

TR--88-024

供動畫使用之物體外形生成系統

參 考 書
不 外 借

中 央 研 究 院
資 訊 科 學 研 究 所
78.7.25
圖 書 室

中研院資訊所圖書室



3 0330 03 000107 2

0107

國立中興大學

碩士論文

TR-88-024

題目：供動畫使用之物體外形生成系統

(A Shape Generation System
for Animation Use)

所別：應用數學研究所

組別：資訊組

研究生：簡水源

指導教授：鄭國揚 博士

中華民國七十八年五月 日

目 錄

誌 謝	i
摘 要	ii
第 一 章	緒 論	1
第 二 章	B-軟楔曲線與曲面	4
2.1	B-軟楔曲線	4
2.2	二次B-軟楔曲線	11
2.3	B-軟楔曲面	14
2.4	透視投影	16
第 三 章	自由形狀物體設計	22
3.1	曲面旋轉	22
3.2	直線軸設計	23
3.3	自然軸設計	26
3.4	密合技術	29
第 四 章	物體導向化模組的建立	34
4.1	物體導向化程式	34
4.2	物體導向化圖形模組	37
第 五 章	動畫系統架構與應用範例	40
5.1	動畫系統架構	40
5.2	應用範例	44
第 六 章	結 論	48
參 考 文 獻	50

誌 謝

誠摯地感謝我的指導教授鄭國揚教授兩年來的悉心指導，不但使得本篇論文得以順利完成，更奠定了我在電腦圖學與電腦視覺等知識的基礎，而確立我學習的方向。鄭老師平日言談風趣，待人親切，研究學問積極嚴謹，讓我受益匪淺，是我日後治學為人的榜樣。

感謝所裡各位師長在電腦與數學知識上傾其所有的教誨，及言行生活的指導與關懷，亦感謝所上提供良好的設備與研究環境，使我擁有充實的學習環境。同時，感謝中央研究院資訊所圖書館人員，在我查詢資料時給予最大的幫忙。

其次，感謝同學們的支持與切磋，以及家人、朋友的關愛與鼓勵。

最後，謹以此文獻給我最敬愛的父母。

摘要

電腦動畫的產生是由一些畫面，在一段時間內藉著路徑的調整來生成連續畫面。立體形狀表現在動畫上是利用結構化架構產生，藉由物體部位的參數傳遞，可造成局部性、整體性或混合式的立體動畫效果。

本系統 (OSGS) 所產生的三維物體是利用 *B*-軟楔曲面 (*B-spline surface*) 來產生線框式立體外形。*B*-軟楔曲面具有局部控制的特性，可藉由控制點的調整產生平滑的曲面。由堆積法原理 (*Lofting Theorem*) 可產生三類基本曲面：(一). 曲面旋轉：對固定軸旋轉掃描產生對稱性物體外形。(二). 直線軸設計：由橫剖面與縱剖面產生直立狀物體外形。(三). 自然軸設計：由橫剖面與脊椎面產生管狀物體外形。由此三類基本曲面可以形成階層式三維物體的部位架構。

本文提出一項密合技術 (*Glue Method*)，將兩個相關的部位以動畫的方式結合成一接合體，當所有相關的部位分別作密合的動作之後，即生成完整的三維物體外形，再用動畫路徑控制系統 (*APCS*) 調整路徑中的變數，即可產生立體圖形的動畫。

第一章 緒論

傳統動畫是由畫家將每一張畫面逐步繪製而成，其中大部份的工作都花在修改一些小動作的變化。電腦動畫 (Computer Animation) 是讓畫家在物體生成系統上畫出一些明顯改變的主畫面，在連續兩張主畫面間調整運動路徑的變化，並傳回其間各畫面的控制點，最後再由物體生成系統自動產生中間畫面 (Inbetweens)，即可造出生動逼真的動畫影像 [4][9]。本文的目的即在建立一套立體物體生成系統 (OSGS) 配合 [17] 的動畫路徑控制系統 (APCS) 產生立體圖形的動畫效果。

在製作本系統前，我們須先決定物體生成的方式，通常在電腦上表現物體外形的方式有三類：(1) 線框表示法 (Wire-frame Representation)，(2) 表面表示法 (Surface Representation)，(3) 立體表示法 (Volume Representation)，[13] 其中線框表示法是最基礎也是最簡單表示法，它是利用數學模式的曲線來圍成自由形狀 (Free-formed Shape) 物體的外形，為了方便使用者在螢幕上調整物體的外形，本文採用的數學曲線模式是 B- 軟楔曲線 [5]，B- 軟楔曲線是畫曲線的理想工具之一，它是藉由物體輪廓上的少數控制點來繪出平面的物體外形，並且可以調整控制點的位置產生局部變化的圖形，這個局部控制 (Local Controlling) 的性質可以在動畫中表現出特殊效果。

爲了表現立體效果，我們將B-軟楔曲線擴充到B-軟楔曲面，B-軟楔曲面是由兩個B-軟楔曲線的幾何內積形成的，它由兩個方向的控制點產生曲面上的片面 [1] [2]，在建造立體圖形時，我們通常利用堆積法原理 (Lofting Theorem) [8] 將一個封閉的B-軟楔曲線用來描述物體的橫剖面 and 另一個描述物體正面外形的開放式的B-軟楔曲線稱爲縱剖面，由橫剖面和縱剖面產生一般的直立形的物體，或是由橫剖面和彎曲狀的脊推線產生管狀的物體。另外，亦可藉由縱剖面對固定軸旋轉來產生對稱性物體。由以上三種方式，我們可以產生簡單的自由形狀物體，在產生複雜的物體時，我們提出一項密合技術 (Glue Method) 將兩相關物體作密接的動作，將貼合物投影到被貼合體的曲面上，並且利用運動路徑的控制產生畫家滿意的密合體，由物體導向化的觀念在所有相關物體作密合的動作，即可生成完整的物體外形。利用物體導向化的觀念，可藉由部位間的資料傳遞來建立結構化之物體架構，亦可由部位屬性的繼承性來產生局部性、整體性或混合式的動畫方式。

本文共分六章：除第一章緒論外，第二章是以數學模式介紹B-軟楔曲線及B-軟楔曲面並說明其物理意義，和三維物體的透視投影，第三章設計藉由B-軟楔曲面所生成的三種基本曲面並提出兩物體間的密合技術，第四章說明物體導向化程式的特性及物體導向化的圖形模組，第五章將分析本系統之架構並依此架構設計一些靜物與動畫的例證

，第六章將說明本系統的優缺點和未來的發展方向，以作為本篇論文的結論。

第二章 B-軟楔曲線與曲面

我們在日常生活中所見的物體，除了少數是由長方體、球體等規則形狀的元件組合而成之外，大多數的自由形狀 (*free-formed shape*) 物體都是由不規則的曲線所形成。一般而言，曲線的種類概分為兩大類，第一類是內插法曲線 (*Interpolation Curve*): 如 *Hermite* 曲線，此類曲線必會通過每個輸入點。第二類是近似法曲線 (*Approximation Curve*): 如 *Bezier* 曲線、*B-軟楔曲線* 和 *B-軟楔曲線* 等。此類曲線並未通過每個輸入點，而是接近輸入點的位置，通常由近似法來產生曲線較優於內插法，因為近似法曲線的形狀可以藉著較少的輸入點作局部性的調整，仍能保持曲線的平滑。本章乃是探討三維物體生成的數學方法，本文選用二次 *B-軟楔曲面* 來生成立體外形，在第一節中將介紹 *B-軟楔曲線* 的數學原理及其特性，第二節中將說明二次 *B-軟楔曲線* 的產生方式及在繪圖時的技巧，第三節中敘述利用 *u* 方向和 *v* 方向的幾何內積 (*tensor product*) 產生 *B-軟楔曲面*，以建立三維物體的輪廓，第四節中說明三維物體轉換到平面上的透視投影公式。

2.1 B-軟楔曲線

B-軟楔曲線 是由 $\vec{p}_0, \vec{p}_1, \dots, \vec{p}_n$ n 個點連成的控制多邊形 (*control polygon*) 所圍成，其定義是：

$$P(u) = \sum_{k=0}^n p_k N_{k,m}(u) \quad (2.1)$$

其中 p_k 表示第 k 個控制點 (control point),

$N_{k,m}(u)$ 為 m 階 B -軟楔的基底函數 (basis function)

此基底函數具有下列的性質：

1. 是 $m-1$ 次方的多項式，且與 n 無關。
2. 在 u 的範圍內恆不為 0，即曲線上的所有點不會落在任一控制點之上，故能提供局部控制曲線的便利性。
3. 將 u 分成 $n+m+1$ 個端點，每個端點 u_j 稱為節點 (knot-point)。
4. 每個 $N_{k,m}(u)$ 是由 m 個子區間所決定，即每次有 m 個子區間會影響 $p(u)$ 的形狀。
5. $N_{k,m}(u)$ 具有遞迴性質：

$$N_{k,1}(u) = \begin{cases} 1 & \text{若 } u_k \leq u \leq u_{k+1} \\ 0 & \text{其他} \end{cases}$$

$$N_{k,m}(u) = \frac{u-u_k}{u_{k+m-1}-u_k} N_{k,m-1}(u) + \frac{u_{k+m}-u}{u_{k+m}-u_{k+1}} N_{k+1,m-1}(u)$$

$$6. \sum N_{k,m}(u) = 1.$$

由以上基底函數的性質，我們知到 B -軟楔曲線是由控制點的次序逐段生成的，每段都是由部份的控制點所影響，因此當調整其中的一個控制點時，並不會造成整個曲線的改變，僅改變由此控制點所產生的曲線段，而且 B -軟楔

曲線不會通過控制點，提供我們作局部控制的方便性。圖 2.1 即是一條 B -軟楔曲線。

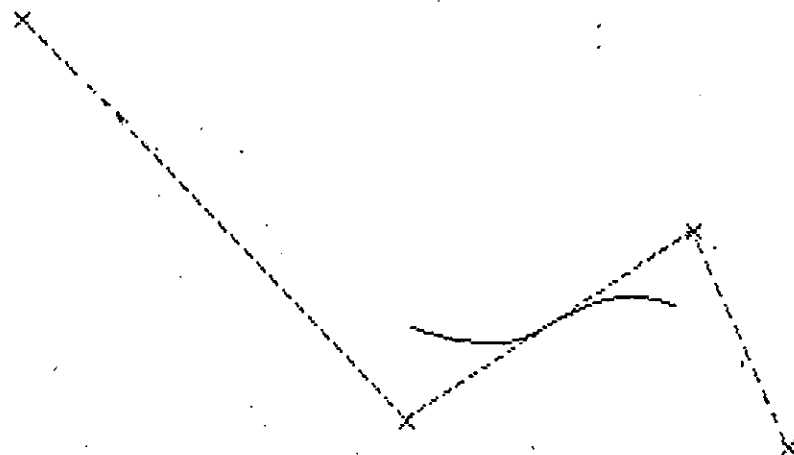


圖 2.1 B -軟楔曲線

在畫 B -軟楔曲線時，只要給定控制點 p_0, p_1, \dots, p_n ，在給定階數 m (通常 $m = 3$ 或 4) 的條件下，即可根據基底函數的遞迴運算，在 p_0, p_1, \dots, p_n 的連線之內產生一條 B -軟楔曲線，所以當我們利用 B -軟楔曲線來畫物體輪廓時，可由下列三個步驟來完成：

- (1) 先將心中所想的圖形，畫出草圖。
- (2) 在草圖上選定一些關鍵點，如轉折處或突起點附近作為 B -軟楔曲線的控制點，由控制點的逐點輸入來產生 B -軟楔曲線。
- (3) 比較 B -軟楔曲線所生成的圖形與原來構想的圖形，若不甚理想，可利用局部控制的方式調整或增減控制點與節點數，達到滿意的圖形。

圖 2.2 即是利用前述步驟所作的鳥的輪廓。



圖 2.2

使用 B -軟楔曲線作圖形，會遭遇到作封密曲線的情況，即封密式的 B -軟楔曲線，是利用重覆點 (*Multiple Knots*) 將控制的首點當作終點，即產生曲線的終止點與起始點相接的封密曲線，當有 n 點重疊時，其基底函數降低為

$$N_{i,k}(u) = N_{0,k}((u-i+n-1) \bmod (n+1))$$

若一個二次 B -軟楔曲線，由基底函數的性質知其每一段曲線是由三個控制點形成，因此，如圖 2.3 所成的封密式 B -軟楔是由 $P_0(u), P_1(u), P_2(u)$ 及 $P_3(u)$ 四段曲線圍成，其控制點依序為 $P_0, P_1, P_2, P_3, P_4(=P_0), P_5(=P_1)$ 。

開放式的 B -軟楔曲線，即一般非封密曲線即是由基底函數的階數決定其形狀，通常取三段或四段較接近理想圖形，其中各區段節點數可由下列定義之：

$$u_i = \begin{cases} 0 & i < k \\ i-k+1 & k \leq i \leq n \\ n-k+2 & i > n \end{cases}$$

圖 2.4 即由相同的控制點、不同的階數，所繪的開放式 B-軟楔曲線。

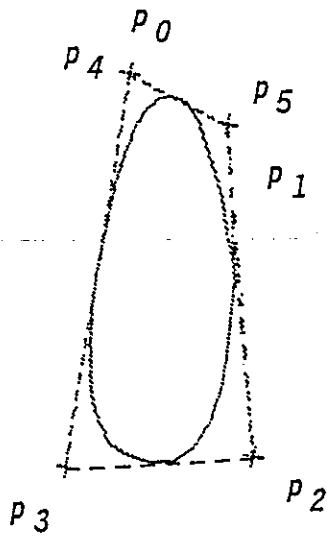


圖 2.3 封閉式 B-軟楔曲線

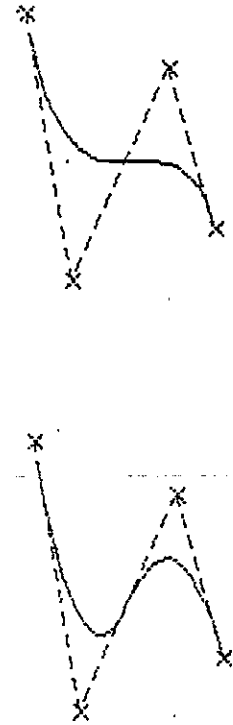


圖 2.4 開放式 B-軟楔曲線

使用 B-軟楔曲線作圖時，我們大多使用二次(三階)或三次(四階) B-軟楔是因為高次 B-軟楔除了運算複雜化之外，其產生的曲線會出現不平滑的振盪現象，這種性質稱為 *Variation Diminishing*，另外由於 B-軟楔曲線是由控制點乘上介於 0 到 1 的基底函數值，曲線的位置是由控制點的加權值所決定，因此 B-軟楔曲線必落在 P_0, P_1, \dots, P_n 所形成的控制多邊形之內，此即 *Convex Hull* 的性質。故以 B-軟楔來繪圖時有兩個好處，除了可以讓我們利用控制面來

預測曲線的形狀以外，另一項好處是由於B-軟楔曲線是一種參數式，其位置與 X, Y 座標軸無關，所以若要將曲線平移或旋轉時，不須更改參數，只要將此曲線的控制點作平移或旋轉即可。

前面我們說過，使用B-軟楔曲線修改圖形時經常要增加節點，其最普遍的方法是利用 *Oslo Algorithm* [8,14]，*Oslo algorithm* 除了增加節點產生更完美的圖形之外，更可將一個封閉的B-軟楔曲線分割成兩個開放式的曲線，其演譯法如下：

Oslo Algorithm

輸入

N = 表示 N 個區段，即 $N+1$ 個控制點。

$P = (p(0), \dots, p(N))$ 表示 $N+1$ 個控制點向量。

K = B-軟楔曲線的階數。

$TAU = (TAU(0), \dots, TAU(N+K))$ 表示原曲線的節點向量。

$T = (T(0), \dots, T(Q))$ 表示新曲線的節點向量， $Q \geq N+K$ 。

輸出

$D = (D(0), \dots, D(Q-K))$ 表示分割控制多邊形的控制點向量

```
procedure loop(K,N,Q,P,TAU,T,D)
begin for j ← 0 to (Q-K) do
    call find(KN,TAU,T,j,MU)
    call SUBDIV(P,K,TAU,T,K,MU,j,D(j))
end
end
```

其中，函數 $\text{find}(\text{KN}, \text{TAU}, \text{T}, \text{j}, \text{MU})$ 是求一數 MU 使得 $\text{TAU}(\text{MU}) \leq \text{T}(\text{j}) < \text{TAU}(\text{MU}+1)$

```
procedure find(KN,TAU,T,j,MU)
  begin for i←0 to (KN-1)
    if (T(j) ≥ TAU(i)) then MU←i
  end
```

而且

```
procedure SUBDIV(P,K,TAU,T,RP1,I,J,PP)
  begin
    r←RP1-1
    if (r > 0) then
      begin
        PP2←0
        PP1←0
        P1←T(J+K-r) - TAU(I)
        P2←TAU(I+K-r) - T(J+K-r)
        if (P1<>0) call SUBDIV(P,K,TAU,T,r,I,J,PP1)
        if (P2<>0) call SUBDIV(P,K,TAU,T,r,I-1,J,PP2)
        PP←(P1*PP1+P2*PP2)/(P1+P2)
      end
    else PP←P(I)
  end
```

2.2 二次 B-軟楔曲線

如前節所述，一般使用 B-軟楔是用二次或三次，因為高次 B-軟楔的圖形會產生振盪曲折的現象，低次圖形能保持連續性及平滑性因為本系統是在 PC 上執行，考慮記憶體容量及執行速度等因素，故選擇二次 B-軟楔。

首先由二次 B-軟楔的基底函數推導均勻二次 B-軟楔曲線的公式。從第一節中基底函數的性質知

$$N_{i,2}(u) = \frac{(u-u_i)^2}{(u_{i+2}-u_i)(u_{i+1}-u_i)} \quad \text{當 } u_i \leq u \leq u_{i+1}$$

$$N_{i,2}(u) = \frac{(u-u_i)(u_{i+2}-u)}{(u_{i+2}-u_i)(u_{i+2}-u_{i+1})} + \frac{(u_{i+3}-u)(u-u_{i+1})}{(u_{i+3}-u_{i+1})(u_{i+3}-u_{i+1})} \quad \text{當 } u_{i+1} \leq u \leq u_{i+2}$$

$$N_{i,2}(u) = \frac{(u_{i+3}-u)^2}{(u_{i+3}-u_{i+1})(u_{i+3}-u_{i+2})} \quad \text{當 } u_{i+2} \leq u \leq u_{i+3}$$

(2.2)

令每一段長度為 1, $u_i = i1$, 以及 $t = (u - u_i) / 1$,

則將式 (2.2) 代入式 (2.1) 得到矩陣表示式

$$P(t) = [t^2, t, 1] 1/2 \begin{bmatrix} 1 & -2 & 1 \\ -2 & 2 & 0 \\ 1 & 1 & 0 \end{bmatrix} \begin{bmatrix} v_i \\ v_{i+1} \\ v_{i+2} \end{bmatrix} \quad (2.3)$$

其中 $t = FR(kv) = kv$ 的分數部份

$i = INT(kv) = kv$ 的整數部份

$k =$ 總共的軟楔段 $= (n+1) - m$ 且 $m=2$

$t \in [0, 1]$

因此，每一段軟楔都是由三個相連的控制點所定義，一個二次 B -軟楔曲線則是一些軟楔段相連接而成且接點均為 C^1 連續。圖 2.5 是一些二次 B -軟楔的例子。

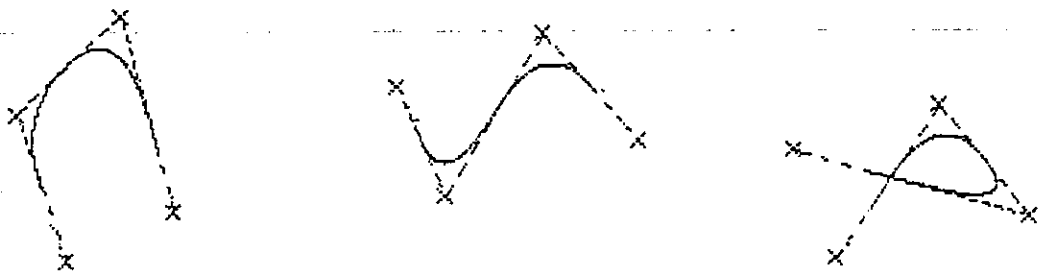


圖 2.5 二次 B -軟楔曲線

由上式我們可以推導出均勻二次 B -軟楔的幾何特性，如圖 2.6 給定的三個控制點 v_i, v_{i+1}, v_{i+2} 所構成的三角形，當 $t=0$ ，代入式 (2.3)，得到

$$p(0) = 1/2(v_i + v_{i+1})$$

表示軟楔的起始位置是在 v_i 和 v_{i+1} 兩個控制點連線的中點，在此點的一次微分值

$$p'(0) = v_{i+1} - v_i$$

表示二次 B -軟楔曲線的方向與 v_i, v_{i+1} 所成直線平行同理，當 $t=1$ 時，

$$p(1) = 1/2(v_{i+1} + v_{i+2})$$

$$p'(1) = v_{i+2} - v_{i+1}$$

即此段軟楔的終點為 v_{i+1} 與 v_{i+2} 的中點，而且其路徑與 $\overrightarrow{v_{i+1}v_{i+2}}$ 同方向。

由以上的討論，我們可由三個控制點形成的多邊形窺知其所形成的均勻二次 B -軟楔。

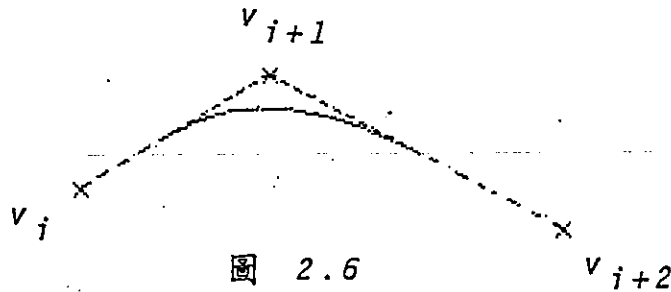


圖 2.6

在一般的情形下，二次 B -軟楔曲線各點都是 C^1 連續的，但是在退化情形下，則不保持 C^1 連續。如圖 2.7(左)之三個控制點共線或(中)二個控制點共點皆可形成直線，(右)二個控制點共點可形成尖角(cusp)，利用此特性我們可以畫出不平滑的曲線部份。

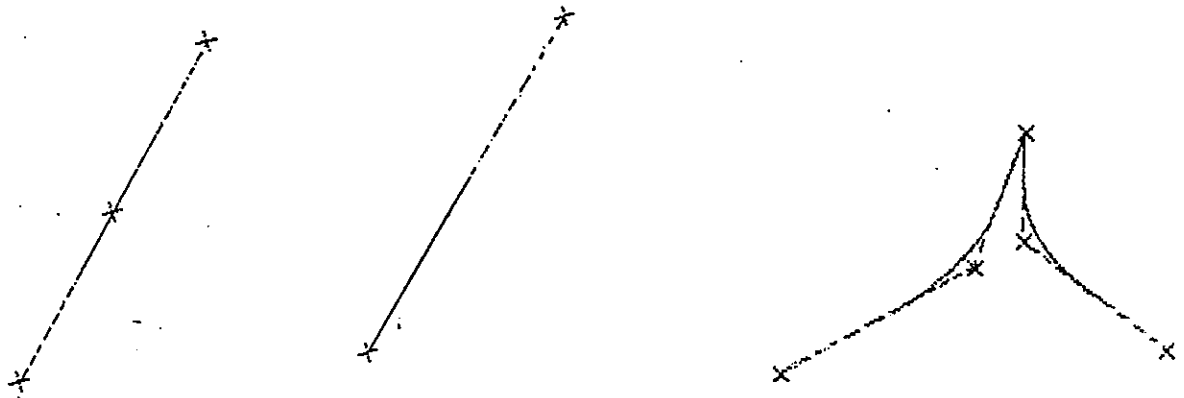


圖 2.7

由前所述，我們可以從控制點的輸入造出各式各樣的圖形。

2.3 B-軟楔曲面

在畫三維物體輪廓時，我們使用B-軟楔曲面產生物體表面的輪廓，其定義為

$$P(u, v) = \sum_{i=0}^n \sum_{j=0}^m p_{i,j} N_{i,k}(u) N_{j,l}(v) \quad (2.4)$$

$$u \in [u_k, u_{m+1}], v \in [v_l, v_{n+1}]$$

B-軟楔曲面是由 $\{p_{i,j}, i=0, \dots, n, j=0, \dots, m\}$ 個控制點圍成的四面網 (rectangle mesh) 或稱片面 (patch) 所產生，其性質與B-軟楔曲線相似，俱有局部控制的特性。當 u 方向或 v 方向的控制點改變時，B-軟楔曲面會隨之變形，形成凸起或下凹的形狀，非常適合動畫裡經常出現的臉部變形等的誇張效果的表現。

由B軟楔產生曲面時，我們先固定參數 u 或 v 為定數先由此定數產生一個同構曲面 (isoparametric curve) [1]，令 $u = u$ 為一定數，先產生 u 方向的B-軟楔曲線稱為 u -曲線 (u -curve)

$$W_j(u) = \sum_{i=0}^n p_{i,j} N_i(u) \quad j=0, \dots, m \quad (2.5)$$

利用 u -曲線再產生 v 方向的曲線，即形成B-軟楔曲面

$$\begin{aligned}
 P(u, v) &= \sum_{i=0}^n \sum_{j=0}^m p_{i,j} N_i(u) N_j(v) \\
 &= \sum_{j=0}^m \left[\sum_{i=0}^n p_{i,j} N_i(u) \right] N_j(v) \\
 &= \sum_{j=0}^m W_j(u) N_j(v) \quad (2.6)
 \end{aligned}$$

圖 2.8 即是二次 B-軟楔曲面的一例。

在應用上我們將 v -曲線方向表為橫剖面 (cross section)，將 u -曲線表為縱剖面 (profile)，橫剖面與縱剖面可以決定物體的形狀，本文將在第三章中詳細敘述以二次 B-軟楔曲面來建立立體物體外形的方式。

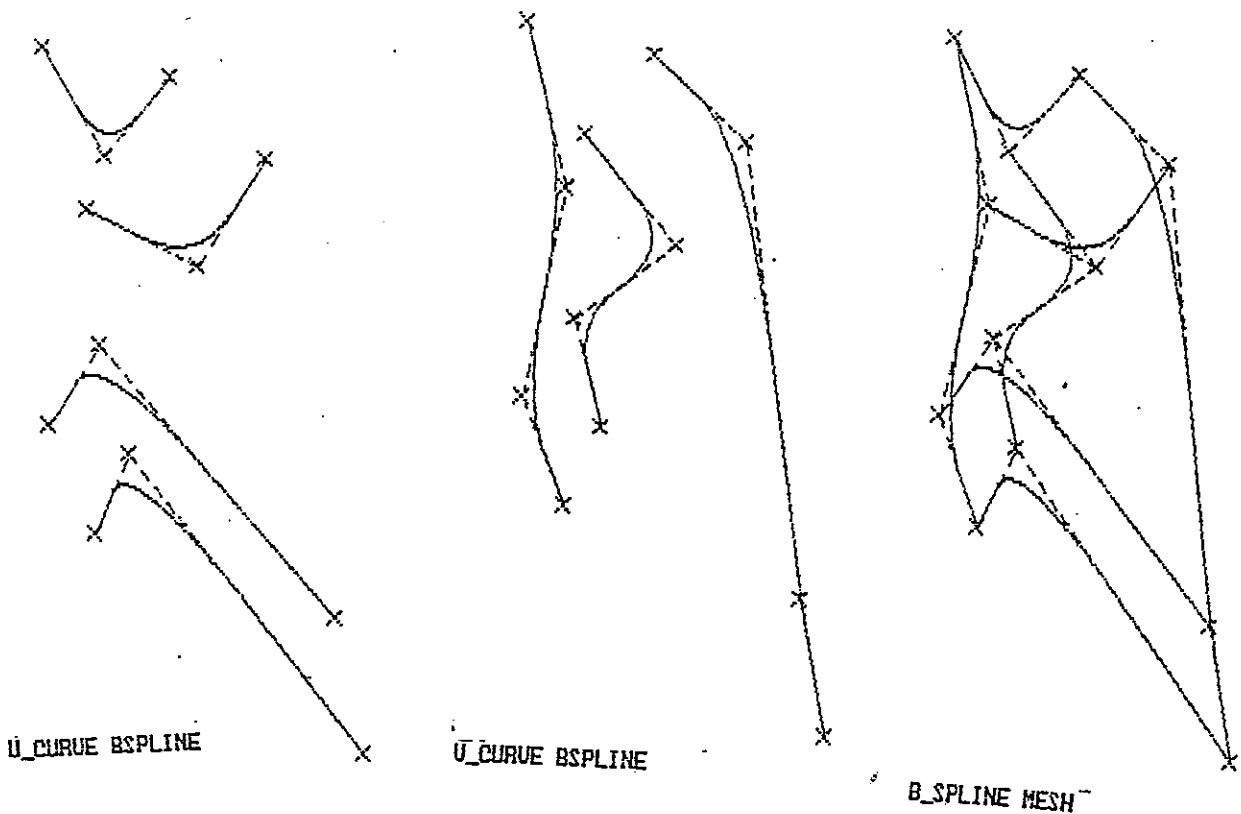


圖 2.8 B-軟楔曲面

2.4 透視投影

三維物體表現在螢幕上是以平面的方式呈現的，其生成的形狀端賴眼睛所在的位置而定。透視投影 (*Perspective projection*) 即是利用觀視者的位置看立體物體的投影方式。

推導透視投影之前，須注意三維的物體在自然界中是以右手座標系來描述，而其在觀視者的眼裡則是以右手座標系表現出來。通常，描述三度空間的位置可以球面座標系統表示，在此系統中，點 P 是用 (ρ, θ, ϕ) 表示，如圖

2.9。 ρ, θ, ϕ 的意義如下：

ρ : P 和原點的距離，

θ : 由正 Z 軸方向看去， OP 在 $X-Y$ 平面上之投影和正 X 軸由逆時針方向量出之角度。

ϕ : OP 和正 Z 軸之夾角。

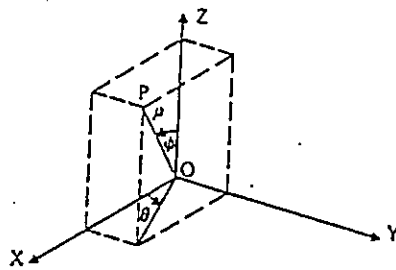


圖 2.9

直角座標系統中的點 (X, Y, Z) ，和球面座標系統中的點 (ρ, θ, ϕ) 的關係如下：

$$X = \rho \sin \phi \cos \theta$$

$$Y = \rho \sin \phi \sin \theta$$

$$Z = \rho \cos \phi$$

因爲螢光幕是二度空間，所以若想在螢光幕上看到物體，就必須先將原來三度空間的物體 (X, Y, Z) ，投影到二度空間的螢光幕上 (Sx, Sy) 。如圖 2.10 其中 P 是觀視者眼睛的位置。因爲在 P 點還存在一個眼睛座標系統 (Xe, Ye, Ze) ，所以在做投影轉換時，我們是先將 (X, Y, Z) 轉換到 (Xe, Ye, Ze) ，在將 (Xe, Ye, Ze) 轉換成 (Sx, Sy) 。現在將各步驟簡述如下：

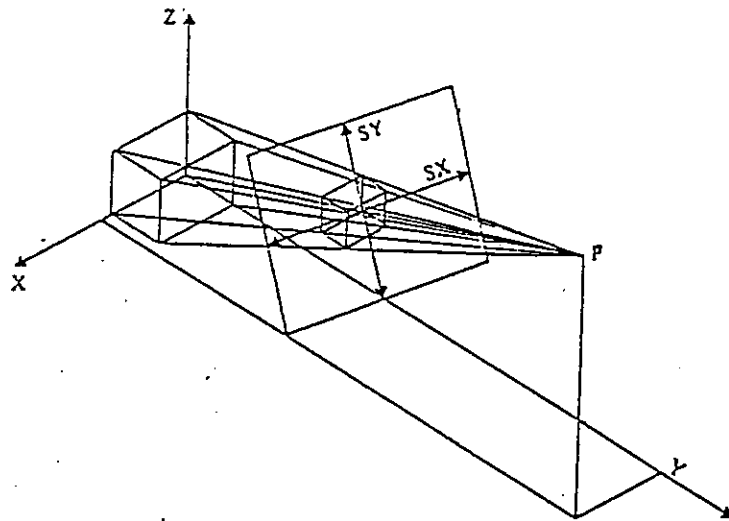


圖 2.10

步驟一：將原點平移至觀視點 $P = (X, Y, Z) = (\rho \sin \phi \cos \theta, \rho \sin \phi \sin \theta, \rho \cos \phi)$ ，如圖 2.11，轉換矩陣爲：

$$A = \begin{bmatrix} 1 & 0 & 0 & \rho \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ -\rho \sin\phi \cos\theta & -\rho \sin\phi \sin\theta & -\rho \cos\phi & 1 \end{bmatrix}$$

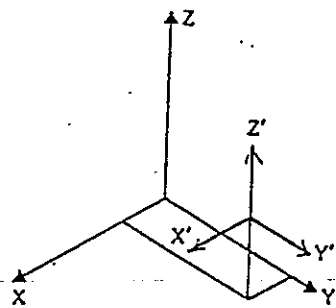


圖 2.11

步驟二：將座標系統對 Z' 軸順時針轉 $90-\theta$ 度，使 Y' 軸和 Z 軸相交。如圖 2.12 轉換矩陣為：

$$B = \begin{bmatrix} \cos(90-\theta) & \sin(90-\theta) & 0 & 0 \\ -\sin(90-\theta) & \cos(90-\theta) & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} \sin\theta & \cos\theta & 0 & 0 \\ -\cos\theta & \sin\theta & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

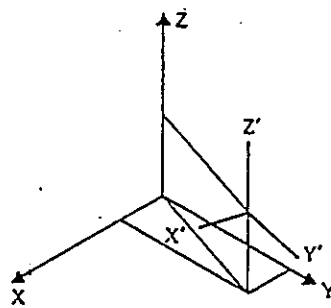


圖 2.12

步驟三：將座標系統對 X' 軸逆時針轉 $180-\phi$ 度（即將點依逆時針轉 $360-(180-\phi)=180+\phi$ 度），使 Z' 軸指向原座標之原點。如圖 2.13。轉換矩陣為：

$$C = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos(180+\phi) & \sin(180+\phi) & 0 \\ 0 & -\sin(180+\phi) & \cos(180+\phi) & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & -\cos\phi & -\sin\phi & 0 \\ 0 & \sin\phi & -\cos\phi & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

步驟四：倒轉 X' 軸，變成左手座標系統。如圖 2.14。轉換矩陣為：

$$D = \begin{bmatrix} -1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

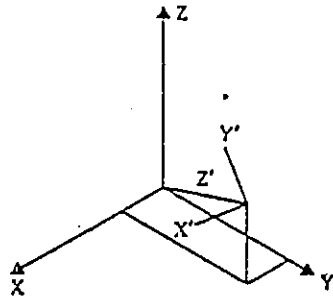


圖 2.13

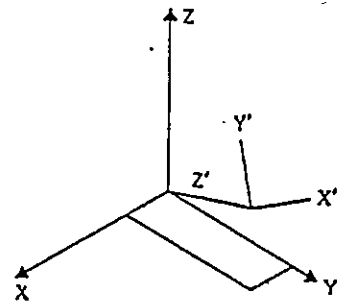


圖 2.14

經過步驟一至步驟四，就已經將原座標 (X, Y, Z) 轉換到眼睛座標 (X_e, Y_e, Z_e) ，四個轉換矩陣可以合成一個轉換矩陣：

$$T = A B C D$$

$$= \begin{bmatrix} -\sin\theta & -\cos\theta \cos\phi & -\cos\theta \sin\phi & 0 \\ \cos\theta & -\sin\theta \cos\phi & -\sin\theta \sin\phi & 0 \\ 0 & \sin\phi & -\cos\phi & 0 \\ 0 & 0 & \rho & 1 \end{bmatrix}$$

所以 $(X_e, Y_e, Z_e, 1) = (X, Y, Z, 1)T$ ，即：

$$X_e = -X \sin\theta + Y \cos\theta$$

$$Y_e = -X \cos\theta \cos\phi - Y \sin\theta \cos\phi + Z \sin\phi$$

$$Z_e = -X \cos\theta \sin\phi - Y \sin\theta \sin\phi - Z \cos\phi + \rho \quad (2.7)$$

步驟五：將 (X, Y, Z) 投影至螢幕上如圖 2.15。其中， D 為原點至影像面之距離。根據相似三角關係，可以導出：

$$S_x = D(X/Z)$$

$$S_y = D(Y/Z)$$

若 (X, Y, Z) 已轉換成 (X_e, Y_e, Z_e) ， D 變成眼睛至螢幕的距離，則上二式就成了

$$S_x = D(X_e/Z_e)$$

$$S_y = D(Y_e/Z_e)$$

以上的投影公式即是透視投影。

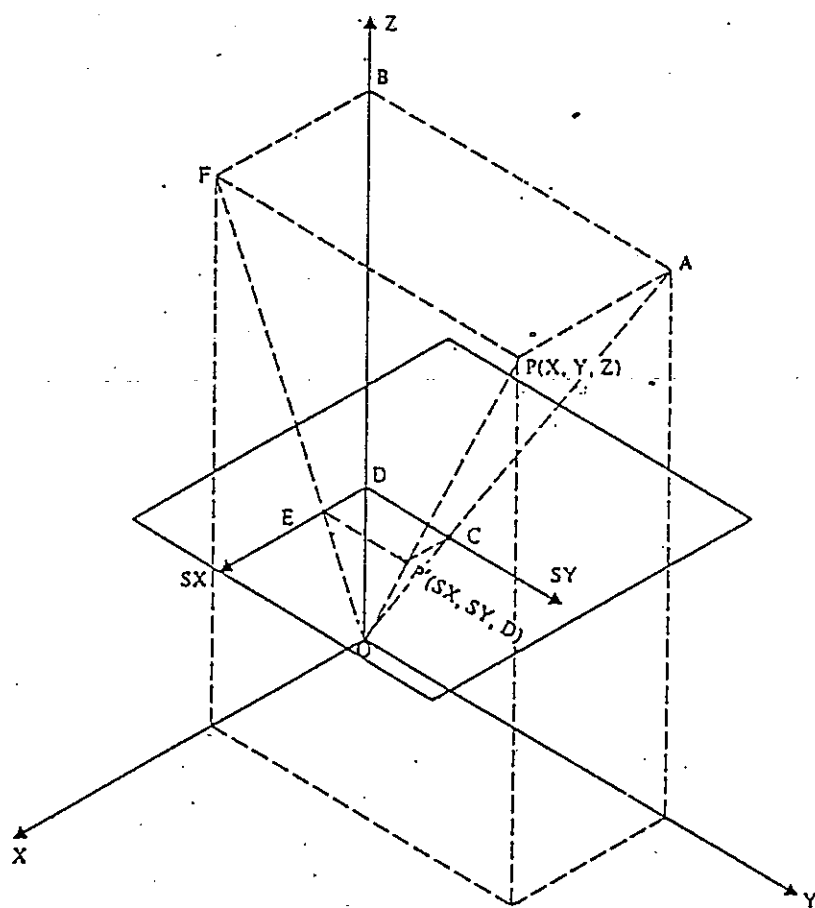


圖 2.15

第三章 自由形狀物體設計

自由形狀的物體，通常可以由左右輪廓的縱剖面 (*profile*) 及上下的橫剖面 (*cross-section*) 圍繞而成。本系統即是將縱剖面及橫剖面用 *B*-軟楔曲線形成，再視物體的種類由曲面旋轉、直線軸設計及自然軸設計造出基本部分，並由第四節的密合技術來產生完整的物體。

3.1 曲面旋轉

曲面旋轉是最簡單的設計方法，就是將一條平面上的 *B*-軟楔曲線，沿著座標軸旋轉，產生對稱性的物體。由於本系統的縱剖面是在 *X-Z* 平面上，因此我們選擇繞著 *Z* 軸旋轉，若縱剖面的參數式為

$$P(u) = X(u)i + Z(u)k \quad 0 \leq u \leq 1$$

當縱剖面繞 *Z* 軸旋轉，形成曲面時，控制點座標的 *Z* 分量保持不變，而 *X* 分量會沿著 *X-Y* 平面旋轉出一個圓形，故其旋轉曲面為

$$r = X(u)\cos\theta i + X(u)\sin\theta j + Z(u)k$$

$$0 \leq u \leq 1$$

$$0 \leq \theta \leq 2\pi$$

圖 3-1 是利用曲面旋轉將 *X-Z* 平面上的縱剖面 *B*-軟楔曲線旋轉產生的花瓶。生活中凡是對稱的物體，都可利用此法產

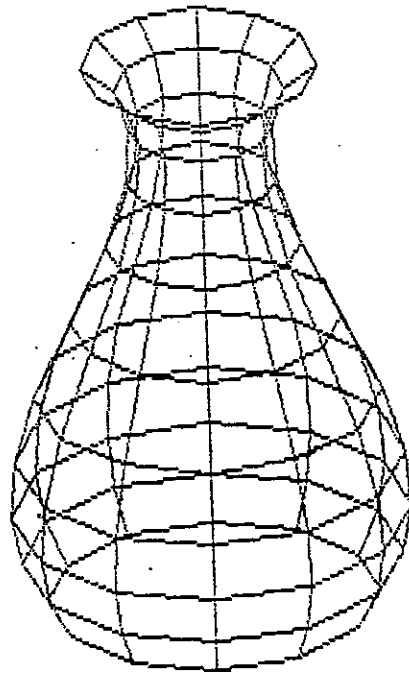


圖 3.1

3.2 直線軸設計

本系統中是將縱剖面定義在 $X-Z$ 平面上，而將橫剖面定義在 $X-Y$ 平面，藉由控制點分別調整縱剖面及橫剖面的形狀，可繪製三維的物體外形，依橫剖面與縱剖面數目之不同可分成以下三種設計方法：

3.2.1 單一縱剖面，單一橫剖面

此法是由一個縱剖面及一個橫剖面所形成，利用的幾何內積產生立體輪廓。

假設縱剖面是

$$P(u) = Xp(u)i + Z(u)k \quad 0 \leq u \leq 1$$

而橫剖面是

$$C(v) = Xc(v)i + Y(v)j \quad 0 \leq v \leq 1$$

則兩者所形成的物體，其參數式為

$$r = Xp(u)Xc(v)i + Xp(u)Y(v)j + Z(u)k,$$

$$0 \leq u \leq 1$$

$$0 \leq v \leq 1$$

圖 3.2 即是利用此法所產生的物體。

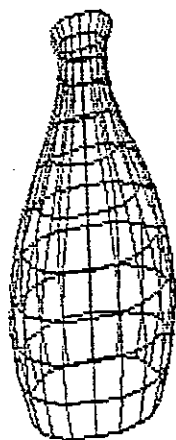


圖 3.2

3.2.2 單一縱剖面，雙橫剖面

此法是由一個縱剖面及上下兩個橫剖面所形成，方便物體的變化，上、下橫剖面分別位在縱剖面的最高和最低處，中間的橫剖面形狀則介於其間，並作漸近的改变。

假設縱剖面是

$$P(u) = Xp(u)i + Z(u)k$$

$$0 \leq u \leq 1$$

而兩橫剖面是

$$C_0(v) = Xc_0(v)i + Y_0(v)j \quad 0 \leq v \leq 1$$

$$C_1(v) = Xc_1(v)i + Y_1(v)j \quad 0 \leq v \leq 1$$

則三者所形成的物體為

$$r = Xp(u)[(1-u)Xc_0(v)i + uXc_1(v)]i + Xp(u)$$

$$[(1-u)Y_0(v) + uY_1(v)]j + Z(u)k$$

$$0 \leq u \leq 1$$

$$0 \leq v \leq 1$$

圖 3.3 為此法所生成的。

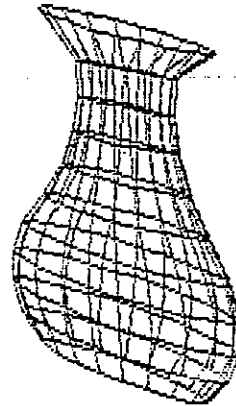


圖 3.3

3.2.3 雙縱剖面，雙橫剖面

此法是由左右兩個縱剖面及上下兩個橫剖面產生橫剖面與縱剖面皆有變化的物體。

假設縱剖面是

$$P_0(u) = Xp_0(u)i + Z_1(u)k \quad 0 \leq u \leq 1$$

$$P_1(u) = Xp_1(u)i + Z_1(u)k \quad 0 \leq u \leq 1$$

而兩橫剖面是

$$C_0(v) = Xc_0(v)i + Y_0(v)j \quad 0 \leq v \leq 1$$

$$C_1(v) = Xc_1(v)i + Y_1(v)j \quad 0 \leq v \leq 1$$

則三者所形成的物體為

$$r = P[(1-u)Xc_0(v)i + uXc_1(v)]i + P[(1-u)Y_0(v) + uY_1(v)]j + [(1-v)Z_0(v) + vZ_1(v)]k$$

$$0 \leq u \leq 1$$

$$0 \leq v \leq 1$$

其中 $P = (1-2v)Xp_0(u) + 2vXp_1(u)$ $0 \leq u \leq 1, 0 \leq v \leq 1$

3.3 自然軸設計

在我們所見的物體中，除了上述所見的直利物體之外，亦有很多管狀的物體。在設計這類的物體時，我們利用一條脊推曲線 (spine curve) 當作軸心，讓橫剖面沿著脊推曲線彎曲前進，形成管狀物。如同縱剖面一樣，假設脊推曲線是定義在 X-Z 平面上，如圖 3.4 所示

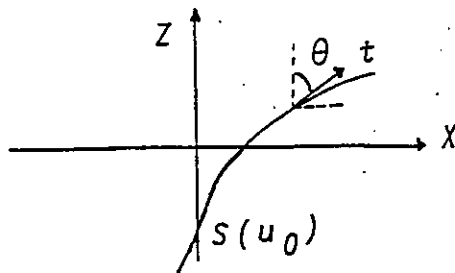


圖 3.4

設脊推曲線的參數式為

$$S(u) = Xs(u)i + Z(u)k \quad 0 \leq u \leq 1$$

而橫剖面是

$$C(v) = X(v)i + Y(v)j \quad 0 \leq v \leq 1$$

為了使橫剖面沿著脊推曲線作垂直方向的彎曲，令 $S(u)$ 在

$u=u_0$ 時的曲線 t 與 Z 軸的夾角為 θ ，則將 $C(v)$ 轉換到以 $S(u_0)$ 為原點且與 t 垂直的平面上的步驟為：

(一). 逆時針方向旋轉 Y 軸 θ 度。

(二). 平移原點到 $S(u_0) = (Xs(u_0), 0, Z(u_0))$ 。

由以上二步驟的矩陣轉換，我們可得：

設橫剖面上的點 $[X \ Y \ Z \ 1]$ ，經轉換後成為 $[X' \ Y' \ Z' \ 1]$

$$[X' \ Y' \ Z' \ 1] \\ = [X \ Y \ Z \ 1] \begin{bmatrix} \cos\theta & 0 & -\sin\theta & 0 \\ 0 & 1 & 0 & 0 \\ -\sin\theta & 0 & \cos\theta & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ Xs(u_0) & 0 & Z(u_0) & 1 \end{bmatrix}$$

($\because Z = 0$)

$$= [X \ Y \ 0 \ 1] \begin{bmatrix} \cos\theta & 0 & -\sin\theta & 0 \\ 0 & 1 & 0 & 0 \\ -\sin\theta & 0 & \cos\theta & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ Xs(u_0) & 0 & Z(u_0) & 1 \end{bmatrix}$$

$$= [X\cos\theta \ Y \ -X\sin\theta \ 1] \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ Xs(u_0) & 0 & Z(u_0) & 1 \end{bmatrix}$$

$$= [X\cos\theta + Xs(u_0) \ Y \ -X\sin\theta + Z(u_0) \ 1]$$

$$\text{即 } X' = X\cos\theta + Xs(u_0)$$

$$Y' = Y$$

$$Z' = -X\sin\theta + Z(u_0)$$

固定的脊推線，依橫剖面數目，分為下列兩種方法

3.3.1 固定脊推線，單一橫剖面

此法是由一個橫剖面和一個脊推曲線組成，

設脊推曲線是

$$S(u) = Xs(u)i + Z(u)k \quad 0 \leq u \leq 1$$

橫剖面是

$$C(v) = X(v)i + Y(v)j \quad 0 \leq v \leq 1$$

利用上述的轉換，求出轉換之後的橫剖面的控制點位置，即可形成管狀物體

$$r = [X(v)\cos\theta + Xs(u)]i + Y(v)j + [-X(v)\sin\theta + Z(u)]k$$

$$0 \leq u \leq 1$$

$$0 \leq v \leq 1$$

圖3.5為此法生成的例子。

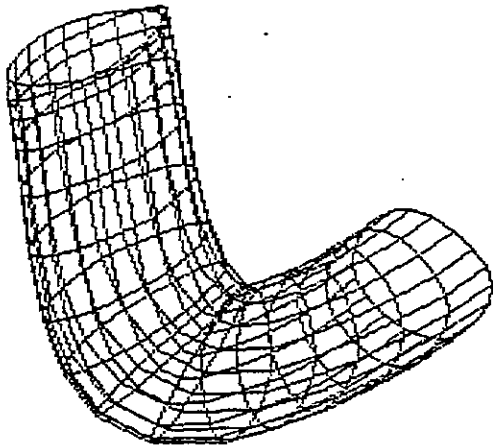


圖 3.5

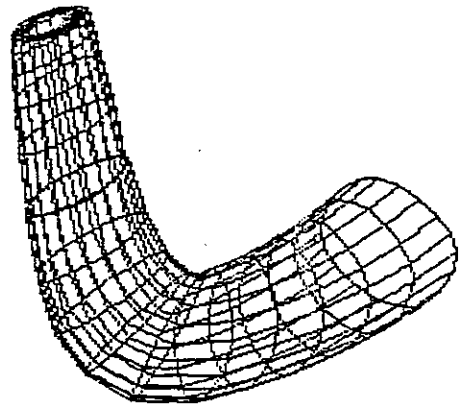


圖 3.6

3.3.2 固定脊推線，雙橫剖面

本法是利用上下二橫剖面及一個脊推曲線組成，

設脊推曲線是

$$S(u) = Xs(u)i + Z(u)k \quad 0 \leq u \leq 1$$

兩橫剖面是

$$C_0(v) = X_0(v)i + Y_0(v)j \quad 0 \leq v \leq 1$$

$$C_1(v) = X_1(v)i + Y_1(v)j \quad 0 \leq v \leq 1$$

先由兩個橫剖面的線性組合產生透視的特性，再轉換到管狀的位置，其管狀參數式為

$$r = [((1-u)X_0(v) + uX_1(v))\cos\theta + Xs(u)]i \\ + [(1-u)Y_0(v) + uY_1(v)]j \\ + [-((1-u)X_0(v) + uX_1(v))\sin\theta + Z(u)]k \\ 0 \leq u \leq 1 \\ 0 \leq v \leq 1$$

圖 3.6 為一例。

3.4 密合技術

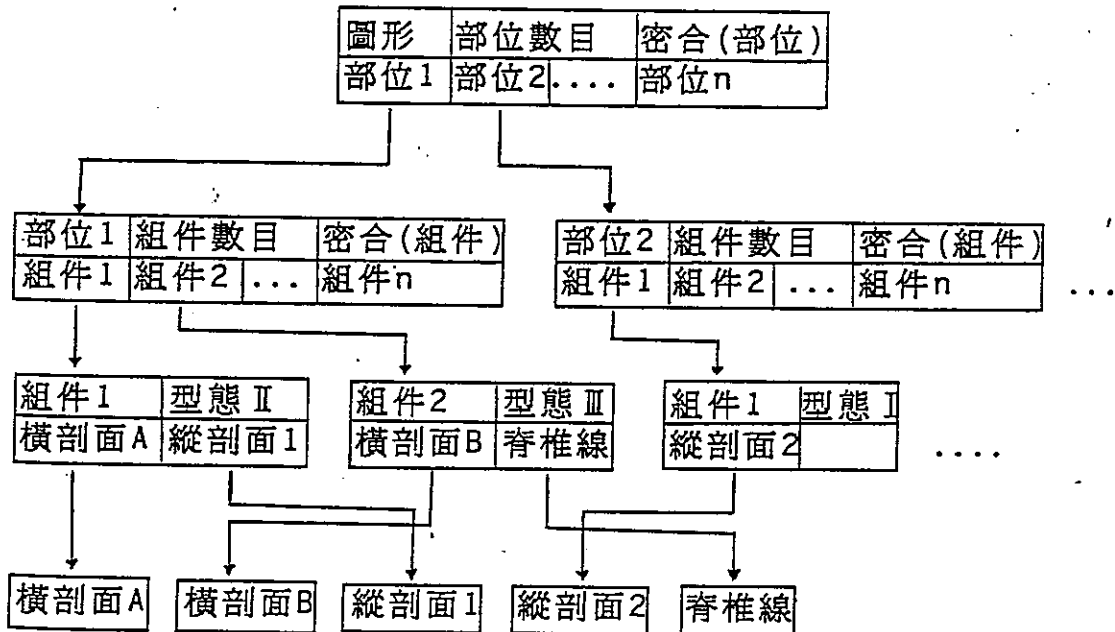


圖 3.7 圖形資料結構

圖3.7是本系統所建立的資料結構，在建立物體圖形時，將圖形檔分成數個物體檔，而物體檔又可區分為數個部位檔，這些部位都是由以上三節的設計方式所生成，部位與部位間是利用密合計術 (GLUE METHOD)，做為兩個部位的銜接工作。其演算法如下：

密合技術演算法

假設

$P(u, v)$: $0 \leq u \leq 1, 0 \leq v \leq 1$ 為貼合物

$P(w, s)$: $0 \leq w \leq 1, 0 \leq s \leq 1$ 為被貼物

$P(u, v) \odot P(w, s)$: 為 $P(u, v)$ 與 $P(w, s)$ 的密合體

$P(u, 0)$: 為 $P(u, v)$ 的貼合面

$H(P(u, 0))$: 為 $P(u, 0)$ 的半線，即貼合面上

$u=0$ 到 $u=0.5$ 的連線

$P1(u, 0)$: $0 \leq u \leq 0.5$ ，為 $P(u, 0)$ 被 $H(P(u, 0))$ 分割的左半面

$P2(u, 0)$: $0.5 \leq u \leq 1$ ，為 $P(u, 0)$ 被 $H(P(u, 0))$ 分割的右半面

S_1, S_2 : 為密合體之 s 值

輸入

$P(u,v), P(w,s), s_1$

輸出

$P(u,v) \odot P(w,s)$

步驟一：利用曲線旋轉、直線軸設計或自然軸設計三種型態之一產生貼合物 $P(u,v)$ 。

步驟二：在同一螢幕平面，利用曲線旋轉、直線軸設計或自然軸設計三種型態方式之一產生被貼物 $P(w,s)$ 。

步驟三：將 $P(u,0)$ 投影到 $P(w,s)$ 。先將 $H(P(u,0))$ 投影到 $P(0,s)$ 曲線上，由畫家給定 s_1 值，分別將左半面及右半面投影到被貼物的前、後半面。因半線經投影後的長度約與半線相等，故 s_2 的值可以由投影半線的長度與 $P(0,s)$ 的長度之比值加上 s_1 求得。

```
procedure proj(P(u,0), P(w,s))
    proj(H(P(u,0), P(0,s))
    s2 = s1 + H(P(u,0)
    P(0,s)
    for s = s1, s2 do
        proj(P1(u,0), P(w,s))
        proj(P2(u,0), P(w,s))
    end
```

步驟四：利用動畫技巧產生密合效果。將 $P(u,v)$ 與投影後的曲面藉不同路徑的動畫過程生成密合體。

```
procedure animate( $P(u,v)$ ,  $P(w,s)$ )
```

```
     $P(u,0)$  為動畫的主畫面1
```

```
     $\text{proj}(P(u,0), P(w,s))$  為動畫的主畫面2
```

```
     $t$  為時間變化
```

```
    for  $v = 0$  to 1
```

```
        for  $t = 0$  to 1
```

```
            調整APCS的 $T, B, C, A$ 四參數
```

```
            產生 $P(u,v)$ 到 $\text{proj}(P(u,v), P(w, s))$ 
```

```
            密合過程的動畫
```

```
        end
```

```
    end
```

今以圖3.8中之壺身及壺嘴說明密合技術：壺嘴是利用自然軸設計法產生的，壺身是由曲面旋轉生成的，當壺嘴要貼近壺身時，壺嘴的貼合面先將橫剖面的半線投影到壺身的縱剖面的起始處，再將左、右半面分別投影到壺身的前、後半面，取適當位置作投影的起點，由於壺嘴的半線與投影在壺身的長度皆為小的數值，為方便運算，我們

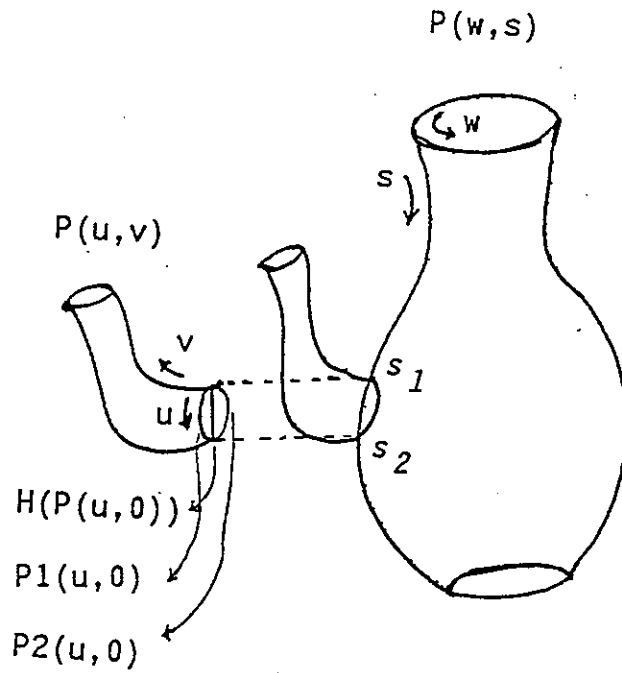


圖3.8

將二者視為相同，因此可以得到投影的終點，進而得知投影到壺身的投影面。再則，將壺嘴中的所有橫剖面依序投影到壺身上，得到各自的投影面，利用動畫路徑控制系統 (APCS) 將壺嘴上的橫剖面作為起始之主畫面，壺身上的投影面當作終止之主畫面，由此二個主畫面產生的動畫結果即是壺嘴在壺身的密合體，並且由動畫的過程亦可表現出各型各樣的壺嘴接法，適宜作動畫中的趣味或誇張效果。

第四章 物體導向化模組的建立

在物體外形的生成環境中，我們將三維物體分成階層式的圖形架構，將此架構中的每個部位當成物體導向化程式中的"物體"利用物體間的資料傳遞，產生複雜的立體外形。本章將簡介物體導向化程式的特性，並推廣其在圖形模式的應用。

4.1 物體導向化程式

物體導向化程式[10][11]可以廣泛應用在程式語言及電腦圖學上，其中惟一的共同點即是以"物體"為核心，由圖4.1所示，物體是由程序(procedure)和資料(data)所組成，二者在程式中擔任運算執行及資料儲存的工作，藉由訊息傳遞(message sending)的技巧，即一種非直接的程式呼叫，從執行完成的程序中送出一個訊息給另一物體，此物體再藉著本身的程序執行，並回應一個訊息再作為另一物體所需的資料，如此的動作在階層式的架構上能迅速達成物體間資料的存取。

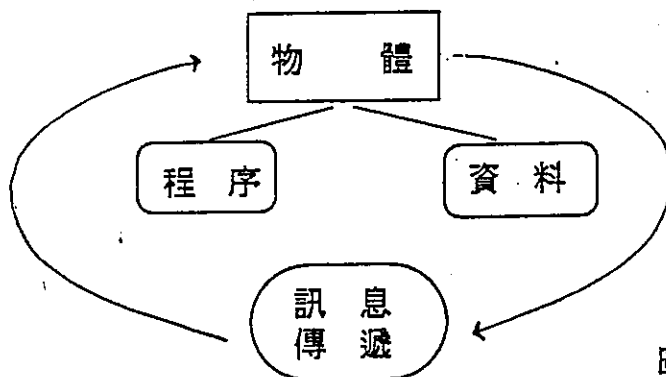


圖 4.1

物體導向化程式與傳統程序導向化程式的差異在於：程序導向化程式是以程序或函數為核心，程式製作之前須先設定程式需要的功能，再利用由上而下的設計方式規劃程式架構，Nikulas定義的PASCAL語言即是利用此概念發展出來的。而物體導向化程式則是以物體為核心，當程式擴充時，只要更改觀念結構，不需要更改整個程式，即更改物體本身及與其相關的訊息，其他部分維持不變。

完整的物體導向化程式俱備四項特性：資訊隱藏 (Information hiding), 資料抽象化 (Data abstraction), 動態束結 (Dynamic binding) 及繼承性 (Inheritance), 茲分述如下：

一. 資訊隱藏

在物體導向化程式中所有物體的值都是類集 (class) 的一個例現 (instance)。類集包括建立與傳送某個物體所需要的所有資料，即它的所有例現的集合，因此每一個例現必屬於唯一的類集，而一個類集可能有很多個例現。每個例現則是例現變數 (instance variable) 的一個值，例現變數可以是某種資料型態或其他的物體，而每種物體都屬於其特定的例現型態之集合。資訊隱藏的目的在維持整個程式系統的可靠性與擴充性。因為程式中各模組是由個別的獨立變數組成，模組間的資料傳遞需靠局部性的程序來產生資料，因此，各模組間作例現變數的傳遞時，除了

改變物體的資料外，尚須保證不影響其他模組間的資料與程序。

二．資料抽象化

資料抽象化的原理是由於目前執行的程式無法對其所使用的資料作設定，利用資料抽象化改變目前所用程式的下屬的狀態，而不需改變其本身的狀態。因此，資料抽象化的目的乃是將所有程序中同類的資料型態用一特定機體連結在一起。

三．動態束結

動態束結是將不同類的資料放入堆疊(stack)中，再充分利用此堆疊作運算。例如：將整數，實數，字元等同時存入一個堆疊中，在傳統程序導向程式下，若同時選擇整數，字元作列印指令，必然會出現錯誤訊息。因為不同類別的資料型態必須在不同的陳述下，經由不同的編譯來執行。在物體導向化程式下，當物體送出一個訊息(整數或字元)時，物體會以其本身的程序作列印指令。

四．繼承性

繼承性是將同一類的例現歸類成同一類集，以別於其他的類集。以伸縮喇叭為例，它是屬於樂器類，又依材料性質區分，其歸類於銅器類。同一類集可以分出多個子集(subclass)或是隸屬於其父集(superclass)之下，子集繼承了

父集的例現變數 (*instance variable*)，類集變數 (*class variable*) 及傳遞方式 (*method*) 等屬性，當父集增加或改變例現變數、類集變數或傳遞變數時，其子集亦將隨之增加或改變。

4.2 物體導向化圖形模組

本文在設計三維物體時是利用上節所述物體導向程式的特性，將三維物體建立成階層式的圖形模組架構，利用各模組間的繼承性傳相同的資料型態，逐次建立起複雜的立體圖形。而在動畫表現上，亦可由物體導向化的觀念產生局部性、整體性或混合式的運動效果。

本系統是以 *B*-軟楔曲面為基礎建立三維物體輪廓，藉由其生成之三類基本曲面組合成複雜的外形。如圖 4.2 生成一階層式的圖形架構定義物體的幾何形狀。*B*-軟楔曲面乃是利用 *u, v* 兩方向之 *B*-軟楔曲線構成，而 *B*-軟楔曲線則是靠控制點的輸入來決定曲線的形狀。

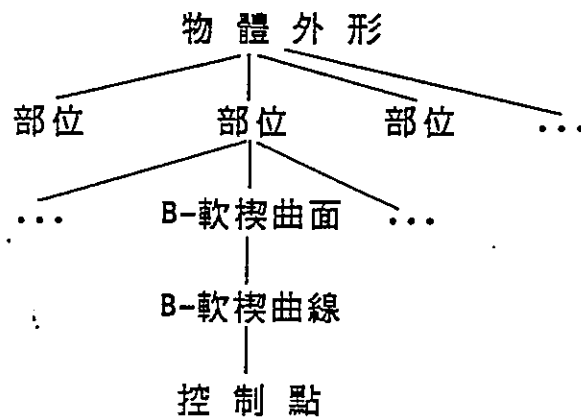


圖 4.2

今以一個人體架構為例說明物體導向化結構在立體外形的生成與動畫控制的過程。在圖4.3中，我們將人體分成身部與頭部，身部又可分成手，軀體及腳三部分，手和腳又可分成左右二部分，整體形成階層式物體架構其中左，右手是手部的例現，其不同的例現值-控制點藉由手部之例現變數-曲面生成法而生成左、右手。同理，左、右腳亦是藉由腳部之曲面生成法而生成。身部則繼承其子集包括軀體，手部與腳部的屬性-密合技術組成，再加上頭部的例現，則組合整個人體架構。

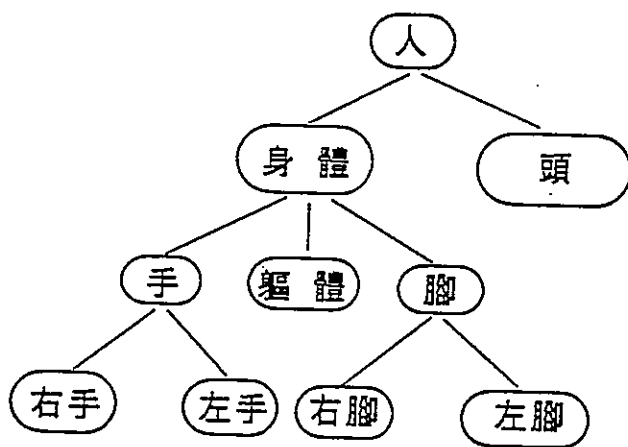


圖 4.3

在動畫過程中，物體各部位可藉運動屬性的傳遞產生三種運動方式：

一．局部性運動：以前述為例，若靜止的人體上，手部作旋轉或屈肘動作，即屬之。

二．整體性運動：當腳部作前進或後退動作，手部維持不動，則屬之。

三. 混合式運動：當腳部作前進或後退動作，且手部作旋轉或屈肘動作，或身體的彎腰等皆屬之。

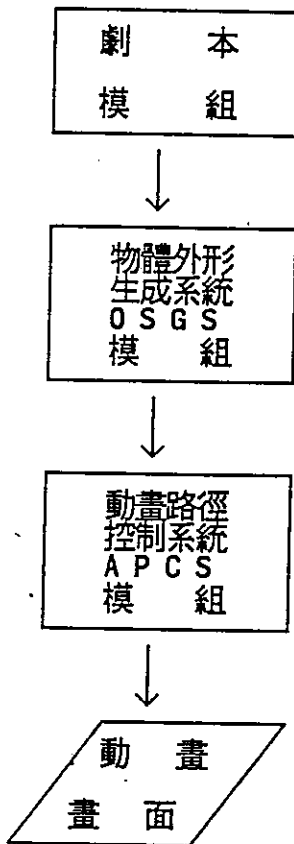
物體導向化圖形模組的優點是，階層式之物體架構具有繼承性的特性。繼承性使圖形模組的類集能繼承其父集的所有屬性，而且在模組間的階層性之搜尋時能維持動態束結的特性，因為在由下往上搜尋時，父集將漸次地包容子集的屬性，直到另一類集開始為止。

第五章 動畫系統架構與圖形範例

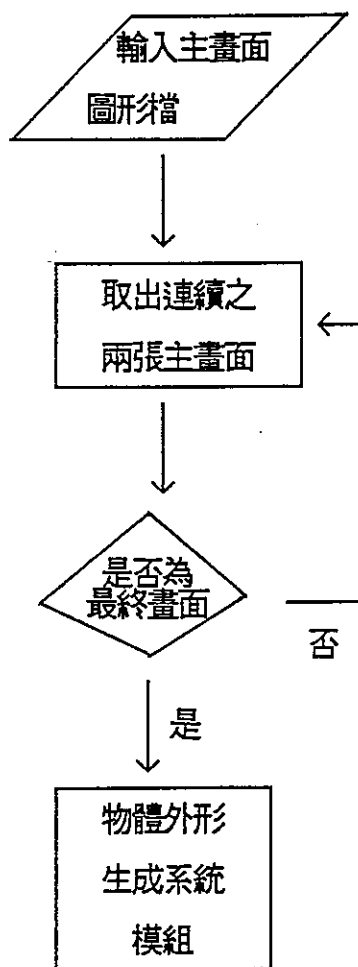
本章將就立體動畫製作過程以流程圖說明，以便讀者了解電腦動畫的製作方式。並將本文所作之『物體外形生成系統(OSGS)』產生之靜態與動態物體之輸入方法及立體物體的生成方法詳加說明之，並輔以一些圖形範例，作為實驗例證。

5.1 動畫系統架構

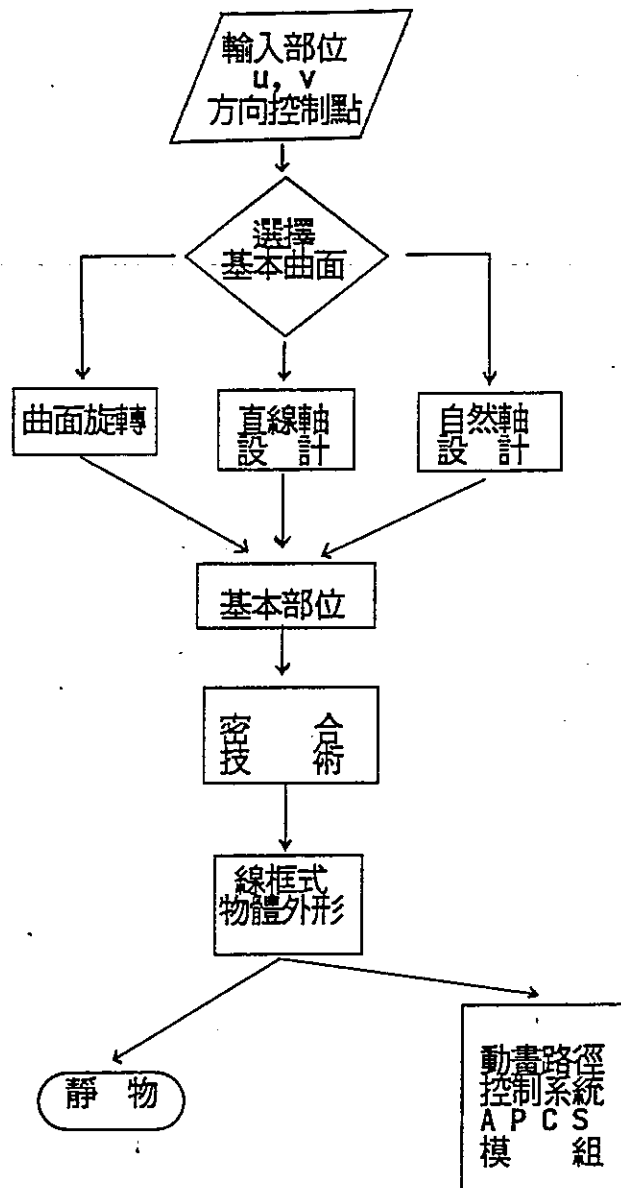
在製作動畫時，可將其製作步驟以下列之流程圖說明：



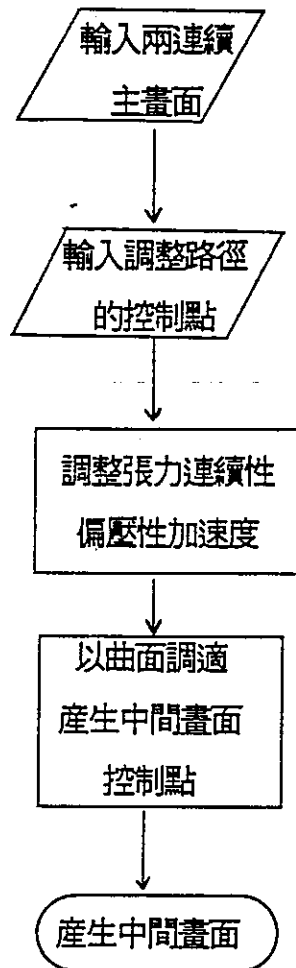
- 一. 劇本模組：確定完整圖形畫面之設計。設計動畫畫面時，須先確定所有造成單調運動變化的畫面，定為主畫面，由起始畫面開始，在兩個連續主畫面間產生中間畫面。



二. 物體外形生成系統模組：由劇本模組擬定之畫面，輸入縱剖面與橫剖面的控制點座標，利用B-軟楔曲面所成之三類基本曲面產生法生成各類部位，由密合技術將相關的部位接合，即可產生靜態之三維物體輪廓。



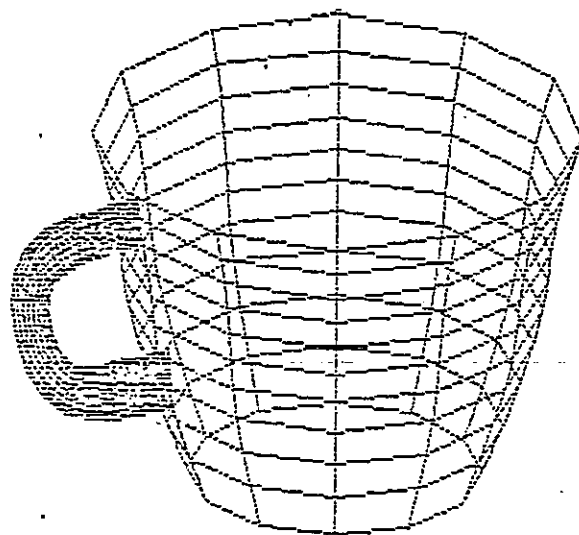
三. 動畫路徑控制系統模組：三維物體的主畫面建立後，依據連續主畫面間的變形利用張力，連續性，偏壓性及加速度四項變數來控制路徑，產生中間畫面。



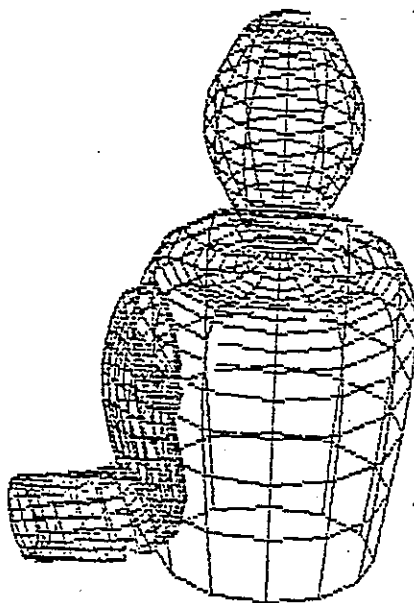
5.2 圖形應用範例

OSGS可以產生三維立體之靜物與動畫畫面，茲舉例如
下：

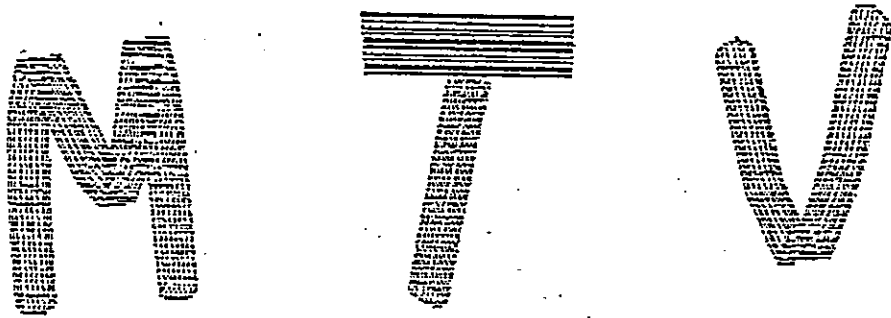
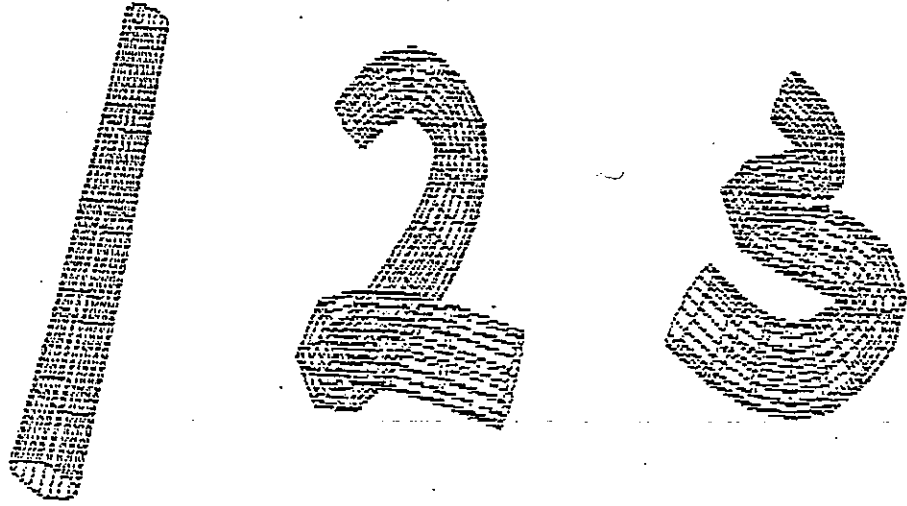
一．咖啡杯



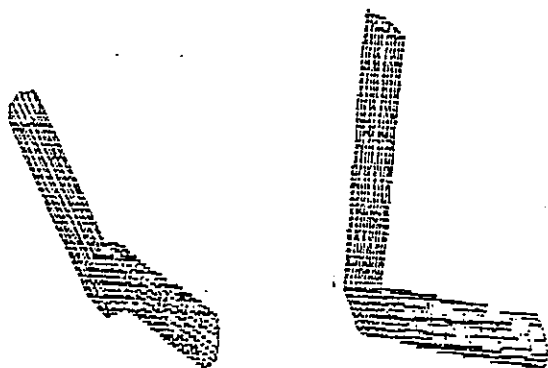
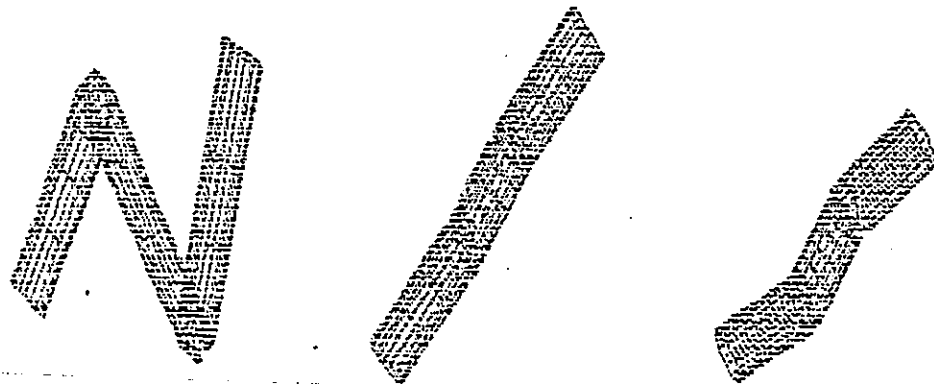
二．機械人



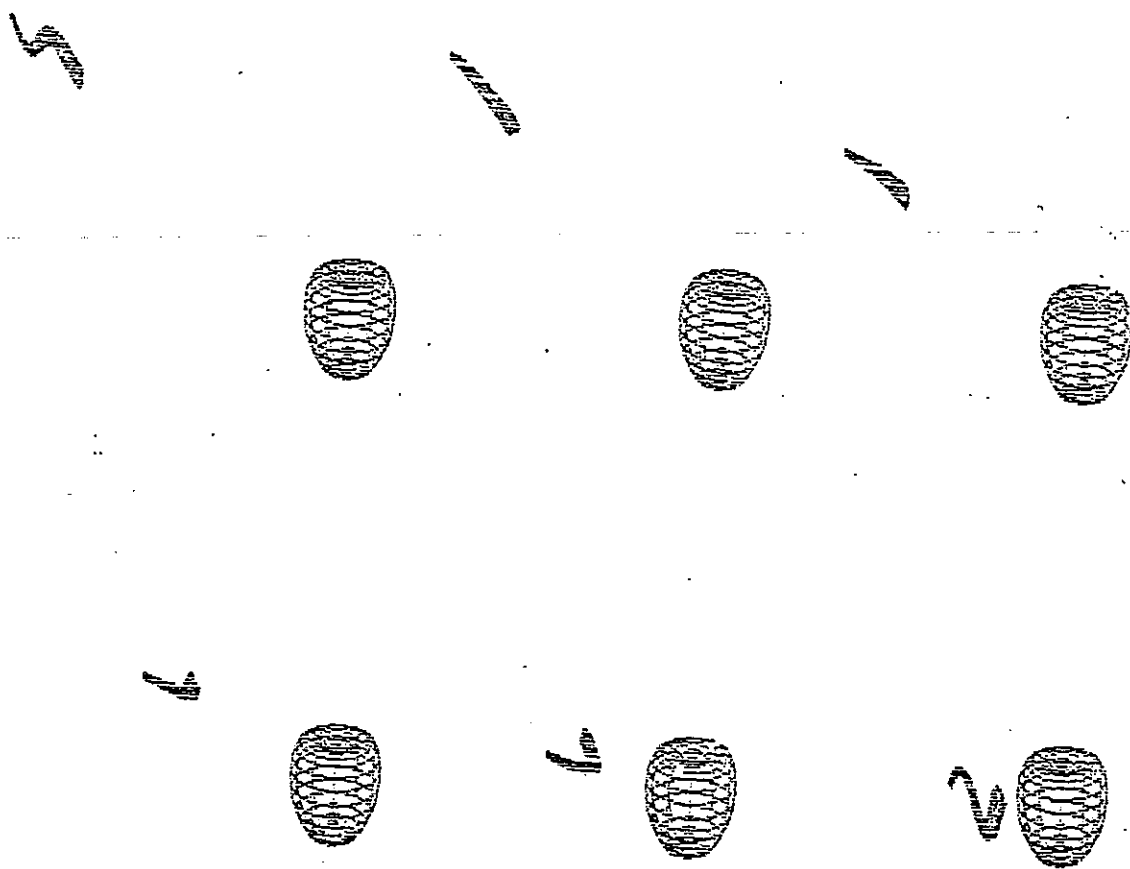
三. 立體藝術字



四. 文字的變化 (N → L)



五. 動畫模擬 (貪吃的蛇)



第六章 結 論

從物體外形生成系統(OSGS)的製作過程中，我獲得了許多電腦繪圖設計與電腦動畫的實務，使理論和實際相互配合。

現在將本系統的優，缺點和未來發展方向，略述如下：

優點：

1. 對靜態或動態自由物體外形的建立，提供卡通畫家或一般人一套簡單，易學的圖形輸入法。
2. 對三維物體外形輪廓生成以物體導向化架構分析，方便複雜物體的建立與擴充。
3. 對電腦卡通動畫的製作，設定完整的系統流程，提供廣告美工的新構想。

缺點：

1. 因為時間因素，至目前為止，本系統所處理的物體偏重於單物體的建立，生成的物體較不生動、自然。
2. 由於本系統在PC-AT上執行，受限於單色螢幕，缺乏對物體着色及環境光源的處理。
3. 沒有處理隱藏線(hidden line)及隱藏面(hidden surface)，不能明顯表示物體的遠近關係。

發展方向：

1. 將本系統架構轉移到 PS-390 繪圖工作站，改善本系統的隱藏線及隱藏面的處理，並加強色彩、光影的處理，產生多彩多姿的立體圖形。
2. 產生生活週遭所見之複雜物體的外形，利用動畫路徑的控制完成的動畫畫面，可藉攝影機將連續畫面拍攝下來製成動畫影片，可廣泛應用在電腦輔助教學 (CAI)，新聞媒體，工商廣告及娛樂節目 (MTV) 上。

參考文獻

1. C. D. Woodward, "Cross-sectional Design of B-spline Surface", Computer and Graphics, Vol.11, No.2, 1987, pp. 193-201
2. C. D. Woodward, "Skinning Technique for Interactive B-spline Surface interpolation", Computer-Aid Design, Vol.20, No.8, Oct.1988, pp.441-451
3. C. Mirolo and E. Pagello, "A Solid Modelling System for Robot Action Planning", IEEE Computer graphics and Applications, Jan.1989, pp.55-69
4. Daniel Thalmann, "Computer Animation Theory and Practice", McGraw-Hill Book Company, 1985
5. D. F. Rogers and J. A. Adams, "Mathematical Elements for Computer Graphics", Lookhead Georgia Company, 1978
6. D. N. Rochelean and K. Lee, "System for Interactive Assembly Modelling", Computer-Aid Design, Vol.19, No.2, Mar. 1987, pp.65-72
7. E. Cohen, Tom Lyche and R. Riesenfeld, "Discrete B-spline and Subdivision Technique in Computer-Aid Geometric Design and Computer Graphics", Computer Graphics and Image Processing, No.14, 1980, pp.87-111
8. I. D. Faux and M. J. Pratt, "Computational Geometry for Design and Manufacture", Ellis Hor-

wood Ltd.1979

9. K. Y. Cheng, "A B-spline Based Computer Animation System", Journal of the Chinese Institute of Engineer, Vol.4, No.1, 1981, pp.39-45
10. L. Mohan and R. L. Kashyap, "An Object-oriented Knowledge Representation for Spatial Information", IEEE trans. on Software Engineering, Vol.14, No.5 May 1988, pp.675-681
11. M. Stefik and D. G. Borow, "Object-oriented programming: Themes and Variations", The AI Magazine, Vol.6, No.4, 1986, pp.40-62
12. N. M. Thalmann and Daniel Thalmann, "Miranim: An Extensible Director-oriented System for the Animation of Realistic Images", IEEE Computer Graphics and Applications, Mar.1985, pp.61-73
13. N. M. Thalmann and Daniel Thalmann, "Image Synthesis: Theory and Practice", Springer-Verlag Book Company, 1987
14. W. Bohm, G. Farin and J. Kahmann, "A Survey of Curve and Surface Methods In GAGD", Computer Aided Geometric Design, 1984, pp.1-60
15. 鄭國揚, "交談式電腦圖形論", 儒林書局, 1984
16. 胡俊之, "以繪圖核心系統為基礎的電腦輔助幾何設計系統", 中興大學74年碩士論文

17. 陳石雄, "一個動畫路徑控制系統之設計", 中興大學78年碩士
論文