

國立暨南國際大學資訊管理研究所

碩士論文

CG 植物擬真影像之生成手法

**Photorealistic Images Synthesis of Computer
Graphic Plants**

指導教授： 陳履恆 博士

廖大穎 博士

研究生： 李欣緯

中華民國九十五年七月

誌謝

在唸研所的這兩年經歷了許多事，首先要感謝陳履恆老師對我的諄諄教誨以及廖大穎老師對於為人處事的指導，並且感謝葉羅堯學長、謝曜辰學長、小黃學長、winner 學長、阿富學長以及沛宇學長對我的細心指教，同時也感謝宗憲、淑娟、恭達、上傑、家禾、文建、小雯常常陪我吃飯，讓我這兩年的生活過的非常充實。

在口試前的這幾天裡也非常感謝 change、小林、阿參、龍貓，常常關心我要記得吃飯，並且感謝 point、文祥以及小溫在我上台北口試時幫我打理口試的東西，要感謝的人實在太多了，如果有遺漏的敬請見量。

李欣緯 誌於

暨南國際大學科技三館409室

二零零六年七月十三號

論文名稱：CG 植物擬真影像之生成手法

校院系：暨南國際大學管理學院資訊管理系

頁數：54

畢業時間：九十五年七月

學位別：碩士

研究生：李欣緯

指導教授：陳履恆 博士、廖大穎 博士

中文摘要

在目前的 CG 領域中產生擬真的植物影像是極為重要的議題，至於要如何產生擬真的植物影像則應該著重在 rendering 的技巧和植物形態特徵的真實呈現上，對於植物形態特徵的展示來說，在模擬植物的成長和建立植物的結構上 L-System 是最為人所熟知的方法，L-System 可以藉由使用自我疊代的規則和文法來提高植物的真實度。但 L-System 還是具有某些的缺點，例如：L-System 將需要大量的文法和複雜的疊代過程才能展現植物和環境間的自然互動，因此我們知道對於用 L-grammar 將植物影像的具體呈現來說，L-System 是非常困難的去進行參數化的手續；所以我們提出了一個叫做 SimEco 的 proto-type 系統用來產生擬真的 CG 植物影像，SimEco 的基本想法就是藉由使用創新的 L-System 以階層的方法來表現植物在植物學上的架構，而此種 L-System 是屬於 context-sensitive 的正規化語言，並且可以用來簡化 L-grammar 的複雜性和模擬植物以及環境間的互動；除了創新的 L-System 之外，我們另外發展一套具備物理模型的 rendering 引擎來配合 L-System 和呈現擬真的植物影像，而為來提高影像的真實度所以我們需要非常多的植物器官模型，因此我們開發一套 LEAF CAD 的工具以及使用 3D MAX 來制作莖、葉子和花等的模型，此外我們還採用 Pre-computed Radiance Transfer[22]的技術來改善植物影像的品質。

關鍵詞: L-system，PRT，植物生成

Title of Thesis : Photorealistic Images Synthesis of Computer Graphic Plants
Name of Institute : Department of Information Management, National Chi Nan
University Pages : 54
Graduation Time : 2006/07 Degree Conferred : Master
Student Name : Shin-Wei Lee Advisor Name : Dr. Lieu-Hen Chen · Dr. Da-Yin Liao

Abstract

Rendering photorealistic images of plants is always an important research issue in the field of computer graphics. To achieve this research goal both rendering technique and simulation of morphology should be considered. Although L-System is the most well-known method to simulate the growth of plants and construct the structure of plants by self-developing and grammars, it is still difficult to describe the interactions between plants and environment. Furthermore, it is very hard to parameterize the botanic properties of plants for representing plants in the format of L-grammar. Thus, we present a proto-type system in this paper, called as SimEco to synthesize photorealistic images of CG plants. The basic concepts of SimEco are to describe the hierarchical botanic structure of plants by adopting context-sensitive L-System. This system simplifies the complexity of L-grammar and simulates the interactions between plants and environment. Besides, we develop a physical rendering engine to manipulate L-System and present the photorealistic images of plants. We also develop a LEAF CAD tool and use 3D MAX to fetch the stem, leaf and flower models. Furthermore, we adopt Pre-computed Radiance Transfer technique [22] to improve the photo-realism of plants.

Keywords: L-system, PRT, generate plants

內容目錄

誌謝	I
中文摘要	II
Abstract	III
內容目錄	IV
圖目錄	VI
表目錄	VIII
第一章 緒論	1
1.1 研究背景	1
1.2 研究動機與目的	2
1.3 研究方法	3
1.4 論文架構	4
第二章 文獻探討和相關研究	6
2.1 L-SYSTEM 的研究	6
2.2 擬真成像的研究	7
第三章 理論基礎	11
3.1 L-SYSTEM 理論基礎	11
3.2 PRECOMPUTED RADIANCE TRANSFER (PRT)理論基礎	13
第四章 系統設計	25

4.1 系統架構.....	25
4.2 系統流程和操作流程	26
第五章 系統實作	30
5.1 匯入的檔案	30
5.1.1 文法描述檔.....	30
5.1.2 3D 植物器官模型的取得.....	31
5.2 植物外觀的呈成	35
5.2.1 莖的貼合.....	35
5.2.2 葉子的貼合.....	39
5.2.2 花的貼合.....	40
5.3 PRT 在 SIMECO 中的使用方式	40
5.4 結果影像	43
第六章 結論	50
參考文獻	52

圖目錄

圖 3-1	與上下文無關的文法	12
圖 3-2	與上下文相關的文法，範例一	12
圖 3-3	與上下文相關的文法，範例二	13
圖 3-4	原始函數 $f(x)$	14
圖 3-5	基底函數 $B_1(x)$ 、 $B_2(x)$ 、 $B_3(x)$	15
圖 3-6	求算 $f(x)$ 的近似函數	15
圖 3-7	associated Legendre polynomials 的前九項函數	18
圖 3-8	SH 的前二十五項係數分佈圖	20
圖 4.1	系統的基本架構	26
圖 4-2	簡易的系統流程	28
圖 4-3	系統的操作流程	29
圖 5.1	以 3D MAX 製作玫瑰花	31
圖 5.2	以 Bezier 曲線拼湊的葉子輪廓影像	32
圖 5.3	標示葉子外觀之頂點或拐點	33
圖 5.4	葉子網格及連接點	33
圖 5.5	葉子模型的形變	34
圖 5.6	莖主軸的生成	36
圖 5.7	系統中莖主軸的貼合影像	36
圖 5.8	莖分枝的示意圖	37
圖 5.9	函數 $f(x) = \frac{1}{cx^p}$	37
圖 5.10	莖主幹和分枝連結的示意圖	38
圖 5.11	系統中莖主幹和分枝連結的貼合影像	39
圖 5.12	葉子的多種形變影像	39
圖 5.13	玫瑰開花的漸近影像	40
圖 5.14	PRT 模擬的流程	41
圖 5.15	PRT 材質說明圖	42
圖 5.16	玫瑰葉子成果圖	43

圖 5.17	日日春葉子成果圖.....	43
圖 5.18	檸檬桉葉子成果圖.....	44
圖 5.19	紅刺蔥葉子成果圖.....	44
圖 5.20	香甜肉桂葉子成果圖.....	45
圖 5.21	月桂葉子成果圖.....	45
圖 5.22	玫瑰光影變化圖.....	46
圖 5.23	日日春光影變化圖.....	46
圖 5.24	PRT 陰影概念圖.....	47
圖 5.25	玫瑰的完成圖(1).....	48
圖 5.26	玫瑰的完成圖(2).....	48
圖 5.27	玫瑰的完成圖(3).....	49

表目錄

表 5.1 部分 PRT 設定的一般項.....	41
--------------------------	----

第一章 緒論

1.1 研究背景

縱觀目前不管是在遊戲製作、動畫電影或虛擬實境的運用上，凡是使用電腦圖學來製作擬真的影像上皆免不了要面對戶外大型的場景，而植物的出現更是這些場景中無可避免的因素，即便是室內的場景也可能會有植物影像的出現，因此植物的擬真成像可以說是電腦圖學中不可或缺的一環，至於目前電腦圖學植物影像的擬真上，大致可分為兩種主要的研究領域，其一是符合植物學形態的植物探討，另一是如何產生迷人的植物擬真影像。

對於植物形態的探討上，主要的方法就是 Lindenmayer 在 1972 所提出的 L-System[10]，藉由考慮文法生產規則(production rule)和符號定義(alphabet)的不斷疊代以符合植物的基本架構，然而為了使植物的 3D 結構可以更具體的展現，所以此後的學者開始經由觀察植物的生長過程，以及探討植物器官彼此的相應關係來強化文法的廣度和深度，同時不斷的加強文法生產規則(production rule)的正確性，藉此產生出擬真的單一植物影像[9][14][15][16]。但是在 L-System 強化的過程中有些學者漸漸的著眼於不單只是想要發展出只適用於植物本身的文法，還想將植物和環境的互動行為也同時加入到影像中，使得植物由原本單一行為的呈現到能展現真實植被的分佈影像[4][5]。

然而在以文法的方式來表達植物學的特性為主流的研究上，有部份的學者是以快速和符合人類對植物意向的表達來探索植物影像的呈現[7][13]，這些的研究內容皆以手動或半手動的方法，經由使用者本身對於植物的概念來製作植物的模型，而不是以文法的方式來表現，雖然這樣的研究欠缺客觀的數據作為依據，不過這樣的方式確實加速了植物模型的產生，並且所產生出來的植物影像也未背離人們常識中

對於植物所應具備的特性，藉此在取得自然植物的模型和簡易的產生植物的影像中獲得了適度的平衡。

在上述學者的研究中大體都只在加強文法和植物的契合度上或以手動以及半手動的方式來快速的表達植物所具備的特徵，當然確實其所呈現出來的影像越來越符合植物在植物學上所應有的表現，然而在電腦圖學的領域上所應該具有的迷人影像卻往往被忽略了，也因為這樣植物的研究上產生了另外一個分野——除了植物形態的呈現外如何獲得擬真的植物影像，例如：植物的組織是非對稱的材質、葉子和植物表皮的半透明以及物體的半影區，那麼如何將這些個特性表現出來呢，像這樣的問題開始在電腦圖學植物的研究慢慢突顯出來，所以近幾年的植物研究範疇幾乎離不開這個議題，即如何改善植物的 rendering 技術[1][2][25]。

1.2 研究動機與目的

經由上述所論，我們可以知道如果想要獲得令人印象深刻的植物影像，除了要有正確的植物文法來表現植物在植物學上的架構外，也要有適當的 rendering 技術來搭配，所幸的是文法的部份經由長年的改進和研究已有豐富的資料可供研究，而 rendering 的問題不只局限在植物的問題而已，同時也存在於各個電腦圖學的領域中，因此可藉由探索其他的研究領域而獲得解決的方案。

就目前的 L-System 而言，L-System 的不斷改進的確使得文法更貼近植物在植物學上應有的表現，但同時在加強文法的功能過程中也使得文法變得更趨複雜，且令人難以理解其疊代後會產生的影像，而且對於文法的撰寫而言，即使是一個善於編排植物文法的專家也需要藉由無數次的試驗才能正確的編寫出植物應有的文法，更遑論像那些第一次接觸文法的人來說，基於這類的因素所以在簡化文法本身的複雜性上是有其必要的。

對於 rendering 技術的改進已有許多的學者相繼的在這個議題上做出貢獻[26]，從早期的 Phong shading、Gouraud shading 到近期的 BRDF/BTDF(BSSRDF)[8]、

Precomputed Radiance Transfer[22]都大大的改進了電腦圖學上影像的呈現，因此在 Microsoft 所發展的 DirectX 中也提供了 Phong shading、Gouraud shading 的 API 來使用，並且在 DirectX9.0 中更進一步的開放非正式版的 Precomputed Radiance Transfer 函式庫給大眾使用，到此我們發現若要改進植物的擬真成像，至少在 rendering 技術的採用上將不再是無所適從的。

然而為了改善使用者不會因為複雜的植物文法而無法撰寫出想要的植物影像，因此我們打算簡化文法的撰寫過程，讓植物文法的編寫只需要簡易的數道文法生產規則(production rule)和幾種符號定義(alphabet)就可以相互疊代出想要的植物影像，因此我們需簡化文法的撰寫流程，然而簡化並不是無限制的，因此有些基本的觀念應該要具備，諸如：為了讓所產生的文法具有植物傳遞生長激素的觀念，所以需要 L-System 中與上下文相關(context-sensitive)的功能，而為了使一株植物文法可以產生多種變化，故不確定性(non-deterministic)的功能也應該具有，並且為了使文法具備合理化，所以非傳播性(non-propagating)的功能也應該要擁有，至此方可以令使用者能簡單的掌握植物的特性和規則，這就是我們研究的目的之一；此外單調的植物色彩已經無法吸引使用者在觀察植物時的目光，所以我們打算以快速和方便的方法來獲得豐富的植物色調，雖然對於植物的影像也有其他的專家學者在做專門的研究探討，但他們大多著重在局部影像的呈現和理論性的探討上，而我們主要是著眼於文法和植物各個器官的搭配上，再採用既有的實際理論來提升影像的品質以呈現出豐富的植物效果，這就是我們研究的第二個目的。

1.3 研究方法

依據上一節本研究所述的目的，因此本論文首要的目標就是簡化文法的撰寫過程並保留有用的功能，也就是要開發一套全新並符合 L-System 的簡意語言，幸運的是目前本實驗室已經開發出一套簡意的 L-System 文法格式，它可以經由簡單的文法疊代來表現植物和環境間的互動關係，因此在本論文系統開發的過程中只需要採用

此既有之格式，再配合 L-System 的文法載入工具來開發系統即可，至於改善植物影像品質的方面，豐富且多樣的植物器官模型是必需的，所以我們自行開發了一套 LEAF CAD 的工具來產生各種形狀的葉子模型，再搭配 3D MAX 的操作來製作植物莖和花的模型，至此只是讓植物的基本形狀能夠具體的呈現，然而對於產生迷人的植物影像仍然有待加強，就目前一般的 Phong 模型來說並不足以用來產生擬真的植物影像，而對於 BRDF/BTDF 的方法而言，因為其在材質的取得上皆是從無生物組織上來獲取的，而我們的方法是打算由植物影像來獲得，對此拍攝時間的急迫性將成為 BRDF/BTDF 資料取得的關鍵因素，因為隨著時間的消逝 BRDF/BTDF 資料的資訊將會有極大的誤差，因此如果比照一般的 BRDF/BTDF 資料取得的方法的話，將無法確切的證明 BRDF/BTDF 資料的正確性，再者 BRDF/BTDF 資料過於龐大且每次使用時都需在重新計算將大大降低系統的效能，根據這幾點的考量，所以我們放棄目前熱門的 BRDF/BTDF 技術，而改採 Sloan 所提出的 Precomputed Radiance Transfer 技術[22]，爾後簡稱為 PRT，會選擇 PRT 來加強植物影像的正確性，主要是因為目前關於漫射面的 PRT(Diffuse surface for PRT)研究在 Microsoft 所發展的 DirectX 中已經有現成的函式庫可以使用，且 PRT 的計算只要模型沒有再次的改變那麼只需要做一次求轉換函數的運算即可，然而如果光源有改變的話只需反覆的做光源函數和轉換函數(transfer function)的內積操作，即可求得適當的光影分布。

本系統主要分為一個前端文法載入模型和後端的 3Drendering 引擎，上述說提到的 LEAF CAD 是用 .NET for C++ 平台來開發的而主系統則是用 .NET for C++ 平台再配合 DirectX9.0c October 2005 來開發。

1.4 論文架構

本論文共分成五個章節，茲分述如下：

第一章為緒論，主要說明研究背景、研究動機與目的、研究方法和論文架構。

第二章為相關文獻探討，主要說明 L-System 和光源成像的歷史，並說明最近關

於 L-system 和植物擬真成像的相關研究。

第三章為我們系統中所使用到的理論基礎，主要說明 L-System 的理論基礎並舉出一些疊代的操作規則和操作方法的實例，以及 PRT 的推導過程和使用的方法。

第四章為系統架構，說明我們的系統包括那些部份以及這些部份各是在做什麼，並說明系統的主要操作過程以及和使用者的互動。

第五章是系統實作，用來闡述系統實際運作時的匯入檔案的取得以及模型的取得和如何成像的方法，最後在說明 PRT 的使用和成果。

第六章為結論，說明目前我們系統中待解決的一些問題。

第二章 文獻探討和相關研究

2.1 L-System 的研究

早期植物影像的成像手法並沒有固定形式，除了以直覺得方式產生影像外，一般植物影像的呈現還是以碎形(fractals)[23]、粒子系統(particle systems)[19]和正規化語言(formal languages)為主，碎形方法和粒子系統所產生的影像只是在某一個時點上符合於植物的形態，至於在此之前或之後的影像就無法保證是否合乎植物的概念，因此這類的方法只有少部份的研究學者在使用，而真正可以說是植物影像手法的大宗應該是正規化語言了，而在這裡所指的正規化語言其實就是 — L-System，這個方法在前面的章節已經有大致的提到，它乃是由德國人 Lindenmayer 在 ACM symposium on Theory of computing 中提出，往後的 L-System 研究就是架構在這個基本上再繼續延伸的。

Prusinkiewicz 等人以 L-System 為基礎並觀察植物實際生長時的一些特性，例如：分枝的結構、生長點的因素，來增強文法的多樣性，如：生長點的控制、分枝角度等，並且為了使植物器官更具有說服力，更引進實際器官的統計數據，像花或果時的生長比例或生長速率[14][16]。

Deussen 等人提出以植被為主軸的研究，探討如何在巨大的場景中進行空間分佈的配置，並且由於在這樣的場景中如果同時置入大量的植物將使得系統整體的效能大幅下降，因此他們在系統中引進以點和線段為基礎的 LOD(level of detail)來增加系統的效能[4][5]。

到目前為止所提的理論都是著重在植物外觀的表現和空間分佈上，那麼現在我們探討另外一種以植物酵素為主的觀點，Renton 等人提出以植物內在激素為依據來模擬植物的生長，並且由於內在的激素會因為生長的環境和四季的變化而有所改變

所以一組文法將可以有不同之影像[20]。

2.2 擬真成像的研究

以下論述乃是根據 Alan Watt 所寫的 3D Computer Graphics[26]加以統整，那麼在說明電腦圖學中擬真影像的成像之前，我們應該先了解影像是如何產生的，下面式子是物體表面影像的成像模型：

$$\begin{aligned} \text{像素上的顏色} = & \text{投射反射(specular)+ 漫射反射(diffuse)} \\ & + \text{周圍光線(ambient)+多重反射(reflected)+透射(transmitted)} \end{aligned}$$

電腦圖學中為了解決上面式子的成像問題，因此 Phong(1975)首先提出了 Phong 模型之反射模型來解決投射反射、漫射反射和周圍光線的問題，繼 Phong 模型之後 Blinn(1977)提出架構在 Torrance 和 Sparrow(1967)的物體表面微平面模型之下的改進反射模型，改變了原本電腦圖學中以實驗數據為主的反射特性而變為有物理模型的根據，上述的方法都是假設物質是等向性的(isotropic)，但一般日常生活中的物質大多是非等向性的(anisotropic)，因此 Nicodemus 等人(1977)[12]提出根據 Fresnel 法則的 BRDF 方法，使得物體表面的成像必需依賴入射光的方向和觀察者的視點來決定，之後 Cook 和 Torrance(1982)以入射的能量而非亮度值來處理模型，而投影項是由物理性的微平面模型得知，並且其高亮光區的顏色變化是依據 Fresnel 的法則，此方法明確的區分出金屬和非金屬的物體，也適當的改進了非完美平面的成像，近期為了解決次表面反射(subsurface scattering)的問題——物體半透明的視覺效果即光線穿進物體後和內部的粒子碰撞後的結果，例如：光線穿過皮膚或玉石的效果，因此 Jensen 等人(2001)[8]藉由改進原本的 BRDF 而提出了 BSSRDF(bidirectional surface scattering distribution function)的方法。

上述的方法都是區域反射的模型主要著重於解決物體間的反射問題，並未涉及環境間對物體影響的全域照明問題，那麼何謂全域照明呢，凡是由光源直接照射入

場景的光線，在射入物體表面有和其他物體產生交互作用的都歸類為全域照明，而 Phong 模型被認定為是區域反射的模型是因為其投射項和漫射項都是由直接照明所產生的，而另外的周圍光線則是由場景中其他物體所自己發散的漫射光線，這也與物體間之交互作用無關，因此我們知道物體表面反射光線的來源有三種：(1)直接光線、(2)周圍光線以及(3)物體交互作用後的集中光束。

那麼在繼續說明全域照明的發展之前，首先應該了解在全域照明的場景中會有那些情形，全域照明的場景總共有四種可能的情況，漫射面到漫射面、投射面到漫射面，漫射面到投射面以及投射面到投射面，這四種情況是 Wallace 等人(1987)提出來的，因此若想要解決全域照明的問題也就是需解決這四種場景的問題，對此早期的學者經由觀察光線的成像過程提出了光跡追蹤法(ray tracing)，一般公認第一個有系統性的光跡追蹤法是由 Whitted(1980)所提出的 Whitted 光跡追蹤法(Whitted ray tracing)或稱為 forwards ray tracing、one pass ray tracing，在追蹤的過程中是由眼睛的方向朝著場景方向追蹤到光源，其主要解決了投射面的交互作用問題，然而其最不合理的一點就是被追蹤的光線不會有漫射的現象，為了克服這個問題可以增加初始的光線和相交點的衍生光線，但一旦採用此方法將造成計算成本大量增加，因此 Cook 等人(1984)等人提出架構在 Monte Carlo 法下的分散式光跡追蹤法(distributed ray tracing)來解決，以均勻的光線分佈取代增加大量的入射光線，然而為了解決投射面到漫射面的問題 Arvo 等人(1986)發表 backwards ray tracing 或稱為 two pass ray tracing，這方法具有兩個階段，第一個階段是由光源射出光線經過投射面的作用直到找到漫射面，然後將照度儲存在照度貼圖中(illumination map)，第二個階段是眼睛追蹤，如同 Whitted 光跡追蹤法，由眼睛的方向經由投射面的作用後找到漫射面就停止，然後使用存在漫射面的照度貼圖來計算光源的能量，雖然光跡追蹤法確實解決了一部分全域照明的問題，但其僅止於投射面的問題上，但大部分的電腦圖學場景都是漫射面的環境，因此有了輻射法(radiosity)的出現來解決漫射面到漫射面的問題，輻射法最早是 Goral 等人(1984)在康乃爾大學發展出來的，爾後關於輻射法的

研究也幾乎都是在康乃爾大學完成的，這個方法是依據封閉空間中能量守恆定理而來，用來解決封閉區間內的物體輻射現象，其方法就是將光源和物體皆視為輻射體，只是光源具有非零值的初始值，在經由不斷的重覆計算每一頂點的輻射值來獲得物體表面的照度，輻射法雖然可以用來處理漫射面的問題但其計算的時間至少和光跡追蹤法一樣多，因此無法在電腦圖學界獲得普遍性的採用。

雖然光跡追蹤法和輻射法解決了一般常見的場景問題，但為了要產生一張影像可能要數小時甚至數天不等，因此後續的許多學者皆致力於計算時間的改善，而為了加快計算的時間因此絕不可能一步步的估計光源照射在每一頂點的累加值，所以 Ramamoorthi 和 Hanrahan[17]提出如果已知環境間的照明輻射(irradiance)如何推估照射在頂點上的入射輻射(incoming irradiance)量的方法，其方法不但證實某一頂點的入射輻射可以經由其所在場景推知，也證實即使旋轉光源函數依然可以快速的求得某一頂點的入射輻射，並由實驗數據證實環境間的照明輻射可以用二次的球型調和級數(Spherical Harmonic, SH)得到近似，同時 Ramamoorthi 和 Hanrahan[18]以環境貼圖當作照明輻射去實作漫射物質的 rendering 效果，Slaon 等人[22]以此為基礎提出名為 PRT(Precomputed Radiance Transfer)的技術，其藉由以球型調和級數表示某一點的光源組成函數，並從中抽離轉換函數以即時的運算全域照明下物體的顏色，包含處理半影區、反覆反射及光暈的效果。

以下我們探討關於近期在植物影像上的研究，Baranoski 等人[1][2]認為植物的組織乃是由細微的多層組織所組成，例如：葉肉組織、海綿組織，因此認為植物的表面類似人類皮膚應具有多層的反射和折射效應，所以藉由引進 BRDF/BTDF 以及提出 ABM 和 FSM 來加強效果，Soler 等人[24]以階層式來表示植物的材質特性而不是以貼圖材質來區分，因此其計算照度時是以整束的 BRDF 特性來表示而不是每一頂點的 BRDF 特性，Lintermann[11]提出了以階層式的方法，將器官彼此間的關係串接的 xfrog, Boudon[3]發展了一套 Tree Pro.以 global-to-local 的方式設計植物，Runions 等人[21]著重在葉子影像的多樣性呈現，他們引進 biologically-motivated 演算法來計

算植物葉子的自然生成，其藉由荷爾蒙的分佈，有效的模擬葉子的成長，並經由荷爾蒙的移動來表現葉脈在葉身上的排列以及裂葉的效果，Ijiri 等人[7]改善了目前植物影像的生成方式，他們以友善的使用者介面讓使用者可以在 2D 的平面上以直覺的方式繪製想要的植物影像，不但可以編排植物器官的位置，也可以製作莖、葉子以及花的形狀外觀，Wang 等人[25]提出在全域照明下快速的產生植物影像的方法，其除了採用 PRT 來表示低頻(lower-frequency)的影像訊號外同時也引進 light-visibility convolution 來表示高頻(higher-frequency)的影像訊號，使得其產生的影像不但有半影區也有全影區(hard shadows)的效果。

以上所列的是近幾年在植物擬真影像的研究，當然還有許許多多的理論和演算法正相繼的被研究以及使用中，在此我們無法一一列出，由此可知植物影像的擬真方法在學界已成為一項重要的議題。

第三章 理論基礎

3.1 L-System 理論基礎

在電腦圖學植物的研究上，L-System 是首先被提出來以平行覆寫系統(parallel rewriting systems)的方式，來模組化植物的形態和生長過程的一套正規化語言(Formal language)[23]，大體說來符合 L-System 定義的系統必需包含三個基本的架構：有限狀態集合(a finite set of states)、轉換函數(next-states functions)和初始狀態(initial arrays)，而這三種結構皆是由符號定義(alphabet)、文法生產規則(production rule)和起始字串(axiom)所組成，符號定義就是在 L-System 中用來疊代字串的所有字元符號，而文法生產規則就是每一個字元符號的疊代規則，至於起始字串即是 L-System 中最初用來定義操作的字串，文法中所有的字串其實都是由起始字串或初始狀態經由疊代文法生產規則而來。

L-grammar 和一般的正規化語言基本上是非常相似的，除了 L-grammar 的疊代過程是同時發生和無法區分非終斷點符號和終斷點符號之外，此外 Lindenmayer 為了使文法中可以表示植物的分枝結構，因此額外引進了 {[,]} 來表示，所以凡是在文法中被 [] 所括住，就表示括號中的文法是一組植物分枝的架構，如果括號中還有括好就表示是分枝的分枝，一般說來若以符號的上下文來區的話，可以將 L-System 分成兩種主要形式，其一為與上下文無關的文法(context-free grammar)，另一為與上下文相關的文法(context-sensitive grammar)，與上下文無關的文法又稱為 0L-System[23]，意即在疊代文法時和它相鄰的符號定義並不會影響疊代的結果，現在假設我們有一個簡單的例子，並且令符號定義為 {0, 1, [,]}，起始字串為 1，而文法生產規則為 { $0 \rightarrow 00$, $1 \rightarrow 0[1][1]$, $[\rightarrow [,] \rightarrow]$ }，那麼前三次疊代的結果應為 0[1][1]，00[0[1][1]0[1][1]] 和 0000[00[0[1][1]][0[1][1]][00[0[1][1]][0[1][1]]]，圖 3-1 中即為字串

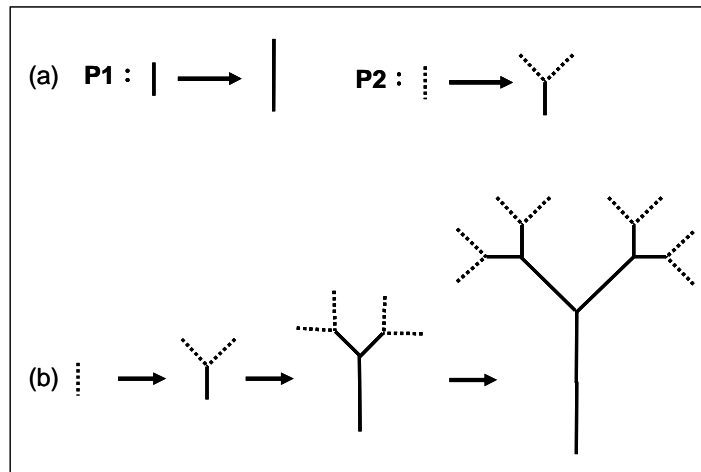


圖 3-1 與上下文無關的文法，(a)為文法生產規則，(b)為文法前三次疊代之結果。

的圖形化結果，與上下文相關的文法又可被稱作 $\langle k, l \rangle$ -L-Systems，其中 k 表示左邊相鄰的符號定義字母， l 表示右邊相鄰的符號定義字母，現假設有兩個簡單的文法，其符號定義皆為 $\{0, 1, [,]\}$ ，起始字串也都為 $l[0]0[0]0$ ，而文法生產規則各為 $\{0 > 1 \rightarrow 1, [\rightarrow [,] \rightarrow]\}$ 和 $\{1 < 0 \rightarrow 1, [\rightarrow [,] \rightarrow]\}$ ，那麼經過兩次疊代的結果應為 $l[0]0[0]0$ 和 $l[1]l[1]l$ ，圖 3-2、3-3 為其圖形表示法之結果，而對於與上下文相關的文

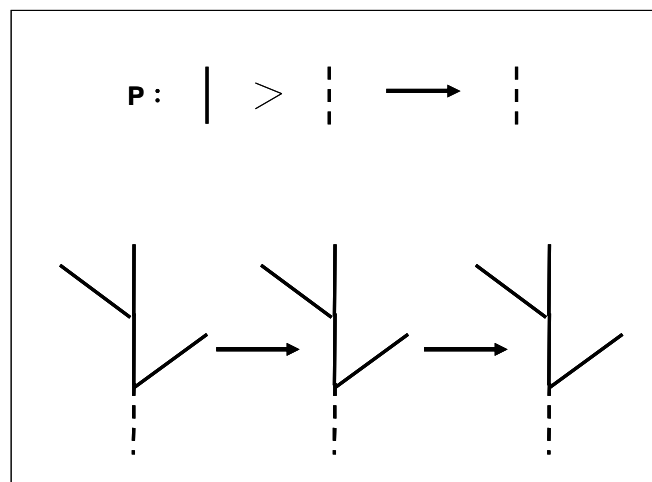


圖 3-2 與上下文相關的文法，範例一

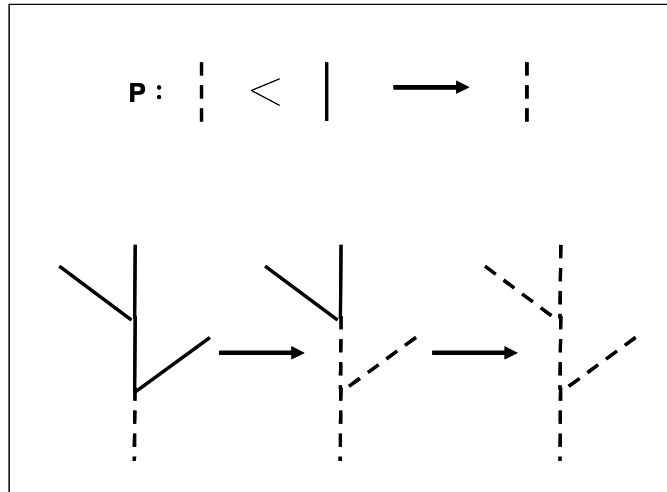


圖 3-3 與上下文相關的文法，範例二

法又可以被分為 1L-System 和 2L-System 兩種，主要的差別在於一個符號定義字母會被兩個符號定義字母所夾，那麼如果產生下一次字串時字母需參照左、右中的其中之一，那就是屬於 1L-System，假如需同時參照左右兩邊，那就是 2L-System；不確定性(non-deterministic)也就是說，同樣的起始字串且套用同樣的文法產生規則卻可以產生多組不同的文法字串，也就是具有不一樣的結果，而非傳播性(non-propagating)則是文法疊代的過程中會有死亡或者消逝的狀況發生，就是會有符號定義字母消失的情形發生。

目前我們所引進的 L-System 是具有不確定性和非傳播性之 $\langle k, l \rangle$ -L-systems，它具備高度的文法彈性度、植物特性的展現力和文法撰寫直覺，因此只需要簡單的文法就可以增加我們系統影像的擬真度。

3.2 Precomputed Radiance Transfer (PRT)理論基礎

PRT 理論基礎的來源是 Sloan、Kautz 和 Snyder 在 Siggraph 2002 年所提出來的，而其主要的架構乃是依據 Spherical Harmonic(SH)再配合 CG 上的光反射原理所推導出來的，它可以用來計算區域光源(area light sources)在 3D 模型上的影響，同時也可

以讓我們藉由 relight 的方式重新計算光源的影響，以即時的獲得全域照明下模型的影像，雖然 PRT 可以用來處理 diffuse 和 glossy 的物體，不過由於在系統中我們將模型都視為是 diffuse 的模型，所以在以下的理論探討中我們主要針對 PRT 於 Diffuse Surface 上的理論性推導，以下推導的過程是依據[6][17][18][22]這幾篇論文歸納得出的。

在上文中我們已經說過 PRT 的理論基礎是架構在 SH 的觀念之下，但到底真正的關係為何我們將這裡簡述一番，在了解何謂 SH 之前首先需明白基底函數，那麼何謂基底函數呢？基底函數就是一些函數的基本單元，且可以藉由加總和乘積各個基底函數的方式以近似原始函數(original function)，現舉一簡單之例子，例如有原始函數 $V(3, 2, -1)$ ，那麼可以假設有基底函數 $B_1(1, 0, 0)$ 、 $B_2(0, 1, 0)$ 、 $B_3(0, 0, 1)$ (基底函數不限制為一定是此三者)，那麼將基底函數投影(projection)後所得的係數即為此基底函數在產生逼近之函數時的貢獻度，如 $V \times B_1 = 3$ 、 $V \times B_2 = 2$ 、 $V \times B_3 = -1$ ，那麼經由這些係數 3, 2, -1 將逼近成 $3 \times B_1 + 2 \times B_2 + (-1) \times B_3 = V$ ，再者若以傅立葉轉換為例，那麼基底函數和原始函數的關係就如同 cosine、sine 函數和欲轉換函數的關係，下面再舉一求複雜函數之近似函數的操作，設有一複雜的原始函數 $f(x)$ ，因其過於複雜而無法得知，如圖 3-4，那麼假設知道其三個基底函數

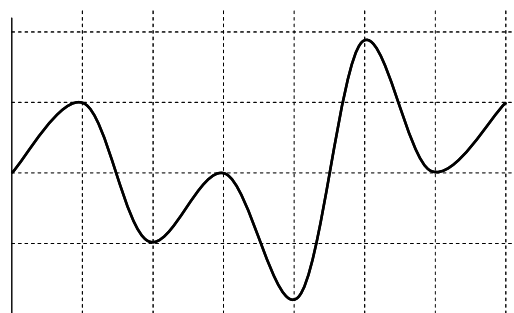


圖 3-4 原始函數 $f(x)$

$B_1(x)$ 、 $B_2(x)$ 、 $B_3(x)$ ，如圖 3-5 所示，那麼如同上例一樣我們只須求出每個基底函數

的乘積值，即將基底函數和原始函數做內積再求其積分值，然後將乘積值乘上

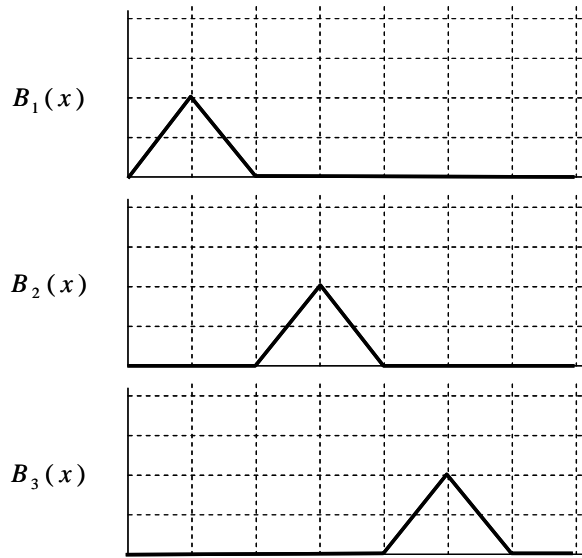


圖 3-5 基底函數 $B_1(x)$ 、 $B_2(x)$ 、 $B_3(x)$

基底函數再加總就可以逼近複雜的 $f(x)$ 函數，如圖 3-6。

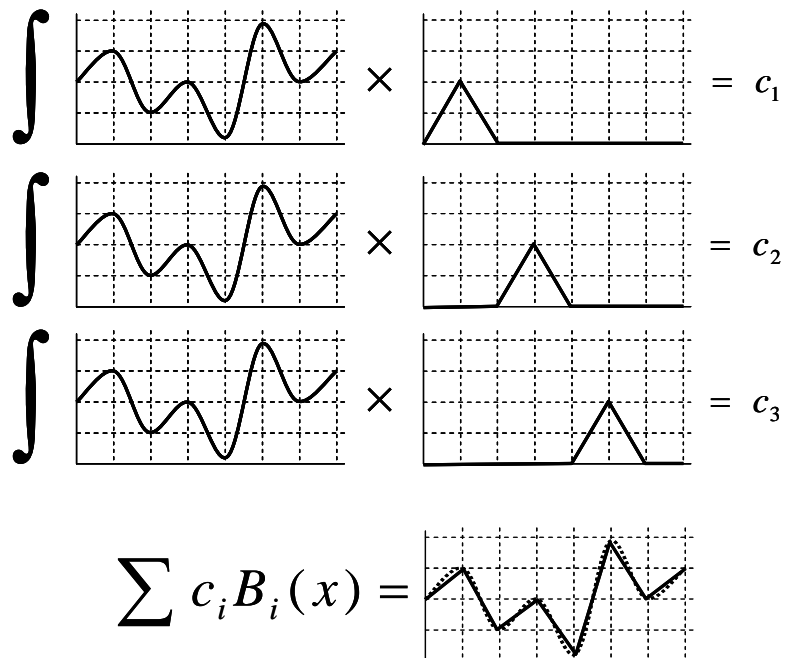


圖 3-6 求算 $f(x)$ 的近似函數

上面我們已經示範過基底函數的基本意義了，然而上述的範例只是說明了線性基底函數(linear basis functions)的用法，它主要是以片段的方式逼近真實的函數(piecewise linear approximation)，但因為 PRT 的目標是在 3D 的空間中，所以考慮的基底函數是正交多項式(orthogonal polynomials)的數學函數，例如：Chebyshev polynomial、Legendre Polynomial、Laguerre polynomial 和 Hermite polynomial 等。

兩函數為正交就是兩函數的內積為零，兩正交向量為正交就是其 dot product 的結果為零，而兩正交多項式為正交，即為兩多項式做內積後，求積分的值為零，也就是若兩多項式相同則有非零值若不同則為零。

$$\int_{-1}^1 P_m(x)P_n(x)dx = 0 \text{ whenever } m \neq n$$

雖然在數學的函式中有許多的正交多項式，但 SH 所採用的基底函數為 Associated Legendre Polynomials，associated Legendre polynomials 的範圍在 $[-1, 1]$ 之中，其函數的表示式主要是依據參數 l 和 m ，其中 $0 < m < l$ ，且 l 和 m 皆為整數，當為正的 m 時：

$$P_l^m(x) = (-1)^m (1-x^2)^{m/2} \frac{d^m}{dx^m} P_l(x)$$

其中 $P_l(x)$ 可以根據 Rodriguez' formula 得知為：

$$P_l(x) = \frac{1}{2^l l!} \frac{d^l}{dx^l} ([x^2 - 1]^l)$$

所以最後 associated Legendre polynomials 在正的 m 時為：

$$P_l^m(x) = \frac{(-1)^m}{2^l l!} (1-x^2)^{m/2} \frac{d^{l+m}}{dx^{l+m}} (x^2 - 1)^l$$

而在負的 m 時，associated Legendre polynomials 為：

$$P_l^{-m}(x) = (-1)^m \frac{(l-m)!}{(l+m)!} P_l^m(x)$$

此外 associated Legendre polynomials 也可以藉由 recurrence relations 求得多項式的解，recurrence relations 的規則如下：

$$\begin{aligned} (l-m)P_l^m &= x(2l-1)P_{l-1}^m - (l+m-1)P_{l-2}^m \\ P_m^m &= (-1)^m (2m-1)!!(1-x^2)^{m/2} \\ P_{m+1}^m &= x(2m+1)P_m^m \end{aligned}$$

因此多項式的結果我們就可以經由上述兩種方法得知，以下為 associated Legendre polynomials 的前九項公式，圖 3-7 為其圖形的表示法。

$$\begin{aligned} P_0^0(x) &= 1 \\ P_1^{-1}(x) &= \frac{1}{2} P_1^1(x) \\ P_1^0(x) &= x \\ P_1^1(x) &= - (1 - x^2)^{1/2} \\ P_2^{-2}(x) &= \frac{1}{2 \cdot 4} P_2^2(x) \\ P_2^{-1}(x) &= - \frac{1}{6} P_2^1(x) \\ P_2^0(x) &= \frac{1}{2} (3x^2 - 1) \\ P_2^1(x) &= - 3x(1 - x^2)^{1/2} \\ P_2^2(x) &= 3(1 - x^2) \end{aligned}$$

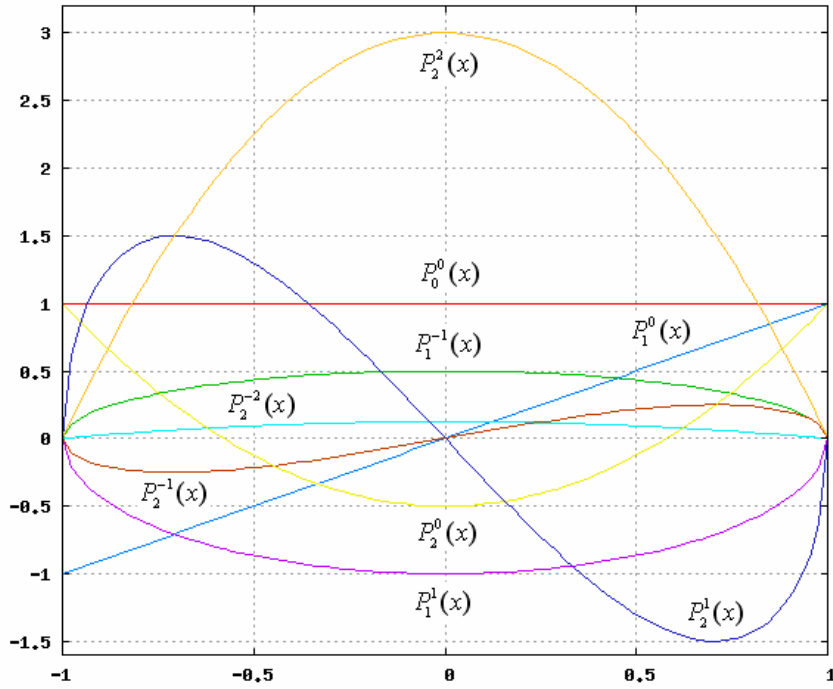


圖 3-7 associated Legendre polynomials 的前九項函數

上面已經詳述了如何求解 associated Legendre polynomials，再來我們將說明 Spherical Harmonic 的定義和 associated Legendre polynomials 的關聯，由數學的理論中(此部分請參照球形座標中的 Laplace's equation)我們得知 Spherical Harmonic 可以用來當作求算球形表面的正交基底函數，由於所求的物體與球體相關因此需將我們一般所熟知的笛卡兒座標系統(Cartesian coordinate)轉為球形極座標系統(spherical polar coordinate)，如下所示，其中 θ 為垂直方向的夾角，而 φ 為水平方向上的夾角：

$$(x, y, z) \rightarrow (\sin \theta \cos \varphi, \sin \theta \sin \varphi, \cos \theta)$$

SH 的公式形式可以經由套用球形座標中的 Laplace's equation 求得，其基本公式如下：

$$Y_l^m(\theta, \varphi) = K_l^m e^{im\varphi} P_l^{|m|}(\cos \theta), \quad l \in N, \quad -l \leq m \leq l$$

其中 K_l^m 是用來正規化函數的參數，其公式表示法如下：

$$K_l^m = \sqrt{\frac{(2l-1)(l-|m|)!}{4\pi(l+|m|)!}}$$

然而在 PRT 中其在意的只有實數的部分，因此所求的 SH 應該是 Real Spherical Harmonics，所以公式可以簡化如下：

$$y_l^m = \begin{cases} \sqrt{2} \operatorname{Re}(Y_l^m), & m > 0 \\ \sqrt{2} \operatorname{Im}(Y_l^m), & m < 0 \\ Y_l^0, & m = 0 \end{cases}$$

$$= \begin{cases} \sqrt{2} K_l^m \cos(m\varphi) P_l^m(\cos\theta), & m > 0 \\ \sqrt{2} K_l^m \sin(-m\varphi) P_l^{-m}(\cos\theta), & m < 0 \\ K_l^0 P_l^0(\cos\theta), & m = 0 \end{cases}$$

下圖 3-8 說明 SH 的前二十五項係數的分佈圖，其中綠色或較亮的部分表示 SH 函數值為正，紅色或較暗的部分表示 SH 函數值為負，因此若 SH 的係數項越多越能表現真實光源的分佈。

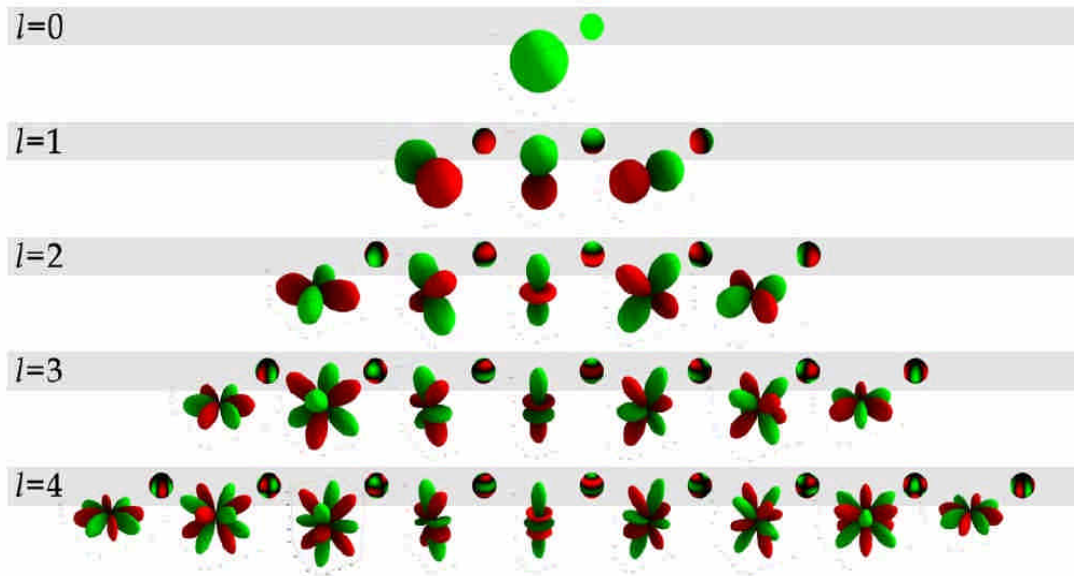


圖 3-8 SH 的前二十五項係數分佈圖[6]

現假設有一函數 $f(x)$ 且其取樣空間為 S ，那麼將可以由下面的積分式求得各取樣空間點在各個 SH 係數的投影值，

$$f_l^m = \int_S f(s) y_l^m(s) ds$$

在求得投影值之後，將各取樣空間的投影值乘上 n 次的 SH 係數再加總就可以重新建構逼近的 $f(x)$ 函數，也就是將原本高頻的訊號轉為低頻的訊號，其中原本 SH 函

$$\tilde{f}(s) = \sum_{l=0}^{n-1} \sum_{m=-l}^l f_l^m y_l^m(s) = \sum_{i=1}^{n^2} f_i y_i(s)$$

數需由參數 l 和 m 表示，現可經由 $i = l(l+1) + m + 1$ 的轉換以 i 表示 SH 函數，因此原本複雜的 $f(x)$ 函數可以藉由內積 f_i 和 $y_i(s)$ 再加總而求得，如上式。

所以假設有一光源函數 $L(\theta, \varphi)$ ，那麼其各取樣空間點在各個 SH 係數的投影值應為下式：

$$l_i = \int_0^{2\pi} \int_0^{\pi} L(\theta, \varphi) y_i \sin \theta d\theta d\varphi$$

那麼光源函數 $L(\theta, \varphi)$ 以 SH 表示，即下如式：

$$\begin{aligned} \tilde{L}(\theta, \varphi) &= l_1 y_1(\theta, \varphi) + l_2 y_2(\theta, \varphi) + \dots \\ &= \sum_{i=1}^{n^2} l_i y_i(\theta, \varphi) \end{aligned}$$

上述已經說明 SH 的基本定義和原理，在此將繼續說明 PRT 的觀念，在 PRT 中為了求算 3D 物體表面的光源分佈，因此其假設物體每個頂點皆由一無限遠的球體所包裹，然後計算球體四面八方入射到頂點的入射光和求得物體幾何特性的轉換函數(transfer function)，那麼就可以得知反射的光，其計算的公式如以下的積分式：

$$\int_0^{2\pi} \int_0^{\pi} L(\theta, \varphi) t(\theta, \varphi) d\theta d\varphi$$

其中 $L(\theta, \varphi)$ 為入射光函數， $t(\theta, \varphi)$ 為轉換函數，那麼如果可以求得入射光函數和轉換函數的 SH 係數，那將可以簡化求全域照射下的光源變化，因為 SH 為正交函數，所以反射的光只需先求內積再加總即可，而每當光源轉變後只需重新計算

$$\int_0^{2\pi} \int_0^{\pi} \tilde{L}(\theta, \varphi) t(\theta, \varphi) d\theta d\varphi = \sum_{i=1}^{n^2} L_i t_i$$

光源函數和轉換函數的乘積即可，所以可以即時的獲得光源在全域照射下的變化。

在光源照射的環境下，一般物體所反射的光亮函數，應如下式所示，其中 0 表示直接照射，1 表示經由一次反彈，以下依此類推，

$$L(p \rightarrow \overset{u}{d}) = L_0(p \rightarrow \overset{u}{d}) + L_1(p \rightarrow \overset{u}{d}) + \dots$$

那麼為了求得反射光的轉換函數，因此需將反射光的組成函數拆解成入射光函數和轉換函數，下列為直接入射光的反射光組成函數，其中 f_r 為物體表面的反射特

$$L_0(p \rightarrow \overset{u}{d}) = \int_{\Omega} f_r(p, \overset{v}{s} \rightarrow \overset{u}{d}) L_{env}(\overset{v}{s}) H_{np}(\overset{v}{-s}) V_p(p \rightarrow \overset{v}{s}) d\overset{v}{s}$$

性， L_{env} 就是投影在將 p 點包裹的無限遠球體的入射光度(incident radiance)，也就是由 p 看出來的環境影像或環境貼圖(environment map)， H_{np} 是一般 CG 光源模型的 cosine 項目或點 p 的幾何特性，而 V_p 就是點 p 的視覺的特性，換句話說就是由點 p 出發是否直接被其他物體遮擋，現假設物體為完美的漫射面，那麼直接入射光的反射光組成函數可以覆寫如下：

$$L_0(p) = \frac{\rho_p}{\pi} \int_{\Omega} L_{env}(\overset{v}{s}) H_{np}(\overset{v}{-s}) V_p(p \rightarrow \overset{v}{s}) d\overset{v}{s}$$

在上式中，物體為完美的漫射面所以對各方向的漫射是一致的，因此給定其反照率(albedo) ρ_p ，其範圍介在 0 和 1 之間，而式子中的 L_{env} ，因為是光源函數所以如果將其以基底函數做投影，所得的光源函數之近似函數將為 $L_{env}(\overset{v}{s}) \approx \sum_i l_i y_i(\overset{v}{s})$ ，再藉由將 $\sum_i l_i y_i(\overset{v}{s})$ 重新代入反射光組成函數中，則可以得到包含轉換函數的反射光組成函數，

$$L_0(p) = \frac{\rho_p}{\pi} \sum_i l_i \int_{\Omega} y_i(\overset{v}{s}) H_{np}(\overset{v}{-s}) V_p(p \rightarrow \overset{v}{s}) d\overset{v}{s}$$

到此得知投影至 SH 基底函數的直接入射光的反射光函數，同時也將反射光函數轉

變成光源係數和轉換函數的組成，因此將光源係數獨立出來就可以獲得直接入射光的反射光函數的轉換函數 t_p^0 ：

$$t_p^0 = \sum_i \frac{\rho_p}{\pi} \int_{\Omega} y_i(s) H_{np}(-s) V_p(p \rightarrow s) ds$$

至此已推導出直接入射光的反射光函數，接下來將說明經由反彈之入射光的反射光函數如何求得，現舉完美漫射面的一次反彈為例，其反射光函數如下：

$$L_1(p) = \frac{\rho}{\pi} \int_{\Omega} L_0(q(s)) (1 - V_p(p \rightarrow s)) H_{np}(-s) ds$$

上式中的 q 表示由 q 點反彈至 p 點的光線，而 $1 - V_p$ 表示會將光線反彈的只有在被遮蔽下，現在再一次將光源係數去除，那麼就可以得到一次反彈的轉換函數 t_p^1 ，並類推出完整的轉換函數 t_p 。

$$\begin{aligned} t_p^1 &= \sum_i \frac{\rho}{\pi} \int_{\Omega} t_q^0 (1 - V_p(p \rightarrow s)) H_{np}(-s) ds \\ t_p^b &= \sum_i \frac{\rho}{\pi} \int_{\Omega} t_q^{b-1} (1 - V_p(p \rightarrow s)) H_{np}(-s) ds \\ t_p &= \sum_{b=0}^{B-1} t_p^b \end{aligned}$$

現假設欲求全域照射下一物體的表面光影分佈，則如上述，可以經由 SH 的投影方式求得轉換函數 t_p ，再求得目前光源函數投影在 SH 上的係數值 l_p ，那麼最後即時的 rendering 公式如下，只需加總 l_p 和 t_p 的內積值即可。

$$\int_S L_p(s) t_p(s) ds \approx \int_S \underline{E}_p(s) \overline{t}_p^0(s) ds = \sum_{i=1}^{n^2} l_{p,i} t_{p,i}$$

在求得頂點的近似值之後，再以 Gouraud shading 的方式求得頂點間的內差值，上述的範例是假設光源只有單一的頻譜，但實作上可以將光源視為具有紅光、綠光和藍光等三種頻譜，且在一般 PRT 的使用上只需要使用前二十五項的 SH 基底函數就可以求得近似的光源效果。

第四章 系統設計

4.1 系統架構

為了產生擬真的植物影像我們提出一個名為 SimEco 的兩階段系統，其第一個階段是文法剖析器，第二個階段是 rendering 引擎，而為了產生影像我們的系統必需引入兩種資料檔案，一個為植物文法的描述檔，另一個為植物器官的模型檔，下面圖 4.1 是我們系統的基本架構，其中文法剖析器是用來處理植物文法的描述檔，它總共包含三個處理流程——場景載入器、器官載入器以及文法規則載入器，場景載入器用來處理文法描述檔中記載場景設定的部分，其中大概是關於光源設定、地表設定以及一些用來計算環境對植物影響的環境參數，器官載入器則是負責從文法描述檔中找出關於植物本體的敘述，包括植物生長的位置、方向、文法符號定義以及各植物器官的細節描述，而當它讀到關於器官描述時它會先到 3D 器官模型資料庫中搜尋器官的模型檔案，然後載入到系統中，而文法規則載入器則是用來讀取植物文法生長規則，例如：文法生產規則和起始字串。

至於 rendering 引擎中共有五種基本的操作過程，首先是文法疊代器，文法疊代器是用來處理在文法規則載入器中，所分析完的文法規則資料，它會依據規則中的描述一步驟一步驟的疊代文法，在計算完後，環境計算器會根據場景載入器中所得的環境影響參數來計算各器官受環境影響的結果，關於模型混合器的部分，它是負責處理連結器官載入器中所得的器官模型檔案，以及植物成長器所計算完的植物資訊，如：器官位置、方向、大小等，除了令器官模型檔案和植物資訊產生關聯外，它還會在此加入某些隨機資訊使得植物外觀的表現更顯豐富，而基本 rendering 器則是運用 DirectX 所提供的 Gouraud shading 來進行簡單的植物影像呈現，當植物成長到適當的程度後，我們可以將基本 rendering 中的植物資訊搭配 PRT 的材質描述傳

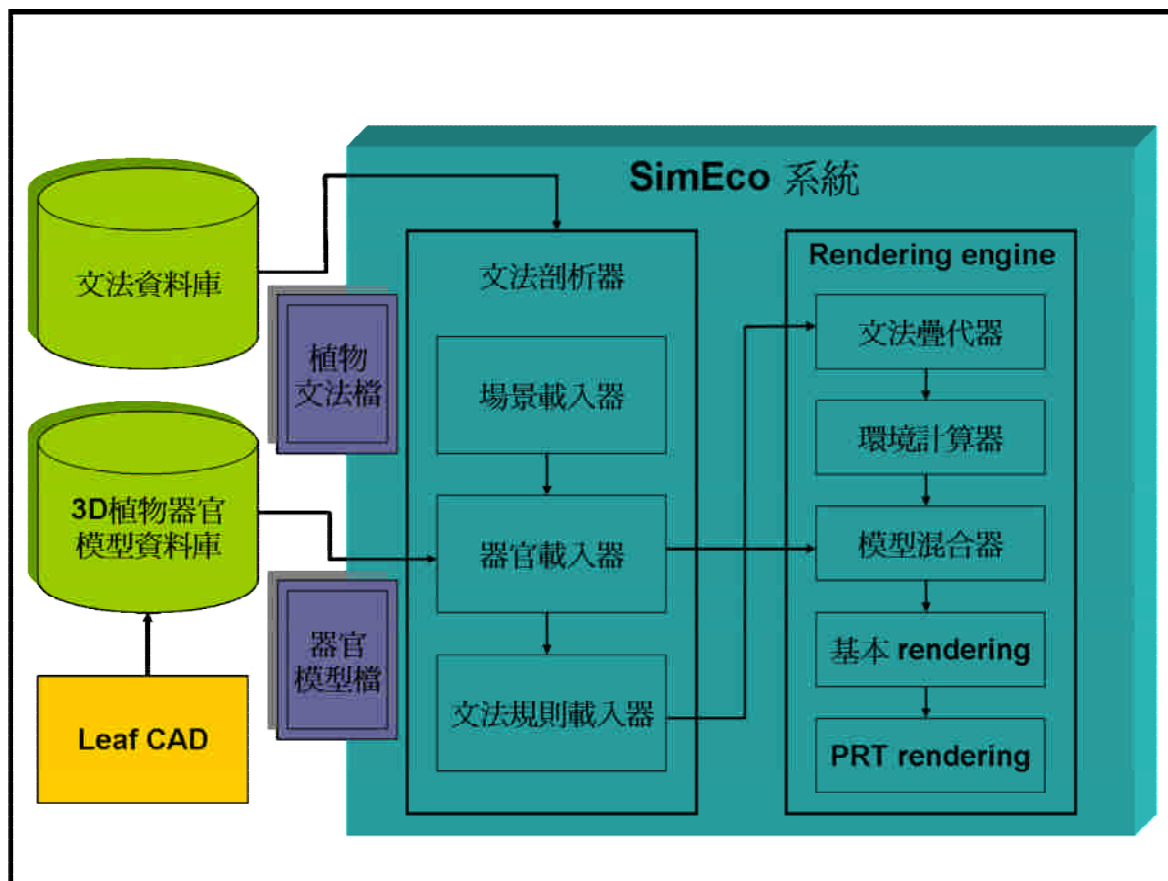


圖 4.1 系統的基本架構

送到 PRTrendering 器中，以進行更高階的 PRT rendering 效果，以上就是我們系統中各部分的功能描述，下面幾節我們會更進一步說明系統的流程是如何。

4.2 系統流程和操作流程

下面我們會說明 SimEco 基本的系統流程和操作流程，在系統流程中我們會再次說明各單位元件的基本功能和執行方法以及資料的傳輸，而在操作流程中我們會著重在使用者操作系統時，系統的決策過程。

在執行 SimEco 時，首先需要載入文法的描述，然後文法剖析器會將文法描述檔中所載的資訊區分成三種資料，一是關於場景的描述或場景參數的部分，二是說明植物主體器官的訊息，三是記載此株植物生長的文法規則，處理的順序是先場景

標籤、植物主體標籤，最後是文法生長標籤，詳細的執行過程如圖 4-2，在執行完文法的剖析後，會將所得的資料傳送到 rendering 引擎中，首先會由文法疊代器取得資訊，然後將文法中的規則疊代一次，再將其傳送到環境計算器中以計算環境對植物的影響，所算出的植物資訊再傳到模型混合器中和原本從 3D 器官模型資料庫中所得的器官模型做連結，做完連結後系統會將這些資訊移至基本 rendering 器中，然後基本 rendering 器會以 DirectX 的 Gouraud shading 進行簡單的植物 rendering，除了 Gouraud shading 的 rendering 外，系統也可以選擇採用 PRT 的方式來進行 rendering，簡單的系統流程可以從圖 4-2 中窺見。

系統的使用方式首先使用者會開啟文法載入視窗然後載入植物文法，當載入完後系統會將檔案傳送到文法剖析器中進行剖析，然後會再送到 rendering 引擎中做 rendering，rendering 的方式是每次從文法疊代器中獲得一次疊代的結果然後進行 rendering，而每次執行完一步驟的基本 rendering 後，系統會判斷是否停止基本 rendering，若停止基本 rendering 後，有兩種方式可以恢復 rendering，一種是再次執行文法疊代器以恢復基本 rendering，另一個方式是進行 PRT 的方式來進行更進一步的 rendering，在執行完 PRT 模擬後，一定需再次執行文法疊代器以進行下一次植物成長的疊代，然後再傳送到進行基本 rendering 器中以進行 rendering，詳細的系統操作過程如圖 4-3 所示。

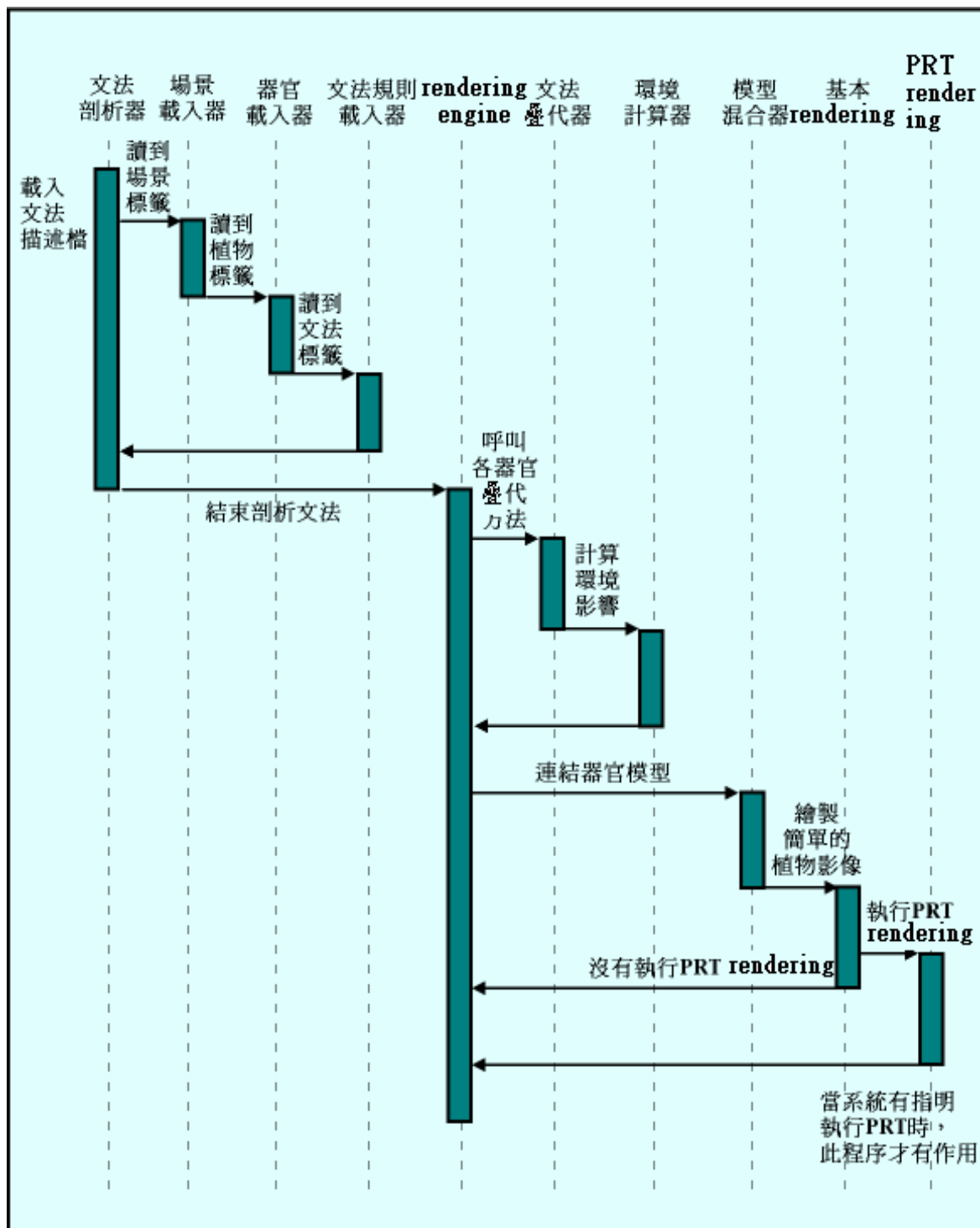


圖 4-2 簡易的系統流程

SimEco系統操作流程

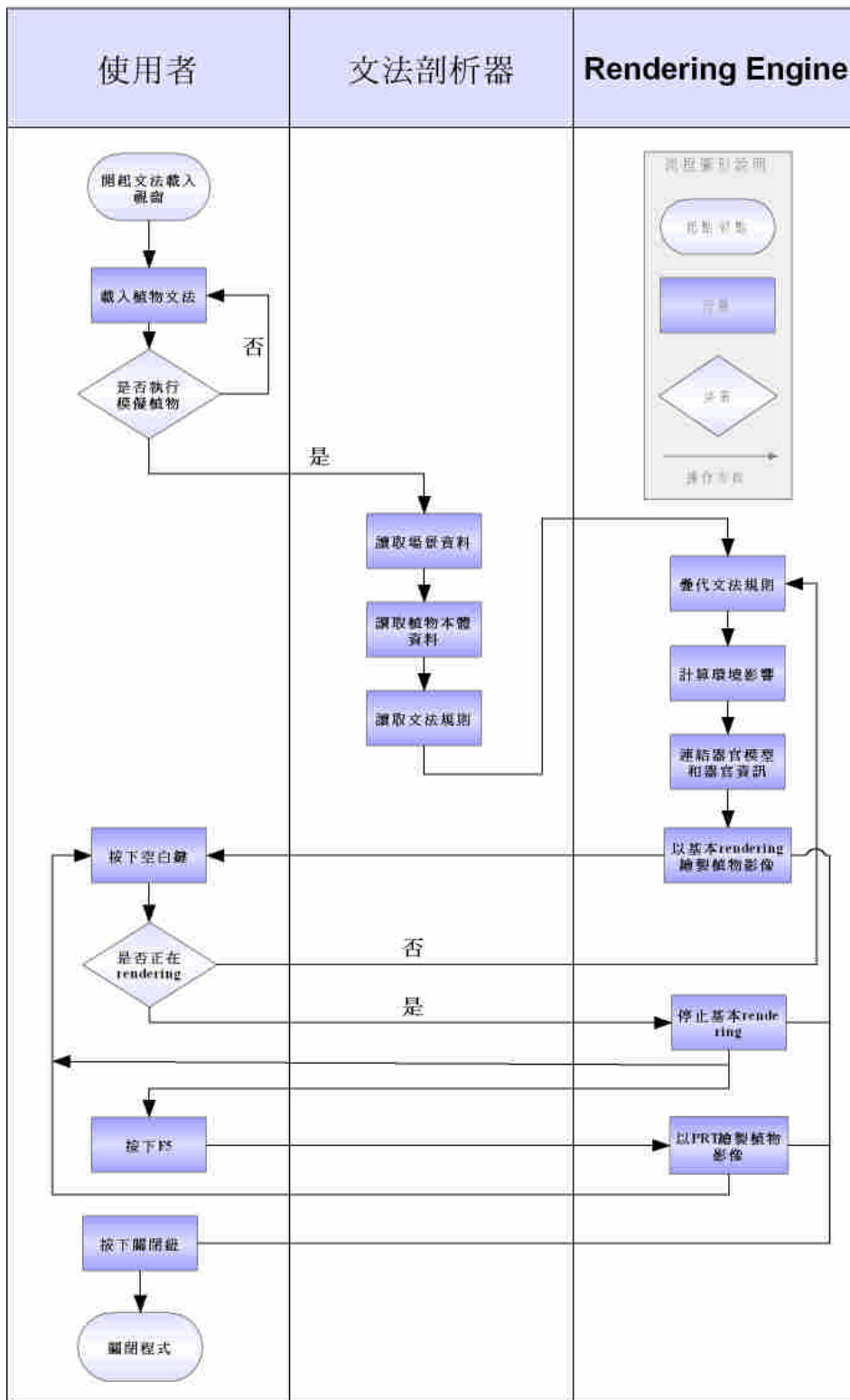


圖 4-3 系統的操作流程

第五章 系統實作

5.1 匯入的檔案

在我們的系統中需要載入兩種檔案，一種是關於文法描述的檔案，另一種是植物器官模型的資料檔，其中文法描述的檔案可以經由使用者自行編撰得知，而器官檔案中主要有三種模型 — 莖、葉子和花，其中葉子模型是由我們自行開發的 LEAF CAD 中獲得，花模型是由 3D MAX 中得到，而莖模型則是由一組簡意的柱狀模型，在執行期間依據莖的位置資訊形變、接合而來，關於模型的取得會在 5.1.2 節中描述。

5.1.1 文法描述檔

文法描述檔，是一個階層式的架構中，主要有三個大架構，場景部分、植物本體和文法規則，以下是一個文法描述檔的簡單範例：

```
環境關鍵字 版本名稱 "系統名稱"{
    環境參數;
    植物關鍵字 "植物名稱"{
        植物參數;
        器官關鍵字 {
            器官階段 階段數 {
                器官參數;
            }
        }
    }
    g
    g
    g
    文法關鍵字 {
        文法參數;
    }
}
```

其中環境參數就是用來描述場景中環境的特性值，而器官參數則是描述器官的生長特性、材質特性以及 PRT 的特性，通常一株植物的器官關鍵字只會出現一次，而器官的階段數目則有數個，至於文法參數，就是記載文法的起始字串以及各階段數的疊代規則。

5.1.2 3D 植物器官模型的取得

首先我們先說明器官中花的取得方式，以花的取得來說，我們為了不加重系統的負擔以及增加系統的彈性，所以我們預先以 3D MAX 7.0 來繪製花的模型，然後再將模型載入到系統中，下圖 5.1，是我們用 3D MAX 製作的玫瑰花影像。

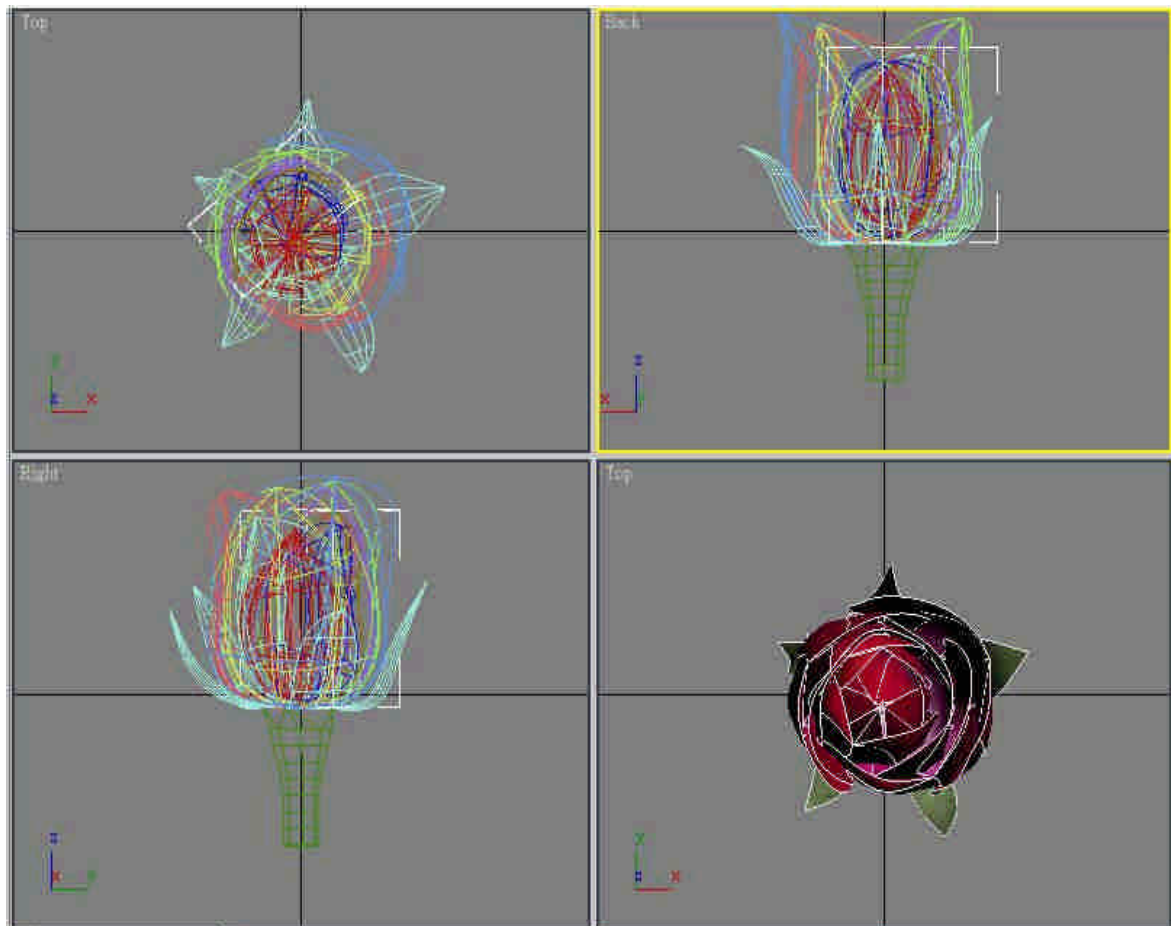


圖 5.1 以 3D MAX 製作玫瑰花

而葉子的取得方面，是以我們自行發展的 LEAD CAD 從實際的植物影像中取得的，首先在 CAD 中會載入想要建模的植物葉子影像，然後我們會以多線段的 2D Bezier 曲線拼湊出葉子影像的大致輪廓，下圖 5.2 是三種不同葉子形狀的 Bezier 建模影像，藍線的部分是 Bezier 曲線的控制點，紅線的地方是 Bezier 曲線，其中圖 5.2(a)、(c)是以兩條多線段的 Bezier 曲線組合而成的，而圖 5.2(b)則是由四條多線段

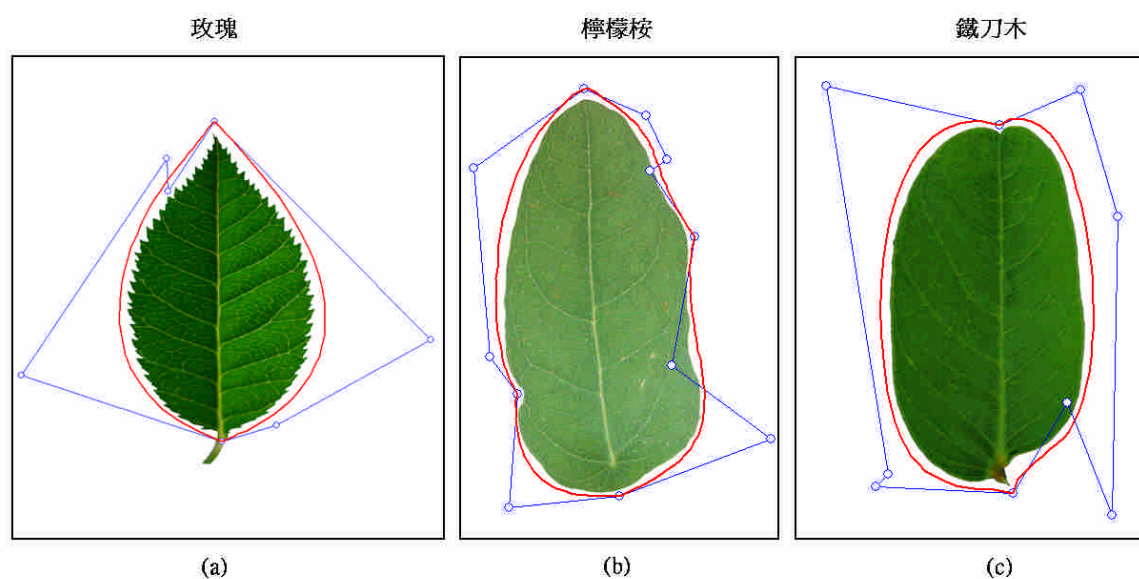


圖 5.2 以 Bezier 曲線拼湊的葉子輪廓影像

的 Bezier 曲線所組合而成的，目前我們的 LEAF CAD 只能勾勒出封閉區間的植物外觀，且由於我們在為葉子建立模型時是以等間格的方式自動去挑選，所以為了避免端點或具特性的拐點(曲率過大的點)可能會被忽略，因此我們輔以手動的方式再去挑選我們認為有必要的頂點或拐點，如此一來將可以在葉子的模型中強調出此葉子應具備的特性，如圖 5.3 和圖 5.4 所示，我們不難發現在圖 5.3 中所強調的頂點或拐點在圖 5.4 的葉子模型圖中都有明確的呈現，葉子除了形狀各有不同外，其和葉柄的連結也各異，在圖 5.4(a)、(b)、(c)中各有一個被紅色圓圈所圈選的淡藍色的頂點，其表示的就是葉子和莖的接點所在，藉此可以產生各種不同的連結影像，到此 LEAF

CAD 已經可以產生 2D 的葉子模型影像，如圖 5.5(a)，但由於我們所需要的葉子影

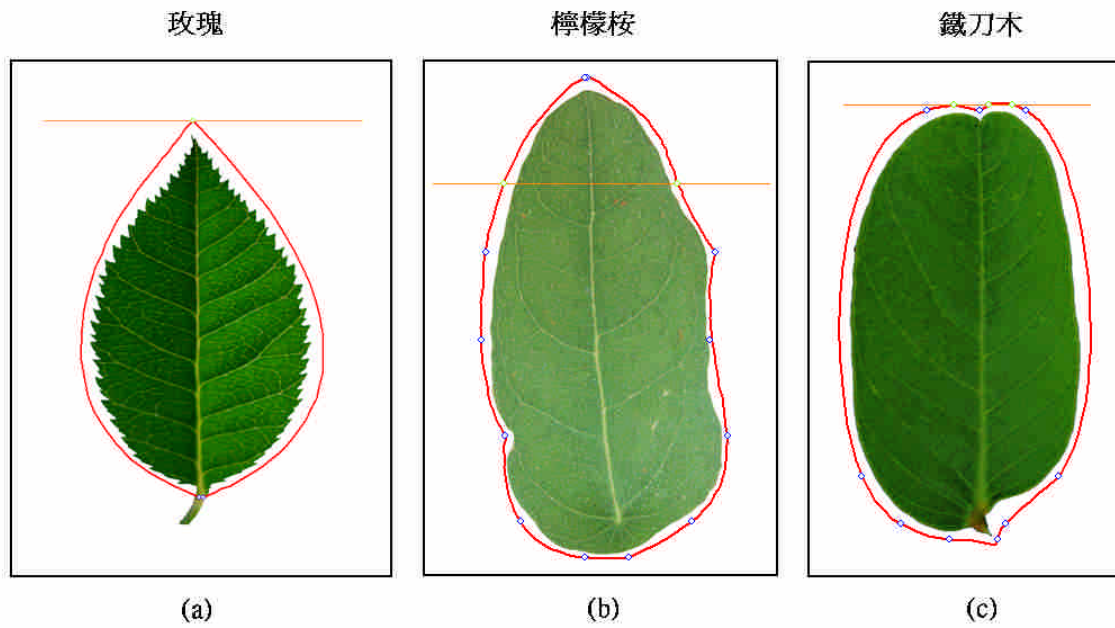


圖 5.3 標示葉子外觀之頂點或拐點

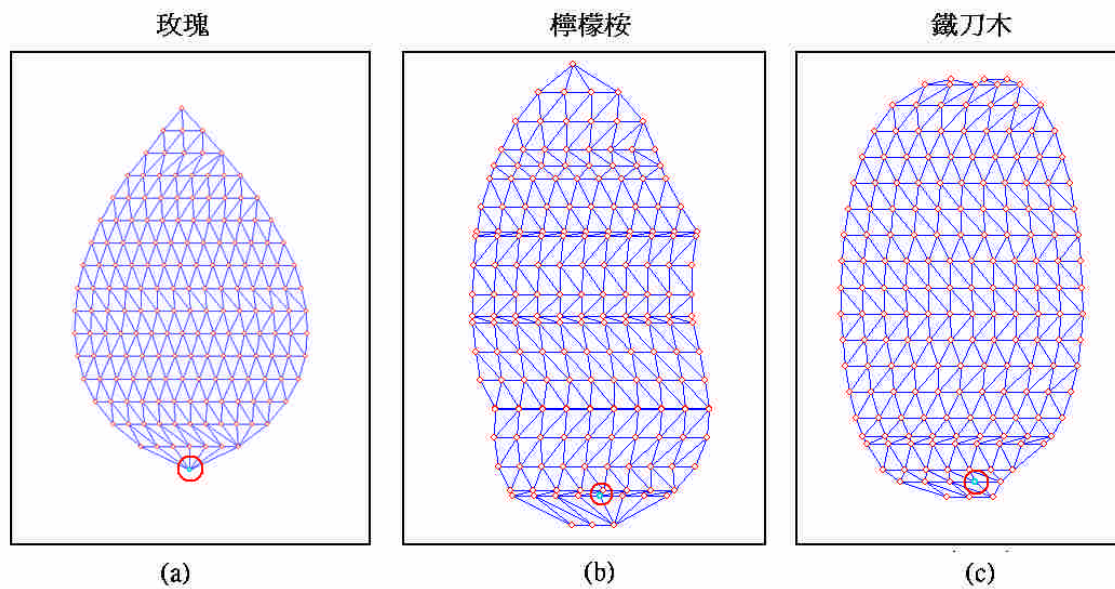


圖 5.4 葉子網格及連接點

像是 3D 的變型影像，因此我們在 LEAF CAD 中引入 3D 的 Bezier 曲面以強化葉子

的形變效果，如圖 5.5(b)、(c)、(d)中，我們藉由控制 Bezier 曲面的控制點來形變原本圖 5.5(a)的葉子模型，然後將變型後的結果保存在 3D 植物器官模型資料庫中，因此藉由 LEAF CAD 我們將可以取得多樣性的植物葉子模型。

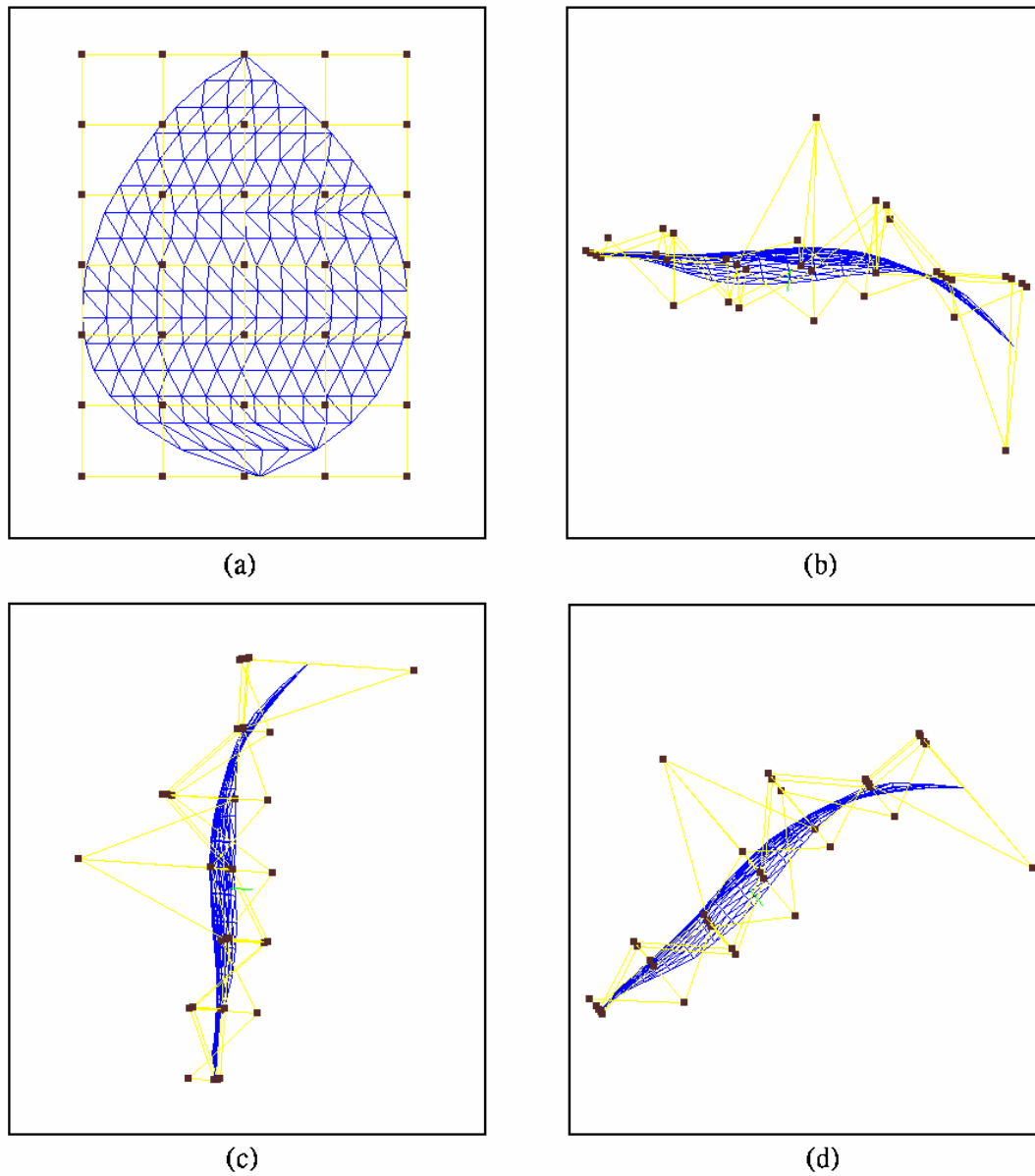


圖 5.5 葉子模型的形變

5.2 植物外觀的呈成

這一節主要說明的是我們系統中是如何貼合出植物的外觀，在自然環境中植物的變化是非常劇烈的，因此如果要真實的呈現植物的外觀將是難以達到，所以在本系統中我們採用折衷的方式，即我們事先計算出基本的植物器官形狀，如 5.1 節所述，然後在實際要產生影像時再將記載植物結構的資料加以連結，以貼合出相應的植物外觀，如此不但可以減低植物生成的複雜度同時也可以加速影像的產生，下面幾個小節我們會分別介紹莖、葉子以及花模型的貼合方式。

5.2.1 莖的貼合

經由觀察植物的莖我們發現植物的莖同時具有平滑和生硬的特色，在接點與接點中間，植物的莖通常表現出平滑的特性，而在接點的兩端一般會有較生硬的轉折，另外在植物分枝和主軸的接點上會有特殊狀的連接，因此關於莖的方面我們主要分成兩種結構來處理，一種是莖的主軸結構(包含植物主體的莖以及分枝後的莖主軸)，另一種是莖的分枝結構，下面我們將分別說明這兩種模式的貼合方式。

首先關於莖的主軸方面，因為要表現出植物莖的平滑和生硬的效果，因此為了強化接點的生硬效果，所以我們思考將組成莖主軸的線段中的每一小段都視為獨立的個體，然後再加強每一線段的平滑度以增加莖的平滑度，最後再將線段彼此連接，系統中實際的貼合如上圖 5.6 所示，首先假設在文法疊代器和環境計算器中已經計算出莖的連接方式，如圖 5.6 中橘色圓點的連線，然後將各線段的端點再延伸一小段以當作控制點，如圖 5.6 中藍色的方形，因此就可以藉由如此的三個控制點(藍點和橘點)的 Bezier 曲線來增加各線段的平滑度，最後再將所得的各線段相連，那麼接點上就可以表現出生硬的感覺，然後再將莖的材質繞著圖 5.6 中紅色的線段做貼圖，

系統中實際的影像如圖 5.7 中所示，其中圖 5.7(a)是還沒做 Bezier 平滑前，圖 5.7(b)則是做完平滑後的結果。

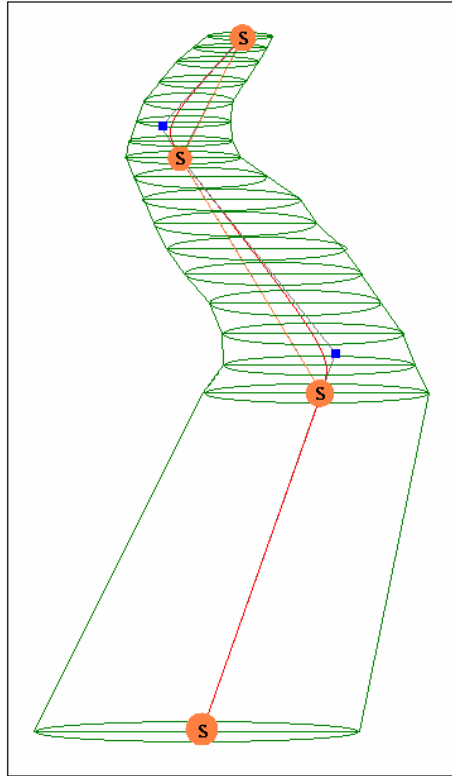


圖 5.6 莖主軸的生成

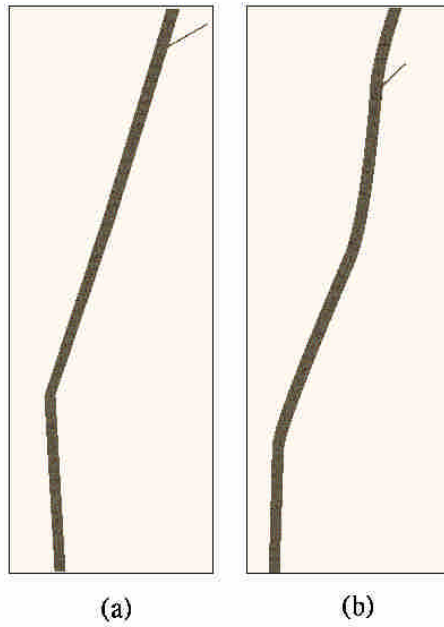


圖 5.7 系統中莖主軸的貼合影像

至於莖的分枝結構，如圖 5.8 所示，其中圖 5.8(a)是分枝水平延伸的結果，而圖 5.8(b)是分枝向上延伸的結果，由圖 5.8 中我們可以發現分枝和主幹的連接是頗為複

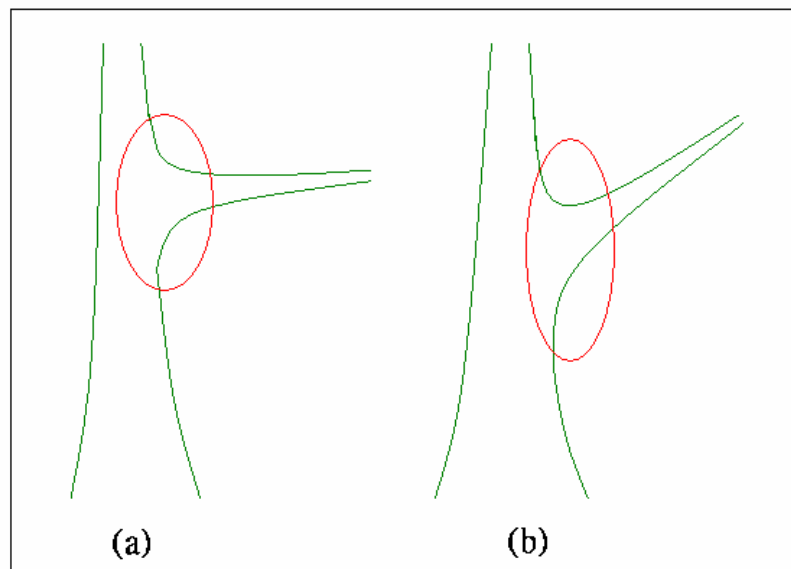


圖 5.8 莖分枝的示意圖

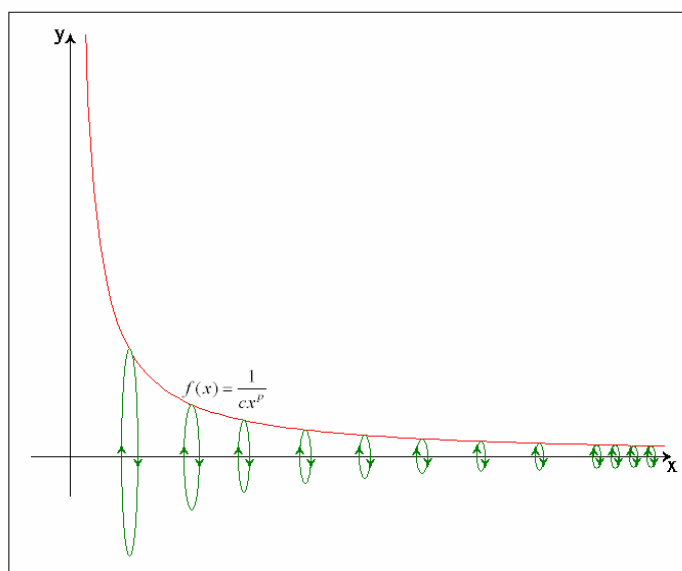


圖 5.9 函數 $f(x) = \frac{1}{cx^p}$

雜的議題，但本系統的目標並不是以解決這個問題為前提，所以我們只是提出一項簡單的方法來產生大致相似的影響，那麼下面就繼續來探討在我們的系統中我們是如何來實作這個問題的，從圖 5.8(a)中可以發現植物的分枝結構是類似函數 $f(x) = \frac{1}{cx^p}$ 的曲線，如圖 5.9 所示，因此為了表示植物分枝的效果所以系統沿著 $f(x) = \frac{1}{cx^p}$ 的曲線去貼合出分枝的形狀，然後再連結所得的莖主幹和分枝結構，如圖 5.10、5.11，其中圖 5.10 是簡單的連結示意圖，而圖 5.11(a)是還沒套用 $f(x) = \frac{1}{cx^p}$ 函數前的影像，圖 5.11(b)則是套用來 $f(x) = \frac{1}{cx^p}$ 函數後的分枝影像。

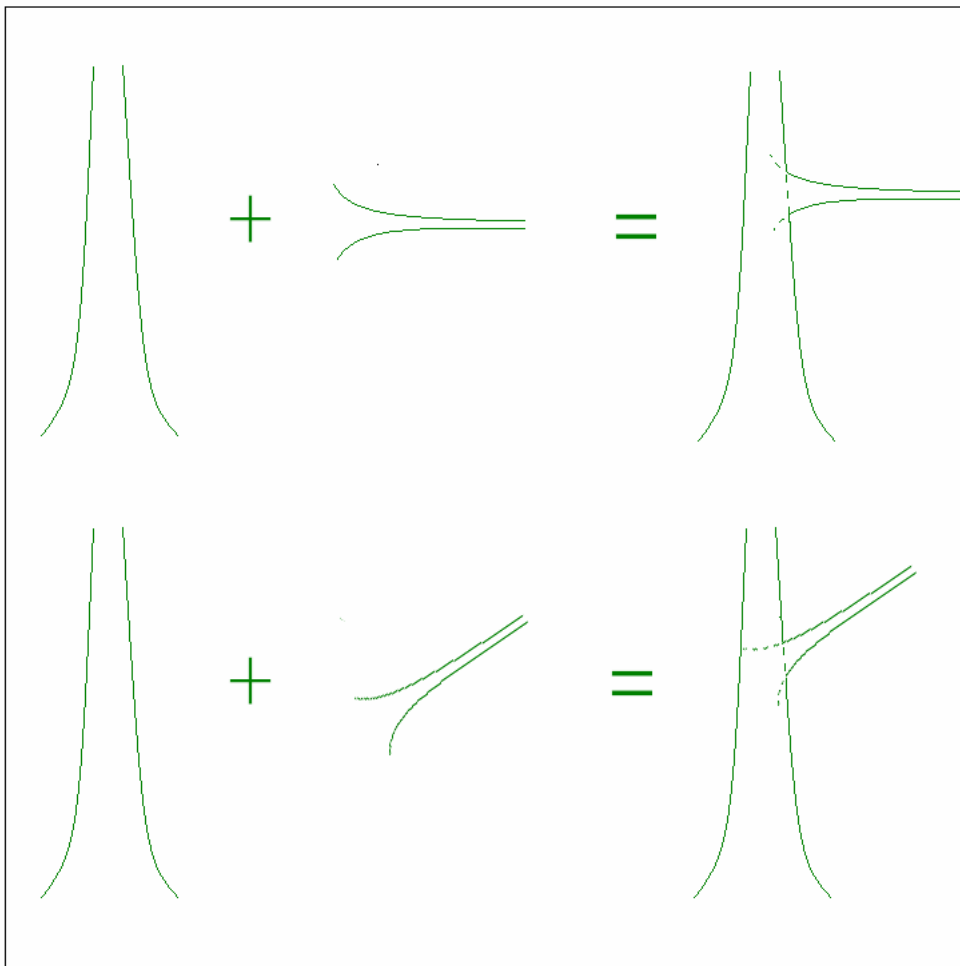


圖 5.10 莖主幹和分枝連結的示意圖

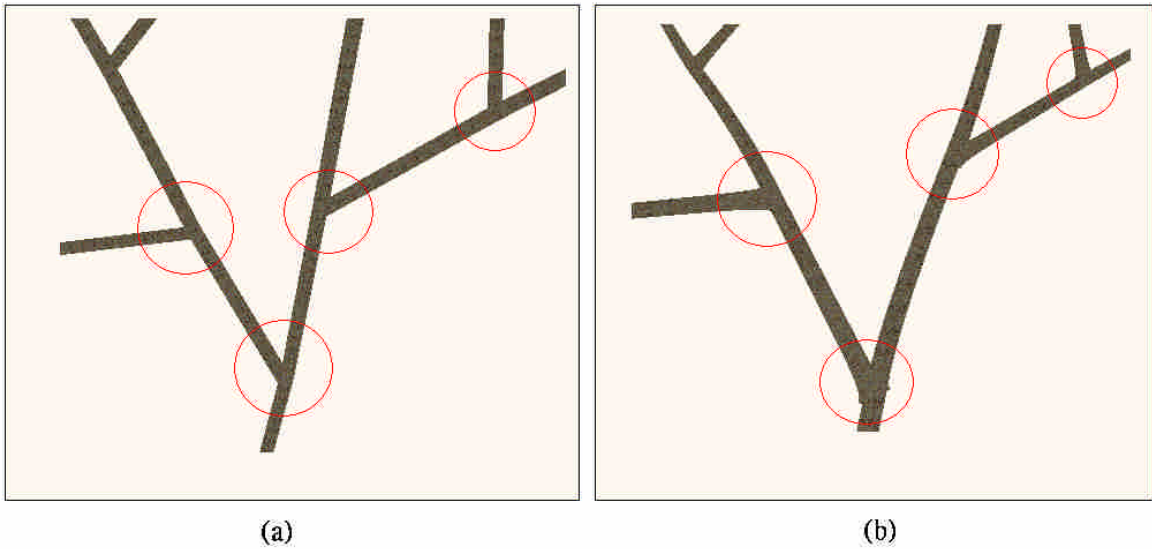


圖 5.11 系統中莖主幹和分枝連結的貼合影像

5.2.2 葉子的貼合

以葉子的影像貼合來說，我們為了表達植物葉子的多變性因此我們藉由對一種植物給定多種的葉子模型，然後在系統執行時再以增加隨機值的方式來強調葉子的多樣化，並且在系統中我們採用 Alpha Blending 的技術來強調葉子邊緣的鋸齒效果，圖 5.12(a)是原本的實際葉子影像再經由 LEAF CAD 操作所得的模型重建而來，至於圖 5.12(b)、(c)、(d)則是從基本模型在經由多次的 Bezier 曲面形變而控制得來。

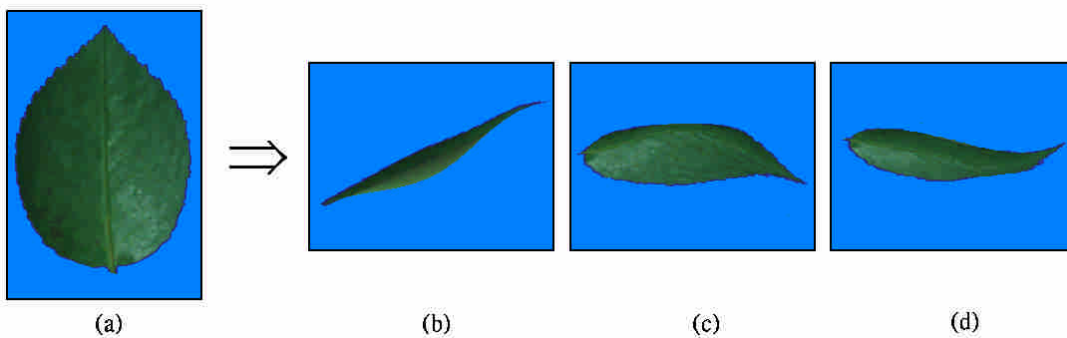


圖 5.12 葉子的多種形變影像

5.2.2 花的貼合

植物給人的感官刺激除了葉子的多變性和豐富度外，就屬花朵最為迷人，然而要以程式去計算完整的開花過程似乎是不可能的，因此我們經由截取開花過程中一些有意義的片段然後在以 3D MAX 重建當時的外觀，再匯入到系統中以模擬草本植物開花的過程，圖 5.13 是我們觀察玫瑰花所製作出來的四個開花過程的漸近模型。

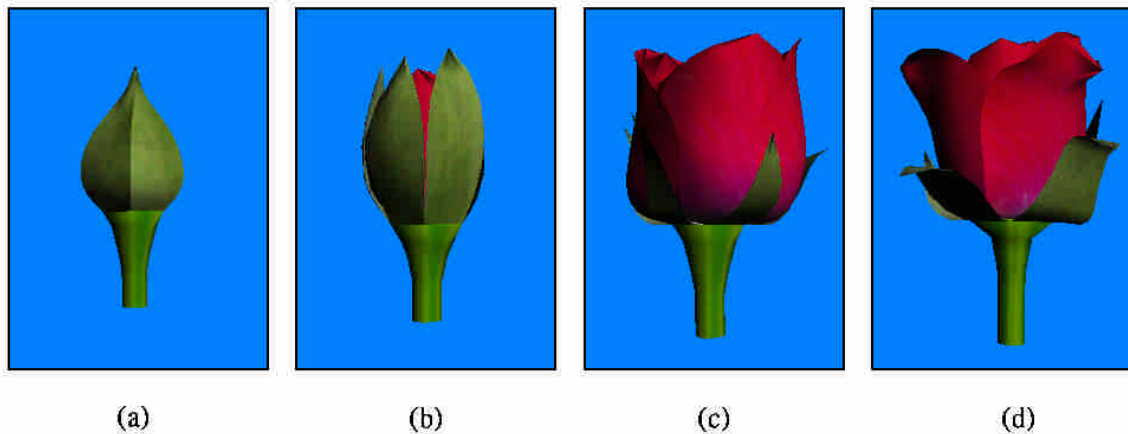


圖 5.13 玫瑰開花的漸近影像

5.3 PRT 在 SimEco 中的使用方式

在 SimEco 中，為了加強影像的擬真度所以我們引進 DirectX 所提的 PRT 方法來模擬植物的影像，因此本節我們將說明 DirectX 中 PRT 的使用方式，如圖 5.14，首先假設我們在混模型合器中已經計算出要 rendering 的植物模型，那麼在進行 PRT 模擬時，可以藉由載入 X 檔案或自行以 `D3DXCreateMesh()` 創建欲模擬的模型，但因為 SimEco 是在執行時才計算植物模型，因此我們需以 `D3DXCreateMesh()` 創造出適用於 DirectX 的 `ID3DXMesh` 模型，然後再將植物模型的資料填入 `ID3DXMesh` 模型中，當取得植物模型後接著還需要初始化每一個材質的物理特性，然後才能進行

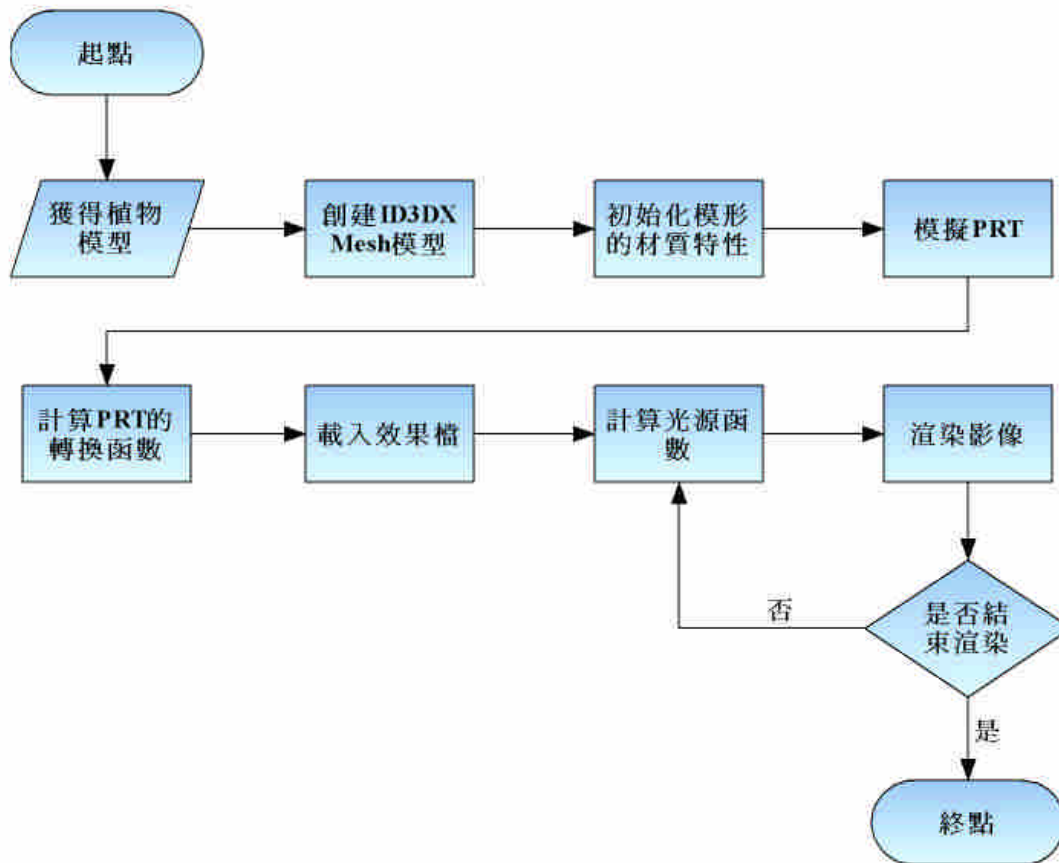


圖 5.14 PRT 模擬的流程

PRT 模擬，在此我們將初始化的資料分成一般項和特殊項，一般項是記載場景中所有物體通用的模擬資料，如表 5.1 所述，表 5.1 中只記載部分一般項的參數說明，至

參數名稱	說明
dwOrder	以幾次係數去逼近原函數
dwNumRays	向球體發散的光線數
dwNumChannels	以多少頻譜模擬光線
dwNumBounces	計算幾次的光線碰撞
fLengthScale	光源影響的範圍

表 5.1 部分 PRT 設定的一般項

於詳細的參數細節請參閱 DirectX9.0 的範例“PRTDemo”中關於參數的設定，而特殊項則是記載物體的物理特性，如：吸收度、透光度，散光特性…等，詳細細節也請參照“PRTDemo”，一般而言如果欲模擬的模型是單一的材質特性的話，那麼只需設定模型中關於 PRT 的一般項和特殊項即可，但在 SimEco 中我們想要表達的是植物的材質特性，因此對於植物的材質我們另外載入相映的 BMP 檔案以儲存影像中 RGBA 的資料，載入的方法就是使用 `D3DXCreateTextureFromFileEx()` 來取得 `IDirectTexture9` 的材質資料，然後系統就可以使用 `PRTSimulationThreadProc()` 來開始進行模擬 PRT，如 PRT 技術理論中所載的，在系統在進行模擬時其實是先求出 PRT 的轉換函數，即物體對光源函數的幾何特性，接著當系統計算完轉換函數後，會用 `LoadEffects()` 來載入效果檔以配合 GPU 來加速 PRT 的 rendering 計算，上述的這些過程都是在模擬過程中先行運算的，然後在系統執行期會不斷以 `UpdateLightingEnvironment()` 計算光源的變化，方法則是在執行期求得變化後的光源函數再將先前計算的轉換函數一起傳送到 GPU 中做 rendering 運算，在 rendering 的同時，系統會計算物體的幾何模型和材質特性以取得物體材質的明暗變化，其中材質的特性可以使用 `SetValue()` 的方式將單一材質特性設定到效果檔中以模擬物體的材質特性，或使用 `SetTexture()` 的方法以產生複雜的材質特性，如下圖 5.15 中，(a) 為單一的 PRT 材質而 (b) 為附加複雜葉子影像的葉子 PRT 模型，這就是 SimEco 中使用 DirectX 的 PRT 的流程，關於上述過程中所用到的函數，請再次參照“PRTDemo”範例。

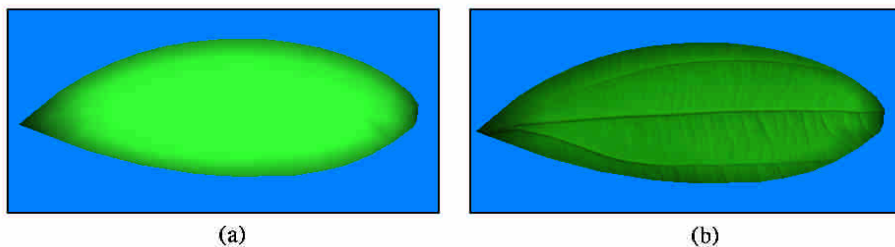


圖 5.15 PRT 材質說明圖

5.4 結果影像

結果影像主要分成兩個組成部分，一個是是以 Leaf CAD 所建構的多組葉子局部影像，另一個是呈現光影變化的 PRT 成果影像，所有的影像都是即使的由 SimEco 根據 L-Grammar 所 rendering 而來的，下圖 5.16 為玫瑰的葉子影像，鋸齒邊緣的效

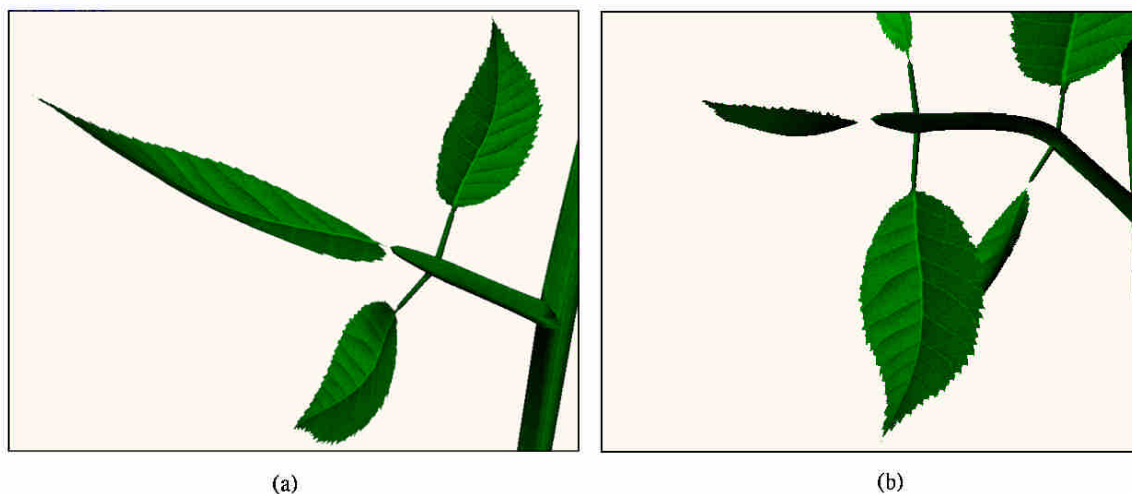


圖 5.16 玫瑰葉子成果圖，(a)、(b)是同株玫瑰不同位置上的葉子影像。

果是由 Alpha Testing 得到的結果，下圖 5.17 為日日春在進行 PRT 計算後的葉子影

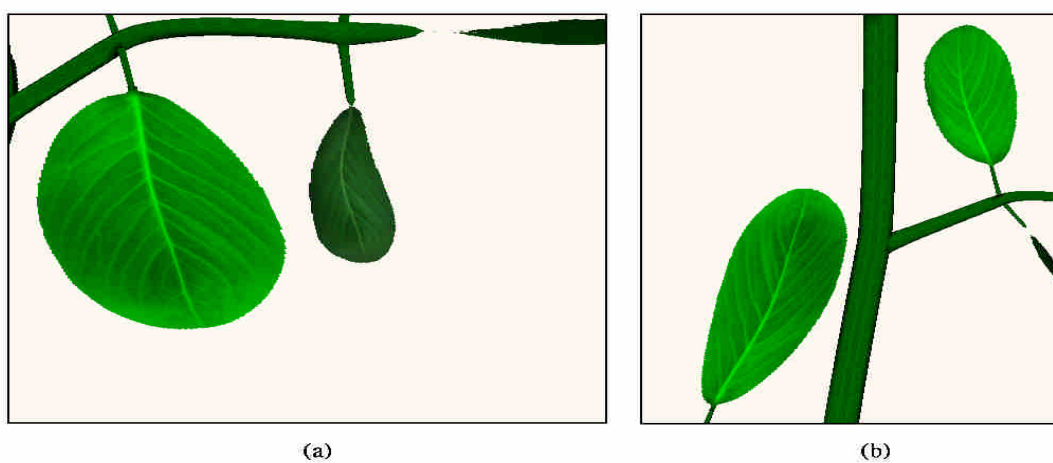


圖 5.17 日日春葉子成果圖，(a)、(b)是同一株日日春不同位置上的葉子影像。

像，下圖 5.18 是檸檬桉的葉子成果影像，而圖 5.19 是紅刺蔥的葉子結果影像，檸檬桉的接點乃是依據 Leaf CAD 在生成時所選擇的连接點，然後再根據连接平面來接

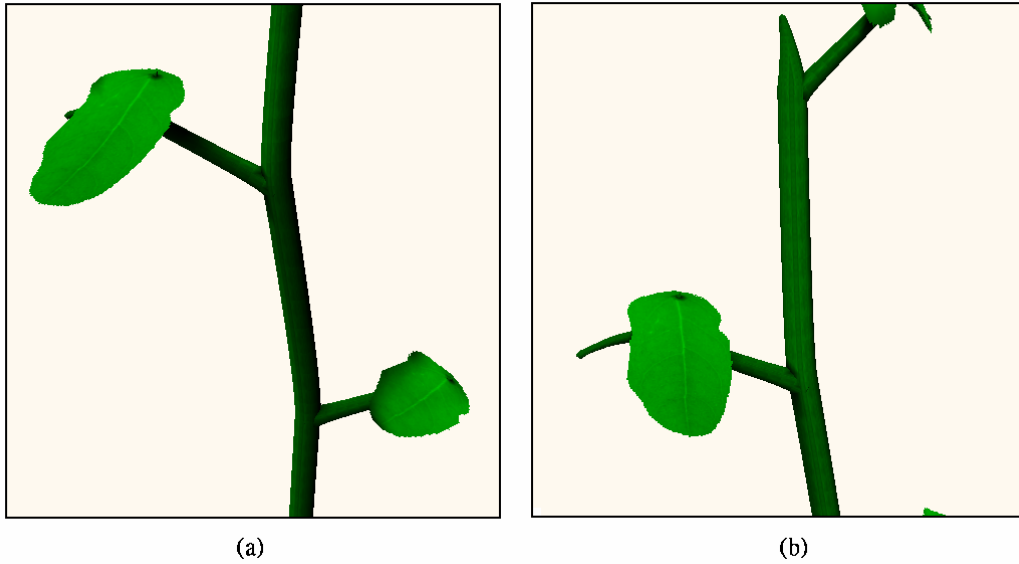


圖 5.18 檸檬桉葉子成果圖，(a)、(b)是同一株檸檬桉不同位置上的葉子影像。

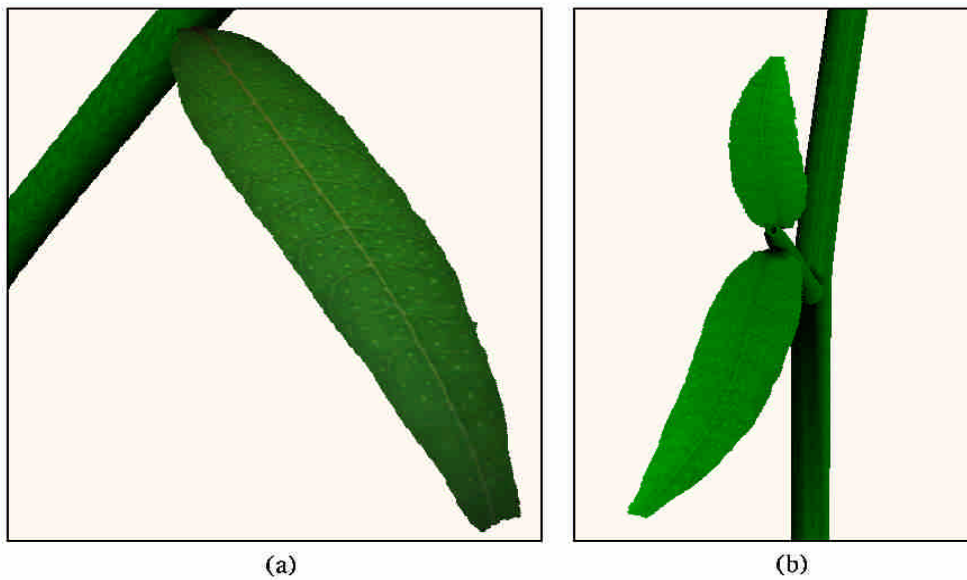


圖 5.19 紅刺蔥葉子成果圖，(a)、(b)是同株紅刺蔥不同位置上的葉子影像。

合檸檬桉葉子的模型而得到，圖 5.20 和圖 5.21 各是香甜肉桂和月桂的葉子影像成

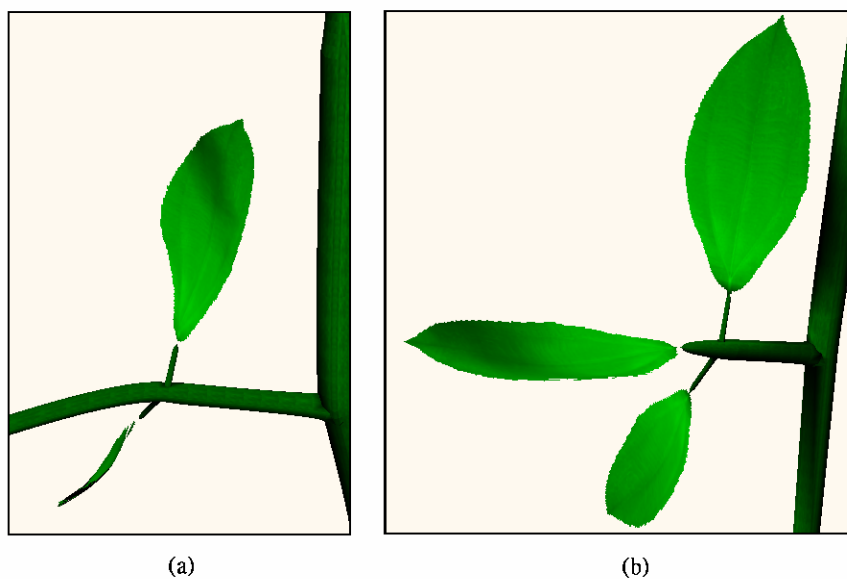


圖 5.20 香甜肉桂葉子成果圖，(a)、(b)是同株香甜肉桂不同位置上的葉子影像。

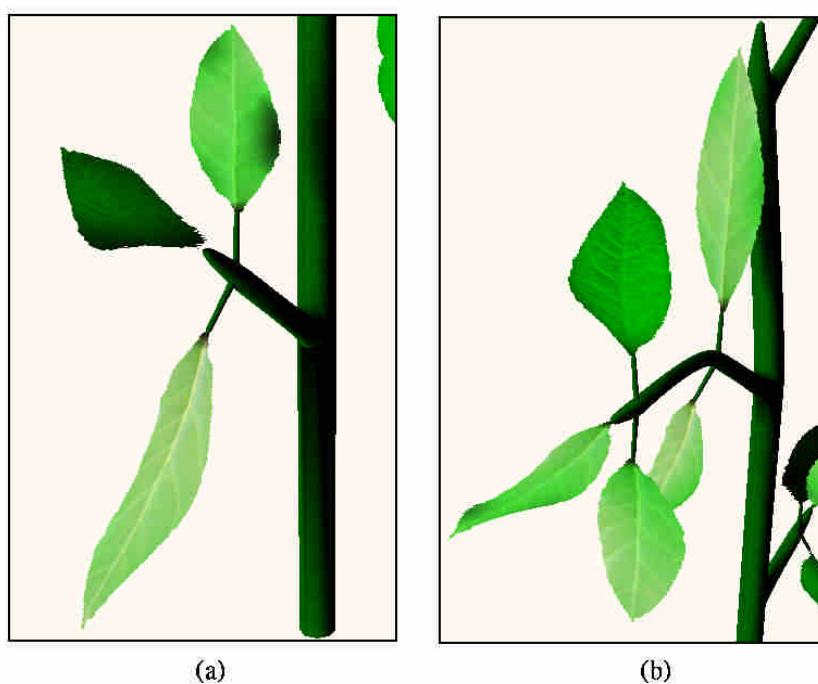


圖 5.21 月桂葉子成果圖，(a)、(b)是同一株月桂不同位置上的葉子影像。

果圖，以下的各圖是揭示 PRT 使用在 SimEco 上的各種效果的影像，下圖 5.22 和 5.23 的光影變化是根據光源由左至右移動，並以 PRT 即時運算得到的，圖 5.22 是玫瑰的葉子在不同方向的入射光下的光影變化，而圖 5.23 是日日春的葉子在不同入射方

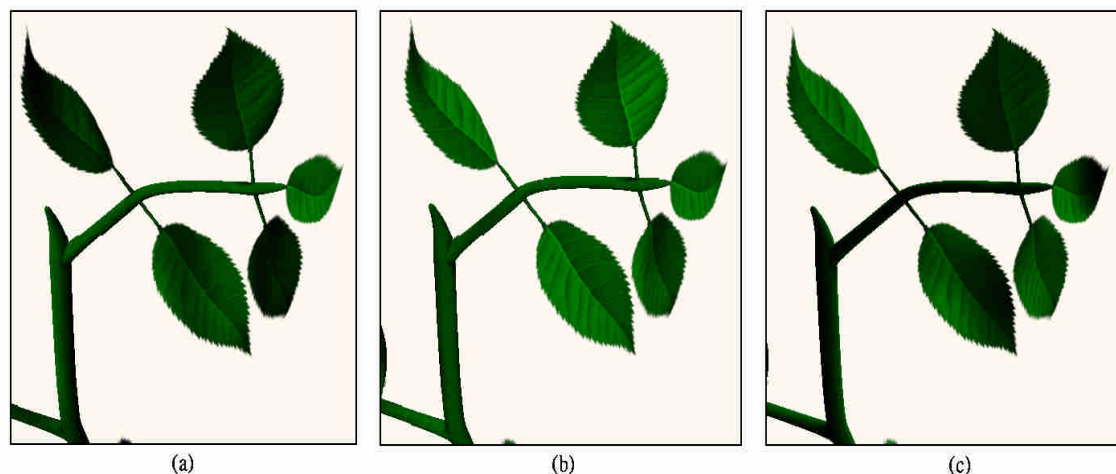


圖 5.22 玫瑰光影變化圖，(a)、(b)、(c)是同一株玫瑰在不同時間點下的局部玫瑰影像，其中光影變化是依據光源由左至右而得知，並以 PRT 計算而得。

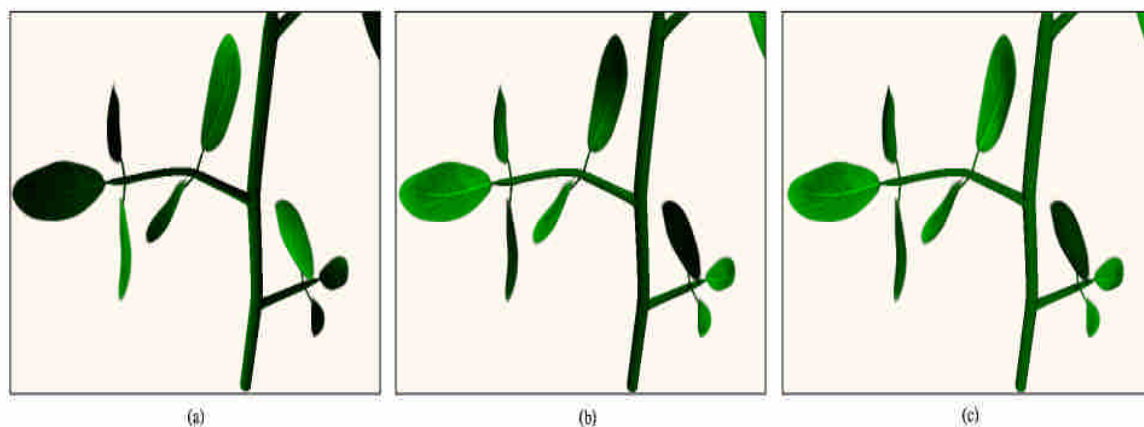


圖 5.23 日日春光影變化圖，(a)、(b)、(c)上的光影變化是依據光源由左至右而得知，並以 PRT 計算而得。

向光源下的光影變化，圖 5.24 則顯示 PRT 中陰影的呈現和變化，(a)、(b)為同樣兩

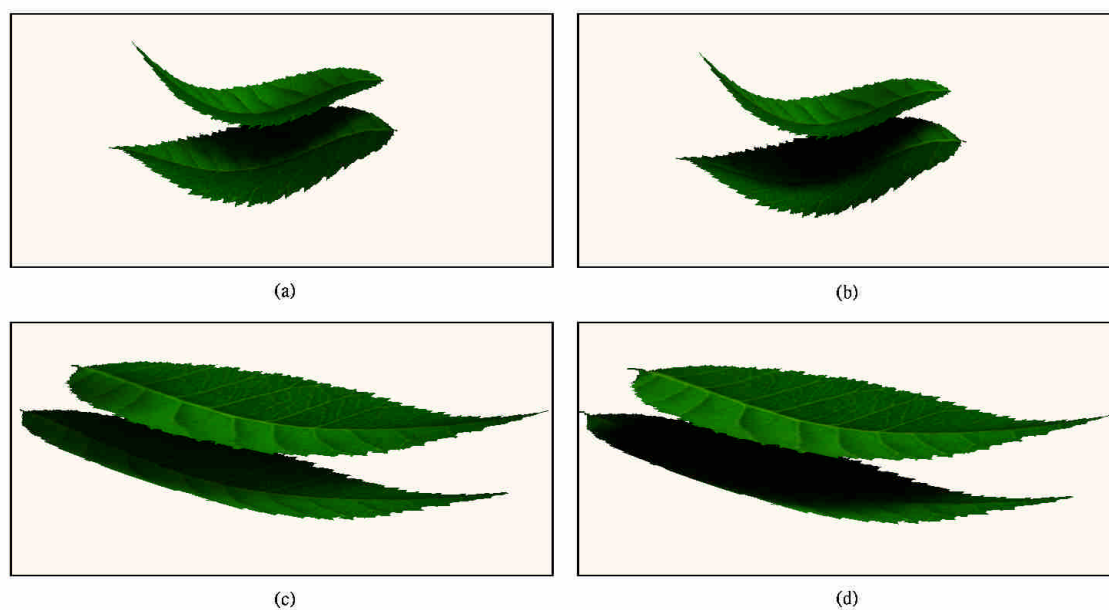


圖 5.24 PRT 陰影概念圖

片葉子在不同入射方向光源下的陰影效果，而(c)、(d)為另外的兩片葉子在不同入射方向光源下的陰影變化效果，最後完整的 PRT 成果圖如圖 5.25、圖 5.26、圖 5.27，在圖 5.25、圖 5.26、圖 5.27 中的玫瑰是以 64410 polygons、169 alphabets 以及經過 16 次疊代所獲得的結果。



圖 5.25 玫瑰的完成圖(1)



圖 5.26 玫瑰的完成圖(2)



圖 5.27 玫瑰的完成圖(3)

第六章 結論

在目前的虛擬實境發展上，植物已經是不可或缺的一環，因此增加植物的真實度將可以大大的提升場景的真實感，且對於研究植物生長和環境互動的學者來說，如果能發展一套所知即所見的植物模擬系統將可以減少植物研究的時程，因此發展出相對應的系統可以確定是一件極為重要的課題，然而要開發如此的系統並不是簡單的事情，不但要了解植物的分佈狀態，如：水源的影響、光源的影響、重力的影響...等，單單上述所提的幾項就是一門專業的課題，更遑論要產生擬真的影像還需要有更先進以及快速的光源模擬理論。

雖然在 SimEco 中確實提高了植物影像的品質，然而在系統中依然有幾項待解決的問題，由於在系統中我們採用 PRT 的技術因此確實提高了影像的能見度，但由於目前的 PRT 技術是以無限遠的光源來模擬光源函數，因此雖然在距離相近的物體間確定有非常好的半影效果，然而距離相隔甚遠的物體間其陰影卻變得非常模糊，目前可能解決的方法可能採用類似 Wang[25]所提的 light-visibility convolution 的方法來解決。

而物體間彼此的碰撞問題也非常的明顯因此急待我們去解決，這類的問題在碰撞偵測(collision detection)的領域已經有很多明確的解法了，因此我們目前的想法是以區域偵測的方式將植物的架構分割成多層次的階層式架構，再輔以現成的 AABB 或 OBB 方法去解決。

至於如何模擬出植物受環境的影響所產生出來的影像，確實也是目前我們系統中不足的地方，但當前的學者關於這方面問題的探討還僅止於猜測或試探性的解法並沒有所謂一般解的出現，因此關於這方面我們目前要做的還是多方的去翻閱、參考既有的文獻。

最後由於目前我們所採用的 PRT 技術只有散光的特性，因此如何處理

BRDF/BTDF 的問題也是我們所要思考的，然而目前關於 BRDF 的取得不是由相似的無機物獲得，就是由實驗數據所導出的近似公式中得到，而我們目前的想法是打算以實驗數據為基礎的近似公式來解決 BRDF/BTDF 的問題。

上述的四項議題就是目前我們覺得在我們系統中比較欠缺的部分，且目前我們的系統對於大量植物的模擬仍有問題，因此我們目前先著重在單株影像的生成，至於多株植物間的模擬將留待未來再做處理。

參考文獻

- [1] G. V. G. Baranoski and J. G. Rokne. *An algorithmic reflectance and transmittance model for plant tissue*. Computer Graphics Forum, Volume 16, Issue 3, pages 141 – 150, 1997.
- [2] G. V. G. Baranoski and J. G. Rokne. *Efficiently simulating scattering of light by leaves*. The Visual Computer 17(8), pages 491 – 505, 2001.
- [3] F. Boudon, P. Prusinkiewicz, P. Federl, C. Godin and R. Karwowski. *Interactive design of bonsai tree models*. Computer Graphics Forum. Proceedings of Eurographics, Volume 22, Number 3, pages 591 – 599, September 2003.
- [4] O. Deussen, P. Hanrahan, B. Lintermann, R. Měch, M. Pharr, and P. Prusinkiewicz. *Realistic modeling and rendering of plant ecosystems*. Proceedings of the 25th annual conference on Computer graphics and interactive techniques, pages 275 – 286, July 1998.
- [5] O. Deussen, C. Colditz, M. Stamminger, and George Drettakis. *Interactive Visualization of Complex Plant Ecosystems*. Proceedings of the conference on Visualization '02, pages 219 – 226, October 2002.
- [6] R. Green. *Spherical Harmonic Lighting: The Gritty Details*. Game Developers' Conference, San Jose, CA, March 2003.
- [7] T. Ijiri, S. Owada, M. Okabe, and T. Igarashi. *Floral diagrams and inflorescences: interactive flower modeling using botanical structural constraints*. ACM Transactions on Graphics (TOG), Volume 24, Issue 3, pages 720 – 726, July 2005.
- [8] H. W. Jensen, S. R. Marschner, M. Levoy, and P. Hanrahan. *A practical model for subsurface light transport*. Proceedings of the 28th annual conference on Computer graphics and interactive techniques, pages 511 – 518, August 2001.
- [9] G. Kokai, Z. Toth, and R. Vanyi. *Evolving artificial trees described by parametric L-systems*. 1999 IEEE Canadian Conference on Electrical and Computer Engineering, Volume 3, pages 1722 – 1727, May 1999.
- [10] A. Lindenmayer and G. Rozenberg. *Developmental systems and languages*. Proceedings of the fourth annual ACM symposium on Theory of computing, pages 214 – 221, May 1972.
- [11] B. Lintermann and O. Deussen. *Interactive Modeling of Plants*. IEEE Computer Graphics and Applications, Volume 19, pages 56 – 65, January 1999.
- [12] F. E. Nicodemus, J. C. Richmond, J. J. Hsia, I. W. Ginsberg, and T. Limperis. *Geometric considerations and nomenclature for reflectance*. Monograph 161,

- National Bureau of Standards (US), October 1977.
- [13] J. L. Power, A.J. B. Brush, P. Prusinkiewicz, and D. H. Salesin. *Interactive arrangement of botanical L-system models*. Proceedings of the 1999 symposium on Interactive 3D graphics, pages 175 – 182, April 1999.
- [14] P. Prusinkiewicz, A. Lindenmayer, and J. Hanan. *Development models of herbaceous plants for computer imagery purposes*. ACM SIGGRAPH Computer Graphics, Proceedings of the 15th annual conference on Computer graphics and interactive techniques'88, Volume 22, Issue 4, pages 141 – 150, June 1988.
- [15] P. Prusinkiewicz, M. S. Hammel, and E. Mjolsness. *Animation of plant development*. Proceedings of the 20th annual conference on Computer graphics and interactive techniques, pages 351 – 360, September 1993.
- [16] P. Prusinkiewicz, L. M'undermann, R. Karwowski, and B. Lane. *The use of positional information in the modeling of plants*. Proceedings of the 28th annual conference on Computer graphics and interactive techniques, pages 289 – 300, August 2001.
- [17] R. Ramamoorthi and P. Hanrahan. *On the relationship between radiance and irradiance: Determining the illumination from images of a convex lambertian object*. Journal of the Optical Society of America (JOSA A), Volume 18, 10, pages 2448 – 2459, October 2001.
- [18] R. Ramamoorthi and P. Hanrahan. *An efficient representation for irradiance environment maps*. Proceedings of the 28th annual conference on Computer graphics and interactive techniques, pages 497 – 500, August 2001.
- [19] W. T. Reeves and R. Blau. *Approximate and probabilistic algorithms for shading and rendering structured particle systems*. ACM SIGGRAPH Computer Graphics, Proceedings of the 12th annual conference on Computer graphics and interactive techniques, Volume 19, Issue 3, pages 313 – 322, July 1985.
- [20] M. Renton, J. Hanan, and P. Kaitaniemi. *The inside story: including physiology in structural plant models*. Proceedings of the 1st international conference on Computer graphics and interactive techniques in Australasia and South East Asia, pages 95 – ff, February 2003.
- [21] A. Runions, M. Fuhrer, B. Lane, P. Federl, A. G. Rolland-Lagan, and P. Prusinkiewicz. *Modeling and visualization of leaf venation patterns*. ACM Transactions on Graphics (TOG), Volume 24, Issue 3, pages 702 – 711, July 2005.
- [22] P. P. Sloan, J. Kautz, and J. Snyder. *Precomputed radiance transfer for real-time rendering in dynamic, low-frequency lighting environments*. ACM Transactions on Graphics (TOG), Proceedings of the 29th annual conference on Computer graphics and interactive techniques SIGGRAPH '02, Volume 21, Issue 3, pages 527 – 536,

July 2002.

- [23] A. R. Smith. *Plants, fractals, and formal languages*. ACM SIGGRAPH Computer Graphics. Proceedings of the 11th annual conference on Computer graphics and interactive techniques, Volume 18, Issue 3, pages 1 – 10, January 1984.
- [24] C. Soler, F. X. Sillion, F. Blaise, and P. Dereffye. *An efficient instantiation algorithm for simulating radiant energy transfer in plant models*. ACM Transactions on Graphics (TOG), Volume 22, Issue 2, pages 204 – 233, April 2003.
- [25] L. Wang, W. Wang, J. Dorsey, X. Yang, B. Guo, and H. Y. Shum. *Real-time rendering of plant leaves*. ACM Transactions on Graphics (TOG), Volume 24 Issue 3, pages 712 – 719, July 2005.
- [26] A. Watt. *3D Computer Graphics*. Third Edition, Addison Wesley, U.S.A., 2000.