

# 行政院國家科學委員會專題研究計畫成果報告

## 虛擬實境中多層次精細度模型技術之研究( )

### Level-of-Detail Modeling Technique for Virtual Reality

計畫編號：NSC 90-2213-E-009-125

執行期限：90年08月01日至91年07月31日

主持人：莊榮宏

共同主持人：

計畫參與人員：

#### 一、摘要

觀視點相依多層次精細度技術希望對模型局部區域即時地改變精細程度，讓顯像所處理的多邊形集中在可見範圍，並依觀視點改變而迅速更新能力，兼具保留視覺效果和互動需求。本論文在幾何特徵保留、拓樸架構調整以及質保留的頂點叢集簡化演算法的根基上，研究出模型局部精細程度隨觀視點移動而能快速調變之演法。

**關鍵詞：**觀視點相依、多重精細度模型、選擇性精細化

#### Abstract

View-dependent Level of detail(LOD) modeling has its strength in selective refinement and effective view and backface culling. In this paper, we propose a view-dependent LOD modeling in which the selective refinement is performed according to a dependency graph. The dependency graph is constructed based on a progressive meshing scheme that is clustering-based and takes both geometry and topology simplification into account.

**Keywords:** view-dependent , LOD modeling, selective refinement

#### 二、緣由與目的

在三維計算機圖學(3D-Computer Graphics)中為了避免物體模型的多邊形網

格在距離遠時投影成畫面上同一個像素(pixel)，因此用較少的多邊形來成像來增加計算速度，故使物體以多層次精細度模型來呈現影像，本文以視觀點不相依LOD[Yan99]的研究成果為基礎，在兼顧幾何特徵保留、拓樸架構調整以及材質保留功能的模型簡化演算法根基上，將其推展到觀視點相依上之應用，並達到

- (一) 在大量多邊形之下維持快速顯像。
- (二) 合理時間內產生合理空間負擔之觀視點相依多精細度結構。
- (三) 簡單迅速之精細度選擇判斷與局部精細與局部粗糙之運算。
- (四) 充份運用動態瀏覽時的共效性，包括時間共效性(temporal coherence)與空間共效性(spatial coherence)。

我們認為，觀視點相依多精細度技術付出的額外代價與瓶頸，最大關鍵問題在於其局部幾何簡化步驟中彼此間的相依關係，我們由此切入重新設計適當的觀視點相依多精細度模型資料結構。

#### 三、相關研究

我們擷取觀視點相依多重精細度模型的最具代表性的幾篇文獻，針對動態運作結構、相依性問題，選擇性精細化的問題一一列出代表的方法。

##### (一) 典型結構：頂點階層樹

目前簡化局部幾何以產生多精細度的方法，大都是持續刪不影響外觀的幾何元素，以「代表頂點(representative vertex)」代替，「一個代表頂點等同一次簡化過程」為基礎，[XESV97,Hop97,LE97,ESV99]都用頂點階層樹(vertex tree)的概念，而簡化

的方法又分為邊線折疊法 (Edge Collapsing)[Hop96,Hop97]和頂點叢集法 (Vertex Clustering)[RB93,LE97]。

[Hop97]將頂點分裂前後頂點當做二元樹的父子節點建構頂點階層樹，可視為由上往下(top-down)的方向產生觀視點相依之二元樹結構，此方法最大的缺限是簡化過程太多，記憶體負擔大。

[LE97]採用頂點叢集法建立頂點階層樹，但並沒有考慮相依性問題。

## (二) 局部精細化的相依問題

在簡化過程中若任意的刪增移動頂點會造成線和面的幾何資訊無法正確地組合，這問題稱之為相依性(dependencies)。若不考慮相依性問題則會造成幾何外形破壞的例子，例如[XESV97]中的反摺(foldover)的例子，相依性問題和「模型上的相鄰幾何元素」與「頂點階層樹的建構次序」有密切關係。

[XESV97]和[Hop97]利用檢查模型上的相鄰幾何元素來解決相依性問題，[XESV97]在建構階層頂點樹時，在簡化過程紀錄縮減邊的週圍頂點資訊，用來判斷是否允許邊線縮減與展開，[Hop97]是紀錄邊線兩側三角形週圍四個三角形的編號，當動態瀏覽時邊線無論縮減或展開都必須在這些紀錄下來的三角形都存在時才可執行。

[ESV99,GTH99,GTLH98]是用維護簡化先後順序來解決相依性問題，[GTH99,GTLH98]運用頂點與三角形層次數的關係，找出簡化步驟的偏序關係，建立一有向無環圖(Directed Acyclic Graph)來紀錄邊線縮減展開的相依關係。[ESV99]則紀錄週遭頂點編號的最大最小值，就可判定合法簡化與否。

## (三) 選擇性精細化的判斷

[LE97,XES97,Hop97]所採用的判斷方法歸納為觀視範圍(view frustum)測試、背向觀視點(back face culling)測試、螢幕投影量誤差、區域照明、三角形數目預算。

## 四、觀視點相依多精細度模型參考結構

本論文注重在解決影響選擇性精細化處理時間的因素，包括選擇性精細化的基礎運作與結構、維持正確模型的額外相依檢查、視點相依之精細或粗糙之選擇依據、動態即時改變局部精細度。本文採用[Yan99]的頂點叢集法為模型簡化單位運算，採用「維護簡化順序」來解決相依性問題，區分簡化步驟運算彼此之間的關係為「替代」與「依賴」關係，並採用以簡化步驟運算當做階層結構的結點，以相依關係為邊線建構相依結構圖(dependency graph)。

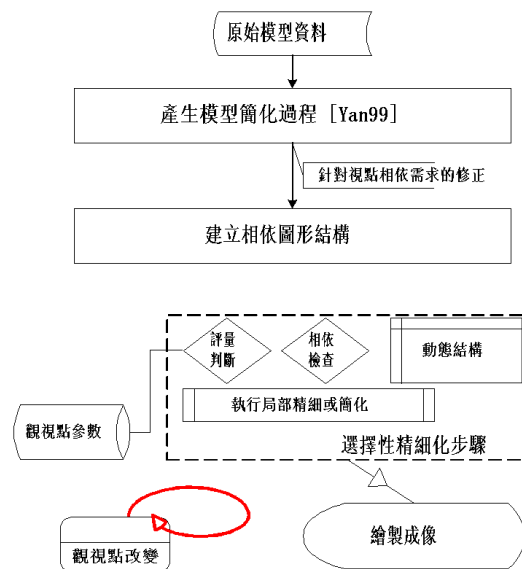


圖 4.1:大要流程圖

本論文方法大要流程圖如圖 4.1 所示。

## (一) 簡化步驟間的相依問題之分析

任兩個頂點叢集簡化步驟  $i, j$  的影響範圍可分為三種，如圖 4.2 所示。圖 4.2(a) 表示  $i, j$  兩不相互影響，圖 4.2(c) 表示  $i$  在空間中完全被  $j$  占領，我們將圖 4.2(c)  $i, j$  兩步驟間情況定義為「替代關係」。圖 4.2(b)  $i, j$  部分重疊且  $i$  的代表點必落在  $j$  的邊界頂點上，定義  $i, j$  兩步驟為「依賴關係」。

對作簡化步驟  $i$ ，可以列出所有的關係於表 4.1，其中被替代和精細依賴為替代和簡化依賴的反方向的關係。

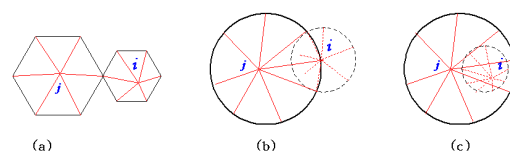


圖 4.2:兩個頂點叢集簡化步驟  $i, j$  影響範圍示意圖

名稱	簡易符號	先後次序	空間性質
替代	$p$	$p < i$	$i$ 的空間完全被 $p$ 覆蓋
被替代	$R_1, R_2, R_3, \dots$	$R_n > i$	$i$ 在空間上和 $R_n$ 重疊
精細依賴	$O_1, O_2, O_3, \dots$	$O_n < i$	$O_n$ 簡化範圍的邊界頂點中有 $i$ 的代表點，空間上相鄰
簡化依賴	$C_1, C_2, C_3, \dots$	$C_n > i$	$i$ 簡化時， $C_n$ 的代表點為簡化範圍的邊界頂點，空間上相鄰

表 4.1:簡化步驟彼此間可能之相依關係

可正確地執行精細化的條件為：

- (1) 投影誤差超過容忍值，必須精細化
- (2) 替代關係  $p$  已經精細化
- (3) 所有精細依賴關係步驟  $O_1, \dots, O_n$  都已精細化

可粗糙化的條件為：

- (1) 投影誤差低過容忍值，可以簡化模型
- (2) 所有被替代關係  $R_1, \dots, R_n$  已經簡化
- (3) 所有簡化依賴關係步驟  $C_1, \dots, C_n$  都已經簡化

## (二) 相依圖的結構

相依圖如圖 4.3 所示，以最粗糙模型 (Base Model) 為中心，由內向外散佈代表基本簡化步驟的節點。當動態瀏覽時，正確的精細化和簡化步驟必須去檢查是否相關連的步驟是否已經早一步完成了。實際上不用一一的去檢查，以圖 4.3 為例考慮三個精細化步驟  $e < f < h$  (由最粗糙模型算起)，當  $h$  要精細化時只需檢查  $f$  就可決定是否可精細化，因  $f$  依賴  $e$ ， $f$  可精細化則  $e$  必已經精細化過了，實驗結果顯示約可消去 1/2 左右的依賴關係數目。

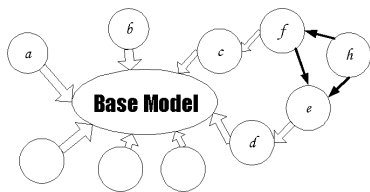


圖 4.3:相依圖形結構，虛箭箭頭為精細依賴關係

建立相依圖的過程分為兩步驟，第一步是讀進漸進式模型，並針對每個精細化步驟建立節點資訊，取得精細化步驟彼此間的相依關係，第二步則篩選依賴關係。

## 五、動態運作機制與誤差量測

當動態巡視時，要加入觀視點參數的考量，圖 5.1 說明了選擇精細化步驟方塊之觀視點參數輸入與繪製成畫輸出之流程。

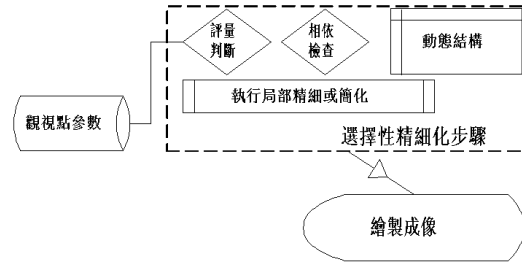


圖 5.1:動態運作機制的流程示意圖

### (一) 動態運作機制與選擇性精細化處理

在相依結構圖上表示模型在不同觀視點下的局部精細程度，用一個旗標(status flag)來表示目前這個節點所代表的精細化步驟是處於「粗糙狀態」或「精細狀態」，對於每個觀視點，相依圖上節點都有一組對應的狀態，如圖 5.2 所示，當觀視點改變時，要將「精細狀態」節點(圖中的 R)之集合『移動』。

要改變相依圖形結構上節點狀態要經過兩道測試手續，一是幾何誤差，另一相依問題測試，由表 5.1 所示四種交叉測試的情況。

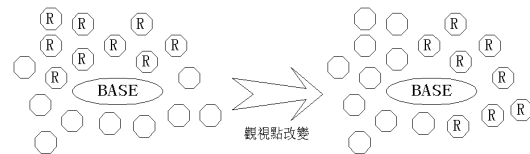


圖 5.2:相依結構圖上因應觀視點改變之狀態變化

		幾何誤差檢查	
		低於容忍值	超過容忍值
相依檢查	不合格	等待?	強迫精細化
	合格	化簡	精細化

表 5.1:幾何誤差評量 v.s.相依關係檢查

由表得知，決定的關鍵在於幾何誤差評量測試上，為了效率，在「精細狀態」和「粗糙狀態」，再加入兩個緩衝狀態「簡化預備狀態」和「精細化預備狀態」。「簡

化預備狀態」為該節點幾何處於精細但通過相依性檢查，「精細化預備狀態」為該節點幾何處於粗糙但通過相依性檢查，如圖 5.3 所示。若節點  $i$  處於精細化預備狀態，經過檢查決定要精細化，除了改變本身的狀態，執行幾何上的改變，也將相關節點相依性檢查並修正其狀態。



圖 5.3: 結點狀態變化的有限狀態機示意圖

本論文動態運作機制的資料結構為兩串列，分別是 Refine-Candidates 放「精細化預備」狀態節點列表，以及 Coarsen-Candidates 放「簡化預備」狀態的節點列表，為避免對同一節做重覆幾何誤差評判，用陣列儲存測試的結果，以四種旗標值來表示。

主要的精細化處理步驟流程即依序處理 Refine-Candidates、Coarsen-Candidates。處理 Refine-Candidates，流程為進行幾何誤差測試、依結果決定是否要精細化，若要精細化則執行改變狀態與模型幾何改變，對子孫節點進行幾何誤差測試，若判定要精細化則將其加入 Refine-Candidates，並執行強制排除相依問題的程式，確保子孫相依性沒有問題。若子孫節點判定為不必精細化，則再判斷相依性測試看是否要加入 Refine-Candidates 串列中。若節點為不必精細化，則一樣對子孫節點進行幾何測試，若子孫判斷要精細化同樣加入 Refine-Candidates 串列中，並執行強制排除相依問題程序。

在處理簡化步驟處理 Coarsen-Candidates 的流程大致和精細化相同，當幾何誤差低於容忍值，將之簡化並檢查替代關係與精細依賴關係節點是否滿足相依條件，若是則放入 Coarsen-Candidates 串列中，並對其相關節點查驗並校正狀態。

## (二) 選擇性精細化評量與決策

選擇性精細化評量過程就是對指定的精細化步驟決定在目前觀點下是否應該進行精細或簡改變。本論文選定「是否在觀

視範圍內」、「是否背向觀視方向」、「是否投在成像平面上誤差小於給定的值」三個測試條件，前兩項目的是經觀視點與精細化在空間上的位置方向來判斷該節需不需進一步精細化，第三項測試由螢幕上成像與近似模型之成像間誤差值來決定精細或粗糙。

## 六、結果與討論

本論文以 C 語言撰寫主體演算法，採用 OpenGL 函式庫和 GLUT 作簡易 GUI 介面。

### (一) 選擇性精細化效果

圖 6.1 為實驗結果，對兔子(bunny)模型在 2R 與 4R 距離的觀點比較，以及正面和側面的比較，可圖中(a)(c)發現觀視點方面的物體網格是比較密的，而圖(b)(d)距離變遠而在畫面佔據的大小變小，模型整體的精細度也隨著下降。

### (二) 動態運作機制效率測試

利用典型、依據邊緣折疊所建構的三元頂點階層樹演算法[Lia99]當對照組，表 6.1 為實驗組與對照組的差異。

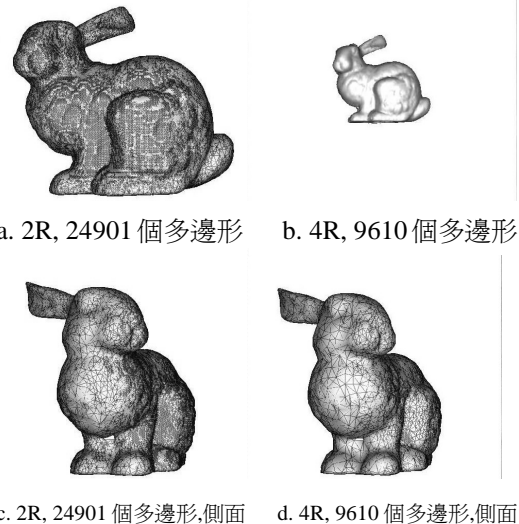


圖 6.1: 選擇性精細化的效果

將實驗組分為甲乙兩組，實驗組甲的叢集範圍比實驗組乙來的大。實驗的結果發現實驗組甲乙的執行時間比對照組平均來的快，平均處理節點數目也比較少，但實驗組在每張畫面上的多邊形數目卻比對

照組還多。

項目	實驗組	對照組
模型簡化法	頂點叢集法[Yan99]	邊緣折疊，代表點取兩端其一
視點相依參考結構	相依圖形結構	二元頂點階層樹，top-down 建構
相依關係處理	定範圍，分類與精簡	頂點及三角形狀態數，動態時重新組織三角形
幾何誤差評判	視界、背向、二維投影誤差	視界、背向、一維投影誤差

表 6.1:演算法差異處列表

### (三) 結論與未來發展方向

本論文承襲一套視點不相依的模型簡化演算法[Yan99]，將簡化過程中因先後次序與幾何相鄰而相互影響的相依關係找出來，並建立相依圖形結構。在動態運作時可以很有效率地減少需要檢測的節點個數，增快選擇性精細化的處理時間。但受限於精細步驟幅度較大的關係，在同樣螢幕誤差容忍值下，三角形數目會比較多，在中遠距離時整體可能不如邊緣折疊法，但在近距離時卻可提供邊緣折疊辦不到的穩定效能。

未來的發展方向以降低三角形數目和消滅突波為方向。實驗結果顯示保留相依性的相依圖方法，三角形數目會高居不下，究竟怎麼樣大的叢集才是最佳的呢？或從鬆綁相依性問題研究以期減少不必要的三角形。突波現象為三角形個數、處理節點數目與選擇性精細化時間突然急劇上升，在實驗中的數據顯示的特殊情況，期望以中斷的選擇性精細化處理或對螢幕誤差容忍值係數動態的調節來消滅這種現象。在加速顯像方向，可以用 Triangle Strip 的技巧來加速顯像過程。

## 七、參考文獻

- [GTLH98] André Guézic , Gabriel Taubin , Francis Lazarus , William Horn. Simplicial Maps for Progressive Transmission of Polygonal Surfaces. In *VRML*, 1998.
- [Hop96] Hugues Hoppe. Progressive Mesh. In *SIGGRAPH*, 1996.
- [Hop97] Hugues Hoppe. View-Dependent Refinement of Progressive Meshes. In *SIGGRAPH*, 1997.
- [LE97] David Luebke and Carl Erikson. View-Dependent Simplification Of Arbitrary Polygonal Environments. In *SIGGRAPH*, 1997.
- [Lia99] EnChyi Liang. View-Dependent Level-of-Detail Modeling for Walkthrough Application, 1999.
- [RB93] J.Rossignac and P.Borrel. Multi-Resolution 3D Approximations for Rendering Complex Scenes. In *Computer Graphics*, 1993.
- [XESV97] Julie C.Xia, J.El-Sena, and A.Varshney. Adaptive real-time level-of-detail-based rendering for polygonal meshes. In *IEEE Transactions on Visualization and Computer Graphics*, 1997.
- [Yan99] Shu-Kai Yang. Material-Preserving Progressive Mesh, 1999.
- [ESV99] J.El-Sana and A.Varshney. Generalized View-Dependent Simplification. In *EROGRAFICS*, 1999.
- [GLH99] André Guézic , Gabriel Taubin , William Horn. A Framework for Streaming Geometry in VRML. In *IEEE Computer Graphics and Applications*, 1999.

