

基於三角形轉換映射的任意視點合成演 算法

研究生：李兆軒

指導教授：杭學鳴 博士

國立交通大學

電子工程學系 電子研究所碩士班



立體影像視覺顯影技術在多媒體通訊應用領域已經成為了非常熱門的研究主題，其中又以任意視點合成技術為最主要的關鍵議題。在近年來，有許多任意視點合成演算法已經被提出，通常都是利用多重影像以及其對應的深度資訊圖來產生虛擬視點的影像以達到任意視點的效果。

在本篇論文中，我們利用計算特徵對應點架構達到較佳的三角化歪斜映射後的結果。以及為了減少不自然的視覺感受，我們同時也利用多重頻段調合技術達到較好的影像品質。在計算特徵對應點架構中，我們計算影像亮度資訊以及相對應深度資訊圖上的梯度相關性來判定是否為特徵點。然而，三角化歪斜應射後的結果往往會在合成影像的物體邊緣處產生人造的不自然現象。因此，我們利用中

值濾波器以及多重頻段調合技術的概念以達到較佳的主觀視覺品質。當我們合成影像的過程中，多重頻段調合技術幫助我們對於物體邊緣處有柔和化的效果。

我們首先利用影像紋理針對前景深的部份映射至目標影像，我們稱之為 TBTW 演算法。比起傳統的畫素對畫素映射演算法，TBTW 演算法的優點在於可耗費較少的運算資源。然而，合成影像往往特別對於物體邊緣處都會產生瑕疵現象。因此我們採用相似的流程僅對於景深圖使用三角映射的方式，我們稱之為 TBDMW 演算法。對於紋理合成的部份我們仍然使用傳統的畫素對畫素映射方式。在本文獻的最後，我們也會針對我們提出的演算法顯示 MPEG 測試影像的合成結果，實驗結果顯示我們的演算法達到與 MPEG 所提供的 VSRS 類似的效果。



A Triangular-warping Based View Synthesis Algorithm for FTV

Student: Chao-Hsuan Li

Advisor: Dr. Hsueh-Ming Hang

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

Abstract

The free viewpoint video technology or free-viewpoint television (FTV) is an emerging research topic in multimedia processing. View synthesis is one key enabling technique of FTV. Several view synthesis schemes have been proposed in the past a few years. Typically, it uses multiple image views and depth maps to generate the intermediate view at nearly any arbitrary viewpoint.

In this study, we adopt the triangular warping technique for either texture mapping or depth map mapping. In the proposed scheme, we suggest a new feature point extraction process and include a multiband blending technique for image artifacts reduction. The new feature point extraction scheme makes the use of the correlation between the gradients of the image luminance intensity and the depth map. Sometimes, texture mapping using triangular warping may cause unnatural artifacts on the image object boundaries due to the discontinuity and the impairment of depth

map. Thus, the median filtering and the multi-band blending techniques are employed to achieve better subjective image quality. The multi-band blending scheme provides smooth transitions when mixing two (left and right) synthesized images.

We first apply the afore-described image warping method to the image texture synthesis. It is called Triangular-based Texture Warping (TBTW) algorithm. It requires much fewer calculations when compared with the traditional pixel-by-pixel warping methods. However, the synthesized images may contain artifacts in the complicated texture regions particularly along object boundaries. Then, we apply a similar data processing flow to the depth map warping and call it, Triangular-based Depth Maps Warping (TBDMW) algorithm. The texture synthesized part is using the traditional pixel-by-pixel warping. Often, with a simpler processing structure the proposed algorithm can provide an image quality comparable with that produced by the more complex MPEG reference software. In this study, we also investigate the effect of the triangular number and the multi-band blending technique.

誌謝

在交通大學的兩年研究生生涯中，首先我要感謝的是指導教授杭學鳴老師，在研究過程中，老師除了在專業學術研究領域上給予我指導外，更教導我對於做學問的方法以及求學的態度，老師也相當關心我們的生活近況。這兩年來，從老師身上不僅僅只學到專業知識，更學會到如何的待人處事與同儕之間相互合作的精神，我也堅信這對於我未來人生規劃有相當大的影響。

研究過程中，我首要感謝蔡彰哲學長的指導，從確定研究目標到完成實驗的過程中，我經歷了不少挫折，而蔡彰哲學長總是不斷鼓舞我並且參與討論給予意見；另一方面，我也非常感謝林鴻志學長的指導，不僅僅在專業科目上一直不斷給予我一些寶貴的意見，也給予我許多幫助，使得我能夠在碩班生涯成熟不少。除此之外，也非常感謝蔡家揚學長以及洪朝雄學長的幫忙，還有曉盈、威宇、郝婷、俊言、智凱、卓翰、書瑋、偵源、復凱、凱翔、婉瑜、怡茹、頌文以及學弟們的陪伴，讓我碩班生活更加精采豐富。

本篇論文得以完成要感謝我的家人以及依凡，在我求學生涯中不斷地鼓勵我。在此，僅將論文獻給曾經幫助過我、支持我的師長以及友人們，有你們的溫暖是我不斷前進的動力。

目錄

摘要	i
Abstract	iii
誌謝	v
目錄	vi
圖目錄	ix
表目錄	xv
第 1 章 緒論	1
1.1 研究背景 (Background)	1
1.2 研究動機及貢獻 (Motivation and Contributions)	2
1.3 論文大綱 (Organization of the Thesis)	3
第 2 章 相關工作	4
2.1 立體影像幾何概述 (Overview of Stereo Geometry)	4
2.2 針孔相機模型 (Pinhole Camera Model)	5
2.3 二維空間轉換映射 (2D Warping)	8
2.4 視點合成參考軟體介紹 (View Synthesis Reference Software)	12
2.4.1 軟體平台概述 (Overview of Software Platform)	12
2.4.2 品質測試指標 (Quality Metrics)	13
2.4.3 MPEG 測試影像介紹 (Introduction to MPEG Test Sequences)	14
第 3 章 傳統任意視點合成的演算法介紹	16
3.1 傳統視點合成演算法概述 (Overview)	16
3.2 前向轉換映射問題 (Forward Warping Problem)	16
3.2.1 物體邊緣處產生的瑕疵現象 (Artifacts on Object Boundary)	16
3.2.2 可信賴性模型 (Reliability Model)	19
3.2.3 三層互相連接之深度值的瑕疵現象分析 (Artifacts of Triple-layer Boundary)	
20	
3.3 使用可靠性的判別演算法設計 (Reliability Check Algorithm)	26
3.3.1 前向轉換映射 (Forward Warping)	26
3.3.2 線性比例疊合 (Linear Blending)	27
3.3.3 可靠性判別法則 (Reliable Check)	29
3.3.4 抑制瑕疵現象 (Artifacts Reduction)	29
第 4 章 基於三角化轉換映射可降低瑕疵現象的任意視點影像合成演算法設計	32
4.1 三角映射視點合成演算法概述 (Overview)	32

4.2 基於三角化映射的視點合成演算法(Triangular-based Warping View Synthesis Algorithm).....	33
4.2.1 擷取特徵點 (Extraction of Feature Points).....	33
4.2.2 特徵點三角化 (Delaunay Triangulation).....	35
4.2.3 三角化紋理投影映射 (Texture Mapping of Triangular).....	35
4.2.4 使用後製濾波器修復裂痕 (Gap Reduction by Post-filtering).....	37
4.2.5 利用多率頻段疊合抑制瑕疵現象 (Artifacts Reduction by Using Multi-band Blending)	38
4.3 模擬與結果 (Simulation and Results).....	41
4.3.1 實驗環境設定	41
4.3.2 實驗模擬結果	42
4.3.2.1 實驗第一部份：MPEG 測試影像提供的景深圖對於合成結果的表現.....	42
4.3.2.2 實驗第二部份：景深圖的差異性對於合成結果的表現.....	58
4.3.2.3 實驗第三部份：攝影機位置的差異性對於合成結果的表現.....	73
第 5 章 基於三角形映射的景深圖資訊的任意視點合成演算法設計.....	89
5.1 概述 (Overview)	89
5.2 傳統的任意視點合成演算法介紹 (Introduction to Conventional View Synthesis Algorithm).....	89
5.2.1 利用畫素對畫素投影映射的視點合成演算法 (View Synthesis Algorithm by Using Pixel-by-pixel Warping).....	89
5.2.2 利用三角形對三角形投影映射的視點合成演算法 (View Synthesis Algorithm by Using Triangle-by-triangle Warping)	90
5.3 基於三角映射的景深圖資訊的視點合成演算法 (View Synthesis Algorithm Based on Triangular Warped Depth Maps)	92
5.3.1 針對景深圖擷取特徵點 (Extraction of Feature Points for Depth Maps).....	93
5.3.2 對於景深圖的特徵點三角化 (Delaunay Triangulation for Depth Maps).....	93
5.3.3 景深圖經由三角形投影映射至目標影像位置 (Depth Maps Projection with Triangular Warping)	94
5.3.4 針對映射後的景深圖進行後續濾波器處理 (Post-filtering on the Projected Depth Maps)	95
5.3.5 利用景深資訊將參考影像映射至虛擬視點畫面 (Froward Warping to the Virtual View Based on Depth Maps).....	96
5.3.6 左右兩張映射後的影像進行線性疊合 (Linear Blending of The Two Warped View)	97
5.3.7 影像邊緣擴張與影像修補 (Boundary Matting and Image Inpainting)	98
5.4 模擬與討論 (Simulation and Results).....	101
5.4.1 模擬結果	101
5.4.1.1 實驗第一部份：MPEG 測試影像提供的景深圖對於合成結果的表現.....	103

5.4.1.2 實驗第二部份：特徵點個數對於合成結果的影響與 PSNR 表現.....	116
第 6 章 結論與未來工作.....	132
6.1 結論.....	132
6.2 未來工作.....	133
參考文獻.....	134
自傳.....	137



圖目錄

圖 2-1 立體影像幾何關係	4
圖 2-2 相機針孔模型示意圖	5
圖 2-3 相機外部參數轉換關係示意圖	7
圖 2-4 目標影像與來源影像之間的三維轉換映射關係	8
圖 2-5 影像空間對應光影射線關係	9
圖 2-6 三維空間中的一齊次點座標與不同影像平面的對應關係示意圖	9
圖 2-7 兩相機點座標與深度值的對應關係示意圖	11
圖 2-8 VSRS 2.0 的程式設計平台設計架構	12
圖 3-1 FB Pattern	17
圖 3-2 BF Pattern	17
圖 3-3 FBF 表示示意圖	21
圖 3-4 不同的瑕疵現象情況分析表示示意圖：(a) Case 2-FBLF, (b) Case 4-FBRB, (c) Case 6-BFLB, and (d) Case 8-BFRF	22
圖 3-5 三層不同深度值的正確映射方向表示示意圖	23
圖 3-6 FB 的情況下可能產生的瑕疵現象映射過程示意圖	25
圖 3-7 FB 的情況下可能產生的瑕疵現象映射過程示意圖	25
圖 3-8 Yang 的演算法設計架構流程圖[16]	26
圖 3-9 合成影像由 (a)左邊 (b)右邊 參考視點所產生	27
圖 3-10 待合成攝影機位置與左右兩邊的參考攝影機距離示意圖	28
圖 3-11 依照左邊參考影像所產生的可依賴性二值化判別圖	30
圖 3-12 Yang et al.[16]合成後產生破洞的影像結果(紅色區域)	31
圖 3-13 Yang et al.[16]最後影像合成結果	31
圖 4-1 利用 Delaunay 三角化的方式將特徵點連結：測試影像 “Champagne Tower” (a)視點 39 (特徵點數目：1496) (b)視點 41 (特徵點數目：1658)	35
圖 4-2 在 Z_{near} 與 Z_{far} 兩個夾擊平面中預測最佳化景深量化值示意圖[12]	36
圖 4-3 測試影像 “Champagne Tower”：(a) 由視點 39 所合成的虛擬視點 40 的目標影像 (b) 由視點 41 所合成的虛擬視點 40 的目標影像 (c) 視點 39 合成虛擬視點 40 所造成的遮蔽區域 (d) 視點 41 合成虛擬視點 40 所造成的遮蔽區域	37
圖 4-4 經過中值濾波法修復細小的遮蔽區域	38
圖 4-5 擁有三個不同子頻帶的多率頻段疊合範例。其中 $k = 1, 2, 3$ 的 $B_{k\sigma}$ 如同左邊影像所示，右邊影像代表相對應被遮蔽區域的權重函數	40
圖 4-6 測試影像 “Newspaper” 由視點 04 與視點 06 所合成的虛擬視點 05 結果比較圖	44
圖 4-7 測試影像 “Newspaper” 由視點 04 與視點 06 所合成的虛擬視點 05 細部結果放大比較圖	45

圖 4-8 不同的視點合成演算法在測試影像 “Newspaper” 視點 05 的前 100 張影像 PSNR 表現.....	46
圖 4-9 不同的視點合成演算法針對測試影像 “Newspaper” 視點 05 的前 100 張影像上的平均 PSNR 表現.....	46
圖 4-10 測試影像 “Kendo” 由視點 03 與視點 05 所合成的虛擬視點 04 結果比較圖.....	47
圖 4-11 測試影像 “Kendo” 由視點 03 與視點 05 所合成的虛擬視點 04 細部放大結果比較圖.....	48
圖 4-12 不同的視點合成演算法在測試影像 “Kendo” 視點 04 的前 100 張影像 PSNR 表現.....	49
圖 4-13 不同的視點合成演算法針對測試影像 “Kendo” 視點 04 的前 100 張影像上的平均 PSNR 表現.....	49
圖 4-14 測試影像 “Champagne tower” 由視點 39 與視點 41 所合成的虛擬視點 40 結果比較圖.....	50
圖 4-15 測試影像 “Champagne tower” 由視點 39 與視點 41 所合成的虛擬視點 40 細部放大結果比較圖.....	51
圖 4-16 不同的視點合成演算法在測試影像 “Champagne tower” 視點 40 的前 100 張影像 PSNR 表現.....	52
圖 4-17 不同的視點合成演算法針對測試影像 “Champagne tower” 視點 40 的前 100 張影像上的平均 PSNR 表現.....	52
圖 4-18 測試影像 “Book arrival” 由視點 08 與視點 10 所合成的虛擬視點 09 結果比較圖.....	53
圖 4-19 測試影像 “Book arrival” 由視點 08 與視點 10 所合成的虛擬視點 09 細部放大結果比較圖.....	54
圖 4-20 不同的視點合成演算法在測試影像 “Book arrival” 視點 09 的前 100 張影像 PSNR 表現.....	55
圖 4-21 不同的視點合成演算法針對測試影像 “Book arrival” 視點 09 的前 100 張影像上的平均 PSNR 表現.....	55
圖 4-22 測試影像 “Newspaper” 利用 DERS 4.9 產生的景深圖由視點 04 與視點 06 所合成的虛擬視點 05 結果比較圖.....	59
圖 4-23 測試影像 “Newspaper” 利用 DERS 4.9 產生的景深圖由視點 04 與視點 06 所合成的虛擬視點 05 細部結果放大比較圖.....	60
圖 4-24 DERS 4.9 產生的景深圖對於視點合成演算法在測試影像 “Newspaper” 視點 05 的前 30 張影像 PSNR 表現.....	61
圖 4-25 DERS 4.9 產生的景深圖對於視點視點合成演算法針對測試影像 “Newspaper” 視點 05 的前 30 張影像上的平均 PSNR 表現.....	61
圖 4-26 測試影像 “Kendo” 利用 DERS 4.9 產生的景深圖由視點 03 與視點 05 所合成的虛擬視點 04 結果比較圖.....	62

圖 4-27 測試影像 “Kendo” 利用 DERS 4.9 產生的景深圖由視點 03 與視點 05 所合成的虛擬視點 04 細部放大結果比較圖.....	63
圖 4-28 DERS 4.9 產生的景深圖對於視點視點合成演算法在測試影像 “Kendo” 視點 04 的前 30 張影像 PSNR 表現.....	64
圖 4-29 DERS 4.9 產生的景深圖對於視點視點合成演算法針對測試影像 “Kendo” 視點 04 的前 30 張影像上的平均 PSNR 表現.....	64
圖 4-30 測試影像 “Champagne tower” 利用 DERS 4.9 產生的景深圖由視點 39 與視點 41 所合成的虛擬視點 40 結果比較圖.....	65
圖 4-31 測試影像 “Champagne tower” 利用 DERS 4.9 產生的景深圖由視點 39 與視點 41 所合成的虛擬視點 40 細部放大結果比較圖.....	66
圖 4-32 DERS 4.9 產生的景深圖對於視點視點合成演算法在測試影像 “Champagne tower” 視點 40 的前 30 張影像 PSNR 表現	67
圖 4-33 DERS 4.9 產生的景深圖對於視點視點合成演算法針對測試影像 Champagne tower” “視點 40 的前 30 張影像上的平均 PSNR 表現	67
圖 4-34 測試影像 “Book arrival” 利用 DERS 4.9 產生的景深圖由視點 08 與視點 10 所合成的虛擬視點 09 結果比較圖.....	68
圖 4-35 測試影像 “Book arrival” 利用 DERS 4.9 產生的景深圖由視點 08 與視點 10 所合成的虛擬視點 09 細部放大結果比較圖.....	69
圖 4-36 DERS 4.9 產生的景深圖對於視點視點合成演算法在測試影像 “Book arrival” 視點 09 的前 30 張影像 PSNR 表現.....	70
圖 4-37 DERS 4.9 產生的景深圖對於視點視點合成演算法針對測試影像 “Book arrival” 視點 09 的前 30 張影像上的平均 PSNR 表現.....	70
圖 4-38 測試影像 “Newspaper” 利用 DERS 4.9 產生的景深圖由視點 04 與視點 06 所合成的虛擬視點 05 結果比較圖.....	74
圖 4-39 測試影像 “Newspaper” 利用 DERS 4.9 產生的景深圖由視點 04 與視點 06 所合成的虛擬視點 05 細部結果放大比較圖.....	75
圖 4-40 DERS 4.9 產生視點 04 與視點 06 的景深圖對於視點合成演算法在測試影像 “Newspaper” 視點 05 的前 30 張影像 PSNR 表現.....	76
圖 4-41 DERS 4.9 產生視點 04 與視點 06 的景深圖對於視點視點合成演算法針對測試影像 “Newspaper” 視點 05 的前 30 張影像上的平均 PSNR 表現.....	76
圖 4-42 測試影像 “Kendo” 利用 DERS 4.9 產生的景深圖由視點 02 與視點 05 所合成的虛擬視點 04 結果比較圖.....	77
圖 4-43 測試影像 “Kendo” 利用 DERS 4.9 產生的景深圖由視點 02 與視點 05 所合成的虛擬視點 04 細部放大結果比較圖.....	78
圖 4-44 DERS 4.9 產生視點 02 與視點 05 的景深圖對於視點視點合成演算法在測試影像 “Kendo” 視點 04 的前 30 張影像 PSNR 表現.....	79
圖 4-45 DERS 4.9 產生視點 02 與視點 05 的景深圖對於視點視點合成演算法針對測試影像 “Kendo” 視點 04 的前 30 張影像上的平均 PSNR 表現.....	79

圖 4-46 測試影像 “Champagne tower” 利用 DERS 4.9 產生的景深圖由視點 38 與視點 42 所合成的虛擬視點 40 結果比較圖.....	80
圖 4-47 測試影像 “Champagne tower” 利用 DERS 4.9 產生的景深圖由視點 38 與視點 42 所合成的虛擬視點 40 細部放大結果比較圖.....	81
圖 4-48 DERS 4.9 產生視點 38 與視點 42 的景深圖對於視點視點合成演算法在測試影像 “Champagne tower” 視點 40 的前 30 張影像 PSNR 表現	82
圖 4-49 DERS 4.9 產生視點 38 與視點 42 的景深圖對於視點視點合成演算法針對測試影像 “Champagne tower” 視點 40 的前 30 張影像上的平均 PSNR 表現.....	82
圖 4-50 測試影像 “Book arrival” 利用 DERS 4.9 產生的景深圖由視點 08 與視點 11 所合成的虛擬視點 09 結果比較圖.....	83
圖 4-51 測試影像 “Book arrival” 利用 DERS 4.9 產生的景深圖由視點 08 與視點 11 所合成的虛擬視點 09 細部放大結果比較圖.....	84
圖 4-52 DERS 4.9 產生視點 08 與視點 11 的景深圖對於視點視點合成演算法在測試影像 “Book arrival” 視點 09 的前 30 張影像 PSNR 表現	85
圖 4-53 DERS 4.9 產生視點 08 與視點 11 的景深圖對於視點視點合成演算法針對測試影像 “Book arrival” 視點 09 的前 30 張影像上的平均 PSNR 表現	85
圖 5-1 畫素對畫素投影映射的視點合成演算法架構設計	90
圖 5-2 三角形對三角形投影映射的視點合成演算法設計	91
圖 5-3 基於三角映射的景深圖資訊的視點合成演算法介紹	93
圖 5-4 利用 Delaunay 三角化重建三角平面模型：測試影像 “Newspaper” (特徵點數目：5656).....	94
圖 5-5 MPEG 測試影像 “Newspaper”經過三角映射後的景深圖，其中(a)由視點 04 經三角映射後的視點 05 的景深圖(b)由視點 06 經三角映射後的視點 05 的景深圖.....	95
圖 5-6 MPEG 測試影像 “Newspaper”經過後續濾波器處理後的景深圖，其中(a)由視點 04 三角映射後的視點 05 景深圖經過後續濾波器的處理結果，(b)由視點 06 三角映射後的視點 05 景深圖經過後續濾波器的處理結果....	96
圖 5-7 MPEG 測試影像 “Newspaper”由鄰近兩個攝影機視點位置正向投影映射後的結果，其中(a)由視點 04 投影映射至視點 05 的目標影像結果以及(b)由視點 06 投影映射至視點 05 的目標影像結果.....	97
圖 5-8 MPEG 測試影像 “Newspaper”經由影像擴張處理之後的結果圖.....	99
圖 5-9 待修補可靠度區域的梯度相關值示意圖	99
圖 5-10 影像修補的步驟示意圖	100
圖 5-11 未經過影像修補前的合成影像結果以及待修補的區域	101
圖 5-12 經過影像修補後的合成影像結果	101
圖 5-13 測試影像 “Newspaper”由視點 04 與視點 06 所合成的虛擬視點 05 結果	

比較圖.....	103
圖 5-14 測試影像 “Newspaper”由視點 04 與視點 06 所合成的虛擬視點 05 細部 結果放大比較圖.....	104
圖 5-15 不同的視點合成演算法在測試影像 “Newspaper” 視點 05 的前 30 張影 像 PSNR 表現.....	105
圖 5-16 不同的視點合成演算法針對測試影像 “Newspaper” 視點 05 的前 30 張 影像上的平均 PSNR 表現.....	105
圖 5-17 測試影像 “Kendo”由視點 03 與視點 05 所合成的虛擬視點 04 結果比較 圖.....	106
圖 5-18 測試影像 “Kendo”由視點 03 與視點 05 所合成的虛擬視點 04 細部結果 放大比較圖.....	107
圖 5-19 不同的視點合成演算法在測試影像 “Kendo” 視點 04 的前 30 張影像 PSNR 表現.....	108
圖 5-20 不同的視點合成演算法針對測試影像 “Kendo” 視點 04 的前 30 張影像 上的平均 PSNR 表現.....	108
圖 5-21 測試影像 “Champagne tower”由視點 39 與視點 41 所合成的虛擬視點 40 結果比較圖.....	109
圖 5-22 測試影像 “Champagne tower”由視點 39 與視點 41 所合成的虛擬視點 40 細部結果放大比較圖.....	110
圖 5-23 不同的視點合成演算法在測試影像 “Champagne tower” 視點 40 的前 30 張影像 PSNR 表現.....	111
圖 5-24 不同的視點合成演算法針對測試影像 “Champagne tower” 視點 40 的 前 30 張影像上的平均 PSNR 表現.....	111
圖 5-25 測試影像 “Book arrival”由視點 08 與視點 10 所合成的虛擬視點 09 結果 比較圖.....	112
圖 5-26 測試影像 “Book arrival”由視點 08 與視點 10 所合成的虛擬視點 09 細部 結果放大比較圖.....	113
圖 5-27 不同的視點合成演算法在測試影像 “Book arrival” 視點 10 的前 30 張 影像 PSNR 表現.....	114
圖 5-28 不同的視點合成演算法針對測試影像 “Book arrival” 視點 10 的前 30 張影像上的平均 PSNR 表現.....	114
圖 5-29 TBTW 演算法在不同特徵點的數目下對於 MPEG 測試影像 “Newspaper” 的視點 05 合成目標細部放大圖.....	118
圖 5-30 TBDMW 演算法在不同特徵點的數目下對於 MPEG 測試影像 “Newspaper” 的視點 05 合成目標細部放大圖.....	119
圖 5-31 TBTW 演算法在不同特徵點的數目下對於 MPEG 測試影像 “Kendo” 的視點 04 合成目標細部放大圖.....	121
圖 5-32 TBDMW 演算法在不同特徵點的數目下對於 MPEG 測試影像 “Kendo”	

的視點 04 合成目標細部放大圖.....	122
圖 5-33 TBTW 演算法在不同特徵點的數目下對於 MPEG 測試影像 “Champagne tower” 的視點 40 合成目標細部放大圖	124
圖 5-34 TBDMW 演算法在不同特徵點的數目下對於 MPEG 測試影像 “Champagne tower” 的視點 40 合成目標細部放大圖	125
圖 5-35 TBTW 演算法在不同特徵點的數目下對於 MPEG 測試影像 “Book arrival” 的視點 09 合成目標細部放大圖	127
圖 5-36 TBDMW 演算法在不同特徵點的數目下對於 MPEG 測試影像 “Book arrival” 的視點 09 合成目標細部放大圖	128



表目錄

表 1 MPEG 測試影像介紹表	15
表 2 造成合成瑕疵現象的情況分析與深度值模式.....	18
表 3 三層不同深度值邊緣表示法.....	23
表 4 三層不同深度值可能造成的瑕疵現象分析表.....	24
表 5 VSRS 2.0 環境參數設定.....	41
表 6 MPEG 測試影像以及對應景深圖	43
表 7 MPEG 測試影像前 100 張合成影像的平均 PSNR 效能表現	57
表 8 DERS 4.9 產生的景深圖以及對應的視點位置	58
表 9 MPEG 測試影像 “Book arrival”對於不同的景深圖合成結果比較表.....	71
表 10 MPEG 測試影像針對前 30 張合成影像的平均 PSNR 效能表現	72
表 11 DERS 4.9 產生的景深圖以及新對應的視點位置	73
表 12 MPEG 測試影像提供的景深圖對於前 30 張合成影像的平均 PSNR 效能表 現.....	86
表 13 DERS 4.9 產生的景深圖對於前 30 張合成影像的平均 PSNR 效能表現...87	
表 14 MPEG 測試影像前 30 張合成影像的平均 PSNR 效能表現	115
表 15 MPEG 測試影像 “Book arrival”對於不同演算法的細部放大結果比較圖	116
表 16 演算法在不同特徵點的個數下對於 MPEG 測試影像 “Newspaper” 的 PSNR 曲線變化圖.....	117
表 17 演算法在不同特徵點的個數下對於 MPEG 測試影像 “Kendo” 的 PSNR 曲線變化圖.....	120
表 18 演算法在不同特徵點的個數下對於 MPEG 測試影像 “Champagne tower” 的 PSNR 曲線變化圖.....	123
表 19 演算法在不同特徵點的個數下對於 MPEG 測試影像 “Book arrival” 的 PSNR 曲線變化圖.....	126
表 20 演算法在不同特徵點個數對於 MPEG 測試影像的 PSNR 表現	129
表 21 特徵點個數於演算法的運行時間比較表.....	130

第1章 緒論

1.1 研究背景 (Background)

在近年來，隨著硬體以及相關演算法的進步，同時伴隨著三維影像顯影技術的成熟，自由視點顯影成像技術(Free Viewpoint Television)已經成為多媒體研究領域上熱門的研究題目[18]。自由視點顯影成像技術主要目的是能夠提供使用者使用有限的影像設備來精確合成三維空間中任意視點的場景資訊，並且能夠有效的壓縮所記錄的資訊，以達到最小的傳輸量。

MPEG 國際標準會議(ISO/IEC Moving Picture Expert Group)在近年來也正式制定了3DAV(3D Audio-Video)標準文件，其目標是推廣現有的視訊標準，其中自由視點顯影成像系統是一個新的互動式立體視訊(Interactive Video System)[21]。

自由視點顯影成像系統是一種建立影像場景成像(Image-based Rendering)的方法。一般而言，建立影像場景成像是利用多台不同位置的攝影機擷取影像，並且依據光學原理內插出不同視點所見的畫面，自由視點顯影成像系統包含了下列功能單元，分別是擷取多重影像(Multi-view Video Capture)、影像校正(Image Correction)[2][3]、場景景深預估(Depth Estimation)[1]、多重視點壓縮技術(Multi-view Video Codec)以及視點合成(View Synthesis)[16]。

在極線幾何(Epipolar Geometry)的估算之下，我們在一個攝影機陣列中，可以利用極線幾何的特性產生三維擷取影像場景的景深圖(Depth Map)，目前為止預測景深圖的演算法已經發展了好幾種，其中使用能量最小化(Energy minimization)的演算法預測的景深圖是最成功[1]，但是景深圖仍有一些邊緣部分不自然的現象有待改善，影像壓縮技術已被廣泛的應用在多媒體通訊上，我們可以利用多重視點壓縮編碼與解碼技術有效減少三維場景資訊的傳輸量與儲存量，其中視點合成對於自由視點顯影成像技術(FTV)是一個重要的技術，視點合成在進行內插影像時，為了增加其合成影像品質我們會搭配景深圖當作參考，在近年

來，已經有許多研究者投入視點合成的技術，縱使預測的景深圖邊緣處並非完美，但是我們希望能夠利用視點合成的演算法設計來改善合成出來的影像品質，在本篇文獻中，我們也會專注於視點合成的演算法設計部分。

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

視點合成演算法可以分成兩大部分，第一個部分為紋理映射(Texture mapping)，第二個部分為降低瑕疵現象(Artifact reduction)，在本篇文獻中，我們計算出梯度相關值來擷取新的特徵點，我們將這些特徵點利用 Delaunay 三角化的方法連結起來，利用三角化大面積的型式去做紋理映射到目標影像的位置上。在降低瑕疵現象(Artifacts)的部分，合成影像的邊緣處有可能因為曝光等原因，我們可以使用多頻段混合(Multi-band Blending)的方式混合影像中的邊緣區域使得細節較清晰，得到品質較佳的合成影像。

傳統的自由視點顯影成像系統大多都是使用排列密集的一維攝影機陣列來擷取影像，自由視點顯影成像在進行影像合成顯影成像時，在多數的研究中，通常紋理映射的部分大都是使用畫素對畫素的方式進行映射(pixel-by-pixel warping)。然而，在本篇論文中我們使用的紋理映射方式是使用三角形對三角形的映射方式(Region-by-region warping)，如果可以精確的建立三角化模型的話，三角形映射方式的確會得到一個較佳的合成影像品質。然而，所以事先選擇特徵點的步驟扮演著一個相當重要的角色，因為選取特徵點的好壞會影響三角形映射的目標影像品質，在本篇論文當中，我們也提出了一個有效的演算法架構並且獲取好的特徵點。

另外，目前有許多研究也針對於合成影像的邊緣處產生不自然的人造現象(Artifacts)提出許多不同的演算法以改善合成影像的品質。在本篇論文中，我們提出了使用多頻段調和(Multi-band Blending)的方式混合影像中的邊緣區域使得細節較清晰，得到品質較佳的合成影像，我們會在第四章之後解釋。

1.3 論文大綱 (Organization of the Thesis)

在本篇論文中，第二章將會簡單介紹極平面圖像與三維空間中的場景彼此之間的關係，並利用該關係來合成虛擬視點影像。在第三章中，我們將會介紹傳統的視點合成演算法設計流程以及前向映射可能會導致的錯誤問題。在本篇論文第四章中，我們將會針對另外的前向映射方法提出新的演算法架構設計，並且加以分析以及討論其合成結果的表現。在本篇論文的第五章，我們將結合傳統的視點合成演算法與基於三角形映射的演算法的優點，成為一種新的視點合成演算法設計架構，並且得到品質較佳的合成影像實驗結果。最後，第六章對本論文的未來研究方向做個簡短的說明以及總結。



第2章 相關工作

2.1 立體影像幾何概述 (Overview of Stereo Geometry)

當相機未校正的情況下，我們可以透過極線幾何特性的限制關係獲得相機內、外部參數的求取，如圖 2-1 所示。表示一個三維空間中典型的立體影像系統示意圖。我們使用兩個針孔成像攝影機來擷取影像資料，並將成像平面置於鏡頭前方。兩台攝影機之間的連線我們稱之為基線(Baseline)，其中基線和成像平面的交錯處稱之為極點(Epipole)，而空間中任一點 P 和兩台攝影機的基線所組成的平面稱為極平面(Epipolar plane)，極平面在成像平面上的相交線稱為極線(Epipolar line)，在極平面上的每一點都會投影到與其對應的極線上的點。

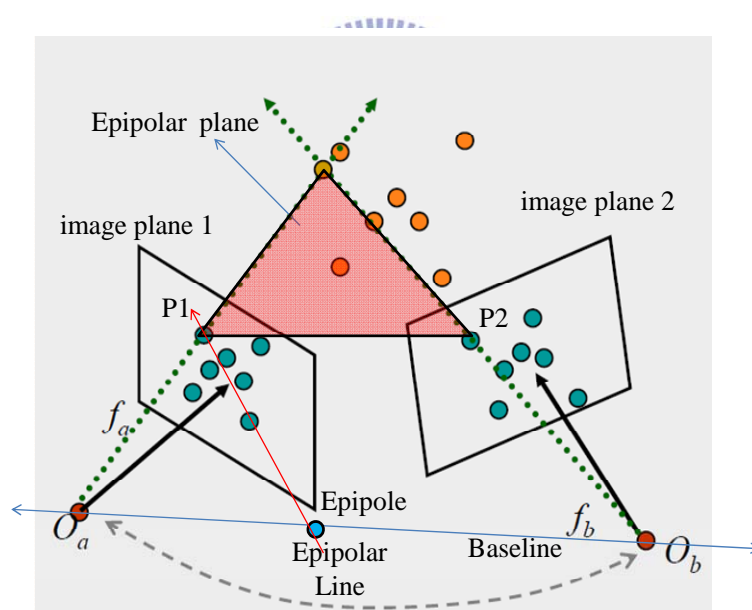


圖 2-1 立體影像幾何關係

由於在極平面上的各點都會投影成每張影像上得到一條極線，因此對應到同一張極平面上的兩條極線，其線上各點也都存在著幾何對應關係。當我們在兩張影像上搜尋互相對應的點時，可藉由此特性將原本的搜尋範圍從在平面上的搜尋轉換為僅在極線上的搜尋，可以針對極線上的對應點搜尋藉以大幅減少搜尋範圍，

在兩張以上的影像之間搜尋一給定的對應點時，我們只需要計算一次極平面即可，這項特性稱之為極線限制(Epipolar constraint)，更詳細的極線幾何定理介紹可以參考[25]。

2.2 針孔相機模型 (Pinhole Camera Model)

近年來電腦視覺的蓬勃發展，相機校正的應用也越來越廣泛[2][3]，在相機校正的過程中，其主要目標為解決三維立體座標空間轉換成二維平面座標空間產生的未知相機參數，一般而言，相機參數可以分成內部參數(Intrinsic Parameters)與外部參數(Extrinsic Parameters)，為相機模組中個座標系統間的相互轉換關係。

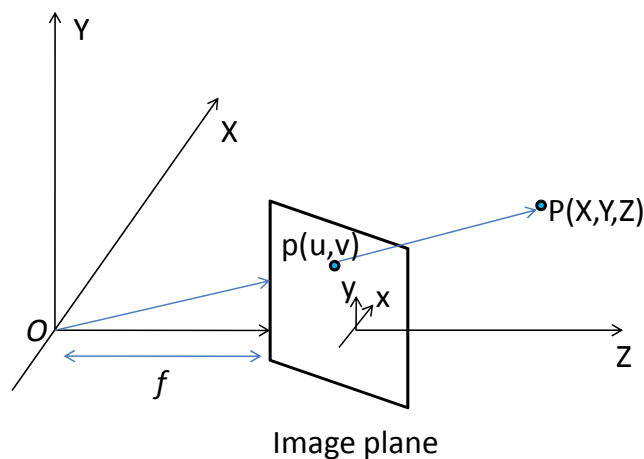


圖 2-2 相機針孔模型示意圖

內部參數與相機位置無關，其參數可視為針孔相機模型(Pinhole camera model)，如圖 2-2 所示，針孔相機模型是利用相機鏡頭的鏡心當作針孔，作針孔成像的轉換，內部參數矩陣包含焦距長、型變參數(Skew factor)、中心點(Principal point)，現代數位變焦鏡頭其內部參數會隨著焦點不同而改變。

將 3D 座標系統上得點投影到 $Z = f$ 的影像平面上， O 為相機中心點、 f 為相機焦距長，故座標系統屬於相機座標系統。依照 Z 與 f 的投影長度，可利用三

角的幾何關係獲得關係式如下式(2.1)表示：

$$(x, y) = \left(\frac{Xf}{Z} + u_c, \frac{Yf}{Z} + v_c \right) \quad (2.1)$$

其中 (u_c, v_c) 為中心點、 f 為鏡頭焦距長、 Z 為深度值，我們將影像空間上的點與三維空間的點用轉換關係式可表示成：

$$s \cdot p = \begin{bmatrix} sx \\ sy \\ s \end{bmatrix} = \begin{bmatrix} f_u & 0 & u_c & 0 \\ 0 & f_v & v_c & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \cdot \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix} = M \cdot P \quad (2.2)$$

其中 s 表示一純量、 p 表示影像平面上的座標點、 P 為三維空間中的對應座標點、 M 是一個 3×4 的矩陣，在這裡我們稱為投影矩陣(Projective matrix)。

承上式(2.2)可改寫成：

$$p = A [I_{3 \times 3} | 0] P_{cam} \quad (2.3)$$

$$A = \begin{bmatrix} f_u & \gamma & u_0 \\ 0 & f_v & v_0 \\ 0 & 0 & 1 \end{bmatrix}_{3 \times 3} \quad (2.4)$$

上述的投影矩陣在環境設定中，影像平面的 x 軸與 y 軸兩個方向的比例是相同的，由於相機的鏡頭失真關係，一般而言，相機的 x, y 軸比例可能不一定相等，可將焦距 f 分為 f_u 與 f_v 以代表 x 軸與 y 軸的不同焦距，內部參數包括了 skew factor γ 表示影像平面 x 軸與 y 軸夾角對於直角的歪斜程度，其內部參數矩陣可表示如式(2.4)。

外部參數與相機的擺設位置與參考點制定有關，其作用為將視界座標轉換為以相機鏡心為原點的座標，有旋轉矩陣與位移矩陣。如圖 2-3 所示。

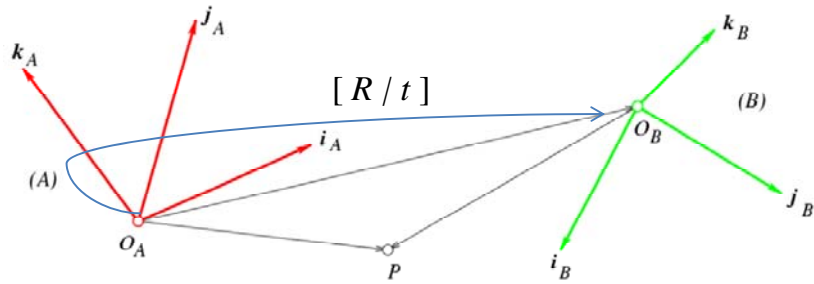


圖 2-3 相機外部參數轉換關係示意圖

其中 R 為世界座標中心與相機座標中心的轉換關係矩陣， t 表示為一個位移向量，可表示為下式(2.5)：

$$\begin{bmatrix} X_{cam} \\ Y_{cam} \\ Z_{cam} \end{bmatrix} = R_{cw} \begin{bmatrix} X_w \\ Y_w \\ Z_w \end{bmatrix} + t_{cw} = [R_{cw} \mid t_{cw}] \cdot \begin{bmatrix} X_w \\ Y_w \\ Z_w \\ 1 \end{bmatrix} \quad (2.5)$$

藉由式(2.3)與式(2.5)可以得知，相機影像平面上的位置可以利用空間中的三維座標經由內部參數以及外部參數得到下式(2.6)：

$$s \cdot \hat{p} = A \cdot P_{cam} = A \cdot [R_{cw} \mid t_{cw}] \cdot \hat{P} = M \cdot \hat{P} \quad (2.6)$$

$$\tilde{p} = \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} \quad \tilde{P} = \begin{bmatrix} X_w \\ Y_w \\ Z_w \\ 1 \end{bmatrix} \quad (2.7)$$

其中 A 為內部參數、世界座標系統與相機座標系統利用旋轉矩陣 R 與平移向量 t 表示其轉換關係，其中 R 為旋轉矩陣、 t 為位移向量。 M 為一個 3×4 的投影矩陣， \tilde{p} 代表在平面影像上的齊次座標點、 \tilde{P} 則代表在三維空間中的其次座標點。

當我們了解相機參數後，便可以透過不同位移、角度甚至是不同焦距所擷取

的影像，依據彼此共同的對應點求得各影像間的對應點轉換關係。

2.3 二維空間轉換映射 (2D Warping)

在 3D 互動式圖學應用領域上，我們盡可能希望利用低成本的硬體設備來建立複雜的幾何模型，三維空間轉換映射在 Image-based rendering (IBR) 是經常使用的技巧[7][11]，在傳統的圖學應用中，藉由鄰近視角的影像資訊當作參考影像合成得到我們合成視角的影像資訊，在轉換視角的過程中，如圖 2-4 所示，Warping 的目的在於我們找出來源影像 $f(x,y)$ 與目的影像 $g(x',y')$ 的轉換關係式 $T(x,y)$ ，3D warping 的缺點是如果合成視角和來源視角的角度相差越大，這時候來源影像的點經由轉換關係式轉換後，會因為遮蔽效應的問題 (Occlusion) 使得顯示的參考資訊因此變少影響顯示影像產生不連續性的空洞，造成影像品質下降。

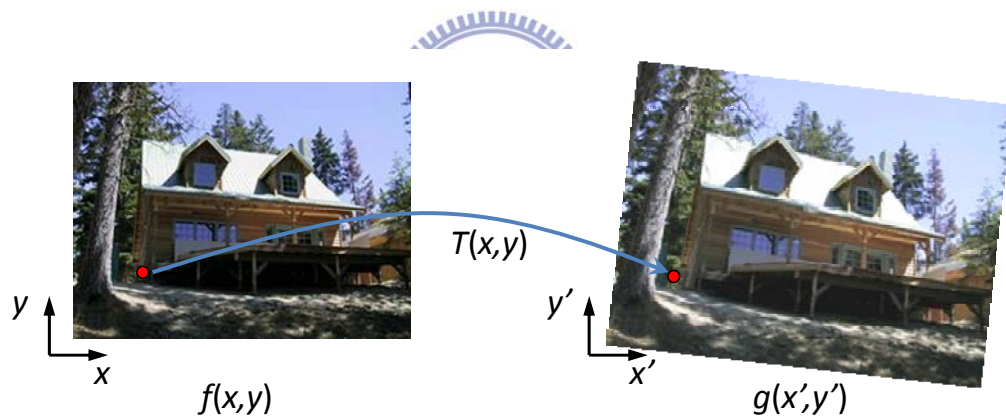


圖 2-4 目標影像與來源影像之間的三維轉換映射關係

考慮一個以 $(\hat{i}, \hat{j}, \hat{k})$ 為基底向量的三維空間座標的方向向量 \bar{d} ，如圖 2-5 所示，我們可以定義一個二維座標的影像平面，假設影像空間的原點在影像平面上的左上角，那麼我們可以在影像平面上的齊次點座標與三維座標可以找到一個對應關係式。其中在影像座標系統的座標軸我們以 (u, v) 表示，方向向量 d 代表對應射線的方向性。映射矩陣 P 與向量 \bar{a} , \bar{b} , \bar{c} 的關係如同式(2.8)所示， \bar{a} , \bar{b} , \bar{c} 與 \bar{d} 有相

同的座標系統，其關係式如圖 2-6 所示、 \bar{a} , \bar{b} 可以表示成一個以 \hat{s} 與 \hat{t} 為基底向量的 $(\hat{i}, \hat{j}, \hat{k})$ 影像座標系統、 \bar{c} 代表從光線原點到影像平面原點的向量。

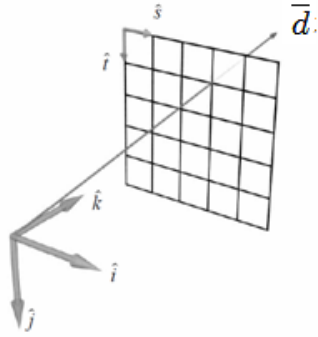


圖 2-5 影像空間對應光射線關係

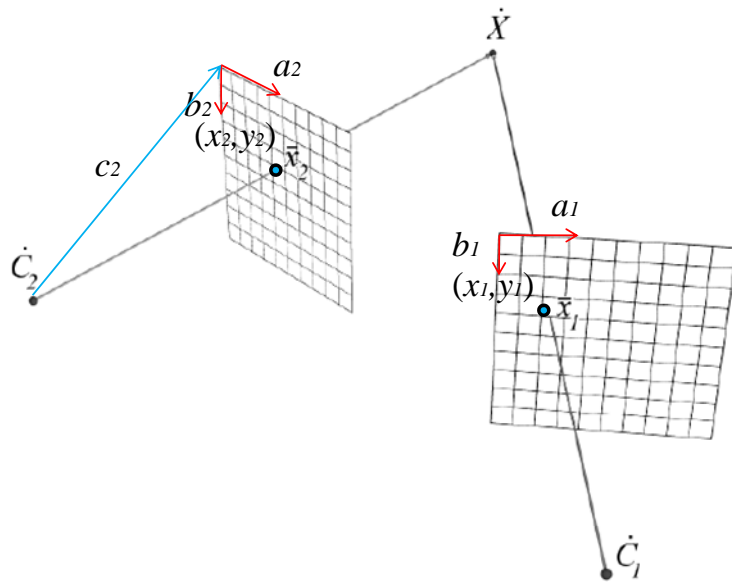


圖 2-6 三維空間中的一齊次點座標與不同影像平面的對應關係示意圖

$$\vec{d} = \begin{bmatrix} d_i \\ d_j \\ d_k \end{bmatrix} = \begin{bmatrix} a_i & b_i & c_i \\ a_j & b_j & c_j \\ a_k & b_k & c_k \end{bmatrix} \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = P \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} \quad (2.8)$$

在一個三維空間座標系統中，兩個不同的針孔相機座標系統皆可對應於同一個點。位於第一個影像平面座標系統的點 \bar{x}_1 以 \hat{C}_1 當作原點可以決定為光線射線 \bar{d}_1 ，其方向與點 \bar{x}_1 的關係為 $\bar{d}_1 = P_1 \bar{x}_1$ 。相同地，位於第二個影像平面座標系統的點 \bar{x}_2

以 \dot{C}_2 當作原點可以決定為光線射線 \vec{a}_2 ， \dot{X} 對應到兩個不同視角對應關係式如下式(2.9)所表示：

$$\dot{X} = \dot{C}_1 + (\vec{a}_1 x_1 + \vec{b}_1 y_1 + \vec{c}_1) / \delta_1 = \dot{C}_2 + (\vec{a}_2 x_2 + \vec{b}_2 y_2 + \vec{c}_2) / \delta_2 \quad (2.9)$$

其中 δ 代表視差值(Disparity)，通常我們會假設兩個影像平面都是平行排列的，如圖 2-7 所示，我們可以表示成下式(2.10)：

$$\frac{1}{\delta_2} \begin{bmatrix} x_2 \\ y_2 \\ 1 \end{bmatrix} \begin{bmatrix} a_{2x} & b_{2x} & c_{2x} \\ a_{2y} & b_{2y} & c_{2y} \\ a_{2z} & b_{2z} & c_{2z} \end{bmatrix} = (\dot{C}_1 - \dot{C}_2) + \frac{1}{\delta_1} \begin{bmatrix} x_1 \\ y_1 \\ 1 \end{bmatrix} \begin{bmatrix} a_{1x} & b_{1x} & c_{1x} \\ a_{1y} & b_{1y} & c_{1y} \\ a_{1z} & b_{1z} & c_{1z} \end{bmatrix} \quad (2.10)$$

最後可以推出以下結果(2.11)：

$$(x_2, y_2) = \left(\frac{k_1 x_1 + k_2 y_1 + k_3 + k_4 \delta_1}{k_9 x_1 + k_{10} y_1 + k_{11} + k_{12} \delta_1}, \frac{k_5 x_1 + k_6 y_1 + k_7 + k_8 \delta_1}{k_9 x_1 + k_{10} y_1 + k_{11} + k_{12} \delta_1} \right) \quad (2.11)$$

其中 K 可表示成下式(2.12)與(2.13)：

$$\mathbf{K} = \begin{bmatrix} k_1 & k_2 & k_3 \\ k_5 & k_6 & k_7 \\ k_9 & k_{10} & k_{11} \end{bmatrix} = \begin{bmatrix} a_{2x} & b_{2x} & c_{2x} \\ a_{2y} & b_{2y} & c_{2y} \\ a_{2z} & b_{2z} & c_{2z} \end{bmatrix}^{-1} \begin{bmatrix} a_{1x} & b_{1x} & c_{1x} \\ a_{1y} & b_{1y} & c_{1y} \\ a_{1z} & b_{1z} & c_{1z} \end{bmatrix} \quad (2.12)$$

$$\begin{bmatrix} k_4 \\ k_8 \\ k_{12} \end{bmatrix} = \begin{bmatrix} a_{2x} & b_{2x} & c_{2x} \\ a_{2y} & b_{2y} & c_{2y} \\ a_{2z} & b_{2z} & c_{2z} \end{bmatrix}^{-1} (\dot{C}_1 - \dot{C}_2) \quad (2.13)$$

3D Warping 可以精確計算視角的旋轉與位移參數，然而我們在計算位移參數的過程中可能會導致兩個問題，鄰近的畫素可以藉由 Warping 的過程中會移動不同距離，因此影像重建將會是一個困難的問題；另一方面，所有的畫素雖然可以經由 3D warping 對應到正確的目標位置，但參考影像被物體遮蔽後，參考影像內本身並不存在的資訊則會造成遮蔽效應的問題(Occlusion)，使得合成影像資訊變少影響顯示影像產生不連續性的空洞，造成影像品質下降。

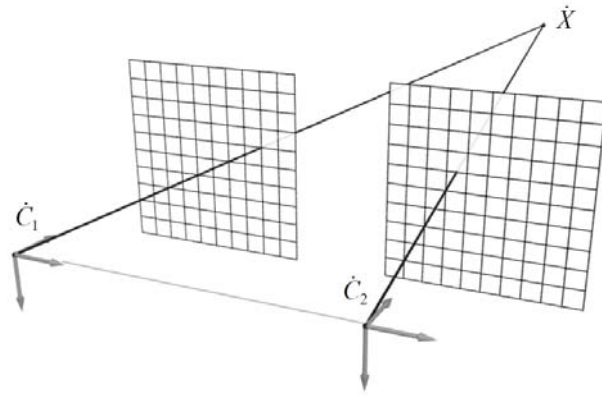


圖 2-7 兩相機點座標與深度值的對應關係示意圖



2.4 視點合成參考軟體介紹 (View Synthesis Reference Software)

2.4.1 軟體平台概述 (Overview of Software Platform)

MPEG 標準會議已經提供景深預測參考軟體(Depth Estimation Reference Software) 以及影像合成參考軟體(View Synthesis Reference Software)，我們使用的影像合成參考軟體 2.0 版本(VSRS 2.0)是從日本名古屋大學的會員網頁下載當作我們的演算法設計平台[19]，其中圖 2-8 代表名古屋大學的程式架構，程式本身需要外掛 OpenCV 才可以使用。在一開始，我們利用 viewsynthesis.cfg 檔案輸入相機參數以及測試影像與深度影像的路徑設定，main.cpp 檔案包含了 viewsynthesis 函式連結到 warping.cpp 檔案，在 viewsynthesis 函式中，首先我們會先初始化相機參數矩陣，根據測試影像與深度影像的資訊加以計算合成影像。本篇論文的最後，我們透過計算 PSNR 的方式來驗證原始拍攝影像與合成影像之間的影像品質。

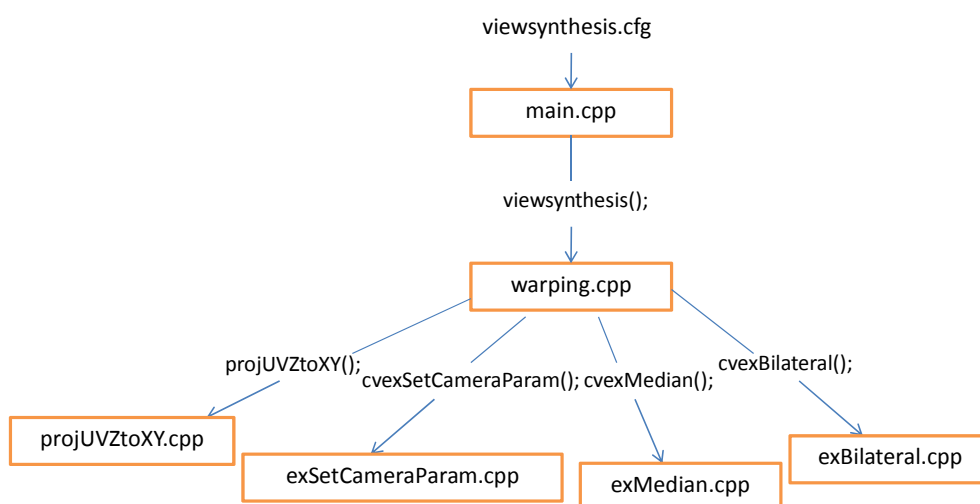


圖 2-8 VSRS 2.0 的程式設計平台設計架構

2.4.2 品質測試指標 (Quality Metrics)

在實驗過程中，為比較不同演算法的合成影像品質，我們採用計算 PSNR 的方式來比較合成影像的品質。PSNR 是針對參考影像與合成影像之間的平方差異值的一種度量定義，為最大值信號和平方差之間的一個比例值。經過影像壓縮之後，輸出影像的訊號與原始影像的訊號會有不同，為了衡量處理後的影像品質，我們常 PSNR 值來認定某個處理程序是否達到優良的效果。

在我們的演算法中，我們會利用左右的鄰近影像來產生中間的合成影像。在測試時，合成影像的位置與實際拍攝影像的位置相同，用此可利用合成影像與參考影像來計算 PSNR。一般而言，計算 PSNR 都會將 RGB 色彩空間轉換至 YUV 色彩空間，而我們通常只使用 Y 頻段來計算 PSNR，Y 頻段表示明亮度 (Luminance)，RGB 與 YUV 之間的轉換關係式如下式(2.14)所示：

$$Y(u, v) = 0.299 \cdot R(u, v) + 0.587 \cdot G(u, v) + 0.114 \cdot B(u, v) \quad (2.14)$$

其計算 PSNR 的方式如下式(2.15)：

$$PSNR = \frac{255^2}{MSE} \quad (2.15)$$

$$MSE = \frac{1}{W \times H} \sum_{s,t=0}^{W-1,H-1} (Y(s, t) - \hat{Y}(s, t))^2 \quad (2.16)$$

其中 W 與 H 分別代表影像寬與影像長， Y 與 \hat{Y} 分別代表參考影像與合成影像。

2.4.3 MPEG 測試影像介紹 (Introduction to MPEG Test Sequences)

FTV 從 MPEG 於 2002 年開始提出計畫案，至今 MPEG 標準會議已經開始陸續針對多重影像的資料壓縮標準提出許多方案。在 2011 年 1 月，已經累積許多來自世界各地的專家學者投注許多心血，集結多個 MPEG FTV 的測試影像 [20][21][22]。

2011 年 3 月 MPEG 委員會正式展開多重影像資料壓縮技術競賽標準(MVC)，所提供的測試影像是一維水平直線排列的攝影機陣列。事實上，由於數學模型的精確度以及物理光學性質的限制，在拍攝的過程中仍有許多不確定的因素仍須克服，因此針對每個不同的測試影像都必須要事先進行相機校正並且取得相機參數。

MPEG FTV 的測試影像可以從 MPEG 會員網頁下載，更詳盡的資料可參考 [20][21]。較常使用來品質驗證的測試影像通常為以下四個測試影像：Champagne tower, Kendo, Book arrival, Newspaper，其測試影像介紹如表 1 所示。



表 1 MPEG 測試影像介紹表

Data set	Champagne tower	Kendo	Book arrival	Newspaper
Image size (Width x height)	1024 × 768	1024 × 768	1024 × 768	1024 × 768
Frame rate	30	30	16.67	30
Total frames	500	400	100	300
Input				
Corresponding Depth map				
Original Viewpoint Pair	39-41	03-05	08-10	04-06
OL-OR	38-42	02-05	07-11	03-07
Synthesized Viewpoint	40	04	09	05
Participant	Nagoya Univ. (Japan)	Nagoya Univ. (Japan)	HHI (Germany)	GIST, Sumsung (Korea)

第3章 傳統任意視點合成的演算法介紹

3.1 傳統視點合成演算法概述 (Overview)

在多媒體領域當中，任意自由視點合成電視(FTV)已經成為一門熱門的研究領域，現在不管產業界或者學術界皆已投入相當大的心力研發。過去已有許多視點合成的演算法被提出，Mori et al.[9]的演算法是假設景深預估(Depth Estimation)的景深圖是穩定且品質良好的，目前預測景深圖的演算法是使用立體視覺最佳化的計算方式，其景深圖品質已經相當良好[1]。然而此演算法中，由於映射後的景深圖會產生不連續性的裂痕與破洞，因此，在影像成像的過程前，Mori 使用中值濾波器(Medium filter)以及雙閾值濾波器(Bilateral filter)將不連續性的裂痕修復。Min et al.[10]的演算法假設景深圖是品質良好的，但即使使用立體視覺最佳化的方法計算景深圖，景深圖仍然有可能會因為雜訊的關係進而推算錯誤，Min 利用簡單的中值濾波器以減少景深圖產生的破洞。然而，大部分的自然場景中總是會包含兩層或者是更多層不同的景深，而這個現象使得預測景深圖的時候邊緣以及不連續的地方容易產生不可採用的錯誤資訊。Yang et al.[16]針對遮蔽效應所產生的空洞加以分析可信賴的邊緣區域，進而改善現有的視點合成演算法的設計架構。

3.2 前向轉換映射問題 (Forward Warping Problem)

3.2.1 物體邊緣處產生的瑕疵現象 (Artifacts on Object Boundary)

首先，我們利用不同的景深模式將瑕疵現象(Artifacts)分類成不同的例子加以分析。在實際場景成像的過程中，相較於景深圖的邊緣區域通常會包含兩種不同的景深值。由於景深圖本身在紋理較平緩的區域容易產生預測錯誤，因此在錯誤

的映射過程中容易導致許多肉眼可見的瑕疵現象。舉例來說，如果背景的畫素映射到前景的位置上，則背景上會產生許多瑕疵現象。事實上，不論是參考影像或是合成影像，我們都會假設其影像平面都是相互平行(Rectified)，目前傳統上的視點合成演算法，大部分前向映射(Forward Warping)通常都是利用左邊以及右邊的參考影像資訊加以合成中間視點的影像。

正如我們所知，根據兩層不同深度值的邊緣有兩種不同的表示法，若是前景位於左半邊而背景的部分位於右半邊，我們稱為 FB Pattern (Foreground Background Pattern)，我們用相同的概念，也可以得到 BF Pattern (Background Foreground Pattern)，其示意圖如圖 3-1 與圖 3-2 所示。

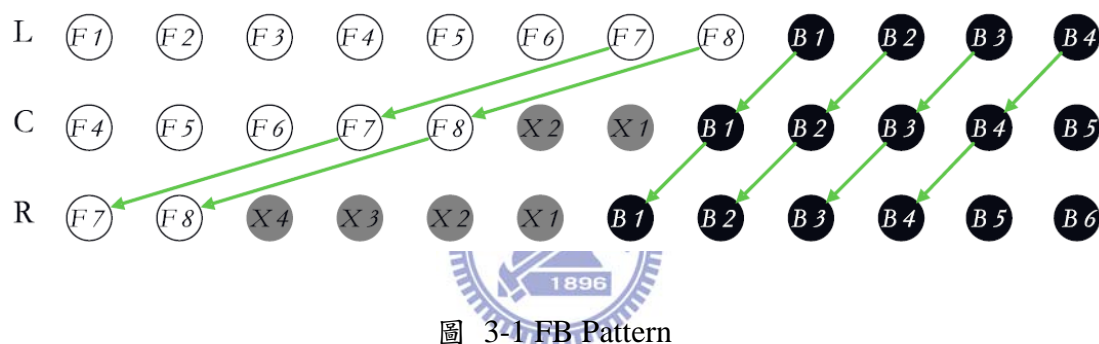


圖 3-1 FB Pattern

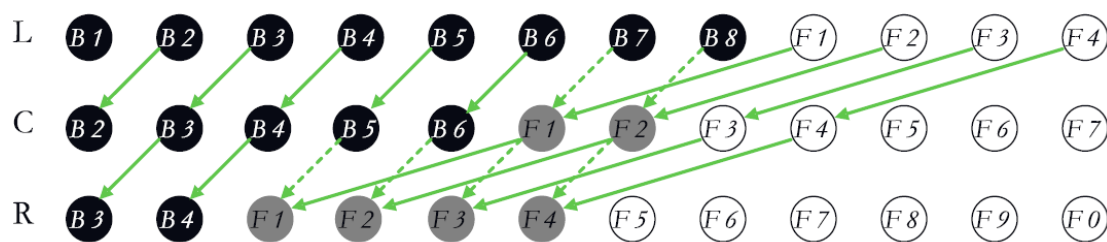


圖 3-2 BF Pattern

其中白色的圓代表是前景的畫素、黑色的圓代表是背景的畫素、灰色的圓代表是無法預測的畫素，綠色實線的部分為正確的投影映射方向、綠色虛線的部分為遮蔽效應造成的投影映射方向。前景的視差值假設為 3、背景的視差值設為 1，X 代表訊號強度與背景相似的畫素。更詳盡的說明如下：

L: 來自左邊參考影像的畫素 (Left reference)

R: 來自右邊參考影像的畫素 (Right reference)

FD: 前景的視差值 (Foreground disparity)

BD: 背景的視差值 (Background disparity)

FG: 正確的前景視差值 (Ground truth disparity of foreground)

BG: 正確的背景視差值 (Ground truth disparity of background)

利用以上的定義方式註解兩層不同深度值的邊緣表示法，我們可以用五種不同的深度值模式來分別表示八種可能產生 Artifacts 的例子，如表 2 所示：

表 2 造成合成瑕疵現象的情況分析與深度值模式

Boundary	Reference	Artifacts case	Depth mode	Artifacts
FB Pattern	Left view	Case 1-FBLB	Mode 1: $BD > FG$	Background
		Case 2-FBLF	Mode 2: $FD < FG$ and $BD < FG$	Foreground
		Complement	Mode 4: $BD \leq FG$ and $FD \geq FG$	none
	Right view	Case 3-FBRF	Mode 3: $FD > FG$	Foreground
		Case 4-FBRB	Mode 2: $FD < FG$ and $BD < FG$	Background
		Complement	Mode 5: $FD \leq FG$ and $BD \geq FG$	none
BF Pattern	Left view	Case 5-BFLF	Mode 3: $FD > FG$	Foreground
		Case 6-BFLB	Mode 2: $FD < FG$ and $BD < FG$	Background
		Complement	Mode 5: $FD \leq FG$ and $BD \geq FG$	none
	Right view	Case 7-BFRB	Mode 1: $BD > FG$	Background
		Case 8-BFRF	Mode 2: $FD < FG$ and $BD < FG$	Foreground
		Complement	Mode 4: $BD \leq FG$ and $FD \geq FG$	none

事實上，在兩層不同深度值的邊緣表示法下，不管是 FB Pattern 或是 BF

Pattern 都會各自有四種不同瑕疵現象的情況產生。而瑕疵現象大致上可分為由前景以及背景產生的兩大種類，在不同的深度值模式下會造成不同的結果。若以 Case 1-FBLB 為例，Case 1 代表瑕疵現象的現象是產生在來自左邊參考影像的 FB Pattern 邊緣上，此瑕疵現象是從背景的部分產生。由觀察而得這八種可能產生瑕疵現象的情形實際上大部分都集中在三種深度值模式，分別為：Mode 1、Mode 2、Mode 3。其中 Mode 1： $BD > FG$ 代表背景的預測視差值大於前景的真實視差值，使得在映射後背景的畫素可能會造成瑕疵現象。Mode 2： $FD < FG$ and $BD < FG$ 代表前景與背景的視差值皆小於前景真實的視差值，但背景預估的視差值通常都小於前景真實的視差值，使得映射過成前景的畫素可能會造成瑕疵現象。Mode 3： $FD > FG$ 代表前景的視差值皆大於前景真實的視差值，使得映射過成前景的畫素可能會造成瑕疵現象。其餘兩種深度值模式 (Mode 4、Mode 5) 在投影映射的過程中所產生的預測錯誤則不在我們的考慮範圍內。事實上，在現在景深預測的運算過程中，由於視差值的幾何限制，因此背景的視差值不可能大於真實的前景視差值，這也代表在 Mode 1 的景深模式下，Case 1 與 Case 7 其實是相同的。另外其它四種產生瑕疵現象的情況 (Case 2, 4, 6, 8) 都是在 Mode 2 下發生的，代表著 Mode 2 是最常見的情形，我們會在之後的章節詳細分析在 Mode 2 下可能會發生瑕疵現象的情況。

3.2.2 可信賴性模型 (Reliability Model)

一般而言，我們假設所有參考影像的影像平面都必須是平行的，因此所有景深圖以及視差值皆是一維的純量。因此在猜測瑕疵現象的區域前，在景深估算的過程中所拍攝的擷取影像與景深圖之間的影像邊緣未必一致。我們必須事先需要知道景深圖的哪些邊緣區域在視點合成的過程中是錯誤的資訊，或者是不可信賴的區域。然而，真實的場景合成影像上的瑕疵現象都是未知的，所以必須利用參考影像上的資訊推導出一套判斷信賴區域的準則。

在圖 3-4 中，我們假設深度值模式 Mode 2 下合成中間視點影像的過程中，可信賴的投影映射為綠色實線的畫素，以 R 表示；不可信賴的投影映射我們標示為紅色虛線的畫素，以 UR 表示。事實上，通常只有靠近景深圖的邊緣部分一半邊的畫素位置才是確實不可信賴的，而邊緣部分上另一半的畫素位置是假性不可信賴的(Pseudo unreliable)。若我們能確切知道何者為背景或是前景的部分，則將可信賴的標示範圍擴大，然而，由於每一層景深的深度值皆不相同，若將可信賴的標示範圍擴大，可能使得位於邊緣區域的深度值產生預測錯誤的結果。

3.2.3 三層互相連接之深度值的瑕疵現象分析 (Artifacts of Triple-layer Boundary)

在章節 3.2.1 我們已經介紹兩層不同深度值的邊緣表示模型。在實際的自然場景中，某些場景物體的面寬看起來十分細小，例如：樹枝、電線杆等。代表與物體之間相對應的深度資訊值或是視差值擁有非常窄小的對應面積，導致了兩層以上或多層不同深度值的邊緣變化。從景深圖的頻域角度來看，越多層不同深度值的邊緣變化同時代表其高頻成份比起兩層深度值的情況下更為複雜許多。

由於多層不同深度值的邊緣情況十分複雜，故在本章節我們將原本複雜的情況簡化成三層不同深度值的邊緣變化情況加以討論與說明。在這裡我們假設任何景深圖的邊緣區域可以用兩層或者是三層不同深度值的邊緣模型表示。然而，現有的景深預測的演算法至今仍沒辦法準確預測視差值的範圍。

理論上，三層的不同深度值的邊緣區域共包含以下三種不同部分：前景層、背景層以及介於前景層與背景層之間的中間層，我們以 MD 代表中間層的視差值，中間層的視差值與前景層以及背景層之間的視差值大小比較關係式如下式(3.1)所示：

$$BD < MD < FD \quad (3.1)$$

同時也可以利用兩層不同深度值的邊緣表示法來簡化此概念。

兩層不同深度值的模型僅包含 FB Pattern 或 BF Pattern 兩種不同的表示法，與兩層不同深度值的模型相反的是三層不同深度值的模型有六種不同的邊緣表示法，分別為：*MBF*、*MFB*、*FMB*、*FBM*、*BFM*以及*BMF*。以 *MBF* 為例，*MBF*代表中間層的視差值位於深度值的邊緣區域最左邊的部分、背景層位於中間的部分、前景層則位於右邊的部分。

雖然在排列順序上這六種不同的邊緣表示法彼此不同，但是以上每種三層的邊緣表示方法我們都可以簡化成由兩種兩層不同深度值的邊緣表示法所組成，如表 3 所示，如果某一層擁有較小的視差值則可視為背景層。另一方面，若擁有較大的視差值則可視為前景層，其中每種兩層不同深度值的邊緣部分我們都可以利用 FB 或 BF Pattern 來表示。

若在三層不同深度值伴隨著不同顏色的情況下，我們在分析瑕疵現象的情況下，則可以用兩層不同深度值的邊緣表示法表示。簡單來說，三層不同深度值的邊緣表示模型中，位於左邊的深度值與位於右邊的深度值有著相同的顏色，但與中間的部分顏色不同，則我們可以將此例子視為兩層不同深度值的邊緣表示。舉例來說，在自然場景中，某些場景物體的面積寬看起來十分細小，其中 *BFB* 是最常見的表示法，*BFB*中間部分則為此場景物體的前景深度值，可以視為分別由 BF Pattern 與 FB Pattern 組合而成。

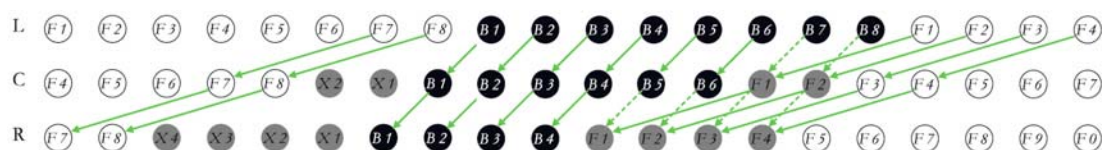
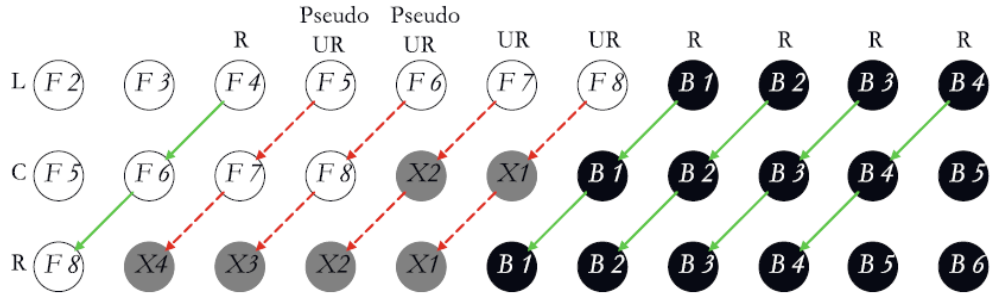
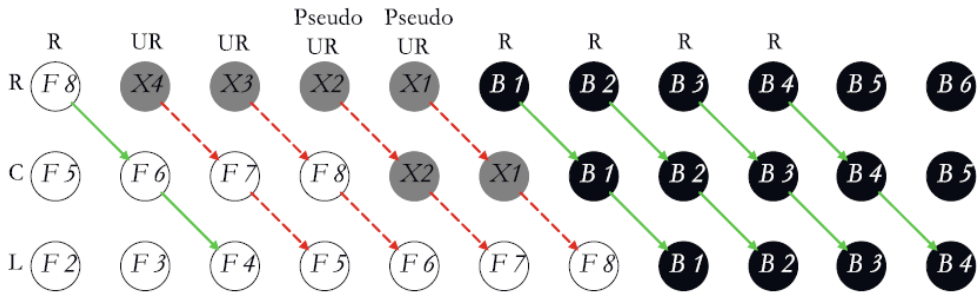


圖 3-3 FBF 表示示意圖

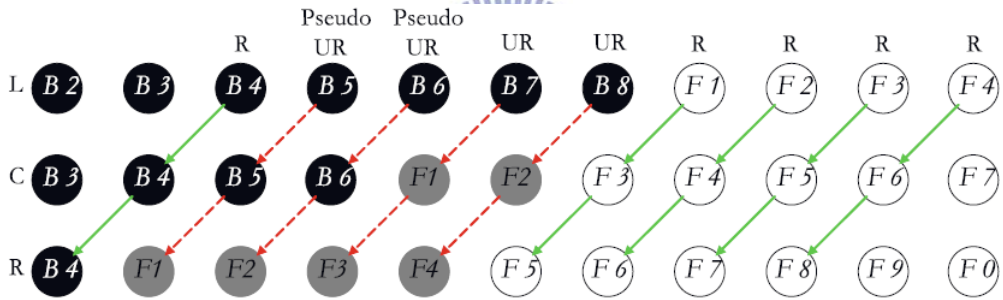
事實上，在 *FMB* 與 *BMF* 兩種不同的表示情況下，若前景部份的深度值與背景部份的深度值的部分顏色相似時，我們也可以分別簡化成 *FBF* 與 *BFB* 的兩種不同的表示方法，如圖 3-3 所示。



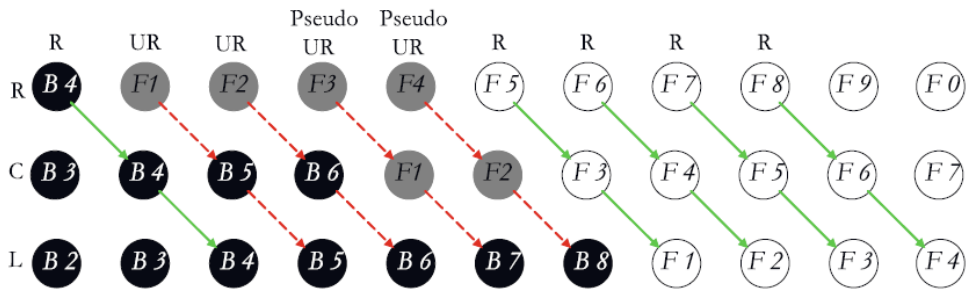
(a) Case 2: FBLF



(b) Case 4: FBRB



(c) Case 6: BFLB



(d) Case 8: BFRF

圖 3-4 不同的瑕疵現象情況分析表示示意圖：(a) Case 2-FBLF, (b) Case 4-FBRB,

(c) Case 6-BFLB, and (d) Case 8-BFRF

表 3 三層不同深度值邊緣表示法

Triple-layer Pattern	Two-layer counter-part	Simplification
MBF	FB-BF	FBF
MFB	BF-FB	BFB
FMB	FB-FB	
FBM	FB-BF	FBF
BFM	BF-FB	BFB
BMF	BF-BF	

由於三層不同深度值的邊緣表示法可以經由簡化與拆解成兩層不同深度值的邊緣模型表示，如表 2 所示。我們可以利用相同的深度值模式來加以分析，其分析結果我們彙總於表 2。從表 2 中可以發現許多三層不同深度值的邊緣表示法都可以簡化成兩層不同深度值的邊緣模型。若以 *BFB pattern* 為例，我們利用左邊參考影像投影至中間的合成過程中來解釋瑕疵現象的發生情況，同樣也可以利用相同的方式解釋其它瑕疵現象可能產生的情況，與兩層不同深度值的情況相同，如下圖 3-6 與圖 3-7 所示，其中圖 3-5 代表正確的投影映射過程。(其中紅色虛線代表錯誤的投影過程)

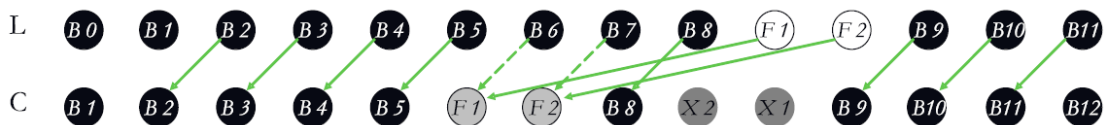


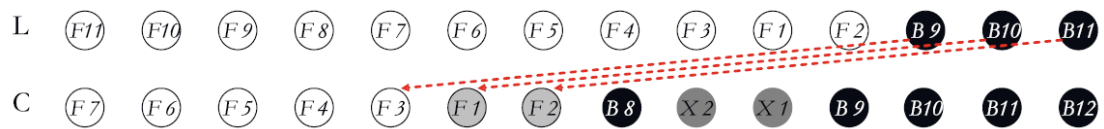
圖 3-5 三層不同深度值的正確映射方向表示示意圖

在圖 3-6 中，示意圖為深度值模式 Mode 1 與 Mode 2 的狀態下，在三層不同深度值的邊緣模型下產生 *BFB Pattern* 所造成的瑕疵現象與兩層不同深度值的邊緣模型下產生 *FB Pattern* 所造成的瑕疵現象比較結果圖，其中綠色實線為正

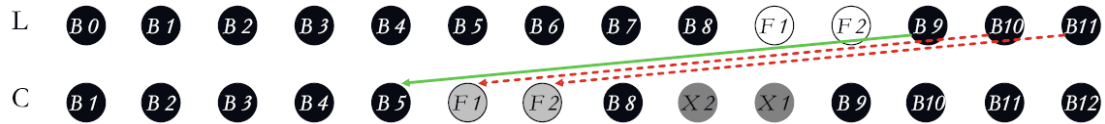
確投影映射的位置、紅色虛線為錯誤投影映射的位置。圖 3-7 則是代表 BF Pattern 所造成的 Artifacts 比較結果圖。從表 2 與表 4 中，我們可以發現不論是 *BFB* 或者是 *FBF* 的邊緣表示模型都可以簡化成兩層不同深度值的模型。

表 4 三層不同深度值可能造成的瑕疵現象分析表

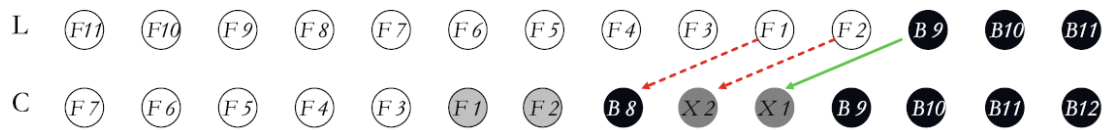
Boundary	Reference	Two-layer part	Depth mode	Artifacts
BFB	Left view	FB	Mode 1	Partial background
			Mode 2	Same artifacts with two-layer model
		BF	Mode 3	Same artifacts with two-layer model
			Mode 2	Partial background , new foreground
	Right view	FB	Mode 3	Same artifacts with two-layer model
			Mode 2	Partial background , new foreground
		BF	Mode 1	Partial background
			Mode 2	Same artifacts with two-layer model
FBF	Left view	FB	Mode 1	Same artifacts with two-layer model
			Mode 2	Partial background , new foreground
		BF	Mode 3	Partial background
			Mode 2	Same artifacts with two-layer model
	Right view	FB	Mode 3	Partial background
			Mode 2	Same artifacts with two-layer model
		BF	Mode 1	Same artifacts with two-layer model
			Mode 2	Partial background , new foreground



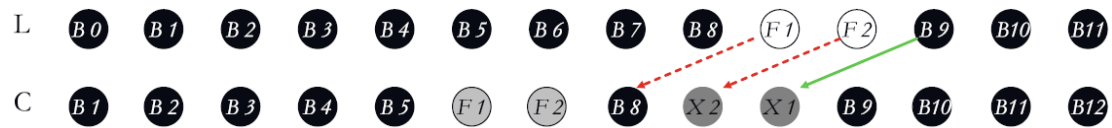
(a) Mode 1, two-layer artifacts at “F3”, “F1” and “F2” in center view.



(b) Mode 1, partial artifacts in triple-layer model at “F1”, “F2” in center view.

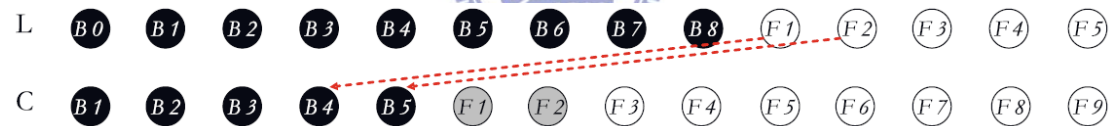


(c) Mode 2, two-layer artifacts at “B8”, “X2” in center view.

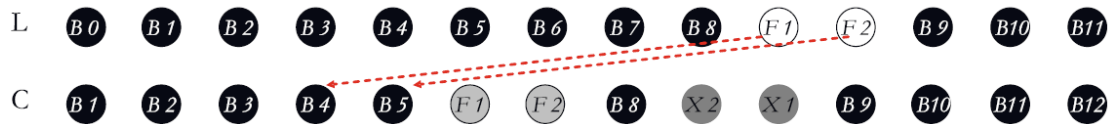


(d) Mode 2, same artifacts at “B8”, “X2” in center view.

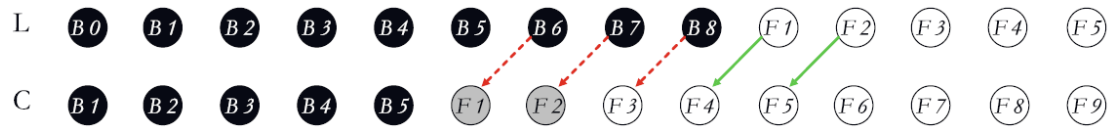
圖 3-6 FB 的情況下可能產生的瑕疵現象映射過程示意圖



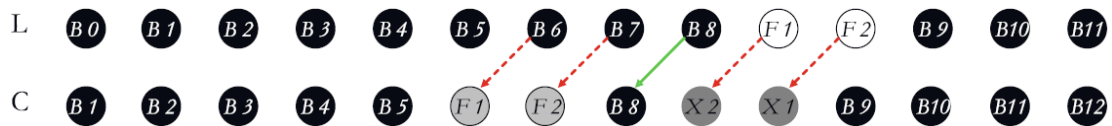
(a) Mode 3, two-layer artifacts at “B4”, “B5” in center view.



(b) Mode 3, same artifacts at “B4”, “B5” in center view.



(c) Mode 2, two-layer artifacts at “F1”, “F2” and “F3” in center view.



(d) Mode 2, partial artifacts in triple-layer model at “F1”, “F2”, new foreground artifacts at “X2”, “X1”.

圖 3-7 FB 的情況下可能產生的瑕疵現象映射過程示意圖

3.3 使用可靠性的判別演算法設計 (Reliability Check Algorithm)

在本節的一開始，我們會簡單描述 Yang[16]的視點合成的演算法架構，其演算法的流程如圖 3-8 所示，其詳細步驟我們會在此章節說明。

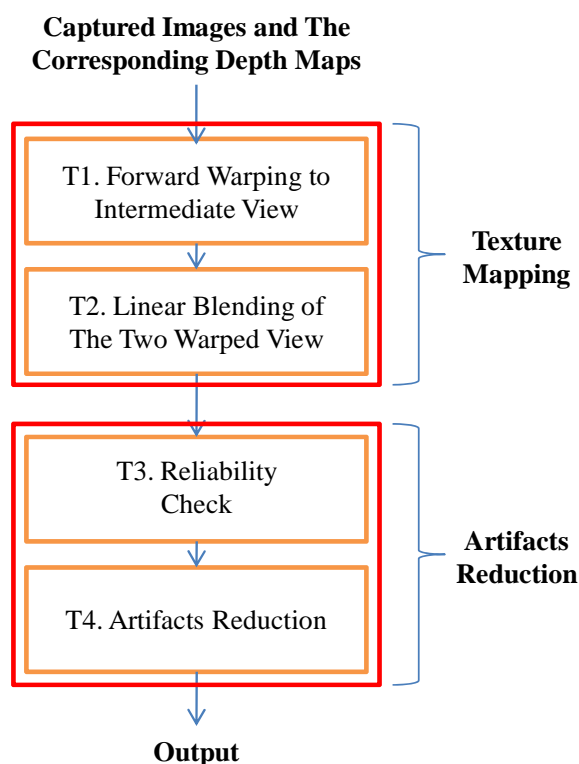


圖 3-8 Yang 的演算法設計架構流程圖[16]

3.3.1 前向轉換映射 (Forward Warping)

首先，我們必須將左邊視點的影像投影映射到右邊視點的影像平面上，反之，也必須將右邊視點的影像投影映射至左邊視點的影像平面上， $I_L(u_L, v_L)$ 以及 $I_R(u_R, v_R)$ 各別代表左邊以及右邊的影像畫素，我們給定各別的視差值 $D_L(u_L, v_L)$

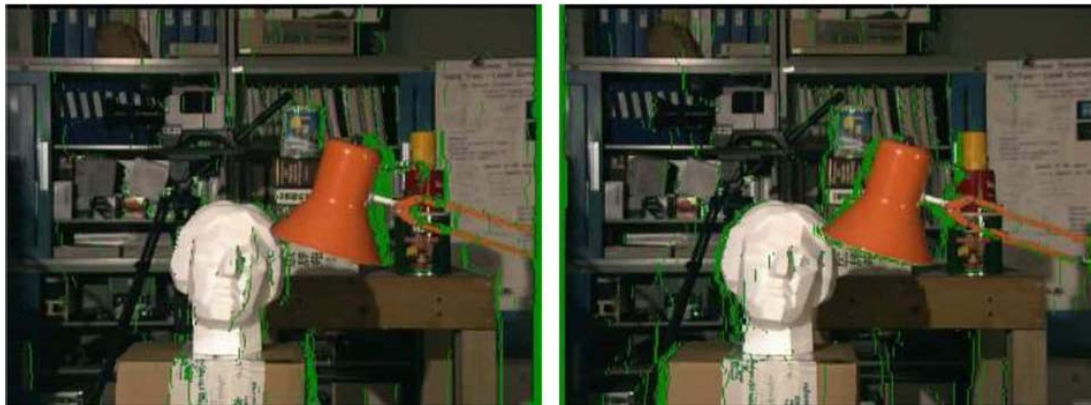
以及 $D_R(u_R, v_R)$ ，則我們可以正向投影映射後的結果表示如下式(3.2)以及(3.3)所示：

$$I_{LC}(u_L, v_L - D_L(u_L, v_L)) = I_L(u_L, v_L) \quad (3.2)$$

$$I_{RC}(u_R, v_R + D_R(u_R, v_R)) = I_R(u_R, v_R) \quad (3.3)$$

其中 I_{LC} 及 I_{RC} 分別代表由左邊以及右邊的參考影像投影映射至中間的合成影像。

從圖 3-9 中，經由分別從左邊與右邊的參考影像平面正向投影映射至中間位置的影像平面，綠色區域代表遮蔽效應所造成無法預測的空洞。



(a)

(b)

圖 3-9 合成影像由 (a)左邊 (b)右邊 參考視點所產生

3.3.2 線性比例疊合 (Linear Blending)

在前向映射之後， I_{LC} 及 I_{RC} 是兩張由不同參考影像映射後的結果。我們將這兩張影像進行 α 比例線性疊合(Linear blending)的運算，在圖 3-10 中，根據中間合成攝影機的位置與左右兩邊參考攝影機的位置差異值計算其比例關係得到 α 。

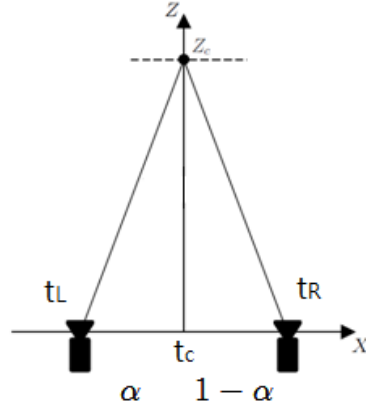


圖 3-10 待合成攝影機位置與左右兩邊的參考攝影機距離示意圖

其中 α 的計算方式如下式(3.4)所示：

$$\alpha = \frac{|t_C - t_L|}{|t_C - t_L| + |t_C - t_R|} \quad (3.4)$$

其中 t_c 代表中間影像的外部參數矩陣的位移位置， t_L 及 t_R 代表左邊及右邊參考影像的外部參數矩陣的位移參數。

$$Occ(u, v) = \begin{cases} 1, & I'(u, v) = 0 \\ 0, & I'(u, v) \neq 0 \end{cases} \quad (3.5)$$

在求得比例值 α 後，若左邊或右邊視點投影映射至中間視點的影像平面上的畫素位置因遮蔽效應的關係產生空洞、沒有任何色彩資訊時，則令 $Occ(u, v)$ 為 1，反之將設為 0，其數學表示式如下式(3.5)所示：

$$dst(u, v) = \begin{cases} (1 - \alpha)I_{LC}(u, v) + \alpha I_{RC}(u, v), & Occ_L(u, v) = 0 \text{ and } Occ_R(u, v) = 0 \\ I_{LC}(u, v) & , Occ_L(u, v) = 0 \text{ and } Occ_R(u, v) = 1 \\ I_{RC}(u, v) & , Occ_L(u, v) = 1 \text{ and } Occ_R(u, v) = 0 \\ \mathbf{0} & , Occ_L(u, v) = 1 \text{ and } Occ_R(u, v) = 1 \end{cases} \quad (3.6)$$

在式(3.6)中，若是左邊視點產生的前向映射結果與右邊視點產生的前向映射結果皆未產生無法預測的空洞時，則合成的目標影像畫素是採用線性疊合並輸出最後影像結果；若左邊視點前向映射後的結果產生無法預測的空洞，而右邊視點前向映射後的結果未因遮蔽效應產生空洞時，則合成的影像畫素採用右邊視點產生的正向投影映射結果。另一方面，若右邊視點前向映射後的結果產生無法預測

的空洞，而左邊視點產生的前向映射結果未因遮蔽效應產生空洞時，則採用左邊視點產生的正向投影映射結果；若左右兩個視點產生的前向映射結果皆產生無法預測的空洞時，則將目標合成影像的畫素位置值設為 0。至此步驟目前出來的結果我們稱為傳統的視點合成方法，而 Yang[16]提出改善的演算法架構我們會在之後的章節詳細說明。

3.3.3 可靠性判別法則 (Reliable Check)

此節最主要的想法已經在節 3.2.2 介紹過，接下來我們將會介紹實作的方法，與式(3.2)與(3.3)相似，我們可以得到分別從右邊與左邊前向映射之後產生的合成參考影像結果 I'_R 與 I'_L 與章節 3.3.2 產生的合成影像，接下來我們會去計算原始影像訊號強度與正向投影映射後的影像訊號強度之間的差異絕對值，其計算方式如下式(3.7)與(3.8)所示：

$$Dif_L(u_L, v_L) = |I_R(u_L, v_L - D_L(u_L, v_L)) - I'_R(u_L, v_L - D_L(u_L, v_L))| \quad (3.7)$$

$$Dif_R(u_R, v_R) = |I_L(u_R, v_R + D_R(u_R, v_R)) - I'_L(u_R, v_R + D_R(u_R, v_R))| \quad (3.8)$$

最後若是差異值小於某個臨界閾值，則認定此畫素對應的深度值是可信賴的，反之，對應的深度值則是不可靠的，如圖 3-11 所示。白色區域即為不可信賴的區域，其中此運算的臨界閾值是可以任意調整的，如式(3.9)所示：

$$Re(u, v) = \begin{cases} true & Dif(u, v) \leq threshold \\ false & Dif(u, v) > threshold \end{cases} \quad (3.9)$$

3.3.4 抑制瑕疵現象 (Artifacts Reduction)

由於遮蔽效應的問題，我們可以利用在 3.2.2 提到的互補原則，重新定義由 3.3.2 所產生的合成影像中的不可靠的畫素。修復後的結果如圖 3-12。紅色區域

代表是左右兩側的視點都是不可信賴的區域，其數學表示式如下式(3.10)所示：

$$I_{new}(u, v) = \begin{cases} I_{LC}(u, v), & Re_{LC} = true \text{ and } Re_{RC} = false \\ I_{RC}(u, v), & Re_{LC} = false \text{ and } Re_{RC} = true \\ I_{dst}(u, v), & Re_{LC} = true \text{ and } Re_{RC} = true \\ hole & , \text{ otherwise} \end{cases} \quad (3.10)$$

其中 Re_{LC} 與 Re_{RC} 是由 Re_L 與 Re_R 投影映射到中間合成視點後的結果。

此演算法最後一個部分是處理無法預測的空洞。一般而言，破洞通常是因為遮蔽效應在正向投影映射的過程中，導致無法預測影像平面上的畫素資訊，而遮蔽效應的問題總是發生在某個視角無法看到的區域。事實上，雖然大部分的破洞經由前面幾個步驟已經被修復，仍舊有少許部分的破洞仍需要被修復。最後我們利用影像修補(Image inpainting)的技巧填補這些殘留的破洞[30]，最後視點合成的結果如圖 3-13。所示。



圖 3-11 依照左邊參考影像所產生的可依賴性二值化判別圖



圖 3-12 Yang et al.[16]合成後產生破洞的影像結果(紅色區域)



圖 3-13 Yang et al.[16]最後影像合成結果

第4章 基於三角化轉換映射可降低瑕疵現象的 任意視點影像合成演算法設計

4.1 三角映射視點合成演算法概述 (Overview)

現今的自由視點顯影成像系統多是使用排列密集的一維攝影機陣列來擷取影像，多數的自由視點合成影像的演算法，通常紋理映射使用畫素對畫素的方式進行投影映射(pixel-by-pixel warping)。然而，在本篇論文中，我們使用的紋理映射方式是將三角形視為一個區域，並且使用三角形對三角形的紋理投影映射方式(Region-by-region warping)。如果可以精確重建三角化模型，三角形映射方式確實會得到一個較佳的合成影像品質[26][27][28]。因此事先選擇特徵點的步驟扮演著一個相當重要的角色，特徵點的位置好壞會影響三角形映射的目標影像品質。在本篇論文當中，我們也提出了一個有效的演算法架構並且獲取好的特徵點。

現今也有許多演算法是利用三角形轉換映射合成目標影像。B. Choi et al.[28]的視點合成演算法利用分群演算法將視差值分群成許多不同的區域，將這些不同的區域分成兩大類，分別是前景層及背景層的部份。然而，分群演算法的精確度是視情況而定並非是完美無誤的，因此若是測試影像的背景非常複雜，則容易發生分群失誤的問題。J. H. Park et al.[26][27]的演算法則是利用網點的概念來計算每塊三角形的能量方程式以去擷取最佳的特徵點，並且利用特徵點重建二維影像平面上的三角模型來產生新的景深圖。J. H. Park 的演算法實驗結果雖然已有不錯的成效，但並未針對視點合成的演算法提出改善的方法。

4.2 基於三角化映射的視點合成演算法(Triangular-based Warping View Synthesis Algorithm)

在本節的一開始，我們簡單描述我們所提出視點合成的演算法，其演算法的流程如錯誤! 找不到參照來源。所示，只針對影像的部份直接進行三角形的轉換紋理映射 (Triangular-based Texture Warping Algorithm)，簡稱 TBTW 演算法，其詳細步驟我們會在後面各節說明。

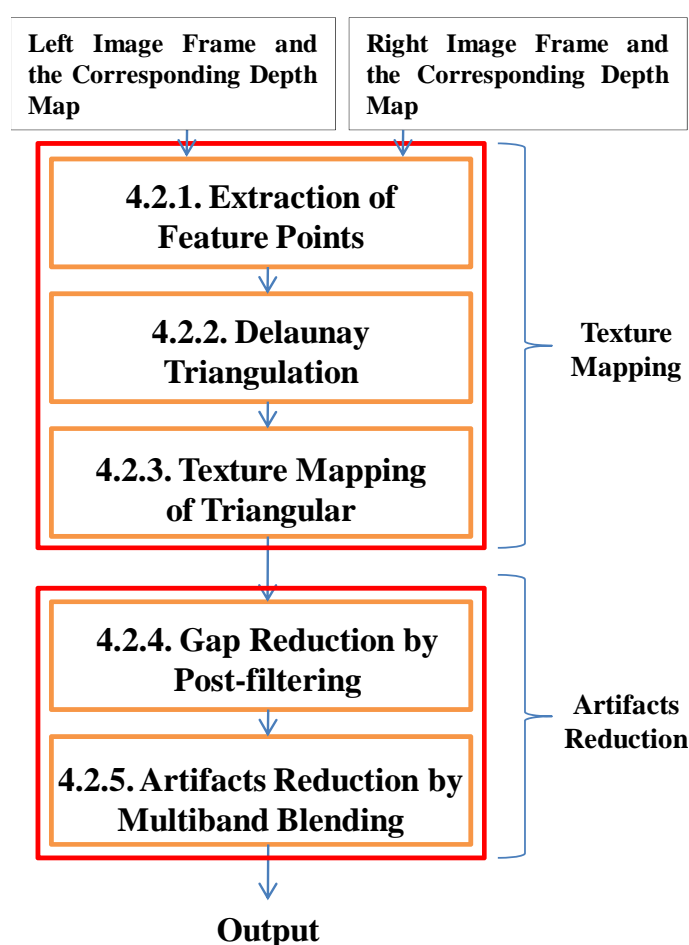


圖 4-1 TBTW 演算法設計流程圖

4.2.1 擷取特徵點 (Extraction of Feature Points)

首先，我們會計算影像灰階值的訊號強度與景深圖每個畫素之間的梯度值關

係。對於每個畫素的位置而言，當訊號之間的梯度值相關性越高，即代表越有可能為我們所需要的特徵點。

由於攝影機擷取影像的過程中容易產生雜訊干擾，所以我們會先將影像經過高斯平滑濾波器以抑制拍攝過程中所產生的雜訊。其中高斯分佈參數 σ 決定高斯函數的寬度。對於數位影像處理來說，我們常用二維離散高斯函數當作平滑濾波器，如(4.1)所示：

$$g(x) = e^{-\frac{x^2+y^2}{2\sigma^2}} \quad (4.1)$$

如同式(4.2)，我們先計算梯度的內積值， ∇ 代表梯度運算子，其中 D 代表景深圖， f 代表已經經過高斯平滑濾波器的灰階影像值，如同式(4.3)所示：

$$E(D, f) = |\nabla D \cdot \nabla f| \quad (4.2)$$

$$f = g_{mask} * F \quad (4.3)$$

其中 g_{mask} 代表高斯平滑濾波器、 F 為原始灰階影像。 ∇D 與 ∇f 均為向量，當畫素位置的 $E(D, f)$ 大於週遭鄰近視窗 M 內的 $E(D, f)$ 值以及大於設定的臨界值，代表 ∇D 與 ∇f 本身數值均相當大，則我們即將此畫素的位置判定為特徵點，其示意圖如圖 4-2 所示[26]。(在此演算法實現的過程中，我們將臨界值設定為 200、期視窗 M 大小設為 8。)

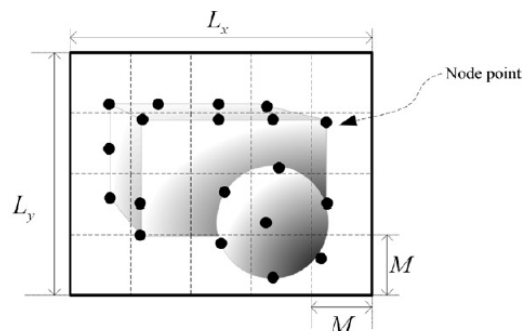


圖 4-2 擷取特徵點示意圖

4.2.2 特徵點三角化 (Delaunay Triangulation)

如同圖 4-3 所示，我們使用 Delaunay 三角化的方式來重建二維影像上的模型。Delaunay 三角化的特性是給定一組任意的特徵點集合，即可決定一組唯一獨立的三角形，其三角形的內角盡可能越大越好。換句話說，Delaunay 三角化可以採用一組特徵點集合，而不需要額外的連結資訊即可精確重建三角化模型 [26][27]，特徵點的數目對於合成結果的影響將在節 5.4.1.2 討論。

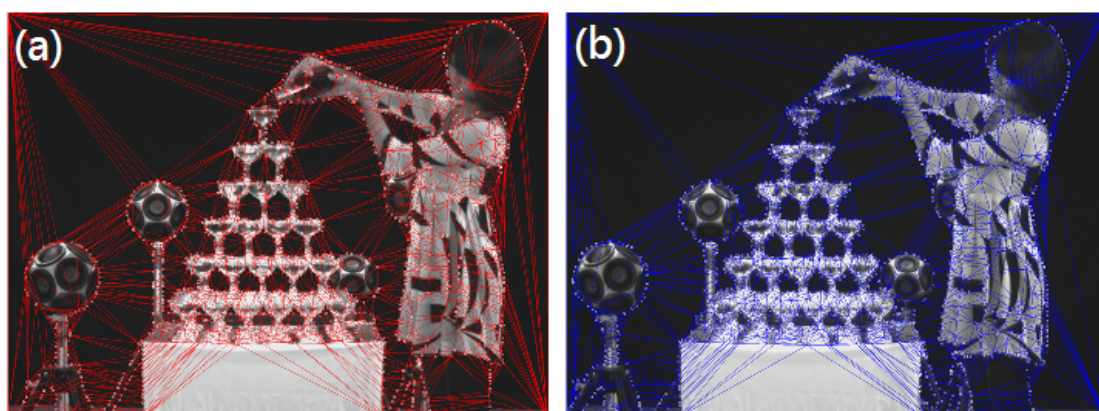


圖 4-3 利用 Delaunay 三角化的方式將特徵點連結：測試影像 “Champagne Tower”
(a)視點 39 (特徵點數目：1496) (b)視點 41 (特徵點數目：1658)

4.2.3 三角化紋理投影映射 (Texture Mapping of Triangular)

傳統的自由視點合成演算法設計架構中，通常紋理投影映射的部分大多都是使用畫素對畫素的方式進行投影映射(pixel-by-pixel warping)。然而，在本章節中，我們使用的紋理映射方式是將三角形視為一個區域，並且進而使用三角形對三角形的紋理投影映射方式(Region-by-region warping)。

對於每塊三角形而言，針對每塊三角形中的畫素位置上的視差值進行統計運算，並挑選出現機率最高的視差值當作此三角形區域的視差值，因此，景深預估的運算精確度在自由視點合成的演算法中是非常重要的，現今有許多效能不錯的演算法已經被提出，例如：動態規劃路徑演算法、圖形切割演算法[1]。然而，

目前景深預估運算出來的景深圖邊緣處仍然有許多錯誤資訊，同時也會造成在合成端的運算結果產生肉眼可見的瑕疵現象。

統計運算完三角形的視差值後，每塊三角形根據不同的視差值投影映射至目標影像平面對應的畫素位置上，影像平面上的視差值與景深值的對應關係圖如圖 4-4 所示。

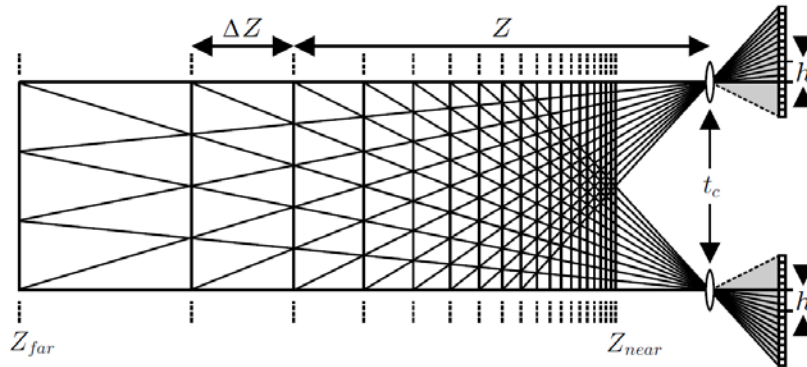


圖 4-4 在 Z_{near} 與 Z_{far} 兩個夾擊平面中預測最佳化景深量化值示意圖[12]

我們可以推導出其對應關係式，如式(4.4)所示：

$$\frac{1}{Z} = \frac{v}{255} \left(\frac{1}{Z_{near}} - \frac{1}{Z_{far}} \right) + \frac{1}{Z_{far}} \quad (4.4)$$

其中 Z_{near} 代表離攝影機位置最近的深度平面、代表 Z_{far} 離攝影機位置最遠的深度平面、 v 代表視差值的灰階訊號值(範圍介於 0 到 255 間)，更詳細的推導過程可以參考[12]。

在紋理投影映射的過程中，針對擁有最小視差值的三角形陸續進行目標影像平面的投影映射，直到較大視差值的三角形映射完成為止。當兩個三角形有重疊現象發生的時候，則顯示較大視差值的三角形，因此我們可以確定在前景的三角形對於在目標影像上相對於背景的三角形擁有較高的優先權。在目標合成影像中，沒有被投影映射到的某些區域是由於遮蔽效應的關係所產生的結果，如圖 4-5 所示[28]。

在此演算法執行步驟中，我們可以獲得兩張分別由左邊以及右點視點投影至

中間視點的合成影像。由圖 4-5 明顯看到此兩張影像因為視點角度的關係，導致合成影像上有許多細小的裂痕跟無法預測的遮蔽區域。在演算法的後續處理步驟中，我們使用尚未映射前的影像上較可靠的資訊來修復映射產生的細小裂痕。在後續兩小節中將介紹如何修復被遮蔽的未知區域資訊。

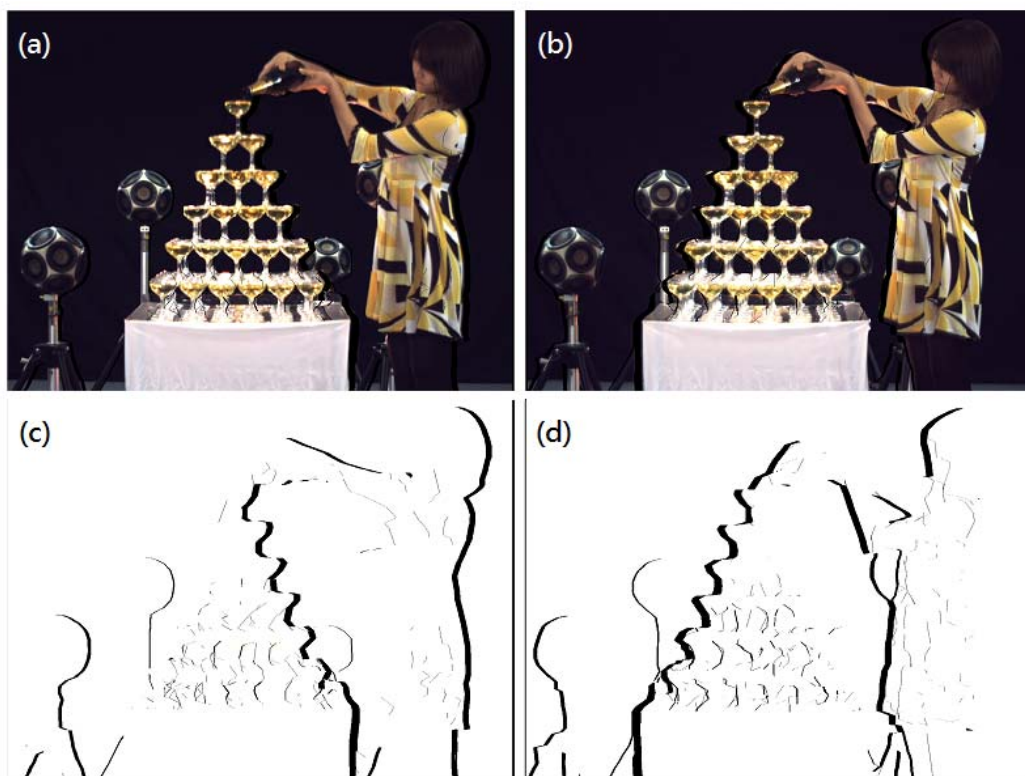


圖 4-5 測試影像 “Champagne Tower”：(a) 由視點 39 所合成的虛擬視點 40 的目標影像 (b) 由視點 41 所合成的虛擬視點 40 的目標影像 (c) 視點 39 合成虛擬視點 40 所造成的遮蔽區域 (d) 視點 41 合成虛擬視點 40 所造成的遮蔽區域

4.2.4 使用後製濾波器修復裂痕 (Gap Reduction by Post-filtering)



圖 4-6 經過中值濾波法修復細小的遮蔽區域

經由三角化紋理投影映射之後，大多都會產生細小的裂痕，而我們解決此問題的方法是透過中值濾波法修復此瑕疵現象。在圖 4-6 中，(a)的圖片是尚未經過製濾波器處理過後的原圖，(b)是經過中值濾波器處理之後的圖片。很明顯地，經由處理後的影像中，原本細小的裂痕現象減少了。

4.2.5 利用多率頻段疊合抑制瑕疵現象 (Artifacts Reduction by Using Multi-band Blending)

在視點合成的過程中，我們嘗試採用多率頻段疊合的方式來抑制瑕疵現象 (Multi-band Blending)，簡稱 MB。多率頻段疊合是一種經常應用在環場成像的技術[13]。以頻域的觀點來看，多率頻段疊合的優點能夠使較低頻帶成分擁有較大的時域解析度，則可以保留較高頻段成分的細節而不會模糊。

在擷取影像的過程中，理想上每個畫素通過每條光射線都應該擁有相同的訊號強度，但實際上真實的拍攝場景並無法達到如此的效果，使得影像在疊合的過程中會導致肉眼可見的色差問題，使用多率頻段疊合的技巧則可以幫助我們擁有較佳的視覺感受。

經過高通濾波器的成像影像可以表示如下式(4.5)所示：

$$B_{\sigma} = I - I_{\sigma} \quad (4.5)$$

$$I_{\sigma} = G_{\sigma} * I \quad (4.6)$$

經由式(4.6)運算，原始影像 I 經過高斯濾波器處理後可得到目標影像 I_{σ} ，其中 G_{σ} 是一個具標準差 σ 的高斯運算子、 B_{σ} 表示為一個主要落入介於波長 $\lambda \in [0, \sigma]$ 的空間頻率訊號。在計算被遮蔽的權重函數的分布比例圖 M ，我們利用式(3.6)找出投影映射後的合成視點影像上被遮蔽所產生的空洞位置。為了找出適合經過多率頻段處理的邊緣區域，因此我們選擇使用影像型態學(morphology)中的侵蝕(erosion)以及膨脹運算子(dilation)的技術找出邊緣區域，如式(4.7)所示， D 代表膨脹運算

子、 E 代表侵蝕運算子。

$$M = D * D * Occ(u, v) - E * Occ(u, v) \quad (4.7)$$

$$M_\sigma = G_\sigma * M \quad (4.8)$$

接著將被遮蔽區域的權重函數分布比例圖經過高斯運算子後， M_σ 代表是主要落入介於波長 $\lambda \in [0, \sigma]$ 頻段區間內的權重函數。因此，利用相同概念套用到其他的子頻帶上，如式(4.9)與式(4.10)，利用較低頻率的帶通濾波器以及針對不同頻段區間內的權重比例函數去做運算，即 $k \geq 1$ ，其中 k 代表頻段的索引數。

$$B_{(k+1)\sigma} = I_{k\sigma} - I_{(k+1)\sigma} \quad (4.9)$$

$$I_{(k+1)\sigma} = G_{\sigma'} * I_{k\sigma} \quad (4.10)$$

$$M_{(k+1)\sigma} = G_{\sigma'} * M_{k\sigma} \quad (4.11)$$

其中經過標準差為 $\sigma' = \sqrt{2k+1}\sigma$ 的高斯運算子運算之後，使得其他子頻帶也能夠落入相同頻帶區間內的波長。

針對每個不同的頻段，利用每張圖片所對應的權重函數分布比例將不同頻段的圖片作線性疊合，並且將最後的影像輸出，如式(4.12)所示：

$$I_k^{multiband} = \sum_{i=1}^n M_{k\sigma}^i B_{k\sigma}^i \quad (4.12)$$

$$I^{multiband} = \sum_{k=1}^m I_k^{multiband} \quad (4.13)$$

多率頻段疊合保留高頻段的訊號在較小的時域範圍內(較小的 $k\sigma$)、低頻段的訊號在較大的時域範圍內(較大的 $k\sigma$)。如此一來，將影像疊合的過程中使得高頻段的影像在線性疊合之後，不會有鬼影的現象發生。若以目標影像拆成三個不同的子頻帶為例，圖 4-7 是三個子頻帶的多率頻段疊合範例，即 $k = 1, 2, 3$ 。

圖 4-7 (a) 表示在第一個頻段區間內($k = 1$)的目標影像 B_σ ，其尺度介於 0 到 σ 的情況下所對應的權重函數 M_σ ，如圖 4-7 (d) 所示。圖 4-7 (b) 表示在第二個頻段

區間內($k=2$)的目標影像 $B_{2\sigma}$ ，其尺度介於 σ 到 2σ 的情況下所對應的權重函數 $M_{2\sigma}$ ，如圖 4-7 (d)所示。圖 4-7 (c)表示在第三個頻段區間內($k=3$)的目標影像 $B_{3\sigma}$ ，其尺度介於 2σ 到 3σ 的情況下所對應的權重函數 $M_{3\sigma}$ ，如圖 4-7 (d)所示。

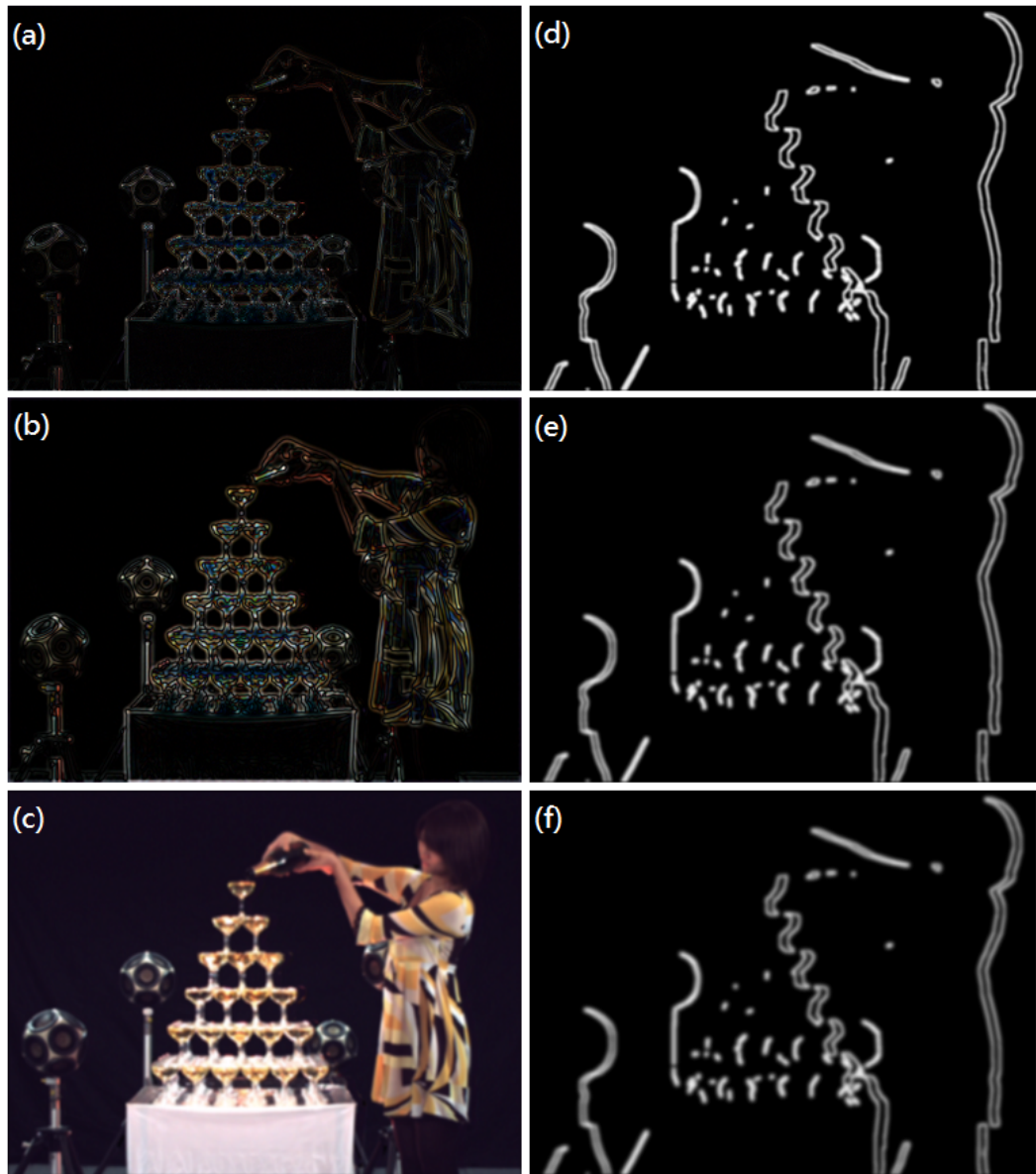


圖 4-7 擁有三個不同子頻帶的多率頻段疊合範例。其中 $k=1, 2, 3$ 的 $B_{k\sigma}$ 如同左邊影像所示，右邊影像代表相對應被遮蔽區域的權重函數

4.3 模擬與結果 (Simulation and Results)

4.3.1 實驗環境設定

在本章節，介紹我們如何設計實驗流程以設定環境參數、並比較我們所提出的演算法架構與傳統的視點合成演算法架構的效能差異。

表 5 VSRS 2.0 環境參數設定

Order	Parameter Name	Meaning
1	<i>flag</i>	Depth Type
2	<i>width</i>	Width Of View Image
3	<i>height</i>	Height Of View Image
4	<i>Z_near_L</i>	Nearest Depth Value Of Left View
5	<i>Z_far_L</i>	Farthest Depth Value Of Left View
6	<i>Z_near_R</i>	Nearest Depth Value Of Right View
7	<i>Z_far_R</i>	Farthest Depth Value Of Right View
8	<i>num_frame</i>	Total Number Of Frames
9	<i>name_cam_param</i>	Camera parameter file name
10	<i>name_cam_L</i>	Left Camera Name
11	<i>name_cam_V</i>	Virtual Camera Name
12	<i>name_cam_R</i>	Right Camera Name
13	<i>name_src_L</i>	Left View Image Name
14	<i>name_src_R</i>	Right View Image Name
15	<i>name_depth_L</i>	Left Depth Map Name
16	<i>name_depth_R</i>	Right Depth Map Name
17	<i>name_out</i>	Output Virtual View Image Name
18	<i>getSynthesisMode</i>	Synthesis Mode
19	<i>getColorSpace</i>	Color Space
20	<i>getPrecision</i>	Precision
21	<i>getFilter</i>	Filter

針對不同的測試影像在參考軟體的平台上模擬實驗結果以及數據，這些測試

影像分別為“Kendo”、“Champagne tower”、“Book arrival”、“Newspaper”(所有測試影像皆包含原始圖檔的景深圖以及實際拍攝視角的影像)。我們的模擬結果將針對 PSNR、主觀視覺以及擷取特徵點的點數數量在 PSNR 上的表現，來加以分析與討論其實驗結果。我們實驗的環境設定與 VSRS 2.0 是相同的，更詳細的參考軟體環境參數設定的方法可以參考[19]。

4.3.2 實驗模擬結果

實驗過程我們共分成三大部分探討與說明，並在本章節的最後，將會針對不同實驗部分的結果影像詳加討論其結果並加以探討與說明：

- 實驗的第一部分我們將利用 MPEG 測試影像中所提供的景深圖當作是輸入測試影像的對應景深圖，進而觀察其最後顯示結果的表現。
- 實驗的第二部份我們利用 MPEG 所提供的測試軟體 DERS 4.9 來產生測試影像的對應景深圖[23]，進而觀察其最後顯示結果的表現。
- 實驗的第三部份我們利用 MPEG 所提供的測試軟體 DERS 4.9 來產生測試影像的對應景深圖，並且針對攝影機的擺放位置不同以觀察其最後顯示結果的主觀視覺優劣性以及平均 PSNR 數值的表現。

最後，我們將會針對不同的視點合成演算法詳加討論及分析視點合成影像的結果。

4.3.2.1 實驗第一部份：MPEG 測試影像提供的景深圖對於合成結果的表現

在實驗第一部分我們將利用 MPEG 測試影像中所提供的景深圖當作是輸入測試影像的對應景深圖並且觀察其成像結果，其輸入規格如表 6 所示。

表 6 MPEG 測試影像以及對應景深圖






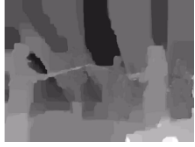


Data set	Champagne tower	Kendo	Book arrival	Newspaper
Input image				
Corresponding Depth map				
Original Viewpoint Pair (OL-OR)	39-41	03-05	08-10	04-06
Synthesized viewpoint	40	04	09	05



圖 4-8 測試影像 “Newspaper” 由視點 04 與視點 06 所合成的虛擬視點 05 結果比較圖

Test sequences: Newspaper (GIST)			
Ground truth		VSRS 2.0 (Integral pixel)	
VSRS 2.0 (Half pixel)		VSRS 2.0 (Quarter pixel)	
TBTW + MB		TBTW	

圖 4-9 測試影像 “Newspaper” 由視點 04 與視點 06 所合成的虛擬視點 05 細部結果放大比較圖

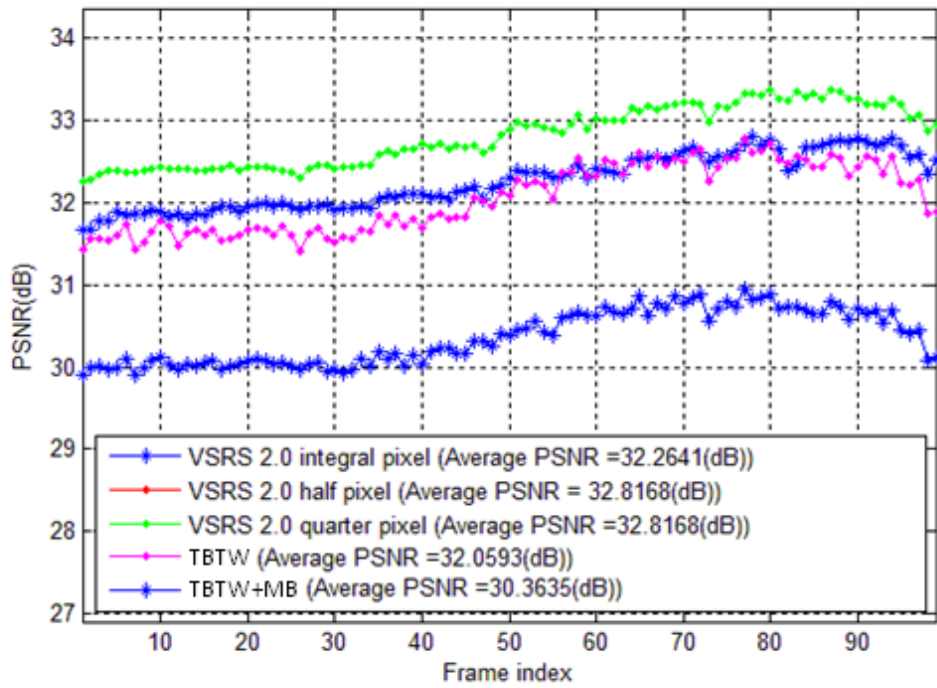


圖 4-10 不同的視點合成演算法在測試影像“Newspaper”視點 05 的前 100 張影像 PSNR 表現

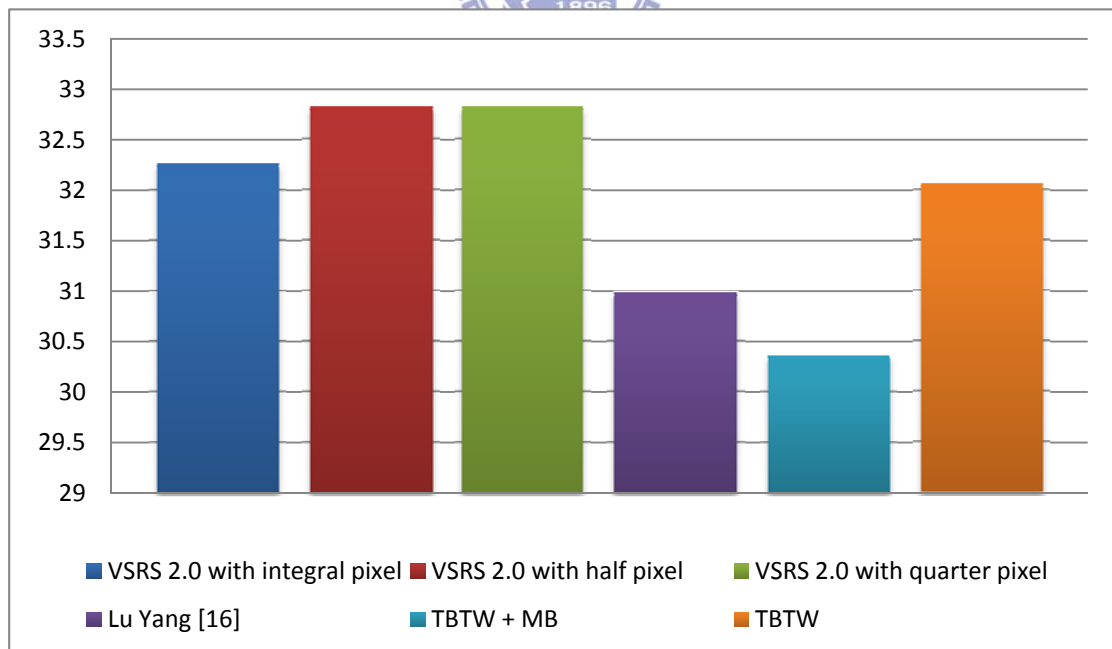


圖 4-11 不同的視點合成演算法針對測試影像“Newspaper”視點 05 的前 100 張影像上的平均 PSNR 表現



圖 4-12 測試影像 “Kendo” 由視點 03 與視點 05 所合成的虛擬視點 04 結果比較圖



圖 4-13 測試影像 “Kendo” 由視點 03 與視點 05 所合成的虛擬視點 04 細部放大結果比較圖

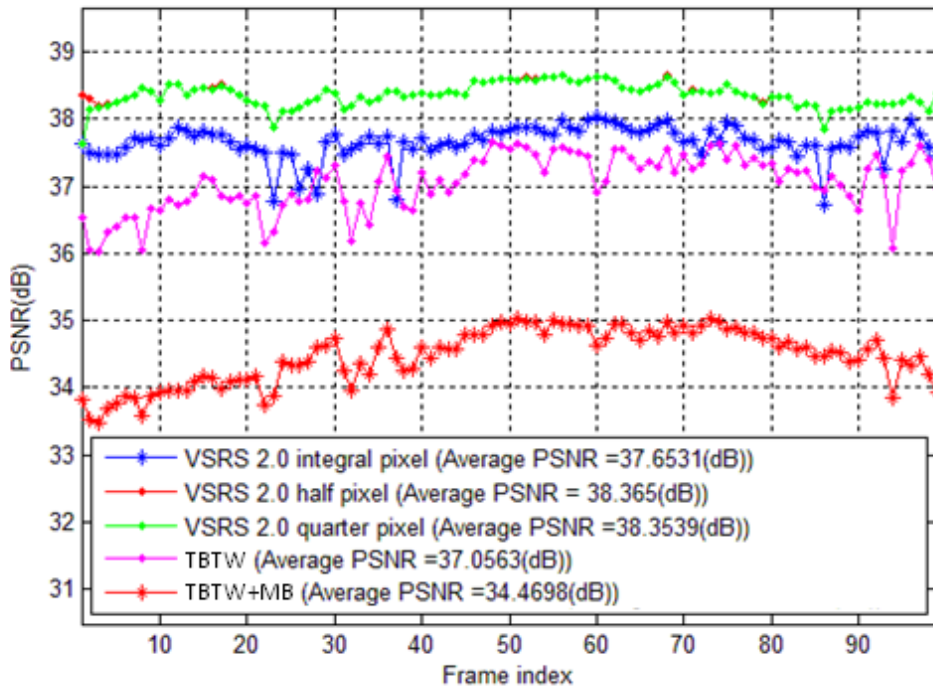


圖 4-14 不同的視點合成演算法在測試影像 “Kendo” 視點 04 的前 100 張影像 PSNR 表現

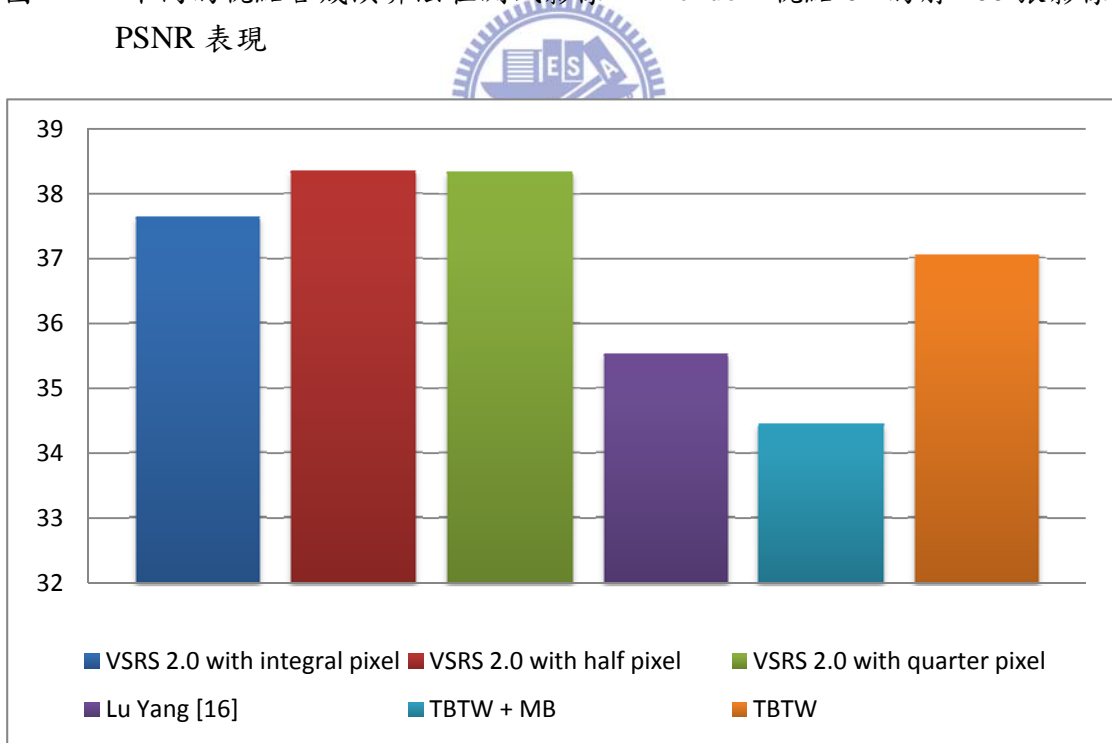


圖 4-15 不同的視點合成演算法針對測試影像 “Kendo” 視點 04 的前 100 張影像上的平均 PSNR 表現

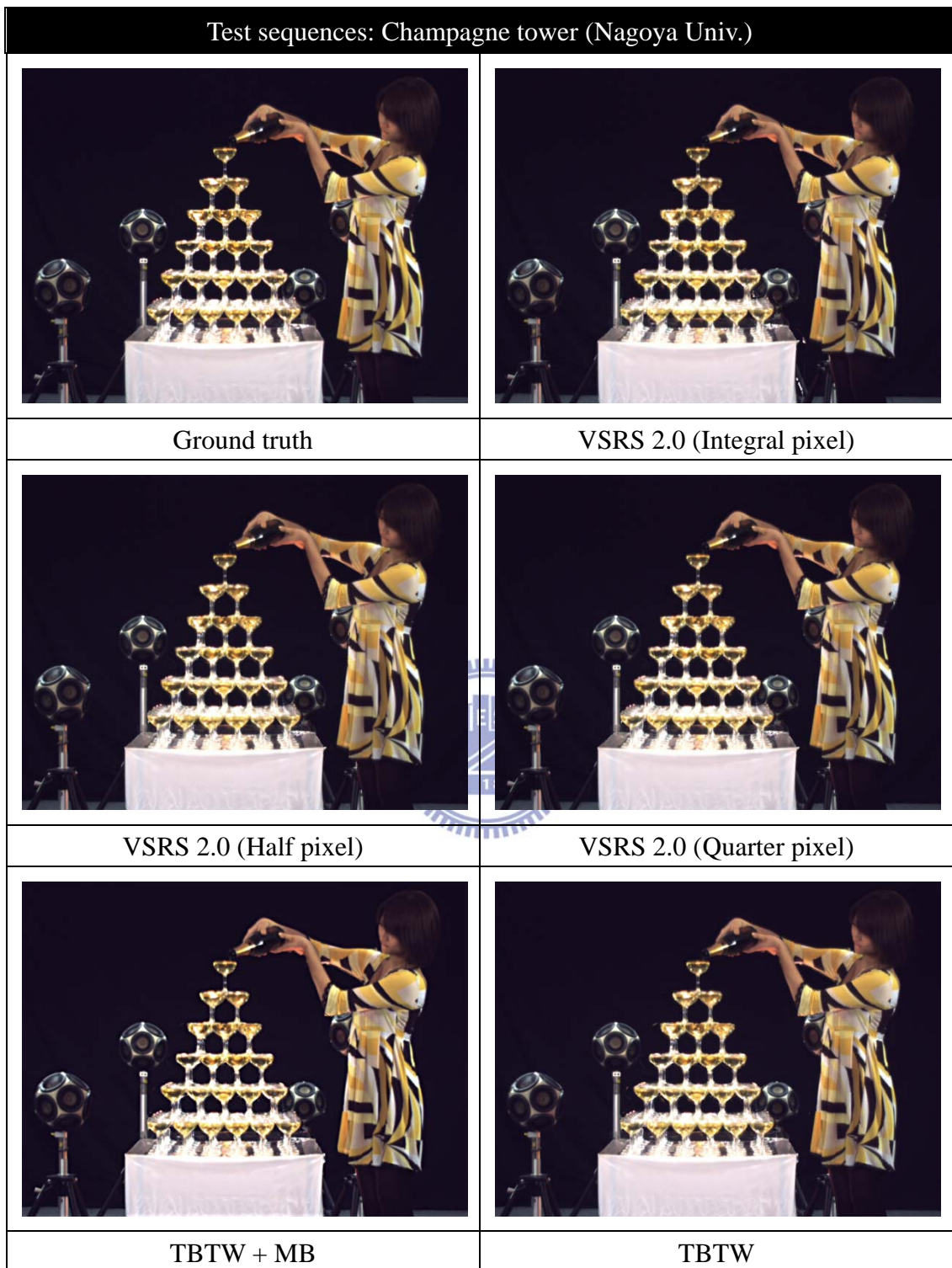


圖 4-16 測試影像 “Champagne tower” 由視點 39 與視點 41 所合成的虛擬視點 40 結果比較圖

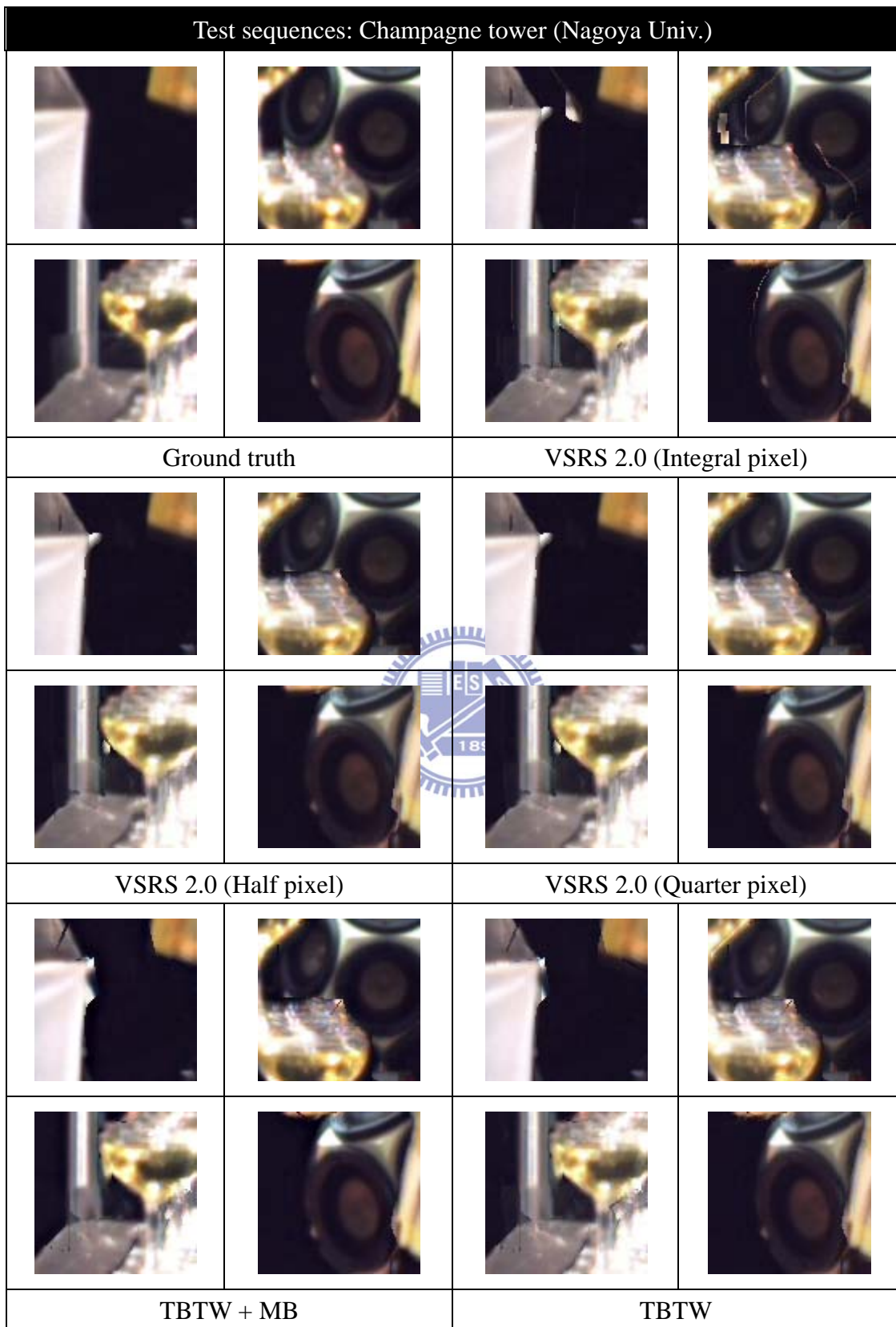


圖 4-17 測試影像 “Champagne tower” 由視點 39 與視點 41 所合成的虛擬視點 40 細部放大結果比較圖

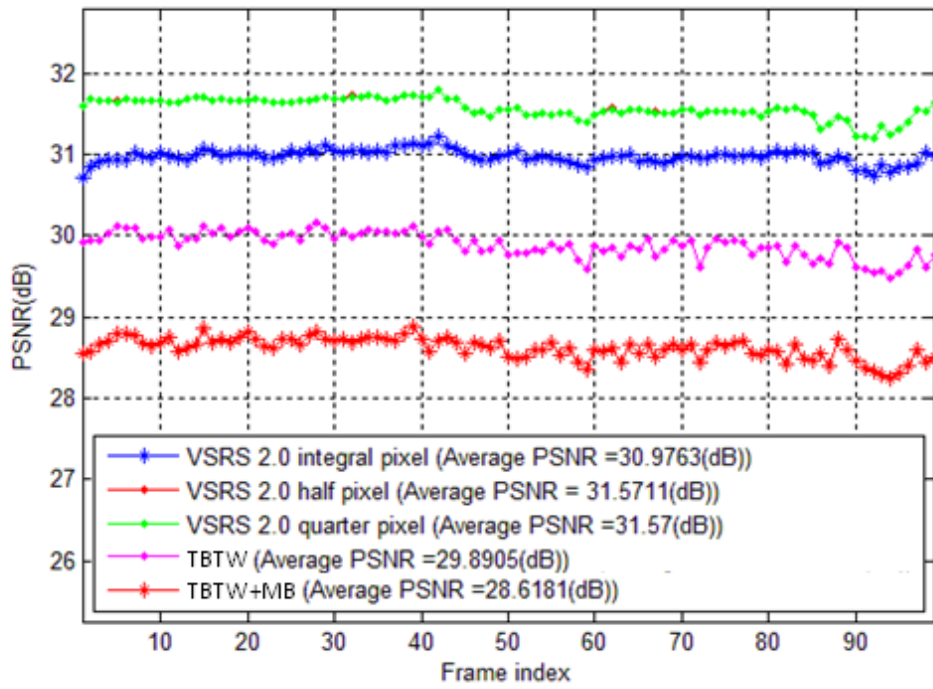


圖 4-18 不同的視點合成演算法在測試影像 “Champagne tower” 視點 40 的前 100 張影像 PSNR 表現

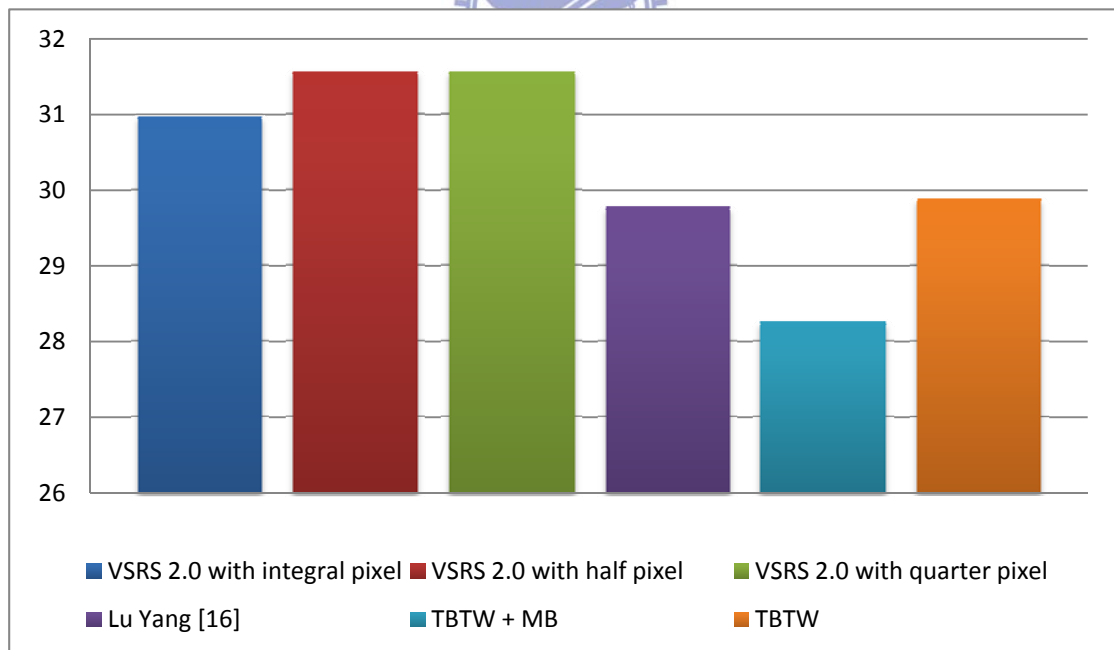


圖 4-19 不同的視點合成演算法針對測試影像 “Champagne tower” 視點 40 的前 100 張影像上的平均 PSNR 表現



圖 4-20 測試影像 “Book arrival” 由視點 08 與視點 10 所合成的虛擬視點 09 結果比較圖

Test sequences: Book Arrival (HHI)							
							
Ground truth				VRS 2.0 (Integral pixel)			
							
VRS 2.0 (Half pixel)				VRS 2.0 (Quarter pixel)			
							
TBTW + MB				TBTW			

圖 4-21 測試影像 “Book arrival” 由視點 08 與視點 10 所合成的虛擬視點 09 細部放大結果比較圖

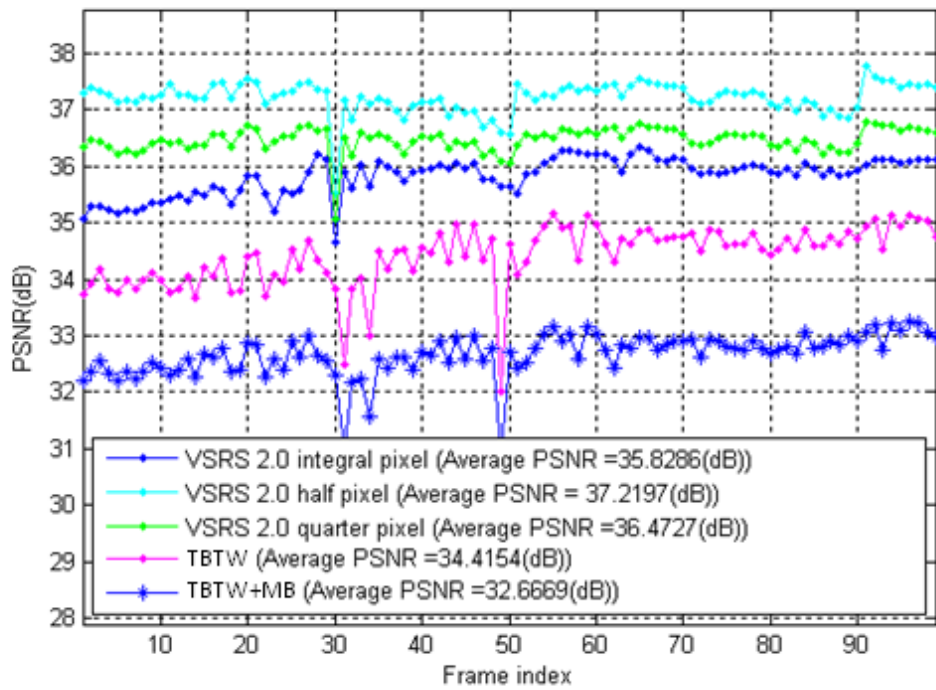


圖 4-22 不同的視點合成演算法在測試影像 “Book arrival” 視點 09 的前 100 張影像 PSNR 表現

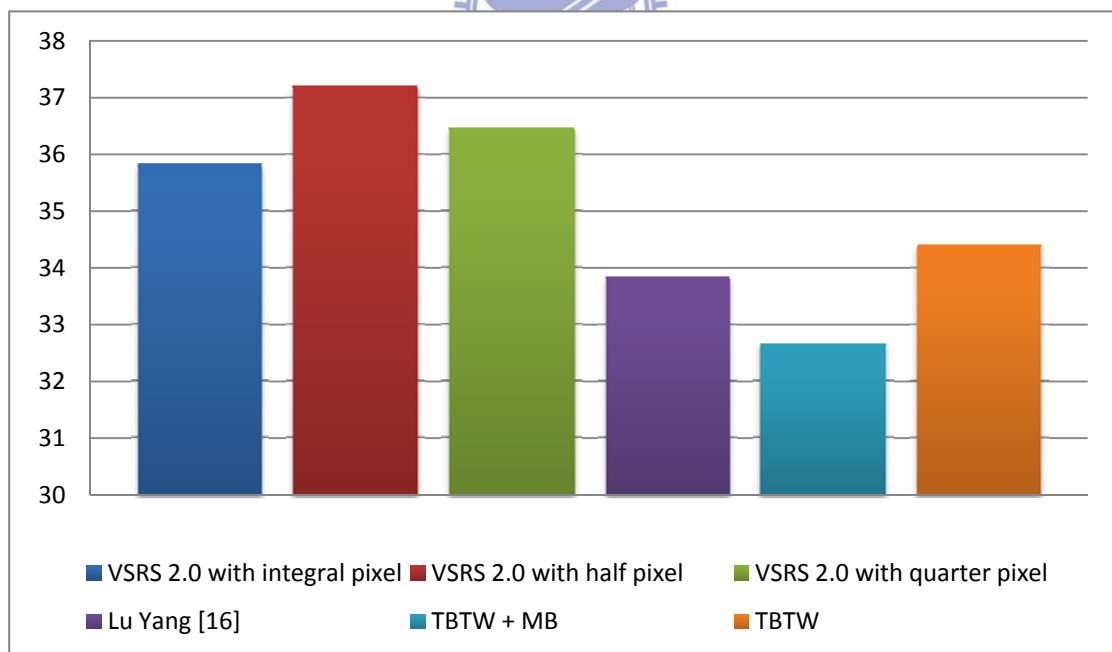


圖 4-23 不同的視點合成演算法針對測試影像 “Book arrival” 視點 09 的前 100 張影像上的平均 PSNR 表現

分析與討論

表 7 表示我們分別針對 MPEG 所提供四種不同的 FTV 測試影像：“Book arrival”、“Champagne tower”、“Newspaper”以及“Kendo”的前 100 張影像進行視點合成演算法的平均 PSNR 的結果。事實上，可以發現我們使用多率頻段疊合合成影像的影像品質在 PSNR 上的表現並非最好。然而，有別於一般傳統的畫素對畫素投影映射結果，三角形對三角型投影映射所產生影像品質的 PSNR 結果與傳統的演算法實際上在 PSNR 數值上並無太大的差異。

在四個 MPEG 所提供的測試影像中，針對測試影像“Newspaper”以及“Kendo”而言，參考軟體 VSRS 2.0 以整數為單位畫素的合成影像結果，可以明顯發現在人的頭部邊緣附近以及報紙上紋理的部分容易產生一些瑕疵現象，如圖 4-9 與圖 4-13 所示。另一方面，在影像播放的過程中產生影像閃爍的現象，使得肉眼視覺感受十分容易查覺。事實上，與我們所提出的演算法相比，主觀的肉眼觀看感受與平均 PSNR 的數值在這個測試影像的結果差異並不大。

對於測試影像“Champagne tower”而言，圖 4-16 表示為 TBTW 演算法以及參考軟體 VSRS 2.0 以整數單位畫素的合成結果。合成影像在酒杯的邊緣附近以及酒杯上較具紋理的部分容易產生一些肉眼可見的不連續現象，如圖 4-17 所示。與我們所提出的 TBTW 演算法相比，由於三角形對三角形映射的過程中，是將整個三角形區域內的紋理直接映射，未做任何修改。因此最後將左右前向映射後的參考影像做線性疊合時，酒杯邊緣具有紋理的部分非常容易產生模糊不清的問題。這是造成我們所提出的演算法架構對於測試影像“Champagne tower”的平均 PSNR 數值低落的主因。從圖 4-19 中，我們的平均 PSNR 數值與[16]是差不多的。

對於測試影像“Book arrival”而言，圖 4-20 表示為 TBTW 演算法以及參考軟體 VSRS 2.0 以整數單位畫素的視點 09 合成結果。合成影像在人的頭部附近以及桌椅較細小的握把部分，容易產生一些肉眼可見的不連續現象，如圖 4-21 所示。對於主觀視覺而言，MPEG 測試影像“Book arrival”的場景中，椅子仍然有

許多肉眼可見的不連續性產生。其原因在於我們的演算法使用特徵點的方式，若要切割較細小的物體是十分困難的，使得測試影像“Book arrival”對於 PSNR 的表現較為差。

表 7 MPEG 測試影像前 100 張合成影像的平均 PSNR 效能表現

Test sequences	Newspaper	Champagne tower	Kendo	Book Arrival
Our proposed method				
TBTW + MB	30.36	28.62	34.47	32.67
TBTW	32.06	29.89	37.06	34.42
VSRS 2.0				
VSRS 2.0 with integral pixel	32.26	30.98	37.65	35.83
VSRS 2.0 with half pixel	32.82	31.57	38.36	37.22
VSRS 2.0 with quarter pixel	32.82	31.57	38.35	36.47
Others				
Lu Yang [16]	30.98	29.78	35.54	33.85

針對物體邊緣處容易產生一些不連續性的瑕疵現象，對於 VSRS 2.0 而言使用以二分之一畫素或四分之一畫素為單位合成影像的結果，可以發現效果相當良好，相較於以整數為畫素單位的合成影像而言，其計算的平均 PSNR 值成長約 0.5 到 1.4dB 左右，且擁有較佳的主觀視覺品質。

我們發現在合成的虛擬視點影像中，有時候會出現肉眼觀察測試影像的結果雖然並未較佳，但是 PSNR 的表現卻反而較好的情況。這有可能是因為使用疊合的合成影像場景較為單純。與傳統的視點合成演算法比較而言，三角形對三角型投影映射的演算法在 PSNR 上的普遍表現通常與傳統合成演算法相近。

4.3.2.2 實驗第二部份：景深圖的差異性對於合成結果的表現

第二部份我們利用 MPEG 所提供的測試軟體 DERS 4.9 來產生 FTV 測試影像 “Champagne tower” 、 “Kendo” 、 “Book arrival” 以及 “Newspaper” 的對應景深圖[23]，如表 8 所示。

我們利用不同的景深圖對於不同的視點合成演算法，觀察對於演算法的合成表現，並針對四種不同的測試影像前 30 張測量其 PSNR 的數值加以討論與分析。

表 8 DERS 4.9 產生的景深圖以及對應的視點位置






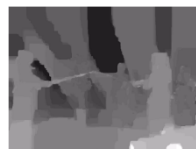






Data set	Champagne tower	Kendo	Book arrival	Newspaper
Input Images				
Experiment part I. (Depth maps by MPEG + VS methods)				
Corresponding Depth map				
Experiment part II. (Depth maps by DERS 4.9 + VS methods)				
Corresponding Depth Maps				
Original Viewpoint Pair (OL-OR)	39-41	03-05	08-10	04-06
Synthesized Viewpoint	40	04	09	05



圖 4-24 測試影像 “Newspaper” 利用 DERS 4.9 產生的景深圖由視點 04 與視點 06 所合成的虛擬視點 05 結果比較圖



Test sequences: Newspaper (GIST)			
			
			
Ground truth		DERS4.9+VSRS 2.0 (Integral pixel)	
			
			
DERS4.9+VSRS 2.0 (Half pixel)		DERS4.9+VSRS 2.0 (Quarter pixel)	
			
			
DERS4.9 + TBTW + MB		DERS4.9 + TBTW	

圖 4-25 測試影像 “Newspaper” 利用 DERS 4.9 產生的景深圖由視點 04 與視點 06 所合成的虛擬視點 05 細部結果放大比較圖

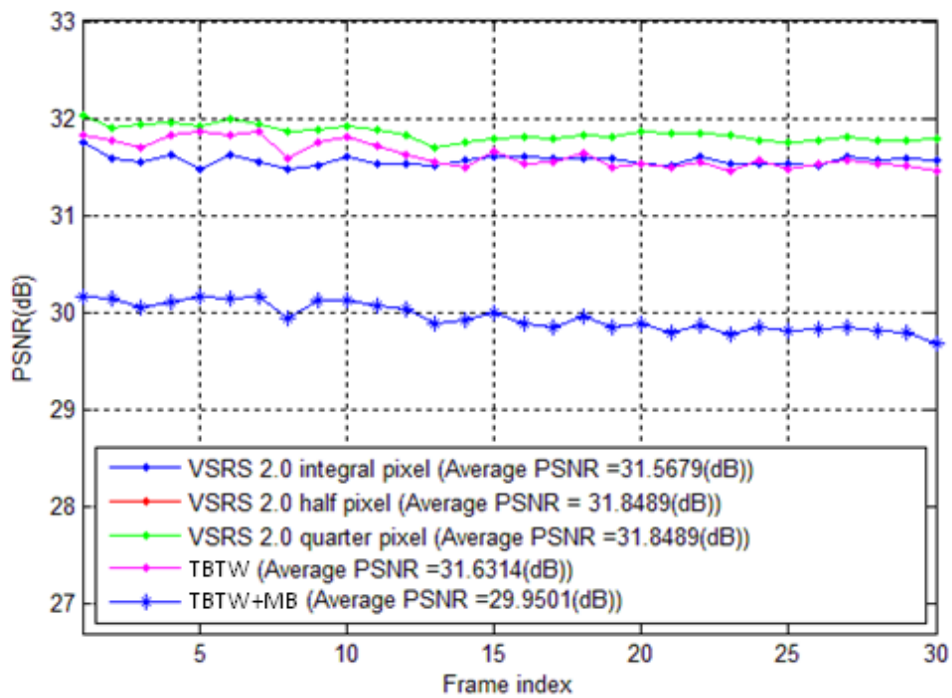


圖 4-26 DERS 4.9 產生的景深圖對於視點合成演算法在測試影像 “Newspaper” 視點 05 的前 30 張影像 PSNR 表現

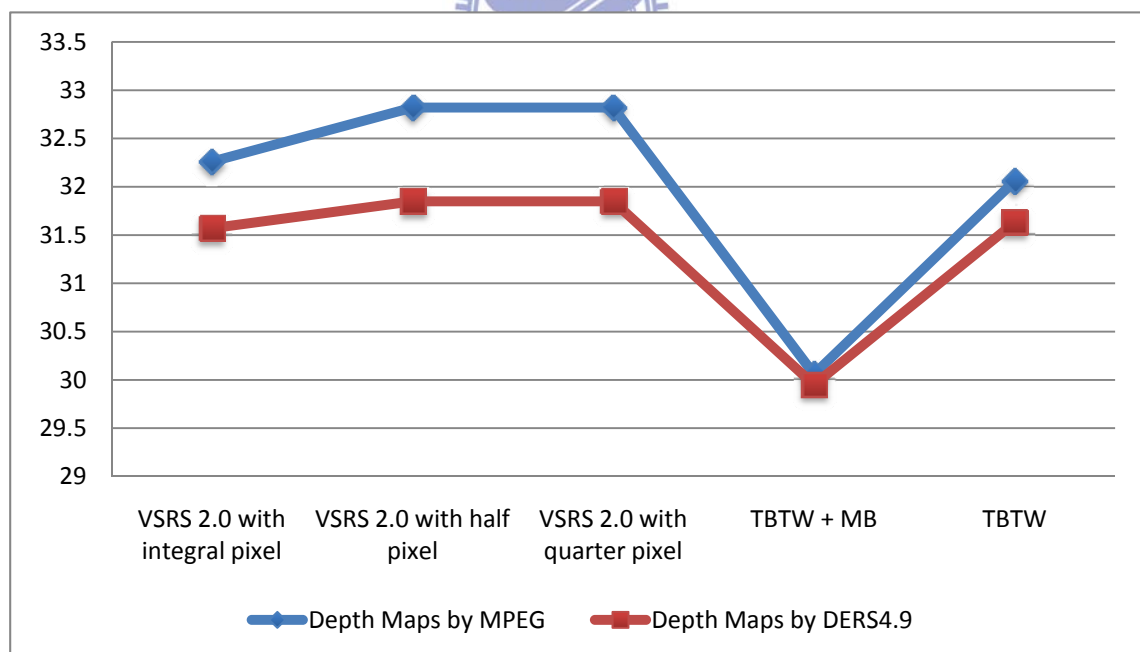


圖 4-27 DERS 4.9 產生的景深圖對於視點視點合成演算法針對測試影像 “Newspaper” 視點 05 的前 30 張影像上的平均 PSNR 表現

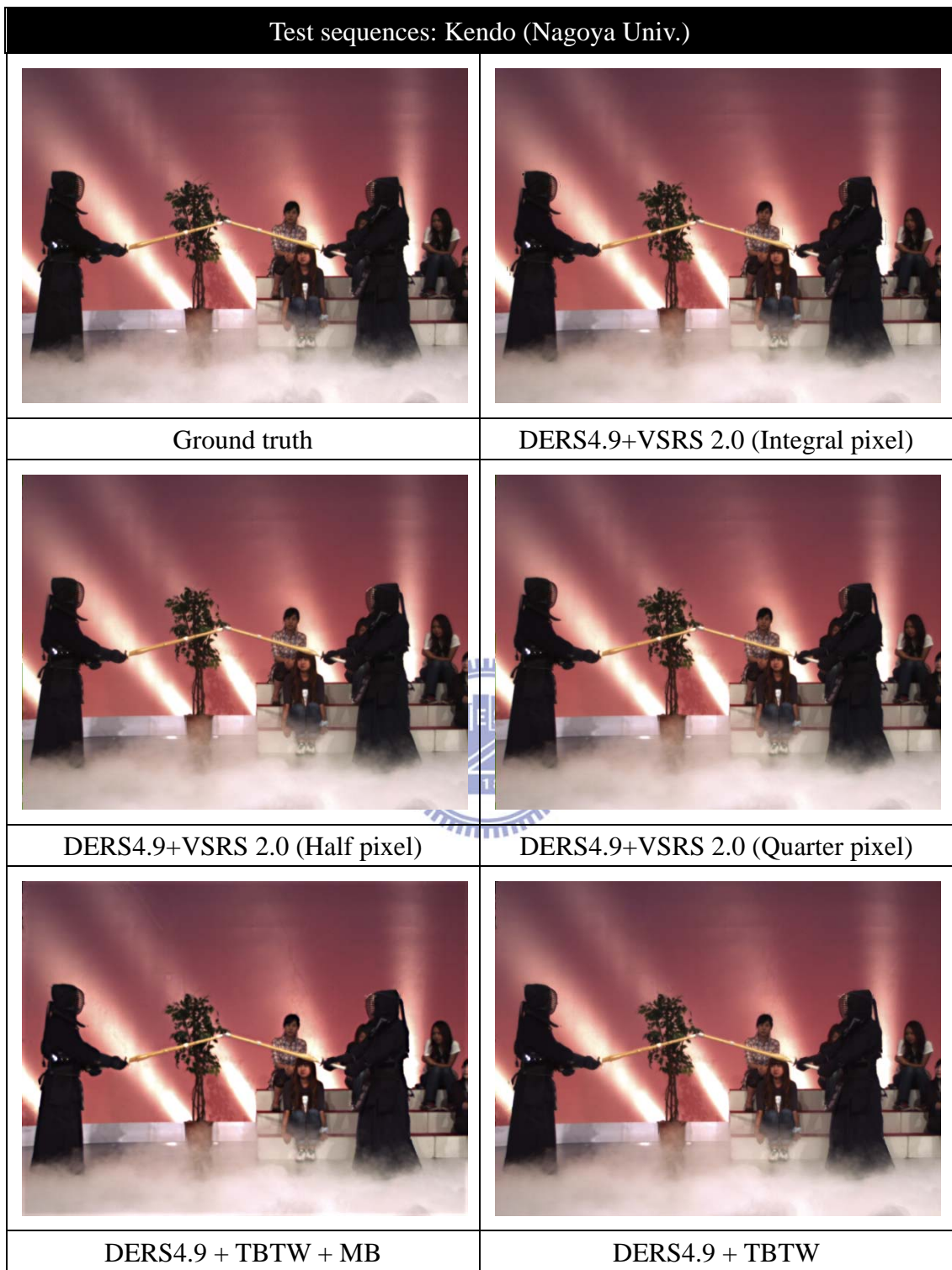


圖 4-28 測試影像 “Kendo” 利用 DERS 4.9 產生的景深圖由視點 03 與視點 05 所合成的虛擬視點 04 結果比較圖

Test sequences: Kendo (Nagoya Univ.)							
Ground truth				DERS4.9+VSRS 2.0 (Integral pixel)			
DERS4.9+VSRS 2.0 (Half pixel)				DERS4.9+VSRS 2.0 (Quarter pixel)			
DERS4.9 + TBTW + MB				DERS4.9 + TBTW			

圖 4-29 測試影像 “Kendo” 利用 DERS 4.9 產生的景深圖由視點 03 與視點 05 所合成的虛擬視點 04 細部放大結果比較圖

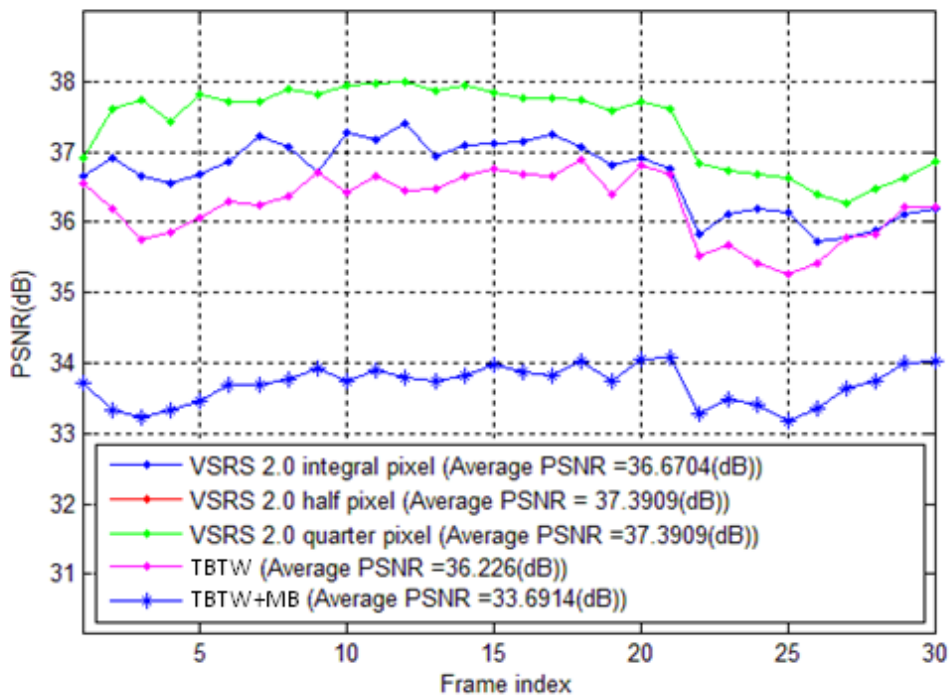


圖 4-30 DERS 4.9 產生的景深圖對於視點視點合成演算法在測試影像 “Kendo” 視點 04 的前 30 張影像 PSNR 表現

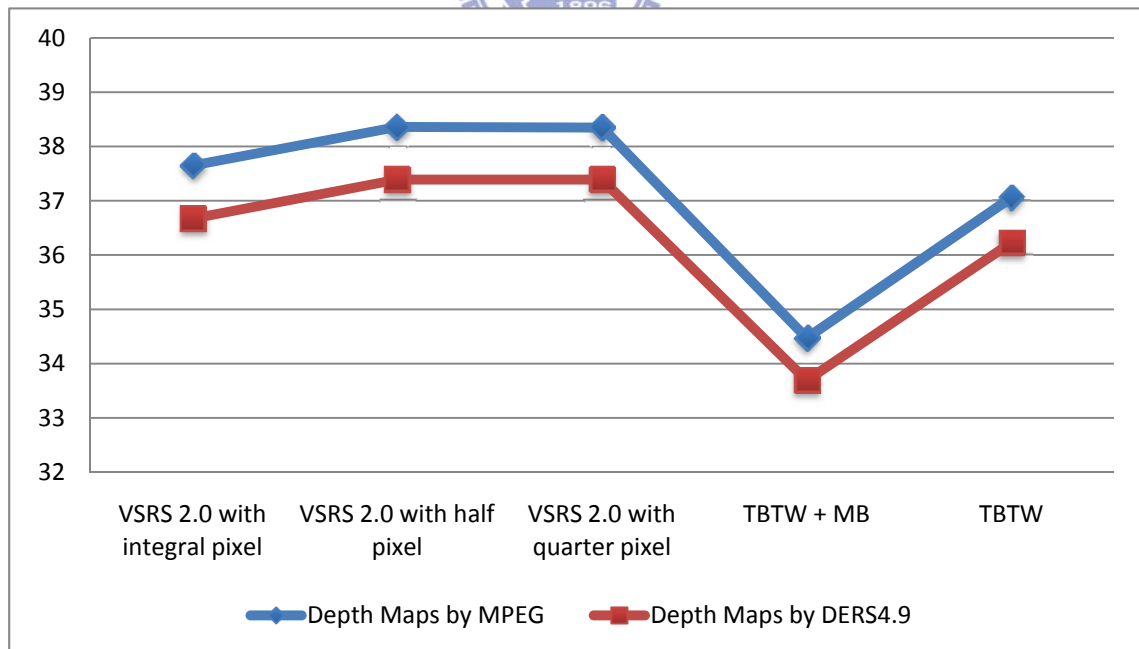


圖 4-31 DERS 4.9 產生的景深圖對於視點視點合成演算法針對測試影像 “Kendo” 視點 04 的前 30 張影像上的平均 PSNR 表現



圖 4-32 測試影像 “Champagne tower” 利用 DERS 4.9 產生的景深圖由視點 39 與視點 41 所合成的虛擬視點 40 結果比較圖

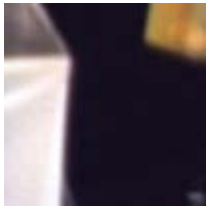
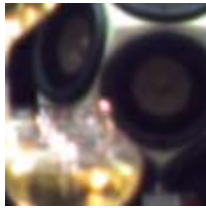
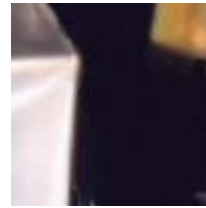
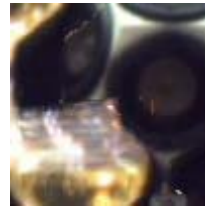
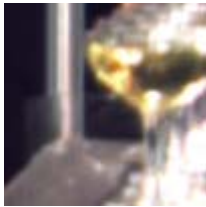
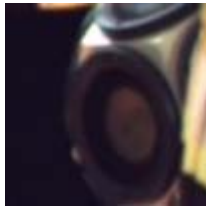


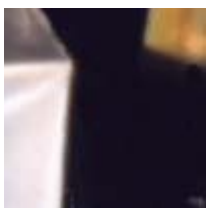

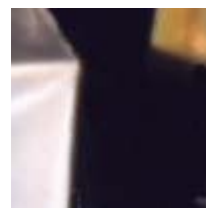

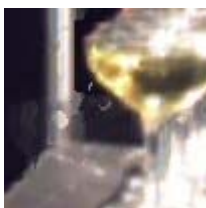
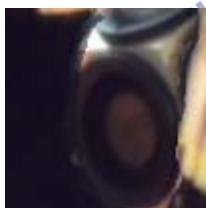

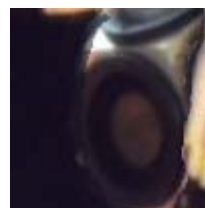

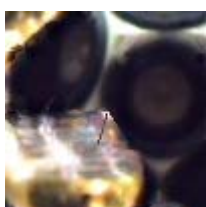
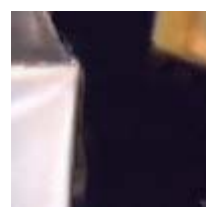
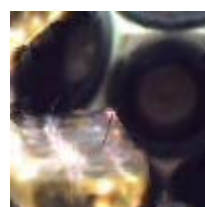
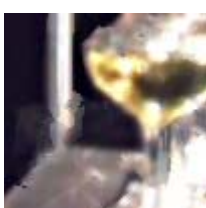
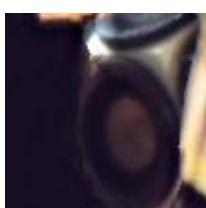
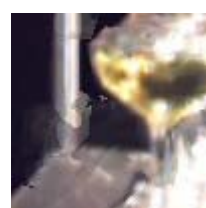
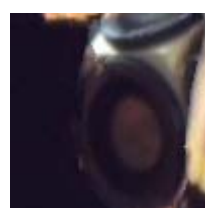
Test sequences: Champagne tower (Nagoya Univ.)			
			
			
Ground truth		DERS4.9+VSRS 2.0 (Integral pixel)	
			
			
DERS4.9+VSRS 2.0 (Half pixel)		DERS4.9+VSRS 2.0 (Quarter pixel)	
			
			
DERS4.9 + TBTW + MB		DERS4.9 + TBTW	

圖 4-33 測試影像 “Champagne tower” 利用 DERS 4.9 產生的景深圖由視點 39 與視點 41 所合成的虛擬視點 40 細部放大結果比較圖

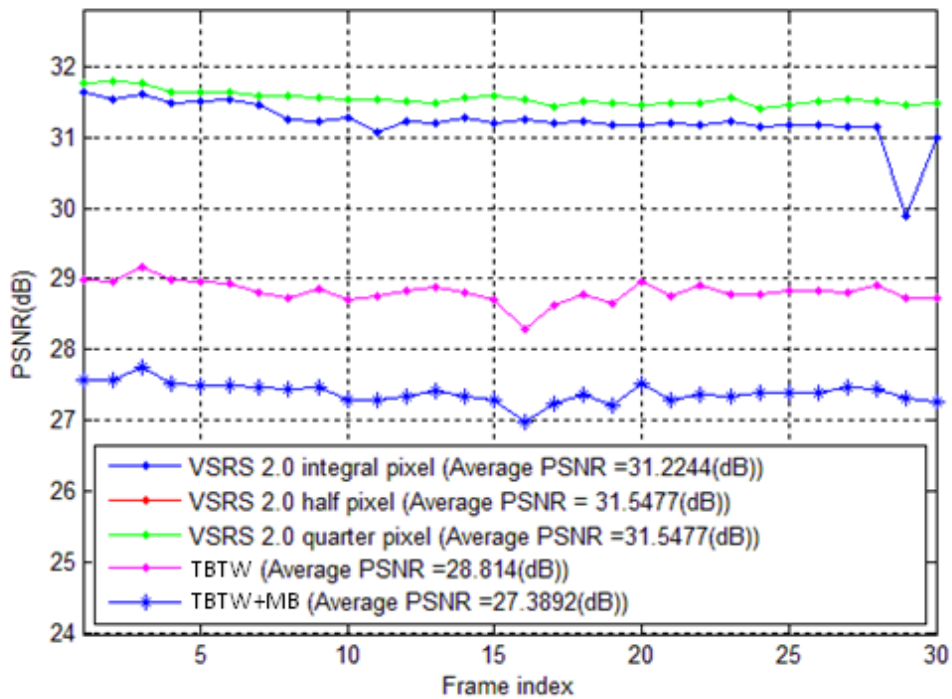


圖 4-34 DERS 4.9 產生的景深圖對於視點視點合成演算法在測試影像“Champagne tower” 視點 40 的前 30 張影像 PSNR 表現

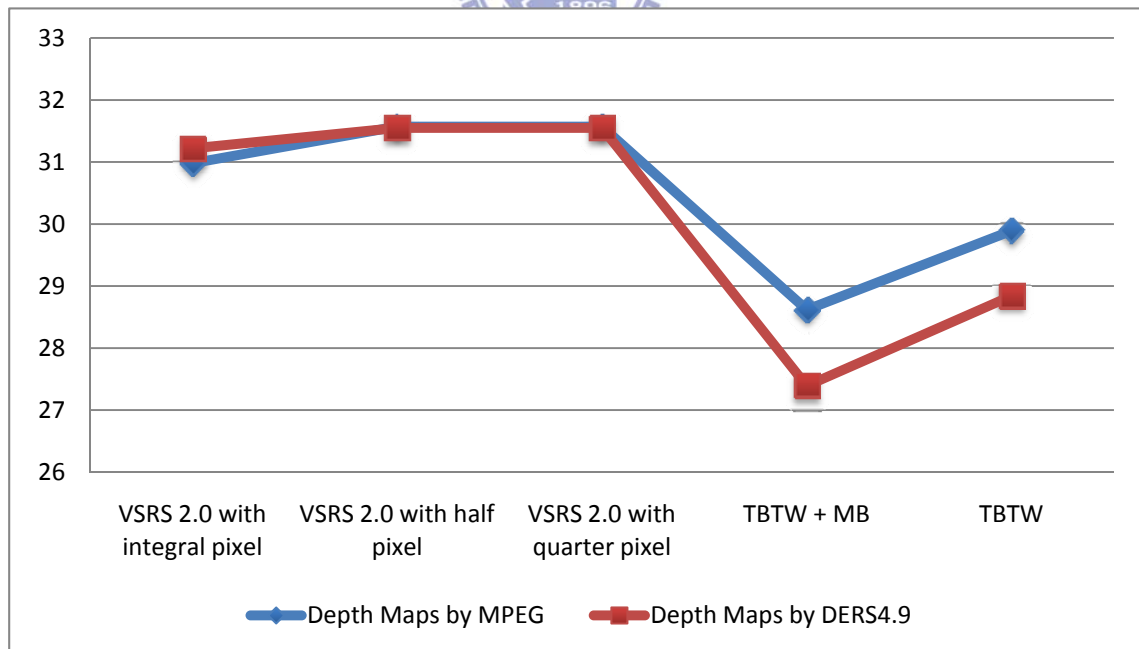


圖 4-35 DERS 4.9 產生的景深圖對於視點視點合成演算法針對測試影像“Champagne tower” “視點 40 的前 30 張影像上的平均 PSNR 表現

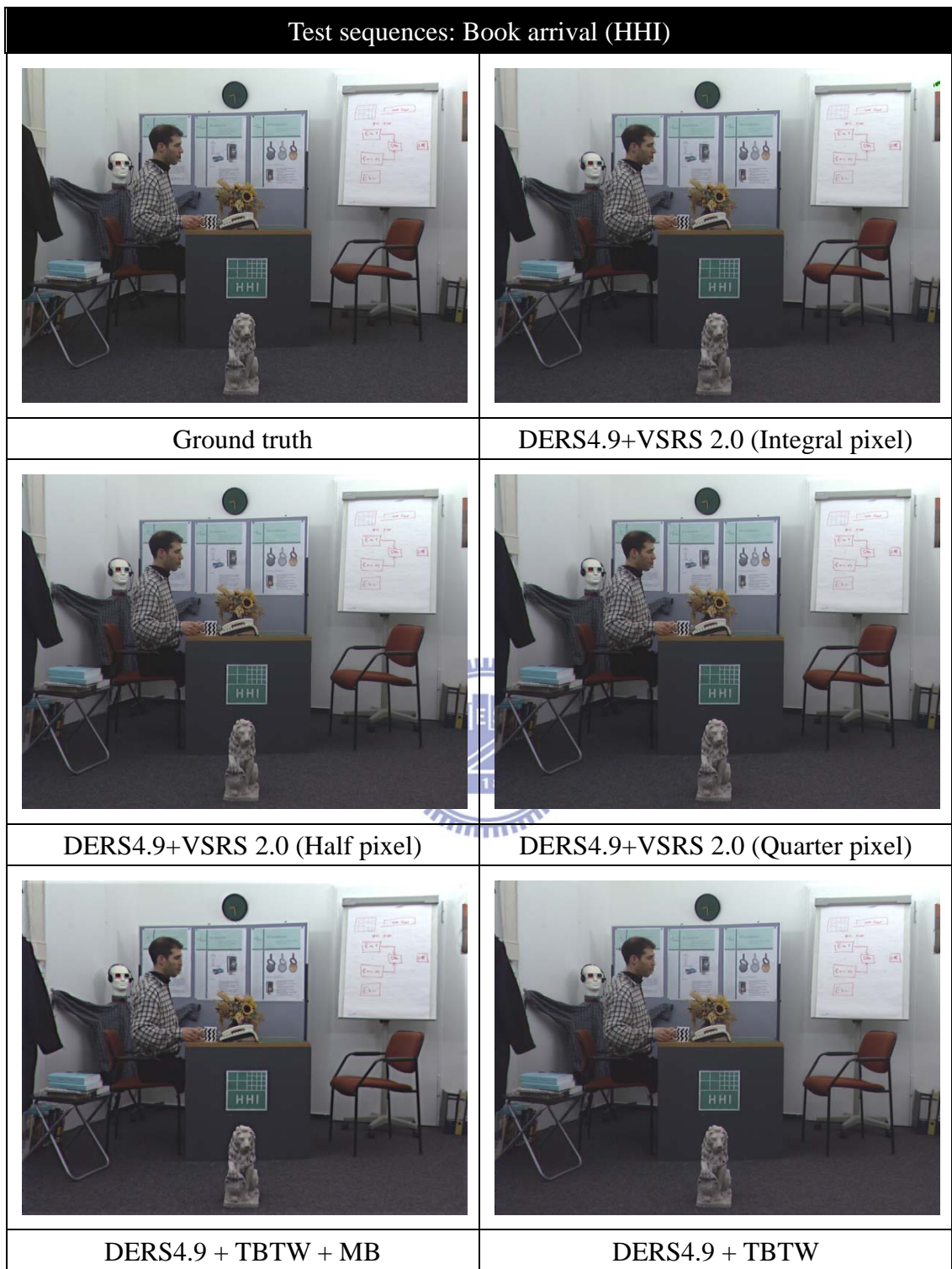


圖 4-36 測試影像 “Book arrival” 利用 DERS 4.9 產生的景深圖由視點 08 與視點 10 所合成的虛擬視點 09 結果比較圖

Test sequences: Champagne tower (Nagoya Univ.)			
Ground truth		DERS4.9+VSRS 2.0 (Integral pixel)	
DERS4.9+VSRS 2.0 (Half pixel)		DERS4.9+VSRS 2.0 (Quarter pixel)	
DERS4.9 + TBTW + MB		DERS4.9 + TBTW	

圖 4-37 測試影像 “Book arrival” 利用 DERS 4.9 產生的景深圖由視點 08 與視點 10 所合成的虛擬視點 09 細部放大結果比較圖

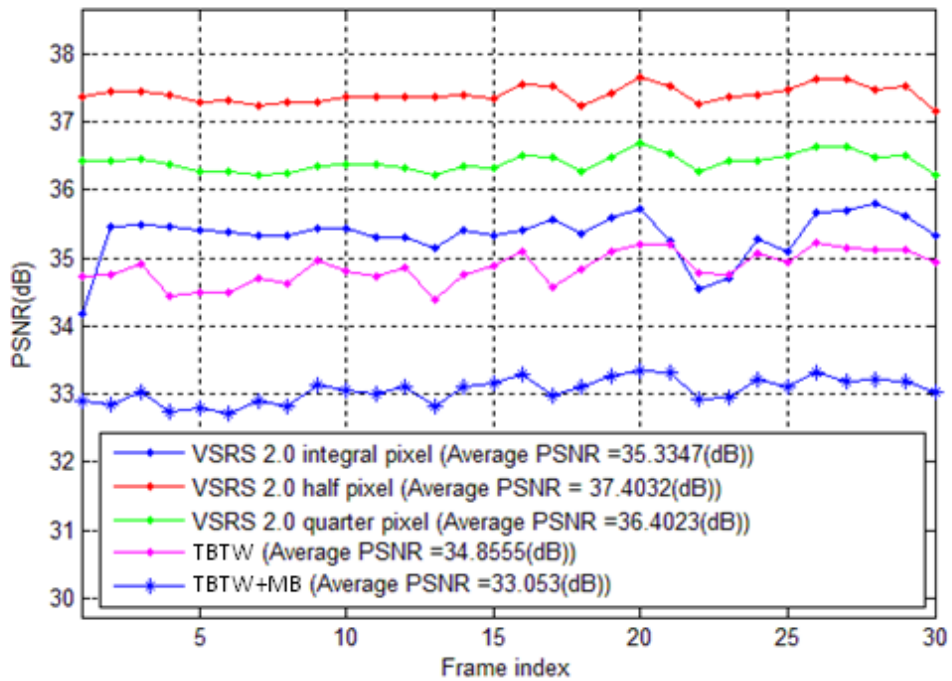


圖 4-38 DERS 4.9 產生的景深圖對於視點視點合成演算法在測試影像 “Book arrival” 視點 09 的前 30 張影像 PSNR 表現

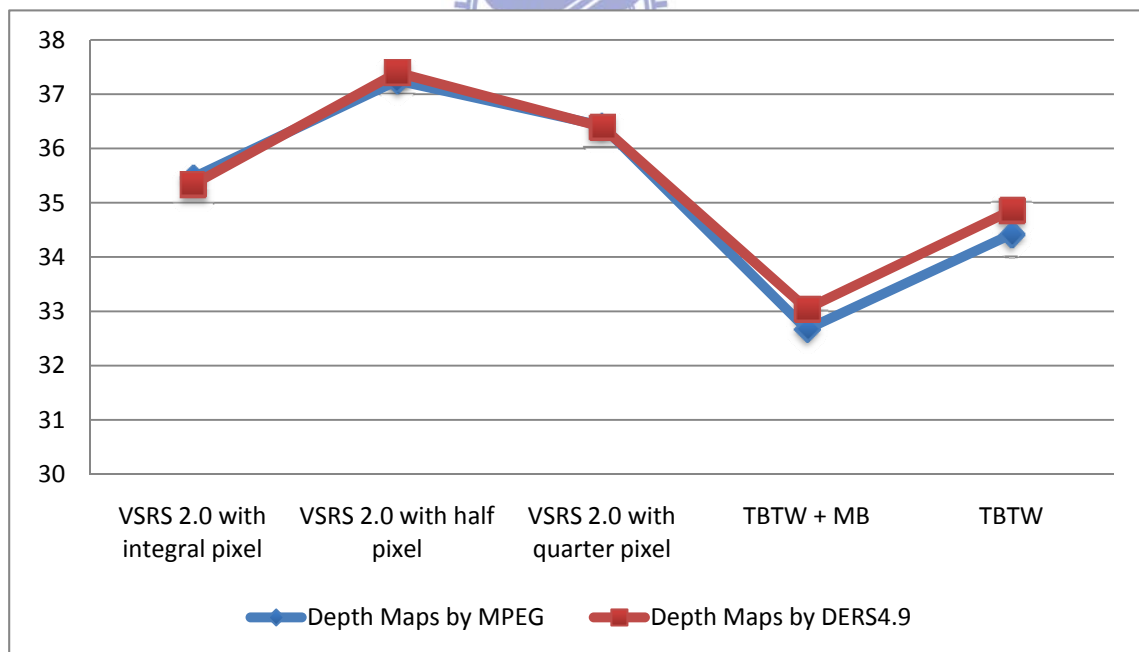




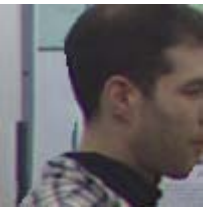




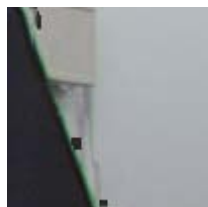


圖 4-39 DERS 4.9 產生的景深圖對於視點視點合成演算法針對測試影像 “Book arrival” 視點 09 的前 30 張影像上的平均 PSNR 表現

實驗結果與分析

在四個測試影像中，我們 MPEG 測試影像 “Newspaper”。參考軟體 VSRS 2.0 以整數為單位畫素的合成影像結果，可以明顯發現在人的右手邊緣部分容易產生一些不連續現象，如圖 4-24 與圖 4-25 所示。以主觀的肉眼觀看感受而言，VSRS 2.0 的結果與我們所提出的演算法差異性不大。但以平均 PSNR 的數值而言，我們的演算法相較 VSRS 2.0 大略高於 0.07dB。以測試影像 “Kendo” 而言，整體的 PSNR 結果與主觀的視覺感受是差不多的。

表 9 MPEG 測試影像 “Book arrival” 對於不同的景深圖合成結果比較表

	DERS 4.9		MPEG Sequences	
				
TBTW				
VSRS 2.0				

對測試影像 “Book arrival” 而言，圖 4-36 表示為我們所提出的演算法以及參考軟體 VSRS 2.0 以整數單位畫素的合成結果。以整體而言，MPEG 測試影像所提供的景深圖產生出來的合成結果相較於 DERS 4.9 產生的景深圖所產生合成結

果來得較佳，但主觀視覺而言，採用 DERS 4.9 的景深圖對於 VSRS 2.0 的合成結果表現較好。

對於測試影像 “Champagne tower” 而言，圖 4-32 表示為我們所提出的演算法以及參考軟體 VSRS 2.0 以整數單位畫素的合成結果。以細部放大的合成影像來看，在酒杯的邊緣附近容易產生一些瑕疵現象，如圖 4-33 所示。以整體而言，DERS 4.9 產生的景深圖品質相較於 MPEG 測試影像所提供的景深圖來看，景深圖的品質的確對於合成影像的 PSNR 表現以及視覺感受有些差異，MPEG 測試影像所提供的景深圖整體 PSNR 的表現較佳。

表 10 表示我們分別針對 MPEG 所提供四種不同的 FTV 測試影像：“Book arrival”、“Champagne tower”、“Newspaper”以及“Kendo”的前 30 張影像進行視點合成演算法的平均 PSNR 的結果。利用 DERS 4.9 產生的景深圖[23]進行視點合成演算法，以輸入的景深圖品質確實對於合成端產生的影像結果以及平均 PSNR 數值來看，MPEG 提供的景深圖對於合成影像的品質較佳。

表 10 MPEG 測試影像針對前 30 張合成影像的平均 PSNR 效能表現

Test sequences	Input Depth maps	Newspaper	Champagne tower	Kendo	Book Arrival
Our proposed VS method					
TBTW + MB	MPEG	30.36	28.62	34.47	32.67
	DERS 4.9	29.95	27.39	33.69	33.05
TBTW	MPEG	32.06	29.89	37.06	34.42
	DERS 4.9	31.63	28.81	36.23	34.86
VSRS 2.0					
VSRS 2.0 with integral pixel	MPEG	32.26	30.98	37.65	35.46
	DERS 4.9	31.57	31.22	36.67	35.33
VSRS 2.0 with half pixel	MPEG	32.82	31.57	38.36	37.25
	DERS 4.9	31.85	31.55	37.39	37.40
VSRS 2.0 with quarter pixel	MPEG	32.82	31.57	38.35	36.42
	DERS 4.9	31.85	31.55	37.39	36.40

4.3.2.3 實驗第三部份：攝影機位置的差異性對於合成結果的表現

第三部份，我們利用 MPEG 所提供的測試軟體 DERS 4.9 來產生測試影像的對應景深圖，測試演算法對於攝影機擺放間隔較大的情況下，輸入不同位置的攝影機所產生的景深圖與原來位置的合成結果觀察其演算法合成結果的優劣性，實驗介紹如表 11 所示。

我們利用前面章節所提出的視點合成演算法以及 VSRS 2.0，分別針對四種 MPEG 測試影像的前 30 張圖片測量其 PSNR 的數值，分析其實驗結果。

表 11 DERS 4.9 產生的景深圖以及新對應的視點位置





Data set	Champagne tower	Kendo	Book arrival	Newspaper
Corresponding Depth map				
Experiment part II. (Depth maps by DERS 4.9 + VS methods)				
Original Viewpoint Pair (OL-OR)	39-41	03-05	08-10	04-06
Experiment part III. (Depth maps by DERS 4.9 + VS methods + New Camera Location)				
Original Viewpoint Pair (OL-OR)	38-42	02-05	08-11	03-07
Synthesized viewpoint	40	04	09	05



圖 4-40 測試影像 “Newspaper” 利用 DERS 4.9 產生的景深圖由視點 04 與視點 06 所合成的虛擬視點 05 結果比較圖

Test sequences: Newspaper (GIST)			
Ground truth		DERS4.9+VSRS 2.0 (Integral pixel)	
DERS4.9+VSRS 2.0 (Half pixel)		DERS4.9+VSRS 2.0 (Quarter pixel)	
DERS4.9 + TBTW + MB		DERS4.9 + TBTW	

圖 4-41 測試影像 “Newspaper” 利用 DERS 4.9 產生的景深圖由視點 04 與視點 06 所合成的虛擬視點 05 細部結果放大比較圖

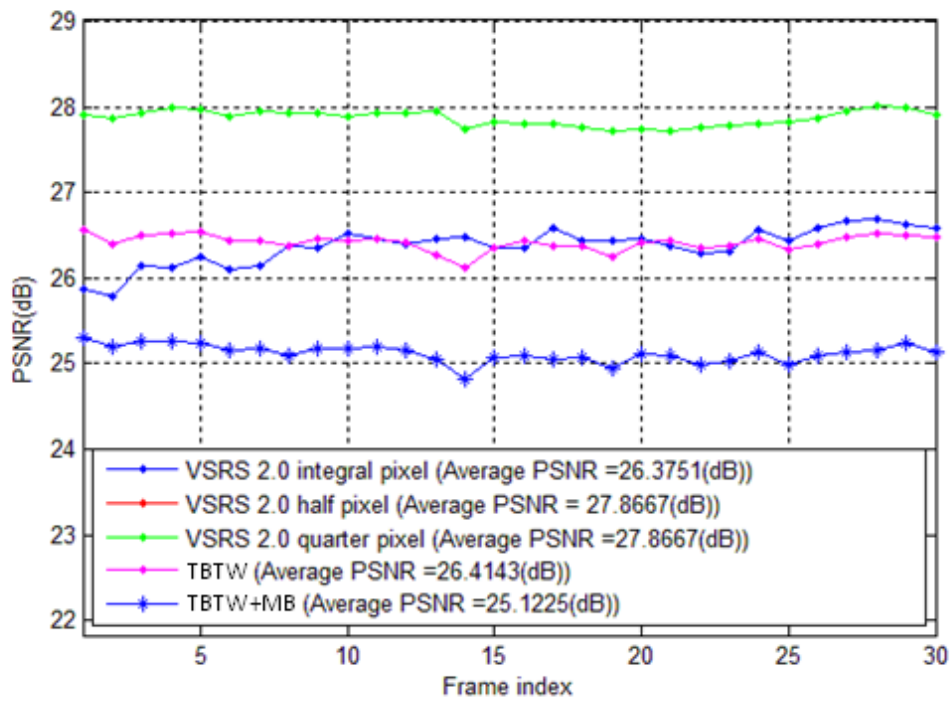


圖 4-42 DERS 4.9 產生視點 04 與視點 06 的景深圖對於視點合成演算法在測試影像 “Newspaper” 視點 05 的前 30 張影像 PSNR 表現

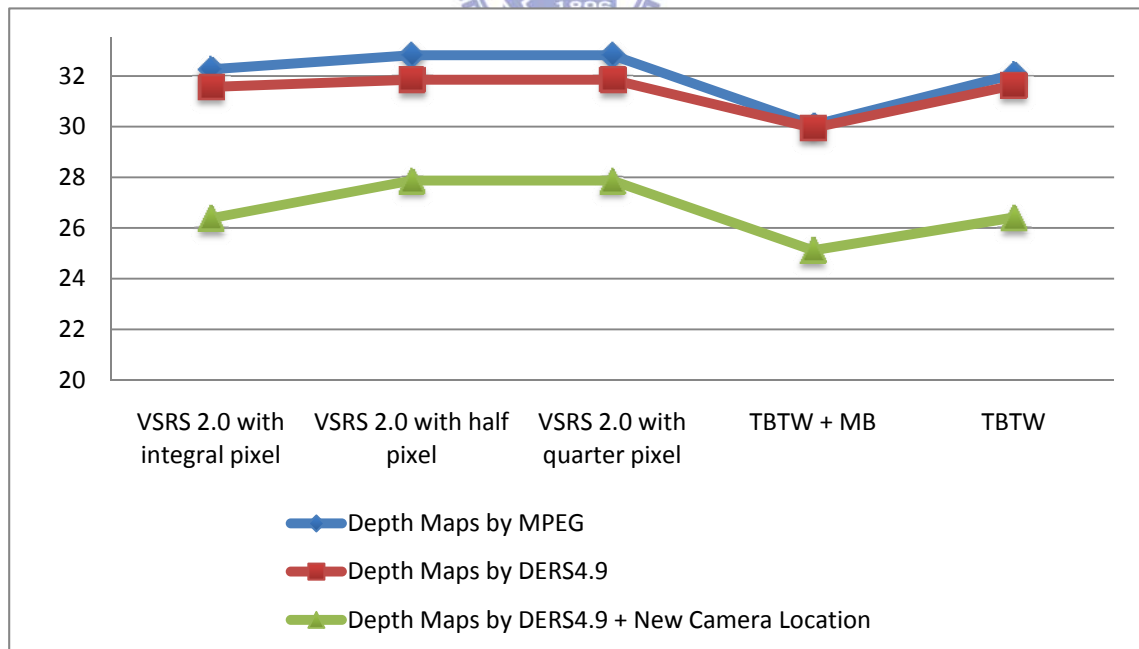


圖 4-43 DERS 4.9 產生視點 04 與視點 06 的景深圖對於視點視點合成演算法針對測試影像 “Newspaper” 視點 05 的前 30 張影像上的平均 PSNR 表現

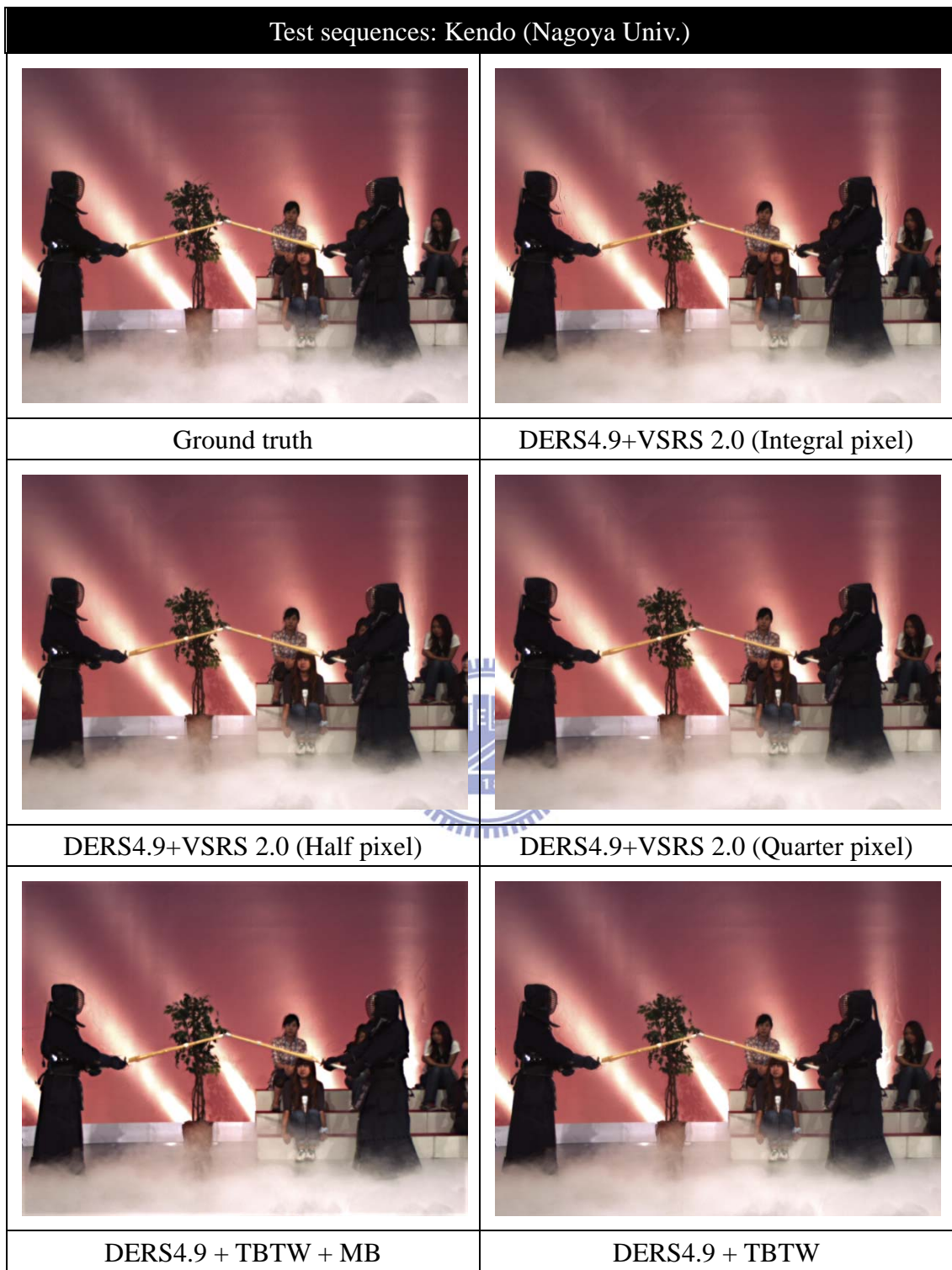


圖 4-44 測試影像 “Kendo” 利用 DERS 4.9 產生的景深圖由視點 02 與視點 05 所合成的虛擬視點 04 結果比較圖

Test sequences: Kendo (Nagoya Univ.)							
							
Ground truth				DERS4.9+VSRS 2.0 (Integral pixel)			
							
DERS4.9+VSRS 2.0 (Half pixel)				DERS4.9+VSRS 2.0 (Quarter pixel)			
							
DERS4.9 + TBTW + MB				DERS4.9 + TBTW			

圖 4-45 測試影像 “Kendo” 利用 DERS 4.9 產生的景深圖由視點 02 與視點 05 所合成的虛擬視點 04 細部放大結果比較圖

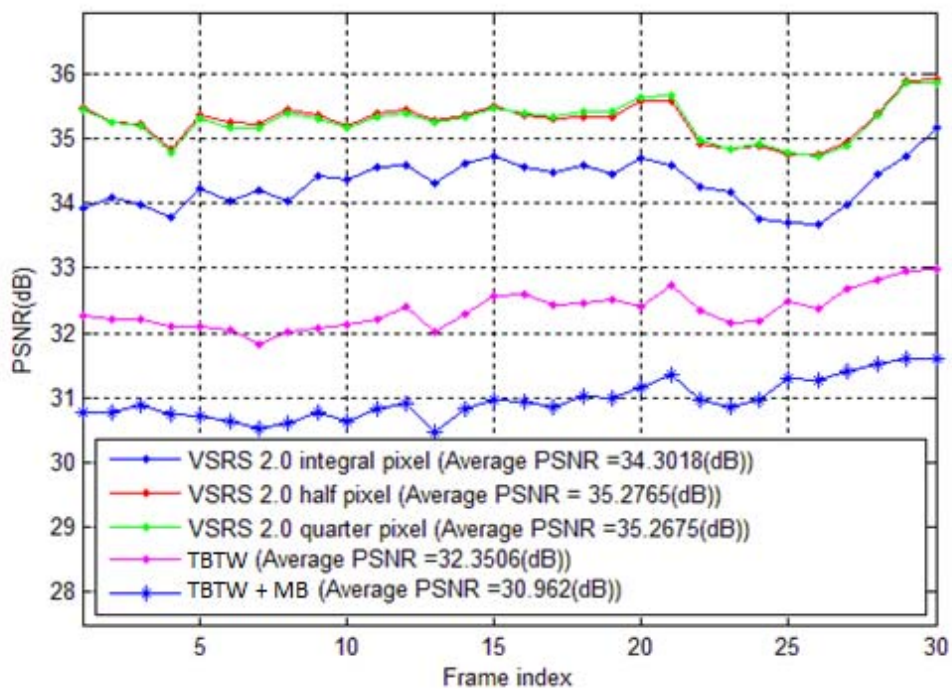


圖 4-46 DERS 4.9 產生視點 02 與視點 05 的景深圖對於視點視點合成演算法在測試影像 “Kendo” 視點 04 的前 30 張影像 PSNR 表現

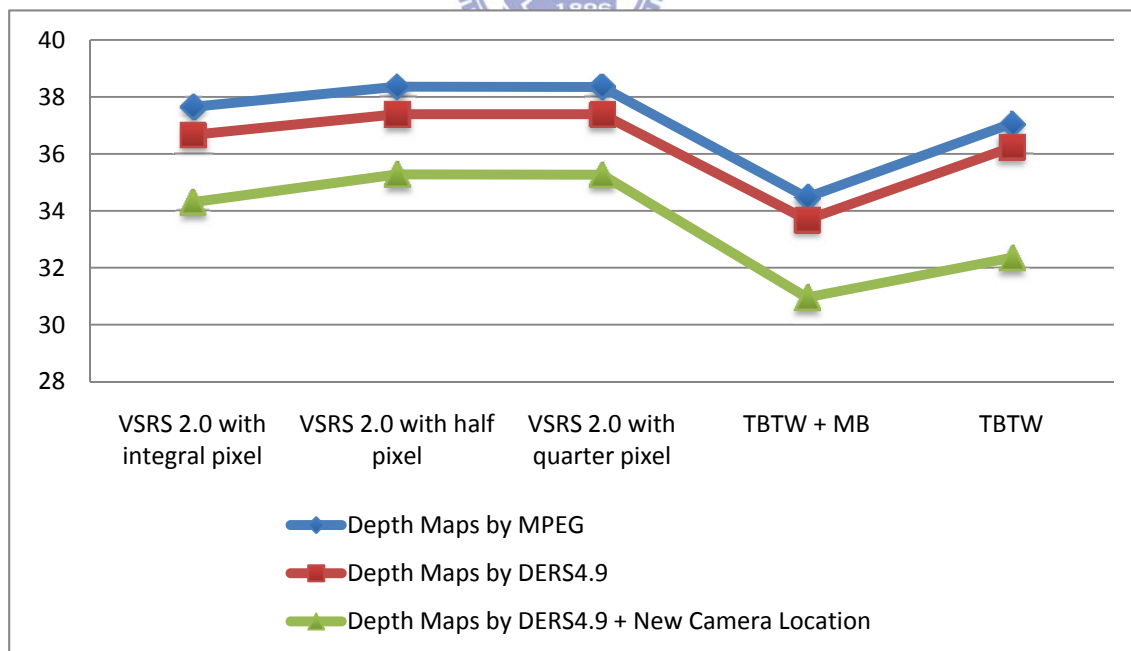


圖 4-47 DERS 4.9 產生視點 02 與視點 05 的景深圖對於視點視點合成演算法針對測試影像 “Kendo” 視點 04 的前 30 張影像上的平均 PSNR 表現



圖 4-48 測試影像 “Champagne tower” 利用 DERS 4.9 產生的景深圖由視點 38 與視點 42 所合成的虛擬視點 40 結果比較圖

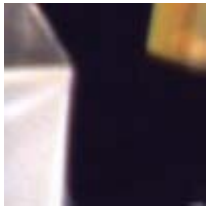
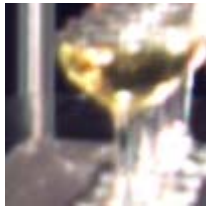
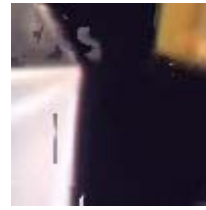


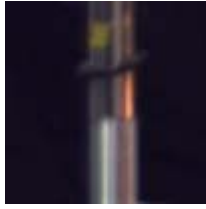

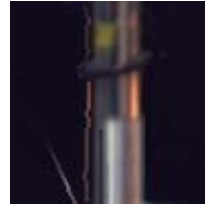
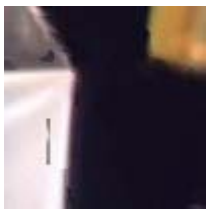

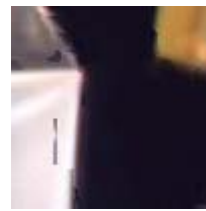

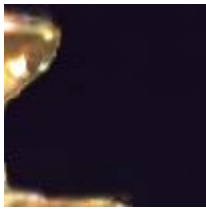
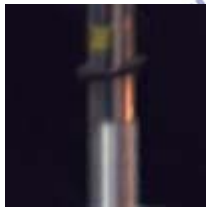
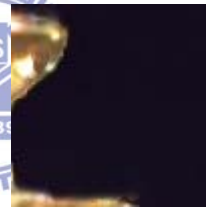
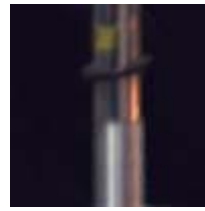
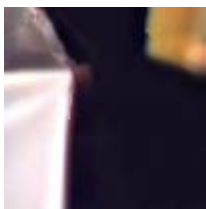
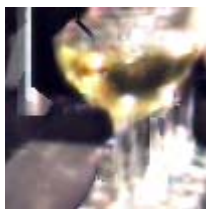
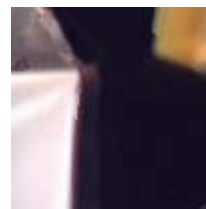
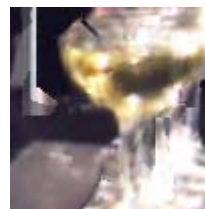

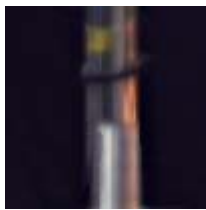
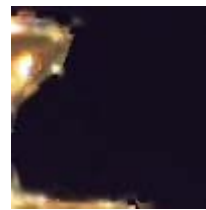

Test sequences: Champagne tower (Nagoya Univ.)			
			
			
Ground truth		DERS4.9+VSRS 2.0 (Integral pixel)	
			
			
DERS4.9+VSRS 2.0 (Half pixel)		DERS4.9+VSRS 2.0 (Quarter pixel)	
			
			
DERS4.9 + TBTW + MB		DERS4.9 + TBTW	

圖 4-49 測試影像 “Champagne tower” 利用 DERS 4.9 產生的景深圖由視點 38 與視點 42 所合成的虛擬視點 40 細部放大結果比較圖

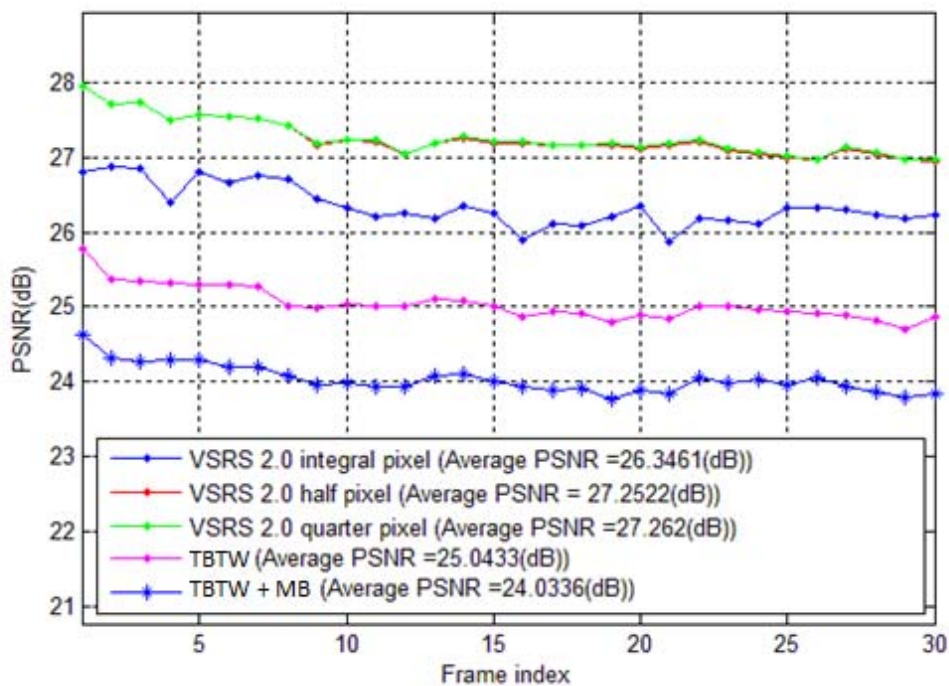


圖 4-50 DERS 4.9 產生視點 38 與視點 42 的景深圖對於視點視點合成演算法在測試影像 “Champagne tower” 視點 40 的前 30 張影像 PSNR 表現

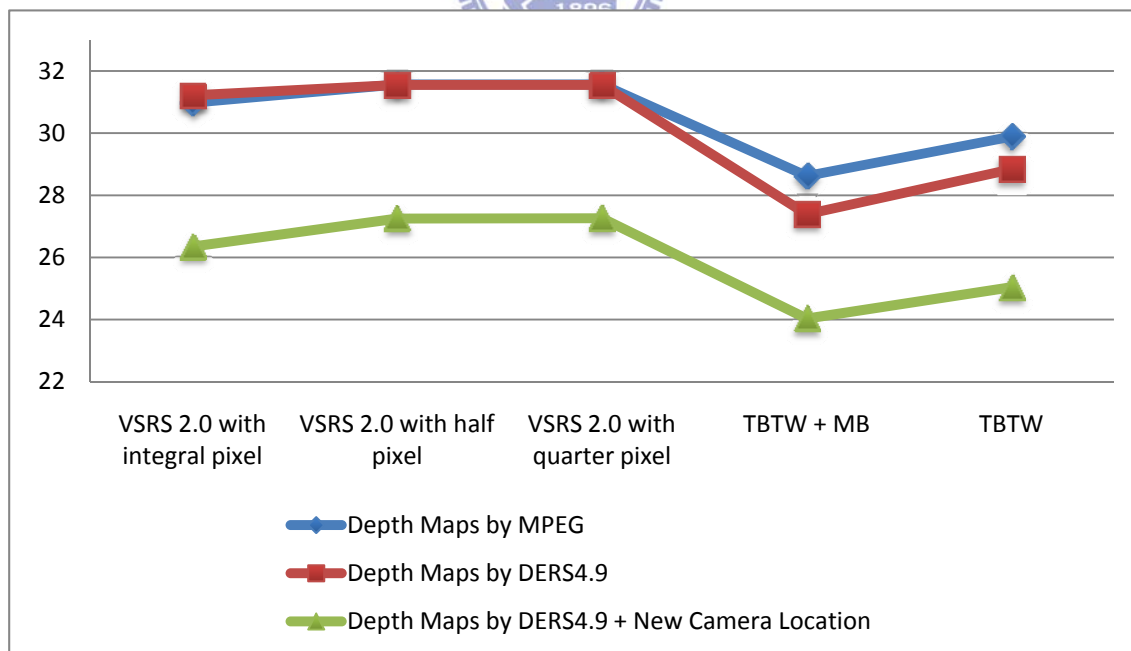


圖 4-51 DERS 4.9 產生視點 38 與視點 42 的景深圖對於視點視點合成演算法針對測試影像 “Champagne tower” 視點 40 的前 30 張影像上的平均 PSNR 表現

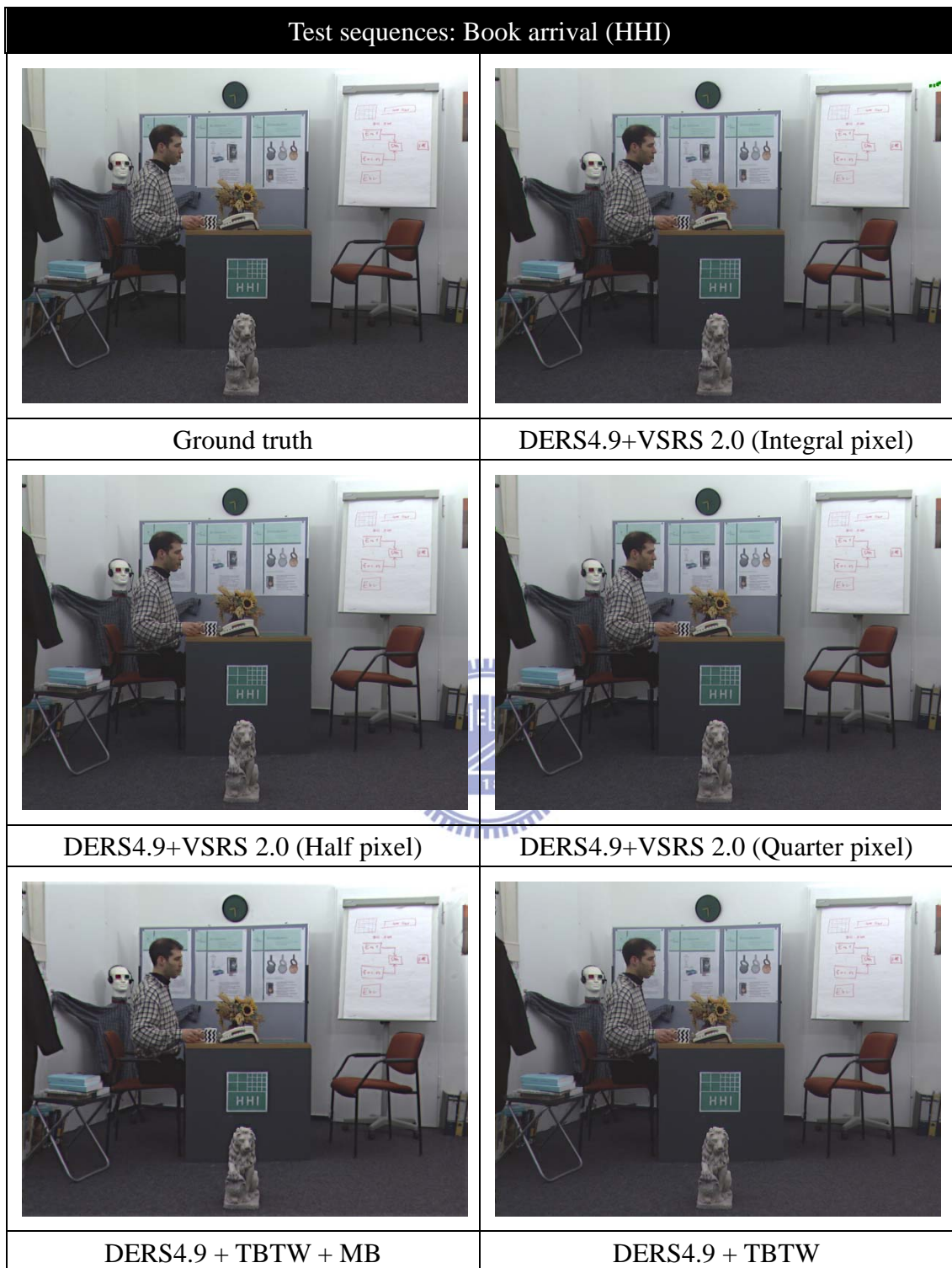


圖 4-52 測試影像 “Book arrival” 利用 DERS 4.9 產生的景深圖由視點 08 與視點 11 所合成的虛擬視點 09 結果比較圖




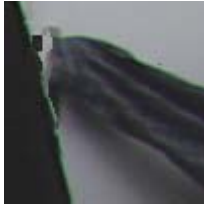


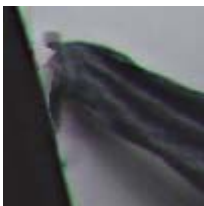
Test sequences: Book arrival (HHI)			
			
			
Ground truth		DERS4.9+VSRS 2.0 (Integral pixel)	
			
			
DERS4.9+VSRS 2.0 (Half pixel)		DERS4.9+VSRS 2.0 (Quarter pixel)	
			
			
DERS4.9 + TBTW + MB		DERS4.9 + TBTW	

圖 4-53 測試影像 “Book arrival” 利用 DERS 4.9 產生的景深圖由視點 08 與視點 11 所合成的虛擬視點 09 細部放大結果比較圖

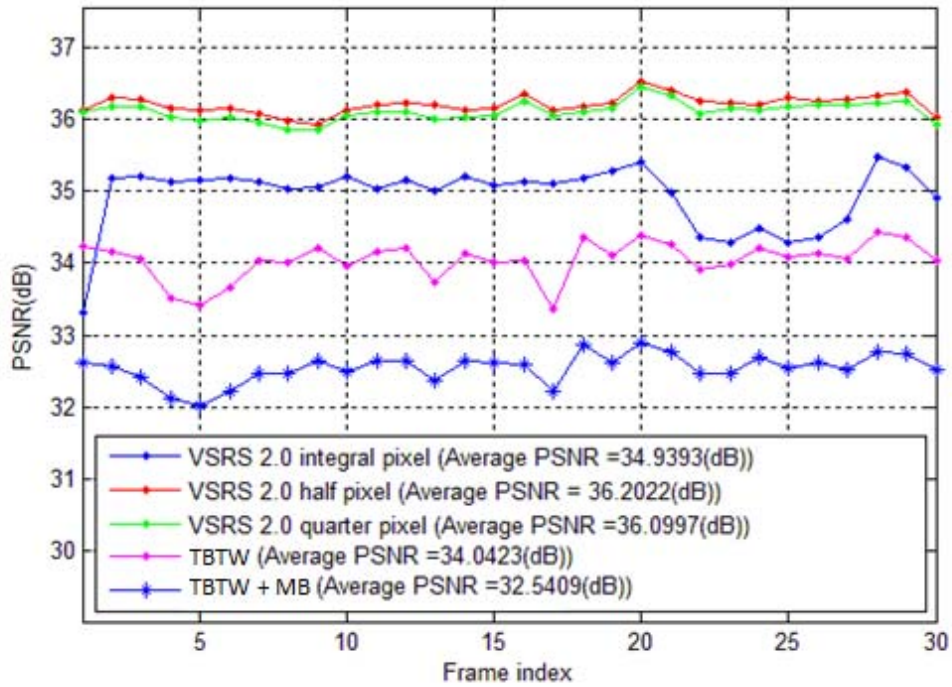


圖 4-54 DERS 4.9 產生視點 08 與視點 11 的景深圖對於視點視點合成演算法在測試影像 “Book arrival” 視點 09 的前 30 張影像 PSNR 表現

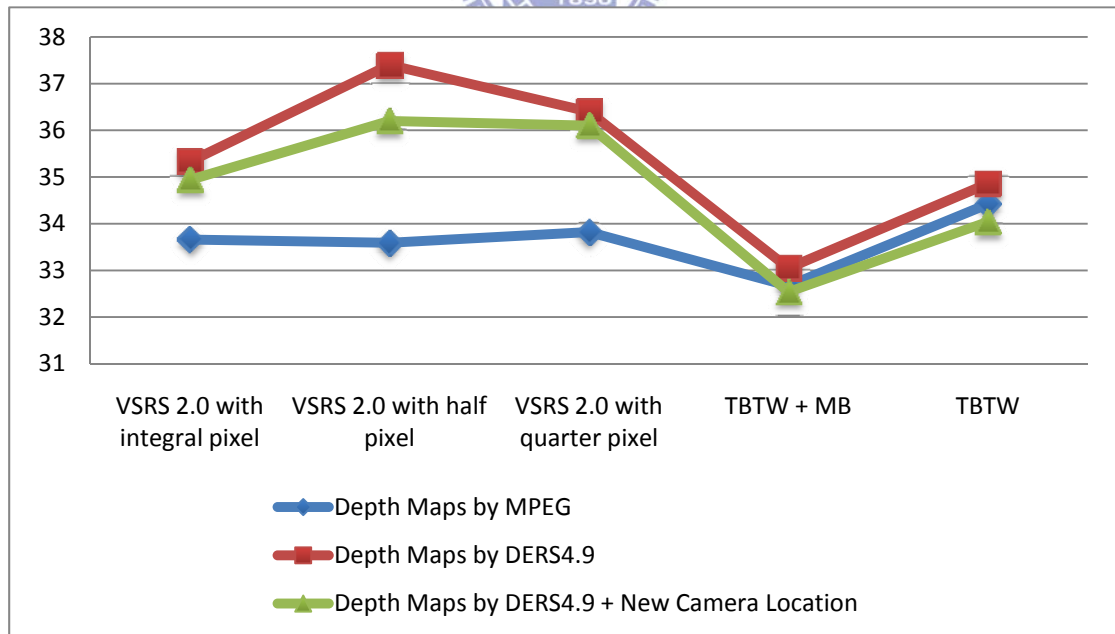


圖 4-55 DERS 4.9 產生視點 08 與視點 11 的景深圖對於視點視點合成演算法針對測試影像 “Book arrival” 視點 09 的前 30 張影像上的平均 PSNR 表現

實驗結果與分析

表 12 表示分別輸入 MPEG 所提供四種不同的 FTV 測試影像景深圖：“Book arrival”、“Champagne tower”、“Newspaper”以及“Kendo”的前 30 張影像進行視點合成演算法的平均 PSNR 的結果。

表 12 MPEG 測試影像提供的景深圖對於前 30 張合成影像的平均 PSNR 效能表現

Test sequences	Newspaper	Champagne tower	Kendo	Book Arrival
Viewpoint pair	04-06	39-41	03-05	08-10
Our proposed VS method				
TBTW + MB	30.36	28.62	34.47	32.67
TBTW	32.06	29.89	37.06	34.42
VSRS 2.0				
VSRS 2.0 with integral pixel	32.26	30.98	37.65	35.46
VSRS 2.0 with half pixel	32.82	31.57	38.36	37.25
VSRS 2.0 with quarter pixel	32.82	31.57	38.35	36.42

表 13 表示利用 DERS 4.9 產生的景深圖[23]進行視點合成演算法的平均 PSNR 結果，左欄表格表示原始位置進行視點合成演算法的平均 PSNR 表現及影像品質，右欄表格為利用較遠的攝影機位置進行視點合成演算法且觀察其平均 PSNR 表現。

對於 MPEG 測試影像“Newspaper”，我們所提出的演算法與參考軟體 VSRS 2.0 以整數為單位畫素的合成影像品質均較前兩個實驗結果差，如圖 4-40 與圖 4-41 所示。以平均 PSNR 的數值而言，我們的演算法相較 VSRS 2.0 大略差不多。以 MPEG 測試影像“Kendo”而言，畫面中仍有許多瑕疵現象仍待改善，但整體的平均 PSNR 表現相較於實驗的第二部分大約下降 0.5 到 1dB 不等。

對於 MPEG 測試影像 “Champagne tower”，圖 4-49 表示為我們所提出的演算法架構以及參考軟體 VSRS 2.0 以整數單位畫素的合成結果，攝影機的擺放位置與目標位置距離較遠時，對於所有視點合成演算法得最後合成影像的平均數值以及主觀視覺表現皆不佳。

對於 MPEG 測試影像 “Book arrival”，實驗的第一部分我們所提出的 TBTW 演算法架構略高於 VSRS 2.0 約 0.8dB，將 DERS 4.9 產生的景深圖當作輸入影像，則普遍 VSRS 2.0 的 PSNR 表現略高於 0.5 到 0.9dB 不等。但以主觀視覺而言，我們所提出的演算法架構在實驗的第二部份表現較 VSRS 2.0 來的優異，其細部結果放大圖可參考圖 4-53。

表 13 DERS 4.9 產生的景深圖對於前 30 張合成影像的平均 PSNR 效能表現

Test sequences	Newspaper		Champagne tower		Kendo		Book Arrival	
Viewpoint pair	04-06	03-07	39-41	38-42	03-05	02-05	08-10	08-11
Our proposed VS method								
TBTW + MB	29.95	25.12	27.39	24.03	33.69	30.96	33.05	32.54
TBTW	31.63	26.41	28.81	25.04	36.23	32.35	34.86	34.04
VSRS 2.0								
VSRS 2.0 with integral pixel	31.57	26.38	31.22	26.35	36.67	34.30	35.33	34.94
VSRS 2.0 with half pixel	31.85	27.87	31.55	27.25	37.39	35.28	37.40	36.20
VSRS 2.0 with quarter pixel	31.85	27.87	31.55	27.25	37.39	35.27	36.40	36.10

對於 VSRS 2.0 而言，以二分之一或是四分之一畫素單位合成影像，可以發現對於攝影機距離較遠所合成出來的平均 PSNR 約下降 0.4dB 到 5dB 左右。攝影機距離越遠，則平均 PSNR 值下降越多，相較於以整數畫素為單位的合成影像而言，PSNR 值約略高於 0.8dB 到 1.5dB。以實驗第二部份而言，以子畫素為單位合成影像品質的平均 PSNR 值約成長 0.3dB 到 0.8dB 左右，相較於實驗的第三部

份，以子畫素為單位合成影像品質的平均 PSNR 值較為穩定。

以實驗整體的第三部分而言，在合成虛擬視點影像的過程中，若是攝影機的擺放位置並非密集的擺放，則對於合成端所產生的影像結果以及平均 PSNR 數值的確有待加強。如果我們再針對三角形對三角型投影映射後的合成影像結果做一些後續的濾波器處理，也許會得到更佳的效果。

結論

針對物體邊緣處容易產生一些不連續性的瑕疵現象，對於 VSRS 2.0 而言使用以子畫素為單位相較以整數畫素為單位的合成影像而言，擁有較佳的主觀視覺品質。與 TBTW 演算法相比，由於三角形對三角形映射的過程中，將三角形區域內的紋理直接映射，並未做任何修改。因此將左右兩張映射後的參考影像做線性疊合時，紋理複雜的部分容易造成影像模糊的問題。

以整體而言，景深圖的品質的確對於合成影像的平均 PSNR 表現有些差異，DERS 4.9 產生的景深圖品質相較於 MPEG 測試影像所提供的景深圖來看，MPEG 測試影像的景深圖對於整體 PSNR 的表現較佳。

在合成虛擬視點影像的過程中，若攝影機的擺放位置並非密集的擺放，則對於合成端所產生的影像結果以及平均 PSNR 數值的確有待加強。攝影機距離越遠，則平均 PSNR 值下降越多，相較於以整數畫素為單位的合成影像而言，平均 PSNR 值表現較佳。以實驗的第二部份相較於第三部份的平均 PSNR 結果，以子畫素為單位合成影像品質的平均 PSNR 值較為穩定許多。與 TBTW 演算法相比，或許我們可以針對三角形對三角型投影映射後的合成影像結果做一些後續的濾波器處理，也許會得到更佳的效果。

第5章 基於三角形映射的景深圖資訊的任意視

點合成演算法設計

5.1 概述 (Overview)

傳統的任意視點合成(FTV)的演算法通常使用的投影映射方式都是針對畫素對畫素提出改善的方法[8][9][16][17]，雖然現今的實驗結果已有明顯的改善效果。然而，對於場景還原的技術而言，自由視點成像在進行成像的過程時，大多數的視點合成演算法通常紋理映射的部分大多數都是採用畫素對畫素的方式進行投影映射(pixel-by-pixel warping)。在本篇論文中前一章中我們提出的紋理映射方式是將三角形視為一個區域，並且利用三角形對三角形的紋理投影映射方式(Region-by-region warping)投影至目標影像。簡單來說，如果可以精確重建三角化模型的話，三角形映射方式確實會得到一個較佳的合成影像品質[26][27][28]。

本章中我們探討的課題是針對第四章的任意視點影像合成的部份，將兩種不同的紋理映射方式結合，藉以改善前面章節提出的演算法的效能，實驗結果也證明我們的方法可有效改善第四章的所提出的演算法效能。

5.2 傳統的任意視點合成演算法介紹 (Introduction to Conventional View Synthesis Algorithm)

5.2.1 利用畫素對畫素投影映射的視點合成演算法 (View Synthesis Algorithm by Using Pixel-by-pixel Warping)

傳統的任意視點合成演算法架構，在進行成像的過程時，大多數的紋理映射的部分都是採用畫素對畫素的方式進行投影映射[8][9][16][17]，其演算法的架構

大致上如下圖 5-1 所示。

一般而言，視點合成演算法分成五個部份：T1) 將景深圖利用畫素對畫素的方式進行投影映射至目標視點的景深圖位置、T2) 針對投影映射後的目標景深圖進行濾波器後續處理、T3) 利用目標景深圖的景深資訊找到參考影像的每個畫素對應位置、T4) 將左右兩張映射後的參考影像線性疊合得到最後輸出影像、T5) 針對無法預測的區域用鄰近的畫素資訊填補並輸出最後影像結果。

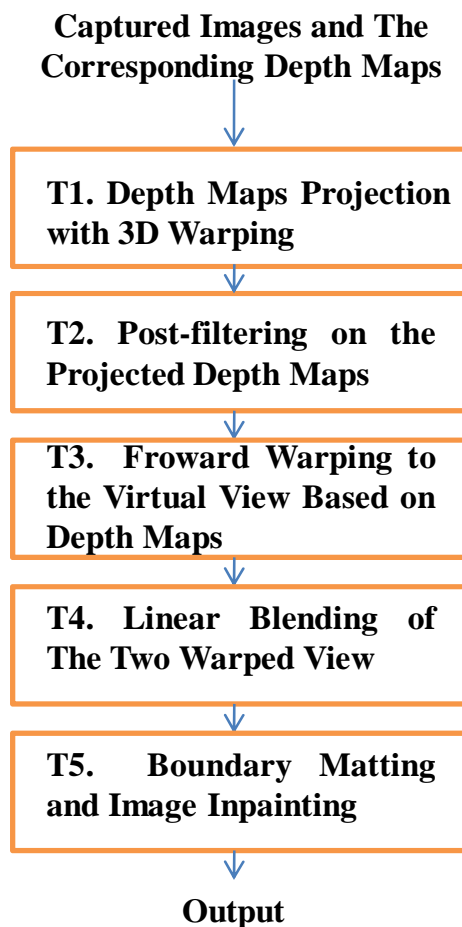


圖 5-1 畫素對畫素投影映射的視點合成演算法架構設計

5.2.2 利用三角形對三角形投影映射的視點合成演算法 (View Synthesis Algorithm by Using Triangle-by-triangle Warping)

事實上，現今也有許多演算法架構在景深估計，也有使用三角形對三角形的

方式找出對應的視差值[26][27][28][29]。B. Choi et al.將所有切割後的三角形利用分群法則的概念，區分出視差值屬於前景以及背景兩大區塊，並且顯影成像 [28]。在前一章節中，我們所提出的演算法是對於紋理映射的部分是採用三角形對三角形的方式進行投影映射(Triangular-based Texture Warping)，簡稱 TBTW 演算法，其演算法的架構大致上如下圖 5-2 所示。

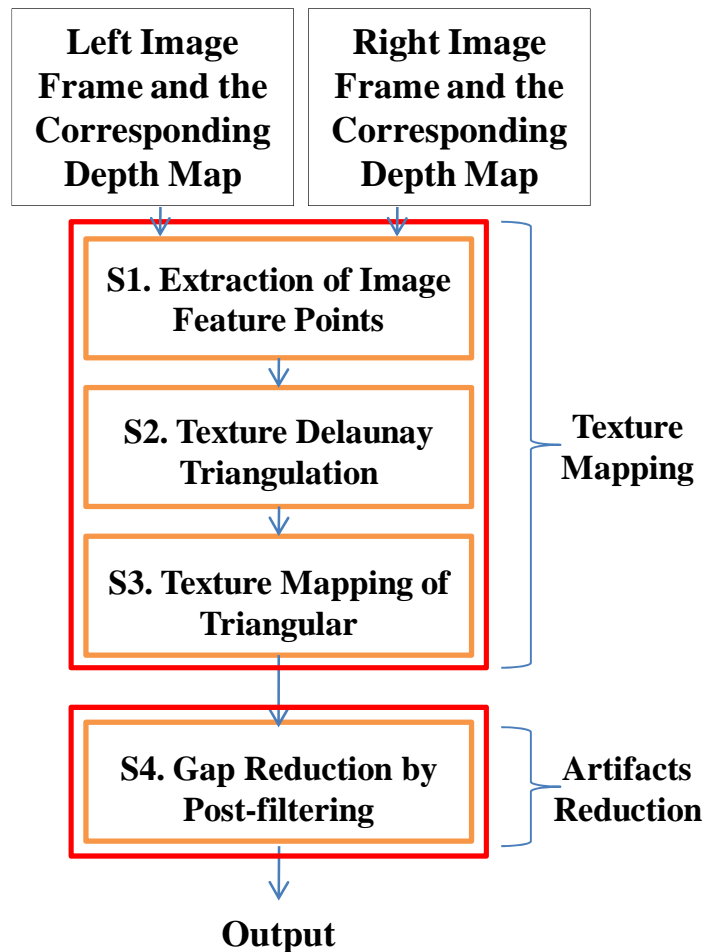


圖 5-2 三角形對三角形投影映射的視點合成演算法設計

一般而言，三角形的視點合成演算法分成四個部份：S1) 利用景深圖以及參考影像的梯度資訊擷取適當的特徵點位置、S2) 將所有特徵點集合建立一個以三角形形成的二維模型、S3) 將參考影像利用三角形對三角形的方式進行投影映射至目標視點的景深圖位置、S4) 將左右兩張映射後的參考影像經過濾波器處理將

裂痕修復破洞並線性疊合後，輸出最後影像結果。

整體而言，由於三角形對三角形投影映射的結果是將整塊三角形區域投影映射至目標位置，未做任何後續濾波器的處理。若是三角形區域內紋理複雜，則會易使得將左右映射後的參考影像線性疊合時，極具紋理的部分則十分容易產生影像模糊不清的問題，進而造成合成影像的品質低落。

5.3 基於三角映射的景深圖資訊的視點合成演算法 (View Synthesis Algorithm Based on Triangular Warped Depth Maps)

一般而言，傳統的畫素對畫素視點合成演算法中，由於正向投影映射的過程中，耗費大量計算資源的程序，隨著自由視點電視的興起，快速的視點合成演算法極待被開發。另一方面，由於三角形對三角形投影映射的結果，是將整塊三角形區域投影映射至目標位置，並未做任何後續濾波器的處理。若是三角形區域內紋理內容豐富的話，則會使得左右兩張經由正向投影映射後的參考影像進行線性疊合時，紋理複雜的部分十分容易造成影像模糊的問題，而使得合成影像的品質下降。

針對以上的缺點我們提出了新的演算法，只針對景深圖資訊使用三角形映射方式，對於紋理映射我們採用畫素對畫素的映射方法(Triangular-based Depth Maps Warping Algorithm)，簡稱 TBDMW 演算法。我們結合三角形對三角形投影映射以及畫素對畫素投影映射的視點演算法，其示意圖如圖 5-3 所示。在本章節的最後，我們根據實驗結果使得新提出的演算法能夠改善顯示成像的品質。

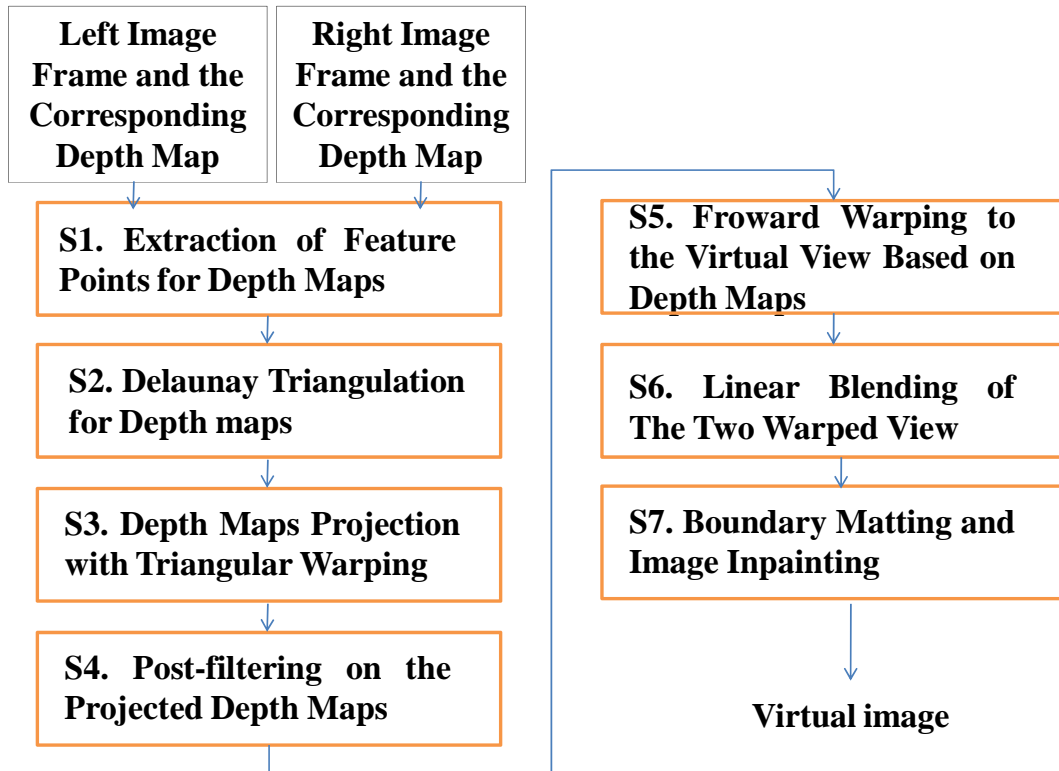


圖 5-3 基於三角映射的景深圖資訊的視點合成演算法介紹



5.3.1 針對景深圖擷取特徵點 (Extraction of Feature Points for Depth Maps)

與 4.2.1 不同，我們計算景深圖每個畫素的梯度值關係，對於每個畫素的位置而言，當訊號位置的梯度值越高，即代表越有可能為特徵點。因此必須先計算景深圖的梯度值， ∇ 代表梯度運算子，其中 D 代表景深圖資訊，如式(5.1)所示：

$$E(D) = |\nabla D| \quad (5.1)$$

當畫素的 $E(D, f)$ 大於週遭鄰近的畫素以及大於設定的臨界值，則我們即將此畫素的位置判定為特徵點。(在此演算法實現的過程中，我們將臨界值設定為 200)

5.3.2 對於景深圖的特徵點三角化 (Delaunay Triangulation for Depth Maps)

利用 Delaunay 三角化的方式將所有特徵點的集合重建二維影像的網點模型，

Delaunay 三角化的特性是給定一組任意的特徵點集合，即可決定一組唯一的三角形平面，使得我們能利用一組特徵點集合而不需要額外連結資訊即可精確重建三角化平面模型，如圖 5-4 所示。此外，對於三角形個數的影響將在 5.4.1.2 節討論

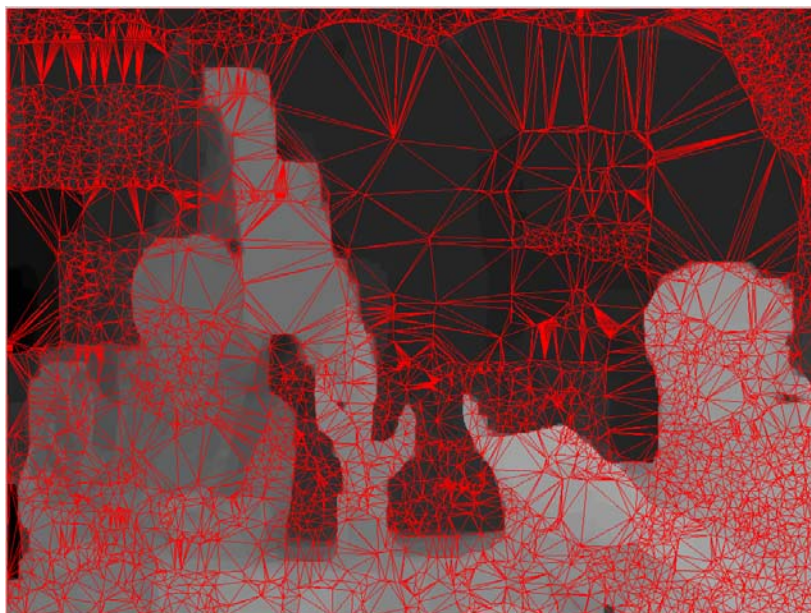


圖 5-4 利用 Delaunay 三角化重建三角平面模型：測試影像 “Newspaper” (特徵點數目：5656)

5.3.3 景深圖經由三角形投影映射至目標影像位置 (Depth Maps Projection with Triangular Warping)

如同 4.2.3 所提，在紋理投影映射的過程中，針對每塊三角形區域中的所有視差值進行統計運算，並挑選出現機率最高的視差值作為此三角形區域的視差值。接著針對擁有最小視差值的三角形投影映射至目標影像，直到較大視差值的三角形映射完成為止。

當兩個三角形區塊有重疊現象時，則我們選擇擁有較大視差值的三角形做為場景成像的區塊。因此，我們可以確定在前景的三角形對於在目標影像上相對於背景的三角形擁有較高的優先權。在目標投影的景深圖中，未被投影映射到的空

洞區域多半是由於遮蔽效應的問題所造成的結果，如圖 5-5 所示。

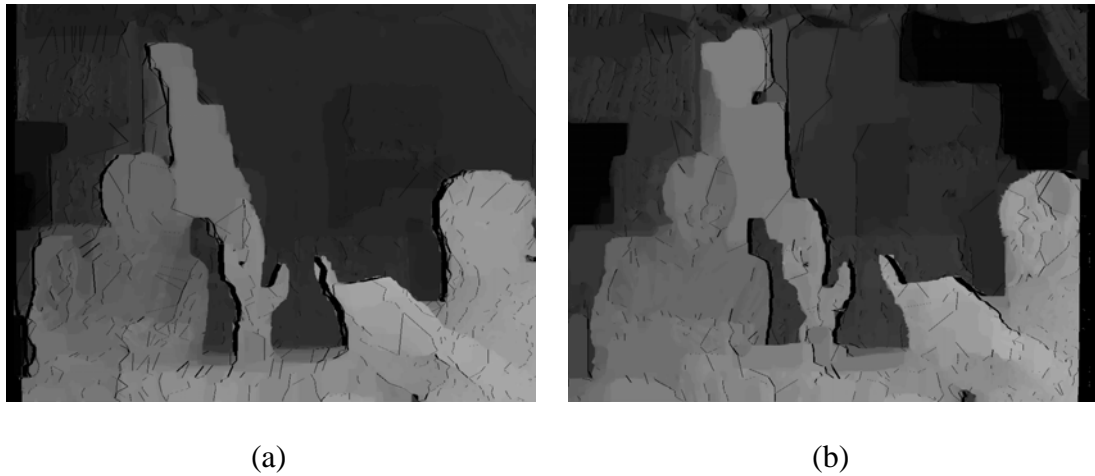


圖 5-5 MPEG 測試影像 “Newspaper” 經過三角映射後的景深圖，其中(a)由視點 04 經三角映射後的視點 05 的景深圖(b)由視點 06 經三角映射後的視點 05 的景深圖

5.3.4 針對映射後的景深圖進行後續濾波器處理 (Post-filtering on the Projected Depth Maps)



因為景深圖對於場景顯示成像的品質有關，所以景深圖的平滑化對於增強自然場景成像品質是必要的。首先，由於影像座標軸的運算精確度以及景深圖的不連續性，針對景深圖上可能會出現映射後所造成細小的裂痕。在式(2.2)中，影像座標 (u, v) 在投影映射的過程中因為精確度的關係，必須進位成最接近的整數值，可能會導致一個畫素寬的精確度錯誤。此錯誤通常我們選擇使用中值濾波器 (Medium filter) 可將此問題修復之。

然而，在投影映射的過程中不僅只有精確度的錯誤會造成空洞的區域，不規則性的景深變化可能導致在某些場景物體產生破洞區域。雖然大部分的區域都是可以經由低通濾波器將雜訊移除，但在影像的物體邊緣容易產生模糊化的問題。考量低通濾波器對於邊緣區域的影響，所以我們選擇使用雙向濾波器 (Bilateral filter) 以解決此問題，其定義關係式如(5.2)所示。

$$h(x) = k^{-1} \iint_D f(\theta)c(\theta, x)s(\theta, x)d\theta \quad (5.2)$$

其中 k 表示正規化的參數、 $c(\theta, x)$ 表示為畫素位置相近程度的分佈函數、 $s(\theta, x)$ 表示訊號間相似程度的分佈函數，其中 c 與 s 的表示式分別如式(5.3)與式(5.4)所示。

$$c(\theta, x) = \exp\left(-\frac{1}{2}\left(\frac{|\theta - x|}{\sigma_d}\right)^2\right) \quad (5.3)$$

其中 σ_d 表示尤拉距離的變異數、 θ 以及 x 表示訊號所在的畫素位置。

$$s(\theta, x) = \exp\left(-\frac{1}{2}\left(\frac{|f(\theta) - f(x)|}{\sigma_r}\right)^2\right) \quad (5.4)$$

其中 σ_r 表示色彩空間訊號間的變異數、 $f(\theta)$ 以及 $f(x)$ 表示訊號的強度。雙向濾波器是一種非線性濾波器，在剛開始被提出來是為了將影像上的雜訊有效地平滑化，且又可將重要的邊緣訊號保留著[9]。我們將圖 5-5(a)以及(b)經過中值濾波器以及雙向濾波器兩階段處理後，則可以得到圖 5-6(a)以及(b)的結果。



(a)

(b)

圖 5-6 MPEG 測試影像 “Newspaper” 經過後續濾波器處理後的景深圖，其中(a) 由視點 04 三角映射後的視點 05 景深圖經過後續濾波器的處理結果，(b) 由視點 06 三角映射後的視點 05 景深圖經過後續濾波器的處理結果

5.3.5 利用景深資訊將參考影像映射至虛擬視點畫面 (Froward Warping to

the Virtual View Based on Depth Maps)

我們利用左邊視點的參考影像投影映射到目標影像的平面上，另一方面，也將右邊視點的參考影像投影映射至目標影像的平面上， $I_L(u_L, v_L)$ 以及 $I_R(u_R, v_R)$ 各別代表左右邊參考影像上的影像色彩值，我們分別根據畫素位置上的視差值 $D_L(u_L, v_L)$ 以及 $D_R(u_R, v_R)$ ，經由正向投影映射製目標影像平面上，其表示如下式(5.5)以及(5.6)所示：

$$I_{LC}(x, y - D_L(x_L, v_L)) = I_L(x_L, v_L) \quad (5.5)$$

$$I_{RC}(x, y + D_R(x_R, v_R)) = I_R(x_R, v_R) \quad (5.6)$$

根據圖 5-6(a)以及(b)的景深圖資訊分別經過正向投影映射後，則可以得到分別為圖 5-7(a)以及(b)的結果。



圖 5-7 MPEG 測試影像 “Newspaper”由鄰近兩個攝影機視點位置正向投影映射後的結果，其中(a)由視點 04 投影映射至視點 05 的目標影像結果以及(b)由視點 06 投影映射至視點 05 的目標影像結果

5.3.6 左右兩張映射後的影像進行線性疊合 (Linear Blending of The Two Warped View)

從鄰近兩台相機所擷取的參考影像可利用線性疊合顯影成像，其數學表示如

式(5.7)所示。 $I(u, v)$ 表示位於虛擬視點影像平面上 (u, v) 的位置、 I_L 、 I_R 分別代表經投影映射後的左右參考影像。其中 α 表示相機距離的權重函數，如式(5.8)所示， t 代表相機參數中的外部參數位移向量[16]。

$$I(u, v) = \begin{cases} (1 - \alpha)I_{LC}(u_L, v_L) + \alpha I_{RC}(u_R, v_R) & (occ_L(u, v) = 0, occ_R(u, v) = 0) \\ I_{LC}(u_L, v_L) & (occ_L(u, v) = 0, occ_R(u, v) = 1) \\ I_{RC}(u_R, v_R) & (occ_L(u, v) = 1, occ_R(u, v) = 0) \\ 0 & (occ_L(u, v) = 1, occ_R(u, v) = 1) \end{cases} \quad (5.7)$$

$$\alpha = \frac{|t - t_L|}{|t - t_L| + |t - t_R|} \quad (5.8)$$

在式(5.7)中，若是左右視點產生的前向映射結果判斷所在畫素位置是否位於遮蔽區域，因而採用線性比例疊合後的目標影像訊號值，並將最後合成影像結果輸出。



5.3.7 影像邊緣擴張與影像修補 (Boundary Matting and Image Inpainting)

經由後續濾波器處理後的景深圖仍有被場景物體遮蔽而產生無法預測的區域，我們可以利用下式(5.9)與(5.10)找出鄰近兩張影像被遮蔽的未知區域。

$$occ_L(u, v) = \begin{cases} 1, & Z_R(u, v) < threshold \\ 0, & Z_R(u, v) > threshold \end{cases} \quad (5.9)$$

$$occ_R(u, v) = \begin{cases} 1, & Z_L(u, v) < threshold \\ 0, & Z_L(u, v) > threshold \end{cases} \quad (5.10)$$

其中 $Z(u, v)$ 表示映射後的景深圖深度值，經由映射後的景深圖，我們給予一個已知的臨界值(此實驗假設為 0)，當景深值小於某個臨界值，則判斷此畫素位置是無法預測的區域。由於攝影機拍攝過程失焦以及畫素精確度的問題，對於場景中的物體邊緣區域進行景深預估是特別困難的，常造成物體邊緣區域容易產生瑕疵現象。為了解決此問題，可以使用邊緣擴張方式將瑕疵現象減少並增進影像

品質。

$$occ_L(u, v) = 1 \quad (occ_L(u, v) = 1, Z_R(u, v) < Z_R(u + \Delta u, v)) \quad (5.11)$$

$$occ_R(u, v) = 1 \quad (occ_R(u, v) = 1, Z_L(u, v) < Z_L(u - \Delta u, v)) \quad (5.12)$$

在式(5.11)與(5.12)中， u 代表被遮蔽區域的寬度($u + \Delta u$ 代表擁有非零的景深值中最近的畫素位置)，表示我們將被遮蔽區域的邊界擴張並將另一張參考影像的資訊填補至此區域，其結果圖如圖 5-8 所示。其中圖 5-8(a)表示未經過影像擴張處理後的結果圖、(b)表示經過影像擴張處理後的結果圖。

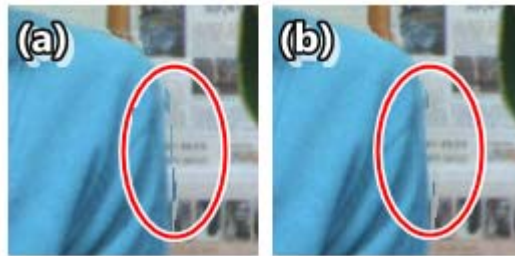


圖 5-8 MPEG 測試影像 “Newspaper” 經由影像擴張處理之後的結果圖

造成影像受損的原因有很多，像是遮蔽物影響、人為塗鴉和搬運時造成的刮痕等等。影像修補(Image Inpainting)是一個有效處理此問題的方法之一。

影像修補的概念即是據受破壞或遮蔽區域附近正確的殘存影像資訊，推斷遭受破壞的區域內未遭受破壞的區域中最合理的色彩及灰度值區塊，並複製經由方程式判斷最理想的影像區域回填影像受損的區域[30]。

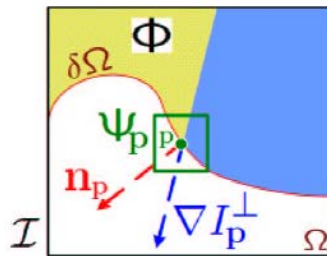


圖 5-9 待修補可靠度區域的梯度相關值示意圖

其中判斷的方程式如(5.13)所示， p 表示為待修復區域上的畫素位置，我們將可靠度方程式 $P(p)$ 分成權重函數 $C(p)$ 以及資料函數 $D(p)$ ，如式(5.14)及式(5.15)

所示。

$$P(p) = C(p)D(p) \quad , \quad p \in \delta\Omega \quad (5.13)$$

$$C(p) = \frac{\sum_{q \in \Psi_P \cap \Omega} C(q)}{|\Psi_P|} \quad (5.14)$$

$$D(p) = \frac{|\nabla I_P^\perp \cdot n_P|}{\alpha} \quad (5.15)$$

其中 $|\Psi_P|$ 代表所 Ψ_P 區塊內包含待修補的畫素總數、 α 代表正規化係數項，一般對於典型的灰階影像值為255、 n_P 代表與待修補區域點 p 邊緣正交垂直的單位向量。我們計算位於點 P 影像邊緣的梯度值以及待修補區域的梯度相關值 $D(p)$ ，找出具有最大可靠值的區塊 $\Psi_{\hat{p}}$ ，如式(5.16)所示。

$$\Psi_{\hat{p}} | \hat{p} = \arg \max_{p \in \delta\Omega^t} P(p) \quad (5.16)$$

接著計算出兩個理想區塊間的最小距離平方差和，如式(5.17)所示，並且將擁有最小平方差值的區塊取代目前待修復的區塊，其修復的順序如圖 5-10所示。

$$\Psi_{\hat{q}} = \arg \min_{\Psi_q \in \Phi} d(\Psi_{\hat{p}}, \Psi_q) \quad (5.17)$$

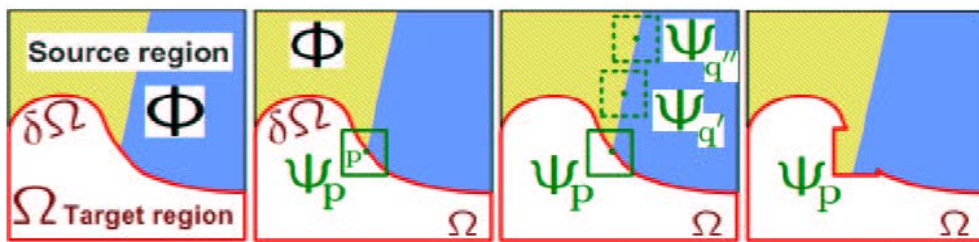


圖 5-10 影像修補的步驟示意圖

事實上，即使擁有左右兩張參考影像上的資訊仍然有些被遮蔽的區域是無法預測的，因此我們可以利用影像修補來解決此問題，如圖 5-11所示，其中圖 5-11(a)表示未經過影像修補前的合成影像結果、下圖(b)中的白色區域代表左右參考影像皆無法預測的被遮蔽區域。經由影像修復後的最後結果如圖 5-12所示。

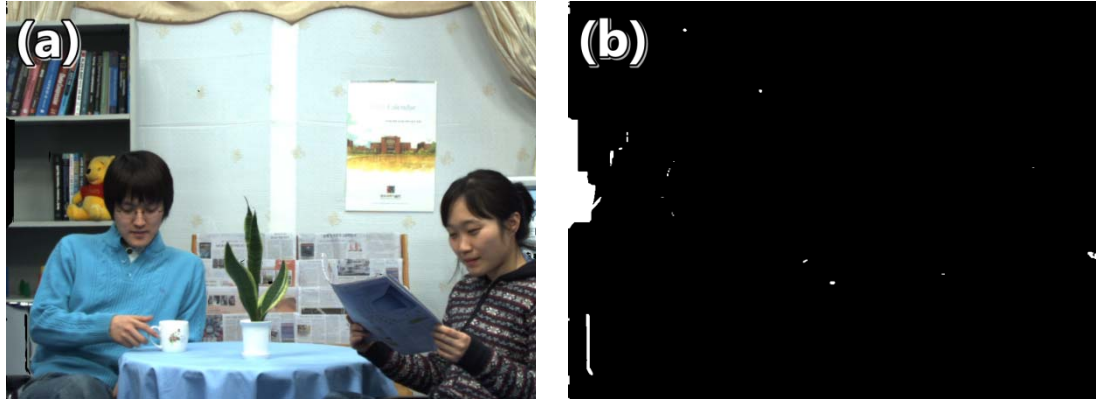


圖 5-11 未經過影像修補前的合成影像結果以及待修補的區域



圖 5-12 經過影像修補後的合成影像結果

5.4 模擬與討論 (Simulation and Results)

在本章節中，我們將會展示新提出的演算法實驗結果。實驗的環境設定與 4.3.1 所述相同。在本章節的最後，我們也將討論許多環境因素對實驗結果的影響。

5.4.1 模擬結果

實驗過程分成二大部分，並在本章節的最後，將會針對不同的視點合成演算法詳加討論及分析視點合成影像的結果加以探討與說明。

- 實驗的第一部分：利用 MPEG 測試影像提供的景深圖當作是輸入測試影像的對應景深圖，並觀察新提出的演算法與 VSRS 2.0 間的合成影像平均 PSNR 表現。
- 實驗的第二部份：根據擷取不同的特徵點個數並觀察該演算法以及 VSRS 2.0 演算法的合成結果平均 PSNR 的表現。



5.4.1.1 實驗第一部份：MPEG 測試影像提供的景深圖對於合成結果的表現

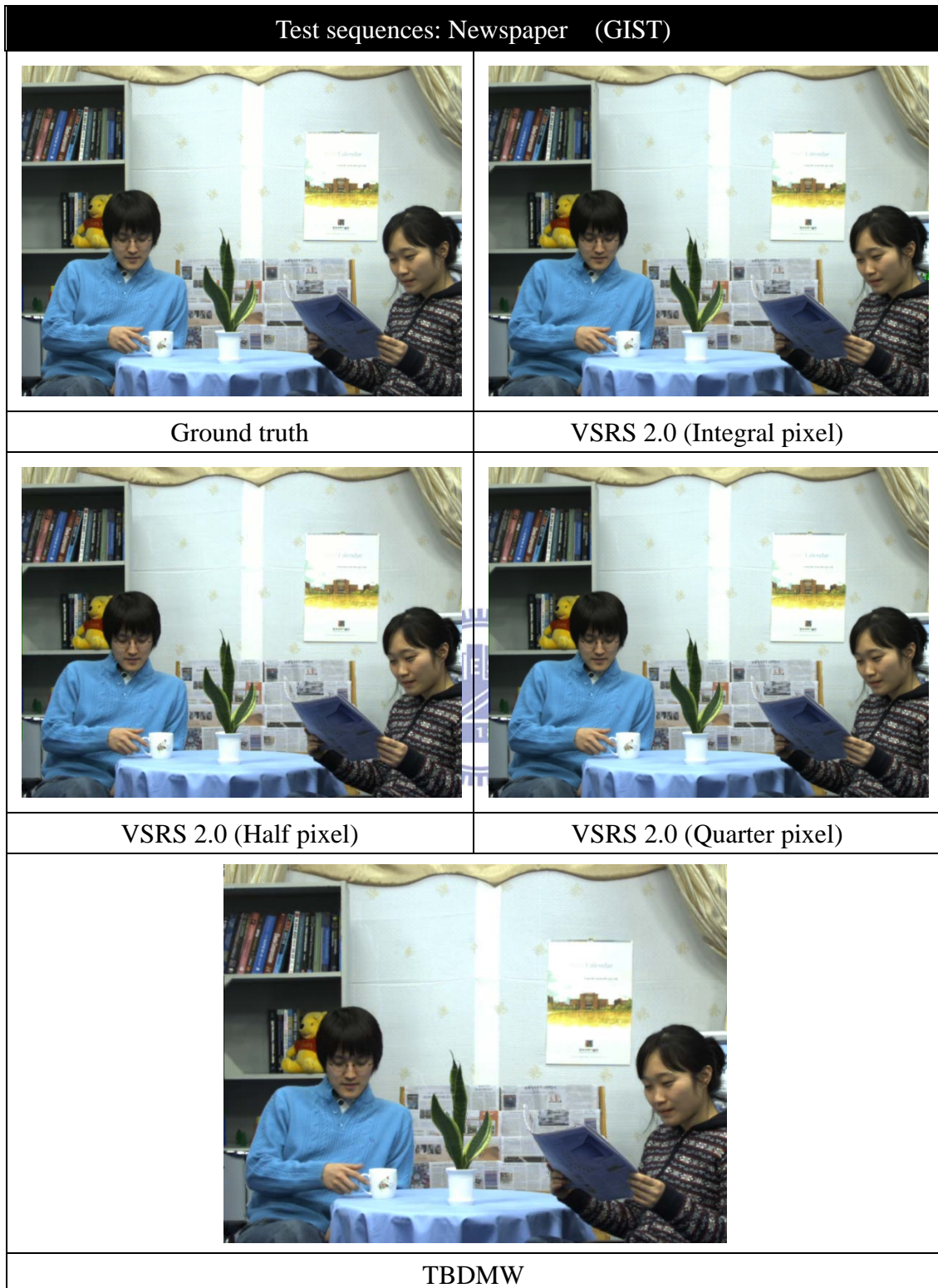


圖 5-13 測試影像 “Newspaper” 由視點 04 與視點 06 所合成的虛擬視點 05 結果比較圖

Test sequences: Newspaper (GIST)			
			
			
Ground truth		VSRS 2.0 (Integral pixel)	
			
			
VSRS 2.0 (Half pixel)		VSRS 2.0 (Quarter pixel)	
			
			
TBDMW			

圖 5-14 測試影像“Newspaper”由視點 04 與視點 06 所合成的虛擬視點 05 細部結果放大比較圖

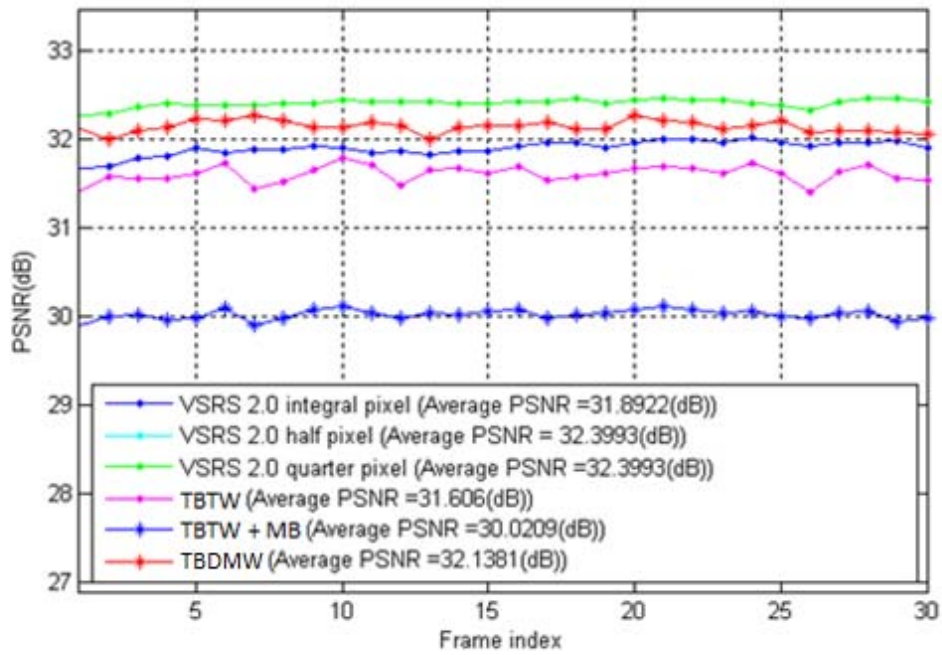


圖 5-15 不同的視點合成演算法在測試影像“Newspaper”視點 05 的前 30 張影像 PSNR 表現

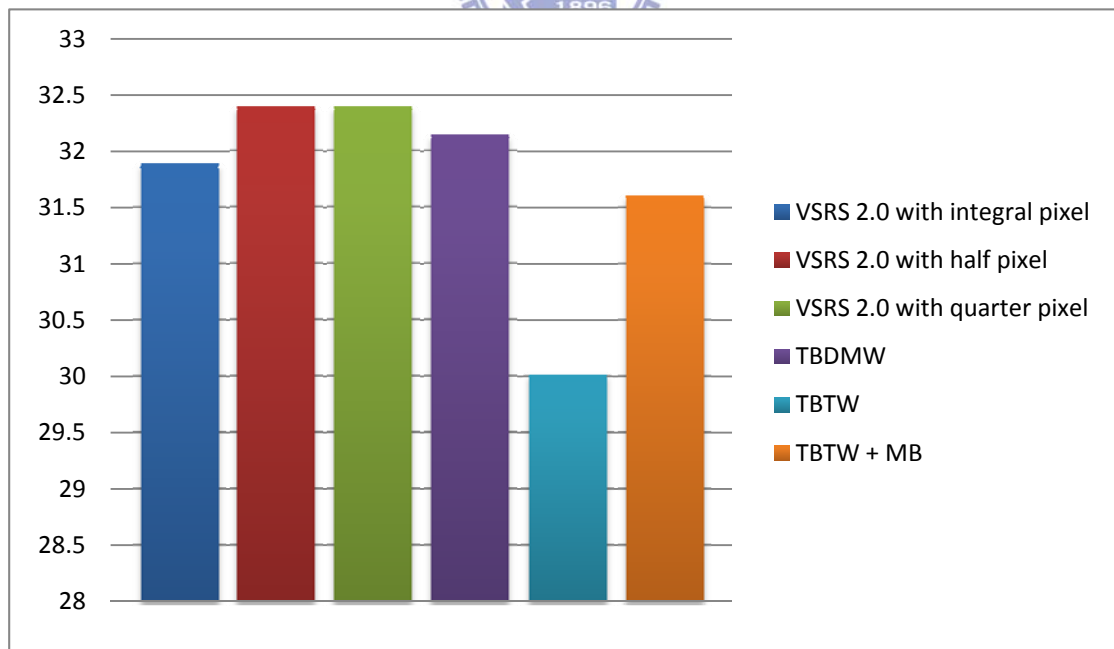


圖 5-16 不同的視點合成演算法針對測試影像“Newspaper”視點 05 的前 30 張影像上的平均 PSNR 表現

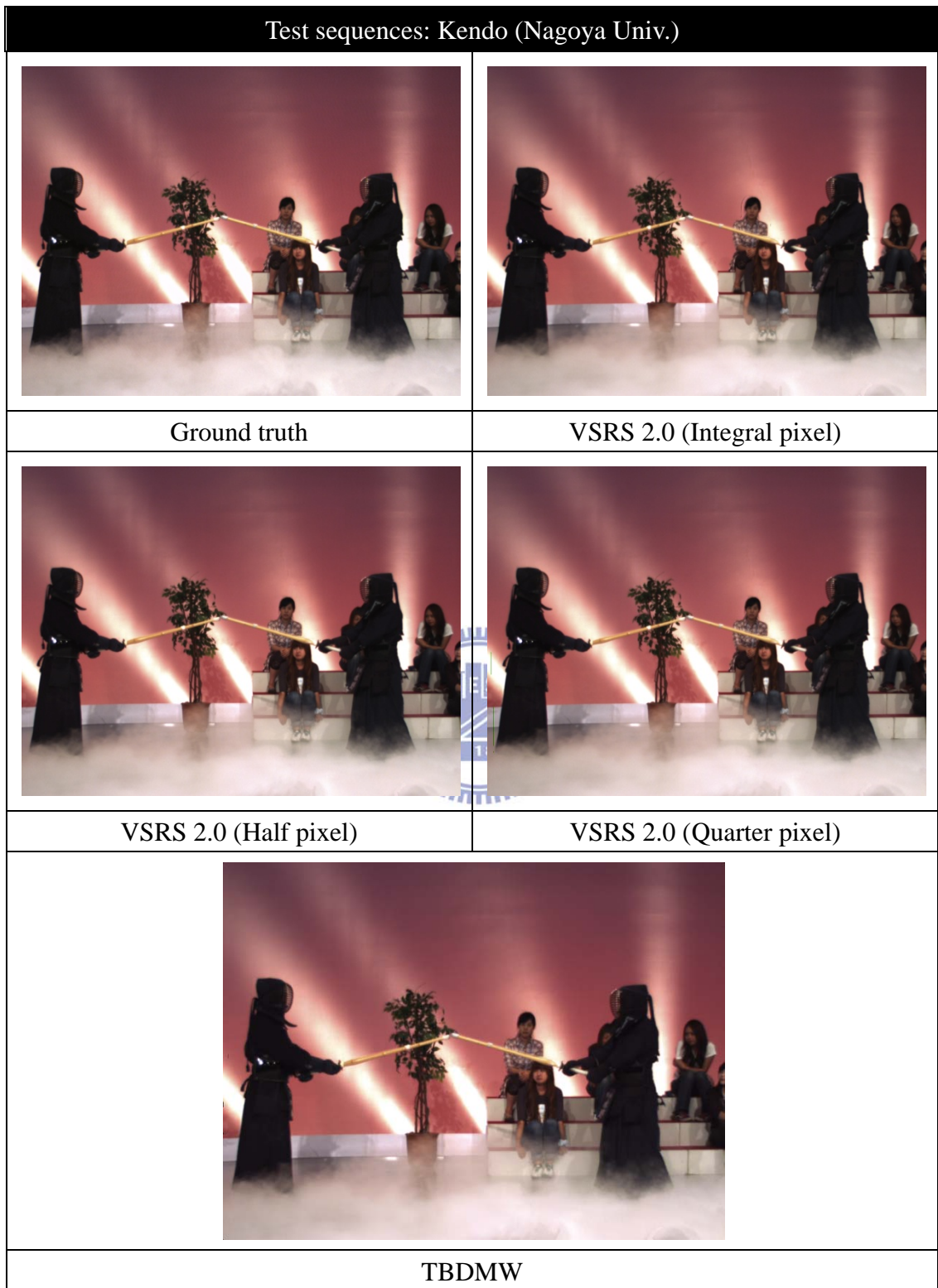


圖 5-17 測試影像“Kendo”由視點 03 與視點 05 所合成的虛擬視點 04 結果比較圖


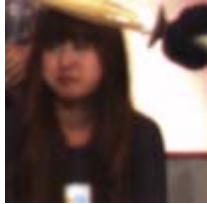
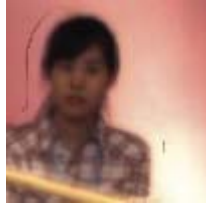




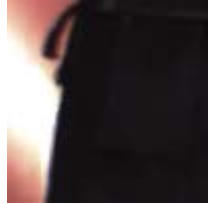
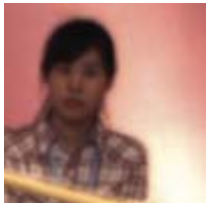

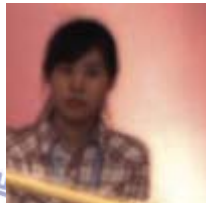
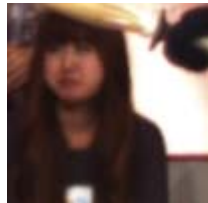
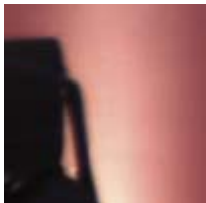

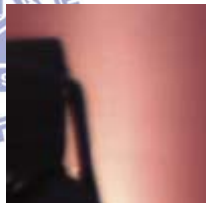
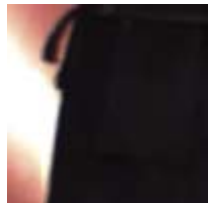
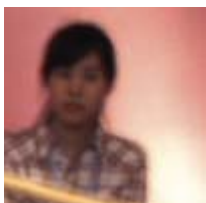
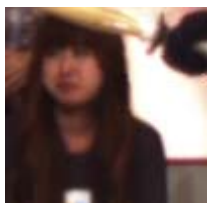
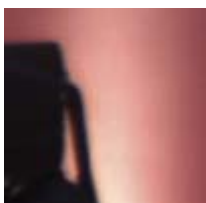

Test sequences: Kendo (Nagoya Univ.)			
			
			
Ground truth		VSRS 2.0 (Integral pixel)	
			
			
VSRS 2.0 (Half pixel)		VSRS 2.0 (Quarter pixel)	
			
			
TBDMW			

圖 5-18 測試影像“Kendo”由視點 03 與視點 05 所合成的虛擬視點 04 細部結果放大比較圖

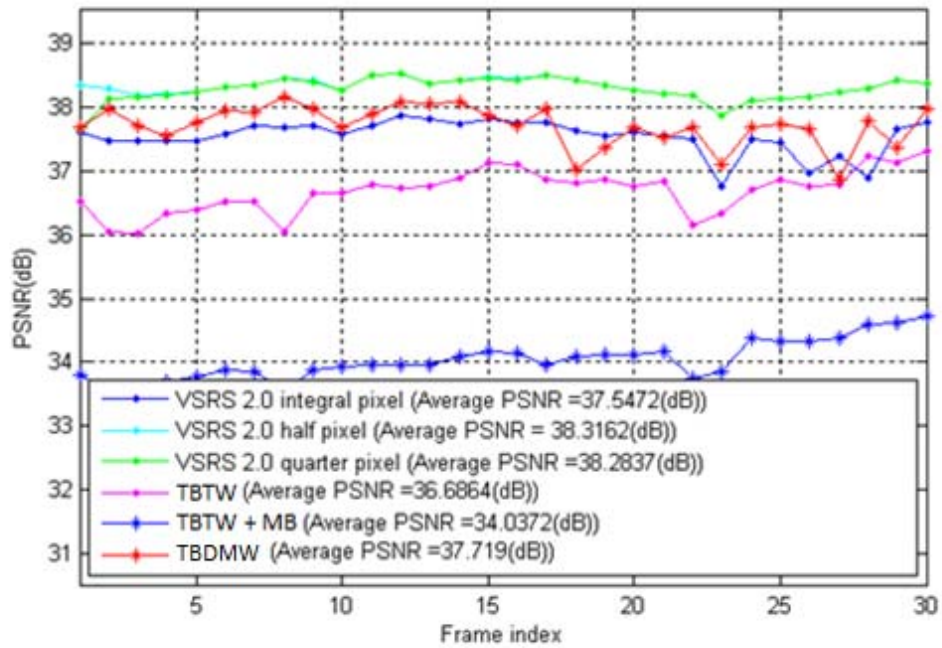


圖 5-19 不同的視點合成演算法在測試影像 “Kendo” 視點 04 的前 30 張影像 PSNR 表現

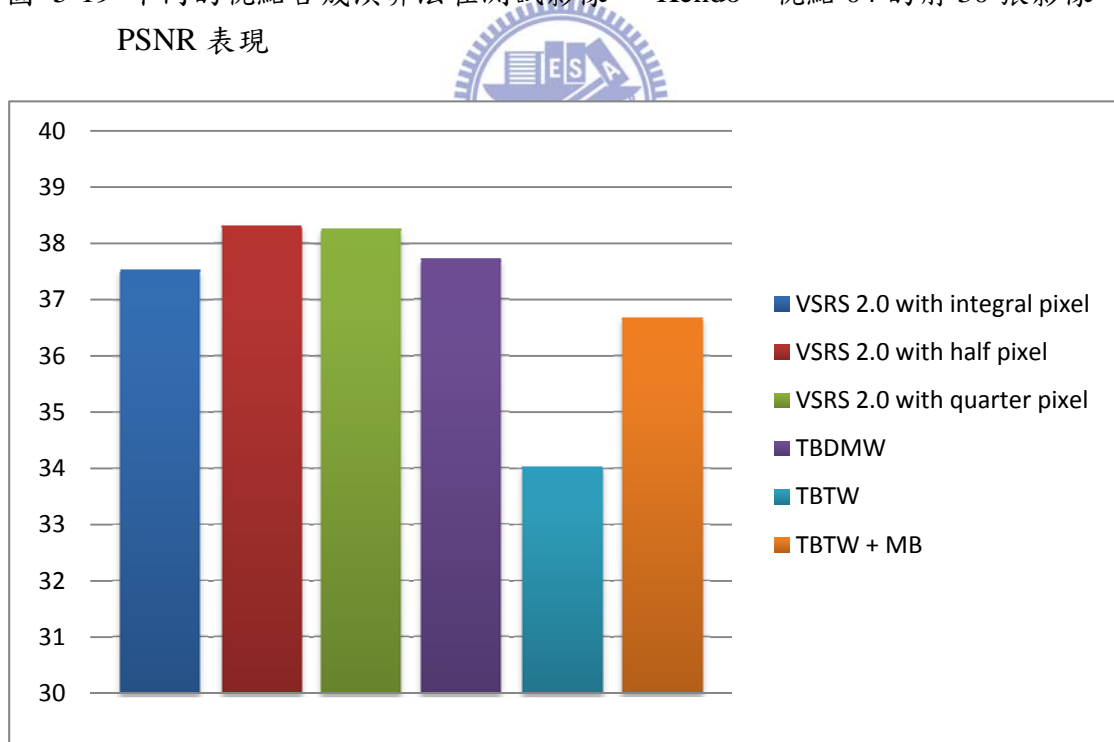


圖 5-20 不同的視點合成演算法針對測試影像 “Kendo” 視點 04 的前 30 張影像上的平均 PSNR 表現

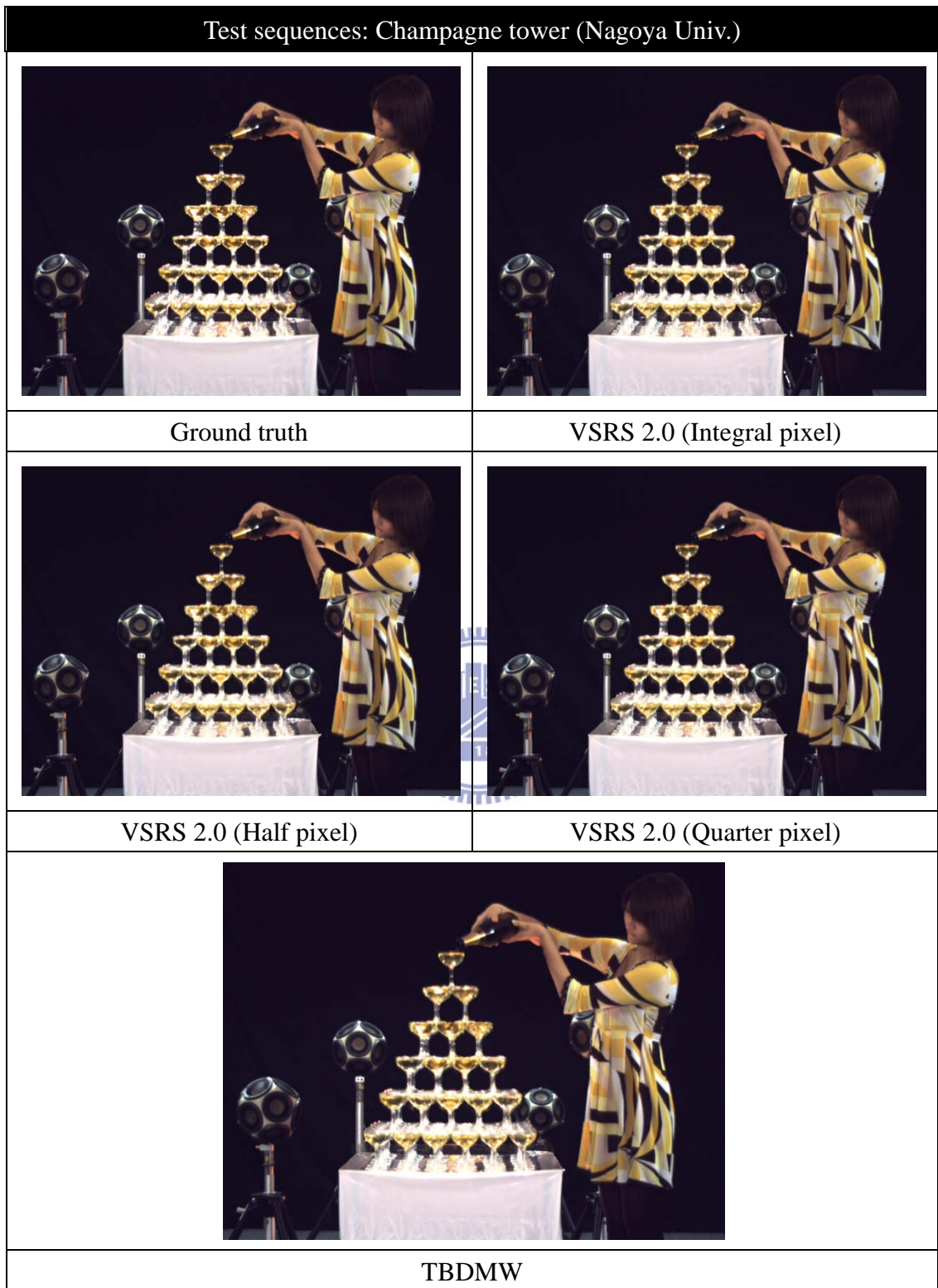


圖 5-21 測試影像 “Champagne tower” 由視點 39 與視點 41 所合成的虛擬視點 40 結果比較圖

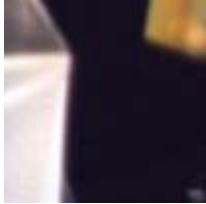


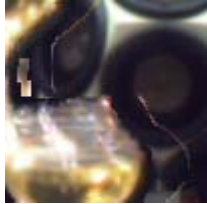
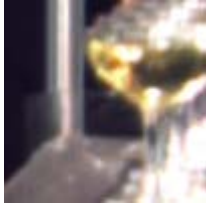
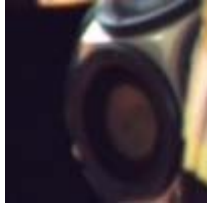
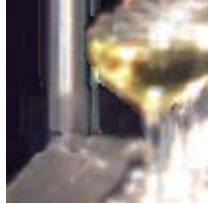
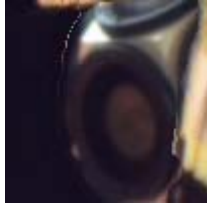





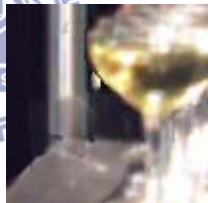



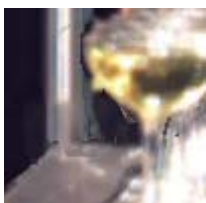
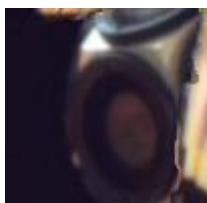
Test sequences: Champagne tower (Nagoya Univ.)			
			
			
Ground truth		VSRS 2.0 (Integral pixel)	
			
			
VSRS 2.0 (Half pixel)		VSRS 2.0 (Quarter pixel)	
			
			
TBDMW			

圖 5-22 測試影像 “Champagne tower”由視點 39 與視點 41 所合成的虛擬視點 40 細部結果放大比較圖

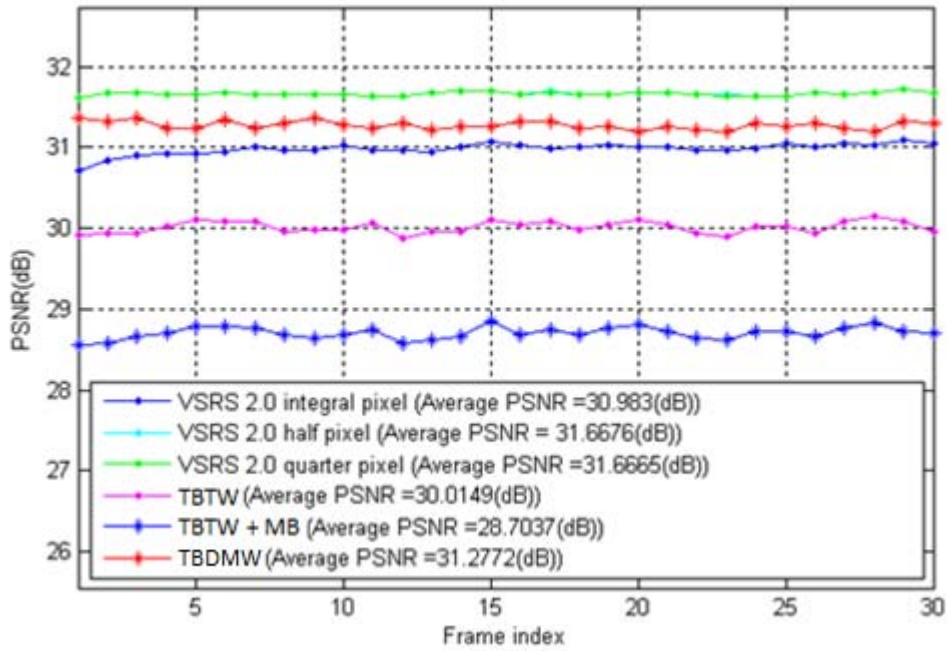


圖 5-23 不同的視點合成演算法在測試影像 “Champagne tower” 視點 40 的前 30 張影像 PSNR 表現

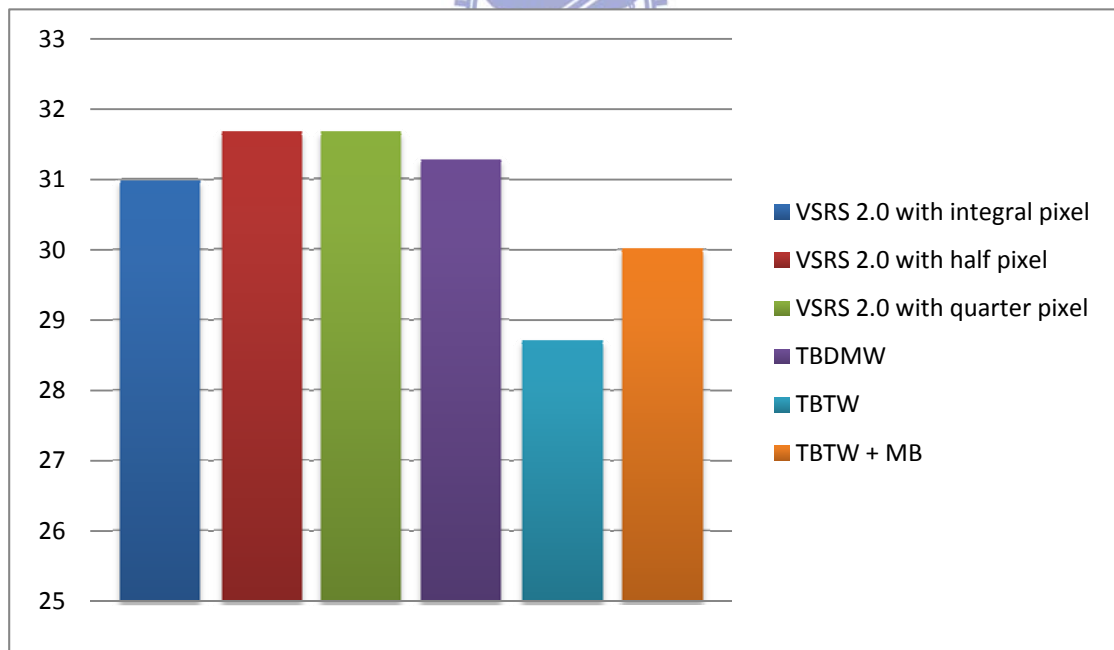


圖 5-24 不同的視點合成演算法針對測試影像 “Champagne tower” 視點 40 的前 30 張影像上的平均 PSNR 表現

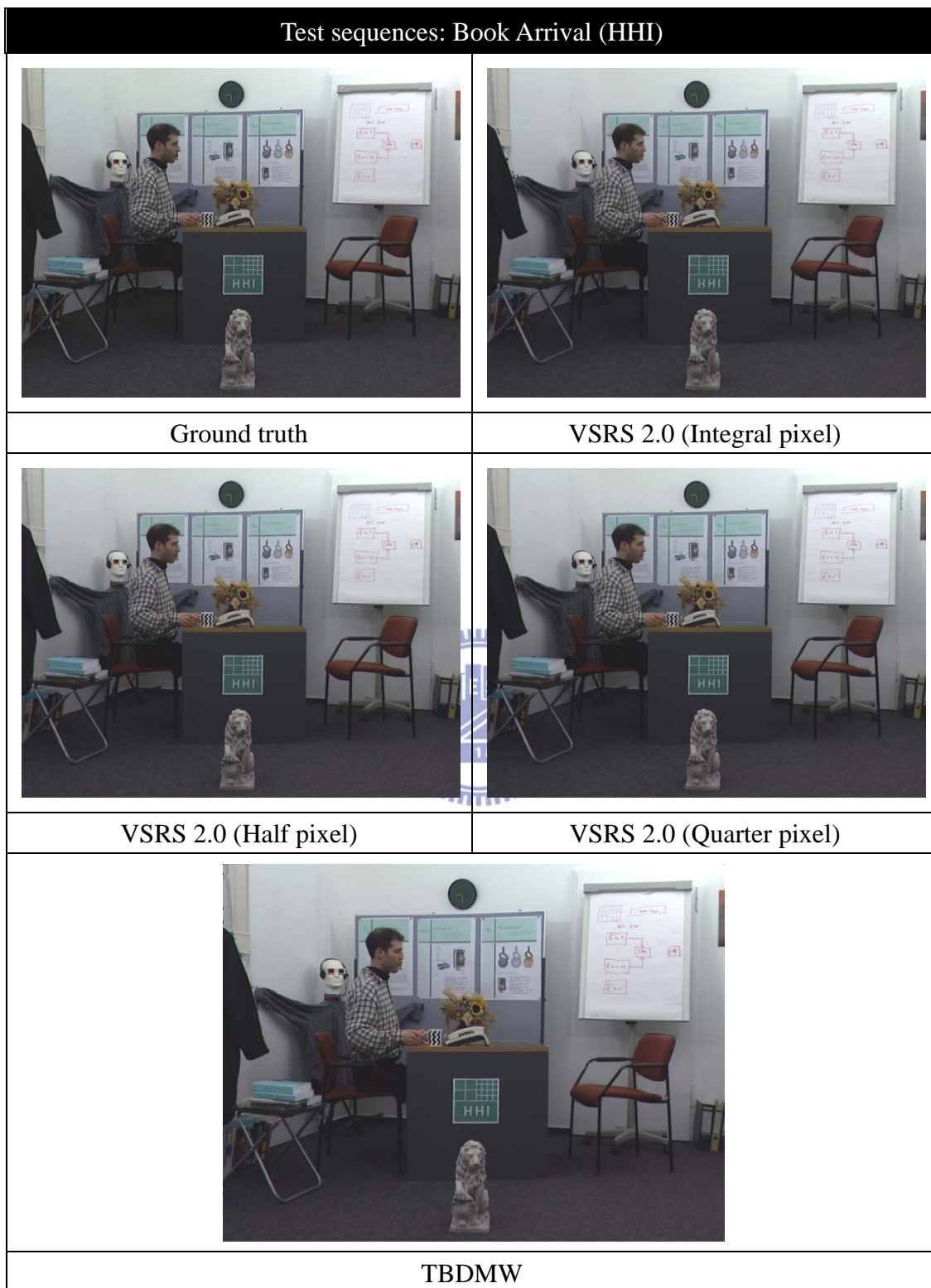


圖 5-25 測試影像 “Book arrival”由視點 08 與視點 10 所合成的虛擬視點 09 結果比較圖





















Test sequences: Book Arrival (HHI)			
			
			
Ground truth		VSRS 2.0 (Integral pixel)	
			
			
VSRS 2.0 (Half pixel)		VSRS 2.0 (Quarter pixel)	
			
			
TBDMW			

圖 5-26 測試影像“Book arrival”由視點 08 與視點 10 所合成的虛擬視點 09 細部結果放大比較圖

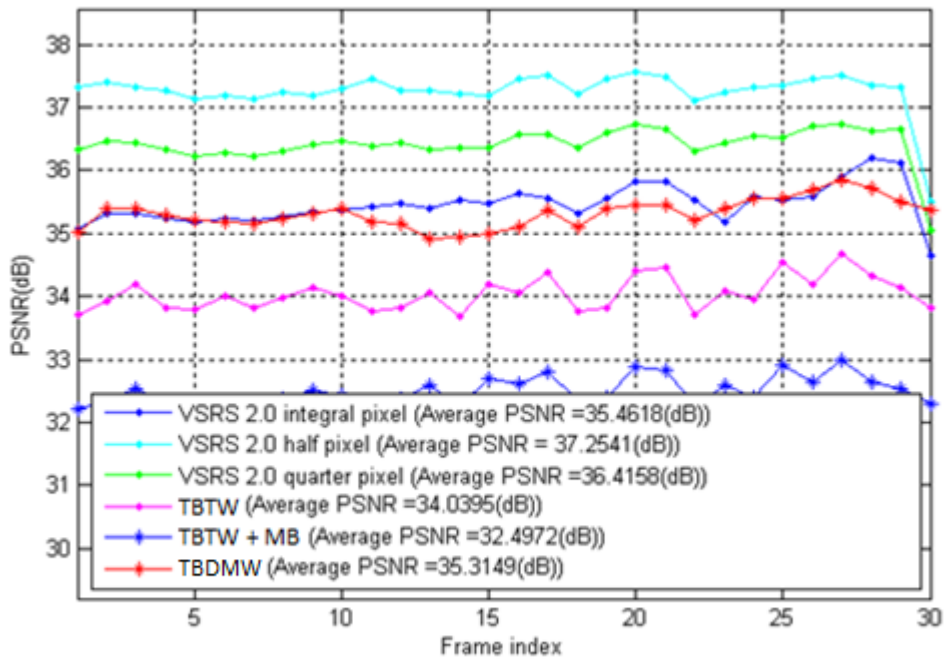


圖 5-27 不同的視點合成演算法在測試影像 “Book arrival” 視點 10 的前 30 張影像 PSNR 表現

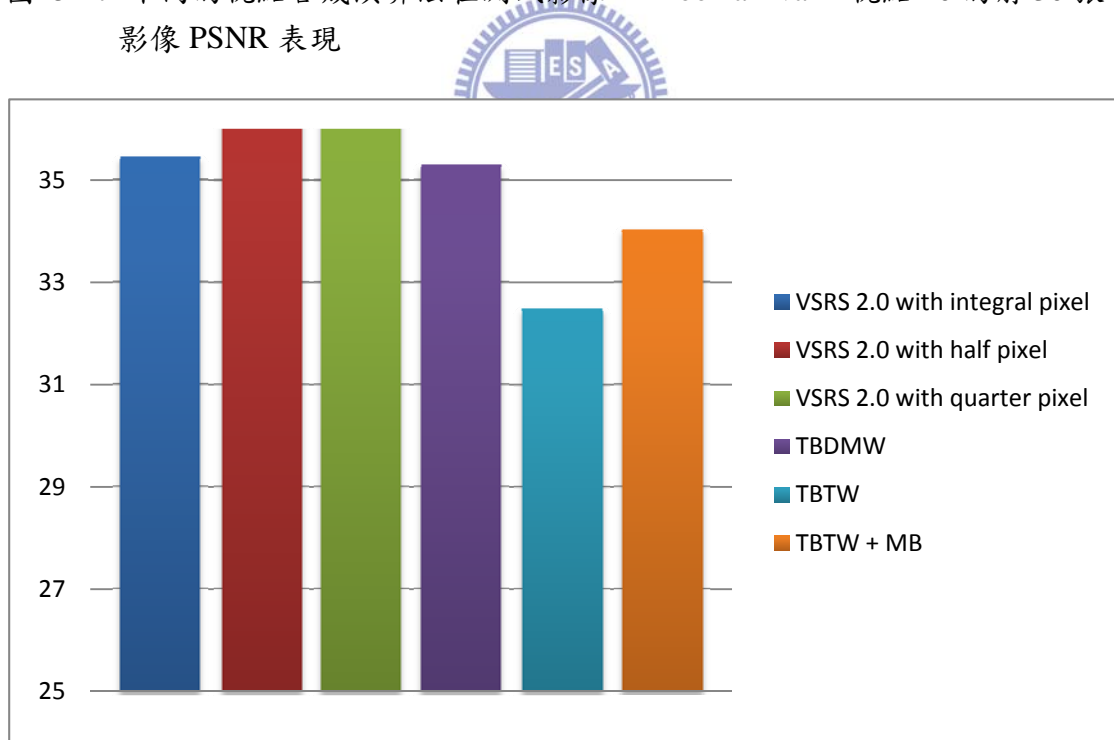


圖 5-28 不同的視點合成演算法針對測試影像 “Book arrival” 視點 10 的前 30 張影像上的平均 PSNR 表現

分析與討論

表 14 表示我們分別針對 MPEG 所提供四種不同的 FTV 測試影像：“Book arrival”、“Champagne tower”、“Newspaper”以及“Kendo”的前 30 張影像進行視點合成演算法的平均 PSNR 的結果。有別於先前提出的演算法以及 VSRS 2.0 的結果顯示，新提出的演算法的平均 PSNR 結果較傳統的演算法成長約 0.17dB 到 1.7dB 不等。以主觀視覺感受來看，整體的視覺感受也有較佳的表現。


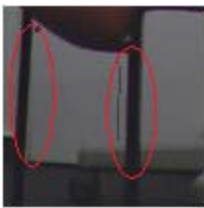


表 14 MPEG 測試影像前 30 張合成影像的平均 PSNR 效能表現

Test sequences	Newspaper	Champagne tower	Kendo	Book Arrival
Method in Chapter 4				
TBTW + MB	30.02	28.70	34.04	32.50
TBTW	31.61	30.01	36.69	34.04
VSRS 2.0				
VSRS 2.0 with integral pixel	31.89	30.98	37.55	35.46
VSRS 2.0 with half pixel	32.40	31.67	38.32	37.25
VSRS 2.0 with quarter pixel	32.40	31.67	38.28	36.42
Method in Chapter 5				
TBDMW	32.14	31.28	37.72	35.31
$\Delta PSNR$ (dB)	0.25	0.3	0.17	-0.15
$\Delta PSNR = PSNR_{proposed} - PSNR_{VSRS\ 2.0}$				

對於測試影像“Book arrival”而言，圖 5-25 表示為我們所提出的演算法架構以及參考軟體 VSRS 2.0 以整數單位畫素的視點 09 合成結果，合成影像結果可以發現在人的頭部附近以及桌椅較細小的握把部分容易產生瑕疵現象，如所示。以平均 PSNR 而言，我們所提出的演算法較傳統視點合成的演算法低約 0.15 dB 左右。以主觀視覺而言，MPEG 測試影像“Book arrival”的場景中，合成影像的椅

子握把部分仍然有許多肉眼可見的不連續性產生，但合成結果已經相當不錯，如表 15 所示。新提出的演算法對於測試影像 “Book arrival” 在最後合成影像的細部表現較 VSRS 2.0 的結果整體表現是差不多的。若 VSRS 2.0 使用以半畫素或四分之一的畫素合成影像的結果，其計算的平均 PSNR 值仍較低 0.2 到 0.6dB 左右不等，原因在於 TBDMW 演算法僅針對整數單位的畫素映射，仍有數值精確度的錯誤待需改進。

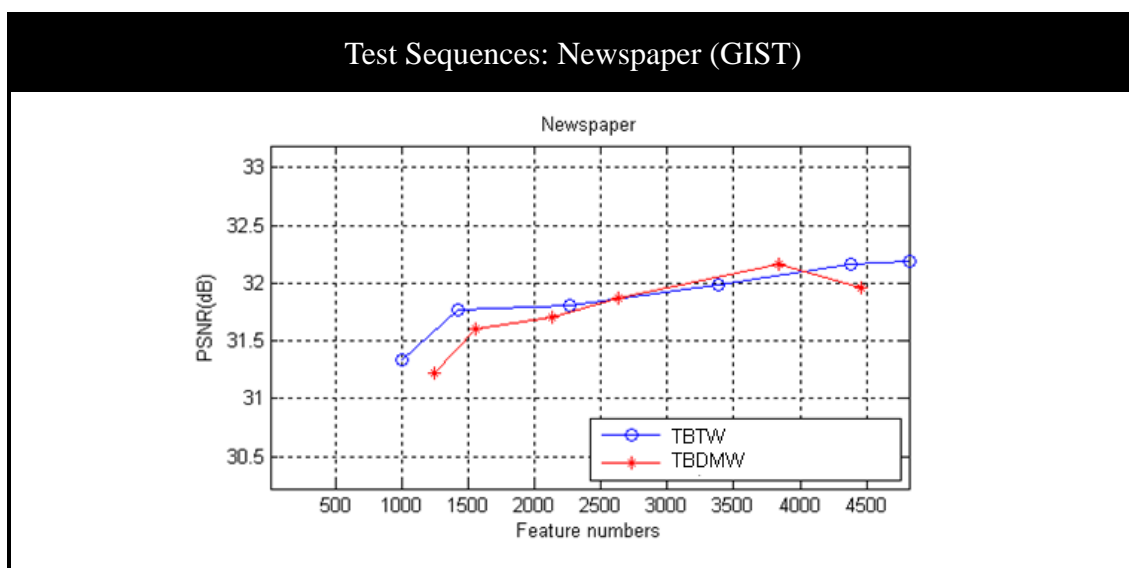
表 15 MPEG 測試影像 “Book arrival” 對於不同演算法的細部放大結果比較圖

Test sequences: Book arrival (HHI)			
			
Ground truth	VSRS 2.0 with Integral Pixel	TBTW	TBDMW

5.4.1.2 實驗第二部份：特徵點個數對於合成結果的影響與 PSNR 表現

本章節我們將會調整演算法內部參數使得調整擷取特徵點的個數，並採用 MPEG 測試影像所提供的景深圖，觀察 4.2 節所提出的 TBTW 演算法與本章節提出的 TBDMW 演算法之間的合成結果 PSNR 表現以及合成影像品質。

表 16 演算法在不同特徵點的個數下對於 MPEG 測試影像 “Newspaper” 的 PSNR 曲線變化圖



Feature numbers	Test Sequences: Newspaper (GIST)			
Ground truth				
999				
1420				

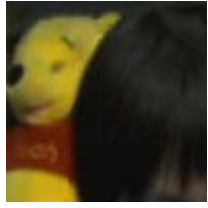
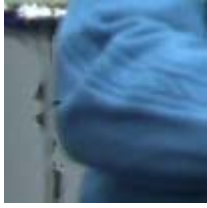


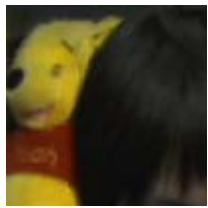



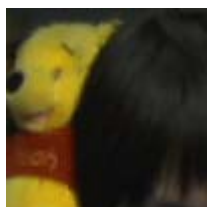



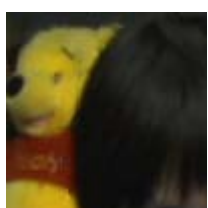
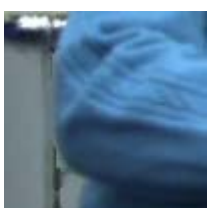


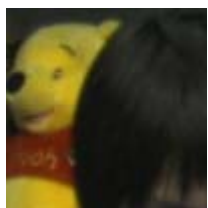
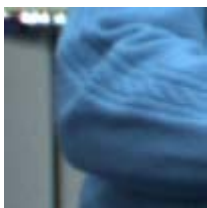


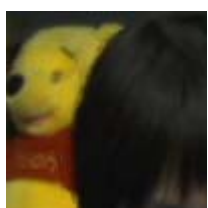
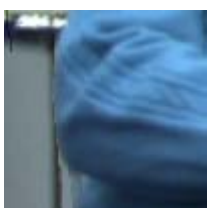


2261				
3383				
4388				
4822				

圖 5-29 TBTW 演算法在不同特徵點的數目下對於MEPG 測試影像 “Newspaper” 的視點 05 合成目標細部放大圖

Feature numbers	Test Sequences: Newspaper (GIST)			
Ground truth				
1237				

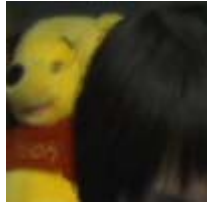



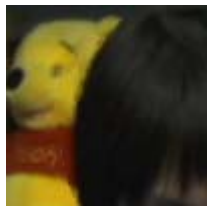



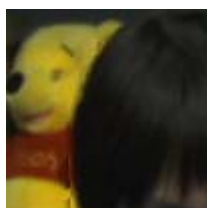
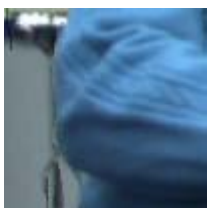


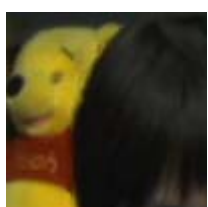
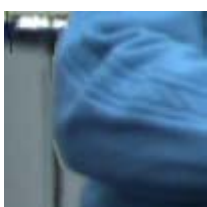


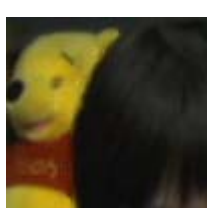



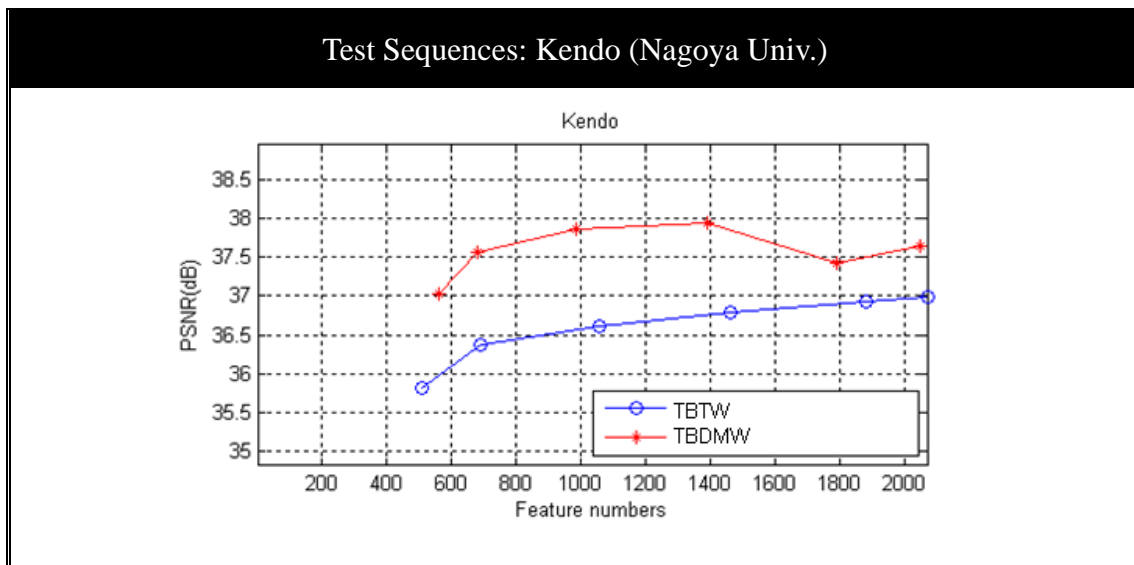
1552				
2126				
2625				
3842				
4463				

圖 5-30 TBDMW 演算法在不同特徵點的數目下對於 MPEG 測試影像
“Newspaper” 的視點 05 合成目標細部放大圖

表 17 演算法在不同特徵點的個數下對於 MPEG 測試影像 “Kendo” 的 PSNR 曲線變化圖



Feature numbers	Test Sequences: Kendo (Nagoya Univ.)			
Ground truth				
507				
690				


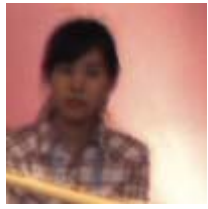
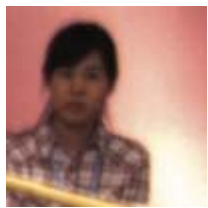
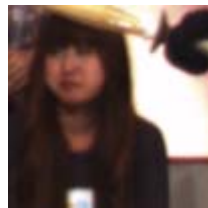
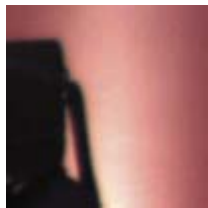
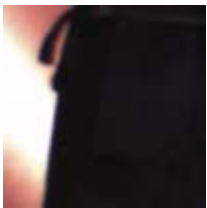
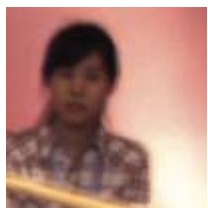
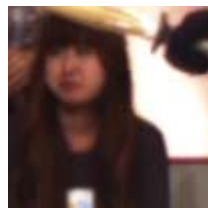
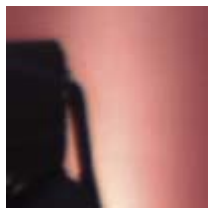
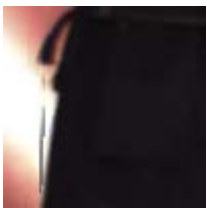
1055				
1460				
1884				
2070				

圖 5-31 TBTW 演算法在不同特徵點的數目下對於MEPG 測試影像” “Kendo” 的視點 04 合成目標細部放大圖

Feature numbers	Test Sequences: Kendo (Nagoya Univ.)			
Ground truth				
563				

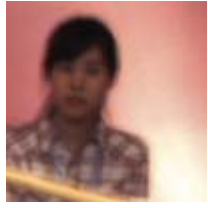
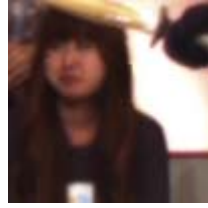
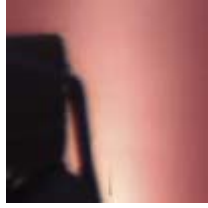

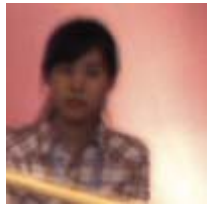
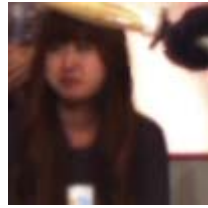


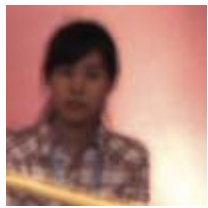
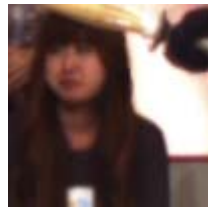


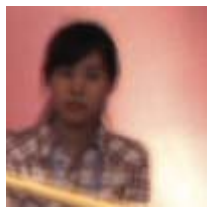
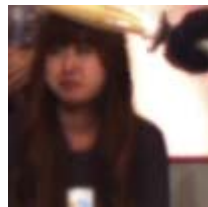
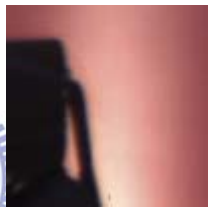
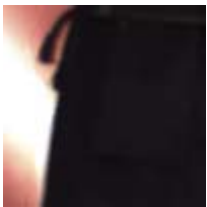
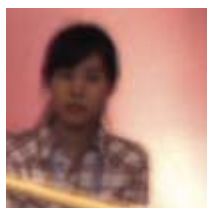
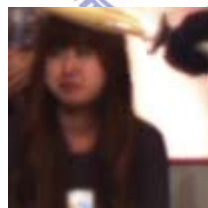
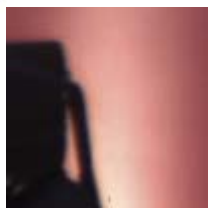
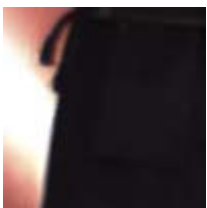
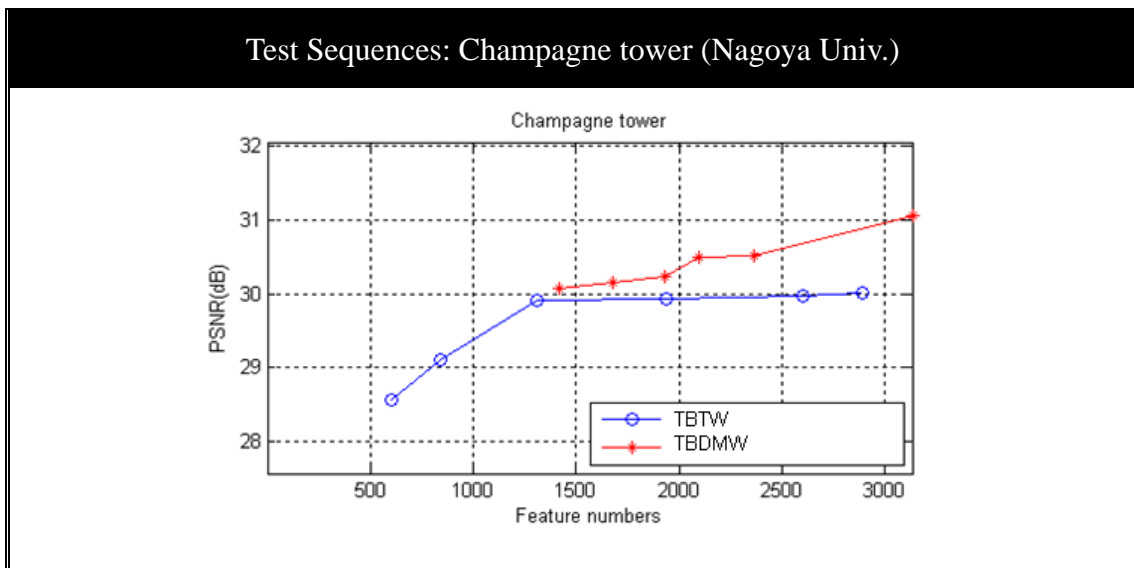



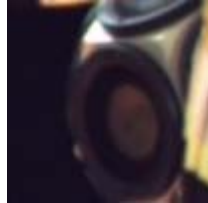

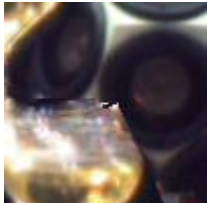




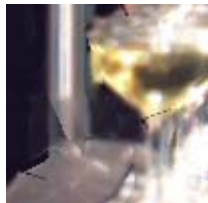
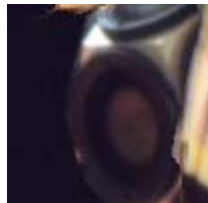
678				
986				
1389				
1790				
2050				

圖 5-32 TBDMW 演算法在不同特徵點的數目下對於 MPEG 測試影像 “Kendo” 的視點 04 合成目標細部放大圖

表 18 演算法在不同特徵點的個數下對於 MPEG 測試影像 “Champagne tower” 的 PSNR 曲線變化圖



Feature numbers	Test Sequences: Champagne tower (Nagoya Univ.)			
Ground truth				
603				
840				

1313				
1941				
2604				
2890				

圖 5-33 TBTW 演算法在不同特徵點的數目下對於 MPEG 測試影像“Champagne tower”的視點 40 合成目標細部放大圖

Feature numbers	Test Sequences: Champagne tower (Nagoya Univ.)			
Ground truth				
1422				

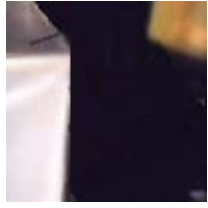

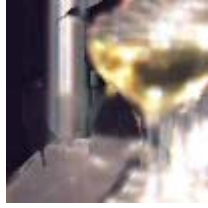

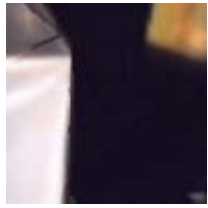
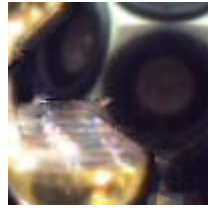
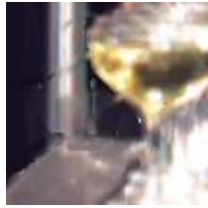

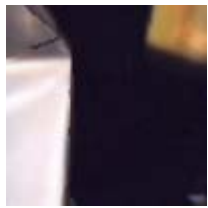
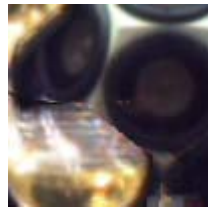
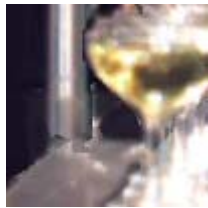

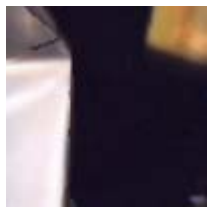

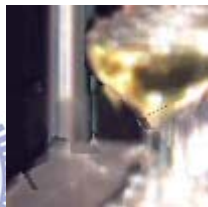
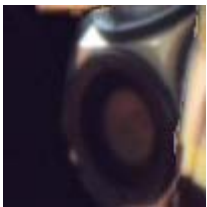
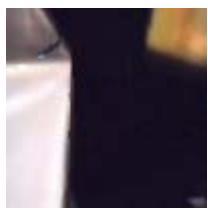

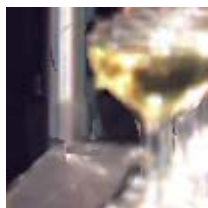
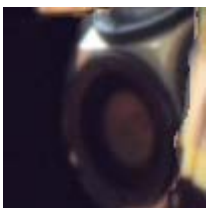
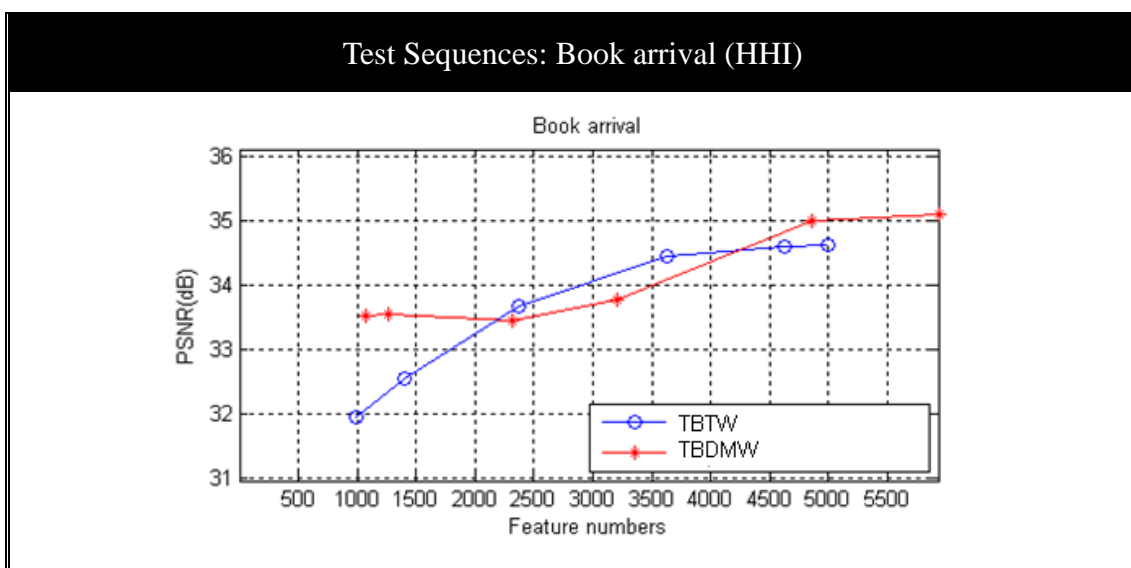
1681				
1931				
2094				
2366				
3132				

圖 5-34 TBDMW 演算法在不同特徵點的數目下對於 MPEG 測試影像
“Champagne tower” 的視點 40 合成目標細部放大圖

表 19 演算法在不同特徵點的個數下對於 MPEG 測試影像 “Book arrival” 的 PSNR 曲線變化圖



Feature numbers	Test Sequences: Book arrival (HHI)			
Ground truth				
996				
1399				







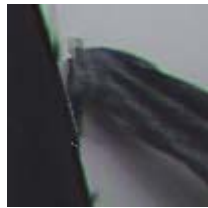

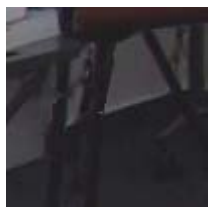
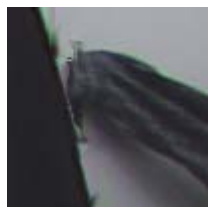
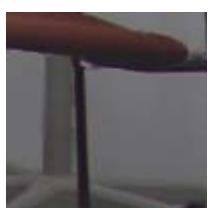

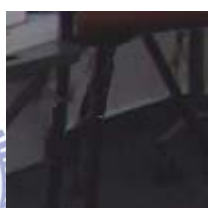
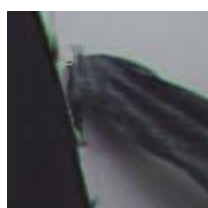






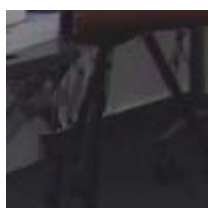
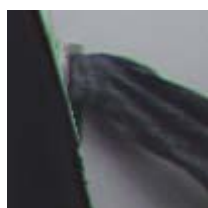
2370				
3632				
4624				
4990				

圖 5-35 TBTW 演算法在不同特徵點的數目下對於 MPEG 測試影像 “Book arrival” 的視點 09 合成目標細部放大圖

Feature numbers	Test Sequences: Book arrival (HHI)			
Ground truth				
1076				




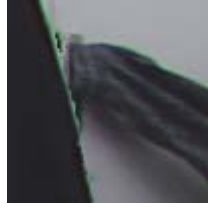




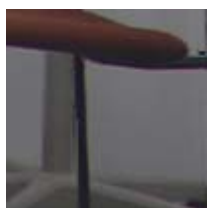



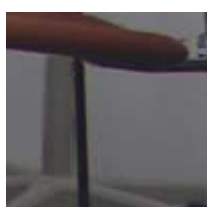

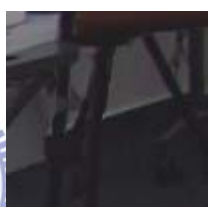
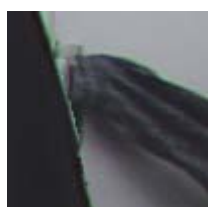
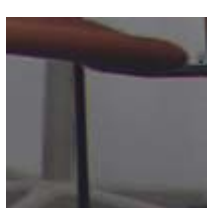

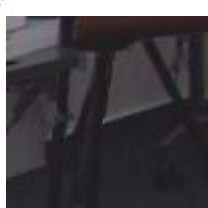
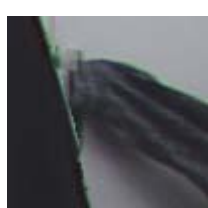
1264				
2314				
3206				
4859				
5931				

圖 5-36 TBDMW 演算法在不同特徵點的數目下對於 MPEG 測試影像 “Book arrival” 的視點 09 合成目標細部放大圖

分析與討論

表 20 展示為不同 MPEG 測試影像針對不同的特徵點個數在 PSNR 上的表現，對於測試影像 “Champagne tower”、 “Kendo”以及 “Book arrival”，新提出的演算法確實對於整體的 PSNR 有較成長約略 0.4 至 0.8dB 不等。然而，對於 MPEG 測試影像 “Newspaper”而言，PSNR 雖然有不錯的表現，但是與前面章節我們所提出的演算法比較，影像品質並無明顯差異。

表 20 演算法在不同特徵點個數對於 MPEG 測試影像的 PSNR 表現

Method in Chapter 4: TBTW							
Newspaper		Champagne tower		Kendo		Book arrival	
Feature Pionts	PSNR	Feature Pionts	PSNR	Feature Pionts	PSNR	Feature Pionts	PSNR
999	31.34	603	28.56	507	35.82	996	31.97
1420	31.77	840	29.10	690	36.38	1399	32.56
2261	31.81	1313	29.92	1055	36.61	2370	33.67
3383	31.99	1941	29.93	1460	36.80	3632	34.46
4388	32.16	2604	29.97	1884	36.93	4624	34.61
4822	32.19	2890	30.02	2070	36.99	4990	34.64
Method in Chapter 5: TBDMW							
Newspaper		Champagne tower		Kendo		Book arrival	
Feature Pionts	PSNR	Feature Pionts	PSNR	Feature Pionts	PSNR	Feature Pionts	PSNR
1237	31.22	1422	30.08	563	37.02	1076	33.53
1552	31.60	1681	30.16	678	37.57	1264	33.55
2126	31.70	1931	30.24	986	37.86	2314	33.47
2625	31.87	2094	30.50	1389	37.94	3206	33.79
3842	32.17	2366	30.52	1790	37.42	4859	35.01
4463	31.96	3132	31.06	2050	37.64	5931	35.12

對於所有的測試影像而言，特徵點的個數越多對於 PSNR 的整體表現越佳，換句話說，三角形的切割數目越多對 PSNR 表現及影像品質越好。然而，三角形的切割數目若是過於精細則與畫素對畫素映射方式是差不多的，因此 PSNR 數值

也有趨於飽和的現象產生。因此發展一個好的評判準則判定整體畫面切割的三角形個數是門非常重要的研究議題。

表 21 特徵點個數於演算法的運行時間比較表

Test sequences : Newspaper (GIST)					
Method in Chapter 4: TBTW			Method in Chapter 5 : TBDMW		
Feature Pionts	PSNR (dB)	Executive time (sec)	Feature Pionts	PSNR (dB)	Executive time (sec)
999	31.34	27.43	507	35.82	57.77
1420	31.77	28.83	690	36.38	58.05
2261	31.81	30.94	1055	36.61	59.47
3383	31.99	33.68	1460	36.80	60.29
4388	32.16	36.59	1884	36.93	63.98
4822	32.19	38.13	2070	36.99	66.07

Feature numbers	TBTW Executive time (sec)	TBDMW Executive time (sec)
999	27.43	57.77
1420	28.83	58.05
2261	30.94	59.47
3383	33.68	60.29
4388	36.59	63.98
4822	38.13	66.07

事實上，若是演算法皆使用畫素對畫素的映射方法，對於紋理複雜的區域有相當良好的效能，但在映射的過程中卻耗費大量的運算資源。在表 21 中，我們對於 MPEG 測試影像 “Newspaper” 可以發現只對於紋理部分使用三角形的映射，並使用 MATLAB 2008a 來模擬實驗過程的運算時間，其計算時間比起使用畫素對畫素的映射方法多了約 30sec 的時間。從表 21 中可得知，當擷取的特徵點個

數越多，相對於整體的運算量有線性增加的趨勢。以運算資源而言，TBTW 演算法對於 TBDMW 演算法有較快的運算速度，但以影像品質而言，TBDMW 演算法對於 TBTW 演算法有較佳的平均 PSNR 表現。因此若可以發展一套好的可靠性判斷方式判別整體的特徵點個數與運算時間，使得演算法能夠取得一個最佳影像品質也是門重要的課題。

結論

事實上，畫素對畫素的映射方法對於紋理複雜的區域有十分良好的功效，在映射的過程中卻耗費大量的運算資源。另一方面，對於 TBTW 演算法而言，對於紋理複雜的區域而言，三角形的映射結果可能使得合成影像產生紋理模糊的問題。因此我們改善 TBTW 的方法，只針對景深圖的資訊使用三角形映射，相較畫素對畫素的映射方式，可以有效的減少運算時間又可以解決 TBTW 演算法造成的合成品質低落的問題。

以運算資源而言，TBTW 演算法有較快的運算速度，但以影像品質而言，TBDMW 演算法有較佳的平均 PSNR 表現。因此若可以發展一套判斷準則判別畫面擷取的特徵點個數以及運算時間，使得演算法既能快速又能達到最佳的影像品質是門重要的研究課題。

第6章 結論與未來工作

6.1 結論

多重視點視訊已逐漸成為多媒體領域中的研究主流，而自由視點視訊是其中重要議題。故本論文以其為研究對象，將現有的影像擷取設備由一維攝影機陣列，利用空間上的相關性來提昇虛擬視點影像的合成結果。

針對MPEG所提供四種不同的MPEG測試影像：“Book arrival”、“Champagne tower”、“Newspaper”以及“Kendo”，觀察合成影像的平均PSNR結果。在合成的虛擬視點影像中，有時候會出現肉眼觀察測試影像的結果雖然並未較佳，但是PSNR的表現卻反而較好的情況，這有可能是因為使用疊合的合成影像場景較為單純。與傳統的視點合成演算法架構比較而言，三角形投影映射的演算法架構在PSNR上的普遍表現通常兩者相近。

另一方面，利用MPEG以及DERS 4.9產生的景深圖進行不同的視點合成演算法所表現的平均PSNR結果確實有著不同的表現。以整體而言，輸入的景深圖品質確實對於合成端產生的影像結果以及平均PSNR數值的確有相當程度上的影響。

事實上，攝影機的擺放距離也會影響到最後合成結果的平均PSNR表現，若攝影機的位置相距較遠，對於合成端所產生的影像結果以及平均PSNR數值確實均會降低，使得被某些物體遮蔽無法看見的區域變大，造成合成影像品質較差。或許可再針對三角形投影映射後的影像結果再做一些後續的濾波器處理，也許會得到更佳的影像品質。

隨著特徵點的個數增加，合成出來的影像品質會逐漸成現增加的趨勢。對於所有的MPEG測試影像而言，本篇論文5.3節所提出的TBDMW演算法相較於本篇論文4.2所提出的TBWT演算法約略成長了約略0.4至0.8dB不等。

以運算資源而言，傳統的畫素對畫素的映射方法對於紋理複雜的區域有十分

良好的功效，在映射的過程中卻耗費大量的運算資源。TBTW 演算法有較快的運算速度，但以影像品質而言，TBDMW 演算法有較佳的平均 PSNR 表現。因此若可以發展一套判斷準則判別畫面擷取的特徵點個數以及運算時間，使得演算法既能快速又能達到最佳的影像品質是門重要的研究課題。

6.2 未來工作

為了取得最佳的特徵點，在實驗中採用了測試影像的深度值與之間的梯度大小作為評判標準。但此方法雖然簡單易懂，但也導致仍有特徵點無法有效地形成最佳的三角形。在演算法技術成熟的今日，我們或許可以測試其他不同的評判擷取節點的方式，來找出一個兼顧影像品質和時間效能的演算法。

特徵點個數對於 PSNR 的整體表現有相關的現象。也就是說，三角形的切割數目越多對於 PSNR 上的表現越好。然而，三角形的切割數目若是過於精細則與 VSRS 2.0 的結果是差不多的。所以能夠有個好的評判準則判定整體畫面切割的三角形個數是非常重要的研究議題。

傳統的任意視點合成(FTV)的演算法架構通常都是針對空間域上的一致性提出改善的方法[8][9][16][17][28]，目前也有許多針對景深預估的演算法架構加強了時間域上的一致性，且實驗結果有明顯的改善效果。然而，現有的場景還原技術而言，並沒有太多演算法設計討論並改善時間域上的一致性。或許在視點合成演算法日漸成熟的今後，能夠更好的演算法可以提出改善影像品質的新架構。

参考文献

- [1] D. Scharstein and R. Szeliski, "A taxonomy and evaluation of dense two-frame stereo correspondence algorithms," *International Journal of Computer Vision*, vol. 47, pp. 7–42, Apr. 2002.
- [2] Z. Zhang, "A Flexible New Technique for Camera Calibration," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, Vol.22, pp.1330-1334, Nov., 2000.
- [3] T. Fujii, T. Kimoto, M. Tanimoto, "A New Flexible Acquisition System of Ray-space Data for Arbitrary Objects," *IEEE trans. on circuits and systems for video technology (CSVT)*, vol. 10, no.2, pp.218-224, 2000.
- [4] S. E. Chen and L. Williams, "View Interpolation for Image Synthesis," *ACM SIGGRAPH*, pp. 279-288, 1996.
- [5] M. Levoy and P. Hanrahan, "Light Field Rendering," *ACM SIGGRAPH*, pp. 31-42, 1996.
- [6] S. J. Gortler, R. Grzeszczuk, R. Szeliski, and M. F. Cohen, "The Lumigraph," *ACM SIGGRAPH*, pp. 43-54, 1996.
- [7] R. Szeliski, "Video Mosaics for Virtual Environments," *IEEE Computer Graphic Applic.*, vol. 16, pp. 22-30, Mar., 1996.
- [8] C. L. Zitnick, S. B. Kang, M. Uyttendaele, S. Winder, R. Szeliski, "High-quality Video View Interpolation using a Layered Representation," *ACM SIGGRAPH*, pp. 600-608, 2004.
- [9] Y. Mori, N. Fukushima, T. Yendo, T. Fujii, and M. Tanimoto, "View Generation with 3D Warping using Depth Information for FTV," *Signal Processing: Image Communication*, pp. 6572, 2009.
- [10] D. Min, D. Kim, S. Yun, K. Shon, "2D/3D freeview video generation for 3DTV system," *Signal Processing: Image Communication*, vol. 24, pp. 31-48, 2009.

- [11] W. Mark, L. Mcmillan, G. Bishop, "Post-rendering 3D Warping," *Proc. Symposium Interactive on 3D Graphics*, pp. 7-1, 1997.
- [12] C. Fehn, "3D-TV using depth-image-based rendering (DIBR)," in *Proc. Picture Coding Symp.*, pp.307-312, Dec. 2004.
- [13] M. Brown, and D. G. Lowe, "Automatic panoramic image stitching using invariant features," *International Journal of Computer Vision (IJCV)*, vol. 74(1), pp.59-73, 2007.
- [14] M. Tanimoto, "Overview of Free Viewpoint Television," *Signal Processing: Image Communication*, pp. 454-461, 2006.
- [15] T. Fujii, M. Tanimoto "Free-viewpoint Television Based on the Ray-Space Representation," *Proc. SPIE : 3-D TV, Video, and Display*, vol.4864, pp. 175-189, 2002.
- [16] 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(JVCI)*, vol.21, Issues 5-6, pp.542-560, Sep., 2010.
- [17] N. Fukushima, T. Yendo, T. Fujii, M. Tanimoto, "Real-time Arbitrary View Interpolation and Rendering System using Ray-Space," *Proc. SPIE Three-Dimensional TV. Video. and Display IV*, vol. 6016, pp. 250-261, 2005.
- [18] M. Tanimoto, "Overview of Free Viewpoint Television," *Signal Processing: Image Communication*, pp. 454-461, 2006.
- [19] M. Tonimoto, "Reference Software for Depth Estimation and View Synthesis," ISO/IEC/WG11/MPEG2008, M15377 , April, 2008.
- [20] M. Tonimoto, "1D Parallel Test Sequences for MPEG-FTV," ISO/IEC/WG11/MPEG2008, M15378, April, 2008.
- [21] M. Tonimoto, "Experimental Framework for 3D Video Coding,"

ISO/IEC/WG11/MPEG2010, W11477, July, 2010.

- [22] I. Feldmann et al., "HHI Test Material for 3D Video," ISO/IEC/WG11/MPEG2010, M15413, July, 2008.
- [23] O. Stankiewicz et al., "A soft-segmentation matching in Depth Estimation Reference Software (DERS) 5.0," ISO/IEC JTC1/SC 29/WG11, M17049, October, 2009.
- [24] R. Hartley, A. Zisserman, Multiple View Geometry in Computer Vision, 2nd ed., Cambridge University Press, Mar., 2004.
- [25] H. H. Baker, R. C. Bolles, "Epipolar-plane Image Analysis: A Technique for Analysis Motion Sequences," *IEEE 3rd workshop Computer Vision: Representation and Control*, pp. 168-178, Oct. 1985.
- [26] J. H. Park and H. W. Park, "Fast View Interpolation of Stereo Images using Image Gradient and Disparity Triangulation," *Signal Processing : Image Communication*, vol. 18, pp. 401-416, 2003.
- [27] J. H. Park, H. W. Park, "A Mesh-Based Disparity Representation Method for View Interpolation and Stereo Image Compression," *IEEE Trans. on Image Processing*, vol. 15, no.7, July, 2006.
- [28] B. Choi, T. Kim, K.-J. Oh, Y.-S. Ho, J.-S. Choi, "Intermediate View Synthesis Algorithm using Mesh Clustering for Rectangular Multiview Camera System," *Proc. SPIE*, vol.49, Feb., 2010.
- [29] D. Farin, R. Peerlings, and H. N. Peter, "Depth-image Representation Employing Meshes for Intermediate-view Rendering and Coding," *IEEE 3DTV Conference 2007*, pp.1-4, May, 2007.
- [30] A. Criminisi, P. Perez, K. Toyama, "Object Removal by Exemplar-based Inpainting," *IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR)*, vol.2, pp.721-728, June, 2003.

自傳

李兆軒，1987 年 2 月 10 日出生於南投市，台中市人。2009 年 6 月畢業於國立中興大學電機工程學系，之後於 2009 年 8 月攻讀國立交通大學電子研究所碩士學位，承蒙杭學鳴教授的指導，進入通訊電子與訊號處理實驗室(CommLab)，主要研究主題為利用攝影機陣列還原任意視點合成演算法設計。論文題目為「基於三角轉換映射之任意視點合成演算法」。

