TR－91－002

使用多面體集合運算於立體物之合成

論文名稱：使用多面體集合運算於立體物之合成　頁數：72

校(院)所組別：　淡　江　　大學(學院)資訊工程研究所　　組

畢業時間及提要別：七十八學年度第二學期碩士學位論文提要。

研究生：朱　首　龍　　指導教授：胡威　博士

論文提要內容：

立體幾何模型(3D Geometric Modeling)在電腦圖學的領域中，是一個重要的部分，目前立體幾何模型的主要表示方法有CSG (Constructive Solid Geometry)和B-rep(Boundary Representation)，在CSG，物體是以一個Tree 來表示，Tree的leaves爲一基本物體(Primitives)，而其Internal nodes爲Regularized Boolean Operation，物體是由leaves的基本物體經由Internal nodes的集合運算而產生；在B-rep，一個物體的表現是由兩種資料組成：一爲拓樸結構，一爲vertices、edges、faces的幾何結構，此兩種模式各有其優缺點，以資料輸入量而言CSG比B-rep小，但以電腦圖學的應用來看，以B-rep之表示方法較方便。

在過去有一些演算法，提供多邊形物體或基本物體的集合運算，以產生更複雜的物體，本論文的目的是要提供類似的演算法，使之不但具有集合運算(交集、聯集、差集)的功能，並且在多邊形物體的定義中加入更多的資訊，以便在集合運算後所產生的物體能有更多的資訊提供給後續的處理，如陰影、上色等。

由於CSG和B-rep此兩種模式各有優缺點，因此在基本物體的產生上，是以CSG的方式來產生，亦即輸入物體的幾何特性，如球心、半徑等，而在內部的處理上，是以B-rep的方式來處理，希望能結合此兩種模式的特性，使得在使用者介面設計上能更具有對使用者友善的特性。

本系統是架構於VAX-750和E&S PS300繪圖工作站上，應用PS300的向量處理和眞實影像表現的功能，由於基本物體都是一些簡單的物體，所以在基本物體的產生上，加入超二次的數學方法和SWEEP架構法，以期能產生基本上較複雜的基本物體，由於超二次的物體能以調整參數的方式來產生相似而有變形的物體，另外SWEEP架構法亦是在立體幾何模型中一個產生物體的重要方式，所以將使得本系統更具交談性與彈性。

Title of Thesis: Using Set Operation of Polyhedra For 3D Object Synthesis

Total Pages:72

Name of Institue: Graduate Institue of Information Engineering, Tamkang University

Graduate Date: 6/9/90

Degree Conferred: Master

Name of Student: Shoou-Long Chu

Advisor: Shaing Hu

朱首龍

Abstract:

3D Geometric Modeling is an important aspect in computer graphics, CSG(Constructive Solid Geometry) and B-rep (Boundary representation) are currently the two primary representation techniques in 3D Geometric Modeling. In CSG, objects are represented by a CSG-Tree, the leaves are primitives. The internal nodes of CSG are Regularized Boolean Operations. Object is created by the primitives through the set operations of the internal nodes. In B-rep, an object is represented by means of the combination of two kinds of data structure : topology structure and geometric structure such as vertices, edges and faces. Both types of modeling have their advantages and disadvantages. While CSG involves less input data and B-rep is more convenient in computer graphic applications.

In the past,there are many algorithms providing the set operations of polyhedra and primitives to construct more complex objects. The purpose of this thesis is to present a similar method, not only has the capability of the set operations, but also provides more information in the definitions of polyhedra objects,which in turn provides more information in objects after the set operations for rendering, such as shading, texture.

Because of the CSG and B-rep have advantages and disadvantages,so,our method uses CSG to produce the primitives,it means to input geometrical characteristics such as sphere center and radius, and B-rep to perform the internal operations. By combining the respective advantages of both types of modeling, it is hoped that a friendly user interface may be produced.

This system is designed on VAX/750 and E&S PS300 graphic workstation, and applies the vector processing and realism image representation capabilities of the PS300. Because the primitives are all simple objects,we add the superquadric method and the sweep construction in the primitives generation to creat primitives which are fundamentally more complex.As the parameters of superquadric method may be adjusted to creat similar objects with different shapes, and as the sweep construction is an important technique for producing objects in 3D Geometric Modeling, so this system is more interactive and flexible.

資訊工程 研究所 朱 首 龍 君所提之論文

使用多面體集合運算於立體物之合成 (題目)，

經本委員會審議，認爲符合碩士資格標準。

論文口試委員會 主任委員 吳憲明 (簽章)

委 員 [illegible]

鄭國揚

中 華 民 國 79 年 5 月 30 日

誌　謝

感謝指導教授鄭國揚博士、胡戚博士在論文撰寫期間，悉心的指導並指出所犯的錯誤，使得論文能順利完成。口試時，吳憲明教授提供的寶貴意見，使我受益匪淺。

同學嚴漢偉、楊玫眞、許丕忠、顏文龍等，在課業上相互砥勵，增廣見聞。研究所就讀期間，家人及小羽的鼓勵與支持，使我專心於學業，無後顧之憂。

特別感謝，在論文撰寫期間，中央研究院資訊科學研究所提供的優良的研究環境，使得論文能順利完成。

最後，謹將本論文獻給我敬愛的父母。

目 錄

第一章 緒論

1.1 簡介

近年來，由於微電子和彩色顯像設備技術的進步，使得這些需求已接近實用的階段，這也使得電腦繪圖方面更加快速地發展；電腦圖學的應用極爲廣泛，例如：廣告業、建築設計、電腦輔助設計(CAD)、電腦輔助教學 (CAI)、國防工業、娛樂業、醫學、模擬以及科學運算視覺化(Scientific Visualization)等等，使得電腦繪圖的地位更加重要。

電腦圖學包括下面四個領域：

1).立體幾何模型 (3D-Geometric Modeling)

2).影像合成 (Image Synthesis)

3).使用者介面管理系統 (User Interface Management System)

4).電腦動畫 (Computer Animation)

(一) 立體幾何模型

立體幾何模型在 CAD/CAM 和電腦繪圖中是一個重要的領域，幾何模型最早是發展於 CAD/CAM 系統， Ivan Sutherland(1963) 是這方面的發展先鋒，直到現在，更

有許多的表現方式(Representation)被提出來。

目前立體幾何模型的表示方法包括有：CSG(Constructive Solid Geometry)、B-rep(Boundary Representation)、Sweep Representation、空間計算 (Spatial Occupancy enumeration)、基本物體表現(Primitive instances)和元件分解(Cell decomposition) 等等。

(二) 影像合成

發展影像合成技術的目的是希望能產生和真實物體照片一樣逼真的合成影像 (Synthetic Image)，基本上，影像合成方面的研究是希望能用電腦，模擬真實物體和光源間的光學現象(Optical Process)，但是光線傳播 (Light Propagation)和交互作用的物理現象非常的複雜，所以通常我們利用簡化的物理模式在電腦彩色顯示器上顯現諸如反射、折射、透明、影深和背影光等光學現象。

目前主要的影像合成方法包括 Scan-Line Algorithm、Phong Shading、Cook-Torrance Shading、Ray-Tracing 和 Radiosity 等，其中以 Ray-Tracing 和 Radiosity 較能產生近似逼真的影像。

(三) 使用者介面管理系統

自從 70 年代底 Smalltalk-80 系統開始，電腦系統的使用者介面(User Interface)愈益強調圖形(Graphics)表示，高度的互動(Intersection)以及直接操縱 (Direct Manipulation)，使用者介面管理因此愈趨複雜， 而有使用者介面管理系統之誕生；使用者介面是任何電腦系統都不可少的部份，其重要性自不待言，近年來使用者管理系統已經成爲電腦圖學研究之重要一支。

(四) 電腦動畫

基本上，電腦動畫是利用電腦來產生一連串的畫面，並將這些畫面以一定的速度連續顯示，藉由人類視覺暫留的特質而產生動感，所以在這方面電腦扮演的角色有：

(1) 提供製圖 (特別指 Key Frames 的圖形)

(2) 產生 Key Frames 之間的變化圖形

(3) 動作轉換之控制

(4) 產生立體表面之顏色

(5) 模擬相機之動作

(6) 產生同步音效

立體圖形的製作通常是採用可調整的曲面表示法；產生中間變化的圖形亦是採用可調整的曲面表示法來釐定變化軌跡；動作轉換之控制則是藉由幾何轉換及時間曲線來

完成，而圖形表面顏色是經由隱面、塗影和光線之計算處理得之；若欲獲得自然景觀的立體圖形尚需經過特別之數學處理才可以；由此可知，電腦動畫之產生是藉由數學及物理公式之運算來模擬畫家心目中想要的圖形及動感效果，除此之外，動畫系統尚需提供一套圖形編輯器，讓使用者能夠修改圖形或動感效果，這樣又涉及了使用者圖形介面之設計；從以上的說明可想像出藉由電腦產生立體動畫之系統設計是一件非常繁複重要的工作。

1.2 研究動機

在設計一個立體物體(3D Object)時，我們需要一些工具，使得在設計時，能夠快速並簡單地完成，這就需要硬體速度、繪圖處理能力與使用者介面(User Interface)的良好配合；舊的模型系統是使用單一的表示方法，而最常使用的是 CSG 與 B-rep [ReV,'82], [ReV,'83]，但是如[Req,'80] 所說，並沒有一個現有的表示方法的特性能夠完全比其它的表示方法更好，也就是說，每一種表示方法都有它的優點，所以，也有結合兩種以上表示方法的系統出現，亦即混合型系統 (Hybrid system)，最有名的就是 GMSolid System [Boyse, Gilchrist,'82]；這最主要

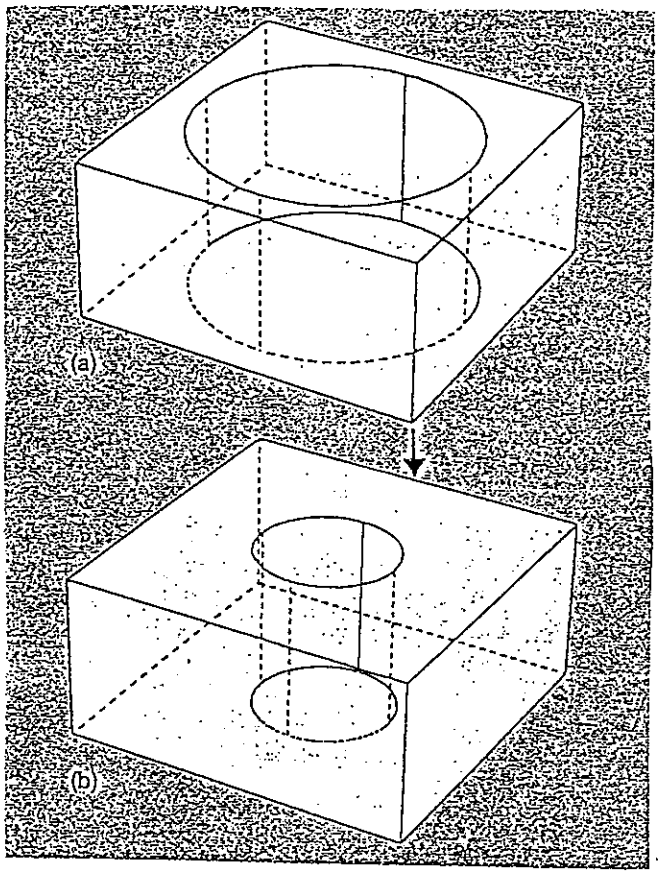


圖 1　物體的修改

是基於系統內部處理以及使用者介面的考慮因素；CSG 最主要的優點是需要的記憶體較小，而且對於如圖 1 的修改過程較爲容易，這是因爲 CSG 儲存了物體全部的定義過程，所以只要修改物體的部份定義就可以達到修改的目的，但是對於一些非集合運算 (Nonset operation)， 如

Local Operation，卻較不容易使用於 CSG 的系統；B-rep 最大的優點便是可以廣泛地使用運算(operation)，但是因爲 B-rep 只有表示物體的形狀，所以，對於如圖 1 的修改過程便不容易做到，要完成這項工作，使用者必須先消除圖 1(a) 中的洞，再削去圖 1(b) 中較小的洞，這是一個很麻煩的程序。

在 CSG，是以一個稱爲 CSG-Tree 的樹狀結構來表示物體，CSG-Tree 的 Leaf Nodes 爲一基本物體(Primitives)，此基本物體通常是一些簡單的幾何物體，如：球體(Sphere)、圓柱體(Cylinder)、圓錐體(Cone)、橢球體 (Ellipsoids)、圓環體 (Torus) 等等，而其 Internal Nodes 爲 Regularized Boolean Operation(交集，Intersection、聯集，Union、差集，Difference)或 Motion (如：旋轉、平移等)；物體的產生是由 Leaf Nodes 的基本物體經由 Internal Nodes 的集合運算而產生；在 B-rep，一個立體物體的表現是由兩種資料組成：一個爲拓樸結構，一個爲 Vertices、Edges、Faces 所組成的幾何結構。

此兩種模式各有其優缺點，以資料輸入量而言，CSG 比 B-rep 小，但以電腦圖學的應用來看，以 B-rep 之表

示法較方便。

在中央研究院資訊工程研究所的設備中，包括有 VAX /750 、E&S PS300 繪圖工作站及多部 SUN WorkStations，在作業系統方面是使用 Ultrix， 亦即 UNIX 系統，因此，在發展上是架構在 UNIX 上，並且使用 C Language 處理， 另外還有一套 USNA 所發展的套裝軟體 CADIG/ ANIMATOR，能夠提供陰影 (Shading)、上色 (Texturing) 的功能，再加上所提供的基本繪圖功能，使得研究工作在基本上已具備一種基礎。

本論文所討論的主要是使用於立體物體合成的集合運算，希望能提供交集、聯集、差集的功能，並且在設計上加入對使用者友善(User Friendly)的考慮，以期在後續的發展上，能夠較容易完成此項要求；在整個系統的架構上，我們希望能盡量結合 CSG 和 B-rep 的優點，並且彼此能彌補對方的缺點，因此，在基本物體的產生上，是以 CSG 的輸入方式所產生(即輸入基本物體的幾何特性，如球心、半徑等)，而在內部的處理上，則是以 B-rep 的方式來處理。

另外，由於一般 CSG 表現方法所處理的基本物體都是一些簡單的幾何物體，如球、長方體、圓柱體、圓錐體

、圓環體、橢球等，在某些應用上稍顯不足，這將增加使用者在設計物體時的困擾與不便，降低對使用者友善的程度，因此，在基本物體產生上，我們加入了超二次的數學方法(Superquadrics method) [Bar,'82] 和 Sweep 架構法；由於超二次的數學方法能夠以調整參數(如 S1、S2)的方式來產生相似而有變形的物體，而 Sweep 架構法亦是在物體模型中一個產生物體的重要方式，所以將使得本系統的基本物體更具多樣性，也使得本系統更具交談性與彈性。

本論文共分爲五章，除了第一章緒論之外，第二章將說明本系統的系統模型，第三章將說明本系統所採用的數學方法、理論基礎及演算法，第四章將說明實驗結果與討論，第五章將說明結論與未來的研究方向。

第二章　系統模型

2.1　系統背景

在複雜影像合成的一個重要的部份，就是資料的產生與描述，以組合成一個物體並且表現出來，有許多早期所發展的技術便是用於處理這些工作，這些技術可以參考計算幾何(Computational geometry)[FAR,'74] 或是電腦輔助幾何設計(Computer-Aided Geometric Design)；最普通的資料產生技術通常需要交談性地輸入特定基本物體結構，並且需要後續的修改功能，以便能產生一個電腦能表現的物體，這些基本物體結構可以是點、線、多邊形或是參數表面(Parametric surfaces)。

有許多複雜的三維物體都是在這種型態的環境下所創造出來，這些三維物體的描述是藉由較簡單的幾何形狀或基本物體組合而成，也就是基本物體經由特定的運算（如：交集、聯集、差集），產生出原先所定義的物體；許多系統已經發展出來 [BrC,'82],[PaR,'77]，藉由面的交集以定義一個物體，並組合成多邊形物體；Levin [LeJ,'76] 提供了使用 Quadric surfaces 的組合技術；Carlson [Car,'82] 則提供了使用 Bicubic surfaces 的組合技術

來組合物體。

2.2 模型的基礎 (Modeling Primitives)

一般來說，物體的表示可以區分爲三個方式：

1. 線條表示方式(Wire-frame representation)
2. 表面表示方式(Surface representation)
3. 體積表示方式(Volume representation)

2.2.1 線條表示方式 (Wire-frame representation)

在此方式中，物體僅是由線段的集合來表示，即是由點(vertices) 和邊(edges)表示，並沒有關於表面和體積的資料，此種表示方式通常太過簡單，而無法用於產生高度的真實物體。

線條表示法是歷史最久也是最簡單的三維立體模型，它的組合方式是由一串點的座標或是如下面所列的指令所組成：

Moveabs A --- 移到一個新的位置

Lineabs B --- 由現在的位置至新的位置畫一條直線

其中 A 和 B 是三個實數 <X, Y, Z> 集合所定義的向量，例如：以圖 2 中的圖形可以表示爲：

Moveabs <<3, 0, 1>>

Lineabs <<5, 0, 1>>, <<6, 0, 5>>, <<3, 0, 1>>

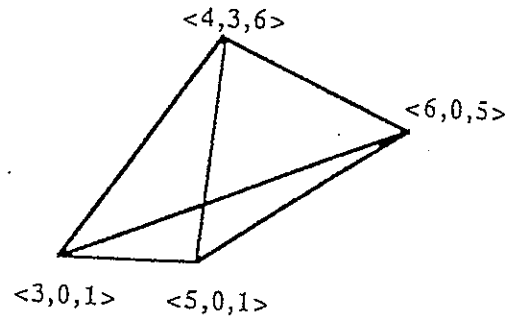


圖 2　四面體

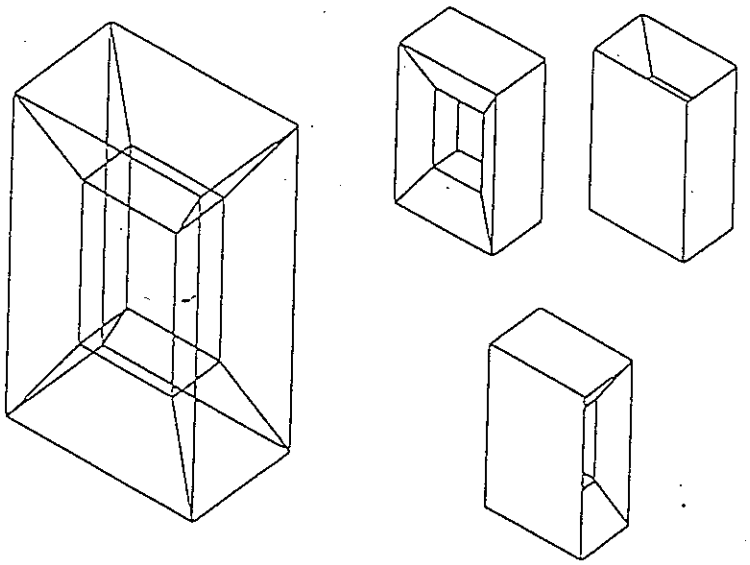


圖 3　混淆的立體物體

，<<4，3，6>>，<<6，0，5>>

Moveabs <<5，0，1>>

Lineabs <<4，3，6>>

一般來說，線條表示方式很容易造成混淆的立體物體，如圖 3 所示，這是最大的缺點。

2.2.2 表面表示方式 (Surface representation)

物體的表示是以基本表面的集合來組成，一般說來，依物體的形狀來分，可分成三種類型來描述：

---由多邊形的集合來描述

---由代數表面(Algebraic surface)的公式來描述

---由表面塊(Patch)來描述

大部份的物體都有曲線的特性(Curvature properties)，我們可以用平面的多面體來趨近；另外，物體也可以用非平面的表面(Nonplaner surface)來表示，而爲了顯示於螢幕時，可以自動轉換成以多邊型爲基礎的方式；當這種表面作爲內部的表示方式時，它們扮演一個非常重要的角色；我們可以區分爲三種型態的表面：

1. 規則性表面(Ruled Surface)：

   如球體(sphere)、圓錐體(cone)、圓柱體(cylinder)、旋轉表面(revolution surface)等，這些

物體可以使用 Sweeping 的技術架構出來。

2. 參數表面(Parametric Surfaced)：

可以用下面的公式來表示：

X = X(U, V)

Y = Y(U, V)

Z = Z(U, V)

3. 任意型式表面(Free-Form Surface)

此種任意型式表面可以很容易地表現大部分複雜的曲線物體；而最常用的是 Piecewise Bicubic Surfaces，例如：Bezier Surfaces， B-spline Surfaces 和 Beta-Spline Surfaces。

2.2.3 體積表示方式 (Volumes/Solids Representation)

體積表示方式(或稱爲立體表示方式)之所以重要是因爲真實物體都是立體的，它給予影像合成的衝擊在於下列兩點原因：

1. 物體的集合運算很難適用於非立體的物體。
2. 表示物體的方式(如：八元樹(Octree))非常適用於光線追蹤(Ray tracing)的技術。

表現立體物體的方法有許多種，其中最重要的有：空間分解(Space Decomposition)，CSG(Constructive Solid

Modeling)和 B-rep。

2.3 立體物體模型 (Solid Modeling)

2.3.1 立體物體的表現方法

大部份的學者認為表現立體物體的方法有六種，但是並不包括會產生混淆的線條表示法 (Wireframe representation)

1. 基本物體表現 (Primitive instances)
2. 空間計算 (Spatial Occupancy enumeration)
3. 元件分解 (Cell decomposition)
4. Sweep 表示法 (Sweep representation)
5. CSG (Constructive Solid Geometry)
6. 邊界表示法 (Boundary Representation)

另外，第七類的立體物體稱之為 Free-Form Solids，Varady 和 Pratt(1984)整合 Free-Form Surfaces 於 Build Solid Modeler 中，它是基於 B-rep 的模型而發展出來。

由於本系統中使用的表示方法有 CSG 、B-rep 和 Sweep 表示法，因此，我們就這三種表示法做一說明。

2.3.1.1 CSG

在 CSG 模型中，每一個基本物體是藉由 Half-space

的組合來定義，使用者並不能直接處理單一的 Half-space ；在表現(instance)方面，使用者可以由基本物體的幾何特性來定義並產生基本物體的 Half-spaces 集合，但是最重要的是不能去處理這些單一的 Half-spaces。

表示 CSG 模型最常用的方式是使用所謂的 CSG-Tree ，其定義如下：

<CSG Tree> ::= <Primitive> |
<CSG Tree> <Set operation> <CSG Tree> |
<CSG Tree> <Rigid motion> <Motion Args>

其中，<Primitive > 是一個基本物體，是由物體的型態和幾何特性參數所組成；<Rigid motion> 則是平移或旋轉；<Set operation> 則是聯集、交集、差集其中之一。

CSG 中，每一個基本物體必須是定義於 E^3 的有界點集合，(Bounded point set of E^3)，由於集合運算並不會破壞邊界，所以 CSG 模型保証在有界集合的定義中。

CSG 最大的優點是所需的記憶體較小，對於基本物體的輸入，例如球體，只要輸入球心的座標和半徑的長度就可以；但是相對的，在表現方式便需要其他表示方式的輔

助，因爲若直接由 CSG 來表示，速度較慢而且很麻煩。

CSG 模型的基本物體通常都是屬於 R-set，所以，當 CSG 的基本物體爲 R-set 時，任何 CSG-Tree 都是 R-set 的合法表示；CSG 系統的功能是依靠可用的基本物體集合而增強；一般說來，在 CSG 系統中，只考慮有界半空間 (Bounded half spaces)，部分商業用系統提供的有界半空間包括：平面(Planar)、圓錐形(Conical)、圓柱形 (Cylindrical)、球形(Spherical)、和圓環形(Toroids)等表面，圖 4 是一個 CSG-Tree 的傳統例子。

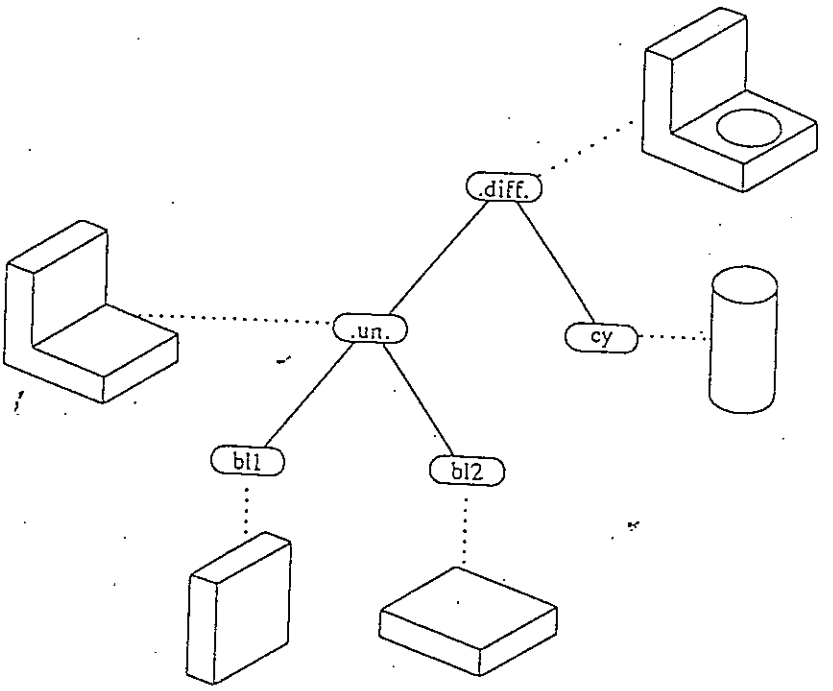


圖 4 CSG-TREE

2.3.1.2 B-rep 模型

B-rep 模型是直接經由邊界表面來表示一個立體物體，也就是在電腦繪圖中使用多面體模型來表現一個物體，並且做隱線(Hidden line)和隱面(Hidden surface)的消除。

B-rep 模型的資料結構一般可以分爲 Polygon-Based，Vertex-Based 和 Edge-Based：

1).Polygon-Based Boundary Model

此種模型亦稱爲多面體模型(Polyhedra model)，此模型中的物體都是以平面(Planar surface)來組成，這些平面是以表格(Table)或是串列(Linked list)的方式來組成一個立體物體，有時平面之間的關係也隱含在這些結構中。通常，這種表示方式是用於繪圖系統的中間圖形檔案(Graphical metafiles)。

2).Vertex-Based boundary Model

此模型與 Polygon-Based 模型不同之處，在於加入 Vertex，也就是平面是以 Vertex 的集合來表示，因此也就多了 Vertex 的座標值，其表示法見圖 5，其中 Vertex 是以順時鐘的方式來表示

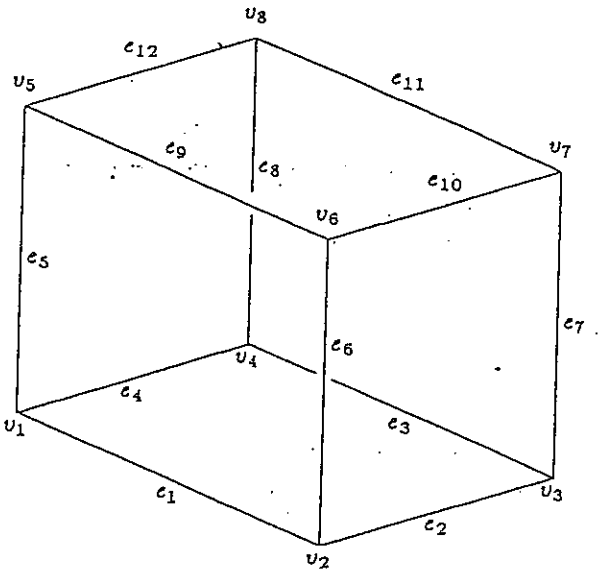


圖 5(a)　一個立體物體

vertex	*coordinates*
v_1	$x_1\ y_1\ z_1$
v_2	$x_2\ y_2\ z_2$
v_3	$x_3\ y_3\ z_3$
v_4	$x_4\ y_4\ z_4$
v_5	$x_5\ y_5\ z_5$
v_6	$x_6\ y_6\ z_6$
v_7	$x_7\ y_7\ z_7$
v_8	$x_8\ y_8\ z_8$

face	*vertices*
f_1	$v_1\ v_2\ v_3\ v_4$
f_2	$v_6\ v_2\ v_1\ v_5$
f_3	$v_7\ v_3\ v_2\ v_6$
f_4	$v_8\ v_4\ v_3\ v_7$
f_5	$v_5\ v_1\ v_4\ v_8$
f_6	$v_8\ v_7\ v_6\ v_5$

圖 5(b)　Vertex-Based 模型

一個平面，這種方法在許多演算法中是相當有用的。例如隱線、隱面的消除，以圖 5 的例子來看，f1、f4 和 f5 在隱面的演算法中將是隱面。

3).Edge-Based Boundary Model

如果曲面是由 B-rep 模型來表示的話，將邊(Edge)包含在此資料結構中將變得非常有用，因爲它能夠儲存曲面交集的資訊(邊對於記錄物體的拓樸關係也是非常有用的)，此模型是藉由邊的封閉順序(Closing sequence)或簡稱爲迴圈 (Loop)來表示一個面的邊界，表面的 Vertex 只有經由邊才能表示出來，其資料結構見圖 6 。

Edge-Based Boundary Model 之中還有一種所謂的 Winged-edge data structure，這種資料結構首先由 Baumgart [Bau,'74,'75]提出來，其結構見圖 7 。

2.3.1.3 Sweep representation

Sweep 表式方法的基礎是沿著一個路徑(Path)或是軌道(Trajectory)移動一個點、曲線或是表面；對於立體物體模型，Sweep 表示方法需要兩個要素，一被移動的物體

edge	*vertices*
e_1	v_1 v_2
e_2	v_2 v_3
e_3	v_3 v_4
e_4	v_4 v_1
e_5	v_1 v_5
e_6	v_2 v_6
e_7	v_3 v_7
e_8	v_4 v_8
e_9	v_5 v_6
e_{10}	v_6 v_7
e_{11}	v_7 v_8
e_{12}	v_8 v_5

vertex	*coordinates*
v_1	x_1 y_1 z_1
v_2	x_2 y_2 z_2
v_3	x_3 y_3 z_3
v_4	x_4 y_4 z_4
v_5	x_5 y_5 z_5
v_6	x_6 y_6 z_6
v_7	x_7 y_7 z_7
v_8	x_8 y_8 z_8

face	*edges*
f_1	e_1 e_2 e_3 e_4
f_2	e_9 e_6 e_1 e_5
f_3	e_{10} e_7 e_2 e_6
f_4	e_{11} e_8 e_3 e_7
f_5	e_{12} e_5 e_4 e_8
f_6	e_{12} e_{11} e_{10} e_9

图 6 Edge-Based 模型

edge	*vstart*	*vend*	*ncw*	*nccw*
e_1	v_1	v_2	e_2	e_5
e_2	v_2	v_3	e_3	e_6
e_3	v_3	v_4	e_4	e_7
e_4	v_4	v_1	e_1	e_8
e_5	v_1	v_5	e_9	e_4
e_6	v_2	v_6	e_{10}	e_1
e_7	v_3	v_7	e_{11}	e_2
e_8	v_4	v_8	e_{12}	e_3
e_9	v_5	v_6	e_6	e_{12}
e_{10}	v_6	v_7	e_7	e_9
e_{11}	v_7	v_8	e_8	e_{10}
e_{12}	v_8	v_5	e_5	e_{11}

vertex	*coordinates*
v_1	x_1 y_1 z_1
v_2	x_2 y_2 z_2
v_3	x_3 y_3 z_3
v_4	x_4 y_4 z_4
v_5	x_5 y_5 z_5
v_6	x_6 y_6 z_6
v_7	x_7 y_7 z_7
v_8	x_8 y_8 z_8

face	*first edge*	*sign*
f_1	e_1	+
f_2	e_9	+
f_3	e_6	+
f_4	e_7	+
f_5	e_{12}	+
f_6	e_9	−

图 7 Winged-Edge Data Structure

和所遵循的軌道；這個物體可以是曲線、表面或是立體物體，軌道則是一條可定義的路徑；通常，Generator 是表示一個區域，即點、曲線或表面； Director 是用於表示路徑或軌道；圖 8 是一個 Sweep 表示方法的例子。

Sweeping 的方式可分爲平移式(Translational)、旋轉式(Rotational)、一般式(General)和動畫式(Animation method)：

1).Translational sweeping

物體 S_t 的產生是由平移一個二維的區域 C，沿著向量 T 移動一距離 d，如圖 9 所示； 曲線所在的平面 P_0 是稱之爲基本平面 (Base plane)，距離 d 的平面 P_1 是稱之爲主要平面 (Cap plane)。

2).Rotational Sweeping

此方程式產生的物體是藉由沿著 A 軸旋轉一個二維的區域所產生； 其中有兩個重點---基本點 B 和半徑函數(Radius Function) P(s)，如圖 10 所示；一般由旋轉式 Sweeping 所產生的物體都是旋轉式物體(Revolution Bodies)，這種物體是將一個曲線沿著某一個軸旋轉 360 度所產生。

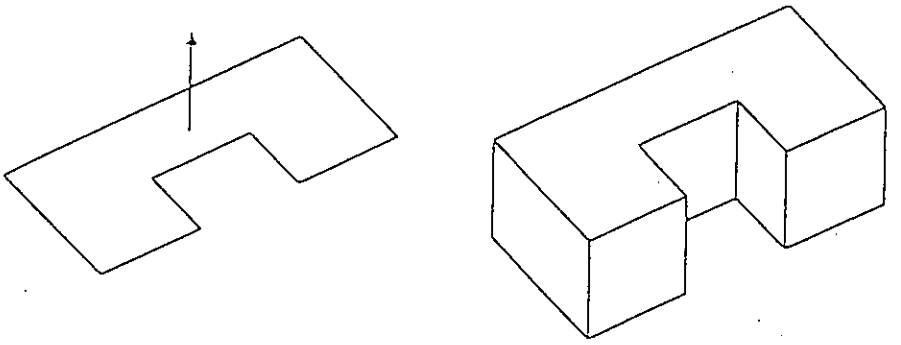


圖 8　由 Sweeping 方式架構的立體物體

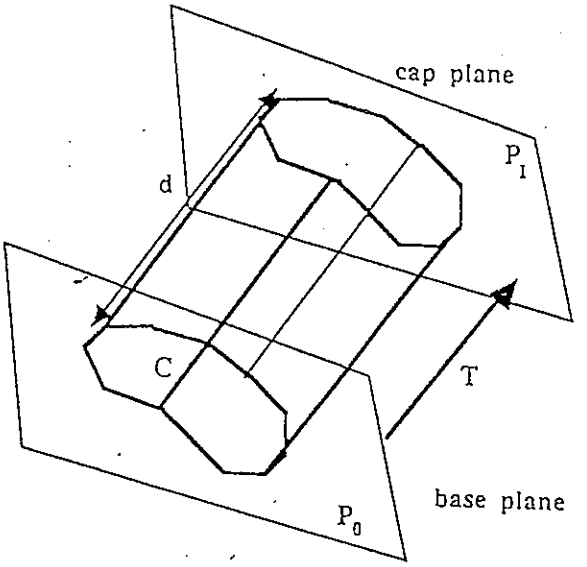


圖 9　Translational Sweeping

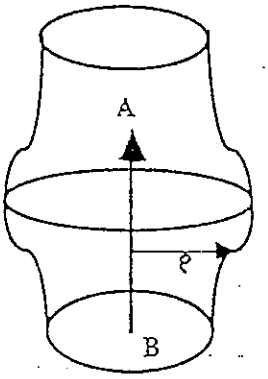


圖 10 Rotational Sweeping

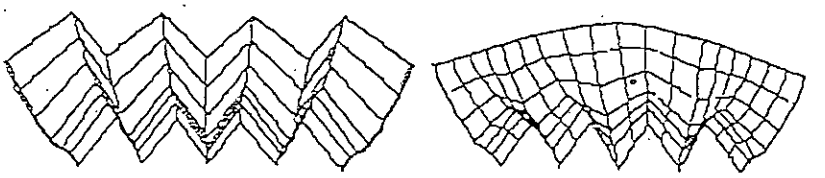


圖 11 C1 曲線沿著 C2 曲線 Sweep

a. C1 沒有變化

b. C2 有變化

3).General Sweeping

另一種重要的 Sweeping 方式是圓錐式 Sweeping (conic sweeping)，類似平移式 Sweeping，但是結合了線條的比例(Scaling)；這個方法是由 Lossing 和 Eshleman (1974) 所發展出來，他們使用了 6 個元素的軌道和原始的曲線，稱之爲 Positional-direction (PD) curve，見圖 11 ；使用 PD Curve，幾乎無限制的 Sweeping 物體都可以定義出來。

4).Animation Method

這種方法是將電腦繪圖動畫中的物體移動觀點應用於 Sweeping 中，由於在電腦繪圖動畫中造成物體移動的現象是依時間不同而將物體顯現在不同的位置，於 Sweeping 的方式中，物體的產生是曲線沿著軸旋轉所產生，而這個旋轉的過程中，可以加入時間的函數，類似動畫的產生，曲線可以隨時間的不同，而有不同的形狀，因此，便可產生許多複雜的物體，我們可以將這過程區別爲下列的方式：

1.Generator 可以區分爲四個屬性---形狀(Shape

)、位置(Position)、規則 (Orientation)和大小(Size)。

2.三維物體的產生是沿著一條路徑 (Director)移動 Generator。

3.在移動的過程中，依照旋轉的規則改變 Generator 的屬性。

2.3.1.4 Hybrid Model

在前面二節，說明了兩種最主要的立體物體模型，在此我們再說明一些討論的重點：

1).CSG 模型：

這是最簡潔的立體物體模型，但是如果直接產生圖形輸出或是用數值演算法來產生資料，速度將會變得很慢，通常這些工作是轉換成別種表示方式來處理。

2).B-rep 模型：

對圖形應用來說是非常有用的，對數值應用來說，如果多面體的空間限制能夠有更大的誤差容忍，則多面體模型更是特別地有用，但是，如果沒有其他表示方法和轉換(Conversion)的幫忙，多面體的產生將會是很困難的。

很明顯的，各種模型的組合方式將優於任何單一的模型，在混合模型中，對於同一組資料有多種的儲存方式，在特定的表現方式則配合使用最適合的資料結構。

在混合型模型中的問題是很難選擇以那一種表示方法爲主要的表示方式，那一種爲第二選擇；在現在的系統架構中，最常用的兩種架構如圖 12 所示，一種是以 CSG 爲主，如圖 12(a)，另一種是以 B-rep 爲主，以 CSG 爲輔，如圖 12(b)。

2.4 系統描述

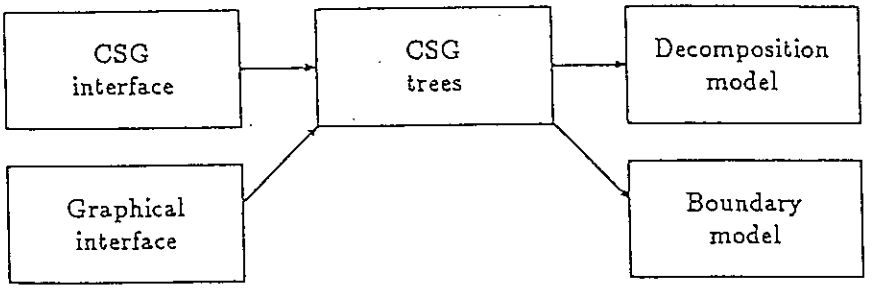
由前面所討論的立體物體模型中，我們可以發現單一的立體物體表示法都有其優缺點，因此，現在所發展的系統都趨向於結合多種表示方法的混合型系統，而本系統的研究也是以混合型爲處理的目標，希望能就使用者介面和內部處理上達到一個更優良的環境。

本系統可分爲以下兩方面來討論：

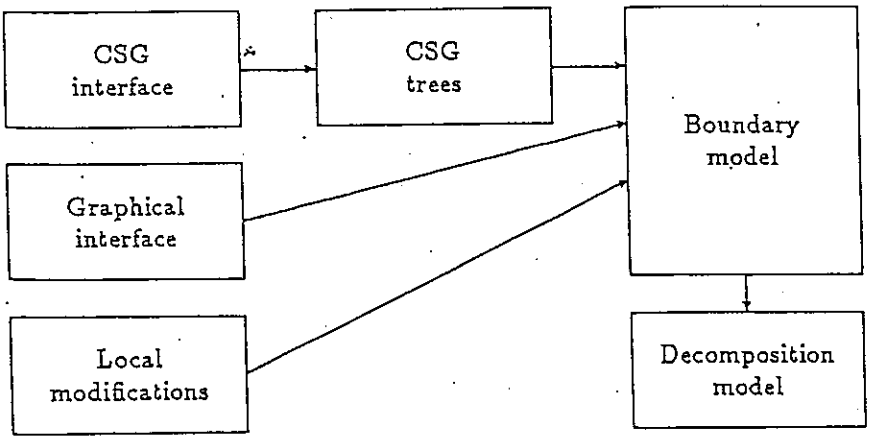
1. 基本物體的產生
2. 集合運算

2.4.1 基本物體的產生

在基本物體的產生上，我們是以類似 CSG 模型的輸



(a)



(b)

圖 12　混合模型的架構

入方式(亦即輸入基本物體的幾何特性)來產生，除了球、圓錐體、圓柱體、圓環體、矩形體等基本物體之外，還加入了超二次(Superquadric)的數學方法 [Bar,82']，超二次數學方法所能產生的物體見圖 13 ；另外，我們還加入用 Sweep 架構法來產生基本物體。

在內部處理上，我們是採用 B-rep 的表示方式，更詳細地說，是採用 Vertex-Based Boundary Model ，當儲存於檔案中時，我們所採用的檔案格式如下：

```
String
m, n
x, y, z
   .
   .
   .
k, a, b, c
   .
   .
   .
```

其中，String 是一說明字串，有 m 個 Vertex，有 n 個 Polygons，(x, y, z)是點座標，k 表示 Polygon是由 k 個 Vertex 所組成的，a，b，c 是點的編號順序，而 Polygon 的點是以順時鐘的方式排列，形成一個封閉的區域。

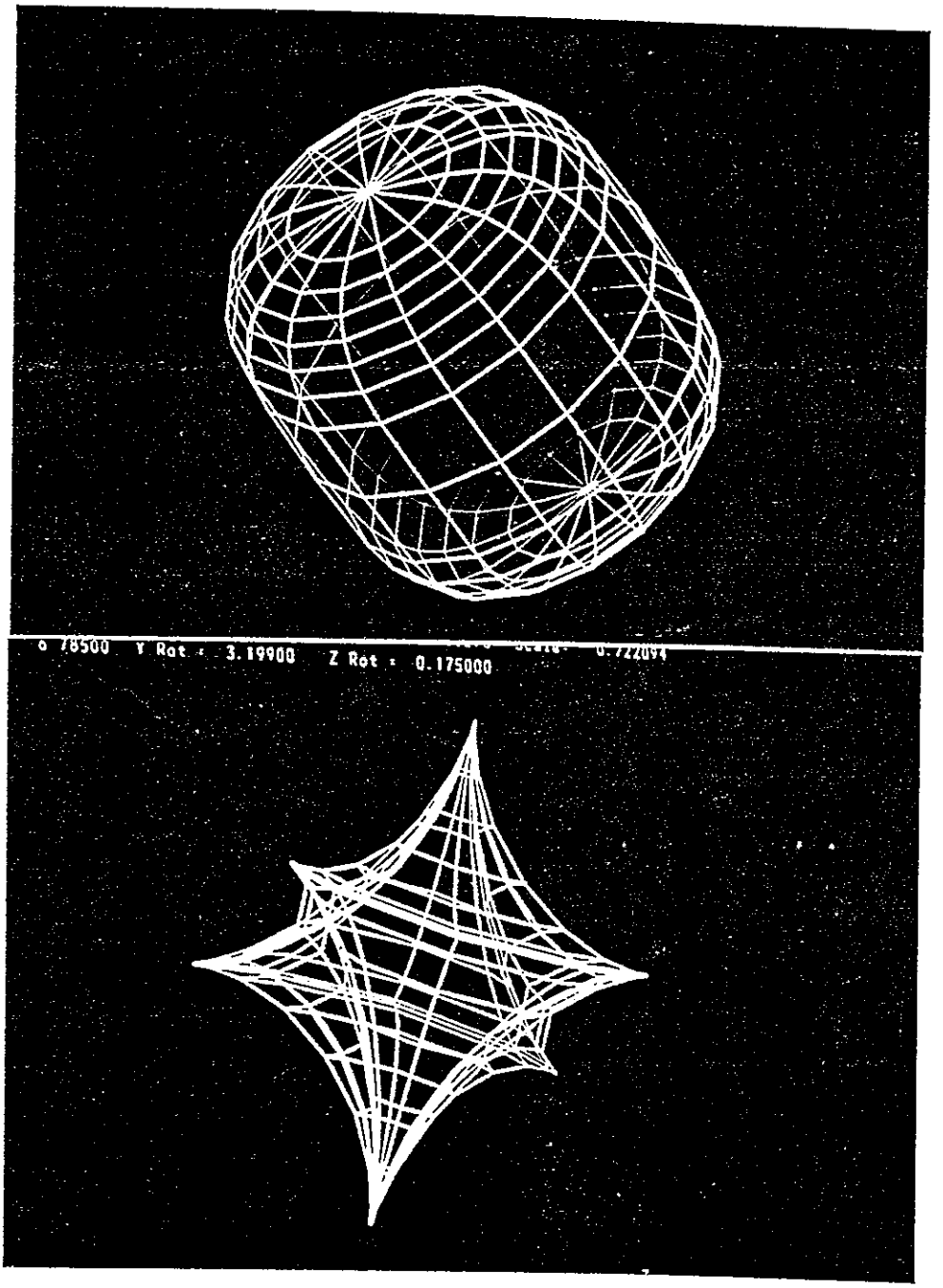


圖 13　超二次數學方法產生的物體

因此，除了我們所提供產生基本物體的方法之外，若要再增加基本物體的產生方法，只要檔案符合前面所說的格式，便可使用我們所提供的集合運算，也使得我們的系統在基本物體的產生上具有擴充性。

2.4.2 集合運算

本系統中所提供的集合運算包括：聯集(Union)、交集(Intersection)和差集(Difference)，其指令如下：

聯集：Setop -u obj1 obj2 obj3

將 obj1 與 obj2 做聯集運算，結果存於 obj3。

交集：Setop -i obj1 obj2 obj3

將 obj1 與 obj2 做交集運算，結果存於 obj3。

差集：Setop -d obj1 obj2 obj3

將 obj1 與 obj2 做交集運算，結果存於 obj3。

除了集合運算之外，還有平移、旋轉、放大、縮小等基本運算。

本系統中，是單純就 B-rep 模型中，表示物體的結構做處理，也就是從面、邊和點來考慮；演算法將在下一章中做說明。

第三章 數學方法及理論基礎

3.1 基本物體產生的方法

本系統中基本物體產生的方法可分爲三種方式：

1. Pfsolid： 這是 USNA 所發展的一套套裝軟體中基本物體產生的方法， 輸入的方式是採用 CSG 的輸入方式，亦即輸入物體的幾何特性，如：球心、半徑，所產生檔案的格式如前一章所說明的，是 B-rep 中的一種 (Vertex-Based Boundary Representation)，所能產生的物體包括： 長方體(Cube)、圓柱體(Cylinder)、圓錐體(Cone)、球體(Sphere)、橢球體(Ellipsoid)、圓環體 (Toroids)等。
2. Superquadric Method：由於 Pfsolid 所能提供的基本物體非常有限，因此，我們加入這種物體產生方式，這種數學方式的最主要優點是能夠架構出複雜的立體物體和表面，並且經由一些交談性的參數便可以容易地改變物體的形狀；這種方式是 Barr 在 1982 年所提出來，這也是本系統最主要產生物體的方式，其詳細的方法將在下一

節中討論。

3. Sweep 架構法：在第二章裡，我們曾提過，此種方法是立體物體模型中一個重要的方法，雖然它所能產生的立體物體種類非常多，但是仍然有所限制，例如：椅子，由於椅子呈現出多軸的情況，以致於無法用 Sweep 架構法描述出來，所以我們得用其它方法來解決；但是，由於 Sweep 架構法能夠產生許多的立體物體，這項優點使我們決定將之使用於基本物體的產生，以使得所能使用的基本物體種類更加具有多樣性，當然亦能使得立體物體的產生過程更加容易。

3.2 超二次數學方法 (Superquadrics Method)

此方法是 Barr 於 1982 年提出，其重點如下：

給定兩條二維曲線：

$$h(w) = \begin{bmatrix} h_1(w) \\ h_2(w) \end{bmatrix} \quad , \; w_0 \le w \le w_1$$

和

$$m(\eta) = \begin{bmatrix} m_1(\eta) \\ m_2(\eta) \end{bmatrix} \quad , \; \eta_0 \le \eta \le \eta_1$$

將兩條曲線做球外積 (spherical product) $X=m \times h$ 則定義了一個表面：

$$x(\eta,w) = \begin{bmatrix} m_1(\eta)h_1(w) \\ m_1(\eta)h_2(w) \\ m_2(\eta) \end{bmatrix} \quad \begin{matrix} w_0 \le w \le w_0 \\ , \eta_0 \le \eta \le \eta_0 \end{matrix}$$

以幾何來說，$h(w)$ 是一條水平曲線，而由 $m(\eta)$ 調整其垂直性；$m_1(\eta)$ 改變 h 相對的大小，其中 $m_2(\eta)$ 可以增大或變小這個比例； η 是由北向南的參數，就像經度一樣，而 w 是由東向西的參數，就像緯度一樣；球外積表面可以用向量 a 來改變其比例，如：

$$a = \begin{bmatrix} a_1 \\ a_2 \\ a_3 \end{bmatrix} \text{ 假設 } m = \begin{bmatrix} m_1(\eta) \\ \\ a_3 m_2(\eta) \end{bmatrix}, \; h = \begin{bmatrix} a_1 h(w) \\ \\ a_2 h(w) \end{bmatrix}$$

$$\text{產生 } x = m \times h = \begin{bmatrix} a_1 m_1(\eta)h_1(w) \\ a_2 m_1(\eta)h_2(w) \\ a_3 m_2(\eta) \end{bmatrix}$$

球外積名字的產生是由單位球體表面而來，當一個半圓掃過一個圓時，便產生一個球體：

$$m(\eta) = \begin{bmatrix} \cos \eta \\ \\ \sin \eta \end{bmatrix}, \; -\pi/2 \le \eta \le \pi/2$$

$$h(w) = \begin{bmatrix} \cos w \\ \sin w \end{bmatrix}, \quad -\pi \le w \le \pi$$

也就是

$$x = m \times h = \begin{bmatrix} \cos \eta \cos w \\ \cos \eta \sin w \\ \sin \eta \end{bmatrix} \quad \begin{matrix} -\pi/2 \le \eta \le \pi/2 \\ , \ -\pi \le w \le \pi \end{matrix}$$

標準 Quadric 表面的基本三角形式表示如下：

Ellipsoids：

$$(x_1/a_1)^2 + (x_2/a_2)^2 + (x_3/a_3)^2 = 1$$

$$x(\eta, w) = \begin{bmatrix} \cos \eta \\ a_3 \sin \eta \end{bmatrix} \times \begin{bmatrix} a_1 \cos w \\ a_2 \sin w \end{bmatrix}$$

$$= \begin{bmatrix} a_1 \cos \eta \cos w \\ a_2 \cos \eta \sin w \\ a_3 \sin \eta \end{bmatrix} \quad \begin{matrix} -\pi/2 \le \eta \le \pi/2 \\ -\pi \le w < \pi \end{matrix}$$

Hyperboloids of one sheet：

$$(x_1/a_1)^2 + (x_2/a_2)^2 - (x_3/a_3)^2 = 1$$

$$x(\eta, w) = \begin{bmatrix} \sec \eta \\ a_3 \tan \eta \end{bmatrix} \times \begin{bmatrix} a_1 \cos w \\ a_2 \sin w \end{bmatrix}$$

$$= \begin{bmatrix} a_1 \sec \eta \cos w \\ a2 \sec \eta \sin w \\ a3 \tan \eta \end{bmatrix} \quad \begin{matrix} -\pi/2 \le \eta \le \pi/2 \\ , \ -\pi \le w < \pi \end{matrix}$$

Hyperboloids of two sheet :

$$(x_1/a_1)^2 - (x_2/a_2)^2 - (x_3/a_3)^2 = 1$$

$$x(\eta,w) = \begin{bmatrix} \sec\eta \\ a_3\tan\eta \end{bmatrix} \times \begin{bmatrix} a_1\sec w \\ a_2\tan w \end{bmatrix}$$

$$= \begin{bmatrix} a_1\sec\eta\sec w \\ a_2\sec\eta\tan w \\ a_3\tan\eta \end{bmatrix}$$

$$-\pi/2 < \eta < \pi/2$$
$$, -\pi/2 < w < \pi/2 \quad \text{(one sheet)}$$
$$\pi/2 < w < 3\pi/2 \quad \text{(two sheets)}$$

Torus :

$$(r - a)^2 = (x_3/a_3)^2 = 1$$

其中 $r = \sqrt{(x_1/a_1)^2 + (x_2/a_2)^2}$

$$x(\eta,w) = \begin{bmatrix} a+\cos\eta \\ a_3\sin\eta \end{bmatrix} \times \begin{bmatrix} a_1\cos w \\ a_2\sin w \end{bmatrix}$$

$$= \begin{bmatrix} a_1(a+\cos\eta)\cos w \\ a_2(a+\cos\eta)\sin w \\ a_3\sin\eta \end{bmatrix} \quad -\pi \le \eta < \pi$$
$$, -\pi \le w < \pi$$

超二次立體物體是基於 Quadric 表面的四個參數形式，其中，每一個三角函數都提升爲指數；

在二維的 sin-cos 曲線：

$x = a\cos^{\epsilon}\theta$ 或是 $(x/a)^{2/\epsilon} + (y/b)^{2/\epsilon} = 1$

$y = b\sin^{\epsilon}\theta$ $\quad -\pi \le \theta < \pi$

稱爲超橢球 (Superellipse) 。

secant-tangent曲線：

$x = a \sec^{\epsilon}\theta$ 或是 $(x/a)^{2/\epsilon} + (y/b)^{2/\epsilon} = 1$

$y = b \tan^{\epsilon}\theta$ $-\pi/2 \le \theta < \pi/2$

$\pi/2 \le \theta \le 3\pi/2$

稱爲 Superhyperbola。

在超二次的數學方法中，指數是平方參數，用於調整物體的形狀，如枕形 (pinch) 、圓形 (round) 、和方形 (square) ：

$\epsilon < 1$ ：形狀成爲方形 (square)

$\epsilon \approx 1$ ：形狀爲圓形 (round)

$\epsilon \approx 2$ ：形狀爲平斜形 (flat bevel)

$\epsilon > 2$ ：形狀爲枕形 (pinch)

超二次的物體爲數學上的立體物體，因爲 ：

1). 其表面將三維空間分爲三個不同的區域，內部 (inside) 、外部 (outside) 和表面邊界 (surface boundary)

2). 有一個 Inside-Outside 函數，可以決定一個點落於那一個區域。

例如：一個單位球體

$$x(\eta,w) = \begin{bmatrix} \cos\eta\cos w \\ \cos\eta\sin w \\ \sin\eta \end{bmatrix} \quad \begin{matrix} -\pi/2 \le \eta \le \pi/2 \\ -\pi \le w < \pi \end{matrix}$$

其 Inside-Outside 函數爲

$$f(X, Y, Z) = X^2 + Y^2 + Z^2$$

$$\text{if} \begin{cases} f(x_o,y_o,z_o) = 1 & (x_o,y_o,z_o) \text{ 在表面上} \\ f(x_o,y_o,z_o) > 1 & (x_o,y_o,z_o) \text{ 落於球體之外} \\ f(x_o,y_o,z_o) < 1 & (x_o,y_o,z_o) \text{ 落於球體之內} \end{cases}$$

Superellipsoids：

表面的位置向量：

$$x(\eta,w) = \begin{bmatrix} C\eta^{\epsilon_1} \\ a_3 S\eta^{\epsilon_1} \end{bmatrix} \times \begin{bmatrix} a_1 C\omega^{\epsilon_2} \\ a_2 S\omega^{\epsilon_2} \end{bmatrix}$$

$$= \begin{bmatrix} a_1 C\eta^{\epsilon_1} C\omega^{\epsilon_2} \\ a_2 C\eta^{\epsilon_1} S\omega^{\epsilon_2} \\ a_3 S\eta^{\epsilon_1} \end{bmatrix} \quad \begin{matrix} -\pi/2 \le \eta \le \pi/2 \\ , -\pi \le w < \pi \end{matrix}$$

ϵ_1：南北方向的平方參數

ϵ_2：東西方向的平方參數

Normal vector ：

$$n(\eta,w) = \begin{bmatrix} 1/a_1 C\eta^{2-\epsilon_1} C\omega^{2-\epsilon_2} \\ 1/a_2 C\eta^{2-\epsilon_1} S\omega^{2-\epsilon_2} \\ 1/a_3 S\eta^{2-\epsilon_1} \end{bmatrix}$$

Inside-Outside function :

$$f(x,y,z) = ((x/a_1)^{2/\epsilon_2} + (y/a_2)^{2/\epsilon_2})^{\epsilon_2/\epsilon_1} + (z/a_3)^{2/\epsilon_1}$$

Superhyperboloids :

表面的位置向量 :

$$x(\eta,w) = \begin{bmatrix} \sec^{\epsilon_1}\eta \\ a_3\tan^{\epsilon_1}\eta \end{bmatrix} \times \begin{bmatrix} a_1 C_\omega^{\epsilon_2} \\ a_2 S_\omega^{\epsilon_2} \end{bmatrix}$$

$$= \begin{bmatrix} a_1\sec^{\epsilon_1}\eta C_\omega^{\epsilon_2} \\ a_2\sec^{\epsilon_1}\eta S_\omega^{\epsilon_2} \\ a_3\tan^{\epsilon_1}\eta \end{bmatrix} \quad , \begin{array}{l} -\pi/2 \le \eta \le \pi/2 \\ -\pi \le w < \pi \end{array}$$

Normal vector :

$$n(\eta,w) = \begin{bmatrix} 1/a_1\sec\eta^{2-\epsilon_1} C_\omega^{2-\epsilon_2} \\ 1/a_2\sec\eta^{2-\epsilon_1} S_\omega^{2-\epsilon_2} \\ 1/a_3\tan\eta^{2-\epsilon_1} \end{bmatrix}$$

Inside-Outside function :

$$f(x,y,z) = ((x/a_1)^{2/\epsilon_2} + (y/a_2)^{2/\epsilon_2})^{\epsilon_2/\epsilon_1} - (z/a_3)^{2/\epsilon_1}$$

Supertoroids :

表面的位置向量 :

$$x(\eta,w) = \begin{bmatrix} a + C\eta^{\epsilon_1} \\ a_3 S\eta^{\epsilon_1} \end{bmatrix} \times \begin{bmatrix} a_1 C_\omega^{\epsilon_2} \\ a_2 S_\omega^{\epsilon_2} \end{bmatrix}$$

$$= \begin{bmatrix} a_1(a + C\eta^{\epsilon_1}) C\omega^{\epsilon_2} \\ a_2(a + C\eta^{\epsilon_1}) S\omega^{\epsilon_2} \\ a_3 S\eta^{\epsilon_1} \end{bmatrix} \quad -\pi \le \eta < \pi, \; -\pi \le w < \pi$$

Normal vector :

$$n(\eta, w) = \begin{bmatrix} 1/a_1 C\eta^{2-\epsilon_1} C\omega^{2-\epsilon_2} \\ 1/a_2 C\eta^{2-\epsilon_1} S\omega^{2-\epsilon_2} \\ 1/a_3 S\eta^{2-\epsilon_1} \end{bmatrix}$$

Inside-Outside function :

$$f(x,y,z)$$

$$= ((x/a_1)^{2/\epsilon_2} + (y/a_2)^{2/\epsilon_2})^{\epsilon_2/\epsilon_1} - a)^{2/\epsilon_1}$$

$$+ (z/a_3)^{2/\epsilon_1}$$

其中， $a = \hat{a}/\sqrt{(a_1^2 + a_2^2)}$

$\hat{a}$ 是 torus 的半徑。

3.3 集合運算演算法

在定義好兩個立體物體之後，第一個物體的每一個面將與第一個物體的每一個面做計算，以求出其交集。

本演算法主要可分爲三個部份：

1. 找出兩個立體物體間的交集部份，亦即面與面之間有交點的部份。
2. 將兩個立體物體中所有的點在交集運算之後，予以分類，可分爲 Inside 、Outside 、On 。

3. 由分類出來的點將兩個立體物體的邊予以分類，由此，便可利用集合運算將所需的面找出來，至此，整個集合運算便完成並產生出結果。

3.3.1 物體的交集

首先，要說明點的資料結構：

```
struct vertex {
                float  x, y, z;
                  int  flag;
              }
```

其中，x，y，z 是點的座標值

flag 是用於判別此點是落於第二個物體的內部、外部、或是表面上。

此交集演算法是一個 N^2 階的演算法，對於第一個物體的每一個多邊形平面都要與第二個物體的每一個多邊形平面做比較；其中一個判斷的準則是：如果兩個多邊形平面之間沒有交集產生的可能性的話，則此部份的處理程序將立即停止，然後再接著做下一對的多邊形平面比較，這個判斷準則能夠減少不必要的處理程序，而加速處理的過程，減少處理所需的時間。

判斷準則一：判斷兩個多邊形平面之間是否有交集的可能

由於每一個多邊形平面都是由點連接而成，所以由多邊形平面其中的三個點就可以算出此多邊形平面的平面方程式；首先，算出第一個多邊形平面的平面方程式，再將第二個多邊形平面的每一邊依序與第一個多邊形平面做比較，由於每一個邊都有兩個端點，我們就稱爲 V_1 和 V_2，由平面方程式的特性，我們將 V_1 和 V_2 的座標值分別代入平面方程式，各別求出一個值 sign1 和 sign2，如果：

sign1 * sign2 > 0 ：

表示此邊位於這個多邊形平面之外或是之內（以平面方程式的 Normal 爲準，正向是外部，反向是內部）。

sign1 * sign2 = 0 ：

表示至少有一點落於此多邊形平面上，如果

sign1 <> 0 ， sign2 = 0

表示 v_2 落於此多邊形平面上。

sign1 = 0 ， sign2 <> 0

表示 v_1 落於此多邊形平面上。

sign1 = 0 ， sign2 = 0

表示 v_1、v_2 皆落於此多邊形平面上。

其中，不爲零的值，如果

大於零 ：表示位於這個多邊形平面之外。

小於零 ：表示位於這個多邊形平面之內。

sign1 * sign2 < 0 ：

表示兩個點分別落於此多邊形平面的兩邊，也就是這個邊與此多邊形所形成的平面有交點，因此，接下來的處理便是求出此交點並判斷此交點是否落於此多邊形內。

線段與平面的交點 ：

平面方程式 ： $AX + BY + CZ + D = 0$

線段方程式 ： $P(t) = P_1 + (P_2 - P_1)t$

$, 0 \le t \le 1$

$A[(x_2 - x_1)t + x_1] + B[(y_2 - y_1)t + y_1] +$

$C[(z_2 - z_1)t + z_1] + D = 0$

$$\Longrightarrow t = \frac{-(D + Ax_2 + By_2 + Cz_2)}{A(x_2 - x_1) + B(y_2 - y_1) + C(z_2 - z_1)}$$

判斷準則二：判斷邊與多邊形平面之交點是否落於此多邊形內

由多邊形平面的平面方程式的係數可以找出一個亦落

在此多邊形平面上的一點，與原先的交點形成一條有一個端點的射線，由此射線與多邊形各邊的交集次數總和可以判別此點是否落於此多邊形內，見圖 14 ，如果由交點開始向外延伸的射線與多邊形的邊有交集的次數總和是奇數，表示此交點是在多邊形內，如果交集的次數總和是偶數或是等於零，表示此交點不在多邊形內。

線段與線段的交點 ：

線段方程式 ： $P(t) = P_1 + (P_2 - P_1)t$

$, 0 \le t \le 1$

線段方程式 ： $P(s) = P_3 + (P - P_3)s$

$, 0 \le s \le 1$

$$x_1 + (x_2 - x_1)t = x_3 + (x - x_3)s \quad ---(1)$$

$$y_1 + (y_2 - y_1)t = y_3 + (y - y_3)s \quad ---(2)$$

$$z_1 + (z_2 - z_1)t = z_3 + (z - z_3)s \quad ---(3)$$

$$d = (y - y_3)(x_2 - x_1) - (y_2 - y_1)(x - x_3)$$

$$t = \frac{1}{d}[(y - y_3)(x_3 - x_1) - (y_3 - y_1)(x - x_3)$$

$$s = \frac{1}{d}[(y_2 - y_1)(x_3 - x_1) - (y_3 - y_1)(x_2 - x_1)$$

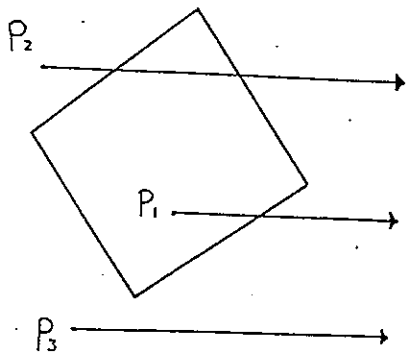


图 14

由 (1),(2) 式求得之 t, s 必須符合 (3) 式

3.3.2 點的分類

由兩平面之間邊與面的關係，便可以將點做一分類，如果邊與多邊形之間有交點，則一個邊就變成兩個邊，並且多了一個點，此點的 flag 就指定爲 ON ，表示落在表面上， 而另兩點則視其落在多邊形的那一側給予 SIN 、SOUT 的屬性，分別表示在表面之內或之外；如果邊與面之間沒有交點，便先給予 USIGN 的屬性，表示未定，其屬性於後會再處理。

當兩個物體的平面都做過交集的測試之後，在有交集的部份就會產生出新的點， 並且給予 ON 、SIN 、SOUT 的屬性， 其餘 USIGN 的部份就是整個平面爲 SIN 或是 SOUT，所以，接下來的工作就是將這些未定的點給予屬性；首先，將每一個多邊形的點做一分析，如果其中有 ON 則不處理，如果只有 SIN 和 USIGN ，或是只有 SOUT 和 USIGN ，則將其餘 USIGN 的點給予 SIN 或是 SOUT ，視另一屬性而定；這些處理將持續到沒有 USIGN 的點存在爲止。

當所有的點都有了確定的屬性之後，每一個邊的屬性

也都可以確定，其規則如下：

假設一個邊 L 的端點爲 V_1 和 V_2，則

V_1	V_2	L
SIN	SIN	IN
SOUT	SOUT	OUT
ON	ON	ON
ON	SIN	IN
ON	SOUT	OUT
SIN	ON	IN
SOUT	ON	OUT

3.3.3 組合物體

當所有邊的屬性也設定好之後，接下來的處理便是依據所指定的運算元（交集、聯集、差集）將所形成物體的面找出來，其規則可分爲三類來處理：

1. 聯集：

   假設第一個物體爲 OBJ1，第二個物體爲 OBJ2，

   形式如：　(OBJ1 UNION OBJ2)

   對於 OBJ1 和　OBJ2 來說，　所要找出的面是由 SOUT 和 ON 屬性的點所形成，而且面的 NORMAL

與原先面的 NORMAL 方向一致，在尋找的過程中，是可以形成一個封閉的多邊形的點爲目標，而這些資料是存在每一個面的資料結構中，因此可以很容易地完成。

2. 交集：

形式： (OBJ1 INTERSECT OBJ2)

在此運算中是要找出以 ON, SIN 爲屬性的點所形成的多邊形，而且所找出的多邊形其 NORMAL 與原先面的 NORMAL 相同。

3. 差集：

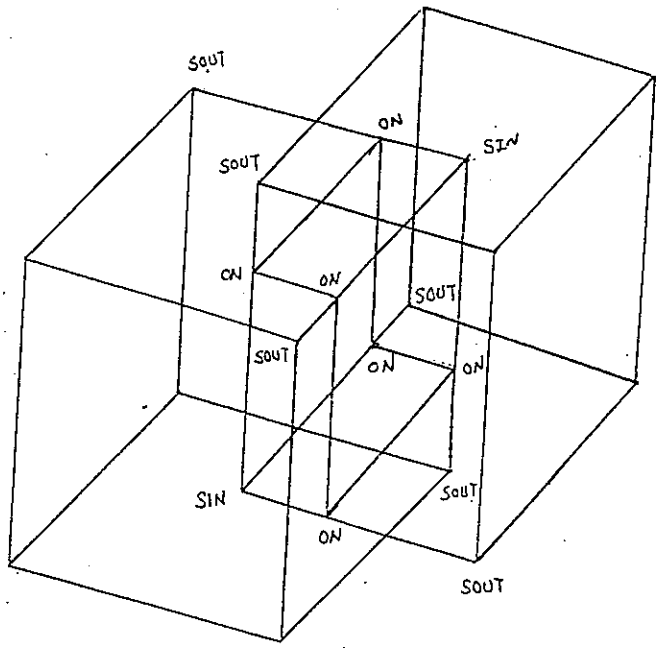
形式： (OBJ1 DIFFERENCE OBJ2)

在此運算中， OBJ1 和 OBJ2 所要使用到的點，其屬性稍有不同，對 OBJ1 來說，其方式和聯集運算過程相同，即找出以 ON, SOUT 爲屬性的點所形成的多邊形，而且 NORMAL 與原先的面相同；對 OBJ2 來說，是找出以 ON, SIN 爲屬性的點所形成的多邊形，而 NORMAL 則與原先的面的 NORMAL 相反。

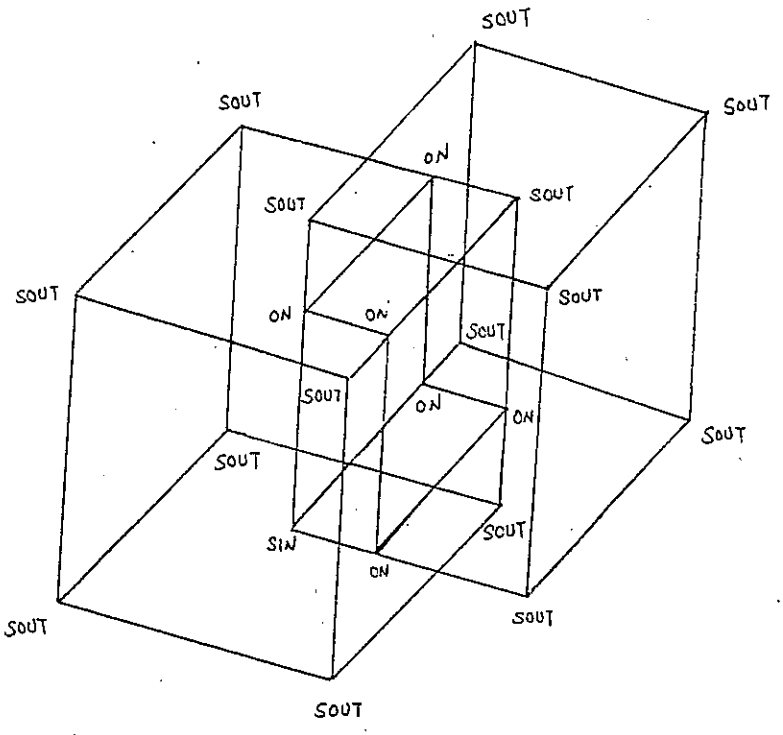
以下圖所示兩個 Cube 爲例：

在第一部份做完之後的結果，其中，

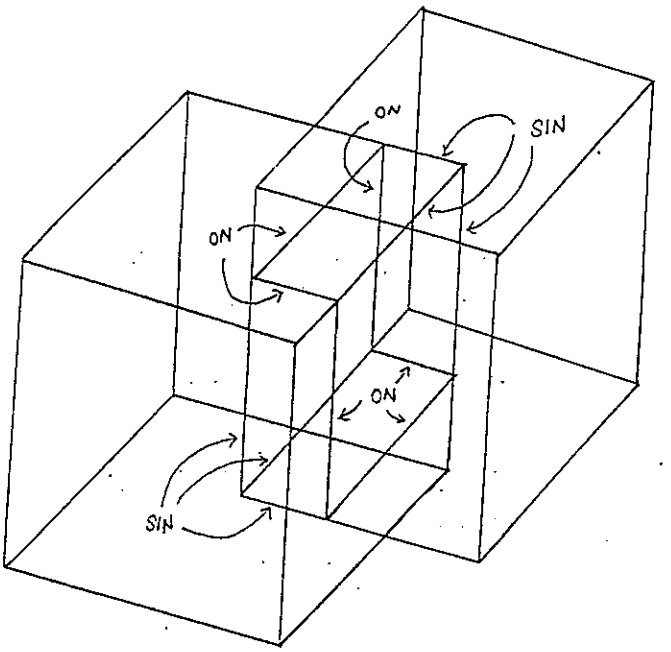
有交集的部份可以將 Vertex 之屬性先予以確定。



在第二部份 Vertex 分類做完之後，其結果如下圖：

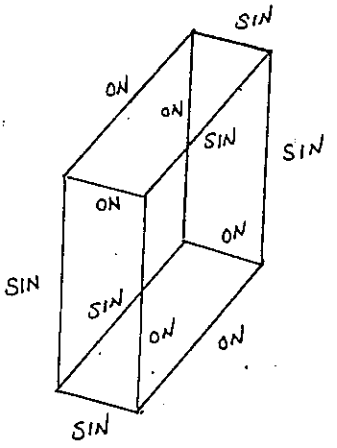


在第三部份是將所有的邊的屬性予以確定：

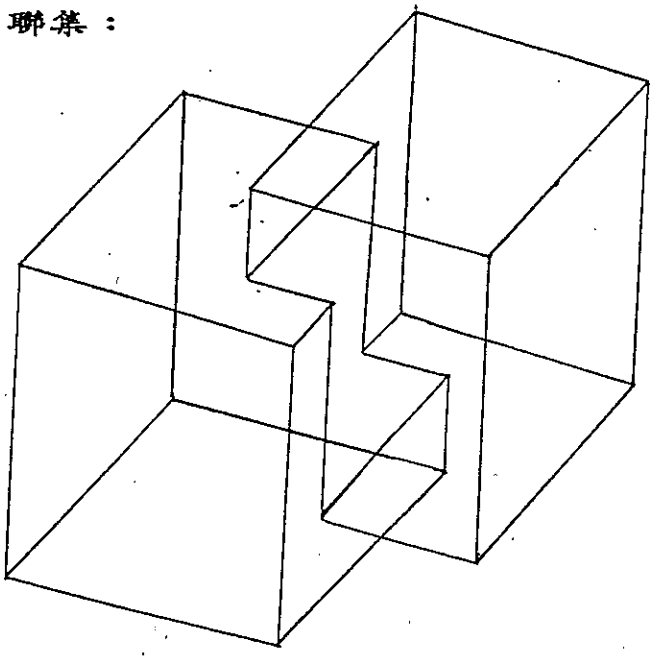


最後是將集合運算的結果輸出：

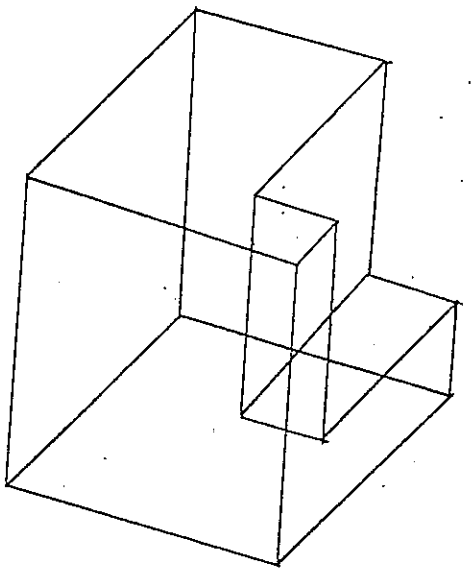
1. 交集：



2. 聯集：



3. 差集：



3.4 使用者介面與限制

基本上來說，本系統所提供的是立體物體模型的一套工具，除了基本的集合運算功能之外，希望能夠讓使用者在不需要具備電腦繪圖的深奧知識之下，亦能使用這套系統，因此，在使用者介面的設計上便需要有特殊的考慮。

由於使用者介面的設計上，必須要配合硬體的功能，而在我們發展的環境上有些實際上的困難，例如：硬體的複雜度，發展時間上的限制，因此，我們只做到了簡單的使用者介面。

使用者介面方面，我們將之分爲兩方面來處理，第一，是屬於技術人員方面，也就是具有電腦繪圖概念的人員，首先，將所要組合成的物體予以分解，將所需要的基本物體置於空間的那一點，做何集合運算等等，安排成一個循序步驟，賦予一個名稱，並且在觀念上，想像爲在一個 bounding box 之中，也就是說，要產生一個物體時，只要按照這些步驟，就可以產生出來，並且是置於此 bounding box 之中；第二方面，是屬於一般使用者方面，也就是不具備電腦繪圖概念的使用者，他只要指定此 bounding box 的大小及位置，這些物體便能自動以其大小比例產生，並置於指定的位置上。

第四章 實驗結果與討論

我們的實驗環境是以 VAX/750、SUN Workstation 和 E&S PS300 繪圖工作站爲主要設備，實驗程式則是發展在 UNIX 作業系統，計算過程是在 VAX/750 及 SUN Workstation 上執行，Rendering 則是在E&S PS300 繪圖工作站。

發展系統時，我們是以 PS300 之 CADIG/ANIMATOR 爲研究基礎，於其下加入集合運算功能。

我們將由下列的圖片來說明實驗的結果：

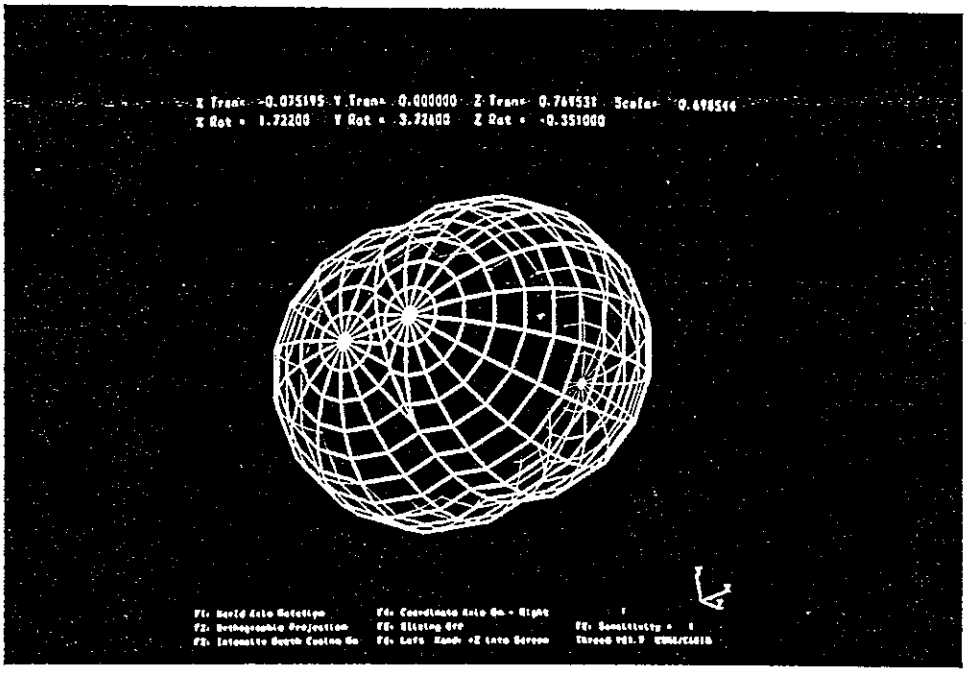


圖 15　兩個圓球體經由聯集運算(UNION Operation)後之結果，以WireFrame 之架構顯示

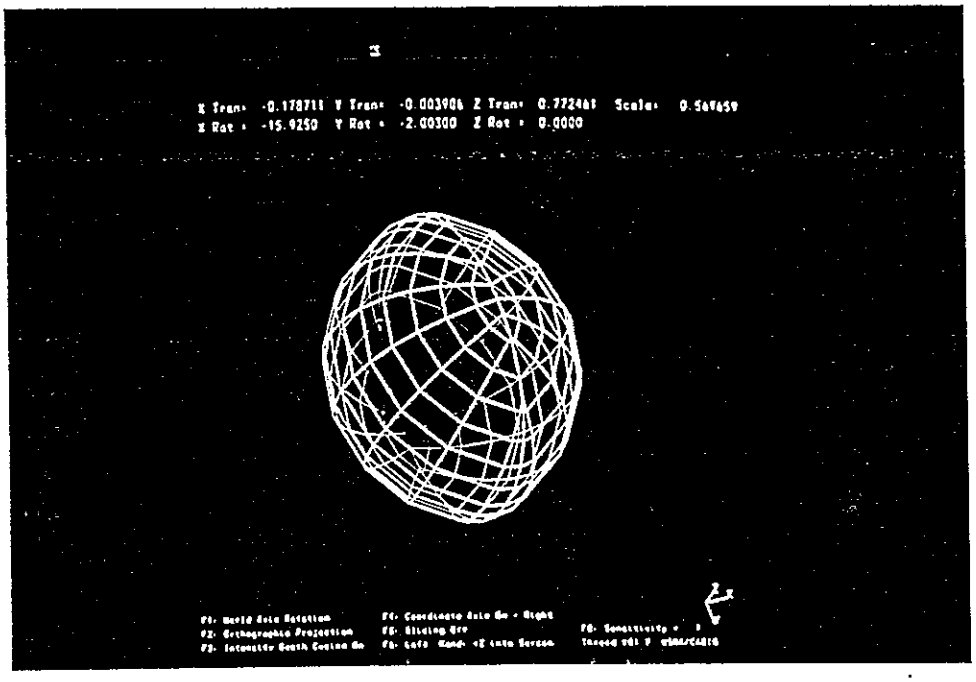


圖 16　兩個圓球體經由交集運算(INTERSECT Operation)後之結果，以 WireFrame 之架構顯示

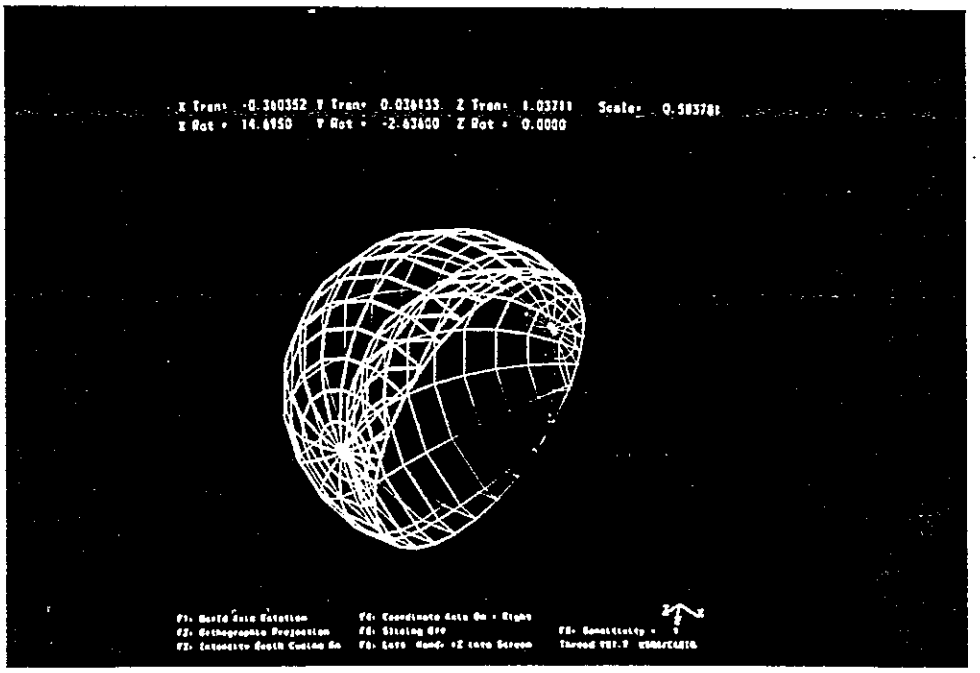


圖 17　兩個圓球體經由差集運算(DIFFERENCE Operation)後之結果，以 WireFrame 之架構顯示

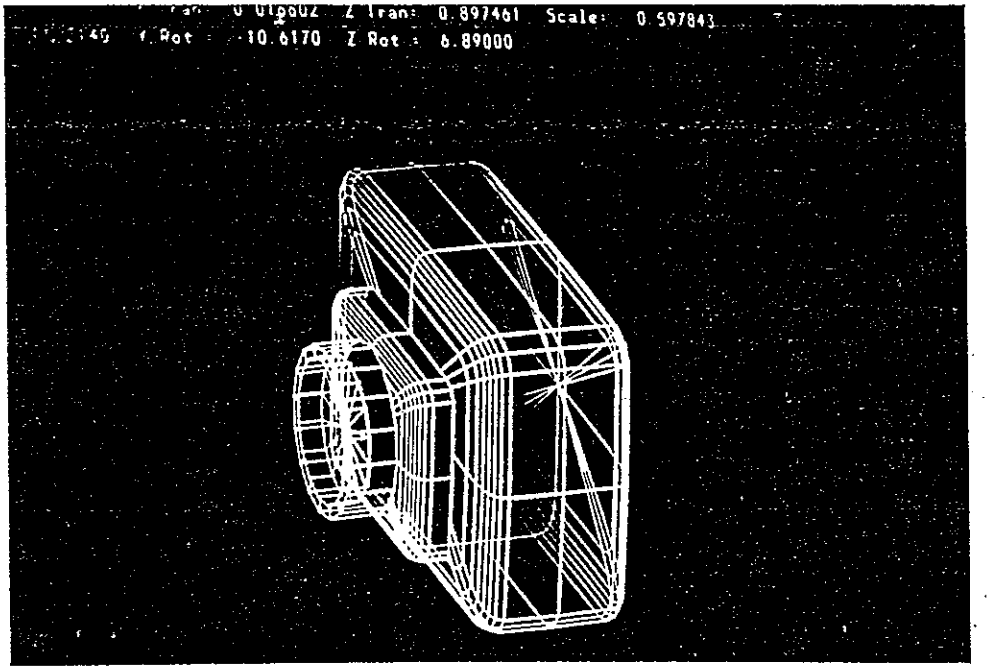


圖 18 由集合運算製作照相機，以 WireFrame 之架構顯示；其中包括三次聯集運算和一次差集運算

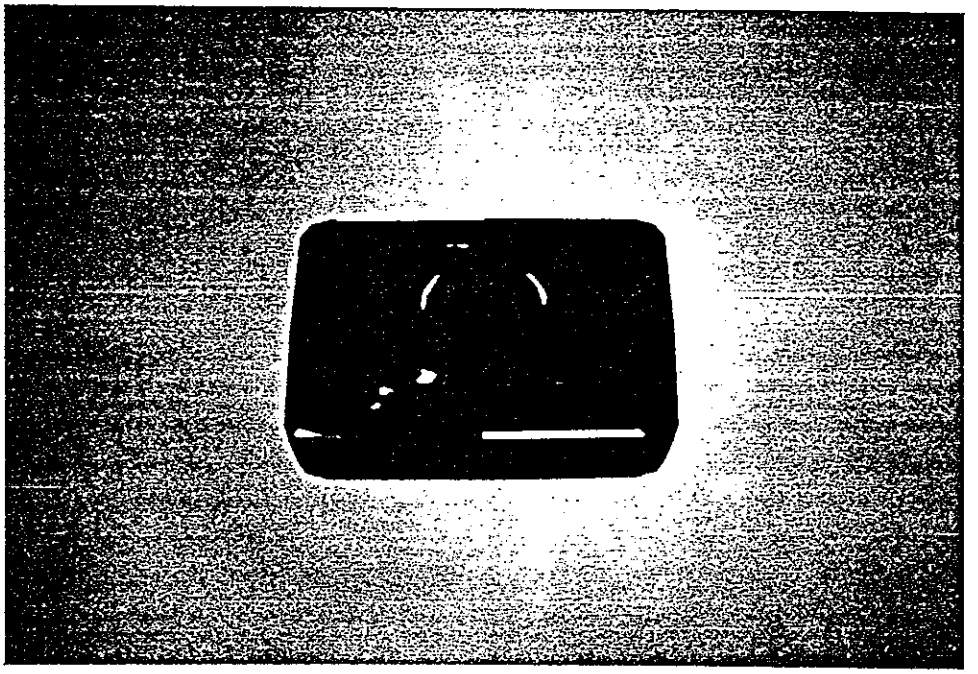


圖 19 由集合運算製作照相機，以 CADIG/ANIMATOR 加上陰影的處理

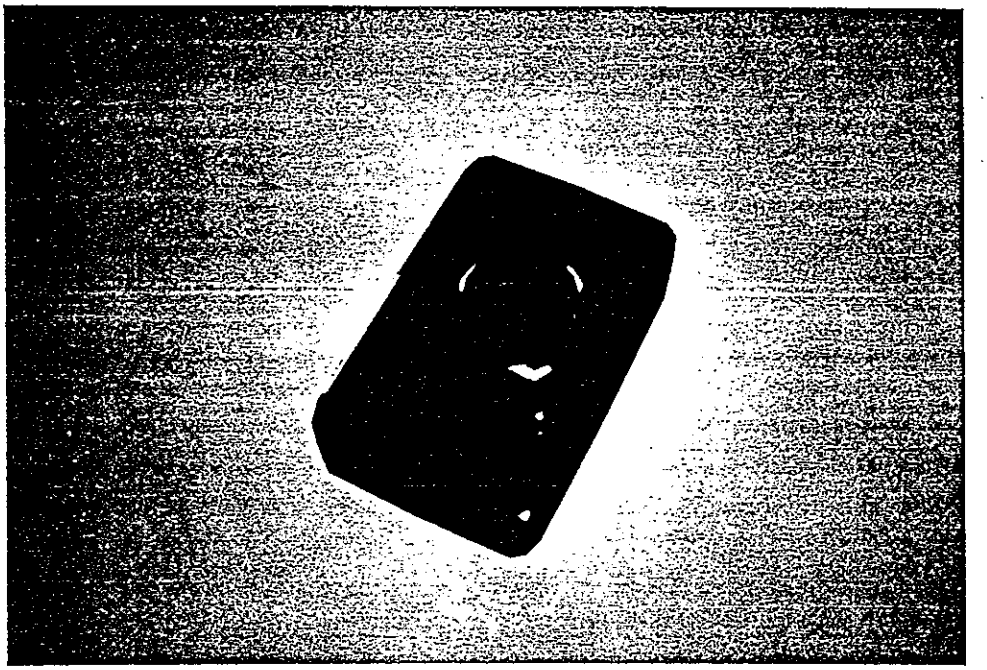


圖 20 由集合運算製作照相機，以 CADIG/ANIMATOR 加上陰影的處理，由另一個角度觀察

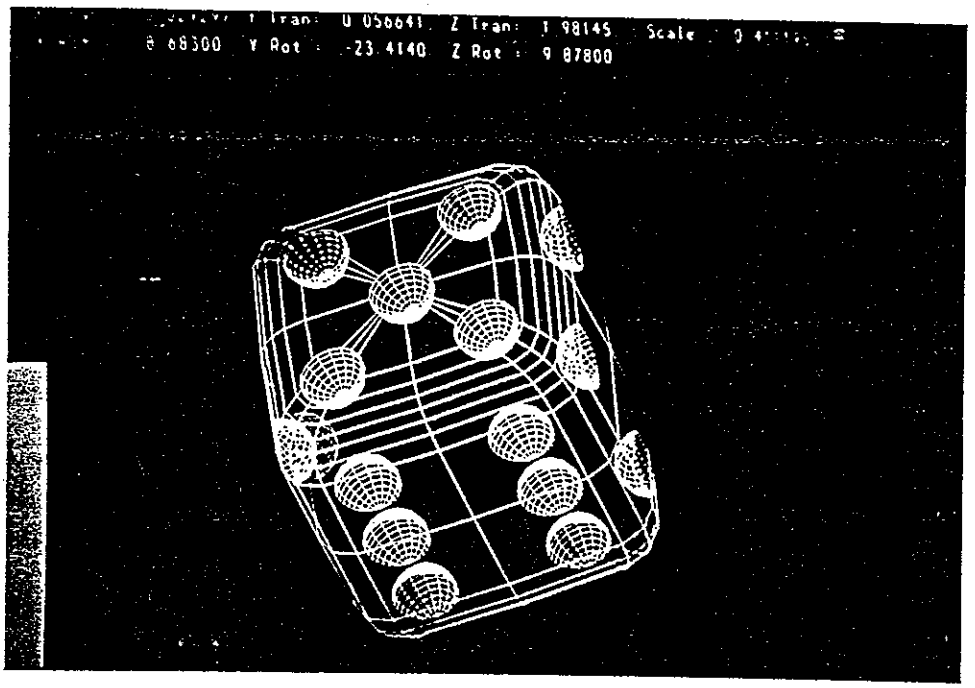


圖 21 由集合運算製作骰子，以 WireFrame 之架構顯示；其中包括 21 次差集運算

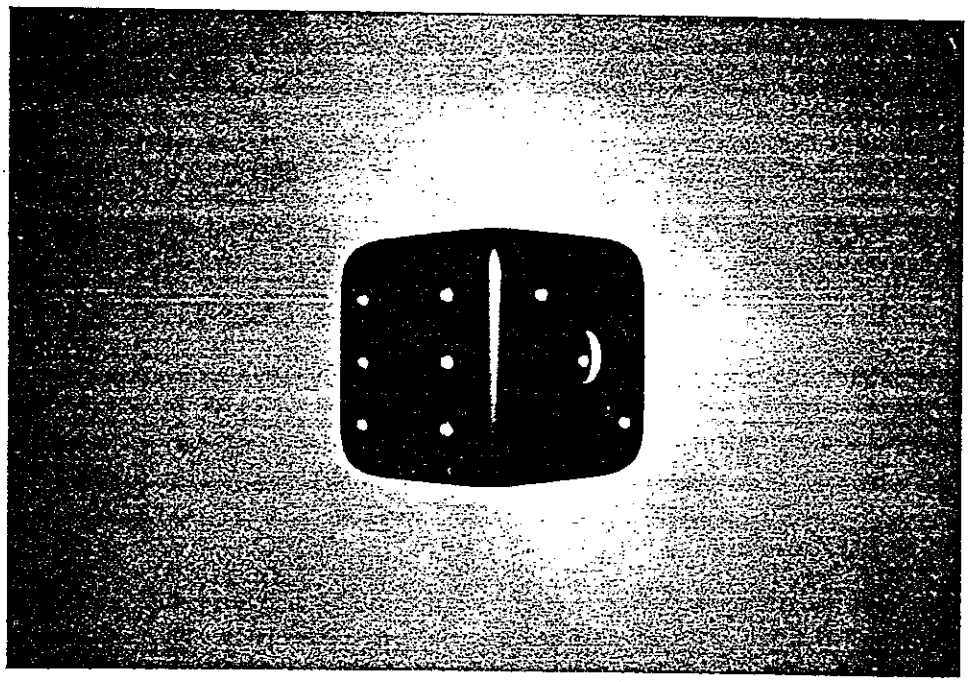


圖 22 由集合運算製作骰子，以
CADIG/ANIMATOR 加上陰影的處理

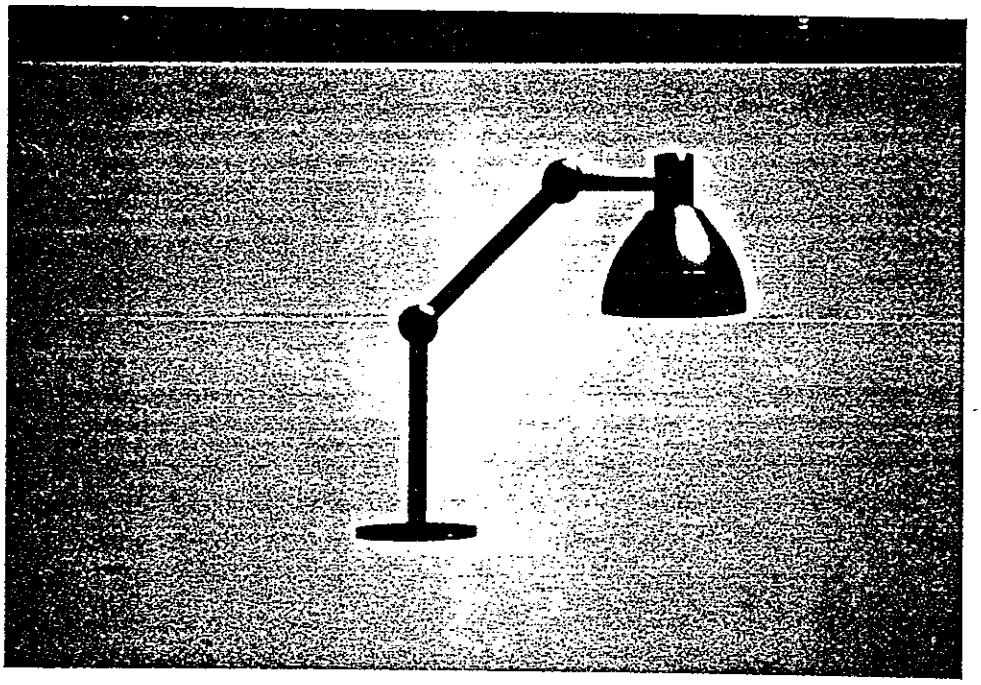


圖 24 由集合運算製作檯燈，以
CADIG/ANIMATOR 加上陰影的處理
，光源置於遠處

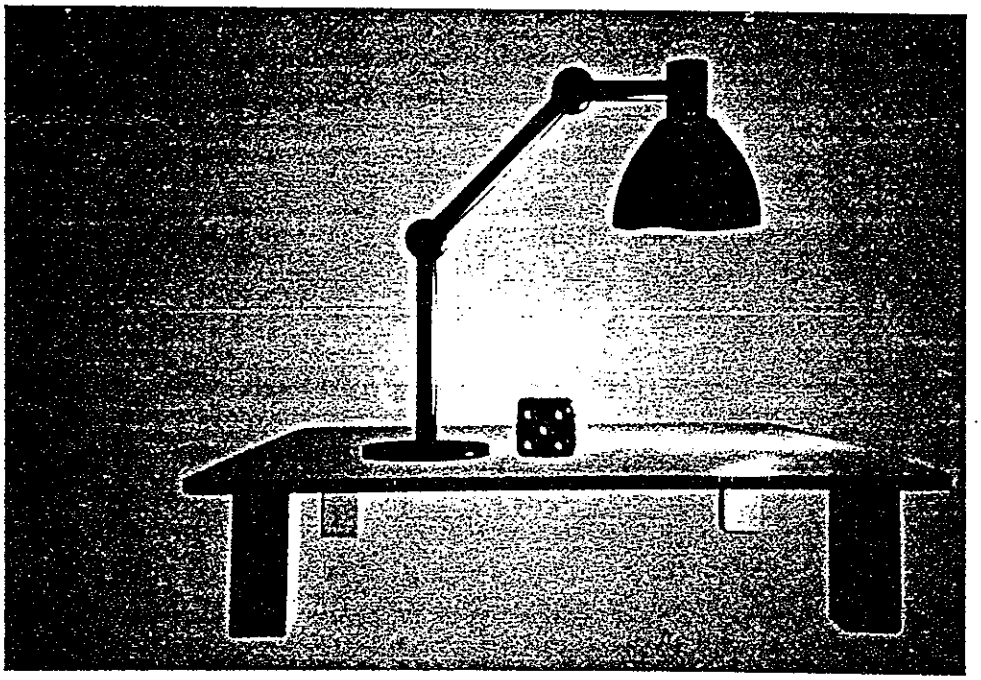


圖 25 將骰子、檯燈置於桌子上以 CADIG/ANIMATOR 加上陰影的處理，光源置於燈罩內

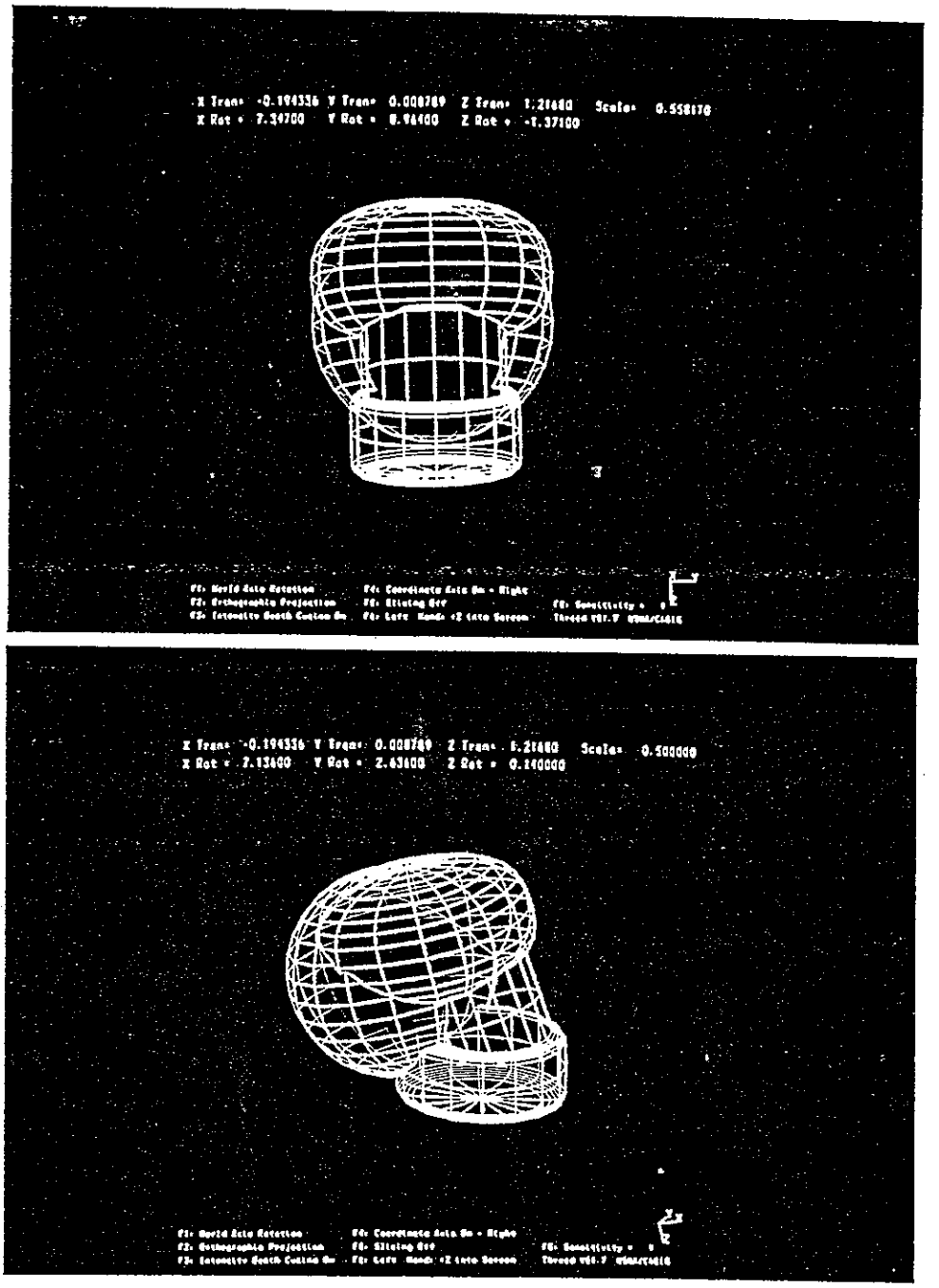


圖 26 由集合運算製作人頭形狀，
以 WireFrame 之架構顯示，
由兩個角度觀察

第五章 結論與未來研究方向

5.1 結 論

電腦繪圖最主要的目的是希望能造出真實逼真迷人的景像，而其中最基本的便是物體模型的製作，本論文所討論的就是提供一套立體物體模型製作過程的工具，讓使用者能夠以簡單直覺的方式來造出所想像的立體物體，以提供給後續的處理，如電腦動畫的製作，這也是我們最基本的目的之一。

在我們的演算法中是以直觀的方式來處理立體物體之間交集的問題，因爲立體物體是以 Vertex-Based Boundary Model 來表示，所以我們就以 Vertex 爲處理的對象，將兩個做集合運算的物體之 Vertex 予以分類，當然包括在處理過程中新產生出來的 Vertex ，然後再依集合運算的運算元之特性，將 Vertex 重新組合成所要的物體；這樣處理的方式有兩項優點：

1). 只要是以 Vertex-Based Boundary Model 表示的立體物體皆能處理。

2). 演算法的過程中只對真正有交集的 Polygon 做複雜的處理，對於其它的 Vertex 只做分類的處理，

因此可以加快處理的速度，節省處理的時間。

另外，本系統中是以 CSG 模型的方式做爲基本物體的輸入方法(亦即輸入物體的基本特性，如球心、半徑)，並非是以點輸入的方式，所以可以簡化輸入的過程，增加對使用者友善(User Friendly)的程度。

5.2 未來研究方向

在我們的研究過程中，首先遇到的問題是程式設計上的問題，當一個立體物體愈趨複雜時，Polygon 數目會愈來愈多，其所佔用的記憶體便會愈來愈多，若是將所有的 Polygon 皆置於主記憶體中，將會造成主記憶體不夠的情況，雖然這種處理方式是速度最快的方式，但是卻只能處理簡單的物體，也就是 Polygon 數目較少的物體，這也就是 Trade-Off 的問題，因此，爲了要達到能處理較複雜物體的目的，我們將物體的 Polygon 資料皆儲存在次記憶體中，只有在需要處理時才將之載入主記憶體中，雖然如此可以使能處理的 Polygon 數目大爲增加，但是同時得犧牲速度，使得處理時間增加，因此，處理速度的增快是我們未來的研究方向之一。

基本上來說，集合運算是立體幾何模型中的一項工具，希望造出甚麼樣的物體完全在於使用者的想法，因爲在

我們的研究環境中尚缺乏一個完善的使用者介面管理系統，所以，在設計物體時常會浪費許多時間，因此，結合立物體模型與使用者介面的系統也是我們未來的研究方向之一。

參考文獻 :

[1]. A.A.G. Requicha, "Representations for rigids solids: thery, methods and systems".Computing Surveys, Vol 12, n 4, december 1980.

[2]. A. Requicha and H. Voeclcker, "Solid Modeling: A Historical Summary and Contemporary Assessment". IEEE CG&A, Vol 2, n 2, 1982.

[3]. A. Requicha and H. Voeclcker, "Solid Modeling: Current status and research directions". IEEE CG&A, October 1983.

[4].A.Requicha and H.Voeclcker, "Boolean Operations in Solid Modeling:Boundary valuation and Merging algorithms",Proceedings of the IEEE, 73(1), Jan, 1985.

[5]. J.W. Boyse and J.E. Gilchrist, "GMSolid:Interactive Modeling for Design and Analysis of Solids". IEEE CG&A, March 1982.

[6]. Forrest, A. R., "Computational Geometry Achievements and Problems",in Computer Aided Geometric Design, Academic Press, New York (1974).

[7]. Levin, Joshua Z., "A Parametric Algorithm for Drawing Pictures of Solid Objects Composed of Quadric Surfaces", Comm. ACM Vol. 19(10) pp. 555-563 (October 1976).

[8]. W. E. Carson, "An Algorithm and Data Structure for 3D Object Synthesis Using Surface Patch Intersections",Computer Graphics, Vol 16, n 3, July 1982.

[9]. B. Baumgart, "Geometric Modelling for Computer Vision",PhD thesis, Standford University, 1974.

[10]. B. Baumgart, "A Polyhedra Representation for Computer Vision", In National Computer Conference, pp. 589-596, AFIPS Conf. Proc., 1975.

[11].A.H.Barr, "Superquadrics and Angel-Preserving Transformations", IEEE Computer Graphics and Applications, Vol.1, No.1, Jan. 1981, pp.11-23.

[12].W.R.Franklin, A.H.Barr, "Faster Caluation of Superquadric Shapes", IEEE Computer Graphics and Applications, Vol.1, No.1, July. 1981, pp.41-47.

[13].A.Kela, "Hierarchical Octree Approximations for Boundry Representation-Based Geometric Models", Computer-Aided Design, Vol.21, No.6, July/Aug. 1989, pp.355-362.

[14].I.Navazo, J.Fontdecaba and P. Brunet, "Extended Octtrees,between CSG Trees and Boundry Representations", EUROGRAPHICS'87, pp.239-247.

[15].I.Carlbom, "An Algorithm for Geometric Set Operations Using Cellular Subdivision Techniques ", IEEE CG&A, May. 1987, pp.44-55.

[16].Weiler,K., "Edge-Based Data Structure for Solid Modeling in Curved-Surface Environment", IEEE CG&A, Vol.5, No.1, 1985, pp.21-40.

[17].Magnenat-Thalmann,N., and D.Thalmann, "Image Synthesis : Theory and Practice", Springer-Verlag, Tokyo, 1987.

[18].David F. Rogers,"Procedural Elements for Computer Graphics", McGraw-Hill, 1985.

[19].Foley, James D. and Van Dam, Andries, "Fundamentals of Interactive Computer Graphics", Addison Wesley Publishing Company, 1982.

[20].D.Badouel and G. Hegron, "Set Operation Evaluation Using Boolean Octree", New Trends in Computer Graphics (Proc. CGI'88), ED. Springer-verlag, pp. 275-287, 1988.

[21].W.C. Thibault and B.F. Naylor, "Set Operations on Polyhedra Using Binary Space Partitioning trees", Computer Graphics, 21(4), July, 1987.

[22].M. Mantyla and R. Sulonen, "GWB:A Solid Modeler with Euler Operators", IEEE CG&A, Sep, 1982.

[23].M. Mantyla and M. Tamminen, "Localized Set Operations for Solid Modeling", ACM Computer Graphics, 17(3), July, 1983.

[24].T.C. Woo, "A Combinational Analysis of Boundary Data Structure Shemata", IEEE CG&A, 5(3), March , 1985.

[25].Michael E. Mortenson, "Geometric Modeling", John Wiley and Sons, Inc., 1985