad (42)$$

跟據(40)式可發現在平行映射演算法下，其視差會是量化景深的線性函數，並且係數只會跟相機參數和最遠與最近景深資訊有關。但即使是連續物體，會因景深量化的操作，使得 \tilde{x}_v 在像素單位的情況下，造成破洞現象。此外，視差又再一次被四捨五入至整數值，容易使邊緣造成瑕疵現象。因此我們可根據(39)式，將映射誤差分成兩類：<1> **像素點誤差 (Pixel Errors)**，由正向映射像素點所造成。 <2> **邊緣誤差 (Boundary Errors)**，由四捨五入算符所造成。邊緣誤差因在像素點下就會存在的本質問題，只能藉由最佳化將之壓抑。因此以下將會加以探討像素點誤差的各種型態與組成成因。

3.1.3 競爭條件 (Race Condition)

前小節分析出破洞是因像素點誤差所造成。以下將進一步探討哪些情況會產生破洞、與破洞所造成的影響。為了方便起見，首先定義從左影像映射至右影像，稱作 $L2R$ ；而右影像映射至左影像，則稱作 $R2L$ 。接著我們根據連續物體與不連續物體，加上遞增遞減現象來加以分類。如下表 1 與表 2 所示：

表 1 左至右方向的景深圖資訊

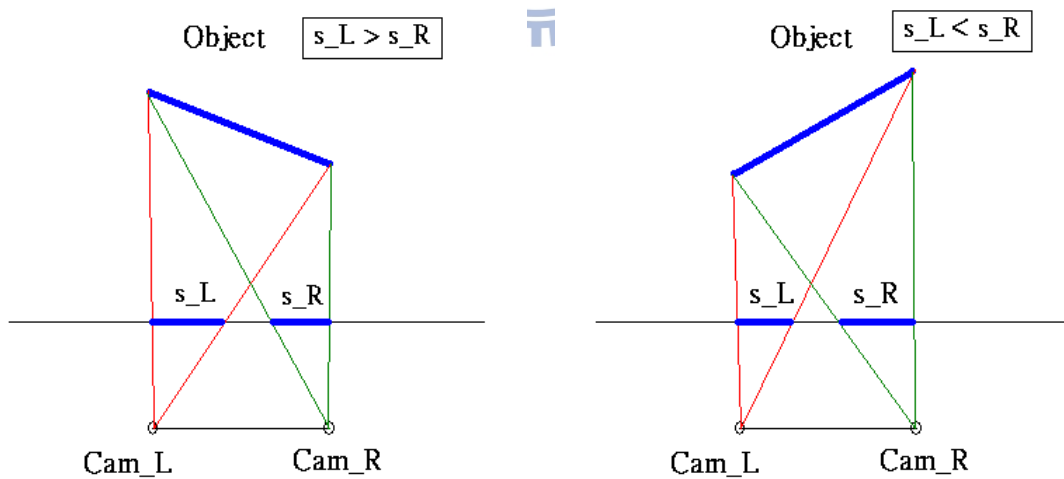
Cases	Comments
$L2RCF$	左至右方向的原始景深圖區域為連續平坦
$L2RMI$	左至右方向的原始景深圖區域為單調遞增
$L2RMD$	左至右方向的原始景深圖區域為單調遞減
$L2RBF$	左至右方向的原始景深圖區域為背景至前景
$L2RFB$	左至右方向的原始景深圖區域為前景至背景

表 2 右至左方向的景深圖資訊

Cases	Comments
$R2LCF$	右至左方向的原始景深圖區域為連續平坦
$R2LMD$	右至左方向的原始景深圖區域為單調遞減
$R2LMI$	右至左方向的原始景深圖區域為單調遞增
$R2LFB$	右至左方向的原始景深圖區域為前景至背景
$R2LBF$	右至左方向的原始景深圖區域為背景至前景

我們以映射的方向，來定義出原始景深圖的單調性，例如 CF 代表連續平坦 (Continuous Flat)； MI 和 MD 分別代表單調遞增 (Monotonic Increasing) 和單調遞減 (Monotonic Decreasing)； FB 和 BF 則代表前景至背景 (Foreground-to-Background) 與背景至前景 (Background-to-Foreground)。因此像 $L2RMI$ ，就代表著當左參考影像映射至右影像時，我們關心左參考影像的景深圖中，從左至右屬於單調遞增的區域在映射至右影像後所造成的影響。

由於對稱性，只需探討表 1 中的五種可能性，而表 2 則直接類比推得相同的性質；並且根據(39)式，可以得知 $L2RCF$ 區域在映射後還是會保持其連續平坦的特性，而沒有破洞的生成。因此我們分別模擬其它四種映射的結果：



(a) L2RMI 示意圖

(b) L2RMD 示意圖

圖 22 L2RMI (左) 與 L2RMD (右) 示意圖

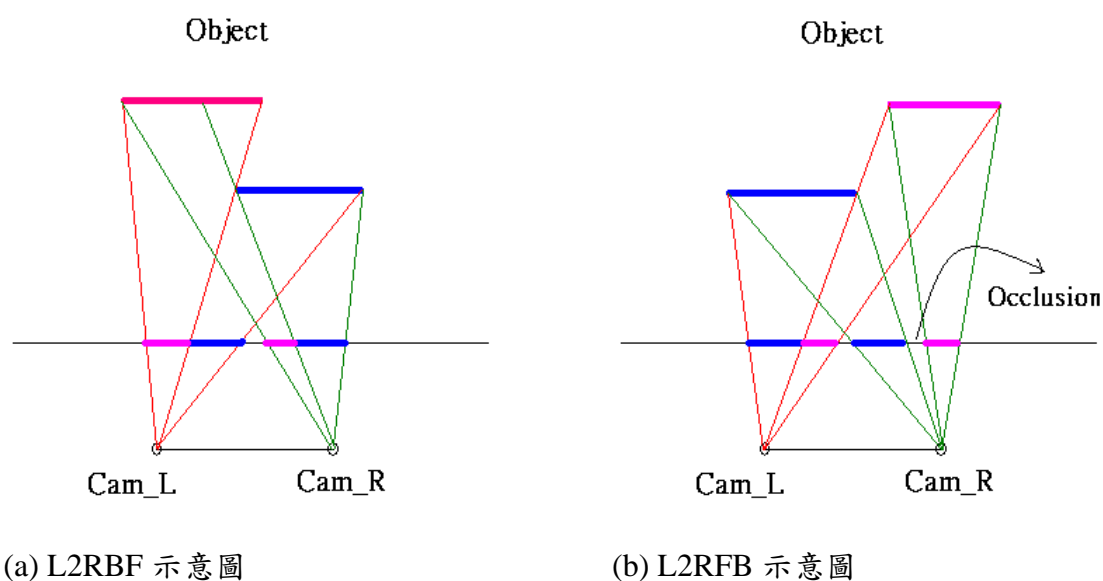


圖 23 L2RBF (左) 與 L2RFB (右) 示意圖

由圖 22 可知， $L2RMI$ 在映射完後的結果，其資料量會減縮，所以可能會產生鋸齒 (Aliasing) 問題；而 $L2RMD$ 則是映射完後資料量會膨脹，但是因正向映射為函數型對應，可預期映射後會存在某些區域，使得景深資訊無法與原始景深圖做對應，而這些區域就是破洞產生的成因。

至於圖 23 則模擬出不連續的景深圖中，其映射後的結果。首先是 $L2RBF$ ，乍看之下在映射完後並沒有破洞或瑕疵現象產生，但若前景 (圖 23-(a) 中較靠近 Cam_R 的水平區域) 並非連續平坦曲線，而是單調遞減趨勢，那麼前景會因前面所分析的 $L2RMD$ 破洞生成，使得背景區域有可能被填補其中，造成景深資訊與原始景深圖做假性的對應；最後是 $L2RFB$ ，我們發現該區域映射完後會產生大洞，而這大洞是因為左影像景深資訊不存在的區域所造成，也稱作**遮蔽區域 (Occlusion)**。

結合上述討論，可以總結出映射後各種可能性與瑕疵現象，如表 3 所示：

表 3 景深資訊映射後可能誤差分析表

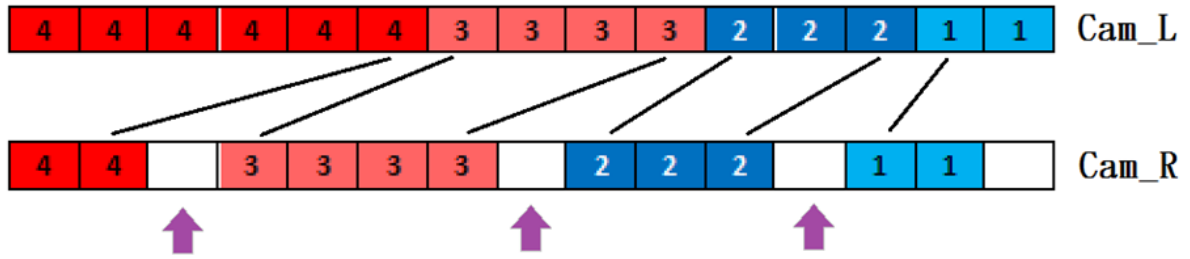
參考影像		(左至右) 映射後影像	
	Cases	Length	Artifacts
連續區域	L2RCF	不變	無
	L2RMI	變短	鋸齒狀
	L2RMD	變長	小洞
不連續區域	L2RBF	變短	前景區域混雜背景區域
	L2RFB	變長	大洞 (遮蔽區域)

根據表 3，可以得知 L2RMD、L2RBF、與 L2RFB 所產生的瑕疵，相較於 L2RMI 現象更為嚴重許多，因此我們會將重心放在這三種區域上，並且在後續章節提出改善方式。首先定義這三種瑕疵現象的名稱，如表 4 所示：

表 4 深資訊映射後各種瑕疵現象

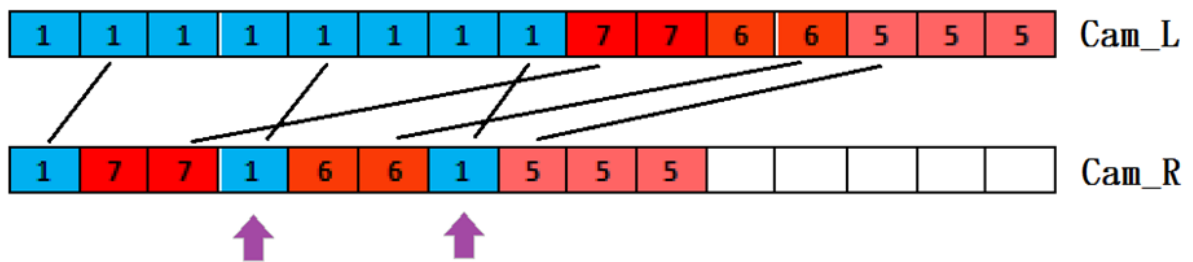
映射誤差	發生區域處	成因
Quantization Error: Type I	L2RMD	一對一函數對應，導致虛擬景深圖破洞
Quantization Error: Type II	L2RBF	QE Type I 中有背景參雜其中
Occlusion Region	L2RFB	未知區域被前景遮蔽

我們把瑕疵主要切割成兩塊，分別是造成小破洞的**量化誤差 (Quantization Errors)**，與造成大破洞的**遮蔽區域 (Occlusion Region)**。而量化誤差又可以區分兩種，一種是單純的空洞 (**Type I**)、與空洞被其它錯誤的景深資訊填補 (**Type II**)。圖 24~圖 26 分別是以上三種誤差的示意圖。



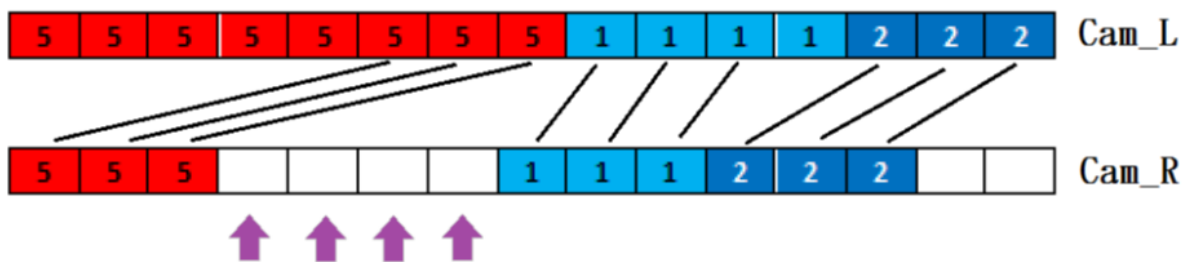
☆ Quantization Error Type I

圖 24 量化誤差型態 I (Quantization Error: Type I) 示意圖



☆ Quantization Error Type II

圖 25 量化誤差型態 II (Quantization Error: Type II) 示意圖



☆ Occlusion Region

圖 26 遮蔽區域 (Occlusion Region) 示意圖

3.2 平行景深映射下的誤差修復 (Parallel Depth Warping Refinement)

根據 3.1 章節的誤差分析可知道量化誤差型態 I 和 II 是因為正向映射的一對一函數對應，導致有未被對應到的破洞生成；但理想狀態下應只會存在遮蔽區域。因此接下來將會提供一個演算法來修復量化誤差，其中發展了一套可信賴模型，根據這個模型來做適當的量化誤差修復工作。

3.2.1 可信賴模型建構 (Reliability Model Construction)

可信賴模型 (Reliability Model)，是指數值運算因受到各種雜訊影響，使得最後的結果可能因誤差太大而無法使用，因此需要估測雜訊的影響程度，並且判斷該結果能不能採納。而這一套估測的標準，就是可信賴模型。一般來說，我們會利用機率模型來做適當的假設，並且推導出適當的估計器 (Estimator) 來使用，但本篇論文會採用直觀的角度，推出合理的估計器。

根據前一小節的表 4，會發現不論是量化誤差、或是遮蔽區域，其原始景深區域的特性，對於映射方向皆有**單調遞減**的趨勢，差別在於區域有無連續。所以我們的想法是偵測景深圖中的連續區域，並且判斷該區域是否為單調遞減。因此接下來會根據連續與單調遞減來設計兩種估計器，而利用這兩種估計器所產生出來的圖分別稱做**連續圖 (Continuity Map)**、與**下降圖 (Descending Map)**。

連續圖 (Continuity Map) 是一種二值化的影像，用來判斷原始景深圖區域是否具有連續景深特性。我們可以直接針對真正的景深範圍做均勻量化 (Uniform Quantization)。若相鄰像素點的景深差值小於步長 (Step Size)，代表兩像素點為連續點，此時標記為 1；反之則標記為 0。如 (43) 式所示。

$$cont_{map}(u, v) = \begin{cases} 1 & \text{if } \varepsilon z_r(u, v) \leq thread_{cont} \\ 0 & \text{if } \varepsilon z_r(u, v) > thread_{cont} \end{cases} \quad (43)$$

$$\varepsilon z_r(u, v) \equiv \frac{1}{|z_{far} - z_{near}|} \cdot \left| \frac{\partial z_r(u, v)}{\partial u} \right| \quad (44)$$

其中 $cont_{map}(u, v)$ 為二值化的連續圖、而 $thread_{cont}$ 則是用來決定估計器 (44) 式的門檻，通常會取 1% ~ 2%。

下降圖 (Descending Map) 它也是一種二值化的影像，用來判斷原始景深圖區域是否具有單調遞減的特性。跟連續圖差不多，若相鄰像素點的視差²差值 (以映射方向為參考) 為正值，代表具有遞減特性，而標記為 1；反之則標記為 0。如(45)所示。

$$desc_{map}(u, v) = \begin{cases} 1 & \text{if } 0 < \varepsilon d(u, v) \leq thread_{desc} \\ 0 & \text{if otherwise} \end{cases} \quad (45)$$

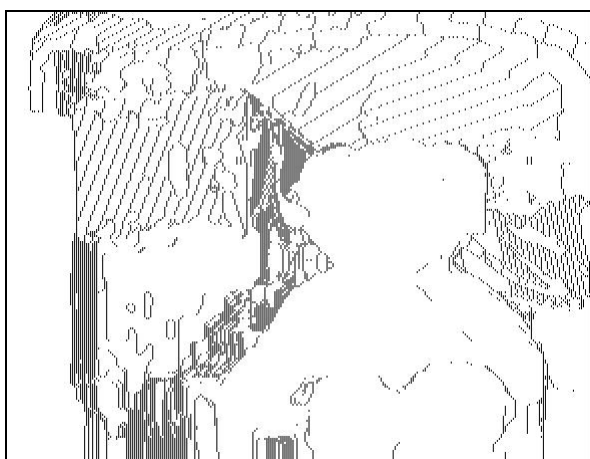
$$\varepsilon d(u, v) \equiv \frac{\partial d(z_r(u, v))}{\partial u} \quad (46)$$

其中 $desc_{map}(u, v)$ 為二值化的下降圖、而 $thread_{desc}$ 用來決定估計器(46)式的門檻。會需要訂立此門檻的原因是，若視差差值過大，很有可能是遮蔽區域發生處，我們不希望將它納入遞減區域的考量，不過該門檻與連續圖的用處有所交集，因此即使將下降圖的門檻放寬鬆，到最後面的誤差修復演算法也不會有過大影響。至於其門檻數值，則依實驗數據做適當調整。

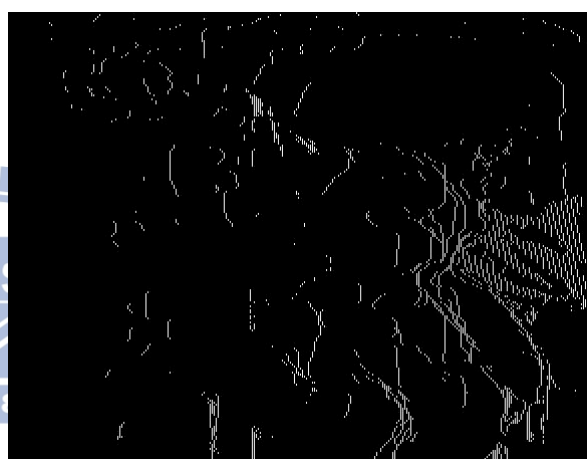
² 這裡的視差在本篇論文中為一相對量，而非絕對正值。若像素往左移動(左影像向右映射)，則視差為負值；反之若像素往右移動(右影像向左映射)，則視差為正值。可參考(36)式來驗證。



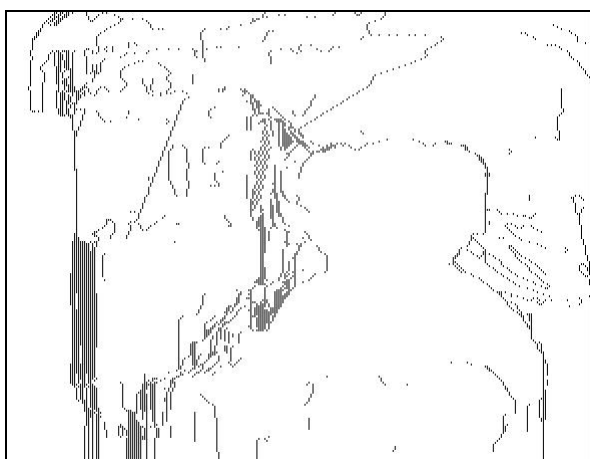
(a) 參考景深圖



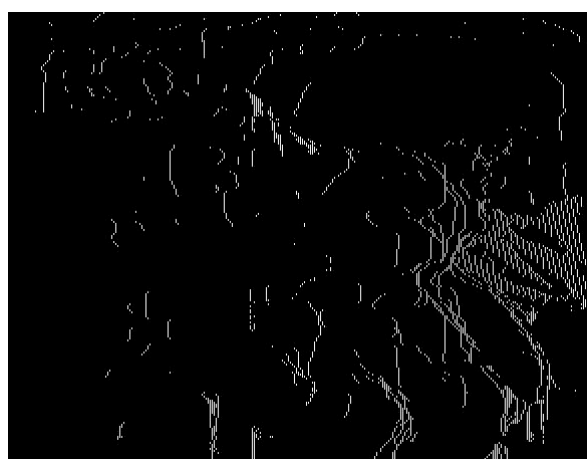
(b) 連續圖 (Continuity Map), $th=1\%$



(c) 下降圖 (Descending Map), $th=5$



(d) 連續圖 (Continuity Map), $th=2\%$



(e) 下降圖 (Descending Map), $th=10$

圖 27 連續圖與下降圖的範例 (白色區域代表標示為 1)

3.2.2 平行映射後景深修復演算法 (Parallel Warped Depth Refinement)

有了前一小節所建立出來的可信賴模型，接下來本小節會介紹如何利用該模型將景深映射後的量化誤差修復。首先把參考景深圖中的連續遞減區域全部擷取出來，利用 (43)和 (45)式將之取交集後，可以很容易得到這些點的集合為 Ω ，其中：

$$\Omega = \{(x, y) \mid cont_{map}(x, y) = 1 \text{ and } desc_{map}(x, y) = 1\} \quad (47)$$

接下來，對任意參考景深影像中的像素點位置 $p(x_r, y_r) \in \Omega$ ，我們分別計算出 $p(x_r, y_r)$ 和相鄰點 $p(x_r + 1, y_r)$ 經由正向映射後所落在虛擬影像中的水平位置座標 $x_{v,fast}$ 與 $x_{v,slow}$ ，如下所示：

$$x_{v,fast} = x_r + d(z_r(x_r, y_r)) \quad (48)$$

$$x_{v,slow} = x_r + 1 + d(z_r(x_r + 1, y_r)) \approx x_r + 1 + d(z_r(x_r, y_r)) + \varepsilon d(x_r, y_r) \quad (49)$$

其中 (49)式可以利用 (46)式來做估計³。而下標分別標示 *fast* 和 *slow*，代表 $p(x_r, y_r)$ 和 $p(x_r + 1, y_r)$ 這兩個像素點在映射完後，之間一定會發生量化誤差所造成的破洞，因為 $p(x_r, y_r)$ 在影像中的水平線上的”移動”的比 $p(x_r + 1, y_r)$ 還要快⁴，而之間的破洞我們必須要利用此兩點的景深資訊來做修補。本篇論文是採用線性內插的動作，如 (50)式所示：

³ 在數值運算上，(49)式左右兩邊等號會成立，亦即：

$$\varepsilon d(u, v) \equiv \frac{\partial d(z_r(u, v))}{\partial u} = d(z_r(u + 1, v)) - d(z_r(u, v))$$

⁴ 若左影像景深映射至右視點，則 $p(x_r, y_r)$ 的移動量會比 $p(x_r + 1, y_r)$ 還要來的大；但是若右影像景深映射至左視點， $p(x_r, y_r)$ 的移動量反而會比 $p(x_r + 1, y_r)$ 還要來的小。只是本篇論文是以左至右為主，因此命名規則或解讀上會以它為準。但最後的結論是相通的。

$$z_v(i, y_r) = z_r(x_r, y_r) + \frac{(i - x_{v,fast}) \cdot [z_r(x_r + 1, y_r) - z_r(x_r, y_r)]}{1 + \varepsilon d(x_r, y_r)} \quad \forall i \in (x_{v,fast}, x_{v,slow}) \quad (50)$$

將所有屬於 Ω 的像素點做過 (50) 式的內插動作後，我們可以保證在不更動到遮蔽區域的情況下，而將量化誤差型態 I 與型態 II 都修復掉。圖 28 為量化誤差修復示意圖。

值得注意的是，目前在自由視點合成系統上的破洞修補，有些只考慮到量化誤差型態 I 的區域，並且是在量化後的景深圖下做線性內插。但由上一章節 2.5.2 可知景深圖的量化並非是單純的線性關係，若在量化後的景深圖下做內插，其實並不太合理，因為尚須考慮到相機參數的影響。因此在真正的景深資訊下做內插動作，也可避免相機內部參數的影響。

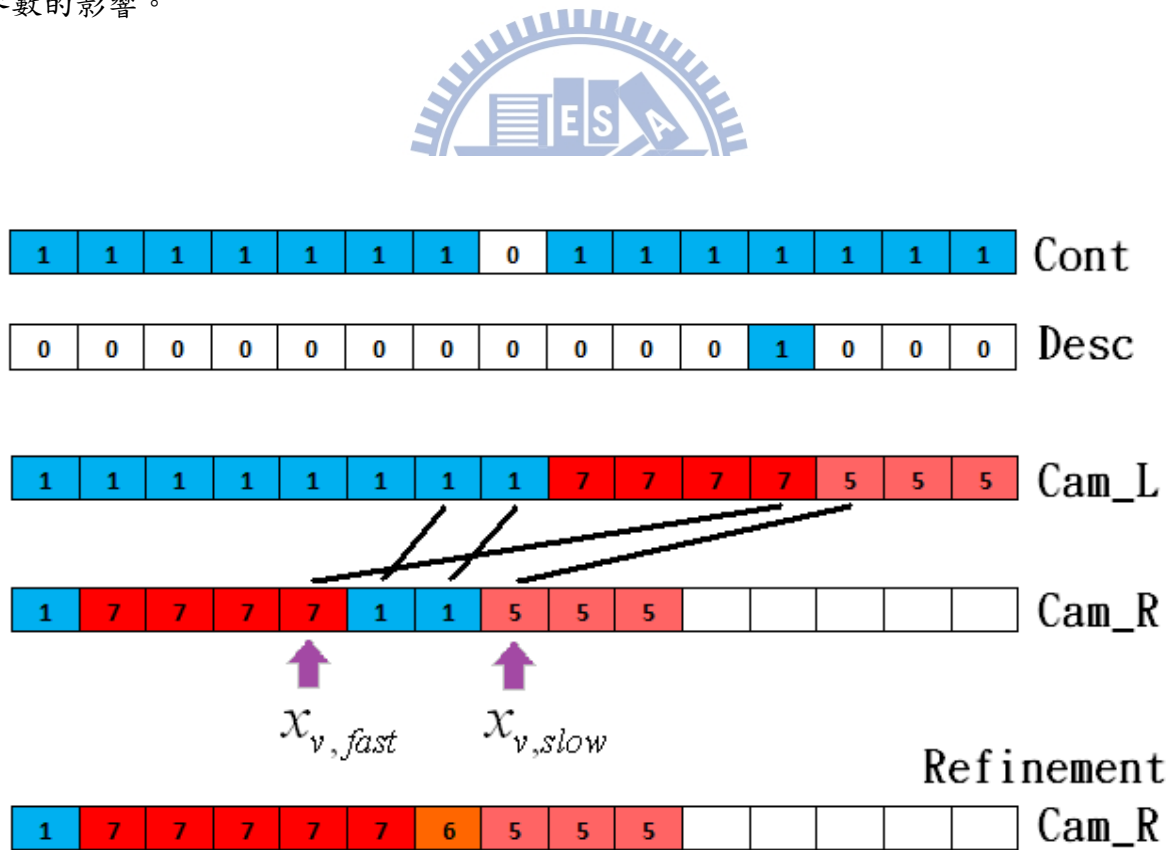


圖 28 景深圖映射後修復示意圖

3.2.3 實驗結果與討論 (Simulation Results and Discussion)

以下將展示在不同門檻的可依賴模型下，其映射後的景深圖修復結果：



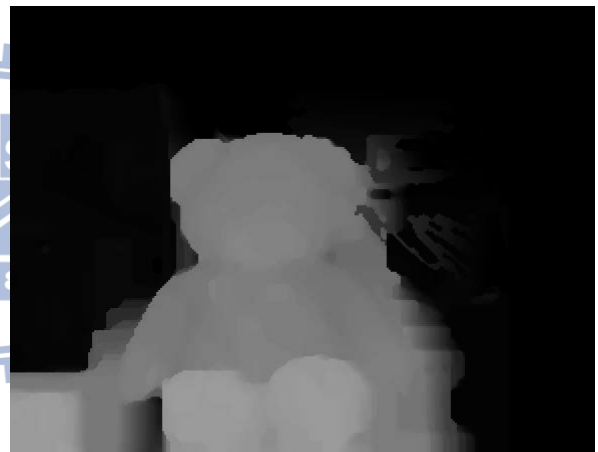
(a) 左邊參考影像



(b) 右邊參考影像



(c) 左邊參考影像景深圖



(d) 右邊參考影像景深圖



(e) 左至右的虛擬映射景深圖

圖 29 原始參考影像、景深圖、與映射後的景深圖



(a) $thread_{cont} = 1\%$ & $thread_{desc} = 100$



(b) $thread_{cont} = 2\%$ & $thread_{desc} = 100$



(c) $thread_{cont} = 3\%$ & $thread_{desc} = 100$



(d) $thread_{cont} = 4\%$ & $thread_{desc} = 100$



(e) $thread_{cont} = 5\%$ & $thread_{desc} = 100$



(f) $thread_{cont} = 100\%$ & $thread_{desc} = 100$

圖 30 連續圖門檻對於平行映射後景深修復演算法的影響



(a) $thread_{cont} = 100\%$ & $thread_{desc} = 1$



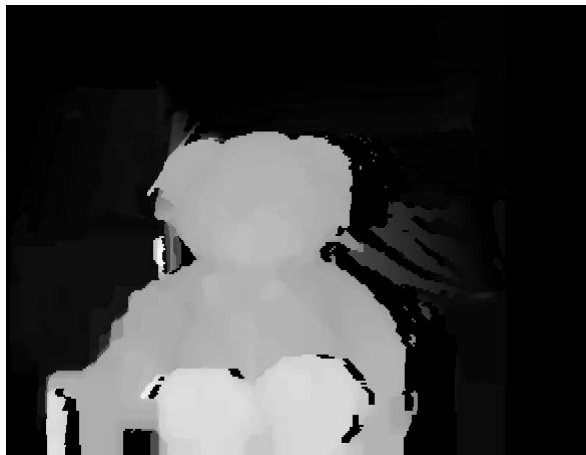
(b) $thread_{cont} = 100\%$ & $thread_{desc} = 2$



(c) $thread_{cont} = 100\%$ & $thread_{desc} = 3$



(d) $thread_{cont} = 100\%$ & $thread_{desc} = 4$



(e) $thread_{cont} = 100\%$ & $thread_{desc} = 5$



(f) $thread_{cont} = 100\%$ & $thread_{desc} = 20$

圖 31 下降圖門檻對於平行映射後景深修復演算法的影響 (一)



(a) $thread_{cont} = 2\%$ & $thread_{desc} = 1$



(b) $thread_{cont} = 2\%$ & $thread_{desc} = 2$



(c) $thread_{cont} = 2\%$ & $thread_{desc} = 3$



(d) $thread_{cont} = 2\%$ & $thread_{desc} = 4$



(e) $thread_{cont} = 2\%$ & $thread_{desc} = 20$



(f) $thread_{cont} = 2\%$ & $thread_{desc} = 100$

圖 32 下降圖門檻對於平行映射後景深修復演算法的影響 (二)

我們總共做了三個實驗。第一個實驗是只考慮連續圖對於誤差修復演算法的影響，所以設定下降圖的門檻至 100。如圖 30 所示。會發現當連續圖的門檻越來越寬鬆，雖然小洞都被修補起來，但卻隨著門檻越大，而慢慢侵蝕掉遮蔽區域。這是因為該實驗的下降圖連遮蔽區域也包含在一起，使得對於寬鬆的連續圖門檻下，將遮蔽區域也視為連續區域，而造成補洞的動作。

第二個實驗則只考慮下降圖對於誤差修復演算法的影響，所以直接讓連續圖的門檻達 100%。如圖 31 所示。也可觀察出當下降圖的門檻越大，遮蔽區域也有被侵蝕掉的現象。這是因為沒有考慮到區域是否連續，而只對單調遞減的區域做補洞的動作，因此很顯然的，部分遮蔽區域也會被補起來。

最後第三個實驗同時考慮到連續圖與下降圖的影響，但也是固定連續圖的門檻下，觀察下降圖門檻對於演算法的影響，在此設定連續圖門檻為 2%。如圖 32 所示。我們可發現在給定適當的連續圖門檻下，即使下降圖門檻再高，也不會因此將遮蔽區域納入修補考量，這是因為連續圖門檻可以幫助於將不連續區域排除掉，也就是會排除掉遮蔽區域。

因此結論是，對於平行景深圖映射下，我們發展了一套演算法機制，可完全修補掉量化誤差；而不需要使用如中值濾波器來修補小洞，並且降低物體邊緣的景深被破壞的可能。

3.3 廣義映射下的誤差分析 (General Warping Error Analysis)

到 3.2 章節為止，我們雖然分析了景深圖映射所產生的各種瑕疵、並且提出改善方式，卻有個基本假設，那就是映射為平行式，而非廣義式的映射。目前相關的文獻論文也都只致力於平行式映射後的誤差分析與改進，且在稀疏多相機合成下，其效果不彰。我們嘗試對廣義映射下的景深圖各類瑕疵做分析，發現複雜程度遠大於平行映射下的瑕疵現象，這是因為廣義映射下的量化誤差並不單只發生於連續遞減區域上，對於連續平坦、甚至是連續遞增的區域上，廣義映射後的景深圖也會發生量化誤差。

不僅如此，即使發展了一套對廣義映射也通用的可信賴模型，對於後續的補洞也複雜許多，因為破洞不再侷限於一維且水平的問題，而是擴增到二維問題上。因此本篇論文並沒有再進一步研究廣義映射下的景深圖修復演算法。

但幸運的是，我們找出一個有別於正向景深圖映射演算法的映射方式 (Forward Depth Warping Algorithm, 簡稱 FDW)，稱做逆向景深圖映射演算法 (Backward Depth Warping Algorithm, 簡稱 BDW)。該映射演算法雖然沒有辦法將量化誤差完全修補，卻可大量減低量化誤差的區域。本篇論文會在 4.4 章節加以介紹與討論。

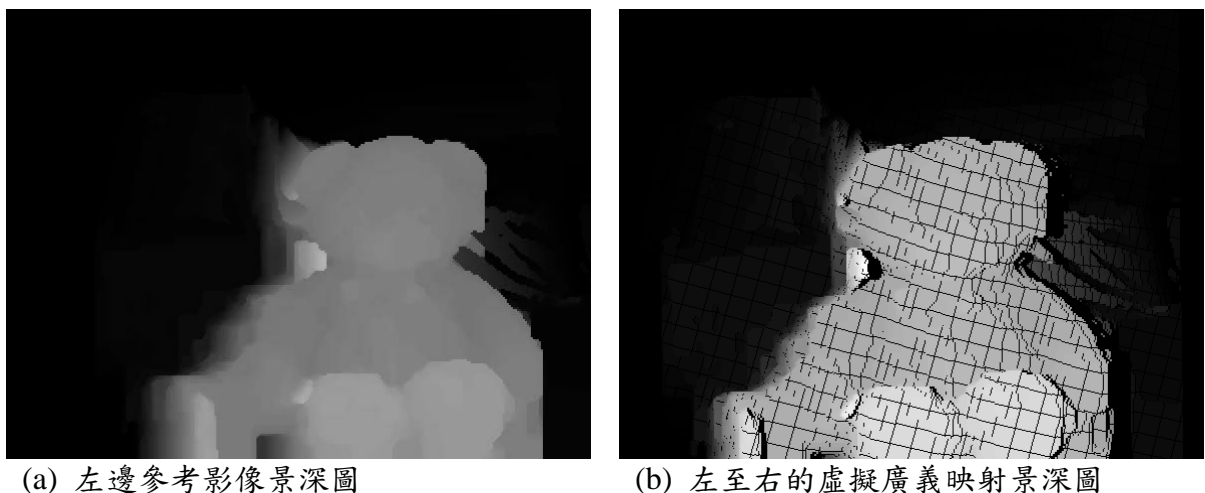


圖 33 廣義映射後的景深圖

Chapter 4 稀疏多相機虛擬視點合成設計與改進(Design and Improvement of Sparse Multi-Camera Virtual View Synthesis)

本篇論文最主要的貢獻，除了平行景深圖映射後的分析與改善外，還有根據傳統的自由視點合成演算法，發展出稀疏多相機虛擬視點合成演算法系統。我們針對稀疏視點合成所遇到的難題，在不同地方下一共提出四種演算法的改進，以其最後能有高品質的合成結果。本章 4.1 節會先介紹傳統的自由視點合成的大致架構。本章 4.2 節則會提出稀疏視點合成的架構。並且會在本章 4.3 小節 ~ 本章 4.6 小節，分別介紹本篇論文的改良方式。包含了 景深圖升取樣方法、逆向景深圖映射演算法、金字塔型影像修補技術、與 瑕疵抑制演算法。

4.1 傳統自由視點合成系統架構 (System Flow Chart of Traditional Free Viewpoint Television)

目前現有的自由視點合成演算法，主要可分成六個區塊。

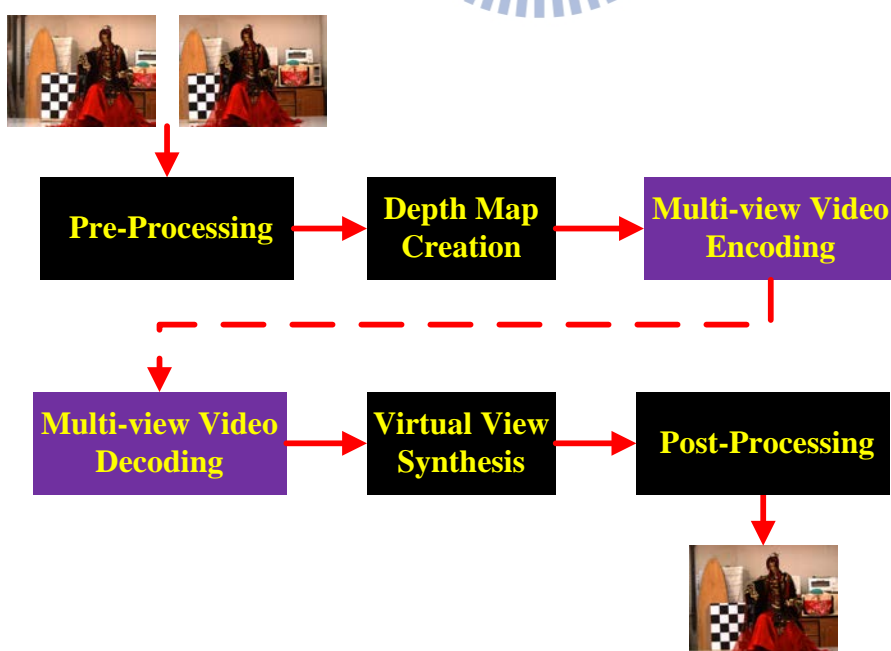


圖 34 傳統自由視點合成演算法架構圖

一開始傳送端會先將影像作**事前處理**，包括色彩校正與幾何校正，接著在做**景深圖估計**，並且做完**影像壓縮**後將資料傳送出去。接收端收到資料後，會對資料做**解壓縮**動作，並且做**虛擬視點合成**。最後再經由**事後處理**，把合成後的影像品質提高。由於本篇論文以影像合成為目的，我們不會介紹影像壓縮與解壓縮的部分。以下會簡介每個小系統的流程圖：

4.1.1 事前處理 (Pre-Processing)

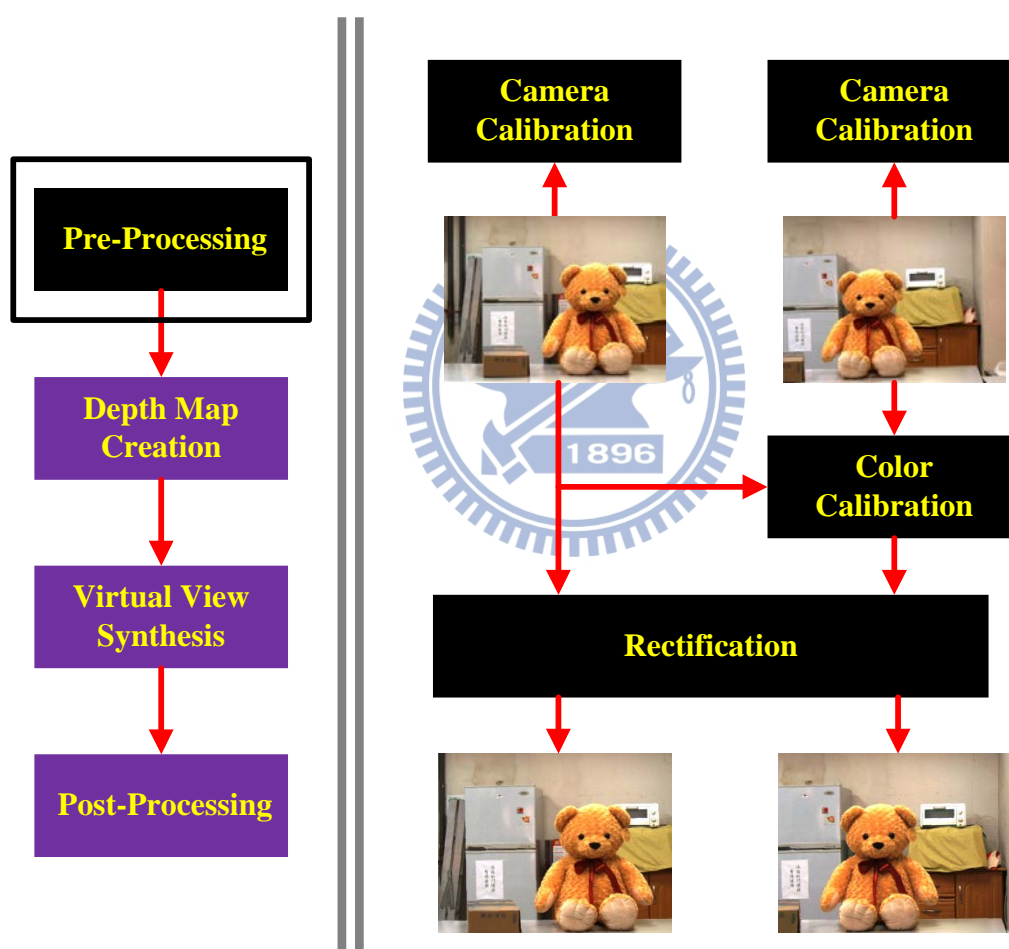


圖 35 事前處理系統流程圖 (Pre-Processing)

事前處理一開始會先做**相機校正**，用來估計每台相機的相機參數。由於光的散射等性質，需要用**色彩校正**將兩張顏色一致化。做完之後，再將影像經過**幾何校正處理**，以便接下來的景深估計能有效率的實現。

4.1.2 景深估計 (Depth Estimation)

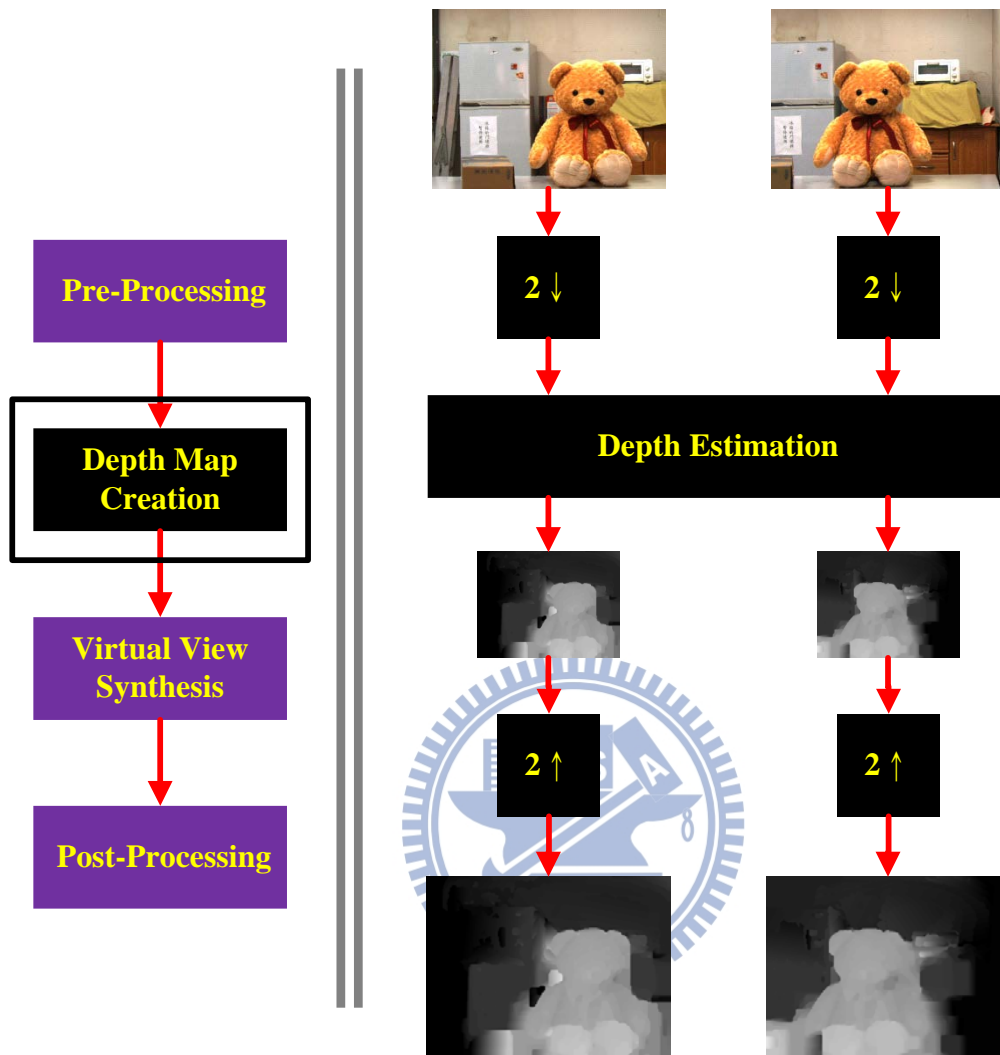


圖 36 景深估計系統流程圖 (Depth Estimation)

景深估計在此採用 2.5.3 小節所介紹的**景深估計參考軟體 (簡稱 DERS)**，並且將輸入端改成只需要左右兩張圖，而修改方式只要分別設定視差空間圖像 (19)式中，讓某一端的參考視差設定為無窮大的數值即可。

一開始會先做**降取樣 (Down Sampling)**；接續做完景深圖估計後，在對景深圖做**升取樣 (Up Sampling)**的動作，普遍原因是降低資料量的傳輸。不過針對稀疏系統架構而言，這樣子做的目的可減少景深估計的計算複雜度，因為對稀疏影像中的資訊而言，有些視差值會達到兩百個像素點以上，對於建立視差空間圖像反而不切實際。此外，一

般使用降取樣或升取樣的方法會採用高斯插值法 (Gaussian Interpolation)、或是雙三次插值法 (Bi-cubic Interpolation)。

4.1.3 虛擬視點合成 (Virtual View Synthesis)

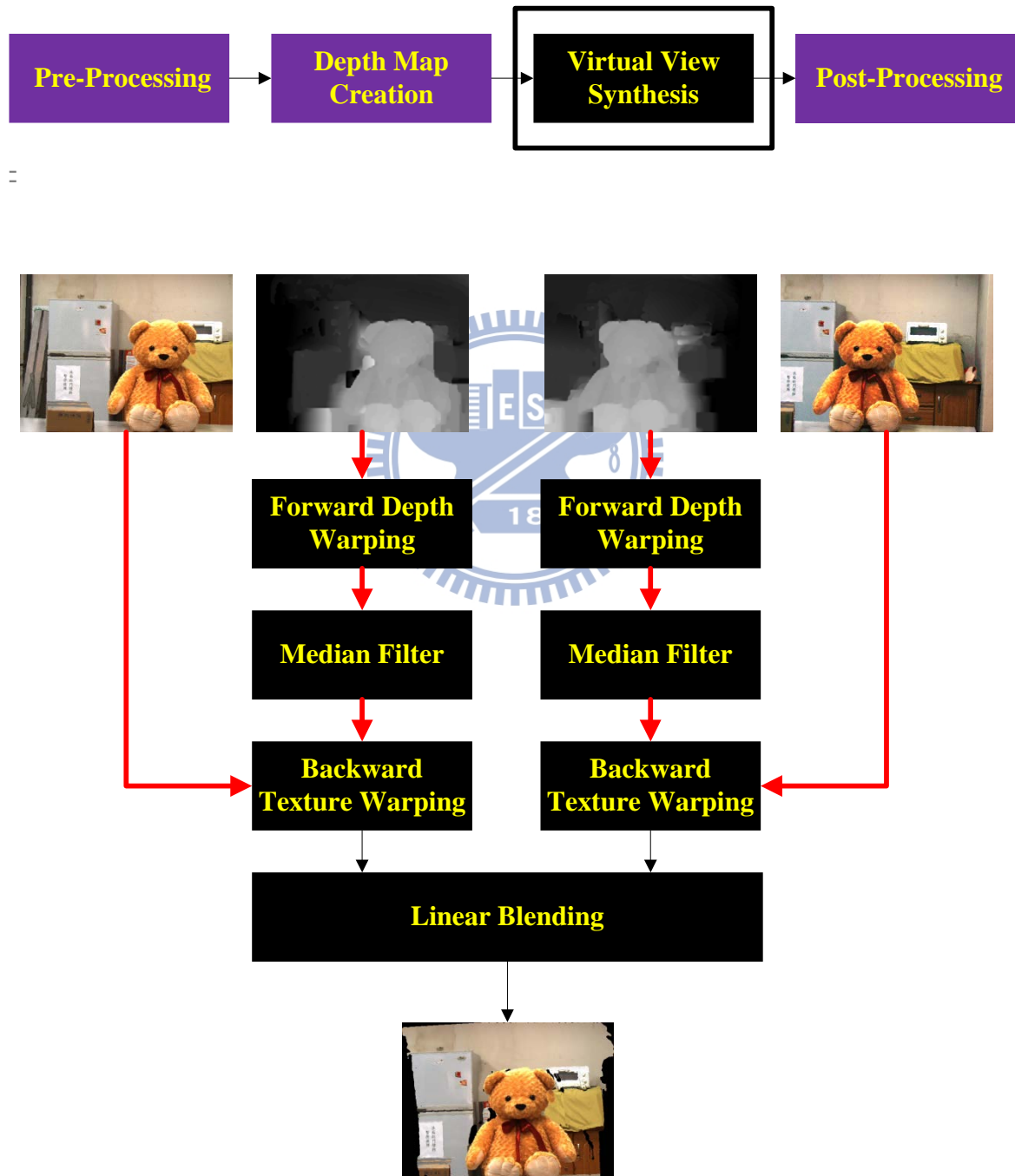


圖 37 虛擬視點系統流程圖 (Virtual View Synthesis)

虛擬視點合成是整個合成系統的重頭戲。在傳統的視點合成演算法上，會先對景深圖作平行**正向投影**至虛擬視點上。由於投影完之後，會有許多不合理的小破洞，需要將這些破洞修補起來。目前現有的視點合成參考軟體 (View Synthesis Reference Software，簡稱 VSRS)，它是採用**中值濾波器**，以 3×3 大小的核心 (Kernel)將這些小破洞修補起來。若是破洞太大，可能需要連續幾次的中值濾波效果才行。

有了虛擬視點的景深圖，再利用**逆向紋理映射**，從虛擬視點上的任意位置，回到參考影像上擷取對應的紋理。最後再將兩張虛擬視點影像，配合遮蔽區域做合適的**線性疊合**。

4.1.4 事後處理 (Post-Processing)

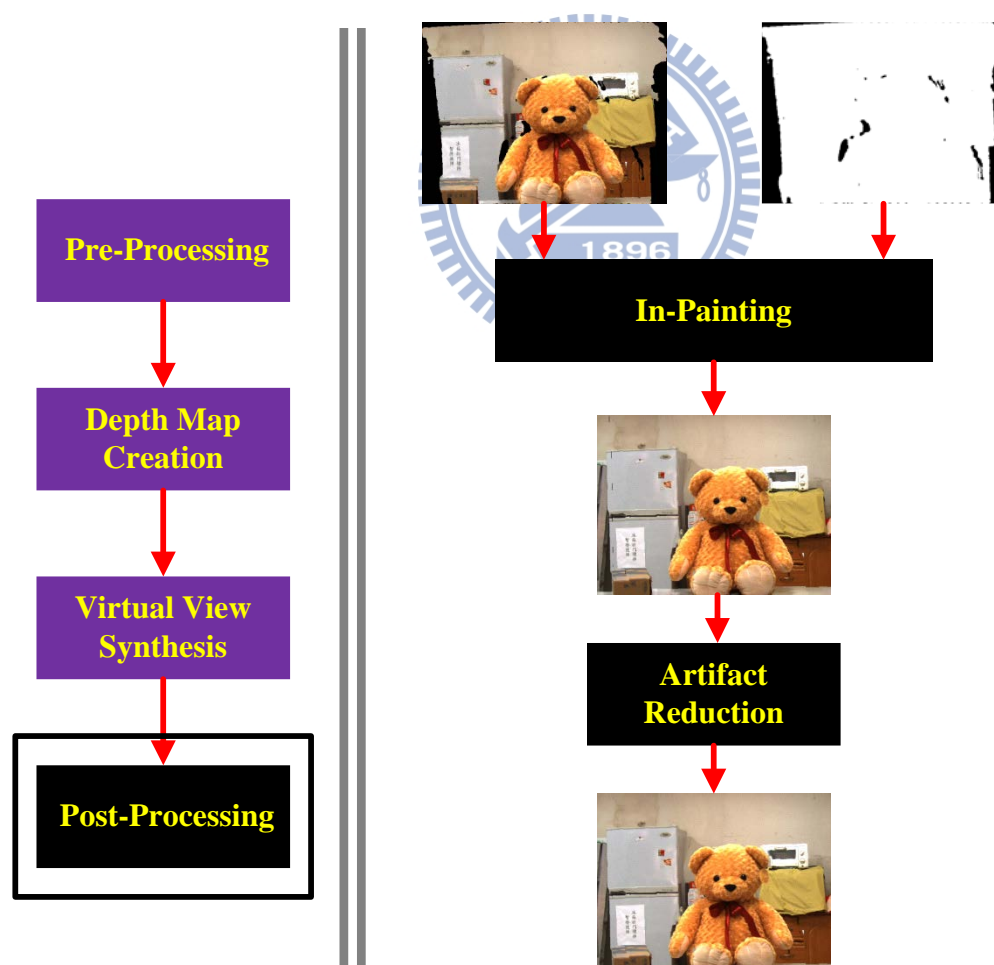


圖 38 事後處理系統流程圖 (Post-Processing)

事後處理的目的在於讓合成後的品質可以更加自然。一開始會先根據遮蔽區域做補洞的動作，目前最常見的手法是採用**影像修補技術 (In-Painting)**。接著再利用一些影像處理的技巧，讓合成影像中的一些瑕疵現象可以降低，不過此一處理必要性不高；即使有瑕疵現象，目前為止也沒有一個標準方法來做**瑕疵抑制**。本篇論文在後面會提供一個簡單的方式來做後置的瑕疵抑制。

4.2 稀疏視點合成系統架構 (System Flow Chart of Sparse View Synthesis)

傳統的自由式點合成系統只考慮到平行且密集成系統，因此我們嘗試設計出稀疏且非平行的自由視點合成系統架構。如圖 40 所示。根據傳統的自由視點合成系統中，探討對於稀疏影像合成過程會有何影響，並且針對一些地方加以改進。圖 40 中，方角矩形的區塊代表現有已經存在的方法；而圓角矩形+陰影區塊則代表本篇論文所提出的各種方法。

首先發現景深圖在做升取樣的時候，使用一般所認知比較好的濾波器，對於景深圖而言不一定適用，因此我們提出一個更簡單、但是用在景深圖上效果卻比較好的**複本插值法 (Duplicated Interpolation)**。詳細情況會在 4.3 章節加以探討。

第二個則是本篇論文的重點。我們提供了**景深圖映射方案 (Depth Warping Scheme)**。如圖 39 所示。若映射方向為水平，則分別使用正向映射、加上 3.2 章節所提出的景深圖修復演算法，將量化誤差修補掉；若映射方向為廣義，則我們提出一套**逆向景深映射演算法 (Backward Depth Warping Algorithm)**，再用**適應性中值濾波器 (Adaptive Median Filter)** 簡單的把剩下的量化誤差修復。詳細流程會在 4.4 章節介紹。

接著考量到真實影像中的色彩沒有一致性，即使前面做過簡單的色彩校正，但並非是完全相同的區域做校正。因此做完逆向紋理映射後，我們可擷取出哪些區域是共通

的，並且對該區域設計等化器來做色彩校正，以確保後面的線性疊合不會發生色彩偏差而有區塊形狀的瑕疵現象產生。而在傳統的合成系統中並未考慮到這一點，主因是它們所採用的影像都是以標準測試照片做為研究上的指標。因此為了真實上的合理性，我們就直接加上色彩校正而不再探討其前後差異。

至於事後處理中，傳統上會直接用背景延伸補洞、或是採用影像修補的技巧。但上述技術卻很容易受到雜訊影響使得補洞效果不夠強健。因此我們提出一套**金字塔型影像修補技術 (Pyramid-based Hole Filling, 簡稱 PBHF)**，讓補洞夠強健。詳細流程會在 4.4.3 章節介紹。最後我們提供一個簡單的**瑕疵抑制演算法 (Artifact Reduction, 簡稱 AR)**，讓因合成所造成的各種不可預期的瑕疵現象能夠減少許多。而詳細流程會在 4.6 章節介紹。

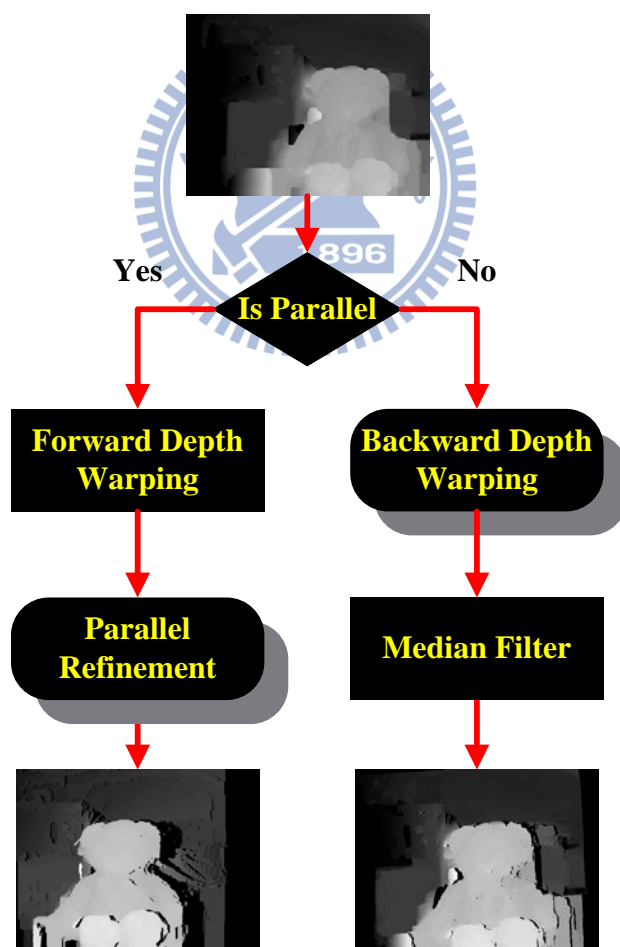


圖 39 景深圖映射方案 (Depth Warping Scheme)

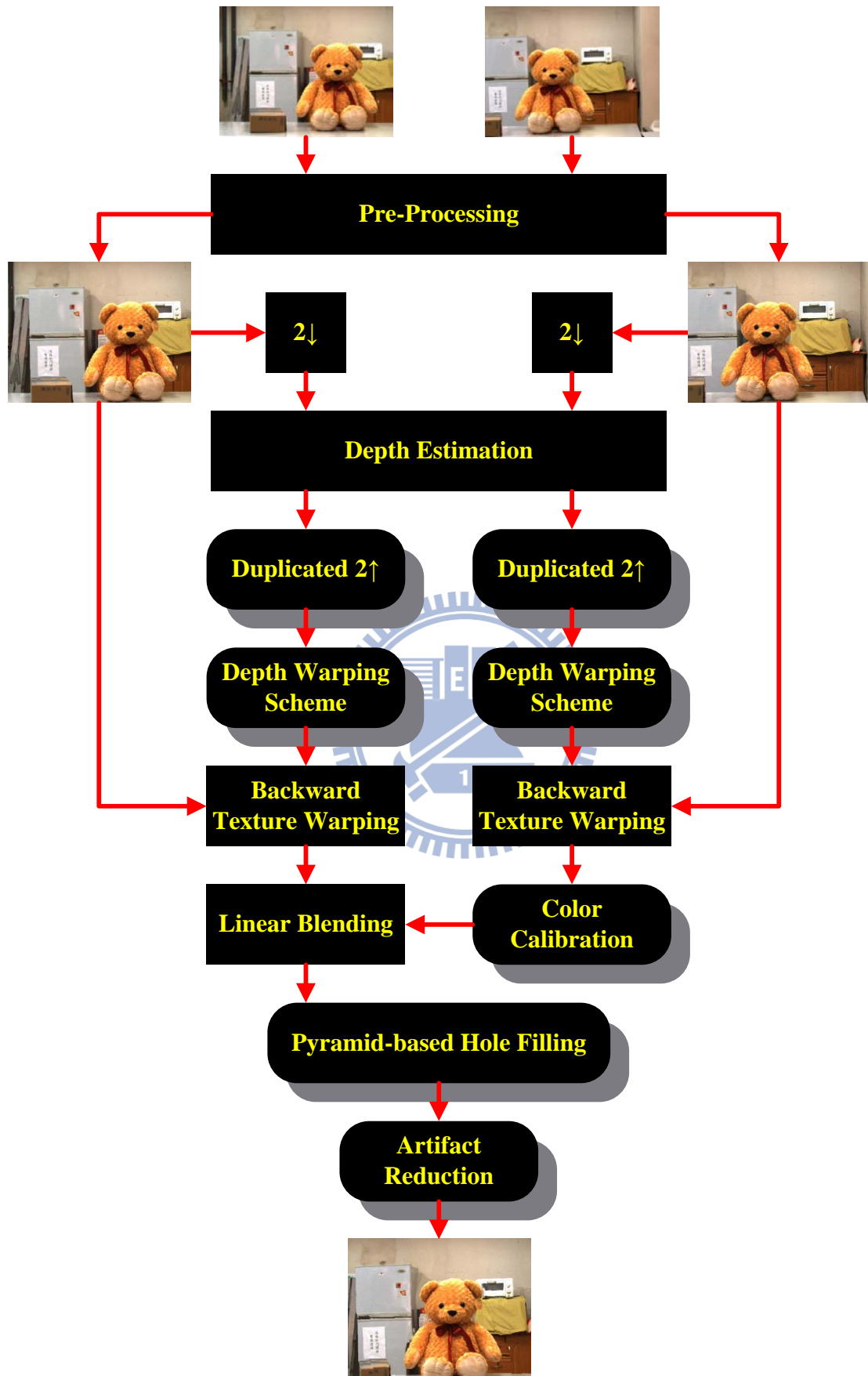


圖 40 稀疏視點合成系統架構圖

4.3 景深圖升取樣方法 (Up-Sampling Method in Depth Map)

4.3.1 各類升取樣方法介紹 (Introduction of Various Up-Sampling Methods)

升取樣在訊號處理的角度上是一種**插值 (Interpolation)** 的動作。而插值有很多種方法，常見的就是設計一個低通濾波器，亦即考慮一個核心函數 (Kernel Function) $h(x, y)$ ，對二維影像 $I(x, y)$ 做摺積 (Convolution)，如 (51)式所示：

$$I_h(x, y) = I(x, y) \otimes h(x, y) = \sum_i \sum_j I(i, j) \cdot h(x-i, y-j) \quad (51)$$

其中 (i, j) 是離散座標，而 $(x, y) \in \mathbb{R}^2$ 。我們只需依照比例直接對 (51)式中的 $I_h(x, y)$ 取函數值即可完成插值動作。為了計算方便，會假設 $h(x, y) = h_x(x) \cdot h_y(y)$ 為一可分離變數的函數，甚至 $h(x, y) = h(x) \cdot h(y)$ (兩分離變數函數型態相同)。有了這一層關係，我們就可定義各種一維低通濾波器。例如高斯核心，如(52)式所示：

$$h_{\text{gauss}}(x; \mu, \sigma) = \text{const} \cdot e^{-\frac{(x-\mu)^2}{\sigma^2}} \quad (52)$$

高斯插值法 (Gaussian Interpolation)，以高斯模型為核心函數) 在影像處理上是很常見的手段，並且也有相關的快速演算法做加速，缺點在於容易將影像模糊化。而在插值法上，表現較好的就屬 **樣條插值法 (Spline Interpolation)**、或是 **雙三次插值法 (Bi-cubic Interpolation)**。在此只介紹雙三次插值法，它考慮周圍共 16 個鄰近像素點，並且將像素點間的交互關係也納入考量，其核心函數為如 (53)式所示：

$$h_{bicubic}(x; a) = \begin{cases} (a+2) \cdot |x|^3 - (a+3) \cdot |x|^2 + 1 & \text{if } |x| \leq 1 \\ a \cdot |x|^3 - 5a \cdot |x|^2 + 8a \cdot |x| - 4a & \text{if } 1 < |x| \leq 2 \\ 0 & \text{otherwise} \end{cases} \quad (53)$$

其中 a 為常數，一般設定成 -0.5 。(53)式乍看之下很複雜，實際上它的函數型態和高斯核心相似，差別在於當 $1 < |x| \leq 2$ ， $h_{bicubic}(x; a = -0.5)$ 為一微小負數量值。雙三次插值法本身會讓有影像銳利化的效果，因此升取樣後的影像，其肉眼看起來不會有模糊的效果。但缺點是計算量較大。

除了上述所介紹的兩個插值法外，本篇論文還會介紹一個更簡單的方法，叫做**複本插值法 (Duplicated Interpolation)**。該方法跟最近鄰近插值法 (Nearest Neighbor Interpolation) 差不多，只是對於兩倍升取樣的情況下，複本插值法是直接將原始的影像備份四次，其核心函數為：

$$h_{dup, 2\uparrow}(x) = \begin{cases} 1 & \text{if } 0 \leq x \leq 0.5 \\ 0 & \text{otherwise} \end{cases} \quad (54)$$

若把 (54)式套回(51)式，可簡化為：

$$I_{2\uparrow}(x, y) = I_h(x/2, y/2) = I(x, y) \otimes \begin{bmatrix} 1 & 1 \\ 1 & 1 \end{bmatrix} \quad (55)$$

其中 \otimes 代表克羅內克積 (Kronecker Product)。複本插值法的運算量少，缺點在於容易產生方塊狀的瑕疵現象。

4.3.2 升取樣對映射後的景深圖影響 (Effects after Depth Warping)

前一小節介紹了三種插值方法，其中以雙三次插值法最佳，但僅對有紋理的影像而言。若針對景深圖影像，則效果會不同！經由實驗發現，**景深圖使用複本插值法，效果會比使用高斯或雙三次插值法還要來的好**，主因是景深圖是用來記錄影像內容物在空間中的相對位置，並非記錄其影像紋理，因此物體邊緣的景深資訊會非常重要。若我們使用一個低通濾波器在景深圖上，很容易會把邊緣資訊破壞掉，造成景深圖映射後產生無法預期的瑕疵現象。圖 41 為原始景深圖做完映射後結果、以及添加各種升取樣方法後的景深圖再做完映射後的結果。



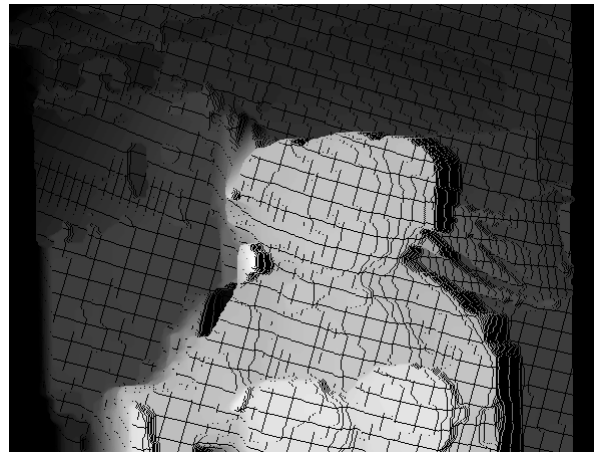
(a) 參考影像景深圖



(e) 參考影像景深圖映射之結果



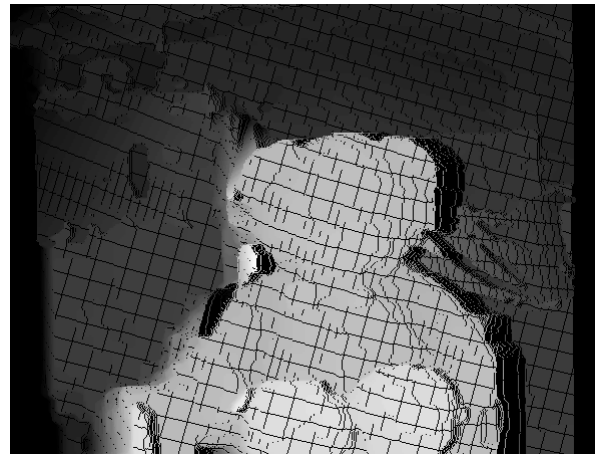
(b) 參考景深圖經高斯插值法之結果



(f) 高斯插值法後景深圖映射之結果



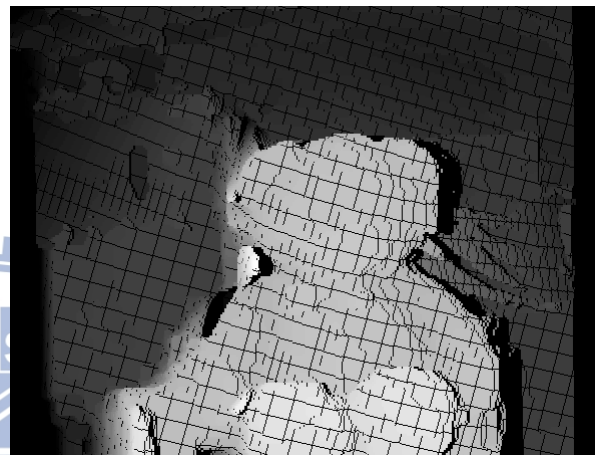
(c) 參考景深圖經雙三次插值法之結果



(g) 雙三次插值法後景深圖映射之結果



(d) 參考景深圖經複本插值法之結果



(h) 複本插值法後景深圖映射之結果

圖 41 各類升取樣方法對景深圖映射後的影響

由圖 41 可發現，在做完升取樣後，除了 (c) 圖略為模糊外，基本上看不太出來有何差異。可是當做完映射後，不論是高斯插值法 (d) 圖、或是雙三次插值法 (f) 圖，其映射後的景深圖邊緣會產生類似鬼影現象的線條狀雜訊。尤其是雙三次插值法 (f) 圖，邊緣更會有類似椒鹽雜訊 (Pepper and Salt Noise) 的瑕疵現象。反觀由複本插值法 (h) 圖所做出來的景深圖映射，邊緣並不會有鬼影現象、也不會有椒鹽雜訊產生，跟原始圖相比只是量化誤差變多。其局部放大示意圖可詳見圖 43。

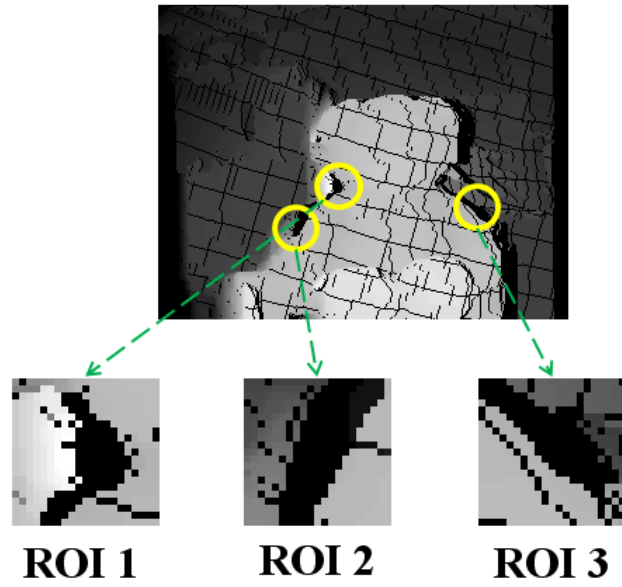


圖 42 景深圖映射後 ROI 編號位置示意圖

	Gaussian	Bi-cubic	Duplicated	None
ROI 1				
ROI 2				
ROI 3				

圖 43 各類升取樣方法對景深圖映射後的放大部分示意圖

4.4 逆向景深圖映射演算法 (Backward Depth Warping Algorithm)

4.4.1 平面下的單應映射 (Homography Induced by a plane)

在影像上若有三個相異點共線，則經由一些轉換至另外一個平面後，該三個點也會維持共線的性質，則稱該轉換為單應映射 (Homography)。經由證明可知單應映射其實就是 2.1.1 小節介紹的投影矩陣中所構成的轉換。因此假設兩影像的對應點分別為 (u_1, v_1) 、 (u_2, v_2) ，則其之間的單應映射 3×3 轉換矩陣 H 會滿足：

$$\begin{bmatrix} u_2 \\ v_2 \\ 1 \end{bmatrix} \sim H \begin{bmatrix} u_1 \\ v_1 \\ 1 \end{bmatrix} \quad (56)$$

若將這些對應點投影回世界座標中，會發現三維度的點集合可構成一個平面，如圖 44 所示。換言之，若世界座標中存在一個平面，分別投影至兩相機影像中，則其對應的影像會存在單應映射的關係。我們把這個性質稱做平面下的單應映射 (Homography Induced by a plane，簡稱 HIP)。

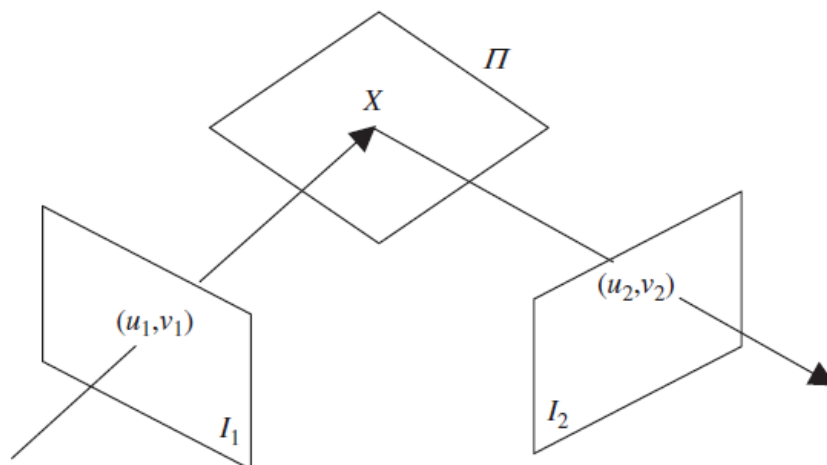


圖 44 平面下的單應映射 (HIP) 示意圖 [17]

根據該特性，不難預知單應映射會跟對應的平面方程式有關。若假設平面方程式在世界座標為 $n^T X = d$ ，且 I_1 與 I_2 兩個相機參數分別是 P_1 與 P_2 ，則可推導出其轉換矩陣會是：

$$H = P_2 \begin{bmatrix} P_1 \\ n^T | d \end{bmatrix}^{-1} \begin{bmatrix} I_{3 \times 3} \\ \bar{0} \end{bmatrix} \quad (57)$$

平面下的單應映射在電腦視覺領域上扮演著一個很重要的角色，例如 2.2 章節所提到的相機校正，就是利用 HIP 的特性簡化模型。

4.4.2 256 張平面下的逆向景深圖映射 (Backward Depth Warping Induced by 256 planes)

在 3.3 章節中有略提一下廣義映射下的誤差來源，但卻因其量化誤差可能性太多種類，以至於沒有一個好方法可以有效的解決。但後來我們有觀察到因為景深圖有做過量化，若量化階層為 N ，則可想像成一張被量化的景深圖其實是由空間中的 N 個不同深度的平面所構成，並且與相機影像平行。根據這個觀察，我們即可實現出量化下的逆向景深圖映射演算法。

首先將一張景深圖根據 N 種不同的量化景深值，分別擷取出 N 個平面。接著利用 (32) 式，將參考景深資訊用 (14) 式取代，並加以整理可得：

$$\begin{bmatrix} u_v \\ v_v \\ 1 \end{bmatrix} \sim H(z_q) \begin{bmatrix} u_r \\ v_r \\ 1 \end{bmatrix} \quad (58)$$

$$\text{其中 } H(z_q) = A + [0 \ 0 \ 1] \otimes \left(\frac{z_q}{N-1} \cdot (z_{near}^{-1} - z_{far}^{-1}) + z_{far}^{-1} \right) \cdot b \quad (59)$$

因此可利用 (59) 式，對每一個平面所屬的 $H(z_q)$ 計算其反矩陣 $H(z_q)^{-1}$ ，並且由虛擬位置逆向單應映射回參考景深圖。由於對不同平面的參考景深圖，可視為二值化的影像資訊，因此只需判斷參考景深圖有無資訊；若擷取到景深資訊，則計算出虛擬位置上的景深資訊。等逆向投影至虛擬位置中 N 張圖都計算完一遍後，再將之疊合；若同一個位置上發生至少一個景深資訊，則取深度值最小的景深資訊。

我們把上述這一套方法稱做 **N 張平面下的逆向景深圖映射演算法 (Backward Depth Warping Algorithm Induced by N planes)**，簡稱 **BDWIP-N**。圖 45 展示出 BDWIP-5 後的結果，並將每層的景深圖與映射後結果放在圖 46 中。

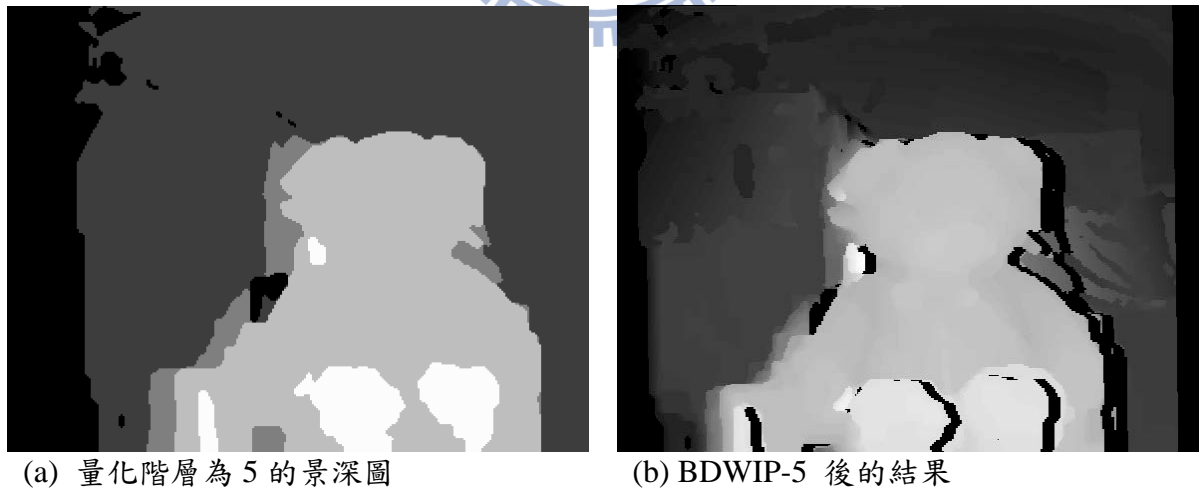


圖 45 量化階層 5 的景深圖與 BDWIP-5 的結果示意圖

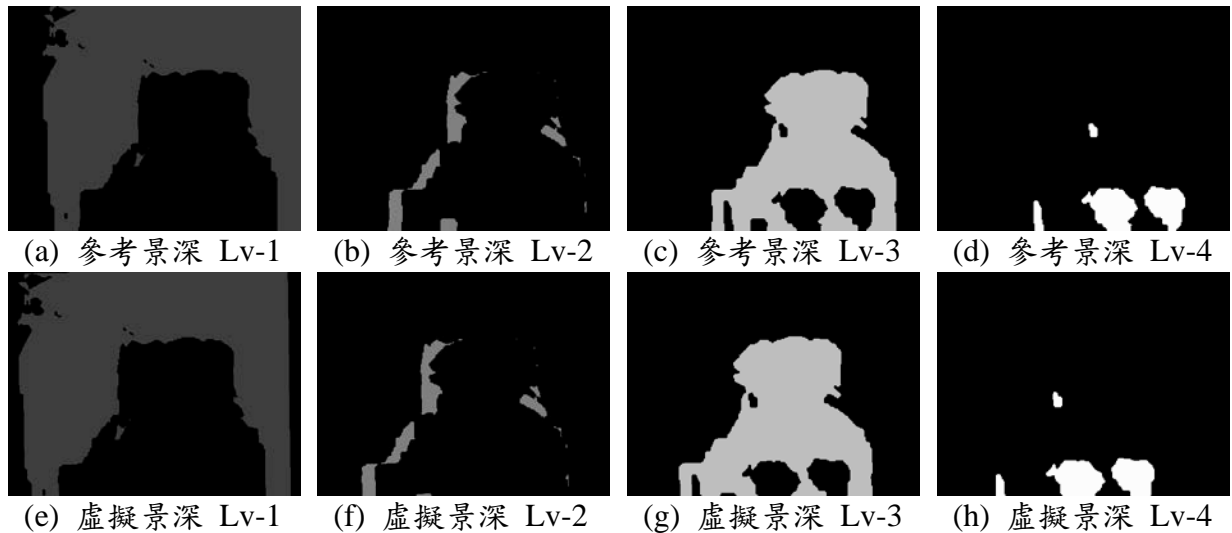


圖 46 BDWIP-5 各層結構圖 (黑色區塊代表不屬於此平面的區域)

值得注意的是，我們只需處理 $N-1$ 張二值化影像，因為對於量化景深為 0 的區域，可視為遮蔽或是未知區域而不需處理它。在真實傳送景深資訊圖中，量化階層一般為 256 層，這代表著我們一共需要做 255 次的單應映射，再將圖合併再一起才算大功告成。

雖然 BDWIP-N 計算量大，卻可以有效地降低量化誤差的產生，原因是在參考影像的景深圖中，若屬於**連續平坦** (簡稱 *CF*) 的區域，則逆向單應映射可以保證虛擬影像的景深圖也保留其連續特性。而且對一張參考景深圖中，大部分的區域都是屬於連續平坦，因此 BDWIP-N 可以有效地降低量化誤差的生成。

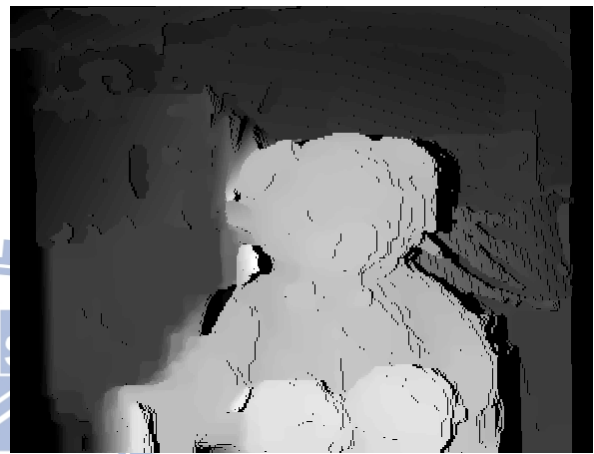
即便如此，BDWIP-N 還是沒辦法完全解決映射後的量化誤差，主因是該演算法並沒有考慮到參考景深圖的**連續遞增** (簡稱 *MI*) 或**連續遞減** (簡稱 *MD*) 區域，所以映射完後還是會殘留一些因上述兩個連續區域所造成的量化誤差。可參考圖 47。



(a) 參考影像景深圖



(b) 正向景深映射演算法結果



(c) BDWIP-256 結果

圖 47 正/逆向景深映射演算法比較

4.4.3 使用適應性中值濾波器消除殘餘的量化誤差 (Survival Quantization Error Reduction using Adaptive Median Filter)

前一小節中介紹一套 BDWIP 演算法，可使景深圖映射後大量減低量化誤差，但剩下的量化誤差尚需有一個方法來做修復。目前現有的視點合成參考軟體是採用中值濾波器來將量化誤差修復掉，缺點是容易讓景深圖中的物體邊緣景深被破壞掉。因此本篇論文採用適應性中值濾波器 (Adaptive Median Filter)[36]，讓邊緣被破壞的可能性再降低一些。

在介紹適應性中值濾波器前，先介紹一下中值濾波器的原理。它是直接取一個區塊的中間值來取代現有的數值。如 (60)式所示。這樣做的目的可以把極高頻段的區域（或是脈衝訊號），用相對低頻段區域的數值取代掉，因此很適合用來濾掉椒鹽雜訊 (Pepper and Salt Noise)。

$$\text{MedianFilter}\{I\}(x, y) = \underset{(u,v) \in S_n(x,y)}{\text{median}} \{I(u, v)\} \quad (60)$$

$$S_n(x, y) = \{(x+u, y+v) \mid -n \leq u, v \leq n\} \quad (61)$$

其中 n 跟核心尺寸 (Kernel Size) 有關，一般會取 $n=1$ ，即以核心大小為 3×3 當作濾波器的尺寸。對於密集平行排列的相機視點合成系統而言，景深圖映射後的量化誤差一般使用 3×3 大小的中值濾波器已經足夠；可是對於稀疏陣列而言， 3×3 大小的濾波器會不太夠。但若使用太大尺寸的核心，物體邊緣的景深資訊更容易遭受到破壞。此外，中值濾波器對於非雜訊的數值，也可能會用不同的數值取代之，造成影像有平滑化的傾向。而適應性中值濾波器會考慮到以上兩點，除了判斷哪些高頻雜訊要用多大的核心尺寸來操作，也可以防止非雜訊的數值被濾波器影響到。

接著簡介適應性中值濾波器的演算法流程。首先假設灰階影像 $I(x, y)$ 在區域 $S_n(x, y)$ 上的色彩強度最大值、最小值、與中間值分別為 $I_{\max}(x, y, n)$ 、 $I_{\min}(x, y, n)$ 、和 $I_{\text{med}}(x, y, n)$ ，而後判斷 (62)式有沒有成立。若不成立的話，代表 $I_{\text{med}}(x, y, n)$ 對於 (x, y) 位置上是脈衝訊號，則讓核心尺寸放大後重新判斷，亦即接續考慮 $S_{n+1}(x, y)$ 區域中，(62)式有沒有成立。直到 (62)式不成立時，代表 $I_{\text{med}}(x, y, n)$ 並非脈衝訊號，我們改判斷 (63)式會不會成立，決策出 $I(x, y)$ 是不是屬於脈衝訊號。若是的話（代表(63)式不成立），則將影像強度值 $I(x, y)$ 用 $I_{\text{med}}(x, y, n)$ 取代，否則不去更動原始影像強度值 $I(x, y)$ 。

$$I_{\min}(x, y, n) < I_{\text{med}}(x, y, n) < I_{\max}(x, y, n) \quad (62)$$

$$I_{\min}(x, y, n) < I(x, y) < I_{\max}(x, y, n) \quad (63)$$

綜合上述，適應性中值濾波器可寫成如 (64)式所示：

$$AMedianFilter\{I\}(x, y) = \begin{cases} I_{med}(x, y, n) & \text{if } I_{min} < I_{med} < I_{max} \\ & \text{and } I(x, y) \in \{I_{min}, I_{max}\} \\ I(x, y) & \text{otherwise} \end{cases} \quad (64)$$

for $n = \arg \min_k \{k \mid 1 \leq k \leq n_{max}\}$

其中 n_{max} 用來限制核心最大尺寸。圖 48-(a) 為原始影像受到椒鹽雜訊影響後示意圖。而圖 48-(b) 與 (c) 則是分別使用 3×3 與 9×9 核心尺寸大小的中值濾波器後的結果。可看出當中值濾波器的尺寸越大，濾完後的影像會越模糊。但使用 9×9 大小的適應性中值濾波器後，影像相較起來比較清晰，如圖 48-(d) 所示。

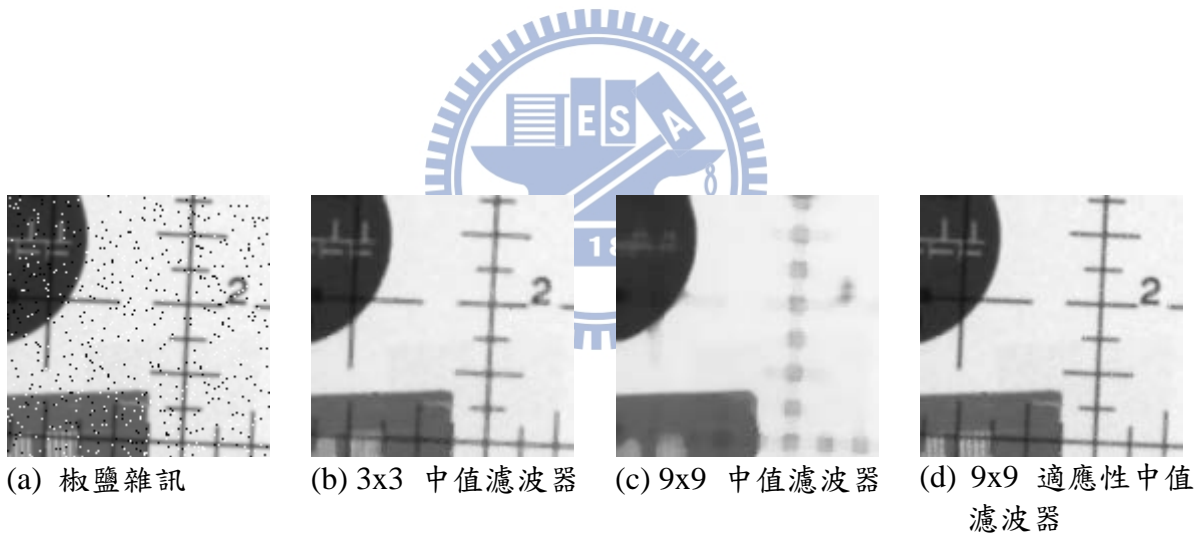


圖 48 添加雜訊與使用(適應性)中值濾波器後示意圖

若我們將(適應性)中值濾波器用在景深圖上，如圖 49 所示。可看出對於少量的量化誤差， 3×3 的中值濾波器 與 9×9 適應性中值濾波器的表現較佳，而 9×9 的中值濾波器會讓景深圖過度平滑化，亦即物體邊緣的景深資訊被破壞得更嚴重。此外也會發現使用適應性中值濾波器結果後，景深圖還是會殘留一些量化誤差的區域，這代表著它不會隨意就將景深破壞掉，並且剩下來的量化誤差絕大多數都是型態 I，可在後續的合成系統上視為極小塊的遮蔽區域來處理它，因此對於最後的合成品質不會有太顯著的影響。

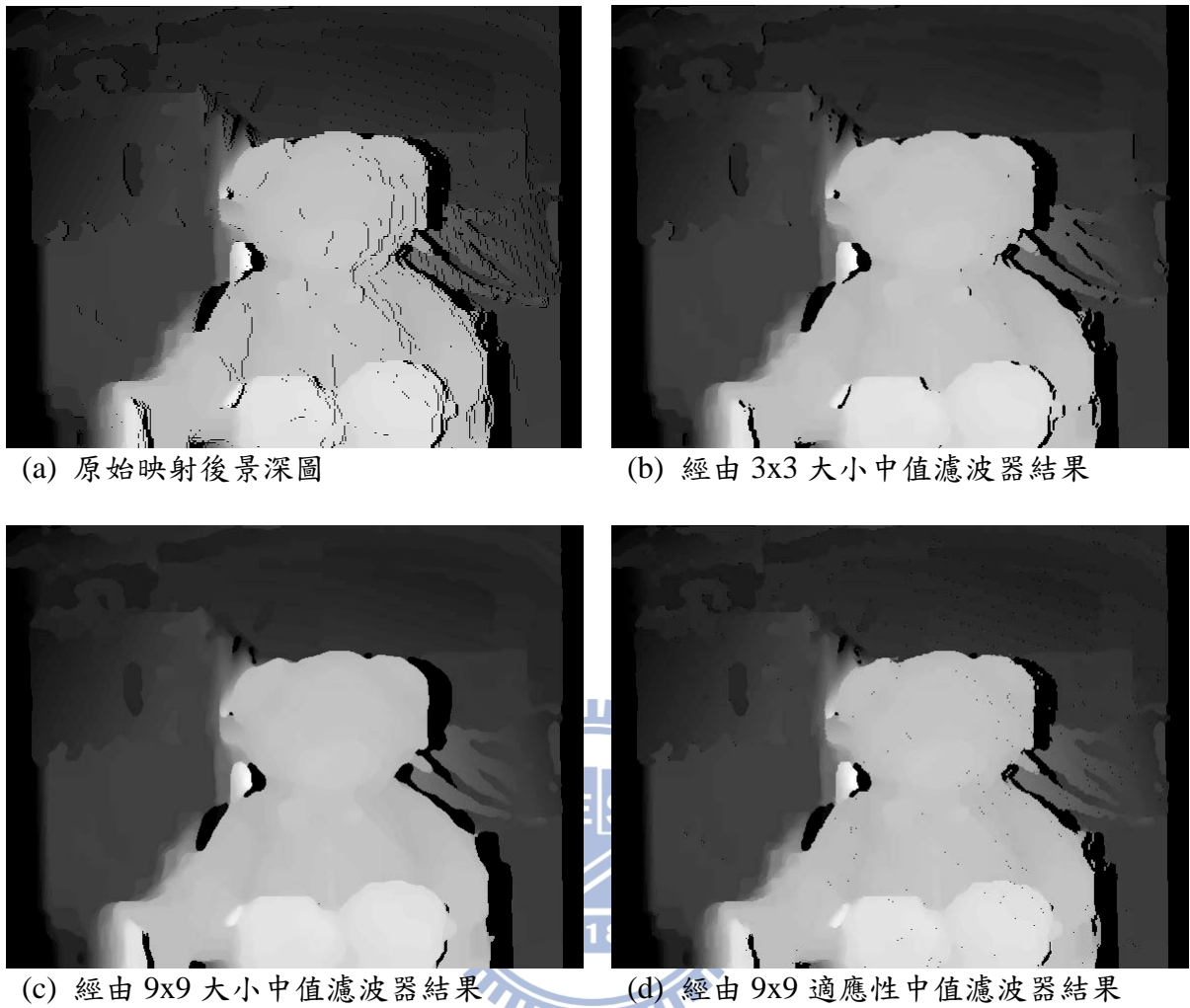


圖 49 景深圖經由(適應性)中值濾波器結果

若想了解關於適應性中值濾波器的其它細節，可參照數位影像處理方面的書[36]。

4.5 金字塔型影像修補技術 (Pyramid-based Hole Filling)

影像修補技術在 2.8 章節曾介紹其目的與原理，一般也被廣泛應用於修補合成技術中的遮蔽區域。可惜影像修補有個很大的問題，那就是非常容易受到雜訊的影響，尤其是破洞邊緣處，主因是影像修補演算法過度依賴破洞邊緣與鄰域的影像資訊，因此需確

保在做影像修補的時候，其破洞邊緣處資訊是完全可以信賴的。但對於影像合成技術來說，非常難做到擷取出完美的遮蔽區域等破洞，難免會有些微的雜訊產生；尤其對稀疏合成系統來說，遮蔽區域遠大/多於傳統式點合成所處理的標準測試照片。因此需要有一套同時顧慮到雜訊的補洞機制。

最直接的方式，就是對合成影像做降取樣。如圖 50 與圖 51 所示：

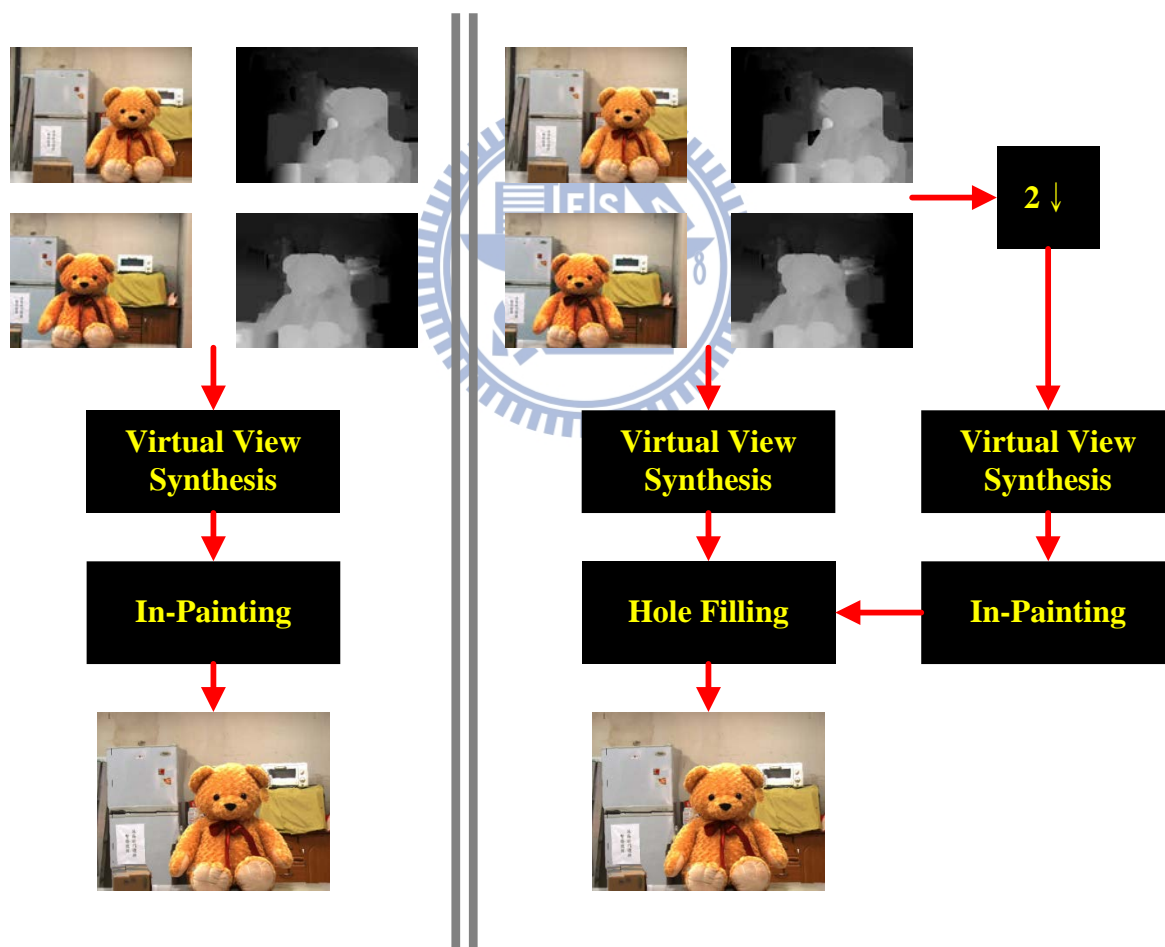


圖 50 PBHF_level_0 (左邊) 與 PBHF_level_1 (右邊) 示意圖

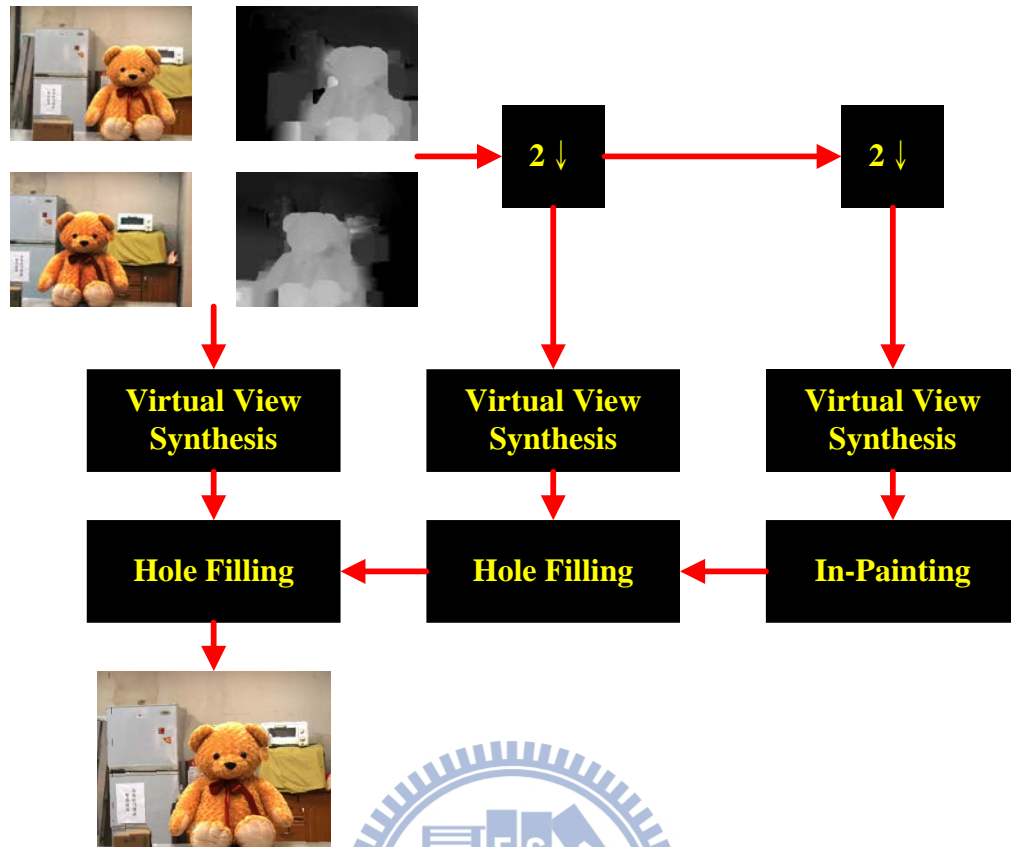


圖 51 PBHF_level_2 示意圖

我們觀察到對於降取樣後的影像，雖然解析度變低，但肉眼也越難觀察出圖片有無受雜訊影響。據此設計了一套稱做**金字塔型影像修補技術 (Pyramid-based Hole Filling, 簡稱 PBHF)**。它的想法類似影像金字塔 (Image Pyramids) 的建構方式，也就是先將參考影像與對應的景深圖，做若干次的降取樣後，再實行合成動作。由於降取樣後的合成影像，雜訊也會被壓抑住 (因解析度變低)，此時再利用現有的影像修補技術來做補洞的動作。

有了最底層的破洞資訊，後續在做升取樣的動作，直接填補上一層合成完的影像；直到填補最初合成大小的影像，即完成金字塔型影像修補法。根據降取樣的多寡次數，我們定義 PBHF 的階層 N (簡稱 **PBHF_level_N**)，代表做完 $(N-1)$ 次的降取樣後的影像資訊。因此若 PBHF 階層數分別為 1 或 2，就代表著 圖 50(右) 與 圖 51，分別做 1 次與 2 次的降取樣。而 PBHF 階層數 0 則如圖 50(左)所示，為一般自由式點合成的影像修補法。

以下做了一個簡單的實驗，將稀疏合成系統中，原本針對景深圖做複本插值法的方法，換成雙三次插值法，目的在於讓景深圖中的物體邊緣處產生不可預期的鬼影與椒鹽雜訊。如圖 52-(a) 所示。接著我們分別使用金字塔型影像修補法階層 0~2，並觀察其遮蔽區域的修補情況，如圖 52-(b)~(d) 所示。而圖 54 則是部分區域的放大示意圖。

由實驗圖中可以發現，對於階層為 0 的 PBHF，也就是一般直接用影像修補法補洞的流程，會因雜訊的關係，使補洞的時候讓雜訊擴大，例如玩具熊的右手臂會因為影像修補的關係，而直接往外延伸出不合理的紋理。至於 PBHF 階層為 1 或 2 的時候，因在低解析度下做影像合成與修補的動作，使得遮蔽區域有被平滑化的效果，但卻也壓低了雜訊的擴增。



(a) 原始合成圖



(b) PBHF_level_0 結果



(c) PBHF_level_1 結果



(d) PBHF_level_2 結果

圖 52 引進雙三次插值法至景深圖後產生雜訊，並用 PBHF 補洞示意圖

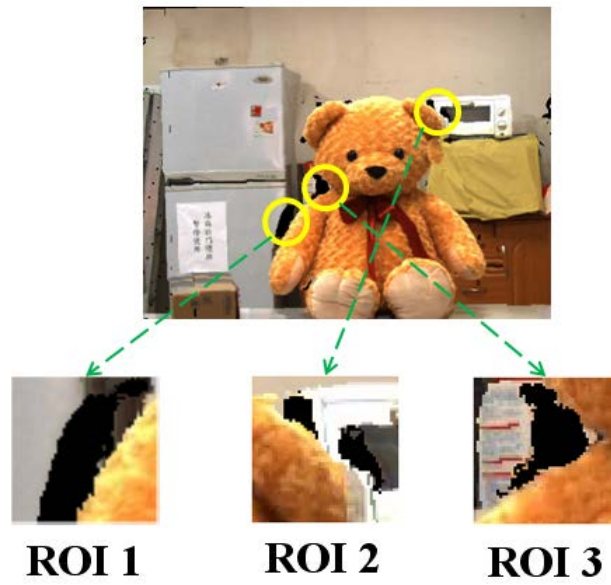


圖 53 金字塔影像修補前 ROI 編號位置示意圖

	(a) 原始合成圖	(b) PBHF_level_0	(c) PBHF_level_1	(d) PBHF_level_2
ROI 1				
ROI 2				
ROI 3				

圖 54 PBHF 補洞放大部分示意圖

4.6 瑕疵抑制演算法 (Artifact Reduction)

4.6.1 形態學影像處理簡介 (Introduction of Morphological Image Processing)

形態學影像處理 (Morphological Image Processing) 是一種用來分析且處理影像幾何結構的一門影像處理技巧，可用來擷取影像的特徵和一些屬性，它的原理來自於數學型態學 (Mathematical Morphology，簡稱 MM)。我們在此不會介紹太多相關的細節，只會簡介型態學影像處理上最基礎的兩種運算規則：擴張 (Dilation) 和侵蝕 (Erosion)；以及在形態學上的梯度 (Gradient)[36]。

在介紹運算規則前會定義一種東西，稱作結構元素 (Structure Element，簡稱 SE)。顧名思義是一種小集合或子影像，用來探討影像中有沒有出現類似該結構元素所組成的結構中，一般會以二維矩陣來表示。而在形態學上的任何運算，結構元素皆扮演著操作影像的角色，因此結構上的選取會很重要。

接著介紹最基本的兩個運算子，分別是擴張算子與侵蝕算子。簡言之，我們可利用自己所定義好的結構元素，分別對影像做出擴張與侵蝕的動作，一般在符號上定義成 \oplus (擴張算子) 與 \ominus (侵蝕算子)。若考慮一張灰階影像 $I(x, y)$ 、與感興趣的結構元素 $S(x, y)$ ，則擴張與侵蝕分別定義為如下 (65)式與 (66)式：

$$(I \oplus S)(x, y) = \max_{(u,v) \in D\{S\}} \{I(x-u, y-v) + S(u, v)\} \quad (65)$$

$$(I \ominus S)(x, y) = \min_{(u,v) \in D\{S\}} \{I(x-u, y-v) - S(u, v)\} \quad (66)$$

其中 $D\{S\}$ 代表 $S(x, y)$ 的定義域。在影像處理中會經常定義結構元素是平坦 (Flat)，所謂平坦的結構元素是指 $S(x, y) = 0$ ， $\forall (x, y) \in D\{S\}$ 。我們將平坦的結構元素符號寫為 $S_F(x, y)$ 。因此 (65)式和 (66)式可再度簡化為：

$$(I \oplus S_F)(x, y) = \max_{(u,v) \in D\{S_F\}} \{I(x-u, y-v)\} \quad (67)$$

$$(I \ominus S_F)(x, y) = \min_{(u,v) \in D\{S_F\}} \{I(x-u, y-v)\} \quad (68)$$

有了上述介紹的擴張與侵蝕，我們就能利用這兩個基本運算單元，操作出很多有趣的事情。例如定義灰階影像 $I(x, y)$ 在形態學上的梯度為如下(69)所示：

$$G\{I, S_F\} = I \oplus S_F - I \ominus S_F \quad (69)$$

一般而言， $S_F(x, y)$ 會定義成對稱且微小的結構來操作梯度運算。本篇論文是採用圓盤結構來實做梯度，在接下來的章節會用到。如圖 55 所示：

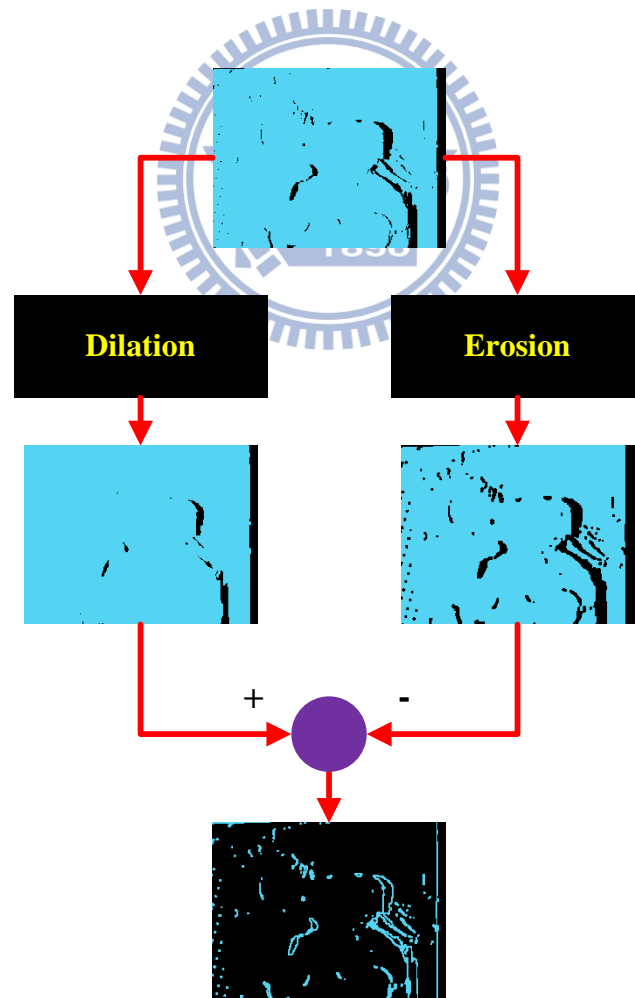


圖 55 形態學梯度演算法架構，以半徑 4 的圓盤大小為結構元素

4.6.2 可信賴性模型建構 (Reliability Model Construction)

本節最終目的在於抑制瑕疵現象，所以先判斷出哪些合成區域可能會發生瑕疵現象，會是一個很重要的起步與方向。可以利用與 3.2.1 小節相同的概念，根據瑕疵現象發生的特性，建構出可依賴模型，再將相對應的估計器用於瑕疵抑制。因此本小節會介紹如何建構出可依賴模型。

本篇論文也是以直觀的角度來建構可依賴模型。觀察到一張合成影像中，瑕疵現象最容易發生在遮蔽區域邊緣或輪廓處，因此最直接的方式，就是分辨出所有遮蔽區域的可能發生處，並且擷取其邊緣。利用前一小節所介紹的形態學梯度演算法可輕易的達到。如圖 56 所示：

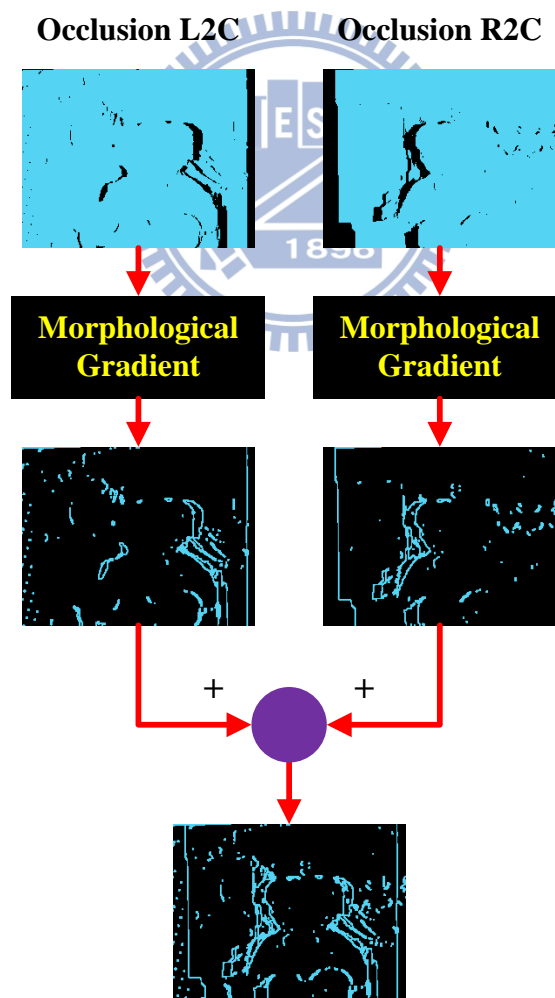


圖 56 AR Map 系統流程圖

圖 56 代表的意思是，先將左右兩參考影像映射至虛擬視點會發生的所有可能遮蔽區域都推得；再利用形態學的梯度算子，計算出遮蔽區域的輪廓，而梯度算子所用到的平坦結構元素，是半徑為 4 (以像素為單位) 的圓盤。我們不利用傳統擷取影像邊緣的演算法原因在於一般的邊緣偵測非常依賴梯度與紋理顏色變化，相對的對雜訊也很敏感，因此估算出來的輪廓可能會比利用形態學所估算出來的輪廓還要雜亂。最後將左右兩影像擷取出來的邊緣區域結合在一起，即完成整個可信賴模型的建構。

此外為了和 3.2.1 小節所建立的可信賴模型做區別，本小節所建立出來的二值化圖，我們稱作**瑕疵降低圖 (Artifact Reduction Map, 簡稱 AR Map)**。

4.6.3 使用中值濾波器來抑制瑕疵現象 (Artifact Reduction Using Median Filter)

利用前一小節所建構出來的可信賴模型後，接下來是想辦法如何將瑕疵現象設法移除掉。前面有稍微分析過，合成後的瑕疵主因為各種數值運算上的不準確性，導致可能產生區塊性的位移、或是不連續性誤差。因此最直接的想法是將這些瑕疵區域平滑化。我們只需針對 AR Map 中所擷取出來的邊緣區域將之平滑化，即完成瑕疵抑制方法。而本篇論文所採用的平滑化方法是**中值濾波器**。

雖然在 4.4.3 小節有稍微簡介中值濾波器的原理，以及適應性中值濾波器的強大效果，但在此並沒有採用適應性中值濾波器，主因是它只會針對椒鹽雜訊來做修復的動作，而對於真實參考影像合成而言，瑕疵現象往往是區域型的不確定性雜訊所形成，所以效果可能不會太強烈；反觀中值濾波器會使合成影像有平滑化的效果，對人眼視覺感受來說相對上較自然一些，因為人眼對於不自然區域的雜訊特別敏感。如圖 57 所示：



(a) 原始合成影像



(b) AR Map 建構



(c) 使用 3x3 適應性中值濾波器後結果



18 (d) 使用 3x3 中值濾波器後結果

圖 57 合成後瑕疵抑制方法與結果示意圖

由上圖可以看出對於物體邊緣部份，適應性中值濾波器並沒有將之修復的很完善，還是會殘留些微的黑色不自然區塊。但對中值濾波器而言，黑色區塊很明顯被壓低許多，代價是周圍附近的合成影像資訊有被模糊化的效果，例如玩具熊腳的邊緣處。圖 59 為其部分放大示意圖。

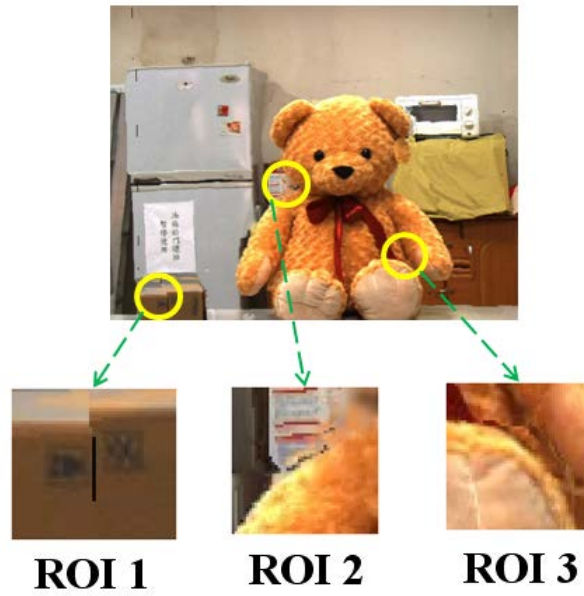


圖 58 瑕疵抑制演算法前 ROI 編號位置示意圖

	(a)原始合成影像	(b)AR Map	(c)適應性中值濾波器	(d)中值濾波器
ROI 1				
ROI 2				
ROI 3				

圖 59 合成後瑕疵抑制方法與結果部分放大示意圖

Chapter 5 實驗結果與討論 (Simulation Results and Discussions)

5.1 質量評估方法 (Quality Assessment)

在實驗過程中，為了要驗證最後的合成品質有改善、或是比目前現有的方法還要好，我們需要有一套評比標準來分析每種方法對於合成的好壞。雖然評比標準只是參考性質用，合成後的人眼主觀品質才是最重要，我們還是會介紹以下兩種在影像處理上常見的評分準則，分別是**峰值信噪比 (Peak Signal to Noise Ratio, 簡稱 PSNR)**、與 **歸一化交叉相關 (Normalized Cross Correlation, 簡稱 CORR)**。

因此本篇論文會以 PSNR 和 CORR 來做標準測試影像上的分析與探討。但對於真實影像的攝影與合成來說，我們只會用 CORR 來做評比，主因是不同位置與角度的影像而言，會有色差的問題，得到的 PSNR 數據意義不大。

5.1.1 峰值信噪比 (Peak Signal to Noise Ratio, or PSNR)

PSNR 是基於統計學上的均方根誤差 (Mean Square Error)，來對原始訊號與受到雜訊影響後的訊號中的功率做統計量的分析，它的含意是計算訊號的**最大功率與雜訊功率比值**。對大多數的訊號而言，其比值容易有很大的動態範圍 (Dynamic Range)，因此會以對數分貝刻度 (Logarithmic Decibel Scale) 表示之。如 (71)式所示：

$$MSE(I, \hat{I}) = \sum_{(x,y) \in ROI} [I(x,y) - \hat{I}(x,y)]^2 \quad (70)$$

$$PSNR(I, \hat{I}) = 10 \times \log_{10} \left[\frac{MAX^2(I)}{MSE(I, \hat{I})} \right] = 10 \times \log_{10} \left[\frac{65025}{MSE(I, \hat{I})} \right] \quad (71)$$

其中 $I(x, y)$ 與 $\hat{I}(x, y)$ 分別代表理想影像強度與合成後的影像強度。而 $MAX(I)$ 則代表影像像素強度的最大值，若針對 8 位元像素而言，其值會是 255。若對於彩色影像而言，我們會選取 $YCbCr$ 中的 Y 通道 (Luminance) 視為影像強度。一般而言，PSNR 在 30dB 以上，其合成品質對於人眼來說算還不錯，但很多時候還是容易看出較不合理的瑕疵區域。

5.1.2 歸一化交叉相關 (Normalized Cross Correlation, or CORR)

CORR 在統計學上是用來比較兩訊號的相關程度，對影像而言，其定義如下(72)式所示：

$$CORR(I, \hat{I}) = \frac{1}{N-1} \sum_{(x,y) \in ROI} \frac{[I(x, y) - mean(I)] \cdot [\hat{I}(x, y) - mean(\hat{I})]}{\sigma_I \cdot \sigma_{\hat{I}}} \quad (72)$$

$$mean(I) = \frac{1}{N} \sum_{(x,y) \in ROI} I(x, y) \quad (73)$$

$$\sigma_I = \sqrt{\frac{1}{N-1} \sum_{(x,y) \in ROI} [I(x, y) - mean(I)]^2} \quad (74)$$





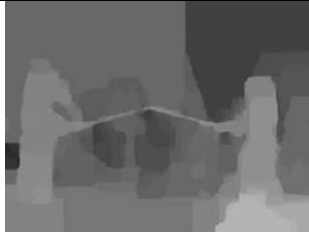
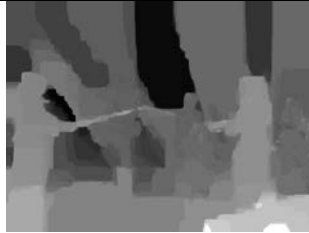
其中 $mean(I)$ 和 σ_I 分別代表影像強度 $I(x, y)$ 的平均值和樣本標準差，如(73)和(74)式所示、而 N 則是所關心的像素點個數。CORR 其實就是一般所熟知的**相關係數 (Correlation Coefficient)**，會用它來做量測指標是因標準測試影像通常會做一些半人工的處理，使得測試影像的變因降低許多，例如色差；但對於真實影像而言，因存在色差或是相機參數校正的不準確性等等，造成難以定義出一套好的質量評估指標。因此本篇論文採用相對上可降低色差差異性的 CORR 來量測真實影像的合成，即使 CORR 並非是很好的質量評估方法。

5.2 測試影像介紹 (Introduction of Test Sequences)

我們分別以四組真實測試影像與一組標準測試影像來做一系列的比較。真實測試照片是利用三台焦距大約是 1800 *pixel units* 的相機⁵來做實拍，如圖 60 所示。相鄰相機距離 15cm ~ 30cm 不等，而拍攝場景距離相機大約 3m ~ 6m。如表 6 所示。利用左邊與右邊參考影像資訊，設法合成出中間參考影像內容，並與之比較；不過由於色差的影響，我們利用 5.1 章節所介紹的 CORR 來做簡單的評比。值得一提，表 6 中的 *Test_02* 與 *Test_04*，其拍攝像機的旋轉角度較大，因此在合成上也會比較困難。

至於標準測試影像，我們是採用名古屋大學 (Nagoya University) 在研究自由式點合成系統上所提供的其中一組影像序列，用來測試平行合成系統中景深圖映射後的量化誤差修補對合成的影響。如表 5 所示。只是一般的標準測試影像對於相機而言屬於密集排列，與本篇論文的宗旨不合，在此並不會花很多篇幅於此討論上。且因它有做過一些人為的前置處理，因此可利用 PSNR 來做簡單的評比。

表 5 Kendo Sequence (Nagoya Univ.)

	左邊參考影像	中間參考影像	右邊參考影像
紋理圖			
景深圖			

⁵ 相機型號與規格為 M6Z 1212-3S, 12.5 ~ 75mm

表 6 真實測試影像











	左邊參考影像	中間參考影像	右邊參考影像
Test_01			
Test_02			
Test_03			
Test_04			



圖 60 實拍真實影像之相機陣列

5.3 環境設定 (Environmental Setting)

本篇論文結合目前現有的相機校正 Toolbox、景深估計參考軟體、虛擬視點合成系統、加上提出的各種演算法技術，在 Matlab 2010a 平台上發展成一套完整的稀疏虛擬視點合成軟體。首先做完相機校正後，需要人工給定幾何校正後的原點座標位置設定、校正後影像大小、以及景深估計的最大與最小搜尋範圍。如表 7 所示：

表 7 真實測試影像的前置規格表

Test Sequence	Spec	Documents
Test_01	bbox	[-13, 18, -13, 18]
	rsize	[448, 584]
	search_range	[30, 90, 30, 90]
Test_02	bbox	[92, 21, 62, 21]
	rsize	[456, 520]
	search_range	[40, 90, 40, 90]
Test_03	bbox	[-12, 22, -12, 22]
	rsize	[448, 608]
	search_range	[30, 80, 30, 80]
Test_04	bbox	[80, 24, 80, 24]
	rsize	[440, 600]
	search_range	[10, 85, 10, 85]

其中 $bbox = [\Delta c_{Lx}, \Delta c_{Ly}, \Delta c_{Rx}, \Delta c_{Ry}]$ 代表幾何校正完後，左右兩影像內部參數中的主軸影像點須分別平移 $(\Delta c_{Lx}, \Delta c_{Ly})$ 與 $(\Delta c_{Rx}, \Delta c_{Ry})$ 。 $rsize = [height, width]$ 代表幾何校正後的影像大小。而 $search_range = [\min_L, \max_L, \min_R, \max_R]$ 則代表右至左與左至右的景深圖估計中的最小與最大搜尋範圍；因為是對稱情況，一般會設定 $\min_L = \min_R$ 與 $\max_L = \max_R$ 。

除了給定好前置規格表外，本合成軟體由於採用多種不同的方法，因此需要給定每個步驟所採用的演算法為何，如表 8 所示：

表 8 合成參數示意表

Parameters	Documents
sampling_method	0: gaussian , 1: bicubic
depth_sampling_method	0: gaussian , 1: bicubic , 2: duplicate
DW_direct_mode	0: forward , 1: backward
DW_case_mode	0: general , 1: parallel
DW_refine_mode	0: median , 1: amedian , 2: PDR
TW_blending_mode	0: blending_off , 1: blending_on , 2: blending_left , 3: blending_right
Pyramid_level	0: level_0 , 1: level_1 , 2: level_2
AR_mode	0: off , 1: on
thread_cont	0 ~ 1% (only useful for setting DW_refine_mode = 2)
thread_dest	positive real number (only useful for setting DW_refine_mode = 2)

表中的 **sampling_method** 用來決定色彩影像的升/降取樣方法。**depth_sampling_method** 用來決定景深圖升取樣的方法。**DW_direct_mode** 可決定使用正/逆向景深映射。**DW_case_mode** 則是映射公式採用廣義或平行情況。**DW_refine_mode** 用來決定景深圖映射後的修補演算法。**TW_blending_mode** 用來決定逆向紋理映射後的線性疊合。**Pyramid_level** 是金字塔型影像修補技術所使用的階層數。**AR_mode** 則是決定要不要使用瑕疵抑制方法。最後的兩參數 **thread_cont** 與 **thread_dest** 是用於本篇論文所發展的量化誤差修補，只有在 **DW_refine_mode=2** 時才會用到。

本篇論文接下來會根據平行景深映射後修復演算法、景深圖升取樣方法、正/逆向景深映射演算法、金字塔型影像修補技術、與瑕疵抑制方法，設定適當的環境參數，並做進一步的實驗與探討。

5.4 平行景深圖映射後修復演算法對合成品質影響 (Parallel Warped Depth Refinement versus Quality of Synthetic Virtual Images)

5.4.1 環境設定 (Environmental Setting)

表 9 環境設定：平行景深圖映射後修復方法對合成品值影響

Parameters	Setting	Parameters	Setting
sampling_method	1	TW_blending_mode	1
depth_sampling_method	2	Pyramid_level	2
DW_direct_mode	0	AR_mode	0~1
DW_case_mode	1	thread_cont	1%
DW_refine_mode	0~2	thread_dest	10

表 9 中代表的意思是，我們以雙三次插值法做為色彩影像的升/降取樣方法；景深圖升取樣方法採用複本插值法；以平行正向映射演算法來做景深圖映射；紋理映射後的線性疊合是採用 2.7 章節所介紹的 (23)式；金字塔影像修補階層則採用 level 2。

接著我們分別以中值濾波器、適應性中值濾波器、以及本篇論文所提出的景深映射後修復演算法，用於映射後景深圖的量化誤差消除方法，來探討對於合成品質上的影響。至於景深圖修復的可信賴模型中，連續圖與下降圖的門檻值會以 1% 和 10 為主。此外，我們也會分別以有/無使用後製瑕疵抑制演算法來看對於合成品質的提高效率。

5.4.2 實驗結果 (Simulation Results)

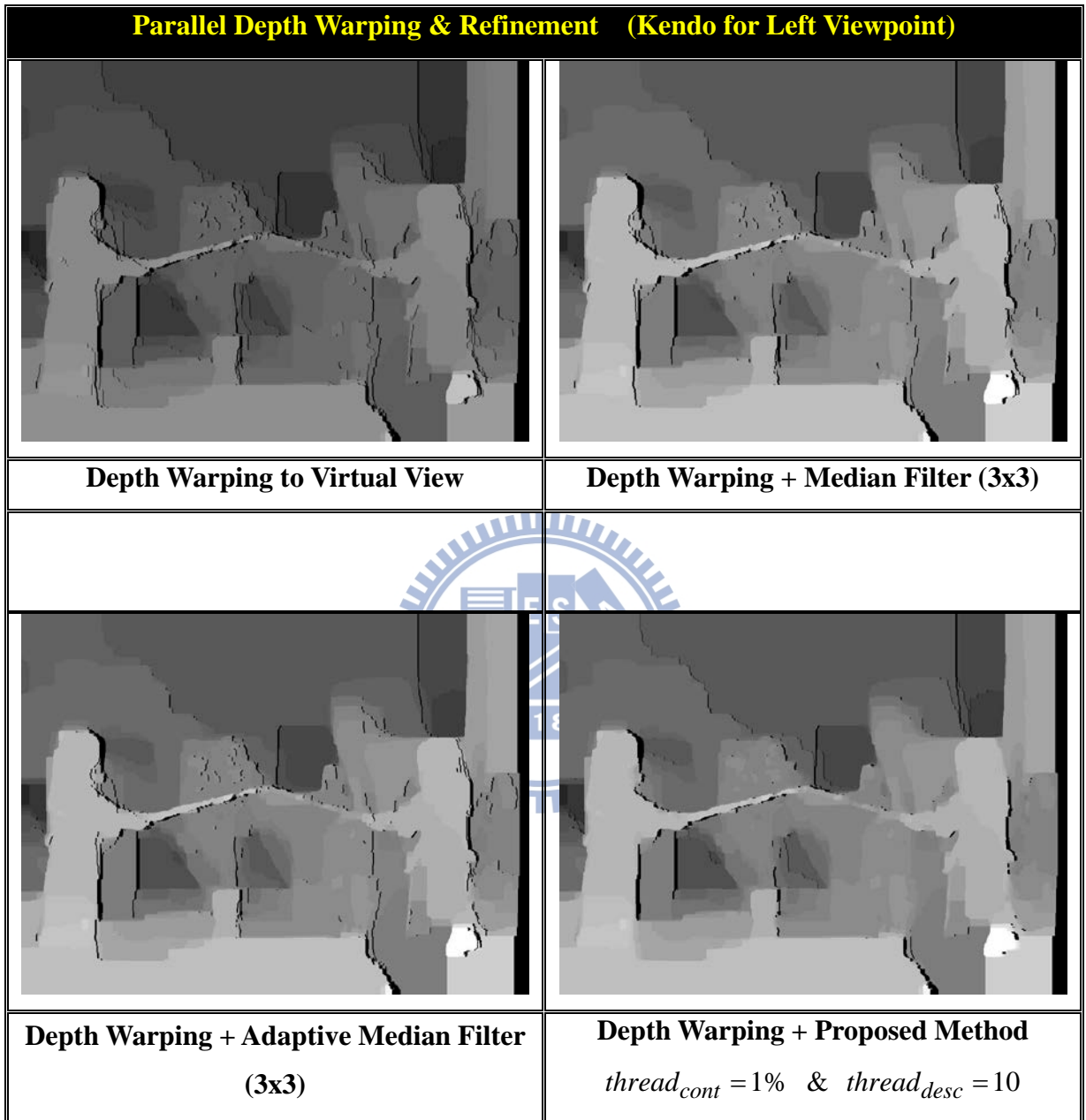


圖 61 左視點平行景深圖映射與針對不同修復演算法之示意圖 (Kendo Sequence)

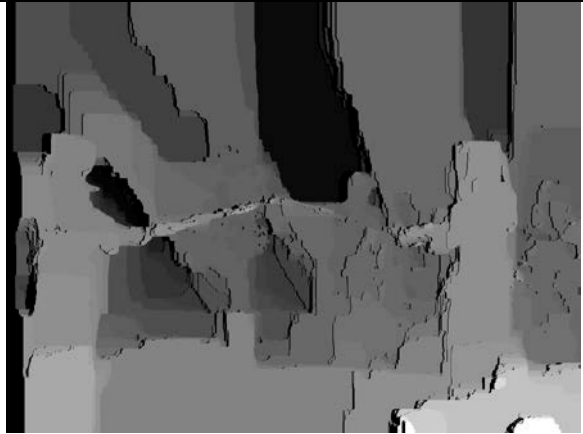
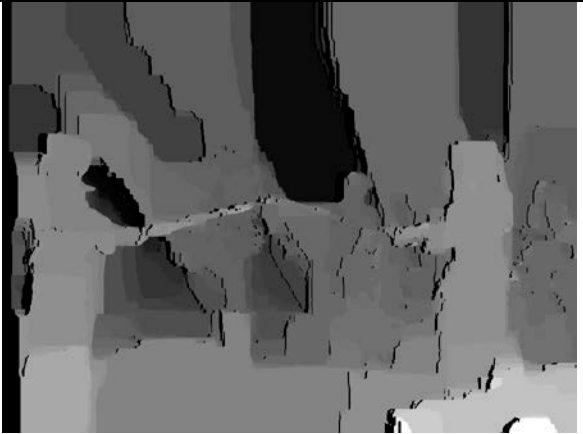
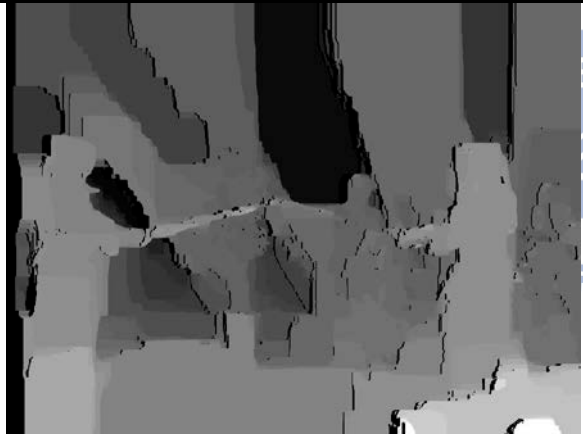

Parallel Depth Warping & Refinement (Kendo for Right Viewpoint)	
	
Depth Warping to Virtual View	Depth Warping + Median Filter (3x3)
	
Depth Warping + Adaptive Median Filter (3x3)	Depth Warping + Proposed Method <i>thread_{cont} = 1% & thread_{desc} = 10</i>

圖 62 右視點平行景深圖映射與針對不同修復演算法之示意圖 (Kendo Sequence)






Standard Sequence: Kendo	
	
Ground Truth	
	
Adaptive Median Filter (3x3)	Median Filter (3x3)
	
Proposed Depth Refinement $thread_{cont} = 1\% \ \& \ thread_{desc} = 10$	Proposed Depth Refinement + AR $thread_{cont} = 1\% \ \& \ thread_{desc} = 10$

圖 63 平行虛擬視點合成針對不同景深修復演算法之示意圖 (Kendo Sequence)

5.4.3 實驗結果討論 (Discussions)

表 10 Kendo Sequence 虛擬視點合成針對各類方法之質量評估表 (PSNR)

		Median Filter	Adaptive Median Filter	Proposed Method
Without AR	PBHF_level_0	33.6659	33.6788	33.8693
	PBHF_level_1	33.6679	33.6806	33.871
	PBHF_level_2	33.6664	33.6801	33.8716
With AR	PBHF_level_0	33.8647	33.885	33.9786
	PBHF_level_1	33.866	33.886	33.9795
	PBHF_level_2	33.8647	33.8855	33.9803
Unit: PSNR				

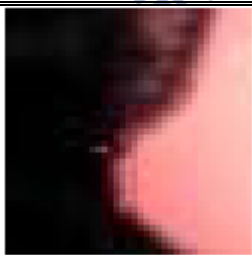
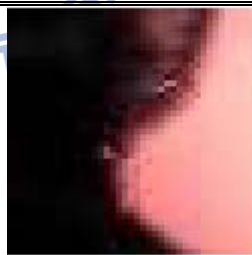
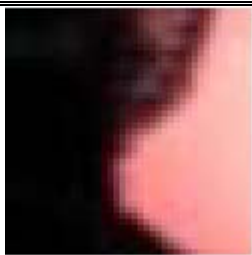
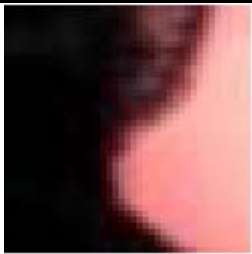



表 11 Kendo Sequence 虛擬視點合成針對各類方法之質量評估表 (CORR)

		Median Filter	Adaptive Median Filter	Proposed Method
Without AR	PBHF_level_0	0.99737	0.99738	0.99749
	PBHF_level_1	0.99737	0.99738	0.99749
	PBHF_level_2	0.99737	0.99738	0.99749
With AR	PBHF_level_0	0.99749	0.9975	0.99755
	PBHF_level_1	0.99749	0.9975	0.99755
	PBHF_level_2	0.99749	0.9975	0.99755
Unit: CORR				

我們針對三種不同的景深圖映射後修補演算法，來做質量評估比較。由表 10 可看出適應性中值濾波器對於合成的影響，平均而言高於中值濾波器約 0.02 dB，這是因為適應性中值濾波器要更新影像強度須有一定的門檻，對於一些邊緣部分的景深資訊，比較不容易被破壞，不過影響程度不大。而本篇論文提出的改進方法則比兩者高約 0.2 dB，這是因為我們利用可依賴性模型，只針對量化誤差部分做修復的動作，而不會破壞邊緣資訊。

值得一提的是 (適應性) 中值濾波器對於金字塔型影像修補的高階層數而言，合成品值反而有降低的趨勢，反觀我們所提出的量化誤差修復改進方式，其合成品值還是會維持一定水準而不降低。主要原因是中值濾波器本身就會使影像有平滑化的效果，而金字塔型影像修補對於高階層數也會使影像的遮蔽區域有模糊化效果，有點像是兩種工具的功能特性有重疊到。至於表 11 中，我們也可看出 CORR 的趨勢和 PSNR 一樣，只是該數據的動態範圍不大，一般會以 PSNR 來評比標準測試照片。

而在主觀品質上，如圖 64 所示 (其放大區域來源在圖 63 中以圓圈表示)。我們可以看出 (適應性) 中值濾波器因為會破壞到正確的景深資訊，所以在物體邊緣處很容易發生量化誤差型態 II (背景顯現於前景破洞處)、或者是鬼影現象 (物體邊緣外移、或是被重複取樣)。反觀本論文提出的修復方法，因為有正確的修補量化誤差，其合成品值逼近正確虛擬影像資訊。

Region 1			
	Median Filter	Adaptive Median Filter	Proposed Method
Without AR			
With AR			
Ground Truth			

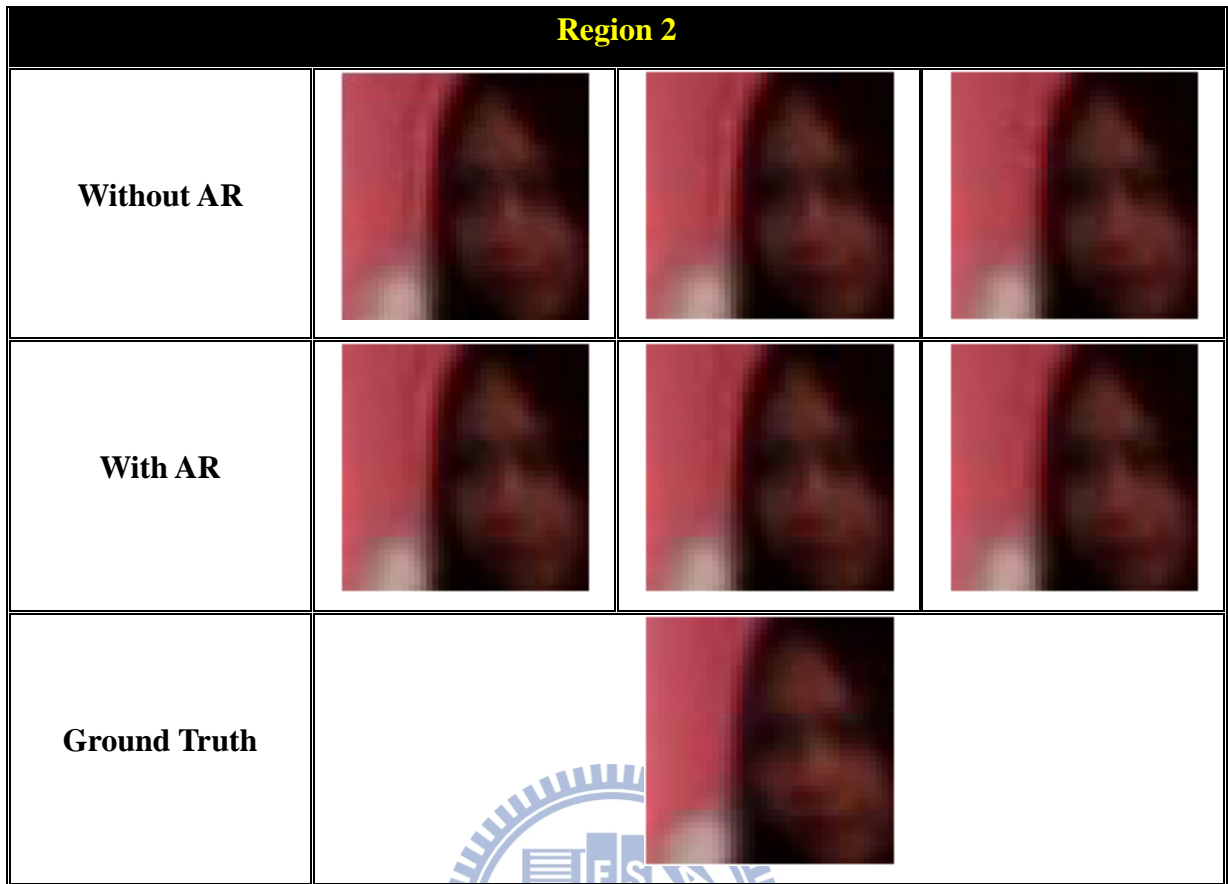


圖 64 Kendo Sequence 虛擬視點合成針對景深修復演算法之部分放大示意圖

最後瑕疵抑制演算法用於合成圖中，可以看到很多細微的瑕疵都被消除、或是減緩許多。若搭配中值濾波器或適應性中值濾波器，可發現 PSNR 多約 0.2dB，但搭配本論文提出的景深修復演算法，其 PSNR 只提高約 0.11dB 左右，這也代表著本論文的景深修復演算法，相較於其它的濾波器而言，在最後合成品質上所造成的瑕疵現象較低一些。

5.4.4 可信賴模型之門檻參數決策 (Threshold Decision about Reliability Model)

本實驗在前面有針對景深圖映射後的各種修復演算法加以實驗與探討，對於本篇論文所提出的演算法架構而言效果最佳；但是所使用的可信賴模型中，須考慮到兩種參數，分別為 **thread_cont** 與 **thread_desc**，用於連續圖與下降圖的生成。以下將會針對 Kendo 標準測試影像，歸納出最佳的門檻值，使得最後的合成品質上，其質量評估可以達到最佳化效果。

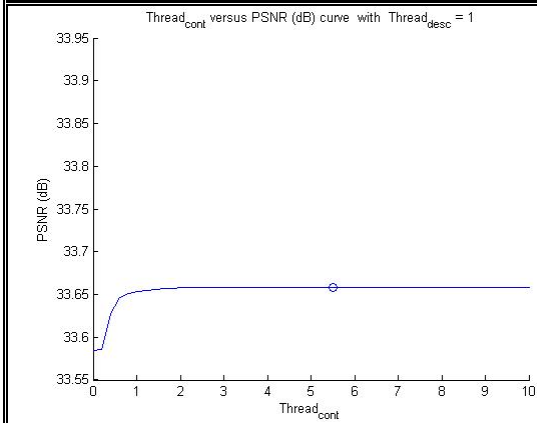
為了簡化問題，我們使用最簡單的合成系統，亦即考慮金字塔型影像修補階層 0、並且不採用瑕疵抑制演算法，如下表 12 所示：

表 12 環境設定：平行景深圖映射後修復方法中門檻參數對合成品值影響

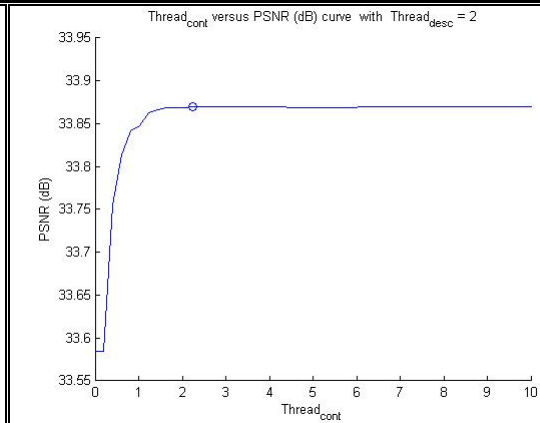
Parameters	Setting	Parameters	Setting
sampling_method	1	TW_blending_mode	1
depth_sampling_method	2	Pyramid_level	0
DW_direct_mode	0	AR_mode	0
DW_case_mode	1	thread_cont	0%~10%
DW_refine_mode	2	thread_dest	1~10

由於下降圖的門檻為一自然數，因此我們可以針對不同整數值的 **thread_desc**，統計出 **thread_cont** 對於最後合成的 PSNR 數值上影響。如圖 65 所示。並且統計出在不同的下降圖門檻情況下，所對應出最佳的連續圖門檻與合成後的 PSNR 值，如表 13 所示。而這些最佳點也會個別標示在曲線上。

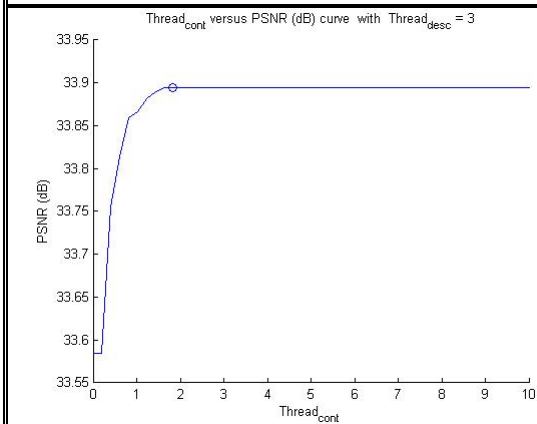
Thread_{cont} 對 合成影像 PSNR 之趨勢線



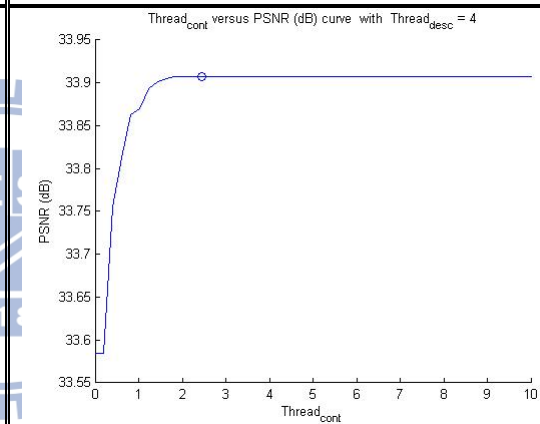
(a) Thread_{desc} = 1



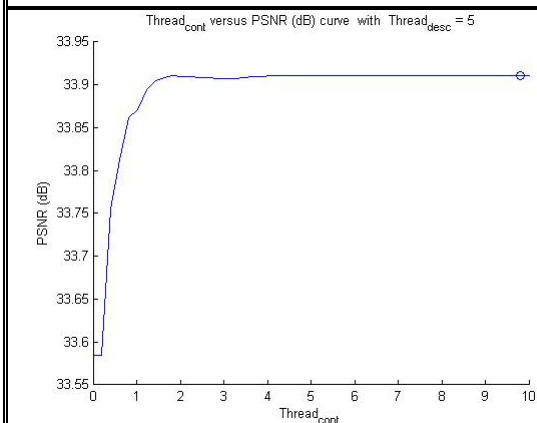
(b) Thread_{desc} = 2



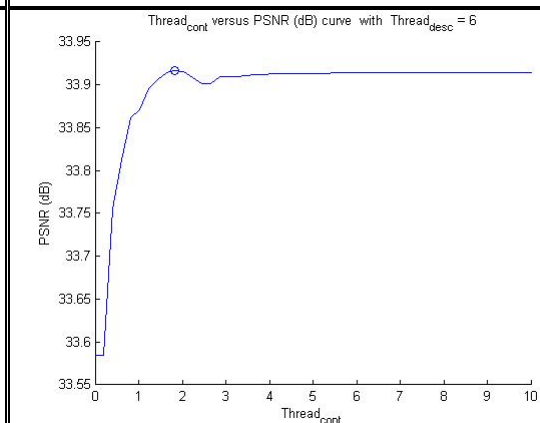
(c) Thread_{desc} = 3



(d) Thread_{desc} = 4



(e) Thread_{desc} = 5



(f) Thread_{desc} = 6

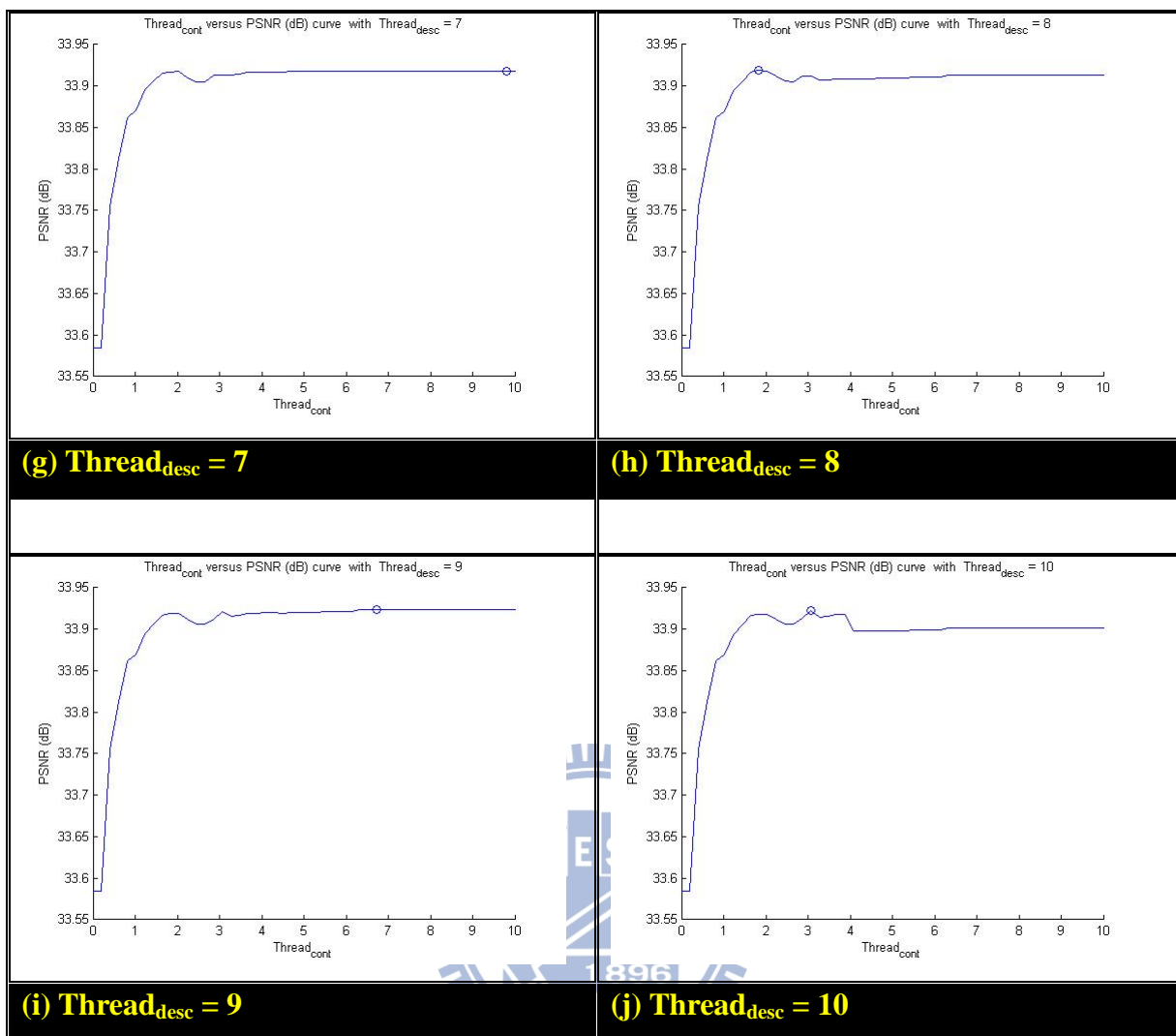


圖 65 不同下降圖門檻下，連續圖門檻對於合成影像 PSNR 之曲線圖

表 13 不同下降圖門檻下，所對應的最佳連續圖門檻與最佳合成影像 PSNR

下降圖門檻	1	2	3	4	5
最佳連續圖門檻 (%)	5.5102	2.2449	1.8367	2.4490	9.7959
最佳 PSNR	33.6582	33.8696	33.8949	33.9078	33.9113

下降圖門檻	6	7	8	9	10
最佳連續圖門檻 (%)	1.8367	9.7959	1.8367	6.7347	3.0612
最佳 PSNR	33.9162	33.9184	33.9186	33.9233	33.9219

由圖 65 可發現當下降圖的門檻值越低，其 PSNR 上界會越小，尤其是當門檻值為 1 的時候，PSNR 下降趨勢較明顯，這是因為下降圖門檻過於嚴苛，使得景深圖映射後的很多量化誤差都被視為遮蔽區域處理而不做修補的動作。

若將下降圖門檻設定過高，雖然 PSNR 上界也會跟著提高，可是連續圖門檻之於 PSNR 的曲線，會發生不穩定的波動現象產生。一個可能性解讀是，對於寬鬆的下降圖門檻，容易把一些遮蔽區域視為量化誤差做修補，若此時選取較小(嚴苛)的連續圖門檻，可以有效地將這些被誤判量化誤差的遮蔽區域排除掉，造成整體的 PSNR 提高；但隨著連續圖門檻不斷的增高，遮蔽區域也慢慢被修補掉，可想成是在紋理映射階段中，較高(寬鬆)的連續圖門檻扮演著擷取周圍已知的參考資訊來填補遮蔽區域的腳色。因此越寬鬆的連續圖門檻，遮蔽區域填補的越多，也就代表著填補的不準確性也提高越多。

因此利用上述討論加上表 13，可以歸納出一個小結論，那就是連續圖門檻在 1% ~ 3% ，其量化誤差修復效果會比較好；而表中雖然有些連續圖門檻達 9%，且 PSNR 也很高，但那是因為遮蔽區域被直接修補的不確定因素所造成。利用此結論，我們設定連續圖門檻值為 2%，並且嘗試統計出下降值門檻對合成影像 PSNR 的曲線圖，如圖 66 所示：

根據圖中曲線可知對於合理的連續圖門檻下，下降圖門檻即使提高到 20，對於 PSNR 是趨於收斂狀態，代表著不會因放寬遮蔽區域判斷的限制而造成 PSNR 下降，這與 3.2.3 小節所討論的實驗是一致的。因此下降圖門檻是可以設定到 20，只要連續圖門檻約略在 1% ~ 3% 之間。

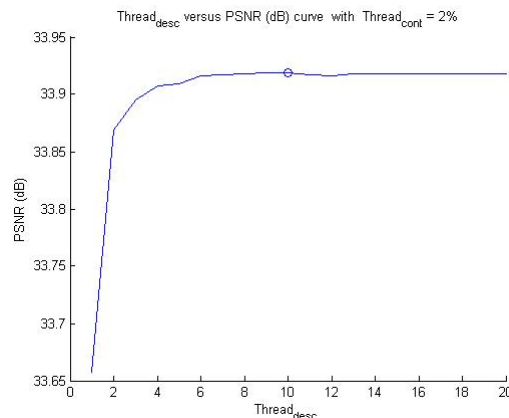


圖 66 連續圖門檻 2% 下，下降圖門檻對於合成影像 PSNR 之曲線圖

5.5 景深圖各種升取樣方法對合成品質影響 (Up-Sampling of Depth versus Quality of Synthetic Virtual Images)

5.5.1 環境設定 (Environmental Setting)

表 14 環境設定：景深圖各種升取樣方法對合成品質影響

Parameters	Setting	Parameters	Setting
sampling_method	1	TW_blending_mode	0
depth_sampling_method	0~2	Pyramid_level	0
DW_direct_mode	0	AR_mode	0
DW_case_mode	0	thread_cont	nope
DW_refine_mode	0	thread_dest	nope

表 14 中代表的意思是，我們以雙三次插值法做為色彩影像的升/降取樣方法；以廣義正向映射演算法來做景深圖映射；使用中值濾波器來做映射後修補動作；紋理映射後的線性疊合是採用 2.7 章節所介紹的 (25)式；金字塔影像修補階層則採用 level 0，並且不使用後置瑕疵抑制演算法來消除雜訊。

接著我們分別以高斯插值法、雙三次插值法、與複本插值法，用於景深圖升取樣方法，來探討對於合成品質的影響。

5.5.2 實驗結果 (Simulation Results)

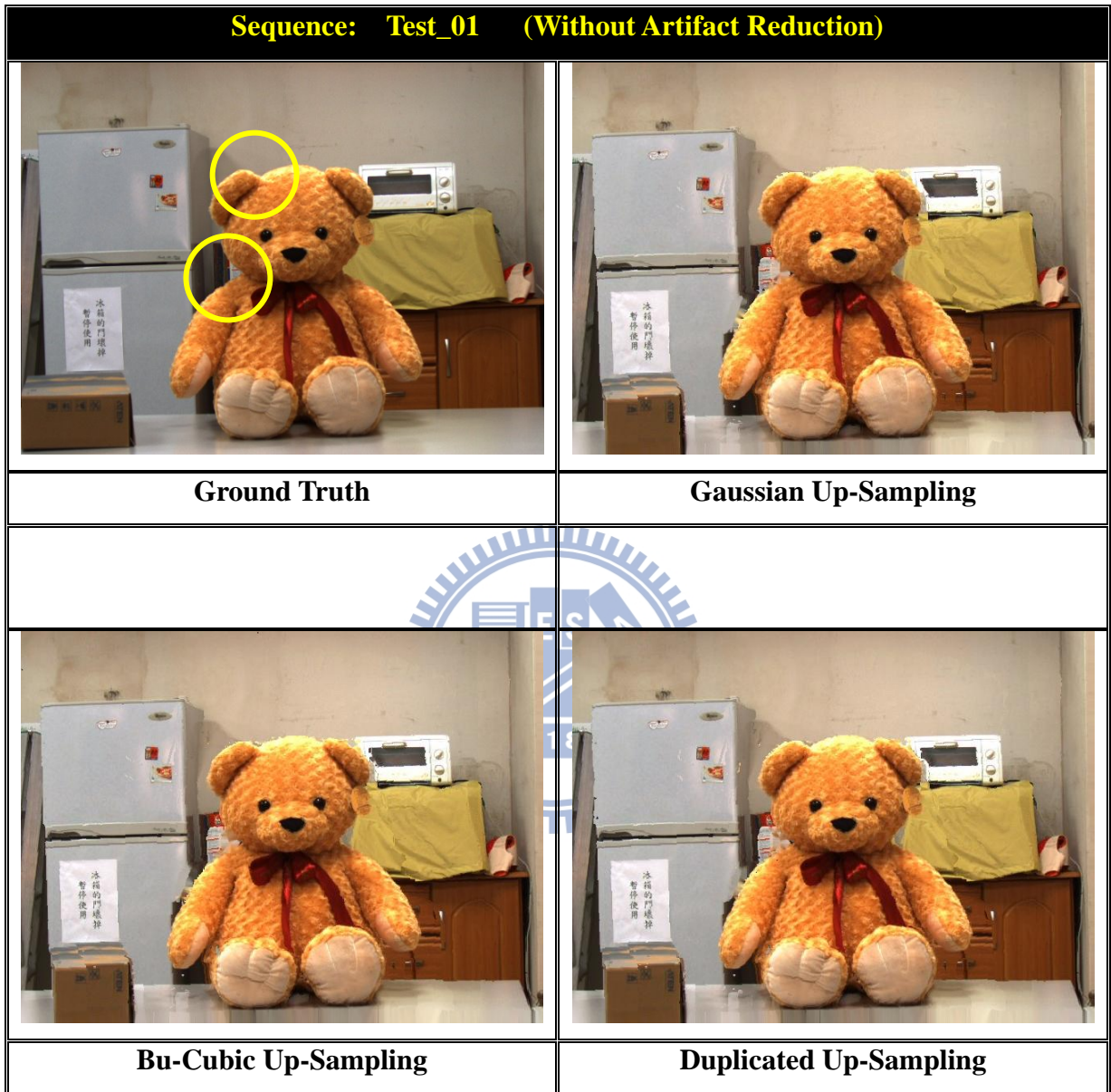


圖 67 虛擬視點合成針對不同升取樣方法之示意圖 (Test_01 Sequence)

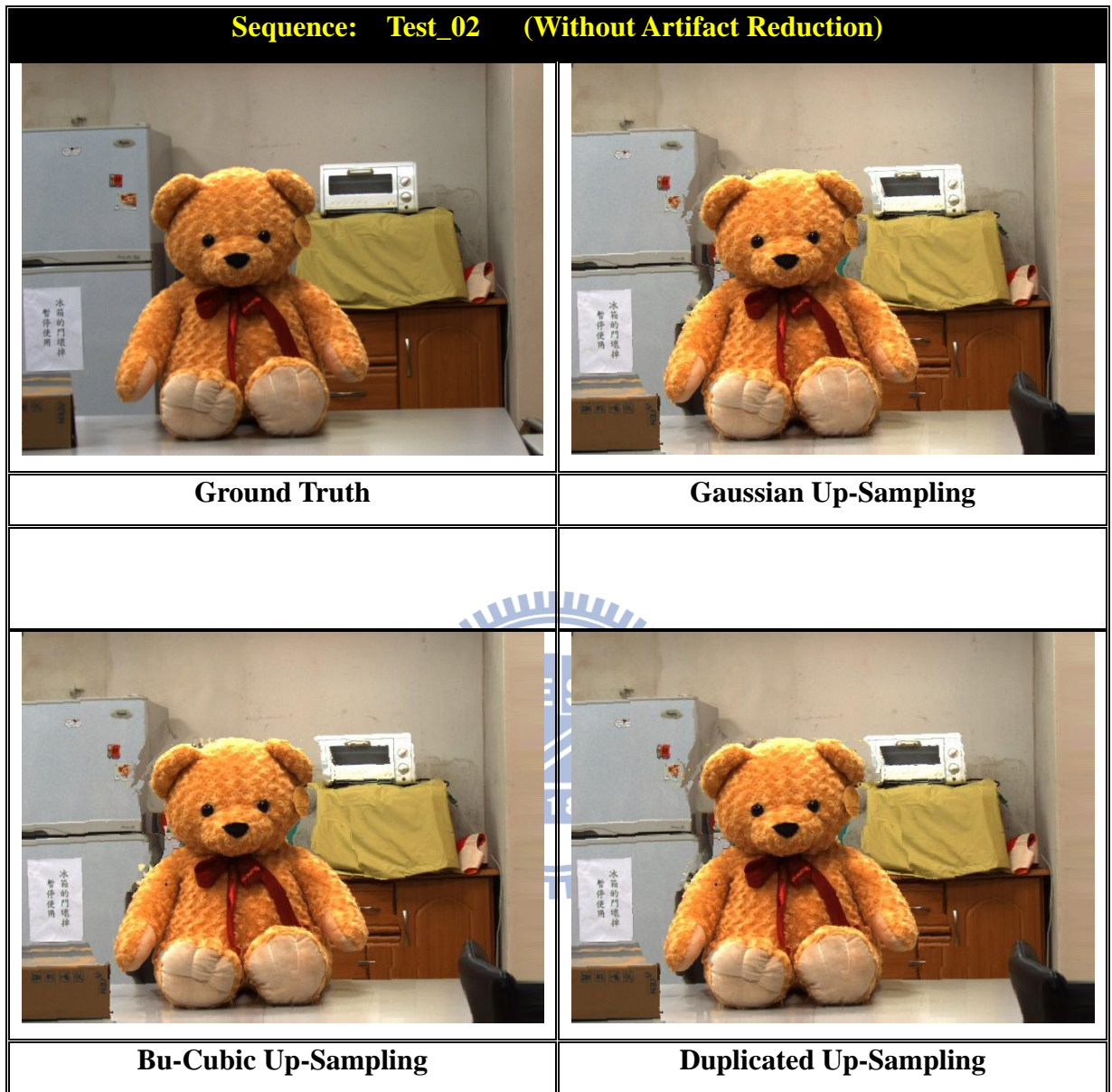


圖 68 虛擬視點合成針對不同升取樣方法之示意圖 (Test_02 Sequence)

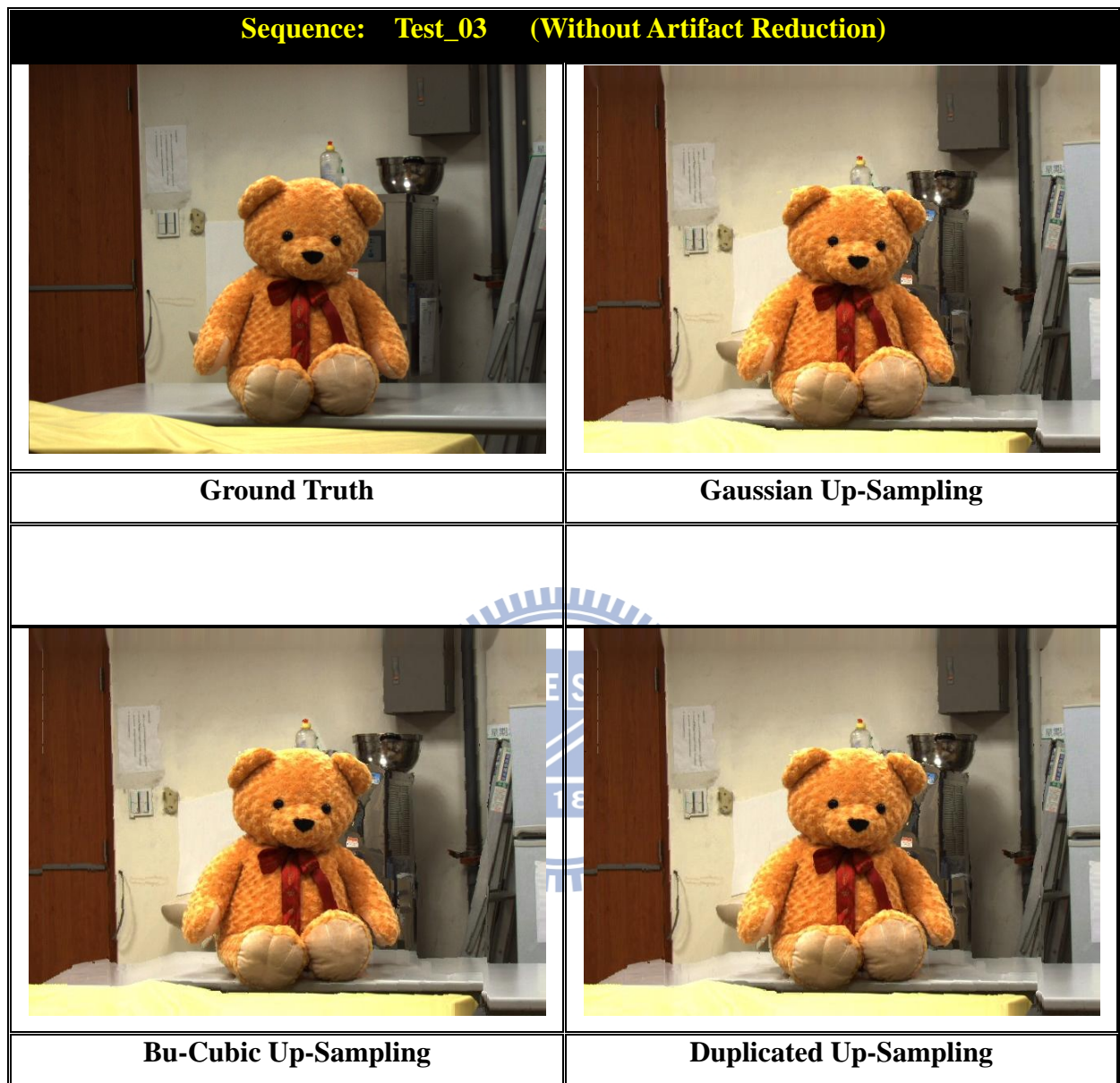


圖 69 虛擬視點合成針對不同升取樣方法之示意圖 (Test_03 Sequence)

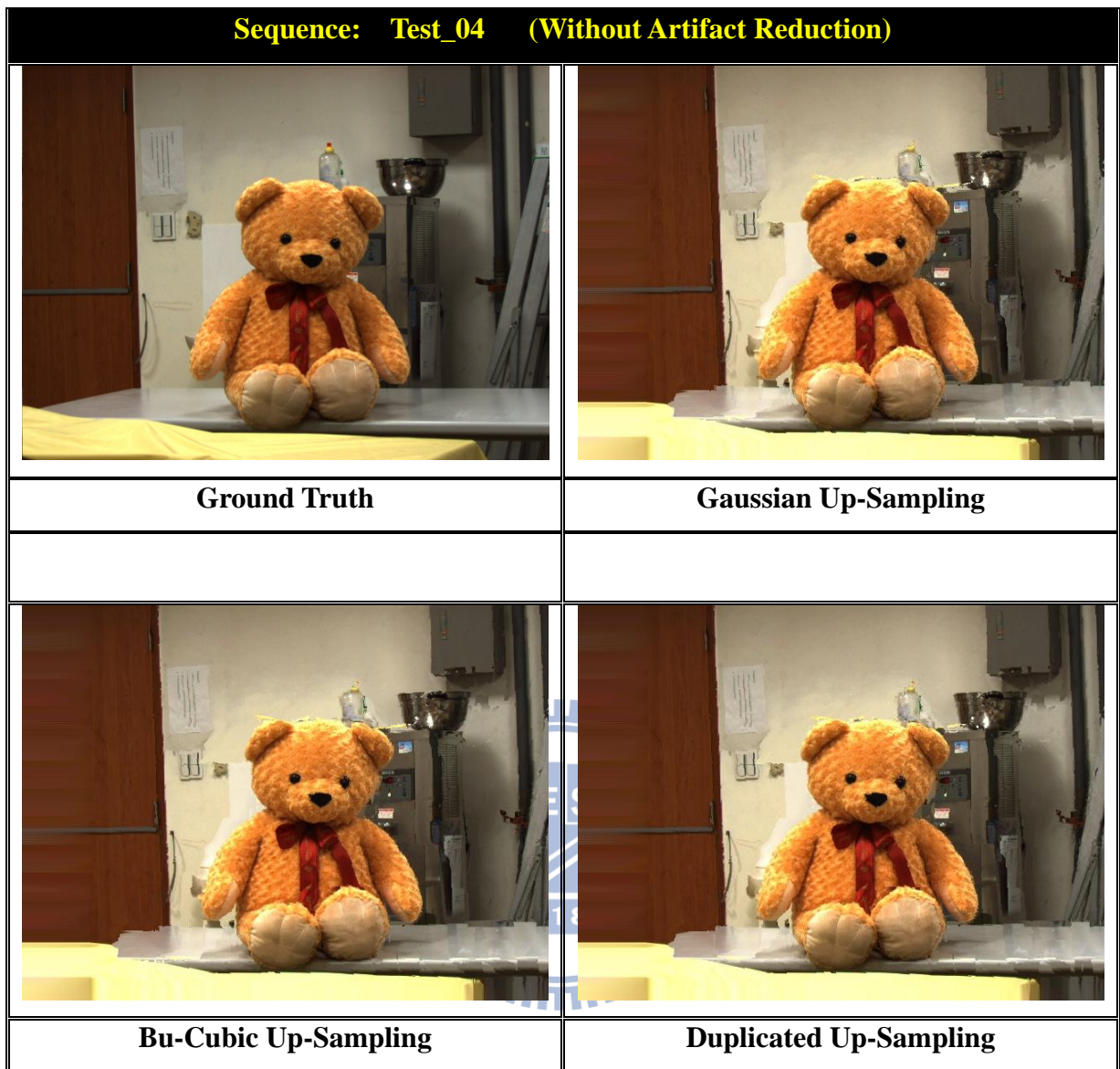


圖 70 虛擬視點合成針對不同升取樣方法之示意圖 (Test_04 Sequence)

5.5.3 實驗結果討論 (Discussions)

表 15 各類測試影像的虛擬視點合成針對不同升取樣方法之質量評估表 (CORR)

	Test_01	Test_02	Test_03	Test_04
Gaussian	0.8125	0.78407	0.67606	0.77135
Bi-Cubic	0.81239	0.78373	0.67572	0.77114
Duplicated	0.81278	0.78293	0.67629	0.77158
Unit: CORR				

由表 15 可發現對於景深圖升取樣方法，複本插值法 (Duplicated Interpolation) 最佳、高斯插值法 (Gaussian Interpolation) 次之、雙三次插值法 (Bi-cubic Interpolation) 最差，這是因為雙三次插值法容易讓景深影像銳利化，也代表著若景深值估錯，則該插值法會讓其雜訊擴大，造成最後的合成品質上會有尖銳狀的瑕疵現象產生。

而高斯插值法會讓景深影像有模糊化的效果，這意味著不僅是雜訊、連物體邊緣景深也跟著被平滑化，使的最後的合成品質上雖沒有如雙三次插值法嚴重，但其物體邊緣或紋理上可能會有模糊的效果。不過在主觀品質上是可以被接受的。

至於複本插值法，它並沒有使用太複雜的核心來做濾波，因此不會破壞到景深資訊。但缺點在於使用後會有細小方塊狀的瑕疵現象產生。可參考圖 71 (其放大區域來源在圖 67 中以圓圈表示)。

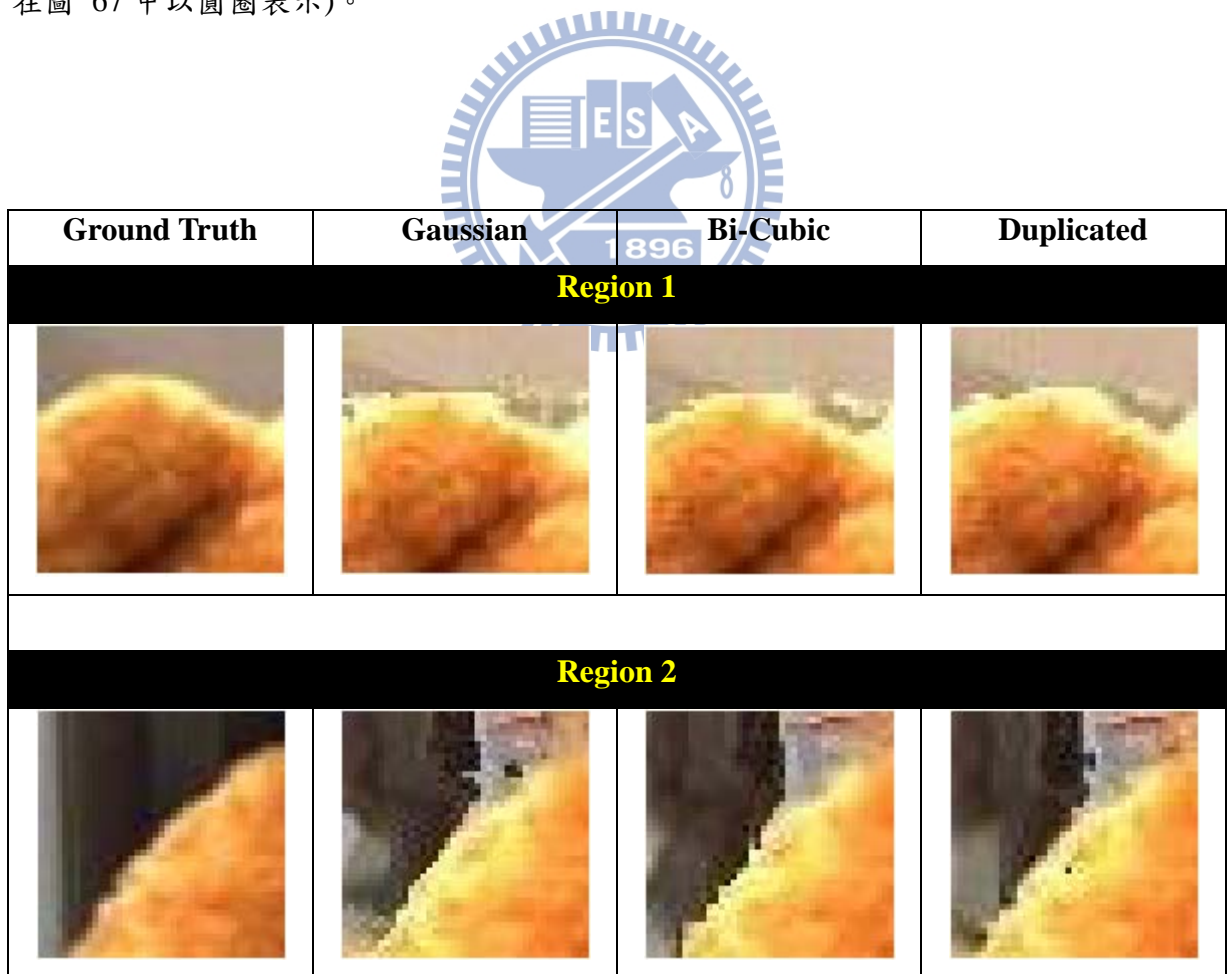


圖 71 Test_01 虛擬視點合成針對景深圖升取樣之部分放大示意圖

5.6 景深圖正/逆向映射對合成品質影響 (Forward/Backward Depth Warping Algorithm versus Quality of Synthetic Virtual Images)

5.6.1 環境設定 (Environmental Setting)

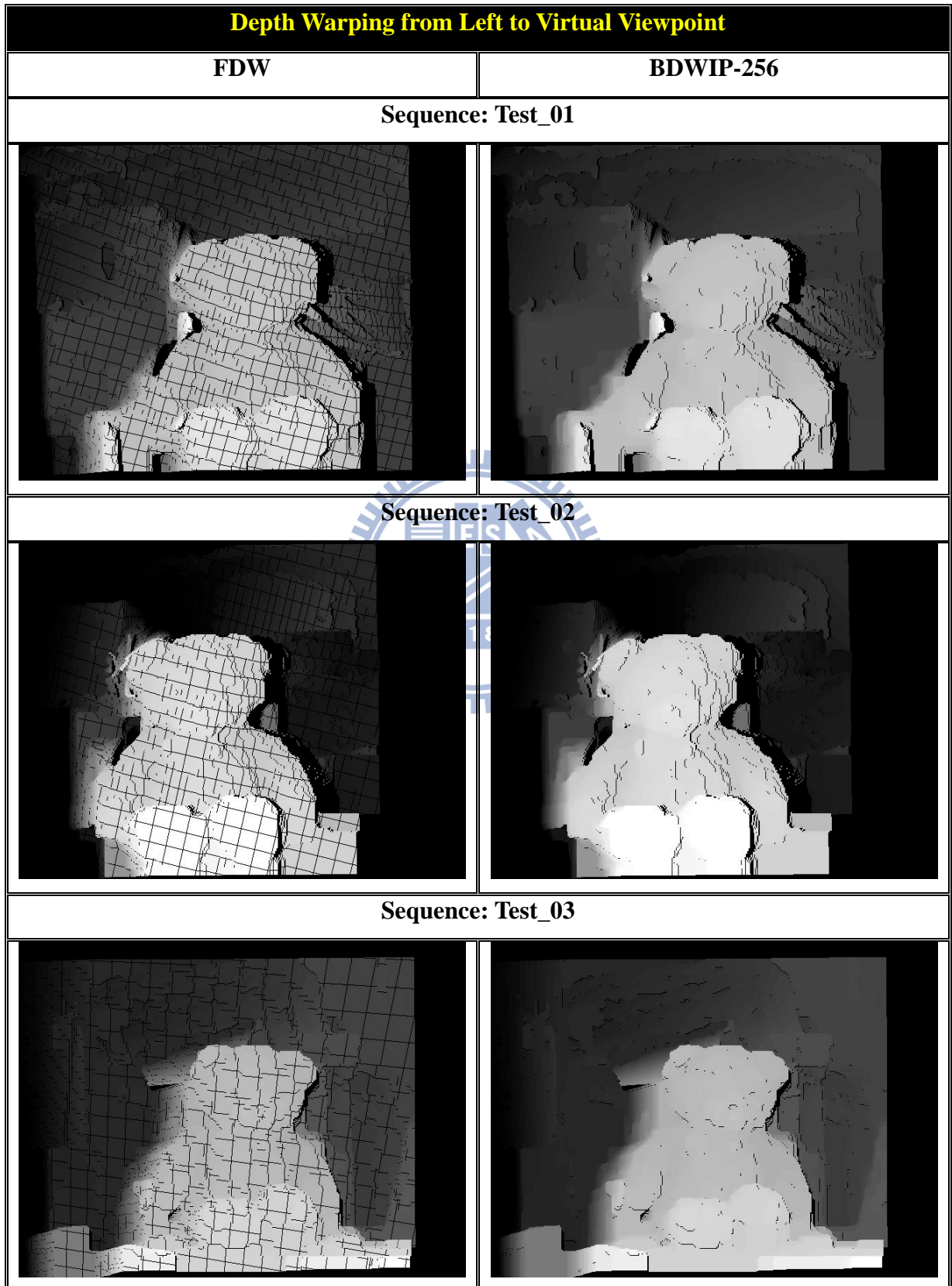
表 16 環境設定：景深圖正/逆向映射對合成品質影響

Parameters	Setting	Parameters	Setting
sampling_method	1	TW_blending_mode	0
depth_sampling_method	2	Pyramid_level	0
DW_direct_mode	0~1	AR_mode	0
DW_case_mode	0	thread_cont	nope
DW_refine_mode	0~1	thread_dest	nope

表 16 中代表的意思是，我們以雙三次插值法做為色彩影像的升/降取樣方法；景深圖升取樣方法採用複本插值法；以廣義映射公式來做景深圖映射；紋理映射後的線性疊合是採用 2.7 章節所介紹的 (25)式；金字塔影像修補階層則採用 level 0，並且不使用後置瑕疵抑制演算法消除雜訊。

接著我們分別以正向或逆向映射演算法作為景深映射機制，並探討對於合成品質的影響。此外，針對正/逆向映射的差異，會分別採取中值濾波器與適應性中值濾波器來消除量化誤差。

5.6.2 實驗結果 (Simulation Results)



Sequence: Test_04

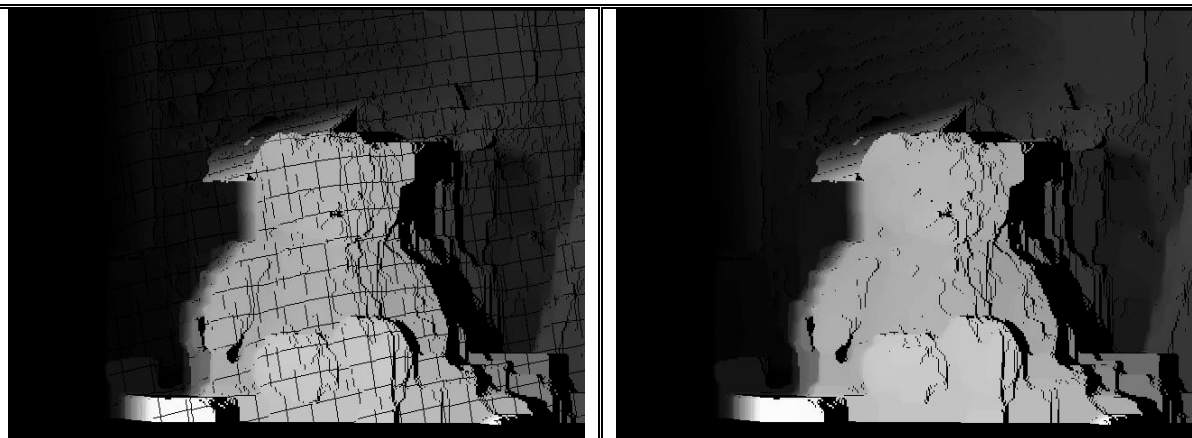


圖 72 各左視點測試影像針對正/逆向景深圖映射示意圖

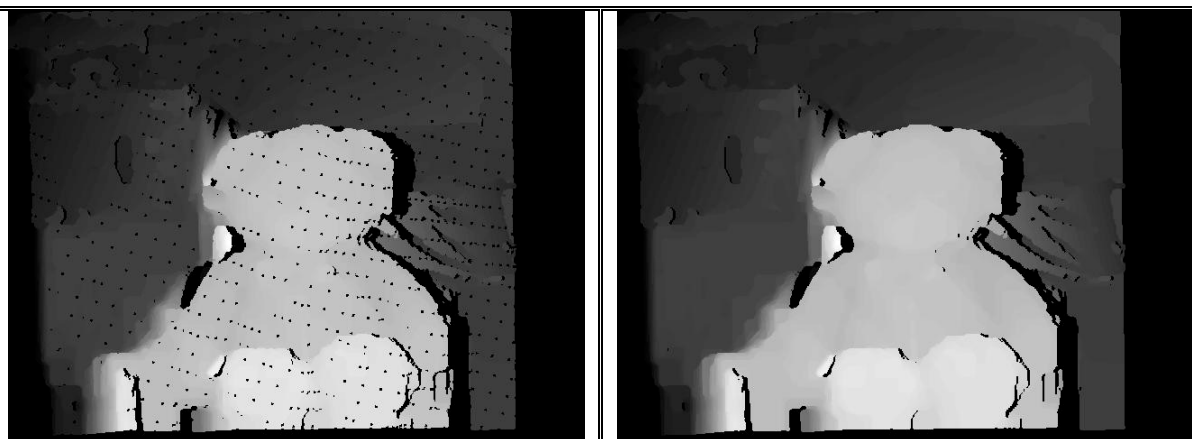


Warped Depth Refinement from Left to Virtual Viewpoint

FDW + Median Filter (3x3)

**BDWIP-256 + Adaptive Median Filter
(3x3)**

Sequence: Test_01



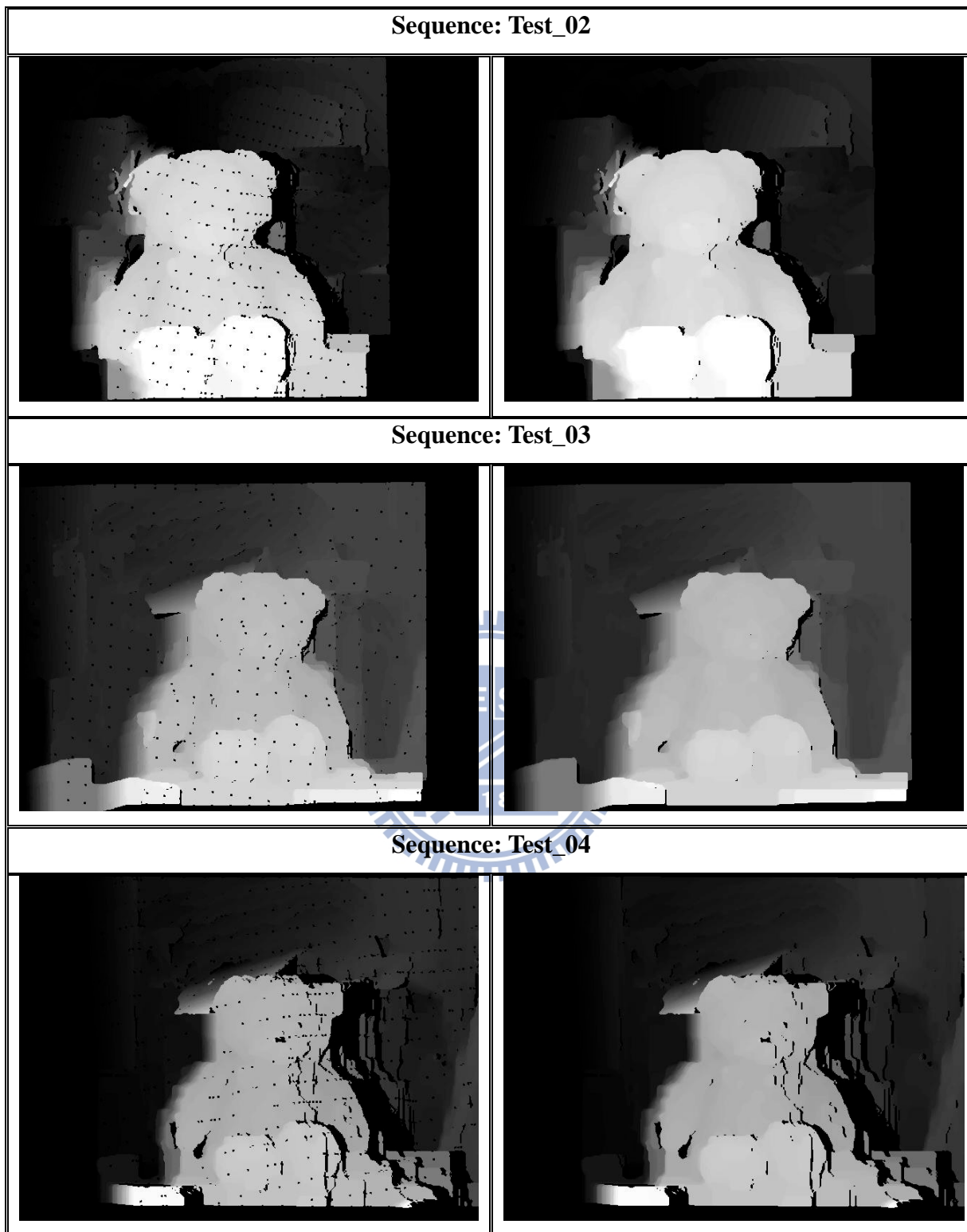
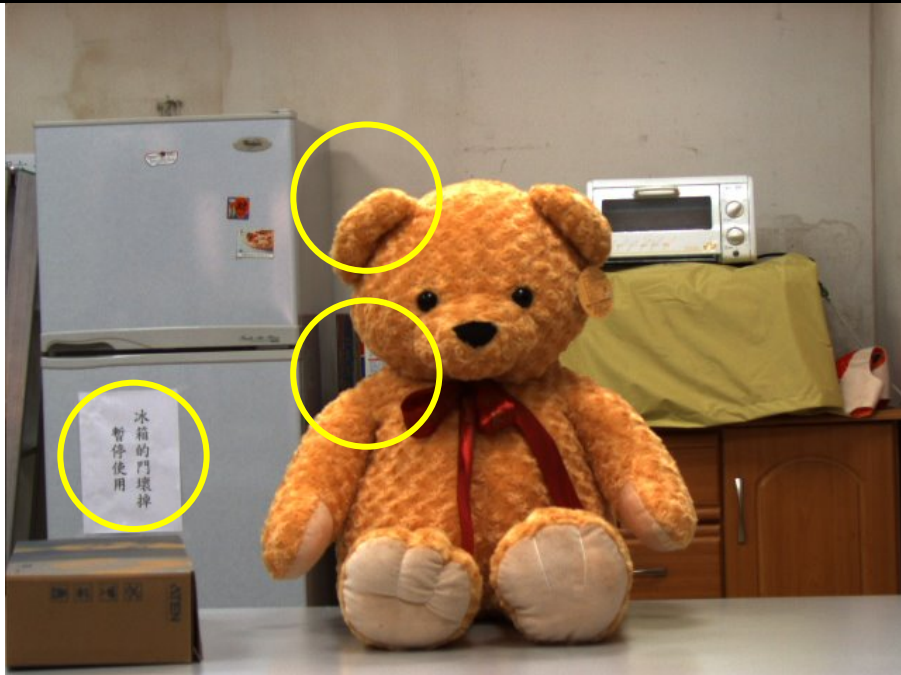


圖 73 各左視點測試影像針對正/逆向景深圖映射後修補示意圖

Sequence: Test_01 (Without Artifact Reduction)



Ground Truth



FDW



BDWIP-256

圖 74 虛擬視點合成針對正/逆向景深映射演算法之示意圖 (Test_01 Sequence)

Sequence: Test_02 (Without Artifact Reduction)



Ground Truth



FDW



BDWIP-256

圖 75 虛擬視點合成針對正/逆向景深映射演算法之示意圖 (Test_02 Sequence)

Sequence: Test_03 (Without Artifact Reduction)



Ground Truth



FDW



BDWIP-256

圖 76 虛擬視點合成針對正/逆向景深映射演算法之示意圖 (Test_03 Sequence)

Sequence: Test_04 (Without Artifact Reduction)



Ground Truth



FDW



BDWIP-256

圖 77 虛擬視點合成針對正/逆向景深映射演算法之示意圖 (Test_04 Sequence)

5.6.3 實驗結果討論 (Discussions)

表 17 各類測試影像的虛擬視點合成針對正/逆向景深映射之質量評估表 (CORR)

	Test_01	Test_02	Test_03	Test_04
FDW	0.81278	0.78293	0.67629	0.77158
BDWIP-256	0.81113	0.78061	0.67436	0.77199
Unit: CORR				

由表 17 可發現正向景深映射演算法於 CORR 上的表現，大多數會比逆向景深映射演算法好，最主要的因素在於景深圖估計不夠準確、與相機參數精確度問題；此外逆向景深映射演算法採用反矩陣運算，對於錯誤的景深資訊，也很容易擷取到錯誤的紋理。因此逆向景深映射演算法在 CORR 表現上未能達到預期的效果。



(a) Forward Depth Warping



(b) Backward Depth Warping

圖 78 正/逆向景深映射對於線性疊合後的影響

不過在主觀品質上，可發現逆向景深映射演算法所產生的瑕疵現象，比正向景深映射演算法還要少。如上圖 78 所示。這是因為參考景深影像的平坦區域對於逆向映射而言，可確保其連續性；反觀正向映射會造成很多不必要的量化誤差，而這些量化誤差又使用不完全正確的濾波器來將之修復，造成瑕疵現象變多。圖 79 為合成後的部分放大示意圖（其放大區域來源在圖 74 中以圓圈表示）：


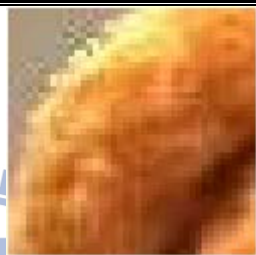







	Ground Truth	FDW	BDWIP-256
ROI 1			
ROI 2			
ROI 3			

圖 79 Test_01 虛擬視點合成針對正逆向景深映射之部分放大示意圖

5.7 金字塔型影像修補技術對合成品值影響 (Pyramid-based Hole Filling versus Quality of Synthetic Virtual Images)

5.7.1 環境設定 (Environmental Setting)

表 18 環境設定：金字塔型影像修補技術對合成品值影響

Parameters	Setting	Parameters	Setting
sampling_method	1	TW_blending_mode	0
depth_sampling_method	2	Pyramid_level	0~2
DW_direct_mode	1	AR_mode	0
DW_case_mode	0	thread_cont	nope
DW_refine_mode	1	thread_dest	nope

表 18 中代表的意思是，我們以雙三次插值法做為色彩影像的升/降取樣方法；景深圖升取樣方法採用複本插值法；以廣義逆向映射演算法來做景深圖映射，並以適應性中值濾波器消除剩餘的量化誤差；紋理映射後的線性疊合是採用 2.7 章節所介紹的 (25) 式；且最後的合成影像不使用後置瑕疵抑制演算法消除雜訊，目的在於展示金字塔影像修補技術的好處。

接著我們分別以金字塔型影像修補技術中的階層 0 ~ 階層 2 做實驗，並且比較對於最後合成品質的影響。值得注意的是，PBHF-level-0 相當於傳統的影像修補技術 (In-Painting)。

5.7.2 實驗結果 (Simulation Results)

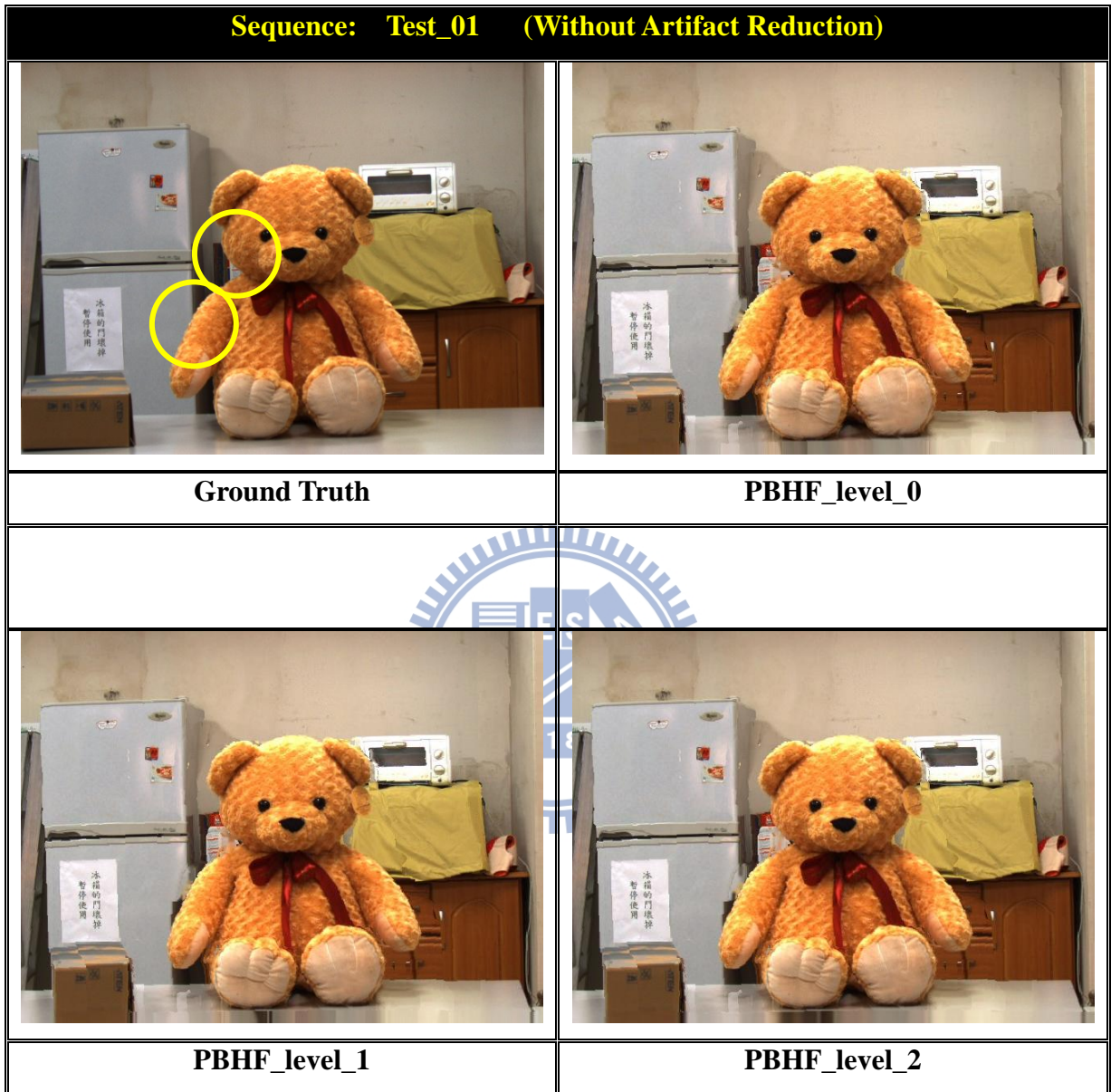


圖 80 虛擬視點合成針對不同金字塔型影像修補階層之示意圖 (Test_01 Sequence)

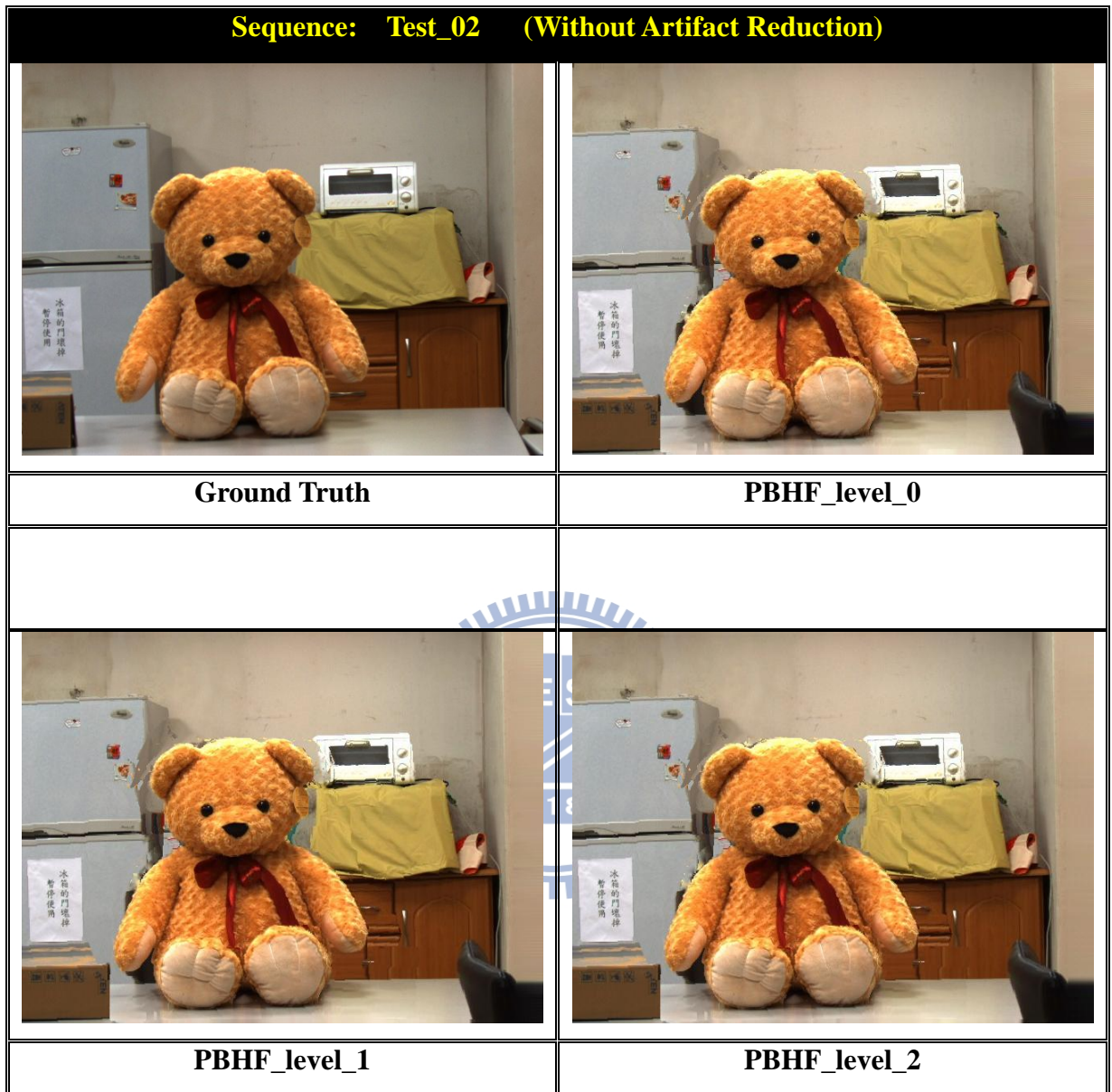


圖 81 虛擬視點合成針對不同金字塔型影像修補階層之示意圖 (Test_02 Sequence)

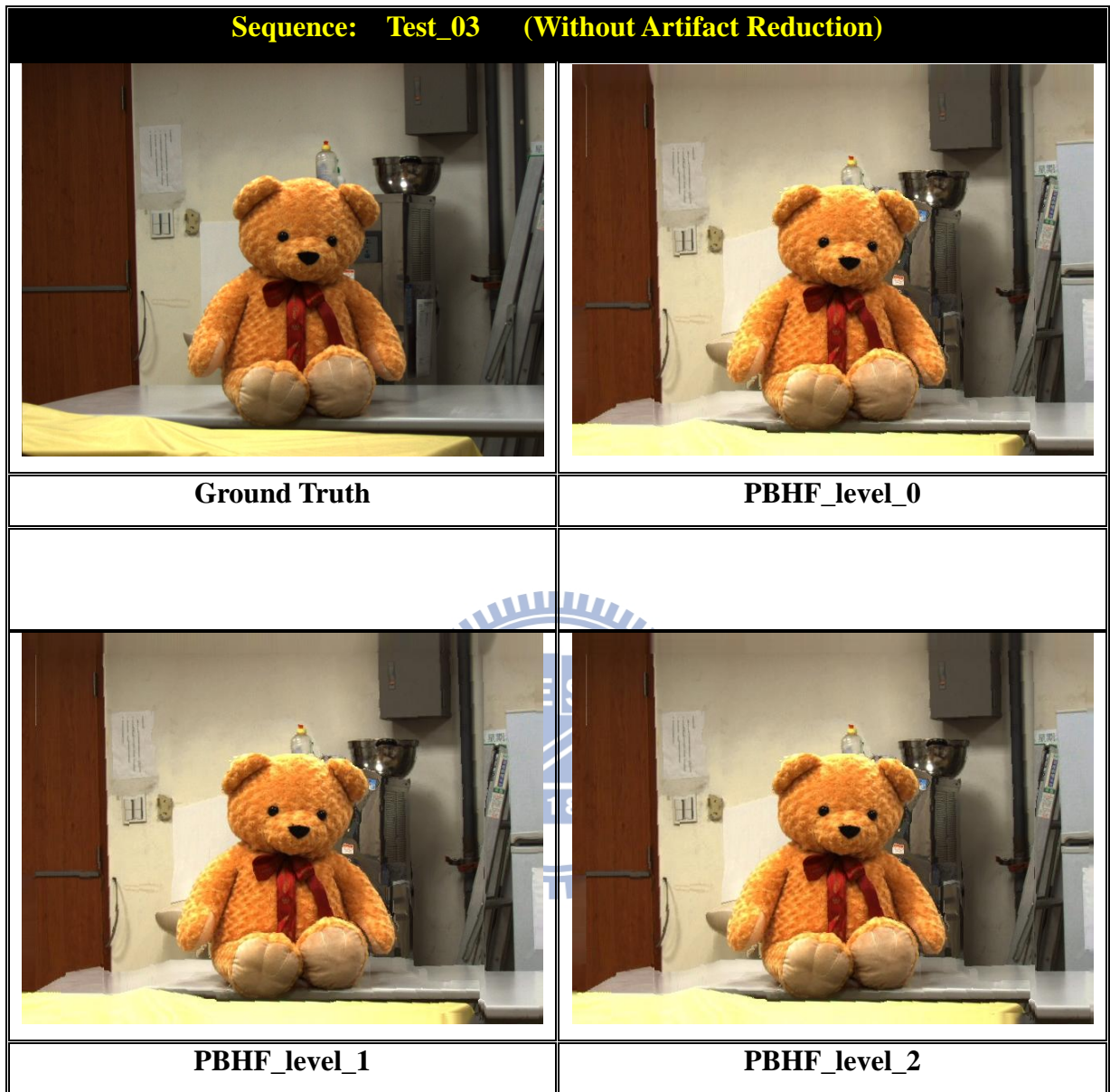


圖 82 虛擬視點合成針對不同金字塔型影像修補階層之示意圖 (Test_03 Sequence)

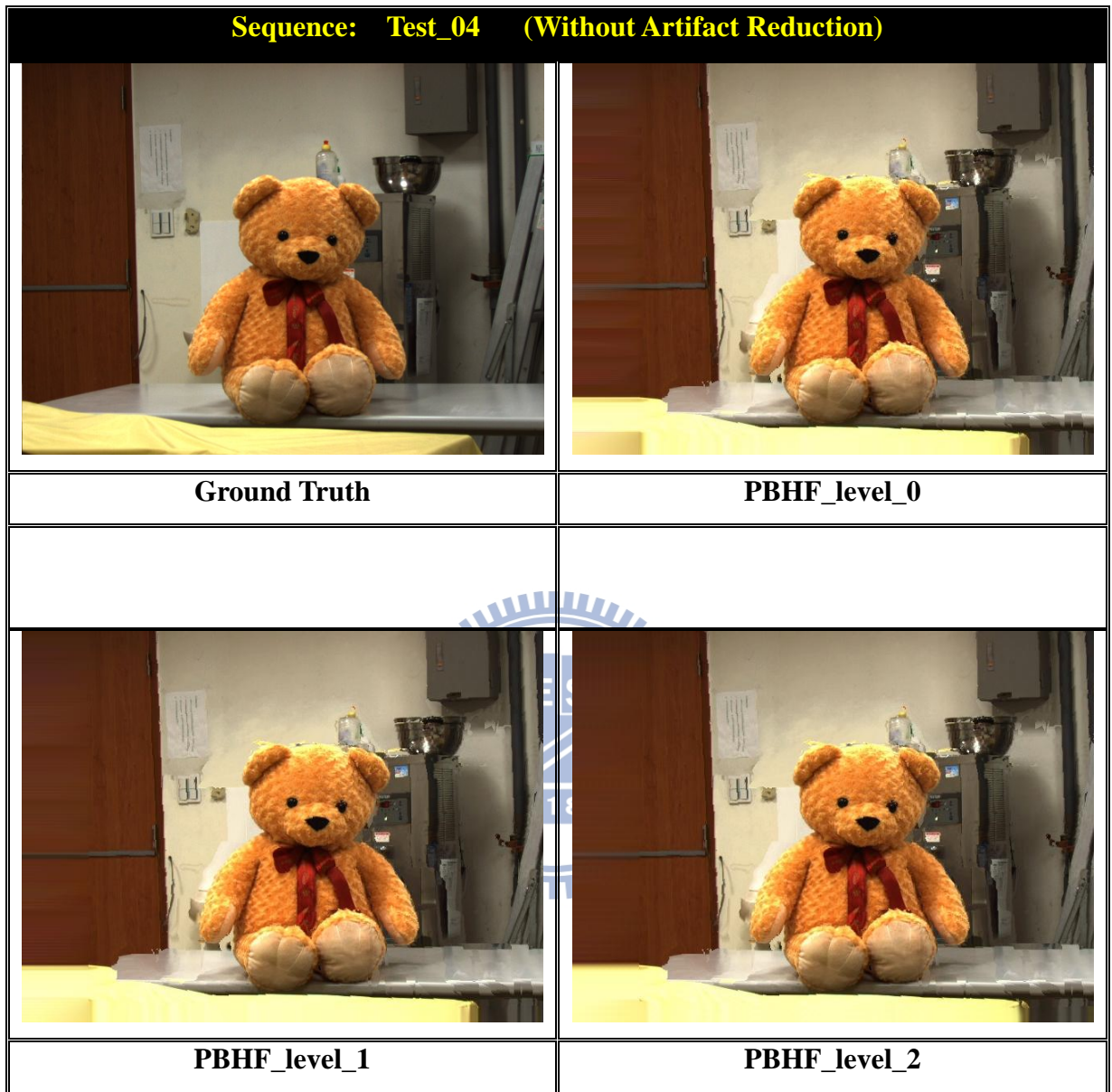


圖 83 虛擬視點合成針對不同金字塔型影像修補階層之示意圖 (Test_04 Sequence)

5.7.3 實驗結果討論 (Discussions)

表 19 各類測試影像的虛擬視點合成針對金字塔型影像修補之質量評估表 (CORR)

	Test_01	Test_02	Test_03	Test_04
PBHF_level_0	0.81113	0.78061	0.67436	0.77199
PBHF_level_1	0.80768	0.78151	0.67469	0.7745
PBHF_level_2	0.80165	0.78081	0.67384	0.77493
Unit: CORR				

由表 19 可以發現金字塔影像補洞，平均而言在階層數為 1 下效果最佳，這是因為階層數越高，其遮蔽區域對於影像修補技術所估算出來的紋理會越加模糊，尤其是越大範圍的遮蔽區域會越嚴重。

不過在主觀品質上，若單純使用影像修補技術來做補洞（相當於 PBHF_level_0），會發現遮蔽區域很容易收到邊緣因合成所產生的雜訊影響。而金字塔型的階層數越高，其雜訊會被壓抑的越低，因此在做影像修補時比較自然一些。如圖 84 所示（其放大區域來源在圖 80 中以圓圈表示）：

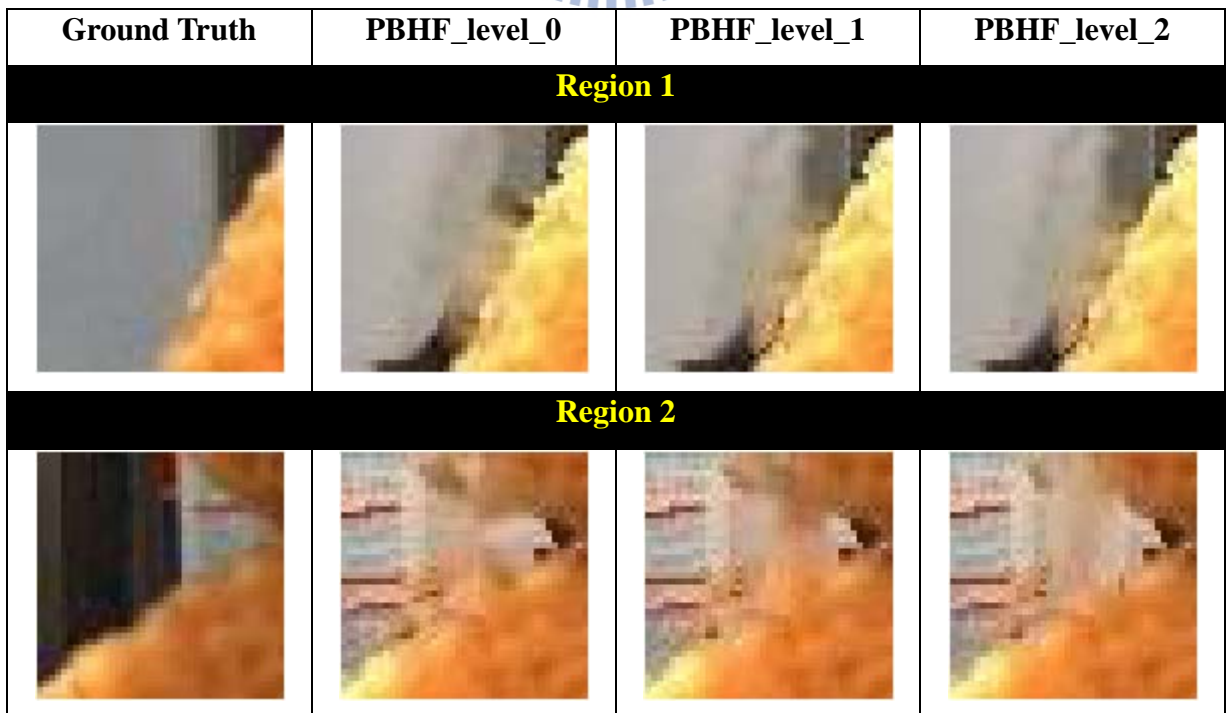


圖 84 Test_01 虛擬視點合成針對金字塔型影像修補階層之部分放大示意圖

5.8 瑕疵抑制方法對合成品值影響 (Artifact Reduction versus Quality of Synthetic Virtual Images)

5.8.1 環境設定 (Environmental Setting)

表 20 環境設定：瑕疵抑制方法對合成品值影響

Parameters	Setting	Parameters	Setting
sampling_method	1	TW_blending_mode	0
depth_sampling_method	2	Pyramid_level	0~2
DW_direct_mode	1	AR_mode	0~1
DW_case_mode	0	thread_cont	nope
DW_refine_mode	1	thread_dest	nope

表 18 中代表的意思是，我們以雙三次插值法做為色彩影像的升/降取樣方法；景深圖升取樣方法採用複本插值法；以廣義逆向映射演算法來做景深圖映射，並以適應性中值濾波器消除剩餘的量化誤差；紋理映射後的線性疊合是採用 2.7 章節所介紹的 (25) 式。

接著分別對於合成圖有/無使用瑕疵抑制演算法來做最後的品質影響。此外，我們也會根據金字塔型影像修補技術在不同的階層下，使用瑕疵抑制演算法後的品質提升有多少加以討論。

5.8.2 實驗結果 (Simulation Results)

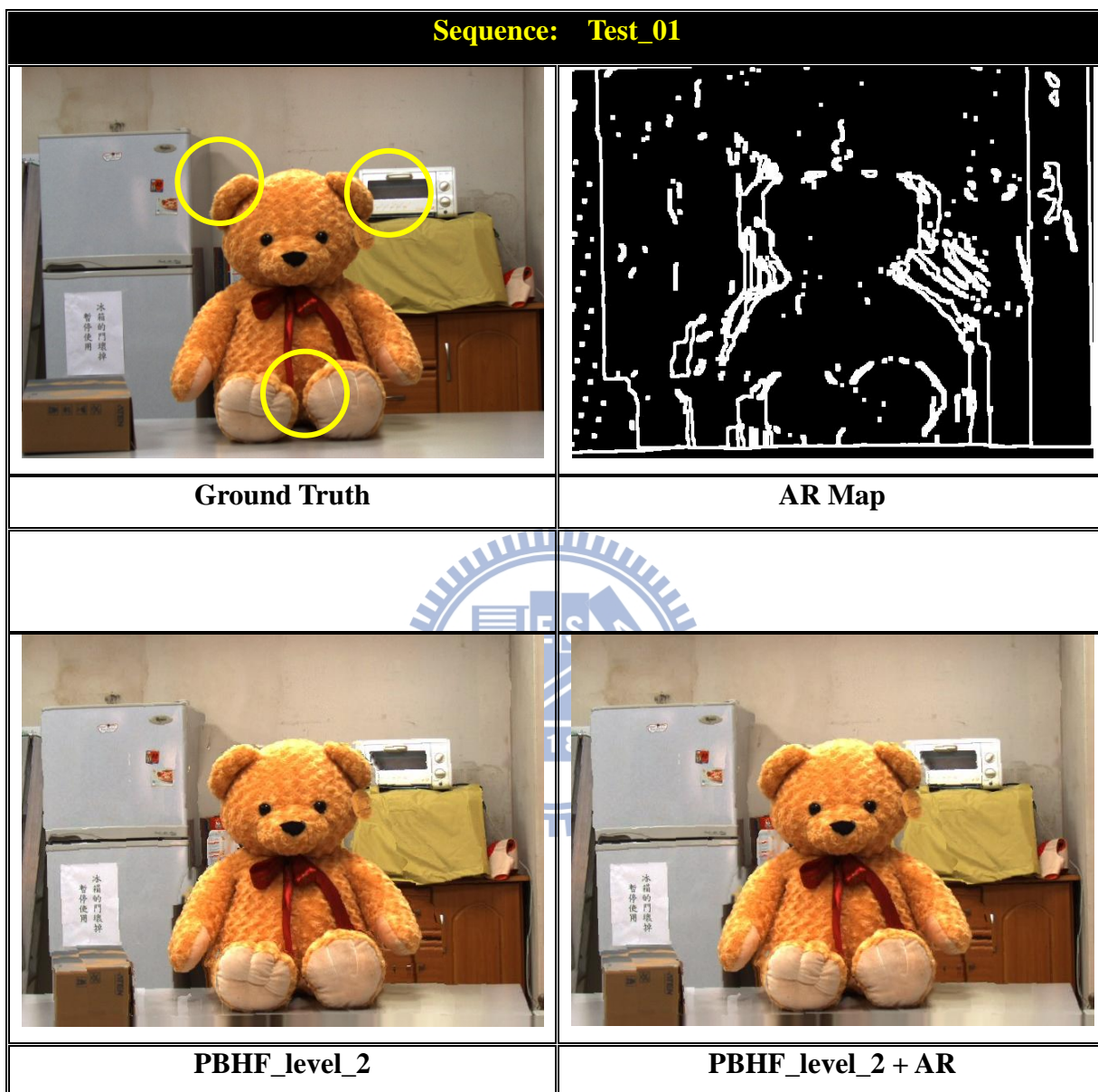


圖 85 虛擬視點合成針對瑕疵抑制方法之示意圖 (Test_01 Sequence)

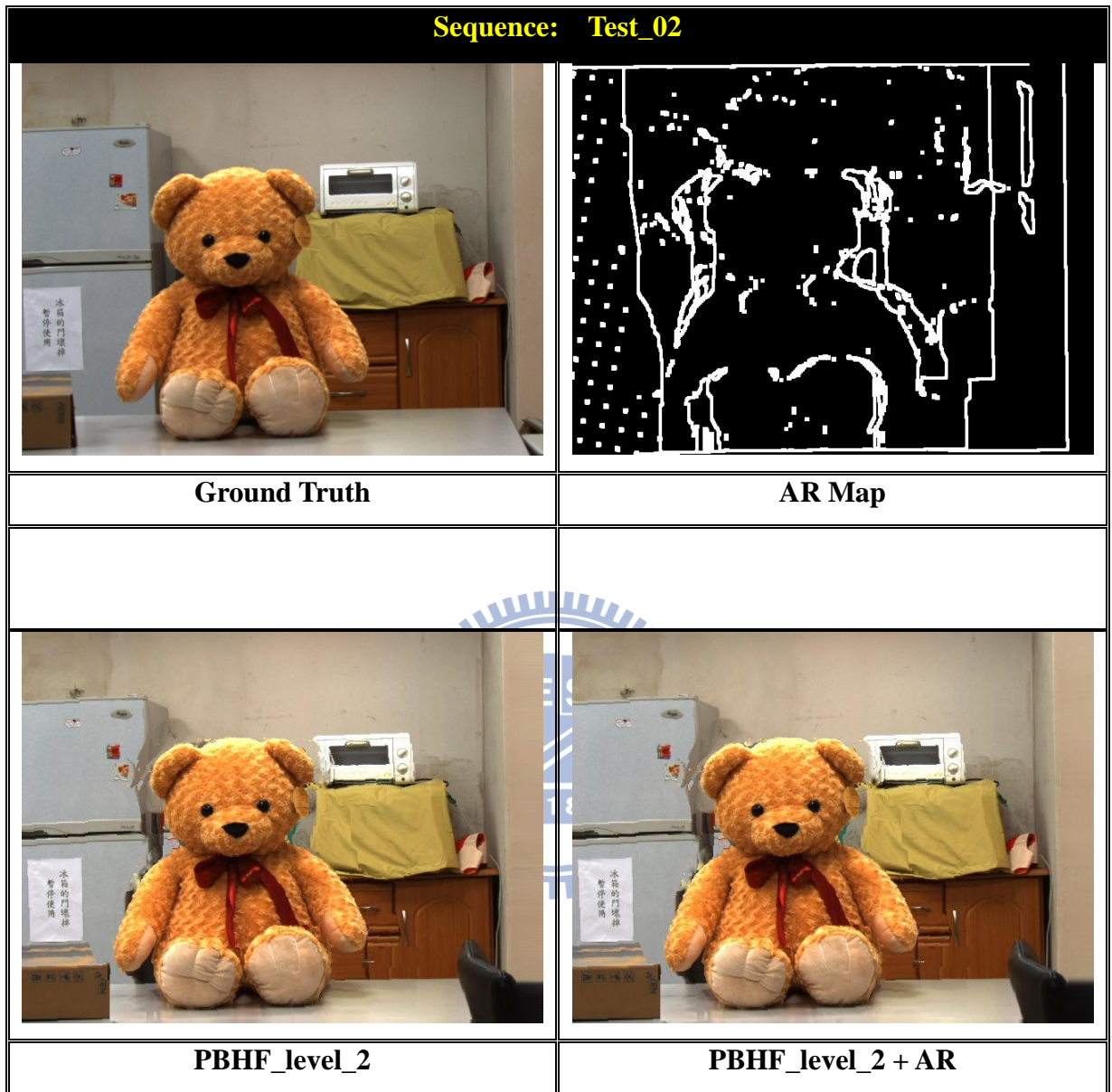


圖 86 虛擬視點合成針對瑕疵抑制方法之示意圖 (Test_02 Sequence)

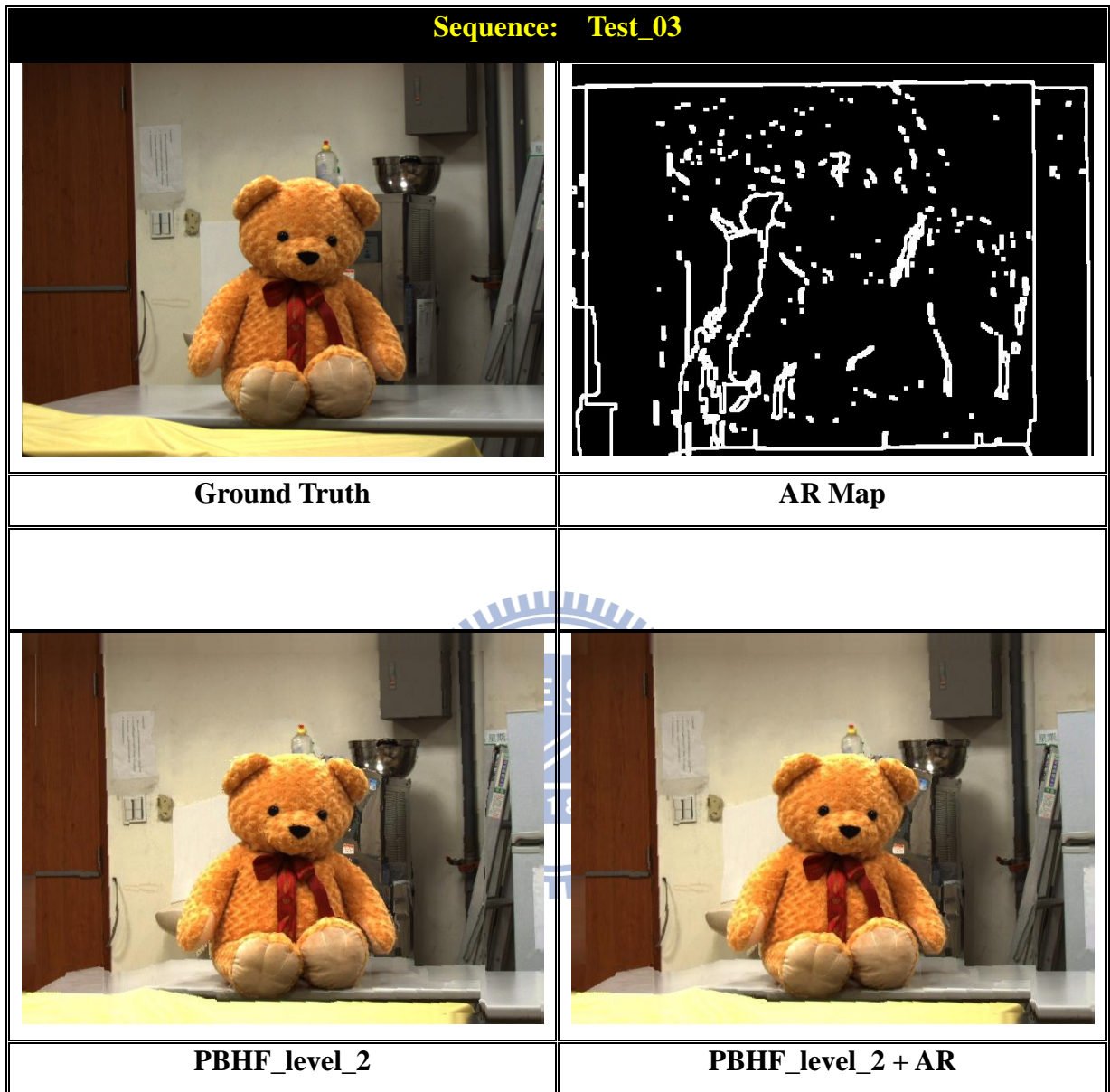


圖 87 虛擬視點合成針對瑕疵抑制方法之示意圖 (Test_03 Sequence)

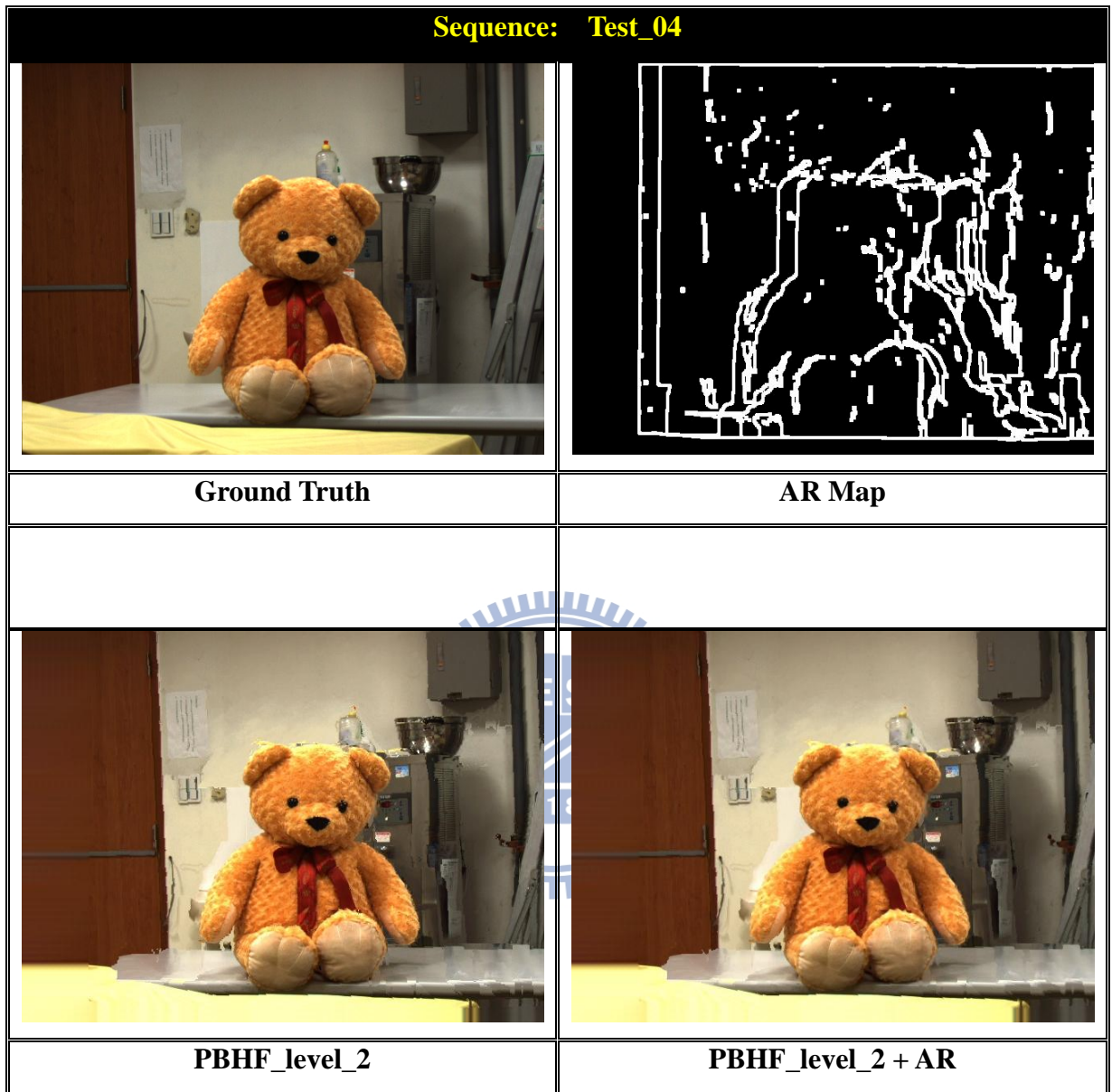


圖 88 虛擬視點合成針對瑕疵抑制方法之示意圖 (Test_04 Sequence)

5.8.3 實驗結果討論 (Discussions)

表 21 各類測試影像的虛擬視點合成針對瑕疵抑制有無之質量評估表 (CORR)

		Test_01	Test_02	Test_03	Test_04
Without AR	PBHF_level_0	0.81113	0.78061	0.67416	0.77188
	PBHF_level_1	0.80768	0.78151	0.67476	0.77402
	PBHF_level_2	0.80165	0.78081	0.6737	0.77541
With AR	PBHF_level_0	0.81224	0.78212	0.67498	0.77321
	PBHF_level_1	0.80879	0.78312	0.6756	0.77535
	PBHF_level_2	0.80275	0.78247	0.67453	0.77675
Unit: CORR					

由表 21 中可以看出對於使用瑕疵抑制演算法後的合成品質，其 CORR 皆有提高的效果，且不會因為金字塔型影像修補階層高的關係使得 CORR 數據降低，這意味著遮蔽區域的邊緣處存在很多雜訊，即使使用高階層的影像修補技術也還是會有瑕疵區域。

而在主觀品質上，如圖 89 所示（其放大區域來源在圖 85 中以圓圈表示）。可看出利用型態學的梯度算子，約略可抓出不可信賴的區域，而使用中值濾波器將這些瑕疵區域平滑掉。但這會有個問題，那就是計算梯度所使用的結構元素，其大小要如何決策。太大的話會造成影像模糊；太小的話瑕疵抑制的效果可能又不高。而在本實驗中，我們是採用半徑為 4 個像素大小的結構元素來算出梯度區域，因為對於真實影像合成品質來說，有需要太多考慮的因素，這些因素若沒納入考慮，其合成品質通常不高。因此我們會以主觀品質為優先考慮，而選擇較大範圍的結構元素來做瑕疵抑制。









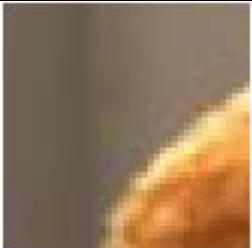



Ground Truth	PBHF_level_2	AR Map	PBHF_level_2 + AR
Region 1			
			
Region 2			
			
Region 3			
			

圖 89 Test_01 虛擬視點合成針對瑕疵抑制方法之部分放大示意圖

5.9 總結 (Summarize)

根據前面幾個小節的實驗數據討論，可統整出本論文所提出的稀疏多相機虛擬視點合成的架構與對應的最佳環境設定參數。對於已校正過的標準測試照片而言，使用本論文提出的景深映射後修復演算法、金字塔型影像修補技術、與後置瑕疵抑制演算法，在最後的合成影像上不論是質量評估或是主觀品質上皆有改善。其最佳架構如表 22 所示：

表 22 平行映射下的最佳稀疏虛擬視點合成系統參數一覽表

Parameters	Setting	Parameters	Setting
sampling_method	1	TW_blending_mode	1
depth_sampling_method	2	Pyramid_level	1~2
DW_direct_mode	0	AR_mode	1
DW_case_mode	1	thread_cont	1% ~ 3%
DW_refine_mode	2	thread_dest	10~20

而在廣義映射下的合成系統上，使用複本插值法於景深圖升取樣方法、金字塔型影像修補技術、與後置瑕疵抑制演算法，對於最後的合成影像，不論在質量評估或是主觀品質上皆有改善。至於逆向景深映射演算法，雖然在質量評估上不如正向景深映射演算法，但於主觀品質上卻有顯著的改善，並且在景深圖映射後的量化誤差也有很大的改進。其架構如表 23 所示：

表 23 廣義映射下的最佳稀疏虛擬視點合成系統參數一覽表

Parameters	Setting	Parameters	Setting
sampling_method	1	TW_blending_mode	0
depth_sampling_method	2	Pyramid_level	2
DW_direct_mode	1	AR_mode	1
DW_case_mode	0	thread_cont	nope
DW_refine_mode	1	thread_dest	nope

值得一提的是，若是針對標準測試影像做合成品質上的驗證，因需要考慮的成因不多，因此在影像疊合上會採用 2.7 章節所提出的 (23)式 (blending_on)，並且於金字塔影像修補技術中使用到階層 1 即可。而對於實拍的真實測試影像，需考慮的因素繁多，每一個合成步驟所隱含的雜訊與誤差累積起來會很可觀，因此在影像疊合上是採用 2.7 章節所提出的(25)式 (blending_off)，並於金字塔影像修補技術中使用到階層 2，以主觀品質為主要考量。

此外，對於本篇論文所提出的平行景深映射後修復演算法中所使用的可信賴模型，其連續圖門檻在 1% ~ 3%、下降圖門檻在 10~20 會有比較好的效果，並且對於合理的連續圖門檻下，下降圖門檻即使設定過大，也不會讓合成影像的 PSNR 整個降低許多。



Chapter 6 結論與未來工作 (Conclusions and Future Work)

6.1 結論 (Conclusions)

目前的自由視點合成系統上，皆只考慮平行且密集陣列的影像資訊，並使用以半人工的方式所產生的標準測試影像來驗證系統的優劣。本論文以稀疏非平行陣列為重點，根據目前現有的方法做改進，進而發展一套虛擬視點合成系統。此外，除了使用現有的標準測試影像與傳統的自由視點合成系統做比較外，並使用自己拍攝的真實影像來做系統驗證。

一開始會先以稀疏平行相機陣列所擷取的影像為主，分析景深圖映射後所產生的各種瑕疵現象，主要分為量化誤差和遮蔽區域；而量化誤差又被細分為型態 I 與型態 II。根據不同的瑕疵現象生成因素，我們提出一套景深映射後修復演算法，以可信賴模型來正確地修復量化誤差。效能上相對於使用(適應性)中值濾波器而言，於最後的合成品質上表現上 PSNR 增加約 0.2dB 左右，且主觀品質上也有改善。

接著我們以稀疏非平行相機陣列所擷取的影像為出發點，根據目前現有的合成方法做探討，並一共改進四個地方而提出對應的四種演算法，分別是景深圖升取樣方法、景深圖逆向映射演算法、金字塔型影像修補技術、瑕疵抑制方法。

景深圖升取樣方法上，我們會發現景深圖做升取樣時，使用複本插值法(Duplicated Interpolation) 的效果，比一般認知的高斯插值法(Gaussian Interpolation) 或雙三次插值法 (Bi-Cubic Interpolation) 的效果還要來的好，這意味著景深圖在插值法上需有另一套的分析模式，不能只單純使用於色彩影像上的插值法。而在合成後的主觀品質上，複本插值法會產生細小方塊狀的不連續瑕疵現象、高斯插值法會產生物體邊緣模糊化效果、雙三次插值法對於物體邊緣上則會有鋸齒狀的瑕疵現象。綜觀而言，複本插值法所產生的瑕疵現象，會少於其它兩者插值法。

而在景深圖映射演算法上，我們發展一套 N 張平面下的逆向景深圖映射演算法 (**Backward Depth Warping Algorithm Induced by N planes**，簡稱 **BDWIP- N**)，相較於現有的正向景深映射演算法而言，可大量降低映射後所產生的量化誤差，並且以**適應性中值濾波器**消除剩下的量化誤差區域。對於合成後的主觀品質上，不合理的瑕疵區域也會減少許多，尤其是物體邊緣處也會更加合理。但是 **BDWIP** 缺點在於計算量大，並且在影像邊緣處的遠景處容易判讀為遮蔽區域。

至於合成後的遮蔽區域，我們發展出**金字塔型影像修補技術 (Pyramid-based Hole Filling)**，簡稱 **PBHF**)，用以改善傳統的影像修補技術 (**Image In-Painting**) 在邊緣雜訊處無法抵抗的問題。而主觀品質上也可以觀察出 **PBHF** 在高階層下的遮蔽區域修補結果會比較自然，缺點是估計出來的區域會越模糊。

此外，本篇論文還提供**瑕疵抑制演算法 (Artifact Reduction)**，簡稱 **AR**)，利用形態學影像處理上的梯度算子來得到**可信賴模型**，並以中值濾波器來做瑕疵抑制。對於合成的主觀品質上，可以看出一些對於不可預期的瑕疵現象，在使用瑕疵抑制演算法後會被大幅降低。

以上稀疏非平行相機陣列所提出的四種演算法，我們以真實影像來做實測，並以 **CORR** 來做合成品質驗證。雖然 **CORR** 不是一個很好的質量評估方法，但是與主觀品質的趨勢大致上吻合。

6.2 未來工作 (Future Work)

一開始本篇論文有針對景深映射後會產生的各種瑕疵現象加以分析，並且提出對應的景深修復方法，該方法需要使用到**連續圖 (Continuity Map)**、**與下降圖 (Descending Map)** 來做為可信賴模型建構。不過這兩張二值化的影像資訊需要個別給定一個門檻值來當作連續遞減區域的判斷依據，或許可利用相機參數、與最近/遠景深資訊來推估合適的門檻值；甚至可對景深區域做分群，讓最後的量化誤差修復能更加精確。此外，也可以將此模型更加簡化，讓系統的實用性再提高。

而針對稀疏非平行相機陣列中所提出的四種演算法也有很多的改進空間。首先是景深圖升取樣的方法，因為跟色彩影像的紋理息息相關，所以若將色彩影像納入考量，對於景深圖的插值法會有很大的改善。第二個是逆向景深映射演算法，雖然可以大量降低量化誤差，但是計算時間長，需要想個方式將計算量壓低、或者在 GPU 上做平行處理的加速。而在廣義景深映射後的量化誤差修補，也需要想一套如可信賴模型的機制將之改進，這對最後的合成品質有很大的改善空間。

第三個是金字塔型影像修補技術，雖然立意點在於壓抑邊緣雜訊下做影像修補動作，但同時也犧牲掉遮蔽區域修補後的解析度；或許先將邊緣雜訊消除掉，在使用影像修補技術會是另外一個方向，但要準確判斷哪些區域屬於瑕疵現象、或是不自然雜訊等，會是一個很困難的工作。而最後的瑕疵抑制演算法，有點類似上述所說的瑕疵區域判斷，但本篇論文只有採用簡單的梯度區域來建構可信賴模型，或許可以考慮更複雜的模型來做改進。

此外，一些前置步驟如色彩校正、相機校正...等，若沒有處理得很完善，對於合成品質會有很大的影響，尤其是本篇論文著重於幾何上的相對關係做推論，並只考慮針孔成像模型，因此在合成上很容易受到數值運算等雜訊的影響；不僅如此，本篇論文也是假設景深估計做的完美的情況下來做後續的合成，因此錯誤的景深資訊很容易造成合成品質偏低，如本篇論文 Chapter 5 所做的實驗中，Test_03 因為相機校正不準確，間接影響到景深估計不準確，而影響到最後的合成階段，使得 CORR 數值偏低。若是真的要發展到應用層面、甚至於產品上，我們可能尚須結合主動式偵測景深設備，如 Kinect。

值得一提的是，逆向景深映射演算法中的量化階層 N 若越高，代表參考景深圖被成群切割地越細碎，這意味著此演算法有非常多的運算資源浪費於判斷二值化影像圖的資訊有無，因此利用插值法的概念，或許可先超估每塊平面對於虛擬試點而言的影響範圍大小，再針對此範圍做逆向單應映射。如此一來，逆向景深映射演算法的運算時間可大幅降低許多。

最後則是將此一系統實現於 Visual C、或其它平台，讓稀疏合成系統能更加具有及時性。



参考文献

- [1] P. Kauff, N. Atzpadin, C. Fehn, M. Muller, O. Schreer, A. Smolic, and R. Tanger, “Depth map creation and image-based rendering for advanced 3DTV services providing interoperability and scalability,” *Signal Process: Image Communication, Special Issue on 3DTV*, pp. 217-234, Feb. 2007.
- [2] 2007. A. Kubota, A. Smolic, M. Magnor, M. Tanimoto, T. Chen, and C. Zhang, “Multiview Imaging and 3DTV,” *IEEE Signal Processing Magazine*, vol. 24, no. 6, pp. 10–21, Nov. 2007.
- [3] A. Smolic and McCutchen, “3DAV Exploration of Video-Based Rendering Technology in MPEG,” *IEEE Trans. Circuits Syst. Video Technol., Special Issue on Immersive Commun.*, vol. 14, no. 9, pp. 348–356, March 2004.
- [4] M. Tanimoto, “Free Viewpoint Television (FTV),” *Digital Holography and Three-Dimensional Imaging (DH)*, Vancouver, Canada, June 18, 2007.
- [5] M. Tanimoto, “Overview of Free Viewpoint Television,” *Signal Processing: Image Communication*, vol. 21, no. 6, pp. 454–461, July 2006.
- [6] S. Zinger, L. Do, and P.H.N. de, “Free-Viewpoint Depth Image Based Rendering,” *Journal Visual Communication and Image Representation*, 21(5-6), pp. 533–541, 2010.
- [7] Z. Zhang, “A flexible new technique for camera calibration,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(11):1330-1334, 2000. Available together with the software at <http://research.microsoft.com/~zhang/Calib/>.
- [8] Y.-S. Kang and Y.-S. Ho, “An Efficient Image Rectification Method for Parallel Multi-Camera Arrangement,” *IEEE Trans. Consum. Electron.*, 57(3), pp. 1041–1048, 2011.

- [9] F. M. Porikli, "Inter-camera color calibration by correlation model function," *IEEE Int. Conf. on Image Processing*, pp. 133–136, 2003.
- [10] A. Olofsson, "Modern Stereo Correspondence Algorithms: Investigation and evaluation," Thesis from Dept. of Electrical Engineering, Linköping Univ., Linköping, Sweden, 2010.
- [11] D. Scharstein and R. Szeliski, "A taxonomy and evaluation of dense two-frame stereo correspondence algorithms," *Int. J. Computer Vision*, 2002. [Online]. Available: <http://www.middlebury.edu/stereo>
- [12] ISO/IEC JTC1/SC29/WG11, MPEG document N11631, "Report on Experimental Framework for 3D Video Coding," Oct. 2010.
- [13] K. J. Oh and Y. S. Ho, "Non-linear bi-directional prediction for depth coding," *Lect. Notes Comput. Sci.* 5879, pp. 522–531, 2009.
- [14] ISO/IEC JTC1/SC29/WG11, MPEG document M15672, "View synthesis software and assessment of its performance," July 2008.
- [15] J. R. Silva, T. T. Santos, and C. H. Morimoto, "Virtual Reality in Brazil: Automatic camera control in virtual environments augmented using multiple sparse videos," *Comput. and Graphics*, vol. 35, iss.2, April 2011.
- [16] R. Szeliski, "Image-Based Rendering," Computer Vision Algorithms and Applications, chap. 13, August 2010.
- [17] G. Chen, Y. Liu, and N. Max, "Real-time view synthesis from a sparse set of views," *Signal Process., Image Commun.*, 22, (2), pp. 188–202, 2007.
- [18] K. Müller, A. Smolic, K. Dix, P. Merkle, P. Kauff, and T. Wiegand, "View Synthesis for Advanced 3D Video Systems", *EURASIP Journal on Image and Video Processing, Special Issue on 3D Image and Video Processing*, vol. 2008, Article ID 438148, 11 pages, 2008.

- [19] R. Hartley and A. Zisserman, Multiple View Geometry in Computer Vision, 2nd ed., Cambridge University Press, 2004.
- [20] D. A. Forsyth and J. Ponce, Computer Vision A Modern Approach, 3rd ed., section 6.2.
- [21] Z. Zhang, "Flexible Camera Calibration By Viewing a Plane From Unknown Orientations," *Proc. ICCV*, pp. 666-673, 1999.
- [22] Camera Calibration Toolbox for Matlab. Available together with the software at http://www.vision.caltech.edu/bouguetj/calib_doc/
- [23] A. Fusiello, E. Trucco, A. Verri, "A compact algorithm for rectification of stereo pairs," *Machine Vision and Applications*, 12, pp. 16–22, 2000. Received: 25, Feb. 1999, Accepted: 2 March 2000.
- [24] J.-X. Chai, X. Tong, S.-C. Chan, and H.-Y. Shum, "Plenoptic Sampling," in *Proc. SIGGRAPH*, pp. 307–318, 2000.
- [25] Q. Chen and G. Medioni, "Image Synthesis from a Sparse Set of Views," *IEEE Visualization 97*, Phoenix, AZ, October 1997.
- [26] C. Lee and Y. S. Ho, "View Synthesis using Depth Map for 3D Video," *Asia-Pacific Signal and Information Processing Association*, pp. 350-357, Oct. 2009.
- [27] J. Wang and M. Cohen, "Image and video matting: A survey," *Foundations and Trends in Computer Graphics and Vision* 3, 2, pp. 97–175, 2007.
- [28] K.-J. Oh, S. Yea, and Y.-S. Ho, "Hole filling method using depth based in-painting for view synthesis in free viewpoint television and 3-D video," in *Proc. Picture Coding Symp.*, pp. 233–236, Piscataway, NJ, USA, May 2009.
- [29] Y. Mori, N. Fukushima, T. Yendo, T. Fujii, and M. Tanimoto, "View generation with 3D warping using depth information for FTV," *Signal Process. Image Commun.*, vol. 24, no. 1–2, pp. 65–72, Jan. 2009.
- [30] A. Telea, "An Image Inpainting Technique Based on the Fast Marching Method," *J.*

Graphics Tools, vol. 9, no. 1, pp. 23-34, 2004.

- [31] L. Yang, T. Yendo, M. P. Tehrani, T. Fujii, M. Tanimoto, "Artifact reduction using reliability reasoning for image generation of FTV," *Journal of Visual Communication on Image Representation*, vol.21, Issues 5-6, pp. 542-560, Sep., 2010.
- [32] M. Sjöström, P. Härdling, L. S. Karlsson, and R. Olsson, "Improved depth-image-based rendering algorithm," *3DTV Conference: The True Vision - Capture, Transmission and Display of 3D Video (3DTV-CON)*, 2011.
- [33] L. Doa, S. Zingera, and Peter H.N., "Warping error analysis and reduction for depth-image-based rendering in 3DTV," *Stereoscopic Displays and Applications XXII, Proceedings of the SPIE*, Volume 7863, pp. 78630B-78630B-9, 2011.
- [34] F. Devernay and A. R. Peon, "Novel view synthesis for stereoscopic cinema: detecting and removing artifacts," in *Proceedings of the 1st international workshop on 3D video processing, 3DVP '10*, 25-30, ACM, New York, NY, USA, 2010.
- [35] L. Wang, J. Liu, J. Sun, Y. Ren, W. Liu, and Y. Gao, "Virtual view synthesis without preprocessing depth image for depth image based rendering," *3DTV Conference: The True Vision - Capture, Transmission and Display of 3D Video (3DTV-CON)*, 2011.
- [36] R.C. Gonzalez and R. E. Woods, Digital Image Processing, 3rd ed., Prentice-Hall, Upper Saddle River, NJ, 2007.

自傳

李讀修，1988年3月5日出生於高雄市小港區。2006年6月於高雄市國立鳳新高級中學畢業，之後進入國立交通大學電子工程學系就讀。2010年6月畢業後繼續攻讀國立交通大學電子研究所，進入通訊電子暨訊號處理實驗室 (Communication Electronics and Signal Processing Lab，簡稱 CommLab)，承蒙杭學鳴教授的指導，主要研究領域為虛擬視點合成 (Virtual View Synthesis)，論文題目為「基於正逆向深度映射演算法下的稀疏多相機虛擬視點合成」 (Sparse Multi-Camera Virtual View Synthesis Using Forward and Backward Depth Warping Algorithms)。

