

TR-91-004

一套產生動畫物體之造型及紋理系統

參 考 書

不 外 借

中 央 研 究 院 資 訊 科 學 研 究 所
798.-1
圖 書

R5935

中研院資訊所圖書室



3 0330 03 000338 3

論文名稱：一套產生動畫物體之造型及紋理系統

頁數：70

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：曾德沛

指導教授：鄭國揚博士

論文提要內容：

本論文的研究目的是製作一供動畫系統使用的物體外形生成系統，並應用實體紋理 (solid texturing) 方法處理物體表面紋理。

本論文的重點有二：一般化掃掠物體 (general sweeping object) 產生方法的修改以及實體紋理應用在一般化掃掠物體的討論。

在眾多的電腦造形方法中，掃掠法是廣被採用的一種，一般化掃掠物體可由一定義在二維平面上的封閉曲線沿一定義於三度空間的軸線掃掠而成。掃掠過程由一大小函數 (scale function) 控制截面在軸線上的大小，以產生粗細不等的外形。由於截面侷限於平面封閉曲線，因此有些外形較複雜的物體就無法以這種方法表現。本文將截面的定義擴充到三度空間的封閉曲線，截面的形狀以動畫技術中的主圖像法產生，能造出更複雜而富變化的物體外形。

一般紋理方法是將二度空間的紋理樣本轉換到三度空間的物體表面上。其缺點是當物體外形太複雜，物體表面紋理會變形失真。實體紋理方法將紋理用定義於三度空間的函數來表現，只要將物體表面上點的座標代入函數就能得到該點的紋理，不需轉換座標系，因此不會有紋理失真的問題。但是動畫中的物體會運動過程中彎曲或扭轉，物體表面紋理也會跟著改變，造成視覺上紋理的不一致。本文針對一般化掃掠物體的特性提出解決的方法。

資訊工程 研究所 曾德沛 君所提之論文

一套產生動畫物體之造型及紋理系統 (題目),

經本委員會審議，認為符合碩士資格標準。

論文口試委員會 主任委員 吳豪明 (簽章)

委員 胡時

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，一年半以來不厭其煩的教
導與指正，使得本研究得以順利完成。論文試時，吳憲益
與胡咸教授所提出的寶貴意見，也使學生深受感。感
導明教誨。

同學楊政真、朱首龍、嚴漢偉、許不忠、以及顏文龍
，在課業上相互研討科學兩位先生對於本研究所進行的
，央黃俊堯、廖晉順、行政院國科會期間，毫無意
以專心於學業，與
，以上最高學業的
，在課業上相互研討科學兩位先生對於本研究所進行的
，央黃俊堯、廖晉順、行政院國科會期間，毫無意
以專心於學業，與
，以上最高學業的
，在課業上相互研討科學兩位先生對於本研究所進行的
，央黃俊堯、廖晉順、行政院國科會期間，毫無意
以專心於學業，與
，以上最高學業的

目 錄 頁次

第 一 章	緒 論	1
第 二 章	理論及方法	7
2-1	B-spline 曲線及曲面簡介	7
2-2	掃掠法簡介	19
2-3	中間圖像法與PCG	24
第 三 章	動畫物體的造型	31
3-1	截面、軸線之輸入及修改	31
3-2	產生物體表面控制點網路	34
3-3	產生四邊形平面組合	38
3-4	圖例說明	40
第 四 章	動畫物體的實體紋理處理	45
4-1	產生紋理方法介紹	45
4-2	變形物體的實體紋理處理	50
4-3	實體紋理函數	52
4-4	實體紋理產生系統	55
第 五 章	系統架構及流程	59
5-1	硬體架構	59
5-2	軟體架構	59
5-3	系統流程	61
第 六 章	實作結果	63
第 七 章	結論	67

第一章 緒論

幾何造型在電腦科技普遍應用後立刻成為不可或缺的工具。用電腦處理物體造型具有精確、迅速、容易分析、修改方便等優點，大大節省了設計及製造上的成本。機械、電子、建築等工業對此依賴尤切。除了在這些實用的工程方面成果輝煌，電腦幾何造型與藝術結合更開拓了全新的視野。1974年坎城影展的審查委員獎頒給一部由電腦設計造型的動畫影片 "La Faim"，由此可見電腦在藝術表現上的能力不容置疑。

製作三度空間電腦動畫可略分為三個步驟：[1]

1. 物體造型 (object modeling)

將動畫所需的背景、道具和人物在三度空間中的外型轉換成電腦內部的表現模式，再輸入電腦。

2. 設定物體、攝物機本身及其相互間的運動狀態，這就是運動設定及同步 (synchronization) 階段。

3. 影像呈現 (Image rendering)

賦予物體逼真生動的外觀，這個步驟負責隱藏面消除、著色、陰影、透明度及表面紋理的處理。

三步驟中以物體造型耗費最多人力、時間和創意。因為日常所見的物體多由曲面構成，要產生外狀複雜但表面平滑（曲線上任一點斜率、曲率均連續）的曲面，不但需要使用到複雜的數學，而且要具備良好的空間觀念。如何將這些繁複難解的數學公式隱藏在造型系統中，並提供簡易方便的介面，讓一般使用者，即使沒有很好的空間觀念也能造出複雜但自然的物體外型供動畫使用，這就是本文的重點。

以下簡介電腦動畫中物體產生方式及內部表現形式，進而釐定論文討論的範疇：

一般物體造型的方法有三：

1. 將實物或模型的 3D 座標以數位輸入裝置存入電腦，或是輸入不同角度拍攝物體的照片，再由電腦重建出物體的三度空間外形，但是製作費時。
2. 以程式產生外形：規則外形物體如球、方塊、錐體等可根據數學公式產生不規則外形物體如自然界中的山、雲、植物等可使用隨機模式 (stochastic model)、碎形公式 (fractal) 產生。這種方法優點是產生外型所需資料量極少，只要少數的參數就能生成十分複雜的外形，但是有缺點：使用者必需具備數學基礎。

3.最常用的方法是經由修改、組合外形較簡單的構件 (primitive) 而得到所需物體外形。產生構件的方法很多，其中“掃掠法”是廣泛採用的一種。掃掠法 (sweeping) 的基本觀念如下：一物體可分為截斷面和一軸線 (spine)，物體外形可視為截斷面沿軸線掃掠而成。如圓柱體由圓形截斷沿一直線掃掠而成。圓環 (torus) 則是由圓形截面沿一圓形軸線掃掠成的。至於截斷面間的形狀變化，多數學者以縱剖面曲線 (profile curve) 來定義縱剖面的形狀，藉此可產生各類不同外形的物體。Thalman 於 1986 年提出一種結合動畫中間形狀產生及掃掠法的新方法，稱為 Propagation control graph (PCG)：一物體可給定許多形狀各不相同的截斷面，各截斷面視為動畫設計中的不同“主畫面” (key frame)，沿軸線掃掠時，截斷面的形狀不再決定於定義縱剖面的數學函數，而是根據動畫法則 (animation law)，如此造成的外形更加富有變化，而且使用上更加方便。

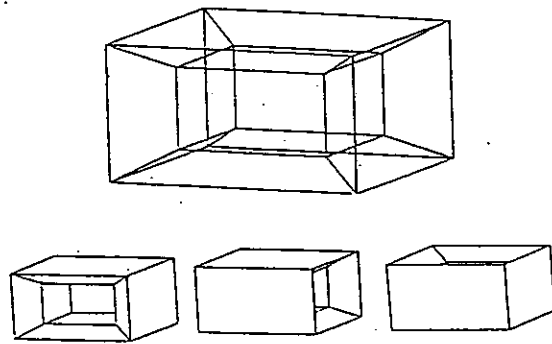
一般在電腦上表現物體外形的方式有如下三種：

1. 線框模式 (wire frame model)：這是最早被採用，也是最接近人類作畫的方式。將物體視為許多線段的組合，產生線段組合的基本動作有二：

`move abs <x,y,z>`: 將目前位置所在移到新的位置上。 $\langle x,y,z \rangle$ 為新位置的座標

`line abs <x,y,z>`: 將目前位置所在的點與下一點的 $\langle x,y,z \rangle$ 連成一直線。

重複以上動作就能表示出物體在空間中的外形。此法的優點是簡單，最大的缺點在於單憑線段無法確定的表示物體真正的形狀，如下圖(1.1)：



圖(1.1)

2. 面模式 (surface model): 將物體視為許多平面或曲面的組合。上圖物體一經指明是由那些面組合而成，線框模式中含混不清的問題便迎刃而解。當物體表面較為複雜或是彎曲度稍大時，必須使用大量的平面來表示才不致遺漏一些細節。為解決這個問題，學者又發展出曲面作為組合的工具。如: Coon's surface patch, Besier Surface patch, B spline surface patch 等。這些方

法是將一曲面分為許多曲面片斷 (patch)，每個片斷由二變數的多項式函數的線性組合構成，各曲面片斷的接合處至少可具有斜率 (slope)連續 (C^1 continuity)的特性，連接這些曲面片斷就能得到一平滑的曲面。但是單憑平面或曲面仍無法完全表現物體真實的空間特性，例如：空心的球由兩個球心相同的曲面構成，但是我們只靠這兩曲面，無法斷定它是具有一定厚度的球殼或只是兩個曲面。

3. 實體模式 (Solid model)：在這個模式下，物體被視為三度空間中的實體，由二維的邊界 (boundary) 分隔成內部和外部。一個實體具有如下幾點特性：佔據有限空間，應能以有限數目的元素 (面、端點) 來描述，具有不變的外形，經某些布林運算 (如：聯集、交集、差集) 後產生的新物體應仍具有實體的特性。由於實體模式能精確的表現物體的外形、體積和質量，因此被廣泛應用在機件設計及應力分析上。

基於以上討論，我們發現 PCG法非常適用於產生動畫的物體外形。使用者不需具備任何數學基礎，只要利用滑鼠在螢幕上畫出物體中少數主要截面及軸線的形狀，再指定各截面在軸線上的位置，就能得到所要的物體造型。以

上過程已經將複雜的三度空間問題簡化到較易理解掌握的平面問題。因此本論文將採用 PCG法來產生物體外形。至於動畫物體的內部表現形式，因為電腦動畫中物體造形的基本要求是生動、富變化，至於體積、質量等物理特性在電腦動畫中並不重要。因此本論文所討論的物體造型並不包括實體模式，只限於面模式。

掃掠法及 PCG法中截斷面均定義為二度空間中的封閉曲線，本文將討論以三度空間中不共面的封閉曲線作為截面所提供的一些非常實用的效果。3D 截面可能會造出不正確的（本身相交 self-intersection）表面，關於使用上的限制也將加以說明。完成物體造型後，本文針對以

PCG法造出物體的特性，討論實物紋理（solid texturing）方法在物體紋理表現上的應用，並製作一紋理處理子系統，使用者只需指定一些簡單參數就能生成富變化的紋理

接下來第二章介紹 B-spline 曲線、曲面的數學方法及掃掠法、PCG 法的理論基礎。第三章詳細說明如何利用 PCG 法產生物體外形以供動畫使用。第四章討論實體紋理方法應用在 PCG 法造出的物體上的限制及解決方法。第五章列出整個系統的流程及硬體架構。第六章附上使用本系統造出的一些影像。第七章提出未來研究方向及結論。

第二章 理論及方法

日常所見的物體，大多數可由曲線和曲面構成，為了造出較自然的物體外形，曲線和曲面是不可或缺的工具。本造型系統使用三次 B-spline 曲線作為基本工具，第一節將介紹 B-spline 的數學性質。第二節介紹以“掃掠”為主的幾種表面產生方法。

2-1 B-spline 曲線簡介 [2]

一般產生曲線的方法大致可分為兩種：1. 內插曲線 (interpolation curve)，這類曲線會通過所有的輸入點。如 Lagrange interpolation, Hermite interpolation, Spline interpolation 等。2. 近似曲線 (approximation curve)，這類曲線並不通過每一個輸入點，而是逼近輸入點。如 Bernstein approximation, B-spline approximation 等。

就設計者的觀點來看，內插法有下列幾項不方便處：

1. 除了輸入點的位置，曲線端點的切線向量之大小及方向也會影響曲線形狀，而二者必須由使用者指定，然而一般使用者較難了解向量大小及方向與曲線形狀之間的關係。
2. 產生曲線後，常須修改其中某個段落的形狀。根據內插曲線的性質，只要曲線中任一輸入點的位置有所變動，整條曲線必須重新計算，因此改變某個段落的形狀會連帶影響到曲線其他部分的形狀。

近似曲線中的 B-spline 曲線，每段曲線形狀完全由相鄰四個控制點的位置決定，不必處理切線向量的問題。要更改曲線中某段形狀，只需調整少數控制點的位置，曲線其他部分仍維持原狀。由於 B-spline 曲線具有這些優點，因此本文採用近似 B-spline 作為繪製曲線的工具。以下對近似 B-spline 曲線的數學性質作一說明：

B-spline 由 P_0, P_1, \dots, P_{n+1} 個控制點及一組基底函數的線性組構成，其定義是：

$$P(t) = \sum_{k=0}^n P_k N_{k,m}(t)$$

其中 P_k 表示第 k 個控制點， $N_{k,m}(t)$ 為 m 階 B-spline

的基底函數 (basic function), 此函數具有下列性質 :

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

$$N_{k,1}(t) = \begin{cases} 1 & t_k \leq t \leq t_{k+1} \\ \text{若} & \\ 0 & \text{其他} \end{cases}$$

$$N_{k,m}(t) = \frac{t - t_k}{t_{k+m-1} - t_k} N_{k,m-1}(t) +$$

$$\frac{t_{k+m} - t}{t_{k+m-1} - t_{k+1}} N_{k+1,m-1}(t)$$

$$6. \sum_{k=0}^n N_{k,m}(t) = 1$$

在本系統的應用上，我們需要一種能表現曲線在空間扭轉的曲線方程式，能滿足這種特性 (torsion $\neq 0$) 的方程式的次數最少是三次，因此我們使用四階 B-spline 函數，並定 t 的區間為 $0 \leq t \leq 1$ 。由此導出的基底函數如下

$$N_{0,4}(t) = \frac{1}{6}(1-t)^3, \quad N_{1,4}(t) = \frac{1}{2}t^3 - t^2 + \frac{2}{3}$$

$$N_{2,4}(t) = \frac{-1}{2}t^3 + \frac{1}{2}t^2 + \frac{1}{2}t + \frac{1}{6},$$

$$N_{3,4}(t) = \frac{1}{6}t^3$$

以下舉一例進一步說明這種三次B-spline曲線的性質

給定 $n+1$ 個有序位置向量（即控制點） Q_0, Q_1, \dots, Q_n ，由這些控制點形成的近似曲線，可由下式表示：

$$P_i(t) = X_0(t)Q_{i-1} + X_1(t)Q_i + X_2(t)Q_{i+1} + X_3(t)Q_{i+2} \quad (2.1)$$
$$i=1, 2, \dots, n-2, \quad 0 \leq t \leq 1$$

$X_0(t), X_1(t), X_2(t), X_3(t)$ 是以 t 為參數的三次多項式。

由於能表現空間曲線的性質的曲線方程式至少是三次，因此我們上述多項式定為三次多項式。首先要求由2.1式表示的曲線必須在座標轉換後形狀仍保持不變。式子2.1中任一段線的矩陣表示式可改寫如下：

$$P(t) = [X_0(t) \ X_1(t) \ X_2(t) \ X_3(t)][Q_0 \ Q_1 \ Q_2 \ Q_3]^T$$

分別考慮曲線的旋轉及平移情況如下。

1. 旋轉：設 R 是 3×3 旋轉矩陣， Q^* 是 Q 經旋轉後的控制點

$$Q^*_0=Q_0R, Q^*_1=Q_1R, Q^*_2=Q_2R, Q^*_3=Q_3R$$

經旋轉後的曲線 $P^*(t)$ 可以下式表示：

$$P^*(t)=[X_0(t) X_1(t) X_2(t) X_3(t)][Q^*_0 Q^*_1 Q^*_2 Q^*_3]^T$$

若 $P^*(t)=P(t)R$ ，則表示 $P(t)$ 轉換成 $P^*(t)$ 後形狀不變

$$\begin{aligned} P^*(t) &= [X_0(t) X_1(t) X_2(t) X_3(t)][Q^*_0 Q^*_1 Q^*_2 Q^*_3]^T \\ &= [X_0(t) X_1(t) X_2(t) X_3(t)][Q_0R Q_1R Q_2R Q_3R]^T \\ &= [X_0(t) X_1(t) X_2(t) X_3(t)][Q_0 Q_1 Q_2 Q_3]^T R \\ &= P(t)R \end{aligned}$$

由此可知由此三次多項式線性組合而成的曲線經旋轉後形狀一定會保持不變。

2. 平移： 設 $M = [m_x m_y m_z]$ 是平移向量

我們希望 $P(t)$ 經平移後形狀仍不變，即 $P^*(t)=P(t)+M$

$$\begin{aligned} P^*(t) &= [X_0(t) X_1(t) X_2(t) X_3(t)][Q^*_0 Q^*_1 Q^*_2 Q^*_3]^T \\ &= [X_0(t) X_1(t) X_2(t) X_3(t)][Q_0+M Q_1+M Q_2+M Q_3+M]^T \\ &= [X_0(t) X_1(t) X_2(t) X_3(t)][Q_0 Q_1 Q_2 Q_3]^T + \\ &\quad (X_0(t) + X_1(t) + X_2(t) + X_3(t))M \\ &= P(t) + (X_0(t) + X_1(t) + X_2(t) + X_3(t))M \end{aligned}$$

如果 $X_0(t) + X_1(t) + X_2(t) + X_3(t) = 1$ 則平移後的曲線形狀可保持不變。

另外我們還希望兩相鄰的曲線段 $P_i(t)$, $P_{i+1}(t)$ 相接處的點具有位置、斜率（一次微分）、曲率（二次微分）均連續的特性，以上條件可列式如下：

$$X_0(t) + X_1(t) + X_2(t) + X_3(t) = 1$$

$$P_i(1) = P_{i+1}(0) \Rightarrow \begin{cases} X_0(1) = 0 \\ X_3(0) = 0 \\ X_1(1) = X_0(0) \\ X_2(1) = X_1(0) \\ X_3(1) = X_2(0) \end{cases}$$

$$P_i'(1) = P_{i+1}'(0) \Rightarrow \begin{cases} X_0'(1) = 0 \\ X_3'(0) = 0 \\ X_1'(1) = X_0'(0) \\ X_2'(1) = X_1'(0) \\ X_3'(1) = X_2'(0) \end{cases}$$

$$P_i''(1) = P_{i+1}''(0) \Rightarrow \begin{cases} X_0''(1) = 0 \\ X_3''(0) = 0 \\ X_1''(1) = X_0''(0) \\ X_2''(1) = X_1''(0) \\ X_3''(1) = X_2''(0) \end{cases}$$

由這16個條件式可解出 $X_0(t)$, $X_1(t)$, $X_2(t)$, $X_3(t)$ 等各三次多項式的所有係數，解之得

$$X_0(t) = \frac{1}{6}(1-t), \quad X_1(t) = \frac{1}{2}t^3 - t^2 + \frac{2}{3}$$

$$X_2(t) = \frac{-1}{2}t^3 + \frac{1}{2}t^2 + \frac{1}{2}t + \frac{1}{6},$$

$$X_3(t) = \frac{1}{3}t^3$$

$$X_0(t) = N_{0,4}(t), \quad X_1(t) = N_{1,4}(t),$$

$$X_2(t) = N_{2,4}(t), \quad X_3(t) = N_{3,4}(t)$$

依上述條件解出的曲線方程式恰為三次 B-spline 曲線。因此我們知道三次 B-spline 曲線具有下列性質：

1. 經座標轉換後曲線形狀保持不變。
2. 曲線上任意兩點間不但斜率連續，曲率也連續。
3. 曲線必落在 Q_0, Q_1, \dots, Q_n 等控制點所形成的控制多邊形之內。這個性質讓使用者能夠利用控制多邊形的形狀預測曲線的形狀，在設計曲線時十分方便。
4. 由於曲線中每一段都是只由四個相鄰控制點所定義，當我們只想修改某一段的形狀時，只需調整該段的控制點，形狀的改變只限於該控制點前後三個控制點所定義的曲線段，其餘曲線段保持不變。 如圖 (2.1)

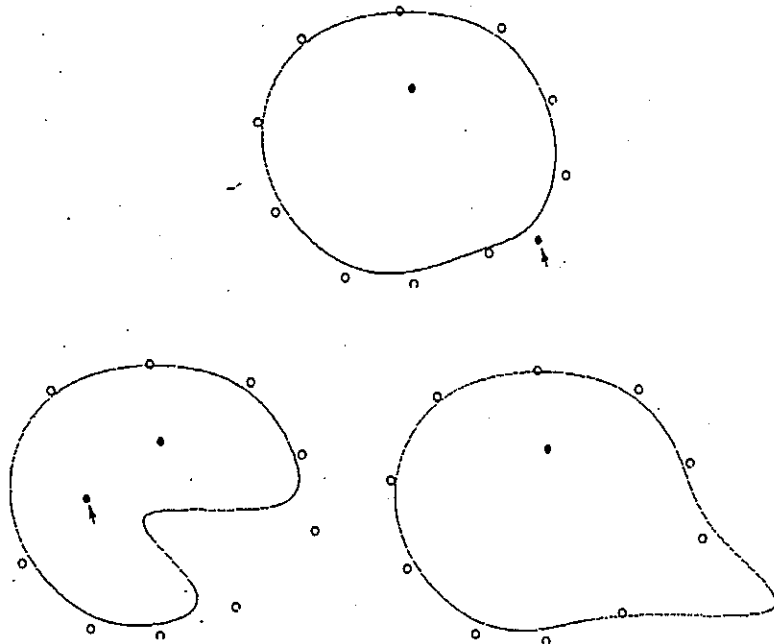


圖 (2.1)

由上圖可知，每移動一個控制點會改變三曲線段的形狀。如果我們想讓改變的範圍更小，可在需要調整的線段上加入更多的控制點，也就是將原有的曲線再細分為更多段，如此每移動一個控制點對曲線段形狀的改變範圍就能更加縮小。值得注意的是，加入控制點後的曲線必須保持原有的形狀。Bohm 於 1980 年提出一演算法 [3] ，可以在不改變 B-spline 原有形狀的前題下加入新的控制點：

： 設一 M 階 B-spline 的 knot sequence 為 ... t_i, t_{i+1}, \dots ，其原有的控制點為 ... Q_i, Q_{i+1}, \dots

$$P(t) = \sum_j N_{j,M}(t) Q_j$$

如果要在 t_i 及 t_{i+1} 間的 t' ($t_i < t' < t_{i+1}$) 上加入一新的控制點 Q ，並保持 B-spline 原有形狀，則控制點須重新計算如下：

$$Q'_j = (1-u_j) \cdot Q_{j-1} + u_j \cdot Q_j$$

其中

$$u_j = \begin{cases} 1 & (j \leq i-M+1) \\ \frac{t'-t}{t_{j+M-1} - t_j} & (i-M+2 \leq j \leq i) \\ 0 & (j \geq i+1) \end{cases}$$

由上式可知，調整後的控制點仍落在原先的控制多邊形上，如下圖(2.2)。

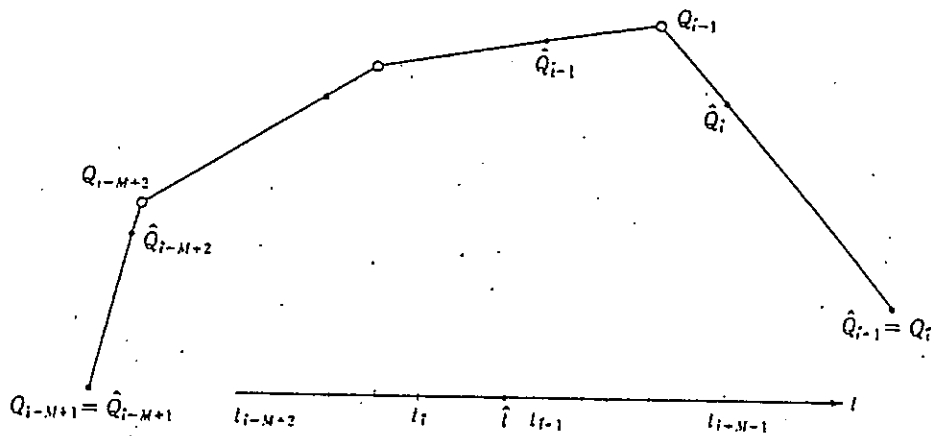


圖 (2.2)

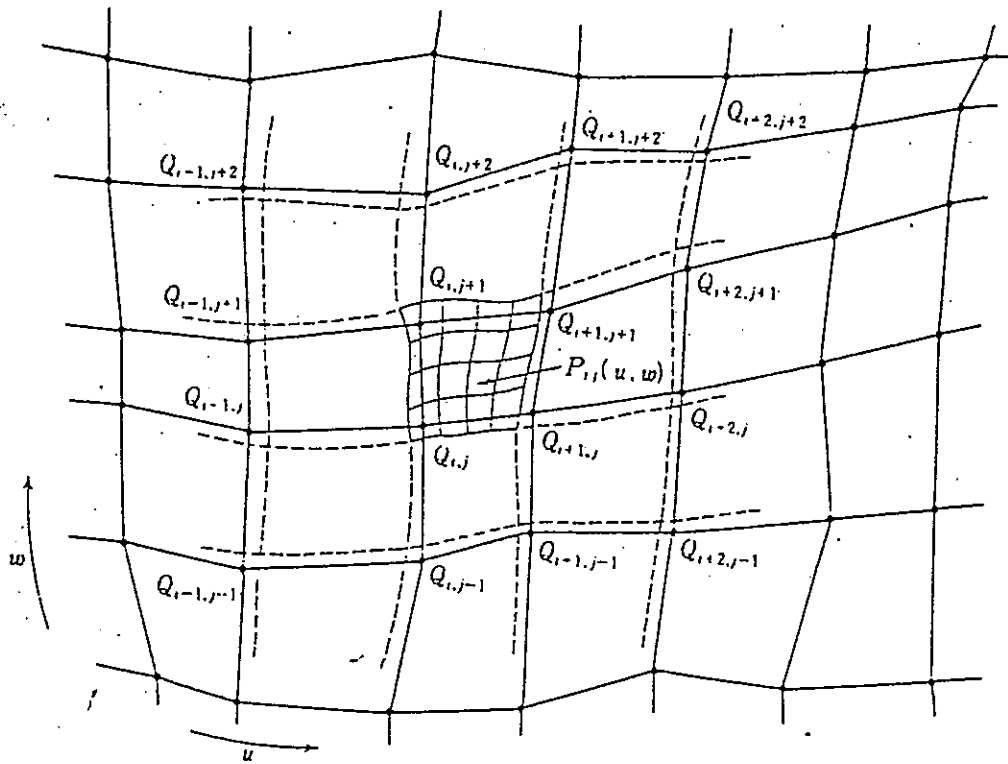
2-1-1 B-spline 曲面

一個Uniform bicubic B-spline 曲面片斷(patch)是由16個控制點及前節導出的基底函數所定義：

$$P(u, v) = \sum_{i=0}^3 \sum_{j=0}^3 Q_{i,j} N_{i,4}(u) N_{j,4}(v)$$

$Q_{i,j}$ 為控制點的三度空間座標。

如圖 (2.3)，控制點通常並不在曲面片斷上。如此產生的曲面具有二次連續 (C^2 continuity) ，及局部控制的性質，只要輸入少數控制點就能產生複雜曲面形狀，非常適合用於描繪三度空間動畫物體。



Uniform bi-cubic B -spline surface patch and surface defining vectors

圖 (2.3)

2-2 掃掠法簡介 [4] [5]

掃掠法的基本觀念是將一平面封閉曲線 (Contour or cross section) 沿著一條軸線 (trajectory or spine) 移動或旋轉而得到物體的外形。根據掃掠方式及軸線定義的不同，可以再細分如下：

1. 平移掃掠 (translational sweeping)：截面沿一條直線移動。如圖 (2.4)

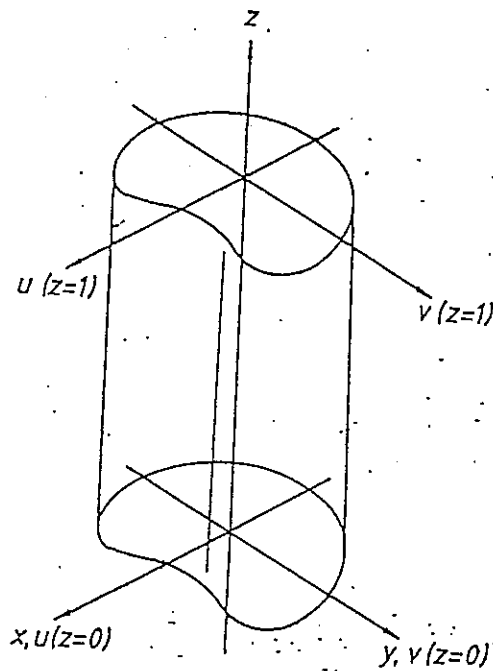


圖 (2.4)

2. 旋轉掃掠 (rotational sweeping) : 截面沿一條直線旋轉。如圖 (2.5)

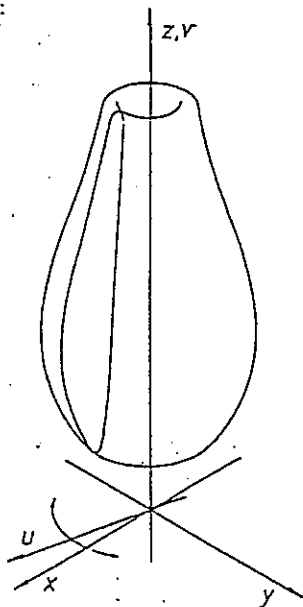


圖 (2.5)

3. 一般化掃掠 (general sweeping) : 截面沿一條三度空間任意曲線移動，移動過程中截面形狀，大小可改變。如圖 (2.6)

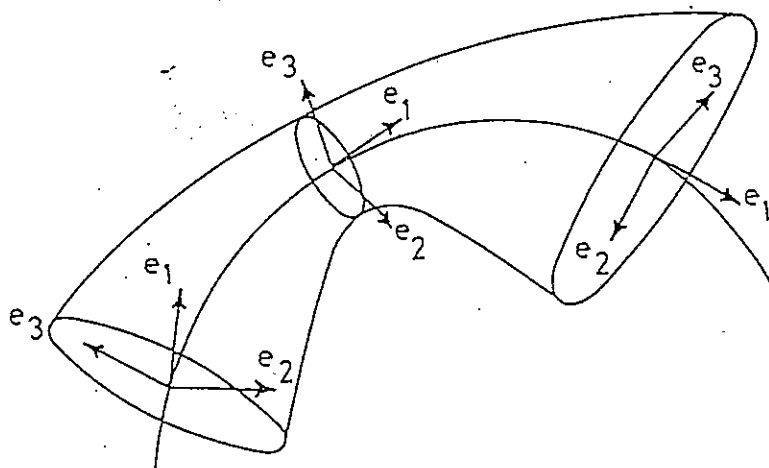


圖 (2.6)

以下詳細說明一般化掃掠方法 [6]。

二度空間封閉曲線 C 定義為：

$$C(v) = (C_x(v), C_y(v)) \quad v_i \leq v \leq v_f, \quad C(v_i) = C(v_f)$$

當參數 v_i 由 v 變動到 v_f ，函數 C_x 及 C_y 分別描繪出封閉曲線對應的 X, Y 座標值。此處 $C(v)$ 可定義為參數化的曲線表示式，如插值曲線或近似曲線。

三度空間軸線 S 定義為：

$$S(u) = (S_x(u), S_y(u), S_z(u)) \quad u_i \leq u \leq u_f$$

$S(u)$ 如同 $C(v)$ 可以是參數化曲線。

接下來 $C(v)$ 須沿 $S(u)$ 掃掠出物體外形。原先定義於 X, Y 平面上的截面必須轉換到三度空間的軸線上。轉換後的 $C(v)$ 上每一點的位置與軸線形狀和軸線在空間中位置有關，也就是說 $C(v)$ 必須由原先定義的 XY 平面上轉換到由 $S(u)$ 所決定的局部座標系 (local coordinate) 上。此座標系必須反映出軸線的形狀而且在軸線上每一點均有定義。一般使用 Frenet frame 作為局部座標系。Frenet frame 是由 $S(u)$ 上任一點的 unit tangent vector (t), unit principal vector (n) 及 unit binormal vector (b) 構成的正交座標系，如圖 (2.7)

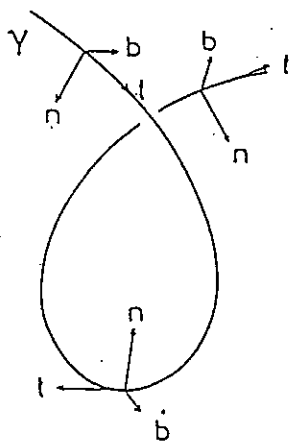


圖 (2.7)

t 指向切線方向， n 指向曲率中心， b 同時垂直 t 及 n 。掃掠的過程就是將 $C(v)$ 的 (x, y, z) 座標轉換成 (n, b, t) 座標再平移到軸線上。最後產生的曲面可表示成：

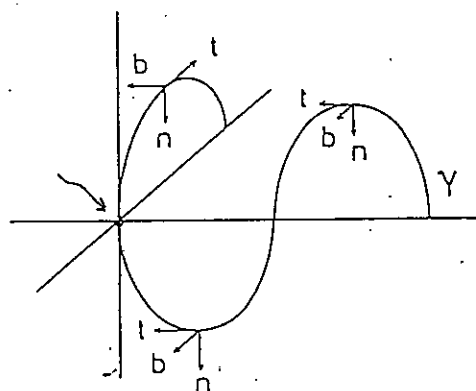
$$T(u, v) = S(u) + C_x(v) \cdot n(u) + C_y(v) \cdot b(u)$$

由上述式子產生的物體，在軸線上任一點截面的形狀都不變(由 $C(v)$ 決定)，為了讓掃掠出的物件外形較富變化，定義兩函數 $F_x(u)$ 及 $F_y(u)$ 分別代表截面在軸線上每一點 x 及 y 方向的Scale factor，用以控制 $C(v)$ 在 x 及 y 方向上的大小變化。加入Scale factor後的一般化掃掠物體表面變成：

$$T(u, v) = S(u) + F_x(u) \cdot C_x(v) + F_y(u) \cdot C_y(v) \cdot b(u)$$

採用Frenet frame有個嚴重的缺點： $S(u)$ 中若包含直線段或有反曲點(inflexion point)，在這些地方曲率為0，則 n 及 b 無定義。為了解決這個問題，學者對Frenet frame 做了如下的修正 [7]：

若 $S(u)$ 為一平面曲線，選擇一垂直於 $S(u)$ 所在平面的單位向量 m 以取代 b 向量， t 仍是切線向量， $n=m \times t$ ，改用 t, m, n 做為軸線的局部座標系。如此一來這個修正後的座標系在 $S(u)$ 上每一點都有定義。採用修正後的局部座標系不但消除了直線段及反曲點的問題，而且由於不需計算每一點的 binormal 向量，處理速度會較快。但是當 $S(u)$ 是由平面曲線和空間曲線組合而成時，兩者相接處的座標系仍不連續。如圖(2.8)，這個問題可用下述方法解決：



圖(2.8)

算出相接處的 binormal 向量及 m 向量之間的角度 θ ，接下來空間曲線上每一點仍使用 Frenet frame，但是須將 Frenet frame 中的 b 及 n 向量以 t 為軸旋轉 θ 角。經此處理 $S(u)$ 整個曲線每一點的局部座標系都有定義，而且各點間的座標系可保持連續性。

上述方法雖然可造出像圖 (2.6) 電話聽筒般具有粗細不同截面的物體，但是各截面的形狀變化是由 F_x , F_y 函數所控制，如果想產生較特殊的外形，如一端圓形一端星狀的話筒， F_x , F_y 函數將變得非常複雜。

1986年 N. Magnenat-Thalmann and D. Thalmann 提出一種結合動畫的“中間圖像法”(in-betweening) 和一般化掃掠法的造型方法，稱為 propagation Control graph (PCG) [8]，物體截斷面的形狀變化不再只靠 F_x , F_y scale function 控制，而是使用主圖像法產生不同形狀、大小的截面間的中間形狀(in-between)讓物體造型的工作更加簡化，而且能造出更富變化的外形。下節先介紹主圖像法，再討論 PCG法。

2-3 中間圖像法與PCG [9]

動畫的原理是利用人類視覺暫留的現象，在一秒鐘內放映20張以上的靜態影像，每張影像中物體形狀及位置都給予細微的變化，使觀者產生“物體正在運動”的錯覺。一部十分鐘長度的動畫需要一萬二千張以上的靜態影像，為了加快繪製的速度，首先由資深作畫員繪出連續動作中

幾個主要轉折動作，稱為主圖像，然後由許多較資淺的作畫員繪出兩主圖像間所有的漸變動作，稱為中間圖像 (in-betweens)。

電腦最初應用在動畫製作上，就是自動由主圖像計算出中間圖像。計算的方法是對兩主圖像中相對應的點位置作內插 (interpolation) 如圖 (2.9)。

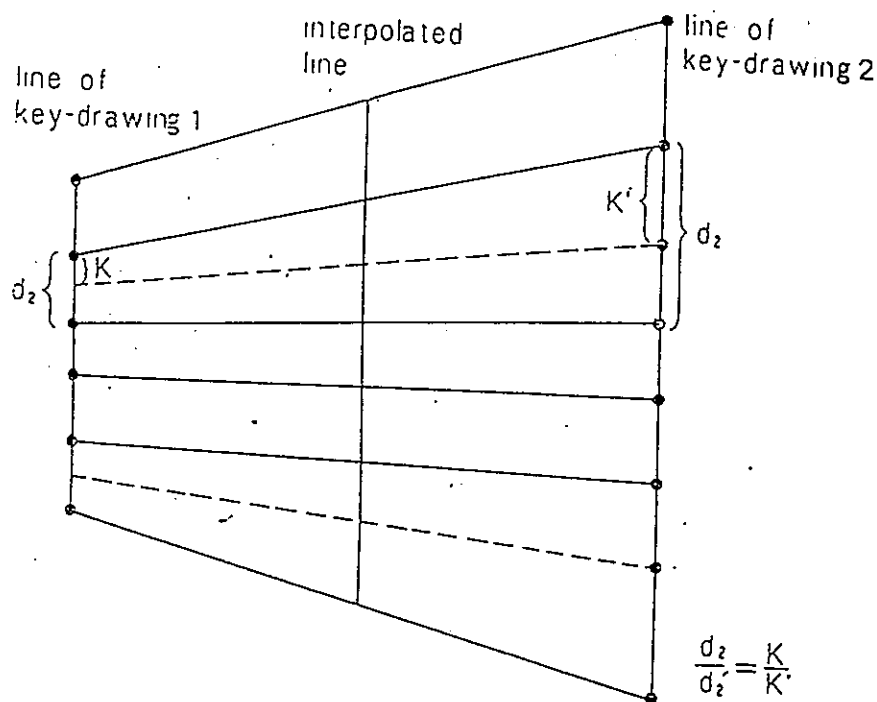


圖 (2.9)

如果兩主圖像上的點數目不相同，必須將適當數目的點加入點數較少的主圖像中，使得所有主圖像的點數均一致。

Burtnyk and Wein於1971年提出如下方法 [9]：

If both figures have the same number of segments
do nothing else (NS1=NS2)

if $NS1 > NS2$ (for $NS2 < NS1$, algorithm is the
same with $NS1, NS2$ interchanged)

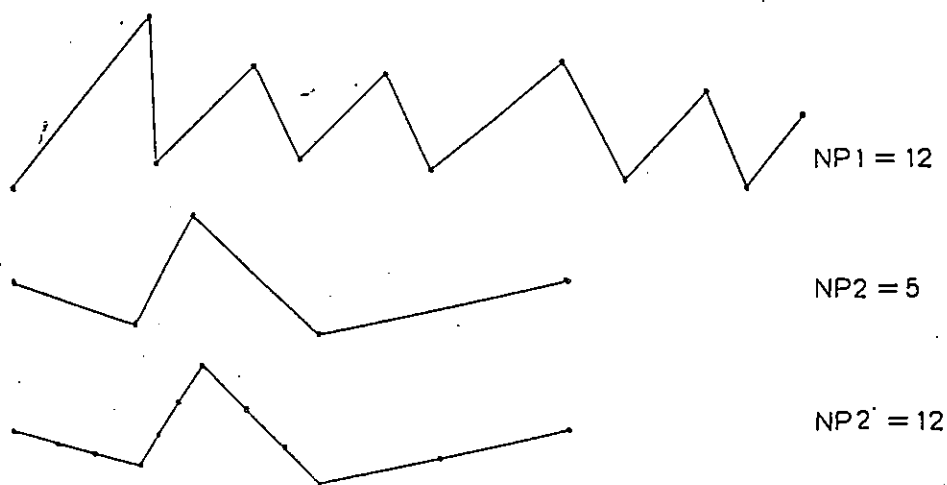
then $RT = (NS1 - 1) \text{ div } (NS2 - 1)$

$RS = (NS1 - 1) \text{ mod } (NS2 - 1)$

RT points are added to the first RS segments

$RT-1$ to the others

上述演算法簡單而快速，所以廣為採用。該方法是將點數較少之主圖像所需增加的點數，平均地加在原有的各段曲線中，如圖(2.10)。Burtnyk的方法原用於解決多邊形的問題，但對於曲線的情況也同樣適用。

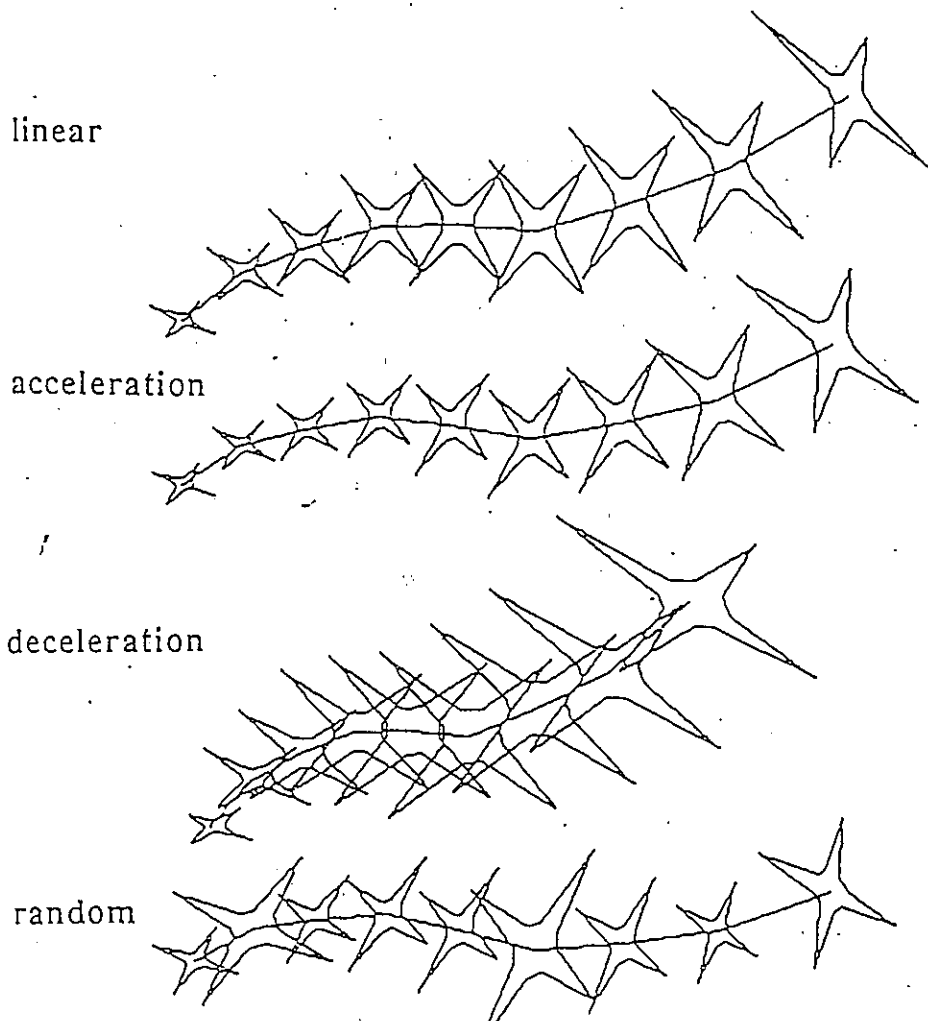


Preprocessing of strokes for in-betweening

圖(2.10)

兩主圖像的點數一致化之後，接著要用內插法求出中間圖像各點的位置。內插的方式可決定兩主圖像間動作的速度，通常使用的方式有四，分別是等速、加速、減速、加速再減速。這些方式統稱為動畫法則(animation laws)如圖(2.11)，構成中間圖像的點位置由下式產生：

$$\text{in-between} = \text{key-frame2-position} * (1 - \text{fract}) + \text{key-frame1-position} * \text{fract}$$

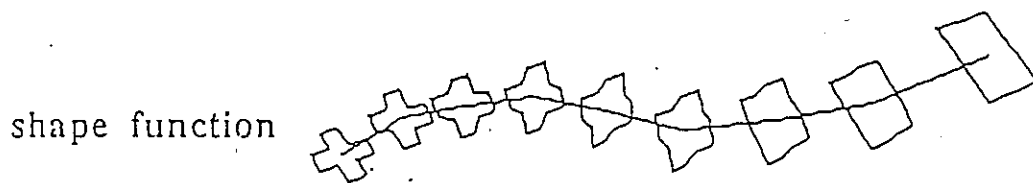


圖(2.11)

PCG 法主要是用於生成Bezier, B-Spline等曲面的方法。構成 propagation control graph 的要件有三項：

1. 一組定義物體截面（二維封閉曲線）的控制點依序連接而成的控制多邊形(control polygon)。
2. 一條三度空間曲線（軸線）。
3. 改變截面特性的演化函數(evolution function)。

演化函數分為大小, 方位(orientation) 及形狀三種。其中大小函數採用動畫法則的四種方式來控制截面掃掠過程中的大小變化, 如圖(2.11)。方位函數使截面在沿軸線掃掠過程中對軸線旋轉, 可造出扭曲變形的外形。而形狀函數允許使用者輸入形狀大小不同的截面, 視為動畫中的主圖像, 利用中間圖像法, 將不同形狀的主圖像經過點數一致化及內插處理後產生漸變的中間形狀, 也就是軸線上兩不同形狀截面間的變形截面。如圖(2.12)

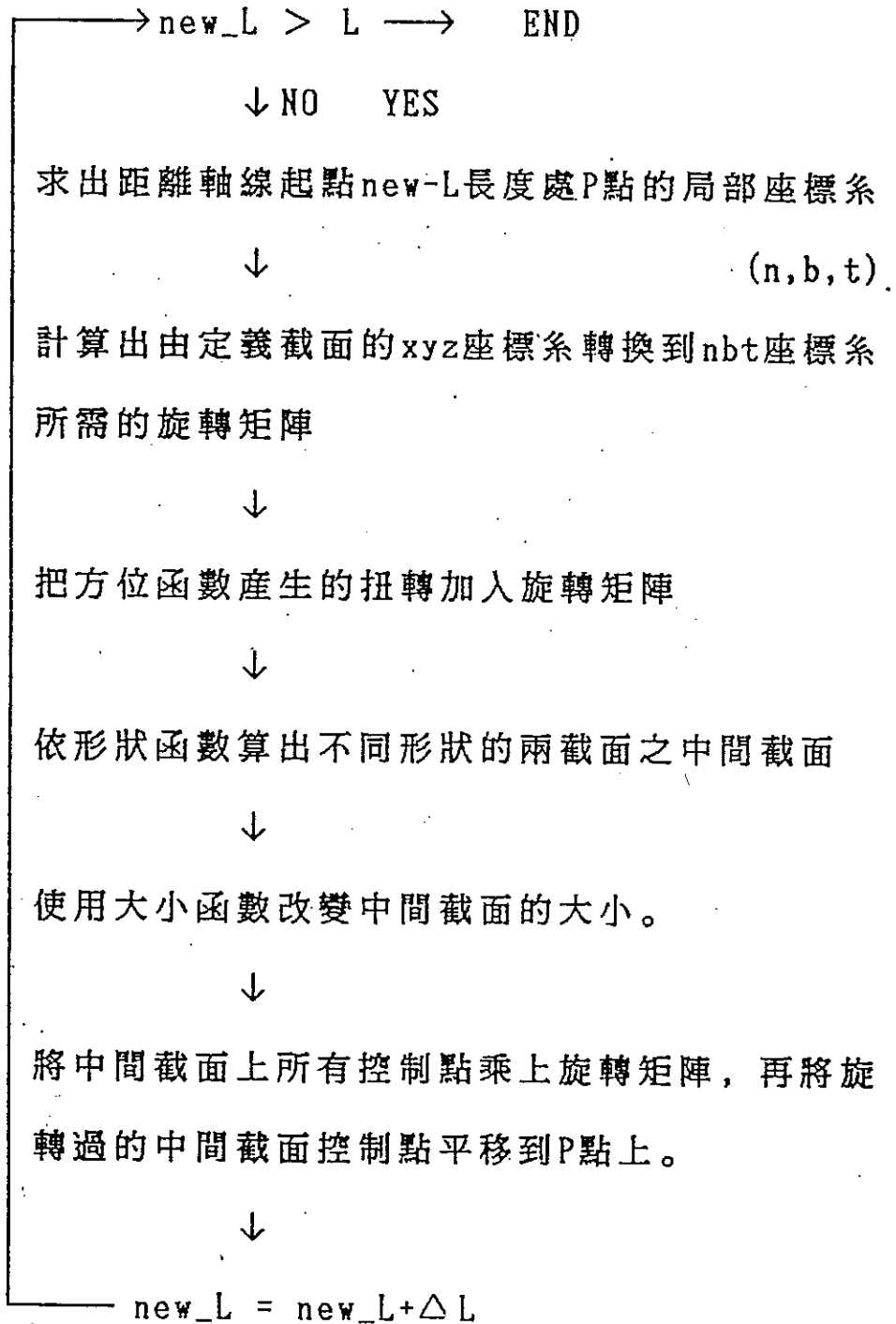


圖(2.12)

整個PCG產生流程如下：

首先算出軸線的總長 L ，決定任二截面間的距離 ΔL

$new_L = 0$



依序連接每一截面的控制點，可形成控制點網路，欲產生實際曲面，還要利用曲面生成公式由各控制點位置及一組基底函數算出曲面上所有點位置。

比較PCG法與上節介紹的一般掃掠法，我們發現兩者基本架構相似，但是PCG加入了動畫中間圖像產生方法的觀念，允許一物體具有數個不同形狀的截面，任二截面間形狀變化利用中間圖像法及動畫法則控制，要產生不同形狀的物體只需給予不同形狀的截面，不像一般化掃掠法必須針對不同形狀的物體設計不同的Scale function，因此不論設計，修改物體外形都較一般化掃掠法方便，而且能產生更富化的複雜外形。本文的造型系統之基本架構即採用PCG法並加以變化而成。

第三章 動畫物體的造型

本系統的物體外形生成部分，可分為三個步驟：1. 截面、軸線輸入及修改。2. 物體表面控制網路(control mesh) 生成。3. 產生四邊形平面組合。以下分別說明各個步驟的作法及功能。

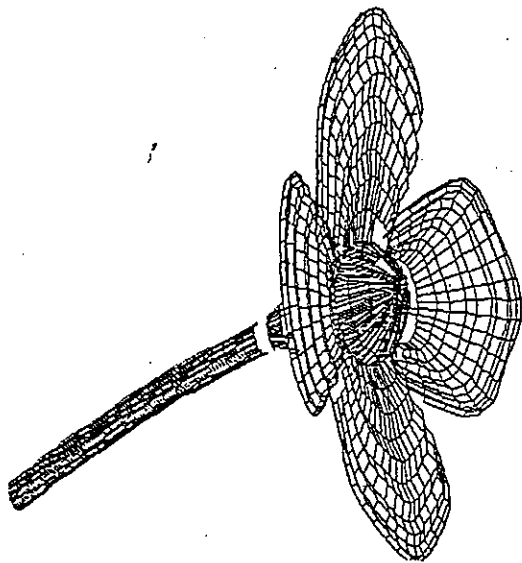
3-1 截面、軸線之輸入及修改。

這個步驟的功能是提供使用者一個交談式(interactive)作畫的環境，使用滑鼠作為輸入的工具，並具備完整的圖形修改功能，讓曲線的產生，修改都能快速方便的進行。

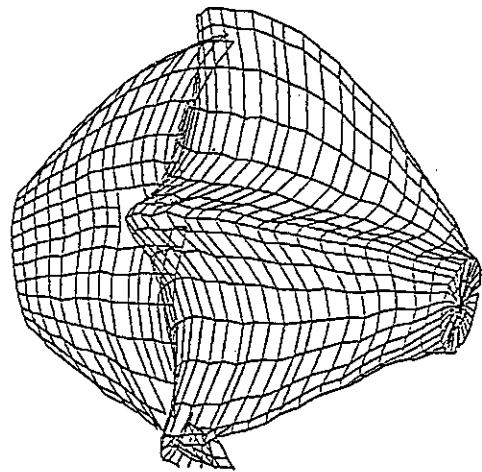
由於三次B-spline曲線有 C^2 連續、局部控制的優點，並能表現空間曲線的特性，因此本系統採用三次B-spline曲線作為定義截面及軸線的曲線。

一般化掃掠法及PCG法的造型模式中截面的定義均侷限於二維封閉曲線，軸線上每一點的橫切面必定是一封閉曲線，無法表現一端為分叉狀或鋸齒狀如花瓣或王冠之類

的物體。如圖(3.1)。為了增強造型系統的表現能力，本系統將截面的定義擴展為三度空間封閉曲線。圖(3.1)是由五平面截面及一立體截面構成，由圖中可看出兩主要截面間的中間截面是如何遂漸改變形狀並構成一朵(四瓣)花的外形。使用三維截面可能會造出自身相交(self intersection)的不正確外形，如圖(3.2)，圖中相交的部分是由於兩相鄰之三維截面在軸線上的位置太過接近所致。雖然使用三維截面有上述缺失，但是只要設計過程中稍微留心即可避免，此外本造型系統還發展了一個檢核程式，能產生如圖(3.1)及(3.2)的物體側視圖，讓使用者能迅速了解物體的大致輪廓並可以輕易發現圖(3.2)中的錯誤情況而加以修正。



圖(3.1)

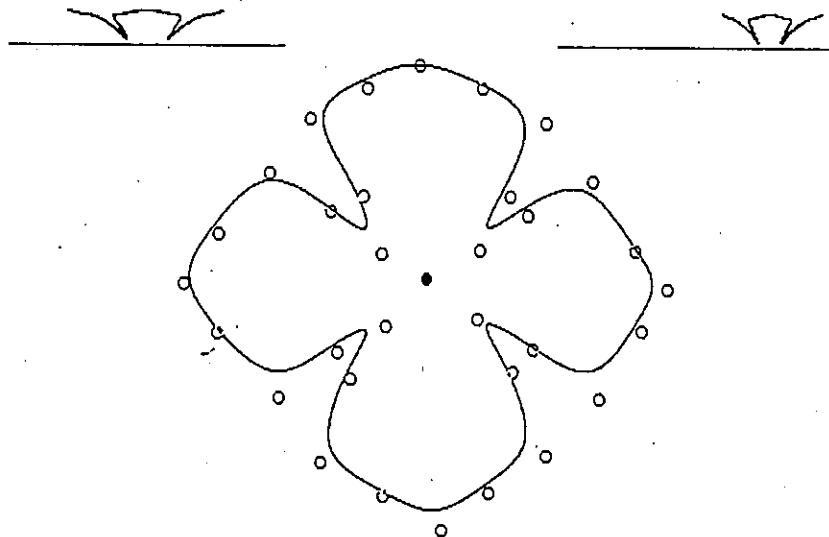


圖(3.2)

截面及軸線的產生可分三步驟：

1. 先將需要的圖形在紙上畫出草圖
2. 在草圖上選定一些關鍵點，如曲線轉折處或突出點附近作為控制點，使用滑鼠在螢幕上適當的地方依序定出這些控制點的位置。如果輸入的曲線是三度空間曲線如三維截面或軸線，螢幕中央會及時顯示曲線在XY平面上的形狀，並在螢幕上方顯示該曲線XZ平面及YZ平面上的側視圖，以幫助了解實際的形狀。如圖(3.3)

FRAME 4



B==== 3D FRAME ====

圖(3.3)

3. 如果螢幕上的圖形與原來構想的圖形仍有出入，可用滑鼠指出需要調整的控制點，加以移動，刪除或是增加新的控制點。

整個輸入過程結束後，因為各截面的控制點數目不一定相同，所以要利用前一章提到的 Bertnyk 演算法將所有截面的控制點數調整到完全一致，至於如何將控制點加入構成截面的 B-spline 中並保持曲線形狀不變，我們採用第二章介紹的 Bohm 演算法來達成。這個步驟的輸出是各截面及軸線控制點的三度空間座標值，以供下個步驟產生中間形狀。

3-2 產生物體表面控制點網路

這個步驟的作用是将前個步驟產生的截面控制點資料視為主圖像，由使用者指定各截面在軸線上的位置及各截面間扭轉角度和各中間截面的大小變化（以動畫法則中的等速、加速、減速、加速再減速表示）。接著產生中間截面的形狀，移到軸線上的正確位置，最後依序連接各截面及中間截面的控制點形成物體表面的控制點網路。有了控制點網路可利用第二章介紹的 bicubic B-spline 表面公式算出物體表面所有點的位置。以下分四部分詳細說明整個流程：

1. 指定各主要截面在軸線上的位置，存入陣列 $pos[i]$ 中，指定任二主截面間的中間截面形狀產生所依循的動畫法則，存入陣列 $law[i]$ 中。如圖 (3.3)

若希望兩主截面間產生扭轉的效果，指定扭轉角度的值。指定截面在軸線上位置時，設軸線總長度為1，位置就用0-1之間的實數表示。

2. 根據任二主截面在軸線上的距離 ($|pos[i+1] - pos[i]|$ 乘上軸線總長)，算出所需的中間截面數。再配合指定的動畫法則，產生所有中間截面的形狀。

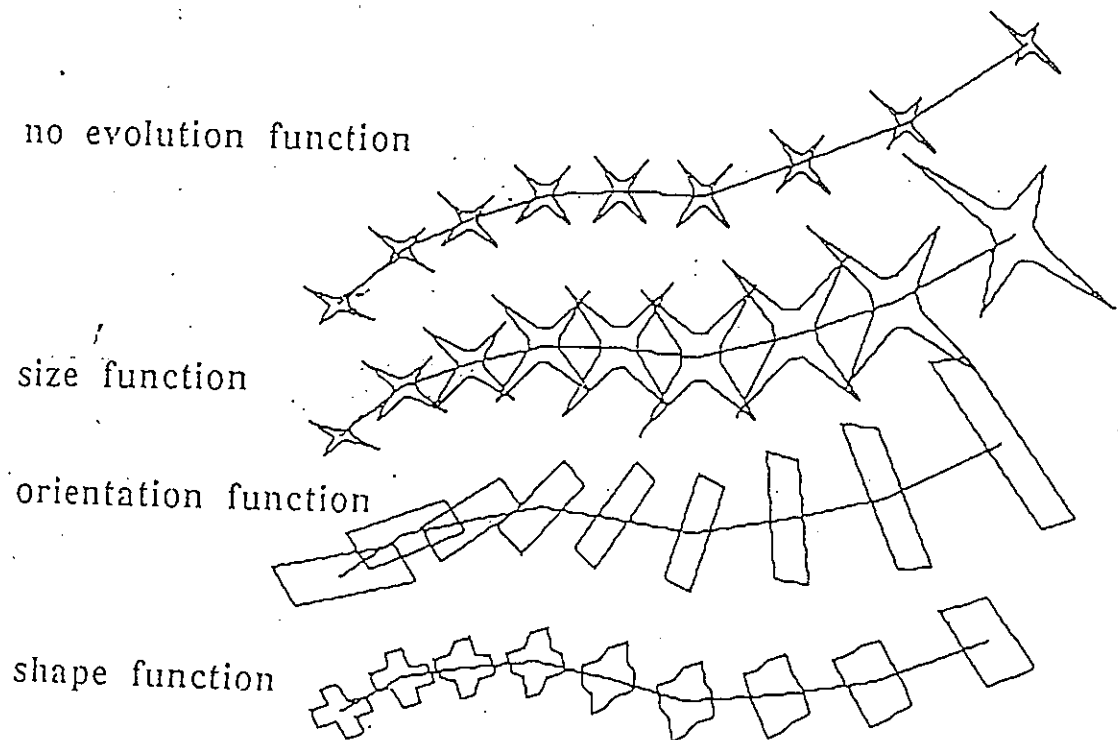


圖 (3.3)

FOR I=first-contour TO last-contour

calculate number of inbetweens needed between
any two contours.

inbetween-amount[I] = | pos[I+1]-pos[I] | ×
spine-length/STEP

FOR J=0 TO inbetween-amount[I]

FOR K=1 TO number-of-control-point-on-a-contour

fract=J/inbetween-amount[I]

CASE law[I]

LINAR:

fract=fract

ACCELERATION:

fract=1-cos(PI×fract/2)

DECELERATION:

fract=sin(PI×fract/2)

ACCELERATION-THEN-DECELERATION:

fract=[1-cos(PI×fract)]/2

inbetween-control-point-position[I]

= contril-piont-position[I]×fract +

control-piont-position[I+1]×(1-fract)

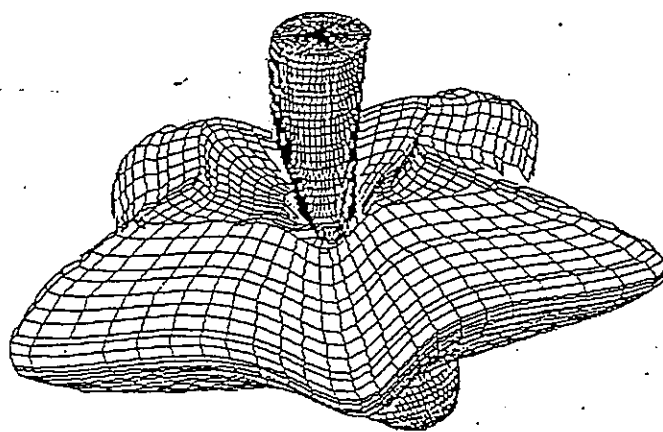
3. 算出每一截面在軸線上的位置，以及該位置上的修改後 Frenet frame 局部座標系 (n, b, t) 。

計算每一截面由 (x, y, z) 座標系轉換到 (n, b, t) 座標系所需的旋轉矩陣。

若二截面間有扭轉，將此扭轉的效應加入旋轉矩陣中。

將所有截面上的控制點座標乘上旋轉矩陣，再平移到軸線上。

如果指定截面位置時有 $\text{pos}[I+1] < \text{pos}[I]$ 的情形，表示掃掠的過程不是單向進行，物體的外形必定會發生疊套或內凹的現象，如圖(3.5)，這種功能使得造型系統的表现能力更加強大。



圖(3.5)

4. 將所有截面的控制點依產生的順序連接而成控制點網路，有了各控制點間連接的關係，我們才能正確的找出相鄰16個控制點代入bicubic B-spline曲面公式以算出曲面的形狀。

3-3 產生四邊形平面組合

由上節所述可知本造型系統能夠生成 bicubic B-spline曲面，但是由於可供使用的顯示及著色軟體只能接受多邊形平面組合的表面模式，所以最後必需由控制點網路產生平面組合的物體表面。因為物體上所有截面的控制點都已求出，我們可以利用這些控制點畫出構成截面的封閉B-spline曲線。在曲線等距離取數點，再把相鄰截面上相對應的點連接起來就能產生物體表面的平面組合。

如果在B-spline曲線上取的點數不夠多，而物體表面形狀較複雜（有些部分曲率變化太大），造出的平面組合可能無法忠實的反映物體實際的外形。因此截面曲線上取點數必須視為可調整的參數，由使用者依據物體表面複雜的程度來決定多邊形平面組合需分割到多細，如圖(3.7)

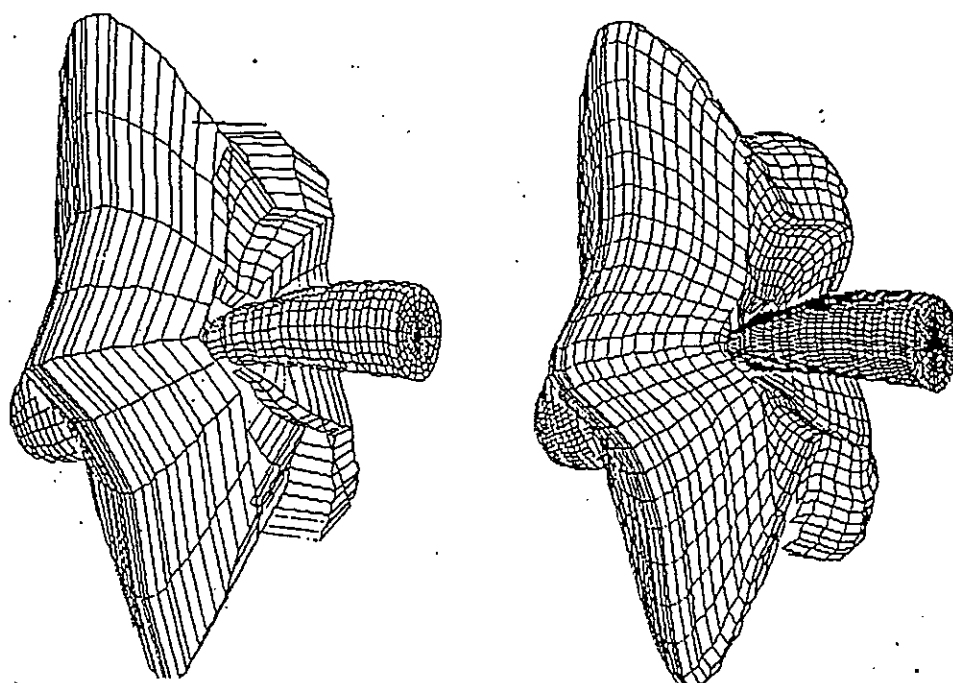


圖 (3.7)

本造型系統的輸出是所有的多邊形平面的端點位置座標，以及每個多邊形是由那些端點組成的描述，如圖 (3.8)

```

POINT    <<3,0,1>>
POINT    <<5,0,1>>
POINT    <<6,0,5>>
POINT    <<4,3,6>>
POLYGON  1, 2, 4
POLYGON  2, 3, 4
POLYGON  1, 3, 4
POLYGON  1, 2, 3
    
```

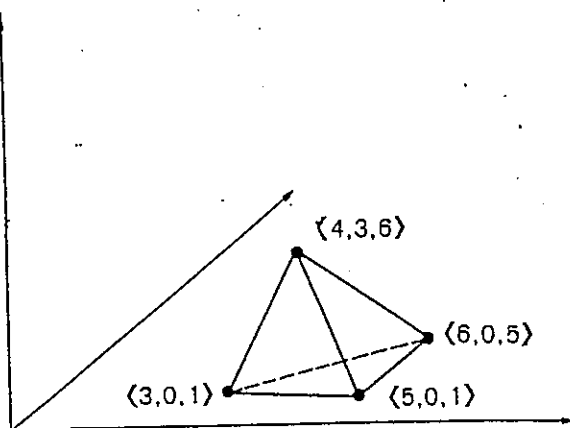
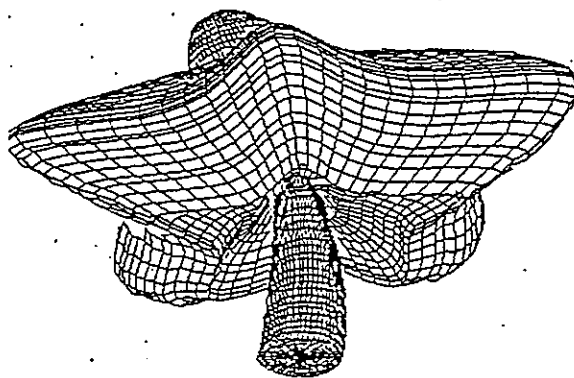


圖 (3.8)

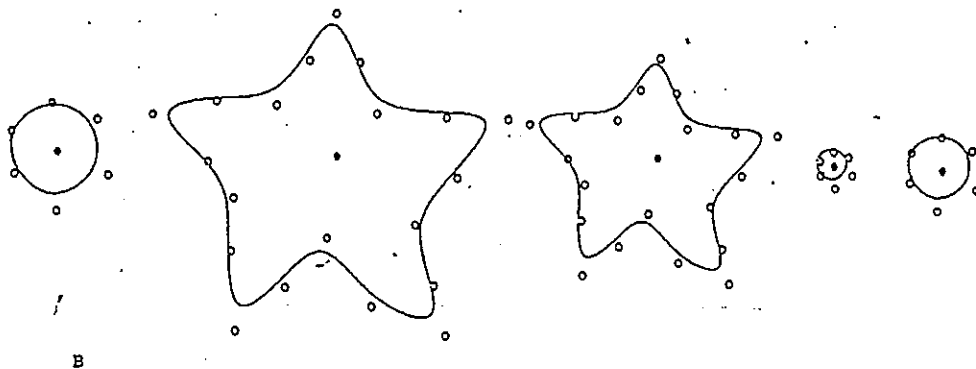
3-4 圖例說明

本節按照上述幾個步驟依序列出物體造型各階段的螢幕輸出：

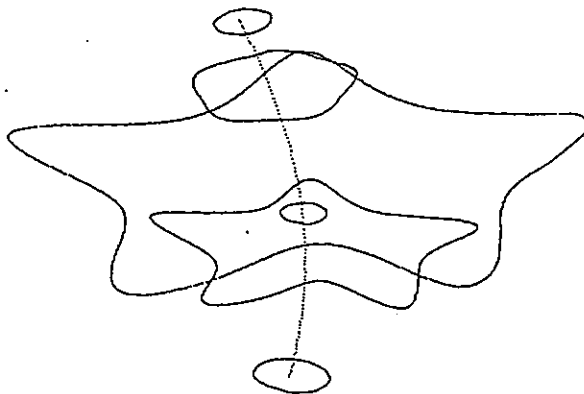
(1) 星形香菇



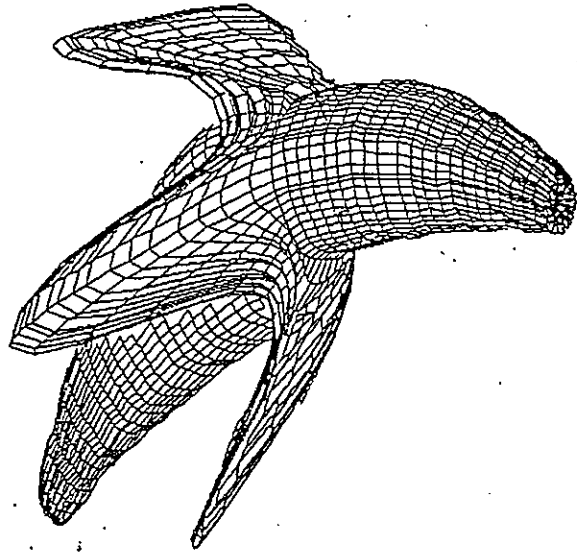
截面設計



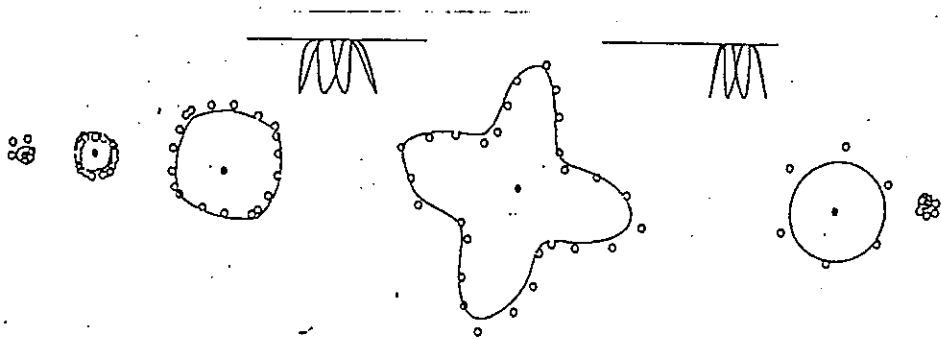
軸線上位置



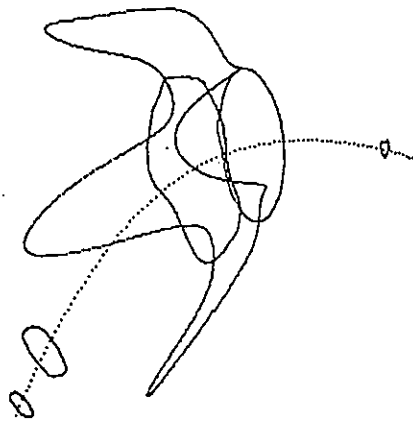
(2) 香蕉



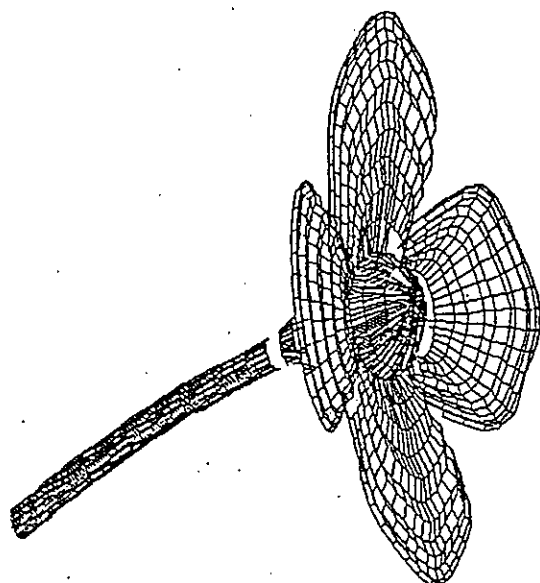
截面設計



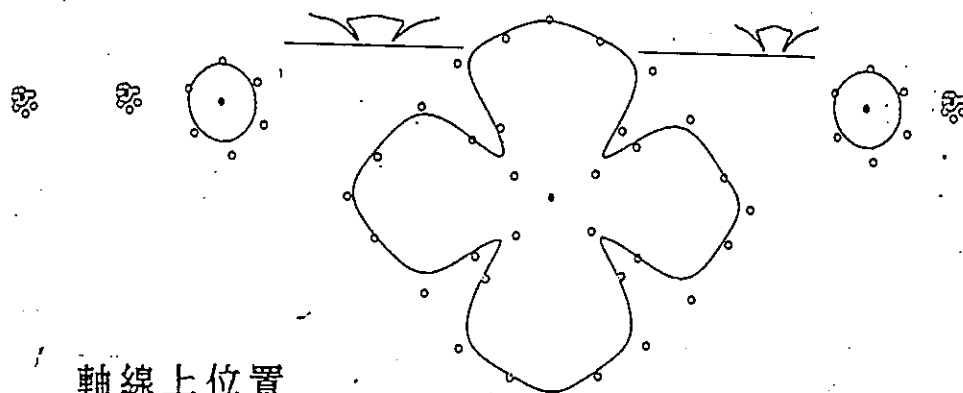
軸線上的位置



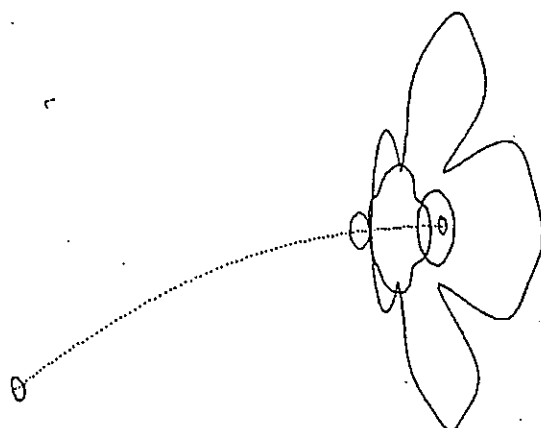
(3) 花



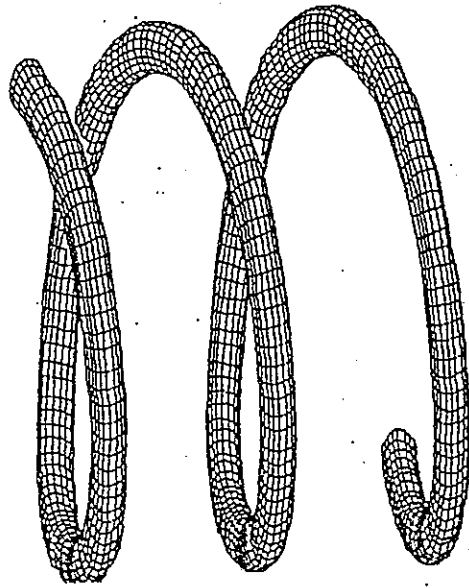
截面設計



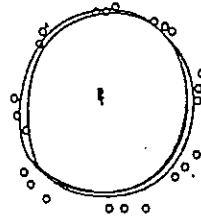
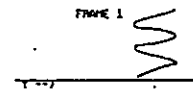
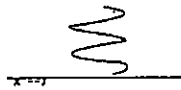
軸線上位置



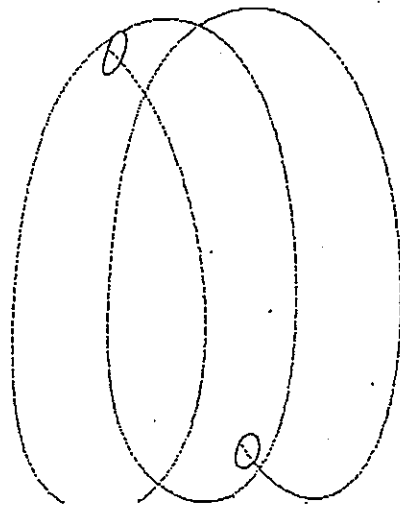
(4) 彈簧



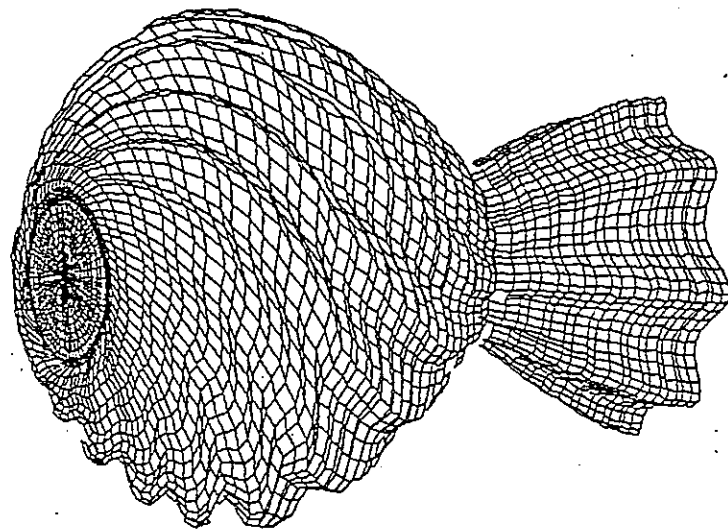
截面設計



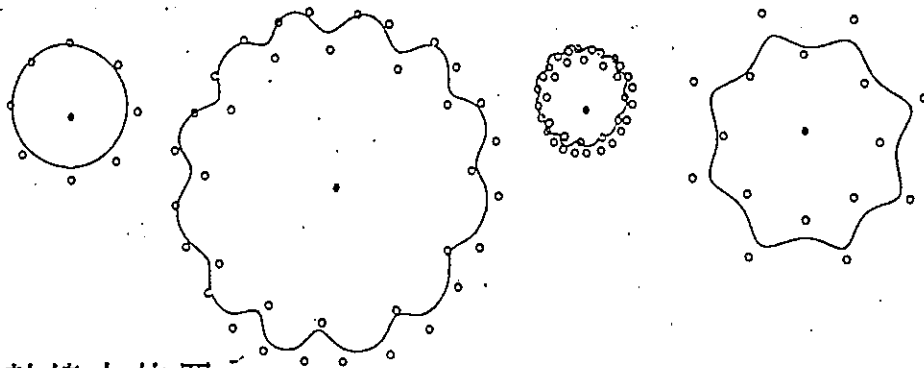
軸線上位置



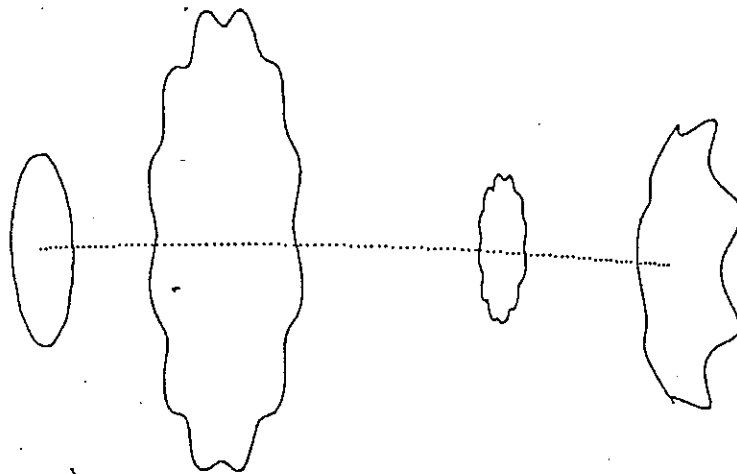
(5) 螺紋花瓶



截面設計



軸線上位置



第 四 章 動 畫 物 體 的 實 體 紋 理 處 理

物體的視覺特性除了外形，還有表面的紋理。所謂紋理泛指物體表面上形狀及顏色的重複樣式 (pattern)，例如木製傢俱上的年輪樣式，橘皮表面的細微而重複的凹凸樣式。同樣一個電腦畫出的球體，未經紋理處理前，看起來只有單一色澤而且顯得非常光滑，像是塑膠或金屬製品。如果加上不同層次的綠色條紋，看起來就像是西瓜；加上分布均勻大小一致的凹洞，就具有高爾夫球的外觀。為了使電腦繪製的物體更逼真，除了研究型造技術，紋理處理也是重要課題。

4 - 1 產生紋理方法介紹

在電繪圖領域中製作紋理的方法可大致分為三類：

1. Texture mapping [11]

1974年由 Catmull首先提出這種方法。處理的過程是將數位化的二維紋理樣式對應到物體的三維座標，也就是

將平面紋理貼到立體的物體表面上。紋理樣式由照片經數位輸入裝置掃描進入電腦，或是經由數學函數直接產生。設紋理樣式定義在 (u, v) 二維座標系，則紋理可視為一函數：

$t = T(u, v)$ t 值代表顏色或透明度等 Shading 係數而物體定義於 (X, Y, Z) 三維座標系。首先須求得二座標系的對應關係 N ，使得 $(u, v) = N(X, Y, Z)$ 如圖 (4,1)

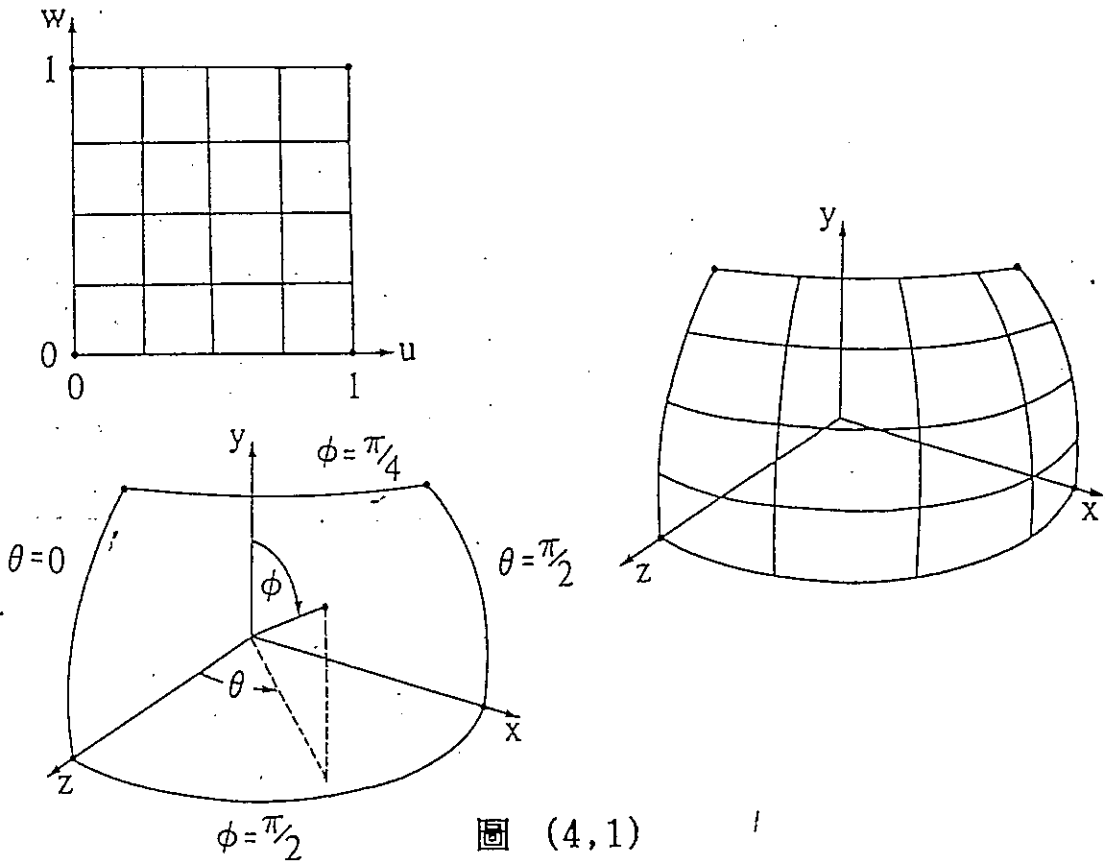


圖 (4,1)

欲求出物體上任一點的 shading 係數，

$$t = T(N(X, Y, Z))$$

2. Bump mapping, 或稱 normal perturbation [12]

利用電腦替物體著色 (shading), 必須考慮光源照射到物體時, 因光線在物體表面各點的入射角度不同而使得物體表面產生色彩明暗層次變化的情形。由光學的散射定 散射定理 (Lambert's law) 得知物體上任一點的散射光度 (deflection intensity) 與該點的散射係數, 該點顏色、光源顏色及入射角度都有關, 如圖 (4.2)

$$I_d = K_d \cdot C_s \cdot \sum_{j=1}^{M_s} I_{Lj} \cdot (N \cdot L_j)$$

I_d : 物體表面某一點的散射光度

K_d : 該點的散射係數

C_s : 該點的顏色

N : 該點的法向量

I_{Lj} : 入射光顏色

L_j : 入射光方向向量

M_s : 入射光數目

圖 (4.2)

如果入射方向與法向量夾角愈小, 表示光線直射, 物體在該點上的散射光度愈大, 也就是愈亮。反之, 夾角愈大表

2. Bump mapping, 或稱 normal perturbation [12]

利用電腦替物體著色 (shading), 必須考慮光源照射到物體時, 因光線在物體表面各點的入射角度不同而使得物體表面產生色彩明暗層次變化的情形。由光學的散射定散射定理 (Lambert's law) 得知物體上任一點的散射光度 (deflection intensity) 與該點的散射係數, 該點顏色、光源顏色及入射角度都有關, 如圖 (4.2)

$$I_d = K_d \cdot C_s \cdot \sum_{j=1}^{M_s} I_{Lj} \cdot (N \cdot L_j)$$

I_d : 物體表面某一點的散射光度

K_d : 該點的散射係數

C_s : 該點的顏色

N : 該點的法向量

I_{Lj} : 入射光顏色

L_j : 入射光方向向量

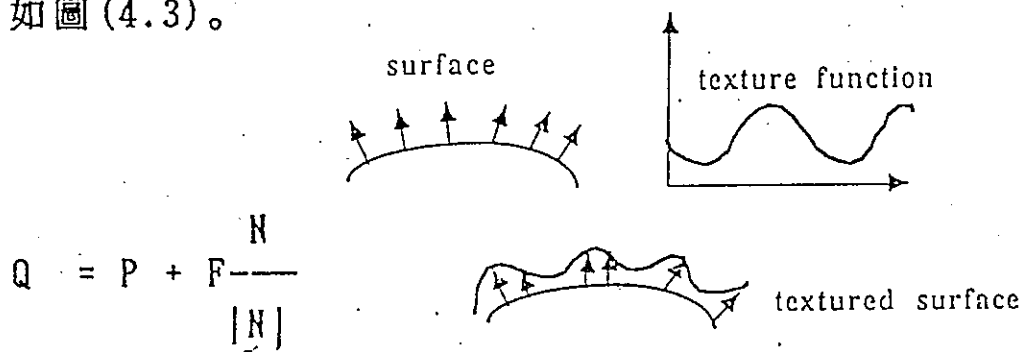
M_s : 入射光數目

圖 (4.2)

如果入射方向與法向量夾角愈小, 表示光線直射, 物體在該點上的散射光度愈大, 也就是愈亮。反之, 夾角愈大表

示光射斜射，物體在該點就愈暗。如果將物體表面每一點法向量的大小及方向予以適度的擾亂 (perturbation)，則經過著色處理後的物體不再只有均勻漸變的明暗度，而會產生凹凸不平皺摺紋理的效果。

1978年 Blinn提出擾亂參數化表面法向量以獲得皺摺效果的方法 [13]。Blinn針對參數化的表面 $P = (X, Y, Z)$ $X = x(u, v)$, $Y = y(u, v)$, $Z = z(u, v)$ ，使用一擾亂函數 $F(u, v)$ 來表示對 (u, v) 點上法向量 N 之 u, v 方向上的改變，如圖 (4.3)。



P : 原來的物體表面

Q : 經擾亂法向量後的物體表面

圖 (4.3)

在著色處理時，並不需要算出擾亂後的表面所有點的位置，只要將經擾亂後的新法向量代入上述 Lambert's law，即可求得經擾亂後的散射光度。

3.Solid texturing [14] [15] [16]

texture mapping 中紋理函數定義在二度空間，而solid texturing 方法中紋理函數定義在三度空間。

$t = T(X,Y,Z)$ solid texturing

$(u,v) = N(X,Y,Z)$ texture mapping

$t = T(N(X,Y,Z))$

實體紋理函數可用週期性函數構成，使其在空間中任一點均有定義。因此在 solid texturing方法中，不需考慮紋理樣式與物體間座標系的對應。使用 texture mapping方法，紋理樣式及物體表面必須切割成相等數目的片斷面 (patch)才能造成正確的對應。當物體外形十分複雜時，物體表面切割出的相鄰 patch 形狀、大小可能相差極大，造成紋理在patch 相接處發生不連續和扭曲的現象，使得物體表面看起來較不自然，如圖(4,4)。Solid texturing 方法可避免這類的問題。

Solid texturing 特別適合處理如原木、大理石等非均質性、具內部紋理的物體。但是如果將文字、標誌、人像等具體圖形貼到物體表面，仍以texture mapping 方法較適合。

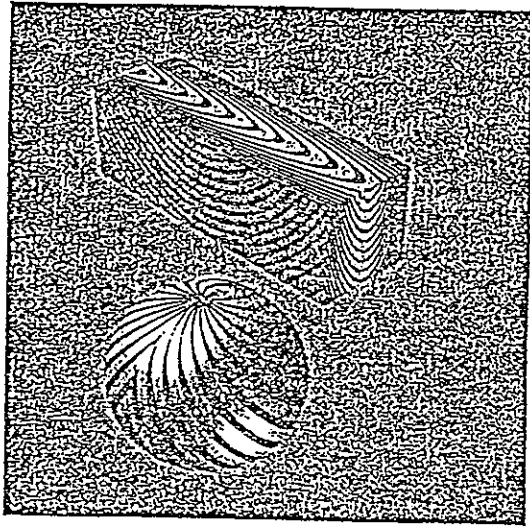


Figure 1

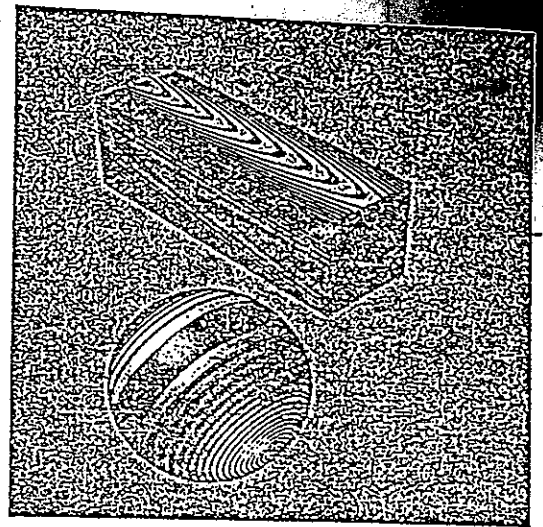


Figure 2

圖(4,4)

4 - 2 變形物體的實體紋理處理

Solid texturing 使用在動畫中的物體時會發生下列問題：如果把實體紋理函數的座標系定義在絕對座標系 (scene coordinate system)，當物體在此座標系內移動或轉動時，物體表面上各點的座標也會隨之改變，因此相同形狀的物體在不同位置上會有不同的紋理，如圖 (4.5) 造成視覺上的不一致。如果將實體紋理函數的座標系定義成相對於運動中物體的相對座標系，當物體運動時此座標系也跟著物體相對運動，則可保證運動中物體紋理的一致性

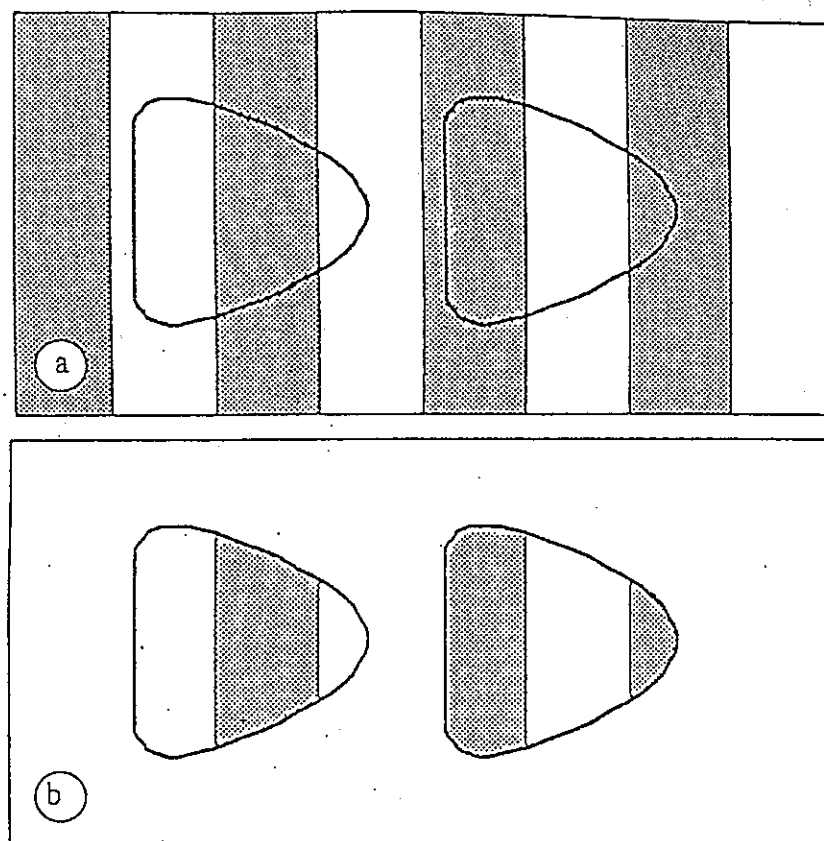


圖 (4.5) a) 同一物體在不同位置

b) 當物體移動時表面紋理會隨著改變

。但是當物體本身發生變形，如彎曲、扭轉時，上述紋理不一致的現象仍然會發生。以下舉一實例說明：我們想利用 solid texturing 造出一尾具有木質紋理的魚，並使用相對座標系定義實體紋理函數。在游動過程中魚的尾部會左右擺動，也就是說魚體本身發生變形，這時魚尾部分的紋理也會隨著擺動而改變。由此觀之，相對座標系在動畫物體變形時已不適用。要處理這種情況，我們需要一個物體內部的座標系，能隨物體變形而改變。以保證變形時紋

理的一致性。本造型系統採用 PCG法塑造物體的外型，首先在 XY 平面上定義主要截面的形狀，利用中間圖像法產生所有的中間截面，再將之轉換到軸線上各自的局部座標系上。物體的變形完全由改變軸線形狀或指定截面間扭轉角度達成，所有截面大小、形狀及在軸線上位置保持不變。針對這種方法造出的物體，我們可將solid texturing 的內部座標系訂成 (n, b, L) 。其中 n, b 是局部座標系中與軸線的切線向量垂直的兩個向量，而 L 是距離軸起點的長度與軸線總長度的比值，也就是在軸線上的位置。因為物體變形只是改變軸線的形狀，所有截面的大小、形狀（截面上各點相對於 n, b 的局部座標）不變，在軸線上的位置（ L 值）也不變，所以一但物體外形造好後，不論如何彎曲、扭轉，表面各點的 (n, b, L) 內部座標值不會隨之改變，因此其紋理不論運動或變形都能保持一致。

4 - 3 實體紋理函數

根據上節的討論，我們在 (n, b, L) 內部座標系內定義了一些實體紋理函數應用在造型系統產生的物體上，以下分別介紹：

1. 木紋函數 wood(n,b)

木紋可視為木材截面上的一組同心圓環所構成，而且具有越接近圓心越密集，越遠離圓心則越疏散的特性。

wood(n,b)

```
{ radius =  $\sqrt{n^2+b^2}$ 
  index = radius / ring_width
  dark_area = 0
  while (index > dark)
    dark_area = dark_area + increment
    ( increment = 0,1,2,3,4,5,.....)
  IF index = dark_area
  THEN return dark_brown
  EISE return light_brown
}
```

2. 斑點紋理 spot(n,b,L)

在物體四周放置形狀、大小、顏色各不相同的球體：

$(n-n_0)^2 + (b-b_0)^2 + (L-L_0)^2 = r^2$ 其中 n_0, b_0, L_0 是由亂數產生，在計算物體上某點的顏色時，將該點的內部座標值代入上式 $(n-n_0)^2 + (b-b_0)^2 + (L-L_0)^2 - r^2$ 計算，所得的值如果小於或等於零，表示該點在球體之內，該點顏色

就取這個球體的顏色。如果所得的值大於零，表示該點在這個球體之外，則再檢查是否在其他球體內部。如此進行下去，則可產生具有不規則形狀斑點的表面紋理。斑點的疏密及大小可依使用者給的參數控制，在應用上更加富有彈性。

3. 橫向條紋 cross(L)

這種紋理只與軸線上位置有關，可由使用者指定軸線上某幾個位置的顏色，這些位置之間各段的顏色可由使用者決定是採用各段端點的顏色或是由兩端點顏色內插而產生漸層的效果。

4. 混色函數

經上述三種紋理函數產生的色彩變化如果仍嫌單調，可使用這個函數加以擾亂，使得物體表面色彩有不規則變化。

$$I_c = T(n, b, L) \cdot P + S \cdot (1 - P)$$

$T(n, b, L)$: 擾亂函數，傳回 0-1 之間的數

S : 物體某一點上的顏色

P : 決定受擾亂的程度之加權量，越接近 1 表示擾亂程度越大。

I_c : 經擾亂後的該點顏色

至於擾亂函數，我們選擇 Gardner 於 1984 年提出的公式 [17]

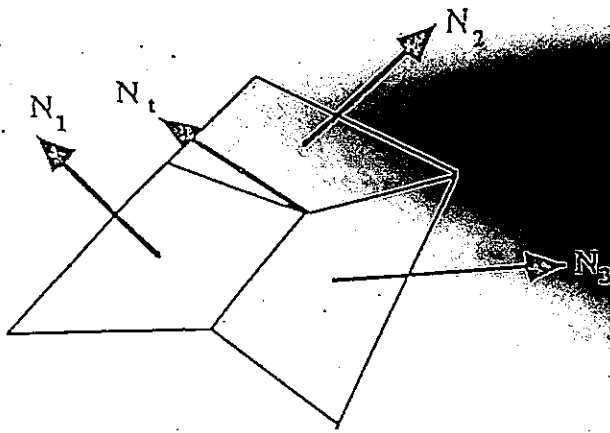
$$T(X_{\theta}, Y_{\theta}, Z_{\theta}) = \sum_{i=1}^n C_i [\sin(W_i X_{\theta} + PX_i) + 1] / 2 * \\ \sum_{i=1}^n C_i [\sin(W_i Y_{\theta} + PY_i) + 1] / 2$$

$$PX_i = \sin(Y_i), PY_i = \sin(X_i);$$

PX_i 及 PY_i 代表相位移函數 (phase shift functions)，用來增加紋理的自然感，以免紋理太規則和有格子狀的情形。

4 - 4 實體紋理產生系統

本系統中實際執行隱面消除及光度計算工作的是美國科羅拉多大學發展出的 Animator 軟體。該軟體的表面著色處理 (surface shading) 採用 Gouraud shading，所有的物體表面在著色前須先轉換成平面多邊形組合的形式，每一平面多邊形中每一點的顏色是由多邊形端點 (vertex) 的顏色作兩次內插求得，如下圖 (4,6) 所示：



Calculation of the vertex normal: $N_t = \frac{N_1 + N_2 + N_3}{3}$

圖 (4,6a) 計算端點法向量 :

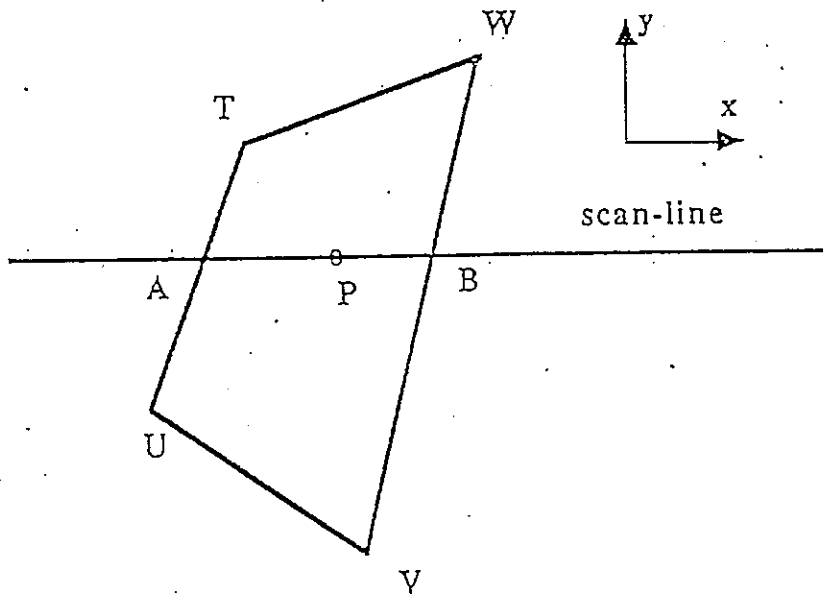


圖 (4,6b) 用內插法計算多邊形上每一點的光度

Intensity interpolation (Gouraud interpolation)

$$I_A = \frac{I_T(Y_A - Y_U) + I_U(Y_T - Y_A)}{Y_T - Y_U}, I_B = \frac{I_W(Y_B - Y_V) + I_V(Y_W - Y_B)}{Y_W - Y_V}$$

$$I_P = \frac{I_A(X_B - X_P) + I_B(X_P - X_A)}{X_B - X_A}$$

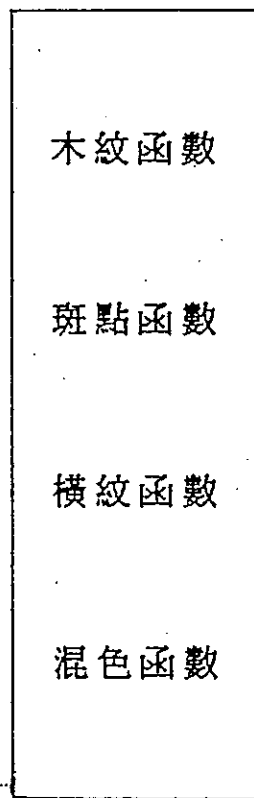
構成多邊形的所有端點之顏色可由使用者自行指定。但由於 Animator 本身並沒有支援實體紋理的功能，因此本系統只將構成截面之封閉多邊形端點（由造型系統產生）的

顏色算出，再由 Animator 以線性內插法算出其餘各點的值這種作法有缺點：當構成物體表面的多邊形組合切割得不夠細時，產生的紋理會有嚴重的鋸齒狀邊緣，使得紋理的效果大打折扣。但也有優點：由於只需各主要及中間截面上的端點顏色，使得計算上的時間大量減少。

定義紋理函數所需的內部座標系 (n, b, L) 在造型系統執行時即可獲得，所以此系統可並入造型系統中，但是為了利於觀察同一物體使用不同實體紋理函數所獲致的效果，以及讓同一物體也能同時具備不同紋理函數所構成的組合紋理 (composite texture)，因此將紋理系統獨立發展，做為整個造型及紋理系統的后處理器 (post processor)。本紋理系統的輸入及輸出如下圖所示：

由造型系統產生之各截面多邊形
的端點位置，及在軸線上的位置

← 使用者輸入的各種參數



實體紋理系統

各多邊形端點的顏色

第 五 章 系 統 架 構 及 流 程

5-1 硬 體 架 構

本動畫物體的造型及紋理系統是在中研院資訊所的VAX 750及SUN4工作站上發展的。使用一台配備EGA卡及滑鼠的PC AT 進行截面、軸線的輸入及修改，產生出的物體多邊形平面組合以及著色處理後的彩色顯像則以 E&S 公司的PS390顯示。

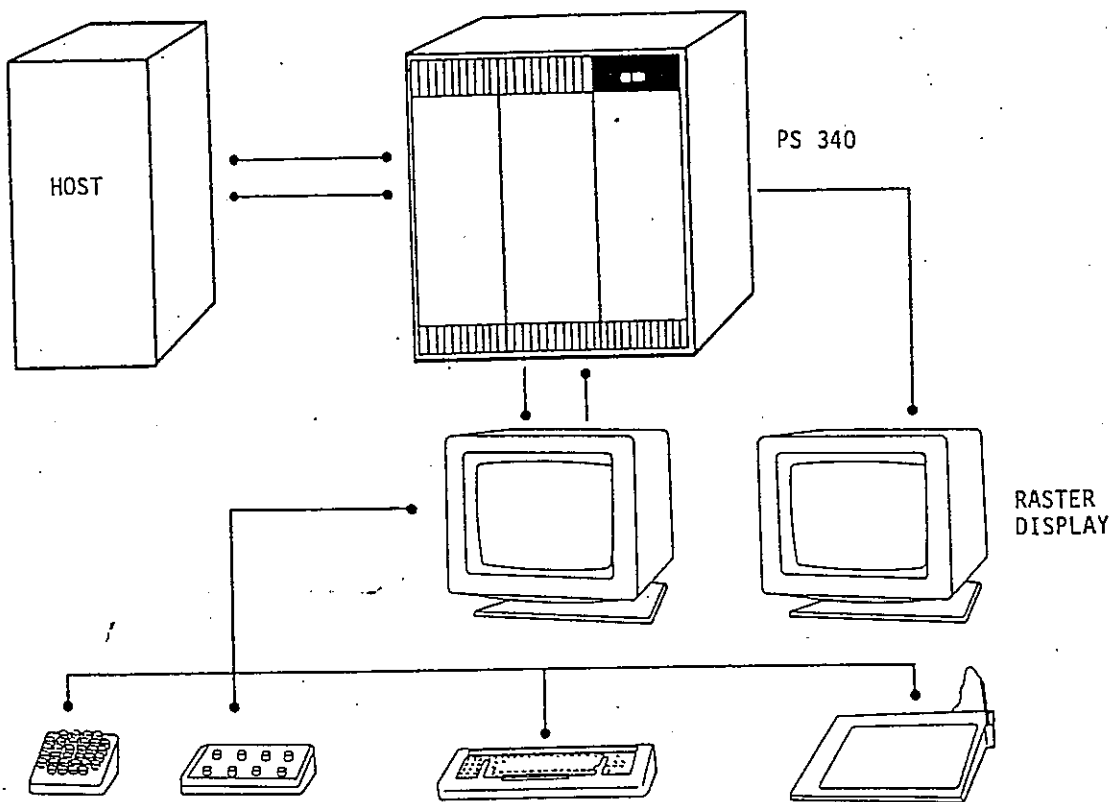
5-2 軟 體 架 構

物體的多邊形平面組合的線框式顯示是利用美國海軍發展的一套繪圖軟體USNA中的THREED程式顯示在PS390上。

著色軟體使用美國柯羅拉多州立大學發展的Animator系統，產生解析度為 600×512 的 raster file。Animator的光度模式(intensity model)是以Gouraud shading為基礎並加以改良，可處理具有透明性質的物體，提供texture mapping的功能，但是不具ray tracing的能力。

PS 340

The PS 340 is a single-user system with the full capability of the PS 330, and additional features for three-dimensional displays of objects defined as polygons. The monochrome and color calligraphic displays can be used to show hidden-line-removed and sectioned views of polygonal objects. An optional color raster display allows the system to display smooth-shaded static images of three dimensional surfaces and solid objects computed locally or generated by the host computer.



The PS 340

IAS0355

□ (5.1)

5-3 系統流程

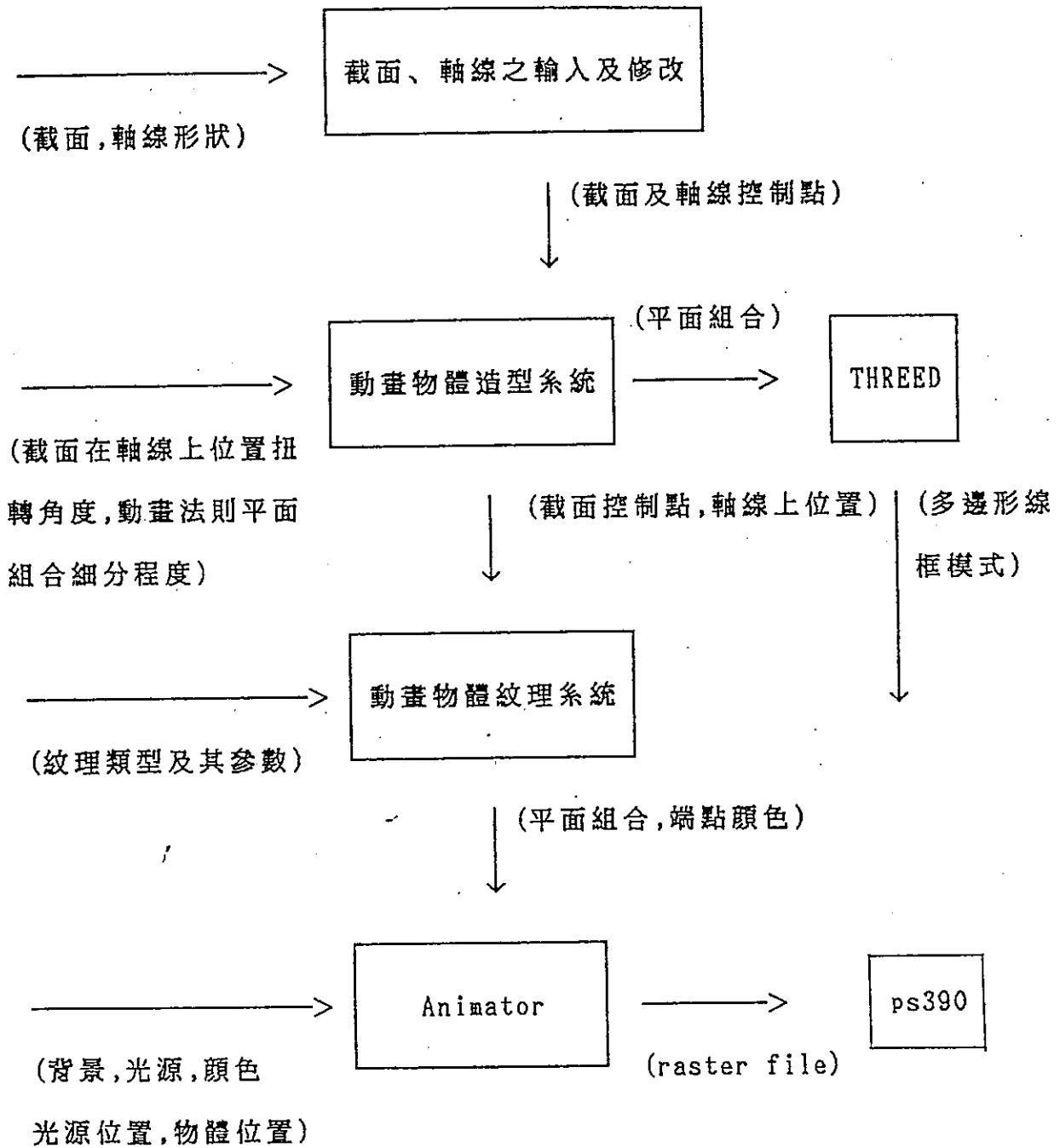


圖 (5.2) 系統主流程

輸入各主要截面在軸線上位置
輸入扭轉角度及動畫法則
輸入平面組合細分程度



算出所有中間截面形狀



將所有截面移到軸線上



連接截面間對應點
形成多邊形平面組合

圖 (5.3) 造型系統流程

輸入紋理類型
及其參數



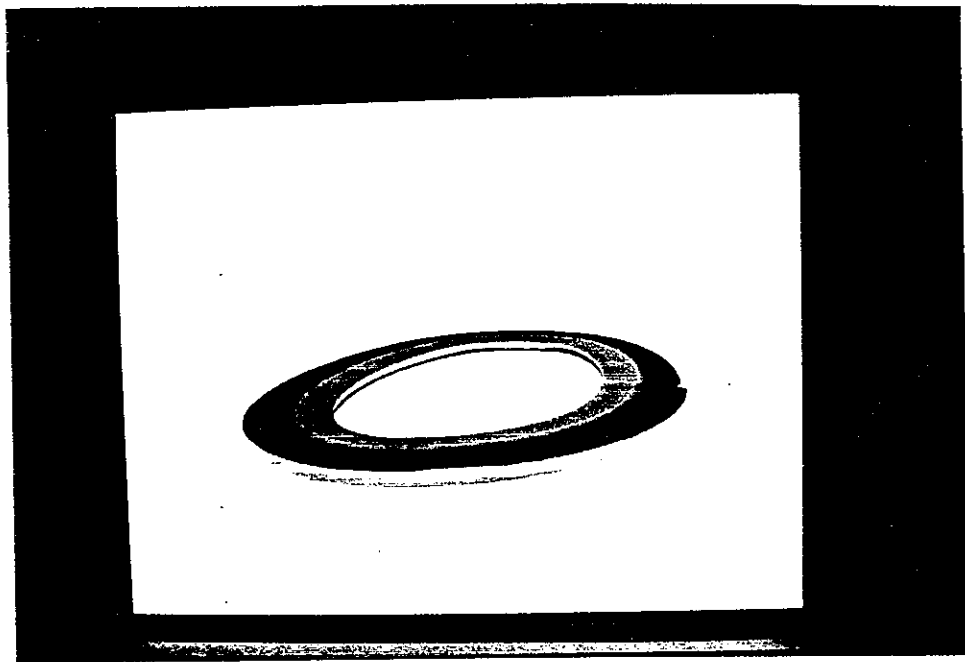
木紋函數
斑點函數
橫紋函數
混色函數



所有多邊形端
點顏色

圖 (5.4) 紋理系統流程

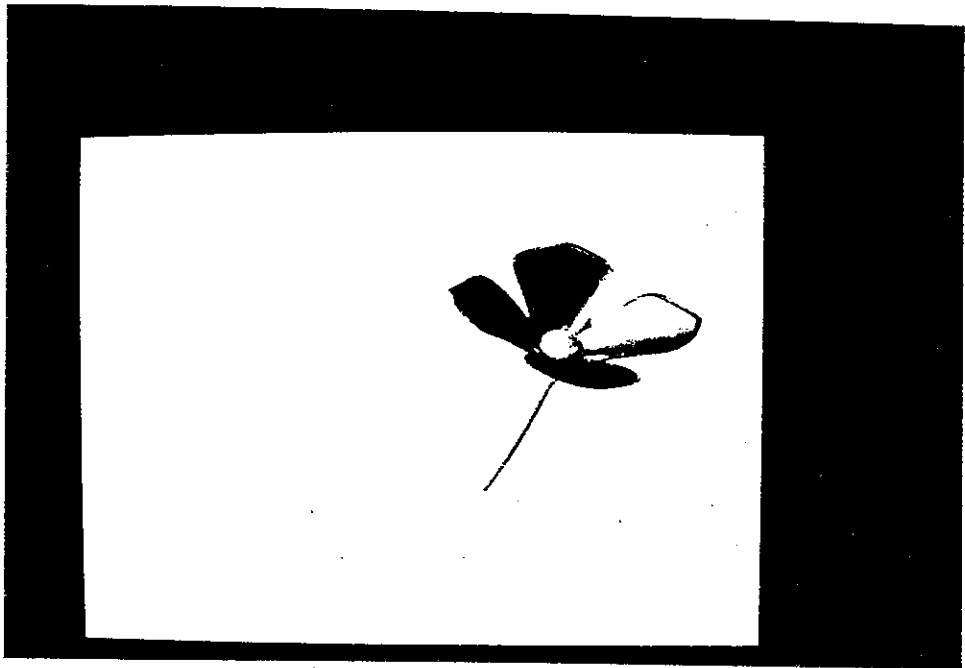
第六章 實作結果



圖(6.1)：環狀條紋彩帶



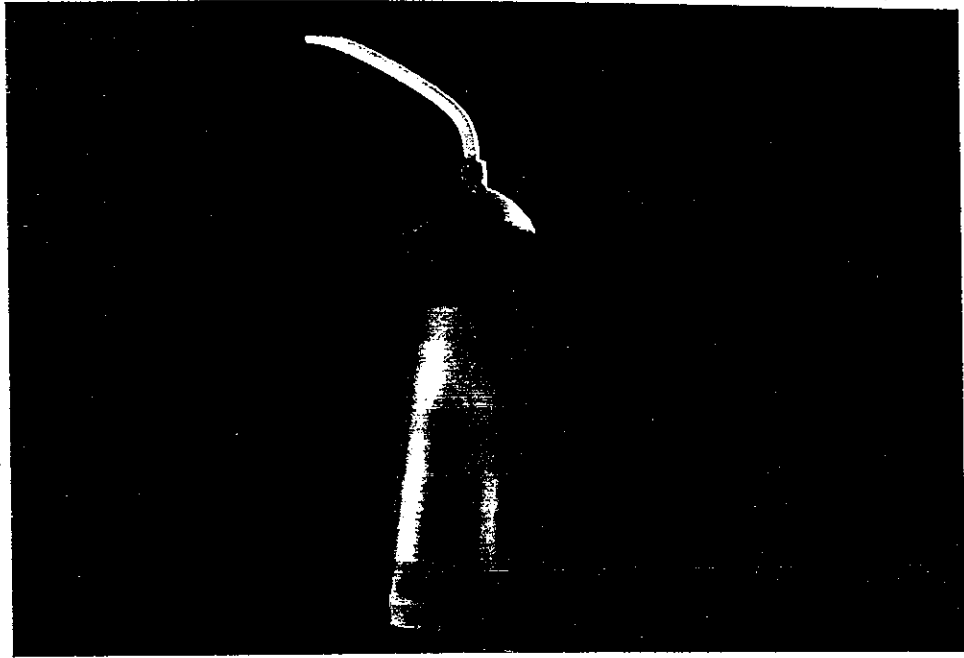
圖(6.2)：扭轉 180 度的彩帶



圖(6.3)：花瓣部分使用 3D 封閉曲線



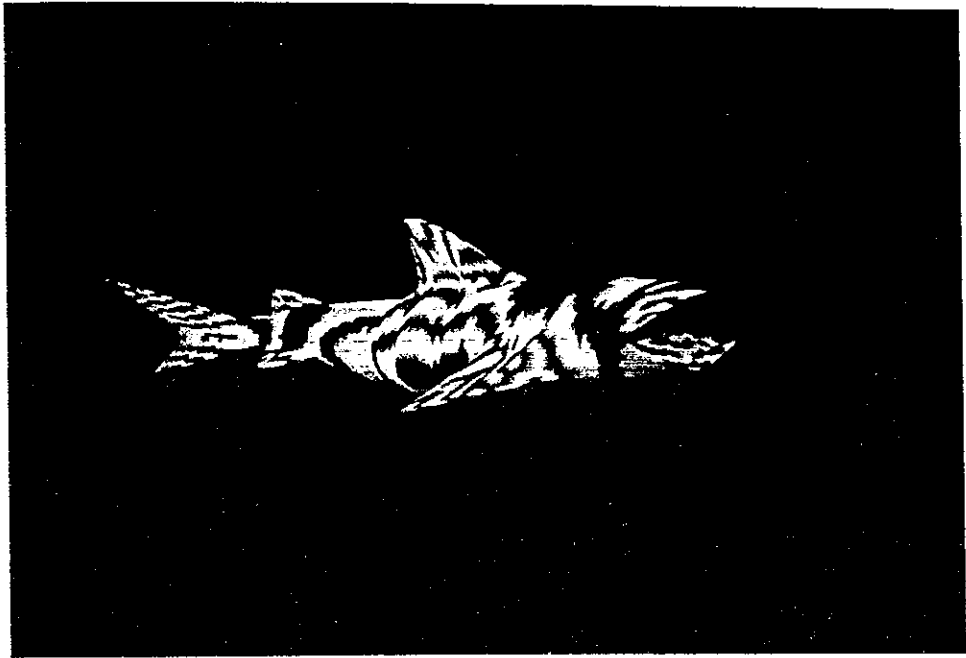
圖(6.4)：香蕉皮部分使用 3D 封閉曲線



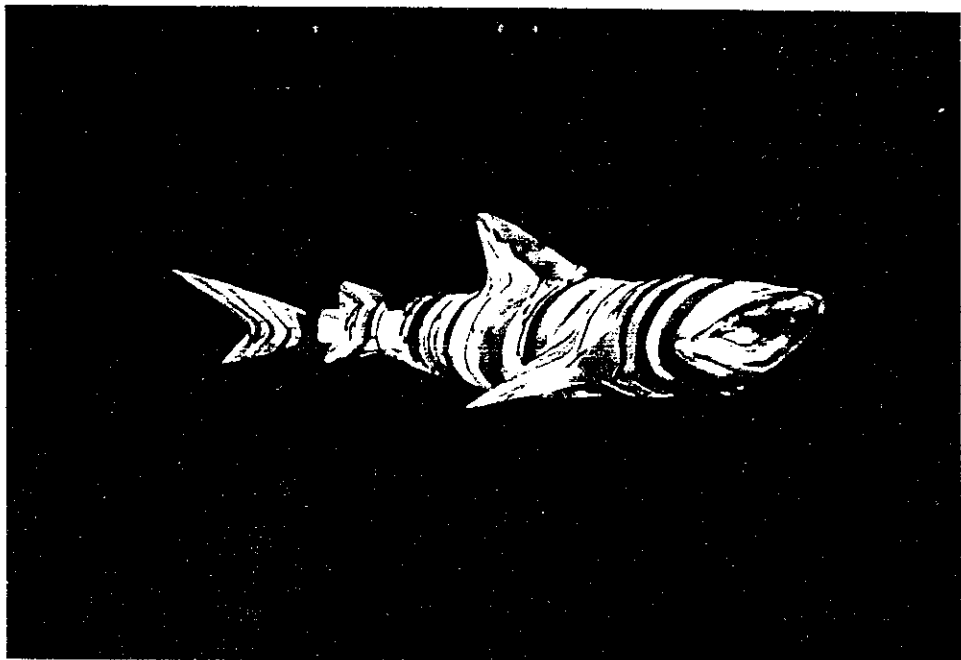
圖(6.5)：牙膏



圖(6.6)：水彩顏料，由橫紋及混色兩種紋理組合而成



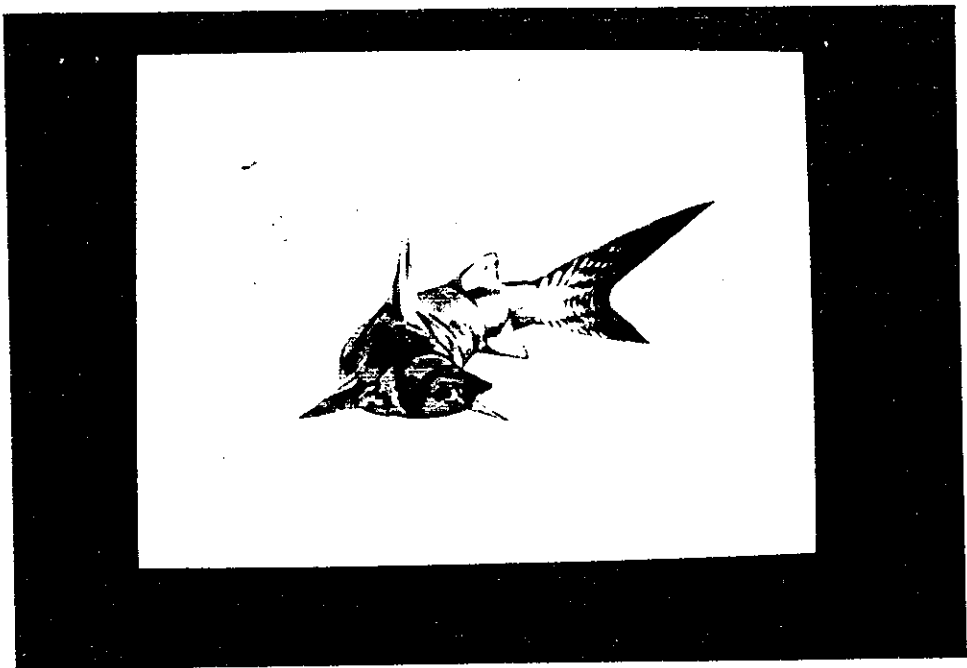
圖(6.7)：木紋鯊魚



圖(6.8)：經過混色函數處理的彩紋鯊魚



圖(6.9)：木紋鯊魚 A



圖(6.10)：木紋鯊魚 B

第七章 結論

本論文提出一套供動畫使用的物體造型及紋理系統。在造型方面，以 PCG法為基礎，可產生bicubic B-spline 曲面，並提供扭轉，彎曲等變形效果，尤其是將截面的定義由二度空間封閉曲線擴展到三度空間封閉曲線，使造型能力更加強大。由於“用截面掃掠出物體外形”的觀念十分容易了解，使用者只需定義少數截面及一條軸線，就能生成複雜而自然的物體外形，將繁複的三度空間造型工作簡化到較易掌握的二度空間問題。此外，本系統還提供許多可調整的造型參數（如動畫法則 扭轉角度等），可以讓使用者描繪出生動而富變化的動畫物體。在紋理處理方面，本論文根據 PCG法生成物體的特性，提出“實體紋理內部座標系”的觀念，解決了實體紋理應用在動畫中變形物體時發生的紋理不一致的問題。

以下仍就物體造型及紋理處理兩方面列出未來研究改進的方向。

物體造型：

- 1.目前系統使用的曲線只限於 B-spline ，雖然B-spline

有 C^2 連續性並具局部控制能力，非常適合造型設計。但是無法精確表現圓弧。加入 rational curve 可以解決這個問題。

2. 本系統造型上最大的限制在於：只能造出具單一軸線的物體（如魚），如果構成物體的軸線超過一條（如具有四肢的動物），則須先就各軸線造出部份物體再加以組合。如何將各部份連接成一整體並維持連接處的平滑，還需要深入研究。
3. 物體產生的程序及參數的設定目前仍由使用者循序輸入指令及數值來達成。如何設計一視覺語言以提供使用者一個方便的圖型界面將是另一重要課題。

紋理處理：

1. 由於著色軟體 Animator 本身並不具有處理實體紋理的功能，因此本系統以設定物體多邊形組合的端點(vertex)顏色來模擬實體紋理的效果。當構成物體的多邊形數目不夠多時，產生的紋理會有鋸齒狀的邊緣。要解決這個問題仍待未來的研究者將實體紋理的功能加入 Animator 程式中。
2. 本系統的紋理函數共有七種，雖然具備組合能力，但是就實際應用而言仍嫌不夠，尤其是缺少 bump mapping

的凹凸紋理效果。為了使系統更完備實用，增加新的紋理函數是必需的。

參考文獻

1. Nadia Magnenat-Thalmann, Daniel Thalmann: Computer Animation : Theory and Practice ,Springer-Verlag,Tokyo 1985.
2. Fujio Yamaguchi: Curves and Surfaces in Computer Aided Geometric Design ,Springer-Verlag,Tokyo 1988. chapter 6.
3. W. Boehm: "Inserting new knots into B-spline curves," Computer Aided Design, Vol.12, No.4, July 1980, pp. 199-201.
4. J. Wijk, Van Jark: " Ray tracing objects defined by sweeping planar cubic spline," ACM Translations on Graphics, Vol.3 , No.3, July 1984, pp. 223-237
5. Coquillart S:"A control-point-based sweeping technique,"IEEE Computer Graphics & Applications, November 1987, pp. 36-45
6. W.F.Bronsvort, P.R.van Nieuwenhuizen,and F.H.Post: "Display of profiled sweep objects," The Visual Computer, No.5, 1989, pp. 147-157.
7. Fopke Klok: "Two moving frames for sweeping along a 3D trajectory," Computer Aided Geometric Design, 3(1986) pp.217-229
8. N. Magnenat-Thalmann, D. Thalmann: "Building complex bodies: combining computer animation with CAD," Computers in Mechanical Engineering, May 1986, pp. 26-33
9. N. Burtnyk, M. Wein: "Computer-generated key-frame animation ," Journal of Society for Motion Piction and Television Engineering, 80:pp. 149-153
10. Nadia Magnenat-Thalmann, Daniel Thalmann: Image Synthesis : Theory and Practice ,Springer-Verlag,Tokyo 1987.
11. Ed Catmull: "A subdivision algorithm for computer display of curved surfaces," PhD thesis, Univ of Utah, December 1974
12. David F. Rogers: Procedural Elements For Computer Graphics. McGraw-Hill, 1985
13. JF. Blinn: "Simulation of wrinkled surfaces," Proc. SIGGRAPH '78, Computer Graphics Vol.12, No.3, pp.286-292
14. Darwyn R.Peachey:"Solid texturing of complex surfaces," Proc SIGGRAPH '85, Computer Graphics, Vol.19, No.3, July 1985, pp 279-286
15. Ken Perlin : " An image synthesizer," Proc. SIGGRAPH '85 , Computer Graphics,Vol.19, No.3, July 1985, pp.287-296
16. J.P.Lewis: "Algorithms for solid noise synthesis," Computer Graphics, Vol.23, No.3, July 1989
17. Geoffrey Y. Gardner : " Simulation of natural scenes using textured quadric surfaces," Computer Graphics, Vol.18, No.3 ,July 1984